УДК 004.27;621.382

Ю.О. Гунченко, д.т.н.,
Л.Я. Мартинович
Одеський національний університет ім. І.Мечникова, Одеса

АЛГОРИТМ ПОБУДОВИ УНАРНИХ ТРІЙКОВИХ ФУНКЦІЙ

Трійкова логіка має ряд переваг порівняно з двійковою. Це такі, як природне представлення чисел зі знаком, коли не потрібно використовувати зворотній чи додатковий код або спеціальний знаковий біт; команда розгалуження по знаку в трійковій машині займає в два рази менше часу, ніж в двійковій; в тривходовому трійковому суматорі перенесення в наступний розряд виникає в 8 ситуаціях з 27, а в двійковому суматорі – в 4 з 8.

В двійковій логіці є всього 4 операції для одного аргумента, тобто унарних. У той же час в трійковій логіці їх 27. Наприклад, трійкова інверсія – унарна операція, яка міняє місцями два з трьох логічних станів: NOT– - інверсія, що міняє місцями 0 і +1; NOT - інверсія, що міняє місцями -1 і +1; NOT+ - міняє місцями -1 і 0.

Використовуючи універсальний пристрій, побудований на основі багатопорогового елемента багатозначної логіки [1], можна отримати усі 27 трійкових унарних операцій.

Таб. 1. Значення вихідних сигналів струмових перемикачів

Сума вхідних струмів	Вихідні сигнали струмових перемикачів			
terlev	СП1, СП2		СП3, СП4	
	$+R_1$	$+L_1$	$–R_1$	$–L_1$
–	0	+	–	0
0	+	0	–	0
+	+	0	0	–

Поєднуючи виходи СП1 – СП4 та константу, сформовану джерелом струму, можна отримати будь-яку трійкову унарну функцію. Реалізація трійкових унарних функцій здіснюється за допомогою наступного алгоритму:

1. Обираємо набір (№ в таблиці істинності) функції, який будемо реалізовувати першим: а) це не нульове значення («+» або «–»), якого в функції менше; б) кщо ненульових значень однакова кількість, то обираємо те, яке відповідає ненульовому вхідному аргументу («+» або «–»); в) кщо є і те, і інше, то обираємо будь-яке.

2. Обираємо RL-функцію, яка реалізує значення з п.1, незалежно

від отриманих значень при інших аргументах.

3. Обираємо набір (№ в таблиці істинності) функції, який будемо реалізовувати другим: а) це не нульове значення («+» або «–»); б) якщо два ненульових значення, то обираємо те, яке відповідає ненульовому вхідному аргументу («+» або «–»); в) інакше, обираємо те, що змінилось при виконанні п.2.

4. Обираємо RL-функцію (або її відсутність), яка реалізує значення з п.3 (можливо, їх буде дві, для компенсації дій функції з п.2). Якщо функція з п.4 змінює значення п.1: а) на нульове, то необхідно додати RL-функцію з п.2; б) інакше, необхідно додати RL-функцію з протилежним значенням у вибраному в п.1 номері функції з п.4.

5. Обираємо RL-функцію (або її відсутність), яка реалізує (або відповідним чином змінює) значення, що залишилось. Якщо функція з п.5 змінює значення п.1: а) на нульове, то необхідно додати RL-функцію з п.2; б)інакше, необхідно додати RL-функцію з протилежним значенням у вибраному в п.1 номері функції з п.5. Якщо функція з п.5 змінює значення п.3: а) на нульове, то необхідно додати RL-функцію з п.4; б) інакше, необхідно додати RL-функцію з протилежним значенням у вибраному в п.3 номері функції з п.5.

6. Оптимізація. Якщо в отриманому наборі функцій є такі, які в сумі дають константу («+1» чи «–1») на всіх наборах вхідних аргументів, то їх можна замінити на константу «+» або «–».

В результаті роботи методу, отримуємо таблицю, що демонструє, які виходи струмових перемикачів необхідно об’єднати та чи потрібно задіяти джерело струму, для того щоб синтезувати обрану функцію.

Висновок. В роботі запропоновано алгоритм побудови унарних трійкових функцій на основі універсального пристрою багатопорогового елемента багатозначної логіки. Даний алгоритм можна використовувати в якості основи для створення алгоритма побудови двомісних трійкових функцій

ВИКОРИСТАНІ ДЖЕРЕЛА

1. Пат. UA 118735 Україна, МПК (2017.01) H03K19/00. Багатопорговий елемент багатозначної логіки / Гунченко Ю.О. Заявка 23.03.2017, опубл. 28.08.2017.