

В.В. Лапко, канд. техн. наук, доц.,
О.В. Самощенко, канд. техн. наук, доц.,
Г.Е. Маргієв, аспірант.

Донецький національний технічний університет, м. Покровськ
aleksandr.samoshchenko@gmail.com

Синтез програмованих схем додавання та віднімання цілих чисел в системі доповняльних кодів

Розроблено математичний опис та визначені властивості програмованого суматора-віднімача беззнакових цілих чисел як операційного приладу схем додавання та віднімання. Запропонований метод формалізованого визначення початкових даних програмованих суматорів-віднімачів невід'ємних цілих чисел для вираження суми та різниці в системі доповняльних кодів. Розроблений доцільний спосіб форматування доповняльного коду операндів, відмінний тим, що два старших розряди вихідного поліному операційного суматора-віднімача дозволяють визначити від'ємне та додатне переповнення або факт відсутності переповнення розрядної сітки результату операції. Показано, що запропоновані схеми фіксації наявності та полярності переповнення інваріантні відносно режиму роботи операційного суматора-віднімача. Коректність розробленого математичного опису та алгоритмів обґрунтовані VHDL-моделюванням рівнянь запропонованих схем додавання та віднімання цілих чисел в системі доповняльних кодів.

Ключові слова: доповняльний код, програмований суматор-віднімач, переповнення, комп'ютерне моделювання.

DOI: 10.31474/1996-1588-2017-2-25-76-83

Вступ

З метою мінімізації апаратних витрат в сучасних процесорах комп'ютерних систем застосовуються багатофункціональні програмовані арифметично-логічні пристрої (АЛП), набір арифметичних операцій в яких ґрунтується на виконанні операцій додавання та віднімання беззнакових цілих чисел [1-3]. В силу цього, при використанні АЛП в схемах додавання та віднімання беззнакових цілих чисел в системі доповняльних кодів важливого теоретичного та практичного значення набувають питання сумісного та погодженого використання режимів додавання та віднімання беззнакових цілих чисел АЛП та доцільного кодування початкових даних при обчисленні суми та різниці цілих чисел в доповняльному коді. Застосування операційного двійкового суматора беззнакових цілих чисел для додавання та віднімання цілих чисел в доповняльному коді досить докладно висвітлено в літературі [1, 4-6]. Але сумісне та погоджене використання арифметичних операцій додавання та віднімання беззнакових цілих чисел в системі доповняльних кодів потребує додаткових досліджень. Актуальним завданням є розробка формального математичного опису АЛП як операційного пристрою схем додавання та віднімання беззнакових цілих чисел в доповняльному коді. Через обмеженість розрядної сітки конкретних АЛП надважливим завданням також стає розробка простих та швидких способів аналізу коректності доповняльного коду

результату. Оптимізація цієї задачі зводиться до мінімізації кількості входів в схемі контролю та використанню в схемі контролю виключно штатних виводів АЛП. Крім того, важливим критерієм якості схем контролю є інваріантність алгоритмів фіксації переповнення розрядної сітки відносно режиму роботи АЛП. В роботі запропонована методика синтезу оптимального операційного пристрою обробки цілих чисел в системі доповняльних кодів на основі програмованого суматора-віднімача беззнакових цілих чисел.

Математичний опис програмованого суматора-віднімача беззнакових цілих чисел як операційного пристрою

В якості базового операційного елементу програмованого суматора-віднімача беззнакових цілих чисел використовуємо канонічний двійковий суматор невід'ємних цілих чисел (рис.1), операційні властивості якого описуються співвідношеннями:

$$S(n,1) = ((n+1,1))_{mV} = \begin{cases} (n+1,1) & \text{при } (n+1,1) < V : E=0; \\ (n+1,1) - V & \text{при } (n+1,1) \geq V : E=1, \end{cases} \quad (1)$$

де $(n+1,1) = (a(n,1) + b(n,1) + e)$ – двійковий код повної суми початкових даних суматора:

$$(n+1,1) = (n) \dots (2) \quad (1),$$

де $(n+1) = E$ – вихідний перенос суматора з ваговим коефіцієнтом $V = 2^n$;

$(n,1) = S(n,1) = S(n) S(n-1) \dots S(2) S(1)$ – двійковий код суми початкових даних на основних виходах суматора;

$a_s(n,1) = a_s(n) a_s(n-1) \dots a_s(2) a_s(1)$, $b_s(n,1) = b_s(n) b_s(n-1) \dots b_s(2) b_s(1)$ – двійкові коди доданків на основних входах суматора; e_s – вхідний перенос до молодших розрядів доданків суматора; $((n+1,1))_{mV}$ – опис процедури обчислення залишку двійкового коду $(n+1,1)$ по модулю V [5,7].

В режимі додавання беззнакових цілих чисел (при коді операції $SB = 0$) операційні властивості програмованого суматора-віднімача задаємо співвідношеннями, аналогічними суматору [1]:

$$S_c(n,1) = (\Sigma_c(n+1,1))_{mV} = \begin{cases} \Sigma_c(n+1,1) & \text{при } \Sigma_c(n+1,1) < V: E_c = 0; \\ \Sigma_c(n+1,1) - V & \text{при } \Sigma_c(n+1,1) \geq V: E_c = 1, \end{cases} \quad (2)$$

де $\Sigma_c(n+1,1) = (a_c(n,1) + b_c(n,1) + e_c)$ – двійковий код повної суми в режимі додавання; E_c , e_c – відповідно вихідний та вхідний перенос в режимі додавання; $a_c(n,1)$, $b_c(n,1)$ – двійкові коди доданків; $S_c(n,1)$ – код суми на основних виходах канонічного суматора (рис.1) програмованого суматора-віднімача в режимі додавання.

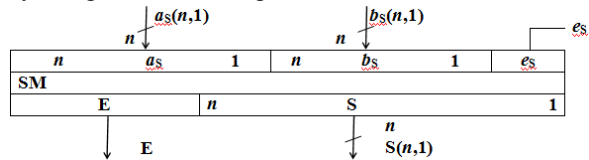


Рисунок 1 – Функціональна схема канонічного двійкового суматора беззнакових цілих чисел

Для реалізації режиму додавання програмованого суматора-віднімача (при коді операції $SB = 0$) в канонічному суматорі, згідно (1) та (2), необхідно виконати операції:

$$a_s(n,1) = a_c(n,1); b_s(n,1) = b_c(n,1); e_s = e_c. \quad (3)$$

Згідно (1) та (3) на виводах канонічного суматора отримаємо:

$$E = E_c, S(n,1) = S_c(n,1). \quad (4)$$

В режимі віднімання беззнакових чисел (при $SB = 1$) операційні властивості програмованого суматора-віднімача, згідно правил арифметики цілих чисел, задаємо співвідношеннями:

$$R(n,1) = (r(n,1))_{mV} = \begin{cases} r(n,1) & \text{при } r(n,1) \geq 0: M=0; \\ V+r(n,1) & \text{при } r(n,1) < 0: M=1, \end{cases} \quad (5)$$

де $r(n,1) = (a_m(n,1) - b_m(n,1) - e_m)$ – різниця початкових даних віднімача; $a_m(n,1)$, $b_m(n,1)$ – відповідно зменшуване та від'ємник на входах віднімача; e_m – вхідна позика віднімача; M – вихідна позика віднімача; $R(n,1)$ – двійковий код різниці на основних виходах канонічного суматора в режимі віднімання початкових даних в програмованому суматорі-віднімачі.

Для реалізації віднімача в базисі канонічного суматора подаємо (5) рівносильним виразом:

$$R(n,1) = (r(n,1))_{mV} = (a_m(n,1) - b_m(n,1) - e_m)_{mV} = \quad (6)$$

$$= (a_m(n,1) + V - 1 - b_m(n,1) - e_m)_{mV} = (a_m(n,1) + \overline{b_m(n,1)} + \overline{e_m})_{mV},$$

де $V - 1 - b_m(n,1) = 1 \cdot 2^n + 1 \cdot 2^{n-1} + \dots + 1 \cdot 2 + 1 - b_m(n) \cdot 2^n - b_m(n-1) \cdot 2^{n-1} - \dots - b_m(2) \cdot 2 - b_m(1) = (1 - b_m(n)) \cdot 2^n + (1 - b_m(n-1)) \cdot 2^{n-1} + \dots + (1 - b_m(2)) \cdot 2 + (1 - b_m(1)) = \overline{b_m(n)} \cdot 2^{n-1} + \overline{b_m(n-1)} \cdot 2^{n-2} + \dots + \overline{b_m(2)} \cdot 2 + \overline{b_m(1)} = \overline{b_m(n)} \overline{b_m(n-1)} \dots \overline{b_m(2)} \overline{b_m(1)} = \overline{b_m(n,1)}$ – порозрядна інверсія зменшуваного $b_m(n,1)$;

$$\overline{e_m} = (1 - e_m) = \begin{cases} 1 & \text{при } e_m = 0, \\ 0 & \text{при } e_m = 1. \end{cases}$$

Для реалізації режиму віднімання беззнакових чисел, згідно (6), в канонічному суматорі необхідно виконати операції:

$$a_s(n,1) = a_m(n,1); b_s(n,1) = \overline{b_m(n,1)}; e_s = \overline{e_m}. \quad (7)$$

Тоді на основних виходах канонічного суматора, згідно (1) та (7), отримаємо:

$$S(n,1) = R(n,1) = (a_m(n,1) + \overline{b_m(n,1)} + \overline{e_m})_{mV} = \quad (8)$$

$$= (a_m(n,1) + V - 1 - b_m(n,1) - e_m)_{mV}.$$

Повна сума на виході суматора (7) визначається, згідно (1) та (5), виразом

$$\Sigma(n+1,1) = a_m(n,1) + \overline{b_m(n,1)} + \overline{e_m} = a_m(n,1) + V - 1 - b_m(n,1) - e_m = V + (a_m(n,1) - b_m(n,1) - e_m) = V + r(n,1). \quad (9)$$

Згідно (1) та (9), отримаємо:

$$E = \begin{cases} 1 & \text{при } r(n,1) \geq 0: M=0; \\ 0 & \text{при } r(n,1) < 0: M=1, \end{cases} \quad (10)$$

де $r(n,1) = \Sigma(n+1,1) - V$.

Таким чином, згідно (10) та (5), на виході E суматора (7) формується стан:

$$E = \overline{m}. \quad (11)$$

Для реалізації режимів додавання (при $SB = 0$) та віднімання (при $SB = 1$) беззнакових чисел, згідно (3) та (7), в канонічному суматорі необхідно виконати операції:

$$a_s(n,1) = a(n,1); b_s(n,1) = b(n,1) \oplus SB; e_s = e \oplus SB, \quad (12)$$

де $a(n,1) = a_c(n,1) / a_m(n,1) = \begin{cases} a_c(n,1) & \text{при } SB = 0, \\ a_m(n,1) & \text{при } SB = 1; \end{cases}$

$b(n,1) = b_c(n,1) / \overline{b_m(n,1)} = \begin{cases} b_c(n,1) & \text{при } SB = 0, \\ \overline{b_m(n,1)} & \text{при } SB = 1; \end{cases}$

$e = e_c / \overline{e_m} = \begin{cases} e_c & \text{при } SB = 0, \\ \overline{e_m} & \text{при } SB = 1. \end{cases}$

Тоді в програмованому суматорі-віднімачі (рис.2) на виходах канонічного суматора, згідно (4) та (11), отримаємо:

$$E = E_c / \overline{m} = \begin{cases} E_c & \text{при } SB = 0, \\ \overline{m} & \text{при } SB = 1. \end{cases} S(n,1) = S_c(n,1) / R(n,1) = \begin{cases} S_c(n,1) & \text{при } SB = 0, \\ R(n,1) & \text{при } SB = 1. \end{cases} \quad (13)$$

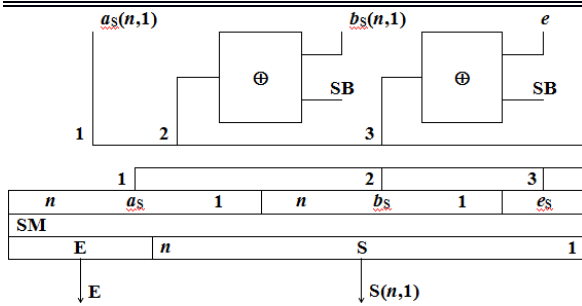


Рисунок 2 – Функціональна схема програмованого суматора- віднімача беззнакових цілих чисел

$$S(n,1) = \begin{cases} (a(n,1) + b(n,1) + e)_{mV} & \text{при } SB = 0, \\ (a(n,1) - b(n,1) - e)_{mV} & \text{при } SB = 1; \end{cases}$$

$$E(\text{при } SB=0) = \begin{cases} 1 & \text{при } (a(n,1) + b(n,1) + e) \geq V, \\ 0 & \text{при } (a(n,1) + b(n,1) + e) < V; \end{cases}$$

$$E(\text{при } SB=1) = \bar{m} = \begin{cases} 1 & \text{при } (a(n,1) - b(n,1) - e) \geq 0, \\ 0 & \text{при } (a(n,1) - b(n,1) - e) < 0. \end{cases}$$

Синтез схем додавання доповняльних кодів цілих чисел на основі програмованого суматора- віднімача беззнакових чисел

Операція додавання цілих чисел зводиться до розв'язання задачі $C = A + B$, (14)

де C – алгебраїчна сума цілих чисел A та B .
При розв'язанні задачі (14) двійковий код суми чисел C будемо формувати на основних виходах суматора- віднімача $S(n,1)$, згідно (2), у вигляді залишку по модулю V суматора:

$$S(n,1) = (C)_{mV} = (A + B)_{mV}. \quad (15)$$

Для визначення форми подання операндів задачі (14) на входах суматора- віднімача трансформуємо (15) до виду

$$S(n,1) = (C)_{mV} = (A + B)_{mV} = V + A + V + B)_{mV} = ((V + A)_{mV} + (V + B)_{mV})_{mV} = (A_{дк}(n,1) + B_{дк}(n,1))_{mV}, \quad (16)$$

де $A_{дк}(n,1) = (V + A)_{mV} \in [0; V-1]$,
 $B_{дк}(n,1) = (V + B)_{mV} \in [0; V-1]$ – відповідно доповняльні коди операндів задачі (14) [6].

Остаточо, згідно (15) та (16), отримаємо

$$S(n,1) = (C)_{mV} = (V + C)_{mV} = C_{дк}(n,1) = (A_{дк}(n,1) + B_{дк}(n,1))_{mV}. \quad (17)$$

Для вирішення задачі (14) в режимі додавання на входах суматора- віднімача, згідно (13), необхідно формувати сукупність даних:

$$a(n,1) = A_{дк}(n,1); b(n,1) = B_{дк}(n,1); e = 0; SB = 0. \quad (18)$$

Для завдання властивостей доповняльних кодів суматора (18), наприклад функції $C_{дк}(C)$, покладемо:

$$C_{дк}(C \geq 0) \in [0; V/2-1]; C_{дк}(C < 0) \in [V/2; V-1], \quad (19)$$

де $[0; V/2-1] \cup [V/2; V-1]$ – множина двійкових наборів n -розрядної двійкової функції $C_{дк}(n,1) \in [0; V-1]$.

Тоді при $V = 2^n$, згідно (19), отримаємо:

$$0 \leq C_{дк}(n,1) \leq [V/2-1] \text{ при } C \geq 0, 2^{n-1} \leq C_{дк}(n,1) \leq [V-1] \text{ при } C < 0. \quad (20)$$

Двійковий код $C_{дк}(n,1)$ довжиною n -біт описується системою нерівностей:

$$0 \leq C_{дк}(n) \cdot 2^{n-1} + C_{дк}(n-1,1) < 2^{n-1} \text{ при } C \geq 0, 2^{n-1} \leq C_{дк}(n) \cdot 2^{n-1} + C_{дк}(n-1,1) \leq 2^n \text{ при } C < 0. \quad (21)$$

Звідси є:

$$C_{дк}(n) = [NC] = \begin{cases} 0 & \text{при } C \geq 0; \\ 1 & \text{при } C < 0, \end{cases}$$

де $C_{дк}(n) = [NC]$ – знак аргументу C функції $C_{дк}(C)$ в поліномі $C_{дк}(n,1)$;
 $C_{дк}(n) = [NC] C_{дк}(n-1,1)$.

Подібні доповняльні коди операндів $A_{дк}(n,1)$ та $B_{дк}(n,1)$ мають аналогічний формат:

$$A_{дк}(n) = [NA] A_{дк}(n-1,1); B_{дк}(n) = [NB] B_{дк}(n-1,1), \quad (22)$$

де $[NA]$, $[NB]$ – відповідно знаковий розряд операндів A та B :

$$[NA] = \begin{cases} 0 & \text{при } A \geq 0, \\ 1 & \text{при } A < 0; \end{cases} [NB] = \begin{cases} 0 & \text{при } B \geq 0, \\ 1 & \text{при } B < 0. \end{cases}$$

Для обчислення області визначення доповняльних кодів суматора (18) подаємо, наприклад, зображення суми $C_{дк}(n,1)$, згідно (17) та правил алгебри обчислення залишків, системою нерівностей:

$$C_{дк}(n,1) = (V + C)_{mV} = \begin{cases} C & \text{при } C \geq 0, \\ V + C & \text{при } C < 0. \end{cases} \quad (23)$$

Звідси випливає, що область визначення функції $C_{дк}(C)$, згідно (20) та (23), визначається співвідношеннями

$$C = \begin{cases} C_{дк}(C0)[0; V/2 - 1], \\ C_{дк}(C < 0)[V/2; V - 1]. \end{cases}$$

Припустимі значення аргументу C функції $C_{дк}(C)$ довжиною n біт обмежуються інтервалом числової вісі

$$C = \begin{cases} C_{дк}(C0)[0; V/2 - 1], \\ C_{дк}(C < 0) \left[\frac{V}{2}; V - 1 \right] - V \\ = \left[\frac{-V}{2}; -1 \right]. \end{cases} \quad (24)$$

Область визначення подібних функцій $A_{дк}(A)$ та $B_{дк}(B)$ суматора (18) задається аналогічними інтервалами:

$$A = \begin{cases} [0; V/2 - 1] & \text{при } [NA] = 0: A \geq 0, \\ [-V/2; -1] & \text{при } [NA] = 1: A < 0; \end{cases} B = \begin{cases} [0; V/2 - 1] & \text{при } [NB] = 0: A \geq 0, \\ [-V/2; -1] & \text{при } [NB] = 1: A < 0. \end{cases} \quad (25)$$

Визначення суми (14), згідно (25), визначається інтервалом $A + B = [-V/2; V/2-1] + [-V/2; V/2-1] = [-V; V-2]$, що значно ширше припустимих значень аргументу C функції $C_{дк}(C)$, що обмежені, згідно (24), інтервалом $[-V/2; V/2-1]$. Тому при $C = (A + B) \in [-V; -(V/2+1)]$ необхідно фіксувати від'ємне, а при $C = (A + B) \in [V/2; V-1]$ – додатне переповнення функції $C_{дк}(C)$.

В силу цього, ознаки переповнення суматора (18) описуються співвідношеннями:

$$\begin{aligned} \text{ОПП} &= \begin{cases} 1 \text{ при } C \in [-V; -(V/2+1)], \\ 0 \text{ при } C \in [-(V/2+1); V/2]; \end{cases} \\ \text{ППП} &= \begin{cases} 1 \text{ при } C \in [V/2; V-2], \\ 0 \text{ при } C \in [V/2; V-2]; \end{cases} \end{aligned} \quad (26)$$

$$\text{ПП} = \text{ОПП} \vee \text{ППП},$$

де ОПП, ППП, ПП – відповідно ознаки від'ємного та додатного переповнення або факту відсутності переповнення розрядної сітки суматора (17).

З метою спрощення ланцюгів фіксації факту та полярності переповнення кількості розрядів суматора подаємо доповняльні коди початкових даних суматора (17) рівносильним виразом

$$\begin{aligned} C_{\text{дк}}(n,1) &= (A_{\text{дк}}(n,1) + B_{\text{дк}}(n,1))_{mV} = \\ &= (A_{\text{дк}}(n,1) + B_{\text{дк}}(n,1) + V)_{mV} = \\ &= ((V/2 + A_{\text{дк}}(n,1))_{mV} + (V/2 + \end{aligned} \quad (27)$$

$+B_{\text{дк}}(n,1))_{mV} = (A_{\text{дк}}^{\text{PP}}(n,1) + B_{\text{дк}}^{\text{PP}}(n,1))_{mV}$, де $A_{\text{дк}}^{\text{PP}}(n,1) = (V/2 + A_{\text{дк}}(n,1))_{mV}$, $B_{\text{дк}}^{\text{PP}}(n,1) = (V/2 + B_{\text{дк}}(n,1))_{mV}$ – відповідно перетворені доповняльні коди операндів суматора (17).

Операцію додавання при відніманні, наприклад, функції $A_{\text{дк}}^{\text{PP}}(A_{\text{дк}})$ замінімо перетворенням коду $A_{\text{дк}}(n,1)$ наступним чином:

$$\begin{aligned} A_{\text{дк}}^{\text{PP}}(n,1) &= (V/2 + A_{\text{дк}}(n,1))_{mV} = \\ &= (V/2 + [NA]A_{\text{дк}}(n-1,1))_{mV} = \\ &= (2^{n-1} + [NA] \cdot 2^{n-1} + A_{\text{дк}}(n-1,1))_{mV} = \\ &= ((1 + [NA]) \cdot 2^{n-1})_{mV} + A_{\text{дк}}(n-1,1) = \\ &= [NA] \cdot 2^{n-1} + A_{\text{дк}}(n-1,1) = [NA] A_{\text{дк}}(n-1,1), \end{aligned} \quad (28)$$

де $((1 + [NA]) \cdot 2^{n-1})_{mV} = [NA] \cdot 2^{n-1}$.

Аналогічні співвідношення, вочевидь, справедливі також для подібної функції $B_{\text{дк}}^{\text{PP}}(B_{\text{дк}})$:

$$B_{\text{дк}}^{\text{PP}}(n,1) = [NB] B_{\text{дк}}(n-1,1). \quad (29)$$

При заміні доповняльних кодів операндів перетвореними виразами (28) та (29) для формування результату $C_{\text{дк}}(n,1)$ на входах суматора необхідно задавати, згідно (27), сукупність двійкових кодів:

$$\begin{aligned} a(n,1) &= A_{\text{дк}}^{\text{PP}}(n,1), \quad b(n,1) = B_{\text{дк}}^{\text{PP}}(n,1), \\ e &= 0, \quad SB = 0. \end{aligned} \quad (30)$$

Повна сума в операційному суматорі (30) на штатних виводах утворює двійковий поліном

$$\begin{aligned} \Sigma(n+1,1) &= E \cdot 2^n + [NC] \cdot 2^{n-1} + C_{\text{дк}}(n-1,1) = \\ &= A_{\text{дк}}^{\text{PP}}(n,1) + B_{\text{дк}}^{\text{PP}}(n,1) = \\ &= V/2 + A_{\text{дк}}(n,1) + V/2 + B_{\text{дк}}(n,1) = \\ &= V/2 + (V + A)_{mV} + V/2 + (V + B)_{mV} = \\ &= (V/2 + V + A)_{mV} + (V/2 + V + B)_{mV} = \\ &= V/2 + A + V/2 + B = \\ &= V + (A + B) = V + C, \end{aligned} \quad (31)$$

де $(V/2 + A)_{mV} = V/2 + A$; $(V/2 + B)_{mV} = V/2 + B$, оскільки $A \in [-V/2; V/2-1]$, $B \in [-V/2; V/2-1]$.

При від'ємному переповненні суматора (30) на інтервалі $C \in [-V; -(V/2+1)]$ повна сума змінюється в межах:

$$\Sigma(n+1,1) = E \cdot 2^n + [NC] \cdot 2^{n-1} + C_{\text{дк}}(n-1,1) =$$

$$\begin{cases} V + C = V + (-V) = 0 < 2^{n-1} \\ \text{при } C = -V, \\ V + C = V - (V/2 + 1) = 2^{n-1} - 1 < 2^{n-1} \\ \text{при } C = -(V/2 + 1). \end{cases}$$

Факт від'ємного переповнення суматора (30) описується комбінацією

$$E[NC] = 00. \quad (32)$$

В області додатного переповнення розрядної сітки при $C \in [V/2; V-1]$ повна сума на виході суматора (30) змінюється в межах:

$$\begin{cases} E \cdot 2^n + [NC] \cdot 2^{n-1} + C_{\text{дк}}(n-1,1) = \\ C + V = V + V/2 = 2^n + 2^{n-1} \text{ при } C = V/2, \\ V + C = V + (V - 2) = V + V - 2 \text{ при } C = V - 2. \end{cases}$$

Наявність факту додатного переповнення розрядної сітки суматора (30) описується комбінацією

$$E[NC] = 11. \quad (33)$$

За коректних значень суми $C \in [-V/2; -1]$ повна сума на виводах суматора (30) змінюється в межах:

$$\begin{cases} E \cdot 2^n + [NC] \cdot 2^{n-1} + C_{\text{дк}}(n-1,1) = \\ V + C = V - V/2 = V/2 = 2^{n-1} \\ \text{при } C = -V/2, \\ V + C = V - 1 = 2^n - 1 = 2^{n-1} + 2^{n-1} - 1 \\ \text{при } C = -1. \end{cases}$$

При $C < 0$ та коректних значеннях суми на виводах суматора (30) формується комбінація

$$E[NC] = 01. \quad (34)$$

При штатних значеннях суми $C \in [0; V/2-1]$ повна сума на виводах суматора (30) змінюється в межах:

$$\begin{cases} E \cdot 2^n + [NC] \cdot 2^{n-1} + C_{\text{дк}}(n-1,1) = \\ V + C = V + 0 = V = 2^n \text{ при } C = 0, \\ V + C = V + V/2 - 1 = 2^n + 2^{n-1} - 1 \\ \text{при } C = V/2 - 1. \end{cases}$$

За відсутності додатного переповнення розрядної сітки суматора (30) повна сума на його виводах описується в старших розрядах комбінацією

$$E[NC] = 10. \quad (35)$$

Комбінація $E[NC]$ на штатних виводах суматора (30), згідно (32)–(35), однозначно характеризує наявність або факт відсутності переповнення суматора. Наявність нештатних значень суми (14), згідно (32)–(33), описується логічними виразами:

$$\begin{aligned} \text{ОПП} &= \overline{E[NC]}; \quad \text{ППП} = E \wedge [NC]; \\ \text{ПП} &= \text{ОПП} \vee \text{ППП} = \overline{E} \cdot [NC] \vee E[NC] = \\ &= E[NC]. \end{aligned} \quad (36)$$

Результати VHDL-моделювання схемотехніки суматора-віднімача в режимі (30) наведені на рис.3. В машинному експерименті завдані та обчислені коди (при $n = 4$, $x = 2^4 = 16$):
 $A = -101$; $B = +010$;
 $A_{\text{дк}}(4,1) = (V + A)_{mV} = 16 + (-101) =$
 $= 111 - 0101 + 1 = 1101 + 1 = 1010 + 1 = [1]011$;
 $B_{\text{дк}}(4,1) = (16 + (+010))_{m16} = [0]010$;

$$\begin{aligned}
 A_{\text{ДК}}^{\text{PP}}(4,1) &= \overline{[1]011} = [0]011; \\
 B_{\text{ДК}}^{\text{PP}}(4,1) &= \overline{[0]010} = [1]010; \\
 E[\text{NC}] C_{\text{ДК}}(3,1) &= A_{\text{ДК}}^{\text{PP}}(4,1) + B_{\text{ДК}}^{\text{PP}}(4,1) = \\
 &= [0]011 + [1]010 = 0[1]101; \\
 E = 0; [\text{NC}] &= [1]; C_{\text{ДК}}(4,1) = [1]101; \\
 \text{ОПП} = \overline{E} [\text{NC}] &= \overline{0} \overline{1} = 0; \text{ППП} = E \wedge [\text{NC}] = \\
 &= 0 \wedge 1 = 0; \\
 \text{ПП} = \overline{E} [\text{NC}] &= \overline{0} \overline{1} = 0.
 \end{aligned}$$

Результати моделювання рівнянь суматора- віднімача в режимі (30) (рис.3) адекватно відображають результати приведених теоретичних розрахунків.

▣ A	B	<= 1011	▣
▣ B	2	<= 0010	▣
▣ SB	0	<= 0	
▣ Eout	0		
▣ P_OVR	0		
▣ N_OVR	0		
▣ OVR_DK	0		
▣ SDK	D		▣

Рисунок 3 - Результати моделювання операції додавання цілих чисел в системі доповняльних кодів

Синтез схем обчислення доповняльних кодів цілих чисел на основі програмованого суматора- віднімача беззнакових чисел

Операція віднімання цілих чисел зводиться до розв'язання задачі:

$$C = A - B, \quad (37)$$

де A, B – відповідно зменшуване та від'ємник різниці цілих чисел.

В режимі віднімання результат на виході програмованого суматора- віднімача, згідно (2), формується у вигляді залишку по модулю V:

$$\begin{aligned}
 (C)_{mV} &= (A - B)_{mV} = (V - V + A - B)_{mV} = \\
 &= ((V + A) - (V + B))_{mV} = \\
 &= ((V + A)_{mV} - (V + B)_{mV})_{mV} = \\
 &= (A_{\text{ДК}}(n,1) - B_{\text{ДК}}(n,1))_{mV}.
 \end{aligned} \quad (38)$$

Рішення задачі (37) при переключенні суматора- віднімача (рис.2) в режим віднімання (при SB = 1) реалізується, згідно (38), завданням операцій:

$$\begin{aligned}
 a(n,1) &= A_{\text{ДК}}(n,1), \quad b(n,1) = B_{\text{ДК}}(n,1), \\
 e = 0, \quad SB &= 1.
 \end{aligned} \quad (39)$$

В n-розрядному віднімачі, згідно (25), область визначення функцій $A_{\text{ДК}}(A)$ та $B_{\text{ДК}}(B)$ задається інтервалом $[-V/2; V/2-1]$. Різниця операндів $(A - B)$ визначається інтервалом $[-(V-1); V-1]$, котрий перевершує припустимий діапазон змінення аргументу функції $C_{\text{ДК}}(C) = (C)_{mV} = (V + C)_{mV} = C_{\text{ДК}}(n,1)$, обмеженої, згідно (24), інтервалом $[-V/2; V/2-1]$. В силу цього при $(A - B) \in [V-1; -(V/2 + 1)]$ необхідно фіксувати від'ємне переповнення, а при $(A - B) \in [V/2; V-1]$ – додатне переповнення розрядної сітки віднімача (39). Області від'ємного та додатного переповнення функції $C_{\text{ДК}}(n,1)$ у віднімача (39) та

суматорі (18) неоднакові, тому переповнення припустимого значення різниці чисел потребує спеціального дослідження.

З метою спрощення алгоритму та схеми фіксації переповнення розрядної сітки віднімача (39) замінимо обчислення різниці доповняльних кодів рівносильним виразом:

$$\begin{aligned}
 C_{\text{ДК}}(n,1) &= (A_{\text{ДК}}(n,1) - B_{\text{ДК}}(n,1))_{mV} = \\
 &= (V/2 + A_{\text{ДК}}(n,1) - (V/2 + B_{\text{ДК}}(n,1)))_{mV} = \\
 &= ((V/2 + A_{\text{ДК}}(n,1))_{mV} - (V/2 + \\
 &+ B_{\text{ДК}}(n,1))_{mV})_{mV} = (A_{\text{ДК}}^{\text{PP}} - B_{\text{ДК}}^{\text{PP}})_{mV}.
 \end{aligned} \quad (40)$$

де, згідно (28) та (29),

$$\begin{aligned}
 A_{\text{ДК}}^{\text{PP}}(n,1) &= \overline{[NA]} A_{\text{ДК}}(n-1,1), \\
 B_{\text{ДК}}^{\text{PP}}(n,1) &= \overline{[NB]} B_{\text{ДК}}(n-1,1).
 \end{aligned}$$

За прийнятих перетворень доповняльний код різниці (37) формується в програмованому суматорі- віднімачі (рис.2) завданням сукупності двійкових кодів:

$$\begin{aligned}
 a(n,1) &= A_{\text{ДК}}^{\text{PP}}(n,1); \quad b(n,1) = B_{\text{ДК}}^{\text{PP}}(n,1); \\
 e = 0; \quad SB &= 1.
 \end{aligned} \quad (41)$$

На виводах вичислителя (41), згідно (38), формуються двійкові коди:

$$\begin{aligned}
 S(n,1) &= C_{\text{ДК}}(n,1) = [\text{NC}] C_{\text{ДК}}(n-1,1) = \\
 &= [\text{NC}] \cdot 2^{n-1} + C_{\text{ДК}}(n-1,1) = \\
 &= (V + (A - B))_{mV} = \\
 &= \begin{cases} (A - B)_{\text{при } A - B \geq 0; E = \overline{m} = 1} \\ V + (A - B) = 2^n + (A - B) \\ \text{при } A - B < 0; E = \overline{m} = 0 \end{cases} \quad (42)
 \end{aligned}$$

В області від'ємного переповнення розрядної сітки віднімача (41) на штатних виходах віднімача, згідно (42), отримаємо:

$$E = \overline{m} = 0;$$

$$[\text{NC}] \cdot 2^{n-1} + C_{\text{ДК}}(n-1,1) =$$

$$\begin{cases} 2^n + (A - B) = 2^n - (V - 1) = 2^n - 2^n + 1 = 1 < 2^n \\ \text{при } A - B = -(V - 1); [\text{NC}] = 0; \\ 2^n + (A - B) = 2^n - \left(\frac{V}{2} + 1\right) = 2^n - 2^{n-1} - 1 = 2^{n-1} - 1 < 2^n \\ \text{при } A - B = -(V/2 + 1); [\text{NC}] = 0 \end{cases}$$

В межах області від'ємного переповнення розрядної сітки суматора- віднімача (41) на штатних виходах формується комбінація

$$E[\text{NC}] = 00. \quad (43)$$

В області додатного переповнення різниці при $C = A - B \in [V/2; V-1]$ на виводах суматора- віднімача (41), згідно (42), формуються коди:

$$E = \overline{m} = 1;$$

$$[\text{NC}] \cdot 2^{n-1} + C_{\text{ДК}}(n-1,1) =$$

$$\begin{cases} A - B = \frac{V}{2} = 2^{n-1} \\ \text{при } A - B = V/2; [\text{NC}] = \\ A - B = V - 1 = 2^n - 1 = 2^{n-1} + 2^{n-1} - \end{cases} \quad (44)$$

В області додатного переповнення розрядної сітки при $C = A - B \in [V/2; V-1]$ на штатних виводах суматора- віднімача (41) формується комбінація

$$E[\text{NC}] = 11. \quad (45)$$

В коректному режимі роботи суматора-віднімача (41) при $A - B \in [0; V/2 - 1]$ та $A - B \geq 0$, згідно (44) в старших розрядах вихідного поліному отримаємо:

$$E = \overline{m} = 1;$$

$$S(n,1) = [NC] \cdot 2^{n-1} + C_{ДК}(n-1,1) =$$

$$= \begin{cases} A - B = 0 < 2^{n-1} \\ \text{при } A - B = 0: [NC] = 0; \\ A - B = \frac{V}{2} - 1 = 2^{n-1} - 1 < 2^{n-1} \\ \text{при } A - B = V/2 - 1: [NC] = 0 \end{cases} \quad (46)$$

За коректних значень різниці $C = A - B \geq 0$, на штатних виводах суматора-віднімача (41) формується комбінація

$$E[NC] = 10. \quad (47)$$

При $A - B < 0$ в області визначення доповняльного коду результату $[-V/2; -1]$, згідно (42), отримаємо:

$$E = \overline{m} = 0;$$

$$[NC] \cdot 2^{n-1} + C_{ДК}(n-1,1) =$$

$$= \begin{cases} V + (A - B) = V - V/2 = 2^{n-1} \\ \text{при } C = -V/2: [NC] = 1; \\ V + (A - B) = V - 1 = 2^n - 1 = 2^{n-1} + 1 \\ \text{при } A - B = -1: [NC] = 1 \end{cases} \quad (48)$$

Таким чином, при $A - B < 0$ переповнення розрядної сітки суматора-віднімача відсутнє при комбінації

$$E[NC] = 01. \quad (49)$$

Наявність або відсутність факту переповнення суматора-віднімача (41) та суматора-віднімача (30), згідно (32)–(35) та (42)–(49), описуються однаковою сукупністю змінних $E[NC]$. Тому ознаки переповнення суматора-віднімача в режимі віднімання описуються логічними виразами (36), отриманими для режиму додавання початкових даних. В силу цього, схеми фіксації переповнення в об'єднаному суматорі-віднімача (рис.4) інваріантні відносно режиму роботи програмованого суматора-віднімача.

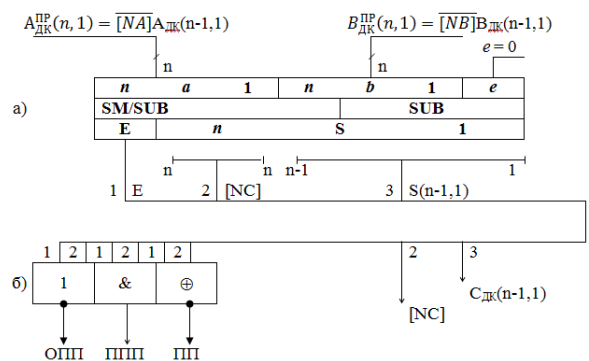


Рисунок 4 – Об'єднана схема додавання-віднімання цілих чисел в системі доповняльних кодів: а) операційна схема; б) схема фіксації переповнення

В об'єднаній схемі додавання та віднімання цілих чисел в системі доповняльних кодів сукупність двійкових кодів на основних інформаційних входах програмованого суматора-віднімача беззнакових цілих чисел, згідно (30) та (41), задається також однаковим набором змінних.

Результати VHDL-моделювання рівнянь об'єднаної схеми додавання та віднімання цілих чисел в системі доповняльних кодів показані на рис.5. В машинному експерименті використані початкові дані та обчислювались результати операції віднімання цілих чисел в доповняльному коді (при $n = 4, V = 2^4 = 16$):

$$A = -011; B = +010; A_{ДК}(4,1) = (16 + (-011))_{m16} = 15 - 011 + 1 = 1111 - 011 + 1 = 1011 + 1 =$$

$$= 1100 + 1 = [1]101;$$

$$B_{ДК}(4,1) = (16 + (+010))_{m16} = [0]010;$$

$$A_{ДК}^{ПП}(4,1) = [1]101 = [0]101;$$

$$B_{ДК}^{ПП}(4,1) = [0]010 = [1]010$$

$$A - B = (-011) - (+010) = -101 < 0 : M =$$

1;

$$E = \overline{M} = 0;$$

$$C_{ДК}(4,1) = (A_{ДК}^{ПП}(4,1) + B_{ДК}^{ПП}(4,1))_{m16} =$$

$$= ([0]101 - [1]010)_{m16} = (-0101)_{m16} = 16 -$$

$$0101 =$$

$$= 15 - 0101 + 1 = 111 - 0101 + 1 = 1010 +$$

$$1 =$$

$$= [1]011 : [NC] = 1;$$

$$ОПП = \overline{E[NC]} = \overline{01} = 0;$$

$$ППП = E \wedge [NC] = 0 \wedge 1 = 0;$$

$$ПП = \overline{E[NC]} = \overline{01} = 0.$$

Машинні коди у VHDL-моделі рівнянь об'єднаної схеми додавання та віднімання чисел в доповняльному коді адекватно відбивають результат теоретичних розрахунків.

⊕ A	<= 1101	0
⊕ B	<= 0010	2
SB	<= 1	
Eout		
P_OVR		
N_OVR		
OVR_DK		
⊕ SDK		B

Рисунок 5 - Результати моделювання режиму віднімання цілих чисел в системі доповняльних кодів

Висновки

Основні результати роботи зводяться до наступного:

– розроблено математичний опис та визначені властивості програмованого суматора-віднімача беззнакових цілих чисел як

операційного пристрою схем додавання та віднімання цілих чисел;
– запропоновано метод подання початкових даних програмованого суматора-віднімача беззнакових чисел для обчислення суми та різниці цілих чисел в доповняльному коді;
– запропоновано доцільний спосіб форматування доповняльних кодів операндів, за якого вихідний поліном операційного суматора дозволяє визначити переповнення доповняльного коду результату операції;
– показано, що запропоновані схеми фіксації наявності факту та полярності переповнення

розрядної сітки операційного суматора інваріантні відносно режиму роботи програмованого суматора-віднімача;
– коректність розроблених алгоритмів обґрунтовано моделюванням математичного опису схем додавання та віднімання цілих чисел в системі доповняльних кодів;
– запропоновані алгоритми та схеми додавання та віднімання цілих чисел в доповняльному коді створюють передумови для розробки конкурентоспроможних пристроїв більш складних машинних операцій.

Список літератури

1. Жуков И.А. Компьютерная схемотехника. Методы построения и проектирования / И.А.Жуков, Н.П.Бабич. – К.: МК-Пресс, 2004. – 576с.
2. Трейстер Р. Радиолюбительские схемы на ИС типа 555 / Р. Трейстер; пер. с англ. В.А. Давыдова, А.И.Зильбермана; под ред. Л.С.Ходоша. – М.: Мир, 1988. – 262с.
3. Texas Instruments Designs – [Електронний ресурс] Режим доступу: <http://www.ti.com/>.
4. Карцев М.А. Арифметика цифровых машин. – М.: Наука, 1969. – 576с.
5. Уэйкерли Дж.Ф. Проектирование цифровых устройств: Т.1 / Дж.Ф.Уэйкерли. – М.: Постмаркет, 2002. – 544с.
6. Святный В.А. Математическое описание компьютерных операций суммирования и вычитания целых чисел при смещенных кодах операндов / В.А.Святный, В.В.Лапко, А.В.Самощенко. // Наукові праці Донецького національного університету. Серія «Інформатика, кібернетика та обчислювальна техніка». – 2016. – Вип.1(22). – с.75-83.
7. Каган Б.М. Цифровые вычислительные машины и системы / Б.М.Каган, М.М.Каневский. – М.: Энергия, 1974. – 680с.

References

1. Zhukov I.A. (2004) Computer circuit design. Methods of construction and design. [*Komp'yuternaya shemotekhnika. Metody postroeniya i proektirovaniya*], MK-Press, Kiev, 576 p.
2. Robert J Traister (1985) *The 555 IC project book*, McGraw-Hill/TAB Electronics, 263 p.
3. Texas Instruments Designs, available at: <http://www.ti.com/>.
4. Kartsev, M.A. (1969) *Arithmetic of digital computers [Arifmetika tsifrovyykh mashin]*, Nauka, Moscow, 576 p.
5. Ue'ykerli Dzh.F. (2002) Digital devices design. : vol 1 [*Proektirovanie cifrovyyh ustroystv TI*], Postmarket, Moscow, 544 p.
6. Sviyatny, V.A. , Lapko, V.V., Samoschenko, A.V.(2016) *Mathematical description of computer operations of summation and subtraction of integers with offset operand codes [Matematicheskoye opisaniye komp'yuternyykh operatsiy summirovaniya i vychitaniya tselykh chisel pri smeshchennykh kodakh operandov]* Naukovi pratsi DonNTU: Informatyka, Kybernetyka ta obchysliuvalna tekhnika, №1 (22), Donetsk National Technical University, Krasnoarmiysk, P. 75-83.
7. Kagan B.M., Kanevsky M.M. (1974) *Digital computers and systems [Tsifrovyye vychislitel'nyye mashiny i sistemy]*, Energy, Moscow, 680 p.

Надійшла до редакції 17.10.2017

В.В. ЛАПКО, А.В. САМОЩЕНКО, Г.Э.МАРГИЕВ

Донецкий национальный технический университет, г. Покровск

СИНТЕЗ ПРОГРАММИРУЕМЫХ СХЕМ СЛОЖЕНИЯ И ВЫЧИТАНИЯ ЦЕЛЫХ ЧИСЕЛ В СИСТЕМЕ ДОПОЛНИТЕЛЬНЫХ КОДОВ

Представлено математическое описание и определены свойства программируемого сумматора-вычитателя беззнаковых целых чисел в качестве операционного модуля схем сложения и вычитания. Предложен метод формализованного определения исходных данных программируемого сумматора-вычитателя неотрицательных целых чисел для вычисления суммы и разности чисел в системе дополнительных кодов. Разработан целесообразный способ форматирования дополнительного кода операндов, отличный тем, что два старших разряда выходного полинома операционного сумматора-вычитателя позволяют определить отрицательное и положительное переполнения или факт отсутствия

переполнения разрядной сетки результата операции. Показано, что предложенные схемы фиксации наличия и полярности переполнения инвариантны относительно режима работы операционного сумматора-вычитателя. Корректность разработанного математического описания и алгоритмов подтверждаются VHDL-моделированием уравнений программируемых схем сложения и вычитания чисел в системе дополнительных кодов.

Ключевые слова: *дополнительный код, программируемый сумматор-вычитатель, переполнение, компьютерное моделирование.*

V. V. LAPKO, A. V. SAMOSHCHENKO, H. E. MARHIEV

Donetsk National Technical University, Pokrovsk

SYNTHESIS ADDITION AND SUBTRACTION PROGRAMABLE SCHEMES OF INTEGERS IN THE TWO'S-COMPLEMENT CODE

To minimize hardware costs in modern processor of computer systems the multifunctional programmable arithmetic logic unit (ALU) is used. ALU instruction set consists of operation device of adding and subtracting integers. Using ALU in the schemes of adding and subtracting unsigned integers in the system of double complement codes is an important theoretical and practical question of joint and coordinated usage of modes of adding and subtracting unsigned integers as well as a question of advantageous coding raw data for calculating the sum and difference. Joint and concerted usage of arithmetic operations of addition and subtraction of unsigned integers in the system of double complement codes requires further research. An urgent task is to develop a formal mathematical description of ALU as an operation device scheme of addition and subtraction of unsigned integers in double complement codes. Because of limit of a bit grid ALU the important task is also to develop a simple and fast way of analyzing the correctness of double complement codes result. Optimization of this task is reduced to minimize the application of a number of inputs in the control scheme and a possibility to use only regular outputs of ALU in a control scheme. In addition, an important quality criterion of control schemes is invariant algorithm for fixing the overflow of a bit grid relatively to the mode of ALU. In this work the synthesis method of optimal operation device processing integers in the system of double complement codes based on programmable adder-subtractor unsigned integers is suggested.

In the article we designed the mathematical description and defined characteristics of programmable adder-subtractor of unsigned numbers as operation device scheme of addition and subtraction integers. It is designed by expedient way of formatting double complement code operands, which are different from the two high bits of output operation adder-subtractor polynomial, and allow determining positive and negative bit grid overflow or the absence of its overflow as an operation result. It is shown that the suggested schemes of fixation presence and polarity overflow are invariant relevantly to the adder-subtractor operational mode. Correctness of the developed mathematical description and algorithms are grounded on the VHDL-modeling of equations of programmable circuits of addition and subtraction of integers in the system of complement codes.

Keywords: *double complement, programmable adder-subtractor, overflow, computer simulation.*