

KATEDRA INFORMATYKI TECHNICZNEJ

Ćwiczenia laboratoryjne z Logiki Układów Cyfrowych

ćwiczenie 204

Temat: Hardware'owa implementacja automatu skończonego pełniącego funkcję automatu niedeterministycznego "NFA with ε - moves" (akceptującego dany język regularny)

1. Wiadomości podstawowe

Automaty skończone niedeterministyczne bez wyjścia odgrywają bardzo istotną rolę w teorii języków formalnych i w problemach akceptacji tych języków przez komputer. Mianowicie automat skończony niedeterministyczny jest modelem matematycznym działania układu fizycznego akceptującego słowa języka formalnego generowanego przez gramatykę bezkontekstową regularną. Automat ten można zrealizować jako program komputerowy lub odpowiedni układ sekwencyjny. Automat skończony niedeterministyczny charakteryzuje się tym, że posiada co najmniej jeden taki stan wewnętrzny, w którym dla danego sygnału wejściowego istnieją co najmniej dwa różne przejścia do następnych stanów. W literaturze automat skończony niedeterministyczny przyjęto oznaczać - zgodnie z terminologią języka angielskiego - symbolem NFA (*Nondeterministic Finite Automaton*).

W ujęciu formalnym automat skończony niedeterministyczny bez wyjścia definiowany jest jako następująca "piątka":

$$\langle Z, Q, \gamma, q_0, F \rangle \quad (1)$$

gdzie:

Z - alfabet wejściowy automatu, interpretowany w zależności od implementacji, również jako zbiór skończony symboli wejściowych lub zbiór skończony sygnałów wejściowych automatu:

$$Z = \{ z_1, z_2, \dots, z_n \}$$

Q - zbiór skończony stanów wewnętrznych automatu:

$$Q = \{ q_0, q_1, q_2, \dots, q_n \}$$

γ - funkcja przejść automatu:

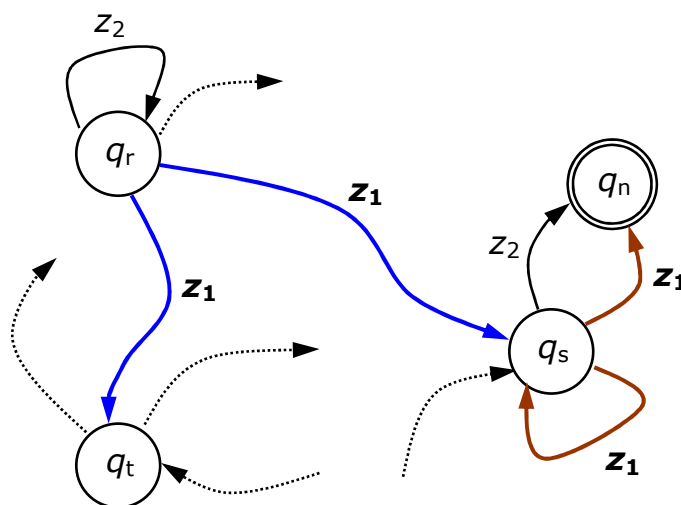
$$\gamma(q(t), z(t)) = \{ q_{i_1}(t+1), q_{i_2}(t+1), \dots, q_{i_m}(t+1) \}$$

q_0 - stan początkowy automatu, $q_0 \in Q$

F - zbiór stanów końcowych automatu, $F \subseteq Q$

Najbardziej komunikatywną formą interpretacji działania automatu skończonego jest graf automatu nazywany również grafem przejść (ang. *transition diagram*). Graf automatu reprezentuje funkcję przejść automatu. Przykładowy fragment grafu automatu niedeterministycznego NFA przedstawiony został na rys. 1. Na rysunku tym pokazano, że

rozpatrywany automat NFA znajdując się w stanie q_r może przejść pod wpływem jednego i tego samego sygnału z_1 do stanu q_s lub q_t . Podobna sytuacja występuje dla przejść pod wpływem z_1 ze stanu q_s .

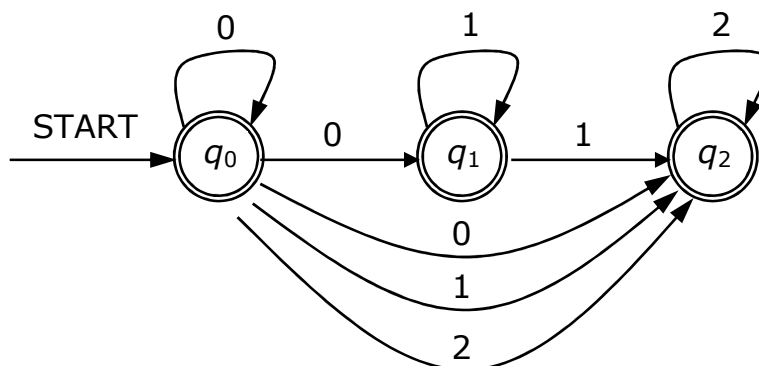


Rys. 1. Fragment grafu automatu niedeterministycznego bez wyjścia

Jak już nadmieniono wyżej, dany automat NFA akceptuje tylko słowa określonego języka formalnego. Oznacza to, że NFA akceptuje tylko określone słowa zbudowane z symboli alfabetu wejściowego automatu. Symbolami alfabetu wejściowego mogą być litery z alfabetu określonego języka naturalnego (na przykład angielskiego lub greckiego) lub słowa zawarte w słowniku danego języka naturalnego, języka programowania lub też innego języka, którego alfabet jest zbiorem określonych ciągów symboli.

Dane słowo podane na wejście NFA jest akceptowane, jeżeli w grafie przejść tego automatu znajduje się co najmniej jedna taka ścieżka, opisana symbolami danego słowa, która prowadzi od wierzchołka reprezentującego stan początkowy automatu do wierzchołka reprezentującego stan końcowy automatu.

Dla przykładu rozpatrzmy automat niedeterministyczny o grafie przejść przedstawionym na rys. 2.



Rys. 2. Graf przejść przykładowego NFA

W automacie tym wszystkie trzy stany q_0 , q_1 i q_2 są stanami końcowymi, przy czym stan q_0 pełni również funkcję stanu początkowego. Na wejście automatu podawane są symbole ze zbioru $Z = \{ 0, 1, 2 \}$. Automat ten akceptuje następujące ciągi symboli wejściowych:

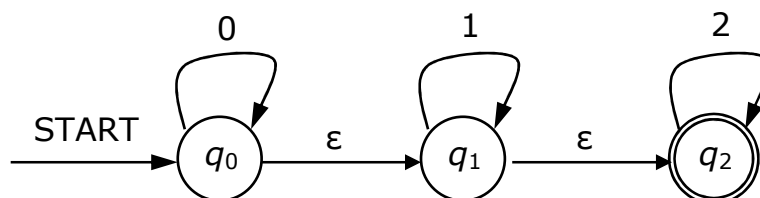
00 ... 0, 00 ... 011 ... 1, 00 ... 011 ... 122 ... 2, 11 ... 1, 11 ... 122 ... 2, 22 ... 2

natomiast nie akceptuje ciągów typu:

11 ... 100 ... 0, 22 ... 200 ... 0, 22 ... 211 ... 1

Automat NFA o grafie przedstawionym na rys. 2 nie akceptuje na przykład słowa 020, ponieważ pod wpływem sygnału "2" automat ten może przejść tylko do stanu q_2 . Będąc w stanie q_2 nie jest zdolny do odebrania sygnału wejściowego "0" (z wierzchołka q_2 grafu wychodzi tylko krawędź z symbolem wejściowym "2").

Przedstawiony wyżej przykładowy model automatu NFA ułatwia zrozumienie jego działania, natomiast jego praktyczna implementacja jako programu lub układu sekwencyjnego jest utrudniona. W rozważaniach teoretycznych, jak również w implementacji komputerowej automatu niedeterministycznego jako układu akceptującego słowa danego języka formalnego posługujemy się najczęściej pojęciem automatu niedeterministycznego z tak zwanymi przejściami "pustymi" ϵ . W literaturze naukowej automat ten występuje pod nazwą: "NFA with ϵ -moves". Automat ten można łatwo określić na podstawie reguł gramatyki bezkontekstowej, generującej słowa danego języka formalnego. Każdy automat NFA z przejściami "pustymi" ϵ można przekształcić na automat NFA bez przejść "pustych" ϵ . Graf przejść przykładowego automatu NFA z przejściami "pustymi" ϵ przedstawiony został na rys. 3. Automat posiadający taki graf przejść jest ekwiwalentny automatowi NFA o grafie przejść z rys. 2. W interpretacji automatu skończonego jako "czarnej skrzynki" obydwa automaty są identyczne, to znaczy akceptują jeden i ten sam język.



Rys. 3. Graf przejść przykładowego automatu niedeterministycznego z przejściami "pustymi" ϵ

Zgodnie z rys. 3 stan q_0 jest stanem początkowym automatu, natomiast stan q_2 jest stanem końcowym. Przejście "puste" ϵ , zgodnie z rozpatrywanym grafem przejść, reprezentuje sobą "pusty" symbol wejściowy ϵ .

W rozpatrywanym przykładzie "pusty" symbol wejściowy ϵ może być interpretowany jako samoczynne przejście z jednego stanu w inny stan. To samoczynne przejście istnieje tylko dla pewnych stanów, na przykład w grafie przedstawionym na rys. 3 istnieje ono tylko dla q_0 i q_1 . Załóżmy, że na wejście rozpatrywanego automatu podany jest symbol wejściowy "1". W stanie początkowym q_0 , zgodnie z rys. 3, nie ma przejścia do stanu q_1 jednakże symbol "pusty" ϵ , występujący w każdej chwili dyskretnej na wejściu automatu, powoduje przejście do stanu q_1 . Będąc w stanie q_1 automat akceptuje zarówno symbol wejściowy "1", pod wpływem którego pozostanie on nadal w stanie q_1 , jak również symbol "pusty" ϵ (interpretowany jako "1") powodujący przejście do stanu q_2 .

W danej chwili czasu automat może znajdować się tylko w jednym ze swoich stanów.

Stąd też w rozpatrywanym przypadku może on pozostać w stanie q_1 (w wyniku działania sygnału "1") lub przejść do stanu końcowego q_2 w wyniku działania sygnału "pustego" E . Będąc w stanie q_1 automat przejdzie pod wpływem sygnału E do stanu końcowego q_2 , co oznacza, że symbol wejściowy "1" został zaakceptowany przez automat.

Przyjmuje się następującą, nieformalną definicję akceptacji danego słowa wejściowego przez automat skończony typu NFA.

Definicja:

Dane słowo wejściowe automatu NFA jest zaakceptowane przez ten automat tylko wtedy, gdy automat znajdzie się w jednym ze swoich stanów końcowych i wszystkie litery (symbole) słowa wejściowego zostały odczytane.

Jeżeli automat NFA znajduje się w stanie końcowym a na wejściu automatu znajduje się jeszcze taka litera słowa wejściowego, dla której w stanie końcowym nie ma żadnego przejścia, wtedy słowo to nie jest akceptowane przez automat. Dla przykładu rozpatrzmy słowo wejściowe 0020. Zgodnie z grafem automatu NFA przedstawionym na rys. 3 pod wpływem symboli 00 automat może pozostawać w stanie q_0 . Kolejnym symbolem wejściowym po symbolach 00 jest "2". Symbol wejściowy "2" w stanie q_0 nie może być odczytany, jednak pod wpływem sygnału E automat przejdzie z q_0 do q_1 a następnie z q_1 do q_2 . W stanie q_2 symbol wejściowy "2" jest akceptowany i automat pozostaje w stanie końcowym q_2 . Ostatnim symbolem rozpatrywanego słowa wejściowego jest "0", które w stanie q_2 nie może być zaakceptowane. Stąd też, zgodnie z podaną wyżej definicją akceptacji słów wejściowych przez NFA, słowo 0020 nie może być zaakceptowane przez automat o grafie z rys. 3.

2. Przebieg ćwiczenia

Automat niedeterministyczny bez wyjść z "pustymi" przejściami ϵ trudno zrealizować jako urządzenie techniczne, ponieważ brak konkretnych metod syntezy tego typu automatu. Należy jednak zwrócić uwagę na to, że automat ten zbudowany jako układ sekwencyjny powinien wykonywać funkcję akceptacji słów określonego języka regularnego. Zgodnie z podaną wcześniej definicją, automat "*NFA with ϵ -moves*" akceptuje dane słowo, jeżeli w grafie przejść tego automatu istnieje co najmniej jedna taka ścieżka, wśród innych ścieżek, która dla danego słowa wejściowego prowadzi od wierzchołka reprezentującego stan początkowy do wierzchołka reprezentującego stan końcowy. Eliminując więc z grafu wszystkie takie ścieżki, które dla danego słowa akceptowanego przez automat prowadzą do nieakceptacji otrzymamy graf pewnego automatu skończonego pełniącego funkcję danego "*NFA with ϵ -moves*" (w rozpatrywanym przypadku pełniącego funkcję automatu NFA o grafie z rys. 3). Schemat ideowy takiego automatu przedstawiony został na rys. 4 a na rysunkach 5 i 6 przedstawiono dwa możliwe warianty wykonawcze tego automatu jako układu sekwencyjnego, akceptującego słowa ze zbioru zadanego wyrażeniem regularnym $0^* 1^* 2^*$.

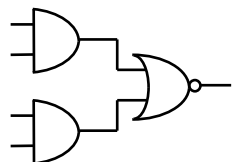
Zadaniem studentów jest zmontowanie układu sekwencyjnego zgodnie ze schematem z rys. 5 lub 6 i zbadanie zachowania automatu po podaniu na jego wejście określonych słów ze zbioru słów akceptowanych i ze zbioru słów nieakceptowanych. W sprawozdaniu z ćwiczenia należy opisać działanie tego automatu na podstawie schematu z rys. 4, zarówno w przypadku słów akceptowanych jak i nieakceptowanych przez ten automat.

2.1. Uwagi na temat schematów

Pierwszy schemat (rys. 4) służy tylko do zaprezentowania w możliwie prosty sposób idei układu logicznego (nie jest uwzględniony zewnętrzny sygnał RESET). Schemat z rys. 5 zbudowany jest w oparciu o bramki AND i OR, dlatego nie może być bezpośrednio połączony przy wykorzystaniu modułów dostępnych w zestawie UNILOG.

Na schemacie końcowym (rys. 6) wykorzystane są dostępne bramki NOR, dodana została również blokada sygnału START jeśli automat jest już w stanie końcowym Q_2 . Sprowadzenie automatu do stanu początkowego (sygnałem START) wymaga użycia sygnału RESET. Dla przerzutników RS aktywny jest stan niski sygnałów SET i RESET. Można użyć dostępnych w laboratorium przerzutników synchronicznych D lub JK, wykorzystując tylko ich wejścia asynchroniczne. Przerzutniki RS można również zbudować z pojedynczych bramek (ze względu na aktywny stan niski dla wejść, musi to być wariant z bramkami NAND).

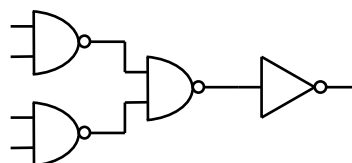
Na schemacie z rys. 6 wykorzystane są dostępne w laboratorium gotowe bloki składające się z bramek NOR i bramek AND na jej obu wejściach:



W razie braku takich bloków (ograniczona liczba modułów) można je zastąpić układem zbudowanym z bramek NAND, korzystając z przekształcenia:

$$\overline{ab + cd} = \overline{ab} \overline{cd} = \overline{\overline{\overline{ab} \overline{cd}}}$$

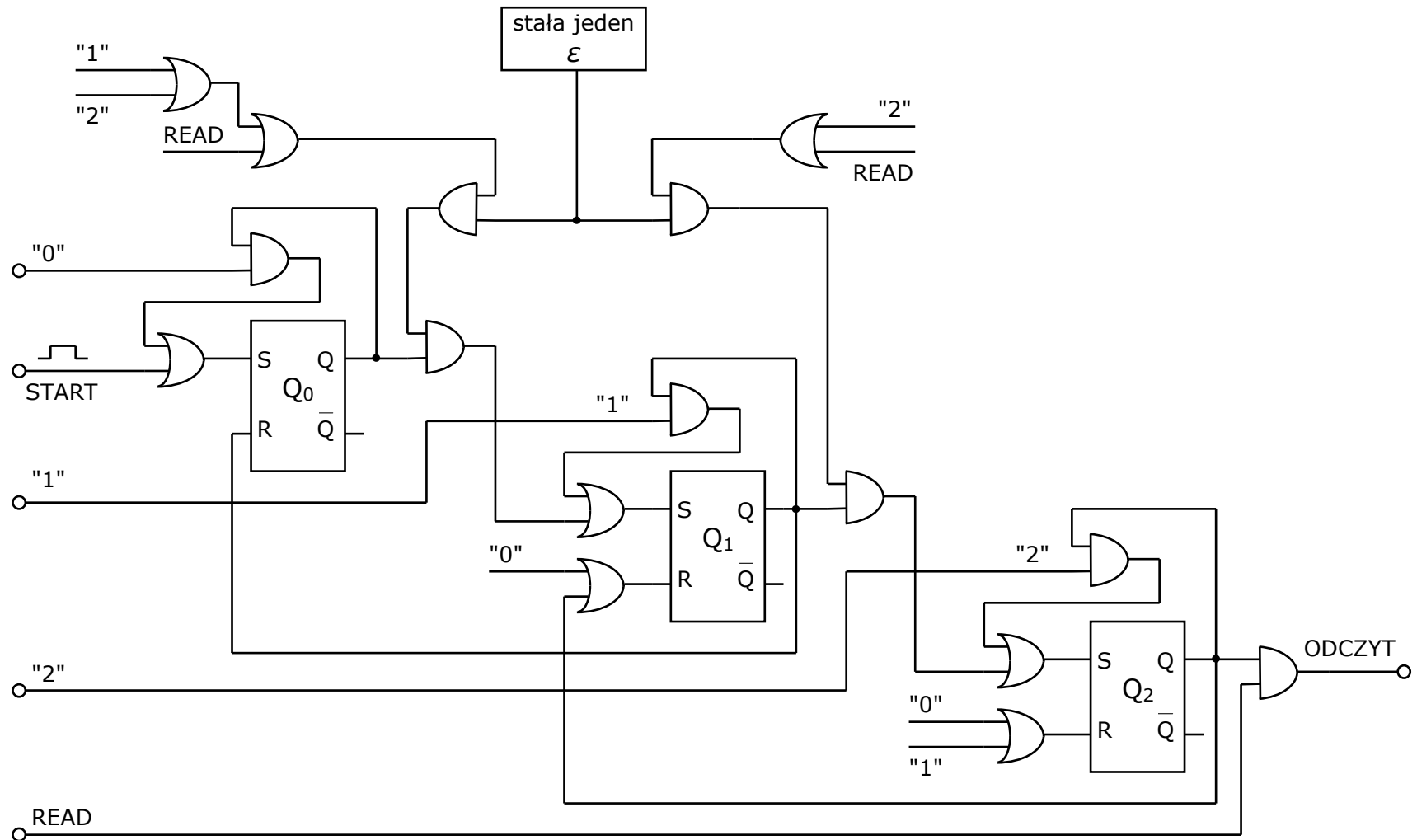
W rezultacie otrzymujemy następujący układ zastępczy:



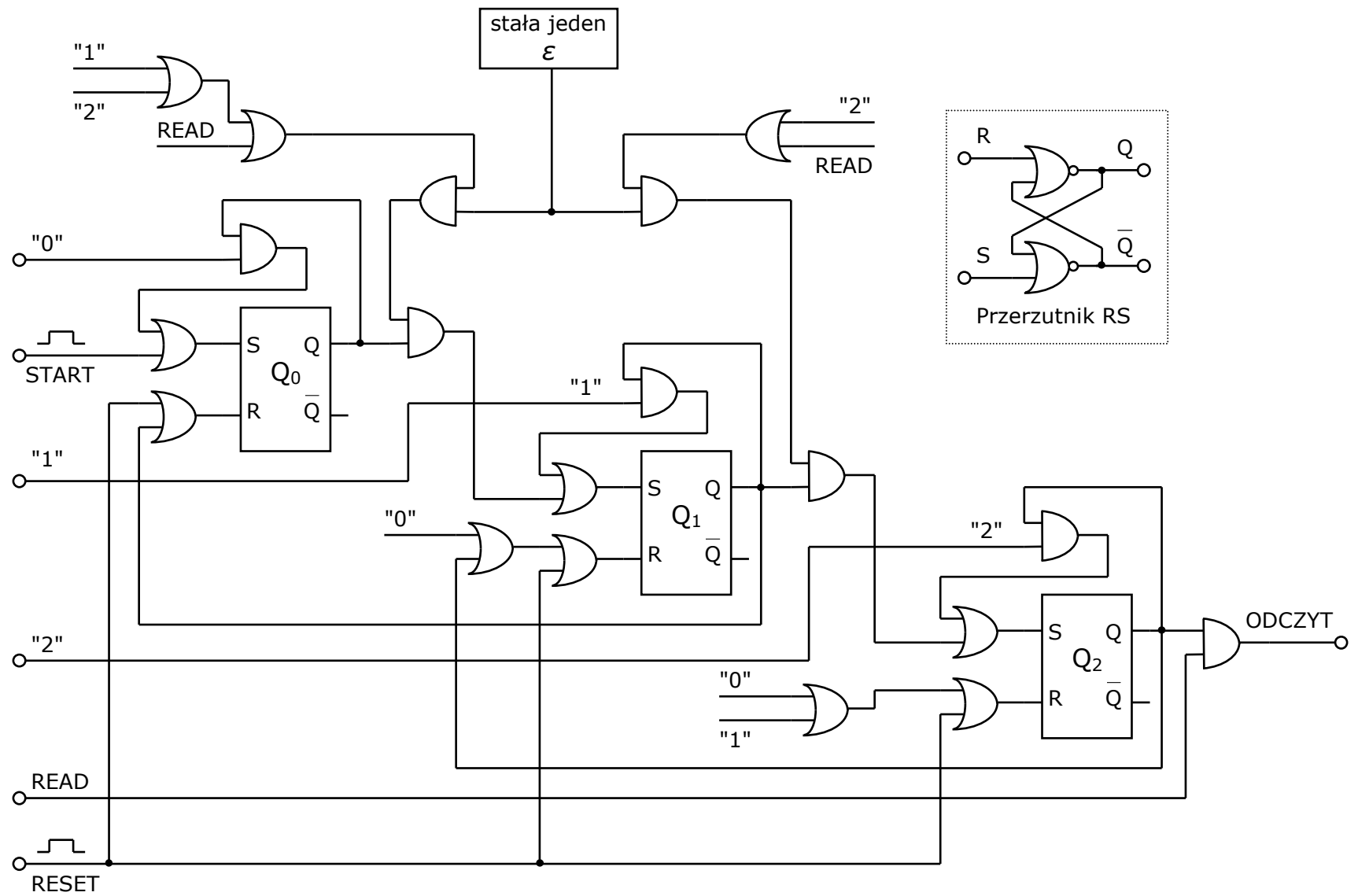
2.2. Uwagi na temat testowania

Testowany w ćwiczeniu automat NFA jest układem sekwencyjnym asynchronicznym, dlatego zmiany stanów zachodzą bezpośrednio po zmianie wejść. Utrudnia to diagnozowanie i wykrywanie ewentualnych błędów w układzie. Niekiedy problemy mogą być również spowodowane sprzecznym sterowaniem wejść SET i RESET przerzutników RS (oba jednocześnie aktywne). Stan przerzutnika Q i jego negacja mogą mieć wtedy taki sam poziom logiczny. W związku z tym warto podzielić układ na bloki, związane z realizowanymi funkcjami, a następnie kolejno je łączyć i testować.

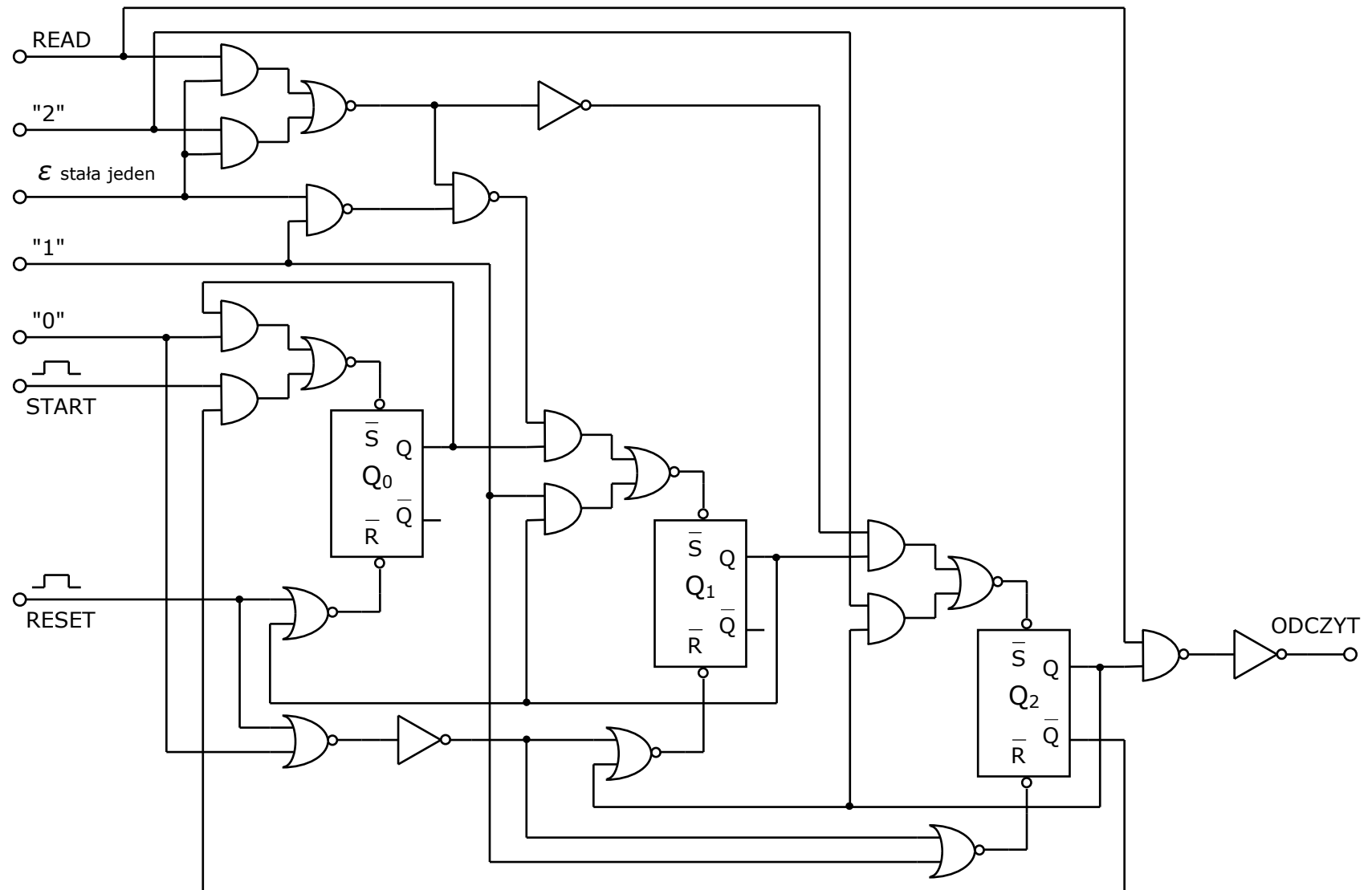
Proponowane jest połączenie w pierwszej kolejności części układu związanej z kasowaniem przerzutników (czyli sterującej wejściami RESET przerzutników RS), pozostawiając na tym etapie wiszące wejścia SET. Wykonując testy można ręcznie ustawić dowolny przerzutnik, podając aktywny sygnał na wejście SET. Należy przy tym pamiętać o wynikających wprost z zasady działania automatu sprzężeniach zwrotnych. Na przykład ustawienie przerzutnika Q_1 (w prawidłowo działającym układzie) spowoduje automatyczne skasowanie jego poprzednika czyli Q_0 .



Rys. 4. Schemat ideowy automatu niedeterministycznego z przejściami "pustymi" ϵ (uproszczony)



Rys. 5. Schemat ideowy automatu niedeterministycznego z przejściami "pustymi" ϵ (AND / OR)



Rys. 6. Schemat ideowy automatu niedeterministycznego z przejściami "pustymi" ϵ