

## **О ВОЗМОЖНОСТЯХ РЕАЛИЗАЦИИ ПРИНЦИПА "ОТ АЛГОРИТМА – К СТРУКТУРЕ" В СИСТЕМАХ АВТОМАТИЗАЦИИ ПРОЕКТИРОВАНИЯ**

к.т.н. О.И. Богатов, А.В. Низовцев, Е.В. Кондратьева,  
В.Н. Кондратьев, И.И. Прокопенко

*Рассматривается методика проектирования цифровых систем, в основу которой положена новая математическая модель элемента вычислительного процесса. На примере конкретной задачи рассматривается автоматизированное построение функциональной схемы вычислительных устройств, реализующих решение данной задачи. Сущность задачи заключается в нахождении суммы по mod 2 многоразрядного двоичного слова (для определенности разрядность слова принята равной 36). Выбор именно такой достаточно простой задачи объясняется тем, что на ее примере можно общедоступными средствами показать все характерные черты общей методики. Одно из полученных технических решений представляет собой конвейерную структуру, позволяющую достигнуть максимального быстродействия, однако имеющую сравнительно большую аппаратную избыточность в части числа функциональных элементов. В другом техническом решении указанная выше аппаратная избыточность сведена к минимуму, позволяя обойтись всего лишь одним функциональным элементом, хотя при этом резко возрастает время реализации всей задачи. Показано, каким способом можно получить все промежуточные решения, в том числе и заданное.*

Аппаратная реализация алгоритмов позволяет получить наиболее оптимальные технические решения, по крайней мере, в части производительности [2]. Конструирование сложных устройств по принципу "от алгоритма – к структуре" кажется вполне естественным. Действительно, если алгоритм, который должен быть схемно реализован, может быть представлен в виде последовательности конечного числа фрагментов, каждый из которых может быть схемно реализован соответствующим устройством, причем существует возможность подключать эти устройства в последовательности, заданной алгоритмом, и передавать соответственно результаты с выходов одних устройств на входы других в соответствии с алгоритмом, то уже можно говорить о схемной реализации алгоритма.

Подобная ситуация уже наблюдалась в первых компьютерах. Естественно, при такой организации общее время реализации алгоритма определяется временем последовательной реализации всех фрагментов, входящих в алгоритм. Эта ситуация изменилась примерно три десятилетия назад в связи с появлением суперкомпьютеров, когда стала возможной конвейерная и параллельная обработка данных, в результате чего начали интенсивно развиваться методы параллельного программирова-

ния. Однако практические задачи требуют дальнейшего увеличения производительности компьютеров [3].

Следующим шагом в этом направлении явилось появление программируемых логических интегральных схем (ПЛИС), а также прикладных программных пакетов для работы с ними, позволивших добиться больших успехов в определенных областях, в частности, в области обработки сигналов [4]. Несмотря на большие возможности прикладных пакетов для работы с ПЛИС, наиболее удачным из которых представляется MAX + PLUS, осуществить принцип “от алгоритма – к структуре” в полном объеме с его помощью не удается.

Для этой цели необходима математическая модель, сочетающая в себе как элементы алгоритма, так и устройства, его реализующего. Одним из требований к такой модели является наличие математического аппарата, позволяющего синтезировать сложные модели из элементарных в едином пространстве и времени. В [1] такая модель была названа ST-моделью. Если положение, смысл которого заключается в том, что все ST-модели определены в едином времени, становится вполне понятным уже на интуитивном уровне, то положение, что все ST-модели определены в едином пространстве, нуждается в некотором уточнении. В данном случае под этим подразумевается, что для всех ST-моделей, как введенных изначально, так и полученных в результате тождественных преобразований, всем их входным и выходным каналам сопоставлены некоторые числа из множества натуральных чисел, причем если выходные каналы одной ST-модели соединены с входными каналами других ST-моделей, то соединенным каналам должны соответствовать одни и те же числа. Введенные в [1] операторы пространственного сдвига с учетом соответствующих тождественных преобразований дают возможность узаконить это положение на формальном уровне.

Таким образом, взаимно однозначное соответствие множества всех входных и выходных каналов, представляющее собой единое пространство и являющееся одной из координат области определения всех ST-моделей, множеству натуральных чисел позволяет, как и в случае единого времени определения всех ST-моделей, в случае необходимости дать возможность продолжать до бесконечности область определения ST-моделей по пространственной координате. С другой стороны, если следовать определенным правилам, то достигается возможность получить схему соединений ST-моделей, соответствующих устройств, предварительно определенных в качестве элементарных, и в результате объединения каналов образующих одну ST-модель, соответствующую всему устройству, схемно реализующему заданный алгоритм полностью.

В соответствии с вышеизложенным на основании предварительно введенной модели в [1] была предложена методика автоматизированного определения структуры устройства, реализующего заданный алгоритм. Структура сводится к  $q$  выражениям:

$$Y_j = \bigcup_{k \in M_j} P^{a(k,j)} D^{b(k,j)} F_j [0 \times 0] (x_0, \dots, x_{l-1}, y_0, \dots, y_{r-1});$$

$$L[i](x_k) = L[\gamma(k) + i](X);$$

$$X = \bigcup_{k=0}^{l-1} \{P^{\gamma(k)}(x_k)\};$$

$$y_k = L[a(k,j) \times b(k,j)](Y_j); \quad 1 \leq j \leq q; \quad k \in M_j,$$

где  $x_k$  –  $l$  переменных ( $0 \leq k \leq l-1$ ), представляющих собой исходные данные алгоритма и  $r$  промежуточных переменных  $y_k$  ( $0 \leq k \leq q-1$ ), являющихся результатами вычисления определенных функций в любых фрагментах алгоритма, а  $M_j$  – множество индексов  $k$  переменных  $y_k$ , в которых значение  $y_k$  определяется фрагментом с идентификатором  $F_j$ .

В настоящей статье рассмотрены примеры некоторых схемных реализаций конкретного алгоритма одной из известных технических задач. Задача состоит в вычислении функции

$$z = \sum_{j=0}^{r-1} x_j \bmod 2. \quad (1)$$

Соответствующая ST-модель элемента, лежащая в основе вычислительной системы, реализующей (1), может быть представлена следующим образом:

– вычисляемая функция  $F: A^2 \rightarrow A^1$ , где  $F_1(x, y) = \bar{x}y \vee x\bar{y}$ ;

$$s_0 = t_0 = s_1 = t_1 = 0; \quad \sigma_0 = 2; \quad \tau_0 = 1; \quad \sigma_1 = \tau_1 = 1;$$

– входное биективное отображение  $\lambda_{0,0}: \{0, 1\} \rightarrow \{0, 1\} \times \{0\}$ ;

– выходное биективное отображение  $\eta_{0,0}: \{0\} \rightarrow \{0\} \times \{0\}$ ;  $\sigma = 2$ ;  $\tau = 1$ .

Пусть алгоритм вычисления (1) представлен выражениями:

$$y_1 = F_1(x_0, x_1); \quad y_2 = F_1(y_1, x_2); \quad y_3 = F_1(y_2, x_3)$$

и т.д. в соответствии с рекуррентным соотношением  $y_j = F_1(y_{j-1}, x_j)$ .

Переходя к соответствующей ST-модели, алгоритм может быть представлен следующим образом:

$$y_1 = F_1(x_0, x_1) = F_1(L[0,1](X)); \quad y_j = F_1(y_{j-1} \cup P^{j+1}L[j](X)). \quad (2)$$

Для  $j > 1$   $z = y_r$ . Если  $j = 2$ , то  $y_2 = F_1(y_1 \cup P^1L[2](X))$ .

$$\text{Обозначим} \quad y(r) = \bigcup_{j=1}^r P^{2(j-1)} y_j, \quad (3)$$

что соответствует значению  $\sigma = 2$ ,  $\tau = 0$ .

$$\text{Следовательно,} \quad P^{2(j-1)} y_j = L[2(j-1)](y(r)) \quad (4)$$

для всех  $1 \leq j \leq r$ .

Из (2) и (4) следует, что

$$P^{2(j-1)}y_j = P^{2(j-1)}F_1(y_{j-1} \cup P^{-j+1}L[j](X)),$$

откуда в соответствии с тождественными преобразованиями получим

$$P^{2(j-1)}y_j = F_1(P^{2(j-1)}y_{j-1} \cup P^{2(j-1)}(P^{-j+1}L[j](X))).$$

Из этого следует, что

$$P^{2(j-1)}y_j = F_1(P^2(P^{2(j-2)}y_{j-1}) \cup P^{2(j-1)}(P^{-j+1}L[j](X))).$$

Тогда с учетом (4) будем иметь

$$P^{2(j-1)}y_j = F_1(P^2L[j-2](y(r)) \cup P^{j-1}L[j](X)).$$

Исходя из (3) и правил тождественных преобразований, получим:

$$y(r) = F_1\left(P^2 \bigcup_{j=0}^{r-2} L[j](y(r)) \cup \bigcup_{j=1}^r P^{j-1} L[j](X)\right); \quad (5)$$

$$z = y_r = L[2(r-1)](y(r)). \quad (6)$$

При значении  $r = 35$  полученному выражению однозначно соответствует схема, представленная на рис. 1.

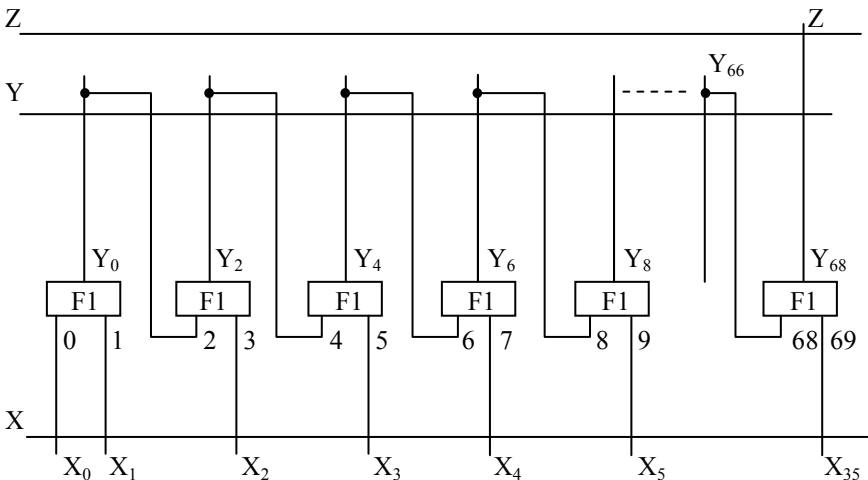


Рис. 1. Схема для  $z = y_{35} = L[2(35-1)](y(35))$

Рассмотрим другой вариант, в котором вместо (3) фигурирует

$$y(r) = \bigcup_{j=1}^r D^{j-1}y_j. \quad (7)$$

Для соответствующей модели элемента будет иметь место:  $\sigma = 0$ ;  $\tau = 1$ .

Тогда в соответствии с (2) вместо (4) получим

$$\mathbf{D}^{j-1} \mathbf{y}_j = \mathbf{L}[\mathbf{0}] \times [\mathbf{j}-1](\mathbf{y}(\mathbf{r})) \quad (8)$$

для всех  $1 \leq j \leq r$ .

Соответственно вместо (5) и (6) получим:

$$\mathbf{D}^{j-1} \mathbf{y}_j = \mathbf{F}_1 \left( \mathbf{D}^1 \bigcup_{j=0}^{r-1} \mathbf{D}^j (\mathbf{L}[\mathbf{0}] \times [\mathbf{j}-1](\mathbf{y}(\mathbf{r}))) \cup \bigcup_{j=1}^r \mathbf{D}^{j-1} \mathbf{P}^{-j+1} [\mathbf{j}](\mathbf{X}) \right); \quad (9)$$

$$\mathbf{z} = \mathbf{D}^1 \mathbf{L}[\mathbf{0}] \times [\mathbf{r}](\mathbf{y}(\mathbf{r})). \quad (10)$$

Анализируя выражение (9), невозможно не обратить внимание на то, что переменная величина  $\mathbf{U}(\mathbf{j}) = \bigcup_{j=1}^r \mathbf{D}^{j-1} \mathbf{P}^{-j+1} [\mathbf{j}](\mathbf{X})$  в этом выражении может быть представлена следующим образом:

$$\mathbf{U}(\mathbf{j}) = \begin{cases} \mathbf{X}, & \mathbf{j} = \mathbf{0}; \\ \mathbf{P}^{-1} \mathbf{D}^1 \mathbf{U}(\mathbf{j}-1), & \mathbf{j} > \mathbf{0}. \end{cases} \quad (11)$$

В этом случае

$$\mathbf{U}(\mathbf{1}) = \mathbf{P}^{-1} \mathbf{D}^1(\mathbf{X}); \quad \mathbf{U}(\mathbf{2}) = \mathbf{P}^{-2} \mathbf{D}^2(\mathbf{X}); \quad \mathbf{U}(\mathbf{k}) = \mathbf{P}^{-k} \mathbf{D}^k(\mathbf{X}),$$

т.е. техническая реализация величины  $\mathbf{U}(\mathbf{j})$  может быть выполнена в виде регистра сдвига. Действительно,

$$\mathbf{D}^k \mathbf{P}^{-k} \mathbf{L}[\mathbf{k}](\mathbf{X}) = \mathbf{D}^k \mathbf{L}[\mathbf{0}](\mathbf{P}^{-k}(\mathbf{X})) = \mathbf{L}[\mathbf{0}](\mathbf{P}^{-k} \mathbf{D}^k(\mathbf{X})) = \mathbf{L}[\mathbf{0}](\mathbf{U}(\mathbf{k})). \quad (12)$$

В таком случае в соответствии с (9) и (10) реализация системы (11) может быть представлена следующими уравнениями:

$$\mathbf{y} = \mathbf{F}_1 \left( \mathbf{D}^1 \bigcup_{j=1}^r [\mathbf{0}] \times [\mathbf{j}-1] \mathbf{y}(\mathbf{r}) \cup \mathbf{L}[\mathbf{0}] \mathbf{U} \right); \quad (13)$$

$$\mathbf{U} = \mathbf{X} \cup \mathbf{P}^{-1} \mathbf{D}^1 \mathbf{U}; \quad \mathbf{z} = \mathbf{D}^1 \mathbf{L}[\mathbf{0}] \times [\mathbf{r}]. \quad (14)$$

Соответствующая схема представлена на рис. 2. Другая техническая реализация (9), (10) может быть получена на основе мультиплексора, пропускающего на вход с индексом 1 соответствующей ST-модели устройства реализации алгоритма вычисления выражения (2) только те  $\mathbf{j}$ -е компоненты вектора  $\mathbf{X}$ , для которых  $\mathbf{j} \leq \mathbf{r} - 1$ . Такой мультиплексор должен иметь не менее  $\mathbf{r}$  входных однобитных каналов. Естественно реализация алгоритма на основе регистра сдвига гораздо проще, чем на основе мультиплексора. Таким образом, формальные тождественные преобразования (12) позволяют получить более компактную техническую реализацию алгоритма.

Сравнивая (5), (6) с (9), (10) (либо (13), (14) и (10)), нетрудно заметить, что реализация (5), (6) соответствует конвейерной обработке данных, в то время как (9), (10) (либо модификация (13), (14), (10)) представляет собой образец последовательной обработки.

В первом случае мы получаем систему с максимальным быстродействием, однако имеющую в составе все  $r$  элементов, реализующих функцию  $F_1(x,y) = \overline{xy} \vee \overline{\overline{xy}}$ . Во втором случае мы имеем систему, включающую в себя лишь один элемент, реализующий эту функцию, однако время реализации станет кратным значению  $r$ . Таким образом, задавая различные значения  $\sigma$  и  $\tau$ , можно получить полный спектр технических реализаций, отличающихся различными временными и аппаратурными характеристиками. При этом представляется целесообразным использовать иерархический принцип. Так, например, исходя из (5), приняв  $r = 3$  можно построить ST-модель нового элемента вычислительной системы, состоящую из 3 элементов, реализующих функцию  $F_1(x,y) = \overline{xy} \vee \overline{\overline{xy}}$ . В этом элементе новая функция представлена как  $F_2 : A^3 \rightarrow A^1$ , а  $F_2(x_0, x_1, x_2) = (x_0 + x_1 + x_2) \bmod 2$ .

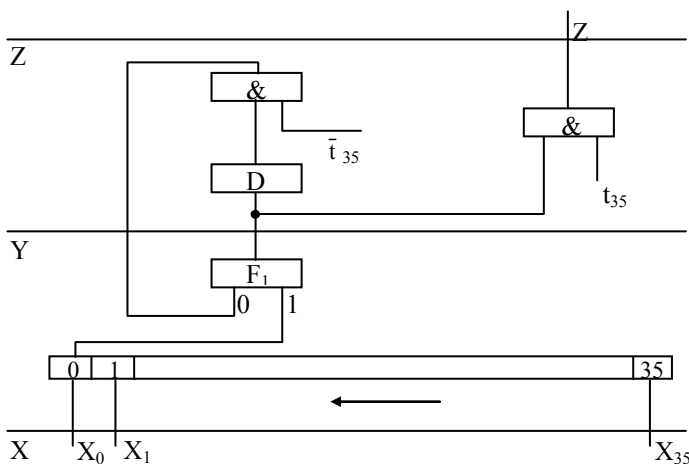


Рис. 2. Техническая реализация выражений (13) и (14)

Естественно, в этом случае будет иметь место  $\sigma = 6$ , а значение  $\tau$  останется неизменным:  $\tau = 1$ . Если взять далее  $r = 4$ , то можно построить еще одну ST-модель элемента вычислительной системы, уже состоящую из 16 элементов реализующих функцию  $F_1(x,y) = \overline{xy} \vee \overline{\overline{xy}}$ , состоящую из 12 элементов для нахождения сумм по модулю 2 переменных  $z_1 = F_2(x_0, x_1, x_2)$ ;  $z_1 = F_2(x_3, x_4, x_5)$ ;  $z_2 = F_2(x_6, x_7, x_8)$  и  $z_3 = F_2(x_9, x_{10}, x_{11})$  соответственно, а также 4 элемента для реализации функции  $z = F_3(z_0, z_1, z_2, z_3) = \sum_{i=0}^{i=3} z_i$ , в целом реализующих функцию:

$$F_4 \rightarrow A^{12} \rightarrow A^1, \quad z^{(0)} = F_4(x_0, x_1, x_2, \dots, x_{11}) = \sum_{j=0}^{11} x_j \bmod 2, \quad (15)$$

для которой в описании соответствующей **ST**-модели будет иметь место  $\sigma = 12$  и  $\tau = 1$ . Далее, чтобы получить реализацию (2) для  $r = 35$ , достаточно, кроме (15) взять еще два элемента, реализующие функцию  $F_4$ :

$$z^{(1)} = F_4(x_0, x_1, \dots, x_{11}) = \sum_{j=12}^{23} x_j \bmod 2; \quad z^{(2)} = F_4(x_0, x_1, \dots, x_{11}) = \sum_{j=24}^{35} x_j \bmod 2$$

и один элемент  $F_2(z^{(0)}, z^{(1)}, z^{(2)})$ , т.е. в итоге 36 элементов, реализующих функцию  $F_1(x, y) = \overline{xy} \vee \overline{\overline{xy}}$ . Соответствующая техническая реализация по количеству элементов совпадает с технической реализацией, представленной (5), (6). Однако, переходя к конкретным техническим решениям, следует заметить, что решение, полученное в соответствии с иерархическим принципом, является более предпочтительным, так как суммарная временная задержка, зависящая от общего числа последовательно включенных элементов в схеме, в данном случае будет меньшей за счет большей степени параллелизма. Включая в методику, основанную на иерархическом принципе также и варианты, подобные (9), (10), или соответственно (13), (14), (10), можно получать технические решения со значительно меньшим числом элементов, реализующих функцию  $F_1(x, y) = \overline{xy} \vee \overline{\overline{xy}}$ , вплоть до заданного числа.

## ЛИТЕРАТУРА

1. Богатов О.И., Низовцев А.В., Кондратьева Е.В., Кондратьев В.Н., Прокопенко И.И. Предпосылки создания метода автоматизированного проектирования цифровых систем, основанного на пространственно-временной модели // Системы обработки информации. – Х.: НАНУ, ПАНМ, ХВУ. – 2002. – Вып. 4(20). – С. 116 – 123.
2. Мухомад Ю.Ф. Проектирование специализированных вычислителей. – Новосибирск: Наука, 1981. – 160 с.
3. Кун С. Матричные процессоры на СБИС. – М.: Мир, 1991. – 328 с.
4. Beacher. Next-Generation Tool Requirements. News & Views. Newsletter for Altera Customers / Third Quarter, August, – 1998. – P. 18 – 19.

Поступила 1.07.2002

**БОГАТОВ Олег Игоревич**, канд. техн. наук, ст. научн. сотр., начальник НИО научного центра при ХВУ. В 1983 окончил Киевское ВИРТУ. Область научных интересов – параллельная обработка информации, САПР.

**НИЗОВЦЕВ Андрей Валерьевич**, нач. лаб. ИВЦ ХВУ. В 1986 году окончил Харьковское ВВКИУ РВ. Область научных интересов – ассоциативный поиск неявной информации.

**КОНДРАТЬЕВА Елена Владимировна**, инженер ИВЦ ХВУ. В 1998 году окончила ХГПУ. Область научных интересов – системы автоматизированного проектирования.

**КОНДРАТЬЕВ Владимир Николаевич**, инженер ИВЦ ХВУ. В 1961 году окончил ХПИ. Область научных интересов – системы автоматизированного проектирования.

**ПРОКОПЕНКО Игорь Иванович**, нач. лаборатории ИВЦ ХВУ. В 1987 году окончил ПВУРЭ. Область научных интересов – аппаратно-программный контроль вычислительных комплексов.