

# ТЕХНОЛОГИИ ДЛЯ СОЗДАНИЯ «УМНЫХ» МАШИН В ЭПОХУ IIoT

ГРЕГ БРАУН (GREG BROWN)

Современные технологии, используемые для разработки распределенного и подключаемого оборудования с возможностью функционирования в режиме реального времени, включают в себя гетерогенную архитектуру обработки данных, аппаратные средства, инструменты для проектирования их программного обеспечения, а также готовность разработанного оборудования функционировать в среде промышленного «Интернета вещей» (IIoT).

Что значит сделать что-то «умным»? Какие это дает преимущества при разработке оборудования для функционирования в экосистеме промышленного «Интернета вещей» (англ. Industrial Internet of Things, IIoT)?

Возможно, понятие «умные» означает способность машин «понять» все то, что разработчик может для них придумать. Такое оборудование имеет высокую, близкую к максимальной, точность, защитные устройства в виде нескольких типов датчиков, новый алгоритм управления или заложенное прогнозирование их обслуживания. А что насчет движения машин или исполнительных механизмов оборудования, направляемого машинным зрением? Или они «умны», так как в них предусмотрено использование нескольких протоколов связи и обмена данными? Является их «разумность» локальной или она имеет в своей основе распределенный характер? Где и как осуществляется их обучение? Означает ли термин «умные» то, что эти машины или оборудование работают в сфере, определенной такими понятиями, как «Индустрия 4.0» (англ. Industry 4.0), IIoT, Made in China 2025 (программа, предусматривающая достижение к 2025 г. полной технологической независимости КНР — от собственного производства чипов до выпуска беспилотных автомобилей), Made in India (программа обеспечения технологического прорыва Индии) и ряд других? И, в конце концов, должны ли такие машины или оборудование сами создавать новые бизнес-модели?

## **ГЕТЕРОГЕННАЯ (НЕОДНОРОДНАЯ) АРХИТЕКТУРА ОБРАБОТКИ ДАНЫХ**

В современной «умной» машине для обработки многоосных алгоритмов движения и машинного зрения необходим быстрый и высокопроизводительный процессор. Сегодня с этим связана достаточно серьезная и насущная проблема: наиболее быстродействующие процессоры были разработаны для рабочих нагрузок серверного типа, и их сложные конвейеры, кэши и т. п. были специально оптимизированы для обеспечения максимальной пропускной способности, а не детерминистических (обусловленных текущей ситуацией) ответов в режиме реального времени. Для многоядерных (до четырех ядер) и специализированных мультиядерных (с числом ядер более четырех) процессоров именно повышение производительности является первоочередной задачей.

Для того чтобы решить указанную проблему и воспользоваться имеющимися в нашем распоряжении процессорами, их задачи должны быть разделены на параллельные операции управления. В свою очередь, операционные системы (ОС) и библиотеки программного обеспечения должны в режиме реального времени обеспечивать безопасность потоков. Это позволит уменьшить нагрузку и упростить сложность программирования, сопровождающих разработку многопоточных приложений на базе многоядерных и специализированных многоядер-

ных процессоров. При переходе к среднечастотному диапазону производительности процессора появляется возможность для увеличения частоты и, следовательно, производительности отдельного потока, что уменьшает площадь кристалла процессора и снижает уровень его собственного энергопотребления.

В современных «умных» машинах с высокоскоростными портами ввода-вывода (I/O) и необходимостью иметь строго детерминированный отклик в режиме реального времени (причем лежащего в субмикросекундном диапазоне) даже самые быстрые процессоры не могут справиться со всеми требованиями по производительности. Решить эту проблему поможет использование гетерогенной (неоднородной) архитектуры обработки данных.

## **ПЯТЬ ПРИМЕРОВ АРХИТЕКТУРЫ**

Неоднородная архитектура позволяет пользоваться различными механизмами обработки данных для оптимизации целого ряда аспектов, связанных с управлением интеллектуальными машинами, а также дает дополнительные преимущества разработчику таких «умных» машин. Примеры гетерогенных архитектур:

- процессоры с цифровыми сигнальными процессорами (DSP);
- процессоры с графическим процессором общего назначения (GPGPU). В этом случае графический процессор может более эффективно использоваться для визуализации предметов в машинной графике, поскольку конечный

пользователь может запрограммировать его на выполнение алгоритмической обработки;

- центральные процессоры (CPU), цифровые сигнальные процессоры (DSP) и графические процессоры общего назначения (GPGPU);
- процессоры с программируемыми пользователем вентильными матрицами (FPGA);
- IP-блоки для конкретного приложения, реализованные на базе любого варианта из перечисленных выше.

Все эти примеры базовой архитектуры могут существовать в дискретных компонентах или быть интегрированы в систему-на-кристалле (англ. system-on-chip, SoC). На некоторых устройствах можно реализовать дополнительные IP-блоки, предназначенные для использования в приложениях, которые относятся к таким машинным сборкам, как устройства DSP с ШИМ-модулем (англ. Pulse Width Modulation module, PWM module). В частности, в последнее время популярной гетерогенной архитектурой стали процессоры с FPGA. Они предоставляют пользователю сразу три элемента обработки, поскольку поставщики устройств FPGA интегрировали в них мощные блоки DSP. FPGA как раз предназначены для обеспечения субмикросекундного уровня обработки данных, аппаратного детерминизма и высокой надежности. Кроме того, они характеризуются еще и высокой гибкостью, возможностью выполнения полной настройки под конкретные индивидуальные требования потребителя, обновления их прошивки непосредственно на местах эксплуатации и исправления ошибок без замены аппаратных средств.

**СТОИМОСТЬ, ПРОИЗВОДИТЕЛЬНОСТЬ И ЭНЕРГОПОТРЕБЛЕНИЕ**

Для того чтобы адаптировать конечную реализацию к потребностям машины, пользователи могут выбрать необходимые параметры производительности, стоимости, габаритов, энергопотребления, а также количество портов ввода-вывода для FPGA и процессоров. Кроме того, у них есть возможность разработать масштабируемую аппаратную платформу, которая будет использовать

общее программное обеспечение для процессоров и IP-блоков в FPGA. С этой целью несколько лет назад были разработаны SoC, интегрирующие процессоры с FPGA, имеющими блоки DSP. Рис. 1 иллюстрирует упрощенные версии основных гетерогенных архитектур.

Напрашивается вопрос: что же означает понятие «лучшая архитектура»? Ответ прост: лучшая архитектура — та, которая лучше всего подходит для решения по технологиям, отвечает пожеланиям клиентов и бизнес-требованиям. Она зависит от конкретной ситуации и конкретного приложения. Один из главных принципов выбора заключается в том, чтобы сосредоточиться на архитектуре, которая обеспечит не сиюминутный выигрыш, а долгосрочную выгоду и поможет решать задачи управления несколькими поколениями машин и оборудования. Инвестиции в более простую архитектуру могут дать существенные разовые выигрыши, но ее частые изменения приведут к напрасным тратам значительных средств.

В тестах на оценку производительности редко сравниваются рассматриваемые здесь архитектуры из-за сложности такой процедуры. Чаще всего тесты проводятся для конкретных CPU (с этой целью используются, например, инструменты CoreMark и SPECint) или для FPGA в условиях конкретно заданных функций. Проблема в том, что для таких исследований необходимо иметь рассчитанную на рабочую нагрузку инфраструктуру с последовательной методологией сравнения соответствующих показателей в форме, ориентированной на конечные приложения. Одними из немногих, кто пытался протести-

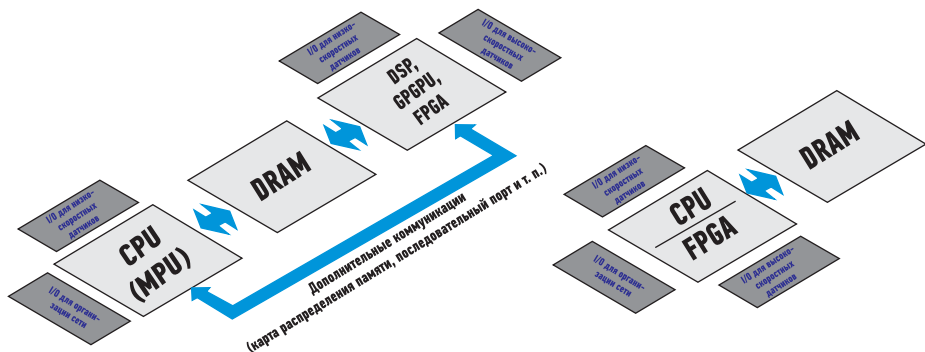
ровать гетерогенные архитектуры, являются сотрудники Исследовательского центра национального научного фонда США (NSF Center for High-Performance Reconfigurable Computing). Для того чтобы иметь возможность выполнять эквивалентный анализ по типу сравнения «яблока с яблоком» (английская идиома для сравнения сопоставимых понятий, величин, показателей, вещей и т. д.) для разных реализаций, исследователям удалось создать специальную инфраструктуру. Для тестирования приложения они использовали несколько критических рабочих нагрузок: дистанционный сбор данных, обработку изображений, управление движением машины или инструментом оборудования, генерацию траекторий и коммуникации.

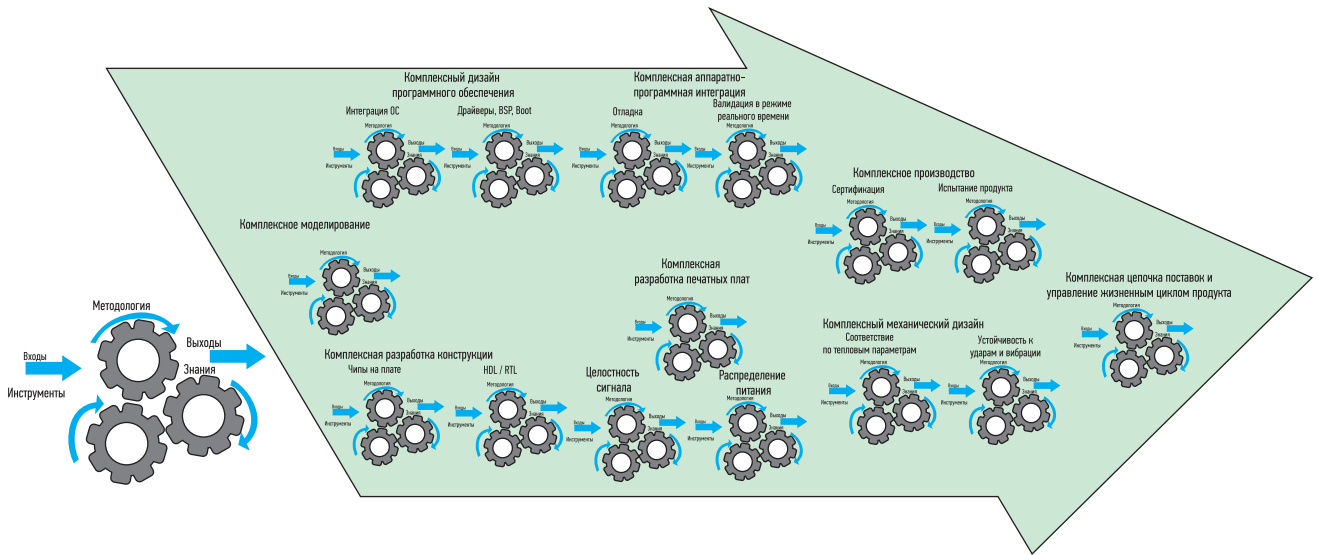
**АППАРАТНОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ: РАЗРАБОТКА ПРОЕКТА В ЦЕЛОМ**

Основной проблемой для эффективного использования любой гетерогенной архитектуры является сложность ее проектирования в процессе разработки как аппаратного, так и программного обеспечения (рис. 2). Выбор отдельных и не интегрированных между собой инструментов может затруднить управление рабочими процессами и данными, что создает определенные риски. Часто больше внимания уделяется созданию приложений для рабочего оборудования и стека программных средств оперативного управления производством, чем непосредственно разработке самой конструкции.

Проектирование готовой к применению и настраиваемой под кон-

**РИС. 1.** ▼ Упрощенные версии основных гетерогенных архитектур





**РИС. 2. ▲**  
Разработка и развертывание настраиваемого под конкретные требования аппаратного и программного решения сочетают в себе целый комплекс основных и вспомогательных процессов

кретные задачи пользователя аппаратно-программной системы для нужд современного управления производством требует наличия большого количества различных инструментов, процессов и методологий. По мере совершенствования процессорных технологий, которые подчиняются закону Мура, не только увеличилась сложность процессоров, но и появилось множество проблем и рисков для заказного встраиваемого оборудования высокопроизводительных систем. Также выросли затраты на инструменты и требуемые знания в области конструирования таких систем и в ряде смежных областей.

Если скорости полупроводниковых устройств увеличились в их ядре, то скорости портов ввода-вывода, где целостность сигнала является проблемой уже на уровне платы, часто требуют специальных инструментов и полного понимания процесса. Современные корпуса с более чем 1000 выводами с малым расстоянием между ними (от 0,5 до 1 мм), нуждающиеся в максимально близком размещении развязывающих или блокировочных конденсаторов, совместно с целым рядом конструктивных проблем, делают необходимым применение высококачественных печатных плат с 10–16 слоями. Все это, в свою очередь, затруд-

няет не только проектирование, но и изготовление и сертификацию устройств, выполненных на таких компонентах. Также выросло количество шин питания, усложнилось управление им, и непростыми задачами, соответственно, стали выбор источника питания и раздача его по шинам.

Что касается выбора ОС, все более популярной становится Linux с модифицированным ядром для работы в режиме реального времени, которая имеет очень широкие возможности для обработки данных и работы с объемами памяти от мегабайт до гигабайт, что позволяет уменьшить стоимость конечного решения. Также немаловажными факторами являются возросшая необходимость в расширении применения сетевых решений и желание проектировщиков заниматься разработкой собственного программного обеспечения. Использование своей версии ОС с разработкой драйвера и надежной аппаратной или программной валидацией<sup>1</sup> в гетерогенной архитектуре требует дополнительного набора инструментов и соответствующего опыта. Еще одно преимущество ОС Linux с модифицированным ядром при работе в режиме реального времени заключается в том, что уже существуют организации и проекты, такие как Linux Foundation и Yocto Project, которые занимают-ся стандартизацией и упрощением ее внедрения.

## ОБЕСПЕЧЕНИЕ ВЫСОКОЙ УСТОЙЧИВОСТИ КОНСТРУКЦИИ К ВОЗДЕЙСТВИЯМ ОКРУЖАЮЩЕЙ СРЕДЫ И ДОЛГОВЕЧНОЙ ЭКСПЛУАТАЦИИ

Чтобы спроектированная конечная конструкция была устойчивой к внешним воздействиям и долгое время сохраняла свою работоспособность в непростых условиях индустриальной среды, для нее необходимо разработать дополнительные защищающие конструктивные элементы: защитную оболочку, корпус, элементы крепления и т. п. Кроме того, требуется провести тестирование устройства и проверить его энергопотребление, функционирование во всем диапазоне рабочих температур, устойчивость к ударам и вибрации, а также еще целый ряд параметров, характерных для конкретных областей его применения. Например, тепловое моделирование может указывать на горячие точки, которые создадут в будущем проблемы с надежностью, если не будут надлежащим образом ликвидированы. Выбор компонентов, конструкция и компоновка платы, организация управления тепловыми потоками, т. е. использование теплоотводов (радиаторов) и принудительного воздушного охлаждения, — все это взаимосвязано и должно быть учтено. Кроме того, устройства должны пройти соответствующую сертифи-

<sup>1</sup>Имеется в виду подтверждение соответствия требованиям путем испытаний и представления объективных свидетельств, выполнения конкретных требований к предусмотренному конкретному использованию, см. ГОСТ Р МЭК 61508-4-2012. — Прим.переводчика

кацию, которая зависит от области назначения и от рабочей среды, в которой планируется его последующее функционирование.

Свое влияние оказывает и сложная цепочка поставок компонентов для устройств и необходимых программных инструментов от различных производителей. Для большинства специалистов, выполняющих разработку образцов оригинального оборудования для его серийных производителей (ОЕМ), объемы выпуска изделий не являются массовыми, они укладываются в категории от мелкосерийного производства (от десятков до сотен штук) до среднесерийного производства с выпуском от сотен до тысяч штук устройств. Поэтому подавляющая часть поставщиков не обеспечивает прямую поддержку или обслуживание. Таким образом, многие OEM-производители оказываются буквально брошенными на произвол судьбы, что еще больше усложняет применение новейших технологий, особенно для промышленных компонентов с долгосрочной (более 10 лет) доступностью.

Одним из вариантов решения этой проблемы является использование компонентов от проверенного поставщика с подтвержденной репутацией и хорошей историей поставок в данной области промышленности. Также важен предоставляемый с продуктом набор программного обеспечения. Обязательно убедитесь в том, что перед вами продукты с открытым исходным кодом: это позволит при необходимости быстро внести изменения и подобрать соответствующие настройки. Кроме того, необходимо знать, является ли готовое к развертыванию исполняемое программное обеспечение основной компетенцией поставщика. Выясните, было ли оно проверено, имеет ли валидацию и подтверждено ли должное функционирование программного обеспечения на практике в условиях его многократного применения. Вы должны понимать, что существует различие между предоставляемым программным обеспечением и программным обеспечением, необходимым для развертывания системы, — это важно для правильного прогнозирования реального времени, связанного с дополнительными работами по проектированию системы, ее соответствующего тестирования и подтверждения соответствия.

## СРЕДЫ РАЗРАБОТКИ

Сегодня достигнут значительный прогресс в разработке языков проектирования и инструментов более высокого уровня, которые абстрагируют некоторые или все сложности, присущие гетерогенным архитектурам. Благодаря этому основное внимание может уделяться разработке непосредственно дифференцированных приложений, а не детальному проектированию низкого уровня.

В связи с такими достижениями в области моделирования, как языки программирования и инструмент синтеза более высокого уровня, а также графическое проектирование систем, появился большой выбор различных инструментов и методологий. Некоторые из них нацелены на выделение главных признаков в сложностях, связанных с разработкой много- и мультядерных процессоров, в то время как остальные — на аппаратные ускорители или IP-ядра для завершения гетерогенных систем / подсистем, включая их порты входа/выхода. Чем полнее абстракция и чем шире набор инструментов, тем лучше такие инструменты должны понимать конечную цель проектирования.

Что касается инструментов, генерирующих аппаратное обеспечение, большинство из них может создать интерфейс HDL IP для целевого оборудования, но, как правило, это будет FPGA, требующая экспертных знаний о разработке HDL-решений и их реализации (о подробных ограничениях по времени, переключателях инструментов и т. д.) для завершения проектирования. Поэтому лишь немногие способны взять на себя весь объем работ от ввода проекта в эксплуатацию до организации полного рабочего процесса.

## ГОТОВНОСТЬ К ПРОМЫШЛЕННОМУ «ИНТЕРНЕТУ ВЕЩЕЙ»

Поскольку сейчас интенсивно эволюционируют IIoT, «Индустрия 4.0», Made in China 2025, Make in India и другие направления развития техники, современные интеллектуальные машины должны уметь взаимодействовать друг с другом, с другим оборудованием, технологиями операций, информационными технологиями и с облаком. Важно понимать, как все это развивается, знать

возможности новых разработок и разбираться в таких вопросах, как архитектура, методология разработки, и в том, какую роль в этом играют инструменты. Рассмотрим суть этой задачи на примере IIoT.

Для того чтобы современная «умная» машина могла считаться IIoT-совместимой (IIoT-capable) или готовой к реализации технологии IIoT (IIoT-ready), в ней должны присутствовать четыре основных элемента:

1. Множественные типы портов ввода/вывода и их расширяемость.
2. Возможность вычисления в режиме реального времени.
3. Возможность обмена данными в режиме реального времени.
4. Расширяемость и гибкость системы.

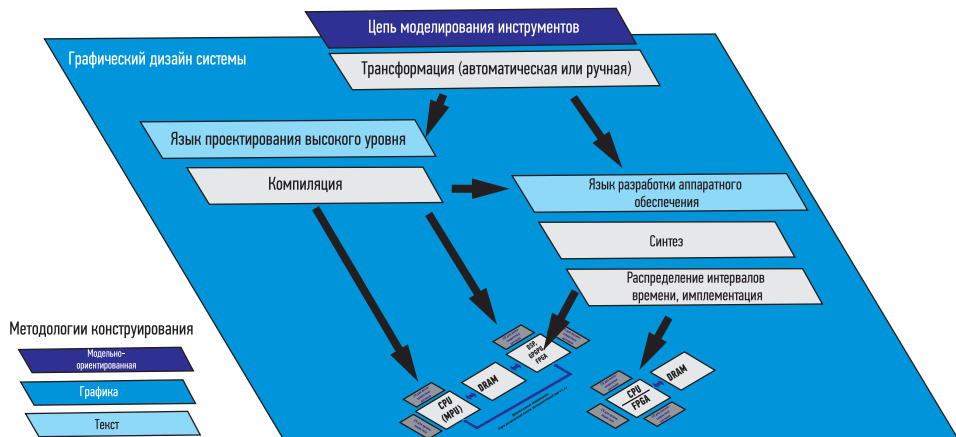
Несколько разновидностей типов портов ввода/вывода и их расширяемость необходимы для добавления стольких типов высокоскоростных и низкоскоростных аналоговых сигналов и датчиков, сколько может потребоваться системе. Уже на этапе проектирования могут понадобиться дополнительные датчики, которые необходимы системе для повышения интеллектуальных знаний о работоспособности оборудования, выполнения более точного высокоскоростного управления, увеличения пропускной способности и расширения возможностей для ее настройки в будущем при изменении сервисов бизнес-моделей.

## БЫСТРОЕ СИНХРОНИЗИРОВАННОЕ ДВИЖЕНИЕ

Для увеличения скорости и улучшения точности ввода-вывода требуются расширенные вычислительные возможности. Кроме того, по мере добавки все большего количества различного программного обеспечения требуется повышение синхронизации при одновременном уменьшении многосекторального управления и джиттера (дрожания фронта импульсов синхронизации вследствие паразитной фазовой модуляции) на наносекундном уровне. Это особенно важно, когда имеется необходимость обеспечения работы оборудования в режиме реального времени.

Некоторые типы самого современного высокотехнологичного

**РИС. 3.** ► Упрощенная схема различных рабочих процессов и методологий, используемых для разработки



оборудования, например установки для лазерной резки и разделения полупроводниковых пластин на кристаллы, требуют 100 нс или более точную синхронизацию при управлении движением луча для обеспечения высокой точности резки. Для достижения необходимой пропускной способности траектории движения должны быть рассчитаны быстро и точно, причем часто — для многоосевого управления. Выбор подходящей гетерогенной архитектуры может существенно помочь в удовлетворении таких непростых требований.

По мере расширения вычислительных возможностей на уровне отдельного контроллера системы становятся все более и более распределенными. А для того чтобы полностью реализовать потенциал распределенного управления, высокоскоростная связь также должна поддерживать функционирование в режиме реального времени. Для удовлетворения будущих потребностей IoT в ближайшей перспективе внимание следует в первую очередь уделить унифицированной сети связи, основанной на последнем обновлении сетей для доставки синхронизированных данных и чувствительных ко времени запросов (англ. time-sensitive networking, TSN) для стандартного протокола Ethernet (IEEE 802.1).

Кроме того, в эпоху развитого IoT потребуется конвергентная сеть, которая сможет запускать медиаконтент, выполнять управление, проводить измерение и обслуживать трафик, гарантируя при этом синхронизацию по времени и задержку, определять приоритеты, пропускную способность и избыточность, а также

иметь высокую отказоустойчивость. Благодаря использованию стандартного Ethernet, который уже сегодня обеспечивает скорости в диапазоне от 1 до 100 Гб, затраты будут сведены к минимуму из-за отсутствия требований создавать запас уникального оборудования и повышать производительность. Для этого часто необходима поддержка существующих на время развертывания протоколов, поэтому система управления может также работать как шлюз, поддерживающий несколько протоколов.

### СТОИМОСТЬ ЖИЗНЕННОГО ЦИКЛА

При использовании архитектур, которые по своей природе являются гибкими и могут расширять систему, важен контроль и управление затратами не только на их разработку, но и на весь жизненный цикл продукта. Мультиплатформенный подход, осуществляющий поддержку машин одноуровневой архитектуры нескольких классов, означает, что аппаратное и программное обеспечение можно использовать повторно. Такую возможность как раз могут обеспечить гетерогенные архитектуры.

Так, например, при использовании FPGA аппаратные решения могут быть обновлены в процессе работы путем включения новых функций, ответа на новые стандарты, с целью повышения производительности и по целому ряду иных причин, причем без необходимости обслуживания на местах. Инструменты, которые можно применить при удаленном развернутом управлении системой для ее обслуживания и отладки, должны быть частью

непосредственно самой программной платформы. Это расширяет возможности развертывания IoT, а такая платформа, основанная на гетерогенных архитектурах, может сэкономить затраты в течение всего жизненного цикла продукта.

Уже на современном этапе развития промышленного «Интернета вещей», когда вы приступаете к определению характеристик и непосредственному конструированию новейших интеллектуальных машин, необходимо учитывать много факторов (рис. 3). Подход к архитектуре управления и организации портов ввода-вывода нужно выбирать с учетом их функционирования в определенной экосистеме. Необходимо оценить и то, сколько потребуется времени для новой разработки, чтобы она удовлетворяла требованиям по управлению и обработке информации, поступающей в режиме реального времени с распределенных, подключенных машин и оборудования. При этом требуется учитывать и то, что обеспечить должное функционирование нужно не только в настоящий момент, но и в будущем. Необходимо также четко уяснить, какой следует использовать подход к платформе, будет ли она разрабатываться силами собственных специалистов компании или путем объединения с внешними разработчиками.

Учитывая всю эту информацию, вы сможете сосредоточить аппаратные и программные ресурсы на настройке и дифференцировании конечных продуктов. Для этого необходимо понять, какие ключевые возможности и технологии помогут стать эффективным участником эпохи IoT. ●