

Разработка архитектуры параллельного процессора для применения в промышленных системах технического зрения

С.А. Байрак, М.М. Татур, М.М. Лукашевич

Белорусский государственный университет информатики и радиоэлектроники,
факультет компьютерных систем и сетей, кафедра ЭВМ
{bairak, tatur, lukashevich}@bsuir.by

Ключевые слова: высокопроизводительные вычисления, алгоритмы обработки изображений, параллельные архитектуры, специализированный процессор

В работе представлены результаты разработки и создания макетного образца высокопроизводительного процессора с оригинальной параллельной архитектурой для решения задач технического зрения в промышленных системах

Введение

Система технического зрения – это программно-аппаратный комплекс, предназначенный для обработки видеоданных и принятия решений о характеристиках реальных объектов и сцен. В зависимости от области применения системы технического зрения, математическое обеспечение, алгоритмы, программное обеспечение и аппаратура значительно отличаются. Современные методы и алгоритмы обработки изображений отличаются большим разнообразием. Архитектура процессора, на котором предполагается их реализация, должна быть оптимальной с точки зрения предоставления различных типов вычислительных ресурсов для их эффективной реализации. Поэтому современный процессор для обработки изображений должен удовлетворять двум основным требованиям: универсальность; вычислительная мощность. С точки зрения аппаратных средств данные для обработки – видеопоток или стационарные изображения – характеризуются следующими основными параметрами: объемом; скоростью поступления; типом методов и алгоритмов для их обработки.

1. Разработка архитектуры параллельного процессора

Характер данных и цель их обработки определяют используемые в системе методы и алгоритмы, которых на сегодняшний день разработано большое количество. При их аппаратной реализации с использованием вычислительной системы можно воспользоваться двумя основными подходами: последовательным; параллельным. Последовательный подход предполагает реализацию методов и алгоритмов путем последовательного выполнения отдельных операций на едином, часто универсальном, вычислительном ядре. Основные плюсы данного подхода – универсальность, простота и гибкость. Основным недостатком – скорость обработки данных. Чаще всего последовательный подход основывается на использовании современных микропроцессорных систем, в том числе и специально предназначенных для решения задач цифровой обработки сигналов (DSP процессоры). Параллельный подход предполагает реализацию методов и алгоритмов путем организации параллельно работающих модулей, выполняющих одновременную обработку различных частей поступающих данных. Для этого чаще всего используют различные микросхемы программируемой логики – FPGA. Основное достоинство данного метода – возможность построения системы с практически неограниченной вычислительной мощностью, а основным недостатком – слабая универсальность и сложность разработки такой системы.

Важным фактором универсальности процессора обработки данных является также наличие достаточного числа интерфейсов, выполняющих следующие основные функции:

- прием данных для обработки, при необходимости в режиме реального времени;
- выдача результата обработки, при необходимости в режиме реального времени;

- организация полнодуплексного канала управления системой обработки.

Входные данные могут поступать в систему обработки посредством стандартного или специализированного интерфейса. В первом случае обеспечивается максимальная универсальность системы обработки данных, так как использование стандартного интерфейса позволяет подключать ее в большое количество уже существующих систем, поддерживающих этот интерфейс.

Структурная схема параллельного процессора обработки изображений представлена на рисунке 1 [1-7]. Основные вычислительные ядра процессора, структурная схема которого представлена на рисунке 1, представлены блоками DSP и FPGA.

DSP блок реализуется на базе специализированного DSP процессора, в задачу которого входит реализация обработки данных с использованием последовательного подхода. На сегодняшний день существует большое количество различных DSP процессоров, отличающихся различной вычислительной мощностью и набором дополнительного функционала под конкретные варианты их применения. Для реализации DSP блока оптимально использовать DSP процессор компании TexasInstruments TMS320C6678. Он представляет собой многоядерный, гетерогенный DSP процессор который состоит из нескольких вычислительных ядер, поддерживающих операции как фиксированной, так и плавающей арифметики.

FPGA блок реализуется на базе микросхемы программируемой логики FPGA. В качестве таковой используется микросхема Artix-7 компании Xilinx. Это младшее семейство седьмой серии, которое обладает достаточными аппаратными ресурсами для реализации алгоритмов и методов обработки изображений с использованием параллельного подхода.

DSP процессор и FPGA связываются между собой посредством локальных высокоскоростных интерфейсов для осуществления межпроцессорного взаимодействия.

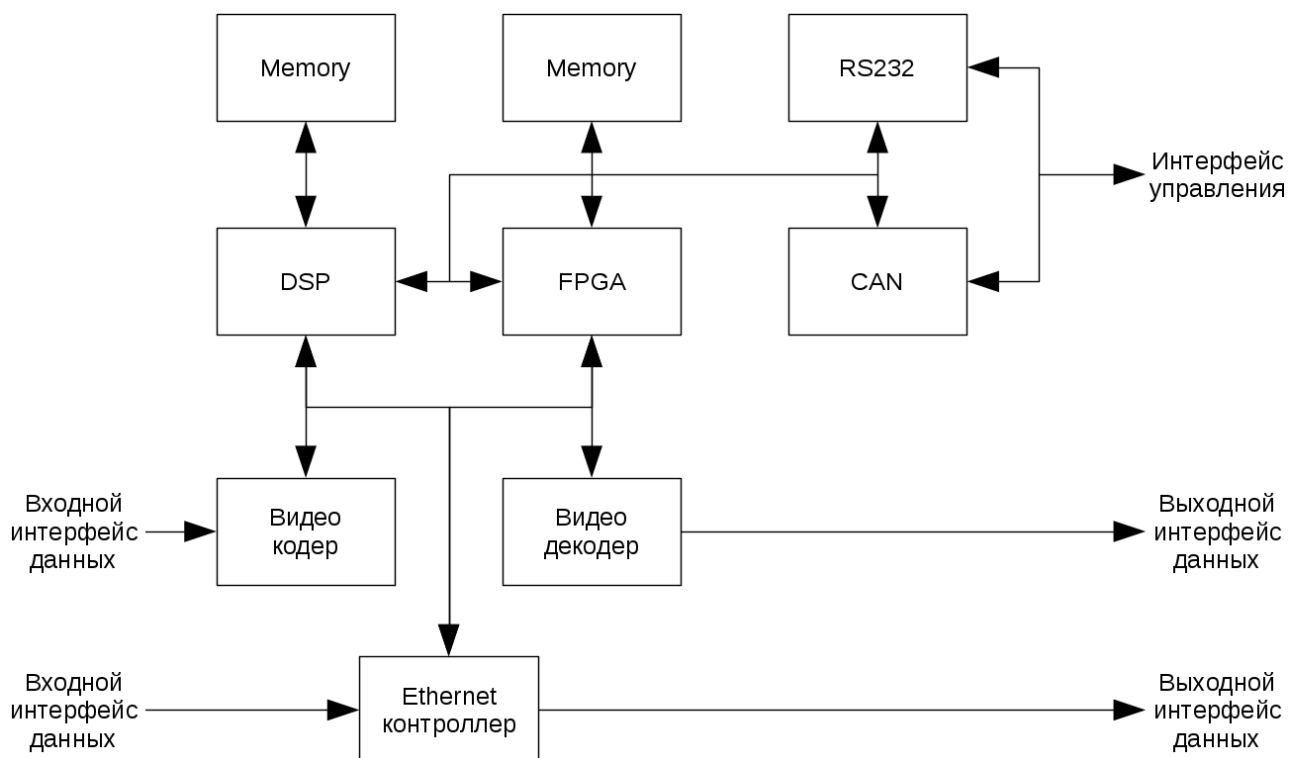


Рисунок 1 – Структурная схема процессора обработки изображений

Большинство алгоритмов и методов обработки практически невозможно реализовать без оперативной памяти, необходимой для буферизации входных/выходных данных, хранения промежуточных данных, формирования, хранения и выдачи результатов обработки. Поэтому каждый вычислительный модуль (DSP и FPGA) имеет собственный модуль памяти, объем которой может меняться в зависимости от конкретных условий применения системы обработки. В качестве модулей памяти используется динамическая память DDR3, представляющая собой оптимальное соотношение объема хранимых данных и скорости доступа. Кроме этого, данный тип памяти поддерживается на аппаратном уровне используемыми микросхемами DSP процессора и FPGA.

Представленная схема построения процессора позволяет реализовать обработку изображений подавляющим большинством существующих методов и алгоритмов. Основной задачей для их оптимальной реализации становится правильное разбиение всех необходимых алгоритмов по двум существующим процессорным ядрам, каждое из которых предназначено для реализации определенного классов алгоритмов: DSP процессор – для реализации последовательных алгоритмов, FPGA – для реализации параллельных. Это позволяет достаточно быстро разработать систему под конкретную задачу и добиться при этом хорошего качества обработки при минимальных затратах вычислительных ресурсов. На рисунке 2 представлена функциональная схема процессора.

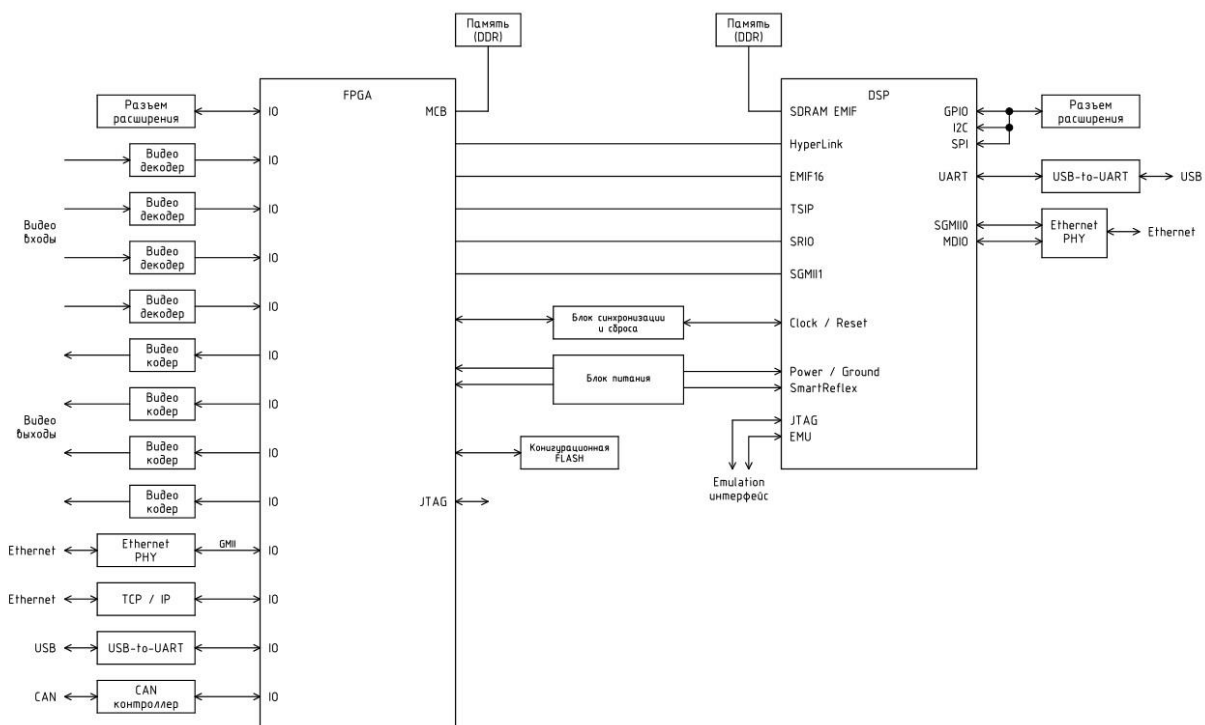


Рисунок 2 – Функциональная схема процессора

2. Выбор печатной платы для макета

Для макетирования универсальных параллельных процессоров необходимо использовать аппаратное обеспечение, позволяющее использовать как параллельный, так и последовательный подходы при реализации необходимых алгоритмов обработки. Это требует наличие как микросхемы процессора, так и микросхемы программируемой логики, связанных между собой высокоскоростными интерфейсами для обмена данным. Такие платы в большинстве случаев разрабатываются под конкретные задачи и отличаются высокой стоимостью и ограниченной универсальностью.

В этом случае оптимально, с точки зрения стоимости и простоты макетирования, является использование различных систем на кристалле – микросхем, которые включают в себя как микропроцессорное ядро, так и набор программируемой логики, связанных между собой

высокоскоростной внутренней шиной. В общем случае и микропроцессорное ядро, и программируемая логика по скорости работы будут уступать отдельным микросхемам, но для макетирования и отработки алгоритмов обработки данных это ограничение не является принципиальным.

В качестве системы на кристалле можно использовать микросхемы платформы Zynq-7000 от компании Xilinx, которые содержат двух-ядерный процессор ARM Cortex-A9, блок программируемой логики, а также широкий набор периферии, что обеспечивает оптимальное сочетание гибкости, программируемости и производительности на одном кристалле. В качестве вычислительного ядра можно использовать отладочную плату MicroZedZynqEvaluationKit.

3. Демонстрационный макет системы технического зрения

На рисунке 3 представлена система обработки видеопотока на базе выбранной печатной платы для макета

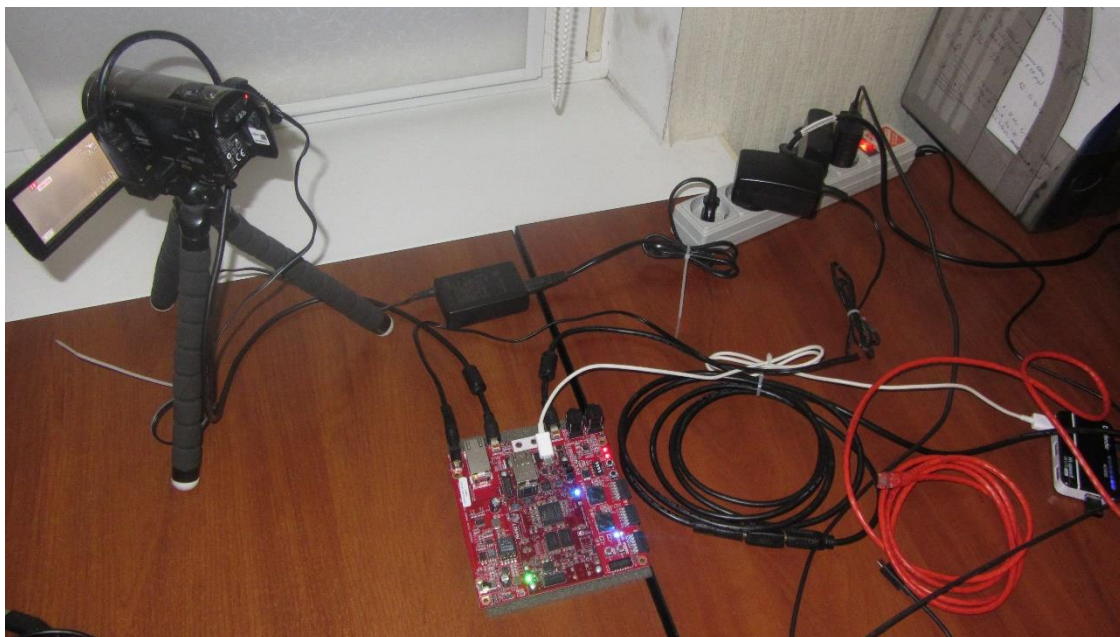


Рисунок 3 – Система обработки видеопотока на базе выбранной печатной платы для макета

Макет системы технического зрения состоит из следующих частей:

1. Блок регистрации видеоданных.
2. Блок интерфейсный.
3. Блок обработки.
4. Блок вывода обработанных видеоданных.

Структурная схема макета представлена на рисунке 4.

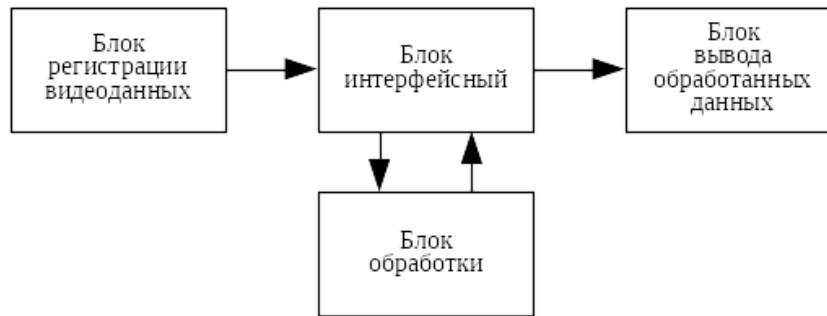


Рисунок 4 – Структурная схема макета системы технического зрения

В качестве блока регистрации видеоданных используется видеочкамаера Panasonic HC-V500 (рисунок 3). Эта камера позволяет получить видеопоток AVCHD – 1920 x 1080 – 60 кад./сек. – 28 Мб/сек по интерфейсу HDMI.

Блок интерфейсный реализован на базе платы MicroZedEmbeddedVisionCarrierCard. В состав этой платы входят различные периферийные модули, из которых в данном макете используются следующие:

- входной HDMI интерфейс на базе микросхемы ADV7611 компании AnalogDevices;
- выходной HDMI интерфейс на базе микросхемы ADV7511 компании AnalogDevices.

Входной HDMI интерфейс выполняет функции приема и декодирования HDMI сигнала в параллельный видеосигнал, для его последующей обработки на соответствующем блоке. Выходной HDMI интерфейс выполняет обратное преобразование видеосигнала из параллельного интерфейса блока обработки в HDMI.

В качестве блока вывода обработанных данных используется обычный монитор со входом HDMI.

Вся обработка видеопотока выполняется на блоке обработки, который реализован на базе платы MicroZedZynqEvaluationKit. В состав данной платы входят:

- вычислительное ядро на базе микросхемы XilinxZynq 7020 SOC;
- набор интерфейсов в составе:
 - 10/100/1000 Ethernet;
 - USB Host 2.0, а также USB-UART мост;
 - MicroSD флеш карта;
 - набор кнопок и светодиодных индикаторов.
- память:
 - оперативная память – DDR3, 1 Гбайт;
 - flash память – QSPI, 128 МБит

Блоки интерфейсный и обработки соединены между собой посредством двух 100 контактных разъемов MicroHeader.

Для программирования и отладки вычислительного ядра используется интерфейс JTAG.

Управление макетом во время работы осуществляется с обычного персонального компьютера через USB интерфейс с использованием USB-UART моста.

4. Система предварительной обработки видеоданных реального времени

На базе макета реализована система улучшения видеопотока с камеры методами предварительной обработки изображений.

Для этого видеоданные с камеры поступают на интерфейсный блок где преобразовываются в параллельную форму и подаются на блок обработки. Так некоторые используемые методы обработки изображений не могут работать с видеоданными «на лету», поступающие данные сначала сохраняются в кольцевом буфере динамической памяти. Размер буфера позволяет хранить до 5 кадров одновременно, что абсолютно достаточно для используемых методов обработки.

Сохраненные данные считываются кадр за кадром и обрабатываются специализированными аппаратными модулями, соединенными между собой в конвейер обработки. Специализированные аппаратные модули реализованы на базе программируемой логики микросхемы Zynq компании Xilinx и позволяют выполнять следующие функции:

- деинтерлейсинг;
- преобразование цветового пространства RGB, YUV 4:4:4, YUV 4:2:2, YUV 4:2:0;
- масштабирование;
- изменение частоты кадров в секунду;
- гамма коррекцию;
- фильтрацию шумов;
- выделение границ.

Структурная схема системы модуля обработки представлена на рисунке 5.

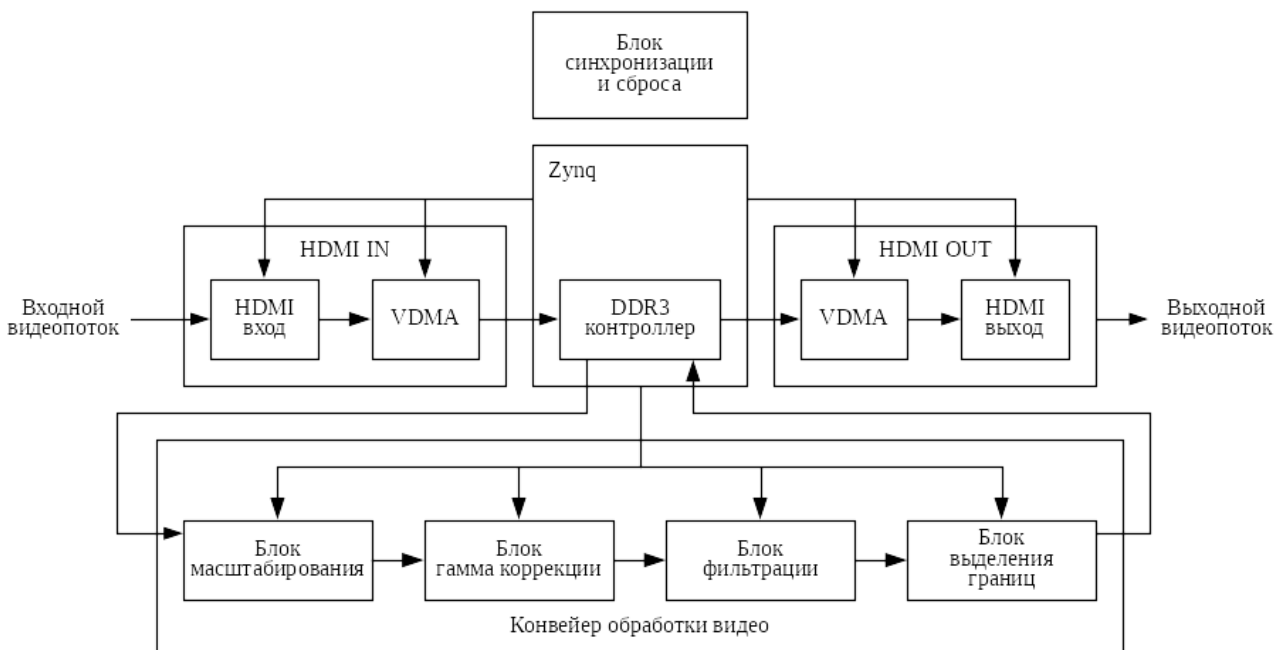


Рисунок 5 – Структурная схемы системы модуля обработки

Заключение

В работе предложена схема построения параллельного процессора, позволяющая реализовать обработку изображений подавляющим большинством существующих методов и алгоритмов. Определено, что основной задачей для их оптимальной реализации является правильное разбиение всех необходимых алгоритмов по двум существующим процессорным ядрам, каждое из которых предназначено для реализации определенного класса алгоритмов: DSP процессор – для реализации последовательных алгоритмов, FPGA – для реализации параллельных. Указанный подход позволяет разработать систему под конкретную задачу и добиться при этом хорошего качества обработки при минимальных затратах вычислительных ресурсов.

Определена структура макета системы технического зрения: блок регистрации видеоданных; блок интерфейсный; блок обработки, блок вывода обработанных видеоданных. На базе макета реализована система улучшения видеопотока с камеры методами предварительной обработки изображений. Основой блока управления является аппаратное ядро процессора Zynq, в состав которого входит встроенный контроллер динамической памяти. Блок масштабирования кроме выполнения функции масштабирования выполняет функции деинтерлейсинга и преобразования цветового пространства. Блоки гамма коррекции, фильтрации и выделения границ выполняют функции, соответствующие их названиям.

Литература

- [1] С.А.Байрак, Д.Н.Одинец, М.М.Татур Параллельный процессор идентификации образов. Минск //Технологии безопасности. 2012, № 1 (22).С.46–47.
- [2] Н.Л.Вереник, Е.Н. Сейткулов, М.М.Татур Разработка проблемно-ориентированных процессоров семантической обработки информации. Минск// Электроника Инфо. 2012, № 8. С.95-98.
- [3] А.В. Белевич , В.И. Луцкий, Д.Н. Одинец, М.М.Татур, НгуенЧунг Тин, ЗыонгКуокХоангМоделирование алгоритмов управления автоматических трансмиссий по обеспечению плавного включения передач//Изд. «Радиотехника». Нейрокомпьютеры: разработка, применение. 2013,№2.С.40-44.
- [4] Н.Л.Вереник, А.И.Гирель, Е.Н.Сейткулов, М.М.Татур Имитационная модель векторного процессора на примере задачи поиска пути в графе //Искусственный интеллект. 2013, №4(62). С.89-100.
- [5] С.Н.Боранбаев, М.Отелбаев, Е.Н.Сейткулов, Н.Л.Вереник, А.И.Гирель, М.М.Татур Алгоритм поиска кратчайшего пути в графе для семантического процессора//Вестник КазНТУим.К.И.Сатпаева, Алматы.2013, №4(98).С.290-296
- [6] Е.Н.Сейткулов, С.Н.Боранбаев, М.Отелбаев, Н.Л.Вереник, А.И.Гирель, М.М.Татур Модель семантического процессора с параллельной архитектурой // Вестник КазНТУим.К.И.Сатпаева, Алматы. 2013, №4(98). С.283-290.
- [7] M.Tatur Problem-Oriented Processors for the Solving of Classification Tasks //Journal of Information, Control and Management Systems (Slovakia). 2013. vol.11. No.2. P.155-164.