

**Алексеевский Петр Иванович,**

ассистент кафедры информатики, информационных технологий и методики обучения информатике, Уральский государственный педагогический университет; 620075, г. Екатеринбург, ул. К. Либкнехта, 9; e-mail: u@puuu.ru.

## **ОБУЧЕНИЕ СТУДЕНТОВ ПРОГРАММИРОВАНИЮ С ИСПОЛЬЗОВАНИЕМ ОТЛАДОЧНЫХ КОМПЛЕКТОВ STM32 DISCOVERY**

**КЛЮЧЕВЫЕ СЛОВА:** программирование; встраиваемые устройства; интернет; микроконтроллеры; информационные технологии; методика информатики в вузе; методика преподавания информатики.

**АННОТАЦИЯ.** В данной статье описывается проблема дополнения содержания курса программирования в вузах для отражения современных востребованных на рынке технологий. Отмечено, что повсеместное распространение устройств, составляющих так называемый «интернет вещей», подразумевает необходимость обучения разработке программного обеспечения для этого класса устройств, чему препятствует недостаточная разработанность содержания курса программирования применительно к данному направлению. Рассмотрен вариант решения данной проблемы, предполагающий дополнение курса программирования изучением основных особенностей разработки программного обеспечения для встраиваемых устройств и интернета вещей. Поскольку изучение программирования предполагает наличие практического компонента, рассмотрены примеры аппаратных платформ, которые могут быть использованы в качестве примеров встраиваемых устройств в процессе обучения программированию. Приведены основные направления использования этих платформ в образовательном процессе. Предложен вариант использования отладочного комплекта STM32 F7 Discovery в качестве целевой платформы для рассмотрения особенностей разработки программного обеспечения для встраиваемых систем в процессе обучения программированию, а также для расширения существующих лабораторных заданий по программированию. Приведены аспекты разработки программного обеспечения, рассмотрение которых возможно на примере этой платформы, и программно-аппаратные средства, позволяющие осуществлять разработку. Отмечены основные темы, изучение которых представляется более наглядным при использовании микроконтроллеров и вычислительных систем на их основе, среди которых низкоуровневое управление вычислительными ресурсами и внешними устройствами, такими как устройства хранения информации и коммуникационные интерфейсы. В связи с необходимостью управления множеством периферийных устройств одновременно в составе единой системы реального времени, отмечена возможность коллективного выполнения лабораторных работ студентами в процессе усвоения учебного материала по программированию.

**Alexeevskiy Petr Ivanovich,**

Assistant Lecturer, Department of Informatics, Computer Technology and Methods of Teaching Informatics, Ural State Pedagogical University, Ekaterinburg, Russia.

## **TRAINING STUDENTS PROGRAMMING USING STM32 DISCOVERY DEVELOPMENT KITS**

**KEYWORDS:** programming; embedded app; add-in; Internet; microcontroller; information technologies; methods of teaching Computer Science at school; methods of teaching Computer Science at university; methods of teaching Computer Science.

**ABSTRACT.** This article explores the problem of extending the curriculum content for the programming course to reflect modern technologies in demand. Options for the programming course extension using the software development for the Internet of Things devices are evaluated. Examples of hardware platforms allowing for the programming training are provided. The use of the STM32 F7 Discovery kit is proposed as the target platform. It is noted that the wide spreading of devices forming a so-called «Internet of Things» implies the need to teach students the software programming for this kind of devices, which is hindered by the insufficient development of the curriculum content in that field. A solution is proposed, suggesting the programming course extension with basics of software development for embedded devices and the Internet of Things. As learning of software programming implies a field research part, examples of hardware platforms which can be used in the process are explored. The main ways to use these platforms in the education process are provided. The use of an STM32 F7 Discovery kit is proposed as the target platform for the embedded software programming research, as well as an extension for existing field research tasks in teaching. Software development aspects which are possible to explore using this platform are presented, along with hardware and software tools to enable the development. Main topics which are visualised through the use of microcontrollers and MCU-based computer systems are listed, including low-level management of computation resources and external devices, such as storage and communication interfaces. Due to the need to control multiple devices at the same time in a self-contained real time environment, a possibility is presented for a group research in the process of learning the software programming.

### **Постановка и актуальность проблемы**

**Р**азвитие вычислительной техники, электроники и глобальных сетей

привело к возникновению такого явления, как интернет вещей (Internet of Things, IoT). Идея объединения обычной бытовой техники и некоторых электроприборов в вычисли-

тельную сеть с возможностью автоматизированного управления была предложена ещё в 1980-х годах, но с развитием встраиваемых вычислительных устройств в начале XXI века появилась возможность и ее практической реализации. С появлением подобных устройств возникла и необходимость в разработке программного обеспечения для управления подключенной полезной нагрузкой.

По состоянию на 2017 год технология IoT занимает ведущее положение среди технологий, влияющих на бизнес-модели компаний, представляющих IT-индустрию [4]. Востребованность данной технологии позволяет сделать вывод о целесообразности обучения программированию для IoT-устройств в рамках процесса профессиональной подготовки IT-специалистов.

Курс программирования для студентов вузов в настоящее время ориентирован преимущественно на разработку программного обеспечения для настольных персональных компьютеров, в том числе для серверных платформ. Современное оборудование такого рода отличается высокой производительностью и значительным объемом вычислительных ресурсов, что дает возможность не обращать внимания на недостатки реализуемых в процессе обучения алгоритмов. Рассмотрение особенностей разработки программного обеспечения для встраиваемых и портативных устройств при этом выделяется в отдельные дисциплины, либо вовсе исключается из учебной программы.

Противоречие между востребованностью IT-специалистов, способных осуществлять разработку для IoT-устройств, и недостаточной разработанностью учебно-методических материалов, используемых в процессе их профессиональной подготовки, возможно разрешить путем расширения содержания курса программирования для студентов вузов подходами к разработке встраиваемого ПО.

#### **Особенности обучения программированию для встраиваемых устройств**

Программное обеспечение для встраиваемых устройств имеет ряд особенностей, не встречающихся в прикладном ПО для персональных компьютеров. В силу специфики применяемого оборудования такое ПО занимает промежуточное положение между прикладным и системным программным обеспечением. В то же время используемое оборудование накладывает определенные ограничения на используемые в программном обеспечении алгоритмы.

Характеристики встраиваемых устройств могут существенно отличаться от таковых у вычислительных устройств общего назначения. Поскольку области применения встраи-

ваемых устройств предполагают максимальное уменьшение размеров, а также снижение энергопотребления и стоимости, вычислительные возможности встраиваемых устройств оказываются ниже, чем у персональных компьютеров, планшетных компьютеров, смартфонов и т. п. При этом элементная база подобных устройств зачастую предполагает наличие специализированных компонентов, более эффективно решающих ряд типовых задач для данного класса оборудования.

Рассмотрение подходов к разработке программного обеспечения для встраиваемых устройств должно опираться на примеры программно-аппаратных платформ и их возможности. Среди современных комплектов оборудования, используемых для обучения непосредственно программированию встраиваемых устройств, можно выделить следующие платформы:

- семейство отладочных плат Arduino и построенные на его базе комплекты;
- одноплатные компьютеры Raspberry Pi или аналогичные;
- комплекты Lego Mindstorms;
- платформа BBC MicroBit;
- семейство отладочных плат STM32 Nucleo;
- семейство отладочных плат STM32 Discovery.

Перечисленные платформы отличаются набором возможностей, такими как способы отладки, интерфейсы подключения к основному вычислительному узлу (микропроцессору или микроконтроллеру) внешних устройств, производительность и энергопотребление.

Платформа Arduino [10] (за исключением Arduino Due, Tre и Yun) базируется на восьмиразрядных микроконтроллерах архитектуры Atmel AVR и позиционируется как универсальная платформа начального уровня для обучения программированию микроконтроллеров. Она используется в качестве основной (и часто — единственной) платформы в различных учебных заведениях среднего и высшего образования в рамках специализированных курсов (в основном — робототехники) [2]. Тем не менее, применимость Arduino в рамках общего курса программирования ограничивается такими недостатками, как низкая производительность и отсутствие каких-либо встроенных в основную плату периферийных устройств.

Одноплатные компьютеры Raspberry Pi [14] создавались как дешевое, но функциональное средство для обучения информатики. Они построены на базе системы на кристалле (SoC) Broadcom BCM2835/6/7 (в зависимости от модели) и представляют собой вычислительную систему, обладающую возможностями компьютеров общего

назначения. Данная платформа является в определенной степени самодостаточной (разработка ПО может осуществляться непосредственно на Raspberry Pi), высокопроизводительной и при этом имеет набор внешних микропроцессорных интерфейсов, таких как GPIO, I<sup>2</sup>C, I<sup>2</sup>S и SPI.

Платформа Raspberry Pi позволяет использовать для разработки ПО любой из основных языков программирования (C, C++, Fortran, Java, Python и др.) и имеет те же возможности отладки, что и настольные компьютеры общего назначения. В то же время, для эффективного использования данной платформы в качестве средства обучения программированию требуется набор внешних периферийных устройств, сопоставимый с таковым у настольных компьютеров. Тем не менее, Raspberry Pi часто применяется в качестве основы для IoT [7] и в робототехнике [2; 8].

Комплекты Lego Mindstorms [13] позиционируются как средство для обучения робототехнике [2]. В этой области они занимают лидирующее положение (до 80% рынка обучающих конструкторов). Однако, ввиду специфической направленности данной платформы, набор внешних интерфейсов ограничен (поддерживается только I<sup>2</sup>C в режиме ведущего устройства).

Платформа BBC Micro:bit [12] — решение, ориентированное преимущественно на школьный курс обучения информатике. Данная платформа поддерживает разработку на различных языках программирования, таких как Python, Scratch, JavaScript, имеет набор встроенных периферийных устройств (LED-матрица, акселерометр, компас) и внешние интерфейсы I<sup>2</sup>C, SPI и GPIO.

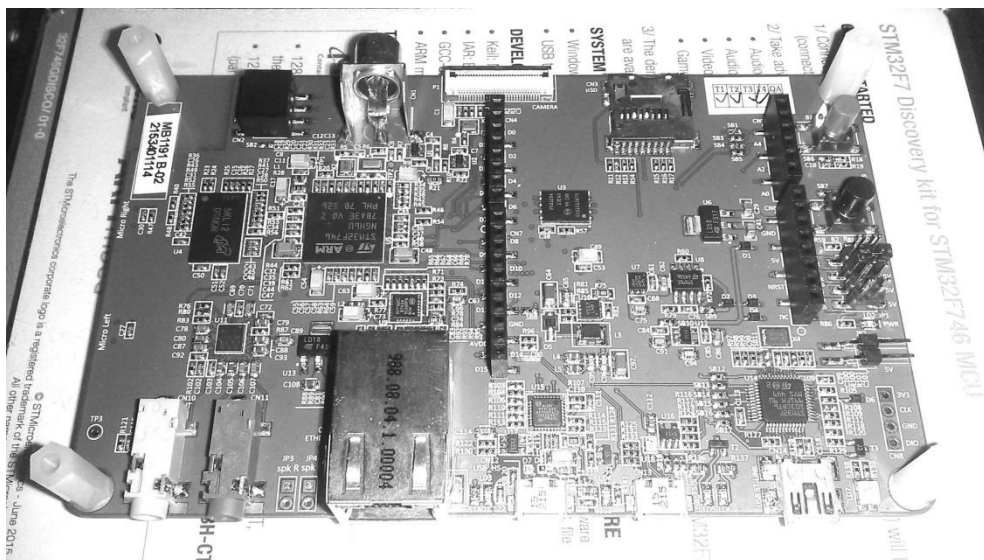
Отладочные платы STM32 Nucleo [15] предназначены для прототипирования устройств на базе микроконтроллеров семейства STM32. Как правило, данные платы имеют разъемы, совместимые с модулями расширения Arduino. Тем не менее, встроенные периферийные устройства на таких платах обычно отсутствуют. Отладка программы на таких платах может осуществляться через встроенный программатор, являющийся вариантом внутрисхемного программатора ST-Link/V2. Разработка программ возможна на различных языках программирования с использованием таких средств, как среда разработки SW4STM32 (на базе Eclipse) и плат-

форма ARM mbed [11].

Отладочные платы STM32 Discovery [9] — это устройства, предназначенные преимущественно для изучения интерфейсов взаимодействия с различными типами периферийных устройств. Эти платы имеют в своем составе микроконтроллер семейства STM32, программатор-отладчик ST-Link/V2, а также периферийные устройства, набор которых зависит от модели платы. Существуют варианты для изучения MEMS-устройств, различных датчиков, дисплеев, коммуникаций. Набор инструментов разработки совпадает с таковым для платформы STM32 Nucleo. Комплекты Discovery успешно применяются для создания макетов и прототипов устройств [3].

Комплекты Nucleo и Discovery допускают расширение функционала основной платформы (особенно важно для Nucleo ввиду отсутствия в составе платформы каких-либо периферийных устройств). Дополнительные устройства могут подключаться с использованием интерфейса Morpho (применяется почти во всех вариантах Nucleo и Discovery) и Arduino-совместимого разъема (преимущественно Nucleo). Специально для серии STM32 Nucleo выпускаются расширения дополнения X-NUCLEO, в основном использующие Arduino-совместимый разъем. Кроме того, следуя документации, существует возможность подключения произвольных устройств, в том числе предназначенных для шины mikroBUS [6] или собранных на макетной плате.

Исходя из возможностей перечисленных устройств, наибольшим потенциалом для обучения программированию встраиваемых устройств обладают отладочные комплекты STM32 Discovery, например, STM32F746G-DISCO (рис. 1). Эта плата содержит все основные компоненты, которые могут использоваться в различных встраиваемых устройствах для IoT — коммуникационные интерфейсы (Ethernet, CAN, USB, UART), микропроцессорные интерфейсы (I<sup>2</sup>C, SPI, GPIO), мультимедийные интерфейсы (S/PDIF, аудиокодек, ЖК-дисплей, интерфейс для подключения камеры), устройства хранения данных и др. Разработка и отладка программного обеспечения может осуществляться как с помощью классических сред разработки, так и онлайн-платформы ARM mbed.



**Рис. 1. Плата STM32F746G-DISCO**

Применение данной платформы в процессе обучения студентов вузов программированию позволяет изучить следующие аспекты разработки ПО:

- особенности реализации типовых алгоритмов для вычислительных устройств, имеющих архитектуру, отличную от используемых в настольных компьютерах;
- организация прямого взаимодействия с внешними устройствами через аппаратные микропроцессорные интерфейсы;
- разработка многопоточных приложений для операционных систем реального времени;
- работа программных алгоритмов в условиях жесткого ограничения вычислительных ресурсов;
- реализация низкоуровневого доступа к устройствам хранения информации, мультимедийным и коммуникационным интерфейсам.

При изучении типовых алгоритмов, таких как сортировка и поиск данных, как правило, рассматривается лишь математическое обоснование изучаемого алгоритма, без учета особенностей его эффективной реализации. Алгоритмы, реализуемые на основе исключительно математического описания, в процессе тестирования могут демонстрировать производительность, оказывающуюся ниже расчетной, ввиду ограничений на способы работы с памятью, накладываемых используемой платформой. При использовании для тестирования настольных ПК на базе процессоров архитектуры x86 или AMD64 снижение производительности обычно оказывается незначительным благодаря наличию в таких процессорах механизмов оптимизации доступа. Если же тестирование осуществляется на микроконтроллерах, конструкция которых в целях оптимизации стоимости не предусматривает наличие таких механиз-

мов, то производительность может быть снижена в несколько раз, а в отдельных случаях выполнение алгоритма может привести к аппаратным сбоям вследствие неверного выравнивания исходных данных.

Решение данной проблемы заключается в рассмотрении возможностей транслятора изучаемого языка программирования, касающихся размещения и выравнивания статических данных, а также алгоритмов, позволяющих гарантировать правильное выравнивание при динамическом распределении памяти.

Тестирование таких алгоритмов на платформе Discovery позволяет выявить недостатки реализации алгоритмов управления памятью ввиду особенностей используемого в ней микроконтроллера архитектуры ARM, заключающихся в невозможности прямого обращения к значению в памяти, адрес которого не кратен размеру этого значения.

Изучение микропроцессорных интерфейсов, таких как I<sup>2</sup>C или SPI, позволяет на наглядных примерах рассмотреть работу с отдельными битами значений. Данные интерфейсы являются последовательными, что обуславливает необходимость сериализации и десериализации данных при обмене ими с внешними устройствами при реализации программного метода передачи данных (так называемый bit-banging). Изучение встроенных в микроконтроллер аппаратных трансиверов для этих интерфейсов при этом дает возможность сделать выводы об оптимальной структуре программы и общего алгоритма ее работы. В частности, механизмы, используемые в микроконтроллерах STM32, позволяют осуществлять прием и передачу данных без участия процессорного ядра.

Операционные системы реального времени, активно используемые во встраиваемых устройствах, предъявляют жесткие

требования к производительности используемых алгоритмов. Поскольку в таких системах требуется максимально быстрый отклик на внешние события, реализуемые студентами алгоритмы должны быть спроектированы таким образом, чтобы не препятствовать обработке событий.

Реализация многопоточности в приложениях для персональных компьютеров общего назначения позволяет в ряде случаев обойти потенциальные проблемы, возникающие при перекрещивании потоков, работающих с внешними устройствами, и их синхронизации. Это связано с невозможностью прямого доступа к оборудованию из прикладной программы в большинстве современных операционных систем. При реализации многопоточного приложения для встраиваемых систем часто приходится создавать собственные компоненты драйвера требуемого устройства, и необходимость в обеспечении корректного взаимодействия с устройством в условиях вытесняющей многозадачности позволяет подробно изучить ряд алгоритмов, роль которых в прикладных программах для компьютеров общего назначения была бы неочевидна.

Ограниченность вычислительных ресурсов микроконтроллеров (малый объем оперативной памяти, сравнительно низкая скорость работы процессорного ядра) дает возможность наглядного сравнения эффективных и неэффективных алгоритмов по ряду параметров. Малый объем оперативной памяти позволяет заведомо исключить ряд неэффективных реализаций алгоритмов, а отладка программного кода с использованием ИТМ (Instrumentation Trace Macrocell) позволяет получить подробную информацию о состоянии процессорного ядра во время выполнения исследуемого алгоритма [5]. Наличие в составе платформы Discovery встроенного программатора-отладчика позволяет осуществлять такую отладку, трассировку и профилирование без использования дополнительного оборудования. В отличие от использования ИТМ, профилирование алгоритмов, выполняющихся на персональном компьютере, обычно осуществляется лишь программными средствами. При выполнении программы под управлением многозадачной операционной системы

неизбежно возникает погрешность, связанная с влиянием других выполняющихся в системе программ на производительность исследуемого алгоритма.

Возможность низкоуровневого доступа к устройствам хранения информации (для платы STM32F746G-DISCO это Flash-память и SD-карта) позволяет изучить алгоритмы структурированного хранения данных на носителях информации, используемые в файловых системах, а также особенности выполнения операций ввода-вывода, в том числе буферизированного.

Наряду с приведенными сценариями, платформа Discovery может использоваться в качестве целевого устройства при выполнении лабораторных заданий на начальных этапах курса программирования в вузах. Ввиду большого количества встроенных периферийных устройств лабораторные задания по темам, связанным с разработкой ПО для встраиваемых систем, могут быть сформированы с ориентацией на коллективную разработку, например, в рамках унифицированного процесса [1].

### Заключение

Возможности платформы STM32 Discovery позволяют обучаемым комплексно рассмотреть принципы работы и программирования основных компонентов вычислительных устройств, как в отдельности, так и в составе сложной вычислительной системы. Это позволяет рассматривать платформу STM32F746G-DISCO как средство, позволяющее студентам наряду с освоением основных элементов курса программирования, таких как изучение языка программирования и основы алгоритмизации, на наглядном примере рассмотреть особенности разработки программного обеспечения для встраиваемых устройств, средства взаимодействия таких устройств в рамках «интернета вещей», а также сформировать навыки работы с микроконтроллерами и устройствами на их основе. Платформа является в определенной степени самодостаточной, но при этом допускает возможность расширения, что позволяет использовать ее также и при изучении специализированных дисциплин.

### ЛИТЕРАТУРА

1. Алексеевский П. И. Обучение программированию студентов на основе методологии унифицированного процесса разработки программного обеспечения // Педагогическое образование в России. — 2014. — № 8. — С. 150–153.
2. Гордиевских В. М., Кораблев А. А. Микроконтроллеры Lego EV3 и Arduino Uno как технологическая основа для курса робототехники в вузе // Вестник Шадринского государственного педагогического университета. — 2016. — № 3 (31). — С. 160–163.
3. Данченко Д. Г. Использование отладочной платы STM32F7 Discovery для макетирования микропроцессорных устройств // Молодой ученый. Казань. — 2017. — № 51. — С. 37–42.

4. «Интернет вещей» (IoT) в России: технология будущего, доступная уже сейчас [Электронный ресурс] : анализ. обзор / PwC. — М., 2017. — Режим доступа: [https://www.pwc.ru/ru/publications/iot/IoT-inRussia-research\\_rus.pdf](https://www.pwc.ru/ru/publications/iot/IoT-inRussia-research_rus.pdf) (дата обращения: 30.05.2018).
5. Копытин С., Дорофеев К. Отладка микроконтроллеров на базе процессоров Cortex-M3 // Компоненты и технологии. — 2008. — № 86. — С. 73–76.
6. Лебедев Н. В., Кучерук В. В. Принципы автоматизации управления приемопередающим каналом и получения данных ввода-вывода на базе программируемых микроконтроллеров STM32 // Успехи современной науки. — 2017. — № 3. — С. 83–86.
7. Линева Ф. А., Киселева С. Д. Интернет вещей на основе Raspberry Pi // Молодежный научно-технический вестник. — М., 2015. — № 5. — С. 22.
8. Слива М. В. Использование миникомпьютера Raspberry Pi для преподавания основ робототехники // Культура, наука, образование: проблемы и перспективы : материалы Всероссийской конференции (Нижегородск, 7 фев. 2014). — Нижегородск, 2014. — С. 326–328.
9. 32F746GDISCOVERY — Discovery kit with STM32F746NG MCU [Электронный ресурс] // STMicroelectronics. — Режим доступа: [http://www.st.com/content/st\\_com/en/products/evaluation-tools/product-evaluation-tools/mcu-eval-tools/stm32-mcu-eval-tools/stm32-mcu-discovery-kits/32f746gdiscovery.html](http://www.st.com/content/st_com/en/products/evaluation-tools/product-evaluation-tools/mcu-eval-tools/stm32-mcu-eval-tools/stm32-mcu-discovery-kits/32f746gdiscovery.html) (дата обращения: 30.05.2018)
10. Arduino — Introduction [Электронный ресурс] // Arduino. — Режим доступа: <https://www.arduino.cc/en/Guide/Introduction> (дата обращения: 30.05.2018).
11. ARM Mbed IoT Device Platform [Электронный ресурс] // Arm Limited. — Режим доступа: <https://www.mbed.com/> (дата обращения: 30.05.2018).
12. Micro:bit [Электронный ресурс] // Micro:bit Educational Foundation. — Режим доступа: <http://microbit.org/teach/> (дата обращения: 30.05.2018).
13. Mindstorms EV3 [Электронный ресурс] // The Lego Group. — Режим доступа: <https://www.lego.com/ru-ru/mindstorms/products/mindstorms-ev3-31313> (дата обращения: 30.05.2018).
14. Raspberry Pi Documentation [Электронный ресурс] // Raspberry Pi Foundation. — Режим доступа: <https://www.raspberrypi.org/documentation/> (дата обращения: 30.05.2018).
15. STM32 MCU Nucleo [Электронный ресурс] // STMicroelectronics. — Режим доступа: <http://www.st.com/en/evaluation-tools/stm32-mcu-nucleo.html?querycriteria=productId=LN1847> (дата обращения: 30.05.2018).

#### REFERENCES

1. Alekseevskiy P. I. Obuchenie programmirovaniyu studentov na osnove metodologii unifitsirovannogo protsessa razrabotki programmnoy obespecheniya // Pedagogicheskoe obrazovanie v Rossii. — 2014. — № 8. — С. 150–153.
2. Gordievskikh V. M., Korablev A. A. Mikrokontrollery Lego EV3 i Arduino Uno kak tekhnologicheskaya osnova dlya kursa robototekhniki v vuze // Vestnik Shadrinskogo gosudarstvennogo pedagogicheskogo universiteta. — 2016. — № 3 (31). — С. 160–163.
3. Danchenko D. G. Ispol'zovanie otladochnoy platy STM32F7 Discovery dlya maketirovaniya mikroprotssessornykh ustroystv // Molodoy uchenyy. Kazan'. — 2017. — № 51. — С. 37–42.
4. «Internet veshchey» (IoT) v Rossii: tekhnologiya budushchego, dostupnaya uzhe seychas [Elektronnyy resurs] : analit. obzor / PwC. — М., 2017. — Rezhim dostupa: [https://www.pwc.ru/ru/publications/iot/IoT-inRussia-research\\_rus.pdf](https://www.pwc.ru/ru/publications/iot/IoT-inRussia-research_rus.pdf) (data obrashcheniya: 30.05.2018).
5. Kopytin S., Dorofeev K. Otladka mikrokontrollerov na baze protsessorov Cortex-M3 // Komponenty i tekhnologii. — 2008. — № 86. — С. 73–76.
6. Lebedev N. V., Kucheruk V. V. Printsipy avtomatizatsii upravleniya priemoperedayushchim kanalom i polucheniya dannykh vvoda-vyvoda na baze programmirovemykh mikrokontrollerov STM32 // Uspekhi sovremennoy nauki. — 2017. — № 3. — С. 83–86.
7. Lineva F. A., Kiseleva S. D. Internet veshchey na osnove Raspberry Pi // Molodezhnyy nauchno-tekhnicheskyy vestnik. — М., 2015. — № 5. — С. 22.
8. Sliva M. V. Ispol'zovanie minikompyutera Raspberry Pi dlya prepodavaniya osnov robototekhniki // Kul'tura, nauka, obrazovanie: problemy i perspektivy : materialy Vserossiyskoy konferentsii (Nizhnevartovsk, 7 fev. 2014). — Nizhnevartovsk, 2014. — С. 326–328.
9. 32F746GDISCOVERY — Discovery kit with STM32F746NG MCU [Elektronnyy resurs] // STMicroelectronics. — Rezhim dostupa: [http://www.st.com/content/st\\_com/en/products/evaluation-tools/product-evaluation-tools/mcu-eval-tools/stm32-mcu-eval-tools/stm32-mcu-discovery-kits/32f746gdiscovery.html](http://www.st.com/content/st_com/en/products/evaluation-tools/product-evaluation-tools/mcu-eval-tools/stm32-mcu-eval-tools/stm32-mcu-discovery-kits/32f746gdiscovery.html) (data obrashcheniya: 30.05.2018)
10. Arduino — Introduction [Elektronnyy resurs] // Arduino. — Rezhim dostupa: <https://www.arduino.cc/en/Guide/Introduction> (data obrashcheniya: 30.05.2018).
11. ARM Mbed IoT Device Platform [Elektronnyy resurs] // Arm Limited. — Rezhim dostupa: <https://www.mbed.com/> (data obrashcheniya: 30.05.2018).
12. Micro:bit [Elektronnyy resurs] // Micro:bit Educational Foundation. — Rezhim dostupa: <http://microbit.org/teach/> (data obrashcheniya: 30.05.2018).
13. Mindstorms EV3 [Elektronnyy resurs] // The Lego Group. — Rezhim dostupa: <https://www.lego.com/ru-ru/mindstorms/products/mindstorms-ev3-31313> (data obrashcheniya: 30.05.2018).
14. Raspberry Pi Documentation [Elektronnyy resurs] // Raspberry Pi Foundation. — Rezhim dostupa: <https://www.raspberrypi.org/documentation/> (data obrashcheniya: 30.05.2018).
15. STM32 MCU Nucleo [Elektronnyy resurs] // STMicroelectronics. — Rezhim dostupa: <http://www.st.com/en/evaluation-tools/stm32-mcu-nucleo.html?querycriteria=productId=LN1847> (data obrashcheniya: 30.05.2018).