

ИМИТАЦИОННАЯ МОДЕЛЬ КОНВЕЙЕРА В МИКРОПРОЦЕССОРЕ

Современные компьютеры построены на микропроцессорах, в которых широко используется конвейерная обработка команд. Команда – эта инструкция к действию процессора. Большинство команд можно представить в виде двоичного кода с несколькими полями:

<Код операции>
<Адрес операнда 1>
<Адрес операнда 2>
<Адрес результата>

Код операции задает вид действия: сложение, вычитание и другие. Операнды – это данные для операции, например, первое и второе слагаемое, сомножители и т. п. В команде задаются адреса в памяти, где расположены операнды и куда помещается полученный результат.

Задача любого процессора – выполнять команды как можно быстрее. Один из самых эффективных способов заключается в при-

менении конвейера. Идея та же самая, что и в конвейере для сборки автомобилей. Первое – должен быть непрерывный поток команд. Второе – каждая микрооперация должна быть как можно проще. И третье – все микрооперации должны выполняться за одно и то же время. Типичную команду разбирают на пять простых микроопераций:

- 1) ВК – выборка команды из памяти;
- 2) ДК – декодирование команды, то есть определение кода операции и выделение адресов;
- 3) ВО – выборка данных из памяти по заданным адресам;
- 4) ИК – исполнение команды в соответствии с кодом операции и использованием считанных операндов;
- 5) ЗР – запись результата операции по заданному адресу в память.

Соответственно, конвейер содержит пять элементарных ступеней, для каждой микро-

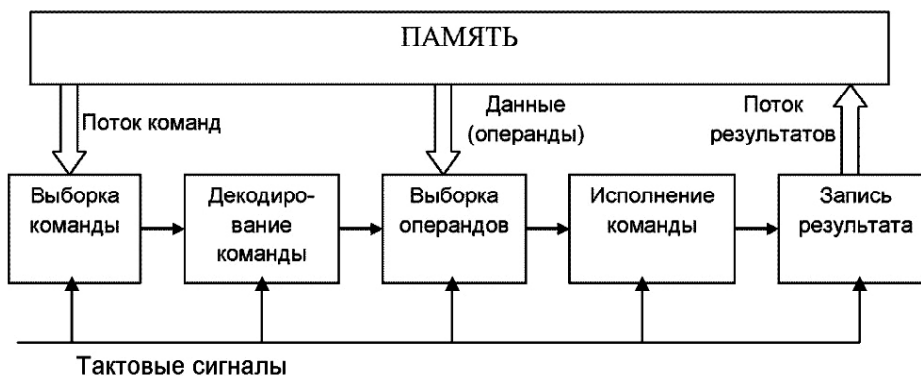


Рис. 1

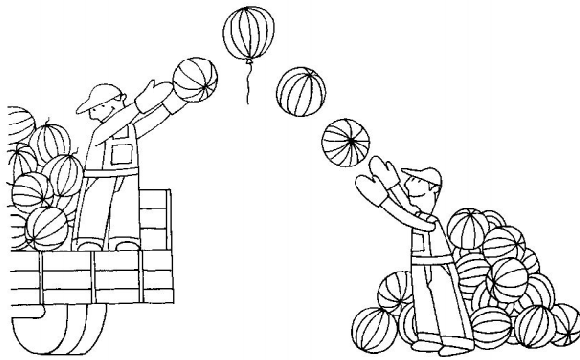
операции (см. рис. 1). Тактовые сигналы сдвигают команду в конвейере вправо на одну ступень и передают полученные промежуточные результаты соседним ступеням.

Для пятиступенчатого конвейера общее время выполнения любой команды остается равным 5-и тактам. Но на выходе результаты получаются в каждом такте. Это происходит, как в автомобильном конвейере, на котором отдельный автомобиль собирают несколько часов, но готовые машины сходят с конвейера каждые 20 минут.

Пусть каждая команда выполняется за время T , а в конвейере имеется N ступеней. Тогда в идеальном конвейере результаты будут появляться через промежутки времени T/N . Это значит, что ускорение в конвейере равно N . Такое утверждение справедливо для непрерывного потока команд. Если конвейер запускается вновь, то надо учитывать время его начальной загрузки, равное числу ступеней N .

Если предположить, что каждая микрооперация выполняется за один такт машинного времени, то реализация последовательности из нескольких (например, трёх) команд с совмещением может быть представлена в виде схемы (см. рис. 2). Из рисунка 2 видно, что результат первой команды будет получен после 5-го такта, второй – после 6-го, а третьей – после 7-го. Время заполнения конвейера равно 5 тактам.

Таким образом, среднее время выполнения последовательности из трех команд будет равно 2.33 такта. С увеличением длины последовательности команд это время



уменьшается и в пределе стремится к 1 такту. Такой эффект наблюдается только при одинаковых длительностях микроопераций. Если эти длительности отличаются, то некоторые устройства будут простаивать, и среднее время выполнения команды увеличится. При большой длине программы (большом количестве команд) это время будет стремиться к максимуму из всех длительностей микроопераций. Например, если все микрооперации выполняются за 1 такт, а одна – за 2 такта, то среднее время выполнения одной команды будет стремиться к 2 тактам.

Эффективность работы любого конвейера определяется величиной разности между средним временем выполнения команды в нем и максимальным временем. Целью предлагаемой работы является исследование влияния длины последовательности команд и соотношения длительностей отдельных микроопераций на эффективность работы конвейера.

В имитационной модели выполнение микроопераций в команде (занятость соответствующих устройств) для наглядности изображено разными цветами, как показано

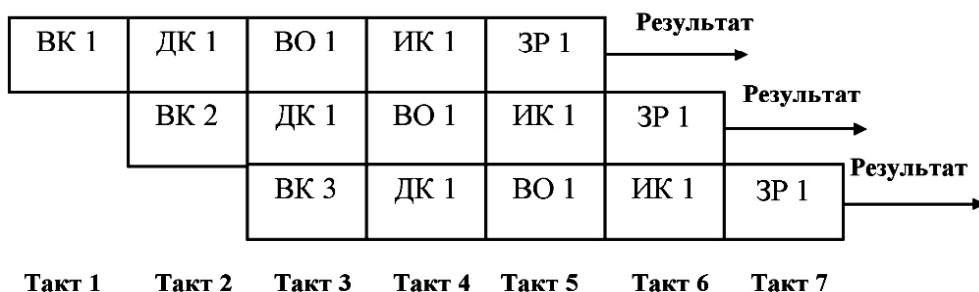


Рис. 2



Рис. 3

на рис. 3, а «простой» устройства представлен черным цветом. Изменение параметров команд и микроопераций сопровождается соответствующим изменением схемы их выполнения, которая выводится на экран (изменяется ширина поля микрооперации).

Исполняемый модуль программы имитационного моделирования имеет имя **Labcv**. Его запуск осуществляется традиционным способом. После запуска на экране появляется основное окно, которое позволяет за-

дать параметры модели и режим моделирования (непрерывный или по тактам).

Исходными данными для программы **Labcv** являются:

а) количество исполняемых команд (от 3 до 10, по умолчанию их 3);

б) длительность любой микрооперации в тактах (от 1 до 30, по умолчанию все значения равны единице).

Все перечисленные значения и режимы моделирования устанавливаются с помощью соответствующих кнопок, как это принято в Windows. Причем, исход-

ные данные можно задать сразу после запуска программы **Labcv**. Нажатие кнопки «**Работа модели**» приводит к выполнению программы в выбранном режиме. Если выбран тактовый режим, то каждый раз для продолжения работы необходимо нажимать на кнопку «**Следующий такт**».

Во время работы модели в обоих режимах на экране цветом представляется реализация команд, выводится количество завершенных на данный момент тактов, среднее

время выполнения команды и суммарный простой устройств в тактах. Работа заканчивается, если выполнены все команды заданной последовательности. При этом программа автоматически завершает работу, устанавливается в режим «**Ввод значений**» и выводит следующие характеристики (см. рис. 3):

- среднее время выполнения одной команды,
- суммарное время ожидания устройств в системе,
- графики зависимости среднего времени выполнения команды от количества команд и от длитель-



Рис. 4

ностей каждой из пяти микроопераций в команде,

– пять графиков зависимости среднего времени ожидания каждого обрабатываемого устройства от длительностей любой из пяти микроопераций в команде.

Перечисленные графики выводятся после нажатия кнопки «Графики» в соответствующие окна, которые можно расположить каскадом и просматривать в любой последовательности, нажав соответствующую кнопку.

Очевидно, что наиболее эффективным является режим работы конвейера, при котором

– все микрокоманды имеют одинаковую длительность;

– обрабатываемые устройства располагаются в порядке, соответствующем порядку следования микрокоманд.

Если одно из этих условий нарушается, то некоторые устройства будут простаивать, а эффективность снизится. Такие ситуации называются *конфликтами*. Они приводят к приостановке выполнения очередной команды (pipeline stall). Эта ситуация называется

«конвейерным пузырьком» (pipeline bubble). Пузырь проходит по конвейеру, не выполняя никакой работы.

Программа **Labcv** позволяет изучить особенности работы конвейера с «пузырем», который приводит к его простоям в течение целой команды. Для этого достаточно на главной форме отметить соответствующее окно и с помощью кнопки справа задать номер команды, после которой «пузырь» возникнет. На рис. 4 приведен пример экранной формы для случая, когда «пузырь» находится после 4-й команды.

Изучение работы простейшего конвейера приводит к выводу: при эффективной работе время выполнения одной команды в нем стремится к одному такту (длительности одной микрооперации). Дальнейшее повышение производительности возможно только за счет параллельного выполнения нескольких команд, то есть использования нескольких параллельно работающих конвейеров (мультиконвейерная система). Учебная имитационная модель мультиконвейерной системы будет описана в следующей статье.

*Орлов Сергей Павлович,
доктор технических наук, профессор,
заведующий кафедрой
Вычислительная техника СамГТУ,*

*Ефимушкина Наталья Владимировна,
кандидат технических наук, доцент,
доцент кафедры Вычислительная
техника СамГТУ.*



Наши авторы, 2012.
Our authors, 2012.

