



Зеленый робот

ПРОГРАММНАЯ ДОКУМЕНТАЦИЯ

Архитектура и взаимодействие модулей

Модульное интегрированное ПО «Зеленый робот»

Проектная редакция · GR-FP-1.0

Содержание

1. Назначение архитектуры
2. Логическая структура
3. Технологический путь команды
4. Путь данных и событий
5. Размещение приложений
6. Программное обслуживание из L3
7. Зависимости первоначального запуска
8. Архитектура виртуальных испытаний
9. Доверие и полномочия
10. Устойчивость и восстановление
11. Временные и ресурсные параметры
12. Интеграция и изменение состава

Термины и сокращения

Сведения о редакции

Архитектура и взаимодействие модулей

ДОКУМЕНТ D02 · АСУП.466459.100-01 13 02	РЕДАКЦИЯ 2.0 · Проектная редакция
ПРИМЕНИМОСТЬ Функциональный профиль GR-FP-1.0	ПРОДУКТ Модульное интегрированное ПО «Зеленый робот»

Обозначения и сокращения раскрыты в словаре этой книги.

1. Назначение архитектуры

Архитектура GR-FP-1.0 распределяет технологическое исполнение, транспорт, пользовательские действия и программное обслуживание между модулями L0–L3. Документ предназначен для проектировщика программного состава и инженера интеграции. Он определяет владельцев состояния, направления обмена, зависимости запуска, размещение и восстановление.

Единицей поставки служит версионизируемый программный артефакт. Единицей настройки служит профиль с редакцией и ссылками на артефакты. Единицей подтверждения действия служит квитанция или результат, связанный с идентификатором операции. Конкретная комбинация оформляется паспортом выпуска.

2. Логическая структура

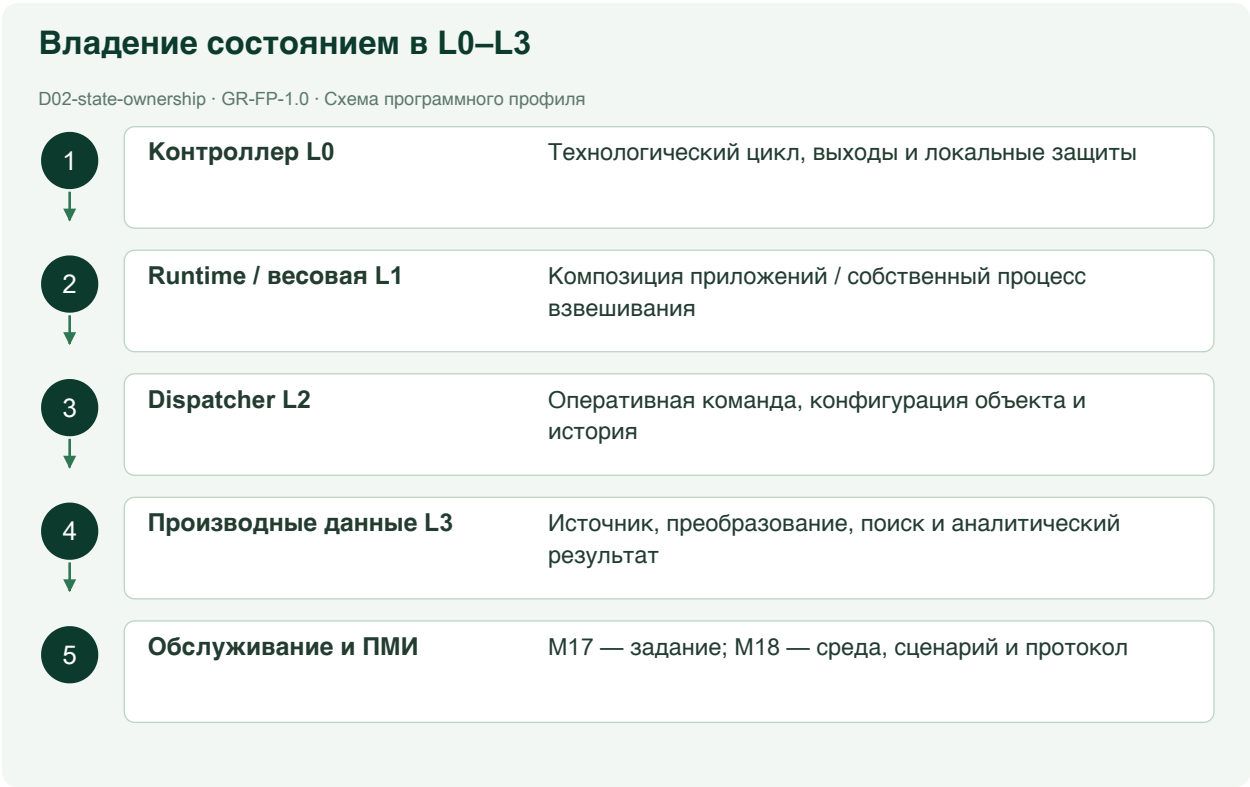
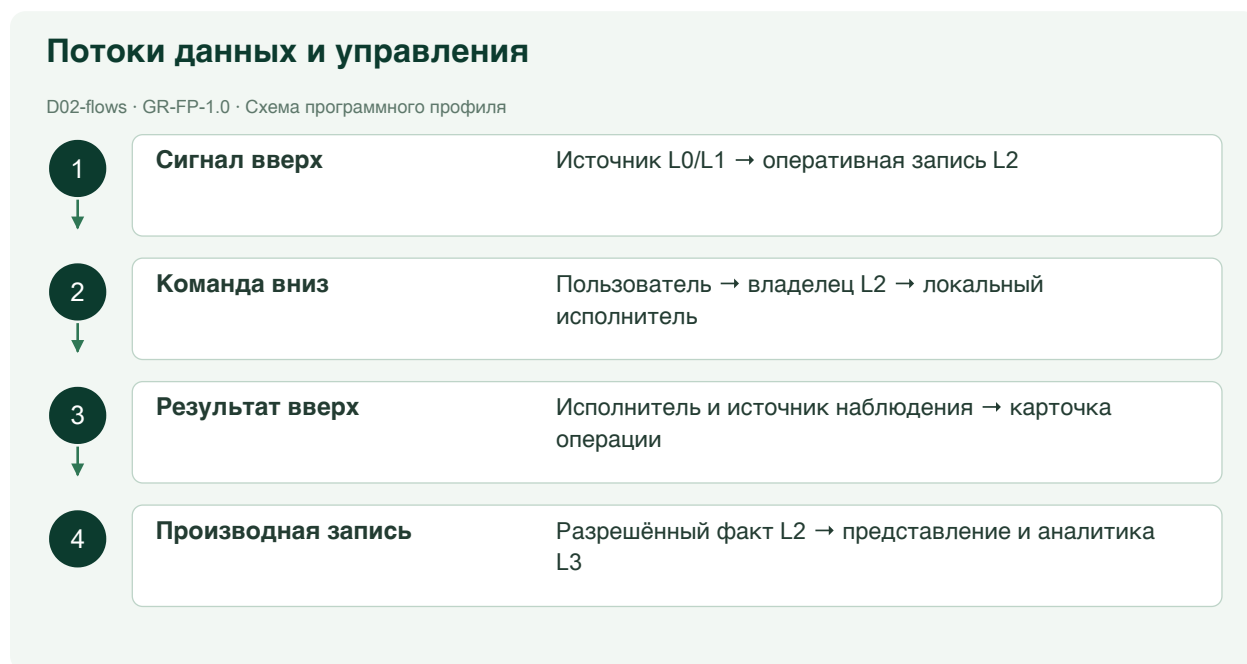


Рисунок D02.1 — Модули исполнения, оперативных данных, производных представлений и обслуживания. Положение модуля показывает его ответственность.

Область	Владелец состояния	Потребители
Технологический цикл и локальная защита	M01 в среде M02	M03/M04, локальная панель и диспетчеризация
Локальная композиция и состояние Runtime	M03	Приложения, HMI, M16/M17
Процесс взвешивания	M19 L1 controller	Киоск, локальная service surface и L2 Integration Pack
Оперативные команды и данные площадки	M11	Операторские представления, M13 и производные данные
Идентичность и сессия	M12	Защищённые интерфейсы модулей
Доставка уведомления	M13	Получатель и интерфейс состояния доставки
Производное представление	M15	Поиск, отчёты и M14
Задание обслуживания	M17	Графический интерфейс L3, M16 и адаптеры компонентов
Прогон испытаний	M18	Представление ПМИ, отчёт и архив доказательств

Владелец принимает изменение, проверяет версию и сохраняет результат. Потребитель обновляет своё представление по снимку и последующим событиям. При восстановлении связи потребитель продолжает чтение с устойчивого курсора либо получает новый согласованный снимок.

3. Технологический путь команды



D02-flows. Потоки данных и управления — последовательность и наблюдаемый результат.

- 1 Пользователь выбирает объект и действие в отраслевом приложении.
- 2 Хост передаёт запрос с идентификатором, контекстом объекта и ожидаемой редакцией состояния.
- 3 Владелец команды проверяет полномочия, срок, состояние и условия действия.
- 4 Транспорт передаёт запрос адресату и возвращает транспортную квитанцию.
- 5 Локальная логика проверяет технологические разрешения и выполняет допустимое действие.
- 6 Исполнитель возвращает результат; наблюдаемые сигналы подтверждают достигнутое состояние.
- 7 Dispatcher связывает этапы одной операции, сохраняет историю и обновляет представление.

Транспортная квитанция подтверждает свой этап доставки. Исполнение и наблюдаемое состояние подтверждаются их источниками. Тайм-аут каждого этапа имеет собственный код и действие восстановления. Повтор запроса с тем же идентификатором возвращает сохранённый результат либо состояние обработки по правилам интерфейса.

3.1. Локальное исполнение

На Linux L1 общие CAN/MQTT/configuration/journal службы предоставляет Runtime. Приложение содержит предметную логику и использует API хоста. На коммуникационном ESP32 соответствующий профиль реализует обмен и обслуживание устройства. Для локальной панели профиль связывает экранное действие с допустимой командой и readback.

M01 завершает решение о физическом воздействии в соответствии с локальными защитами. Профиль фиксирует поведение при потере внешней связи и перезапуске. Наладчик проверяет это поведение методами D16 перед передачей объекта.

3.2. Весовая

M19 использует локальный Go controller как владельца процесса взвешивания. Киоск получает снимок состояния и последовательность событий; запрос действия содержит ожидаемую последовательность expected_seq. Локальное обслуживание обращается к той же весовой. L2 получает данные площадки через профильный адаптер и Integration Pack.

Масса поступает от драйвера весов либо M20. На границе адаптера проверяются единица, время, стабильность и качество. Преобразование тонн в килограммы выполняется в определённой точке профиля и отражается в схеме данных. Запись взвешивания получает устойчивый идентификатор для хранения и повторной передачи.

4. Путь данных и событий

Источник присваивает данным идентичность, время, качество и редакцию схемы. Прикладной владелец принимает их, проверяет порядок и сохраняет факт. Производные представления указывают исходный факт и версию преобразования. Читатель видит время источника и возраст данных, а также состояние канала.

Свойство	Правило
Идентичность	Устройство, узел, объект и площадка имеют отдельные устойчивые идентификаторы
Время	UTC в машинном обмене; пользовательское представление применяет выбранный часовой пояс
Качество	Значение сопровождается статусом и причиной изменения качества
Порядок	Курсор или последовательность назначается владельцем потока
Повтор	Обработка учитывает идентификатор исходного события и правила идемпотентности
Происхождение	Производный результат сохраняет ссылку на исходные записи

Свойство	Правило
Версия	Схема, конфигурация, пакет и выполняемая программа имеют собственные поля

4.1. Журнал событий

Источник хранит историю согласно политике. Экран работает с ограниченным окном: бюджет строк соответствует примерно двум–трём высотам области. При движении по истории страница подгружается с нужной стороны, отображаемые строки за пределами окна освобождаются. Устойчивый ID и смещение видимой строки сохраняют положение читателя.

Ручной просмотр истории приостанавливает автопрокрутку. Кнопка **К текущим событиям** возвращает актуальный фрагмент и возобновляет следование потоку. Размер кеша и live-буфера конечен и входит в профиль UI. Событие и время имеют спокойное обычное начертание; уровень передаётся согласованным маркером и подписью. Проверка охватывает длительную историю, быстрый поток, изменение размера, задержки и восстановление связи.

5. Размещение приложений

Приложение поставляется отдельным неизменяемым артефактом с манифестом. Манифест задаёт ID, версию, владельца, digest frontend, маршруты, API-префиксы, регистрацию, health и совместимость. Runtime собирает композицию и активирует согласованный набор приложений.

Регистрация `ioot.hmi-app-registration/2.0` описывает страницы, типизированные настройки, события, уведомления, действия и контракты данных. У приложения ровно одна начальная страница. Действие связывает метод, endpoint, требуемый доступ и подтверждение. Хост передаёт контекст объекта и управляет общей навигацией.

Один frontend применяется в локальном HMI и во всплывающем окне объекта L2 с подходящим host adapter. Профиль изоляции ограничивает доступ приложения его контекстом и заявленными возможностями. Совместимость проверяется до изменения активной композиции.

6. Программное обслуживание из L3



D02-deployment. Первоначальный запуск по зависимостям — последовательность и наблюдаемый результат.

M17 формирует план изменения желаемого программного состава. План включает исходную редакцию узла, пакеты, зависимости, операции, требуемые права, проверки ресурсов и восстановление. Пользователь подтверждает проверенный план; исполнитель получает ограниченные типизированные шаги.

Этап	Ответственный	Сохраняемый результат
Выбор профиля и пакетов	Интерфейс L3 / M17	Желаемый состав и параметры
Предварительная проверка	M17 и адаптеры компонентов	Ошибки полей, совместимость, ресурсы и план
Авторизация применения	M12 и владелец операции	Решение, субъект, область и срок
Подготовка артефактов	M16 или профильный исполнитель	Digest, staging и исходное состояние
Применение	Исполнитель компонента	Выполненный шаг и новая редакция
Подтверждение	Health/readback адаптер	Фактический состав и качество
Завершение или восстановление	M17	Итог задания и связь с предыдущим составом

Интерфейс L3 отображает задание и вызывает основной API M17. Производная карточка оборудования в M15 содержит ссылки и наблюдаемое состояние. M16 исполняет разрешённые операции узла с TTL и ограничением ресурсов. Программный план учитывает отдельные данные пакета, выполняемую версию ESP32, выполняемую версию AT32 и аппаратную идентификацию цели.

При прерывании клиент повторно открывает сохранённый jobId. Координатор сопоставляет завершённые шаги с фактическим readback и определяет продолжение. Откат выполняется по подготовленной процедуре компонента; миграции данных имеют собственное условие обратимости и резервную копию.

7. Зависимости первоначального запуска

Базовая установка L3 включает интерфейс входа, начальную идентичность, координатор обслуживания и каталог доступных пакетов. Наладчик подготавливает доверие и права, затем создаёт площадку и выбранный профиль. Дальнейший порядок учитывает зависимости пакетов:

- 1 Идентичности и доверие — доступ пользователя и сервисов.
- 2 Хранилища и служебные каталоги — устойчивое состояние компонентов.
- 3 Транспорт площадки — брокер, listener, ACL и ограничения.
- 4 Узлы L1 — исполнитель обслуживания, Runtime и связь.
- 5 Приложения — манифесты, композиция, параметры объекта.
- 6 Встраиваемые цели — подтверждение идентичности, программного состава и канала обслуживания.
- 7 Диспетчеризация и производные представления — объекты, история и результаты.
- 8 Испытания — применимые методы, протокол и передача пользователю.

В автономном профиле каталог, пакеты, зависимости и доверенные материалы доступны из локального носителя или локального сервиса. Установщик проверяет полный граф зависимостей до запуска изменений. Адреса, порты и объёмы хранилищ задаются параметрами профиля.

8. Архитектура виртуальных испытаний

Профиль	Исполняемый предмет	Подменяемая среда
E01	Поставляемый Runtime и приложения в подходящей ОС/архитектуре	CAN/MQTT endpoints, сигналы и время сценария

Профиль	Исполняемый предмет	Подменяемая среда
E02	Образ конкретной ESP32-цели либо выделенный logic artifact	Поддерживаемая модель CPU, flash, NVS, TWAI/USB и сети
E03	Образ конкретной AT32-цели либо переносимая PLC/CANopen логика	Память, загрузка, HAL, CAN, таймеры и сигналы
E04	Модель технологического процесса	Воздействия, динамика, датчики и отказы
E05	Координатор и оценщик результатов	Расписание сценария, контрольный набор, наблюдение и сброс

Manifest испытательного профиля указывает метод исполнения каждого участника. Для бинарной эмуляции фиксируются точная цель, ABI, карта памяти и реализованная периферия. Для переносимой логики фиксируются исходная ревизия, сборочные параметры и интерфейс HAL. Эти сведения задают применимость выводов протокола.

Модель E04 получает воздействия и возвращает сигналы с заданной динамикой. Сценарий задаёт начальное состояние, последовательность событий, seed и модель времени. E05 запускает участников, ожидает готовность, выполняет шаги, собирает результаты и восстанавливает исходное состояние. Повтор использует те же артефакты и входы.

9. Доверие и полномочия

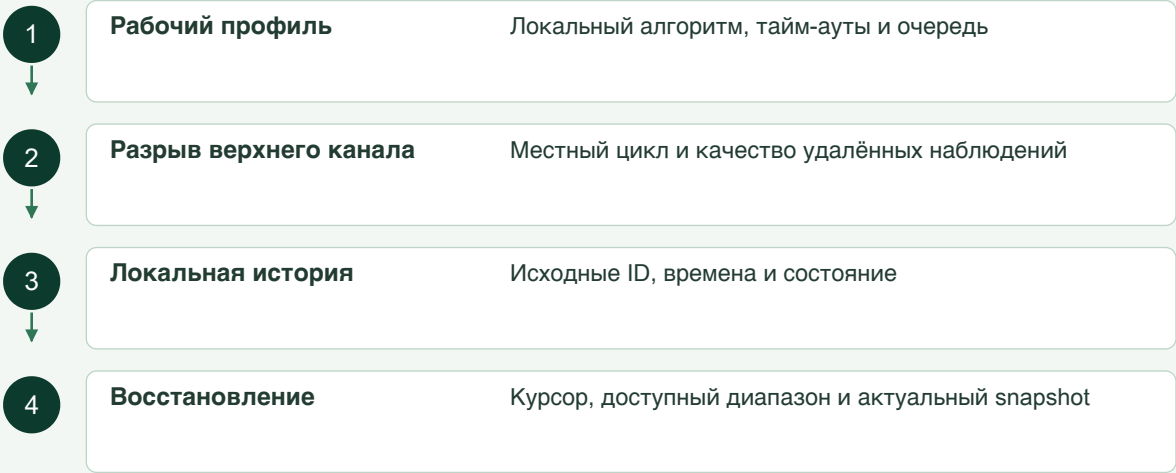
Пользовательская сессия, сервисная идентичность и идентичность устройства имеют отдельные жизненные циклы. Защищённый MQTT-профиль использует взаимную проверку сертификатов и области доступа площадки. Интерфейс обслуживания использует полномочия на конкретное действие и целевой узел. Секрет передаётся как защищённый материал через установленный канал; журналы и экспорт содержат его идентификатор.

Профили ограничивают размер сообщения, очередь, число параллельных операций, время ожидания и длительность сессии. Ротация доверия подготавливает новый путь, проверяет его и затем завершает переход. D07 задаёт последовательность действий администратора.

10. Устойчивость и восстановление

Автономная работа и возврат связи

D02-autonomy · GR-FP-1.0 · Схема программного профиля



D02-autonomy. Автономная работа и возврат связи — последовательность и наблюдаемый результат.

Компонент хранит необходимое ему устойчивое состояние отдельно от исполняемого артефакта. Резервная копия связывает данные с редакцией схемы. Применение пакета использует staging и подтверждение результата; предыдущий состав сохраняется на срок политики восстановления.

Отказ	Архитектурная реакция	Проверка
Потеря L2/L3	Локальный профиль продолжает свой согласованный цикл	Разрыв канала и сверка локального состояния
Повтор сообщения	Владелец сопоставляет ID и возвращает определённый результат	Повторная доставка одинакового запроса
Перезапуск сервиса	Восстанавливаются устойчивые записи и курсоры	Restart между принятием и завершением операции
Смена версии	Проверяются схемы и зависимости, выполняется миграция	Прямой переход и предусмотренное восстановление
Потеря узла во время обновления	Задание сохраняет шаг и ожидает readback	Возврат узла в той же операции
Сбой производного представления	Представление восстанавливается из доступных источников	Перестроение индекса с проверкой происхождения

11. Временные и ресурсные параметры

Паспорт профиля задаёт период PLC-цикла, тайм-ауты heartbeat/команд, порог устаревания данных, лимиты очередей, объём истории, частоту обновления UI и ресурсы процессов. Значение указывается с единицей и основанием. Измерение проводится на заявленной архитектуре с согласованной нагрузкой.

Результат функционального испытания связывается с моделью времени. Проверка производительности фиксирует реальные часы, нагрузку, ресурсы и распределение задержек. Для обмена определяются точки начала и конца измерения, например «принятие команды — readback результата». D16 устанавливает форму протокола.

12. Интеграция и изменение состава

Интегратор начинает с идентификаторов, схем и владельцев состояния из этой книги. Затем выбирает интерфейс D08, сверяет параметры D03, готовит профиль D04 и выполняет ввод D05. Изменение схемы сохраняет явную версию, миграцию и совместимость потребителей. Новая версия приложения проходит проверку регистрации и композиции до установки на объект.

Термины и сокращения

Обозначение	Значение
API	Программный интерфейс взаимодействия компонентов.
HMI	Операторский человеко-машинный интерфейс.
ПМИ	Программа и методика испытаний с заданными шагами, критериями и формой результата.
PLC	Программируемая логика контроллера и её прикладной цикл.
CANopen	Профиль обмена по CAN со словарём объектов, состояниями и сервисами.
MQTT	Протокол обмена сообщениями через брокер; в данном профиле используется версия 5.
digest	Контрольная сумма данных или артефакта по указанному алгоритму.
readback	Обратное чтение фактического состояния или выполняемой версии у её источника.
health	Проверка состояния процесса или компонента.
endpoint	Конечная точка обмена или присоединяемый участник, определённый контекстом интерфейса.

Обозначение	Значение
TTL	Срок действия сообщения, команды или временного разрешения.
ACL	Правила доступа к объектам, данным и операциям.
ABI	Двоичный интерфейс совместимости исполняемого кода.
HAL	Интерфейс программной абстракции аппаратных функций.
seed	Начальное значение генератора случайной последовательности для повторения сценария.
staging	Подготовленная область пакета или конфигурации перед активацией.
manifest	Манифест: описание состава, версий, контрольных сумм и зависимостей.
frontend	Клиентская часть приложения, формирующая пользовательский интерфейс.
Runtime	Среда исполнения и общие программные службы приложения.
NVS	Энергонезависимое хранилище конфигурации в профиле ESP32.

Сведения о редакции

Редакция	Применимость и содержание
2.0 · 13.09.2026	Проектная редакция. Профиль GR-FP-1.0. Архитектура и взаимодействие модулей: функции, параметры, процедуры и проверяемые результаты.