УДК 681.325.5

РЕКУРСИВНІ МЕТОДИ ТА АЛГОРИТМИ УНІВЕРСАЛЬНИХ АДИТИВНИХ ПЕРЕТВОРЕНЬ В АМ-СИСТЕМАХ ЧИСЛЕННЯ

О.Д. Азаров, О.І. Черняк

Анотація. Запропоновано рекурсивні методи та алгоритми, що дозволяють в *АМ*-системах числення виконувати універсальні адитивні перетворення на основі елементарних адитивних перетворень.

Аннотация. Предложены рекурсивные методы и алгоритмы, позволяющие в *АМ*-системах счисления выполнять универсальные аддитивные преобразования на основе элементарных аддитивных преобразований.

Annotation. The recursive methods and algorithms allowing in *AM*-numerical systems to carry out universal additive transformations by means of elementary additive transformations are offered.

Ключові слова: рекурсивні методи, *АМ*-система.

Вступ

При організації розподілених обчислень виникає проблема, що полягає у необхідності реалізації великої кількості інформаційних зв’язків між пристроями. Одним з відомих підходів до вирішення даної проблеми є порозрядне конвеєрне оброблянння послідовних кодів, що дозволяє значно зменшити кількість інформаційних зв’язків без суттєвого зменшення продуктивності. Для вирішення широкого кола задач таке оброблянння вимагає використання надлишкових систем числення, оскільки лише у них можливе порозрядне виконання усіх арифметичних операцій у єдиному потоці за рахунок обмеження перенесення у старші розряди. Від довжини перенесення залежать апаратні витрати пристроїв, що реалізують таке оброблянння. Серед відомих надлишкових позиційних систем числення для порозрядного оброблянння найбільш широко використовується знакорозрядна [1,2,3,4] з цифрами -1, 0, 1. Для передавання одного розряду у цій системі числення потрібно два інформаційних зв’язки. Відомо також порозрядне оброблянння у кодах золотої пропорції з цифрами 0 і 1 [5]. Це дозволяє передавати один розряд коду через один інформаційний зв’язок.

Актуальність

Можна створити безліч систем числення з будь-якою кількістю цифр, що дозволяють виконувати порозрядне оброблянння. Кожна з них має різну довжину перенесення при додаванні і тому потребує різних апаратних витрат для реалізації пристроїв. Отже, актуальною є задача визначення системи числення, що забезпечує мінімальні апаратні витрати на реалізацію порозрядних пристроїв при мінімальній кількості інформаційних зв’язків. Для вирішення даної задачі авторами запропоновано клас систем числення, що узагальнює відомі та дозволяє створювати нові системи числення з можливістю порозрядного оброблянння [6]. Вони названі *АМ*-системами числення. Будь-яка *АМ*-система числення може бути описана за допомогою множини цифр $\{0...c_{k-1}\}$, основи системи числення w і адитивного співвідношення ${}^{t}A^{\tau,p} : w^{tp} = R^{\tau,p}$, де $R^{\tau,p} = \sum_{i=0}^{p} r_i \cdot w^{\tau i}$ - граничне значення ($r \in \{0...c_{k-1}\}$). Наявність адитивних співвідношень в *АМ*-системах числення дозволяє ввести операції адитивного перетворення кодів (*А*-перетворення), що змінюють код при збереженні його числового еквіваленту. *А*-перетворення здійснюють перенесення і запозичення та можуть використовуватись при додаванні і відніманні кодів. За напрямком перенесення *А*-перетворення поділяються на перетворення з перенесенням у старші розряди (*AL*) і перетворення з перенесенням у молодші розряди (*AR*). Для здійснення *А*-перетворень необхідно виконання певних умов. Необхідна умова i-го *AL*-перетворення: значення розрядів від 0-го до i-го повинно бути не меншим ніж відповідне граничне значення, а значення ($i+t$)-го розряду повинно бути меншим ніж старша цифра. Необхідна умова i-го *AR*-перетворення: значення розрядів від 0-го до i-го повинно бути не більшими ніж різниця між максимальним кодом у цих розрядах і відповідним граничним значенням, а значення ($i+t$)-го розряду повинно бути більшим нуля. За достатніми умовами виконання *А*-перетворення поділяються на елементарні (*E*), універсальні (*U*) та повні (*F*). Для виконання *EA*-перетворень достатньо, щоб у кожному розряді перетворюваного коду виконувались умови, що не допускали б переповнення розрядів чи виникнення у них від’ємних значень. Однак можливі випадки, у яких не всі розряди перетворюваного коду задовольняють достатній умові *EA*-перетворення, хоча у цілому значення коду задовольняє необхідній умові. Оскільки у таких випадках *EA*-перетворення не виконуються, то вони не можуть використовуватись як перенесення і запозичення при виконанні порозрядного додавання і віднімання. Для цього слід застосовувати *UA*-перетворення.

Достатні умови *UA*-перетворень подібні до необхідних умов. Відмінність полягає у тому, що для i-го *UA*-перетворення перевіряються розряди не від 0-го до i-го, а від ($i-\tau b$)-го до i-го. Перевірка достатніх умов та виконання *UA*-перетворень у загальному випадку є досить складними для реалізації. Тому актуальною є розробка спрощених методів та алгоритмів *UA*-перетворень в *АМ*-системах числення.

Мета

Метою даної статті є підвищення ефективності проектування пристроїв порозрядної обробки за рахунок виконання універсальних адитивних перетворень в *AM*-системах числення як послідовності елементарних адитивних перетворень.

Задачі

Для досягнення даної мети необхідно вирішити такі задачі:

1) проаналізувати усі можливі ситуації, що виникають при виконанні достатніх умов *UA*-перетворень;
2) для кожної ситуації визначити послідовність *EA*-перетворень.

Розв'язання задач

Методи виконання *UA*-перетворень на основі *EA*-перетворень зручно представити у рекурсивному виді. Для визначення послідовності *EA*-перетворень, що приводять до виконання *i*-го *UAL*-перетворення над розрядами коду від *i*-го до (i-τb)-го, необхідно розглянути три можливих випадки виконання достатніх умов *UA*-перетворень. У кожному з цих випадків вважається, що $x_{i+t} < c_{k-1}$. Перший випадок – якщо у розрядах від від *i*-го до (i-τb)-го виконується умова *i*-го *EAL*-перетворення, тобто для кожного *j*-го з порівнюваних розрядів коду від *i*-го до (i-τp)-го виконується $x_{i-j} \geq r_{\tau p-j}$:

$$\underset{0<j<\tau p}{\forall}(x_{i-j} \geq r_{\tau p-j}) .$$

У даному випадку *i*-е *UAL*-перетворення зводиться до *i*-го *EAL*-перетворення:

$${}^{t}UAL_{i}^{\tau,p}(X_{i-\tau b}^{\tau b+t})={}^{t}EAL_{i}^{\tau,p}(X_{0}^{n-1}) .$$

Після цього виконання *i*-го *UAL*-перетворення завершується. Другий випадок – якщо умова *EAL*-перетворення виконується тільки для групи розрядів від *i*-го до (i-τg)-го, де $g<p$, і для деякого $0 \leq v \leq g(\tau+1)/t$ значення (i-τg+tv)-го розряду цієї групи більше (τg-tv)-го розряду граничного значення:

$$\underset{0\leq j\leq g}{\forall}(x_{i-\tau j} \geq r_{\tau(p-j)}) \wedge x_{i-\tau(g+1)} < r_{\tau(p-g-1)} \wedge \underset{0\leq v\leq g(\tau+1)/t}{\exists}(x_{i-\tau g+tv} > r_{\tau g-tv}) .$$

У даному виконується безумовне (i-τg+t(v-1))-е *EAR*-перетворення, а потім знову рекурсивно виконується *i*-те *UAL*-перетворення:

$${}^{t}UAL_{i}^{\tau,p}(X_{i-\tau b}^{\tau b+t})={}^{t}UAL_{i}^{\tau,p}({}_{-}^{t}EAR_{i-\tau g+t(v-1)}^{\tau,p}(X_{0}^{n-1})) .$$

Тут $_{-}EAR$ – безумовне *EAR*-перетворення, що виконується без перевірки умови. При виконанні безумовного *EAR*-перетворення за рахунок додавання у молодших розрядах може виникнути переповнення. Для зменшення величини переповнення другий випадок потрібно аналізувати, починаючи з v=0 і збільшуючи кожного разу v на одиницю. Третій випадок – якщо не виконується умова першого чи другого випадку для *i*-го *UAL*-перетворення. У даному випадку виконується (i-t)-е *UAL*-перетворення, а після його завершення знову рекурсивно виконується *i*-те *UAL*-перетворення.

$${}^{t}UAL_{i}^{\tau,p}(X_{i-\tau b}^{\tau b+t})={}^{t}UAL_{i}^{\tau,p}({}^{t}UAL_{i-t-\tau g}^{\tau,p}(X_{i-t-\tau(b+g)}^{\tau p+t})) .$$

UAL-перетворення у третьому випадку знову може призвести до рекурсивного *UAL*-перетворення у молодших розрядах перетворюваного коду, і так далі, поки номер перетворення не менший ніж i-τb. Якщо після останнього *UAL* перетворення умова відповідного *EAL*-перетворення не виконується, то *i*-е *UAL*-перетворення завершується. Отже, рекурсивний метод *i*-го *UAL*-перетворення на основі *EA*-перетворень такий. Перед початком виконання перетворення глобальній змінній *m* присвоюється значення *i*. Далі викнуються такі дії.

п.1 Якщо виконується умова $m<i$-τb, то закінчити перетворення.

п.2 Якщо не виконується умова $\underset{0<j<\tau p}{\forall}(x_{m-j} \geq r_{\tau p-j})$, то перейти на п.4.

п.3 Виконати ${}^{t}EAL_{m}^{\tau,p}(X_{0}^{n-1})$ і закінчити перетворення.

п.4 Якщо для не виконується умова

$$\underset{0\leq g<p}{\exists}(\underset{0\leq j\leq g}{\forall}(x_{m-\tau j} \geq r_{\tau(p-j)}) \wedge x_{m-\tau(g+1)} < r_{\tau(p-g-1)} \wedge \underset{0\leq v<g(\tau+1)/t}{\exists}(x_{m-\tau g+tv} > r_{rg-tv})),$$

то перейти на п.6.

п.5 Виконати ${}_{-}^{t}EAR_{m-\tau g+t(v-1)}^{\tau,p}(X_{0}^{n-1})$, перейти на п.7.

п.6 Виконати ${}^{t}UAL_{m-t-\tau g}^{\tau,p}(X_{m-t-\tau(b+g)}^{\tau b+t})$.

п.7 Виконати ${}^{t}UAL_{m}^{\tau,p}(X_{m-\tau b}^{\tau b+t})$.

Більш детально універсальне *AL*-перетворення на основі елементарних *A*-перетворень представлено граф-схемою алгоритму на рис. 1.

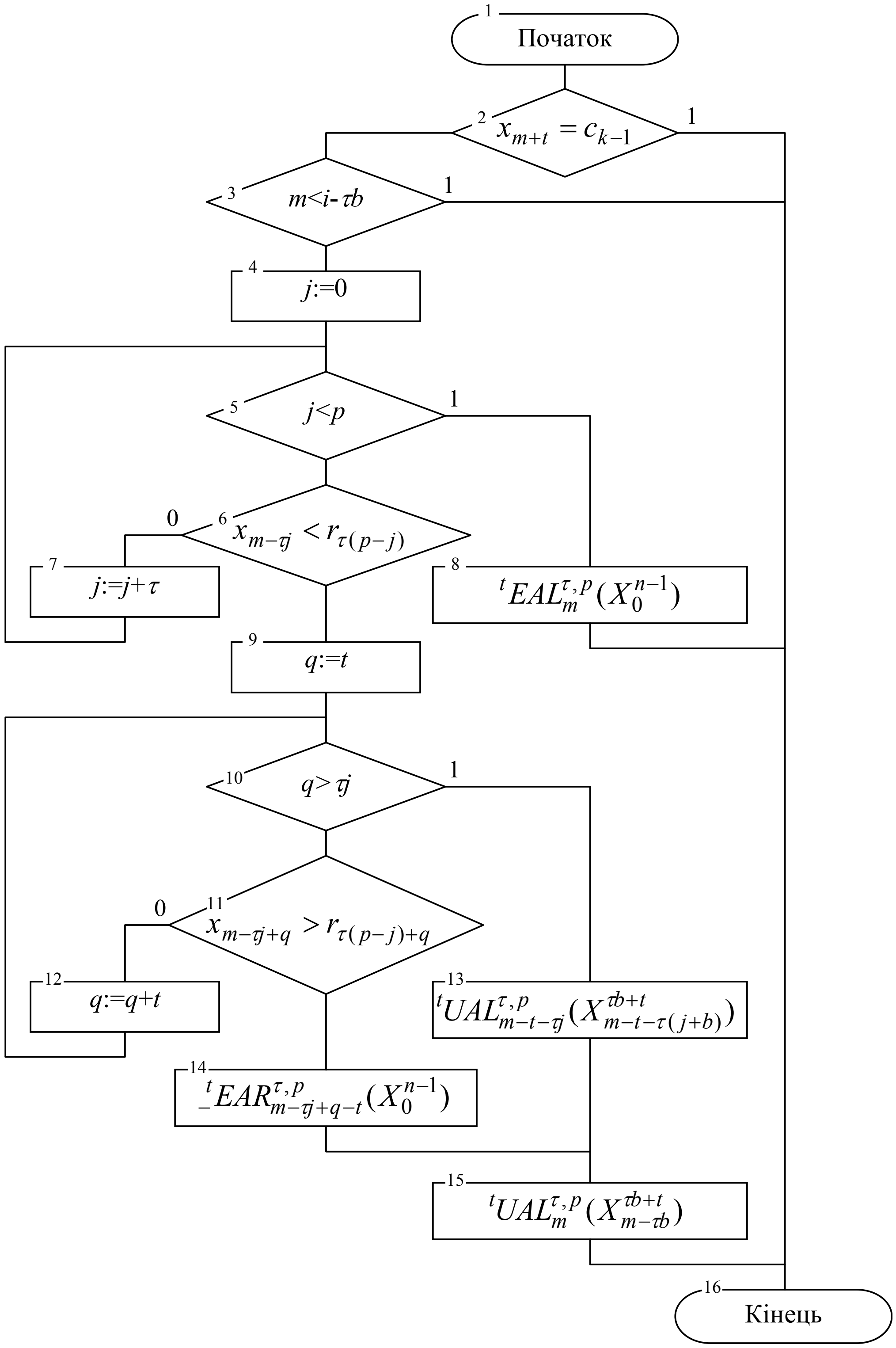

Рисунок 1 – Граф-схема алгоритму виконання *UAL*-перетворення на основі *EAL*- і *EAR*-перетворень

Для визначення послідовності *EA*-перетворень, що приводять до виконання *UAR*-перетворення над розрядами коду від *i*-го до (*i*-*τb*)-го, необхідно також розглянути три випадки. В кожному з цих випадків вважається, що $x_{i+t} > 0$. Перший випадок – коли в розрядах від *i*-го до (*i*-*τb*)-го виконується умова *EAR*-перетворення:

$$\forall_{0<j<\tau p} (x_{i-j} + r_{\tau p-j} \le c_{k-1}).$$

У даному випадку *i*-е *UAR*-перетворення зводиться до *i*-го *EAR*-перетворення:

$$^{t}UAR_{i}^{\tau,p}(X_{i-\tau b}^{\tau b+t}) = {}^{t}EAR_{i}^{\tau,p}(X_{0}^{n-1}).$$

Після цього виконання *i*-го *UAR*-перетворення завершується. Другий випадок – якщо умова *EAR*-перетворення виконується тільки для групи розрядів від *i*-го до (*i*-*τg*)-го (де *g*<*p*) і для деякого $0 \le v \le g(\tau+1)/t$ значення (*i*-*τg*+*tv*)-го розряду цієї групи менше величини $c_{k-1} - r_{\tau g - tv}$:

$$\forall_{0\le j\le g} (x_{i-\tau j} + r_{\tau(p-j)} \le c_{k-1}) \wedge x_{i-\tau(g+1)} +$$
$$+ r_{\tau(p-g-1)} > c_{k-1} \wedge \exists_{0\le v\le g(\tau+1)/t} (x_{i-\tau g+tv} < c_{k-1} - r_{\tau g-tv}).$$

У даному випадку виконується безумовне (*i*-*τg*+*t*(*v*-1))-е *EAL*-перетворення, а потім знову рекурсивно виконується *i*-те *UAR*-перетворення:

$$^{t}UAR_{i}^{\tau,p}(X_{i-\tau b}^{\tau b+t}) = {}^{t}UAR_{i}^{\tau,p}({}_{-}^{t}EAL_{i-\tau g+t(v-1)}^{\tau,p}(X_{0}^{n-1})).$$

Тут $\_EAL$ – безумовне *EAL*-перетворення, що виконується без перевірки умови. При виконанні безумовного *EAL*-перетворення за рахунок віднімання у молодших розрядах може виникнути від’ємне значення. Для зменшення абсолютної величини від’ємного значення другий випадок потрібно аналізувати, починаючи з *v*=0 і збільшуючи кожного разу *v* на одиницю. Третій випадок – коли не виконується умова першого чи другого випадку для *i*-го *UAR*-перетворення. У даному випадку рекурсивно виконується (*i*-*t*-*τg*)-е *UAR*-перетворення, а після його завершення знову рекурсивно виконується *i*-те *UAR*-перетворення.

$$^{t}UAR_{i}^{\tau,p}(X_{i-\tau b}^{\tau b+t}) = {}^{t}UAR_{i}^{\tau,p}({}^{t}UAR_{i-t-\tau g}^{\tau,p}(X_{i-t-\tau(b+g)}^{\tau b+t})).$$

Отже, виконання *UAR*-перетворення у третьому випадку знову може призвести до рекурсивного виконання *UAR*-перетворення у молодших розрядах перетворюваного коду, і так далі, поки номер перетворення залишається не меншим ніж *i*-*τb*. Якщо після виконання останнього *UAR* перетворення умова відповідного *EAR*-перетворення не виконується, то *i*-е *UAR*-перетворення завершується.

Метод виконання *i*-го *UAR*-перетворення на основі *EA*-перетворень такий. На кожній ітерації виконується *UAR*-перетворення з номером *m*. Перед початком виконання перетворення глобальній змінній *m* присвоюється значення *i*. Далі виконуються такі дії.

п.1 Якщо виконується умова *m* =*i*-*τb*, то завершити перетворення.

п.2 Якщо не виконується умова

$$\forall_{0<j<\tau p} (x_{m-j} + r_{\tau p-j} \le c_{k-1}),$$

то перейти на п.4.

п.3 Виконати $^{t}EAR_{m}^{\tau,p}(X_{0}^{n-1})$ і завершити перетворення.

п.4 Якщо не виконується умова

$$\forall_{0\le j\le g} (x_{m-\tau j} + r_{\tau(p-j)} \le c_{k-1}) \wedge x_{m-\tau(g+1)} +$$
$$+ r_{\tau(p-g-1)} > c_{k-1} \wedge \exists_{0\le v\le g(\tau+1)/t} (x_{i-\tau g+tv} < c_{k-1} - r_{\tau g-tv}),$$

то перейти на п.6.

п.5 Виконати ${}_{-}^{t}EAL_{m-\tau g+t(v-1)}^{\tau,p}(X_{0}^{n-1})$, перейти на п.7.

п.6 $^{t}UAR_{m-t-\tau g}^{\tau,p}(X_{m-t-\tau(b+g)}^{\tau b+t})$

п.7 $\quad {}^{t}UAR_{m}^{\tau,p}(X_{m-\tau b}^{\tau b+t})$.

Більш детально універсальне *AR*-перетворення на основі елементарних *A*-перетворень представлено граф-схемою алгоритму на рис. 2.

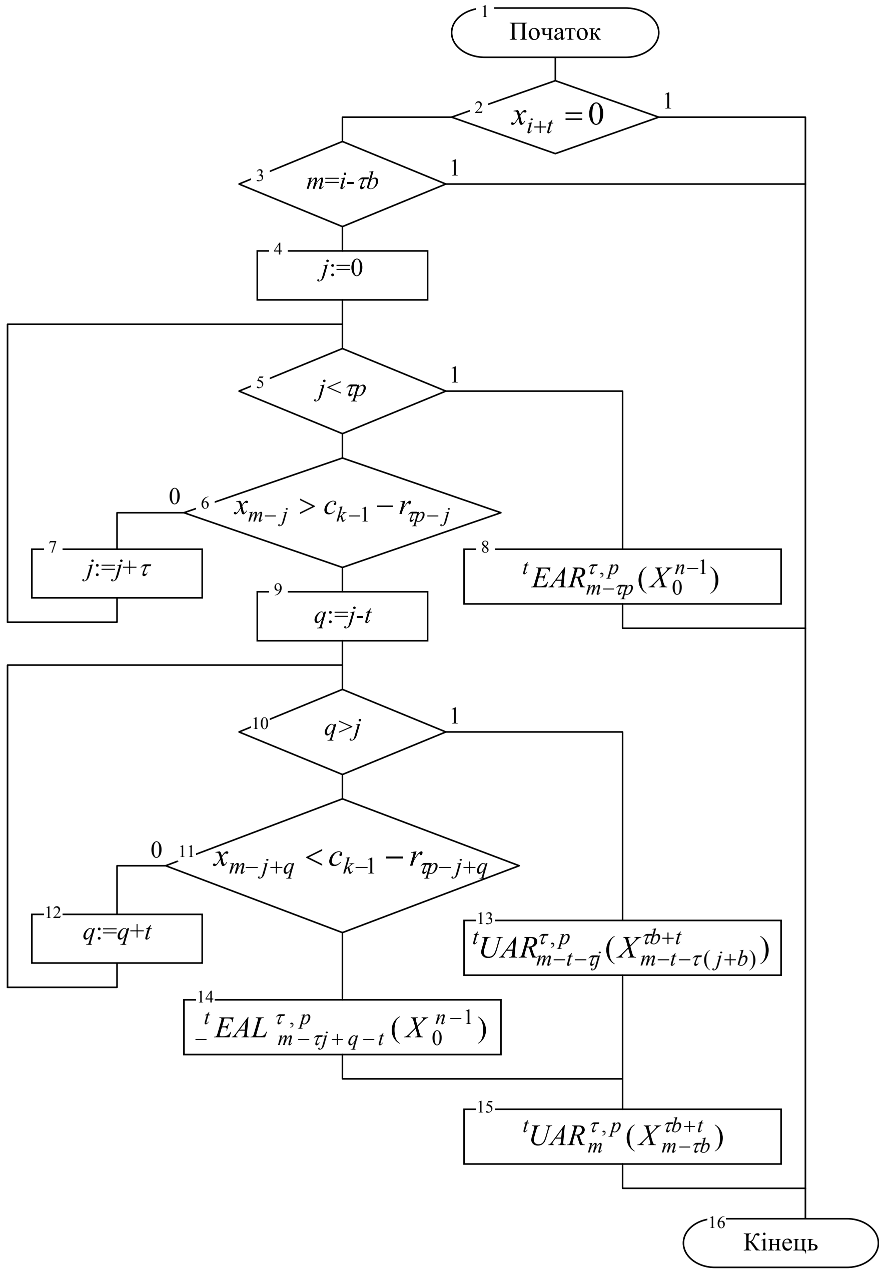

Рисунок 2 – Граф-схема алгоритму виконання *UAR*-перетворення на основі *EAR*- і *EAL*-перетворень

Нижче наведено приклади виконання *UAL*- та *UAR*-перетворень за допомогою *EA*-перетворень для деякої конкретної *AM*-системи числення.

Параметри *AM*-системи числення:

$$\left\{ \begin{array}{l} C_4 = \{0,1,2,3\}; \\ {}^{1}A^{1,3} : 10000 = 3221 \end{array} \right\}.$$

Код числа: $X_0^9 = 111331101$.

${}^1UAL_5^{1,3}$ -перетворення:

1	1	1	3	3	1	1	0	1	
				-1	+3	+2	+2	+1	${}_-^1EAR_3^{1,3}$
1	1	2	3	2	4	3	2	2	
		+1	-3	-2	-2	-1			${}^1EAL_5^{1,3}$
1	1	3	0	0	2	2	2	2	

Код числа: $X_0^9 = 2020003322$.

${}^1UAR_6^{1,3}$ -перетворення:

2	0	2	0	0	0	3	3	2	2	
					+1	-3	-2	-2	-1	${}_-^1EAL_3^{1,3}$
2	0	2	0	0	1	0	1	0	1	
		-1	+3	+2	+2	+1				${}^1EAR_6^{1,3}$
2	0	1	3	2	3	1	1	0	1	

Як видно з проведеного аналізу, рекурсивні методи і алгоритми *UAL*- та *UAR*-перетвореннь дозволяють виконувати їх на основі *EA*-перетворень обох типів подібним чином. При виконанні кожного з *UA*-перетворень аналізується три випадки. У першому випадку *i*-те *UA*-перетворення зводиться до *i*-го *EA*-перетворення того ж типу. У другому випадку *i*-те *UA*-перетворення потребує попереднього виконання безумовного *v*-го *EA*-перетворення протилежного типу і подальшого рекурсивного повторення виконання *i*-го *UA*-перетворення того ж типу. У третьому випадку *i*-те *UA*-перетворення потребує рекурсивного виконання *UA*-перетворення того ж типу у молодших розрядах. Завершується рекурсивний алгоритм *i*-го *UA*-перетворення після виконання *i*-го *EA*-перетворення того ж самого типу або, якщо після аналізу не виконується жодна з описаних умов.

Висновок

Запропоновані рекурсивні методи і алгоритми дозволяють виконувати універсальні адитивні перетворення за допомогою скінченної послідовності елементарних адитивних перетворень у будь-якій *АМ*-системі числення, що спрощує розробку таблиць переходів при проектуванні пристроїв порозрядного обробляння у цих системах числення.

Список літератури

1. Avizenis A. Binary-compatible signet-digit arithmetic. IN: AFIPS Conf Proc. – Vol. 26 – P1. – 1964 – P.663.
2. Ch. Frougny, On-line finite automata for addition in some numeration systems. Theoretical Informatics and Applications 33 (1999), 79–101.
3. Самофалов К.Г., Луцкий Г.М. Основы построения конвейерных ЭВМ.- Киев: Вища школа, 1981. – 234 с.
4. Каляев А.В. Многопроцессорные системы с программируемой архитектурой. – М.: Радио и связь, 1984. – 240 с.
5. Методи конвеєрної порозрядної обробки послідовних кодів золотої пропорції / О.І. Черняк, О.Д. Азаров // Вісник ВПІ. – 1996. - №1. – С. 14-17.
6. Системи числення з адитивними та мультиплікативними співвідношеннями між вагами розрядів / О.Д.Азаров, О.І.Черняк, П.О.Черняк // Вісник ВПІ. – 2001. - №1. – С. 58-64.

Відомості про авторів

Азаров Олексій Дмитрович, д.т.н., професор, директор інституту інформаційних технологій та комп’ютерної інженерії, завідувач кафедри обчислювальної техніки Вінницького національного технічного університету; 21021, м. Вінниця, Хмельницьке шосе, 95; т.(0432)43-90-02, azarov@lili.vstu.vinnica.ua.

Черняк Олександр Іванович, інженер кафедри обчислювальної техніки Вінницького національного технічного університету, 21021, м. Вінниця, Хмельницьке шосе, 95; т.(0432)43-90-02,, alexandr.chernyak@gmail.com.