

УДК 004

***ПРИНЦИПЫ И МЕТОДЫ СИСТЕМНОЙ ИНЖЕНЕРИИ В
ПРОЕКТИРОВАНИИ И ВНЕДРЕНИИ УМНОГО ДОМА НА БАЗЕ HOME
ASSISTANT***

Казнов А.И.,

студент,

Калужский государственный университет им. К.Э. Циолковского,

Калуга, Россия

Соколов Н.В.,

старший преподаватель кафедры информатики и информационных технологий

Калужский государственный университет им. К.Э. Циолковского,

Калуга, Россия

Аннотация

В работе рассматривается применение принципов и методов системной инженерии при проектировании и внедрении умного дома на базе платформы Home Assistant. Анализируются ключевые проблемы интеграции различных устройств и подсистем, а также способы их решения с помощью системного подхода. Исследование демонстрирует, как системная инженерия способствует созданию масштабируемой и автоматизированной среды.

Ключевые слова: умный дом, Home Assistant, системная инженерия, интеграция, автоматизация, IoT, проектирование.

***PRINCIPLES AND METHODS OF SYSTEM ENGINEERING IN
DESIGNING AND IMPLEMENTING A SMART HOME BASED ON HOME
ASSISTANT***

Kaznov A.I.,

student,

Kaluga State University named after K.E. Tsiolkovsky,

Kaluga, Russia

Sokolov N.V.,

Senior Lecturer at the Department of Computer Science and Information Technology

Kaluga State University named after K.E. Tsiolkovsky,

Kaluga, Russia

Abstract

The paper examines the application of principles and methods of systems engineering in the design and implementation of a smart home based on the Home Assistant platform. Key issues of integrating various devices and subsystems are analyzed, as well as ways to solve them using a systems approach. The study demonstrates how systems engineering contributes to the creation of a scalable and automated environment.

Keywords: smart home, Home Assistant, systems engineering, integration, automation, IoT, design.

Современные технологии стремительно развиваются, и одним из самых востребованных направлений становится создание умных домов — автоматизированных систем, способных повысить комфорт, безопасность и энергоэффективность жилых помещений [8]. Платформа Home Assistant представляет собой открытое программное обеспечение для интеграции множества устройств и управления ими в единой экосистеме, что делает её мощным инструментом для построения подобных систем [3]. Однако процесс проектирования и внедрения таких систем сопряжён с множеством технических и организационных вызовов, связанных с разнообразием устройств, необходимостью обеспечения надёжности и безопасности, а также адаптацией под индивидуальные потребности пользователей. Эти вызовы обусловлены тем, что современный умный дом является сложной киберфизической системой, требующей междисциплинарного подхода к своему созданию и обслуживанию.

Для решения этих задач критически важно применять системный подход, который лежит в основе системной инженерии — междисциплинарной методологии, ориентированной на комплексное управление жизненным циклом сложных систем [9]. Принципы и методы системной инженерии позволяют эффективно управлять требованиями, проектировать архитектуру системы, проводить тестирование и интеграцию различных компонентов, а также управлять изменениями и конфигурациями. В данной статье рассматривается применение системной инженерии на примере создания умного дома с использованием Home Assistant, что позволяет повысить качество и эффективность разработки таких решений, обеспечивая их целостность, надежность и способность к эволюции.

Современный умный дом, как уже говорилось выше, представляет собой сложную киберфизическую систему, включающую разнообразные устройства и подсистемы, такие как системы освещения, климат-контроля, безопасности, мультимедиа и бытовой техники. Каждая из этих подсистем может использовать различные протоколы связи (Zigbee, Z-Wave, Wi-Fi, Bluetooth, MQTT и другие) [2], стандарты и аппаратные платформы, что значительно усложняет их интеграцию в единую управляемую систему. Отсутствие единого отраслевого стандарта приводит к проблемам совместимости, дублированию функций, сложностям при централизованном управлении и, как следствие, к увеличению стоимости владения и снижению пользовательского опыта.

Дополнительными вызовами являются обеспечение масштабируемости системы при увеличении числа подключённых устройств и поддержание её устойчивости к сбоям. Для сохранения высокой надёжности требуется продуманная архитектура, способная эффективно распределять вычислительные и коммуникационные нагрузки, а также реализовывать механизмы отказоустойчивости и резервирования. Например, отказ одного датчика не должен приводить к полному прекращению работы всей системы безопасности, вместо этого система должна переконфигурироваться и использовать данные с

других сенсоров или уведомить пользователя о неисправности. Кроме того, с ростом числа устройств возрастает и потенциальная угроза для кибератак, что требует реализации комплексных мер информационной безопасности [7], включая строгую аутентификацию, сквозное шифрование данных, регулярное обновление программного обеспечения и управление правами доступа. Игнорирование этих аспектов может привести к утечке конфиденциальных данных пользователя или даже к физическому взлому.

Рассмотрим конкретную ситуацию автоматизации управления освещением, климатом и системой безопасности. Для корректной работы такой системы необходимо обеспечить синхронизацию данных с различных сенсоров (датчики движения, температуры, открытия дверей), своевременное выполнение сценариев автоматизации и возможность удалённого управления. Это требует применения продвинутых методов мониторинга состояния устройств, обработки событий в реальном времени и масштабируемой архитектуры, что напрямую связано с задачами системной инженерии. Без системного подхода попытка реализации такой автоматизации может вылиться в набор разрозненных скриптов, которые конфликтуют друг с другом, приводя к непредсказуемому поведению системы, например, одновременной попытке кондиционера охладить помещение и обогревателю — нагреть его.

Применение системной инженерии в проектировании умного дома основывается на нескольких фундаментальных принципах, которые обеспечивают целостность и эффективность создаваемого решения [4].

Целостность системы (системный подход) является ключевым принципом системной инженерии. Он предполагает рассмотрение объекта исследования как единого целого, а не набора разрозненных элементов. В контексте умного дома это означает необходимость учета взаимосвязей и взаимодействий между всеми подсистемами (освещение, безопасность, климат). Например, сценарий «Вечерний киносеанс» должен не просто включить телевизор, но также приглушить освещение, закрыть шторы и отрегулировать

температуру кондиционера, обеспечивая согласованную и эффективную работу всей системы в целом [9]. Игнорирование этого принципа приводит к созданию «островков» автоматизации, которые не взаимодействуют между собой, значительно снижая полезность системы.

Иерархия и модульность позволяют структурировать сложную систему на взаимосвязанные уровни и отдельные функциональные блоки (модули). Такая организация упрощает разработку, тестирование и последующее обслуживание умного дома. Модули могут разрабатываться, тестироваться и модернизироваться независимо, сохраняя при этом совместимость с остальной системой через чётко определенные интерфейсы. Например, подсистема освещения может быть выделена в отдельный модуль, взаимодействующий с ядром Home Assistant через стандартизированный API. Это позволяет заменить всю подсистему освещения (например, перейти с wi-fi лампочек на Zigbee) без необходимости переписывать логику работы всего умного дома, при условии, что новый модуль реализует тот же интерфейс.

Управление требованиями — это систематический процесс выявления, анализа, документирования и контроля требований к системе на протяжении всего её жизненного цикла. Для умного дома это включает тщательный сбор пожеланий пользователей таких как функциональные требования (я хочу, чтобы свет в прихожей включался при моем приходе), технических характеристик устройств и ограничений по безопасности и надежности (нефункциональные требования - система должна реагировать на событие открытия двери менее чем за 2 секунды или все данные должны шифроваться). Детальное документирование требований обеспечивает прозрачность процесса разработки для всех участников проекта (инженеров, пользователей, монтажников) и позволяет избежать несоответствий между ожиданиями заказчика и финальной реализацией, обеспечивая выполнение всех заявленных функций.

Верификация и валидация обеспечивают проверку правильности разработки и соответствия системы требованиям. Верификация отвечает на

вопрос: «делаем ли мы систему правильно?», то есть проверяет, соответствуют ли компоненты и интеграции техническим спецификациям и проектной документации. Это достигается путём модульного и интеграционного тестирования. Валидация же отвечает на вопрос: «делаем ли мы правильную систему?», проверяя, соответствует ли конечный результат ожиданиям пользователей и целям проекта. Например, верификация может проверить, что скрипт автоматизации освещения технически исполняется без ошибок. Валидация же убедится, что свет включается именно тогда, когда это удобно пользователю, а не с задержкой или, наоборот, слишком рано.

Реализация вышеописанных принципов осуществляется через конкретные методы системной инженерии, которые находят своё практическое применение на каждом этапе проекта умного дома.

Моделирование и проектирование архитектуры системы являются фундаментальными этапами. С помощью моделирования создаются схемы взаимодействия компонентов, описываются их функции и связи, что позволяет визуализировать структуру системы и выявить потенциальные узкие места ещё на ранних стадиях. Для умного дома на Home Assistant это может включать в себя диаграммы, отображающие, какие устройства подключены к каким шлюзам (Zigbee-мост, MQTT-брокер), как данные взаимодействуют между компонентами и где находятся потенциальные точки отказа [1]. Такой подход позволяет оптимизировать сетевую топологию и размещение оборудования до начала физического монтажа.

Анализ требований и их документирование предполагает систематический сбор и формализацию всех требований к системе. На этом этапе проводятся интервью с пользователями, анализируются технические спецификации приобретаемого оборудования, учитываются нормативные ограничения. Все выявленные функциональные (управление светом, климатом), нефункциональные (быстродействие, надежность) и пользовательские требования заносятся в спецификацию. Это не статичный документ; он живёт на

протяжении всего проекта и изменяется по мере поступления новых требований или корректировки старых. Документирование обеспечивает прозрачность и контроль на всех этапах проекта, служа источником истины для всех участников.

Прототипирование и тестирование используются для проверки корректности реализации компонентов и их интеграции. Создание прототипов, например, на макетной плате с ESP8266 и датчиком движения, позволяет оперативно выявлять и устранять ошибки на аппаратном и программном уровне, проверять работоспособность автоматизаций и сценариев в условиях, близких к реальным, до их окончательного развертывания [5]. Тестирование проводится как на уровне отдельных модулей (unit-тестирование скриптов автоматизации), так и на уровне всей системы (system testing) для обеспечения её надёжности и безопасности. Особое внимание уделяется тестированию на отказ: что произойдет, если пропадет связь с устройством, отключится электричество или сработает ложное срабатывание датчика.

Управление изменениями и конфигурациями необходимо для контроля версий программного обеспечения, аппаратных компонентов и настроек системы. В процессе эксплуатации умного дома часто возникают новые требования или необходимость обновления. Этот метод обеспечивает, что все изменения вносятся системно, документируются и не ломают существующую функциональность. Для Home Assistant это может означать использование системы контроля версий (например, Git) для хранения конфигурационных файлов (YAML), что позволяет отслеживать все изменения, при необходимости откатываться к предыдущим стабильным версиям и легко восстанавливать систему после сбоя.

Интеграция различных подсистем — одна из ключевых задач, где методы системной инженерии проявляются наиболее ярко. Применяются методы стандартизации интерфейсов и протоколов (например, использование MQTT в качестве единой шины данных для всех устройств), а также разрабатываются механизмы обмена данными между подсистемами освещения, климат-контроля

и безопасности. Это обеспечивает согласованное функционирование системы и позволяет реализовать комплексные сценарии автоматизации, повышая удобство и безопасность для пользователей. Например, подсистема безопасности при срабатывании датчика движения на первом этаже может не только включить сирену и отправить уведомление, но и через подсистему освещения включить весь свет в доме для отпугивания злоумышленника.

Далее рассмотрим практическое применение описанных принципов и методов на примере проекта умного дома, развернутого на базе платформы Home Assistant.

Проектирование начинается с тщательного сбора и анализа требований пользователя [2]. На этом этапе выявляются все ключевые функции и сценарии: управление освещением (включение по расписанию, по движению, голосом), климат-контролем (поддержание комфортной температуры, учет погодных условий), безопасность (уведомления об открытии дверей, имитация присутствия), а также специфические задачи, например автоматизация управления калиткой и воротами. Важно учесть не только пожелания, но и технические ограничения (наличие wi-fi покрытия, энергопотребление) и строгие требования к надежности (система безопасности должна работать даже при отсутствии интернета) и безопасности (шифрование видеопотока с камер).

Следующим этапом идёт проектирование архитектуры. На данном этапе система разбивается на логические узлы и компоненты. В рамках исследования был реализован блок управления калиткой на базе микроконтроллера ESP8266 [6], который обеспечивает автоматический контроль времени подачи напряжения на электромеханический привод для предотвращения его перегрева и излишнего энергопотребления. Данный блок через wi-fi подключен к MQTT-брокеру, который является частью экосистемы Home Assistant, что обеспечивает надежный обмен сообщениями. Также была внедрена система управления светом в подъезде, интегрированная с обычными механическими выключателями. Для этого было реализовано отслеживание положения

выключателей с помощью датчиков открытия (конечных выключателей), что позволяет синхронизировать физическое состояние с программной логикой [2] Home Assistant, обеспечивая корректное реагирование на действия пользователя как с физического выключателя, так и из мобильного приложения [2].

Этап использования интеграций и автоматизаций позволяет применить как готовые интеграции (для устройств Zigbee через координатор), так и собственные, написанные на основе ESPHome для управления самодельными устройствами на ESP8266. Для взаимодействия между компонентами активно использовался протокол MQTT [3], выступающий в роли стандартизированного шинного интерфейса. Автоматизации же были настроены на основе сложных сценариев: включение освещения при обнаружении движения в сумеречное время суток [6], управление климатом в зависимости от времени суток, присутствия людей и данных с внешнего метеодатчика, а также оповещения о событиях безопасности (открытие двери в нерабочее время) через Telegram-бота.

Тестирование и отладка системы проводились итеративно и поэтапно. Начиналось всё с модульных тестов (проверка работы каждого датчика, каждого реле, каждого скрипта в изоляции). Далее проводилась интеграционная проверка (корректно ли MQTT-сообщение от датчика движения приводит к выполнению сценария освещения). Финальным этапом стала комплексная проверка работы всех подсистем под нагрузкой в течение длительного времени. Особое внимание уделялось проверке устойчивости системы к сбоям: моделировалась потеря связи с wi-fi, отключалось электричество, проверялась корректность обработки этих исключительных ситуаций и восстановления работоспособности.

Этап масштабируемости и поддержки были заложены в архитектуру изначально. Использование модульного подхода позволяет быстро добавлять новые устройства и подсистемы без значительных изменений в основной системе. Например, добавление нового датчика температуры включает в себя лишь его сопряжение с Zigbee-координатором и добавление одной сущности в сценарий климат-контроля, а не переделку всей системы в целом.

Обеспечение безопасности данных и устройств было реализовано комплексно: обязательная аутентификация всех пользователей в Home Assistant с разграничением прав (гостевой доступ без управления безопасностью), шифрование передачи данных (HTTPS для веб-интерфейса, TLS для MQTT), изоляция IoT-сети от основной сети пользователя с помощью правил маршрутизации, а также регулярное обновление программного обеспечения всех компонентов, включая прошивки микроконтроллеров [10]. Это позволило значительно снизить риски несанкционированного доступа и повысить надёжность системы в целом.

Несмотря на положительные результаты, были выявлены и некоторые проблемы. Так, сложность интеграции устройств с разными протоколами (например, задержки, присущие для mesh-сетей Zigbee) иногда приводит к небольшим задержкам в коммуникации и нестабильной работе некоторых компонентов при большой загрузке сети. Также значительные требования к безопасности требуют постоянного обновления и мониторинга системы, что увеличивает нагрузку на техническую поддержку. Для решения этих проблем рекомендуется, где это возможно, использовать более стандартизированные и надежные протоколы [3], внедрять автоматические системы мониторинга состояния (например, Healthchecks.io для отслеживания доступности системы) и строго придерживаться политики регулярного обновления программного обеспечения.

Заключение

Системная инженерия играет ключевую роль в проектировании и внедрении умных домов, обеспечивая структурированный и комплексный подход к разработке сложных автоматизированных систем. Как продемонстрировано в работе, применение её принципов и методов, а именно системного подхода, модульности, управления требованиями, верификации и валидации позволяет создавать надёжные, масштабируемые и адаптивные

решения, способные эффективно интегрировать разнообразные устройства и обеспечивать высокий уровень комфорта и безопасности.

Платформа Home Assistant, будучи гибкой и открытой, предоставляет отличную основу для применения методологий системной инженерии. Она позволяет реализовать модульную архитектуру, централизованное управление и сложную автоматизацию, отвечающую строгим и индивидуальным требованиям пользователей.

В перспективе развитие умных домов будет связано с дальнейшей автоматизацией на основе искусственного интеллекта и машинного обучения, улучшением семантической интероперабельности между подсистемами разных производителей и усилением мер кибербезопасности. Системная инженерия продолжит оставаться фундаментом, обеспечивающим успешное внедрение этих инноваций, управление сложностью и поддержание высокого качества, надежности и безопасности систем умного дома на всех этапах их жизненного цикла.

Библиографический список:

1. Home Assistant Architecture Overview. [Электронный ресурс]. — Режим доступа — URL: https://developers.home-assistant.io/docs/architecture_index/ (дата обращения: 23.08.2025).
2. Home Assistant Documentation. [Электронный ресурс]. — Режим доступа — URL: <https://www.home-assistant.io/docs/> (дата обращения: 23.08.2025).
3. Home Assistant: A Comprehensive Guide to Smart Home Automation - Resonatetoinnovate, 2025. [Электронный ресурс]. — Режим доступа — URL: <https://resonatetoinnovate.com/our-blog/smart-home-essentials/home-assistant-smart-home-automation-guide/> (Дата обращения 25.08.2025).
4. ISO/IEC/IEEE 15288:2023. Systems and software engineering — System life cycle processes. — Geneva: ISO, 2023. [Электронный ресурс]. — Режим доступа — URL: <https://www.iso.org/standard/81702.html> (дата обращения: 25.08.2025)

5. Khusamov. Схема моего умного дома на основе ESP8266 - habr.com, 2023. [Электронный ресурс]. — Режим доступа — URL: <https://habr.com/ru/articles/543536/> (дата обращения: 20.06.2025).
6. Viktor-235. Четыре года с умным домом: личный опыт и интересные сценарии - habr.com, 2023. [Электронный ресурс]. — Режим доступа — URL: <https://habr.com/ru/companies/banki/articles/765804/> (дата обращения: 21.08.2025).
7. Кузнецов И. М. IoT и системы управления умным домом // Огарёв-Online. 2017. №2 (91). [Электронный ресурс]. — Режим доступа — URL: <https://cyberleninka.ru/article/n/iot-i-sistemy-upravleniya-umnym-domom> (дата обращения: 28.08.2025).
8. Михайлов С. С. Основные принципы работы "умного дома" // Вестник науки. 2022. №8 (53). [Электронный ресурс]. — Режим доступа — URL: <https://cyberleninka.ru/article/n/osnovnye-printsipy-raboty-umnogo-doma> (дата обращения: 28.08.2025).
9. Системная инженерия. Принципы и практика / А. Косяков, Свит Н. Уильям, Сеймур Дж. Сэмюэль, Бимер М. Стивен ; перевод А. А. Слинкин. — Саратов : Профобразование, 2017. — 624 с. — ISBN 978-5-4488-0042-9. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/64063.html> (дата обращения: 25.08.2025). — Режим доступа: для авторизир. пользователей.
10. Ходжамаммедов М. М., Мухаммедов М. М., Рахымов К. Д., Рустамов Р. Р. Безопасность интернета вещей (IoT) защита умных устройств и сенсорных сетей от взлома // Символ науки. 2024. №12-1-2. [Электронный ресурс]. — Режим доступа — URL: <https://cyberleninka.ru/article/n/bezopasnost-interneta-veschey-iot-zaschita-umnyh-ustroystv-i-sensornyh-setey-ot-vzloma> (дата обращения: 28.08.2025).

Оригинальность 75%