



Описание функциональных характеристик,  
информация для установки и эксплуатации  
программы

## **IVA R**

(Модуль интеграции IVA R для сервера видеоконференций IVA MCU)

# СОДЕРЖАНИЕ

АННОТАЦИЯ.....	3
1. ОБЩИЕ СВЕДЕНИЯ .....	4
1.1. Обозначение и наименование программы .....	4
1.2. Программное обеспечение, необходимое для функционирования программы.....	4
1.3. Языки программирования, на которых написана программа .....	4
2. ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ .....	5
2.1. Классы решаемых задач и назначение программы .....	5
2.2. Функциональные ограничения на применение .....	6
3. ОПИСАНИЕ ЛОГИЧЕСКОЙ СТРУКТУРЫ.....	7
3.1. Алгоритм запуска программы.....	7
3.2. Алгоритм работы модулей программы .....	9
3.2.1. Алгоритм обработки SIP Request.....	9
3.2.2. Алгоритм работы модуля обработки SIP регистраций.....	10
3.2.4. Алгоритм работы модуля API .....	11
3.3. Используемые методы .....	12
3.4. Структура программы с описанием функций составных частей и связи между ними .....	14
3.5. Связи программы с другими программами .....	14
4. ИСПОЛЬЗУЕМЫЕ ТЕХНИЧЕСКИЕ СРЕДСТВА .....	16
5. ВЫЗОВ И ЗАГРУЗКА.....	17
6. ВХОДНЫЕ ДАННЫЕ.....	18
7. ВЫХОДНЫЕ ДАННЫЕ .....	19
8. УСТРАНЕНИЕ НЕИСПРАВНОСТЕЙ, УСОВЕРШЕНСТВОВАНИЕ, ПОДДЕРЖКА.....	20

## АННОТАЦИЯ

В настоящем документе приводится описание программного обеспечения «Модуль интеграции IVA R для сервера видеоконференций IVA MCU», обеспечивающего работу B2BUA сервера, работу сервера SIP регистрации и регистрацию на внешних SIP-серверах платформы IVA MCU.

Настоящий документ содержит общие сведения о программе модуля интеграции, её функциональное назначение, описание логической структуры, сведения о используемых технических средствах, о вызове и загрузке программы, а также входные и выходные данные.

Данное описание программы предназначено для специалиста, имеющего навык работы с программным обеспечением.

## 1. ОБЩИЕ СВЕДЕНИЯ

### 1.1. Обозначение и наименование программы

Программное обеспечение модуля интеграции IVA R для сервера видеоконференций IVA MCU (далее по тексту – Программа) является самостоятельным модулем, структурно входящим в систему IVA-Main платформы IVA MCU.

Программа представляется в виде deb-пакета и устанавливается на операционную систему Debian помощью инструмента apt.

### 1.2. Программное обеспечение, необходимое для функционирования программы

Системные средства, используемые программой, должны быть представлены операционной системой Debian Linux (не ниже версии 9 Stretch) с разрядностью 64bit.

Для функционирования программы необходим следующий набор виртуальных пакетов:

- java8-runtime-headless (стандартный пакет Debian);
- iptables (стандартный пакет Debian);
- fail2ban (стандартный пакет Debian);
- bash (стандартный пакет Debian);
- service-scripts (пакет IVA-системы);
- sudo (стандартный пакет Debian).

### 1.3. Языки программирования, на которых написана программа

Программа написана на объектно-ориентированном языке программирования Java.

## 2. ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ

### 2.1. Классы решаемых задач и назначение программы

Программа предназначена для решения следующего ряда задач:

- Реализация логики р2р звонков с возможностью вызова внешнего сервера для определения направления звонка;
- Скрытие REFER на REINVITE при активном звонке;
- Поддержка распараллеливания звонка при наличии нескольких адресов регистрации у абонента;
- Автоматическая поддержка звонков с разным Expired Timeout при контроле SIP сессий;
- Поддержка аутентификации при исходящем звонке;
- Звонок пользователю, зарегистрированному на данном сервере;
- Поддержка выбора транспорта для исходящего звонка (TCP/UDP/TLS или автоматический выбор);
- Поддержка входящего транспорта SIP – WSS.
- Авторизация по паролю (Digest);
- Авторизация без пароля;
- Регистрация абонентов;
- Возможность регистрации по одному логину с различных устройств;
- Вызов внешнего API для проверки параметров регистрации абонента;
- Вызов внешнего API для информирования о поддержке регистрации абонента.
- При звонке на данные SIP сервера, модуль должен поддерживать авторизацию на этих серверах;
- В случае ошибок регистрации модуль должен в автоматическом режиме пробовать восстанавливать регистрацию;

- Модуль предоставляет API для получения статуса регистрации на серверах;
- Модуль должен в автоматическом режиме обновлять список серверов, где необходимо регистрироваться.
- Блокировки IP по числу ошибок регистрации;
- Блокировки IP по максимальному числу звонков с одного IP (с возможностью настроить список сетей, которые не требуют блокировки)
- Обновление в автоматическом режиме списка внешних IP адресов;
- Работа со всеми доступными сетевыми интерфейсами;
- Поддержка транспортных протоколов: TCP/TLS/UDP/WS/WSS/AUTO.
- Поддержка регистрации на S4B;
- Преобразование вызовов MS-SIP (S4B) в SIP и обратно;
- Автоматическое переключение транспорта в случае недоступности используемого.

Программа также должна:

- 1) Проводить логирование своей работы;
- 2) Работать в среде Linux в режиме сервиса с автоматическим запуском при старте ОС;
- 3) Соответствовать стандартам RFC 3261.
- 4) Поддерживать:
  - не менее 2000 активных звонков;
  - не менее 10 звонков в секунду;
  - не менее 1000 зарегистрированных пользователей.

## 2.2. Функциональные ограничения на применение

Программа не имеет функциональных ограничений.

### 3. ОПИСАНИЕ ЛОГИЧЕСКОЙ СТРУКТУРЫ

#### 3.1. Алгоритм запуска программы

Алгоритм запуска программы (рисунок 1) состоит из следующих этапов:

1) При включении ЭВМ в сеть происходит запуск операционной системы Linux;

2) ОС Linux запускает программу по команде:

*“service voip-signalling-gateway start”*;

3) Загружаются модули программы (описание и функции модулей программы приведены в разделе 3.4);

4) Считываются настройки;

5) Производится автоматический запуск всех модулей.

Последовательность работы и действия, осуществляемые модулями приведены ниже:

а) Работа модуля логирования. Происходит открытие файла. Начинается запись протокола работы;

б) Работа модуля SIP сигнализации. Открытие TCP/UDP портов в соответствии с настройками системы. Переход в ожидание SIP запросов от SIP клиентов;

в) Работа модуля NAT. Загрузка правил NAT из внешней систем;

г) Работа модуля регистрации в сервисе обнаружения сервисов. Регистрация на серверах регистраций приложений/сервисов;

д) Работа модуля API. Открытие портов для приема API переход в ожидание вызовов со стороны сервера управления;

е) Работа модуля обработки SIP регистраций. Запуск;

ж) Работа модуля обработки B2BUA вызовов. Запуск;

з) Работа модуля регистрации на внешних SIP регистраторах. Загрузка с сервера управления списка серверов, где необходимо зарегистрироваться. Регистрация на внешних SIP серверах и поддержка активной регистрации на этих серверах.

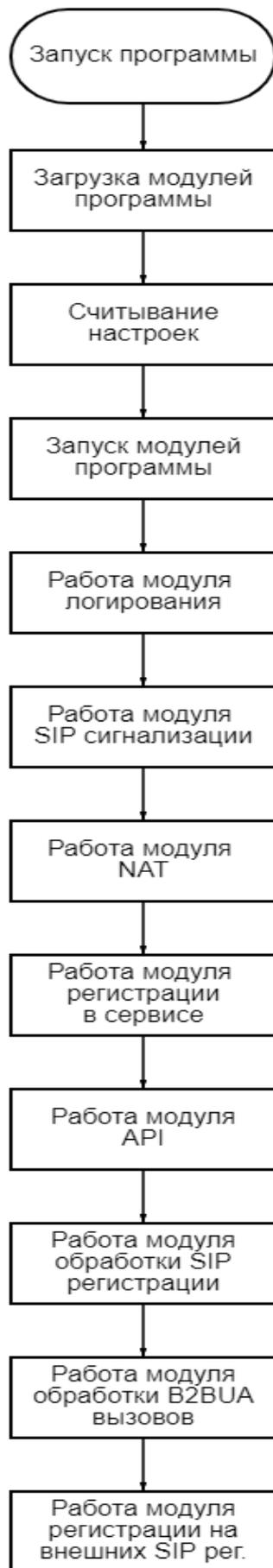


Рисунок 1 – Блок-схема алгоритма запуска программы

## 3.2. Алгоритм работы модулей программы

### 3.2.1. Алгоритм обработки SIP Request

Блок-схема данного алгоритма представлена на рисунке 2.

Описание работы алгоритма обработки SIP запросов приведено ниже:

1) Осуществляется приём SIP запроса;

2) Проверяется факт отношения SIP запроса к B2BUA диалогу:

- Если факт установлен – то он передаётся в модуль обработки B2BUA

вызовов.

- Если нет – вызывается метод REGISTER.

- Если метод успешен – передаётся в модуль обработки SIP регистраций.

- Если нет – применяется метод INVITE и передаётся в модуль обработки

B2BUA вызовов для создания нового B2BUA вызова.

- Если метод не успешен – запрос не обрабатывается.

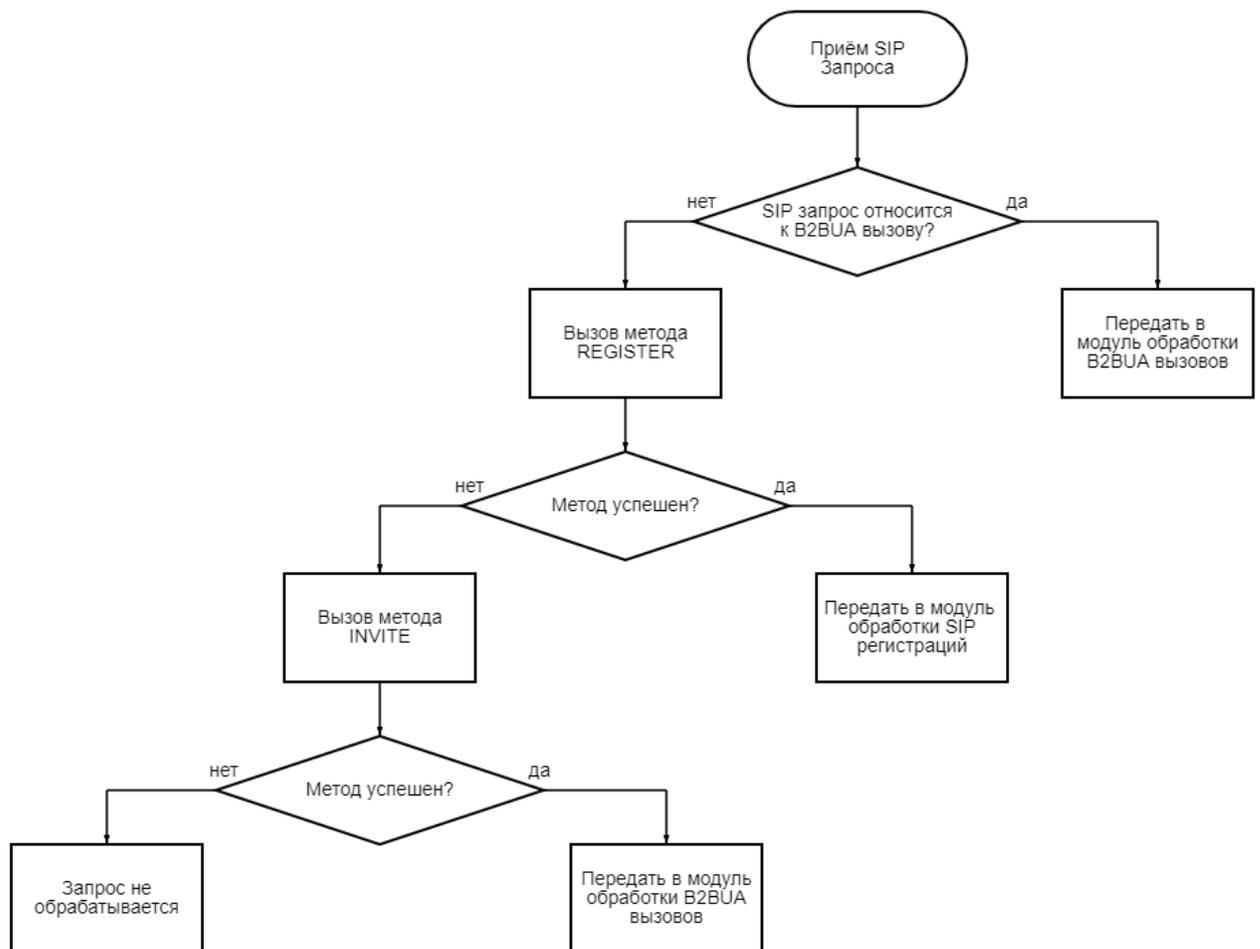


Рисунок 2 – Блок-схема алгоритма обработки SIP запроса

### 3.2.2. Алгоритм работы модуля обработки SIP регистраций

Алгоритм работы данного модуля представлен ниже:

1. Запросить проверку пароля на внешней системе, если корректно, то запомнить абонента в Location Table
2. Проинформировать сервер бизнес-логики о появлении новой SIP сессии.

### 3.2.3. Алгоритм работы модуля обработки V2BUA вызовов

Блок-схема алгоритма при активном вызове представлена на рисунке 3.

Описание работы алгоритм данного модуля при активном вызове приведено ниже:

1. Обработывается активный вызов;
2. Вызов метода REFER. Варианты: leg с которой пришел запрос отключить или создать новый Leg с новыми данными;
3. Вызов метода BYE. Варианты: отправить сообщение другому участнику или удалить V2BUA диалог;
4. Отправить сообщение другому участнику.

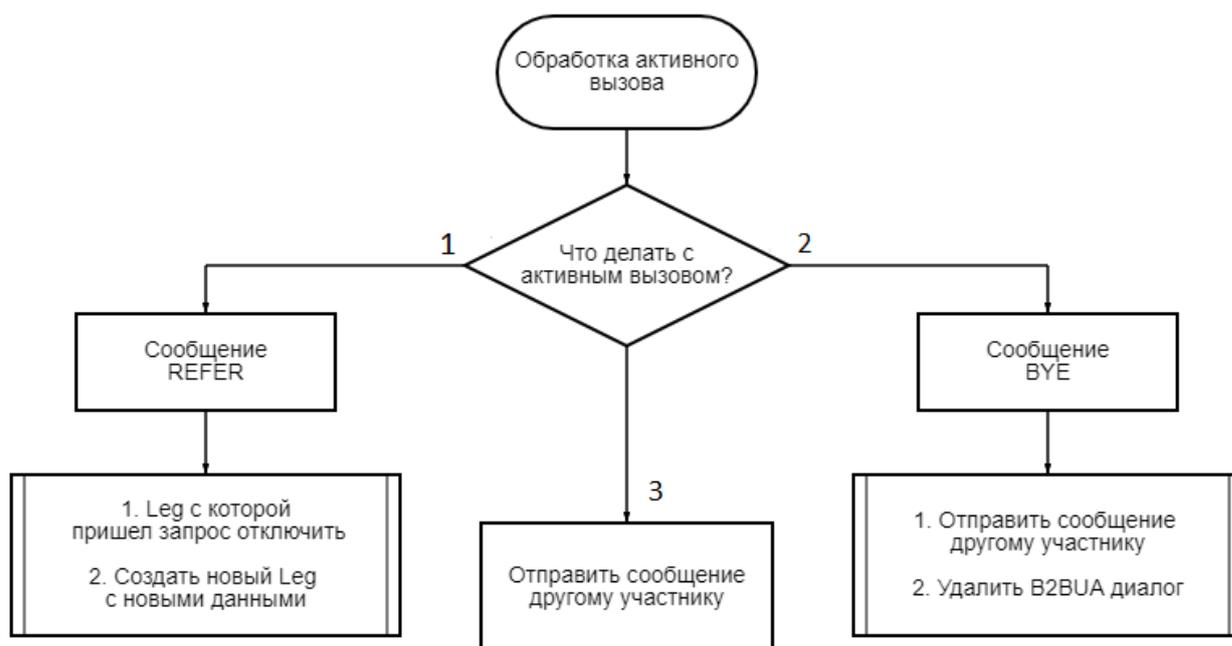


Рисунок 3 – Блок-схема алгоритма обработки V2BUA активного вызова

Блок-схема алгоритма при новом вызове представлена на рисунке 4.

Описание работы алгоритма данного модуля при новом вызове приведено ниже:

1. Обработывается новый вызов INVITE;
2. Вызывается серверный метод для определения что делать со звонком;
3. Предоставляется выбор: перенаправить на другой URI, вызвать зарегистрированного абонента, отбить звонок.



Рисунок 4 – Блок-схема алгоритма обработки V2VUA нового вызова

### 3.2.4. Алгоритм работы модуля API

Алгоритм обработки API методов описан ниже:

- 1) Принять API запрос (вызов);
- 2) Проверить, выполняется ли запрос с разрешённой стороны;
- 3) Выполнить API запрос и вернуть результат.

### 3.3. Используемые методы

Программа в ходе своего функционирования использует шесть основных и семь вспомогательных методов, функции которых представлены в таблицах 1 и 2 соответственно.

Таблица 1 – Основные методы, применяемые программой

Наименование метода	Функция метода
INVITE	Используется для установления мультимедийного сеанса между пользовательскими агентами. Содержит медиаинформацию о вызывающем абоненте в теле сообщения. Успешный запрос INVITE устанавливает диалог между двумя пользовательскими агентами, который продолжается до тех пор, пока не будет отправлен BYE для завершения сеанса
ACK	Используется для подтверждения окончательных ответов на метод INVITE. ACK всегда идет в направлении INVITE. ACK может содержать тело SDP (характеристики среды), если оно не доступно в INVITE
OPTIONS	Используется для запроса информации о функциональных возможностях терминала адресата
BYE	Используется для завершения установленного сеанса. Это SIP-запрос, который может быть отправлен вызывающим или вызываемым абонентом для завершения сеанса
REGISTER	Выполняет регистрацию агента пользователя. Этот запрос отправляется пользовательским агентом на сервер регистратора.
CANCEL	Используется для завершения сеанса, который не установлен. Пользовательские агенты используют этот запрос для отмены ожидающей попытки вызова, инициированной ранее

Таблица 2 – Расширенные методы, применяемые программой

Наименование метода	Функция метода
INFO	Используется пользовательским агентом для отправки информации о сигнализации вызова другому пользовательскому агенту, с которым он установил мультимедийный сеанс
MESSAGE	Используется для отправки мгновенного сообщения с использованием SIP
NOTIFY	Используется пользовательскими агентами для получения определенного события. Обычно NOTIFY запускается в диалоге, когда между подписчиком и уведомителем существует подписка
PRACK	Используется для подтверждения получения надежной передачи предварительного ответа
REFER	Используется пользовательским агентом для обращения к другому пользовательскому агенту для доступа к URI для диалога
SUBSCRIBER	Используется пользовательскими агентами для установления подписки с целью получения уведомления о конкретном событии
UPADATE	Используется для изменения состояния сеанса, если сеанс не установлен. Пользователь может изменить кодек с помощью UPDATE

3.4. Структура программы с описанием функций составных частей и связи между ними

Структура программы представлена в таблице 3.

Таблица 3 – Структура и функции модулей программы

Наименование модуля	Функция модуля
Модуль обработки V2BUA вызовов	Отвечает за реализацию V2BUA обработку звонков
Модуль обработки SIP регистраций	Отвечает за регистрацию SIP абонентов в SIP регистраторе и хранение Location Table
Модуль API	Отвечает за прием и обработку вызовов API
Модуль регистрации на внешних SIP регистраторах	Отвечает за регистрацию на внешних SIP серверах
Модуль регистрации в сервисе обнаружения сервисов	Отвечает за регистрацию на сервере регистраций приложений/сервисов
Модуль SIP сигнализации	Отвечает за реализацию SIP сигнализации с различными транспортом
Модуль логирования	Отвечает за логирование работы
Модуль NAT	Отвечает за работу NAT

3.5. Связи программы с другими программами

Программа взаимодействует с различными компонентами, принимая от них и передавая данные.

Схема связи при регистрации пользователей в системах видеоконференцсвязи:

*SIP User => SIP REGISTER => Программа == (API) => Server*

Схема связи при звонке со стороны клиента:

*SIP User => SIP INVITE => Программа == (API) => Server – проверка доступа*  
*Программа == (SIP) => Media Server*

Схема связи при звонке в сторону клиента:

*Media Server => SIP INVITE => Программа == (API) => Server – проверка*  
*доступа*

*Программа == (SIP) => SIP User*

Схема связи при регистрации ВКС на внешних серверах:

*Программа == (API) => IVCS Server – получение списка серверов, где*  
*необходимо зарегистрироваться => SIP REGISTER => External Registrar Server*

Интеграция с сервисом fail2ban.

Программа предоставляет default правила для работы fail2ban.

По умолчанию настроены следующие правила:

– блокировка SIP трафика с данного IP при ошибке регистрации: 10  
ошибок за 300 секунд – полная блокировка;

– блокировка SIP трафика с данного IP при входящих звонках: 10  
звонков за 300 секунд – полная блокировка.

Блокировка может сниматься через администрирование системы в  
настройке сети.

#### 4. ИСПОЛЬЗУЕМЫЕ ТЕХНИЧЕСКИЕ СРЕДСТВА

Для осуществления работоспособности и надёжного функционирования программы должна использоваться электронно-вычислительная машина с заданными минимальными техническими характеристиками (табл. 4).

Таблица 4 – Минимальные требования к ЭВМ для работы программы

Параметр	Значение
Количество ядер процессора	2 ядра
Объем видеопамяти	2 Гб
Тактовая частота процессора	2 ГГц

Для поддержки графического отображения информации необходим монохромный жидкокристаллический экран с диагональю не менее 18 дюймов.

## 5. ВЫЗОВ И ЗАГРУЗКА

Загрузка программы должна происходить автоматически при запуске операционной системы (см. раздел 3.1).

Вызов и управление программой осуществляется следующими командами из командной строки:

*“service voip-signalling-gateway status” – выводит на экран информацию о состоянии программы;*

*“service voip-signalling-gateway stop” – останавливает работу программы;*

*“service voip-signalling-gateway start” – запускает программу;*

*service voip-signalling-gateway restart – перезапускает программу.*

## **6. ВХОДНЫЕ ДАННЫЕ**

Входными данными программы являются:

- а) Файлы программы с настройками;
- б) SIP-пакеты;
- в) API методы.

## **7. ВЫХОДНЫЕ ДАННЫЕ**

Выходными данными программы являются:

- а) Журнал работы с записями логов;
- б) SIP-пакеты;
- в) API методы.

8. Необходимое гарантийное обслуживание, а также модификация и усовершенствования программы осуществляется разработчиком ООО «ЦТЖД». Техническая поддержка осуществляется разработчиком в рамках гарантийного обслуживания или на условиях соглашения, может быть передана лицу, имеющему необходимые компетенции и персонал.