

Systemy operacyjne

Wykład nr 2:

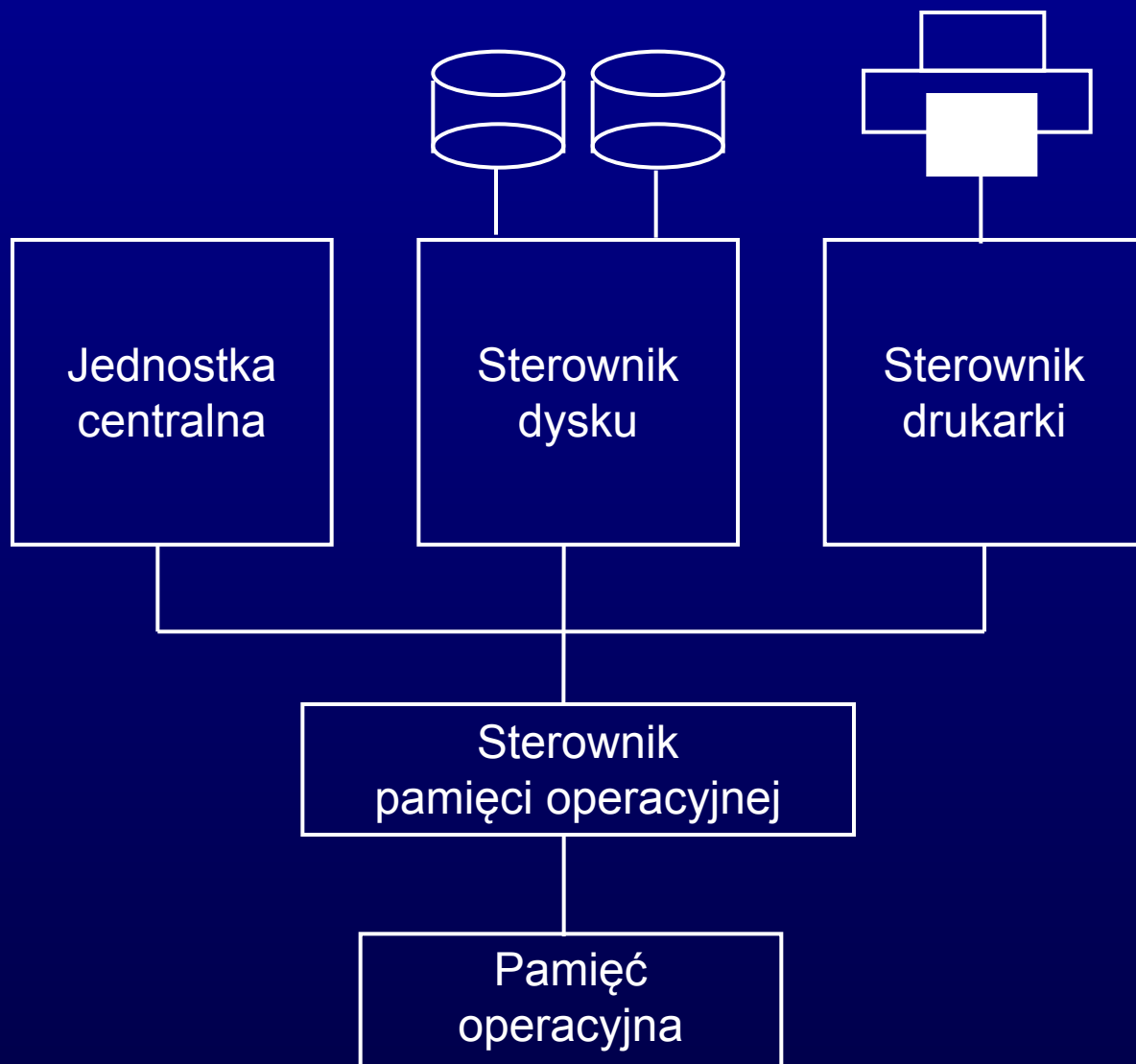
Struktura systemu komputerowego a
system operacyjny

Piotr Bilski

Informacje wstępne

- Omawiany system komputerowy zgodny z architekturą von Neumanna
- Zasadnicze elementy: procesor, sterowniki urządzeń i szyna systemowa
- Rozruch systemu komputerowego ma charakter sprzętowy

Schemat systemu komputerowego



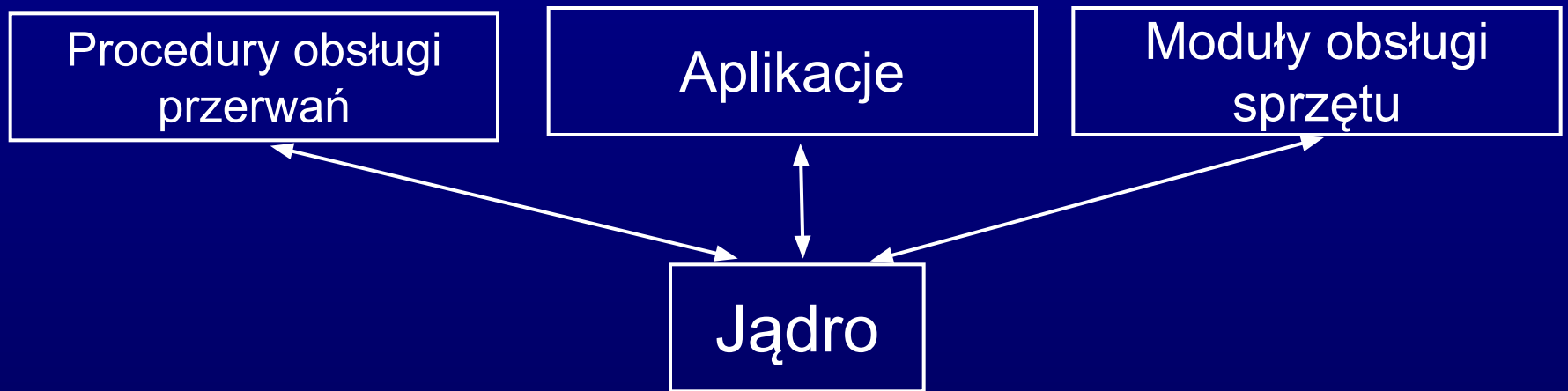
Start systemu komputerowego

- Początkowo kontrolę nad systemem pełni program rozruchowy (bootstrap)
- Znajduje się na ogół na płycie głównej w pamięci ROM lub podobnej
- Określa stan początkowy wszystkich elementów systemu
- Musi wiedzieć, gdzie jest system operacyjny

Start systemu operacyjnego

- Informacje o systemie operacyjnym oraz jego lokalizacji znajdują się w początkowym sektorze dysku twardego
- Najważniejsze jest jądro – „ścisle centrum” systemu operacyjnego
- Pierwszym procesem musi być proces inicjujący wszystkie inne (np. init)

Schemat systemu operacyjnego



- Modułowa budowa systemu ułatwia wprowadzanie zmian
- Dąży się do minimalizacji jądra (mikrojądro)

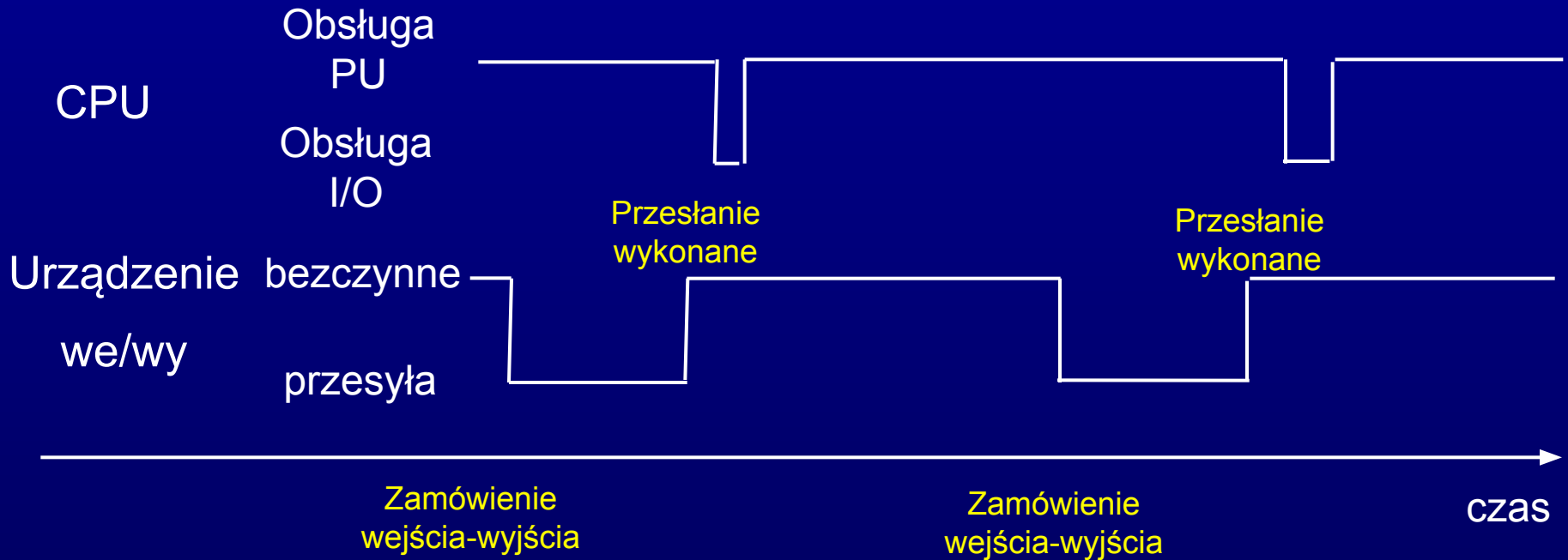
Systemy sterowane przerwaniem

- Zdarzenie sygnalizowane jest przerwaniem (interrupt)
- Źródło zdarzenia może być sprzętowe lub programowe
- Procedura obsługi przerwania systemu operacyjnego nazywana jest wywołaniem systemowym lub funkcją systemową

Obsługa przerwania programowych

- Gdy przerwania nie występują, system jest beczynny
- Zjście zdarzenia jest sygnalizowane za pomocą pułapki/wyjątku (trap/exception)
- Przyczyną wyjątku jest błąd programowy (np. dzielenie przez zero) lub wymaganie obsługi programu użytkownika przez system operacyjny
- Każde przerwanie ma własną procedurę

Obsługa przerwania sprzętowych



- Obsługa przerwania wymaga przejścia do konkretnego miejsca w pamięci (adres startowy procedury)
- Wektor przerwania przechowuje wskaźniki do poszczególnych procedur

Wywołanie systemowe

- Procedura systemu operacyjnego odpowiedzialna za wykonanie operacji (np. obsługę urządzenia)
- Konieczne przechowywanie adresu przetwarzanego rozkazu
- Obecnie najczęściej wykorzystuje się tu stos
- Przykłady implementacji: rozkaz trap

Schemat operacji wejścia-wyjścia

- Współpraca procesora i sterownika urządzenia
- Model synchroniczny i asynchroniczny
- Gdy procesor czeka na zakończenie operacji wejścia/wyjścia, wykonuje rozkaz wait (lub równoważną pustą pętlę)
- W systemie operacyjnym za czekanie odpowiada proces bezczynności

Synchroniczne a asynchroniczne wejście/wyjście

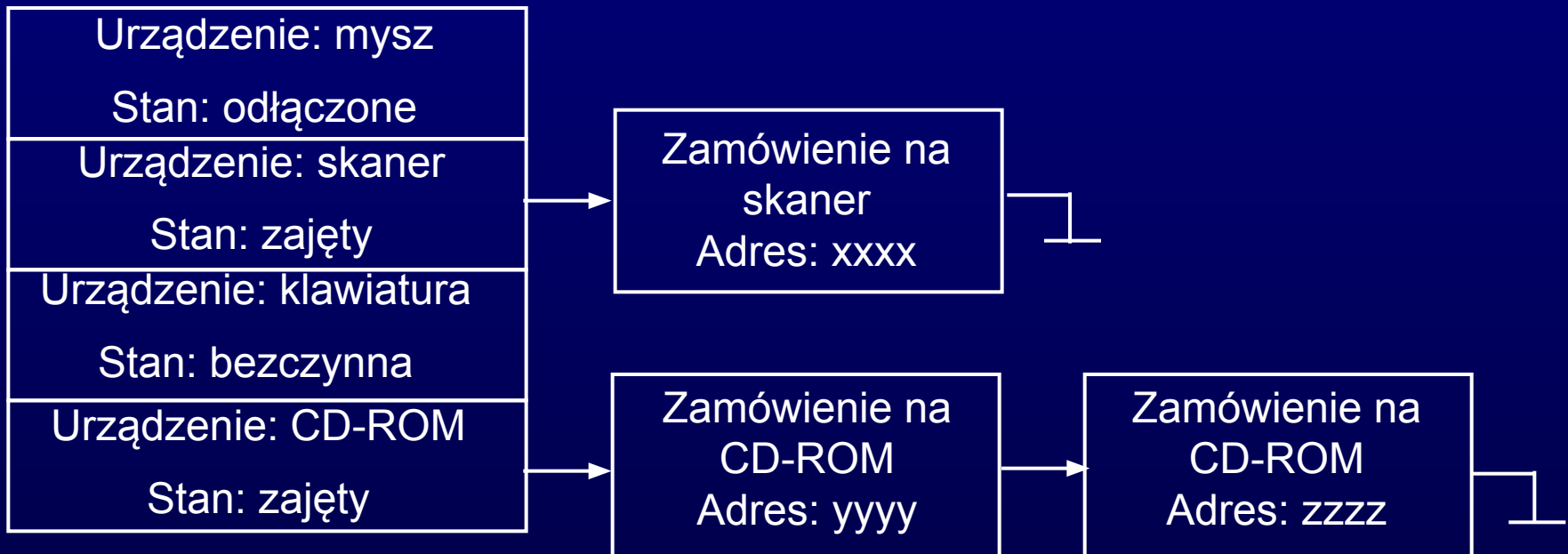


Zachowanie procesora podczas przerwania

- Synchroniczne
 - Specjalny rozkaz wejścia w pętlę czekania (wait)
 - Wada: tylko jedno urządzenie obsługiwane!
- Asynchroniczne
 - Jeśli brak innych operacji do wykonania, to wejście w pętlę czekania
 - Jeśli są inne urządzenia do obsługi, trzeba je obsłużyć
 - Rozwiązanie wydajne, obecnie standard

Obsługa wielu przerwań

- Tablica stanów urządzeń pozwala przechowywać żądania dostępu do urządzeń, które nie mogą być obsłużone natychmiast



Procedura obsługi przerwania

- Identyfikacja urządzenia odpowiedzialnego za przerwanie
- Pobranie informacji o stanie urządzenia z tablicy
- Wykonanie odpowiedniej procedury
- Oddanie sterowania programowi użytkownika lub systemowi operacyjnemu

DMA a system operacyjny

- Program użytkownika lub system operacyjny zgłasza żądanie komunikacji z szybkim urządzeniem wejścia-wyjścia (np. macierzą dyskową)
- SO wybiera bufor do odczytu lub zapisu
- Moduł obsługi urządzenia (device driver) konfiguruje jego rejestry
- Sterownik DMA wykonuje resztę pracy

Ochrona pamięci

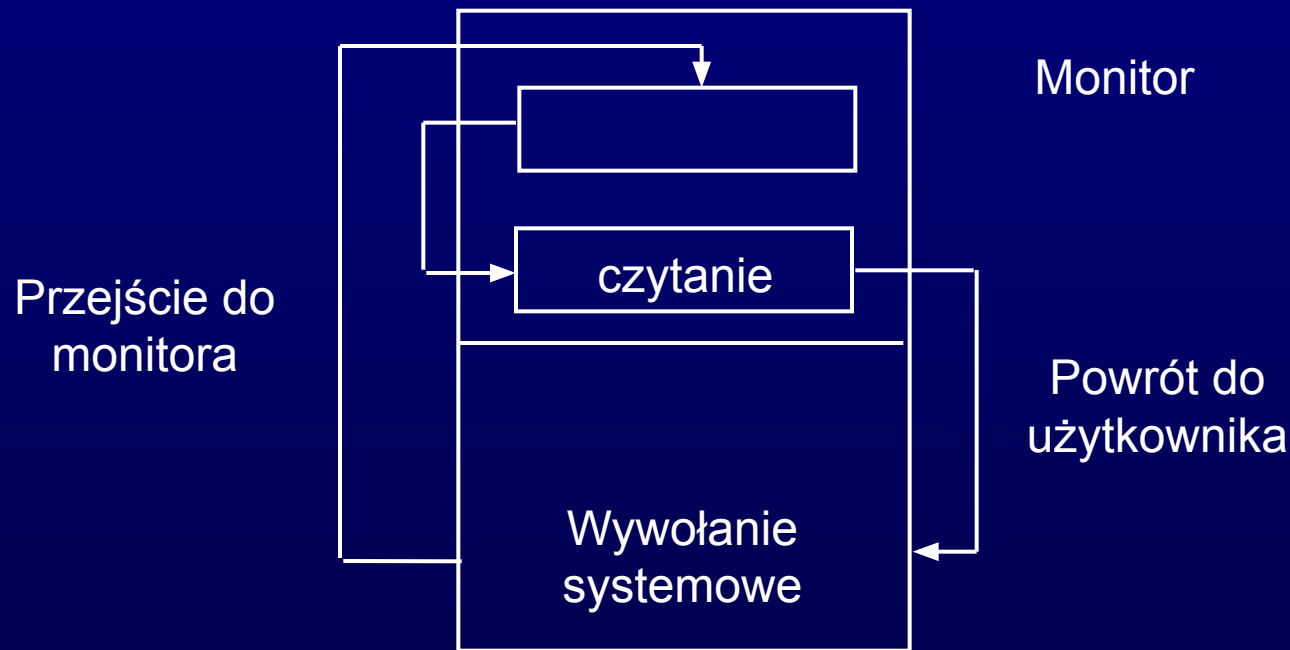
- Wieloprogramowość wymusza ochronę programów przed sobą
- Współczesne systemy (Windows, Linux) zapewniają taką ochronę, starsze (DOS) – nie
- Sprzęt powinien wykrywać takie sytuacje (pułapki!) – SO je obsługiwać

Tryby pracy SO

- Procesor wyposażony jest w bit trybu
- Tryby mają na celu ochronę sprzętu i SO
- Tryb użytkownika – obsługuje pracę programów użytkownika
- Tryb nadzorcy – zarezerwowany dla SO
- Programy użytkownika zlecają zadania dla SO (wywołania systemowe)
- Przejście do wykonania programu obsługi przerwania (wektor przerwań) powoduje ustawienie sprzętu w tryb nadzorcy

Ochrona wejścia-wyjścia

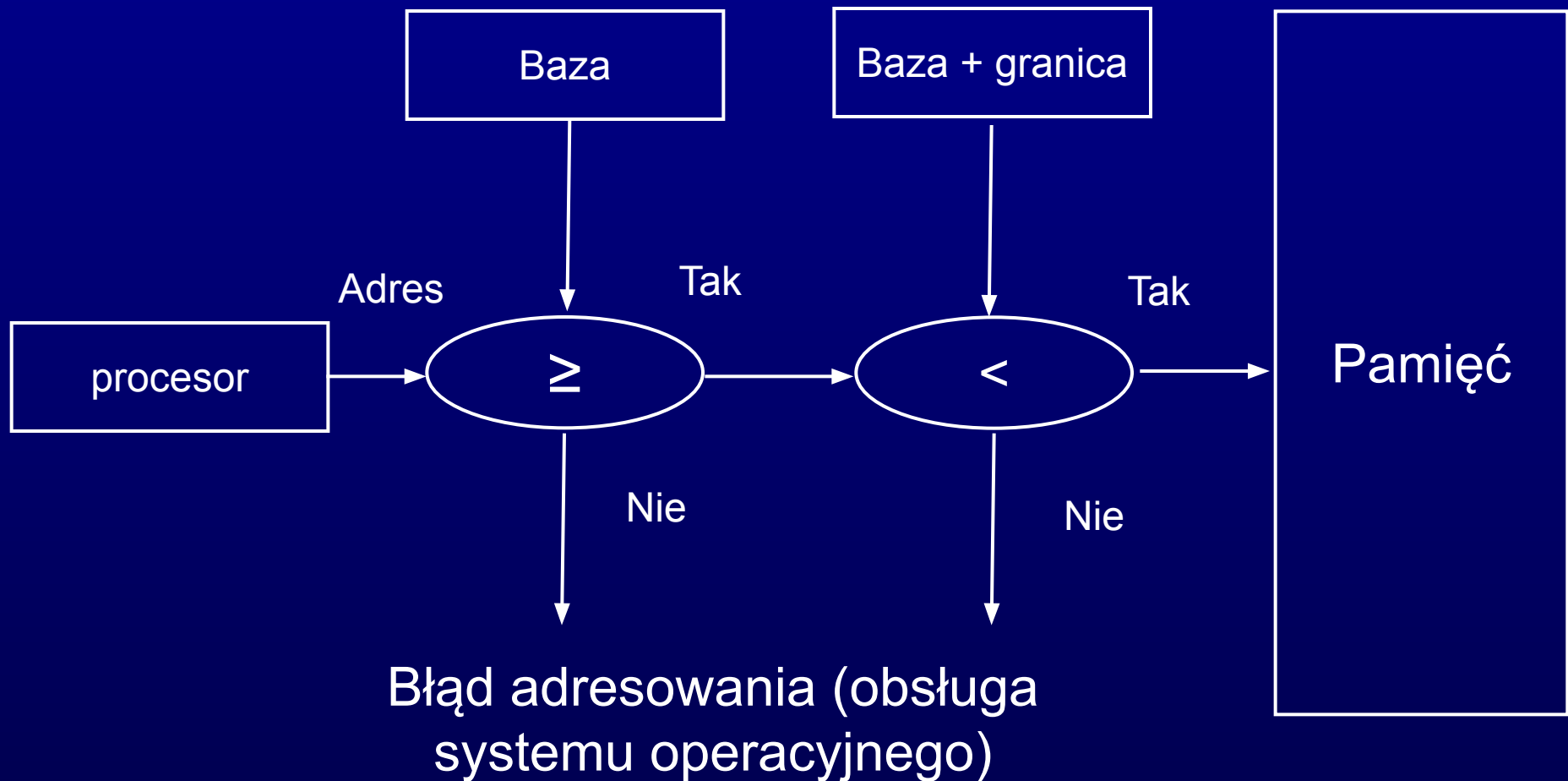
- Rozkazy wejścia-wyjścia wykonuje tylko system operacyjny



Ochrona pamięci

- Podstawowej ochronie podlegają:
 - Wektor przerwań
 - Procedury obsługi przerwań
- Dodatkowo programy są wzajemnie chronione
- Wsparcie sprzętowe: rejestr bazowy i graniczny
- Każdy wygenerowany adres jest weryfikowany!

Ilustracja działania rejestrów



Tylko SO ma dostęp do tych rejestrów!

Ochrona procesora

- Cel: zapobieganie zawieszeniom
- Środek: czasomierz
 - Stały
 - Zmienny
- Zastosowanie: systemy z podziałem czasu
- Kwant czasu – ilość czasu przydzielona procesowi, użytkownikowi itp.

Praca czasomierza

- W ustalonych momentach generuje przerwanie
- Umożliwia odebranie kontroli nad procesorem dowolnemu programowi
- Związany z przełączaniem kontekstu
- Używany do obliczania czasu systemowego