



Сертифицируемая бортовая ОСРВ JetOS



Солоделов Ю.А., вед. инженер

24 мая 2017 г.

Москва, "OS DAY 2017"



Содержание

❖ 1. Предпосылки

- Концепция ИМА (на чем запускается бортовое ПО?)
- Проблемы с VxWorks 653 (как возникла задача создания ОСРВ?)
- Сертифицируемость по DO-178С (чем осложняется задача создания ОСРВ?)

❖ 2. Организация работы

- НИР 2016 г.
- НИР 2017-19 гг.

❖ 3. Требования

- ARINC 653
- Поддержки многоядерности
- Платформонезависимость
- Кибербезопасность

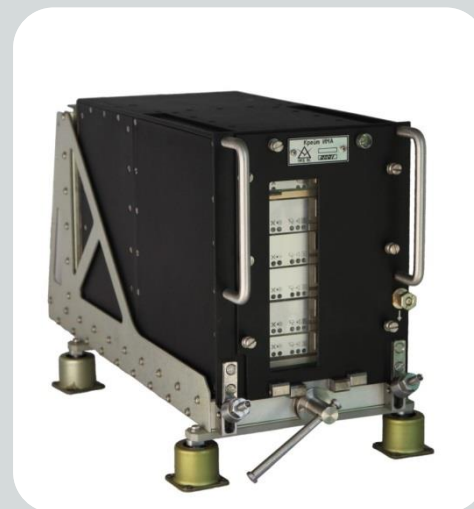
❖ 4. Аспекты

- Язык программирования
- Инструменты
- Перспективы в других отраслях



Предпосылки: концепция ИМА

❖ Интегрированная модульная авионика



- Миниатюризация компонентной базы
- Несколько функциональных приложений на одном вычислителе
- Критически важна ОСРВ



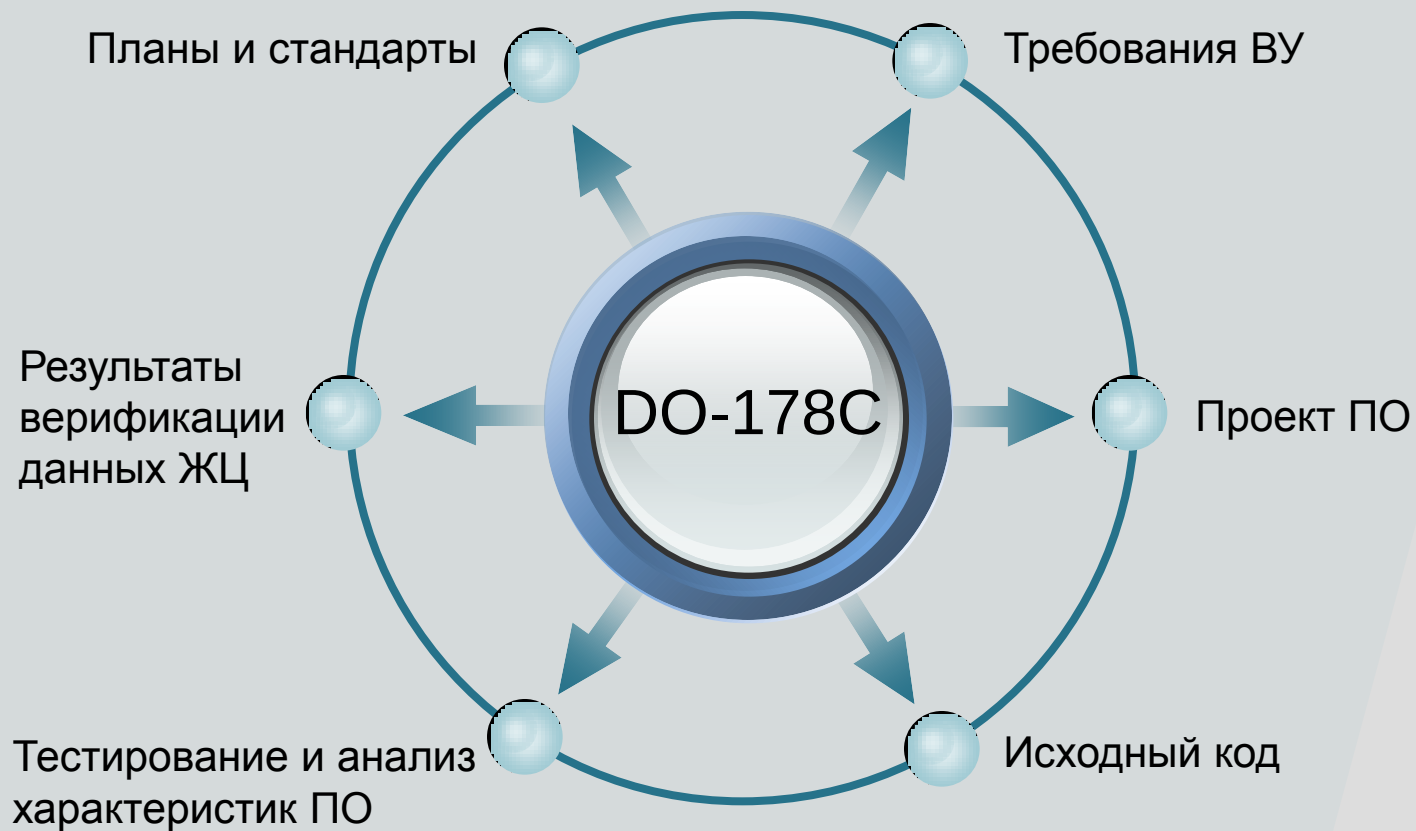
Предпосылки: проблемы с VxWorks 653

- ❖ VxWorks 653 - Основная ОСРВ, применявшаяся в отечественной программе ИМА до 2015 г.
- ❖ Wind River (создатели VxWorks) – подразделение Intel
- ❖ Сопоставимые продукты в мире:
 - PikeOS (Германия)
 - Integrity (США)
 - LynxOS-178 (США)





Предпосылки: сертифицируемость по DO-178C





Организация работы: две НИР

НИР

2016

- Предварительное прототипирование с целью подготовки к созданию архитектуры
- Создание задела по процессной работе, по требованиям и архитектуре

2017-19

- Завершение постановки процессов разработки в соответствии с DO-178C
- Поэтапная разработка всех данных жизненного цикла на базе имеющегося задела



Организация работы: НИР 2016 г.

- ❖ Первоначально планировалась постановка процессов для нескольких компаний, создающих несколько разнородных компонентов: файловую систему, графику и т.п.
- ❖ Параллельно с постановкой процессов DO-178C проводилось прототипирование и создание задела по требованиям и исходному коду
- ❖ Прототипирование велось на базе кода “РОК” с открытым исходным кодом





Организация работы: НИР 2017-2019 гг.

- ❖ Продолжение постановки процессов для меньшего количества участников (ГосНИИАС, ИСП РАН и ДС БАРС)
- ❖ Уменьшение состава компонентов: акцент перенесен на ядро и основные системные компоненты
- ❖ Разработка всех данных жизненного цикла DO-178C на основе созданного задела: требования ВУ, проект ПО, код, тесты, результаты верификации...
- ❖ Работы по графическим компонентам продолжаются, но (пока что) независимо от OCPV



DZ•SYSTEMS





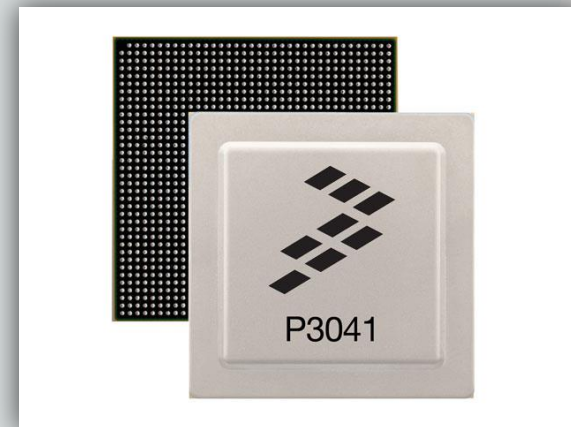
Требования: ARINC 653

- ❖ ARINC 653 – стандарт, регламентирующий режимы работы бортовых функциональных приложений и их API
- ❖ В JetOS закладывается поддержка редакции 2015 г. (учитывающей многоядерность)
- ❖ Поддержка ARINC 653 позволяет бесшовно интегрироваться с другими бортовыми стандартами – графикой (ARINC 661), бортовой сетью AFDX (ARINC 664) и др.
- ❖ Поддержка ARINC 653 открывает доступ к бортовым legacy-приложениям



Требования: поддержка многоядерности

- ❖ Как правило, авиационные аппаратные платформы оснащаются многоядерными COTS-чипами
- ❖ Проблема влияния ядер (из-за совместного использования ресурсов) до сих пор в мире не решена
- ❖ Вызов, стоящий перед всеми разработчиками бортового ПО в мире
- ❖ Разработка ведется для четырехъядерного процессора





Требования: платформонезависимость

- ❖ Основной архитектурой процессора для авионики по-прежнему является PowerPC, однако ARM завоевывает позиции
- ❖ Также широко используется MIPS, особенно в военной технике
- ❖ В перспективе – Эльбрус?
- ❖ Best practices: в архитектуру закладывается разделение на платформонезависимую и платформозависимую (BSP) части
- ❖ Платформонезависимая: пишется на C и включает ядро и системные компоненты
- ❖ BSP: пишется для каждой платформы с применением ассемблера





Требования: кибербезопасность

- ❖ В НИР 2016 года проводилось два исследования на тему кибербезопасности для бортовой ОСРВ от компании АМДЭФ и кафедры ИУ-8 МГТУ им. Н.Э. Баумана
- ❖ В НИР 2017-19 заложен анализ кода на предмет выявления потенциально опасных конструкций
- ❖ Ситуация с кибербезопасностью упрощается жесткими требованиями ARINC 653, т.е. бортовая ОСРВ изначально находится в менее угрожаемой ситуации, нежели обычные ОСРВ





Аспекты: язык программирования

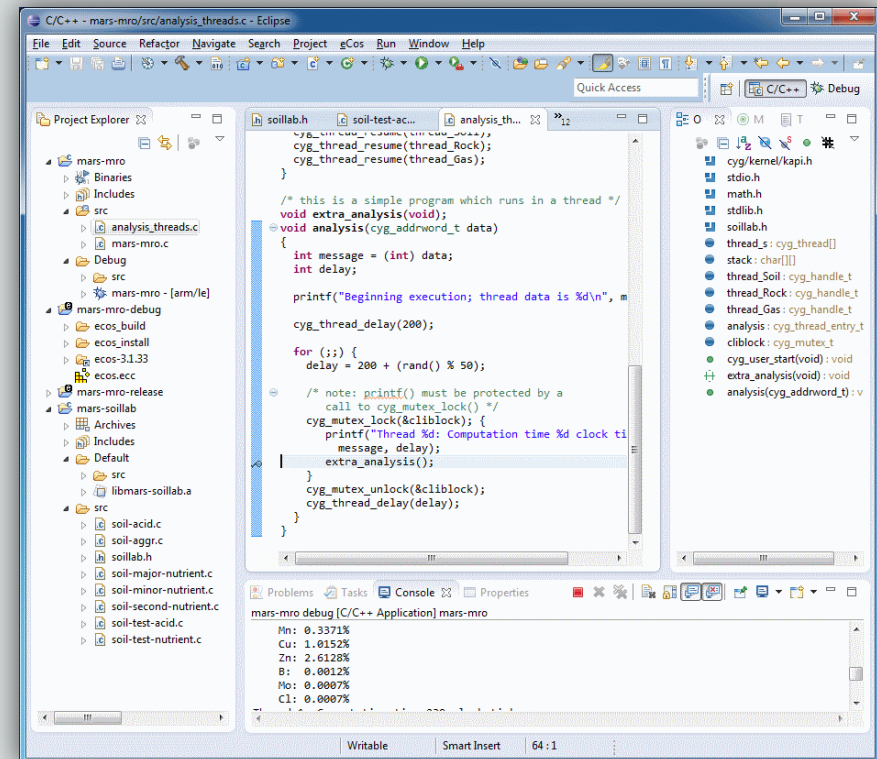
- ❖ Разработка ведется на языке C в соответствии со стандартом на кодирование, разработанным для данного проекта (как требует DO-178C)
- ❖ В принципе использование C++ в бортовом ПО не запрещается (см. DO-332 – дополнение к DO-178C), но сильно ограничивается, т.к. объектно-ориентированная парадигма и синтаксические возможности C++ сильно усложняют верификацию кода





Аспекты: инструменты

- ❖ IDE (среда разработки функциональных приложений)
- ❖ Монитор (компонент, позволяющий отслеживать состояние системы в реальном времени)
- ❖ Трассировочные компоненты ОСРВ





Аспекты: перспективы в других отраслях

- ❖ Требования, предъявляемые DO-178C – одни из самых жестких в мире
- ❖ Зарубежный опыт показывает, что системы, подготовленные для сертификации по DO-178C, могут успешно сертифицироваться для применения в других областях транспорта и промышленности – в медицине, космосе, железнодорожном транспорте, станкостроении и т.п. (стандарты EN 61508, EN 50128 и т.п.)
- ❖ Наряду с ARINC 653 необходим POSIX (для адаптации legacy-приложений из неавиационных областей)





Спасибо за внимание

Ваши вопросы!



Солоделов Ю.А., ГосНИИАС
yasolodelov@2100.gosniias.ru