

Architektury systemów komputerowych

Lista 5

$x_5 = 7$ (minimum na bdb)

1. (a) Zbuduj licznik synchroniczny liczący od 0 do 5 (kolejne stany: 000, 001, 010, 011, 100, 101, 000, 001...). Zadanie rozwiąż w dwóch wersjach – w jednej użyj przerzutników J-K, w drugiej – przerzutników D.
(b) Zbuduj układ synchroniczny, przyjmujący kolejno stany 0, 3, 1, 5, 0, 3, Użyj przerzutników S-R.
2. Zbuduj licznik liczący w sposób następujący: 0, 1, 2, 3, 4, 5, 4, 3, 2, 1, 0, 1, 2, 3, 4, 5, ... Użyj przerzutników J-K.
3. Zaprojektuj licznik synchroniczny $L_{n+1} = (L_n + 3) \bmod 8$ (kolejne stany powinny kodować 0, 3, 6, 1, 4, 7, 2, 5, 0, 3...), używając przerzutników MUX-NOT (patrz lista 4, zadanie 6).
4. Zaprojektuj jako automat Moore'a układ synchroniczny o dwóch wejściach i dwóch wyjściach, który zachowuje się następująco:
 - (a) jeśli na wejściu są dwa zera, układ powinien liczyć „w górę”, tzn. w kolejnych taktach zegara na wyjściu powinny się pojawiać: 0, 1, 2, 3, 0, 1, 2, 3, 0, 1, 2, 3, ...
 - (b) jeśli na wejściu są dwie jedynki, układ powinien liczyć „w dół”, tzn. w kolejnych taktach zegara na wyjściu powinny się pojawiać: 0, 3, 2, 1, 0, 3, 2, 1, 0, 3, 2, 1, ...
 - (c) jeśli na wejściu jest jedynka i zero, układ powinien liczyć w następujący sposób: 0, 2, 1, 3, 0, 2, 1, 3, ...
 - (d) kombinacja wejściowa zero, jedynka nie jest wykorzystywana.
5. Zaprojektuj automat Mealy'ego, który w kolejnych taktach zegara czyta na swoim jednobitowym wejściu cyfry binarne. Na wyjściu ma się pojawić 1 jeśli przeczytany do tej pory ciąg cyfr kończy się na 010 lub 1001, a 0 w przeciwnym wypadku.
6. Zaprojektuj układ wykrywający na wejściu sekwencję 1010 (układ ma ustawić jedynkę na wyjściu dokładnie wtedy, gdy przeczytany do tej pory ciąg znaków kończy się zadaną sekwencją) w dwóch wersjach: raz jako automat Mealy'ego, raz jako automat Moore'a. W obydwu wersjach postaraj się użyć jak najmniejszej liczby stanów (ale nie musisz formalnie udowadniać, że użyte liczby stanów są minimalne).
7. Rozważamy automat do sprzedaży gazet o następującej specyfikacji. Gazeta kosztuje 35 centów. Automat przyjmuje monety 5-centowe, 10-centowe i 25-centowe. Ma dwa osobne pojemniki na monety (każdy podzielony na trzy części na odpowiednie rodzaje monet): w większym są monety przyjęte za już wydane gazety, w mniejszym – monety aktualnie wrzucane. Jeśli w mniejszym pojemniku znajdzie się dokładnie 35 centów, to automat wydaje gazetę i przenosi monety z mniejszego pojemnika do większego. Jeśli wartość wrzuconych monet przekroczy 35 centów, to automat albo wydaje gazetę, zwraca resztę (z monet w pojemniku większym) i przenosi monety z pojemnika mniejszego do większego, albo oddaje wszystkie monety z pojemnika mniejszego. (jeśli reszty nie umie wydać). Dodatkowo, w każdej chwili, jeśli zostanie wciśnięty specjalny przycisk 'Cancel', automat oddaje wszystkie monety z pojemnika mniejszego. Stan automatu będzie kodował m.in. ile monet jest w większym pojemniku: np. ustawienie przerzutnika N_1 na jeden, oznacza, że mamy co najmniej jedną monetę 5-centową, ustawienie N_2 oznacza, że mamy co najmniej dwie monety jednocentowe, itd. Ile takich przerzutników wystarczy do naszych celów?

Oczywiście możesz założyć, że np. po wrzuceniu monety 5-centowej pewnien sygnał wejściowy dla automatu ustawiany jest na 1 oraz, że ustawienie jakiegoś sygnału wyjściowego na 1 spowoduje pewną akcję (np. wydanie z większego pojemnika monety 5-centowej albo przeniesienie wszystkich monet z pojemnika mniejszego do większego).

Dospecyfikuj pozostałe szczegóły i zaprojektuj automat jako automat Mealy'ego. Nie musisz go implementować na poziomie przerzutników i bramek logicznych. Wystarczy specyfikacja wejść, wyjść i stanów oraz opis funkcji wyjścia i zmiany stanów. Stanów będzie pewnie dość sporo - być może będziesz potrafił zaproponować jakiś sposób opisu automatu, który nie będzie wymagał rysowania całego grafu przejść (w przykładzie z wykładu narysowaliśmy pełny graf, ale miał on tylko kilka stanów).

- 8* (2pkt.) Pokaż, że działanie każdego automatu Mealy'ego (przy założeniu, że jego wyjścia mogą się zmieniać tylko raz na cykl zegarowy, czyli, że są one zapamiętane w przerzutnikach) da się symulować za pomocą automatu Moore'a.
9. (a) Jakie liczby są reprezentowane w systemie uzupełnień do dwóch (na ośmiu bitach) jako: 00110011, 10101010, 10110001, 11111111, 00000000?
 (b) Jak w systemie uzupełnień do dwóch (na ośmiu bitach) są reprezentowane następujące liczby 0, -7, 12, 127, -127?
10. (a) Jak zmieni się wartość liczby w reprezentacji uzupełnień do 2 jeśli z prawej strony dopiszemy do niej zero?
 (b) Jak w reprezentacji uzupełnień do 2 konwertuje się liczby 16-bitowe do 32-bitowych?
11. Rozważmy wyliczanie wyrażenia $x = x_1 + x_2 + \dots + x_n$ w 8-bitowej reprezentacji uzupełnień do 2 (sumujemy „od lewej do prawej”). Załóżmy, że poprawna wartość x mieści się w przedziale $[-128, 127]$, ale wyniki pośrednie mogą z niego wyskakiwać (np. $x = 100 + 100 - 100$). Czy ostateczna wartość x wyliczona w arytmetyce uzupełnień do 2 będzie poprawna?

Emanuel Kieroński