

УДК 681.3

ПРОЕКТИРОВАНИЕ ГЛОБАЛЬНО АСИНХРОННЫХ СИСТЕМ С ПРОИЗВОЛЬНОЙ ЛОКАЛЬНОЙ СИНХРОНИЗАЦИЕЙ

В. Б. Мараховский,

доктор техн. наук, профессор

В. Ф. Мелехин,

доктор техн. наук, профессор

Санкт-Петербургский государственный политехнический университет

Рассматриваются задачи проектирования устройства (синхростратума), выполняющего функции глобальной синхронизации в массивно параллельных системах с произвольной локальной синхронизацией блоков; методика проектирования синхростратума в классе самосинхронных автоматов с использованием синхронных прототипов с различными дисциплинами синхронизации.

Ключевые слова — синхронизация, логическое время, физическое время, синхронные и асинхронные автоматы, массив автоматов, синхростратум, проектирование.

Введение

Системы с высокой степенью параллелизма работы составляющих их блоков получили название массивно параллельных систем. Координация взаимодействия блоков на основе асинхронных принципов (глобальная синхронизация) в таких системах привлекательна тем, что позволяет использовать преимущества синхронного и асинхронного принципов построения аппаратных средств и свести к минимуму их недостатки.

Более подробно о проблеме синхронизации, о концепциях физического и логического времени и о различном понимании терминов «синхронность» и «асинхронность» специалистами в области программного обеспечения и в области аппаратных средств можно прочесть в статье [1]. Напомним лишь, что физическое время может быть преобразовано в логическое или с помощью введения внешних часов (в синхронных системах), или с использованием методологии самосинхронизации (в асинхронных системах). Логическое время называют глобальным, если в нем осуществляется синхронизация работы структурных блоков системы, и локальным — действующим внутри структурного блока.

Как было показано в работах [1–6], асинхронная система может быть декомпозирована на подсистему синхронизации (синхростратум) и синхронизируемую подсистему ее блоков (условно называемую «процессорным стратумом»). Такой

подход дает возможность использовать для построения глобально асинхронных систем синхронные прототипы, что обеспечивает преемственность и эволюцию аппаратных средств.

Выделение синхростратума существенно упрощает переход от синхронного прототипа системы к ее асинхронной реализации. Синхростратум выполняет роль распределенных асинхронных часов, осуществляющих глобальную синхронизацию блоков системы, и в логическом времени может быть построен в соответствии с системой синхронизации прототипа.

Отказ от общих системных часов требует асинхронного взаимодействия синхростратума с блоками системы, которое может быть организовано на основе хендшейка (работы в соответствии с принципом «запрос-ответ»). Для организации взаимодействия по такому принципу необходимы сигналы окончания переходных процессов в блоках. Эта задача может быть решена различными способами в зависимости от типа и структуры блока, его размера, возможной вариации длительности переходного процесса в нем и т. д. Например, сигнал окончания переходного процесса может быть выработан с помощью встроенной задержки, включенной параллельно с блоком, самосинхронного проектирования или стартовых локальных часов, управляемых счетчиком синхросигналов. Возможно также применение синхростратумов второго и последующих уровней.

В дальнейшем нас не будет интересовать вопрос о том, как построены блоки системы. Важным является лишь наличие сигналов инициации и завершения переходного процесса в блоке.

В работе [1] показана возможность построения синхростратумов глобально асинхронных, локально произвольных (ГАЛП) систем для произвольных синхронных систем-прототипов, использующих двухтрековую (двухпроводную, двухтактную) двухфазную синхронизацию. Предлагаемая статья посвящена описанию способов перехода от других систем синхронизации прототипов к спецификации синхростратумов, осуществляющих глобальную синхронизацию блоков систем, и построению соответствующих эффективных реализаций синхростратумов.

Постановка задачи проектирования синхростратума в глобально асинхронных системах

Будем рассматривать систему параллельно работающих блоков. Блоки выполняют функции приема, хранения, обработки и выдачи данных. В соответствии с существующими в настоящее время принципами и технологиями проектирования цифровых устройств будем считать, что все блоки системы могут быть представлены синхронным автоматом либо сетью синхронных автоматов. Блок может содержать устройство обработки данных (в том числе процессор с программным управлением), порты для приема и выдачи данных при информационных обменах с другими блоками, контроллеры управления обменом, реализующие определенный протокол. Все устройства внутри блока синхронизируются от одного источника. Но источники синхронизирующих импульсов у разных блоков могут быть разными. То есть будем рассматривать локально синхронную, но глобально асинхронную систему.

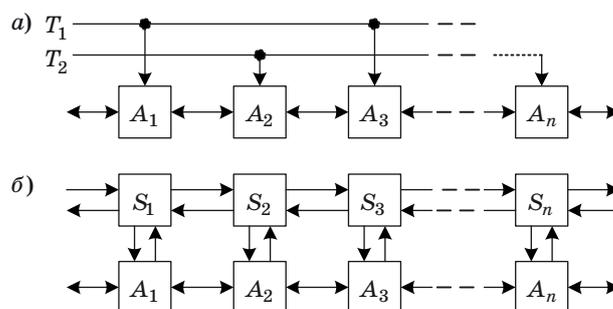
Для координации работы блоков в системе будем использовать выделенную для этого подсистему — синхростратум [1]. Синхронизируемые блоки системы образуют подсистему, называемую процессорным стратумом. Блоки процессорного стратума имеют между собой связи по данным и управлению. Топология связей в сети может быть различной и зависит от назначения системы и способа параллельной обработки данных, выбранного при разработке архитектуры системы. Процессорный стратум имеет связи и с внешней средой.

Декомпозиция системы на процессорный стратум и синхростратум позволяет разделить задачи проектирования процессорного стратума, решающего задачи собственно обработки данных, и синхростратума, решающего задачи координа-

ции во времени параллельно работающих блоков. Будем рассматривать задачу проектирования синхростратума как устройства, реализующего автоматное управление асинхронными процессами в системе. Моделью устройства является асинхронный автомат в классе самосинхронных схем [7, 8].

Различные подходы к проектированию синхростратума можно рассмотреть на примере системы с простейшей топологией сети связей — цепочке (одномерном массиве) взаимодействующих автоматов. При анализе задачи проектирования синхростратума будем рассматривать синхронный прототип и соответствующий вариант системы с асинхронным синхростратумом. На рис. 1, а показан пример синхронной системы с одномерным массивом автоматов, которая используется в качестве синхронного прототипа; на рис. 1, б — соответствующая асинхронная система с выделенным в отдельную подсистему синхростратумом.

У синхронных систем основным принципом организации взаимодействия блоков является выделение двух фаз в работе каждого блока. Взаимодействующие блоки разделяются на «ведущие» и «ведомые», или «активный» и «пассивный». В первой фазе ведущий блок воспринимает информацию от соседей, которые в этой фазе являются ведомыми, и в соответствии с заложенным в него алгоритмом изменяет свое состояние. Блоки, находящиеся в состоянии «ведомый», не меняют состояние своих выходов. Во второй фазе ведущий блок становится ведомым, а ведомый — ведущим. Переходные процессы в ведущем блоке не могут повлиять на состояние ведомого блока. Ведомый в это время не восприимчив к изменениям входных сигналов. При такой организации взаимодействия вариации длительности переходных процессов в блоке, если они не превышают длительность фазы, не оказывают влияния на работоспособность системы. Разделение физического времени, в котором работают все блоки (явля-



■ Рис. 1. Одномерный массив автоматов: а — с двухтактной двухфазной синхронизацией; б — с синхронизацией от синхростратума

ющиеся динамическими системами при рассмотрении их на уровне физических сигналов), на фазы реализуется в синхронных системах с помощью внешнего генератора синхронизирующих импульсов (внешних часов). Это простейший механизм связывания логического и физического времени [1].

Существенно отметить, что в синхронной системе-прототипе (см. рис. 1, а) имеются два уровня взаимодействий, использующих принцип «ведущий-ведомый». Нижний уровень — это двухфазная организация работы самих синхронных автоматов. На этом уровне принцип «ведущий-ведомый» реализуется в памяти синхронного автомата, построенной на триггерах, синхронизируемых фронтом и спадом тактовых импульсов [1].

Верхний уровень — это двухтактная синхронизация взаимодействия автоматов в массиве двумя последовательностями импульсов T_1 и T_2 . Импульсы T_1 и T_2 изменяются во времени следующим образом:

$$(T_1, T_2) = (0, 0) \rightarrow (1, 0) \rightarrow (0, 0) \rightarrow (0, 1) \rightarrow (0, 0) \rightarrow (1, 0) \rightarrow \dots$$

Следует отметить, что мы рассматриваем последовательность (T_1, T_2) в синхронной системе как прототип для асинхронного синхростратума. При этом существенны такие детали, как то, что не могут одновременно изменить свое значение два сигнала T_1 и T_2 (поскольку понятие одновременности связано с точностью измерения). Поэтому в последовательности (T_1, T_2) присутствуют коды $(0, 0)$, что для работы синхронной системы, реагирующей на события в виде перепадов значений T_1 и T_2 , не существенно.

Система синхронизации разбивает множество автоматов $\{A_1, A_2, A_3, \dots\}$ на два подмножества: подмножество нечетных автоматов $\{A_1, A_3, A_5, \dots\}$ и подмножество четных автоматов $\{A_2, A_4, A_6, \dots\}$. Нечетные синхронизируются последовательностью T_1 , а четные — T_2 . При $(T_1, T_2) = (1, 0)$ нечетные автоматы изменяют свое состояние в соответствии с реализуемыми в них функциями переходов. Аргументами функции переходов каждого автомата являются переменные, характеризующие текущее состояние автомата, а также переменные, представленные входными сигналами от соседних автоматов в массиве. Таким образом, в рассматриваемый интервал времени нечетные автоматы являются ведущими, а четные — ведомыми. При $(T_1, T_2) = (0, 1)$ ведущими становятся четные автоматы, а нечетные — ведомыми.

Для связи верхнего и нижнего уровня синхронизации в автоматах могут быть использованы

различные средства, например умножители частоты или внутренний синхростратум. Единственным требованием является организация взаимодействия синхростратума с блоками системы по принципу «запрос-ответ». В дальнейшем нас будет интересовать только верхний уровень синхронизации.

Для большей наглядности в последующих рассуждениях для подмножества нечетных автоматов будем использовать обозначение $\{A_i\}$, а для подмножества четных автоматов — обозначение $\{B_j\}$.

Автоматы объединены в цепь (см. рис. 1, а) в целях совместной реализации некоторого алгоритма. Каждый k -й шаг алгоритма, реализуемого цепью из автоматов, выполняется за два последовательных такта T_1, T_2 . При этом синхронная работа цепи автоматов может быть представлена в виде фрагмента сигнального графа (СГ)

$$\begin{aligned} \dots \rightarrow +T_1(k) \rightarrow \{+a_i(k)\} \rightarrow -T_1(k) \rightarrow \{-a_i(k)\} \rightarrow \\ \rightarrow +T_2(k) \rightarrow \{+b_j(k)\} \rightarrow -T_2(k) \rightarrow \\ \rightarrow \{-b_j(k)\} \rightarrow +T_1(k+1) \rightarrow \{+a_i(k+1)\} \rightarrow \\ \rightarrow -T_1(k+1) \rightarrow \{-a_i(k+1)\} \rightarrow +T_2(k+1) \rightarrow \dots, \end{aligned}$$

где обозначены следующие события:

- $+T_1(k)$ — фронт импульса T_1 на k -м шаге алгоритма;
- $+a_i(k)$ — завершение переходного процесса в автомате A_i при $T_1=1$;
- $-T_1(k)$ — спад импульса T_1 на k -м шаге алгоритма;
- $-a_i(k)$ — завершение процесса в автомате A_i при $T_1=0$;
- $\pm T_2$ — фронт и спад импульсов T_2 ;
- $\pm b_j$ — завершение переходных процессов в автомате B_j ;
- $\{\pm a_i\}$ ($\{\pm b_j\}$) — завершение переходных процессов во всех нечетных (четных) автоматах.

Изменения значений синхросигналов T_1 и T_2 могут происходить лишь после завершения переходных процессов во всех автоматах, инициированных предыдущим изменением значения одного из этих сигналов.

Алгоритм поведения системы определяется автоматными уравнениями, устанавливающими причинно-следственные отношения между событиями в автоматах и в их ближайших соседях по графу взаимосоединений, т. е. определяющими частичный порядок на событиях (в логическом времени).

По аналогии с синхронным прототипом для асинхронной системы (см. рис. 1, б) обозначим:

- T_{1i} и T_{2j} — сигналы запроса (сигналы инициации) автоматов A_i и B_j соответственно;
- $\{T_{2j}[A_i]\}$ — сигналы запроса для подмножества автоматов B_j , являющихся непосредственными соседями автомата A_i в графе взаимосоединений;

$\{T_{1i}[B_j]\}$ — сигналы запроса для подмножества автоматов A_i , являющихся непосредственными соседями автомата B_j в графе взаимосоединений;

a_i и b_j — сигналы ответов от автоматов A_i и B_j соответственно.

Будем говорить, что СГ поведения синхростратума специфицирует глобальную параллельную синхронизацию асинхронной системы, соответствующей синхронному прототипу, если удовлетворяются следующие два условия:

1) сигнальный граф синхронного прототипа гомоморфен СГ синхростратума относительно отображения $\{\pm T_{lj}(k)\} \rightarrow \{\pm T_l(k)\}$, $l \in \{1, 2\}$;

2) сигнальный граф синхростратума соответствует отношению следования между событиями в каждом автомате и в его непосредственных соседях.

Соответственно получим фрагмент СГ

$$\begin{aligned} +T_{1i}(k) \rightarrow +a_i(k) \rightarrow -T_{1i}(k) \rightarrow -a_i(k) \rightarrow \{+T_{2j}[A_i](k)\}; \\ +T_{2j}(k) \rightarrow +b_j(k) \rightarrow -T_{2j}(k) \rightarrow \\ \rightarrow -b_j(k) \rightarrow \{+T_{1i}[B_j](k+1)\}. \end{aligned}$$

Заметим, что такая формализация асинхронной глобальной синхронизации справедлива не только для одномерного массива автоматов, но и для массива с взаимосоединениями, соответствующими произвольному графу Кёнига [1].

Спецификация и реализация синхростратума с использованием синхронного прототипа с двухтактной двухфазной синхронизацией

Синхростратум является самосинхронной схемой. Методы и средства проектирования синхронных и самосинхронных схем имеют существенные различия. К сожалению, в рамках данной статьи невозможно изложить процесс проектирования самосинхронных схем. В настоящее время достаточно развиты теория самосинхронных схем, методы их проектирования и системы автоматизации проектирования (например, PETRIFY, FORCAGE и др.). В качестве языков спецификации самосинхронных схем обычно используют сигнальные графы, диаграммы изменений и сети Петри. Покажем здесь лишь ход проектирования синхростратумов, отметим основные особенности, возникающие при постановке и решении частных задач, приведем конечные результаты.

В синхронном прототипе выделяется ряд сигналов, с помощью которых можно корректно описать порядок срабатываний синхронизируемых блоков (координацию их взаимодействия). На этих сигналах строится сигнальный граф, который является исходной спецификацией синхро-

стратума. Как правило, непосредственно по этому СГ невозможно построить схему, поскольку он содержит некорректности.

В теории проектирования синхронных автоматов рассматриваются некоторые явления, связанные с разбросом времени распространения сигналов по сети элементов. Это риски сбоя — кратковременные «ложные» сигналы, не предусмотренные логическим проектированием. Методика проектирования предусматривает исключение влияния этих ложных сигналов на переходы автомата из одного состояния в другое. Как уже отмечалось, для этого используются организация памяти автомата по принципу «ведущий-ведомый» и внешние часы.

В асинхронных автоматах такого универсального механизма нет. Поэтому при проектировании асинхронного автомата необходимо исключать ситуации, приводящие к появлению ложных сигналов. Это используется, в частности, при проектировании синхронных триггеров, синхронизируемых перепадом уровня синхросигнала. Такой триггер строится на базе асинхронных триггеров. На входах асинхронных триггеров не должно быть ложных сигналов. При проектировании систем, содержащих сеть взаимодействующих автоматов, модели асинхронных автоматов не прижились по двум причинам.

1. Асинхронные автоматы тактируются изменением состояния входа, и новый такт может поступить лишь после завершения в автомате всех переходных процессов, вызванных предыдущим тактом. В этом смысле временное поведение асинхронных автоматов отличается от поведения синхронных только тем, что длительность такта не постоянна.

2. Методы проектирования асинхронных автоматов и получающиеся решения во много раз сложнее, чем синхронных, из-за необходимости борьбы с состязаниями и применения противочасового кодирования.

В самосинхронных автоматах (новом понимании асинхронных автоматов) используется так называемая согласованная модель автомата и среды, в которой автомат вырабатывает метки времени для среды, а среда вырабатывает метки времени для автомата (метки времени фактически являются сигналами окончания переходных процессов). В самосинхронных автоматах, как и в синхронных, нет необходимости бороться с состязаниями сигналов. Но при этом появляется новая проблема — выработка сигналов окончания переходных процессов. При ее решении в процессе разработки спецификации на самосинхронный автомат в виде СГ с использованием синхронного прототипа и могут возникнуть некорректности следующих типов.

1. Сигнальные графы являются подклассом сетей Петри. Функционирование СГ, как и сети Петри, связано с распространением маркеров. Каждая возможная комбинация маркеров в СГ связана с набором значений сигналов на выходах элементов схемы, обозначенных соответствующими сигналами. Набор значений этих сигналов представляет состояние схемы. Любой СГ можно развернуть в граф смены состояний, который обычно называют диаграммой Малера. Появление одинаковых состояний в диаграмме Малера называется конфликтом. При возникновении конфликта нарушается однозначное соответствие маркировки СГ и состояний схемы, что недопустимо. Например, если в СГ изменения одного и того же сигнала непосредственно следуют друг за другом или после некоторой последовательности значений различных сигналов непосредственно следует та же самая последовательность сигналов с противоположными значениями, то в этих случаях всегда возникает конфликт [8].

2. Все сигналы СГ должны быть корректны по переключениям. Это означает, что в любой последовательности сигналов не должны встречаться две копии сигнала с одним и тем же знаком, между которыми отсутствует тот же сигнал с противоположным знаком. Кроме того, в параллельных ветвях (не альтернативных) СГ не должен встречаться один и тот же сигнал с любыми знаками.

Некорректности можно обнаружить как по виду СГ, так и с использованием диаграмм Малера, по которым строятся таблицы истинности логических функций [8].

Некорректности устраняются расширением СГ с помощью введения дополнительных сигналов. Существуют методы, позволяющие привести исходное описание к виду, пригодному для реализации, и обеспечивающие введение минимального количества дополнительных сигналов (переменных). При этом можно получить несколько вариантов корректного СГ. Формальные методы синтеза графа существуют и реализованы, однако достижение минимальной сложности реализации — это творческий процесс, поскольку минимальное описание не всегда приводит к минимальной схеме.

Далее корректный СГ переводится в диаграмму Малера, по которой строятся таблицы истинности функций сигналов и находятся сами функции. Это первый этап логического синтеза. Далее идет процесс схемотехнической реализации полученных функций. Если каждая из этих функций реализуется одним из базисных элементов, то самосинхронная схема построена. (Имеются в виду базисные элементы самосинхронных схем [8].) В противном случае требуется снова вводить

промежуточные переменные и повторять всю процедуру синтеза.

К сожалению, обычные методы представления таких функций в виде суперпозиции функций базисных элементов непригодны, так как получающиеся схемы, как правило, не являются схемами, не зависящими от задержек элементов. Таким образом, процесс синтеза схемы, реализующей получаемые при логическом синтезе блоков синхростратума функции, требует особого подхода, основанного на теории самосинхронных схем и отличного от средств и методов синтеза комбинационных схем в синхронных автоматах.

Рассмотрим проектирование синхростратума для асинхронной системы (см. рис. 1, б). Примем для примера, что массив содержит 8 автоматов.

Обозначим:

T_j — сигнал инициализации работы автомата A_j , формируемый блоком S_j синхростратума;

P_j — сигнал ответа автомата A_j о завершении в нем инициированных процессов; сигнал поступает из A_j в S_j .

Сигнальный граф, построенный непосредственно на этих сигналах, не может быть реализован самосинхронной схемой, так как содержит некорректности. Для их исключения требуется введение локальных промежуточных переменных.

В работе [1] приведен правильный СГ для параллельной синхронизации рассматриваемого нами одномерного массива из 8 автоматов. В этом графе для каждого модуля синхростратума S_j введены две промежуточные переменные X_j и Y_j .

С использованием опыта проектирования построен еще один вариант корректного СГ (рис. 2).

В нем введено только по одной промежуточной переменной h_j в каждый модуль S_j синхростратума. Это позволяет упростить систему связей между модулями синхростратума по сравнению с другим решением [1].

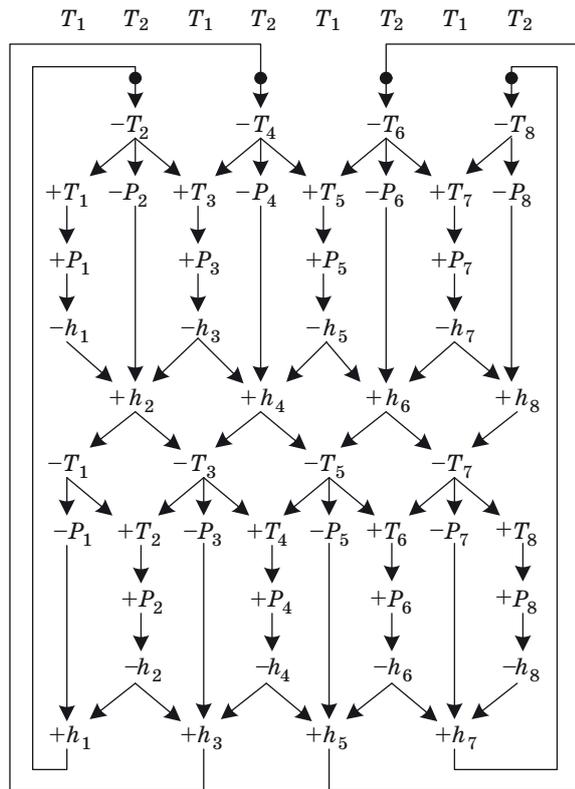
Строка над графом указывает на соответствие локальных сигналов синхростратума сигналам синхронизации T_1 и T_2 прототипа. В самом графе сигналы $\pm T_j$ представляют изменения выходных сигналов модулей S_j синхростратума, иницирующих фазы работы j -го автомата. Сигналы $\pm P_j$ представляют изменения сигналов ответа j -го автомата; сигналы $\pm h_j$ соответствуют изменениям локальных промежуточных переменных h_j .

По СГ на рис. 2 получены следующие собственные функции элементов (функции базисных элементов) модуля S_j синхростратума:

$$T_j = \overline{T_{j-1}} \vee h_{j-1} h_{j+1} \vee T_{j+1};$$

$$h_j = h_{j-1} \vee P_j \vee h_{j+1}.$$

Таким образом, каждый блок S_j синхростратума имеет 2 выхода, на которых формирует сиг-

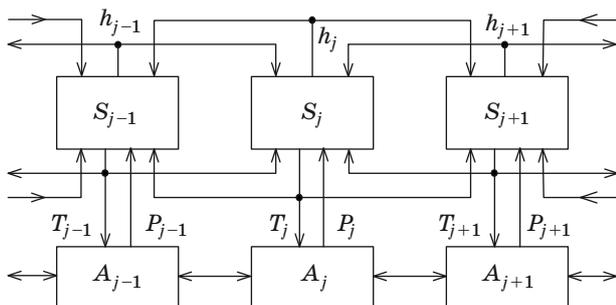


■ Рис. 2. Сигнальный граф для двухтактной двухфазной синхронизации массива из 8 автоматов

налы T_j и h_j , и 5 входов, на которые поступают сигналы: $P_j, h_{j-1}, h_{j+1}, T_{j-1}, T_{j+1}$. Соответствующая структура показана на рис. 3.

Просматривая пути по графу рис. 2, можно определить, что синхростратум добавляет задержку $D_{ss} = 6\tau_{gate}$ (τ_{gate} — задержка одного вентиля) ко времени цикла работы автомата.

Из графа рис. 2 можно также увидеть, что заданный порядок следования событий допускает параллельную работу нечетных и четных автоматов в противоположных фазах синхронизирующих сигналов. Например, рассмотрим строку графа, соответствующую одному моменту логического времени: $+T_1, -P_2, +T_3, -P_4, +T_5, -P_6, +T_7$,



■ Рис. 3. Структура синхростратума в асинхронной системе, соответствующая СГ на рис. 2

$-P_8$. В этот момент нечетные автоматы находятся в фазе «активен», а четные — «пассивен». Такая организация работы системы увеличивает ее производительность.

Спецификация и реализация синхростратума с использованием синхронного прототипа с четырехтактной синхронизацией

Поскольку синхростратум (см. рис. 3) является самосинхронной схемой, его ячейки взаимодействуют друг с другом по принципу «запрос-ответ», т. е. имеют четыре фазы работы: запрос, ответ, снятие запроса, снятие ответа. По этой причине можно сделать предположение, что синхростратум, построенный по прототипу синхронной системы с четырехтактной синхронизацией, будет обладать наименьшей сложностью.

Представим себе четырехтактную синхронизацию, осуществляемую с помощью четырех последовательностей синхросигналов T_1, T_2, T_3 и T_4 , в которых каждая пара импульсов смежных последовательностей перекрывается на половину длительности тактового импульса. Эти синхросигналы изменяются во времени следующим образом:

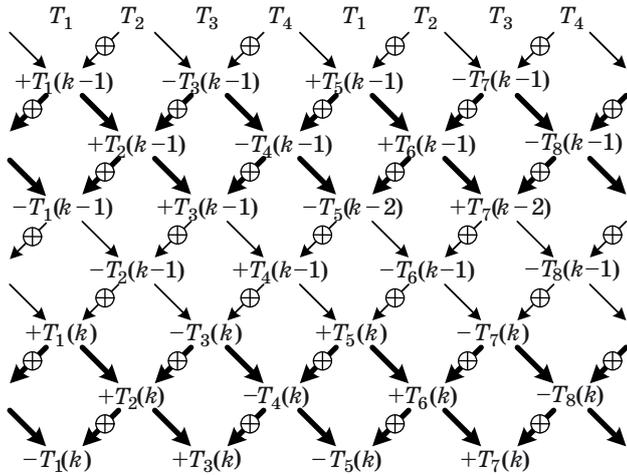
$$(T_1, T_2, T_3, T_4) = (1, 0, 0, 1) \rightarrow (1, 1, 0, 0) \rightarrow (0, 1, 1, 0) \rightarrow (0, 0, 1, 1) \rightarrow (1, 0, 0, 1) \rightarrow \dots$$

Возможность построения синхронной системы прототипа, представляющей одномерный массив взаимодействующих автоматов (см. рис. 1, а), с использованием внешнего генератора указанной последовательности тактовых импульсов не вызывает сомнений.

Для четырехтактной параллельной синхронизации одномерного массива автоматов можно построить СГ, являющийся спецификацией соответствующего синхростратума. На рис. 4 представлен фрагмент развертки такого СГ.

Дадим некоторые пояснения по составлению этого графа. Взаимодействие блока синхростратума S_i с соседними блоками и автоматом A_i (см. рис. 1, б) осуществляется по принципу «запрос-ответ». При этом допускается любая конечная задержка между запросом и ответом, включая нулевую. Рассмотрим вначале спецификацию на синхростратум для случая, когда ответ автомата на запрос приходит без задержки. Это означает, что $P_i = T_i$. Сигналы P_i можем не рассматривать.

В строке над графом на рис. 4 указано соответствие выходных сигналов синхростратума T_j , которые инициируют работу автоматов, синхросигналам прототипа $T_1 - T_4$. Сигналы $\pm T_j(k)$ — изменения T_j в моменты k логического времени. Мы



■ Рис. 4. Развертка сигнального графа для четырехтактной параллельной синхронизации

рассматриваем проектирование синхростратума из 8 автоматов, поэтому $j \in \{1, 2, \dots, 8\}$. Следуя синхронному прототипу, будем считать, что поведение сигналов T_5, T_6, T_7, T_8 в логическом времени полностью совпадает с поведением сигналов T_1, T_2, T_3, T_4 .

При четырехтактной синхронизации для реализации одного шага алгоритма цепочкой автоматов в синхронном прототипе требуется 4 такта. В асинхронной системе реализации $(k-1)$ -го шага алгоритма соответствуют 4 строки событий на рис. 4. Нечетные строки соответствуют изменениям нечетных сигналов, четные — изменениям четных сигналов. Условием изменения сигнала T_j служат события в соседних блоках синхростратума, чему соответствуют дуги графа.

Нетрудно заметить, что 7 строк событий, показанных на графе, соответствуют следующей последовательности вырабатываемых синхростратумом сигналов (T_1, T_2, \dots, T_8):

$$(10011001) \rightarrow (11001100) \rightarrow (01100110) \rightarrow \\ \rightarrow (00110011) \rightarrow (10011001) \rightarrow (11001100) \rightarrow \\ \rightarrow (01100110),$$

представляющей поведение синхронного прототипа, при этом граф не содержит некорrekтностей. Дополнительные переменные и связи в синхростратуме вводить не требуется. Существенно отметить, что при полном соответствии поведения синхронного прототипа и асинхронной системы в логическом времени в физическом времени они ведут себя по-разному.

Символом \otimes маркированы связи, в разрывы которых вставлены автоматы A_j . Вход стрелки в символ \otimes соответствует входу T_j автомата A_j . Выход стрелки из символа \otimes соответствует выхо-

ду P_j автомата A_j . Жирные стрелки, направленные в вершины $+T_j(k)$, определяют условия изменения состояний j -х автоматов в первой фазе их работы; тонкие стрелки, направленные в вершины $-T_j(k)$, определяют условия для запоминания измененных состояний в j -х автоматах во второй фазе их работы.

Из графа рис. 4 следует, что модуль синхростратума реализуется C -элементом с автоматным уравнением

$$T'_j = T_{j-1} \bar{P}_{j+1} \vee T_j (T_{j-1} \vee \bar{P}_{j+1}),$$

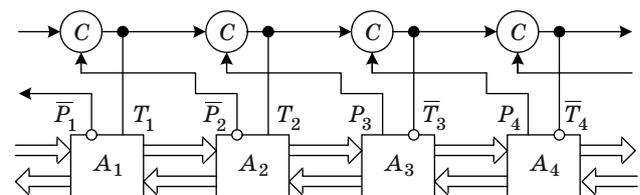
где \bar{P}_{j+1} — сигнал ответа $(j+1)$ -го автомата.

Структура одномерного массива с таким синхростратумом показана на рис. 5.

Вершины графа на рис. 4, в которых изменяются моменты логического времени, вызывают переходы в автоматах. Нетрудно видеть, что смена состояний автоматов с номерами $j = 4l + 1$ и $j = 4l + 2, l = 0, 1, 2, \dots$, осуществляется при $T_j = 1$, а автоматов с номерами $j = 4l + 3$ и $j = 4l + 4, l = 0, 1, 2, \dots$, — при $T_j = 0$. Поэтому для второй пары автоматов в схеме на рис. 5 изменена полярность управляющих сигналов. Заметим, что если тот же синхростратум использовать для волновой синхронизации одномерного массива автоматов [1], то все переходы автоматов осуществляются при $T_j = 1$, т. е. полярность тактирующих сигналов для всех автоматов одинакова.

В структуре рис. 5 дополнительная задержка, вносимая синхростратумом в работу автоматов, равна $D_{ss} = 4\tau_c = 8\tau_{gate}$, где τ_c — задержка C -элемента. Сложность модуля синхростратума определяется сложностью реализации C -элемента (от 8 до 12 транзисторов в КМОП-технологии) и числом проводов, соединяющих его с другими модулями (в данном случае их два).

Последняя схема синхростратума отличается от ранее описанных схем здесь и в работах [1–6] асимметрией взаимодействия ее модулей и способом подключения управляемых ею автоматов. Автоматы A_j вставляются в разрывы проводов сигналов T_j после разветвления (тем не менее, схема остается независимой от задержек в проводах). Асимметрия взаимодействия модулей опре-



■ Рис. 5. Одномерный ГАЛП-массив автоматов для синхронного прототипа с четырехтактной двухфазной синхронизацией

деляет направление распространения синхронизирующих сигналов, и синхростратум является асинхронным конвейером. Это различие становится заметным, если степень вершин графа взаимосоединений автоматов больше, чем два.

В случае произвольного графа взаимосоединений между автоматами при переходе от четырехтактной синхронизации к спецификации синхростратума требуется раскрасить этот граф в четыре цвета. В однородных массивах это сделать достаточно легко, в то время как в произвольном графе взаимосоединений могут возникать конфликты и такое разбиение автоматов на подмножества может оказаться сложной задачей. В этом смысле двухтактная синхронизация предпочтительнее, так как она требует раскраски графа взаимосоединений автоматов только в два цвета, что равносильно построению двудольного графа Кёнига по произвольному графу. Это может быть сделано достаточно просто [1].

Заметим, что при четырехтактной синхронизации пары синхропоследовательностей (T_1, T_3) и (T_2, T_4) являются двухтактными, т. е. можно снова перейти к двухтактной синхронизации автоматов, если использовать для координации взаимодействия автоматов только те выходы синхростратума, которые соответствуют, например, последовательностям T_2 и T_4 . Отсюда следует, что модуль синхростратума для двухтактной синхронизации может быть построен из двух модулей синхростратума для четырехтактной синхронизации, т. е. из двух C -элементов и одного инвертора, как это показано на рис. 6, а.

Преимуществом такой схемы модуля синхростратума является минимум связей с соседними модулями, что в ряде случаев имеет первостепенное значение, например при построении саморемонтирующихся отказоустойчивых систем.

При параллельной синхронизации одномерного массива автоматов синхростратум представляет собой генерирующую структуру. Для осуществления генерации с помощью начальной установки в каждый нечетный модуль записывается со-

стояние $C_1 = C_2 = 1$, а в каждый четный — состояние $C_1 = C_2 = 0$, и синхростратум замыкается в кольцо. При этом необходимо, чтобы кольцо имело четное число модулей.

Очевиден и другой способ организации генерации синхростратумом управляющих сигналов. Вместо замыкания его в кольцо можно организовать генерирующую схему на входе синхростратума, тогда условие четности числа его модулей снимается.

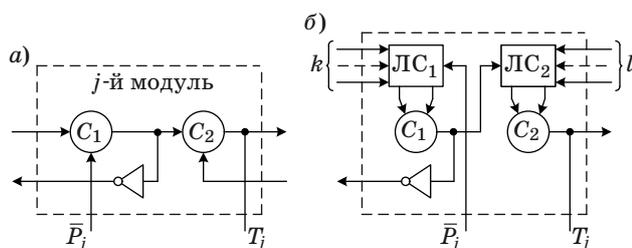
На основе модуля синхростратума для линейного массива автоматов нетрудно построить модуль синхростратума для координации взаимодействия автоматов со структурой межсоединений в виде произвольного графа Кёнига. Структура такого модуля, соответствующего вершине графа с k входными дугами и l выходными дугами, показана на рис. 6, б. Она содержит два двухвходных C -элемента — C_1 и C_2 , инвертор и две логические схемы: $ЛС_1$ осуществляет сжатие $k + 1$ сигналов запроса в два выходных сигнала [8], а $ЛС_2$ сжимает в два выходных сигнала $l + 1$ сигналов ответа.

Еще раз подчеркнем, что синхростратум, построенный из таких модулей в соответствии с произвольным графом Кёнига, может использоваться как для параллельной синхронизации (координации взаимодействия автоматов), так и для волновой. Отличие заключается лишь в назначении моментов логического времени и во внутреннем устройстве (функциях переходов) управляемых автоматов.

Заключение

Конечно, проблема глобальной асинхронной координации (синхронизации) блоков вычислительных и управляющих устройств не закрыта. Предложенный здесь подход, как и любое общее решение, хорош для всех случаев, но может оказаться не лучшим в каждом конкретном случае. Например, использование этого подхода для построения однобитного конвейерного регистра дает более сложные схемы, чем известные реализации асинхронных однобитных конвейеров. Поэтому очень важно понимать, в каких случаях можно ожидать положительных результатов.

На основании проведенных исследований можно сделать заключение, что для любой системы, представляющей собой массив блоков (реализующих информационные процессы) с межблочными соединениями, соответствующими произвольному графу Кёнига, можно спроектировать систему асинхронной глобальной синхронизации. При этом синхронная параллельная система-прототип может иметь любую систему синхронизации от общих часов.



■ Рис. 6. Модуль синхростратума для двухтактной синхронизации прототипа: а — для одномерного массива; б — для массива со структурой межсоединений в виде графа Кёнига

Проектирование системы асинхронной глобальной синхронизации сводится к решению двух задач:

1) в блоках системы необходимо спроектировать средства формирования сигналов окончания иницированного процесса;

2) спроектировать синхростратум, соответствующий синхронному прототипу.

При обсуждении ГАЛП-методологии основное внимание было сфокусировано на «глобальной

асинхронности»; «локальная произвольность» подразумевает широкие возможности по организации взаимодействия между синхростратумом и процессорным стратумом. С этой целью могут быть использованы самосинхронные устройства, токовые сенсоры (в КМОП-схемах ток протекает только во время переходного процесса), старто-стопные локальные генераторы (часы), параллельные инкорпорированные задержки (включая задержки, зависящие от данных) и др.

Литература

1. **Мараховский В. Б., Мелехин В. Ф.** Проектирование средств синхронизации блоков глобально асинхронных систем с произвольной локальной синхронизацией // Информационно-управляющие системы. 2010. № 1. С. 29–38.
2. **Угрюмов Е. П.** Цифровая схемотехника. — СПб.: БХВ-Петербург, 2007. — 800 с.
3. **Varshavsky V. I., Chu T.-A.** Self-Timing — Tools for Hardware Support of Parallel, Concurrent and Event-Driven Process Control: Proc. of the Conf. on Massively Parallel Computing Systems (MPCS). May 1994. P. 510–515.
4. **Varshavsky V. I., Marakhovsky V. B., Chu T.-A.** Logical Timing (Global Synchronization of Asynchronous Arrays) // Parallel Algorithm / Architecture Synthesis: Intern. Symp., Aizu-Wakamatsu, Japan. IEEE CS Press, Mar. 1995. P. 130–138.
5. **Varshavsky V. I., Marakhovsky V. B., Chu T.-A.** Asynchronous Timing of Arrays with Synchronous Prototype: Proc. of the Second Intern. Conf. on Massively Parallel Computing Systems (MPCS'96), Ischia, Italy, May 1996. P. 47–54.
6. **Varshavsky V. I., Marakhovsky V. B.** Global Synchronization of Asynchronous Arrays in Logical Time // Parallel Algorithm / Architecture Synthesis: Proc. of the Second Intern. Symp., Aizu-Wakamatsu, Japan. IEEE CS Press, Mar. 1997. P. 207–215.
7. **Varshavsky V.** Hardware Support of Parallel Asynchronous Processes / University of Technology, Digital Systems Laboratory. — Helsinki, Sept. 1987. Iss. 2. Ser. A: Research Reports. — 235 p.
8. **Варшавский В. И.** и др. Автоматное управление асинхронными процессами в ЭВМ и дискретных системах / Под ред. В. И. Варшавского. — М.: Наука, 1986. — 308 с.