

## АВТОМАТИЗАЦИЯ СОЗДАНИЯ КИБЕР-ФИЗИЧЕСКИХ МУЗЕЙНЫХ ЭКСПОНАТОВ С ИСПОЛЬЗОВАНИЕМ СИСТЕМЫ НАУЧНОЙ ВИЗУАЛИЗАЦИИ НА КРИСТАЛЛЕ

© 2021 г. К. В. Рябинин<sup>a,\*</sup>, М. А. Колесник<sup>b,\*\*</sup>

<sup>a</sup> ФГБОУ ВО «Пермский государственный национальный исследовательский университет»,  
614990 Пермь, ул. Букирева, 15, Россия

<sup>b</sup> ГКБУК «Пермский краеведческий музей»,  
614000 Пермь, ул. Монастырская, 11, Россия

\*E-mail: kostya.ryabinin@gmail.com

\*\*E-mail: kolesnik.ma@outlook.com

Поступила в редакцию 10.10.2020 г.

После доработки 20.10.2020 г.

Принята к публикации 12.01.2021 г.

Статья посвящена вопросам автоматизации процесса создания автономных модулей научной визуализации на базе систем на кристалле с настраиваемым осязаемым пользовательским интерфейсом. Такие модули могут быть использованы в роли интерактивных экспонатов в рамках концепции так называемых умных музеев. Ключевой идеей автоматизации является генерация итогового программного обеспечения средствами онтологически управляемой платформы SciVi. В рамках этой платформы путем расширения управляющих онтологий организована поддержка генерации кода для систем на кристалле Raspberry Pi и Orange Pi. Алгоритм работы генерируемого программного обеспечения описывается в платформе SciVi высокоуровневым образом при помощи диаграмм потоков данных. При этом научная визуализация имеет аппаратную поддержку через графический API OpenGL ES, а поддержка осязаемого пользовательского интерфейса обеспечивается подключением специализированных библиотек и средств операционной системы для взаимодействия с внешними периферийными устройствами. Эффективность предложенных методов и средств подтверждена на практике при разработке нескольких кибер-физических экспонатов для выставки «Превращения» в Детском музейном центре, филиале Пермского краеведческого музея (г. Пермь).

DOI: 10.31857/S0132347421030092

### 1. ВВЕДЕНИЕ

Концепция т. н. умных музеев (англ. Smart Museum) [1] – закономерное следствие цифровизации институтов культуры. Суть этой концепции состоит во внедрении программируемой микроэлектроники в повседневные музейные практики. В рамках концепции умного музея параллельно решаются следующие задачи:

1. Мониторинг активности посетителей.
2. Организация навигации в музейном пространстве.
3. Организация интерактивных экспонатов.

Технологической базой для решения этих задач зачастую выступает Интернет вещей (англ. Internet of Things, IoT) [1]. Ключевой проблемой, при этом, оказывается высокий порог вхождения при освоении соответствующих программных инструментов и аппаратных средств ввиду их недостаточной высокоуровневости. Это заставляет музейных сотрудников обращаться к сторонним

ИТ-специалистам для развертывания специфических программно-аппаратных решений в заданных условиях. Актуальными оказываются вопросы автоматизации процессов создания таких решений и разработка высокоуровневого настраиваемого инструментария, который, в идеале, могли бы освоить люди без глубоких специальных знаний в области программирования и электроники.

Мониторинг активности посетителей и предоставление им электронных средств навигации актуальны в первую очередь для музеев, располагающих большими площадями, в то время как потребность в интерактивных экспонатах не зависит ни от размера, ни от тематики музея. В рамках решаемых в ходе данного исследования практических задач первоочередным было именно обеспечение интерактивности.

Важным требованием, предъявляемым к интерактивности экспонатов в рамках концепции

умного музея, является обеспечение естественности интерфейса и его привязка к музейному предмету. Взаимодействуя с экспонатом, посетитель должен получать некоторый новый опыт, не теряя при этом нити повествования той истории, которую рассказывает музейная экспозиция. С практической точки зрения это означает, что традиционные кнопочные интерфейсы далеко не всегда оказываются адекватны поставленной задаче, ведь они зачастую ассоциируются в сознании пользователя с традиционным человеко-машинным общением, ставшим для многих обыденностью. Это, в свою очередь, нарушает “погружение” посетителя в экспозицию, выводя его из нужного контекста, и ограничивает новый опыт, получаемый от физического посещения музея.

Проблему контекстуальной привязки, обеспечения естественности и нового опыта для посетителя может решить внедрение так называемых осязаемых пользовательских интерфейсов (англ. Tangible User Interface, TUI), переводящих интерактивные музейные экспонаты в плоскость кибер-физических систем (англ. Cyber-Physical System, CPS).

По аналогии с обобщенной кибер-физической системой, кибер-физический музейный экспонат состоит из специфического аппаратного пользовательского интерфейса (в общем случае обеспечивающего и прямую, и обратную связь с посетителем) и виртуального объекта, состояние которого управляется физическим интерфейсом (и влияет на состояние этого интерфейса при условии наличия обратной связи). При этом сам музейный предмет (экспонируемый на выставке артефакт, имеющий научное или культурное значение) может быть как физическим (фактически, частью осязаемого интерфейса), так и виртуальным (оцифрованной копией реального прототипа или оригинальной цифровой моделью какого-либо процесса/явления). В случае полностью виртуального артефакта, для его демонстрации посетителю адекватным является использование методов и средств научной визуализации. Научная визуализация, сочетая в себе когнитивную ясность и визуальную эстетику [2], позволяет точно передавать именно те черты экспоната, которые оказываются наиболее значимыми в контексте конкретной выставки.

Данная статья посвящена разработке программного инструментария, автоматизирующего построение и развертывание кибер-физических музейных экспонатов, основанных на технологиях Интернета вещей и научной визуализации. Практическим приложением этой работы стало создание нескольких музейных предметов для выставки “Превращения” в Детском музейном центре, филиале Пермского краеведческого музея, что позволило провести как теоретическое

исследование, так и практическую проверку предложенного подхода.

## 2. ТРЕБОВАНИЯ К КИБЕР-ФИЗИЧЕСКИМ МУЗЕЙНЫМ ЭКСПОНАТАМ

Эффективное использование кибер-физических систем в музейных выставках имеет определенную специфику, которая может быть выражена в следующих требованиях, предъявляемых к программно-аппаратному составу этих систем:

1. Быстрая и простая сборка (кибер-физические экспонаты должны состоять из интероперабельных модулей, не требующих заводских условий для сборки).

2. Быстрое и простое развертывание (для обеспечения мобильности выставок, кибер-физические экспонаты должны легко устанавливаться и демонтироваться).

3. Быстрое и простое включение/выключение (в идеале должно быть достаточно включения/выключения электроэнергии для запуска/остановки кибер-физических экспонатов).

4. Автономность (кибер-физические экспонаты не должны требовать постоянного обслуживания).

5. Интуитивный интерфейс (принцип человеко-машинного взаимодействия с кибер-физическими экспонатами должен быть понятен даже неподготовленному посетителю).

6. Низкая цена (в условиях зачастую ограниченного бюджета, выделяемого на формирование выставок, кибер-физические экспонаты должны состоять из максимально дешевых компонентов).

7. Заменяемые части (в случае поломки, должна быть возможность быстро найти и заменить вышедшие из строя части кибер-физических экспонатов).

8. Реконфигурируемость (при изменении выставки, должна быть возможность переиспользования модулей старых кибер-физических экспонатов для создания новых).

Для одновременного удовлетворения всех указанных требований мы предлагаем в качестве аппаратной основы кибер-физических экспонатов использовать либо микроконтроллерные системы, либо системы на кристалле (англ. System on Chip, SoC). Выбор конкретного аппаратного обеспечения зависит от специфики демонстрации кибер-физического экспоната.

## 3. МУЗЕЙНАЯ ПРАКТИКА ПРИМЕНЕНИЯ КИБЕР-ФИЗИЧЕСКИХ ЭКСПОНАТОВ

Микроконтроллерные системы целесообразно использовать тогда, когда в цифровой инфраструктуре музея для данного экспоната уже име-

ются средства визуализации, такие, например, как интерактивный киоск, или соединенный с монитором компьютер. Кроме того, микроконтроллерная система может использовать для визуализации мобильные устройства посетителя. Эти случаи были в подробностях рассмотрены в наших предыдущих работах [3, 4].

Если же в наличии нет подходящей рабочей станции, пригодной для визуализации виртуальной составляющей экспоната, наиболее экономически выгодным вариантом является построение кибер-физического экспоната на основе SoC (например, Raspberry Pi или Orange Pi). Выбор аппаратуры в пользу SoC определяет в данном случае их невысокая цена, сочетающаяся, при этом, с производительностью, достаточной для обеспечения качественной научной визуализации. Дополнительными важными достоинствами SoC являются их компактность и низкое энергопотребление, что позволяет осуществлять их монтаж даже в условиях очень ограниченного пространства.

В данной работе мы рассматриваем случай построения кибер-физических экспонатов с использованием SoC. В настоящее время SoC активно используются как в научных, так и в культурологических проектах. Например, популярная библиотека научной визуализации VTK успешно портирована на Raspberry Pi [5]. Существуют примеры использования Raspberry Pi как основы приборов для научных исследований [6], а также в качестве центрального компонента интерактивных музейных экспонатов с осязаемым интерфейсом, ориентированных на широкую аудиторию посетителей, включающую людей с ограниченными возможностями зрения [7]. Кроме того, Raspberry Pi широко применяется в робототехнике, в частности, для создания автономных роботов-экскурсоводов [8].

Однако, во всех вышеописанных примерах, использование SoC сопряжено с необходимостью низкоуровневого программирования управляющих модулей и взаимодействий, что делает процесс разработки конечных продуктов трудоемким и осуществимым лишь с участием высококвалифицированных профессионалов в области информационных технологий. Таким образом, вопрос создания высокоуровневого инструментария для работы с SoC остается весьма актуальным, особенно в контексте внедрения таких систем в музейную практику.

#### 4. АВТОМАТИЗИРОВАННАЯ ГЕНЕРАЦИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ СИСТЕМ НА КРИСТАЛЛЕ

В качестве основы высокоуровневого инструментария для создания кибер-физических музейных экспонатов из SoC предлагается использо-

вать платформу визуального проектирования программно-аппаратных систем научной визуализации SciVi (<https://scivi.tools/>) [9]. Данная платформа может служить генератором программного обеспечения для интерактивного рендеринга, в том числе поддерживающего взаимодействие с аппаратными человеко-машинными интерфейсами [10]. Ключевой особенностью SciVi является ее гибкость в отношении адаптации к разнообразным сценариям работы. Эта гибкость достигается использованием онтологий как основы функционирования платформы: поведение SciVi полностью декларируется лежащей в ее основе онтологической базой знаний (БЗ), описывающей поддерживаемые форматы и типы данных, алгоритмы их обработки и визуализации, способы интерактивного взаимодействия с пользователем, особенности программно-аппаратного окружения, протоколы взаимодействия с другим программным обеспечением и т.п.

Одной из базовых функций SciVi является генерация программного обеспечения для интерактивной визуализации научных данных. Именно эта функция и может быть использована при создании кибер-физических музейных экспонатов. В рамках наших предыдущих исследований были разработаны методы и средства для генерации программного обеспечения кибер-физических музейных экспонатов, функционирующих на базе микроконтроллеров, и использующих для визуализации внешнюю аппаратуру (киоски, оснащенные компьютером и монитором, или мобильные устройства посетителей, подключенные к внутримузейной беспроводной компьютерной сети) [3, 4]. В данной работе мы обобщаем этот подход до поддержки SoC, что позволяет сделать кибер-физические экспонаты полностью автономными, не зависящими ни от внешнего оборудования, ни от цифровой инфраструктуры музея.

##### *4.1. Высокоуровневая настройка посредством диаграмм потоков данных*

Настройка платформы SciVi на решение конкретной задачи интерактивной визуализации осуществляется посредством диаграмм потоков данных (англ. Data Flow Diagram, DFD). В этом контексте DFD выступают визуальным языком программирования, позволяющим наглядно, при помощи высокоуровневых операторов (блоков DFD), связанных по данным, описать требуемые алгоритмы отображения виртуального музейного экспоната и интерактивного взаимодействия с ним. При этом пользователю предоставляется соответствующий графический интерфейс, обеспечивающий возможность строить DFD путем добавления операторов из заранее заготовленной палитры. Подобный способ записи алгоритмов хорошо зарекомендовал себя на практике в раз-

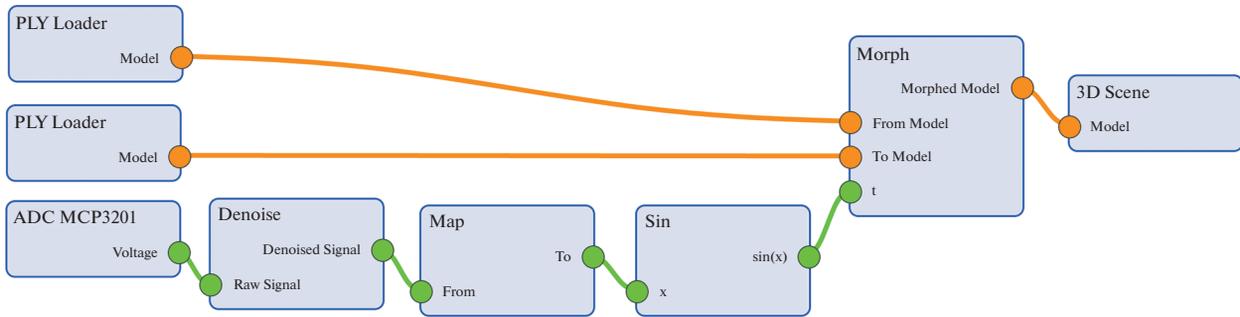


Рис. 1. DFD в среде платформы SciVi, описывающая работу кибер-физического музейного экспоната.

личных системах обработки и визуализации данных, например, KNIME [11] или Blender 3D [12].

Ключевым отличием SciVi от других систем, использующих DFD, является подход к формированию палитры операторов, доступных пользователю: в SciVi эта палитра формируется автоматически на основе онтологической БЗ о поддерживаемых аппаратных элементах, для которых платформа может генерировать прошивку, и программных методах обработки/визуализации данных. Таким образом, палитра операторов DFD может быть пополнена или изменена без модификации программного кода SciVi, лишь путем изменения БЗ.

DFD, описывающая работу одного из кибер-физических экспонатов, созданных в рамках данного исследования, представлена на рис. 1.

Этот экспонат посвящен наглядной демонстрации экогеографического правила Аллена [13] о том, что теплокровные животные в полярных широтах имеют более округлую форму тела, чем представители тех же семейств, живущие ближе к экватору. Это связано с необходимостью минимизировать площадь поверхности тела для сокращения потерь тепла в более холодном климате. Корпус экспоната оформлен в виде схематичной карты Земли, на которой при помощи рычага посетитель может выбрать интересующую широту. Положение рычага детектируется соединенным с ним потенциометром, напряжение на котором измеряется при помощи аналогово-цифрового преобразователя (АЦП) MCP3201, соединенного с SoC через шину SPI.

За захват данных с АЦП на DFD отвечает оператор “ADC MCP3201”. Полученные данные подвергаются фильтрации с целью подавления возможных помех (фильтр “Denoise”, использующий метод скользящего среднего), затем отображаются на диапазон  $[0; \pi]$  (оператор “Map”) и преобразуются функцией синуса (оператор “Sin”) в плавную кривую  $[0; 1]$ ,  $[1; 0]$ . Кривая определяет изменение параметра  $t$  у оператора “Morph”, который осуществляет по этому параметру линейную интерполяцию формы двух 3D-моделей, загружае-

мых операторами “PLY Loader” из файлов формата PLY. Модели были созданы специально для данного экспоната в программе Blender 3D, состоят примерно из 10 000 полигонов каждая, имеют одинаковую топологию и представляют собой “полярную” (песец) и “экваториальную” (корсак) формы “среднестатистического” млекопитающего, мелкого представителя семейства псовых (лат. canidae). Результаты рендеринга этих моделей средствами платформы SciVi (с применением псевдоглобального освещения, имитируемого методом matcap [14]) представлены на рис. 2.

В соответствии с приведенной DFD, когда рычаг в составе экспоната указывает на северный или южный полюс на карте,  $t = 0$  и отображается “полярная” форма животного. На экваторе, в свою очередь,  $t = 1$ , и отображается “экваториальная форма”. В промежуточных широтах отображаемая форма принимает вид, определяемый линейной интерполяцией граничных форм. Следует отметить, что результат линейной интерполяции форм при  $t = 0.5$  (средняя широтная полоса) внешне напоминает европейскую лисицу. Это подтверждает корректность использованной математической модели с учетом принятого уровня художественной стилизации.

#### 4.2. Управление посредством онтологий

Каждому оператору DFD соответствует фрагмент онтологии из БЗ SciVi. Пример такого фрагмента, описывающего оператор загрузки 3D-модели “PLY Loader”, приведен на рис. 3. Онтологии создаются и редактируются при помощи высокоуровневого графического редактора ONTOLIS [15].

“PLY Loader” в терминах системы SciVi является источником данных (“DataSource”), имеет настройку (“Setting”) “PLY File” типа “файл” (“File”) и выходной сокет (“Output”) “Model” типа “3D-модель” (“3Dmodel”). Для генерации кода важна реализация этого оператора, выраженная на онтологии вершиной “PLY Parser”. Эта реализация выполнена на языке C++, предназначена для исполнения на стороне клиента (“ClientSideWorker” –

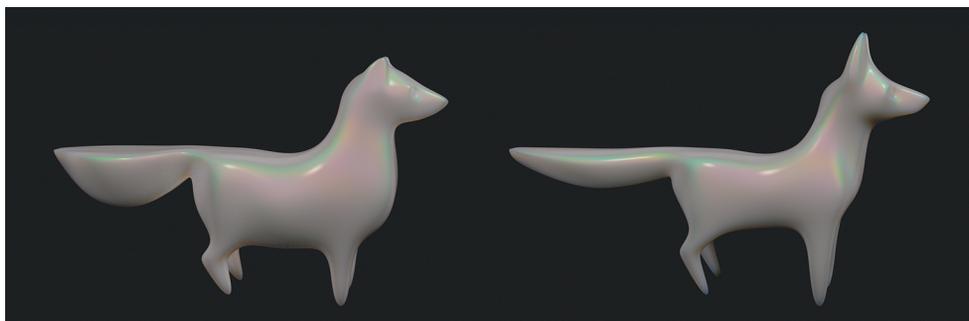


Рис. 2. Результат визуализации 3D-моделей “полярной” (слева) и “экваториальной” (справа) формы млекопитающего для демонстрации правила Аллена.

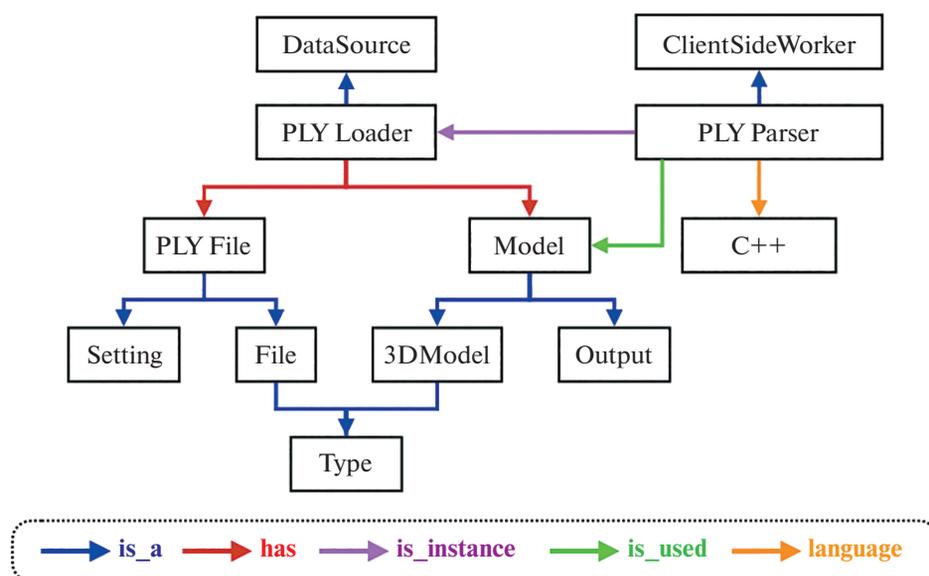


Рис. 3. Фрагмент онтологии из БЗ SciVi, описывающий оператор загрузки 3D-модели.

в данном случае это означает локальное исполнение без привлечения сетевых ресурсов) и используется для означивания выходной модели (“Model”). Ссылка на код реализации оператора содержится в атрибуте “path” вершины “PLY Parser” (сервисные атрибуты вершин на приведенной иллюстрации не обозначены в целях экономии места).

Вместе с реализацией для некоторых операторов хранятся также сведения об особых параметрах компиляции, компоновки и настройки операционной системы (ОС). Например, для оператора “ADC MCP3201” помимо исходного кода онтологией описывается еще и ряд настроек ОС, связанных со включением шины SPI, по которой к SoC подключается соответствующий внешний модуль АЦП.

Генератор кода в составе SciVi автоматически обходит онтологию операторов в соответствии с заданной DFD и собирает итоговую программу для целевой SoC из найденных реализаций, а так-

же необходимые скрипты для компиляции и установки. Таким образом, онтология операторов фактически выступает семантическим индексом для репозитория реализаций этих операторов.

#### 4.3. Стек используемых технологий

Стек основных технических и программных средств, на основе которых функционирует генерируемое программное обеспечение (ПО), представлен на рис. 4.

“SciVi Renderer” представляет собой основное приложение научной визуализации, отвечающее за отображение виртуальной части кибер-физического экспоната. Исходный код этого приложения автоматически генерируется платформой SciVi на языке C++. Оно использует графический API OpenGL ES (доступный на большинстве SoC через библиотеку Mesa) для рендеринга и различные библиотеки доступа к периферийному оборудованию

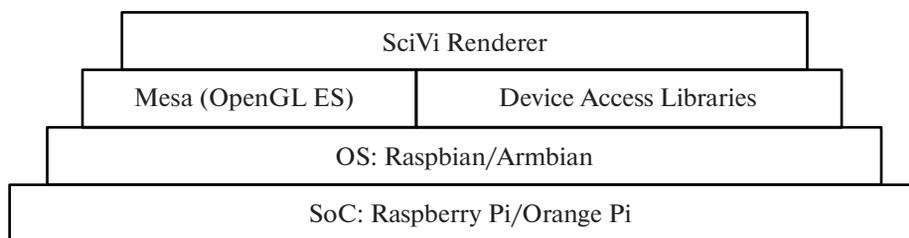


Рис. 4. Стек основных программно-аппаратных средств в составе кибер-физического музейного экспоната.

(“Device Access Libraries”) для взаимодействия с аппаратной частью TUI. Целевыми операционными системами (ОС), под управлением которых затем будет работать сгенерированное ПО, выступают Raspbian (для SoC Raspberry Pi) и Armbian (для SoC Orange Pi). При необходимости список поддерживаемых SoC и ОС легко можно расширить, дополнив БЗ SciVi соответствующими описаниями. В контексте создания кибер-физических музейных экспонатов предпочтение отдается Orange Pi, так как эта модель одноплатных компьютеров, с одной стороны, обладает достаточно высокой производительностью, но с другой — отличается низкой ценой.

В качестве экрана для отображения виртуальной составляющей экспоната может быть использован любой доступный монитор или проектор. В нашем случае был использован 7-дюймовый HDMI-совместимый мини-монитор разрешением  $1024 \times 600$  пикселей. Производительность рендеринга сгенерированным ПО требуемых 3D-моделей на Orange Pi PC2 для данного разрешения составляет 16 кадров в секунду (англ. Frames per Second, FPS), что достаточно для восприятия плавных движений.

Компиляция и развертывание сгенерированного ПО осуществляется классической последовательностью команд `configure`, `make`, `sudo make install`. Установка осуществляется с правами суперпользователя, так как в общем случае включает в себя ряд настроек ОС, необходимых для корректного функционирования целевого ПО. К таким настройкам относится, например, автоматический вход в ОС и запуск необходимого ПО при старте SoC. Таким образом, после однократной процедуры развертывания, SoC становится автономно функционирующим кибер-физическим экспонатом, не требующим дополнительной настройки.

#### 4.4. Соответствие заявленным требованиям

Кибер-физические экспонаты на основе SoC, программная часть которых генерируется платформой SciVi, соответствуют всем требованиям, отмеченным в разделе 1:

1. Простота и скорость сборки обусловлена модульностью SoC и их периферийных устройств, а также модульным характером DFD, при помощи которых осуществляется программная настройка.

2. Простота и скорость развертывания обеспечена автоматической генерацией платформой SciVi всего необходимого ПО и скриптов для его установки.

3. Простота включения обеспечивается автозапуском всего необходимого ПО, что позволяет включать SoC лишь подачей электропитания, без необходимости совершения каких-либо ручных настроек на старте. Простота выключения обеспечивается установкой режима “только чтение” для основной файловой системы, что позволяет отключать SoC путем простого выключения электропитания без риска повреждения контента карты памяти. Обе настройки записаны в автоматически генерируемый скрипт установки.

4. Автономность обеспечивается тем, что сгенерированное ПО включает в себя все необходимые настройки ОС SoC и периферийных устройств.

5. Интуитивность интерфейса — вопрос, скорее относящийся к дизайну экспоната. Автоматический контроль этого параметра никак не учтен в платформе SciVi, однако совместимость SciVi с TUI открывает широкие возможности для обеспечения интуитивности интерфейса и привязки его к контексту конкретной музейной выставки.

6. Низкая стоимость обеспечивается тем, что SoC и совместимые с ними периферийные устройства (сенсоры, актуаторы) достаточно дешевы, а платформа SciVi является OpenSource-проектом и может быть использована бесплатно. Так, например, стоимость одного кибер-физического экспоната для выставки “Превращения” составила около \$130.

7. Элементная база SoC может быть подобрана практически в любом магазине радиоэлектроники.

8. Реконфигурируемость обеспечивается возможностью описывать алгоритм работы при помощи DFD и автоматически генерировать код итогового ПО в среде платформы SciVi.

## 5. ЗАКЛЮЧЕНИЕ

Проведенное исследование было направлено на разработку методов и средств автоматизации создания кибер-физических музейных экспонатов. Ключевая идея состоит в использовании платформы SciVi, позволяющей по высокоуровневым описаниям автоматически генерировать ПО систем научной визуализации на кристалле, то есть ПО, обеспечивающее научно-точный рендеринг данных на аппаратном обеспечении SoC. Предложенные методы и средства прошли успешную практическую проверку при разработке выставки “Превращения” в Детском музейном центре, филиале Пермского краеведческого музея (г. Пермь). Ввиду малых размеров SoC и мониторов итоговые размеры экспонатов оказались небольшими, что помогло оптимально использовать маленькие экспозиционные площади.

Производительность рендеринга сгенерированного ПО на целевой SoC (Orange Pi PC2 под управлением ОС Armbian) на реальных данных составляет 16 FPS, что, в целом, удовлетворяет требованиям, предъявляемым к интерактивному музейному экспонату, однако тем не менее является сравнительно низкой. Для решения этой проблемы в дальнейшем планируется изучить вопрос возможной оптимизации алгоритмов рендеринга с учетом аппаратной специфики SoC. Кроме того, планируется организовать поддержку кросс-компиляции ПО для SoC на стороне платформы SciVi так, чтобы для развертывания программной части кибер-физического экспоната достаточно было только установить подготовленный в SciVi бинарный пакет, не производя компиляцию и компоновку программ на стороне целевого устройства.

## 6. БЛАГОДАРНОСТИ

Выражаем благодарность С.Л. Островскому и У.Ш. Сайрановой за предоставленную нам возможность участия в создании выставки “Превращения”, А.С. Кошелеву за помощь в создании корпусов кибер-физических экспонатов.

## СПИСОК ЛИТЕРАТУРЫ

1. *Chianese A., Piccialli F.* Designing a Smart Museum: when Cultural Heritage Joins IoT // Third International Conference on Technologies and Applications for Smart Cities (I-TASC'14). IEEE. 2014. 7 p. <https://doi.org/10.1109/NGMAST.2014.21>.
2. *Manakov D.V.* Visual Analytics and Data Abstraction Models // Proceedings of 28th International Conference on Computer Graphics and Vision “GraphiCon 2018”. Tomsk, 2018. P. 146–150.
3. *Ryabinin K.V., Kolesnik M.A.* Adaptive Scientific Visualization Tools for a Smart Paleontological Museum // Programming and Computer Software. Pleiades Publishing, Ltd. 2019. V. 45. № 4. P. 180–186. <https://doi.org/10.1134/S0361768819040066>
4. *Ryabinin K.V., Kolesnik M.A., Akhtamzyan A.I., Sudarikova E.V.* Cyber-Physical Museum Exhibits Based on Additive Technologies, Tangible Interfaces and Scientific Visualization // Scientific Visualization. National Research Nuclear University “MEPhI”. 2019. Q. 3. V. 11. № 4. P. 27–42. <https://doi.org/10.26583/sv.11.4.03>
5. *Ibanez L.* Raspberry Pi likes VTK [Электронный ресурс]. Kitware, 2012. <https://blog.kitware.com/raspberry-pi-likes-vtk/>.
6. *Buscher N., Ojeda A., Francoeur M., Hulyalkar S., Claros C., Tang T., Terry A., Gupta A., Fakhraei L., Ramanathan D.S.* Open-Source Raspberry Pi-Based Operant Box for Translational Behavioral Testing in Rodents // Journal of Neuroscience Methods. Elsevier, 2020. V. 342. <https://doi.org/10.1016/j.jneumeth.2020.108761>
7. *Rossetti V., Furfari F., Leporini B., Pelagatti S., Quarta A.* Enabling Access to Cultural Heritage for the Visually Impaired: an Interactive 3D Model of a Cultural Site // Procedia Computer Science. Elsevier, 2018. V. 130. P. 383–391. <https://doi.org/10.1016/j.procs.2018.04.057>
8. *Diallo A.D., Gobe S., Durairajah V.* Autonomous Tour Guide Robot Using Embedded System Control // Procedia Computer Science. Elsevier, 2015. V. 76. P. 126–133. <https://doi.org/10.1016/j.procs.2015.12.302>
9. *Ryabinin K.V., Chuprina S.I.* Using Scientific Visualization Systems to Automate Monitoring of Data Generated by Lightweight Programmable Electronic Devices // Programming and Computer Software. Pleiades Publishing, 2018. V. 44. № 4. P. 278–285. <https://doi.org/10.1134/S0361768818040102>
10. *Ryabinin K., Chuprina S., Belousov K.* Ontology-Driven Automation of IoT-Based Human-Machine Interfaces Development // Lecture Notes in Computer Science. Springer, 2019. V. 11540. P. 110–124. [https://doi.org/10.1007/978-3-030-22750-0\\_9](https://doi.org/10.1007/978-3-030-22750-0_9)
11. KNIME [Электронный ресурс]. <https://www.knime.com>.
12. Blender 3D [Электронный ресурс]. <https://www.blender.org>.
13. *Allen J.A.* The Influence of Physical Conditions in the Genesis of Species // Radical Review. 1877. V. 1. P. 108–140.
14. *Barros C.* MatCap – Render & Art Pipeline Optimization for Mobile Devices [Электронный ресурс]. 2019. <https://medium.com/playkids-tech-blog/matcap-render-art-pipeline-optimization-for-mobile-devices-4e1a520b9f1a>.
15. *Chuprina S., Nasraoui O.* Using Ontology-Based Adaptable Scientific Visualization and Cognitive Graphics Tools to Transform Traditional Information Systems into Intelligent Systems // Scientific Visualization. National Research Nuclear University “MEPhI”. 2016. Q. 1. V. 8. № 1. P. 23–44.