POLITECHNIKA ŁÓDZKA
INSTYTUT PODSTAW ELEKTROTECHNIKI

XI KRAJOWA KONFERENCJA TEORIA OBWODÓW I UKŁADY ELEKTRONICZNE

ŁÓDŹ – RYTRO
26–29 października 1988

Bogdan Maciej WILAMOWSKI
Dariusz Jacek CHOLEWIŃSKI
Politechnika Gdańska

SYMULATOR ASYNCHRONICZNYCH UKŁADÓW CYFROWYCH */

Streszczenie - W pracy zaprezentowano efektywny algorytm symulatora asynchronicznych układów cyfrowych wykorzystujący zasadę zachowania ładunku. Przedstawiono dynamiczne modele bramek logicznych oraz opis działania i korzystania z programu napisanego w języku TURBO-PASCAL. Symulator działa w dwóch etapach, w pierwszym następuje kompilacja układu z uwzględnieniem wpływu wzajemnych połączeń pomiędzy bramkami wraz z redukcją liczby parametrów modelu, a następnie symulowany jest układ z zachowaniem stałego kroku czasowego.

SIMULATOR OF ASYNCHRONIZED DIGITAL CIRCUITS

Summary - The effective algorithm of a simulator for digital asynchronized circuits based of charge conservation principle is presented. The dynamic models of digital gates and simulator program written in TURBO-PASCAL is described. The simulator works in two steps: first circuit is compiled taking into account all effect related to mutual interconnection between gates and number of gate model parameters is reduced, next circuit is simulated with constant time step.

1. WSTĘP

Komputerowa analiza stanów przejściowych w układach cyfrowych jest zagadnieniem stosunkowo złożonym, a czasy obliczeń są znacznie dłuższe niż w przypadku analizy stałoprądowej, czy analizy częstotliwościowej. Czyni to analizę dynamiczną dużych układów bardzo trudną. W rozwoju algorytmów można zauważyć tendencję do wykorzystywania specyficznych właściwości układowych dla skrócenia czasów obliczeń [1]. Przykładami takiego podejścia mogą być metody "waveform relaxation" [2] lub "charge conservation"[3][4]. Z drugiej strony daje się zauważyć rozwój algorytmów wykorzystujących jedynie analizę stanów logicznych, gdzie analiza stanów przejściowych zastępowana jest jedynie parametrem opóźnienia. Pozwala to na znaczne skrócenie obliczeń i na analizę bardzo złożonych układów, ale ze względu na pomijanie analiz stanów przejściowych uzyskuje się często wyniki różniące się znacznie od rzeczywistych przebiegów w układzie. Celem niniejszej pracy jest opracowanie symulatora dużych układów cyfrowych, którego wyniki analizy będą możliwie zbliżone do rzeczywistych przebiegów w układzie.

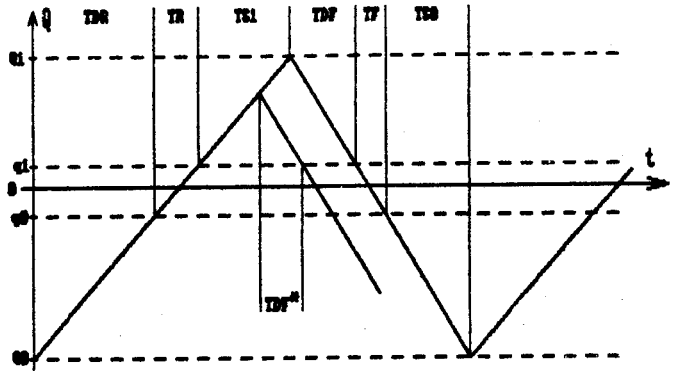
/* Pracę wykonano w ramach CPBP 02.16

2. PODSTAWY ALGORYTMU

Algorytm symulatora bazuje na zasadzie zachowania ładunku. W stosunku do wersji podstawowej [3][4] wprowadzono następujące uproszczenia i ulepszenia:

- Przyjęto, że inercja bramki logicznej wynika z konieczności zmiany umownego wewnętrznego ładunku bramki. W zależności od wielkości prądów ładujących lub rozładowywujących zmiana tego wewnętrznego ładunku Q może następować wolniej lub szybciej.
- Ograniczono maksymalne wewnętrzne ładunki Q0 i Q1, które mogą zgromadzić się w stanach "0" i "1". Pozwoliło to na zapewnienie stabilności algorytmu przy zbyt dużych krokach czasowych.
- Aktualne ładunki wewnętrzne oblicza się na liczbach całkowitych co znacznie przyspiesza obliczenia i zmniejsza obszar zajmowanej pamięci, a jednocześnie nie wpływa w istotny sposób na dokładność obliczeń.
- Przyjęto w obliczeniach stałe - średnie wartości prądów ładujących i rozładowywujących. Uprościło to obliczenia, ale uzyskiwane przebiegi wyjściowe mają charakter liniowo-narastający. Łatwa jest oczywiście taka modyfikacja algorytmu, gdzie prądy ładujące i rozładowywujące będą funkcją ładunku wewnętrznego.

Stan na wyjściu bramki jest pochodną zgromadzonego ładunku wewnętrznego Q. Ilustruje to rysunek 1. Gdzie Q0 i Q1 są maksymalnymi ładunkami jakie mogą zgromadzić się w stanach logicznych "0" i "1". Poziomy q0 i q1 odpowiadają ładunkom przy których zaczyna narastać lub zaczyna opadać napięcie wyjściowe bramki. Czasy TDR i TDF odpowiadają opóźnieniom przy przełączaniu ze stanów "0"=>"1" i "1"=>"0" odpowiednio. Czasy TR i TF są czasami narastania i opadania przebiegów wyjściowych. Czas TS0 i TS1 są czasami ustalania się stanów wewnętrznych bramki w stanach "0" i "1". Przy przełączeniu wejścia w czasie krótszym od TS1 opóźnienie impulsu TDF* będzie odpowiednio mniejsze, tak jak ma to miejsce w rzeczywistym układzie.



Rys. 1. Zmiany wewnętrznego ładunku Q w funkcji czasu.

Przebieg napięć wyjściowych określany jest zależnością:

$$U_{wy} = \begin{cases} U1 & \text{dla } Q > q1 \\ U0 + (Q - q0) * alfa & \text{dla } q0 < Q < q1 \\ U0 & \text{dla } Q < q0 \end{cases}$$

gdzie: U0 i U1 są poziomami napięć odpowiadających stanom "0" i "1"

alfa = (U1 - U0) / (q1 - q0)

3. MODEL BRAMKI

Dynamiczny model elementarnej bramki opisany jest przez 14 parametrów:

TR0 - czas narastania impulsu wyjściowego nieobciążonej bramki
TF0 - czas opadania impulsu wyjściowego nieobciążonej bramki
TDR0 - czas opóźnienia przed narastaniem nieobciążonej bramki
TDF0 - czas opóźnienia przed opadaniem nieobciążonej bramki
CZ0 - pojemność wyjściowa nieobciążonej bramki
DTDR - współczynnik określający zmianę TDR od pojemności obciążenia
DTDF - współczynnik określający zmianę TDF od pojemności obciążenia
DIR - współczynnik określający zmianę TDR od prądu obciążenia
DIF - współczynnik określający zmianę TDF od prądu obciążenia
IWR - średni prąd wejściowy w czasie narastania impulsu wejściowego
IWF - średni prąd wejściowy w czasie opadania impulsu wejściowego
CW - pojemność wejściowa bramki
V1 - napięcie odpowiadające stanowi logicznemu "1"
V0 - napięcie odpowiadające stanowi logicznemu "0"

Na początku obliczeń w czasie kompilacji układu określane są efektywne parametry dynamiczne bramek przy wykorzystaniu podanych niżej zależności:

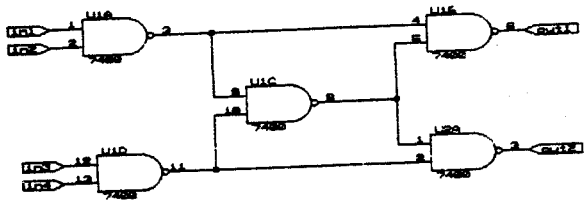
AL = V1 - V0 - amplituda logiczna
IR = CZ0 * AL / TR0 - średnia wydajność prądowa bramki w narastaniu
IF = CZ0 * AL / TF0 - średnia wydajność prądowa bramki w opadaniu
CZ = CZ0 + ΣCW + ΣCS - sumaryczna pojemność wyjściowa bramki
TR = CZ*AL/(IR+ΣIWR) - efektywny czas narastania impulsu wyjściowego
TF = CZ*AL/(IF+ΣIWF) - efektywny czas opadania impulsu wyjściowego
TDR = TDR0 + DTDR*(CZ-CZ0) + DIR*ΣIWR - efektywny czas opóźnienia TDR
TDF = TDF0 + DTDF*(CZ-CZ0) + DIF*ΣIWF - efektywny czas opóźnienia TDF
DR = KROK*(q1-q0)/TR - wzrost ładunku Q w jednym kroku czasowym
DF = KROK*(q1-q0)/TF - malenie ładunku Q w jednym kroku czasowym
Q0 = q0-(q1-q0)*TD0/TR - minimalna wartość wewnętrznego ładunku
Q1 = q1+(q1-q0)*TD1/TF - maksymalna wartość wewnętrznego ładunku

Ostatnie cztery parametry są pamiętane i stosowane w obliczeniach jako charakterystyczne parametry dynamiczne dla każdej bramki.

4. OPIS SYMULATORA

Program symulatora napisany został w języku TURBO PASCAL wersji 4.0. Jako format danych wejściowych przyjęto standard INTERGRAPH. Dane te są automatycznie generowane z rozpowszechnionego w Polsce programu ORCAD. Przykładowo rysunek 2 przedstawia schemat analizowanego układu, a na rysunku 3 podany jest format danych wejściowych automatycznie wygenerowanych programem NETLIST. Dane te składają się z dwóch części. W części pierwszej oznaczonej %PART wszystkim ponumerowanym elementom schmatu przyporządkowane są nazwy (numery) tych elementów w katalogu. Zawarty

ewentualnie w nazwie kod literowy określa parametry dynamiczne układu. W części drugiej oznaczonej %NET opisane są wszystkie węzły układu przez podanie ich nazwy (jeśli była sprecyzowana) lub numeru Nxxxxx wraz z opisem końcówek elementów dołączonych do tych węzłów. Koniec opisu sieci kończy się znakiem $.



Rys. 2. Schemat analizowanego układu

```
%PART
7400            U1
7400            U2
%NET
N00001                  U1-14 U2-14
N00002                  U1-3 U1-9 U1-4
N00003                  U1-7 U2-7
N00004                  U1-8 U1-5 U2-1
N00005                  U1-11 U1-10 U2-2
IN1                     U1-1
IN2                     U1-2
OUT1                    U1-6
IN3                     U1-12
OUT2                    U2-3
IN4                     U1-13
$
```

Rys. 3. Struktura danych o układzie z rysunku 2

Dane wejściowe wspomagane są biblioteką złożonych układów cyfrowych w której zawarta jest informacja o strukturze wewnętrznej tych układów przedstawionej jako sieć połączeń elementarnych bramek logicznych. Format opisu elementów bibliotecznych i sposób ich przygotowania jest identyczny jak w przypadku wprowadzania danych o symulowanym układzie. Tak opracowany element biblioteczny można wprowadzić do biblioteki przy wykorzystaniu odpowiedniej opcji programu. Program dopuszcza również definiowanie modeli z użyciem wcześniej zdefiniowanych struktur. Przykładowo można budować rejestry z użyciem wcześniej określonych przerzutników. Dla użytkownika "widoczny" jest wyłącznie element znajdujący się na schemacie i nie jest możliwa obserwacja wewnętrznych węzłów elementów cyfrowych.

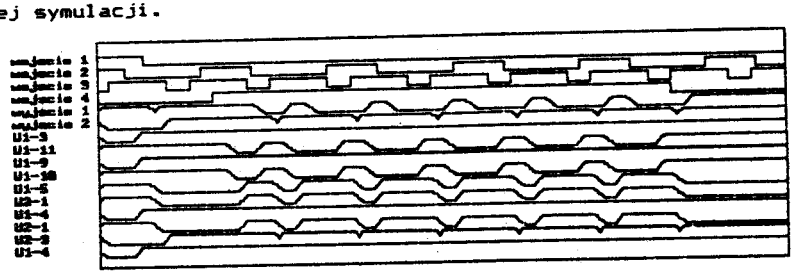
Inny zbiór biblioteczny zawiera zestawienie wszystkich używanych modeli elementarnych bramek po 14 parametrów każda. Parametry te nie zależą od typu realizowanej funkcji logicznej, a jedynie od właściwości dynamicznej bramki.

Analiza układu poprzedzona jest jego kompilacją. W trakcie tego procesu badane są wszystkie wzajemne połączenia bramek, obliczane są łączne pojemności wszystkich węzłów i obciążalność bramek. Na tej podstawie obliczane są rzeczywiste czasy opóźnień i nachylenia zboczy. Każda składowa bramka symulowanego układu od tego momentu posiada tylko cztery własne parametry charakteryzujące jej własności dynamiczne obliczone z wykorzystaniem danych o układzie i 12 parametrów modelu określonego typu bramki.

Dla opisu zewnętrznych sygnałów pobudzających należy wprowadzić następujące informacje:

1. węzeł pobudzany sygnałem
2. wartość początkowa sygnału logicznego 0 lub 1
3. typ sygnału: P - sygnał okresowy, N - sygnał nieokresowy
4. okres dla sygnałów okresowych (w sekundach)
5. zbocze 1 - moment pierwszej zmiany poziomu logicznego
6. zbocze 2 - moment drugiej zmiany poziomu logicznego

Węzeł opisać można poprzez podanie jego nazwy lub numeru, względnie poprzez podanie numeru elementu i numeru jego końcówki. Dane o sygnałach pobudzających i o obserwowanych węzłach wprowadzać można zarówno z klawiatury jak też poprzez przywołanie nazwy pliku, w którym zawarta jest ta informacja. Pliki zawierające informację o zewnętrznych sygnałach pobudzających posiadają rozszerzenie nazwy .INP, a pliki zawierające obserwowane węzły mają rozszerzenie .OUT. Podczas kolejnych analiz układu można przywołać dane zawarte w tych zbiorach, modyfikować je i używać do kolejnej symulacji.

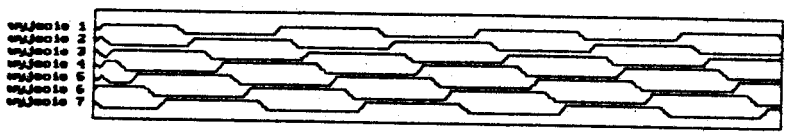


Rys. 4. Wyniki symulacji układu z rysunku 2

Program umożliwia jednoczesną obserwację na ekranie 20 przebiegów napięciowych w układzie. Wybór następuje poprzez podanie numeru lub nazwy węzła, oraz słownego opisu, który będzie umieszczony obok przebiegu na ekranie. Rysunek 4 ilustruje wyniki symulacji układu przedstawionego na rysunku 2.

Program umożliwia uzyskiwanie wyników bardzo zbliżonych do przebiegów w rzeczywistym układzie i pozwala na symulację układów w nietypowych połączeniach. Przykładowo rysunek 5 ilustruje zachowanie się układu

oscylatora pierścieniowego powstałego w wyniku kaskadowego połączenia 7 inwerterów. Przebiegi wyjściowe ilustrują zachowanie się układu w stanie nieustalonym w chwilę po załączeniu napięcia zasilania. Obserwowane przebiegi przejściowe zgodne są z przebiegami obserwowanymi w czasie analizy stanów przejściowych programem PSPICE.



Rys. 5. Wyniki analizy stanów przejściowych w 7 stopniowym pierścieniu oscylatora po załączeniu napięcia zasilania

5. PODSUMOWANIE

W pracy przedstawiono opis algorytmu i pakietu analizy układów cyfrowych. Analiza cyfrowych układów synchronicznych taktowanych zegarem jest stosunkowo prosta. Nie występują tam bowiem pasożytnicze "wyścigi" i "hazardy", dlatego możliwe jest ograniczenie analizy do badania stanów logicznych i sprawdzania maksymalnych opóźnień. W przypadku układów asynchronicznych zagadnienie jest znacznie bardziej złożone.

Cechą specyficzną przedstawionego algorytmu jest to, że mimo przyjęcia wielu uproszczeń algorytm bazuje na fizyce zachodzących zmian w przełączanych układach i dlatego wyniki symulacji zbliżone są do rzeczywistych nawet wtedy jeżeli układ zostanie połączony w sposób niezgodny z jego przeznaczeniem. Symulator przeznaczony jest właśnie do symulacji asynchronicznych układów cyfrowych.

6. LITERATURA

[1] Lighter M.R.: Modeling and simutation of VLSI digital systems. Proc. IEEE, 75, pp. 786-795, 1987

[2] Lelarasmee E., Ruehli A.,Sangiovanni-Vincentelli A.: The waveform relexation method for time-domain analysis of large-scale integrated circuits. IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, CAD-1, pp. 131-145, 1982

[3] Wilamowski B.: Efektywny algorytm analizy stanów przejściowych w układach scalonych. VIII Krajowa KTOiUE, wrzesień 14-17, 1985.

[4] Wilamowski B., Hamilton D., Staszak Z.: CHARCO - IC transient analysis program. IEEE International Circuit and System Conference, San Jose, pp. 548-551, May 1986.