

УДК 519.682.1: 681.142.2

А.П. СОБЧАК, Н.Д. СМИРНОВ

Национальный аэрокосмический университет им. Н.Е.Жуковского «ХАИ», Украина

АППАРАТНАЯ РЕАЛИЗАЦИЯ УСТРОЙСТВА СЖАТИЯ УПРАВЛЯЮЩЕЙ ИНФОРМАЦИИ НА ПРОГРАММИРУЕМЫХ ЛОГИЧЕСКИХ ИНТЕГРАЛЬНЫХ СХЕМАХ

В статье затрагиваются проблемы создания автоматизированных систем, вопросы, связанные с реализацией электронного компилятора, и один из вариантов их решения. В частности, рассматриваются некоторые аспекты построения одного из компонентов электронного компилятора – устройства сжатия управляющей информации. Приведен пример практической реализации такого устройства с использованием микросхем с программируемой логикой.

компилятор, программируемые логические интегральные схемы, сжатие, информация, автоматизация, проектирование

Введение

В условиях интенсивного развития ЭВМ и микропроцессоров проблема организации технологии подготовки и обработки информации в них относится к числу наиболее актуальных. В самом деле, с переходом от больших ЭВМ к мини- и микроЭВМ эта технология практически не претерпела существенных изменений. До настоящего времени не разработаны унифицированные правила, определяющие строгий порядок проектирования систем. Этим во многом объясняется то обстоятельство, что так и не было создано единых теорий проектирования автоматизированных систем управления (АСУ), измерения (АСИ), АСУ технологическим процессом (АСУ ТП) и т.д. [1].

Одним из способов ускорения обработки информации и повышения эффективности при моделировании является создание автоматизированных систем (АС), удовлетворяющих требованиям настройки баз данных и баз знаний на решение требуемой проблемы. Для проектирования автоматизированных систем предлагаются рекурсивные системы интерпретированных алгоритмических алгебр (Р-СИАА), используемые также для разработки технологии

синтеза рекурсивных автоматов с программируемой логикой (РАПЛ) [2].

Формулирование проблемы

Основными проблемами создания автоматизированных систем (АС) является построение математической модели и синтез РАПЛ для решения задач управления. При проектировании РАПЛ как символического процессора осуществляется синтез грамматик, а также создание операционной и управляющей структур, реализующих следующие функции:

- электронная компиляция входных проблемно-ориентированных языков для АС;
- автоматизация проектирования алгоритмов и программ для базовой ПЭВМ АС;
- сжатие управляющей информации и супермодульное проектирование программ.

Сжатие управляющей информации имеет большое значение при создании АС, так как позволяет сохранить объем требуемых ресурсов либо выполнить на ЭВМ с ограниченными ресурсами гораздо больший объем работ, чем это возможно. Примерами сжатия информации являются программы архивации файлов. В большинстве случаев эффектив-

ность сжатия достигается за счет рационального кодирования информации.

Решение проблемы

Одним из способов решения проблемы создания аппаратно реализованного устройства сжатия управляющей информации, как части электронного компилятора предлагается применение новой элементной базы, т.е. программируемых логических интегральных схем (ПЛИС) [3].

Программируемые логические интегральные схемы становятся в последнее время все более распространенной и привычной элементной базой для разработчиков цифровых устройств. Последние годы характеризуются резким ростом плотности упаковки элементов на кристалле, многие ведущие производители либо начали серийное производство, либо анонсировали ПЛИС с эквивалентной емкостью более 1 миллиона логических вентиляей.

Основными достоинствами ПЛИС [3] при использовании их в средствах обработки сигналов являются:

- высокое быстродействие;
- возможность реализации сложных параллельных алгоритмов;
- возможность программирования или изменения конфигурации непосредственно в системе;
- наличие средств системы автоматизированного проектирования (САПР), позволяющих провести полное моделирование системы;
- совместимость при переводе алгоритмов на уровне языков описания аппаратуры (VHDL, AHDL, Verilog и др.);

Система автоматизированного проектирования MAX+PLUS II предлагает полный спектр возможностей логического дизайна: разнообразные средства описания проектов с иерархической структурой, мощный логический синтез, компиляцию с заданными временными параметрами, разделение на час-

ти, функциональное и временное тестирование (симуляцию), тестирование нескольких связанных устройств, анализ временных параметров системы, автоматическую локализацию ошибок, а также программирование и верификацию устройств. В системе MAX+PLUS II можно как читать, так и записывать файлы на языке AHDL и файлы трассировки в формате EDIF, файлы на языках описания аппаратуры VerilogHDL и VHDL, а также схемные файлы OrCAD. Кроме того, система MAX+PLUS II читает файлы трассировки, созданные с помощью программного обеспечения (ПО) Xilinx, и записывает файлы задержек в формате SDF для удобства взаимодействия с пакетами, работающими с другими промышленными стандартами.

Особенностью современных ПЛИС является возможность тестирования узлов с помощью порта JTAG (B-scan), а также наличие внутреннего генератора (Osc) и схем управления последовательной конфигурацией [4].

Искомая структура [1] показана на рис. 1, где Rг1 – регистр для приема входной информации; Дш1 – дешифратор для расшифровки входных лексем; ССр – схема сравнения приоритетов операций *, + и скобок; Стек1 – для хранения скобок с и знаков операций +, *; Стек2 – для хранения операндов; Rг2 – регистр форматирования для выходной информации; Сч1 – счетчик для проверки баланса скобок с, d; Дш2 – дешифратор для расшифровки содержимого Сч1; Rг3 – регистр для хранения кодов переходов в алгоритме; шифратор – для кодирования этих переходов; К1, К2 – информационные шины; память – для записи сжатой информации; Сч3 – счетчик формирования кодов результатов генерации.

Представленное на рис. 1 устройство сжатия управляющей информации имеет следующую систему микроопераций и логических условий:

y1: Rг1 := К1 – занесение входной лексемы в Rг1 с информационной шины К1;

y2: ССр – запуск схемы сравнения;

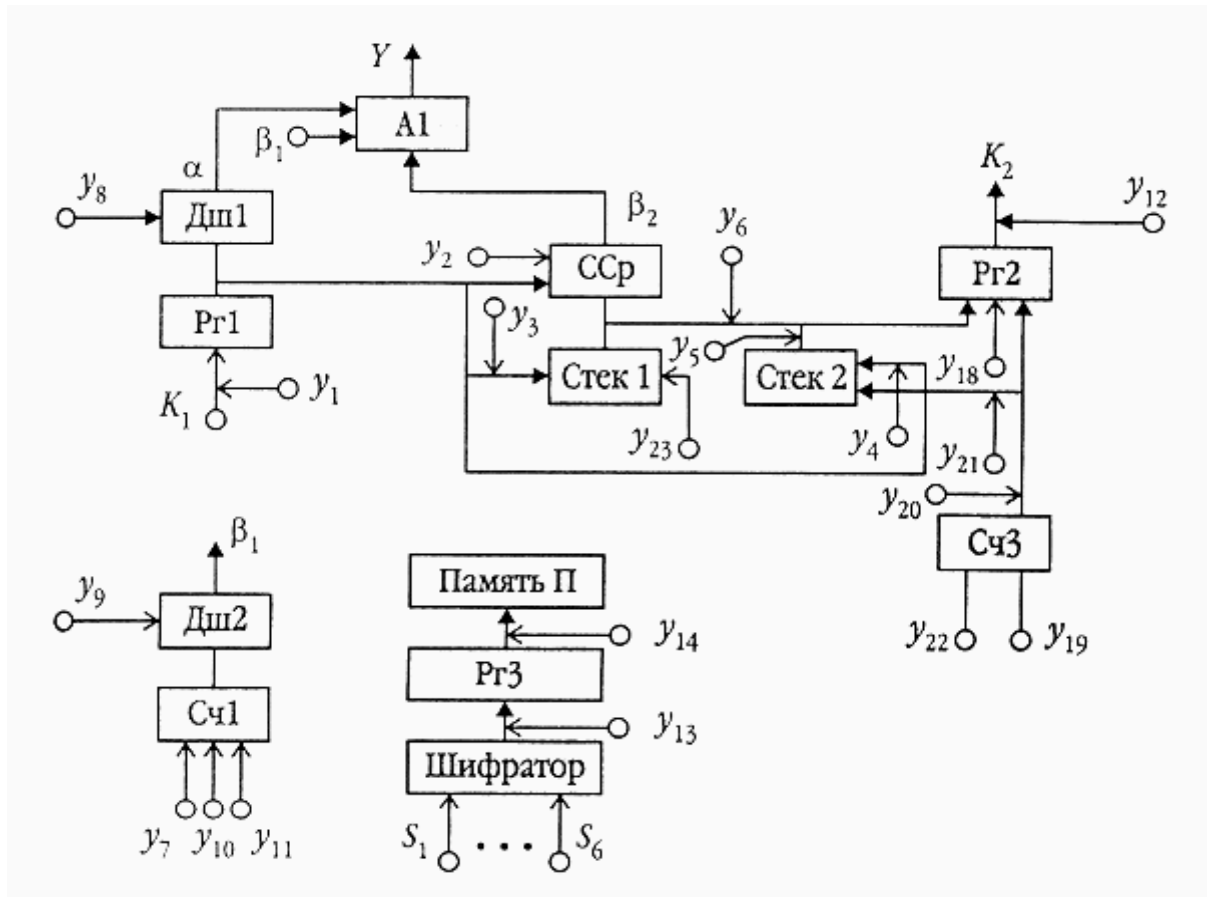


Рис. 1. Устройство сжатия информации

y3: Стек1 := Рг1 – перезапись информации из Рг1 в Стек1;

y4: Стек2 := Рг1 – перезапись информации из Рг1 в Стек1;

y5: Рг2 := Стек2 – перезапись информации из Стек2 в Рг2;

y6: Рг2 := Стек1 – перезапись информации из Стек1 в Рг2;

y7: Сч1 := Сч + 1 – занесение 1 в Сч1;

y8: Дш1 – запуск Дш1;

y9: Дш2 – запуск Дш2;

y10: Сч1 := Сч - 1 – считывание 1 из Сч1;

y11: Сч1 := 0 – обнуление Сч1;

y12: K2 := Рг2 – передача информации из Рг2 в информационную шину K2;

y13: Рг3 := Ш – формирование кода перехода в Рг3;

y14: П := Рг3 – запись в память кода перехода;

y18: формирование кода =;

y19: формирование кода Fi;

y20: Рг2 := Сч3;

y21: Стек2 := Сч3;

y22: Сч3 := 0;

y23: вытолкнуть левую скобку из стека 1.

Предлагаемая реализация данного устройства в среде MAX+PLUSII приведена на рис. 2, 3.

В качестве счетчиков в схеме использованы поставляемые в комплекте библиотек элементов MAX+PLUSII логические параметризованные модули (ЛПМ) LPM_COUNTER, которые занимают несколько большее количество ресурсов ПЛИС, чем описанное на основе базовых элементов, но в то же время имеют высокую степень кастомизации, хорошо отработаны на практике и широко используются.

То же самое касается регистров и модуля памяти. Они реализованы в виде ЛПМ LPM_SHIFTRREG и LPM_RAM_DQ.

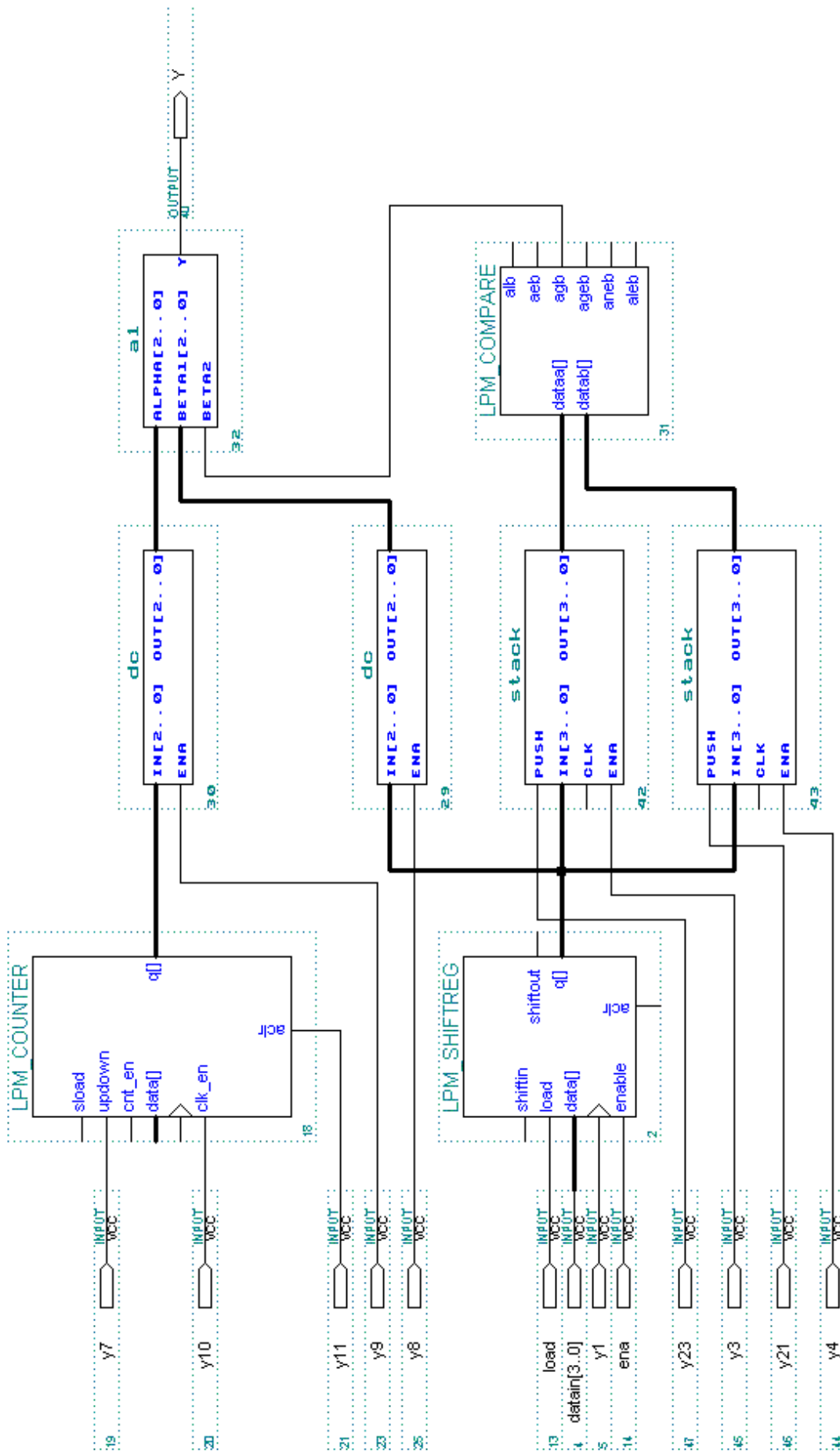


Рис. 2. Реализация устройства сжатия управляющей информации в среде MAX+PLUSII

Стеки Стек1 и Стек2 (рис. 1) в предлагаемой аппаратной модели реализованы на регистрах сдвига LPM_SHIFTRREG (рис. 2). Графическая модель стека в среде MAX+PLUSII приведена на рис. 4.

Модель дешифратора, который, как уже было отмечено выше, служит для расшифровки содержимого счетчика Сч1, предлагается реализовать на языке описания аппаратуры AHDL [5]. Текст программы приведен на рис. 5.

```

SUBDESIGN dc
(
  in[2..0], ena      : INPUT;
  out[2..0]         : OUTPUT;
)
BEGIN
  IF ena THEN
    TABLE
      in[] => out[];
      H"00" => H"00";
      H"01" => H"00";
      H"02" => H"00";
      H"03" => H"01";
      H"04" => H"00";
      H"05" => H"02";
      H"06" => H"00";
      H"07" => H"00";
    END TABLE;
  END IF;
END;

```

Рис. 5. Листинг модели дешифратора

Состояния выхода с кодами «00» соответствуют правильному балансу скобок входящего выражения.

Состояния «01», «02» могут соответствовать состояниям счетчика, когда открывающая скобка уже поступила, а закрывающая – еще нет, либо ошибочному состоянию, когда закрывающая скобка появилась перед открывающей.

Заключение

Таким образом, предложена модель аппаратной реализации устройства сжатия управляющей информации, как часть электронного компилятора, что позволяет:

1. Улучшить эксплуатационные характеристики действующих ЭВМ за счет организации ЭК в виде сопроцессора, функционирующего в режиме оптимизации.

2. На 2 – 3 порядка повысить быстродействие процесса компиляции за счет микропрограммной реализации.

3. Организовать процесс компиляции в целях удобства отладки программ в режимах трансляции и интерпретации за счет соответствующего функционирования аппаратных средств ЭК.

4. Реализовать ЭК в виде единой универсальной структуры, параметрически настраиваемой на режим генерации объектного кода, исходя из множества языков программирования.

5. Производить распараллеливание программ в процессе компиляции.

Литература

1. Жихарев В.Я., Илюшко В.М., Чумаченко И.В. Проектирование электронных компиляторов: Монография. – Х.: Факт, 1999. – 88 с.
2. Жихарев В.Я. Обобщенная модель символического процессора // Открытые информационные и компьютерные технологии. Вып. 1. – Х.: ХАИ. – 1998. – С. 138 – 144.
3. Угрюмов Е.П. Цифровая схемотехника. – СПб.: БХВ-Петербург, 2001. – 528 с.
4. Стешенко В.Б. ПЛИС фирмы ALTERA: проектирование устройств обработки сигналов. – М.: ДОДЕКА, 2000. – 128 с.
5. Антонов А.П. Язык описания цифровых устройств AlteraHDL: Практический курс. – М.: ИП «Радиософт», 2001. – 224 с.

Поступила в редакцию 3.08.2005

Рецензент: д-р техн. наук, проф. В.В. Печенин, Национальный аэрокосмический университет им. Н.Е. Жуковского "ХАИ", Харьков.