Требования предъявляемые к операционным системам в АСУ ТП. На основе опыта ОСРВ Embox

Антон Бондарев

OS:Day, 5 июня 2026

Требования к ОС для АСУ ТП

Высказывание одного разработчика АСУ ТП:

“Когда я говорю о системе (АСУ-ТП) в ней вообще не должно быть заметно ОС, она только мешает“

Требования к ОС для АСУ ТП

- Время реакции:
  - Химическое производство: нас **не интересует** время реакции, у нас процессы занимают минуты
  - Металлургическое производство: нас **прежде всего интересует** время реакции. Техпроцессы прокатных станов микросекунды

Требования к ОС для АСУ ТП

- Время реакции:
  - Химическое производство: нас **не интересует** время реакции, у нас процессы занимают минуты
  - Металлургическое производство: нас **прежде всего интересует** время реакции. Техпроцессы прокатных станов микросекунды

Требования к ОС для АСУ ТП

- Надежность: Должна быть надежной
- Безопасность: Информационная безопасность не должна мешать функциональной безопасности
- Протоколы: должна обеспечивать все протоколы и интерфейсы

Требования к ОС для АСУ ТП

- Разделили на три уровня:
  - Полевой: время реакции, надежность, дешевые платформы
  - ПЛК: время реакции, надежность, различные протоколы и аппаратные платформы
  - SCADA: Облака, виртуализация и стандартизация

Требования к ОС для АСУ ТП

Должна обеспечивать совместимость прикладных программ написанных на языках МЭК

Разработка встроенных систем

“Когда я говорю о системе (АСУ-ТП) в ней вообще не должно быть заметно ОС, она только мешает“

ОС должна обеспечивать запуск и работу необходимого прикладного ПО

Требования к ОС для АСУ ТП

- Новая реальность: АСУ объекты КИИ и появление ИИ. Информационная безопасность критична.
  - Нужны доверенные аппаратные платформы
  - Нужны новые подходы к обеспечению
- Старая реальность:
  - Проектирование системы, а не отдельных частей
  - Нужен системный подход. Метод построение систем предприятий

Требования к ОС для АСУ ТП

- Использование российской микроэлектроники (доверенных платформ)
- Использование новых подходов к безопасности
- Создание **экосистемы**

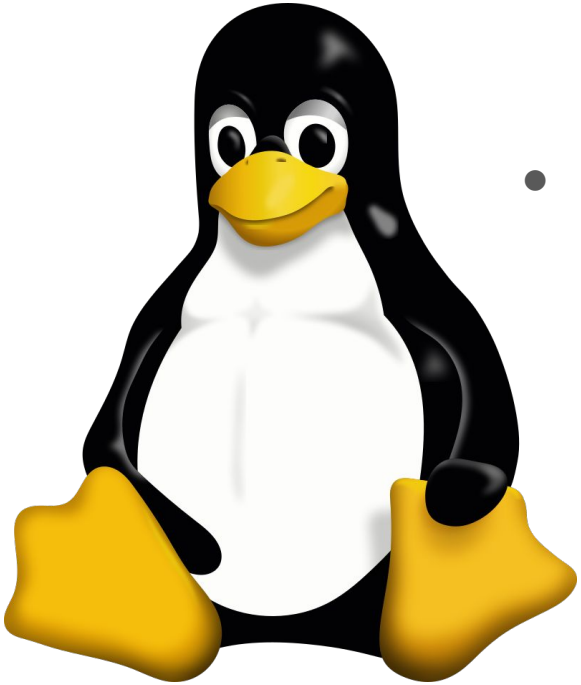
Примеры экосистем

- **Arduino** — аппаратная платформа вокруг открытого IDE, позволяющего быстро делать простые вещи
- **ESP** — аппаратная платформа вокруг открытого IDE, позволяющего быстро делать WiFi устройства
- **STM32** — экосистема для микроконтроллеров, основа собственное открытое IDE которая позволяет быстро создавать устройства
- **RaspberryPi** — аппаратная платформа для быстрого создания легких специализированных ПК. Получила развитие благодаря открытой экосистеме Линукс
- **Linux** — наиболее развитая программная экосистема

Свойства экосистем

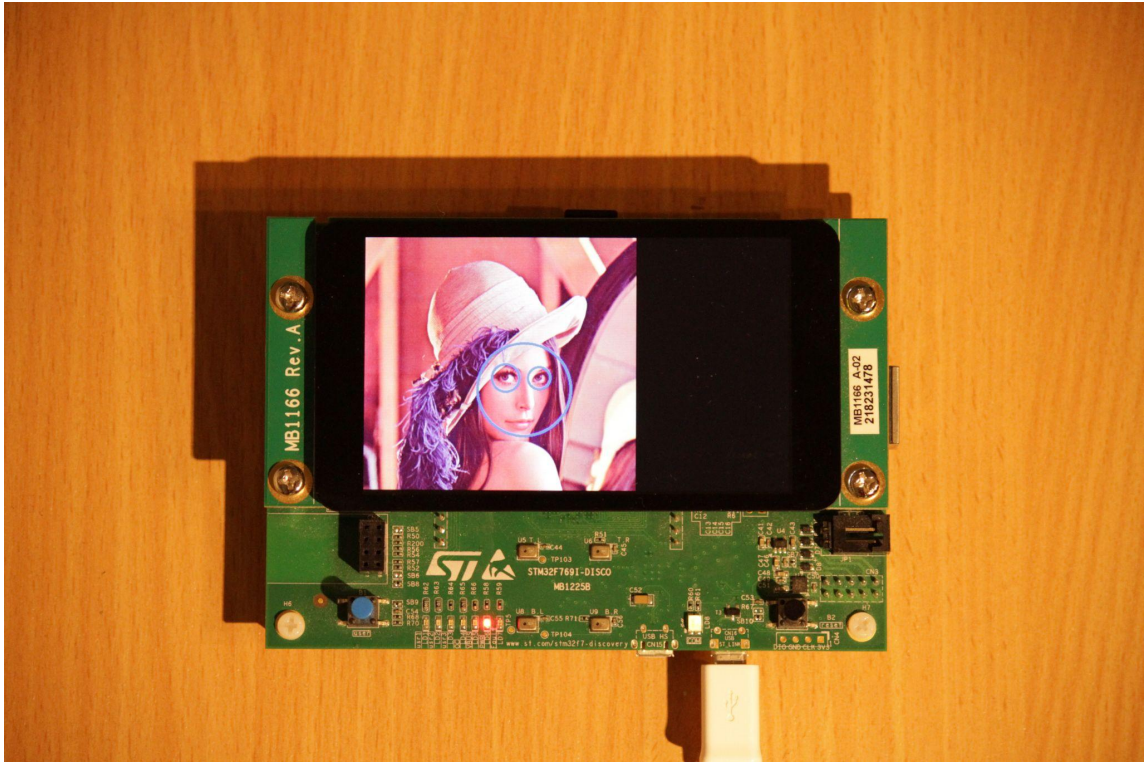
- **Открытость —** возможность присоединиться к экосистеме разным участникам. Повышает доверие и позволяет консолидировать усилия.
- **Технологичность —** наличие технического преимущества перед существующими экосистемами
- **Масштабируемость —** однотипность решаемых задач и подходов
- **Экономичность —** возможность получить выгоду от внедрения и развития экосистемы

Обзор: Linux



- Самая лучшая ОС в мире:
  - Гигантское количество ПО на любой вкус
  - Самая широкая поддержка оборудования
  - Самое большое комьюнити
- Универсальное ядро:
  - Требует относительно много ресурсов
  - Трудно обеспечить реал-тайм
  - Трудно сертифицировать и верифицировать (много кода)
  - Сложности с безопасностью (следствие универсальности системы)

Linux без Linux



А что так можно было? :)

Embox

Embox — свободная операционная система реального времени (RTOS), разрабатываемая для встроенных систем.

Основная идея использование ПО **Линукс** в более безопасном и детерминированном, менее ресурсоемком и энергопотребляющем окружении

Embox

Операционная система под конкретную задачу, основанная на **специальном DSL языке** и использующая:

- Статическую информацию о задачах устройства
- Статическую конфигурацию системы
- Статический анализ зависимостей
- Статическую проверку параметров системы

Embox build system:
KBuild + OpenEmbedded + DevTree

Embox

- Свойства полученной системы:
  - Отсутствует избыточная функциональность
    - Экономия вычислительных ресурсов
    - Экономия энергопотребления
  - Упрощение сертификации системы
  - Невозможность запуска вредоносных приложений

Embox на российских чипах

- Микроконтроллеры
  - RISC-V
    - НИИЭТ ВГ015
    - Микрон Амур32
    - Миландр MDR1206fi
    - Байкал Baikal-u
  - ARM
    - ЭЛВИС ELIoT

Embox на российских чипах

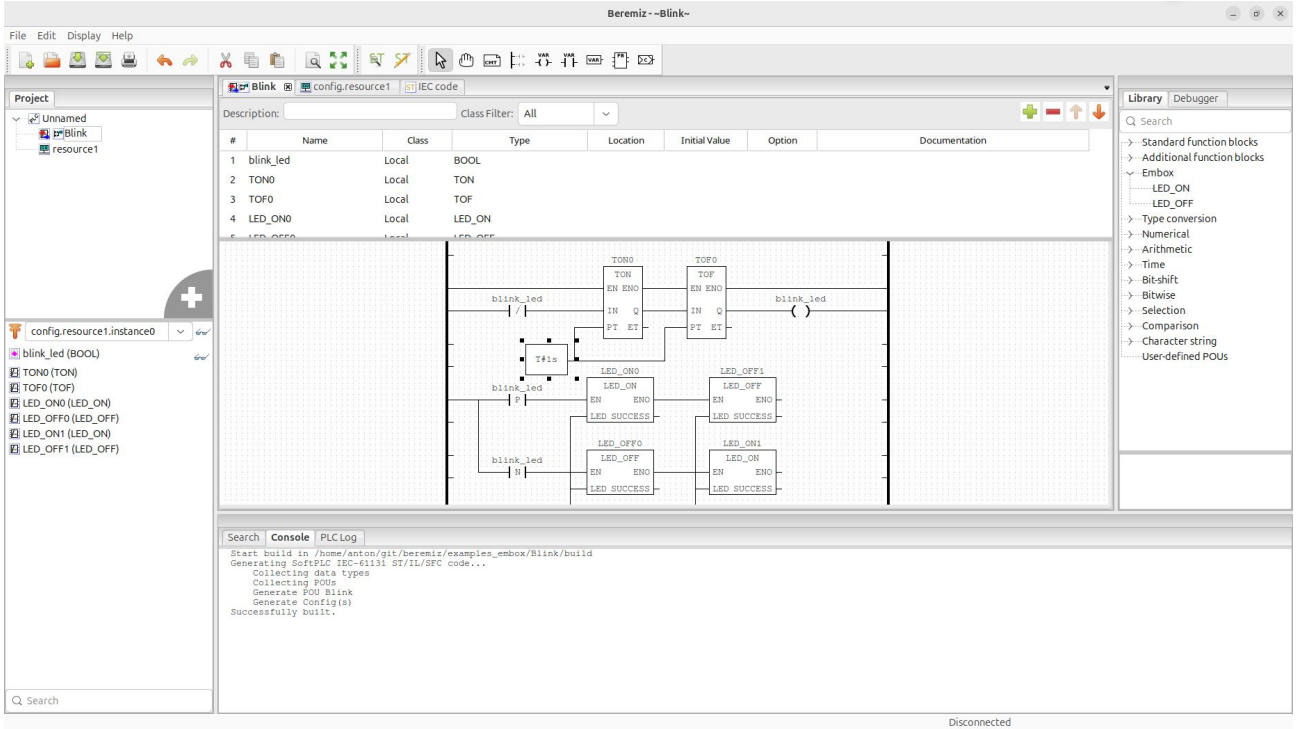
- НТЦ Модуль
- Байкал-Электроникс
- МЦСТ
- НИИСИ (КОМДИВ)
- ЭЛВИС

Приложения АСУ-ТП

- В АСУ-ТП применяются (Программируемые Логические Контроллеры) ПЛК. Их разработка регламентирована стандартом МЭК 61131
- Раздел 3 (МЭК 61131-3) специфицирует языки программирования для ПЛК
- Открытые проекты для средства разработки для ПЛК
  - Beremiz (МЭК 61131-3)
  - 4Diac forte (IEC 61499)

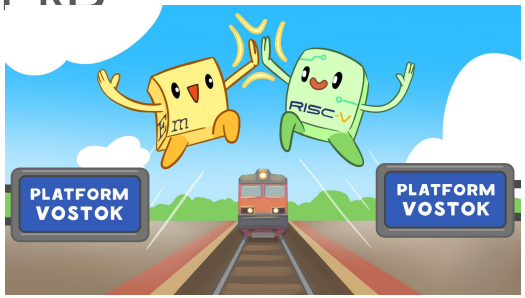
Приложения АСУ-ТП

- Генерируются допустимые функциональные блоки



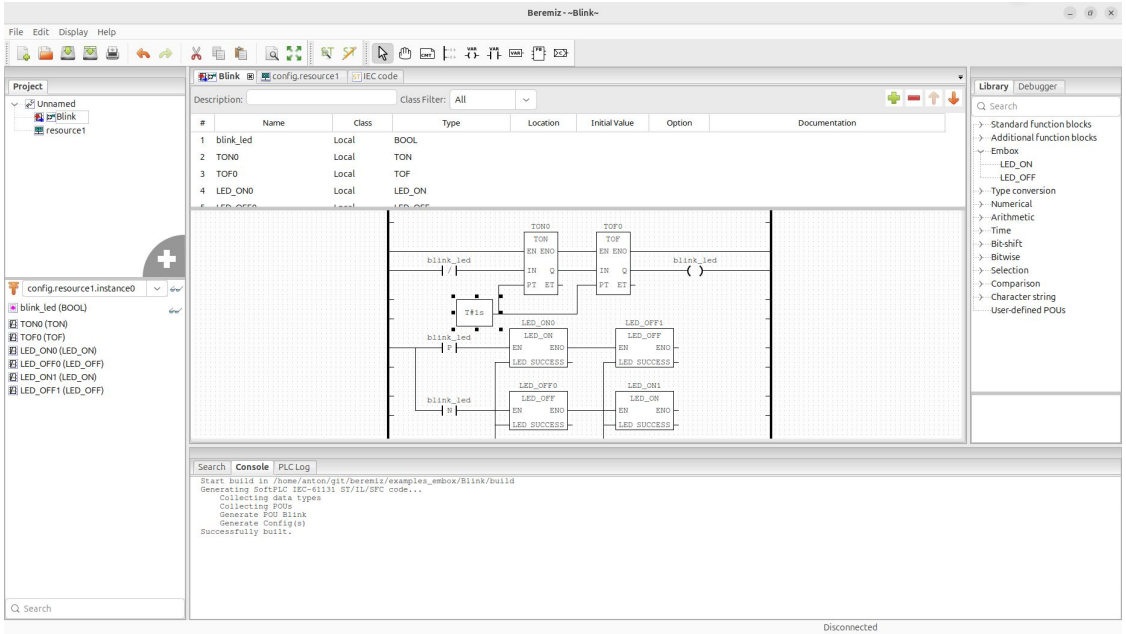
Embox на К1921ВГ015

- НИИЭТ (входит в ГК Элемент)
- RISC-V, ПЗУ 1mb, ОЗУ 256 + 64 кБ
- Разработали только драйвера
- Перенос реальных проектов



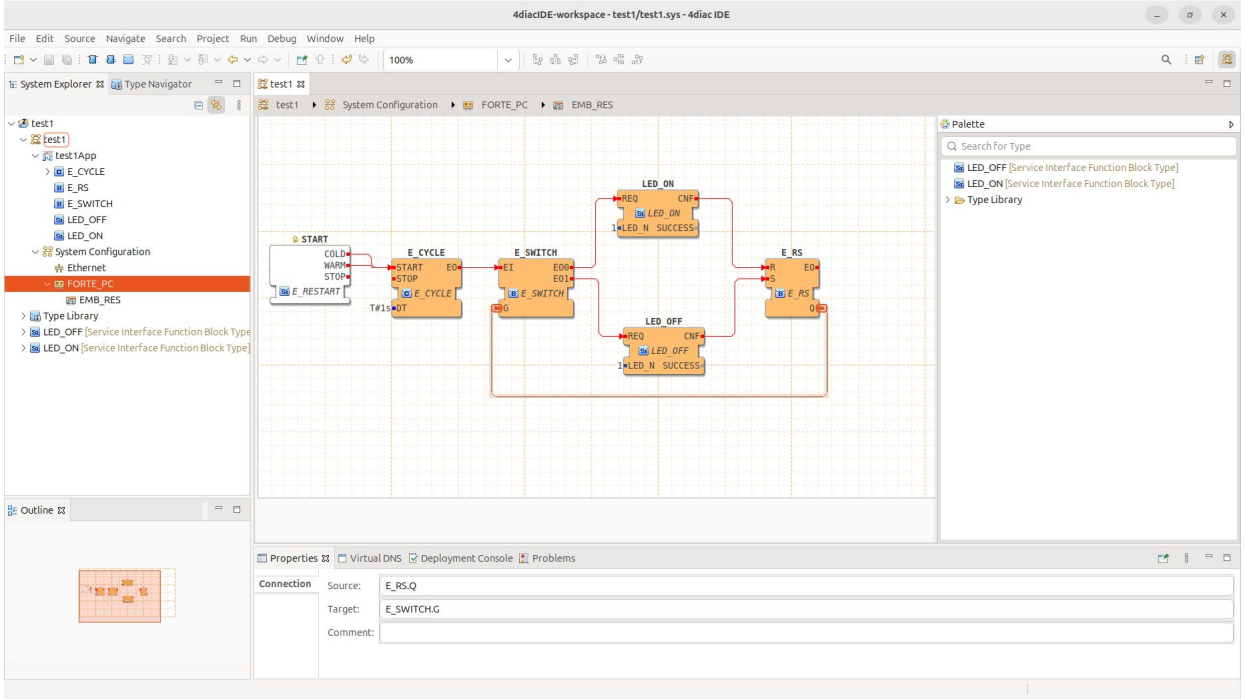
Приложения АСУ-ТП (МЭК 61131)

- Открытый Beremiz. Генерируются требуемые блоки



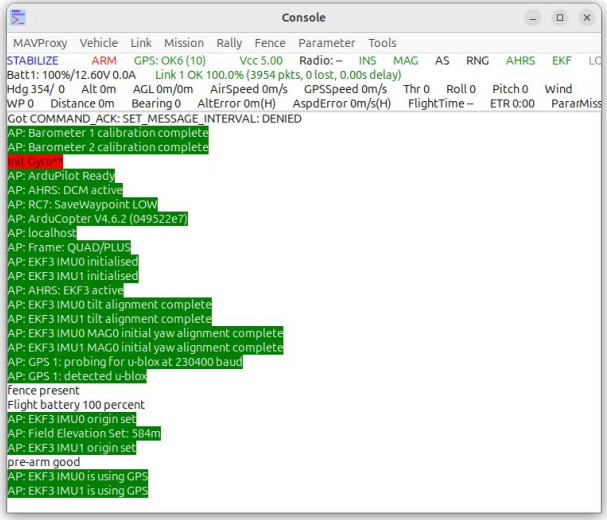
Приложения АСУ-ТП (МЭК 61499)

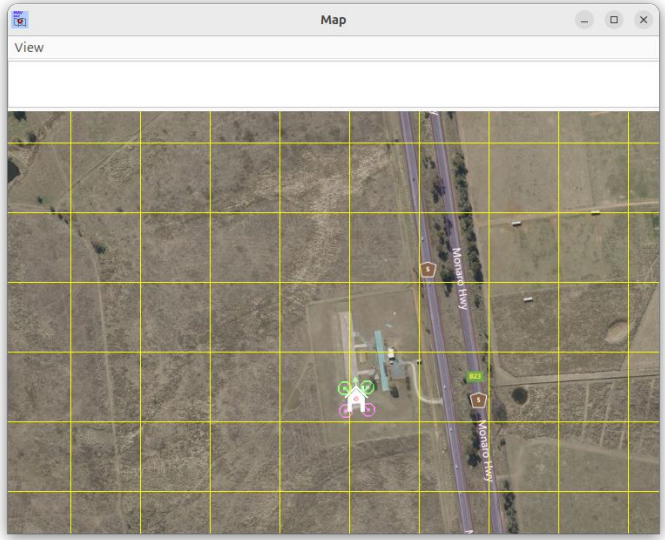
- 4DIac-forte. Запустили runtime в том числе на российских МК



Ardupilot (БАС)

- Симуляция
- STM32
- Российские МК на RISC-V (в том числе ВГ015)
- https://github.com/embox/embox/wiki/Ardupilot





Embox - открытая экосистема

- Открытая платформа позволяющая:
  - Существенно расширить возможности для построения умных устройств на российской элементной базе
  - Ускорить и упростить разработку конечных устройств
  - Сократить расходы на поддержку как конечных устройств так и микроэлектроники
  - Использовать в образовании и промышленности
  - Развивать в нужных направлениях

Embox - открытая экосистема

	Вклад	Выгода
Производители микроэлектроники	Разработка драйверов и поддержка чипов	Прикладное ПО для создания конечных устройств
Разработчики платформы	Разработка драйверов Поддержка чипов Разработка и поддержка системной части	Снижение затрат на поддержание экосистемы
Производители устройств	Разработка драйверов Поддержка своих платформ Разработка библиотек	Прикладное ПО для создания конечных устройств. Микросхемы с драйверами
Интеграторы	Поддержка своих платформ Разработка библиотек	Готовые к применению аппаратные платформы с драйверами
Промышленные компании	Финансирование Апробация в промышленности Формулировка требований	Снижение затрат на поддержание своей цифровой инфраструктуры Улучшение качества устройств

Embox - открытая экосистема

https://niiet.ru/21-04-2026/

Производитель микроконтроллеров **НИИЭТ** (входит в Группу компаний «Элемент», MOEX:ELMT), компания – разработчик комплектующих для БпЛА «**Чистое небо**» **в кооперации** с разработчиком **открытой** российской операционной системы реального времени (ОСРВ) **Embox** успешно адаптировали ПО с открытым исходным кодом для автономных полетов ArduPilot под российские микроконтроллеры К1921ВГ015 на архитектуре RISC-V.

https://github.com/embox/ardupilot

Экосистема

Экосистема должна включать множество компонент и участников. В том числе образовательные и научные коллективы и программы, базовые технологии используемые в АСУ ТП, коммерческие компании производящие продукты и оказывающие услуги, промышленные компании и так далее.

Прикладное ПО АСУ ТП нуждается в операторе аналогичном “Центру исследований безопасности системного ПО”

Embox - может закрыть часть проблем в АСУ. Но в экосистеме необходимы и другие ОС закрывающие конкретные проблемы отрасли. Как минимум это Линукс с достаточным уровнем доверия. И KasperskyOS обеспечивающая общую информационную безопасность

Ссылки

Страница проекта

http://embox.github.io/

Репозиторий проекта
https://github.com/embox/

Youtube канал

https://www.youtube.com/@embox-rtos

Телеграмм канал
https://t.me/embox_chat

Contacts

Anton Bondarev

- anton.bondarev2310@gmail.com
- **@anton_bondarev_embox**
-