



ЭКСПОНЕНТА

ЦЕНТР ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ
И МОДЕЛИРОВАНИЯ

РИТМ SDR USRP

Российская универсальная платформа
программно-определяемого радио для
прототипирования, разработки и
тестирования сложных радиосистем

Отечественная замена
NI USRP

Сокращаем путь от идеи
к рабочему радиостенду



**СДЕЛАНО
В РОССИИ**

**МИНПРОМТОРГ
РОССИИ**



Что такое программно-определяемое радио — и почему это важно



Раньше: аппаратное радио

- ✗ Каждый стандарт связи — отдельная микросхема
- ✗ Смена стандарта = новое железо = новые затраты
- ✗ Разработка нового устройства — годы
- ✗ Тестирование алгоритма требует физического прибора
- ✗ Нет гибкости: запаяли — переделать нельзя

Теперь: программно-определяемое радио

- ✓ Алгоритмы — это код, а не схема
- ✓ Смена стандарта = новая прошивка, то же железо
- ✓ Прототип за недели, а не за годы
- ✓ Тест в модели → сразу на реальный эфир
- ✓ Один прибор — любые задачи: связь, РЛС, навигация

Почему мы этим занялись

Годами стандартом SDR-разработки был NI USRP. После его ухода российским инженерам понадобилась совместимая альтернатива. **PIPM SDR USRP создан именно для этой задачи.**



Для чего применяется SDR?



Приём/передача
радиосигналов



Цифровая обработка
сигналов



Спектральный анализ
и мониторинг эфира



Прототипирование
систем связи



Эмуляция и
исследование
протоколов связи
(Wi-Fi, LTE/5G, GNSS, LoRa,
BT, TETRA, DVB-S2/T2/S2X
...)



Запись
и воспроизведение
I/Q квадратур сигналов



Удалённое управление
и автоматизация
измерений



Связь, радиолокация,
радионавигация

Современный этап развития радиосвязи



Вызовы отрасли



Многообразие стандартов

LTE/5G, Wi-Fi, GNSS, LoRa, TETRA, DVB
и исследования в области 6G



Увеличение полосы частот

Современные и перспективные стандарты
требуют расширения мгновенной полосы



Быстрый выход на рынок

Ускорение перехода от идеи к прототипу,
минимизация затрат



Обратная совместимость

Наработки в GNU Radio / LabVIEW /
MATLAB, Python

Наш ответ: ПИТМ SDR USRP



Единая SDR-платформа для разных классов
радиосистем



Сверхширокая полоса частот для
приёма и передачи



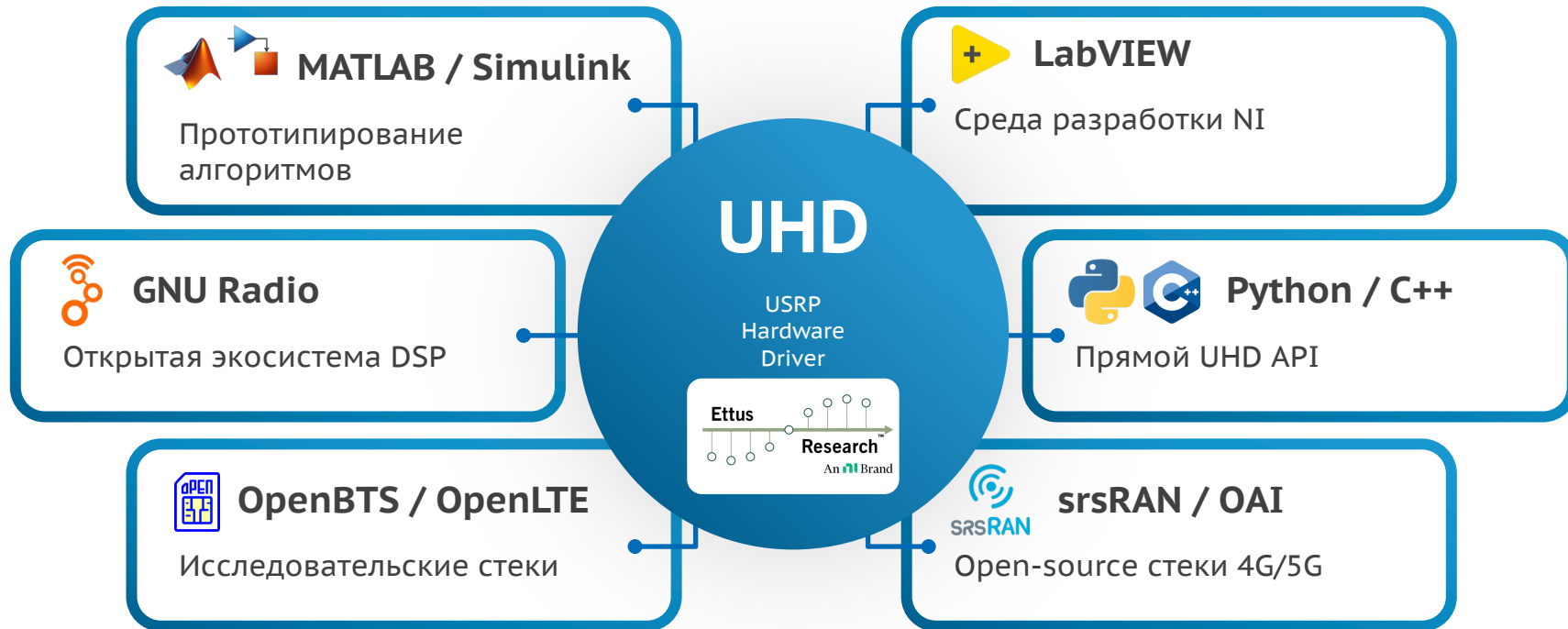
Встроенные
высокопроизводительные
IP-ядра для ЦОС



Полная совместимость с UHD
(USRP Hardware Driver)



UHD – один язык для всей экосистемы SDR



UHD стал отраслевым стандартом благодаря единому API для управления SDR-оборудованием

NI USRP недоступен 🙅. Что дальше?



С 2022 года поставки NI USRP в Россию прекращены.
Команды, которые работали на E320, X310, X410, оказались перед выбором.

Китайские клоны, OEM SDR платформы

ANTSDR, Puzhi и др.

- ✗ Неполная совместимость с UHD или ее отсутствие
- ✗ Требуется глубокое погружение в отладку стыка ПО и железа
- ✗ Неполная проверка режимов и диапазонов

РИТМ SDR USRP

🇷🇺 Российская платформа

- ✓ UHD-совместимость
- ✓ Встроенные IP-ядра
- ✓ Локальная поддержка
- ✓ MIMO 8×8



РИТМ SDR USRP — единственная российская платформа с полной UHD-совместимостью, встроенными IP-ядрами и локальной инженерной поддержкой

Техническая реализация



Ядро платформы – связка ПЛИС Zynq UltraScale+ и радиочастотного модуля ADRV9009

75 МГц – 6 ГГц

Диапазон частот

2 TX + 2 RX + 2 ORX

Каналы

Полная поддержка UHD

Совместимость с NI

TX 450 / RX 200 МГц

Полоса на канал

2× SFP+ 10Gb, 2× 1GbE

Каналы

Свободные ARM-ядра

Пользовательские алгоритмы

MIMO 2×2 → 8×8

Масштабирование

2× USB 3.0, CAN, NVMe

Интерфейсы

Связь, РЛС, навигация

Применение

**Набор встроенных IP-ядер:
DDC / DUC / AGC / DPD / FFT /
OFDM / LDPC и др.**



**Гибкая архитектура
и масштабирование
до систем MIMO 8×8**

Гибкая архитектура SDR-платформы

РИТМ SDR USRP

Вычислительные ресурсы

Трансивер

01 ARM-ядра

Интеграция алгоритмов
пользователя

02 ПЛИС

Высокопроизводительные
IP-ядра ЦИТМ «Экспонента»

03 РЧ-ПЛАТА

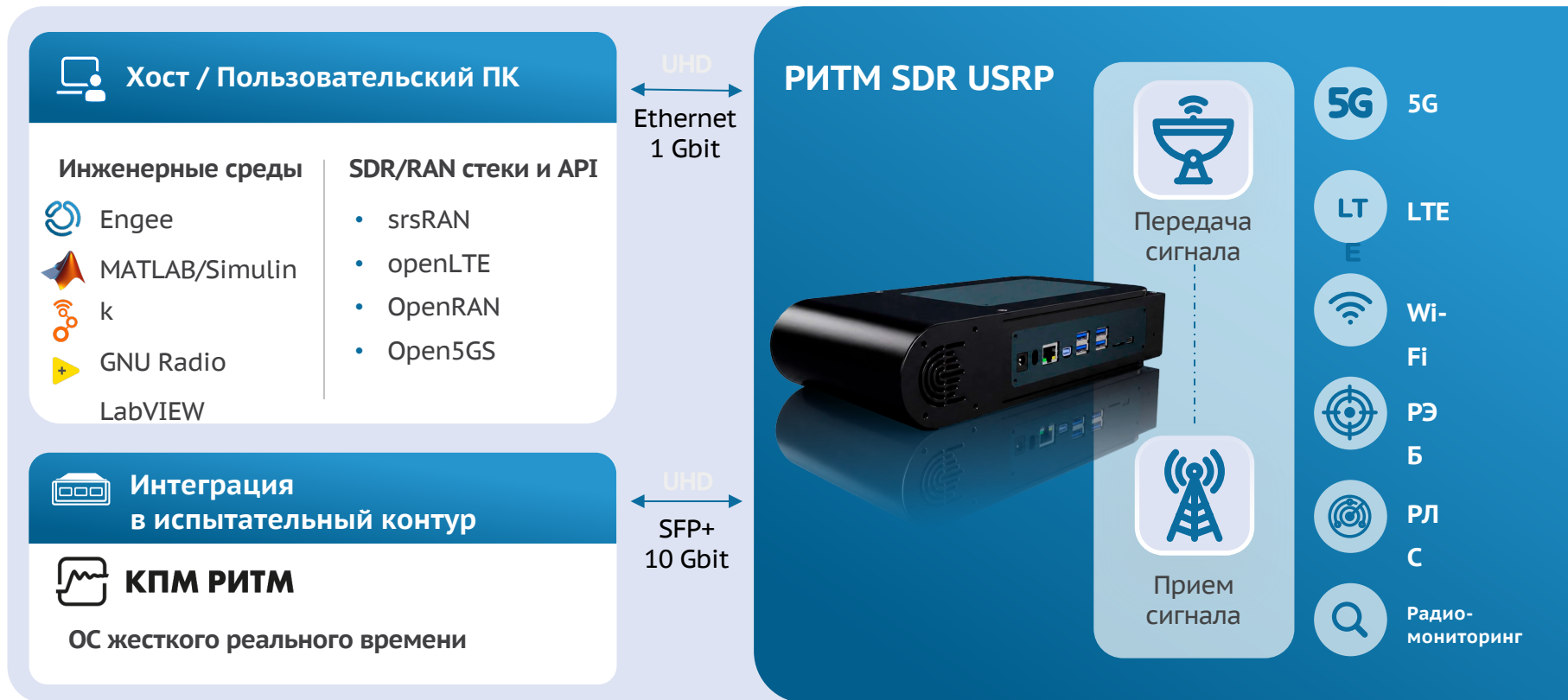
Цифра ↔ Аналог

Конфигурация
и управление

Выбор и переключение
IP-ядер «на лету»

Пользовательские алгоритмы адаптации,
управления и расчета параметров

Экосистема ПО и интеграция



Сравнение с NI USRP: E320, X310, X410



Совместимость с E320, X310 и X410 за счет поддержки UHD-интерфейса

NI USRP E320

- 70 МГц – 6 ГГц
- до 56 МГц
полосы
- 2×2 MIMO



NI USRP E320

- 10 МГц – 6 ГГц
(для типовой UBX)
- До 160 МГц
(для дочерней платы UBX)
- 2 слота под
дочерние платы



NI USRP X410

- 1 МГц – 7,2 ГГц
- до 400 МГц на
канал
- 4×4 MIMO



ПИТМ SDR USRP

 Российская платформа

- 75 МГц – 6 ГГц
- TX до 450 МГц
- RX до 200 МГц
- MIMO 2×2 → 8×8
- Локальная
инженерная
поддержка



Известные сценарии
работы и UHD-
интерфейс


Широкая
полоса
TX / RX

Единый форм-фактор
и интеграция в
существующие стенды

Локальная
прошивка и
инженерная
поддержка

РITM SDR USRP vs типовые китайские SDR



	Клоны бюджетных USRP (серия B2xx)	OEM SDR-платформы	РITM SDR USRP Российская платформа
Диапазон частот	70 МГц – 6 ГГц	75 МГц – 6 ГГц	75 МГц – 6 ГГц
Мгновенная полоса	до 56 МГц	TX до 450 МГц / RX до 200 МГц	TX до 450 МГц / RX до 200 МГц
Масштабирование	✗	✗	2x2 → 8x8
UHD-совместимость	✓	✗	✓
Высокоскоростной SFP	✗	✗	✓
В/п IP-ядра	✗	✗	✓
Локальная поддержка в РФ	✗	✗	✓
Риск для проекта	Подходит для старта, но много зависит от пользователя	Хорошая база, но нужна самостоятельная интеграция	Ниже риск: платформа + интерфейсы + сопровождение
Пример	ANTSDR E200	Puzhi P159	

Встроенные высокопроизводительные IP-ядра

Программно-конфигурируемый цифровой тракт предобработки (Digital Front-End, DFE)

Задача



IP-ядро



Практическая ценность: меньше ручной разработки, быстрее путь к рабочему стенду

Базовый набор IP-ядер

Базовые IP-ядра, готовые к использованию

- Цифровое АРУ (AGC)
- Цифровой DDC 1...128
- Цифровой DUC 1...128

Системы связи

- FFT/IFFT
- OFDM мод/демод: LTE, 5G NR, Wi-Fi, DVB-T2
- LDPC для DVB-S2/S2X

Линеаризация

Работа с внешним усилителем мощности (PA)

- Цифровые предискажения (DPD)
- Снижение пик-фактора (PC-CFR)

Синхронизация и декодирование

- Декодер Витерби
- Эквалайзер LTE DL (LTE-EQ)
- Кадровая/частотная синхронизация LTE (LTECS)

Сценарии применения PИТМ SDR USRP



01 Open-source стеки связи и базовые станции



LTE/5G

Wi-Fi

UE

Базовые станции

02 РЛС, радионавигация и спецрадиосистемы



Обработка
а
сигналов

РЛС

Радионавигация
Спецрадиосистемы

03 Тестирование радиотехнических устройств



Оценка режима
и устойчивости

Проверка трактов

Проверка алгоритмов

04 Прототипирование радиолиний и модемов



Собственный PHY

Каналы
связи

Видео и
данные

Телеметрия и
управление

05 Радиоизмерения и анализ сигналов



Прием

Запись
T/O

Анализ
спектра

Генерация

06 Для вузов и НИР

Лабораторные работы, учебные стенды и опытные проекты на базе Engee, GNU Radio, LabVIEW и MATLAB





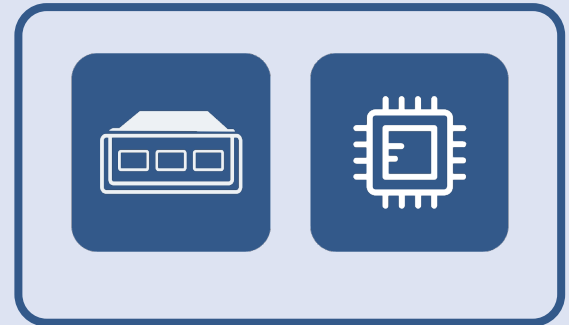
Что такое Engage?



Engage – российская платформа для математических расчетов, модельно-ориентированного проектирования и программирования сложных технических систем.



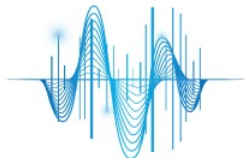
ГЕНЕРАЦИЯ КОДА,
ПОЛУНАТУРНОЕ ТЕСТИРОВАНИЕ



Engae для Радиотехнических Систем



ЦОС



Дискретные

Фильтры

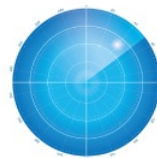
Сигналы

Статистика

Математика

Источники

Радиолокация



Обнаружение сигналов

Среда и цель

Передатчики и приемники

Генераторы сигналов

Связь



Каналы

Приемники

Перемежение

Фильтры

Модуляци

Источники

Радиочастотные ухудшения и компоненты



Верифицированные библиотеки системных объектов и функций



- EngaeDSP
- EngaePhased
- EngaeComms
- EngaeRF
- EngaeRadar



Автоматическая генерация кода Verilog, C++

Мульти-доменное моделирование в

Edgee



newmodel_1 * +



newmodel_1



Библиотека



Готов

Euler

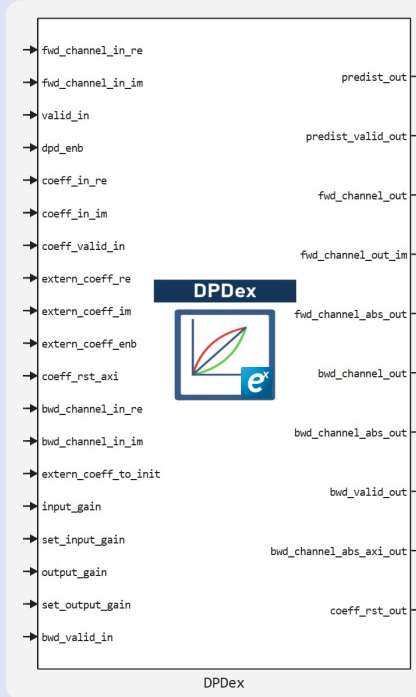
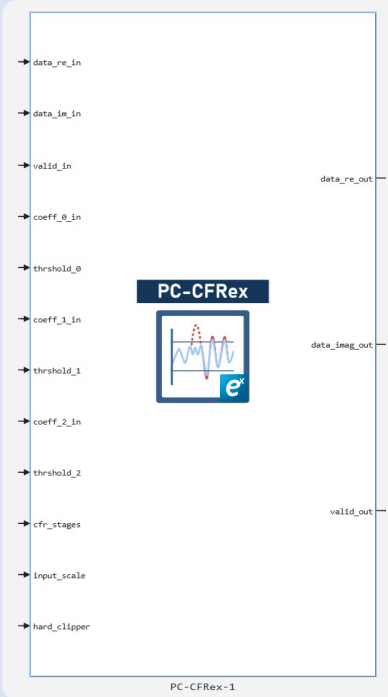
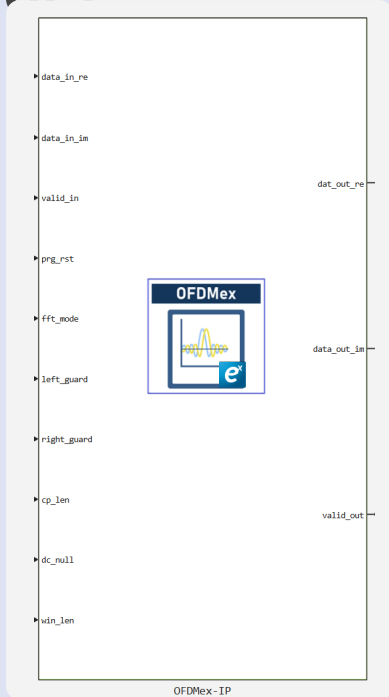
10с

0%

100%



Модели IP-ядер собственной разработки в среде моделирования Encore



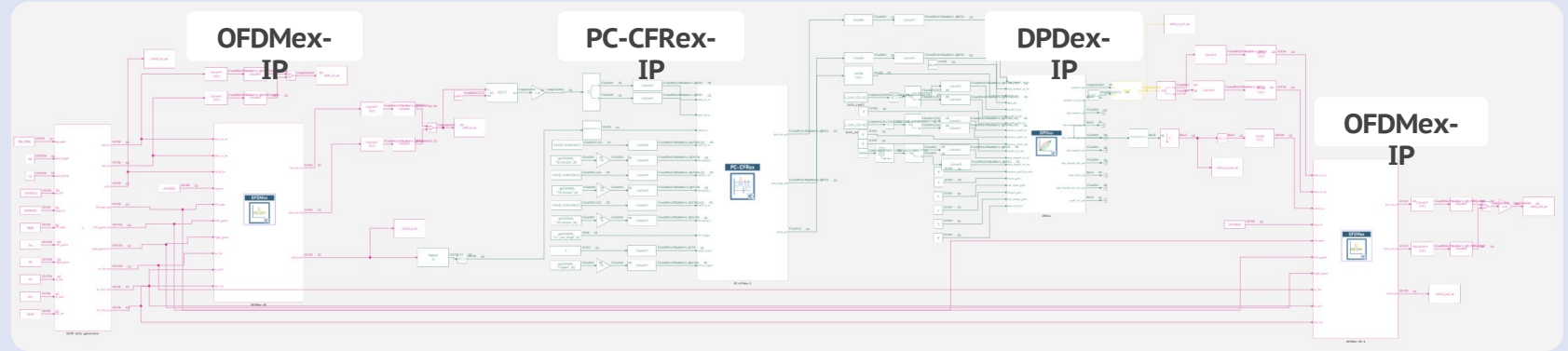
Референс-модель DFE в Engae



$F_s = 30.72$ МГц

$F_s = 122.88$ МГц

$F_s = 30.72$ МГц



Программируемый генератор данных QAM-256

OFDM-модуляция

Снижение пик-фактора

Ввод цифровых предискажений

OFDM-демодуляция



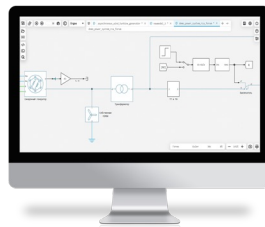
COM

- Микроконтроллеры и отладочные платы Arduino, STM, Миландр, МИК32
- Промышленные логические контроллеры (ПЛК)
- Платы сбора данных
- GPS-модули
- Медиаустройства



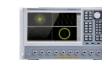
UDP

- Средства визуализации: Unigine или Unreal Engine
- Камера для машинного зрения



VISA

- Генераторы сигналов
- Осциллографы
- Спектроанализаторы и векторные анализаторы цепей (VNA)
- Системы сброса данных и коммутации



CAN

- Блоки управления двигателями (ЭБУ)
- Электронные системы ТС
- Медицинские приборы
- Робототехнические системы





ЭКСПОНЕНТА

ЦЕНТР ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ
И МОДЕЛИРОВАНИЯ

Спасибо за внимание!

ЦИТМ «Экспонента» обеспечивает
научно-производственные организации
передовыми технологиями моделирования
для ускорения научных исследований
и разработки сложных технических систем.

☎ +7 (495) 009 65 85

🌐 info@exponenta.ru

📍 [@exponenta_ru](https://www.instagram.com/exponenta_ru)





ЭКСПОНЕНТА

ЦЕНТР ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ
И МОДЕЛИРОВАНИЯ

PIITM SDR USRP в Engee



**СДЕЛАНО
В РОССИИ**

**МИНПРОМТОРГ
РОССИИ**





Software Defined Radio - программно-определяемое радио
Universal Software Radio Peripheral - универсальный
программный радиоприёмник

- * Коротко о ПИТМ SDR USRP
- * Подключение и настройка ПИТМ SDR USRP
- * Формирование и передача сигналов
- * Аналоговая петля
- * Захват Wi-Fi сигнала
- * Дальнейшие планы



PI1M SDR USRP



SDR-платформа на основе Zynq
XCZU15EG Ultrascale+ и **ADRV9009**:

2xTX - до **450МГц** полоса каждого канала

2xRX - до **200МГц** полоса каждого канала

2xORX - до **450МГц** полоса каждого канала

Диапазон частот: **75МГц-6000МГц**

Выходная мощность - **15 дБм**



Вычислительные ресурсы:

LUT - 341K

DSP - 3528

BRAM - 744

APU - 4xARM-A53

DDR4 - 4GB

RPU - 2xARM-R5



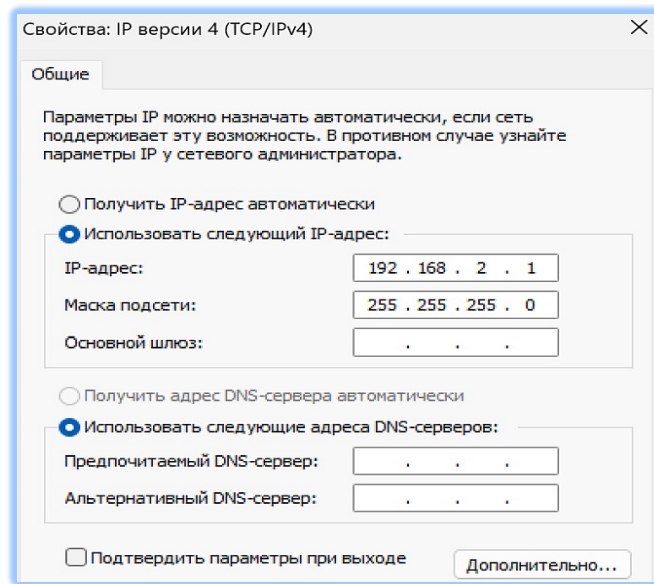
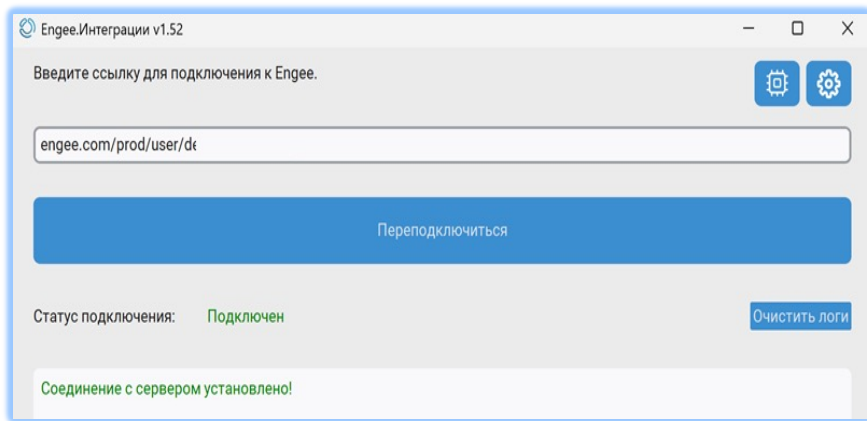
Основные интерфейсы: **2xSFP**
(до 28Gb), **2x1GbEth**, **2xUSB3.0**,
2xCAN, **NVME M.2**



Подключение РИТМ SDR USRP



```
engee.package.install("Engee-Device-Manager")
engee.package.start("Engee-Device-Manager")
```



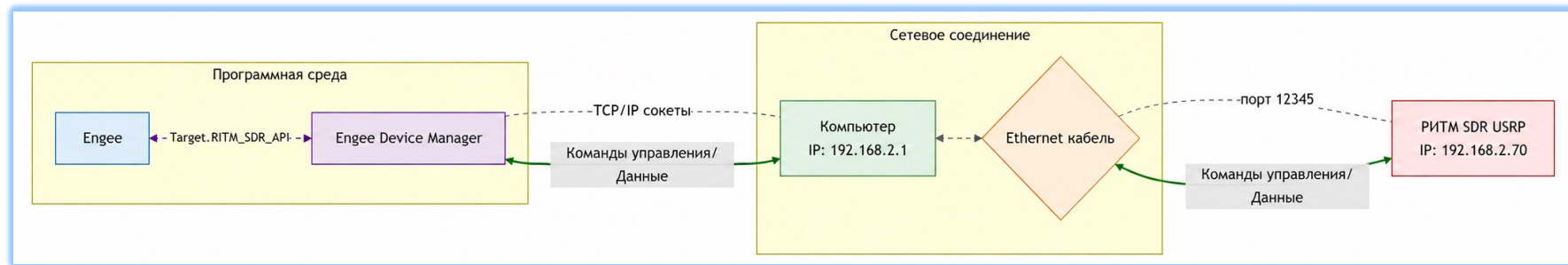
GitBash: ./engee-device-manager.exe -d

```
523ur@Uran MINGW64 ~/OneDrive/Desktop
$ ./engee-device-manager.exe -d
2026-05-20 17:35:00.187 INFO | devices:<module devices>:17 - Successfully imported module: devices.can.can
2026-05-20 17:35:00.201 INFO | devices:<module devices>:28 - Successfully imported module: devices.com.com
2026-05-20 17:35:00.223 INFO | devices:<module devices>:44 - Successfully imported module: devices.hid.hid
2026-05-20 17:35:00.289 INFO | devices:<module devices>:55 - Successfully imported module: devices.modbus.modbus_master
2026-05-20 17:35:00.389 INFO | devices:<module devices>:66 - Successfully imported module: devices.visa.visa
2026-05-20 17:35:00.415 INFO | devices:<module devices>:77 - Successfully imported module: LCard L-502
2026-05-20 17:35:00.425 INFO | devices:<module devices>:89 - Successfully imported module: Analog Discovery
2026-05-20 17:35:00.439 INFO | setup.cli:cli:35 |
```

Engее.Интеграции и SOCKET



SOCKET — интерфейс для обмена данными между процессами в Engее и вашим ПК.



Engae.Интеграции и SOCKET



На основе рассмотренных базовых операций (*Socket*, *connect*, *send*, *receive*, *close*) и принципов постобработки данных построена вся библиотека управления РИТМ SDR USRP.

```
1 response = SOCKET.receive(socket, 35)
2 answer = response == nothing ? "" : String(response.data)
```

```
"COMMAND:SDR_SERVER_VERSION=v5.1.5\r\n"
```

```
1 const SDR_PORT = 12345
2 mutable struct RITMClient
3     server_ip::String
4     socket::Union{Nothing, SOCKET.Socket}
5     function RITMClient(server_ip::String="192.168.2.70")
6         return new(server_ip, nothing)
7     end
8 end
9 function connect(client::RITMClient)::Bool
10     try
11         socket = SOCKET.Socket("AF_INET", "SOCK_STREAM")
12         SOCKET.connect(socket, client.server_ip, SDR_PORT)
13         client.socket = socket
14         @logmsg "✓ Подключено к $(client.server_ip):$SDR_PORT"
15         return true
16     catch e
17         @logmsg "X Ошибка подключения: $e"
18         return false
19     end
20 end
```

Targets.RITM_SDR_API



```
1  engee.package.install("Engee-Device-Manager")
2  engee.package.start("Engee-Device-Manager")
3  using Main.EngeeDeviceManager.Targets.RITM_SDR_API
4  import Main.EngeeDeviceManager.Targets.RITM_SDR_API
```

```
1  RITM_SDR_API.@log_on
2  client = RITM_SDR_API.RITMClient("192.168.2.70")
3  !RITM_SDR_API.connect(client) && (RITM_SDR_API.@logmsg "Ошибка подключения"; return);
```



Логирование включено
✓ Подключено к 192.168.2.70:12345
Автоматически создан лог-файл: logs/log_2026-05-25_10-55-28.log

```
1  RITM_SDR_API.@logmsg "Версия сервера: ${RITM_SDR_API.get_version(client)}";
```

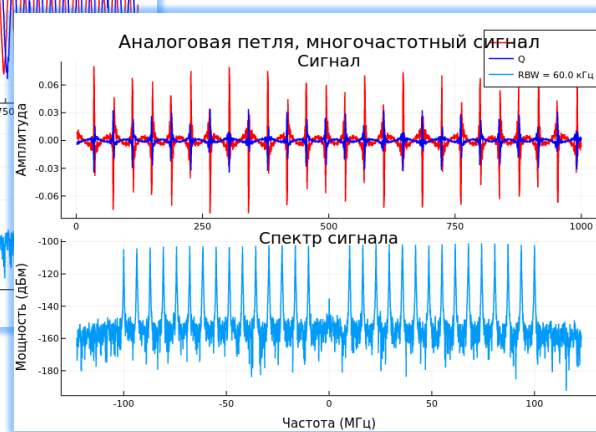
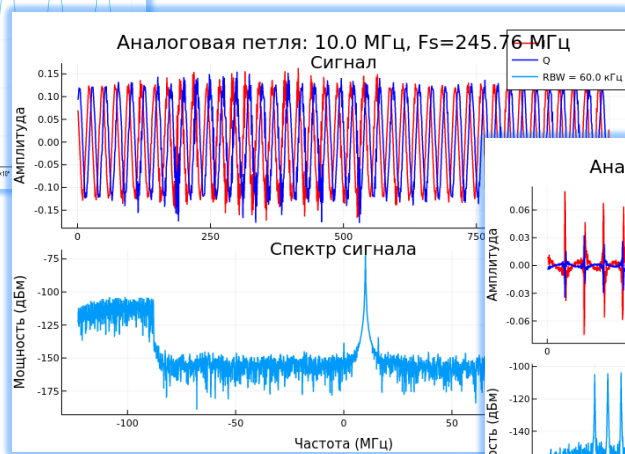
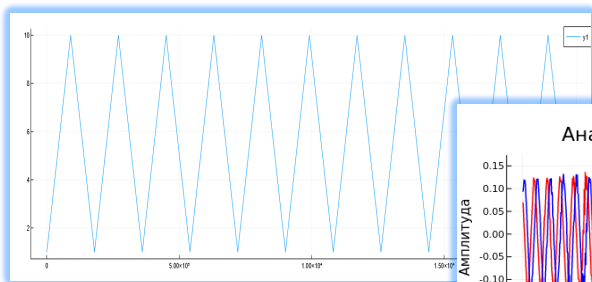


Версия сервера: v5.1.5

Формирование и передача сигналов (Аналоговая петля)



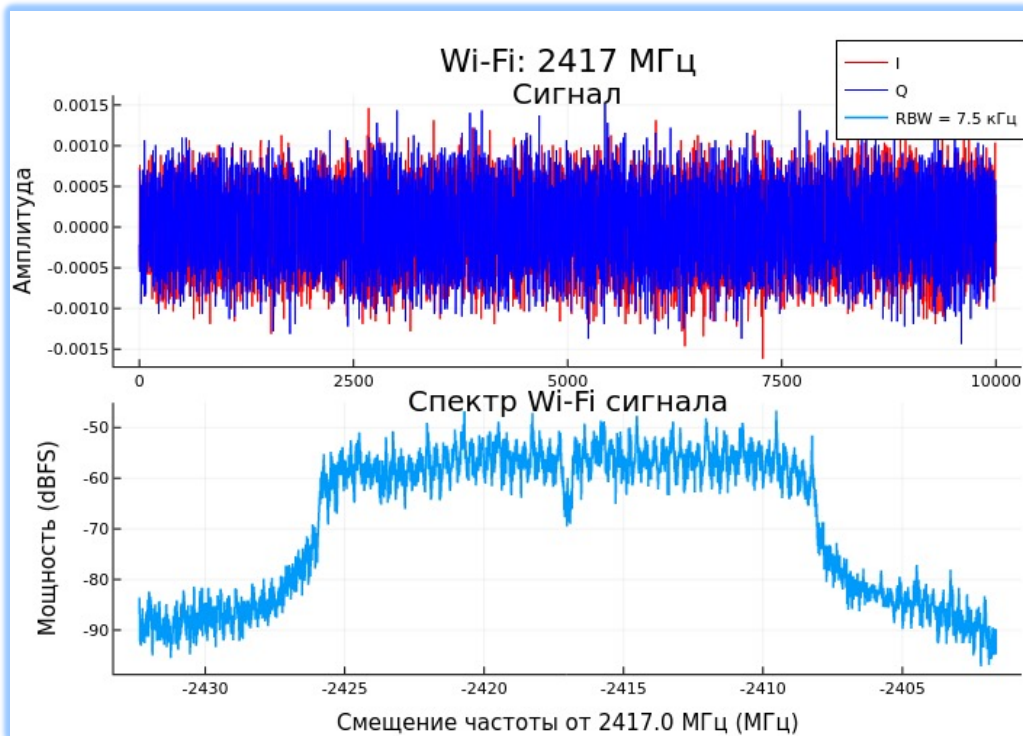
Аналоговая петля — это режим работы PITM SDR USRP, при котором переданный сигнал замыкается прямо в эфире и принимается обратно через радиочастотный тракт.



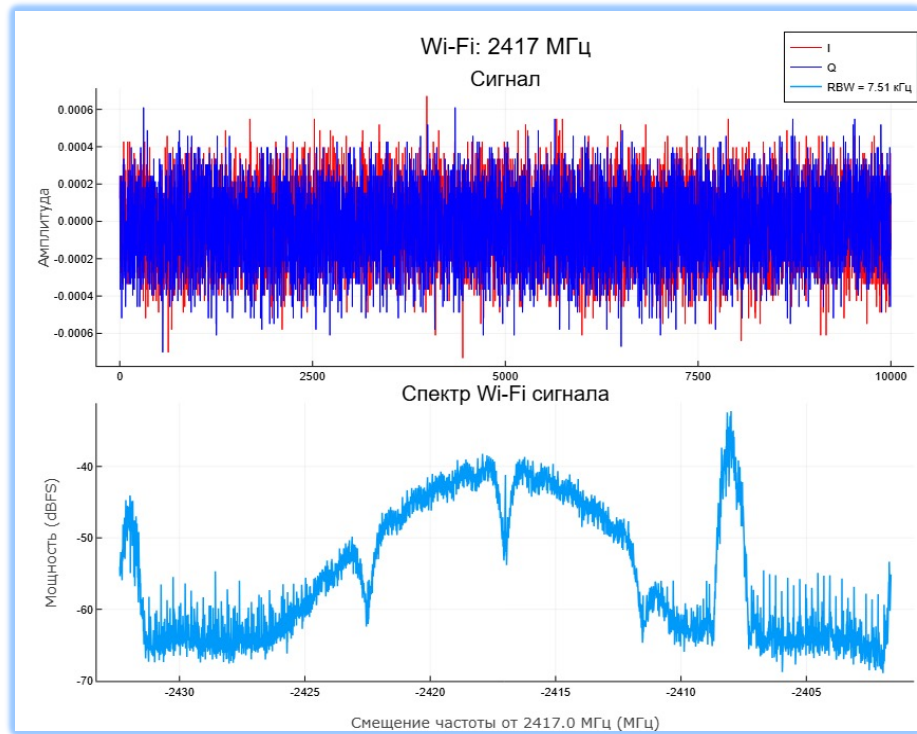
Захват Wi-Fi сигнала



```
1 RITM_SDR_API.@log_on
2 wifi_freq_mhz = wifi_channel_frequency("2.4GHz", 2)
3 RITM_SDR_API.@logmsg "Частота Wi-Fi: $wifi_freq_mhz МГц";
4
5 client = RITM_SDR_API.RITMClient("192.168.2.70")
6 RITM_SDR_API.connect(client) || (exit(1))
7 configure_sdr(client, Int(wifi_freq_mhz * 1e6))
8 RITM_SDR_API.agc_set_mode(client, 1)
9 sleep(0.5)
10
11 iq_data = receive_iq(client, 600000)
12 if iq_data != nothing
13     RITM_SDR_API.@logmsg "записано $(length(iq_data)) отсчетов";
14 end
15
16 RITM_SDR_API.disconnect(client)
17 RITM_SDR_API.close_log();
```



Захват Wi-Fi сигнала



Дальнейшие планы

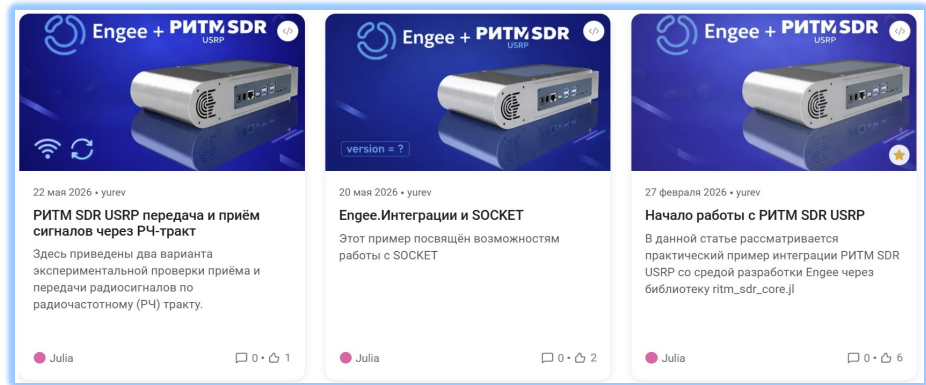


Примеры

- Демодуляция LTE-подобных сигналов через IP ядра PИТM SDR USRP
- FM-радиоприёмник в скриптах и моделях
- Глушилка Wi-Fi сигнала

Разработка

- Функции потоковой обработки
- Блоки потоковой обработки
- Приложение под управление PИТM SDR USRP





ЭКСПОНЕНТА

ЦЕНТР ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ
И МОДЕЛИРОВАНИЯ



+7 (495) 009 65 85

info@exponenta.ru

@exponenta_ru



Спасибо за внимание!

