

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЧЕРНІГІВСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ**

**ВІЛЬНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ
В ОСВІТІ, НАУЦІ ТА БІЗНЕСІ**

**Тези доповідей шостої всеукраїнської науково-практичної
конференції**

(м. Чернігів, 15 травня 2015 р.)

Чернігів 2015

УДК 004.4
ББК 32.973-018.2
В46

Друкується за рішенням вченої ради Чернігівського національного технологічного університету (протокол № 6 від 30 червня 2015 року).

Вільне програмне забезпечення в освіті, науці та бізнесі : тези доповідей шостої
В46 Всеукраїнської науково-практичної конференції. – Чернігів : Черн. нац. технол. ун-т, 2015.
– 22 с.

У збірник включені тези доповідей, які були представлені на шостій всеукраїнській науково-практичній конференції «Вільне програмне забезпечення в освіті, науці та бізнесі». У доповідях розглянуто наукові та методичні питання щодо використання вільного програмного забезпечення під час організації навчального процесу, перевірки знань, вивчення сучасних технологій та їх впровадження в наукову роботу та на виробництві.

Голова редакційної колегії:

Казимир В.В., д-р техн. наук, професор

Редакційна колегія:

Мельниченко Д.І.

Роговенко А.І.

Стасюк С.С.

© Чернігівський національний
технологічний університет, 2015

ЗМІСТ

О. Verovko REVIEW OF OPEN SOURCE ELECTRONIC DESIGN AUTOMATION SYSTEMS	4
М. Verovko ANALYSIS AND COMPARISON OF OPEN SOURCE ELEARNING SYSTEMS	5
Крива М., Роговенко А. ВИКОРИСТАННЯ МОДУЛЯ ESP8266 У ВБУДОВАНИХ СИСТЕМАХ.....	7
Врублевский А. ПРОГРАММНЫЙ КОМПЛЕКС ФОРМИРОВАНИЯ ПРОГРАММ ПЕРЕДАЧ ДЛЯ ТЕЛЕВИЗИОННЫХ КАНАЛОВ.....	9
Мостіпан А. РОЗРОБКА ANDROID-ДОДАТКУ ДЛЯ ВЗАЄМОДІЇ З СЕРВЕРНОЮ ЧАСТИНОЮ СИСТЕМИ КЕРУВАННЯ ПАТРУЛЯМИ ДОБРОВІЛЬНОЇ НАРОДНОЇ ДРУЖИНИ	10
Тур Г. ОТКРЫТЫЙ СБОРЩИК МУСОРА ВОЕНМ GC ДЛЯ ЯЗЫКОВ C/C++	12
Андрущенко Р., Ульченко Д. ОБЕСПЕЧЕНИЕ ВЗАИМОДЕЙСТВИЯ УСТРОЙСТВ «ИНТЕРНЕТА ВЕЩЕЙ» С СЕРВИСОМ THINGSPEAK	13
Хворостенко Р. РОЗРОБКА ВЕБ-ДОДАТКУ ДЛЯ ВЗАЄМОДІЇ З КЛІЄНТСЬКОЮ ЧАСТИНОЮ СИСТЕМИ КЕРУВАННЯ ПАТРУЛЯМИ ДОБРОВІЛЬНОЇ НАРОДНОЇ ДРУЖИНИ	16
Грицай Д. ОСОБЕННОСТИ РАЗРАБОТКИ СИСТЕМ НА МОДУЛЕ НА ПРИМЕРЕ "FreePCX"	17
Шоломий Ю., Стасюк С. ПОСТРОЕНИЕ РАСПРЕДЕЛЕННЫХ ОТКАЗОУСТОЙЧИВЫХ СИСТЕМ НА ОСНОВЕ РЕШЕНИЯ COROSYNC+РАСЕМАКЕР.....	21

REVIEW OF OPEN SOURCE ELECTRONIC DESIGN AUTOMATION SYSTEMS

O.V. Verovko

Electronic design automation system (EDA, ECAD) is a set of software tools to facilitate the development of electronic devices, creation of microcircuits and circuit boards. Such complex allows creation of a schematic circuit of the designed device using graphical interface, development and modification of database of radio-electronic components with verification of the integrity of the signals.

Modern software packages allow performing the automatic alignment of the elements, automatic dilution the tracks on the drawing of multilayer PCB by connecting terminals of radioelectronic components in accordance with the concept. The introduced scheme may be directly or via an intermediate link file («netlist») converted into a preform designed PCB with varying degrees of automation.

Electronic design automation systems may have the ability to model the developed device and research its work before it will be implemented in hardware.

The review of the open source electronic design automation systems intended for the construction of circuit boards is presented below.

Basically open source electronic design automation systems can be divided into two groups. First group is represented by programs developed by various companies associated with the production of circuit boards or selling components. The second group is the set of programs developed by amateurs.

The most popular representatives of the first group are **Express PCB, Pad2Pad and PCB Artist. [1-3]** Like many programs of this class, Express PCB, Pad2Pad and PCB Artist are created to promote the services of their companies and have reasonable limits, consisting in the fact that as the result we'll get the project in a proprietary format that can be sent only to a particular manufacturer of printed circuit boards. However the ability to get printed version of the circuit are sufficient for many developers.

There is also free ECAD, developed by commercial organization and available for widespread use. It is s **DesignSpark PCB that was proposed by RS Components. [4]** Simple and free registration on the website of the company is required do activate the program. DesignSpark PCB contains no restrictions on the number of circuit elements or on time of use. In contrast to the above programs, DesignSpark PCB does not try to bind users to a particular manufacturer and generates output files in popular industrial formats Gerber, DXF, Excellon, IDF, LPKF. This program is executed at a very high professional level and includes all the necessary components, such as schematic editor and editor of printed circuit boards. Using the schematic editor user can easily paint circuit and connections. The circuit may comprise a plurality of sheets linked together in a complete project. The functions of auto assembly and autorouting are also available. The support for popular simulators, such as LTSpice, LSSpice, TopSpice and TINA is implemented in DesignSpark PCB. The ability to import projects of these programs to create printed circuit boards is provided for users. Interface of the program includes a specialized calculator, which allows calculating the track width and the resistance, the optimal current density and the track temperature increasing.

The most widely used representatives of the second group are **gEDA, KiCad and FreePCB. Each of the presented programs** provides the standard set of documentation sufficient to order the PCB at a third party.

gEDA is a powerful open source EDA, designed to develop electronic circuits and printed circuit boards, licensed under the GNU GPL. [5] Founded by Ales Hvezda, gEDA runs under Linux, but some software included in the package, now ported to Windows. Currently gEDA project offers a developed set of free software tools for electronics design, including programs for

circuit design, attributes control, creation of lists of items and netlists in more than twenty formats, analog and digital simulation and PCB layout design. Using the programs gEDA it is possible to create printed circuit boards having up to 8 layers (and soon more), with an unlimited number of components and connections.

KiCad is an open source EDA, developed by Jean-Pierre Charras and licensed under the GNU GPL that provides ability to develop electronic circuits and printed circuit boards. [6] KiCad consists of schematic editor Eeschema, editor of PCB Pcbnew and the viewer of Gerber called Gerbview. Schematic editor provides creation of single-sheet and hierarchical circuits, electrical rules control (ERC), creation of a netlist for Pcbnew or Spice. PCB editor provides a development of board containing from 1 to 16 layers of copper and 12 technical layers (silk screen, solder mask etc), the generation of technological files for PCB and print of layers in the format PostScript. Gerber viewer allows viewing of Gerber-files.

FreePCB is boards' editor licensed under the GNU GPL and developed by Allan Wright. [7] However it is possible to import the netlist in PADS-PCB format of any third-party schematic editor, such as **TinyCAD**. In addition, live broadcast of the netlist from simulation Ltspice is also available. FreePCB have no ability to print, but for this purpose it is possible to use any free Gerber viewer such as ViewMate. Besides, the latest version of the program allows the export of PCB layout in graphic files PNG, which can be viewed and printed in a graphics program (eg Paint). FreePCB supports up to 16 layers allows working with both inch and metric units. PCB size can be up to 1524 × 1524mm. FreePCB generates output files in the extended Gerber RS274X format and drill files in the format Excellon. These standard CAM (Computer-aided manufacturing - an automated system of technological preparation of production) files are the necessary and sufficient documentation to order printed circuit boards from other producers.

Each of the presented electronic design automation systems provides sufficient functionality to perform the development of PCB.

References: 1. (2015) ExpressPCB website. [Online]. Available: <http://www.expresspcb.com/> 2. (2015) Pad2Pad website. [Online]. Available: <http://www.pad2pad.com/> 3. (2015) PCB Artist website. [Online]. Available: <http://www.4pcb.com/free-pcb-layout-software/index.html> 4. (2015) Design Spark website [Online]. Available: <http://www.rs-online.com/designspark/electronics/> 5. (2015) gplEDA website [Online]. Available: <http://www.gpleda.org/> 6. (2015) KiCadEDA website [Online]. Available: <http://www.kicad-pcb.org> 7. (2015) FreePCB website [Online]. Available: <http://www.freepcb.com/>

UDC 378.14

ANALYSIS AND COMPARISON OF OPEN SOURCE ELEARNING SYSTEMS

M.V. Verovko

Chernihiv National University of Technology

Due to a high popularity of the eLearning the development of the standards for elearning systems has been the task of great importance. The first edition of collection of standards and specifications developed for elearning systems was released in 2004 [1]. It is known as **SCORM** (*Sharable Content Object Reference Model*) and realized based on XML standard. **SCORM** includes requirements for the structure of courses and packages of educational material, for the interaction of content objects and learning management system and for ordering and navigation. The most widely used elearning systems that support the standard SCORM, are:

- Sakai;
- Moodle;
- ILIAS.

All presented elearning systems are open source software.

Sakai is free educational software platform developed by Sakai Project and distributed under the Educational Community License, which is the type of open source license. Systems of such type are often called Course Management Systems (CMS), Learning Management Systems (LMS) or Virtual Learning Environments (VLE) and can be used for teaching, research and collaboration [2].

Many of the features typical for course management systems, such as document distribution, a gradebook, discussion, live chat, assignment uploads, and online testing are available in Sakai. Additionally Sakai can be used not only as course management system, but also as a collaborative tool for research and group projects. Current function can be performed by using of the ability to modify the settings of the tools according to required roles and to change the privileges of different users. Wiki, mailing list distribution and archiving, and an RSS reader are also available in Sakai. Depending of requirements the core tools of Sakai can be augmented with additional modules intended for specific tasks e.g. sites for collaborative projects, teaching and portfolios.

Open source elearning system Sakai is used in more than 300 EU institutions.

Moodle (*Modular Object-Oriented Dynamic Learning Environment*) is a flexible open source learning platform classed as Learning Management System (LMS) or Virtual Learning Environment (VLE) depending of selected tasks. Moodle can be used as elearning system of University, Corporate training, School and Other sectors. It also can be used to develop a private website for dynamic online courses that is available due to realized comprehensive, customizable and secure learning management features.

As a leading virtual learning environment, Moodle can be used not only in education, but also in such environments as training, development and business settings. The features of the Moodle site could be extended by including additional plugins that are available in Moodle plugins directory. Users of Moodle system have the ability to add their own developed plugins. TCPDF library that allows the generation of PDF documents from pages are included as default. Moodle also provides such innovative learning features as calendar and Gradebook.

Moodle supports a set of interoperability features such as:

- authentication, based on LDAP;
- enrollment, based on IMS Enterprise;
- quizzes and quiz questions that allows import/export in a set of formats: GIFT, IMS QTI, XML and XHTML;
- integration with other Content Management Systems such as Drupal or Joomla.

Moodle is the most popular elearning system that nowadays has a user-base of 53,357 registered sites with 70,129,116 users in 7,840,470 courses in 225 countries [3].

ILIAS is an open source web-based Learning Management System that was one of the first Learning Management Systems used in universities. *ILIAS* is the acronym from German title “Integriertes Lern-, Informations- und Arbeitskooperations-System (Integrated Learning, Information and Work Cooperation System)”. Current elearning system provides learning content management and tools for collaboration, communication, evaluation and assessment.

There are a lot of features to develop and conduct online-courses, provided by ILIAS. ILIAS system also provides abilities to create learning content, offer assessments and exercises, run surveys and support communication and cooperation among users.

The main feature of ILIAS is usage of concept of Personal Desktop and Repository. Repository is the basic element of the ILIAS system that consists of all content, courses and other materials structured in categories and described by metadata. The other key element is Personal Desktop that represents the individual workspace of each learner, author, tutor and administrator. The Personal Desktop contains such elements as selected items from the repository, learning tools (mail, tagging, calendar etc.) and e-portfolio and personal blogs.

Initially developed within the VIRTUS project at University of Cologne ILIAS became one of the most distributed elearning systems.

The comparison of the considered open source elearning systems is presented below (Table 1).

Table 1. Comparison of open source elearning systems

	Sakai	Moodle	ILIAS
Type	Course Management System	Course Management System	Learning Management System
Written in	Java	PhP	PhP
License	Educational Community License	GNU General Public License	GNU General Public License
Operating system	Cross-platform	Cross-platform	Cross-platform
Standard	SCORM	SCORM	SCORM
Ukrainian language	No	Yes	Yes
Personal desktop	Yes	Yes	Yes
Mobile version	No	Yes	No

Each of the presented systems provides a wide range of abilities to perform learning process [4]. However the most popular is still Moodle system. But considering the tendency of the nearest years that the number of Moodle users became lower, the conclusion can be made that the popularity of other open source elearning systems increased.

Reference: 1. Advanced Distributed Learning (ADL), Sharable Content Object Reference Model (SCORM®) 2004 2nd Edition. Overview, 2004. 2. Michael Korcuska, Alan Mark Berg. Sakai Courseware Management: The Official Guide (1st ed.). Packt Publishing. – 2009 - p. 504. 3. (2015) Moodle website. [Online]. Available: <https://moodle.net/stats/> 4. V.V. Kazymyr, M.V. Tevkun, I.S. Posadska, O.P. Drozd. “Methods of collection data to evaluate the quality of distance learning system”. Вісник Чернігівського державного технологічного університету: зб. наук. праць. – Чернігів: ЧНТУ, – 2013.– № 69. – С. 144-153.

УДК 004.716

ВИКОРИСТАННЯ МОДУЛЯ ESP8266 У ВБУДОВАНИХ СИСТЕМАХ

Крива М.Я, Роговенко А.І.

Чернігівський національний технологічний університет, Україна

З розвитком вбудованих систем, поставало питання про децентралізацію вузлів систем, з зв'язку з тим щоб кожен вузол системи працював автономно і не залежав від інших. Тому зараз існує багато апаратних рішень для вирішення цього питання. Один із них є використання бездротової технології передачі даних – WiFi. Існує багато рішень на ринку, що дозволяють використовувати цю технологію у вбудованих системах, але майже всі з них мають велику вартість, тому перехід до бездротового способу передачі даних уповільнюється.

Модуль ESP8266 – це апаратно-програмна платформа схема якого включає в собі 32-бітний мікроконтролер сімейства Xtensa – Tensilica's L106 Diamond series зі своїм набором GPIO, в тому числі і SPI, UART та I2C, флеш-пам'ять та кварц. Функціональна схема мікроконтролера зображена на рисунку 1. В контролер вбудований власний TCP/IP стек та є апаратна підтримка протоколу передачі WiFi, що дозволяє використовувати модуль для обміну даними наприклад між клієнтом та датчиками руху, температури, пересування та іншими.

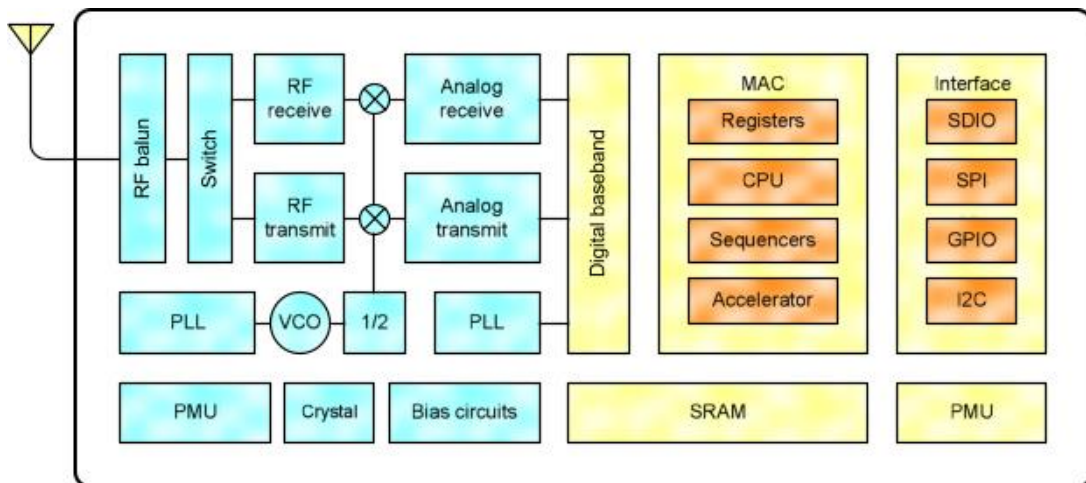


Рис. Функціональна схема модуля ESP8266

Завдяки своїй низькій ціні, ESP8266 може виступати конкурентом для таких аналогів як CC3000 та MTK7681. Модуль має власний SDK, що дозволяє самостійно використовувати увесь потенціал платформи, шляхом написання особистої прошивки. Наприклад: виступати в якості точки доступу або клієнтом у WiFi мережі, керування периферійними пристроями що підключені до портів вводу/виводу.

SDK написана на мові програмування – C, що є дуже важливим для керуванням п'ам'ятю контролера.

ESP8266 може використовуватися як UART-WiFi, перехідник. Тобто використовувати модуль можна як зовнішній пристрій для будь-якого мікроконтролера, який підтримує інтерфейс обміну даними UART, з метою керування роботою його роботою через термінал.

Окрім можливості написання власних прошивок для модуля, існують неофіційні прошивки, що дозволяють писати програмне забезпечення для ESP8266 на мовах програмування як Python (MicroPython) та Lua (NodeMCU, NodeLua). Але їх використання є раціональним лише у випадках невеликих проєктів. У випадку складних проєктів, доводиться користуватися офіційним SDK, бо сам модуль містить лише 40кб RAM.

Офіційний SDK включає в себе підтримку MQTT – протоколу обміну повідомленнями по TCP/IP, в нашому випадку, між сервером та пристроєм, що дає можливість надсилати дані з пристроїв до хмари. Крім того, SDK надає можливість записувати власні дані до флеш-пам'яті модуля, шляхом запису констант до секції даних. Але для спрощення роботи з файловою системою, на практиці використовується SPIFFS (SPI Flash File System), що надає змогу створювати та керувати файлами в файльовій системі, як зі звичайними файлами в мові C (fopen, fwrite, fread).

Але для побудови складних систем автоматизації та керування, на практиці, використовують зовнішній контролер, як сервер для обробки даних, а ESP8266 – як передавач інформації до цього сервера. Це пов'язано з тим, що досліджуваний модуль має не велику продуктивність, по відношенню до контролерів, що спеціалізуються на обробці інформації. Тому у більшості випадках, ESP8266 є лише способом посилання даних до централізованої системи обробки даних, а не повноцінним TCP сервером. Крім того, часто застосовується практика підключення ESP8266 до зовнішнього контролера як UART-WiFi міст, з метою керування системою віддалено, або для отримання/передавання певних даних через мережу.

Модуль ESP8266, за замовчуванням, підтримує спосіб керування через AT-команди, що подаються на нього через UART інтерфейс. Це дає змогу відразу використовувати увесь потенціал модуля. В залежності від версії заводського програмного забезпечення, AT-команди можуть змінюватись. На даний момент актуальною версією AT-команд для ESP8266 є 0.22.

Таким чином, можна відмітити, що використання програмно-апаратної платформи ESP8266 у сучасних вбудованих системах є дуже перспективним. Завдяки своїй вартості та доступності, вбудовані системи можна перевести на сучасний рівень, шляхом переходу зі звичайних методів передачі інформації по радіоканалам, до сучасних методів передачі інформації до комп'ютерних мереж.

Список використаних джерел: 1. *Режим* доступу: http://esp8266.ru/download/esp8266-doc/4B-AT-Espressif_AT_Command_Examples_v0.4.pdf. 2. *Режим* доступу: <http://habrahabr.ru/company/coolrf/blog/238443/>. 3. Таненбаум Е. Комп'ютерні мережі / Таненбаум Е. – Спб: Питер, 2003. – 992 с.

УДК 004.5

ПРОГРАММНЫЙ КОМПЛЕКС ФОРМИРОВАНИЯ ПРОГРАММ ПЕРЕДАЧ ДЛЯ ТЕЛЕВИЗИОННЫХ КАНАЛОВ.

Врублевский А.В.

Черниговский национальный технологический университет

В настоящее время у каждого канала есть своя телепрограмма. Использование телепрограммы обусловлено тем, что она существенно облегчает жизнь человека, при этом человек волен рассчитать свое время для просмотра нужной ему телепередачи. Количество которой увеличивается с каждым годом. В ходе работы были проанализированы разные системы веб-пауков для получения нужной нам информации о телепередачах.

Веб-паук – это программа, «ползающая» по Интернету определенным образом и с определенной целью. Цель может состоять в сборе информации или в понимании структуры какого-либо Web-сайта и его полезности. На применении пауков основаны современные поисковые машины, такие как Google и Yandex. Такие пауки автоматически извлекают данные из Web-сайта и передают их другим приложениям, которые индексируют контент этого Web-сайта с целью формирования наилучшего набора поисковых терминов.[1]

Данный веб-паук был написан посредством библиотеки SeleniumWebdriver, которую часто используют при парсинге сайтов

Selenium WebDriver, или просто WebDriver – это драйвер браузера, то есть не имеющая пользовательского интерфейса программная библиотека, которая позволяет различным другим программам взаимодействовать с браузером, управлять его поведением, получать от браузера какие-то данные и заставлять браузер выполнять какие-то команды. По своей сущности Selenium WebDriver представляет собой: спецификацию программного интерфейса для управления браузером, референсные реализации этого интерфейса для нескольких браузеров, набор клиентских библиотек для этого интерфейса на нескольких языках программирования. [2]

С драйвером пользователи не работают непосредственно. Они работают с прикладными программами, которые, посредством драйверов, взаимодействуют с теми или иными устройствами. Драйвер не имеет пользовательского интерфейса. Драйвер имеет только программный интерфейс, его назначение состоит в том, чтобы дать возможность прикладным пользовательским программам взаимодействовать с устройством.

Веб-паук обрабатывает информацию с сайта по нужным критериям, то есть можно установить, по какой дате нужна телепрограмма, установить рубрику нужных телеканалов, установить нужно ли парсить данные описания телепередачи или только название и время, без этих данных сбор информации проходит значительно быстрее, по причине того, что описания телепередач находятся на отдельных страницах, и их тогда парсер должен тоже

обходит. И еще у некоторых передач описания нет, и поэтому при обработке информации он парсит данные без информации о телепередачах.

Можно установить браузер, через который будет осуществляться сбор информации. По умолчанию выбран консольный, легкий и быстрый браузер PhantomJS, он предназначен специально для парсинга.

Phantomjs – это возможность работать с WebKit из консоли используя JavaScript и без браузера.[3]

Также в настройках устанавливается срока подключения к базе, так, как она изменяется в зависимости от того база лежит в месте с программой или же на sql server и имени sql server.

Данный проект находится в стадии разработки и поэтому ему не доступен некоторый функционал как у других веб-пауков. Он может быть улучшен, посредством добавления нескольких сайтов для парсинга[4], улучшением визуального интерфейса, расширением рубрик для парсинга и т.п.

Список использованных источников: **1.** Веб-паук: [электронный ресурс]. – Режим доступа: <http://www.ibm.com/developerworks/ru/library/l-spider/> **2.** Selenium webdriver: [электронный ресурс]. – Режим доступа: <http://habrahabr.ru/post/152971/> **3.** Консольный браузер для парсинга Phantomjs [электронный ресурс]. – Режим доступа <http://phantomjs.org/> **4.** Взгляд на сайт глазами поискового робота: [электронный ресурс]. – Режим доступа: <http://vitod.ru/statiy/vzglyad-na-sajt-glazami-poiskovogo-robota.html>

УДК 004.5

РОЗРОБКА ANDROID-ДОДАТКУ ДЛЯ ВЗАЄМОДІЇ З СЕРВЕРНОЮ ЧАСТИНОЮ СИСТЕМИ КЕРУВАННЯ ПАТРУЛЯМИ ДОБРОВІЛЬНОЇ НАРОДНОЇ ДРУЖИНИ

Мостіпан А. В.

*студент 3-го курсу Славутиської філії Національного технічного університету України
«Київський політехнічний інститут»*

науковий керівник: **Савельєв М. В.**, ст. викладач СФ НТУУ «КПІ»

Постановка проблеми у загальному вигляді. Останнім часом, у зв'язку з підвищенням кримінального рівня на території України все більшої актуальності набуває проблема підвищення безпеки мирного населення. У зв'язку з цим, виникає необхідність у підвищенні якості та результативності роботи правоохоронних органів.

Аналіз останніх досліджень і публікацій. Вирішенням такого питання вже займалася московська компанія «М2М телематика» на замовлення міської влади Москви, у напрямку забезпечення спецтранспортних засобів підрозділів Міністерства внутрішніх справ (МВС).

Виділення невирішених раніше частин загальної проблеми. Зазначена московська компанія пропонувала використовувати систему управління мобільними підрозділами за умови, що навігаційні пристрої будуть розміщені безпосередньо в автотранспортних засобах підрозділів МВС. Однак, вона не розглядала можливість використання цих засобів безпосередньо представниками МВС та добровільною народною дружиною (ДНД). Тому мета нашого проекту полягає у розробці програмного забезпечення автоматизованої системи керування мобільними патрулями (СКМП) добровільної народної дружини, на що не вказували попередні дослідники.

Виклад основного матеріалу. Для розв'язання зазначеної проблеми нами було вирішено створити СКМП патрульно-постової служби. Запропонована система передбачала реалізацію проекту на основі клієнт-серверної архітектури. Клієнтська складова являє собою мобільний додаток для операційної системи Android.

Основна ідея мобільного додатку полягає в тому, що патрульний, який використовує мобільний-додаток, підтримує зв'язок з командним центром, повідомляючи йому про свій стан та отримує координати місця події. Патрульний також може бачити автоматично встановлений оптимальний маршрут до отриманого місця.

При створенні мобільного додатку, постало питання у виборі операційної системи на якій він буде реалізований. Зараз у світі існують три найвідоміші та широко розповсюджені операційні системи для мобільних пристроїв:

- Android;
- iOS;
- Windows Phone.

Нами було обрано саме Android з огляду на те, що дана операційна система займає більшу частину світового ринку та має наступні переваги порівнянні з іншими:

- відкритість платформи – що надає можливість легкого створення додатків;
- порівняно низька ціна на пристрої, які використовують дану операційну систему.

Тож мобільний додаток створено на базі операційної системи Android, яка, в свою чергу, передбачає використання об'єктно-орієнтованої мови програмування Java[1] та розширеної мови розмітки XML. Для реалізації обміну даними між мобільним додатком та сервером був використаний текстовий формат обміну даними JSON[2].

Нижче наведено вимоги до Android додатку: 1) можливість зв'язку з сервером для обміну даними та координатами місцезнаходження мобільного пристрою; 2) сповіщення стану мобільного патрулю; 3) відображення на карті пристрій, на якому встановлено додаток; 4) візуалізація отриманих координат подій та маршруту до них; 5) документування необхідних даних; 6) створення протоколу правопорушень.

Використання карти на операційній системі Android можливе завдяки обраному сервісу Google Maps, а саме інтерфейсу програмування додатків «Google Maps Android API v2» [3]. Останній дозволяє автоматично отримувати доступ до серверу карт Google, завантажувати та відображати карти, маніпулювати картографічним зображенням, показувати збережені координати у вигляді окремих точок з назвою та описом.

На даному етапі проекту, який пов'язаний з розробкою Android-додатку, вдалося реалізувати: 1) зв'язок з «веб-додатком»; 2) візуалізацію на карті мобільного пристрою; 3) візуалізацію отриманих координат місця подій та маршруту до них.

Такий підхід до реалізації проекту є, на наш погляд, ефективним, оскільки він надає можливість диспетчеру відслідковувати місцезнаходження усіх патрульних, та відправляти патрульних на завдання через картографічний сервіс. Диспетчер також може відслідковувати кількість правопорушень у місті, та маршрути патрульних, скласти оптимальні маршрути до пункту призначення патрульного.

Висновки та перспективи подальших досліджень. На даному етапі роботи над проектом створено прототип програмного комплексу, який потребує більш точного опрацювання специфічних деталей, які повинні бути визначені в результаті взаємодії розробників із представниками органів правопорядку. На нашу думку, запропонована система спрощує роботу добровільної народної дружини, оскільки має такі переваги:

- дозволяє дізнатися найкоротший маршрут та точне місце, куди необхідно пройти патрульному;
- при оформленні правопорушень можна легше заповнювати всі форми та бланки, автоматично генерувати звіти про правопорушення, що призводить до зменшення обсягів рукописних звітів;
- при оформленні правопорушення, до бланку можна додати фото та відео матеріали.

Таким чином, розроблена система має вагомим практичне значення, оскільки спрямована на допомогу та оптимізацію роботи співробітників МВС та ДНД.

Обраний частина проекту потребує подальшого дослідження, оскільки ефективність його реалізації на практиці передбачає вивчення особливостей картографії, комп'ютерної

обробки великих обсягів інформації та проведення моніторингу статистичних даних правопорушень.

Список використаних джерел: 1. «Java Book». – [Електронний ресурс]. – Режим доступу: <http://www.java2s.com/Book/Java> 2. «JSON». – [Електронний ресурс]. – Режим доступу: <http://www.json.org/> 3.«Google Developers».– [Електронний ресурс]. – Режим доступу: <https://developers.google.com/maps/documentation/android>

УДК 004.451.33

ОТКРЫТЫЙ СБОРЩИК МУСОРА ВОЕНМ GC ДЛЯ ЯЗЫКОВ C/C++

Тур Г.Н.

Черниговский национальный технологический университет

Для языков программирования с ручным управлением памятью характерными являются ошибки при работе с памятью, такие как утечки памяти, использование памяти после ее освобождения и прочие. Решением данной проблемы является использование сборщиков мусора либо подсчет ссылок на объекты.

Для решения проблем ручного управления памятью при работе с языками C и C++, был создан сборщик мусора Boehm GC. Сборщик мусора распространяется свободно и является кроссплатформенным, он может работать на большинстве систем которые поддерживают виртуальную память.

Данный сборщик мусора в своей работе использует алгоритм Mark-sweep. Идея алгоритма заключается в том, что вся куча представляется как граф, если к некоторому блоку памяти не существует путь в графе от указателя на стеке, то данный блок можно освобождать, поскольку он уже не используется и является недостижимым. При активации сборщик работает в два этапа. В первом он отмечает все достижимые объекты, во втором этапе адреса всех недостижимых блоков памяти помещаются в список свободных блоков памяти. Наибольшей трудностью при этом является нахождение указателей на стеке и в блоках памяти, поскольку в языках C и C++ нет разницы между указателями и числами. Алгоритм работы сборщика мусора является не копирующим, что приводит к такому явлению как фрагментация кучи, что является причиной падения производительности после длительного времени работы программы.

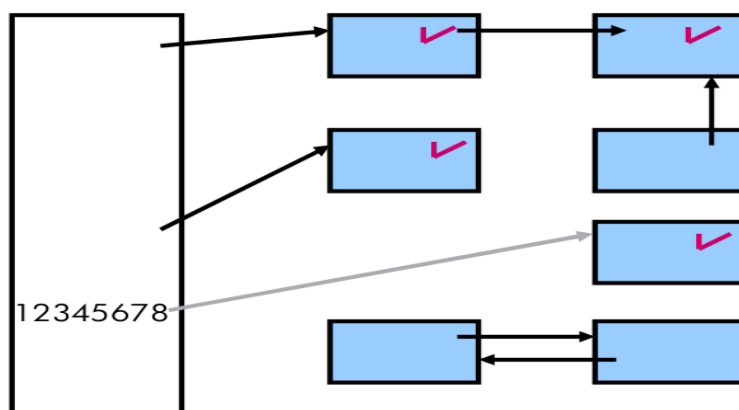


Рис. Структура кучи и стека при первом этапе сборки мусора

Сборщик мусора имеет два интерфейса один для C, другой для C++. Интерфейс для языка C является предельно простым, для инициализации используется функция `GC_INIT()`, для выделения памяти `GC_MALLOC()`. Память можно также освободить вручную используя `GC_FREE()`, но зачастую это ненужно. Также существуют функции для регистрации финализаторов – функций которые вызываются при освобождении блока

памяти.

Интерфейс для языка C++ является менее тривиальным. Основной причиной этого является присутствие деструкторов в языке C++. Сборщик мусора не может обеспечить естественный порядок вызова деструкторов, как бы их вызывал сам программист. Также деструктор может быть вызван спустя неопределенное время после того как объект вышел из области видимости, что может привести к задержке освобождения ресурсов операционной системы, таких как файловые дескрипторы, примитивы синхронизации и прочих. Поэтому рекомендуется выносить подобную логику из деструкторов, что противоречит основной идиоме C++ RAII.

Также данный сборщик мусора позволяет работать в режиме поиска утечек памяти, заменив все вызовы malloc на GC_MALLOC, что возможно при линковке с параметром -DREDIRECT_MALLOC=GC_malloc. Такой режим работы позволяет упростить отладку приложений. При этом он детектирует адрес объекта в памяти, его размер, а также на платформе Linux/x86 есть возможность отследить цепочку вызовов при выделении данного блока памяти. При этом можно также освобождать недостижимые участки памяти.

С появлением стандарта языка C++11 необходимость в сборщике мусора частично отпала из-за появления умных указателей которые решают проблему с вызовом деструкторов.

Тем не менее при использовании сборщика мусора все же возможно допустить утечку памяти, причем найти ее гораздо сложнее. Типичной проблемой является хранение ссылок на неиспользуемые объекты в контейнерах, что приводит к утечкам памяти.

Список использованных источников: 1. The Boehm Collector for C and C++. [Электронный ресурс] – Режим доступа: <http://www.drdoobs.com/the-boehm-collector-for-c-and-c/184401632> 2. A garbage collector for C and C++. [Электронный ресурс] – Режим доступа: <http://www.hboehm.info/gc/>.

УДК 004.738.5

ОБЕСПЕЧЕНИЕ ВЗАИМОДЕЙСТВИЯ УСТРОЙСТВ «ИНТЕРНЕТА ВЕЩЕЙ» С СЕРВИСОМ THINGSPEAK

Р.Б. Андрущенко, Д.О. Ульченко

Черниговский национальный технологический университет

Роб Ван Краненбург говорил: «Интернет вещей – это концепция пространства, в котором все из аналогового и цифрового миров может быть совмещено – это переопределяет наши отношения с объектами, а также свойства и суть самих объектов» [1].

Понятие «интернета вещей» – одно из наиболее цитируемых понятий в ИТ [2]. В наши дни рынок интернета вещей (или IoT – Internet of Things) растет очень быстро и динамично. Он испытывает значительное влияние под действием мобильных и облачных технологий и, по предсказаниям аналитиков, будет очень быстро развиваться, так как находится пока на начальном этапе своего развития и имеет на данный момент практически неограниченный потенциал возможностей [2].

На протяжении последних пяти лет во всем мире растет интерес к интернету вещей. В мире проводятся привлекающие внимание международные форумы, конгрессы и саммиты, посвященные интернету вещей, а в мировых СМИ регулярно обсуждаются последствия появления интернета вещей и его влияние на социальные преобразования. Существуют организации, занимающиеся данной тематикой, такие как JCA-IoT (группа по совместной координационной деятельности в области интернета вещей), IoT-GSI (глобальная инициатива по стандартам интернета вещей). Исследования в сфере интернета вещей ведут CDMA Development Group, IPSO Alliance, Dynamic Spectrum Alliance, EnOcean

Alliance, GSMA, HART Communication Foundation, IEEE Standards Association, International Society of Automation, IPv6 Forum, Modbus Organization, Object Management Group, Open Geospatial Consortium, OPC Foundation, Wave2M, Weightless Special Interest Group, ZigBee Alliance [3].

Некоторым прорывом в развитии беспроводных коммуникаций между устройствами интернета вещей посредством WiFi можно назвать появление на свет в 2014 году модуля компании Espressif Systems – ESP8266, который при своей дешевизне (в марте 2015 года его стоимость составляла 3\$) позволяет помимо клиентских функций также реализовывать сервер с возможностью одновременного подключения нескольких клиентов и WiFi точку доступа. Помимо этого, данный модуль может выступать в роли посредника для устройств, имеющих UART-интерфейс, позволяя им также осуществлять беспроводные коммуникации. Модуль имеет встроенный микропроцессор, прошивку для которого можно написать самостоятельно. Данная возможность позволяет вообще не использовать дополнительные микроконтроллеры для обработки получаемых или передаваемых данных, что дает возможность проектировать еще более дешевые устройства [4,5].

По заявлению производителя, возможны следующие варианты применения модуля: домашняя автоматизация, mesh-сети, промышленное WiFi-управление, IP-камеры, сети датчиков, ключи безопасности, мобильные миниатюрные устройства и т.д. [5].

При наличии точки доступа с выходом в сеть Интернет, устройство на основе ESP8266 может взаимодействовать с ресурсами сети Интернет. В модуле присутствует внутренняя реализация стека протоколов TCP/IP [4,5].

Как вариант использования модуля, имеющего выход в сеть Интернет, можно рассмотреть взаимодействие с сервисом ThingSpeak. ThingSpeak – открытый проект, исходные тексты которого доступны на GitHub (<https://github.com/iobridge/thingspeak>). С помощью данного сервиса и модуля ESP8266 можно создать программно-аппаратный комплект, который позволяет реализовать удаленное взаимодействие и управление теми или иными системами, даже если источник управления и ведомые устройства находятся на противоположных точках земного шара.

Например, рассмотрим реализацию устройства, которое фиксирует температуру в помещении и отправляет данные на сервис ThingSpeak. По этим данным пусть строится график изменения температуры по времени, который можно посмотреть с любого устройства, имеющего выход в сеть Интернет.

ThingSpeak может обрабатывать GET- и POST-запросы. Результат запроса сервис может возвращать в формате XML, JSON или же простым текстом с ответом (например, «0» – в случае неудачи, «id» - в случае успеха).

Основным интерфейсом взаимодействия с сервисом являются так называемые «каналы» (channels), которые могут быть публичными и приватными. Каналы представляют собой набор упорядоченных по времени структур с определенными полями. В каждой структуре присутствуют стандартные поля, такие как «дата и время измерения», «название» и т.п. Кроме того можно добавлять пользовательские поля (fields). В этих полях хранятся пользовательские данные. По каждому каналу можно создать графики по нужному пользовательскому полю структуры. Для данного примера в простейшем случае достаточно объявить одно поле, в котором будет храниться измеренная температура.

Для обеспечения безопасности, вызов большинства методов сервиса требуют API-key. Зная этот ключ, можно производить изменения в своем канале. Пример GET-запроса на сохранение измеренной температуры приведен ниже:

```
http://184.106.153.149/update?api_key=XXXXXXXXXXXX&field1=23.4
```

Ответом на данный запрос является ID сохраненной структуры.

Чтобы получить ранее измеренные данные, можно послать запрос следующего вида:

```
http://184.106.153.149/channels/<channelID>/feeds?results=<N>
```

<channelID> – ID канала,

<N> – количество записей (в ответе будут отображены N последних записей) [6].

Пример ответа слишком громоздок, его можно найти по ссылке <https://thingspeak.com/docs/channels>.

В разработанном устройстве-сенсоре нужно отправлять только измеренную температуру. А результаты измерений можно посмотреть с любого устройства, имеющего выход в сеть Интернет.

Схема электрическая принципиальная изображена на рисунке 1. В ней присутствуют датчик температуры DS18B20, микроконтроллер ATMEGA8L с обвязкой в виде нескольких резисторов и конденсаторов, модуль ESP8266 и один светодиод, сигнализирующий о состоянии устройства. Схема питается напряжением 3.3 вольт. Внешний кварц не используется.

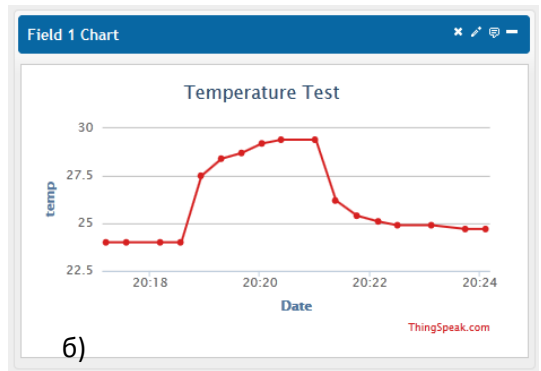
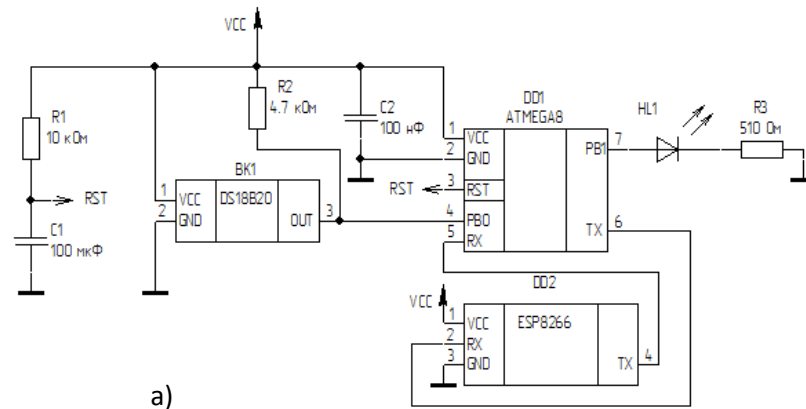


Рис. Схема устройства (а) и результат измерений (б)

Стоимость компонентов: модуль ESP8266 – 3\$, микроконтроллер ATMEGA8L – 1.5\$, датчик DS18B20 – 1.10\$, резисторы, светодиод и конденсаторы – менее 0.1\$. Таким образом, общая стоимость всех компонентов – менее 6\$.

Программа (с учетом программной реализации интерфейса 1-Wire) занимает 63% ресурсов микроконтроллера, EEPROM – 6.2%. Устройство можно реализовать и без дополнительного микроконтроллера. Тогда его стоимость снижается еще на 1.5\$ и составляет 4\$. Микроконтроллер использовался с целью показать реализацию взаимодействия ESP8266 с другими устройствами по UART-интерфейсу.

Таким образом, уже сейчас можно проектировать и создавать «умные» устройства с минимальными временными и финансовыми затратами, которые могут взаимодействовать между собой и с сетью Интернет по беспроводной связи. Возможность проектирования эффективных устройств, использующих сервис ThingSpeak, была доказана путем создания работающего испытательного образца, работа которого была успешно протестирована.

Список использованных источников: 1. Электронный ресурс [geektimes.ru](http://geektimes.ru/post/149593/) – <http://geektimes.ru/post/149593/> 2. Электронный ресурс [compress.ru](http://compress.ru/article.aspx?id=24290) – <http://compress.ru/article.aspx?id=24290> 3. Электронный ресурс [itunews.itu.int](https://itunews.itu.int/ru/Note.aspx?Note=4373) – <https://itunews.itu.int/ru/Note.aspx?Note=4373> 4. Espressif Systems ESP8266EX Beginner's

УДК 004.5

РОЗРОБКА ВЕБ-ДОДАТКУ ДЛЯ ВЗАЄМОДІЇ З КЛІЄНТСЬКОЮ ЧАСТИНОЮ СИСТЕМИ КЕРУВАННЯ ПАТРУЛЯМИ ДОБРОВІЛЬНОЇ НАРОДНОЇ ДРУЖИНИ

Хворостенко Р.Ю.

Славутицька філія Національного технічного університету України

«Київський політехнічний інститут»

науковий керівник: **Савельєв М. В.**, ст. викладач СФ НТУУ «КПІ»

Постановка проблеми у загальному вигляді. У зв'язку з погіршенням економічного становища на території України відбувається підвищення кримінального рівня, тому постає потреба підвищення ефективності роботи правоохоронних органів.

Аналіз останніх досліджень і публікацій. Вирішенням такого завдання вже займалася московська компанія «М2М телематика» на замовлення міської влади Москви, у напрямку забезпечення транспортних засобів підрозділів Міністерства внутрішніх справ (МВС) навігаційною системою, що значно підвищила швидкість реагування правоохоронних органів на надзвичайні ситуації.

Виділення невирішених раніше частин загальної проблеми. Московська компанія «М2М телематика» пропонувала використовувати систему управління мобільними патрулями за умови наявності навігаційних пристроїв, що розміщені безпосередньо в транспортних засобах підрозділів МВС. Однак не розглядалася можливість використання системи безпосередньо представниками МВС та добровільною народною дружиною (ДНД). Тому мета проекту полягає у розробці програмного забезпечення автоматизованої системи керування мобільними патрулями (СКМП) добровільної народної дружини, яку можна використовувати незалежно від наявності навігаційних пристроїв транспортних засобів.

Виклад основного матеріалу. Для розв'язання зазначеної проблеми нами було вирішено створити СКМП патрульно-постової служби, яку можна використати незалежно від наявності навігаційних пристроїв транспортних засобів. Дана система передбачала реалізацію проекту на основі клієнт-серверної архітектури. Серверна сторона являє собою веб-додаток, та базу даних.

Основна ідея веб-додатку полягає в тому, що диспетчер, який знаходиться в диспетчерському центрі використовує веб-додаток для взаємодії за патрульними. додатку полягає в тому, що патрульний, який використовує мобільний-додаток, підтримує зв'язок з командним центром, повідомляючи йому про свій стан та отримує координати місця події. Патрульний також може бачити автоматично встановлений оптимальний маршрут до отриманого місця.

При створенні веб-додатку, постало питання у виборі веб-фреймворку завдяки якому буде реалізовуватись веб-додаток. Порівнюючи багаточисленні веб-фреймворки, було обрано веб-фреймворк Django[1]. Цей вибір пов'язаний з тим, що Django безкоштовний веб-фреймворк, який дозволяє швидко і якісно розробляти веб-додатки. Веб-фреймворк Django передбачає використання мови програмування Python. Для взаємодії серверної сторони із клієнтською використовуються повідомлення текстового формату JSON[2]. Серверна частина створювалась на базі операційної системи Linux.

Для взаємодії з навігаційною системою було обрано картографічний сервіс Google Maps, та інтерфейс програмування веб-додатків до даного картографічного сервісу «Google Maps API v3»[3]. Останній надає змогу взаємодіяти підключати картографічний сервіс

Google Maps до веб-додатку, маніпулювати картографічним зображенням, розставляти маркери по координатам, та знаходити оптимальні маршрути між маркерами.

Нижче наведено вимоги до веб-додатку: 1) можливість диспетчеру відслідковувати місцезнаходження всіх патрульних; 2) сповіщення диспетчера про стани мобільних патрулів; 3) можливість відслідковувати маршрути патрулів; 4) відправка координат місцезнаходження, з коротким описом, надзвичайної ситуації; 5) можливість знаходження оптимальних маршрутів між патрульними і місцезнаходженням надзвичайної ситуації; 6) збереження в базі даних протоколів правопорушень, отриманих від мобільних пристроїв патрульних; 7) візуалізація кримінального рівня різних районів міста; 8) захищеність веб-додатку від втручання сторонніх користувачів.

На даному етапі проекту, який пов'язаний з розробкою веб-додатку, вдалося виконати всі основні вимоги.

Такий підхід до реалізації проекту є, на наш погляд, ефективним, оскільки він надає можливість диспетчеру відслідковувати маршрути усіх патрульних, відправляти патрульних на завдання через картографічний сервіс, що може значно заощадити час необхідних для сповіщення патрульних. Також існує можливість зберігати всі протоколи в базі даних, та відеоматеріали, аудіоматеріали, фотоматеріали, які прикріпленні к відповідним протоколам. Диспетчер може відслідковувати кількість правопорушень у місті, та відслідковувати кримінальний рівень різних районів міста. Веб-додаток здатен будувати оптимальні маршрути до пунктів призначення патрульних.

Висновки та перспективи подальших досліджень. На даному етапі роботи над проектом створено прототип програмного комплексу, який потребує більш точного опрацювання специфічних деталей, які повинні бути визначені в результаті взаємодії розробників із представниками органів правопорядку. На нашу думку, запропонована система підвищує ефективність роботи добровільної народної дружини, оскільки надає змогу:

- дізнатися оптимальний маршрут та точне місце, куди необхідно пройти патрульному для виконання завдання;

- при оформленні правопорушень можна легше заповнювати всі форми та бланки, автоматично генерувати звіти про правопорушення та відправляти все в єдину базу даних, що призводить до зменшення обсягів рукописних звітів;

- при оформленні правопорушення, до бланку можна додати фото та відео матеріали;

- відслідковувати кримінальний рівень різних районів міста, що дозволить краще проробити маршрути патрулювання робітників правоохоронних органів.

Таким чином, розроблена система має вагомим практичне значення, оскільки спрямована на допомогу та оптимізацію роботи співробітників МВС та ДНД.

Обраний частина проекту потребує подальшого дослідження, оскільки ефективність його реалізації на практиці передбачає вивчення особливостей картографії, комп'ютерної обробки великих обсягів інформації та проведення моніторингу статистичних даних правопорушень.

Список використаних джерел: 1. «Django Book». – [Електронний ресурс]. – Режим доступу: <http://djangobook.ru/> 2. «JSON». – [Електронний ресурс]. – Режим доступу: <http://www.json.org/> 3. «Google Maps API v3». – [Електронний ресурс]. – Режим доступу: <https://developers.google.com/maps/web/>

ОСОБЕННОСТИ РАЗРАБОТКИ СИСТЕМ НА МОДУЛЕ НА ПРИМЕРЕ "FreePCX"

Д. Д. Грицай

Научно-исследовательский институт компьютерных технологий

Технологии «Систем-на-модуле» известны уже более 10 лет и сегодня стали особенно актуальны в связи с потребностями рынка в компактных системах со сверхмалым энергопотреблением, предназначенных для мобильных приложений. Благодаря технологии «систем-на-модуле» разработчики встраиваемых систем имеют возможность ускорить вывод конечного изделия на рынок, используя при этом модули разных производителей. Концепция СнМ позволила снизить затраты на проектирование и производство, устранив факторы риска при разработке системы. Поскольку в стандартизированных модулях СнМ уже интегрированы чипсет, процессор, память и другие необходимые компоненты, то разработчики могут сконцентрировать усилия на создании собственно встраиваемого приложения. Кроме того, стандартизированные модули СнМ обеспечивают высокую масштабируемость изделий, включая поддержку процессоров разных типов, что очень полезно OEM-производителям, поскольку позволяет легче переносить приложения между различными архитектурами.

Такой подход позволяет вывести готовую систему на рынок гораздо быстрее, чем при разработке всей системы с нуля. Кроме того, благодаря стандартизации, отпадает необходимость привязываться к какому-то одному поставщику модулей. В принятом отраслевом стандарте определены размеры модулей СнМ, физическое расположение разъемов и назначение контактов.

Сегодня платформа СнМ находит все более широкое признание у специалистов, задача которых заключается в создании таких встраиваемых решений, которые отличаются малыми габаритами, высокой производительностью и низким энергопотреблением. Сочетание таких достоинств модулей СнМ позволило на их основе строить системы, отвечающие требованиям по скорости вывода изделия на рынок, снижению издержек и минимизации рисков при разработке, упрощению возможной модернизации и увеличению срока эксплуатации. Все это – залог дальнейшего роста рыночной доли модулей СнМ и оборудования на их основе.

Была разработана вторая редакция Системы-на-Модуле FreePCX, которая предназначена для тестирования встраиваемых микропроцессорных устройств, как в качестве самостоятельного вычислительного узла, так и в связке с готовыми, уже имеющимися, вычислительными модулями.

Назначение системы осталось прежним – изучение, настройка и отладка систем на кристалле, а также тестирование программного обеспечения для встроенных систем. В новой версии расширился набор периферийных устройств, а, следовательно, расширилась и область применения связанная с установленными интерфейсами.

Система-на-Модуле (СнМ) – функционально законченный фрагмент устройства, оформленный в виде отдельного модуля (печатной платы) изготовленного в соответствии с требованиями заказчика, предназначенный для использования в аппаратных комплексах. Модули позволяют разбивать сложные задачи на более мелкие. СнМ имеют низкое тепловыделение, компактный размер и большой набор интерфейсов, выводимых через стандартные разъемы. Благодаря модулям в рамках одной встраиваемой системы, можно удачно совместить компьютерную функциональность и клиентское приложение, а использование в качестве основного программируемого элемента ПЛИС позволяет гибко конфигурировать систему под конкретные задачи. Как и прежде, всю систему FreePCX можно разделить на три подсистемы:

- Аппаратная подсистема;

- Подсистема представления устройства в ПЛИС (СнК);
- Подсистема программного обеспечения.

Для максимального «насыщения» на печатной плате были установлены следующие периферийные устройства: программируемая логическая интегральная схема – Xilinx Spartan 3A 1400 (XC3A1400A-5FT256I), динамическая память DDR SDRAM объемом 64 Мегабайта – микросхема фирмы Micron MT46V64M8, энергонезависимая NOR флэш-память размером 8 Мегабайт – Spansion S29GL064N90, SPI флэш-память размером 32 Мегабита для хранения конфигурационного файла ПЛИС – Atmel AT45DB321D, SPI флэш-память размером 32 Мегабита для пользовательских данных – Atmel AT45DB321D, I2C EEPROM объемом 1 Мегабит для хранения логов и конфигурационных настроек – Atmel AT24C1024, 10/100 Мегабитный сетевой интерфейс МП основанный на PHY SMSC LAN8713A, VGA выход с резистивным ЦАП для вывода информации на монитор, два интерфейсных разъема PS/2 для подключения устройств ввода – мыши и клавиатуры, интерфейс USB HOST версии 2.0 реализованный на PHY SMSC USB3300 (интерфейс подключения ULPI) для подключения различных USB устройств, разъем для подключения карт памяти – MicroSD, цифровой I2C датчик температуры с аппаратных watchdog'ом – NXP LM75A, два отладочных интерфейса RS232 (RX и TX), часы реального времени фирмы Dallas – DS1338Z, аппаратный расширитель портов с интерфейсом I2C производителя Texas Instruments PCF8574A, аппаратный преобразователь интерфейсов – I2C <-> OneWire – DS2482, цифровой датчик температуры с W1 интерфейсом – DS18B20Z, динамик Buzzer, разъем JTAG, кварцевые генераторы на 50 и 25 МГц, светодиоды и кнопки общего назначения, выключатель; DC-DC преобразователи 3.3 В 2.5 В 1.2 для ПЛИС и периферийных устройств; Источник Vref напряжения для DDR – 1.25 В.

При таком большом наличии периферийных устройств можно решить большинство задач, связанных с применением встраиваемых устройств.

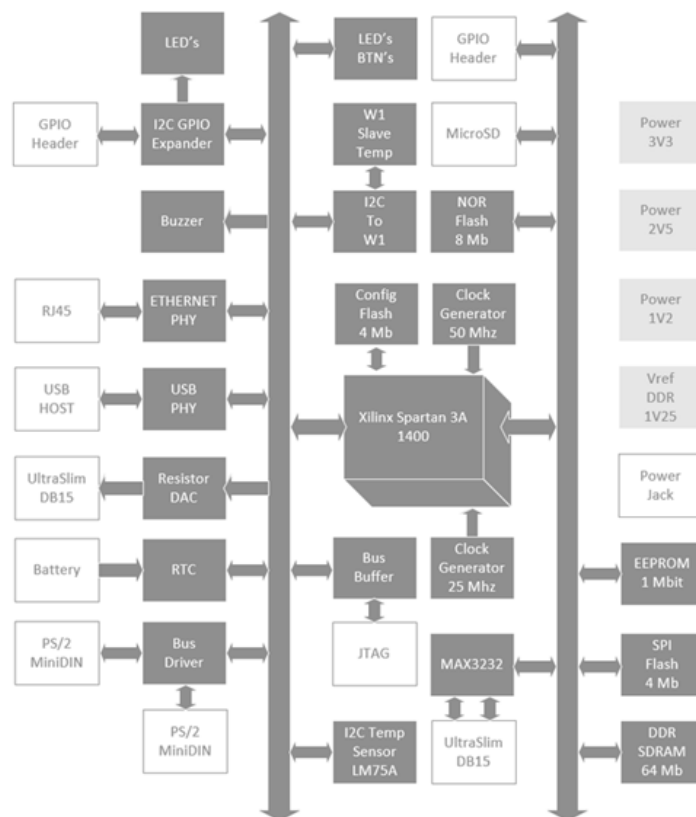


Рис. Структура FreePCX

При выборе синтезируемого на ПЛИС процессора для Системы-на-Модуле был проведен анализ существующих решений. Наиболее предпочтительными оказались: LEON3, Microblaze и OpenRISC. Выбор в очередной раз пал на фирменный процессор

Microblaze – официально поддерживаемый и развиваемый фирмой Xilinx. Основной аргументацией послужило оптимальное использование ресурсов и задействование специфических аппаратных блоков (например, как MULT18), благодаря которым экономится основные части ПЛИС – Slices и LUT. А также наличие большинства готовых IP-CORE's что существенно увеличивает скорость и стоимость конечного изделия. Все описанные модули, кроме контроллеров USB HOST и SD/MMC, можно найти в составе Xilinx EDK 14.X.

Данная Система-на-Кристалле совмещает следующие IP-CORE's:

Ядро процессора Microblaze;

Системные шины PLB и LMB;

Контроллер внешней флэш-памяти;

Мульти портовый контроллер внешней динамической памяти;

Блочную RAM и контроллер доступа к ней;

Контроллер отладки реального времени;

Модуль сетевого интерфейса Ethernet;

Модуль работы с внешними флэш картами SD и MMC (Поддержка режима DMA);

Контроллер вывода информации на монитор с интерфейсом VGA (Поддержка режима DMA);

Контроллер USB 2.0 (Поддержка режима DMA);

Модуль таймера;

Контроллер прерываний, сторожевой таймер – Watchdog, модуль GPIO (General Purpose Input/Output);

Контроллер внешних устройств ввода с интерфейсом PS/2, SPI контроллер, I2C контроллер, 2 модуля UART, модуль тактирования и модуль сброса.

Особого внимания заслуживает IP-CORE ядра Microblaze, которое было настроено на максимальную производительность, которую позволяла вместить в себя сама ПЛИС. Ядро включает такие модули: модуль расширенного FPU (Extended Float Point Unit), модуль 64-х разрядного умножителя (Integer Multiplier), модуль деления (Integer Divider), модуль Pattern Comparator, модуль Reverse Load/Store and Swap Instructions, модуль Barrel Shifter, модуль Additional Machine Status Register Instructions, модули аппаратных исключений, кэш инструкций: 8 Кбайт, кэш данных: 8 Кбайт, поддержка Write-Back Storage Policy, Full MMU, модуль аппаратной отладки, Full Processor Version Registers.

Благодаря наличию таким расширениям, а особенно как модули операций с плавающей запятой и деления, существенно повышается общая производительность системы при вычислительных операциях и работой с мультимедийными данными.

Основным программным элементов СнМ является операционная система GNU/Linux и универсальный загрузчик второго уровня U-Boot.

При включении модуля с SPI Flash загружается конфигурационный файл в ПЛИС, в котором проинициализирована блочная память. В BRAM располагается загрузчик первого уровня – SrecBootloader который ищет в NOR Flash образ загрузчика второго уровня – U-Boot и загружает его в динамическую память по указанным адресам. Во второй микросхеме SPI Flash находятся системные переменные загрузчика и ядро операционной системы. После предварительной инициализации периферийных устройств U-Boot загружает ядро ОС в ОЗУ, которое также находится в последовательной памяти и передает ему управление. Основная файловая система (RootFS) размещена в NOR Flash, а пользовательские приложения, занимающие основную часть дискового пространства, находятся на накопителе microSD, которая монтируется в конце этапа загрузки.

Стоит также отметить тот факт, что все, периферийные устройства, имеющиеся на модуле поддерживаются ядром Linux, и могут быть перенастроены в любой момент из пользовательских приложений.

Список использованных источников: 1. Таненбаум Э. Архитектура компьютеров. СПб.: Питер, 2007. - 848 с. 2. Бибило П.Н. Основы языка VHDL. / П.Н. Бибило - М.: Издательство ЛКИ, 2007 г. - 328с. 3. Электронный ресурс linux-sunxi.org - <http://linux-sunxi.org/U-Boot>

УДК 004.056

ПОСТРОЕНИЕ РАСПРЕДЕЛЕННЫХ ОТКАЗОУСТОЙЧИВЫХ СИСТЕМ НА ОСНОВЕ РЕШЕНИЯ COROSYNC+RACEMAKER

Шоломий Ю.Е., Стасюк С.С.

Черниговский национальный технологический университет

Кластер (применительно к компьютерам) - это группа серверов и других ресурсов, подключенных посредством аппаратных средств, сетей и программного обеспечения, которые функционально представляют себя единой системой [1]. Есть много причин по которым кластеры являются востребованными, даже для небольших предприятий. К таким причинам относятся: высокая доступность, балансировка нагрузки, параллельной обработки данных, систем управления и масштабируемость. Цель кластеризации можно сформулировать так: создать группу компьютеров в виде единой, полностью интегрированной системы, которая отличается высокой производительностью и отказоустойчивостью.

В общем случае, кластер состоит из двух основных компонентов: кластерное ядро (Cluster Engine) и менеджер ресурсов кластера (Cluster Resource Manager). Не являются исключением и отказоустойчивые кластеры. Кластерное ядро - это "промежуточное" программное обеспечение, которое обеспечивает низкий уровень кластеризации объектов (транспорт, обмен сообщений). Одной из наиболее распространенных реализаций кластерного ядра является Corosync [5].

Менеджер ресурсов кластера - это программное обеспечение, которое находится на следующем уровне стека кластерного программного обеспечения и представляет собой "мозг" кластера. Он предоставляет средства для настройки кластера и управляет кластером в фоновом режиме. Наиболее широко используется менеджер ресурсов Pacemaker [6], совместимый с Corosync.

На третьем уровне стека находятся кластерные ресурсы - пользовательские приложения, конфигурация сети, виртуальные IP-адреса, то есть все, работоспособность чего надо поддерживать. Для взаимодействия менеджера ресурсов кластера с ресурсами используются ресурс-агенты (Resource Agents). Ресурс-агент - это скрипт, который выполняет запуск или остановку ресурса и может возвращать состояние ресурса, т.е. запущен ресурс или остановлен. Кластеризованные ресурсы не должны управляться вручную, т.е. запускаться или останавливаться, а только через интерфейс менеджера ресурсов. Существует несколько стандартов ресурс-агентов, но наиболее широко используемым является OCF (Open Cluster Framework) [2, 3].

Таким образом, стек кластерного программного обеспечения можно представить в виде блок-схемы на рисунке 1.

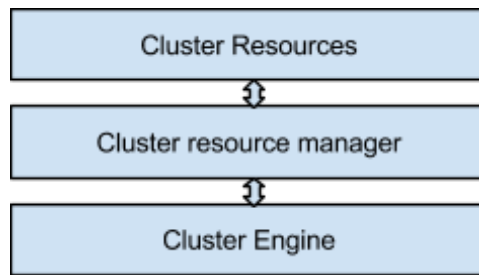


Рис 1. Стек кластерного программного обеспечения

Пример развертывания кластерной системы с помощью Corosync и Pacemaker в среде Fedora Linux приведен в работе [7].

На основе работы [7] было проведено развертывание кластерной системы, содержащей web-сервер Apache2 и виртуальный IP-адрес в среде Ubuntu Linux 14.04 [8]. Блок-схема системы приведена на рисунке 2.

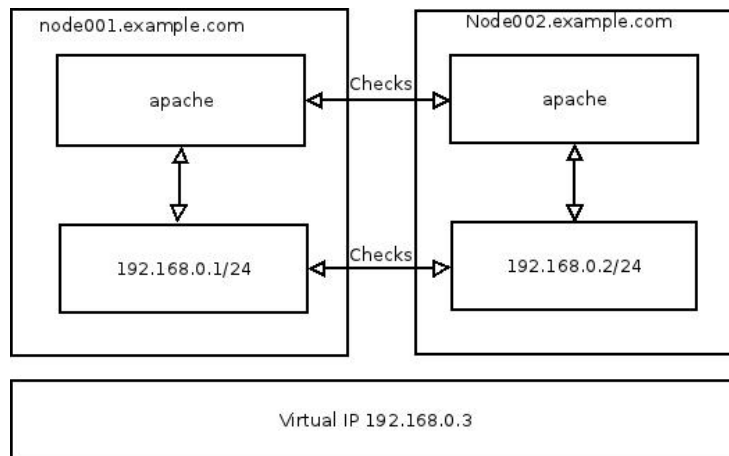


Рис. 2. Блок-схема отказоустойчивого web-сервера

Отказоустойчивость такой системы достигается за счет наличия избыточного вторичного web-сервера, который берет на себя функции первого в случае, если основной web-сервер дал сбой. Прозрачность перехода с первичного сервера на вторичный для клиента достигается за счет того, что при доступе используется единый IP адрес - виртуальный IP (VIP).

Список использованных источников:

1. ClusterLabs. Электронный ресурс. Режим доступа: <http://clusterlabs.org/>
2. Corosync. Электронный ресурс. Режим доступа: <http://corosync.github.io/corosync/>
3. Pacemaker. Электронный ресурс. Режим доступа: <http://clusterlabs.org/pacemaker.html>
4. Andrew Beekhof Pacemaker 1.1 Clusters from Scratch. Creating Active/Passive and Active/Active Clusters on Fedora, Edition 5.
5. Ubuntu 14.04.2 LTS Release. Электронный ресурс. Режим доступа: <http://releases.ubuntu.com/14.04/>

НАУКОВЕ ВИДАННЯ

**ВІЛЬНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ
В ОСВІТІ, НАУЦІ ТА БІЗНЕСІ**

**Тези доповідей шостої
всеукраїнської науково-практичної конференції**

Підписано до друку 10.07.15. Формат 60x84/16. Умов. друк. арк. – 1,34.
Тираж 100 пр. Замовлення № 272/15.

Редакційно-видавничий відділ Чернігівського державного технологічного університету
14027, Україна, м. Чернігів, вул. Шевченка, 95.
Свідоцтво про внесення суб'єкта видавничої справи до Державного реєстру видавців,
виготівників і розповсюджувачів видавничої продукції
серія ДК № 4802 від 01.12.2015 р.