

УДК 681.3

СИСТЕМНОЕ ВРЕМЯ И СИНХРОНИЗАЦИЯ СИСТЕМ¹

В. И. Варшавский,

доктор техн. наук, профессор
университет Айдзу (Япония)

В статье проводится анализ концепции причинной обусловленности как основы правильного асинхронного поведения. Предлагается декомпозировать систему на процессорный стратум и синхро-стратум, функционирующий подобно распределенным асинхронным часам. Универсальный синхро-стратум обеспечивает возможность прямого перехода от синхронного прототипа к асинхронной системе. Приводятся несколько типов спецификаций и аппаратной реализации синхро-стратума вместе с их сравнительными характеристиками.

In the paper, the concept of causal conditionality is analyzed as the basis for correct asynchronous behavior. It is suggested to decompose the system to Processors Stratum and Synchro-Stratum which acts like a distributed asynchronous clock. A universal Synchro-Stratum provides the possibility of direct transition from the synchronous prototype to an asynchronous system. Several types of specification and hardware implementation of Synchro-Stratum are given with their characteristics compared.

Введение

Время является одним из наиболее сложных и плохо понимаемых объектов Природы, хотя большинство из нас полагают, будто отлично знают, что это такое. История, психология, искусство, микро- и макрофизика, космология и ряд других наук занимаются разными аспектами Времени и имеют на него, вообще говоря, разные точки зрения. Предметом этой статьи будет Время в искусственных системах, или Искусственное Время. Специфика этого предмета требует, по-видимому, специального рассмотрения. Прежде чем перейти к изложению основного содержания статьи, я позволю себе некоторые спекуляции на эту тему.

Начиная, вероятно, с Исаака Ньютона, достаточно точно сформулировавшего это понятие, Время рассматривается как независимая глобальная физическая переменная (Физическое Время). Все определения Физического Времени, с которыми нам удалось познакомиться, так или иначе связаны с процедурой измерения Времени² и понятием одновременности. Измерение, в свою очередь, всегда связано с его точностью, и одновременность является абстракцией в пределах точности измерительного прибора (часов).

Теория относительности ввела в рассмотрение понятие события и зависимость интервала измеренного времени между событиями от позиции наблюдателя. При этом, хотел ли этого Эйнштейн или нет, было введено понятие информационного обмена (сигнала о событии). Более того, событие по своей природе дискретно и возникает естественный вопрос, дискретно или непрерывно физическое время? С точки зрения базового определения, оно непрерывно. С другой стороны, оно гранулируется, по крайней мере, точностью измерительного прибора. Поскольку любой измерительный процесс требует расхода энергии и является вмешательством в измеряемый процесс, то соотношение неопределенности в его форме, связанной со временем, может дать информацию о минимальном кванте времени и, надо полагать, о возможной достижимой точности измерения³. Соотношение неопределенности в форме, связывающей энергию и время, имеет вид $\Delta E \Delta t \geq \hbar$. Это означает, что энергию взаимодействия между измерительным прибором и исследуемой системой можно учесть только с точностью до $\hbar / \Delta t$. Хотя сегодня микроэлектроника весьма далека от этого ограничения, но для FED⁴, использование ко-

¹Varshavsky V. System Time and System Timing. C. Nehaniv & M. Ito, eds., Algebraic Engineering, World Scientific Press, 1999. – P. 38–57.

²Это естественно, как для любой физической переменной. Измерительным прибором для Времени являются Часы.

³Честно говоря, на мои вопросы к физикам о смысле этой формы соотношения неопределенности я не получил вразумительного ответа. Возможно, я спрашивал не тех людей.

⁴Future Electron Device, таких как Single Electron Transistor Wave Function Rearrangement Devices Quantum Dot Cell и т. д.

торых ожидается во второй декаде следующего века, время смены энергетического уровня (состояния) на величину ΔE распределено по Пуассону с параметром $\tau = \hbar / \Delta E$ ¹.

Как это ни парадоксально, но определение Времени, связанное с последовательностью дискретных событий, было дано значительно раньше, еще Аристотелем. Согласно Аристотелю, Время – это способ упорядочивания событий, отражающий причинно-следственные отношения между ними. С этой точки зрения Время есть логическая абстракция (Логическое Время) и суть его отражается в гениальной фразе Аристотеля: «Время отсутствует, если ничего не происходит».

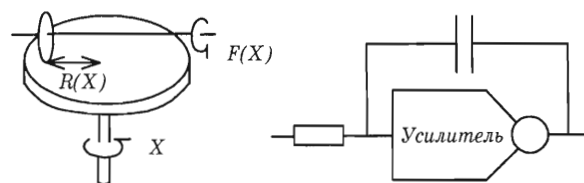
Оба определения, как физического, так и логического времени были сделаны применительно к естественному времени, связанному с независимыми от человека процессами, протекающими в Природе. Начиная с самых ранних периодов истории, у человека возникла необходимость в создании искусственных систем (устройств), поведение и использование которых было связано со временем. Во-первых, это часы. Функционирование всех известных часов связано с процессами, протекающими в непрерывном физическом времени. В большинстве случаев часы осуществляют преобразование непрерывного физического времени в последовательность дискретных событий. Во-вторых, это системы, устанавливающие логический порядок событий, как например, календари. Однако по-настоящему проблема времени в искусственных системах возникла при создании средств моделирования процессов, как аналитических, так и технических.

Аналитические непрерывные модели, такие как дифференциальные уравнения, включают непрерывное физическое время как независимую переменную без каких-либо специальных проблем, тем более, что и само определение независимого физического времени было в той или иной мере связано с созданием дифференциального исчисления.

Аналитические модели дискретных процессов, такие как уравнения в конечных разностях или конечные автоматы, оперируют с дискретным временем, вернее, с номерами шага процесса, которые в рамках этих моделей, на первый взгляд, никак не связаны с естественным временем. Но это справедливо только для изолированных систем, и как только мы попытаемся организовать или описать² взаимодействие системы с внешней средой, мы будем вынуждены заняться взаимодействием формального и естественного времени.

¹ При этом $\tau \approx 6 \times 10^{-4}$ пс/эВ. Для продвинутых микросистемных устройств характеристические энергии переключения имеют порядок FJ ($1 FJ \approx 600$ эВ) и Δt имеет порядок долей пикосекунды без учета энергии перезаряда паразитных емкостей.

² В рамках аналитических моделей это, по-видимому, одно и то же.



■ Рис. 1. Схемы интеграторов

Косвенно с этим связаны трудности перехода от абстрактной к структурной модели автомата и определения асинхронного автомата в классической структурной теории автоматов³.

Как только мы переходим к техническим системам, моделирующим те или иные процессы, мы с неизбежностью сталкиваемся с тем фактом, что физические устройства, реализующие техническую систему, функционируют в естественном физическом времени, а аналитические процессы, которые они моделируют, заданы в искусственном формальном времени. Таким образом, проблема введения времени в искусственную систему (Системное Время) и проблема организации взаимодействия между системным и физическим временем становится одной из проблем проектирования.

Впервые эта проблема в явном виде возникла в начале века при создании и использовании в режиме реального времени механических дифференциальных анализаторов (МДА). Одним из базовых элементов МДА является дисковый интегратор (рис. 1)⁴.

Если маленький диск радиуса r перемещается относительно центра большого диска на величину $R(X)$ и угол поворота оси большого диска при этом равен X , то угол поворота оси маленького

диска равен $F(X) = \int_0^X \frac{R(X)}{r} dX$. Как нетрудно видеть, поведение интегратора не зависит от времени и в качестве независимой⁵ может быть использована любая переменная. Если же мы хотим работать с физическим временем как независимой переменной, то для этого достаточно иметь постоянную скорость вращения ω оси большого диска.

При этом $F(t) = \int_0^t \frac{\omega \cdot R(t)}{r} dt$ и синхронный двига-

³ Ничего другого, кроме как «асинхронный автомат – это автомат, переходы которого инициируются изменением состояния входа и любой переход завершается стабильным состоянием», придумано не было, но каждый, кому пришлось читать студентам лекции по структурной теории автоматов, знает, сколь трудно, а порой и невозможно ответить на вопросы студентов: «Чем же все-таки отличается синхронный автомат от асинхронного?».

⁴ С его помощью при наличии следящей системы в цепи обратной связи реализуется и операция дифференцирования. Запаздывание в цепи обратной связи сказывается только на точности вычислений.

⁵ Или зависимой.

тель, обеспечивающий постоянную скорость вращения, играет роль общих для системы часов. Ситуация принципиально изменяется при переходе к электронным дифференциальным анализаторам. Их базовым компонентом является интегрирующий усилитель (рис. 1, б), поведение которого в явном виде зависит от физического времени, и значения всех переменных являются функцией времени. Теперь нам опять требуются часы для того, чтобы сопоставить значения различных переменных в один и тот же момент времени¹. Уже здесь наметились некоторые проблемы, связанные с тем, что усилитель обладает собственной задержкой и изменения распространяются по анализатору с некоторой задержкой. Хотя в большинстве применений этим можно было бы пренебречь, однако для моделирования очень быстрых процессов было необходимо прибегать к масштабированию времени².

История возникновения дискретных вычислительных средств начинается еще в древности³. Однако вопрос о времени, в котором функционируют соответствующие технические средства, возник значительно позже. Тем более, что значительно позже возникли формальные модели поведения таких устройств. Механические вычислительные устройства, такие как машины Паскаля, Лейбница, Однера, аналитическая машина Беббиджа, имели в качестве единицы системного времени однократный поворот ведущего вала⁴, который хотя и протекал в реальном физическом времени, но в процессе работы это фактически никак не учитывалось. Введение в начале XX века электрического привода⁵ никак не изменило ситуацию. Жесткость механического привода и зубчатых передач исключали ощутимую задержку передачи информации и, например, в зависимости от длины переноса при суммировании изменялась не задержка, а момент на ведущем валу⁶. В калькуляторах, где были автоматизированы операции умножения и деления, последовательность шагов вычисления задавалась последовательностью оборотов ведущего вала (системными часами).

Заметим как исторический курьез, что идея формирования шагов системного времени по готовности данных (то, что позже было названо «ар-

хитектурой потока данных») была впервые использована в XVIII веке в вычислительном бюро Прони, которое занималось составлением навигационных таблиц. Группа высококвалифицированных математиков, в которую входил Лагранж, разрабатывала методы вычислений (алгоритм) и систему таблиц, выходные данные которых были результатом одной арифметической операции над входными (программа). Оператор-вычислитель получал исходные таблицы и заполнял одну или несколько выходных таблиц, которые поступали следующему оператору. Оператор начинал очередной шаг вычислительного процесса, когда к нему поступали полные исходные данные для этого шага⁷.

Настоящие проблемы взаимодействия физического и системного (логического) времени возникли, по-видимому, с появлением и широким использованием релейных устройств в промышленной автоматике и системах сигнализации и блокировки на железных дорогах. Эти проблемы были связаны с опасностью возникновения гонок и состязаний, характер которых зависел от вариаций времен срабатываний контактов реле. Именно борьба с гонками и состязаниями впервые поставила задачу организации логического поведения, инвариантного к поведению компонент в физическом времени. Вызванные запросами практики работы Шестакова и Гаврилова (СССР), Шеннона (США), Накамуры (Япония) привели к созданию теории релейных устройств и структурной теории конечных автоматов (Мур, Мили, Хаффмен и др.). Появление электронных ламп и транзисторов до поры до времени не изменяло общую ситуацию.

Модель конечного автомата $S(t+1) = F(S(t), X(t+1)); Y(t+1) = \Phi(S(t), X(t+1))$ – модель Мили; $Y(t) = \Psi(S(t))$ – модель Мура – включала в себя параметр t как номер шага процесса, который не был связан с физическим временем и был уже, по сути, системным искусственным временем⁸. Именно попытки исключить параметр t из автоматных уравнений привели к понятию асинхронного автомата, для которого текущее состояние определяется устойчивым решением уравнения $S = F(S, X(t))$. При этом параметр t сохраняет свое значение нумератора последовательности состояний входа. Коль скоро модель асинхронного автомата была предложена и изучалась применительно к задачам реального проектирования, то сразу же возникли вопросы связи логического поведения автомата с физическим поведением его компонент, ответы на которые искали на логическом уровне (противогоночное кодирование и др.).

¹ Опять заметим, что одновременность достижима лишь в пределах точности часов.

² Масштабированное время есть не что иное, как искусственное системное время.

³ Вспомним камешки, зарубки на дереве, узелки, применявшиеся для счета и, наконец, счетные доски и абак. Заметим также, что математическое мышление древних людей было алгоритмично – в древнем Египте математическая задача считалась решенной, только если был найден алгоритм соответствующих вычислений на абаке.

⁴ Или однократное перемещение рычага в машине с пропорциональным рычагом.

⁵ Калькуляторы Reinmetall, Mercedes, Facit, Fenix и т. д.

⁶ Что могло быть компенсировано запасом мощности и изменяло не единицу системного времени, а ее длительность в реальном физическом времени.

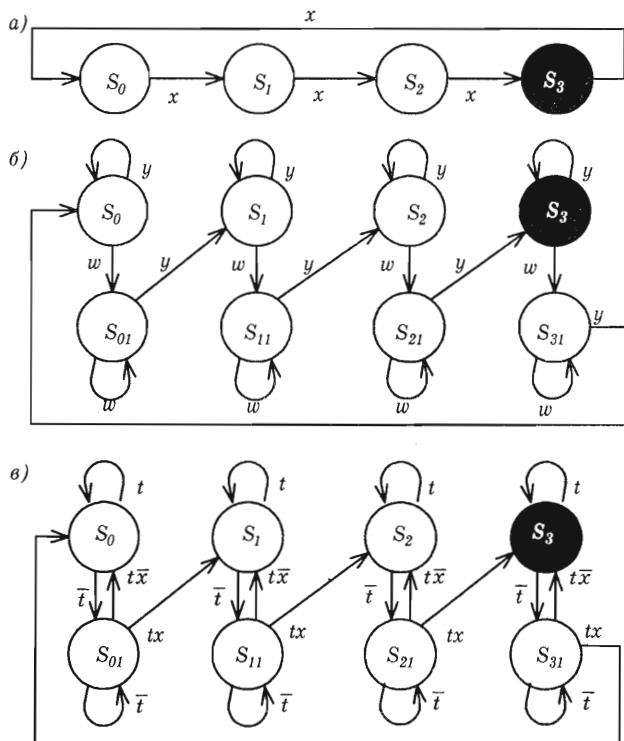
⁷ Чем не мультипроцессорная data-flow computer system?

⁸ Обратим внимание на содержательную близость модели конечного автомата и разностного уравнения.

Теория алгоритмов (нормальные алгоритмы Маркова, машины Тьюринга и Поста и др.) и абстрактная теория автоматов (регулярные события Клини и др.) развивались, также игнорируя реальное время. Системное время в этих моделях рассматривалось как шаг алгоритма или номер позиции символа во входном слове. Однако уже здесь стали очевидными некоторые противоречия между структурной и абстрактной теорией автоматов.

Действительно, например, по регулярному событию $xxxx(xxxx)^*$, используя прямой алгоритм МакНотона-Ямады, можно построить представляющий это событие конечный автомат (рис. 2, а)¹. С точки зрения абстрактной теории, все это абсолютно корректно, но, с точки зрения структурной теории, такой автомат является не счетчиком по модулю 4, а генератором. Для обеспечения корректной работы автомата мы должны ввести либо пробел (спейсер), разделяющий два последовательных символа x , либо сигнал смены символа t . Тогда, чтобы обеспечить введение системного времени и корректную работу автомата, мы должны изменить входной алфавит и регулярное событие, представление которого есть счет по модулю 4.

Пусть $y = x \& t$, $w = \bar{x} + \bar{t}$ (синхронный вариант) или $y = x$, $w = \bar{x}$ (асинхронный вариант) и со-



■ Рис. 2. Введение времени в граф автомата

¹ Счетчик по модулю 4.

ответствующее регулярное событие имеет вид $ywywywyw(ywywywyw)^*$. Изменение регулярного события приводит к изменению распознающего это событие конечного автомата (рис. 2, б)².

Несмотря на значительное число работ по асинхронным автоматам и несомненные успехи теории, наиболее простым способом введения системного времени³ все-таки на много лет стало использование внешних часов (синхронная реализация). При этом для рассмотренного примера реализация конечного автомата по принципу работы «основной-вспомогательный» задавалась графом, приведенным на рис. 2, в.

Использование внешних часов или, точнее говоря, сигнала от внешних часов в качестве дополнительной глобальной переменной, формирующей шаги процесса смены дискретных состояний устройства, имеет простую структуру (рис. 3, а).

Часы представляют собой периодический калибровочный процесс, который делит процессы, протекающие в непрерывном физическом мировом времени в среде и устройстве, на дискретные шаги. При этом должно действовать соглашение о том, что и в среде, и в устройстве длительность фазовых переходов из одного дискретного состояния в другое не превосходит⁴ длительности периода работы часов. В рамках приведенной структуры часы представляют собой не что иное, как калиброванную задержку. Важно при этом отметить, что события в часах не имеют причинной связи ни с событиями в среде, ни с событиями в устройстве. Все три объекта имеют, среди прочих, только одну общую причину происходящих в них событий – течение мирового физического времени. Заметим на будущее, что отсутствие причинно-следственных отношений между указанными событиями или, другими словами, нарушение причинной обусловленности поведения является одним из источников неприятностей, с которыми сталкивается синхронная реализация.

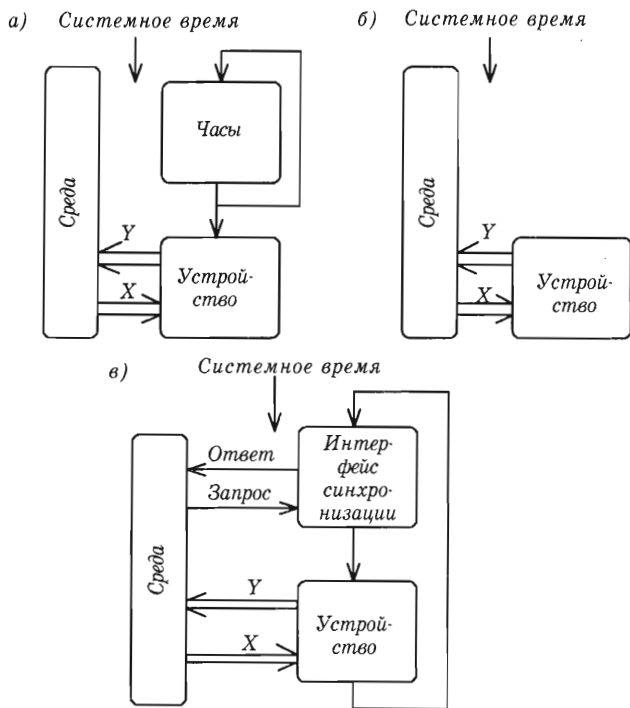
В классической асинхронной реализации (рис. 3, б) инициатором переходного процесса в устройстве является внешняя среда⁵. При этом, опять-таки, поведение во времени внешней среды в значительной мере причинно независимо от поведения устройства. В тех случаях, когда внешняя среда является частью системы, а часто так оно и есть, требуется постановка задачи интерфейса между системным и естественным временем и ее решение. Од-

² Мы получим тот же самый автомат, если осуществим подстановку yy^* и ww^* вместо y и w соответственно, но это создает лишь иллюзию асинхронности в рамках языка регулярных событий.

³ Или, точнее говоря, преобразования физического времени в системное.

⁴ Более корректно, в пределах точности поддержания периода калибровочного процесса, строго меньше.

⁵ Изменение состояния входного вектора X . Заметим при этом, что и в синхронной модели часы фактически являются для устройства частью внешней среды и сигнал времени T может быть включен во входной вектор – $X' = \{X, T\}$.



■ Рис. 3. Взаимодействие устройства и среды во времени

нако, если внешняя среда или существенная ее часть является искусственной системой, мы вправе попытаться ввести в интерфейс между ними причинную обусловленность. Эту задачу решает так называемая «согласованная реализация» (рис. 3, в).

Суть согласованной реализации, предложенной в пионерских работах Д. Маллера, состоит в том, что внешняя среда навязывает ритм смены состояний устройству, а устройство – внешней среде. При этом, по крайней мере, один выход устройства изменяется только после завершения очередного переходного процесса в нем. Развитие модели Маллера привело к созданию нового класса устройств с причинно обусловленным поведением – самосинхронных (не зависящих от скорости, нечувствительных к задержкам) устройств.

**Модель Маллера [5, 6]¹.
Причинно-обусловленные устройства**

Принципиальным в подходе Маллера к изучению поведения асинхронных устройств явился отказ от детерминированной модели. Действительно, поскольку асинхронное поведение в общем случае связано с непредсказуемыми вариациями длительностей процессов перехода из одного состояния в другое в физическом времени, состояние

¹ Изложение результатов Маллера дается по работе [7, гл. 10]. Ряд утверждений приводится без доказательств, которые можно найти там же.

асинхронного устройства в любой фиксированный момент физического времени не определено. В ряде случаев мы имеем достаточно полное представление о вероятностном характере вариаций задержек и о параметрах соответствующих распределений вероятностей². Однако никакой особой пользы вероятностный подход к изучению логического поведения не приносит. В действительности нас интересует ответ на вполне определенный вопрос: «Как и при каких условиях из поведения, недетерминированного в физическом времени, получить поведение, детерминированное в логическом времени?». Для получения ответа на этот вопрос в модели Маллера изучается полная совокупность логических возможностей.

Полная схема, или схема Маллера³, определяет как конечное множество S из N объектов a, b, c, \dots , называемых состояниями схемы. Поведение схемы задается множеством допустимых последовательностей состояний:

$$\left\{ \begin{array}{l} a(1), a(2), \dots \\ b(1), b(2), \dots \\ \dots \dots \dots \end{array} \right\},$$

обладающих следующими свойствами:

$$a(i) \neq a(i+1) \quad \forall i;$$

каждое состояние $a \in S$ является началом, по крайней мере, одной допустимой последовательности;

если $a(1), a(2), a(3), \dots$ – допустимая последовательность, то $a(2), a(3), \dots, a(2)$ – также допустимая последовательность;

если $a(1), a(2), \dots$ и $b(1), b(2), \dots$ – допустимые последовательности и $a(2) = b(1)$, то $a(1), b(1), b(2), \dots$ – допустимая последовательность⁴.

Схема может быть задана ориентированным графом возможных соседних переходов из одного состояния в другое. Состояния $a \in S$ и $b \in S$ находятся в отношении aRb , если существует логическая возможность непосредственного перехода из состояния a в состояние b , т. е. если на графе существует дуга, ведущая из a в b . Отношение R порождает множество окрестностей $a \quad Q(a) \in S$, таких, что $b \in Q(a)$, если и только если aRb . Последовательность $a(1), a(2), \dots, a(m)$ называется R -последовательностью, если $a(i)Ra(i+1)$ для $1 \leq i \leq m-1$. Каждому пути на графе соответствует R -последовательность состояний⁵.

² Как, например, в случае смены квантовых состояний или выхода электронного арбитра из метастабильного состояния.

³ Или просто схема.

⁴ Последние два свойства формируют свойство марковости допустимой последовательности – вся предыстория поведения содержится в текущем состоянии.

⁵ Но не каждая R -последовательность является допустимой.

Мы будем говорить, что состояние a предшествует состоянию b (aFb – отношение следования), если существует R -последовательность (путь на графе), ведущая из a в b . Отношение F рефлексивно (a, a есть R -последовательность) и транзитивно (из aFb и bFc следует aFc). Если $a \neq b$ и aFb , то a и b принадлежат допустимой последовательности.

Состояния a и b называются эквивалентными (aEb), если aFb и bFa , т. е. если в графе имеется цикл, содержащий a и b . Отношение эквивалентности разбивает все множество состояний S на классы эквивалентности A, B, C, \dots . Отношение следования имеет место и для классов эквивалентности AFB , если существуют два состояния $a^* \in A$ и $b^* \in B$, такие, что a^*Fb^* . И, если AFB , то $\forall (a \in A) \& (b \in B) aFb$. Если AFB и BFA , то $A = B$.

Отношение F для классов эквивалентности определяет частичное упорядочение классов¹. Любой допустимой последовательности состояний соответствует единственная конечная последовательность классов эквивалентности и для любой допустимой последовательности состояний существует последний класс эквивалентности $A(m)$, называемый заключительным. В частичном порядке на классах эквивалентности существует один или несколько максимальных классов. Если заключительный класс для некоторой допустимой последовательности не является максимальным, то он называется псевдомаксимальным. Согласно определению Маллера, схема называется независимой от скорости² относительно состояния u , если все допустимые последовательности, начинающиеся с u , имеют один и тот же заключительный класс.

Теорема [7, 10.2.5]. Схема C не зависит от скорости относительно u , если и только если класс эквивалентности $U(u \in U)$ предшествует только одному максимальному классу K и не предшествует ни одному псевдомаксимальному классу.

Определенная выше схема C является абстрактным объектом. Структурная теория предусматривает более детальное описание, в частности, представление состояний сигналами, а сигналов – логическими элементами. Структуризация требует изменения³ определения схемы.

Как обычно в структурной теории автоматов, схема определяется как совокупность n логических элементов, соединенных между собой. Каждый логический элемент имеет k_i входов, $k_i \leq n$, и один выход x_i . Выходы логических элементов называются узлами схемы и n -местный набор значений переменных в узлах схемы $X = \{x_0, x_1, \dots, x_{n-1}\}$, является

ее состоянием⁴. Каждый вход логического элемента соединен⁵ только с одним выходом и никакие два выхода не соединены между собой. С каждым логическим элементом⁶ связана логическая функция $x'_i = f_i(X)$, где x'_i называется следующим значением x_i . Если x'_i явно зависит от x_i , то логический элемент есть элемент памяти. В противном случае мы имеем комбинационный элемент. Выход логического элемента в некотором состоянии X_j называется стабильным, если $x_i \oplus f_i(X_j) = 0$, и возбужденным, если $x_i \oplus f_i(X_j) = 1$. В возбужденном состоянии входы логического элемента уже изменились⁷, а выходная переменная еще не изменилась. Если в некотором состоянии X_j возбуждено m переменных, то существует 2^m логических возможностей для смены состояний, порождающих граф полной схемы Маллера. Если каждой структурированной схеме соответствует абстрактная схема, то все определения и результаты, полученные для абстрактной модели⁸, могут быть распространены на структурированную модель. При этом, однако, специфика структурированной модели оправдывает рассмотрение других подходов.

В соответствии с определением возбужденного и стабильного состояний логического элемента в бинарной модели Маллера каждая переменная может принимать четыре значения: $\{0, 1\}$ – стабильные значения и $\{0^*, 1^*\}$ – возбужденные значения. Диаграмма состояний при этом называется диаграммой Маллера. Пример диаграммы Маллера для схемы

$$x_0 = x_2 + \bar{x}_3; \quad x_1 = \bar{x}_3; \quad x_2 = x_1 \bar{x}_3;$$

$$x_3 = x_0 x_2 + x_3 (x_0 + x_1)$$

относительно начального состояния $0^*0^*0^*0$ приведен на рис. 4, a^9 . Заметим, что, несмотря на четырехзначное кодирование, схема имеет не 4^n , а 2^n возможных состояний, из которых в нашем примере четыре состояния недостижимы из рассмотренного начального состояния. Для состояний a и b диаграммы Маллера естественным образом определяется отношение непосредственного следования aRb и понятие R -последовательности. Каждой R -последовательности $X(1), X(2), \dots$,

⁴ Как правило, рассматриваются бинарные переменные, хотя нет никаких специальных ограничений на значность переменных. Так, в работе [19] рассматриваются многозначные самосинхронные коды.

⁵ Функционально связан.

⁶ Кстати, уже здесь мы можем сделать предварительное замечание об отличии асинхронной модели от синхронной. В синхронной модели логическая функция элемента определяется как $x_i(t+1) = f_i(X(t))$, где t – специально выделенная переменная, сигнал логического времени.

⁷ Возникла причина изменения состояния выхода.

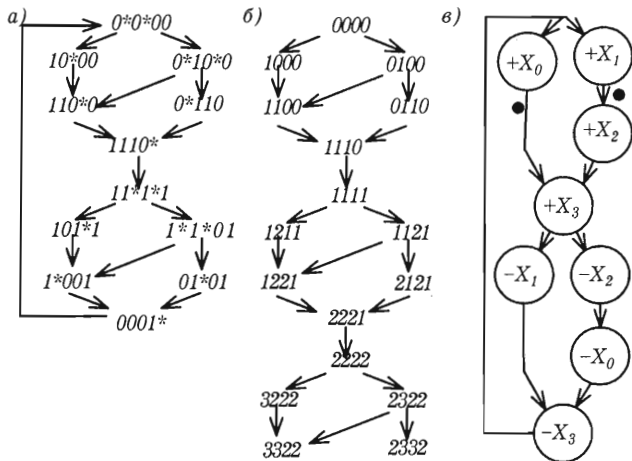
⁸ Полной схемы Маллера.

⁹ Поскольку, как мы уже отмечали выше, одновременность в физическом времени недостижима, то в диаграмме отмечены только соседние переходы.

¹ Заметим, что это упорядочение является «хронологическим» (временным в логическом времени).

² Независимой от длительностей процессов смены состояний в физическом времени.

³ Детализации.



■ Рис. 4. Диаграмма Маллера (а), кумулятивная диаграмма Маллера (б), сигнальный граф (в)

$X(m)$ можно поставить в соответствие последовательность кумулятивных состояний (К-состояний) $\alpha(1), \alpha(2), \dots, \alpha(m)$. Кумулятивное состояние $\alpha(j)$ есть вектор $\alpha(j) = [\alpha_0(j), \alpha_1(j), \dots, \alpha_{n-1}(j)]$ с целочисленными компонентами $\alpha_i(j)$, такой что $\alpha_i(j) = 0$ и $\alpha_i(j+1) = \alpha_i(j) + [x_i(j) \oplus f_i(X(j))]$. Иными словами, компонента кумулятивного состояния $\alpha(j)$ есть число изменений переменной x_i от начального состояния $\alpha(1)$ до состояния $\alpha(j)$. Кумулятивная диаграмма для примера рис. 4, а приведена на рис. 4, б. В кумулятивной диаграмме $\alpha \leq \beta$, если $\alpha_i \leq \beta_i \forall i$; $\gamma = \alpha \vee \beta$, если $\gamma_i = \max(\alpha_i, \beta_i)$; и $\gamma = \alpha \wedge \beta$, если $\gamma_i = \min(\alpha_i, \beta_i)$. При частичном упорядочении $\alpha \leq \beta$ кумулятивная диаграмма есть структура. Маллером показано, что, если эта структура полумодулярна, то поведение соответствующей схемы не зависит от скорости и схема называется полумодулярной.

Диаграмма Маллера для полумодулярных схем имеет простое локальное свойство. Пусть $Q(X)$ – окрестность состояния X в диаграмме Маллера – $XRY \forall Y \in Q(X)$; $D(X(1))$ – множество состояний¹, достижимых из $X(1)$. Тогда для полумодулярности соответствующей схемы (кумулятивной диаграммы) необходимо и достаточно, чтобы, если $x_i = 0*(1*) \forall X \in D(X(1))$, тогда $y_i \neq 0, 1*(1, 0*) \forall Y \in Q(X)$. Иными словами, для полумодулярности схемы необходимо и достаточно, чтобы каждый раз, когда выход логического элемента возбуждается (возникает причина для изменения выхода), возбуждение снимается только путем изменения выхода (причина сохраняется до тех пор, пока не реализуется ее следствие). Это свойство мы будем называть причинной обусловленностью.

Объем информации, содержащийся в диаграмме Маллера, вообще говоря, избыточен. Вместо графа смены состояний может быть использована

но событийное описание поведения – частичный порядок на событиях, состоящих в смене значений переменных. Языками такого описания являются сигнальные графы (СГ), диаграммы изменений (ДИ), маркированные сети Петри (МСП) и др. [8, 9, 10]. Все эти языки обладают одинаковыми изобразительными возможностями. На рис. 4, в приведен СГ (ДИ)² для примера на рис. 4, а. Маркеры на дугах обозначают текущее состояние схемы. Для ДИ аналогично диаграмме Маллера может быть построена кумулятивная диаграмма и известен набор алгебраических свойств, обеспечивающих полумодулярность соответствующих схем. По ДИ однозначно восстанавливается диаграмма Маллера. Логические функции переменных задаются диаграммой Маллера следующим образом: если X – состояние рабочего цикла диаграммы Маллера³, то если $x_i = 0(1)$, тогда $f_i(X) = 0(1)$, и если $x_i = 0*(1*)$, тогда $f_i(X) = 1(0)$ (это задание, вообще говоря, неоднозначно, так как функция может быть определена произвольно на состояниях, не входящих в рабочий цикл). Последнее обеспечивает задачу синтеза полумодулярных схем, спецификация которых задана ДИ или диаграммой Маллера. Задача синтеза, однако, этим не ограничивается по следующим причинам.

1. При переходе от ДИ к диаграмме Маллера в последней могут встретиться противоречивые состояния, т. е. состояния, в которых при одних и тех же булевых значениях возбуждены разные переменные. В этом случае для снятия противоречий необходимо введение дополнительных переменных, развязывающих противоречия. Естественно, что введение дополнительных переменных не должно нарушать свойство полумодулярности.

2. Синтез реальных схем всегда осуществляется в некотором функциональном базисе. Получающиеся в результате формального синтеза логические функции могут не входить в функциональный базис и должны быть представлены как суперпозиции базисных функций. При этом, опять-таки, схема должна сохранять свойство полумодулярности относительно выходов всех логических элементов.

Интересно отметить связь алгебраических свойств кумулятивной диаграммы Маллера и логических свойств соответствующих схем. Например, если кумулятивная диаграмма полумодулярна и дистрибутивна, то элемент 2И-НЕТ или 2ИЛИ-НЕТ является функционально полным в классе полумодулярных схем. Для недистрибутивных схем⁴ функционально полный набор содержит элементы 2И-НЕТ и 2ИЛИ-НЕТ.

² СГ являются подмножеством ДИ.

³ Множество состояний, достижимых из начального.

⁴ Содержательно недистрибутивность соответствует возникновению очередного возбуждения переменной более чем в одном состоянии.

¹ Множество $D(X(1))$ иногда называют рабочим циклом.

Как мы уже говорили выше, логическое время определяется частичным порядком на событиях, отражающим причинно-следственные отношения между ними. Для кумулятивной диаграммы Маллера момент логического времени в каждом состоянии может быть определен, в зависимости от семантики специфицируемого поведения, по крайней мере, тремя способами, а именно:

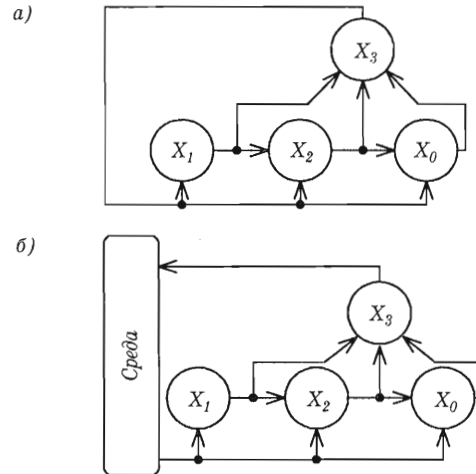
$$1) T(\alpha) = \sum_{j=0}^{n-1} \alpha_j, \text{ в случае, когда каждое собы-}$$

тие, состоящее в смене значения переменной, формирует шаг логического времени;

2) $T(\alpha) = \max_j \alpha_j$, в случае, когда шаг логического времени формируется сменой значения «опережающей» переменной;

3) $T(\alpha) = \alpha_k$, в случае, когда шаг логического времени формируется сменой одной выделенной переменной¹.

Для ДИ и ее кумулятивной развертки определение моментов логического времени не столь очевидно. События ДИ определяют смену состояний и, следовательно, моменты логического времени естественно связать с дугами (маркерами). Комбинация маркеров однозначно определяет текущее состояние и, если сопоставить рис. 4, а, б и в, то нетрудно видеть, что для случая 1 событие $+X_0$ (возбужденное состояние 0^*) присутствует в двух моментах логического времени. Таким образом, если в ДИ маркер на дуге $+X_1 \rightarrow +X_2$ имеет значение логического времени t , то маркер на дуге $+X_0 \rightarrow +X_3$ «размазан» по моментам времени $t \rightarrow t+1$. Таким образом, понимание логического времени для состояний и событий, вообще говоря, различно. Логическое время для состояния единственно для всей схемы в силу того, что в каждый момент физического времени схема находится в одном и только одном состоянии². Логические времена для различных событий могут быть различны в один и тот же момент физического времени, и одно и то же логическое время для разных событий может быть в различные моменты физического времени. Интерпретация событийного логического времени зависит от соглашения, которое мы примем для этой интерпретации³. Например, если события $E_1(t_1) \& \dots \& E_k(t_k)$ явля-



■ Рис. 5. Схема Маллера для спецификации на рис. 4

ются причиной события $E_m(t_m)$, мы можем считать, что $t_m = \max(t_1, \dots, t_k) + 1$ (напомним, что моменты логического времени маркируют выходные дуги соответствующих событий и нарушение непосредственного следования моментов времени на входных и выходных дугах одного события не является чем-то необычным). Для случая 3 в нашем примере, если выделена переменная x_3 , то в каждом состоянии маркировок дуг все маркеры имеют одно и то же значение логического времени. Если же в качестве переменной, формирующей значения логического времени, выбрана переменная x_1 или x_2 , то ситуация становится более сложной.

До сих пор мы рассматривали модели автономных схем и их поведение. Однако в реальных ситуациях любое устройство должно взаимодействовать и взаимодействует с внешней средой. Рассмотрим простейшую возможность организации такого взаимодействия.

На рис. 5, а приведена схема, специфицированная диаграммой Маллера (ДИ) на рис. 4. Согласно определению схемы Маллера, значения переменных x_i отнесены к выходам логических элементов и значения каждой переменной на входах других элементов идентичны в любой момент физического времени. Это приводит к соглашению Маллера о задержках: *все задержки приведены к выходу логического элемента и разбросом задержек в проводах после их разветвления можно пренебречь*⁴. Понятие независимости от скорости состоит в том, что поведение схемы в смысле сохранения частичного порядка на событиях не изменится, если последовательно выходу любого

¹ Выделенная переменная при этом может интерпретироваться как синхронизирующая и является в некотором смысле аналогом синхросигнала в системе с внешними часами.

² Это утверждение не совсем точно в силу того, что при смене состояния переменная проходит непрерывное множество значений и граница между значениями 0 и 1 размыта. Анализ режима переключения выходит за рамки этой статьи, однако сказанное может быть использовано с требуемой для наших целей степенью точности.

³ В действительности интерпретация в значительной мере зависит от семантики описываемых процессов.

⁴ Соглашение Маллера о задержках выполняется не всегда и, вообще говоря, зависит от разницы длин проводов после разветвления и конкретной топологии. Вопросы теории схем, не чувствительных к задержкам в проводах, выходят за рамки этой статьи и для нашего дальнейшего изложения соглашение Маллера является вполне удовлетворительным.

элемента включить произвольную задержку. В качестве такой задержки может быть включена внешняя среда (рис. 5, б). При этом мы можем ввести дополнительное обозначение для выходного сигнала среды e_i и, если среда включена последовательно выходу логического элемента x_i , то интерфейс между схемой и средой прост и очевиден $e_i = x_i$, $x_i = \bar{e}_i$. Моделью схемы с точки зрения среды является инвертор с произвольной последовательной задержкой. Сигнал e_i может рассматриваться как инициирующий (синхронизирующий) вход схемы, а сигнал x_i – как сигнал о завершении переходных процессов в схеме. Последнее, вообще говоря, не совсем точно. Если в качестве интерфейсного сигнала выбрана, например, переменная x_2 , то, как следует из рис. 4, в, при $x_2 = 1$ может быть не завершено переключение переменной x_0 , а при $x_2 = 0$ – переменной x_1 . Завершение этих переключений проверяется в следующем цикле работы композиции схема–среда, и это открывает ряд новых возможностей, при которых часть внутренних переменных схемы переключается параллельно работе внешней среды¹.

Поскольку моделью разомкнутой полумодулярной схемы является инвертор с последовательной задержкой, то моделью разомкнутой полумодулярной схемы с последовательным инвертором является задержка. Это соображение открывает возможность композиции полумодулярных схем и их блочного синтеза, при котором разомкнутые полумодулярные блоки с последовательными инверторами включаются как задержки, последовательные выходы полумодулярной схемы, координирующей поведение этих блоков.

Синхронная схема с общими часами также может быть условно специфицирована ДИ (рис. 6). При этом, во-первых, моделью часов является калиброванная задержка и, во-вторых, если задержка, равная периоду синхронизации, заведомо больше любой возможной длительности переходных процессов в устройстве², то связями от событий в устройстве к событиям $\pm T$ можно пренебречь.

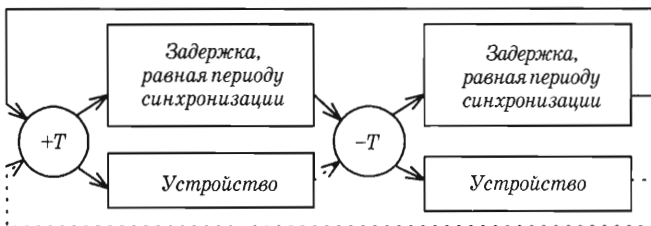


Рис. 6. Сигнальный граф для случая синхронизации от внешних часов

¹ Использование этих возможностей тесно связано с семантикой поведения.

² А это – неперемное условие корректности поведения при синхронизации от общих часов.

Стратегия распределенной синхронизации

Мы должны отчетливо понимать, что система синхронизации поведения устройства от внешних часов включает в себя не только собственно часы, но и систему доставки сигналов от часов в точки их взаимодействия с устройством. Не будет преувеличением сказать, что, как только мы вводим в рассмотрение не только часы, но и систему доставки сигналов времени³, мы переходим к рассмотрению проблемы пространства-времени. Эта проблема для искусственных систем имеет свою специфику⁴. В VLSI and VLSI-системах мы фактически имеем дело с проводной метрикой, заданной на графе связей между модулями. Два модуля A и B являются соседними относительно сигнала x_i , если существует провод, передающий сигнал x_i , от модуля A к модулю B . Расстояние⁵ от модуля A до модуля B относительно сигнала x_i , $D(AB_{x_i})$ есть время, в течение которого сигнал x_i распространяется от модуля A до модуля B . Заметим при этом, что если сигналы x и y распространяются от A к B по разным проводам, то $D(AB_x) \neq D(AB_y)$.

Задержка распространения сигнала по проводу определяется двумя факторами: во-первых, распределенными параметрами RC (постоянная времени $\tau_1 = RC l^2 / 2 \approx 100$ пс/мм), во-вторых, ограничениями на плотность тока в проводе. Плотность тока⁶, приводящая к миграции атомов, для алюминия равна 10^5 А/см² = 1 мА/μ² и время заряда емкости провода постоянным током⁷ определяется как $\tau_2 = VCl / I \approx 100$ пс/(мм·мА). Если доминирующим фактором является плотность тока, т. е. если

$$\frac{\tau_1}{\tau_2} = \frac{RII}{2V} \ll 1, \text{ то поверхность провода можно счита-}$$

тать эквипотенциальной обкладкой конденсатора. В этом случае говорят об эквипотенциальной или эквихронной зоне. Задержки, вообще говоря, существенно больше, так как провод нагружен на входы транзисторов, входные емкости которых на единицу площади в 50–100 раз больше, чем у провода. Оценка размеров эквихронной зоны достаточно сложна и зависит от принятых гипотез и критериев, однако по общим оценкам эти размеры имеют порядок 1 мм. Из понятия эквихронной зоны, независимо от ее размеров, следуют два важных вывода:

³ Заметим, что не только времени, но и информационных сигналов.

⁴ В частности, для VLSI и VLSI-систем, с которыми мы и будем далее иметь дело.

⁵ Это совсем не экзотический способ измерения расстояний. Например, на дорожных картах Новой Зеландии расстояние между городами приводится в часах и минутах for driving.

⁶ Это ограничение для постоянного тока. Для знакопеременных токов допустимое значение плотности может быть в несколько раз больше.

⁷ При увеличении ширины провода в равной мере увеличивается его емкость и допустимый ток.

1) совокупность эквихронных зон порождает совокупность локальных времен;

2) система доставки сигналов времени должна обеспечивать взаимную координацию локальных времен у взаимодействующих эквихронных зон.

Использование общих часов для синхронизации поведения VLSI предъявляет соответствующие требования к системе доставки сигналов времени. Как правило, проводная система доставки сигналов времени от общих часов представляет собой Н-дерево (рис. 7), в котором точки ввода сигнала времени в эквихронную зону равноудалены от общего источника этих сигналов. Балансировка Н-дерева для современных скоростей синхронизации (2–4 ГГц) и современных размеров VLSI представляет собой весьма непростую задачу и каждый новый шаг на этом пути дается все большей и большей кровью.

Основные проблемы, возникающие при всех модификациях Н-дерева, следующие: разброс физических и технологических параметров, приводящий к разбросу задержек, и большая мощность, необходимая для перезарядки проводов Н-дерева¹. Эти проблемы, вообще говоря, ограничивают число терминальных точек в Н-дереве и степень дробления системы на модули, которые рассматриваются как эквихронные.

Вернемся к рассмотрению механизма работы часов. Любые известные часы, от солнечных и водяных до точнейших атомных, имеют в своей основе какой-либо периодический процесс, протекающий в физическом времени. Часы, используемые в VLSI, представляют собой генератор (clock generator) с калиброванной задержкой в цепи отрицательной обратной связи (рис. 8, а)². Какие требования предъявляются к периоду генерации? Толь-

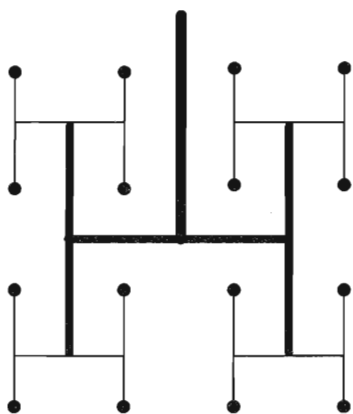


Рис. 7. Н-дерево

¹ Так, при 32 эквихронных зонах и частоте 400 МГц только на перезаряд емкости проводов Н-дерева требуется около 2,5 Вт.

² Мы пока здесь будем рассматривать изолированное (автономное) устройство. Взаимодействие с внешней средой представляет собой специальный вопрос.

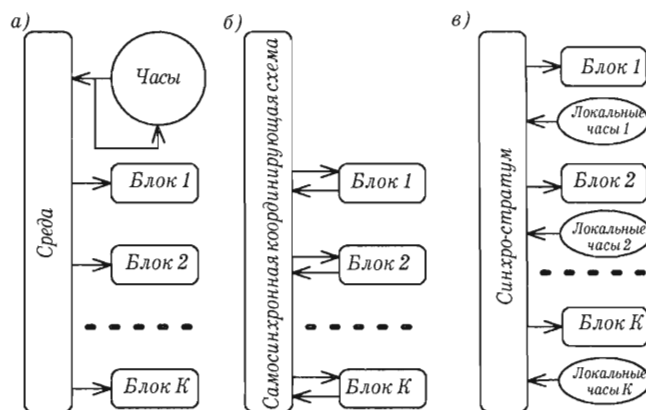


Рис. 8. Способы синхронизации

ко одно – длительность периода должна быть достаточной для завершения переходных процессов и в рамках этого ограничения точность поддержания периода не имеет существенного значения. Более того, период вообще может быть эластичным, если мы можем управлять задержкой в зависимости от режима работы VLSI (задержка, зависящая от данных, от параметра и т. д.) или тем или иным способом фиксировать моменты окончания переходных процессов (рис. 8, б). Однако, как мы уже говорили выше, камнем преткновения является система распределения и доставки сигналов времени. При этом очевидно, что координироваться между собой должны только локальные сигналы времени у соседних по графу связей модулей. Это приводит нас к идее децентрализованной синхронизации (рис. 8, в). Другими словами, мы попытаемся показать, что могут быть созданы распределенные, асинхронные часы, в которых пассивная проводная система доставки синхросигналов заменяется активной средой. Такое устройство будем называть синхро-стратумом.

Синхро-стратум

В качестве модели для следующего шага нашего изложения мы будем рассматривать одномерный клеточный массив из N синхронных автоматов Мура A_j (модулей, блоков), в котором каждый автомат A_j связан информационным обменом только с двумя своими непосредственными соседями (A_{j-1}, A_{j+1}). Если $S_j(t)$ – состояние автомата A_j в момент логического времени t , то

$$S_j(t+1) = F_j[S_{j-1}(t), S_j(t), S_{j+1}(t)]. \quad (1)$$

Заметим³, что если поведение массива задано системой логических уравнений (1), то известными преобразованиями пространства состояний и функций могут быть построены массивы с эквивалентным поведением:

³ И это нам будет полезно в дальнейшем.

$$\begin{aligned}
 S_j(t+1) &= F_j[S_{j-1}(t+1), S_j(t), S_{j+1}(t)]; \\
 S_j(t+1) &= F_j[S_{j-1}(t), S_j(t), S_{j+1}(t+1)]; \\
 S_j(t+1) &= F_j[S_{j-1}(t+1), S_j(t), S_{j+1}(t+1)]. \quad (2)
 \end{aligned}$$

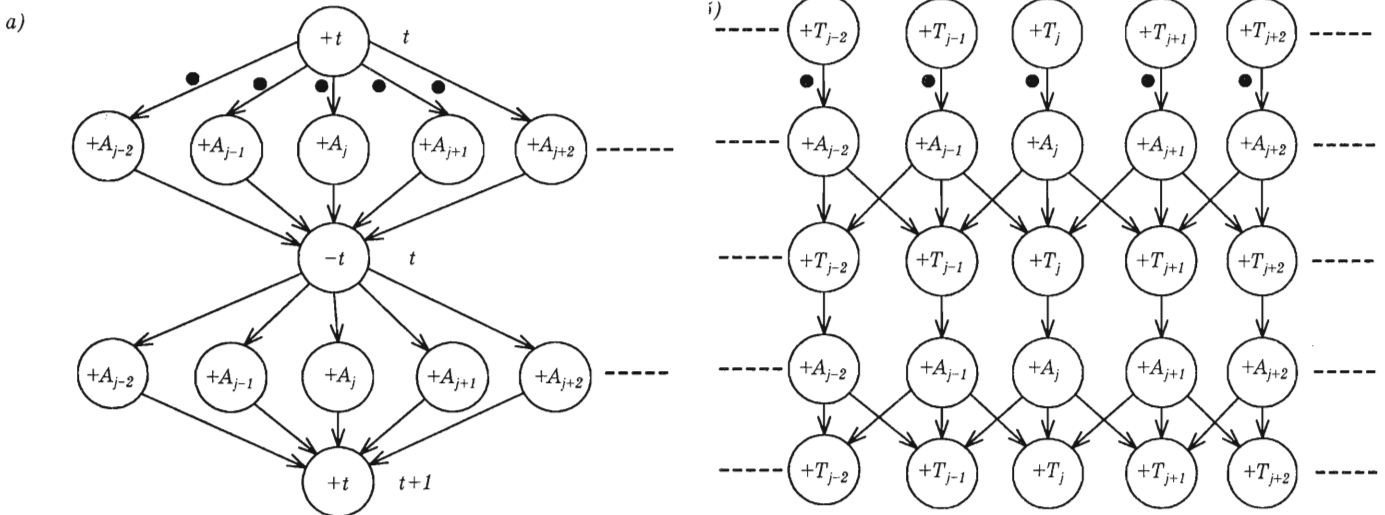
В уравнениях (1) и (2) t – целочисленная переменная, представляющая глобальное (общее для всего массива) логическое время. Введем понятие локального логического времени T_j , представляющего значение глобального логического времени на входе автомата A_j . При этом уравнение (1) имеет вид

$$\begin{aligned}
 S_j(T_j+1) &= F_j[S_{j-1}(T_{j-1}=T_j), S_j(T_j), \\
 &S_{j+1}(T_{j+1}=T_j)]. \quad (3)
 \end{aligned}$$

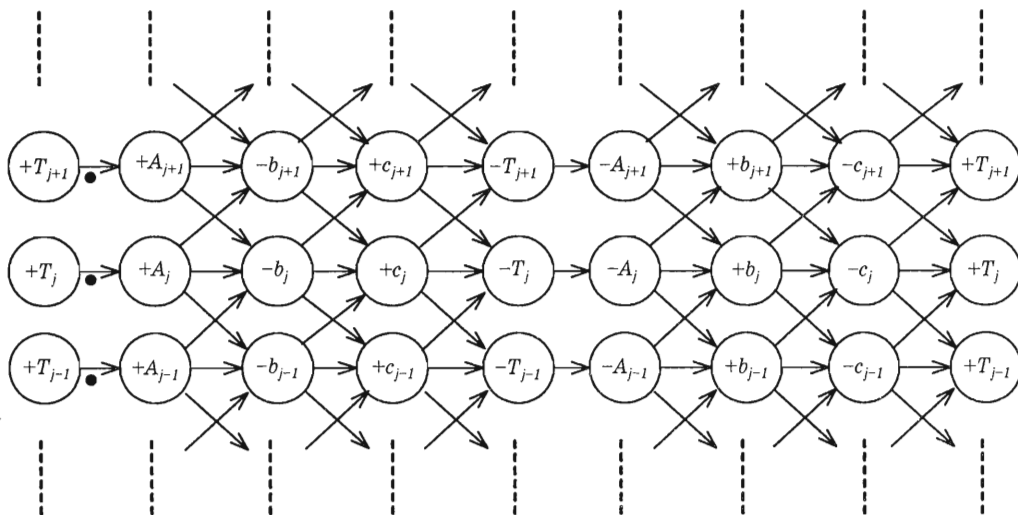
Из уравнения (3) с очевидностью следует, что для обеспечения корректности временного поведения массива нам достаточно координировать значения локальных логических времен только у ближайших соседей. На рис. 9, а приведена ДИ для глобальной синхронизации массивов от общего синхросигнала t , а на рис. 9, б – ДИ для эквивалентной координации локальных времен.

ДИ на рис. 9, б корректна, однако ей не соответствует никакая схема, так как соответствующая ДИ диаграмма Маллера содержит противоречивые состояния. Для снятия противоречий в ДИ должны быть введены дополнительные переменные, как, например, на рис. 10.

Синтезированная по этой спецификации схема имеет вид



■ Рис. 9. Диаграмма изменений для синхро-стратум



■ Рис. 10. Диаграмма изменений для реализации синхро-стратума на рис. 9

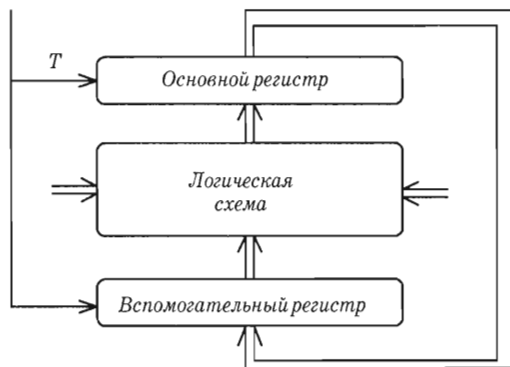
$$\begin{aligned} T_i &= \overline{c_{i-1}c_i c_{i+1} + b_i(c_{i-1} + c_i + c_{i+1})}, \\ c_i &= \overline{b_{i-1}b_i b_{i+1} + T_i(b_{i-1} + b_i + b_{i+1})}, \\ b_i &= \overline{a_{i-1}a_i a_{i+1} + c_i(a_{i-1} + a_i + a_{i+1})}, \end{aligned} \quad (4)$$

что, собственно, принципиально решает задачу о возможности реализации синхро-стратума. Однако может быть приведен ряд аргументов в пользу более детального анализа реализации синхро-стратума.

Во-первых, схема (4) достаточна сложна. Схема для каждой точки синхронизации содержит 42 транзистора в CMOS реализации¹ и, что более важно, она требует шесть входящих проводов от соседних точек синхронизации и столько же исходящих проводов к ним².

На практике используются различные системы синхронизации, включающие несколько синхросигналов (синхросерий), и каждый синхронный прототип приводит к своей структуре синхро-стратума.

Основная идея синхронизации сигналами от часов связана тем или иным образом с организацией поведения блоков схемы по принципу «основной–вспомогательный». Стратегии синхронизации мы будем рассматривать на модели массива из клеточных автоматов, автоматы которой являются автоматами Мура, построенными на основе двухрегистровой схемы, работающей по принципу «основной–вспомогательный», или «ведущий–ведомый» (рис. 11). При одном значении сигнала T от часов автомат изменяет свое текущее состояние путем записи нового состояния с выходов логической схемы в основной регистр (рабочая фаза). При другом значении T текущее состояние не изменяется, а только переписывается из основного регистра во вспомогательный (пас-



■ Рис. 11. Структура автомата с двухфазным разнополярным управлением

¹ Дополнительная задержка, вносимая синхро-стратумом, равна 6τ на полный цикл синхронизации, где τ – задержка одного вентиля.

² С ростом числа соседей пропорционально возрастает число проводов и соответственно число транзисторов.

сивная фаза). Такой принцип работы называется двухфазным с разнополярным управлением. В рабочей фазе неизменным является состояние вспомогательного регистра, хранящего предыдущее состояние автомата, а в пассивной фазе неизменно состояние основного регистра, хранящего текущее состояние.

Существенным фактором для временной координации поведения является неизменность выходов соседних автоматов (входов данного автомата) в рабочей фазе. В работах [11–15] описано несколько реализаций синхро-стратума. Здесь, в дополнение к схеме (4), мы приведем только одно новое решение. Рассмотрим фрагмент ДИ на рис. 12.

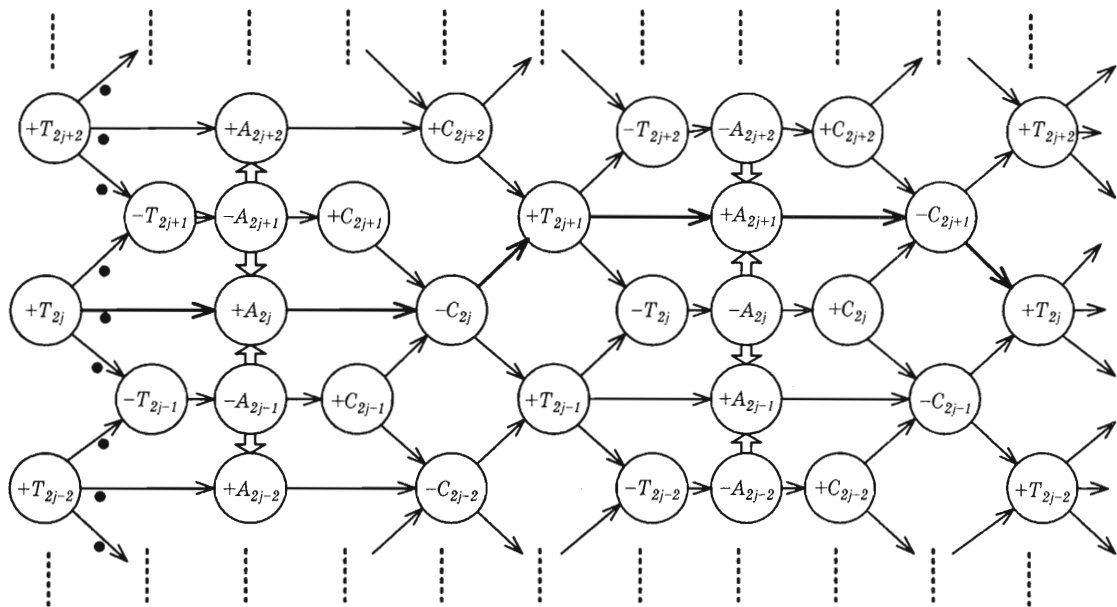
Схема, синтезированная по этой ДИ, имеет вид

$$\begin{aligned} T_i &= \overline{T_{i-1}T_{i+1}(C_{i-1} + C_{i+1})}, \\ C_i &= \overline{C_{i-1}A_i C_{i+1}}. \end{aligned} \quad (5)$$

Из схемы (5) следует, что этот синхро-стратум имеет сложность 14 транзисторов на точку синхронизации в CMOS реализации, четыре входящих и четыре выходящих провода для интерфейса между ячейками синхро-стратума. По этим параметрам он проигрывает лучшему из известных [15] (см. рис. 5) синхро-стратумов³. Однако схема (5) обладает одним замечательным свойством – предельным быстродействием. Действительно, если мы обратимся к структуре автомата «основной–вспомогательный», то заметим, что длительности переходных процессов в разных фазах, вообще говоря, различны. В рабочей фазе длительность переходного процесса складывается из задержки логической схемы и задержки записи в регистр. В пассивной фазе переходный процесс содержит только запись в регистр. Пусть в ДИ на рис. 12 рабочий цикл в автоматах реализуется при $T_j = 1$ (знаки \uparrow и \downarrow в ДИ обозначают время и направление передачи информации с основных регистров автоматов A_{j-1} и A_{j+1} на логическую схему автомата A_j) и пусть задержка в логической схеме $\tau_{logic} = 2\tau_{gate}$. Тогда полный цикл синхронизации определяется путем, выделенным на рис. 12 жирными стрелками. Нетрудно видеть, что дополнительная задержка, вносимая синхро-стратумом, равна $4\tau_{gate}$ на полный цикл изменения T_j (заметим, что для минимальной схемы [15; рис. 5] дополнительная задержка, вносимая синхро-стратумом, равна $10\tau_{gate}$ и задержка для схемы (5), по-видимому, минимальна).

Мы будем считать, что полный цикл смены значений локального сигнала времени $\rightarrow -T_j \rightarrow +T_j$ представляет собой один шаг локального логического времени, и введем обозначения $1(k)$ и $0(k)$ для значений $T_j(k) = 1$ и $T_j(k) = 0$ соответствен-

³ Десять транзисторов, один входящий и один выходящий провод для внутривстратумного интерфейса.



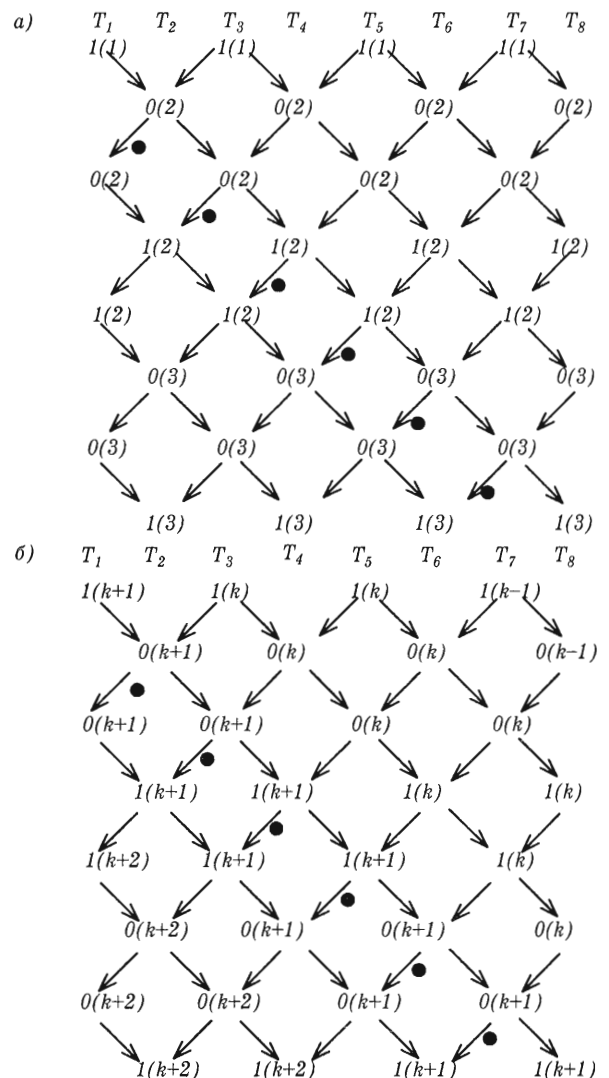
■ Рис. 12. Диаграмма изменений для быстрого синхро-стратума

но. Пусть в начальный момент времени $T_j(1) = 1$ для любого j . Тогда проекция кумулятивной развертки ДИ, представленной на рис. 12, имеет вид рис. 13, а. При этом автоматные уравнения различны для четных и нечетных автоматов и имеют вид

$$S_{2j}(t+1) = F_{2j}[S_{2j-1}(t), S_{2j}(t), S_{2j+1}(t)],$$

$$S_{2j+1}(t+1) = F_{2j+1}[S_{2j}(t+1), S_{2j+1}(t), S_{2j+2}(t)]. \quad (6)$$

Размышления над диаграммой на рис. 13, а позволяют получить ряд любопытных соображений о временном поведении системы. Если мы зафиксируем автомат A_1 в состоянии 1(1), то система¹ придет в устойчивое состояние с распределением логических времен: 1, 2, 2, 2, 2, 3, 3, 3, 3, 4, Из этого следует, что как в устойчивом состоянии, так и в процессе функционирования² в разных точках синхро-стратума могут существовать разные логические времена. Более того, если мы можем управлять ритмом работы одного из автоматов или соответствующей ячейкой синхро-стратума, то эта точка массива будет генератором ритма для всего массива и от нее по массиву будут распространяться «волны времени». Мы можем попытаться организовать поведение так, чтобы в устойчивом состоянии и на фронте «волны времени» значение логического времени было одинаковым для всех автоматов. Такую возможность открывает диаграмма на рис. 13, б. В этой диаграмме для нечетных автоматов шаг логического времени формируется циклом $\rightarrow +T_j \rightarrow -T_j$, а для четных – циклом $\rightarrow -T_j \rightarrow +T_j$. При этом, как



■ Рис. 13. Локальное распределение логического времени

¹ Как это видно из распределения маркеров на рис. 13, а.

² При вариациях длительностей переходных процессов у разных автоматов.

это видно из рис. 13, б, автоматные уравнения имеют период 4:

$$\begin{aligned} S_{4j}(t+1) &= F_{4j}[S_{4j-1}(t), S_{4j}(t), S_{4j+1}(t)], \\ S_{4j+1}(t+1) &= F_{4j+1}[S_{4j}(t), S_{4j+1}(t), S_{4j+2}(t)], \\ S_{4j+2}(t+1) &= F_{4j+2}[S_{4j+1}(t), S_{4j+2}(t), S_{4j+3}(t)], \\ S_{4j+3}(t+1) &= F_{4j+3}[S_{4j+2}(t), S_{4j+3}(t), S_{4j+4}(t)]. \end{aligned} \quad (7)$$

Таким образом, меняя вид автоматных уравнений, можно переходить от параллельной к волновой синхронизации.

Переход от одномерного к двумерному массиву связан только с увеличением числа соседей и уравнения (5) приобретают вид

$$\begin{aligned} T_{i,j} &= \overline{T_{i-1,j} T_{i,j-1} T_{i+1,j} T_{i,j+1} (C_{i-1,j} + C_{i,j-1} + C_{i+1,j} + C_{i,j+1})}, \\ C_{i,j} &= \overline{C_{i-1,j} C_{i,j-1} A_{i,j} C_{i+1,j} C_{i,j+1}}. \end{aligned} \quad (8)$$

Синхро-стратум может быть синтезирован и для произвольного графа связей. Логические функции для его элементов при этом обобщаются естественным образом на случай произвольного графа. Пусть V_j – множество индексов окрестности автомата A_j на графе связей, т. е. $V_j = \{i\}$ для любых i таких, что A_i имеет информационную связь с A_j . Тогда функции элементов синхро-стратума имеют вид:

$$C_j = \overline{A_j \prod_{i \in V_j} C_i}; \quad T_j = \overline{\prod_{i \in V_j} T_i \prod_{i \in V_j} C_j}. \quad (9)$$

Для организации взаимодействия типа¹ рис. 12 (9) необходимым и достаточным условием корректности поведения синхро-стратума является бихроматичность графа связей [12]. Однако для синхро-стратума типа рис. 10 (4) такого ограничения на граф связей нет и функции элементов имеют вид²

$$\begin{aligned} T_j &= \overline{\prod_{i \in V_j, i=j} c_i + b_j \bigcup_{i \in V_j, i \neq j} c_i}, \\ c_j &= \overline{\prod_{i \in V_j, i=j} b_i + T_j \bigcup_{i \in V_j, i \neq j} c_i}, \\ b_j &= \overline{\prod_{i \in V_j, i=j} a_i + c_j \bigcup_{i \in V_j, i \neq j} a_i}. \end{aligned} \quad (10)$$

¹ Равно как и для всех других, описанных ранее типов синхро-стратума.

² Дополнительной проблемой здесь, как и в случае (9), является отображение – представление функции как суперпозиции базисных элементов, что выходит за рамки этой статьи. Заметим только, что для случая (9) эта задача принципиально проще.

Заключение

Теперь уместно задать себе несколько вопросов. Первый из них: «Что дало нам введение синхро-стратума?» Известно, что декомпозиция как таковая не может открыть никаких новых функциональных возможностей. Действительно, мы всегда можем рассматривать композицию автомата A_j со множеством внутренних состояний $\{S_{ij}\}$ и ячейки синхро-стратума T_j со множеством внутренних состояний $\{X_{ij}\}$ как один автомат W_j со множеством внутренних состояний $\{Y_{ij}\} = \{S_{ij}\} \times \{X_{ij}\}$ и соответствующей функцией переходов. Более того, используя изоэдренные процедуры размещения состояний и минимизации автоматов, мы можем вообще завуалировать наличие синхро-стратума в исходном представлении. Однако при этом для каждого массива клеточных автоматов мы должны решать задачу заново. Весь смысл декомпозиции системы на автоматный стратум и синхро-стратум состоит в создании возможности разделения задач организации функционального и временного поведения³. Именно такое разделение позволило нам фактически доказать⁴, что по любому синхронному прототипу массива клеточных автоматов единообразным путем может быть построен его асинхронный аналог. Это инженерный аспект решаемой задачи. С концептуальной точки зрения синхро-стратум является носителем логического времени и создает в системе поле локальных логических времен. Понятие синхро-стратума открывает широкие и еще до конца не использованные возможности для размышлений и спекуляций на тему искусственного системного времени.

Второй, не менее важный вопрос: «Является ли наличие синхронного прототипа и синхро-стратума достаточным для построения асинхронного аналога?» На этот вопрос с определенностью можно ответить: «Нет!» Действительно, асинхронное поведение должно быть причинно-обусловленным, а это означает, что синхронизируемые блоки (автоматы) должны иметь интерфейс с синхро-стратумом, обеспечивающий прямое или опосредованное (косвенное) взаимодействие типа «запрос–ответ» (см. рис. 8). В этом интерфейсе сигнал локального логического времени может и должен рассматриваться как сигнал «запрос» и, получив его, блок (автомат) должен формировать сигнал «ответ», разрешающий синхро-стратуму формировать следующий шаг локального времени. Как и какие дополнения или изменения должны быть сделаны, чтобы блок синхронного прототипа мог быть использован во взаимодействии с синхро-стратумом, – это вопрос конкретного проектирования. Проектировщик может

³ Такое разделение является одной из привлекательных черт синхронных моделей.

⁴ Хотя такое утверждение в явном виде в статье не содержится.

использовать широкий спектр возможных решений, от самосинхронизации до параллельной встроенной задержки. Важно одно: задача может быть решена децентрализованно независимыми и, быть может, различными способами для разных блоков. Такая свобода в решении задачи организации локального временного интерфейса позволяет наряду с известными системами синхронизации: FS – полностью синхронная (общие часы), FA – полностью асинхронная (самосинхронная и др.), LAGS – локально асинхронная – глобально синхронная, GALS – глобально асинхронная – локально синхронная, выделить еще одну: GALA – глобально асинхронная – локально произвольная (система с синхро-стратумом).

И, наконец, последний вопрос: «Может ли любая задача, решаемая синхронным массивом из клеточных автоматов, быть решена соответствующим массивом с синхро-стратумом?».

Ответ на этот вопрос связан с трактовкой и пониманием понятия системного времени. Одна из трудных задач для асинхронного массива клеточных автоматов – это задача взаимодействия фронтов волн, распространяющихся во встречных направлениях, при реализации волновых алгоритмов. Хотя эта задача в принципе решаемая¹, при попытках ее прямого решения в структуре асинхронных массивов² авторам не всегда удается избежать конфликтных³ ситуаций. Введение синхро-стратума снимает проблему асинхронного проектирования и, если алгоритм взаимодействия реализуем в синхронном прототипе, то он автоматически реализуем в асинхронном варианте. Однако, поскольку в синхронном варианте одновременность в физическом времени понимается как одновременность в логическом времени⁴, то, по видимому, задачи, так или иначе связанные с синхронизацией в физическом времени, в принципе не могут быть перенесены из синхронных массивов на асинхронные. Типичным примером является задача Дж. Майхилла о синхронизации цепи стрелков [16, 17]. Для синхронного массива кле-

точных автоматов задача формулируется следующим образом.

Имеется одномерный массив из автоматов Мура с двухсторонним взаимодействием. Каждый из N автоматов имеет n внутренних состояний (n не зависит от N) и в начальный момент времени $T = 0$ все автоматы находятся в пассивном состоянии S_0 . В момент времени $T = 1$ на крайний автомат массива поступает внешний инициализирующий сигнал. В момент времени $T = 2N$ все автоматы должны одновременно перейти в финальное состояние⁵ S_f .

Все известные решения этой задачи базируются на последовательном делении отрезков массива пополам путем сравнения скоростей распространения сигналов. Поскольку скорости распространения сигналов (задержки сигналов в автоматах) измеряются в единицах логического времени, то переход от синхронного прототипа к асинхронному массиву прост и очевиден. Однако момент синхронизации (перехода каждого автомата в финальное состояние) происходит при значении его локального времени, равном $2N$, а мы уже отмечали выше, что в асинхронном массиве одни и те же значения локального логического времени могут существовать в различные моменты физического времени, и задача теряет свой первоначальный смысл.

Приведенная задача теряет свой первоначальный смысл для внешнего наблюдателя, являющегося частью внешнего физического мира с единым физическим временем. Однако, если внешним наблюдателем является искусственная система со своей логической организацией, то могут быть найдены осмысленные интерпретации асинхронного в физическом времени решения этой задачи. Так или иначе проблема синхронизации цепи стрелков демонстрирует те трудности, которые возникают или могут возникнуть при организации в реальном времени интерфейса между искусственной системой и внешним физическим миром. Но это уже тема другой статьи.

Автор пользуется приятной возможностью выразить свою искреннюю признательность профессору К. Неханиву (Университет Айдзу, Япония) за приглашение прочитать на рабочем семинаре «Полугруппы и инженерная алгебра» лекцию, которая легла в основу этой статьи, доктору Т.А. Чу (Acorn Networks Inc., USA) и профессору В. Мараховскому (Университет Айдзу, Япония), с которыми автор в течение нескольких лет работал над проблемами синхро-стратума, а также В. Смоленскому (Университет Айдзу, Япония) за огромную помощь в подготовке этой статьи.

¹ И это следует из нашего изложения (см. также [15]).

² Как, например, в случае counter-flow архитектуры [18].

³ Арбитражных.

⁴ С точностью до длительности синхротакта.

⁵ Вся цепь стрелков должна одновременно выстрелить.

Литература

1. Reichenbach H. The Direction of Time / Ed. M. Reichenbach. – University of California Press, Berkeley and Los Angeles, 1958.
2. Reichenbach H. The philosophy of space & time. – New York, 1958.
3. Von Wright G.H. Explanation and Understanding. – London, 1971.
4. Yang Z., Marsland T. Global States and Time in Distributed Systems. – IEEE Computer Society Press, Los Alamos, Ca., 1994.
5. Muller D. E. A theory of asynchronous circuits // Report of University of Illinois. – 1955. – N 66.
6. Muller D. E., Bartky W. C. A theory of asynchronous circuits. I. IL // Report of University of Illinois. 1956. – N 75; 1957. – N 78.
7. Miller R. E., Switching Theory. Sequential circuits and machines. Vol. II. – John Wiley & Sons. Inc., 1966.
8. Varshavsky V. Hardware support of parallel asynchronous processes. – Helsinki University of Technology, Digital Systems Laboratory, Series A: Research Reports. – N 2, Sept. 1987.
9. Self-timed control of concurrent processes / Ed. V. Varshavsky. – Kluwer Academic Publishers, 1990.
10. Kishinevsky M., Kondratiev A., Taubin A., Varshavsky V. Concurrent hardware. – John Wiley & Sons, 1994.
11. Varshavsky V., Chu T. A. Self-timing – tools for hardware support of parallel, concurrent and event-driven process control // Proceedings of the Conference on Massively Parallel Computing Systems (MPCS'94), May 1994. – P. 510–515.
12. Varshavsky V., Marakhovsky V., Chu T. A. Logical timing (Global synchronization of asynchronous arrays) // Parallel Algorithm/Architecture Synthesis International Symposium, Aizu-Wakamatsu, Japan. IEEE CS Press. – March 1995. – P. 130–138.
13. Varshavsky V. asynchronous interaction in massively parallel computing systems // IEEE First ICA3PP, Proceedings of IEEE First International Conference on Algorithms and Architectures for Parallel Processing, Vol. 2. – Brisbane, Australia. – P. 951–953.
14. Varshavsky V., Marakhovsky V., Chu T. A. Asynchronous timing of arrays with synchronous prototype // Proceedings of the Second International Conference on Massively Parallel Computing Systems (MPCS'96), May 1996. – P. 47–54.
15. Varshavsky V., Marakhovsky V. Global synchronization of asynchronous arrays in logical time // Proceedings of the Second Aizu International Symposium on Parallel Algorithm / Architecture Synthesis, Aizu-Wakamatsu, Japan. IEEE CS Press, March 1997. P. 207–215.
16. Goto E. A minimum time solution of the firing squad problem // Dittoed course notes for applied mathematics 298. – Harvard University, May 1962.
17. Varshavsky V., Marakhovsky V., Peschansky V. Synchronization of Interacting Automata, Math. System Theory. – 1970. – Vol. 4. – N 3.
18. Sproull R. F., Sutherland I. E., Molnar C. E. Counter flow pipeline processor architecture // Technical Report SMLI TR-94-25, SUN Microsystems Laboratories Inc. CA 94043, April 1994.
19. Varshavsky V. Logic design and quantum challenge // Proceedings of International Workshop on Physics and Computer Modeling of Devices Based on Low-Dimensional Structures. Aizu-Wakamatsu, Japan. IEEE CS Press. – Nov. 1995.