**3GPP TSG-RAN WG1 Meeting #102-e R1-200xxxx**

**e-Meeting, August 17th – 28th, 2020**

**Agenda Item: 7.2.1**

**Source: Moderator (ZTE)**

**Title: Email discussion #2 for 2-step RACH**

**Document for: Discussion**

# Introduction

This document is intended to address the following remaining issues by email discussion.

[102-e-NR-2step-RACH-02] Email discussion/approval of addressing issues #2, #7.2, #8 as in the summary:

* Modulation order of MsgB PDSCH
* Default TDRA table for extended CP
* Resource overhead of MsgA PUSCH

By 8/20, with follow-up potential CR(s) by 8/25 – Li (ZTE)

# Modulation order of MsgB PDSCH (issue #2)

R1-2005605 proposed to limit the modulation order of MsgB PDSCH, similar to the principle of Msg2 PDSCH scheduled with RA-RNTI.

***Proposal 1:***

* Adopt the TP#1 in 38.214, to limit the modulation order of MsgB PDSCH.

**Reasons for change**

To limit the modulation order of MsgB PDSCH, similar to the PDSCH scheduled with RA-RNTI

**Summary of changes**

Implement the above update

**Specs/Sections impacted**

TS 38.214, Section 5.1.3.1

-------------------------**Text proposal #1 starts for TS 38.214** ----------------------------

5.1.3.1 Modulation order and target code rate determination

<Unchanged Text Omitted>

The UE is not expected to decode a PDSCH scheduled with P-RNTI, RA-RNTI, msgB-RNTI, SI-RNTI and *Qm* > 2

<Unchanged Text Omitted>

------------------------- **Text proposal #1 ends for TS 38.214** -------------------------------

Any comments?

|  |  |
| --- | --- |
| Company | Comments |
| Nokia | OK for TP#1 |
| Intel | It is not clear to us why we need to restrict MsgB PDSCH with QPSK. In our view, MsgB may include RRC message, which is similar to Msg4 and can have large payload size. In this case, higher modulation order can be used to reduce amount of resource for MsgB PDSCH.  Hence, we do not need this TP. |
| Qualcomm | We are ok to support TP#1.  Similar to msg2 PDSCH, msgB PDSCH is scheduled by a group-common DCI and is expected to be decoded by multiple UEs monitoring a same msgB-RNTI. Typically, the payload size of msgB PDSCH is larger than that of msg2 PDSCH. To ensure the reliability of demodulation and decoding, it makes sense to restrict the modulation order in a similar way as msg2 PDSCH. |

# Default TDRA table for extended CP (issue #7.2)

R1-2006609 proposed to capture the default TDRA table 6.1.2.1.1-3 for extended CP for MsgA PUSCH, since both normal CP and extended CP are expected be supported for MsgA PUSCH.

***Proposal 2:***

* Adopt the TP#2 in 38.213, to capture the default TDRA table of extended CP for MsgA PUSCH.

**Reasons for change**

To capture the default TDRA table 6.1.2.1.1-3 of extended CP for MsgA PUSCH

**Summary of changes**

Implement the above updates

**Specs/Sections impacted**

TS 38.213 Section 8.1A

-------------------------**Text proposal #2 starts for TS 38.213** ----------------------------

8.1A PUSCH for Type-2 random access procedure

<Unchanged Text Omitted>

If a UE does not have dedicated RRC configuration, or has an initial UL BWP as an active UL BWP, or is not provided *startSymbolAndLengthMsgA-PO*, *msgA-PUSCH-timeDomainAllocation* provides a SLIV and a PUSCH mapping type for a PUSCH transmission by indicating

- first *maxNrofUL-Allocations* values from *PUSCH-TimeDomainResourceAllocationList*, if *PUSCH-TimeDomainResourceAllocationList* is provided in *PUSCH-ConfigCommon*

- entries from table 6.1.2.1.1-2 for normal CP or table 6.1.2.1.1-3 for extended CP in [6, TS 38.214] according to the higher layer parameter *cyclicPrefix*, if *PUSCH-TimeDomainResourceAllocationList* is not provided in *PUSCH-ConfigCommon*

<Unchanged Text Omitted>

------------------------- **Text proposal #2 ends for TS 38.213** -------------------------------

Any comments?

|  |  |
| --- | --- |
| Company | Comment |
| Nokia | OK for TP#2 |
| Intel | We are fine with the TP |
| Qualcomm | No need to have TP#2. The wording in current release is clear enough. |

# Resource overhead of MsgA PUSCH (issue #7.1)

R1-2006609 proposed to capture the same assumption of Msg3 resource overhead for MsgA. The resource overhead per PRB is assumed to be zero for Msg3, which should also be applied for MsgA PUSCH

***Proposal 3:***

* Adopt the TP#3 in 38.214, to capture the assumption of resource overhead for MsgA.

**Reasons for change**

To capture the same assumption of Msg3 resource overhead for MsgA

**Summary of changes**

Implement the above updates

**Specs/Sections impacted**

TS 38.214 Section 6.1.4.2

-------------------------**Text proposal #3 starts for TS 38.214** ----------------------------

#### 6.1.4.2 Transport block size determination

<Unchanged Text Omitted>

The UE shall first determine the number of REs (*NRE*) within the slot:

- A UE first determines the number of REs allocated for PUSCH within a PRB  by

- , where is the number of subcarriers in the frequency domain in a physical resource block,  is the number of symbols *L* of the PUSCH allocation according to Clause 6.1.2.1 for scheduled PUSCH or Clause 6.1.2.3 for configured PUSCH,  is the number of REs for DM-RS per PRB in the allocated duration including the overhead of the DM-RS CDM groups without data, as described for PUSCH with a configured grant in Clause 6.1.2.3 or as indicated by DCI format 0\_1 or DCI format 0\_2 or as described for DCI format 0\_0 in Clause 6.2.2, and  is the overhead configured by higher layer parameter *xOverhead* in*PUSCH-ServingCellConfig*. If the  is not configured (a value from 6, 12, or 18), the  is assumed to be 0. For Msg3 or MsgA PUSCH transmission the  is always set to 0. In case of PUSCH repetition Type B,  is determined assuming a nominal repetition with the duration of *L* symbols without segmentation.

<Unchanged Text Omitted>

------------------------- **Text proposal #3 ends for TS 38.214** -------------------------------

Any comments?

|  |  |
| --- | --- |
| Company | Comment |
| Nokia | OK for TP#3 |
| Intel | We are fine with the TP |
| Qualcomm | TP#3 looks good to us. |

# Summary

The final proposals and the potential CRs are to be updated…