

МОДЕЛИ АРХИТЕКТУРЫ MISC ПРОЦЕССОРА

ЛОБОДА В.Г., ЕЛЬЧАНИНОВ Д.Б., ЦУКАНОВ В.Ю.

Предлагаются и исследуются структурные, программные и графовые L – модели процессора с перестраиваемой архитектурой типа MISC для общего и специализированного применения.

1. Введение

Известны два типа цифровых автоматов – с программируемой процедурой (ЦАПП) и с программируемой структурой (ЦАПС) [1].

Принцип программирования процедуры недостаточно гибок, так как ЦАПП имеет жёсткую структуру и воздействовать на такой автомат можно, лишь задавая его исходное состояние и последовательность входных сигналов. ЦАПП часто не позволяет реализовать одновременно высокую точность вычислений и высокую скорость работы. Принцип программирования процедуры исключает адаптацию автоматов к изменяющимся условиям работы.

При программировании структуры ЦАПС в исходном состоянии остаётся не полностью синтезированным и синтезируется окончательно программным путём под воздействием входных управляющих сигналов в зависимости от условий работы и решаемой задачи. ЦАПС более гибкий по сравнению с ЦАПП. Очевидно, должен существовать автомат с программируемой процедурой и структурой-ЦАПС. ЦАПС может быть принят в качестве базового процессора для построения ЭВМ и вычислительных систем. Такой процессор, обладающий открытой архитектурой, предложено назвать Multipurpose Instruction Set Computer – MISC [2].

В [3] была представлена и предварительно исследована на теоретическом уровне модель ЦАПС. Данная работа продолжает исследования ЦАПС со структурой типа MISC с применением принципа двухуровневого программирования настройки [1].

2. Элементная база структурного моделирования

Логическая структура MISC состоит из двух частей [2] – основной (host) RISC и управляющей памяти (например, Flash). Host – процессор выполняет несколько десятков простых команд, а другие команды, не принадлежащие к числу простых, преобразуются в адрес соответствующей микропрограммы Flash. Host часть – одноуровневый RISC-процессор с глобальным микропрограммированием задачи (после компиляции команд высокого уровня), а система (Host + Flash) – двухуровневый CISC-процессор с программированием (путём «склейки» готовых микропрограмм из Flash памяти).

Следовательно, для построения структурного эмулятора архитектуры MISC, в частности host – ядра, необходимо использовать разрядно-модульные секции микропроцессорных наборов [4].

Анализ функциональных возможностей микропроцессорных секций целесообразно проводить с позиции их представления в виде абстрактных автоматов [5].

Условия универсальности структуры любой микропроцессорной секции должны быть следующие:

$$1) \quad x_i \geq 2; \quad y_i \geq 2; \quad Q_1 \geq 2, \quad (1)$$

где x_i – операнд, принимаемый по i -му каналу связи; y_j – выходной результат, отмеченный на j -м канале связи; Q_1 – множество состояний секции (регистров).

$$\begin{aligned} 2) \quad & Q(l+1) = S_1(Q_1, x_i, x_0); \\ & Q(l+1) = S_2(Q_1, Q_2, x_0); \\ & Q(l+1) = S_3(x_1, x_2, x_0); \\ & y_j = \lambda_1(Q_1, x_i, x_0); \\ & y_j = \lambda_2(Q_1, Q_2, x_0); \\ & y_i = \lambda_3(x_1, x_2, x_0); \end{aligned} \quad (2)$$

здесь x_0 – буква управляющего слова, поступающая по нулевому каналу; $Q(l+1)$ – функция переходов, определяющая выполнение некоторой операции в секции.

3) Должны выполняться микрооперации алгебраического сложения, инкремента, сдвига (все виды), конъюнкции, дизъюнкции.

4) Такие микрооперации должны выполняться над содержимым любых источников информации в соответствии с полным графом коммутации дз.

Если выполняются все четыре условия, то $K_1 = K_2 = K_3 = K_4 = 1$. Тогда структурная полнота (степень универсальности) определяется как

$$K_u = \sum_{i=1}^4 K_i / \sum_{i=1}^4 K_{i\max} = 1. \quad (3)$$

При невыполнении одного из условий соответствующий коэффициент K_i уменьшается, тогда $K_u < 1$.

Таким образом, структура секции операционного автомата должна содержать блоки выполнения операций, коммутации, внутренней памяти, расширения разрядности, формирования внутреннего рабочего цикла.

Универсальная секция управляющего автомата должна позволять реализовать полный перечень переходов в микропрограмме, строить управляющие автоматы любой разрядности и любой конфигурации.

Межмодульная связь, как правило, магистральная.

Из-за существующего коммерческого перекоса в сторону однокристалльных микропроцессоров в последние годы характеристики многокристалльных наборов БИС существенно не улучшались. Самым совершенным комплектом (из последних) остаются БИС K1804 [4], которые можно принять как инструмент первоначальной эмуляции MISC архитектуры. Комплект K1804 весьма обширен по своей ориентации, а коэффициенты структурной полноты K_u для всех типовых секций равны единице [4].

3. Структурные модели

Анализ этапов развития технологии средств вычислительной техники показывает, что на каждом таком этапе оптимальным оказывается процессор, близкий то к RISC, то к CISC архитектурам. Суще-

составляющих комбинационные схемы цифровых автоматов канала "КЗЦА - ОА". Этим также достигается открытость архитектуры процессора.

КАМК эмулируется микросхемой K1804ВУ4, которая позволяет реализовать типовые конструкции микропрограммы в микросхеме K1804ВМ1 - 16 разрядном процессорном элементе. Для организации условных переходов в микропрограммах в модель включён мультиплексор МХ, предназначенный для выбора необходимого признака условного перехода. Другие прямые и обратные связи в модели введены в соответствии с РТМ на комплект ИС типа K1804.

4. Программное моделирование

Для программной эмуляции полученных структурных моделей использовалась библиотека программных моделей элементов микропроцессорного набора K1804 [7], ОЗУ, Flash, Cache и вспомогательных фиктивных элементов, необходимых для организации взаимодействия БИС, СИС и МИС в составе общей программной модели MISC [6, 8]. В процессе реализации программной модели проверялась правильность установки связей и прохождения информации между узлами процессора; а также определялась необходимая полнота сформированных информационных и управляющих сигналов.

5. Моделирование L-сетями Петри

Моделирование MISC-процессора проводилось с помощью L-сетей Петри [9, 10], которые основаны на цветных [11] и нагруженных [12] сетях Петри.

В качестве примера модели MISC-процессора приведем описание процесса вычисления функции

$$f(x) = \frac{x}{\sin x}$$

с использованием 2-сети Петри.

Алгоритм вычисления функции $f(x)$ можно представить в виде сети Петри N_C , изображенной на рис. 3.

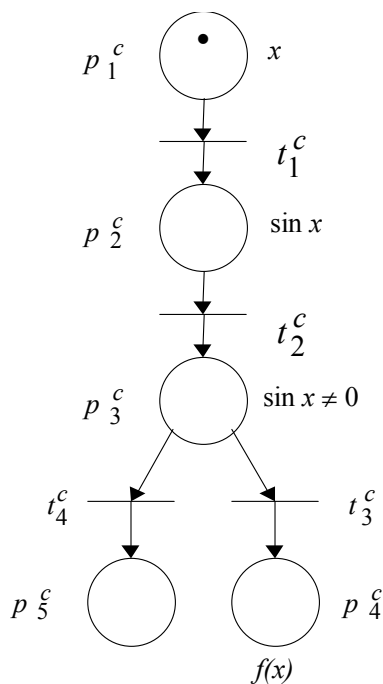


Рис. 3. Сеть Петри для вычисления $f(x)$

Таблица 1

Позиция	Метка в позиции
p_1^c	Начало вычисления функции $f(x)$
p_2^c	Вычислено значение $\sin x$
p_3^c	Проверено условие $\sin x \neq 0$
p_4^c	Конец вычисления функции $f(x)$
p_5^c	Вычисление прервано

Таблица 2

Переход	Запуск перехода
t_1^c	Вычисление значения $\sin x$
t_2^c	Проверка условия $\sin x \neq 0$
t_3^c	Вычисление значения функции $f(x)$
t_4^c	Прерывание вычислений из-за невыполнения условия $\sin x \neq 0$

Что означает наличие метки в позициях сети и какие действия соответствуют запуску переходов, показано в таблицах 1 и 2 соответственно.

Чтобы показать процесс вычисления функции $f(x)$, сопоставим метке в позиции p_1^c сеть Петри, изображенную на рис. 4. Позиции $p_1^e - p_4^e$ моделируют ячейки памяти MISC-процессора. В табл. 3 показано, что хранится в каждой ячейке. Сеть Петри N_0 содержит позиции, моделирующие ячейки памяти для хранения промежуточных результатов вычисления функции $f(x)$, и переходы, моделирующие необходимые операции для вычисления этой функции.

Дуги в сети Петри N_0 отсутствуют.

Далее метку, которой сопоставлена сеть Петри, будем называть макрометкой [9, 10]. Таким образом, сеть Петри, соответствующая макрометке, образует первый уровень, а сеть Петри N_C с макрометкой - второй уровень 2-сети Петри.

Затем сопоставим дуге сети Петри N_C из позиции p_1^c в переход t_1^c следующее условие-разрешение запуска перехода: позиция p_1^e содержит одну метку.

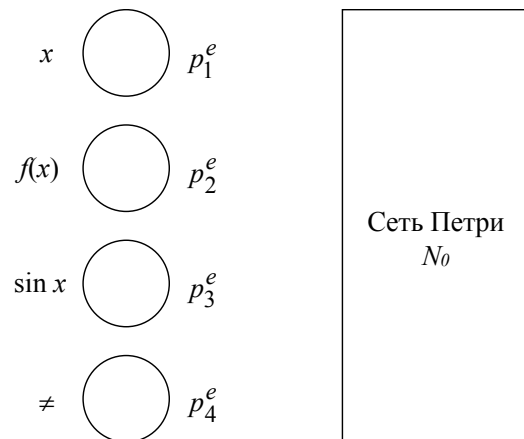


Рис. 4. Преобразованная сеть Петри

Таблица 3

Позиция	Что хранится
p_1^e	Значение x
p_2^e	Значение функции $f(x)$
p_3^e	Значение $\sin x$
p_4^e	результат проверки условия $\sin x \neq 0$

Дуге из перехода t_1^c в позицию p_2^c сопоставим следующее преобразование макрометки: изменение сопоставленной ей сети Петри на сеть, изображенную на рис. 5, где сеть Петри N_1 моделирует процесс вычисления функции $\sin x$.

Следовательно, появление метки в позиции p_1^e разрешает запуск перехода t_1^c , который удаляет макрометку из позиции p_1^c и добавляет ее в позицию p_2^c . Таким образом, моделируется запись значения x в соответствующую ячейку памяти и настройка MISC-процессора на вычисление функции $\sin x$. Переходы t_1^e и t_2^e моделируют соответственно начало и конец вычисления функции $\sin x$.

Далее сопоставим дуге сети Петри N_c из позиции p_2^c в переход t_2^c следующее условие-разрешение запуска перехода t_2^c : позиция p_3^c содержит одну метку. Дуге из перехода t_2^c в позицию p_3^c сопоставим следующее преобразование макрометки: изменение сопоставленной ей сети Петри на сеть, изображенную на рис. 6, где сеть Петри N_2 моделирует процесс проверки условия $\sin x \neq 0$.

Следовательно, появление метки в позиции p_3^c разрешает запуск перехода t_2^c , который удаляет макрометку из позиции p_2^c и добавляет ее в позицию p_3^c . Таким образом, моделируется запись значения $\sin x$ в соответствующую ячейку памяти и настройка MISC-процессора на проверку условия $\sin x \neq 0$. Переход t_1^e моделирует начало проверки условия, а

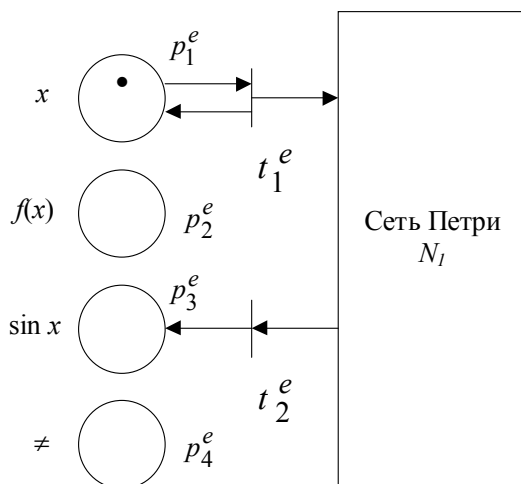


Рис. 5. Сеть Петри N_1

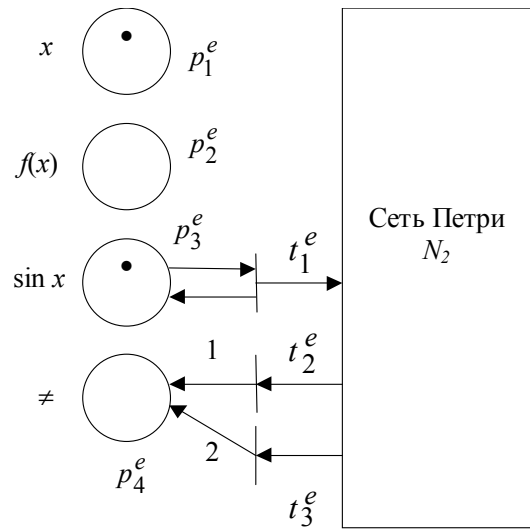


Рис. 6. Сеть Петри N_2

переходы t_2^e и t_3^e – конец проверки с результатом соответственно $\sin x \neq 0$ и $\sin x = 0$.

Затем сопоставим дуге сети Петри N_c из позиции p_3^c в переход t_3^c следующее условие разрешения запуска перехода t_3^c : позиция p_4^c содержит одну метку. Дуге из перехода t_3^c в позицию p_4^c сопоставим следующее преобразование макрометки: изменение сопоставленной ей сети Петри на сеть, изображенную на рис. 7, где сеть Петри N_3 моделирует процесс вычисления отношения $\frac{x}{\sin x}$.

Следовательно, появление одной метки в позиции p_4^c разрешает запуск перехода t_3^c , который удаляет макрометку из позиции p_3^c и добавляет ее в позицию p_4^c . Таким образом, моделируется запись проверки условия $\sin x \neq 0$ в соответствующую ячейку памяти и настройка MISC-процессора на вычисление отношения $\frac{x}{\sin x}$. Переходы t_1^e и t_2^e моделируют начало вычисления, а переход t_3^e – конец вычисления отношения $\frac{x}{\sin x}$.

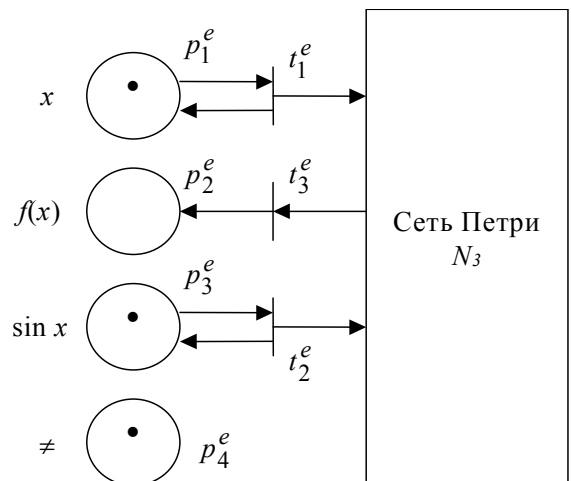


Рис. 7. Сеть Петри N_3

Наконец, сопоставим дуге сети Петри N_C из позиции p_3^c в переход t_4^c следующее условие – разрешение запуска перехода t_4^c : позиция p_4^e содержит две метки. Дуге из перехода t_4^c в позицию p_5^c сопоставим следующее преобразование макрометки: изменение сопоставленной ей сети Петри на сеть, изображенную на рис. 3.

Следовательно, появление двух меток в позиции p_4^e разрешает запуск перехода t_4^c , который удаляет макрометку из позиции p_3^c и добавляет ее в позицию p_5^c . Таким образом, моделируется запись проверки условия $\sin x \neq 0$ в соответствующую ячейку памяти и прерывание вычисления функции из-за невыполнения этого условия.

Предложенная модель отражает концепцию совместного проектирования аппаратных и программных средств (в англоязычной литературе известную как *hardware-software codesign*). При таком подходе модель программы и модель выполняющего ее процессора объединяются в единую модель, представленную в виде 2-сети Петри. Таким образом, можно проследить за изменением архитектуры MISC-процессора во время выполнения программы, что позволяет оптимальным образом адаптировать их друг к другу в целях одновременной реализации высокой скорости работы и высокой точности вычислений.

6. Заключение

В результате выполнения всех этапов моделирования MISC – архитектуры были сформулированы следующие общие рекомендации: MISC процессор может быть использован как центральный в ЭВМ общего назначения и как специализированный в составе вычислительных систем с производительностью не хуже современных коммерческих БИС (СБИС) при очевидной простоте комплексирования архитектуры на готовых схемах

Литература. 1. Каляев А.В. Микропроцессорные системы с программируемой архитектурой. М.: Радио и связь, 1984. 240 с. 2. Сизов К.А. Микропроцессор будущее-

го: RISC, CISC или MISC ? // Библиотека информационных технологий: Сб. статей. Вып. 1 / Под ред. Г.Р. Громова. М.: Наука, 1990. С. 118-124. 3. Бережная М.А., Лобода В.Г., Цуканов В.Ю. К вопросу проектирования структуры процессора // Радиозлектроника и информатика, 1998, №2 (3) .С.120-124. 4. Комплект БИС К1804 в процессорах и контроллерах / В.М. Мещеряков, И.Е. Лобов, С.С. Глебов и др. М.: Радио и связь, 1990. 256 с. 5. МикроЭВМ: В 8 кн. : Практическое пособие / Под ред. Л.Н. Преснухина. Книга 4. Управляющие системы "ЭНЦ" / Ю.Е. Чичерин. М.: Высш. шк., 1988. 192 с. 6. Иванников А.Д. Моделирование микропроцессорных систем. М.: Энергоатомиздат, 1990. 144 с. 7. Автоматизация проектирования аналого-цифровых устройств / Под ред. Э.И. Гитиса. М.: Энергоатомиздат, 1987. 184 с. 8. Собкин Б.Л. Автоматизация проектирования аналого-цифровых приборов на микропроцессорах. М.: Машиностроение, 1986. 128с. 9. Ельчанинов Д. Б. Моделирование иерархических структур L-сетями Петри // НТЖ. Автоматизация, телемеханизация и связь в нефтяной промышленности. М.: ВНИИОЭНГ, 1998. № 2. С. 7-8. 10. Ельчанинов Д. Б. Микропроцессорные иерархические системы управления на базе L-сетей Петри // Сб. статей "Актуальные проблемы современной науки в исследованиях молодых ученых г. Харькова". Х.: АО "Бизнес Информ", 1998. С. 20-23. 11. Технология системного моделирования / Е. Ф. Аврамчук, А. А. Вавилов, С. В. Емельянов и др. М.: Машиностроение; Берлин: Техник, 1988. 520с. 12. Слепцов А. И., Юрасов А. А. Автоматизация проектирования управляющих систем гибких автоматизированных производств. К.: Техніка, 1986. 110 с.

Поступила в редколлегия 17.05.99

Рецензент: д-р техн. наук Руденко О.Г.

Лобода Виталий Гаврилович, канд. техн. наук, профессор. Научные интересы: коммутация вычислительных структур. Увлечения: мотоцикл. Адрес: Украина, 310144, Харьков, ул. Ком. Уборевича, 12, кв. 55, тел. 65-73-65.

Ельчанинов Дмитрий Борисович, аспирант ХТУРЭ. Научные интересы: структурированные системы Петри. Увлечения: пчеловодство. Адрес: Украина, 310096, Харьков, пр. Героев Сталинграда, 189, кв. 153.

Цуканов Виталий Юрьевич, аспирант ХТУРЭ. Научные интересы: алгоритмическое и программное обеспечение функционально-ориентированных процессоров. Увлечения: автомобиль. Адрес: 310111, Украина, Харьков, ул. Познанская, 2, кв. 67, тел 10-42-63.