

“МОДУЛЬ ВАЛИДАТОРА ДЛЯ
ВЗАИМОДЕЙСТВИЯ С ПЛАТФОРМОЙ
"СБЕРТРОЙКА" ("СБЕРТРОЙКА СОЦ",
"СБЕРТРОЙКА ПРО") С ФУНКЦИОНАЛОМ
ОБРАБОТКИ БИЛЕТНОЙ ПРОДУКЦИИ
СИСТЕМЫ”

РУКОВОДСТВО ПО ЭКСПЛУАТАЦИИ

Системные требования

- 2-х ядерный процессор ARM Cortex-A7 с тактовой частотой от 1 ГГц
- 1 Гб оперативной памяти
- 8 Гб eMMC
- Считыватель бесконтактных банковских карт BCAM-01
- Модемный модуль Mini PCI-e с поддержкой LTE и GPS/GNSS (SIM7600E и аналоги)
- Графический цветной дисплей с RGB интерфейсом подключения
- Кнопки GPIO
- Звукоизлучатель
- Linux 5.10 и выше

Установка программного обеспечения

Данное программное обеспечение распространяется в виде deb-пакетов и может быть установлено пакетным менеджером, поддерживающим данный формат (dpkg, opkg и другие).

Для установки - воспользуйтесь соответствующими инструкциями по установке пакетов для вашего пакетного менеджера.

На валидаторах используется пакетный менеджер собственной разработки - grpkg (remote pkg), который производит установку deb-пакетов согласно инструкциям полученным от программного сервера.

Запуск и выключение программного обеспечения

После установки программного обеспечения, оно автоматически будет зарегистрировано как сервис и будет запускаться автоматически при подаче питания на валидатор.

Выключение программного обеспечения так же происходит в автоматическом режиме.

При отключении питания валидатора, происходит переход на питание от батарейки, программное обеспечение отслеживает такое состояние и приступает к сохранению данных, хранящихся в оперативной памяти на диск, после чего, программное обеспечение выключается сервисной службой (systemd).

Описание основных операций валидатора

Разблокировка валидатора

Для разблокировки валидатора, следуйте указаниям на дисплее валидатора - необходимо последовательно поднести карту “выхода” и карту “водителя”. При этом в ПО происходит проверка наличия данного маршрута в маршрутном задании и создается запись о разблокировке в лог Системных Событий.

Оплата проезда при помощи карты «Тройка» и оплата при помощи бесконтактной банковской карты

В программном обеспечении реализованы процедуры обработки карт «Тройка» и бесконтактных банковских карт. Для обработки/оплаты проезда необходимо поднести карту к считывателю, при этом будет осуществлена проверка нахождения идентификатора в стоплистах, проверка остатка на карте «Тройка» и расчет текущего тарифа в соответствии с маршрутом и данными о текущей остановке.

В случае успешного проведения операции будет воспроизведен звуковой сигнал и отобразится зеленый круг с белой галочкой. В случае ошибки отобразится крест на красном фоне и будет воспроизведен звуковой сигнал.

Режим просмотра

Для просмотра информации по карте «Тройка», необходимо перевести валидатор в режим просмотра, нажав на кнопку в верхней части устройства. Выход из режима просмотра будет осуществлен автоматически при отсутствии действий в течении нескольких секунд.

Блокировка валидатора

Для блокировки валидатора, необходимо поднести карту «водителя» к считывателю.

Методика проведения испытаний

Проверка технологических операций

Данный раздел содержит сценарий технологических операций, необходимых для настройки, просмотра и изменения режимов работы стационарного валидатора.

№ п/п	Наименование функции, задачи	Описание действий	Ожидаемый результат
1.	Включение питания валидатора	Включить питание валидатора кнопкой на блоке питания стенда.	Включение валидатора, отображение сообщения «Поднесите карту выхода» на дисплее валидатора.

№ п/п	Наименование функции, задачи	Описание действий	Ожидаемый результат
2.	Открытие смены	Поднести к валидатору технологическую карту выхода, поднести технологическую карту водителя	Последовательное отображение сообщений: «Поднесите карту выхода», «Поднесите карту водителя», «Поднесите карту».
3.	Закрытие смены	Поднести к валидатору технологическую карту водителя.	Блокировка терминала, Отображение на экране сообщения «Поднесите карту выхода».
4.	Просмотр информации по карте	Нажать на кнопку, расположенную на корпусе валидатора. Поднести карту к валидатору	Отображение доступной информации по карте «Тройка».
5.	Просмотр технологической информации	При открытом рейсе нажать на кнопку просмотра карт. Поднести технологическую карту администратора и еще раз нажать на кнопку.	Отображение на экране технологической информации: ID системы, ID терминала, версии ПО и т.п.
6.	Просмотр текущей информации по открытой смене	При открытой смене нажать на кнопку просмотра карт. Поднести технологическую карту водителя и еще раз нажать на кнопку.	Отображение на экране валидатора данных о количестве зарегистрированных поездок по типам оплаты.
7.	Проверка автоматического обновления ПО валидатора	Загрузить в АРМ Перевозчика архив с микропрограммой валидатора. Подождать пока произойдет обмен данными. Осуществить просмотр технологической информации.	Отображение на экране технологической информации: версии ПО, совпадающей с версией, загруженной в АРМ Перевозчика микропрограммой.

№ п/п	Наименование функции, задачи	Описание действий	Ожидаемый результат
8.	Проверка автоматического обновления НСИ	Добавление нового тестового маршрута. Обмен данными. Открытие смены на добавленный маршрут.	Успешное открытие смены терминала на добавленный маршрут.

Проверка обработки транспортных и бесконтактных банковских карт

№ п/п	Наименование функции, задачи	Описание действий	Ожидаемый результат
1.	Предъявление карты «Тройка» с недостаточным ресурсом	Приложить карту к валидатору.	Ошибка обработки. Отображение сообщения "Недостаточно средств" и информации об остатке ресурса на дисплее устройства.
2.	Предъявление карты «Тройка» с ресурсом, достаточным для оплаты проезда	Приложить носители с к валидатору.	Успешная регистрация поездки отображение информации об остатке ресурса на дисплее устройства.
3.	Проверка ресурса карты «Тройка» после успешной оплаты поездки	Осуществить просмотр информации по карте	Корректный остаток ресурса
4.	Обработка ББК	Приложить тестовые банковские карты к валидатору	Успешная регистрация поездки

Проверка обработки карт, находящихся в стоплисте

№ п/п	Наименование функции, задачи	Описание действий	Ожидаемый результат
1.	Предъявление ББК, находящейся в транспортном стоплисте	Приложить ББК к валидатору	Ошибка обработки. Отображение сообщения "Стоплист" на экране устройства.

Взаимодействие с системой управления валидаторами

Взаимодействие с системой управления валидаторами требуется для передачи системных событий и обновления баз данных.

Обновления баз данных происходят по расписанию, настраиваемому в конфигурационном файле, системные события передаются системе управления валидаторами непосредственно при возникновении события.

Демонстрация взаимодействия

- Подключится к валидатору по SSH и открыть лог-файл системы управления валидаторами:
- `tail -f /var/log/afcps.log`
- Провести операцию по банковской и по карте тройка

В логе системы управления валидаторами отобразится следующая запись о транзакции по ББК:

```
2025-01-31 16:03:09 D/afcps: >>> 'Off-Line/Transaction/Upload':  
Header: map[Content-Length:[3659] Content-Type:[application/x-  
www-form-urlencoded]]
```

```
2025-01-31 16:03:09
```

```
D/afcps:>>>Body:'TransactionList={"values":[[{"TT", "0001", 0,  
"20250131160309", "3343472495", "-1", "-1", "-1", "",  
"",1000003, "1002","Точиев Ахмед Асхабович", 5, "", null,  
"", "", null, "", 0, 1, 1, false, false, true, "", "", "Номер  
1", "", "F", "R", null, 283534, "33000-283534-3343472495-2",  
"1738-3393-8980-02", "20250131160309", 2, -1, -1, 1738339389,
```

```
15, 17, "Автовокзал №2", "Автовокзал №2", 0, 0, 1, 2, 2, 2,
null, 0, 3, "1-20-10-18-319", 3, null, null,
"2200241014580030082", null, {"Amount":"200", "AuthCode":"00",
"CardAid":"A0000006581010", "CardType":"MIR",
"ERN":"C749576F_3343472495_0000000097", "MaskedPAN":"*****",
"Mode":0, "OpTE":1738339389000, "OpTS":1738339389000,
"OpTags":[...], "PANHash":"*****", "PrSt":1,
"RNN":"202501311603068002", "RespMsg":"Одобрено",
"TID":"3343472495", "Type":0}, {"DiagData":{"AVer":"3.15",
"Mod":"usk04_xx", "TZone":"+0300"},
"ExtensionData":{"Num":8002, "ServerTime":0},
"TransactionReceiptAdditionalData":{},
"rideSegmentInfoNSI":["10", 6, 6, "15", "17", "2025-01-31
16:03:09", "2025-01-31 16:03:09"], "rideSegmentInfoSDBP":["10",
6, 6, "15", "17", "2025-01-31 16:03:09", "2025-01-31
16:03:09"]}, "*****"], "header":["transactionType",
"transactionVersion", "transactionStatus", "transactionTime",
"terminalSerialNumber", "terminalFNSerialNumber",
"terminalFNRegistrationNumber", "terminalKKTSerialNumber",
"lat", "lon", "userID", "userPersonnelNumber",
"userPersonnelFIO", "userPersonnelRoleId", "operatorCardUID",
"operatorCardPAN", "terminalOwnerName", "terminalOwnerINN",
"terminalOrganizationId", "terminalODFINN", "transportType",
"routeMode", "routeType", "isCycled",
"isBookingWithoutTicketData", "isStraight", "civilNumber",
"busGosRegNumber", "routeName", "routeNumber",
"routeTariffingType", "tariffType", "rideId", "rideSegmentId",
"ticketSeries", "ticketNumber", "rideTimeDate",
"workDayNumber", "fiscalDocSignature", "fiscalDocNum",
"fiscalDocDate", "enterStationId", "exitStationId",
"enterStationName", "exitStationName", "operationId",
"ticketId", "ticketsCount", "ticketsSum", "finalTicketPrice",
"baseTicketPrice", "usedServiceRuleId", "sumForDiscount",
"sellTariffType", "sellTariffId", "paymentType",
"sellTransportCardCategory", "sellTransportCardUID",
"sellTransportCardPAN", "personalPassengerData",
"transportCardTransactionData", "transactionAdditionalData",
"hash"]}]'
```

2025-01-31 16:03:10 D/afcps: <<< 'Off-Line/Transaction/Upload':
93 bytes (HTTP code 200 '200 OK')

2025-01-31 16:03:10 D/afcps: <<< Body:
'{"data":{"LoadResult":"OK","TransactionListHash":"4d6eae67102e
245b468ccec492a06564f98c683"}}'2025-01-31 16:03:10 I/afcps:
SBERTR-Transaction #202501311603068002 has been sent

Запись по карте «Тройка»:

2025-01-31 16:07:49 I/afcps: Sending 1 transactions...

2025-01-31 16:07:49 D/afcps: Mfrc: UID:"04922a52206f82"
number:"3932526524" type:83 troika:{app_code:279
ticket_type:433 ticket_number:3932526524
date_time_trip_beg:3202087 the_rest:2400
dump:"E\xdb\x1e\xa6Y{\xce*\xd1\x14\x82\xe9\x07\xe0\x08\x00a\xb8
N\x00\x00\x02X\x00\x00\x00\x00.k\xc8B\x1fa\xb8N\x00\x00\x02X\x0
0\x00\x00\x00.k\xc8B\x1f"}

2025-01-31 16:07:49 D/afcps: URI:
`https://am.06.tkp2.sbertroika.tech/Off-
Line/Transaction/Upload?Hash=928fecdf21894253b57e8f49818a3d0c01
2027d6&AgentId=1000003&TerminalId=xxxxxxx`

2025-01-31 16:07:49 D/afcps: >>> 'Off-Line/Transaction/Upload':
Header: map[Content-Length:[2418] Content-Type:[application/x-
www-form-urlencoded]]

2025-01-31 16:07:49 D/afcps: >>> Body:
'TransactionList={"values":[[{"TT", "0001", 0, "20250131160726",
"4288134720", "-1", "-1", "-1", "", "", 1000003, "1002",
"Точиев Ахмед Асхабович", 5, "", null, "", "", null, "", 0, 1,
1, false, false, true, "", "О 574 CB 06", "Номер 1", "", "F",
"R", null, 283534, "33000-283534-4288134720-78", "1738-3396-
4619-0", "20250131160726", 78, -1, -1, 1738339646, 15, 17,
"Автовокзал №2", "Автовокзал №2", 0, 0, 1, 2, 2, 2, null, 0, 3,
"1-20-10-18-319", 1, null, "04922a52206f82", "3932526524",
null, {"PrSt":1, "amount":2, "cardBalance":"00000960",
"cardData":"...", "cardName":"Тройка - Кошелёк",
"cardNumber":"3932526524", "cardSubType":"279",
"cardType":"ТРОЙКА", "cardUID":"04922a52206f82",
"carrierId":"", "region":""}, {"DiagData":{"AVer":"3.15",
"Mod":"usk04_xx", "TZone":"+0300"}, {"ExtensionData":{"Num":190,
"ServerTime":0}, {"TransactionReceiptAdditionalData":{}},
"rideSegmentInfoNSI":["10", 6, 6, "15", "17", "2025-01-31
16:07:26", "2025-01-31 16:07:26"], "rideSegmentInfoSDBP":["10",
6, 6, "15", "17", "2025-01-31 16:07:26", "2025-01-31
16:07:26"]}], "*****"]], "header":["transactionType",
"transactionVersion", "transactionStatus", "transactionTime",
"terminalSerialNumber", "terminalFNSerialNumber",
"terminalFNRegistrationNumber", "terminalKKTSerialNumber",
"lat", "lon", "userID", "userPersonnelNumber",
"userPersonnelFIO", "userPersonnelRoleId", "operatorCardUID",
"operatorCardPAN", "terminalOwnerName", "terminalOwnerINN",
"terminalOrganizationId", "terminalODFINN", "transportType",
"routeMode", "routeType", "isCycled",
"isBookingWithoutTicketData", "isStraight", "civilNumber",


```
"busGosRegNumber", "routeName", "routeNumber",  
"routeTariffingType", "tariffType", "rideId", "rideSegmentId",  
"ticketSeries", "ticketNumber", "rideTimeDate",  
"workDayNumber", "fiscalDocSignature", "fiscalDocNum",  
"fiscalDocDate", "enterStationId", "exitStationId",  
"enterStationName", "exitStationName", "operationId",  
"ticketId", "ticketsCount", "ticketsSum", "finalTicketPrice",  
"baseTicketPrice", "usedServiceRuleId", "sumForDiscount",  
"sellTariffType", "sellTariffId", "paymentType",  
"sellTransportCardCategory", "sellTransportCardUID",  
"sellTransportCardPAN", "personalPassengerData",  
"transportCardTransactionData", "transactionAdditionalData",  
"hash"]}]'
```

2025-01-31 16:07:50 D/afcps: <<< 'Off-Line/Transaction/Upload':
93 bytes (HTTP code 200 '200 OK')

2025-01-31 16:07:50 D/afcps: <<< Body:
'{"data":{"LoadResult":"OK","TransactionListHash":"2a0d2b6a655c
09452d5dbffd9889d84c33250e83"}}'

2025-01-31 16:07:50 I/afcps: SBERTR-Transaction
#197001010300000190 has been sent

Здесь видно, что были сгенерированы транзакции и произведена отправка транзакций на сервер.