

## МЕТОДИКА ПРОЕКТИРОВАНИЯ ОПТИМАЛЬНОЙ АРХИТЕКТУРЫ ПАМЯТИ АППАРАТНО-ПЕРЕСТРАИВАЕМЫХ МУЛЬТИПАРАЛЛЕЛЬНЫХ СПЕЦПРОЦЕССОРОВ

д.т.н., проф. Г.А. Поляков, Е.Г. Толстолужская

*Описывается методика формализованного проектирования оптимальной архитектуры памяти и синтеза схемы распределения данных в памяти мультипараллельных аппаратно-перестраиваемых спецпроцессоров, дается формализация и определяется семантика основных этапов методики.*

**Постановка проблемы.** Пути повышения эффективности спецпроцессоров (СП) систем реального масштаба времени (РМВ) являются применение методов параллельной обработки данных, с одной стороны, и переход от монофункциональных СП с “жесткой“ структурой к аппаратно-перестраиваемым спецпроцессорам (АПСП), обеспечивающим решение конкретного множества задач при варьируемых требованиях и ограничениях (время решения, тактовая частота, используемый аппаратный ресурс), с другой стороны. Существующие САПР не обеспечивают проектирование этого класса процессоров. Необходимость их применения в системах РМВ делает крайне актуальным создание системы автоматического проектирования мультипараллельных аппаратно-перестраиваемых спецпроцессоров. Одной из наиболее сложных задач при этом является задача автоматического проектирования оптимальной архитектуры памяти.

**Анализ последних достижений и публикаций.** Анализ результатов международных симпозиумов по САПР последних лет [1, 2] выявил тенденции:

- сокращение субъективной роли человека и расширение сферы автоматического решения задач проектирования с целью решения проблем T2M и SoC;
- придание особого внимания автоматизации проектирования наиболее сложного класса устройств – заказным СБИС (ASIC), при этом, в основном, проектируются монофункциональные ASIC;
- известные САПР пока не поддерживают проектирование мультипараллельных аппаратно-перестраиваемых спецпроцессоров, предназначенных для реализации в РМВ заданного множества алгоритмов.

**Формулирование целей статьи.** В статье излагается методика реше-

ния одной из наиболее сложных задач проектирования параллельных цифровых устройств на системном уровне – синтеза оптимальной архитектуры памяти и схемы распределения данных для аппаратно-перестраиваемых мультипараллельных процессоров, определяется формализация основных этапов решения задачи, рассматривается семантика этапов.

**Исследования и результаты.** Исходными данными являются:

- множество мультипараллельных моделей решения задач  $P = \{P^v\}$ , представленное соответствующим множеством временных параллельных граф-схем (ВПГС)  $G^v(P^v, U^v, T^v)$ ,  $v = 1 \dots kz$ , [3 – 5];

- данные об интервале активности  $AI_j^v$  [5] каждого из операторов задач  $P_j^v \in P^v$ , реализующих операции записи данных в память или чтения данных из памяти (операторов типа = или \*),  $v = 1 \dots kz$ ,

$$AI_j^v = (t_j^{Hv} \dots t_j^{Hv} + \max_i (t_{il}^{Hv}) + t_{il}^{Ov}), \quad l \in M, \quad (1)$$

$t_j^{Hv}$  – момент начала реализации в модели  $v$ -й задачи оператора  $P_j^v$  записи/чтения, рассматриваемого как входная вершина подграфа – транзитивного замыкания  $MS_j^v$ , выходными вершинами которого являются операторы  $P_{il}^v$  типа =, “ближайшие” к  $P_j^v$  в каждом из  $m$  маршрутов ( $l = 1, 2, \dots, m$ ), порождаемых оператором  $P_j^v$ ;  $t_{il}^{Hv}$  – момент начала реализации  $i$ -го оператора  $P_{il}^H$ , представляющего собой выходную вершину  $l$ -го ( $l = 1, \dots, m$ ) маршрута из  $MS_j^v$ ;  $t_{il}^{Ov} = wt$  (время записи данных)-временная глубина оператора  $P_{il}^v \in MS_j^v$ , имеющего тип = (запись),  $P_{il}^v$  – выходной оператор  $P_i^v$   $l$ -го маршрута транзитивного графа  $MS_j^v$ ,  $M = 1, 2, \dots, m$ ;  $m = |M|$ ;

- данные о величинах циклов доступа к памяти для записи  $wt$  и чтения  $rt$  из памяти различных типов (регистров, RG; регистровых файлов, RGF; основной памяти, RAM; постоянной памяти, ROM).

Задачами методики являются:

- определение оптимального количества уровней и типы уровней памяти в иерархии памяти аппаратно-перестраиваемого спецпроцессора (одноуровневая регистровая память; одноуровневая основная память;

оперативная, постоянная; двухуровневая регистровая и основная память);

- определение количества модулей памяти различных типов (RG, RGF, RAM, ROM) в составе памяти АПСП;
- синтез оптимального распределения данных в различных модулях памяти, обеспечивающего реализуемость с помощью АПСП каждой из  $kz$  параллельных моделей задач;
- синтез множества таблиц физических адресов данных для каждой из временных моделей  $G^v(P^v, U^v, T^v)$ ,  $v = 1 \dots kz$ .

**Обобщенный алгоритм проектирования архитектуры памяти и схемы распределения данных.** Обобщенная блок-схема алгоритма проектирования архитектуры памяти и решения задачи оптимального распределения данных в памяти спецпроцессора, обеспечивающего возможность реализации синтезированных ранее параллельных моделей задач [4], представлена на рис. 1.

**Формализация основных этапов проектирования.** Основные этапы проектирования представлены символами 4, 5, 6, 8, 9, 10 (рис. 1).

**Символ 4** (рис. 1) строит для каждого оператора-данного  $P0_p^v \in P^v$  ( $p = 0, 1, \dots, d^v - 1$ ;  $v = 1 \dots kz$ ), расположенного на нулевом ярусе мультипараллельной модели  $G^v(P^v, U^v, T^v)$ , множества  $AIS_p^v$  интервалов активности путем объединения интервалов активности  $AI_j^v$  внешних для  $P0_p^v$  операторов  $P_j^v \in W_p$ , имеющих тип = или \* (запись/чтение), в соответствии со следующими соотношениями:

$$ALS_p^v = \bigcup_{j \in W_p^v} AI_j^v(P_j^v). \quad (2)$$

Это обеспечивает синтез набора  $AIS^v$  интервалов активности для операторов-данных  $v$ -й задачи ( $v = 1 \dots kz$ ):

$$ALS^v = \bigcup_{p=0}^{d^v-1} ALS_p^v \quad (3)$$



Рис. 1. Обобщенная блок-схема алгоритма проектирования архитектуры памяти и решения задачи оптимального распределения данных в памяти спецпроцессора

**Символ 5** (рис. 1) разделяет множество  $P0^v = \{P0_p^v\}$ ,  $p \in 0, 1, 2, \dots, d^v - 1$ , операторов-данных  $P0_p^v$  0-го временного яруса  $v$ -й модели на подмножества совместимости  $CS_\xi^v$  ( $\xi = 1 \dots rs^v$ ), удовлетво-

ряющие условиям а), б), в):

$$а) CS_{\xi}^v = \{P0_p^v\}, p \in 0, 1, 2, \dots, d^v - 1 \quad (4)$$

для  $p_1 \neq p_2$ ;  $p_1, p_2 \in 0, 1, 2, \dots, d^v - 1$  и для  $tp_1 = \text{type}(P0_{p_1}^v) \in TYP^v$  и  $tp_2 = \text{type}(P0_{p_2}^v) \in TYP^v$ , где  $TYP^v$  – множество типов операторов-данных  $v$ -й модели, и выполняется условие  $\text{type}(P0_{p_1}^v) = \text{type}(P0_{p_2}^v)$ ;

$$б) CS^v = \bigcup_{\xi=1}^{rs^v} CS_{\xi}^v; \quad (5)$$

$$в) CS_{\xi_1}^v \cap CS_{\xi_2}^v = \emptyset \text{ для } \xi_1 \neq \xi_2 \text{ и } \xi_1, \xi_2 \in 0, 1, \dots, rs^v. \quad (6)$$

Условие а) обеспечивает включение в каждое из подмножеств совместимости  $CS_{\xi}^v$  только операторов-данных  $v$ -й модели, имеющих одинаковый тип  $tp$ . Условие б) означает рассмотрение всех операторов-данных  $P0_p^v$  соответствующей  $v$ -й модели задачи. Условие в) означает, что интервалы активности любых двух операторов  $P0_{p_1}^v, P0_{p_2}^v$  любого множества совместимости  $CS_{\xi}^v$  ( $\xi = 1 \dots rs^v$ ) не пересекаются, что определяет их временную совместимость.

**Символ 6** (рис. 1) рассчитывает оптимальный состав типов памяти, обеспечивающий удовлетворение требований/ограничений по времени (стоимости/сложности), и определяет оптимальное количество модулей памяти каждого типа, необходимых для выполнения спецпроцессором каждой из  $kz$  мультипараллельных моделей задач.

### **Возможные варианты архитектуры памяти АПСП:**

а) одноуровневая регистровая память (модули памяти – регистры, регистровые файлы: RG, RGF).

б) одноуровневая основная память (модули памяти – RAM, ROM).

в) двухуровневая память (модули памяти – RG, RGF и RAM, ROM).

Введем следующие сокращения: память 1-го (регистрового уровня) – M1; компоненты памяти на регистровом уровне – RM (Register Memory) и RF (Register File); основная память (2-й уровень памяти) – BM (Basic Memory); компоненты памяти на 2-м уровне M2 типа RAM (Address Memory), типа ROM – (Output Memory). Величину цикла доступа к компонентам типа RAM и типа ROM будем обозначать CA и CO соответственно.

Реализация этапа включает следующие шаги:

а) постановка во взаимно однозначное соответствие каждому подмножеству  $CS_{\xi}^v \in CS^v$  совместимости  $v$ -й модели, имеющему конкретный тип  $tr \in TYP^v$ , компонента типа  $RM|RF$  памяти  $M1$  первого уровня – если для любого из операторов-данных  $P0_p^v$   $v$ -й параллельной модели отсутствует хотя бы одна такая пара смежных интервалов активности  $AI_r^v$  и  $AI_{r+1}^v$  с номерами  $r$  и  $r + 1$ , что для “интервала пассивности” между ними  $PI(p, p+1) = t_{r+1}^H - t_r^K$  и величин  $CR_{\xi} | CF_{\xi}$  циклов доступа к компонентам памяти  $RM_{\xi} | RF_{\xi}$  выполняется условие

$$PI(r, r+1) < CR_{\xi} | CF_{\xi}; \quad (7)$$

тем самым синтезируется множество  $M1^v$  компонентов типа  $RM_{\xi}^v$  или/и  $RF_{\xi}^v$  одноуровневой регистровой памяти, необходимых для выполнения  $v$ -й параллельной модели;

б) постановка во взаимно однозначное соответствие каждому подмножеству  $CS_{\xi}^v$  совместимости  $v$ -й модели, имеющему конкретный тип  $tr \in TYP^v$ , компонента памяти  $RAM|ROM$  – если для каждой пары смежных интервалов активности  $AI_r^v$  и  $AI_{r+1}^v$  с номерами  $r$  и  $r + 1$  любого из операторов-данных  $v$ -й параллельной модели выполняется условие

$$PI(r, r+1) < CR_{\xi} | CF_{\xi}; \quad (8)$$

тем самым синтезируется множество  $M2^v$  компонентов типа  $RAM_{\xi}^v$  или/и  $ROM_{\xi}^v$  одноуровневой основной памяти  $M2^v$ , необходимых для выполнения  $v$ -й параллельной модели;  $\xi = 1 \dots rs^v$ ;  $rs^v = \left| CS^v \right|$ ;

в) невыполнение для каждой  $v$ -й параллельной модели условий (7) и (8) эквивалентно необходимости разбиения множества совместимости  $CS^v$   $v$ -й модели на два подмножества  $CS1^v, CS2^v$ :

$$CS^v = CS1^v \cup CS2^v, \quad (9)$$

в  $CS1^v$  входят те подмножества  $CS_{\xi}^v$   $v$ -й модели, для которых выполня-

ется (7), во второе – подмножества  $CS_{\xi}^V \in CS^V$ , для которых имеет место (8). Множество  $CS1^V$  порождает для  $v$ -й модели множество  $M1^V$  компонентов регистровой памяти ( $RM_{\xi}^V | RF_{\xi}^V$ ) с соответствующими характеристиками (разрядность, количество регистров); множество  $CS2^V$  порождает множество  $M2^V$  компонентов основной памяти ( $RAM_{\xi}^V | ROM_{\xi}^V$ ) с их характеристиками (разрядность, количество ячеек для представления элементов соответствующего подмножества совместимости  $CS_{\xi}^V$ ). Тем самым синтезируется множество  $M1, 2^V$  компонентов двухуровневой памяти, необходимых для выполнения  $v$ -й параллельной модели;  $\xi = 1 \dots rs^V$ :

$$M1, 2^V = M1^V \cup M2^V. \quad (10)$$

Каждому компоненту  $RAM_{\xi}^V$  и/или  $ROM_{\xi}^V$  основной памяти  $M2^V$ , имеющему номер  $\xi = 1 \dots rs^V$  и разрядность  $bq_{\xi}$   $v$ -й параллельной модели, сопоставляется набор данных  $CS_{\xi}^V$  мощности  $cs_{\xi}^V = |CS_{\xi}^V|$ , которые интерпретируются как содержимое компонента  $RAM_{\xi}^V$  и/или  $ROM_{\xi}^V$  основной памяти  $M2^V$ .

Для каждого компонента  $RAM_{\xi}^V$ ,  $ROM_{\xi}^V$  с номером  $\xi$  ( $\xi = 1 \dots rs^V$ ) данным, входящим в соответствующий набор  $CS_{\xi}^V$ , присваиваются адреса  $adr_{\xi}^V = 0, 1, \dots, cs_{\xi}^V - 1$ , ( $cs_{\xi}^V = |CS_{\xi}^V|$ ), и формируется файл ВА (Basic Adress) базовых адресов данных для каждого  $\xi$ -го компонента памяти  $RAM_{\xi}^V$ ,  $ROM_{\xi}^V$   $v$ -й модели

$$BA_{\xi}^V = \{adr_{\xi}^V\}. \quad (11)$$

**Символы 8, 9, 10** (рис. 1) обеспечивают решение следующих трех задач:

- определение оптимального состава модулей памяти типов  $tr \in \{RM, RF, RAM, ROM\}$  и разрядности ( $bq$ ) в составе памяти  $M1,2$  СП,

поддерживающей реализацию (в режиме раздельной реализации моделей) любой из  $kz$  параллельных моделей;

- синтез схемы оптимального распределения данных в различных модулях  $M_{1,2}$  памяти АПСП, обеспечивающей реализуемость каждой из  $kz$  параллельных моделей задач;
- формирование множества файлов физических адресов, задающих оптимальное распределение данных  $kz$  моделей по различным компонентам общей памяти  $M_{1,2}$  АПСП.

Реализация этапа включает следующие шаги:

1. Представление множества  $M_2^v$  компонентов типа  $RAM_{\xi}^v$ ,  $ROM_{\xi}^v$  основной памяти для  $v$ -й ( $v \in 1 \dots kz$ ) параллельной модели объединением подмножеств компонентов памяти, типы которых принадлежат множеству типов  $tr \in TYP$  и разрядность  $bq_{\xi}$  которых соответствует заданным значениям  $bq \in Razr^v$  операторов-данных  $v$ -й модели

$$M_2^v = M_2^v(RAM) \cup M_2^v(ROM); \quad (12)$$

$$M_2^v(RAM) = \bigcup_{bq \in Razr^v} M_2^v(RAM(bq)) = \bigcup_{bq \in Razr^v} \left( \bigcup_{\xi=1}^{rs_{\xi}^v} RAM_{\xi}^v(bq) \right); \quad (13)$$

$$M_2^v(ROM) = \bigcup_{bq \in Razr^v} M_2^v(ROM(bq)) = \bigcup_{bq \in Razr^v} \left( \bigcup_{\xi=1}^{rs_{\xi}^v} ROM_{\xi}^v(bq) \right), \quad (14)$$

где  $bq$  – фиксированное значение разрядности данных (например,  $bq = 16|32|64$ ).

2. Определение за  $kz$  шагов оптимального количества модулей памяти данных различных типов  $tr$  фиксированной разрядности  $bq \in Razr$  и соответствующего количества модулей адресной памяти в иерархии памяти АПСП:

- **Выполнение первого шага** ( $v = 1$ ) описывается, в общем случае, для различных значений  $bq \in Razr^v$  разрядности данных первой модели ( $v = 1$ ) соотношением

$$M^1_{-СП} = \left( \bigcup_{bq} M^1(bq) \right) \cup \left( \bigcup_{bq} M^2(RAM, bq) \right) \cup \left( \bigcup_{bq} M^2(ROM, bq) \right) \quad (15)$$

и, следовательно, на первом шаге в иерархию памяти СП включаются



только компоненты памяти, поддерживающие реализацию первой задачи ( $v = 1$ ).

• **Выполнение произвольного  $v$ -го последующего шага** ( $v = 2 \dots kz$ ). Представим для последующих рассуждений (15) в следующем общем виде:

$$M^v\_СП = M1^v\_СП \cup M2^v\_СП(RAM) \cup M2^v\_СП(ROM), \quad (16)$$

определив составляющие правой части (16) следующим образом:

$$M1^v\_СП = \bigcup_{r=1}^v M1^r; \quad (17)$$

$$M2^v\_СП(RAM) = M2^{v-1}\_СП(RAM) \cup M2^v(RAM) \setminus M2^{v-1}\_СП(RAM); \quad (18)$$

$$M2^v\_СП(ROM) = M2^v(ROM) \setminus M2^{v-1}\_СП(ROM). \quad (19)$$

Как следует из (17) – (19), реализация произвольного  $v$ -го шага ( $v = 2 \dots kz$ ) включает следующие операции:

• дополнение множества компонентов памяти 1-го уровня, введенных за  $v - 1$  предшествующих шагов в интересах реализации любой из рассмотренных  $v - 1$  моделей, множеством компонентов памяти  $v$ -го уровня, необходимых для выполнения  $v$ -й модели;

• введение в иерархию памяти СП множества компонентов RAM|ROM, которые имеют одинаковую разрядность  $bq$  и могут использоваться при выполнении как любой из рассмотренных  $v - 1$  моделей, так и рассматриваемой  $v$ -й модели;

• дополнительный ввод в иерархию памяти СП подмножества компонентов RAM|ROM, которые необходимы для обеспечения выполнения  $v$ -й задачи и отсутствуют в сформированной за  $v - 1$  предшествующих шагов иерархии памяти СП.

Выполнение соотношений (17) – (19) для всех типов  $tp$  и всех значений разрядности  $bq$ , определяемых соотношением (18),

$$tp \in \bigcup_{v=1}^{kz} \left( \bigcup_{\rho=1}^{rs^v} \text{Typ}_{\rho}^v \right); \quad bq \in \bigcup_{v=1}^{kz} \left( \bigcup_{\rho=1}^{rs^v} \text{Razr}_{\rho}^v \right) \quad (20)$$

завершает на последнем шаге (при  $v = kz$ ) процесс проектирования оптимальной архитектуры памяти данных АПСП.

Синтез оптимального распределения данных в модулях памяти данных СП заключается в формировании для каждого компонента основной памяти данных RAM( $tp, bq$ ) и/или ROM( $tp, bq$ ),  $tp \in \text{Typ\_СП}$ ,

$bq \in \text{Razr\_СП}$ , набора  $DS(tp, bq)$  совместимых операторов-данных различных параллельных моделей путем включения в  $DS(tp, bq)$  максимально возможного количества подмножеств совместимости  $CS_{\xi}^V$ , имеющих совпадающие значения параметров  $(tp, bq)$  для различных моделей

$$DS(tp, bq) = \bigcup_{v \in NZ(tp, bq)} (CS_{\xi}^V(tp, bq)), \quad (21)$$

при этом  $NZ(tp, bq)$  – множество номеров задач, содержащих подмножества совместимости  $CS_{\xi}^V$  с одинаковыми значениями параметров  $(tp, bq)$ .

Синтез для каждого компонента памяти данных  $RAM_{\mu}(tp, bq)$  и/или  $ROM_{\mu}(tp, bq)$  адресного файла  $DRA_{\mu} = \left\{ dra_{\mu\lambda}^{v(\delta)} \right\}$  (Data Real Address), определяющего порядок доступа к конкретному данному  $dat_{\xi\lambda}^v$  требуемого подмножества совместимости  $CS_{\xi}^V(tp, bq)$   $v$ -й модели ( $\xi \in NZ_{\mu}(tp, bq)$ ), поддерживаемой  $\mu$ -м ( $\mu = 1, 2, \dots$ ) компонентом основной памяти.

Синтез выполняется за  $nz_{\mu} = |NZ_{\mu}(tp, bq)|$  шагов, шаг с номером  $\delta = 1, 2, \dots, nz_{\mu}$  включает:

- формирование реального адреса  $dra_{\mu}^{v(\delta)}$  начала множества данных  $dat_{\xi\lambda}^{v(\delta)} \in CS_{\xi}^{v(\delta)}$   $v$ -й модели в  $RAM_{\mu} / ROM_{\mu}$

$$dra_{\mu}^{v(\delta)} = dra_{\mu}^{v(\delta-1)} + cs_{\xi}^{v(\delta)}, \quad (22)$$

где  $dra_{\mu}^1 = 0$ ,  $cs_{\xi}^{v(\delta)} = |CS_{\xi}^{v(\delta)}|$ ,  $v(\delta) \in NZ_{\mu}(tp, bq)$  – номер модели, поставленный во взаимно однозначное соответствие номеру шага  $\delta$ ;

- формирование реального адреса  $dra_{\xi\lambda}^{v(\delta)}$  произвольного элемента  $dat_{\xi\lambda}^{v(\delta)}$  с номером  $\lambda \in CS_{\xi}^{v(\delta)}$  в компоненте  $RAM_{\mu} / ROM_{\mu}$ ,  $\lambda = 0, 1, 2, \dots, cs_{\xi}^{v(\delta)}$ :

$$DRA_{\mu} = \left\{ dra_{\mu\lambda}^{v(\delta)} \right\}. \quad (23)$$

Синтез адресного файла (23) для  $\mu$ -го компонента памяти

RAM<sub>μ</sub>/ROM<sub>σ</sub> заканчивается при  $\delta = nz_{\mu}$ . Формирование множества всех адресных файлов для компонентов основной памяти АПСП выполняется аналогичным образом.

#### **Выводы.**

1. Для широкого класса систем РМВ крайне актуальным является оперативное создание мультипараллельных аппаратно-перестраиваемых спецпроцессоров.

2. Существующие САПР ориентированы в основном на проектирование заказных СБИС и не поддерживают создание АПСП.

3. Разработанная методика обеспечивает полную формализацию синтеза архитектуры памяти и схемы распределения данных для АПСП и является основой для системы автоматического проектирования рассматриваемого класса спецпроцессоров.

#### **ЛИТЕРАТУРА**

1. *THE EDN System Design Series, Part I, II, 38<sup>th</sup> DAC-Design Automation Conference 2001.*
2. *41st Design Automation Conference June 7-11, 2004 San Diego Convention Center San Diego, Calif.*
3. Толстолужская Е.Г. Методика функционального проектирования мультипараллельных перестраиваемых спецпроцессоров // Системы обработки информации. – Х.: ХВУ. – 2004. – Вып. 1. – С. 213 – 218.
4. Толстолужская Е.Г. Методика формализованного синтеза мультипараллельных архитектурно-ориентированных моделей решения задач // Збірник наукових праць. – К.: Інститут проблем моделювання в енергетиці ім. Г.Є. Пухова. – 2003. – Вып. 22. – С. 206 – 215.
5. Поляков Г.А., Умрихин Ю.Д. Автоматизация проектирования сложных цифровых систем коммутации и управления. – М.: Радио и связь, 1988. – 304 с.

Поступила 29.07.2004

**ПОЛЯКОВ Геннадий Алексеевич**, доктор технических наук, профессор, академик Академии наук Прикладной Радиоэлектроники. Область научных интересов – адаптивные самоорганизующиеся технологии и системы с параллельной обработкой данных.

**ТОЛСТОЛУЖСКАЯ Елена Геннадиевна**, младший научный сотрудник ХУ ВС. Область научных интересов – автоматизация проектирования цифровых устройств.

---