

Programmable Interval Timer

Me

September 13, 2016

Contents

1	Programmable Interval Timer	2
1.1	Каналы PIT	2
1.2	Режимы PIT	2
1.3	Частота работы PIT	3
1.4	Взаимодействие с PIT	3

1 Programmable Interval Timer

Измерение времени и генерация периодических прерываний довольно важная часть ОС. Например, с их помощью мы отмеряем время, которое поток исполнения может занимать процессор, а как только отмеренное время истечет мы "сместим" поток исполнения и заменим его другим, таким образом можно организовать вытесняющую многозадачности.

Programmable Interval Timer (PIT, так же известен как intel 8254) - устройство, которое умеет генерировать сигналы с заданной частотой, или отмерять заданные интервалы времени и генерировать сигнал по истечению заданного интервала.

1.1 Каналы PIT

На самом деле в PIT совмещены сразу 3 таймера, которые называют каналами. Но для свободного использования нам доступен только 0 канал. Поэтому отсюда и далее речь пойдет только про один таймер.

1.2 Режимы PIT

Более или менее все таймеры, не только PIT, обладают двумя основными режимами работы:

- periodical - таймер генерирует сигналы периодически с заданным интервалом;
- one shot - таймер отмеряет заданный интервал времени и генерирует один сигнал;

Конкретные таймеры могут добавлять различные вариации, называть их по разному и прочее. Например, для PIT мне известно 6 режимов работы, но все они разделяются на one shot-ы и periodical.

В качестве примера one shot-а рассмотрим режим 0, который так же известен как Interrupt On Terminal Count. В этом режиме вы задаете начальное значение счетчика, задаете как часто счетчик уменьшается и по достижению 0 PIT генерирует сигнал.

Как пример periodical рассмотрим режим 2, который так же известен как Rate Generator, в этом режиме так же нужно задать, как часто уменьшается счетчик и начальное значение этого счетчика. Но по достижению 0 не только генерируется сигнал, но еще и значение счетчика сбрасывается в начальное значение и отсчет начинается заново.

1.3 Частота работы PIT

Собственная частота PIT 1193182 Гц. Это довольно много, что делать если мы хотим генерировать сигналы чуть реже? Выше уже упоминалось, что и one shot и periodical режимы используют некоторое начальное значение. Чем больше начальное значение, тем дольше займет обратный отсчет до 0, и тем реже будут генерироваться сигналы. Начальное значение состоит из 16 бит или 2-х байт.

Суммируя все вместе, если вы хотите, чтобы таймер генерировал сигналы с заданной частотой, то начальное значение можно посчитать по следующей формуле:

$$Init = \frac{1193182}{Freq_{out}}$$

1.4 Взаимодействие с PIT

Взаимодействие с PIT осуществляется с помощью портов ввода/вывода. Есть общий командный порт 0x43, а кроме того для каждого канала есть свой порт данных, например, для канала 0 это порт 0x40.

Формат команды, которая записывается в порт команд 0x43 довольно замысловатый:

- бит 0 (BCD) - определяет используемый формат чисел, вам точно не нужен BCD, так что этот бит должен быть равен 0;
- биты с 1 по 3 задают режим работы;
- биты 4 и 5 задают какие байты инициализирующего значения вы хотите записать в порт данных и хотите ли вообще;
- биты 6 и 7 отвечают за выбор канала, в нашем случае канал всегда 0.

Пару слов о задании начального значения. Как уже сказано биты 4 и 5 порта команд указывают на то, хотим ли мы задать младший или старший байт начального значения, или, например, что мы хотим задать оба или ни один из них. Кодированы эти варианты довольно очевидным образом:

- установите 4-ый бит, если хотите задать младший байт;
- установите 5-ый бит, если хотите задать старший байт.

Если вы установили оба бита, то первая запись в порт данных задает младший байт, а вторая задает старший байт. После задания начального значения таймер начинает работу и будет генерировать прерывания. PIT зачастую подключен к 0 входу Master контроллера прерываний, соответственно, там и нужно ждать сигналов.