UNIVERSITY
OF OULU

FACULTY OF INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING

Markus Kyllönen

DEVELOPMENT OF AN AUTOMATED DEBUG LOG PROCESSING SYSTEM FOR CSI MECHANISM

Master's Thesis
Degree Programme in Computer Science and Engineering
December 2022

Kyllönen M. (2022) Development of an Automated Debug Log Processing System for CSI Mechanism. University of Oulu, Degree Programme in Computer Science and Engineering, 55 p.

ABSTRACT

In the development industry of mobile telecommunication devices, various challenges are faced regarding debugging different errors. One significant difficulty in user equipment (UE) development is the manual processing of massive debug logs of several gigabytes in size. A part of this manual processing may be handled by the automated system, freeing up employees' time for other assignments, resulting in increased efficiency. The work of this thesis is conducted from the perspective of the channel state information (CSI) mechanism in Fifth Generation Mobile Network New Radio (5G NR) access technology. The purpose of this thesis is to present an example of automated CSI log processing system, and to this end, present basics of CSI mechanism and its standards. The main focus of this thesis is to verify that CSI reporting follows standards of technical specifications provided by the 3rd Generation Partnership Project (3GPP). The presented system contains two major functionalities for verifying the correctness of CSI reporting. The first functionality verifies that all expected CSI reports are transmitted successfully denoting that the processed debug log is not containing any unexpectedly dropped CSI reports. There are several fault reasons for untransmitted CSI reports, and one example is incorrect bit length calculations for CSI report fields, resulting in decreased performance of communication between the Network and UE. Regarding this, the second functionality of the system verifies that the corresponding UE software has calculated bit lengths of different quantities correctly according to 3GPP specifications. These faults are faced in the development phase of devices, for example, when developing new features. As a result of this thesis, a fairly significant system is successfully developed to speed up the manual debugging process. Especially in larger debug log cases, the system provides the ability to investigate the whole log remarkably fast compared to manual debugging.

Keywords: 3GPP, 5G NR, 5G Physical Layer, L1

Kyllönen M. (2022) Automaattisen virheenkorjaus lokijärjestelmän kehitys CSI-mekanismille. Oulun yliopisto, Tietotekniikan tutkinto-ohjelma, 55 s.

TIIVISTELMÄ

Matkaviestintälaitteiden kehitysteollisuudessa kohdataan monenlaisia haasteita virheiden korjaukseen liittyen. Yksi merkittävä haaste käyttäjän laitteiden kehityksessä on manuaalinen lokien prosessointi, erityisesti kooltaan suurien ja useiden gigatavujen lokien tapauksissa. Osa tästä prosessista voitaisiin hoitaa automaattisella järjestelmällä vapauttaen virheistä vastuulla olevien työntekijöiden aikaa muihin tehtäviin, lisäen tehokkuutta. Tämä työ on suoritettu viidennen sukupolven langattoman tietoliikenneverkkotekniikan (fifth generation mobile network new radio, 5G NR) radiokanavatilatietoon (channel state information, CSI) perustuvan säädön perspektiivistä. Työn tarkoituksena on esittää esimerkki automaattisen lokiprosessintijärjestelmän kehitysprosessista, ja tätä varten tarjota perustiedot CSI-mekanismista ja sen standardeista. Pääpaino tässä työssä on varmistaa, että CSI-raportointi noudattaa teknisiä standardeja, jotka usean standardointijärjestön yhteistyöorganisaatio (The 3rd Generation Partnership Project, 3GPP) on määrittänyt. Työssä esitetty systeemi sisältää kaksi päätoiminnallisuutta CSI-raportoinnin oikeuden varmistamiseen. Ensimmäinen toiminnallisuus varmistaa, että kaikki odotetut CSI-raportit ovat lähetetty onnistuneesti, joka merkitsee, ettei loki sisällä odottamattomasti pudotettuja CSI-raportteja. Pudotetuille CSI-raporteille on useita vian syitä, ja yksi esimerkki tästä on väärin lasketut bittien pituudet, joka heikentää kommunikaation suorituskykyä verkon ja käyttäjän laitteen välillä. Tähän liittyen, toinen systeemin päätoiminnallisuus varmistaa, että käyttäjän laitteen ohjelmisto on laskenut bittien pituudet eri parametreille oikein 3GPP-standardien mukaisesti. Nämä virheet kohdataan käyttäjän laitteen kehityksen vaiheessa, esimerkiksi, kun kehitetään uusia ominaisuuksia. Työn lopputulos on onnistunut, jossa kohtuullisen merkittävä järjestelmä on kehitetty nopeuttamaan manuaalista virheenkorjausta. Erityisesti suurempien lokien tapauksissa, järjestelmä tarjoaa kyvyn tutkia kokonainen loki huomattavan nopeasti verrattuna manuaaliseen virheiden korjaukseen.

Avainsanat: 3GPP, 5G NR, 5G Fyysinen kerros, L1

TABLE OF CONTENTS

FOREWORD

I want to thank MediaTek Wireless Finland Oy, especially my manager Jari Heinonen and technical leader Teemu Virtanen, for giving me an opportunity to write my master's thesis in your team about such an interesting software project where I could utilize my skills to create something new. During this project, I gained remarkable experience in the software industry from which I will benefit significantly in my future career. Special thanks go to my technical supervisor, Aki Hietala, for great guidance and continuous support throughout the project. Also, I want to thank every colleague who has been involved in this project, especially for sharing your professional technical knowledge and expertise with me.

Oulu, December 1st, 2022

Markus Kyllönen

LIST OF ABBREVIATIONS AND SYMBOLS

16QAM	16 Quadrature Amplitude Modulation
256QAM	256 Quadrature Amplitude Modulation
3GPP	The 3rd Generation Partnership Project
5G	Fifth Generation Mobile Network
64QAM	64 Quadrature Amplitude Modulation
AI	Artificial Intelligence
BM	Beam Management
BWP	Bandwidth Part
CE	Control Element
CQI	Channel Quality Indicator
CRI	CSI Reference Signal Resource Indicator
CSI	Channel State Information
CSI-RS	CSI Reference Signal
DFT	Discrete Fourier Transform
DL	Downlink
FR1	Frequency Range 1
FR2	Frequency Range 2
gNB	Next Generation Node B
i1	A single wideband indication of PMI
IE	Information Element
L1	Physical Layer in 5G
L1-RSRP	L1 Reference Signal Received Power
LI	Layer Indicator
M-MIMO	Massive Multiple-Input Multiple-Output
MAC	Medium Access Control
MIMO	Multiple-Input Multiple-Output
mmWave	Millimeter Wave
NR	New Radio
NZP	Non-zero Power
P-1	The first phase of beam management procedure
P-2	The second phase of beam management procedure
P-3	The third phase of beam management procedure
PCell	Primary Cell
PDCCH	Physical Downlink Control Channel
PDSCH	Physical Downlink Shared Channel
PMI	Precoding Matrix Indicator
PRB	Physical Resource Block
PUCCH	Physical Uplink Control Channel
PUSCH	Physical Uplink Shared Channel
QPSK	Quadrature Phase-Shift Keying
RI	Rank Indicator
RRC	Radio Resource Control
RSRP	Reference Signal Received Power
RX	Receiver

SB	Subband
SCell	Secondary Cell
SCS	Subcarrier Spacing
SF	Subframe
SFN	System Frame Number
SINR	Signal-To-Interference-Plus-Noise Ratio
SS	Synchronization Signal
SS/PBCH	Synchronization Signal and Physical Broadcast Channel
SSBRI	SS/PBCH Block Resource Indicator
TB	Transport Block
TX	Transmitter
UE	User Equipment
UI	User Interface
WB	Wideband
i_1	a single wideband indication of PMI
i_2	a subband indication of PMI
i_x	codebook index x
K_s^{CSI-RS}	the number of CSI-RS resources in the resource set
N_1	horizontal dimension of antenna panel array
N_2	vertical dimension of antenna panel array
N_g	number of panels in multi-panel antenna array
N_{max}	maximum N value, which is calculated from PMI, CQI, and LI quantities using all possible ranks
$N_{reported}$	reported N value, which is calculated from PMI, CQI, and LI quantities using reported rank
n_{RI}	the number of allowed RI values
O_1	determination of sweeping step in horizontal direction
O_2	determination of sweeping step in vertical direction
O_p	the number of zero padding bits
v	value of the rank
X_1	PMI information field used for wideband PMI reporting
X_2	PMI information field used for either wideband PMI reporting or for each subband PMI reporting
GHz	gigahertz
kHz	kilohertz
ms	millisecond
µ	parameter representing subcarrier spacing

1. INTRODUCTION

In the mobile communication system development industry, it may be challenging to investigate errors and bugs due to the massive size of debugging logs. Even if the fault reason was simple, it would be a way too time-consuming process to explore the whole debug log by hand. The main goal of this thesis is to develop an automated system which could handle a part of error debugging process. This requires getting familiar with channel state information (CSI) reporting feature in fifth generation mobile network (5G) new radio (NR). The work of this thesis is done for MediaTek Wireless Finland Oy. The following sections will give an introduction about the problem, and how this problem will be solved by the work of this thesis.

1.1. Background of the Problem

Mobile devices can communicate with the base station from different locations when the device is moving. Therefore, the quality of the radio channel between the base station and the mobile device is varying. To keep the communication channels between the base station and mobile device as stable as possible, the transmission signal has to be modified constantly. Consider the data transmission channel from the base station to the mobile device. The data are transmitted to the mobile device from the transmitter of the base station. This transmitter is responsible for modifying the characteristics of the transmitted signal. To be able to modify the characteristics of this signal, the base station shall identify mobile device information, such as where it is located. Because of this, an initial signal should be sent to the mobile device denoting that the base station needs the information about the mobile device. Once the base station informs the mobile device to send information about itself, the mobile device should perform different calculations relating to channel quality and transmit the information back to the base station. This information is used to modify the signal to be transmitted by the base station transmitter, to create as good a signal as possible.

In this process, there can be several issues. This subject matter of this thesis is from the mobile device modem development perspective, therefore, the focus will be only on mobile device modem-related issues. However, it is worth mentioning that the reason for the issue might be also due to the unexpected behavior of the Network. The first major issue to study is related to dropped reports. This occurs when the mobile device is trying to transmit channel-quality information back to the Network, but transmission fails. In this case, the base station is not receiving the information about the mobile device, and it is unable to modify the characteristics of the transmission signal. This will lead to decreased performance of communication between the mobile device and the base station. The second major issue to be studied is incorrect functioning in the software of the mobile device. For example, the sizes of transmitted reports should follow technical specifications. If sizes differ from specifications, the base station may not be able to receive reports correctly leading to other issues in future operations.

In mobile device modem development, these issues are investigated from error debug logs. These logs are created at the time of failure and provided to engineers for further investigation. The sizes of debug logs are usually massive, and manual processing is fairly slow. Also, there are many restrictions in manual processing. For example, logs are mainly processed from the specific timing locations, where the issue is expected to happen due to other unexpected behaviors.

1.2. Solution to the Problem

To tackle the above problem, in this thesis, an automated log processing system is developed to speed up error processing. It is expected to have a significant role in identifying different issues. Compared to manual processing, an automated system should be able to go through massive logs relatively fast, verifying that reporting is according to technical specifications. Also, a notable benefit of the automated log processing system is that the whole log can be processed, instead of investigating the issue from only specific timing locations. This way, unexpected issues might be found outside of the expected failure range. For example, an engineer might spend several hours verifying that the modem of the mobile device is functioning as expected, whereas an automatic system could find the issue within minutes.

Referring to the previous section, this channel quality modification reporting procedure is called 5G CSI reporting. An ideal situation in CSI reporting would be successfully formed and sent reports to the base station. In the following, 5G terminology will be used: user equipment (UE) will denote the mobile device, while the 5G base station will be called Next Generation Node B (gNB).

1.3. Explanation of Areas to Study with Relevant Terms

For understanding the solution proposed in this thesis, it is crucial to explore 5G specifications provided by the 3rd Generation Partnership Project (3GPP). 3GPP provides a complete system description for mobile telecommunications covering cellular telecommunications technologies such as radio access, core network and service capabilities [1]. In this work, an important aspect in CSI reporting is to verify that CSI reporting is functioning according to these specifications.

The first major aspect to study regarding CSI is the undesirable dropping of CSI reports, which may result in decreasing the performance of communication between gNB and UE. In other words, data throughput would be decreased. There are acceptable reasons why CSI reports should not be transmitted by the UE, for example, in particular Network configurations. However, the main focus of this thesis is on undesirably cancelled transmissions of CSI reports. One task for the developed automation system is to verify that the debug log includes all transmissions of requisite CSI reports. If CSI reports have not been received by the gNB, the system will identify the reason for the untransmitted CSI report, and notify the engineer who is responsible for debugging the issue. If the reason is previously known and accepted, it can be ignored. Otherwise, the system will mark the location where the drop occurred and the engineer can investigate the issue. There are also limitations on how the system can perform because debugging logs might be extremely complex. Briefly, the task for the system regarding dropped CSI reports is to exclude acceptable drop reasons and mark the unknown reasons for the engineer to resolve. This exclusively may increase the debugging process significantly, since debugging logs can include millions of lines to go through.

The second major issue to study regarding CSI reporting is incorrect bit length calculation for different quantities, which again may cause dropped CSI reports, decreasing the performance of data transmission. The 3GPP defines bitwidth calculations for different reporting quantities and mapping orders for different CSI reports in a technical specification [2] to which reference will be made in development of the bit length verification script feature. That feature verifies that the bitwidth for every single quantity is calculated correctly.

Once the knowledge of relevant CSI information is acquired from Chapter 2, log processing script design will be described in Chapter 3. The design chapter will give a short introduction about error debugging process flow and how the developed system will be part of it. Also, it will provide information on how different features of the system, such as calculations of bit lengths, will be implemented. An actual implementation of these functions will be introduced in Chapter 4. The implementation part provides functionalities in a pseudocode format and illustrations with pictures.

To verify that the developed system is functioning as desired, it will be tested in Chapter 5, where the script output will be provided for different cases. The testing phase includes the plan for testing, actual testing and verification. At the end of this thesis in Chapter 6, results of the work will be discussed. An important topic to discuss is success of the system, is it really improving working efficiency significantly. Also, the future work will be described, how the system could be developed even further.

2. THE 3GPP SPECIFICATIONS OF 5G NR CSI AND RELATED WORK

This thesis revolves around the 5G feature CSI. This chapter will give a basic knowledge about the CSI reporting to understand the topic relevant to developed automation log processing system.

2.1. CSI Reporting Procedure in a High Level

At a high level, CSI indicates the quality of the radio signal between a UE and gNB. The UE measures various radio channel quality quantities and reports the results to gNB [3]. More specifically, the UE receives downlink (DL) CSI reference signal (CSI-RS) from gNB, from which it estimates the channel and sends the quality information back to the gNB using uplink (UL) transmission [4]. Once the gNB receives the CSI report with information about different quantities, it schedules DL data transmissions according to reported quantities [5]. In CSI reporting, these CSI reports can be considered as acquisition reports. The purpose of this CSI operation is to generate as good radio channel as possible between the UE and gNB by optimizing channel characteristics, such as correct modulation, code rate, and beam forming [6].

Figure 1 represents a high level CSI reporting procedure in understandable manner, which includes three main different signal components:

1. CSI reference signal (CSI-RS) from gNB to UE,
2. CSI report transmission signal using physical uplink shared channel (PUSCH) or physical uplink control channel (PUCCH), see below, and
3. DL data transmission signals

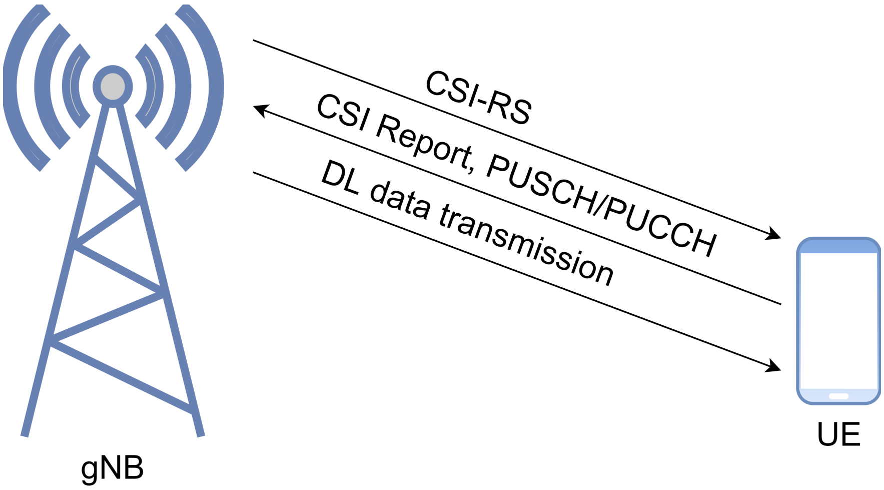


Figure 1. CSI operation illustration at a high level with three different signal components. Adapted from [5].

Exploring Figure 1, CSI reporting is started by sending CSI-RS to the UE by the gNB. The UE measures channel quality quantities and forms the CSI report, which is sent back to the gNB for further use. CSI reports will be transmitted on PUCCH or PUSCH, depending on *CSI-MeasConfig* of the serving cell, see Section 2.3. *CSI-MeasConfig* is an important information element (IE) for this work, configuring each CSI-RS for the serving cell in which *CSI-MeasConfig* is located [7]. Once the CSI Report has been transmitted to the gNB successfully, it performs different calculations to enhance the radio signal between the gNB and the UE, creating the most optimized DL data transmission as possible. The usage of CSI-RS provides flexibility in time domain and frequency domain resource allocation, but as disadvantage, it is causing overhead [8].

2.2. Beam Management

5G NR technology classifies two different frequency bands called FR1 (Frequency Range 1) and FR2 (Frequency Range 2). FR1 is considered as sub-6GHz, which covers mid and low-frequency bands below 6 GHz (Gigahertz). FR2 covers frequency bands above 24 GHz and it is defined as millimeter wave (mmWave). Both of these frequency bands experience respective advantages and disadvantages. FR1 allows the signal to cover larger geographical areas with the cost of lower DL throughput. Because FR2 covers considerably higher frequency ranges, transmission speed in DL direction can be faster. However, FR2 covers smaller geographical areas [9].

Due to high frequency bands in 5G NR, directional communication is required to compensate propagation loss [10]. In the case of mid and low-frequency bands, transmission of the signal is possible for relatively wider angles compared to high frequency bands. To reach the same ranges with higher frequency bands, multiple antennas are needed. Due to multiple antenna array, the signal pattern appears as a beam-shaped, therefore, multiple beams are required to cover wider areas. Because of this, beam management (BM) is essential [11]. Figure 2 illustrates shapes of low and high-frequency radiations of the signal. Left side of the figure represents a single antenna port configuration with low frequency, and the right side represents a multiple antenna configuration with high-frequency. In CSI reporting, Layer 1 reference signal received power (L1-RSRP) reports are used for BM.

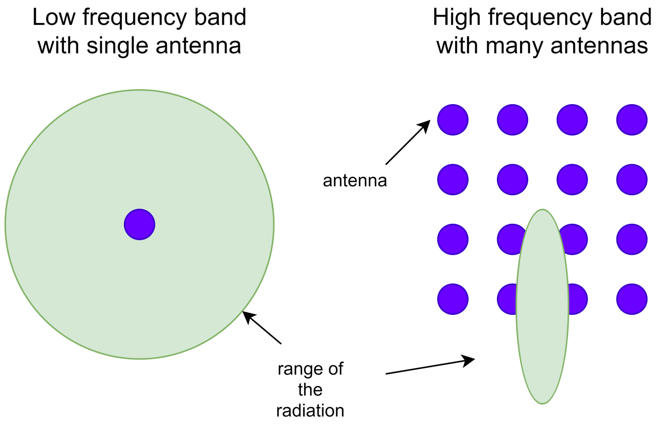


Figure 2. Shapes of low and high-frequency radiations of the radio signal. Adapted from [11].

To further understand the beam management procedure between the UE and gNB, 3GPP technical specification [12] shall be explored. The specification defines three different beam management phases to find the best signal between UE and gNB.

2.2.1. P-1 Procedure

The first phase in the beam management procedure is to enable measurement of the UE to support selection of the gNB transmitter (TX) beam and the UE receiver (RX) beam. This phase is called P-1 and it is illustrated in Figure 3. Beams of the qNB are swept and the UE selects a beam with the best signal and reports it back to the gNB for initial beam selection [12, 11].

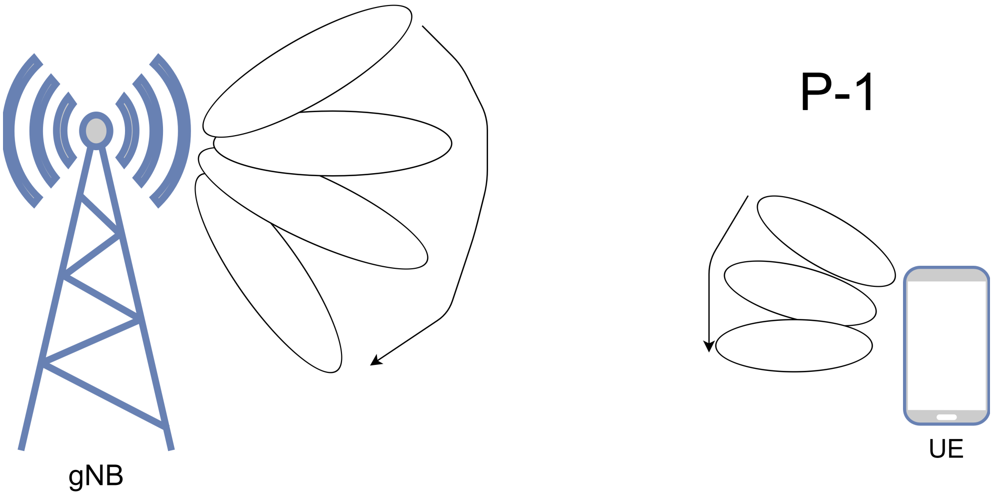


Figure 3. P-1: Selection of initial gNB TX and UE RX beam by sweeping and measuring different gNB beams by the UE.

2.2.2. P-2 Procedure

The figure 4 illustrates the second phase P-2 of the beam management procedure. Once the best UE RX beam has been selected in P-1 phase, next step is to refine gNB TX beams to sweep a narrower beam over a narrower range to the UE. Also, the number of beams in gNB TX set can be reduced. In this phase, it is possible to change initial gNB TX beam to a better one according to the UE measurements [12, 11].

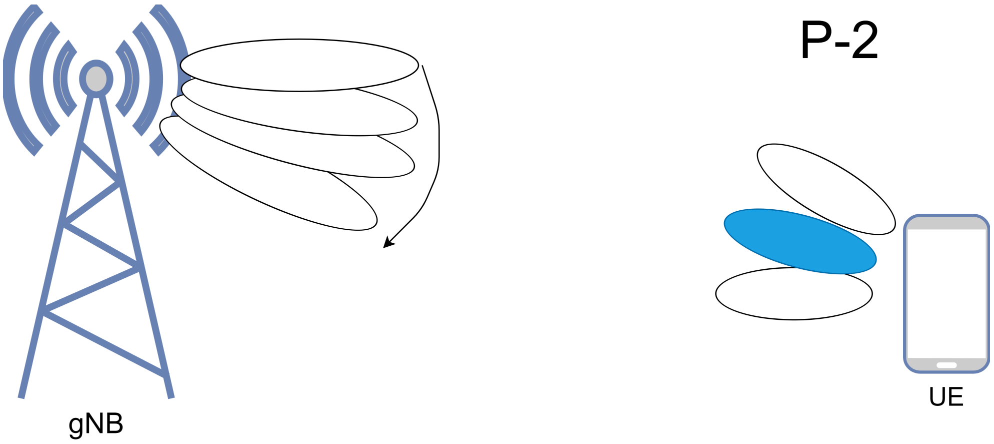


Figure 4. P-2: Refinement of gNB TX beams and possibly change of the beam according to UE measurements.

2.2.3. P-3 Procedure

The third phase P-3 is illustrated in Figure 5. In P-3 procedure, the UE performs measurements using the same gNB TX beam and possibly changes the UE Rx beam. In this case, it is expected that the UE is supporting beamforming, otherwise it is not used [12, 11].

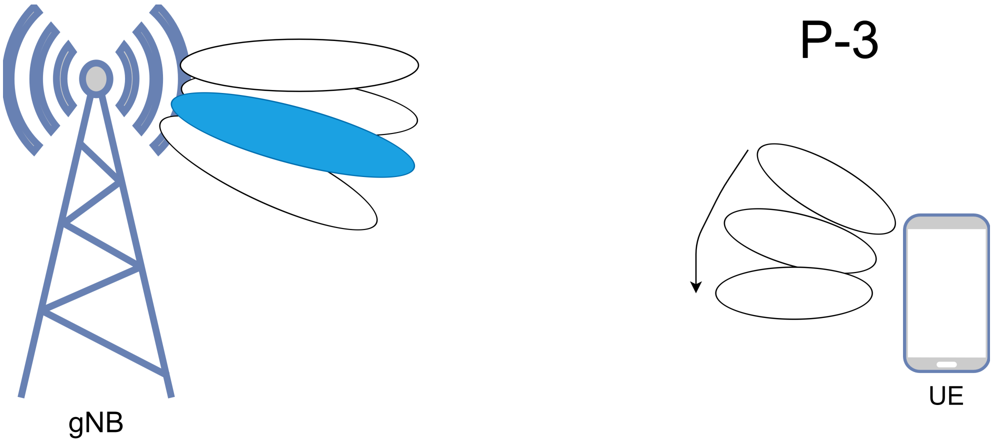


Figure 5. P-3: Measurement of different UE RX beams for selection using the same gNB TX beam.

2.3. Aperiodic and Periodic Reports

The system developed in this work shall be able to handle CSI reports with different configurations. There are three configurations for CSI reports [13]:

1. Aperiodic (using PUSCH),

2. Periodic (using PUCCH),
3. Semi-persistent (using PUCCH while downlink control information (DCI) activated by PUSCH)

The focus in this thesis will be only on aperiodic and periodic reports. The main difference between aperiodic and periodic configuration is triggering of CSI reporting. Periodic CSI reporting does not require any dynamic triggering or activation, while aperiodic CSI reporting has to be triggered by DCI [13]. In addition, aperiodic CSI reporting can be activated by a medium access control (MAC) control element (CE) [14]. The illustration of differences between aperiodic and periodic reporting is represented in Figure 6. In aperiodic reporting, X value represents the time gap between the trigger and the CSI-RS, expressed in slots. The value Y represents the time gap between the triggering point and transmission of CSI report. Periodic reporting is simpler, because the reporting occurs in specific time intervals based on periodicity and timing offset [15].

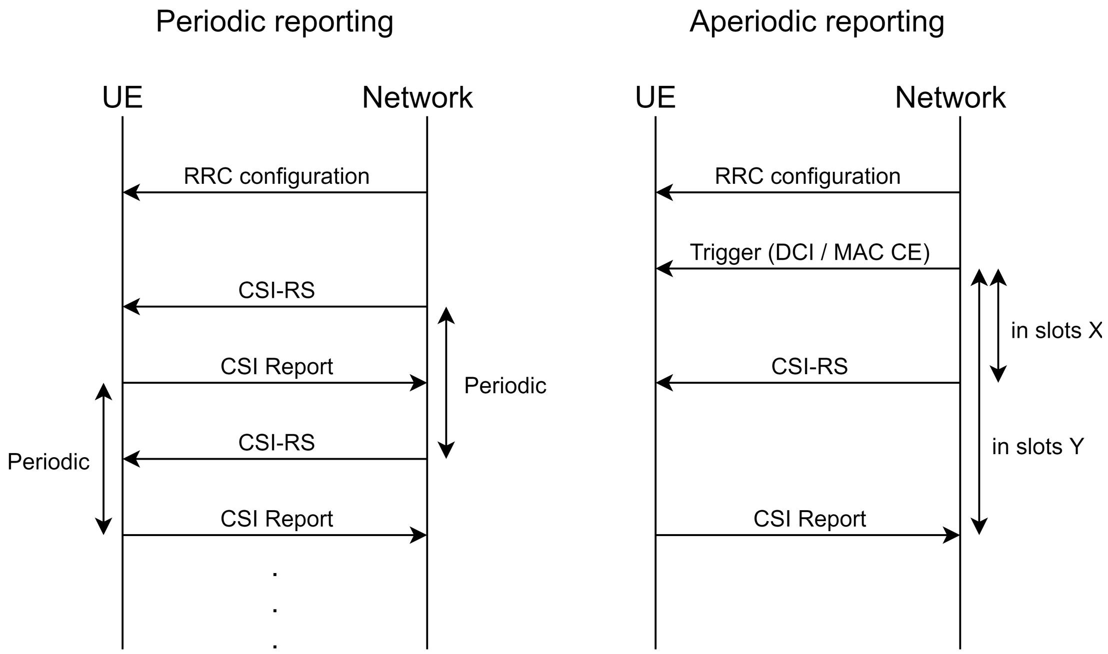


Figure 6. Aperiodic reporting compared to periodic reporting. Adapted from [8].

Since periodic reports are sent periodically in specific time intervals, periodical configuring is expected. The IE *CSI-ResourcePeriodicityAndOffset* configures a periodicity and an offset for periodic CSI resources and reports, which are transmitted on PUCCH. The periodicity and the offsets are given in a slot level [7]. In the work of this thesis, different UL bandwidth part (BWP) numerologies specified for NR shall be taken into account. 5G NR supports five different subcarrier spacing (SCS) numerologies with following frequency bands [16]:

1. μ= 0, frequency = 15 kHz,
2. μ= 1, frequency = 30 kHz,

3. μ= 2, frequency = 60 kHz,
4. μ= 3, frequency = 120 kHz,
5. μ= 4, frequency = 240 kHz

The numerologies above have an impact on CSI report transmission times which should be considered in the design of the automated log processing system. Clearly, the larger the numerology is used, the smaller the time should be between sent reports. This will be taken into account in the development of the system's CSI report drop check feature, which will be introduced in more detail in Chapter 3.

The IE *CSI-ReportConfig* configures either a periodic or an aperiodic report, which shall be sent on the PUCCH or the PUSCH of the cell. In the IE, configuration type is defined in the setting *reportConfigType* [7]. From the developed system perspective, an important aspect is to identify the configuration of processed report in order to perform appropriate operations.

2.4. Report Quantities

To be able to calculate and verify bit lengths of CSI reports, one must get familiar with different report quantities. CSI reports are formed by various report quantities, depending on configuration of the Network. In this thesis, the main aspect is to calculate bitwidth for every configured quantity and verify that the mapping orders match with the Network configuration [2]. As introduced earlier, the IE *CSI-ReportConfig* consists of different configurations for the CSI reporting. The higher layer parameter *reportQuantity* can be found from that IE and it defines a UE report quantity configuration. In other words, which quantities shall be reported on CSI report. The *reportQuantity* can be set to one of the following higher parameter quantity configurations [13]:

1. 'none',
2. 'cri-RI-PMI-CQI' (CSI reference signal resource indicator, rank indicator, precoding matrix indicator, channel quality indicator),
3. 'cri-RI-i1' (CSI reference signal resource indicator, rank indicator, a single wideband indication of PMI),
4. 'cri-RI-i1-CQI' (CSI reference signal resource indicator, rank indicator, a single wideband indication of PMI, channel quality index),
5. 'cri-RI-CQI' (CSI reference signal resource indicator, rank indicator, channel quality index),
6. 'cri-RSRP' (CSI reference signal resource indicator, reference signal received power),
7. 'cri-SINR' (CSI reference signal resource indicator, signal-to-interference-plus-noise ratio),

8. ’ssb-Index-RSRP’ (synchronization signal and physical broadcast channel block resource indicator, reference signal received power),
9. ’ssb-Index-SINR’ (synchronization signal and physical broadcast channel block resource indicator, signal-to-interference-plus-noise ratio), or
10. ’cri-RI-LI-PMI-CQI’ (CSI reference signal resource indicator, rank indicator, layer indicator, precoding matrix indicator, channel quality indicator)

The parameter configurations above define which different quantities may be included in CSI report. Based on the quantities, the bitwidth calculations and mapping orders will be performed by the automated system developed in this work. The more detailed explanations for these individual quantities are presented in the following sections.

2.4.1. CQI

Channel quality indicator (CQI) is the quantity in CSI reporting, which is the most important one for determining the achievable data rate of multimedia transmission [17]. Basically, CQI indicates how good is the signal between the gNB and the UE. The CQI reporting is based on either quadrature phase-shift keying (QPSK), 16 quadrature amplitude modulation (16QAM), 64 quadrature amplitude modulation (64QAM), or 256 quadrature amplitude modulation (256QAM) depending on the table used. CQI indices can be integer values between 0 and 15, the higher the value, the better the signal. Since the index has 16 different values, CQI requires 4 bits to be reported. In addition, CQI value can get whole number values between 0 and 3. In this case, the value is called subband differential CQI value and CQI index is reported as a 2-bit value [13]. CQI can be reported either in subband or wideband granularity, further explanation can be found in Section 2.5.

2.4.2. PMI

Precoding matrix indicator (PMI) quantity is related to beamforming in CSI-RS and it depends on the codebook configured by the Network. The codebook is a set of precoders, that transforms the physical downlink shared channel (PDSCH) data to another set of data mapped to each antenna port [18]. The higher layer parameter *codebookType* configures the type of codebook for the UE. Codebook configurations differ from each other in antenna port configurations and different support of ranks, which again affect on what codebook indices shall be included in reporting. There are several types of different codebooks defined in NR as follows [13].

Type I Single-Panel Codebook

It can be expected that type of the PMI codebook is a single-panel, when higher layer parameter *codebookType* is set to *’typeISinglePanel’*. In a single-panel configuration, the CSI reporting is possible with every configuration of antenna ports:

2/4/8/12/16/24/32. Also, it is possible to use any rank: 1/2/3/4/5/6/7/8. Depending on antenna port configuration and used rank, single-panel codebook reporting could be divided into three different categories [13]:

1. 2 antenna ports: UE is required to report only one codebook index
2. >2 antenna ports and used rank 2/3/4: UE is required to report $i_{1,1}$, $i_{1,2}$, $i_{1,3}$, and i_2
3. >2 antenna ports and used rank 1/5/6/7/8: UE is required to report $i_{1,1}$, $i_{1,2}$, and i_2

Higher layer parameter *n1-n2* configures the values of N_1 and N_2 defining the dimensions of antenna port [13], see Figure 7. According to N_1 and N_2 values, discrete fourier transform (DFT) oversampling values O_1 and O_2 can be defined for each configuration. Overall, O_1 and O_2 values are used to determine the sweeping steps of a beam in a beam management. The value O_1 determines the sweeping step in horizontal direction, and O_2 value determines the sweeping step in vertical direction. The higher these values are, the smaller in steps beams can be swept [18]. Number of CSI-RS antenna ports can be calculated using configured N_1 and N_2 values with the following equation [13]:

$$\text{Number of CSI-RS antenna ports} = 2 * N_1 * N_2 \tag{1}$$

Figure 7 gives an illustration about single panel configuration including N_1 and N_2 values. In the figure, blue and red lines are representing antenna ports.

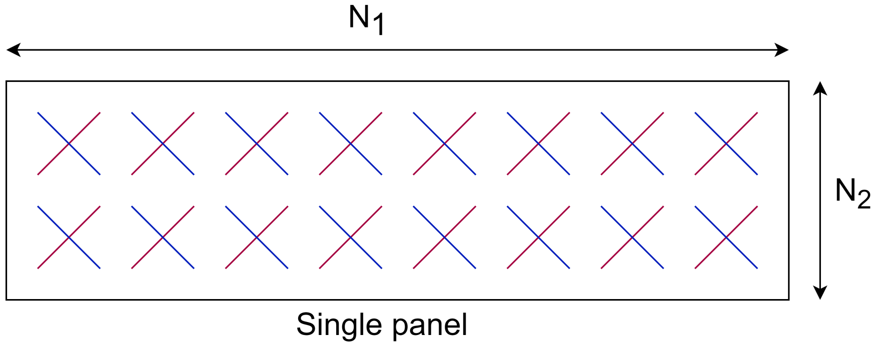


Figure 7. Single panel antenna port configuration (N_1 = 8, N_2 = 2). Adapted from[18].

Type I Multi-Panel Codebook

When the UE is configured with higher layer parameter *codebookType* set to *'typeI-MultiPanel'*, multi-panel is expected. As the name indicates, type I multi-panel consists of multiple panels compared to single-panel configuration with only one panel. In addition to N_1 and N_2 values, the third N_g value is required to indicate the amount of panels in each supported configuration. Therefore, type I multi-panel codebook is configured with three values N_1, N_2, and N_g, which are configured in the higher layer parameter *ng-n1-n2*. The purpose of O_1 and O_2 values are similar to single-panel configuration. However, because the configuration of O_1 and O_2 values depends on N_1, N_2, and N_g values, different configuration table is required compared to single-panel configuration [13].

The amount of antenna port configurations is smaller in multi-panel configuration, there are three different numbers of CSI-RS antenna ports that can be used: 8/16/32. Also, only four ranks are supported: 1/2/3/4. Based on rank, *codebookmode*, and N_g value, reported codebook indices are defined as follows [13]:

1. used rank 1: UE is required to report $i_{1,1}$, $i_{1,2}$, $i_{1,4}$
2. used rank 2/3/4: UE is required to report $i_{1,1}$, $i_{1,2}$, $i_{1,3}$, and $i_{1,4}$
3. *codebookmode* set to '1' and N_g = 2: UE is required to report $i_{1,4,1}$
4. *codebookmode* set to '1' and N_g = 4: UE is required to report $i_{1,4,1}$, $i_{1,4,2}$, and $i_{1,4,3}$
5. *codebookmode* set to '2': UE is required to report $i_{1,4,1}$, $i_{1,4,2}$, $i_{2,0}$, $i_{2,1}$, and $i_{2,2}$

The number of CSI-RS antenna ports for multi-panel configuration can be calculated using N_1, N_2, and N_g values with the following equation [13]:

$$\text{Number of CSI-RS antenna ports} = 2 * N_g * N_1 * N_2 \tag{2}$$

Figure 8 illustrates multi panel configuration including N_g, N_1, and N_2 values. In this case, multiple panels are configured in the CSI antenna port array.

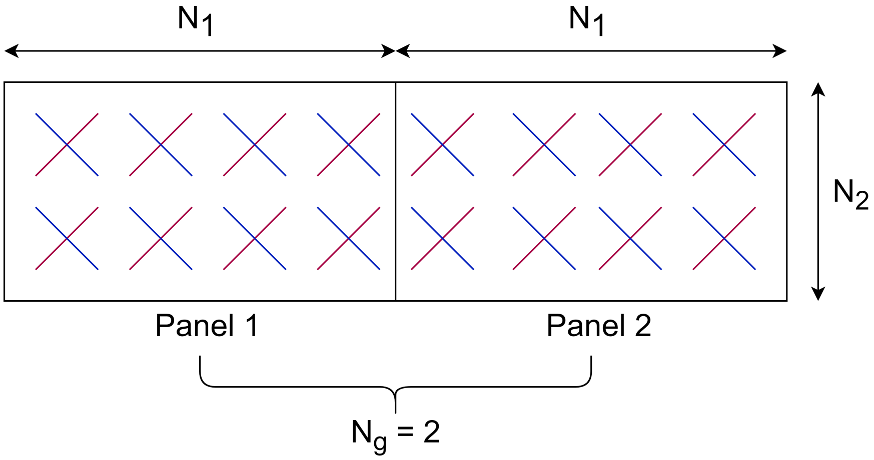


Figure 8. Multi panel antenna port configuration (N_g = 2, N_1 = 4, N_2 = 2). Adapted from [18].

Type II Codebooks

Another type of codebooks is Type II. The difference between Type I and Type II codebooks is related to beam management. In Type I codebook, one specific beam is selected from group of beams, while Type II codebook combines beams from a selected group. Therefore, Type II codebook requires a bigger amount of processing and it is much more complicated [18]. The 3GPP technical specification has defined four different Type II codebooks [13]:

1. Type II Codebook,
2. Type II Port Selection Codebook,
3. Enhanced Type II Codebook,
4. Enhanced Type II Port Selection Codebook

In this work, the further knowledge of Type II codebooks is not essential, because the focus is mainly on Type I configurations. It is reasonable to give more in-depth focus on Type I single panel configurations, since those are the most commonly used in commercial networks [19].

2.4.3. CRI

The higher layer parameters *CSI-ResourceConfig* and *NZP-CSI-RS-ResourceSet* may configure the UE with one or several non-zero power (NZP) CSI-RS resource set(s). For periodic reports, only one CSI-RS resource set can be configured [13]. Maximum number of NZP CSI-RS resources is 8 per resource set [7]. Each resource set references to one or more configured CSI reference signals, and can be used to describe measurements by the UE. In addition, NZP-CSI-RS-ResourceSet may point to a set of synchronization signal (SS) blocks, especially for beam management and mobility measurements [6].

The purpose of CSI reference signal resource indicator (CRI) is to identify the strongest CSI-RS and report the best DL refined beam [20]. It is an extremely useful indicator, because it can quickly point to the best CSI-RS resources for the gNB usage. Also, CRI allows the gNB to switch between CSI-RS beams which are usually more directional compared to synchronization signal and physical broadcast channel (SS/PBCH) beams [6].

2.4.4. RI

As mentioned in previous sections, codebooks are used for beamforming in CSI. Codebooks are features of massive multiple-input multiple-output (M-MIMO) technology in 5G NR [21] which is highly related to rank indicator (RI) reporting quantity. Capacity of the MIMO channel depends on the selected rank, which again depends on the antenna correlation. The maximum RI value is related to the number of TX and RX antennas, which indicates the non-existent correlation between antennas. In other words, it means the best performance with no interference to each other. To simplify, RI indicates how well MIMO antenna configurations are functioning [22, 23].

In 5G NR, there are eight different possible ranks in increasing order from number 1 to number 8, as introduced in previous sections. Configuration of *CodebookConfig*

IE restricts possible rank values through *ri-Restriction* parameter. The name of the configuration parameter may vary depending on the type of used codebook and the selected rank has an effect on PMI reporting [2, 13, 7].

2.4.5. LI

Layer indicator (LI) is used by the gNB when transmitting physical downlink control channel (PDCCH), and when transmitting phase tracking reference signal (PT-RS) for the PDSCH. The purpose of the LI is to identify the strongest MIMO layer that the UE would receive, assuming the gNB has precoded its transmissions by means of the PMI [24]. The reported PMI includes precoded matrix, and LI indicates which column corresponds to the strongest layer of the codeword with the largest reported wideband CQI value. In the case where two wideband CQIs are reported with equal value, LI corresponds to the strongest layer of the first codeword [25].

2.4.6. L1-RSRP

If *CSI-ReportConfig* configures the UE with the higher layer parameter *reportQuantity* set to 'cri-RSRP' or 'ssb-Index-RSRP', L1-RSRP reporting is expected. In L1-RSRP computation, the UE may be configured with three different kind of resource configurations [13]:

1. CSI-RS resources
2. SS/PBCH block resources, or
3. both CSI-RS and SS/PBCH block resources

The maximum number of resource sets is up to 16 CSI-RS resource sets in the UE configuration and each set contains up to 64 resources. Also, the maximum number of different CSI-RS resources is 128 over all resource sets [13]. L1-RSRP can be further divided to two types whether the differential L1-RSRP should be used or not. This configuration depends on two factories; *nrofReportedRS* configuration and *groupBasedBeamReporting* parameter. If *nrofReportedRS* is set to value one, the UE shall report the L1-RSRP value as a 7-bit value. The possible values are between $-140dBm$ and $-44dBm$ with $1dB$ step size. When *nrofReportedRS* is configured to be larger than value one, or if *groupBasedBeamReporting* parameter is set to enabled, differential L1-RSRP based reporting shall be used by the UE. In differential L1-RSRP reporting, the L1-RSRP value is a 4-bit value with $2dB$ step size. L1-RSRP measurement reports shall be sent only for an active BWP by the UE. The greater the L1-RSRP value, the better the radio signal [7, 13].

2.5. Subband and Wideband

PMI and CQI reporting may be configured in two frequency granularities, either in wideband or in subband. In a wideband CQI reporting, a UE reports a wideband CQI for each codeword for the entire CSI reporting band. One CQI for each codeword is reported for each subband in the CSI reporting band, if a UE is configured to use subband CQI reporting. The CQI frequency granularity is configured by the higher layer parameter *cqi-FormatIndicator* [13].

The frequency granularity of PMI reporting is similar to CQI reporting. The higher layer parameter *pmi-FormatIndicator* configures the UE to either use subband or wideband PMI reporting. Wideband PMI is reported for the entire CSI reporting band, if PMI is reported to be wideband. In subband PMI reporting, there is an exception when two antenna ports are configured. With a two antenna port configuration, PMI is reported for each subband in the CSI reporting band. When the antenna port configuration is something other than two and subband PMI is configured, reporting is divided into two parts. In the first part, a single wideband indication i_1 should be reported for each subband in the CSI reporting band. The other part has one subband indication i_2 which shall be reported for each subband in the CSI reporting band [13].

Bit-string *csi-ReportingBand* indicates a subset of subbands in the BWP which CSI shall use in reporting. In *csi-ReportingBand*, each bit represents one subband and the value of each bit determines if it shall be reported or not. If the value of the bit equals 1, the subband has to be reported. The field *csi-ReportingBand* should be absent when reporting band is consisting of less than 24 physical resource blocks (PRB), meaning that subbands shall not be configured [7]. Figure 9 represents an example of active BWP with 12 subbands configured by *csi-ReportingBand*.

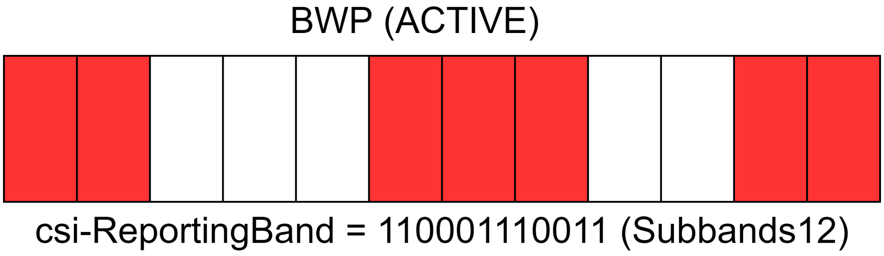


Figure 9. Subset of subbands in active BWP while 12 subbands configured by *csi-ReportingBand*. Subbands marked with red color shall be reported.

The 3GPP technical specification defines when CSI reporting shall use a wideband frequency granularity, depending on three higher layer parameters *reportQuantity*, *cqi-FormatIndicator*, and *pmi-FormatIndicator* [13]:

1. *reportQuantity* = ’cri-RI-PMI-CQI’ or ’cri-RI-LI-PMI-CQI’, cqi-FormatIndicator = ’widebandCQI’ and pmi-FormatIndicator = ’widebandPMI’
2. *reportQuantity* = ’cri-RI-i1’
3. *reportQuantity* = ’cri-RI-CQI’ or ’cri-RI-i1-CQI’ and cqi-FormatIndicator = ’widebandCQI’
4. *reportQuantity* = ’cri-RSRP’ or ’ssb-Index-RSRP’ or ’cri-SINR’, or ’ssb-Index-SINR’

If any of these condition above is true, a UE shall use wideband frequency granularity. Otherwise, subband configuration is expected [13].

In wideband reporting, the report has to be fixed in a size in case the reported rank changes affecting bit lengths during reporting. The number of zero padding bits depends on CSI-RS antenna port configuration. If only one antenna port is configured, the number of zero padding bits O_p shall be zero, and for more than one antenna port, the following equation is used [2]:

$$O_p = N_{max} - N_{reported} \tag{3}$$

Values N_{max} and $N_{reported}$ are calculated using PMI, CQI, RI, and LI report quantities. When determining the N_{max} value, bit lengths for PMI, CQI, and LI are calculated and combined using all possible ranks. The number of outputs will be the same as the number of possible ranks. From the outputs, the value with the greatest bit length is assigned to value N_{max}. PMI, CQI, and LI bit lengths for $N_{reported}$ value are calculated using only the reported rank, otherwise the calculation is similar [2].

3. LOG ANALYSIS SYSTEM DESIGN

After acquiring the basic knowledge of CSI reporting, design of functionalities of the system shall be done. The design will be presented in this chapter. At first, different tools will be described. This includes the tools used in the developing phase to design and implement the system, and example tools which could be applied in further use of the system. To identify the purpose of this system, an example of CSI related error debugging processing flow is introduced. Also, supported log formats shall be defined. To be able to implement the system, it is crucial to design the main functionality together with different calculation functions. To illustrate these functionalities, different design state diagrams are provided. The goal of this chapter is to design a reliable and well-functioning system to ease the implementation phase, which will be described in Chapter 4.

3.1. Tools Used

For actual implementation of software of the automated system, the Python programming language is used. Python is an interpreted, object-oriented, high-level programming language with dynamic semantics [26]. It is extremely attractive in implementation of this kind of script because it is relatively simple to use with high-level built-in data structures. The system in this thesis is implemented using Python, but the source code in its entirety will not be provided to give a better understanding of relevant information about the topic of this thesis. The automated system is built on large-scale software which includes plenty of irrelevant terms and different architectures regarding this thesis. Therefore, this thesis provides solutions in a pseudocode format to make it more understandable for the reader, full source code would be way too confusing. Also, if the solution of this thesis will be further applied for other uses, it will be easier to reverse-engineer the solution from a pseudocode format. In further application use cases, the usage of Python programming language would be recommended.

3.2. Error Processing Flow

As mentioned in the introduction, the main goal of this project is to create an automated system to simplify and speed up the error resolving process in the mobile communication development industry. Error logs requiring handling might contain millions of lines and processing them may take way too high amount of time for a human. Also, manual log processing is usually focusing on a specific fault location while an automated system can process the whole debug log fairly fast. Thus, other possible unexpected behaviors might be found from random locations. The work of this thesis focuses on creating a system to handle some parts of this process, not to fully replace the engineer. The first major functionality of the system is to handle CSI report dropping, and the second major functionality is to verify that bit lengths for different quantities are calculated correctly. These are further described later in this chapter.

To provide an understanding of log processing in error debugging, an example of error processing flow is represented in Figure 10 below. The figure illustrates a high-level representation from a software engineer's perspective in which kind of error-solving process the solution of this thesis is needed. Basically, when UE software is not working as expected, it raises an error. In the same Figure, two different examples of error sources are represented. For example, failures may occur in internal UE software testing, or errors might be received from customer tests, which means it is essential to resolve and fix them as soon as possible. The error submitter provides relevant information about the error along with a debug log to the engineer for resolving. The engineer resolves and creates a possible fix for the issue, and then provides it back to the original error submitter.

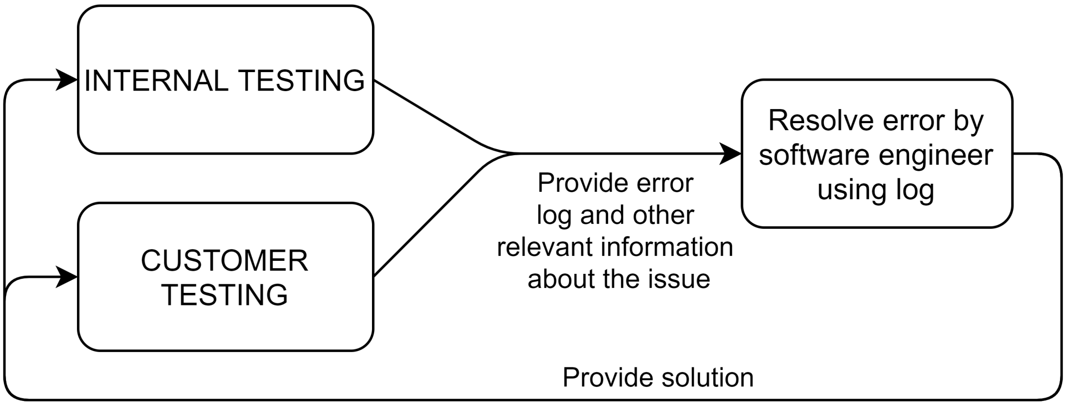


Figure 10. High level error processing flow.

Figure 11 represents the error processing flow with addition of automated log processing system. Between error submission and error resolving by the engineer, the system performs its operations and creates a final report as output for the engineer. If the system verifies that there are not any dropped reports and bit lengths have been calculated correctly, the engineer may ignore these fault reasons which may have required plenty of working hours to go through. If dropped reports exist or there are incorrect bit length calculations, the system will parse relevant information from the log for the engineer to speed up the resolving and fixing process. In some cases, script output will find complete solution for the error, for example, if there are incorrect bit length calculations for report quantities and it can be confirmed that UE software is not functioning as desired. In case the debug log contains dropped reports, the fault reason might be still unknown, and the engineer has to investigate the issue through log. However, the script has marked locations for untransmitted reports, and most likely the issue can be found close to these locations, which again will speed up the debug process.

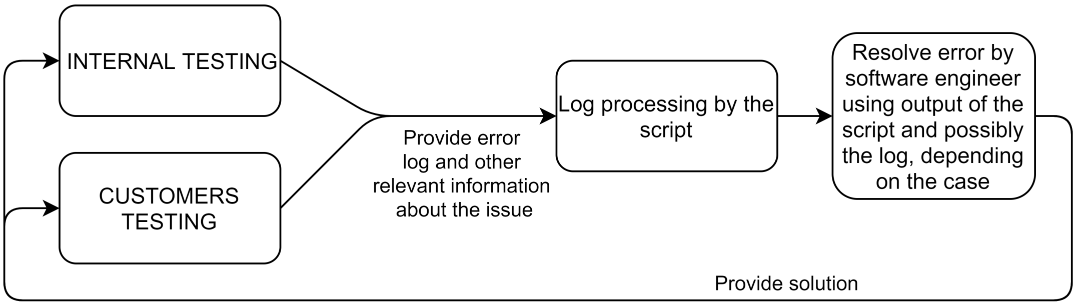


Figure 11. High level error processing flow with addition of automated log processing system developed in this work.

Even though this process might appear extremely simple, it has a significant possibility to speed up the error resolving process and may save the engineer's working time for other tasks, which could have an outstanding impact on working efficiency.

3.3. Format of Logs to Be Processed

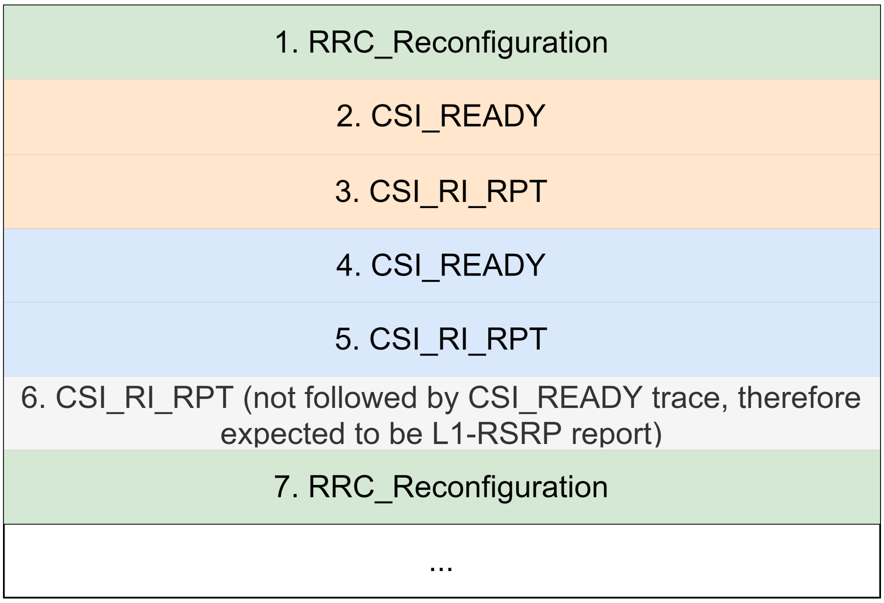
To be able to design the automated log processing system, the basic format of debug logs shall be introduced. In this work, the log consists of traces of different UE functionalities during communication with the gNB. As mentioned in the previous chapter, these logs are provided for resolving from internal software testing or from customer testing. For this work, it is essential to introduce three different major traces to be able to design the main functionality of the automated log processing system. Table 1 represents three example traces used to verify CSI reporting operation.

Table 1. Three major traces illustrated to be able to design the main functionality of the automated log processing system

Trace name	**Purpose**
RRC_Reconfiguration	Modify an RRC connection.
CSI_READY	Provides information about UE measurements, such as measured rank and CQI.
CSI_RI_RPT	Represents sent CSI report and provides information about different calculations, such as bit lengths.

To understand the purpose of each trace in Table 1, terms shall be discussed. The first trace is called *RRC_Reconfiguration*, which is the procedure to modify a radio resource control (RRC) connection, including modifications, additions, and releases of different configurable instances [7]. For example, different types of used CSI reports are configured in *RRC_Reconfiguration*. The second defined trace is called *CSI_READY*, which provides information about UE measurements, such as measured rank and CQI. Measured rank from this trace will be utilized in the implementation of the bit length verification feature. The third trace is called *CSI_RI_RPT*, which represents the sent CSI report and provides information about different calculations, such as bit lengths. In this work, all these traces are provided from other layers, therefore, further description is not essential. The purpose of this work is to verify that information of these traces related to CSI reporting is according to 3GPP technical specifications.

Another important aspect is the sorting of relevant traces. For the design, it is assumed that traces are sorted by timestamps. An example of sorting is illustrated in Figure 12. First, *RRC_Reconfiguration* configures an RRC connection. After an RRC Reconfiguration, *CSI_READY* trace gives information about CSI report, which is followed by *CSI_RI_RPT* providing information about calculated bit lengths and pointing that CSI report has been sent. *CSI_READY* trace shall not be followed by *CSI_RI_RPT* trace only if CSI report is L1-RSRP related, as shown in item 6 of Figure 12.



1. RRC_Reconfiguration
2. CSI_READY
3. CSI_RI_RPT
4. CSI_READY
5. CSI_RI_RPT
6. CSI_RI_RPT (not followed by CSI_READY trace, therefore expected to be L1-RSRP report)
7. RRC_Reconfiguration

...

Figure 12. An example of sorted debug log format to be processed.

3.4. Main Function Design with Bitwidth Calculations

The initial design phase in the log processing automation system development is to create a basic functionality of the script including verification of bit lengths. Report drop feature addition is excluded from this phase and will be discussed in the next section. Figure 13 illustrates the initial design in a state diagram. The system starts by subscribing traces from the debug log and adding these traces into the queue. If the log contains relevant traces, the process is started by selecting the first trace from the queue. Referring to the previous section, there are three different possible traces in the initial design. As illustrated in a state diagram of the design, these traces form three different main branches.

RRC_Reconfiguration trace handling

If the trace is equal to *RRC_Reconfiguration*, the system basically updates RRC configuration and next operations will utilize this new configuration. To be able to perform operations, such as bit length calculations, it is expected that RRC is configured first. Therefore, the system will not perform any operations until the first *RRC_Reconfiguration* trace is faced.

CSI_READY trace handling

The second possible main branch in the design is *CSI_READY* trace handling. Because this trace provides only information about UE measurements it is expected that *CSI_RI_RPT* trace is following *CSI_READY* trace. This should be taken into account in the system design. If the following subscribed trace is not equal to *CSI_RI_RPT* trace, the debug log is invalid for this system and an assertion error is raised. Assuming that the next trace is correct, bit length calculations will be performed based on RRC configuration. In case bit length calculations match with the information found from the *CSI_RI_RPT* trace, this branch has been successfully passed. If calculations differ from each other, the warning is raised about incorrect bit length calculations.

CSI_RI_RPT trace handling

CSI_RI_RPT trace case is the third main branch to handle. In the case, where *CSI_RI_RPT* trace is not followed by *CSI_READY* trace, it is expected that report is L1-RSRP related. This can be verified by *reportQuantity* higher layer parameter, which shall be ’cri-RSRP’ or ’ssb-Index-RSRP’. Otherwise, the debug log is invalid and an assertion error is raised. After verifying the quantity of the report, L1-RSRP-related bit length calculations are performed and compared to *CSI_RI_RPT* trace information. Again, if calculations match with each other, the branch has been successfully passed. Otherwise, the warning will be raised about incorrect L1-RSRP bit length calculations.

After the main branch has been gone through, the system will continue selecting traces from the queue of subscribed traces until the queue is empty. Once the queue is empty, a summary report of the process is created and the system will finish the processing.

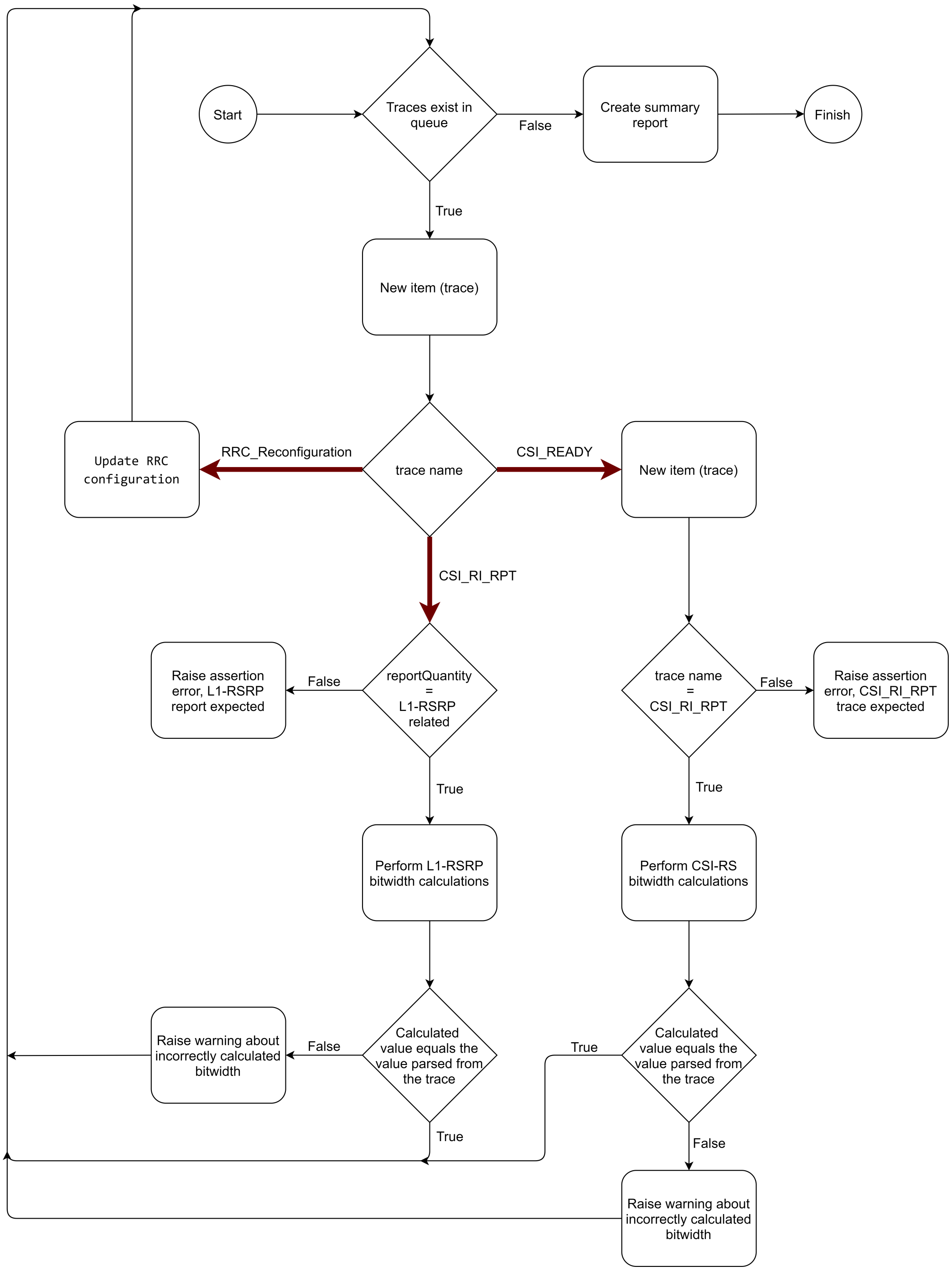


Figure 13. State diagram of the main functionality design including bitwidth calculations.

3.5. Main Function Design with Addition of Report Drop Feature

In addition to the main function design including bit length calculations, the report drop feature shall be designed. The purpose of this feature is to verify that the UE transmits every CSI report successfully to the gNB. If transmission of the report is unsuccessful, drop has occurred. Figure 14 illustrates the main design with addition of the report drop feature. To be able to verify successful transmissions of CSI reports, two additional traces should be taken into account.

UL_BWP trace

As introduced earlier, different NR SCS numerologies have an impact on CSI report transmission times in both periodic and aperiodic reporting. The purpose of *UL_BWP* trace is to configure numerologies for each UL BWP. Continuing to the next iteration loop in main function, these configured numerology values are used to verify correct timing for each CSI report.

APERIODIC_TRIG trace

APERIODIC_TRIG trace is representing aperiodic trigger DCI, which is including offset value Y, as introduced in Figure 6. When continuing to the next iteration loop, it is expected that time differences between configured aperiodic CSI reports and DCI trigger equal to offset value Y in a slot level.

Once relevant numerology and a possible trigger for an aperiodic report are configured, the report drop check function may be executed. This function is located after a new CSI report item has been subscribed from the queue of traces, as shown in Figure 14. The functionality of aperiodic and periodic report drop verifications differ from each other because periodic reporting is based on periodicity and offset values, while aperiodic reporting is based on triggering. In aperiodic reporting, *UL_BWP* trace triggers the aperiodic report, and the CSI report is expected to be sent at the time slot after the configured offset value in the next iteration loop of the main function. In periodic reporting, periodicity time between two CSI reports is calculated between two consecutive main function iteration loops, expecting that new subscribed item is configured with the same CSI report id. Further description of this function is represented in Chapter 4.

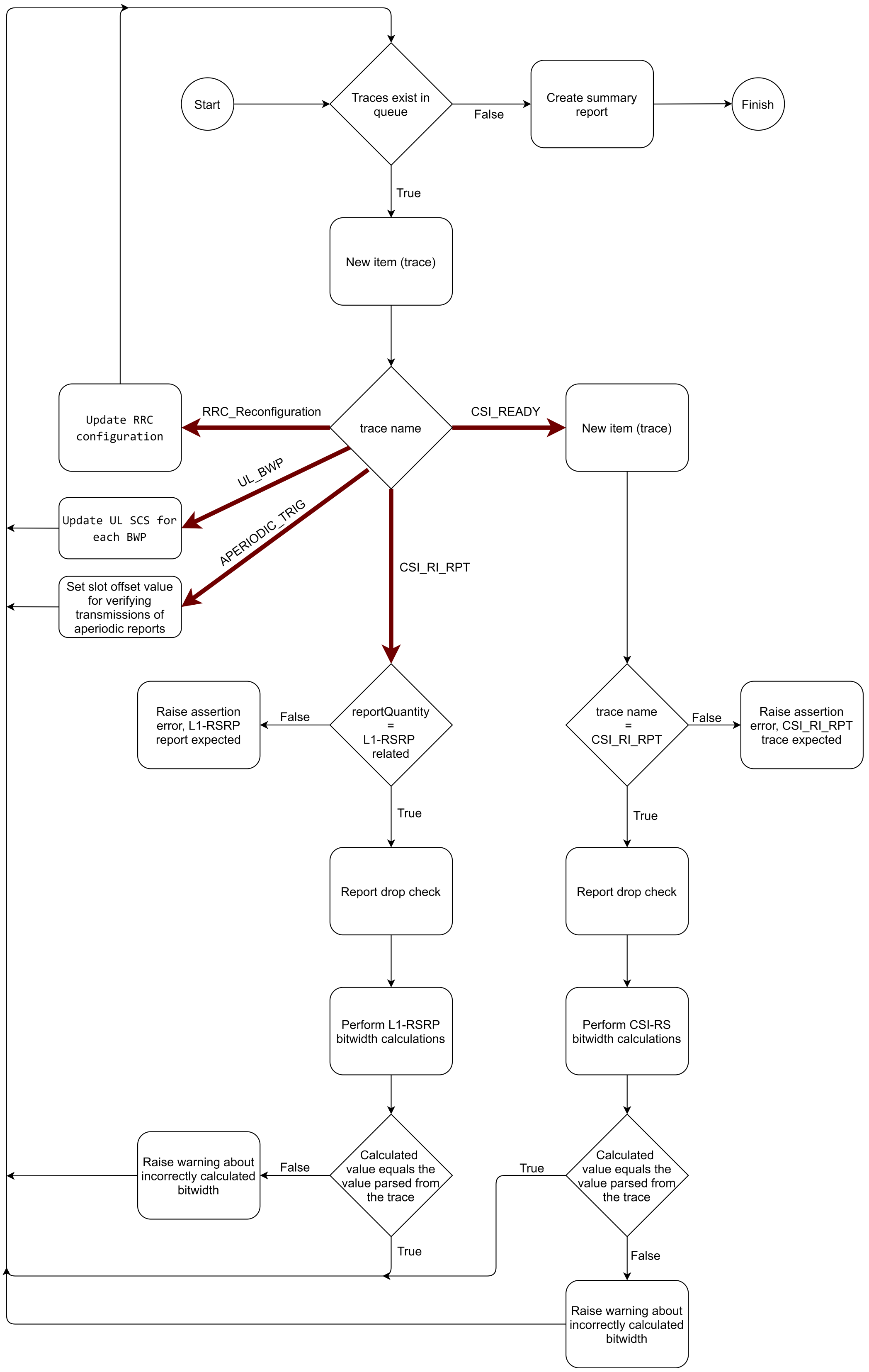


Figure 14. State diagram of the main functionality design with addition of report drop functionality.

3.6. CSI-RS Related Bitwidth Calculations

3GPP technical specification about multiplexing and channel coding define the creation of different reporting quantities in CSI report [2]. It includes calculation of bit lengths for each quantity, and in which order quantities shall be reported within the report. The system should be designed in a such way, that sizes of quantities are calculated and CSI reports are formed according to this technical specification. Following sections will introduce mapping orders, bit length calculations, and zero padding bit calculations for acquisition reports, based on 3GPP specification [2].

3.6.1. Report Quantity Bit Length Calculations

To verify that bit length calculations of report quantities are following 3GPP technical specifications, multiplexing and channel coding specification shall be explored [2]. This specification defines bitwidth calculations for report quantities, such as CRI, LI, RI, and CQI, as illustrated in Table 2. As mentioned earlier, type I single panel configuration is the most common in commercial networks. Therefore, the system will support only type I single panel codebook configurations. In addition, rank support is restricted to cover only ranks 1-4.

Table 2. Bit length calculation formulas of wideband (WB) CQI, subband (SB) CQI, RI, LI, and CRI quantities. Adapted from [2]

Bit lengths of CRI, LI, RI and CQI (type I single panel, ranks 1-4)				
WB CQI	**1 antenna port**	**2 antenna ports**	**4 antenna ports**	**>4 antenna ports**
	4	4	4	4
SB CQI	2	2	2	2
RI	0	min(1, ceil($\log_2 n_{RI}$))	min(2, ceil($\log_2 n_{RI}$))	ceil($\log_2 n_{RI}$)
LI	0	ceil($\log_2 v$)	ceil(min(2, $\log_2 v$))	cel(min(2, $\log_2 v$))
CRI	ceil($\log_2(K_s^{CSI\text{-}RS}$)	ceil($\log_2(K_s^{CSI\text{-}RS}$)	ceil($\log_2(K_s^{CSI\text{-}RS}$)	ceil($\log_2(K_s^{CSI\text{-}RS}$)

As it can be seen in the Table 2, CQI calculation either with subband or wideband configuration is not requiring any usage of mathematical calculation formulas as in RI, LI, and CRI calculations. The calculation for CQI quantity is only based on the number of antenna ports. When it comes to RI, LI and CRI, couple of terms shall be explained. In RI calculation, n_{RI} parameter represents the number of allowed RI values. For example, if *ri-Restriction* codebook configuration parameter is set to value of 3, there are only two possible ranks to be reported and n_{RI} equals value of 2. Calculation of LI quantity is based on the value v, which indicates value of the rank. Lastly, CRI quantity is based on the number of CSI-RS resources in the corresponding resource set, which is defined in K_s^{CSI-RS} value [2].

The bit length calculation of PMI quantity is much more complicated, compared to other report quantities. PMI calculation is divided into two information fields X_1 and X_2. Information field X_1 is used for wideband PMI reporting, while X_2 field is used for either wideband PMI reporting or for each subband PMI reporting. Also, reporting depends on configured *codebookMode*, which can be set to either value of 1 or value of 2. Tables 3 and 4 illustrate PMI calculations for reports with more than two CSI-RS antenna ports based on measured rank, antenna port count, values N_1; N_2; O_1; O_2, and configured *codebookMode* [2]. In case of single panel configuration with two CSI-RS antenna ports, bit length of PMI is 2 for rank 1, and 1 for rank 2. With a single CSI-RS antenna port configuration, bit length of PMI shall be 0. [2].

Table 3. Single panel type I bit length calculations of codebook 1 PMI fields X_1 and X_2. Adapted from [2]

PMI X_1 and X_2 bit lengths, codebookMode = 1		
	X_1 bit length for wideband PMI	**X_2 bit length for wideband PMI or per subband PMI**
Rank = 1, antenna port count > 2, N2 > 1	log$_2$(N1*O1) + log$_2$(N2*O2)	2
Rank = 1, antenna port count > 2, N2 = 1	log$_2$(N1*O1) + log$_2$(N2*O2)	2
Rank = 2, antenna port count = 4, N2 = 1	log$_2$(N1*O1) + log$_2$(N2*O2) + 1	1
Rank = 2, antenna port count > 4, N2 = 1	log$_2$(N1*O1) + log$_2$(N2*O2) + 2	1
Rank = 2,antenna port count > 4, N2 = 1	log$_2$(N1*O1) + log$_2$(N2*O2) + 2	1
Rank = 3 or 4, antenna port count = 4, N2 = 1	log$_2$(N1*O1) + log$_2$(N2*O2)	1
Rank = 3 or 4, antenna port count = 8 or 12, N2 = 1	log$_2$(N1*O1) + log$_2$(N2*O2) + 2	1
Rank = 3 or 4, antenna port count > 16, N2 = 1	log$_2$(N1*O1) + log$_2$(N2*O2) + 2	1

Table 4. Single panel type I bit length calculations of codebook 2 PMI fields X_1 and X_2. Adapted from [2]

PMI X_1 and X_2 bit lengths, codebookMode = 2		
	X_1 bit length for wideband PMI	**X_2 bit length for wideband PMI or per subband PMI**
Rank = 1, antenna port count > 2, N2 > 1	log$_2$((N1*O1 / 2)) + log$_2$((N2*O2) / 2)	4
Rank = 1, antenna port count > 2, N2 = 1	log$_2$((N1*O1 / 2))	4
Rank = 2, antenna port count = 4, N2 = 1	log$_2$((N1*O1 / 2)) + 1	3
Rank = 2, antenna port count > 4, N2 = 1	log$_2$((N1*O1 / 2)) + log$_2$((N2*O2) / 2) + 2	3
Rank = 2,antenna port count > 4, N2 = 1	log$_2$((N1*O1 / 2)) + 2	1
Rank = 3 or 4, antenna port count = 4, N2 = 1	log$_2$(N1*O1) + log$_2$(N2*O2)	1
Rank = 3 or 4, antenna port count = 8 or 12, N2 = 1	log$_2$(N1*O1) + log$_2$(N2*O2) + 2	1
Rank = 3 or 4, antenna port count > 16, N2 = 1	log$_2$(N1*O1) + log$_2$(N2*O2) + 2	1

3.6.2. Mapping Order of Report Quantities

As introduced earlier, 3GPP provides mapping orders for CSI fields within each CSI report, depending on frequency granularity and type of the report. CSI reports can be transmitted either in one or two parts. The CSI report is formed in one part in case PMI and CQI report quantities are configured as wideband granularity. Otherwise, the CSI report is transmitted in two parts. The field of each report quantity in CSI report shall be empty if the quantity is not configured by higher layer parameter *reportQuantity* [2]. In the context of this work, these mapping orders are only used to define in which CSI report part each configured report quantity should be placed. The mapping order itself inside of each formed report is irrelevant in terms of the system functionality. To be able to map and perform appropriate calculations in different CSI report forming scenarios by the script, four different acquisition CSI report parts shall be described [2].

One-part CSI report

The first and simplest type of CSI report mapping format is a one-part report, which is represented in Figure 15. One-part reporting is used when both higher layer parameters *pmi-FormatIndicator* and *cqi-FormatIndicator* are configured with wideband frequency granularity, signifying that subband reporting is not needed [2]. Wideband CQI for the second transport block (TB) can be ignored from the design of this system, since it is supporting only ranks 1-4 denoting that second TB reporting is not required.

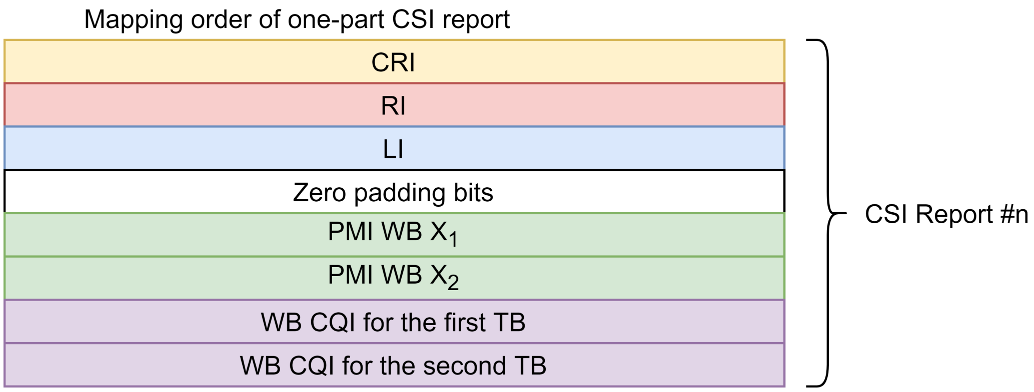


Figure 15. Report quantity mapping order of one-part CSI report. Adapted from [2].

Two-part CSI report, CSI part 1

In the case that higher layer parameter *pmi-FormatIndicator* or *cqi-FormatIndicator* is configured with subband frequency granularity, two-part reporting is expected [2]. Figure 16 represents CSI fields of CSI part 1. Since this thesis is focusing only on type I codebook configuration, the field of indicator of the number of non-zero wideband amplitude coefficients for layers can be excluded from the design of the log processing system. CSI report part 1 is reported together with two other parts including wideband CSI part 2, and subband CSI part 2.

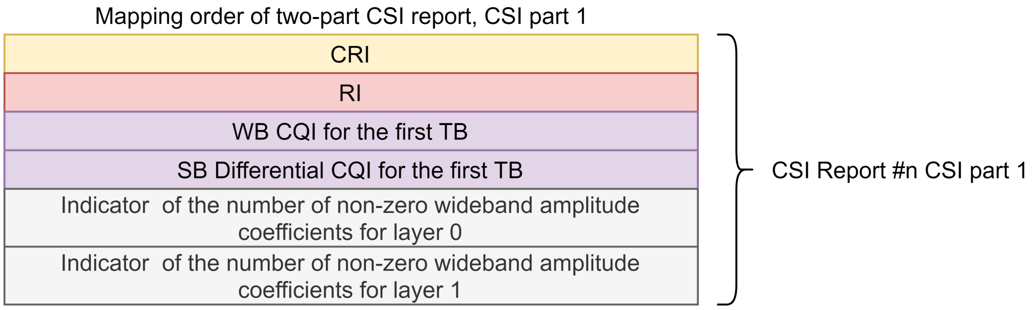


Figure 16. Report quantity mapping order of two-part CSI report, CSI part 1. Adapted from [2].

Two-part CSI report, CSI part 2 WB

The second part of two-part CSI reporting is divided into two categories including CSI part 2 wideband, and CSI part 2 subband. Figure 17 represents CSI fields of CSI part 2 wideband including WB CQI for the second TB, LI, PMI WB X_1, and PMI WB X_2 [2]. As well, the second TB WB CQI can be ignored in the design of the system developed.

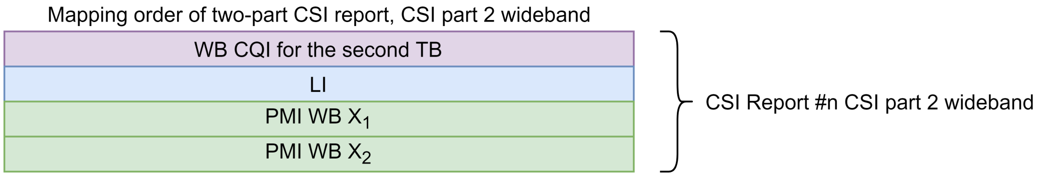


Figure 17. Report quantity mapping order of two-part CSI report, CSI part 2 wideband. Adapted from [2].

Two-part CSI report, CSI part 2 SB

As mentioned previously, report quantities with subband configuration are reported in the second part of CSI report. Figure 18 represents CSI fields of CSI part 2 subband. Even and odd subbands configured in *csi-ReportingBand* are divided into two different CSI fields of CSI report, for both differential CQI subbands and PMI information field X_2 subbands [2]. Again, differential subband CQI for the second TB can be excluded from the design of the developed system.

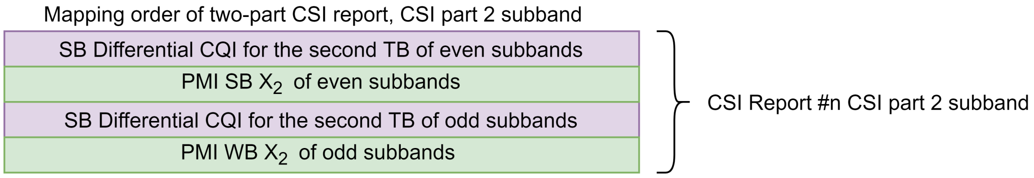


Figure 18. Report quantity mapping order of two-part CSI report, CSI part 2 subband. Adapted from [2].

3.7. L1-RSRP Related Bitwidth Calculations

L1-RSRP related BM reports are another type of CSI reports in addition to acquisition reports. Bit length calculations of these reports are reasonably more straightforward including the fields: CRI, SS/PBCH block resource indicator (SSBRI), RSRP, and differential RSRP. The Table 5 represents calculation formulas of fields. CRI calculation is similar as in acquisition reports, K_s^{CSI-RS} represents the number of CSI-RS resources in the corresponding resource set. In the situation, where reporting quantity is set to 'ssb-Index-RSRP', SSBRI calculation shall be considered in which K_s^{SSB} represents the configured number of SS/PBCH blocks in the corresponding resource set. The size of RSRP is a fixed value of 7, and for each differential RSRP value the size is 4 [2].

Table 5. BM report bit length calculations of CRI, SSBRI, RSRP, and differential RSRP values. Adapted from [2]

Bit lengths of CRI, SSBRI, and RSRP	
CRI	$\log_2(K_s^{\text{CSI-RS}})$
SSBRI	$\log_2(K_s^{\text{SSB}})$
RSRP	7
Differential RSRP	4

Also, BM report forming is more straightforward because the frequency granularity of L1-RSRP related reports is always wideband. Therefore, the CSI report is always formed as one-part report. Figure 19 represents the mapping order of CRI, SSBRI, RSRP, and differential values of BM CSI report. Depending on resource configuration, report includes up to four either CRI or SSBRI resource indicators, one RSRP value, and up to four differential RSRP values [2].

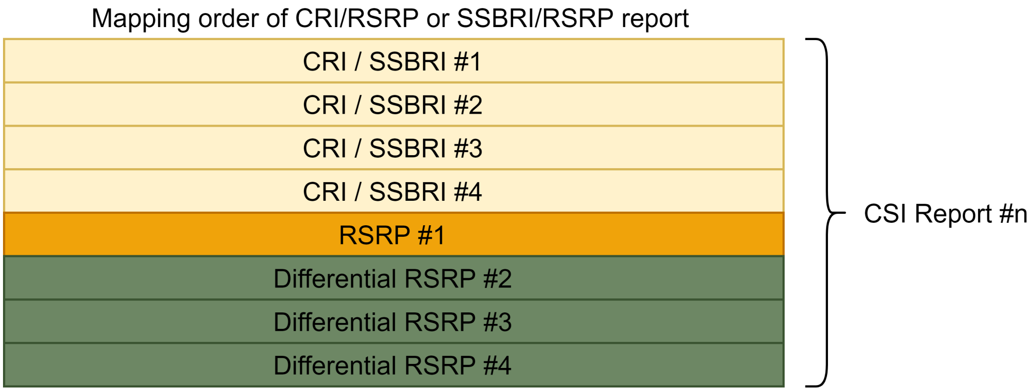


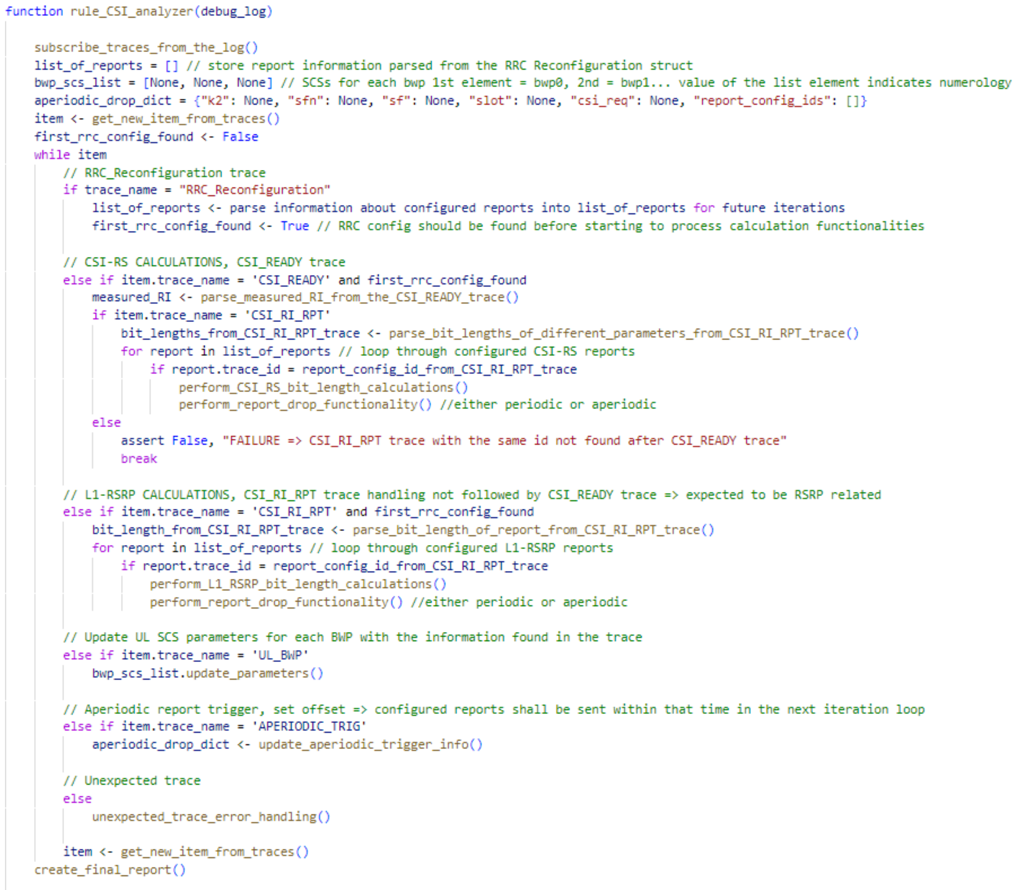
Figure 19. Mapping order of L1-RSRP related BM report. Adapted from [2].

4. SYSTEM FUNCTIONALITY IMPLEMENTATION

This chapter focuses on implementation of the automated log processing system, providing a more detailed explanation of different parts of software functionalities in a pseudocode format.

4.1. Main Function

In previous chapter, Figure 14 included design representation of the main function in a state diagram. The following figure 20 represents pseudocode implementation of the main function. This function takes a debug log as input, subscribes desired traces from the log, and processes these traces in loop iterations. In each iteration, a new item is selected from a list of subscribed traces and will be processed in a specific statement, introduced in Table 1. These statements are including functionalities introduced in this work: calculation of bit lengths, drop verification of reports, update of UL subcarrier spacings for BWPs, and trigger of aperiodic reports. In addition to the design, error handling is added for unexpected traces. After all traces in the list of subscribed traces have been processed, the final report is created. The next chapter related to the testing of the system will introduce examples of final reports in different use cases.



```
function rule_CSI_analyzer(debug_log)

    subscribe_traces_from_the_log()
    list_of_reports = [] // store report information parsed from the RRC Reconfiguration struct
    bwp_scs_list = [None, None, None] // SCSs for each bwp 1st element = bwp0, 2nd = bwp1... value of the list element indicates numerology
    aperiodic_drop_dict = {"k2": None, "sfn": None, "sf": None, "slot": None, "csi_req": None, "report_config_ids": []}
    item <- get_new_item_from_traces()
    first_rrc_config_found <- False
    while item
        // RRC_Reconfiguration trace
        if trace_name = "RRC_Reconfiguration"
            list_of_reports <- parse information about configured reports into list_of_reports for future iterations
            first_rrc_config_found <- True // RRC config should be found before starting to process calculation functionalities

        // CSI-RS CALCULATIONS, CSI_READY trace
        else if item.trace_name = 'CSI_READY' and first_rrc_config_found
            measured_RI <- parse_measured_RI_from_the_CSI_READY_trace()
            if item.trace_name = 'CSI_RI_RPT'
                bit_lengths_from_CSI_RI_RPT_trace <- parse_bit_lengths_of_different_parameters_from_CSI_RI_RPT_trace()
                for report in list_of_reports // loop through configured CSI-RS reports
                    if report.trace_id = report_config_id_from_CSI_RI_RPT_trace
                        perform_CSI_RS_bit_length_calculations()
                        perform_report_drop_functionality() //either periodic or aperiodic
            else
                assert False, "FAILURE => CSI_RI_RPT trace with the same id not found after CSI_READY trace"
                break

        // L1-RSRP CALCULATIONS, CSI_RI_RPT trace handling not followed by CSI_READY trace => expected to be RSRP related
        else if item.trace_name = 'CSI_RI_RPT' and first_rrc_config_found
            bit_length_from_CSI_RI_RPT_trace <- parse_bit_length_of_report_from_CSI_RI_RPT_trace()
            for report in list_of_reports // loop through configured L1-RSRP reports
                if report.trace_id = report_config_id_from_CSI_RI_RPT_trace
                    perform_L1_RSRP_bit_length_calculations()
                    perform_report_drop_functionality() //either periodic or aperiodic

        // Update UL SCS parameters for each BWP with the information found in the trace
        else if item.trace_name = 'UL_BWP'
            bwp_scs_list.update_parameters()

        // Aperiodic report trigger, set offset => configured reports shall be sent within that time in the next iteration loop
        else if item.trace_name = 'APERIODIC_TRIG'
            aperiodic_drop_dict <- update_aperiodic_trigger_info()

        // Unexpected trace
        else
            unexpected_trace_error_handling()

        item <- get_new_item_from_traces()
    create_final_report()
```

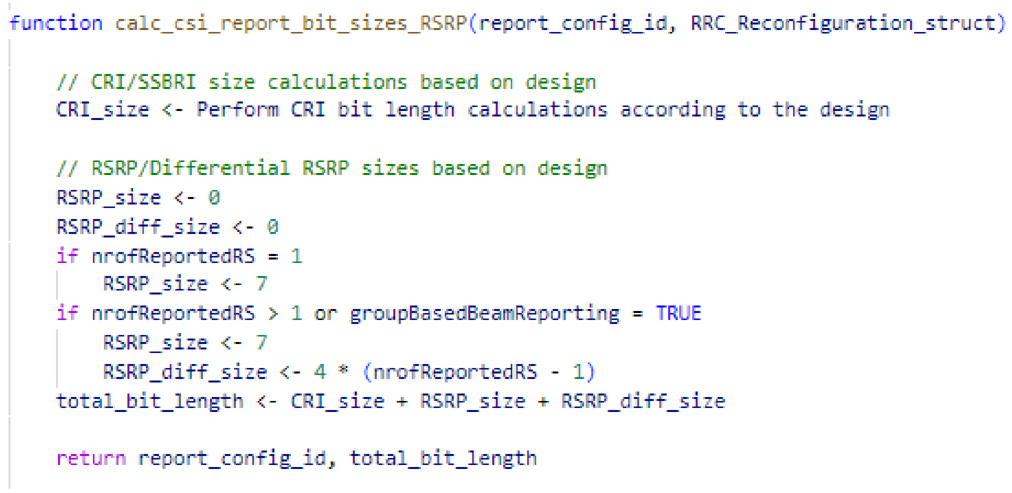
Figure 20. High level pseudocode representation of main functionality of the system.

4.2. Bitwidth Calculation Functions

As described in previous chapter, bit length calculations can be divided into two parts, BM related L1-RSRP report calculations and acquisition CSI-RS report calculations. The main function includes two statements for both of these calculation types, as illustrated in Figure 20. The following sections will introduce implementations of L1-RSRP and CSI-RS bit length calculation functions. Both of these functions return calculated bit lengths according to 3GPP specification [2], which are compared to bit length values parsed from the *CSI_RI_RPT* trace. If the values differ from each other, warning will be raised about incorrect bit lengths for further investigation.

4.2.1. RSRP Related Bitwidth Calculation Function

Figure 21 represents implementation of L1-RSRP bit length calculation function. As input, the function takes RRC configuration struct along with the report ID, which are used to find the right report information from the list of configured reports. Calculation of the CRI size is described in Chapter 3. After performing calculations, the function will return the report configuration ID and total bit length of the one-part report.

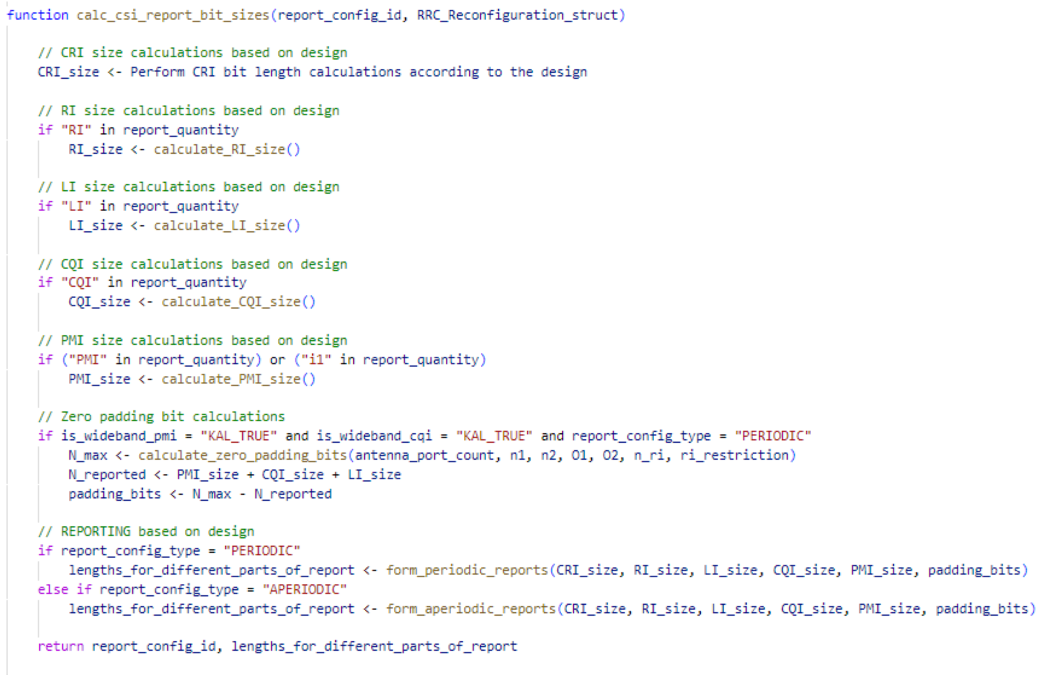


```
function calc_csi_report_bit_sizes_RSRP(report_config_id, RRC_Reconfiguration_struct)

    // CRI/SSBRI size calculations based on design
    CRI_size <- Perform CRI bit length calculations according to the design

    // RSRP/Differential RSRP sizes based on design
    RSRP_size <- 0
    RSRP_diff_size <- 0
    if nrofReportedRS = 1
        RSRP_size <- 7
    if nrofReportedRS > 1 or groupBasedBeamReporting = TRUE
        RSRP_size <- 7
        RSRP_diff_size <- 4 * (nrofReportedRS - 1)
    total_bit_length <- CRI_size + RSRP_size + RSRP_diff_size

    return report_config_id, total_bit_length
```

Figure 21. Pseudocode representation of L1-RSRP bit length calculation function.

4.2.2. CSI-RS Bitwidth Calculation Function

As introduced, another type of bit length calculation function is related to acquisition reports, including calculations for different report quantities. This function is represented in Figure 22. Calculation formulas for quantities CRI, RI, LI, and CQI can be found from the Table 2, and for PMI from Tables 3 and 4. Again, calculations are based on RRC configuration and the configured report ID. In the case of a one-part CSI report, zero padding bits shall be calculated. For this, N_{max} value is calculated in a separate function, which will be described in the next section. At the end of the function, the report will be formed depending on the type of the report according to the design. Different parts of reports will be returned with report configuration id for further processing in main function.

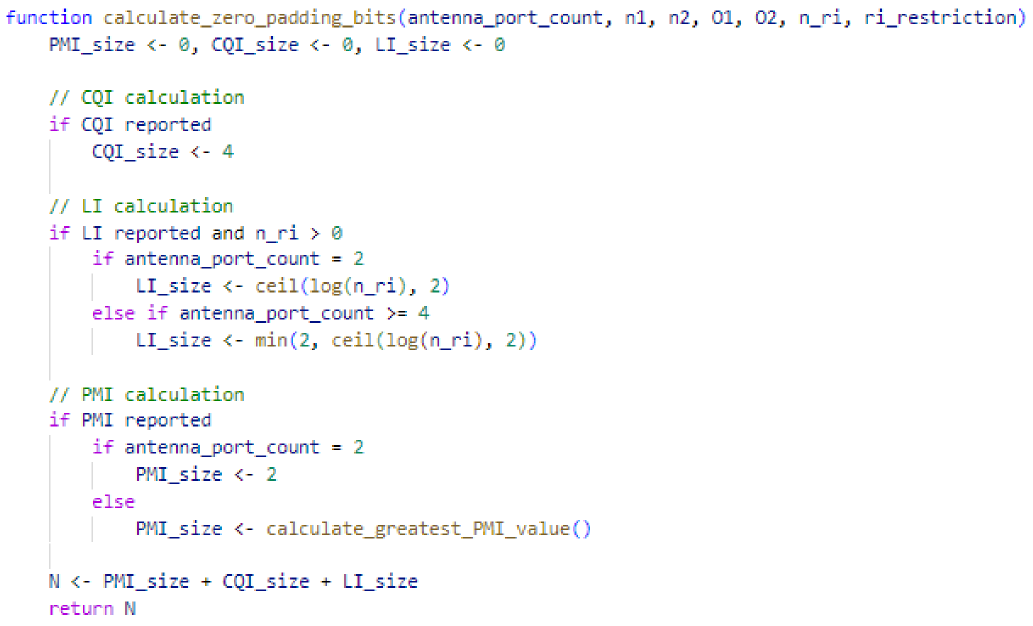


```
function calc_csi_report_bit_sizes(report_config_id, RRC_Reconfiguration_struct)

    // CRI size calculations based on design
    CRI_size <- Perform CRI bit length calculations according to the design

    // RI size calculations based on design
    if "RI" in report_quantity
        RI_size <- calculate_RI_size()

    // LI size calculations based on design
    if "LI" in report_quantity
        LI_size <- calculate_LI_size()

    // CQI size calculations based on design
    if "CQI" in report_quantity
        CQI_size <- calculate_CQI_size()

    // PMI size calculations based on design
    if ("PMI" in report_quantity) or ("i1" in report_quantity)
        PMI_size <- calculate_PMI_size()

    // Zero padding bit calculations
    if is_wideband_pmi = "KAL_TRUE" and is_wideband_cqi = "KAL_TRUE" and report_config_type = "PERIODIC"
        N_max <- calculate_zero_padding_bits(antenna_port_count, n1, n2, O1, O2, n_ri, ri_restriction)
        N_reported <- PMI_size + CQI_size + LI_size
        padding_bits <- N_max - N_reported

    // REPORTING based on design
    if report_config_type = "PERIODIC"
        lengths_for_different_parts_of_report <- form_periodic_reports(CRI_size, RI_size, LI_size, CQI_size, PMI_size, padding_bits)
    else if report_config_type = "APERIODIC"
        lengths_for_different_parts_of_report <- form_aperiodic_reports(CRI_size, RI_size, LI_size, CQI_size, PMI_size, padding_bits)

    return report_config_id, lengths_for_different_parts_of_report
```

Figure 22. Pseudocode representation of acquisition report bit length calculation function.

4.2.3. Zero Padding Bits

To keep one-part CSI acquisition reports fixed in size during reporting, zero padding bits shall be calculated. Zero padding bit calculation function is represented in Figure 23, and it is called in acquisition report calculation function. Since the work of this thesis is restricted to ranks 1-4, calculations are reasonably simple. Calculations of report quantities CQI and LI are following formulas introduced in Table 2. PMI calculation is slightly different because the output will be the greatest PMI value calculated from all possible ranks, instead of reported rank. However, the same formulas are used for PMI calculations, introduced in Tables 3 and 4.



```
function calculate_zero_padding_bits(antenna_port_count, n1, n2, O1, O2, n_ri, ri_restriction)
    PMI_size <- 0, CQI_size <- 0, LI_size <- 0

    // CQI calculation
    if CQI reported
        CQI_size <- 4

    // LI calculation
    if LI reported and n_ri > 0
        if antenna_port_count = 2
            LI_size <- ceil(log(n_ri), 2)
        else if antenna_port_count >= 4
            LI_size <- min(2, ceil(log(n_ri), 2))

    // PMI calculation
    if PMI reported
        if antenna_port_count = 2
            PMI_size <- 2
        else
            PMI_size <- calculate_greatest_PMI_value()

    N <- PMI_size + CQI_size + LI_size
    return N
```

Figure 23. Pseudocode representation of zero padding bit calculation function.

4.3. Csi Report Drop Feature Functionality

In order to implement CSI report drop functionality, a further analysis of frame structure and timing in 5G NR is required. Figure 24 represents the frame structure and timing of 5G NR. As illustrated, frames are labeled as values in the range from 0 to 1023 due to 10-bit value of system frame number (SFN), which represents timing in 10ms intervals. Each frame contains 10 subframes, and every subframe has a timing of 1ms. Therefore, the timing of each frame is 10 ms and timing interval of whole SFN is 10.24 seconds. Furthermore, the amount of slots in each subframe may vary depending on configured numerologies [16], which are discussed in Chapter 2. Table 6 represents the number of slots per frame and subframe for each configured SCS numerology. In the implementation phase of the system, these timing factories are crucial to be considered. Implementation is divided into two parts, one for periodic reports and other for aperiodic reports, these will be described further in following sections.

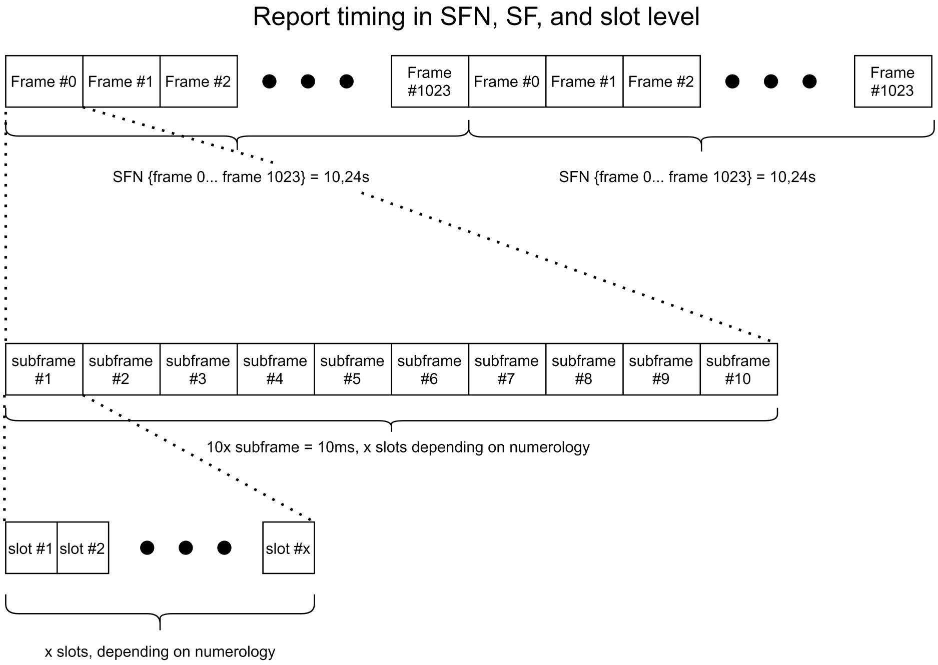


Figure 24. 5G NR frame structure and report timing in SFN, SF, and slot level.

Table 6. Number of slots per frame and subframe depending on numerology. Adapted from [16]

numerology	number of slots per frame	number of slots per subframe
0 = 15kHz	10	1
1 = 30 kHz	20	2
2 = 60 kHz	40	4
3 = 120 kHz	80	8
4 = 240 kHz	160	16

4.3.1. Periodic Report Drop Check

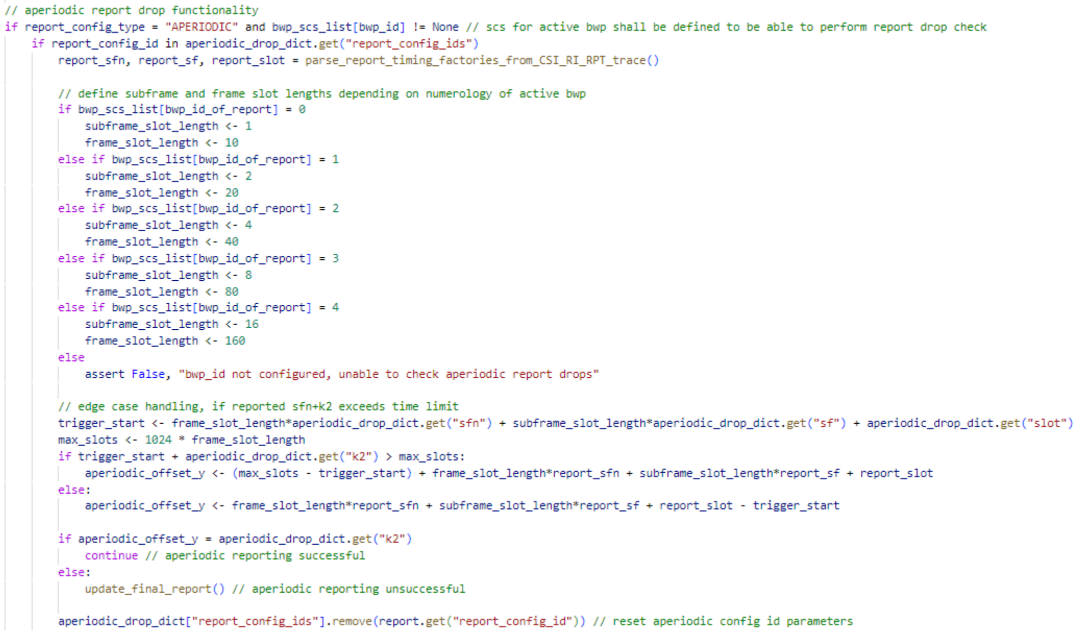
High level periodic report drop check function is represented in pseudocode format in Figure 25. The function is executed alongside the bit length calculation function, as illustrated in Figure 20 of the main function. Periodic report timing is verified according to the time difference between two consecutive reports with the same id between loop iterations of the main function. In periodic reporting, time differences are calculated on a slot level, as illustrated in Figure 24. In order to calculate time difference between two reports, initial SFN, subframe (SF), and slot timing factories have to be defined. In future iterations, these previous values will be used as a reference for calculating timing difference. After performing this verification, previous initial values are replaced with the current iteration values, and again, the next iteration will use these values in calculations as a reference value. Importantly, the maximum number of slots shall be taken into account in timing calculations. If the expected time of the report exceeds the number of maximum slots, the calculation will continue from the value 0. As illustrated in the Figure 25, SCS for active BWP has to be defined, being able to perform time interval calculations, since numerology affects on report timing.

```
// Periodic report drop functionality
if report_config_type = "PERIODIC" and bwp_scs_list[bwp_id] != None // scs for active bwp shall be defined to be able to perform report drop check
    if sfn_of_report = None // not defined
        sfn_of_report <- set_first_sfn_value()
        sf_of_report <- set_first_sf_value()
        slot_of_report <- set_first_slot_value()
    else:
        if bwp_scs_list[bwp_id_of_report] = 0
            subframe_slot_length <- 1
            frame_slot_length <- 10
        else if bwp_scs_list[bwp_id_of_report] = 1
            subframe_slot_length <- 2
            frame_slot_length <- 20
        else if bwp_scs_list[bwp_id_of_report] = 2
            subframe_slot_length <- 4
            frame_slot_length <- 40
        else if bwp_scs_list[bwp_id_of_report] = 3
            subframe_slot_length <- 8
            frame_slot_length <- 80
        else if bwp_scs_list[bwp_id_of_report] = 4
            subframe_slot_length <- 16
            frame_slot_length <- 160
        else:
            assert False, "FAILURE, invalid bwp id/numerology"

        timing_start <- frame_slot_length*report.get("sfn") + subframe_slot_length*report.get("sf") + report.get("slot")
        max_slots <- 1024 * frame_slot_length
        if timing_start + periodicity > max_slots
            time_diff <- (max_slots - timing_start) + frame_slot_length*sfn_from_trace + subframe_slot_length*sf_from_trace + slot_from_trace
        else:
            time_diff <- frame_slot_length*sfn_from_trace + subframe_slot_length*sf_from_trace + slot_from_trace - timing_start

        if periodicity = time_diff
            continue // periodic reporting successful
        else:
            update_final_report() // periodic reporting unsuccessful
        sfn_of_report <- sfn_from_new_trace
        sf_of_report <- sf_from_new_trace
        slot_of_report <- slot_from_new_trace
```

Figure 25. High level representation of periodic report drop check function in a pseudocode format.

4.3.2. *Aperiodic Report Drop Check*

Another part of verifying CSI report UE transmissions is aperiodic report drop check. Instead of calculating time difference between two reports as in periodic case, the timing difference is calculated between aperiodic trigger and transmitted report. As described in Figure 20 of the design phase, aperiodic trigger trace *APERIODIC_TRIG* updates local trigger list *aperiodic_drop_dict* with offset information. Once the information is updated, the program will continue on the next iteration, and when the corresponding report is found, the calculation will be performed between time slots of report and trigger. However, the calculation of time difference is similar to the periodic case, with an exception of comparable value, which is aperiodic trigger in aperiodic case. To continue calculations in future iterations, reset of local list *aperiodic_drop_dict* parameters should be executed. High level representation of this feature is illustrated in Figure 26.



```
// aperiodic report drop functionality
if report_config_type = "APERIODIC" and bwp_scs_list[bwp_id] != None // scs for active bwp shall be defined to be able to perform report drop check
    if report_config_id in aperiodic_drop_dict.get("report_config_ids")
        report_sfn, report_sf, report_slot = parse_report_timing_factories_from_CSI_RI_RPT_trace()

        // define subframe and frame slot lengths depending on numerology of active bwp
        if bwp_scs_list[bwp_id_of_report] = 0
            subframe_slot_length <- 1
            frame_slot_length <- 10
        else if bwp_scs_list[bwp_id_of_report] = 1
            subframe_slot_length <- 2
            frame_slot_length <- 20
        else if bwp_scs_list[bwp_id_of_report] = 2
            subframe_slot_length <- 4
            frame_slot_length <- 40
        else if bwp_scs_list[bwp_id_of_report] = 3
            subframe_slot_length <- 8
            frame_slot_length <- 80
        else if bwp_scs_list[bwp_id_of_report] = 4
            subframe_slot_length <- 16
            frame_slot_length <- 160
        else
            assert False, "bwp_id not configured, unable to check aperiodic report drops"

        // edge case handling, if reported sfn+k2 exceeds time limit
        trigger_start <- frame_slot_length*aperiodic_drop_dict.get("sfn") + subframe_slot_length*aperiodic_drop_dict.get("sf") + aperiodic_drop_dict.get("slot")
        max_slots <- 1024 * frame_slot_length
        if trigger_start + aperiodic_drop_dict.get("k2") > max_slots:
            aperiodic_offset_y <- (max_slots - trigger_start) + frame_slot_length*report_sfn + subframe_slot_length*report_sf + report_slot
        else:
            aperiodic_offset_y <- frame_slot_length*report_sfn + subframe_slot_length*report_sf + report_slot - trigger_start

        if aperiodic_offset_y = aperiodic_drop_dict.get("k2")
            continue // aperiodic reporting successful
        else:
            update_final_report() // aperiodic reporting unsuccessful

        aperiodic_drop_dict["report_config_ids"].remove(report.get("report_config_id")) // reset aperiodic config id parameters
```

Figure 26. High level representation of aperiodic report drop check function in a pseudocode format.

5. TESTING AND VERIFICATION OF THE SYSTEM

This chapter focuses on testing and verification of functionalities of the developed system giving a few examples how this kind of system functionality could be verified. The purpose of the testing phase is to verify that the system manages to find incorrectly calculated bit lengths and dropped CSI reports by intentionally creating bugs on the UE software. Once different scenarios of testing have been gone through, various outputs of the script will be illustrated with figures. In this section, complete testing with all possible conditions is not conducted, since the intention is to provide few examples that can be applied in more precise use case tests. Lastly, results will be discussed and evaluated.

5.1. Testing Plan

The testing plan includes three different test cases for verifying bit length calculation functionality and report drop check functionality:

1. Incorrect rank test
2. Test of dropped cross-carrier reports
3. Periodic CSI report drop test

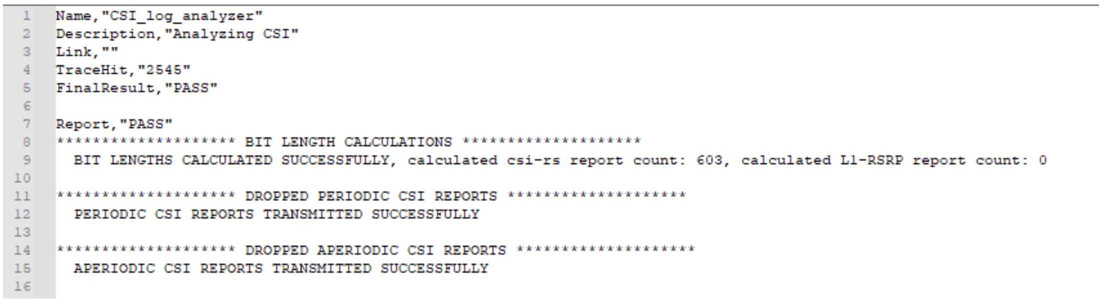
In terms of bit length functionality, the first test case relates to incorrect RI measurements in UE. According to the design, measured rank value affects LI, PMI, and zero padding bit calculations. To verify that the system finds incorrect results of bit length calculations, UE software is modified by changing the measured rank to a different value, creating an intentional bug. Therefore, the system performs bit length calculations using incorrect rank value, and bit lengths shall not match. It is worth mentioning that measured rank has no effect on the one-part acquisition report size, because RI is calculated using the greatest possible rank value, and zero padding bits are used to keep the size of the report fixed. Therefore, this test shall be performed with a two-part CSI report configuration.

The second test case in terms of both total bit length calculation and report dropping is related to cross-carrier scheduling, in which scheduling grants and assignments are transmitted on a different serving cell than the corresponding data [27]. In aperiodic reporting, multiple reports can be triggered by the same trigger, and these reports may be located in both primary cell (PCell) and secondary cell (SCell). In this test, configured SCell reports are intentionally dropped by creating a bug in UE software, failing total bit length calculations. Total combined bit lengths for transmitted reports shall not be correct, since SCell configured reports are not transmitted.

To simulate report dropping, a simple solution is to create a bug on UE software by skipping the transmission of every other CSI report. The system is expecting these missing reports to appear, resulting in the time difference between two consecutive reports being too long, and the system recognizes the issue as dropped CSI report.

5.2. Verifying the Functionalities of the System by Different Tests

To verify the correctness of the system functionalities, three different test cases are conducted. In these tests, UE software is built and simulated with intentionally created bugs outputting the debug log, which is afterward executed by the automated log processing system. Figure 27 illustrates the output of the system when CSI reports are transmitted successfully with correct bit lengths.

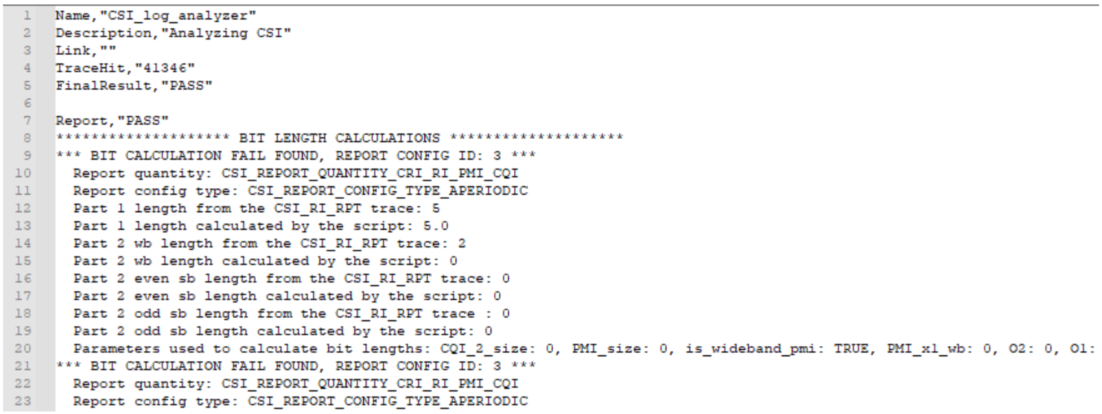


```
Name,"CSI_log_analyzer"
Description,"Analyzing CSI"
Link,""
TraceHit,"2545"
FinalResult,"PASS"

Report,"PASS"
******************** BIT LENGTH CALCULATIONS ********************
  BIT LENGTHS CALCULATED SUCCESSFULLY, calculated csi-rs report count: 603, calculated L1-RSRP report count: 0

******************** DROPPED PERIODIC CSI REPORTS ********************
  PERIODIC CSI REPORTS TRANSMITTED SUCCESSFULLY

******************** DROPPED APERIODIC CSI REPORTS ********************
  APERIODIC CSI REPORTS TRANSMITTED SUCCESSFULLY

```

Figure 27. System output without any corresponding issues in the debug log.

5.2.1. Incorrect Rank Test

As mentioned in the testing plan, the first testing case verifies bit length functionality by modifying measured rank on *CSI_READY* trace. In this test setup, the number of antenna ports is two, quantity of the report is configured to *cri-RI-PMI-CQI*, and configuration type of the report is aperiodic. Measured rank is modified into the trace by hard-coding measured rank to value 3, which corresponds to rank 4. Since the real measured rank in this test is 1, incorrect rank value 3 shall affect on bit length of PMI quantity. According to the design, expected bit length shall be 2 for rank 1 with two antenna ports configuration. Because modified measured rank is 4, the rank is invalid creating an exception in the system. The output of the system is represented in Figure 28. As it can be seen in the figure, the report is aperiodic in two parts and PMI fields shall be reported on the second part. Because modified rank value is invalid for two antenna port configuration, part 2 wideband length equals value 0. Therefore, bit length calculations do not match and the issue is identified. Output of the log includes relevant information for debugging the issue, which are parsed from the log. Usually, it may require plenty of time to manually parse these information by hand.



```
Name,"CSI_log_analyzer"
Description,"Analyzing CSI"
Link,""
TraceHit,"41346"
FinalResult,"PASS"

Report,"PASS"
******************** BIT LENGTH CALCULATIONS ********************
*** BIT CALCULATION FAIL FOUND, REPORT CONFIG ID: 3 ***
  Report quantity: CSI_REPORT_QUANTITY_CRI_RI_PMI_CQI
  Report config type: CSI_REPORT_CONFIG_TYPE_APERIODIC
  Part 1 length from the CSI_RI_RPT trace: 5
  Part 1 length calculated by the script: 5.0
  Part 2 wb length from the CSI_RI_RPT trace: 2
  Part 2 wb length calculated by the script: 0
  Part 2 even sb length from the CSI_RI_RPT trace: 0
  Part 2 even sb length calculated by the script: 0
  Part 2 odd sb length from the CSI_RI_RPT trace : 0
  Part 2 odd sb length calculated by the script: 0
  Parameters used to calculate bit lengths: CQI_2_size: 0, PMI_size: 0, is_wideband_pmi: TRUE, PMI_x1_wb: 0, O2: 0, O1:
*** BIT CALCULATION FAIL FOUND, REPORT CONFIG ID: 3 ***
  Report quantity: CSI_REPORT_QUANTITY_CRI_RI_PMI_CQI
  Report config type: CSI_REPORT_CONFIG_TYPE_APERIODIC
```

Figure 28. Output of the system in bit length calculation failure situation.

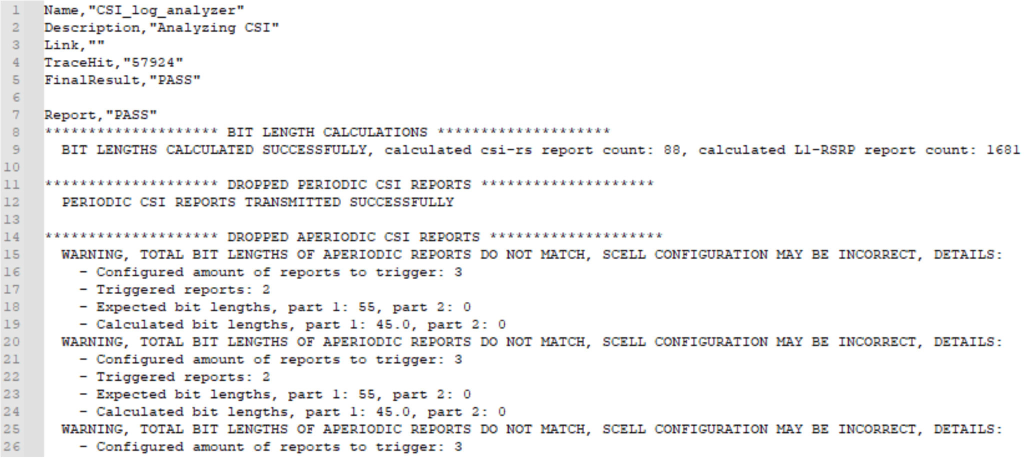
5.2.2. Test of Dropped Cross-Carrier Reports

Cross-carrier scheduling related test illustrates how non-configured SCell reports affect total bit lengths in aperiodic reporting. As mentioned in testing plan, some of CSI report resources might be located in a different serving cell. In aperiodic reporting, multiple reports can be triggered at the same time, and some of the reports can be configured in the SCell. If there is an issue, where SCell reports are missing for some reason, those reports cannot be transmitted. This results as incorrect bit lengths, when all triggered reports are combined with each other. The system will identify the fault, and it can be assumed that the report has been dropped due to incorrect configuration of SCell resources.

For verifying this functionality, three different CSI reports are configured for individual aperiodic triggering, with the following configurations:

1. index of the report: 0, *reportQuantity*: ’cri-RSRP’, bit length of the report: 10
2. index of the report: 1, *reportQuantity*: ’cri-RSRP’, bit length of the report: 10
3. index of the report: 2, *reportQuantity*: ’ssb-Index-RSRP’, bit length of the report: 35

From these configurations, resources for the report with index 1 are configured in SCell, and other report resources are in PCell. Again, a bug is created on UE software, in which SCell resources are intentionally dropped. Therefore, the size of the report with index 1 cannot be calculated and will not be transmitted. The total number of bits shall be incorrect, resulting in a size value of 45, as illustrated in Figure 29. The figure illustrates output of the script for this test.

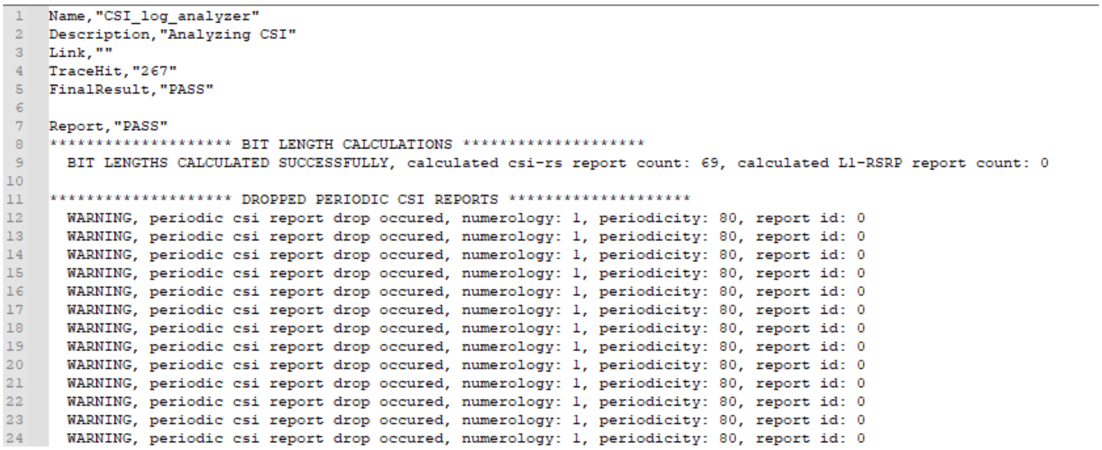


```
Name,"CSI_log_analyzer"
Description,"Analyzing CSI"
Link,""
TraceHit,"57924"
FinalResult,"PASS"

Report,"PASS"
******************** BIT LENGTH CALCULATIONS ********************
  BIT LENGTHS CALCULATED SUCCESSFULLY, calculated csi-rs report count: 88, calculated L1-RSRP report count: 1681

******************** DROPPED PERIODIC CSI REPORTS ********************
  PERIODIC CSI REPORTS TRANSMITTED SUCCESSFULLY

******************** DROPPED APERIODIC CSI REPORTS ********************
  WARNING, TOTAL BIT LENGTHS OF APERIODIC REPORTS DO NOT MATCH, SCELL CONFIGURATION MAY BE INCORRECT, DETAILS:
    - Configured amount of reports to trigger: 3
    - Triggered reports: 2
    - Expected bit lengths, part 1: 55, part 2: 0
    - Calculated bit lengths, part 1: 45.0, part 2: 0
  WARNING, TOTAL BIT LENGTHS OF APERIODIC REPORTS DO NOT MATCH, SCELL CONFIGURATION MAY BE INCORRECT, DETAILS:
    - Configured amount of reports to trigger: 3
    - Triggered reports: 2
    - Expected bit lengths, part 1: 55, part 2: 0
    - Calculated bit lengths, part 1: 45.0, part 2: 0
  WARNING, TOTAL BIT LENGTHS OF APERIODIC REPORTS DO NOT MATCH, SCELL CONFIGURATION MAY BE INCORRECT, DETAILS:
    - Configured amount of reports to trigger: 3
```

Figure 29. Output of the system in misconfigured SCell resource situation.

5.2.3. Periodic CSI Report Drop Test

The purpose of the third test is to verify the functionality of report drop-checking. This can be easily conducted by dropping every other transmission of the CSI report by creating an intentional bug in the UE software. In this test, the CSI report is configured as a periodic report, with a periodicity of value 80 and a numerology of value 1. Therefore, periodic reports should be transmitted every forty slots. Since every other report is dropped due to an intentional bug, the time difference between consecutive reports is 80 slots, and the system identifies the issue as a report drop. Figure 30 represents the output of the report when a report drop has occurred. In an aperiodic reporting case, this kind of test could be performed by increasing the offset value of the trigger, and the system could identify too lengthy time difference between the transmission of the report and the trigger.



```
Name,"CSI_log_analyzer"
Description,"Analyzing CSI"
Link,""
TraceHit,"267"
FinalResult,"PASS"

Report,"PASS"
******************** BIT LENGTH CALCULATIONS ********************
  BIT LENGTHS CALCULATED SUCCESSFULLY, calculated csi-rs report count: 69, calculated L1-RSRP report count: 0

******************** DROPPED PERIODIC CSI REPORTS ********************
  WARNING, periodic csi report drop occured, numerology: 1, periodicity: 80, report id: 0
  WARNING, periodic csi report drop occured, numerology: 1, periodicity: 80, report id: 0
  WARNING, periodic csi report drop occured, numerology: 1, periodicity: 80, report id: 0
  WARNING, periodic csi report drop occured, numerology: 1, periodicity: 80, report id: 0
  WARNING, periodic csi report drop occured, numerology: 1, periodicity: 80, report id: 0
  WARNING, periodic csi report drop occured, numerology: 1, periodicity: 80, report id: 0
  WARNING, periodic csi report drop occured, numerology: 1, periodicity: 80, report id: 0
  WARNING, periodic csi report drop occured, numerology: 1, periodicity: 80, report id: 0
  WARNING, periodic csi report drop occured, numerology: 1, periodicity: 80, report id: 0
  WARNING, periodic csi report drop occured, numerology: 1, periodicity: 80, report id: 0
  WARNING, periodic csi report drop occured, numerology: 1, periodicity: 80, report id: 0
  WARNING, periodic csi report drop occured, numerology: 1, periodicity: 80, report id: 0
  WARNING, periodic csi report drop occured, numerology: 1, periodicity: 80, report id: 0
```

Figure 30. Output of the system in periodic report drop failure situation.

5.2.4. Evaluation of the Results

As it can be seen from the system output figures, the output format is fairly simple. The system dumps the output into a text file since text files are usually relatively fast to process. Also, text files can be accessed easily from every device with general applications, therefore no special software is required. However, there are limitations on text file format if the system would be developed further producing larger outputs and the debug information would again need plenty of manual work. The intention of this work was to decrease the amount of manual debug log processing, not to increase it by making additional tricky elements. Therefore, another relevant option would be creating a simple user interface (UI), where the engineer would be able to filter outputs for the corresponding issue. After all, considering the scale of the system, the text file solution is the most suitable for this work.

In terms of functionalities, the system performed as expected. The main purpose of the system was to verify bit lengths of different reports and identify report drops in the whole debug log to speed up the debugging process. Output figures above illustrate that the system manages to identify incorrect bit length calculations and missing transmissions of CSI reports, according to the 3GPP specifications introduced. Exploring more detailed Figure 29, it can be seen that the number of relevant subscribed traces is 57924, which is an extremely huge amount, and would be processed manually very slowly. In these kind of cases, the debug log is investigated from specific timing locations where the issue is expected to occur since it is nearly impossible to go through the whole debug log manually. The significant benefit of the system is the ability to go through all traces rapidly, allowing unexpected issues to be identified outside of the investigated timing location.

6. DISCUSSION

Since compliance of CSI reports to the standard is key, the most challenging part of the project was beginning the development, because the system development required quite a lot of knowledge about the CSI feature in 5G NR. The initial plan was to create a system that supports all different types of codebook configurations, but the system ended up having only single panel Type I configuration support because it was found out that Type I single panel configurations are considerable most commonly used. Additionally, it was most reasonable to restrict rank support to ranks 1-4, and different reporting types to aperiodic and periodic. With these restrictions, the thesis can focus in more detail on these particular configurations without making it too confusing but still maintaining relevance due to the importance of the case considered.

Again, the restriction of the system verification and testing plan was challenging, because the system could have been tested with an endless amount of use cases. Therefore, three different test cases were conducted to cover the widest and most relevant verification areas possible. These tests were conducted in a such way that adapting tests to other use cases would be as straightforward as possible.

Since the biggest challenge in this work was how to restrict the topic to keep it understandable, future work includes many opportunities. In future work, the system could be further developed providing support for other configuration types, such as Type II configurations. In this case, it would be worth considering adding more bit length calculation functions to avoid too messy system implementation. Also, rank support would be extended for all eight ranks provided by 3GPP. From the CSI point of view, the most notable impact would be additional bit fields for second transport blocks, as illustrated in Figure 15, for example. Since the system is currently supporting only aperiodic and periodic verification of transmitted CSI reports, the functionality could be extended to cover also semi-persistent reports. The semi-persistent feature would be some kind of combination of aperiodic and periodic report drop check functions introduced in this work, therefore, it would be straightforward extending it in this direction. An ultimate future goal would be creating smarter system using artificial intelligence (AI), that could identify all possible known errors regarding CSI failures and handle errors without any assistance of an engineer. However, there are an endless amount of limitations, because communication networks and devices are constantly evolving and it is nearly impossible to create that wide AI-powered system since its training could be hard.

Overall, development process of the system was a great success. As mentioned, manual debugging of massive logs might take several hours or even full working day, while the automated system can handle the same log in a few minutes. This working time could be released for another assignments, such as development of new features, increasing efficiency of the working time. The system handles its duties by mimicking the human behavior of manual error debugging, but it is capable of processing quickly a huge amount of information. Thus, this system represents an important tool to speed up the error debugging process. Of course, it is possible that the fault reason is totally unknown and the system would not be able to identify the cause. However, this system excludes the most common CSI related issues and the cause could be explored elsewhere.

7. SUMMARY

The intention of this thesis was to develop an automated system for investigating debug logs in terms of the 5G NR CSI feature. The main aspect is to verify that transmissions of CSI reports are according to technical specifications provided by 3GPP. In order to start the development process, one had to get relevant knowledge of 5G NR, especially the CSI feature. Also, different 3GPP specifications shall be explored. For this thesis work, some restrictions had to be made. Different configurations were studied, but the focus of this thesis ended up being on Type I single panel configurations since these are the most commonly used in commercial networks. Even though 3GPP specifications provide support for eight different ranks, this thesis focuses only first four ranks. In addition, 3GPP specifications provide aperiodic, periodic, and semi-persistent reporting configurations. In this work, semi-persistent reporting is ignored and the focus is on aperiodic and periodic reporting. However, extension for semi-persistent support would be a straightforward future step along with eight rank support. Overall, the system was developed successfully to significantly assist software engineers' error processing flow regarding CSI errors.

The first major functionality for the system was verifying CSI report bit lengths according to 3GPP specifications. CSI reports are formed by different report quantities and these should follow 3GPP technical specifications. At first, the main functionality of the system is introduced in the design phase including only bit length calculation functionality. Later, different report quantity mapping orders for different report parts were introduced, on how CSI reports are formed. After the design phase, the implementation of the system is illustrated with the help of functions in pseudocode format. The actual program is built on large-scale software and it is containing lots of irrelevant substances, therefore, the system is not provided in its entirety to make it understandable for the reader. After the implementation phase, the system is tested with three different test cases, and two of these are related to the bit length calculation feature. This testing verifies that the system is functioning as desired. Tests are conducted by modifying the source code of UE, creating an intentional bug, which the system should identify. As mentioned, the system achieved good results in terms of bit length calculation functionality.

The second major purpose of the system was to create functionality for verifying correct transmissions of CSI reports. In other words, verifying that debug log is not including any dropped CSI reports. This thesis introduces two different report drop check functionalities for both aperiodic and periodic reports. After the main functionality is introduced along with bit length functionality, the report drop check functionality is added to the design with two additional traces. The design phase introduces the functionality at a high level, more detailed explanation is introduced in the implementation phase, where functions are provided in pseudocode format. Once the system is implemented, the last chapter introduces testing of the report drop check functionality. Also, this was done by creating an intentional bug in UE software, therefore, the system was able to identify the issue. As mentioned in the evaluation of tests, the system has a significant role in debugging larger debug logs. Finally, tests in terms of report drop check functionality succeeded well.

8. REFERENCES

[1] About 3GPP. URL: `https://www.3gpp.org/about-3gpp`. Accessed 30.6.2022.

[2] Multiplexing and channel coding (3GPP TS 38.212 version 16.2.0 Release 16). URL: `https://www.etsi.org/deliver/etsi_ts/138200_138299/138212/16.02.00_60/ts_138212v160200p.pdf`. Accessed 30.6.2022.

[3] 5G/NR - CSI RS. URL: `https://www.sharetechnote.com/html/5G/5G_CSI_RS.html`. Accessed 29.6.2022.

[4] 5G NR Reference Signals (DMRS, PTRS,SRS and CSI-RS). URL: `https://www.techplayon.com/5g-nr-reference-signals-dmrs-ptrssrs-and-csi-rs/`. Accessed 29.6.2022.

[5] 5G NR Downlink CSI Reporting. URL: `https://www.mathworks.com/help/5g/ug/5g-nr-downlink-csi-reporting.html`. Accessed 4.8.2022.

[6] An Introduction to CSI-RS. URL: `https://info-nrlte.com/2021/05/09/an-introduction-to-csi-rs/`. Accessed 7.7.2022.

[7] Radio Resource Control (RRC); Protocol specification (3GPP TS 38.331 version 16.1.0 Release 16) . URL: `https://www.etsi.org/deliver/etsi_ts/138300_138399/138331/16.01.00_60/ts_138331v160100p.pdf`. Accessed 5.7.2022.

[8] 5G/NR - CSI Report. URL: `https://www.sharetechnote.com/html/5G/5G_CSI_Report.html`. Accessed 5.7.2022.

[9] What is Sub-6 GHz & mmWave in 5G and Why mmWave bands are required? URL: `https://www.techplayon.com/what-is-sub-6-ghz-mmwave-in-5g-and-why-mmwave-bands-are-required/`. Accessed 5.8.2022.

[10] 5G NR Beam Managament SA, NSA | Beam Management in 5G NR. URL: `https://www.rfwireless-world.com/5G/5G-NR-Beam-Management.html`. Accessed 5.8.2022.

[11] 5G/NR - Beam Management. URL: `http://www.sharetechnote.com/html/5G/5G_Phy_BeamManagement.html`. Accessed 5.8.2022.

[12] 5G; Study on New Radio (NR) access technology (3GPP TR 38.912 version 14.0.0 Release 14). URL: `https://www.etsi.org/deliver/etsi_tr/138900_138999/138912/14.00.00_60/tr_138912v140000p.pdf`. Accessed 5.8.2022.

[13] Physical layer procedures for data (3GPP TS 38.214 version 16.2.0 Release 16). URL: `https://www.etsi.org/deliver/etsi_ts/138200_138299/138214/16.02.00_60/ts_138214v160200p.pdf`. Accessed 30.6.2022.

[14] Medium Access Control (MAC) protocol specification (3GPP TS 38.321 version 16.1.0 Release 16). URL: `https://www.etsi.org/deliver/etsi_ts/138300_138399/138321/16.01.00_60/ts_138321v160100p.pdf`. Accessed 5.7.2022.

[15] 3GPP TR 38.802: "Study on New Radio Access Technology Physical Layer Aspects". URL: `https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3066`. Accessed 9.8.2022.

[16] Physical channels and modulation (3GPP TS 38.211 version 16.2.0 Release 16). URL: `https://www.etsi.org/deliver/etsi_ts/138200_138299/138211/16.02.00_60/ts_138211v160200p.pdf`. Accessed 5.7.2022.

[17] Yin H., Guo X., Liu P., Hei X. & Gao Y. (2020), Predicting channel quality indicators for 5g downlink scheduling in a deep learning approach. URL: `https://arxiv.org/abs/2008.01000`.

[18] 5G/NR - CSI RS Codebook . URL: `http://www.sharetechnote.com/html/5G/5G_CSI_RS_Codebook.html`. Accessed 6.7.2022.

[19] Kolomatski A. (2021, January), 5g nr typical beamforming cases. URL: `https://www.linkedin.com/pulse/5g-nr-typical-beamforming-cases-andrew-kolomatski?trk=public_profile_article_view`. Accessed 10.8.2022.

[20] Johnson C. (2019) 5G New Radio in Bullets. Independently Published. URL: `https://books.google.co.uk/books?id=NoRjyAEACAAJ`.

[21] Dreifuerst R.M., Heath R.W. & Yazdan A. (2022), Massive mimo beam management in sub-6 ghz 5g nr. URL: `https://arxiv.org/abs/2204.06064`.

[22] Chou W.H., Pao W.C., Tsai C.C., Yeh T.Y. & Pan J.Y. (2021) An adaptive rank selection method in 3gpp 5g nr systems. In: 2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), pp. 1912–1916.

[23] RI (Rank Indicator, Rank Index). URL: `https://www.sharetechnote.com/html/Handbook_LTE_RI.html`. Accessed 8.7.2022.

[24] Cox C. (2021) Link Adaptation, John Wiley and Sons, Ltd, chap. 10. pp. 191–204. URL: `https://onlinelibrary.wiley.com/doi/abs/10.1002/9781119602682.ch10`.

[25] Lei W., Soong A., Jianghua L., Yong W., Classon B., Xiao W., Mazzarese D., Yang Z. & Saboorian T. (2019) 5G System Design: An End to End Perspective. Springer International Publishing. URL: `https://books.google.co.uk/books?id=fZyuDwAAQBAJ`.

[26] What is Python? Executive Summary. URL: `https://www.python.org/doc/essays/blurb/`. Accessed 3.8.2022.

[27] Carrier Aggregation, NR LTE related tech oriented blog. URL: `https://info-nrlte.com/tag/cross-carrier-scheduling/`. Accessed 13.9.2022.