

## Програмираме таймери и броячи

### Програмираме таймери и броячи

Най-често програмираме таймери и броячи за следните цели:

- генериране на сигнали с програмируем период за нуждите на външни устройства – например аналого-цифрови преобразуватели;
- задаване по програмен път на скоростта на работа на серийния интерфейс;

3

## Програмираме таймери и броячи

За задаване на различни времеинтервали, броене на определени събития или за генериране на различни сигнали в микропроцесорните системи се използват програмирами таймери и броячи. Те представляват интегрални микросхеми, които са пригодени да работят със сигналите на системната магистрала и функциите им могат да се задават и променят по програмен път.

2

### Програмираме таймери и броячи

- измерване на времето между появата на външни събития;
- броене на външни събития;
- периодично генериране на заявки за прекъсване на микропроцесорната система например за разпределение на времето в режим на многозадачна работа.

4

### Програмираме таймери и броячи

За изпълнение на тези функции в състава на микропроцесорната фамилия 8086 са включени интегралните микросхеми 8253 и 8254. От гледна точка на функции и на програмиране те са почти идентични.

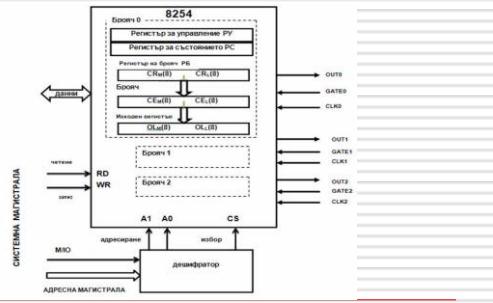
5

## Програмираме таймер 8254

Блоковата схема на програмирамия таймер 8254 е представена на фиг. 7.7. Той се състои от три еднакви устройства – СТ0, СТ1 и СТ2. Всяко от тези устройства съдържа регистър за управление РУ, регистър на състоянието РС, регистър на брояча РБ, брояч и изходен регистър. Всяко устройство има два входа CLK и GATE, и един изход OUT.

6

## Програмираме таймер 8254



7

## Програмираме таймер 8254

Взаимодействието на таймера със системната магистрала се осъществява чрез двупосочните изводи за данни **D7-D0**, входовете CS, RD, WR за сигналите за управление и входо-вете **A1**, **A0** за адресиране. За да се осъществи обмен с микропроцесора, сигнал **CS** трябва да е равен на лог. 0, посоката на обмен (вход или изход) се определя от сигналиите RD и WR, а състоянието на входовете **A1**, **A0** определя частта от таймера, с която ще се извърши обмена на данни.

8

## Програмираме таймер 8254

Адресирането на схемата е онагледено в табл. 7.6.

При всички останали комбинации на сигналите на входовете, изводите за данни D7-D0 преминават във висок импеданс.

9

10

## Програмираме таймер 8254

### Режим 0

Този режим е предназначен основно за броене на събития. Записът на ново число **N** започва със запис на младшия байт. След запис на старшия байт започва намаляване на съдържанието на брояча с единица при всеки преход от 1 в 0 на тактовия сигнал **CLK**. При достигане на нула (след **N** импулс на вход **CLK**) изход **OUT** преминава в лог.1 и процесите спират.

11

## Програмираме таймер 8254

За да се стартира брояча отново трябва да се извърши нов запис. Сигнал GATE = 1 разрешава броенето, а при GATE = 0 броенето спира. Броенето спира и при запис на младшия байт от ново число.

12

## Програмираме таймер 8254

### □ Режим 1

Този режим е предназначен за генериране на единичен отрицателен импулс на изход OUT с продължителност N периода на входния сигнал CLK. Броят N се записва в регистъра на брояча RB. Отрицателният преход 1 → 0 на вход CLK при GATE = 1 предизвиква преход 1 → 0 на изход OUT и стартиране на броенето (изваждането).

13

## Програмираме таймер 8254

При достигане на RB = 0 (т.е. след N импулса на вход CLK), изход OUT преминава от лог. 0 в лог.1 и отброяването спира. За да се стартира нов цикъл трябва да се извърши запис в регистъра на брояча RB.

14

## Програмираме таймер 8254

### □ Режим 3

Този режим е предназначен за генериране на непрекъсната поредица от правоъгълни импулси с програмируема честота. Този режим е аналогичен на предходния режим 2, с тази разлика, че преходът от 1 в 0 на изхода OUT става при достигане на половината от зададеното число N за отброяване.

15

## Програмираме таймер 8254

Ако N е нечетно, времето, през което изход OUT е в състояние 1, ще е с един период на CLK по дълго от времето, през което ще е равен на лог.0. Сигнал GATE = 1 разрешава броенето, а GATE = 0 броенето спира.

16

## Програмираме таймер 8254

### □ Режим 4

Този режим е предназначен за генериране на единичен отрицателен импулс след отброяване на N импулса на вход CLK. Единичният импулс е с продължителност 1 такт на сигнала CLK. След това отброяването спира. За да се започне нов цикъл трябва да се извърши запис в регистъра на брояча RB.

17

## Програмираме таймер 8254

### □ Режим 5

Този режим е аналогичен на режим 4 с тази разлика, че след генерирането на единичния импулс, се извършва презапуск. Т.е. генерират се единични отрицателни импулси с продължителност един такт на сигнала на вход CLK и период на повторение N такта на сигнала на вход CLK.

18

## Програмираме таймер 8254

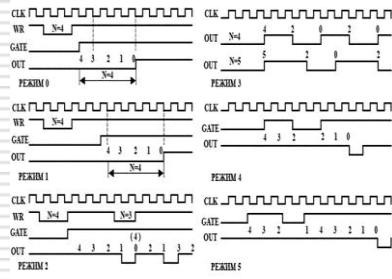
Времедиаграмите, поясняващи работата на таймера при описаните режими, са представени на фиг. 3.9.

### Инициализация на таймера

При стартиране на микропроцесорната система, както във всички останали програмируеми схеми, така и в програмируемия таймер трябва да се запишат начални стойности в регистрите за управление на таймерите.

19

### Времедиаграма на режимите на работа на таймер 8254



20

## Край на част 10

21