Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Комп’ютерна логіка

Лабораторна робота №5

«Синтез цифрових автоматів на тригерах»

Виконала:

студентка групи ІВ-71

Молчанова В. С.

Залікова книжка № ІВ-7110

Перевірив Верба О. А.

Київ

2017 р.

# Тема

Синтез цифрових автоматів на тригерах

# Мета

Вивчити методи структурного синтезу керуючих автоматів із жорсткою логікою, одержати навички в їх налагодженні та експериментальному дослідженні.

# Теоретичні відомості

Цифровий автомат, що має два і більше станів, є послідовносною схемою. Ознакою такої логічної схеми є наявність петель. Під петлею розуміється шлях з виходу логічного елемента на його вхід безпосередньо або через інші елементи.

Автомат може виконувати функцію управління для пристроїв обробки інформації.

Послідовність мікрооперацій, виконання яких призводить до виконання операції, називають мікроалгоритмом даної операції.

Один з підходів теорії цифрових автоматів до побудови структурних автоматів полягає в представленні будь-якого автомата у вигляді композиції елементарних автоматів Мура, що мають назву тригерів (рис. 5.1).

Автомат містить комбінаційну схему (КС) і пам'ять (П), що складається з тригерів *Тi*. Входами КС є виходи *Q1*,...,*Qm* тригерів і вхідні сигнали (логічні умови) *х*1,...,*хk*, що формуються в операційному пристрої. КС виробляє керуючі сигнали *y*1,...,*yp* для операційного пристрою і функції збудження тригерів *q*1,...,*qm*, що визначають перехід автомата з одного стану в інший. Кожній з безлічі станів {*а*1,...,*аm*} відповідає визначений набір стану, так і від вхідних сигналів, називають автоматом Мілі. Його функціонування визначається виразами

,

.

Можна виділити чотири основні функціональні типи тригерів: RS-тригери, JK-тригери, D-тригери і T-тригери. Умовні графічні позначення синхронних тригерів і системи підграфів переходів, що пояснюють спосіб зміни стану тригерів, зображені на рис. 5.2. Тригери мають тільки два стани: нульовий стан – при =0 і =1, та одиничний стан – при =1 і =0. Перехід тригерів з одного стану в інший визначається інформаційними сигналами, а момент переходу – перепадом синхросигналу С (в даному випадку перепад з 1 в 0). Асинхронні входи тригерів R і S дозволяють встановлювати початковий стан тригерів.

Вихідними даними для синтезу автомата є схема операційного пристрою, що дає можливість зробити перелік керуючих сигналів для кожного вузла, і змістовний мікроалгоритм операції, що включає опис мікрооперацій і логічних умов. Побудова схеми і розробка мікроалгоритму є взаємозалежними процесами.



|  |  |  |  |
| --- | --- | --- | --- |
| JK-тригер | T-тригер | RS-тригер | D-тригер |
|  |  |  |  |
| *а*) | | | |
|  |  |  |  |
| *б*) | | | |
| *Рис. 5.2. Синхронні тригери: а – умовні графічні означення; б – система підграфів переходів* | | | |

|  |
| --- |
|  |
|  |

Синтез автомата включає наступні етапи:

1) складання списку керуючих сигналів, що забезпечують виконання кожної мікрооперації;

2) визначення тривалості кожного керуючого сигналу (в числі тактів) і періоду тактуючих сигналів автомата;

3) одержання закодованого мікроалгоритму;

4) відмітка станів автомата;

5) складання графа автомата;

6) кодування станів автомата;

7) складання структурної таблиці автомата;

8) одержання МДНФ функцій збудження

тригерів і керуючих сигналів;

9) представлення функцій збудження тригерів

і керуючих сигналів в операторной формі;

10) побудова схеми керуючого автомата.

Розглянемо приклад синтезу автомата Мілі.

Розглянемо приклад синтезу автомата. Будемо вважати, що після виконання пунктів 1, 2 і 3 одержано закодований структурний мікроалгоритм (рис. 5.3), де  (=1,5,7,9) – управляючи сигнали, що подаються на операційний пристрій, а – логічні умови (зовнішні для автомата вхідні сигнали). Для переходу до графа автомата треба зробити розмітку станів. Розмітка станів для циклічного автомата Мілі здійснюється наступним чином.

Символом *аi* відзначається вхід вершини (логічної або операторної), наступної за початковою, а також вхід кінцевої вершини; входи всіх вершин, наступних за операторними, повинні бути відзначені різними символами 

|  |  |  |
| --- | --- | --- |
| *Таблиця 5.1* | | |
| *Таблиця кодування станів автомата Мілі* | | |
| Стан | Код стану | |
| *Q*1 | *Q*2 |
| *a*1 | 0 | 0 |
| *a*2 | 0 | 1 |
| *a*3 | 1 | 1 |
| *a*4 | 1 | 0 |

Відмітимо, що спосіб кодування впливає на правильність формування керуючих сигналів і складність автомата. Можливість формування сигналів, не передбачених графом автомата, при неоптимальному кодуванні станів обумовлена появою “гонок”, що пов'язано з розкидом часу переключення окремих тригерів автомата. Наприклад, при переході автомата зі стану 10 у стан 01 під час переключення тригерів можлива поява станів 00 або 11 (в залежності від того, який із тригерів раніш спрацьовує). Ці проміжні стани при використанні тригерів із внутрішньою затримкою не впливають на правильність переключення автомата, однак можуть привести до появи короткочасних помилкових керуючих сигналів. Для усунення цього недоліку можна використовувати протигоночне сусіднє кодування. При сусіднім кодуванні перехід автомата з одного в будь-який інший припустимий для даного автомата стан здійснюється переключенням тільки одного тригера, внаслідок чого ”гонки” не виникають. В автоматах, що не допускають сусіднього кодування, необхідно вводити додаткові стани.

Крім станів, визначених таким чином, може виникнути необхідність введення додаткових станів. Наприклад, якщо мікрооперація виконується по перепаду керуючого сигналу і мікроалгоритм має петлю, яка охоплює операторную вершину з таким сигналом, або кілька вершин, що слідують підряд, то для забезпечення перепадів керуючого сигналу при кожному черговому виконанні мікрооперації необхідно вводити додаткові стани. В цих станах автомат не повинний виробляти керуючі сигнали. Додаткові стани можуть знадобитися для забезпечення протигоночного кодування (див. далі).

Мікроалгоритм на рис. 5.3 відзначений чотирма різними станами (*а*1, *а*2, *а*3, *а*4). Граф розглянутого автомата Мілі зображений на рис. 5.4.

Число вершин графа дорівнює числу станів *аi* автомата. Кожному переходові автомата з одного стану в інший відповідає дуга графа. Дузі приписується набір логічних умов, при якому здійснюється перехід автомата з одного стану в інший, а також набір керуючих сигналів, що відповідають даному переходові.

|  |
| --- |
|  |
| *Рис. 5.4. Граф автомата Мілі* |

Кількість тригерів, необхідних для організації пам'яті автомата (див. рис. 5.1) визначається із співвідношення *m*>]log2*М*[, де *М* – число станів автомата. Кожному станові *аi* повинна відповідати одна визначена комбінація значень *Q*1,...,*Qm*. Для розглянутого приклада вибираємо коди станів відповідно до табл. 5.1. Для організації пам'яті будемо використовувати JK-тригери.

Структурна таблиця автомата складається по його графу. Кожен рядок (табл. 5.2) відповідає визначеному переходові автомата з одного стану в інший. В ній записують вихідний стан, стан переходу, коди цих станів, значення логічних умов, що забезпечують перехід, необхідні значення керуючих сигналів і функцій збудження тригерів. Значення функції збудження визначаються відповідно до таблиці переходів тригера відповідного типу. В кожному рядку для *i*-го тригера розглядаються переходи *QiS*→*QiS*+1 з попереднього (ПС) в наступний (НС) стан. Довільні значення(0 або 1) сигналів позначаються в таблиці знаком \*. Збільшення невизна чених станів, як правило, спрощує схему автомата.

На підставі структурної таблиці автомата визначаємо МДНФ функцій збудження і керуючих сигналів. Аргументами функцій *Ji, Ki* та *yi* є значення *Q1S, Q2S, x1* і *x2*.

Використовуючи діаграми Вейча (рис. 5.5), одержуємо:

;;;

;;

;

.

Функціональна схема автомата зображена на рис. 5.6, де УПС – установлення початкового стану, ТС – тактуючі сигнали.

Відмінність синтезу автомата Мура полягає в наступному. Розмітка станів автомата в цьому випадку здійснюється іншим чином. Символом *а*1 відзначаються початкова і кінцева вершини; всі операторні вершини відзначаються різними символами *аj*.

|  |
| --- |
|  |
| *Рис. 5.6. Функціональна схема автомата* |

В даному випадку, як і при оцінці станів автомата Мілі, може виникнути необхідність введення додаткових станів.

На графі автомата Мура дугам приписують набори логічних умов, що забезпечують відповідний перехід автомата. Керуючі сигнали записують у вершинах графа, тому що вони не залежать від логічних умов.

На рис. 5.7 і 5.8 показані закодований відмічений мікроалгоритм і граф автомата Мура.

Кодування станів автомата Мура можна виконувати так само, як і для автомата Мілі. Однак при відповідному кодуванні керуючі сигнали можна знімати і безпосередньо з виходів тригерів автомата Мура (комбінаційна схема для формування функцій yj не потрібна). При цьому число тригерів повинне бути не менше числа керуючих сигналів, які не повторюють один одного і не є константами.

|  |  |
| --- | --- |
|  |  |
| *Рис. 5.7. Графічна схема закодованого мікроалгоритму*  *автомата Мура* | *Рис. 5.8. Граф автомата Мура* |

В цьому випадку відповідний рівень керуючих сигналів після спрацьовання тригерів установлюється швидше, ніж в автомата Мілі. Крім того при будь-якому переході виключається можливість короткочасного формування помилкових керуючих сигналів. Приклад такого кодування станів автомата, що відповідає графові на рис. 5.8, приведений в табл. 5.3.

В даному випадку автомат має три тригери, причому *y*1=*Q*1, *y*7=*Q*1, *y*5=*Q*2, *y*9=*Q*3.

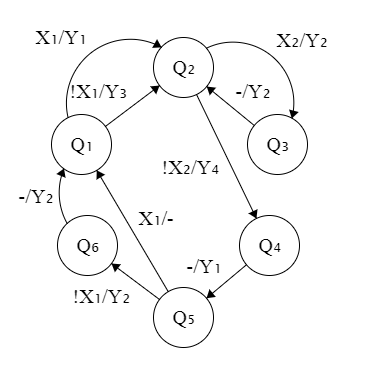
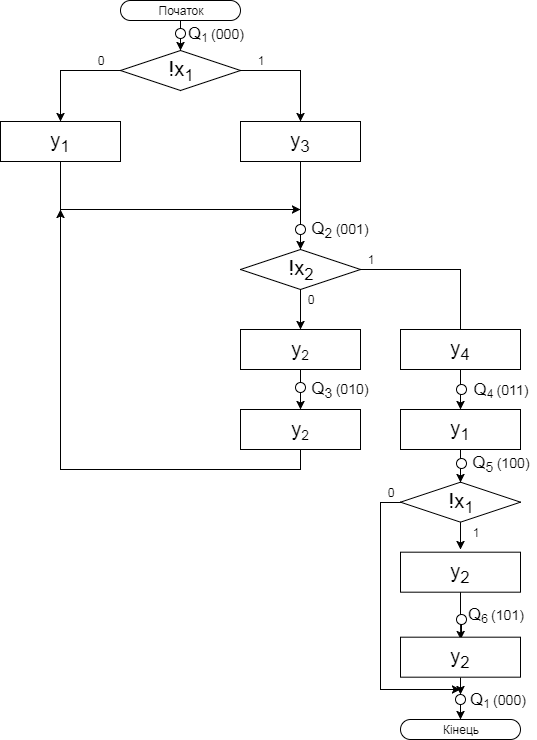
Співпадаючі сигнали *y*1 і *y*7 знімаються з одного тригера.

# Хід роботи

Номер моєї залікової книжки у двійковій системі дорівнює 111000110. Тому мій варіант завдання виглядатиме так:

|  |  |
| --- | --- |
| Порядок з'єднання фрагментів | 3, 4, 2 |
| Послідовність логічних умов |  |
| Послідовність вихідних сигналів |  |
| Сигнал, тривалістю 2*t* |  |
| Тип тригерів | JK |
| Тип автомата | Мілі |
| Логічні елементи | 3АБО-НЕ, 3І |

Побудування графічної схеми мікроалгоритму та графа:



Табличний вигляд схеми мікроалгоритму:

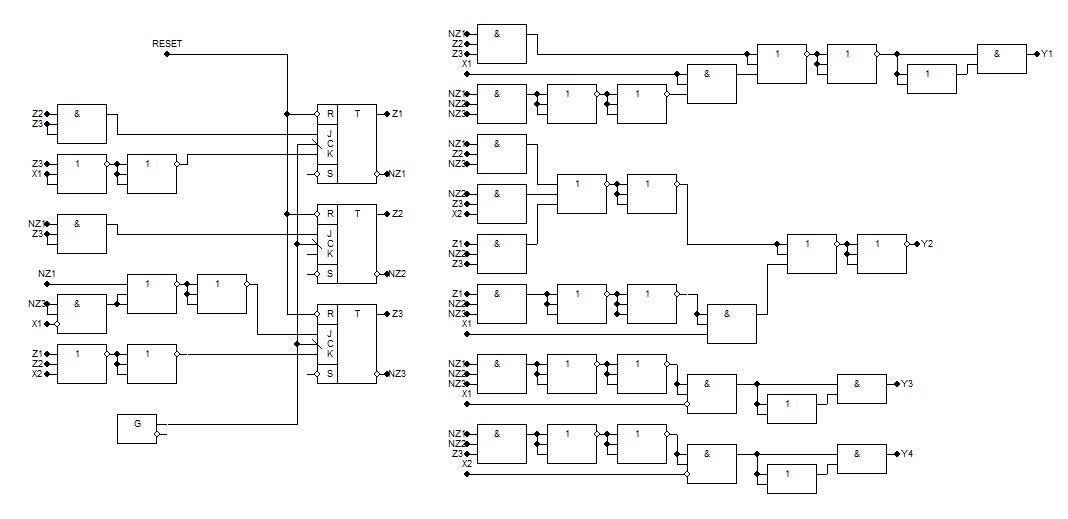
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Qt | Qt+1 | x1 | x2 | y1 | y2 | y3 | y4 | JK1 | JK2 | JK3 |
| 000 | 001 | 1 | - | 1 | 0 | 0 | 0 | 0- | 0- | 1- |
| 000 | 001 | 0 | - | 0 | 0 | 1 | 0 | 0- | 0- | 1- |
| 001 | 010 | - | 1 | 0 | 1 | 0 | 0 | 0- | 1- | -1 |
| 001 | 011 | - | 0 | 0 | 0 | 0 | 1 | 0- | 1- | -0 |
| 010 | 001 | - | - | 0 | 1 | 0 | 0 | 0- | -1 | 1- |
| 011 | 100 | - | - | 1 | 0 | 0 | 0 | 1- | -1 | -1 |
| 100 | 101 | 0 | - | 0 | 1 | 0 | 0 | -0 | 0- | 1- |
| 100 | 000 | 1 | - | 0 | 0 | 0 | 0 | -1 | 0- | 0- |
| 101 | 000 | - | - | 0 | 1 | 0 | 0 | -1 | 0- | -1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *J1* | | *Z2* | |  |  |  |  | *J2* | | *Z2* | |  |  |  |  | *J3* | | *Z2* | |  |  |  |
| *Z1* | *Z3* | - | - | - | - |  |  | *Z1* | *Z3* | - | - | 0 | 0 |  |  | *Z1* | *Z3* | - | - | - | - |  |
| - | - | - | - | *x1* |  | - | - | 0 | 0 | *x1* |  | - | - | - | - | *x1* |
|  | - | - | - | - |  |  | - | - | 0 | 0 |  |  | - | - | 0 | 0 |
|  | - | - | - | - |  |  |  | - | - | 0 | 0 |  |  |  | - | - | 1 | 1 |  |
|  | *Z3* | 1 | 1 | 0 | 0 |  |  |  | *Z3* | - | - | 1 | 1 |  |  |  | *Z3* | - | - | - | - |  |
|  | 1 | 1 | 0 | 0 | *x1* |  |  | - | - | 1 | 1 | *x1* |  |  | - | - | - | - | *x1* |
|  |  | 0 | 0 | 0 | 0 |  |  |  | - | - | 0 | 0 |  |  |  | 1 | 1 | 1 | 1 |
|  |  | 0 | 0 | 0 | 0 |  |  |  |  | - | - | 0 | 0 |  |  |  |  | 1 | 1 | 1 | 1 |  |
|  |  |  | *x2* | |  |  |  |  |  |  | *x2* | |  |  |  |  |  |  | *x2* | |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| *K1* | | *Z2* | |  |  |  |  | *K2* | | *Z2* | |  |  |  |  | *K3* | | *Z2* | |  |  |  |
| *Z1* | *Z3* | - | - | 1 | 1 |  |  | *Z1* | *Z3* | - | - | - | - |  |  | *Z1* | *Z3* | - | - | 1 | 1 |  |
| - | - | 1 | 1 | *x1* |  | - | - | - | - | *x1* |  | - | - | 1 | 1 | *x1* |
|  | - | - | 1 | 1 |  |  | - | - | - | - |  |  | - | - | - | - |
|  | - | - | 0 | 0 |  |  |  | - | - | - | - |  |  |  | - | - | - | - |  |
|  | *Z3* | - | - | - | - |  |  |  | *Z3* | 1 | 1 | - | - |  |  |  | *Z3* | 1 | 1 | 1 | 0 |  |
|  | - | - | - | - | *x1* |  |  | 1 | 1 | - | - | *x1* |  |  | 1 | 1 | 1 | 0 | *x1* |
|  |  | - | - | - | - |  |  |  | 1 | 1 | - | - |  |  |  | - | - | - | - |
|  |  | - | - | - | - |  |  |  |  | 1 | 1 | - | - |  |  |  |  | - | - | - | - |  |
|  |  |  | *x2* | |  |  |  |  |  |  | *x2* | |  |  |  |  |  |  | *x2* | |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *Y1* | | *Z2* | |  |  |  |  | *Y2* | | *Z2* | |  |  |  |  | *Y3* | | *Z2* | |  |  |  |  | *Y4* | | *Z2* | |  |  |  |
| *Z1* | *Z3* | 0 | 0 | 0 | 0 |  |  | *Z1* | *Z3* | 0 | 0 | 1 | 1 |  |  | *Z1* | *Z3* | 0 | 0 | 0 | 0 |  |  | *Z1* | *Z3* | 0 | 0 | 0 | 0 |  |
| 0 | 0 | 0 | 0 | *x1* |  | 0 | 0 | 1 | 1 | *x1* |  | 0 | 0 | 0 | 0 | *x1* |  | 0 | 0 | 0 | 0 | *x1* |
|  | 0 | 0 | 0 | 0 |  |  | 0 | 0 | 0 | 0 |  |  | 0 | 0 | 0 | 0 |  |  | 0 | 0 | 0 | 0 |
|  | 0 | 0 | 0 | 0 |  |  |  | 0 | 0 | 1 | 1 |  |  |  | 0 | 0 | 0 | 0 |  |  |  | 0 | 0 | 0 | 0 |  |
|  | *Z3* | 1 | 1 | 0 | 0 |  |  |  | *Z3* | 0 | 0 | 1 | 0 |  |  |  | *Z3* | 0 | 0 | 0 | 0 |  |  |  | *Z3* | 0 | 0 | 0 | 1 |  |
|  | 1 | 1 | 0 | 0 | *x1* |  |  | 0 | 0 | 1 | 0 | *x1* |  |  | 0 | 0 | 0 | 0 | *x1* |  |  | 0 | 0 | 0 | 1 | *x1* |
|  |  | 0 | 0 | 1 | 1 |  |  |  | 1 | 1 | 0 | 0 |  |  |  | 0 | 0 | 0 | 0 |  |  |  | 0 | 0 | 0 | 0 |
|  |  | 0 | 0 | 0 | 0 |  |  |  |  | 1 | 1 | 0 | 0 |  |  |  |  | 0 | 0 | 1 | 1 |  |  |  |  | 0 | 0 | 0 | 0 |  |
|  |  |  | *x2* | |  |  |  |  |  |  | *x2* | |  |  |  |  |  |  | *x2* | |  |  |  |  |  |  | *x2* | |  |  |

# 

Найдовший шлях у схемі: 6 елементів АБО-НЕ, 3 елемента І, тригер та фільтр з двох елементів, тобто тривалість такту дорівнює 6 + 3 + 6 + 2 = 17



# Висновок

Я вивчила методи структурного синтезу керуючих автоматів із жорсткою логікою, отримала навички в їх налагодженні та експериментальному дослідженні.