

Разработка программных средств для систем цифровой обработки сигналов (ЦОС) [1] представляет собой гораздо более трудоемкий процесс, чем программирование однопроцессорных систем, что объясняется, с одной стороны, более сложной теоретической базой параллельных вычислений, а с другой стороны, отсутствием гибких и эффективных инструментальных средств параллельного программирования.

Модель мобильных вычислений в виде *параллельной виртуальной машины* (ПВМ), предложенная в [2], позволяет описывать вычислительный процесс в различных вычислительных системах, включая традиционные (фон неймановские) многопроцессорные системы и нетрадиционные системы с архитектурами потоков данных и запросов, имеющие значительный потенциал в области ускорения вычислений [4].

В основе концепции ПВМ предполагается описание алгоритма в самой общей - универсальной форме, не зависящей от механизмов организации вычислительного процесса в том или ином классе вычислительных систем. Универсальная форма вычислительного алгоритма [2] рассматривается как категория, определенная над объектами информационного пространства (ИП).

Введенные понятия необходимы в дальнейшем, так как, во-первых, нельзя понять, осмыслить и правильно использовать нечто: объект, процесс, явление, не поставив его в отношение с другими и не осознав его как элемент в разнообразии подобных. Во-вторых, нельзя правильно спланировать и произвести сложноорганизованную последовательность действий: подготовка, выработка и реализация решений, распределенных по многочисленным исполнителям, если предварительно не будет глубоко понята и понятийно освоена предметная область, в которой предстоят действия. В-третьих, нельзя говорить о законах управления в предметной области, имея дело с единичным, а не с общим, характеризующим целые совокупности подобных объектов.

В данной работе согласно предлагаемого здесь аксиоматического подхода на базе [2] и на основании предложенной в [3] методики вводятся следующие основные **объекты**: **O**, **XO** и **МК**. Причем, множество операций $O_i \in O$ включает в себя список простейших операций алгоритма: начальный блок, арифметический блок, логический блок, блок перехода, конечный блок.

Множество характера обмена $XO_i \in XO$ содержит в себе последовательный, параллельный, последовательно- параллельный и параллельно- последовательный обмен данными между простейшими операциями алгоритма. Множество команд $МК_m \in МК$ представляет совокупность команд микропроцессорного комплекта со следующими условными командами I, R, G, B, F, L, P, A, S, M, D, PP, K, AP, например, для общности рассмотрения, набор команд X86.

Между двумя множествами **A** и **PR** существует определенное отображение, определяемое выбираемым набором микросхем - chipset.

Определение 1. Под равенства фрагментов RO_j и RO_k микропрограммы **PR**: $RO_j = RO_k$ обработки информации подразумевается равенство длин фрагментов $|RO_j| = |RO_k|$ и их совпадение с точностью до команды $МК^{(j)}_1 = МК^{(k)}_1, \forall j=1, |RO_j|$.

Тогда получим семейство $(X_i)_{i \in I}$ подмножеств равных фрагментов множества **PR**, которое образует покрытие его подмножества **A**, если $A \subset \bigcup X_i$. В частности, если **X** – покрытие **PR**, то $\bigcup X_i = PR$. В результате получается разбиение множества **PR** как всякое его покрытие $(X_i)_{i \in I}$ такое, что: а) $X_i \cap X_j = \emptyset$, каково бы ни было $i \in I, j \in I, i \neq j$;

б) $X_i \cap X_k \neq \emptyset$ для каждой пары различных индексов (i, k) из **I** (подмножества попарно не пересекаются).

Теорема 1. Пусть $(X_i)_{i \in I}$ есть разбиение множества **PR** и существует отношение $R\{RO_j, RO_k\}$ между двумя общими элементами RO_j и RO_k из **PR**, удовлетворяющее условиям рефлексивности $R\{RO_j, RO_j\}$, симметричности $R\{RO_j, RO_k\}$ и $R\{RO_k, RO_j\}$ и транзитивности: отношение $R\{RO_j, RO_k\}$ и $R\{RO_k, RO_l\}$ влечет $R\{RO_j, RO_l\}$. Тогда всякое отношение **R**, удовлетворяющее вышеперечисленным условиям, является отношением эквивалентности в **PR**.

Определяемое этим отношением разбиение $(X_i)_{i \in I}$, рассматриваемое как подмножество множества **PR** является фактормножеством множества **PR** по отношению **R** и обозначается **PR/R**. Его элементы называются классами эквивалентности $a_i \in L$ по **R**, каждый из которых имеет свой порядок $q_i \in Q$. Таким образом имеем список классов эквивалентности: $1, 2, 3, \dots, 1, \dots, L$ и список порядка классов: $q_1, q_2, q_3, \dots, q_1, \dots, q_L$.

Вновь полученные множества **A**, **PR**, **L**, **Q** являются вспомогательными и вместе с тремя основными **O**, **X** и **МК** образуют шкалу **M** множеств, т.е. допускают любые попарные математические операции, которые приводят к появлению дополнительных вспомогательных множеств. Причем, задание некоторого числа элементов множеств шкалы, отношений между общими элементами этих множеств и отображений подмножеств некоторых из этих множеств в другие сводится в конечном итоге к заданию единственного элемента одного из множеств шкалы.

Теорема 2. На шкале M зададим известное число явно сформулированных свойств общего элемента множества M и пусть S – пересечение подмножеств множества M , определяемых этими свойствами. Тогда элемент $s \in S$ определяет в базе O, XO, MK структуру рода S , которая характеризуется схемой образования M , исходя из O, XO, MK , и свойствами, определяющими S .

Всем структурам одного и того же рода придается специальное название, которые составляют список структур уже известных: конвейерная, матричная, векторная, и предлагаемых автором: почти конвейерная, конвейерно-векторная, вроде конвейерно-векторная, вроде векторно-конвейерная, векторно-конвейерная, почти векторная на базе нечетких множеств. Свойства, определяющие S , являются **аксиомами** рассматриваемых структур как характерные высказывания о базисных множествах и родовых структурах в виде: $Dx_j = Dx_j (O, XO, MK, A, PR, S_1, \dots, S_l)$. Фактически аксиомы задают ограничения на объективную интерпретацию базисных множеств и родовых структур в виде определенных зависимостей между их элементами. При этом всякое предложение, являющееся следствием предложения « $s \in S$ », т.е. аксиом, определяющих T , относится к **теории структур рода S** .

Аксиома 1. Если множество XO определяется одним элементом XO_1 – последовательный обмен данными между операциями O , тогда на заданной элементной базе MK реализуется структура рода **Конвейер- S_K** : $Dx_K: \exists ((O_i, XO_1) \Rightarrow A^j, MK_m) \Rightarrow PR^j) \in S_K$.

Аксиома 2. Если множество XO определяется одним элементом XO_2 – параллельный обмен данными между операциями O , тогда на заданной элементной базе MK реализуется структура рода **Вектор - S_v** : $Dx_v: \exists ((O_i, XO_2) \Rightarrow A^j, MK_m) \Rightarrow PR^j) \in S_v$.

Аксиома 3. Если на множестве XO превалируют элементы XO_3 последовательного обмена данными между операциями O , тогда на заданной элементной базе MK реализуется структура рода **Конвейерно-векторная S_{KV}** : $Dx_{KV}: \exists ((O_i, XO_3) \Rightarrow A^j, MK_m) \Rightarrow PR^j) \in S_{KV}$.

Аксиома 4. Если на множестве XO превалируют элементы XO_4 параллельного обмена данными между операциями O , тогда на заданной элементной базе MK реализуется структура рода **Векторно-конвейерная- S_{VK}** : $Dx_{VK}: \exists ((O_i, XO_4) \Rightarrow A^j, MK_m) \Rightarrow PR^j) \in S_{VK}$.

Введением аксиом в структуру рода завершается задание ядра аксиоматической теории или, образно говоря, гена, с помощью которого в свернутом виде хранится все бесконечное тело теории. При концептуальном проектировании аксиомы играют роль математических прототипов программ, которые будучи реализованными в виде СУБД станут контролирующими целостность базы данных при операциях над данными: их вводе, изменении и удалении. Соответствующим образом разработанные программы будут проверять все содержимое базы, включая вводимый в данный момент элемент, на соответствие тем ограничениям, которые сформулированы в аксиомах.

Интерпретация букв структуры рода T выражениями естественного языка является лингвистической интерпретацией, с помощью которой задаются имена множеств, элементы которых будут задавать объектную или теретико-множественную интерпретацию.

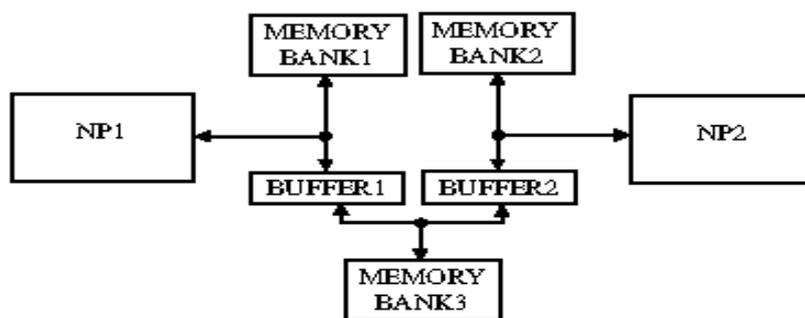


Рис. 1. Многопроцессорная конфигурация 2-го типа.

Автором предлагается реализовать полученные "гены" на базе современных отечественных нейропроцессоров NeuroMatrix NM6403 и NM6404[4], которые представляют собой высокопроизводительные DSP ориентированные RISC процессоры. В их состав входят два основных блока: 32-бит RISC ядро и запатентованный 64-бит VECTOR сопроцессор для поддержки операций над векторами с элементами переменной разрядности. Причем, последний процессор NM6404 совместим по системе команд с предыдущей версией NM6403 и содержит дополнительную память, которую можно использовать для организации многопроцессорных систем. На Рис. 1 показана одна из трех конфигураций такой системы.

Два имеющихся идентичных программируемых интерфейса для работы с внешней памятью различного типа и два коммуникационных порта, аппаратно совместимых с портами TMS320C4x, обеспечивают возможности построения мультипроцессорных систем, например, **конвейерно-векторной** структуры согласно вышеприведенной **аксиомы** на Рис.2

Нейропроцессор NM6403 работает с частотой 40 МГц, а NM6404– 133 МГц. Они изготовлены по технологии КМОП 0.25 мкм и располагаются в корпусе PQFP 256. Основу обоих нейропроцессоров составляет RISC – ядро с 5 - ти ступенчатым 32 - бит конвейером с

использованием 32- и 64-бит команды(обычно выполняется две операции в одной команде). Процессор имеет 2Мбит внутреннее ОЗУ с доступом к внутренней памяти соседа и два адресных генератора с адресным пространством 16 GB. Схема имеет два 64-бит программируемых интерфейса с SDRAM/SRAM/DRAM /Flash ROM разделяемой памятью. При этом обеспечивается 4 одновременных доступа к внутренней памяти.

Имеющиеся два коммуникационных порта осуществляют аппаратную совместимость с TMS320Cх . Кроме того, имеется JTAG – совместимый отладочный интерфейс. Система управления потребляемой мощностью стабилизирует работы всей схемы в целом. Одним из достоинств нейропроцессоров NM6403 и NM6404 является наличие встроенного дdVECTOR сопроцессора, который обеспечивает работу с длиной векторных операндов и результатов от 1- до 64-бит. Формат данных это целые числа, упакованные в 64-бит блоки и в форме слов переменной длины от1- до 64-бит каждое. Эти схемы обеспечивают поддержку векторно-матричных и матрично-матричных операций, при этом перезагрузка матрицы коэффициентов осуществляется за 16 тактов. Достоинствами данной схемы являются: а) использование ее для широкого класса задач от систем ЦОС до обработки видеоизображений, б) возможность эмуляции нейронных сетей и осуществление векторно-матричных вычислений для реализации сложных конвейерно-векторных и векторно-конвейерных мультимикропроцессорных систем, приведенных на следующем Рис. 3.

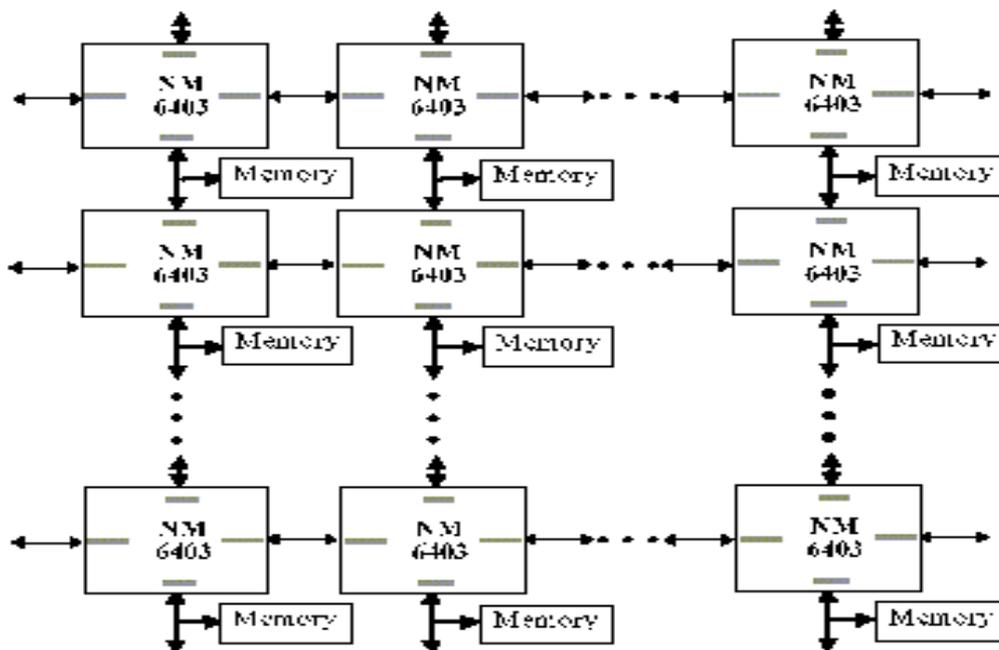


Рис. 2. Конвейерно-векторная структура на базе нейропроцессора NM6403.

Таким образом предложенная здесь теория структур- это формальная теория, позволяющая предельно строго описывать концептуальные модели, и, следовательно, предоставляющая потенциальную возможность алгоритмизировать их построение и использование в программных продуктах. Во-вторых, модели, заданные в аппарате теории структур, имеют форму аксиоматической теории и позволяют в свернутом виде хранить бесконечное тело теории, т.е. являются своего рода геном теории, из которого при желании можно вырастить все ее тело. При этом ген или ядро теории позволяет с помощью правил вывода осуществлять развертывание теории в разных направлениях, сообразуясь с задачами, решаемыми при проведении концептуальных исследований или при проектировании. В-третьих, этот аппарат позволяет эффективно описывать именно концептуальные, а не общепринятые динамические модели, предоставляя возможность задавать сеть взаимосвязанных понятий и выражать сложные производные понятия, проводя очень тонкие различия между ними.

Литература

1. Никитина Н.К. По лестнице концептуального развития // PC WEEK/ RE № 25, 30 ИЮНЯ, 1998, С. 41-42
2. Н. Бурбаки Элементы математики. Кн. III Общая топология. Основные структуры М.: Физматгиз, 1958с.
3. Ручкин В.Н. Проектирование и выбор специализированных средств обработки информации. М.: Московский государственный открытый университет, 1997. 120 с.: ил.
4. Виксне П.Е., Фомин Д.В., Черников В.М. Однокристалльный цифровой нейропроцессор с переменной разрядностью операндов// Известия Вузов, Приборостроение, 1996, т.39, №7

Ryazan Institute of the Moscow State Open University
390046, Ryazan, Kolkhoznaya St., 2, tel. (0912) 77-41-48

The working out the program means of the digital processing signals (DPS) [1] is much more difficult than the programming of single-processor systems. It can be explained, from the one hand, with the more complicated base of the parallel calculations, and, on the other hand, with the lack of flexible and effective instrumental means of the parallel programming.

In this work the author inputs the following basic objects in accordance with [2,3]: **O**, **XO**, and **MK**. The multitude of the operations $O_1 \sqsubset \mathbf{O}$ includes the list of the simplest operations of the algorithm: the initial bloc, the arithmetical bloc, the logical bloc, the transfer bloc and the ending bloc.

The multitude of the character changing $XO_1 \sqsubset \mathbf{XO}$ includes the consecutive, parallel, consecutive-parallel and parallel-consecutive change between the simplest operation of the algorithm.

The multitude of $MK_m \sqsubset \mathbf{MK}$ commands is a totality of the microprocessor kit commands with the following conventional commands: **I, R, G, B, F, L, P, A, S, M, D, PP, K, AP**, the set of X86 commands.

Between the **A** and **PR** multitudes there is a reflection which is found by the chipset.

Definition1. The equation of the RO_j and RO_k of the **PR** microprogram: $RO_j = RO_k$ of the information processing is the equation of the fragments lengths $|RO_j| = |RO_k|$ and their coincidence with the accuracy to the command $MK^{(j)}_1 = MK^{(k)}_1, \forall j=1, |RO_j|$.

Theorem 1. Let $(X_i)_{i \in I}$ be the division of the PR multitude and there is the relation $R\{RO_j, RO_k\}$ between two common elements RO_j и RO_k из PR, satisfied the reflexivity conditions $R\{RO_j, RO_j\}$, symmetry $R\{RO_j, RO_k\}$ and $R\{RO_k, RO_j\}$ and transitivity relation $R\{RO_j, RO_k\}$ и $R\{RO_k, RO_l\}$ appeals $R\{RO_j, RO_l\}$. Then any satisfying the latter conditions relation R , is the relation of the equivalence in **PR**.

Theorem 2. Let's give the known number of the formulated characteristics of the M multitude element n the **M** scale and let **S** be the crossing of the sub-multitudes of the M multitude, defined with these characteristics. Then the $s \in S$ element defines in the **O, XO, MK** base the *sort S structure*, which is characterized with the M formation scheme, proceeded from **O, XO, MK**, and properties defining **S**.

The special name is given to the all structures of the same sort. They combine the list of the already known structures: conveyor, matrix, vector, and suggested by the author: almost conveyor, conveyor-vector, somehow conveyor-vector, somehow vector-conveyor, vector-conveyor, almost vector on the vague multitudes base. The properties defining **S**, are the *axioms* of the structures looked upon as a distinctive statements about the basis multitudes and sort structures in the kind: $Dx_j = Dx_j(\mathbf{O}, \mathbf{XO}, \mathbf{MK}, \mathbf{A}, \mathbf{PR}, S_1, \dots, S_L)$. In fact, the axioms give the limits on the objective interpretation of the basis multitudes and sort structures in the kind of the certain dependence between their elements. With it any suggestion which is the the consequence of the suggestion « $s \in S$ », so that axioms defining **T**, related to the *sort S structure theory*.

Axiom 1. If the **XO** multitude is defined with one element of XO_1 – the consecutive exchange of the data between the operations **O**, then the structure of the **Conveyor-S_K:Dx_K: $\exists ((O_i, XO_1) \Rightarrow A^j, MK_m) \Rightarrow PR^j$) $\in S_K$ is realized with the help of the given chipset.**

Axiom 2. If the **XO** multitude is defined with one element of XO_1 – the parallel exchange of the data between the operations **O**, then the structure of the **Conveyor-S_K:Dx_K: $\exists ((O_i, XO_1) \Rightarrow A^j, MK_m) \Rightarrow PR^j$) $\in S_K$ is realized with the help of the given chipset.**

Axiom 3. If the XO_3 elements of the consequent data exchange between the **O** operations are predominated in the **XO** multitude, then the **Conveyor-vector structure of the sort S_{KB}:Dx_{KB}: $\exists ((O_i, XO_k) \Rightarrow A^j, MK_m) \Rightarrow PR^j$) $\in S_{KB}$ is realized on the given chipset **MK**.**

Axiom 4. If the XO_4 elements of the parallel data exchange between the **O** operations are predominated in the **XO** multitude, then the **Vector-conveyor of the sort S_{KB}:Dx_{KB}: $\exists ((O_i, XO_k) \Rightarrow A^j, MK_m) \Rightarrow PR^j$) $\in S_{KB}$ is realized on the given chipset **MK**.**

The author suggests to realize the received "genes" on the base of the contemporary Russian neuroprocessors NeuroMatrix NM6403 и NM6404[4], which are the high-productive DSP orientated RISC processors. They consist of two main blocs: 32-bit RISC "nuclear" and patent 64-bit VECTOR co-processor for the vector operation support with the elements of the variable discharge. The latter processor NM6404 can be combined by the command system with the former version NM6403 and contains the additional memory which can be used for the multiprocessor system organization.

REFERENCES

1. Nikitina N.K. Upstairs with the conceptual development. // PC WEEK/ RE № 25, JUNE, 30, 1998, P. 41-42
2. N. Bourbakee. The elements of mathematics. V. III. Common Topology. Basic Structures. Moscow, 1958c.
3. Ruchkin V.N. Projection and choice of specialized tools of information processing. M.: the Moscow state open university, 1997. 120 p.: silt.
4. Vixne P.E., Fomin D.F. Tchernikov V.M. Monocrystal digital neuroprocessor with the variable operand discharge//High School News, Device Designing, 1996, v.39, №7