

ОБЗОР РЕШЕНИЙ ДЛЯ РЕАЛИЗАЦИИ ПРОТОТИПА ЗАМКА МОБИЛЬНОЙ БИОМЕТРИЧЕСКОЙ СИСТЕМЫ ДОСТУПА M-BLOCK

А.А. Коровянский

В статье рассматриваются существующие средства для реализации прототипа замка мобильной биометрической системы доступа M-Block. В результате сравнительного анализа определяется наиболее подходящее решение.

Введение

При реализации любого программно-аппаратного комплекса большое значение имеет обзор, сравнительный анализ и выбор наиболее подходящего средства для реализации. Мотивацией написания данной статьи является решение данной задачи для прототипа замка мобильной биометрической системы доступа M-Block, разрабатываемой на факультете компьютерных наук ОмГУ им. Ф.М. Достоевского.

Биометрическая система доступа (БСД) — это программно-аппаратный комплекс, решающий задачу разграничения физического доступа к целевому объекту путём проверки биометрических данных. Таким образом, для открытия дверей используются специальные замки и биометрические алгоритмы, верифицирующие человека по отпечатку пальца, внешности, голосу или другому биометрическому фактору.

В основе M-Block лежит идея использования возможностей смартфонов и других современных мобильных устройств для создания инновационной и промышленно значимой мобильной БСД. Система должна обеспечивать высокий уровень безопасности, поддерживать различные факторы биометрической авторизации и иметь возможность централизованной и простой настройки политик доступа.

На рис 1. представлена принципиальная схема M-Block. Три ключевых компонента системы: мобильное приложение, приложение замка и приложение сервера.

Последовательность событий при взаимодействии пользователя с системой:

1. Пользователь подходит к двери с целью открыть её.

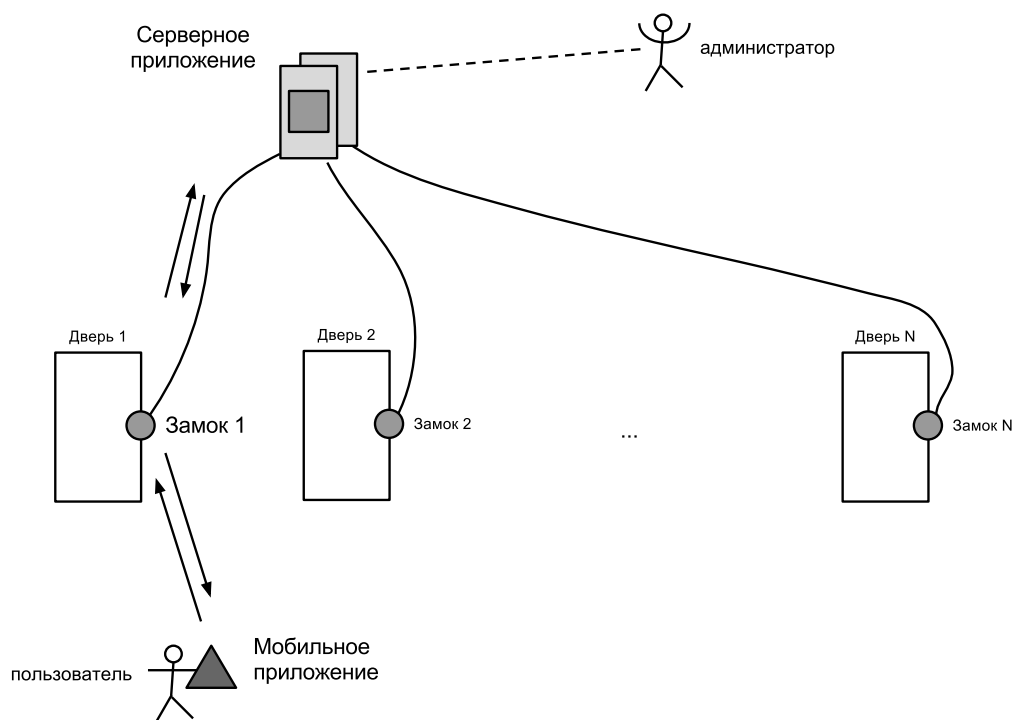


Рис. 1. Схема M-Block

2. Мобильное приложение M-Block устанавливает защищённое соединение с приложением замка M-Block.
3. Приложение замка передаёт мобильному приложению необходимую информацию для верификации (например, какие биометрические данные нужно получить от пользователя).
4. Пользователь с помощью мобильного приложения формирует необходимые биометрические данные для верификации (например, для голосовой биометрии — цифровую запись определённой фразы).
5. Мобильное приложение передаёт сформированные данные замку.
6. Приложение замка устанавливает защищённое соединение с сервером, передаёт ему биометрические данные и другую необходимую информацию.
7. Сервер M-Block запускает необходимый биометрический алгоритм, верифицирует и авторизует пользователя.
8. Сервер сообщает приложению замка результаты верификации и авторизации.
9. В случае положительного ответа приложение замка открывает дверь, в противном случае пользователю передаётся информация о причинах отказа.

Как и любая другая БСД, M-Block является весьма сложным программно-аппаратным комплексом. Более того, требования к ней могут меняться в процессе разработки, тестирования и апробации. Поэтому в первую очередь разработка системы должна начинаться с реализации рабочего прототипа, к которому предъявляется жёсткое условие возможности добавления и изменения требований в процессе разработки.

Целью данной статьи является обзор и выбор наиболее подходящего средства для реализации прототипа аппаратной и программной части замка.

1. Критерии оценки

Для определения лучшего средства решения поставленной задачи необходимо сформулировать критерии оценки. Важно отметить, что перечисленные ниже критерии имеют значимость для реализации прототипа и могут быть неподходящими при выборе средств для реализации промышленной версии.

1.1. Взаимодействия с электромеханическим устройствами

Для того, чтобы открывать и закрывать дверь, замок M-Block должен взаимодействовать с промышленным электромеханическим дверным замком. Поэтому устройство должно обязательно предоставлять программно-аппаратные возможности для управления электромеханическим приводом.

1.2. Скорость и удобство разработки

Для прототипа большое значение имеет скорость и удобство разработки, поэтому устройство должно поддерживать программирование на высокоуровневом языке программирования с использованием удобной и эффективной IDE и других инструментов разработки.

1.3. Вычислительные ресурсы

Приложение замка в ходе своей работы должно обмениваться сообщениями с мобильным приложением и сервером M-Block, обрабатывать и передавать значительные объёмы биометрических данных. Поэтому для прототипа замка ставятся следующие требования: частота процессора - не менее 128 МГц, размер оперативной памяти - не менее 100 МБ, размер внутренней памяти (или подключаемой) - не менее 100 МБ.

1.4. Беспроводное взаимодействие

Замок должен устанавливать беспроводное защищённое соединение с мобильным приложением M-Block, поэтому прототип должен поддерживать как минимум один из современных способов беспроводной передачи данных на небольших расстояниях - Bluetooth или WiFi.

1.5. Взаимодействие по локальной сети

Замок должен устанавливать защищённое соединение с сервером M-Block, поэтому прототип должен обеспечивать поддержку популярных стандартов локальных сетей Ethernet или WiFi.

1.6. Стек TCP/IP

Исходя из поставленной задачи, прототип замка должен обеспечивать работу со стеком протоколов TCP/IP и предоставлять возможности реализации безопасного протокола на основе TCP и UDP сокетов. Также в качестве транспорта для протокола взаимодействия между замком и сервером может быть выбран HTTP, поэтому крайне желательна поддержка HTTP/HTTPS при программировании устройства замка.

1.7. Многозадачность

Прототип замка должен одновременно поддерживать связь с мобильным приложением и сервером M-Block, поэтому решение для прототипа должно предоставлять средства для организации многозадачности.

1.8. Документация разработчика и поддержка

Качество документации, наличие сторонних расширений, количество статей и различных учебных материалов в сети Интернет, активность сообщества разработчиков для каждого конкретного решения — все это является очень важным критерием для прототипа, так как минимизирует время на решение вопросов, возникающих в процессе разработки.

1.9. Стоимость

Для прототипа замка стоимость не является ключевым фактором, так как прототип является уникальной и единичной реализацией. Однако данный параметр должен рассматриваться как дополнительный и значимый, в случае равенства в сравнении решений по вышеперечисленным критериям.

2. Существующие решения

На данный момент существуют различные средства решения сформулированной задачи. В ходе подготовительной работы автором статьи было выделено подмножество предпочтительных решений, то есть заведомо более подходящих для поставленной задачи среди всего спектра решений. В их число вошли решения на базе Arduino, AllWinner A1X, Android и Raspberry Pi. В рамках данной статьи будут детально рассмотрены все предпочтительные решения и в результате сравнительного анализа по сформулированным критериям будет выявлено наиболее подходящее.

2.1. Arduino

Расширяемая аппаратно-вычислительная платформа, основными компонентами которой являются платы ввода/вывода, подключаемые шилды и среда разработки на языке Processing. Arduino предлагает различные модели плат, но все они основаны на микроконтроллере Atmel AVR, имея небольшие вычислительную мощность и размер по памяти [1].

Таблица 1. Arduino

Критерий	Комментарий	Оценка
Взаимодействие с электромеханическими устройствами	Arduino поддерживает прямое программирование портов GPIO, что является одной из его самых сильных сторон [2].	+
Скорость и удобство разработки	Язык Processing, а также IDE для программирования Arduino являются достаточно хорошими и проверенными решениями, но значительно проигрывают в скорости и удобстве разработки современным высокоуровневым языкам программирования и соответствующим им IDE. Также при использовании большого количества дополнительных шилдов может возникнуть серьёзная проблема нехватки GPIO разъёмов на плате и необходимости взаимоисключения шилдов.	+/-
Вычислительные ресурсы	Решения на базе Arduino имеют очень ограниченную и явно недостаточную для прототипа вычислительную мощность. Частота микроконтроллера не превышает 16 МГц, а размер оперативной памяти - 8 КБ [3].	-
Беспроводное взаимодействие	Arduino позволяет реализовать беспроводное взаимодействие по стандартам Bluetooth и WiFi за счет использования специальных плат или подключения дополнительных соответствующих шилдов. Так, например, взаимодействие по Bluetooth возможно за счёт использования платы Arduino BT/ArduinoBT или подключения шилда XBee [5].	+
Взаимодействие по локальной сети	Путем подключения дополнительных шилдов Ethernet [7] и WiFi [6] возможна реализация взаимодействия по соответствующему стандарту локальной сети. При этом плата Arduino может выступать как в роли клиента [9], так и сервера [8].	+

Критерий	Комментарий	Оценка
Стек TCP/IP	Стандартные библиотеки Arduino характеризуются очень низкоуровневым подходом к реализации взаимодействия по протоколам стека TCP/IP и отсутствием поддержки некоторых протоколов или определённых их возможностей. Например, Arduino не поддерживает взаимодействие по протоколу HTTPS по причине недостаточной вычислительной мощности.	–
Многозадачность	Arduino не имеет встроенных средств для обеспечения многозадачности. Поддержка многозадачности возможна только за счёт использования специальных сторонних библиотек или RTOS [10]. Однако оба варианта приведут к значительному усложнению реализации. В итоге многозадачность в Arduino является весьма проблемным вопросом с большим количеством «подводных камней».	–
Документация разработчика и поддержка	Официальный сайт Arduino предоставляет большое количество обучающих материалов и хорошую документацию разработчика. Также вокруг платформы существует большое сообщество активных разработчиков, что значительно облегчает процесс решения сложных вопросов, возникающих в процессе разработки.	+
Стоимость	80€ [11]	

2.2. Raspberry Pi

Одноплатный компьютер размером с кредитную карту, имеющий аппаратные характеристики, достаточные для комфортной работы современных десктопных и мобильных операционных систем. За счёт наличия 26 портов GPIO и сигнального процессора Raspberry Pi представляет хорошие возможности для разработки на его базе различных программно-аппаратных решений [12].

Таблица 2. Raspberry Pi

Критерий	Комментарий	Оценка
Взаимодействие с электромеханическими устройствами	Raspberry Pi имеет 26 портов GPIO и поддерживает их прямое программирование [13]. Также стоит отметить факт существования проекта raspberry-gpio-python, позволяющего управлять GPIO на высокоуровневом языке Python [14].	+

Критерий	Комментарий	Оценка
Скорость и удобство разработки	Официально поддерживаемым языком для разработки приложений на Raspberry Pi является Python [15], для которого существует большое количество различных IDE. Оба эти фактора обеспечивают хорошие возможности для скоростной и комфортной разработки приложений под Raspberry Pi.	+
Вычислительные ресурсы	Raspberry обладает достаточными физическими характеристиками, имея 700 МГц процессор, 256 или 512 МБ памяти [15] и возможность подключения до 32 ГБ внешней памяти.	+
Беспроводное взаимодействие	Реализация взаимодействия по Bluetooth и WiFi возможна путём подключения к USB порту Raspberry Pi WiFi/Bluetooth адаптера.	+
Взаимодействие по локальной сети	Raspberry Pi предоставляет встроенные средства для поддержки стандарта Ethernet и за счёт расширения позволяет обеспечить поддержку WiFi. При этом Raspberry Pi может выступать как в роли клиента, так и в роли сервера.	+
Стек TCP/IP	Стандартной операционной системой Raspberry Pi является Raspbian [16] — оптимизированная и преднастроенная версия Debian. Raspbian содержит в себе полноценную реализацию стека TCP/IP, предоставляя необходимые средства для создания TCP и UDP сокетов, а также взаимодействия по протоколам HTTP и HTTPS.	+
Многозадачность	Raspbian, также как и Debian, является многозадачной операционной системой.	+
Документация разработчика и поддержка	Для Raspberry Pi в сети Интернет можно найти большое количество документации и различных учебных материалов. Также вокруг Raspberry Pi сформировалось большое и очень активное сообщество разработчиков, которое активно развивает платформу, каждый день появляются новые статьи о решении самых различных задач с помощью Raspberry Pi. Все это позволяет очень быстро найти ответ на различные вопросы, возникающие в процессе разработки для Raspberry Pi.	+
Стоимость	21.60€ [17]	

2.3. Android

Открытая мобильная операционная система, основанная на ядре Linux. На данный момент на рынке представлено огромное количество различных устройств под управлением Android, в том числе смартфонов, планшетов, электронных книг, домашних мультимедиа центров, фотоаппаратов и телевизоров [18].

Для решения сформулированной задачи наиболее удачным форм-фактором является смартфон, так как при прочих равных он имеет компактный размер, необходимые аппаратные возможности и наиболее качественную документацию. Поэтому оценка Android будет даваться именно на основе возможностей Android смартфонов.

Таблица 3. Android

Критерий	Комментарий	Оценка
Взаимодействие с электромеханическими устройствами	Существующие Android смартфоны не имеют GPIO портов и возможностей для прямого взаимодействия с электромеханическими устройствами [19]. Тем не менее, Android предоставляет два обходных решения. Первое заключается в поддержке взаимодействия с внешними устройствами по USB [20], что подходит не для всех устройств и внесёт дополнительную сложность в реализацию прототипа. Вторым вариантом является взаимодействие в рамках подхода Android ADK [20], при этом в дополнении к устройству на базе Android для реализации прототипа замка необходимо использовать дополнительное решение, например, плату Arduino ADK.	+/-
Скорость и удобство разработки	Стандартным языком для разработки приложений для Android является язык Java, а стандартной IDE — Eclipse [18]. Эти два фактора обеспечивают хорошие возможности для скоростной и комфортной разработки Android приложений.	+
Вычислительные ресурсы	Практически все современные Android смартфоны соответствуют необходимым физическим требованиям [19] в силу более высоких минимальных требований для работы ОС Android [21].	+
Беспроводное взаимодействие	Практически все современные смартфоны на Android содержат WiFi и Bluetooth модули. [19].	+
Взаимодействие по локальной сети	Практически все современные смартфоны на Android содержат WiFi модуль и не содержат порт Ethernet [19].	+

Критерий	Комментарий	Оценка
Стек TCP/IP	ОС Android содержит полноценную реализацию стека TCP/IP, а Android SDK предоставляет все необходимые средства для создания TCP и UDP сокетов и взаимодействия по протоколам HTTP и HTTPS [23].	+
Многозадачность	Android является многозадачной операционной системой, также предоставляя стандартные средства для разработки многопоточных приложений [24].	+
Документация разработчика и поддержка	Официальный сайт Android Developers содержит исчерпывающее количество документации и учебных материалов по разработке приложений на базе Android SDK. Также на тему разработки приложений для Android было сделано большое количество докладов на различных конференциях, написано огромное количество статей и обсуждений на различных форумах.	+
Стоимость	От \$160 [25]	

2.4. AllWinner A1X

Семейство одноядерных однокристальных систем. На данный момент представлено решениями A10, A13, A10s и A31. Содержит одно ядро ARM Cortex-A8 как основной процессор CPU и графический процессор GPU Mali 400 [26].

Самого по себе AllWinner A1X недостаточно для решения поставленной задачи, однако на его базе существует ряд интересных решений: BeagleBone [28], Hackberry [30], Cubieboard [27], Gooseberry [29] и ряд других. Для поставленной задачи наиболее подходящими вариантами являются решения, содержащие порты GPIO и доступные для заказа — это CubieBoard и BeagleBone. Данные устройства весьма схожи между собой по техническим характеристикам и были разработаны позже Raspberry Pi с целью составить достойную конкуренцию за счет более высоких аппаратно-вычислительных возможностей при сравнимой с Pi цене. Поэтому оценка AllWinner A1X будет даваться именно на основе возможностей CubieBoard и BeagleBone.

Таблица 4. AllWinner A1X

Критерий	Комментарий	BeagleBone	CubieBoard
Взаимодействие с электро-механическими устройствами	BeagleBone и CubieBoard поддерживают прямое программирование портов GPIO. Для BeagleBone это возможно по умолчанию средствами языка Bash [31], для CubieBoard необходимо поставить Linux в качестве ОС.	+	+/-
Скорость и удобство разработки	Предустановленной ОС для CubieBoard является Android, программирование которого возможно на Java в Eclipse IDE. Для BeagleBone предустановленной ОС является Angstrom [32], программирование которого возможно на JavaScript в Cloud9 IDE [28]. При установке Linux на CubieBoard для написания программ можно использовать C/C++ и Eclipse IDE [27]. Все перечисленные варианты должны обеспечить достаточно высокую скорость и удобство разработки.	+	+
Вычислительные ресурсы	CubieBoard и BeagleBone обладают достаточными физическими характеристиками. BeagleBone: 720 МГц процессор, 256 МБ RAM, возможность подключения до 32 ГБ внешней памяти [28]. CubieBoard: 1 ГГц процессор, 1 ГБ RAM, 4 ГБ внутренней NAND памяти и возможность подключения до 32 ГБ внешней памяти [27].	+	+
Беспроводное взаимодействие	Реализация взаимодействия по Bluetooth и WiFi теоретически возможна путем подключения к USB порту соответствующих адаптеров, однако на практике сообщается о проблемах и необходимости внесения патчей в ядро системы.	+/-	+/-
Взаимодействие по локальной сети	BeagleBone и CubieBoard имеют встроенный Ethernet порт и все необходимые средства для написания как клиентских, так и серверных приложений.	+	+

Критерий	Комментарий	BeagleBone	CubieBoard
Стек TCP/IP	Angstrom, Android, Ubuntu, как и все другие популярные варианты ОС для CubieBoard и BeagleBone, имеют полноценную реализацию стека TCP/IP и предоставляют необходимые для прототипа сетевые возможности.	+	+
Многозадачность	Angstrom, Android, Ubuntu как и все другие популярные варианты ОС для CubieBoard и BeagleBone являются многозадачными операционными системами.	+	+
Документация разработчика и поддержка	CubieBoard и BeagleBone значительно проигрывают Raspberry Pi в качестве документации разработчика, по количеству статей и обсуждений, в размере и активности сообщества разработчиков, по наличию сторонних расширений и приложений, а также другим смежным параметрам. Например, на первой странице выдачи Google по запросу «raspberrypi gpio» можно найти ссылку на специально созданную для этого библиотеку raspberrypi-gpio-python [14] и ещё 8 обучающих статей по этому вопросу. Запрос «beaglebone gpio» — в выдаче библиотеки beaglebone-gpio для языка C и 4 статьи. Запрос «cubieboard gpio» — в выдаче только спецификации cubieboard и безуспешные обсуждения на форумах.	+/-	-
Стоимость	CubieBoard — \$60 [34]. BeagleBone — \$89 [33]		

3. Заключение

Из сравнительного анализа и оценок для каждого решения следует:

1. Наиболее подходящим средством для реализации прототипа замка M-Block является Raspberry Pi, так как данное решение получило положительные оценки для всех критериев, не получив ни одной негативной оценки.
2. Arduino не может рассматриваться как возможный вариант для решения,

так как имеет несоответствия по критериям «Вычислительные ресурсы» и «Стек TCP/IP».

3. Android и AllWinner A1X стоит признать подходящими и весьма хорошими вариантами для прототипа, но проигрывающими в сравнении с Raspberry Pi. Android — из-за не возможности прямого программирования GPIO портов, а решения на базе AllWinner A1X — из-за недостаточного качества документации и низкой активности сообщества.

ЛИТЕРАТУРА

1. Официальный сайт Arduino [Электронный ресурс]. URL: <http://www.arduino.cc/> (дата обращения: 31.03.2013).
2. Arduino Analog Input Pins // Официальный сайт Arduino [Электронный ресурс]. URL: <http://arduino.cc/en/Tutorial/AnalogInputPins> (дата обращения: 31.03.2013).
3. Arduino Memory // Официальный сайт Arduino [Электронный ресурс]. URL: <http://arduino.cc/en/Tutorial/Memory> (дата обращения: 31.03.2013).
4. Arduino Bluetooth Shield // Официальный сайт Arduino [Электронный ресурс]. URL: <http://arduino.cc/en/Main/ArduinoBoardBluetooth> (дата обращения: 31.03.2013).
5. XBee Shield Wiki // Seeed Wiki [Электронный ресурс]. URL: http://www.seeedstudio.com/wiki/XBee\%C2\%AE_Shield (дата обращения: 31.03.2013).
6. Arduino WiFi Shield // Официальный сайт Arduino [Электронный ресурс]. URL: <http://arduino.cc/en/Main/ArduinoWiFiShield> (дата обращения: 31.03.2013).
7. Arduino Ethernet Shield // Официальный сайт Arduino. URL: <http://arduino.cc/en/Main/ArduinoEthernetShield> (дата обращения: 31.03.2013).
8. WebServer // Официальный сайт Arduino [Электронный ресурс]. URL: <http://arduino.cc/en/Tutorial/WebClient> (дата обращения: 31.03.2013).
9. WiFi Twitter Client // Официальный сайт Arduino [Электронный ресурс]. URL: <http://arduino.cc/en/Tutorial/WiFiTwitterClient> (дата обращения: 31.03.2013).
10. RTOS for Arduino // Out There [Электронный ресурс]. URL: <http://arduino.cc/en/Tutorial/WiFiTwitterClient> (дата обращения: 31.03.2013).
11. Arduino Starter Kit // Arduino Store [Электронный ресурс]. URL: http://store.arduino.cc/ww/index.php?main_page=product_info&cPath=2_23&products_id=185 (дата обращения: 31.03.2013).
12. Официальный сайт Raspberry Pi [Электронный ресурс]. URL: <http://www.raspberrypi.org/> (дата обращения: 31.03.2013).
13. RPi Hardware // eLinux.org [Электронный ресурс]. URL: http://elinux.org/RPi_Hardware (дата обращения: 31.03.2013).
14. Python library for GPIO access on Raspberry Pi // Google Code [Электронный ресурс]. URL: <https://code.google.com/p/raspberry-gpio-python/> (дата обращения: 31.03.2013).

15. FAQs // Официальный сайт Raspberry Pi [Электронный ресурс]. URL: <http://www.raspberrypi.org/faqs> (дата обращения: 31.03.2013).
16. Официальный сайт Raspbian [Электронный ресурс]. URL: <http://www.raspbian.org/> (дата обращения: 31.03.2013).
17. Raspberry Pi // RS Online [Электронный ресурс]. URL: <http://uk.rs-online.com/web/generalDisplay.html?id=raspberrypi> (дата обращения: 31.03.2013).
18. Официальный сайт Android Developers [Электронный ресурс]. URL: <http://developers.android.com> (дата обращения: 31.03.2013).
19. Сравнение Android устройств // Википедия [Электронный ресурс]. URL: http://ru.wikipedia.org/wiki/Сравнение_устройств_с_Android (дата обращения: 31.03.2013).
20. USB Host and Accessory // Android Developers [Электронный ресурс]. URL: <http://developer.android.com/guide/topics/connectivity/usb/index.html> (дата обращения: 31.03.2013).
21. Android Hardware Requirments // Android Hardwares [Электронный ресурс]. URL: <http://androidhardwares.com/android-hardware-development/android-hardware-requirements/> (дата обращения: 31.03.2013).
22. Performing Network Operations // Android Developers [Электронный ресурс]. URL: <http://developer.android.com/training/basics/network-ops/index.html> (дата обращения: 31.03.2013).
23. Performing Network Operations // Android Developers [Электронный ресурс]. URL: <http://developer.android.com/training/basics/network-ops/index.html> (дата обращения: 31.03.2013).
24. Painless Threading // Android Developers [Электронный ресурс]. URL: <http://android-developers.blogspot.com/2009/05/painless-threading.html> (дата обращения: 31.03.2013).
25. Самый дешевый в мире Android смартфон // АйМобилка [Электронный ресурс]. URL: <http://icellphone.ru/kitajcy-pokazali-samyj-deshevyj-v-mire-android-smartfon.html> (дата обращения: 31.03.2013).
26. AllWinner A Serial // Официальный сайт AllWinner Technology [Электронный ресурс]. URL: <http://www.allwinnertech.com/en/product/A-Serial.html> (дата обращения: 31.03.2013).
27. Официальный сайт Cubieboard [Электронный ресурс]. URL: <http://linux-sunxi.org/Cubieboard> (дата обращения: 31.03.2013).
28. Официальный сайт GooseBerry [Электронный ресурс]. URL: <http://beagleboard.org/bone> (дата обращения: 31.03.2013).
29. Официальный сайт Gooseberry [Электронный ресурс]. URL: <http://gooseberry.atSPACE.co.uk/> (дата обращения: 31.03.2013).
30. Официальный сайт Hackberry [Электронный ресурс]. URL: <https://www.miniand.com/products/Hackberry%20A10%20Developer%20Board> (дата обращения: 31.03.2013).
31. Accessing GPIO // BeagleBone for Dummies [Электронный ресурс]. URL: <http://bbfordummies.blogspot.ru/2009/07/1.html> (дата обращения: 31.03.2013).
32. Angstrom Distribution // BeagleBoard.org [Электронный ресурс]. URL: <http://>

- beagleboard.org/project/angstrom/ (дата обращения: 31.03.2013).
33. BeagleBone A6 // Adafruit [Электронный ресурс]. URL: <https://www.adafruit.com/products/513> (дата обращения: 31.03.2013).
 34. CubieBoard // DFRobot [Электронный ресурс]. URL: http://www.dfrobot.com/index.php?route=product/product&filter_name=cubieboard&product_id=881#.UVxjTZMqwUR (дата обращения: 31.03.2013).