

ВВЕДЕНИЕ В ИНТЕРНЕТ ВЕЩЕЙ

Для подготовки к «Кейс-турниру»
от компании «МГБот» и
Благотворительного фонда «Велес»

Автор - Котов М. К.



MGBOT®



ПРИ ПОДДЕРЖКЕ
ФОНДА
ПРЕЗИДЕНТСКИХ
ГРАНТОВ

ОГЛАВЛЕНИЕ

ИНТЕРНЕТ ВЕЩЕЙ – ОПРЕДЕЛЕНИЕ.....	3
ПРИМЕНЕНИЕ ИНТЕРНЕТА ВЕЩЕЙ.....	6
СВЯЗАННЫЕ ТЕХНОЛОГИИ.....	12
ПРИМЕРЫ ИСПОЛЬЗОВАНИЯ ИНТЕРНЕТА ВЕЩЕЙ.....	16
ЗАДАЧИ.....	20
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	43

ИНТЕРНЕТ ВЕЩЕЙ - ОПРЕДЕЛЕНИЕ

Термин «Интернет вещей» появился благодаря предпринимателю Кевину Эштону, который являлся одним из основателей центра Auto-ID в Техническом университете города Бостон (MIT). Команда, в которой Эштон работал, придумала идею о том, что можно связывать объекты с Интернетом с помощью метки RFID. Впервые он использовал фразу «Интернет вещей» в 1999 году, и с тех пор она стала популярной.



ИНТЕРНЕТ ВЕЩЕЙ – концепция вычислительной сети физических предметов, обладающих встроенными технологиями для обмена информацией между собой и окружающей средой. Поговорим эту технологию называют также IoT – Internet of Things.

На сегодняшний день количество устройств, подключенных к интернету вещей, уже давно превысило количество людей. Эти устройства могут совершать обмен данными через сети или облачные платформы, подключенные к Интернету вещей – иначе говоря, *платформы для интернета вещей*.

Говоря об устройствах с технологией Интернета вещей, сначала в голове всплывают сложные механизмы, типа роботов-манипуляторов, суперкомпьютеров, но IoT охватывает значительно более разнообразные и простые устройства, например, электронику, которая никогда ранее не была подключена к сети: холодильники, кофемашины, стиральные машины, розетки, чайники.

Изначально IoT применялся только в самых современных на тот период проектах с использованием дорогого оборудования и был недоступен обычному человеку. Развитие Интернета вещей за последнее десятилетие и внедрение её в нашу повседневную жизнь произошло благодаря следующим факторам:

- Создание устройств с современными цифровыми технологиями стало более доступно
- Число продуктов с Wi-Fi резко увеличилось
- Смартфоны стали неотъемлемой частью жизни людей
- Функционал смартфонов стал позволять осуществлять управление другими устройствами

Интернет вещей можно условно разделить на разные категории, среди которых можно выделить *локальный* и *глобальный* IoT, *проводной* и *беспроводной*, а также *бытовой* и *промышленный*.



Глобальный или всемирный Интернет вещей охватывает устройства, которые могут располагаться удаленно друг от друга и обычно при их взаимодействии используется беспроводное соединение. Локальный интернет вещей использует как проводной, так и беспроводной способ соединения. Например, система «Умный дом» - локальный, закрытый проект, где все устройства связаны между собой, и взаимодействие с ними предусмотрено только внутри данной локации – то есть, жилого помещения. Связываться с ним может только владелец жилья, посторонний человек не сможет повлиять на данную экосистему благодаря предусмотренной безопасности.



Беспроводной



Проводной

Проводной интернет вещей часто используется на закрытых объектах, где следует соблюдать меры безопасности, чтобы никто не мог каким-либо способом взломать систему снаружи, либо вывести данные изнутри. Также более высокая надежность передачи данных или особенности окружающей среды могут обуславливать использование вышеуказанного способа. Такая система, где все устройства взаимодействуют друг с другом, но при этом используется провод, обычно применяется на промышленных объектах, визуализация информации происходит напрямую на мониторах, где операторы могут отслеживать состояние датчиков и исполнительных устройств. Беспроводной Интернет вещей в настоящее время – самый популярный способ организации сети взаимодействия устройств. Благодаря высокой скорости передачи данных, мобильности и интеграции с любым смартфоном беспроводные интернет вещи появились у каждого человека и окружают нас везде, начиная от автобусной остановки и заканчивая системой управления роботом-манипулятором на автомобильном заводе.



Промышленный



Бытовой

Промышленный Интернет вещей (IIoT – Industrial Internet of Things) применяют на производстве, складах, заводах и в лабораториях. Его задача - автоматизировать и упростить производственные процессы. Бытовой IoT, чаще всего его называют просто IoT — это бытовые приборы, объединенные друг с другом. Он применяется в быту, чтобы автоматизировать рутинные задачи или удаленно управлять различными устройствами.

ПРИМЕНЕНИЕ ИНТЕРНЕТА ВЕЩЕЙ

В настоящее время технология Интернета вещей применяется во многих отраслях. Как и любую другую современную технологию, Интернет вещей пытаются использовать во всех подходящих и неподходящих случаях. Возможно, лет через десять, Интернет вещей «найдет свое место» и укрепит позиции в конкретных областях, как это было с электричеством, компьютерами, автомобилями и т.д. На данный момент можно выделить следующие основные направления:



Больше всего интернет вещей используется в **ПРОИЗВОДСТВЕ**. Связано это в первую очередь с тем, что создание новых материалов и продукции в легкой и тяжелой промышленности является самым жизненно необходимым и приоритетным занятием для человечества. Соответственно, производство всегда обладает большими денежными и человеческими ресурсами, чтобы осваивать новые технологии, применять их для улучшения жизни людей. Для производства даже выделяют отдельное направление интернета вещей – *промышленный Интернет Вещей*.

Примеры IIoT:

- Датчики на оборудовании и станках, которые собирают данные об их работе и помогают предотвращать поломки.
- Системы климат-контроля, которые анализируют температуру и влажность, автоматически регулируют микроклимат в цехе или на складе.
- Датчики на продукции и деталях, которые помогают в инвентаризации и выявлении брака.
- Системы удаленного управления производственным оборудованием, например, дистанционного запуска станков.

- Датчики загрязнения, которые анализируют выбросы предприятия и следят за соблюдением экологических норм.

IIoT нужен, чтобы снизить затраты на производстве и избежать убытков. Датчики на оборудовании помогут понять, что пора провести техобслуживание. Система климат-контроля проследит, чтобы товар не испортился в жару.

Данные, которые собирают все датчики, можно загружать в системы аналитики, чтобы строить прогнозы и анализировать работу всего производства. [1]



Интернет вещей в производстве (оператор конвейерного модуля отслеживает состояние системы)

РИТЕЙЛ (розничная торговля) – в настоящее время активно расширяет поле деятельности с применением Интернета вещей. И это не только когда покупатель применяет карту для оплаты товаров или производит покупку с помощью смартфона или браслета, но и более передовые технологии, которые все больше начинают появляться в знакомых нам магазинах. Сюда входят «умные тележки», которые автоматически считают весь выбранный вами товар и формируют оплату с вашего онлайн-кошелька и, как итог – вы приобретаете все товары просто пройдя мимо считывающего терминала и кассира. Также существуют роботы-мерчендайзеры, перемещающиеся вдоль стеллажей и считающие нехватку товаров, а в свободное время могут помочь Вам найти в магазине определенную позицию. «Умные» полки с помощью датчика давления определяют нехватку продукции с учетом ее приобретения покупателями, а затем делают заказ поставщику, при этом менеджер по закупкам товара только лишь отслеживает весь процесс, и ему не приходится принимать решения и рассчитывать нужное количество и дату доставки. Все вышеперечисленное не могло бы существовать без интернета

вещей – одной из основополагающей технологии для разработки современных систем управления.



Интернет вещей в ритейле (покупатель сканирует QR-код для оплаты лежащих в тележке товаров)

УМНЫЙ ДОМ – одно из самых упоминаемых словосочетаний, когда человека спрашивают: «Что такое Интернет вещей?». Действительно, сейчас «Умный дом» и IoT тесно связаны между собой и представляют собой систему управления функциями жилого помещения. При этом, раньше, когда человек говорил, что он установил в своей квартире/доме такую систему, интернет вещей при этом не подразумевался. «Умный дом» представлял собой автоматическую систему управления с обратной связью. Сюда относится включение освещения по датчику движения или звука, включение воды по датчику расстояния и так далее. Интернет вещей же открыл «второе дыхание». Теперь «Умный дом» – это система, которая синхронизирована с владельцем, а точнее его состоянием организма и пожеланиями. Когда человек находится в помещении, система может изменить угол освещения и тип света, например, для чтения электронной книги. Умный холодильник составит список продуктов для правильного питания с учетом состояния здоровья и режима владельца и многое другое. Другими словами, человек является частью экосистемы «Умного дома» и все процессы строятся вокруг безопасности и комфорта его жизнедеятельности.



Интернет вещей в умном доме (человек настраивает через приложение освещение и акустическую систему перед просмотром фильма)

ЛОГИСТИКА – динамично развивающееся направление с применением современных технологий, где Интернет Вещей играет не последнюю роль. Если вы подумали, что не знакомы с примерами применения технологии в данной сфере, то это скорее всего не так. Большинство людей сейчас пользуется заказом товаров по почте. Обычно сервис располагает функцией трекинга (отслеживания) груза начиная от склада и заканчивая дверью вашего дома. При заказе товара, вам выдается специальный трек-номер, который можно ввести в базу отслеживания и узнать, где находится ваша посылка. Отслеживание происходит благодаря тому, что на посылке есть специальный трек-код или QR-код, который считывается специальным оборудованием, синхронизированным с общей базой данных. Это наглядный и простой пример того, как Интернет вещей объединяет систему отслеживания посылок, клиентов и, собственно, самих товаров с встроенной технологией для взаимодействия.

Также, сам человек может стать «интернет-вещью». Например, на современно оборудованном промышленном предприятии сотрудники могут носить специальный жилет или каску с нанесенной меткой, которая считывается при перемещении человека из одного цеха в другой. Соответственно, он отслеживается в системе, и начальник отдела всегда может знать, где и сколько проводил сотрудник времени. Это повышает эффективность работы и позволяет провести анализ данных для оптимизации рабочего пространства.

Логистические компании используют системы отслеживания грузовых автомобилей, что позволяет понять, где находится тот или иной груз, как распределить автомобили по всей карте доставки для экономии топлива и сокращения рабочего дня.



Интернет вещей в логистике (сотрудник нефтяной компании тестирует каску, отслеживающую местоположение и состояние здоровья человека [2])

«УМНЫЙ ГОРОД» с интернетом вещей — это концепция единого управляемого пространства, которое повышает безопасность и комфорт горожан и следит за эффективностью

работы всех служб. Концепция состоит из базовых подсистем, которые охватывают всю жизнь мегаполиса:

- Управление городским освещением
- Автоматизированный учет воды, тепла, газа, электричества
- Видеонаблюдение на транспорте и фиксация правонарушений
- Управление лифтами
- Общественный Wi-Fi
- Мониторинг и охрана люков
- Контроль за наполненностью мусорных баков
- Обеспечение связи и безопасности при транспортировке нефти, газа, бензина
- Управление светофорами и перекрёстками
- Мониторинг экологической обстановки города
- ИТ-системы для общественного транспорта [3].



Интернет вещей в умном городе (городской житель ждет нужный ему автобус через приложение трекинга общественного транспорта)

МЕДИЦИНСКИЙ ЙОТ, иногда называемый IoT в здравоохранении, относится к одним из самых быстро развивающихся отраслей по увеличению количества IoT-устройств. В настоящее время создается широкий спектр устройств и приложений IoT, специально разработанных для нужд здравоохранения, таких как датчики для удаленного мониторинга пациента, телемедицинских консультаций и доставки лекарств. Медицинский Интернет вещей также опирается на технологии искусственного интеллекта и машинного обучения для улучшений традиционных медицинских устройств, таких как интеллектуальный ингалятор для людей с астмой. Специальное оборудование позволит всегда своевременно предупредить об опасности здоровью, вести профилактику образа жизни.

Медицинский Интернет вещей может помочь улучшить результаты лечения пациентов за счет отслеживания пациентов в режиме реального времени, расширенной диагностики, роботизированной хирургии и многое другое.



Интернет вещей в медицине (врач проводит сеанс телемедицины и отслеживает показатели здоровья пациента в реальном времени)

СИТИ-ФЕРМЕРСТВО - перспективное направление в сельском хозяйстве, когда продукты (овощи, ягоды, зелень) выращивают в городе, а не за его пределами. Это общемировая тенденция: такой подход обеспечивает колоссальную экономию логистики и ресурсов, что очень важно для перенаселенных территорий. Снижение затрат позволяет снизить стоимость продуктов, не говоря уже о том, что люди получают возможность съесть свежие овощи и зелень, которые не прошли тысячи километров, чтобы добраться до стола к покупателю. Но просто выращивая цветок на подоконнике, вы еще не становитесь сити-фермером. Это направление подразумевает использование современных технологий как цифровых (умные устройства, Интернет Вещей), так и выращивания (аэропоника, гидропоника и другие).

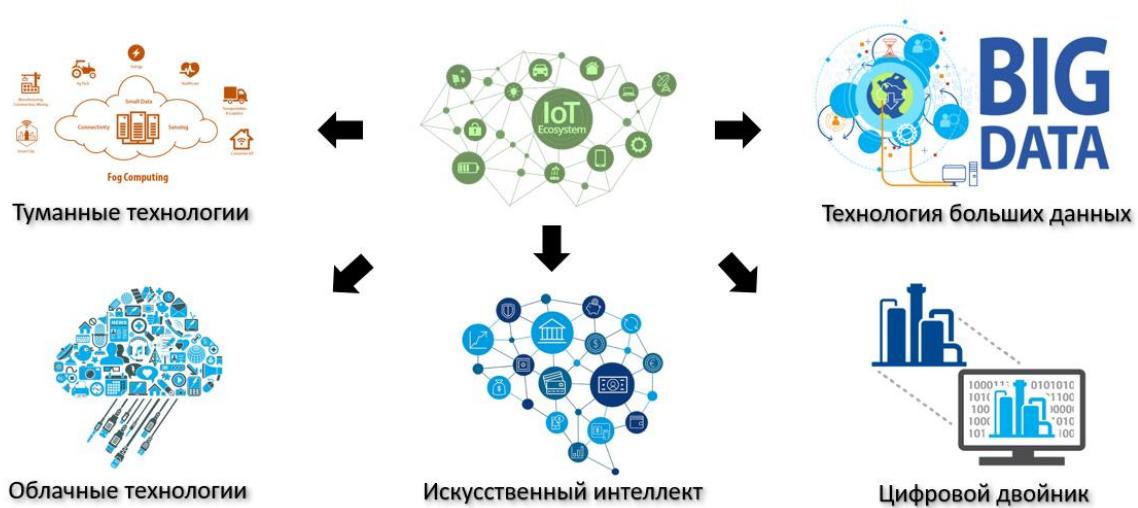
Применение Интернета Вещей позволяет удаленно отслеживать параметры окружающей среды выращиваемых культур, собирать данные со всех участков для анализа происходящих процессов, организовывать эффективные системы автополива, освещения и многое другое.



Интернет вещей в сити-фермерстве (макет умной теплицы используется как площадка для выращивания растений в условиях городской среды)

СВЯЗАННЫЕ ТЕХНОЛОГИИ

Как самостоятельная технология, Интернет Вещей редко используется для создания серьезных и больших проектов. Чаще всего её используют в совокупности с другими. В таком случае, IoT еще больше раскрывается как направление, способствующее достижению целей в рамках какого-либо концепта. Одними из наиболее популярных технологий, связанными с интернетом вещей, являются: *облачные и туманные технологии, Big Data Technology (технология больших данных), цифровой двойник и искусственный интеллект*.



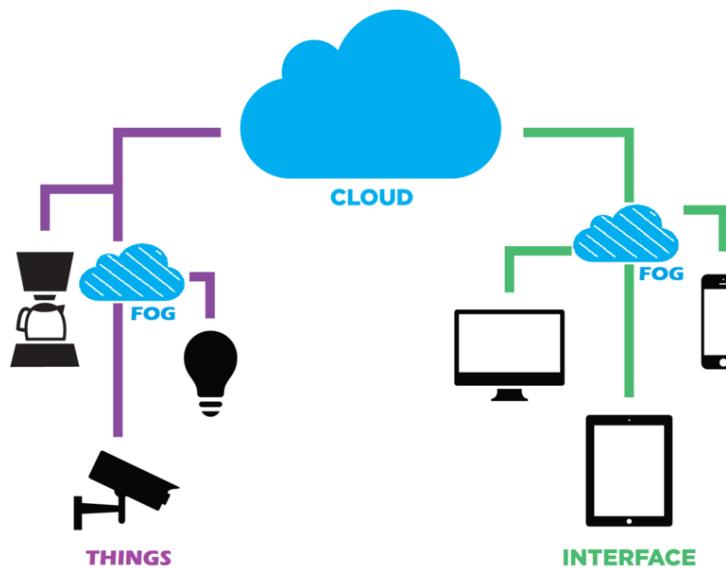
ОБЛАЧНЫЕ ТЕХНОЛОГИИ (или облачные вычисления, *cloud computing*) – технологии распределенной обработки цифровых данных, с помощью которых компьютерные ресурсы предоставляются интернет-пользователю как онлайн-сервис. Большинство людей в современном мире пользуются данной технологией – хранят фотографии на Гугл диске, пользуются онлайн-таблицами. Для Интернета Вещей облачный сервер является одним из ключевых элементов во всей системе. С помощью него можно передавать, хранить и анализировать данные, иметь доступ к системе из любой точки мира. Важным моментом является то, что данные хранятся не в памяти оборудования на месте, а собираются в единую базу данных в облаке – это как раз характеризует взаимодействие облачных технологий и Интернета Вещей.

Развитие технологий в области софта и аппаратной части, появление новых протоколов связи привели к расширению интернета вещей (IoT). Количество устройств растёт день ото дня,

и они генерируют огромный объём данных. Поэтому возникает потребность в удобной архитектуре системы, способной обрабатывать, хранить и передавать эти данные.

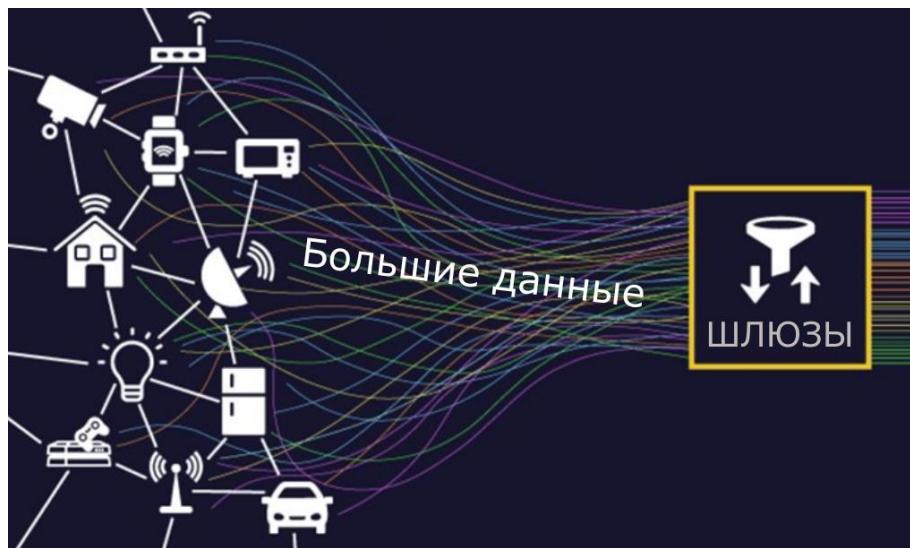
Сейчас для этих целей используют облачные сервисы. Однако становящаяся всё более популярной технология **ТУМАННЫХ ВЫЧИСЛЕНИЙ (FOG)** способна дополнить облачные решения, масштабировав и оптимизировав инфраструктуру IoT. По сути, туман – это «субтехнология» облака.

«Облака» способны закрыть большинство запросов IoT. Например, обеспечить мониторинг служб, быструю обработку любых объёмов данных, генерируемых устройствами, а также их визуализацию. Туманные же вычисления эффективнее при решении задач реального времени. Они обеспечивают быстрый отклик на запросы и минимальную задержку при обработке данных. То есть «туман» именно дополняет «облако», расширяет его возможности [4].



Множество устройств, используемых в Интернете вещей, собирают множество единиц информации практически в реальном времени. Поэтому именно Интернет вещей является одним из самых значительных факторов роста больших данных. Устройства Интернета вещей собирают данные в реальном времени, генерируя огромный объем информации. Согласно прогнозам International Data Corporation (IDC), к 2025 году в мире будет 55,7 млрд устройств, подключенных к Интернету вещей, которые будут генерировать 73,1 зеттабайт данных. Это приводит к большим данным Интернета вещей, с которыми могут не справиться традиционные инструменты обработки и управления данными. Поэтому, на помощь приходит такая технология как **BIG DATA** - совокупность непрерывно увеличивающихся объемов информации одного контекста, но разных форматов представления, а также методов и средств для эффективной и быстрой обработки [5]. Обычно данные Интернета вещей направляются в центральное расположение, где их можно анализировать и интерпретировать, а также

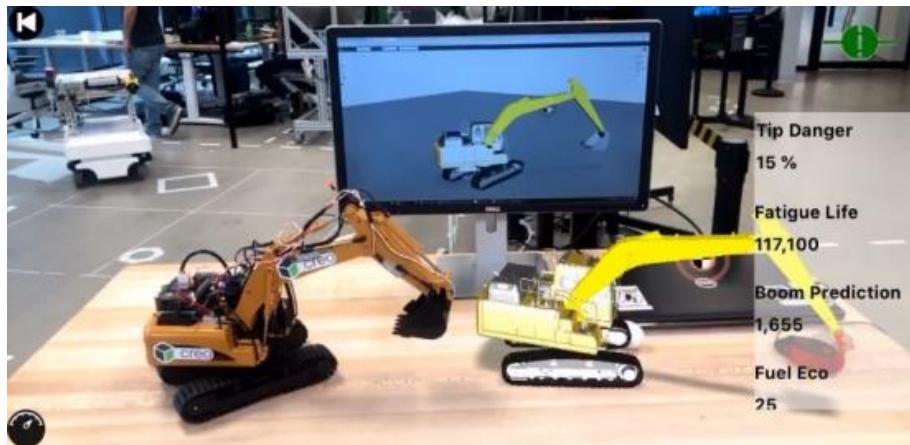
принимать решения на их основе. К сожалению, это не всегда так просто, как может показаться. Когда наборы данных становятся настолько большими и сложными, сделать выводы на их основе и внести улучшения становится затруднительно [6].



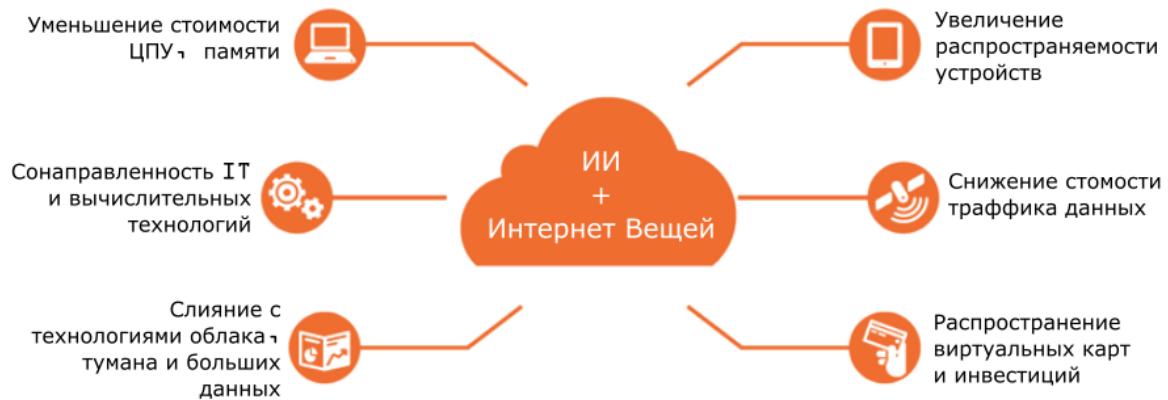
ЦИФРОВОЙ ДВОЙНИК (Digital Twin) – это виртуальная интерактивная копия реального физического объекта или процесса, которая помогает эффективно управлять им, оптимизируя бизнес-операции. Например, цифровой двойник завода позволяет моделировать расположение оборудования, перемещение сотрудников, рабочие процессы и внештатные ситуации. Именно интерактивность отличает понятие цифрового двойника от термина «информационная модель изделия» (ИМИ). В отличие от ИМИ, цифровой двойник не ограничивается сбором данных, полученных во время разработки и изготовления продукта, а продолжает собирать и анализировать информацию в течение всего жизненного цикла реального объекта, например, с помощью устройств Интернета вещей (IoT).

Таким образом, цифровой двойник можно рассматривать в качестве виртуального прототипа реального объекта или процесса, который содержит все данные о нем, включая историю и информацию о текущем состоянии. Интерактивный анализ этих данных с помощью технологий Big Data позволяет эффективно выполнять следующие важные управленческие функции:

- получение точной информации о производительности системы
- прогнозирование будущих состояний с помощью предиктивной аналитики
- удаленное управление объектом в режиме реального времени [7]



ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ находит свое применение во всех современных технологиях. В обсуждениях о развитии таких технологий часто упоминаются термины «искусственный интеллект» и «Интернет вещей». Многие уже используют оба понятия как синонимы, ведь эти две технологии часто взаимодополняют функционал друг друга. Но в таком случае необходимо поговорить о третьем понятии — искусственном интеллекте вещей, *Artificial Intelligence of Things (AloT)*.



Вместе две технологии создают интеллектуальные связанные системы, в которых ИИ выполняет роль «мозга», а Интернет вещей — «тела». IoT устройства собирают и передают данные от множества источников, поддерживая процесс обучения ИИ, чтобы тот знал, как правильно осуществлять автоматизацию нужных процессов. Благодаря искусственному интеллекту системы Интернета вещей могут обучаться и принимать решения по управлению данными и их анализу, что непременно увеличивает производительность [8].

ПРИМЕРЫ ИСПОЛЬЗОВАНИЯ ИНТЕРНЕТА ВЕЩЕЙ

МЕДИЦИНА

ADAMM — это носимый интеллектуальный монитор астмы, предназначенный для выявления симптомов приступа астмы до его начала, позволяющий владельцу управлять им до того, как приступ усугубится. Он вибрирует, чтобы уведомить человека, носящего его, о надвигающемся приступе астмы, а также может одновременно отправить текстовое сообщение назначенному опекуну.



Другие функции устройства включают в себя обнаружение ингалятора — устройство может обнаруживать и отслеживать использование ингалятора, если пациент не может вспомнить, использовали ли они его. ADAMM работает совместно с приложением и веб-порталом, помогая пациентам, страдающим астмой, устанавливать напоминания о приеме лекарств, просматривать данные с устройства и напоминать себе о плане лечения [9].

ПРОИЗВОДСТВО

Ярким примером применения Промышленного Интернета Вещей является проект компании Harley Davidson, которая производит знаменитые мотоциклы. Основной проблемой, с которой столкнулась компания, была медленная реакция на запросы потребителей в условиях возросшей конкуренции и ограниченная возможность совершенствования дилерами пяти выпускаемых моделей. В период с 2009 по 2011 год компания провела масштабную реконструкцию своих производственных площадок. В результате была создана единая сборочная площадка, выпускающая мотоциклы всех пяти моделей с возможностью их кастомизации, при этом заказчику предлагается выбор из более 1300 вариантов. В ходе всего производственного процесса используются специальные датчики. Каждый станок, каждая деталь имеет радиометку, которая однозначно идентифицирует изделие и его

производственный цикл. Данные от датчиков передаются в платформу обработки данных, выполняющую роль интеграционной шины для сбора данных с датчиков и различных информационных систем, как внутренних производственных и бизнес-систем компании Harley Davidson, так и информационных систем дочерних компаний.

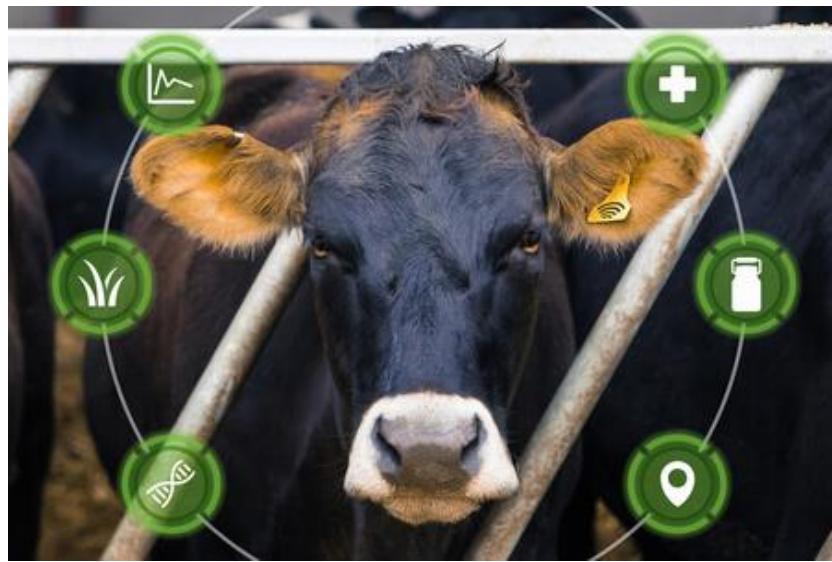


В результате компания Harley Davidson достигла весьма впечатляющих результатов:

- производственный цикл удалось сократить с 21 дня до 6 часов (каждые 89 секунд с конвейера сходит мотоцикл, полностью настроенный под своего будущего владельца);
- реализовано сквозное управление изделием на всем его жизненном цикле;
- стоимость акций компании выросла более чем в 7 раз: с уровня 10 долларов в 2009 году до 70 долларов в 2015 году [10].

СЕЛЬСКОЕ ХОЗЯЙСТВО

Только 15% территории Японии пригодны для ведения сельского хозяйства. Чтобы компенсировать скучные природные ресурсы, страна делает ставку на разработку умных технологий, которые позволяют повысить производительность труда фермеров. Автоматика, датчики и сложная техника используются во многих областях сельского хозяйства, в том числе и в животноводстве, которое пока развито очень слабо. С помощью интернета вещей фермеры следят за физическим состоянием коров и даже их настроением. Система, разработанная одной из крупнейших в Японии IT-корпораций Fujitsu, получила название GyuHo SaaS («шагающая корова» с японского) или Connected Cow («подключённые коровы»).



На животное надевается специальный браслет, который считает шаги, сделанные в течение дня. Данные об активности стада отправляются в облако, анализируются и передаются на смартфон или компьютер фермера. Информация обновляется каждый час, благодаря этому специалисты могут корректировать кормление, доение и сон животных. Заболевание у «подключённых коров» можно обнаружить на ранней стадии, ведь животное, которому нездоровится, будет двигаться меньше. Но главная задача умного браслета — вычислить благоприятный период для зачатия. Датчик позволяет сделать это с большой долей вероятности, так как в период течки число шагов, сделанных коровой, увеличивается в разы. В результате, по данным разработчиков системы, успешность искусственного оплодотворения с 44% вырастает до 90%. Также система предсказывает дату родов и позволяет следить за процессом дистанционно. Активно внедрять систему в Японии начали в 2013 году. К системе подключили около 40 тысяч коров. По информации Forbes, к 2017 году технология Fujitsu использовалась на 64 фермах в Японии, Корее, Польше, Румынии и Турции [11].

УМНЫЙ ГОРОД

Амстердамская инициатива «Умный город», начатая в 2009 году, в настоящее время включает более 170 проектов, совместно разработанных местными жителями, правительством и бизнесом. Эти проекты разрабатываются на основе беспроводных устройств, которые служат для повышения способности города принимать решения в режиме реального времени. Целью проектов является сокращение трафика, экономия энергии и повышение общественной безопасности. Чтобы стимулировать усилия местных жителей, город ежегодно проводит соревнование «Умный Город», принимая предложения по улучшениям, которые вписываются в структуру города. Примером приложения, разработанного жителями города, является

«MobyPark», которое позволяет владельцам парковочных мест сдавать их в аренду людям за определенную плату. Данные, полученные из этого приложения, затем могут быть использованы городом для определения спроса на парковку и транспортных потоков в Амстердаме.



Кроме того, в ряде домов установлены интеллектуальные счетчики энергии, а также предусмотрены стимулы для тех, кто активно сокращает потребление энергии. Другие инициативы включают гибкое уличное освещение (smart lighting), которое позволяет муниципалитетам контролировать яркость уличных фонарей, и интеллектуальное управление дорожным движением, где движение контролируется в режиме реального времени, и информация о текущем времени движения по определенным дорогам транслируется, чтобы позволить автомобилистам определить лучшие маршруты [12].

ЗАДАЧИ

ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ

Ардуино (англ. Arduino) – это популярная и всемирно известная платформа для получения базовых навыков по конструированию электронных систем. Контроллер Ардуино представляет собой плату с процессором и памятью.

На плате есть контакты для подключения датчиков и исполнительных устройств. В процессор Arduino можно загрузить программу, которая будет управлять всеми этими устройствами по заданному алгоритму.

«Умные» системы – это системы, управление которыми осуществляется в автоматическом или автоматизированном режиме (с использованием режима диалога) с помощью смартфона через платформу интернета вещей.

ЙоТик® 32 – это контроллер, который можно программировать в среде Arduino IDE, созданный на базе микроконтроллера ESP32. С помощью этого контроллера можно научиться конструировать «умные» системы.

Узнать больше можно на сайте <http://iotik.ru>

ЗАДАНИЕ 1. Вывод текста на экран монитора

Цель:

Понять назначение, основные функции и принцип работы контроллера, закрепить ключевые понятия и научиться читать скетчи или создавать программы для вывода текста на экран монитора компьютера.

Ключевые слова и понятия:

Контроллер, программа, вывод текста.

Результат:

Вывод текста на экран монитора компьютера.

СПРАВКА

Скетч – это программа, написанная для платформы Arduino и имеющая определенную структуру. Скетч обязательно содержит две функции: функцию **setup** (метка – установка, инициализация) и функцию **loop** (обозначение главного цикла программы).

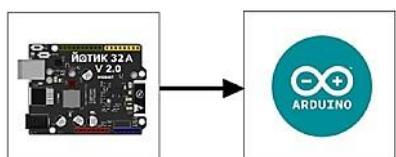
Оборудование:

Контроллер ЙоТик® 32 А v2.0.

ИНСТРУКЦИИ

Основное задание

1. Изучить работу контроллера.
2. Собрать схему.



3. Создать программу (или воспользоваться скетчем), которая выводит текст на экран монитора компьютера.
4. Запрограммировать контроллер, проверить работоспособность программы.
5. Изменить текст и проверить правильность работы программы.
6. Создать свою (пользовательскую) библиотеку скетчей.
7. Сохранить программу, имя файла латинскими буквами.



Скетч <http://books.mgbot.ru/sketches.zip>

Скетч

```
void setup() { //функция настроек
  Serial.begin(115200); //скорость передачи данных
  Serial.print("Как "); //вывести данные, написанные в кавычках
```

```
Serial.print("дела?");  
Serial.println("Привет!"); // вывести данные в одну строку  
}  
void loop() {} // главный цикл
```

ЗАДАНИЕ 2. Ввод данных в режиме диалога

Цель:

Освоить основные понятия и научиться использовать и редактировать скетчи под однотипные задачи.

Ключевые слова и понятия:

Диалог, контроллер, программа.

СПРАВКА

Режим диалога предполагает, что ввод данных осуществляется по запросу.

Результат:

Программа для выполнения операций сложения, вычитания, умножения и деления с вводом данных по запросу (в режиме диалога).

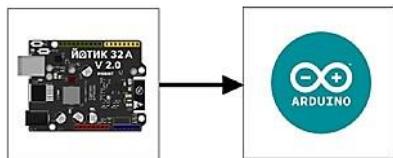
Оборудование:

Контроллер ЙоТик® 32 А v2.0.

ИНСТРУКЦИИ

Основное задание

1. Составить программу (использовать скетчи) для вычисления суммы, ввод которых осуществляется в режиме диалога (по запросу).
2. Запрограммировать контроллер.
3. Проверить работоспособность, провести несколько опытов с разными числами (слагаемыми).



Дополнительное задание

1. Проанализировать скетч и отредактировать программу для реализации всех простых математических операций (умножения, вычитания, деления).
2. Провести опыты, проверить результаты.
3. Сохранить в своей (пользовательской) библиотеке скетчей.

Скетч

```
// В мониторе порта выберите параметр "Нет конца строки"  
int inc1, inc2; // переменные для двух слагаемых  
int i; // флаг  
void setup() {  
  Serial.begin(115200);  
  Serial.println("Введите первое слагаемое");
```

```
}

void loop()
{
    if (Serial.available() > 0 and i == 0) { // если данные пришли и флаг сброшен
        inc1 = Serial.parseInt(); // то введенное число – первое слагаемое
        Serial.println("Первое слагаемое: ");
        Serial.println(inc1);
        Serial.println("Введите второе слагаемое");
        i = 1;
    }

    if (Serial.available() > 0 and i == 1) { // если данные пришли и флаг поднят
        inc2 = Serial.parseInt(); // то введенное число – второе слагаемое
        Serial.println("Второе слагаемое: ");
        Serial.println(inc2);
        Serial.println("Сумма равна: ");
        Serial.println(inc1 + inc2);
        Serial.println("-----");
        Serial.println("Введите первое слагаемое");
        i = 0;
    }
}
```

ЗАДАНИЕ 3. Управление встроенным светодиодом

Цель:

Закрепить знания о принципе работы и основных возможностях контроллера, научиться понимать (создавать) скетч для управления встроенным светодиодом.

Ключевые слова и понятия:

Контроллер, светодиод.

Результат:

Включение / выключение светодиода по определенному времени.

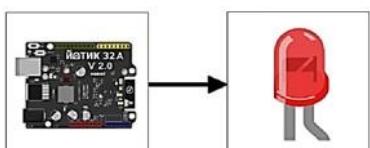
Оборудование:

Контроллер ЙоТик® 32 А v2.0.

ИНСТРУКЦИИ

Основное задание

1. Прочитать справку о светодиоде, дополнительно – найти информацию в интернете.
2. Создать программу (использовать скетч) для включения /выключения светодиода на контроллере по времени.
3. Запрограммировать контроллер, отладить.
4. Создать программу (использовать скетч) для реализации различных временных задержек при включении/выключении светодиода по времени.
5. Собрать схему и проверить работоспособность.
6. Провести серию опытов.



Дополнительное задание

1. Создать программы (использовать скетчи) для реализации различных приемов в создании световых эффектов.
2. Ввести программу в контроллер, отладить.
3. Провести серию опытов и результаты занести в таблицу.

СПРАВКА

Диод – это электронный (полупроводниковый) элемент, который пропускает электрический ток в одном направлении.

Светодиод (LED) – светоизлучающий диод – это диод, который преобразует электрический ток в свет.

Обозначение светодиода в схеме



Скетч

```
#define LED 18 // pin 18
void setup() {
  pinMode(LED, OUTPUT); // pin на выход
```

```
}

void loop()
{
    digitalWrite(LED, HIGH); // включить светодиод
    delay(1000); // задержка 1 сек
    digitalWrite(LED, LOW); // выключить светодиод
    delay(1000); // задержка 1 сек
}
```

ЗАДАНИЕ 4. Текущее время в интернете

Цель:

Закрепить знания по возможностям контроллера и узнать точное время, используя NTP сервер.

Ключевые слова и понятия:

Сетевой протокол, NTP-сервер, контроллер, Wi Fi соединение.

Результат:

Получение сведений о точном времени с NTP сервера.

Оборудование:

1. Наличие Wi Fi соединения.
2. Контроллер ЙоТик® 32 А v2.0.

СПРАВКА

NTP (англ. Network Time Protocol – протокол сетевого времени) – **сетевой протокол** для синхронизации внутренних часов компьютера (контроллера).

Сетевой протокол – набор правил, позволяющий осуществлять соединение и обмен данными между двумя и более включёнными в сеть устройствами.

ИНСТРУКЦИИ

Основное задание

1. Собрать по образцу и проверить работоспособность.



2. Написать программу (воспользоваться скетчем) для получения сведений о текущем времени на NTP-сервере через Wi Fi.
3. Запрограммировать контроллер и отладить программу.
4. Определить время по всем часовым поясам и для конкретного часового пояса.
5. Записать показания в таблицу (по образцу) – города подобрать самостоятельно:

Город	Часовой пояс	Текущее время
Санкт-Петербург	GMT + 03:00	
	GMT + 02:00	
	GMT + 01:00	
	GMT + 00:00	
	GMT + 04:00	
	GMT + 05:00	
	GMT + 07:00	

Дополнительные задания

1. Создать будильник, таймер, секундомер с использованием светодиода.
2. Создать программу для вывода на экран монитора компьютера требуемого часового пояса (запрашиваем, например, город, Лондон – выводится его текущее время).
3. Сделать программу для следующих городов: Санкт-Петербург, Лондон, Нью-Йорк, Владивосток, Новосибирск.

Скетч

```
#include "TOD.h"

template<class T> inline Print &operator <<(Print &obj, T arg) {
    obj.print(arg);
    return obj;
}

TOD RTC;

uint8_t lastminute = 0;
uint8_t lastsecond = 45;
char printbuffer[50];
bool TODvalid = false;
char ssid[] = "Wi-Fi"; // Введите логин от WiFi
char password[] = "Pass"; // Введите пароль от WiFi

void setup()
{
    Serial.begin(115200);
    if(RTC.begin(ssid, password))TODvalid = true;
    lastminute = RTC.minute();
}

void loop()
{
    if(RTC.second() != lastsecond && TODvalid)// выводим время каждую секунду
    {
        lastsecond = RTC.second();
        sprintf(printbuffer, " UTC Time:%02i:%02i:%02i.%03i\n", RTC.hour() + 3, RTC.minute(),
        RTC.second(), RTC.millis());
        Serial << printbuffer;
    }
}
```

```
if (RTC.minute() == lastminute + 3 && TODvalid) // проверка каждые 3 минуты
{
    lastminute = RTC.minute();
    RTC.begin(ssid, password);
}
}
```

СПРАВКА

Понятие **часовой пояс** имеет **два основных значения**:

Географический часовой пояс – условная полоса на земной поверхности шириной ровно 15° ($\pm 7,5^\circ$ относительно среднего меридиана). Средним меридианом нулевого часового пояса считается гринвичский меридиан.

Административный часовой пояс – участок земной поверхности, на котором в соответствии с некоторым законом установлено определённое официальное время.

ЗАДАНИЕ 5. Количество подписчиков в инстаграм

Цель:

Научиться использовать контроллер для определения количества подписчиков в социальных сетях на примере инстаграма.

Ключевые слова и понятия:

Инстаграм, контроллер, Wi-Fi соединение.

Результат:

Определение количества подписчиков в инстаграме.

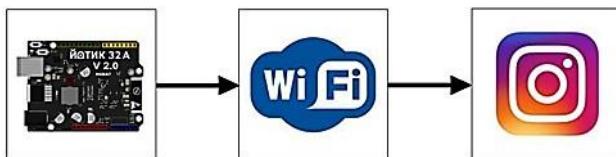
Оборудование:

1. Наличие Wi Fi соединения.
2. Контроллер ЙоТик® 32 A v2.0.
3. Аккаунт в инстаграм.

ИНСТРУКЦИИ

Основное задание

1. Найти в интернете сведения об инстаграме: что это такое, какие есть возможности.
2. Собрать схему.



3. Создать диалоговую программу (использовать скетч) для подсчета количества подписчиков в инстаграме.
4. Вывести данные о количестве подписчиков в инстаграме на монитор компьютера.
5. Сохранить скетч в пользовательской библиотеке.

Скетч

```
#include <InstagramStats.h>
#include <WiFi.h>
#include <WiFiClientSecure.h>
#include <JsonStreamingParser.h>
char ssid[] = "MGBot"; // логин вай-фай
char password[] = "Terminator812"; // пароль вай-фай
WiFiClientSecure client;
InstagramStats instaStats(client);
unsigned long delayBetweenChecks = 300000; // время между проверками
unsigned long whenDueToCheck = 0;
```

```

String userName = "instaccount"; // логин инстаграма для проверки
void setup()
{
    // instaStats._debug = true;
    Serial.begin(115200);
    Serial.print("Connecting WiFi: ");
    Serial.println(ssid);
    WiFi.mode(WIFI_STA);
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED)
    {
        Serial.print(".");
        delay(500);
    }
    Serial.println("");
    Serial.println("WiFi connected");
    Serial.print("IP address: ");
    Serial.println(WiFi.localIP());
}
void getInstagramStatsForUser()
{
    Serial.println("Instagram statistics for: " + userName); // выводим информацию о
    количестве подписчиков в Инстаграм
    InstagramUserStats response = instaStats.getUserStats(userName);
    Serial.print("Number of followers: ");
    Serial.println(response.followedByCount);
    Serial.println();
}
void loop()
{
    unsigned long timeNow = millis(); // выводим информацию каждые 5 минут
    if ((timeNow > whenDueToCheck))
    {
        getInstagramStatsForUser();
        whenDueToCheck = timeNow + delayBetweenChecks;
    }
}

```

ЗАДАНИЕ 6. Измерение освещенности

Цель:

Освоить принцип работы датчика освещенности и научиться проводить опыты по измерению освещенности в разных условиях.

Ключевые слова и понятия:

«Умная» система, плата расширения, датчик освещенности, смартфон, платформа интернета вещей Blynk.

Результат:

Понимание принципа работы датчика освещенности и организация проведения опытов по измерению величины освещенности в разных условиях.

Оборудование:

1. Контроллер ЙоТик® 32 A v2.0.
2. Плата расширения MGB-DA20.
3. Плата расширения MGB-D10.
4. Датчик освещенности MGS-L75.
5. Доступ к платформе Blynk.

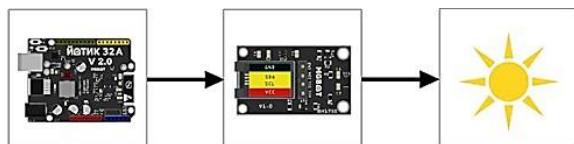
СПРАВКА

Плата расширения имеет несколько слотов для подключения разных датчиков, через нее осуществляется подключение к контроллеру.

ИНСТРУКЦИИ

Основное задание

1. Изучить принцип работы датчика освещенности.
2. Освоить основные единицы измерения освещенности.
3. Собрать схему для измерения текущей освещенности.



4. Создать программу (или использовать скетчи), запрограммировать контроллер, отладить.
5. Настроить смартфон для получения данных через интернет платформу Blynk.
6. Сохранить скетч в пользовательской библиотеке.
7. Провести серию опытов по измерению освещенности в разных условиях.
8. Заполнить таблицу (Лк):

Условия освещенности	Освещенность (Лк)
1. Искусственное освещение	
2. Естественное освещение:	
А) Пасмурная погода	
Б) Солнечная погода	
В) Ночное время	
Г) Дневное время	

Дополнительное задание

- Составить программу (воспользоваться скетчом) для определения освещенности окружающей среды (кабинета) в автоматическом режиме через платформу Blynk.
- Внести (скопировать) значение освещенности за 3–5 дней в таблицу (по времени), используя возможности электронной таблицы.
- Вычислить среднее значение с использованием функции в электронной таблице.
- Сравнить полученные значения освещенности и записать выводы:

Время	Даты					Среднее значение
14:00						
16:00						
18:00						

СПРАВКА

Что такое освещенность, в каких единицах она измеряется

- Чем больший световой поток попадает на поверхность освещаемого предмета, тем лучше этот предмет видно.
- Световым потоком** можно назвать количество света, излучаемого светильником или отдельной лампочкой.
- Освещенность** это величина, равная световому потоку, падающему на единицу площади освещаемой поверхности.

Освещенность измеряется в люксах (Лк).

Скетч считывания датчика освещенности

```
#include <Wire.h> // библиотека для I2C
#include <BH1750FVI.h> // библиотека для ДО
BH1750FVI LightSensor_1; // определение ДО
void setup()
```

```

{
  Serial.begin(115200);
  LightSensor_1.begin();
  LightSensor_1.setMode(Continuously_High_Resolution_Mode);
}
void loop()
{
  float l = LightSensor_1.getAmbientLight(); // считывание датчика освещенности
  Serial.println("Освещенность равна: " + String(l, 1) + " lx");
  delay(250);
}

```

Скетч передачи данных на платформу Blynk

```

#include <Wire.h> // библиотека для I2C
#include <BH1750FVI.h> // библиотека для ДО
BH1750FVI LightSensor_1; // определение ДО
#define BLYNK_PRINT Serial
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h> // конфигурация блинка
char ssid[] = "login"; // Логин Wi-Fi
char pass[] = "pass"; // Пароль от Wi-Fi
char auth[] = "token"; // Токен
IPAddress blynk_ip(139, 59, 206, 133); // конфигурация блинка
#define UPDATE_TIMER 1000
BlynkTimer timer_update; // настройка таймера для обновления данных с сервера Blynk
void setup()
{
  Serial.begin(115200);
  delay(512);
  LightSensor_1.begin();
  LightSensor_1.setMode(Continuously_High_Resolution_Mode);
  Blynk.begin(auth, ssid, pass, blynk_ip, 8442); // подключение к серверу Blynk
  timer_update.setInterval(UPDATE_TIMER, readSendData); // включаем таймер обновления
  данных
}

```

```
}

void readSendData() {
    float l = LightSensor_1.getAmbientLight(); // считывание датчика освещенности
    Blynk.virtualWrite(V0, l); delay(25); // Отправка данных датчика Освещенности (V0)
}

void loop()
{
    Blynk.run(); // запуск Blynk
    timer_update.run(); // запуск таймера
}
```

ЗАДАНИЕ 7. Измерение температуры воздуха

Цель:

Понять принципы работы датчика температуры, влажности воздуха, атмосферного давления, освоить ключевые понятия по использованию разной шкалы измерения температуры и закрепить знания по сборке схемы для измерения температуры.

Ключевые слова и понятия:

«Умная» система, температурная шкала, плата расширения, датчик температуры, влажности и атмосферного давления.

Результат:

Освоена работа датчика и получены данные в соответствии с планом работы.

Оборудование:

1. Контроллер ЙоТик® 32 А v2.0.
2. Плата расширения MGB-DA20.
3. Плата расширения MGB-D10.
4. Датчик температуры, влажности воздуха и атмосферного давления MGS-THP80.

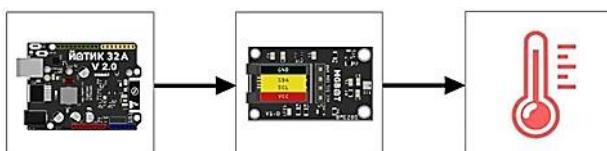
СПРАВКА

Для измерения температуры применяются три основные шкалы (по фамилиям ученых): Цельсия, Кельвина и Фаренгейта. Для перевода из одной шкалы в другую существуют формулы.

ИНСТРУКЦИИ

Основное задание

1. Изучить назначение и принципы работы датчика температуры, влажности и атмосферного давления.
2. Собрать схему для измерения температуры в градусах Цельсия.



3. Создать программу (воспользоваться скетчем) для измерения температуры (шкала Цельсия).
4. Запрограммировать контроллер, отладить программу и проверить работоспособность.
5. Настроить смартфон для управления измерением температуры через платформу интернета вещей Blynk.
6. Отредактировать программу для преобразования температуры из шкалы Цельсия в шкалу по Кельвину и / или Фаренгейту.
7. Настроить смартфон для управления измерением температуры через платформу интернета вещей Blynk.

8. Найти в интернете формулу перевода из одной шкалы в другую.
9. Снять показания температуры за 3–5 дней (в определенное время) в кабинете или на улице.
10. Заполнить таблицу показаний температуры за 3–5 дней и перевод из шкалы Цельсия в другие (Кельвина и Фаренгейта) – использовать электронную таблицу.

Даты (Время 12:00 дня)	Температура по шкале Цельсия	Температура по шкале Кельвина	Температура по шкале Фаренгейта

11. Сохранить скетч в пользовательской библиотеке.

Дополнительное задание

1. Проанализировать недельный график температур в кабинете (один раз в час), полученный с помощью датчика и термометра (использовать шкалу Цельсия).
2. Проанализировать недельный график температур на улице (один раз в час), полученный с помощью датчика и наружного термометра.
3. Сделать выводы: есть ли разница в показаниях температур, полученных с помощью датчика от температур, полученных с помощью термометра (записать ниже).

Вывод (словесное описание): _____

Скетч

```
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BME280.h>
Adafruit_BME280 bme280;
void setup() {
  Serial.begin(115200);
  bool bme_status = bme280.begin();
  if (!bme_status)
    Serial.println("Could not find a valid BME280 sensor, check wiring!");
}
```

```
void loop() {  
    float t = bme280.readTemperature();  
    Serial.println("Air temperature = " + String(t, 1) + " *C");  
    delay(250);  
}
```

ЗАДАНИЕ 8. Измерение влажности воздуха

Цель:

Закрепить знания по основам работы с датчиком температуры, влажности воздуха и атмосферного давления, совершенствовать работу по сборке схем.

Ключевые слова и понятия:

«Умная» система, датчик температуры, влажности воздуха и атмосферного давления, контроллер, гигрометр.

Результат:

Будет освоена работа с датчиком, собрана схема и проведены все измерения в соответствии с заданием.

Оборудование:

1. Контроллер ЙоТик® 32 A v2.0.
2. Плата расширения MGB-DA20.
3. Плата расширения MGB-D10.
4. Датчик температуры, влажности воздуха и атм. давления MGS-THP80.

СПРАВКА

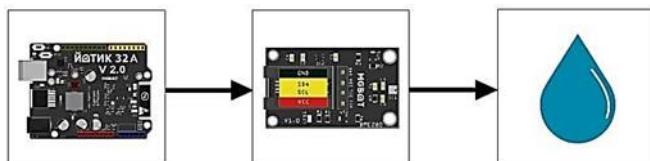
Влажность воздуха измеряется в процентах (%).

При этом: чем выше влажность, тем больше величина влажности в процентах, максимальная величина 100%.

ИНСТРУКЦИИ

Основное задание

1. Собрать схему для измерения влажности воздуха.



2. Составить программу (воспользоваться скетчем) для измерения влажности воздуха так, чтобы измерения производились с интервалом один час*.

3. Запрограммировать контроллер, отладить программу.
4. Настроить смартфон для управления измерений влажности воздуха через платформу интернета вещей Blynk.
5. Провести опыты в помещении: в течение 5 дней записывать показания комнатного гигрометра в определенное время.
6. Через 5 дней сравнить показания гигрометра и показаниями влажности воздуха, полученной с помощью датчика влажности.
7. Сохранить скетч в пользовательской библиотеке.

* Можно сделать в течение двух-трех учебных занятий.

Запиши вывод: _____

Скетч

```
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BME280.h>
Adafruit_BME280 bme280;
void setup() {
  Serial.begin(115200);
  bool bme_status = bme280.begin();
  if (!bme_status)
    Serial.println("Could not find a valid BME280 sensor, check wiring!");
}
void loop() {
  float h = bme280.readHumidity();
  Serial.println("Air humidity = " + String(h, 1) + " %");
  delay(250);
}
```

ЗАДАНИЕ 9. Измерение температуры почвы

Цель:

Познакомиться с назначением и принципами работы датчика для измерения температуры и влажности почвы и научиться использовать или создавать скетч для измерения температуры почвы, производить измерения.

Ключевые слова и понятия:

«Умная» система, датчик для измерения температуры и влажности почвы, контроллер, плата расширения.

Результат:

Усвоены принципы работы датчика для измерения температуры почвы, правильно собрана схема и проведены все измерения в соответствии с планом работы.

Оборудование:

1. Контроллер ЙоТик® 32 А v2.0.
2. Плата расширения MGB-DA20.
3. Датчик температуры и влажности почвы MGS-TH50.

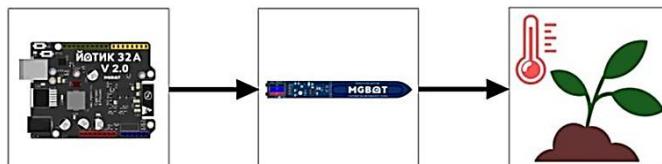
СПРАВКА

Температура почвы измеряется в градусах по шкале Цельсия.

ИНСТРУКЦИИ

Основное задание

1. Изучать принципы работы датчика для измерения влажности почвы.
2. Освоить порядок постановки опытов.
3. Собрать схему для измерения температуры почвы.



4. Составить программу (использовать скетч) для измерения температуры почвы.
5. Запрограммировать контроллер.
6. Проверить работоспособность программы.
7. Сохранить скетч в пользовательской библиотеке.
8. Самостоятельно поставить любой из опытов по описанию.
9. Самостоятельно оформить таблицу и сделать выводы.

Описание опытов:

Сравнение показаний температуры при сухой, сырой и только что поливной почве (вместо почвы для эксперимента можно взять другие безопасные материалы, например, крахмал).

Дополнительное задание

Описание опытов:

Сравнение и анализ температур почвы и температур воздуха при кабинетных и уличных измерениях.

Скетч

```
#define SOIL_TEMPERATURE 35
void setup()
{
    Serial.begin(115200);
    delay(512);
}

void loop()
{
    float adc1 = analogRead(SOIL_TEMPERATURE);
    float t = ((adc1 / 4095.0 * 5.0) - 0.3) * 100.0;
    Serial.print("Soil temperature = ");
    Serial.println(t);
}
```

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. <https://mcs.mail.ru/blog/iot-dlya-umnyh-chasov-iot-umnyh-stankov-internet-veshchey>
2. <https://nplus1.ru/material/2019/12/20/iot-arctic>
3. <https://www.euromobile.ru/solutions/umnyj-gorod/internet-veshchey-v-umnom-gorode/>
4. <https://habr.com/ru/company/cloud4y/blog/467711/>
5. <https://www.bigdataschool.ru/wiki/большие-данные-big-data>
6. <https://www.purestorage.com/ru/knowledge/big-data/internet-of-things-and-big-data.html>
7. <https://www.bigdataschool.ru/blog/digital-twin-plm-iot-big-data.html>
8. <https://iotji.io/ru/aiot-rol-iskusstvennoho-intellekta-v-internete-veshchey/>
9. <https://internetandthings.com/ru/10-examples-medicine/>
10. <https://www.it.ua/ru/knowledge-base/technology-innovation/promyshlennyj-internet-veschey>
11. <https://politsib.ru/news/29197-umnoe-selskoe-hozajstvo-cetyre-primera-iz-raznyh-stran>
12. https://ru.wikipedia.org/wiki/Умный_город

