

Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ

**58-я научная конференция
аспирантов, магистрантов и студентов**

Сборник статей

18–22 апреля 2022 года
Минск, БГУИР

УДК 004.021+004.2+004.4+004.6+004.7+004.9
ББК 32.97+32.81

58-я конференция аспирантов, магистрантов и студентов учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», 18-22 апреля 2022 г., БГУИР, Минск, Беларусь: сборник статей. – Мн. – 2022. – 85с.; ил.

В сборнике опубликованы статьи по материалам 58-й научной конференции аспирантов, магистрантов и студентов БГУИР. Материалы одобрены оргкомитетом и публикуются в авторской редакции.

Для научных и инженерно-технических работников, преподавателей, аспирантов, магистрантов и студентов вузов.

СОДЕРЖАНИЕ

СЕКЦИЯ «ВСТРАИВАЕМЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ РЕАЛЬНОГО ВРЕМЕНИ»	6
УДК 621.396.....	7
БЫСТРЫЙ ПОИСК GPS-СИГНАЛА.....	7
FAST GPS-SIGNAL ACQUISITION.....	12
ИССЛЕДОВАНИЕ МЕТОДОВ СЕГМЕНТАЦИИ ГОЛОСОВЫХ СИГНАЛОВ.....	13
ВВЕДЕНИЕ	13
ГРУБАЯ РАЗМЕТА СИГНАЛА НА ПЕРИОДЫ ОСНОВНОГО ТОНА	13
МЕТОД СЕГМЕНТАЦИИ ПУТЕМ ПОДГОНКИ ФОРМЫ СИГНАЛА.....	14
МЕТОД ОТБОРА ЛОКАЛЬНЫХ МАКСИМУМОВ	16
ОПИСАНИЕ ЭКСПЕРИМЕНТА.....	17
VOICE SIGNAL SEGMENTATION METHODS.....	19
ПРОГРАММНЫЙ МОДУЛЬ РАСПОЗНАВАНИЯ ОБРАЗОВ ДЛЯ МИКРОКОМПЬЮТЕРОВ НА ОСНОВЕ НЕЙРОННЫХ СЕТЕЙ СЕМЕЙСТВА YOLOV5.....	37
SOFTWARE MODULE FOR PATTERN RECOGNITION FOR MICROCOMPUTERS BASED ON NEURAL NETWORKS OF YOLOV5 FAMILY	40
СЕКЦИЯ «ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ»	41
АЛГОРИТМ ПОИСКА ЗАИМСТВОВАНИЙ В МУЗЫКАЛЬНЫХ ПРОИЗВЕДЕНИЯХ	42
Подход для ОБРАБОТКИ и СРАВНЕНИЯ ПОЛИГАРМОНИЧЕСКИХ АКУСТИЧЕСКИХ СИГНАЛОВ	44
Визуализация базовых операций цифровой обработки сигналов с использованием технологий PYTHON.....	49
УДК 519.711.3:629.5	52
Применение Математической модели PID-регулятора для симуляции движения судомодели.....	52
UDC 519.711.3:629.5	56

APPLICATION OF A MATHEMATICAL MODEL OF THE PID CONTROLLER TO SIMULATE THE MOTION OF A SHIP MODEL.....	56
ПРОГРАММНОЕ СРЕДСТВО ДЛЯ АНАЛИЗА МАРШЕВЫХ ТЕСТОВ.....	59
ПРОГРАММНО-АППАРАТНОЕ СРЕДСТВО ВИЗУАЛИЗАЦИИ РАБОТЫ АКСЕЛЕРОМЕТРА И ГИРОСКОПА.....	62
АЛГОРИТМ ФОРМИРОВАНИЯ КОЛЕБАНИЙ СТОЛА ВИБРОСТЕНДА.....	65
СЕКЦИЯ «ВЫСШАЯ МАТЕМАТИКА».....	68
УДК.....	69
ШИФР ЦЕЗАРЯ И ГЕНЕТИЧЕСКИЙ АЛГОРИТМ.....	69
UDK.....	77
МНОГОКРИТЕРИАЛЬНАЯ ОПТИМИЗАЦИЯ ПО ПАРЕТО.....	78
MULTI-CRITERIA PARETO OPTIMIZATION.....	83
МОДЕЛИРОВАНИЕ РАСПРОСТРАНЕНИЯ РЕКЛАМЫ.....	95
ДИНАМИЧЕСКОЕ МОДЕЛИРОВАНИЕ КОНКУРЕНЦИИ ПРОМЫШЛЕННОГО ПРОИЗВОДСТВА РОССИИ И БЕЛАРУСИ НА ОСНОВЕ МОДЕЛИ ЛОТКИ-ВОЛЬТЕРРЫ.....	97
АНАЛИЗ АЛГОРИТМОВ КЛАССИФИКАЦИИ ОБЪЕКТОВ НА ДАННЫХ ДИСТАНЦИОННОГО ЗОНДИРОВАНИЯ ЗЕМЛИ.....	132
ПРИМЕНЕНИЕ МИКРОКОНТРОЛЛЕРОВ В СИСТЕМАХ УПРАВЛЕНИЯ И МОНИТОРИНГА.....	142
СЕКЦИЯ «МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ ТЕХНИЧЕСКИХ СИСТЕМ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ»	146
СОЗДАНИЕ ПРИВОДОВ КОЛЁСНО-ШАГАЮЩИХ ДВИЖИТЕЛЕЙ МОБИЛЬНЫХ МАШИН С АДАПТИВНЫМ УПРАВЛЕНИЕМ.....	147
ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ МЕХАТРОННОЙ СИСТЕМЫ ПАРАЛЛЕЛЬНОЙ КИНЕМАТИКИ НА ГИБРИДНОМ ПРИВОДЕ ПРЯМОГО ДЕЙСТВИЯ ..	149
ВЕРСИОНИРОВАНИЕ И КОНТРОЛЬ ЦЕЛОСТНОСТИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ СИСТЕМ РЕАЛЬНОГО ВРЕМЕНИ.....	152
СЕКЦИЯ «ФИЛОСОФИЯ, ЛОГИКА И ЖИЗНЬ».....	155
СОЦИАЛЬНО-ЭКОЛОГИЧЕСКАЯ КАТАСТРОФА ЧЕРНОБЫЛЯ: РЕТРОСПЕКТИВА.....	Ошибка! Закладка не определена.
ШАХМАТЫ КАК ЛОГИКО-ФИЛОСОФСКИЙ ФЕНОМЕН	Ошибка! Закладка не определена.
ЭВОЛЮЦИЯ ОТНОШЕНИЯ ЧЕЛОВЕКА К СМЕРТИ В ЕВРОПЕЙСКОЙ ФИЛОСОФИИ.....	Ошибка! Закладка не определена.

ТЕОРЕТИЧЕСКИЕ И ПРАКТИЧЕСКИЕ АСПЕКТЫ ЭМОЦИОНАЛЬНОГО ИНТЕЛЛЕКТА	ОШИБКА! Закладка не определена.
ГЕНЕЗИС КОНЦЕПЦИИ ПОСТГУМАНИЗМА	ОШИБКА! Закладка не определена.
ПРОБЛЕМА СОЦИАЛЬНОЙ НЕСПРАВЕДЛИВОСТИ В СОЦИАЛЬНОЙ ФИЛОСОФИИ ДЖ.А. КОЭНА	ОШИБКА! Закладка не определена.
СЕКЦИЯ «ФИЗИКА»	172
ЭФФЕКТИВНОСТЬ ИЗУЧЕНИЯ ФИЗИКИ В УСЛОВИЯХ ПРОЕКТА «ЭВРИСТИКА В ФИЗИКЕ» НА ФАКУЛЬТЕТЕ КСИС В БГУИР	ОШИБКА! Закладка не определена.
ЛУЧШАЯ РАБОТА 6-ГО СЕЗОНА «ЭВРИСТИКА В ФИЗИКЕ»: САМОДЕЛЬНАЯ ЭЛЕКТРОГИТАРА	ОШИБКА! Закладка не определена.
СЕКЦИЯ «ИНФОРМАТИКА»	176
ПОСТРОЕНИЕ И ОЦЕНКА ЭФФЕКТИВНОСТИ НЕЙРОСЕТЕВЫХ МОДЕЛЕЙ КЛОНИРОВАНИЯ ГОЛОСА	ОШИБКА! Закладка не определена.
АНАЛИЗ ХАРАКТЕРИСТИК ФИЗИЧЕСКИ НЕКЛОНИРУЕМОЙ ФУНКЦИИ ТИПА АРБИТР НА FPGA ARTIX-7	ОШИБКА! Закладка не определена.
ИСПОЛЬЗОВАНИЕ ГЕНЕРАТОРОВ КОНСТАНТ ДЛЯ ВНЕДРЕНИЯ ВОДЯНЫХ ЗНАКОВ В ПРОЕКТНЫЕ ОПИСАНИЯ	ОШИБКА! Закладка не определена.

**СЕКЦИЯ «ВСТРАИВАЕМЫЕ ВЫЧИСЛИТЕЛЬНЫЕ
СИСТЕМЫ РЕАЛЬНОГО ВРЕМЕНИ»**

УДК 621.396

БЫСТРЫЙ ПОИСК GPS-СИГНАЛА

Гомолко В. А., студент гр.950702

Белорусский государственный университет информатики и радиоэлектроники¹
г. Минск, Республика Беларусь

Вашкевич М.И. – канд. тех. наук,
доцент

Аннотация. В работе рассмотрен модуль быстрого поиска GPS-сигнала с применением когерентного интегрирования в качестве способа снижения ложного срабатывания системы быстрого поиска спутникового сигнала. Представлено описание корреляционных свойств сигнала, структура спутникового сигнала, принцип генерации псевдослучайной последовательности. Приведены графики для начальных значений сигнал/шум с использованием когерентного интегрирования.

Ключевые слова. Обнаружение, двоичная фазовая манипуляция, глобальная навигационная спутниковая система, псевдослучайный шум, коды Голда, когерентное интегрирование.

Введение.

Спутниковая навигация предназначена для определения местоположения объектов, скорости, времени, а также направление движения приемника. При этом современная глобальная навигационная спутниковая система (далее – ГНСС) позволяет определять местоположение в реальном времени.

На 2022 год три спутниковые системы обеспечивают полное покрытие и бесперебойную работу для всего земного шара – GPS (США), ГЛОНАСС (РФ), BeiDou (Китай). ГНСС не требуют от пользователя передачи каких бы то ни было данных.

Статья рассматривает повышения точности срабатывания модуля быстрого поиска спутникового сигнала посредством увеличения продолжительности когерентного интегрирования. При этом статья описывает принцип генерации спутникового сигнала, его корреляционные свойства и принцип работы блока быстрого поиска спутникового сигнала.

Типы сигналов GPS.

Грубый/Приобретенный код (англ. *Course/Acquisition code*) – каждый спутник использует свой уникальный код для реализации технологии множественного доступа с кодовым разделением (англ. *code division multiple access*). Коды C/A принадлежат к семейству кодов золотого псевдослучайного шума (англ. *Pseudorandom Noise gold code*). Алгоритм создания золотого псевдослучайного шума (далее – ПСШ) был разработан в 1967 году и используется до сих пор. Длительность кода составляет 1 мс.

Сигнал $P(Y)$ – зашифрованный сигнал, созданный для вооруженных сил США. Передается на частотах $L1$ (1575,42 МГц) и $L2$ (1227,60 МГц). Длительность кода составляет 7 дней.

Принцип генерации сигнала.

Спутниковый GPS-сигнал формируется по следующему принципу: сигнал с несущей модулируется суммой C/A-кода по модулю 2 вместе с навигационными данными. В зависимости от значения кодовой последовательности изменяется фаза сигнала: передается без изменений по фазе, либо изменяется на 180° – двоичная фазовая манипуляция (англ. *binary phase-shift keying*). Математическая модель описывается следующим образом:

$$s(t) = A_c \cdot C(t) \cdot D(t) \cdot \cos(2\pi \cdot (f_{L1} + f_D) \cdot t) \quad (1)$$

где,

A_c : амплитуда C/A кода

$C(t)$: C/A код

$D(t)$: навигационные данные

f_{L1} : частота несущей $L1$

f_d : Доплеровская частота

Генератор C/A кода представляет собой два полинома G_1 и G_2 , работающих на частоте 1,023 МГц. Упрощенное описание реализации полиномов требует два 10-битный регистра и логические элементы XOR. В процессе реализации полинома G_1 берется первый и третий регистр,

Из рисунков видно, что пиковое значение имеет место при нулевом временном сдвиге и наблюдается низкая кросс-корреляция двух спутников.

Поиск сигнала.

Для проверки на присутствие полезного сигнала во входных данных используется процесс обнаружения (англ. *acquisition*). Поиск происходит в двумерном пространстве: фазе кода и частоте в диапазоне ± 10 кГц для покрытия всего ожидаемого доплеровского диапазона [2]. Каждая ячейка пространства поиска показывает значение корреляции по Доплеру и кодовой задержке.

В данной работе используется метод, основанный на круговой свертке в частотной области при помощи БПФ. Метод позволяет сократить количество комбинаций: все возможные значения фазовой задержки определяются при выполнении круговой свертки и остается перебрать комбинации, соответствующие смещению по Доплеру, что с учетом используемого шага в 250 Гц составит

81 вариант, благодаря чему получается значительно быстрее получить результат и именно поэтому поиск называется быстрым. Операция круговой свертки описывается выражением:

$$s_1(m) * s_2(m) \xleftrightarrow{\text{БПФ}} S_1(k) \cdot S_2(k) \quad (6)$$

Полное математическое описание процесса поиска между полученным сигналом $s(n)$ и локально-сгенерированным $s_{\text{ЛС}}(n)$ представлено далее [3]:

$$y(n) = \sum_{m=0}^{N-1} s(m) \cdot s_{\text{ЛС}}(m-n) \quad (7)$$

$$Y(k) = \sum_{n=0}^{N-1} y(n) \cdot W_N^{n \cdot k} = \sum_{n=0}^{N-1} \sum_{m=0}^{N-1} s(m) \cdot s_{\text{ЛС}}(m-n) \cdot W_N^{n \cdot k} \quad (8)$$

где, $W_N = e^{\frac{-j \cdot 2 \cdot \pi}{N}}$

$$Y(k) = \sum_{m=0}^{N-1} s(n) \cdot W_N^{n \cdot k} \sum_{n=0}^{N-1} s_{\text{ЛС}}(m-n) W_N^{-(m-n) \cdot k} = S(k) \cdot S_{\text{ЛС}}^*(k) \quad (9)$$

где, $S_{\text{ЛС}}^*(k)$ – комплексное сопряжение сигнала $S_{\text{ЛС}}(k)$.

$$y(n) = S(k) \cdot S_{\text{ЛС}}^*(k) \quad (10)$$

$$P = |y(n)|^2 \quad (11)$$

После завершения процесса обнаружения необходимо выяснить, присутствует ли спутник. Определяется присутствие при превышении порогового значения. Если P превышает порог, то далее информация о наличии полезного сигнала передается в блок слежения, параллельно которому происходит декодирование навигационных данных и определение местоположения приемника.

Когерентное интегрирование.

Стандартные значения сигнал/шум в приемнике от -26 дБ до -18 дБ при ширине полосы пропускания в 2 МГц [4]. Предположим, синтезирован сигнал с доплеровским смещением – 2,1 кГц, смещением по коду – 500 символов, отношением сигнал/шум – -26 дБ. В следствие работы блока поиска с заданными параметрами получен следующий результат:

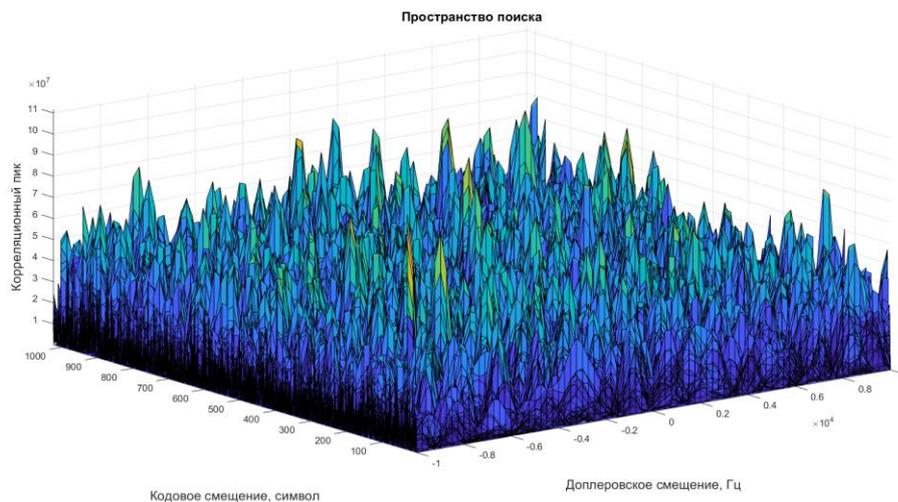


Рисунок 3 – Пространство поиска спутникового сигнала № 1 при отношении сигнал/шум -26 дБ без применения когерентного интегрирования

При данных значениях сигнал/шум система с высокой вероятностью выполнит ложное срабатывание (англ. *false alarm*), поскольку корреляция сигнала с подобным уровнем амплитуды шума превысит стандартный порог, поэтому необходимо улучшить отношение сигнал/шум.

Одним из решений данной проблемы является использование когерентного интегрирования (англ. *coherent integration*): мы аккумулируем множество последовательностей сигнала с шумом, причем начальная фаза сигнала во всех последовательностях должна совпадать; в результате интегрирования отношение сигнал/шум улучшается [5]. Таким образом увеличение продолжительности интегрирования приведет к улучшению упомянутого ранее отношения. Недостатком данного метода является необходимость совпадения начальной фазы. На один бит информации приходится 20 периодов псевдослучайного шума, что эквивалентно 20 мс, интегрировать больший период не имеет смысла ввиду разницы фаз, что приведет к взаимному компенсированию и отсутствию улучшения. Рассмотрим процесс для нескольких стартовых значений сигнал/шум: -26 дБ и -18 дБ (рисунок 6).

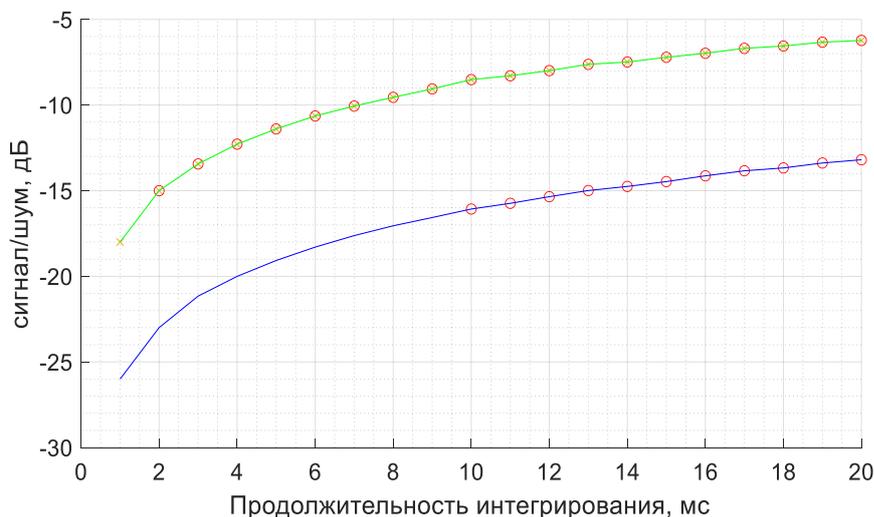


Рисунок 4 – График зависимости продолжительности когерентного интегрирования и уровня сигнал/шум для значений -26 дБ и -18 дБ

Как видно из графика со временем увеличения продолжительности интегрирования процесс улучшения замедляется. С улучшением отношения сигнал/шум возрастает точность обнаружения спутника №1 и его параметров, точные срабатывания системы обозначены кругом. В результате тестирования минимальное отношение сигнал/шум, при котором спутник определяется корректно является -15 дБ; далее вместе с ним рассматривается и -12 дБ:

a)

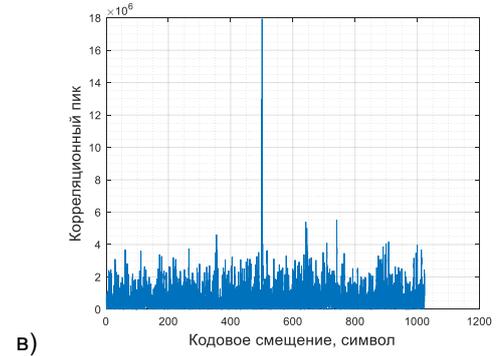
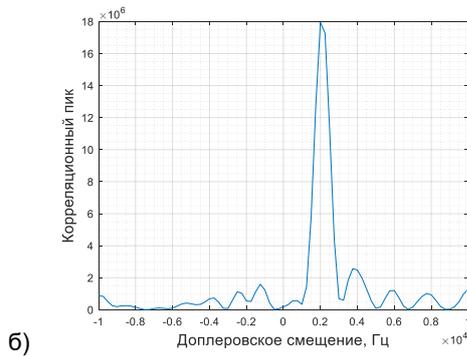
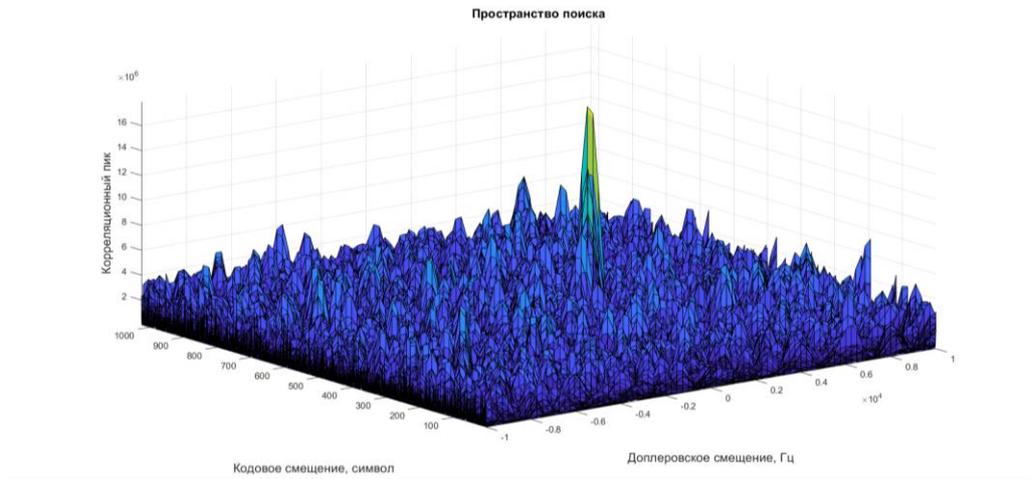


Рисунок 5 – а) Пространство поиска спутникового сигнала № 1 при отношении сигнал/шум -18 дБ с периодом когерентного интегрирования 2 мс б) Доплеровское смещение в) Кодовое смещение

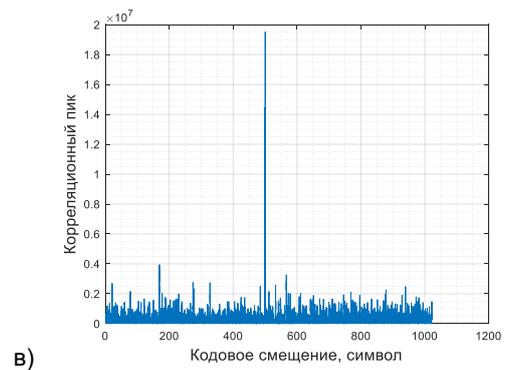
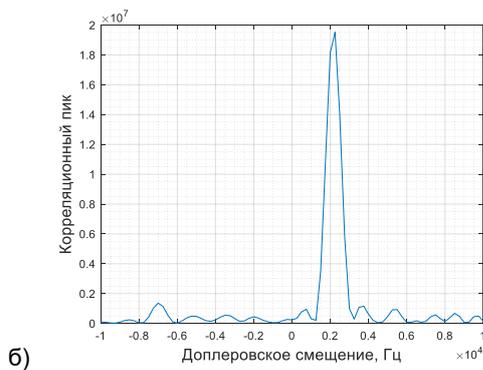
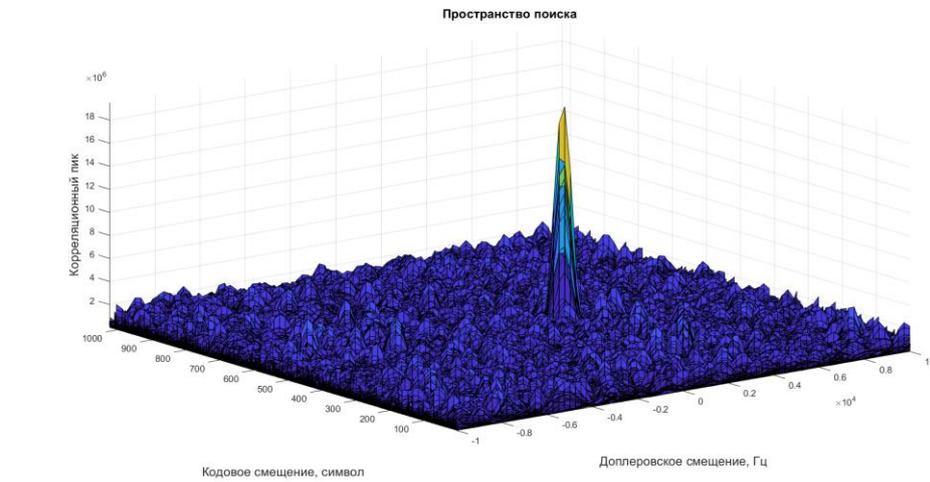


Рисунок 6 – а) Пространство поиска спутникового сигнала № 1 при отношении сигнал/шум -18 дБ с периодом когерентного интегрирования 4 мс. б) Доплеровское смещение. в) Кодовое смещение.

Как видно из рисунков алгоритм получения спутникового сигнала GPS оценил примерные значения параметров: доплеровское и кодовое смещения. Более точные значения определяются в процессе слежения (англ. *tracking*). Основная задача слежения заключается в удержании синхронности сигнала с целью извлечения навигационных данных.

Вывод.

В работе рассмотрен способ снижения ложного срабатывания системы быстрого поиска спутникового сигнала путем улучшения отношения сигнал/шум. Результат показывает то, что отношение сигнал/шум возможно улучшить, и это повышает точность срабатывания системы получения сигнала. Важность рассмотренного метода заключается в первую очередь в скорости работы алгоритма и уменьшении ложных срабатываний. Моделирование проводилось в MATLAB.

Список использованных источников:

1. Doberstein D. *Fundamentals of GPS receivers : a hardware approach* – New York : Springer, ©2012. С. 245.
2. Tsui J. B. Y. *Fundamentals of global positioning system receivers: a software approach*. – John Wiley & Sons, 2005.
3. Hussain, A.; Ahmed, A.; Magsi, H.; Soomro, J.B.; Bukhari, S.S.H.; Ro, J.-S. Adaptive Data Length Method for GPS Signal Acquisition in Weak to Strong Fading Conditions. *Electronics* 2021, 10, 1735. <https://doi.org/10.3390/electronics10141735>.
4. Joseph, A. and Petovello, M. (2010). *Measuring GNSS Signal Strength. Inside GNSS*. November.
5. Лайонс Р. *Цифровая обработка сигналов. Второе издание. Пер. с англ./ Р. Лайонс // М. : ООО «Бином-Пресс», 2006. С. 412.*

DC 621.396

FAST GPS-SIGNAL ACQUISITION

Gomolko V. A.

Belarusian State University of Informatics and Radioelectronics¹, Minsk, Republic of Belarus

Vashkevich M.I. – PhD

Annotation. This paper considers a way to reduce false alarm for the GPS-signal acquisition. In article the description of the correlation properties of the signal, the structure of the satellite signal, the principle of generating a pseudorandom sequence is considered. Figures are given for the initial signal/noise values on the receiver using coherent integration.

Keywords. Acquisition, binary phase shift keying, global navigation satellite system, pseudorandom Noise, golden codes, coherent integration.

УДК 616.71

ИССЛЕДОВАНИЕ МЕТОДОВ СЕГМЕНТАЦИИ ГОЛОСОВЫХ СИГНАЛОВ

Пастернак В. В., Стецко В.Ю., студенты гр.950702

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Вашкевич М.И. – канд. техн. наук

Аннотация. В работе рассмотрены методы сегментации голосовых сигналов, а именно Waveform Matching метод и Peak-picking метод. Результатом работы этих методов является извлечение частоты основного тона голосового сигнала. Методы проверялись на тестовых сигналах, имеющих синусоидальную форму и различную частотную модуляцию. Для проверки производительности методы сравнивались по отношению параметров частотной пертурбации к их теоретическим значениям.

Ключевые слова. Сегментация голосовых сигналов, частота основного тона.

Введение

Меры искажения голоса, такие как джиттер и шиммер, зависят от точного извлечения частоты основного тона из аудиосигнала. Метод сегментации, а точнее, его способность точно определить циклические границы напрямую влияет на точность этих измерений, особенно если в сигнале присутствуют шум и модуляция. В связи с этим весьма актуальна задача повышения точности работы методов сегментации.

- В данной работе исследуются два метода сегментации голосовых сигналов [1]:
- метод отбора локальных максимумов (англ. *PP – peak-peaking method*);
 - метод подгонки формы сигнала (англ. *WM – waveform matching method*).

Грубая разметка сигнала на периоды основного тона

Входной информацией для обоих методов является грубая разметка сигнала на периоды основного тона. Грубая разметка задается последовательностью $I_N(i)$, $i = 1, 2, \dots, N_c + 1$, где N_c – число периодов основного тона в исходном сигнале. Значение $I_N(i)$ определяет границу между $(i - 1)$ -м и i -м периодом основного тона. Для получения разметки $I_N(i)$ анализируемый сигнал $x(n)$ пропускается через КИХ-фильтр нижних частот с частотой среза:

$$F_c = 1,5 \cdot F_0 \quad (1)$$

где F_0 – грубая оценка частоты основного тона.

Фильтрация необходима, чтобы удалить из сигнала все гармонические компоненты за исключением основной гармоники. Для определения F_0 из входного сигнала выбирался фрагмент голосового сигнала длительностью 50 мс (рисунок 1), от которого затем вычислялась автокорреляционная функция (АКФ) (рисунок 2). Положение первого пика АКФ для $\tau > 0$ определяет задержку соответствующую периоду основного тона T_0 . Далее частота основного тона определялась как $F_0 = 1/T_0$.

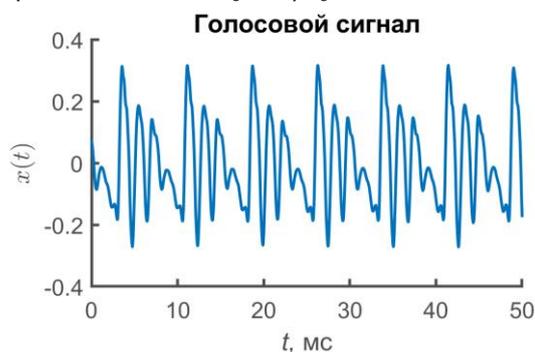


Рисунок 1 – Пример голосового сигнала

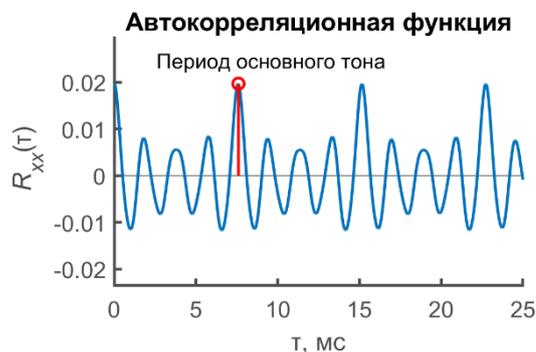


Рисунок 2 – Автокорреляционная функция голосового сигнала

После фильтрации определяются точки пересечения нуля из положительного значения сигнала в отрицательное. Убрав задержку, вызванную фильтром, эти точки можно перенести на исходный сигнал и использовать их в качестве грубой разметки $I_N(i)$. На рисунке 3 показан фильтрованный сигнал с полученной грубой разметкой $I_N(i)$. А на рисунке 4 показан исходный сигнал с нанесенной грубой разметкой. Можно видеть, что полученная указанным образом грубая разметка позволяет эффективно определить приблизительные границы периодов основного тона.

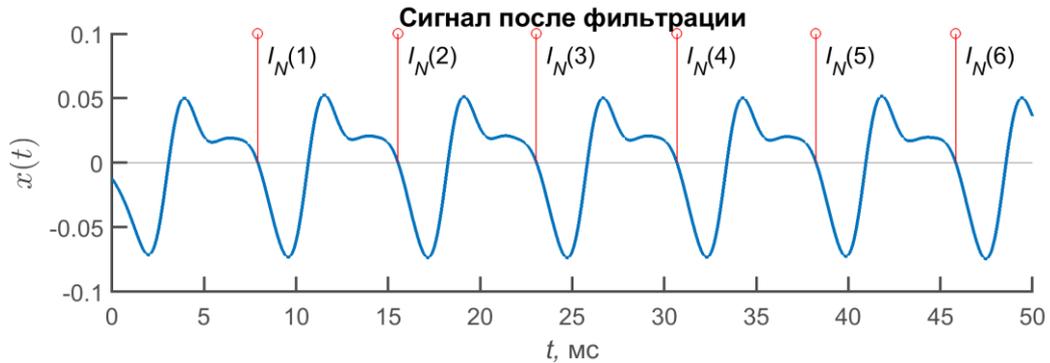


Рисунок 3 – Фильтрованный с грубой разметкой

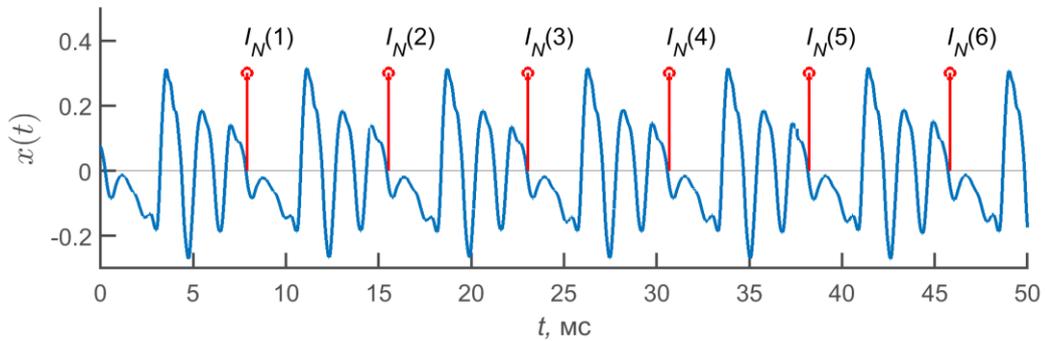


Рисунок 4 – Исходный сигнал с грубой разметкой

Метод сегментации путем подгонки формы сигнала

Суть метода *подгонки формы сигнала* состоит в поиске минимумов между точками грубой разметки так, чтобы среднеквадратичная ошибка между двумя соседними формами сигнала была минимальной, чтобы далее найти частоту основного тона, используя формулу:

$$F(i-1) = \frac{f_s}{P(i) - P(i-1) + \delta'} \quad (2)$$

где f_s – частота дискретизации сигнала.

Полная процедура метода заключается в следующем:

1. Находим местоположение абсолютного минимума – точку $P(1)$, которая находится между первыми двумя точками грубой разметки – $I_N(1)$ и $I_N(2)$.
2. Далее создаем цикл для $i = 2$ до $N_C + 1$ и проделываем операции внутри цикла.
 - А) Находим первоначальное предположение:

$$P(i) = I_N(i) + (P(i-1) - I_N(i-1)) \quad (3)$$

Пример найденных точек $P(i)$ согласно выражению (3) показан на рисунке 5.

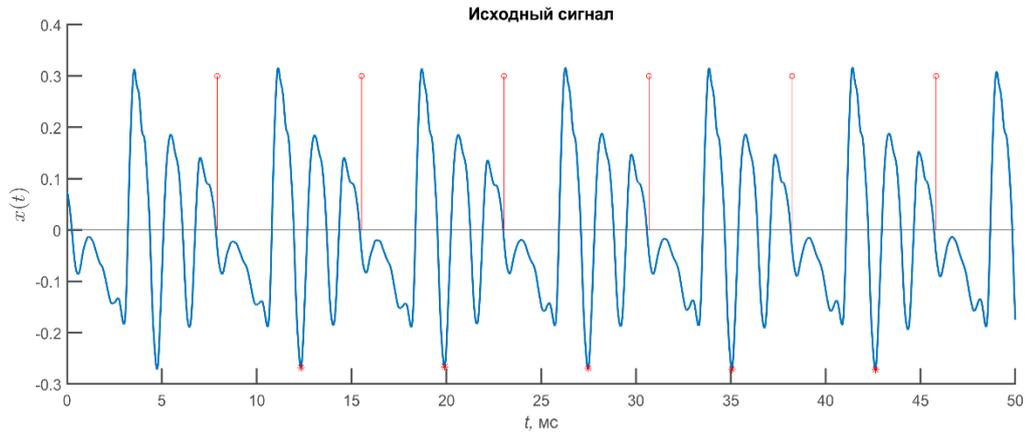


Рисунок 5 – Сигнал с минимумами между точками грубой разметки

Б) Устанавливаем верхние и нижние пределы видимости:

$$J_1 = P(i) - PERC \cdot (P(i) - P(i-1)) \quad (4)$$

$$J_2 = P(i) + PERC \cdot (P(i) - P(i-1)) \quad (5)$$

где $PERC$ – заданный параметр. Например, 0.05 – 0.15 (т.е. допускается изменение на 5-15%).

В) Находим точку J_m между пределами видимости J_1 и J_2 , чтобы среднеквадратичная ошибка $ERR(J_m)$ была минимальной, где

$$ERR(j) = \frac{1}{j - P(i-1)} \cdot \sum_{k=P(i-1)}^{j-1} (x(k + (j - P(i-1)))) - x(k))^2 \quad (6)$$

где $x(n)$ – анализируемый сигнал.

Г) Если $J_m = J_1$ или J_2 , расширяем поиск за пределы J_1 или J_2 до тех пор, пока не будет найден минимум функции $ERR(j)$.

Д) Пусть $P(i) = J_m$. Следует обратить внимание, что $P(i)$ – целое число, следовательно, разница $P(i) - P(i-1)$ дает оценку периода основного тона с точностью, ограниченной частотой дискретизации.

Е) Чтобы улучшить оценку частоты, найдем полином второй степени, проходящий через точки $ERR(J_m - 1)$, $ERR(J_m)$ и $ERR(J_m + 1)$, чтобы найти положение точки минимума.

Расстояние между точкой минимума и J_m равно:

$$\delta = -0.5 \cdot \frac{ERR(J_m + 1) - ERR(J_m - 1)}{ERR(J_m + 1) - 2 \cdot ERR(J_m) + ERR(J_m - 1)} \quad (7)$$

Ж) Частота $(i-1)$ -ого цикла равна:

$$F(i-1) = \frac{f_s}{P(i) - P(i-1) + \delta} \quad (8)$$

Метод подгонки формы сигнала реализован в *Matlab* следующим образом:

<pre>function [F,F_time] = WM_method(x, fs, l_N) % x – входной сигнал % fs – частота дискретизации % l_N – грубая разметка входного сигнала PERC = 0.05; Nc = length(l_N); F = zeros(1,Nc-3); [~,P(1)] = min(x(l_N(1):l_N(2))); P(1) = P(1) + l_N(1); for i = 2:Nc-2 P(i) = l_N(i) + (P(i-1) - l_N(i-1)); J1 = P(i) - round(PERC * (P(i) - P(i-1))); J2 = P(i) + round(PERC * (P(i) - P(i-1)));</pre>	<pre>function [err_cur] = ERR(j, P, i, buff) sum = 0; for k=P(i-1):(j-1) sum = sum + (buff(k+(j-P(i-1)))) - buff(k))^2; end err_cur = (1/(j-P(i-1))) * sum; end ----- function [err_Jm_plus1, err_Jm_minus1, err_Jm] = ERR_Jm(Jm, P, i, buff) sum = 0; for k=P(i-1):(Jm+1-1) sum = sum + (buff(k+(Jm+1-P(i-1)))) - buff(k))^2;</pre>
--	--

<pre> search_complete = false; while (~search_complete) err_min = inf; Jm = []; for j = J1:J2 err_cur = ERR(j, P, i, x); if (err_cur < err_min) err_min = err_cur; Jm = j; end end if (Jm == J1) J1 = J1 - round(0.5*(J2-J1)); elseif (Jm == J2) J2 = J2 + round(0.5*(J2-J1)); else search_complete = true; end end P(i) = Jm; [err_Jm_plus1, err_Jm_minus1, err_Jm] = ERR_Jm(Jm, P, i, x); delta = -0.5 * ((err_Jm_plus1 - err_Jm_minus1)/(err_Jm_plus1 - 2*err_Jm + err_Jm_minus1)); F(i-1) = fs / (P(i) - P(i-1) + delta); end </pre>	<pre> end err_Jm_plus1 = (1/(Jm+1-P(i-1))) * sum; sum = 0; for k=P(i-1):(Jm-1-1) sum = sum + (buff(k+(Jm-1-P(i-1))) - buff(k))^2; end err_Jm_minus1 = (1/(Jm-1-P(i-1))) * sum; sum = 0; for k=P(i-1):(Jm-1) sum = sum + (buff(k+(Jm-P(i-1))) - buff(k))^2; end err_Jm = (1/(Jm-P(i-1))) * sum; end </pre>
---	--

Метод отбора локальных максимумов

Метод *отбора локальных максимумов* использует нахождение отрицательных пиков на каждом фрагменте сигнала между точками грубой разметки. К точке отрицательного пика применяется интерполяция второго порядка. После проделанных действий высчитывается частота основного тона на каждом фрагменте:

$$F(i-1) = \frac{Fs}{P(i) - P(i-1)} \quad (9)$$

где Fs – частота дискретизации, $P(i)$ – местоположение отрицательного пика на каждом фрагменте.

На рисунке 6 показан исходный сигнал с найденными отрицательными пиками между точками грубой разметки.

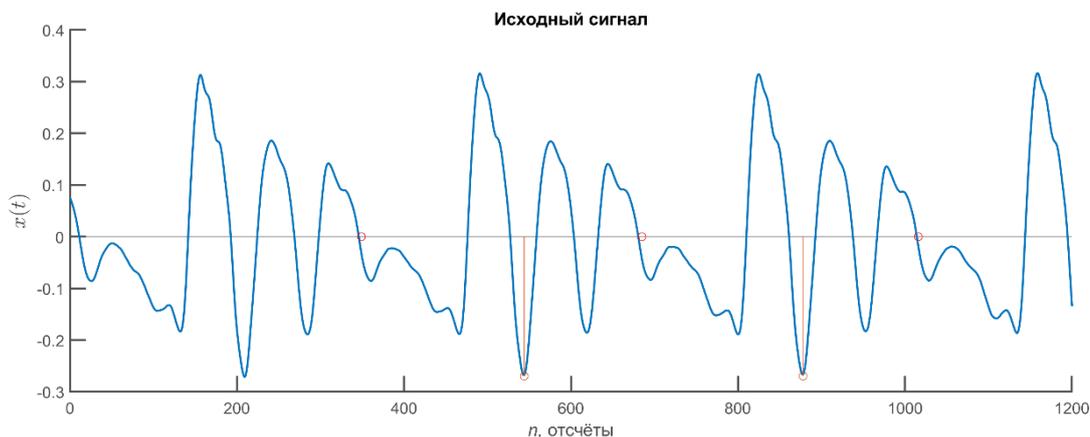


Рисунок 6 – Временное представление сигнала с отрицательными пиками и грубой разметкой

Найденный массив частоты основного тона проверяется по заданному процентному отклонению для нахождения неверной частоты, если она имеется. Данные заносятся в массив логический значений.

Метод отбора локальных максимумов реализован в *Matlab* следующим образом:

```

function [F,ERR,F_time] = PPM(buff, fs, I_N)
% buff – входной сигнал
% fs – частота дискретизации
% I_N – грубая разметка входного сигнала

```

```

PERC = 0.05;
Nc = length(L_N);

for i = 1:Nc - 1
    J1 = L_N(i);
    J2 = L_N(i + 1);
    [~, Pi] = min(buff(J1:J2));
    Pi = Pi + J1 - 1;
    Pr(i) = Pi + (-0.5 * (buff(Pi + 1) - buff(Pi - 1)))/(buff(Pi + 1) - 2 * buff(Pi) + buff(Pi - 1));

    if (i > 1)
        F(i - 1) = fs / (Pr(i) - Pr(i - 1));
    end
end

for i = 1:Nc - 3
    ERR(i) = F(i + 1) - F(i) >= F(i) * PERC;
end
end

```

На рисунке 7 представлены результаты поиска частоты основного тона по методам *отбора локальных максимумов* и *подгонки формы сигнала* соответственно.

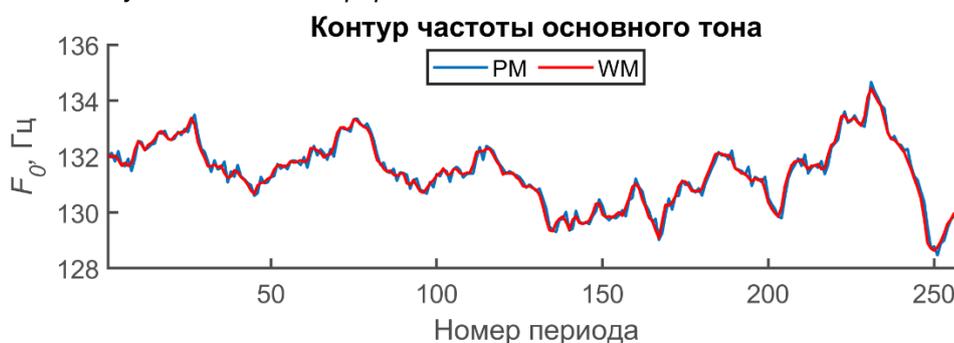


Рисунок 7 – Частоты основного тона по каждому из методов

Описание эксперимента

Для проведения экспериментов использовались полигармонические сигналы с добавлением амплитудной и частотной модуляции. Для получения полигармонического колебания использовался «смещенный» синус:

$$S_b(\alpha) = \begin{cases} 0,5 \left(1 - \cos\left(\frac{1+r}{2}\alpha\right) \right), & 0 \leq \alpha < \frac{2\pi}{1+r}, \\ 0,5 \left(1 - \cos\left(\frac{1+r}{2r}\alpha + \frac{r-1}{r}\pi\right) \right), & \frac{2\pi}{1+r} \leq \alpha < 2\pi. \end{cases} \quad (10)$$

Степень «смещенности» синуса определяется параметром r , при $r = 1$ получается чистый синус. Входной аргумент α – фазовое значение находящееся в диапазоне от 0 до 2π . Обратите внимание, что если в качестве аргумента в $S_b(\alpha)$ попадет $\alpha > 2\pi$, то необходимо вычесть из него 2π и только затем произвести вычисление по формуле (10). В случае, если после первого вычитания окажется, что α все ещё больше 2π вычитание следует повторить. И так до тех пор, пока α не попадет в диапазон $0 \leq \alpha < 2\pi$.

Параметрами частотно-модулированного (ЧМ) сигнала являются: f_0 – частота основного тона (Гц); f_m – частота модуляции (Гц); k_f – индекс модуляции. Для генерирования ЧМ-сигнала использовалось выражение:

$$s(n) = S_b(\alpha(n)), \quad (11)$$

где

$$\alpha(n) = 2\pi \sum_{m=0}^n \frac{f_0}{f_s} \left(1 + k_f \cos\left(\frac{2\pi f_m}{f_s} m\right) \right). \quad (12)$$

Параметрами амплитудно-модулированного (АМ) сигнала являются: f_s – частота дискретизации (Гц); f_{am} – частота модуляции (Гц); k_a – индекс модуляции. Для генерирования АМ-сигнала использовалось выражение:

$$s_{AM}(n) = s(n) * \left(1 + k_a \cos\left(\frac{2\pi f_{am} n}{f_s}\right) \right). \quad (13)$$

На рисунке 8 приведен сегмент тестового сигнала длиной 100 миллисекунд после амплитудной модуляции.

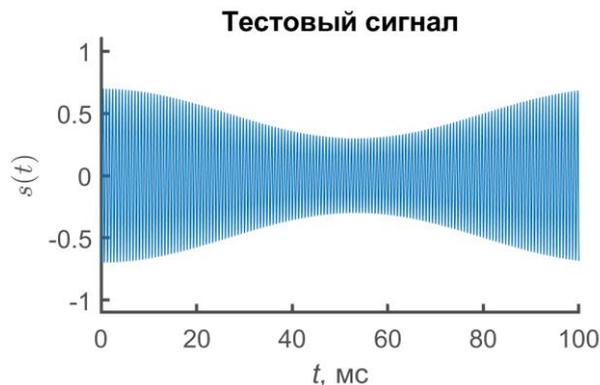


Рисунок 8 – Сегмент тестового сигнала после амплитудной модуляции

Для сравнения производительности между методами сегментации были созданы 20 тестовых сигналов, отличающиеся величиной частотной ($f_0 = 150$ Гц). На рисунке 9 приведен пример контура частоты основного тона (ЧОТ) тестового сигнала, а на рисунке 10 десятимиллисекундный сегмент тестового сигнала.

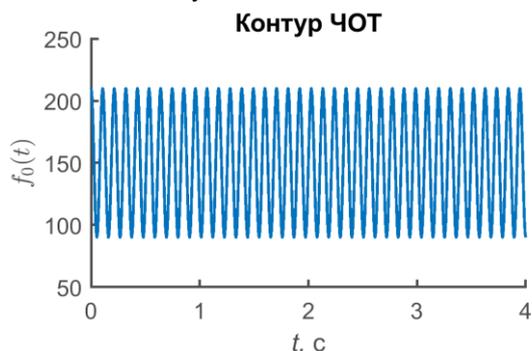


Рисунок 9 – Контур частоты основного тона тестового сигнала

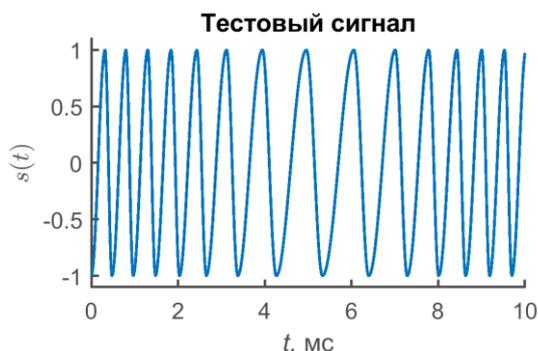


Рисунок 10 – Сегмент тестового сигнала

К каждому тестовому сигналу также была добавлена амплитудная модуляция. На основе оцененной приведенными методами частот основного тона для каждого сигнала высчитывается параметр $PF1$ как мера частотного отклонения в сигнале:

$$PF1 = \frac{1}{N-1} \sum_{i=1}^{N-1} \frac{|f(i+1) - f(i)|}{0.5 * [f(i+1) + f(i)]} \times 100 \quad (14)$$

где N – количество частот основного тона, $f(i)$ – частота основного тона в i -ом фрагменте сигнала.

Параметр $PF1$ используется для определения среднего отклонения первого порядка.

Поскольку все тестовые сигналы моделировались компьютером, теоретические значения частоты основного тона и теоретические $PF1$ параметры были известны.

Сравнение полученных параметров пертурбации $PF1$ с теоретическими значениями $PF1$ изображено на рисунке 11 и с увеличенным масштабом на рисунке 12.

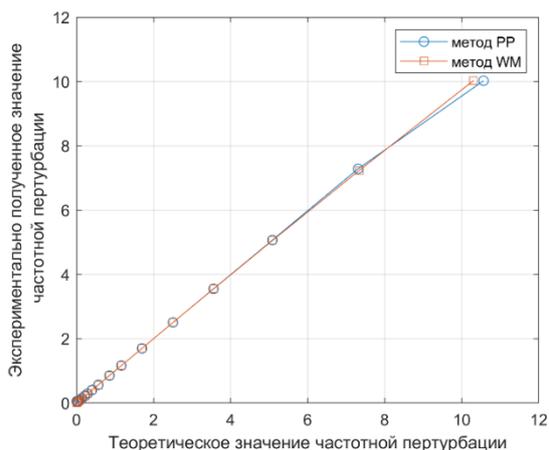


Рисунок 11 – Отношение экспериментально полученной частотной пертурбации к теоретическому значению

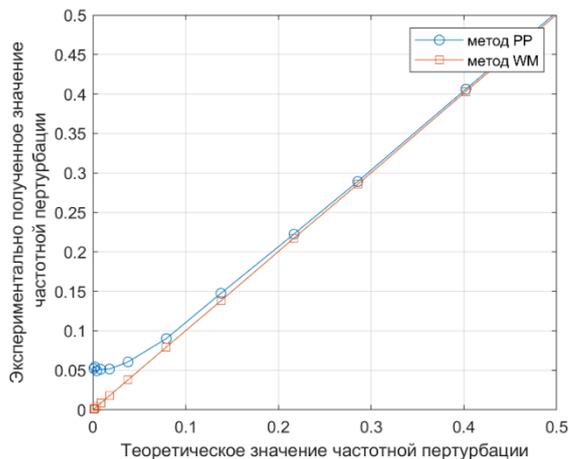


Рисунок 12 – Отношение экспериментально полученной частотной пертурбации к теоретическому значению (увеличенный масштаб)

Таким образом, в работе рассмотрены методы сегментации голосовых сигналов, а именно метод подгонки формы сигнала и метод отбора локальных максимумов. Из полученных графиков видно, что частоты основного тона извлекаются достаточно точно, однако при небольших коэффициентах частотной модуляции лучше использовать метод подгонки формы сигнала. При больших коэффициенте частотной модуляции не имеет значения какой метод использовать. Работоспособность методов показана путём MATLAB-моделирования.

Список использованных источников:

1. Comparison of F_0 Extraction Methods for High-Precision Voice Perturbation Measurements / Ingo R. Titze, Haixiang Liang // *Journal of Speech and Hearing Research*, 1993. – P. 14.
2. A Comparison of Healthy and Disordered Voices Using Multi-Dimensional Voice Program, Praat, and TF32 / Lap-Ching Keung [et al.] // *Journal of Voice*, 2022 – P. 16.

UDC 616.71

VOICE SIGNAL SEGMENTATION METHODS

Pasternak V.V., Stetsko V.Y.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Vashkevich M.I. – PhD in Technology

Annotation. The paper considers voice signal segmentation methods, namely the Waveform Matching method and the Peak-Picking method. The result of these methods is the extraction of the fundamental frequency of the voice signal. The methods were tested on test signals having a sinusoidal shape and various frequency modulation. To test the performance, the methods were compared in relation of the frequency perturbation parameters to their theoretical values.

Keywords. Voice signal segmentation, fundamental frequency.

АНАЛИЗ АРХИТЕКТУРНЫХ РЕШЕНИЙ ПРОЦЕССОРА SHA-3

Ероховец В.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Станкевич А.В. – канд. техн. наук)

Развитие информационных технологий требует постоянного совершенствования средств, обеспечивающих информационную безопасность. Криптографические хэш-функции используются в алгоритмах цифровой подписи, для проверки подлинности и целостности полученного сообщения и позволяют вычислить для произвольного сообщения переменной длины хэш-значение фиксированной длины. Одна из таких функций описывается алгоритмом SHA-3, принятым в 2015 году в качестве стандарта Национальным институтом стандартов и технологий США (NIST).

SHA-3 (Кескак) – алгоритм хеширования сообщений переменной длины. В алгоритме предложено использовать новое решение, такое как криптографическая губка. При таком решении процесс вычисления хэш-значения разбивается на два этапа: впитывание – преобразование блоков входного сообщения с помощью функции перемешивания; и отжимание – выделение из вектора состояний частей итогового хэш-значения и дальнейшее преобразование с помощью функции перемешивания. Структура конструкции губки представлена на рисунке 1 [1].

Алгоритм перемешивания представляет собой итеративный алгоритм с заранее заданным числом итераций, что позволяет использовать как схемы с обратной связью (итеративные вычислительные устройства), так и реализовывать конвейерные вычисления. Используемые внутри него раундовые функции модифицируют введенный в стандарте массив состояний, который формируется из входного вектора сообщения. Следует иметь в виду, что поскольку длина входного сообщения заранее не известна, то количество использований функции перестановки f (рисунок 1) будет переменной величиной. В зависимости от требуемых характеристик устройства, возможны несколько вариаций архитектур [2].

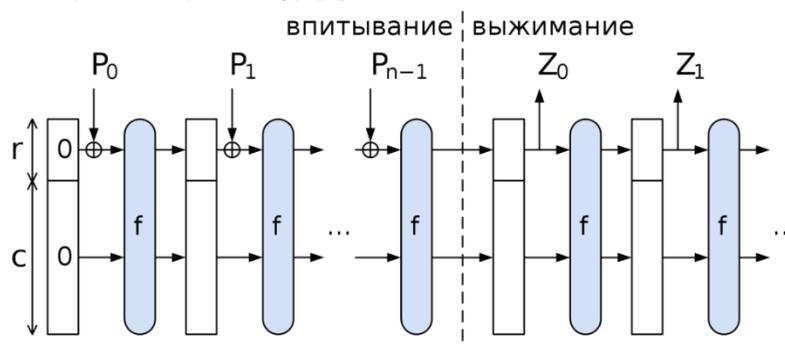


Рисунок 1 – Структурная схема блока, выполняющего функцию “криптографической губки”

Для сравнения были выбраны архитектура с использованием блока управления и памяти, представленная на рисунке 2, и вычислительного блока, использующего последовательное вычисление шагов раунда алгоритма, представленная на рисунке 3.

Основными различиями архитектур стал подход к организации вычислений и хранению массива состояний. Для архитектуры с блоком управления и памятью необходимо иметь некоторое количество состояний, которые описывают процесс вычисления в конкретный момент. Этими состояниями и оперирует блок управления. Вычисление раундовых функций выделено в отдельные вычислительные блоки, которые взаимодействуют с памятью массива состояний через устройство управления [3].

Использование последовательной схемы обусловлено отказом от хранения массива состояний в памяти. Вследствие этого схема вычислений приобретает несколько другой вид. При организации вычислений раундовых функций друг за другом в одной схеме необходимо передавать изменённый в предыдущем блоке массив на блок следующей раундовой функции. Для организации цикла в схеме используется мультиплексор, который позволяет подать на вход вычислителя стартовый вектор или выходной вектор предыдущего такта работы алгоритма для корректной работы устройства [3].

В обоих рассмотренных случаях необходимо внешнее управление процессом вычисления и приёма входного сообщения. В схеме вычисления SHA-3 данные процессы выделены в отдельные структурные блоки. Полная структурная схема устройства, реализующего стандарт хеширования SHA-3, представлена на рисунке 4.

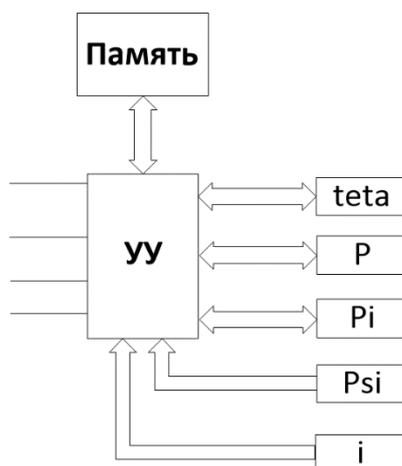


Рисунок 2 – Структурная схема вычислителя Кессак на основе архитектуры процессора общего назначения

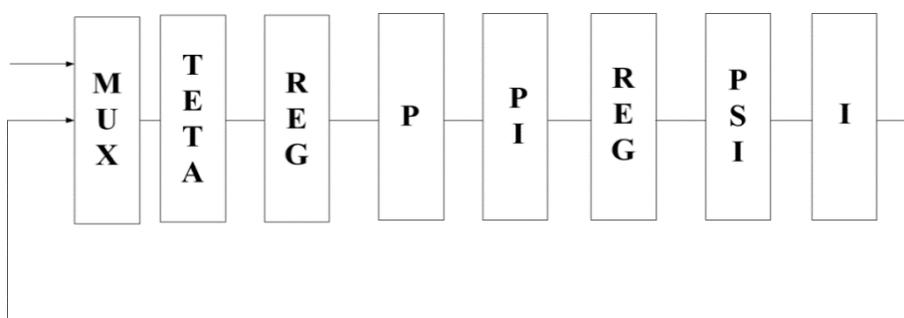


Рисунок 3 – Структурная схема вычислителя Кессак, использующего последовательное вычисление шагов раунда алгоритма



Рисунок 4 – Структурная схема предлагаемого устройства

Полученные архитектурные решения описаны на языке VHDL и реализованы в среде Xilinx ISE. Их сравнение проводилось по следующим критериям: пропускная способность, производительность, выраженная в тактовой частоте устройства и занимаемых ресурсах кристалла FPGA. Сравнительная характеристика реализаций для кристалла XC6SLX75T представлена в таблице 1.

Таблица 1 – Результаты операции place-and-route аппаратных реализаций устройств

Реализация	Тактовая частота(МГц)	Slice	LUT
С памятью	130	824	2418
Последовательная	210	1311	5235

Полученные результаты позволили сделать вывод о целесообразности использования той или иной архитектуры устройства хеширования на основе алгоритма SHA-3. Были оценены основные параметры устройства, критически важные для конечных пользователей.

Список использованных источников:

1. Wikipedia[Электронный ресурс]. – Электронные данные. – Режим доступа: <https://ru.wikipedia.org/wiki/SHA-3/>
2. OpenCores[Электронный ресурс]. – Электронные данные. – Режим доступа: <https://opencores.org/projects/sha3>
3. Compact FPGA Implementations of the Five SHA-3 Finalists. / S. Kerckhof [et al.]/ Researchgate, January 2011. P.71-74

ВЕЛОКОМПЬЮТЕР НА БАЗЕ МИКРОКОНТРОЛЛЕРА

Юрченко Е.С.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Порхун М.И. – магистр техн. наук

В работе представлен велокомпьютер на базе микроконтроллера ATmega328. Устройство позволяет измерять скорость, расстояние, а также отслеживать местоположение. Представлена структура и описание самого устройства. Приведён обобщённый алгоритм работы системы. Проведена проверка корректности работы устройства в среде Proteus.

Современный велокомпьютер – это компактное электронное устройство позволяющее контролировать широкий спектр показателей при езде на велосипеде. Нынешний рынок не ограничивает потребителей в выборе этих чрезвычайно полезных и многофункциональных аксессуаров. Бортовые велогаджеты снискали должное признание в среде как профессионалов, так и рядовых велосипедистов. Они незаменимы для мониторинга и анализа спортивных тренировок, а также позволяют оценить собственные возможности во время длительных заездов и непродолжительных прогулок [1].

Устройство обеспечивает измерение и отображение следующих параметров:

- Текущая скорость;
- Пройденное расстояние за поездку;
- Общее пройденное расстояние;
- Средняя скорость пути за поездку;
- Максимальная скорость пути за поездку;
- Текущая дата и время;
- Продолжительность поездки.

Структурная схема устройства представлена на рисунке 1

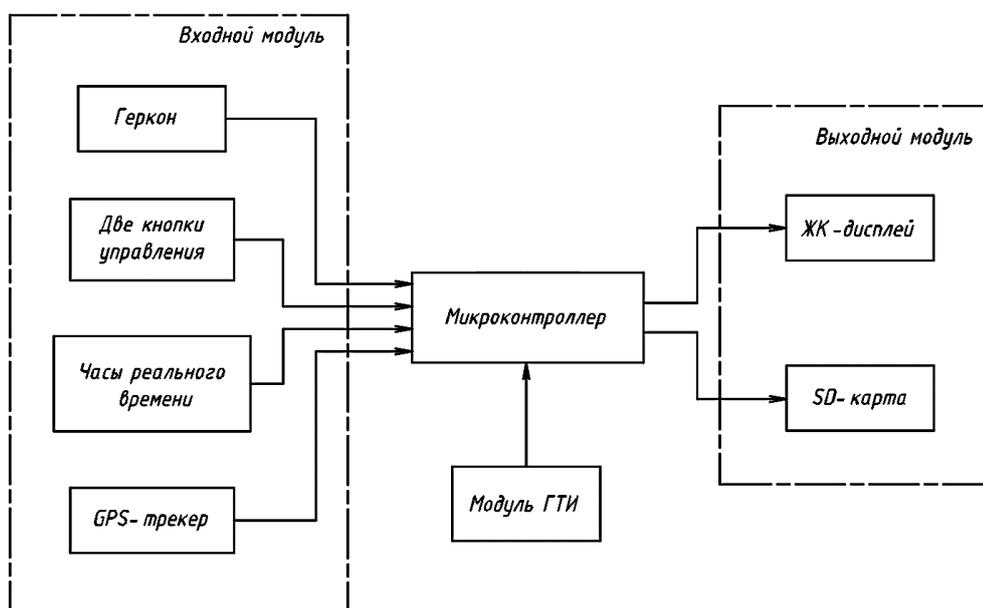


Рисунок 1 – Структурная схема велокомпьютера

Устройство реализовано на базе микроконтроллера ATmega328 [2]. Считывание данных о частоте оборотов колеса выполняется при помощи геркона, закрепленного на вилке велосипеда. С помощью кнопок управления «RES/SET» и «MENU» осуществляется взаимодействие пользователя с интерфейсом устройства. Кнопка «RES/SET» устанавливает и сбрасывает встроенный таймер, сохраняет общий пробег. Посредством кнопки «MENU» ведётся переключение между страницами интерфейса велокомпьютера. Устройство также предоставляет пользователю возможность настроить дату, время и диаметр колеса велосипеда с помощью этих кнопок. Часы реального времени (RTC) используются для получения информации о текущей дате и времени [4]. Для отображения информации использован двухстрочный ЖК-индикатор LM016L [3]. Также

велосипед велоконピューтер ведет фиксацию местоположения пользователя с помощью GPS-трекера и передает данные на SD карту.

Далее приводится обобщенный алгоритм работы устройства.

На первом этапе выполняется начальная настройка микроконтроллера. Далее запускается главный цикл программы, в котором последовательно выполняются следующие действия:

1. Осуществляется считывание сигналов, пришедших с кнопок управления;
2. Производится проверка флага настройки. Если он включен перейти к пункту 3, иначе перейти к пункту 4;
3. Производится обработка нажатий кнопок в режиме настройки. Далее перейти к пункту 5;
4. Производится обработка нажатий кнопок в основном режиме;
5. Производится расчет мгновенной скорости, пройденного расстояния, средней и максимальной скорости;
6. Выполняется прием данных с GPS-трекера, их обработка и передача на SD-карту;
7. Выводится интерфейс велосипеда на дисплей.

Для проверки работоспособности устройства была проведена симуляция в среде моделирования Proteus.

На первой странице интерфейса велосипеда (рисунок 2) отображается текущая скорость, пройденное расстояние за поездку и текущая длительность поездки.

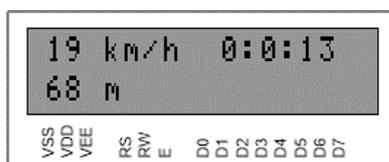


Рисунок 2 – Первая страница интерфейса велосипеда

При однократном нажатии на кнопку «MENU» происходит переход на следующую страницу интерфейса, которая отображает текущую дату и время (рисунок 3).

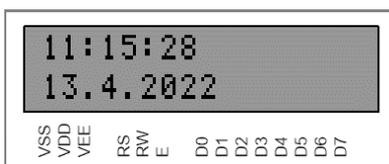


Рисунок 3 – Отображение текущей даты и времени

При последующих нажатиях на кнопку «MENU» выполняется дальнейший переход на страницы отображения средней и максимальной скорости соответственно. Помимо того, отображается общее пройденное расстояние за время работы устройства (рисунок 4).



Рисунок 4 – Отображение скоростей и общего расстояния: а) отображение средней скорости; б) отображение максимальной скорости

На рисунке 5 представлен пример файла с GPS координатами (история местоположений), которые были записаны на SD-карту.

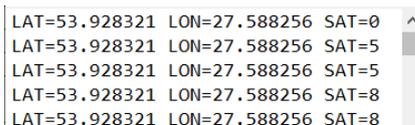


Рисунок 5 – Файл с координатами, которые были получены с GPS-трекера

В результате был реализован велосипед на базе микроконтроллера ATmega328. Устройство обладает всеми основными функциями современных велосипедов. Описан обобщенный алгоритм работы системы. Было проведено моделирование в среде Proteus, которое показало, что все функции работают корректно.

Список использованных источников:

1. Статья про велокомпьютеры [Электронный ресурс]. – Режим доступа: <https://web.archive.org/web/20160918222717/http://www.velomagazin.ru/articles/vybor-velosipednogo-kompyutera/>
2. Описание микроконтроллера ATmega328 [Электронный ресурс]. – Режим доступа: https://www.mouser.com/pdfdocs/gravitech_atmega328_datasheet.pdf
3. Описание ЖК дисплея LM016L [Электронный ресурс]. – Режим доступа: <https://www.farnell.com/datasheets/305035.pdf>
4. Описание часов реального времени DS3231 [Электронный ресурс]. – Режим доступа: <https://datasheets.maximintegrated.com/en/ds/DS3231.pdf>

ЭЛЕКТРОННАЯ КНИГА НА БАЗЕ ESP32

Каленик С.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Порхун М.И. – магистр техн. наук

В работе рассматривается задача реализации электронной книги на базе модуля ESP32. Приведена структурная схема устройства и её описание, а также обобщённый алгоритм функционирования устройства. Проверка работоспособности устройства выполнена опытным путём.

Электронная книга – это устройство, использующее экран на электронных чернилах для отображения текста. В отличие от обычных матричных экранов, текст не пропадает при подаче питания, а также не требуется свет для отображения. Благодаря этому уменьшается нагрузка на глаза и читателю приятней использовать электронную книгу, чем, например, телефон. Еще одной отличительной чертой данного типа устройств является низкое энергопотребление, поскольку электричество тратится лишь на смену текста на экране, в отличие от обычных матричных экранов, в которых включенный экран постоянно потребляет много энергии [1].

Структурная схема устройства представлена на рисунке 1.

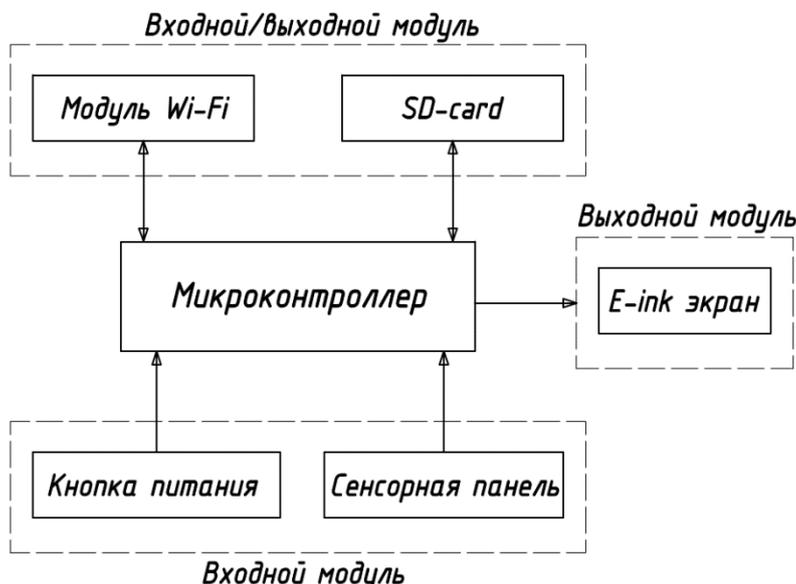


Рисунок 1 – Структурная схема электронной книги

Центральной частью схемы является микроконтроллер. Он обеспечивает взаимодействие всех элементов устройства. Микроконтроллер взаимодействует с Wi-Fi модулем для предоставления возможности загружать книги при помощи телеграмм-бота. Также микроконтроллер связан с модулем SD-карты для чтения и записи книг. На E-ink экране [2] отображается текст книги, выбранной пользователем. Управление электронной книгой осуществляется при помощи сенсорной панели. Также для экономии заряда батареи используется кнопка питания для включения/выключения устройства.

Обобщённый алгоритм работы системы приведён ниже:

1. На первом этапе устройство сканирует все файлы в памяти и добавляет их имена в список книг, если формат файла поддерживается;
2. Далее формируется главное меню устройства, в центре которого отображается список из 3 книг, а также указатели для перемещения по списку книг (указатель слева – отображение предыдущих 3 книг из общего списка, справа – следующие 3 книги). Список книг отображается в виде вытянутого по горизонтали прямоугольника, в центре которого указано название книги без учёта формата файла и пути к нему;
3. На следующем шаге пользователь должен при помощи нажатия на соответствующие пункты меню на экране выбрать книгу из отображаемых или сменить страницу для дальнейшего поиска и выбора нужной книги;
4. Далее, когда выбрана необходимая книга, по её полному пути в директории происходит чтение файла;

5. На следующем этапе происходит обработка текста входе которой определяется, сколько слов поместится в одну строку на экране. Определение происходит следующим образом:
 - 5.1. За стартовый индекс начала строки принимается ноль;
 - 5.2. Выбирается N следующих символов. Число N задаётся больше максимального количества символов в строке. Индекс конца строки равен сумме стартового индекса и N;
 - 5.3. Производится проверка поместится ли текст, размещенный между стартовым и конечным индексами;
 - 5.4. Если текст не помещается, тогда происходит смещение конечного индекса таким образом, чтобы сместить конечный индекс ровно на длину одного слова с конца. Далее происходит выполняется возвращение к пункту 5.3;
 - 5.5. Сохранение индексов начала и конца итоговой строки;
 - 5.6. Если индекс конца строки меньше, чем длина текста книги, тогда стартовый индекс равен конечному. Переход к пункту 5.2;
6. Далее необходимо рассчитать итоговое количество страниц книги, которое равно общему количеству строк, деленное на количество строк, которые можно отобразить на экране. Результат округляется в большую сторону;
7. Происходит отображение текста текущей страницы при помощи индексов, подсчитанных в 5 пункте. Также отображается номер текущей страницы относительно количества всех страниц в книге;
8. В случае нажатия на экран происходит переход на следующую или предыдущую страницу. Левая сторона сенсора отвечает за переход назад, а правая – вперёд;
9. В случае нажатия на текущий номер страницы или общее количество страниц происходит возвращение к пункту 1;
10. В случае нажатия на кнопку питания происходит отображение рисунка уведомляющего пользователя об включении/выключении устройства.

На рисунке 2 приведено фото разработанной электронной книги с активным режимом чтения.

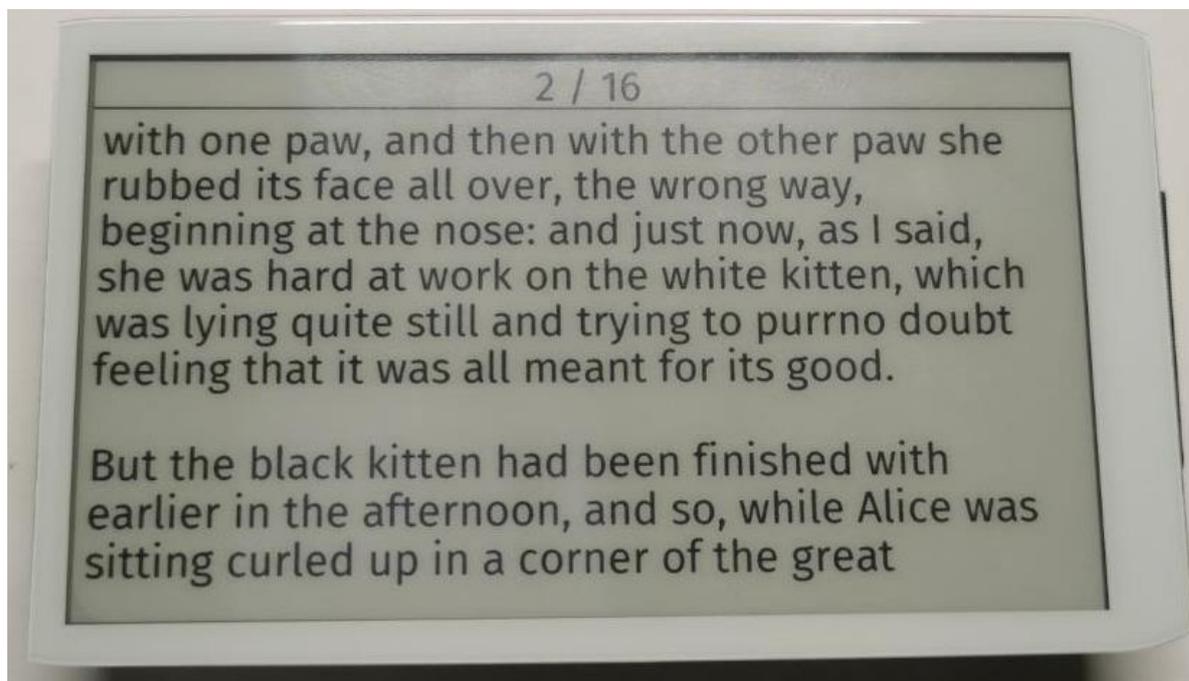


Рисунок 2 – Пример отображения текста книги

В данной работе приведён процесс разработки электронной книги на базе микроконтроллера ESP32 с использованием экрана на электронных чернилах и сенсорной панели. Также приведена структурная схема с кратким описанием взаимодействия элементов в ней. Рассмотрен общий алгоритм функционирования устройства. Также выполнена проверка работоспособности устройства опытным путем.

Список использованных источников:

5. Электронная книга (устройство) [Электронный ресурс]. – Режим доступа: <https://en.wikipedia.org/wiki/E-reader>

6. Техническое описание E-ink экрана [Электронный ресурс]. – Режим доступа: https://github.com/Xinyuan-LilyGO/LilyGo-EPD47/blob/master/Display_datasheet.pdf

ТЕСТ ИЗНОСОУСТОЙЧИВОСТИ ЭНЕРГОНЕЗАВИСИМОЙ ПАМЯТИ МИКРОКОНТРОЛЛЕРОВ

Трапашко И.И., Деменковец Д.В.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Леванцевич В.А. – ст. преподаватель

В работе приводятся краткие сведения об энергонезависимой памяти микроконтроллеров STM32F103C8T6 и ATmega328P, приведены результаты тестов износоустойчивости, включая коэффициент запаса и описание возникших ошибок. Описан алгоритм уменьшения износа памяти.

Для анализа и тестирования в качестве тестовых экземпляров были выбраны flash-память микроконтроллера STM32F103C8T6 и EEPROM микроконтроллера ATmega328P.

STM32F103C8T6 является 32-разрядным микроконтроллером на основе процессора Arm Cortex M3. Процессор основан на гарвардской архитектуре, что позволяет одновременно выбирать инструкции и выполнять чтение/запись в память. Память программ, данные, регистры и порты ввода-вывода располагаются в одном адресном пространстве размером 4 гигабайта. Области адресного пространства, которым не соответствует физическая память или регистры, помечены как «Зарезервированные» [1]. В качестве энергонезависимой памяти выступает flash-память, в которой хранится код программы. К ней через интерфейс flash-памяти подключена шина инструкций. Микроконтроллер относится к устройствам со средней плотностью памяти, её размер составляет 128 килобайт. Память разделена на 128 страниц размером по 1 кб. После сброса микроконтроллера память доступна только для чтения и заблокирована для записи. Чтобы разблокировать память, необходимо записать определённое значение в регистр управления. Ячейка, в которую производится запись, должна быть предварительно очищена (значение всех бит должно равняться 1). При попытке записать значение в не очищенную ячейку, запись произведена не будет и будет возвращена ошибка «programming error». Очистка памяти производится постранично. Доступ к данным производится блоками по 16 бит. Время записи блока данных размером 16 бит в память составляет от 40 до 70 микросекунд, время очистки страницы – от 20 до 40 миллисекунд. В микроконтроллере отсутствует EEPROM, поэтому для хранения пользовательских данных используется область flash-памяти, не занятую кодом [2]. Тестирование памяти проводилось по алгоритму, изображённому на рисунке 1.

```
РазблокироватьПамять()  
Пока НетОшибкиЗаписи  
  ОчиститьСтраницу()  
  ЗаписатьНовоеЗначение()  
  Если ЗаписаноУспешно  
    ПрочитанноеЗначение = ПрочитатьЗаписанноеЗначение()  
    Если ЗаписанноеЗначение != ПрочитанноеЗначение  
      Ошибка = "Записанное значение не равно прочитанному значению"  
    Иначе  
      Ошибка = "Ошибка программирования памяти"  
    Итерация++  
  ЗabloкироватьПамять()
```

Рисунок 1 – Алгоритм тестирования flash-памяти микроконтроллера STM32F103C8T6

По данным производителя, flash-память STM32F103C8T6 рассчитана на минимум 10 тысяч циклов перезаписи без ошибок [3].

ATmega328P – микроконтроллер на основе 8-разрядного процессора AVR с гарвардской архитектурой. Память микроконтроллера представлена тремя линейными адресными пространствами: память данных, память программ (flash-память) и EEPROM. Размер EEPROM составляет 1 килобайт. Доступ к этой памяти осуществляется с помощью регистров, которые находятся в пространстве данных. Данные передаются по 8-ми разрядной шине данных. EEPROM, в отличие от flash-памяти, не требует очистки перед записью. Память выдерживает 100 тысяч циклов чтения-записи без ошибок. Среднее время записи одного байта в память составляет 3,3 миллисекунды [4]. Тестирование проводилось алгоритму, аналогичному, как и для контроллера STM32F103C8T6, изображённому на рисунке 1.

Были проведены тесты для трёх смежных ячеек памяти контроллеров. Результаты тестов представлены в таблице 2.

В микроконтроллере STM32F103C8T6 ошибка проявилась следующим образом: ячейка памяти не была корректно очищена, из-за чего последующая запись не была выполнена.

Таблица 1 – Результаты тестов энергонезависимой памяти микроконтроллеров.

Микроконтроллер	Тип памяти	Заявленный ресурс, кол-во циклов	Среднее количество циклов до первой ошибки	Коэффициент запаса износоустойчивости
STM32F103C8T6	Flash	10 000	1 411 234	141,1
ATmega328P	EEPROM	100 000	6 531 131	65,3

Ожидаемое значение после очистки: 0xFFFF, фактическое: 0xDFFF. Для микроконтроллера ATmega328P прочитанное значение (0xB3) оказалось не равно записанному значению (0xB2). Для обоих микроконтроллеров последующие ошибки были аналогичны первым.

Количество итераций между последующими ошибками приведено на рисунке 2. По оси x указан порядковый номер ошибки, по оси y – количество успешных циклов чтения-записи после предыдущей ошибки. Для микроконтроллера STM32F103C8T6 среднее количество циклов между ошибками после первой ошибки составило 13, медианное – 4. Для ATmega328P – 825 и 32 соответственно.

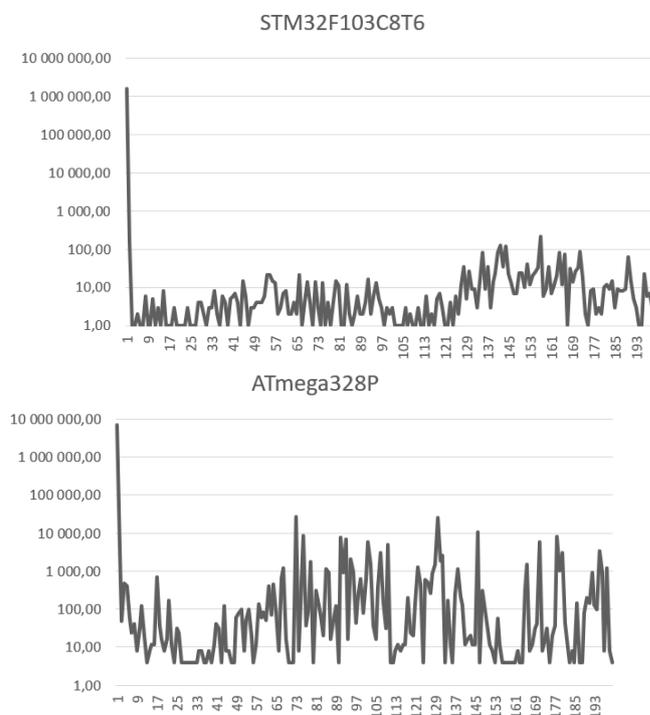


Рисунок 1 – Количество успешных циклов чтения-записи между ошибками

Количество циклов чтения-записи до первой ошибки при тестировании смежных ячеек в обоих микроконтроллерах отличалось не более чем на 5%. Таким образом можно сделать вывод, что износ ячейки памяти не оказывает влияния на смежные ячейки.

Поскольку flash-память не позволяет изменять значение отдельных элементов без предварительной очистки страницы, её нельзя напрямую использовать как EEPROM. Существует алгоритм, позволяющий эмулировать работу EEPROM. Алгоритм основан на использовании как минимум двух физических страниц. Рядом с каждым 16-битным словом хранится его виртуальный адрес. Когда приходит запрос на изменение записанного значения, новое значение записывается в следующую свободную ячейку, сохраняя свой виртуальный адрес, но изменяя физический. Таким образом, происходит накопление истории изменений значения, и чтобы узнать значение переменной по определённому виртуальному адресу, необходимо найти последнюю запись с заданным адресом. Когда страница заполняется, производится копирование данных во вторую страницу с сохранением только актуальных значений переменных. После этого первая страница очищается. При заполнении второй страницы данные копируются в первую, вторая страница очищается и цикл продолжается [5].

Одно из преимуществ данного алгоритма состоит в том, что он позволяет регулировать нагрузку на ячейки памяти, тем самым продлевая срок их службы. Чем больше физических

страниц используется для эмуляции виртуальных, тем реже каждая из них подвергается записи и очистке. Размер страницы flash-памяти составляет 1 килобайт, количество циклов перезаписи равно 10 тысячам циклов. Следовательно, ресурс страницы составляет 10 000 килобайт. При использовании двух страниц для эмуляции EEPROM ресурс возрастает до 20 000 килобайт. Использование большего количества страниц увеличивает ресурс памяти.

Список использованных источников:

1. RM0008 Reference manual (STM32F101xx, STM32F102xx, STM32F103xx, STM32F105xx and STM32F107xx advanced Arm®-based 32-bit MCUs) [Электронный ресурс]. – Режим доступа: https://www.st.com/resource/en/reference_manual/cd00171190-stm32f101xx-stm32f102xx-stm32f103xx-stm32f105xx-and-stm32f107xx-advanced-arm-based-32-bit-mcus-stmicroelectronics.pdf. – Дата доступа: 05.04.2022.
2. PM0075 Programming manual [Электронный ресурс]. – Режим доступа: https://www.st.com/resource/en/programming_manual/pm0075-stm32f10xxx-flash-memory-microcontrollers-stmicroelectronics.pdf. – Дата доступа: 05.04.2022.
3. STM32F103x8 STM32F103xB Datasheet [Электронный ресурс]. – Режим доступа: <https://www.st.com/resource/en/datasheet/stm32f103c8.pdf>. – Дата доступа: 05.04.2022.
4. ATmega328P. 8-bit AVR Microcontroller with 32K Bytes In-System Programmable Flash. Datasheet. [Электронный ресурс]. – Режим доступа: https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf. – Дата доступа: 05.04.2022.
5. EEPROM emulation in STM32F10x microcontrollers. [Электронный ресурс]. – Режим доступа: https://www.st.com/resource/en/application_note/cd00165693-eprom-emulation-in-stm32f10x-microcontrollers-stmicroelectronics.pdf. – Дата доступа: 05.04.2022.

УСТРОЙСТВО НА БАЗЕ ПЛИС ДЛЯ УЛУЧШЕНИЯ КАЧЕСТВА ИЗОБРАЖЕНИЙ В ВИДИМОМ И БЛИЖНЕМ ИНФРАКРАСНОМ ДИАПАЗОНАХ

Жук В. Г.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Шемаров А. И. – канд. тех. наук

В настоящей работе предлагается метод улучшения изображения путем объединения видимого и ближнего инфракрасного диапазонов. Цель — представить простой и аппаратно-эффективный метод слияния, в результате чего получается объединенное улучшенное изображение.

Большая часть существующих работ в области компьютерного зрения основаны на том, что все световые лучи отраженные от объектов, смогут достичь наблюдателя без каких-либо изменения, так как множество датчиков были разработаны для работы в хороших погодных условиях. Однако частицы в атмосфере рассеивают свет, что приводит к ухудшению видимости. Величина рассеяния зависит главным образом от типа и размера частиц и их концентрации в атмосфере [1]. Визуальным результатом рассеяния света является потеря контраста и размытие отдаленных объектов. Усиление контраста можно использовать для устранения дымки на изображении, но поскольку эффект дымки и тумана не является постоянным на изображении, этот метод нельзя применять глобально. Существуют методы устранения дымки на одном изображении, основанные на физической модели того, как дымка ухудшает изображение, но этот алгоритм может давать неудовлетворительный результат на изображения без тумана или дымки.

Однако, в 2008 году, Шауль предложили новый подход к улучшению изображения в условиях дымки [2]. Вместо того, чтобы использовать сложные алгоритмы устранения дымки на одном изображении на основе физических моделей, он рассматривают одну и ту же сцену с двумя изображениями в разных цветовых диапазонах. Одно изображение делается в видимой области спектра, а другое — в ближней инфракрасной. Так как из закона рассеивания Рэлея, который гласит, что более длинные волны меньше рассеиваются частицами в воздух. Из данного закона также следует, что инфракрасный свет менее подвержен рассеиванию (инфракрасный спектр охватывает длины волн от 780 нм до 3 мм), чем видимый, из чего можно сделать вывод, что инфракрасное изображение могут быть в меньшей степени искажены дождем, снегом или же атмосферной дымкой, в отличии от обычного снятого изображения с матрицы.

Шауль предлагают многомасштабный алгоритм слияния изображений, который использует два изображения, одно визуальное и одно инфракрасное. Вместо стандартных решений мелкомасштабной декомпозиции с пирамидальной декомпозицией или вейвлет-преобразованием Шауль предложил использовать структуру оптимизации взвешенных наименьших квадратов (WLS). В предложенном ими решении, артефакты гало, проблема, возникающая при объединении изображений по большей части, избегаются. Шауль мотивируют свой выбор WLS вместо вейвлет-подхода, указывая на то, что WLS является переполненным, что приводит к более надежному решению.

В общих чертах процесс проектирования можно определить четырьмя этапами:

1. Получение одной и той же сцены видимого и инфракрасного изображения с разными настройками.
2. Преобразование видимого изображения в соответствующие цветовые пространства YCbCr и HSV.
3. Замена канала яркости видимого изображения на на канал инфракрасного.
4. Преобразовать полученное изображение в цветное пространство видимого.

Адаптация алгоритма декомпозиции алгоритм Шауля обеспечивает представление изображений с несколькими разрешениями путем выполнения декомпозиции в несколько слоев. Использование алгоритма только с одним слоем, $n = 1$, дает результат, который не сильно отличается от результата, полученного при использовании большего количества слоев. Однако это значительно ускоряет расчеты. Использование одного слоя также значительно уменьшит объем и сложность реализации, поскольку необходимо рассчитать только одно составное изображение.

Чтобы в полной мере воспользоваться возможностями, которые может предоставить решение ПЛИС, и использовать тот факт, что большую часть обработки изображений можно выполнять попиксельно без межпиксельных помех, проектирование выполнено конвейерным параллельным способом. Пиксели изображения подаются в канал друг за другом, таким образом,

что новый пиксель вводится, в то время как несколько предыдущих пикселей все еще на разных стадиях обработки. Это делается для всех входных сигналов. Сущности в верхних регистрах определяются перед началом выполнения. Псевдокод для трех этапов указан в формуле (1):

$$\begin{aligned} visD\acute{e}tail &= visDetail - VIS_DETAIL_BRIGHTNESS, \\ nirD\acute{e}tail &= nirDetail - NIR_DETAIL_BRIGHTNESS, \\ visC\acute{o}arse &= visCoarse + VIS_COARSE_BRIGHTNESS, \\ nirC\acute{o}arse &= nirCoarse + NIR_COARSE_BRIGHTNESS, \\ visC\acute{o}ntrast &= visContrast + VIS_CONTRAST_BRIGHTNESS, \\ nirC\acute{o}ntrast &= nirContrast + NIR_CONTRAST_BRIGHTNESS \end{aligned}$$

$$visC\acute{o}ntrast = \frac{visD\acute{e}tail - visC\acute{o}arse}{visC\acute{o}arse},$$

$$nirC\acute{o}ntrast = \frac{nirD\acute{e}tail - nirC\acute{o}arse}{nirC\acute{o}arse},$$

$$composite = (visC\acute{o}ntrast, nirC\acute{o}ntrast) \cdot visC\acute{o}arse + COMPOSITE_BRIGHTNESS \quad 1),$$

Значения в верхнем регистре — это наборы данных которые определены перед началом выполнения слияния. Чтобы лучше проиллюстрировать проблемы, возникающие при проектировании, эти константы принимаются равными нулю, следовательно, произойдет упрощение уравнением (1) к уравнениям (2):

$$visContrast = (visDetail - visCoarse) \cdot \frac{1}{visCoarse},$$

$$nirContrast = (nirDetail - nirCoarse) \cdot \frac{1}{nirCoarse},$$

$$composite = (visContrast, nirContrast) \cdot visC\acute{o}arse \quad 2),$$

где все величины имеют значения в пикселях, которые могут принимать любое значение от 0 до 255. В полной реализации ПЛИС, используются константы, поскольку ПЛИС не может напрямую выполнять деление, кроме как посредством битового сдвига, в приведенных выше выражениях деление на visCoarse и nirCoarse требует особого внимания.

Поскольку интенсивность пикселей находится в диапазоне от 0 до 255, существует 256 возможных результатов деления на входное значение пикселя. Быстрый и менее ресурсозатратный вариант: выполнить деление ограниченного числа значений — поместить результат в интерполяционную таблицу. Таким образом можно избежать введения представлений, которое истощает ресурсы ПЛИС и увеличивает время вычислений. Однако для приложения ПЛИС справочная таблица должна содержать только целочисленные приближения к результату деления.

Список использованных источников:

1. Karlsson. J. *FPGA-Accelerated Dehazing by Visible and Near-infrared Image Fusion: Independent thesis Advanced level: 11.06.2015 / J. Karlsson // Mälardalen University, 2015. – 48 P.*
2. L. Schaul, C. Fredembach, and S. Süsstrunk, *Color image dehazing using the near-infrared/L. Schaul // The proceedings of the 16th IEEE International conference on Image processing,- 2009.p .1629 - 1632 P.*

IP-ЯДРО ЦИФРОВОГО АУДИО ЭКВАЛАЙЗЕРА ДЛЯ ПЛИС XILINX ZYNQ

Петровский Д.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Петровский Н.А. – канд. тех. наук

Аннотация. В докладе показана реализация цифрового многополосного эквалайзера на языке описания аппаратуры VHDL на основе MAC-ядра DSP48E1 для арифметики с фиксированной точкой. Полосовые БИХ-фильтры получены на основе аппроксимаций Баттерворта. Полученное цифровое устройство оформлено в виде IP-ядра и выполнен синтез для кристалла Xilinx Zynq 7010. Полученная тактовая частота работы эквалайзера позволяет обрабатывать аудио сигнал в реальном времени.

Цифровой многополосный эквалайзер является одним из основных устройств применяемых в аудиотехнике для коррекции АЧХ записывающих и воспроизводящих устройств. Существует два основных подхода к реализации таких систем: обработка аудиосигнала в частотной или временной области. В рамках данного доклада применяется подход на основе временно анализа с помощью полосовых фильтров на основе аппроксимации Баттерворта. Применение ПЛИС для задач цифровой обработки сигналов является предпочтительной с точки зрения возможности выполнить обработку с минимальной возможной технической задержкой, а также параллелизма вычислений.

Аудио эквалайзер функционирует в соответствии со структурной схемой (см рис.1), где цифровой аудио сигнал $x[n]$, поступающий на устройство через входной регистр данных, и передается на k БИХ фильтров N -ого порядка с соответствующими коэффициентами разностного уравнения $\{a, b\}$, которые разделяют частотный диапазон на k поддиапазонов. Результат фильтрации от каждого из фильтров поступает на блоки умножения, где производится умножение на коэффициент ослабления (g_1, g_2, \dots, g_k) , заданный в регистрах устройства управления. Результат блоков умножения передается в дерево сумматоров для формирования окончательного результата $y[n]$ и записи его в регистр результата [1].

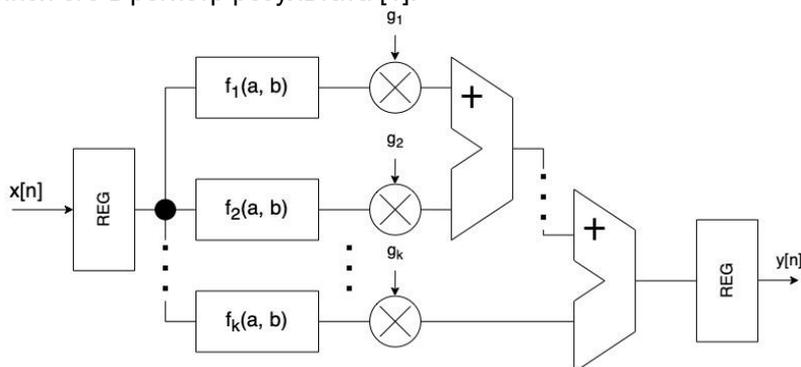


Рисунок 1 – Общая структурная схема k-полосного цифрового эквалайзера

Для моделирования и проектирования IP-ядра аудиэквалайзера задано $k = 3$ субполос и аппроксимация фильтра Баттерворта 7-ого порядка. Формат исходных данных и результата: импульсно-кодовая модуляция в 16 бит, частота дискретизации 44100 Гц. Устройство для обработки аудиоданных проектировалось на языке описания аппаратуры VHDL [2].

Пример результата обработки разработанного эквалайзера показан на рис. 2, где изображены тестовый аудио сигнал длительностью до 40 секунд (рис. 2а), его спектрограмма (рис. 2б) и спектрограмма полученного на выходе эквалайзера результата (рис. 2в).

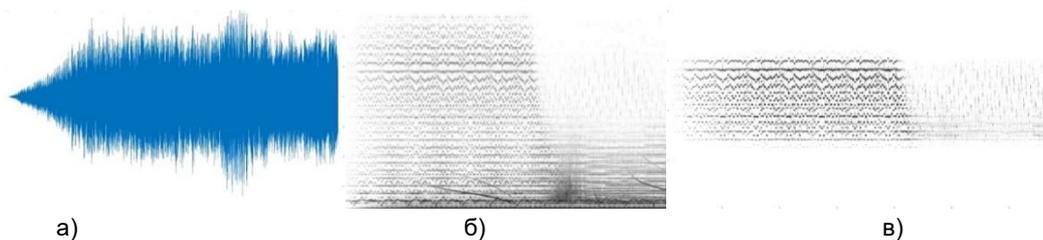


Рисунок 2 – Результаты моделирования: а) временное представление тестового аудио сигнал (40 сек, $f_s = 44100$ Гц) б) спектрограмма исходного сигнала (формат преобразования N = 1024, перекрытие 50%) в) спектрограмма обработанного аудио сигнала где $g_1 = -96dB$, $g_2 = 0dB$, $g_3 = -96dB$

Результаты проектирования фильтра нижних, средних и высоких частот показаны на рис 3, где приводится z-плоскость и АЧХ\ФЧХ соответствующих фильтров. Как можно заметить результирующая АЧХ при усилении в 0дБ почти не вносит искажений в обработанный аудио сигнал.

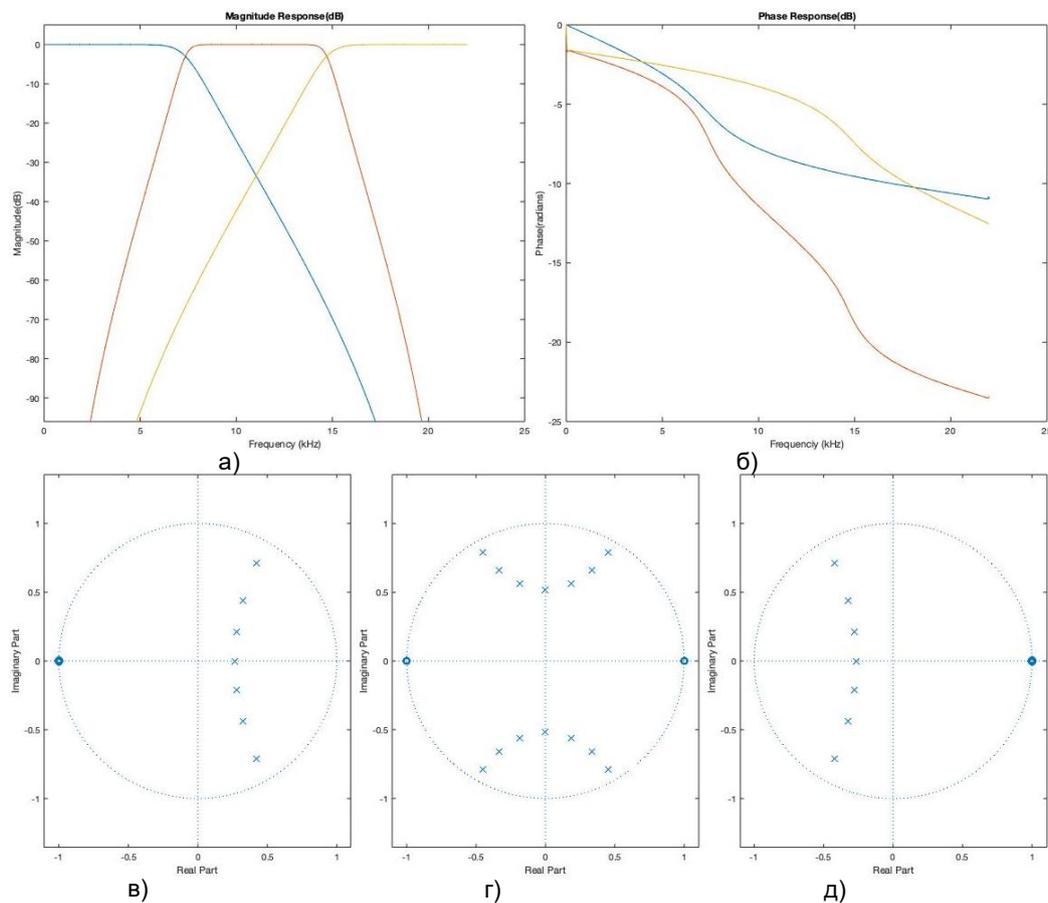


Рисунок 3 – Характеристики БИХ фильтров а) АЧХ б) ФЧХ в) отображение на Z плоскость фильтра нижних частот г) отображение на Z плоскость полосового фильтра д) отображение на Z плоскость фильтра верхних частот

В результате синтеза VHDL-описания эквалайзера аудио сигналов для кристалла серии Zynq 7010, было получены следующие значения показателей затраченных ресурсов, смотреть таблицу 1, где количество LUT и FF использовано 4 и 2 % соответственно от общего числа имеющихся ресурсов. Основная нагрузка при реализации эквалайзера легла на DSP48E1C модулей, было израсходовано 76 %.

Таблица 1 – Затраты ресурсов FPGA Zynq 7010 для 3-х полостного аудио эквалайзера

Тип ресурса	Количество	Всего	Использовано, %
LUT	795	17400	4
FF	911	35200	2
DSP48E1C	61	80	76

Из результатов синтеза видно, что период тактовых импульсов составляет 43.008 нс., что соответствует 23.251 МГц.

Анализируя полученные результаты, можно сделать вывод, что данная схема трехполосного эквалайзера занимает незначительные ресурсы на кристалле ПЛИС и сможет эффективно обрабатывать входные данные поступающие с частотой дискретизации 44100 кГц в реальном масштабе времени с минимальной задержкой.

Список использованных источников:

1. Вологдин Э.И. Методы и алгоритмы обработки звуковых сигналов. - СПб.: СПбГУТ, 2009. - 96 с.
2. Бибило П.Н. Основы языка VHDL. Издание второе, исправленное и дополненное. – СОЛОН-Р, 2002

ПРОГРАММНЫЙ МОДУЛЬ РАСПОЗНАВАНИЯ ОБРАЗОВ ДЛЯ МИКРОКОМПЬЮТЕРОВ НА ОСНОВЕ НЕЙРОННЫХ СЕТЕЙ СЕМЕЙСТВА YOLOV5

Ковбаса Г.А.¹, магистрант гр.055741

Белорусский государственный университет информатики и радиоэлектроники¹
г. Минск, Республика Беларусь

Азаров И.С. – докт. техн. наук

Аннотация. Проблема распознавания образов приобрела в настоящее время большой масштаб в связи с развитием автомобильных автопилотов, персональных ассистентов, технологий виртуальной и дополненной реальности. Однако существующие технологии не обладают достаточной эффективностью для быстрого обучения моделей и детектирования образов множества объектов в условиях получения выборки в реальном времени. В данной работе представлен компактный программный модуль на базе комбинации сети YOLOv5 и ShuffleNet V2, размер и количество параметров которого было оптимизировано для последующей работы на встраиваемых системах. Программный интерфейс модуля спроектирован с учетом возможности портирования на различные платформы.

Ключевые слова. Распознавание образов, YOLOv5, оптимизация нейронной сети, трансферное обучение.

Существующие технологии детектирования объектов на основе нейронных сетей не обладают достаточной эффективностью на микрокомпьютерах в условиях получения выборки в реальном времени в связи с высокой вычислительной сложностью сети и большим количеством параметров.

Целью данного исследования является решение проблемы распознавания образов в режиме реального времени для маломощных и встроенных систем, а также применение подхода трансферного обучения для быстрого переобучения нейронной сети.

Концепция разработки программного модуля распознавания образов велась с учетом следующих принципов:

1. Кроссплатформенность модуля.
2. Применение современных технических подходов для детектирования объектов.
3. Высокая скорость детектирования на встраиваемых системах без мощного GPU.

Целевой платформой для применения программного модуля распознавания образов является аппаратно-программная платформа на базе микрокомпьютера Raspberry Pi Model 4B «Мультизадачный робот» [1]. Структурная схема интеграции модуля представлена на рисунке 1. Интерфейс для взаимодействия с модулем также предоставляет возможность автономной работы и тестирования вне системы «Мультизадачный робот».

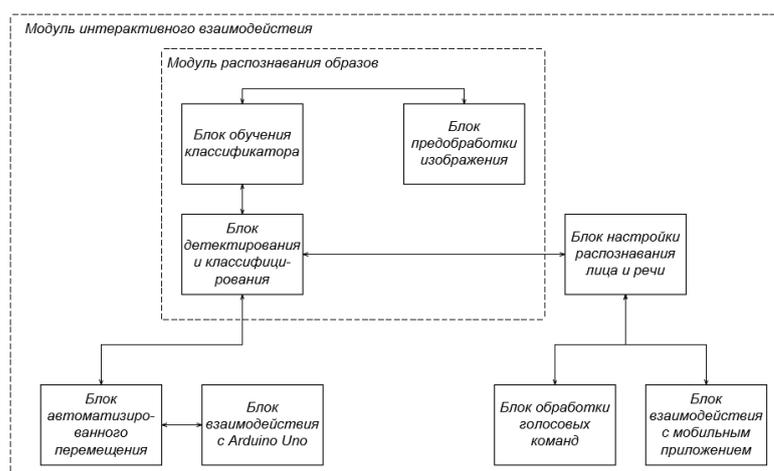


Рисунок 1 – Интеграция структуры программного модуля в аппаратно-программную платформу «Мультизадачный робот»

Программный модуль распознавания образов, как отображено на рисунке 1, состоит из трех блоков. Блок детектирования и классифицирования включает в себя реализацию модели нейронной сети, методы тестирования полученной модели и детектирования объектов входного изображения или видеопотока. Блок обучения классификатора предоставляет методы обучения и

оптимизации нейронной сети. Блок предобработки изображений выделен для процедур подготовки тестовой и обучающей выборки.

1. Архитектура нейронной сети

В основе механизма распознавания образов данного программного модуля лежит нейронная сеть YOLOv5 [2]. В отличие от иных передовых подходов на основе сверточных нейронных сетей, таких как Fast R-CNN и SSD, YOLO предоставляет наиболее эффективную сеть по соотношению скорости работы и точности детектирования, что подтверждается испытаниями модели на датасете MS COCO 2017 [3].

Для обеспечения высокой производительности детектирования на микрокомпьютерах предлагается облегченный вариант сети YOLOv5, backbone часть которой будет заменена на архитектуру сети ShuffleNet V2 [4]. Оригинальная архитектура нейронной сети YOLOv5 представлена на рисунке 2.

Из рисунка 2 можно увидеть, что оригинальная модель использует CSPDarknet в качестве backbone сети. Модели CSP, среди которых и CSPDarknet, основаны на DenseNet [5]. Сеть DenseNet содержит в себе компактно соединенные блоки (dense), соединяющие между собой слои нейронной сети, что позволяет уменьшить проблему исчезающего градиента, поддержать распространение функций и уменьшить количество сетевых параметров [6].

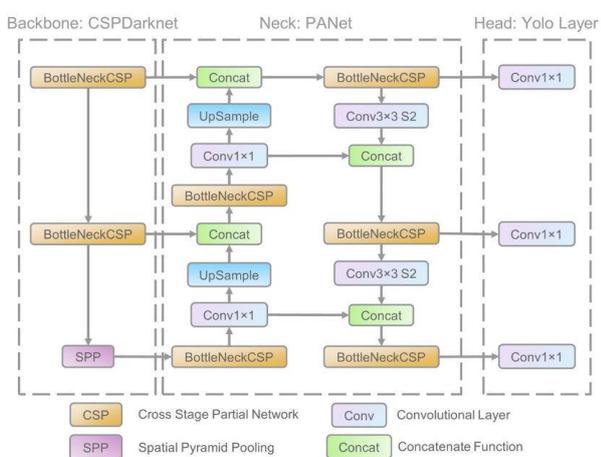


Рисунок 2 – Архитектура оригинального YOLOv5 [7]

ShuffleNet V2 — эффективная CNN для мобильных устройств и встраиваемых систем. Она заимствует сетевую архитектуру быстрого доступа, аналогичную DenseNet. Но в отличие от DenseNet, в ShuffleNet V2 поэлементное сложение заменено конкатенацией, аналогично изменению PANet в YOLOv5 [5]. Использование ShuffleNetV2 в представленной работе позволяет уменьшить среднее время детектирования образов на изображении, при этом точность детектирования изменяется не более чем на 5 процентов по результатам тестирования, что отображено в таблице 1.

2. Сжатие нейронной сети

Для уменьшения количества параметров сети и снижения требований к вычислительной мощности оборудования в данном исследовании используется метод итеративной обрезки каналов и слоев, последовательность которого продемонстрирована на рисунке 3.



Рисунок 3 – Последовательность операций

В соответствии с рисунком 3 процесс сжатия нейронной сети состоит из следующих шагов:

1. Разреженное обучение. Разреженность слоев сети для обрезки каналов обеспечивается применением L1 регуляризации к коэффициентам масштабирования [8]. Степень разреженности сети после обучения напрямую влияет на результат обрезки каналов и слоев.

2. Обрезка каналов.
3. Обрезка слоев.
4. Тонкая подстройка весовых коэффициентов.
5. Квантование.

В процессе обрезки каналов оценивается коэффициент γ заданного слоя и производится удаление весов, которые меньше порогового значения. При обрезке слоев для каждого слоя сортируется среднее значение γ , для сокращения выбирается слой с наименьшей величиной данного коэффициента.

Итерации обучения и обрезки каналов и слоев производятся до достижения максимально возможного сжатия модели при сохранении исходной точности. В результате данных процедур размер модели был успешно сжат до 45% от исходной величины, точность осталась практически неизменной, что отображено в таблице 1.

После сжатия модели было применено статическое квантование при помощи средств фреймворка PyTorch [9]. Для последующей оценки результатов производилось квантование в Float16 и Int8, полученные данные приведены в таблице 1.

3. Результаты тестирования на MS COCO

В результате данной работы были получены обученные модели, способные детектировать до 80 классов, представленных в датасете MS COCO 2017 [10]. Для распознавания дополнительных категорий объектов возможно применение трансферного обучения на предварительно обученной модели.

Как показано в таблице 1, значение mAP_{0.5} сети YOLOv5-Shuffle-P&T после процедур сжатия уменьшилось менее чем на 5% по сравнению с исходной сетью YOLOv5-Shuffle. В свою очередь точность распознавания квантованной модели YOLOv5-Shuffle-q16 остается на уровне 87,1% от уровня YOLOv5s на датасете MS COCO 2017. При этом количество параметров было уменьшено на 78%. Время детектирования стало на 56,8% меньше, чем у YOLOv4-Tiny, что соответствует требованиям для выполнения детектирования в реальном времени.

Таблица 1 – Результаты тестирования моделей на MS COCO 2017 на устройстве с Intel i7-8550U

Model	Params	File size	mAP@0.5	mAP@0.5:0.95	Inference time	GFlops
YOLOv5l	46.5M	91.4	67.3	49.0	320ms	109.1
YOLOv5s	7.2M	14.5	56.8	37.4	131ms	16.5
YOLOv4-tiny	5.9M	23.1	42.0	22.0	125ms	6.9
YOLOv5-Shuffle	5.39M	10.9M	51.6	35.7	84ms	5.9
YOLOv5-Shuffle-P&T	3.6M	6.4	50.1	32.5	65ms	3.4
YOLOv5-Shuffle-q16	-	6.4	49.1	32.0	54ms	2.6
YOLOv5-Shuffle-q8	-	3.2	47.4	27.6	37ms	1.7

В будущих исследованиях эффективность алгоритмов квантования и итеративной обрезки каналов и слоев будет продолжать изучаться для уменьшения влияния на точность обнаружения. Результаты данной работы позволят в дальнейшем применять разработанный технологический подход в системах мониторинга помещений, для персональных роботов-ассистентов для личного пользования, в сферах образования и слуг.

Список использованных источников:

1. Мультизадачный робот с функцией слежения за объектом / Ковбаса Г. А. [и др.] // Компьютерные системы и сети : сборник тезисов докладов 56-й научной конференции аспирантов, магистрантов и студентов, Минск, апрель-май 2020 года / Белорусский государственный университет информатики и радиоэлектроники. - Минск : БГУИР, 2020. - С. 20-21.
2. YOLOv5. [Электронный ресурс]. – Режим доступа: <https://github.com/ultralytics/yolov5>. – Дата доступа: 28.12.2021.
3. A review: Comparison of performance metrics of pretrained models for object detection using the TensorFlow framework / Sánchez Hernández, Sergio & Romero, H & Morales, A. // IOP Conference Series: Materials Science and Engineering. 844. 012024, June 2020. DOI: 10.1088/1757-899X/844/1/012024.
4. ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design / Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng and Jian Sun, // arXiv:1807.11164v1 [cs.CV] 30 Jul 2018. DOI: 10.48550/ARXIV.1807.11164

5. YOLOv5 New Version - Improvements And Evaluation. [Электронный ресурс]. – Режим доступа: <https://blog.roboflow.com/yolov5-improvements-and-evaluation/>. – Дата доступа: 08.02.2022.
6. Review: DenseNet — Dense Convolutional Network (Image Classification). [Электронный ресурс]. – Режим доступа: <https://towardsdatascience.com/review-densenet-image-classification-b6631a8ef803>. – Дата доступа: 02.01.2022.
7. A Forest Fire Detection System Based on Ensemble Learning / Xu, Renjie & Lin, Haifeng & Lu, Kangjie & Cao, Lin & Liu, Yunfei // Forests. 12. 217, February 2021. DOI:10.3390/f12020217.
8. Регуляризация L1 и L2: определение и применение в разных сферах. [Электронный ресурс]. – Режим доступа: https://codernet.ru/articles/drugoe/regulyarizacziya_l1_i_l2_opredelenie_i_primenenie_v_raznyix_sferax/. – Дата доступа: 06.01.2022.
9. PyTorch. Quantization. [Электронный ресурс]. – Режим доступа: <https://pytorch.org/docs/stable/quantization.html>. – Дата доступа: 08.02.2022.
10. COCO – Common Objects in Context. [Электронный ресурс]. – Режим доступа: <https://cocodataset.org/>. – Дата доступа: 15.02.2022.

UDC 004.93

SOFTWARE MODULE FOR PATTERN RECOGNITION FOR MICROCOMPUTERS BASED ON NEURAL NETWORKS OF YOLOV5 FAMILY

Kovbasa G.A.¹

Belarusian State University of Informatics and Radioelectronics¹, Minsk, Republic of Belarus

Azarov E.S. – Doctor of Technical Sciences

Annotation. The problem of pattern recognition has now acquired a large scale in connection with the development of car autopilots, personal assistants, virtual and augmented reality technologies. However, existing technologies are not efficient enough for fast training of models and detection of images of many objects in the conditions of obtaining a sample in real time. This paper presents a compact software module based on a combination of the YOLOv5 network and ShuffleNet V2, the size and number of parameters of which have been optimized for subsequent work on embedded systems. The program interface of the module is designed with the possibility of porting to different platforms.

Keywords. Pattern recognition, YOLOv5, neural network optimization, transfer learning.

**СЕКЦИЯ «ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ»**

АЛГОРИТМ ПОИСКА ЗАИМСТВОВАНИЙ В МУЗЫКАЛЬНЫХ ПРОИЗВЕДЕНИЯХ

Петровец В.Н., Раловец А.А., Деменковец Д.В.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Мельник Н.И. – ст. преподаватель

В работе рассматривается алгоритм поиска заимствований и сравнения музыкальных произведений. Приводится описание алгоритма.

Новинки в музыкальном мире распространяются с большой скоростью. Современный слушатель все чаще улавливает в новых музыкальных произведениях старые и давно полюбившиеся мелодии. И поэтому проблема плагиата в музыке является серьезной и актуальной.

Целью работы является разработка алгоритма для быстрого анализа музыкальной композиции и проверки ее на заимствования из других музыкальных композиций. Ключевой идеей является то, что среди всех характеристик звука, при разных искажающих факторах неизменной остается частотно-амплитудная характеристика.

Суть алгоритма состоит в последовательности действий, представленной на рисунке 1.

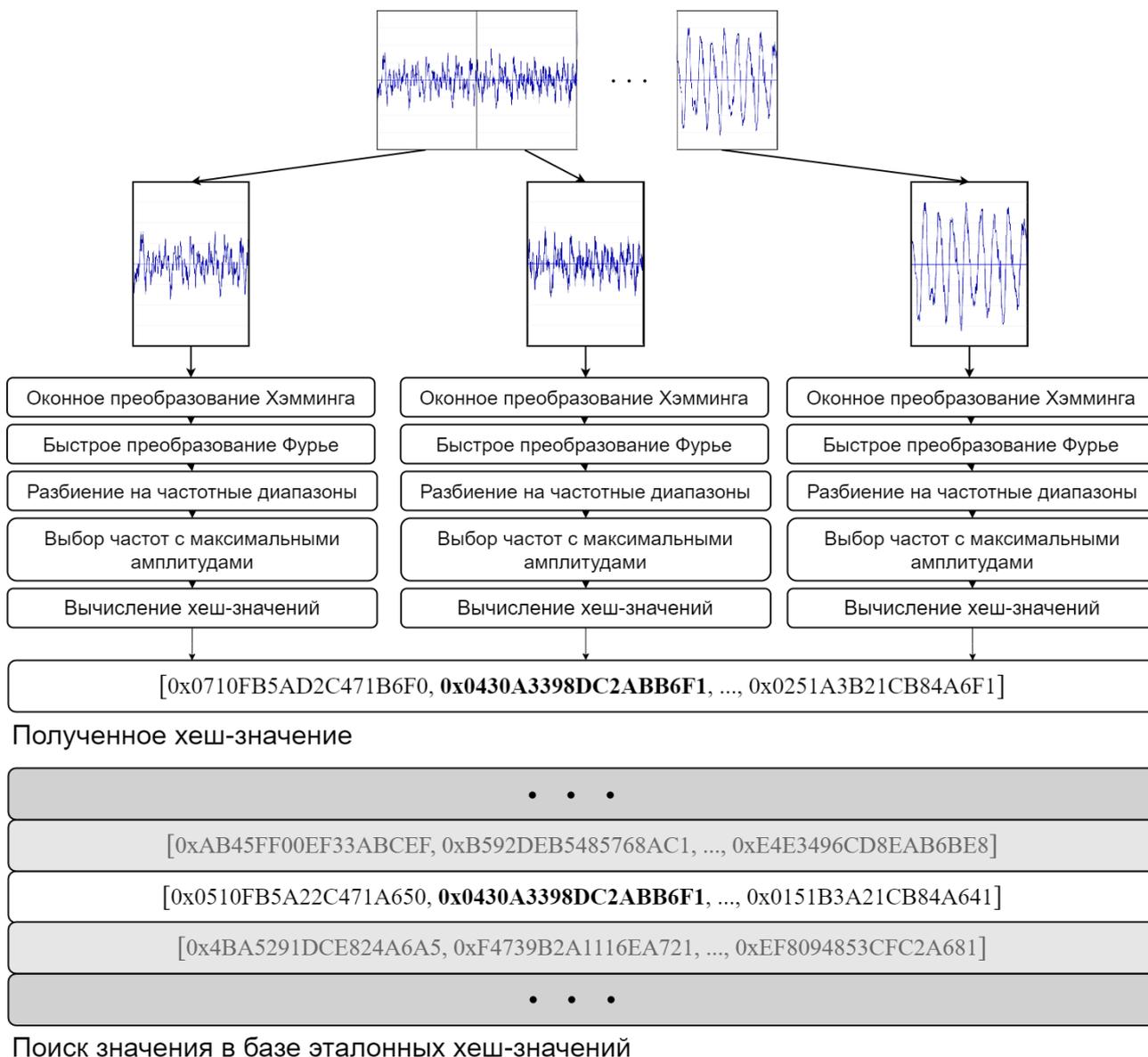


Рисунок 1 – Последовательность действий алгоритма анализа на заимствования

В начале происходит преобразование аналогового звукового сигнала в цифровой вид с частотой дискретизации 44100 Гц и квантованием 16 бит [1]. Результатом является файл формата Waveform Audio File Format [2].

Далее выполняется деление на отрезки (40 мс), для каждого из которых применяется оконное преобразование Хэмминга (коэффициент наложения 50%, размер 2048 байт). Затем выполняется быстрое преобразование Фурье (2048 значений) [3]. Входными данными для БПФ является массив комплексных чисел, действительная часть которых – образцы оцифрованного сигнала, принадлежащие одному окну; мнимая часть – нуль. После заполнения исходной комплексной матрицы, содержащей информацию об одном окне, выполняется БПФ-анализ для этого окна.

Результат БПФ-анализа всех окон – набор частотных характеристик для каждого окна. Частотная характеристика окна представлена на рисунке 2.

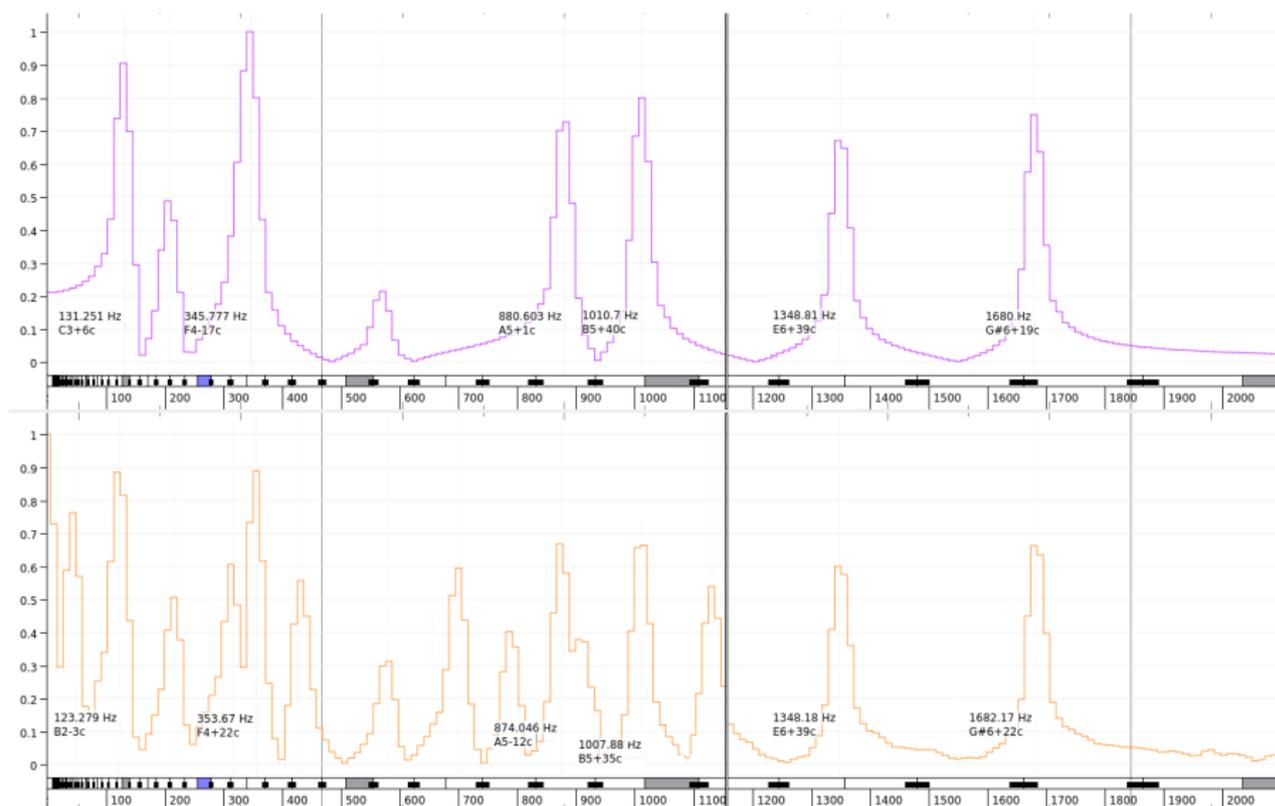


Рисунок 2 – Частотные спектры эталонного и записанного сигналов (сверху – эталонный, снизу – записанный)

После этого из спектра выбираются частоты с максимальными амплитудами в predetermined диапазонах частот. Эмпирическим путем были выбраны 6 диапазонов частот: 100-399 Гц, 400-699 Гц, 700-999 Гц, 1000-1299 Гц, 1300-1599 Гц, 1600-1900 Гц. Комбинация этих частот формирует хеш-значения для каждого окна. Хеш-функция соединяет частоты в виде последовательности из 9 байтов. Итоговая совокупность таких последовательностей представляет собой цифровую сигнатуру музыкального произведения. Для поиска заимствований сравниваются цифровые подписи песен. Чем больше строгих совпадений хеш-значений – тем больше вероятность, что данное произведение – искомое.

При реализации алгоритма особое внимание уделяется определению корректного количества и размера шага частотных диапазонов. При большом числе диапазонов и маленьком размере шага музыкальные дорожки должны быть практически идентичны для получения значимых результатов. В случае же малого количества диапазонов и большого размера шага, даже совсем не похожие дорожки могут выдавать одинаковые сигнатуры.

В результате выполнения работы был реализован алгоритм выделения амплитудно-частотных характеристик музыкальных композиций, выделены оптимальные числовые характеристики. Разработанный алгоритм также применим в других задачах, например распознавание музыки, поиск похожих по жанру композиций, определение схожести голоса.

Список использованных источников:

1. Ричард Лайонс. Цифровая обработка сигналов: Второе издание. Пер. с англ. / Ричард Лайонс – М.: ООО "Бином-пресс", 2006 г. – 656 с.

2. WAVE PCM soundfile format [Электронный ресурс]. – Режим доступа: <http://soundfile.sapp.org/doc/WaveFormat/> – Дата доступа: 01.04.2022.

3. Интегральные – EqWorld [Электронный ресурс]. – Режим доступа: <http://eqworld.ipmnet.ru/ru/auxiliary/aux-inttrans.htm> – Дата доступа: 01.04.2022.

ПОДХОД ДЛЯ ОБРАБОТКИ И СРАВНЕНИЯ ПОЛИГАРМОНИЧЕСКИХ АКУСТИЧЕСКИХ СИГНАЛОВ

Шпаковский А.П., Деменковец Д.В.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Леванцевич В.А. – м.т.н., старший преподаватель

Данная работа является описанием подхода к цифровой обработке полигармонических сигналов в частотной области с описанием функционирования отдельных специфических алгоритмов сравнения сигналов и идентификации потенциального источника.

В самых различных областях человеческой деятельности часто возникает необходимость в анализе окружающих звуковых сигналов с целью определения их природы, вычислении характеристик объектов, которых их порождают, изучении поведения сигналов в процессе их взаимодействия и определении функциональных свойств частей механизмов на основании имеющихся «эталонов» звучания. Все эти задачи, а также широкое множество других задач, связанных с обработкой и сравнением полигармонических [1] сигналов, можно решить с применением универсального набора математических подходов, описанных ниже.

До начала анализа сигнала необходимо сделать выбор области, в которой будет проводиться исследование. Классическими вариантами являются временная и частотная области [2].

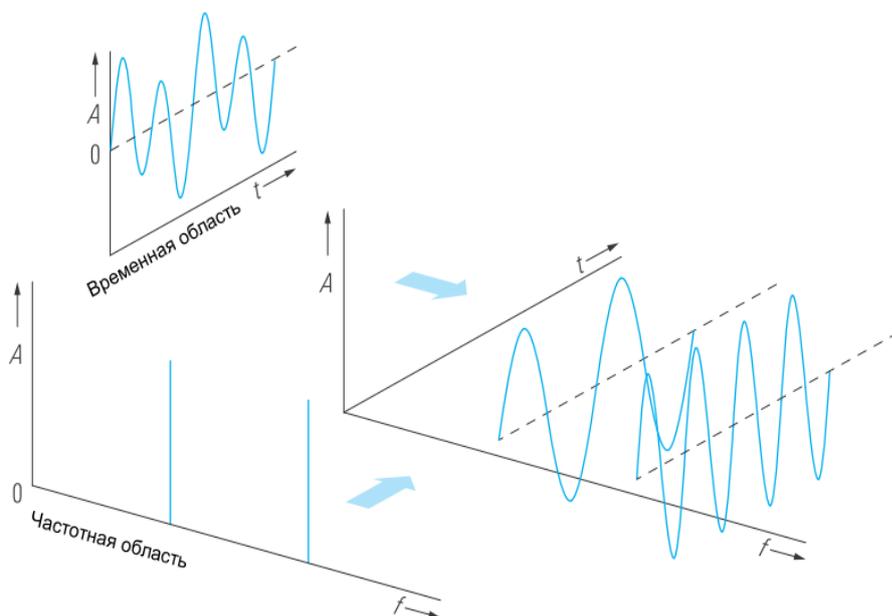


Рисунок 1 – Сигналы, исследуемые во временной и частотной областях

Эти два представления связаны друг с другом операцией преобразования Фурье, поэтому каждый сигнал во временной области имеет характерный частотный спектр и между этими областями имеется четкая связь [3].

Сигнал во временной области показывает изменение состояния системы во времени, в то время как частотная характеристика позволяет понять, на какие частоты или диапазоны частот приходятся отдельные составляющие сигнала. В таком виде сигнал гораздо меньше зависит от характеристик записывающего устройства, а для анализа достаточно сравнительно небольшой

последовательности значений, поэтому наиболее целесообразно исследовать сигнал именно в частотной области.

После выбора области представления обрабатываемого сигнала определяется его формат для последующего применения преобразования Фурье (в данном случае, быстрого преобразования Фурье – БПФ) и перевода исходного сигнала в частотную область [4]. При этом необходимо придерживаться необходимых условий для применения БПФ (ограничение по длине последовательности и свойство периодичности). Непериодические сигналы, как правило, тем или иным способом приводятся к периодическим. Для выполнения этих ограничений предлагается группировка значений сигнала по $2^{12} = 4096$ значений. Выбор этого количества значений удовлетворяет требованию к ширине окна БПФ, а последовательность соответствующей длины является достаточно короткой, чтобы её можно было считать условно периодической. После группировки, выполняется последовательный перевод каждой части исследуемого сигнала в частотную область. После этого преобразования происходит переход к анализу исследуемого сигнала.

При решении задачи распознавания сигнала ключевой задачей является уменьшение времени выполнения алгоритма, потому что при решении задачи напрямую необходима обработка больших массивов данных. Сложность заключается в поиске отрезков, максимально схожих с отрезками сигнала, переданного для распознавания.

В качестве решения данной задачи предлагается разработать алгоритм уникальной идентификации каждого окна. При этом данные, необходимые для поиска максимально схожего сигнала из ранее записанных, представляются структурой, изображенной на рисунке 2.

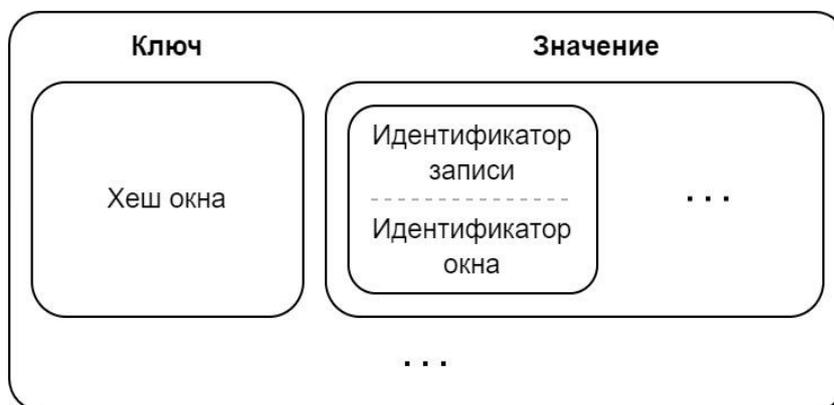


Рисунок 2 – Общий вид структуры хранения данных о сигналах

Данная структура представляет собой набор пар ключ-значение. Ключом является хеш [5] отдельного окна, вычисление которого описывается следующей последовательностью действий:

1. В спектре окна выбирается промежуток частот (обычно от 0 до 100 Гц). Он может изменяться в зависимости от решаемой задачи.

2. Промежутки частот получают условные целочисленные индексы, начиная с нуля и шагом v единицу. Например, при выборе шага частот в 20 Гц, получим диапазон частот 0-19 Гц с индексом 0, 20-39 Гц с индексом 1 и т.д.

3. Далее производится анализ выбранного отрезка. Последовательно выполняется обход всех его значений для поиска максимальных значений амплитуд с соответствующими частотами, которые приходятся на промежутки с каждым индексом. Эта пара сохраняется в промежуточной структуре.

4. Полученные частоты максимальных амплитуд по каждому из промежутков переводятся в строковое представление в виде десятичных чисел с последующей конкатенацией [6]. Результирующая строка образует хеш анализируемого окна.

Поле «значение» структуры на рисунке 2 состоит из списка записей, в каждой из которых идентификатор записи – это идентификатор исходного сигнала, которому соответствует номер входной последовательности. Идентификатор окна – его порядковый номер в исходном сигнале.

После формирования описанной структуры выполняется идентификации сигнала. Она состоит из следующих этапов:

1. По значению хеша сигнала, подлежащего распознаванию, извлекается список «Значение» из структуры, представленной на рисунке 2.

2. Для каждого значения идентификатора формируется дополнительная структура «идентификатор окна» – «количество попаданий».

3. Далее идентификатор акустического сигнала, которому соответствует максимальное число суммарных попаданий, принимается как наиболее точно соответствующий распознаваемому сигналу.

Описанный подход позволяет идентифицировать акустические сигналы и использовать его для косвенного определения характеристик порождающих их источников.

Список использованных источников:

1. Классификация математических моделей сигналов [Электронный ресурс]. – Электронные данные. – Режим доступа: <http://bourabai.ru/signals/ts0108.htm>

2. Связь между временной и частотной областями представления сигналов [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://radioprogram.ru/post/1002>

3. Овсянников, В.А. Методы формирования и цифровой обработки сигналов: учебно - метод. пособие: в 2 ч. Ч. 2: Дискретное преобразование Фурье, фильтрация и модуляция / В. А. Овсянников. – Минск: БГУИР, 2010. – 136 с.

4. Быстрое преобразование Фурье [Электронный ресурс]. – Электронные данные. – Режим доступа: <http://www.ets.ifmo.ru/denisov/dsp/lec12.htm>

5. Хеширование – что это и зачем [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://www.kaspersky.ru/blog/the-wonders-of-hashing/3633/>

6. Конкатенация – толковый словарь [Электронный ресурс]. – Электронные данные. – Режим доступа: https://dic.academic.ru/dic.nsf/fin_enc/24101

ОСОБЕННОСТИ ПЕРЕНОСА МУЗЫКАЛЬНОГО ОБРАЗОВАНИЯ В ОНЛАЙН ФОРМАТ

Юшкевич А.О., Фадеева Е.Е.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Парамонов А.И. – канд. техн. наук, доцент

В работе рассматриваются возможные подходы к переносу музыкального образования в онлайн формат обучения. При этом основное внимание уделяется взаимодействию учащихся и преподавателей. Также предложены технические решения, которые позволят обойти ограничения, связанные с различными типами устройств у пользователей.

Онлайн форма обучения становится все более популярна и востребована. Обществом неоднократно обсуждались преимущества и недостатки такого формата. Однако независимо от различных мнений переход образовательных процессов от очного формата к онлайн формату (дистанционному либо смешанному) неминуемо происходит. Этому дополнительно поспособствовали обстоятельства последних лет, связанные с пандемией и ограничениями в перемещениях обучаемых. В связи с переходом обучения в онлайн возникает потребность в программном обеспечении, которое позволит организовать занятия на необходимом уровне качества. Тенденция перехода в онлайн коснулась различных форм проведения занятий и различных предметных областей. И далеко не каждая сфера образования может использовать известные стандартные системы электронного образования [1]. Одной из таких нетривиальных областей является музыкальная сфера.

Обзор проблематики позволил выявить специфику подачи материалы на уроках при обучении музыкальным инструментам и определить функциональные требования к программному обеспечению для организации таких уроков онлайн. Для моделирования системы и выявления специфики в разработке подобного программного средства выбран популярный музыкальный инструмент – гитара.

Одной из проблем существующих систем дистанционного образования является поддержка сугубо процесса обучения и практически полным отсутствием мотивационных аспектов [2]. Для повышения увлеченности учащихся инструменты онлайн обучения должны активно использовать принципы геймификации. Например, предлагается реализовать таблицы лидеров, по которым учащиеся смогут сравнивать себя с однокурсниками, а также внедрить систему накопления «опыта» для оценивания ранга студента на основе изученных им навыков.

Сами уроки предлагается организовать в виде наборов слайдов с пояснениями техники игры заданной композиции на определенном инструменте. И поскольку одну и ту же композицию можно играть на разных музыкальных инструментах, то у одной композиции может быть несколько уроков, и, соответственно, для каждого инструмента будет свой набор слайдов. Слайды – это интерактивные наборы комбинаций поясняющего текста, изображений и аудио файлов. Содержание слайда будет зависеть от навыка, которому он обучает. В частности, для гитары предполагаются следующие навыки: базовые аккорды, баррэ аккорды, переходы аккордов и последовательности аккордов. Для каждого из этих навыков ученику будут доступны пояснительные тексты, изображения с правильной постановкой руки и аудио файлы соответствующего навыка.

Важной функцией приложения будет возможность прямого взаимодействия учащихся и учителей. У преподавателей будет возможность создать несколько классов, распределить своих учеников по этим классам и отслеживать как индивидуальную статистику учащегося, так и в сравнении со всей группой. Одним из способов взаимодействия является возможность студентом поделиться записью своего видео во время прохождения урока. Ученик записывает либо свое исполнение определенного навыка, например аккордовой последовательности, либо всей песни целиком. После отправки такой записи учитель сможет просмотреть ее и оставить комментарий. При этом у учителя есть возможность оставить комментарий не только в текстовом формате, но и записать видео с пояснениями. Такая форма взаимодействия дает возможность взаимодействовать с системой ученику и учителю асинхронно и не привязываться по времени друг к другу (что особенно удобно при нахождении их в разных часовых поясах).

Предлагаемое решение позволит нивелировать недостатки стандартных систем дистанционного обучения и вносит разнообразие в процесс обучения, что способствует лучшему усвоению материала. Со стороны преподавателя открываются новые возможности взаимодействия со студентами и облегчается процесс передачи знаний учащимся. В итоге

повышается уровень музыкального образования с обеих сторон, что ведет к улучшению качества образования в целом.

Список использованных источников:

1. Парамонов, А. И. Проблемы дистанционного образования и их прикладные решения в образовательных технологиях / Парамонов А. И. // Высшее техническое образование : проблемы и пути развития : материалы X Международной научно-методической конференции, Минск, 26 ноября 2020 года / Минск : БГУИР, 2020. – С. 182–187.
2. Sheldon, Lee. *The Multiplayer Classroom: Designing Coursework as a Game. Second Edition* / CRRP Press, 2020. – 338 p.

ВИЗУАЛИЗАЦИЯ БАЗОВЫХ ОПЕРАЦИЙ ЦИФРОВОЙ ОБРАБОТКИ СИГНАЛОВ С ИСПОЛЬЗОВАНИЕМ ТЕХНОЛОГИЙ PYTHON

Морозов Ю.В., Деменковец Д.В.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Мельник Н.И. – ст. преподаватель

В данной работе было рассмотрено решение для реализации программного средства для более наглядной демонстрации базовых операций цифровой обработки сигналов. Описывается программное средство, разработанное с использованием технологий Python, реализующее визуализацию базовых операций цифровой обработки сигналов. Также предложена архитектура построения реализуемого программного средства.

Цифровая обработка сигналов (ЦОС) стала одной из новейших и самых мощных технологий, которая внедрилась и закрепилась в такие области науки и техники, как: коммуникация, радиолокация, медицина, метеорология, цифровое аудио- и телевизионное вещание и т.д. В настоящее время ЦОС относится к числу базовых знаний, которые необходимы ученым и инженерам всех отраслей без исключений.

Изучение базовых операций ЦОС не требует глубоких и всесторонних знаний по математике, однако предмет считается сложным для понимания. Основной причина сложности понимания является в способе подачи материала. В связи с этим, было принято решение реализовать программное средство, которое будет наглядным образом демонстрировать то, как выполняются базовые операции цифровой обработки сигналов. При этом делается упор на удобство использования данного средства.

В программном средстве реализованы такие базовые операции ЦОС, как: генерация гармонического сигнала, генерация полигармонического сигнала, преобразование Фурье с визуализацией частотного спектра, генерация полигармонического сигнала на основе отдельно взятых гармоник, которые были получены при помощи преобразования Фурье, корреляция одномерных сигналов, корреляция двумерных сигналов (изображений).

Реализованная в программном средстве генерация гармонического сигнала представлена на рисунке 1. Для генерации присутствует возможность ввода амплитуды, частоты и начальной фазы сигнала.

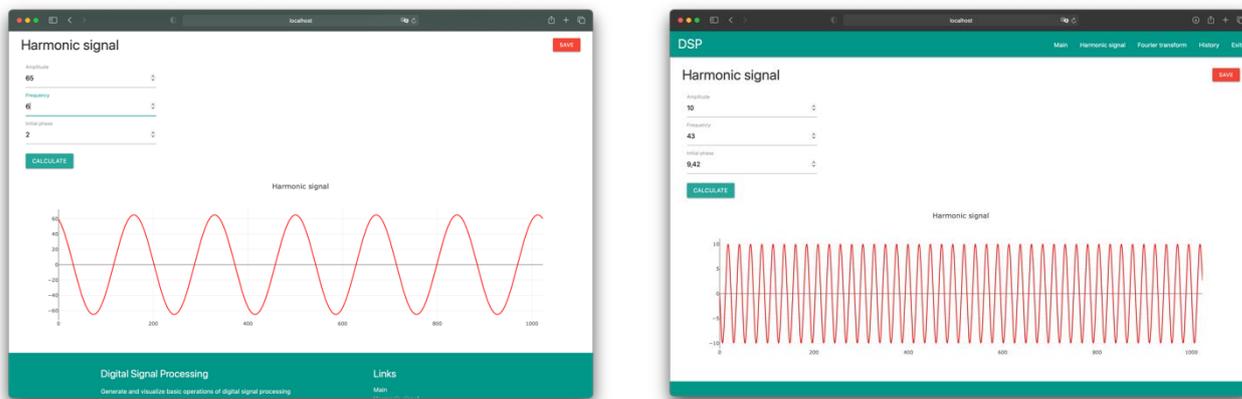


Рисунок 1 – Визуализация генерации гармонических сигналов

Визуализация генерации полигармонического сигнала с выводом разложенных гармоник и преобразованием Фурье отображена на рисунке 2. Для генерации требуется вводить через запятую амплитуды и частоты для каждой гармоники. Также присутствует возможность выбора простых гармоник, которые будут участвовать в формировании полигармонического сигнала.

Корреляция двух одномерных сигналов представлена на рисунке 3. Предусмотрена возможность выбора участка корреляции при помощи перемещения ползунка под результирующим сигналом.

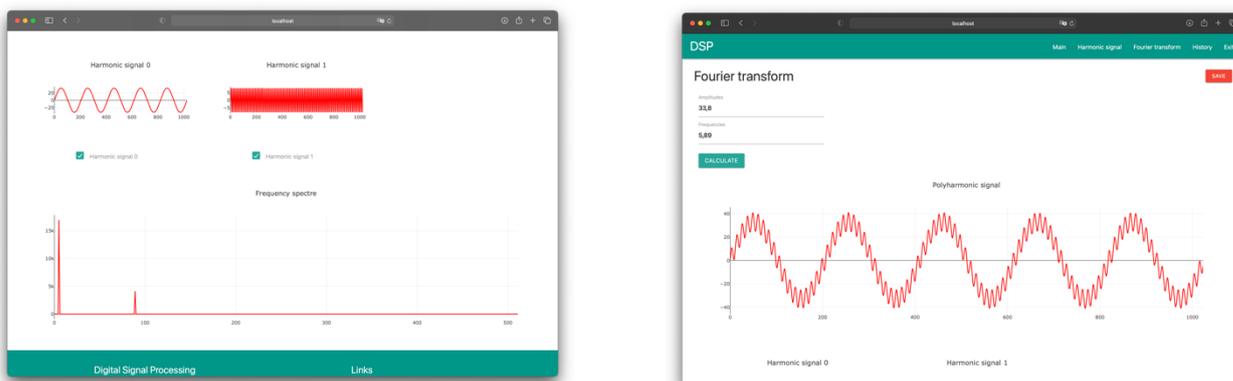


Рисунок 2 – Визуализация генерации полигармонического сигнала с преобразованием Фурье

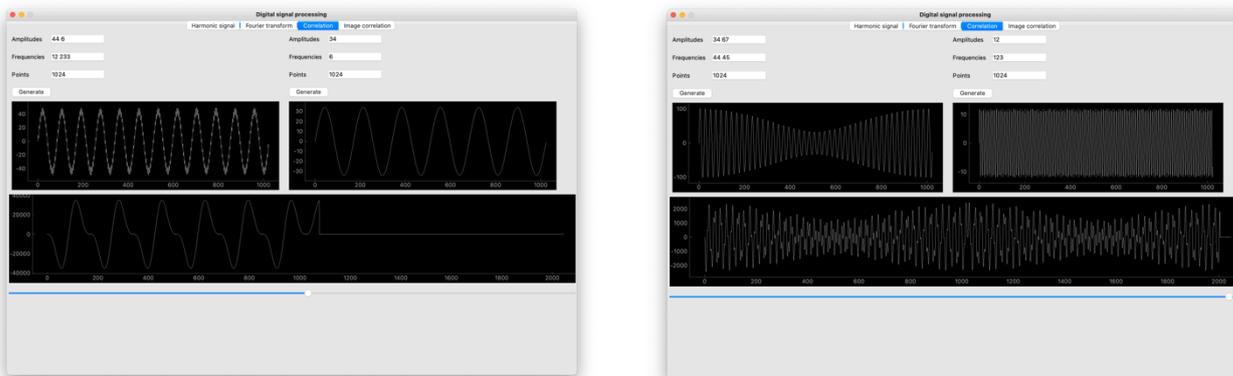


Рисунок 3 – Визуализация корреляции двух полигармонических сигналов

Корреляция двух двумерных сигналов (изображений) представлена на рисунке 4. В качестве результата корреляции представлены двумерный сигнал (результат корреляции), с помощью которого на исходном изображении выделена область с наибольшим корреляционным совпадением. Также предусмотрена возможность построения и просмотра 3D поверхности результата корреляции, на котором четко отображено место с наибольшим совпадением.

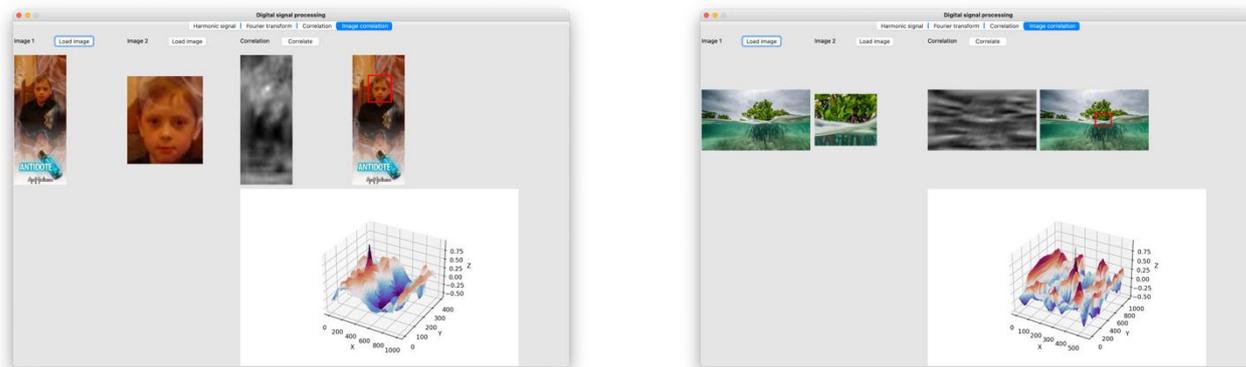


Рисунок 4 – Визуализация корреляции двух двумерных сигналов (изображений)

В результате работы были продемонстрированы базовые операции цифровой обработки сигналов с использованием современных технологий. Данное программное средство можно использовать в качестве обучающего материала по предмету «Цифровая обработка сигналов».

Список использованных источников:

1. *Digital Signal Processing: Principles, Algorithms and Applications: United States Edition* / John G. Proakis: Prentice Hall International; Subsequent edition, 1996. – 1016 p.
2. *Digital Image Processing, Global Edition* / Rafael C. Gonzalez, Richard E. Woods: Pearson; 4th edition, 2018. – 1024 p.
3. *Лабораторный практикум по курсу "Цифровая обработка сигналов" для студентов специальности «Программное обеспечение информационных технологий»/И.Г. Алексеев, П.Ю. Бранцевич, Е.С. Костюк – Мн.: БГУИР, 2006. – 26 с.*

УДК 519.711.3:629.5

ПРИМЕНЕНИЕ МАТЕМАТИЧЕСКОЙ МОДЕЛИ PID-РЕГУЛЯТОРА ДЛЯ СИМУЛЯЦИИ ДВИЖЕНИЯ СУДОМОДЕЛИ

Шульга К.В., студент гр.851003, Деменковец Д.В., ст. преподаватель
Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Деменковец Д.В. – ст. преподаватель

Аннотация. В работе рассматривается применение PID-регулятора для реализации автоматического управления судомоделью. Описывается принцип работы регулятора и его основные составляющие. Представлена схема автоматического управления судомоделью на базе PID-регулятора. Описывается программное средство, моделирующее движение судомодели на основе математической модели PID-регулятора.

Ключевые слова. ПИД, ПИД-регулятор, регулятор, автопилот, судомодель.

В последние годы все больше внимания уделяется автономным транспортным средствам: наземным, морским и воздушным. Примерами этого являются автомобили и суда с автопилотами, различные беспилотные аппараты и эта тенденция продолжается по мере совершенствования технологий и применений в различных сферах. Автономность снижает многие категории риска и по определению исключает человеческий фактор, подверженный ошибкам [1]. Исходя из вышеперечисленного было принято решение спроектировать и реализовать математическую модель автоматического управления для модели судна.

Автоматическое управление судомоделями применяется для решения широкого спектра задач: охрана и патрулирование водоемов на территории заповедников, определение рельефа дна водоема, любительская рыбалка, исследование миграции рыб, поиск скоплений рыбы с помощью эхолота. Все эти задачи сводятся к необходимости построения маршрута движения судомодели и следования ему вне зависимости от воздействия внешних факторов: воздействие ветра, подводные течения, волны и другие. Общая схема автоматического управления судомоделью представлена на рисунке 1.

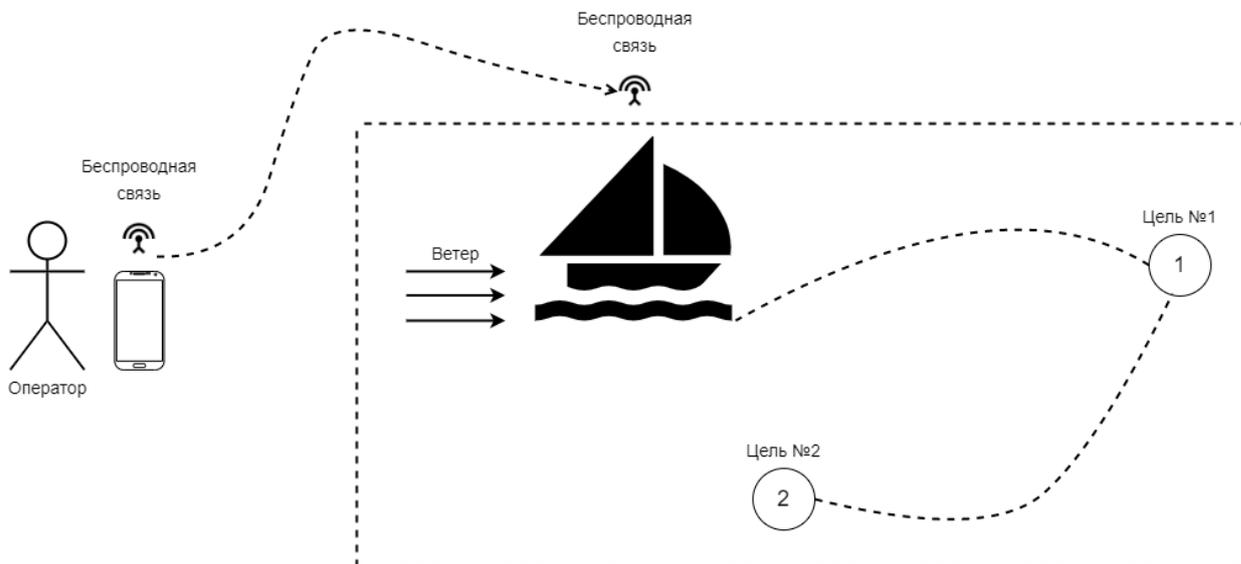


Рисунок 1 – Общая схема автоматического управления судомоделью

Для того чтобы судомодель достигла конечной точки маршрута необходимо реализовать автоматическое управление и корректировку движения модели судна. Для решения этой задачи было использовано устройство – регулятор. Регулятор воздействует на движение модели с помощью управляющих сигналов, которые влияют на скорость движения и угол поворота руля (изменяются в диапазоне от -1 до 1) [2]. Для расчета управляющих сигналов на вход регулятора подаются данные об расстоянии до точки маршрута (цель) и отклонение от курса в градусах.

Чтобы получить вышеупомянутые данные используется GPS-приемник и электронный компас. Схема работы регулятора представлена на рисунке 2.

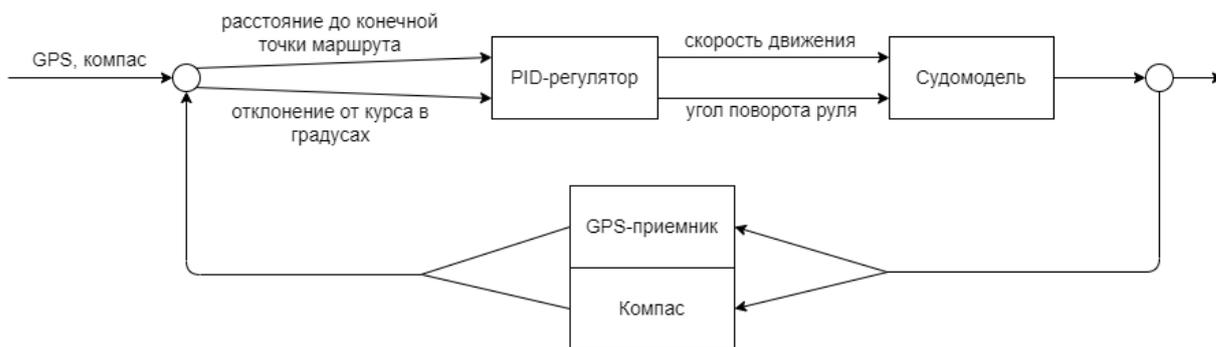


Рисунок 2 – Схема работы регулятора

Одним из способов получения точных и качественных управляющих сигналов воздействия является PID-регулятор. Общая схема цикла обратной связи PID-регулятора представлена на рисунке 3.

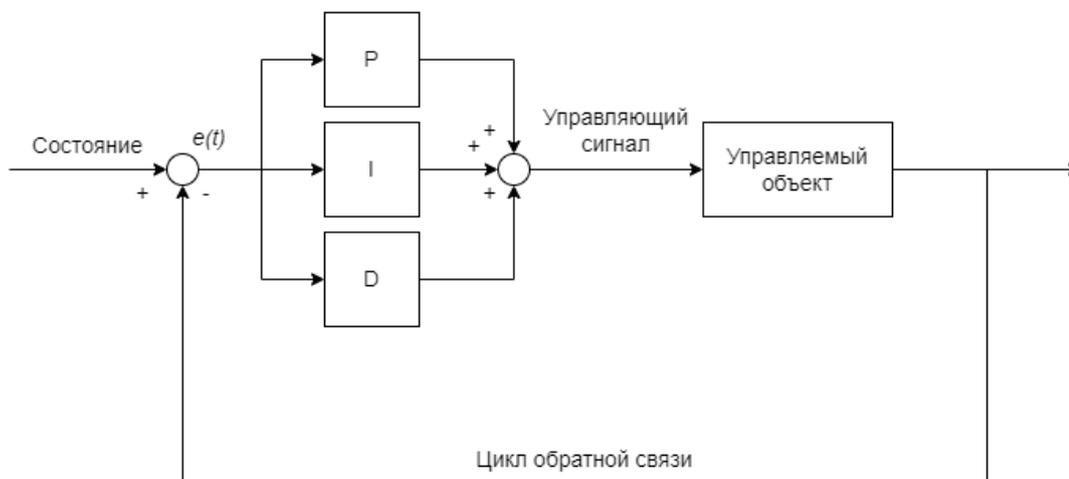


Рисунок 3 – Общая схема цикла обратной связи PID-регулятора

Управляющий сигнал PID-регулятора рассчитывается по формуле [2]:

$$u(t) = P + I + D = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt}, \quad (1)$$

где P – пропорциональная составляющая; I – интегральная составляющая; D – дифференциальная составляющая; K_p – коэффициент усиления пропорциональной составляющей; K_i – коэффициент усиления интегральной составляющей; K_d – коэффициент усиления дифференцирующей составляющей; e – рассогласование (отклонение величины от заданной).

Как видно из формулы, управляющий сигнал является суммой трех составляющих: пропорциональной (1-е слагаемое), интегральной (2-е слагаемое) и дифференциальной (3-е слагаемое).

Пропорциональная составляющая зависит от рассогласования и отвечает за реакцию на мгновенную ошибку регулирования. Однако при использовании только пропорциональной составляющей значение регулируемой величины никогда не стабилизируется на заданном значении. Существует так называемая статическая ошибка, которая равна такому отклонению регулируемой величины, которое обеспечивает выходной сигнал, стабилизирующий выходную величину именно на этом значении [3,4].

Интегральная составляющая содержит в себе накопленную ошибку регулирования, которая является дополнительным источником выходной мощности и позволяет добиться максимальной

скорости достижения цели при отсутствии перерегулирования. Интегральная составляющая позволяет регулятору со временем учесть статическую ошибку, то есть учитывать то, что происходило в прошлом [3,4].

Дифференциальная составляющая зависит от скорости изменения параметра, вызывающей реакцию регулятора на резкое изменение измеряемого параметра, возникшее, например, в результате внешнего возмущающего воздействия [3,4].

Влияние каждой отдельной составляющей очень существенно и в значительной степени зависят от подбора коэффициентов. Если коэффициенты усиления подобраны неверно, то результирующая модель будет неточной или некорректно работать.

На рисунке 4 представлены графики достижения целей в зависимости от коэффициентов усиления [2].

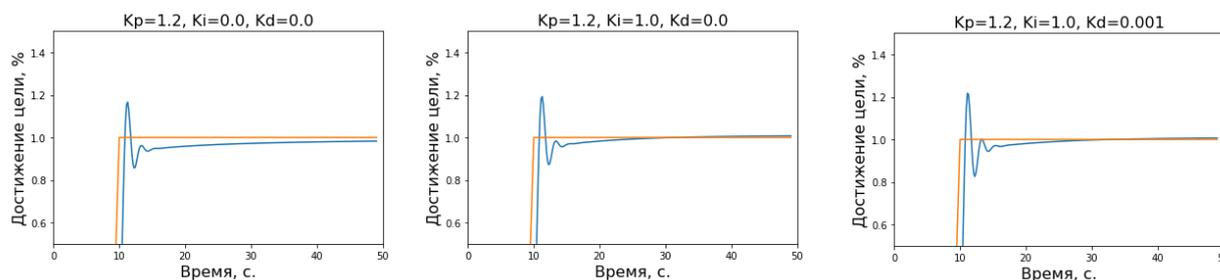


Рисунок 4 – Графики достижения целей в зависимости от коэффициентов усиления

Проанализировав данные графики, можно сделать вывод, что, подобрав правильные коэффициенты можно увеличить точность и тем самым сократить время достижения целей.

Для программной реализации необходимо выполнить дискретизацию стандартной модели PID-регулятора. Дискретная формула имеет следующий вид [4]:

$$Y_i = \frac{1}{X_p} \cdot \left[E_i + \tau_D \cdot \frac{\Delta E_i}{\Delta t_{изм}} + \frac{1}{\tau_u} \sum_{i=0}^n E_i \Delta t_{изм} \right] \cdot 100\% , \quad (2)$$

где X_p – полоса пропорциональности; E_i – рассогласование; T_D – постоянная времени дифференцирования; ΔE_i – разность между двумя соседними измерениями E_i и E_{i-1} ; $\Delta t_{изм}$ – время между двумя соседними измерениями T_i и T_{i-1} ; τ_u – постоянная времени интегрирования; $\sum E_i$ – накопленная в i -й момент времени сумма рассогласований (интегральная сумма).

С целью апробации математической модели PID-регулятора на примере автоматизированного управления судомоделью было разработано программное средство на языке программирования Python со следующими функциями:

1. Возможность установки нескольких точек маршрута (целей).
2. Возможность выполнить автоматизированное управление моделью судна для достижения заданных целей.
3. Возможность перехода в ручной режим управления с визуализацией управляющих сигналов (режим симулятора).
4. Возможность добавить воздействие внешней силы на судомодель (эмуляция течения или воздействие ветра).
5. Отображение показателей судомодели: скорость, угол поворота руля, данные с компаса.

В результате выполненной работы была спроектирована математическая модель PID-регулятора, на основе которой было построено программное средство.

Для реализации вышеописанных функций были спроектированы и запрограммированы следующие модули:

1. `pid.py`, `feedback_loop.py` – содержат логику работы PID-регулятора и цикла обратной связи.
2. `compass.py`, `gps.py` – содержат функции для получения данных с виртуальных компаса и GPS-приемника.
3. `gui_drawer.py`, `boat_drawer.py`, `setpoint_drawer`, `dashboard_drawer.py` – содержат функционал отрисовки и визуализации.

Пример работы программного средства представлен на рисунке 5.

Были получены результаты для пяти наборов данных и построены пять графиков, которые демонстрируют работу регулятора. Каждый график показывает изменение расстояния до цели маршрута с течением времени. Полученные графики показаны на рисунке 6.

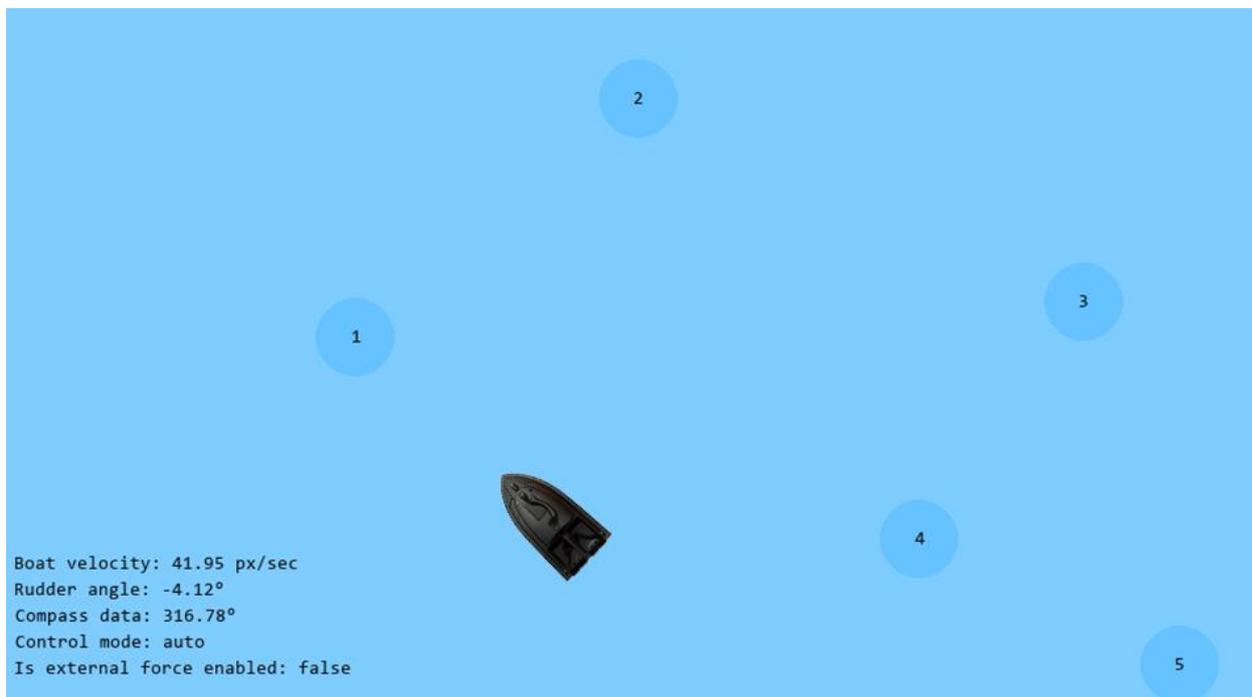


Рисунок 5 – Запущенное программное средство

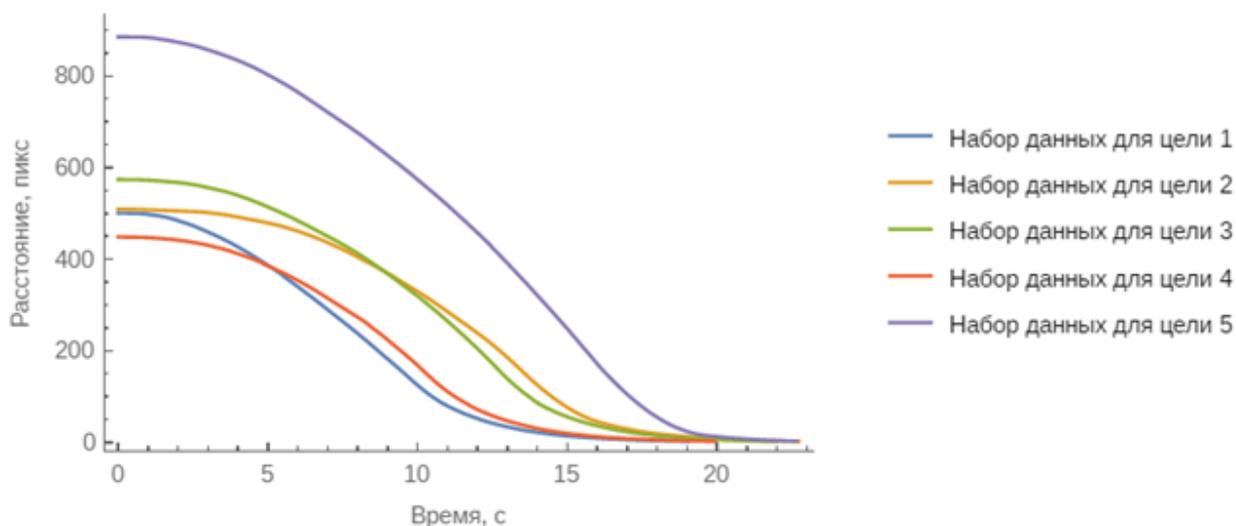


Рисунок 6 – График, демонстрирующий изменение расстояний до цели с течением времени

Тестирования полученного прототипа подтвердило возможность практического применения математической модели на реальной физической. В последующем планируется реализация программно-аппаратного модуля управления судомоделью, где будет применена математическая модель и разработанные алгоритмы для управления физической судомоделью в реальных природных условиях.

Список использованных источников:

1. Visioli, Antonio. *Practical PID control* / Antonio Visioli – London : Springer-Verlag London Limited, 2006. – 310 p.
2. *Handbook of Pi And Pid Controller Tuning Rules* / Aidan O'Dwyer – Imperial College Press, 2006. – 564p.
3. Astrom, Karl J. *PID controllers: theory, design, and tuning* / Karl Johan Astrom, Tore Hagglund – Research Triangle Park, NC : International Society for Measurement and Con, 1995. – 354 p.
4. Шульга, К. В. *Применение PID регулирования для управления судомоделью* / К. В. Шульга, С. И. Билалов, Д. В. Деменковец // *Компьютерные системы и сети: 57-я научная конференция аспирантов, магистрантов и студентов*,

UDC 519.711.3:629.5

APPLICATION OF A MATHEMATICAL MODEL OF THE PID CONTROLLER TO SIMULATE THE MOTION OF A SHIP MODEL

Shulga K.V., Demenkovets D.V.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Demenkovets D.V. – senior lecturer

Annotation. This paper discusses the application of a PID controller for the implementation of automatic control of a ship model. The principle of operation of the controller and its main components are described. The scheme of automatic control of the ship model based on PID controller is presented. The software that simulates the motion of the ship model based on the mathematical model of PID controller is described.

Keywords. PID, PID-controller, controller, autopilot, ship model.

СРАВНИТЕЛЬНЫЙ АНАЛИЗ РАЗЛИЧНЫХ МОДИФИКАЦИЙ КРИПТОСИСТЕМ RSA

Цыбулько К. Д.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Иванюк А. А. – доктор технических наук,
профессор кафедры информатики

Криптосистема RSA с несколькими простыми числами и RSA с возведением в степень — это варианты RSA, которые позволяют улучшить производительность алгоритма дешифрации. В этой работе были представлены и проанализированы оба алгоритма. Так же были смоделированы системы шифрования и экспериментальные результаты ускорения сравнивались с математическими расчётами. В дополнение, были представлены оптимальные вариации, которые позволяют достичь достаточной производительности, при этом не ухудшая безопасность систем.

RSA [1] является наиболее широко используемой криптосистемой с открытым ключом. Используется для защиты веб-трафика, электронной почты и некоторых беспроводных устройств. Поскольку RSA основан на модульной арифметике больших чисел, он может быть медленным в определенных условиях. Например, при работе в сильно загруженной сети сервера, расшифровка RSA значительно снижает количество запросов SSL в секунду, с которыми сервер может справиться. Как правило, производительность RSA улучшается с помощью специального оборудования.

В этой статье будут рассмотрены два простых варианта RSA, предназначенных для ускорения процесса дешифрации. Важно подчеркнуть, что алгоритмы имеют обратную совместимость: система, использующая один из этих алгоритмов, должна иметь возможность взаимодействия с системами, созданными для стандартного RSA.

Первая модификация — это использование нескольких простых чисел при вычислении N . Алгоритм генерации ключа принимает в качестве входных данных параметр безопасности N и дополнительный параметр b . Он генерирует пару ключей RSA следующим образом:

1) Генерация b различных простых чисел p_1, \dots, p_b каждая длиной $\left\lceil \frac{n}{b} \right\rceil$ бит. Установить $N = \prod_{i=1}^b p_i$. Для 1024-битной системы не желательно использование $b > 3$ (т. е. $N = pqr$) по соображениям безопасности, обсуждаемым ниже;

2) Выбирается тот же e , который используется в стандартных открытых ключах RSA. Затем вычисляется $d = e^{-1} \bmod \varphi(N)$. И, как и в классическом алгоритме, необходимо убедиться, что e взаимно простое с $\varphi(N) = \prod_{i=1}^b (p_i - 1)$.

Кодирование производится по тому же алгоритму, как и в классическом RSA. А декодирование выполняется с использованием китайской теоремы об остатках (CRT) [2]. Пусть $r_i = d \bmod (p_i - 1)$. Чтобы расшифровать зашифрованный текст C , сначала вычисляется $M_i = C^{r_i} \bmod p_i$ для каждого i , $1 \leq i \leq b$. После чего уравнения объединяются в систему и применяется CRT, чтобы получить $M = C^d \bmod N$. Использование китайской теоремы об остатках занимает незначительное время по сравнению с возведением в степень.

Сравним работу по расшифровке по вышеописанной схеме с работой, проделанной при расшифровке текста, зашифрованного классическим RSA. Важно, что стандартное дешифрование RSA с использованием CRT требует два полных возведения в степень по модулю $\left\lceil \frac{n}{2} \right\rceil$ битных чисел. В расшифровке RSA с несколькими простыми числами требуется b возведений в степень по модулю $\left\lceil \frac{n}{b} \right\rceil$ битовых чисел. Используя стандартный алгоритм, вычисление $x^d \bmod p$ требует времени $O(\log d \log^2 p)$. Но в случае, когда d порядка p , время выполнения равно $O(\log^3 p)$. Следовательно, асимптотика ускорения RSA с несколькими простыми числами по сравнению со стандартным RSA равна:

$$\frac{2 \cdot \left(\frac{n}{2}\right)^3}{b \cdot \left(\frac{n}{b}\right)^3} = \frac{b^2}{4} \quad (1)$$

Для 1024-битного RSA мы можем использовать не более $b = 3$ (т. е. $N = pqr$), что дает теоретическое ускорение примерно в 2,25 раза по сравнению со стандартным дешифрованием RSA. На практике, эксперименты (реализованные с использованием языка Java) показывают, что среднее ускорение равно 1,73 раза по сравнению со стандартным RSA.

Безопасность данной модификации зависит от сложности факторизации целых чисел, которые образуют N . Для того, чтобы простые множители N было тяжело найти, нужно убедиться, что они не попадают в диапазон метода эллиптических кривых (ECM) [3]. В настоящее время 256-битные простые множители рассматриваются в рамках ECM, поскольку работа по нахождению такие факторы находятся в пределах объема работ, необходимых для факторингового проекта RSA-512. Следовательно, для 1024-битных модулей не следует использовать более трех множителей.

Можно еще больше ускорить расшифровку RSA, используя модуль вида $N = p^{b-1}q$, где p и q простые числа по $\left[\frac{n}{b}\right]$ бит каждый. Когда N имеет длину 1024 бита, мы можем использовать не более $b = 3$, т. е. $N = p^2q$, тогда простые числа p, q имеют длину 341 бит.

Алгоритм генерации ключа принимает в качестве входных данных параметр безопасности N и дополнительный параметр b . Он генерирует ключи RSA следующим образом:

- 1) Генерация 2 различных простых числа p и q , каждое длиной $\left[\frac{n}{b}\right]$ бит. Установить $N = p^{b-1}q$;
- 2) Выбирается тот же e , который используется в стандартных открытых ключах RSA. Затем вычисляется $d = e^{-1} \bmod (p-1)(q-1)$;
- 3) После чего вычисляется пара дополнительных секретных ключей $r_1 = d \bmod (p-1)$ и $r_2 = d \bmod (q-1)$;

Кодирование производится точно так же, как и в классическом RSA. А декодирование выполняется следующим образом:

В первую очередь рассчитываются $M_1 = C^{r_1} \bmod p$ и $M_2 = C^{r_2} \bmod q$, таким образом имеем $M_1^e = C \bmod p$ и $M_2^e = C \bmod q$. При помощи леммы Гензеля [4] можем вычислить такое (M_1') , что $(M_1')^e = C \bmod p^{b-1}$. Стоит так же отметить, что данная операция намного менее затратна, нежели возведение в степень (p^{b-1}). И последним шагом, как и в предыдущей модификации является использование CRT для поиска такого M , что $M = M_1' \bmod p^{b-1}$ и $M = M_2 \bmod q$.

После чего, как и в RSA с несколькими простыми числами, можно получить исходное значение с помощью китайской теоремы об остатках.

Далее сравним время, необходимую для расшифровки с использованием RSA с несколькими степенями, с временем, которое требуется для стандартного RSA. Для RSA с несколькими степенями расшифровка требует двух возведений в степень по модулю $\left[\frac{n}{b}\right]$ битовых чисел и $b-2$ поднятия Гензеля (лемма Гензеля). Поскольку поднятие Гензеля происходит намного быстрее, чем возведение в степень, в расчёт берется только время для двух возведений в степень. Как отмечалось ранее, полное возведение в степень с использованием модульной арифметики, занимают кубическое время. Итак, ускорение данной модификации RSA по сравнению со стандартным RSA составляет примерно:

$$\frac{2 \cdot \left(\frac{n}{2}\right)^3}{2 \cdot \left(\frac{n}{b}\right)^3} = \frac{b^3}{8} \quad (2)$$

Для 1024-битного RSA b снова должно быть не более трех (т. е. $N = p^2q$), давая теоретическое ускорение в 3,38 раза по сравнению со стандартным расшифрованием RSA. Реальный опыт показывает, что на практике мы получаем среднее ускорение в 2,30 раза.

Безопасность RSA с несколькими степенями зависит от сложности разложения целых чисел на множители. Так же, как и в RSA с несколькими простыми числами, нужно убедиться, что простые множители N не совпадают и не попадают в диапазон ECM. Следовательно, для 1024-битных систем можно использовать не более $b = 3$, т. е. $N = p^2q$.

В результате работы были проанализированы и реализованы два варианта ускорения работы дешифрации алгоритмом RSA. Каждый из алгоритмов на практике показал увеличение производительности примерно в 2 раза. Так же, была проанализирована безопасность модификаций, из математическое модели были вынесены значения коэффициентов, которые обеспечивают увеличение производительности, не увеличивая уязвимость шифрования.

Список использованных источников:

1. Thermal behavior of the YAG precursor prepared by sol-gel combustion process / F. Qiu [et al.] // *Ceramics International*, 2005. – P. 663-665.
2. C. Ding, D. Pei and A. Salomaa Chinese Remainder Theorem, Finland 1996
3. R. Silverman and S. Wagstaff Jr. "A Practical Analysis of the Elliptic Curve Factoring Algorithm." *Math. Comp.* 61(203):445-462. Jul. 1993
4. H. Cohen. *A Course in Computational Algebraic Number Theory*, vol 138 of Graduate Texts in Mathematics. Springer-Verlag, 1996, p. 139

ПРОГРАММНОЕ СРЕДСТВО ДЛЯ АНАЛИЗА МАРШЕВЫХ ТЕСТОВ

Петровская В.В., Деменковец Д.В.,

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Ярмолик В.Н. – д.т.н., профессор

В работе рассматриваются основные модели неисправностей запоминающих устройств. Механизмы тестирования ЗУ на основе маршевых тестов и программное средство для анализа маршевых тестов.

Полупроводниковая память является основополагающим и важным сегментом рынка полупроводников. Микросхемы памяти необходимы практически во всех электронных продуктах. Согласно данным World Semiconductor Trade Statistics [1], в 2020 году мировые продажи памяти составили самую большую долю любого вспомогательного продукта – 27% от общего рынка. Результаты исследований показывают, что отказы оперативной памяти (ОЗУ) составляют до 70% от общего числа отказов вычислительных систем. Причиной неисправных состояний является наличие физических или механических дефектов, либо множества подобных дефектов [2].

Поскольку механизм тестирования основан на сравнении логического поведения неисправной памяти с исправной памятью, физические дефекты представляют в виде математических моделей неисправностей [3]. Неисправности классифицируют по количеству участвующих в них ячеек памяти. К одиночным моделям неисправностей относятся константные (stuck-at faults – SAF) и переходные неисправности (transition faults – TF), к неисправностям, в которых участвуют несколько ячеек памяти – неисправности взаимного влияния (coupling faults – CF) и кодочувствительные неисправности (pattern sensitive faults – PSF) [4,5].

В настоящее время для тестирования запоминающих элементов (ЗУ), как массива памяти используются маршевые тесты. Преимущества маршевых тестов заключаются в двух фактах. Во-первых, покрытие известных моделей неисправностей может быть математически доказано, хотя нельзя иметь никакого представления о связи между моделями и физическими дефектами реальных микросхем. Во-вторых, время тестирования для маршевых тестов обычно линейно зависит от размера памяти, что делает их приемлемыми с промышленной точки зрения [6].

Классический маршевый тест состоит из маршевых элементов, которые последовательно применяются к каждой ячейке памяти. В начале маршевого элемента указывается порядок следования адресов, в состав элемента входят операции чтения и записи. Например, маршевый тест MATS+ $\{\uparrow\downarrow(w0), \uparrow(r0, w1), \downarrow(r1, w0)\}$ состоит из трех маршевых элементов [4]. Первый записывает в каждую ячейку памяти ноль, знак $\uparrow\downarrow$ показывает порядок адресов, убывающий или возрастающий. Второй маршевый элемент считывает из ячейки памяти ноль и записывает единицу, адресная последовательность при этом будет возрастающая (\uparrow). Последний элемент в маршевом тесте MATS+ считывает единицу и записывает ноль во все ячейки, порядок адресов убывающий (\downarrow).

С помощью разработанных алгоритмов маршевые тесты преобразуются в неразрушающие и могут быть запущены во время работы системы без потери исходных данных. В неразрушающих версиях маршевых тестов операции чтения/записи нуля и единицы преобразуются в операции чтения/записи прямого или инверсного содержимого ячейки. Маршевый тест TMATS+, преобразованный по методу Николаидиса, состоит из начального $\{\uparrow(rd); \uparrow\downarrow(rd^*)\}$ и базового $\{\uparrow(rd, wd^*); \uparrow\downarrow(rd^*, wd)\}$ тестов. Результаты всех операций чтения сжимаются в сигнатуру. Начальный тест формирует эталонную сигнатуру, а в результате выполнения базового теста получают реальную сигнатуру. Неравенство сигнатур сигнализирует о наличии в памяти дефектов [4,7].

С целью автоматизации проведения и анализа маршевых тестов было разработано программное средство. Приложение позволяет моделировать и тестировать неисправности в массиве ЗУ. На основе полученных с помощью программы результатов тестирования проводится анализ эффективности маршевых тестов.

В разработанной программе реализовано классическое и неразрушающее тестирование по методу Николаидиса, а также новый метод неразрушающего тестирования, основанный на базе четных адресных последовательностей [8]. Программное средство позволяет моделировать одиночные неисправности, неисправности взаимного влияния и кодочувствительные неисправности. В качестве порядка адресов используется счетчиковая, псевдослучайная последовательности и код Грея. Приложение поддерживает многократное тестирование с изменяемыми адресными последовательностями. Изменение последовательностей адресов при многократном тестировании происходит с применением масок.

Вначале работы программного средства пользователь выбирает тип тестирования, размер памяти, вид неисправности, название маршевого теста и адресную последовательность. При необходимости пользователь может записать и выполнить новый маршевый тест. В случае многократного тестирования необходимо выбрать количество тестов и механизм изменения адресной последовательности, для неразрушающего тестирования дополнительно выбирается степень сигнатурного анализатора. После проверки корректности введенных данных инициализируются все необходимые объекты-фабрики, счетчики и списки, формируется конфигурация выбранного типа неисправности. Далее создается массив ячеек ЗУ, маршевый тест и выбираются виды тестируемых неисправностей. В случае неразрушающего тестирования инициализируются сигнатурные анализаторы для исходного и конечного содержимого памяти. В цикле по количеству тестируемых неисправностей выполняется маршевый тест. Если было обнаружено несоответствие ожидаемого и прочитанного значения текущая неисправность добавляется в список обнаруженных. Если тест неразрушающий, то сравниваются эталонные и реальные сигнатуры. Итерации продолжаются, пока не будут проверены все тестируемые неисправности.

Сложные модели неисправностей, в которых участвует две и более ячейки памяти, реализованы в программе с помощью реактивных расширений. На рисунке 1 представлена UML диаграмма, отражающая связи между классами ячейки памяти (Cell) и сложными неисправностями (CouplingFault, NeighborhoodPattern SensitiveFault). Когда содержимое ячейки изменяется, в неисправность, в которой она участвует, приходит уведомление. Если переход состояния ячейки является условием активизации данной неисправности, то обработчик уведомления нарушает содержимое зависимых ячеек. Рассмотренный подход отражает идею поведенческого шаблона проектирования «наблюдатель» (Observer).

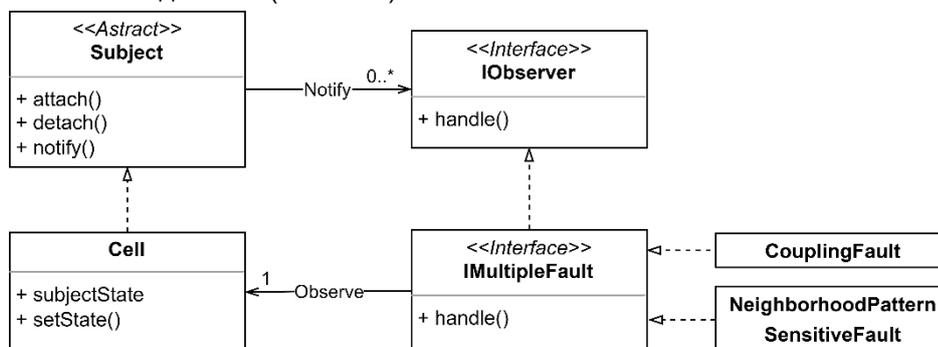


Рисунок 1 – Диаграмма классов, отражающая механизм уведомлений

Эффективность маршевого теста зависит от его покрывающей способности и времени, которое требуется для его выполнения. Покрывающая способность теста – это отношение числа обнаруженных неисправностей к количеству всех возможных неисправностей. В таблице 1 представлена оценка покрытия неисправностей тестами MATS+ и March Y, полученные в результате работы программы.

Таблица 1 – Оценка покрытия неисправностей тестами MATS+ и March Y

Тест	Сложность	Неисправность				
		SAF	TF \uparrow	TF \downarrow	CFin	CFid
MATS+	5N	100%	100%	0%	75%	37,5%
March Y	8N	100%	100%	100%	100%	50%

Полученные результаты аналогичны представленным ранее в работах по тестированию ОЗУ, что свидетельствует о корректности алгоритма программного средства. Наиболее актуальными являются исследования эффективности неразрушающих тестов с четным повторением адресов, в частности, их способности покрывать кодочувствительные неисправности.

Список использованных источников:

1. World Semiconductor Trade Statistics [Электронный ресурс] – Режим доступа: <https://www.wsts.org> – Дата доступа: 21.02.2022.
2. Селедец, В.Н. Программное средство сравнительной оценки маршевых тестов ОЗУ / В.Н. Селедец, В.А. Леванцевич // Компьютерные системы и сети: 55-я юбилейная научная конференция аспирантов, магистрантов и студентов, Минск, 22-26 апреля 2019 г. / Белорусский государственный университет информатики и радиоэлектроники. – Минск, 2019. – С. 152–153.
3. Anumol, A. Detection of Faults in SRAM Using Transient Current Testing / A. Anumol, K. Kumar // IOSR journal of VLSI and Signal Processing. – 2013. – № 2(1). – P. 21–26.
4. Ярмолик, С.В. Маршевые тесты для самотестирования ОЗУ: монография / С.В. Ярмолик, А.П. Занкович, А.А. Иванюк; под общ. ред. В.Н. Ярмолика. – Минск: Изд. центр БГУ, 2009. – 271 с.
5. Ярмолик, В.Н. Анализ и синтез маршевых тестов запоминающих устройств / В.Н. Ярмолик, В.А. Леванцевич, Д.В. Деменковец // Цифровая трансформация – 2021. – № 2. – С. 45–55.

6. Hamdioui, S. *Testing Embedded Memories: A Survey* / S. Hamdioui // *International Doctoral Workshop on Mathematical and Engineering Methods in Computer Science*. – 2013. – P. 32–42.
7. Mrozek, I. *MATS+ transparent memory test for Pattern Sensitive Fault detection* / I. Mrozek, V.N. Yarmolik // *15th International Conference on Mixed Design of Integrated Circuits and Systems / Poznan, Poland – 2008*. – P. 493–498.
8. *Неразрушающее тестирование запоминающих устройств на базе двойных адресных последовательностей* / В.Н. Ярмолик [и др.] // *Доклады БГУИР*. – 2021. – 19(4). – С. 43–51.

ПРОГРАММНО-АППАРАТНОЕ СРЕДСТВО ВИЗУАЛИЗАЦИИ РАБОТЫ АКСЕЛЕРОМЕТРА И ГИРОСКОПА

Авсяник Е.С., Мередов К., Деменковец Д.В.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Леванцевич В.А. – старший преподаватель

В работе описан процесс создания программно-аппаратного средства для визуализации работы гироскопа и акселерометра. Описаны основные принципы работы этих датчиков. Предлагаются варианты применения и улучшения разработанного средства.

Положение в пространстве, направление движения и ускорение – это одни из тех данных, которые являются основными для логической части разнообразных контролируемых устройств. К ним относится не только специальное оборудование самолетов, вертолетов или кораблей, изменение ориентации которого важно для навигации и пилотирования, но и самые обычные устройства, используемые людьми каждый день.

Датчики, определяющие положение и направление в пространстве, такие как акселерометр или гироскоп, выполняющие сбор данных, используются в повседневной бытовой жизни. Широко применяются, например, в смартфонах, фитнес-браслетах, планшетах, ноутбуках, экранах телевизоров или мониторов, а также в технологиях виртуальной и дополненной реальности (VR&AR). В смартфонах и фитнес-браслетах позиционирование используется в качестве шагомера или как опция для управления дополнительными функциями устройства. Позиционирование также возможно с помощью детектора отображаемых элементов мобильного устройства. Наиболее распространенная применяемая функция — это определение текущей ориентации экрана планшета или положение в пространстве электрического самобалансирующегося транспортного средства (гироскутер). Например, в зависимости от передаваемых контроллером параметров формат изображения дисплея настраивается соответствующим образом - либо портретный (широкий), либо альбомный (узкий).

Гироскоп представляет собой устройство, реагирующее на изменение углов ориентации контролируемого тела. В классическом представлении это инерционный предмет, который быстро вращается на подвесах. Как результат, вращающийся предмет всегда будет сохранять свое направление, а по положению подвесов можно определить угол отклонения [1].

В действительности электронные гироскопы построены по другой схеме и устроены значительно сложнее. Достоинствами гироскопа являются: высокая точность в определении углов наклона по всем трем осям XYZ, или другое название – крен, тангаж и рысканье, нулевое направление. От нулевого направления и выполняются измерения. Оно не зависит от притяжения земли и может быть любым, практически не подвержено влиянию внешних факторов или сторонних сил. Среди недостатков гироскопа отмечаются следующие: гироскопом невозможно определять ускорение, сложность конструкции увеличивает конечную стоимость готового устройства, время для определения смены положения значительно выше, чем в случае использования акселерометра.

Акселерометр – это устройство, которое измеряет проекцию кажущегося ускорения, то есть разницы между истинным ускорением объекта и гравитационным ускорением. На простом примере такая система представляет собой некоторую массу, закрепленную на подвесе, обладающим упругостью [2]. Если такую систему повернуть под углом или предать линейное ускорение, то упругий подвес отреагирует на движение под действием массы и отклонится. По этому отклонению определяется ускорение. К основным преимуществам акселерометра можно отнести: быстрое определение изменения действия вектора силы, удобное использование в качестве датчика ускорения, конструктивно проще и дешевле гироскопа. К минусам следует отнести следующие факторы: положение относительно земли определяет с большой погрешностью, акселерометр сильно зависит от воздействия гравитации, необходимость использования коэффициента поправки в зависимости от текущего места и воздействия внешних факторов, чувствителен к сторонней вибрации, не может производить измерение угла наклона при ускорении.

Гироскоп и акселерометр похожи по возможностям, тем не менее, отличаются определяемой характеристикой. В первом случае – положение, для второго – направление воздействия силы. Поэтому функциональность их востребована немного в разных жизненных областях.

Задачей проекта стала разработка программного средства для качественной и эффективной визуализации работы датчиков гироскопа и акселерометра, и проверки их работоспособности. Для этого было разработано устройство с этими датчиками и реализована программа-визуализатор.

Для аппаратной части был выбран микроконтроллер STM32, а конкретно модель «STM32F103C8T6». Данный микроконтроллер имеет ряд преимуществ над более простыми, такими как «Arduino Nano». К примеру, STM32 позволяет делать полноценные 32-битные вычисления, что значительно повышает производительность устройства. Также в его составе есть такой компонент, как DMA (dynamic memory access) и NVIC (nested vector interrupt controller), позволяющие более эффективно работать с памятью контроллера и планированием обработки прерываний и значительно повысить скорость обработки [3]. Помимо всего прочего данные микроконтроллеры имеют гибкие возможности установки экономичных режимов работы для уменьшения потребления энергии. Это позволяет увеличить время работы устройства от элементов питания или аккумуляторов.

В данной работе к микроконтроллеру подключается акселерометр и гироскоп. Передача данных осуществляется по интерфейсу I2C. Через заданные промежутки времени датчики отсылают сообщения об изменении положения объекта в пространстве по трем координатам. Микроконтроллер обрабатывает эти значения и отправляет их на SD-карту и на подключенный персональный компьютер по интерфейсу UART. Подключение модуля SD-карты осуществляется через интерфейс SPI - последовательный синхронный стандарт передачи данных в режиме полного дуплекса, предназначенный для обеспечения простого и недорогого высокоскоростного сопряжения микроконтроллеров и периферии. Данные сохраняются на SD-карту в формате удобном для последующего чтения, благодаря чему даже при отсутствии подключения к компьютеру, будет возможность визуализировать сохраненную информацию. Разрабатываемое программное средство визуализации должно принимать обработанные данные с микроконтроллера, отображать их с помощью 3D модели, считывать данные с SD-карты устройства, кодировать и декодировать их.

В результате было получено приложение, интерфейс которого изображен на рисунке 1. Также было произведено тестирование разработанного программно-аппаратного средства визуализации работы гироскопа и акселерометра. 3D-модель представлена с нескольких ракурсов. Пользователь может изменять угол обзора по своему усмотрению. Часть интерфейса относится к настройке виртуального COM-порта и его сопряжению с компьютером.

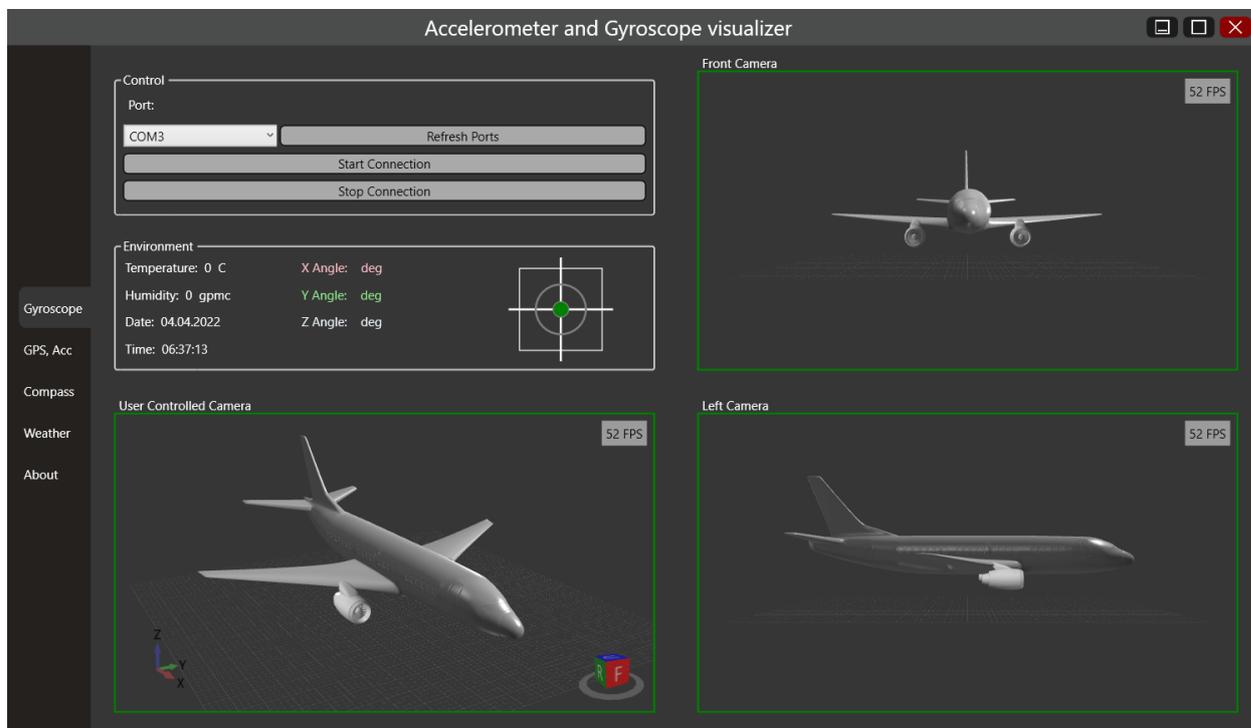


Рисунок 1 – Интерфейс программного средства

Полученное программно-аппаратное средство может записывать данные об изменении положения объекта в пространстве и отображать их на 3D модели. Такое устройство вполне может стать частью более сложной системы, например, устройства отслеживания местоположения, такого как GPS-трекер, устройство по отслеживанию сохранности грузов или качества дорог. При

использовании дополнительно датчик GPS, устройство будет способно отслеживать не только перемещение по карте, но и любые резкие изменения положения объектов в пространстве.

Список использованных источников:

1. Гироскоп. Виды и устройство. [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://electrosam.ru/glavnaja/slabotochnye-seti/oborudovanie/giroskop/>. – Дата доступа: 24.03.2022.
2. Акселерометр. Виды и устройство. [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://electrosam.ru/glavnaja/slabotochnye-seti/oborudovanie/akselometr/>. – Дата доступа: 18.03.2022.
3. Шахнов В.А. Микропроцессоры и микропроцессорные комплекты интегральных микросхем : справочник. В 2 т. / под ред. В. А. Шахнова. – М. : Радио и связь, 1988. – Т. 1. – 368 с.

АЛГОРИТМ ФОРМИРОВАНИЯ КОЛЕБАНИЙ СТОЛА ВИБРОСТЕНДА

Шибут Д. С.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Лапицкая Н.В. – канд. тех. наук

Рассмотрены проблемы, возникающие при метрологической аттестации и поверке вибродатчиков. Показано, что при формировании сигналов возбуждения вибростенда необходимо учитывать его механические и электрические особенности. Описан алгоритм формирования сигналов с учетом коррекции нелинейных искажений вносимых вибростендом. Приведены результаты работы программного средства.

Одним из важнейших параметров, характеризующих техническое состояние машин и агрегатов, являются параметры вибрации. Для контроля вибрации используют различные контрольно-измерительные приборы и первичные вибродатчики, которые периодически подвергают метрологической аттестации или поверке с целью обеспечения достоверности контролируемых параметров [1].

При проведении поверки образцовый и поверяемый датчик особым образом размещаются на столе вибростенда. Для испытаний формируется сигнал возбуждения, который поступает на вибростенд, что приводит к возвратно-поступательному движению его стола. Вибросигналы с образцового и поверяемого датчика поступают в ЭВМ, где оцифровываются и обрабатываются с помощью специализированного программного обеспечения, после чего делается вывод о характеристиках поверяемого датчика [2].

В силу механических и электрических особенностей конструкции вибростенда, его амплитудная и амплитудно-частотная характеристики неравномерны и, как следствие, движение стола вибростенда не будет соответствовать параметрам заданного возбуждающего сигнала [3]. Так если подать на стенд синусоидальный сигнал:

$$u(t) = U \sin(2\pi ft) \quad (1)$$

ускорение на столе вибростенда будет несинусоидальным:

$$a_v(t) = \sum_{i=1}^N A_i \sin(2\pi i f t + \varphi_i) \quad (2)$$

При этом возникает задача измерения и стабилизации первой гармоники. Помимо этого, при проведении поверки ВИП требуется обеспечить воспроизведение на столе вибростенда синусоидального испытательного сигнала с коэффициентом гармоник менее 5%.

Чтобы компенсировать высшие гармоники обусловленные вибростендом, на него необходимо подать не синусоидальный, а сложный сигнал:

$$u_{stim}(t) = u(t) + u_{add}(t) \quad (3)$$

где $u(t)$ – синусоидальный сигнал;

$u_{add}(t)$ – добавочный сигнал.

Добавочный сигнал представляет собой функцию вида:

$$u_{add}(t) = \sum_{i=2}^N U_i \sin(2\pi i f t + \varphi_i) \quad (4)$$

где U_i, φ_i – амплитуда и фаза i -ой гармоники.

Для определения конкретных значений U_i и φ_i предложен следующий алгоритм:

1. Формирование возбуждающего сигнала $u(t) = U_1 \sin(2\pi ft)$.
2. Определение амплитуд гармоник возбуждающих колебаний стола вибростенда $A_i, i = 1..N$.
3. Определение амплитуд гармоник добавочного сигнала:

$$U_i = U_1 \frac{A_i}{A_1 K_i}, K_i = \frac{A_{f,i}}{A_f}, i = 2..N. \quad (5)$$

4. Задание шага по фазе $\Delta\varphi = \frac{\pi}{2}$.
5. Задание начальных фаз высших гармоник $\varphi_{i,0} = 0, i = 2..N$.
6. Определение первой пробной фазы высших гармоник $\varphi_i = \varphi_{i,0} + \Delta\varphi, i = 2..N$.
7. Формирование возбуждающего сигнала:

$$u(t) = U_1 \sin(2\pi ft) + \sum_{i=2}^N U_i \sin(2\pi i ft + \varphi_i) \quad (6)$$

8. Определение амплитуд высших гармоник вибросигнала, отражающего виброускорение колебаний стола вибростенда $A_{1,i}, i = 2..N$.
9. Формирование сигнала возбуждения:

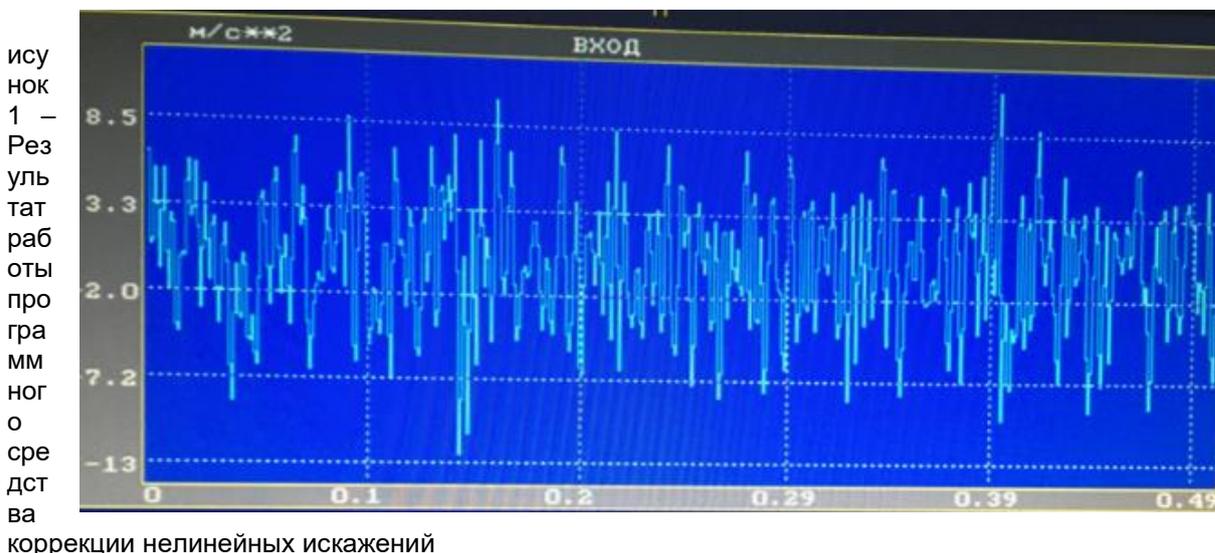
$$u(t) = U_1 \sin(2\pi ft) + \sum_{i=2}^N U_i \sin(2\pi i ft - \varphi_i) \quad (7)$$

10. Определение амплитуд высших гармоник вибросигнала, отражающего виброускорение колебаний стола вибростенда $A_{2,i}, i = 2..N$.
11. Определение приближения начальных фаз высших гармоник.
Если $A_{1,i} > A_{2,i}$, то $\varphi_{i,0} = \varphi_{i,0} - \Delta\varphi$, в противном случае $\varphi_{i,0} = \varphi_{i,0} + \Delta\varphi; i = 2..N$.
12. Определение следующего значения шага по фазе $\Delta\varphi = \frac{\Delta\varphi}{2}$.
13. Если $\Delta\varphi > \frac{\pi}{180}$, то переход к шагу 6.
14. Формирование испытательного сигнала в виде:

$$u(t) = U_1 \sin(2\pi ft) + \sum_{i=2}^N U_i \sin(2\pi i ft + \varphi_{i,0}) \quad (8)$$

Число гармоник, которые участвуют в формировании добавочного сигнала, выбирается порядка $N = 6-10$.

На основе приведенного алгоритма было разработано программное средство формирования сигнала возбуждения с учетом коррекции нелинейных искажений. Результат работы приведен на рисунке 1.



Экспериментальные исследования показали, что предложенный алгоритм позволяет обеспечить формирование гармонических вибрационных сигналов возбуждения вибростенда с коэффициентом гармоник, не превышающим 3%.

Список использованных источников:

1. Неразрушающий контроль. Справочник. Том 7. Книга 2. Вибродиагностика /Ф.Я. Балицкий и др. М.: Машиностроение, 2005. – 485 с.

2. МИ 18-73-88. Методические указания. Виброметры с пьезоэлектрическими и индукционными преобразователями. Методика поверки. Зарегистрированы ВНИИМС 21.06.1988 г.

3. Бранцевич П.Ю., Моделирование вибрационных сигналов с заданными характеристиками/ П.Ю. Бранцевич, Н.В. Лапицкая, В.А. Леванцевич// Информационные технологии и системы 2021(ИТС 2021): материалы Междунар. науч. конф. , Минск, 24 ноября 2021г./ БГУИР. – Минск, 2012. – С.49 – 50

СЕКЦИЯ «ВЫСШАЯ МАТЕМАТИКА»

УДК

ШИФР ЦЕЗАРЯ И ГЕНЕТИЧЕСКИЙ АЛГОРИТМ

Протьюко М.А., студент гр.050502

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Борисенко О.Ф. – канд. физ.-мат. наук, доцент

Аннотация: Используя наличие современных технологий и нововведений возможно внедрить новые подходы к решению таких задач, как поиск динамически развивающейся системы шифрования. Данная работа содержит теоретический материал необходимый для дальнейшего развития данной идеи.

Ключевые слова: генетический алгоритм, подстановочные шифры, моноалфавитные и полиалфавитные шифры, целевая функция, кроссингвер, турнирный отбор

Введение

С распространением глобальной передачи информации необходимо было придумать способы ее сокрытия от нежелательных лиц. Начиная от сдвиговых и подстановочных шифров и заканчивая хэш-функциями и эллиптическими кривыми. Методы сокрытия информации и ее преобразования, а если конкретнее, стеганография и криптография, развивались постепенно с ростом вычислительных возможностей и наших знаний в теории чисел. Казалось бы, давно уже должна была исчезнуть возможность нахождения задачи, которую невозможно вычислить. Но такие задачи все еще существуют. Они относятся к задачам с отсутствием четкого решения или с необходимостью огромнейшего перебора (задача коммивояжера, задача поиска больших простых чисел, задача поиска натуральных делителей и т.д.).

Но все же, для некоторых из этих задач был найден свой особый подход. В последнее время стали распространяться нейронные сети, но в моей работе более интересен их прародитель, а именно, генетические алгоритмы.

Генетические алгоритмы берут многое из области биологии. Используя стратегию, вдохновленную развитием и эволюцией живых существ, решаются самые разнообразные проблемы вычислительной техники: используя этот алгоритм, возможно находить оптимальные пути передвижения из точки А в точку Б, причем первоначальным участникам передвижения не будет ничего известно о местоположении цели («муравьиный» алгоритм, применение – выбор путей передачи пакетов между маршрутизаторами), также возможно использовать генетический алгоритм для нахождения решения функций, где посредством сравнений с желаемых результатом, находятся желаемые решения, ну, и разумеется, генетические алгоритмы используются для моделирования эволюционных процессов и создания искусственного интеллекта в играх, пусть даже и не столь впечатляющего.

Поскольку на современном этапе развития криптографии все очевидные решения уже были использованы, к тому же, теперь недостаточно просто шифровать что-либо каким-то конкретным хорошо показавшим себя способом, теперь необходимо делать это динамически (т.е. ключи и порядок их использования непостоянны), желательнее используя несколько тактик (в протоколах безопасности используются комбинации нескольких шифров, отдельные для ключа, и совершенно другие для сообщения), поэтому необходимо использовать больше изобретательности.

Со всем вышеперечисленным на ум приходит идея: а что может нам дать в этой ситуации генетический алгоритм? Какие результаты, желательнее с возможностью практического применения он может нам дать?

Возможно ли создать систему, имитирующую основные факторы, приведшие к созданию шифров таким образом, чтобы она, используя генетический алгоритм, повторила стратегию, используемую человеком?

В данной работе я попытаюсь ответить на эти вопросы.

Генетический алгоритм. Теория.

Генетические алгоритмы — это адаптивные методы поиска, которые в последнее время используются для решения задач оптимизации. В них используются как аналог механизма генетического наследования, так и аналог естественного отбора. При этом сохраняется биологическая терминология в упрощенном виде и основные понятия линейной алгебры.

Основной идеей генетических алгоритмов является организация «борьбы за существование» и «естественного отбора» среди пробных решений. Поскольку генетические алгоритмы используют биологические аналогии, то и применяющаяся терминология напоминает биологическую. Так, одно пробное решение, записанное в двоичной (или десятичной) форме, называется особью или хромосомой, а набор всех пробных решений – популяцией.

Генетический алгоритм чаще всего состоит из следующих этапов:

- Отбор
- Скрещивание /мутации
- Выбор индивидуума с максимальной приспособленностью
- Поддержание популяции

Логические связи между каждым этапом схематически изображены на рис.1.1

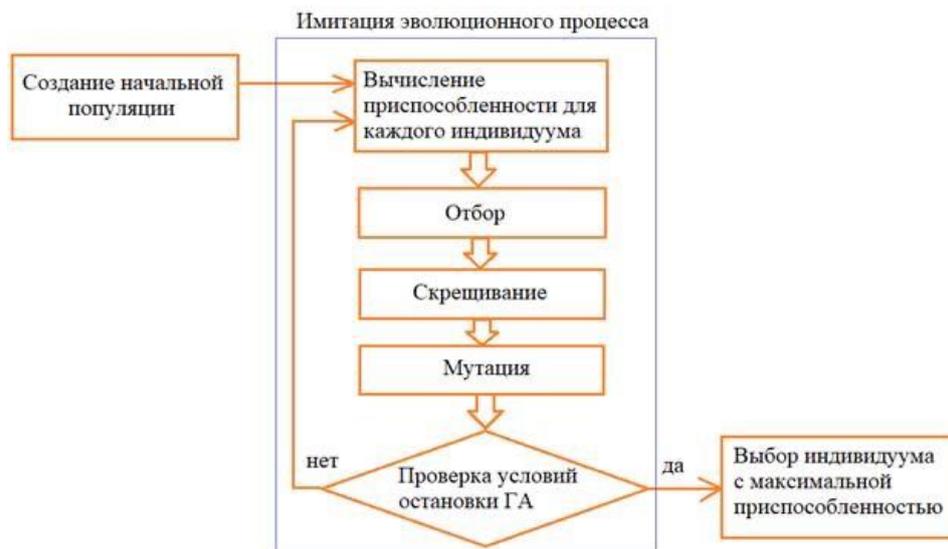


Рисунок 1.1 – Схема генетического алгоритма

Рассмотрим каждый этап подробнее:

Вычисление приспособленности для каждого индивидуума:

Чаще всего вычисляется с помощью целевой функции (fitness function). Чем меньше значение целевой функции, тем более приспособленной является особь, т.е. пробное решение, использовавшееся в качестве аргумента целевой функции.

К примеру, если необходимо найти минимум некой функции, целевая функция будет выбирать ту особь, значение вывода которой, подставленное в аргумент даст минимальное значение.

Есть случаи, когда можно обойтись без целевой функции, если составить условия существования таким образом, чтобы соревнование между особями происходило само собой. Можно использовать ограничение по времени жизни и количеству особей в системе, использовать принцип накопления энергий (кто имеет больше всего запасов, тот способен образовывать лучшее поколение) и т.д.

Скрещивание (кроссинговер):

Процесс размножения (рекомбинация), заключается в обмене участками хромосом между родителями. Например, пусть скрещиваются две хромосомы 111111 и 000000. Определяем случайным образом точку разрыва хромосомы, пусть, это будет 3: 111|111 000|000. Теперь хромосомы обмениваются частями, стоящими после точки разрыва, и образуют двух новых потомков: 111000 и 000111.

Иными словами, геномы двух и более особей меняются какими-либо частями между собой.

Мутации:

Случайные изменения генома особи в одном и более местах. Вероятность их возникновения регулируется самостоятельно или распределяется случайным образом.

Мутации способны улучшить или ухудшить приспособленность особи-потомка. Задача отсеивания последних приходится на вышеописанный первый пункт.

Процесс размножения:

На основе исходной популяции создается новая, с применением вышеописанных стратегий. В случае использования кроссинговера, необходимо определять родителей различными способами (по понятию схожести, наибольшей приспособленности и т. д.).

Существует несколько способов отбора в популяцию, это турнирный отбор и метод рулетки.

При турнирном отборе (tournament selection) из популяции, содержащей N особей, выбираются случайным образом t особей, и лучшая из них особь записывается в промежуточный массив. Эта операция повторяется N раз. Особи в полученном промежуточном массиве затем используются для скрещивания (также случайным образом). Размер группы строк, отбираемых для турнира, часто равен 2. В этом случае говорят о двоичном (парном) турнире. Вообще же t называют численностью турнира. Преимуществом данного способа является то, что он не требует дополнительных вычислений.

В методе рулетки (roulette-wheel selection) особи отбираются с помощью N «запусков» рулетки, где N — размер популяции. Колесо рулетки содержит по одному сектору для каждого члена популяции. Размер i -го сектора пропорционален вероятности попадания в новую популяцию $P(i)$, вычисляемой по формуле:

$$P(i) = \frac{f(i)}{\sum_{i=1}^N f(i)}, \quad 1)$$

где $f(i)$ — пригодность i -й особи. Ожидаемое число копий i -ой хромосомы после оператора рулетки определяются по формуле $N_i = P(i)N$.

При таком отборе члены популяции с более высокой приспособленностью с большей вероятностью будут чаще выбираться, чем особи с низкой приспособленностью.

Другие способы отбора можно получить на основе модификации выше приведенных. Так, например, в рулеточном отборе можно изменить формулу для вероятности попадания особи в новую популяцию.

Простейшие шифры

Прежде чем приступить к рассмотрению принципов работы шифров, необходимо ознакомиться с некоторыми общепринятыми терминами:

Шифром будем называть систему или алгоритм, трансформирующий произвольное сообщение в такую форму, которую не сможет прочитать никто кроме тех, кому это сообщение предназначено.

Ключом будем называть такую символьную последовательность, знания которой позволяют привести зашифрованный текст в его изначальный вид. Причем, чаще всего считается, что именно ключ — гарантия невозможности расшифровки текста, даже если алгоритм шифрования известен.

Под *шифрованием* будем понимать процесс преобразования открытого текста в зашифрованный текст. Соответственно, *дешифрование* — это процесс, обратный шифрованию, преобразующий зашифрованный текст в открытый.

Для дальнейшей работы необходимо будет рассмотреть следующие виды шифрования:

Шифр Цезаря:

Самый старейший из шифров, в нынешних реалиях, увы, не используемый из-за своей простоты, а, следовательно, и с весьма наглядной расшифровкой методом грубой силы (перебором всех значений ключа).

В шифре Цезаря каждая буква в открытом тексте смещается на три позиции так, что буква «А», например, замещается буквой «D». Буква «В» замещается буквой «Е» и так далее. Конец алфавита замыкается на его начало так, что буква «Х» замещается буквой «А», а буква «У» — буквой «В», «Z» — «С». В шифре Цезаря каждая буква смещается на 3 позиции, однако в более широком смысле этот шифр можно рассматривать, как смещение на некое целое число позиций (k), причем число k будет играть роль ключа.

На рисунке 2.1 схематически изображено преобразование, выполняемое шифром Цезаря.

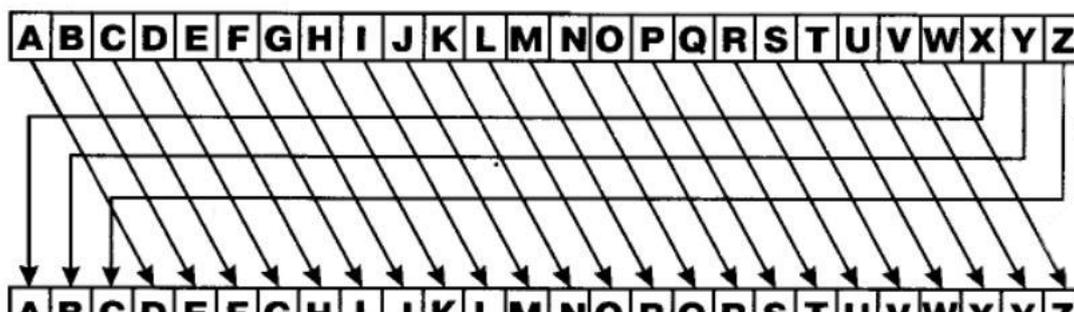


Рисунок 2.1 – Шифр Цезаря, схематическое преобразование

В итоге, получим:
 Ключ: 3
 Открытый текст: Hello
 Зашифрованный текст: Khoor

Сразу становится заметно, что метод частотного анализа также будет весьма полезен в случае необходимости расшифровки. Чем длиннее будет сообщение, тем больше вероятность потери секретности.

Математическое определение шифра Цезаря (сдвигового шифра):

При произвольном ключе k , где $k \in Z_{26}$, где Z_{26} -множество целых чисел, $0 \leq k \leq 25$, и произвольном открытом тексте p в виде кортежа, где $p = (p_1, p_2, p_3, \dots, p_m)$ и $p_i \in Z_{26}$ для $0 \leq i \leq m$, результирующий зашифрованный текст c будет представлен кортежем $c = (c_1, c_2, c_3, \dots, c_m)$ и $c_i \in Z_{26}$ для $0 \leq i \leq m$.

При этом функция шифрования для $E_k(p)$ для сдвигового шифра определяется следующим образом:

$c_i = E_k(p_i) = p_i + k \pmod{26}$ для $0 \leq i \leq m$. А функция дешифрования $D_k(c)$ определяется, как $p_i = D_k(c_i) = c_i - k \pmod{26}$ для $0 \leq i \leq m$.

Еще одна особенность шифра Цезаря, это бессмысленность многократной шифровки. При повторении процесса шифрования, два ключа будут представлять не что иное, как ключ, представляющий сумму смещений первых двух, что не дает никакого практического смысла.

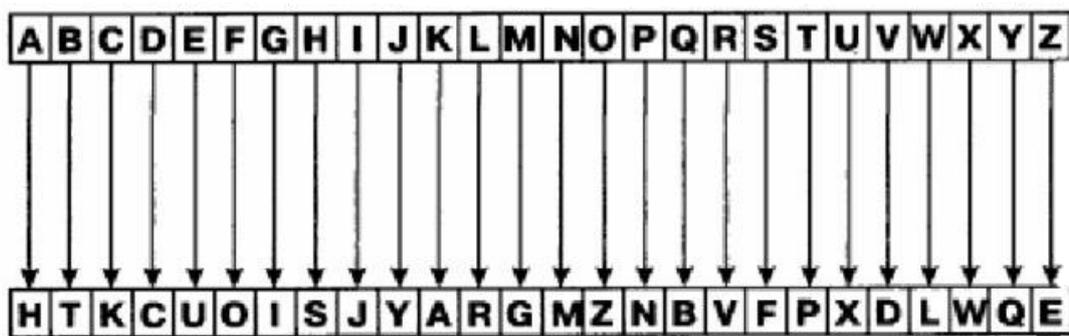
Далее следует ознакомиться с немного более совершенными классическими шифрами, развивающими идею смещения относительно ключа.

Простой подстановочный шифр

В подстановочном шифре каждый символ заменяется заранее определенным символом подстановочного алфавита, что относит его, как и шифр Цезаря, к моноалфавитным подстановочным шифрам. Это означает, что существует однозначное соответствие между символами в открытом тексте и символами в тексте зашифрованном. Такое свойство шифра делает его также уязвимым для атаки, основанной на частотном анализе.

Схематический он изображен на рисунке 2.2.

Рисунок 2.2 – Подстановочный шифр, схематическое преобразование



Математическое определение подстановочного шифра (шифра простой замены)

Пусть K есть множество всех перестановок элементов в Z_{26} $K = \{(k_1, k_2, k_3, \dots, k_{26}): k_i \in Z_{26}\}$. Выберем произвольный ключ k , где $k \in K$. Выберем произвольный текст p , где $p = (p_1, p_2, \dots, p_m)$, где $p_i \in Z_{26}$ для $1 \leq i \leq m$. Пусть результирующий зашифрованный текст c есть: $c = (c_1, c_2, \dots, c_m)$, где $c_i \in Z_{26}$ для $1 \leq i \leq m$. Далее определим функцию шифрования $E_k(p)$, как $c_i = E_k(p_i) = k[p_i]$ для $1 \leq i \leq m$. Функция дешифрования $D_k(c)$ определяется, как $p_i = D_k(c_i) = E_k^{-1}(c_i)$ для $1 \leq i \leq m$.

В итоге, получим:
 Ключ: HTKCUOISJYARGMZNBVFPXDLWQE
 Открытый текст: Hello
 Зашифрованный текст: SURRZ

Иными словами, в подстановочном шифре мы придумываем свой собственный алфавит. Чем больше алфавит языка, который мы используем, тем больше длина ключа. К примеру, в английском алфавите 26 букв, а значит у нас будет 26-значный ключ с $26! = 403\ 291\ 461\ 126\ 605\ 635\ 584\ 000\ 000$ различными значениями. Но, увы, длина ключа ни насколько не повлияла на сложность расшифровки.

Этот шифр крайне уязвим к частотному анализу, поскольку частота букв остается той же.

Шифр Виженера

Ранее известный как «нераскрываемый шифр», он во многом превзошел своих предшественников.

Шифр Виженера представляет собой полиалфавитный подстановочный шифр. Это означает, что для подстановки используются многие алфавиты, благодаря чему частоты символов в зашифрованном тексте не соответствуют частотам символов в тексте открытом. Следовательно, в отличие от моноалфавитных подстановочных шифров наподобие шифра Цезаря, шифр Виженера не поддается простому частотному анализу.

В сущности, шифр Виженера меняет соответствие между открытыми и зашифрованными символами для каждого очередного символа. Он основывается на таблице, вид которой приведен на рисунке 2.3. Каждая строка этой таблицы – не что иное, как шифр Цезаря, сдвинутый на число позиций, соответствующее позиции в строке. Строка А сдвинута на 0 позиций, строка В – на 1 и так далее.

В шифре Виженера такая таблица используется в сочетании с ключевым словом, при помощи которого шифруется текст.

Для шифрования вы повторяете ключ столько раз, сколько необходимо для достижения длины открытого текста, просто записывая символы под символами открытого текста. Затем вы получаете поочередно каждый символ зашифрованного текста, беря столбец, определенный по символу открытого текста, и пересекая его со строкой, определенной по соответствующему символу ключа. Например, первый символ открытого текста G в сочетании с первым символом ключа P, как видно из рисунка 2.6, дадут столбец и строку, пересекающиеся на символе V, каковой и будет первым символом шифрованного текста. Все последующие символы определяются аналогичным образом.

Открытый текст

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Рисунок 2.3 – Шифр Виженера, схематическое преобразование

Ключ: PRO PA GA NDA PROP AGAN DAPR OPA GAND
 Открытый текст: GOD IS ON OUR SIDE LONG LIVE THE KING
 Зашифрованный текст: VFR XS UN BXR HZRT LUNT OIKV HWE QIAJ

Математическое определение подстановочного шифра (шифра простой замены):

Пусть даны произвольный ключ k , где $k = (k_1, k_2, k_3, \dots, k_n)$, где $k_i \in Z_{26}$ для $1 \leq i \leq n$, и произвольный открытый текст p , где $p = (p_1, p_2, \dots, p_m)$, где $p_j \in Z_{26}$ для $1 \leq j \leq m$. Пусть зашифрованный текст s представлен, как $s = (s_1, s_2, \dots, s_m)$, где $s_j \in Z_{26}$ для $1 \leq j \leq m$. Тогда мы определим функцию шифрования E_{k_i} , как $s_j = E_{k_i}(p_j)$, где $E_{k_i}(p) : p_j \rightarrow p_j + k_i \pmod{26}$, а функция дешифрования $D_{k_i}(s)$ определяется, как $p_j = D_{k_i}(s_j)$, где $D_{k_i}(s) : s_j \rightarrow s_j - k_i \pmod{26}$.

Иными словами, шифр Виженера – это повторяющийся шифр Цезаря с разным смещением для каждого элемента в ключе.

Проектирование системы

Основная цель системы – создать динамический меняющийся алгоритм, способный подстраиваться под входные условия (скорость обработки данных, количество ключей и т.д.). Это достижимо, если придерживаться стратегии, используемой в генетических алгоритмах.

Для того, чтобы создать толчок к развитию, необходимо иметь то, что нужно развивать. Необходимо создать геном, способный охарактеризовать индивидуума таким образом, чтобы условия создания открытого и закрытого текстов выполнялись.

Для простоты будем использовать английский алфавит из 26 букв. Учитывая математическое определение шифров, нам необходимы ключи и сдвиг.

Иными словами, если разбить действия, совершаемые алгоритмом гена на простые шаги, получится приблизительно следующее:

- 1) Взять букву изначального текста
- 2) Определиться со сдвигом
- 3) Начать сдвиг

4) Передать полученную букву

Повторять, пока не будет достигнут конец слова.

Мне хотелось бы достичь масштабируемости в своем геноме, который возможно будет образовать, поэтому, прежде чем создавать систему, нужно определиться с ее условиями, при этом помня о необходимости дешифровки, поскольку нам нужен обратимый результат.

У всех шифров есть три ключевых элемента — ключ, алфавит, и операция, которая объединяет их вместе (позволяет им коррелировать), поэтому, включим их в геном, причем так, чтобы при необходимости они могли образовать следующие последовательности:

Шифр Цезаря:

1 ключ, 1 операция, 1 алфавит (порядок следования английских букв)

Шифр подстановок:

26 ключей, 1 операция, 2 алфавита (изначальный и смещенный)

Шифр Виженера:

1 <ключей <26, 1 операция, число алфавитов, совпадающее с числом ключей (каждая позиция ключа дает смещение изначального алфавита на одну позицию (математически: операции на кольцевом множестве Z_{26})

Поскольку в каждом из шифров используется 1 операция, стоит начать с ее непосредственной кодировки.

Во всех этих шифрах можно использовать любую обратимую математическую операцию (сложение, умножение, возведение в степень и т.д.), на данный момент рассмотрим шифр на основе сложения, помня, что при необходимости, операцию можно заменить.

Поскольку алгоритм предполагает возможность расшифровки, возможна кодировка не конкретной операции (прибавить или отнять), а самой необходимости произвести некую операцию, зависящую от внешне заданных условий.

Иными словами, в зависимости от характеристик значение в гене может представлять собой как сложение, так и вычитание в случаях шифрования и дешифрования соответственно.

Допустим, непосредственно операция будет кодироваться 1.

Следующий неизменный элемент — алфавит, допустим, английский (возможно с любым другим языком или последовательностью, организованной известным вам образом). Допустим, мы будем кодировать каждую букву в том же порядке, что и в английском алфавите. Получим последовательность кодов от 2-27.

С такими характеристиками операций гена уже возможно воссоздать шифр Цезаря, который будет кодироваться следующим образом, с учетом того, что считывание гена происходит слева направо и при считывании позиции смещение происходит на 1 цифру вправо:

5 1 5 1 5 1...5 1 5 1 — ген длиной в слово, которое мы можем зашифровать

Алгоритм дешифровки не меняется, стоит только использовать второе значение 1 - обратную операцию.

Полученный шифр подстановки (Виженера):

7 1 4 1 15 1 ...9 1 — основное условие для получения чистого шифра подстановки — цифры между единицами не должны повторяться.

Первый пришедший на ум вариант, как можно заметить, весьма избыточен. Да, мы получили малое количество элементов для кодировки, но учитывая факт развития популяции, вероятность того, что за приемлемое время (максимум 200-300 поколений) в каждой нечетной/четной позиции гена образуется повторяющаяся последовательность от начала до конца длиной в кодируемое слово весьма мала. Для слова длиной в 5 букв нам понадобится ген длиной в 10 цифр, причем различных значений в этих 10 цифрах - 2. Вероятность возникновения такой последовательности очень мала, из чего следует, что мутаций будет необходимо очень и очень много, да и индивидуумов тоже.

И еще один минус: как много попыток уйдет на получение способности индивида обнаруживать конец слова, учитывая то, что его ген имеет фиксированную длину и, скорее всего, индивид будет рождаться не согласованно с изменением длины слова (слово родителей может весьма отличаться от слова, доставшегося детям, что представляет собой проблему — в чем смысл динамической развивающейся системы, если она способна разеваться только в рамках слов какой-либо фиксированной длины?)

Стоит добавить еще одну возможную кодировку — код смещения чтения собственного гена. По умолчанию он — единица, но, если добавить возможность перемещаться по всему геному от начала и до конца при необходимости?

Для того, чтобы это сделать, в любом случае, нужно знать фиксированную длину гена, что уже не позволит попросту устанавливать количество кода в созависимости с словом. Но в данном случае, мы можем и не столкнуться с ситуацией, когда ген должен быть слишком большим, чтобы не быть способным зашифровать слово большой длины.

Допустим, длина нашего генома — 55 символов, тогда, по аналогии с кольцом для алфавита, нам необходимо будет кольцо Z_{55} . Нам понадобятся цифры от 28-55 для кодировки смещения.

Один из примеров для шифра Цезаря:

5 1 27 – выигрышная последовательность, ведь стоит ей появиться, и она закрепится.

Для шифра Цезаря без возможности сдвига по собственному гену мы получили всего две выигрышные вариации, здесь же множество вариантов, и это лишь первые, что приходят на ум. Они самые рациональные, но возможны будут варианты просто с смещением на 4 рабочие позиции гена, а после назад. Разброс и сжатость кодировки позволяют увеличить работоспособность и приспособляемость индивида.

Следует заранее оговорить тот момент, что скорее всего при выполнении алгоритма чтения гена программа будет сама брать последовательно элементы слова и решать кода ей выйти из цикла, но можно пойти и более тяжелым путем, закодировав и эти две операции.

Тогда получим приблизительно следующий набор команд:

- 1 - +/-
- 0 - взять слово
- 2 - выйти из цикла
- 3-28 – алфавит
- 29-56 смещение (для длины генома в 55 символов)

Способность взять элемент слова потенциально может дать индивидууму возможность кодировать какую-то часть информации, но не ее всю, что может быть проблемой.

Теперь, зная приблизительно геном, которого мы можем добиться и его самые желательные вариации следует заняться разработкой непосредственно самого генетического алгоритма.

Первое, с чем следует определиться, это с количеством мутаций. Допустим, для начала они будут образовываться случайным образом в 1-5 позициях.

Я не уверена в целесообразности работы кроссинговера, поскольку одна такая операция может полностью лишиться смысла весь естественный отбор, разве что дав возможность выжить только тем индивидуумам, код которых будет буквально продублирован дважды. Не думаю, что это целесообразно.

Следующий элемент - окружающая среда. Допустим, у нас имеется 10 изначальных индивидуумов в начальной популяции. Они используют общее пространство, но двигаться не способны.

Следующий вопрос — использовать целевую функцию, турнирный отбор, или соревнование?

Если реализовывать соревнование по принципам естественного отбора, получим приблизительно следующее:

Приоритет будет отдаваться индивидуумам способным преобразовывать слово, посредством прибавления количества итераций их жизни (если ген индивида способен зашифровать слово полностью - +20 возможных циклов, если способен дешифровать его - +40 и возможность создать свою абсолютную копию).

Также можно контролировать преобладающий ген в системе давая приоритет по размножению тем индивидуумам, которые пережили всех остальных. То есть, новый индивид способен будет появляться только при наличии свободной позиции.

Также возможно будет влиять на процесс с помощью введения проверки с неким подобием турнира, если индивидуум гарантированно покажет свою негодность (к примеру, если добавить с систему отслеживание поколений, при наличии индивидуума, дающего популяцию, все еще не продвинувшуюся в основной задаче за 50-100 циклов, возможно, стоит от него избавиться).

В итоге, мы должны получить динамическую систему, в которой будет выживать от 10 индивидуумов. Идея для соревнования может быть такой: для того, чтобы выжить, индивидуум должен зашифровать слово таким образом, чтобы через какое-то время только этот индивидуум был способен правильно дешифровать его, следовательно, выжить.

Использовать данную стратегию отбора стоит при наличии системы, стабильно дающей необходимый результат, для того чтобы определить, что созданная система удовлетворяет поставленным требованиям, стоит пройти по нескольким тестам.

Тесты:

- Тест на длинные слова
- тест на простейшие последовательности (только с двумя возможными типами кодировок)
- тест на рабочую последовательность кода (корреляция между его длиной и количеством поколений перед возникновением результата)
- тест при различных начальных условиях (разное время жизни, размер популяции, условия роста, наличия fitness функции)
- тест на мутации (какое их количество оптимально и в зависимости от какой длины кода и количества кодируемых операций)
- тест на добавление возможности выхода
- тест на добавление возможности выбирать элементы слова (возможно, даже непоследовательно)
- тест на кроссинговер

Заключение:

Основная цель этой работы - проектирование алгоритма для создания аналогии с естественным отбором и живыми существами для того, чтобы проверить, какие внешние условия будут влиять на развитие конкретных стратегий и посмотреть, к чему это в дальнейшем приведёт.

В работе делался упор на ту модель, которую я лично считаю наиболее рабочей. Только наличие нескольких практических результатов и данных позволит сказать, является ли она таковой. Для этого стоит пройти по всем тестам, что я описала выше, и посмотреть, что произойдет.

Пока у меня имеется лишь теоретический материал, подтверждаемый формулами цифрами и догадками, но я уверена, что, начав практическое исполнение этой системы я столкнусь с многими занимательными вещами. Поэтому, дальнейшее развитие моего проекта - написание такого генетического алгоритма с помощью программных средств и современного аппаратного обеспечения, причем явно будет недостаточно создать просто рабочий алгоритм, необходима будет возможность его анализа и сбора информации о индивидуумах и их развитии, желательно удобным образом.

Возможно, если этот проект удастся, я смогу увеличить его масштаб для возможности кодирования более современных и криптостойких алгоритмов. Но это в более дальнем будущем.

Список использованных источников:

1. Торстейнсон П., Ганеш Г.А. Криптография и безопасность в технологии .NET: учеб. пособие / П. Торстейнсон, Г. А. Ганеш; пер. с англ. — М.: БИНОМ.: Изд-во Лаборатория знаний, 2015. — 479 с.
2. Панченко Т.В., Генетические алгоритмы: учеб. пособие / под ред. Ю. Ю. Тарасевича.; Изд-во Астраханского ун-та, 2007. — 87 с.
3. Протко М.А., Принципы построения алгоритмов шифрования, V Международная научно-практической конференция студентов, аспирантов, преподавателей, Армавир, 202. — 3 с.

UDK

CAESAR'S CIPHER AND GENETIC ALGORITHM

Protko M.A.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Borisenko O.F. – PhD in Physics and Mathematics

Annotation. Using the availability of modern technologies and innovations, it is possible to introduce new approaches to solving problems such as the search for a dynamically developing encryption system. This work contains the theoretical material necessary for the further development of this idea.

Keywords. Genetic algorithm, substitution ciphers, monoalphabetic and polyalphabetic ciphers, objective function, crossover, tournament selection

МНОГОКРИТЕРИАЛЬНАЯ ОПТИМИЗАЦИЯ ПО ПАРЕТО

Ершов О.В., студент гр.153501, Скворцов А.В., студент гр.153501

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Луцакова И. Н. – канд. физ.-мат. наук

Аннотация. В работе излагаются идеи оптимизации по Парето, линейной свёртки, метода идеальной точки и метода контрольных показателей. Приведены алгоритмы определения принадлежности к множеству границы Парето и определения лучшего варианта по методу идеальной точки. В качестве примера рассмотрена многокритериальная задача выбора игроков в футбольный клуб.

Ключевые слова. Оптимизация по Парето, метод идеальной точки, метод контрольных показателей.

Многокритериальная оптимизация — это раздел математического программирования, а также метод решения задач, которые состоят в поиске наилучшего (оптимального) решения, удовлетворяющего нескольким несводимым друг к другу критериям [1,2,3]. Существует 3 основных типа задач многокритериальной оптимизации. Рассмотрим их на примере задачи с двумя критериями f_1 и f_2 .

- 1) В задаче критерий f_1 приоритетнее, чем критерий f_2 , или критерий f_2 приоритетнее, чем f_1 .
- 2) Задача сводится к минимизации функции вида $\alpha * f_1 + \beta * f_2$, где $\alpha + \beta = 1$. Данный метод носит название линейной свёртки.
- 3) Критерии f_1 и f_2 равнозначны в условиях рассматриваемой задачи.

При равнозначности критериев f_1 и f_2 задача решается оптимизацией по Парето, а также используются метод идеальной точки и метод контрольных показателей. Решению этой задачи мы уделим наибольшее внимание, как самой трудной среди указанных трёх типов задач. Сначала рассмотрим алгоритм оптимизации по Парето на примере двухкритериальной задачи, а потом приведём пример задачи оптимизации для семи критериев на основе нашей первой задачи.

Задача 1: необходимо выбрать лучших нападающих в футбольный клуб по критериям максимальной личной результативности (количество забитых голов) и командной сыгранности (выполненные голевые передачи).

Интерпретируем задачу в математическую запись: из множества игроков $\{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9\}$ выбрать лучших игроков по критериям:

- 1) Наибольшего количества забитых голов: $W_1 = f_1(x_k) \rightarrow \max$
- 2) Выполненных голевых передач: $W_2 = f_2(x_k) \rightarrow \max$

Решением задачи является оптимальное по Парето множество игроков $\{x_1, x_3, x_4, x_6, x_8\}$, доставляющее максимум критериям W_1 и W_2 .

Таблица 1 – Футболисты и их показатели для задачи с двумя критериями

№ футболиста	Фамилия	Голы	Передачи
1	Зинченко	12 шт.	12 шт.
2	Яшин	10 шт.	2 шт.
3	Соловьев	6 шт.	13 шт.
4	Попов	14 шт.	0 шт.
5	Косьмин	0 шт.	12 шт.
6	Уткин	13 шт.	10 шт.
7	Дзюба	12 шт.	6 шт.
8	Брежнев	2 шт.	15 шт.
9	Медведев	2 шт.	5 шт.

Целью оптимизации по Парето является поиск оптимальных (доминирующих) точек. Оптимальными точками являются не улучшаемые значения – точки, лучше которых нельзя подобрать по всем критериям. Это означает, что без уточнения приоритетности критериев, при дальнейшем выборе будут рассматриваться варианты из множества оптимальных точек. Если в

угле, ограниченном лучами, берущими начало в точке и направленными вдоль осей по направлению улучшения полезности, не найдется ни одной точки, то данная точка является оптимальной. В данном примере оптимальными точками являются футболисты с номерами 1, 3, 4, 6, 8 (см. рисунок 1). Указанные точки образуют границу Парето, которая была названа американскими экономистами «Северо-восточная стена». Для любой точки, не входящей в границу Парето, всегда найдется точка на «Северо-восточной стене», превосходящая ее по всем критериям.

Рассмотрим данную оптимизационную задачу.

Алгоритм 1:
Шаг 1: где $0 \leq i < n$ и $x = \{x_0, x_1, \dots, x_{n-1}\}$ - футболистов, голам $x_0 \geq x_1 \geq \dots \geq x_{n-1}$ - футболистов, пасам $y_0 \geq y_1 \geq \dots \geq y_{n-1}$.
Шаг 2: в котором мы будем хранить имена футболистов, прошедших отбор. Обозначим их через G .
Шаг 3:

```
bool flag = true;
for (int k = 0; k < i && flag; k++){
    for (int t = 0; t < j; t++){
        if( (x[k],y[t]) consists P){
            flag = false;
            break;
        }
        else {
            G = (x[i],y[j]);
        }
    }
}
```

алгоритм для решения задачи.
 Пусть $P = \{(x_i, y_j)\}$, $0 \leq j < n$. Пусть это множество упорядоченное по $\dots \geq x_{n-1}$. Пусть $y = \dots \geq y_{n-1}$.
 Создадим массив, в

Пояснение:

На шаге 1 футболисты упорядочиваются по количеству забитых голов и по количеству передач (в общем случае упорядочивание происходит по всем критериям). Это нужно для того, чтобы знать, какие футболисты сделали больше голов/пасов и число этих футболистов. Каждого футболиста зададим парой значений (координат), где индексы i и j обозначают число футболистов, которые забили больше голов и пасов, соответственно, чем данный футболист. Если в процессе сортировки встречаются два одинаковых значения, то проверяем их на Парето и худший элемент удаляем из множества. Когда каждому футболисту присвоена пара значений (x_i, y_i) , можем переходить к шагу 2. В результате работы циклов из шага 3 мы получим множество G , в котором хранятся отобранные футболисты. Идея метода состоит в том, чтобы сперва упорядочить исходное множество по критериям, а потом сравнивать футболистов, которые забили больше голов с теми, кто дал больше пасов, и если эти множества пересекутся, то проверяемый игрок не входит во множество границы Парето.

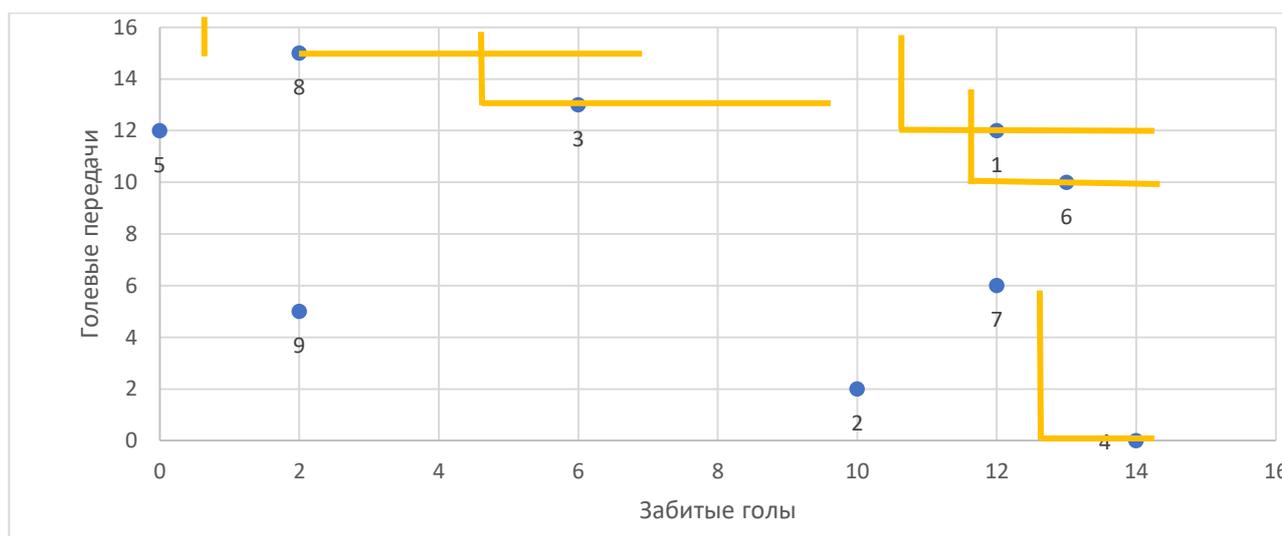


Рисунок 1 - Диаграмма отобранных футболистов

На примере двухкритериальной задачи можно рассмотреть ещё один интересный и удобный метод выделения практически полного множества Парето, а именно метод линейной свёртки. Данный метод «сворачивает» критерии нашей задачи в один мультикритерий, по которому и будем делать выборку игроков. Напомним условие наших критериев.

1) Заданы критерии:

$$W_1 = f_1(x_k) \rightarrow \max, W_2 = f_2(x_k) \rightarrow \max. \quad (1)$$

2) Производим «сворачивание» этих критериев в один критерий:

$$W = \alpha f_1(x_k) + \beta f_2(x_k) \rightarrow \max$$

(2)

где $\alpha + \beta = 1$.

Далее решаем классическую задачу с одним критерием. Самое сложное в данном методе - это выбрать значения α и β . Обычно это делает человек, работающий с задачей, или нейросеть, которая заменяет человека. Для двух критериев есть удобный способ расстановки значений α и β , позволяющий отследить изменение множества лучших футболистов по мультикритерию в зависимости от самих весовых коэффициентов, которые отражают значимость критериев.

Пусть $\alpha = 1 - \gamma$, а $\beta = \gamma$, где γ - шаг свёртки, который мы определяем в начале решения.

$$W^{(k)} = \gamma f_1(x) + (1 - \gamma) f_2(x) \rightarrow \max \quad (4)$$

Тогда параметр γ свёртки последовательно принимает значения от 0 до 1:

$$0 = \alpha_0 < \alpha_1 < \alpha_2 < \dots < \alpha_n = 1. \quad (5)$$

Значимость критериев $f_1(x)$ и $f_2(x)$ постепенно меняется от одного к другому. Из-за постепенного изменения значимости критериев мы можем отследить, какой футболист лучший в зависимости от каждого критерия. Такой метод позволяет частично определить множество точек границы Парето. Это связано с тем, что линейная свёртка упускает «вогнутые» точки, а лучшими выборами в таком случае являются выпуклые точки.

Рассмотрим многокритериальную оптимизацию по Парето на примере задачи с 7 критериями.

Таблица 2 – Футболисты и их показатели для задачи с семью критериями

№	Фамилия	Голы, ед.	Пасы, ед.	Скор., м/с	Возраст, лет	Опыт, лет	Возраст, лет	Цена, \$M
1	Зинченко	2	2	6	4	0	8	0.2
		1	1	4.		7	2	1

2	Яшин	0	1	2	3	4	6	6	0	2	5	3	9
3	Соловьев		6	1	5	5	9	7	5	5	2	2	8
4	Попов	4	1	0	5	5	1	6	9	6	2	2	1
5	Косьмин		0	1	9	3	3	6	5	2	2	2	1
6	Уткин	3	1	0	5	5	5	6	1	2	6	3	1
7	Дзюба	2	1	6	6	2	4	8	7	1	3	3	1
8	Брежнев		2	1	5	3	9	5	5	1	0	3	7
9	Медведев		2	5	2	4	9	6	6	6	3	2	6

При такой задаче мы сталкиваемся с проблемой интерпретации в графическом виде, поэтому можно использовать так называемую лепестковую диаграмму (см. рисунок 2).

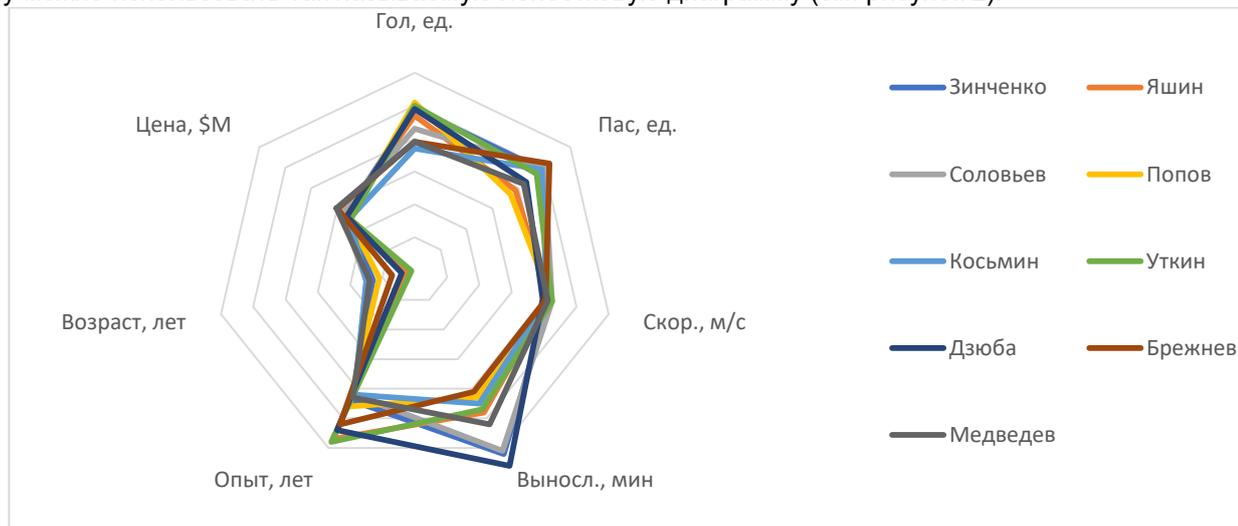


Рисунок 2 - Лепестковая диаграмма для задачи с 7 критериями.

В данной задаче будем определять точки, принадлежащие границе Парето, следующим образом: точка 1 доминирует над точкой 2, если график точки 2 полностью входит в график точки 1 и не пересекает его. Но для этого нужно изменить направление увеличения полезности по некоторым осям, а именно по осям возраста и цены, потому что чем моложе и дешевле футболист – тем лучше.

На примере этой задачи видно, что одним графическим методом уже не обойтись, и даже лепестковая диаграмма не так удобна и может привести к ошибкам. Поэтому для задач с большим количеством критериев лучше использовать обобщенный алгоритм оптимизации по Парето из первой задачи на двух критериях.

В продолжение многокритериальной оптимизации по Парето рассмотрим метод идеальной точки. Идеальной точкой или точкой абсолютного максимума называют точку в критериальном пространстве, в которой все критерии достигают своих максимальных значений. Если эта точка принадлежит достижимому множеству, то все эффективное (паретовское) множество состоит из этой единственной точки, и проблемы как таковой в этом случае нет. Однако идеальная точка обычно лежит вне множества и поэтому нереализуема. В связи с этим ее иногда называют утопической. Идея метода состоит в том, чтобы на множестве найти точку, наиболее близкую к идеальной. Следующий алгоритм описывает процедуру выбора наиболее предпочтительного объекта.

Алгоритм 2:

Шаг 1: Формирование «идеального объекта».

$$f_1^{max} = x_0, f_2^{max} = y_0, f_3^{max} = z_0, \dots, f_n^{max} = h_0 \quad (6)$$

Шаг 2: Определение для каждого объекта многокритериальной метрики (расстояния) до «идеального объекта».

$$W_j = \sqrt{\sum_{i=0}^k \left(\frac{f_i(x_j) - f_i^{max}}{f_i^{max}} \right)^2} \quad (7)$$

где W – глобальный критерий, который минимизирует расстояние до идеальной точки, $f_i(x)$ – это значение i -ого критерия, x – номер элемента, f_i^{max} – максимальное значение i -ого критерия.

Шаг 3: Анализ множества объектов на соответствие (степень близости) «идеальному объекту».

$$W_j = \sqrt{\sum_{i=0}^k \left(\frac{f_i(x_j) - f_i^{max}}{f_i^{max}} \right)^2} \rightarrow \min \quad (8)$$

Пояснение:

В данной формулировке задачи идеальная точка может иметь координаты x_0, y_0 и т.д. в зависимости от количества критериев. x_0, y_0 – это максимальные значения забитых голов, отданных пасов, лучшей скорости, выносливости и т.д. Не обязательно брать за максимальные значения именно x_0, y_0, z_0, h_0 и т.д., можно взять любые достаточно большие значения, но в данной формулировке такие значения удобны. На шаге 3 находим минимальное значение W . В зависимости от задачи можем искать ещё и близкие к минимальному значения в случае, если нужно найти, например, пятерку лучших игроков. Для этого один раз пройдем по множеству W и найдём самое маленькое значение. В результате работы алгоритма по нахождению минимального значения будем знать минимальное расстояние, которое хранится в переменной min и номер объекта,

```
for(int j = 0; j < n; j++)
{
    for(int i = 0; i < k; i++)
    {
        W[j] ^ 2 += ((f[i](x[j]) - max(f[i]))/(max(f[i]))) ^ 2;
    }
    W[j] = sqrt(W[j] ^ 2);
}
```

соответствующего данному расстоянию.

Осталось рассмотреть метод контрольных показателей. Данный метод используется для поиска сбалансированного решения, которое наиболее удалено от минимальной границы по всем показателям. В общем виде идея выглядит следующим образом:

Пусть имеется n -критериальная задача:

$$W_i = f_i(x) \rightarrow \max, i = \overline{1, n} \quad (9)$$

Назначим каждой функции $f_i(x)$ её контрольный показатель – нижнюю границу:

$$f_i(x) \geq f_i^* \quad (10)$$

Введём глобальный критерий W , который максимизирует минимальное расстояние от функции $f_i(x)$ до нижней границы f_i^* :

$$W = \min_{1 \leq i \leq n} \left\{ \frac{f_i(x)}{f_i^*} \right\} \rightarrow \max_x \quad (11)$$

Исходя из очевидного неравенства (слева) получаем ограничение задачи (справа):

$$\min_{1 \leq i \leq n} \left\{ \frac{f_i(x)}{f_i^*} \right\} \leq \frac{f_i(x)}{f_i^*} \Rightarrow f_i(x) \geq f_i^* W$$

(12)

Решая задачу, находим такое W , при котором все функции $f_i(x)$ принимают значения, наиболее удаленные от своих нижних границ f_i^* .

Таблица 3 – перерасчёт данных для метода контрольных показателей

k	Фамилия x	Г ол, ед. $f_1(x)$ $/f_1^*$	П ас, ед. $f_2(x)$ $/f_2^*$	С кор., м/с $f_3(x)$ $/f_3^*$	В ыносл., мин $f_4(x)$ $/f_4^*$	О пыт, лет $f_5(x)$ $/f_5^*$	В озраст, лет $f_6(x)$ $/f_6^*$	Ц ена, \$M $f_7(x)$ $/f_7^*$	М in
	Зинченко	1 2	1 2	4 .6	8 0	7 0	2 4	1 0.2	1 .47
	Яшин	1 0	2	4 .3	6 6	2 0	3 5	9 .9	1 .02
	Соловьев	6 3	1	5 .5	7 9	5	2 2	8 .6	1 .2
	Попов	1 4	0	5 .5	6 1	9	2 6	1 1.4	
	Косьмин	0	1 2	3 .9	6 3	5	2 2	1 1.8	
	Уткин	1 3	1 0	5 .5	6 5	2 1	3 6	1 2.2	1
	Дзюба	1 2	6	2 .6	8 4	1 7	3 3	1 0.8	1 .04
	Брежнев	2	1 5	3 .5	5 9	1 5	3 0	7 .1	
	Медведев	2	5	4 .2	6 9	6	2 3	6 .8	

В результате оценки методом контрольных показателей получаем, что Зинченко самый «безопасный» футболист, максимально удаленный от минимальных границ критериев.

В данной работе рассмотрены проблемы многокритериальных задач и некоторые способы их решения, а именно, многокритериальная оптимизация по Парето, метод линейной свертки, метод контрольных показателей и метод идеальной точки. Все эти методы работают при любом количестве критериев в задаче и при правильном использовании могут значительно сократить количество вариантов, из которых делается выбор. Логичнее всего сразу оптимизировать заданное множество по Парето, тем самым сократив его в разы или даже до пары значений, а затем, применив метод идеальной точки уже для усеченного множества, найти самый лучший, взвешенный вариант или найти сразу несколько удовлетворяющих нас вариантов, указав предварительно интервал отклонения от самого лучшего варианта. Отметим, что даже для поиска одного варианта следует рассмотреть не только самую близкую к идеальной точке, но и близкие к ней.

Список использованных источников:

1. Мир математики: Висенц Торра, Математика и выборы. Принятие решений. / Пер. с исп. – М.: Де Агостини, 2014 – 160 с.
2. Ehrgott, M. *Multicriteria Optimization*. / M. Ehrgott // Springer; 2nd edition, 2005
3. T'kindt, V. *Multicriteria Scheduling. Theory, Models and Algorithms*/ V.T'kindt, J.-C. Billaut – Springer, 2nd edition 2006. - 359p.

UDC 519.8

MULTI-CRITERIA PARETO OPTIMIZATION

Ershov O.V., Skvortsov A.V.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Lushchakova I. N. – PhD in Physics and Mathematics

Annotation. We consider the ideas of Pareto optimization, linear aggregation method, the ideal point method and the benchmark method. Algorithms for determining whether the element belongs to the Pareto boundary set and for determining the best option by the ideal point method are given. As an example, we consider the multi-criteria problem of selecting players for a football club.

Keywords. Pareto optimization, ideal point method, benchmark method.

УДК

ПРАКТИЧЕСКОЕ ПРИМЕНЕНИЕ ФРАКТАЛОВ В СОВРЕМЕННОЙ ЖИЗНИ

Жуковская Е.Г., Буцкевич М.А., студенты гр. 124403

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Примичева З.Н., канд. физ.-мат. наук, доцент

Аннотация: Фракталы – это удивительные фигуры с определёнными свойствами, которые окружают нас повсюду. Они бывают разными и применяются в различных сферах, о которых вы даже не подозревали. Эта статья представит вам понятие фрактала, расскажет историю этого термина и покажет его практическое применение. Приятного прочтения.

Ключевые слова: математика, фрактал, Бенуа Мандельброт, геометрическая фигура, термин, геометрический, алгебраический, стохастический, самоподобие, формулы, природные объекты, наука, медицина, информатика, современное искусство, языки программирования.

Введение:

Математика – удивительное явление, которое окружает нас постоянно. Мы можем её не замечать или даже вовсе не знать о ней, но она всегда присутствует в нашей жизни. Удивительную красоту природы, человека или животного всегда можно рассмотреть со стороны геометрии. Но нельзя это сделать, используя только простые фигуры такие как, треугольник, окружность, сфера и т.д. Для этого нужны более сложные и утончённые понятия. Одним из них и является фрактал.

История фракталов:

Понятие фрактала вошло в нашу жизнь не так давно, всего в конце 20 века. Термин фрактала был введён Бенуа Мандельбротом в 1975 году, хотя научному математическому сообществу стал известен только после выхода книги учёного «Фрактальная геометрия природы» в 1977 году. Фрактал – это сложная геометрическая фигура, обладающая свойством самоподобия, т.е. составленная из нескольких частей, каждая из которых подобна всей фигуре целиком. Данное понятие является самым распространённым. Мандельброт немного по-другому растолковал этот термин: фракталы – семейство фигур, состоящих из неправильно фрагментированных форм в окружающем нас мире [1].

Раньше фракталы не воспринимались учёными всерьёз и их изучение не являлось полноценной наукой. Но благодаря приложенным усилиям Бенуа, фрактал признали как математический термин, а фрактальная геометрия стала полноценной наукой.

Виды фракталов:

Фракталы делятся на 3 вида: геометрические, алгебраические и стохастические.

Геометрические фракталы (рис.1) – это фигуры, которые обладают нижеперечисленными свойствами:

- являются приближённо само подобными;
- обладают сложной структурой при любом увеличении;
- могут быть построены рекурсивными процедурами.

Если фигура не соответствует хотя бы одному из свойств, то её нельзя назвать фракталом.

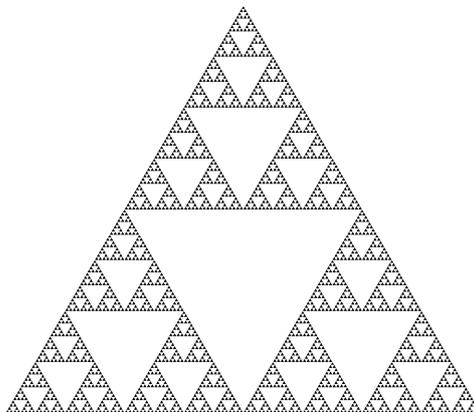


Рисунок 1 – треугольник Серпинского

Алгебраические (динамические) фракталы построены на тяжеловесных формулах. Если вписать эти формулы в специальные программы и совершать несколько итераций, то получится обладающая самоподобием фигура, а именно геометрический фрактал. Они напрямую связаны между собой, т.к. геометрические фракталы невозможно построить без определённых формул (алгебраических фракталов). Множество Мандельброта (рис.2) относится к алгебраическим фракталам.

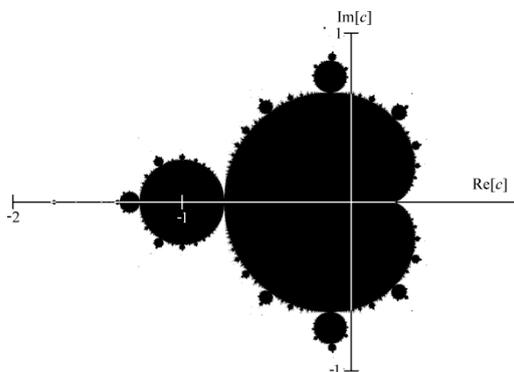


Рисунок 2 – Множество Мандельброта

Стохастические фракталы (рис.3) – зачастую это природные объекты, которые имеют свойства фракталов. Строят их с помощью случайных фракталов.



Рисунок 3 – стохастический фрактал папоротник

Применение фракталов:

Фракталы применяются в многих областях жизни современного человека.

В естественных науках учёные применяют фракталы для моделирования различных пористых материалов (к примеру, в нефтехимии). Также применяются в физике и радиоэлектронике для моделирования нелинейных процессов, волн, воды, пламени, облаков и т.д.

В медицине фракталы используются для построения кровеносной системы и некоторых органов человека. Ещё они нашли применение в локализации опухолей. Выяснилось, что опухоль имеет фрактальные свойства. Поэтому при её моделировании можно найти очаг воспаления.

Фракталы широко применяются во многих сферах информатики. К примеру, сжатие изображений построено на теории фракталов. Такой способ был разработан в компании Microsoft, но, к сожалению, не получил широкой огласки. Фракталы используются в современной компьютерной графике, особенно 3D (рис.4-5).



Рисунок 4 – капуста Романеско



Рисунок 5 – фрактальное дерево

С помощью новейших технологий и программ можно создать удивительные изображения. Для программистов фрактал является своеобразным «учебником» для освоения графического языка программирования.

Самое удивительное, что фракталы набирают огромную популярность в искусстве. Многие художники используют их, чтобы разнообразить свои работы и придать им свою неповторимую изюминку. Также фракталы используются, чтобы с нуля создавать невообразимые шедевры современно искусства. В некоторых европейских странах успешно прошли первые выставки подобных работ.

Мы изучили фракталы и с помощью языков C++ и Python построили снежинку Коха и Кривую дракона. C++ и Python – это современные языки программирования, которые повсеместно используют в IT-компаниях.

Для того, чтобы создать снежинку Коха (рис.6) нам потребовалось вывести ряд определённых формул (рис.7).

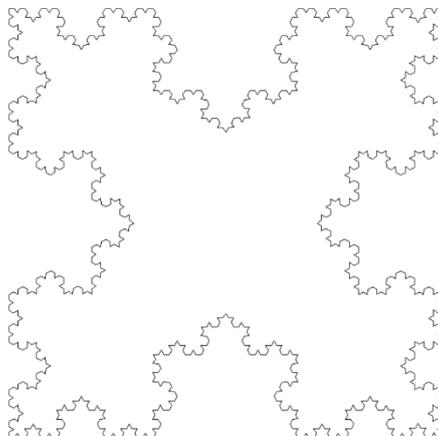


Рисунок 6 – изображение снежинки Коха, которую нам удалось получить

$$\begin{aligned} &A(x_A, y_A), \\ &B(x_B, y_B), \\ C &\left(\frac{x_A+x_B}{2} + \frac{y_A-y_B}{2\sqrt{3}}, \frac{y_A+y_B}{2} + \frac{x_B-x_A}{2\sqrt{3}} \right). \end{aligned}$$

(1) Точки координат A, B и C

$$\begin{aligned} A_1 &\left(\frac{2x_A+x_B}{3}, \frac{2y_A+y_B}{3} \right), \\ B_1 &\left(\frac{x_A+2x_B}{3}, \frac{y_A+2y_B}{3} \right). \end{aligned}$$

(2) Точки координат A1 и B1

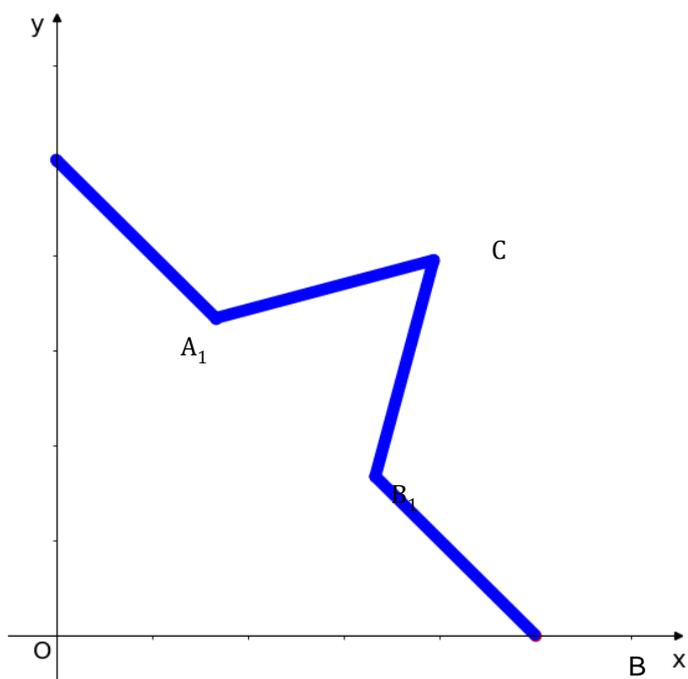


Рисунок 7 – снежинка Коха в системе координат

В ходе нашей работы мы наглядно поняли, как связаны алгебраические и геометрические фракталы и как с помощью программы перевести формулы в изображение.

Для Кривой дракона не требуется выведение формул, нужно лишь понять закономерность поворотов. К примеру возьмём лист бумаги и сложим его 1 раз. У нас появилась линия сгиба, которую мы обозначим, как поворот вправо. Если мы сложим эту же бумагу во второй раз, то у нас повторится наш 1 поворот, но он будет отзеркаленным (т.е. появится поворот влево) и появится дополнительный поворот вправо. Сложив бумагу в третий раз, все повороты повторятся в отзеркаленном виде и появится ещё один поворот вправо. Если это всё преобразовать в код на Python, используя библиотеку операций “черепашка”, и повторить подобные “отзеркаливания” много раз, то у нас получится Кривая дракона (рис.8).

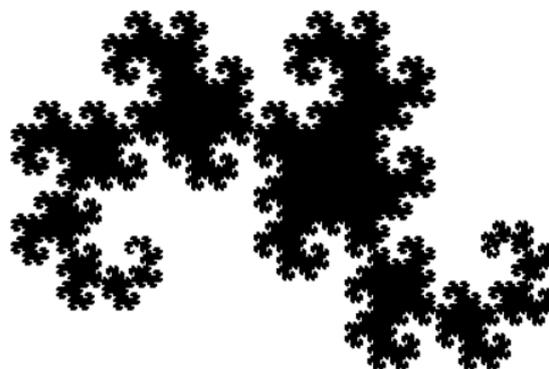


Рисунок 8 - изображение Кривой дракона, которую нам удалось получить

Список использованных источников:

2002.

1. Манделъброт, Б. Фрактальная геометрия природы. — Москва: Институт компьютерных исследований,

2. <https://elementy.ru/posters/fractals/fractals>

3. <https://studfile.net>

UDK

PRACTICAL APPLICATION OF THE FRACTALS IN MORDEN LIFE

Zhukovskaya K.G. and Butskevich M.A.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Primicheva Z.N. – PhD in Physics and Mathematics, Associate Professor

Annotation: Fractals – astonishing figures with a specific characteristic, they surround us everywhere. They are all different and can be used in various of scopes. This article will present you the term of fractal, retell its history and most importantly the article will show you the practical implementation of the fractals.

Keywords: fractal, Benoit Mandelbrot, geometric figures, self-similarity, Python, C++

УДК

ПРОГРАММА ВИЗУАЛИЗАЦИИ И ПОДСЧЕТА ЧИСЛА ФРОБЕНИУСА НА ОСНОВЕ ДВУХКОНТУРНОЙ СЕТИ И ФОРМУЛЫ РЁДСЕТА

Химич Н. А., Быков А. Д., студенты гр. 124405.
Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Примичева З. Н. – канд. физ.-мат. наук

Аннотация. В данной работе рассматриваются числа Фробениуса и составлена программа для их нахождения с помощью двухконтурных сетей и цепных дробей.

Ключевые слова. Число Фробениуса, двухконтурная сеть, формула Рёдсета.

Введение. Пусть a_1, a_2, \dots, a_n — натуральные числа, взаимно простые в совокупности (наибольший общий делитель всех чисел равен 1). Числом Фробениуса $g(a_1, a_2, \dots, a_n)$ называется наибольшее целое число, непредставимое в виде линейной комбинации

$$x_1 a_1 + x_2 a_2 + \dots + x_n a_n, \#(1)$$

где x_1, x_2, \dots, x_n — целые неотрицательные числа.

Задачу о нахождении числа Фробениуса также называют “Задачей Фробениуса с обменом монет”, поскольку число Фробениуса — это максимальная сумма, которую нельзя представить купюрами номиналом a_1, a_2, \dots, a_n .

Множеством NR называется конечное множество всех целых чисел, непредставимых в виде линейной комбинации (1), числом Фробениуса - наибольшее число из этого множества.

Известна формула, полученная Сильвестром [1], нахождения числа Фробениуса при $n = 2$:

$$g(a_1, a_2) = a_1 a_2 - a_1 - a_2. \#(2)$$

Нахождение числа Фробениуса при $n = 3$ с использованием цепных дробей рассматривается в работах Сельмера, А. О. Бейера и О. Дж. Рёдсета [2].

При $n \geq 4$ известны формулы для нахождения $g(a_1, a_2, \dots, a_n)$ лишь для некоторых частных случаев. Показано [3], что для фиксированного n решение можно найти за полиномиальное время, а при произвольно выбранном n такая задача является NP-трудной.

Нахождение числа Фробениуса с использованием двухконтурных сетей изучено А.В. Устиновым [4]. Формула нахождения числа Фробениуса при помощи цепных дробей получена Рёдсетом [3].

В статье рассматривается нахождение числа Фробениуса двумя способами: с помощью двухконтурных сетей и цепных дробей. Составлен алгоритм и написана программа нахождения числа Фробениуса при $n = 3$ указанными способами.

Формула Джонсона [5]. При нахождении числа Фробениуса $g(a, b, c)$ можно избавляться от общих делителей аргументов

$$g(d \cdot a, d \cdot b, c) = d \cdot g(a, b, c) + c \cdot (d - 1). \#(3)$$

Учитывая данную формулу, в дальнейшем будем считать, что $\text{НОД}(a, b) = \text{НОД}(b, c) = \text{НОД}(a, c) = 1$.

Двухконтурная сеть. Допустим, что $a = \max(a, b, c)$. Каждой тройке чисел a, b, c поставим в соответствие двухконтурную сеть – ориентированный граф с a вершинами $0, 1, \dots, a - 1$ и ребрами двух типов: $j \rightarrow j + b \pmod{a}$ и $j \rightarrow j + c \pmod{a}$ весами $w_b = b$ и $w_c = c$ соответственно.

На рисунке 1 приведен пример двухконтурной сети для $a = 7, b = 3, c = 5$. Черным пунктиром обозначены связи с весом $w_b = 3$. Синими обозначены связи с весом $w_c = 5$.

Красным изгибом обозначен самый длинный путь с самым большим суммарным весом из вершины 0.

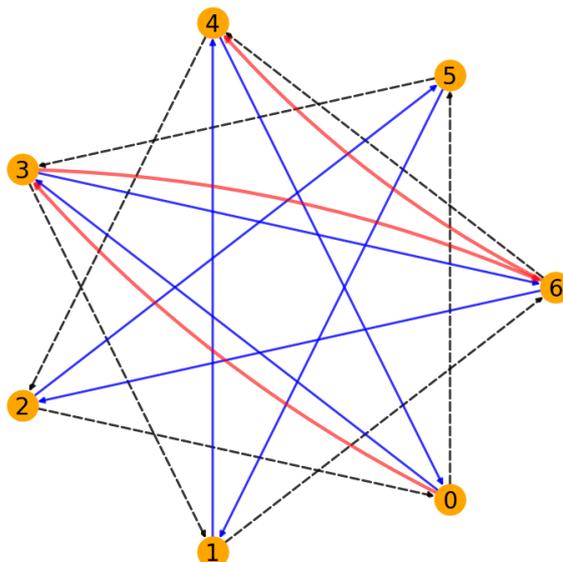


Рисунок 1 - Схема двухконтурной сети для $a = 7, b = 3, c = 5$

Введем функцию $time(x, y) = bx + cy$, которая определяет суммарный вес пути, где x - количество ребер весом wb , а y - количество ребер весом wc . Для каждого кратчайшего пути из вершины 0 до каждой другой вершины двухконтурной сети поставим в соответствие клетку с координатами (x, y) и запишем в нее значение $time(x, y)$ - сумма весов маршрута. Получится диаграмма, приведенная на рисунке 2.1. Построим вторую диаграмму, для каждого значения $time(x, y)$ запишем в клетку с координатами (x, y) значение $point(x, y) = time(x, y) \pmod a$ (рисунок 2.2). Из построения получим: чтобы дойти из вершины 0 до вершины, записанной в клетку полученной диаграммы, координаты которой (x, y) , надо пройти x ребер с весом wb , и y ребер с весом wc .

Диаграмма (рисунок 2.1), построенная в соответствии с графом (рисунок 1), имеет L -образную форму. Тогда пусть точка $C = (x_C, y_C)$ и $E = (x_E, y_E)$ - крайние правые точки ступеней построенной L -образной диаграммы. В примере: $time(x_C, y_C) = 11$ и $time(x_E, y_E) = 9$ соответственно.

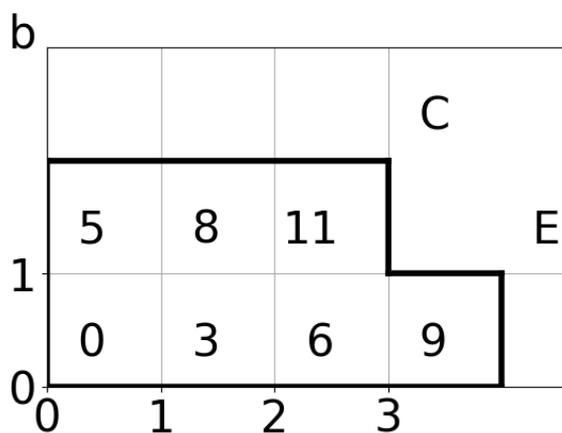


Рисунок 2.1

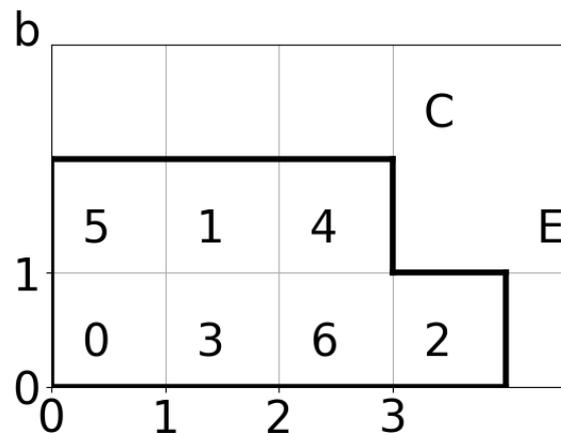


Рисунок 2.2

Теорема 1. Число Фробениуса равно:

$$g(a, b, c) = \max\{t(x_C, y_C), t(x_E, y_E)\} - a - b - c. \#(4)$$

Формула Рёдсета. Составим дробь $\frac{a}{S_0}$, где целое число S_0 определяется из условия:

$$bS_0 \equiv c \pmod{a}, \quad 0 \leq S_0 < a.$$

Построим числовую последовательность $\{S_i\}$, используя алгоритм Евклида в виде:

$$\left\{ \begin{array}{l} a = S_{-1} = q_1 S_0 - S_1, \quad 0 \leq S_1 < S_0; \\ S_0 = q_2 S_1 - S_2, \quad 0 \leq S_2 < S_1; \\ S_1 = q_3 S_2 - S_3, \quad 0 \leq S_3 < S_2; \\ \dots \\ S_{m-2} = q_m S_{m-1} - S_m, \quad 0 \leq S_m < S_{m-1}; \\ S_{m-1} = q_{m+1} S_m, \quad 0 \leq S_{m+1} < S_m. \end{array} \right.$$

Тогда число $\frac{a}{S_0}$ можно представить в виде цепной дроби:

$$\frac{a}{S_0} = q_1 - \frac{1}{q_2 - \frac{1}{\dots - \frac{1}{q_{m-1} - \frac{1}{q_m}}}}$$

Также определим целочисленную последовательность $\{P_i\}$: $P_{-1} = 0, P_0 = 1, P_{i+1} = q_{i+1}P_i - P_{i-1}, i = 0, 1, 2, \dots, m$. Полагая, что $\frac{S_{-1}}{P_{-1}} = \infty$, и так как $q_i \geq 2$, последовательность $\{P_i\}$ монотонно возрастает, $\{S_i\}$ монотонно убывает, значит имеет место неравенство:

$$0 = \frac{S_{m+1}}{P_{m+1}} < \frac{S_m}{P_m} < \dots < \frac{S_0}{P_0} < \frac{S_{-1}}{P_{-1}} = \infty$$

Найдем номер $v, -1 \leq v \leq m$, такой, что $\frac{S_{v+1}}{P_{v+1}} \leq \frac{c}{b} < \frac{S_v}{P_v}$.

Теорема 2. Пусть a, b, c – положительные целые числа и a, b взаимно простые. Число Фробениуса определяется по формуле:

$$g(a, b, c) = -a + b \cdot (S_v - 1) + c \cdot (P_{v+1} - 1) - \min(b \cdot S_{v+1}, c \cdot P_v). \# \quad (5)$$

Результат работы. Результатом работы является составленная нами программа, которая быстро подсчитывает число Фробениуса двумя способами: на основе двухконтурной сети и формулы Рёдсета. А также рисует двухконтурную сеть, две диаграммы и подробный ход нахождения числа Фробениуса с использованием цепных дробей. Скриншот работы программы для примера $a = 12, b = 17, c = 11$ приведен на рисунке 3.

Части интерфейса программы. Программа условно поделена на четыре части: поле ввода данных, поле вывода численных данных, 3 графика, относящиеся к методу двухконтурных сетей, и 1 график с подсчётом через формулу Рёдсета.

Часть ввода данных представлена в виде надписи «Введите 3 числа:» и трех полей ввода, где пользователь может ввести три числа, для которых будет найдено число Фробениуса.

Ниже находится кнопка «Посчитать», нажатие на которую запускает программу.

Часть вывода численных данных представлена в виде четырех полей: в поле «Упрощенные числа» выводятся числа a, b, c после применения формулы Джонсона (3). В поле «Делители» выводятся общие делители аргументов функции, полученные при применении формулы Джонсона. В поле «Число Фробениуса» выводится посчитанное число Фробениуса с помощью Двухконтурной сети. И в поле «NR множество» выводится NR множество.

Три графика «Граф», «Сумма пути до вершины» и «Путь до вершины» соответствуют рисункам 1, 2.1 и 2.2 в пункте «Двухконтурные сети».

График «Число Фробениуса через цепные дроби» содержит в себе алгоритм, описанный в пункте «Формула Рёдсета».

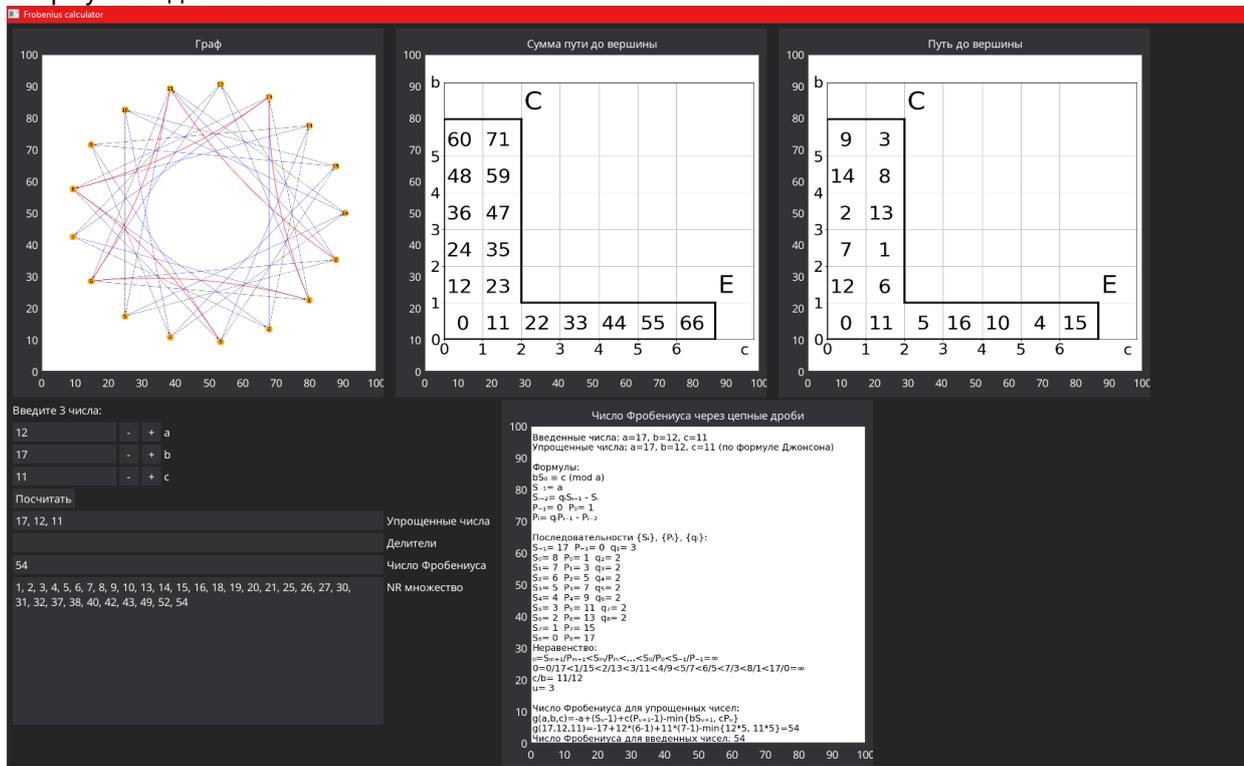


Рисунок 3 - Результат работы программы для примера $a = 12, b = 17, c = 11$.

Вывод. Сконструированная программа позволяет быстро и наглядно находить число Фробениуса для $n = 3$, поскольку решение этой задачи занимает огромное количество времени. За счет наглядности геометрического способа и структурированного вывода нахождения числа Фробениуса через цепные дроби, пользователь способен в краткие сроки обучиться методам нахождения числа Фробениуса.

Список использованных источников:

1. Sylvester J.J. Question 7382 // *Educ. Times*. 1884. V. 37. P. 26; *Mathematics from the Educational Times, with additional papers and solutions* // *Mathematical questions, with their solutions, from the "Educational Times"*. London: F. Hodgson, 1884. V. 41. P. 21.
2. Rodseth O.J. On a linear Diophantine problem of Frobenius // *J. reine angew. Math.* 1978. Bd. 301. S. 171–178.
3. Kannan R. Lattice translates of a polytope and the Frobenius problem // *Combinatorica*. 1992. V. 12, N 2. P. 161–177.
4. A. V. Ustinov, Geometric proof of Rødseth's formula for Frobenius numbers, *Trudy Mat. Inst. Steklova*, 2012, Volume 276, 280–287
5. Johnson S.M. A linear diophantine problem // *Can. J. Math.* 1960. V. 12. P. 390–398.

UDC

**PROGRAM FOR VISUALIZATION AND CALCULATION OF THE
FROBENIUS NUMBER ON THE BASIS OF A DUAL CIRCUIT
NETWORKS AND THE RÖDSET FORMULA**

Khimich N. A., Bykov A. D.

*Belarusian State University of Informatics and Radioelectronics,
Minsk, Republic of Belarus*

Primicheva Z. N. - Ph.D. in Physics and Mathematics

Annotation. The article discusses Frobenius numbers and developed program, which designed to calculate the Frobenius number by dual circuit networks and continued fractions.

Keywords. Frobenius number, dual circuit networks, Rödset formula.4

МОДЕЛИРОВАНИЕ РАСПРОСТРАНЕНИЯ РЕКЛАМЫ

Попов В.Д., Дундер Н.А., студенты гр. 172301

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Ефремов А. А. – канд. экон. наук, доцент

Аннотация. Статья посвящена описанию и построению модели распространения рекламы. Построение данной дифференциальной модели основано на задаче из макроэкономической динамики.

Дифференциальное уравнение является одним из фундаментальных понятий математики, широко применяемое в различных областях современных наук. Данный тип уравнений – это уравнения, в которые входит неизвестная функция под знаком дифференциала или же производной. Дифференциальное уравнение, получают в итоге изучения любого реального явления, подобный процесс называют дифференциальной моделью этого явления. Данный раздел математики является частным примером математических моделей, которые могут строиться при изучении окружающего нас мира. Для построения моделей методами дифференциального исчисления немало важную роль играют знания законов предметной области изучаемых явлений или процессов. В моделях экономической динамики, достаточно обширно применяют дифференциальные уравнения. В них отражается не только зависимость переменных от времени, но и их взаимосвязь во времени.

В повседневной жизни, на производстве, в научно-исследовательской, инженерной и другой деятельности человек постоянно сталкивается с решением задач. Реальные объекты и процессы бывают столь многогранны и сложны, что лучшим способом их изучения часто является построение и исследование модели, отображающей определённые свойства и критерии изучаемого объекта. Примером физической модели является фотография, модель самолёта, модель кристаллической решётки, макет здания и т.д.

Математическая модель – искусственно созданный объект в виде математических знаковых формул. Пример математической модели – математические соотношения, позволяющие рассчитать оптимальный план работы какого-либо предприятия, а также система линейных, нелинейных уравнений. Таким образом, технология математического моделирования системы во многом зависит от решаемых задач моделирования и особенностей системы, что позволяют выделить и описать этапы решения задачи.

Основная идея нашей работы заключается в описании модели распространения рекламы.

Чтобы реклама была эффективной, необходимо доносить ее до наибольшей аудитории с максимальной частотой. Основная цель рекламы: заставить потребителя выполнить желаемое рекламодателем действие (убедить потребителя купить товар, воспользоваться услугой, придать известность фирме).

Рассмотрим данную тему на примере задачи макроэкономической динамики.

Чтобы выбрать средства распространения, рекламодатель решает, насколько широким должен быть охват рекламы и как часто она должна повторяться для достижения рекламных целей.

$$\frac{dx}{dt} = kx(N - x) \quad (1),$$

где N – количество потенциальных покупателей; x – клиенты торговой организации;

k – положительный коэффициент пропорциональности.

При этом, $x = N/\gamma$, где γ – коэффициент, показывающий во сколько раз клиентов меньше, чем количество потенциальных покупателей.

Интегрируя уравнение (1) при $t = 0$ получаем:

$$\frac{1}{N} \ln \frac{x}{N - x} = kt + C$$

Полагая $NC=C_1$, приходим к равенству:

$$\frac{x}{N-x} = Ae^{Nkt}, A = e^{C_1}$$

Если последнее уравнение решить относительно x и произвести замену $P=1/A$, то получим соотношение:

$$x = N \frac{Ae^{Nkt}}{Ae^{Nkt}+1} = \frac{N}{1+Pe^{-Nkt}} \quad (2)$$

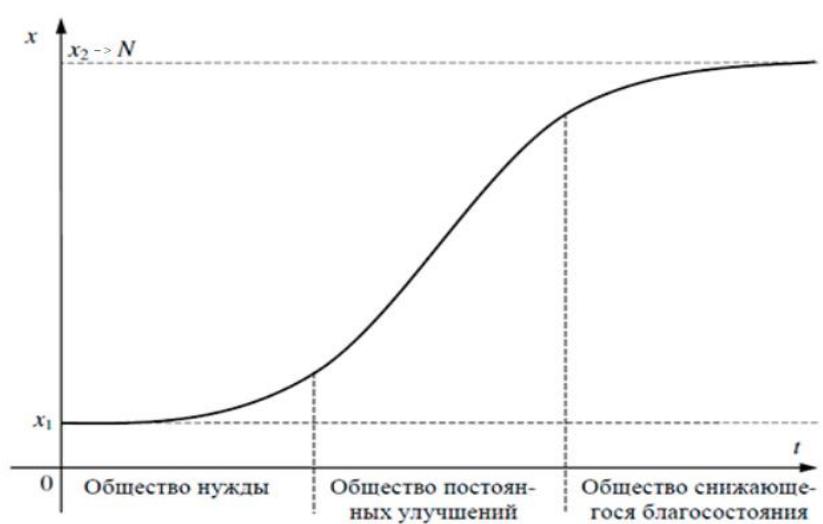
Такое уравнение (2) называется уравнение логической кривой. Вернёмся к изначальным переменным.

$$x = \frac{N}{1+(\gamma-1)e^{-Nkt}} \quad (3)$$

Пусть количество потенциальных покупателей $N=100$. Положительный коэффициент пропорциональности $k= 1/2000(0,0005)$. Примем коэффициент $\gamma = 3$. Время распространения рекламы возьмём за 2 недели ($t=14$). Подставим данные значение в формулу (3) получим: $x \approx 50$.

Из этого сделаем вывод, что при данных условиях за 2 недели распространения рекламы $\approx 50\%$ потенциальных покупателей станут реальными клиентами торгового учреждения.

Данное выражение является решением задачи и дает возможность определить количество людей до которых дошла реклама за время t .



Таким образом, математический анализ распространения рекламы торгового предприятия не могут предоставить вполне достоверной информации для моделирования и прогнозирования поведения клиентов и потенциальных покупателей.

Зачастую экономические исследования ограничиваются только арифметикой. Это распространенная ошибка руководителей торговых предприятий.

Сегодня достаточно занять определенное место на рынке и периодически напоминать потребителю о себе. В условиях конкуренции каждое решение должно быть подкреплено аналитическим исследованием.

Список использованных источников:

1. Логистическая прямая на примерах [Электронный ресурс] – Режим доступа: <https://naprimerax.org/posts/55/logisticheskaja-krivaja> – Дата доступа 9.04.2022.
2. Средства распространения рекламы, используемые в рекламной деятельности [Электронный ресурс] – Режим доступа <https://studbooks.n et/996165/marketing..> – Дата доступа 9.04.2022.

ДИНАМИЧЕСКОЕ МОДЕЛИРОВАНИЕ КОНКУРЕНЦИИ ПРОМЫШЛЕННОГО ПРОИЗВОДСТВА РОССИИ И БЕЛАРУСИ НА ОСНОВЕ МОДЕЛИ ЛОТКИ-ВОЛЬТЕРРЫ

Шарафанович Я.О., Васенко К.А., студенты гр. 172602

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Ефремов А. А. – канд. экон. наук, доцент

Аннотация. Статья посвящена описанию и построению модели распространения конкурирующих процессов производства России и Беларуси. Построение данной дифференциальной модели основано на задаче из экономики.

Дифференциальное уравнение является одним из фундаментальных понятий математики, широко применяемое в различных областях современных наук. Данный тип уравнений помимо функции содержит её производные. Порядок входящих в уравнение производных может быть различен (формально он ничем не ограничен). Производные, функции, независимые переменные и параметры могут входить в уравнение в различных комбинациях или отсутствовать вовсе, кроме хотя бы одной производной.

Большое значение, которое имеют дифференциальные уравнения для математики и особенно для её приложений, объясняется тем, что к решению таких уравнений сводится исследование многих физических и технических задач. Дифференциальные уравнения играют существенную роль и в других науках, таких, как биология, экономика и электротехника; в действительности, они возникают везде, где есть необходимость количественного (числового) описания явлений (коль скоро окружающий мир изменяется во времени, а условия изменяются от одного места к другому).

Математическая модель – математическое представление реальности, один из вариантов модели как системы, исследование которой позволяет получать информацию о некоторой другой системе. Математическая модель, в частности, предназначена для прогнозирования поведения реального объекта, но всегда представляет собой ту или иную степень его идеализации.

Фазовый портрет – графическое изображение системы на фазовой плоскости (или в многомерном пространстве), по координатным осям которого отложены значения величин переменных системы. Поведение переменных во времени при таком способе представления для каждой начальной точки описывается фазовой траекторией. Совокупность таким фазовых траекторий для любых начальных условий представляет собой фазовый портрет.

Модель Лотки–Вольтерры – модель взаимодействия двух видов типа «хищник – жертва», названная в честь её авторов (Лотка, 1925; Вольтерра 1926), которые предложили модельные уравнения независимо друг от друга.

Такие уравнения можно использовать для моделирования систем «хищник – жертва», «паразит – хозяин», конкуренции и других видов взаимодействия между двумя видами.

Основная идея нашей работы заключается в описании модели конкурирующих процессов производства Беларуси и России.

Рассмотрим данную тему на реальном примере экономики.

Математическая модель конкуренции продукции машиностроения экономической системы учитывает следующие показатели:

x – индекс промышленного производства машин и оборудования для Беларуси;

y – индекс промышленного производства машин и оборудования для России;

a – коэффициент, характеризующий государственную поддержку и выделяемые субсидии на производство;

b – коэффициент влияния расходов на рост промышленного производства (степень подавления);

c – коэффициенты, характеризующий величину постоянных производственных издержек для России;

d – коэффициент, характеризующий снижение себестоимости продукции России;

e – коэффициент доступности ресурсов.

Тогда модель Лотки-Вольтерры имеет вид:

$$\begin{cases} \frac{dx}{dt} = ax - ex^2 - bxy \\ \frac{dy}{dt} = -cy + dx \end{cases} \quad (1)$$

Математическая модель экономической системы, опираясь на введенные обозначения, учитывает:

ax - увеличение скорости роста промышленного производства, связанное с государственной

поддержкой предприятий;

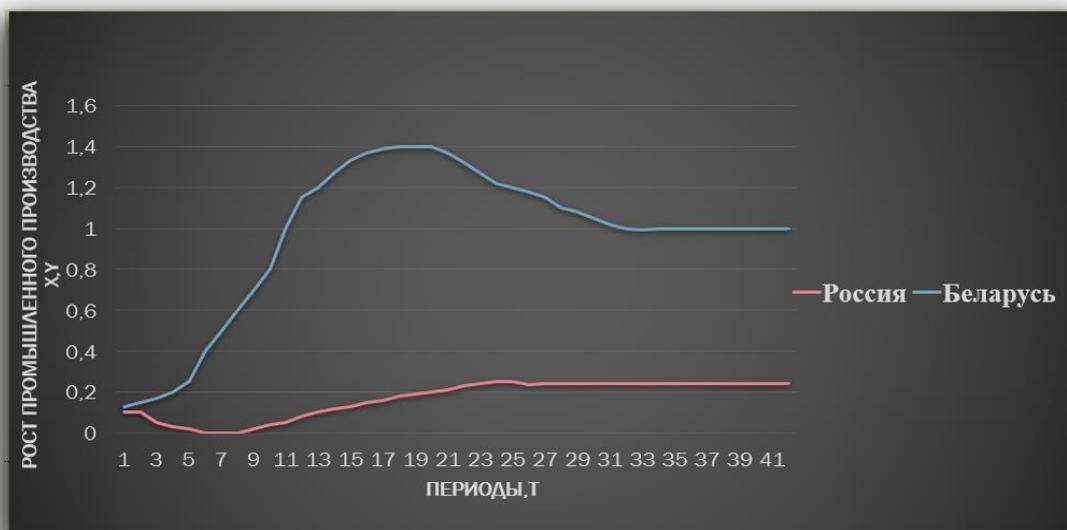
ex^2 - снижение скорости роста промышленного производства, связанное с внутренней конкуренцией за ресурсы;

$bху$ - снижение скорости роста промышленного производства в связи с Дополнительными расходами, связанными с возрастанием конкуренции со стороны российских производителей;

$су$ - снижение скорости роста промышленного производства, связанное с постоянными издержками;

$dxу$ - прирост удельных расходов за счет конкуренции на рынке ввиду более выгодных условий в энергетическом плане за счет снижения себестоимости.

Предположим, что Беларусь располагает богатой сырьевой и топливной базой, способной удовлетворить потребности государства на 70%, то есть показатель доступности ресурсов $e = 0.7$ при прочих равных условиях.



Из графика видно, что при заданных начальных условиях индекс роста промышленного производства Беларуси растет огромными темпами, и уже при $t = 10$, что соответствует 2021 году, достигает наивысшего значения 1,4 при прочих неизменных показателях. Затем при $t = 11,29$ наблюдается постепенный спад, связанный с истощением сырьевых ресурсов и переизбытком произведенной продукции, сходящийся к постоянному значению 1. В период спада Белорусского производства активизируется Российское производство - так, после $t = 10$ временных периодов практически неизменного производства, к периоду $t = 28$ Российское производство выйдет на показатель роста 0,3 и тем самым достигнет устойчивого значения. В итоге при $t = 28$ будет достигнуто устойчивое значения производства для обеих стран. Такой результат ожидается ко второй половине 2027 года.

Таким образом, интерпретируя данные результаты, можно отметить, что в случае, если Беларусь найдет новый источник восполнения своей ресурсной базы, то в кратчайшие сроки будет налажено интенсивное производство, график которой будет устойчивым. Также расширение сырьевой и энергетической базы может послужить стимулом к созданию новых подходов в производстве с использованием современных технологий высокой точности и диверсификации. В данном случае спады почти не характерны для промышленного производства, и значения индексов роста стремятся к стационарным значениям при $t = 28$.

Список использованных источников:

1. Апонин, Ю.М. Математическая модель сообщества хищник – жертва с нижним порогом численности жертвы / Ю.М. Апонин, Е.А Апонина // Компьютерные исследования и моделирование. – 2009. - № 1 (3). - С. 51-56.
2. Информация о результатах анализа развития производства машин и оборудования для сельского хозяйства государств – членов ТС и ЕЭП: периодич. обзор / Департамент промышленной политики Евразийской Экономической Комиссии. – Москва, 2014. – 23 с.

СЕКЦИЯ «ЭЛЕКТРОННЫЕ ВЫЧИСЛИТЕЛЬНЫЕ МАШИНЫ»

«FBOX» – модульный MIDI-СЕКВЕНСЕР НА БАЗЕ ФЛОППИ-ПРИВОДОВ

Калютчик А.А.¹, студент гр. 950504

Яценко В.П.¹, студент гр. 950504

Белорусский государственный университет информатики и радиоэлектроники¹
г. Минск, Республика Беларусь

Перцев Д.Ю. – канд. техн. наук

Аннотация. Данная работа - результат разработки курсового проекта по дисциплине “Схемотехника”. Целью данной работы является продемонстрировать работу флоппи-приводов в качестве MIDI-секвенсора. Отличительные особенности представленного решения - модульность и масштабируемость.

Ключевые слова. MIDI, секвенсер, USB, Floppotron, ATmega, Arduino.

Введение.

В 1990-е, когда в персональных компьютерах использовались флоппи-дисководы (англ. Floppy Disk Drive, FDD), при включении ПК издавал много звуков. Эти звуки создавали своего рода музыку, когда двигатели флоппи-дисководов трещали и дребезжали, жесткий диск раскручивался и жужжал, а различные периферийные устройства выполняли самодиагностику. Компьютерным энтузиастам захотелось собрать эти звуки в оркестр. Благодаря дешевым микроконтроллерным платам вроде Arduino или Raspberry Pi, появившимся в 2010-х годах, решить эту задачу стало в целом несложно: понадобились навыки программирования и пайки, а также усидчивость [1].

В 2011 году польский инженер Павел Задрожняк собрал «Floppotron» – музыкальный инструмент, состоящий из двух флоппи-приводов [2]. В 2016 году проект обновился до версии 2.0 и состоял из 64 флоппи-приводов, 8 жёстких дисков и 2 планшетных сканеров [3].

В данной статье представлен результат разработки курсового проекта по дисциплине «Схемотехника» студентами специальности «Вычислительные машины, системы и сети», в рамках которого реализована масштабированная до 15 флоппи-приводов версия «Floppotron 1.0». Целью работы было не только создание экземпляра музыкального инструмента, но и разработка модульного корпуса. Разработка получила название «Fbox» (англ. Floppy box, или Floppotron box).

Разработанная реализация логически состоит из:

- контроллера массива флоппи-приводов;
- преобразователя USB-MIDI;
- блока электроснабжения.

Преобразователь USB-MIDI.

MIDI (англ. Musical Instrument Digital Interface – цифровой интерфейс музыкальных инструментов) – стандарт цифровой звукозаписи для обмена данными между электронными музыкальными инструментами [4]. Традиционным для MIDI является разъем DIN-5, однако для подключения разработанного музыкального инструмента к источнику воспроизведения (ноутбук или ПК) был выбран более распространенный интерфейс внешней периферии – USB. При этом инструмент распознается музыкальным ПО как обычное MIDI-аудиоустройство и, как результат, не требует дополнительных действий со стороны пользователя.

Преобразователь реализован на базе 8-разрядного RISC-микроконтроллера Microchip (Atmel) AVR ATmega8A. Его преимуществом является большой набор готовых библиотек, выпускается в двух вариантах корпусов: DIP-28 и TQFP-32. Был выбран первый вариант (см. рис. 1) ввиду доступности и простоты монтажа на печатную плату.

Ввиду отсутствия у микроконтроллера аппаратной поддержки USB, была использована программная библиотека V-USB [6], позволяющая эмулировать поддержку протокола USB. Для этого библиотека предъявляет следующие минимальные требования:

- 2 Кб Flash-памяти;
- 128 байт ОЗУ;
- одно свободное аппаратное прерывание.

Быстродействие ATmega8A хватает для программной реализации USB 1.1 совместимого «Low-speed» устройства.

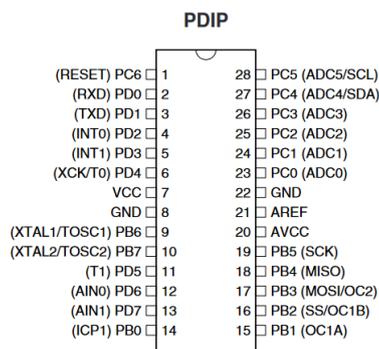


Рисунок 1 – Распиновка микроконтроллера ATmega8A в корпусе DIP-28 [5]

Минимальной тактовой частотой, при которой микроконтроллер успевает обрабатывать USB пакеты, является частота в 12 МГц. Имеется и поддержка больших частот: 15, 16, 16.5, 20 МГц (при этом, на 20 МГц, используемый МК работает нестабильно [7, 8]). В представленной реализации (см. рис. 2) Atmega8A затактирован внешним кварцевым резонатором на 16 МГц. Номинал нагрузочных конденсаторов C1 и C2 выбран равным 22пФ [9]. Правильная и стабильная работа обеспечивается использованием блокировочного конденсатора C4 емкостью 0.1 мкФ совместно с конденсатором большей емкости (конденсатор C3 на 10 мкФ) [10].

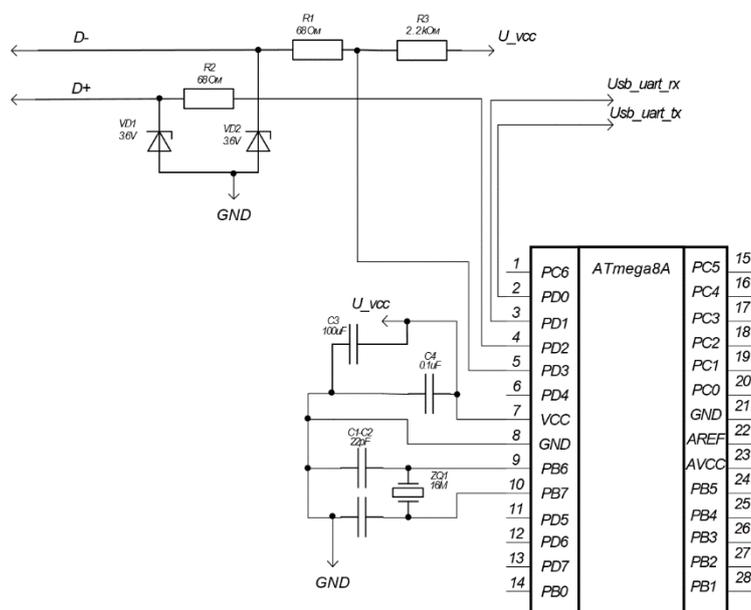


Рисунок 2 – Электрическая схема USB-MIDI преобразователя

По стандарту USB, максимальное напряжение на линиях D+ и D- при логической единице составляет 3.3 В. Поэтому прямое подключение данных линий к микроконтроллеру, питающемуся от 5 В, небезопасно. Для того, чтобы согласовать уровни напряжений использовались стабилитроны на 3.6 В, рекомендованные авторами библиотеки. Типичная схема включения стабилитрона [11] представлена на рисунке 3. Резисторы R1 и R2 номиналами 68 Ом, совместно со стабилитронами, выполняют функции делителей напряжения. При этом они ограничивают максимальный ток через стабилитрон. Номиналы R1 и R2 можно выбрать в пределах 50...200 Ом.

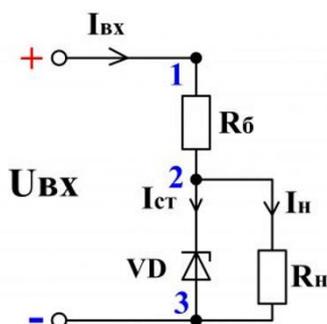


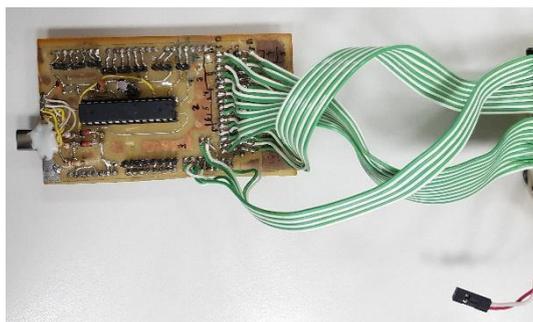
Рисунок 3 – Схема включения стабилитрона [11]

Резистор R3 на 2.2 кОм подтягивает линию D- к питанию 5 В, обеспечивая тем самым в первоначальный момент времени состояние логической единицы. Это необходимо для корректного определения хостом подключенного устройства. Номинал резистора можно выбирать в широких пределах, от одного до сотен кОм.

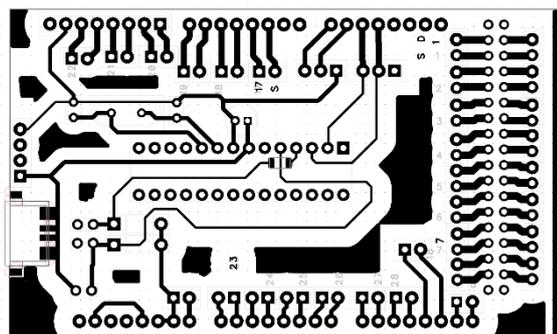
Питание ATmega8A осуществляется посредством USB от источника воспроизведения.

Физически преобразователь USB-MIDI выполнен в виде платы расширения (shield) (см. рис. 4) для удобного подключения к отладочной плате Arduino Mega. Взаимодействие преобразователя с контроллером массива флоппи-приводов происходит по последовательному интерфейсу UART. Физически взаимодействие осуществляется по дорожкам на печатной плате, соединяющим пины RX и TX ATmega8A и пины RX2 и TX2 микроконтроллера ATmega2560, лежащего в основе Arduino Mega.

Плата расширения также является продолжением контроллера массива флоппи-приводов и содержит все необходимые разъемы для подключения FDD, к тому же предусматривает отладочные разъемы для легкого доступа к интерфейсам Hardware UART Serial 2 и Hardware UART Serial 3 Arduino Mega, без необходимости снимать плату расширения.



а



б

Рисунок 4 – Плата расширения: а – вместе с разъемами для подключения FDD, б – проектирование

Плата расширения, изготовленная методом ЛУТ [12], после вытравливания потребовала доработок. Компоненты по возможности были выбраны в выводных корпусах для удобства монтажа.

Для разработки принципиальных схем использовался KiCad. Трассировка плат выполнялась с помощью DipTrace.

Контроллер массива флоппи-приводов.

В флоппи-приводах источником звука является шаговый двигатель, который перемещает считывающую головку и формирует звук. Изменяя скорость шагового двигателя можно изменять частоту извлекаемого звука, т.е. ноты [13]. Таблица соответствия скорости двигателя и ноты была взята из open-source проекта MorphyClassic [14].

Дисководы для 3,5-дюймовых дискет используют четырехконтактный разъем для подключения питания (первые устройства требовали питания от 5 и 12 вольт одновременно, но поздние модели уже использовали только 5 вольт) и 34-контактный разъем с линиями данных и управления (см. рис. 5) [1].

2	Connector Clamp
4	(Spare)
6	Select 3 (NDS3)
8	Index (NINDEX/SECTOR)
10	Select 0 (NDS0)
12	Select 1 (NDS1)
14	Select 2 (NDS2)
16	Drive Motor Enable (NMOTORON)
18	Direction
20	Step (NSTEP)
22	Write Data (NWRITEDATA)
24	Write Gate (NWRITEGATE)
26	Track 00 (NTRK00)
28	Write Protect (NWRITEPROTECT)
30	Read Data (NREADDATA)
32	Side Select (NSIDeselect)
34	Connector Clamp

Рисунок 5 – Назначение пинов 34-контактного разъема FDD [15]

Для создания контроллера массива флоппи-приводов наибольший интерес представляют контакты 10, 18 и 20: контакт 10 отвечает за выбор флоппи-привода (при обычном использовании к одному разъему 34 pin могут подключаться 2 диска одновременно), контакт 18 – выбор направления движения шагового двигателя, контакт 20 – пин исполнения шага двигателем.

Контроллер массива флоппи-приводов реализован на базе отладочной платы Arduino Mega (см. рис. 6), к 30 цифровым выходам которой через плату расширения подключаются 15 флоппи-приводов, разбитых на стойки по 3 штуки в каждой. Данное разбиение было выполнено на основе экспериментально определенных диапазонов частот, характерных для каждой группы флоппи-приводов.

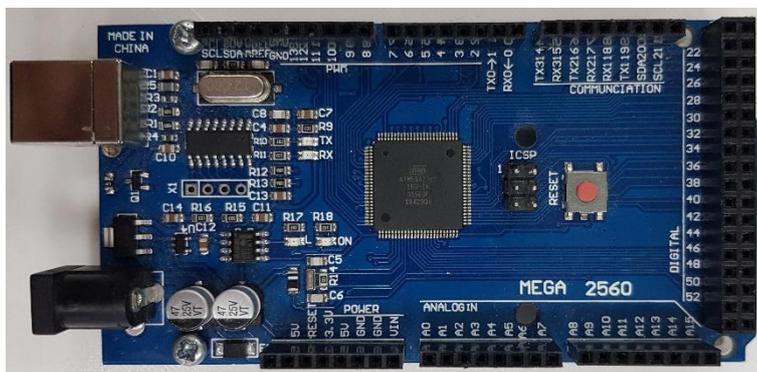


Рисунок 6 – Отладочная плата Arduino Mega

Интерфейс Hardware Serial 2 на плате Arduino Mega используется для подключения преобразователя USB-MIDI. Запитывается Arduino Mega от блока электроснабжения, через соответствующие разъемы.

Схема взаимодействия устройств «Fbox» представлена на рисунке 7.

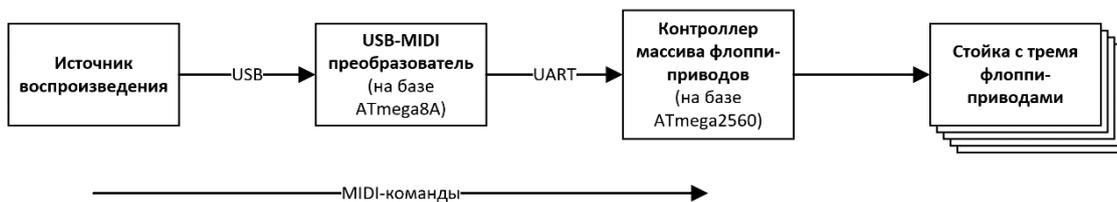


Рисунок 7 – Схема взаимодействия устройств «Fbox»

Блок электроснабжения.

Блок электроснабжения запитывает «Fbox». Напряжение питания унифицировано и составляет 5 В. Примерное потребление тока для каждой составной части «Fbox» в усредненном режиме эксплуатации приведено в таблице 1.

Таблица 1 – Потребление устройств «Fbox» в усредненном режиме эксплуатации.

Замеряемое значение	Значение
Контроллер массива флоппи-приводов, мА	400
Интерфейс USB-MIDI, мА	100, но питается от устройства воспроизведения

Замеряемое значение	Значение
Контроллер массива флоппи-приводов, мА	400
Интерфейс USB-MIDI, мА	100, но питается от устройства воспроизведения
Стойки с FDD, среднее потребление на стойку * количество стоек, мА	800 * 5

Итого, необходим блок питания (далее – БП) с током по линии 5 В не менее 5 А. Ввиду наличия компьютерных БП форм-фактора ATX, было принято решение использовать именно такой экземпляр с внесением в БП минимальных модификаций, не влияющих на его работоспособность. В частности, оставлено подключение только к линии 5 В и выведена кнопка включения БП.

Ко входу блока электроснабжения подключается вышеупомянутый БП, затем напряжение расходится по всему «Fbox» через соответствующие разъемы.

Ввиду простоты, а также планов на дальнейшую его модификацию и объединение с компьютерным БП, для реализации (см. рис. 8) была использована обычная паечная макетная плата со стандартным шагом отверстий 2.54 мм. Блок электроснабжения комплектуется набором проводов и переходников, необходимых для сопряжения с другими блоками и блоком питания. Основные использованные разъемы – также с шагом 2.54 мм.

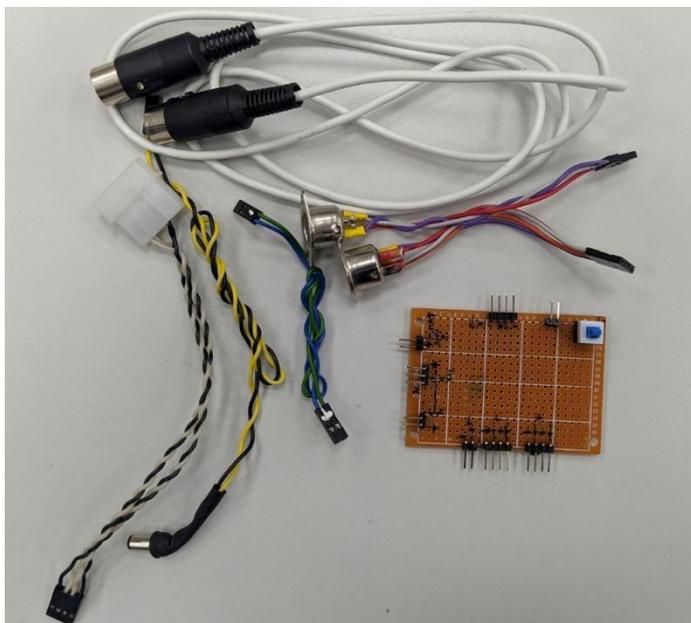


Рисунок 8 – Блок электроснабжения, без блока питания

Разработка корпуса.

У флоппи-привода есть особенность: шаговый двигатель не способен менять громкость исполняемой ноты. Павел Задрожняк в оригинальном флоппотроне [3] управляет громкостью проигрываемых нот путем увеличения числа флоппи-приводов. Представленная реализация также пользуется данным приемом: партию каждого музыкального инструмента исполняют три флоппи-привода. Еще одна особенность флоппи-привода: не все шаговые двигатели жужжат одинаково.

Предварительно отобранная тройка флоппи-приводов физически размещается в стойке, распечатанной на 3D-принтере (см. рис. 9). Поэтапная разработка 3D-моделей велась с помощью ПО Autodesk Fusion 360.

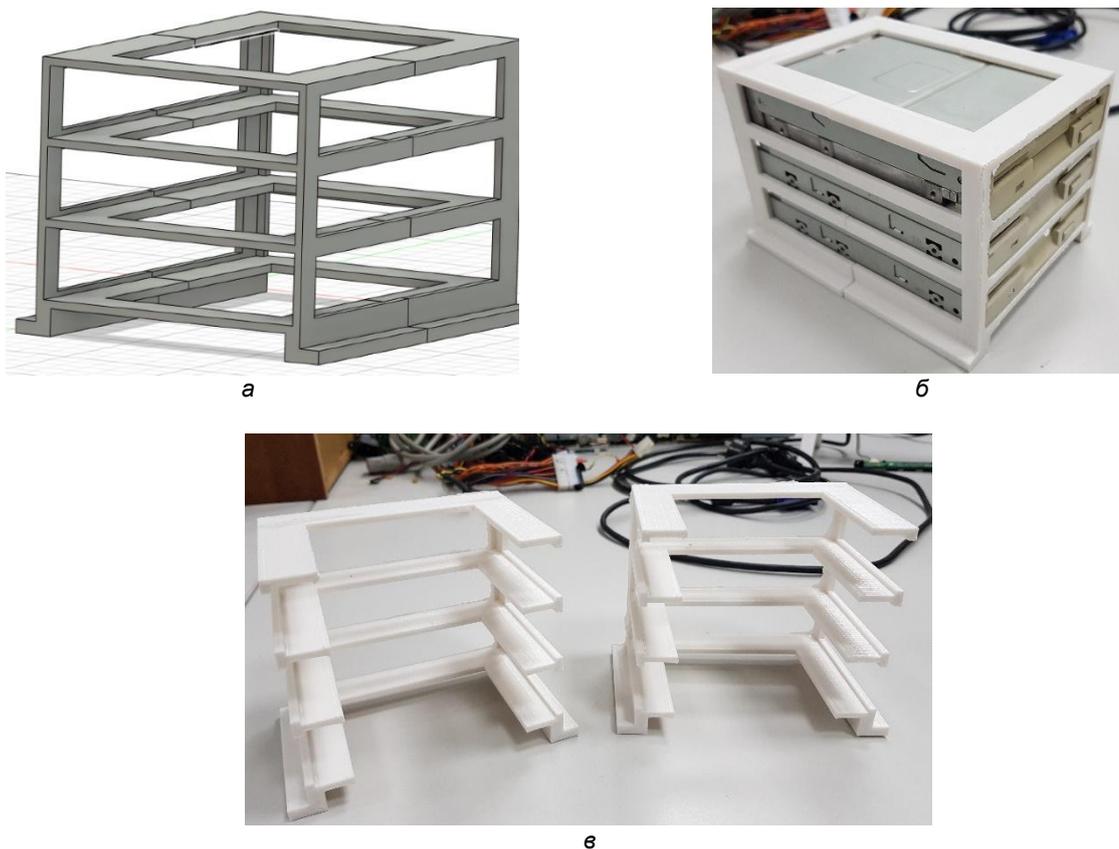


Рисунок 9 – Стойка для флоппи-приводов: а – 3D-модель; б – собранная стойка; в – результат 3D-печати

Всего таких стоек было распечатано пять штук. На рисунке 10 по середине видна первая версия стойки. От данного дизайна было принято решение отказаться для уменьшения времени печати (наиболее критичный параметр) и улучшения акустических характеристик (ввиду минимизации препятствий для звука). При этом надежность стойки (с вставленными флоппи-приводами) не пострадала, а затраты времени на печать и расход пластика (в метрах) уменьшились (с 14 часов 22 минут до 6 часов 56 минут, с 61.75 до 26.50 метров пластика).



Рисунок 10 – 15 флоппи-приводов, размещенных в 5 стойках

Впоследствии конструкция стойки еще немного упростилась (см. рис. 11): на практике выяснилось, что придерживать верхний флоппи-привод по всему периметру нет необходимости.



Рисунок 11 – Финальная версия стойки для флоппи-приводов

Модульности разработки (см. рис. 12) невозможна без унификации как корпуса для стоек, так и корпуса с контроллером массива флоппи-приводов и блоком электроснабжения. Унифицированный корпус минимизирован с оглядкой на габариты стойки с флоппи-приводами. Модульности также предполагает стандартизацию способа фиксации в корпусе как стоек с флоппи-приводами, так и блока электроснабжения вместе с контроллером массива флоппи-приводов. Таким стандартным решением стали прямоугольные пазы в профиле из PLA-пластика, также используемые для фиксации стенок корпуса.

Можно отметить стенки с перфорацией в виде пчелиных сот. Данное решение призвано минимизировать препятствия для звука. Дизайн стенок с перфорациями также позволяет уменьшить время печати и расход пластика за счет уменьшения сопротивления скручиванию. Однако стенки не подвергаются таким деформациям в составе корпуса.

Идея модульности предусматривает свободное перемещение блоков. Для взаимной фиксации корпусов предусмотрены пазы в виде равнобедренной трапеции.

В отличие от боковых, передняя и задняя стенки закрепляются с помощью М3 винтов. Для предупреждения слизывания резьбы отверстия в профиле из PLA-пластика используются резьбовые втулки (см. рис. 13). Их использование предусмотрено при 3D-моделировании.

Итого корпус представляет собой четыре профиля с пазами двух видов и стенок (боковые с перфорацией, передняя с вырезами чуть меньшими, чем флоппи-привод, задняя с вырезами под шильдик разъемов (англ. IO shield)) (см. рис. 14а). Два верхних профиля соединены перемычкой для предотвращения прокручивания, а два нижних, помимо этого, содержат упор для стойки с флоппи-приводами или кареток с блоком электроснабжения или контроллером массива флоппи-привода, совмещенного с USB-MIDI преобразователя (см. рис. 14б).



а



б



в

Рисунок 12 – Размещенные в корпусе: *а* – контроллер массива флоппи-приводов и блок электроснабжения;
б – стойка с флоппи-приводами; *в* – демонстрация модульности корпуса

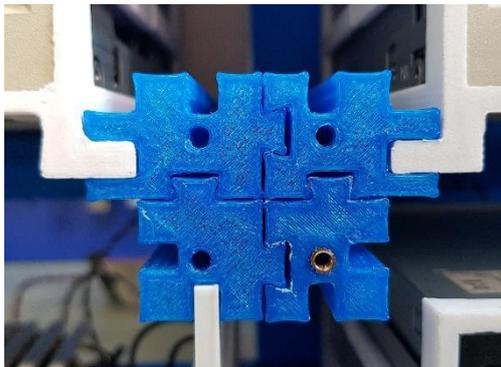


Рисунок 13 – Демонстрация взаимной фиксации корпусов и применения резьбовых втулок

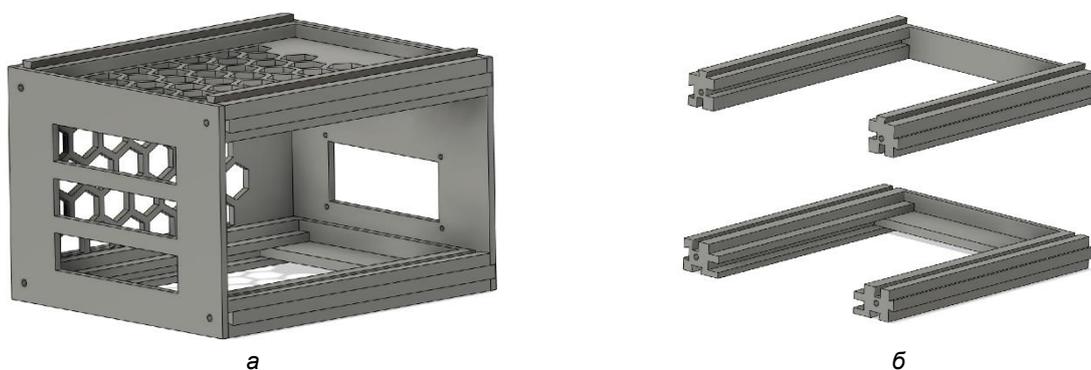


Рисунок 14 – 3D-модель: *а* – унифицированный корпус; *б* – основа корпуса

По состоянию на начало апреля 2022 года готово 4 из 6 блоков: блок с контроллером массива флоппи-приводов и блоком электроснабжения и 3 блока со стойками из флоппи-приводов (см. рис. 15).



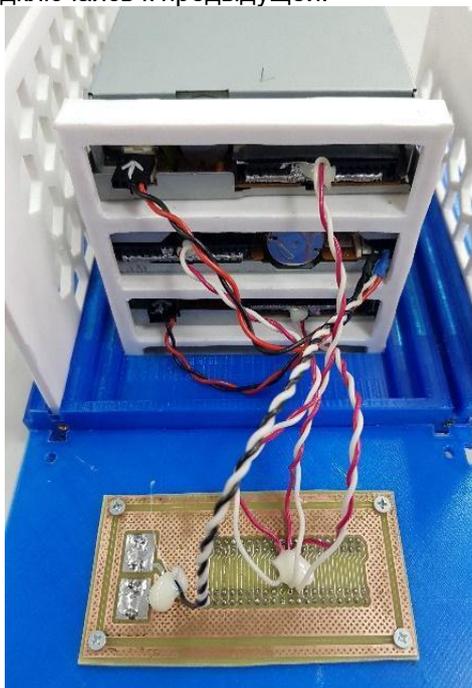
Рисунок 15 – 4 из 6 блоков «Fbox»

Разработка интерфейсных плат.

Для реализации идеи модульной структуры необходимо было продумать подключение стоек с флоппи-приводами к контроллеру массива флоппи-приводов и блоку электроснабжения, которые размещены в одном корпусе. Для этого разработаны интерфейсные платы (см. рис. 16а), несущие на себе необходимые разъемы. Данные платы предполагается монтировать в корпуса с стойками (см. рис. 16б), а подключение к главному блоку (корпус, в котором размещены контроллер массива флоппи-приводов и блок электроснабжения) осуществлять стандартными шлейфами PATA 40pin (см. рис. 16в), которые обычно применяются для соединения жестких дисков. Данные кабеля имеют достаточное количество контактов, а также являются достаточно распространенными. При этом, для подключения каждой стойки выделяется по 6 контактов (для адресного управления

каждым флоппи-приводом отдельно). При этом задумывалось, что каждая стойка может подключаться к предыдущей (в виде цепи), передавая сигналы от неиспользуемых контактов дальше. Поэтому имеется 2 разъема: входной и выходной.

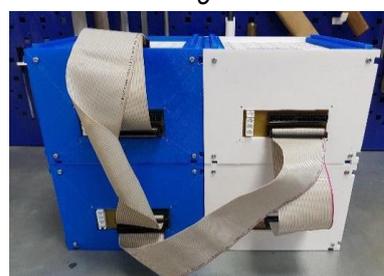
Питание стоек осуществляется по схожему принципу: питание для следующей стойки можно брать, подключаясь к предыдущей.



а



б



в

Рисунок 16 – Интерфейсная плата: а – подключенная к флоппи-приводам; б – вмонтированная в задние стенки корпусов; в – взаимное подключение блоков с помощью шлейфов PATA 40pin

Заключение.

В результате выполнения работы был изготовлен MIDI-секвенсер на базе флоппи-приводов (см. рис. 15). Преимуществом представленного рабочего образца является модульность конструкции. Проект «Fbox» был представлен на Дне открытых дверей ФКСИС.

Данная работа выполнялась на базе студенческого инженерно-образовательного центра кафедры ЭВМ БГУИР.

Список использованных источников:

1. Мелодия для флоппитрона без оркестра [Электронный ресурс]. – Режим доступа: <https://nplus1.ru/material/2020/03/10/singingFDDs>. – Дата доступа: 04.04.2022.
2. Evil floppy drives – English translation [Электронный ресурс]. – Режим доступа: <http://silent.org.pl/home/2011/09/29/evil-floppy-drives-english-translation/>. – Дата доступа: 04.04.2022.
3. Return of the Floppies [Электронный ресурс]. – Режим доступа: <http://silent.org.pl/home/2016/07/06/return-of-the-floppies/>. – Дата доступа: 04.04.2022.
4. MIDI [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/wiki/MIDI>. – Дата доступа: 04.04.2022.
5. ATmega8/L datasheet [Электронный ресурс]. – Режим доступа: https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-2486-8-bit-AVR-microcontroller-ATmega8_L_datasheet.pdf. – Дата доступа: 04.04.2022.
6. V-USB - A Firmware-Only USB Driver for Atmel AVR Microcontrollers [Электронный ресурс]. – Режим доступа: <https://www.obdev.at/products/vusb/index.html>. – Дата доступа: 04.04.2022.
7. ATmega8 умер генератор? - AVR - Форум по радиоэлектронике [Электронный ресурс]. – Режим доступа: <https://forum.cxem.net/index.php?topic/187340-atmega8-%D1%83%D0%BC%D0%B5%D1%80-%D0%B3%D0%B5%D0%BD%D0%B5%D1%80%D0%B0%D1%82%D0%BE%D1%80/>. – Дата доступа: 04.04.2022.
8. V-USB and HIDKEYS example working with 20MHZ? | AVR Freaks [Электронный ресурс]. – Режим доступа: <https://www.avrfreaks.net/forum/v-usb-and-hidkeys-example-working-20mhz>. – Дата доступа: 04.04.2022.
9. Параметры кварцевых резонаторов [Электронный ресурс]. – Режим доступа: http://www.gaw.ru/html.cgi/txt/el_mech/GEYER_EL/parametr.htm. – Дата доступа: 04.04.2022.
10. Чистое питание для каждой микросхемы, часть 2: Выбор и использование блокировочных конденсаторов [Электронный ресурс]. – Режим доступа: <https://radiopro.ru/post/461>. – Дата доступа: 04.04.2022.
11. Принцип работы и маркировка стабилитронов * diodov.net [Электронный ресурс]. – Режим доступа: <https://diodov.net/stabilitrn-printsip-raboty-i-markirovka-stabilitrnov/>. – Дата доступа: 04.04.2022.

12. Лазерно-утюжная технология [Электронный ресурс]. – Режим доступа: <https://chem.net/master/45.php>. – Дата доступа: 04.04.2022.
13. Новая жизнь дискет: что такое флопотрон и как его сделать [Электронный ресурс]. – Режим доступа: <https://tproger.ru/articles/floppotron-what-is-it-and-how-to-make/>. – Дата доступа: 04.04.2022.
14. Sammy1Am/MoppyClassic: Moppy has been replaced with Moppy 2.0! [Электронный ресурс]. – Режим доступа: <https://github.com/Sammy1Am/MoppyClassic>. – Дата доступа: 04.04.2022.
15. Распиновка 5 1/4 inch floppy [Электронный ресурс]. – Режим доступа: https://old.pinouts.ru/HD/5_12_floppy.shtml. – Дата доступа: 04.04.2022.

«FBOX» – MODULAR MIDI SEQUENCER BASED ON FLOPPY DRIVES

Kaliutchyk A.¹

Yaschenko V.¹

Belarusian State University of Informatics and Radioelectronics¹, Minsk, Republic of Belarus

Pertsau D. – PhD in Computer Science

Annotation. This work is the result of the development of a course project in the discipline "Circuit Engineering". The purpose of this work is to demonstrate the operation of floppy drives as a MIDI sequencer. Distinctive features of the presented solution are modularity and scalability.

Keywords. MIDI, sequencer, USB, Floppotron, ATmega, Arduino.

КОМПЬЮТЕРЫ НА ОСНОВЕ ТЕРНАРНОЙ ЛОГИКИ И ПЕРСПЕКТИВЫ ИХ РАЗВИТИЯ

Шнип А.Д., Кассирова А.Д., студенты гр.024401, Селезнёв И.Л.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Селезнёв И.Л. – доц., канд. тех. наук

Аннотация. В данной статье рассматривается возможность создания троичного компьютера, дается определение тернарной логики, затрагиваются существующие реализации троичных электронно-вычислительных машин. Анализируются перспективы применения тернарной логики в оптических и квантовых компьютерах.

Ключевые слова. Тернарная логика, троичные компьютеры, ЭВМ «Сетунь», тернарный оптический компьютер, квантовый компьютер, архитектура, оперативная память, процессор, особенности программирования.

Троичные компьютеры представляют собой электронно-вычислительные машины, основанные на тернарной логике. Они являются потенциально более мощным средством обработки информации, поскольку могут оперировать большими объемами данных по сравнению с двоичными компьютерами.

Тернарная логика представляет собой систему, основанную на трех утверждениях: «Нет», «Да», «Неизвестно» или 0, 1, 2. Вариант обозначения -1, 1, 0 принято называть симметричным. Тернарная логика включает в себя все возможности бинарной и обеспечивает ряд преимуществ:

- плотность записи информации выше в 1,58 раза;
 - наличие положительной и отрицательной цифр в тернарной симметричной логике позволяет представлять числа любого знака;
 - округление чисел с плавающей точкой путем отсечения младших разрядов;
 - обработка в 2,85 раз больше информации за один такт.
- Троичные компьютеры имеют свои аналоги единиц измерения информации:
- трит – троичный разряд в троичной системе счисления, аналог бита в двоичной системе;
 - трайт – минимальная адресуемая единица в памяти троичного компьютера, аналог байта.
- Один трайт равен шести тритам (возможно равенство трем или девяти тритам).

Основные проблемы построения троичных компьютеров связаны со способом хранения информации и построением специализированного процессора. Вышеперечисленные преимущества применения тернарной логики обуславливают интерес к созданию троичных компьютеров. Одним из возможных решений названных проблем может стать применение оптических и квантовых технологий [1].

Первой в мире полноценной реализацией компьютера, основанного на тернарной логике, стала машина «Сетунь», созданная под руководством советского ученого Н. П. Брусенцова в 1959 году. В таблице 1 представлены сводные характеристики электронно-вычислительной машины «Сетунь» и компьютера «IBM 7090», являвшегося ее современником, основанным на бинарной логике.

Таблица 1 – Характеристики ЭВМ «Сетунь» и IBM 7090

ЭВМ	Тактовая частота	Оперативная память	Объем жесткого диска	Число операций в секунду	Пропускная способность шины памяти
Сетунь	200 кГц	162 трайта (192 байта)	4000 трайт (4740 байт)	4000	54 трайта (64 байта)
IBM 7090	500 кГц	32 кБ	143 кБ	100000	-

Всего было выпущено порядка 50 экземпляров троичного компьютера «Сетунь», однако вскоре производство прекратилось. Одной из основных причин стала малая производительность троичного компьютера по сравнению с двоичными аналогами. Также можно отметить, что объем оперативной памяти IBM 7090 превышал объем оперативной памяти ЭВМ «Сетунь» в 166,67 раз.

Особенности программирования троичного компьютера

Программирование компьютеров, основанных на тернарной логике, имеет ряд существенных особенностей. Так, для реализации оператора троичного ветвления в двоичных машинах придется использовать две проверки и соответственно две команды двоичного ветвления. В троичном компьютере потребуется всего один оператор, имеющий сразу три ветвления -1, 0, 1, что сокращает объем кода и увеличивает производительность программы. Выигрыш, обеспечиваемый

применением команд троичного ветвления, становится еще более ощутимым с возрастанием количества анализируемых вариантов продолжения программы.

Необходимое количество команд для реализации ветвления равно $\log_n P$, округленного до целого числа, где P – число вариантов, n – система счисления. Очевидно, что для любого P :

$$\log_3 P < \log_2 P. \quad (1)$$

Также следует отметить, что для обеспечения реализации цикла с условием на двоичной машине потребовалось бы использовать две проверки: одну для прекращения цикла, вторую для выбора одной из частей цикла. На троичной машине это реализуется с помощью одной команды, которая проверяет, необходимо ли завершать цикл, и при этом тут же выбирает одну из двух процедур в качестве тела цикла на случай его продолжения [2].

Троичный компьютер показывает наибольшую эффективность в задачах, которые требуют множественных сравнений величин или имеют элемент неопределенности. Например, это задачи по обработке изображений, цвет которых хранится в трех каналах, или задачи по расчету пространственных координат.

Существующие реализации элементов троичного компьютера

Примером реализации оперативного запоминающего устройства (ОЗУ) троичного компьютера является ОЗУ в ЭВМ «Сетунь». Оно выполнено на основе ферритовых сердечников по схеме с прямым выбором. Устройство состоит из двух блоков:

- ферритовый куб, состоящий из 18 плат с запоминающими сердечниками и координатными трансформаторами, а также формирователи импульсов для обращения к кубу;
- блок управления кубом, состоящий из регистра адреса, дешифраторов принимаемых в этот регистр кодов, усилителей считывания, схемы, формирующей импульс, 6-разрядного регистра ОЗУ и схемы управления этим регистром.

Также имеется «панель куба», на которой размещены ключи, позволяющие заблокировать прием кода операции и прием кода в регистр адреса и осуществить периодическое выполнение принятой команды, например для отладки ОЗУ и других устройств.

Для хранения троичных разрядов 1, 0, -1 используется два ферритовых сердечника. Для хранения состояний 1 и -1 сердечники намагничиваются в противоположных направлениях. При хранении 0 внутренние и внешние слои каждого сердечника намагничены в различных направлениях так, что суммарный поток в каждом из них близок к нулю.

Вариантом реализации центрального процессора (ЦП) троичного компьютера является процессор ЭВМ «Сетунь-70». Он оборудован памятью магазинного типа, продолжением которой служит основная память компьютера. Роль магазина при этом может выполнять любая страница оперативной памяти. ЦП состоит из операционного устройства и устройства управления. Операционное устройство осуществляет выполнение арифметических и логических операций, используя в качестве магазина (стека) страницы оперативной памяти. Устройство управления включает в себя все основные регистры процессора, которые определяют режим работы компьютера. Основным элементом логических схем центрального процессора является магнитный усилитель, питаемый импульсами тока.

Перспективы реализации элементов троичного компьютера

Использование тернарной логики в электронно-вычислительных машинах дает выигрыш в оптимизации алгоритмов вычисления, однако не обеспечивает достаточной скорости обработки информации. Из-за ограничений технических возможностей электрического элемента, троичные компьютеры не смогут использовать весь потенциал тернарной логики. Принципиально новые решения проблемы дает использование оптических компьютеров, основанных на жидкокристаллических элементах, способных обеспечить необходимую скорость обработки данных.

Тернарный оптический компьютер (ТОК), использующий волоконную оптику и несимметричную тернарную логику, интерпретирует отсутствие света как 0 и две ортогональные поляризации света как 1 и 2.

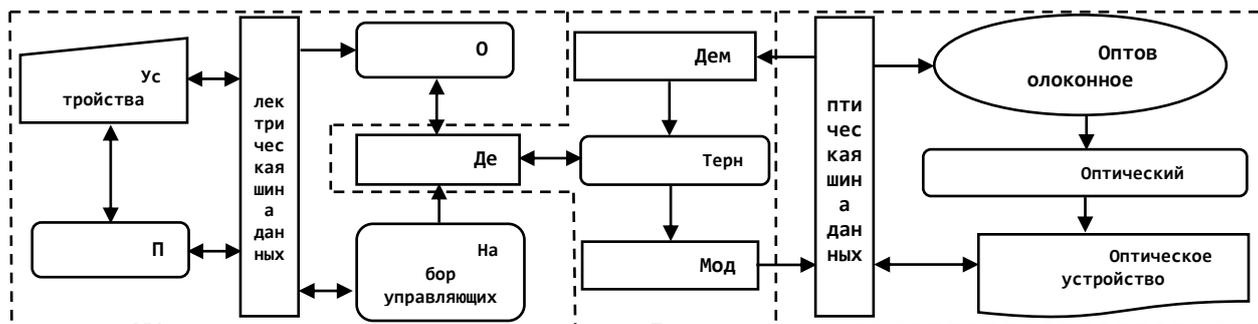


Рисунок 1 – Схема устройства оптического тернарного компьютера

Архитектура процессора ТОК, представленная на рисунке 1, состоит из оптического операционного устройства (ООУ), электрического устройства управления (ЭУУ) и контроллера данных (КД).

Оптическое операционное устройство состоит из оптической шины данных, оптоволоконного кольца, выступающего в качестве кэша, оптического элемента операций и оптического устройства ввода/вывода. В ООУ поступает оптический сигнал из модулятора контроллера данных.

Электрическое устройство управления включает в себя электрическую шину данных, набор управляющих процессоров, ПЗУ и ОЗУ, а также устройства ввода/вывода. Набор процессоров ЭУУ управляет программами, хранящимися в ПЗУ, и рассылает необработанные электрические управляющие сигналы всем элементам компьютера.

Контроллер данных состоит из дешифратора, тернарного ОЗУ, модулятора и демодулятора. Дешифратор предназначен для перевода данных из двоичной системы в троичную и наоборот. Модулятор преобразует электрический сигнал тернарного ОЗУ в оптический сигнал и посылает его в оптическую шину данных ООУ. Демодулятор конвертирует оптический сигнал в электрический. Тернарное ОЗУ состоит из ряда полупроводниковых микросхем и используется для хранения тернарных кодов данных, световых сигналов разной полярности и данных чувствительных световых элементов, получающих поляризованные сигналы.

На рисунке 2 представлена схема сумматора ТОК.

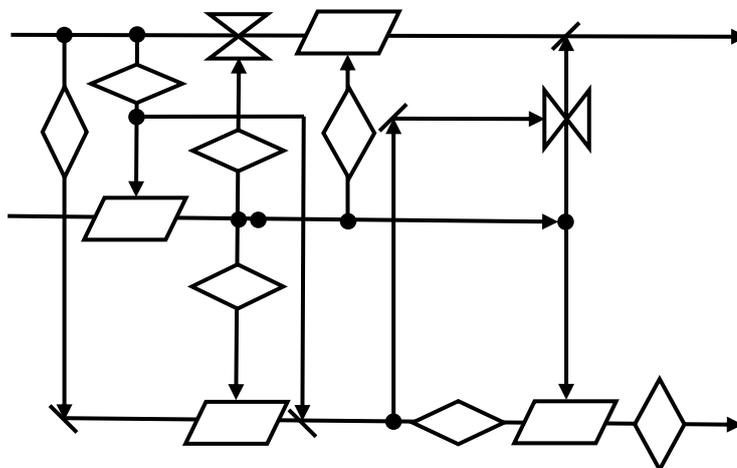


Рисунок 2 – Сумматор тернарного оптического компьютера

Сумматор основан на следующих элементах:

- жидких кристаллах (ЖК1...4), предназначенных для изменения поляризации света при воздействии на него другим сигналом;
- вертикальных (ВП1...3) и горизонтальных (ГП1...4) поляризаторах, по которым может проходить только вертикально поляризованный или горизонтально поляризованный свет соответственно;
- оптических блокираторов (ОБ1...2), препятствующих прохождению сигнала через него при воздействии другого сигнала.

Устройство работает с данными на входах *a* и *б*, результат операции передается на выход *в*. Если результат сложения выходит за пределы разряда, организуется его перенос в выход *г* [3].

Тернарная логика применима и в квантовых компьютерах. Они используют явления квантовой механики и основывают свои вычисления на кубитах – аналогах бита в обычных компьютерах. Наиболее важный для квантовых компьютеров управляемый клапан Тоффли, который способен

выполнить любую из операций NOT, AND и XOR, можно существенно упростить, используя тернарную логику. Для построения подобного вентиля с n управляющими кубитами на основе бинарной логики потребовалось бы $12n - 11$ двухкубитных вентилях и $n - 1$ вспомогательных кубита. Использование тернарной логики требует не более $2n - 1$ двухкубитных вентилях без использования вспомогательных кубит [4].

На сегодняшний день в открытых источниках не упоминается об использовании троичных компьютеров. Однако, перспективными направлениями в применении тернарной логики являются разработка квантовых и оптических компьютеров на основе троичной системы счисления. Троичные компьютеры могут использоваться в ситуациях, где требуется большое количество вычислений, например, прогнозирование погоды, предсказание землетрясений, сложная обработка графики и изображений, поиск в больших базах данных, взлом шифров и т.д. Помимо этого, одно из возможных применений троичных устройств – специализированные сопроцессоры в суперкомпьютерах, помогающие в сложных математических расчетах.

Список использованных источников:

1. История создания троичного компьютера / Н.А. Краснянский // Молодой исследователь Дона, 2021.
2. Особенности программирования троичной машины: новые возможности и новые задачи / А.А. Бурцев // Труды научно-исследовательского института системных исследований Российской академии наук, 2020. – С.49-60.
3. Ternary Optical Computer Architecture / Jin Yi [et al.] // School of Computer and Technology, Shanghai University, 2005.
4. Simplifying quantum logic using higher-dimensional Hilbert spaces / Benjamin P. Lanyon [et al.] // Nature Physics, 2008.

UDC 004.382

COMPUTERS BASED ON TERNARY LOGIC AND PROSPECTS FOR THEIR DEVELOPMENT

Shnip A.D., Kassirova A.D., Seleznev I.L.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Seleznev I.L. – Associate Professor, candidate of technical sciences

Annotation. This article discusses the ability to build a ternary computer, defines three-valued logic, discusses existing implementations of ternary computers. The prospects for the use of three-valued logic in optical and quantum computers are analyzed.

Keywords. Ternary logic, ternary computer, computer "Setun", ternary optical computer, quantum computer, architecture, random access memory, processor, programming features.

ВИДЫ И КЛАССИФИКАЦИЯ ИНФОРМАЦИОННЫХ СИСТЕМ

Туровец Н.О., Ващиллов А.Д., Марков А.Н., Мигалевич С.А.

*Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь*

*Научный руководитель: Нестеренков С.Н. – канд.техн.наук, доцент кафедры ПОИТ,
декан факультета компьютерных систем и сетей Белорусского государственного
университета информатики и радиоэлектроники*

Аннотация. Проведена классификация информационных систем с целью выбора одной из них для выполнения нагрузочного тестирования на эту систему. Установлено, что существует большое число разных методов и подходов к представлению классификации информационных систем.

Ключевые слова: информационные системы, классификация, признаки, выбор.

Информационная система типичной современной организации является весьма сложным образованием, построенным в многоуровневой архитектуре клиент-сервер, которое пользуется многочисленными внешними сервисами и, в свою очередь, предоставляет собственные сервисы. Современная информационная система представляет собой использование персонального компьютера в качестве поиска и переработки информации [1].

Рассмотрим существующие классификации информационных систем по некоторым признакам.

Первая классификация вводится исходя из трех существующих уровней управления: высшего, среднего и низшего. Выделяются три класса систем, обеспечивающие решение задач, возникающих на различных уровнях управления [2]:

- системы, обеспечивающие работу высшего уровня управления; в их задачу входит обеспечение стратегических конкурентоспособных преимуществ;
- системы, обеспечивающие работу среднего уровня управления; в их задачу входит обеспечение принятия управленческих решений;
- системы, обеспечивающие работу низшего уровня управления; в их задачу входит обеспечение текущих бизнес-операций.

В информационных системах среднего уровня выделяют три вида [2]:

- системы предоставления информации – предоставляют менеджерам предопределенные и регламентированные сообщения (отчеты, справки) о текущих бизнес-операциях;
- системы поддержки принятия решений – в их задачу входит предоставление менеджерам набора диалоговых и специальных средств проектирования альтернатив для использования в обеспечении процесса принятия решений;
- информационная система руководителей – обслуживают менеджеров высшего уровня с целью мгновенного формирования критической информации, а также информируют о состоянии проектов по ключевым факторам.

В классе информационных систем низшего уровня также можно выделить три вида [2]:

- системы обработки операций – выполняют записи об операциях и сделках, модернизируют базы данных, обрабатывают данные, выводят разнообразные отчеты и справки;
- системы управления процессами – управляют физическими процессами производства продукции, подачи товаров, материалов и т.п.;
- системы автоматизации офиса – поддерживают коммуникации, делопроизводство и производительность офиса.

В высшем классе информационных систем не проводится выделения различных видов информационных систем [2].

Данная классификация показывает роль информационных систем в обеспечении процесса управления и в обеспечении информационных систем решения различных задач в зависимости от уровня управления.

Вторая классификация информационных систем вводится исходя из области их применения [3]:

- информационные системы для организации (маркетинговая информационная система, финансовые, кадровые и др. информационные системы);
- информационная система для анализа заданной предметной области (информационная система для анализа текущего состояния малых предприятий в РБ, информационная система для мониторинга загрязнения окружающей среды);
- информационная система для сбора информации только на продажу (юридические и адресные базы данных).

Данная классификация показывает особенности работы информационной системы с информацией.

Еще одна классификация, в которой также отражены особенности работы информационной системы с информацией [4]:

- системы справочно-информационного обслуживания;
- справочно-библиографические системы;
- системы выборочного распространения информации.

Представленная классификация рассматривает только узкий аспект, характеризующий особенности распространения информации информационной системой.

Следующая классификация вводится исходя из существующих задач, которые обеспечивают информационные системы, и уровней управления, на которых они применяются:

- корпоративные информационные системы;
- системы оперативного управления и учета;
- аналитические информационные системы.

Корпоративные информационные системы обеспечивают интегрированное решение задач управления предприятием и поддержку принятия управленческих решений на основе автоматизации процессов, процедур и способов осуществления деятельности корпорации.

Системы оперативного управления и учета поддерживают, главным образом, оперативный уровень управления и учета. Среди этих систем существует деление по типам ресурсов и объектам, управление которыми они обеспечивают:

- бухгалтерский учет (финансы);
- бюджетирование (управленческий учет финансов);
- торговые операции (продукция, расчеты с контрагентами);
- складской учет (материалы и готовая продукция);
- кадровый учет (персонал);
- управление документами (информационные ресурсы);
- управление проектами (инновационная деятельность);
- управление производством (технологические процессы).

Аналитические информационные системы поддерживают стратегический уровень управления. Обеспечивают подготовку аналитической информации и отчетности, поддержку принятия решений. Выделяются две группы таких продуктов [4]:

- интегрированные системы, выполняющие обработку и анализ больших объемов информации, содержащихся в базах данных корпоративной информационной системы;
- тиражируемые аналитические системы. Данная классификация не позволяет решить проблему четкого сегментирования существующих информационных систем.

На рисунке 3 представлена классификация информационных систем по некоторым признакам.

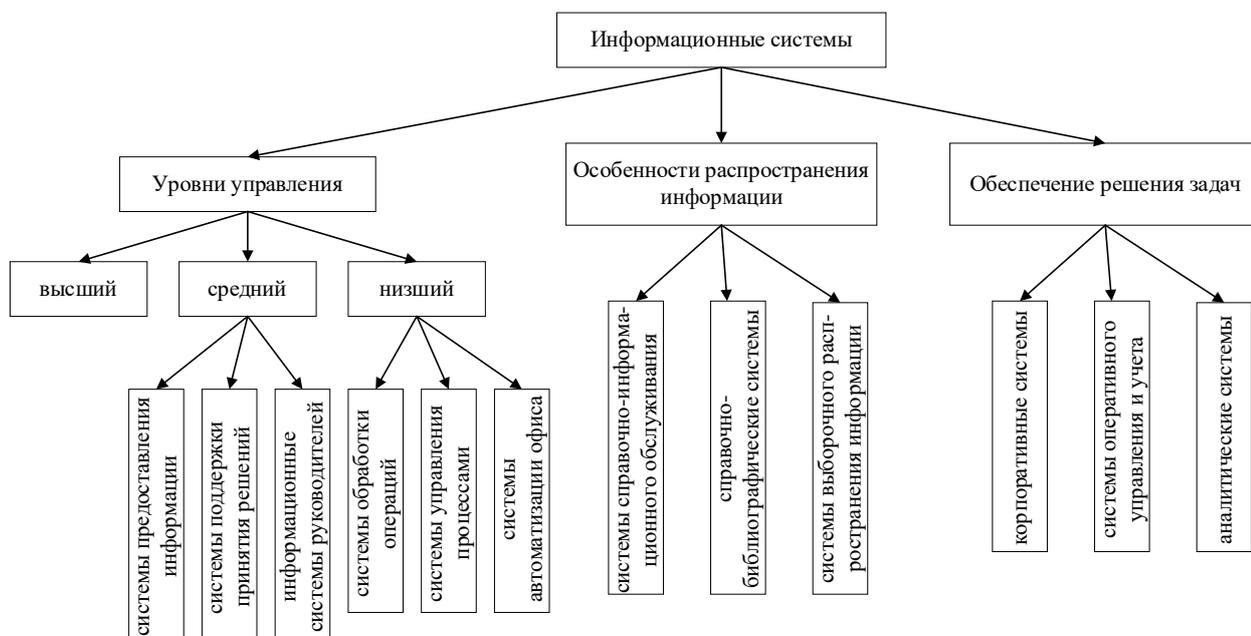


Рисунок 3 – Классификация информационных систем по некоторым признакам

Введение подобной классификации информационных систем позволило бы разделить их по типам и сферам применения. На сегодняшний день это важно для решения практических задач, когда новые информационные системы становятся массовыми и нужно провести их сравнение.

Кроме того, информационные системы можно классифицировать в соответствии с Приложением 1 к Положению о порядке технической и криптографической защиты информации в информационных системах, предназначенных для обработки информации, распространение и (или) предоставление которой ограничено (в редакции приказа Оперативно-аналитического центра при Президенте Республики Беларусь 12.11.2021 № 195) [5]:

- класс 6-частн – негосударственные информационные системы, в которых обрабатывается общедоступная информация (в том числе общедоступные персональные данные) и которые не имеют подключений к открытым каналам передачи данных.

- класс 6-гос – государственные информационные системы, в которых обрабатывается общедоступная информация (в том числе общедоступные персональные данные) и которые не имеют подключений к открытым каналам передачи данных.

- класс 5-частн – негосударственные информационные системы, в которых обрабатывается общедоступная информация (в том числе общедоступные персональные данные) и которые подключены к открытым каналам передачи данных.

- класс 5-гос – государственные информационные системы, в которых обрабатывается общедоступная информация (в том числе общедоступные персональные данные) и которые подключены к открытым каналам передачи данных.

- класс 4-ин – информационные системы, в которых обрабатываются персональные данные, за исключением специальных персональных данных, и которые не имеют подключений к открытым каналам передачи данных.

- класс 4-спец – информационные системы, в которых обрабатываются специальные персональные данные, за исключением биометрических и генетических персональных данных, и которые не имеют подключений к открытым каналам передачи данных.

- класс 4-бг – информационные системы, в которых обрабатываются биометрические и генетические персональные данные и которые не имеют подключений к открытым каналам передачи данных.

- класс 4-юл – информационные системы, в которых обрабатывается информация, составляющая коммерческую и иную охраняемую законом тайну юридического лица, распространение и (или) предоставление которой ограничено (за исключением сведений,

составляющих государственные секреты, и служебной информации ограниченного распространения), и которые не имеют подключений к открытым каналам передачи данных.

– класс 4-дсп – информационные системы, в которых обрабатывается служебная информация ограниченного распространения и которые не имеют подключений к открытым каналам передачи данных.

– класс 3-ин – информационные системы, в которых обрабатываются персональные данные, за исключением специальных персональных данных, и которые подключены к открытым каналам передачи данных.

– класс 3-спец – информационные системы, в которых обрабатываются специальные персональные данные, за исключением биометрических и генетических персональных данных, и которые подключены к открытым каналам передачи данных.

– класс 3-бг – информационные системы, в которых обрабатываются биометрические и генетические персональные данные и которые подключены к открытым каналам передачи данных.

– класс 3-юл – информационные системы, в которых обрабатывается информация, составляющая коммерческую и иную охраняемую законом тайну юридического лица, распространение и (или) предоставление которой ограничено (за исключением сведений, составляющих государственные секреты, и служебной информации ограниченного распространения), и которые подключены к открытым каналам передачи данных.

– класс 3-дсп – информационные системы, в которых обрабатывается служебная информация ограниченного распространения и которые подключены к открытым каналам передачи данных.

Такая схема классификации информационных систем поможет проводить детальное изучение работы систем и оценить эффективность выбранного вида для конкретной реализации путем проведения тестирования.

Список использованной литературы

1. Информационные системы [Электронный ресурс]. – Режим доступа: https://www.elibrary.ru/download/elibrary_24105935_43594491.pdf
2. Устимова, Г. М. Информационные системы менеджмента / Г. М. Устимова. – СПб : ДиаСофтЮп, 2000. – 357 с.
3. Концепция информационного обеспечения МП: справочник / А. А. Михайлов [и др.]. – М. : 1996. – 153 с.
4. Вовчак И.С. Информационные системы и компьютерные технологии в менеджменте: / Вовчак И.С – Т.: ТНТУ им. И.Пулюя, 2001. – 354 с
5. Приказ оперативно-аналитического центра при президенте Республики Беларусь 12 ноября 2021 г. № 195.

UDC 681.518.3

TYPES AND CLASSIFICATION OF INFORMATION SYSTEMS

Turovets N.O., Vashchylau A.D., Markov A.N., Migalevich S.A.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus (style T-institution)

Nesterenkov S.N. – PhD, associate professor, Dean of the Faculty of Computer Systems and Networks

Annotation. A survey and classification of information systems is carried out in order to select one of them for performing load testing on this system. It has been established that there are a large number of different methods and approaches to presenting the classification of information systems.

Keywords. information systems, classification, signs, choice.

ПРЕДВАРИТЕЛЬНАЯ ОБРАБОТКА ИЗОБРАЖЕНИЙ РУКОПИСНЫХ ПОДПИСЕЙ ДЛЯ СИСТЕМ РАСПОЗНАВАНИЯ

Ахунджанов У.Ю., аспирант

Объединенный институт проблем информатики Национальной академии наук Беларуси
г. Минск, Республика Беларусь

Старовойтов В.В. – д-р техн. наук, профессор

Аннотация. Описана процедура предварительной обработки изображений подписи с целью ее последующего распознавания. Она включает преобразование изображения полутоновое представление, бинаризацию, фильтрацию, поворот и масштабирование в заданный шаблон.

Ключевые слова: цифровое изображение подписи, предварительная обработка, распознавание.

Исходные цифровые изображения подписей (ИЦП) могут иметь разный цвет, размер, ориентацию в плоскости. Для сравнения нескольких ИЦП важно преобразовать их в максимально одинаковое представление. Это задача предварительной обработки изображений.

Предварительная обработка изображений является важным этапом обработки изображения [1]. От выбора методов обработки изображения зависит точность их последующего распознавания. Этапы предварительной обработки подписи напрямую зависят от качества изображения подписи.

Цель данной работы состоит в разработке универсальной процедуры предварительной обработки ИЦП для распознавания поддельных подписей, выполненных в режиме off-line.

Опишем основные этапы предварительной обработки ИЦП и кратко используемые алгоритмы:

1. Оцифровать ИЦП.
2. Преобразовать исходное ИЦП в полутоновое изображение подписи (ПИП).
3. Преобразовать ПИП в бинарное изображение подписи (БИП) методом Отсу.
4. Выполнить медианную фильтрацию БИП.
5. Выполнить поворот БИП применяя метод главных компонент к БИП.
6. Вырезать часть БИП, попавшую в описывающий прямоугольник.
7. Выполнить масштабирование БИП до заданного размера. Обычно этот этап уменьшает ИЦП.

Более детально использованные в работе алгоритмы описаны в книгах [1-2].

Для оценки работоспособности описанной процедуры предобработки подробно описаны соответствующие алгоритмы. Их программная реализация осуществлена на языке Python. На рисунке 1 представлены изображения подписи, полученные после выполнения этапов 1, 2, 4, 5, 7.



Рисунок 1 – Пример предварительной обработки ИЦП от цветного представления до бинарного фиксированного размера

Работоспособность разработанных программ проверена на нескольких доступных базах оцифрованных изображений подписей. Выполненные экспериментальные исследования с применением данных алгоритмов показали, что описанная процедура предварительной обработки подписи повышает точность ее распознавания по сравнению с распознаванием БИП, полученным после этапа 3.

Данный подход к предварительной обработке изображений подписи позволяет повысить эффективность алгоритмов анализа и распознавания подписей. Описанная процедура предварительной обработки ИЦП будет использована в системе распознавания поддельных рукописных подписей, выполненных в режиме off-line.

Список использованных источников:

1. Гонсалес, Р. Цифровая обработка изображений / Р.Гонсалес, Р. Вудс. — М.: Техносфера, 2006 — 616 с.
2. Шапиро, Л. Компьютерное зрение / Л.Шапиро, Дж.Стокман; Пер. с англ. — 3-е изд. — М.: БИНОМ. Лаборатория знаний, 2015. — 763 с.

ГРАФИЧЕСКИЕ ПРОЦЕССОРЫ В ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМАХ ОБРАБОТКИ ИНФОРМАЦИИ

Гвоздкова У.Т., студент гр.024401, Ляшенко М.С., студент гр.024401, Селезнёв И.Л.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Селезнёв И.Л. – доц., канд. тех. наук

Аннотация. В данной статье рассматривается использование графического процессора в вычислительных системах обработки информации. Анализируется увеличение скорости выполнения сегментации медицинских изображений с использованием технологии параллельного программирования.

Ключевые слова. Сегментация, графический процессор, Nvidia CUDA, OpenCL, медицинское изображение, параллелизм, GPGPU.

Графические процессоры получили широкое распространение благодаря своей высокой производительности и особенностям архитектуры, рассчитанной на массивно параллельные вычисления. Графический процессор может выполнять не только рендеринг графики, но и такие задачи, как математические вычисления и физическое моделирование.

Особый интерес представляет программная аппаратная архитектура CUDA фирмы Nvidia, позволяющая производить вычисления с использованием графических процессоров, поддерживающих технологию GPGPU (General Purpose computing on Graphics Processing Units, произвольные вычисления на видеокартах).

Высокопроизводительные возможности параллельных расчетов графических процессоров могут помочь в решении множества задач, например, получение и обработка медицинских изображений. Цифровая обработка изображений позволяет улучшить их качество.

Важной задачей при обработке и анализе изображений является сегментация – разделение изображения на области, для которых выполняется определенный критерий однородности, при этом предполагается, что области соответствуют реальным объектам, или их частям, а границы областей соответствуют границам объектов [1]. Описанные задачи также нашли свое применение и в обработке медицинских изображений (поиск и выделение структур и объектов организма). В математическом аппарате медицинских изображений выделяют три вида сегментаций. Для получения более эффективного результата используется последовательное задействование различных методов сегментации с целью надежной поддержки для диагностирования и лечения.

В связи с тем, что алгоритмы анализа медицинских изображений работают с отдельными пикселями при большом количестве операций, для их выполнения требуется высокая вычислительная мощность. Исходя из этого, появляется проблема ускорения работы алгоритмов обработки медицинских изображений, которая решается за счет использования параллельных вычислений [2].

Разрешение – основная характеристика, прямо влияющая на время обработки, поэтому необходимо применять высокопроизводительные методы для решения этой задачи. Применение параллельного программирования может быть задействовано в качестве такого метода. Для распараллеливания вычислений можно использовать как центральный процессор, так и графический, с применением его технологии GPGPU вычислений при использовании Nvidia CUDA или OpenCL [3]. В качестве метода сегментации можно использовать оператор Собеля, таким образом обозначить роль параллелизма в обработке медицинских изображений. Использование современных графических процессоров для математических расчетов является одним из наиболее перспективных способов увеличения производительности, что также обусловлено большей стоимостью аналогичных по производительности многоядерных процессоров.

Перспективность применения графических процессоров и технологий, поддерживающих параллельные вычисления, подтверждается тем, что наиболее значимыми характеристиками графических процессоров является высокая производительность и большая вычислительная мощность, что и востребовано в таких сферах, как медицинская.

Список использованных источников:

1. *Обработка медицинских изображений в графической базе данных и идентификация пациентов [Электронный ресурс]. – Режим доступа: <https://eduherald.ru/ru/article/view?id=17717>. – Дата доступа: 05.04.2022.*
2. Агибалов О. *Современные графические процессоры как средства оптимизации параллельных вычислений* / О. И. Агибалов, А.А. Золотарёв – Наука и АСУ, 2014. – с. 60-63
3. *Performance analysis of CUDA, OpenACC and OpenMP programming models on TESLA V100 GPU process* / M. Khalilov, A. Timoveev // *Journal of Physics: Conference Series*, 2021. – P. 1-2.

МАЯТНИК НЬЮТОНА С ДИНАМИЧЕСКОЙ ПОДСВЕТКОЙ

Пищ Р.С.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Стракович А.И. – магистр, ассистент кафедры ЭВМ

Показан подход к созданию маятника ньютона со светодиодными шарами. Рассмотрены различные технические особенности и варианты работы с ними.

Маятник или же колыбель Ньютона — это устройство, которое демонстрирует эффект сохранения импульса и энергии с помощью серии качающихся стальных шариков. Отклонив крайний шар и отпустив его, он, ударившись о неподвижные шары, передает энергию, которая выталкивает крайний шар в колыбели, если в колыбели взять два шара произойдет точно такая же передача энергии и после удара двух шаров энергия вытолкнет два шара с другой стороны. Обязательными условиями являются использование шаров одинакового размера. Также шары должны быть прикреплены к двум разным нитям, одинаковой длины, так чтобы в покое они касались друг друга.

Для улучшения эстетического вида маятника к нему добавляется динамическая подсветка, которая будет подсвечивать шары различным цветом в зависимости от нахождения их в состоянии движения или покоя. Для этого необходимо прикрепить к шарам светодиоды. Использование RGB светодиодов позволит настроить любой цвет. Одним из вариантов являются адресные светодиоды WS2812b или их аналоги. Светодиод можно прикрепить непосредственно на шар, но лучше всего разместить его внутри шара.

Шары являются самыми важными элементами в маятнике. Их нужно сделать все одинакового размера. Материал шаров должен быть упругим, а также рассеивать свет. Существует несколько вариантов из какого материала сделать шар.

Первый вариант — это распечатать шары на 3D-принтере. Недостатком данного подхода будет неоднородность шара, что скажется на рассеивании света, а из-за плотности пластика шар будет слишком легким. Также из-за ступенчатости печати шар будет выглядеть не эстетично.

Второй вариант заключается в использовании стеклянных шаров. Недостаток заключается в том, что стекло сложно обработать для расположения светодиода внутри и шар сложно приобрести.

Третий вариант – шар из эпоксидной смолы. Форму и смолу просто купить. Шар можно сделать в домашних условиях. В дальнейшем шар легко обработать для создания углубления под светодиод. Однако материала может сказаться на работе маятника.

Четвертый вариант — это объединение металлического каркаса и эпоксидной смолы, что позволит получить упругий удар и необходимое рассеивание света. Основная сложность заключается в процессе создания данного шара.

Для определения положения шаров используются датчики. Датчики определения препятствий для мобильных роботов являются наиболее подходящими, однако из-за размера их необходимо располагать под шарами. Также в крышке корпуса необходимо делать отверстия, чтобы в качестве препятствия был шар, а не сама крышка, что скажется на внешнем виде проекта. Небольшие по габаритам лазерные датчики приближения можно расположить позади шаров на отдельном выступе, однако в таком случае общий вид маятника будет отличаться от привычного.

Наиболее лучше подходят датчики для определения магнитного поля. Их можно расположить под крышкой корпуса, а в шары поставить магниты. В таком случае также имеется небольшой недостаток, он заключается в необходимости расположения шаров очень близко к крышке корпуса. Можно использовать герконы и датчики холла. Для каждого шара следует использовать отдельный датчик.

Основным управляющим устройством, для работы с датчиками и светодиодами, может выступать практически любой микроконтроллер с GPIO. Наиболее простым решением является использование платформы Arduino с микроконтроллером ATmega328[1]. Для настройки цвета шаров и выбора режима добавляются экран и кнопки для управления. В качестве режимов можно реализовать подсветку только крайнего шара, находящегося в движении, или режим передачи импульса, для чего необходимо отслеживать время движения крайних шаров и включать последовательно светодиоды на неподвижных шарах.

Несмотря на то, что электронная составляющая очень проста, проект является сложным для производства, так как необходимо очень точно расположить шары на одном расстоянии друг от

друга и на одной высоте. Необходимую точность можно достичь если стойки и перекладки для крепления шаров делать на станке с ЧПУ.

Список использованных источников:

1. ATmega328 Datasheet [Электронный ресурс] – Режим доступа : [http://ww1.microchip.com/downloads/en/DeviceDoc/ ATmega48A-PA-88A-PA-168A-PA-328-P-DS-DS40002061A.pdf](http://ww1.microchip.com/downloads/en/DeviceDoc/ATmega48A-PA-88A-PA-168A-PA-328-P-DS-DS40002061A.pdf) – Дата доступа : 30.03.2022

ШАХМАТНАЯ ДОСКА СО СВЕТОДИОДНОЙ ИНДИКАЦИЕЙ ПОЛЯ

Гринь Т.С.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Стракович А.И. – магистр, ассистент кафедры ЭВМ

Показан подход к созданию шахматной доски со светодиодной индикацией игрового поля. Рассмотрены различные технические проблемы и варианты работы с ними.

Шахматная доска со светодиодной индикацией поля дает различные возможности для игры в шахматы, а именно подсвечивание возможных ходов, что применимо при обучении. Второй возможностью является игра против искусственного интеллекта (ИИ). В самом простом случае ИИ подсвечивает свой ход, а человек передвигает фигуры, при этом неправильная расстановка отслеживается, предотвращая нечестную игру.

Общее строение корпуса поля заключается в использовании промежуточного каркаса между электронными компонентами клетки и ее рассеивающим свет верхом. Данный каркас предотвращает засвечивание соседних клеток. Одним из вариантов его создания является печать на 3d-принтере.

В большей части существующих шахматных досок с подсветкой для обнаружения положения фигур в них встраиваются магниты, а внутри клеток устанавливаются либо герконы, либо датчики Холла. Из-за недостаточной дальности и надежности работы герконов наиболее лучшим решением является использование датчиков Холла. Также обнаруживать положение фигур можно с помощью NFC-меток, но данный способ является избыточным и экономически нецелесообразным.

Существуют различные варианты расположения датчиков Холла. Первый вариант заключается в использовании платы в качестве боковой стенки каркаса клетки. В таком случае датчик можно расположить близко к поверхности клетки. Данный подход позволяет установить различную высоту каркаса, однако возникают ограничения на максимальный размер клетки ввиду стандартизированной длины ножек датчика холла в корпусе ТО-92.

Вторым вариантом является расположение датчика на плате снизу клеток. При таком подходе размер клетки по шире и длине может быть любым, а высота каркаса будет зависеть от длины ножек датчика. Для того чтобы не была видна тень от датчика желательно выбрать наиболее чувствительный, чтобы его можно было расположить чуть дальше от поверхности доски.

Из-за большого количества датчиков возникает проблема с подключением их к микроконтроллеру. Датчики можно подключить матричным способом или использовать расширители портов. Одним из вариантов подключения является использование ПЛИС. В самом простом варианте на ПЛИС реализуется мультиплексор и микроконтроллер постоянно опрашивает каждый датчик. В таком случае между микроконтроллером и ПЛИС необходимо только 7 проводов (6 адресных + 1 данных).

Более предпочтительным вариантом является определение смены состояния датчика и генерирование прерывания. При возникновении прерывания микроконтроллер считывает с ПЛИС адрес изменившегося состояния датчика и его новое состояние.

Для подсветки используются установленные внутри клетки светодиоды. Так же, как и датчики Холла, светодиоды можно установить на плате сбоку клетки. В таком случае подсветка будет достаточно рассеянная, чтобы не было видно тени от датчика Холла. Однако по этой же причине будут проблемы с яркостью подсветки. Для решения данной проблемы следует использовать по два светодиода на клетку и расположить их на плате внизу клетки. Следует использовать адресные светодиоды WS2812b[1] или их аналоги, что позволит подключить все светодиоды к микроконтроллеру только через один провод.

Основным управляющим устройством, для работы со светодиодами и ПЛИС, может выступать практически любой микроконтроллер с GPIO, например платформа Arduino[2] с микроконтроллером ATmega328 или микроконтроллеры серии ESP[3]: ESP32, ESP8266. Для базовой работы возможностей обоих микроконтроллеров достаточно, однако в случае использования сложных алгоритмов ИИ за счет большей производительности преимущество за ESP. В ESP также как и в ПЛИС используется напряжение 3.3V, из-за чего нет необходимости в преобразователях логического уровня.

Список использованных источников:

1. WS2812b [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://cdn-shop.adafruit.com/datasheets/WS2812B.pdf> - Дата доступа: 03.04.2022
2. Arduino [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://www.arduino.cc/> - Дата доступа: 02.04.2022
3. Espressif [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://www.espressif.com/> - Дата доступа: 03.04.2022

вычислений за малый промежуток времени. Кроме того, существуют ограничения, затрудняющие реализацию подобных комплексных алгоритмов напрямую на устройствах на платформе Android.

Таким образом, было принято решение, что наиболее оптимальным и высокопроизводительным методом реализации такого приложения является клиент-серверная архитектура, в которой мобильное приложение является клиентом, отправляющим изображения на удалённый сервер, который, в свою очередь, обрабатывает принятый кадр и возвращает результат. Далее, задачей клиента остаётся только вывести полученное от сервера изображение. При использовании такой архитектуры приложения, сложные вычисления выполняются на полноценных, более мощных, графических процессорах. В дополнение к этому, такой подход даёт возможность создания кроссплатформенного приложения без необходимости изменения серверной части, что сильно ускоряет процесс его разработки.

Данный подход также имеет и свои недостатки. Для работы приложению необходимо постоянное подключение к Интернету. Более того, скорость работы приложения напрямую зависит от скорости обмена информацией между клиентом и сервером.

В таблице 1 представлены параметры тестовых систем. В таблицах 2 и 3 представлены временные метрики для клиента и сервера, соответственно.

Таблица 1 – Параметры тестовых систем

	Смартфон	Laptop-клиент	Сервер
CPU	HiSilicon Kirin 970	Intel Core i7-8550U	4x Intel Xeon Platinum 8180M
GPU	Mali-G72 MP12	Nvidia MX250	Nvidia Tesla K80
NIC	WI-FI	Intel Dual Band Wireless-AC 8265	?

Таблица 2 – Временные метрики клиента

	Разрешение камеры	Кодирование изображения, мс			Отправка запроса на сервер (и ожидание ответа), мс			Декодирование ответа от сервера, мс			Средняя частота кадров, к/с
		Мин.	Ср.	Макс.	Мин.	Ср.	Макс.	Мин.	Ср.	Макс.	
Смартфон	1280x720	38	40	42	470	500	1500	63	75	92	1,08
	1920x1080	77	80	86	650	900	1700	75	105	124	0,7
	2816x1584	140	166	168	843	1100	2100	165	180	193	0,55
Laptop-клиент	1920x1080	13	17	27	490	550	820	13	30	49	1,15

Таблица 3 – Временные метрики сервера

Разрешение изображения	Декодирование изображения от клиента, мс	Предсказание нейросети, мс	Постобработка изображения, мс
1280x720	40	5	166
1920x1080	40	6	170
2816x1584	80	6,5	250

Исходя из результатов тестирования можно судить о том, что «бутылочным горлышком» данной реализации является код, ответственный за кодирование-декодирование изображений.

Список использованных источников:

1. Levin A. A Closed Form Solution to Natural Image Matting / A. Levin, [et al.] // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 2008. – Volume 30. – Issue 2. – P. 228-242. – DOI: 10.1109/TPAMI.2007.1177.
2. Xu N. Deep Image Matting [Electronic Resource] / arXiv. – Mode of access: <https://arxiv.org/abs/1703.03872>. – Date of access: 06.04.2022.

3. *Real-Time High-Resolution Background Matting [Electronic Resource] / University of Washington. – Mode of access: <https://grail.cs.washington.edu/projects/background-matting-v2>. – Date of access: 06.04.2022.*

4. *Chuang Y. A Bayesian Approach to Digital Matting / Y.-Y. Chuang, [et al.] // Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001. – 2001. – Kauai, HI, USA. – DOI: 10.1109/CVPR.2001.990970.*

5. *Zheng Y. Learning Based Digital Matting / Y. Zheng, C. Kambhamettu // 2009 IEEE 12th International Conference on Computer Vision. – 2009. – Kyoto, Japan. – DOI: 10.1109/ICCV.2009.5459326*

МОБИЛЬНОЕ ПРИЛОЖЕНИЕ ДЛЯ МОНИТОРИНГА СЕРДЕЧНО-СОСУДИСТОЙ СИСТЕМЫ ЧЕЛОВЕКА

Шукелович Д.В., студент гр.850502

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Луцик Ю.А. – канд. технических наук, доцент

Проект содержит в себе только программную реализацию и не затрагивает работу с аппаратным обеспечением.

Контроль за сохранностью нашего здоровья был, есть и будет одной из необходимых процедур для нормальной жизнедеятельности человека. Для осуществления такого контроля необходимо проводить ряд анализов и процедур, направленных на получение сведений о состоянии работы организма. Среди этих процедур могут быть как анализы, проводимые в лаборатории, так и обследования пациента специальным оборудованием, такие как УЗИ, МРТ и ЭКГ [1].

В данной работе затрагивается использование электрокардиограммы. Снятие электрокардиограммы является необходимой процедурой для анализа врачом работы сердечно-сосудистой системы и постановки соответствующего диагноза. Сегодня одна из основных причин проблем со здоровьем человека, связана именно с проблемами работы сердца.

Однако для получения врачебного заключения требуется обращаться в лечебное учреждение, что требует определённых затрат времени и не всегда подходит для людей, которым требуется постоянный контроль за деятельностью сердечно-сосудистой системы. Поэтому были разработаны устройства, позволяющие собирать информацию о работе сердца человека в реальном времени без необходимости обращаться к врачу.

Подобные устройства могут быть представлены в разной форме, как например стационарные кардиографы или портативные устройства [2]. Портативные устройства в свою очередь могут разрабатываться как самостоятельное, полноценное оборудование либо как вспомогательное устройство, работающие в связке со смартфоном или планшетом. Такие портативные устройства используются для непосредственного сбора и передачи информации на оборудование пользователя, на котором заранее установлено приложение для обработки информации. Приложение принимает данные работы устройства, обрабатывает их и выдаёт результат на экране смартфона, планшета.

Целью данной работы является разработка программного обеспечения для получения и передачи информации о деятельности сердечно-сосудистой системы со стороны портативного устройства и программного обеспечения обработки и отображения полученной информации на стороне мобильного устройства. В результате работы программное обеспечение должно предоставлять пользователю информацию о текущем состоянии сердечно-сосудистой системы и должно обеспечивать:

- соединение мобильного устройства пользователя и вспомогательного оборудования с помощью технологии блютуз;
- принятие данных, переданных вспомогательным устройством;
- фильтрацию принятого сигнала;
- анализ сигнала. Определение частоты работы сердца и отклонений в его работе [3];
- вывод на экран смартфона электрокардиограммы и сведений об обнаруженных в ходе работы приложения.

Результатом работы приложения должно стать отображение информации в понятном для пользователя виде о диагнозе возможных нарушений в работе его сердечно-сосудистой системы и возможность просмотра кардиограммы в удобном для восприятия виде. Полученные результаты, конечно не позволяют поставить точный диагноз, но позволяют увидеть возможные отклонения от нормы.

Приложение создаётся для смартфонов, работающих на базе операционной системы Android, и разработано на языке Java. Выбор платформы был сделан в пользу распространённости смартфонов, использующих операционную систему Android, а выбор языка был сделан благодаря простоте разработки мобильных приложений на этом языке.

Список использованных источников:

1. Электрокардиография (ЭКГ) – это метод исследования и регистрация электрической деятельности сердца. [Электронный ресурс]. – Электронные данные. – Режим доступа: <http://heart-master.com/clinic/diagnostic/ecg/> – Дата доступа: 01.02.2022.

2. Перечень портативных устройств для получения ЭКГ на дому [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://habr.com/ru/company/medgadgets/blog/390533/> – Дата доступа: 04.02.20223.
3. How to read an ECG [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://geekymedics.com/how-to-read-an-ecg> – Дата доступа: 02.03.20223.

АНАЛИЗ АЛГОРИТМОВ КЛАССИФИКАЦИИ ОБЪЕКТОВ НА ДАННЫХ ДИСТАНЦИОННОГО ЗОНДИРОВАНИЯ ЗЕМЛИ

Куприянова Д.В., аспирант

Фролов И.И., канд. техн. наук, доцент кафедры ЭВМ

Перцев Д.Ю., канд. техн. наук, доцент кафедры ЭВМ

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Фролов И.И. – канд. техн. наук, доцент

Аннотация. Обобщены сведения по способам классификации спутниковых изображений, сделан их сравнительный анализ для последующего практического применения.

Ключевые слова. Спутниковые снимки, алгоритмы классификации объектов.

Введение.

Системы дистанционного зондирования Земли (ДЗЗ) нашли широкое применение в различных областях жизнедеятельности человека:

- картография;
- метеорология;
- сельское хозяйство и другие.

Одной из проблем применения систем ДЗЗ является создание программного обеспечения, которое принимая на вход необработанные данные, автоматически выявляет кластеры объектов, затем их классифицирует и, в соответствии с решаемой задачей, «принимает» решение, что делать дальше (например, программное обеспечение фиксирует постоянное уменьшение лесного массива и, соответственно, об этом необходимо уведомить пользователя).

Алгоритмы классификации объектов на данных ДЗЗ

Несмотря на множество использующих сенсоров с различными техническими характеристиками, фиксируемой спектральной областью, подходы к классификации объектов в большинстве случаев сводится к применению одного из следующих методов [1] (рисунок 1):

- методы на основе сопоставления с шаблоном (англ., template matching-based methods) [2-9];
- методы, основанные на знаниях (англ., knowledge-based methods) [10-12];
- анализ изображений на основе объектов (англ., OBIA-based methods) [13-15];
- методы на основе машинного обучения (англ., machine learning-based methods), в т.ч. методы на основе глубоких нейронных сетей (англ., deep-learning neural network methods) [16-23].

Методы на основе сопоставления с шаблоном являются одним из простейших и старых методов, включающих два шага:

- создание шаблона для каждого объекта, который необходимо распознать;
- оценка схожести шаблона с изображением с учетом всех возможных аффинных преобразований. Чаще всего в качестве оценки схожести упоминаются следующие метрики: сумма абсолютных разностей (SAD), сумма квадратов разностей (SSD), нормализованная взаимная корреляция (NCC) и евклидово расстояние (ED).

Методы, основанные на знаниях. Основной сложностью является формирование знаний и правил об объекте, в соответствии с которыми в дальнейшем будет определяться, о каком объекте идет речь. В целом, анализ статей показывает 2 основных подхода к формированию знаний:

- знания о геометрической форме и радиометрических свойствах;
- знания о контексте (например, отбрасывание тени от высоких объектов).

Анализ изображений на основе объектов – один из подходов, разработанный для преодоления ограничений подходов, основанных на анализе классических изображений, и применении знаний о спектральной, текстурной и контекстной информации (объектах).

Развитие *методов машинного обучения* и, особенно, развитие классификаторов и способов представления признаков привело к существенному прогрессу в области детекции объектов на спутниковых снимках.

Методы на основе глубоких нейронных сетей нашли широкое применение при обработке естественных фотографий и видеопотока и являются одними из наиболее перспективных. Однако

существенной проблемой в данной области является найти качественно подготовленный достаточно большой тестовый набор для обучения глубокой нейронной сети.

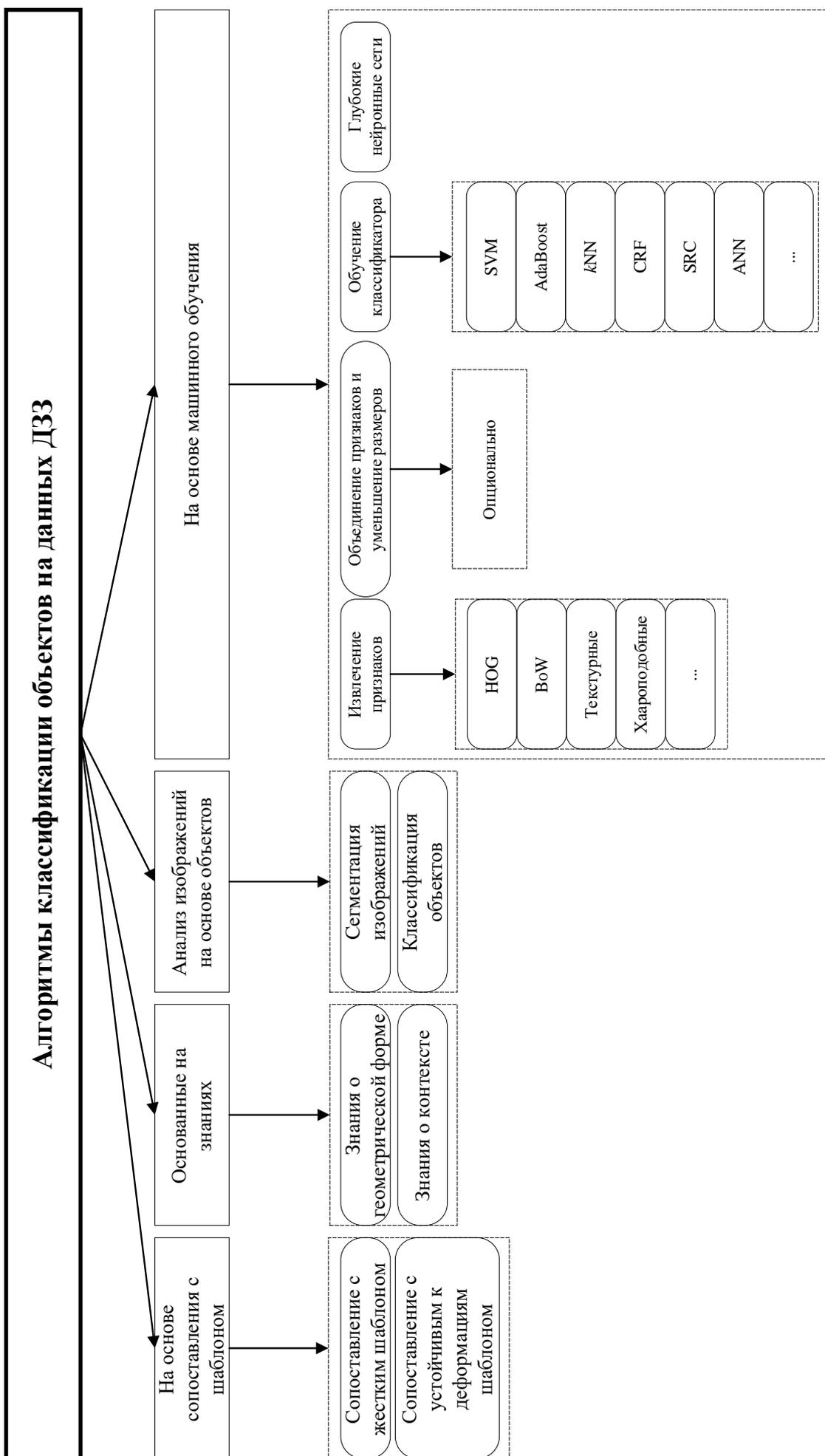


Рисунок 1 – Алгоритмы классификации объектов на данных ДЗЗ

Заключение

Рассмотрены и классифицированы разные подходы к определению и классификации объектов на спутниковых снимках Земной поверхности, выявлены их достоинства и недостатки. На основе полученных сведений можно сделать вывод, что подходы с применением алгоритмов машинного обучения, а также глубокие нейронные сети несмотря на некоторые недостатки по состоянию на 2021 год являются наиболее перспективными для последующего применения на практике.

Список использованных источников:

1. Cheng, G. A survey on object detection in optical remote sensing images / G. Cheng, Junwei Han // *ISPRS Journal of Photogrammetry and Remote Sensing*. – 2016. – Volume 117. – P.11-28.
2. Stankov, K. Detection of Buildings in Multispectral Very High Spatial Resolution Images Using the Percentage Occupancy Hit-or-Miss Transform / K. Stankov, Dong-Chen He // *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*. – 2014. – Volume 7. – Issue 10. – P. 4069 - 4080. DOI: 10.1109/JSTARS.2014.2308301.
3. Weber, J. Spatial and spectral morphological template matching / J. Weber, S. Lefèvreb // *Image and Vision Computing*. – 2012. – Volume 30. – Issue 12. – P. 934-945.
4. Chaudhuri, D. Semi-Automated Road Detection From High Resolution Satellite Images by Directional Morphological Enhancement and Segmentation Techniques / Chaudhuri D., Kushwaha N., Samal A. // *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*. – 2012. – Volume 5. – Issue 5. – P. 1538-1544.
5. Lin, Y. Rotation-invariant object detection in remote sensing images based on radial-gradient angle / Lin, Y., He, H., Yin, Z., Chen, F. // *IEEE Geoscience and Remote Sensing Letters*. – 2015. – Volume 12. – Issue 4. – P. 746-750.
6. Liu, G. Interactive geospatial object extraction in high resolution remote sensing images using shape-based global minimization active contour model / Liu, G., Sun, X., Fu, K., Wang, H. // *Pattern Recognition Letters*. – 2013. – Volume 34. – Issue 10. – P.1186-1195.
7. Tao, C. Airport detection from large IKONOS images using clustered SIFT keypoints and region information / Tao, C., Tan, Y., Cai, H., Tian, J. // *IEEE Geoscience and Remote Sensing Letters*. – 2011. – Volume 8. – Issue 1. – P.128-132.
8. Leninisha, S. Water flow based geometric active deformable model for road network / Leninisha, S., Vani, K. // *ISPRS Journal of Photogrammetry and Remote Sensing*. – 2015. – Volume 102. – P.140-147.
9. Weidner, U. Towards automatic building extraction from high-resolution digital elevation models / Weidner, U., Förstner, W. // *ISPRS Journal of Photogrammetry and Remote Sensing*. – 1995. – Volume 50. – Issue 4. – P.38-49.
10. Ok, A.O. Automated detection of arbitrarily shaped buildings in complex environments from monocular VHR optical satellite imagery / Ok, A.O., Senaras, C., Yuksel, B. // *IEEE Transactions on Geoscience and Remote Sensing*. – 2013. – Volume 51. – Issue 3. – P.1701-1717.
11. Akçay, H.G. Building detection using directional spatial constraints / Akçay, H.G., Aksoy, S. // *IEEE International Geoscience and Remote Sensing Symposium*. – 2010. – P.1932-1935.
12. Contreras, D. Monitoring recovery after earthquakes through the integration of remote sensing, GIS, and ground observations: the case of L'Aquila (Italy) / Contreras, D., Blaschke, T., Tiede, D., Jilge, M. // *Cartography and Geographic Information Science*. – 2016. – Volume 43. – Issue 2. – P.115-133.
13. Hussain, M. Change detection from remotely sensed images: From pixel-based to object-based approaches / Hussain, M., Chen, D., Cheng, A., Wei, H., Stanley, D. // *ISPRS Journal of Photogrammetry and Remote Sensing*. – 2013. – Volume 80. – P.91-106.
14. Li, X. Identification of forested landslides using LiDAR data, object-based image analysis, and machine learning algorithms / Li, X., Cheng, X., Chen, W., Chen, G., Liu, S. // *Remote Sensing*. – Volume 7. – Issue 8. – P.9705-9726.
15. Liu, L. Airplane detection based on rotation invariant and sparse coding in remote sensing images / Liu, L., Shi, Z. // *Optik*. – 2014. – Volume 125. – Issue 18. – P.5327-5333.
16. Li, E. Robust Rooftop Extraction From Visible Band Images Using Higher Order CRF / Li, E., Femiani, J., Xu, S., Zhang, X., Wonka, P. // *IEEE Transactions on Geoscience and Remote Sensing*. – 2015. – Volume 53, Issue 8. – P.4483-4495.
17. Yao, X. A coarse-to-fine model for airport detection from remote sensing images using target-oriented visual saliency and CRF / Yao, X., Han, J., Guo, L., Bu, S., Liu, Z. // *Neurocomputing*. – 2015. – Volume 164. – P.162-172.
18. Zhang, W. A generic discriminative part-based model for geospatial object detection in optical remote sensing images / Zhang, W., Sun, X., Wang, H., Fu, K. // *ISPRS Journal of Photogrammetry and Remote Sensing*. – 2015. – Volume 99. – P.30-44.
19. Cheng, G. Learning Rotation-Invariant and Fisher Discriminative Convolutional Neural Networks for Object Detection / Cheng, G., Han, J., Zhou, P., Xu, D. // *IEEE Transactions on Image Processing*. – 2019. – Volume 28. – Issue 1. – P.265-278.
20. Li, K. Rotation-Insensitive and Context-Augmented Object Detection in Remote Sensing Images / Li, K., Cheng, G., Bu, S., You, X. // *IEEE Transactions on Geoscience and Remote Sensing*. – 2018. – Volume 56. – Issue 4. – P.2337-2348.
21. Long, Y. Accurate Object Localization in Remote Sensing Images Based on Convolutional Neural Networks / Long, Y., Gong, Y., Xiao, Z., Liu, Q. // *IEEE Transactions on Geoscience and Remote Sensing*. – 2017. – Volume 55. – Issue 5. – P.2486-2498.
22. Tang, T. Vehicle Detection in Aerial Images Based on Region Convolutional Neural Networks and Hard Negative Example Mining / Tang, T., Zhou, S., Deng, Z., Zou, H., Lei, L. // *Sensors*. – Volume 17. – Issue 2. – P.336.
23. Li, K. Object detection in optical remote sensing images: A survey and a new benchmark / K.Li, [et. al.] // *ISPRS Journal of Photogrammetry and Remote Sensing*. – 2020. – Volume 159. – P.296-307.

UDC 004.932.2, 004.93'14

ANALYSIS OF OBJECT CLASSIFICATION ALGORITHMS OF REMOTE SENSING DATA

Kupryianava D.¹, Frolov I.¹, Pertsau D.¹

Belarusian State University of Informatics and Radioelectronics¹, Minsk, Republic of Belarus

Frolov I. – PhD in Technical Sciences

Annotation. The information of satellite images classification methods is generalized; their comparative analysis is made for subsequent practical application.

Keywords. Satellite images, Object classification algorithms.

СИСТЕМА РАСПОЗНАВАНИЯ ОБЪЕКТОВ НА СНИМКАХ ЗЕМНОЙ ПОВЕРХНОСТИ НА ОСНОВЕ СВЕРТОЧНОЙ НЕЙРОННОЙ СЕТИ U-NET

Куприянова Д.В.

Некревич Ю.И.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Никульшин Б.В. – канд. техн. наук, доцент

Представлены результаты эксперимента с применимостью сверточной нейронной сети U-Net для сегментации множества объектов на снимках земной поверхности.

Развитие авиакосмической промышленности привело не просто к активному освоению космического пространства, но и появлению множества потоков информации, получаемых от спутников, находящихся на орбите Земли, и, как результат, постоянно возрастающим требованиям к качеству и скорости их автоматизированной обработки. Получаемые снимки имеют большую ценность во многих отраслях: сельское хозяйство, метеорология, георазведка и т.д. Однако существенной проблемой является качественный анализ снимков земной поверхности, поскольку они содержат большой объем информации при относительно невысокой детализации самих объектов.

Целью работы является разработка системы сегментации и классификации объектов на аэрофотоснимках различного пространственного разрешения. В качестве поддерживаемых классов объектов определены строения, поле, вода, дорога, зеленые насаждения (например, лес).

Схематично разработанная система представлена на рисунке 1.



Рисунок 2 – Структурная схема предлагаемой системы

Блок входных данных загружает изображение в одном из поддерживаемых форматов (tiff, jpg, ppm), после чего изображение подвергается предварительной обработке, включающей конвертацию в изображение в градациях серого и масштабирование к заданному размеру.

Затем полученное изображение передается на блок сегментации. В данном случае используется сверточная нейронная сеть U-Net, созданная в 2015 году для сегментации биомедицинских изображений в отделе Computer Science Фрайбургского университета [1]. Архитектура сети представляет собой полносвязную сверточную сеть, модифицированную так, чтобы она могла работать с меньшим количеством обучающих образов при увеличении точности сегментации.

Архитектура сети показана на рисунке 2, состоит из двух частей (сужающейся и расширяющейся) и соответствует схеме кодер-декодер:

- кодировщик уменьшает пространственное измерение с помощью объединения слоев;
- декодер восстанавливает детали объекта и пространственное измерение.

Также существуют быстрые соединения от кодера к декодеру, чтобы «помочь» декодеру лучше восстановить детали объекта.

Блок постобработки в перспективе будет выполнять анализ полученных результатов, выявление явно ошибочных участков (например, разрывы в близко расположенных отрезках дорожного полотна).

Блок визуализации сохраняет полученный результат и выводит его на экран.

Тестовый набор включает 25 изображений (пример оригинального снимка земной поверхности показан на рисунке 3.а, результат распознавания – на рисунке 3.б), заранее размеченных вручную. Обучение для поддерживаемых классов выполнялось отдельно. Для оценки качества распознавания всех поддерживаемых классов объектов использовался коэффициент Жаккара:

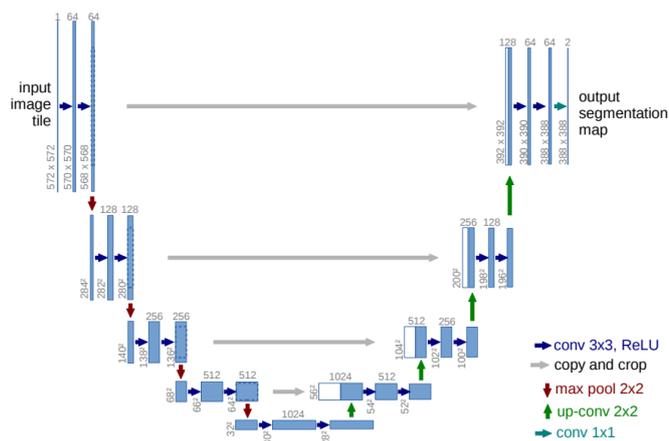


Рисунок 2 – Базовая архитектура U-Net

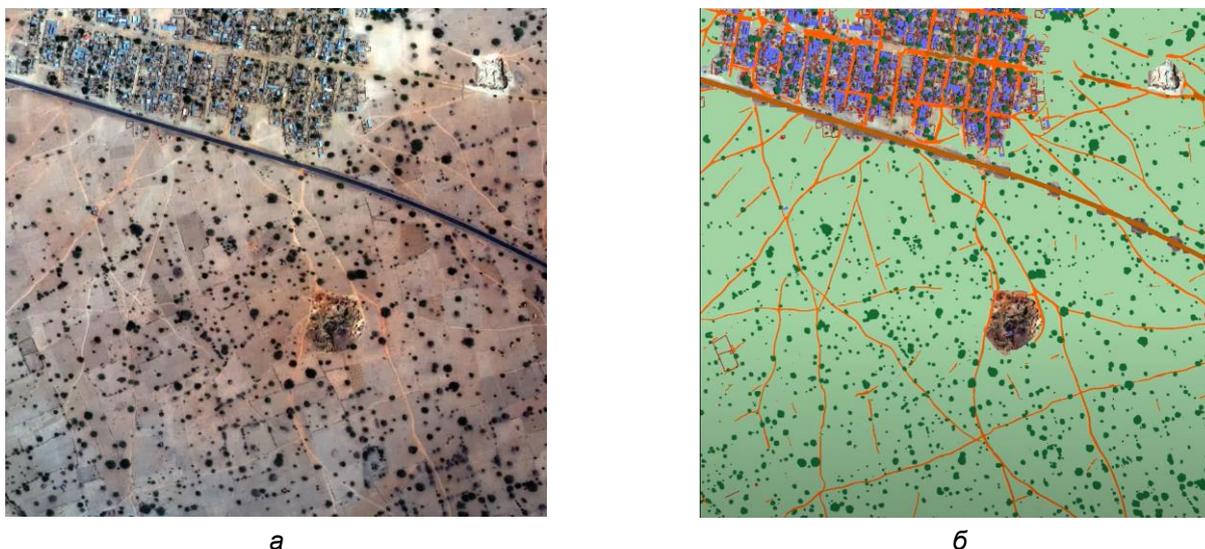


Рисунок 3 – Пример фотографии земной поверхности: а – оригинальный снимок, б – после сегментации

$$J = \frac{1}{n} \sum_{i=1}^n \frac{A_i \cap B_i}{A_i \cup B_i}, \quad (1)$$

где n – количество классов, которые поддерживаются разработанным алгоритмом;

A_i и B_i – координаты областей исходной и предсказанной масок соответственно.

Тестирование проводилось с применением центрального процессора Intel Core i7-9700 и видеокарты NVIDIA RTX 2060 RTX с 8 Гб видеопамати.

Представлен алгоритм сегментации объектов на снимках земной поверхности с помощью 5 заранее построенных моделей на основе сверточной нейронной сети U-Net (соответственно для строений, полей, воды, дороги, зеленых насаждений).

Полученный усредненный коэффициент Жаккара для всех поддерживаемых классов объектов составляет 0,64, что является довольно низким коэффициентом, что связано с низким качеством обучающей выборки и высокими требованиями к необходимому аппаратному обеспечению, проявляющимися в длительном процессе обучения.

В дальнейшем планируется усовершенствовать тестовый набор данных для обучения, исследовать алгоритмы постобработки для выявления явных ошибок (например, разрывы в близко расположенных отрезках дорожного полотна), а также исследовать альтернативные архитектуры нейронных сетей.

Список использованных источников:

1. U-Net: Convolutional Networks for Biomedical Image Segmentation [Electronic Resource] / ArXiv. – Mode of access: <https://arxiv.org/abs/1505.04597>. – Date of access: 06.04.2022.

ЗАМЕНА ФОНА НА ИЗОБРАЖЕНИИ ВЫСОКОГО РАЗРЕШЕНИЯ В РЕАЛЬНОМ ВРЕМЕНИ

Исмаил А-К.Х.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Перцев Д.Ю. – канд. техн. наук, доцент

Представлен алгоритм матирования изображения высокого разрешения на основе рекуррентной нейронной сети с использованием алгоритмов кодирования-декодирования изображения. Также представлено приложение на платформе Android, реализующее разработанный алгоритм. Приведены результаты анализа производительности.

Матирование изображения I является задачей из области компьютерного зрения и основано на разделении заднего плана B (от англ. background) и переднего плана F (от англ. foreground):

$$I = \alpha F + (1 - \alpha)B, \quad (1)$$

где F – изображение переднего плана;

B – изображение заднего плана;

α – карта прозрачности переднего плана, элементы в которой принимают значение в диапазоне $[0, 1]$ (значение 0 – пиксель с координатой (x, y) принадлежит заднему плану, значение 1 – переднему плану).

Алгоритм, представленный в данной работе, имеет существенное преимущество перед классическими алгоритмами матирования [1,2]: в то время, как более старые алгоритмы требуют наличия заранее подготовленного фона [3], тримапа [4] или набросков [5], данный алгоритм лишён этого недостатка. Это достигается за счёт рекуррентной архитектуры нейронной сети, и позволяет использовать его в более широком классе задач, так как отсутствует необходимость прямого пользовательского вмешательства. Таким образом, данный алгоритм наилучшим образом подходит для использования в мобильных приложениях.

На рисунке 1 представлена архитектура нейронной сети, осуществляющая матирование. Она состоит из кодера, рекуррентного декодера, выходного блока, а также блока повышения разрешения. Перед попаданием в первый блок, изображение высокого разрешения уменьшается. Данный шаг позволяет ускорить обработку кадра, при этом из-за применения фильтра повышения разрешения на выходе сети, существенной потери качества на выходе не происходит. Блок кодирования извлекает полезные признаки на изображении, уменьшает полученный кадр в 2 раза и передаёт на следующий кодер. Данный этап повторяется 4 раза. Таким образом, кодеры извлекают признаки в масштабе 12, 14, 18 и 116. Затем рекуррентный декодер также поэтапно обновляет состояния сети и декодирует признаки, понижая количество каналов исходного изображения до одного – альфа. Выходной блок выполняет финальное извлечение признаков и генерирует коэффициенты альфа, а также формирует передний план, который затем, вместе с исходным изображением в высоком разрешении, передаются фильтру для повышения разрешения результирующих изображений.

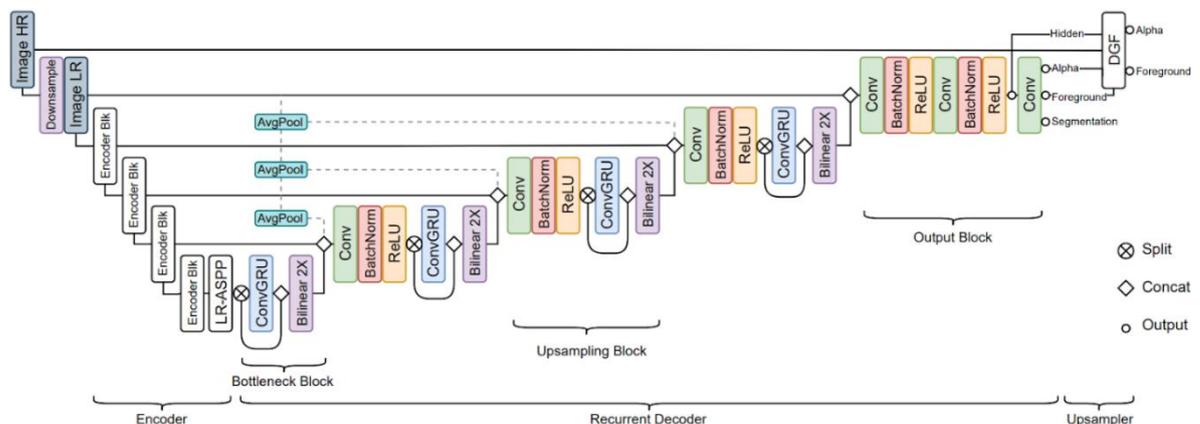


Рисунок 1 – Архитектура нейронной сети

Одной из ключевых проблем представленной реализации алгоритма является то, что на данный момент не существует мобильного устройства способного выполнить такой объём вычислений за малый промежуток времени. Кроме того, существуют ограничения, затрудняющие реализацию подобных комплексных алгоритмов напрямую на устройствах на платформе Android.

Таким образом, было принято решение, что наиболее оптимальным и высокопроизводительным методом реализации такого приложения является клиент-серверная архитектура, в которой мобильное приложение является клиентом, отправляющим изображения на удалённый сервер, который, в свою очередь, обрабатывает принятый кадр и возвращает результат. Далее, задачей клиента остаётся только вывести полученное от сервера изображение. При использовании такой архитектуры приложения, сложные вычисления выполняются на полноценных, более мощных, графических процессорах. В дополнение к этому, такой подход даёт возможность создания кроссплатформенного приложения без необходимости изменения серверной части, что сильно ускоряет процесс его разработки.

Данный подход также имеет и свои недостатки. Для работы приложению необходимо постоянное подключение к Интернету. Более того, скорость работы приложения напрямую зависит от скорости обмена информацией между клиентом и сервером.

В таблице 1 представлены параметры тестовых систем. В таблицах 2 и 3 представлены временные метрики для клиента и сервера, соответственно.

Таблица 1 – Параметры тестовых систем

	Смартфон	Laptop-клиент	Сервер
CPU	HiSilicon Kirin 970	Intel Core i7-8550U	4x Intel Xeon Platinum 8180M
GPU	Mali-G72 MP12	Nvidia MX250	Nvidia Tesla K80
NIC	WI-FI	Intel Dual Band Wireless-AC 8265	?

Таблица 2 – Временные метрики клиента

	Разрешение камеры	Кодирование изображения, мс			Отправка запроса на сервер (и ожидание ответа), мс			Декодирование ответа от сервера, мс			Средняя частота кадров, к/с
		Мин.	Ср.	Макс.	Мин.	Ср.	Макс.	Мин.	Ср.	Макс.	
Смартфон	1280x720	38	40	42	470	500	1500	63	75	92	1,08
	1920x1080	77	80	86	650	900	1700	75	105	124	0,7
	2816x1584	140	166	168	843	1100	2100	165	180	193	0,55
Laptop-клиент	1920x1080	13	17	27	490	550	820	13	30	49	1,15

Таблица 3 – Временные метрики сервера

Разрешение изображения	Декодирование изображения от клиента, мс	Предсказание нейросети, мс	Постобработка изображения, мс
1280x720	40	5	166
1920x1080	40	6	170
2816x1584	80	6,5	250

Исходя из результатов тестирования можно судить о том, что «бутылочным горлышком» данной реализации является код, ответственный за кодирование-декодирование изображений.

Список использованных источников:

6. Levin A. A Closed Form Solution to Natural Image Matting / A. Levin, [et al.] // *IEEE Transactions on Pattern Analysis and Machine Intelligence*. – 2008. – Volume 30. – Issue 2. – P. 228-242. – DOI: 10.1109/TPAMI.2007.1177.
7. Xu N. Deep Image Matting [Electronic Resource] / arXiv. – Mode of access: <https://arxiv.org/abs/1703.03872>. – Date of access: 06.04.2022.
8. Real-Time High-Resolution Background Matting [Electronic Resource] / University of Washington. – Mode of access: <https://grail.cs.washington.edu/projects/background-matting-v2>. – Date of access: 06.04.2022.

9. Chuang Y. *A Bayesian Approach to Digital Matting* / Y.-Y. Chuang, [et al.] // *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001.* – 2001. – Kauai, HI, USA. – DOI: 10.1109/CVPR.2001.990970.

10. Zheng Y. *Learning Based Digital Matting* / Y. Zheng, C. Kambhampettu // *2009 IEEE 12th International Conference on Computer Vision.* – 2009. – Kyoto, Japan. – DOI: 10.1109/ICCV.2009.5459326

ПРИМЕНЕНИЕ МИКРОКОНТРОЛЛЕРОВ В СИСТЕМАХ УПРАВЛЕНИЯ И МОНИТОРИНГА

Полудневич С.С., студент гр.924404

Селезнёв И.Л., канд. тех. наук, доцент

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Селезнёв И.Л. – канд. тех. наук, доцент

Аннотация. В данной статье рассматриваются понятия систем управления и мониторинга, дается описание встраиваемой системы как основной их составляющей. Затрагиваются основные области применения микроконтроллеров в таких системах, их спецификации и основные характеристики, приводится обзор современного рынка микроконтроллеров и тенденции развития.

Ключевые слова. Система, контроллер, управление, мониторинг, микроконтроллер, встраиваемая система, периферия, архитектура, устройство, интерфейс, система на кристалле, память, сенсор, процессор, интеграция.

Системы управления представляют собой специализированные решения для контроля функционирования ИТ-инфраструктуры. Такие системы зачастую применяются в совокупности с системами мониторинга для сбора данных и контроля параметров системы управления. Для решения различных задач в данных системах чаще всего применяются встраиваемые решения, позволяющие автоматизировать технологические процессы.

Встраиваемые системы можно определить как специализированные (заказные) микропроцессорные системы, непосредственно взаимодействующие с объектом контроля управления и, возможно, объединенные с ним конструктивно. Это системы от контроллера офисного дверного замка до инфокоммуникационной «начинки» заводов-автоматов, интеллектуальных энергетических систем, зданий, транспортных систем, умных городов, домов.

Основными особенностями встраиваемой вычислительной системы являются:

- работа в реальном масштабе времени (в большинстве случаев);
- наличие сенсоров и исполнительного устройства управления;
- автономность работы (отсутствие оператора, ограничения электропитания);
- высокие требования по надежности и безопасности функционирования;
- критические применения, связанные со здоровьем и жизнью человека.

Современные встраиваемые системы находят применение практически во всех сферах деятельности человека – от контроля за домашними электроприборами до применения в ядерных реакторах и космических аппаратах. Системы управления и мониторинга различных объектов, перечисленные в таблице 1, не могли бы реализовываться без встраиваемых систем [1].

Таблица 1 – Примеры применения встраиваемых систем

Авиационные и военные системы	Автопилоты самолетов, авионика и навигационные системы, системы автоматической посадки, системы наведения, управление двигателем.
Потребительская электроника	Телевизоры, духовки, посудомоечные машины, плееры DVD, стереосистемы, системы безопасности, управление поливом газонов, термостаты, фотокамеры, часы, автоответчики, декодеры кабельного телевидения.
Промышленное оборудование	Управление лифтами, системы наблюдения, роботы, станки с ЧПУ, программируемые логические контроллеры, промышленные системы автоматизации и управления.
Роботизированные системы	Промышленные роботы, автономные транспортные средства, космические исследовательские роботы (роботы-марсоходы, зонды, космические станции).

Встраиваемые системы создаются на базе микропроцессоров или микроконтроллеров. В любом случае в основе продукта лежит интегральная схема, которая обычно предназначена для выполнения вычислительных операций в реальном времени. Работа в реальном времени является крайне важной составляющей в работе встраиваемой системы, поскольку зачастую к таким системам применяются жесткие критерии по времени отклика управляющего исполнительного устройства на приход информации с внешнего датчика. На рисунке 1 показано взаимодействие внутри встраиваемой системы в реальном масштабе времени. Время получения сигнала с сенсора – t_1 , t_2 – время выдачи управляющего воздействия на исполнительное

устройство, t_3 – крайний срок выдачи управляющего воздействия. Если по какой-либо причине выдача управляющий сигнал будет выработан после t_3 , то он будет бесполезен или, возможно, вреден. Например, система управления стеклоподъемником в автомобиле – если встраиваемая система игнорирует сигнал датчика положения стекла, либо стекло, либо подающий стекло механизм могут быть испорчены.

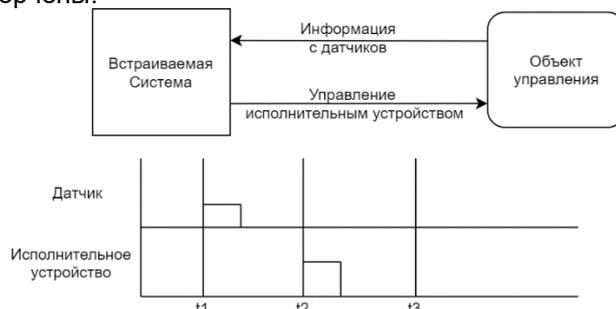


Рисунок 1 – Работа встраиваемой системы в реальном времени

Вычислительные системы могут быть представлены тремя группами элементов: обработчики, устройства памяти, интерфейсы. С группой обработчиков связаны термины «процессор» и «микроконтроллер». Микроконтроллеры встраиваемых систем проектируются как автономные системы, содержащие в себе множество специализированных частей. Главным отличием микропроцессоров от микроконтроллеров является специализация микроконтроллеров, в то время как, процессоры можно применять для любого рода задач, при подключении соответствующих устройств. Для описания микроконтроллеров часто используется термин «система на кристалле» (System on Crystal, SoC), поскольку в современных системах управления и мониторинга существует тенденция к интеграции периферии встраиваемой системы и микроконтроллера в единое целое. Системы на кристалле обычно состоят из трех основных цифровых системных блоков: процессор, память и логические компоненты. Процессорное ядро реализует поток управления, когда каждой управляющей программой однозначно устанавливаются последовательности выполнения операций обработки данных, что позволяет задавать один из возможных алгоритмов работы всей интегральной схемы. Память используется по ее прямому назначению — хранение кода программы процессорного ядра и данных. Наконец, логические компоненты используются для реализации специализированных аппаратных устройств обработки данных, состав и назначение которых определяются конечным приложением. Типовая схема системы на кристалле обычно включает в себя так же и набор периферийных устройств и контроллеров: контроллер прерываний, таймеры, контроллер сети, контроллер последовательного канала, контроллер памяти, контроллер ПДП (прямого доступа к памяти). Интерфейсы используются для обмена данными между системой на кристалле, внешними устройствами и сенсорами, а также для программирования контроллера с компьютера. На рисунке 2 представлен типичный вариант схемы SoC.

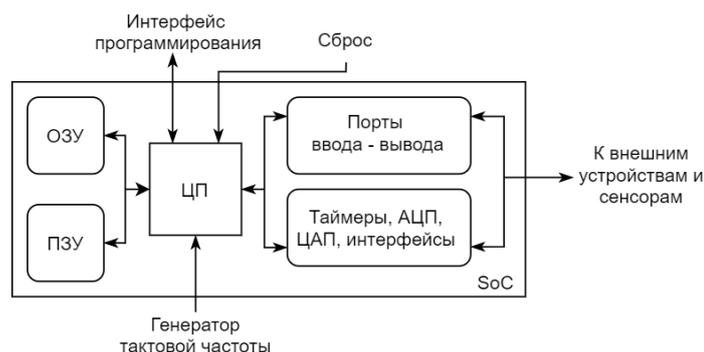


Рисунок 2 – Структура системы на кристалле

Системы на кристалле должны обладать определенным набором характеристик, специфичных для сферы применения встраиваемой системы. Такие спецификации могут, например, включать в себя разрядность процессорного ядра, возможность работы в режиме пониженного энергопотребления, требования к архитектуре микроконтроллера, поддержку определенных интерфейсов, наличие таймеров, АЦП и остальной периферии. В таблице 2 приведены часто используемые распространенные архитектуры микроконтроллеров.

Таблица 2 – Архитектуры микроконтроллеров

Архитектура	Разработчик	Производитель	Основные серии
-------------	-------------	---------------	----------------

AVR	Atmel	Atmel	tinyAVR, megaAVR, XMEGA
Motorola HC11	Motorola	NXP semiconductors	6800, 68000, 88000
ARMv7, ARMv8-A	ARM	STMicroelectronics, NXP	STM32F103xx
PIC	Microchip	Microchip Technologies	PIC16, PIC18, PIC32

Объем выпуска микроконтроллеров с однократно программируемой памятью программ постоянно сокращается из-за быстрого развития и удешевления Flash-технологий. Особенности архитектуры AVR также является развитая система прерываний, программируемые биты защиты, процессорное 32-битное ядро с упрощенным набором команд (Reduced Instruction Set Computer, RISC). Такие серии контроллеров, как mega328, mega32u4, tiny85 являются наиболее популярными на сегодняшний день, и находят применение во многих промышленных областях за счет своей дешевизны изготовления, простоты и эффективности [2]. Микроконтроллеры архитектуры ARMv7 и ARMv8 на сегодняшний день продолжают тенденцию развития 32-битных систем с ядрами Cortex-M [3]. Их особенностями является наличие большого числа портов ввода-вывода, что позволяет применять их в системах управления лифтами, умными домами, мониторинга состояния зданий, охранно-пожарных систем и иных мест, требующих обработки данных с большого количества сенсоров.

Немаловажно и наличие специфической периферии в микроконтроллерах для разных сфер применения систем мониторинга и управления. Например, без такой периферии, как таймеры, АЦП, ЦАП, не обходится практически ни одна встраиваемая система. Поддержка интерфейсов коммуникации, как параллельных, так и последовательных, является неотъемлемой частью систем мониторинга, в которых важно поддерживать обмен данными между контроллерами. В таблице 3 приведены наиболее распространенные современные контроллеры и особенности их периферии.

Таблица 3 – Специфическая периферия контроллеров

Серии контроллеров	Особенности периферии
dsPIC	10-битный 4-канальный ШИМ модуль, 12-бит АЦП, 11 16-битных таймеров
STM32F101	51 порт ввода-вывода, поддержка I2C, CAN, SPI, USART интерфейсов, RTC, NVIC [3]
Mega	8-битный процессор, RC генератор, продвинутый режим энергосбережения.

В микроконтроллерах серии dsPIC важной частью периферии является интегрированный в систему на кристалле ШИМ – контроллер с поддержкой 4 каналов и 10-битной разрядности. Наличие такой периферии позволяет использовать этот микроконтроллер в системах управления двигателями, асинхронными и синхронными моторами и другой электромеханикой, где требуется контроль вращения с помощью широтно-импульсной модуляции. Основу периферии STM32 составляет от 32 до 64 портов ввода-вывода и наличие привязанных к ним интерфейсов обмена данными I2C, CAN, USART [3]. В отличие от STM и PIC контроллеров, например, контроллер ATmega328 реализует продвинутое 8-битную RISC архитектуру, поддержку выбора частоты работы ядра, 6 режимов пониженного энергосбережения для сохранения регистров в условиях чрезвычайно низким энергообеспечением.

Перспективными направлениями развития являются снижение энергосбережения у 8-битных контроллеров AVR для расширения областей их применения в системах управления с повышенной автономностью. Также, микроконтроллеры ARM с разрядностью 32 бита имеют тенденцию к наращиванию такой периферии, как например АЦП, ЦАПы, последовательные интерфейсы и контроллеры векторных прерываний, из-за растущих потребностей в управлении и мониторинге масштабных промышленных предприятий и систем управления лифтами и умными домами [3].

Список использованных источников:

1. Баррет, Ф. Встраиваемые системы. Проектирование приложений на микроконтроллерах семейства 68HC12/HCS12 с применением языка С / С. Ф. Баррет, Д. Дж. Пак. : ДМК, Москва, 2007. – с. 22-26
2. Белов А. Микроконтроллеры AVR. От азов программирования до создания практических устройств / А. В. Белов – Наука и техника, 2020. – с. 72-75
3. Developing a New Generation of Smart Automation with ARM Cortex-M3 [Электронный ресурс]. – Режим доступа: <https://www.radiolocman.com/review/article.html?di=134036>. – Дата доступа: 20.03.2021.

UDC 004.31-022.53:681.5

APPLICATION OF MICROCONTROLLERS IN CONTROL AND MONITORING SYSTEMS

Poludnevich S.S., Seleznev I.L.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Seleznev I.L. – PhD in Technical Sciences

Annotation. This article discusses the concepts of control and monitoring systems, provides a description of the embedded system as their main component. The main areas of application of microcontrollers in such systems, their specifications and main characteristics are touched upon, an overview of the modern microcontroller market and development trends are given.

Keywords. System, controller, control, monitoring, microcontroller, embedded system, peripherals, architecture, device, interface, system on crystal, memory, sensor, processor, integration.

**СЕКЦИЯ «МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ
ТЕХНИЧЕСКИХ СИСТЕМ И ИНФОРМАЦИОННЫЕ
ТЕХНОЛОГИИ»**

УДК 629.11.073+(016.3)

СОЗДАНИЕ ПРИВОДОВ КОЛЁСНО-ШАГАЮЩИХ ДВИЖИТЕЛЕЙ МОБИЛЬНЫХ МАШИН С АДАПТИВНЫМ УПРАВЛЕНИЕМ

Алехнович Н. Г.

*Белорусский национальный технический университет,
г. Минск, Республика Беларусь*

*Комяк И. М. – старший преподаватель,
кафедра «Машиноведения и деталей машин»*

Аннотация. Построение компьютерной модели колесно-шагающего движителя с адаптивным управлением для получения кинематических характеристик движения машины в различных условиях эксплуатации

Ключевые слова: движитель, кинематика колёсно-шагающего механизма, адаптивное управление

Необходимость создания движителя тягово-транспортных средств для разнообразных условий передвижения с неизменным сохранением высокой эффективности и надёжности его использования во всём диапазоне этих условий привело к созданию движителя, воплощающего в себе положительные качества колеса и шагания. Созданный в БНТУ колёсно-шагающий движитель использует при движении четыре опорных башмака цилиндрической формы, которые крепятся на концах двух взаимно перпендикулярных штанг. Перенос опорных башмаков осуществляется с помощью сложно-вращательного движения штанг, складывающегося из плоскопараллельного движения каждой штанги на двух кривошипах относительно ступицы и вращение вместе со ступицей. В работе движителя органически сочетается принцип шагания (поочередный перенос башмаков с отрывом их от опорной поверхности) с принципом качения (прокат на каждом башмаке в процессе его контакта с опорой) именно поэтому, а также в следствие вращательного характера движений переноса башмаков многократно повышена скорость движения (до 30 км/ч), а также снижены виброактивность и инерционные нагрузки движителя. Кроме того, благодаря вращательному движению привода, предложенный движитель может быть установлен взамен обычных колёс на серийно выпускаемых машинах без существенных их переделок. Демонтируются только колёсные редукторы, функции которых выполняют сами механизмы шагающих колёс.

Указанное выше сложное движение штанг осуществлено с помощью размещённого в ступице движителя планетарного зубчатого механизма.

Для данной схемы колёсно-шагающего движителя была произведена оптимизация геометрических параметров звеньев с целью сведения к минимуму вертикальных колебаний оси колеса при шагании, в результате чего амплитуда этих колебаний не превышает 0,8% от расстояния между осью и опорой поверхностью грунта. Были также разработаны устройства, устраняющие колебания линейной скорости оси движителя в процессе шага.

Для выбора оптимальных решений при проектировании кинематики колёсно-шагающего механизма была разработана его трёхмерная компьютерная модель.

Для создания компьютерной модели использовалось графическое приложение для создания компьютерной модели использовалось графическое приложение 3D Studio Max Release 3.1. Разработанная модель позволяет отследить характер следования и траекторию движения на грунте, наличие колебаний оси колеса, определить потенциал преодоления препятствий.

Смоделированы следующие ситуации работы движителя: движение по твёрдой опорной поверхности; движение по поверхности с низкой несущей способностью; преодоление ступенчатых преград (лестничных маршей); линейно расположенных дискретных преград (сооружений и конструкций с периодической структурой); преодоление хаотично расположенных дискретных преград (обломков строительных сооружений, пней, брёвен).

Результаты компьютерного моделирования подтвердили теоретические расчёты в отношении кинематики привода, его энергетических показателей, способности преодолевать препятствия.

Приведенные исследования выявили: необходимость управления колёсно-шагающими движителями мобильных машин с помощью адаптивных систем. В настоящее время проводятся работы по созданию аппаратуры управления на базе микропроцессорной техники, включающей в себя системы искусственного зрения и ориентации в пространстве. Применение адаптивных систем управления колёсно-шагающими движителями обеспечит выбор рациональных режимов

движения в сложных условиях, повысит курсовую устойчивость и безопасность движения таких машин.

Транспортные средства с разработанными движителями могут использоваться как в сельском хозяйстве, так и на лесоразработках, при обследовании заражённых или труднодоступных территорий в том числе и морского дна.

Список литературы

1. Скойбеда А.Т. Колёсно-шагающий движитель мобильных машин/ А.Т. Скойбеда, И.М. Комяк, Д.А. Грамович, Д.А. Егоров// Современные методы проектирования машин. Расчёт, конструирование и технология изготовления. Сб. науч. тр. вып. 1. В 3-х т. – т. 2/ Под общей ред. П.А. Витязя. – Мн.: УП «Технопринт», 2002. – с. 182-186

2. Скойбеда А.Т. Колёсно-шагающий движитель и его динамические преимущества по сравнению с колесом/ А.Т. Скойбеда, И.М. Комяк, В. Н. Жуковец// Механика 2011: сб. науч. тр. V Белоруск. конгресса по теоретической механике, Минск, 26-28 окт. 2011.: в 2-х т./ объединенный институт машиностроения НАН Беларуси; ред. : М.С. Высоцкий [и др.] – Минск, 2011. – т. . – с. 138-144.

UDC 629.11.073+(016.3)

THE MAKING OF DRIVES OF WHEEL-WALKING PROPULSORS OF MOBILE MACHINES WITH ADAPTIVE CONTROL

Alechnovitsch N. A.

Belarussian national technical university, Minsk, Republic of Belarus

Komyak I. M. – senior lecturer

Annotation. The construction of a computer model of a wheel-walking propeller with adaptive control to obtain the kinematic characteristics of the movement of the machine in various operating conditions.

Keywords: propulsor, kinematics of the wheel-walking mechanism, adaptive control.

УДК 004.424.2

ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ МЕХАТРОННОЙ СИСТЕМЫ ПАРАЛЛЕЛЬНОЙ КИНЕМАТИКИ НА ГИБРИДНОМ ПРИВОДЕ ПРЯМОГО ДЕЙСТВИЯ

Кузнецов В.В.

Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь

Карпович С.Е. – доктор техн. наук, профессор

Аннотация. Предложенная в работе мехатронная система параллельной кинематики на гибридном приводе прямого действия рекомендуется для использования при создании прецизионных систем перемещений сборочного и оптико-механического оборудования микро- и нанoeлектроники с разрешением до нанометрового уровня.

Ключевые слова: мехатронная система параллельной кинематики, привод прямого действия, кинематика, динамика, имитационное моделирование

На основании метода синтеза схем исполнительных механизмов [1-3] параллельной кинематики для мехатронных систем перемещений была обоснована мехатронная система перемещений с шестью степенями свободы на гибридном шестикоординатном приводе прямого действия и исполнительном механизме параллельной кинематики в виде раскрывающегося тетраэдра. Предложенное решение защищено патентом на полезную модель.

Разработана математическая модель аналитического решения в явном виде функций положения всех звеньев в зависимости от входных параметров гибридного привода включая функции скоростей всех характерных точек и вычисления соответствующих линейных и угловых скоростей и ускорений.

Выбранный механизм параллельной кинематики с шестью степенями свободы (рисунок 1) состоит из трех параллельных кинематических цепей, содержащих три подвижных треугольных шатуна в виде двойных элементов 11–12, 13–14, 15–16, связанных с одной стороны вращательными шарнирами между звеньями 17 и 23, 18 и 24, 19 и 24, 20 и 25, 21 и 25, 22 и 23 с подвижной треугольной платформой 26, а с другой стороны связанных сферическими шарнирами 27, 28, 29 с тремя автономно управляемыми линейными шаговыми двигателями 8, 9, 10, линейные направляющие которых 5, 6, 7 с одной стороны, шарнирно соединены между собой, а с другой стороны жестко закреплены на соответствующих подвижных сегментных модулях 2, 3, 4 кольцевого шагового двигателя, неподвижная направляющая которых представляет основание замкнутого кольцевого статора 1. Он работает следующим образом. При секторных перемещениях трёх сегментных модулей 2, 3, 4 в соответствии с заданными на входе системой управления законами их движения по неподвижной кольцевой направляющей основания статора 1 кольцевого шагового двигателя и одновременно при линейных перемещениях, реализуемых тремя линейными шаговыми двигателями 8, 9, 10 в соответствии с заданными на входе с системой управления законами их движения по подвижным линейным направляющим 5, 6, 7 через сферические шарниры 27, 28, 29, шатуны 11–12, 13–14, 15–16 и соответствующие вращательные шарниры 17 и 23, 18 и 24, 19 и 24, 20 и 25, 21 и 25, 22 и 23 приводится в пространственное движение платформа 26.

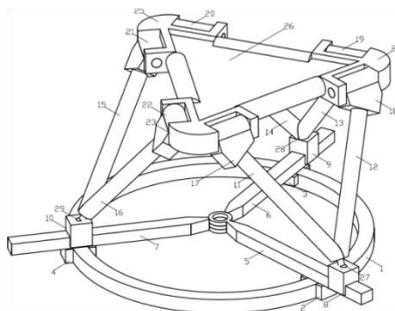


Рисунок 1 – Схема мехатронной системы на шестикоординатном гибридном приводе прямого действия

В зависимости от конкретно задаваемого секторного положения относительно статора 1 сегментных модулей 2, 3, 4 и конкретных положений линейных подвижных модулей 8, 9, 10 рабочая платформа 26 принимает однозначное и соответствующие им положение и ориентацию в трёхмерном пространстве. Тем самым предложенным механизмом параллельной кинематики обеспечивается реализация в мехатронной системе перемещения платформы с шестью степенями свободы в трёхмерном пространстве.

Предложенный гибридный привод, схемное решение которого представлено на рисунке 1.3, состоит из кольцевого привода прямого действия с тремя степенями свободы и трёх линейных шаговых двигателей, закреплённых на трёх сегментных подвижных модулях.

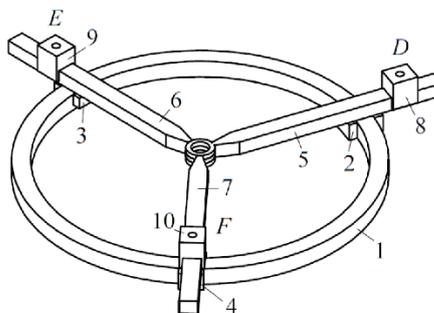


Рисунок 2 – Гибридный привод на кольцевом и линейных шаговых двигателях

Таким образом с помощью предложенного механизма параллельной кинематики могут быть реализованы прецизионные движения с шестью степенями свободы по шести независимым координатам в трёхмерном пространстве, включая три линейные и три угловые, обеспечивая в то же время высокие кинематические и динамические характеристики перемещений при высокой точности их реализации. При этом обеспечивается реализация дополнительного управляемого разворота вокруг вертикальной оси в диапазоне ± 360 градусов, что расширяет область управляемого движения с перспективой построения на его основе современного прецизионного, фрезерного, сверлильного оборудования, прецизионных промышленных роботов и платформ, например, для телескопов, лазеров, телекоммуникационных антенн.

Предложенные в работе математические модели, алгоритмы и программное обеспечение имитационного моделирования разработанной мехатронной системы перемещений на исполнительном механизме с шестью степенями свободы и гибридном приводе прямого действия могут быть рекомендованы к использованию при разработке проектов встраивания рассматриваемой системы в конкретное оборудование, которое заключается в численном формировании рабочей области, калибровки кинематических параметров и траекторных характеристик и формировании алгоритма управления в соответствие с технологической операцией в конкретное оборудование.

Дальнейшие исследования по данной работе предполагается проводить в направлении разработки алгоритмической основы построения системы управления на современной элементной базе с оптимизацией использования регуляторов, реализующих требуемое качество формирования траекторий в режиме реального времени.

Список литературы

1. Кузнецов, В.В. Формирование шаговых траекторий для реализации тригонометрической функции / В.В. Кузнецов, А.Ф. Марко // Теоретическая и прикладная механика [Электронный ресурс] : международный научно-технический сборник / Белорусский национальный технический университет ; редкол.: А. В. Чигарев (пред. редкол.). – Минск : БНТУ, 2020. – Вып. 35. – С. 146–149.
2. Карпович, С.Е. Алгоритмизация формирования шаговых траекторий для систем перемещений на шаговых двигателях / С.Е. Карпович, В.В. Кузнецов // Теоретическая и прикладная механика [Электронный ресурс] : международный научно-технический сборник / Белорусский национальный технический университет ; редкол.: А. В. Чигарев (пред. редкол.). – Минск : БНТУ, 2020. – Вып. 35. – С. 167–171.
3. Кузнецов, В.В. Алгоритмизация бесколлизионных перемещений трёх планарных позиционеров на одном статоре / В.В. Кузнецов, В.В. Поляковский // Актуальные вопросы машиноведения : сборник научн. трудов / Объед. инст. машиностр. Нац. академии наук Беларуси – Минск, 2019. – Вып. 6. – С. 64–67.
4. Кузнецов, В.В. Алгоритмизация математических моделей и имитационное исследование мехатронной системы перемещений с шестью степенями свободы / В.В. Кузнецов // Сборник научных работ студентов Республики Беларусь «НИРС 2018» / редкол. : И. А. Старовойтова (пред.) [и др.]. — Минск : Изд. центр БГУ, 2019. – С. 171–176.
5. Карпович, С.Е. Мехатронные системы параллельной кинематики для фотолитографического оборудования / С.Е. Карпович, В.В. Кузнецов, А.Ю. Войтов // Теоретическая и прикладная механика [Электронный ресурс] : международный научно-технический сборник / Белорусский национальный технический университет ; редкол.: А. В. Чигарев (пред. редкол.). – Минск : БНТУ, 2019. – Вып. 34. – С. 17–22.

6. Кузнецов, В.В. Алгоритмизация прямой задачи кинематики мехатронной системы на шестикординатном гибридном приводе прямого действия / В.В. Кузнецов // Теоретическая и прикладная механика [Электронный ресурс] : международный научно-технический сборник / Белорусский национальный технический университет ; редкол.: А. В. Чигарев (пред. редкол.). – Минск : БНТУ, 2019. – Вып. 34. – С. 195–200.

UDC 004.424.2

SIMULATION OF A MECHATRONIC SYSTEM PARALLEL KINEMATICS ON HYBRID DRIVE DIRECT ACTION

Kuznetsov V.V.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Karpovich S.E. – Ph.D., Full Professor

Annotation. The proposed mechatronic system of parallel kinematics on a hybrid direct-action drive is recommended for use in creating precision systems for displacement of assembly and optomechanical equipment for micro- and nano-electronics with a resolution up to the nanometer level.

Keywords. mechatronic parallel kinematics system, direct drive, kinematics, dynamics, simulation

УДК 004.424.2

ВЕРСИОНИРОВАНИЕ И КОНТРОЛЬ ЦЕЛОСТНОСТИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ СИСТЕМ РЕАЛЬНОГО ВРЕМЕНИ

Марко А.Ф.

*Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь*

Карпович С.Е. – доктор техн.наук, профессор

Аннотация. Представлено программное обеспечение для версионирования и контроля целостности систем реального времени, как программный элемент расширения Visual Studio.

Ключевые слова: программное обеспечение, контроль целостности, версионирование

С развитием электрических приводов и возможностей их применения в индустриально-производственных и транспортных системах, стала очевидна необходимость полной интеграции составляющих элементов электропривода: механики, электрических машин, силовой электроники, микропроцессорной техники и программного обеспечения для наиболее полного использования возможностей современного электропривода, и построения на его основе многокоординатных мехатронных систем перемещения [2]. Традиционные системы перемещений используют ПЛК-технология для выполнения задачи управления и включают в свой состав следующие аппаратные и программные элементы: ПК для визуализации, ПЛК с различными сопроцессорами, ввод/вывод через полевую шину, управление движением через параллельный интерфейс LPT, операционную систему и различные языки программирования.

Проведённый анализ современных программно-аппаратных средств показал, что наиболее эффективной технологией для реализации управления системами многокоординатных перемещений в режиме реального времени является технология EtherCAT, внедрение которой требует разработки дополнительных программных приложений [1]. В настоящей работе рассматривается программное обеспечение для версионирования и контроля целостности при управлении системами многокоординатных перемещений в режиме реального времени. Разработка данного программного обеспечения выполнялась в интегрированной среде разработки Visual Studio и комплексной системе управления версиями Team Foundation Server [3].

Для управления в реальном времени всё большее распространение получает технология EtherCAT. EtherCAT-технология предоставляет разработчикам систем управления технологическими процессами и сложным оборудованием полностью интегрированное решение, обеспечивающее стандартную и надёжную сеть обмена управляющей информацией. При этом количество задействованных полевых шин и интерфейсов уменьшается, обеспечивая тем самым унификацию всех процессов управления, гибкость структуры при практически неограниченном количестве устройств и малое время реакции на события, а также обеспечивается возможность переконфигурирования системы управления без необходимости её полного отключения. В связи с наметившимся внедрением технологии EtherCAT в прецизионное технологическое оборудование актуальной и важной является разработка специальных инструментов, позволяющих разрабатывать программное обеспечение системы управления в множестве версий и тем самым с постоянным изменением кода. При разработке и эксплуатации программного обеспечения (ПО) для систем перемещений важной задачей является обеспечение их целостности, необходимой для предотвращения незапланированных изменений. Контроль за целостностью в предложенном ПО обеспечивается на этапе разработки с помощью внедрения процесса версионирования в интегрированную среду разработки Visual Studio (VS) и систему управления версиями Team Foundation Server (TFS), а на этапе эксплуатации – через формирования и сравнение контрольных сумм. В процессе проектирования программного средства было выделено два функциональных модуля: версионирования и контроля целостности (рисунок 1).

Модуль версионирования реализован в виде расширения VS и позволяет обновлять версии ПО при внесении изменений. Версионирование возможно в двух основных режимах: в режиме с версионированием только проектов, файлы которых находятся в текущих изменениях и в режиме с версионированием всех проектов на основе истории их изменений.

В процессе разработки была реализована концепция [3] формирования версий для проектов, как принадлежащих к версионизируемому решению, так и для проектов, подключённых из других решений по ссылке.

В свою очередь проекты, принадлежащие к версионизируемому решению, подразделяются на

основные проекты, которые являются источником версии последнего релиза и обычные проекты. Определение типа проекта выполняется с помощью структурного анализа файла решения и конфигурационных файлов.

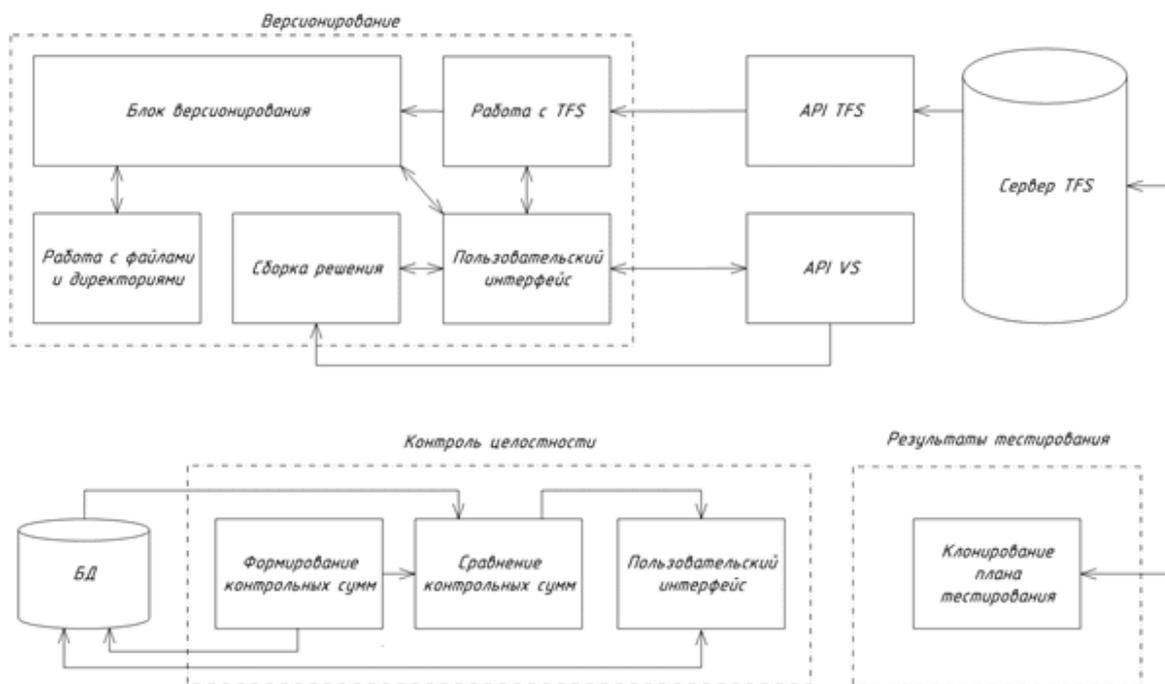


Рисунок 1 – Схема взаимосвязи функциональных модулей

Версия проекта состоит из двух частей: ручной части (первые три старших разряда версии), определяемой последней версией релиза и автоматической части, соответствующей номеру сохранения в системе TFS, в котором был изменён проект. Каждый проект содержит текстовый файл AssemblyInfo, который хранит версию проекта. Для её получения или изменения используются регулярные выражения. В результате после сборки релиза будут сформированы исполняемые файлы, в которых сохранится номер актуальной версии.

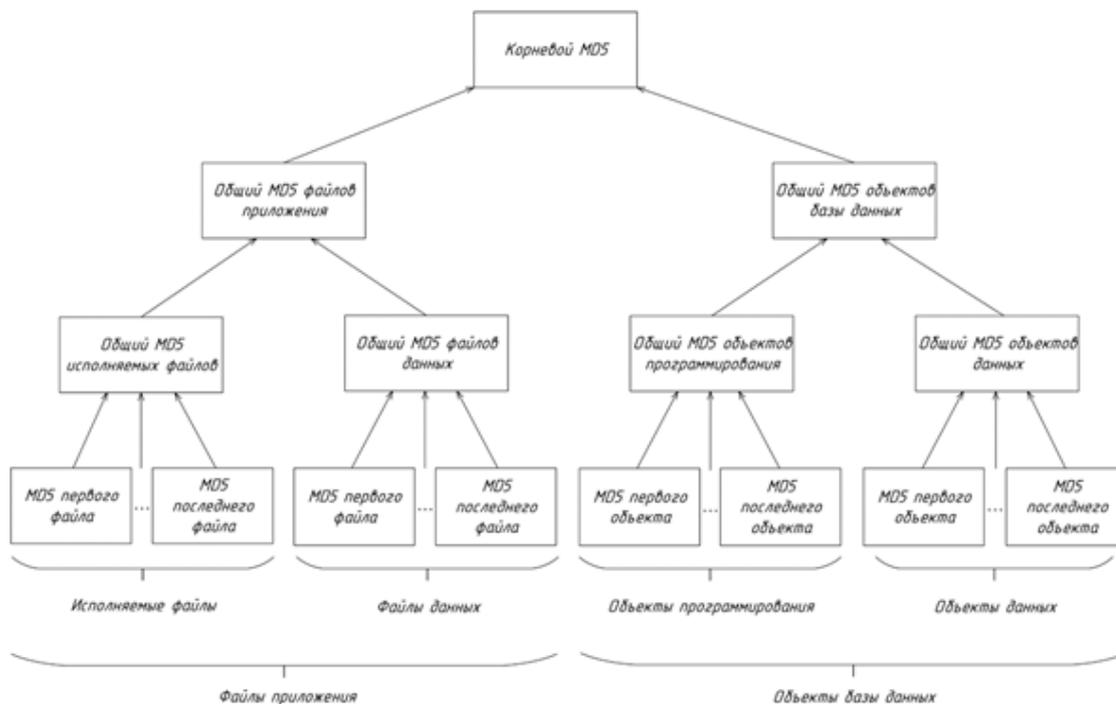


Рисунок 2 – Схема процесса формирования контрольных сумм

Модуль контроля целостности предназначен для определения незапланированных изменений данных в процессе эксплуатации. Для обеспечения целостности данных в процессе эксплуатации необходимо учитывать, что некоторые данные не могут оставаться неизменными, поэтому выделяются части, которые должны быть неизменными в процессе эксплуатации, и части, которые могут изменяться.

Программная часть любой системы, как правило, состоит из множества различных объектов, таких как исполняемые файлы, файлы данных и объекты баз данных. Формирование контрольных сумм имеет многоступенчатый характер и выполняется для каждого типа по-разному. Основная цель многоступенчатого формирования контрольных сумм заключается в удобстве представления информации о состоянии каждой подгруппы и в повышении скорости нахождения изменённых файлов.

Заключение. В работе решена задача по разработке программного средства, позволяющего внедрить процесс версионирования в среду VS и систему TFS, контролировать целостность ПО в процессе его эксплуатации.

Список литературы

1. Системы многокоординатных перемещений в исполнительные механизмы для позиционного технологического оборудования / С. Е. Карпович, [и др.]. – Минск : Бестспринт, 2013. –208 с.

2. Аваков, С.М. Система перемещений для оптико-механического оборудования микроэлектроники / С.М. Аваков, В.В. Жарский, С.А. Русецкий // Доклады БГУИР. – 2007. – № 6. – С. 39–43.

3. Программирование расширения интегрированной среды разработки VS2017 / А. Ф. Марко, К. В. Чеушев. – Технические средства защиты информации : тез. докл. XVI Белорус.-рос. науч.-техн. конф., Минск, Респ. Беларусь, 5 июня 2018 г. / Белорус. гос. ун-т информатики и радиоэлектроники. – Минск, 2018. – С. 63.

UDC 004.424.2

VERSION AND INTEGRITY CONTROL OF SOFTWARE FOR REAL-TIME SYSTEMS

Marko A.F.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Karpovich S.E. – Ph.D., Full Professor

Annotation. The software for versioning and monitoring the integrity of real-time systems is presented as a software element of the Visual Studio extension.

Keywords. software, integrity control, versioning

СЕКЦИЯ «ФИЛОСОФИЯ, ЛОГИКА И ЖИЗНЬ»

ХОРРОР КАК ТЕНДЕНЦИЯ СОВРЕМЕННОЙ КУЛЬТУРЫ

Зотова П.П.

*Белорусский государственный университет
г. Минск, Республика Беларусь*

Лагунова Е.Н. - преподаватель

В работе рассматривается концепция «антропологии ужаса» как философской парадигмы XXI века, проводится анализ возникновения и развития хоррора в литературе и театральном искусстве. Кроме того, автором поставлены задачи раскрыть интерес к феномену ужаса в современной философии; показать тенденцию перехода от трагедии к ужасу; выявить причины востребованности жанра хоррор в современной массовой культуре.

В последние десятилетия феномен ужаса – это средоточие интереса интеллектуального направления философии и доминанта культуры. Широкая распространенность проблемы ужаса доказывает существование такой парадигмы философии XXI в., как «антропология ужаса».

Существующие антропологические модели исчерпаны, происходит активный поиск новых, находящихся вне традиционной рационалистической философии. Ужас – одна из проблем, не замеченная классической философией.

В XIX в. ужас как составляющую структуры духовного опыта новой философии предсказывали Фридрих Ницше, Серен Кьеркегор и Федор Михайлович Достоевский.

В экзистенциальной философии ужас раскрывает бытие. Ужас и смерть (ничто) взаимодополняемы. Смерть ужасна, а ужас смертелен. Ужас становится позитивной онтологической категорией.

Понятие «ужас» меняет взгляд на антропологию, открывает ее непсихологический смысл. Новатором в разработке онтологических аспектов ужаса стал немецкий философ Мартин Хайдеггер, а этических – русский мыслитель Николай Бердяев. Изучение ужаса объединило русскую и западную философские школы.

Хайдеггером изложены различия понятий «ужас» и «страх». Из-за схожести этих понятий ужасом ошибочно именуют страшное, а страхом – ужасное. Принципиальное отличие ужаса от страха: ужас показывает присутствию его собственность и несобственность, раскрывая возможность бытия. Ужас – состояние бытия, где человек находит фундаментальные смыслы своего существования. Через ужас возможно найти саму сущность бытия и осознать подлинность существования.

Определение нравственного значения ужаса, как и противопоставление страха и ужаса принадлежит Бердяеву. Страх эмпирически мотивирован – ужас трансцендентен. Ужас способен преображать, а страх – нет.

В XIX в. ужас экзистенциально значим, но связан со страхом категорией безобразного, катарсисом. В конце XX – начале XXI вв. ужас становится самостоятельной категорией.

Традиционно ужас – элемент начала трагедии. В XX в. ужас проникает в повседневность. Механизм ценностной трансформации, когда трагедия сменяется ужасом и абсурдом, показан русским философом и писателем Владимиром Карловичем Кантором. Теперь человеческое бытие описывается категориями – «превращение», «приговор», «процесс».

В эстетической сфере реабилитируется категория безобразного. Ужасное становится основой самостоятельной структуры безобразного. Судя по обилию монструозных форм в современном искусстве, можно сказать, что различие между безобразным и прекрасным становится архаизмом. Вездесущность современного ужаса обусловлена потерей смысла. Пустота – метафора ужаса, ужас – сущность пустоты. Этим же объясняется и популярность жанра «horror».

Хоррор – жанр литературы и кино, цель которого – напугать реципиента, вселить чувство тревоги и страха, создать напряженную атмосферу ужаса, передавая эффект «саспенс».

При переживании страха существование становится более ярким. Притягательность ужаса – основа хоррора. Дозированный страх – антитеза скуки, и в таком качестве, как отмечает норвежский философ и писатель Ларс Свендсен, он представлен в Античности и Средневековье. В середине XVIII в. ужасающее – центральная категория эстетики.

Остановимся подробнее на том, каким образом хоррор предстает в таких видах искусства как литература и театр.

Корни современного литературного хоррора уходят в фольклор и мифологию. Там уже затронуты вопросы, волнующие человека до сих пор. Там же представлены и первые антропоморфные существа, добрые и злые духи. Античные писатели неосознанно апеллируют к эмоциям человека, с которыми тот не всегда способен совладать.

Средневековая эпидемия чумы способствует появлению сюжетов с участием скелетов и гниющих трупов. Перу Марии Французской принадлежат истории об оборотнях. Очень популярны сюжеты о Владе III Цепеше – графе Дракуле.

Страхи человека в большинстве своем существуют вне времени. Потому сюжеты, заимствованные из мифологии и фольклора, остаются актуальными и в XXI в.

В жанре хоррор пробовали себя многие русские писатели-классики – А.С. Пушкин, Н.В. Гоголь, И.С. Тургенев. Среди особенностей литературы ужасов Серебряного века отметим следующие:

- Инверсия. Привычные вещи здесь переворачиваются с ног на голову, даже наука вместо двигателя прогресса превращается в оружие Дьявола («Студный бог», В.Я. Брюсов). Зазеркальный мир – аналог существования души после смерти. В «страшных» новеллах Серебряного века традиционный мифологический образ оборотня сменяется зверем, вынужденным носить человеческую личину («Черный Дик», Н.С. Гумилев). Будущее представляет собой умирающее человечество. Людей уничтожают либо машины, либо оторванность от природы, либо эпидемия. Тесно взаимосвязаны образы живого и мертвого. Даже Дьявол из искусителя превращается в проводника между мирами.

- Перверсия. Быт, ценности и мир извращаются. Убийство и пытки приносят главным героям наслаждение, смерть – единственное благо. Извращаются законы природы. Перверсируются мечты («Теперь, когда я проснулся», «Республика Южного Креста», В.Я. Брюсов; «Красный смех», Л.Н. Андреев). Реальность подстраивается под законы хоррора.

- Телесность. Эта черта, появившаяся в литературе ужасов Серебряного века, более характерна для хоррора конца XX в. Она возникает на стыке литературоведения, культурологии, психологии, социологии и философии. В Серебряном веке акцентируется внимание на «маниакальности» и необычных страстях.

В «страшных» новеллах Серебряного века почти не фигурирует сюжет о духах и призраках. Причина тому проста: духи и призраки нематериальны, а монстры и люди телесны. Однако новеллы Серебряного века населены другими чудовищами – вампирами, живыми мертвецами, маньяками, оборотнями, эпидемиями безумия.

После 1920-х гг. хоррор как жанр русской литературы приходит в упадок, но находит свое место и укореняется в самой жизни. Любой может быть врагом. В литературе превалирует оптимизм. Терапевтическая функция хоррора – приносить своеобразную эмоциональную разрядку – не коррелирует с такой тенденцией. Поэтому жанр хоррора с поправкой на аудиторию находит себя в детской литературе. Частый локус современного русского хоррора – пионерский лагерь. Поскольку лагерь воспринимается пространством, никому не принадлежащим, где совершаются совместные регламентированные действия. Последнее 10-летие XX в. отмечено популярностью страшных рассказов «новых классиков» – Виктора Пелевина, Людмилы Петрушевской.

В театральном искусстве, давнем спутнике литературы, также заметен интерес к хоррору. С Античности страшное – неотъемлемая часть театральной эстетики. В народной культуре – это мистические карнавалы и маски. В Средневековье – площадные мистерии с натуралистичным изображением пыток и казней.

В начале XX в. иррациональный страх и тревога как возможность передать «трагедию каждого дня» интересны представителям «новой драмы». В драме рубежа XIX–XX вв. ужасное – угроза, рожденная из бессознательного внутри героя. Открыть трагизм через психоанализ становится возможным благодаря концепции Postmemory Марианны Хирш.

Работая с ужасным, театр использует кинематографические способы нагнетания напряжения в пространстве сцены. Место действия пьесы – предельно безликое пространство, отражающее некие состояния героев и намекающее на волю дома. Главный герой – ужас столкновения человека и непостижимого, «враждебного», живущего в самом человеке.

В хоррор-спектакле речевая партитура погружает зрителя в напряженную ирреальную атмосферу, позволяя за внешне бытовым сюжетом услышать мистической истории. Элементы, присущие хоррору (нагнетание напряженной атмосферы музыкой, игрой со временем, обнаружение в персонажах зловещего, жуткие места), вплетены в интеллектуальную драму, выявляют ее суть, временами доводя события пьесы до гротеска.

Таким образом, хоррор как тенденция современной культуры, явно видимая и проговариваемая в искусстве и философии, и неявно присутствующая в других социальных сферах человеческой деятельности, свидетельствует о фундаментальной трансформации культуры и смене приоритетов в философской антропологии, остро реагирующей на актуальные процессы в жизни общества.

Список использованных источников:

1. Бердяев, Н. А. О назначении человека. Опыт парадоксальной этики / Н.А. Бердяев. – Москва: АСТ; Фолио, 2003. – 701 с.
2. Кантор, В. К. Ужас вместо трагедии (творчество Франца Кафки) / В.К. Кантор // Вопросы философии. – 2005. – № 12. – С. 65-77.
3. Козьмина, Е.Ю. Авантюрно-философская фантастика XX века и философская повесть / Е.Ю. Козьмина // Вестник Костромского государственного университета им. Н.А. Некрасова. – 2015. – № 5. – С. 96– 100.

4. Муратова, И.А. Телесность как доминанта культуры постмодерна/ И.А. Муратова // Исторические, философские, политические и юридические науки, культурология и искусствоведение. Вопросы теории и практики. – 2013. – № 1. – С. 143–146.

5. Хирш, М. Что такое постпамять [Электронный ресурс]. – Режим доступа: <https://urokiistorii.ru/articles/chto-takoe-postpamyat> – Дата доступа: 19.04.2022.

6. Хайдеггер, М. Бытие и время / М. Хайдеггер. – Москва: Академический Проект, – 2011. – 460 с.

ПРОБЛЕМЫ ИНФОРМАЦИОННОГО ОБЩЕСТВА

Павлович Ю.В., студент гр.140301

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Шепетюк В.В. – канд. филос. наук, доцент кафедры философии БГУИР

В работе рассматриваются основные проблемы информационного общества на современном этапе его развития. Обращается внимание на фундаментальный характер преобразований во всех сферах жизни человека, связанных с информационно-коммуникационными технологиями, а также на ряд опасностей и рисков, которые несет с собой данный тип общества, не всегда позволяющих рассматривать его как гармоничный проект желаемого будущего для человеческой цивилизации.

Термин «информационное общество» ввел в научный оборот профессор Токийского технологического института Ю. Хаяши еще в 1969 году. Прошло 50 лет и сегодня термин «информационное общество» прочно занял свое место в лексике политических деятелей, экономистов, преподавателей и ученых. В большинстве случаев это понятие ассоциируется с уровнем развитием информационных технологий и средств телекоммуникации [1].

Информационное общество – новая историческая фаза развития цивилизации, в которой главными продуктами производства являются информация и знания.

На современном этапе информация является главным источником развития общества. Проникновение информационных технологий практически во все социально-экономические сферы влечет за собой появление новых потребностей и нового образа жизни человечества. Информационное общество формируется в основном под влиянием двух процессов: интенсивный технический прогресс (внедрение компьютерных технологий во все сферы общественной жизнедеятельности людей) и мощный поток информации (содержание и актуальность научных знаний быстро изменяется (4-5 лет) в связи с новыми потребностями развивающегося общества).

С ростом коммуникаций появилась возможность достаточно быстро получить самую разнообразную информацию без особых усилий. Это расширяет возможности человека, способствует выбору им наиболее оптимального и эффективного способа его деятельности, но, с другой стороны, стимулирует появление эффекта лени и пассивности в силу повышенной доступности информации и знаний. Таким образом, информационное общество ориентируется на развитие творческого и интеллектуального потенциала, как индивида, так и общества в целом, однако, в тоже время создает благоприятные условия для культивирования потребительского мировоззрения, нивелирующего значимость как научно-технических решений, так и традиционных гуманистических ценностей, выработавшихся человечеством столетиями.

Примечательно, что на протяжении ряда лет в Беларуси действует государственная программа поэтапного перехода к информационному обществу – «Электронная Беларусь», которая предполагает создание единой информационной инфраструктуры с широким использованием информационных технологий в государственном управлении, народном хозяйстве, образовании, медицине, торговле и других областях жизнедеятельности [2].

Сегодня Беларусь уже имеет ряд серьезных достижений в области внедрения информационных технологий. Например, внедрена и успешно действует автоматизированная система коммунальных платежей. В медицине внедряются автоматизированные информационные системы, призванные освободить врачей от устаревшей бумажной работы. В сфере образования действует система централизованного тестирования и обработки экзаменационных бланков, что позволило увеличить скорость их обработки в 10 раз и свести вероятность ошибки к минимуму. Разработан национальный интернет-портал «Культура Беларуси», а также информационно-технологическая платформа для ресурсов Национальной библиотеки, что позволяет всю книжную продукцию, издаваемую в республике, регистрировать в электронном виде и оперативно формировать сводные электронные каталоги. Постоянно совершенствуется работа по внедрению информационных технологий в деятельность таможенных и налоговых органов, государственной автомобильной инспекции и т.д.

Отмечая позитивные стороны информационного общества, не следует забывать и об отрицательных его моментах и проблемах.

Одна из серьезных проблем информационного общества - это проблема осуществления защиты информации и противодействия информационной преступности. Все больше появляется компьютерных специалистов, действующих вне закона – хакеров, основным занятием которых является получение несанкционированного доступа к компьютерам, базам данных и другой информации. Огромное количество распространенных вирусных программ требуют от пользователей осторожности, владения необходимыми навыками защиты личной информации и своих цифровых устройств. Наконец, новым явлением в области информационной преступности

является информационный экстримизм и терроризм, в результате которого может дестабилизироваться функционирование не только конкретных информационных систем, но и целых государств, что может вести к социальным конфликтам, войнам и гибели людей.

В дополнение к этому можно выделить еще одну очень важную проблему информационного общества – проблему выбора. Что выбрать потребителю из огромного количества товаров, услуг, предлагаемых поисковыми системами? В приложении не только к экономике, но и к сфере культуры аналогичным образом будет актуальная проблема т.н. потребительской навигации. В год производится 700 тысяч песен, 6 тысяч кинофильмов, миллионы текстов и книг. Что из этого следует почитать, послушать, посмотреть в первую очередь? Ведь даже если человек не будет пить, есть и спать, а будет только слушать современную музыку, ему все равно не удастся ознакомиться с малой толикой того, что появляется каждый год.

Таким образом, один человек уже не может качественно систематизировать информацию, рассмотреть ее достаточно объективно с различных сторон. Ему требуются помощники в виде СМИ, финансовых аналитиков, политологов, маркетологов, экспертов, блогеров, других авторитетных медийных личностей, представляющих массивы данных в упрощенном и адаптированном для среднестатистического обывателя виде. Все это в итоге сужает мировоззренческий горизонт личности, ее эрудированность и предоставляет широкие возможности манипулирования обществом с помощью специально подобранной информации, преследующей цели, которые зачастую идут вразрез с гармоничным развитием цивилизации.

Одним из способов решения этой проблемы будет, видимо, противоречивое, но необходимое усиление контроля со стороны государственных органов за процессами производства, обработки и распространения информации, что уже и происходит в некоторых странах (например, в Китае).

По своей психической сущности люди достаточно внушаемы и потому пока еще мало изученную опасность для человека в информационном обществе представляет психологический феномен, называемый виртуализацией общества. Его суть заключается в том, что реальные физические объекты, процессы и явления подменяются их виртуальными образами, моделями, которые очень похожи на отображения объективной реальности, но таковыми не являются. Именно эти свойства, а также высокая динамичность информационной сферы общества и позволяют создавать в нем виртуальную реальность, которая и воспринимается человеком, наряду с реальностью физической [3, с. 21].

Потеря личностью способности ориентироваться в постоянно меняющемся внешнем мире, перенасыщенном информацией может сформировать потребность в использовании различных средств ухода от стрессогенной действительности: в употреблении наркотиков, алкоголя, а также привести к чрезмерному увлечению виртуальными мирами, снижению способности человека к взаимодействию с окружающей средой и другими людьми, самоизоляции, апатии и даже социофобии. Уже сейчас выделен специальный термин – т.н. «киберболезнь», обозначающий широкий спектр паталогических постоянных людей, связанных с вредоносной зависимостью от виртуальной реальности. В настоящее время по всему миру распространено огромное количество геймеров, которые довольно успешно и комфортно живут и действуют в виртуальном мире компьютерных игр, а к обычному миру мало приспособлены [3, с. 22]. Иными словами, на мой взгляд, Интернет, как глобальное виртуальное пространство, не только помогает, объединяет, но и разделяет людей. Возможность виртуального общения подчас заменяет реальное общение, вследствие чего люди могут чувствовать себя все более одиноко и подавлено, не понимая причину этих болезненных явлений.

В заключении хочу отметить, что, по моему мнению, для предупреждения и преодоления отмеченных проблем обществу и государству нужно постоянно думать о повышении информационной культуры и гигиены общества, т.е. обучению людей правилам безопасного потребления и использования информации. Кроме этого крайне важно не просто внедрять повсеместно передовые технологические решения, но постоянно проверять их на предмет содействия сохранению в человеке его лучших качеств и свойств, чтобы человек оставался самостоятельным, здравомыслящим и дееспособным существом. Только в этом случае информатизация общества будет способствовать прогрессу, повышению интеллектуального потенциала и культурного уровня людей.

Список использованных источников:

1. Мачкарин, О.Д. Актуальные проблемы «информационного» общества [Электронный ресурс] / О.Д. Мачкарин // Вестник СПбГУ. – 2015. – сер. 17.2015. вып.2. – Режим доступа: <https://cyberleninka.ru/article/n/aktualnye-problemy-informatsionnogo-obschestva/viewer>. – Дата доступа: 30.03.2022.
2. Информационная культура, как решение современных информационных проблем [Электронный ресурс]. – Режим доступа: <https://sites.google.com/site/dianatinatkina2/-question2>. – Дата доступа: 30.03.2022.
3. Логвинова, С.А. Проблемы современного информационного общества [Электронный ресурс] / С.А. Логвинова // Известия ВГПУ. – 2014. – №1 (262). – Режим доступа: http://izvestia.vspu.ac.ru/content/izvestia_2014_v262_N1/izv%20VGPU%. – Дата доступа: 31.03.2022.

4. Государственная политика Российской Федерации в области развития информационного общества / А.В. Коротков [и др.]; под ред. А.В. Короткова. – М.: ООО «Трейн», 2007. – 472 с.

ОБРАЗЫ ЧЕЛОВЕКА В ДИНАМИКЕ ЕВРОПЕЙСКОЙ КУЛЬТУРЫ

Скалозуб К.А., студентка гр.150502

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Миськевич В.И. – канд. филос. наук, доцент

Аннотация. В тезисах рассматриваются факторы, влияющие на формирование образа человека. Эксплицируются разные временные периоды; в ходе сравнения их условий жизни, социального, экономического, политического развития выявляются причины тех или иных качеств характера человека.

Ключевые слова. Человек, личность, индивидуальность, история, влияние, эпоха, формирование, факторы.

Меня всегда интересовал вопрос формирования личности человека, его «души». Очевидно, что основные жизненные принципы и черты характера формируются в детстве, когда на нас в большей мере влияют родители. То, как они ведут себя в обществе и, в частности, в семье. Наш несформировавшийся разум запоминает их поведение, и мы думаем, что по такому образу нужно действовать и нам. Также велико влияние начального окружения, которое состоит из детей, подобных своим родителям. Но все люди изначально находятся под влиянием окружающей среды, в которой они живут, норм и традиций их страны. Всем нормам и принципам свойственно меняться с течением времени: меняется эпоха – меняется все. Как бы мы этого не хотели и не сопротивлялись, наши поступки всегда зависят от времени, в котором мы живем. Если мы сейчас заходим жить по принципу и традициям средневековых людей, у нас бы ничего не вышло.

Начнем с самого начала, а именно, кто такой человек.

Человек – высшая ступень развития живых организмов на Земле. Человек – существо, обладающее разумом, абстрактным мышлением и речью.

Актуальность рассматриваемой мной проблемы заключается в том, что человек изучался, изучается и будет изучаться многими науками, одними из которых являются социология, психология, физиология, психология, медицина. На сегодняшний день не утверждена точная история происхождения человека. Существует несколько версий данного явления, но установить что-то новое или сократить то, что имеется до более конкретного утверждения практически невозможно.

Чтобы выявить факторы, наиболее существенные для формирования личности и образа человека, я рассмотрела несколько эпох, начиная с периода греко-римской древности.

Немецкий философ Альберт Швейцер писал: «Древний грек — более гражданин, чем человек. Деятельная преданность интересам полиса воспринималась им как нечто совершенно естественное» [1]. В Древней Греции существовала такая система образования, в которой акцент делался на формирование личности и определение ценностных ориентаций человека. Главной задачей тогда было не сформировать профессионала в какой-либо отрасли, а разносторонне развитого человека.

Образ «идеального героя» составлялся на основе поэм Гомера, в которых молодой человек был образован, в совершенстве развит физически, обязательно знал поэзию и обладал навыком игры на музыкальных инструментах. Существует и другой образ молодежи: знание поэм заменяется знанием основ римского законодательства и нравственности.

Полис – древнегреческий город – гарантировал человеку свободу, социальную безопасность и возможность осуществления его стремлений. Благодаря таким условиям существования, человек чувствовал себя независимым, значительным и самостоятельным, продолжая развиваться.

В Средневековье ситуация была другая. Проявления индивидуальности, смелости и честности обесценивались. Учитывая то, что селения были изолированы, информация в обществе распространялась очень медленно, в отличие от слухов, и становится неудивительным состояние людей, недалекое от истерии и постоянные перепады настроения.

Слова Достоевского, которые выразили смысл средневековой библейской драмы, можно отнести к человеку того времени: «Здесь Бог с дьяволом борются, а поле битвы – сердца людей» [1]. То есть, все действия происходят в человеческом сердце, которое в данной ситуации является только сценой, сам человек роли не играет.

«Распалась связь времен» - слова Гамлета, которые описывают следующий этап: эпоху Возрождения. Радикально поменялось отношение к жизни и личности [1]. Жизнь стала одной из

самых главных ценностей и возможностью реализовать себя, показать свою суть, самобытность, оставить свое имя в истории.

Возрождение можно считать временем индивидуализации личности. Придается значимость уникальности и неповторимости. В отличие от Средневековья, на произведениях искусства находились подписи, из которых можно было узнать имена создателей.

Люди стали неконтролируемы, потому что обладали правом индивидуального выбора, и никто не мог быть уверен, каким именно он будет. Крайняя попытка сохранить традиционные устои не закончилась успехом. Те события, которые произошли в эпоху Возрождения можно назвать революцией, так как границы норм были полностью изменены или даже разрушены, индивидуальность получила «свободу», стала наравне с всеобщностью и имела такие же права на существование.

В XVII в. человек стал полностью зависим от окружающей среды и общества, которое хочет удивить, впечатлить и убедить. Героя того времени характеризовали не действия, как это было в эпоху Возрождения, а эмоции и чувства. Бенедикт Спиноза достаточно прозрачно описывал «наличие» у человека свободы, когда говорил, что человек, думающий, что он свободен, подобен брошенному камню, который думает, что летит.

Человек – лишь малая частица природы, во всем подчиненная ее законам, существовавшим до людей и независимо от их мыслей и желаний. Мыслителем, ярко воплотившим дух этого времени, был французский философ Мишель де Монтень возвестил наступление глубокого перелома в духовном климате Европы. В общем и целом барокко – это стиль мышления и поведения людей «растерявшейся эпохи», разуверившейся во всем унаследованном и вместе с тем еще не нашедшей почвы для нового символа веры. Это был дух, буквально потрясенный открывшейся ему изменчивостью всего и во всем – реалий окружающего мира и способа их рассматривать и познавать, изменчивостью истин, за исключением одной – истины сомнения, присущей собственному уму. Наиболее верный путь к познанию себе подобного – познать себя. Сосредоточенность мысли на движении, на процессе означала, что важнее всего не бытие само по себе, а его бесконечные превращения, многоликость ипостасей. Это умонастроение ярко выразил английский поэт Джон Донн: «Дела обстоят тем лучше, чем чаще они меняются» [1]. В этом ключевая идея барокко. Во всем – динамизм, стремление к пределу, чрезмерности.

Человек XVIII в. ощущал единым не только окружающий мир, но и себя самого. В XIX в. картина мира множится, дробится, и человек начинает ощущать себя как сложное существо, живущее двумя, а то и более, жизнями. Вспомним слова одного из величайших композиторов XVIII в. Йозефа Гайдна, обращенные к крупнейшему музыканту следующего поколения Людвигу ван Бетховену: «Вы производите на меня впечатление человека, у которого несколько голов, несколько сердец и несколько душ» [1].

Велико было влияние истории, в частности трудов Ф. Гизо, О. Тьерри, А. Тьера, сформулировавших представление о классах и их борьбе. Реалисты развивают новый тип историзма, опираясь на эти открытия. Диалектика Гегеля координируется с диалектическим изображением жизни в борьбе противоречий у писателей-реалистов.

Совершенно исключительное значение для формирования концепции человека в XX в. имели работы создателя психоанализа Зигмунда Фрейда. Он обнаружил в человеческой психике глубинную биологическую «память» о всех предшествовавших стадиях его биологического развития. Эта так или иначе воздействующая на психику человека «память» (родовая, генетическая, органическая, биологическая) и есть грозное «Оно» — глубинная, бессознательная часть душевного аппарата человека, содержащая инстинктивные сексуальные и агрессивные влечения, которые ограничиваются, направляются и сдерживаются с помощью других структур («Я», «Сверх-Я», «Я-идеал»), возникших уже в процессе общественного развития и до определенной степени нейтрализующих «Оно» [1].

В итоге, что мы получаем? Первым и одним из главных факторов влияния на формирование личности является наследственность: многие черты характера достаются нам от родителей. Окружающая среда, культура, географическое положение местности, в которой человек живет: так или иначе, это все влияет на образ жизни индивида, деятельность и поведение в обществе. И наконец, самый главный фактор – это инициатива самого человека. В любом случае, каждый человек индивидуален, его мышление, видение ситуаций. Стоит признать, что сейчас мы обладаем свободой мнений, слова, можем выбирать тот вид деятельности, которая нам наиболее симпатизирует, чего не было у людей раньше. Человек меняется, развивается, совершенствуется, и это будет продолжаться всегда.

Список использованных источников:

1. Пустовит А.В. История европейской культуры: учеб. пособие – Киев: ДП «Издательский дом «Персонал», 2004 – 423 с.
2. Философский энциклопедический словарь – Ильичев Л.Ф., Федосеев П.Н., Ковалев С.М., Панов В.Г. Л.Ф. - М.: Советская энциклопедия, 1983. – 840 с.

IMAGES OF HUMAN IN THE DYNAMICS OF EUROPEAN CULTURE

Skalozub K.A., student gr.150502

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Miskevich V.I. – Candidate of Philos. sciences, associate professor

Annotation. The theses consider the factors influencing the formation of a person's image. Different time periods are explicated; in the course of comparing their living conditions, social, economic, political development, the reasons for certain qualities of a person's character are revealed.

Keywords. Man, personality, individuality, history, influence, era, formation, factors.

ПСИХОЛОГИЧЕСКИЙ АНАЛИЗ ЗНАЧЕНИЯ ОПЫТА ДЕТСТВА ВО ВЗРОСЛОЙ ЖИЗНИ

Катковская Е.А., студентка гр. 150502

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Миськевич В.И. – канд. филос. наук, доцент

Аннотация. В тезисах рассматривается проблема влияния детских психотравм на формирование личности взрослого человека. Затрагивается процесс воспитания ребёнка и его взросление, становление личности. Анализируются наиболее часто встречающиеся психологические особенности, появление которых тесно связано с детством.

Ключевые слова. Психотравма, семья, родители, ребёнок, синдром, негативное прошлое, переживания, травма, газлайтинг.

В последнее время я всё чаще и чаще стала замечать, что поведение людей цикличное и время от времени они совершают одни и те же ошибки, наступают на одни и те же грабли. Это всё связано с тем, что шаблоны нашего поведения формируются и закладываются в нашем подсознании ещё в детстве. В первые два года своей жизни ребёнок на уровне чувств учится определять, любят его или нет. Следующие три года жизни в основном определяют то, как он будет в течении всей жизни взаимодействовать с людьми. Родители ребёнка – его главные учителя. Общение с ними сильнее всего влияет на формирование личности индивида.

Психоанализ опыта детства во взрослой жизни – метод наблюдения и изучения личности с целью объяснения поведения человека неосознаваемыми детскими запечатлениями, переживаниями и решениями. Если мы говорим о проявлениях детских переживаний, которые, как и генетические особенности, определяют нашу сущность и самооценку, то мы имеем в виду ту часть личности, которую называют «внутренний ребёнок». Внутренний ребёнок – сумма позитивных и негативных эмоций, которые проживает ребёнок по мере взросления. Он живёт в каждом из нас и отвечает за наши желания, искренность, раскованность, честность и эмоциональность.

В процессе психотерапевтической работы с людьми, которые страдают от различных психологических проблем, оказывается, что причина недомоганий – травма, полученная в детские годы. Как правило, детскую травму нам наносят родители, родственники, первое социальное окружение. Формированию ранних детских психотравм способствуют такие факторы, как: сложность экономической ситуации в семье, своеобразие семейного уклада, низкий образовательный уровень родителей. Одной из главных причин психологического травмирования детей становится неблагоприятная обстановка в семье. Сюда входят вербальное и невербальное подавление и унижение, эмоциональное и физическое игнорирование, физическое или сексуальное насилие, проживание с депрессивным родителем, родителем с психическим заболеванием или родителем, имеющим вредные привычки, присутствие при насилии над одним из членов семьи, потеря родителя вследствие развода. Например, если ребёнок не чувствует любви и ласки, то становится беспокойным и пугливым. Если же ребёнок растёт в атмосфере ненависти и лжи (даже та, на которую родители не обращают внимание, например, когда мы хотели сладость или игрушку, нам часто говорили: «Это не продаётся»; «если смотреть телевизор больше часа в день – ослепнешь» и др.), то во взрослой жизни он сталкивается со сложностями контроля эмоций и недоверием.

Психологических проблем, с которыми сталкиваются люди на протяжении жизни очень много, но я рассмотрела наиболее часто встречающиеся, которые люди могут и не замечать.

Глубокую рану психике может нанести газлайтинг – одна из форм психологического насилия, тактика манипуляции, которая встречается повсеместно. Основной задачей газлайтинга является заставить свою жертву сомневаться в правдивости своих мыслей и чувств, правдивости окружающего мира, воспоминаний и жизненного опыта. Страдать от газлайтинга дети могут из-за родителей, которые так пытаются скрыть свои собственные ошибки, неправоту, неуверенность в себе, и укрепить свой авторитет в глазах ребёнка, сделав его послушным. Примером может служить отрицание правды («Чего ты плачешь, это же не больно?», «Мы тебе ничего не обещали, не придумывай себе», «Ничего такого не было»), способностей ребёнка и его правоты («Если бы не мы с папой, ты бы не справился», «Я старше, я лучше знаю»). Общась со сверстниками, ребёнок также может стать жертвой газлайтинга, ведь часто подростки укрепляют свой авторитет, унижая других. Однажды сказав девочке: «А ты бы хотела иметь красивые волосы?», – можно зародить в ней комплекс и неуверенность из-за внешнего вида, который будет преследовать её и во взрослой жизни. Становясь старше, мы встречаемся с неуверенностью в себе, боязнью довериться и открыться другим людям, сложностью в постановке целей и принятии решений,

проблемами с отстаиванием своих личных границ и собственного мнения. Мы не раз становились жертвами газлайтинга, даже не замечая этого.

Наиболее популярной проблемой, по моему мнению, является «синдром отличника» – убеждение того, что идеал в любой сфере деятельности может и должен быть достигнут, стремление быть лучше других. Отличие людей с таким синдромом от «перфекционистов» в том, что они ориентируются на окружающих, а не на качество работы. Такие люди ставят заведомо трудно выполнимые задачи, любят похвалу и вся их деятельность направлена в основном на то, чтобы получить признание и одобрение других людей. Такой синдром возникает у людей из-за того, что их недолюбили в детстве, и похвалу за хорошие оценки они воспринимают как проявление любви. Признание успехов помогает ненадолго повысить заниженную самооценку. Ещё одной из причин является то, что родители ориентируют ребенка на достижение целей, значимость которых признана обществом и которых они сами не смогли достичь, пренебрегая его интересами, желаниями и состоянием здоровья.

Многие люди страдают «синдромом отложенной жизни». Его проявления состоят в том, что человек считает, что не заслужил что-либо, не живёт настоящей жизнью, а только готовится к ней. Они откладывают важные дела и покупки до какого-то определённого момента: «вот когда похудею...», «получу премию ...», «вот когда перееду...» и др. Это связано с тем, что родители не уделяли должного внимания поступкам ребёнка, а порой и вообще забывали про него, вследствие чего не сформировали у ребенка механизм решения задач, требующих немедленного выполнения. А некоторые, наоборот, чрезмерно опекая и поощряя подобные поведения, разрешали ничего не делать. Ещё одна причина кроется в привычке наших родителей откладывать деньги на черный день, делать запасы продуктов, откладывать вещи для подходящего случая и т.д. Человек живет будущим, не ощущая настоящей жизни и ограничивает себя. Часто родители заставляли нас есть то, что мы не хотели, перед тем как разрешить нам сладости. И во взрослой жизни нам сложнее позволить себе приятное, без какой-либо на то причины.

Одной из психологических травм, проявляющихся во взрослом возрасте является «синдром жертвы». Это психологическое состояние, которое характеризуется чувством беспомощности и перекладыванием ответственности на окружающих людей. Он формируется как стратегия поведения для преодоления стрессовых и травмирующих ситуаций. Основная причина в том, что ребёнка сильно опекали и не научили преодолевать трудности, и, сталкиваясь с проблемами, он не справляется и становится подавленным, утрачивает веру в собственные силы. У человека возникает ощущение, что он не способен что-то менять и влиять на окружающий мир. В результате этого, чтобы защитить собственную психику от чрезмерных негативных эмоций, он перекладывает вину на судьбу, случай или окружающих, но не на себя.

Более редким, но достаточно серьёзным является «синдром Адели» – это психическое расстройство, характеризующееся любовной зависимостью и болезненной одержимостью объектом привязанности. Предрасполагающими факторами в появлении этого синдрома являются гиперопека со стороны семьи, недостаточная забота, нехватка внимания со стороны матери или отца. Объект привязанности ошибочно наделяется качествами, которые ребёнок не получил от родителей (добротой, щедростью, внимательностью). Людям с таким синдромом тяжело отпускать людей, в них развивается маниакальное желание контролировать каждый шаг своего предмета обожания, мазохизм.

Часто встречается «синдром Мюнхгаузена». Его проявление состоит в том, что человек убеждает всех и самого себя, что он болен, что ему нужна госпитализация, симулирует различные болезненные состояния, и даже может намеренно причинять себе вред. Люди поступают так с целью обратить на себя внимание, получить сострадание и вызвать эмпатию. Стать его причиной может насилие в детском возрасте, недостаток внимания, или, наоборот, гиперопека. Человек, которому в детстве не хватило заботы и ласки, хочет почувствовать себя нужным и важным.

На основе изученной информации я провела наблюдения за своим кругом общения и заметила, что у многих в поведении проявляются различные признаки детских психотравм и это, так или иначе, сказывается на общении с окружающими людьми. Исходя из проведённого исследования можно сделать вывод, что влияние психологических травм на поведение детей и взрослых очень велико. Каждый родитель должен следить за тем, какой пример он подаёт ребёнку, и анализировать свои слова и поступки, которые могут оказать воздействие на формирование личности индивида. Последствиями психотравм могут быть трудности в общении и взаимодействии с людьми, проблемы с принятием себя, тревожность, апатия, и даже преступное поведение. Для работы с ней лучше всего обратиться к квалифицированному психологу или психотерапевту — они смогут оказать адекватную помощь при детских психологических травмах и найти пути их преодоления, вне зависимости от времени, которое прошло после травмирующего события.

Список использованных источников:

1. Бёрн Э. Психоанализ и психотерапия для несведущих. – М.: ООО «Издательство АСТ», 2002. – 416 с.
2. Фрейд З. Введение в психоанализ. – М.: ООО «Издательство АСТ», 2019. – 760 с.
3. Левин П. Исцеление от травмы. М.: Весь, 2011. – 128 с.

PSYCHOLOGICAL ANALYSIS OF THE SIGNIFICANCE OF CHILDHOOD EXPERIENCE IN ADULT LIFE

Katkovskaya E.A., student gr. 150502

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Miskevich V.I. – Candidate of Philos. sciences, associate professor

Annotation. The theses consider the problem of the influence of childhood psychotrauma on the formation of an adult's personality. The process of raising a child and his growing up, the formation of a personality is touched upon. The most common psychological features, the appearance of which is closely related to childhood, are analyzed.

Keywords. Psychic trauma, family, parents, child, syndrome, negative past, experiences, trauma, gaslighting.

РОЛЬ МАССМЕДИА В ФОРМИРОВАНИИ МИРОВОЗЗРЕНИЯ СОВРЕМЕННОГО ЧЕЛОВЕКА

Альхимович Н.Г., студентка гр. 150502

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Миськевич В.И. – кандидат философских наук, доцент

Аннотация. В тезисах рассматриваются актуальные проблемы формирования мировоззрения человека в современном мире. Подчеркивается, что сегодня мировоззрение человека испытывает влияние технологических новшеств, в частности средств массовой информации. В тезисах исследуются положительные и отрицательные стороны воздействия СМИ как на общественное сознание, так и на мировоззрение личности.

Ключевые слова: мировоззрение, средства массовой информации, личность.

Одна из особенностей стадии постиндустриального общества, на которой человечество в данный момент находится, является неотъемлемая роль средств массовой информации в формировании картины современного мира. Массмедиа не только транслируют происходящие в странах процессы, но и оказывают влияние на восприятие окружающих вещей, принимают на себя создание и распространение норм поведения и социальных ценностей.

На заре цивилизации человек владел четырьмя видами коммуникации: устной речью, музыкой, живописью и письменностью. В современном же мире видов массовой информации стало в разы больше, что неизбежно вывело потребности человека в информации на качественно новый уровень. Если столетия назад документированию подлежали самые основные и наиболее ценные факты, а новости доходили довольно долго, то сейчас люди постоянно получают огромные объемы данных, чей поток не прекращается ни на секунду. Причем информация приходит из различных сфер одновременно, что подразумевает особенный характер подачи и эмоциональное воздействие.

Важными характеристиками СМИ, наиболее полно сформировавшимися только в XXI веке, стали оперативность и доступность. Такие функции массовой информации, как образование и просвещение, решение повседневных задач, коммуникация и самореализация, в той или иной степени присутствовали в жизни человека и ранее. Несмотря на то, что в современном мире существует большое количество разнообразных источников, не всегда достоверных и объективных, легкий и быстрый доступ к ним превратил СМИ в мощное средство формирования общественного мнения и мировоззрения личности.

Во-первых, СМИ могут видоизменять представления человека о некоторых вещах. Например, постоянный просмотр криминальных хроник, изучение материалов о случаях насилия может вызвать равнодушие вследствие того, что человек привыкает к ежедневному потоку ужасающих новостей и в некотором роде вырабатывает к ним иммунитет. Этой точки зрения придерживается Ричард Харрис, применяя к вышеописанному результату просмотра сцен насилия термин «десенсбилизация». Возможен и обратный результат, когда во всех источниках, интересующих конкретного человека, присутствуют однотипные данные, что может вызвать несвойственную ранее негативную реакцию. То есть опасность в данном случае заключается в том, что отношение человека к конкретной вещи формируется не только на основании подробного ее анализа и оценки ее места в реальности, но и на частоте ее встречаемости в жизни человека, что ярко свидетельствует об искажении восприятия.

Во-вторых, следует отметить необходимость изучения рассматриваемого вопроса о влиянии массовой информации на мировоззрение современного человека, потому как наблюдается парадоксальная различность этого самого влияния на разных людей. С одной стороны, можно сказать, что человек практически никогда не покидает информационное пространство (в особенности с появлением Интернета), что может привести к расшатыванию психического равновесия личности вследствие нахождения в постоянном напряжении и, как следствие, состоянии тревоги. Согласно данным опроса TunnelBear более 39% респондентов не чувствуют безопасность, пользуясь Интернетом. Худший сценарий развития данной проблемы вкупе с иными проявлениями тотального контроля описывается в антиутопии Джорджа Оруэлла «1984». Население тоталитарного государства Океания находится под постоянным надзором правящей верхушки, а неусыпный контроль осуществляется посредством телеэкранов, которые почти не замолкают, что негативно сказывается на душевном состоянии и восприятии главного героя романа.

С другой же стороны, массмедиа способны постоянно утолять «информационный голод» человека. Особенно этому способствует доступность обратных каналов связи и многообразие способов комментирования последних новостей, газетных статей или материалов Интернет-ресурсов. Иначе говоря, СМИ стали более масштабной площадкой для выражения собственного мнения в желаемой, но допустимой форме. Это в свою очередь способствует тому, что человек чувствует себя значимой частью общества, способной менять текущий ход событий, ведь демонстрация и аффилиация, по классификации американского психолога Генри Мюррея, является одной из основных социальных потребностей человека.

В-третьих, учитывая тот факт, что несколько десятилетий назад информация в широкий доступ, как правило, поступала из официальных источников, а сейчас любые сведения может распространять, по сути, кто угодно при наличии устройства с доступом в Интернет, одновременно может существовать несколько источников с диаметрально противоположными версиями произошедшего. Как следствие, человек вынужден подходить более тщательно к выбору ресурсов, заслуживающих доверия, и подвергать всю получаемую информацию сомнению и последующей проверке, ведь в век информационных технологий подделать кадры с места событий, извратить комментарии их участников или выдать одно за другое стало значительно проще. Так, к примеру, 25 марта 2022 года в одно и то же время в сети Интернет публиковались новости, информирующие о смерти российского политического деятеля Владимира Жириновского, и о его нахождении в тяжелом состоянии. В результате люди в настоящее время вынуждены более избирательно и тщательно формировать свою позицию относительно тех или иных вещей.

В-четвертых, обратной стороной многообразия СМИ в жизни общества является то, что человек, сталкиваясь с постоянным потоком информации, начинает искать пути выхода из информационного поля. В итоге это может привести к тому, что человек замыкается в себе и живет в отчуждении от остального мира, предпочитая ни коим образом не выражать и не разделять свою позицию касаясь происходящих в мире процессов. Так, например, в опросах, проводимых в телеграмм-каналах медиаресурсов с целью узнать наиболее интересные для аудитории темы, актуальные новости являются последними по популярности. За этот вариант голосует в среднем 16% аудитории канала.

Всё вышеперечисленное показывает негативное влияние массмедиа на формирование личности человека и системы его взглядов. Теперь следует добавить к уже упомянутым выше аспектам еще несколько примеров положительной роли, которую СМИ играют в жизни современного общества. Стоит начать с того, что СМИ и Интернет сочетают в себе функцию посредников в общении между людьми, а также поставщиков свежих данных и знаний из любой области. Благодаря массмедиа человек сейчас имеет как никогда много возможностей для саморазвития, творческого поиска и расширения собственного кругозора. Это способствует приобретению необходимого жизненного опыта и умения взаимодействовать с другими членами социума. Так, немецкий философ Эрих Фромм считал, что наше сознание – наш проводник в бескрайнем мире. Оно способно правильно отражать реальность. Надо только неукоснительно и с любовью следовать по пути познания. В наше время этому в значительной мере способствуют массмедиа.

Кроме того, памятуя о том, что массмедиа нередко освещают не самые позитивные события из жизни общества, стоит отметить, что, основываясь на психоаналитической теории австрийского психолога Зигмунда Фрейда, накапливающаяся «агрессивная» энергия (проявление разрушительного начала – Танатоса) должна разряжаться, к примеру, в наблюдении за жестокими действиями. Нельзя отрицать, что эта функция СМИ необходима, ведь нераскрытые эмоции могут видоизменять систему оценок и общее отношение человека к действительности и к самому себе.

Таким образом, как можно судить из вышеизложенного, роль массмедиа в формировании мировоззрения современного человека довольно неоднозначна, поскольку, во-первых, в целом возможен как положительный, так и отрицательный эффект, а, во-вторых, любой из представленных аспектов может по-разному повлиять на ту или иную личность. Это в свою очередь зависит от темперамента и типа личности человека, условий существования, окружения и еще многих факторов. Именно по этой причине для современной философии вопрос роли СМИ в жизни человека должен оставаться актуальным и подвергаться всестороннему изучению с целью поиска наиболее эффективных и действенных подходов. Более того, несмотря на то что сами по себе средства массовой информации существуют уже много сотен лет, но сейчас они находятся на небывалой ранее точке развития, к разным формам и проявлениям которой человечество еще не может быть готовым в силу скорости происходящих изменений. В заключение хочется привести отрывок из высказывания английского писателя, футуролога и публициста Артура Кларка:

«Мы становимся расой созерцателей, а не созидателей. Но, с другой стороны, смотреть телевизор – все равно что дышать: без него не обойтись. Человек может дольше прожить без еды, чем без информации. Загвоздка – в образовательной системе: людей надо обучать разборчивости. Только самодисциплина сможет противостоять тем волшебным силам, которые еще только будут вызваны нами к жизни. Иначе все станем морскими губками» [2].

Список использованных источников:

1. Шалаева А.В. Медиафилософия V. Способы анализа медиареальности – Санкт-Петербург: Издательство Санкт-Петербургского философского сообщества, 2010 – 279 с.
2. 12 изумительно точных мыслей Артура Кларка о настоящем и будущем человечества [Электронный ресурс]. – Режим доступа: <http://izbrannoe.com/news/mysli/12-izumitelno-tochnykh-mysley-artura-klarka>. – Дата доступа: 22.03.2022.
3. Влияние массовой коммуникации на формирование мировоззрения личности [Электронный ресурс]. – Режим доступа: <https://cyberleninka.ru/article/n/vliyanie-massovoy-kommunikatsii-na-formirovanie-mirovozzreniya-lichnosti/viewer>. – Дата доступа: 27.03.2022.

THE ROLE OF MASS MEDIA IN SHAPING THE WORLDVIEW OF A MODERN MAN

Alkhimovich N.G., student gr. 150502

*Belarusian State University of Informatics and Radioelectronics
Minsk, Republic of Belarus*

Miskevich V.I. – PhD, Associate Professor

Annotation. The actual problems of the formation of a person's worldview in the modern world are considered in the theses. It is emphasized, that today a person's worldview is influenced by technological innovations, in particular mass media. The theses consider the positive and negative aspects of the impact of the media on both the public consciousness, as a whole, and the worldview of the individual.

Keywords: worldview, mass media, personality.

СЕКЦИЯ «ФИЗИКА»

ОПТИМИЗАЦИЯ РАБОТЫ УЛЬТРАЗВУКОВОЙ ОЧИСТИТЕЛЬНОЙ ВАННЫ

Родевич Р.А., Шебеко В.П.

*Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь*

Научный руководитель: Савилова Ю.И.—доцент кафедры физики

Аннотация. В ходе работы были проведены опыты с использованием кавитометра, датчика и анализатора спектра. Были установлены зависимости активности кавитации от уровня жидкости, температуры и содержания газа.

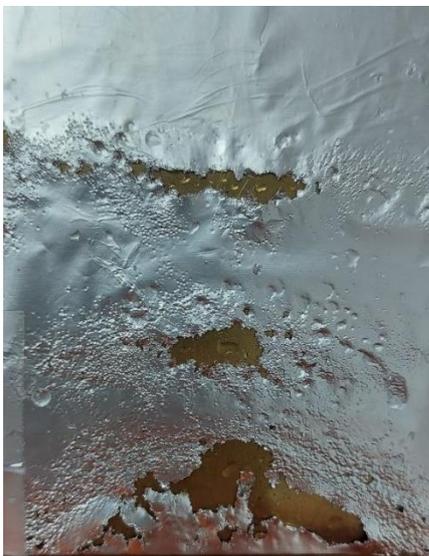
Ключевые слова: кавитация, активность, ультразвуковые волны, ультразвуковая ванна.

Введение.

Кавитация – процесс образования пузырьков в жидкостях и их последующее схлопывание, при котором высвобождается большое количество энергии и другие физические процессы. Сам процесс сопровождается гидроударами и характерным шумом. Кавитация возникает в результате местного понижения давления в жидкости, которое может происходить либо при увеличении скорости, (гидродинамическая кавитация), либо при прохождении акустической волны большой интенсивности во время полупериода разрежения (акустическая кавитация).

Принцип кавитации применяется в многих областях медицины таких как нейрохирургия, офтальмология, онкология, отоларингология, кардиология и урология^[1]. Также в промышленных целях применяется очистка при помощи ультразвуковой ванны. При очистке таким способом требуется правильно поместить деталь в ванну. При неправильном размещении детали есть вероятность неравномерной очистки или разрушения детали(рисунок 1).

Рисунок 1



Основная часть.

Для регистрации кавитации используются специализированное оборудование. К подобному можно отнести кавитометры, специализированные датчики и анализаторы спектра.

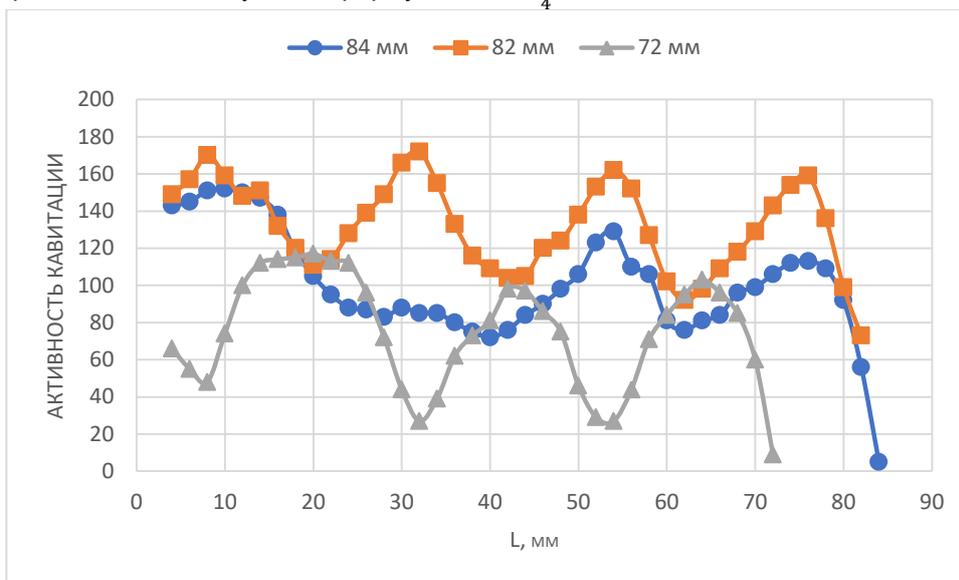
Принцип работы ультразвуковой ванны следующий: пьезоэлектрические излучатели преобразует электрические колебания, вырабатываемые транзисторным генератором, в упругие механические колебания соответствующей частоты. Через плоское дно ванны колебания передаются в рабочую среду. Воздействие на физико-химические процессы в жидкостях осуществляется в основном за счет вторичных акустических эффектов – кавитации и акустических течений.

Ванна состоит из емкости в которую ко дну монтируется держатель преобразователя. В нем предварительно установлен преобразователь в этом случае происходит непосредственный контакт преобразователя с жидкостью.

На распределение кавитации влияют несколько основных факторов: уровень жидкости, температура и газосодержание.

Уровень жидкости:

На рисунке 2 представлены зависимости активности кавитации от расстояния L между датчиком и излучателем для разных уровней жидкости h в ванне. Из представленных данных видно, что при изменении уровня жидкости существенно меняется активность кавитации и характер её распределения в объёме жидкости. Это обусловлено скорее всего изменением условий образований стоячей компоненты звукового поля. Также меняется звук и невооружённым взглядом видно изменение на поверхности воды. При любом уровне можно наблюдать эффект стоячей волны. При расстоянии 82 мм наблюдается самая большая активность кавитации. Это обусловлено тем что для данного h условия образования стоячей волны наиболее близки к идеальным, определяемым следующей формулой $h = \lambda \frac{\lambda}{4}$

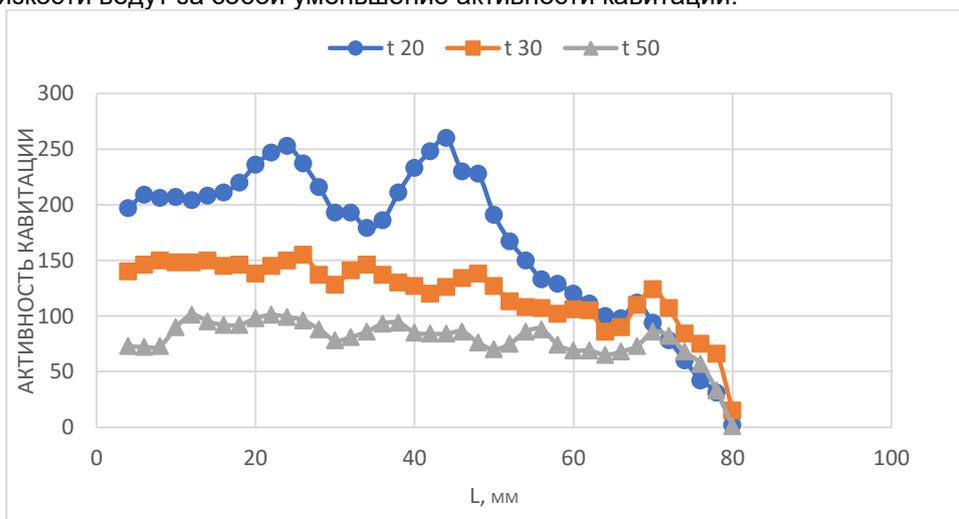


Уровень жидкости в ванне, мм: 1(●)—84 ; 2(■)—82; 3(▲)—72

Рисунок 2 – Распределение активности кавитации вдоль оси излучателя

Температура:

При изменении температуры повышается плотность вязкость жидкости. Увеличение плотности и вязкости ведут за собой уменьшение активности кавитации.



Температура жидкости в ванне, С°: 1(●)—20 ; 2(■)—30; 3(▲)—50

Рисунок 3—изменение активность кавитации в зависимости от температуры

Газосодержание:

Для исследования зависимости содержания газа в жидкости от распределения мы использовали минеральную воду.

На графике представлен изменение активности кавитации в процессе дегазации минеральной воды. В самом начале опыта наблюдается выделение большого количества пузырьков газа (рисунок 2). Примерно на 272 и 350 секундах происходят изменения.

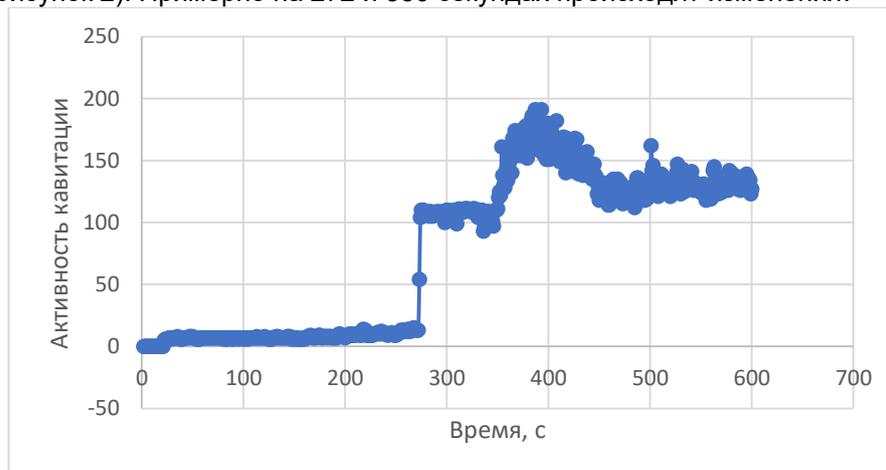


Рисунок 4 –изменение активности кавитации в зависимости от содержания газа

Заключение.

Выполнено исследование влияния температуры жидкости, ее уровня в реакторном стакане и дегазации на активность кавитации A в неоднородном ультразвуковом поле.

Показано, что при уровнях жидкости, наиболее близко соответствующих идеальным условиям формирования стоячей волны, распределение активности кавитации вдоль оси излучателя представляет собой чередующиеся максимумы и минимумы, расстояние между которыми примерно равно четверти длины волны ультразвука на данной частоте

При увеличении температуры жидкости активность кавитации уменьшается, меняется также характер ее распределение в объеме жидкости. При повышении температуры максимумы и минимумы проявляются все слабее, т.е. распределение становится более однородным.

В экспериментах с водой, пересыщенной углекислым газом, показано, что по мере дегазации, т.е. при уменьшении газосодержания активность кавитации повышается. Выделено две стадии дегазации: переход от одной стадии к другой сопровождается скачкообразным повышением активности кавитации.

Полученные результаты будут использоваться при оптимизации ультразвуковых технологических процессов в жидкости.

Список литературы

- 1.Л.Р. Гаериллов *ФОКУСИРОВАННЫЙ УЛЬТРАЗВУК ВЫСОКОЙ ИНТЕНСИВНОСТИ В МЕДИЦИНЕ.*
- 2.КАВИТАЦИЯ В ВОДНЫХ РАСТВОРАХ УГЛЕКИСЛОГО ГАЗА А.В. КОТУХОВ, Н.А. ЖАРКО, В.С. МИНЧУК, Н.В. ДЕЖКУНОВ.

OPTIMIZING THE PERFORMANCE OF THE ULTRASONIC CLEANING BATH

Rodevich R. A.,

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Savilova J.I. – assistant professor of the Department of Physics

Annotation. . In the course of the work, experiments were carried out using a cavitometer, a sensor and a spectrum analyzer. The dependences of cavitation activity on the liquid level, temperature and gas content were established.

Keywords. cavitation, activity, ultrasonic waves, ultrasonic bath.

СЕКЦИЯ «ИНФОРМАТИКА»

ИНТЕРНЕТ ВЕЩЕЙ. МЕТЕОСТАНЦИЯ НА БАЗЕ ARDUINO.

Богомаз К.С., Макарич Д.А., Юшкевич М.А.

*Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь*

Научный руководитель: Анисимов В.Я. – Доцент, кандидат физико-математических наук

Волорова Н.А. - Доцент, кандидат технических наук

Аннотация. Создан умный метеорологический модуль на базе Arduino. К модулю привязано мобильное приложение для мониторинга состояния датчиков модуля. Реализована безопасность передачи данных между устройствами. На базе выполненной работы написана документация по архитектуре IoT и ее описание.

Ключевые слова: метеостанция, Arduino, интернет вещей, умный дом.

Введение

Интернет вещей — это концепция сети передачи данных между устройствами. Внутри IoT люди могут общаться с «вещами», а «вещи» — общаться между собой.

Интернет вещей стал неотъемлемой частью повседневной жизни человека. Все мы смотрим прогноз погоды, который отчасти формируется за счет общения множества датчиков между собой. В повседневной жизни нас окружают умные приборы: умный чайник, розетка, увлажнитель воздуха, робот пылесос. которые избавляют нас от повседневной рутины.

Концепция умного дома - это концепция при которой все типовые процессы за человека будет выполнять техника; при этом значительную роль в такой системе занимает и безопасность. Это система видеонаблюдения, сигнализация, автоматическое оповещения правоохранительных органов в случае несанкционированного проникновения на частную территорию. Немаловажное значение имеет комфортное проживание, что обеспечивается удобным управлением освещением, параметрами температуры, влажности, уровня CO_2 .

Целью данной работы было создания метеорологического модуля, который помогает в наблюдении за параметрами в помещении, дает советы по улучшению микроклимата, выдаёт оценку качеству рабочей атмосферы.

Архитектура

IoT (рисунок 1) отличается от уровневой модели OSI, т.к. он содержит более сложный граф потоков. Это определяется тем, что в данном решении как правило присутствуют два физических местоположения - конечные (периферийные) устройства и backend сервера или облако. Некоторые уровни на данном графике являются необязательными и могут использоваться для решения определенных задач.

Физический уровень. Схема метеостанции.

На Рисунке 2 в упрощенной форме изображена схема метеостанции. Особенность модуля ... это подключение WI-FI модуля(esp-8266-01s). Он подключен через резисторы номинала 2кОм. Это необходимо потому что для работы ему необходимо небольшое постоянное напряжение, в противном случае его ресурс очень быстро исчерпается.

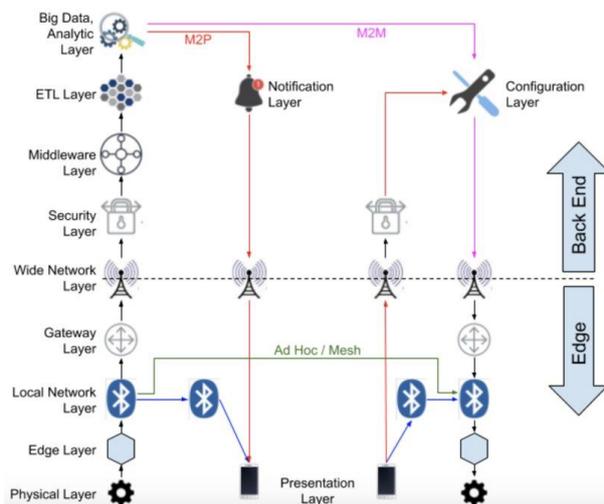


Рисунок 1. Архитектура IOT

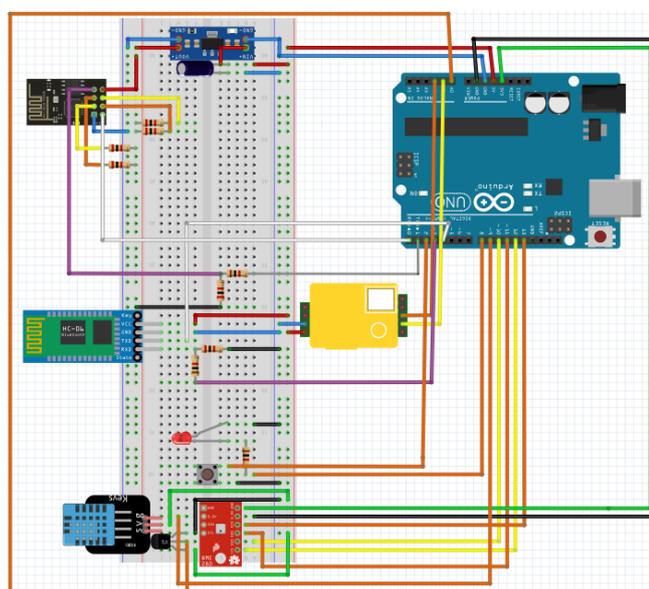


Рисунок 2. Схема метеостанции

Серийные входы/выходы (Рисунок 3 пины 2 и 7). подключены через подтягивающий вниз резистор номиналом 1кОм, с целью исключения помех при передаче данных с главной платы(Arduino) на esp-8266-01s. По такому же принципу подключен и bluetooth(HC-06) модуль.

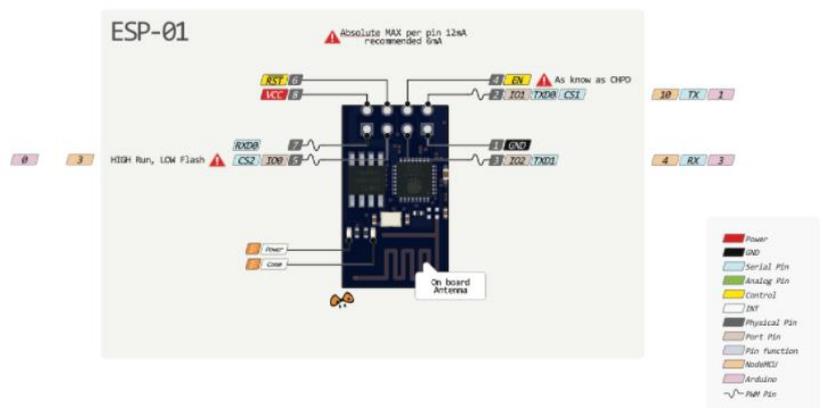


Рисунок 3. Спецификация платы ESP-8266-01S

Физический уровень. Считывание данных.

Считывание данных происходит раз в 30 мин или по нажатию на кнопку на устройстве. Процесс считывания занимает около 1 секунды, после чего мы получаем данные температуры, влажности, давления и углекислого газа. После чего мы отправляем данные в Serial 3 порт для передачи их на esp-8266-01s(WI-FI модуль) и последующей отправки на сервер.

На данном уровне возникает проблема перевода аналогового сигнала в понятные нам единицы измерения. Это преимущественно осуществляется с помощью библиотек, но для датчика LM35DS(датчик температуры) необходимо преобразование (1) и преобразование давления в мм.рт.с (2).

$$T_{lm35ds} = S_{lm35ds} \times 0.48828125, \quad (1)$$

$$P_{bme280} = S_{bme280} \times 0.007500637554192, \quad (2)$$

где T_{lm35ds} - значение температуры, °C.

S_{lm35ds} - значение аналогового сигнала.

P_{bme280} - значение давления, мм.рт.с.

S_{bme280} - значение давления, Па.

Уровень локальной сети. Общение с пользователем.

Общение с окружением происходит посредством HC-06(bluetooth). Для реализации требуется программно открыть новый порт для общения с Arduino UNO. Bluetooth защищен паролем для того, чтобы случайный пользователь не мог подключиться и испортить работу метеостанции. Через HC-06 посредством приложения могут быть переданы данные о точке доступа, к которой стоит подключиться метеостанции, различные команды, например команда на считывание данных или команда на перезагрузку.

Особенностью данного слоя является обмен данными в формате байт, что нужно учитывать при обработке команд.

Уровень шлюзов. Выход в глобальную сеть.

Общение происходит через встроенный в Arduino UNO Serial 3 com port. На wi-fi модуль могут приходить либо показания с датчиков, либо команда на обновление пароля

и имени точки доступа. Как только модуль установил соединение с точкой доступа он отвечает статусом успеха, который в свою очередь передаётся на bluetooth модуль и далее на приложения пользователя. Если на esp приходят данные с датчиков, модуль отправляет их на сервер.

Здесь решаются следующие задачи: проверка подключения к точке доступа, правильность введенных данных, обработка сбоев сети, валидация данных, безопасность. Проверка на подключение метеостанции к точке доступа происходит постоянно. Этот статус отправляется по bluetooth на устройство пользователя и пользователь может принять меры по сопряжению метеостанции с интернетом в случае сбоев сети. Также если метеостанции по каким-то причинам отключилась от точки доступа, например отсутствия энергопитания или программная ошибка, то она сама принимает 5 попыток подключения и в случае неудачи сигнализирует пользователя. Про проблему безопасности будет рассказано на уровне безопасности.

Уровень ETL. Запрос данных с сервера.

Данные с метеостанции обязаны приходиться в определенном формате в теле https запроса вместе со специальным токеном в заголовке определяющим уникальность метеостанции и защищенность данных (метео-токен), далее, на сервере идёт проверка, обработка и трансформация данных, также проверяется зарегистрирована ли такая метеостанция, валиден ли токен, валидация входящих параметров с датчиков и т.д.. После того как валидность данных проверена и они сгруппированы по объектам, данные записываются в соответствующие таблицы.

Уровень шлюзов. Запрос метео-данных с сервера.

Короткая продолжительность, лаконичность запроса к БД, а также предельно ясная структура достигаются путем группировки данных температуры, влажности, давления, Co2 по отдельным таблицам. В параметрах строки запроса одного пути указывается то для какой именно метеостанции и какие именно данные нужны (температура/давление/влажность/уровень Co2).

Уровень безопасности. Токен метеостанции.

При подключении метеостанции пользователем через мобильное приложение метеостанция получает от приложения специальный токен который будет храниться на ней для дальнейшей авторизации метеостанции на сервере, данный токен после занесения измениться не может.

Уровень безопасности. Комнаты

Комнаты представляют собой способ организации доступа разных людей к одной метеостанции. Комнаты бывают публичные и приватные, публичные комнаты можно найти в поиске по их имени, вступить в них как участник и получить доступ к данным с метеостанции. Публичные комнаты удобны в случаях, если имеется какая-то интересная зона для наблюдений и все, кто хочет, могут посмотреть эти данные, например показатели в каком-нибудь общественном помещении или в зоне метео исследований. Закрытые комнаты в свою очередь нельзя найти по их имени, а вступить в них можно лишь по приглашению от администратора, такие комнаты хорошо подходят для семейной метеостанции в доме или метеостанции в общем помещении для всех жильцов.

Клиент. Bluetooth соединение.

После авторизации пользователю необходимо выбрать устройство(метеостанцию), к которой он подключится с помощью bluetooth. После подключения к устройству мы можем обмениваться с ним данными. После завершения работы с bluetooth terminal вызывается метод disconnect мы отключаемся от устройства и отключаем bluetooth.

Клиент. Контент приложения.

На следующем этапе пользователь может просматривать данные температуры, влажности, давления и уровень CO₂. Далее данные группируются по дню, месяцу, неделе и году. На основе этих данных строятся графики для каждого датчика за определенный промежуток времени. Что касается данных с CO₂, то для пользователя отображается PieChart со стрелкой, показывающей уровень CO₂ в помещении. На данном этапе присутствуют три диапазона значений:

- 1) Зеленый, который варьируется от 400 до 800 млн - это норма;
- 2) Желтый - это средний уровень, который находится в пределах от 800 до 1300 млн;
- 3) Красный - слишком высокий уровень CO₂ со значениями от 1300 до 5000. В данном случае приходит уведомление. Об уведомлениях мы поговорим дальше;

Клиент. Уведомления

На стороне клиента имеется механизм уведомлений. Реализованы следующие уведомления как: высокое содержание CO₂, слишком высокая температура, ошибки с датчиков метеостанции. Приложение на основе полученных данных проводит анализ и в случае - высокой температуры, уровня CO₂. Пользователю приходит уведомление о том, что какой то показатель выше среднего и необходимую (возможную) инструкцию, что можно сделать для нормализации, например проветрите комнату слишком душно $t > 22$ или проветрите комнату слишком грязный воздух $ppm(co_2) > 1300$ и т.д.

Список использованных источников

1. *The Weather Machine: A Journey Inside the Forecast Книга, Эндрю Блум/ Книга. – М.: Изд. Ессо, 2019. – 136 с.*
2. *Интернет вещей с ESP8266, Эндрю Блум/ Книга. – М.: Изд. BHV, 2019. – 76 с.*
3. *Изучаем Arduino: инструменты и методы технического волшебства Книга, Джереми Блум/ учебно-методический комплекс. – М.: Изд. центр Петербург, 2015. – 431 с.*
4. *Оптимизация энергопотребления Arduino [Электронный ресурс]. – Режим доступа: <https://alexgyver.ru/gyverpower/>*
5. *Работа с датчиком BME [Электронный ресурс]. – Режим доступа: <https://www.arduino.cc/reference/en/libraries/adafruit-bme280-library/>*
6. *AWS Documentation [Электронный ресурс]. - Режим доступа: <https://docs.aws.amazon.com/>*
7. *Arduino Documentation [Электронный ресурс]. - Режим доступа: <https://docs.arduino.cc/>*

UDC 004.056.53

INTERNET OF THINGS. METEOSTATION BASED ON ARDUINO.

Bogomaz K.S., Makarich D.A., Yushkevich M.A.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus (style T-institution)

Anisimov V.Y. – docent, candidate of physical and mathematical sciences

Volorova N.A. - docent, candidate of technical sciences

Annotation. Smart meteorological module based on Arduino has been created. Mobile application is connected to the module to monitor the module's sensors. Implemented security of data transfer between applications. On the basis of the work performed, documentation on the IOT architecture and its description were written.

Keywords. meteostation, Arduino, internet of things, smart home

БЫСТРОЕ ПОСТРОЕНИЕ ГРАФОВОЙ БД WEB-САЙТА. ГРАФ ЗНАНИЙ КАК СРЕДСТВО ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ИНТЕРНЕТ ИСТОЧНИКА

Зорко П. А., Кулевич А. О.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Пилецкий И. И. – канд. физ.-мат. наук, доцент

Приводится описание архитектурных решений быстрого построения компонента тематического прототипа графической БД, графа знаний из открытых интернет-источников с целью глубокого анализа данных сайта, выявления скрытых зависимостей в некоторой научной области. Описываются принятые решения, демонстрируются, результаты работы компонента получение данных с web-сайта.

Одна из больших технических задач — это получение данных из конкретного сайта [1]. Но, еще более сложная проблема — это анализ данных размещенных на данном сайте. Что приводит к появлению ряда новых возможностей моделирования структуры сайта, одна из которых - возможность предложить простой способ представления свойств отношений в виде графовой модели. Что позволяет анализировать свойства сайта с помощью графовых моделей. Многие сайты формально поддерживает тройки RDF (Resource Description Framework), в которых тройка имеет вид «субъект — предикат — объект» и называется триплетом (тройкой). Множество RDF-утверждений образует ориентированный граф, в котором вершинами являются субъекты и объекты, а рёбра отображают отношения. С помощью графовых технологий можно преобразовать представление сайта в графовую БД со свойствами.

На сайте Open Food Facts [2] имеется общедоступный набор данных с 6,2 миллионами троек по пищевым продуктам, их ингредиентам, аллергенам, фактам питания и многому другому. Импортируя тройки RDF получается тематическая графовая БД.

Общее представление полученной БД продемонстрировано на рисунке 1.

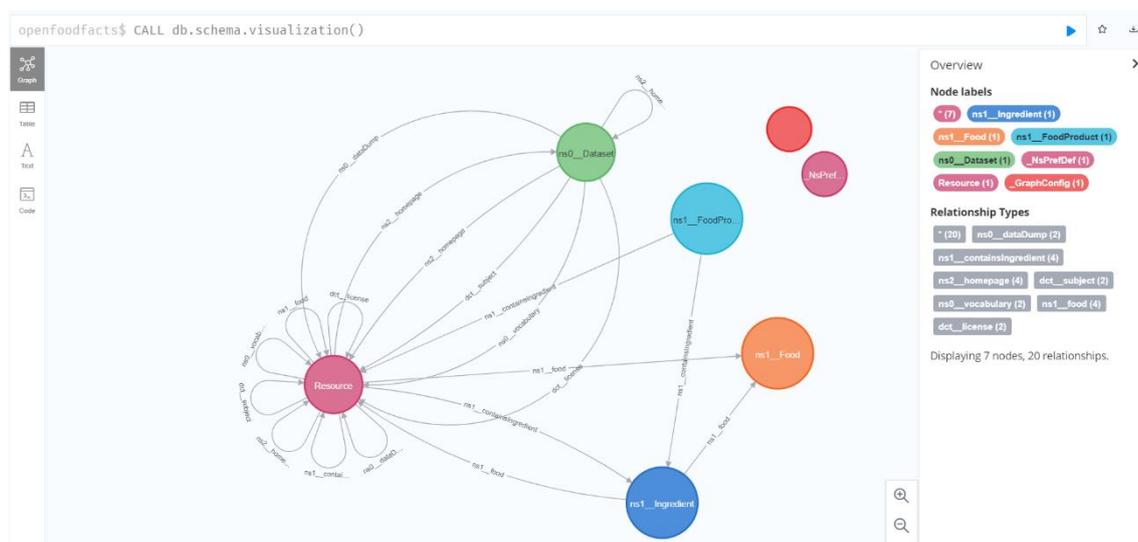


Рисунок 1 – Общее представление схемы графовой базы данных

В данной БД основными типами узлов являются:

- FoodProduct – продукты;
- Food – ингредиенты, которые могут содержаться в продуктах;
- Ingredient – промежуточные узлы, соединяющие узлы FoodProduct и Food, в которых указывается, сколько ингредиента содержится в продукте.

Основными типами связей являются:

- containsIngredient – соединяет продукт FoodProduct и промежуточный узел Ingredient;
- food – соединяет промежуточный узел Ingredient и ингредиент Food.

В полученной базе данных можно искать различную информацию о продуктах и составе их ингредиент. К примеру, с помощью запроса, представленного ниже, можно узнать, какой набор общих ингредиентов у двух продуктов. Результат запроса показан на рисунке 2.

```
MATCH (prod1:Resource { uri: 'http://world-fr.openfoodfacts.org/product/5053827196642/coco-pops-kellogg-s'})
MATCH (prod2:Resource { uri: 'http://world-fr.openfoodfacts.org/product/5010358227290/2-snack-pork-pies-spar'})
MATCH (prod1)-[:ns1__containsIngredient]->(x1)-[:ns1__food]->(sharedIngredient)<-[:ns1__food]-
(x2)<-[:ns1__containsIngredient]-(prod2)
RETURN prod1, prod2, x1, x2, sharedIngredient
```

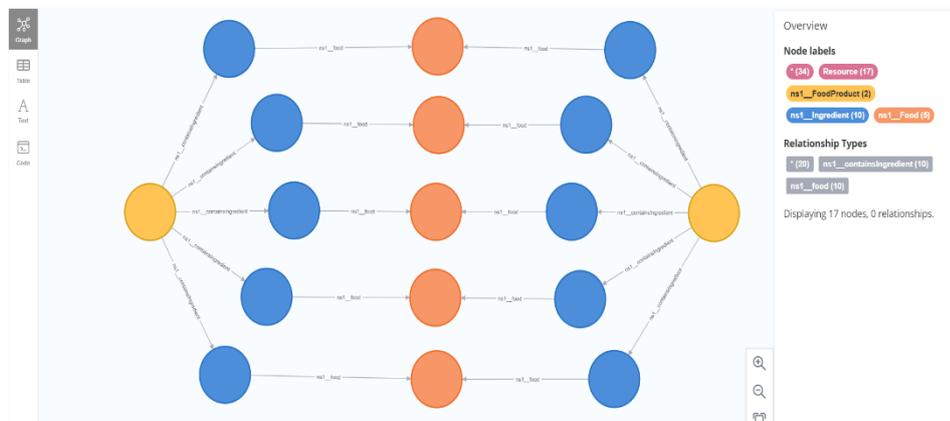


Рисунок 2 – Вывод общих ингредиентов у двух продуктов

Используя данный граф знаний, мы легко можем продемонстрировать 15 самых популярных ингредиентов, которые используются в продуктах. Для этого нам понадобится рассмотреть только узлы FoodProduct, Ingredient, Food и связи между ними [3].

Следующий запрос используется для подсчета использования каждого ингредиента и выводит топ 15 из них:

```
MATCH (:ns2__FoodProduct)-[:ns2__containsIngredient]->()-[:ns2__food]-(i:ns2__Food)
RETURN i.ns2__name as name, count(*) as Popularity
ORDER BY Popularity DESC
LIMIT 15
```

Сохраним полученные данные в файл формата csv, чтобы в дальнейшем их использовать.

Используя возможности языка Python, построим гистограмму популярности ингредиентов.

Результат представлен на рисунке 3.

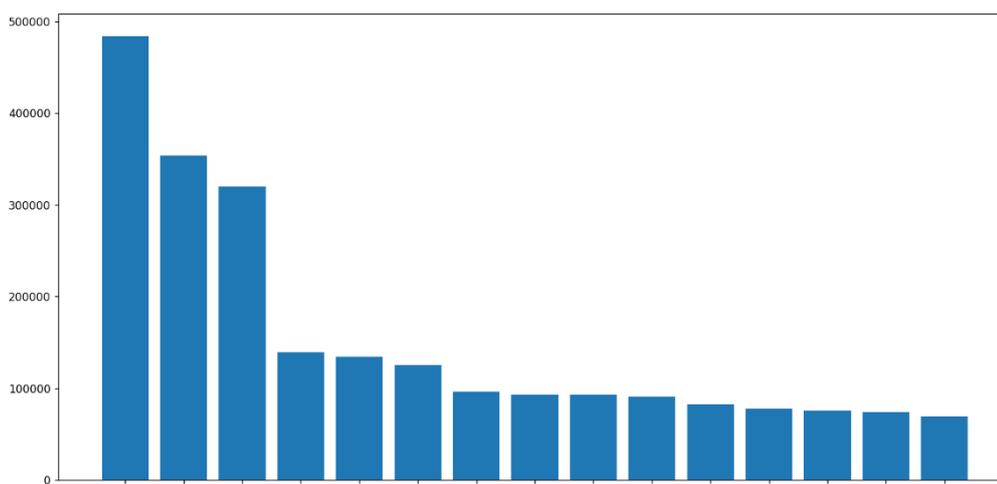


Рисунок 3 – Гистограмма популярности ингредиентов

Результатом научной работы является граф знаний продуктов питания Open Food Facts, к которому в дальнейшем можно применять различные алгоритмы машинного обучения. К примеру, разбить ингредиенты на «плохие» и «хорошие», а затем классифицировать продукты на полезные и нет.

Список использованных источников:

1. Ян Робинсон, Джим Вебер, Эмиль Эифрем. Графовые базы данных: новые возможности для работы со связанными данными / пер. с англ. Р.Н. Рагимова; науч. ред. А.Н. Киселев. – 2-е изд. – М.: ДМЛ Пресс, 2016. – 18 с.

2. Open Food Facts [Электронный ресурс] / Режим доступа: <https://world.openfoodfacts.org> Дата доступа: 25.03.22.
3. Neo4j [Электронный ресурс] / Режим доступа: <https://neo4j.com/developer/graph-data-science/graph-algorithms/>
Дата доступа: 10.02.22

ИДЕНТИФИКАЦИЯ ЦИФРОВЫХ УСТРОЙСТВ С ПОМОЩЬЮ БИСТАБИЛЬНЫХ ЭЛЕМЕНТОВ, РЕАЛИЗОВАННЫХ НА FPGA

Кайки М.

Белорусский государственный университет информатики и радиоэлектроники

г. Минск, Республика Беларусь

Иванюк А.А. - д.т.н., профессор кафедры информатики

Аннотация. Данная работа посвящена изучению внутрикристалльных и межкристалльных статических ячеек памяти, построенных на двух инверторах с обратной связью. Приведён пример расчёта расстояний по Хэммингу в троичной и пятеричной системах счисления для анализа различия нестабильных идентификаторов. Построены тепловые карты ячеек статической памяти с учётом нестабильных элементов, рассмотрены возможности применения подобных физически неклонированных функций в системах защиты цифровых схем и идентификации.

Ключевые слова. Физически неклонированная функция, кристалл, статическая память, FPGA.

Введение.

Согласно отчету Организации экономического сотрудничества и развития, за 2020 год во всем мире было продано контрафактных товаров на сумму около 2 триллионов долларов [1]. Данный факт является плохим признаком для потребителей и компаний, которые заказывают детали из разных источников по всему миру для создания своих продуктов, которые в большинстве случаев должны обладать высокой надёжностью и точностью выполнения работ. Изготовители контрафакта, как правило, используют сложные механизмы изменения, включающие множество стадий копирования и редактирования внутренней структуры продукта, в частности, микросхемы, что затрудняет проверку её происхождения и подлинности. Следовательно, компании могут получить имитационные детали, а их продукты станут менее надёжны и эффективны. В результате борьбы с контрафактной продукцией были предложены несколько технологий, среди них – усложнение структуры товара, применение водяных знаков, специальных рельефов, меток, системы маркировки и прослеживания товаров. Так, например, компания Intel для подтверждения авторских прав на этапе фотолитографического процесса наносила на кристаллы своих процессоров Intel 8086 специализированную метку. Одной из технологий защиты авторских прав при изготовлении микросхем является внедрение в структуру специализированных идентификаторов, единожды программируемых на одном из этапов производства, а затем считываемых при инициализации устройства или в процессе подтверждения соответствия при закупке комплектующих. Однако, данный метод защиты имеет большой недостаток – в случае утечки данных об алгоритме программирования идентификаторов злоумышленник может воспользоваться данным алгоритмом и внедрить его в контрафактную продукцию. Так, например, в январе 2022 года компания-производитель графических адаптеров Nvidia стала жертвой кибератаки, ответственность за которую взяла на себя хакерская группировка LAPSUS\$, которая похитила с серверов около 1 Тбайт конфиденциальной информации, в частности, были получены ключи идентификации видеоадаптеров в системе, а также методы обхода ограничений для процессов получения криптовалюты. Впоследствии хакерская группировка обнародовала сертификаты для подписи кода, и злоумышленники уже начали использовать их для подписи вредоносных программ. Украденные сертификаты использовались для подписи, бэкдоров и троянов удаленного доступа. Подобные случаи указывают на необходимость разработки уникальных, неповторимых идентификаторов, получение которых не должно зависеть от методов разработки цифровой системы.

Защита цифровых устройств:

Как указывается в [2], одним из методов схемной обфускации как инструмента защиты цифровых устройств является внедрение генераторов констант (ГК). ГК — разновидность непрозрачных предикатов. ГК представляют собой схемы, генерирующие одно логическое значение постоянно. Сложность реализации ГК заключается в создании нераспознаваемых и неминимизируемых средством синтеза схем. Вариации технологических процессов изготовления интегральных схем вносят в их физическую структуру случайные, непредсказуемые изменения, делающие каждый экземпляр цифрового устройства уникальным, неповторимым и невозпроизводимым [3]. Использование физически неклонированных функций позволяет решить данный класс задач с необходимыми условиями уникальности. Физически неклонированная функция (англ. Physical Unclonable Function (PUF)) – это функция, воплощенная в физической

структуре, которую просто оценить, но трудно охарактеризовать, смоделировать или воспроизвести [4].

Синтез схемы ФНФ основанной на ячейках статической памяти в FPGA:

При реализации идентификаторов мы использовали ФНФ на основе поведения стандартной памяти SRAM, доступной в любом цифровом чипе. Для получения ячеек статической памяти на базе ПЛИС использовалось описание двух последовательно соединённых инверторов с обратной связью, которые в процессе синтеза преобразовываются в две ячейки LUT-1 (Рисунок 1). Каждая ячейка SRAM имеет свое собственное предпочтительное состояние при подаче питания на логический блок, т.е. при конфигурации ПЛИС. Эта случайность выражается в начальных значениях «неинициализированной» памяти SRAM. Следовательно, ответ SRAM дает уникальный и случайный набор нулей и единиц. Этот шаблон подобен отпечатку пальца, поскольку он уникален для конкретной SRAM и, следовательно, для конкретного чипа. Проводя экспериментальную часть работы в совместной с БГУИР учебной лаборатории компании SK hynix memory solutions Eastern Europe, авторы упаковывали такие идентификаторы в массивы 32-ух разрядных регистров численностью в 512 слов на 4-ех различных кристаллах отладочной платы Nexys-4 компании Digilent. Извлечение значений из ячеек производилось при помощи тридцати двух 512-ти разрядных мультиплексоров и 32-ух разрядного софт-процессора MicroBlaze в виде IP-ядра компании Xilinx. При проведении эксперимента, каждая из плат конфигурировалась, отправляла на хост-машину последовательность из 512 слов, затем процесс повторялся 99 раз. Стоит отметить, что в процессе конфигурации ПЛИС, происходит её сброс, соответственно ячейки статической памяти возвращаются к исходному состоянию. Для обеспечения независимости измерений и недопущения перегрева ячеек, каждому процессу конфигурации предшествовала пауза длительностью в 20 секунд, что позволило нивелировать влияние переходных процессов и повышенный температуры в кристалле на результаты тестирования.

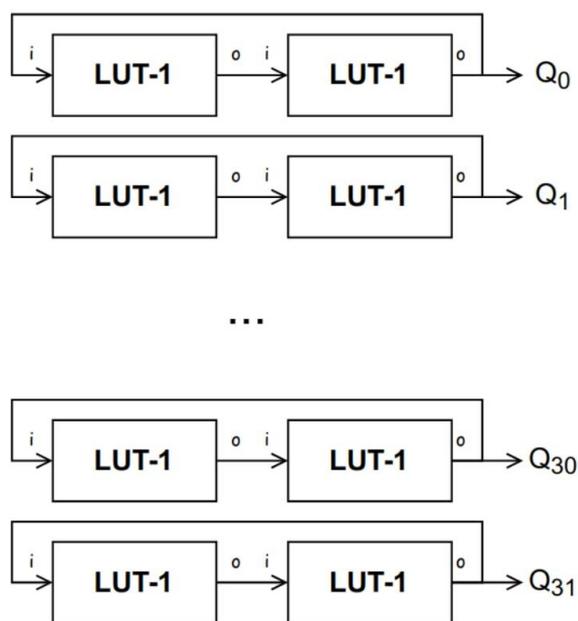


Рисунок 1. Схематическое исполнение ячейки статической памяти на двух инверторах с обратной связью с использованием элементной базы ПЛИС 7 серии компании Xilinx

Анализ основных характеристик, полученных ФНФ:

Проведенные исследования показывают, что каждая ячейка статической памяти при инициализации может принимать либо значение логического «0», либо логической «1», однако, как показывает мировая практика [5], часть ячеек не имеет чётких значений при инициализации между сбросами и может вести себя нестабильно, что приводит к невозможности использования в ГК и, как в следствие в системах идентификации. Для обеспечения надёжности и уникальности идентификаторов необходимо исключить из реализации нестабильные ячейки, для этого на этапе производства статическая память проходит проверку на пригодность к использованию в качестве идентификатора, затем полученная информация используется для программирования либо конфигурирования систем коррекции ФНФ [5]. Используя полученные данные в результате эксперимента, мы определили нестабильные ячейки памяти, рассчитали их распределения относительно логической единицы и нанесли полученные данные на тепловые карты для каждого из кристаллов. На рисунке 2 изображена одна полная тепловая карта и четыре фрагмента карт,

взятых из одного расположения, различных кристаллов. На рисунке крестиками обозначены нестабильные ячейки, черными – стабильные нули, белыми – стабильные единицы.

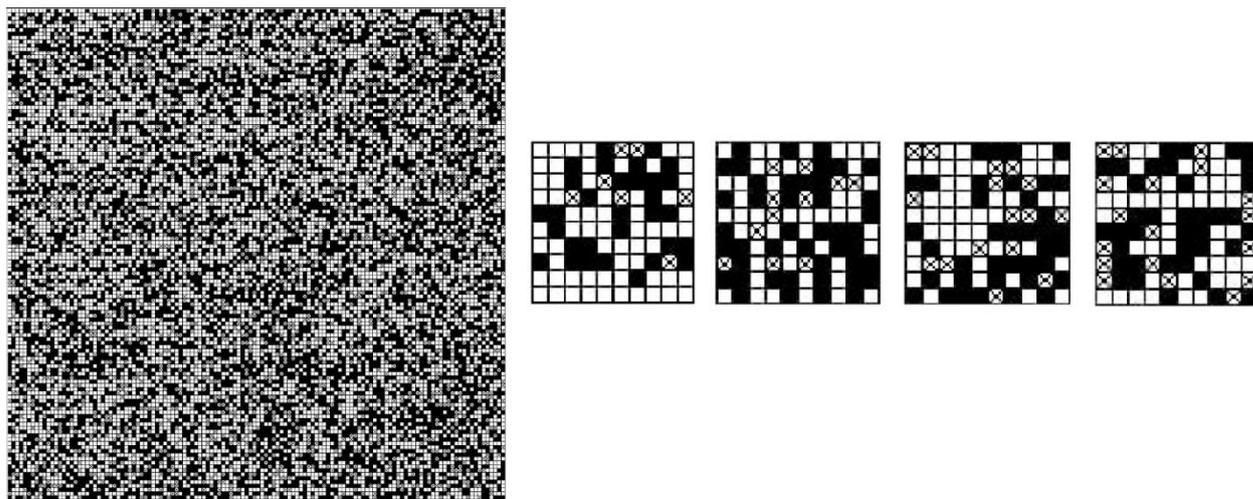


Рисунок 2. Тепловая карта сборок статической памяти на различных кристаллах.

Проанализировав полученные данные, мы пришли к выводу, что нестабильные ячейки более склонны к установке в логическую единицу, чем в ноль, а их количество относительно общей массы ячеек не превышает 15%, при этом, стоит отметить, что стабильными ячейками мы считали только те, которые на протяжении всех 100 опытов не изменяли свой логический уровень.

Метрики различия и уникальности идентификаторов:

При проведении анализа метрик различия нами были применены три различные функции расстояний по Хэммингу, одна для троичного и две для пятеричного алфавитов, так, каждый символ в алфавите соответствует диапазону вероятностей выпадения единицы в каждой из ячеек памяти. В троичном алфавите (0, X, 1) за единицу либо ноль принимались лишь те ячейки, которые не изменяли свои значения во всех экспериментах, остальные принимали значение X в пятеричном алфавите, каждый из символов соответствовал 20% приращению вероятности появления единичного символа (Таблица 2). Веса второй целевой функции расстояний по Хэммингу в случае 5-ричного алфавита были подобраны исходя из наблюдений, что среди ячеек имеют место быть элементы, стремящиеся к одному из двоичных уровней, но не принимающие однозначного значения на протяжении всех тестов.

Таблица 2 – Коэффициенты целевой функции расстояния по Хэммингу для пятизначного алфавита.

	0	L	X	H	1
0	0	0,25	0,5	0,75	1
L	0,25	0	0,25	0,5	0,75
X	0,5	0,25	0	0,25	0,5
H	0,75	0,5	0,25	0	0,25
1	1	0,75	0,5	0,25	0

Приведём пример расчёта удельного расстояния по Хэммингу для пятизначного алфавита:

Слово 1:	0	0	X	1	0	L	1	H
Слово 2:	L	0	1	X	0	1	0	L
Частичное расстояние:	0,25	0	0,5	0,5	0	0,75	1	0,5

Общее расстояние: $0,25 + 0 + 0,5 + 0,5 + 0 + 0,75 + 1 + 0,5 = 3,5/8 = 0,4375$

Исходя из полученных расстояний по Хэммингу (Таблица 3), можно сделать вывод: среднее удельное расстояние стремится к значению в 0,49, при этом ни в одной из тепловых карт не было получено абсолютно одинаковых или противоположных друг другу слов.

Таблица 3 – Расстояния по Хэммингу.

	Плата 1	Плата 2	Плата 3	Плата 4	Среднее
Расстояние Хэмминга (0, X, 1) (0,5 – 0,5)					
Среднее,%	0,4886	0,4901	0,4867	0,4926	0,4895
Максимальное,%	0,8593	0,8906	0,8437	0,8750	0,8671
Минимальное,%	0,0781	0,0468	0,0781	0,0312	0,0585
Расстояние Хэмминга (0, L, X, H, 1) (0,25 – 0,25 – 0,25 – 0,25)					
Среднее,%	0,4932	0,4941	0,4920	0,4955	0,4937
Максимальное,%	0,8671	0,8984	0,8281	0,8750	0,8671
Минимальное,%	0,1171	0,0546	0,1015	0,0390	0,0782
Расстояние Хэмминга (0, L, X, H, 1) (0,1 – 0,4 – 0,4 – 0,1)					
Среднее,%	0,4945	0,4948	0,4942	0,4963	0,4949
Максимальное,%	0,8625	0,9031	0,8375	0,8750	0,8695
Минимальное,%	0,1125	0,0591	0,0968	0,0343	0,0757

Так как генераторы на основе ФНФ действительно случайны, что позволяет использовать их в аппаратных реализациях криптографических алгоритмов, например, в качестве генераторов секретных ключей [6], для использования ФНФ SRAM необходимо убедиться в характеристиках уникальности между разными идентификаторами как на одном кристалле, так и между несколькими. Для определения таких параметров авторами использовалась нормированная в диапазоне [0;1] метрика единообразия, которая вычислялась для каждого слова, при этом, была применена двоичная система счисления, где нулю соответствовали вероятности выпадения единицы от 0 до 0,5, а единице – от 0,51 до 1 (Формула 1). При этом, стоит отметить что при вычислении характеристики уникальности идентификаторов, имел место быть случай, с нулевой уникальностью. Такой случай возможен только тогда, когда идентификатор состоит полностью из единиц либо нулей. В результате проведения эксперимента был обнаружен всего один такой идентификатор из 2048 на 4-ех различных кристаллах. На рисунке 3 приведена гистограмма полученных значений уникальности идентификаторов среди 4-ех различных кристаллах.

$$U = 1 - 2 \times \left| \frac{WH(V)}{N} - 0.5 \right|, \quad (1)$$

где V – слово разрядности N .

Таблица 4 – Полученные значения внутрикристалльного единообразия идентификаторов.

	Плата 1	Плата 2	Плата 3	Плата 4	Среднее
Среднее	0,8244	0,8133	0,8211	0,8248	0,8209
Максимальное	1	1	1	1	1
Минимальное	0,1875	0,1250	0,1250	0	0,1093

Помимо подтверждения свойств уникальности и неповторимости идентификатора в одном кристалле необходимо также подтверждать их межкристалльное различие. Для проверки последнего, нами были получены метрики различия в виде расстояний по Хэммингу между 4-мя кристаллами ПЛИС, при этом применялось вычисление расстояний «каждый с каждым», что позволило проверить все возможные сочетания идентификаторов. Межкристалльное среднее удельное расстояние по Хэммингу для троичной системы счисления составило 0,491, максимальное – 0,875, минимальное – 0. Полученные расстояния указывают на то, что в среднем идентификаторы между кристаллами по-прежнему уникальны между собой, однако, имеют место

быть случаи, когда на двух различных кристаллах идентификаторы совпадают. При вычислениях для 4-ех различных кристаллов, мы получили всего 2 случая совпадения из 1572864, что составляет 0,000127% от общего количества сравнений.

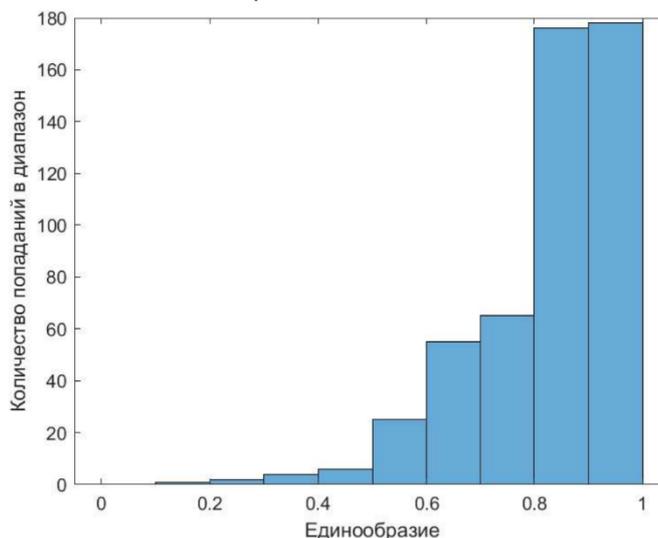


Рисунок 3. Гистограмма средней уникальности между 4-мя кристаллами.

Заключение:

Проведя исследование и анализ поведения ячеек статической памяти, реализованной на FPGA при инициализации, их метрик уникальности и различия мы получили данные о количестве стабильных и нестабильных ячеек, убедились в склонности к инициализации SRAM в уровень логической единицы, построили тепловые карты для 4-ех различных кристаллов ПЛИС, провели анализ межкристальной уникальности идентификаторов. Полученные данные свидетельствуют о том, что применение физически неклонированных функций на основе статической памяти в силу характеристик случайности и уникальности пригоден для использования в механизмах защиты цифровых схем от несанкционированного использования, копирования, модификации, а также в системах идентификации устройств. Кроме этого, были выявлены повторяющиеся межкристальные и нестабильные внутрикристальные идентификаторы, что указывает на необходимость разработки инструментов коррекции для достижения действительной уникальности и дальнейшего использования в системах идентификации.

Список использованных источников

1. Counterfeiting and piracy in 2021 – the global impact Danny Grajales Pérez-y-Soto International Chamber of Commerce 11 May 2021
2. Сергейчик, В. В. Генераторы констант как базовые примитивы схемной обфускации / В. В. Сергейчик // Компьютерные системы и сети: материалы 50-й научной конференции аспирантов, магистрантов и студентов (Минск, 24-28 марта 2014 г.). – Минск: БГУИР, 2014. – С. 78 - 79.
3. Иванюк А. А. Синтез симметричных путей физически неклонированной функции типа арбитр на FPGA / А. А. Иванюк // Информатика. - 2019. – Т.16, № 2. –С. 99-108.
4. Заливако С.С., Иванюк А.А. Схемная реализация комбинированной физически неклонированной функции для генерирования действительно случайных числовых последовательностей. Доклады БГУИР. 2013; (7):37-43.
5. Farha, Fadi & Ning, Huansheng & Liu, Hong & Yang, Laurence & Chen, Liming. Physical Unclonable Functions Based Secret Keys Scheme for Securing Big Data Infrastructure Communication. Information Sciences (2019).
6. Иванюк, А. А. Физическая криптография и защита цифровых устройств / А. А. Иванюк, С. С. Заливако // Доклады БГУИР. – 2019. – №2 (120). – С. 50–58.

UDC 004.056.53

IDENTIFICATION OF DIGITAL DEVICES USING BISTABLE ELEMENTS IMPLEMENTED ON FPGA

Kaiky M.

Belarusian State University of Informatics and Radioelectronics¹, Minsk, Republic of Belarus

Ivaniuk A.A. – Doctor of Sciences in Technology

Annotation. This work is devoted to the study of on-chip and inter-chip static memory cells built on two feedback inverters. An example of calculating Hamming distances in ternary and quinary number systems is given to analyze the difference between unstable identifiers. Thermal maps of static memory cells are constructed considering unstable elements, and the possibilities of using such physically non-cloneable functions in digital circuit protection and identification systems are considered.

Keywords. Physically non-cloneable function, chip, static memory, FPGA.

ИССЛЕДОВАНИЕ ПРИМЕНИМОСТИ ПАРАУНИТАРНЫХ БАНКОВ ФИЛЬТРОВ В НЕЙРОННЫХ СЕТЯХ

Кузьма В.В., Леченко А.В

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

В работе представлены результаты применимости параунитарных банков фильтров для уменьшения вычислительной сложности сверточных нейронных сетей в задаче сегментации изображения. В данной работе рассматривалась нейронная сеть U-Net, которая была обучена распознавать пешеходные переходы на 3083 размеченных дорожных изображениях.

Каждый год на дорогах гибнет около 300000 пешеходов, что составляет до 26% всех смертей в дорожно-транспортных происшествиях [1]. Это делает обнаружение пешеходных переходов важной задачей для обеспечения безопасности автомобильных средств. Для решения данной задачи можно использовать сверточные нейронные сети [2]. Они используются для сегментирования изображений, в том числе для выделения дорожных объектов, таких как пешеходные переходы [3]. В данной ситуации наиболее важным является время распознавания объекта. Одним из способов уменьшения времени распознавания является понижение размерности распознаваемого изображения методами пространственной фильтрации сигналов.

В данной работе представлено решение задачи сегментации пешеходных переходов на изображениях с использованием параунитарного банка фильтров на основе алгебры кватернионов [4], далее Q-ПУБФ и нейронной сети U-net [5]. Q-ПУБФ используется для понижения размерности входного изображения нейронной сети с целью понижения вычислительной сложности сегментации изображений.

Банк фильтров – цифровая система, состоящая из секций анализа и синтеза. Входной сигнал разбивается при помощи фильтров блока анализа на субполосные составляющие. На вход подается изображение размером $N \times N$. Разделение на субполосы позволяет наиболее декоррелировать входные данные. При этом, Q-ПУБФ концентрирует перцептуальную значимую информацию в низкочастотном канале. Для четырехполосного фильтра, размер канала равен $N/4 \times N/4$, то есть размер изображения уменьшен в 16 раз. Этот низкочастотный канал передается на вход U-net, которая на выходе выдает маску определяющую вероятность того, что данный пиксель принадлежит пешеходному переходу. Но эта маска также размера $N/4 \times N/4$. Для ее восстановления до размеров исходного изображения можно использовать блок синтеза Q-ПУБФ. Операции в блоке синтеза, являются обратными операциями блока анализа, т. е. позволяют восстановить исходный сигнал из его субполосных компонент.

Для проверки предложенного решения был выбран набор данных из 3083 изображений, на части из них пешеходного перехода не было. Снимки были сделаны как днем, так и ночью, а также при плохих погодных условиях. Все изображения трехцветные, размером $256 \times 256 \times 3$. Система анализа Q-ПУБФ преобразует входное изображение в тензор $16 \times 64 \times 64 \times 3$. Из этого тензора выделяется низкочастотная компонента исходного изображения размером $64 \times 64 \times 3$ и передается на вход нейронной сети. На выходе получаем маску, которая представляет собой двумерный тензор размером 64×64 . Каждое значение маски представляет собой вероятность того, что соответствующий пиксель принадлежит пешеходному переходу на входном изображении. Это решение сравнивалось с нейросетью той же архитектуры, которая распознавала изображения исходного размера [6]. Для корректности сравнения, из маски 64×64 системой синтеза Q-ПУБФ восстанавливалась маска размера 256×256 . Маска 64×64 использовалась как низкочастотный канал, значение остальных каналов задавалось нулями.



Рисунок 1 – Пример изображения (слева) и его маски (справа)

В качестве метрики качества сегментации использовалась Intersection over Union (IoU) [7]. Она измеряет сходство между предсказанной и размеченной областями. Чтобы её вычислить нужно найти отношение пересечения областей к их объединению. IoU вычислялось на масках размером 256x256.

Набор данных был случайным образом разделен на обучающий и проверочный наборы. Обучающий набор состоит из 2159 изображений (70% от общего набора данных), а набор для проверки составляет 924 изображения. Нейронная сеть тренируется на обучающем наборе, а затем предсказывает область пешеходного перехода на наборе проверки.

Результаты работы натренированных приведены в таблице 1. Использование Q-ПУБФ в сочетании с U-net ведет к незначительному снижению результата качества сегментации, по сравнению с работой U-net на исходных изображениях, однако значительно сокращает время распознавания. Таким образом, параунитарные банки фильтров могут использоваться в качестве инструмента для повышения производительности сверточных нейронных сетей. Более того, согласно проведенному эксперименту, использование параунитарных банков фильтров оказывает незначительное влияние на результат.

Таблица 1 – Результаты работы.

Решение	Длительность тренировки, мин	Длительность распознавания 924 изображений, с	IoU
U-Net	67	202	0.9496
Q-ПУБФ + U-Net	11	21	0.9165

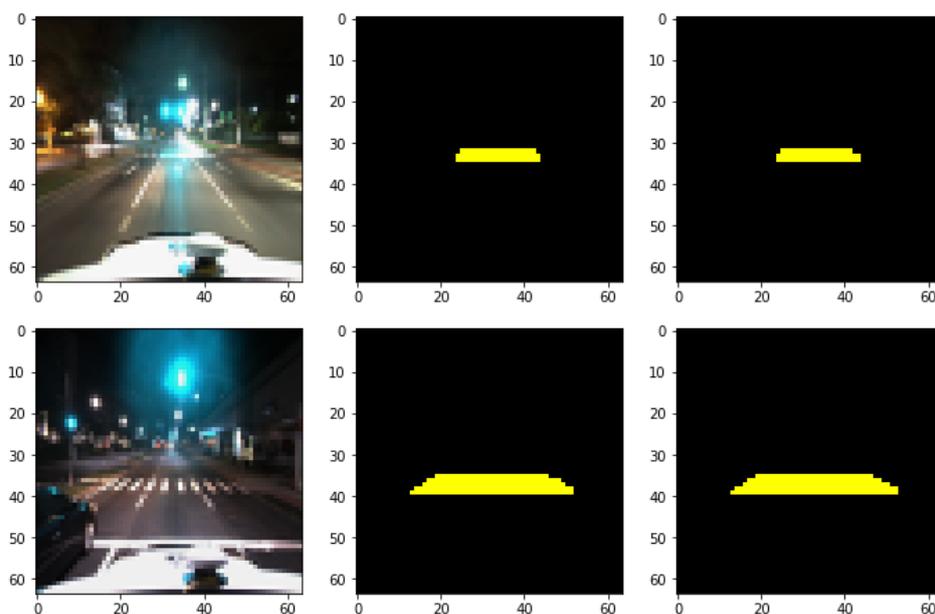


Рисунок 2 – НЧ канал исходного изображения, предсказанная область и область истинности

Список использованных источников:

1. W. H. Organization et al., "Global status report on road safety 2018: summary," World Health Organization, Tech. Rep., 2018.
2. Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard and L. D. Jackel: *Backpropagation Applied to Handwritten Zip Code Recognition*, *Neural Computation*
3. M. A. Malbog, "Mask r-cnn for pedestrian crosswalk detection and instance segmentation," in 2019 IEEE 6th International Conference on Engineering
4. Парфенюк М., Петровский А.А. Параунитарные банки фильтров на основе алгебры кватернионов: теория и применение / М. Парфенюк, А. А. Петровский // «Цифровая обработка сигналов» - 2008 – 15 с.
5. O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*.
6. E. Adaska, A. Lechanka, "UNetX: Real-time Pedestrian Crosswalk Segmentation on Mobile Device", in 2021 PRIP 15th International Conference on Pattern Recognition and Information Processing.
7. M. A. Rahman and Y. Wang, "Optimizing intersection-over-union in deep neural networks for image segmentation," in *International symposium on visual computing*. Springer, 2016, pp. 234–244.

УДК 004.383.3

АРХИТЕКТУРА БЛОКА ОБРАБОТКИ ИЗОБРАЖЕНИЙ(ISP) НА БАЗЕ ПРОЦЕССОРА M1

Протасов А.П. Стаселько И.Д. Позняков Т.Д.

*Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь*

Волорова Н.А - Доцент, кандидат технических наук, зав.кафедрой,

Сиротко С.И. - Доцент, кандидат технических наук

Аннотация: Экспериментально исследована архитектура и программное устройство графического процессора(ISP) в процессоре Apple M1 с помощью языка C и внешнего устройства обработки изображений.

Датчики изображения используются во многих типах устройств сбора изображений, таких как цифровые фотоаппараты, видеокамеры и камеры видеонаблюдения. В последнее время область их применения расширилась и теперь включает интеллектуальные устройства, а полученные изображения предназначены не только для хранения, но и для взаимодействия между человеком и компьютером. Для удовлетворения многих целей датчиков изображения роль улучшения изображения как никогда важна.

ISP (Image Signal Processor) - процессор сигналов изображения является одним из неоптических устройств, которые повышают качество захваченных необработанных изображений. Данный процессор использует несколько алгоритмов обработки изображений, включая шумоподавление и баланс белого, а также другие алгоритмы улучшения изображения. Специальная аппаратная реализация обеспечивает высокое качество изображения и производительность обработки за счет масштабируемости и гибкости.

Функции ISP могут быть реализованы и на выделенном оборудовании, процессоре общего назначения или процессоре параллельных вычислений. Реализация ISP на процессоре общего назначения может быть подходящей не только для высокого качества изображения сложных алгоритмов, но также для масштабируемости и гибкости звука. Однако, стоимость реализации последнего высока из-за большого объема вычислений, и необходима высокопроизводительная платформа, такая как настольный ПК.

ISP (Image Signal Processor) - это небольшой встроенный в чип M1 процессор для обработки изображений, полученных из камеры устройства.

Процессор состоит из своего локального ЦП, который может запускать локальные алгоритмы обработки изображений, такие как АвтоФокус. Так же в ISP присутствуют встроенные модули для выполнения профессиональных процессов для изображений, интерфейсы передачи изображений. Для обработки изображений различного качества и цвета, в ISP используются различные алгоритмы и принципы, такие как GWA. GWA (предположение серого мира) - Предположение, что цвет в каждом канале датчика в среднем равен серому по всему изображению (полю зрения). Оно является одним из наиболее распространенных предположений, сделанных при попытке оценить спектральное распределение источника света. Обычно каждый канал датчика усредняется независимо. BF - двусторонний фильтр, AC - автоматический контраст, а LTI - переходное улучшение яркости.

Цветные изображения, которые поступают через датчик изображения, могут показывать цвета, отличные от тех, которые видны невооруженным глазом. Чтобы исправить это, используют процесс баланса белого (WB). Алгоритм WB GWA предполагает, что средний цвет изображения - серый. Аналогично, алгоритм Retinex (WR) с белыми пятнами предполагает, что пиксель максимальной интенсивности является белым. Поскольку эти предположения могут быть статистически неверными, итеративный баланс белого (IWB) итеративно уточняет белые пиксели, в то время как голосование по освещению проверяет условия освещения.

$$C_{WB}(x, y) = C(x, y) \left(\sum_x \sum_y R + \sum_x \sum_y G + \sum_x \sum_y B \right) / 3 \sum_x \sum_y C$$

Рисунок 1 – Алгоритмы GWA для ISP в M1

Где C представляет собой один из R, G и B, а CWB представляет значение цвета после балансировки белого. После процесса GWA, Demosaicing - это алгоритм для создания полных каналов RGB, который достигается за счет интерполяции цветных пикселей, которые отсутствуют в изображениях, снятых датчиком. Существует множество алгоритмов, включая эвристические методы, подходы в частотной области, подходы к реконструкции. На рисунке 2 показан поток Adaptive Homogeneity-Directed Demosaicing, Используемый в ISP чипа M1:

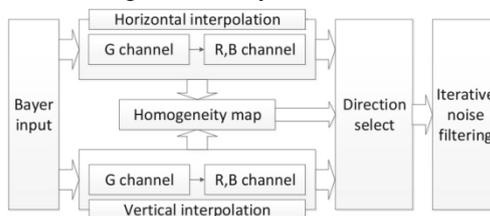


Рисунок 2 – поток Adaptive Homogeneity-Directed Demosaicing

После demosaicing блок цветокоррекции находит особенности цвета и исправляет цветовые артефакты. Уравнение цветокоррекции может быть вычислено в сочетании с последующим преобразованием цветового пространства. Поскольку уравнение цветокоррекции может быть объединено с уравнением преобразования цветового пространства, промежуточный процесс для сохранения значений результата цветокоррекции может быть удален, тем самым сокращая цикл доступа к памяти.

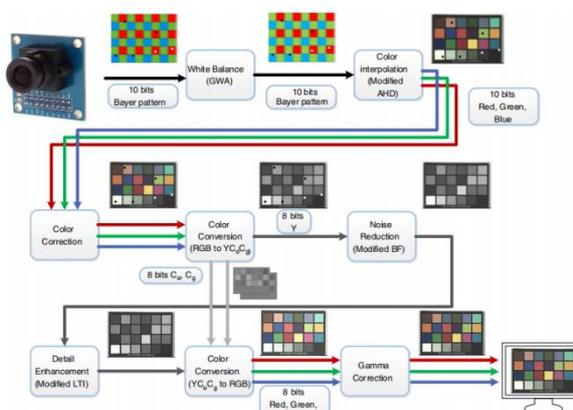


Рисунок 3 – предположительная цепочка ISP

Чтобы проверить производительность предложенной полной цепочки ISP, качество полученных изображений должно сначала пройти коммерчески доступный тест качества изображения.

Эксперименты проводились с использованием датчика изображения CMOS со спецификацией, показанной на рисунке 4. На рисунках 6 и 8 показаны части тестовых шаблонов.

Specification	
Active array size	2592 × 1944
Output formats	10-bit RGB raw
Lens size	1/4 in.
Input clock frequency	6 to 27 MHz
Max S/N ratio	36 dB

Рисунок 4 – тестовые шаблоны

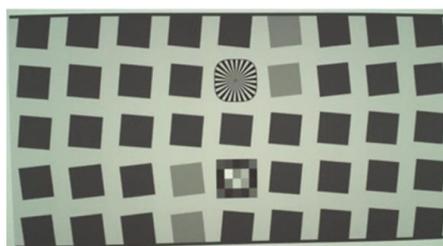


Рисунок 5 – результат, полученный на ISP процессора M1

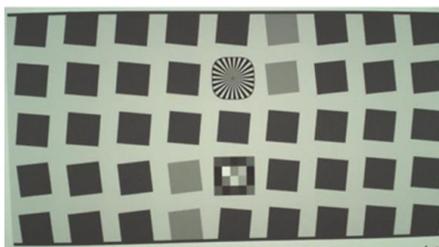


Рисунок 6 – тестовый шаблон качества изображения на доступном датчике изображений на базе CMOS

В данной научной работе было проведено исследование Image Signal Processor в процессоре Apple M1 на примере доступного датчика изображения на базе CMOS. В ходе тестирований данного датчика изображений с использованием языка C и предполагаемыми алгоритмами достигнут результат того, что тесты соответствуют стандартам обработчика изображений на процессоре M1. Целью дальнейших исследований будет являться полная отладка ISP на аппаратном уровне, дальнейшее тестирование процессора в других условиях и с другим программным и аппаратным обеспечением для выявления возможных ошибок и уязвимостей, а так же для предположения оптимизации и энергоэффективности ISP.

Список литературы

1. Mayhand Handley, *M1 Explainer*. – 2021. – № 1. – С. 8–27.
2. Patrick Hansen, Alexey Vilkin, Yury Khrustalev, James Imber, Dumidu Talagala, David Hanwell -- The Role of Image Signal Processing in Efficient Deep Learning Vision Systems. – 2020. – № 1. – С. 2–12.
3. Maxime Pelcat -- Models of Architecture for DSP Systems. – 2020. – № 1. – С. 5–10.

UDC 004.383.3

IMAGE SIGNAL PROCESSOR ARCHITECTURE OF APPLE M1 PROCESSOR

Protasov A.P., Staselko I.D, Poznyakov T.D.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus (style T-institution)

Volorova N.A. - Associate Professor, Candidate of Technical Sciences, Head of the Department

Sirotko S.I. - Associate Professor, Candidate of Technical Sciences

Annotation. We validated the approach by comparing training results from simulated raw images against raw images captured in-lab. We found that processing images with an ISP improves accuracy by an average of 7.0% for a broad set of MobileNet CNNs. Our results indicate that the ISP has a more significant impact, and our results on the larger ResNet-50 and are consistent with this trend.

Keywords. microprocessor technology, microcontroller, electrostatic discharge

ИСПОЛЬЗОВАНИЕ АСИНХРОННЫХ D-ТРИГГЕРОВ ДЛЯ ГЕНЕРАЦИИ УНИКАЛЬНЫХ ЦИФРОВЫХ ИДЕНТИФИКАТОРОВ

Кохновский С.И.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Иванюк А.А. – д.т.н., профессор

В работе представлены результаты эксперимента с использованием асинхронных D-триггеров в качестве устройств генерации цифровых идентификаторов на этапе инициализации ПЛИС.

Достаточно давно была предложена идея использования случайных внутренних физических особенностей цифровых устройств в качестве средств их идентификации. Эта идея была воплощена в физически неклонировуемых функциях (ФНФ). Среди множества различных вариантов реализации особый интерес вызывают расположенные на кристалле ФНФ, что само по себе является значительным преимуществом из-за возможности прямого использования запускаемыми на этом же кристалле приложениями. Одним из первых вариантов реализации стали ФНФ, основанные на принципах работы кольцевых осцилляторов (RO-PUFs) [1].

В качестве источника энтропии выбрана схема на основе D-триггера с кольцевым осциллятором (ROLD), детально описанная в работе [2]. Реализация источника случайности была произведена на плате быстрого прототипирования Digilent Nexys 4 в САПР Xilinx Vivado 2020.2 с использованием элементов комбинационной логики – LUT-блоков. Для проведения эксперимента использованы 8 IP-компонент, содержащих по 32 вышеописанных источника энтропии. С целью получения данных из компонент использован процессор MicroBlaze на рабочей частоте 100 MHz.

На этапе инициализации схема ведёт себя подобно функционированию ФНФ типа SRAM, которые используют в качестве источника случайности колебания свойств транзисторов, возникающих на этапе производства, на микросхеме [3]. Для демонстрации данного поведения вычислено значение вероятности $P_i = \frac{1}{1000} \sum_{e=1}^{1000} Q_i^e$, $Q_i^e \in \{0,1\}$ генерации единичного символа $Q_i = '1', i \in [0; 255]$ на 1000 экспериментах источником случайности (рисунок 1).

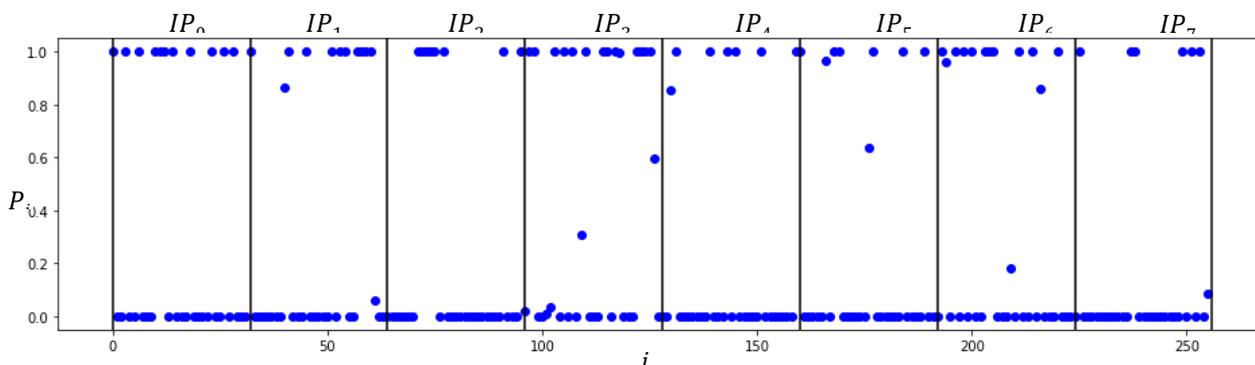


Рисунок 1 – Значения вероятности генераций единичного символа источниками энтропии

В результате эксперимента на этапе инициализации стабильное нулевое значение $P_i = 0$ приняли 170 генераторов, в стабильном единичном значении $P_i = 1$ пребывали 70 источников случайности, а 16 находились в метастабильном состоянии $0 < P_i < 1$.

Данный результат показывает наличие у асинхронных D-триггеров свойств ФНФ типа SRAM, что делает возможным их использование для генерации случайных уникальных идентификаторов. Одним из возможных вариантов использования уникального идентификатора является их применение в качестве начального значения для генерации последовательностей случайных чисел.

Список использованных источников:

1. Gassend, B., Clarke, D., van Dijk, M., Devadas, S.: *Silicon physical random functions*. In: *ACM Conference on Computer and Communications Security*, pp. 148–160. ACM Press, New York, NY, USA (2002)
2. Кохновский, С. И., Иванюк А. А. Влияние длительности работы кольцевого осциллятора на статистические характеристики последовательности бит, сгенерированной аппаратным генератором случайных чисел [Электронный ресурс] / С. И. Кохновский, А. А. Иванюк // *Информационные технологии и системы 2021 (ИТС 2021) : материалы междунар. науч. конф. – Минск : БГУИР, 2021. – 248 с. – С. 126-127. – Режим доступа: https://its.bsuir.by/m/12_130111_1_157684.pdf. – Дата доступа: 08.04.2022.*

3. Holcomb, D.E., Burleson, W.P., Fu, K.: *Initial SRAM state as a fingerprint and source of true random numbers for RFID tags*. In: *Proceedings of the Conference on RFID Security (2007)*

Научное издание

КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ

**58-я научная конференция
аспирантов, магистрантов и студентов**

Сборник статей

(Минск, 18-22 апреля 2022 года)

В авторской редакции

Ответственный за выпуск *Е.В. Калабухов*

Компьютерная верстка: *Е.С. Чижов*