

Программируемый таймер KP580BI53

(Отечественный аналог Intel i8253)

Управление периферийными устройствами часто требует от микропроцессорной системы точного задания временных интервалов между управляющими сигналами. Сделать это программно зачастую невозможно из-за сложного временного расписания обмена данными внутри микропроцессорной системы. К тому же вставка тактов ожидания в программу тормозит работу процессора. Такие задачи поручают программируемым таймерам (ПТ). С помощью ПТ процессор может формировать временные интервалы произвольной длительности, производить синхронизацию внешних устройств, организовывать счетчики событий, вести счет текущего времени и т. д.

Программируемый таймер KP580BI53 содержит три независимых счетчика с максимальной частотой счета 2 МГц. Каждый из трех счетчиков может быть запрограммирован на различные режимы работы и счета (двоичный или двоично-десятичный).

Таймер KP580BI53 может работать как:

- программируемый тактовый генератор;
- счетчик событий;
- бинарный перемножитель;
- цифровой одновибратор;
- часы реального времени.

Описание структуры и функционирование

Структурная схема программируемого таймера представлена на рис. 1. Блок логики чтения-записи управляет обменом данными между тремя счетчиками и шиной данных. Регистр управления позволяет программно настроить любой из трех счетчиков на желаемый режим работы.

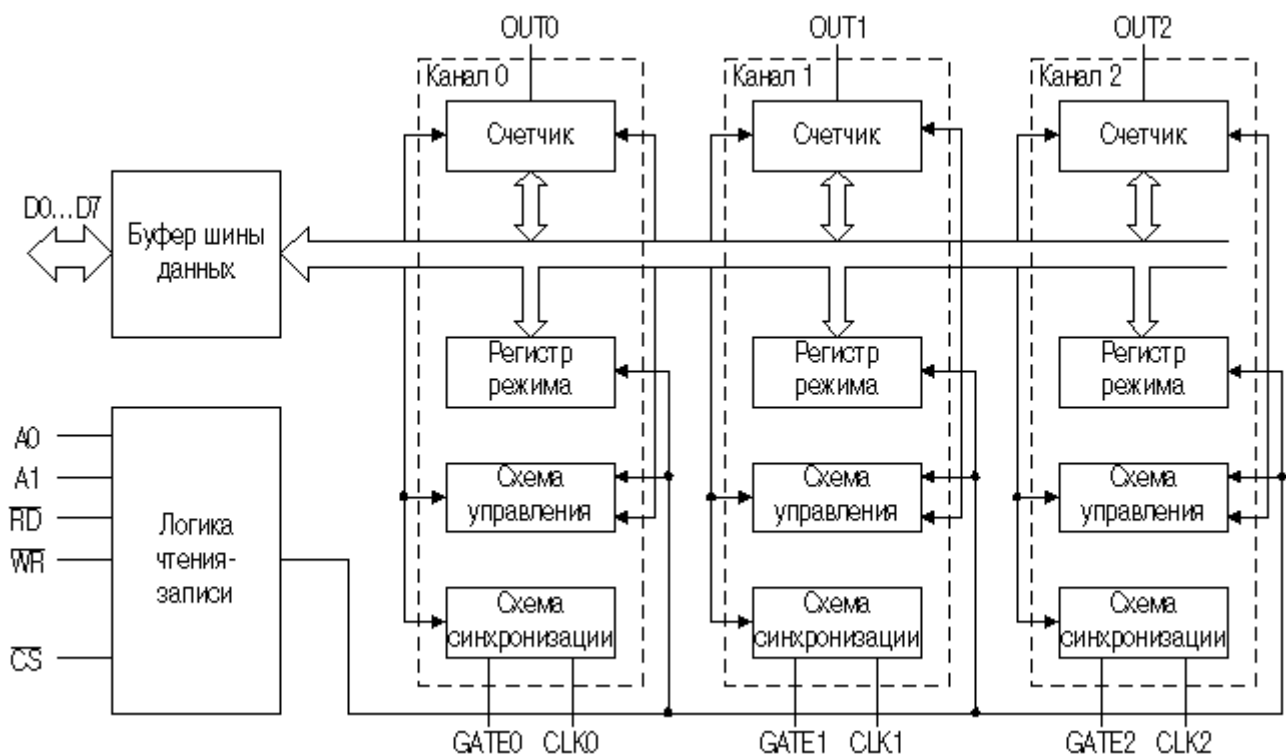


Рис. 1. Структура программируемого таймера KP580BI53

Доступ к счетчикам и регистру управления осуществляется в соответствии с табл. 1.

Таблица 1. Доступ к каналам ввода/вывода и регистру управляющего слова

A1	A0	/RD	/WR	/CS	
0	0	0	1	0	Записать счетчик 0
0	1	0	1	0	Записать счетчик 1
1	0	0	1	0	Записать счетчик 2
1	1	0	1	0	Загрузить режим работы
0	0	1	0	0	Читать счетчик 0
0	1	1	0	0	Читать счетчик 1
1	0	1	0	0	Читать счетчик 2
1	1	1	0	0	Нет операции, шина данных в третьем состоянии
X	X	X	X	1	Микросхема не выбрана, шина данных в третьем состоянии

Назначение этих и других выводов микросхемы приведено в табл. 2.

Таблица 2. Назначение выводов

D0...D7	Двунаправленная шина данных
/WR	Write. Запись. По низкому уровню на этом входе микропроцессор записывает данные в ПТ
/RD	Read. Чтение. Низкий уровень на этом входе информирует ПТ, что процессор хочет прочитать состояние счетчика
/CS	Chip Select. Выбор микросхемы. Низкий уровень инициирует обмен между процессором и ПТ. Не оказывает воздействия на работу счетчиков таймера
A0, A1	Адресные входы. Позволяют выбрать один из трех счетчиков для операции чтения/записи состояния счетчиков
CLK0...CLK2	Тактовые входы каждого из трех счетчиков
GATE0...GATE2	Входы разрешения счетчиков. Уровень "1" - разрешение счета
OUT0...OUT2	Выходы счетчиков

В состав ПТ входит буфер канала данных, логика чтения/записи, определяющая, к какому каналу обращается процессор, и трех независимых каналов. Каждый канал включает в себя 16-разрядный счетчик, регистр режима, схему управления и схему синхронизации.

В состав счетчика входят регистр хранения, буферный регистр и собственно счетчик. Регистр хранения содержит значение константы счета. В начале цикла работы канала константа счета из регистра хранения переписывается в счетчик, и затем по тактовым импульсам на входе CLK происходит декремент содержимого счетчика. Содержимое счетчика в любой момент времени может быть переписано в буферный регистр и прочитано процессором. В регистр режима записывается управляющее слово, определяющее режим работы канала. Схема управления синхронизирует работу отдельных схем в соответствии с запрограммированным режимом работы и работу канала с работой процессора. Схема синхронизации формирует серию внутренних импульсов с длительностью, определяемой внутренними времязадающими цепями, и периодом, равным периоду внешних тактовых импульсов.

Программирование и режимы работы

Каналы таймера полностью независимы друг от друга - каждый может иметь свой режим работы. Счетчик в каждом канале представляет собой 16-разрядный счетчик с предустановкой, работающий на вычитание в двоичном или двоично-десятичном коде. Таким образом, максимальное число при счете - 216 (при работе в двоичном коде) или 104 (при работе в двоично-десятичном коде) - получается при загрузке всех нулей в счетчик канала.

Режимы работы ПТ задаются при начальной установке. Процессор должен задать каждому каналу в указанной последовательности:

Режим работы. Определяется записью в ПТ управляющего слова по адресу A0=1, A1=1.

Число для счетчика (один или два байта в зависимости от управляющего слова для данного канала).

Формат управляющего слова приведен в табл. 3. Порядок программирования ПТ произвольный, т. е. можно сначала запрограммировать режимы работы всех каналов, а затем загрузить счетчики. Счетчик канала должен быть обязательно загружен именно тем количеством байтов, которое было запрограммировано в управляющем слове (значения разрядов D5 и D4). При загрузке содержимого счетчика двухбайтовым числом первым записывается младший байт, затем старший.

Таблица 3. Формат управляющего слова для выбора режима работы

D7	D6	D5	D4	D3	D2	D1	D0
Выбор канала	Число байтов для загрузки счетчика			Режим работы канала		1 = двоично-десятичный счет 0 = двоичный счет	
00 - канал 0	00 - чтение на лету			000 - режим 0			
01 - канал 1	01 - чтение/запись младшего байта			001 - режим 1			
10 - канал 2	10 - чтение/запись старшего байта			010 - режим 2			
	11 - чтение/запись слова			011 - режим 3			
				100 - режим 4			
				101 - режим 5			

Существует шесть режимов работы каждого канала.

Режим 0 - выдача сигнала прерывания по конечному числу. При работе в этом режиме на выходе канала появляется уровень "0" сразу же после установления режима работы. После загрузки числа в счетчик канала выход остается в "0" и счетчик начинает считать, если на входе разрешения установлен уровень "1". После того как достигается конечное число, на выходе устанавливается уровень "1" и остается до тех пор, пока канал не будет перезагружен режимом работы или новым числом.

Режим 1 - ждущий мультивибратор с программно-устанавливаемой длительностью сигнала. В этом режиме выход канала после загрузки числа в счетчик канала устанавливается в уровень "0" после первого тактового сигнала, следующего за передним фронтом на управляющем входе. Одновременно начинается счет, а при достижении конечного числа на выходе устанавливается уровень "1".

Режим 2 - генератор тактовых сигналов. В этом режиме на выходе канала через число периодов тактовой частоты, записанное в счетчике канала, появляется уровень "0" длительностью в один период тактовой частоты.

Режим 3 - генератор прямоугольных сигналов. В этом режиме на выходе канала будет уровень "1" в течение первой половины интервала времени, определяемого числом в счетчике, и уровень "0" в течение второй половины.

Режим 4 - программно-управляемый строб. После установки режима 4 на выходе канала появляется уровень "1". Когда число полностью загружено в счетчик канала и на управляющий вход подан уровень "1", начинается счет, и при достижении конечного числа на выходе появляется импульс уровня "0" длительностью в один период тактовой частоты.

Режим 5 - схемотехнически управляемый строб. Работа канала в этом случае аналогична работе в режиме 4 с той лишь разницей, что счетчик канала после загрузки начинает счет только по переднему фронту на управляющем входе. Кроме того, если во время счета на управляющем входе снова появится передний фронт сигнала, то счет будет начат сначала.

Существуют задачи, в которых процессор должен периодически считывать содержимое счетчиков каналов. Примером такой задачи может служить счет событий. Во время работы счетчика его текущее содержимое может быть передано в буферный и прочитано двумя способами: при помощи обычной операции чтения или с помощью специальной схемы "чтения на лету", активизируемой вводом специального управляющего слова.

При первом способе для обеспечения стабилизации показаний счета работа счетчика должна быть приостановлена подачей на управляющий вход уровня "0" (в режимах 0, 2, 3, 4) или с помощью внешней логической схемы, приостанавливающей подачу импульсов на тактовый вход канала.

Второй способ позволяет процессору считывать содержимое счетчика, не прерывая процесса счета. Для этого в ПТ по адресу $A0=1$, $A1=1$ осуществляется запись управляющего слова, определяющего в соответствии с табл. 3 режим чтения на лету ($D4=0$, $D5=0$). Разряды $D6$ и $D7$ определяют номер канала, состояние разрядов $D0..D3$ произвольно. По этой команде в буферном регистре защелкивается текущая величина счета, далее следует обычная операция чтения содержимого счетчика. Отметим, что данная команда не изменяет содержимого регистра режима. При этом способе чтения нельзя предварительно по каждому каналу производить запись управляющих слов и лишь затем производить чтение счетчиков каналов.

Особенность построения внутренней схемы таймера требует, чтобы операция чтения содержимого счетчика была выполнена до конца, т. е. если запрограммирована загрузка двух байтов, то нельзя, прочитав один младший байт, перегружать счетчик новой величиной.

Пример программирования

Часто программируемый таймер используется для создания тактовой частоты для контроллера последовательного порта передачи данных. Такой тандем оказывается очень гибким устройством с точки зрения его настройки на скорость передачи и режимы работы.

Рассмотрим настройку такого устройства на частоту передачи данных 4800 бит/с. Дополнительные условия: на вход $CLK0$ таймера подается тактовая частота 2 МГц, последовательный контроллер работает с внутренним делителем на 16. Выбрав в этом случае константу счета счетчика 0, равной 26, мы получим скорость последовательного порта $2000000/26/16=4808$ бит/с, что соответствует погрешности установки частоты -0,17%. Это не превышает предельно допустимой погрешности в 3%.

В табл. 4 приведена процедура программирования счетчика 0 программируемого таймера на константу счета 26.

Таблица 4. Пример программирования таймера KP580BI53

A1,A0	Операция	Регистр	Данные	Комментарий
11	Зп	Регистр команд	00111100 (36H)	Счетчик 0, чтение/запись слова, режим 3
00	Зп	Счетчик 0	00011010 (1AH)	Младший байт константы счета (26 десятичное)
00	Зп	Счетчик 0	00000000 (00H)	Старший байт константы счета

Заметим, что можно было бы здесь обойтись программированием на запись только младшего байта (старший все равно равен нулю), но в общем случае следует предусматривать изменение константы счета по ходу работы системы в больших пределах и заранее программировать счетчик на ввод/вывод слова.

Поскольку счетчики в таймере полностью независимы, остальные два счетчика можно использовать для других целей. Например, для часов реального времени или генерации звуковых сигналов заданной частоты и длительности.