

System obsługi przerwania sprzętowych

Najbardziej aktywnym elementem komputera jest procesor (CPU). Niektóre z bloków funkcjonalnych, które składają się na jego otoczenie, wykonują operacje rozciągające się w czasie na wiele cykli zegarowych. Urządzenia te nie wymagają udziału procesora dla wykonania zleconych im zadań. Są to zwykle mniej lub bardziej niezależne kontrolery. Na ogół wystarczy, by procesor zaprogramował rejestry sterujące tego urządzenia, wydając tym samym polecenie wykonania operacji (odebranie znaku przez łącze szeregowo, zapisanie sektora na dysku).

Zainicjowana przez CPU operacja może trwać tysiące cykli zegarowych procesora, ale musi się kiedyś zakończyć (choćby błędem). Musi więc istnieć sposób poinformowania procesora o tym fakcie. Istnieją zasadniczo dwie możliwości:

- procesor okresowo wysyła zapytanie do urządzenia. Częstotliwość „odpytywania” jest ustalana przez wykonywany program i stanowi kompromis pomiędzy szybkością reakcji procesora na zakończenie operacji a spowolnieniem jego pracy nad zadaniem pierwszoplanowym. Kod procedury odpytywania musi być zintegrowany z programem, co powiększa jego rozmiar. Zbyt częste odpytywanie hamuje wykonywanie bieżącego programu. Zbyt długie przedziały czasowe między kolejnymi odpytywaniami i nierównomiernie napływający strumień danych mogą spowodować, że część z nich zostanie utracona na skutek przepełnienia nie opróżnianego bufora wewnętrznego obsługiwanego urządzenia;
- urządzenie samo zgłasza wykonanie zadania. W najbliższym dogodnym dla siebie momencie procesor przerywa realizację aktualnie wykonywanego programu, zapamiętuje stan swoich rejestrów, i przechodzi do realizacji procedury reagującej na zgłoszenie urządzenia. Kod tej procedury na ogół nie znajduje się w ciele programu, lecz w pamięci stałej komputera. Po tym odtwarza stan swoich rejestrów i powraca do wykonywania przerwanej operacji.

Pierwsza metoda jest znana pod nazwą *polling* i nie znajduje praktycznego zastosowania w komputerach PC. W drugim przypadku mówimy, że wystąpiło przerwanie (*interrupt*) sprzętowe (pochodzące od urządzenia). Ta metoda jest w PC szeroko wykorzystywana. Znajduje ona zastosowanie w obsłudze urządzeń intensywnie wymieniających dane z pamięcią operacyjną, a więc przede wszystkim: łącz szeregowych i równoległych, kontrolerów

dysków (twardych i elastycznych), kart sieciowych oraz urządzeń, które dostarczają dane w sposób nieregularny - klasycznym przykładem jest tutaj klawiatura komputera. Trudno sobie wyobrazić bardziej rozsądne podejście do obsługi klawiatury, niż reagowanie na naciśnięcie klawisza za pomocą obsługi przerwania pochodzącego od sterownika klawiatury.

W przypadku konieczności odczytania sektora z twardego dysku czas pozycjonowania głowic nad żadaną ścieżką i wyszukania sektora jest o dwa rzędy wielkości większy niż czas potrzebny na przesłanie zawartych w tym sektorze 512 bajtów. Procesor jest więc zaangażowany w tę operację tylko w jej małym ułamku czasowym.

Procesor posiada jednak zwykle tylko jedno wejście INT sygnalizujące przerwanie sprzętowe. Ponadto architektura PC zakłada, że każde przerwanie ma pewien priorytet, tzn. obsługa zgłoszeń uwzględnia nie tyle moment ich nadejścia, co raczej stopień ich ważności. Już samo to założenie wymaga rozróżniania zgłoszeń poprzez przyporządkowanie każdemu z urządzeń oddzielnego kanału wejściowego.

Konieczne stało się więc zastosowanie specjalizowanego układu zajmującego się wyłącznie ich obsługą. Funkcję kontrolera przerwania w komputerze XT pełni układ 8259A, zaś w modelach AT i PS/2 — dwa takie układy. W nowszych rozwiązaniach płyt głównych komputerów AT układy te wchodziły przeważnie w skład jednego z układów wysokiej skali integracji, tworzących „obudowę” procesora (*chipset*). Ze względu jednak na konieczność zachowania zgodności architektury można się odwoływać do ich rejestrów.

Kontroler 8259A ma osiem linii wejściowych, które przyjmują sygnały zgłoszeń. Układy 8259A mogą być łączone w kaskady, powielając tym samym liczbę dostępnych wejść. Istnieje również (nie wykorzystywana w architekturze PC) możliwość sterowania zewnętrznym buforem pośredniczącym między układem a silnie obciążonymi magistralami systemowymi. Warto też wspomnieć, że w modelach IBM PC/XT i AT wejścia IR_n reagują na zbocze impulsu przerwania, co stwarza niebezpieczeństwo interpretacji impulsu zakłócającego jako zgłoszenia przerwania. Inaczej zagadnienie to rozwiązano w architekturach EISA, MCA i PCI, gdzie wyzwalamie przerwania odbywa się poziomem napięcia. Możliwe jest tutaj dzielenie jednej linii zgłoszeń przez kilka urządzeń (*shared interrupt*).

Cykl przyjęcia zgłoszenia

Układ 8259A dysponuje trzema 8-bitowymi rejestrami:

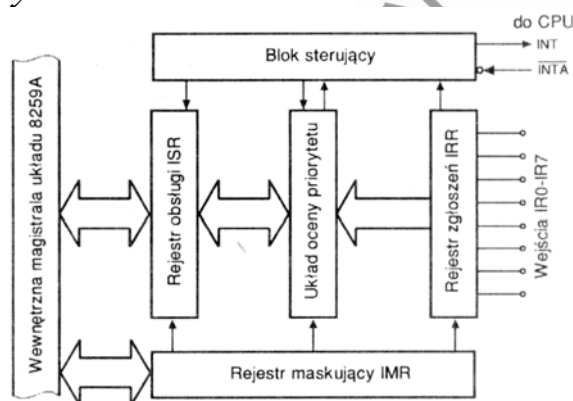
IMR - (*Interrupt Mask Register*) rejestr maski przerw

IRR - (*Interrupt Request Register*) rejestr zgłoszenia przerw

ISR - (*Interrupt Service Register*) rejestr obsługi przerw

Pozycja bitowa n każdego z nich odpowiada jednemu z wejść IR_n . Schematycznie przedstawia to rysunek

Schemat blokowy układu 8259A



Urządzenie połączone z wejściem o numerze n zgłasza przerwanie wymuszając wysoki poziom logiczny na linii IR_n . Ustawiając bit n w rejestrze IMR (*Interrupt Mask Register*) można spowodować ignorowanie wszelkich przerw nadchodzących tą linią.

Jeżeli wspomniany bit jest wyzerowany, informacja o przerwaniu jest przekazywana dalej i powoduje ustawienie bitu n w rejestrze IRR (*Interrupt Request Register*). Priorytet zgłoszeń na wejściach IR_0 — IR_7 nie jest jednakowy i maleje ze wzrostem numeru wejścia. Ponieważ jednocześnie mogą nadchodzić zgłoszenia z wielu linii, układ oceny priorytetu wybiera jednorazowo tylko jeden - ten o najwyższym priorytecie. Jednocześnie jest aktywowany sygnał INT, informujący procesor o konieczności obsługi przerw. Jeżeli obsługa przerw nie jest programowo zablokowana (bit IE rejestru stanu procesora), ten ostatni reaguje sygnałem potwierdzenia na wyjściu INTA, które jest połączone z wejściem - \sim INTA układu 8259A.

Wybrany przez układ oceny priorytetu bit rejestru IRR przekazywany jest teraz do rejestru ISR (*Interrupt Service Register*). Oznacza to, że w tym momencie obsługiwane jest przerwanie z linii o numerze odpowiadającym ustawionej pozycji bitowej rejestru ISR.

Procesor wysłała teraz drugi impuls \sim INTA. Kontroler przerwań odpowiada wystawieniem na szynę danych D7-D0 wektora przerwań o następującej budowie:

7	6	5	4	3	2	1	0
---	---	---	---	---	---	---	---

bity 7-3 bity te, stanowiące tzw. *offset*, można ustalić na etapie programowania kontrolera;

bity 2-0 numer ustawionego w rejestrze ISR bitu.

Procesor traktuje ten bajt jako numer procedury obsługi przerwania, do wykonania której teraz przejdzie.

Jeżeli kontroler pracuje w tzw. trybie automatycznym (AEIO — *Automatic End Of Interrupt*), w momencie przejścia do wykonywania procedury obsługi przerwania zerowany jest odpowiedni bit rejestru ISR. W przeciwnym wypadku (tryb EOI) wyzerowania tego bitu dokonuje sama procedura obsługi przerwania wysyłając rozkaz EOI (*End Of Interrupt*) do kontrolera. Tryb pracy kontrolera (AEIO/EOI) określany jest w fazie programowania.

Niezależnie od trybu pracy, moment wyzerowania bitu rejestru ISR odpowiadającego aktualnie obsługiwanemu kanałowi oznacza gotowość kontrolera do przyjęcia następnego zgłoszenia. Należy jeszcze raz podkreślić, że zgłoszenia nie są obsługiwane w miarę nadchodzenia, istotny jest ich priorytet. Oznacza to, że przy pełnej kolejce oczekujących na liniach IR1-7 zgłoszeń, zarejestrowanie zgłoszenia na linii IR0 spowoduje zrealizowanie go w pierwszej kolejności, tj. po zakończeniu aktualnie wykonywanego cyklu obsługi.

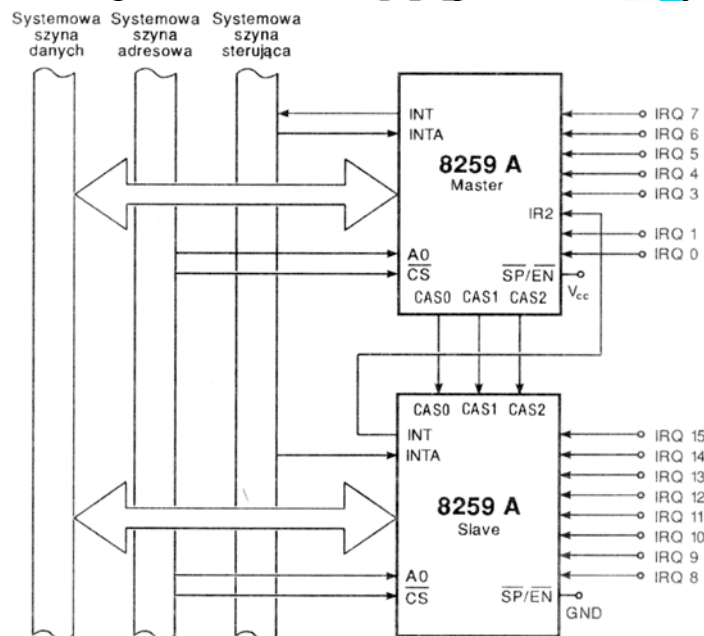
Kaskadowe łączenie kontrolerów przerwań

Dla zwielokrotnienia liczby wejść układy 8259A dają się łączyć w grupy. Teoretycznie możliwe jest połączenie 9 układów (jeden *Master* + 8 układów *Slave*) i uzyskanie $8 \cdot 8 = 64$ linii przerwań. W modelach AT i PS/2 zastosowano kaskadowe połączenie dwóch kontrolerów uzyskując tym samym $8 + 7 = 15$ wejść, bowiem jedno z wejść układu *Master* „zużywamy” na dołączenie układu *Slave*. Należy nadmienić, że połączenie takie prowadzi do

przegrupowania priorytetów wejść. Wejścia IR8-15 (układu *Slave*) otrzymują wyższy priorytet niż wejścia IR3-7 układu *Master*.

Układy połączone są między sobą trzema liniami adresowymi CAS0-CAS2, które dla kontrolera *Master* są wyjściami, a dla *Slave* wejściami. Każdy z kontrolerów rozpoznaje swoją rolę (*Master* lub *Slave*) badając poziom sygnału na wejściu \sim SP/EN. Zero logiczne na tym wejściu konfiguruje układ jako *Slave*, a jedynka logiczna jako *Master*.

Do obydwu kaskadowo połączonych kontrolerów doprowadzony jest z procesora sygnał \sim INTA, lecz tylko układ *Master* łączy się swoim wyjściem INT z wejściem INTR procesora. Schemat kaskadowego połączenia dwóch układów 8259A w komputerach AT i PS/2 przedstawia rysunek



Obsługa zgłaszanych przerw przez tak połączone układy nieco się komplikuje. Przerwania nadchodzące do układu *Master*, tj. IRQ0, IRQ1, IRQ3-IRQ7 są obsługiwane tak, jak to opisano wcześniej. Odebranie zgłoszenia na linii układu *Slave* (IRQ8-IRQ15) powoduje wyzwolenie następującej reakcji:

- zgłoszenie ustawia bit n rejestru IRR w układzie *Slave* i powoduje, po opracowaniu priorytetów, wysłanie sygnału INT do układu *Master*
- układ *Master* otrzymuje zgłoszenie na linii IR2, co powoduje ustawienie bitu 2 w jego rejestrze IRR
- układ *Master*, po opracowaniu priorytetów własnych zgłoszeń, wystawia sygnał INT do procesora

- procesor (jeśli nie zablokowano obsługi przerw sprzętowych) odpowiada pierwszym sygnałem $\sim\text{INTA}$
- układ *Master* adresuje liniami CAS0-CAS2 układ *Slave* odpowiadający za obsługę zgłoszonego przerwania
- w obu układach następuje jednoczesne przesunięcie bitu odpowiadającego obsługiwanej linii kanałowi (bit 2 w układzie *Master* i bit n w układzie *Slave*) z rejestru IRR do rejestru ISR
- drugi impuls $\sim\text{INTA}$ wygenerowany przez CPU nakazuje układowi *Slave* wystawienie na szynę danych (D7-D0) bajtu danych – wektora przerw, traktowanego jako numer procedury obsługi przerwania zgłoszonego na linii n
- w trybie AEOI są zerowane rejestry ISR (bit n w układzie *Slave* i bit 2 w układzie *Master*). W trybie EOI procedura obsługi przerwania wysyła dwa sygnały EOI: jeden do układu *Master* i drugi do układu *Slave*

Przerwanie niemaskowalne (NMI)

Przerwanie to jest zaliczane do grupy przerw sprzętowych, bowiem nie można go wygenerować instrukcją kodu maszynowego. W przeciwieństwie do przerw odbieranych liniami IRQ, które obsługiwane są przez dedykowany do tego celu kontroler 8259A, sygnał przerwania NMI jest kierowany bezpośrednio do końcówki NMI procesora. Wysoki sygnał logiczny na tym wejściu prowadzi, natychmiast po zakończeniu wykonywania bieżącego rozkazu, do wywołania procedury **INT 2**. Przerwanie takie nie może być zamaskowane ani rozkazem **CLI** procesora, ani przez operacje na rejestrze IMR kontrolera 8259A. Ma ono najwyższy priorytet ze wszystkich przerw sprzętowych.

Wywołanie przerwania NMI w przypadku komputerów PC ma miejsce po stwierdzeniu błędu parzystości pamięci operacyjnej, co należy uznać za sytuację uniemożliwiającą dalsze użytkowanie komputera. Procedura obsługi **INT 2** wyprowadza na monitor stosowny komunikat (*Parity error at ...*) i wstrzymuje pracę systemu. Jedynym wyjściem w tym momencie jest ponowny restart systemu w nadziei, że błąd parzystości spowodowany został na przykład chwilową utratą styku lub przegrzaniem modułów RAM, a nie ich trwałym uszkodzeniem. Źródłem tego przerwania może też być sygnał generowany przez uszkodzony koprocesor arytmetyczny.

Przerwanie niemaskowalne nie jest „absolutnie nie do zamaskowania”. Rozpatrzmy moment włączenia komputera i uruchomienia procedur inicjalizujących BIOS-u. Jednym z ich zadań jest budowa tablicy wektorów przerwań, tj. stałego miejsca w „niskim” obszarze pamięci operacyjnej, w którym umieszczane są czterobajtowe adresy punktów wejścia do procedur obsługi wszystkich przerwań. Jednocześnie inicjalizowane są wszystkie inne rejestry systemu. Własną inicjalizację przeprowadzają układy obsługi pamięci dynamicznej. Może się zdarzyć, że układ kontroli parzystości RAM stwierdzi błąd i wywoła tym samym procedurę obsługi przerwania **INT 2**. Adres punktu wejścia do tej procedury może jednak nie być jeszcze ustalony i wskazywać przypadkowe miejsce w pamięci. Procesor podejmując wykonanie programu od tego miejsca, najprawdopodobniej zawiesi się.

Jedną z pierwszych operacji, jakie musi wykonać BIOS uruchamiając komputer, jest więc zamaskowanie przerwania NMI. Odpowiada to ustawieniu bitu 7 portu o adresie **0A0h** (w XT) lub **070h** (w AT). Manipulując samodzielnie tym bitem należy zachować szczególną ostrożność - port ten bowiem steruje również pracą pamięci konfiguracji CMOS-RAM.

Bezpiecznie można to zrobić w następujący sposób:

```
unsigned char i;
/*...*/
i = inportb(0x70);
i &= 0x7f; /* wyzeruj bit 7, NMI aktywne */
outportb(0x70,i);
/*...*/
i=inportb(0x70);
i |= 0x80; /* ustaw bit 7, NMI zablokowane */
outportb(0x70, i);
/*...*/
```