

Aleksander GÓRNY

SYSTEM OPERACYJNY SORT

Streszczenie. W opracowaniu tym został omówiony zaprojektowany i zrealizowany system operacyjny "SORT", który jest przeznaczony dla komputerowego systemu Odra 1003S. Daje on możliwości laboratoryjnego sterowania modelami procesów w czasie rzeczywistym.

Pierwsza część opracowania przedstawia możliwe istniejące konfiguracje występujące w "SORT" w zależności od tego jakiego języka użyto do obliczeń.

W drugiej części omówiono podstawowe części, które składają się na system operacyjny "SORT".

Prezentowany system operacyjny SORT został zaprojektowany i zrealizowany w Instytucie Kompleksowych Systemów Sterowania Politechniki Śląskiej.

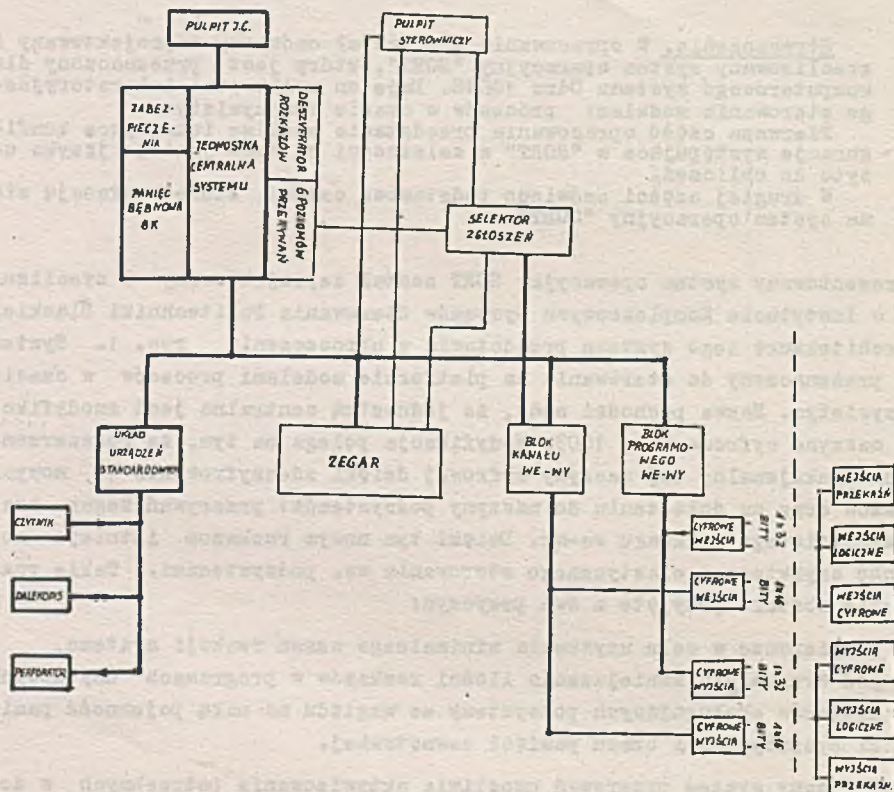
Architekturę tego systemu przedstawia w uproszczeniu rys. 1. System jest przeznaczony do sterowania na platformie modelami procesów w czasie rzeczywistym. Nazwa pochodzi stąd, że jednostką centralną jest zmodyfikowana maszyna cyfrowa Odra 1003. Modyfikacja polega na tym, że rozszerzono zakres funkcjonalny tej maszyny cyfrowej dzięki zdeszyfrowania 19 nowych rozkazów oraz na dołączeniu do maszyny podsystemów: przerywań, zegara czasu rzeczywistego i kanału we-wy. Dzięki tym nowym rozkazom istnieje możliwość szybkiego i elastycznego sterowania ww. podsystemami. Takie rozwiązanie zostało przyjęte z dwu przyczyn:

- po pierwsze w celu uzyskania minimalnego czasu reakcji systemu,
- pod drugie dla zmniejszenia ilości rozkazów w programach użytkowych systemów obsługujących podsystemy ze względu na małą pojemność pamięci operacyjnej i braku pamięci zewnętrznej.

Opracowany system przerywań umożliwia aktywizowanie potrzebnych w dowolnym momencie czasu programów użytkowych, zastosowanych w systemie. Każde zgłoszenie wyższego poziomu przerywa aktualnie wykonywany program w taki sposób, że istnieje możliwość odtworzenia go od momentu przerwania i wywołuje program poziomu zgłoszonego. Z pulpitu sterowniczego istnieje możliwość zgłoszenia dowolnego poziomu np. poziomu realizującego polecenia operatora systemu ingerującego w pracę systemu. Najwyższy poziom jest zarezerwowany dla zgłoszeń sytuacji alarmowych, wynikłych z błędnej pracy programów użytkowych (Nz) albo dla zgłoszeń o uszkodzeniu części systemu. Pozostałe poziomy mogą zostać przydzielone kanałowi we-wy i zegarowi czasu rzeczywistego, umożliwiając tym samym odmierzenie odcinków czasu rzeczywistego i wczytanie lub wyprowadzenie danych z urządzeń kanału we-wy

kiedy wymaga tego stan procesu. (Istnieje zawsze możliwość programowego wczytania danych z kanału we-wy do jednostki centralnej).

Kanał we-wy jest tak wykonany, że umożliwia przyjmowanie informacji cyfrowej, licznikowej oraz logicznej. Do kanału tego przy minimalnych zmianach można podłączyć kilka różnych urządzeń konwencjonalnych czy przemysłowych posiadających jednak układy sterowania mogące deszyfrować ich własne adresy oraz wysyłać sygnały gotowości czy potwierdzenia przyjęcia informacji.



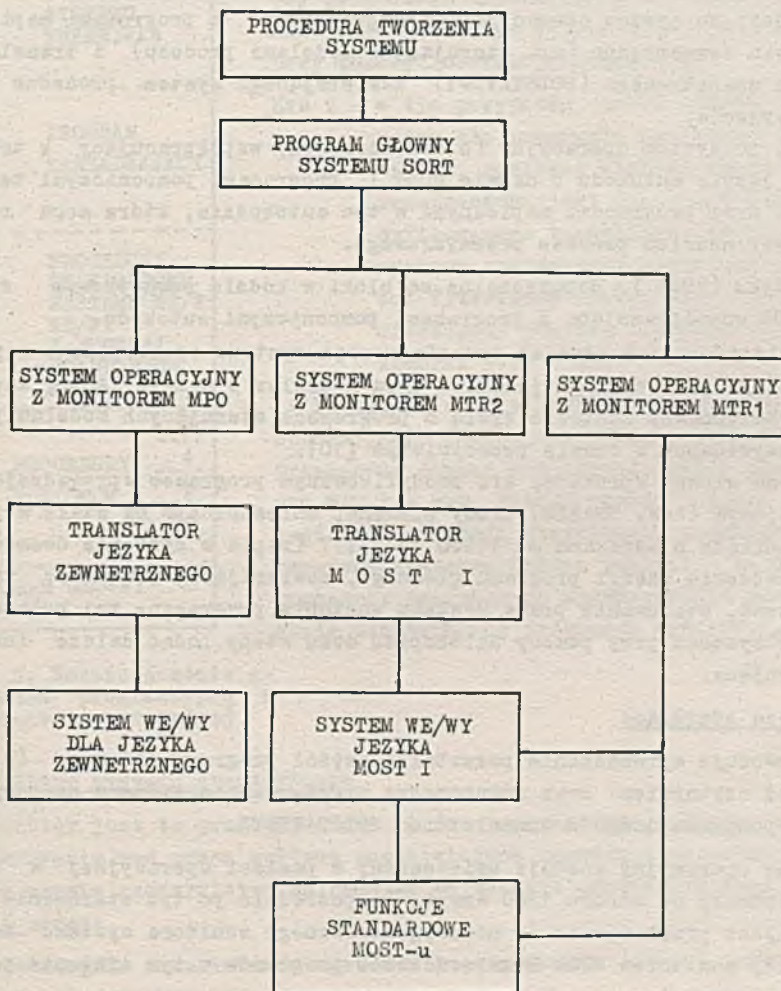
Rys. 1. Schemat blokowy systemu ODRA 1003 S

1. Zadania systemu operacyjnego SORT

Systemy operacyjne są jednymi z najbardziej istotnych składowych w oprogramowaniu maszyn sterujących. System operacyjny SORT zaprojektowany został dla laboratoryjnego sterowania modelami procesów przemysłowych w czasie rzeczywistym przy pomocy systemem sterowania Odra 1003s. Jest on rozszerzeniem idei monitorów, tj. programów przeznaczonych do nadzorowania takich operacji maszyny jak sterowanie programów wynikowych czy systemów programowania. Można więc powiedzieć, że monitor jest częścią systemu

operacyjnego, w którego skład wchodzi oprócz tego translatory języków, procedury biblioteczne i ewentualnie inne monitory.

Nazwa monitor nie jest jedyną przyjętą nazwą dla tego rodzaju programów; jako synonimy można m.in. wymienić takie nazwy jak: program zarządzający, program nadzorczy, dyrygent, (z ang. supervisory program, executive routines, operating systems, director). W dalszej części artykułu zostaną podane najbardziej charakterystyczne własności systemu operacyjnego SORT. Składa się on z procedury tworzenia systemu, monitora systemu, programu zarządzającego systemem przetwarzania uwarunkowanego czasowo, systemu wejścia/wyjścia, translatorów języka MOST I i języka zewnętrznego o nazwie PODSTAWOWY (p. rys. 2 i rys. 3). Przeznaczenie poszczególnych części sy-



Rys. 2. Schemat systemu SORT

stemu operacyjnego najlepiej ilustruje sposób ich wykorzystania. Omawiając bardzo skrótowo kolejno jego części będziemy wzorować się, o ile to będzie możliwe, na przyjętym sposobie przedstawienia systemów programowania maszyny cyfrowej Odra 1003.

2. Procedura tworzenia systemu

Program główny który zostaje dostarczony użytkownikowi systemu Odra 1003S zawiera wszystkie części składowe systemu operacyjnego SORT. Wykorzystując procedurę tworzenia systemu użytkownik ma możliwość wygenerowania z tego głównego programu jednego z dwu możliwych (na razie) systemów operacyjnych współpracujących z danym językiem programowania.

- pierwszy, to system operacyjny współpracujący z programami napisanymi w języku zewnętrznym (np. sterującymi modelami procesu) i translatorem języka zewnętrznego (PODSTAWOWY) zawierającego system procedur wejścia/wyjścia,
- drugi, to system operacyjny (o dwu wersjach) współpracujący z translatorem języka autokodu o nazwie MOST I, programami pomocniczymi tego autokodu oraz programami napisanymi w tym autokodzie, które mogą również sterować modelem procesu przemysłowego.

W języku (MOST I) dopuszczalne są bloki w kodzie zewnętrznym systemu Odra 1003 współpracujące z programami pomocniczymi autokodu.

Przewidziana jest również trzecia wersja systemu operacyjnego współpracująca z specjalistycznym językiem typu assembler o nazwie JAS-S, który to język zrealizowany został z myślą o programach sterujących modelami procesów przemysłowych w czasie rzeczywistym [10].

Program główny wprowadza się zmodyfikowanym programem wprowadzająco-wyprowadzającym (tzw. STAŁYM) taśmy binarne, umieszczonym na stałe w pamięci w komórkach o adresach od 17600 do 17777 (zapis w systemie ósemkowym). Po wprowadzeniu części programu głównego, zawierającej procedurę tworzenia systemu, sterowanie pracą systemu zostanie przekazane tej procedurze. Operator systemu przy pomocy dalekopisu może wtedy podać dalsze instrukcje sterujące.

Instrukcja sterująca

MPO - powoduje wprowadzenie pozostałej części programu głównego (z taśmy pod czytnikiem) oraz równoczesne wygenerowanie systemu operacyjnego współpracującego z translatorem PODSTAWOWYM.

System operacyjny zostaje umieszczony w pamięci operacyjnej w komórkach począwszy od adresu 1200 wzwyż. Bezpośrednio po tym sterowanie pracą systemu jest przekazywane do nowowygenerowanego monitora systemu nazywanego dalej monitorem MPO. Rozmieszczenie programów w tym systemie pokazane jest na rys. 4a.

Instrukcja sterująca

MTR1 - wprowadza również pozostałą część programu głównego, generując zarazem system operacyjny współpracujący z programami pomocniczymi języka MOST I oraz programami napisanymi w tym autokodzie, z możliwością dołączenia bloków programów napisanych w języku zewnętrznym wg reguł określonych w instrukcji [8]. System zostaje umieszczony w pamięci operacyjnej od komórki 17477 w dół.

Rys. 4b przedstawia podział pamięci w przypadku powyższego systemu operacyjnego. Wytworzenie systemu operacyjnego, który nie tylko będzie współpracował z programami pomocniczymi autokodu czy programami napisanymi w tym języku ale również z translatorem języka MOST I dokonuje się przy pomocy instrukcji sterującej.

MTR 2 - w tym przypadku obszar pamięci przeznaczony dla programów wynikowych jest niewielki, ponieważ większa część pamięci przeznaczona jest dla zapamiętania zmodyfikowanego translatora autokodu oraz systemu operacyjnego. Pokazano to na rys. 4c. Wytworzony system operacyjny jest umieszczony poniżej translatora, tj. od komórki 5777 w dół.

Po wygenerowaniu jakiegokolwiek wersji systemu operacyjnego sterowanie pracą systemu zostaje przekazane danemu monitorowi, który posiada nazwę MPO lub MTR. Dalszy rozwój pracy zależny jest od instrukcji sterujących podanych przez operatora systemu. Obszar pamięci zajęty przez procedury tworzące po wygenerowaniu monitora zostaje udostępniony programom wynikowym.

M O N I T O R	MONITOR PO- STAWOWY WRAZ Z REZYDENTEM SYSTEMU PRZERYWAN	
	PROGRAM ZARZĄDZAJACY	
R E Z Y D E N T	PROCEDURY STANDARDOWE SYSTEMU WE/W WE/WY I FUNKCJI STANDARDO- WYCH	
		5
	PROCEDURY	4
	POZIOMOW	3
		2
		1
	PROGRAM GŁÓWNY	0

Rys. 3. Rozmieszczenie elementów programowych w pamięci operacyjnej

3. Monitor systemu operacyjnego

Monitor jest to program, który oprócz podstawowego zadania jakim jest nadzorowanie nad pracą systemu umożliwiającą kontakt z procesem sterowanym w czasie rzeczywistym ma również za zadanie ułatwienia jego eksploatacji przez:

- wprowadzanie translatorów systemów programowania i (lub) programów wynikowych do wykonania,
- wprowadzanie programów bibliotecznych,

- sterowanie systemem wejść-wyjść,
- zapewnienie łączności operatora z systemem,
- sterowanie programami uwarunkowanymi czasowo.

Składa się z trzech części: monitora rezydenta, monitora zmian i monitora przekształceń.

Monitor rezydent zawsze znajduje się w pamięci jednostki centralnej systemu w trakcie jego pracy. Jego zadanie to wprowadzanie zadań do pamięci w celu ich wykonania, sterowanie systemem wejść-wyjść oraz wprowadzenia monitora przekształceń, gdy programy zmiennomiejscowe wymagają przekształceń ich na postać o adresacji bezwzględnej.

Monitor zmian może zajmować obszar pamięci obok obszaru monitora rezydenta. Jest on wywoływany wówczas, gdy występuje potrzeba zmian wywołanych wprowadzeniem nowych zadań (np. dołączenie wspólnych procedur bibliotecznych dla programów znajdujących się na różnych poziomach przerwania).

a)

00000	01200	15570	17777
TRANSLATOR JEZYKA ZEWNETRZNEGO SYSTEM WE/WY	SYSTEM OPER. Z MONITOREM MPO	PROCEDURY POZIOMOW	STAŻY

b)

00000	15500	17570	17777
PROCEDURY POZIOMOW	SYSTEM OPER. Z MONITOREM MTR1	SYSTEM WE/WY I FUNKCJE STANDARDOWE Jś MOST I	STAŻY

c)

00000	06000	15500	17570	17777
PROCEDURY POZIOMOW	SYSTEM OPER. Z MONITOREM MTR2	TRANSLATOR JEZYKA MOST I	SYSTEM WE/WY I FUNKCJE STANDARDOWE	STAŻY

Rys. 4. Podział pamięci dla różnych monitorów

W przypadku systemu operacyjnego SORT monitor rezydent i monitor zmian mogą znajdować się w czasie pracy systemu równocześnie w pamięci operacyjnej. Istnieje jednak możliwość pracy bez monitora zmian tylko z monitorem rezydentem. Powoduje to z jednej strony zwiększenie pojemności pamięci dla programów wynikowych oraz zmniejsza czas reakcji systemu. Z drugiej strony ogranicza jednak możliwości operatora systemu. Opis zakresu działania monitora można najlepiej zrozumieć przez zapoznanie się z listą instrukcji sterujących - systemu SORT 12.

Przy realizacji programu monitora skorzystano z istniejących procedur - programów bibliotecznych maszyny cyfrowej Odra 1003, które zostały zmodyfikowane przed ich wykorzystaniem. Są to programy 03-IV-35, 03-IV-1, 03-IV-2, 03-IV-141 oraz programy pomocnicze i translator języka MOST I.

Każdy rodzaj błędu wykryty przez system lub procedury współpracujące z systemem powodują odwołanie się do monitora wywołującego procedurę błędów, drukującą numer błędu oraz podejmującą decyzję co do dalszego sterowania.

4. Program zarządzający przetwarzaniem uwarunkowanym czasowo

Procedurami uwarunkowanymi czasowo nazywamy procedury przeznaczone do przetwarzania, wywoływane na żądanie informacji przychodzącej ze specjalnych urządzeń wejść-wyjść poprzez układ przerywań.

Dotychczas w większości maszyn cyfrowych stosuje się tzw. przetwarzanie wsadowe nazwane także przetwarzaniem seryjnym. Polega ona na tym, że do maszyny cyfrowej wprowadza się program wynikowy i wykonuje się go dopóty, dopóki nie wyczerpią się jego dane, po czym wprowadza się następny program.

Procedury uwarunkowane czasowo różnią się od takiego rodzaju przetwarzania tym, że:

1. W programie głównym (najczęściej znajdującym się na poziomie zerowym) nie przewidziano kiedy nastąpi zgłoszenie żądania przetwarzania uwarunkowanego czasowo, ponieważ zgłoszenie żądania przychodzi od urządzenia (z procesu), a nie od programu.
2. Zgłoszenia żądań przetwarzania muszą być realizowane na bieżąco, a więc odpowiednie czynności należy wykonywać wtedy, kiedy nadejdą zgłoszenia.

Należy tu uwzględnić fakt, że zgłoszenia mogą nadchodzić nieprzerwanym strumieniem, niezależnie które operacje aktualnie wykonuje program główny albo procedura uwarunkowana czasowo, będąca na niższym poziomie od zgłoszonego.

Wszystkie zgłoszenia (żądania) od procesu czy operatora lub urządzeń we-wy zostały w systemie Odra 1003S podzielone na sześć poziomów głównych, których selekcja jest urządzeniowa, tzn. jeżeli pojawi się żądanie na poziomie wyższym od aktualnie wykonywanego to następuje natychmiastowe przerywanie i obsłużenie żądania.

Jeżeli jednak żądanie jest na niższym poziomie od wykonywanego to żądanie zostanie zapamiętane w celu późniejszego jego obsłużenia.

Każdy z poziomów głównych może posiadać kilka podpoziomów zgłoszeń, które rozpoznawane i selekcjonowane są już na danym poziomie programowo, np. na podstawie parametrów procesu rejestrowanych przez cyfrowym kanał we-wy (patrz rys. 1).

System operacyjny SORT ma zawsze w swym wyposażeniu program zarządzający systemem przetwarzania uwarunkowanego czasowo, który jest traktowany, jak pokazano na rys. 3, jako część monitora rezydenta.

Omówimy obecnie w skrócie sposób powiązania tego programu z monitorem rezydenta oraz z procedurami przetwarzania uwarunkowanego czasowo, które są przez niego obsługiwane.

Działanie programu zarządzającego systemem przetwarzania uwarunkowanego czasowo zapoczątkowuje monitor rezydent systemu przerywać. Przekazuje on sterowanie programowi zarządzającemu, jeśli po przeanalizowaniu przerywania programu z zewnątrz stwierdzi, że jest wymagana operacja przetwarzania uwarunkowanego czasowo. Po zakończeniu procedury wywołanej zgłoszeniem, sterowanie wraca do monitora rezydenta systemu przerywać, który przywraca stan istniejący w chwili przerywania programu i odwołuje się do niego.

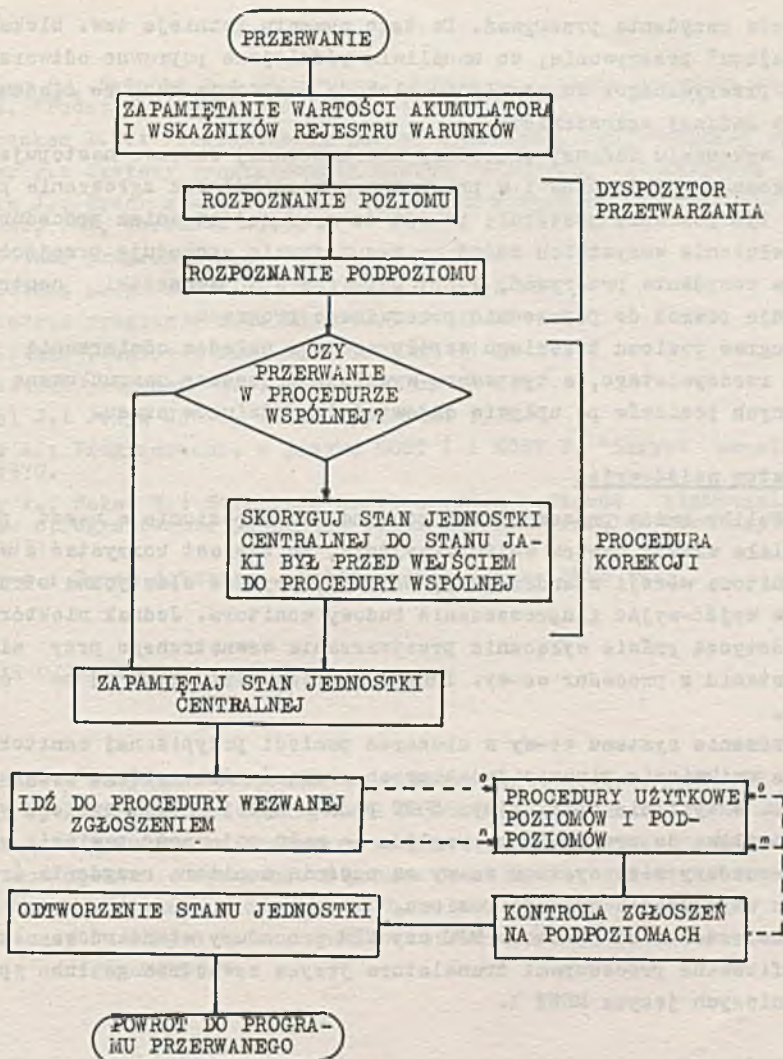
Do zadań programu zarządzającego należy więc:

1. Rozpoczęcie wykonywania procedury uwarunkowanej czasowo, gdy zostanie ona dostępna.
2. Zidentyfikowanie poziomu zgłoszonego oraz wykonanie operacji na kanale we-wy w celu określenia, która procedura na danym poziomie jest żądana (identyfikacja podpoziomu).
3. Analizowanie błędów wykrytych przez procedury uwarunkowane czasowo i podjęcie właściwego działania.

Rysunek 5 przedstawia bardzo poglądowo sieć działań realizacji współdziałania procedur uwarunkowanych czasowo, znajdujących się na różnych poziomach. Ponieważ realizacja procedury przerywania zmienia stan warunków i wartość akumulatora programu przerywanego to z chwilą, gdy wystąpi przerywanie (zgłasza się poziom wyższy od wykonywanego) zostanie w pierwszym etapie zapamiętany stan wskaźników rejestru warunków i stan rejestru akumulatora. Po tych czynnościach program o nazwie dyspozytor przerywać rozpoznaje poziom i o ile potrzeba również podpoziom z powodu którego nastąpiło przerywanie.

System operacyjny SORT zezwala na używanie tych samych standardowych procedur we-wy przez procedury uwarunkowane czasowo i znajdujące się na różnych poziomach.

Na używanie wspólnych funkcji standardowych w języku MOST I zezwala również system SORT ale tylko w przypadku współpracy z monitorem MTR (po modyfikacji również z monitorem MPO). Powyższe własności komplikują program zarządzający, który musi posiadać dodatkowo procedury korekcji. Przez nią



Rys. 5. Schemat systemu obsługi zgłoszenia

następuje komunikacja procedur użytkowych (sterujących) z poszczególnymi poziomami z systemem we-wy i systemem funkcji standardowych. W przypadku przerwania (rys. 5) procedura korekcji decyduje czy stan jednostki centralnej należy potraktować jako stan dla przyszłego odtworzenia czy skorygować go do stanu jednostki centralnej przed wejściem do standardowej procedury, wspólnej dla procedur poziomów. Zależy to od tego, czy w momencie przerywania była wykonywana procedura wspólna czy nie. Skorygowany stan jednostki centralnej zostaje zapamiętany w odpowiednim obszarze

monitora rezydenta przerywać. Do tego momentu istnieje tzw. blokada "zastaniająca" przerywania, co umożliwia późniejsze poprawne odtworzenie programu przerywanego. Po usunięciu blokady następuje skok ze śladem do procedury żądanej zgłoszeniem.

Po wykonaniu żądanej procedury uwarunkowanej czasowo następuje kontrola zgłoszeń podpoziomów i w przypadku jakiegokolwiek zgłoszenia podpoziomu na tym poziomie następuje powrót do wybranej żądaniem procedury.

Obsłużenie wszystkich żądań na tym poziomie spowoduje przejście do monitora rezydenta przerywać, który odtwarza stan jednostki centralnej i powoduje powrót do poprzednio przerwanej procedury.

Program poziomu trzeciego współpracuje z układem odmierzania odcinków czasu rzeczywistego, a tym samym wywołuje procedury uwarunkowane czasowo z różnych poziomów po upływie odpowiednich odcinków czasu.

5. System wejść-wyjść

Jeżeliby każda procedura wynikowa na danym poziomie a nawet podpoziomie miała własny system wejścia-wyjścia, to zamiast korzystać z włączonej do monitora wersji standardowej, można by uzyskać elastyczną strukturę systemów wejść-wyjść i uproszczenie budowy monitora. Jednak niektóre procedury dotyczą prawie wyłącznie przetwarzania wewnętrznego przy minimalnym korzystaniu z procedur we-wy. Inne natomiast nastawione są na czynności we-wy.

Umieszczenie systemu we-wy w obszarze pamięci przypisanej monitorowi powoduje uniknięcie pisania dodatkowych rozkazów dla każdego programu wynikowego. W systemie operacyjnym SORT podana wyżej wersja systemu we-wy nie jest możliwa do przyjęcia ze względu na małą pojemność pamięci wynoszącą 8K. Procedury więc systemu we-wy są częścią monitora rezydenta (rys. 3) i są one wspólne dla każdego poziomu.

W zależności od monitora MPO czy MTR procedury standardowe we-wy są zmodyfikowane procedurami tłumacza języka zewnętrznego lub programów pomocniczych języka MOST I.

6. Zakończenie

Przedstawiony w niniejszym artykule system operacyjny SORT został stworzony dla Systemu Odra 1003S, który bazuje na maszynie cyfrowej Odra 1003. System ten został wykorzystany przy badaniu na platformie modeli takich procesów jak: proces transportowo-wydobywczy na kopalni oraz proces sekwencyjny na przykładzie rozruchu turbogeneratorskiego.

LITERATURA

- [1] Węgrzyn S., Nałęczki K.: Struktury, algorytmy, języki systemów sterowania, "Podstawy sterowania", PWN, 1971.
- [2] Mc Cracken D.D.: Programowanie maszyn cyfrowych, PWN, Warszawa 1962.
- [3] Fisher P.: Systemy programowania maszyn cyfrowych, WNT, Warszawa 1971.
- [4] Martin J.: Programowanie maszyn cyfrowych w systemach uwarunkowanych czasowo, WNT, Warszawa 1970.
- [5] Biblioteka programów m.c. Odra 1003 nr O3-IV-1
- [6] Biblioteka programów m.c. Odra 1003 nr O3-IV-2
- [7] Biblioteka programów m.c. Odra 1003 nr O3-IV-1
- [8] Biblioteka programów m.c. Odra 1003 nr O3-IV-2
- [9] Biblioteka programów m.c. Odra 1003 nr O3-V-1
- [10] Hendel J.: Język JAS-S, Praca dyplomowa 1972.
- [11] Górny A.: Programowanie w języku MOST I i MOST P, "Skrypt uczelniany" 1970.
- [12] Górny A., Wołek S.: Struktura kompleksowych systemów sterowania i teoria oprogramowania jednostek centralnych, "Przegląd Górniczy" nr 7-8/1971.
- [13] Górny A.: Praca doktorska, Politechnika Śląska IKSS 1973.

ОПЕРАЦИОННАЯ СИСТЕМА SORT

Резюме

В этой работе обсуждено операционную систему SORT, проект которой уже реализован, предназначен для электронной вычислительной системы ODRA 1003S. Эта система дает возможность управлять на платформе моделями процессов в действительном времени. Первая часть работы представляет существенные конфигурации, которые выступают в системе SORT, в зависимости от языка на каком идут вычисления. Во второй части, описаны основные части, входящие в состав операционной системы SORT.

OPERATIONAL SYSTEM "SORT"

Summary

In the paper the Operational System "SORT" intended for computer system "ODRA 1003 S" has been discussed designed and realized. It makes possible the laboratory model control in the real time. The first part of this paper presents the possible configurations existing in "SORT", depending upon language used for calculations. The second part deals with fundamental parts of the Operational "SORT" System.