

О МИКРОСТРУКТУРЕ ЭЛЕМЕНТАРНЫХ МАШИН
ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЫ

Э.В. Евреинов

§ I. В в е д е н и е

В работе [I] были сформулированы основные проблемы, которые необходимо решить для построения вычислительной системы высокой производительности.

Одной из этих проблем является вопрос о структуре вычислительной системы (ВС). Было показано, что наиболее перспективным является построение ВС из большого числа элементарных машин (ЭМ), представляющих собой универсальные электронные вычислительные машины (ЭВМ) с развитой системой операций, достаточно высокой производительностью, возможностью совместной (параллельной) работы с другими машинами и обмена информацией по соответствующим каналам связи, а также возможностью изменения системы операций в процессе решения задач. Вполне понятно, что ЭМ должны также обеспечивать высокую точность вычислений, высокую надежность работы и быть в то же время достаточно экономичными.

Отметим, что все эти требования, как и требования к любой ЭВМ, в настоящее время трудно четко определить. Это обстоятельство существенно затрудняет постановку задачи о разра-

ботке структуры элементарной машины. Преодолеть эти трудности можно двояким способом: либо провести работы по уточнению вышеприведенных требований, либо выработать новые требования, поддающиеся более четкой формулировке и в то же время обеспечивающие максимально возможное выполнение указанных требований на данном уровне вычислительной техники. По-видимому, второй способ является более реальным.

Действительно, требование универсальности, под которой понимается возможность решения разнообразных задач на одной и той же машине [2], влечет за собой введение логически полного набора операций и бесконечной памяти [3,4]. Если ввести логически полный набор операций принципиально нетрудно, то вполне понятно, что на практике приходится исходить из конечной, но достаточно большой памяти. В этом случае возникает необходимость в оценке объема памяти, что требует конкретного знания классов решаемых задач и методов их решения. Анализ существующих задач с этой точки зрения до сих пор не проводился. Тем более трудно предусмотреть заранее параметры тех задач, которые понадобятся решать через несколько лет. В этом отношении приходится ограничиться только некоторыми прогнозами.

Выполнение требований высокой производительности и высокой точности вычислений также в значительной мере обусловлены классами решаемых задач и алгоритмами.

Высокая производительность обусловлена соответствующим выбором операций, наиболее приспособленных для решения данного класса задач, скоростью выполнения операций, объемом памяти, применяемым алгоритмом решения. Все это в значительной мере влияет на структуру ЭМ. А так как в настоящее время нет сложившейся теории построения ЭВМ, то возникают дополнительные трудности, связанные с оценкой производительности ЭВМ. Можно оценить производительность только конкретных типов структур ЭВМ при решении известных классов задач [5,6,7].

Высокая точность вычислений связана с соответствующим выбором количества разрядов в слове. Потеря точности вычислений происходит из-за погрешностей при выполнении операций. Величина погрешности зависит от количества выполняемых над исходными числами операций и их характера (сложение, вычитание, умножение, деление и т.д.). Поэтому для оценки количества разрядов в слове так же требуется знание классов решаемых задач. В настоящее время сделана попытка оценить погрешность, возникающую при решении конкретных классов задач, на машинах с фик-

сированным количеством разрядов [8]. Для произвольного класса алгоритмов такая оценка не сделана. Таким образом, вопрос о количестве операций, выполняемых над исходным числом, и требуемой точности вычислений, остается открытым.

Итак, уточнение требований универсальности, производительности, точности вычислений влечет за собой необходимость конкретного знания классов решаемых задач. Ясно, что при построении ВС с расчетом на задачи, которые могут возникнуть уже после ее построения, невозможно исходить из конкретных задач. Поэтому приходится ограничиваться лишь прогнозом в отношении требуемых параметров машины. На практике данное противоречие преодолевается на основе так называемых "волевых" решений, что приводит к излишним затратам на разработку машин и к снижению их производительности.

Устранить указанное противоречие можно путем максимального сокращения разрыва между сроками решения данного класса задач и изготовления соответствующей машины. Сокращение времени разработки и изготовления ЭВМ до минимума возможно на основе применения ЭВМ на всех стадиях работ. Принципиальная возможность автоматической разработки ЭВМ за практически небольшое время показана в ряде исследований, как об этом говорится в работе [1].

Хуже обстоит дело с автоматизацией процесса производства, так как трудно обеспечить соответствующую скорость изготовления элементов ЭВМ. Чтобы преодолеть эту трудность, следует (как указывалось в работе [1]) добиваться, с одной стороны, наибольшей универсальности стандартного элемента (СЭ) элементарной машины (т.е. один и тот же элемент должен осуществлять:

- а) функции запоминания;
- б) логические функции $\vee$, $\wedge$, $\neg$;
- в) задержку сигналов;
- г) стандартизацию сигналов и т.д.),

с другой стороны, - минимума технологических операций на изготовление СЭ. При этом СЭ или даже ряд СЭ мыслится как единый кусок твердого тела.

Если предположить, что СЭ используются также и в качестве элементов связи между СЭ, то время изготовления ЭМ и ВС будет определяться скоростью изготовления СЭ. Таким образом, решение вопроса сводится к обеспечению соответствующей скорости изготовления СЭ. Далее, так как производительность ВС в конечном счете определяется общим количеством СЭ [см.1], то по

мере появления новых классов задач выполнение требований универсальности, производительности и точности будет автоматически обеспечиваться также скоростью производства СЭ.

Наконец, то обстоятельство, что соответствующая надежность ЭВМ также во многом определяется избыточностью элементов [9, 10, 11], а на экономичность при условии микроминиатюризации влияют затраты не на материалы, из которых изготавливается ВС, а только на способ производства элементов, дает основание утверждать, что основным требованием при создании ВС является обеспечение соответствующей скорости изготовления СЭ.

Возможны два пути создания ВС в зависимости от требования к скорости изготовления СЭ.

I - й путь заключается в достижении максимально возможной технологической скорости изготовления ВС. Очевидно, время изготовления ВС должно быть сравнимо со временем решения задач. Каждый СЭ предназначается для выполнения одной какой-либо функции, что несколько упрощает технологию изготовления СЭ, но (вместе с тем) накладывает ограничения на классы решаемых задач. Однако достижение таких скоростей в изготовлении СЭ на практике трудно осуществимо, поэтому более реалиен II-й путь построения ВС, позволяющий существенно снизить скорость изготовления СЭ.

II - й путь предполагает, что структура ЭМ меняется путем соответствующей настройки^{х)} СЭ (в зависимости от класса решаемых задач). Особенно следует подчеркнуть, что при II пути не требуется предварительного знания класса решаемых задач, вследствие чего СЭ изготавливаются заранее и на их изготовление можно отвести достаточно большое время. В результате скорость изготовления СЭ ВС может быть существенно снижена.

Наконец, скорость изготовления отдельного СЭ может быть снижена и за счет перехода к групповому изготовлению элементов (т.е. когда в едином процессе одновременно производится целая группа СЭ). В этом случае необходимо добиваться возможно меньшего разнообразия как для отдельного СЭ, так и для схем, блоков и устройств машины в целом. Другими словами, при производстве ВС возникает необходимость в использовании струк-

х) Под настройкой понимается выбор функций из заданного набора, выполняемого СЭ в каждый данный момент времени.

тур с возможно большей однородностью.

В связи с вышесказанным возникает задача разработки однородной структуры элементарной машины.

В данной работе рассматриваются возможные подходы к созданию однородной структуры ЭМ, предъявляющей минимальные требования к скорости изготовления СЭ. В основном рассматриваются однородные структуры ЭМ, обеспечивающие автоматическую перестройку ЭМ в зависимости от классов решаемых задач.

Возможны два уровня однородности ЭМ: однородность на уровне устройств блоков, узлов и однородность на уровне СЭ. Первый подход будет в дальнейшем называться макроструктурным, второй - микроструктурным.

§ 2. Макроструктурный подход к построению однородной элементарной вычислительной машины

Прежде чем перейти к рассмотрению макроструктуры ЭМ, выскажем некоторые соображения, которые следует учитывать при построении ЭМ.

Будем при этом исходить из блок-схемы ЭМ, представленной на рис. 1. Основные затраты оборудования ЭМ приходятся на устройство памяти. Затраты оборудования на арифметическое устройство (АУ), на устройства управления (УУ) и связи (УС) составляют незначительную часть общих затрат. Вместе с тем устройства памяти имеют очень простую структуру и большую однородность, а устройства АУ, УУ при незначительных затратах оборудования имеют достаточно сложную логическую структуру и большое разнообразие логических схем. Учитывая вышесказанное, можно было бы пренебречь временем изготовления устройств АУ и УУ и исходить только из времени изготовления устройств памяти. Однако это не так.

Высокая логическая сложность устройств АУ и УУ и большое разнообразие схем предъявляют высокие требования как к процессу изготовления таких схем, так и к автоматической установке для производства ЭМ. Особенно большие трудности возникают при изготовлении связей между элементами, где довольно резко проявляется разнообразие схем. Возникает задача построения таких схем АУ и УУ, которые были бы по сложности одного порядка со схемами устройств памяти и, следовательно, не предъявляли бы высоких требований к технологии и средствам автоматизации изготовления.

Отметим, что устройство памяти из-за ограничений, накладываемых на рабочую частоту элементов машины, а также требования большого объема памяти и достаточно высокой скорости выполнения операций должно допускать произвольную выборку информации по заданному адресу. Следовательно, структура устройства памяти является относительно сложной, и требование, чтобы сложность АУ и УУ была одного порядка с устройством памяти, не является слишком жестким.

Рассмотрим возможные пути построения ЭМ с относительно простой макроструктурой.

1. Построение ЭМ с ограниченным набором операций. Одним из путей упрощения структуры является переход к построению ЭМ с ограниченным набором операций. Очевидно, что уменьшение числа различных операций в ЭМ приводит к упрощению структуры устройства управления. Выбор же самих операций влияет на простоту структуры арифметического устройства.

В работах [12,13] указывается на возможность построения универсальных машин с весьма ограниченным набором операций (например, сдвиг, сложение, условный переход или вычитание (сложение) и условный переход). Такие наборы при относительной простоте схем АУ и УУ позволяют реализовать любые алгоритмы. Следует отметить, что ограниченный набор операций не позволяет обеспечить достаточно высокую производительность.

Свободной в известной мере от указанных недостатков могла бы быть элементарная машина, реализующая операцию выборки слова. Выполнение этой операции может быть представлено следующим образом.

Пусть требуется выполнить некоторую операцию R над словами α и β и получить результат $R(\alpha, \beta) = \gamma$. Тогда в памяти машины для каждой возможной пары слов α, β по соответствующему адресу $R(\alpha, \beta)$ следует заранее запомнить результат операции γ . В этом случае выполнение операции сводится к составлению адреса из слов, над которыми производится операция, и выборки из памяти результата по составленному адресу, т.е. в конечном счете - к реализации табличной операции (аналогично таблицам умножения, деления, сложения и т.д.).

Очевидно, что при достаточном объеме памяти принципиально возможно осуществить любую операцию и получить максимально высокую производительность.

Операция выборки может быть также в принципе распростра-

нена и на n слов, т.е. $R(\alpha_1, \alpha_2, \dots, \alpha_n) = \gamma$. Осуществление операции выборки не может привести к схемам более сложным, чем схемы устройства памяти, так как операции записи в память и считывания с памяти включают в себя операцию выборки. Следовательно, применение операции выборки позволяет строить схемы ЭМ со сложностью, не превышающей сложности схем устройства памяти.

Необходимый объем памяти для запоминания таблиц операций сильно зависит от разрядности слов. В связи с этим нерационально запоминать таблицы для выполнения операций над много-разрядными числами. Поэтому приходится использовать таблицы для выполнения операций над словами с меньшим числом разрядов, хотя это и приводит к некоторому снижению производительности ЭМ.

2. Построение ЭМ с ограниченным числом разрядов в слове.

Требование одновременной выборки и обработки произвольного числа разрядов слова чрезмерно усложняет структуру ЭМ. В связи с этим возникает вопрос об ограничении числа одновременно выбираемых или обрабатываемых разрядов.

Имеют место два крайних случая:

1) когда выборка и обработка производится по одному двоичному разряду. В этом случае максимально упрощается структура ЭМ, но увеличивается время выполнения операций, а следовательно, уменьшается производительность машины;

2) когда выбирается максимальная длина слова, которая требуется для выполнения операций с заданной точностью. В этом случае, хотя структура машины относительно проста при сохранении максимально возможной производительности, затраты оборудования значительно возрастут.

Первый случай находит свое практическое осуществление в машинах последовательного действия [14], второй случай - в машинах параллельного действия [15].

Промежуточным вариантом является построение машины параллельно-последовательного действия [16]. В этом случае удастся сочетать простоту структуры машин последовательного действия с высокой производительностью машин параллельного действия. Здесь допускается одновременная выборка и обработка некоторого ограниченного количества двоичных разрядов слова. Удобно это количество разрядов принять за элемент слова - "букву". В этом случае каждая буква далее неделима, что позволяет из-

бежать операций сдвига, и адресуема, что удобно при обработке информации. Адресуемость букв позволяет выполнять операции над словами произвольной длины и эффективно использовать память [17].

Снижение производительности, вызываемое переходом к операциям последовательного или параллельно-последовательного действия может быть значительно компенсировано за счет параллельного выполнения операций. В частности, по-видимому, очень эффективен метод, предложенный в работе [18]. В этой работе вместо обычного способа размещения данных, предлагается помещать в каждой ячейке памяти разряды разных слов. В этом случае можно выполнять операцию сразу над несколькими словами. Выигрыш во времени за счет одновременного выполнения операций над несколькими словами получается тем больше, чем короче слова. Сложность структуры в этом случае не больше, чем в машинах последовательного действия.

3. Построение ЭМ с ограниченным набором микроопераций.

Простая структура элементарной машины может быть получена введением в устройства АУ и УУ некоторого набора микроопераций, из которых можно программным способом образовать более сложные операции [19]. В этом случае АУ и УУ могут быть представлены в виде матриц, управляющих друг другом. В одной матрице могут быть записаны микрооперации, а в другой — последовательность выполнения и типы микроопераций, необходимые для реализации данной сложной операции. Структура ЭМ получается не сложнее структуры памяти, так как матрицы по сути дела являются небольшими устройствами постоянной памяти.

При достаточно большом наборе микроопераций можно осуществлять широкий класс сложных операций и тем самым обеспечивать необходимую производительность. Если предусмотреть возможность записи информации в матрицы программным способом, то можно в некоторых границах менять набор микроопераций и последовательность их выполнения. В этом случае можно выбирать, в известных пределах, такие операции, которые обеспечивают решение данной задачи с наибольшей эффективностью.

4. Построение ЭМ из простейших машин, автоматов, модулей.

Вышеперечисленные пути построения ЭМ основаны на представлении машины в виде единого функционального целого. При этом сложность структуры различных устройств определялась сложностью наибольшего по затратам оборудования устройства, в

частности, устройства памяти. В настоящее время наметился еще и другой путь, при котором вся вычислительная машина состоит из некоторых простейших машин, соединенных по определенному закону. Такие соединения однородных машин получили название итеративных цепей [20].

Одним из возможных вариантов подобного вида является построение ЭМ из простейших машин с минимальным числом ячеек с произвольной выборкой. В частности, возможно построение машины, подобной универсальной машине Тьюринга с двумя внутренними состояниями [21], но с конечным объемом внешней памяти. В таком случае в качестве внешней памяти может быть использована своего рода динамическая память с хранением информации в некоторой среде (например, волноводах, линиях задержки и т.п.). Структура памяти при этом становится исключительно простой, так как нет необходимости в произвольной выборке слов, а значит, и в соединениях для связи с каждым элементом памяти. Простота остальной части машины обусловлена минимумом внутренних состояний. Если каждая из простейших машин будет предназначена для выполнения какой-нибудь операции или микрооперации, то можно получить достаточно высокую производительность при относительно простой структуре ЭМ.

Возможен также вариант построения ЭМ из некоторых простейших автоматов, аналогичный предложенному в работе [22].

Большое развитие получило направление, основанное на построении итеративных цепей из однородных модулей [23, 24]. В качестве модуля используется вычислительное устройство, приспособленное для выполнения некоторого ограниченного набора операций и имеющее определенное число ячеек памяти для хранения информации. Каждый модуль имеет каналы связи с соседними модулями для обмена информацией.

Возможны различные варианты построения элементарной машины из модулей.

При централизованном управлении модулями команды задаются из единого центрального устройства управления. В каждый момент выполняется одна и та же операция всеми модулями [23]. При таком подходе к построению ЭМ производительность ее может быть достаточно высокой, но эффективность использования модулей будет довольно низкой.

При децентрализованном управлении каждый модуль имеет регистр для хранения команды и может выполнять в каждый данный момент любую операцию из заданного набора, независимо от опе-

раций, выполняемых другими модулями. Фактически каждый модуль в данном случае является вычислительным устройством с ограниченным объемом памяти [24]. При этом эффективность использования модулей, а также производительность ЭМ значительно выше, чем при централизованном управлении.

Рассмотренные пути построения ЭМ показывают, что макро-структурный подход к построению ЭМ в принципе позволяет получить однородную макроструктуру. В случаях 1-3 сложность структуры ЭМ определяется сложностью структуры памяти, а в случае 4 - сложностью структуры модуля.

Хотя при макроструктурном подходе и удается в значительной степени повысить производительность вычислений, однако трудно гарантировать достижение максимальной производительности для любого класса задач. Существенно также влияет на производительность и требование высокой точности вычислений. Следовательно, макроструктурный подход не снимает необходимость знать классы решаемых задач для достижения высокой производительности. Для получения максимально простой структуры ЭМ требуется провести исследования по анализу вариантов построения структуры памяти или модулей. При этом может оказаться, что найденные варианты будут слишком сложны для автоматического процесса производства.

§ 3. Микроструктурный подход к построению элементарной машины

Применение принципов микроминиатюризации [25,26,27] заставляет по-новому подойти к определению требований к стандартным элементам и соединениям между ними. Эти требования особенно усиливаются в связи с необходимостью автоматизировать процесс производства для обеспечения высокой скорости изготовления СЭ.

Прежде всего элементы должны быть достаточно малых размеров (линейные размеры порядка микрона) и достаточно простыми, с тем, чтобы их изготовление сводилось к выполнению небольшого числа технологических операций. Вместе с тем, для осуществления любой заданной схемы требуется, чтобы СЭ были универсальными, т.е. имели элементы памяти и выполняли полный набор логических функций V , $\&$, $!$. Уменьшение габаритов элементов приводит к всевозрастающей роли соединений между элемен-

тами. В ряде работ [28,29] указывалось, что уменьшение габаритов вычислительной машины прежде всего определяется соединениями между элементами. В связи с этим логично предположить, что для построения соединений требуется затратить не меньшее количество технологических операций, чем для производства СЭ. Следовательно, можно ввести некоторый элемент соединения и принять, что затраты на его производство того же порядка, что и для СЭ. Для универсальности можно потребовать, чтобы СЭ выполнял также функцию соединения. Усиление роли элементов соединений и использование в качестве последних самих СЭ приводит к необходимости применения принципа близкодействия, который заключается в том, что только соседние элементы непосредственно воздействуют друг на друга.

Другим важным принципом является принцип однородности на уровне СЭ, обусловленный требованием высокой скорости производства СЭ. Указанные принципы приводят к необходимости создания из стандартных элементов **в** **ы** **ч** **и** **с** **л** **и** **т** **е** **л** **ь** **н** **о** **й** **с** **р** **е** **д** **н**. Для вычислительной среды характерна однородность не только элементов, но и соединений между ними. Вычислительные среды нашли широкое применение в вычислительной технике непрерывного действия [30, 31, 32].

В технике непрерывного действия для моделирования полей используются два метода: метод сплошных сред и метод сеток. В качестве модели для метода сплошных сред применяется поле электрического тока в сплошной проводящей среде. Для метода сеток в качестве модели используется электрическая сетка, основанная на дискретном представлении элементарных объемов моделируемой области и на замещении их электрическими цепями с сосредоточенными параметрами [33]. Электрическая сетка и тем более сплошная среда характеризуются высокой однородностью как элементов, так и соединений между ними. Отметим также и некоторые ограничения, которые характерны для упомянутых вычислительных сред. Элементы среды не являются универсальными и выполняют только одну определенную (хотя, может быть, и сложную) операцию. Другим, еще более существенным ограничением для этих вычислительных сред, является то, что все элементы среды одновременно выполняют при моделировании одну и ту же операцию и передают и принимают информацию по одним и тем же направлениям.

Для построения элементарной машины требуется вычислительная среда, в которой можно было бы одновременно реализовать

различные операции и различные направления обмена информацией.

На основании вышесказанного сформулируем некоторые требования и принципы, которым должна удовлетворять вычислительная среда, предназначенная для построения элементарных машин.

1) Вычислительная среда должна состоять из универсальных элементов, каждый из которых должен выполнять функции V , $\&$, Γ , хранить информацию и осуществлять соединения с другими элементами.

2) При соединении элементов должен соблюдаться принцип близкодействия, т.е. элементы соединяются только со своими ближайшими соседями.

3) Вычислительная среда должна быть однородной, т.е. состоять из одних и тех же стандартных элементов с одинаковыми конфигурациями соединений между ними.

4) Вычислительная среда должна обеспечивать возможность настройки различных элементов на выполнение различных функций и задания различных направлений обмена информацией.

По-видимому, существует большое количество самых разнообразных вычислительных сред, удовлетворяющих этим требованиям.

Представляет большой теоретический и практический интерес отыскание сплошной вычислительной среды, удовлетворяющей заданным требованиям, по аналогии со сплошными требованиями в технике непрерывного действия. Следует отметить, что возможность моделирования "поведенческих актов" управляющих систем сплошными средами показана в работе [34], где говорится, что сплошная среда обладает свойством запоминания, если у ее элементов возникает спонтанная активность. Построение ЭМ на этой основе встречает большие трудности из-за сложности и недостаточной изученности явлений в сплошной среде.

Более плодотворным, по-видимому, является использование идеи дискретной вычислительной среды.

Мы будем различать дискретные вычислительные среды двух типов:

- 1) с индивидуальным поведением элементов,
- 2) с коллективным поведением элементов.

Под вычислительной средой с индивидуальным поведением элементов будем понимать такую среду, в которой возможна индивидуальная настройка любого отдельного элемента среды на выполнение какой-либо одной функции. При этом под выполняемой элементом функцией понимаются: функция памяти, выполнения операций

$\&$, V , Γ , прием и передача информации по заданным направлениям.

Под вычислительной средой с коллективным поведением элементов будем понимать такую среду, в которой выполнение той или иной функции определяется групповым взаимодействием элементов, каждый из которых выполняет одну и ту же операцию. Под выполняемой функцией понимается то же, что и в случае вычислительной среды с индивидуальным поведением элементов. Возможность применения вычислительной среды с коллективным поведением элементов для моделирования простейших актов поведения показана в работе [35].

Рассмотрим теперь более подробно особенности построения и работы вышесказанных вычислительных сред.

1. Вычислительная среда с индивидуальным поведением элементов.

Для простоты рассмотрим двумерную вычислительную среду (все рассуждения будут справедливы и для случая n -мерной вычислительной среды).

Если расположить эту среду в прямоугольной системе координат (рис.2), то каждый элемент среды α_{ij} будет определен координатами i, j .

Каждый элемент среды соединен с ближайшими соседними элементами $\alpha_{i,j-1}$; $\alpha_{i,j+1}$; $\alpha_{i-1,j}$; $\alpha_{i+1,j}$. Предположим, что с элементами $\alpha_{i-1,j-1}$; $\alpha_{i-1,j+1}$; $\alpha_{i+1,j+1}$; $\alpha_{i+1,j-1}$ - соединения не допускаются.

Чтобы удовлетворить требованию универсальности, положим, что любой элемент среды может быть настроен на выполнение одной из следующих функций: V , $\&$, Γ ; запоминания 0 и 1; задержки; соединения с соседними элементами.

Вполне очевидно, что при заданном наборе функций требование универсальности заведомо выполняется, так как для реализации логически полного набора функций было бы достаточно, например, введения функций $\&$, Γ и задержки.

Функция соединения с соседними элементами нуждается в более подробном объяснении. Зададимся целью построить такой элемент соединения, который обеспечивал бы любое возможное соединение (коммутацию) с соседними элементами, из полного перебора всевозможных соединений. Так как по принципу близкодействия соседними элементами считаются элементы, расположенные в непосредственной близости по осям координат, то такой перебор невелик и практически осуществим. Пример подобного элемента показан на рис.3.

Нуждается также в пояснении термин "настройка элемента". Положим, что каждой функции f_{ij} элемента α_{ij} , в том числе и каждому виду соединений с соседними элементами, соответствует определенная функция включения P_{ij} , принимающая значение 0 или 1. Предположим, что $P_{ij} = 1$, если элемент выполняет функцию f_{ij} , и $P_{ij} = 0$ для f_{ij} . Тогда, задавая набор функций включения P_{ij} , можно настроить элемент на выполнение набора функций f_{ij} .

В общем случае элемент среды может быть представлен в виде некоторой функциональной схемы. Как видно из рис.3, функция включения P_{ij} может быть осуществлена в виде некоторого накопительного регистра сдвига с числом ячеек, соответствующим общему количеству состояний P_{ij} . На вход этого регистра могут быть поданы сигналы при условии их одновременного поступления по координатам настройки i, j . Функциям f_{ij}, f_{zj} соответствуют коммутации входов и выходов, f_{zj} - соответствует функциям $V, \&, \uparrow$; запоминания, задержки и соединения. Такой способ задания настройки назовем координатным. Путем подачи требуемой последовательности сигналов настройки на входы i, j элемента α_{ij} , а на вход S - сигнала тактовой частоты настройки, можно настроить элемент α_{ij} на выполнение требуемой функции. Вполне очевидно, что, посылая сигналы настройки по соответствующим координатам, можно настроить любой элемент среды на выполнение той или иной функции.

Подчеркиваем, что универсальность элемента и осуществимость любого соединения с соседними элементами обеспечивает принципиальную возможность построения в вычислительной среде любой многотактной схемы, а следовательно, и любой вычислительной машины.

Естественно, что возможность практической реализации схемы вычислительной машины будет определяться общим количеством элементов в вычислительной среде.

Приведенная функциональная схема элемента вычислительной среды может быть существенно упрощена за счет отбрасывания избыточных функций и перехода к многокоординатным средам.

Следует также подчеркнуть, что так как элемент среды является достаточно простым и имеет ограниченное количество соединений, то можно надеяться на строгое математическое решение задачи о выборе типа среды и ее элемента. Работы в этой области начаты многими авторами [20, 36, 37].

В вычислительной среде с индивидуальным поведением элементов большое значение имеет способ задания настройки элементов. Выше был приведен координатный способ, при котором настройка задавалась по специальным линиям связи, расположенным параллельно осям координат. Возможен также способ задания настройки по схеме информационного дерева, образованного соединением некоторых элементов ветвления. Пример элемента ветвления показан на рис.4. По каналам I передается настроечная информация, с помощью которой устанавливаются соответствующие состояния функции включения P_{ij} . По каналам II передается разрешение на настройку выбранного элемента. По каналам III настроечная информация производит непосредственно настройку выбранного элемента. Настройка данного элемента разрешена, если триггер T_1 находится в состоянии 1. В этом случае настроечная информация, поступающая по каналам I, передается в каналы III. После завершения передачи настроечной информации для данного элемента по каналам I передается информация, управляющая выбором пути (T_2 устанавливается в состояние 0 или 1). Далее, по каналу II передается информация включения настройки очередного элемента (T_1 устанавливается в состояние 0, а предыдущее состояние T_1 передается по каналам II к соседним элементам ветвления. Выбор соседних элементов определяется состоянием T_2).

В силу высокой однородности среды, по-видимому, возможно строгое математическое решение вопроса о наиболее рациональном способе настройки элементов среды.

2) Вычислительная среда с коллективным поведением элементов.

В качестве первого приближения для построения вычислительной среды с коллективным поведением элементов можно использовать результаты, полученные в работе [35]. В этой работе за элемент среды принимается некоторый пороговый элемент, соединенный с восемью ближайшими соседними элементами. Этот элемент возбуждается только при поступлении сигналов не менее чем на два входа. Предполагается, что все связи проводят сигналы в обоих направлениях в течение некоторого времени τ .

Если на элемент подать сигнал, то он перейдет в возбужденное состояние и выдаст сигнал по всем связям к соседним элементам за некоторое время, меньшее τ . После выдачи сигнала элемент находится в состоянии невозбудимости в течение времени R , причем $2\tau < R < 3\tau$. Схема такой среды при-

ведена на рис.5. В работе [35] показано, что в подобной среде возможно осуществлять простейшие функции алгебры логики, запоминание информации, выбор направления передачи информации. Другими словами, имеется принципиальная возможность построения любой сложной функции или любой схемы вычислительной машины. Задание той или иной функции в вычислительной среде достигается одновременным возбуждением определенной конфигурации элементов.

Основная трудность реализации схем вычислительной машины в вычислительной среде с коллективным поведением элементов заключается в необходимости одновременного возбуждения большого числа элементов среды в определенной конфигурации. Эта трудность частично снимается, если ограничиться построением относительно простых схем. Можно применить вычислительную среду для построения универсального элемента с возможностью индивидуальной настройки на заданную функцию и на заданное направление передачи или приема информации. В этом случае универсальный элемент может быть образован из слоя вычислительной среды и слоя, обеспечивающего одновременное возбуждение элементов среды для выполнения той или иной функции.

Применение вычислительных сред с коллективным поведением элементов позволяет уменьшить требования к технологии микроминиатюризации, так как элемент среды является довольно простым и имеет постоянные неперекрывающиеся связи с соседними элементами.

В заключение отметим, что микроструктурный подход приводит к построению вычислительных сред. Вычислительные среды позволяют принципиально по-новому подойти к вопросам построения вычислительных машин. Фактически из вычислительной среды можно построить вычислительную машину любой сложности. Вместе с тем высокая однородность среды позволяет свести до минимума требования к технологии, а также применить групповую технологию для изготовления элементов. Достаточная простота элементов и связей между ними, в принципе, позволяют построить строгую математическую теорию вычислительных сред и, следовательно, ставить вопрос о выборе оптимальных параметров вычислительной среды.

§ 4. Некоторые проблемы, связанные с построением вычислительных сред

Построение вычислительных сред связано с решением ряда проблем.

Прежде всего сюда относятся вопросы разработки математической теории вычислительных сред.

Можно предположить, что простота элементов и связей между ними позволят разработать аксиоматическую теорию сред, аналогично тому, как это сделано в исчислении высказываний, исчислении предикатов и т.п. Очень важными являются также вопросы разработки математического языка описания сред, в том числе описание связей между элементами, заданий направлений передачи информации, функций, выполняемых элементами среды, способов настройки среды и т.п.

Необходимо также провести исследование свойств двумерных, трехмерных, n -мерных вычислительных сред. Не менее важную роль играют вопросы, связанные с исследованием различных типов сред, например, сложных сред, состоящих из слоев с элементами соединений и слоев с функциональными элементами.

Непосредственно для построения вычислительных сред необходимо провести исследования по определению минимально допустимого набора функций в элементе, минимально допустимого количества связей между элементами, выбора оптимального элемента среды и числа связей. В связи с построением схем из вычислительной среды необходимо решить следующие задачи:

- а) разработать оценки возможности построения схем заданной сложности из данного количества элементов;
- б) разработать алгоритмы переработки вычислительной среды в заданную схему;
- в) разработать методы настройки вычислительной среды на выполнение заданной функции.

Большое значение имеют также исследования, связанные с построением вычислительной среды из реальных элементов вычислительной техники (параметроны, туннельные диоды, криотроны и т.п.). Построение реальных вычислительных сред требует также рассмотрения вопросов расчета надежности среды, построения схем в средах с минимальным числом соединений, с минимальными затратами энергии, минимальными затратами времени на распространение сигналов и т.д.

Играют важную роль также вопросы разработки методов моде-

лирования вычислительной среды. Применение моделирования на электронных вычислительных машинах для исследования логической структуры среды и физических процессов, протекающих в ней, может сыграть решающую роль при исследовании и выборе среды.

Применение вычислительных сред для построения вычислительных машин позволяет по-новому подойти к вопросам автоматизации процессов проектирования и изготовления электронных вычислительных машин.

В связи с этим необходимо исследовать влияние структуры вычислительной среды на сложность процессов проектирования и изготовления и установить зависимость между сложностью вычислительной среды и требованиями к технологии.

З а к л ю ч е н и е

Неопределенность требований, предъявляемых к универсальным вычислительным машинам и, в частности к элементарным машинам вычислительной системы, приводит к необходимости построения машин со структурой, изменяющейся в зависимости от решаемых задач. Требования высокой технологичности и малых затрат времени на производство машин приводят к необходимости строить вычислительные машины высокой однородности.

Весьма перспективными в этом отношении являются вычислительные среды, которые принципиально позволяют создавать вычислительные машины со структурой, изменяющейся в соответствии с решаемой задачей, и которые удовлетворяют требованию однородности. Применение вычислительных сред существенно упрощает требования к автоматизации проектирования и изготовления вычислительных машин.

Вычислительные среды также могут обеспечивать построение вычислительных машин с максимально высокой производительностью, надежностью и экономичностью.

П Р И Л О Ж Е Н И Е

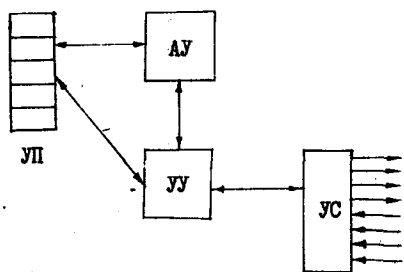


Рис. 1. Блок-схема элементарной машины:
УП-устройство памяти; АУ-арифметическое устройство;
УУ-устройство управления; УС-устройство связи с соседними машинами.

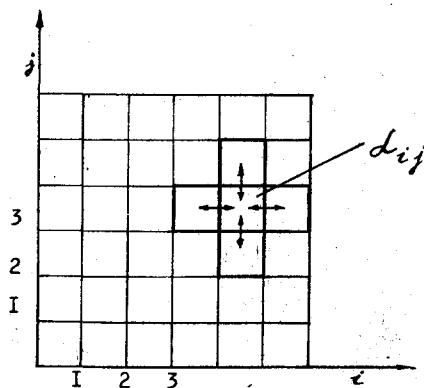


Рис. 2. Вычислительная среда с индивидуальным поведением элементов:

L_{ij} - элемент вычислительной среды.

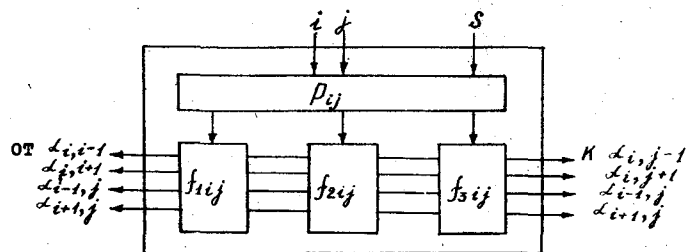


Рис. 3. Функциональная схема L_{ij} элемента вычислительной среды с индивидуальным поведением элементов:
 P_{ij} - функция включения; f_{1ij} - функция коммутации входов элемента;
 f_{2ij} - функция элемента; f_{3ij} - функция коммутации выходов элемента;
 i, j, s - входы.

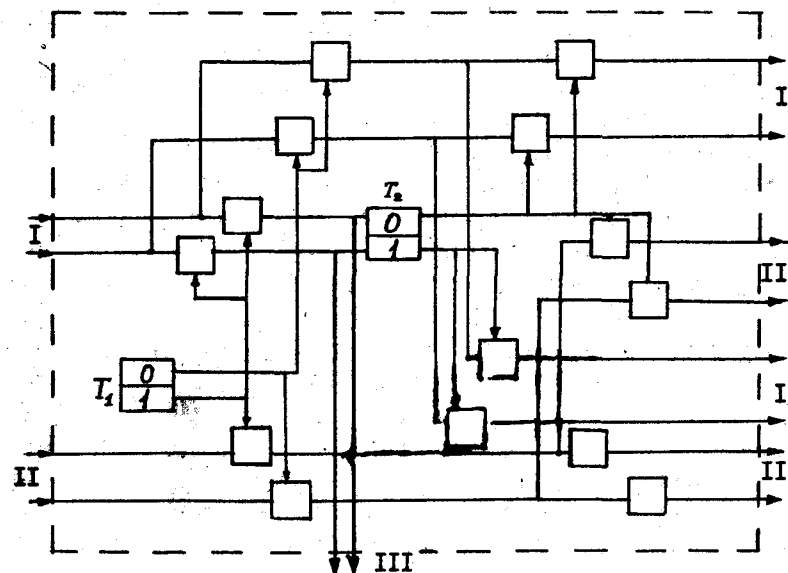


Рис. 4. Элемент ветвления дерева :

$\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}$ - триггер, I - каналы передачи информации настройки;
□ - схема, II - каналы включения настройки элемента,
III - каналы настройки данного элемента.

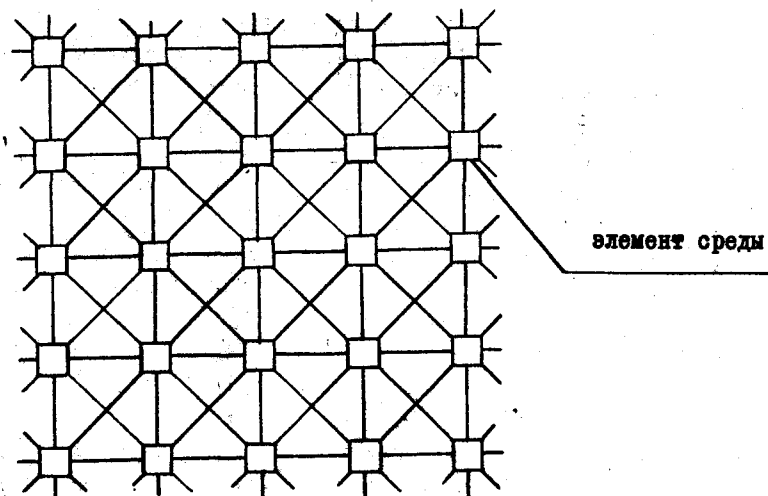


Рис. 5. Схема вычислительной среды с коллективным поведением элементов

ЛИТЕРАТУРА

1. Евреинов Э.В., Косарев Ю.Г. О возможности построения вычислительных систем высокой производительности. Новосибирск, Изд-во СО АН СССР, 1962.
2. Лебедев С.А., Мельников В.А. Общее описание БЭСМ и методика выполнения операций. М., Физматгиз, 1959.
3. Turing A.M. On computable numbers, with an application to the Entscheidungsproblem. - Proc.London.Math.Soc., ser.2, 1936, 42, pp. 230-265.
4. Белякин Н.В. Универсальность вычислительной машины с потенциально бесконечной внешней памятью. В сб. "Проблемы кибернетики", вып. 5, М., Физматгиз, 1961, 77-86.
5. Мамонов Е.И. Оптимальная скорость работы и другие технические показатели оперативных устройств хранения информации электронных автоматических цифровых вычислительных машин. Конференция "Пути развития советского математического машиностроения и приборостроения", ч.П, 1956, 56-62.
6. Базилевский Ю.Я., Шрейдер Ю.А. Методы оценки производительности универсальных цифровых машин с программным управлением. В сб. "Вопросы теории математических машин". М., Физматгиз, 1958, 127-134.
7. Рамеев Б.И. Универсальная автоматическая цифровая машина типа "Урал". Конференция "Пути развития советского математического машиностроения и приборостроения", ч.1, 1956, 38-52.
8. Householder A.S. Numerical mathematics from the viewpoint of electronic digital computers. - Nachrichtentechn. Fachber., 1956, 223, pp. 21-25.
9. Нейман Д. Вероятностная логика и синтез надежных организмов из ненадежных компонент. В сб. "Автоматы". М., ИЛ, 1956, 68-139.
10. Мур Э.Ф., Шеннон К.Э. Надежные схемы из ненадежных реле. В сб. "Кибернетический сборник". № I, М., ИЛ., 1960, 109-148.
11. Widrow B., Pierce W.H., Angell J.B. Birth, Life and Death in Microelectronic Systems. -IRE Transaction on Military Electronics. July, 1961, No.3, pp.191-201.
12. Китов А.И., Криницкий Н.А. Электронные цифровые машины и программирование. М., Физматгиз, 1959.
13. Глушков В.М. Теория алгоритмов. Киев, 1961.
14. Гильман А.М. К проекту вычислительной машины последовательного действия. Конференция "Пути развития советского математического машиностроения и приборостроения", ч.1, 1956, 82-91.
15. Базилевский Ю.Я. Универсальная электронная вычислительная машина "Стрела". - Приборостроение, 1957, № 3.
16. Gerwin R. Ein elektronischer Rechner ohne Elektronenrohren. - "VDI-Nachr.", 1959, No.9-10.
17. Брукс Ф.П., Блоу Д.А., Бухгольц У. Переработка данных по-разрядно и кусками. - "Кибернетический сборник", М., ИЛ., 1961, № 2, 169-186.
18. Shooman W. Parallel computing with vertical data. - Proc.EJCC, 1960, pp.111-115.
19. Wilkes M.V. Micro-programming. - Proc.EJCC, 1958, December 3-5, pp.18-20.
20. Holland I.H. Iterative circuit computers. - Proc.WJCC, 1960, pp.259-265.
21. Шеннон К.Э. Универсальная машина Тьюринга с двумя внутренними состояниями. В сб. "Автоматы". М., ИЛ, 1956.
22. Gill A. Cascaded finite-state Machines. - IRE Trans. on EC, 1961, September, EC-10, pp.366-370.
23. Unger S.H. A computer Oriented toward Spatial Problems. - Proc.IRE, 1958, 46, 10, pp. 1744-1750.
24. Holland J.H. A universal computer capable of executing an arbitrary number of subprograms simultaneously. - Proc. EJCC, December 1-3, 1959, pp.108-113.
25. New Molecular Electronics. - Electronics, 1959, 12, 6, p.43.
26. Микроминиатюризация радиоэлектронной аппаратуры. Л., Судпромгиз, 1962.
27. Слуцкая В.В. Тонкие пленки в технике сверхвысоких частот. М.-Л., Госэнергоиздат, 1962.
28. Rice R. The impending revolution in computer technology. - Proc.EJCC, 1959, pp.43-46.
29. Buck D.A., Shoulders K.R. An approach to microminiature printed systems. - Proc. EJCC, 1959, pp.55-59.
30. Тетельбаум И.М. Электрическое моделирование. М., Физматгиз, 1959.
31. Веников В.А. Применение теории подобия и физического моделирования в электротехнике. М., ГЭИ, 1949.
32. Карплюс У. Моделирующие устройства для решения задач теории поля. М., ИЛ, 1962.

33. Гутенмахер Л.И. Электрические модели. М., Изд-во АН СССР, 1949.
34. Гельфанд И.М., Цетлин М.Л. О непрерывных моделях управляющих систем. - ДАН СССР, 1960, 131, № 6, 1242-1245.
35. Балаховский М.С. О возможности моделирования простейших актов поведения дискретными однородными средами. В сб. "Проблемы кибернетики", 5, М., Физматгиз, 1961, 272-277.
36. Caldwell S.H. Switching Circuits and Logical Design. 1958.
37. Hennie F.C. Iterative Arrays of Logical Circuits. 1961.