СИСТЕМНЫЙ ТАЙМЕР

Это устройство подключено к линии запроса на прерывание IRQ0 и вырабатывает прерывание INT 8h приблизительно 18,2 раза в секунду (точное значение - 1193180/65536 раз в секунду).
При инициализации BIOS устанавливает свой обработчик для прерывания таймера. Этот обработчик каждый раз увеличивает на 1 текущее значение четырехбайтовой переменной, располагающейся в области данных BIOS по адресу 0000:046Ch - счетчик тиков таймера. Если этот счетчик переполняется (прошло более 24 часов с момента запуска таймера), в ячейку 0000:0470h заносится 1.

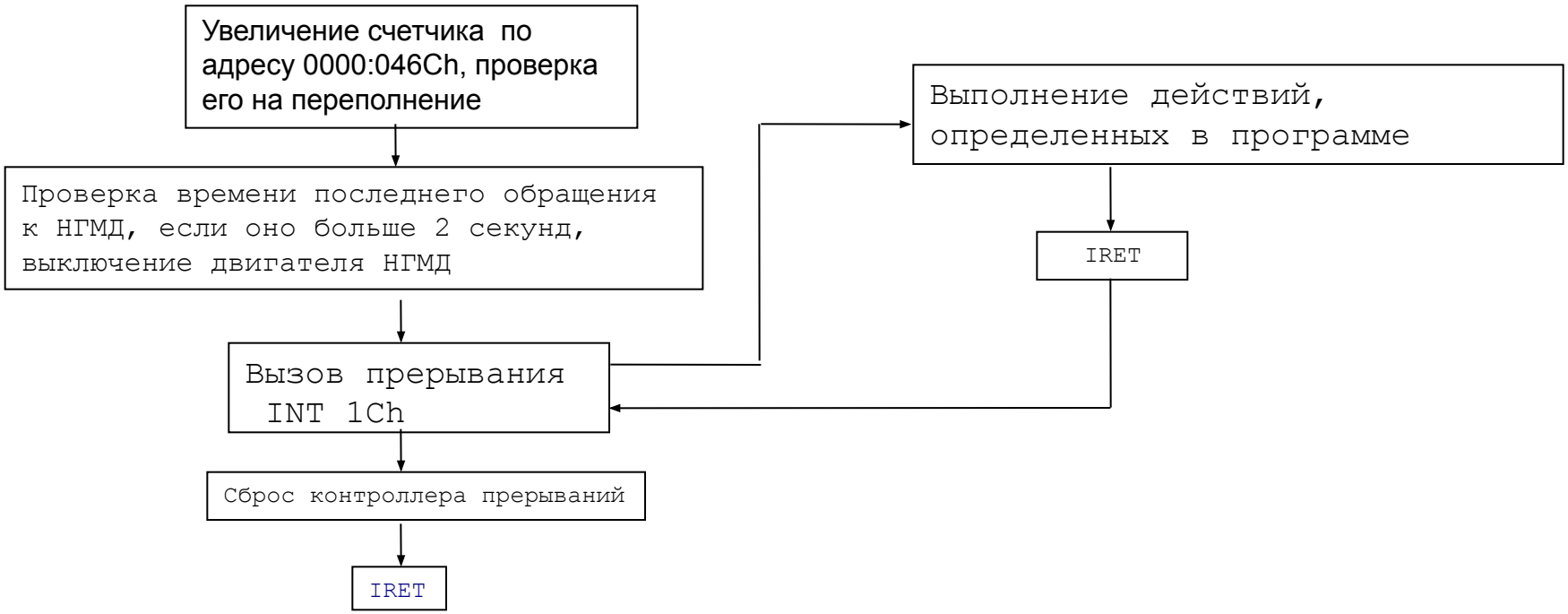
Другое действие, выполняемое стандартным обработчиком прерывания таймера - контроль за работой двигателей НГМД. Если после последнего обращения к НГМД прошло более 2 секунд, обработчик прерывания выключает двигатель. Ячейка с адресом 0000:0440h содержит время, оставшееся до выключения двигателя. Это время постоянно уменьшается обработчиком прерывания таймера. Когда оно становится равно 0, обработчик выключает двигатель НГМД.

Третье действие, которое выполняет обработчик прерывания таймера - вызов прерывания INT 1Ch. После инициализации системы вектор INT 1Ch указывает на команду IRET, т.е. ничего не выполняется. Программа может установить собственный обработчик этого прерывания для того чтобы выполнять какие-либо периодические действия.

Необходимо отметить, что прерывание INT 1Ch вызывается обработчиком прерывания INT 8h до сброса контроллера прерывания, поэтому во время выполнения прерывания INT 1Ch все аппаратные прерывания запрещены. В частности, запрещены прерывания от клавиатуры.

Обработчик прерывания INT 1Ch должен заканчиваться командой IRET. Если же вы подготавливаете собственный обработчик для прерывания INT 8h, перед завершением его работы необходимо сбросить контроллер прерываний. Это можно сделать, например, так:

```
mov     al, 20h
out     20h, al
```



механизм обработки прерывания таймера

Таймер обычно реализуется на микросхеме Intel 8253 (для компьютеров IBM PC и IBM XT) или 8254 (для компьютеров IBM AT и IBM PS/2). Отечественные аналоги: К1810ВИ53 и К1810ВИ54

Таймеры 8253 и 8254 состоят из трех независимых каналов, или счетчиков. Каждый канал содержит регистры:
состояния канала RS (8 разрядов);
управляющего слова RSW (8 разрядов);
буферный регистр OL (16 разрядов);
регистр счетчика CE (16 разрядов);
регистр констант пересчета CR (16 разрядов)

Каналы таймера подключаются к внешним устройствам при помощи трех линий:
GATE - управляющий вход;
CLOCK - вход тактовой частоты;
OUT - выход таймера.

Регистр счетчика CE работает в режиме вычитания. Его содержимое уменьшается по заднему фронту сигнала CLOCK при условии, что на вход GATE установлен уровень логической 1.

В зависимости от режима работы таймера при достижении счетчиком CE нуля тем или иным образом изменяется выходной сигнал OUT.

Буферный регистр OL предназначен для запоминания текущего содержимого регистра счетчика CE без остановки процесса счета. После запоминания буферный регистр доступен программе для чтения.

Регистр констант пересчета CR может загружаться в регистр счетчика, если это требуется в текущем режиме работы таймера.

Возможны шесть режимов работы таймера.
Они разделяются на три типа:
Режимы 0, 4 - однократное выполнение функций.
Режимы 1, 5 - работа с перезапуском.
Режимы 2, 3 - работа с автозагрузкой.

В режиме однократного выполнения функций перед началом счета содержимое регистра констант пересчета CR переписывается в регистр счетчика CE по сигналу CLOCK, если сигнал GATE установлен в 1. В дальнейшем содержимое регистра CE уменьшается по мере прихода импульсов CLOCK. Процесс счета можно приостановить, если подать на вход GATE уровень логического 0. Если затем на вход GATE подать 1, счет будет продолжен дальше. Для повторения выполнения функции необходима новая загрузка регистра CR, т.е. повторное программирование таймера.

При работе с перезапуском не требуется повторного программирования таймера для выполнения той же функции. По фронту сигнала GATE значение константы из регистра CR вновь переписывается в регистр CE, даже если текущая операция не была завершена.
В режиме автозагрузки регистр CR автоматически переписывается в регистр CE после завершения счета. Сигнал на выходе OUT появляется только при наличии на входе GATE уровня логической 1. Этот режим используется для создания программируемых импульсных генераторов и генераторов прямоугольных импульсов (меандра).

Канал 0 используется в системных часах времени суток (не следует путать с часами реального времени, реализованными на другой микросхеме). Этот канал работает в режиме 3 и используется как генератор импульсов с частотой примерно 18.2 Гц. Именно эти импульсы вызывают аппаратное прерывание INT 8h. Канал 1 используется для регенерации содержимого динамической памяти компьютера. Выход канала OUT используется для запроса к каналу прямого доступа DMA, который и выполняет обновление содержимого памяти.

Канал 2 подключен к громкоговорителю компьютера и может быть использован для генерации различных звуков или музыки, либо как генератор случайных чисел. Канал использует режим 3 таймера 8253/8254.

Программирование таймера на уровне портов

Таймеру соответствуют четыре порта ввода/вывода со следующими адресами:

40h - канал 0;

41h - канал 1;

42h - канал 2;

43h - управляющий регистр.

формат управляющего регистра

№ разрядов	Содеожание
0 123 45 67	BCD: 0 - двоичный счет; 1 - двоично-десятичный счет. M: 000 - режим 0; 001 - режим 1; X10 - режим 2; X11 - режим 3; 100 - режим 4; 101 - режим 5. RW: 00 - код команды CLC (запомнить CE); 01 - чтение/запись старшего байта; 10 - чтение/запись младшего байта; 11 - чтение/запись младшего, затем старшего байта. SC: 00 - канал 0; 01 - канал 1; 10 - канал 2; 11 - код команды RBC (чтение состояния канала).

Поле BCD определяет формат константы, использующейся для счета - двоичный или двоично-десятичный. В двоично-десятичном режиме константа задается в диапазоне 1-9999.
Поле RW определяет способ загрузки констант через однобайтовый порт. Если в этом поле задано значение 00, это управляющее слово будет использоваться для фиксации текущего содержимого регистров счетчика CE в буферном регистре OL с целью чтения программой. Это код команды CLC - фиксация регистров. Код канала, для которого будет выполняться фиксация, должен быть указан в поле SC. Поля M и BCD при этом не используются.
Поле SC определяет номер канала, для которого предназначено управляющее слово. Если в этом поле задано значение 11, будет выполняться чтение состояния канала.

формат команды RBC чтения слова состояния канала

равно 0.
1 - выбор канала 0.
1 - выбор канала 1.
1 - выбор канала 2.

STAT: 0 - читать состояние каналов;
1 - не читать состояние каналов.
CNT:
0 - запомнить текущее содержимое CE;
1 - не запоминать содержимое CE.

код команды RBC - 11.

Формат слова состояния канала напоминает формат регистра управляющего слова, за исключением двух старших разрядов 7 и 6

6р FN: флаг перезагрузки констант;

7р OUT: состояние выхода OUT.

Разряд FN используется, в основном, в режимах 1 и 5 для определения, произошла ли загрузка константы из регистра CR в регистр счетчика CE. Разряд OUT позволяет определить состояние выходной линии канала OUT в момент выполнения команды RBC.

Для программирования канала таймера необходимо выполнить следующую последовательность действий:
вывести в порт управляющего регистра с адресом 43h управляющее слово;
требуемое значение счетчика посылается в порт канала (адреса 40h...42h), причем вначале выводится младший, а затем старший байты значения счетчика.
Сразу после этого канал таймера начнет выполнять требуемую функцию.

Для чтения текущего содержимого счетчика CE необходимо выполнить следующее:
вывести в порт управляющего регистра код команды CLC (команда запоминания содержимого регистра CE);
вывести в порт управляющего регистра код команды запроса на чтение/запись в регистры канала (поле RW должно содержать 11);
двумя последовательными командами ввода из порта нужного канала вывести младший и старший байты текущего состояния счетчика CE.

Для чего вам может понадобиться перепрограммирование каналов таймера? Если вам надо повысить точность измерения времени, выполняемого с помощью канала 0 таймера, вы можете увеличить частоту генерируемых этим каналом импульсов (стандартно 18,2 Гц). По окончании измерений режим работы канала необходимо восстановить для правильного функционирования системы. Канал 2, подключенный к громкоговорителю, вы можете использовать для генерации различных звуков или музыки, о чем мы расскажем немного позже. Этот же канал может быть использован для генерации случайных чисел.

Средства BIOS для работы с таймером

Для работы с таймером (точнее говоря, для работы с каналом 0 таймера) BIOS содержит две функции прерывания INT 1Ah. Они позволяют прочитать текущее содержимое счетчика и изменить его.

Функция 00h предназначена для чтения содержимого счетчика таймера:

На входе: AH = 00h.

На выходе: CX = старший байт счетчика;
DX = младший байт счетчика;
AL = 0, если с момента перезапуска таймера прошло более 24-х часов.

Функция 01h позволяет изменить содержимое счетчика таймера:

На входе: AH = 01h;

CX = старший байт счетчика;
DX = младший байт счетчика.

На выходе: не используются.

Функцию чтения таймера можно использовать для организации программной задержки. Так как работа таймера не зависит от производительности процессора, быстродействие системы не будет влиять на формируемую задержку.
Однако следует учитывать, что точность формирования задержки определяется частотой обновления счетчика таймера (18.2 Гц), и может оказаться недостаточной для некоторых приложений.

BIOS компьютеров IBM AT содержит еще две интересные функции для работы с таймером. Это функции 83h и 86h прерывания INT 15h.
Функция 83h позволяет запустить таймер на счет, указав адрес некоторого байта в оперативной памяти. Программа, запустившая таймер, сразу после запуска получает управление. По истечении времени, заданного при запуске таймера, функция устанавливает старший бит указанного байта в единицу, сигнализируя таким образом программе о завершении указанного временного интервала. Программа может также отменить работу таймера в этом режиме.
Эту функцию удобно использовать для организации выполнения каких-либо действий параллельно с отсчетом времени, например, можно ограничить время для ввода пароля.

формат вызова функции 83h прерывания INT 15h:

На входе: AH = 83h;
AL = код подфункции:
0 - установить интервал, запустить таймер;
1 - отменить работу таймера;

CX = старший байт времени работы счетчика, задается в микросекундах;
DX = младший байт счетчика;

ES:BX = адрес байта, в котором по истечении интервала времени старший бит будет установлен в 1.

На выходе: не используются.

Функция 86h специально предназначена для формирования задержек. Она позволяет определять время задержки в микросекундах, что достаточно удобно для многих задач. Во время выполнения задержки разрешены прерывания.

Формат вызова функции:
На входе: AH = 86h;
CX = старший байт времени задержки, задается в микросекундах;
DX = младший байт времени задержки.

На выходе: не используются.

Одно из наиболее распространенных применений таймера - генерация звуковых сигналов и воспроизведение музыки. Таймер позволяет воспроизводить музыку в фоновом режиме, т.е. во время работы программы может звучать музыка.

Как мы уже говорили, канал 2 микросхемы 8254 связан с громкоговорителем компьютера. Однако громкоговоритель не просто соединен с выходом OUT канала 2. Порт вывода 61h также используется для управления громкоговорителем. Младший бит порта 61h подключен ко входу GATE канала 2 таймера. Этот бит при установке в 1 разрешает работу канала, т.е. генерацию импульсов для громкоговорителя. Дополнительно для управления громкоговорителем используется бит 1 порта 61h. Если этот бит установлен в 1, импульсы от канала 2 таймера смогут проходить на громкоговоритель.

для включения звука надо выполнить следующие действия:
запрограммировать канал 2 таймера на нужную частоту (т.е. загрузить регистр счетчика канала нужным значением);
для включения звука установить в 1 два младших бита порта 61h.
Так как остальные 6 битов порта 61h используются для других целей, установка младших битов должна выполняться таким образом, чтобы значения остальных битов не были изменены. Для этого вначале надо считать байт из порта 61h в рабочую ячейку памяти, установить там нужные биты, затем вывести новое значение байта в порт 61h.
Очевидно, что для выключения звука надо сбросить два младших бита порта 61h в 0 (при этом нельзя изменять значение остальных битов этого порта).
Мелодия (одноголосая), как известно, состоит из нот, разделенных или не разделенных паузами. При проигрывании мелодии необходимо для каждой ноты программировать соответствующим образом канал 2 таймера и включать громкоговоритель (с помощью порта 61h) на определенное время, равное длительности ноты. Затем программа должна выключить динамик и выдержать паузу перед проигрыванием следующей ноты, если такая пауза требуется.

Программа может генерировать звуки и другим способом, не используя таймер. Для этого нужно сбросить младший бит порта 61h и, управляя битом 1 этого порта, формировать импульсы для громкоговорителя. Т.е. программа должна устанавливать этот бит то в 0, то в 1 с некоторым периодом. Высота генерируемого звука будет соответствовать этому периоду.

Можно также комбинировать эти два способа, получая разнообразные звуковые эффекты.

Для определения значения, которое должно быть записано в регистр счетчика канала 2 таймера, надо разделить 1193180 на требуемую частоту в герцах.

Генерация случайных чисел

Для генерации случайных чисел лучше всего использовать канал 2 в режиме 3. В регистр счетчика канала мы занесем значение, равное диапазону нужных нам случайных чисел. Например, если мы запишем в регистр число 80 и запустим канал таймера, получаемые случайные числа будут лежать в диапазоне от 0 до 79.

```
void rnd_set(int bound) {
// Устанавливаем режим 3 для второго канала таймера
outp(0x43, 0xb6);
// Загружаем регистр счетчика таймера - сначала/
/ младший, затем старший байты
  outp(0x42, bound & 0x00ff);
   outp(0x42, (bound &0xff00) >> 8);
// Разрешаем работу канала
outp(0x61, inp(0x61) | 1);}
```