

POLSKA AKADEMIA NAUK
INSTYTUT PODSTAW INFORMATYKI

ORGANIZACJA MASZYN CYFROWYCH I MIKROPROGRAMOWANIE

Materiały z Ogólnopolskiego Sympozjum
Warszawa 24 – 26 września 1975 r.

Tom II

WARSZAWA 1977
PAŃSTWOWE WYDAWNICTWO NAUKOWE



REDAKTOR NAUKOWY
INSTYTUTU PODSTAW INFORMATYKI PAN

Zdzisław Pawlak

REDAKTOR

Mirosław Thor

Okladkę projektowała

Alicja Szubert-Olszewska

11. 688194,2

II

REDAKTOR WYDAWNICZY
INSTYTUTU PODSTAW INFORMATYKI PAN

Jan Lipski

Printed in Poland

Państwowe Wydawnictwo Naukowe
Oddział w Łodzi 1977

Wydanie I. Nakład 1300+120 egz. Ark. wyd. 17,50. Ark. druk. 17,00.
Papier offsetowy kl. V, 80 g. 70×100. Oddano do składania w lutym 1977 r.
Podpisano do druku w lipcu 1977 r. Druk ukończono w sierpniu 1977 r.
Zam 80/77. W-9 Cena zł 55,-

Zakład Graficzny Wydawnictw Naukowych
Łódź, ul. Żwirki 2

EO-77/1863/32
6.10, 55r



Thanasis KAMBURELIS
Ośrodek Badawczo-Rozwojowy
Maszyn Cyfrowych MERA-ELWRO

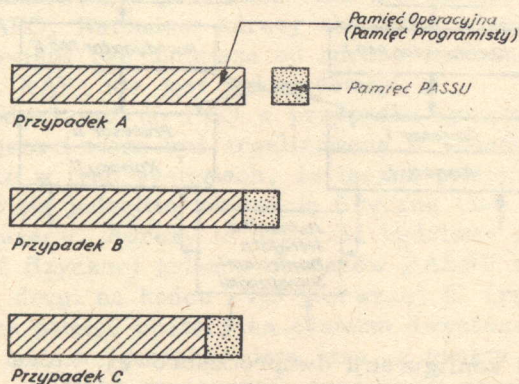
O ORGANIZACJI PAMIĘCI KANAŁÓW WEJŚCIA-WYJŚCIA
W DWUPROCESOROWYCH SYSTEMACH JS EMC

1. WPROWADZENIE

Jednostki centralne współczesnych maszyn cyfrowych współpracują jednocześnie z wieloma urządzeniami zewnętrznymi. Dla zapewnienia jednoczesnych przesyłań informacji do/lub z urządzeń zewnętrznych przyjmuje się różne kanały wejścia-wyjścia typu multipleksorowego mające zdolność organizowania wielu (np. do 256) jednoczesnych i niezależnych przesyłań informacji na zasadzie podziału czasu. Każde takie niezależne przesyłanie (podkanał) posiada komplet informacji opisujących jednoznacznie przebieg danego przesyłania. Informacje podkanału są przechowywane w EMC Jednolitego Systemu w czterech 32-bitowych słowach (czyli w 16 bajtach) zwanych Słowami Sterującymi Urządzeń Zewnętrznych (SSU). W nowych maszynach JS EMC przyjmuje się na ogół 512 podkanałów, dla których trzeba przewidzieć pamięć o pojemności 16 Kb (dla przechowywania SSU kanałów). Pamięć tę - tj. pamięć słów sterujących urządzeń - oznaczamy będziemy dalej PASSU.

Pamięć PASSU jest elementem występującym tylko na poziomie szczegółowej struktury logicznej i przez to nie jest ona dostępna dla programisty.

Przy projektowaniu wewnętrznej organizacji logicznej nowego komputera powstaje problem, jak zbudować pamięć PASSU oraz jak zorganizować współpracę kanałów z tą pamięcią w systemach jedno- i wieloprocesorowych.



Rys. 1. Trzy typowe organizacje pamięci słów sterujących urządzeń zewnętrznych

Dotychczasowe podejścia w organizacji PASSU (rys. 1) polegały przeważnie na przyjęciu jednego z trzech rozwiązań:

A. Wbudowanie w jednostce centralnej odrębnego i niezależnego bloku pamięci (np. o pojemności 4, 8 lub 16 Kb) dla przechowywania słów sterujących podkanałów.

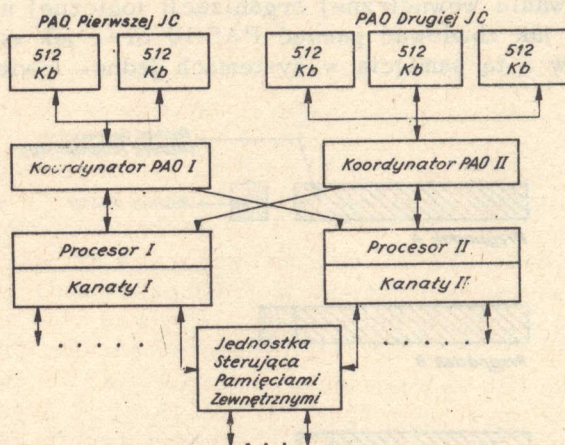
B. Rozszerzenie pojemności pamięci operacyjnej (PAO) komputera o dodatkową pojemność dla PASSU (np. zamiast "okrągłego" bloku 256 Kb przyjmuje się - w tej samej konstrukcji - blok "nieokrągły" o pojemności $256 + 4$ Kb).

C. Pomniejszenie pojemności pamięci operacyjnej o pojemność pamięci PASSU (np. zamiast pojemności 256 Kb programista dysponuje pamięcią o pojemności $256 - 4$ Kb).

Przypadek A pozwala na niezależną pracę pamięci PAO i PASSU, ale jest kosztowny w realizacji, gdyż wymaga niezależnych układów sterujących pracą bloku PASSU. Natomiast przypadki B i C są stosunkowo tanie w realizacji (szczególnie w przypadku pamięci ferrytowych). Przypadek C można uzyskać tylko środkami logicznymi. Nie zachodzi zatem konieczność opracowania specjalnego bloku lub specjalnej części "nadmiarowej" pamięci operacyjnej. W przypadku C przyjmuje się na ogół, że ostatni obszar (o pojemności wystarczającej dla przechowywania SSU) aktualnie zainstalowanej pamięci operacyjnej jest pamięcią PASSU. Odpowiednie środki logiczne organizują dostęp do właściwej PAO w czasie pracy programu lub do obszaru PASSU w czasie pracy kanałów.

Takie rozwiązanie (tj. przypadek C) zostało na przykład przyjęte w polskiej maszynie jednolitego systemu R-32 [1], [2].

Niemniej jednak takie rozwiązanie nie jest wystarczające w przypadku systemów dwuprocessorowych (rys. 2), w których wymagany jest wysoki stopień niezawodności. W szczególności w takich systemach, w których uszkodzenie dowolnego bloku PAO (a więc i bloku zawierającego PASSU) nie może spowodować przerwy w kontynuowaniu obliczeń określonych ważnych programów. Często takie programy są jednocześnie realizowane w niezależnych blokach PAO, procesorach i kanałach (wspólnie z innymi programami w systemie wieloprogramowym).

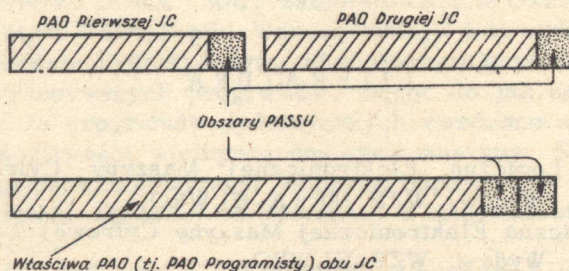


Rys. 2. Przykład konfiguracji dwuprocessorowej, którą można zrealizować w Jednolitym Systemie EMC

W niniejszym opracowaniu zostanie przedstawione rozwiązanie organizacji logicznej pamięci PASSU w dwuprocessorowych systemach Jednolitego Systemu EMC, które to rozwiązanie spełnia warunki niezawodności, jednolitości adresowania pamięci operacyjnej oraz oszczędności sprzętu.

2. OPIS KONCEPCJI

Proponowane rozwiązanie zakłada, że w każdym procesorze systemu dwuprocessorowego wprowadza się specjalne rejestry; zwane RAPO i RAP1, w których zapisuje się odpowiednio początkowe adresy absolutne obszarów PASSU w pamięci operacyjnej (z dokładnością do numeru bloku 16 K-bajtowego). Adresy te ustawia się na podstawie informacji (ustawionych na pulpicie konfiguracji systemu) opisujących aktualną konfigurację pamięci operacyjnej na poszczególnych jednostkach centralnych (JC). Zakłada się dalej, że obszar PASSU (o pojemności 16 Kb) danej JC zajmuje najstarsze adresy jej pamięci operacyjnej (rys. 3). W ten sposób obszary PASSU nie znajdują się na końcu łącznej pamięci operacyjnej danego systemu dwuprocessorowego, lecz na końcu "własnej" pamięci operacyjnej. Przy czym pamięci operacyjne poszczególnych JC mogą być niesymetryczne, tzn. mogą mieć różne pojemności.



Rys. 3. Fizyczne rozmieszczenie obszarów PASSU w pamięci operacyjnej (rys. górny) oraz rozmieszczenie tych obszarów z punktu widzenia programisty

W przypadku systemów jednoprocessorowych - gdzie każda PAO ma własną adresację - organizacja odwoływań do obszaru PASSU jest prosta i polega na powiększeniu adresu podkanału (lub jego krotności) o adres zawarty w rejestrze RAPO. Natomiast adresy absolutne odwoływań do obszarów właściwej PAO powinny być mniejsze od adresu zapisanego w PAPO (w tym przypadku rejestr RAP1 nie jest wykorzystany).

Organizacja odwoływań do PAO w przypadku systemów dwuprocessorowych jest trudniejsza i może być zrealizowana w sposób następujący:

1. Zakłada się w tych systemach, że łączna pamięć operacyjna systemu dwuprocessorowego ma jednolitą adresację fizyczną (lub absolutną) zadaną na pulpicie konfiguracji. Adresy te oznaczać będziemy przez F . Z punktu widzenia adresacji fizycznej jeden z obszarów PASSU znajduje się na końcu łącznej PAO, zaś drugi na końcu PAO pierwszej JC (rys. 3). Zakłada się również, że łączna pamięć operacyjna systemu dwuprocessorowego posiada jednolitą i ciągłą adresację rzeczywistą (tzn. z punktu widzenia programisty systemowego). Adresy rzeczywiste oznaczymy przez R .

2. Przy odwoływaniach dowolnego procesora do wspólnej pamięci operacyjnej przyjmuje się, że:

Adres $F := R$, gdy R jest mniejsze od zawartości rejestru RAPO. Jeśli dane odwołanie miało miejsce w pierwszej JC, to zachodzi zwrot do własnej pamięci operacyjnej. Jeśli zaś w drugiej JC, to zachodzi zwrot do pamięci operacyjnej pierwszej JC (czyli zwrot krzyżowy),

. Adres $F := R + 2^{14}$, gdy R jest równe lub większe od zawartości rejestru RAPO.

3. Przy odwołaniach do pamięci operacyjnej dowolnej JC, adres rzeczowy R powinien spełniać warunek:

$$R < /RAP1/ - 2^{14}.$$

4. Odwołania kanałów danej JC są realizowane względem własnego obszaru PASSU. Oznacza to, że odwołania kanałów pierwszej JC odbywają się względem adresu bazowego zapisanego w RAPO, zaś drugiej JC względem adresu bazowego rozpisanego w RAP1.

Opisane wyżej rozwiązanie organizacji pamięci PASSU zostało przyjęte do realizacji w nowo opracowywanej polskiej EMC Jednolitego Systemu, w której początkowo przewidywano odrębny fizyczny blok PASSU (16 Kb) oprócz ferrytowych bloków pamięci operacyjnej (po 512 Kb).

LITERATURA

- [1] Architektura Logiczna Elektronicznej Maszyny Cyfrowej R-32, AL-2045901-2. Wydaw. WZE MERA-ELWRO.
- [2] Struktura Logiczna Elektronicznej Maszyny Cyfrowej R-32, OL-2045901-2. Wydaw. WZE-ELWRO.

SPIS TREŚCI

Wiesława Pulczyn, Janusz Sochacki - 3SM System symulacji maszyn mikroprogramowanych	3
Przemysław Bąkowski - Opis i symulacja makroprocesorów przy wykorzystaniu języka OSM i systemu 3SM	11
Jan Popiel - Symulacja maszyn cyfrowych w systemie 3SM	17
Henryk Krawczyk - Pewne problemy mikrodiagnostyki	25
B. Courtois, G. Saucier - Designing a fault-tolerant computer: its security, reliability, performance and cost	37
Ch. Robach, G. Saucier - Processor testability and design consequences	49
Marian Budka - Algorytm generowania łatwo diagnozowalnych modułowych struktur układów cyfrowych	69
Piotr Godlewski, Jan Klimowicz, Jerzy Andrzej Szaniawski - Program DALGO Implementacja algorytmów generacji i weryfikacji testów DALG-II i TEST-DETECT w języku Assemblera IBM/370	81
Witold Komorowski - Dydaktyczny model komputera	89
Thanasis Kamburelis - O organizacji pamięci kanałów wejścia-wyjścia w dwuprocessorowych systemach JS EMC	95
Włodzimierz Zuberek - Wykorzystanie pamięci bębnowej do buforowania urządzeń znakowych w maszynach ODRA-1300	99
Andrzej Pająk - Optymalizacja rozmieszczania elementów scalonych na płytach montażowych	107
Jacek Majewski - Zastosowanie metody podziałów i ograniczeń dla rozwiązania zagadnienia rozmieszczania	121
Marian Bogdan - Wyznaczanie fragmentów książki sygnału przy wspomaganym projektowaniu płytek drukowanych	129
Władysław M. Turski - Changing nature of software problems	145
Jan Kazimierzczak - Koncepcja automatu parametrycznego i jego zastosowanie w strukturze systemu cyfrowego	157
Jerzy Mysior, Jacek Olszewski - Interpretative description of computers and operating systems	169
Jan Bielecki, Waław Iszkowski, Marek Maniecki, Marek Suchanek, Aleksander Wigura - Implementacja supervisora o wielopoziomowej asynchronicznej obsłudze przerw	181
Andrzej Pająk, Grzegorz Prochowski - Implementacja oprogramowania wielostanowiskowego systemu rejestracji i wstępnego przetwarzania danych - KR TM-20	197
Piotr Misiurewicz - Egzekutor dla minikomputera MERA 300	209
Helena Dryzek - Zastosowanie markowskich modeli procesu przetwarzania	215
Andrzej Stokalski - optymalizacja sterowań markowskich modeli w systemach operacyjnych	233
Krzysztof Sacha - Matematyczna analiza systemu wieloprogramowego	251
Andrzej Dzierżykraj - Metoda analitycznego modelowania procesów dyskretnych	257