

КОНСТРУКТОР ИНТЕРАКТИВНЫХ ФЛЕШ-ЗАДАНИЙ И ЕГО ПРИМЕНЕНИЯ

М.С. Долинский, Ю.В. Решетько,
ГГУ им. Ф. Скорины, г. Гомель

Введение

В материалах [1–6] описаны программно-методические средства, разработанные в ГГУ им. Ф. Скорины, которые могут быть использованы при разработке программно-аппаратных систем, и уже эффективно применяются в учебном процессе. Данная статья посвящена описанию разработанного программного средства [7], ориентированного на индивидуализацию, интенсификацию и повышение качества учебного процесса, в том числе и по предметам, обучающим анализу и синтезу программно-аппаратных систем.

В настоящее время система образования испытывает серьезные проблемы. С одной стороны, объем требуемых к усвоению знаний стремительно увеличивается, а с другой стороны, не менее стремительно увеличивается дифференциация знаний, умений, навыков, уровней мотивации и других психолого-педагогических и физических характеристик, как учащихся, так и педагогов.

Адекватным средством разрешения всех этих противоречий является внедрение компьютеризированных систем обучения. Серьезным препятствием на этом пути является отсутствие эффективных средств разработки интерактивных электронных учебных пособий, которыми могли бы пользоваться непосредственно педагоги-предметники, не имеющие профессионального компьютерного образования.

Одним из самых перспективных подходов в создании интерактивных электронных учебных пособий является использование технологии Macromedia Flash. Как правило, такие учебные пособия эффективны при использовании в реальном учебном процессе. Однако этот подход имеет два существенных недостатка, в значительной степени сдерживающих его применение: высокие требования к профессиональной подготовке разработчика электронного учебного пособия (в частности, объектно-ориентированное программирование, специфическая среда разработки) и значительная трудоемкость (от нескольких человеко-часов на одно задание).

Одним из вариантов разрешения этого противоречия может послужить предлагаемый нами способ автоматизации создания флеш-заданий, который позволяет, сохраняя все достоинства применения флеш-технологий, снизить трудоемкость создания заданий до нескольких человеко-минут на одно задание, а требования к разработчику до пользовательского уровня (текстовый редактор, файловая система). Кроме того, обеспечена возможность взаимодействия конструктора с системой дистанционного обучения DL.GSU.BY (далее, система DL), что позволяет легко организовывать и проводить фронтальные занятия, а также эффективно задействовать все возможности системы DL для вновь создаваемых электронных интерактивных учебных пособий.

Полную теорию по конструктору (включая видео-уроки по разработке заданий), программные средства и примеры их применения можно найти на сайте DL в специальном учебном курсе «Создание флеш-заданий для DL».

Общие принципы создания интерактивных заданий

Имеется набор конструктивных элементов, из которых конструируются задания. Конструктивные элементы делятся на две группы: активные и пассивные. Активные элементы обеспечивают взаимодействие с обучаемым с помощью мыши и/или клавиатуры. Пассивные элементы служат для оформления задания. Для формирования задания разработчик использует «Конструктор заданий». Результат создания задания – сохраняемый на диске текстовый файл, описывающий задание.

Для выполнения заданий используется «Плейер». Он считывает созданный в «Конструкторе» текстовый файл и обеспечивает отображение задания пользователю и взаимодействие с ним в процессе выполнения задания. Когда задание выполнено правильно, «Плейер» оповещает об этом пользователя. Если задание интегрировано в DL, информация о том, правильно или неправильно выполнено задание пользователем, также передается в систему DL.

Возможные области применения

На сегодняшний день интерактивные задания с помощью такого подхода создавались нами для следующих вузовских предметов: «Программирование», «Организация и функционирование ЭВМ», «Архитектура вычислительных систем», «Проектирование аппаратно-программных вычислительных средств». Далее приводится пример использования данной технологии для построения комплекса обучающих заданий по теме «Проектирование цифровых устройств, заданных описаниями или таблицами истинности с помощью минимизации логических функций картами Карно»:

Например, пусть имеются два задания на разработку устройств:

1. Спроектируйте схему, которая работает в соответствии со следующей таблицей истинности

x1	x2	x3	x4	y1	y2	y3	y4
0	0	0	0	0	0	1	0
0	0	0	1	1	0	0	0
0	0	1	0	1	0	0	0
0	0	1	1	0	0	1	0
0	1	0	0	1	0	0	0
0	1	0	1	0	0	1	0
0	1	1	0	0	0	1	0
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	0
1	0	0	1	0	0	1	0
1	0	1	0	0	0	1	0
1	0	1	1	1	0	0	0
1	1	0	0	0	0	1	0
1	1	0	1	1	0	0	0
1	1	1	0	1	0	0	0
1	1	1	1	0	0	1	0

2. На входе дана последовательность из 4 бит. На выход подать сначала все нули исходной последовательности, затем единицы.

Во втором случае вначале требуется составить таблицу истинности по описанию задания.

Таким образом, в общем случае, процесс выполнения задания можно разбить на несколько этапов:

1. Составить таблицу истинности по описанию задания.
2. Для каждого выхода внести 1-ки в соответствующую карту Карно.
3. Покрыть все 1-ки минимальным количеством кораблей (с площадью степенью числа 2) максимальной площади.
4. Выписать логическую функцию для выполненного покрытия.
5. Нарисовать схему в системе проектирования HLCCAD [2].

На лекции, посвященной данной теме (специальность «Программное обеспечение», 1 курс, предмет «Организация и функционирование ЭВМ»), вначале всем рассказывается, как решать поставленные задачи, а потом предлагается в командах решать такие задачи. Студенты используют личные ноутбуки, по одному, как минимум, на команду, как для работы в HLCCAD, так и для отсылки решений на автоматическую проверку в систему DL. На проекторе все время отображаются текущие результаты – какая команда сколько задач решила. В названии команды присутствуют название группы и фамилии членов команды. Все компьютеры и студентов, и преподавателя находятся в сети университета посредством беспроводного соединения.

Однако некоторым студентам, а иногда и командам, недостаточно общего пояснения. В этот момент на помощь приходят описываемые ниже обучающие задания. В них полное задание разбивается на «микроскопические» шаги, которые по отдельности легче выполняются. Кроме того, на каждый из этих шагов имеется подсказка в электронной версии только что прочитанной лекции. Часто бывает, что пройдя данное обучение, команды успевают еще на лекции вернуться к выполнению контрольных заданий. Кроме того, эти учебные задания могут использовать при самостоятельной подготовке студенты, по какой-то причине пропустившие лекцию, или желающие в индивидуальном порядке закрепить пройденный материал.

На рисунке 1 представлено задание, в котором студент должен с помощью кликов мышки по перечисляемым полям (содержащим для выбора ?, 0, 1) установить систематический перечень всех возможных входных значений на данном устройстве (определить область определения булевой функции от четырех переменных x_1, x_2, x_3, x_4). Первые три строки вначале задания также содержали вопросы.

На рисунке 2 представлено задание, в котором студент должен с помощью кликов мышки по перечисляемым полям (содержащим для выбора ?, x_1, x_2, x_3, x_4) выбрать наименования аргументов карты Карно (названия переменных).

На рисунке 3 представлено задание, в котором студент должен с помощью кликов мышки по перечисляемым полям (содержащим для выбора ?, 00, 01, 10, 11) выбрать наименования строк и столбцов карты Карно.

На рисунке 4 представлено задание, в котором студент должен с помощью кликов мышки по перечисляемым полям (содержащим для выбора ?, " "(пробел), 1) выбрать значения клеток карты Карно. Каждый знак вопроса должен быть заменен на 1 или пробел (вместо 0, для лучшей читаемости). Такое задание дается для каждой выходной переменной.

На рисунке 5 представлено задание, в котором студент должен покрыть все единички минимальным количеством

имеющихся кораблей максимальной площади. Корабль не может покрывать пустые поля (поля, в которых функция имеет значение 0). Все корабли имеют площадь, равную степени двойки. Они представлены в верхней области задания. Студент может мышкой указать требуемый корабль, перенести его в требуемую область, повернуть, если нужно и накрыть область из одних единичек. Отметим, что имеются корабли (представленные двумя половинками), которые поддерживают покрытие смежных крайних строк (верхнюю вместе с нижней или левую вместе с правой). Таких заданий также предлагается четыре – по числу выходных переменных.

На рисунке 6 представлено задание, в котором по выполненным покрытиям требуется написать логические функции. В правом верхнем углу задания приводится пример выполнения такого задания. Для упрощения задания студентам вначале предлагается выписать логическую функцию отдельно по термам (использованным при покрытии кораблями). При вводе верных ответов текст подсвечивается зеленым цветом, пока он не содержит ошибок. При вводе ошибочных символов они сразу подсвечиваются красным цветом. В последнем поле ввода предлагается ввести полную логическую функцию выхода.

На рисунке 7 представлено задание, в котором по имеющимся логическим функциям требуется составить функциональную схему устройства из конструктивных элементов системы проектирования HLCCAD, перетаскивая их мышкой на нужную позицию в схеме. Заметим, что и здесь имеется определенная подсказка студентам: если корпус поднесен на свою позицию, то он «прыгает» на нее (с расстояния в несколько пикселей – задается автором задания) и больше не перемещается.

На рисунке 8 представлено задание, в котором студент должен составить схему по логическим функциям, как это делается с обычными пазлами, перетаскивая мышкой в нужное место фрагменты схемы.

На рисунке 9 представлено задание, в котором студент должен для заданных входных значений, просчитать значения на всех контактах схемы и выходных, и промежуточных и указать эти значения с помощью выбора в перечисляемых полях, содержащих ?, 0, 1.

Практика применения подобного обучения показала, что предложенному методу решения задач обучается 100 % студентов, хотя время обучения может сильно варьироваться.



Рисунок 1 – Область определения функции

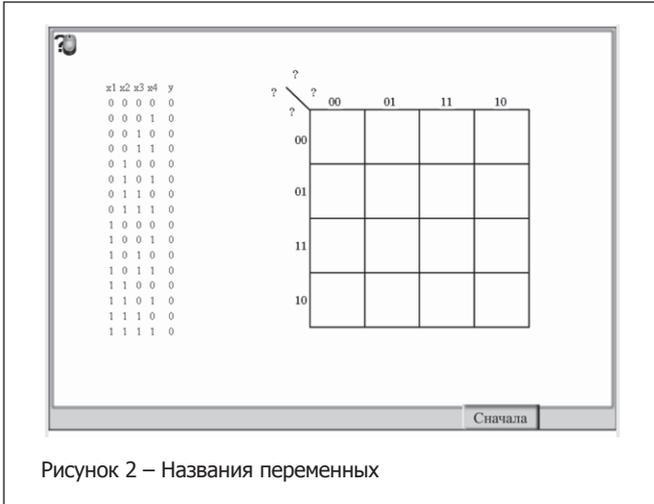


Рисунок 2 – Названия переменных



Рисунок 5 – Покрытие единиц на карте Карно

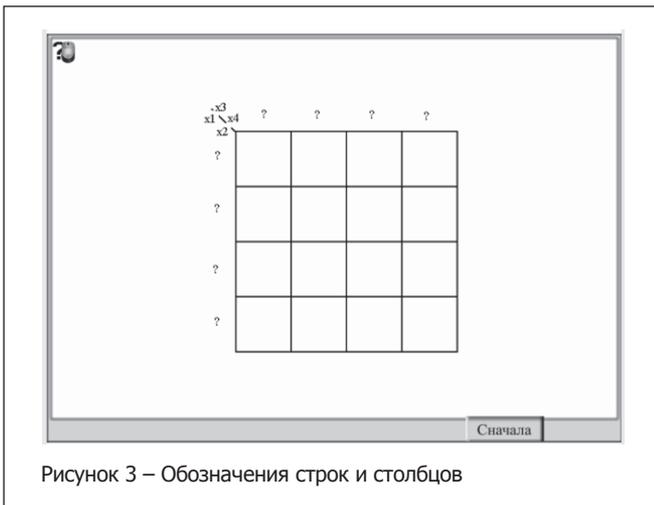


Рисунок 3 – Обозначения строк и столбцов



Рисунок 6 – Логические функции для выходных переменных

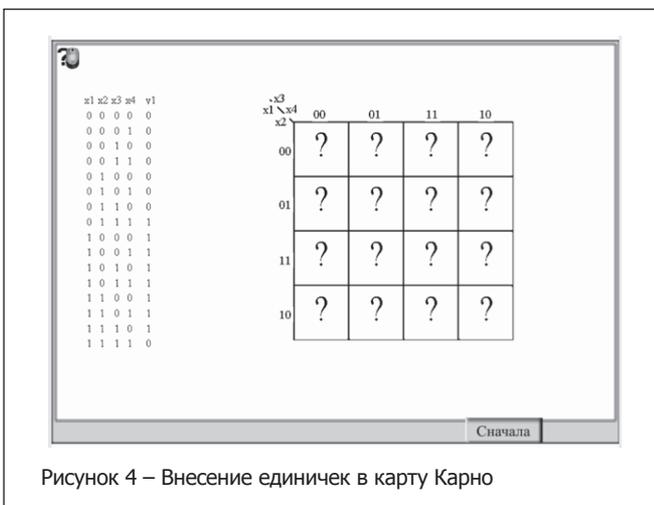


Рисунок 4 – Внесение единиц в карту Карно



Рисунок 7 – Составить схему по логическим функциям

Конструктивные элементы заданий

Задания конструируются из активных и пассивных элементов. К активным элементам заданий относятся такие, которые позволяют обучаемому выполнять некоторые действия мышью и/или клавишами «стрелки», одновременно определяя, правильно ли выполнено задание. Пассивные

элементы предназначены для красочного и информативного оформления заданий.

Активные элементы на момент написания статьи включают в себя таны, перечисляемые поля, области выделения (связные и независимые), области внесения, точки соединения, таймер, поля ввода.

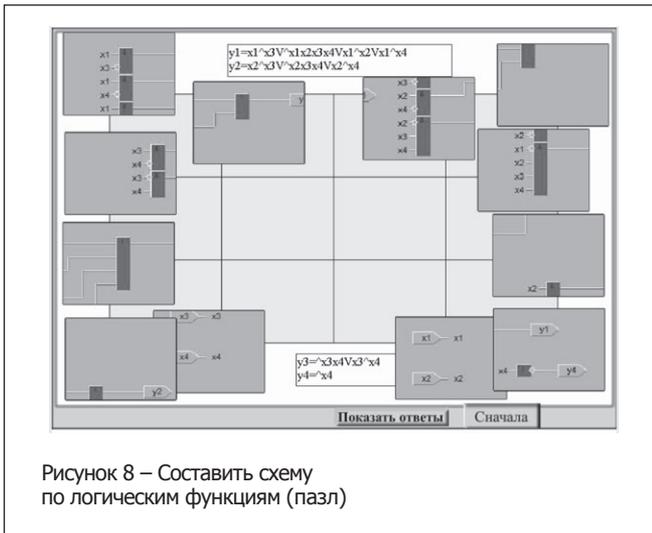


Рисунок 8 – Составить схему по логическим функциям (пазл)



Рисунок 9 – Просчитать значения на всех контактах схемы

Таны – это графические или текстовые элементы, которые при выполнении задания студент может перемещать с помощью мыши и поворачивать с помощью клавиш (стрелки влево и вправо). Таны делятся на графические и текстовые. Графические таны делятся на стандартные (с параметрами, поставляются в конструкторе), пользовательские (нарисованные авторами заданий средствами конструктора), произвольные рисунки (jpeg, png, разрешение не больше 742x500, размер не более 150 Кбайт). Автор задания может создавать собственные графические таны встроенными средствами конструктора (отрезками линий и фрагментами слайвов второго порядка). Текстовые таны создаются пользователем в конструкторе набором соответствующих текстов и выбором цвета текста и фона, а также шрифта и размера текста. В целях сокращения времени на создание заданий, обеспечена возможность использовать ранее созданные ресурсы: база заданий, база пользовательских танов (из всех заданий базы), библиотека пользовательских танов-рисунков (тематически сгруппированных).

Кроме танов имеются также следующие активные элементы заданий:

– перечисляемые поля – области, для которых автор задания указывает возможные варианты ответов, из которых

последовательными кликами студент выбирает правильный, по его мнению, ответ;

– области выделения – клики, в которых обозначают правильный ответ;

– области внесения – для правильного выполнения задания достаточно внести тан в любое место этой области;

– точки соединения – для правильного выполнения задания требуется кликнуть мышью в некоторые из них;

– таны, перечисляемые поля и поля ввода встречались в представленных выше заданиях.

Приведем примеры заданий, в которых используются еще не проиллюстрированные конструктивные элементы.

Эти задания использовались при изучении темы «Введение в ассемблер Intel 8086».

На рисунке 10 представлены области внесения: название регистра достаточно внести в нужную ячейку сетки.

На рисунке 11 представлено задание, использующее точки соединения, – необходимо соединить фрагменты программы с ответами, которые они получают для выходной переменной *r* для заданных значений входных переменных *x* и *y*.

На рисунке 12 представлено задание, где используются области выделения. Студент обязан кликнуть в тех местах программы, где совершена ошибка.



Рисунок 10 – Область внесения

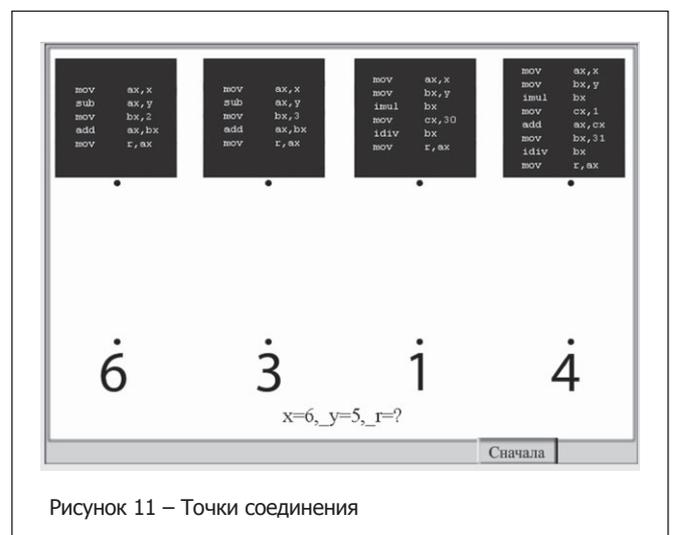


Рисунок 11 – Точки соединения

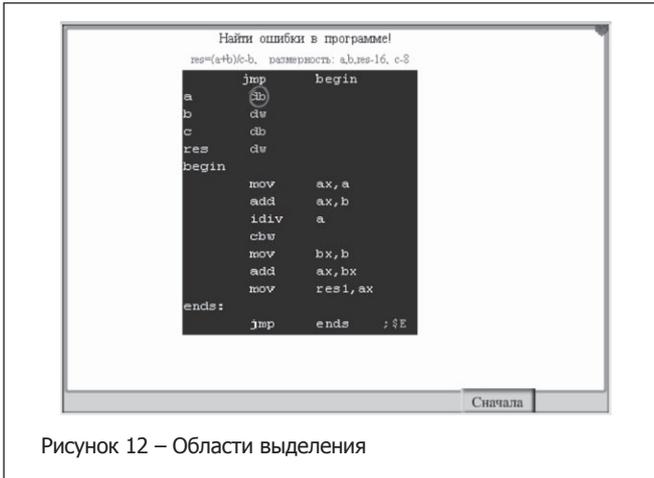


Рисунок 12 – Области выделения

Заключение

Описанные в данной работе задания активно используются в учебном процессе математического факультета Гомельского государственного университета им. Ф. Скорины и обеспечивают существенную интенсификацию и повышение качества обучения по соответствующим дисциплинам.

Задания, созданные в описанном в данной работе конструкторе заданий, использовались авторами в широком спектре применения от вуза до дошкольного обучения [7]. Как и предполагалось, задания в силу своей интерактивности и красочности привлекательны для обучаемых, а в силу мощности возможностей конструктора имеют высокую эффективность в обучении. Важными достоинствами конструктора

являются простота использования и существенное снижение трудоемкости создания заданий.

Литература:

1. Долинский, М.С. Использование новых информационных технологий при обучении проектированию цифровых систем и программированию / М.С. Долинский, М.А. Кугейко // Электроника инфо. – Минск, 2010. – № 4(73). – С. 10–13.
2. Долинский, М.С. Редактирование, симуляция и отладка аппаратного обеспечения с помощью HLCCAD / М.С. Долинский, И.В. Коршунов // Электроника инфо. – Минск, 2010. – № 6(75). – С. 22–26.
3. Долинский, М.С. Среда WINTER для редактирования, симуляции и отладки программного обеспечения мультимикропроцессорных систем / М.С. Долинский, И.В. Коршунов // Электроника инфо. – Минск, 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. – Минск, 2011. – № 3. – С. 53–57.
5. Долинский, М.С. Автоматический синтез микропрограммных автоматов по С-МПА программам / М.С. Долинский, И.В. Коршунов. – 2012. – № 2. – С. 97–100.
6. Долинский, М.С. Учебный микропроцессор TCPU и его применение / М.С. Долинский, И.В. Коршунов. – 2012. – № 3. – Принят к публикации.
7. Долинский, М.С. Конструирование интерактивных флеш-заданий на развитие мышления / М.С. Долинский, Ю.В. Решетько, М.А. Долинская // Информатизация образования. – Минск, 2011. – № 3(64). – С. 21–33.

ООО «Белпромэнергоэффект»

SvetaLED®

ЭНЕРГОСБЕРЕГАЮЩИЕ РЕШЕНИЯ ДЛЯ ВАШЕГО ПРЕДПРИЯТИЯ

УСТРОЙСТВА КОМПЕНСАЦИИ РЕАКТИВНОЙ МОЩНОСТИ – СВЕТОДИОДНОЕ ОСВЕЩЕНИЕ SVETALED: ОФИСНОЕ, ПРОМЫШЛЕННОЕ, УЛИЧНОЕ, СПЕЦИАЛЬНОЕ СВЕТОДИОДНЫЕ ЛАМПЫ E27, E14, MR 16, GU5,3, GU10

ГАРАНТИЯ ПРОИЗВОДИТЕЛЯ, СКИДКИ МИНИМАЛЬНЫЕ ЦЕНЫ

www.belpee.by
тел./факс: (017) 262 88 84
(029) 363 25 66

БелПлата тел. +375 17 287 85 66
 факс +375 17 287 85 65
 тел.моб. +375 29 684 43 09
 220068, г. Минск, ул. Некрасова, 114,
 оф. 238, 2 этаж, e-mail: info@belplata.by

Разработка и поставка печатных плат:
 любой класс точности, широкий спектр покрытий, изготовление образцов от 5 дней.

Поставка фотошаблонов

Поставка трафаретов:
 из нержавеющей стали и латуни.

Материалы для печатных плат:
 защитные маски, маркировочные краски, фоторезисты, паяльные пасты.

Поставка изделий из феррита:
 любые виды сердечников CI, EE, EEM, EP, EER, ETD, EC, EF, ED, EFD, EI, EPO, EPX, EPC и т.д.

Поставка электронных компонентов:
 STMicroelectronics, NXP Semiconductors, Vishay, Holtek Semiconductor.

www.belplata.by