

## Obsługa zegarów CTC i RTC.

### 4.1. Wiadomości wstępne.

System odmierzania czasu, jest oparty na układach 8253 i 8254 (drugi układ różni się posiadaniem rozkazu *Read-Back*). System ten ma w praktyce istotne znaczenie:

- wiele funkcji systemowych, np. odświeżanie pamięci dynamicznej, wymaga okresowego wykonywania z dokładnie określoną częstotliwością;
- mechanizmy napędowe stacji dysków elastycznych wymagają podawania bardzo precyzyjnych impulsów;
- system operacyjny prowadzi ewidencję plików dyskowych i katalogów, w oparciu o czas systemowy (jest to jedyny sposób ustalenia, który z plików zawiera aktualne dane);
- czas wykonywania pewnych fragmentów programu musi być niezależny od obecnego w danym systemie procesora i szybkości jego taktowania. Na przykład czas wykonania klasycznej pętli *for* w bardzo dużym stopniu zależy od sprzętu, więc wiele programów wymaga wbudowanych opóźnień lub konsoli upływu czasu rzeczywistego.

**CTC** - zegar systemowy (licznik, stoper).

**RTC** - (*real time clock*) zegar czasu rzeczywistego.

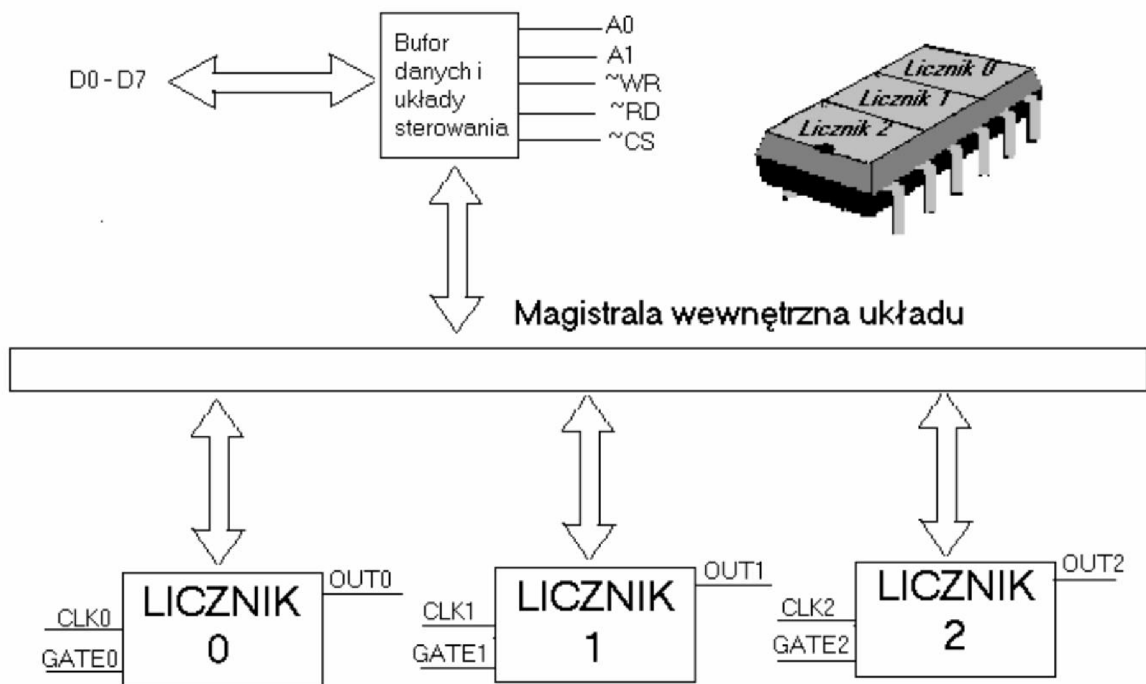
(„ *Anatomia PC*” , P Metzger A Jałowicki, „ *Encyklopedia informatyki*” S Kruk).

### 4.2. Ogólny opis układu odmierzania czasu.

#### 4.2.1. Układ czasowy (8253 i 8254).

Składa się z trzech w pełni niezależnych generatorów oznaczonych numerami 0, 1 i 2. Bloki te zawierają niezależne wejścia GATE i CLK oraz wyjście OUT. Są połączone wewnętrzną 8-bitową magistralą. (*Rys.4.1*).

Rys.4.1. Schemat blokowy układu czasowego 8253/8254.



Zestaw dodatkowych sygnałów sterujących umożliwia procesorowi dwukierunkowy dostęp do rejestrów sterujących każdego z nich. Centralnym elementem każdego z generatorów jest programowany licznik CE, którego ustawiona wstępnie w fazie programowania zawartość, może być zmniejszana na różne sposoby (zależnie od ustawionego trybu pracy). 16-bitowa liczba przeznaczona do zapisania w rejestrze CE danego generatora jest dostarczona (w porcjach po 8 bitów) magistralą wewnętrzną układu do dwóch stowarzyszonych z danym rejestrem CE rejestrów pośrednich  $CW_H$  i  $CW_L$ . Tam oczekuje na przepisanie do CE pierwszym impulsem na wejściu CLK, który nadejdzie po zaprogramowaniu danego generatora. W podobny sposób, tj. poprzez jednobajtowe rejestry pośrednie  $CR_H$  i  $CR_L$ , odbywa się transport zawartości rejestru CE w rozkazach sprawdzających jego stan. Dla przykładu zostanie opisany układ 8253:

#### 4.2.2. Tryby pracy.

Każdy z trzech generatorów układu 8253 może być zaprogramowany niezależnie do pracy w jednym z sześciu trybów (tylko dwa z nich znajdują zastosowanie w PC).

##### **Tryb 0:**

W stanie początkowym, wyjście OUT znajduje się w stanie 0. Po załadowaniu do wybranego licznika CE wartości N (oznaczającej liczbę zliczonych impulsów taktujących) następuje odliczenie impulsów podawanych na wyjście CLK do chwili wyzerowania licznika CE, co powoduje pojawienie się na wyjściu OUT stanu 1. Stan ten utrzymywany będzie aż do nadejścia następnego słowa sterującego, programującego generator do pracy w trybie 0. Odliczanie impulsów taktujących ma miejsce tylko wtedy, gdy na wejście GATE, podany jest wysoki poziom logiczny.

### **Tryb 1:**

W stanie początkowym, wyjście OUT jest w stanie 1. Pierwszy impuls podany na wejście CLK powoduje pojawienie się na wyjściu OUT stanu 0, który utrzymuje się do końca zliczania impulsów taktujących. Pierwszy impuls pojawiający się na wyjściu CLK po przejściu licznika CE przez stan 0000, powoduje ponownie wystawienie na wyjściu OUT 1. Układ wytwarza, więc pojedynczy impuls niskiego poziomu logicznego o czasie trwania  $N \cdot T_{CLK}$ , gdzie N jest pierwotną zawartością licznika CE, a  $T_{CLK}$  jest okresem przebiegu taktującego podawanego na wejście CLK. Stan sygnału GATE musi być utrzymany na poziomie logicznej jedynki przez cały cykl zliczania. Jeżeli sygnał GATE zmieni swój poziom na 0, jego powrót do poziomu wysokiego spowoduje ponowne załadowanie licznika CE wartością N.

Możliwe jest zapisanie nowej wartości N w czasie pracy generatora (cyklu zliczania), ale będzie ona uwzględniona dopiero od następnego impulsu GATE.

### **Tryb 2:**

Działanie tego trybu jest analogiczne do trybu 1, aż do momentu osiągnięcia przez licznik CE wartości 0001, tj. tuż przed końcem odliczania. Stan wyjścia OUT (znajdującego się dotychczas w stanie 1) zostaje wówczas na czas jednego okresu przebiegu taktującego sprowadzony do niskiego poziomu logicznego. Licznik CE ładowany jest powtórnie wartością N i cykl powtarza się. Zliczanie ma miejsce tylko dla  $GATE=1$  i jest wstrzymywane na czas, gdy  $GATE=0$ . W trybie tym generowany jest okresowo impuls szpilkowy o okresie  $N \cdot T_{CLK}$  i szerokości  $T_{CLK}$ .

### **Tryb 3:**

Działanie tego trybu jest analogiczne do trybu 1, aż do momentu osiągnięcia przez licznik CE wartości  $N/2$ . Następuje wówczas zmiana stanu wyjścia OUT z 1 na 0. Stan  $OUT=0$  utrzymuje się do momentu wyzerowania licznika CE, po czym zmienia się na 1. Opisany proces przebiega tylko wtedy, gdy na wejście GATE podany jest stan 1, w przeciwnym razie na wyjściu OUT utrzymuje się stały wysoki poziom logiczny. Działanie układu w tym trybie, sprowadza się do generacji przebiegu prostokątnego o okresie  $N \cdot T_{CLK}$ .

#### **Tryb 4:**

W stanie początkowym wyjście OUT jest w stanie wysokim. Pierwszy impuls podany na wyjście CLK wpisuje do licznika CE zadaną wartość  $N$ . W stanie  $GATE=1$  następne impulsy taktujące powodują zmniejszenie licznika do osiągnięcia wartości 0000. Poziom na wyjściu OUT opada wówczas do 0 na czas  $T_{CKL}$ . W trybie tym mówimy o programowym wyzwoleniu licznika, bowiem moment rozpoczęcia odliczania jest określony dokonaniem zapisu do rejestru sterującego.

#### **Tryb 5:**

Licznik jest wyzwolany sprzętowo poprzez zmianę poziomu na wejściu GATE z 0 na 1. Powoduje to zapisanie do licznika CE wartości  $N$  i rozpoczęcie odliczania. Sygnał na wyjściu OUT przyjmuje wartość 0 po  $N+1$  cyklach przebiegu taktującego, począwszy od impulsu wyzwalamającego GATE. Przedwczesne (przed zakończeniem odliczania), ponowne podanie impulsu GATE powoduje odnowienie wartości  $N$ , a więc generowany impuls nigdy nie będzie krótszy od założonej wartości.

Cechą wspólną wszystkich trybów jest odliczanie „w dół” od ustawionej wartości początkowej 16-bitowego licznika CE. Generator rozpoczyna pracę w danym trybie natychmiast po załadowaniu rejestru sterującego.

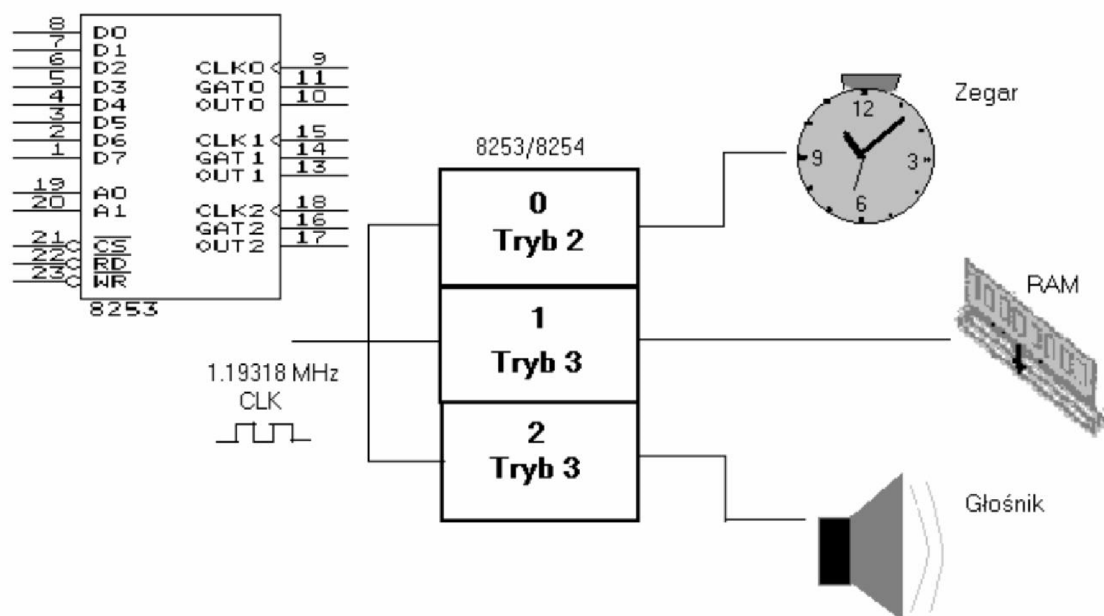
Wszystkie generatory układu scalonego 8253 zainstalowanego w komputerze PC/XT lub AT są taktowane tą samą częstotliwością zegarową równą 1,19318 MHz. Sygnał CLK jest wytwarzany w oddzielnym układzie generującym w sposób niezależny od procesora.

Przyporządkowanie generatorów układu 8253 (Rys.4.2.):

Generator nr 0 – ustawiony w tryb 2 – zegar CTC.

Generator nr 1 – ustawiony w tryb 3 – RAM.  
 Generator nr 2 – ustawiony w tryb 3 – Głośnik.

Rys.4.2. Przyporządkowanie generatorów układu 8253.



#### 4.2.3. Opis i znaczenie końcówek układu.

**GATE0 – GATE2** (*Gate*)-wejścia sygnałów aktywujących, bramkujących generatory 0-2.

**CLK0 – CLK2** (*Clock*)-wejścia sygnałów taktujących dla generatorów 0-2.

**OUT0 – OUT2** (*Output*)-wyjścia generatorów 0-2.

**D0 – D7** 8-bitowa, dwukierunkowa magistrala danych. Kierunek transmisji ustalony jest sygnałami  $\sim$ WR i  $\sim$ RD

$\sim$ CS (*Chip Select*)-niski poziom logiczny na tym wejściu wprowadza układ w stan programowania. Poziom logiczny tego wejścia nie ma wpływu na pracę generatorów.

$\sim$ WR (*Write*)-procesor wymusza niski poziom logiczny na tym wejściu, chcąc zapisywać dane do wewnętrznych rejestrów generatorów.

$\sim$ RD (*Read*)- procesor wymusza niski poziom logiczny na tym wejściu, chcąc odczytywać dane z wewnętrznych rejestrów generatorów.

**A0 – A1** wejścia adresujące w trybie programowania jeden z generatorów 0-2 lub rejestr stanu układu. Możliwe są następujące kombinacje linii adresowych:

- 00-licznik CE0,
- 10-licznik CE1,
- 01-licznik CE2,
- 11-rejestr stanu,

**Vcc** wejście napięcia zasilającego (+5V).

**GND** masa zasilania.

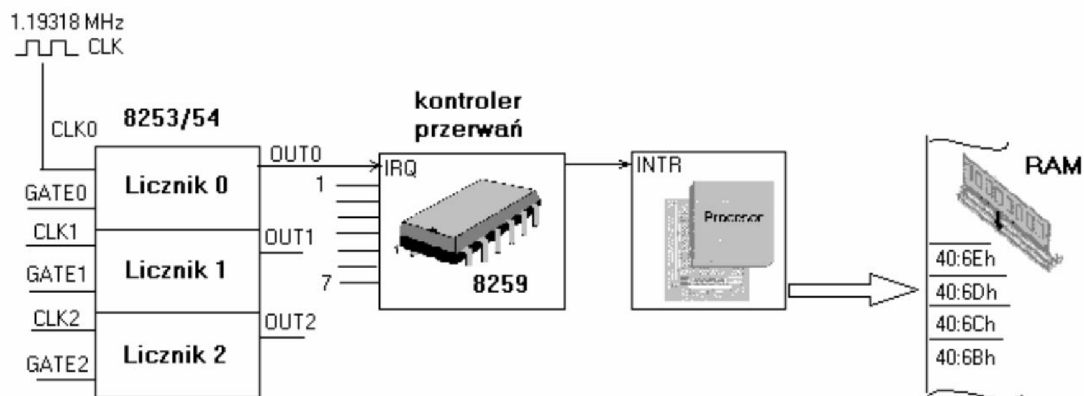
### 4.3. Zegar systemowy CTC.

Jak zaznaczono wcześniej, zegar systemowy został zrealizowany przez generator nr 0, układu 8253 pracujący w trybie 2 (*Rys.4.2.*). Taktowany sygnałem CLK, z częstotliwością równą 1,19318 MHz. Licznik CE rozpoczyna odliczanie od wartości FFFFh (współczynnik podziału 65536), czego wynikiem są generowane impulsy (18,2065 Hz), wywołujące tzw. zegarowe przerwanie sprzętowe na linii IRQ0, które jest obsługiwane poprzez procedurę INT 08h. Procedura ta inkrementuje każdorazowo stan 32-bitowego licznika zlokalizowanego w obszarze danych BIOS:

- 0040h:006Ch – mniej znaczące słowo licznika;
- 0040h:006Eh – bardziej znaczące słowo licznika;
- 0040h:0070h – bajt będący wskaźnikiem przepełnienia licznika.

Licznik ten jest podstawowym źródłem informacji o upływającym czasie, wykorzystywanym przez system operacyjny (*Rys.4.3.*). Po każdej inkrementacji, jest sprawdzane czy wartość licznika nie przekracza 1573040 impulsów, co odpowiada 24 godzinom, jeśli tak, to jest on zerowany.

Rys.4.3. Zasada działania zegara systemowego.



Licznik ten zlicza jednak czas od momentu startu systemu, a zatem nie pozwala jednoznacznie określić bieżącej daty i czasu. Dlatego też komputery są standardowo wyposażone w RTC.

#### 4.4. RTC-(*real time clock*) zegar czasu rzeczywistego.

Zegar czasu rzeczywistego (z podtrzymaniem baterijnym), którego zawartość jest automatycznie przepisywana do pamięci komputera w chwili startu systemu.

Informacje te są zapisane w pamięci CMOS-RAM, pamięć ta zawiera także informacje o konfiguracji sprzętowej oraz kalendarz.

#### 4.5. CMOS-RAM (Zwana też NVR - *Non Volatile RAM*).

Pamięć nie ulotna o swobodnym dostępie. Pamięć ta charakteryzuje się stosunkowo dużym czasem dostępu, (co w tym przypadku nie odgrywa żadnej roli) i bardzo małym poborem prądu. W przeciwieństwie do pamięci RAM, nie wymaga ona odświeżania, a jedynie stałego zasilania napięciem rzędu kilku wolt. Jest to realizowane za pomocą baterii litowej lub małego akumulatora, doładowywanego zwykle z zasilacza sieciowego komputera. Na podstawie informacji zawartej w pamięci CMOS jest inicjalizowany również zegar systemowy DOS-u. Zegar może pracować w trybie 12 lub 24-godzinnym, rozpoznawać czas letni i zimowy, określać

czas letni i zimowy, określać dzień tygodnia i miesiąca. System DOS nie korzysta z tych wbudowanych w układ możliwości.

#### 4.5.1. Układ MC146818.

Funkcję pamięci konfiguracji i zegara czasu rzeczywistego pełni w PC/AT układ scalony MC146818. Najistotniejszą część tego układu stanowi 64-komórkowa pamięć CMOS-RAM. Do każdej 8-bitowej komórki pamięci mamy swobodny dostęp, tzn. odczyt i zapis nie są poddane żadnym ograniczeniom technicznym.

Znaczenie pierwszych 64 bajtów jest ściśle określone, natomiast postać informacji zapisywanych powyżej adresu 3Fh nie jest znormalizowana. W zależności od producenta, w obszarze tym są składowane różne informacje rozszerzonego programu konfiguracyjnego (*Extended Setup*). Pierwsze dziesięć bajtów (00h-09h) jest przeznaczonych na dane zegara i kalendarza, następne sześć (0Ah-0Fh) to specjalne rejestry stanu, pozostałe natomiast zawierają niezwiązane z tym tematem, różne parametry konfiguracyjne takie jak: rozmiar pamięci, typy dysków elastycznych i twardych itd.

#### 4.5.2. Opis zawartości pamięci.

Adres Opis

00h	sekunda zegara czasu rzeczywistego RTC (w kodzie BCD)
01h	sekunda ustawienia alarmu, budzika (BCD)
02h	minuta zegara czasu rzeczywistego (BCD)
03h	minuta ustawienia budzika (BCD)
04h	godzina zegara czasu rzeczywistego (BCD)
05h	godzina ustawienia budzika
06h	dzień tygodnia (1 - niedziela, 2 - poniedziałek itd.)
07h	dzień miesiąca (BCD)
08h	miesiąc (BCD)
09h	rok (BCD, dwie ostatnie cyfry)

0Ah rejestr stanu A (zegar czasu rzeczywistego)

    bity 0-3    współczynnik podziału czasu (0110 – 1024)

    bity 4-6    podstawa czasu (010 – 32768 Hz)

    bit 7      jeśli ustawiony, to trwa aktualizacja

0Bh rejestr stanu B (zegar czasu rzeczywistego)



- bit 0 czas letni (1-aktywna, 0-wyłączona i tak jest domyślnie)
- bit 1 jeśli ustawiony, to 24-godzinny czas pracy (0-12 godzinny)
- bit 2 jeśli wyzerowany, to zapis binarny (1-dane w kodzie BCD i tak jest domyślnie)
- bity 3-6 generacja przerwania (1-aktywna, 0-zablokowana i tak jest domyślnie)
  - bit 3 sygnał prostokątny,
  - bit 4 od cyklu aktualizowania,
  - bit 5 od alarmu,
  - bit 6 okresowego,
- bit 7 tak jak w rej. A

#### 0Ch rejestr stanu C (obsługi CMOS)

- bity 0-3 zarezerwowane
- bit 4 jeśli 1 to źródłem jest przerwanie od alarmu
- bit 5 jeśli 1 to źródłem jest przerwanie od cyklu aktualizacji
- bit 6 jeśli 1 to źródłem jest przerwanie okresowe
- bit 7 jeśli 1 to żądanie IRQ od MC146818

#### 0Dh rejestr stanu D (obsługi CMOS)

- bity 0-6 zarezerwowane
- bit 7 jest bitem wiarygodności danych
  - 1-poziom napięcia baterii prawidłowy, dane wiarygodne
  - 0-zbyt niski poziom napięcia, dane niewiarygodne

#### 0Eh bajt diagnostyczny dla CMOS

- bity 0-1 zarezerwowane
- bit 2 jeżeli ustawiony, to poprawny czas
- bit 3 gdy włączony to niewłaściwy typ dysku twardego
- bit 4 gdy 1-rozmiar pamięci niezgodny z zapisem w CMOS
- bit 5 gdy 1-niezgodność konfiguracji z zapisem w CMOS
- bit 6 gdy 0-błąd sumy kontrolnej
- bit 7 gdy 0-przerwa w zasilaniu (wyczerpana bateria)

#### 0Fh bajt wyłączenia - bity określają przyczynę restartu systemu

## 4.6. Kod BCD (*Binary Coded Decimal* - kodowanie dwójkowo dziesiętne).

Format BCD jest akceptowany przez funkcje BIOS. Dwójkowy zapis liczb dziesiętnych przy użyciu cyfry 0 i 1. Zasada jest prosta: cyfra określająca jednostki jest zawarta w bitach 0-3 zaś cyfra określająca dziesiątki w bitach 4-7. Przykładowo, liczba dziesiętna  $94_{10}=10010100_{\text{BCD}}$ . Mimo, iż zapis w kodzie BCD nie jest oszczędny w stosunku do zapisu binarnego (dwójkowego) stosowany jest on ze względu na łatwą zamianę liczb. W kodzie binarnym liczba  $94_{10}=01011110_2$ .

## 4.7. Wiadomości praktyczne.

### 4.7.1. Programowanie generatora 8253/8254.

Do generatorów, dostajemy się poprzez porty o adresach:

- 040h - licznik generatora 0,
- 041h - licznik generatora 1,
- 042h - licznik generatora 2,
- 043h - rejestr sterujący.

Aby programować układ, przesyłamy do rejestru sterującego rozkaz o budowie:

- Bity 7-6    wybór generatora:
  - 00-generator 0,
  - 01-generator 1,
  - 10-generator 2,
  - 11-w ukł.8253 zabronione, dla 8254 rozkaz *Read-Back*,
- bity 5-4    rodzaj operacji:
  - 00-przepisanie zawartości licznika CE do rejestrów  $CR_H$  i  $CR_L$ ,
  - 01-zapis lub odczyt tylko mniej znaczącego bajtu licznika CE (LSB),
  - 10-zapis lub odczyt tylko bardziej znaczącego bajtu licznika CE (MSB),
  - 11-zapis lub odczyt obydwu bajtów licznika CE,
- bity 3-1    tryb pracy generatora:
  - 000-tryb 0,
  - 001-tryb 1,
  - 010-tryb 2,
  - 011-tryb 3,
  - 100-tryb 4,

- 101-tryb 5,
- bit 0      sposób odliczania:
  - 0-zliczanie w kodzie binarnym,
  - 1-zliczanie w kodzie BCD,

W układzie 8253 nie ma możliwości odczytania stanu początkowego licznika CE, ustawionego trybu pracy ani tego, czy generator programowany był jednym, czy dwoma bajtami oraz w jakim kodzie zlicza. Umożliwia to układ 8254 dzięki rozkazowi *Read-Back*.

Wysyłamy pod adres 043h rozkaz o postaci:

- bity 7-6    zawsze 11 jest to sygnatura tego rozkazu,
- bit 5      0-rozkaz podania stanu licznika CE,
- bit 4      0-rozkaz podania statusu,
- bit 3      1-wybór licznika 2,
- bit 2      1-wybór licznika 1,
- bit 1      1-wybór licznika 0,
- bit 0      zawsze 0,

Na porcie wybranego licznika otrzymujemy:

- dla stanu licznika, gdy ustawimy bit 5, jeden lub dwa bajty (w zależności od sposobu zaprogramowania),
- dla określenia statusu, gdy ustawimy bit4:
  - bit 7      stan wyjścia OUT: 1-wysoki,0-niski,
  - bit 6      0-licznik CE załadowany i może być odczytany,  
1-CE jeszcze nie jest załadowany,
  - bity 5-4    tryb programowania licznika CE:
    - 00-kombinacja zabroniona,
    - 01-zapis lub odczyt tylko mniej znaczącego bajtu CE,
    - 10-zapis lub odczyt tylko bardziej znaczącego bajtu CE,
    - 11-zapis lub odczyt obydwu bajtów CE,
  - bity 3-1    tryb pracy generatora:
    - 000-tryb 0,
    - 001-tryb 1,
    - 010-tryb 2,
    - 011-tryb 3,
    - 100-tryb 4,
    - 101-tryb 5,
  - bit 0      sposób odliczania:
    - 0-zliczanie w kodzie binarnym,
    - 1-zliczanie w kodzie BCD.

### 4.7.2. Programowanie za pomocą CMOS-RAM.

Dostęp do danych zawartych w CMOS-ie zapewniają dwa porty o adresach 70h i 71h. Port 70h jest rejestrem adresowym pamięci CMOS. Do zaadresowania 64 bajtów w pamięci CMOS wystarcza 6 najmniej znaczących pozycji bitowych tego rejestru, a 7 bit steruje przełącznikiem przerwania NMI. Natomiast port 71h jest dwukierunkowym rejestrem danych. I tak, aby odczytać wartość komórki o adresie n należy przesłać ten adres do portu 70h a następnie z portu 71h odczytać żadaną wartość. W podobny sposób odbywa się zapisywanie do CMOSu (zamiast odczytywać, zapisujemy bajt do portu 71h). Jednak, aby zapisać do komórek 10h-4Dh, wymaga to uaktualniania sumy kontrolnej. Suma ta jest zapisana w bajtach 4Eh (LSB) i 4Fh (MSB) i stanowi 16-bitowy wynik zwykłego dodawania zawartości wszystkich bajtów w obszarze komórek 10h-4Dh.

### Obsługa zegara (przerwania):

Przerwania sprzętowe:

Przerwanie **INT 8h** (IRQ 0 - obsługiwane przez pierwszy układ sterownika przerwań 8259). Przerwanie zegara systemowego.

Przerwanie **INT 70h** (IRQ 8- obsługiwane przez drugi układ sterownika przerwań 8259). Przerwanie zegara czasu rzeczywistego. Jest zgłaszane przez zegar czasu rzeczywistego po upływie czasu, na jaki był on ustawiony.

Przerwania programowe BIOS i DOS.

Mając dostęp do listy funkcji choćby nawet podstawowych przerwań - INT 21h, INT 1Ah - otrzymujemy sporą gamę gotowych procedur, które można wykorzystać w programach bez potrzeby ich dołączania do kodu programu.

### 4.7.3. Przerwania BIOS.

Przerwanie **INT 1Ah** umożliwia dostęp zarówno do kalendarza i zegara czasu rzeczywistego zawartego w pamięci CMOS, jak i zegara systemowego DOS-u.

Funkcja **00h**

Działanie:

Funkcja ta odczytuje stan 32-bitowego licznika impulsów zegara systemowego, zlokalizowanego pod adresem 0040:006Eh (bardziej znaczące słowo) i 0040:006Ch (mniej znaczące słowo). Zwracana jest

liczba impulsów zliczonych od godziny 00:00. Jeżeli system pracował dłużej niż 24 godziny, to rejestr AL zawiera wartość różną od zera (w istocie AL zwraca stan znacznika przepełnienia licznika, umieszczonego pod adresem 0040:0070h). Wywołanie tej funkcji powoduje wyzerowanie rejestru przepełnienia.

Parametry wejściowe:

AH 00h

Wartości zwracane:

AL zawartość znacznika przepełnienia

CX bardziej znaczące słowo licznika impulsów

DX mniej znaczące słowo licznika impulsów

**Przykład:** („ BIOS dla programujących w językach C i C++ ”, B Taylor).

```
unsigned char czytajCTC( unsigned long *zmienna )
{
union REGS regs;
unsigned long rezultat;
regs.h.ah = 0; /* AH = 0 - odczytanie licznika */
int86(0x1a, &regs, &regs); /* Wywołanie funkcji */
rezultat = regs.x.cx; /* CX = starsze 16 bitów */
rezultat <<= 16; /* przesunięcie */
rezultat |= regs.x.dx; /* DX = młodsze 16 bitów */
*zmienna = rezultat; /* Zapisanie do zmiennej
programu wywołującego */
return( regs.h.al ); /* AL = znacznik
przekroczenia północy */
}
```

Funkcja **01h**

Działanie:

Wpisuje do licznika zegara systemowego zadaną wartość. Zeruje znacznik przepełnienia.

Parametry wejściowe:

AH 01h

CX bardziej znaczące słowo licznika impulsów

DX mniej znaczące słowo licznika impulsów

### Funkcja **02h** - Działanie:

Funkcja ta odczytuje stan zegara czasu rzeczywistego z CMOS (układ MC146818).

Parametry wejściowe:

AH 02h

Wartości zwracane:

AH 00h

CL minuty (BCD)

CH godziny (BCD)

DH sekundy (BCD)

DL kod rodzaju czasu

0-czas standardowy

1-czas letni

CF 1 w przypadku wystąpienia błędu

### Funkcja **03h**

Działanie:

Ustawia stan zegara czasu rzeczywistego

Parametry wejściowe:

AH 03h

CL minuty (BCD)

CH godziny (BCD)

DH sekundy (BCD)

DL kod rodzaju czasu (jak wyżej)

Wartości zwracane:

AH 00h

CF 1-w przypadku wystąpienia błędu

### Funkcja **04h**

Działanie:

Odczytuje datę z kalendarza CMOS.

Parametry wejściowe:

AH 04h

Wartości zwracane:

CL dwie ostatnie cyfry roku (BCD)

CH stulecie, wiek (BCD)

DL dzień (BCD)

DH miesiąc (BCD)

CF 1 w przypadku wystąpienia błędu

**Przykład:** („ BIOS dla programujących w językach C i C++” , B Taylor).  
Odczytanie bieżącej daty z RTC i jej konwersja na liczby całkowite.

```
void odczytDatyRTC unsigned int *r,
                    unsigned char *m,
                    unsigned char *d )
{
#define zamiana( bcd ) ( (bcd >> 4) * 10 )
#define reszta( bcd ) ( bcd & 0x0F )
union REGS regs;
int pomocnicza;
  regs.h.ah = 4; /* AH = 4 odczytaj bieżącą datę */
  int86(0x1a, &regs, &regs); /* Wywołanie funkcji */
                        /* CH = stulecia w BCD */
  *r = zamiana(regs.h.ch); /* Zamiana dziesiątek */
  *r += reszta(regs.h.ch); /* dodanie reszty */
  *r *= 100;              /* Pomnożenie przez 100 */
                        /* CL = rok w BCD */
  pomocnicza = zamiana(regs.h.cl); /* dziesiątki */
  pomocnicza += reszta( regs.h.cl ); /* reszta */
  *r += pomocnicza;        /* dodanie lat */
                        /* DH = miesiąc BCD */
  *m = zamiana( regs.h.dh ); /* dziesiątki */
  *m += reszta( regs.h.dh ); /* reszta */
                        /* DL = dzień BCD */
  *d = zamiana( regs.h.dl ); /* dziesiątki */
  *d += reszta( regs.h.dl ); /* reszta */
}
```

## Funkcja 05h

Działanie:

Ustawia datę w kalendarzu CMOS.

Parametry wejściowe:

- AH 05h
- CL dwie ostatnie cyfry roku (BCD)
- CH stulecie, wiek (BCD)
- DL dzień (BCD)
- DH miesiąc (BCD)

Wartości zwracane:

CF 1 w przypadku wystąpienia błędu

### Funkcja 06h

Działanie:

Programuje ona czas alarmu zegara czasu rzeczywistego w CMOS. Osiągnięcie przez zegar zadanego czasu alarmu wyzwała przerwanie 4Ah, włącza linie przerwań IRQ8.

Parametry wejściowe:

AH 06h

CL minuta alarmu (BCD)

CH godzina alarmu (BCD)

DH sekunda alarmu (BCD)

Wartości zwracane:

CF 1 w przypadku wystąpienia błędu

### Funkcja 07h

Działanie:

Funkcja ta powinna poprzedzać wywołanie funkcji 06h, usuwając z rejestrów alarmu ewentualnie wcześniejsze wartości. Nie wyłącza ona linii przerwania IRQ8.

Parametry wejściowe:

AH 07h

Wartości zwracane:

CF 1 w przypadku wystąpienia błędu

Przerwanie INT 1Ch, obsługa przerwania zegara czasu systemowego. Przerwanie wywoływane przez procedurę BIOS-u obsługi zegara (przerwanie zegarowe 08h), około 18,2 razy na sekundę (układ 8253/8254). Użytkownik może wpisać adres swojej procedury do tablicy wektorów przerwań pod adres tego przerwania. Procedura ta będzie okresowo wywoływana przez BIOS. Powrót z tej procedury, winien nastąpić instrukcją IRET.

#### 4.7.4. Przerwania DOS.

Przerwanie INT 21H, funkcja 2AH - pobierz datę

Parametry wejściowe:

AH 2Ah

Wartości zwracane:

CX rok (1980-2099)



DH miesiąc (1-styczeń, 2-luty...)  
DL dzień (1-31)  
AL dzień tygodnia (0-niedziela, 1-poniedziałek...)

**Przykład:** („ BIOS dla programujących w językach C i C++ ”, B Taylor).

```
void odczytDatyDOS (unsigned int *r,  
                   unsigned char *m,  
                   unsigned char *d,  
                   unsigned char *ktory)  
{  
union REGS regs;  
  regs.h.ah = 0x2A;  
  intdos( &regs, &regs );  
  *r = regs.x.cx;  
  *m = regs.h.dh;  
  *d = regs.h.dl;  
  *ktory= regs.h.al;  
}
```

Przerwanie **INT 21H**, **funkcja 2BH** – ustaw datę

Parametry wejściowe:

AH 2Bh  
CX rok (1980-2099)  
DH miesiąc (1-styczeń, 2-luty...)  
DL dzień (1-31)

Wartości zwracane:

AL 0-date wprowadzono prawidłowo  
AL 0FFh-nieprawidłowo wprowadzona data

Przerwanie **INT 21H**, **funkcja 2CH** - pobierz czas

Parametry wejściowe:

AH 2CH

Wartości zwracane:

CH godzina (0-23)  
CL minuta (0-59)  
DH sekunda (0-59)  
DL setne sekundy (0-99)

Przerwanie **INT 21H**, **funkcja 2DH** - ustaw czas

Parametry wejściowe:

- AH 2Dh
- CH godzina (0-23)
- CL minuta (0-59)
- DH sekunda (0-59)
- DL setne sekundy (0-99)

Wartości zwracane:

- AL 0-czas wprowadzono prawidłowo
- AL 0FFh-nieprawidłowo wprowadzony czas

Przerwanie **INT 21H**, **funkcja 38H**-odczyt informacji o sposobie zapisu i czasu dla różnych krajów

Parametry wejściowe:

- AH 38h
- DS:DX adres 32-bajtowego bloku
- AL 0

Wartości zwracane:

- CF 0-wykonanie poprawne
- CF 1-błąd a w AX kod błędu

32-bajtowy blok zawiera:

- 2 bajty dotyczące formatu daty/czasu
- 1 bajt symbolu waluty (ASCII)
- 1 bajt ustawiony na 0
- 1 bajt jako separator części tysięcznych (ASCII)
- 1 bajt ustawiony na 0
- 24 bajty użyte wewnętrznie

Przykładowe wartości formatu data/czas

- 0-USA (H:M:S M/D/Y)
- 1-EUROPA (H:M:S D/M/Y)
- 2-JAPONIA (H:M:S D:M:Y)