

ТЕХНОЛОГИЯ ИНТЕРАКТИВНОГО ОБУЧЕНИЯ ОСНОВАМ ЦИФРОВОЙ ЭЛЕКТРОНИКИ

М.С. Долинский, ГГУ им. Ф. Скорины, г. Гомель

Введение

На сегодня имеется достаточно оснований говорить о следующих, набирающих силу, тенденциях:

- резкая дифференциация студентов внутри одной группы по знаниям, умениям, навыкам и мотивации к занятиям;
- готовность значительной части студентов получать оценки без соответствующих знаний, используя все возможные средства обмана преподавателя;
- стремление существенной части студентов пропускать занятия в максимальном количестве;
- неготовность большинства студентов усваивать теоретический материал непосредственно на лекции;
- отсутствие почти у всех студентов навыков эффективной учебной работы на практических занятиях (поддержание тишины, самостоятельность в выполнении заданий, наличие тетрадки по предмету, ручки и пр.).

В таких условиях уже не срабатывают или срабатывают очень плохо стандартные методики обучения, как правило, ориентированные на хорошо подготовленных и мотивированных на обучение студентов.

Данная работа посвящена описанию модифицированной методики обучения, учитывающей описанные выше негативные тенденции. Отметим также, технической базой такого подхода к обучению являются комплексы программных средств, разработанных в ГГУ им. Скорины под руководством автора [1–7].

Содержание обучения

Технология обучения ориентирована на недельные циклы. Каждая неделя начинается с теоретического занятия, затем одно или несколько практических обучающих занятий и в конце недели – контрольная работа. Далее представлена методика обучения.

Организация и функционирование ЭВМ.

Раздел 1. Элементная база вычислительной техники: введение в предмет; синтез комбинационных схем по таблицам истинности; минимизация булевых функций картами Карно; базовые комбинационные схемы: дешифратор, шифратор, мультиплексор, сумматор; базовые схемы памяти: триггер, регистр, счетчик, ОЗУ, ПЗУ.

Раздел 2. Микропроцессор: введение в ассемблер, простейшая обработка символьных строк, таблица команд Intel 8086, Форматы команд Intel 8086, дополнительные сведения об ассемблере.

Раздел 3. Микропрограммирование: Основные понятия, синтез операционного автомата, синтез управляющего автомата с жесткой логикой, синтез управляющего автомата с программируемой логикой.

Архитектура вычислительных систем.

Раздел 1. Введение в предмет: технология работы на практических занятиях; микропрограммирование на С-МПА; генерация схем по микропрограммам; разбор решений контрольной №1; архитектура вычислительных систем – основные понятия и тенденции развития; архитектура учебного процессора TCPU.

Раздел 2. Развитие архитектуры процессоров семейства Intel: архитектура микропроцессора Intel 8086, особенности архитектуры МП Intel 80286, 80386, 80486, Pentium, команды MMX, архитектура Sandy Bridge.

Раздел 3. Не-Inel-овские архитектуры процессоров: процессоры цифровой обработки сигналов; транспьютеры, реляционные ассоциативные процессоры (процессоры баз данных); ЭВМ, управляемые потоками данных; супер-компьютеры.

Теоретические занятия

Материал для занятий по каждой теме тщательно отобран по количеству и качеству. С одной стороны, он содержит все необходимые фундаментальные сведения, а с другой – рассчитан на усвоение в течение 20–30 минут лучшими (до трети) студентами. Высвободившееся время используется для работы над только что пройденным материалом в малых группах (2–3 человека) при решении специально подобранных задач нарастающей сложности на изучаемую тему.

Теоретические занятия проводятся в аудитории, снабженной проектором, разворачивающимся экраном, переносным компьютером лектора, точками беспроводного доступа в сеть университета, разводкой питания для переносных компьютеров студентов. Таким образом, на каждом столе (то есть практически перед каждым студентом) имеется персональный компьютер, который во время изложения преподавателем теоретического материала может использоваться для альтернативы или дополнения большого экрана, на который проецируется лекторский материал. После завершения лекционной части студенты могут использовать свои ПК для повторного просмотра теории, а затем задать возникшие вопросы преподавателю.

Для активизации познавательной деятельности студентов с первой минуты лекции (а еще точнее, с первой минуты перемены перед лекцией) студентам доступна командная олимпиада, в которой предложена серия задач возрастающей сложности на изучаемую тему. После завершения объяснения новой теории преподавателем, студенты, при необходимости совещаясь в рамках созданных команд по 2–3 человека, должны проектировать схемы/писать программы/отвечать на вопросы по теме лекции и отправлять на автоматическую проверку. В этот момент на экране отображается динамически изменяющаяся таблица результатов командной олимпиады. Одновременно, для наименее подготовленных студентов доступно обучение по данной теме, пройдя которое, можно решить некоторое количество начальных задач командной олимпиады. Задачи подобраны так, чтобы ни одна, даже самая сильная команда, не могла решить все предложенные задачи (иначе, задачи добавляются), и в то же время, ни одна команда не смогла бы не решить ни одной задачи (поскольку для первых задач предоставляется обучение).

Практические занятия

Практические занятия бывают трех видов: обучение, контрольная работа, контрольные срезы. Важной особенностью обучения является предоставление студентам возможности выбора уровня сложности обучения: индивидуальные задания или обучение. Каждую неделю одно из практических занятий включает контрольную работу. В контрольной работе всем студентам предлагается 20–30 заданий разной сложности. Первая группа заданий включает задачи на тему, которая проходится на текущей неделе. Поскольку задания для всех студентов одни и те же, студенты могут обсуждать их решение после завершения контрольной работы.

Для борьбы с нечестными студентами, отсылающими чужие решения на контрольной работе или во время обучения, введены контрольные срезы. Контрольный срез включает 10 задач на изучаемые в течение семестра темы. Принципиальное отличие контрольных срезов – у каждого студента свой вариант, и свои задачи по каждой из тем. Таким образом, серьезно усложняется возможность посылать на контрольном срезе чужие решения. Важное отличие контрольных срезов заключается также и в том, что их, в отличие от контрольной работы, запрещено писать с собственных ноутбуков – только со стационарных машин в учебном компьютерном классе, под специальным аккаунтом. Кроме того, строгая типизация задач на контрольном срезе однозначно указывает на темы, которые изучены плохо студентом. Каждый студент имеет право писать контрольный срез от одного и более раз, пока его не устроит полученная оценка. Особое внимание во время проведения практических занятий всех видов (обучение, контрольная работа, контрольный срез) уделяется созданию рабочей атмосферы: тишина, наличие тетрадок по предмету и т.д.

Система оценивания

Система оценивания выстроена таким образом, чтобы максимально способствовать достижению следующих целей:

- все студенты ходят на все теоретические и практические занятия;
- на каждом занятии каждую минуту работает каждый студент на пределе своих возможностей;
- минимизировать возможность получения незаработанной оценки.

Оценка выставляется как минимальная из оценки контрольного среза и средней за семестр. Средняя за семестр вычисляется как средняя по контролю теории, контролю практики, обучению, индивидуальным задачам, бонусам и пропускам. Бонусы используются преподавателем для поощрения добросовестного отношения к учебе и эффективной учебно-познавательной деятельности. Например, количество бонусов, заработанное командой на теоретическом занятии прямо пропорционально количеству решенных задач. Колонка «Пропуски» вычитает баллы, причем в геометрической прогрессии в зависимости от количества пропущенных занятий. Таким образом, неотработанный пропуск семи и более занятий делает невозможным получение положительной оценки даже самым лучшим студентом.

Самостоятельная управляемая работа студентов

Вся работа по предмету ведется посредством сайта DL.GSU.BY. Это обеспечивает возможность эффективной самостоятельной работы студентов дома. В частности, они могут отрабатывать пропущенные занятия, решая индивидуальные задания или работая над задачами в обучении.

Существенную помощь в самостоятельной работе оказывает студентам форум, в котором можно задать вопрос по предмету и получить ответ от кого-то из студентов или преподавателей. Ссылки на содержательные вопросы и ответы систематизируются, и данный материал становится дополнительным мостиком для перехода от обучения к решению индивидуальных задач.

Виды практических заданий

1. Проектирование цифровых устройств.

На рисунке 1 представлено задание, в котором студент должен разработать схему описанного устройства в среде HLCCAD [2] и отправить эту схему на тестирование. Полу-

Имя HLCCAD проекта: Error.PRJ
Входное устройство: Error

Название	Размерность	Тип
X	8	вход
Y	8	вход
POS	3	выход
S	4	выход
NE	1	выход

На входы X, Y подаются два 8-ми битных числа. Определить первый несовпадающий бит (POS) и указать сколько всего бит не совпадают в числах (S). Если числа разные, то NE = 0, иначе (если 2 числа идентичны) NE = 1, а POS и S равны 0.

Пример:
X: 10010101
Y: 10010101
NE: 1
POS: 0
S: 0

Отправка решения:

Рисунок 1 – Задание на проектирование цифровых устройств

Тест № 1 / 10

English

Посмотреть ответ

Следующий тест

Отправить

Рисунок 2 – Задание на симуляцию стандартных цифровых узлов

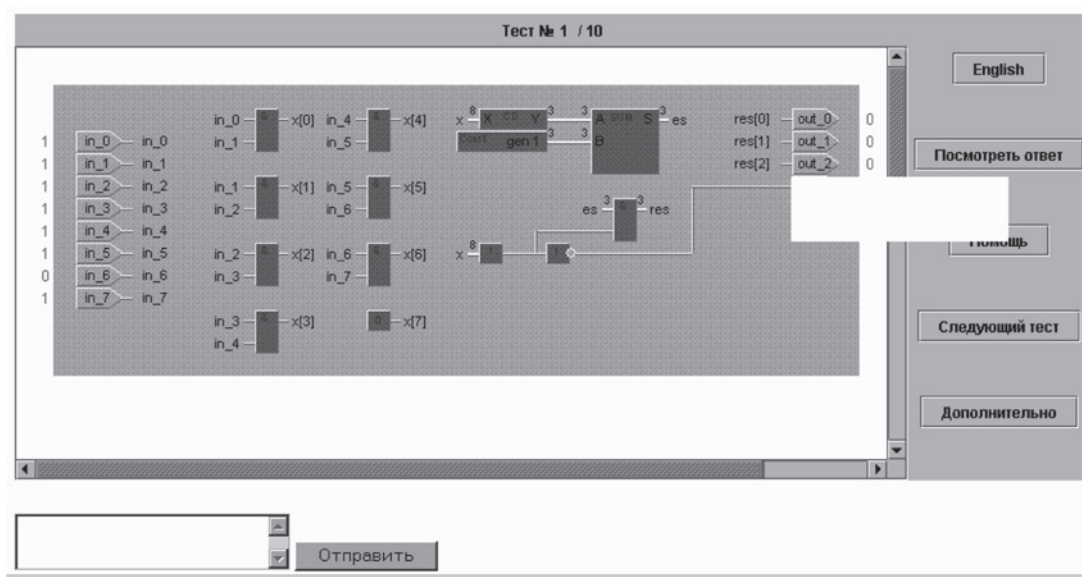


Рисунок 3 – Задание на симуляцию схем цифровых устройств

ченный проект запускается на симуляцию на заранее подготовленном множестве тестов. В случае обнаружения ошибок (несовпадение результатов на выходе симулируемой схемы с эталонными результатами), студент получает соответствующее сообщение с указанием контактов, содержащих ошибочные

значения и модельного времени ошибки. Студент также имеет возможность взять тест и просимулировать его на своем компьютере с целью диагностирования и исправления ошибки и последующей отсылки исправленного проекта на проверку.

2. Симуляция стандартных цифровых узлов.

На рисунке 2 представлено задание, в котором студенту предлагается условное графическое отображение стандартного элемента. Десять раз на его входах появляются случайные комбинации 0 и 1. Для каждой их них студент должен просчитать выход (изменив в случае необходимости значение на нужном выходе кликом мышки). Таким образом, формируется тестовый файл для системы HLCCAD (случайно сгенерированные входные данные и эталонные ответы, созданные студентом). В системе HLCCAD запускается моделирование эталонного устройства (схема которого представлена студенту на картинке) на полученном от студента тестовом файле. В случае несовпадения результатов моделирования и ответов студента, последнему выдается сообщение о времени и месте ошибки.

3. Симуляция схем цифровых устройств.

На рисунке 3 представлено задание, которое содержательно аналогично описанному выше. Отличие заключается в том, что студенту предлагается схема, состоящая из нескольких стандартных устройств. И для получения правильных ответов на выходных контактах студент должен уметь «просчитать схему вручную».

4. Проектирование цифровых устройств (по программе на ассемблере).

На рисунке 4 представлено задание, которое содержательно соответствует первому из описанных заданий. Отличие этого задания заключается в том, алгоритм работы устройства описан не словесно, как в первом случае, а определяется приведенным текстом

```
Имя HLCCAD проекта : 81630.PRJ !!!
Входное устройство : 81630

Составить схему, которая работает как приведенная ниже программа

      jmp      begin
a      dw      1
c      dw      1
b      db      6
d      db      2
res     dw      1
begin:
      mov     al,d
      cmp     al,0
      jle     met
      mov     ax,a
      add     ax,c
      idiv    d
      cbw
      mov     res,ax
      jmp     fin
met:
      mov     al,b
      imul    d
      sub     ax,a
      add     ax,c
      mov     res,ax
fin:   jmp     fin      ;$e

Отправка решения:
Обзор...  Отправить  DelTA
```

Рисунок 4 – Задание на проектирование цифровых устройств (по программе на ассемблере)

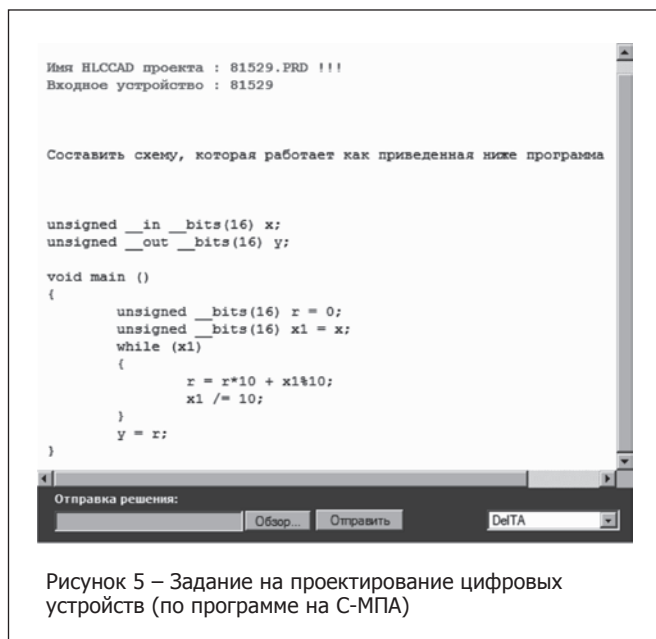


Рисунок 5 – Задание на проектирование цифровых устройств (по программе на С-МПА)

программы на ассемблере процессора Intel 8086. Для выполнения задания требуется проанализировать текст этой программы, понять, что она делает, переформулировать словесно/формульно условие задачи, а после этого разработать и отправить на проверку схему, которая делает аналогичную работу.

5. Проектирование цифровых устройств (по программе на С-МПА).

На рисунке 5 представлено задание, которое соответствует предыдущему заданию. Отличие этого задания заключается в том, что алгоритм работы устройства определяется приведенным текстом программы на языке микропрограммирования С-МПА. Для выполнения задания требуется проанализировать текст этой программы, понять, что она делает, переформулировать словесно/формульно условие задачи, а после этого разработать и отправить на проверку схему, которая делает аналогичную работу.

6. Разработка программы на ассемблере Intel 8086 (С-МПА).

На рисунке 6 представлено задание, в котором студент должен написать программу, в памяти которой объявлены соответствующие входные и выходные переменные и написать программу на языке ассемблера Intel 8086 (или языке микропрограммирования С-МПА), получающую результаты по исходным данным. Для разработки и отладки программы используется среда Winter [3]. После отсылки решение запускается на множестве эталонных тестов. Если при каких-то входных данных программа студента получает неправильный результат, об этом сообщается студенту и ему предоставляется возможность забрать все тесты и добиться работоспособности своей программы, найдя и исправив все ошибки в своей среде Winter.

7. Симуляция инструкций.

На рисунке 7 представлено задание, в котором студенту предлагается инструкция (на рисунке

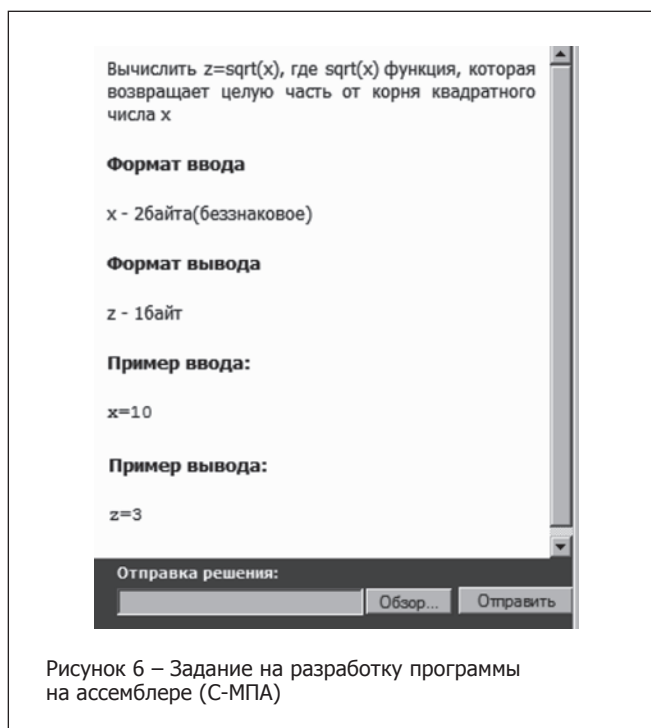


Рисунок 6 – Задание на разработку программы на ассемблере (С-МПА)

инструкция целочисленного деления IDIV), обрамленная несколькими инструкциями пересылки исходных данных и результатов. Исходные данные генерируются случайно и отображаются в верхней части экрана. Студент должен вычислить результат и ввести его в поле ввода. Количество подобных тестов определяется автором задания (на рисунке задание содержит 4 теста). По сгенерированным входным данным и ответам студента формируется тестовый файл, на котором запускается правильное решение этой задачи. Если оно выдает ответы, не совпадающие с ответами студента, последний получает соответствующую информацию.

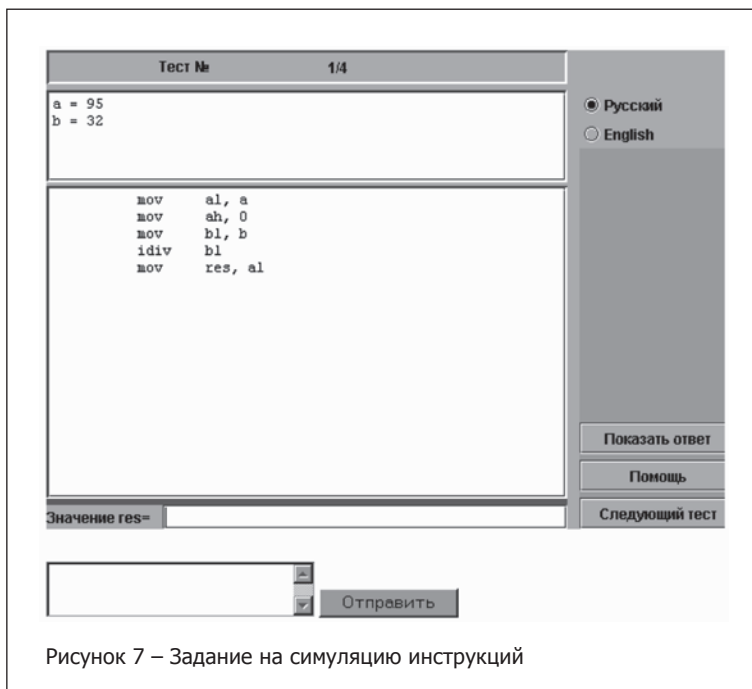


Рисунок 7 – Задание на симуляцию инструкций

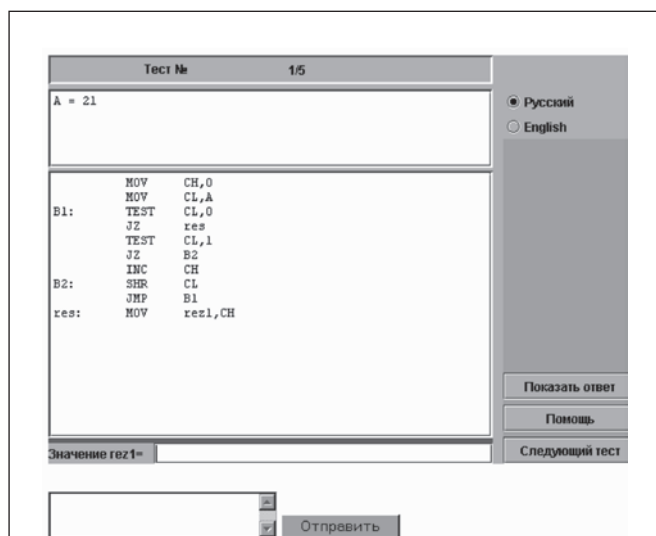


Рисунок 8 – Задание на симуляцию программ на ассемблере Intel 8086

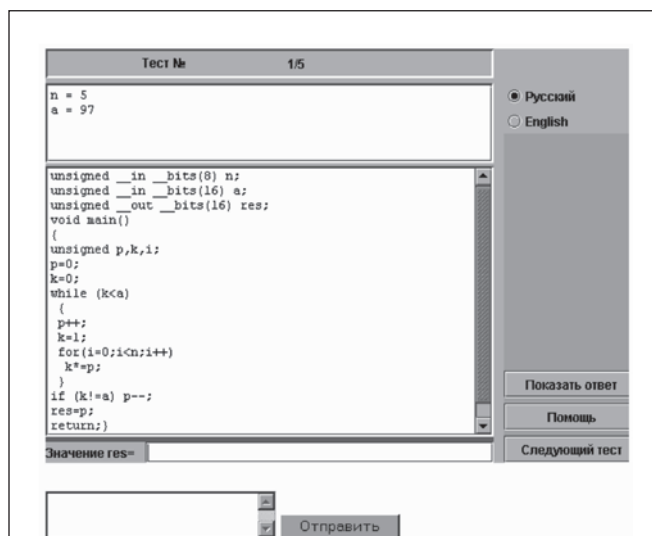


Рисунок 9 – Симуляция программы на языке микропрограммирования С-МПА

8. Симуляция программ на ассемблере Intel 8086.

На рисунке 8 представлено задание, отличие которого от предыдущего задания заключается в том, что студенту предлагается понять, что делает представленная программа на языке ассемблера Intel 8086, и для случайно сгенерированных входных данных вычислить правильные результаты.

9. Симуляция программы на языке микропрограммирования С-МПА.

На рисунке 9 представлено задание, отличие которого от предыдущего задания заключается в том, что для понимания студенту предъявляется программа, написанная на языке микропрограммирования С-МПА [4, 5].

10. Разработать программу по схеме.

На рисунке 10 представлено задание, в котором студенту предлагается понять, что делает схема и написать программу на ассемблере Intel 8086 (другой вид такого задания – на языке микропрограммирования С-МПА), которая выполняет ту же работу.

11. Тестирование программ.

На рисунке 11 представлено задание, в котором студенту предлагаются условие задачи и исходные тексты ее неправильных решений (эти решения получены из протокола тестирования данной задачи, то есть это реальные неправильные решения, выполненные студентами в разные годы). Студент должен придумать такие тесты, чтобы все неправильные решения выдали на этих тестах неверные результаты.

Заключение

В данной статье описана разработанная автором и многократно апробированная технология обучения, ориентированная на работу в группах обучаемых с принципиально различными уровнями мотивации и предварительной подготовки. Серьезной технической основой методики является разработанная инструментальная система дистанционного обучения (Distance Learning Belarus – <http://dl.gsu.by>). Внедрение данной технологии обучения с сентя-

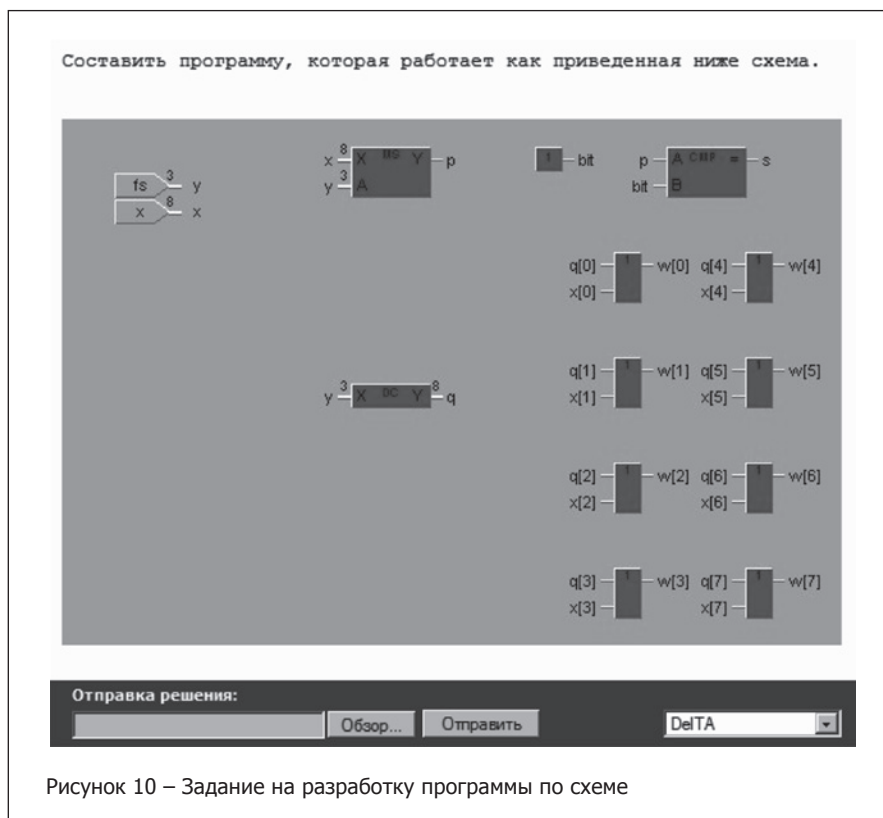


Рисунок 10 – Задание на разработку программы по схеме

Составить тесты в формате WInter для следующей задачи.
Руководство по созданию тестов для WInter.

Набор тестов должен выявить следующие неверные программы.

Обратите внимание!

В тестах перед входными переменными обязательно должно быть две пустые строки. Перед выходными - одна.

Пример файла с одним тестом

Посчитать количество прописных гласных букв латинского алфавита (A, E, I, O, U), находящихся в исходной строке. Признак конца строки - символ #

Формат ввода

IN_STR - строка символов, заканчивающаяся решеткой (#)

Формат вывода

RES 2 байта

Пример ввода:

ADCDEFGHI#

Пример вывода:

2

Отправка решения:

Обзор...

Отправить

DelTA

Рисунок 11 – Задание на тестирование программ

понравилось / не понравилось / предлагаете изменить в лекциях / практике / системе оценивания».

Литература:

1. Долинский, М.С. Использование новых информационных технологий при обучении проектированию цифровых систем и программированию / М.С. Долинский, М.А. Кугейко // Электроника инфо. – 2010. – № 4. – С. 10–13.

2. Долинский, М.С. Редактирование, симуляция и отладка аппаратного обеспечения с помощью HLCCAD / М.С. Долинский, И.В. Коршунов // Электроника инфо. – 2010. – № 6. – С. 22–26.

3. Долинский, М.С. Среда WINTER для редактирования, симуляции и отладки программных систем / М.С. Долинский, И.В. Коршунов // Электроника инфо. – 2011. – № 2. – С. 53–56.

4. Долинский, М.С. Технология разработки алгоритмически сложных цифровых систем с помощью автоматического синтеза микропрограммных автоматов / М.С. Долинский, И.В. Коршунов // Электроника инфо. – 2011. – № 3. – С. 53–57.

5. Долинский, М.С. Автоматический синтез микропрограммных автоматов по С-МПА программам / М.С. Долинский, И.В. Коршунов // Электроника инфо. – 2012. – № 1. – С. 95–98.

6. Долинский, М.С. Учебный микропроцессор TCPU и его применение / М.С. Долинский, И.В. Коршунов // Электроника инфо. – 2013. – № 2. – С. 59–63.

7. Долинский, М.С. Конструктор интерактивных флеш-заданий и его применение / М.С. Долинский, Ю.В. Решетько // Электроника инфо. – 2013. – № 10. – С. 56–63.

бря 2009 года обеспечило значительные сдвиги качества обучения, особенно наименее подготовленной и мотивированной категории студентов. В то же время, наиболее подготовленные студенты также удовлетворены таким подходом к обучению. В частности, это показывается результатами анкетирования по завершению семестра, где всем студентам предлагается ответить на вопросы «Что



поставка электронных компонентов

контрактное производство

тел.: +375 17 290 0082
факс: +375 17 290 0084
e-mail: info@horntrade.net

