

INSTYTUT ŁĄCZNOŚCI

REFERATY PROBLEMOWE

Zeszyt 49

Jerzy Laube

ANALIZATOR SYGNATUR - UNIWERSALNY
PRZYRZĄD DIAGNOSTYCZNY DO URZĄDZEN CYFROWYCH



Warszawa - styczeń 1982

681.326.7

I N S T Y T U T Ł Ą C Z N O Ś C I

KOŁO ZAKŁADOWE STOWARZYSZENIA ELEKTRYKÓW POLSKICH

Na prawach rękopisu

R E F E R A T Y P R O B L E M O W E

Zeszyt 49

Jerzy Laube

ANALIZATOR SYGNATUR - UNIWERSALNY
PRZYRZĄD DIAGNOSTYCZNY DO URZĄDZEŃ CYFROWYCH

Warszawa - styczeń 1982

S-8971

Zespół Redakcyjny:

dr inż. Stanisław Sońta, mgr inż. Andrzej Stągrowski,
mgr inż. Krystyna Frączek

Opracował:

mgr inż. Jerzy Laube

BIBLIOTEKA
Instytutu Łączności
Nr 5-8971

Zakład Metod Eksploatacji Sieci i Urządzeń Telekomunikacyjnych
/Z-24/

Instytut Łączności, Oddział w Gdańsku

80-252 Gdańsk, ul. Jaśkowa Dolina 15, tel. 41-80-91 w. 36

Praca problemu węzłowego nr 13.04.Y.02.02

Opiniował: inż. Paweł Godlewski

Maszynopis dostarczono dnia 15.VII.1981 r.

W referacie przedstawiono ogólne zasady tzw. analizy sygna-
tur, będącej metodą testowania urządzeń cyfrowych, zwłaszcza
mikroprocesorowych. Przedstawiono zasady projektowania urzą-
dzeń przeznaczonych do testowania metodą analizy sygna-
tur oraz podano podstawowe właściwości odpowiedniego przyrządu
diagnostycznego, tzw. analizatora sygna-
tur, którego prototyp
został skonstruowany w Zakładzie Metod Eksploatacji Sieci i
Urządzeń Telekomunikacyjnych /Z-24/ Ił.

Redaktor: mgr K. Juskiewicz

Montaż tekstu: F. Miłkiewicz

Wpłynęło do Działu Wydawniczego Instytutu Łączności
w Warszawie, ul. Szachowa 1 dnia 23.X.1980 r.

Nakład 70 egz.

S P I S T R E Ś C I

Jerzy Laube

ANALIZATOR SYGNATUR - UNIWERSALNY PRZYRZĄD DIAGNOSTYCZNY DO URZĄDZEŃ CYFROWYCH

	Str.
1. Wprowadzenie	1
2. Ogólne zasady analizy sygnatur	2
3. Zasada działania analizatora sygnatur	3
4. Projektowanie dla potrzeb analizy sygnatur w przypadku urządzeń mikroprocesorowych	8
4.1. Ogólne zasady	8
4.2. Proces "rozszerzania jądra" systemu	9
4.3. Ogólne zasady budowy programu testującego	14
5. Zasady postępowania w przypadku urządzeń realizowanych w technice małej i średniej skali integracji	18
6. Podsumowanie	21
Wykaz literatury	22

Jerzy Laube

ANALIZATOR SYGNATUR - UNIWERSALNY PRZYRZĄD DIAGNOSTYCZNY DO URZĄDZEŃ CYFROWYCH

1. WPROWADZENIE

Wprowadzenie na szeroką skalę urządzeń cyfrowych, a zwłaszcza mikroprocesorowych, stwarza wiele nowych problemów. Okazuje się mianowicie, że urządzenia te są bardzo trudne do naprawy. Testowanie urządzeń opartych o układy scalone małej i średniej skali integracji, o ile nie jest wsparte specjalizowanymi układami uruchomieniowymi, jest niezwykle żmudne i pracochłonne. W przypadku urządzeń mikroprocesorowych wręcz niezbędne jest posiadanie złożonych i kosztownych urządzeń diagnostycznych, takich jak analizatory logiczne, emulatory itp. Istnieje zatem z jednej strony pilna potrzeba uwzględniania problemów późniejszego testowania już na etapie projektowania - tzw. projektowanie dla potrzeb testowania /design for testability/, a z drugiej strony potrzeba opracowania prostych /nie wymagających wysoko kwalifikowanych kadr/ oraz w miarę tanich i uniwersalnych metod testowania.

Wydaje się, że najbardziej obiecującą propozycją w tej dziedzinie jest tzw. analiza sygnatur /signature analysis/. Jest ona od 1977 r. propagowana przez firmę Hewlett-Packard, która też wprowadziła pierwsze urządzenie analizujące: Signature Analyzer 5004A, a następnie udoskonaloną wersję: Signature Multimeter 5005A, będący połączeniem analizatora sygnatur i multimetru cyfrowego. W ostatnim czasie analiza sygnatur zdobywa sobie, szczególnie w USA, coraz większe uznanie.

W Zakładzie Z-24 Ił wykorzystano metodę analizy sygnatur do testowania Analizatora Połączeń Telefonicznych "Artus", który ma być produkowany w Wydziale Konstrukcyjno-Warsztatowym Ił. W tym celu skonstruowany został specjalny przyrząd

diagnostyczny tzw. analizator sygnatur. Analizator sygnatur jest jednak przyrządem uniwersalnym i może znaleźć szersze zastosowanie do testowania układów cyfrowych. Ze względu na to, że stosowanie analizatora wymaga uwzględnienia specjalnych wymagań analizy sygnatur już na etapie projektowania nowych wyrobów, z zasadami jej stosowania powinni w pierwszej kolejności zapoznać się projektanci urządzeń cyfrowych.

2. OGÓLNE ZASADY ANALIZY SYGNATUR

Analiza sygnatur, ogólnie rzecz biorąc, polega na analizowaniu długich ciągów cyfrowych, z jakimi spotykamy się na wszystkich węzłach urządzenia cyfrowego podczas jego pracy. Analizy tej dokonuje się za pomocą odpowiedniego przyrządu - analizatora sygnatur. Analizator, opracowany w Z-24, zamienia podany na jego wejście ciąg cyfrowy w czytelną dla człowieka postać tzw. sygnatury, która składa się z czterech znaków kodu heksadecymalnego, z tym że ze względu na lepszą czytelność na wyświetlaczu 7-segmentowym stosuje się nietypowy zbiór tych znaków, a mianowicie: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, C, F, H, P, U. Różnych sygnatur może być $16^4 = 65,536$. Sygnatura jest w procesie testowania jak gdyby wizytówką danego węzła sieci. Przez porównanie sygnatury otrzymanej z analizatora z sygnaturą wzorcową, podaną w dokumentacji, można stwierdzić prawidłową pracę danego węzła. Sygnatury wzorcowe są natomiast otrzymywane empirycznie w wyniku badania za pomocą analizatora urządzenia pracującego prawidłowo, zbadanego uprzednio innymi metodami. Po wyodrębnieniu fałszywej sygnatury, śledząc sygnatury wstecz drogi sygnału, można wyodrębnić układ scalony pracujący nieprawidłowo.

Podczas testowania urządzenie testowane musi być pobudzone do pracy, aby wystąpiły niezbędne zmiany stanów badanych węzłów. Konieczny jest do tego celu zespół cyfrowych ciągów pobudzających. Muszą one być okresowe, tak aby kolejne pomiary na tym samym węźle dawały powtarzalne sygnatury. Ciągi te podaje się na wejściowe węzły badanej sieci logicznej

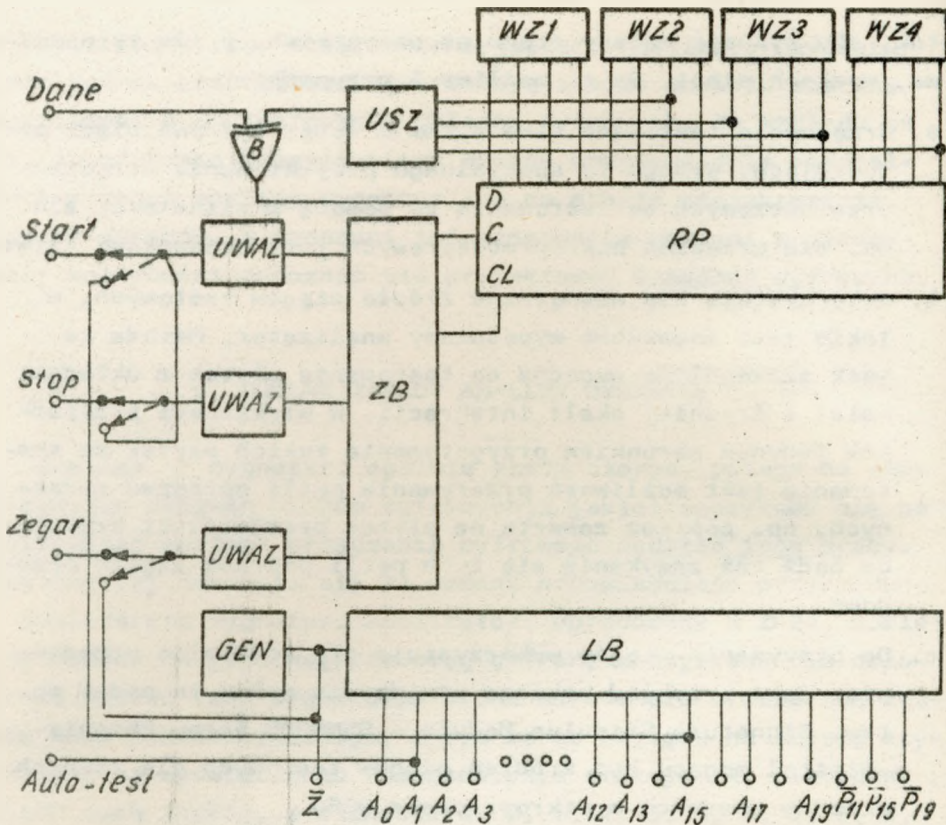
tak, aby wymusić zmiany stanu na wszystkich, o ile to możliwe, węzłach sieci. Są tu możliwe 3 przypadki:

- a. Urządzenie testowane samo wytwarza sobie cyfrowe ciągi pobudzające. Wymaga to specjalnego projektowania urządzeń przeznaczonych do testowania za pomocą analizatora, ale np. dla urządzeń mikroprocesorowych jest stosunkowo łatwe.
- b. Wykorzystuje się wewnętrzne źródło ciągów testowych, w jakie jest dodatkowo wyposażony analizator. Metoda ta jest szczególnie wygodna do testowania płytek z układami małej i średniej skali integracji. W większości przypadków jedynym warunkiem przygotowania takich płytek do testowania jest możliwość przerywania pętli sprzężeń zwrotnych, np. poprzez zawarte na płytce przełączniki krosujące bądź też zamykanie się tych pętli poprzez złącze zewnętrzne.
- c. Do otrzymania ciągów wykorzystuje się oddzielne urządzenia. Jako przykład takiego urządzenia można tu podać np. tzw. Signature Stimulus Module - SSM8080 firmy Phoenix-Digital mogący być źródłem ciągów testowych dla różnych układów opartych o mikroprocesor 8080.

3. ZASADA DZIAŁANIA ANALIZATORA SYGNATUR

Budowa, opracowanego w Z-24 analizatora sygnatur /rys.1/ jest następująca. Składa się on z dwóch głównych części: części analizującej badany węzeł oraz wewnętrznego źródła ciągów testowych.

Część analizująca składa się z: 16-bitowego rejestru przesuwającego RP zaopatrzonego w odpowiednią pętlę sprzężenia zwrotnego USZ, bramki B, zespołu czterech wyświetlaczy 7-segmentowych z dekoderni WZ1, WZ2, WZ3, WZ4 dołączonych do wyjść rejestru i tworzących sygnaturę, z zespołu bramkującego ZB sterującego procesem wczytywania informacji z wejścia DANIE oraz z trzech identycznych układów wybierania aktywnego zbocza UJAZ.



Rys. 1. Schemat blokowy analizatora sygnatur
 UWAZ - układ wybierania aktywnego zbrocza, USZ - układ sprzężenia zwrotnego, WZ - wyświetlanie znaku, ZB - zespół bramkujący, RP - 16-bitowy rejestr przesuwający, GEN - generator taktujący, LB - licznik binarny

Wewnętrzne źródło ciągów testowych stanowi generator taktujący GEN i wielobitowy licznik binarny LB.

W przypadku korzystania z zewnętrznego źródła ciągów testowych urządzenie działa następująco: Sygnały START, STOP, ZEGAR są dostarczane z badanego urządzenia. Sygnał START inicjuje proces wpisywania do rejestru danych z badanego węzła w takt impulsów na wejściu ZEGAR. Proces rejestracji, trwający na ogół ułamek sekundy, kończy sygnał STOP. Następnie proces rejestracji zostaje przez zespół bramkujący wstrzymany na okres około 2 sekund w celu umożliwienia odczytu sygnatury. Potem rejestr zostaje wyzerowany i rejestracja zaczyna się od nowa. Kolejne rejestracje na tym samym węzle dają

identyczne sygnatury ze względu na okresowość ciągów testowych. Układ sprzężenia zwrotnego USZ składa się z 4-wejściowej sumy modulo 2, podłączonej jak na rys. 1 /powszechnie obowiązujący standard/. Zadaniem USZ jest wydłużenie cyklu rejestru przesuwającego. Sygnał z USZ, który jest uzależniony od poprzedniej zawartości rejestru jest podawany na wejście bramki exclusive-or oznaczonej B. Na drugie wejście tej bramki podawany jest sygnał z badanego węzła. W zależności od tego czy sygnał z USZ jest zerem, czy jedynką logiczną - dane z badanego węzła wpisują się w formie niezanegowanej bądź zanegowanej. W rezultacie analizator może rozróżniać bardzo długie ciągi cyfrowe. Prawdopodobieństwo uzyskania różnych sygnatur dla różnych ciągów cyfrowych wynosi 99,998%.

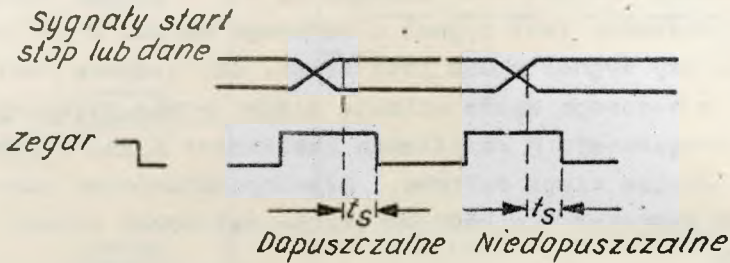
Źródło wewnętrznych ciągów testowych może być wykorzystane w przypadku, gdy urządzenie badane nie ma możliwości generowania własnych ciągów testowych /zwłaszcza układy MSI i SSI/. Rozszerza to zakres zastosowań przyrządu. Testowanie z wykorzystaniem wewnętrznego źródła zostanie bliżej opisane w pkt. 5.

Analizator jest również zaopatrzony w układ samotestujący umożliwiający sprawdzenie prawidłowości jego działania. Do tego celu wykorzystuje się gniazdo AUTO-TEST. Zwarcie tego gniazda z wejściem DANE przy jednoczesnym załączeniu wewnętrznego źródła ciągów testowych daje charakterystyczną sygnaturę, której występowanie dowodzi prawidłowego działania przyrządu.

Wejścia analizatora ZEGAR, START, STOP, DANE charakteryzują się stosunkowo dużą opornością wejściową i małą pojemnością wejściową, praktycznie więc nie zakłócają one pracy typowych układów TTL lub układów MOS kompatybilnych z TTL. Jednak, aby proces pomiarowy mógł przebiegać prawidłowo, sygnały dostarczane do wejść analizatora powinny spełniać określone warunki.

Ogólnie można powiedzieć, że jeśli chodzi o wejście ZEGAR, to podstawowym parametrem jest tu górna częstotliwość grani-

czna, natomiast pozostałe wejścia są scharakteryzowane przez tzw. czas ustalania t_g względem sygnału ZEGAR. Jest to minimalny czas poprzedzający aktywne zbocze sygnału zegarowego, w ciągu którego na rozpatrywanym wejściu stan logiczny musi być ustalony. Zagadnienie to ilustruje rys. 2.



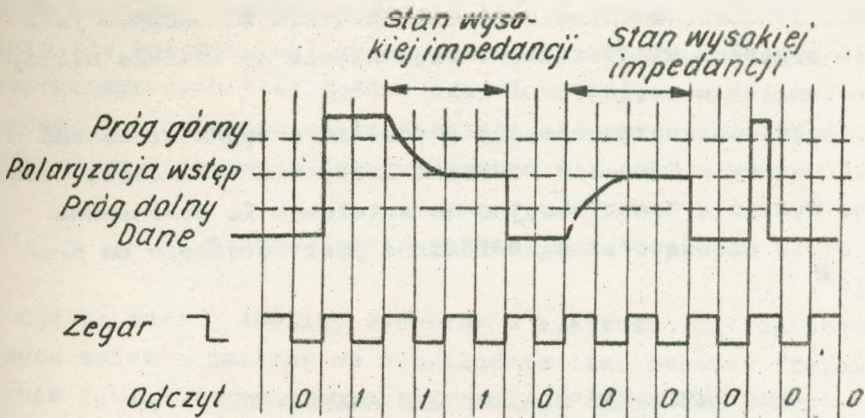
Rys. 2. Ilustracja czasu ustalania t_g

Sygnal podawany na wejście ZEGAR powinien być zupełnie wolny od przypadkowych zakłóceń impulsowych, gdyż ze względu na stosunkowo dużą dopuszczalną szybkość działania analizatora mogą one spowodować nieprawidłową rejestrację. Natomiast pozostałe wejścia analizatora są w pewnym zakresie odporne na zakłócenia impulsowe.

Wejścia START i STOP są wejściami typu "zatrzaśk" bramkowanymi sygnałem ZEGAR. Wymagane jest więc, aby były one wolne od zakłóceń jedynie w tej połowie okresu zegarowego, która bezpośrednio poprzedza aktywne zbocze zegara /o ile jest to przedział czasu nie mniejszy od czasu ustalania t_g /.

Informacja z wejścia DANE jest natomiast wczytywana w momencie pojawienia się aktywnego zbocza zegara /wejście wyzwalone zboczem/; wymagane jest więc tu jedynie przestrzeganie czasu ustalania t_g .

Wejścia analizatora są ponadto w pewnym zakresie odporne na zakłócenia statyczne. Dzięki pętli histerezy oraz wstępnej polaryzacji wejść możliwy jest również jednoznaczny odczyt informacji z węzłów typu trójetanowego. Działanie analizatora w przypadku podłączenia wejścia DANE do węzła tego typu ilustruje rys. 3.



Rys. 3. Wczytywanie danych z węzła typu trójstanowego

W przypadku systemów mikroprocesorowych źródłem sygnału ZEGAR są najczęściej zegar systemu bądź sygnały odczytu lub zapisu danych. Dla mikroprocesora 8080 będą to $\Phi 2/TTL$, DBIN, MEMR, MEMW.

Typowymi źródłami sygnałów START i STOP są natomiast najwyższe /często nade używane/ bity adresowe, wyjścia dekodowników adresowych, linie "chip enable", linie potwierdzenia zgłoszenia przerwania, sterowane softwarowo flagi, rejestry i porty WE-WY; przy czym możliwe jest pobieranie zarówno STARTU jak i STOPU z tego samego węzła i utworzenie tym samym jednego sygnału START/STOP.

W przypadku mikroprocesora 8080 szczególnie dogodnym źródłem sygnału START/STOP jest często nie wykorzystywany w systemie sygnał INTE, który może być ustawiany programowo. Natomiast przy stosowaniu jako źródła sygnału START/STOP linii adresowych lub dekodowników adresowych należy zwrócić uwagę na to, że mikroprocesor 8080 w czasie trwania niektórych stanów T_4 , T_5 ma nieokreślony stan szyny adresowej. Można to traktować jako występowanie zakłóceń na sygnale START/STOP i dzięki opisanym powyżej własności wejść START i STOP analizatora, zakłócenia te mogą być pominięte przez odpowiedni dobór sygnału ZEGAR /np. przez podanie na wejście ZEGAR sygnału MEMR i ustawienie jako aktywnego zbocza narastającego/. Ogólnie jednak w przypadku mikroprocesora 8080 należy raczej

uniknąć, o ile to możliwe, stosowania linii adresowych jako źródła sygnałów START i STOP. Zagadnienie to zostało bliżej przedstawione w pkt. 4.

Prototyp opisanego powyżej analizatora sygnatur został skonstruowany w Gdańskim Oddziale Instytutu Łączności. Obecnie na Wydziale Konstrukcyjno-Warsztatowym IŁ wykonywana jest seria próbna oraz przewidziane jest wdrożenie do produkcji*.

4. PROJEKTOWANIE DLA POTRZEB ANALIZY SYGNATUR W PRZYPADKU URZĄDZEŃ MIKROPROCESOROWYCH

4.1. Ogólne zasady

Najbardziej ogólne zasady, jakich należy przestrzegać w przypadku testowania urządzeń cyfrowych metodą analizy sygnatur, są następujące:

1. Powinny być wytworzone odpowiednie, okresowe testowe ciągi pobudzające.
2. Powinny być przerwane ścieżki sprzężeń zwrotnych /aby można było na podstawie śledzenia sygnatur zlokalizować błąd/.
3. Powinny być dostarczone do analizatora sygnały START, STOP i ZEGAR.

Szczególnie wygodną z punktu widzenia analizy sygnatur klasą urządzeń są urządzenia mikroprocesorowe, dla których spełnienie powyższych zasad jest stosunkowo proste. Dzięki strukturze szynowej stosunkowo łatwo jest przrwać pętle sprzężeń zwrotnych, które w więkzości przypadków zamykają się przez szynę danych. Ponadto programowalność mikroproce-

* Orientacyjna cena wynosi 30 tys. zł. W sprawie zakupu można kontaktować się z Wydziałem Konstrukcyjno-Warsztatowym Gdańskiego Oddziału IŁ bądź z autorem /tel. 418091 w. 36/.

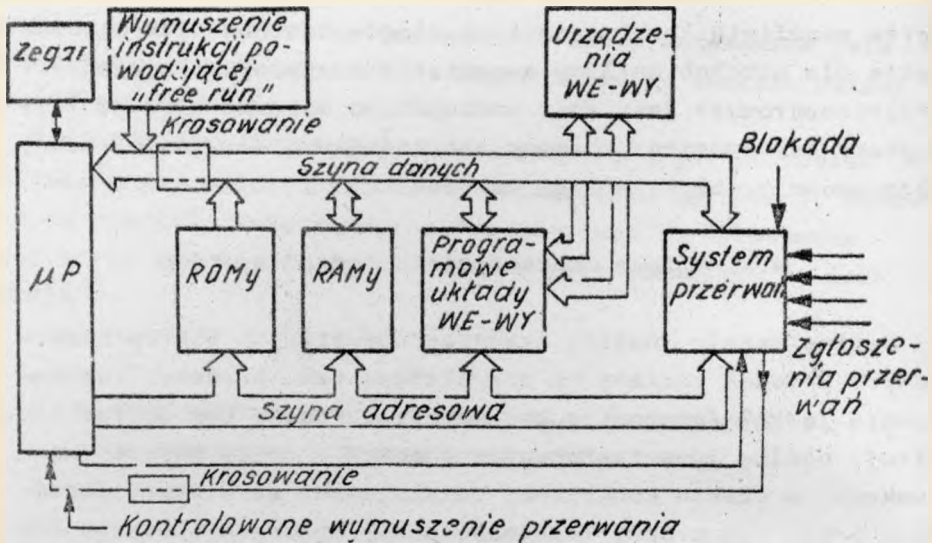
sorów umożliwia łatwą generację ciągów testowych. Projektowanie dla potrzeb analizy sygatur w przypadku urządzeń mikroprocesorowych jest dość szczegółowo opisane w [1] i [2]. W pracy tej omówione zostaną natomiast wybrane, najbardziej podstawowe problemy z tego zakresu.

4.2. Proces "rozszerzania jądra" systemu

Ogólne zasady analizy sygatur w systemach mikroprocesorowych opisane zostaną na przykładzie tzw. procesu "rozszerzania jądra" /expanding the kernel/. Proces ten dobrze ilustruje ogólną ideę testowania, chociaż w rzeczywistych warunkach, w czasie konkretnej działalności serwisowej określona przez ten proces kolejność poszczególnych operacji może nie być najbardziej optymalną.

Aby system mikroprocesorowy mógł prawidłowo pracować, muszą być sprawne w kolejności: zasilacz, zegar, mikroprocesor, poszczególne elementy podłączone do szyny danych oraz reszta systemu. Zasilaczem i zegarem nie będziemy się więcej zajmować ze względu na to, że ich działanie może być łatwo sprawdzone tradycyjnymi metodami. Przez "rozszerzanie jądra" będziemy rozumieć proces stopniowej rozbudowy systemu przez dodawanie do mikroprocesora, będącego właściwym jądrem systemu, pozostałych elementów takich jak: pamięci ROM, RAM, urządzenia WE-WY itd. - z jednoczesnym sprawdzaniem po każdym kolejnym etapie rozbudowy, czy system jeszcze nadal pracuje poprawnie.

Na rys. 4 przedstawiono ogólny schemat systemu mikroprocesorowego. Dla potrzeb analizy sygatur dodatkowo umieszczono jeszcze zespół krosujący, który umożliwia przerwanie szyny danych pomiędzy mikroprocesorem a resztą systemu, zapewniono możliwość wymuszenia na szynie danych instrukcji powodującej cykliczny obieg przestrzeni adresowej oraz możliwość blokady systemu przerwań. Ponadto wszystkie elementy mające dostęp do szyny danych umieszczono na podstawkach oraz część przestrzeni pamięci ROM przeznaczono na specjalny program testujący.



Rys. 4. Schemat systemu mikroprocesorowego przystosowanego do wymagań analizy sygnatur

Proces rozszerzenia jądra w powyższym systemie można podzielić na dwa zasadnicze etapy: rodzaj pracy "free run" oraz testowanie z użyciem programu testującego.

1. Rodzaj pracy "free run"

Tryb "free run", czyli praca z otwartymi pętlami sprzężeń zwrotnych, polega na wymuszeniu na pewnej możliwie małej liczbie węzłów sieci cyfrowej odpowiednich pobudzeń z jednoczesnym przerwaniem ścieżek sprzężeń zwrotnych tak, aby pobudzenia te mogły rozchodzić się swobodnie po całej sieci logicznej. Wymuszenie rodzaju pracy "free run" w systemie mikroprocesorowym polega na przerwaniu szyny danych oraz wymuszeniu na wejściach szyny danych mikroprocesora instrukcji powodującej cykliczny obieg przestrzeni adresowej. Oprócz tego należy wyjąć z podstawek /ewentualnie zablokować na drodze elektrycznej/ wszystkie RAMy, programowe urządzenia WE-WY, zablokować system przerwań i początkowo usunąć także ROMy.

Jedną z możliwości uzyskania rodzaju pracy "free run" - charakteryzującą się tym, że nie wymaga żadnych przeróbek w układzie badanym - jest wyjęcie z podstawek buforów szyny

danych i włożenie na ich miejsce odpowiednich atrap układów scalonych. Powinny one mieć wymuszone na tych nóżkach, które korespondują z wejściami szyny danych mikroprocesora /w przypadku systemu 8080 - kontrolera 8228 lub 8238/ odpowiednie stany 0 lub 1 logicznej, a pozostałe nóżki odłączone lub usunięte. Kombinacja stanów wymuszona w ten sposób na wejściach szyny danych mikroprocesora powinna tworzyć kod instrukcji wymuszającej "free run" /np. instrukcja NOP/. Odpowiednie atrapy układów scalonych bardzo prosto można wykonać nawet ze zwykłych układów scalonych. Powinny mieć one wejścia na tych nóżkach, które będą po włożeniu do podstawek korespondować z wejściami szyny danych mikroprocesora. Należy tylko odpowiednie nóżki zewrzeć z nóżką masy, połączyć przez opornik z nóżką zasilania bądź usunąć.

Tablica 1. Zespół sygnałów na szynie adresowej podczas pracy trybem "free run" wymuszonej instrukcją NOP

Podłączenia: START \downarrow A15; STOP \downarrow A15; ZEGAR \downarrow DBIN					
Sygnał	Sygnatura	Sygnał	Sygnatura	Sygnał	Sygnatura
+5V	0001	A5	0AFA	A11	1293
A0	UUUU	A6	UPFH	A12	HAP7
A1	5555	A7	52F8	A13	3C96
A2	CCCC	A8	HC89	A14	3827
A3	7F7F	A9	2H70	A15	755U
A4	5H21	A10	HPPO		

Proces "rozzerzania jądra" systemu rozpoczynamy od sprawdzenia samego mikroprocesora. Tablica 1 pokazuje sygnatyry zdjęte z szyny adresowej mikroprocesora 8080 pobudzonego do rodzaju pracy "free run" instrukcją NOP. Dzięki odpowiedniemu dobraniu sygnału podawanego na wejście ZEGAR można bez przeszkód pobierać sygnał START/STOP z linii adresowej A15, pomimo charakterystycznych dla tego mikroprocesora stanów

nieustalonych na liniach adresowych. Zapewnia to interwał pomiarowy analizatora równy pełnemu obiegowi przestrzeni adresowej. Sygnał podawany na wejście DANE analizatora jest próbkowany dokładnie raz dla każdego adresu.

Jeżeli sygnatury z tablicy 1 oraz sygnatury na wyjściach dekoderek adresowych były prawidłowe, możemy przejść do pierwszego etapu "rozszerzania jądra". W tym celu dodajemy do systemu /wkładając z powrotem do podstawki/ pierwszy ROM. ROM ten będzie w czasie przepatrywania przez mikroprocesor odpowiadającej mu przestrzeni adresowej, wytwarzał odpowiednie sygnały na liniach szyny danych. Sygnały te, dzięki przerwaniu szyny danych, nie będą jednak zwrotnie oddziaływały na mikroprocesor. Dlatego sygnatury zdjęte z szyny danych będą charakteryzowały zawartość ROMu i mogą posłużyć do sprawdzenia jego działania. Jeżeli wszystko było w porządku dodajemy drugi ROM i ponownie sprawdzamy sygnatury na szynie danych. Sygnatury te /inne niż poprzednio/ będą charakteryzowały łącznie pierwszy i drugi ROM. Z wszystkimi następnymi ROMami postępujemy analogicznie.

Zgodność otrzymanych sygnatur z wzorcowymi oznacza, że: a/ mikroprocesor może obiegać prawidłowo przestrzeń adresową, co - jak dowodzi statystyka różnych uszkodzeń mikroprocesorów - oznacza, że jest duże prawdopodobieństwo, że może on i inne swoje funkcje wypełniać prawidłowo [1]; b/ obwody dekodowania adresów są dobre; c/ ROMy są dobre; d/ szyna danych jest wolna od błędów. Wszystko to razem pozwala przejść do drugiego etapu testowania, z użyciem programu testującego.

2. Testowanie z użyciem programu testującego

Pamięci typu RAM nie nadają się do testowania za pomocą "free run", gdyż ich zawartość nie jest wówczas ściśle zdefiniowana. Testowanie RAMów polega na wykonywaniu cyklicznego programu testującego, który zawiera instrukcje zapisu i odczytu. Aby to umożliwić, zwieramy z powrotem przerwany szynę danych i usuwamy wymuszenie instrukcji powodującej "free run". Program testujący może być zawarty:

- w specjalnym ROMie testowym
- lub zajmować część przestrzeni adresowej w jednym z normalnych ROMów systemu.

Jeżeli jest specjalny ROM testowy, wkładamy go w miejsce normalnego ROMu zajmującego początkową część przestrzeni adresowej; jeżeli natomiast program testujący mieści się w części normalnego ROMu należy wówczas, np. przez odpowiednie przekrosowanie adresu, umożliwić restart systemu od tego miejsca, w którym znajduje się początek programu testującego.

Dalszy przebieg testowania wygląda następująco: Podłączamy odpowiednio sygnały START/STOP i ZEGAR analizatora /będzie to dokładniej opisane w pkt. 4.3/. Poprzez reset inicjujemy wykonywanie się pętli programu testującego i sprawdzamy sygnatury na szynie danych. Jeżeli są one prawidłowe, upewniamy się, że wszystkich połączeń dokonaliśmy prawidłowo i z jeszcze większym prawdopodobieństwem możemy uważać, że mikroprocesor działa sprawnie.

Wkładamy do podstawki pierwszy RAM, sprawdzamy na szynie danych odpowiadający mu komplet sygnatur. Podobnie postępujemy z innymi RAMami.

Następnie wkładamy kolejno do podstawek programowe układy WE-WY, a program testujący zmusza je do wysyłania danych w obu kierunkach. Aby można było przetestować właściwe działanie transmisji w kierunku do mikroprocesora, trzeba zasymulować napływ do programowych układów WE-WY danych z zewnątrz. Dane te muszą być jednak ściśle zdeterminowane i zsynchronizowane z programem testującym, aby otrzymywane sygnatury były powtarzalne. Dane takie można uzyskiwać np. poprzez odpowiednie skrosowanie wyjść rejestrów, kontrolowanych przez mikroprocesor i uprzednio sprawdzonych. Program testujący, oprócz testowania programowych urządzeń WE-WY, może również służyć jako źródło ciągów testowych umożliwiających badanie metodą analizy sygnatur dołączonych do nich urządzeń WE-WY.

Jeżeli w systemie mikroprocesorowym istnieje system przerwań, również i on powinien być testowany. W normalnych warunkach żądania przerwań są zgłaszane asynchronicznie, a przerwanie może nastąpić w dowolnym momencie aktualnie rea-

lizowanego programu. Ponadto system przerwań wraz z mikroprocesorem tworzy układ z zamkniętej pętli, która - aby system mógł być efektywnie testowany - powinna być rozwarta. Wszystko to razem wymaga pewnych dodatkowych zabiegów. Najdogodniejszym miejscem do rozwarcia pętli jest wejście przerywania do mikroprocesora /linia INT/. Po jej rozwarciu zgłoszenie przerywania przez obwody systemu przerwań nie będzie dochodziło do mikroprocesora. Natomiast pobudzanie wejścia INT mikroprocesora jest w czasie testowania dokonywane w sposób kontrolowany /poprzez przełącznik lub krosowanie/ powodując skok do specjalnego testowego podprogramu obsługi przerywania. Podprogram ten dostarcza pobudzających ciągów umożliwiających sprawdzenie obwodów systemu przerwań. Jeżeli system przerwań jest wielokanałowy, to wówczas komplety sygatur powinny być zbierane wielokrotnie, przy czym w pierwszym etapie odblokowuje się tylko jedno przerywanie przy wszystkich pozostałych zablokowanych, a w kolejnych etapach zwiększa się liczba odblokowanych /zasada "rozszerzania jądra"/. Podczas testowania systemu przerwań wszystkie elementy asynchroniczne wchodzące w jego skład powinny być zsynchronizowane lub zablokowane.

4.3. Ogólne zasady budowy programu testującego

Układanie odpowiedniego programu testującego nie jest skomplikowane. Często wystarczy 200-300 bajtów pamięci tak, że może on być nawet dopisany w już istniejącym systemie i umieszczony w jakiejś nie wykorzystywanej luźno obszarze pamięciowego.

Program ten powinien przede wszystkim wykonywać się cyklicznie i uaktywniać jeden raz w ciągu każdego cyklu odpowiednią linię sterującą, dostarczającą sygnał START/STOP do analizatora. Może to być np. sterowana programowo linia WE-WY, sygnał z dekodera adresowego lub wyższy nie używany w systemie bit adresowy.

W przypadku mikroprocesora 8080, ze względu na omówione już nieokreślone stany na liniach adresowych, pobieranie sy-

gnału START/STOP z tych linii jest podczas wykonywania się programu testującego niewygodne. Należy zwrócić uwagę, że podczas realizacji programu wykonywane są instrukcje różnego typu, a nie tylko odczyt jak w przypadku "free run". Dlatego stosowanie linii odczytu pamięci MEMR lub linii DBIN jako źródła sygnału ZEGAR nie jest korzystne /dane o instrukcjach zapisu byłyby przez analizator pomijane/. Korzystniej jest zamiast linii odczytu pamięci stosować zegar systemu / ϕ 2TTL dla μ P 68080/. Wówczas nie ma już jednak możliwości wyeliminowania wpływu nieokreślonych stanów na liniach adresowych, a zatem i pobierania z nich sygnału START/STOP bez dodatkowego hardware'u. Wygodnym źródłem sygnału START/STOP jest natomiast linia INTE, często w ogóle nie wykorzystywana i która może być w prosty sposób ustawiana programowo.

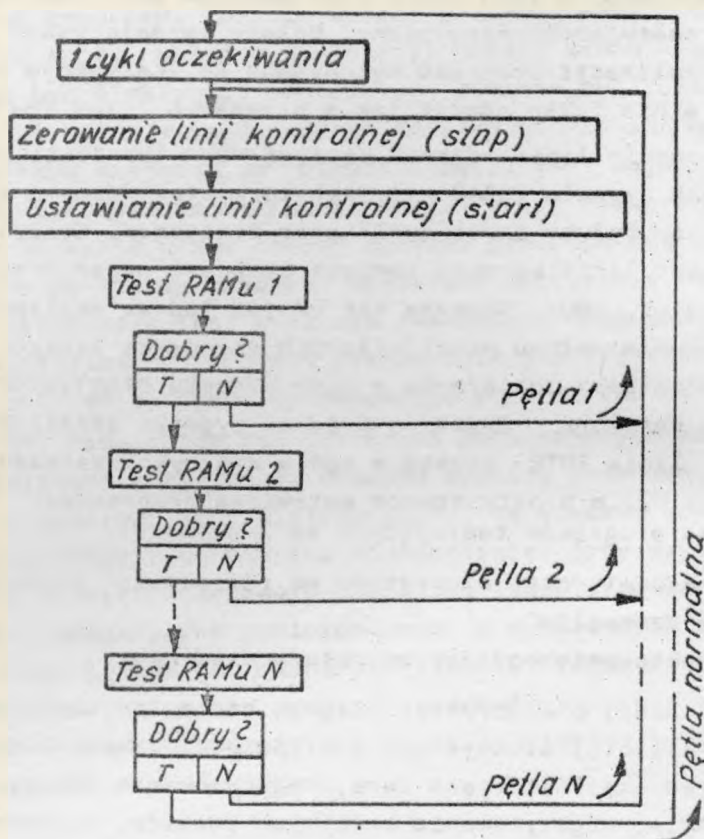
Pisanie programów testujących ma 2 aspekty:

- a/ dobór odpowiednich algorytmów do testowania poszczególnych podzespołów,
- b/ zaprojektowanie ogólnej struktury programu.

Jeśli chodzi o algorytmy testowe, to np. do testowania RAMów najczęściej stosuje się następujące: szachownica, wędrująca jedynka, wędrujące zero, magazynowanie zawartości ROMu w RAMie, magazynowanie kolejnych adresów. W przypadku testowania innych podzespołów stosuje się algorytmy odpowiednio dobrane do pełnionych przez nie funkcji. Bliższe dane na ten temat można znaleźć w [1], [2], [3], [5].

Rozpatrując ogólną strukturę programu testującego, warto zwrócić uwagę na programy typu "pętla na błędzie" /loop on fail/, które pozwalają na sprawniejszy przebieg testowania. Programy tego typu są stosowane jako narzędzie do samotestowania komputerów. Ogólną strukturę takiego programu, w przypadku przeznaczenia go do testowania RAMów pokazano na rys.5. Program ten zawiera na początku 2 rozkazy służące do uzyskania sygnału START/STOP, a dalsza jego część jest podzielona na fragmenty poświęcone testowaniu poszczególnych RAMów. Algorytm testujący jest tak skonstruowany, że daje w wyniku liczbę charakteryzującą badany podzespół /np. znany algorytm

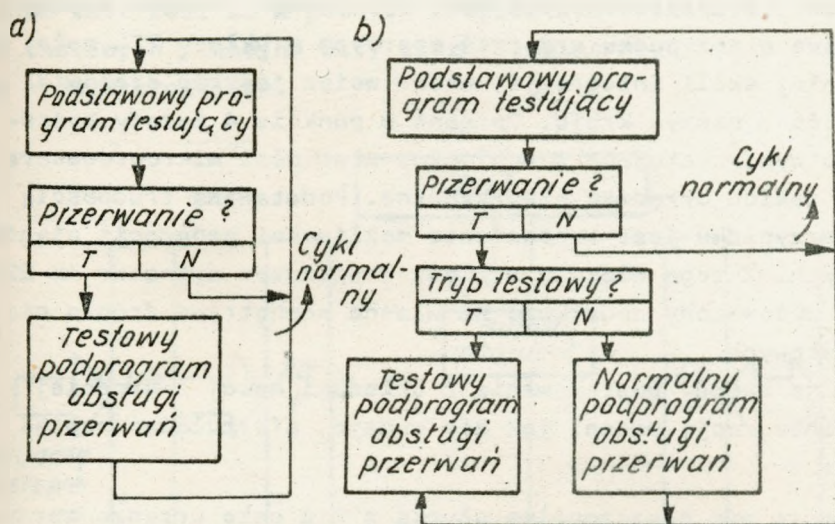
start programu



Rys. 5. Schemat programu typu "pętla na błędzie"

eweryfikacji sum kontrolnych/. Liczba ta jest następnie porównywana z liczbą charakterystyczną dla podzespołu prawidłowego i w zależności od wyniku porównania, program przechodzi dalej /jeśli było w porządku/ lub wraca z powrotem. W ten sposób już sama długość /mierzona w cyklach zegarowych/ pętli testowej wskazuje, który z elementów jest uszkodzony. Pozwala to na jego szybką identyfikację za pomocą sygnatury otrzymanej w wyniku dotknięcia sondą DANE analizatora do zasilania. Nie jest nawet konieczne wyjmowanie i kolejne wkładanie z powrotem do podstawek badanych elementów, o ile nie zachodzą takie uszkodzenia, które zakłócają pracę elementów sprawnych /np. sklejanie z zerem na nóżce układu scalonego

podłączonej do szyny danych/. Należy też zwrócić uwagę, że niezbędne jest w tym wypadku stworzenie katalogu błędnych sygnałów oraz że wykryty zostaje tylko pierwszy uszkodzony element i dopóki nie zostanie on naprawiony, nic nie można powiedzieć o stanie dalszych elementów.



Rys. 6. Sposoby organizacji testowego podprogramu obsługi przerwania

Jeżeli w systemie istnieje system przerwania, to część podstawowa programu testującego może się łączyć z testowym podprogramem obsługi przerwania na 2 sposoby przedstawione na rys. 6. W wersji z rys. 6a podprogram testowy zastępuje normalny program obsługi przerwania, tzn. zajmuje jego miejsce w przestrzeni adresowej poprzez wstawienie testowego ROMu bądź przekrośowanie adresu. W wersji z rysunku 6b istnieją natomiast dwie alternatywne możliwości obsługi przerwania, które są wybierane dzięki dodatkowemu predykatowi określającemu jeden z dwóch trybów pracy - normalny bądź testowy.

5. ZASADY POSTĘPOWANIA W PRZYPADKU URZĄDZEŃ REALIZOWANYCH W TECHNICIE MAŁEJ I ŚREDNIEJ SKALI INTEGRACJI

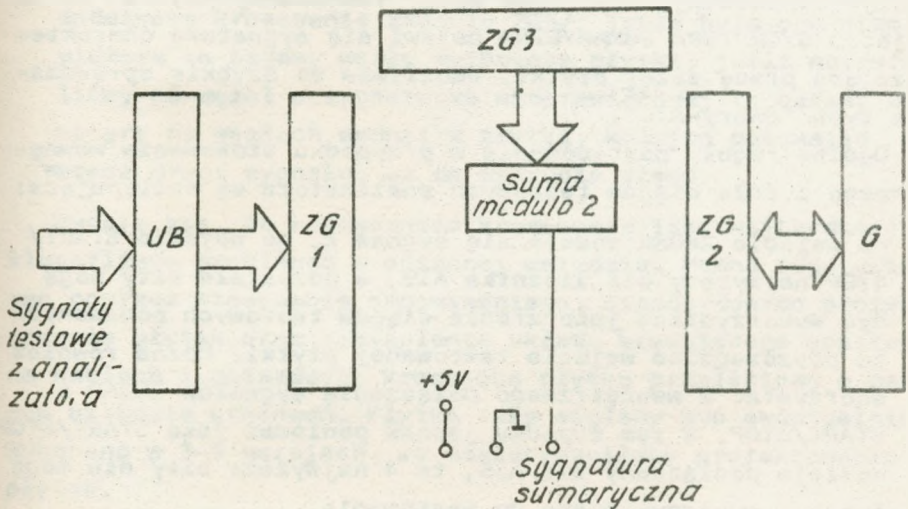
Analiza sygatur, jakkolwiek jest przeznaczona głównie do urządzeń makroprocesorowych może być jednak również bardzo przydatna w przypadku urządzeń opartych o układy TTL małej i średniej skali integracji, które wciąż jeszcze stanowią większość w naszym kraju. Opisane w punkcie 4 metody postępowania wykorzystujące specyficzne własności mikroprocesora są dla takich urządzeń nieprzydatne. Podstawową trudnością w tym przypadku jest zapewnienie możliwości generacji ciągów testowych. Z tego właśnie względu analizator sygatur AS 221 został wyposażony dodatkowo we własne wewnętrzne źródło ciągów testowych.

Ogólne zasady postępowania z układami małej i średniej skali integracji można, jak się wydaje, sformułować następująco:

- a. Testuje się poszczególne płytki a nie całe urządzenie, przy czym do testowania z użyciem analizatora sygatur nadają się płytki z logiką kombinacyjną i na ogół również sekwencyjną. /Trudności mogą sprawiać układy sekwencyjne o bardzo dużej liczbie przerzutników/.
- b. Źródłem ciągów testowych podawanych na wejściowe wężły testowanej płytki oraz sygnałów START, STOP i ZEGAR jest odpowiednie urządzenie zewnętrzne /tzn. nie sama testowana płytka/. W najprostszym przypadku można do tego celu użyć wewnętrznego źródła ciągów testowych analizatora. Jeśli natomiast dostępny jest komputer, to może on stanowić wygodne, bo programowalne, źródło ciągów testowych.
- c. Płytki przeznaczone do testowania za pomocą analizatora powinny być tak zaprojektowane, aby ich wewnętrzne pętle sprzężeń zwrotnych zamykały się przez złącze brzegowe lub aby było możliwe rozwieranie tych pętli w innych sposób. Ponadto wszystkie elementy asynchroniczne /przerzutniki

monostabilne, lokalne generatory/ powinny być odłączone, zablokowane bądź zezynchronizowane.

Rozpatrzmy obecnie najprostszy wariant, w którym wykorzystuje się wewnętrzne źródło ciągów testowych analizatora. Główną zaletą tego wariantu jest jego prostota. Należy się jednak zastrzec, że w pewnych przypadkach takie źródło ciągów testowych /kolejne bity licznika binarnego/ może okazać się niewystarczające.



Rys. 7. Schemat płytki pośredniczącej służącej do doprowadzenia sygnałów testowych z analizatora do badanej płytki

UB - układy buforujące; ZG1, ZG2, ZG3 - zespoły gniazd do krosowania; G - gniazdo dopasowane do badanej płytki

Do umożliwienia w miarę wygodnego doprowadzenia sygnałów testowych z analizatora /dostępnych na złączu ELTRA 881 025/ do badanej płytki, zaprojektowana została odpowiednia płytka pośrednicząca, przedstawiona na rys. 7. Płytkę tę zawiera 3 zespoły gniazd ZG1, ZG2, ZG3, pomiędzy którymi można dokonywać krosowania za pomocą giętkich kabelków zakończonych odpowiednimi miniaturowymi wtykami. Do zespołu ZG1 doprowadzone są

poprzez układy buforujące UB sygnały testowe z analizatora. Poszczególne gniazda zespołu ZG2 są połączone z łączówkami gniazda G dopasowanego do złącz brzegowych testowanych płytek. Poprzez odpowiednie krosowanie między ZG1 i ZG2 można doprowadzić sygnały testowe z analizatora do wejściowych węzłów testowanej płytki /podłączonej do gniazda C/. Zespół gniazd ZG3 pełni rolę dodatkową. Jest on połączony z wejściami wielobitowej sumy modulo 2 /3xUCY74180/. Jeżeli łączówki ZG3 zostaną odpowiednio skrosowane z tymi łączówkami ZG2, którym odpowiadają wyjściowe węzły testowanej płytki, to na wyjściu SYGNATURA SUMARYCZNA pojawi się sygnatura charakteryzująca pracę całej płytki. Umożliwia to szybkie sprawdzanie typu "dobry-zły".

Ogólne reguły postępowania w przypadku stosowania wewnętrznego źródła ciągów testowych analizatora są następujące:

- a. Na wejście ZEGAR podaje się sygnał Z, na wejście START/STOP najwyższy bit licznika A19, a pozostałe bity mogą być wykorzystane jako źródła ciągów testowych podawanych na poszczególne wejścia testowanej płytki. Można również skorzystać z wewnętrznego połączenia sygnałów ZEGAR i START/STOP. W tym wypadku jednak ponieważ jako START/STOP zostaje podłączony bit A15, to 4 najwyższe bity nie mogą już być wykorzystywane do testowania.
- b. W przypadku testowania układów kombinacyjnych sygnały testowe rozdziela się następująco:
 - Krosuje się kolejne bity licznika z poszczególnymi wejściami funkcji logicznych w dowolnej kolejności. Można zauważyć, że w trakcie pomiaru sprawdzana jest cała tablica prawdy analizowanej funkcji.
 - Dopuszczalne jest podłączanie tego samego bitu licznika na wiele wejściowych węzłów płytki, jeżeli są one wejściami różnych funkcji logicznych.
- c. W przypadku układów sekwencyjnych reguły są podobne z tym, że nie jest obojętna kolejność, ale:
 - wejścia zegarowe przerzutników powinny być podłączane do bitów o najwyższych częstotliwościach,

- wejścia zerujące i ustawiające przerzutników należy podłączać do bitów o najniższych częstotliwościach;
- w przypadku innych układów sekwencyjnych należy tak rozdzielać sygnały testujące, aby warunki pracy tych układów w czasie testowania były możliwie zbliżone do naturalnych.

d. Przebieg testowania jest następujący: po dokonaniu wg dokumentacji niezbędnych podłączeń i krosowań sprawdzamy kolejno: sygnaturę sumaryczną /jeśli zostało wykonane dodatkowe krosowanie zespołu ZG3/; jeśli była ona nieprawidłowa to badamy węzły wyjściowe płytki; jeśli natrafiliśmy na węzeł o sygnaturze nieprawidłowej to badamy sygnatury na węzłach wewnątrz płytki, kolejno posuwając się wstecz drogi sygnału, aż do wykrycia błędu.

Wydaje się, że konieczność krosowania jest najbardziej kłopotliwym problemem w opisanym metodzie. Można tego uniknąć poprzez stosowanie odpowiedniego, standardowego projektowania płytek oraz zastąpienie układu krosującego opartego na kablach i gniazdach, krosującą płytką przejściową o sztywnym układzie krosowań. Płytką taką mogłaby być ewentualnie wykonana w 3-5 wersjach, co uelastyczyłoby projektowanie płytek.

6. PODSUMOWANIE

Wydaje się, że analiza sygnatur może być bardzo użyteczna w odniesieniu do urządzeń cyfrowych różnego typu. Największe korzyści daje ona w odniesieniu do urządzeń mikroprocesorowych, które są jednocześnie najtrudniejsze do testowania za pomocą tradycyjnych technik. Może być ona jednak również bardzo pożyteczna przy testowaniu urządzeń opartych na układach scalonych mniejszej skali integracji przy odpowiednim projektowaniu /możliwość przerywania pętli sprzężeń zwrotnych/.

Podstawowymi zaletami metody są uniwersalność, bardzo pro-

ste oprzyrządowanie oraz możliwość posługiwania się nią bez wnikanía w zasadę działania testowanego urządzenia. Stosowanie analizy sygnatur wymaga uwzględnienia jej specyficznych wymagań w procesie projektowania, jednakże w wielu wypadkach możliwe jest, kosztem niewielkich przeróbek, spełnienie tych wymagań również dla urządzeń już istniejących.

W ostatnim czasie analiza sygnatur zdobywa sobie coraz większe uznanie także i w naszym kraju. Zaprezentowany w niniejszym artykule analizator może więc znaleźć, jak się wydaje, szerokie zastosowanie.

BIBLIOTEKA

Instytutu Łączności

5-8971

WYKAZ LITERATURY

1. A designers guide to signature analysis. Hewlett-Packard's Application Note 222.
2. Application articles on signature analysis. Hewlett-Packard's Application Note 222-2.
3. A signature analysis case study of a 6800-based display terminal. Hewlett-Packard's Application Note 222-11.
4. Gordon G., Nadig H.: Hexadecimal signatures identify troublespots in mikroprocessor systems. Electronics Magazine, 1977, No 5.
5. Implementing signature analysis for production testing with the HP 3060A Board Test System. Hewlett-Packard's Application Note 222-1.
6. Laube J.: Analiza sygnatur - nowe spojrzenie na projektowanie i testowanie urządzeń cyfrowych. PAK, 1980, nr 12.

Dotychczas ukazały się :

1. Białobrzeski R., Sońta S.: Zastosowanie testu chi kwadrat Pearsona do weryfikacji hipotezy statystycznej, na podstawie empirycznej gęstości prawdopodobieństwa. Grudzień 1977.
2. Blinkiewicz A., Mędrzycki B., Hutnik M., Samblerski R.: Zastosowanie pamięci kasetowej PK-1 do rejestracji danych w systemie komutacyjnym E-10. Styczeń 1978.
3. Orłowski A.: Optymalizacja układu ogranicznika dynamiki zwłaszcza dla radiofonii krótkofalowej. Luty 1978.
4. Frączek K.: Zasady opracowywania wymagań techniczno-eksploatacyjnych na urządzenia pomiarowe w resorcie łączności. Marzec 1978.
5. Białobrzeski R., Dudziewicz J.: Minimalna częstość próbkowania sygnału losowego przy pomiarze jego mocy średniej. Marzec 1978.
6. Lewandowski W.: Wprowadzenie komutacji teledacyjnej kanałów cyfrowych w powszechnej telefonicznej sieci komutacyjnej z centralami elektronicznymi E-10. Kwiecień 1978.
7. Dudziewicz J.: Ogólne wytyczne w sprawie prowadzenia i dokumentowania prac naukowo-badawczych wykonywanych w Instytucie łączności. Kwiecień 1978.
8. Stagrowski A.: Metoda detekcji i pomiaru impulsów o maksymalnych i minimalnych czasach trwania w ciągu. Maj 1978.
9. Chamski J.: System CTI-B dla maszyny cyfrowej R-10. Maj 1978.
10. Puchalski E.: Kompensator napięcia stałego stosowany w układach do sprawdzania przetworników termoelektrycznych i mikropotencjometrów. Czerwiec 1978.
11. Kozłowski A.: Elektroniczny sygnalizator przywołania abonenta w aparacie telefonicznym CB. Wrzesień 1978.
12. Stasiński L.: Wyładowania łukowe w.cz. na izolatorach odciągów pionowych anten radiofonicznych. Październik 1978.
13. Walaszek S.: Zastosowanie uogólnionego rozwiązania układu o trzech stanach do analizy niezawodności. Styczeń 1979.
14. Sońta S.: Aparatura automatyczna badań sieci łączności międzymiastowych systemu ABA-3. Luty 1979.

15. Godlewski P.: Język programowania badań w systemie ABA2 i ABA3. Marzec 1979.
16. Waśniewski A.: Kombinatoryczne aspekty planowania badań sieci telekomunikacyjnej za pomocą systemu ABA-3. Kwiecień 1979.
17. Brennek L., Lebidziuk B.: System edycji, przechowywania i translacji programów w języku SAWIK dla minikomputera MERA 305. Maj 1979.
18. Godlewski P.: Aparatura sterująca systemu badaniowego ABA-3 - architektura urządzenia. Czerwiec 1979.
19. Chamski J.: Centrum eksploatacji technicznej w systemie E 10. Lipiec 1979.
20. Porada M.: Komunikat o badaniach zakłóceń impulsowych w łącach telefonicznych. Sierpień 1979.
21. Sońta S.: Generacja sygnałów losowych niezależnych obciążających kanały telefoniczne. Wrzesień 1979.
22. Karwowska-Lamparska A.: Koncepcja systemu WIDEOTEKS. Październik 1979.
23. Kowalska J.: Próba eksploatacyjna automatycznej aparatury badaniowej ABA-2 - analiza wyników, wnioski. Listopad 1979.
24. Tyrowicz M.: System zdalnej rejestracji kontroli obiektów specjalnych - REKO - . Grudzień 1979.
25. Frydrych Z.: Uwagi o wymiarowaniu wiązek łączy międzycentralowych. Styczeń 1980.
26. Frydrych Z.: O niezawodności sieci telekomunikacyjnej. Luty 1980.
27. Kisto M.: Automatyzacja stacjonarnych pomiarów propagacyjnych. Marzec 1980.
28. Mieszczanek J.: Analiza i projektowanie oscylatorów kwarcowych pracujących w układzie Pierce'a-Colpitts'a. Kwiecień 1980.
29. Frydrych Z.: Niektóre problemy projektowania dróg kolejnego wyboru. Maj 1980.
30. Laube J.: Wybrane metody projektowania cyfrowych zespołów funkcjonalnych na przykładzie projektu generatora połączeń telefonicznych. Czerwiec 1980.

31. Kowalski Z.: Pasmowe tłumienności czwórników i ortotelefoniczne tłumienności odniesienia. Lipiec 1980.
32. Proga I.: Analiza i ocena odgromników zagranicznych oraz niezbędnego do nich osprzętu na podstawie badań i obserwacji w warunkach eksploatacyjnych. Sierpień 1980.
33. Godlewski P., Zejdel A.: System automatycznej kontroli obecności i ruchu załogi AKOR. Wrzesień 1980.
34. Waśniewski A.: Problem minimalizacji czasu badania sieci w systemie ABA-3. Październik 1980.
35. Kuśmirek Z.: Impedancja wewnętrzna źródła i jej pomiar. Listopad 1980.
36. Kowalski Z.: Zasady określania tłumienności pasmowej na podstawie danych punktowych. Grudzień 1980.
37. Kowalski Z.: Punktowe aproksymaty tłumienności pasmowej przy równomiernej gęstości wagi. Styczeń 1981.
38. Frydrych Z.: Wykorzystanie sygnałów informacyjnych dla poprawy jakości załatwiania ruchu w sieci telefonicznej. Luty 1981.
39. Lech J.: Analiza możliwości szacowania średniej 1-minutowej oraz 5-sekundowej mocy szumów w kanale telefonicznym na podstawie wyników pomiarów średniej 375-milisekundowej. Marzec 1981.
40. Strużak R.: O optymalnym przydziale mocy i częstotliwości radiokomunikacyjnym stacjom nadawczym. Kwiecień 1981.
41. Kawecki A.: Określenie kumulatywnego rozkładu prawdopodobieństwa natężeń opadów atmosferycznych w Polsce dla potrzeb radiokomunikacji. Maj 1981.
42. Trehcieński J.: Korzyści z wprowadzania cyfrowych centrów komutacyjnych do telefonicznych sieci strefowych. Czerwiec 1981.
43. Chamski J.: Metody badań oprogramowania użytkowego centrum eksploatacji technicznej w systemie komutacji elektronicznej E-10. Lipiec 1981.
44. Kotz F.: Problemy sterowania zapłonem tyrystorów w przekształtnikach wielofazowych. Sierpień 1981.

45. Flisek T., Klimczewska I.: Wpływ służby "zapamiętaj i przekaż" na wielkość generowanego i załadowanego ruchu w krajowej sieci telegraficznej. Wrzesień 1981.
46. Stankiewicz S.: Kalkulator - jednostka sterująca typu K77 automatycznego stanowiska pomiarowego. Październik 1981.
47. Kowalski Z.: Metody wyznaczania parametrów punktowych aproksymat tłumienności pasmowej. Listopad 1981.
48. Kowalski Z.: Metoda wyznaczania najefektywniejszych 3-punktowych aproksymat tłumienności pasmowej. Grudzień 1981.

Biblioteka

IZ

S-8971