

ВЫЧИСЛИТЕЛЬНЫЙ МАКРОКОНВЕЙЕР С ПЕРЕМЕННЫМ ТАКТОМ РАБОТЫ

Г.В. Зайцев

Введение

Конвейеризация является одним из основных способов повышения производительности вычислительных систем и широко используется на всех уровнях структурной иерархии устройств. В данной работе

рассматривается верхний уровень указанной иерархии, а конвейерная структура этого уровня называется макроконвейером. Анализируемая система (рис. 1) содержит ряд последовательно соединенных процессоров (Пр), разделенных буферными запоминающими устройствами. Каждый из процессоров является вычислительным устройством, содержащим при необходимости внутреннее рабочее запоминающее устройство (ЗУ). На вход макроконвейера поступают последовательно во времени блоки

Рассматривается вычислительная система с конвейеризацией на уровне задач, содержащая несколько последовательно соединенных процессоров с буферными запоминающими устройствами (БЗУ) между ними. Предлагаются алгоритмы управления конвейером и рассчитывается необходимый объем БЗУ. Анализируются два способа построения БЗУ: в виде двухпортовой памяти и в виде памяти из нескольких банков. Рассматриваются временные диаграммы прохождения задач по конвейеру.

данных, относящиеся к последовательно решаемым независимым задачам. Над каждым блоком данных в каждом процессоре производятся необходимые операции обработки. В принципе, БЗУ могут отсутствовать, однако, если процессоры полностью используют как объем, так и быстродействие внутренних ЗУ, то отсутствие БЗУ приводит к резкому снижению пропускной способности макроконвейера. Поэтому в данной работе рассматривается конфигурация макроконвейера, содержащего БЗУ.

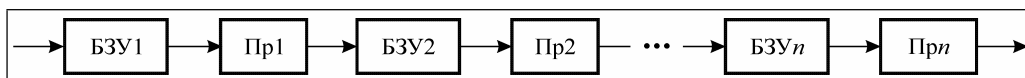


Рис. 1. Структурная схема макроконвейера

Организация БЗУ не вызывает затруднений в случае, когда макроконвейер решает одну и ту же повторяющуюся задачу над последовательно поступающими блоками данных. При этом обычно каждое БЗУ строится в виде ЗУ с двумя банками, один из которых принимает данные от предыдущего процессора, а второй выдает данные на последующий процессор. Для следующего блока данных (следующей задачи) банки меняются местами.

Однако в ряде случаев возникает необходимость обработки в макроконвейере различных задач, требующих различного времени решения в процессорах. Организация БЗУ при этом несколько усложняется и является предметом рассмотрения настоящей работы.

Приведем один из примеров подобного макрокон-

вейера, относящийся к области цифровой обработки информации в радиолокационных станциях (РЛС). Возможная структура устройства обработки принимаемых сигналов приведена на рис. 2. На вход аналого-цифрового преобразователя (АЦП) поступают прошедшие необходимую аналоговую обработку сигналы. Информация от одного зондирования накапливается в БЗУ1 и далее передается в программируемый процессор сигналов (ППС), выполняющий его первичную обработку (оптимальную фильтрацию сигнала и выделение отметок, превысивших порог). Результаты передаются через БЗУ2 в процессор вторичной обработки (ПВО), реализующий алгоритмы работы с обнаруженными отметками (обнаружение объектов, завязка трасс, сопровождение).



Рис. 2. Макроконвейер обработки сигнала в РЛС

Требуемая производительность и пропускная способность указанных процессоров существенно различны, также, как и характер выполняемых алгоритмов. В связи с этим они, как правило, реализуются в различных устройствах, образующих макроконвейер рис. 2.

Поток сигналов на вход такого устройства определя-

ется сменой режимов зондирования РЛС и в современных многофункциональных РЛС является довольно неравномерным. На рис. 3 приведен типичный фрагмент временной диаграммы работы импульсной РЛС с фазированной антенной решеткой. На верхней диаграмме показаны излучаемые передатчиком импульсы, на ниж-

ней – интервалы приема (набора информации в БЗУ1). Перед каждым тактом зондирования луч антенны устанавливается в необходимое угловое положение. В такте 1 зондирования производится обзор пространства с помощью импульса с линейной частотной модуляцией.

Период такого обзорного зондирования определяется максимальной дальностью РЛС и, как правило, довольно значителен, а интервал набора информации в БЗУ1 охватывает всю просматриваемую дальность.

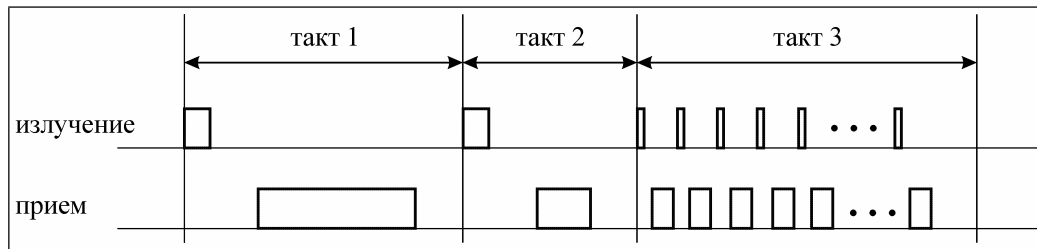


Рис. 3. Временная диаграмма работы РЛС

Такт 2 зондирования используется для сопровождения ранее обнаруженного объекта с помощью импульса с фазокодовой манипуляцией. При небольшой дальности до объекта длительность этого такта существенно меньше, чем такта 1, а интервал набора информации, в силу известной дальности, также невелик.

Такт 3 используется для сопровождения низко летящего объекта с помощью квазинепрерывного сигнала с высокой частотой повторения, позволяющего при большой длительности такта эффективно подавить отражения от местных предметов. При этом внутри такта 3 присутствует целый ряд интервалов набора информации в соответствии с числом излучаемых импульсов. Таким образом, на вход макроконвейера приемного устройства через неодинаковые интервалы времени поступают массивы информации различного объема, требующие различной обработки.

Очевидный и часто используемый способ организации такого конвейера состоит в требовании к процессорам обеспечить решение всех задач за длительность минимального такта зондирования. Однако это может более, чем на порядок, увеличить необходимую производительность процессоров, которые к тому же большую часть времени будут простаивать. В данной работе предлагается более экономный способ организации описанного макроконвейера.

Приведенные ниже методы организации макроконвейера используются с середины 80-х годов в текущих разработках процессоров радиолокационных сигналов.

Базовая модель макроконвейера

Рассмотрим модель описанного выше макроконвейера, являющуюся основой последующего анализа. Для сокращения текста в дальнейшем изложении термин конвейер используется в качестве синонима термина макроконвейер. Назовем этапом конвейера БЗУ с последующим за ним процессором, а временной интервал, за который обрабатываются данные некоторой задачи в любом устройстве конвейера, – соответствующим тактом работы этого устройства. Более точно, тактом работы процессора, относящимся к некоторой задаче, назовем временной интервал, за который данные этой задачи обрабатываются в рассматриваемом процессоре, включая время приема входных данных и выдачи результа-

тов. Тактом работы БЗУ по записи (соответственно, считыванию), относящимся к решению некоторой задачи, назовем временной интервал, в течение которого может производиться запись (считывание) данных этой задачи. Очевидно, такты работы любого БЗУ по записи (считыванию) должны согласовываться с тактами работы предыдущего (последующего) процессоров конвейера.

В частности, в каждый такт работы входного устройства (входной такт) на вход конвейера поступают входные данные очередной задачи. При этом различные задачи будем считать независимыми и никак не связанными между собой. Например, для конвейера приемного устройства такты, показанные на рис. 3, являются входными тактами.

Рассматриваемая модель конвейера описывается следующими условиями.

У1. Длительность входных тактов конвейера может принимать любое значение от нуля до T_m .

У2. Длительность тактов обработки информации фиксированной задачи одинакова для всех процессоров конвейера и равна длительности входного такта этой задачи.

У3. Максимально возможный объем V_i информации, поступающей на вход i -го этапа конвейера и относящейся к фиксированной задаче, пропорционален длительности T входного такта этой задачи, т. е.

$$V_i = k_i T, \quad i = 1, 2, \dots, n,$$

где k_i – константы, а n – количество этапов конвейера.

У4. Обработка данных фиксированной задачи в процессоре некоторого этапа конвейера начинается только после окончания ее обработки на предыдущем этапе и после окончания обработки предыдущей задачи в рассматриваемом процессоре.

Условие У1 говорит о том, что произвольно длинные входные такты невозможны. В противном случае объем входного БЗУ стал бы бесконечным, что не имеет смысла. Условие У2 является основой конвейерного режима работы, обычно для конвейерных систем и обеспечивает минимально возможную производительность процессоров. Условие У3 о пропорциональности объема входной информации длительности такта является хорошим приближением для ряда систем, например, для описанного выше конвейера приемного устройства. Очевидно, условиями 3 могут быть описаны любые конвейеры при соот-

ветствующем выборе констант k_i .

Условие У4 продиктовано стремлением к упрощению управляющих устройств процессоров. Отказ от условия У4 может привести к конвейерному способу обработки задачи сразу на нескольких процессорах. Этот случай может быть сведен к рассматриваемому путем объединения таких процессоров в один укрупненный процессор.

В последующих разделах обсуждается установившийся режим работы конвейера, описываемый следующими условиями.

У5. На входе конвейера такты следуют друг за другом без интервалов между ними.

У6. До рассматриваемого интервала времени закончена обработка хотя бы одной задачи с длительностью входного такта T_m .

Эти условия в сочетании с условиями У2, У3 приводят к полной и постоянной загрузке процессоров конвейера. Кроме того, следствием условия У6 является тот факт, что задержка информации любой задачи на каждом этапе конвейера не может быть меньше T_m . Действительно, при решении задачи с длительностью входного такта T_m , эта задержка, по условию У2, имеет величину не менее T_m , а в силу полной и постоянной загрузки процессоров не может сократиться и для всех последующих задач.

С другой стороны, очевидно, что задержки величиной T_m достаточно для реализации конвейерного режима. Увеличение этой задержки сверх минимально необходимой приводит к увеличению необходимого объема БЗУ. Поэтому дальнейшее рассмотрение установившегося режима проводится при следующем предположении.

У7. В установившемся режиме задержка информации на каждом этапе конвейера равна минимально возможной величине T_m .

Рассмотрим теперь способы управления конвейером и необходимый объем БЗУ. При этом ограничимся только такими способами построения БЗУ, при которых БЗУ всегда доступно предыдущему процессору на запись, а следующему – на считывание. Это условие приводит к отсутствию конфликтов при обращении к БЗУ и максимизирует пропускную способность конвейера.

Проанализируем сначала установившийся режим работы, так как именно он предъявляет максимальные требования к объему БЗУ.

Двухпортовое БЗУ

Первый очевидный способ организации БЗУ состоит в использовании двухпортового ЗУ с независимыми обращениями по входу (запись) и выходу (считывание). При этом алгоритм работы БЗУ элементарен и состоит в следующем.

Алгоритм 1. Запись. Записываемая информация последовательно и циклически заполняет имеющийся объем БЗУ.

Считывание А. Считывание в процессор из БЗУ информации очередной задачи начинается немедленно после выполнения условия У4: обработка данной задачи закончена в предыдущем процессоре и обработка предыдущей задачи закончена в рассматриваемом процессоре.

Считывание С. Такт считывания из БЗУ очередной задачи начинается через время T_m после начала записи этой задачи.

Для установившегося режима оба способа считывания эквивалентны из-за полной и постоянной загрузки процессоров. Разделение этих способов необходимо при рассмотрении переходных режимов. Далее будет показано, что способ считывания А (соответственно, способ С) определяет асинхронный (синхронный) режим работы конвейера, откуда и происходит мнемоническое обозначение этих способов.

Единственной задачей при использовании двухпортового БЗУ является обеспечение достаточного его объема.

Утверждение 1. При использовании двухпортового ЗУ в качестве БЗУ i -го этапа конвейера необходимый объем БЗУ составляет не более $3k_i T_m$, $i = 1, 2, \dots, n$.

Для доказательства рассмотрим временные диаграммы тактов записи и считывания i -го БЗУ, приведенные на рис. 4. Каждый временной такт обозначен прямоугольником соответствующей длительности, а его номер указан цифрой внутри прямоугольника. Подобные же обозначения используются и на последующих рисунках. В установившемся режиме эти диаграммы имеют одинаковый вид и отличаются лишь задержкой на T_m . Поэтому можно пользоваться одной временной диаграммой, например, диаграммой записи. При этом состояние по считыванию в момент t будет определяться диаграммой записи для момента $t - T_m$.

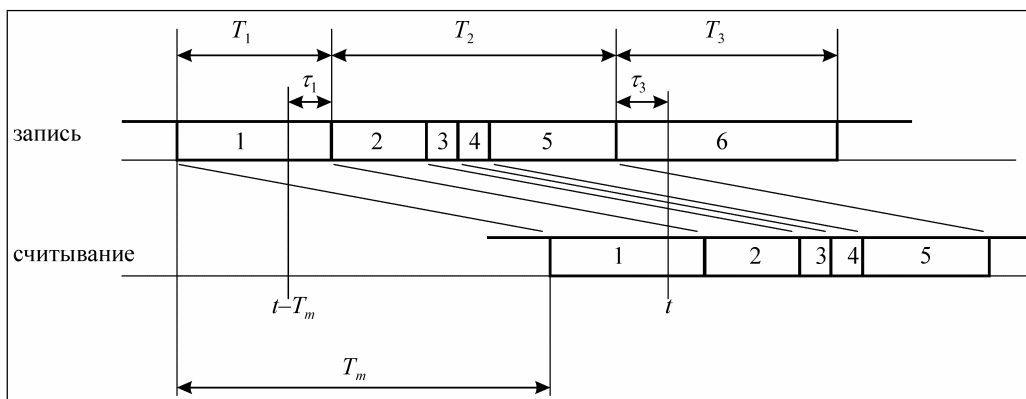


Рис. 4. Временная диаграмма макроконвейера

Определим объем информации V_t , находящийся в буфере в произвольный момент t . Он равен объему записанной к этому моменту информации W_t минус объем считанной информации R_t , причем такты, для которых считывание к рассматриваемому моменту полностью закончено, можно не учитывать. Пусть T_1 – длительность текущего такта считывания, T_2 – длительность целых тактов за интервал времени $[t - T_m, t]$, а T_3 – длительность текущего такта записи (рис. 4). Обозначим через τ_1 и τ_3 длительность отрезков тактов T_1 и T_3 , попавших внутрь интервала $[t - T_m, t]$. Тогда, обозначая через $W(x)$ и $R(x)$ соответственно объем записанной и считанной информации за интервал x , имеем

$$V_t = W_t - R_t = W(T_2) + W(T_1) + W(\tau_3) - R(T_1 - \tau_1). \quad (1)$$

Очевидно, при произвольном порядке записи и считывания $W(\tau_3) \leq W(T_3)$, а $R(T_1 - \tau_1) \geq 0$. Поэтому $V_t \leq W(T_2) + W(T_1) + W(T_3)$, и, согласно условию У2, каждое слагаемое не превосходит $k_i T_m$. Следовательно, $V(t) \leq 3k_i T_m$, что и требовалось доказать.

Из приведенного доказательства видно, что весь объем БЗУ используется лишь при чрезвычайно неблагоприятных обстоятельствах, а именно, когда интервалы T_1, T_2, T_3 близки к T_m . Предыдущий процессор, решающий задачу такта T_3 , почти все результаты записывает в БЗУ за интервал τ_3 , а последующий – большую часть входных данных такта T_1 считывает в конце этого такта за интервал τ_1 . Тем не менее, объем БЗУ близкий к $3k_i T_m$ в общем случае может понадобиться. Достоинством такого варианта является возможность использования процессором частей предшествующего и последующего БЗУ, выделяемых на задачу текущего такта, на протяжении всего этого такта как своей оперативной памяти.

Объем БЗУ можно существенно сократить, наложив минимальные условия на порядок записи и/или считывания. Назовем процедуру считывания из БЗУ i -го этапа конвейера равномерно ограниченной, если за интервал τ от начала такта из БЗУ считывается объем информации не менее, чем $k_i \tau$, $i = 1, 2, \dots, n$. Аналогично процедуру записи в БЗУ i -того этапа конвейера равномерно ограниченной, если за интервал τ от начала такта в БЗУ записывается объем информации не более, чем $k_i \tau$, $i = 1, 2, \dots, n$.

Утверждение 2. При равномерно ограниченных процедурах записи и считывания в БЗУ i -го этапа конвейера необходимый объем двухпортового БЗУ составляет $k_i T_m$. В случае равномерной ограниченности одной из процедур записи или считывания необходимый объем двухпортового БЗУ составляет $2k_i T_m$, $i = 1, 2, \dots, n$.

Для доказательства снова рассмотрим временную диаграмму рис. 4 и равенство (1). В силу равномерной ограниченности процедуры считывания имеем

$$R(T_1 - \tau_1) \geq k_i (T_1 - \tau_1), \quad (2)$$

а в силу равномерной ограниченности процедуры записи –

$$W(\tau_3) \leq k_i \tau_3. \quad (3)$$

Подставляя (2) и (3) в (1), получаем:

$$V_t \leq k_i (T_2 + T_1 + \tau_3 - T_1 + \tau_1) = k_i T_m.$$

Аналогично доказывается и вторая часть утверждения 2. Равномерно ограниченные процедуры записи и считывания естественным образом организуются в случае, когда внутренние ЗУ предшествующего и последующего процессоров достаточно велики, чтобы для любого такта ввести все входные данные и произвести их обработку без обращения к БЗУ. В этом случае любой такт можно начать со считывания из БЗУ всей входной информации, затем произвести полностью ее обработку и лишь затем вывести выходные результаты. При такой обработке процедуры записи и считывания, очевидно, равномерно ограничены.

В качестве примера рассмотрим конвейер обработки сигнала в РЛС (рис. 2). Входное БЗУ1, принимающее данные от АЦП, должно иметь объем не менее $2k_1 T_m$, так как процедура записи определяется внешними физическими условиями и не может быть равномерно ограничена. БЗУ2 между ПК0 и ПНО может иметь объем $k_2 T_m$ при соответствующей организации алгоритмов обработки, находящейся в руках разработчика.

БЗУ из нескольких банков однопортовых ЗУ

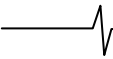
При организации БЗУ из нескольких банков однопортовых ЗУ по-прежнему необходимо обеспечить возможность независимого обращения к БЗУ по записи и по считыванию для выполнения условий работы конвейера. При произвольных процедурах записи и считывания это означает, что к некоторому банку можно обратиться по считыванию только тогда, когда запись в него закончена и производится в другой банк. Отсюда следует, что объем каждого банка БЗУ i -го этапа конвейера не имеет смысла делать больше $k_i T_m$, $i=1, 2, \dots, n$. Действительно, при записи в один банк большего объема информации задержка между записью и считыванием превысит T_m , что противоречит условию У7.

Рассмотрим вопрос о необходимом количестве банков объемом $k_i T_m$, $i=1, 2, \dots, n$. При этом напомним, что в данном пункте рассматривается лишь установившийся режим. Для упрощения устройства управления и удобства оперирования с банками положим, что информация любого банка должна быть записана в один банк.

Утверждение 3. Для организации БЗУ i -го этапа конвейера достаточно трех банков однопортовых ЗУ, каждый объемом $k_i T_m$, $i=1, 2, \dots, n$.

Доказательство проведем конструктивным путем, рассмотрев алгоритм работы описанного БЗУ, не приводящий к конфликтам при обращении к нему. При этом для краткости назовем указателем записи $w(t)$ (соответственно, указателем считывания $r(t)$) целочисленную функцию, равную в момент t номеру банка, в который должно происходить текущее обращение по записи (соответственно, по считыванию).

Алгоритм 2. Запись. Поступающая информация последовательно заполняет объем каждого банка.



Указатель записи может изменяться лишь в момент смены тактов записи. При смене тактов указатель записи не изменяется, если время от начала первого такта записи в данный банк до конца начинающегося такта не превышает величины T_m ; в противном случае указатель записи получает приращение 1 (mod 3).

Способы считывания А и считывания С идентичны таковым для алгоритма 1.

Переходя к доказательству утверждения 3, заметим, что для отсутствия конфликтов в БЗУ необходимо и достаточно, чтобы в любой момент времени выполнялось неравенство $w(t) \neq r(t)$. Причем, достаточно рассмотреть лишь моменты изменения функций $w(t)$ и $r(t)$, т.е. моменты смены тактов по записи и по считыванию.

Моменту изменения $r(t)$ с i на $i+1(\text{mod } 3)$ соответствует временная диаграмма рис. 4 с $\tau_1=0$. Тогда, если $\tau_3=0$, то за интервал $(t - T_m, t)$ информация, согласно алгоритму 2, записывалась в банк $i+1(\text{mod } 3)$, полностью его заполнила и $w(t)=i+2(\text{mod } 3)$. Если же $\tau_3>0$, то информация текущего такта записи не размещается в банке $i+1(\text{mod } 3)$, и $w(t)$ переключился на $i+2(\text{mod } 3)$ в начале текущего такта записи. В любом случае в момент изменения $r(t)$ с i на $i+1(\text{mod } 3)$ выполняется неравенство $w(t) \neq r(t)$.

Моменту изменения $w(t)$ с i на $i+1(\text{mod } 3)$ соответствует временная диаграмма 4 с $\tau_3=0$. При этом за интервал $(t - T_m, t)$ не могло быть более одного изменения $w(t)$, так как с момента первого такого изменения до момента t информация может быть размещена в одном банке и по алгоритму 2 так и записывается. Тогда $r(t) = i-1(\text{mod } 3)$, если таковое переключение было внутри интервала $(t - T_m, t)$, и $r(t)$ меняет свое состояние с $i-1(\text{mod } 3)$ на $i(\text{mod } 3)$, если таковое переключение было в момент $t - T_m$. В любом случае $w(t) \neq r(t)$. Утверждение доказано.

При описанном способе организации БЗУ так же, как и для утверждения 1, части входного и выходного БЗУ для i -того процесса, относящиеся к текущему такту его работы, могут использоваться процессором на протяжении всего такта как собственная оперативная память.

Из приведенных рассуждений легко следует, что двух банков БЗУ, вообще говоря, недостаточно. В

некоторых более частных случаях за счет усложнения аппаратуры управления конвейером и накладывания довольно жестких условий на соотношения скоростей записи и считывания БЗУ можно обойтись и двумя банками. Однако в данной работе эти варианты из-за их специфичности рассматриваться не будут.

Переходные режимы работы макроконвейера

До сих пор рассматривался лишь установившийся режим работы конвейера, описываемый условиями У5–У7 и характеризующийся полной и постоянной загрузкой всех процессоров при задержке информации на каждом этапе конвейера на время T_m . К переходным режимам относятся начало и окончание работы конвейера, а также ситуации, возникающие при наличии на входе свободных от работы временных интервалов («пустых» тактов). Способы организации конвейера, рассмотренные ранее и описанные там алгоритмы и утверждения об объемах БЗУ остаются справедливыми, как нетрудно проверить, и для переходных режимов, так как установившийся режим предъявляет наибольшие требования как к производительности процессоров, так и к объему БЗУ.

Тем не менее, переходные режимы имеют свои особенности. Рассмотрим, например, начало работы конвейера. Временная диаграмма начала работы зависит от способа считывания в алгоритмах 1 и 2. При считывании типа А в случае, когда первые входные такты имеют длительность меньше T_m , задержка на каждом этапе конвейера также меньше T_m и определяется задачей наиболее длинного такта, прошедшей данный процессор (временная диаграмма рис. 5 для двухэтапного конвейера). При появлении более длинного такта задержка на каждом этапе возрастает, а в работе процессора появляются свободные интервалы (рис. 5). При считывании типа С задержка на каждом этапе фиксирована и сразу входит в установившийся режим (рис. 6). Назовем поэтому конвейер со считыванием типа А во всех БЗУ асинхронным, а со считыванием типа С – синхронным конвейером. Таким образом, для синхронного конвейера временные диаграммы тактов на всех этапах конвейера имеют одинаковый вид и отличаются лишь задержкой.

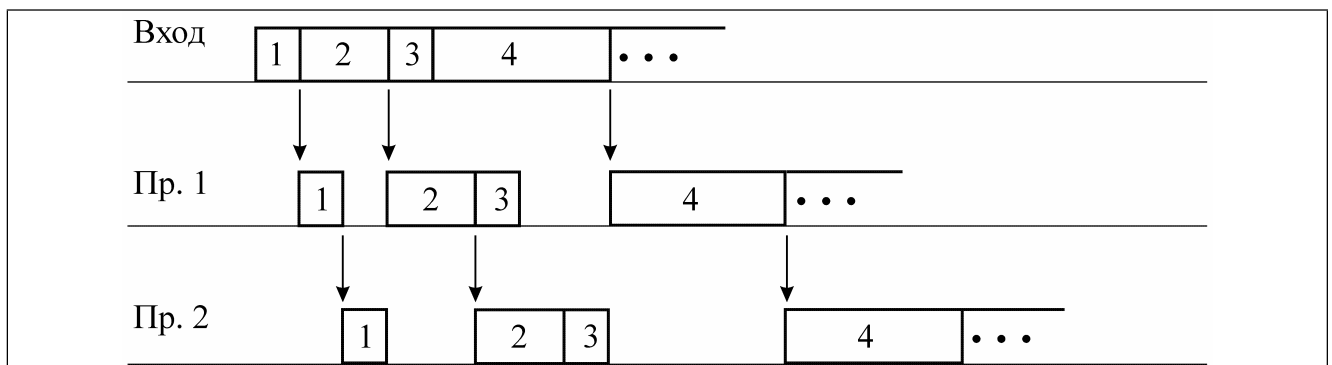


Рис. 5. Начало работы асинхронного конвейера

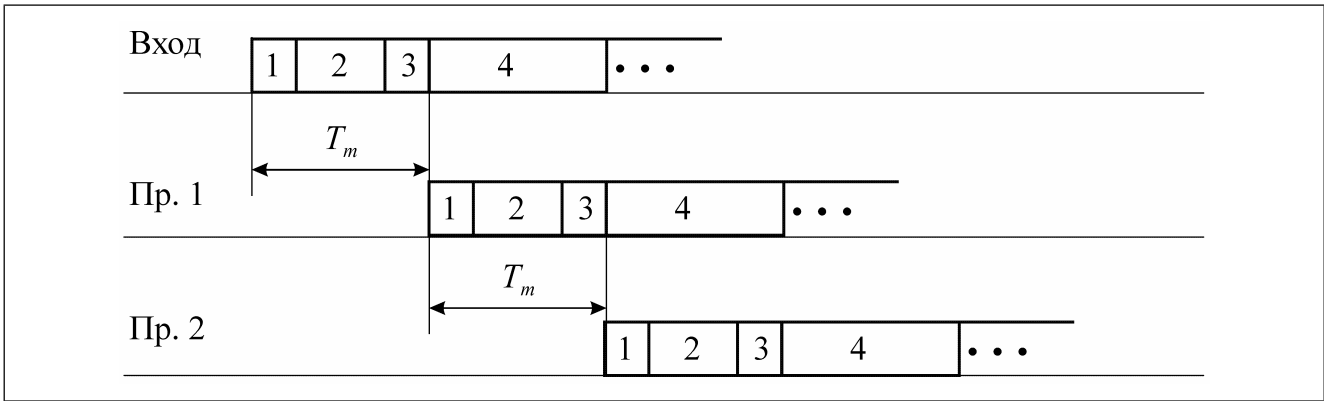


Рис. 6. Начало работы синхронного конвейера

Аналогичный эффект наблюдается и при наличии на входе конвейера свободных от работы тактов. В синхронном конвейере они передаются на выход без изменений с задержкой nT_m , где n – число этапов конвейера. В асинхронном конвейере свободные интервалы могут привести к сокращению задержки для последующих задач (рис. 7). При этом для со-

кращения задержки с nT_1 до nT_2 необходимо, чтобы свободные интервалы на входе имели суммарную длительность не менее $n(T_1 - T_2)$, $T_2 < T_1$, однако задачи между ними должны располагаться таким образом, чтобы позволить уменьшить задержку на каждом этапе на $T_1 - T_2$ (рис. 8).

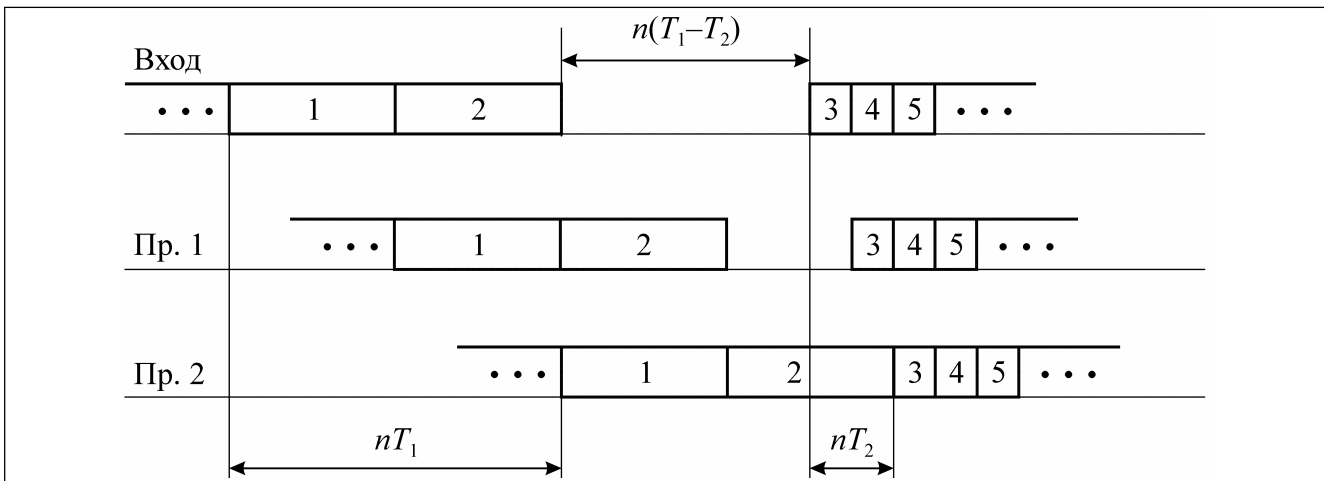


Рис. 7. Сокращение задержки в асинхронном конвейере за счет свободного интервала

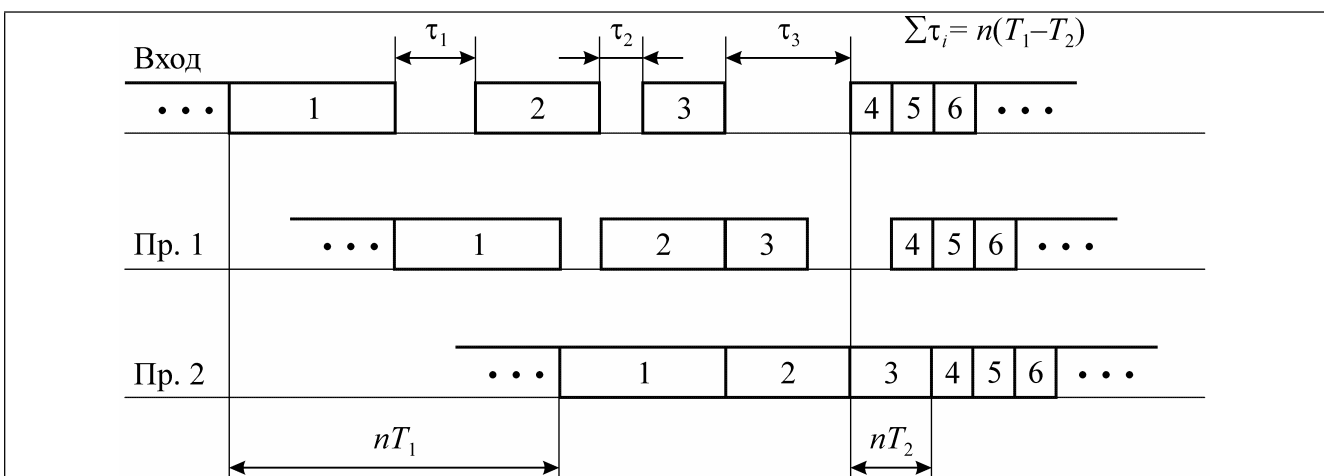


Рис. 8. Сокращение задержки в асинхронном конвейере за счет нескольких свободных интервалов

Таким образом, в асинхронном конвейере автоматически минимизируется время прохождения ин-

формации по конвейеру. Однако при этом для данной задачи это время зависит от всей предыстории

конвейера и поэтому трудно прогнозируемо. Наоборот, для синхронного конвейера время прохождения любой задачи фиксировано и равно nT_m , поэтому всегда известно, когда результаты любой задачи появятся на выходе конвейера.

В некоторых приложениях для сокращения задержки в синхронном конвейере бывает необходимо перейти от работы с максимальным тактом T_{m1} к работе с максимальным тактом T_{m2} , $T_{m2} < T_{m1}$. Временная диаграмма такого перехода аналогична временной диаграмме уменьшения задержки в асинхронном конвейере (рис. 7) и требует на входе свободного от работы интервала длительностью не менее $n(T_{m1} - T_{m2})$. Обратный переход от T_{m2} к T_{m1} , $T_{m2} < T_{m1}$, затруднений не вызывает, но приводит к прогрессивно увеличивающемуся вдоль конвейера простоя процессоров (аналогично временной диаграмме рис. 5).

Модификации базовой модели и алгоритмов работы БЗУ

До сих пор обсуждалась модель конвейера, описанная ранее условиями У1–У4. Рассмотрим, к чему приводят небольшие изменения этих условий.

а) Условие У3 о строгой пропорциональности объема информации длительности такта довольно жестко. В случае, когда оно не выполняется, введем для всех задач, обрабатываемых конвейером, коэффициенты

$$k_{ij} = v_{ij} / T_{ij}, \quad i=1,2,\dots,n; j=1,2,\dots \quad (4)$$

где i – номер этапа, j – номер задачи, v_{ij} – объем входной информации i -того этапа для j -той задачи, T_{ij} – время обработки процессором i -того этапа этой задачи. Выберем для каждого этапа максимальный коэффициент

$$k_{im} = \max_j k_{ij}, \quad i = 1,2,\dots,n \quad (5)$$

Тогда, как нетрудно проверить, все ранее изложенное остается справедливым и для рассматриваемого случая при замене k_i на k_{im} . При этом, правда, объем i -го БЗУ будет частично недоиспользоваться при обработке задачи с $k_{ij} < k_{im}$.

б) Рассмотренные структуры макроконвейера работоспособны и при отказе от точного равенства времен обработки фиксированной задачи на всех этапах конвейера (условие У2). Все представленные выводы справедливы и для случая, когда время обработки любых этапов конвейера может быть меньше длительности входного такта.

При этом для синхронного конвейера, по существу, ничего не меняется, за исключением некоторого простоя процессоров на тех этапах, где время обработки меньше, а для асинхронного конвейера может

наблюдаться уменьшение времени задержки последующих задач, аналогичное таковому при появлении свободных интервалов. Подобное ускорение прохождения задач по конвейеру может привести к увеличению необходимого объема БЗУ, так что при расчете коэффициентов k_{im} по формулам (4), (5) за величину T_{ij} необходимо принимать истинное время решения j -ой задачи, в то время как для синхронного конвейера величину T_{ij} достаточно приравнять величине входного такта для этой задачи. В результате объем БЗУ асинхронного конвейера может существенно возрасти, что является его недостатком. Устранение этого недостатка может быть произведено путем отказа от возможности обращения к БЗУ по записи в любой момент времени и введения разрешения на запись в БЗУ из процессора только при наличии достаточного объема свободной памяти для размещения результатов обработки очередной задачи. Однако такой метод приводит к усложнению способа управления конвейером.

в) Отказ от второй части условия У4 приводит к возможности считывания из БЗУ в процессор сразу нескольких задач. Временная диаграмма работы конвейера при этом не изменяется, если каждую такую «пачку» задач рассматривать как укрупненную макрозадачу. Такой режим работы может в некоторых случаях привести к уменьшению необходимого объема БЗУ. Например, если все возможные длительности тактов кратны некоторой величине δ , а за время δ из БЗУ успевают считываться информация любого такта, то при банковской организации БЗУ i -го этапа достаточно иметь 2 банка объемом $k_i T_m$. Однако такие случаи довольно специфичны и поэтому подробнее в данной работе рассматриваться не будут.

Заключение

Таким образом, два простых рассмотренных выше способа организации макроконвейера показывают, что работа с переменным тактом требует некоторого увеличения объема БЗУ в сравнении с макроконвейером с фиксированным тактом. Однако это увеличение не слишком велико (не более, чем в 1,5 раза) и является вполне приемлемой платой за минимизацию требований к производительности процессоров и получаемую гибкость устройства. В связи с этим подобные макроконвейеры должны найти широкое применение в различных областях использования вычислительной техники и, в частности, в трактах обработки сигналов многофункциональных РЛС.