

Henryk MAŁYSIAK
Bolesław POCHOPIEŃ
Antoni STARCZYŃSKI
Piotr STRÓŻYNA

Instytut Informatyki Czasu Rzeczywistego
Politechniki Śląskiej

SONDA EMULUJĄCA MIKROPROCESOR Z80 W SYSTEMIE RTDS

Streszczenie. Praca zawiera opis sondy emulującej mikroprocesor Z80 w systemie RTDS-8, zaprojektowanej, wykonanej i uruchomionej w Zespole Teorii Automatów Cyfrowych Instytutu Informatyki Czasu Rzeczywistego Politechniki Śląskiej. Mikroprocesor Z80 jest emulowany w pełnym zakresie swoich rzeczywistych charakterystyk, łącznie z procesem odwołania pamięci dynamicznych, dla dowolnej konfiguracji systemu prototypowego.

1. WPROWADZENIE

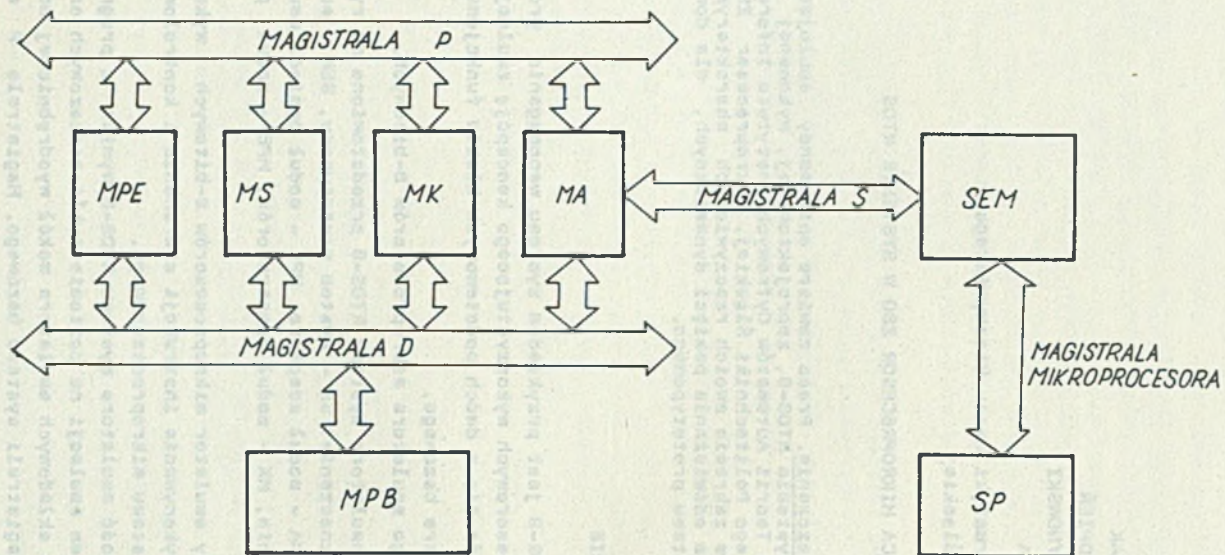
System RTDS-8 jest przykładem systemu wspomagania projektowania układów mikroprocesorowych wykorzystującego koncepcję emulacji układowej. System ten składa się z dwóch podstawowych bloków funkcjonalnych:

- mikrokomputera bazowego,
- uniwersalnego emulatora mikroprocesorów 8-bitowych.

Strukturę emulatora systemu RTDS-8 przedstawiono na rys. 1, przyjmując następujące oznaczenia: SP - system prototypowy, SEM - sonda emulująca mikroprocesor, MA - moduł adaptera, MPE - moduł pamięci emulowanej, MS - moduł śladowania, MK - moduł komparatorów, MPB - moduł procesora bazowego.

Uniwersalny emulator mikroprocesorów 8-bitowych wykorzystuje metodę forsowanego wykonywania instrukcji sterowania i kontrolowania stanu uruchamianego systemu mikroprocesorowego.

Uniwersalność emulatora systemu RTDS-8 wynika z programowego sterowania przebiegiem emulacji na poziomie cykli rozkazowych oraz skonfigurowania elementów składowych emulatora wokół wyodrębnionej magistrali P, niezależnie od magistrali systemu bazowego. Magistrala P emulatora posiada cechy szerokiej klasy magistral systemów mikroprocesorowych, a równocześnie zawiera specjalne linie statusowo-sterujące wymagane dla sterowania emulacją mikroprocesora oraz współpracy elementów składowych emulatora. W wyniku tego emulator systemu RTDS posiada stałą strukturę urządzeniową,



Rys. 1. Struktura emulatora systemu RTDS-8

a emulacja żądanego mikroprocesora wymaga połączenia właściwej sondy emulującej z danym mikroprocesorem oraz zainicjowania w systemie RTDS-8 odpowiedniego programu sterującego.

Koncepcja emulacji układowej polega na symulowaniu mikroprocesora w uruchamianym zastawie przez układ analogicznego mikroprocesora umieszczonego w sondzie umożliwiającej jego sterowanie i kontrolowanie aktualnego stanu. Moduł sondy stanowi układ o konstrukcji zależnej od typu emulowanego mikroprocesora. Z jednej strony moduł ten emuluje wszystkie sygnały mikroprocesora, wyprowadzając je bezpośrednio do podstawki mikroprocesora w systemie prototypowym, z drugiej zaś wypracowuje sygnały pewnej uniwersalnej magistrali S, będącej odpowiednikiem magistrali P, z pominięciem sygnałów wzajemnej współpracy modułów adaptera, komparatora i śladowania [2].

Sonda emulująca mikroprocesor Z80 w systemie RTDS-8 została zaprojektowana, wykonana i uruchomiona w Zespole Teorii Automatów Cyfrowych Instytutu Informatyki Czasu Rzeczywistego Politechniki Śląskiej.

2. SONDA EMULUJACA

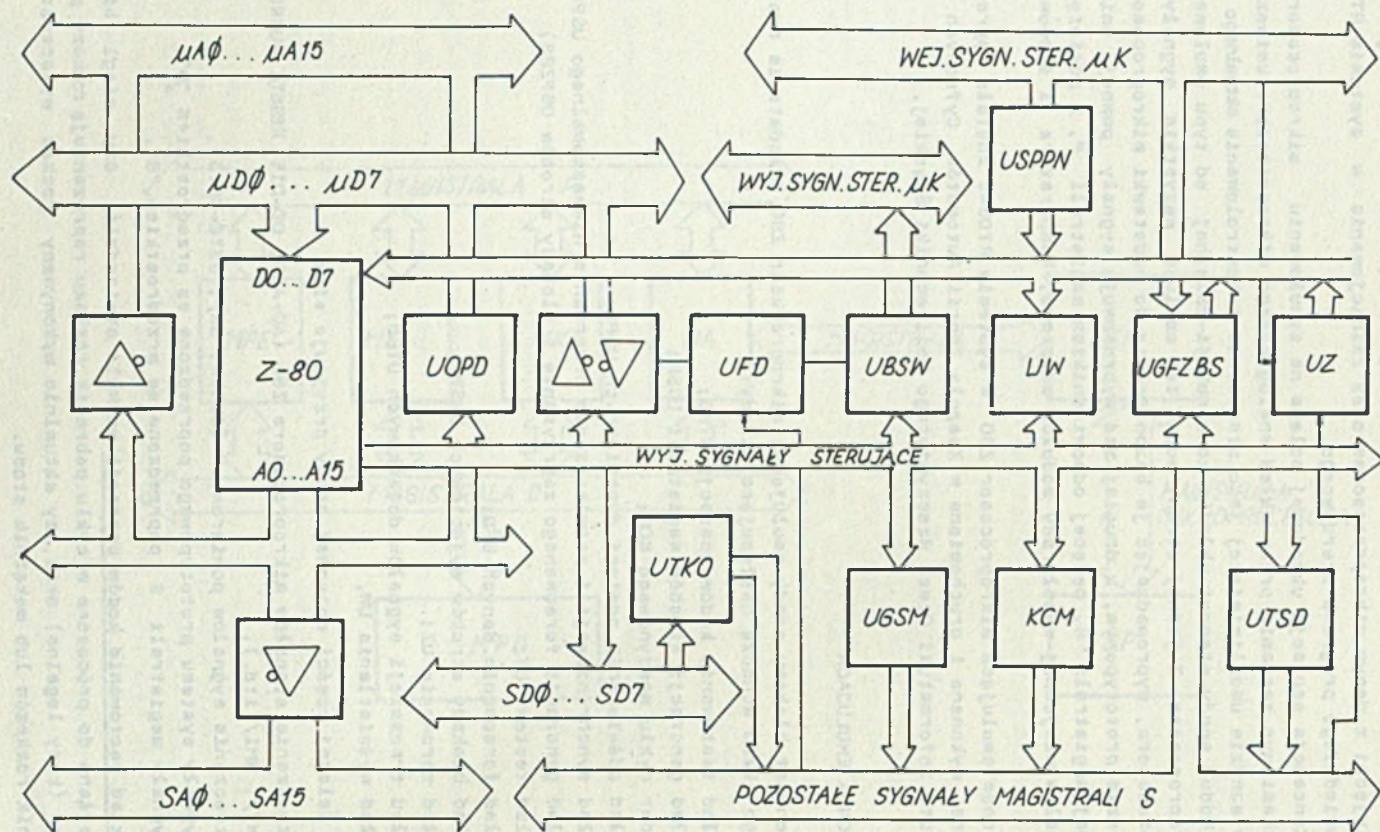
Schemat blokowy sondy emulującej mikroprocesor Z80 przedstawia rys. 2. W jego skład wchodzi następujące układy:

- układ testowania kodów operacji UTKO;
- układ generacji strobów magistrali UGSM;
- koder cyklu maszynowego KCM;
- układ odwołania pamięci dynamicznych UOPD;
- układ synchronizacji i potwierdzenia przerwania niemaszkalnego USPPN;
- układ generacji forsowanego zatrzymania i blokady strobów UGFZBS;
- układ testowy UT;
- układ forsowania danych UFD;
- układ blokady strobów wyjściowych UBSW;
- układ zerowania UZ;
- układ transmisji sygnałów dodatkowych UTSD;
- układ wyświetlania UW.

W dalszej części opisowej pracy przyjmuje się:

- a) oznaczenia sygnałów mikroprocesora Z80 ($A\phi$ -A15, $D\phi$ -D15, RESET/, BUSREQ/, INT/, NMI/ itd.),
- b) oznaczenia sygnałów pomiarowych (GUSER, GX/, GST ϕ -GST3),
- c) sygnały systemu prototypowego poprzedzone są przedrostkiem "μ",
- d) sygnały magistrali S poprzedzone są przedrostkiem "S".

Układ testowania kodów operacji pozwala stwierdzić, czy ciągi bitów przesyłane do procesora w cyklu pobrania rozkazu reprezentują rozkazy procesora (kody legalne) oraz czy aktualnie wykonywany rozkaz wykorzystuje licznik rozkazów lub wskaźnik stosu.



Rys. 2. Schemat blokowy sondy emulującej mikroprocesor Z80

Sygnaly wyjściowe układu sygnalizują odpowiednio:

SCP/ - wykonywanie rozkazu wykorzystującego licznik rozkazów,
SSP/ - wykonywania rozkazu wykorzystującego wskaźnik stosu,
SIL/ - wystąpienie kodu nielegalnego.

Sygnalami wejściowymi UTKO są:

SD ϕ ...SD7 - sygnaly magistrali danych,
RESET/ - sygnal zerowania systemu,
SMRD/ - strob odczytu z pamieci,
SOPF/ - strob pobrania rozkazu.

W skład układu wchodzi zasadniczo pamięci PROM (I-512x8; II, III, IV-256x4), licznik bajtów, demultiplekser wyboru bajtów i rejestr wyboru pamięci.

W pamięciach PROM zapisane są informacje o wszystkich kombinacjach słów 8-bitowych, jakie mogą pojawić się w cyklu pobrania rozkazu na magistrali danych.

Pamięć I umożliwia analizę pierwszego bajtu kodu operacji. Wykorzystuje się jedynie połowę pamięci. Pamięć II umożliwia analizę drugiego bajtu kodu operacji, o ile pierwszy bajt miał wartość \$DD lub \$FD. Pamięć III umożliwia analizę drugiego bajtu kodu operacji, o ile pierwszy bajt miał wartość \$ED. Pamięć IV umożliwia analizę drugiego kodu operacji, o ile pierwszy bajt miał wartość \$CB oraz czwartego bajtu dla rozkazów o kodach operacji na bajtach pierwszym, drugim i czwartym. Licznik bajtów pozwala wyodrębnić w czasie kolejna bajty przy rozkazach z wielobajtowym kodem operacji.

Licznik przełączany jest narastającym zboczem sygnału SMRD/. Demultiplekser wyboru bajtu dokoduje stan licznika bajtów. Wyjścia demultipleksersa doprowadzone są do wejść wybierających pamięci PROM tak, aby zapewnić odblokowanie pamięci odpowiadającej aktualnie analizowanemu bajtowi kodu operacji. Ponieważ trzeci bajt rozkazu nie zawiera kodu operacji, dlatego odpowiadające mu wyjście służy jedynie do przełączania licznika bajtów na czwarty bajt oraz wyzerowania rejestru wyboru pamięci. Demultiplekser strobowany jest sygnałem odczytu z pamięci SMRD/. Rejestr wyboru pamięci przechowuje informację o tym, która z pamięci PROM (II, III lub IV) ma być odblokowana podczas analizy drugiego bajtu kodu operacji. Informacja ta zostaje wpisana do rejestru z pamięci I na podstawie analizy pierwszego bajtu kodu operacji rozkazu. Wyjścia rejestru podane są na wyjścia wybierające pamięci II, III i IV. Pamięć IV aktywna jest podczas analizy bajtu drugiego (gdy pierwszy bajt równy był \$BC) oraz czwartego. Z tego względu rejestr wyboru pamięci zerowany jest przy przełączeniu licznika bajtów z bajtu trzeciego na czwarty. Umożliwia to odblokowanie pamięci IV niezależnie od poprzedniej zawartości rejestru.

Po wyzerowaniu układu sygnałem RESET/ licznik bajtów ustawia się na stan odpowiadający pierwszemu bajtowi. Podczas pierwszego cyklu pobrania

rozkazu odblokowana zostaje pamięć I. Dla części rozkazów o kodach jedno-bajtowych generowane są aktywne wartości sygnałów SSP/ i SCP/, a nieaktywna wartość sygnału SECB powoduje, że licznik bajtów pozostaje w stanie odpowiadającym pierwszemu bajtowi. Informacja wpisana do rejestru wyboru pamięci jest nieistotna. Dla rozkazów o kodach wielobajtowych sygnały SSP/ i SCP/ są nieaktywne, natomiast narastające zbocze sygnału SMRD/ przełącza licznik bajtów do stanu odpowiadającego drugiemu bajtowi. Ponadto do rejestru wyboru pamięci wpisana zostaje informacja, która pamięć ma być następnie odblokowana.

Jedno z wyjść demultipleksersa jest doprowadzone do wejścia wybierającego każdej z pamięci II, III i IV i w następnym cyklu odczytu z pamięci (SMRD/ - aktywny) zostaje odblokowana jedna z tych pamięci, zgodnie z informacją zapamiętaną w rejestrze wyboru pamięci. Generowane są aktywne wartości sygnałów SSP/, SCP/ i SIL/. Dla rozkazów o kodach 2-bajtowych nieaktywna wartość sygnału FOURTHB (sygnał wyjściowy pamięci II oznaczający rozkaz czterobajtowy - kod operacji na czwartym bajcie) sprowadza licznik bajtów do stanu początkowego odpowiadającemu pierwszemu bajtowi. Natomiast w przypadku, gdy zachodzi konieczność analizy czwartego bajtu rozkazu, sygnał FOURTHB/=0 zezwala na zmianę stanu licznika bajtów na odpowiadający trzeciemu bajtowi.

Trzeci bajt rozkazu zawiera argument, dlatego żaden PROM nie jest w kolejnym cyklu odczytu z pamięci odblokowywany. Jedynie zerowany jest rejestr wyboru pamięci PROM i licznik bajtów.

Podczas odczytu czwartego bajtu (zawierającego kod operacji) odblokowywana jest pamięć IV zawierająca sygnał STL/. Pamięć IV wykorzystywana jest dwukrotnie. Jej wyjście Q_1 (SILII - nielegalny bajt: drugi bajt kodu - pierwszego bajtu $\$CB$) odblokowywane jest wyłącznie przy analizie czwartego bajtu, natomiast wyjście Q_0 (SILIV - nielegalny bajt: czwarty bajt kodu) w obu przypadkach.

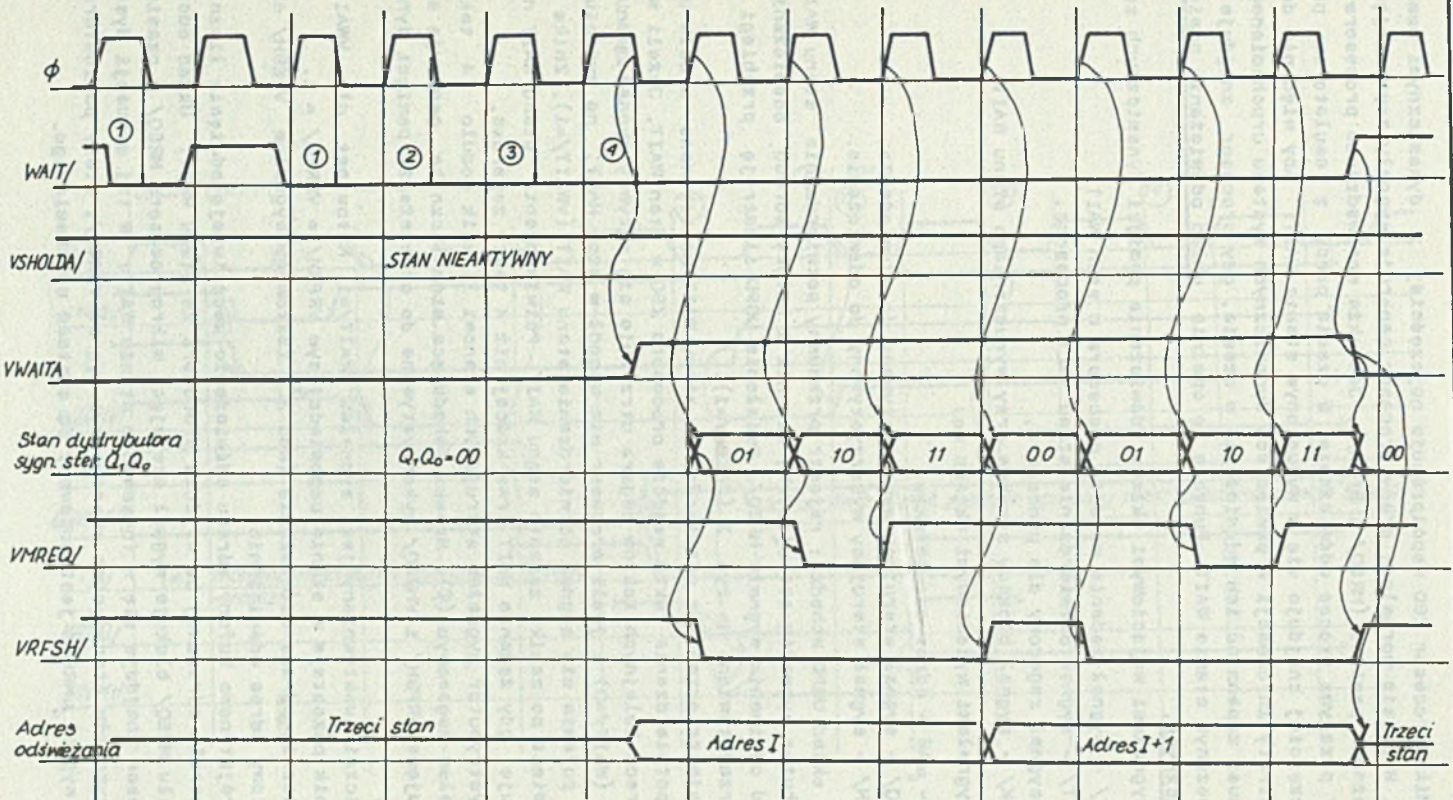
Układ generacji strobów magistrali S wypracowuje sygnały strobów magistrali S (SIOR/, SIOW/, SMRD/, SMWR/, SINTA/, SOPF/) na podstawie sygnału VNMIA i sygnałów sterujących procesora (M1/, MREQ/, IOR/, RD/, WR/).

Układ zbudowany jest z bramek serii superszybkiej. Aby zapewnić wymagania czasowe emulatora, zbocza niektórych sygnałów są opóźnione.

Stroby SIOW/ i SIOR/ zastąpiono jednym strobem oznaczającym cykl komunikacji z urządzeniem wejścia-wyjścia VIORQ/:

$$VIORQ/ \equiv SIOW/ \equiv SIOR/.$$

Koder cyklu maszynowego procesora zbudowany jest z pamięci PROM (256x4). Na wejścia adresowe pamięci podawane są sygnały sterujące procesora i sygnał potwierdzenia przerwania niemaskowalnego VNMIA. Zawartość pamięci stanowią 4-bitowe kody STC_0 - STC_3 cykli maszynowych. Został przewidziany również kod dla cyklu RFSH (odświeżania pamięci), jednakże w cyklu tym nie jest wypracowywany żaden strob magistrali S, zatem cykl ten nie będzie programowo identyfikowany.



Rys. 3. Przebiegi czasowe w układzie odświeżania pamięci dynamicznych

Mikroprocesor Z80 współpracuje najczęściej z dynamicznymi pamięciami RAM. W czasie normalnej pracy procesor generuje sygnały zapewniające odświeżanie pamięci (MREQ/ i RFSH/). Jednakże wprowadzenie procesora w stan WAIT przerywa proces odświeżania. W czasie pracy z emulatorem procesor najczęściej znajduje się w wymuszonym stanie WAIT. Aby więc nie dopuścić do utraty informacji w pamięciach dynamicznych systemu uruchamianego, sonda musi zapewniać ich odświeżanie w czasie, gdy procesor znajduje się w wymuszonym stanie WAIT. Funkcję tę spełnia układ odświeżania pamięci dynamicznych.

Sygnałami wejściowymi układu odświeżania pamięci dynamicznych są:

- WAIT/ - sygnał żądania wejścia procesora w stan WAIT,
- μ HALT/ - sygnał potwierdzenia stanu HALT procesora,
- ϕ - sygnał zegarowy dla procesora,
- VBLOK/ - sygnał blokady strobów przy wymuszeniach stanu WAIT.

Sygnałami wyjściowymi układu są:

- $\mu A \phi$ - $\mu A6$ - adres odświeżania,
- VMREQ/ - sygnał sterujący wykorzystywany do odświeżania,
- VRFSH/ - sygnał sterujący wykorzystywany do odświeżania.

W skład UOPD wchodzi: rejestr przesuwany potwierdzenia stanu WAIT, dystrybutor sygnałów sterujących, licznik i bufory adresu odświeżania oraz układ odcinania sygnału MREQ/. Działanie UOPD ilustrują przebiegi czasowe przedstawione na rys. 3 (VBLOK/=1).

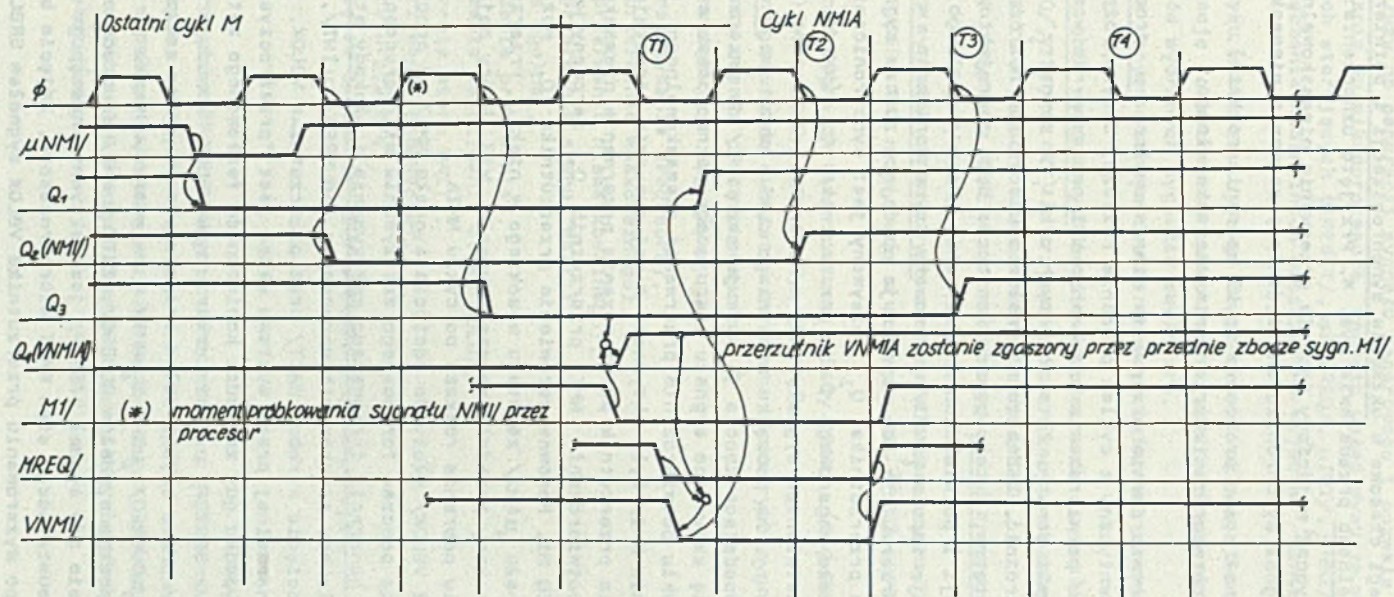
Rejestr przesuwany potwierdzenia stanu WAIT zastosowano w celu uzyskania potwierdzenia faktu wejścia procesora Z80 w stan WAIT. Jeżeli w ciągu czterech kolejnych taktów zegara utrzymuje się aktywny sygnał żądania stanu WAIT (WAIT/=0) i jeśli procesor nie wszedł w stan HALT, na wyjściu rejestru pojawia się sygnał potwierdzenia stanu WAIT (VWAIT/=1). Znika on natychmiast po zaniku żądania stanu WAIT. Potwierdzenie stanu WAIT nie występuje, gdy żądanie WAIT trwa krócej niż 4 takty zegarowe.

Dystrybutor sygnałów sterujących stanowi licznik modulo 4 taktowany sygnałem zegarowym (ϕ). Bramki dekodujące stan licznika generują sygnały sterujące VRFSH/ i VMREQ/ wykorzystywane do odświeżania pamięci dynamicznych.

Licznik uaktywniany jest sygnałem VWAIT/=1. Natomiast dla VWAIT/ = 0 licznik pozostaje w stanie odpowiadającym VRFSH/ = VMREQ/ = 1.

Licznik adresu odświeżania (mod 2^7) taktowany sygnałem VRESH/ określa 7-bitowy adres odświeżania.

Trójstanowe bufory adresu odświeżania umożliwiają odcięcie licznika od magistrali adresowej po wyjściu procesora ze stanu WAIT. Układ odcinania sygnału MREQ/ blokuje sygnał sterujący mikroprocesora MREQ/ w czasie, gdy procesor znajduje się w forsowanym stanie WAIT. W tej sytuacji dystrybutor sygnałów sterujących wypracowuje sygnał VMREQ/, który po zbuforowaniu jako sygnał μ MREQ/ jest podawany do systemu uruchamianego.



Rys. 4. Przebiegi czasowe w układzie synchronizacji i potwierdzenia przerwania niemaszkalnego

Przebiegi czasowa w układzie synchronizacji i potwierdzenia przerwania niemaskowalnego przedstawia rys. 4. Przyjęto oznaczenia:

- μ NMI/ - sygnał wejściowy żądania przerwania niemaskowalnego,
- NMI/ - sygnał synchronizowany żądania przerwania niemaskowalnego,
- M1/ - sygnał stanu procesora (cykl pobrania rozkazu),
- VNMIA/ - potwierdzenie przerwania niemaskowalnego.

Cykl potwierdzenia przerwania niemaskowalnego w procesorze Z80 jest niemal identyczny z cyklem pobrania rozkazu. Jedyne różnice polegają na tym, że w cyklu potwierdzenia przerwania NMI procesor nie wczytuje informacji z magistrali danych. Zatem obu tych cykli "z zewnątrz" praktycznie nie można rozróżnić. Jednakże dla celów emulacji ich rozróżnienie jest niezwykle istotne. Z tego powodu konieczne było zaprojektowanie układu synchronizacji i potwierdzenia przerwania niemaskowalnego. Przerzutnik Q_1 zapamiętuje impuls μ NMI/ generowany asynchronicznie w systemie uruchamianym. Przerzutnik ten rejestruje opadające zbocze sygnału μ NMI/. Następnie stan przerzutnika Q_1 wpisany jest synchronicznie opadającym zboczem sygnału zegarowego ϕ do przerzutnika Q_2 (NMI/). Zapewnia to utrzymanie ustalonego sygnału NMI/ w momencie próbkowania linii NMI/ przez procesor, tj. na początku ostatniego cyklu maszynowego w każdym cyklu M. Kolejne opadające zbocze ϕ wpisuje stan NMI/ do przerzutnika Q_3 . Pojawienie się aktywnego sygnału sterującego mikroprocesora M1/ oznacza początek cyklu potwierdzenia przerwania niemaskowalnego (jeżeli w momencie próbkowania NMI/ było aktywne). W tej sytuacji narastające zbocze sygnału M1 ustawia przerzutnik potwierdzenia przerwania niemaskowalnego Q_4 (VNMIA). W cyklu potwierdzenia NMI przerzutnik Q_1 zerowany jest sygnałem VNMIA/ = VNMIA MREQ M1. Ponowne ustawienie przerzutnika Q_1 możliwe jest po przejściu sygnału μ NMI/ ze stanu wysokiego w niski. Przerzutnik VNMIA zostanie wyzerowany przednim zboczem sygnału M1/, który pojawi się w najbliższym cyklu pobrania rozkazu po cyklu NMIA.

Sygnał VBLOK/ służy do odcięcia sygnału NMI/ od procesora. Sytuacja ta ma miejsce podczas forsowanego zatrzymania mikroprocesora.

Układ generacji forsowanego zatrzymania i blokady strobów składa się z układu blokady strobów mikroprocesora (μ BUSREQ/, μ INT/, μ NMI/, μ RESET/), układu odcięcia strobu WAIT/ oraz przerzutnika VBLOK.

Przy normalnej pracy systemu układ jest "przeźroczysty" dla strobów.

W przypadku gdy zachodzi konieczność forsowanego zatrzymania procesora, sygnał SFSTOP/ ustawia przerzutnik VBLOK. Powoduje to odcięcie sygnału μ WAIT/ przez szybki układ tranzystorowy oraz zatrzaśnięcie sygnałów μ RESET/, μ BUSREQ/, μ INT i μ NMI/ w rejestrze. Bramki i przerzutnik NMI/ blokują dostęp sygnałów BUSREQ/, INT/ i NMI/ do procesora.

W czasie gdy sygnał SFSTOP/ jest aktywny, następuje generacja sygnału WAIT/ wymuszającego stan WAIT mikroprocesora. Zdjęcie blokady strobów następuje po wyzerowaniu przerzutnika VBLOK sygnałem SRBLOK/.

Układ testowy służy do generacji sygnału zegarowego i pozwala na podanie napięcia zasilania do mikroprocesora. Ponadto układ wymusza stany nieaktywne na wejściach sterujących (μ RESET/, μ BUSREQ/, μ INT/, μ NMI/, μ WAIT/) oraz umożliwia dołączenie nominalnych obciążeń na wyjścia danych (μ D ϕ - μ D7).

Po włączeniu wtyku kabla emulującego do podstawki układu testowego umożliwia on realizację rozkazów mikroprocesora i obserwację na ekranie oscyloskopu przebiegów wybranych sygnałów emulatora.

Układ forasowania danych steruje dwukierunkowym buforem danych między magistralami SD i D.

Układ blokady strobów wyjściowych umożliwia odcinanie sygnałów sterujących (μ M1/, μ IORQ/, μ RD/, μ WR/) w przypadku komunikacji z pamięcią emulowaną oraz po wejściu procesora w stan wstrzymania (HOLD).

Sygnał wyjściowy przycisku RESET/ wprowadzono na układ zerowania, zgodnie z zaleceniami katalogowymi dla mikroprocesora Z80.

Układ transmisji sygnałów dodatkowych umożliwia zmianę sygnałów pomiarowych (GST ϕ -GST, SUSER, GX) na sygnały o poziomie TTL (SST ϕ -SST3, SUSER, SX/).

W celu wzmocnienia sygnałów wykorzystano układy 8212 (SA ϕ -SA15, SD ϕ -SD7, D ϕ -D7, A7-A14) oraz 8216 (A15, HALT/, BUSAK/).

Układ wyświetlania służy do wyświetlania stanu niektórych sygnałów pozwalających wizualnie ocenić pracę procesora.

Wyświetlane są następujące sygnały:

- BUSREQ (żądanie wstrzymania procesora),
- INT (żądanie przerwania maskowalnego),
- NMI (żądanie przerwania niemaskowalnego),
- WAITR (żądanie przerwania stanu "WAIT"),
- RESET (zerowanie ogólne),
- INTA (potwierdzenie przerwania),
- HOLDA (potwierdzenie wstrzymania stanu HOLD),
- WAITA (potwierdzenie stanu WAIT),
- HALT (stan zatrzymania procesora),
- RUN (stan pracy procesora),
- POWER (stan napięcia zasilania +5 V systemu uruchamianego).

3. PODSUMOWANIE

Emulator mikroprocesora Z80 wspomaga uruchamianie i integrację systemów mikroprocesorowych, opartych na procesorze ZILOG Z80. Mikroprocesor Z80 jest emulowany w pełnym zakresie swoich rzeczywistych charakterystyk łącznie z procesem odwołania pamięci dynamicznych, dla dowolnej konfiguracji systemu prototypowego.

Zestaw funkcji użytkowych obejmuje:

- konfigurowanie pamięci;
- ładowanie, wyprowadzanie i modyfikacje zawartości pamięci;
- ustawianie stanu początkowego mikroprocesora;
- wykonywanie programu krokowo lub w czasie rzeczywistym z deklarowanymi punktami zatrzymania;
- wyprowadzanie pełnej informacji o stanie systemu;
- śladowanie przebiegu programu w czasie rzeczywistym.

LITERATURA

- [1] Rozwój systemu RTDS-8. Sprawozdanie z pracy naukowo-badawczej Instytut Informatyki Czasu Rzeczywistego Politechniki Śląskiej. Gliwice 1983.
- [2] System RTDS-8. Tom III. Uniwersalny emulator mikroprocesorów 8-bitowych. MERA ELZAB. Zabrze 1983.

Recenzent: Prof. dr inż. Stefan Węgrzyn

Doc. dr hab. inż. Andrzej Gościński

Wpłynęło do Redakcji: 30.04.1984 r.

ЗОНД ЭМУЛИРУЮЩИЙ МИКРОПРОЦЕССОР В СИСТЕМЕ РТДС-8

Резюме

Работа даёт описание зонда, эмулирующего микропроцессор в системе РТДС-8. Зонд был спроектирован, сделан и приведён в действие в Коллективе Теории Цифровых Автоматов Силезского Политехнического Института. Микропроцессор Z80 эмулируется в полном размере своих действительных характеристик, вместе с освежением динамических запоминающих устройств для произвольного расположения прототипной системы.

THE Z80 MICROPROCESSOR EMULATING PROBE IN RTDS-8 SYSTEM

Summary

A description of the Z80 microprocessor emulating probe in RTDS-8 system is presented in the paper. The probe has been designed and developed in the Automata Theory Research Team of the Real-Time Computer Science Institute at the Silesian Technical University. Microprocessor Z80 is emulated in a full range of its real characteristics, including dynamic RAM refresh process, for any configuration of prototype system.