

АО «СМАРТКОМ»

УТВЕРЖДАЮ

(Генеральный директор)

_____ Ф. И. О.
«_____» _____ 2024 г.

**ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ БАЗОВОЙ СТАНЦИИ 4G LTE/LTE-A,
ОБЕСПЕЧИВАЮЩЕЕ КОНТРОЛЬ, УПРАВЛЕНИЕ И ВЫПОЛНЕНИЕ
БАЗОВОЙ СТАНЦИЕЙ СВОЕГО ФУНКЦИОНАЛЬНОГО
ПРЕДНАЗНАЧЕНИЯ (ПО БС)**

АРХИТЕКТУРА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

ЛИСТ УТВЕРЖДЕНИЯ

СМТН.020101.003 ЛУ

СОГЛАСОВАНО

Подп. и дата	
Инв.№ дубл.	
Взам.инв.№	
Подп. и дата	
Инв.№ подл.	

УТВЕРЖДЕН
СМТН.020101.003 ПО

**ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ БАЗОВОЙ СТАНЦИИ 4G LTE/LTE-A,
ОБЕСПЕЧИВАЮЩЕЕ КОНТРОЛЬ, УПРАВЛЕНИЕ И ВЫПОЛНЕНИЕ
БАЗОВОЙ СТАНЦИЕЙ СВОЕГО ФУНКЦИОНАЛЬНОГО
ПРЕДНАЗНАЧЕНИЯ (ПО БС)**

АРХИТЕКТУРА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

СМТН.020101.003 ПО

На 22 листах

Инь.№ подл.	Подп. и дата	Взам. инв.№	Инь.№ дубл.	Подп. и дата

Оглавление

1 Общие сведения о программе	1
1.1 Предназначение настоящего документа	1
1.2 Предложения для разных категорий читателей	1
1.3 Ссылки	1
1.4 Глоссарий	1
2 Структура сети LTE	3
3 eNB	3
4 MME	4
5 SGW	4
6 PGW	5
7 OMC (NMS)	5
8 Сетевые интерфейсы eNB	5
8.1 LTE-Uu	5
8.2 S1-MME	6
8.3 S1-U	6
8.4 X2-C	6
8.5 X2-U	6
9 eNB Протоколы плоскости управления	6
10 eNB Протоколы плоскости пользовательских данных	7
11 BBU Общее описание	8
12 Структура встроенного ПО платы управления	9
12.1 OAM	10
12.1.1 Структура ПО OAM	10
12.1.2 Компоненты управления конфигурацией	11
12.1.3 Компоненты управления тревогами	11
12.1.4 Компоненты управления производительностью	12
12.1.5 Компоненты обслуживания внешнего API (TR069)	12
12.2 CU-CP	12
12.2.1 Описание функций	13
12.3 CU-UP	15
12.3.1 Описание базовых функций	15
13 Описание процесса BPU	18
13.1 Структура процесса	18
13.2 Механизм обмена сообщениями между потоками	19

1 Общие сведения о программе

1.1 Предназначение настоящего документа

В основном этот документ описывает программную архитектуру системы стека протоколов 4G, реализованную в базовых станциях NGCOM и содержит рекомендации по разработке программного обеспечения в последующих версиях. Содержание может отражать не только состояние ПО в выпущенной реализации, но и анализ проблем с точки зрения долгосрочной эволюции, с расчетом на поддержание согласованности продукта и минимизацию затрат на его разработку.

1.2 Предложения для разных категорий читателей

Системный персонал: рекомендуется ознакомиться со всеми разделами;

Разработчики: рекомендуется ознакомиться с релевантными главами по интересам;

Системный персонал по тестированию и проверке: рекомендуется ознакомиться со всеми главами;

1.3 Ссылки

Наименование документа	№ документа	Версия	Автор
NR_gNB_L1_API_Definition		3.0	Европейский институт по стандартизации в области телекоммуникаций
3GPP TS 36.211		A70	
3GPP TS 36.212		A90	
3GPP TS 36.213		A90	
3GPP TS 36.300		F30	
3GPP TS 36.321		F30	
3GPP TS 36.331		F30	

1.4 Глоссарий

AM	Acknowledged Mode (режим подтверждения)
CU	Control Unit (блок управления)
CU-CP	Control Unit Control Plane (блок управления - плоскость управления)
CU-UP	Control Unit User Plane (блок управления - плоскость пользовательских данных)

DU	Data Unit (блок данных)
GAO	Global Application Object (глобальный объект приложения)
GFO	Global Frame Object (глобальный объект рамки)
LTE	Long Term Evolution (сети связи поколения 4G)
MAC	Medium Access Control Layer
OAM	Operation Administration and Maintenance
ODI	Online Debug Interface
ODS	Online Debug Service
OFP	Open Fast Path
PDCP	Packet Data Convergence Protocol
PDU	Protocol Data Unit
PHY	Physical Layer
RB	Resource Block
RLC	Radio Link Control
ROHC	Robust Header Compression
RRC	Radio Resource Control
RNTI	Radio Network Temporary Identifier
S1	The interface between the E-UTRAN and EPC
SDU	Service Data Unit
SIB	System Information Broadcast
TM	Transparent Mode
UDP	User Datagram Protocol
UM	Unacknowledged Mode
VRB	Virtual Resource Block

2 Структура сети LTE

Следующая диаграмма представляет типичную структуру сети LTE:

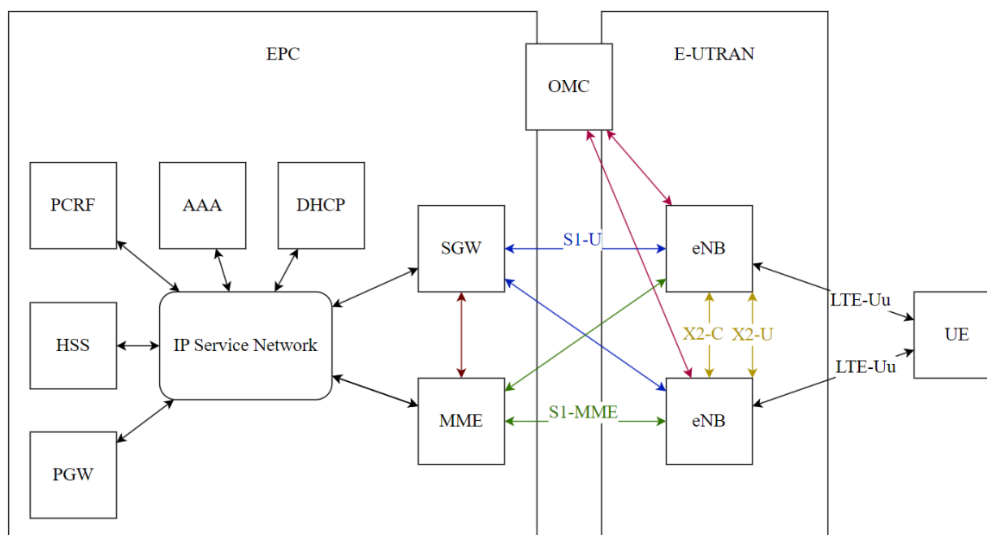


Рисунок 1. Топология сети LTE

3 eNB

Сеть доступа LTE состоит из eNB. Интерфейсом между eNB и UE является LTE Uu, обеспечивающий протоколы control plane и user plane для эфирного интерфейса; взаимодействие между eNB обеспечивает интерфейс X2, включая X2-C для control plane и X2-U для user plane; в то же время eNB взаимодействует с базовой сетью EPC через интерфейс S1, взаимодействует с MME через интерфейс S1-MME плоскости управления и взаимодействует с SGW через интерфейс S1-U плоскости пользователя. eNB выполняет следующие функции:

- Функции для управления радиоресурсами:
 - Управление радиоприемником,
 - Управление доступом к радиосвязи,
 - Управление мобильностью соединения,
 - Динамическое распределение ресурсов между UE как по восходящей, так и по нисходящей линии связи (планирование);
- Сжатие IP-заголовков и шифрование потока пользовательских данных;
- Выбор MME при подключении к UE, когда маршрут к MME не может быть определен на основе информации, предоставляемой UE;
- Маршрутизация данных пользовательского уровня к обслуживающему шлюзу;

- Планирование и передача пейджинговых сообщений (исходящих от ММЕ);
- Планирование и передача широковещательной информации (исходящей от ММЕ или O&M);
- Настройка измерений и отчетов об измерениях для мобильности и планирования;
- Планирование и передача сообщений PWS (включая ETWS и CMA), исходящих от ММЕ;
- Обработка CSG.

4 ММЕ

Каждый UE подключен к ММЕ, который предоставляет услуги и отвечает за взаимодействие с уровнем управления. ММЕ включает в себя следующие функции:

- Сигнализация NAS;
- Безопасность сигнализации NAS;
- Управление безопасностью AS (Access Stratum);
- Передача сигналов между узлами CN для обеспечения мобильности между сетями доступа 3GPP;
- Доступность UE в режиме ожидания (включая управление и выполнение повторной передачи вызова);
- Управление списком зон отслеживания (для UE в режиме ожидания и в активном режиме);
- Выбор PDN GW и обслуживающего GW;
- Выбор ММЕ для передачи данных с изменением ММЕ;
- Выбор SGSN для передачи данных в сети доступа 2G или 3G 3GPP;
- Роуминг;
- Аутентификация;
- Функции управления абонентами, включая создание выделенного абонентского номера;
- Поддержка передачи сообщений PWS (включая ETWS и CMAS);
- Обеспечение оптимизации пересылки сообщений.

5 SGW

Каждый UE подключен к SGW, который предоставляет услуги и отвечает за взаимодействие с пользовательским интерфейсом. SGW обеспечивает следующие функции: Локальная точка привязки мобильности для обеспечения процедуры handover внутри eNB;

- Привязка мобильности для обеспечения процедуры handover в сети 3GPP;

- Буферизация пакетов по нисходящей линии связи в режиме ожидания E-UTRAN и запуск иницируемой сетью процедуры запроса на обслуживание;
- Перехват трафика для COPM;
- Маршрутизация и пересылка пакетов;
- Маркировка пакетов транспортного уровня в восходящей и нисходящей линиях связи;
- Учет детализации данных пользователя и QCI для межоператорской оплаты;
- Учет трафика UL и DL для каждого UE, PDN и QCI в целях дальнейшего биллингования.

6 PGW

Каждый UE подключен к PGW, который предоставляет услуги и включает в себя следующие функции:

- Фильтрация пакетов для каждого пользователя (например, путем глубокой проверки пакетов);
- перехват трафика для COPM;
- Выделение IP-адресов UE;
- Маркировка пакетов транспортного уровня в нисходящей линии связи;
- Учет детализации данных пользователя в целях дальнейшего биллингования за обслуживание на уровне UL и DL, регулирование и контроль тарифов;
- Учет DL трафика пользователей на основе APN-AMBR;

7 OMC (NMS)

Центр эксплуатации и технического обслуживания, ответственный за эксплуатацию и техническое обслуживание различных узлов сети

8 Сетевые интерфейсы eNB

Согласно рисунку 1 между eNB и другими узлами сети LTE используются следующие сетевые интерфейсы: Uu, S1-MME, S1-U, X2-C, X2-U.

8.1 LTE-Uu

Данный интерфейс между UE и eNB поддерживает множество протоколов плоскости управления: RRC, PDCP, RLC, MAC и PHY; также поддерживаются множество протоколов плоскости пользовательских данных: PDCP, RLC, MAC и PHY.

8.2 S1-MME

Данный интерфейс между eNB и MME поддерживает протокол плоскости управления S1AP, который базируется на передаче по SCTP.

8.3 S1-U

Данный интерфейс между eNB и SGW используется для работы протокола плоскости пользовательских данных GTP-U, который базируется на передаче по UDP.

8.4 X2-C

Данный интерфейс между разными eNB поддерживает протокол плоскости управления X2AP, который базируется на передаче по SCTP.

8.5 X2-U

Данный интерфейс между разными eNB используется для работы протокола плоскости пользовательских данных GTP-U, который базируется на передаче по UDP.

9 eNB Протоколы плоскости управления

eNB использует множество разных протоколов плоскости управления (сигнализации) на интерфейсах LTE-Uu, S1-MME, и X2-C, управляющих процедурами enb и Callp. К этим протоколам относятся RRC/ S1AP/ X2AP/ PDCP/ SCTP. Стек протоколов плоскости управления для eNB показан на Рисунке 2:

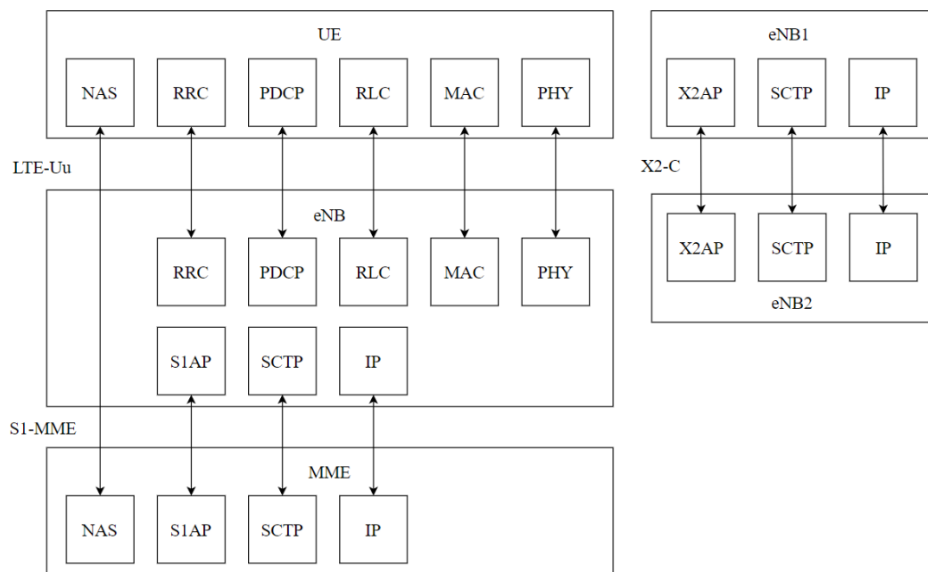


Рисунок 2. Стек протоколов плоскости управления eNB

10 eNB Протоколы плоскости пользовательских данных

eNB использует множество разных протоколов плоскости пользовательских данных на интерфейсах LTE-Uu, S1-U и X2-U, эти протоколы показаны на Рисунке 3:

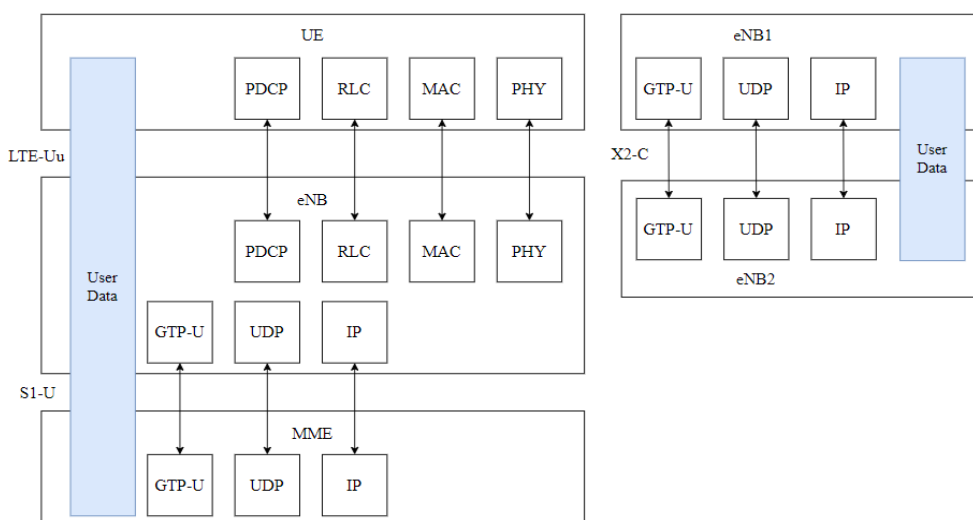


Рисунок 3. eNB Протоколы плоскости пользовательских данных

11 BBU Общее описание

BBU в основном состоит из металлического корпуса, платы базовой частоты, главной платы управления, компонентов объединительной платы (шины), модуля вентилятора и модуля питания. Два слота (два нижних, правые нижние, левые нижние или один нижний, зависит от модели) используются для установки контроллера основной платы управления, а остальные шесть слотов используются для установки платы модуля базовой частоты. Модуль вентилятора и модуль питания могут быть расположены соответственно с обеих сторон, что позволяет гибко настраивать их в зависимости от потребностей. Плата управления может быть продублирована, в этом случае платы работают в режимах активном и резервном, поддерживая оперативный режим горячего резервирования. При полной загрузке можно вставить до шести плат базовой частоты (зависит от модели), а одна плата базовой частоты поддерживает 3-6 сот. При полной загрузке поддерживается до 18-36 сот (зависит от модели).



Рисунок 4. Внешний вид eNB (иллюстрация, конкретная модель может отличаться)

Основная плата управления использует в качестве основного процессора V4860 или

другие (зависит от модели) с 8 ядрами, тактовой частотой 1,8 ГГц и 6 ГБ оперативной памяти. Он включает в себя следующие основные модули - GTP U, PDCP, OAM, eNB, Call и т.д., а также отвечает за задачи управления системой и технического обслуживания. Плата управления базовой полосой также основана на процессоре B4860 или похожем, здесь в основном используются его функции RLC и MAC, и дополнительно оснащена DSP для управления функциями, связанными с физическим уровнем.

На рисунке 5 показано распределение протоколов обработки для каждого процессора BPU4860 в BBU, который включает в себя 8 процессорных ядер PowerPC и 6 DSP-ядер SC3900. Он объединяет большое количество периферийных устройств, таких как Ethernet, SRIO, ускоритель алгоритмов MAPLE и т.д. Для обработки протоколов физического уровня используется DSP. PowerPC используется для обработки протоколов на уровнях MAC и RLC. Медленный поток платы управления в основном обрабатывает протоколы плоскости управления PDCP/RRC/S1AP/X2AP; Быстрый поток в основном обрабатывает пользовательскую плоскость.

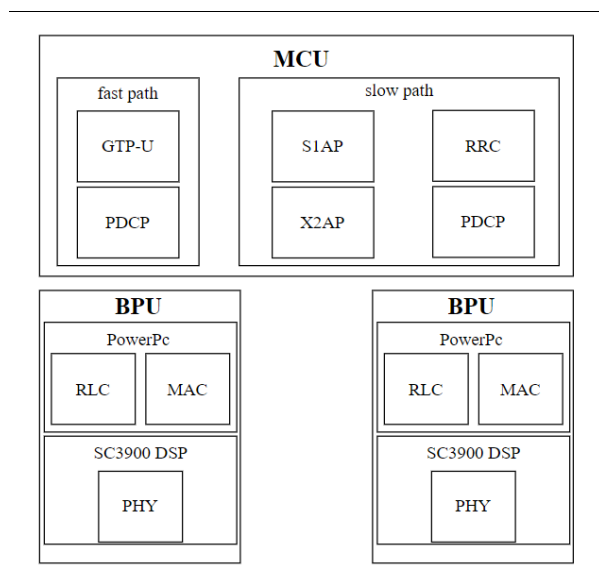


Рисунок 1. Распределение протоколов BBU

12 Структура встроенного ПО платы управления

ПО платы управления в основном состоит из трех частей: модуль быстрого потока (CU-UP), модуль медленного потока (CU-CP) и модуль управления (OAM). Структура ПО платы управления представлена на следующем рисунке.

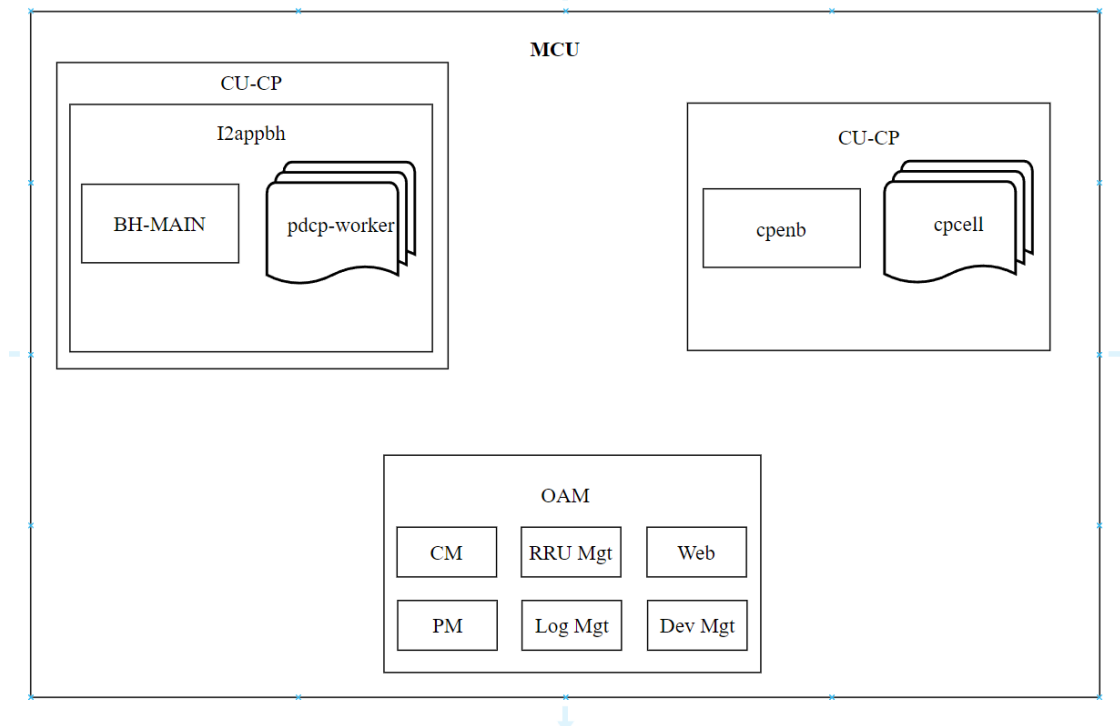


Рисунок 6. Структура ПО платы управления

12.1 OAM

Блок OAM в основном обеспечивает функции управления конфигурацией, управления тревогами (alarm), управления производительностью, управления RRU (включая ПО для RRU), управления устройством, управление стеком протоколов BBU и программным интерфейсом в сторону сетевой системы управления (ОМС или NMS).

12.1.1 Структура ПО OAM

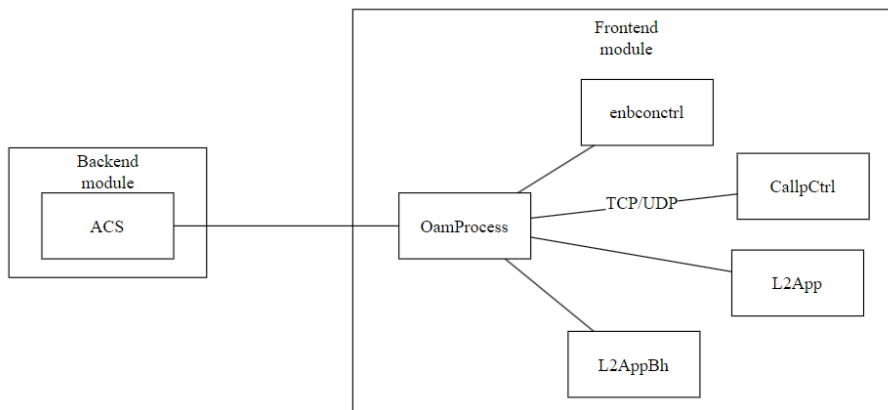


Рисунок 7. Структура модуля ОАМ

Модуль ОАМ в основном обеспечивает управление конфигурацией, генерацию и отключение аварийных сигналов, а также функции сбора и хранения данных о производительности для различных стеков протоколов. Для внешних потребителей, общение с которыми построено на программном интерфейсе TR069 (или аналогичных публичных или проприетарных протоколах в зависимости от модели), он предоставляет такие функции, как добавление, удаление, запрос и изменение параметров, запрос, синхронизация и активное предоставление информации о тревогах, а также файлов отчетов о производительности.

Модуль ОАМ разделен на несколько компонентов: управление конфигурацией, управление сигнализацией, управление производительностью и обслуживание внешнего интерфейса TR069 (или аналогов).

12.1.2 Компоненты управления конфигурацией

Компонент управления конфигурацией отвечает за чтение и запись XML-файлов конфигурации, поддержание согласованности и авторизации данных, а также за сохранность данных и другие связанные с этим задачи. Одновременно предоставляет клиентский прокси-интерфейс для вызова процессов стека протоколов.

Имя модуля	Описание функций
NrConfdMib	Модуль хранения и обработки данных. Обеспечивает назначение/получение и другие операции.
OamCmServer	Модуль обработки взаимодействия со стеком протоколов. Предоставляет услуги подписки, уведомления и другие службы обмена сообщениями для стека протоколов.
NrOamProxy	Клиентский прокси-модуль. Предоставляет соответствующие интерфейсы для вызовов стека протоколов.

12.1.3 Компоненты управления тревогами

Компонент управления аварийными сигналами отвечает за сохранение, генерацию,

отключение и упреждающее представление информации о тревоге системам управления сетью. Одновременно предоставляя клиентский прокси-интерфейс для вызова процессов стека протоколов.

Имя модуля	Описание функций
OamFmServer	Сервер получает запросы от различных прикладных процессов и организует доставку информации об авариях по протоколу TR069 или другому внешнему API.
OamFmClient	Клиентский прокси-модуль. Предоставляет соответствующие интерфейсы для вызовов стека протоколов.

12.1.4 Компоненты управления производительностью

Компонент управления производительностью отвечает за функционирование модели данных о производительности, создание файлов производительности и отправку их в систему сетевого управления. Одновременно предоставляет клиентский прокси-интерфейс для вызова процессов стека протоколов.

Имя модуля	Описание функций
OamPmServer	Выполняет роль сервера, который регулярно извлекает показатели производительности и сохраняет их в специальных файлах
OamPmClient	Клиентский прокси-модуль. Предоставляет соответствующие интерфейсы для вызовов стека протоколов.

12.1.5 Компоненты обслуживания внешнего API (TR069)

Компоненты API отвечают за обслуживание вызовов внешнего API (TR069) и предоставляют интерфейсы для управления конфигурацией, аварийными сигналами и производительностью, в применении к тому или иному варианту API.

Имя модуля	Описание функций
Tr069App	Обслуживание внешнего API (TR069)
Tr069State	Отслеживание состояния внешнего API

12.2 CU-CP

12.2.1 Описание функций

Обеспечивает обработку протоколов RRC, S1AP, X2AP.

Модуль CU-CP состоит из процесса cрnb и процесса cрcell. Процесс cрnb служит внешним интерфейсом, в основном обрабатывая сообщения, не связанные с UE, которые при необходимости могут быть перенаправлены в процесс cрcell. Процесс cрcell в основном обеспечивает обработку бизнес-логики на уровне UE и может быть расширен до нескольких процессов cрcell в зависимости от количества сот.

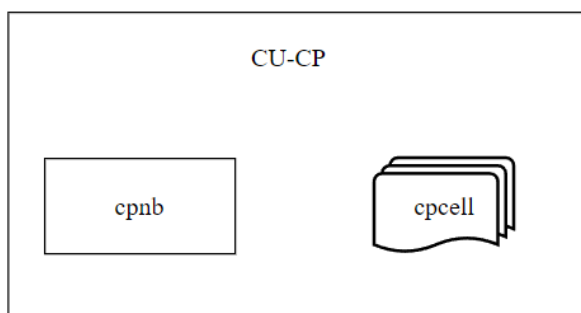


Рисунок 8. Структура модуля CU-CP

12.2.1.1 Описание процесса cрnb

В основном отвечает за обработку основной логики на уровне базовой станции, однопоточный режим обработки

12.2.2 Описание функций submodule cрnb

Имя модуля	Описание функций
X2EndPoint	Взаимодействие с другими eNB, получение сообщений протокола X2 и пересылки этих сообщений в процесс cрcell. Также этот модуль отвечает за отправку сообщений по протоколу X2 соседним eNB процессам cрcellapp.
X2SctpServer	Выполняет роль сервера SCTP с интерфейсом X2.
S1EndPoint	Взаимодействие с MME, получение сообщений протокола S1AP и пересылка этих сообщений в процесс cрcell. Также этот модуль отвечает за

	отправку сообщений в ММЕ от лица процесса <code>сrcell</code> .
<code>ContextIdPool</code>	Управление пулом идентификаторов контекста UE, распределяющее доступный диапазон идентификаторов контекста для <code>сrcells</code> .
<code>TeIdPool</code>	Управление пулом TEID (tunnel endpoint identifier), распределение доступных диапазонов TEID для <code>сrcell</code> .
<code>СrEnbAppToСrCellAppInterf ace</code>	Интерфейс между процессами <code>сrеnb</code> и <code>сrсellаpp</code> .
<code>СrEnbAppToBhControlInterfa ce</code>	Интерфейс между процессами <code>сrеnbаpp</code> и <code>СР-UP</code> .

12.2.2.1 Описание процесса `сrсellаpp`

Этот процесс в основном отвечает за обработку логики на уровне UE в однопоточном режиме обработки. Основными подмодулями `сrсell` являются следующие:

Имя модуля	Описание функций
<code>FlowIdRegistry</code>	Назначение идентификаторов потоков каждому абоненту. Идентификатор потока уникален для всей базовой станции.
<code>СrCellAppToBhControlInterf ace</code>	Интерфейс между процессами <code>Сrсell</code> и <code>CU-UP</code> .
<code>ContextIdPoolProxy</code>	Извлечение диапазона значений UEID из <code>сrеnb</code> , соответствующих <code>сrсell</code> , создание пула идентификаторов и выделение идентификаторов в адрес UE для использования.
<code>TeIdPoolProxy</code>	Получение диапазона значений TEID из <code>сrеnb</code> , соответствующих <code>сrсell</code> , создание пула идентификаторов и выделение идентификаторов в адрес UE для использования.
<code>X2EndPointProxy</code>	Обработка сообщений протокола X2.
<code>S1EndPointProxy</code>	Обработка сообщений протокола S1ap.
<code>СrCellAppToСrEnbAppInter face</code>	Интерфейс в сторону процесса <code>сrеnb</code> .
<code>CellContext</code>	Хранение общей информации о соте и ресурсах, обработка информации о соте, поддержка пула пользователей (UE) и другой информации соты.

NbrCellCtrl	Обработка информации о соседних сотах.
UeRegistry	Хранение информации о выделенных ресурсах типа UeClass.
UePool	Управление пулом ресурсов UeClass.
UeClass	Инкапсулированный класс UE.
UeState	Хранение информации о статусе UE.
RrcState	Хранение информации о статусе RRC.
S1State	Хранение информации о статусе S1.
SecState	Хранение информации о статусе аутентификации AS.
RbState	Хранение информации о статусе RB.
MeasConfCtrl	Контроль измерений состояний.
HoState	Хранение информации о статусе Handover.

12.2.2.2 ASN

Процесс ASN использует инструменты кодирования и декодирования, предоставляемые ASN Lab, которые вызываются в виде статических библиотек. Система устанавливает ассоциации внутренних определений вызовов с определениями вызовов третьей стороны, прописанных в классе-обёртке Asnlab, преобразует его в формат, определенный Asnlab, вызывает соответствующий интерфейс и, наконец, завершает работу по кодированию и декодированию.

12.2.2.3 Межпроцессный интерфейс

Организует коммуникации типа FIFO между процессами crgnb и crcell.

12.3 CU-UP

12.3.1 Описание базовых функций

Модуль CU-UP обеспечивает поддержку протоколов GTPU и PDCP, включая следующую базовую функциональность:

- Передача пользовательских данных
- Обслуживание PDCP SN (Sequence Number)

- Сжатие и распаковка заголовков
- Шифрование и дешифрование данных
- Сегментация и маршрутизация Bearer (пользователей)

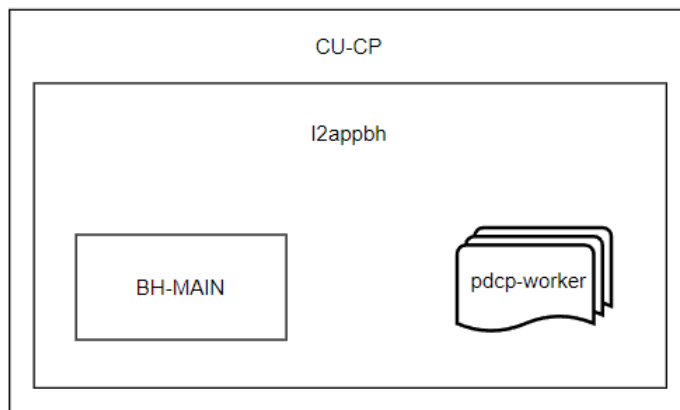


Рисунок 9. Структура функции CU-UP

Процессы L2appBh (PDCP) и l2app (MAC/RLC) могут быть развернуты на одном и том же узле или на разных сетевых узлах, и оптимизация передачи данных может выполняться отдельно на основе характеристик этих двух методов развертывания.

Когда UP и DU развертываются на одном узле, интерфейс PDCP-RLC использует общую память. Когда UP и DU развертываются на разных узлах, пакеты данных на интерфейсе PDCP-RLC должны передаваться по сети.

Для повышения производительности обработки процесс UP поддерживает многопоточную архитектуру, состоящую из одного потока планирования (диспетчера) ввода-вывода (bh main) и нескольких рабочих потоков (pdcп worker). Поток планирования отвечает за получение восходящих и нисходящих пользовательских пакетов, а также сообщений от CU-CP, а затем помещает их в очередь одного из рабочих потоков. Каждый рабочий поток последовательно извлекает пакеты данных из своей собственной рабочей очереди для обработки и отправляет сообщения внешним получателям по мере необходимости после обработки. Дизайн процесса обработки сообщений подразумевает, что l2appBh является системой, работающей по принципу один вход – множество выходов.

Чтобы уменьшить нагрузку на систему, вызванную синхронизацией и блокировкой между рабочими потоками, поток планирования помещает пакеты данных, принадлежащие одному и тому же потоку DRB, в очередь одного и того же рабочего потока. Это позволяет избежать частых блокировок и операций ожидания между рабочими потоками и позволяет

обрабатывать пакеты данных в том порядке, в котором они были получены.

Поток планирования и рабочий поток по-прежнему должны поддерживать блокировку при чтении и записи в одну и ту же очередь. Чтобы свести к минимуму частоту блокировок между потоком планирования и рабочим потоком, рабочий поток создан таким образом, чтобы извлекать сообщения из рабочей очереди пакетами для обработки, вместо того чтобы извлекать только одно сообщение за раз.

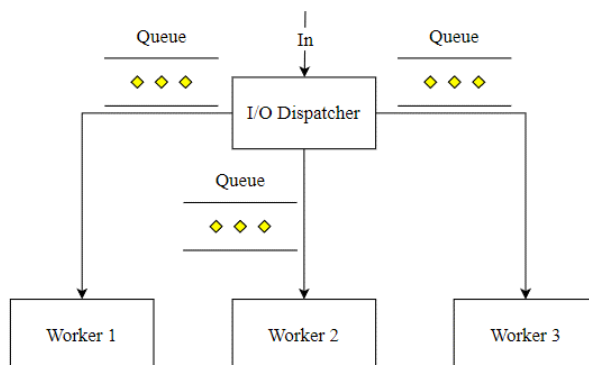


Рисунок 10. Структура функции l2arph

Система поддерживает гибкую настройку количества рабочих потоков. Если система обеспечивает аппаратное ускорение, количество рабочих потоков может быть уменьшено для экономии ресурсов процессора. Если система не поддерживает аппаратное ускорение, необходимо настроить большее количество рабочих потоков в зависимости от ситуации для удовлетворения требований к производительности.

13 Описание процесса ВРУ

Основными функциями ВРУ являются L2app (MAC/RLC) и РНУ. L2app в основном используется для реализации планирования беспроводных ресурсов, включая два подуровня протокола: управление доступом к мультимедиа (MAC) и управление беспроводным каналом (RLC).

13.1 Структура процесса

LTE предъявляет очень высокие требования к вычислительным возможностям ВРУ, включая цикл обработки, пропускную способность пользователей, полосу пропускания и задержку. По мере увеличения числа пользователей и бизнес-процессов сложность планирования значительно возрастет. Поэтому в архитектуре программного обеспечения используется многопоточная система параллельной обработки данных. Каждый поток может быть динамически развернут в соответствии с текущей загрузкой процессора.

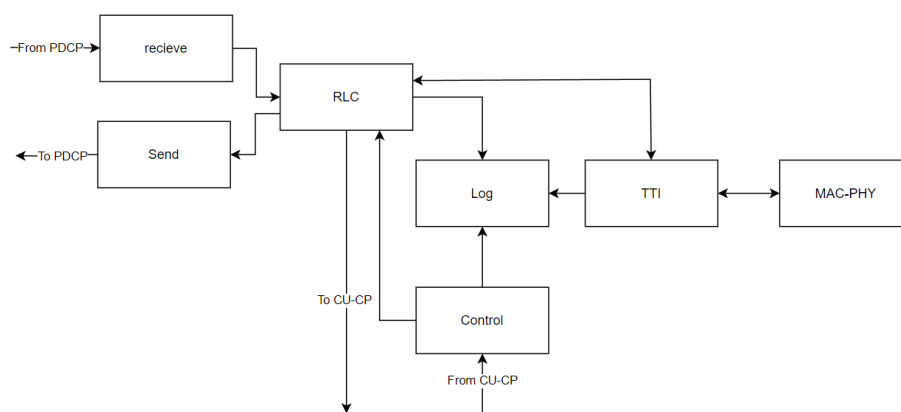


Рисунок 11. Схема распределения потоков L2app

Всю функцию L2app можно рассматривать как управляемое внешними сообщениями, которые воздействуют на работу L2app, генерируя сообщения для взаимодействия между потоками в L2app. С точки зрения обработки внешних сообщений, разделение труда потоков заключается в следующем:

Поток управления отвечает за инициализацию системы, получение сообщений CU-CP и не является блокирующим;

Поток ТТІ обрабатывает сообщения MAC-PHY. Когда в интерфейсе нет сообщений, поток блокируется. Когда физический уровень генерирует сообщение, он запускает поток ТТІ через семафор. Этот поток также отвечает за обработку пакетов нисходящего канала RLC;

Поток RLC отвечает за обработку сообщений PDCP, отправку сообщений плоскости управления, предварительную обработку нисходящего канала RLC и обработку протокола восходящего канала RLC;

Принимающий поток отвечает за обработку сообщений от PDCP;

С точки зрения базовой функциональной разделение труда между каждым потоком выглядит следующим образом:

- Управляющие потоки
 - Запуск и инициализация процессов
 - Прием и пересылка сообщений CU-CP
- Поток RLC
 - Поддержание потока
 - Реализация предварительной обработки нисходящего канала RLC
 - Реализация функции объекта восходящей линии связи RLC
- Поток ТТІ
 - Реализация функциональности уровня протокола MAC
 - Реализация алгоритма планирования QoS
 - Реализация управления пакетами RLC в нисходящем канале
- Принимающий поток
 - Отвечает за считывание пакетов данных из PDCP во внутреннюю циклическую очередь для чтения нижестоящими потоками RLC
- Отправляющий поток
 - Отвечает за отправку данных из потоков RLC в PDCP

Потоки ТТІ необходимы для взаимодействия с физическим уровнем и должны быть способны отвечать в режиме реального времени. Кроме того, чтобы обеспечить минимальную задержку обработки, потоки RLC настроены на отклик в режиме реального времени, когда ресурсов процессора достаточно. Приоритет управляющих потоков относительно низок. Потоки приема и отправки не требуют реакции в режиме реального времени.

13.2 Механизм обмена сообщениями между потоками

Настраиваемые циклические очереди сообщений FIFO используются для обмена данными между внутренними потоками, как показано на рисунке ниже. Каждая очередь сообщений обеспечивает безопасность чтения и записи в потоке с помощью тегов для чтения/записи.

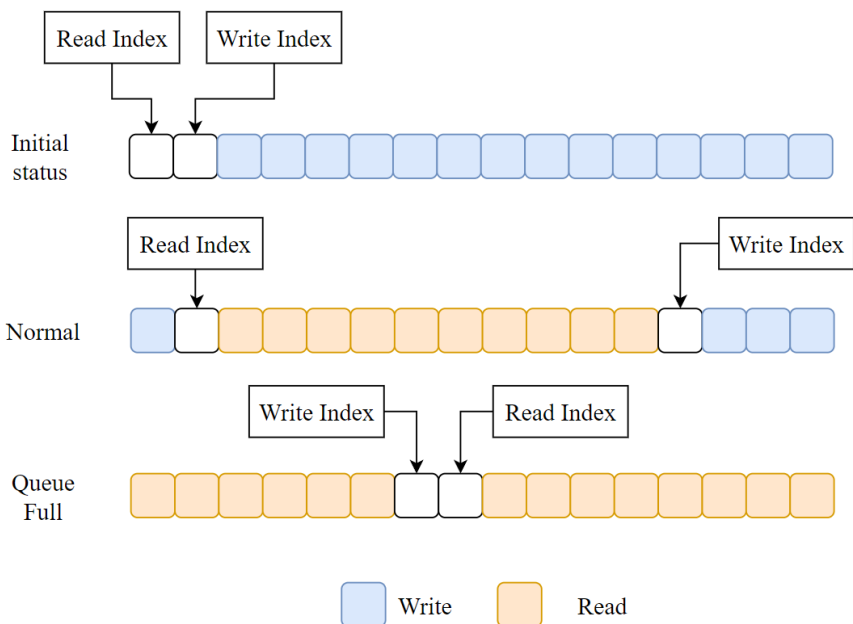


Рисунок 12. Схематическое изображение циклической очереди FIFO

В приведенной выше модели для каждой очереди сообщений одному потоку разрешено выполнять непрерывную запись, а другому потоку разрешено выполнять непрерывное параллельное чтение. Она не поддерживает одновременное чтение или запись в очередь сообщений несколькими потоками. Поэтому для разных комбинаций отправителей и получателей требуются разные очереди. Очереди сообщений, поддерживаемые системой в настоящее время, приведены в таблице ниже.

Таблица 3-1 Внутренняя очередь сообщений

Отправка	Получение
Поток управления	Поток RLC
Поток RLC	Поток TTI
Поток TTI	Поток RLC

Время обработки полученных сообщений варьируется в зависимости от различных потоков, как показано в таблице ниже.

Таблица 3-2. Время обработки внутренних сообщений от DU

Получение	Режим получения
Поток управления	При использовании метода опроса системного тактового цикла точность относительно низкая и не зависит от реального времени.
Поток RLC	Запускается по семафору, обеспечивая высокую производительность в режиме реального времени.
Поток TTI	Запускается сообщениями РНУ в каждом временном интервале с точностью, соответствующей самому интервалу.

Все внутренние сообщения имеют одинаковую структуру, как показано в таблице ниже.

Таблица 3-3. Структура внутренних сообщений DU

Позиция	Тип	Описание
msgClass	1 байт	Не используется, резерв
msgType	1 байт	Тип сообщения определяется получателем по мере необходимости.
msgData	80 байт	Тело сообщения

Для повышения эффективности, объём текста сообщения в структуре сообщений составляет всего 80 байт, что может удовлетворить потребности подавляющего большинства сценариев. Когда требуется передать сообщения размером более 80 байт, они передаются через общую память.