

BIULETYN. TECHNICZNY

7.2900/78



M

E

T

A

8(198)
1978

Redakcja Kolegium w składzie:

mgr Z. Bieguszevska-Kochan, mgr W. Borucki (redaktor działu „Ekonomika”),
mgr B. Drożak, mgr inż. J. Dziewięcki (redaktor naczelny), J. Esikowski,
mgr inż. R. Farfał, dr hab. M. Greniewski,
doc. dr hab. inż. A. Janicki (redaktor naukowy), inż. L. Kowalski,
mgr J. Kutrowska (sekretarz redakcji), mgr inż. L. Krzystalik, inż. R. Maciesowicz,
mgr E. Mańkiewicz-Cudny, red. T. Podwysocki, dr inż. R. Pregiel,
mgr inż. A. Teodorczuk, mgr inż. T. Ustaborowicz,
mgr inż. M. Wajcen (redaktor działu „Technika”)

Warunki prenumeraty

Jednostki gospodarki uspołecznionej, instytucje, organizacje i wszelkiego rodzaju zakłady pracy zamawiają prenumeratę w miejscowych Oddziałach RSW „Prasa-Książka-Ruch”, w miejscowościach zaś, w których nie ma Oddziałów RSW – w urzędach pocztowych. Czytelnicy indywidualni opłacają prenumeratę wyłącznie w urzędach pocztowych i u doręczycieli. Prenumeratę roczną w cenie 516 zł należy zamawiać do 25 listopada na rok następnny, półroczną do 10 czerwca na II półrocze.

ZJEDNOCZENIE PRZEMYSŁU AUTOMATYKI I APARATURY POMIAROWEJ „MERA”



„MERA”

BIULETYN PRZEMYSŁU
KOMPUTEROWYCH SYSTEMÓW
AUTOMATYZACJI I POMIARÓW

WARSZAWA, SIERPIEŃ 1978

SPIS TREŚCI

L. Krzystolik	Rozwój systemów automatyki pod wpływem techniki mikroprocesorów	3
J. Łączyński	Możliwości zastosowania mikroprocesorów w elektronicznej aparaturze kontrolno-pomiarowej	6
K. Rzymkowski P. Witort	Zastosowanie mikroprocesorów w blokach sterujących systemu CAMAC	11
J. Szyller B. Krzesaj-Janyszek	Koncepcja uniwersalnego, wielozadaniowego mikroprocesorowego systemu operacyjnego	15
B. Jackiewicz	Współpraca przetworników analogowo-cyfrowych z systemami mikroprocesorowymi	19
B. Krzesaj-Janyszek J. Szyller	Wybranie zagadnienia ochrony pamięci w wielozadaniowych systemach mikroprocesorowych	24
J. Kołodziejski	Pomiary i badania układów mikroprocesorowych	27
J. Kern K. Wildt	Mikroprocesorowy kontroler testera układów scalonych	31
H. Więcek S. Duraj	System diagnostyki uszkodzeń w walcowni WCB 2000 Huty Katowice	34
P. Ruszkarski J. Witewski	Budowa i możliwości zastosowań prostych układów kalkulatorowych	36
J. Kiszkurno	Zastosowanie programowanego kalkulatora do wyznaczania i numerycznej analizy statycznych charakterystyk elementów i struktur półprzewodnikowych	43

Opracowanie redakcyjne: Redakcja Biuletynu "Mera", ul. Patriotów 77, 04-950 Warszawa /tel. 12-41-71/. Druk: Dział Wydawnictw "Mera-Pnefal", ul. Patriotów 77, 04-950 Warszawa /tel. 12-41-64/. Zam. 158/78. Nakład 2300 egz.

mgr inż. LUDOMIR KRZYSTOLIK
Zjednoczenie „Mera”

ROZWÓJ SYSTEMÓW AUTOMATYKI POD WPŁYWEM TECHNIKI MIKROPROCESORÓW

Najistotniejszym wynikiem wprowadzenia techniki mikroprocesorów lub rozpatrując szerzej to zagadnienie - techniki obwodów makroskalonych, jest osiągnięcie jakościowego skoku w kształtowaniu stosunku kosztu do wartości użytkowej urządzeń elektroniki. Fakt ten potwierdza to, że kompletny, działający system mikrokomputerowy można dziś uzyskać kosztem około 3000 zł. dewizowych. Pojawienie się tanich elementów makroskalonych realizujących złożone funkcje implikuje głębokie zmiany w strukturze układów automatyki i pomiarów otwierając nowe pola ich zastosowań. Celem niniejszego opracowania jest przegląd głównych kierunków tych zmian i wpływu techniki mikroprocesorów na rozwiązania typowych problemów w dziedzinie automatyki i pomiarów.

Układy centralnej rejestracji i sterowania złożonych procesorów technologicznych

Klasyczna struktura takich układów przewiduje wyposażenie procesu w podsystem regulacji analogowej zrealizowany w technice elektronicznej lub elektropneumatycznej, wyspecjalizowane układy zabezpieczeń, blokad, sygnalizacji, wyspecjalizowane automaty sekwencyjne i scentralizowany układ informacyjny zrealizowany przy pomocy rozwiniętego systemu minikomputerowego, któremu jednocześnie powierza się funkcje naurzędzonego sterowania i kontroli wyżej wymienionych podsystemów. W układzie informacyjnym, w olbrzymiej większości dotychczasowych realizacji praktycznych, minikomputer rezerwowany jest tradycyjnymi, analogowymi środkami przekazywania informacji o procesie.

Tak opisany typowy hierarchiczny system CRIS zawiera nadmiary sprzętowe, gdyż z technicznego punktu widzenia istnieje możliwość zastąpienia przez minikomputer funkcji realizowanych przez centralną część podsystemu analogowego, podsystem sygnalizacji, automaty sekwencyjne itd. i zorganizowania silnie scentralizowanego systemu o czystej technice "oszczędnej" strukturze. O niepowodzeniu sys-

temów scentralizowanych zadecydowały względy niezawodnościowe.

Rozpatrywane systemy stosowane są do sterowania i kontroli wielkich obiektów technologicznych /bloki energetyczne, duże instalacje w przemyśle rafineryjnym i petrochemicznym itp. /, których wartość przekracza 1 mld zł, a ryzyko wtórnych strat, przy awariach wynikłych z utraty kontroli nad istotnymi fragmentami procesu jest olbrzymie. Stąd na • czasie jest postulat oczekiwanej dyspozycyjności systemów CRIS powyżej 97% dla wszystkich funkcji decydujących o kontynuowaniu procesu.

Zrozumiała jest więc dążność do eliminowania ze struktury węzłów, których niesprawność mogłaby naruszyć jednocześnie kilka ważnych funkcji systemu. Problem można zilustrować następującym przykładem: do współczesnej nastawni bloku energetycznego wprowadza się równolegle około 1000 informacji dla operatora. W nieustalonych stanach procesu wynika stąd ryzyko częściowej utraty kontroli przez operatora wskutek jego ograniczonych możliwości percepcyjnych. Należy więc wprowadzić system informacyjny złożony z monitorów numeryczno-graficznych i zapewnić automatyczną selekcję informacji dla operatora. W klasycznej realizacji działanie tego systemu zależy od niezawodności minikomputera, w tym jego procesora, kanałów i pamięci zewnętrznej, tak więc wprowadzenie systemu monitorowego, jako podstawowego środka łączności operatora z obiektem wymaga na ogół zdublowania systemu centralnego i specjalnych rozwiązań w dziedzinie oprogramowania lub zorganizowania systemu monitorowego w układzie autonomicznym tak, jak to ma miejsce w systemie ESIW zaprojektowanym przez „Mera-ZAP-Mont.”

Przedstawione wyżej przesłanki spowodowały sformułowanie koncepcji wprowadzenia do systemów CRIS struktur zdecentralizowanych. W takiej strukturze poszczególne funkcje systemu realizowane są przez wydzielone moduły sprzętowe pracujące jako autonomiczne podsyste-

temy. Komunikują się one z obiektem za pośrednictwem wspólnej magistrali danych kontrolowanej przez wyodrębniony mikrokomputerowy sterownik komunikacyjny.

Magistrala danych otrzymuje sygnały z obiektu za pośrednictwem urządzeń sprzężenia /USO/ w postaci wstępnie przetworzonej.

Jest oczywiste, że realizacja tak zaprojektowanej struktury wymaga wprowadzenia do układu licznych modułów realizujących równoległe funkcje przetwarzania i komutacji, dla których mikroprocesory i inne elementy makroskalone stanowią idealną bazę wyjściową. Przydzielenie sterownikom mikroprocesorowym wydzielonych funkcji pozwala akceptować ich ograniczone szybkości działania i zakresy możliwości funkcjonalnych, niski ich koszt zezwala w potrzebnych przypadkach stosowanie redundancji jako środka dla podwyższenia niezawodności systemu.

Pojawiły się już pierwsze realizacje systemów CRIS o strukturze zdecentralizowanej. Przykładem, może być system TDC-2000 /Total Distributed Control/ firmy "Honeywell".

Jednym z trudnych problemów związanych z wdrażaniem CRIS złożonych procesów technologicznych jest organizacja transmisji danych w relacji obiekt - system - operator. Typowy wielki obiekt sterowania i kontroli rozłożony jest w promieniu 3-5 km. Krótki, wymagany czas dostępu do informacji powstającej w jednym z kilku tysięcy punktów kontrolnych /rząd milisekund/ narzucał dotąd potrzebę równoległej transmisji sygnałów, z przewagą postaci analogowej sygnału wobec wysokiego poziomu zakłóceń spowodowanego instalacjami elektroenergetycznymi obiektu. Prowadzi to w praktyce do potrzeby instalowania rozległej i bardzo kosztownej sieci połączeń kablowych. W systemie CRIS Janikowskich Zakładów Sódowych zrealizowanym przez Mera-PIAP i „Mera-Pnefal” zainstalowano ok. 250 km nowych połączeń kablowych.

Rozwiązanie problemu możliwe jest przez wprowadzenie do systemów CRIS układów szybkiej cyfrowej transmisji szeregowej i wprowadzenie głównej magistrali danych do kilku pętli jedno lub dwuprzewodowych. Trzeba jednak wówczas zapewnić bardzo wysoki stopień poprawności przekazywania danych. W przytoczonym już dla przykładu systemie TDC-2000 transmisja zorganizowana jest przy pomocy 1-3 pętli przewodu koncentrycznego na odległość do 1,5 km. Szybkość transmisji wynosi 256 Kbd. Słowo transmitowane zawiera 31 bitów, w tym 16 bitów zawiera pole danych, 6 bitów pole adresu źródła. 3 bity kod operacji, 5 bitów pole kodu kontrolnego. Procedura kontroli obejmuje:

- sprawdzenie "bit po bicie" /każdy bit zawiera impuls dodatni i ujemny/;
- kontrolę według kodu 31, 26 BCH zapewniającą wykrycie wszystkich błędów do 5 bitów, wszystkich kombinacji dwóch błędów przypadkowych, wszystkich inwersji 98,5% błędów

6 bitowych i 95% błędów obejmujących więcej niż 6 bitów,

- potwierdzenie odbioru pakietu danych z ewentualną kontrolą echa tj. sprawdzeniem przez nadajnik "kopii" przesłanych danych zwróconej przez odbiornik.

Taka organizacja transmisji danych wymaga oczywiście zastosowania mikroprocesorów w jej układach, jeśli uzyskane korzyści nie mają być zniwelowane kosztami sprzętu obsługującego samą tylko transmisję.

Systemy regulacji analogowych procesów wolnozmiennych

Od szeregu lat czynione są próby zastąpienia klasycznych systemów regulacji automatycznej procesów analogowych centralnymi wielokanałowymi regulatorami cyfrowymi w zastosowaniach do obiektów wieloobwodowych. Podstawą tych przedsięwzięć są nadzieje na obniżenie kosztu i rozszerzenie możliwości funkcjonalnych. Do czasu pojawienia się techniki mikrokomputerów trudno było uzyskać konkurencyjne relacje kosztu do wartości użytkowej i systemy bezpośredniej regulacji cyfrowej uzyskiwały znaczenie praktyczne tam, gdzie postać sygnałów obiektowych lub postać algorytmu regulacji utrudniały zastosowanie klasycznych systemów analogowych takich jak np. : krajowe systemy URS-INTELEKTRAN, PNEFAL czy EFTRONIK.

Wprowadzenie techniki mikrokomputerowej pozwala na przyjęcie nowego kierunku w rozwoju podsystemów regulacji procesów analogowych. Kierunek ten prowadzi do struktury hybrydowej, w której z dotychczasowego analogowego podsystemu pozostają przetworniki pomiarowe, zmodyfikowane stacyjki operacyjne, wzmacniacze wyjściowe regulatorów i elementy wykonawcze. Natomiast moduły funkcjonalne regulatorów formujące algorytm regulacji zastępowane są centralnym mikrokomputerem lub w uproszczonej postaci zachowane jako regulatory rezerwowe.

Wprowadzenie mikrokomputera obsługującego do 20-30 obwodów regulacji umożliwia, przy zachowaniu korzystnych relacji kosztów, wzbogacenie systemu o możliwość realizacji niekonwencjonalnych algorytmów regulacji, ułatwienie, a nawet automatyzację zmian struktury obwodów regulacyjnych czego wymagają niektóre obiekty np. przy zmiennym ich obciążeniu. Ten sam sterownik centralny może być wykorzystany do realizacji funkcji sygnalizacji technologicznej, rejestracji cyfrowej, sterowania sekwencyjnego, jeśli zakres tych funkcji dla rozpatrywanego procesu jest zbyt mały, aby dla ich realizacji opłacało się zastosować odrębne wyspecjalizowane środki. Przykładem realizacji omawianego kierunku może być system UOC opracowany przez firmę CATCO-USA i wykorzystujący mikrokomputer TM 9900 firmy "Texas Instruments".

systemy sterowania procesami dyskretnymi

Procesy dyskretne stanowią obszerną klasę

w przemyśle elektromaszynowym, hutniczym, spożywczym i wielu innych. Automatyzacja tych procesów, ze wzrostem skali i seryjności produkcji nabiera znaczenia. Najpopularniejsze dotąd techniki ich automatyzacji to technika przekaźników elektromechanicznych i technika sztywnoprogramowych układów elektronicznych. Układy przekaźnikowe charakteryzują się ograniczoną szybkością działania i dużą wrażliwością na warunki otoczenia/ zapylenie, wilgotność, drgania/. Układy elektroniczne sztywnoprogramowane w realizacji z układów małej skali integracji są znacznie kosztowniejsze i sprawiają trudności przy wdrożeniu, ze względu na komplikacje przy modyfikacji algorytmów, częst. wymaganej w trakcie rozruchu. Sterowniki swobodnie programowane, szczególnie jeśli zastosować w nich pamięci półprzewodnikowe typu EPROM, czy EAROM pozwalają uzyskać korzystne relacje kosztu do wartości użytkowej i skrócić czas projektowania i wdrożenia, dzięki uniezależnieniu budowy sterownika od algorytmu sterowania. Oplaca się je stosować już dla stosunkowo nieskomplikowanych obiektów od 25 torów wejścia/wyjścia. Zastosowanie techniki mikroprocesorowej pozwala na znaczne podwyższenie niezawodności tych układów /teoretycznie do ok. 80 000 godzin czasu między awariami/.

Przykładem realizacji omawianego kierunku jest wdrażany aktualnie do produkcji w „Mera-ZAP-Mont” licencyjny system INTELSTER-PC.

Można przewidywać w najbliższym pięcioleciu burzliwy rozwój zastosowań sterowników programowo logicznych do automatyzacji procesów transportu technologicznego, automatów obróbczych procesów obróbki powierzchniowej itd. Wprowadzenie techniki mikroprocesorowej spowodowało burzliwy rozwój konstrukcji i zastosowań również w takich dziedzinach jak sterowanie cyfrowe obrabiarek i manipulatorów /robotów/ przemysłowych.

Nowe obszary zastosowań automatyki

Gwałtownie poprawiające się pod wpływem rozwoju podzespołów makroskalonych, relacje kosztu do wartości użytkowej urządzeń automatyki elektronicznej otwierają nowe pola dla jej zastosowań. Należy tu wymienić: automatyzację sprzętu gospodarstwa domowego - ste-

rowanie pralek automatycznych, kuchni i piekarników, szczególnie mikrofalowych itp. Wiadomo jakie znaczenie społeczne ma postęp i racjonalizacja w sferze gospodarstwa domowego, wiadomo również jaki problem stanowi obecnie sprawa niezawodności elektromechanicznych elementów automatyki zmechanizowanego sprzętu gospodarstwa domowego. Obwody scalone wielkiej skali integracji stanowią z pewnością najlepszą bazę dla uzyskania optymalnych rozwiązań w tym zakresie.

Drugim bardzo interesującym obszarem jest automatyzacja pojazdów z silnikami spalinowymi. Szacuje się, że zastąpienie obecnie stosowanych układów przygotowania i dawkowania paliwa do silników samochodowych elektronicznymi układami o dostatecznie rozwiniętym algorytmie, umożliwiłoby zarówno spadek zużycia paliwa o ok. 10%, jak i znaczne zmniejszenie zanieczyszczenia atmosfery toksycznymi składnikami gazów wylotowych.

Dwie wyżej wymienione nowe dziedziny zastosowań mogą w perspektywie najbliższych lat zaangażować ponad 70% całej produkcji ilościowej mikroprocesorów!

W niniejszym artykule dokonano krótkiego przeglądu niektórych tylko zagadnień związanych z wprowadzeniem techniki mikroprocesorowej do systemów automatyki. Wskazano na niektóre problemy, których korzystne rozwiązanie możliwe jest przy pomocy tej techniki. Jak wielki umożliwia ona postęp można zilustrować kolejnym przykładem. Komputer AN/UYK-30 firmy "Hughes" - przy parametrach:

- długość słowa 16 bitów
- szybkość operacji 340-660 tys. op/s
- pamięć operacyjna 64 k słów
- MTBT - 20000 godzin
- zakres temperatury pracy -55°C do +125°C

mieści się w objętości 320 cm³, tj. w objętości miniaturowego odbiornika radiowego.

Nasuwa się wniosek, że należy prace badawczo-rozwojowe tak ukierunkować, aby z chwilą pojawienia się możliwości dostaw obwodów scalonych wielkiej skali integracji w najkrótszym możliwym cyklu rozpocząć seryjną produkcję urządzeń i systemów automatyki w pełni wykorzystujących zalety omawianej techniki.

MOŻLIWOŚCI ZASTOSOWANIA MIKROPROCESORÓW W ELEKTRONICZNEJ APARATURZE KONTROLNO-POMIAROWEJ

W rozwoju elektronicznej aparatury pomiarowej zaznaczają się ostatnio dwie dominujące tendencje: przystosowanie przyrządów pomiarowych do pracy w systemach pomiarowych oraz automatyzacja ich pracy z wprowadzeniem elementów przetwarzania danych. Opracowanie takich systemów mikroprocesorowych wychodzi naprzeciw wymienionym tendencjom rozwojowym, pozwalając konstruktorom aparatury realizować ekonomiczne konstrukcje o znacznie ulepszonych właściwościach metrologicznych i użytkowych. W artykule niniejszym zostaną omówione ważniejsze aspekty zastosowania układów mikroprocesorowych w aparaturze pomiarowej oraz zostaną podane przykłady konkretnych ich rozwiązań.

1. Funkcje mikroprocesorów w elektronicznych przyrządach pomiarowych

Zastosowanie mikroprocesorów w konstrukcji przyrządów pomiarowych pozwala zrealizować nie tylko takie tradycyjne funkcje przyrządów, jak zbieranie informacji pomiarowych i ich wizualizacja, a także szereg nowych funkcji możliwych dotychczas do uzyskania jedynie w rozbudowanych systemach pomiarowo-informacyjnych, takich jak: przetwarzanie danych pomiarowych, automatyzacja nastaw i obsługi, autokalibracja, kontrola prawidłowości obsługi itp.

W ostatnich dwóch latach opracowano w produjących firmach, produkujących przyrządy pomiarowe, szereg ciekawych konstrukcji, wykorzystujących mikroprocesory. Różnią się one między sobą stopniem automatyzacji, zastosowanymi algorytmami przetwarzania danych, organizacją itp. Daje się zauważyć, że walczą ze sobą o lepsze pomysły konstruktorów dążącego do polepszenia właściwości użytkowych przyrządów i dopuszczalnym ekonomicznie

stopień dodatkowego skomplikowania aparatury. Analizując dostępne dane techniczne znajdujących się już na rynku mikroprocesorowych przyrządów pomiarowych, można dojść do uogólnionego poglądu na temat funkcji realizowanych w nich przez mikroprocesory. Są one następujące:

Automatyczne zerowanie i kalibracja. Przyrządy pomiarowe z wbudowanym mikroprocesorem pracują z zasady cyklicznie, z wielokrotnie zwykle powtarzaniem cyklu pomiaru. Pomiedzy cyklami pomiarowymi co pewien okres czasu wejście przyrządu jest odłączane i jest mierzony aktualny "poziom zerowy", a następnie do wejścia automatycznie podłączany jest wbudowany wzorzec wielkości mierzonej. Wyniki obydwu pomiarów wprowadzone do pamięci mikroprocesora, służą następnie do automatycznej korekcji kolejnych wyników pomiarowych. Zastosowanie tej procedury pozwala zrealizować w przyrządzie pomiarowym użytkową dokładność pomiaru bardzo bliską dokładności wbudowanego wzorca, eliminując uciążliwą dla obsługującego procedurę częstej kontroli zerowania i kalibracji.

Automatyczny wybór zakresu pomiarowego w przyrządach wielozakresowych z jednoczesną optymalizacją, tzn. ustawienie zakresu, dającego największą dokładność lub największą ilość znaczących miejsc wyniku pomiaru. Procedura ta realizowana jest drogą analizy logicznej wyniku i odpowiedniego sterowania automatycznym przełączaniem zakresów. Znaczenie jej jest podobne, jak procedury już opisanej.

Wprowadzenie prostych transformacji wyników pomiaru /nieliniowych lub liniowych/. Możliwe jest uzyskanie tą drogą korekcji występujących w torze pomiarowym nieliniowości, a także przekształcenie wyników pomiarów w

taki sposób, aby odczyt uzyskać wprost w jednostkach fizycznych. W przyrządach, których wskazania są np. funkcją temperatury, stosowana jest automatyczna korekcja wyniku połączona z pomiarem wielkości wpływającej. Program obliczeń może być na stałe umieszczony w pamięci mikroprocesora i wywoływany automatycznie lub inicjowany na żądanie mierzącego, który może ręcznie wprowadzić niektóre dane do programu.

Złożone przetwarzanie wyników pomiarowych. Ilość możliwych sposobów przetwarzania jest ogromna, zwłaszcza przy wykorzystaniu pełnych szybkości obliczeniowych mikroprocesora. Możliwe jest zastosowanie zarówno procedur natychniastowego przetwarzania jak i procedur wtórnego przetwarzania pierwotnych wyników pomiarów. Spotyka się operowanie zarówno na pojedynczych wynikach pomiarów, jak i na uprzednio zgromadzonych w pamięci całych zbiorach danych pomiarowych. W grupie złożonego przetwarzania danych szczególną rolę odgrywają obliczenia statystycznych parametrów zbioru zmierzonych wielkości. Znaczna większość mierników jedno- lub dwuparametrowych wielkości posiada możliwość przeprowadzenia obliczenia średniej zbioru i dyspersji. Niektóre przyrządy podają histogram rozkładu wartości zbioru. Zastosowanie odpowiednich algorytmów przetwarzania pozwala w wielu wypadkach zrealizować bezpośredni pomiar takich wielkości, których metodami klasycznymi nie można było uzyskać. Bardzo łatwo zrealizować też można algorytm kontroli odchyłki kolejnej wartości mierzonej od wartości nominalnej, połączonej z sygnalizacją przekroczenia zadane go progu. Algorytm ten bywa często stosowany w przyrządach np. dla celów testowania elementów i nadzorowania szeregu procesów technologicznych.

Budowanie odpowiedniego formatu wyników pomiaru. Przyrządy z wbudowanym mikroprocesorem posiadając odpowiednią pamięć, wydają zazwyczaj wynik pomiaru łącznie z jednostką fizyczną, a często z dodatkowym komentarzem, zazwyczaj w kodzie ASCII. Pozwala to na bezpośrednią, elegancką rejestrację wyników pomiarów za pomocą drukarek alfanumerycznych - w uzasadnionych wypadkach również w postaci tablic.

Programowanie nastaw przyrządów pomiarowych. Specyficzną własnością niektórych przyrządów pomiarowych lub źródeł sygnałów jest możliwość ich ciągłego lub dyskretnego przestrajania /cecha np. woltomierzy selektywnych, generatorów częstotliwości/. Wbudowany mikroprocesor pozwala na zestawienie sekwencyjnie realizowanego programu przestrajania z ewentualnym równoczesnym programowaniem towarzyszącym nastaw innych parametrów. Użytkuje się w ten sposób możliwość automatycznych pomiarów np. całych charakterystyk częstotliwościowych - właściwość szczególnie cenną przy powtarzanych wielokrotnie pomiarach kontrolnych. Zaprogramowane nastawy mogą

być realizowane w stałych odstępach czasowych lub w momentach czasowych z góry zaprogramowanych. W tym ostatnim przypadku zazwyczaj możliwe jest uzyskanie dodatkowego wydruku tabeli nastaw czasowych.

Sterowanie urządzeniami współpracującymi z przyrządem pomiarowym. Problem ten występuje często w bardziej złożonych przyrządach pomiarowych /np. w analizatorach/, w których zachodzi konieczność utrzymywania określonych reżimów przyrządu lub stanowiska pomiarowego. Dla przykładu można tu wymienić stabilizację lub programowanie temperatury, pól magnetycznych itp. Mikroprocesor realizuje wówczas odpowiedni algorytm regulacji lub sterowania cyfrowego, często kilkuparametrowego. Ponieważ procedura ta obciąża mikroprocesor zwykle w niewielkim stopniu, bywa on zazwyczaj wykorzystany do realizacji innych funkcji, opisanych powyżej.

Zapewnienie współpracy w ramach systemów pomiarowo-informacyjnych. Opisane powyżej funkcje mikroprocesora w przyrządzie pomiarowym zapewniają automatyzację szeregu procedur, związanych zarówno z przebiegiem pomiaru jak i przetwarzaniem danych pomiarowych. Jednocześnie mikroprocesor umożliwia łatwą realizację sprzężenia przyrządu pomiarowego z innymi urządzeniami poprzez standardowy interfejs cyfrowy. Ułatwia to budowę systemów pomiarowo-informacyjnych o strukturze hierarchicznej, gdy przyrządy pomiarowe nowej generacji z wbudowanym mikroprocesorem wykonują autonomicznie szereg zadań, odciążając jednostkę centralną systemu. Szereg przyrządów może być z powodzeniem stosowanych jako samodzielne, małe systemy pomiarowe.

Z przedstawionego zestawienia funkcji mikroprocesorów w aparaturze pomiarowej wynika że wymagania techniczne stawiane mikroprocesorom są bardzo zróżnicowane, zależnie od rodzaju automatyzowanego przyrządu i złożoności stosowanych algorytmów. Do przyrządów o mało złożonym przetwarzaniu wystarczy stosować proste cztero lub ośmiobitowe mikroprocesory z prostą listą rozkazów. Do konstrukcji złożonych przyrządów pomiarowych wykorzystywane są mikroprocesory 8 lub 16 bitowe posiadające rozbudowany, wektoryzowany system przerw i obszerną listę rozkazów. Bardzo często najbardziej dogodna jest arytmetyka dziesiętna. W niektórych zastosowaniach wymagana jest duża szybkość wykonywania operacji matematycznych. Najkorzystniejsze są wówczas mikroprocesory o strukturze plasterkowej, umożliwiające budowę procesora o dowolnej żądanej długości słowa, zdolnego do realizacji operacji matematycznych równoległe na słowach o maksymalnej długości.

2. Ważniejsze problemy projektowania przyrządów pomiarowych z mikroprocesorami

Konstruktor elektronicznych przyrządów pomiarowych, pragnący opracować zautomatyzo-

wany przyrząd z użyciem mikroprocesora, staje przed całym szeregiem zupełnie nowych problemów koncepcyjnych, technicznych i organizacyjnych w porównaniu z metodologią projektowania klasycznych przyrządów pomiarowych. Pokonanie ich wymaga zwykle nowego spojrzenia na funkcje przyrządu pomiarowego, umiejętności formalnego opisu działania przyrządu, niezbędnego dla opracowania algorytmów jego pracy, a także umiejętności wykorzystania środków informatyki w procesie projektowania. Oto ważniejsze z owych zagadnień:

Opracowanie koncepcji przyrządu. Na etapie tym należy rozstrzygnąć /w oparciu o studia nad możliwymi sposobami zastosowania przyrządu/ zakres wprowadzanej automatyzacji przyrządu. Należy przy tym przeprowadzić ocenę wzrostu użyteczności przyrządu w stosunku do wzrostu ceny w wyniku jego rozbudowy. W wielu przypadkach, podejmując decyzję o automatyzacji przyrządu, należy rozwiązać problem dostosowania pracy części pomiarowej przyrządu do wymogów sterowania lub przetwarzania cyfrowego. Często wymaga to dużego nakładu pracy badawczej, zwłaszcza w przypadku przyrządów o analogowej zasadzie działania, przyrządów o skrajnych zakresach pomiarowych, o dużej wymaganej szybkości działania itp.

Komunikacja: użytkownik - przyrząd. Projektujący musi brać stale pod uwagę, iż przeciętny użytkownik przyrządu pomiarowego winien komunikować się z przyrządem przy jego obsłudze w sposób prosty, łatwy do wyuczenia i zapamiętania. Jednocześnie muszą być spełnione wymogi łatwego komunikowania się z mikroprocesorem, stanowiącym z punktu widzenia wymiany informacji - część centralną przyrządu. Najbardziej dogodnym rozwiązaniem jest zastosowanie w miejsce tradycyjnych pokręteł i przełączników - klawiatury funkcjonalnej dla wyboru reżimów pracy, wyboru funkcji lub parametru oraz dziesiętnej klawiatury numerycznej /czasem: ograniczonej alfanumerycznej/ dla zadawania wartości liczbowej. Wprowadzane nastawy muszą być kontrolowane na odpowiednich wyświetlaczach numerycznych lub alfanumerycznych, zastępujących tradycyjne skale. W niektórych przypadkach celowe jest zastosowanie wbudowanej lub podłączonej drukarki. Czasami wprowadzenie danych łączy się z kontrolą ich poprawności formalnej w mikroprocesorze. Zasadą jest zawsze takie rozwiązanie komunikacji z przyrządem, by stany wszelkich nastaw mogły być w każdej chwili bezpośrednio odczytane.

Wyniki pomiarów muszą być uwidocznione lub rejestrowane w odpowiednim formacie, w zapisie dziesiętnym łącznie z jednostkami względnie z odpowiednim komentarzem opisowym.

W niektórych przyrządach pomiarowych, w których stosowane jest wielopunktowe programowanie nastaw, może być celowe zastosowa-

nie zewnętrznej pamięci stałej /np. karty magnetycznej/ dla utrwalenia wprowadzonego programu i jego odtwarzania po ponownym załączeniu przyrządu.

Jak widać z powyżej przytoczonych uwag, stosowane rozwiązania konstrukcyjne są podobne do używanych powszechnie w kalkulatorach elektronicznych, gdzie wykazały swoje zalety. Opracowanie oprogramowania. Przyrządy pomiarowe z wbudowanym mikroprocesorem posiadają zawsze stałe oprogramowanie, umieszczone w stałej pamięci. Przy produkcji średnio i wielkoseryjnej opłaca się zoptymalizować oprogramowanie w taki sposób, aby zminimalizować wielkość pamięci stałej i operacyjnej, gdyż odbija się to korzystnie na wynikowym koszcie produkcji przyrządu.

Optymalizacja oprogramowania możliwa jest np. przez wybór arytmetyki dziesiętnej /wynika to z faktu, że dane wejściowe i wyjściowe są w reprezentacji dziesiętnej/. Innym środkiem jest zastosowanie zasady mikroprogramowania z efektywnym wykorzystaniem całej listy rozkazów mikroprocesora.

Minimalizacja pamięci operacyjnej jest możliwa przez przydzielanie tych samych obszarów pamięci do realizacji sekwencyjnie występujących elementów programu, wykorzystywanie zasady stosu itp. W przyrządach, których praca przebiega w czasie rzeczywistym, można uzyskać spore oszczędności sprzętowe, organizując zegar czasu rzeczywistego środkami programowymi bezpośrednio w mikroprocesorze.

Bardziej złożone przyrządy pomiarowe, realizujące kilka zadań równocześnie, wymagają zazwyczaj zastosowania efektywnego systemu operacyjnego. W takim przypadku przy projektowaniu oprogramowania wskazane jest posłużenie się metodą symulowania pracy przyrządu pomiarowego na maszynie cyfrowej i sprawdzenia w tych warunkach opracowanego oprogramowania. Szereg firm produkujących mikroprocesory dostarcza zestawy mikroprocesorowe specjalnie przeznaczone do prac tego rodzaju.

Ogólnie można powiedzieć, że opracowanie poprawnego oprogramowania dla mikroprocesorowych przyrządów pomiarowych wymaga zwykle współpracy specjalistów - programistów znających bardzo dobrze wszystkie techniczne właściwości systemu mikroprocesorowego. Okoliczność ta wymaga podkreślenia, gdyż wśród konstruktorów aparatury pomiarowej w kraju daje się zauważyć swoista fascynacja możliwościami sprzętu mikroprocesorowego przy równoczesnym niedocenianiu trudności i złożoności przygotowania oprogramowania.

3. Przykłady wykorzystania mikroprocesorów w aparaturze pomiarowej

Mikroprocesory znalazły dotychczas zastosowanie w następujących grupach przyrządów pomiarowych:

- w częstotliwościach cyfrowych,
- w woltomierzach cyfrowych i przyrządach pochodnych jak pII-metry, wielokanałowe zestawy zbierania danych pomiarowych itp.,
- w przyrządach do pomiaru impedancji,
- w różnych analizatorach funkcji czasowych,
- w oscyloskopach elektronicznych
- w syntezerach częstotliwości.

Dla zobrazowania korzyści stąd płynących, omówione będą bliżej trzy przykłady przyrządów mikroprocesorowych:

• Woltomierz cyfrowy posiadający możliwość pomiaru /równoczesnego/ napięć i prądów stałych i zmiennych. Zrealizowanie równoczesnego pomiaru napięć i prądów stałych i zmiennych pozwala na zrealizowanie z pomocą mikroprocesora pomiaru dodatkowych wielkości pochodnych, a mianowicie:

- mocy prądu stałego: $P_{\text{dc}} = U_{\text{dc}} \cdot J_{\text{dc}}$

- oporności i przewodności: $R = \frac{U}{J}$ $G = \frac{1}{R}$

- impedancji zespolonej: $\hat{Z} = \frac{\hat{U}}{\hat{J}} = R + jX$

- skutecznej wartości napięcia: $U_{\text{sk}} = \frac{1}{T} \sqrt{\int_0^T U^2 / dt}$

- "średniej wyprostowanej" napięcia:

$$U_{\text{sr}} = \frac{1}{T} \int_0^T U / dt$$

- mocy chwilowej: $P_{\sim} = U / t \cdot J / t$

Górna granica częstotliwości, do której możliwe jest uzyskanie poprawnych wyników, zależy bezpośrednio od szybkości realizowania operacji w mikroprocesorze. W przyrządach tego typu stosowane są zawsze: automatyczna kalibracja i korekcja dryftu zera, a także często korekcja nieliniowości przetwornika a/c. Zakresy pomiarowe mogą być programowane ręcznie lub zdalnie i wybierane automatycznie. Woltomierze z reguły pozwalają ponadto obliczać:

- średnią z n pomiarów: $U = \frac{1}{n} \sum U_n$

- dyspersję mierzonej wielkości: $\sigma = \frac{1}{n} \sum (U - U_n)^2$

a także sumę, różnicę i stosunek dwóch wielkości mierzonych. Często spotyka się również sygnalizację przekroczenia zadanych poziomów. Przyrządy pozwalają też zazwyczaj na pomiar temperatury standardowym czujnikiem, przy czym wynik podawany jest bezpośrednio w jednostkach temperatury.

• Syntezier częstotliwości

Syntezier częstotliwości jest źródłem sygnału sinusoidalnego o dokładnej częstotliwości i amplitudzie. Funkcje mikroprocesora w przyrządzie tego typu polegają głównie na sterowaniu częstotliwością i napięciem. W szczególności realizowane jest:

- przestrajanie częstotliwości w całym paśmie pokrywającym przez syntezier skokowo i z dowol-

na dyskretnością, bezpośrednio z klawiatury numerycznej lub zdalnie.

- wobulacja częstotliwości według dowolnej zależności, określonej programem i w całym paśmie częstotliwości pracy. Parametry wobulacji są obliczane automatycznie stosownie do zadanej częstotliwości środkowej, dewiacji i ilości dyskretnych skoków,

- zestawienie programu pomiarów, polegające na kolejnym, automatycznym podawaniu przez syntezier szeregu sygnałów pomiarowych o zaprogramowanej częstotliwości i poziomie. W rezultacie zastosowania mikroprocesora uzyskuje się znaczne uproszczenie obsługi syntezera, wyższą dokładność parametrów napięcia wyjściowego i możliwość zestawienia prostych stanowisk o pracy automatycznej.

• Chromatograf gazowy

Chromatograf gazowy ze względu na dużą złożoność czynności obsługowych, konieczność przeprowadzania dużej ilości obliczeń "on line" i "off line" jest tym przyrządem pomiarowym, w którym zastosowanie mikroprocesora jest szczególnie uzasadnione.

Metoda chromatografii gazowej polega na rozdzielaniu mieszaniny analizowanych substancji w stanie gazowym /lub wprowadzonych w stan gazowy/, określeniu po rozdzieleniu procentowej zawartości poszczególnych składników oraz identyfikacji /pośredniej/ tych składników. Badaną próbkę dozuje się impulsowo do dozownika podgrzewanego do odpowiedniej stałej temperatury. Przez dozownik przepływa w sposób ciągły i kontrolowany gaz nośny, który dozowaną próbkę transportuje do tzw. kolumny chromatograficznej. Kolumna stanowi swego rodzaju "linię opóźniającą" o różnym opóźnieniu dla różnych składników, w wyniku skomplikowanych procesów fizyko-chemicznych, zachodzących między gazem a wypełniaczem umieszczonym w kolumnie. Dzięki temu rozdzielone składniki pojawiają się na wyjściu kolumny przesunięte w czasie i "rozmyte" w stosunku do impulsu wejściowego. Powoduje to swoistą modulację właściwości fizyko-chemicznych gazu, wypływającego z kolumny. Zmiany te zostają następnie przetworzone na ciągły sygnał elektryczny za pomocą detektora chromatograficznego. Każdemu składnikowi mieszaniny przyporządkowany jest charakterystyczny pik, którego powierzchnia jest miarą ilości składników, a opóźnienie wierzchołka pików w stosunku do momentu dozowania, zwane czasem retencji - służy do identyfikacji składnika. Obydwie wielkości muszą być mierzone z niedokładnością nie większą niż $\pm 1 - 2\%$.

Proces rozdziału w kolumnie chromatograficznej jest silnie uzależniony od temperatury. Stąd też kolumna musi być termostatyzowana z dokładnością rzędu 0,2 °C. Dla wielu analiz niezbędne jest także stosowanie narostu temperatury w czasie, przy czym narost ten musi być bardzo dokładnie odtwarzany. Współczesny

chromatograf gazowy z mikroprocesorem stanowi złożony system pomiarowo-informatyczny i regulacyjny, w którym można wyodrębnić następujące zespoły funkcjonalne:

- właściwy chromatograf gazowy, zawierający układy pomiarowe fizyko-chemiczne,
- zespół programowanych nastaw głównych parametrów analizy,
- zespół regulatorów temperatury szeregu fragmentów chromatografu,
- zespół pomiarów i rejestracji cyfrowej głównych parametrów analizy chromatografu,
- zespół przetwarzania danych: pierwotnych i wtórnych,
- pulpit operatora systemu.

• Programowanie nastaw parametrów analizy obejmuje:

- nastawienie temperatur dozownika i detektorów chromatograficznych,
- nastawianie temperatury początkowej i końcowej oraz szybkości narostu temperatury termostatu kolumn chromatograficznych,
- sterowanie wielkością przepływu gazu nośnego,
- sterowanie dozownikiem próbek gazu,
- sterowanie parametrami obróbki danych pomiarowych,
- sterowanie wydrukiem.

Każdy ze sterowanych parametrów może być programowo zmieniony kilkakrotnie w czasie trwania analizy w zadanym w trakcie programowania momencie czasowym.

Programowanie odbywa się za pomocą klawiatury funkcjonalno-cyfrowej; odpowiednia wartość parametrów i dane czasowe wprowadzane są do pamięci RAM mikroprocesora. W trakcie analizy mikroprocesor stale sprawdza wskazania zegara wewnętrznego, porównując je z zawartością pamięci i w przypadku zgodności - uruchamia odpowiednią procedurę zmiany parametru.

• Regulacja temperatury.

W zespole chromatograficznym regulowane są temperatury dozownika detektorów i termostatu kolumn. Zastosowanie mikroprocesora umożliwia bezpośrednie sterowanie i regulację cyfrową według optymalnych algorytmów z wykorzystaniem systemu przerwań. Upraszcza się przy tym znacznie sterowanie narostem temperatury termostatu według zadanych z góry programów czasowych.

• Pomiary głównych parametrów analizy.

Główne parametry analizy, a mianowicie: - wszystkie temperatury, przepływ gazu nośnego są centralnie rejestrowane, wielkości zmierzone są przeliczane na odpowiednie wielkości fizyczne i zapamiętane do końca analizy, a następnie wydrukowane. Wydruk może też następować na żądanie w każdym momencie czasowym /dla kontroli/.

• Pierwotna obróbka danych.

Sygnal z wyjścia detektora chromatograficznego jest skwantowany i w postaci dyskretnej poddany obróbce wyników, celem obliczenia powierzchni i czasu retencji /on line/ poszcze-

gólnych pików. W trakcie obróbki występuje:

- wstępna filtracja cyfrowa dla poprawienia stosunku sygnał/szum
- różniczkowanie sygnału i badanie znaku i wartości pochodnej dla określenia początku, końca, wierzchołka i ewentualnych punktów przegięcia pików
- obliczanie powierzchni pików i korekcja wyników ze względu na obecność dryft linii zerowej.
- matematyczny rozdział pików zachodzących na siebie. Operacje te są przeprowadzane w czasie rzeczywistym przez mikroprocesor, przy czym odpowiednie w danej chwili algorytmy obróbki dobierane są automatycznie w wyniku badania pochodnej. Wyniki pierwotnej analizy wyprowadzane są w formie wydruku "na bieżąco" oraz są zapamiętane dla wtórnej obróbki danych.

• Wtórna obróbka danych.

Wtórna obróbka danych, wykonywana na wynikach obróbki pierwotnej, obejmuje zwykle: - obliczanie procentowej zawartości składników w mieszaninie z uwzględnieniem współczynników korekcji czułości detektora, - obliczanie zawartości wagowej składników w stosunku do określonej substancji wzorcowej, - obliczanie tzw. indeksu retencji, umożliwiającego jednoczesną identyfikację składników.

Redagowanie raportu końcowego zachodzi również z pomocą mikroprocesora przez nadanie odpowiednich opisów identyfikacyjnych, stabilizowanie wyników itp.

• Diagnoza systemu.

Odpowiedni algorytm umożliwi również dialogową diagnozę wszelkich niepoprawnych kroków w zakresie przygotowania warunków analizy oraz wykrywanie występujących w czasie jej trwania niesprawności, jak np. zgaśnięcie płomienia w detektorze, zaniku przepływu gazu nośnego itp. Wyniki diagnozy mogą być przedstawione w formie wydruku. Możliwość ta ogromnie ułatwia obsługę chromatografu operatorowi, który z reguły staje bezradny wobec wszelkich usterek, ujawniających się w pracy tak złożonego systemu jak chromatograf.

••••

Efektem zastosowania mikroprocesorów w przyrządach pomiarowych jest zwiększenie funkcjonalności przyrządów, uzyskanie nowych możliwości pomiarowych, znaczna poprawa użytkowych właściwości metrologicznych. W rezultacie uzyskuje się znacznie korzystniejszą relację pomiędzy użytecznością przyrządów pomiarowych, a ich ceną mimo oczywistego wzrostu tej ostatniej. Z tego powodu należy oczekiwać dalszego upowszechnienia zastosowania mikroprocesorów w aparaturze pomiarowej i stałego wzrostu asortymentu zautomatyzowanych z ich pomocą przyrządów pomiarowych.

ZASTOSOWANIE MIKROPROCESORÓW W BLOKACH STERUJĄCYCH SYSTEMU CAMAC

W większości eksperymentów wymagających automatycznego przetwarzania danych można wyróżnić trzy zasadnicze etapy:

- pomiary - podczas których następuje gromadzenie i zbieranie danych, a niejednokrotnie wstępne ich przetworzenie.
- sterowanie - które może być niezależne /wywoływanie określonych działań w zadanych okresach czasu/ lub zależne /wynikające z dialogu między blokami pomiarowymi a urządzeniem sterującym/.
- końcowy etap eksperymentu, w którym podane są wyniki pomiarów lub przeprowadzana kontrola procesu sterowania przedstawiona w najwygodniejszej dla użytkownika formie

Różnorodność zagadnień, z którymi zetknęli się konstruktorzy aparatury CAMAC w każdym z wymienionych etapów spowodowały powstanie różnych rodzin bloków /wejściowych - wprowadzających dane do systemu, wyjściowych - wprowadzających wyniki, interfejsów komputerowych, bloków funkcjonalnych przetwarzających informację, pamięciowych, sterujących itd/ jak również powstanie kilku znormalizowanych wariantów architektury złożonych systemów wielokasetowych /gałąź szeregową równoległą, kasetą systemową/. Podstawowe warianty połączeń systemu ilustruje rys. 1.

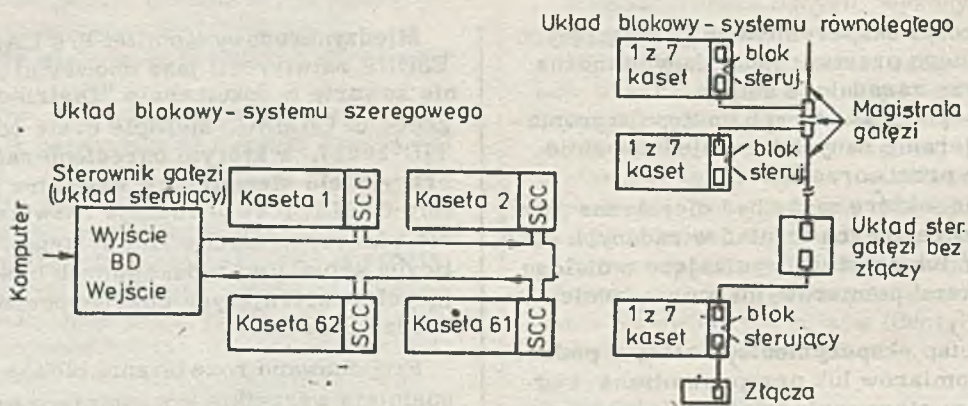
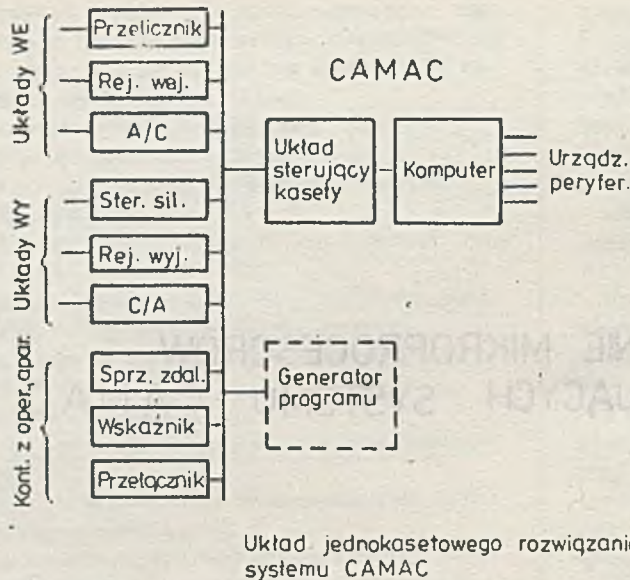
Znaczny wzrost proponowanych rozwiązań nowych konfiguracji systemu został wyraźnie przyspieszony przez masowe wprowadzenie scalonych elementów półprzewodnikowych. Powiększający się stale asortyment i stopień scalenia oferowanych układów umożliwia realizację coraz bardziej niezawodnych, wykonujących bardziej skomplikowane funkcje bloków systemu CAMAC, a przede wszystkim bloków interfejsowych i sterujących, w których podstawowym elementem jest mikroprocesor. Pojawienie się różnych typów mikroprocesorów umożliwiło powstanie nowego bardzo dynamicznie rozwijanego w systemie CAMAC kierunku tzw. - rozproszonej inteligencji.

Międzynarodowy Komitet d/s CAMAC - ESONE zatwierdził jako obowiązujące zalecenia zawarte w dokumencie "Distributed Intelligence in CAMAC. Multiple crate controllers" TID-26617, w którym określono zasady współpracy wielu sterowników wewnątrz jednej kasety CAMAC. Tego rodzaju rozwiązanie ogromnie rozszerza możliwości systemu. Obecnie liczba aktualnie produkowanych bloków sterujących zawierających mikroprocesory jest bardzo znaczna.

Proponowane rozwiązania bloków sterujących spełniają wszystkie wcześniejsze wymagania systemu, rozbudowując jedynie jego możliwości przy zachowaniu pełnej wymienności. Należy podkreślić, że mikroprocesory są stosowane coraz częściej w różnych blokach CAMAC zarówno wejściowych, wyjściowych jak i niektórych funkcjonalnych, jednakże ze względu na fakt, iż w określonych warunkach autonomiczne bloki sterujące umożliwiają wyeliminowanie droższych i bardziej złożonych systemów minikomputerowych ich liczba wzrasta nieco szybciej.

Podstawowe cechy autonomicznych sterowników kasety

Produkowane obecnie autonomiczne sterowniki kasety CAMAC bazujące na mikroprocesorach są zazwyczaj blokami o podwójnej szerokości /2M/ umieszczonymi w kasecie CAMAC na stanowiskach 24, 25 /stanowisko 25 jest stanowiskiem sterującym/ i zasilane z zasilaczem kasety. Spełniają one wszystkie funkcje właściwe sterownikom kasety opisane w PN 72 T-06530 lub dokumencie ESONE 4100. Wyposażone są w pamięć stałą typu EPROM o pojemności 2 + 8k słów 8-bitowych, pamięć o swobodnym dostępie RAM o pojemności 2 + 4k słów 8-bitowych, ośmiopozomowy kontroler przerwań oraz asynchroniczny nadajnik-odbiornik umożliwiający szczegółową komunikację z urządzeniem zewnętrznym np. dalekopisem.



Rys. 1. Podstawowe konfiguracje połączeń w systemie CAMAC

Autonomiczne sterowniki kasety z mikroprocesorem posiadają /zwykle wyprowadzoną na płytę czołową/ magistralę mikroprocesora, co umożliwia dołączenie dodatkowych modułów pamięci RAM/ROM będących również blokami CAMAC, a także umożliwia też eksperymentatorowi bezpośredni dostęp do pamięci. Niektóre sterowniki posiadają wyprowadzenia przeznaczone do podłączenia akumulatorów podtrzymujących zasilanie, w celu zabezpieczenia zawartości pamięci RAM przed zniszczeniem w przypadku awarii sieci. Sterowniki autonomiczne przeznaczone są głównie do sterowania jednokasetowych zestawów CAMAC. W niektórych jednak konfiguracjach praca ich może być również kontrolowana przez inny komputer.

Do transmisji danych w sterownikach kasety z mikroprocesorem najczęściej wykorzystywane są łącza szeregowe. Osiągana wówczas szybkość transmisji jest stosunkowo niewielka i wynosi 96000 do 19200 bodów. W przypadku, gdy wymagana jest znacznie większa szybkość transmisji danych wykorzystywany jest kanał bezpośredniego dostępu do pamięci /DMA/.

Zgodnie z zaleceniami normalizacyjnymi Komitetu ESONE określonymi w dokumencie TID 26617 sterowniki kasety powinny spełniać następujące podstawowe wymagania:

- umożliwiać budowę jednokasetowych zestawów CAMAC, w których centralna jednostka sterująca jest autonomicznym sterownikiem kasety,
- umożliwiać budowę jednokasetowych zestawów CAMAC, w których znajduje się kilka sterowników i zadania związane z obsługą eksperymentu są podzielone między nimi, przy czym jeden z nich pełni rolę sterownika głównego /Master Crate Controller/, a pozostałe pracują jako sterowniki pomocnicze /Auxiliary Crate Controller/,
- umożliwiać budowę zestawów wielokasetowych, w których poszczególne sterowniki pracują jako sterowniki pomocnicze. Takie rozwiązanie ułatwia lokalną obsługę eksperymentu i umożliwia przeprowadzenie wstępnej obróbki danych.

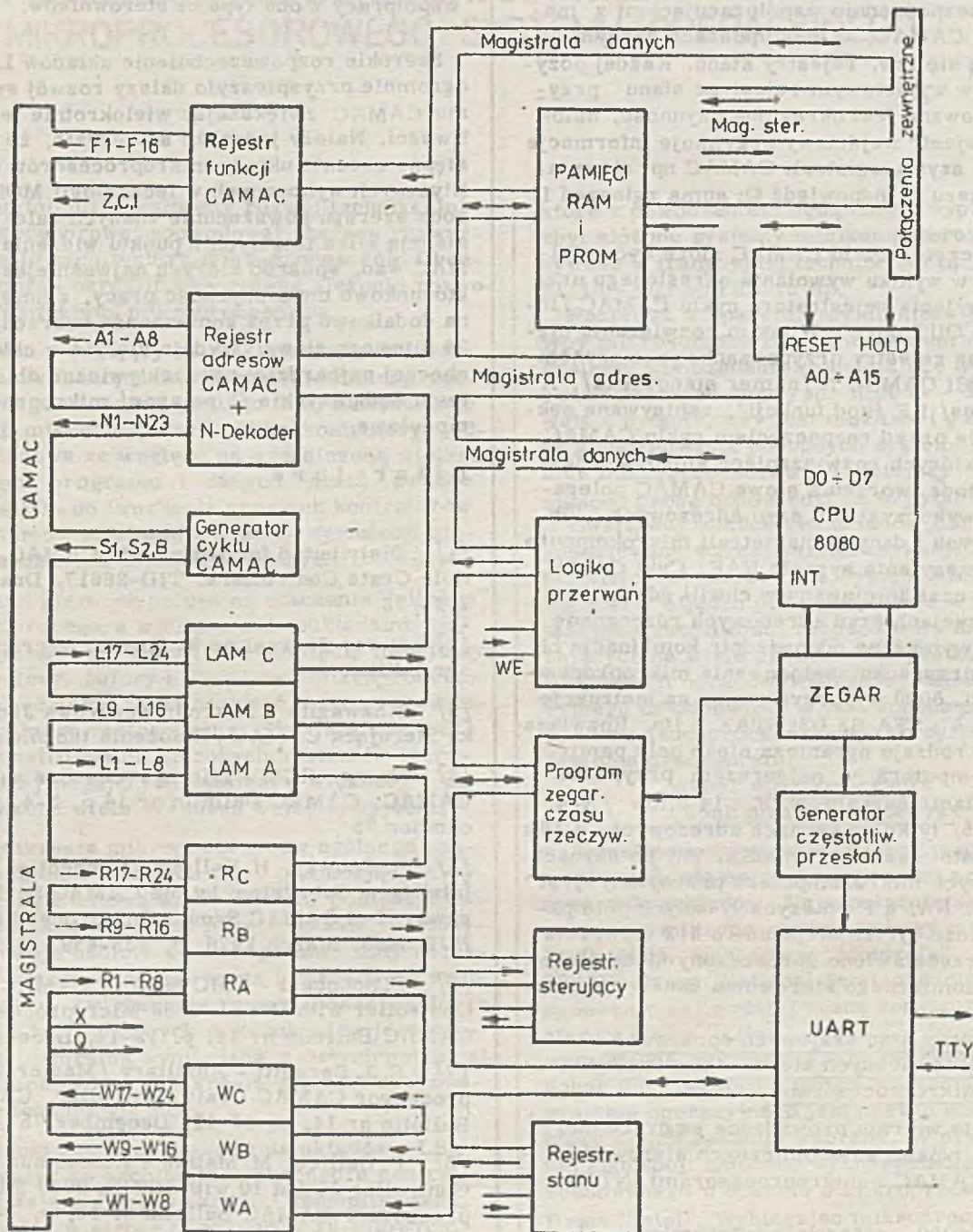
W przypadku awarii części systemu wybrane, najistotniejsze, zadania mogą być obsługiwane lokalnie przez sterownik pomocniczy.

Opis układu autonomicznego sterownika kasyety zawierającego mikroprocesor

W układzie elektronicznym dowolnego sterownika CAMAC zawierającego mikroprocesor można wydzielić dwie zasadnicze części:

- mikrokomputer,
- logika CAMAC.

W skład mikrokomputera wchodzi: mikroprocesor pełniący rolę centralnej jednostki sterującej /najczęściej, ze względu na bogate oprogramowanie INTEL 8080A lub TIC 6800/, pamięci danych RAM i program ROM, układ kontrolera przerwań, asynchroniczny nadajnik-odbiornik, zegar czasu rzeczywistego. Ponadto układ zawiera blok pośredniczący pomiędzy mikrokomputerem i magistralą CAMAC. Od strony mikrokomputera blok pośredniczący stanowi zespół urządzeń wejścia/wyjścia. Ze względu



Rys. 2. Schemat blokowy sterownika kasyety CAMAC z mikroprocesorem

na różną długość słowa systemu CAMAC /24 bity/ i najczęściej stosowanych mikroprocesorów /8 bit/ każdy rejestr bloku pośredniczącego pomiędzy szynami magistrali CAMAC i szynami magistrali mikrokomputera stanowi zespół trzech urządzeń wejścia lub wyjścia. Np. dane z szyn "R" /READ/ magistrali CAMAC są zapisywane w 24 bitowym rejestrze podczas jednego cyklu CAMAC /1 μs/, a odczyt tych danych przez mikrokomputer odbywa się po kolei w trzech cyklach mikrokomputera /z trzech 8 bitowych rejestrów - urządzeń wejścia/. Podobnie obsługiwane są inne rejestry bloku pośredniczącego. Poza urządzeniami we/wy bezpośrednio współpracującymi z magistralą CAMAC, w bloku pośredniczącym znajdują się tzw. rejestry stanu. Każdej pozycji bitu w wyjściowym rejestrze stanu przyporządkowana jest określona czynność, natomiast rejestr wejściowy przyjmuje informacje o stanie szyn magistrali CAMAC np: akceptacja rozkazu X; odpowiedź Q; suma zgłoszeń L itp.

Rozpoczęcie cyklu CAMAC może być zainicjowane w wyniku wywołania określonego urządzenia wyjścia rejestratora cyklu CAMAC /instrukcja OUT port/. W takim rozwiązaniu niezbędne są rejestry przyporządkowane szynom magistrali CAMAC N /numer stanowiska/, A /podadres/ i F /kod funkcji/, zapisywane sekwencyjnie przed rozpoczęciem cyklu CAMAC.

W niektórych rozwiązaniach stosowana jest inna metoda tworzenia słowa CAMAC polegająca na wykorzystaniu szyn adresowych lub adresowych i danych magistrali mikrokomputera do przesyłania sygnału NAF. Cykl CAMAC jest wówczas inicjowany w chwili, gdy na pewnych pozycjach szyn adresowych rozpoznane zostaną wymagane odpowiednie kombinacje bitów. W przypadku zastosowania mikroprocesora INTEL 8080 wykorzystywane są instrukcje MOV M, A : STA B2 B3; STAX B itp. Rozwiązanie tego rodzaju ogranicza nieco pole pamięci mikrokomputera w najgorszym przypadku /przesyłanie sygnału NAF - 14 bitów /N-4, A-5, F-5/ tylko po szynach adresowych/ o 16k słów. Natomiast w przypadku, gdy po szynach adresowych mikrokomputera przesyłany jest tylko np. NA, a F po szynach danych pole pamięci może być zmniejszone o 512 słów. Na rys. 2 przedstawiono uproszczony układ blokowy autonomicznego sterownika kasyety z mikroprocesorem.

Obecny stan krajowych opracowań autonomicznych sterowników kasyety z mikroprocesorami

Obecnie w kraju prowadzone są prace nad różnymi typami autonomicznych sterowników kasyety CAMAC z mikroprocesorami INTEL 8080 A:

- 1/ CMC 8080 - sterownik autonomiczny konwencjonalny przeznaczony do sterowania jednokasetowego zestawu CAMC posiadający:
 - pamięć RAM o pojemności 8 k słów;
 - pamięć EPROM o pojemności 2 k słów;
 - uniwersalny asynchroniczny nadajnik-odbiornik o szybkości transmisji do 9600 baudów;
 - programowany zegar czasu rzeczywistego;
 - 8- poziomowy kontroler przerwań.

2/ CMC 8080 - wersja o zwiększonej pojemności pamięci EPROM do 8 k słów.

3/ MKC 80 - sterownik posiadający możliwość pracy jako sterownik pomocniczy, /zgodny z zaleceniami ESONE/ posiadający:

- pamięć RAM o pojemności 1 k słów;
- pamięć EPROM o pojemności 2 k słów;
- uniwersalny asynchroniczny nadajnik-odbiornik;
- programowany zegar czasu rzeczywistego;
- 8- poziomowy kontroler przerwań.

Ponadto opracowywane są różne wersje bloków pamięci RAM i ROM przeznaczone do współpracy z obu typami sterowników.

Szerokie rozpowszechnienie układów LSI ogromnie przyspieszyło dalszy rozwój systemu CAMAC zwiększając wielokrotnie jego możliwości. Należy jednakże stwierdzić, że dotychczasowe obecnie układy mikroprocesorów monolitycznych wykonanych w technologii MOS oprócz szeregu powszechnie znanych zalet posiadają kilka istotnych z punktu widzenia CAMAC wad, spośród których najważniejszą jest stosunkowo mała szybkość pracy, zmniejszona dodatkowo przez konieczność tworzenia 24 bitowego słowa. Wydaje się, że w chwili obecnej najbardziej perspektywiczne dla systemu będą szybkie /bipolarne/ mikroprocesory modułowe.

L i t e r a t u r a

- [1] Distributed Intelligence in CAMAC. Multiple Crate Controllers. TID-26617, Draft 7/28/76
- [2] Borer Journal nr 8. Special Microprocessor Edition
- [3] S.Szwaglis - Mikroprocesorowa Jednostka Sterująca CAMAC /Założenia techniczne/
- [4] Barnes RCM - Microprocessors and CAMAC. CAMAC Bulletin nr 14 p. 3-4, December 75
- [5] R. Conway, H. Halling - Connection of an Intelligent Subsystem by the CAMAC Serial Highway 2-d CAMAC Symposium proceedings, EUR 5485, March 1976, p. 435-439
- [6] E. Schöberl - CMC 8080: A CAMAC Crate Controller with INTEL 8080 Microprocessor CAMAC Bulletin nr 14, p. 12-14, December 75
- [7] E. J. Barsotti - Auxiliary /Master Microprocessor CAMAC Crate Controller. CAMAC Bulletin nr 14, p. 17-19, December 75
- [8] P. Gallice, M. Mathis - Autonomus Crate Controller JCAM 10 with INTEL 8080 Microprocessor. CAMAC Bulletin nr 14, p. 7-10, December 75
- [9] St. Kościacz, K. Rzymkowski - Mikroprocesory. PTJ 2, 1976
- [10] K. Rzymkowski - Typowe zestawy pomiarowe w systemie CAMAC PTJ 6, 1973
- [11] K. Rzymkowski - Mikroprocesory scalone, rękopis PTJ 1978

mgr inż. JERZY SZYLLER
mgr inż. BARBARA KRZESAJ-JANYSZEK
Instytut Technologii Elektronowej

KONCEPCJA UNIWERSALNEGO, WIELOZADANIOWEGO MIKROPROCESOROWEGO SYSTEMU OPERACYJNEGO

Ośmioletni, gwałtowny rozwój techniki mikroprocesorowej spowodował znaczne rozwarstwienie architektury mikroprocesorów. Obecnie można określić dwa główne kierunki rozwoju architektury mikroprocesorów.

Jeden polegający na maksymalnym scaleniu modułów funkcjonalnych systemu mikroprocesorowego w postaci jednego mikroukładu /one-chip microcomputer/. Mikrokomputery jednoukładowe ze względu na ograniczoną wielkość pamięci programu i danych służą przede wszystkim do tworzenia prostych kontrolerów i systemów sterowania oraz do symulacji niezbyt złożonych układów logicznych.

Drugi kierunek polega na otaczaniu "silnego" mikroprocesora wieloma mikroukładami towarzyszącymi, takimi jak: pamięci, interfejsy, kontrolery, bufora i dekodery. Mikroprocesory te będziemy nadal nazywać mikroprocesorami ogólnego zastosowania. Stanowią one bazę dla konstruowania złożonych systemów sterowania wymagających pamięci o większej pojemności i wielu urządzeń wejścia/wyjścia.

W zakresie mikroprocesorów ogólnego zastosowania 8-bitowych tzn., posiadających wyróżnioną 8-bitową szynę danych, postępuje dalsze zwiększanie mocy przetwarzania, co znajduje swoje odbicie we wzbogacaniu listy rozkazów, sposobów adresowania i rozbudowie arytmometru. Zwiększanie "mocy" poszczególnych bloków funkcjonalnych mikroukładów jest tendencją naturalną wynikającą z dużych potencjalnych możliwości w zakresie gęstości upakowania w technologii MOS.

Wzrost gęstości upakowania układów LSI, powoduje, że architektura procesora centralnego staje się coraz bardziej skomplikowana, ewoluując w stronę procesorów 16-bitowych. Zatem możliwości systemów operacyjnych dla takiego sprzętu muszą proporcjonalnie wzrastać.

Obecnie można bez obawy popełnienia błędu stwierdzić, że "silne" mikroprocesory 8-bitowe mogą służyć do konstruowania systemów,

które z powodzeniem będą mogły wypierać niezbyt złożone systemy minikomputerowe wykonywane w tradycyjnej technice jak np. PDP 11/10 - 40, MITRA 15 itp.

Ponieważ w oparciu o wymienione i podobne typy mikrokomputerów tworzą uniwersalne systemy wielozadaniowe działające w reżimie czasu rzeczywistego /np. RSX 11 - S/M/, powstaje pytanie, czy jest możliwe i uzasadnione konstruowanie podobnych systemów w technice mikroprocesorowej. Niniejszy artykuł stanowi próbę odpowiedzi na to pytanie.

Gwałtowny spadek cen układów LSI przyniósł spodziewany wzrost zainteresowania systemami wieloprocessorowymi. System wielozadaniowy oparty o jeden procesor, kryje w sobie potencjalne możliwości rozproszenia funkcji sterujących na wiele procesorów. Wnikliwa analiza funkcji opracowywanego systemu operacyjnego da odpowiedź w jaki sposób przejść z systemu jednoprocessorowego do systemu z wieloma procesorami.

Wybór mikroprocesora i konfiguracji podstawowej

Aktualnie mikroprocesory ogólnego zastosowania stanowią największy procent produkowanych mikroukładów. Najpopularniejsze w tej grupie są mikroprocesory rodziny intelowskiej 8080, 8085, 6800 firmy "Motorola" oraz SC/MP firmy "National Semiconductor". Zdecydowanie najbardziej udaną konstrukcją, Z-80 oferuje firma "Zilog". Poza Z - 80, wszystkie wymienione mikroprocesory reprezentują podobne możliwości i z punktu widzenia tworzenia systemu operacyjnego jest prawie obojętne, który z nich zostanie wybrany. W przedstawionej koncepcji założono wykorzystanie systemu zbudowanego w oparciu o mikroprocesor 8080 firmy "Intel". Wybierając mikroprocesor 8080 kierowano się następującymi kryteriami:

- jest to mikroprocesor najpopularniejszy w naszym kraju;
- 8085 i Z - 80 są mikroprocesorami kompatybilnymi z 8080 w zakresie listy rozkazów;

- istnieją realne perspektywy produkcji odpowiednika mikroprocesora linii intelowskiej w krajach RWP.

Ze względu na możliwość realizacji w warunkach krajowych, jako konfigurację podstawową przyjęto:

- mikroprocesor 8080 wraz z układami 8224 i 8228
- pamięć do 64 K, ewentualnie układ ochrony pamięci proponowany w artykule "Wybrane zagadnienia ochrony i rozszerzenia pamięci w systemach mikroprocesorowych".
- układ przerwań oparty o kontroler przerwań 8214 lub w wersji bardziej rozbudowanej /64 poziomy przerwań/ zastąpiony dziewięcioma kontrolerami 8259.

Wymieniona konfiguracja jest wystarczająca do zrealizowania w pierwszej fazie podstawowych modułów systemu operacyjnego:

- dyrygenta zadań,
- modułu przerwań,
- modułu ekstrakodów systemowych.

W drugiej fazie polegającej na dołączeniu do systemu programów obsługi istniejących urządzeń wejścia/wyjścia oraz monitora zapewniającego współpracę systemu z operatorem, zostanie wykorzystana konfiguracja rozszerzona o:

- dalekopis,
- drukarkę,
- monitor ekranowy,
- czytnik/perferator,
- programator pamięci,
- zegar czasu rzeczywistego,
- inne urządzenia użytkownika.

W fazie trzeciej istniejący system operacyjny zostanie zmodyfikowany do postaci dyskowej w oparciu o dwie stacje dysków elastycznych.

Koncepcja systemu operacyjnego

Pojęcie zadania

Zadanie stanowi podstawową jednostkę programu, której działanie jest inicjowane i kontrolowane przez system operacyjny. Zadanie może być całkowicie niezależne od innych zadań w systemie lub może komunikować się z innymi zadaniami wykorzystując odpowiednio mechanizmy zawarte w podstawowych modułach systemu operacyjnego.

Każde zadanie może znajdować się w jednym z trzech stanów: WYKONYWANIA, CZEKANIA lub GOTOWOŚCI. Zadanie znajduje się w stanie WYKONYWANIA gdy zajmuje zasób procesora. Oczywiście w systemie jednoprosesorowym jest w danym momencie czasowym tylko jedno takie zadanie. W stanie CZEKANIA zadanie oczekuje na przydział zasobu. W stanie GOTOWOŚCI zadanie jest gotowe do zajęcia zasobu procesora.

Zadanie dzieli się na dwie części: tablicę stanu /TSZ/ oraz ciało zadania. W tablicy stanu zadania mieszczą się informacje o stanie w jakim ono się znajduje oraz opis parametrów charakteryzujących je. Tablica ma długość

64 bajty i nie jest bezpośrednio dostępna dla zadania. Ciało zadania może mieć maksymalną długość 48 K bajtów. W ciele każdego zadania programista może określić obszar przeznaczony na stos. Stos rozciąga się od adresu opisanego przez wskaźnik stosu, aż do adresu początku ciała zadania. Zadanie może odwoływać się poza przydzielony mu obszar pamięci do obszaru wspólnego, który znajduje się w ostatnich 8 K pamięci, dzielonego z innymi zadaniami.

Synchronizacja zadań

Synchronizacja dostępu zadań do zasobów systemu realizowana jest za pomocą dwóch podstawowych mechanizmów. Pierwszy polega na ustawianiu lub gaszeniu bitów warunków, drugi realizowany jest poprzez wykonywanie operacji P i V na semaforach licznikowych w sensie Dijkstry.

Realizację mechanizmu synchronizacji za pomocą warunków zapewnia istnienie czterech rejestrów w tablicy stanu zadania /TSZ/ oraz istnienie odpowiednich ekstrakodów działających na zawartości tych rejestrów. W TSZ każdego zadania określone są cztery rejestry 16-bitowe: rejestr warunków /RW/; rejestr deklaracji warunków znaczących /RD/; rejestr oczekiwania /RO/ oraz rejestr adresu TSZ /RA/. Każde zadanie ma możliwości ustawienia za pomocą ekstrakodu "Ustaw warunek" dowolnego warunku /ów/ w rejestrze warunków w swojej TSZ lub TSZ dowolnego zadania. Ustawienie określonego warunku w TSZ może być również spowodowane pojawieniem się zdarzenia zewnętrznego w postaci przerwania. Wykonywane zadanie ma możliwość sprawdzania w każdym momencie za pomocą ekstrakodu "Czekaj na warunek", czy określony /e/ warunek /ki/ RW wystąpił w dowolnej TSZ. Jeśli wystąpienie określonego warunku jest niezbędne do dalszej pracy zadania, wówczas w sytuacji gdy nie jest on spełniony, działanie zadania zostaje wstrzymane do momentu ustawienia warunku przez inne zadanie lub wskutek wystąpienia zdarzenia zewnętrznego.

Mechanizm synchronizacji za pomocą warunków jest niewystarczający gdy o dostęp do zasobu ubiega się nie jedno, lecz wiele zadań. Ten problem rozwiązuje zastosowanie semaforów licznikowych związanych z zasobami, do których dostęp ma być synchronizowany. Zasobami, które wymagają synchronizacji dostępu za pomocą semaforów są przede wszystkim: same zadania, informacja /zbiory/ oraz fizyczna pamięć. Z każdym tego typu zasobem związany jest semafor licznikowy składający się z minimum dwóch rejestrów: 8-bitowego rejestru licznika /RL/ oraz 8-bitowego rejestru nagłówka /RN/. Początkowy stan rejestru licznika dla zasobu wielokrotnego /patrz 3,4 pojęcie PULI/ określa ilość /n/ zasobów elementarnych /np. ogniów/ a dla zasobu jednostkowego /np. bufor w pamięci/ jest równy jed-

ności. Zawartość rejestru nagłówka wskazuje adres pierwszego zadania o najwyższym priorytecie czekającego na dostęp do zasobu. Zadania ubiegające się o dostęp do zasobu wykonują ekstrakod "Czekaj na zasób", który równoznaczny jest z wykonaniem operacji dijkstrowskiej P /semafor/. Gdy po odjęciu jednostki zawartość RI, jest mniejsza od zera zadanie ustawione jest w kolejce do zasobu i system wstrzymuje działanie zadania. Zadanie zwalniane wykorzystany zasób wykonuje ekstrakod "Zwolnij zasób", który odpowiada operacji dijkstrowskiej V /semafor/. Operacja ta polega na dodaniu jednostki do zawartości RI i sprawdzeniu czy jest mniejsza lub równa zeru. Jeśli tak, system przydziela zwalnianemu zasób pierwszemu, czekającemu w kolejce zadaniu.

Semafory synchronizujące dostęp do zadania znajdują się w TSZ, natomiast semaforów związanych z informacją lub pamięcią znajdują się w pierwszych lub ostatnich 8 K pamięci w tablicach opisujących zbiory informacji, buforów lub PULĘ.

Moduły monitora czasu rzeczywistego

Monitor czasu rzeczywistego składa się z trzech modułów: dyrygenta zadań, obsługi przerwania i realizacji ekstrakodów systemowych. W każdym z modułów zawarte są podstawowe mechanizmy umożliwiające synchronizację dostępu zadań do zasobów systemu. Mechanizmy te mogą być wykorzystane przez zadania do tworzenia wyższych poziomów funkcjonalnych w systemie operacyjnym /np. zarządzania zbiorami/. Moduły są w wysokim stopniu parametryzowalne, co pozwala w prosty sposób zmieniać podstawowe parametry systemu takie jak: kolejność zadań, ilość i rodzaj obsługi przerwania, czy też definiować lub usuwać ekstrakody lub semaforów.

Moduł dyrygenta zadań spełnia trzy podstawowe funkcje:

- zapamiętuje w ciele przerwanej zadania aktualną zawartość wszystkich rejestrów procesora 8080,
- według określonego algorytmu przeszukuje kolejkę zadań aby znaleźć zadanie, któremu może przydzielić zasób procesora,
- inicjuje działanie wybranego zadania.

Kolejka zadań ma wyróżniony początek. Zadania są uszeregowane w kolejce od najwyższego priorytetu do najniższego. W TSZ każdego zadania znajduje się adres TSZ zadania następnego w kolejce. Algorytm wyboru zadania, które ma zainicjować dyrygent jest bardzo prosty. Dyrygent sprawdza czy kolejne zadanie jest w stanie GOTOWOŚCI, jeśli tak inicjuje jego działanie. Jeśli zadanie jest w stanie CZEKANIA, dyrygent sprawdza, czy nie zostały spełnione warunki, na które czeka zadanie; jeśli tak inicjuje zadanie, jeśli nie bada następne.

Moduł dyrygenta podejmuje pracę w kilku

przypadkach, z których dwa zasługują na podkreślenie. Pierwszy, gdy zadanie musi poczekać na żądany zasób. Wtedy dyrygent rozpoczyna przeglądanie kolejki od zadania następnego po przerwaniu. Przypadek drugi, gdy wystąpi zdarzenie znaczące. Wtedy dyrygent rozpoczyna przeglądanie kolejki od wyróżnionego początku.

Modułem umożliwiającym reakcję systemu na zdarzenia zewnętrzne jest moduł obsługi przerwania. Moduł ten składa się z tabeli opisu przerwania /TOP/ oraz programów obsługi przerwania. Wielkość TOP jest zależna od typu kontrolerów przerwania jakie są użyte w systemie; dla kontrolera 8214 TOP zajmuje 64 kolejne bajty pamięci a dla dziewięciu kontrolerów 8259 aż 512 kolejnych bajtów pamięci. Charakterystyka każdego przerwania nieściśle się w TOP na ośmiu kolejnych bajtach. Każde przerwanie charakteryzują następujące parametry: adres programu obsługi przerwania, sposób obsługi przerwania, adres zadania związanego z przerwaniem, przyczyna przerwania, warunki opisujące czy przerwanie wystąpiło i czy jest znaczące. W czasie dołączania do systemu programu obsługi przerwania w odpowiedniej pozycji TOP umieszcza się parametry opisujące przerwanie. Najciekawszym parametrem jest sposób obsługi przerwania. Za pomocą tego parametru można określić w jaki sposób system będzie reagował na przerwanie. Podstawowymi sposobami reakcji na przerwanie, które zostaną zaimplementowane będą:

- obsługa natychmiastowa; polegająca na wykonaniu dowolnego programu obsługi przerwania
- realizacja polegająca na:
 - ustawieniu warunku w TOP opisującego, że wystąpiło przerwanie /bez przejścia do początku kolejki dyrygenta/
 - ustawieniu warunku w TSZ określonego zadania /bez przejścia do początku kolejki dyrygenta/
 - dla przerwania znaczącego, ustawieniu odpowiednich warunków w TOP lub TSZ, przerwaniu wykonywanego zadania i zainicjowania pracy dyrygenta od wyróżnionego początku kolejki zadań.

Moduł realizacji ekstrakodów systemowych składa się z trzech części: dekodera, tablicy adresów i programów realizujących ekstrakody. Wywołanie ekstrakodu systemowego polega na wykonaniu rozkazu przerwania programowego RST 2. W następnym bajcie pamięci za rozkazem mieści się identyfikator ekstrakodu oraz kod określający miejsce łącznika ekstrakodu, łącznik ekstrakodu może znajdować się bezpośrednio za identyfikatorem w kolejnych bajtach pamięci. Za identyfikatorem może mieścić się tylko adres miejsca pamięci, w którym znajduje się łącznik. Łącznik może również znajdować się na stosie zadania lub w rejestrach CPU. Ekstrakod systemowy może być również bezparametrowy.

Wykonując rozkaz RST 2 zadanie powoduje

wywołanie dekodera ekstrakodów, który określa na podstawie identyfikatora jaki ekstrakod ma być wykonany. W tablicy adresów umieszczone są rzeczywiste adresy programów realizujących ekstrakody. Tablica adresów jest wypełniana podczas dołączania ekstrakodów do systemu. Tablica ma długość 64 bajtów i umożliwia dołączenie do systemu 32 ekstrakodów. Po wykonaniu programu ekstrakodu następuje wykonanie następnego rozkazu za wywołaniem ekstrakodu lub w szczególnych przypadkach wymuszenie pracy dyrygenta.

Ekstrakody są programami, które realizują mechanizm synchronizacji dostępu zadań do zasobów na najniższym poziomie systemu operacyjnego, to znaczy rozszerzają możliwości maszyny "surowej". Podstawowymi funkcjami ekstrakodów systemowych są wszelkie operacje związane z warunkami, semaforami i tablicami systemowymi.

Mapa pamięci systemu mikroprocesorowego

Zakładając, że w systemie znajduje się pamięć o maksymalnej, bezpośrednio adresowanej pojemności 64 K, została ona podzielona na trzy obszary: zajmowany przez system operacyjny, zajmowany przez zadania użytkowe oraz wspólny, do którego mają dostęp wszystkie zadania i system. Prosta jednostka ochrony pamięci zapewnia ochronę obszaru pierwszych 8 K pamięci, w której znajduje się system przed bezpośrednim dostępem z obszaru aktualnie działającego zadania. Działające zadanie ma możliwość bezpośredniego adresowania wyłącznie obszaru zajmowanego przez swoje ciało oraz obszaru wspólnego, który mieści się na ostatnich 8 K pamięci.

W przypadku adresowania przez zadanie obszaru niedozwolonego, jednostka ochrony pamięci generuje przerwanie o wysokim priorytecie, co umożliwia reakcję systemu operacyjnego na to zdarzenie. Stos każdego zadania mieści się w początkowym obszarze ciała zadania. Przepędniawo stosu będzie więc automatycznie powodowało wygenerowanie przerwania związanego z przekroczeniem obszaru adresowania dozwolonego dla danego zadania.

W obszarze pamięci wspólnym dla zadań mieszczą się bufor i procedury wykorzystywane przez więcej niż jedno zadanie oraz tzw. PULA. PULA jest obszarem pamięci dzielnym na zwarte, 16-bajtowe segmenty dalej zwane ogniwami. W każdym ogniwie pierwszy bajt określa adres następnego. Zakłada się że na PULĘ przeznaczane jest 4 K pamięci, a więc wszystkich ogniw jest 256. Koncepcja PULI ogniw została zaimplementowana, aby zoptymalizować wykorzystanie pamięci przez zadania. Każde zadanie mające zapotrzebowanie na niewielki obszar pamięci np. w celu przesłania meldunku do innego zadania czy też parametrów procedurze, może zająć ogniwo i zwrócić je do PULI po wykorzystaniu.

Obsługa wejścia/wyjścia

Wiele systemów sterowania procesami prze-

mysłowymi wymaga wprowadzania i odczytu wyczerpującej informacji o stanie procesu. Oczywiście zadania użytkowe sterujące bezpośrednio procesami ze względu na ograniczenia czasowe i adresowe /nie dostępny cały obszar pamięci/ nie mogą zawierać obsługi urządzeń wejścia/wyjścia. Autonomiczną obsługę urządzeń wejścia/wyjścia zapewniają specjalne zadania systemowe.

Zadania wejścia/wyjścia umożliwiają pełną obsługę poszczególnych urządzeń lub grup urządzeń wejścia/wyjścia. Współpraca zadań zwykłych z zadaniami wejścia/wyjścia polega na przesłaniu do zadań wejścia/wyjścia informacji określających typ operacji jaką ma wykonać urządzenie oraz format danych podlegających operacji. Informacja jest przesyłana między zadaniami pod postacią określonego meldunku. Meldunki są ustawiane w priorytetowej kolejce do zadania wejścia/wyjścia, które kolejno je obsługuje. Gdy zadanie obsłuży ostatni meldunek z kolejki, zadanie wejścia/wyjścia przechodzi w stan oczekiwania. W chwili nadejścia pierwszego meldunku zadanie wejścia/wyjścia przechodzi w stan gotowości. Oczywiście powyższy schemat współpracy zadań jest znacznie uproszczony. W rzeczywistości, w zależności od treści meldunku, zwykle zadania mogą np. : czekać lub nie czekać na odpowiedź zadania wejścia/wyjścia o stanie transmisji.

Założono, że każde zadanie zwykle ma możliwość przesyłania meldunków wejścia/wyjścia do sześciu urządzeń logicznych różnych typów /patrz TSZ/. Przypisanie urządzeń logicznych, które wywołuje zadanie zwykle, urządzeniom fizycznym definiowane jest w TSZ i może być dynamicznie zmieniane. Po wykonaniu ekstrakodu wejścia/wyjścia system operacyjny . na podstawie informacji zawartej w TSZ przesyła meldunek wejścia/wyjścia do odpowiedniego zadania wejścia/wyjścia obsługującego fizyczne urządzenie wejścia/wyjścia. Pozwala to na całkowite uniezależnienie zadań użytkowych od obsługi fizycznych w/w urządzeń.

Szczupłe ramy niniejszego opracowania nie pozwalają na dokładny opis koncepcji. Podano jedynie sposoby realizacji podstawowych modułów systemu operacyjnego umożliwiających synchronizację dostępu zadań do zasobów systemu. Podstawowe moduły są w trakcie realizacji. Wydaje się, że projektowanie, pisanie, i uruchamianie takiego systemu jest celowe z wielu względów.

Podobny system został zrealizowany na potrzeby płockiej Petrochemii w oparciu o procesor 8-bitowy MOM - 1000. Analizując architekturę 8-bitowych mikroprocesorów okazało się, że daje ona znacznie większe możliwości konstruowania wielozadaniowego systemu operacyjnego niż procesor MOM - 1000. Zestawienie obu systemów da bogaty materiał porównawczy pozwalający opracować metodykę tworzenia złożonych systemów operacyjnych dla małych komputerów.

L i t e r a t u r a

- [1] P. B. Hansen - Operating System Principles Prentice Hall, Inc. 1972
- [2] S. Madnick, J. Donovan - Operating Systems Mc Graw Hill, 1978.
- [3] A. C. Schaw - The Logical Design of Operating Systems Prentice - Hall, Inc. 1974.
- [4] P. B. Hansen - Concurrent PASCAL a Programming Language for Operating System Design, April 1974.
- [5] N. Wirth - The programming language PASCAL Revised Report, Nov. 1972.
- [6] J. B. Peatman - Microcomputer Based Design Mc Graw Hill, 1977.
- [7] B. Souček - Microprocessors and Microcomputers John Wiley and Sons, 1976.
- [8] Materiały - Second Symposium on Microprocessing and Microprogramming, 1976, Venice.
- [9] Electronics - Microprocessors, April 15, 1976.
- [10] Proceedings of the IEEE, Vol. 64, no. 6, June 1976.
- [11] Intellec MDS Reference Manuals 1976.
- [12] Intel 8080, 8085, 8048 Microcomputers Systems User's Manuals.
- [13] J. Szyller - Architektura Mikroukładów Procesora Centralnego /artykuł opublikowany w materiałach z konferencji "Mikroprocesory - budowa i możliwości zastosowań" Katowice, 1978/.
- [14] Praca zbiorowa - Języki Projektowania Systemów Operacyjnych. Politechnika Warszawska, 1977.



inż. BOGUSŁAW JACKIEWICZ
Przemysłowy Instytut Elektroniki

WSPÓŁPRACA PRZETWORNIKÓW ANALOGOWO-CYFROWYCH Z SYSTEMAMI MIKROPROCESOROWYMI

W większości współcześnie używanych elektronicznych przyrządów pomiarowych nośnikami wartości mierzonej są analogowe sygnały napięciowe. Z tego też względu systemy mikroprocesorowe stosowane w elektronicznej aparaturze pomiarowej z zasady współpracują z przetwornikami analogowo-cyfrowymi, stanowiącymi dla tych systemów podstawowe źródło informacji o wartości mierzonej.

Systemy mikroprocesorowe mogą współpracować zarówno z przetwornikami integracyjnymi jak i z kompensacyjnymi. Przetworniki integracyjne zapewniają możliwość uzyskania znacznej rozdzielczości przy stosunkowo niewielkiej prędkości działania. Dlatego są stosowane głównie w laboratoryjnych i przenośnych woltomierzach cyfrowych, w których zasadnicze znaczenie mają rozdzielczość i do-

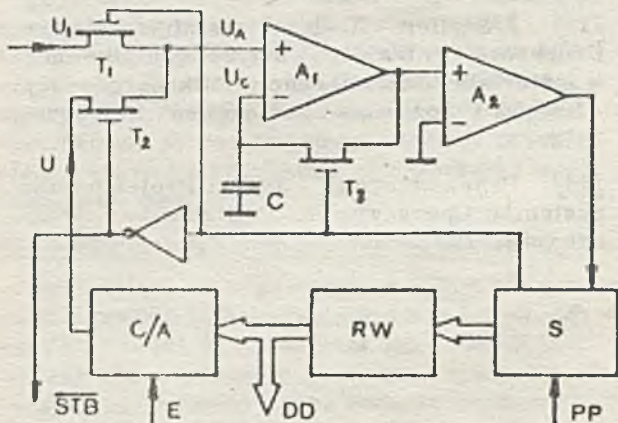
kładność pomiaru, natomiast czas przetwarzania nie odgrywa większej roli. W pozostałych przypadkach z reguły stosowane są przetworniki kompensacyjne, działające na zasadzie porównywania mierzonego sygnału z napięciem wzorcowym, wytwarzanym za pomocą specjalnych dzielników rezystorowych, sterowanych sygnałami cyfrowymi [1].

Podstawową zaletą przetworników kompensacyjnych jest duża szybkość działania, sięgająca 100000 konwersji na sekundę. Przetworniki kompensacyjne są często używane w połączeniu z układami próbkująco-pamiętającymi, zapewniającymi możliwość przetwarzania wartości chwilowych mierzonego przebiegu. Połączenie przetworników kompensacyjnych z systemami mikroprocesorowymi zapewniająco- mi możliwość cyfrowej obróbki sygnałów po-

miarowych, pozwala realizować skuteczną eliminację zakłóceń i błędów systematycznych.

Przetwornik z układem sterującym

Zasadę działania typowego kompensacyjnego przetwornika analogowo-cyfrowego z prostym układem próbkująco-pamiętającym przedstawiono na rys. 1. W stanie spoczynkowym klucze tranzystorowe T_1 i T_3 są otwarte, a klucz T_2 zamknięty, powodując ładowanie kondensatora C do napięcia prawie równego napięciu



Rys. 1. Przetwornik analogowo-cyfrowy z prostym układem próbkująco-pamiętającym. C/A - rezystorowy dzielnik cyfrowy, E - napięcie wzorcowe, RW - rejestr wyniku przetwarzania, S - układ sterujący, A_1 - wzmacniacz różnicowy, A_2 - komparator, T_1 , T_2 , T_3 - klucze tranzystorowe.

wejściowemu. Sygnał rozpoczynający cykl przetwarzania powoduje wyłączenie kluczy T_1 i T_3 i włączenie klucza T_2 /rys. 2/. Komparator A_2 jest sterowany sygnałem, stanowiącym wzmożoną różnicę napięcia dzielnika cyfrowego C/A i zapamiętanego na kondensatorze C napięcia wejściowego. Stan wyjścia komparatora zależy od znaku tej różnicy. W przypadku $U_N < U_1$ wyjście komparatora znajduje się w stanie "0", natomiast przy $U_N > U_1$ przyjmuje stan "1".

Dzielnik cyfrowy składa się z zespołu dokładnych rezystorów, dołączanych do napięcia wzorcowego za pomocą kluczy tranzystorowych. Dzielnik cyfrowy wraz z wbudowanym źródłem napięcia wzorcowego zazwyczaj tworzy jednolity podzespół, nazywany przetwornikiem cyfrowo-analogowym.

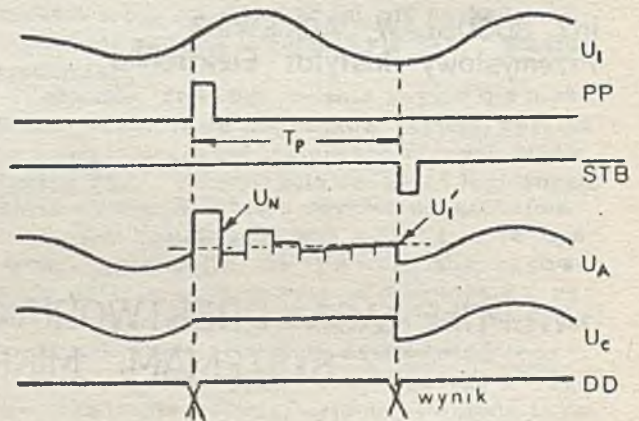
W układzie przedstawionym na rys. 1 dzielnik cyfrowy C/A jest połączony z rejestrem wyników przetwarzania RW, złożonym z przerzutników typu D. Rejestr jest sterowany sygnałami układu sterującego S, wymuszającego włączanie kolejnych bitów dzielnika cyfrowego. Włączanie rozpoczyna się od najbardziej zna-

czącego bitu i następuje w kolejności malejących wag aż do najmniej znaczącego bitu dzielnika. Po włączeniu każdego kolejnego bitu i usłabieniu się procesorów przejściowych następuje odczytanie stanu komparatora i podjęcie decyzji o pozostawieniu na stałe lub wyłączeniu danego bitu.

Po zakończeniu pełnego cyklu pracy przetwornika napięcie dzielnika cyfrowego jest z dokładnością do najmniej znaczącego bitu równe wartości sygnału wejściowego, a stan rejestru RW reprezentuje wynik przetwarzania.

Rejestr wyników przetwarzania i układ sterujący opisanego przetwornika mogą być zrealizowane w oparciu o typowe układy scalone serii TTL. W przetworniku wchodzącym w

Rejestr wyników przetwarzania i układ sterujący opisanego przetwornika mogą być zrealizowane w oparciu o typowe układy scalone serii TTL. W przetworniku wchodzącym w skład mikroprocesorowego woltomierza próbkującego opracowanego w Przemysłowym Instytucie Elektroniki zastosowano 12-bitowy przetwornik cyfrowo-analogowy - wytwarzany seryjnie w PIE technologią hybrydową. Rejestr wyników przetwarzania złożony z 6 podwójnych przerzutników MCY 7474N sterowany jest z układu

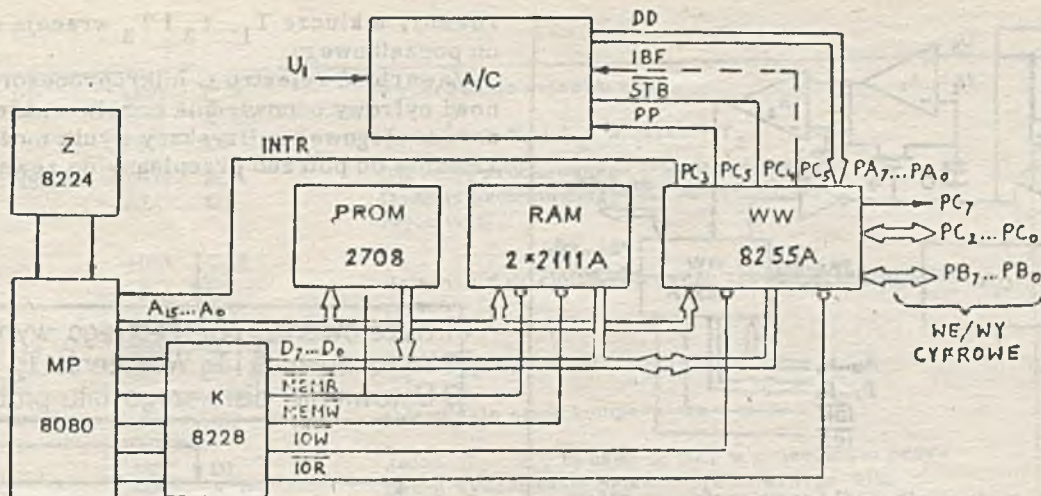


Rys. 2. Przebiegi czasowe w przetworniku przedstawionym na rys. 1. U_1 - napięcie wejściowe. PP - sygnał rozpoczynający przetwarzanie, STB - sygnał zakończenia przetwarzania, DD - wynik przetwarzania.

Czas trwania cyklu przetwarzania przetwornika wyznaczony jest częstotliwością generatora zegarowego współpracującego systemu mikroprocesorowego i w opisywanym rozwiązaniu wynosi 240 us.

Połączenie przetwornika z systemem mikroprocesorowym

Przetwornik przedstawiony na rys. 1 przystosowany jest do współpracy z systemem mikroprocesorowym z rys. 3. Część centralna systemu składa się z trzech układów scalonych LSI:

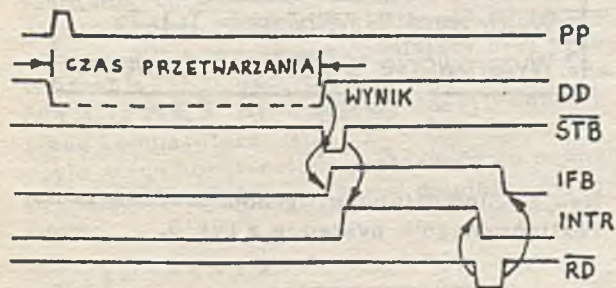


Rys. 3. Połączenie przetwornika analogowo-cyfrowego z systemem mikroprocesorowym 8080

mikroprocesora 8080A, kontrolera systemu 8228 i generatora impulsów zegarowych 8224. Układy PROM typu 2708 i RAM typu 2111A pełnią funkcję pamięci systemu. Wymiana sygnałów pomiędzy systemem mikroprocesorowym i pozostałymi układami przyrządu odbywa się za pośrednictwem zespołu programowanych, trójstanowych bramek wejściowo-wyjściowych /WW/ wykonanych w postaci specjalnego układu scalonego LSI-MOS.

Współpraca z przetwornikiem zajmuje 12 z 24 bramek układu wejściowo-wyjściowego typu 8255A. Bramki $PA_7...PA_0$ wykorzystywane są jako wprowadzające dane z przetwornika do systemu mikroprocesorowego. Bramka PC_3 działa jako wyjście sygnału PP inicjującego cykl przetwarzania. Stan wyjścia tej bramki jest sterowany za pomocą instrukcji "BIT SET" i "BIT RESET" wysyłanych programowo na adres rejestru sterującego układu 8255A.

Zależności czasowe sygnałów decydujących o współpracy przetwornika z systemem mikroprocesorowym przedstawiono na rys. 4. Zakończenie procesu przetwarzania jest sygnalizowane za pomocą zanegowanego impulsu \overline{STB} . Bramki $PA_7...PA_0$ są programowane jako tzw. wejścia strobowane, do których informacja jest wpisywana podczas ujemnego zbocza impulsu \overline{STB} . Dodatkowo zbocze tego impulsu



Rys. 4. Przebiegi czasowe w systemie z rys. 3.

wprowadza bramkę wyjściową PC_3 w stan "1", tworząc sygnał INTR wykorzystywany do przerywania dotychczas wykonanego programu w celu przyjęcia danych z przetwornika.

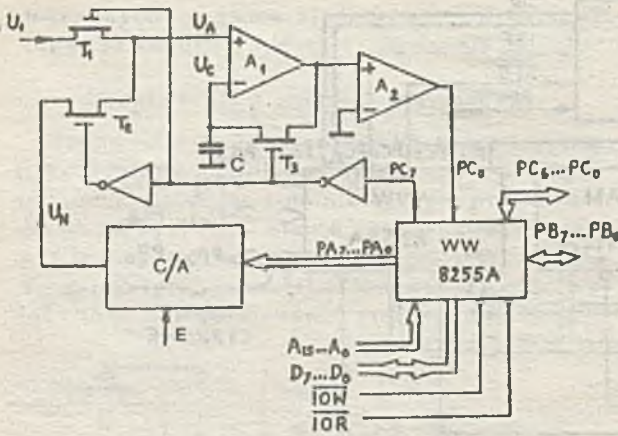
Sygnał IFB występujący na wyjściu PC_5 układu 8255A oznacza zajętość wewnętrznego rejestru bramek wejściowych $PA_7...PA_0$. Sygnał ten może być wykorzystywany do blokady uruchomienia nowego cyklu pracy przetwornika przed odczytaniem poprzednich wyników. Sygnał IFB jest automatycznie zerowany impulsami RD, sygnalizującymi odczytanie informacji zawartej w rejestrze układu wejściowo-wyjściowego 8255A.

Opisany przetwornik jest zupełnie samodzielnym układem, współpracującym z systemem mikroprocesorowym jedynie za pomocą sygnałów rozpoczynających cykle przetwarzania i sygnałów gotowości wyników. Zaletą takiego rozwiązania jest możliwość wykorzystywania mikroprocesora do wykonywania innych zadań podczas pracy przetwornika. Wadą jest konieczność budowy specjalnego układu sterującego, wymagającego stosowania szeregu dodatkowych podzespołów.

Programowa realizacja przetwornika

Innym rozwiązaniem omawianego problemu jest wykorzystanie systemu mikroprocesorowego do programowego sterowania cyklem pracy przetwornika analogowo-cyfrowego. W tym przypadku realizacja przetwarzania nie wymaga stosowania żadnych dodatkowych układów sterujących, a jedynymi niezbędnymi podzespołami są przetwornik cyfrowo-analogowy i komparator.

Przykład realizacji 8 bitowego przetwornika analogowo-cyfrowego sterowanego programowo za pomocą systemu mikroprocesorowego 8080 ilustruje rys. 5. Dzielnik cyfrowy sterowany jest sygnałami szyny danych systemu za pośrednictwem bramek $PA_7...PA_0$ układu wejściowo-wyjściowego 8255A. Klucze T_1, T_2



Rys. 5. Przetwornik analogowo-cyfrowy sterowany programowo za pomocą systemu mikroprocesorowego.

T_3 są połączone z bramką PC_7 , natomiast sygnały komparatora są wprowadzane za pomocą bramki PC_0 .

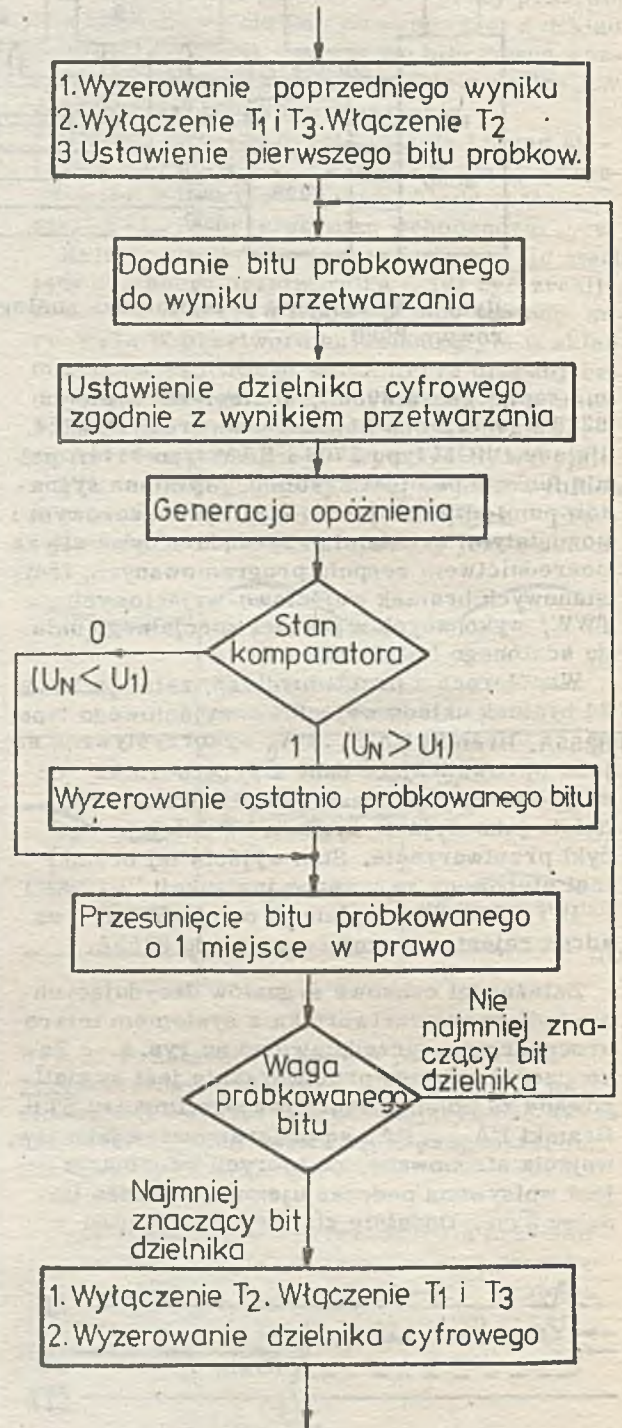
Sieć działań przetwornika i jego program sterujący przedstawiono na rys. 6 i 7. Dane sterujące procesem przetwarzania przechowywane są w dwóch rejestrach mikroprocesora. W rejestrze B pamiętany jest aktualnie próbkowany bit, natomiast rejestr C zawiera wynik dotychczas wykonanych operacji przetwarzania.

Nowy cykl pracy przetwornika rozpoczyna się od włączenia kluczy T_1 i T_3 , wyłączeniu klucza T_2 i wpisaniu najbardziej znaczącego bitu /tj. liczby 08 w układzie szesnastkowym/ do rejestrów B i C. Wysłanie zawartości rejestru C na adres bramek wyjściowych $PA_7...PA_0$ powoduje włączenie najbardziej znaczącego bitu dzielnika cyfrowego. Po odczekaniu czasu niezbędnego dla zakończenia procesów przejściowych, dokonywany jest odczyt stanu komparatora. Gdy przetwarzany sygnał jest większy od aktualnego napięcia wyjściowego dzielnika, stan komparatora jest zerowy i zawartość rejestru C /wynik przetwarzania/ pozostaje niezmienną; w przeciwnym przypadku próbkowany bit rejestru C jest zerowany.

Przesunięcie zawartości rejestru B o jedno miejsce w prawo /tj. w kierunku mniej znaczących bitów/ przygotowuje mikroprocesor do próbkowania następnego bitu, co jest wykonywane przez dodanie zawartości rejestru B do zawartości rejestru C i wysłania tej ostatniej na adres dzielnika cyfrowego. Pętla programu sterująca próbkowaniem bitów jest powtarzana aż do wystąpienia przeniesienia po kolejnym przesunięciu zawartości rejestru B w prawo. Ustawienie bitu przeniesienia w stanie 1 powoduje wykonanie instrukcji, kończących cykl pracy przetwornika. Dzielnik cyfrowy jest ze-

rowany, a klucze T_1 , T_2 i T_3 wracają do stanu początkowego.

Zawartość rejestru C mikroprocesora stanowi cyfrowy odpowiednik przetwarzanego sygnału analogowego. Uzyskany wynik może być zależnie od potrzeb przepisany do zewnętrznej

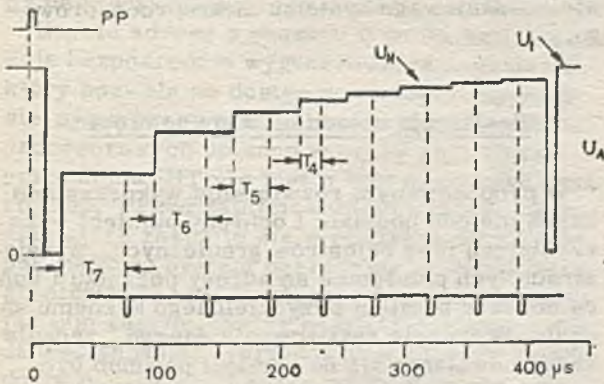


Rys. 6. Sieć działań programu przetwarzania, realizowanego w systemie z rys. 5.

1/ Inny przykład programu sterującego podano w [2].

PP:	MVI C, 0	Wyzerowanie rejestru C
	MVI A, 80	
	OUT CC	Wyłączenie T_1 i T_3 , włączenie T_2
NS:	MOV B, A	Wpisanie próbkowanego bitu do rejestru C
	ADD C	Dodanie próbkowanego bitu do zawartości rejestru C
	MOV C, A	
	OUT CA	Ustawienie dzielnika cyfrowego zgodnie z zawartością rejestru C
DW:	MOV A, B	Generacja opóźnienia
	ADI EO	
	JC DW	
	IN CC	Odczytanie stanu komparatora
	RCC	
	JNC DL	Jeżeli $U_N = U_1$, to skok do DL, w przeciwnym przypadku wyzerowanie ostatnio próbkowanego bitu rejestru C
	MOV A, B	
	XRA C	
	MOV C, A	
DL:	MOV A, B	Przesunięcie zawartości rejestru B w prawo
	RRC	Jeżeli próbkowany bit był najmniej znaczącym bitem dzielnika cyfrowego, to zakończenie przetwarzania.
	JNC NS	W przeciwnym przypadku powtórzenie wykonania pętli programu
	MVI A, 0	
	OUT CA	Wyzerowanie dzielnika cyfrowego
	OUT CC	Włączenie T_1 i T_3 , wyłączenie T_2
	RET	

Rys. 7. Program przetwarzania analogowo-cyfrowego, zapisany w języku symbolicznym mikroprocesora 8080. CA adres zespołu bramek PA, połączonych z wejściami sterującymi dzielnika cyfrowo-analogowego, CC adres bramek PC, łączących system mikroprocesorowy z wyjściem komparatora i z wejściem sterującym kluczami T_1 , T_2 , T_3 .



Rys. 8. Przebiegi czasowe sygnałów przetwornika analogowo-cyfrowego sterowanego systemem mikroprocesorowym wg programu przedstawionego na rys. 7 / Częstotliwość zegarowa 2MHz/. PP - sygnał uruchamiający cykl przetwarzania, U_A - napięcie wejściowe wzmacniacza Λ , z rys. 5. RD - sygnały odczytywania stanu komparatora, T_1, \dots, T_4 - opóźnienia odczytu stanu komparatora w stosunku do momentów włączenia kolejnych bitów dzielnika cyfrowego.

pamięci lub używany do następnych operacji przewidzianych programem. Przebiegi czasowe sygnałów występujących podczas procesu przetwarzania przedstawiono na rys. 8. Poważną zaletą opisywanego sposobu sterowania stanowi łatwość generacji opóźnień, niezbędnych dla zapewnienia poprawności odczytu stanów komparatora po włączeniu kolejnych bitów dzielnika cyfrowego. Z zasady działania przetwornika kompensacyjnego wynika, że wymagane opóźnienia zależą od wag próbkowanych bitów; największe opóźnienie jest potrzebne po włączeniu pierwszego, najbardziej znaczącego bitu.

W przypadku programu przedstawionego na rys. 7 opóźnienie jest generowane za pomocą wielokrotnego dodawania liczby EO_{16} do zawartości rejestru B mikroprocesora. Przy częstotliwości zegara systemu mikroprocesorowego równej 2MHz powoduje to generację opóźnienia 50 us po włączeniu bitu D_7 , 33 us po D_6 ,

20 us po D_5 i 11 us po pozostałych bitach. Zmiana dodawanej liczby pozwala w szerokich granicach regulować wprowadzane opóźnienia.

Literatura

[1] D. Fullagar, P. Bradshaw, B. O'Neill - interfacing data converters and microprocessors, Electronics, Dec. 9, 1976.

[2] A. Mrozowski - Analog output chips shrink a-d conversion software. Electronics, June 23, 1977.

WYBRANE ZAGADNIENIA OCHRONY PAMIĘCI W WIELOZADANIOWYCH SYSTEMACH MIKROPROCESOROWYCH

Charakterystyczną cechą systemów wielozadaniowych jest to, iż w pamięci operacyjnej znajduje się jednocześnie, obok systemu operacyjnego, wiele zadań realizowanych w określonej kolejności pod nadzorem tego systemu. W przypadku prawidłowego podziału obszaru pamięci pomiędzy poszczególne zadania i przy braku błędów w programach znajdujących się w pamięci, system powinien działać poprawnie. Nie można jednak założyć, że prawdopodobieństwo wystąpienia błędu w części adresowej rozkazów jest równe zeru. W rzeczywistych warunkach różne przypadkowe błędy występujące w procesie adresowania pamięci mogą powodować niekontrolowaną ingerencję jednego zadania w obszar pamięci przeznaczony dla innych zadań, co może zniszczyć zapisane w tym obszarze informacje.

Wyeliminowanie możliwości zniszczenia jednego programu przez inny osiąga się stosując odpowiednie systemy ochrony pamięci. W ogólnym przypadku pod pojęciem systemu ochrony pamięci nazywanego niekiedy systemem ochrony adresów pamięci [8], rozumie się środki układowo-programowe uniemożliwiające wzajemną interferencję programów znajdujących się jednocześnie w pamięci.

Systemy ochrony z reguły działają na adresach fizycznych, chociaż jest możliwa organizacja ochrony pamięci również na poziomie adresów wirtualnych. Najbardziej rozpowszechnione są dwa systemy ochrony pamięci: z wykorzystaniem masek oraz przez stosowanie kluczy [8].

W systemach mikroprocesorowych najczęściej spotyka się reżim pracy jednozadaniowy, lub wielozadaniowy, ale o stosunkowo niewielkiej liczbie zadań. W pierwszym przypadku programy sterujące pracą systemu zapisywane są w pamięci stałej typu ROM, co eliminuje możliwość ich zniszczenia. W systemach wielozadaniowych, w których większość programów zapisana jest w pamięci typu zapis/odczyt

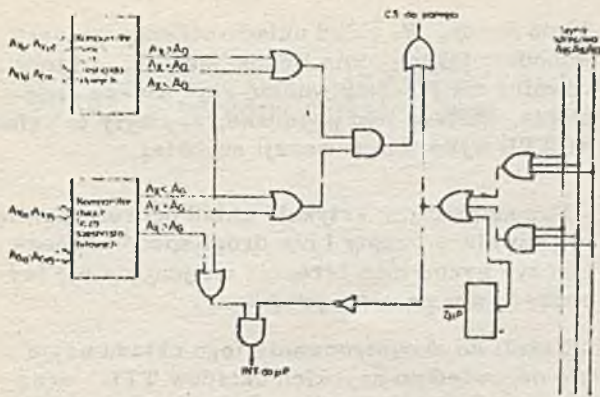
istnieje konieczność zapewnienia ochrony pamięci, gdyż w pamięci tego typu /zwanej pamięcią RAM/ można zniszczyć zapisaną informację poprzez zapis pod błędnie podanym adresem. W systemie mikroprocesorowym zabezpieczenie ochrony pamięci powinno być jednak stosunkowo proste, gdyż stosowanie rozbudowanych i drogich systemów ochrony stosowanych w dużych systemach komputerowych, może okazać się po prostu nieopłacalne w porównaniu z kosztami całego systemu mikroprocesorowego, którego podstawową zaletą powinien być niski koszt.

W niniejszym artykule przedstawiono realizację takiego systemu ochrony pamięci dla wielozadaniowego systemu mikroprocesorowego.

Realizacja układu ochrony pamięci

W proponowanym rozwiązaniu wykorzystano znaną metodę podziału i ochrony pamięci wykorzystującą ideę rejestrów granicznych. W rejestrach tych pamiętane są adresy początku i końca obszaru pamięci przydzielonego każdemu zadaniu. W trakcie realizowania danego zadania każde odwołanie się do pamięci powinno być poprzedzone sprawdzeniem, czy pojawiający się adres należy do obszaru przydzielonego danemu zadaniu. Osiąga się to porównując każdy pojawiający się adres z adresami granicznymi zapisanymi w rejestrach granicznych układu odmowy. Próba kontaktu zadania z obszarem niezarezerwowanym powinna powodować przerwanie wykonywanego programu i odpowiednią reakcję systemu operacyjnego. Na rys. 1 przedstawiono schemat blokowy prostego układu realizującego ochronę pamięci w systemie mikroprocesorowym, którego zasada działania oparta jest o wyżej opisaną metodę.

Głównym układem przedstawionej realizacji technicznej jest komparator porównujący szesnastobitowe liczby. W komparatorze będą po-



Rys. 1. Schemat blokowy układu ochrony pamięci

równywane adresy graniczne przydzielone każdemu zadaniu przez system operacyjny z bieżącym adresem znajdującym się na szynie adresowej.

Założmy, że pamięć systemu została podzielona w następujący sposób:

- pierwsze 8K bajtów pamięci /adresy od 0 do 8K/ zajmują programy systemu operacyjnego,
- kolejny obszar pamięci /od 8K do 56K bajtów/ przeznaczony jest dla zadań,
- ostatnie 8K bajtów pamięci /od 56K do 64 K bajtów/ stanowi obszar wspólny, który może być wykorzystywany przez wszystkie programy, niezależnie od przydzielonych im granic adresowych /umożliwia to wykorzystywanie przez różne zadania wspólnych danych i procedur/. Mapę pamięci opisywanego systemu ilustruje rys. 2.

Użycie adresu z obszaru 0 do 8K bajtów powoduje bezpośrednio wygenerowanie sygnału CS, który pozwala na dostęp do pamięci /sygnały sterujące pracą pamięci w systemach mikroprocesorowych opisano w pracy [5]. Sygnał przerwania INT nie jest w tym przypadku generowany. Przed zainicjowaniem każdego zadania, które znajduje się w obszarze od 8K do 56K bajtów, system operacyjny przesyła adresy graniczne do układu ochrony pamięci określając tym samym obszar pamięci dostępny dla danego zadania. Wartości tych adresów /adres dolny $A_{D0}-A_{D15}$ i adres górny $A_{G0}-A_{G15}$ / mogą być umieszczone w czterech ośmiobitowych portach /np. układy 8212/. Przesyłanie tych adresów do portów będzie się odbywało w opisywanej konfiguracji 8-bitową szyną danych, tak że każdy adres będzie przesyłany w dwóch kolejnych bajtach.

Przy każdym odwołaniu się do adresu z obszaru 8K do 56K bajtów nastąpi porównanie podanego adresu / $A_{X0}-A_{X15}$ / z adresami granicznymi przechowywanymi w układzie ochrony pamięci. Jeżeli podany adres będzie się zawierał w wyznaczonych granicach /tzn. będzie się znajdował w przydzielonym aktualnie działającemu zadaniu granicach/, to zostanie wygenerowany sygnał CS. Sygnał ten pozwoli na do-

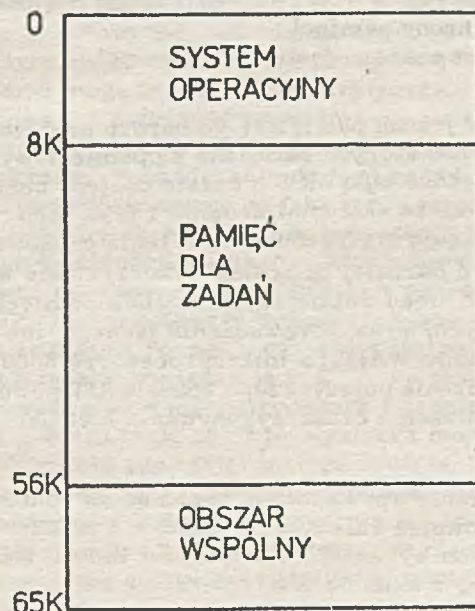
stęp do wskazanego przez adres miejsca pamięci.

Jeżeli podany adres nie będzie się mieścił w wyznaczonych granicach, to zostanie wygenerowany sygnał INT, który spowoduje przerwanie wykonywanego zadania i przekazanie sterowania systemowi operacyjnemu. Tak więc, każde odwołanie się do pamięci poprzez adres wykraczający poza wyznaczone granice, będzie powodowało przerwanie wykonywanego zadania, co uniemożliwi wykonanie operacji w nieprzydzielonym zadaniu obszarze /oczywiście głównie niepożądanym jest zapis/, zabezpieczając tym samym zapisane tam informacje. Podanie adresu z obszaru ostatnich 8K bajtów pamięci /powyżej 56K bajtów/ zawsze powoduje generowanie sygnału CS i niepojawienie się sygnału INT. Ta część pamięci nie będzie chroniona żadnymi specjalnymi środkami. Pierwsze 8K bajtów również nie będzie chronione przed dostępem z zadań, lecz w przypadku zapisania systemu operacyjnego do pamięci typu ROM nie będzie zachodziła potrzeba chronienia tego obszaru.

W opisywanym układzie zastosowano również prosty mechanizm, który pozwala na dostęp do wszystkich miejsc pamięci dla systemu operacyjnego z pominięciem układu ochrony. Funkcję tę spełnia jednobitowy wskaźnik /flag/, którego zapalenie będzie umożliwiało systemowi operacyjnemu dostęp do pamięci po podaniu każdego adresu z obszaru 0 do 64K bajtów.

Uwarunkowania czasowe

W systemach przeznaczonych do sterowania bardzo ważnym parametrem jest wymagany czas reakcji systemu na określone zdarzenia.



Rys. 2. Mapa pamięci

Wyżej opisany układ należy więc tak zaprojektować, aby nie wprowadzał on zbyt dużych opóźnień w pracy całego systemu.

W systemach, w których wykorzystano mikroprocesor 8080 firmy "Intel" czas cyklu zegarowego wynosi 500 ns. Maksymalną szybkość pracy mikroprocesora determinuje m. in. czas dostępu do pamięci tzn. czas, który upływa od podania adresu do pojawienia się danych na wyjściu pamięci. Jeżeli czas dostępu do pamięci jest większy od czasu cyklu zegarowego mikroprocesora, to procesor nie może pracować z maksymalną szybkością i musi wchodzić w stan, w którym oczekuje na odpowiedź z pamięci /stan ten zwykle nazywa się WAIT/. Aby mikroprocesor 8080 mógł pracować bez stanu WAIT. czas dostępu do wszystkich pamięci znajdujących się w systemie musi być rzędu 500 ns. /dla nowszych typów pamięci wynosi on przeciętnie 450 ns/.

Stosując dodatkowy układ jakim jest komparator, powoduje się oczywiście wydłużenie czasu dostępu do pamięci o czas potrzebny na porównanie adresów. Jeżeli opóźnienie wprowadzane przez układ ochrony pamięci będzie stosunkowo niewielkie, wtedy stosując odpowiednio szybkie pamięci w systemie /np. o czasie dostępu rzędu 300 ns/, można uzyskać rozwiązanie, które nie będzie spowolniało systemu w zauważalny sposób tzn. mikroprocesor będzie mógł pracować bez dodatkowych stanów oczekiwania na pamięć. Dodatkowy stan WAIT nie będzie więc potrzebny po wprowadzeniu układu ochrony pamięci, jeżeli

$$t_{\text{kom.}} + t_{\text{ACC}} \leq t_{\text{cyklu zeg. mikroproc.}}$$

gdzie:

$t_{\text{komp.}}$ jest opóźnieniem wprowadzanym przez układ ochrony pamięci,
 t_{ACC} jest czasem dostępu do pamięci.

Należy jednak pamiętać, że bardzo szybkie pamięci /do których zaliczane są pamięci wykonane technologią MOS o czasie dostępu poniżej 300 ns/ są układami drogimi i jeśli tego nie wymagają krytyczne ograniczenia czasowe może być bardziej opłacalne wykorzystanie w systemie nieco wolniejszych układów pamięci /i tańszych/ oraz wprowadzenie jednego lub kilku stanów WAIT /w mikroprocesorze 8080 wprowadzenie pojedynczego stanu WAIT powoduje wydłużenie czasu wykonywania instrukcji o 500 ns/.

Opóźnienie wprowadzane przez układ ochrony będzie również zależało od układów z jakich zostanie on wykonany. Komparator można zbudować np. z układów SN 7485, które zależnie od typu będą wprowadzały opóźnienie od 18 do

38 ns każdy. W skład układu ochrony musi wchodzić także pewna liczba bramek, które również nie powinny wносить zbyt dużego opóźnienia, dlatego jest pożądane, aby były to układy TTL wykonane w wersji szybkiej.

- Proponowany w artykule układ ochrony pamięci pozwala w prosty i nie drogi sposób zabezpieczyć przed interferencją znajdujące się jednocześnie w pamięci programy.

- Jeżeli do skonstruowania tego układu użyje się odpowiednio szybkich układów TTL oraz pamięci o właściwie dopasowanych czasach dostępu, to wprowadzenie tego układu do konfiguracji systemu nie zmniejszy szybkości systemu.

- Układ ochrony pamięci w przedstawionej wersji nie zapewnia ochrony obszaru, w którym znajduje się system operacyjny. W przypadku zapisania programów systemu operacyjnego do pamięci typu ROM /w której możliwy jest tylko odczyt/, ochrona pamięci nie jest potrzebna. Jeżeli system operacyjny zostanie wpisany do pamięci typu RAM /do której można również wpisywać/ i ochrona byłaby pożądana, to przedstawiony układ można zmodyfikować w stosunkowo prosty sposób, jednak będzie to wymagało dekodowania informacji związanej z cyklem maszynowym mikroprocesora.

L i t e r a t u r a :

[1] H. Raphael: How to expand a microcomputer's memory. Electronics December 23, 1973 p. 67-68

[2] S. Hoener, W. Roehder: Modular multi-microprocessor architecture with virtual memory. Second Symposium on Micro Architecture, Euromicro 1976

[3] Intel 8080, 8085, 8048 Microcomputer Systems User's Manual

[4] B. Hansen: Operating System Principles. Practice Hall Inc. 1972

[5] B. Krzesaj-Janyszek: Zagadnienia współpracy mikroprocesorów z pamięciami - artykuł opublikowany w Materiałach z konferencji "Mikroprocesory - budowa i zastosowanie" Katowice 1978

[6] J. Szyller, B. Krzesaj. Koncepcja uniwersalnego, wielozadaniowego, mikroprocesorowego systemu operacyjnego ukaże się w Biuletynie "Mera" nr 8/197.

[7] A. J. T. Colin: Wstęp do systemów operacyjnych. PWN Warszawa 1976

[8] E. A. Drozdow, W. A. Komarnickij, A. P. Piatibratow: Mnogoprogramnyje cifrowyje wyzislitelnyje masziny. OTKE Moskwa 1974



POMIARY I BADANIA UKŁADÓW MIKROPROCESOROWYCH

Zagadnienia ogólne testowania układów mikroprocesorowych

Układy scalone tworzące systemy mikroprocesorowe, a więc jednostka centralna, pamięci stałe ROM i pamięci aktywne RAM oraz układy wejścia/wyjścia lub inne jeszcze układy pomocnicze wymagają odrębnego traktowania podczas opracowywania metodyki ich testowania i pomiarów. Wykorzystuje się przy tym z jednej strony doświadczenia z badań i pomiarów różnych cyfrowych układów scalonych MSI i LSI a przede wszystkim pamięci i rejestrów, z drugiej zaś doświadczenia z badań i testowania pakietów. Ten ostatni sposób podejścia znajduje zastosowanie do samych mikroprocesorów, które będąc oddzielnymi układami scalonymi są pod względem złożoności i liczby realizowanych funkcji - równoważne dużym blokom czy modułom funkcjonalnym.

Wewnętrzna złożoność mikroprocesorów powoduje, że winny być one testowane nie tylko u wytwórcy lecz także u ich użytkowników, w zasadzie przed zamontowaniem na płycie drukowanej. Z praktyki amerykańskich firm wynika bowiem, że użytkownicy odrzucają ok. 1 + 5% dostarczanych im mikroprocesorów, jako nie spełniających postawionych wymagań [7]. Sytuacja taka ma miejsce mimo, iż każdy układ scalony jest mierzony w zakładzie wytwórczym co najmniej dwukrotnie:

- ostrzowo, gdy wszystkie struktury znajdują się jeszcze razem na płycie półprzewodnika,
- po dokonaniu montażu i zahermetyzowaniu układów w obudowach.

Oczywiście te dwie operacje kontrolne pozwalają na wyeliminowanie mikroprocesorów o zdecydowanie złej jakości. Jednakże dyskwalifikującymi mogą być też takie wady jak: nie-

znaczne odchyłki w zależnościach czasowych sygnałów, nieprawidłowa praca przy obniżonym napięciu zasilającym albo nie wykonywanie niektórych sekwencji rozkazów. Przyczyną takich wad może być wspomniana już wewnętrzna złożoność mikroprocesora oraz jego dynamiczny charakter pracy.

Złożoność struktury powoduje, że podczas wykonywania rozkazów sygnały przepływają wielokrotnie różnymi drogami między funkcjonalnymi blokami układu. Ponieważ w układzie jest wiele sprzężeń pojemnościowych, a działanie szeregu stopni układu zależy od poziomu napięcia w wewnętrznych węzłach /ładowanych lub rozładowywanych w takt impulsów zegarowych/, zależności czasowe a co za tym idzie częstotliwość pracy i napięcia zasilania mogą być tu sprawą krytyczną. Stąd wynika też zależność prawidłowego działania mikroprocesorów od kolejności wykonywanych rozkazów. W nawiązaniu do tego zagadnienia warto zwrócić uwagę na fakt braku technicznych możliwości przetestowania wszystkich możliwych kombinacji rozkazów, które może wykonywać mikroprocesor na różnych wzorach danych. Dla m -bitowego mikroprocesora o liczbie rozkazów równej n całkowita liczba kombinacji byłaby bowiem rzędu 2^{mn} , co nawet przy średniej klasie mikroprocesora i czasie trwania jednego testu ok. $1 \mu s$ wyklucza możliwość praktycznej realizacji takiego zadania. Wypracowanie natomiast pewnych typowych wzorów /sekwencji/ instrukcji, podobnie jak to miało miejsce w przypadku pamięci, jest dla mikroprocesorów za trudne, gdyż przy istniejącym większym stopniu skomplikowania nie ma tu takiej regularności w budowie wewnętrznej jaka występuje w matrycach pamięci.

W praktyce stosuje się obecnie kilka metod testowania mikroprocesorów. Techniczne podstawy tych metod oraz ich zakresy wykorzystania będą omówione poniżej.

Metody testowania mikroprocesorów

Najprostszą, najtańszą i najbardziej czystą metodą testowania układów mikroprocesorowych jest sprawdzenie poprawności ich działania w kompletnym systemie mikroprocesorowym, podczas wykonywania zestawu rozkazów przewidzianych dla tego systemu. Wadą jest tu jednak konieczność wymiany układów w systemie i ograniczona możliwość lokalizacji uszkodzeń oraz przesłedzenia funkcjonalnych charakterystyk mikroprocesora. Stąd mogą też wynikać trudności przy sprawdzaniu układów o granicznych wartościach parametrów albo pochodzących z tzw. drugiego źródła /od innego niż główny producent/. Dodanie do takiego kompletnego systemu mikroprocesorowego /lub mikrokomputera/ pośredniczących układów dla umożliwienia współpracy z klawiaturą i monitorem ekranowym, oraz uzupełnienie liczby układów pamięciowych, można uznać za sposób budowy jednego z prostrzych testerów mikroprocesorowych. Przydatność jego jest jednak ograniczona do stosowania u użytkowników mikroprocesorów /i to raczej nie w skali masowej/ oraz dla prac serwisowych [8].

Dość prostą i stosunkowo tanią jest też druga ze stosowanych metod testowania a mianowicie porównanie mikroprocesora badanego z tego samego rodzaju układem traktowanym jako wzorcowy, o sprawdzonej uprzednio poprawności działania /ang. comparison testing lub signature testing/. Na wejścia obu układów podaje się ustalone wzory danych i porównuje stany sygnałów na wyjściach, będące wynikiem wykonania określonych rozkazów. Dobór danych i sekwencji rozkazów może mieć na celu skrupulatniejsze sprawdzenie niektórych wewnętrznych bloków mikroprocesora albo może być dokonany tak aby uwzględnić przede wszystkim pewne przewidywane zastosowanie mikroprocesora.

Wadą metody porównawczej jest konieczność zapewnienia bardzo dobrej jakości układu wzorcowego, ograniczenie możliwości badań charakterystycznych, wreszcie trudności z osiągnięciem dobrej synchronizacji obu mikroprocesorów. Pewnym udoskonaleniem w tym zakresie jest taki sposób testowania, przy którym układ wzorcowy posiada korzystniejsze warunki pracy i oddzielne taktowanie. Sposób ten opracowany w firmie "Adar" nazwany został jako testowanie w regulowanym naturalnym otoczeniu mikroprocesora /ang. CNE - conditioned natural environment/ [5].

Inną wersję metody porównawczej reprezentuje testowanie z rozpoznawaniem wzoru sygnałów /ang. pattern recognition/ nazywane też

testowaniem z zapamiętaną odpowiedzią /ang. stored response/. Zamiast samego wzorcowego układu stosuje się tutaj każdorazowo do porównania wyników wykonania rozkazu zapamiętany wzór sygnału wyjściowego, otrzymany uprzednio przy wykorzystaniu znanego, dobrego jakościowo mikroprocesora, który mógł zasadniczo nawet tylko jeden raz wykonać prawidłowo żadaną sekwencję rozkazów. Elastyczność programu a zatem możliwość charakteryzacji układu jest w tej metodzie dostatecznie dobra lecz koszty sprzętu są dość wysokie.

Metoda ta została rozbudowana w firmie "Fairchild Systems Technology" i występuje pod nazwą techniki zapamiętanej odpowiedzi /ang. stored response technique/ lub metody LEAD /od ang. Learn, Execute and Diagnose a więc Nauczanie, Wykonywanie i Diagnoza/ [4]. W pierwszej fazie przygotowania testów, na podstawie wypracowanego wcześniej programu, generuje się sekwencje sygnałów doprowadzonych do wejść mikroprocesora wzorcowego. Odpowiedzi mikroprocesora są zapamiętywane w pamięci testera i w czasie testowania służą do porównania z odpowiedziami układu badanego. System testujący symuluje naturalne środowisko mikroprocesora, w postaci współpracujących pamięci, urządzeń peryferyjnych i realizacji systemu przerwań. Jeśli znana jest szczegółowo wewnętrzna budowa mikroprocesora, wówczas można opracować z pomocą komputera symulacyjny program działania tego mikroprocesora, podający pobudzenia układu i związane z nimi odpowiedzi na wyjściu. Teoretycznie rodzaj i liczba pobudzeń winny być tak dobrane, by każdy tranzystor w układzie co najmniej jeden raz był przełączony ze stanu włączenia do stanu wyłączenia i odwrotnie. Tak opracowany program jest następnie załadowany do pamięci buforowej testera, skąd w postaci ciągu sygnałów przechodzi w momencie testowania na wejście mikroprocesora i do komparatorów umieszczonych na jego wyjściu. Ten rodzaj testowania wymaga dość dużych nakładów na zaangażowany sprzęt, nie jest dostatecznie elastyczny i może tylko w niewielkim stopniu dostarczyć informacji o przyczynach i lokalizacji uszkodzeń. Eliminuje jednak układy o uszkodzeniach zupełnych i ogranicza zakres prac programowych nad testami [1, 3].

Najbardziej wszechstronną jest jednakże metoda testowania, w której wykorzystuje się algorytm generowania wzorów sygnałów dla badanego mikroprocesora /ang. pattern generation lub algorithmic pattern generation/ [1, 2]. Odpowiednie sekwencje rozkazów wraz z operandami i adresami są generowane w czasie rzeczywistym przez sterujący procesor lub procesory w testerze. Dla efektywnego wykorzystania metody algorytmicznego generowania wzorów sygnałów konieczna jest dobra znajomość organizacji i architektury mikroprocesora

ra. Ogólne koszty przygotowania algorytmu a także sprzętu pomiarowego są w tym przypadku dość wysokie.

Połączenie dwóch sposobów testowania - techniki zapamiętanej odpowiedzi i algorytmicznej generacji sygnałów dało podstawę nowej metodzie zaproponowanej przez pracowników firmy "Macrodata" [2] i nazwanej metodą diagnostycznego naśladownictwa /ang. diagnostic emulation/. W lokalnej, szybkiej pamięci testera działającego wg tej metody przechowuje się krótkie wzory danych zawierających różnego typu rozkazy stosowane w mikroprocesorze a cały program testowania przebiega pod kontrolą procesorów działających wg opracowanego algorytmu i mogących modyfikować na bieżąco program testów.

Testery mikroprocesorów

Wyróżnić można dwie grupy automatycznych testerów przeznaczonych dla klasyfikacji, pomiarów i badań mikroprocesorów:

1. Duże uniwersalne systemy pomiarowe umożliwiające pełne testowanie i charakteryzację praktycznie dowolnych systemów cyfrowych, przeznaczone dla producentów takich wyrobów, a głównie wykorzystywane podczas ich opracowywania i częściowo w trakcie produkcji;
2. Automatyczne testery, zorientowane na wyróżnione grupy wyrobów, o mniejszych możliwościach w zakresie pomiarów parametrycznych i charakteryzacji, lecz także o dużej wydajności ze względu na ich przeznaczenie dla potrzeb produkcji mikroprocesorów lub dla masowych użytkowników tych układów.

Systemy pomiarowe z grupy pierwszej są zwykle wielostanowiskowe i współpracować mogą z proberami oraz automatycznymi podajnikami układów. W mniejszej skali wyposażenie takie mogą mieć również testery z drugiej grupy. Do najbardziej znanych automatycznych systemów pomiarowych zaliczyć można [7]:

- System Sentry VII firmy "Fairchild Systems Technology"
- System Sentry VIII firmy "Fairchild Systems Technology" specjalizowany dla układów VLSI
- System S 3260 firmy "Tektronix"
- System MD 501 firmy "Macrodata"

W drugiej grupie testerów na uwagę zasługują:

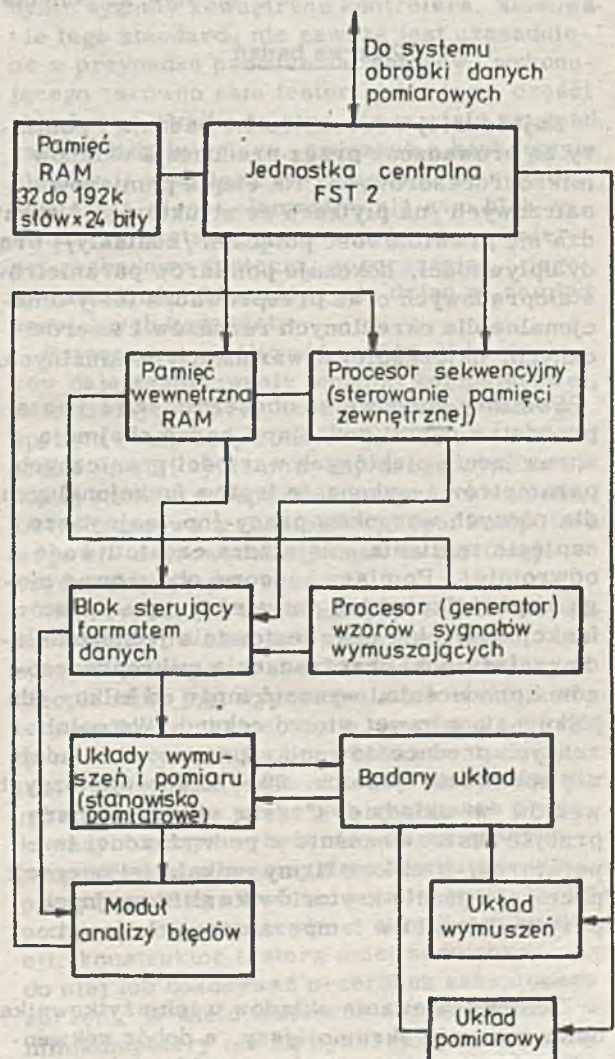
- tester Q 8000 firmy "Megatest"
- tester MK 17 firmy "Adar"

Na rys. 1 i 2 podane zostały schematy blokowe odpowiednio systemu Sentry VII i testera Q 8000, potraktowanych jako reprezentatywne pozycje w obu omawianych grupach. Pośrednie miejsce między wskazanymi grupami automatycznych testerów zajmuje system MD 154 firmy "Macrodata", powstały na bazie produkowanych przez tę firmę testerów do pamięci.

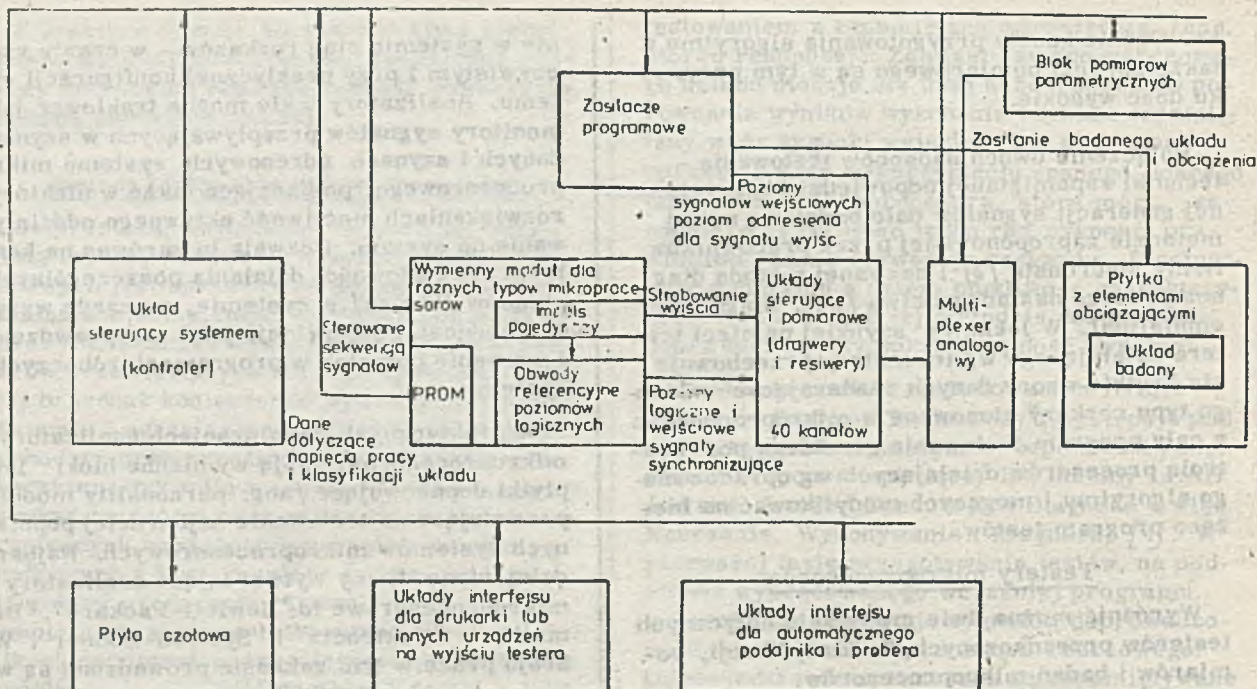
Na oddzielne omówienie zasługują analizatory do badań systemów mikroprocesorowych [9]. Pozwalają one śledzić realizowany aktual-

nie w systemie ciąg rozkazów - w czasie rzeczywistym i przy praktycznej konfiguracji systemu. Analizatory takie można traktować jako monitory sygnałów przepływających w szynach danych i szynach adresowych systemu mikroprocesorowego, posiadające także w niektórych rozwiązaniach możliwość aktywnego oddziaływania na system. Pozwala to zarówno na kontrolę prawidłowości działania poszczególnych układów /kostek/ w systemie, a przede wszystkim jednostki centralnej, jak i na sprawdzenie i usuwanie pomyłek w programach roboczych systemu.

W praktycznych rozwiązaniach analizatory mikroprocesorowe mają wymienne bloki lub płytki dopasowujące /ang. personality module/, pozwalające na testowanie najbardziej popularnych systemów mikroprocesorowych. Najbardziej znane firmy wytwarzające analizatory mikroprocesorowe to: "Hewlett-Packard", "Biomation", "Tektronix" i "Systron-Donner". W kraju prace w tym zakresie prowadzone są w Instytucie Elektroniki Politechniki Łódzkiej.



Rys. 1. Schemat blokowy systemu pomiarowego Sentry VII



Rys. 2. Schemat blokowy testera Q8000

Zakres badań

Najbardziej wszechstronne badania i pomiary są prowadzone przez producenta układów mikroprocesorowych. Na etapie pomiarów ostrzowych /na płytkach ze strukturami/ sprawdza się prawidłowość połączeń /kontakty/ i prądy upływności, dokonuje pomiarów parametrów stałoprądowych oraz przeprowadza testy funkcjonalne dla określonych rozkazów i wzorów danych, najczęściej w warunkach nominalnych.

Pomiary końcowe są obszerniejsze i poza przedstawionymi rodzajami badań obejmują sprawdzenie niektórych wartości granicznych parametrów i wykonanie testów funkcjonalnych dla różnych warunków pracy /np. najwyższe napięcie zasilania - najniższa częstotliwość i odwrotnie/. Pomiary końcowe obejmować mogą rzędu kilkuset lub powyżej tysiąca testów funkcjonalnych a czas testowania jednego układu, zależnie od przeznaczenia mikroprocesorów i producenta, wynosić może od kilku do kilkunastu a nawet więcej sekund. Wg opinii znanych producentów mikroprocesorów, udaje się sprawdzić tylko ok. 90 + 95% wewnętrznych węzłów w układzie. Często układy muszą przejść jeszcze badanie w podwyższonej temperaturze. Niektóre firmy unikają tej operacji przez zawężenie kryteriów kwalifikacyjnych przy testowaniu w temperaturze pokojowej.

Zakres testowania układów u ich użytkownika bywa znacznie skromniejszy, a dobór sekwencji rozkazów i wzorów danych ukierunkowany jest przez znane zastosowanie.

Literatura

- [1] A. C. L. Chiang: Test Schemes for microprocessor chips, 'Computer Design, April 1975 pp. 87-92
- [2] D. Hackmeister, A. C. L. Chiang: Microprocessor test technique reveals instruction pattern sensitivity, Computer Design, Dec. 1975 pp. 81-85
- [3] D. Izumi: The challenge of microprocessor chip testing, Proc. 1975 Western Electronic Show and Convention /Session 27 Automatic Testing in the Manufacturing Process/.
- [4] B. Schusheim: A flexible approach to microprocessor testing, Computer Design, March 1976 pp. 67-72.
- [5] R. L. Alonso, D. W. Madsen, H. Thaler: Conditioned natural environment testing /CNE/ - a low cost approach to full electrical testing of microprocessor and related components, 1977 Semiconductor Test Symposium.
- [6] K. Jefferies: Microprocessor testing techniques, Electronic Engineering, Jan. 1978, pp. 61-62.
- [7] S. E. Scrupski: Why and how users tests microprocessors, Electronics, March 2, 1978 pp. 97-104.
- [8] J. W. Neese: Microprocessor system validation and failure isolation with portable tester, Computer Design, Sept. 1977 pp. 105-111.
- [9] N. A. Robin: The logic analyzer: a computer trouble-shooting tool, Computer Design, March 1976 pp. 89-96.

mgr inż. JERZY KERN
mgr inż. KRZYSZTOF WILDT
Przemysłowy Instytut Elektroniki

MIKROPROCESOROWY KONTROLER TESTERA UKŁADÓW SCALONYCH

W każdym automatycznym testerze układów scalonych lub elementów dyskretnych znajduje się zespół, realizujący w czasie procesu testowania funkcje sterowania i programowania poszczególnych bloków wykonawczych i pomiarowych testera. Zespół taki, noszący nazwę kontrolera, spełnia w różnych testerach na ogół różne zadania, zależnie od stopnia złożoności testera, konfiguracji jego bloków, rodzaju testowanych elementów itp. Ze wzrostem asortymentu wyrobów półprzewodnikowych produkowanych przez przemysł krajowy rośnie bardzo szybko liczba testerów, różniących się od siebie funkcjonalnie i konstrukcyjnie. Istnieje oczywiście teoretyczna możliwość pomiarów wszystkich wyrobów półprzewodnikowych przez całkowicie uniwersalne systemy pomiarowe, ale rozwiązanie takie nie jest możliwe ze względów praktycznych.

Skomplikowanie i złożoność dużych systemów pomiarowych powoduje nie tylko wzrost kosztów sprzętu, oprogramowania i serwisu przekraczający granice opłacalności, ale pogarsza ich niezawodność do tego stopnia, iż niejednokrotnie czasy przestoju systemów przekraczają znacznie czas ich rzeczywistego wykorzystania. Optymalnym rozwiązaniem, zarówno z punktu widzenia producenta testerów, jak również ich użytkownika, byłoby budowanie różnych konstrukcyjnie i funkcjonalnie testerów z modułów, w maksymalnym stopniu zunifikowanych. O ile w niektórych firmach, specjalizujących się w konstrukcji automatycznych testerów pomiarowych udało się zunifikować część bloków wykonawczych w postaci typoszeregu modułów zasilaczy programowanych, generatorów itp., to nie zrealizowano do chwili obecnej unifikacji w zakresie kontrolerów.

Wprowadzenie standardowych interface w rodzaju standardu IEC rozwiązuje problem łączenia ze sobą zespołów produkowanych przez różne firmy w systemy i umożliwia samodzielne budowanie systemów pomiarowych przez ich

użytkowników, pod warunkiem dostępności na rynku poszczególnych części składowych systemu. Pomijając fakt, że standard IEC nie rozwiązuje problemu unifikacji konstrukcyjnej kontrolerów systemowych, standaryzując jedynie sygnały zewnętrzne kontrolera, stosowanie tego standardu nie zawsze jest uzasadnione w przypadku producenta testerów, wykonującego zarówno sam tester, jak i jego części składowe. Wynika to stąd, że w wielu przypadkach układy interface, związane z konkretnym blokiem, i spełniające wymagania standardu, zawierają więcej elementów niż sam blok wykonawczy systemu, co prowadzi do nadmiernej rozbudowy systemu, pogorszenia jego niezawodności i kosztów, nie dając w zamian specjalnych korzyści.

Potencjalną możliwością unifikacji kontrolerów daje zastosowanie techniki komputerowej, umożliwiającej przy użyciu takiego samego sprzętu realizację różnych jakościowo i ilościowo funkcji systemowych, drogą wyposażenia systemów w odpowiednie oprogramowanie. Dotychczasowa technologia sprzętu komputerowego nie doprowadziła jednak do unifikacji części sterujących systemu.

W przypadkach konstrukcji prostych stosunkowo testerów elementów dyskretnych lub specjalizowanych testerów układów scalonych nieopłacalne jest wyposażenie ich w minikomputery ogólnego zastosowania, zarówno ze względu na koszty, jak i niepotrzebną rozbudowę testerów. Niezależnie od tego stosowanie minikomputerów w większych systemach produkowanych przez wytwórców nie związanych bezpośrednio z przemysłem komputerowym posiada szereg utrudnień konstrukcyjnych. Dysponując minikomputerem o określonej konstrukcji, konstruktor testera musi dostosować się do niej lub dokonywać przeróbek zakupionego sprzętu. W wielu testerach wyposażonych w minikomputery nie są wykorzystywane pewne części konstrukcyjne minikomputera, natomiast w kalkulowane w koszty minikomputera

dostarczone oprogramowanie firmowe jest z reguły niewykorzystywane. Konstruktorzy systemu opracowują zwykle własne oprogramowanie systemowe, uwzględniając specyficzne zastosowanie minikomputera i konfigurację sterowanego obiektu.

Pojawienie się w ostatnich latach mikroprocesorów powinno umożliwić efektywne rozwiązanie konstrukcji kontrolerów systemowych, pozwalając na pogodzenie sprzecznych dotychczas wymagań

- unifikacji układowej

- optymalizacji konstrukcji testerów.

Konstruktor testera, dysponując modułami stanowiącymi elementy systemu mikrokomputerowego będzie mógł budować kontrolery o różnym stopniu złożoności, ale składające się z takich samych modułów podstawowych. W najprostszym przypadku mogą to być kontrolery jednopłytkowe zawierające jednostkę centralną i stałą pamięć programu, przy bardzo złożonych systemach - wielomikroprocesorowe kontrolery z rozbudowaną pamięcią i wieloma urządzeniami peryferyjnymi.

Obecnie w Przemysłowym Instytucie Elektroniki opracowany jest system mikrokomputerowy stanowiący kontroler o średnim stopniu złożoności, przeznaczony do sterowania testerami pomiarów statycznych i dynamicznych analogowych układów scalonych powszechnego użytku. Przeznaczenie kontrolera zdeterminowane jest jedynie opracowanym oprogramowaniem i doбором modułów na poziomie pakietów. Typoszereg opracowanych pakietów powinien umożliwić w przyszłości budowę kontrolerów dowolnych testerów.

Struktura układowa kontrolera

Kontroler MKT-1 jest jednomikroprocesorowym systemem, którego jednostka centralna zawiera mikroprocesor scalony Intel 8080A wraz ze specjalizowanymi układami pomocniczymi jak generator 8224, kontroler 8228, bufor 8212, oraz pamięć PROM 1 K bajt przeznaczoną na program monitora systemowego.

Jednostka centralna powiązana jest z innymi modułami systemu interfacem typu magistrali, przy czym struktura interface narzucona jest architekturą mikroprocesora. Są to szyny - adresowa, 16 bitowa oraz 8 bitowa szyna danych, wraz z sygnałami sterującymi "Odczyt pamięci", "Zapis do pamięci".

Poszczególne moduły systemu w postaci pakietów to:

- dekodery i rejestry przerwań zewnętrznych, o maksymalnej liczbie sygnałów przerwaniowych równej 16
- moduł pamięci EPROM - 4 K bajty
- moduł pamięci RAM - 4 K bajty
- moduły jednostek sterujących typowych urządzeń peryferyjnych produkcji krajowej -

- czytnik CT 2100
- perforator DT 105S

• pamięć kasetowa PK-1

• drukarka DZM-180

- moduł jednostki sterującej pulpitu operatorskiego.

System zaopatrzony jest również w moduł wzmacniacza logicznego, zwiększającego obciążalność szyn do 16 odbiorników TTL oraz zawierającego dekodery dowolnych 4 bitów adresowych. W systemie założono, że wszystkie rejestry jednostek sterujących urządzeń peryferyjnych oraz rejestry wejściowe i wyjściowe testera są komórkami pamięci operacyjnej adresowanymi szyną adresową w taki sam sposób jak pamięci RAM i EPROM systemu. Ponieważ sterowany tester jest zewnętrznym obiektem o charakterze nadrzędnym, sygnały przerwaniowe generowane są jedynie przez układy związane z testerem i informują one procesor centralny o zdarzeniach zaistniałych w sterowanym obiekcie. Tak więc synchronizacja działania urządzeń zewnętrznych z systemem odbywa się zarówno drogą programową /testowanie rejestrów systemu jednostek zewnętrznych/ jak również poprzez obsługę sygnałów przerwań.

Całe oprogramowanie systemowe wprowadzone jest do pamięci typu EPROM, natomiast zbiory danych, stanowiące program testowania zakodowane w postaci zerojedynkowej, zgodnie z przyjętymi kodami bloków wykonawczych i pomiarowych testera mogą być umieszczane zarówno w pamięciach typu EPROM jak również w pamięciach półprzewodnikowych RAM, zależnie od przeznaczenia kontrolera.

W przypadku, gdy urządzenie steruje testerem realizującym kontrolę produkcyjną niewielkiego stosunkowo asortymentu układów scalonych lub elementów dyskretnych, wówczas program testowania, realizujący określony algorytm testowania winien być umieszczony w pamięci EPROM, co pozwala na wyeliminowanie peryferyjnych urządzeń wejściowych typu czytnika taśmy lub pamięci kasetowej. Poza tym, system gotów jest zawsze do pracy po wyłączeniu zasilania.

W przypadku sterowania systemem o większym stopniu uniwersalności, przeznaczonym do testowania tak szerokiego asortymentu różnych układów scalonych, że zbiór wszystkich możliwych programów testujących zajmowałby zbyt dużą pojemność pamięci, programy pomiarowe powinny być wprowadzane poprzez czytnik taśmy lub pamięć kasetową do pamięci RAM przed rozpoczęciem procesu testowania. Drukarka i perforator służą do wyprowadzania cyfrowych wyników pomiarowych poszczególnych testów lub treści programów pomiarowych znajdujących się w pamięci systemu.

Struktura oprogramowania

Oprogramowanie kontrolera MKT-1 można podzielić na trzy części:

1. Monitor

2. Programy sterowania testerem pomiarowym
3. Programy operacji pomocniczych

Monitor jest programem nadrzędnym w stosunku do pozostałych części oprogramowania. Realizuje on standardowe funkcje systemowe związane z:

- zarządzaniem pamięcią RAM
- zarządzaniem pamięcią kasetową
- obsługą rutynową urządzeń peryferyjnych
- przyjmowaniem i rutynową obsługą przerw zewnętrznych.

Ponadto monitor zawiera program loadera bootstrapowego oraz realizuje funkcje inicjalizacji systemu i zarządzania stosem systemowym.

Programy sterowania testerem pomiarowym realizują wszelkie operacje związane z obsługą sterowanego testera. Tester komunikuje się z kontrolerem za pomocą sygnałów przerwanowych trzech klas:

1. Żądanie pomiaru na stanowisku o numerze N
2. Koniec testu
3. Żądanie przesłania następnego testu.

Przerwanie "żądania pomiaru na stanowisku o numerze N" inicjuje rozpoczęcie wykonywania programu pomiarowego o nazwie wyspecyfikowanej na pulpicie operatorskim zgłaszającego się stanowiska N. Przerwanie takie może być przyjęte i obsłużone efektywnie tylko wtedy gdy nie są wykonywane pomiary na innym stanowisku pomiarowym. Jeżeli przerwanie takie pojawi się w trakcie wykonywania pomiarów na innym stanowisku, to monitor "zapamiętuje" zgłoszenie stanowiska a jego efektywna obsługa rozpocznie się po zakończeniu aktualnie wykonywanych pomiarów. Powyższa procedura uwarunkowana jest własnościami konstruowanego równolegle testera układów scalonych, dla którego przeznaczona jest opiswana wersja kontrolera.

Obsługa przerwania "żądanie pomiaru" polega na wykonaniu systemowych operacji identyfikacyjnych /w celu uniezależnienia się od zakłóceń/, wyszukaniu programu pomiarowego o żądanej nazwie i przesłaniu treści pierwszego testu pomiarowego do żądającego stanowiska. Po zakończeniu wykonywania przez tester ostatnio przysłanego testu, tester generuje sygnał przerwania "koniec testu", co oznacza gotowość wyników testu do pobrania. Po wykonaniu rutynowej obsługi przerwania, związanej z jego identyfikacją, pobierane są wyniki testu po czym inicjowany jest program "procesora oceny wyniku".

Program ten dokonuje obróbki wyniku zgodnie z treścią wykonywanego programu pomiarowego, tzn. wykonuje jedną z następujących operacji:

1. zapamiętanie wyniku
2. porównanie wyniku z wartością referencyjną i zaklasyfikowanie
3. porównanie wyniku z wartością referencyjną i zapamiętanie wyniku
4. wykonanie operacji arytmetycznej /dodawanie, odejmowanie, dzielenie/ na pobranym ostatnio wyniku i uprzednio zapamiętanym
5. tak jak w pkt. 4 plus porównanie wyniku operacji arytmetycznej z wartością referencyjną
6. tak jak w pkt. 4 plus zapamiętanie wyniku operacji arytmetycznej
7. tak jak w pkt. 5 plus zakwalifikowanie układu badanego
8. nic nie rób.

Po wykonaniu jednej z powyższych operacji może być wykonany wydruk wyniku - jeżeli było to zaprogramowane. Po pewnym czasie od momentu wysłania przerwania "koniec testu", związanego z działaniem układów komutacyjnych w testerze generowane jest przerwanie "żądanie przesłania następnego testu".

Po wykonaniu rutynowej obsługi przerwania inicjowany jest program "sequencera", który określa, na podstawie informacji zakodowanej w treści programu, kolejność wykonywanych testów. Na podstawie wyniku ostatniego testu oraz informacji zawartej w programie pomiarowym "sequencer" może:

- wykonać skok do testu N jeżeli wynik był "zły"
- wykonać skok do testu N jeśli wynik był "dobry"
- wykonać skok bezwarunkowy
- nie wykonywać skoku - przesłać kolejny test
- jeśli układ został zaklasyfikowany spowodować zakończenie wykonywania programu pomiarowego

Programy operacji pomocniczych realizują obsługę operacji związanych z przesyłaniem treści programów pomiarowych pomiędzy różnymi urządzeniami peryferyjnymi, w tym również pamięcią RAM. Dla zachowania konsekwencji konstrukcyjnej przy konstrukcji oprogramowania wprowadzono również modułową architekturę. Rozwiązanie takie pozwoli w przyszłości na łatwe modyfikacje oprogramowania oraz dostosowanie do specyficznych właściwości innych testerów.

Przedstawiona koncepcja organizacji kontrolera testerów pomiarowych zakłada, że moduły opracowane dla kontrolera MKT-1 będą stosowane również do konstrukcji kontrolerów wieloprocessorowych przeznaczonych do bardziej złożonych testerów pomiarowych lub systemów przeznaczonych do badań inżynierskich.

SYSTEM DIAGNOSTYKI USZKODZEŃ W WALCOWNI WCB 2000 HUTY KATOWICE

Szybka lokalizacja zakłóceń, a przede wszystkim ustalenie pierwotnej przyczyny awarii jest bardzo istotnym czynnikiem zwiększającym wydajność nowoczesnych instalacji.

Duża liczba styków, których zmiana stanu sygnalizuje zakłócenie, przestrzenne rozproszenie urządzeń w których są one zainstalowane oraz krótkie odstępy czasowe w jakich mogą nastąpić zmiany ich stanu, uzasadniają wprowadzenie automatycznego nadzorowania położenia tych styków oraz centralnej rejestracji i diagnostyki sytuacji alarmowych.

Prace nad komputerowym systemem rejestracji i diagnostyki uszkodzeń w walcowni ciąglej zostały podjęte w Instytucie Systemów Sterowania w Katowicach, przy współpracy z Biurem Projektów Hutniczych w Gliwicach i Zakładem Urządzeń Komputerowych w Zabrze. System bazuje na minikomputerze MERA 400 i mikrokomputerach opracowywanych na bazie licencyjnej jednostki sterującej monitorów ekranowych MERA 7950.

Charakterystyka funkcjonalna systemu

Funkcje systemu

Zadaniem systemu jest śledzenie i wykrywanie na bieżąco zmian położenia styków. Informujących, o stanach awaryjnych urządzeń elektrycznych, mechanicznych, smarowniczych i hydraulicznych walcowni.

Zadziałanie któregośkolwiek ze styków powoduje wydrukowanie na drukarce alarmowej odpowiedniego komunikatu zawierającego:

- datę i czas zegarowy zakłócenia,
- numer i nazwę urządzenia,
- numer podzespołu,
- tekst komunikatu,
- ewentualny znak nierozróżnienia chronologii.

Powrót styku do stanu normalnego nie będzie rejestrowany. Wydruk komunikatów alarmowych następuje zgodnie z kolejnością powstania poszczególnych zakłóceń. W przypadku kilku jednoczesnych zakłóceń system przechowuje treść komunikatów w pamięci buforowej do czasu wydrukowania ich na drukarce alarmowej. Docelowo system będzie wspomagał dozór techniczny przy analizowaniu i ustalaniu pierwotnej przyczyny powstania awarii.

Dane techniczne systemu:

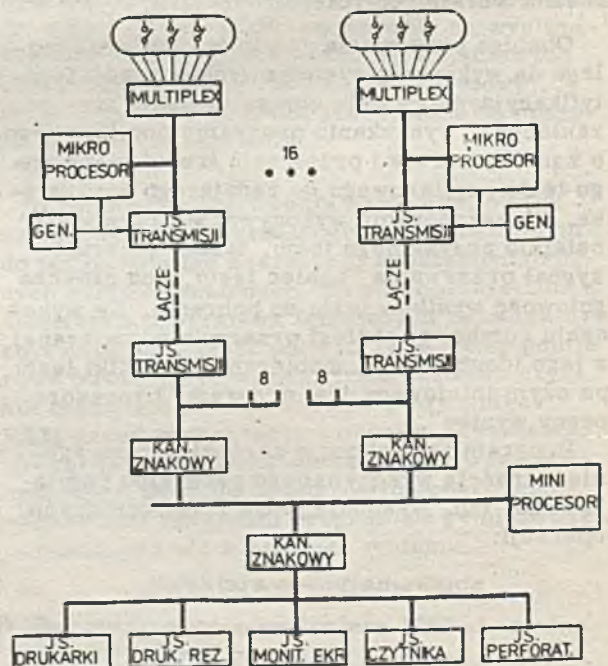
Kontrolowany obiekt stawia wysokie wymagania systemowi. Wiązą się one z następującymi danymi:

- liczba wejść dwustanowych - 4000,
- liczba grup sygnałowych - 16 po max. 400 punktów sygnałowych, związanych ze sobą funkcjonalnie lub technologicznie,
- rozdzielczość czasowa sygnałów wejściowych - 10 ms. Dotyczy to styków z tej samej grupy sygnałowej. Przy częstszych zakłóceniach komunikaty nie są gubione, lecz nie ma gwarancji zachowania kolejności wydruków.
- wydruk czasu z dokładnością - 0,1 s.
- maksymalna odległość kontrolowanych styków od minikomputera - 800 m.

Architektura systemu

Struktura połączeń

Przyjęto koncepcję systemu zdcentralizowanego, w którym lokalne mikrokomputery połączone są z centralnym minikomputerem po-



Rys. 1. Schemat blokowy zdcentralizowanego systemu diagnostyki

przez łącza cyfrowe o strukturze gwiazdziej. Gwiazdzista struktura połączeń charakteryzuje się wzajemnie jednoznaczny przyporządkowaniem mikrokomputerów do jednostek sterujących transmisją /point - to - point/. Struktura taka umożliwia stosunkowo szybkie przekazanie danych, dając mały poślizg pomiędzy momentem wykrycia uszkodzenia a czasem przypisanym mu w minikomputerze.

Łącza cyfrowe nie wymagają stosowania modemów. Poszczególne znaki danych przesyłane są szeregowo. Schemat blokowy zdecentralizowanego systemu diagnostyki ilustruje rys. 1.

Podział zadań w systemie

Każdą grupę sygnałową obsługuje odrębny mikrokomputer, spełniający rolę multipleksera i koncentratora danych. Przegląda on cyklicznie wszystkie podległe mu wejścia dwustanowe i porównuje ich stan z odpowiednimi bitami tablicy stanów. W przypadku wykrycia niezgodności formuowany jest 3 bajtowy meldunek alarmowy, buforowany a następnie przekazywany łączem cyfrowym do minikomputera. Meldunek zawiera:

- 6-bitowy adres podgrupy 8 wejść dwustanowych,
- wskaźnik nierozróżnialności chronologii,
- 8-bitowy wynik porównania,
- oraz znak kontrolny CRC.

Zadaniem minikomputera jest wczytywanie napływających strumieni danych z mikrokomputerów, przypisanie im daty i czasu zegarowego oraz opracowanie i wydrukowanie komunikatów alarmowych na drukarce znakowo-mozaikowej.

Struktura oprogramowania

Struktura oprogramowania systemu odpowiada dwupoziomowej strukturze hardware'u. Oprogramowanie użytkowe każdego z poziomów posiada budowę modułową.

Na poziomie mikrokomputerów, program użytkowy zawiera dwa główne, pracujące asynchronicznie moduły:

- moduł śledzenia stanów urządzeń i przygotowania meldunków binarnych,
- moduł wysyłania meldunków,

Program użytkowy korzysta z tablicy stanów zawierającej aktualne stany wszystkich punktów pomiarowych. Przygotowane meldunki kolejowane są /na zasadzie FIFO/ w obszarze buforowej pamięci.

Schematyczny podział pamięci mikrokomputerów ilustruje rys. 2. Program użytkowy przechowywany jest w pamięci stałej, a tablica stanów i obszar buforowy w pamięci danych.

Program użytkowy	Tablica stanów	Obszar buforowy
------------------	----------------	-----------------

Rys. 2. Schematyczny podział pamięci mikrokomputerów

Na poziomie minikomputera, program użytkowy zawiera 3 główne moduły:

- moduł wczytywania i kolejowania /na zasadzie FIFO/ meldunków napływających z mikrokomputerów. W module tym poszczególnym meldunkom przypisywany jest czas i nr grupy,
- moduł przetwarzania binarnej informacji meldunków do postaci komunikatów alfanumerycznych zrozumiałych dla dyspozytorów i umieszczenie tych komunikatów w wyjściowym obszarze buforowym minikomputera /na zasadzie FIFO/,

- moduł wprowadzania komunikatów.

Moduł przetwarzania korzysta z obszaru tablic zawierających nazwy i numery urządzeń i podzespołów oraz teksty alarmów. Każdy z modułów posiada inny priorytet. Najwyższy priorytet posiada moduł pierwszy /wczytywanie/ a najniższy - moduł drugi /przetwarzania/. Wymienione powyżej moduły użytkowe będą pracować pod kontrolą systemu operacyjnego czasu rzeczywistego. Schematyczny podział pamięci minikomputera przedstawia rys. 3.

System operacyjny	Program użytkowy	Obszar tablic	Obszar buforowy
-------------------	------------------	---------------	-----------------

Rys. 3. Schematyczny podział pamięci operacyjnej minikomputera

Zastosowanie mikrokomputerów w systemach rejestracji zdarzeń alarmowych i diagnostyki uszkodzeń umożliwiło przyjęcie struktury zdecentralizowanej charakteryzującej się:

- zmniejszonym obciążeniem jednostki centralnej dzięki wstępnej selekcji i przetworzeniu danych w lokalnych mikroprocesorach,
- zmniejszonym czasem reakcji systemu na zdarzenia alarmowe,
- obniżonym kosztem okablowania przez doprowadzenie sygnałów dwustanowych tylko do lokalnych szaf kablowych, usytuowanych w pobliżu kontrolowanych urządzeń,
- zwiększoną odpornością na zakłócenia przez skrócenie okablowania, wyeliminowanie wejść przerywających oraz zabezpieczenie przesyłanych danych nadmiarowymi znakami kontrolnymi,
- zwiększoną niezawodnością systemu dzięki autonomicznie pracującym mikrokomputerom, wyposażonym w stosunkowo dużą pamięć buforową.

L i t e r a t u r a

- [1] System diagnostyki uszkodzeń w urządzeniach elektrycznych i mechanicznych walcowni. Wstępne założenia systemu - opracowanie BIPROHUT w Gliwicach.
- [2] Rejestrator Wejść Dwustanowych. Wytyczne projektowania instalacji RWD - opracowanie IASE we Wrocławiu.
- [3] W. Lipajew, K. Kolin. Oprogramowanie podstawowe maszyn cyfrowych. PWN, W-wa 1975.
- [4] E. Yourdon. Projektowanie systemów o działaniu bezpośrednim. WNT, W-wa 1976.

BUDOWA I MOŻLIWOŚCI ZASTOSOWAŃ PROSTYCH UKŁADÓW KALKULATOROWYCH

Artykuł przedstawia zarys budowy i działania, wprowadzonego do seryjnej produkcji w Naukowo Produkcyjnym Centrum Półprzewodników CFMI, układu kalkulatorowego MC74007 stanowiącego podstawowy element ośmiódziałaniowego kalkulatora kieszonkowego. Ponadto przedstawiono nietypowe możliwości zastosowań tego układu. Układ kalkulatorowy MC74007 jest pierwszym produkowanym seryjnie w kraju układem scalonym wielkiej skali integracji.

Układ kalkulatorowy MC74007 został zrealizowany na płycie krzemowej o wymiarach 4,5 x 4,5 mm w technologii P-MOS z branką aluminiową przy wykorzystaniu techniki implantacji jonów. Układ składa się z ponad 10 tys. tranzystorów.

Funkcje układu kalkulatorowego

Układ MC74007 wykonuje cztery podstawowe działania arytmetyczne oraz funkcje typu x^2 , $1/x$, \sqrt{x} i obliczenia procentów. Ponadto posiada rejestr pamięci M dostępny z klawiatury, na którego zawartości można wykonywać operacje dodawania i odejmowania aktualnie wyświetlanych argumentów oraz operację zerowania. Układ MC74007 działa w reżimie pracy z automatyczną stałą. Operacje dodawania, odejmowania, mnożenia i dzielenia są wykonywane w ten sposób, że poza wykonaniem czynności określonej operacją powodują zapamiętanie w jednym z rejestrów danych pierwszego argumentu mnożenia oraz drugiego argumentu dodawania, odejmowania i dzielenia. Zapamiętana wartość może być wykorzystana jako drugi argument przy powtórnym naciśnięciu tego samego klawisza operacyjnego /pierwszy argument jest wyświetlany na wskaźniku/.

Budowa i działanie układu

Układ kalkulatorowy stanowi w istocie specjalizowaną maszynę cyfrową. Działanie układu polega na wykonywaniu programu znajdującego się w pamięci stałej /o pojemności 512 słów/. Rozkazy odczytywane są z pamięci wg

zawartości licznika rozkazów. Pobrany z pamięci rozkaz umieszczony jest w rejestrze rozkazów, następnie podlega zdekodowaniu i inicjuje wykonanie czynności określonej częścią operacyjną rozkazu. W przypadku, gdy aktualnie wykonywany rozkaz nie jest rozkazem skoku /efektywnym/, dokonywane jest zwiększenie zawartości licznika rozkazów o 1. W przypadku, gdy wykonywany jest rozkaz skoku, do licznika rozkazów wpisywana jest wartość adresu skoku /o ile rozkaz skoku jest efektywny/.

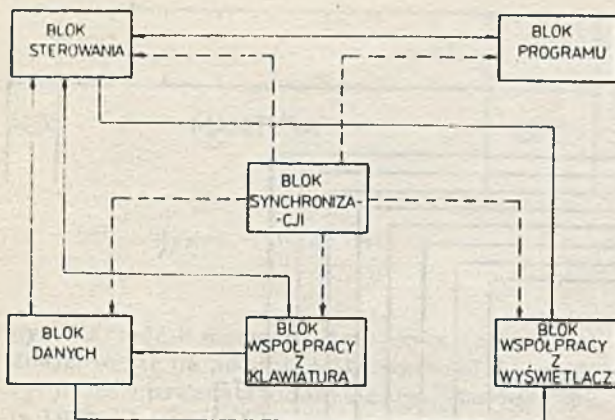
Układ kalkulatora MC74007 jest układem szeregowym. Przesyłanie informacji między rejestrami kalkulatora odbywa się szeregowo bit po bicie, również arytmometr jest układem szeregowym. Odczyt rozkazów z pamięci odbywa się równolegle. Działanie układu jest synchronizowane dwiema fazami wewnętrznego układu zegarowego.

W układzie można wyróżnić sześć bloków funkcjonalnych:

1. synchronizacji,
2. sterowania,
3. programu,
4. danych,
5. współpracy z klawiaturą,
6. współpracy z wyświetlaczem.

Blok synchronizacji służy do wytwarzania sygnałów zegarowych synchronizujących działanie układu kalkulatora. Głównym elementem bloku jest oscylator o częstotliwości nominalnej 400 kHz. W układzie wytwarzane są dwa podstawowe sygnały zegarowe o częstotliwości 200 kHz.

Blok sterowania wytwarza sygnały sterujące odcytem rozkazów z pamięci, przepływem informacji w kalkulatorze, działaniem arytmometru oraz układów wyświetlania i odczytu z klawiatury. Zasadniczym elementem bloku sterowania jest główna matryca sterująca oraz liczniki sterujące służące do wybierania bitu



Rys. 1. Schemat blokowy układu kalkulatorowego MC 74007

w ramach tetrady /licznik T/, adresowania tetrad w rejestrach danych /licznik D/ oraz do sterowania zapisem informacji z klawiatury /licznik W/.

Blok programu składa się z pamięci stałej /ROM/, w której zapisany jest program działania kalkulatora oraz z odpowiednich układów sterujących odczytem. Program kalkulatora składa się z 512 słów o długości 12 bitów. Zawartość odczytanej komórki pamięci zapisywana jest do 12-bitowego rejestru rozkazów.

Blok danych służy do przetwarzania i przechowywania informacji w kalkulatorze. Blok składa się z następujących elementów:

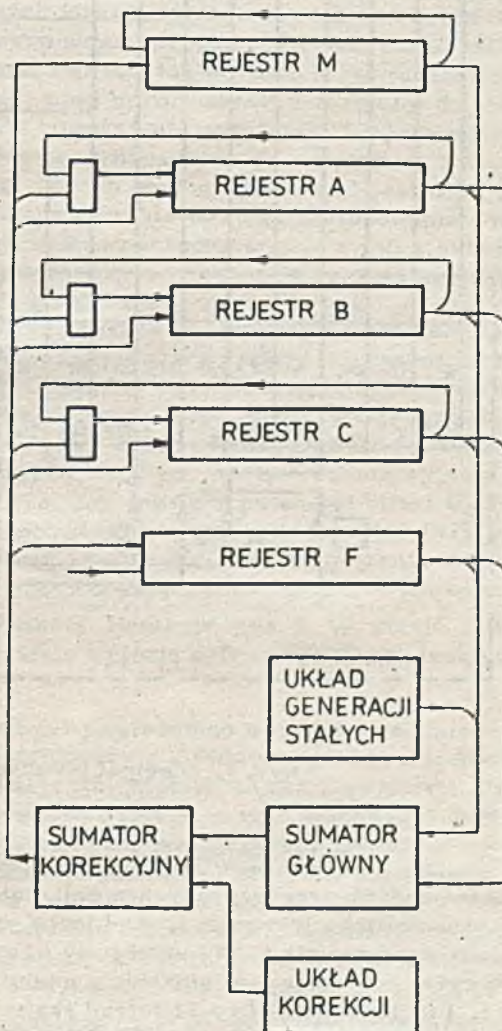
- trzy rejestry uniwersalne A, B, C,
- rejestr pamięci M,
- rejestr flagowy F /rejestr stanu układu i stanu programu/,
- sumator,
- układ generacji stałych.

Wszystkie rejestry są przesuwными rejestrami szeregowymi o długości 44 bitów. Schemat bloku danych ilustruje rys. 2. Dane z rejestrów podawane są szeregowo na szyny wejściowe sumatora, który wykonuje na nich operacje dodawania i odejmowania. Jako sumator zastosowano układ szeregowego sumatora liczb w kodzie BCD z nadmiarem 3 złożonego z sumatora głównego i sumatora korekcyjnego. Dane wyjściowe z sumatora przesyłane są na szynę wejściową rejestrów. Układ generacji stałych wytwarza stałe liczbowe o wartościach 0, 1 i 7 /w kodzie BCD z nadmiarem 3/ wykorzystywane w programie kalkulatora.

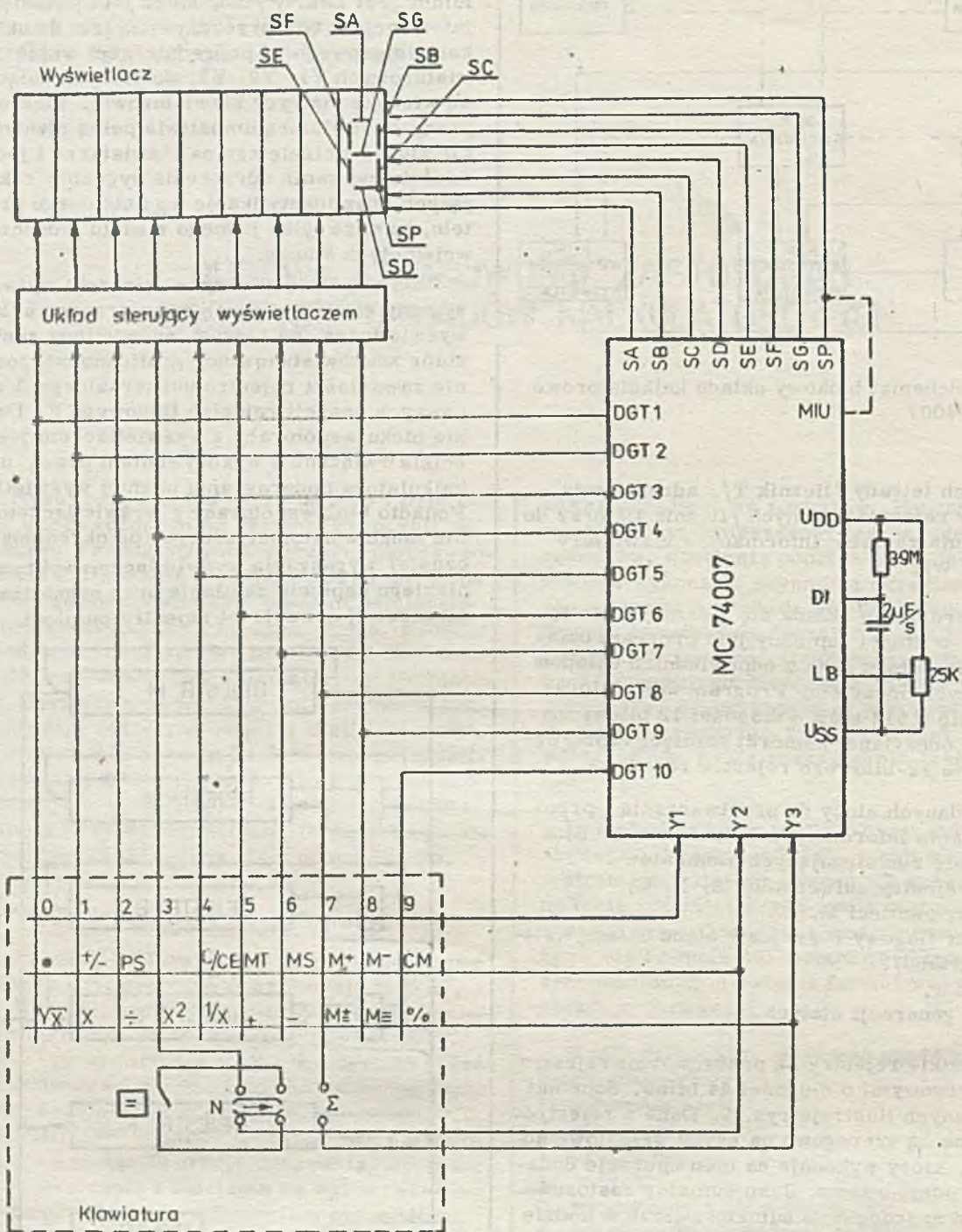
Blok współpracy z klawiaturą zapewnia obsługę klawiatury kalkulatora zorganizowanej w postaci matrycy złożonej z 10 kolumn i 3 wierszy. W węzłach matrycy umieszczone są klucze. Kolumny matrycy są kolejno sterowane z wyjść DGT 1 + 10 /rys. 3/. Jeżeli jeden z

trzech węzłówysterowanej w danej chwili kolumny jest zwarty /tzn. klucz jest wciśnięty/, informacja o tym przekazywana jest do układu kalkulatorowego za pośrednictwem wejść klawiaturowych Y1, Y2, Y3, do których dołączone są wiersze matrycy klawiaturowej. Blok współpracy z klawiaturą umożliwia pełną identyfikację klucza wciśniętego na klawiaturze i jednocześnie zapewnia odrzucenie sygnałów zakłócających oraz identyfikację wg ustalonego priorytetu, zawsze tylko jednego z wielu jednocześnie wciśniętych kluczy.

Blok współpracy z wyświetlaczem wytwarza sygnały sterujące pracą zewnętrznego układu wyświetlacza, na którym wyświetlony zostaje zbiór znaków stanowiący graficzne odwzorowanie zawartości rejestru uniwersalnego A oraz pewnych pozycji rejestru flagowego F. Działanie bloku współpracy z wyświetlaczem jest ściśle związane z wykonywaniem przez układ kalkulatora podprogramu obsługi wyświetlacza. Ponadto blok współpracy z wyświetlaczem spełnia funkcje automatycznego /po określonym czasie/ wygaszania wyświetlacza, wykrywania niskiego napięcia zasilania oraz sygnalizacji niezerowej zawartości rejestru pamięci.



Rys. 2. Schemat bloku danych



Rys. 3. Schemat połączeń kalkulatora ośmiodzianianowego

Postać danych

Postać danych przetwarzanych w kalkulatorze przedstawiona jest na rys. 4. Liczby przedstawione są w zapisie znak moduł przy czym każda cyfra kodowana jest binarnie z nadmiarem 3. Liczba składa się z 11 tetrad /każda tetrada ma długość 4 bitów/, podzielonych na 3 grupy o następującym znaczeniu:

1. AUX reprezentuje tetradę znaku liczby oraz spełnia funkcje pomocnicze,
2. MANTYSA przedstawia cyfry danej liczby,
3. DPT przedstawia położenie kropki dziesiętnej.

Najbardziej znacząca cyfra mantysy znajduje się z lewej strony /LSD/. Położenie kropki dziesiętnej liczone jest od prawej do lewej stro-

	MDS								LDS			
AUX	MANTYSA								DPT			

Rys. 4. Postać danych

ny. Wartość 0 oznacza, że kropka dziesiętna znajduje się na pozycji LSD, wartość 7 - na pozycji MSD przedstawionej liczby. Wartość pola DPT większe od 7 są używane jako wartości pomocnicze w trakcie wykonywania obliczeń.

Lista rozkazów

Algorytm działania kalkulatora opisany jest całkowicie przez zapisaną w pamięci programu /ROM/ sekwencję rozkazów. Słowo rozkazowe ma długość 12 bitów. Budowę słowa rozkazowego przedstawia rys. 5.

R1 R2 R3	R4 R5 R6 R7 R8 R9 R10 R11 R12
PODSTAWOWY KOD OPERACJI	PRZEDŁUŻENIE KODU OPERACJI ADRESY REJESTRÓW, ADRESY SKOKÓW, MASKA

Rys. 5. Budowa słowa rozkazowego

Lista rozkazów układu kalkulatorowego MC74007 obejmuje sześć grup instrukcji, a mianowicie:

1. rozkazy skoków,
2. rozkazy przesunięć,
3. rozkazy arytmetyczne;
4. rozkazy przesłań,
5. rozkazy flagowe,
6. rozkazy specjalne.

Grupa rozkazów skoków obejmuje skoki bezwarunkowe i warunkowe, przy czym adres skoku jest podany bezpośrednio w treści rozkazu. Rozkazy przesunięć obejmują przesunięcia wartości rejestrów o jedną tetradę w lewo lub w prawo. Rozkazy arytmetyczne to rozkazy DODAJ i ODEJMIJ. Grupa rozkazów przesłań obejmuje wszelkie możliwe w układzie przesłania międzyrejestrów /wg rys. 2/. W przypadku rozkazów przesunięć, przesłań i arytmetycznych argumentami operacji mogą być całe słowa danych lub tylko ich poszczególne części w zależności od zawartości tzw. pola maski rozkazu.

Rozkazy flagowe dotyczą operacji wykonywanych na poszczególnych bitach rejestru flagowego i obejmują operacje testowania, zerowania i zapalania wskazanych w rozkazie pozycji rejestru flagowego.

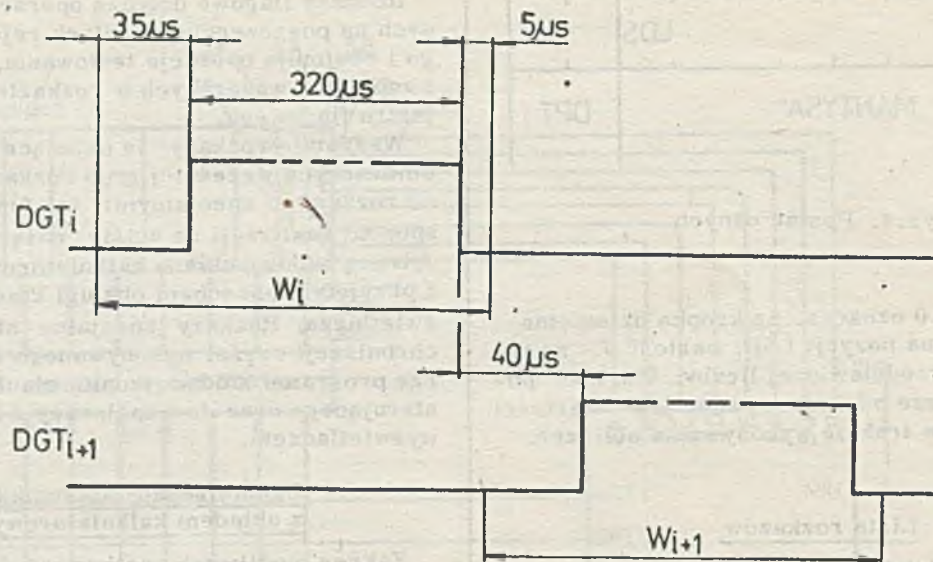
Wszystkie rozkazy nie należące do żadnej z omówionych wcześniej grup rozkazowych nazwano rozkazami specjalnymi. Ich funkcje oraz sposób realizacji są ściśle związane z wewnętrzną budową układu kalkulatorowego oraz z przyjętym sposobem obsługi klawiatury i wyświetlacza. Rozkazy specjalne służą do synchronizacji części wykonywanego w kalkulatorze programu z odpowiednim stanem licznika sterującego oraz do współpracy z klawiaturą i wyświetlaczem.

Ogólne zasady komunikacji z układem kalkulatorowym

Zakres możliwych zastosowań danego układu kalkulatorowego jest określony przez zespół realizowanych przez układ funkcji oraz przez sposób wprowadzania i wyprowadzania informacji do i z układu. W przypadku układu MC74007 repertuar wykonywanych przez układ funkcji jest dość ograniczony. Realizacja bardziej złożonych funkcji wymagałaby zastosowania dodatkowego układu programującego. Układ MC74007 został zaprojektowany jako układ jednostrukturalny, stąd wprowadzanie informacji do układu może odbywać się wyłącznie za pośrednictwem wejść przeznaczonych do współpracy z klawiaturą. Sposób obsługi tych wejść jest dość specyficzny i zwykle wymaga zastosowania dodatkowego układu pośredniczącego między układem kalkulatorowym i pozostałą częścią projektowanego urządzenia.

Ogólne zasady współpracy układu kalkulatorowego z klawiaturą zostały już omówione, pozostało jeszcze przedstawienie odpowiednich zależności czasowych. Czas wysterowania każdego z tzw. wyjść cyfrowych DGT 1 + DGT 10 wynosi ok. 320 μ s. W tym czasie wyjście jest aktywne, tzn. posiada potencjał bliski $U_{SS} = 0$ V. Okresy aktywności sąsiednich wyjść DGT przedzielone są okresem tzw. wygaszenia międzycyfrowego trwającym ok. 40 μ s. Wymienione zależności ilustruje rys. 6. W stanie W11 wszystkie wyjścia cyfrowe DGT są nieaktywne.

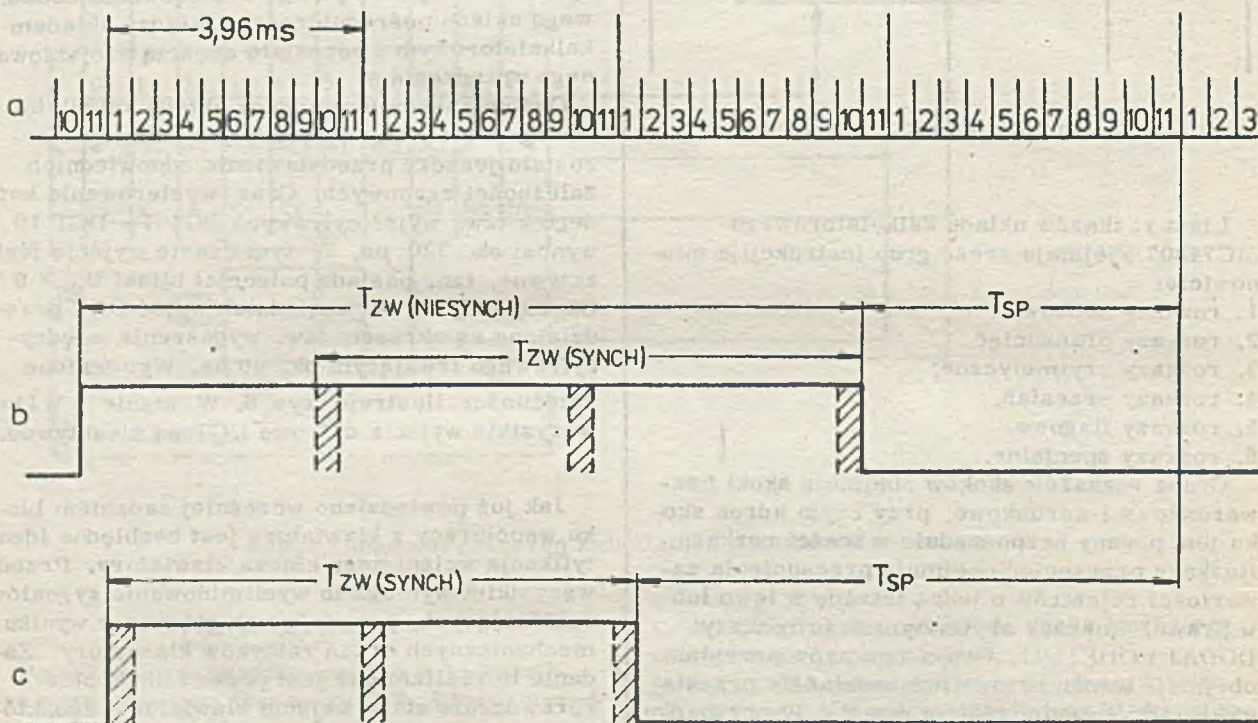
Jak już powiedziano wcześniej zadaniem bloku współpracy z klawiaturą jest bezbłędna identyfikacja wciśniętego klucza klawiatury. Przede wszystkim wymaga to wyeliminowania sygnałów zakłócających, powstających głównie w wyniku mechanicznych drgań zestyków klawiatury. Zadanie to realizowane jest przez kilkakrotne sprawdzenie stanu wejścia klawiaturowego, które przeszło w stan aktywny w celu ustalenia, czy w stanie tym dane wejście pozostaje przez dostatecznie długi okres czasu. Zbyt krótki okres aktywności danego wejścia jest interpretowany jako sygnał zakłócający.



Rys. 6. Harmonogram wysterowania wyjść DGT i / $i = 1 + 10$ /

Odpowiedni harmonogram przedstawia rys. 7. Wynika z niego, że całkowitą gwarancję zaakceptowania informacji z klawiatury uzyskuje się po minimum 3-krotnym wysterowaniu kolumny, w której znajduje się wciśnięty klucz. W przypadku zsynchronizowania zwarcia klucza

z okresem wysterowania kolumny, w której ten klucz się znajduje, minimalny czas zwarcia wynosi $T_{zw} = 8,3$ ms. Jeśli zwarcie klucza występuje w sposób przypadkowy, czas zwarcia powinien wynosić co najmniej 12 ms.



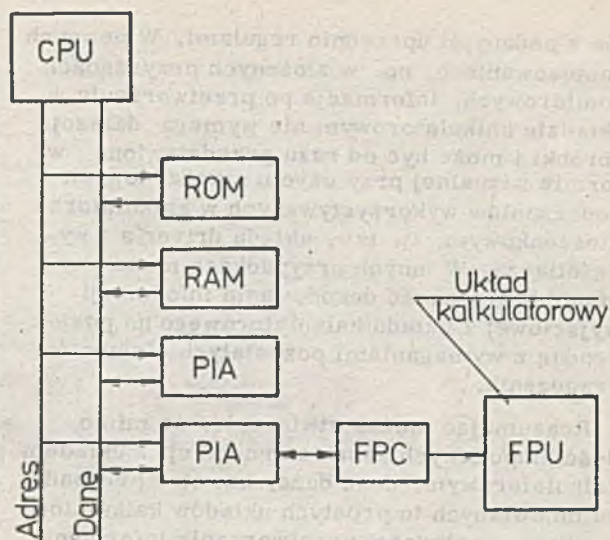
Rys. 7. Harmonogram wprowadzenia informacji do układu kalkulatorowego: a/ stany licznika sterującego w; b/ zwarty klucz znajduje się w kolumnie sterowanej z wyjścia DGT 10; c/ zwarty klucz znajduje się w kolumnie sterowanej z wyjścia DGT 1

W zależności od wymagań urządzenia powyższe dane umożliwiają wybranie odpowiedniego reżimu współpracy z układem kalkulatorowym. W przypadku krótszych czasów zwarcia wejść klawiatury Y informacja wejściowa będzie potraktowana jako sygnał zakłócający i zostanie odrzucona. Po zaakceptowaniu informacji z wejść klawiatury Y blok współpracy z klawiaturą wymaga, by klawiatura, przed wprowadzeniem następnej danej, pozostawała przez pewien okres czasu w stanie spoczynku. Czas ten musi być równy co najmniej czasowi jednego pełnego obiegu wewnętrznego licznika sterującego W /liczonego od stanu początkowego W1 włącznie/. Dobrym przybliżeniem tego okresu jest czas liczony od końca okresuysterowania wyjścia DGT 10 do początku drugiegoysterowania wyjścia DGT 1. Jak wynika z rys. 7, w najmniej korzystnym przypadku zwarcia klucza sterowanego z wyjścia DGT1, wymagany czas spoczynku klawiatury wynosi ok. 8 ms.

W przypadku zbyt krótkich okresów spoczynku informacja pochodząca od kolejnego wciśnięcia klucza nie zostanie przez układ zaakceptowana. Tego typu zdarzenie układ zinterpretuje jako przedłużenie aktywnego stanu poprzednio zwartego klucza. Całkowity czas wprowadzenia danej do układu kalkulatorowego w przypadku całkowicie synchronizowanego reżimu pracy, zawiera się więc w granicach od ok. 12,3 ms w przypadku klucza z kolumny DGT11, do ok. 15,9 ms, w przypadku klucza z kolumny DGT1. Dla niesynchronizowanego reżimu pracy czas ten wydłuża się do ok. 20 ms, tzn. o 4 + 8 ms.

Wszystkie podane wyżej czasy są prawdziwe jedynie w przypadku, gdy układ kalkulatorowy wykonuje podprogram obsługi klawiatury i wyświetlacza. Zagadnienie to jest szczególnie istotne, zwłaszcza przy wyprowadzaniu informacji z układu i zostanie rozważone dokładniej w trakcie omawiania tego procesu.

W przypadku układu MC74007 informację wyjściową można otrzymać jedynie wykorzystując wyjścia przeznaczone do sterowania wyświetlaczem kalkulatora. Przy pobieraniu informacji z układu kalkulatorowego istotne znaczenie ma moment, w którym można już przystąpić do dekodowania informacji z wyjść cyfrowych i segmentowych. Układ MC74007 jest tak zorganizowany, że w trakcie przetwarzania informacji, przy użyciu odpowiednich podprogramów, wszystkie wyjścia segmentowe są nieaktywne, a informacja z klawiatury nie jest akceptowana. Wyświetlanie i pobieranie informacji z klawiatury możliwe jest jedynie w trakcie wykonywania podprogramu obsługi klawiatury i wyświetlacza. Przejdźmy do podprogramu można najprościej wykryć analizując stan wyjścia segmentowego SP sterującego wyświetlaniem kropki dziesiętnej. Wyświetlana liczba zawiera zawsze kropkę dziesiętną, przy czym w przypadku liczby całkowitej znaj-



Rys. 8. Rozszerzenie możliwości arytmetycznych systemu mikroprocesorowego przez dołączenie układu kalkulatorowego

duże się ona z prawej strony najmniej znaczącej cyfry. Wykrycie aktywnego stanu na wyjściu SP oznacza, że można przystąpić do dekodowania informacji wyjściowej, oczywiście z zachowaniem synchronizacji ze stanami wyjść cyfrowych DGT, wykorzystując wyjścia cyfrowe DGT1 + DGT9 oraz wyjścia segmentowe SA + SG i SP. Przyporządkowanie wyjść segmentowych segmentom układu wyświetlacza podano na rys. 4. Wyjścia segmentowe związane z wyświetlanym znakiem są aktywne, tzn. posiadają potencjał bliski $U_{SS} = 0V$ przez cały czas sterowania wyjścia DGT' zgodnego z pozycją wyświetlanego znaku. Liczba stanowiąca informację wyjściową z układu kalkulatorowego wyświetlana jest w czasie DGT2 + DGT9, przy czym w czasie DGT2 wyświetlana jest jej najbardziej znacząca cyfra. Na pozycji wyświetlacza sterowanej z wyjścia DGT10 wyświetlane są znaki: minus, nadmiar, niedomiar oraz znak sygnalizujący zbyt niskie napięcie zasilania. W czasie aktywnego stanu wyjścia DGT10, wyświetlacz jest nie wykorzystywany i wszystkie wyjścia segmentowe są w stanie nieaktywnym.

Możliwości zastosowań układu kalkulatorowego

Wszystkie zastosowania prostych układów kalkulatorowych sprowadzają się do wykorzystania ich jako specjalizowanych elementów przetwarzania informacji. Układ kalkulatorowy spełnia w urządzeniu rolę złożonego arytmometru posiadającego dodatkowo zdolność wykonywania operacji mnożenia i dzielenia, a czasami, jak w przypadku układu MC74007, nawet pierwiastkowania. Algorytmy wykonywania obliczeń pozostają oczywiście identyczne jak dla kalkulatora kieszonkowego, w którym układ jest wykorzystywany. Podstawowa różnica w stosunku do standardowego zastosowania polega na zastąpieniu klawiatury odpowiednim układem elektronicznym symulującym jej działanie zgod-

nie z podanymi uprzednio regulami. W pewnych zastosowaniach, np. w złożonych przyrządach pomiarowych, informacja po przetworzeniu w układzie kalkulatorowym nie wymaga dalszej obróbki i może być od razu przedstawiona w formie wizualnej przy użyciu standardowych podzespołów wykorzystywanych w kalkulatorze kieszonkowym, tj. tzw. układu driver'a i wyświetlacza. W innych przypadkach może wystąpić konieczność dekodowania informacji wyjściowej z układu kalkulatorowego na postać zgodną z wymaganiami pozostałych elementów urządzenia.

Reasumując można stwierdzić, że mimo dość kłopotliwych zasad komunikacji z układem kalkulatorowym, dość duże, nawet w przypadku omawianych tu prostych układów kalkulatorowych, możliwości przetwarzania informacji uzasadniają szersze wykorzystywanie tego typu układów. Podstawowe, wynikające stąd korzyści, to znaczne przyspieszenie procesu projektowania i uruchamiania jednostki arytmetycznej urządzenia oraz zmniejszenie wymiarów i zwiększenie niezawodności całego urządzenia.

Na zakończenie przedstawiona zostanie koncepcja włączenia układu kalkulatorowego do systemu mikroprocesorowego. Zdolności przetwarzające systemów mikroprocesorowych są zwykle ograniczone do arytmetyki stałoprzecinkowej. Jedną z możliwości ich rozszerzenia o arytmetykę zmiennoprzecinkową jest dołączenie do systemu układu kalkulatora cztero działaniowego traktowanego jako urządzenie zewnętrzne /rys. 8/. Połączenie i współpraca między jednostką centralną CPU a jednostką arytmetyki zmiennoprzecinkowej FPU /czyli układem kalkulatorowym/ może być dokonana za pośrednictwem układu sterującego jednostkę arytmetyki zmiennoprzecinkowej FPC. Jako FPC można wykorzystać odpowiedni układ adaptera interfejsu peryferyjnego PIA. Komu-

nikacja z mikroprocesorem odbywa się przy wykorzystaniu sygnałów przerwań.

Wykorzystywanie układu kalkulatorowego jako jednostki arytmetyki zmiennoprzecinkowej może okazać się tańsze i łatwiejsze niż stosowanie skomplikowanego programu zapamiętywanego w pamięci ROM. Ponadto działanie takiej jednostki FPU może odbywać się równolegle z pracą CPU, co zwiększa efektywną szybkość działania systemu. Układ kalkulatorowy posiada określone zdolności przetwarzania informacji, które w dużym stopniu narzucone są przez treść programu umieszczonego w pamięci stałej. Pamięć stała jest elementem, w którym najłatwiej można wprowadzić zmiany w stosunku do aktualnie produkowanego układu. Zmiana programu kalkulatora, przeprowadzona w oparciu o znajomość jego listy rozkazów, umożliwi dokonanie gruntownej zmiany wykonywanych przez układ funkcji. W przypadku uzasadnionym względami ekonomicznymi /tj. długością serii/, zmiana programu kalkulatora mogłaby być dokonana bez większych trudności.

Na zakończenie autorzy pragną wyrazić podziękowanie dr Krystynie Siekierskiej i mgr inż. Sławomirowi Bronowskiemu za wiele cennych uwag dotyczących realizacji układów MOS wielkiej skali integracji.

L i t e r a t u r a

- [1] W. N. Carr, J. P. Mize: Projektowanie i zastosowanie układów MOS wielkiej skali integracji WNT, Warszawa 1976
- [2] H. Schmidt: BCD-logic - cykl artykułów zamieszczonych w Electronic Design, numery 13-19, 1973
- [3] P. H. Stacken: Using a calculator chip to extend a microprocessor's capabilities. Computer Design, Sep., 1975.

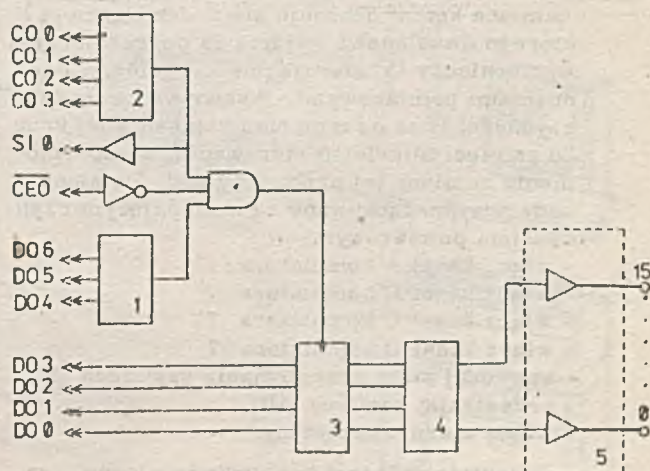
SYSTEM DIAGNOSTYKI USZKODZEŃ W WALCOWNI WCB 2000 HUTY KATOWICE

Szereg metod pomiarowych struktur półprzewodnikowych polega na przetwarzaniu charakterystyk statycznych. Dobrym przykładem jest tu wyznaczanie profilu domieszkowania półprzewodnika na podstawie przebiegu charakterystyki pojemnościowo-napięciowej utworzonej na powierzchni badanego materiału bariery Schottky'ego. Podobnie cennym narzędziem badawczym zjawisk zachodzących w złączach Schottky'ego i p-n jest analiza charakterystyk prądowo-napięciowych przedstawionych w skali liniowo-logarytmicznej, czy logarytmiczno-logarytmicznej.

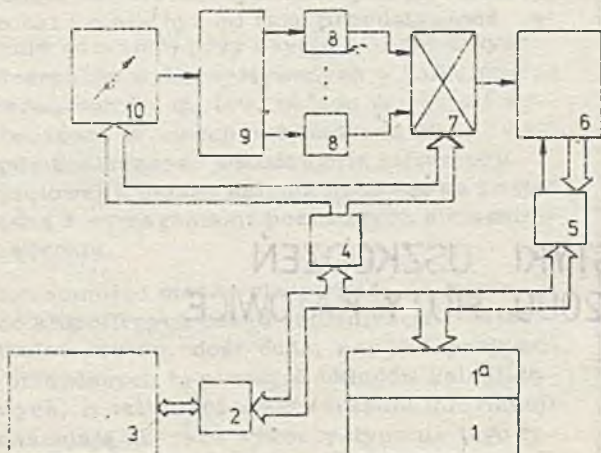
Porównując pod kątem elastyczności przy realizacji algorytmów przetwarzania oraz dokładności uzyskiwanych wyników metody analogowe i cyfrowe uznać należy wyraźną wyższość tych ostatnich. Jednak ich efektywność staje się zadawalająca dopiero wówczas, gdy układ pomiarowy nabiera właściwości systemu, to znaczy wtedy gdy ma miejsce przesyłanie danych pomiarowych i rozkazów sterujących czynnościami pomiarowymi bez udziału człowieka. Punktem wyjścia do budowy takiego systemu był zestaw graficzny firmy "Hewlett-Packard" złożony z kalkulatora stołowego HP9820A i plottera HP9862A. Schemat blokowy całości ilustruje rys. 1, zaś rys. 2 schemat blokowy interfejsu rozkazów sterujących, który należało dobudować, by system mógł spełniać swoje funkcje. Zbudowany on został w oparciu o krajowe cyfrowe układy scalone TTL.

W miejsce kalkulatora HP9820A użyty może być również inny model tej firmy: HP9810A, HP9821A, HP9830A i B posiadający takie samo złącze [1] do podłączenia urządzeń peryferyjnych, zarówno wyjściowych jak i wejściowych. Do wprowadzenia danych służy inter-

fejs HP11203A (5) przetwarzający kod BCD na ciąg znaków w kodzie ISO 7 [2]. Pośredniczy on też w wyzwalaniu pomiaru woltomierza cyfrowego (6) typu V534 firmy "Meratronik". Do sterowania służy interfejs (4). Zapisanie rozkazu w rejestrze (3) - rys. 2 następuje po porównaniu w dekodерze (2) numeru urządzenia wyjściowego podanego w instrukcji typu WRITE [3] z ustawieniem dekodera. Dzięki



Rys. 1. Kalkulatorowy system pomiarowy do wyznaczania charakterystyk statycznych: 1- kalkulator HP9820A, 1a-linia komunikacji wewnętrznej kalkulatora, 2-interfejsplottera, 3-plotter HP9862A, 4-interfejs rozkazów sterujących /szczegóły - rys. 2/, 5-interfejs wprowadzania danych cyfrowych, 6-woltomierz cyfrowy serii V530 "Meratronik", 7-komutator analogowy, 8-przetwornik natury sygnału mierzonego, 9- obiekt pomiaru, 10-zasilacz sterowany ZF 75 produkcji ZD ASPAN



Rys. 2. Interfejs rozkazów sterujących: 1-układ eliminacji znaków o kodzie mniejszym od 20_8 , 2-układ badań zgodności adresów, 3-czterobitowy rejestr pamiętający, 4-dekoder rozkazów sterujących, 5-blok wzmacniaczy dwustanowych sygnałów sterujących

układowi eliminacji znaków o kodzie $< 20_8$ (1) zapis nie dokonuje się gdy znak przesłany nie jest cyfrą lub literą. Upraszcza to postać instrukcji zapisu. Zapamiętywanie czterech najmniej znaczących bitów otrzymywanych znaków daje 16 różnych rozkazów sterujących. Zamiana kodów dokonuje się w dekodерze (4), którego dwustanowe wyjścia za pośrednictwem wzmacniaczy (5) sterują poszczególnymi czynnościami pomiarowymi. Wykonywanie każdej czynności trwa od momentu zapisania jej kodu do pamięci interfejsu sterowania, aż do momentu zamiany jej przez inny kod. Ustalone kody przyporządkowane są następującym czynnościom pomiarowym:

- włącz kanał A komutatora (7)
- włącz kanał B komutatora (7)
- włącz kanał C komutatora (7)
- włącz kanał D komutatora (7)
- spowoduj start przestrajania zasilacza (10)
- przestrajaj zasilacz (10)
- włącz alarm akustyczny

Komutator (7) jest konstrukcją własną, zasilacz to zasilacz funkcyjny ZF 75 produkcji ZD ASPAN dający na wyjściu napięcie narastające ze stałą, ustawianą ręcznie prędkością. Wzrost napięcia zasilacza jest zatrzymywany

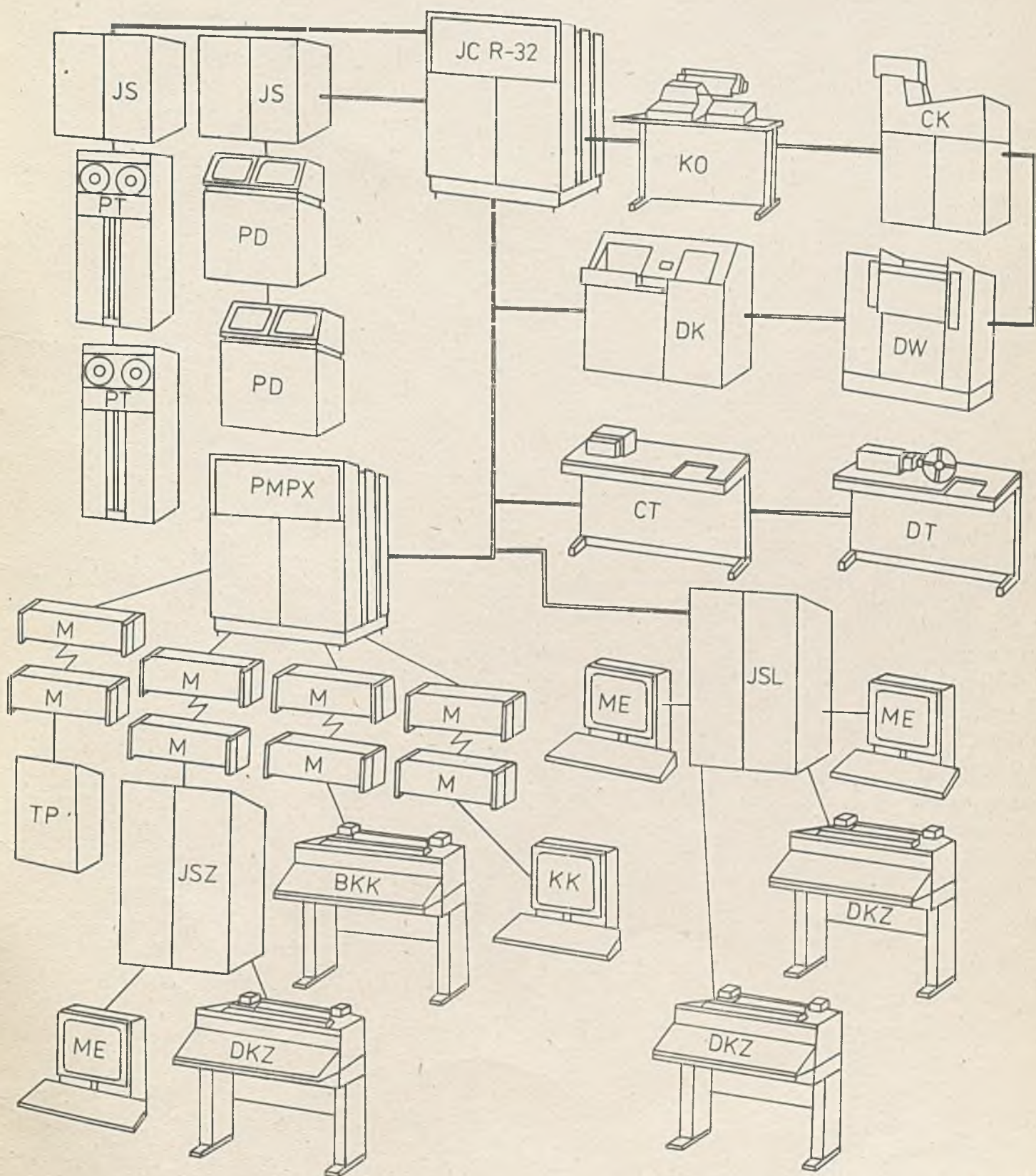
na czas pomiarów na przewidzianych programach kanałach. Wartość kolejnych przyrostów jest proporcjonalna do czasu wykonywania rozkazu przestrajania wynikającego z programu obsługującego pomiar. Jeden z kanałów komutatora (7) rezerwowany jest dla pomiaru napięcia wymuszającego badany przebieg. Przetworniki analogowo-analogowe (8) dają na wyjściu napięcie jednego wolta, gdy wielkość mierzona na wejściu osiąga przewidziane maksimum. Wymieniony wśród czynności pomiarowych alarm akustyczny włączany jest w momencie, gdy program napotyka konieczność zmiany zakresu jednego z przetworników (8), co wymaga ręcznej obsługi.

Automatyzacja tylko prostych, ale wielokrotnie powtarzających się w czasie wykonywania jednej charakterystyki czynności upraszcza bardzo układy sterowania, eliminuje konieczność przystosowania laboratoryjnych przyrządów używanych jako bloki systemu, a nie zwiększa w sposób istotny czasu pomiarów, zważywszy, że interpretacja uzyskanych badań musi odbywać się w trybie konwersacyjnym prowadzonym z wykorzystaniem plottera.

Zastosowany sposób sterowania nadaje się również do stosowania w systemach mniej elastycznych, których jednostki sterujące nie mają interpretatora języka wyższego rzędu, jak w kalkulatorze HP9820A. W systemie przeznaczonym do wykonywania rutynowych pomiarów, gdzie ewolucja programu pomiarów nie jest przewidywana, funkcję kontrolera sprawować może któryś ze znanych 8-bitowych mikroprocesorów. Problem stanowi urządzenie do przedstawiania wyników przetworzonej informacji w formie wykresów. Wydaje się, że można go w zadawalający sposób rozwiązać dostępnymi środkami.

L i t e r a t u r a

- [1] 9810A/9820A Calculator Service Manual. Hewlett-Packard Company Loveland 1972.
- [2] Hewlett-Packard Calculator 11203A BCD Interface. Installation and Service Manual. Hewlett-Packard Company Loveland 1974.
- [3] Hewlett-Packard Calculator 11202A I/O Interface. Installation and Service Manual. Hewlett-Packard Company Loveland 1974.



JC - jednostka centralna, KO - konsola operatorska, CK - czytnik kart, DK - dziurkarka kart, CT - czytnik taśmy papierowej, DT - dziurkarka taśmy papierowej, DW - drukarka wierszowa, JS - jednostka sterująca, JSL - jednostka sterująca lokalna ME, JSZ - jednostka sterująca zdalna ME, PT - pamięć taśmowa, PD - pamięć dyskowa, ME - monitor ekranowy, DKZ - drukarka kopii znakowa, PMPX - multiplexor programowany /procesor komunikacyjny/, M - modem, KK - końcowe urządzenie konwersacyjne, TP - terminal programowany, BKK - buforowane końcowe urządzenie konwersacyjne

