

Список литературы:

1. Азаркевич, О. Ю. Мобильное приложение для журнала-тренажера по программе начальной школы / О.Ю. Азаркевич, Д.И. Черняк, А.С. Кобайло // 67-я научно-техническая конференция учащихся, студентов и магистрантов: сб. науч. работ : в 4-х ч. – Минск: БГТУ, 2016 г. С. 121–124.

УДК 681.5.08

**ЭМУЛЯТОР СПЕЦИАЛИЗИРОВАННОГО МИКРОПРОЦЕССОРА
ДЛЯ УЧЕБНОГО ПРОЦЕССА**

А. С. КОБАЙЛО

Учреждение образования

«Белорусский государственный технологический университет»

Эмулятор специализированного микропроцессора с усеченным набором команд предназначен для изучения основ архитектуры микропроцессорных систем и программирования на языке Ассемблера на начальном уровне. Базируется на упрощенной структуре, позволяющей имитировать выполнение основных команд языка Ассемблера. Содержит средства визуального отслеживания хода выполнения команд и программ.

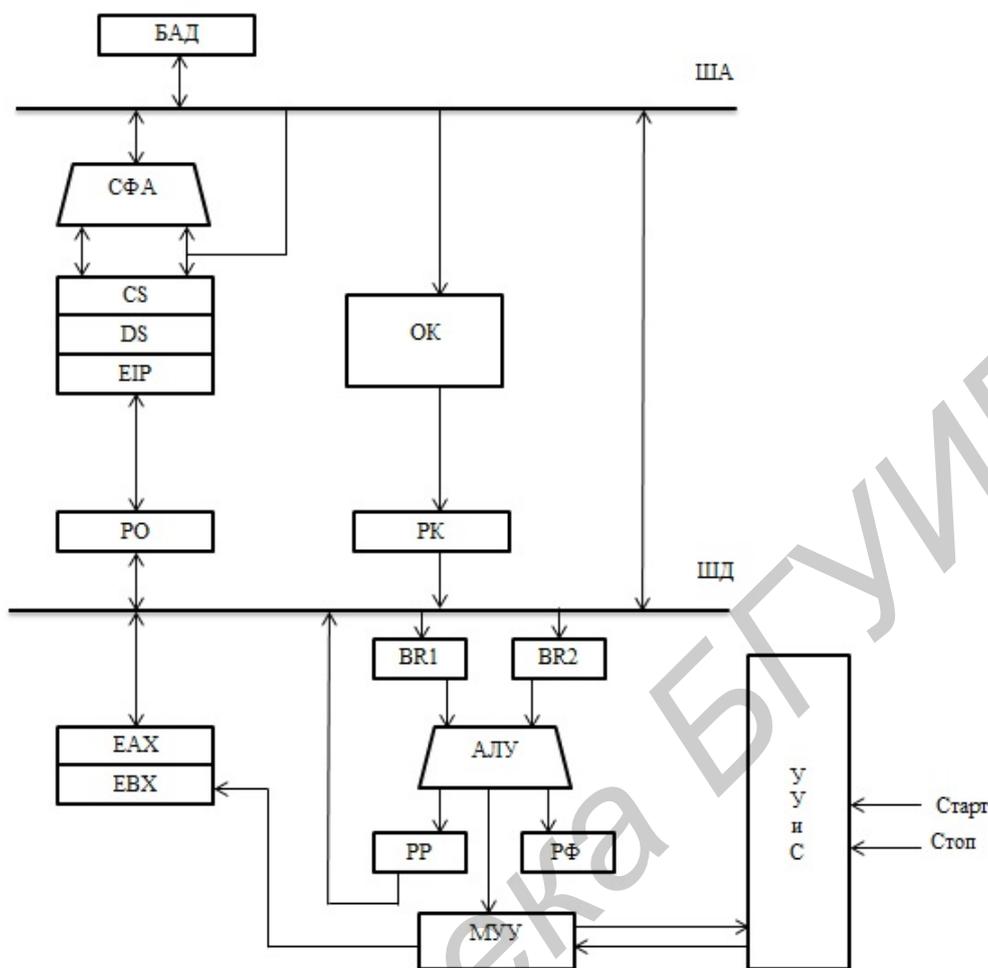
Ключевые слова: эмуляция, микропроцессор, команда, архитектура, регистр, язык Ассемблера.

Средства эмуляции систем различного назначения широко используются при изучении таких систем и методики работы с ними. В частности, изучение архитектуры микропроцессора и микропроцессорных систем и их программирования предполагает наличие либо самого микропроцессора, либо его программной модели – эмулятора. Использование для этих целей известных программных средств, например, эмулятора микропроцессора Intel 8086 *Emu 8086*, требует достаточно глубоких знаний в области архитектуры вычислительных систем и согласно прилагаемой инструкции пользователя этого эмулятора – знания и навыков программирования на языках Ассемблера и на начальной стадии изучения этих компонентов может оказаться сложным и в силу этого неэффективным. Для облегчения изучения начинающими программистами архитектурной организации микропроцессора и основ программирования на языке Ассемблера разработан эмулятор некоторого специализированного микропроцессора с усеченным набором выполняемых им ассемблеровских команд.

Структура микропроцессора содержит все основные блоки типового современного микропроцессора и представлена на рисунке 1.

Сокращения на схеме:

- БАД – буфер адреса данных, используется для передачи и считывания данных из оперативной памяти;
- СФА – сумматор физического адреса, блок формирования физического адреса из сегмента и смещения;
- CS – регистр сегмента команд, косвенно адресует сегмент памяти, в котором хранится выполняемая программа;



Рисунок

Рисунок 1 – Структурная схема специализированного микропроцессора

- DS – регистр сегмента данных, косвенно адресует сегмент памяти, содержащий обрабатываемые данные;
- EIP – программный счетчик (указатель программы), хранит адрес очередной команды, которая должна извлекаться из сегмента команд памяти;
- BR1, BR2 – буферные регистры операндов АЛУ, используются для временного хранения операндов арифметико-логических операций;
- РО – регистр обмена, который используется для обмена между устройством сегментации и внутренней шиной процессора;
- ОК – очередь команд;
- РК – регистр команд;
- EAX – аккумулятор, в основном участвует в арифметических операциях и предназначен для хранения одного из операндов или результата операции;
- EBX – регистр данных, в основном в нем хранится начальный адрес обрабатываемого массива данных;
- АЛУ – арифметико-логическое устройство, предназначенное для выполнения арифметических, логических и сдвиговых операций. Операнды поступают в АЛУ из регистров общего назначения и/или памяти. Результат операции возвращается в регистр общего назначения или в память;
- РФ – регистр флагов содержит 13 признаков (флагов), характеризующих состояние микропроцессора. Шесть флагов устанавливается в соответствии с

результатом выполненной в АЛУ операцией;

- РР – регистр результата;
- МУУ – местное устройство управление, формирует сигналы управления работой устройства обработки;
- УУиС – устройство управления и синхронизации;
- ША – шина данных;
- ШД – шина адреса.

Работу данного приложения можно условно разбить на несколько частей:

- визуальное отображение работы микропроцессора;
- вывод информации из регистров;
- вывод адреса команды и операнда источника;
- подключение команд Ассемблера и операндов.

В режиме реальной адресации микропроцессор реализует сегментную организацию памяти, при которой физический адрес ячейки памяти формируется путем сложения базового адреса сегмента и относительного адреса ячейки внутри сегмента. Так как микропроцессор оперирует логическими адресами, состоящими из сегмента и смещения, а оперативная память работает с линейными адресами, возникает необходимость преобразования логического адреса в линейный физический. Эту функцию выполняет СФА. Для получения физического адреса СФА сдвигает 16-ти битный сегмент влево на 4 разряда (тем самым, умножая его на 16) и прибавляет к этому числу 16-тибитное смещение. В результате получается 20-ти битный линейный физический адрес, необходимый для работы с оперативной памятью.

В режиме реальной адресации микропроцессор реализует сегментную организацию памяти, при которой физический адрес ячейки памяти формируется путем сложения базового адреса сегмента и относительного адреса ячейки внутри сегмента.

Так как микропроцессор оперирует логическими адресами, состоящими из сегмента и смещения, а оперативная память работает с линейными адресами, возникает необходимость преобразования логического адреса в линейный физический. Этой работой занимается СФА. Для получения физического адреса СФА сдвигает 16-ти битный сегмент влево на 4 разряда (тем самым, умножая его на 16) и прибавляет к этому числу 16-ти битное смещение. В результате получается 20-ти битный линейный физический адрес, необходимый для работы с оперативной памятью.

Команды процессора один к одному соответствуют командам языка Ассемблера. Фактически, они и представляют собой более удобную для человека символьную форму записи – мнемокоды команд и их аргументов. При этом одной команде языка Ассемблера может соответствовать несколько вариантов команд процессора.

Система команд специализированного микропроцессора включает десять команд – ADD, NOP, JMP, SUB, DIV, MUL, MOV, RET, JZ, JNZ.

Окно программного средства информационной системы эмуляции специализированного микропроцессора условно можно разделить на 3 области:

- визуальное отображение работы микропроцессора;
- вывод информации из регистров;
- вывод адреса команды и операнда источника.

Для начала работы программы нужно создать файл с входными данными, в котором должны быть записаны выполняемые команды языка Ассемблера и операнды данных команд. Так как программное средство поддерживает только десять команд (ADD, NOP, JMP, SUB, DIV, MUL, MOV, RET, JZ, JNZ), то в файле должны объявляться только такие команды, которые отвечают заданным свойствам данной программы. Пример используемого файла представлен на рисунке 2.

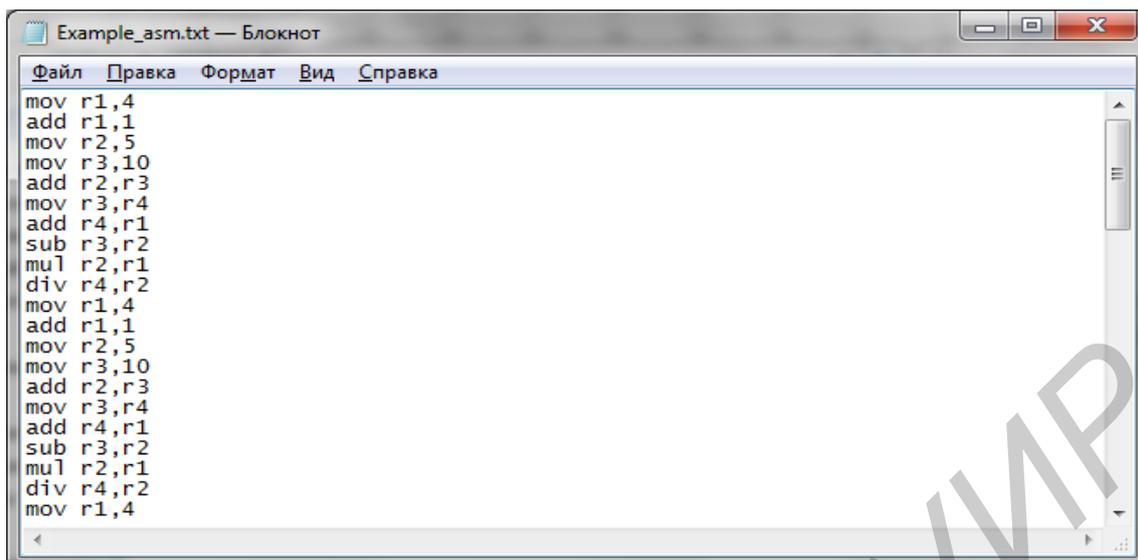


Рисунок 2 – Файл с входными данными

Далее в главном окне программного средства, с помощью кнопки «Загрузить», нужно выбрать ранее описанный файл входных данных. При успешной загрузке файла с входными параметрами, программное средство занесет исходные данные в свою память и в главном окне приложения автоматически отобразится используемая информация. Главное окно программного средства после успешной загрузки файла с входными параметрами представлено на рисунке 3.

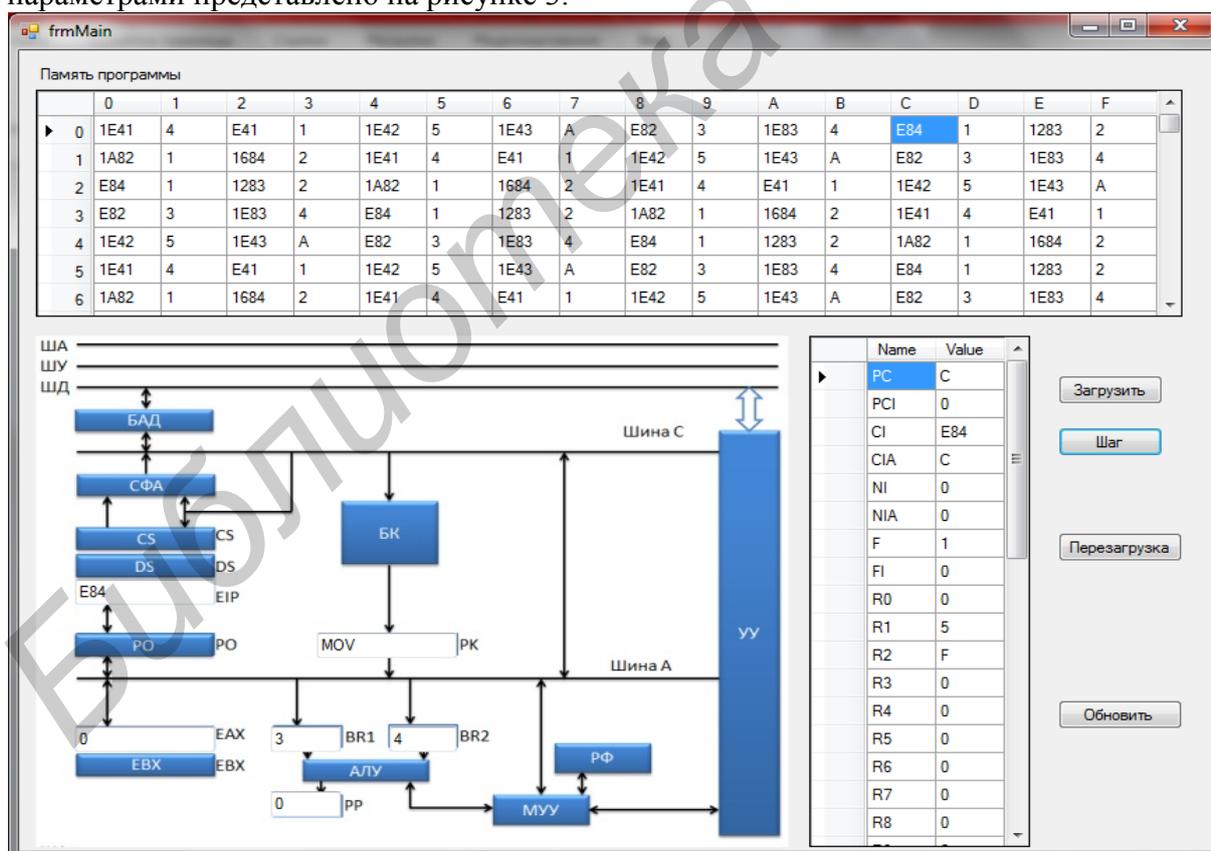


Рисунок 3 – Главное окно программного средства после успешной загрузки файла с входными параметрами

При работе в пошаговом режиме (нажатие кнопки «Шаг») в главном окне отоб-

ражается постепенное изменение всех параметров регистров и элементов работы микропроцессора, соответствующих конкретной очередной исполняемой команде – регистра исполняемой команды, адреса исполняемой команды, значения регистров общего назначения, результата исполняемой операции, значение аккумулятора, значения заданных операндов команд и название используемой операции.

Данное программное средство обладает интуитивно понятным интерфейсом. В случае неправильного управления программным средством появляются уведомления, позволяющие пользователю скорректировать свои действия для дальнейшей работы.

Программа эмуляции может быть использована в учебных целях для наглядной демонстрации работы микропроцессора, а также в для создания студентами высокоэффективных программ.

УДК 681.5.08

РАЗРАБОТКА ИННОВАЦИОННЫХ ТЕХНОЛОГИЙ В ПРОЦЕССЕ ОБУЧЕНИЯ НА ПРИМЕРЕ СОЗДАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ ИССЛЕДОВАНИЯ СОСУДИСТОЙ СТРУКТУРЫ ГЛАЗНОГО ДНА

А. С. КОБАЙЛО

Учреждение образования

«Белорусский государственный технологический университет»

С целью повышения эффективности обучения специальным дисциплинам в рамках высшей школы целесообразно практиковать закрепление изучаемой информации и приобретение практических навыков на решении актуальных задач из различных областей человеческой деятельности. Одной из таких областей является медицина, где среди множества подлежащих решению задач выделяется и такая, как исследование сетчатки глазного дна. В рамках изучения методологий проектирования информационных систем разработано программное обеспечение системы автоматизации данного процесса.

Ключевые слова: нейронные сети, персептрон, распознавание изображений, нейронные сети, обратное распространение ошибки, глазное дно.

Безусловно эффективным способом мотивации изучения студентами специальных дисциплин в соответствии с программой высшей школы и, следовательно, повышения эффективности их усвоения является ориентация на создание в рамках учебного процесса инновационных технологий, ориентированных на применение в конкретных областях народного хозяйства и других важнейших сферах человеческой деятельности. В качестве примера реализации этого направления рассмотрим решение задачи обработки изображений в области медицины.

На данный момент в медицине широко используется информация, представленная в виде изображений. Наличие большого количества подобной информации заставляет уделять больше внимания к автоматизации ее обработки и анализа. Одним из направлений медицины, нуждающимся в обработке большого объема данных, является офтальмология – исследование сосудистой структуры глазного дна.

Глазное дно исследуется при помощи офтальмоскопа. Оценивается состояние диска (соска) зрительного нерва (видимая при офтальмоскопии начальная, интраокулярная часть зрительного нерва), сетчатки, сосудов глазного дна.

Данные с офтальмоскопа будут использоваться для анализа в программном средстве. Задача распознавания осложнена тем, что работа офтальмоскопа не