3GPP TSG-RAN WG2 Meeting #125 R2-240XXXX

Athens, Greece, Feb. 26th – Mar. 1st, 2024

Source: CATT

Title: [C123]Issue on Execution Condition of Subsequent CPAC

Agenda Item: 7.4.1.3.2

Document for: Discussion and Decision

# Introduction

In this contribution, we discuss RIL [C123], i.e., the execution condition evaluation issue in subsequent CPAC.

|  |
| --- |
| **[RIL]**: C123 **[Delegate]**: CATT (Rui) **[WI]**: Mob **[Class]**: 1 **[Status]**: ToDo **[TDoc]**: R2-24xxxxx **[Proposed Conclusion]**: v114**[Description]**: For MN-initiated subsequent CPAC, both condRRCReconfig and condExecutionCondSCG can be present  for one candidate cell in MCG VarConditionalReconfig. Only one of them should evaluated during the subsequent CPACBut according to the current spec, the UE will evaluate both the condRRCReconfig and condExecutionCondSCG if both are present, which is not intended.**[Proposed Change]**: UE should only performs the evaluation to the the valid execution condition evaluation when there are two execution conditions maintained for one candidate cell in MCG VarConditionalReconfig. |

# Discussion

**Background**

In subsequent CPAC, about execution condition, the following agreements were achieved in previous meeting and captured in current Rel-18 spec [1].

|  |
| --- |
| * For SN initiated inter-SN SCG selective activation, source SN generates the execution conditions for the initial CPC.
* For SN-initiated SCG selective activation, candidate SN generates execution conditions for subsequent CPC.
* Proposal 3 (option2): For MN-initiated subsequent CPAC, the execution condition configuration is provided as following:

MN generates the execution conditions (A4 event) for initial CPAC execution, and the measID refers to the measurement configuration associated with MCG;candidate SN generates the execution conditions (A3/A5 event) for subsequent CPC execution, and the measID refers to the measurement configuration associated with SCG. |

Based on the agreements above, the subsequent CPAC execution conditions were captured in [1] as follow:

* The legacy field *condExecutionCond-r16* is reused for initial execution conditions configuration for MN-initiated subsequent CPAC or for SN-initiated intra-SN subsequent CPAC configured via SRB3;
* The legacy field *condExecutionCondSCG-r17* is reused for initial execution conditions configuration for SN-initiated subsequent CPAC;
* Introduce a new field *subsequentCondReconfig-r18* for subsequent execution conditions configuration for MN/SN-initiated subsequent CPAC.

|  |
| --- |
| CondReconfigToAddMod-r16 ::= SEQUENCE { condReconfigId-r16 CondReconfigId-r16, condExecutionCond-r16 SEQUENCE (SIZE (1..2)) OF MeasId OPTIONAL, -- Need M condRRCReconfig-r16 OCTET STRING (CONTAINING RRCReconfiguration) OPTIONAL, -- Cond condReconfigAdd ..., [[ condExecutionCondSCG-r17 OCTET STRING (CONTAINING CondReconfigExecCondSCG-r17) OPTIONAL -- Need M ]], [[ condExecutionCondPSCell-r18 SEQUENCE (SIZE (1..2)) OF MeasId OPTIONAL, -- Cond condReconfigCHO-WithSCG subsequentCondReconfig-r18 SubsequentCondReconfig-r18 OPTIONAL, -- Need M securityCellSetId-r18 SecurityCellSetId-r18 OPTIONAL, -- Need M scpac-ConfigComplete-r18 ENUMERATED {true} OPTIONAL -- Cond CPAC ]]} |

**[C123] Issue in MN-initiated subsequent CPAC**

We elaborate the issue in a procedure manner to a UE configured with MN-initiated subsequent CPAC.

*Step 1:* UE is configured with MN-initiated subsequent CPAC. For a candidate, the initial execution condition is in condExecutionCond for the candidate in MCG VarConditionalReconfig.

*Step 2:* Initial CPAC change is triggered, condExecutionCondSCG for the candidate in MCG VarConditionalReconfig is replaced with the one from the subsequentCondReconfig. But the initial execution condition is still kept in condExecutionCond. Both condExecutionCond and condExecutionCondSCG will be present for a candidate PSCell in the MCG VarConditionalReconfig.

|  |
| --- |
| 2> if the *RRCReconfiguration* message is applied due to a conditional reconfiguration execution and the *subsequentCondReconfig* is included in the entry in *VarConditionalReconfig* containing the *RRCReconfiguration* message:3> for each *condReconfigId* included in *condExecutionCondToAddModList* within *subsequentCondReconfig*:4> replace within *VarConditionalReconfig* the entry in *condExecutionCond* or *condExecutionCondSCG* with the matching *condReconfigId* value; |

**Observation 1: For MN-initiated subsequent CPAC, both *condExecutionCond* and *condExecutionCondSCG* can be stored for and associated with one candidate cell in MCG *VarConditionalReconfig*.**

*Step 3:* After the initial CPPAC procedure, UE is supposed to evaluate the execution condition in condExecutionCondSCG(i.e. the one from subsequentCondReconfig) for the subsequent CPAC evaluation. However, UE will evaluate both the execution conditions in condExecutionCond and condExecutionCondSCG of the candidate according to the current spec below,

|  |
| --- |
| 2> for each *measId* included in the *measIdList* within *VarMeasConfig* indicated in the *condExecutionCond,* *condExecutionCondSCG,* or *condExecutionCondPSCell* associated to *condReconfigId:*3> if the *condTriggerConfig* is not configured with *nesEvent*:4> if the *condEventId* is associated with *condEventT1*, and if the entry condition applicable for this event associated with the *condReconfigId*, i.e. the event corresponding with the *condEventId(s)* of the corresponding *condTriggerConfig* within *VarConditionalReconfig*, is fulfilled for the applicable cell; or……4> if the *condEventId* is associated with *condEventA3*, *condEventA4* or *condEventA5*, and if the entry condition(s) applicable for this event associated with the *condReconfigId*, i.e. the event corresponding with the *condEventId(s)* of the corresponding *condTriggerConfig* within *VarConditionalReconfig*, is fulfilled for the applicable cells for all measurements after layer 3 filtering taken during the corresponding *timeToTrigger* defined for this event within the *VarConditionalReconfig*:5> consider the event associated to that *measId* to be fulfilled; |

*Step 4:* As a result, a subsequent CPAC execution procedure may be wrongly triggered by the execution condition (i.e., the one in condExecutionCond) for initial CPAC.

**Observation 2: For MN-initiated subsequent CPAC, the subsequent CPAC procedure may be wrongly triggered by the execution condition (i.e., the one in condExecutionCond) for initial CPAC.**

In order to address the issue mentioned above, there are two solutions can be considered, as follows,

**Solution 1: For MN-initiated subsequent CPAC, UE choses the valid execution condition to evaluate for subsequent CPAC procedure. And UE removes the stored *condExecutionCondSCG* when UE performs reconfiguration with sync for SCG or SCG release.**

In solution 1, The UE should decide which execution condition to evaluate when there are two execution conditions maintained for one candidate cell in MCG VarConditionalReconfig,e.g., if UE has valid SCG configuration, the UE will evaluate the candidate PSCell based on the condExecutionCondSCG.

But if UE always evaluates the *condExecutionCondSCG* when two execution conditions maintained for one candidate cell, it should ensure that the *condExecutionCondSCG* is valid execution condition. Considering that in some cases the SN-configured execution condition (i.e., *condExecutionCondSCG*) is invalid, e.g.,

* for legacy PSCell change/CPC execution, it does not support the target SN to provide valid *condExecutionCondSCG* in the RRCReconfiguration which includes the reconfiguration with sync for target SN, so the stored *condExecutionCondSCG* should be invalid due to it is associated with the *measConfig* configured by the source SCG. Hence the stored *condExecutionCondSCG* should be removed, furthermore, the *condExecutionCondSCG* can be removed also and applied the new SN-configured execution condition after successful subsequent CPAC execution;
* for SCG release, due to no SCG measConfig is stored, the *condExecutionCondSCG* should be removed;
* for SCG failure, the network will release the SCG or inform UE to perform PSCell change, then the *condExecutionCondSCG* will be removed after SCG release or reconfiguration with sync SCG as above discussion, and anyway the execution condition evaluation is stopped upon SCG failure even if the network has no response message for the SCG failure.

Based on above, if UE releases the *condExecutionCondSCG* upon UE performs reconfiguration with sync for SCG or SCG release, it will ensure the UE will not store invalid *condExecutionCondSCG*.

There is another issue on execution condition for subsequent CPA is raised in RIL [V136], i.e., the initial execution condition in condExecutionCond could be used for later subsequent CPA.

With solution 1, the initial execution condition in condExecutionCond is kept, and UE could continue to perform subsequent CPA based on the stored condExecutionCond after the SCG is released. So RIL [V136] can be addressed by solution 1. A TP is provided in Annex 1.

**Solution 2: For MN-initiated subsequent CPAC, the initial execution condition (i.e., the one in condExecutionCond is removed after the initial CPAC procedure).**

In solution 2, the UE needs to release the execution condition configured by MN (i.e., *condExecutionCond*) which is invalid execution condition for subsequent CPC when successful subsequent CPAC execution. A TP is provided in Annex 2.

For the solutions listed above, solution 2 is simpler, and solution 1 is more complex but it can address the CPA issue indicated in RIL [V136].We need further discussion to do the down selection between the solutions.

Therefore, it is proposed that,

**Proposal 1: For Execution Condition issue in MN-initiated subsequent CPAC, RAN2 down selects between the following solutions,**

* **Solution 1: For MN-initiated subsequent CPAC, UE choses the valid execution condition to evaluate for subsequent CPAC procedure. And UE removes the stored *condExecutionCondSCG* when UE performs reconfiguration with sync for SCG or SCG release.TP in annex 1 is adopted.**
* **Solution 2: For MN-initiated subsequent CPAC, the initial execution condition (i.e., the one in condExecutionCond is removed after the initial CPAC procedure).TP in annex 2 is adopted.**

# Conclusion

Based on the previous analysis in section 2, our proposals are summarized as follows:

**Proposal 1: For Execution Condition issue in MN-initiated subsequent CPAC, RAN2 down selects between the following solutions,**

* **Solution 1: For MN-initiated subsequent CPAC, UE choses the valid execution condition to evaluate for subsequent CPAC procedure. And UE removes the stored *condExecutionCondSCG* when UE performs reconfiguration with sync for SCG or SCG release.TP in annex 1 is adopted.**
* **Solution 2: For MN-initiated subsequent CPAC, the initial execution condition (i.e., the one in condExecutionCond is removed after the initial CPAC procedure).TP in annex 2 is adopted.**

# Reference

[1] 38.331-i00

Annex 1 TP for solution 1

The text proposal in this section is written based on the 38.331-i00 spec [1].

 *START OF CHANGE*

#### 5.3.5.3 Reception of an *RRCReconfiguration* by the UE

……

2> if the *reconfigurationWithSync* was included in *spCellConfig* of an MCG:

3> if T390 is running:

4> stop timer T390 for all access categories;

4> perform the actions as specified in 5.3.14.4.

3> if T350 is running:

4> stop timer T350;

3> if *RRCReconfiguration* does not include *dedicatedSIB1-Delivery* and

3> if the active downlink BWP, which is indicated by the *firstActiveDownlinkBWP-Id* for the target SpCell of the MCG, has a common search space configured by *searchSpaceSIB1*:

4> acquire the *SIB1*, which is scheduled as specified in TS 38.213 [13], of the target SpCell of the MCG;

4> upon acquiring *SIB1*, perform the actions specified in clause 5.2.2.4.2;

2> if the *reconfigurationWithSync* was included in *spCellConfig* of an SCG:

 3> remove the *condExecutionCondSCG* in the entry which *subsequentCondReconfig* is included within MCG *VarConditionalReconfig*;

2> if the *RRCReconfiguration* message is applied due to a conditional reconfiguration execution and the *subsequentCondReconfig* is included in the entry in *VarConditionalReconfig* containing the *RRCReconfiguration* message:

3> for each *condReconfigId* included in *condExecutionCondToAddModList* within *subsequentCondReconfig*:

4> replace within *VarConditionalReconfig* the entry in *condExecutionCond* or *condExecutionCondSCG* with the matching *condReconfigId* value;

……

 *NEXT OF CHANGE*

#### 5.3.5.4 Secondary cell group release

The UE shall:

1> as a result of SCG release triggered by E-UTRA (i.e. (NG)EN-DC case) or NR (i.e. NR-DC case):

2> reset SCG MAC, if configured;

2> for each RLC bearer that is part of the SCG configuration:

3> perform RLC bearer release procedure as specified in 5.3.5.5.3;

2> for each BH RLC channel that is part of the SCG configuration:

3> perform BH RLC channel release procedure as specified in 5.3.5.5.10;

2> release the SCG configuration;

2> for all application layer measurement configurations that are part of the SCG configuration:

3> inform upper layers about the release of the application layer measurement configurations;

3> discard any application layer measurement reports which were not yet submitted to lower layers for transmission;

2> remove all the entries within the SCG *VarConditionalReconfig*, if any;

2> if SCG release was triggered by NR (i.e. NR-DC case):

3> remove all the entries in the *condReconfigList* within the MCG *VarConditionalReconfig* for which the *RRCReconfiguration* within *condRRCReconfig* does not include the *masterCellGroup* with *reconfigurationWithSync* and for which *subsequentCondReconfig* is not present, if any;

3> remove the *condExecutionCondSCG* in the entry which *subsequentCondReconfig* is included within MCG *VarConditionalReconfig*;

2> else (i.e. EN-DC case):

3> perform *VarConditionalReconfiguration* CPC removal as specified in TS 36.331 [10] clause 5.3.5.9.7;

2> stop timer T310 for the corresponding SpCell, if running;

2> stop timer T312 for the corresponding SpCell, if running;

2> stop timer T304 for the corresponding SpCell, if running.

NOTE: Release of cell group means only release of the lower layer configuration of the cell group but the *RadioBearerConfig* may not be released.

 *NEXT OF CHANGE*

##### 5.3.5.13.4 Conditional reconfiguration evaluation

The UE shall:

1> for each *condReconfigId* within the *VarConditionalReconfig*:

2> if the *RRCReconfiguration* within *condRRCReconfig* includes the *masterCellGroup* including the *reconfigurationWithSync*:

3> if the associated *condExecutionCondPSCell* is configured:

4> consider the cell which has a physical cell identity matching the value indicated in the *ServingCellConfigCommon* included in the *reconfigurationWithSync* within the *masterCellGroup* in the received *condRRCReconfig* to be applicable cell; and

4> consider the cell which has a physical cell identity matching the value indicated in the *ServingCellConfigCommon* included in the *reconfigurationWithSync* within the *secondaryCellGroup* within the *nr-SCG* within the received *condRRCReconfig* to be applicable cell;

3> else:

4> consider the cell which has a physical cell identity matching the value indicated in the *ServingCellConfigCommon* included in the *reconfigurationWithSync* within the *masterCellGroup* in the received *condRRCReconfig* to be applicable cell;

2> else if the *RRCReconfiguration* within *condRRCReconfig* includes the *secondaryCellGroup* including the *reconfigurationWithSync*:

3> if the cell which has a physical cell identity matching the value indicated in the *ServingCellConfigCommon* included in the *reconfigurationWithSync* within the *secondaryCellGroup* within the received *condRRCReconfig* is not the PSCell:

4> consider the cell to be applicable cell;

2> if *condExecutionCondSCG* is configured:

3> in the remainder of the procedure, consider each *measId* indicated in the *condExecutionCondSCG* as a *measId* in the *VarMeasConfig* associated with the SCG *measConfig*;

2> if the *condExecutionCondPSCell* is configured:

3> in the remainder of the procedure, consider each *measId* indicated in the *condExecutionCondPSCell* as a *measId* in the *VarMeasConfig* associated with the MCG *measConfig*;

2> if *condExecutionCond* is configured:

3> if it is configured via SRB3 or configured within *nr-SCG* or within *nr-SecondaryCellGroupConfig* (specified in TS 36.331[10]) via SRB1:

4> in the remainder of the procedure, consider each *measId* indicated in the *condExecutionCond* as a *measId* in the *VarMeasConfig* associated with the SCG *measConfig*;

3> else:

4> in the remainder of the procedure, consider each *measId* indicated in the *condExecutionCond* as a *measId* in the *VarMeasConfig* associated with the MCG *measConfig*;

2> if both *condExecutionCond* and *condExecutionCondSCG* are configured for one candidate configuration, and the candidate configuration is configured for subsequent CPAC:

 3> in the remainder of the procedure, consider *condExecutionCondSCG* as the evaluation condition;

2> for each *measId* included in the *measIdList* within *VarMeasConfig* indicated in the *condExecutionCond,* *condExecutionCondSCG,* or *condExecutionCondPSCell* associated to *condReconfigId:*

3> if the *condTriggerConfig* is not configured with *nesEvent*:

4> if the *condEventId* is associated with *condEventT1*, and if the entry condition applicable for this event associated with the *condReconfigId*, i.e. the event corresponding with the *condEventId(s)* of the corresponding *condTriggerConfig* within *VarConditionalReconfig*, is fulfilled for the applicable cell; or

4> if the *condEventId* is associated with *condEventD1*, and if the entry conditions applicable for this event associated with the *condReconfigId*, i.e. the event corresponding with the *condEventId(s)* of the corresponding *condTriggerConfig* within *VarConditionalReconfig*, is fulfilled for the applicable cell during the corresponding *timeToTrigger* defined for this event within the *VarConditionalReconfig*; or

4> if the *condEventId* is associated with *condEventD2*, and if the entry conditions applicable for this event associated with the *condReconfigId*, i.e., the event corresponding with the *condEventId(s)* of the corresponding *condTriggerConfig* within *VarConditionalReconfig*, is fulfilled for the applicable cell during the corresponding *timeToTrigger* defined for this event within the *VarConditionalReconfig*; or

4> if the *condEventId* is associated with *condEventA3*, *condEventA4* or *condEventA5*, and if the entry condition(s) applicable for this event associated with the *condReconfigId*, i.e. the event corresponding with the *condEventId(s)* of the corresponding *condTriggerConfig* within *VarConditionalReconfig*, is fulfilled for the applicable cells for all measurements after layer 3 filtering taken during the corresponding *timeToTrigger* defined for this event within the *VarConditionalReconfig*:

5> consider the event associated to that *measId* to be fulfilled;

……

 *NEXT OF CHANGE*

*– CondReconfigToAddModList*

The IE *CondReconfigToAddModList* concerns a list of conditional reconfigurations to add or modify, with for each entry the *condReconfigId* and the associated fields.

***CondReconfigToAddModList* information element**

-- ASN1START

-- TAG-CONDRECONFIGTOADDMODLIST-START

CondReconfigToAddModList-r16 ::= SEQUENCE (SIZE (1.. maxNrofCondCells-r16)) OF CondReconfigToAddMod-r16

CondReconfigToAddMod-r16 ::= SEQUENCE {

 condReconfigId-r16 CondReconfigId-r16,

 condExecutionCond-r16 SEQUENCE (SIZE (1..2)) OF MeasId OPTIONAL, -- Need M

 condRRCReconfig-r16 OCTET STRING (CONTAINING RRCReconfiguration) OPTIONAL, -- Cond condReconfigAdd

 ...,

 [[

 condExecutionCondSCG-r17 OCTET STRING (CONTAINING CondReconfigExecCondSCG-r17) OPTIONAL -- Need M

 ]],

 [[

 condExecutionCondPSCell-r18 SEQUENCE (SIZE (1..2)) OF MeasId OPTIONAL, -- Cond condReconfigCHO-WithSCG

 subsequentCondReconfig-r18 SubsequentCondReconfig-r18 OPTIONAL, -- Need M

 securityCellSetId-r18 SecurityCellSetId-r18 OPTIONAL, -- Need M

 scpac-ConfigComplete-r18 ENUMERATED {true} OPTIONAL -- Cond CPAC

 ]]

}

CondReconfigExecCondSCG-r17 ::= SEQUENCE (SIZE (1..2)) OF MeasId

SubsequentCondReconfig-r18 ::= SEQUENCE {

 condExecutionCondToReleaseList-r18 CondExecutionCondToReleaseList-r18 OPTIONAL, -- Need N

 condExecutionCondToAddModList-r18 CondExecutionCondToAddModList-r18 OPTIONAL, -- Need N

 ...

}

CondExecutionCondToAddModList-r18 ::= SEQUENCE (SIZE (1.. maxNrofCondCells-r16)) OF CondExecutionCondToAddMod-r18

CondExecutionCondToAddMod-r18 ::= SEQUENCE {

 condReconfigId-r18 CondReconfigId-r16,

 condExecutionCond-r18 SEQUENCE (SIZE (1..2)) OF MeasId OPTIONAL, -- Need M

 condExecutionCondSCG-r18 OCTET STRING (CONTAINING CondReconfigExecCondSCG-r17) OPTIONAL, -- Need M

 ...

}

CondExecutionCondToReleaseList-r18 ::= SEQUENCE (SIZE (1.. maxNrofCondCells-r16)) OF CondReconfigId-r16

-- TAG-CONDRECONFIGTOADDMODLIST-STOP

-- ASN1STOP

| ***CondReconfigToAddMod* field descriptions** |
| --- |
| ***condExecutionCond***The execution condition that needs to be fulfilled in order to trigger the execution of a conditional reconfiguration for CHO, CPA, intra-SN CPC without MN involvement, MN initiated inter-SN CPC, or SN initiated intra-SN subsequent CPAC without MN involvement. When configuring 2 triggering events (Meas Ids) for a candidate cell, the network ensures that both refer to the same *measObject.* The network configures at most one from *condEventD1, condEventD2* or *condEventT1* for the same candidate cell. For CPA and for MN-initiated inter-SN CPC, the network only indicates *MeasId*(s) associated with *condEventA4*. For intra-SN CPC and intra-SN subsequent CPAC, the network only indicates *MeasId*(s) associated with *condEventA3* or *condEventA5*. |
| ***condExecutionCondPSCell***The execution condition that needs to be fulfilled for the associated PSCell in order to trigger the execution of a conditional reconfiguration for CHO with candidate SCG(s). The Meas Ids refer to the *measConfig* associated with the MCG. When configuring 2 triggering events (Meas Ids) for a candidate cell, network ensures that both refer to the same *measObject*. The network only indicates *MeasId(s)* associated with condEventA4. |
| ***condExecutionCondSCG***Contains execution condition that needs to be fulfilled in order to trigger the execution of a conditional reconfiguration for SN initiated inter-SN CPC, SN initiated inter-SN subsequent CPAC, SN initiated intra-SN subsequent CPAC with MN involvement, or MN initiated inter-SN subsequent CPAC. The Meas Ids refer to the *measConfig* associated with the SCG. When configuring 2 triggering events (Meas Ids) for a candidate cell, network ensures that both refer to the same *measObject*. For each *condReconfigId*, the network always configures either *condExecutionCond* or *condExecutionCondSCG* (not both except in subsequent CPAC). *CondExecutionCondSCG* is used as the subsequent execution condition for conditional reconfiguration evaluation of subsequent CPAC. The network only indicates *MeasId*(s) associated with *condEventA3* or *condEventA5*. |
| ***condRRCReconfig***The *RRCReconfiguration* message to be applied when the condition(s) are fulfilled. The *RRCReconfiguration* message contained in *condRRCReconfig* cannot contain the field *conditionalReconfiguration* or the field *daps-Config*. |
| ***scpac-ConfigComplete***This field indicates whether the configuration contained in *condRRCReconfig* for subsequent CPAC is a complete configuration. |
| ***subsequentCondReconfig***Contains the execution conditions that need to be fulfilled in order to trigger the execution of a subsequent CPAC. If the field is configured, the configuration of candidate PSCells for subsequent CPAC is supported. The subsequent execution condition is used for conditional reconfiguration evaluation for other candidate cells when the *RRCReconfiguration* message contained in *condRRCReconfig* has been applied. |

 *END OF CHANGE*

Annex 2 TP for solution 2

The text proposal in this section is written based on the 38.331-i00 spec [1].

 *START OF CHANGE*

#### 5.3.5.3 Reception of an *RRCReconfiguration* by the UE

……

2> if the *RRCReconfiguration* message is applied due to a conditional reconfiguration execution and the *subsequentCondReconfig* is included in the entry in *VarConditionalReconfig* containing the *RRCReconfiguration* message:

3> if the *RRCReconfiguration* message is included in the MCG *VarConditionalReconfig*:

4> remove the *condExecutionCond* in the entry which *subsequentCondReconfig* is included within the MCG *VarConditionalReconfig*;

3> for each *condReconfigId* included in *condExecutionCondToAddModList* within *subsequentCondReconfig*:

4> replace within *VarConditionalReconfig* the entry in *condExecutionCond* or *condExecutionCondSCG* with the matching *condReconfigId* value;

2> if the *reconfigurationWithSync* was included in *spCellConfig* of an MCG; or

2> if the *reconfigurationWithSync* was included in *spCellConfig* of an SCG and the CPA, CPC, or subsequent CPAC was configured:

3> remove all the entries in the *condReconfigList* within the MCG and the SCG *VarConditionalReconfig* except for the entries in which *subsequentCondReconfig* is present, if any;

3> remove all the entries within *VarConditionalReconfiguration* as specified in TS 36.331 [10], clause 5.3.5.9.6, if any;

3> for each *measId* of the MCG *measConfig*, if configured, and for each *measId* of the SCG *measConfig*, if configured, if the associated *reportConfig* has a *reportType* set to *condTriggerConfig*:

4> if the *reportConfigId* is not associated with any *measId* indicated by the *condExecutionCond* or the *condExecutionCondSCG* in an entry of *condReconfigList* in *VarConditionalReconfig* in which *subsequentCondReconfig* is included:

5> remove the entry with the matching *reportConfigId* from the *reportConfigList* within the *VarMeasConfig*;

4> if the associated *measObjectId* is only associated to a *reportConfig* with *reportType* set to *condTriggerConfig*; and

4> if the *measObjectId* is not associated with any *measId* indicated by the *condExecutionCond* or the *condExecutionCondSCG* in an entry of *condReconfigList* in *VarConditionalReconfig* in which *subsequentCondReconfig* is included:

5> remove the entry with the matching *measObjectId* from the *measObjectList* within the *VarMeasConfig*;

4> remove the entry with the matching *measId* from the *measIdList* within the *VarMeasConfig*;

……

 *END OF CHANGE*