УДК 681.3
©Л. Мороз[1],Т. Борецький[1], Т. Луковський[1],С. Войтусік[1],2015

МОДИФІКОВАНИЙ CORDIC-МЕТОД ОБЧИСЛЕННЯ СИНУСА-КОСИНУСА

Подано математичний опис методу обчислення тригонометричних синуса-косинуса за допомогою гібридного CORDICа із застосуванням принципу перекодування кута.
Ключові слова: обчислення синуса–косинуса, гібридний CORDIS–метод, перекодування коду кута.

The paper presents the mathematical description of the calculation method of trigonometric sine-cosine by the hybrid CORDIC method with application of the principle of angle recoding.
Keywords: calculation of sine-cosine, hybrid CORDIC method, angle recoding.

1. ВСТУП

У даній роботі викладено достатньо простий метод для обчислення синуса-косинуса, придатний для реалізації на апаратних платформах з обмеженими обчислювальними ресурсами – мікроконтролерах або ПЛІС.

2. МЕТА РОБОТИ

Метою роботи є подання математичного опису методу для обчислення синуса-косинуса за допомогою гібридного методу CORDIC із застосуванням принципу перекодування кута.

3. ОПИСВІДОМИХ АЛГОРИТМІВ

Відомо, що у загальному випадку обертання поколудеякого вектора, заданого координатами $\{x_{i-1}, y_{i-1}\}$,на кут мікрообертання α_iможна подати як:

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} cos\alpha_i & -sin\alpha_i \\ sin\alpha_i & cos\alpha_i \end{bmatrix} \begin{bmatrix} x_{i-1} \\ y_{i-1} \end{bmatrix} = \\ cos\alpha_i \begin{bmatrix} 1 & -tan(\sigma_i a_i) \\ tan(\sigma_i a_i) & 1 \end{bmatrix} \begin{bmatrix} x_{i-1} \\ y_{i-1} \end{bmatrix}, \quad (1)$$

де$\sigma_i = \{-1, +1\}$– вказує напрямок обертання вектора (у даному випадку – двостороннє обертання за або проти годинникової стрілки).

[1]Національний Університет "Львівська політехніка"

Існують два найпопулярніші методи апаратної реалізації такого обертання:

- метод Волдера (відомий як CORDIC- метод) [1,2];
- метод Мадісетті-Квентуса-Віллсона (назвемо його MKW-метод) [4,10].

У CORDIC - методі кут мікрообертання вибирається з умови $a_i = arctan(2^{-i})$, звідси$tan(\sigma_i a_i) = \sigma_i 2^{-i}$.Тоді рівняння (1) запишеться у вигляді

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = cos\left(arctan(2^{-i})\right)\begin{bmatrix} 1 & -\sigma_i 2^{-i} \\ \sigma_i 2^{-i} & 1 \end{bmatrix}\begin{bmatrix} x_{i-1} \\ y_{i-1} \end{bmatrix}. \qquad (2)$$

Як бачимо, тут множення на $tan(\sigma_i a_i)$ заміняється на просту операцію зсуву на i розрядів вправо, а σ_i визначає тип операції – віднімання чи додавання.

Рівняння традиційного (conventional) CORDICа мають вигляд:

$$\begin{gathered} x_i = x_{i-1} - \sigma_i 2^{-i} y_{i-1}; \\ y_i = y_{i-1} + \sigma_i 2^{-i} x_{i-1}; \\ z_i = z_{i-1} - \sigma_i \alpha_i; \\ \sigma_i = sign(z_{i-1}), i = 1 \ldots m. \end{gathered} \qquad (3)$$

Початкові умови у даному випадку є такими:

$$\varphi = \sum_{i=1}^{m} a_i 2^{-i}, a_i = \{0,1\}, \ \varphi \in 0,\pi/4],$$

вхідний кут, який подається (перекодовується) у вигляді:

$$\varphi \approx \sum_{i=1}^{m} \sigma_i arctan(2^{-i}).$$

а решта умов:

$$x_0 = P_c; y_0 = 0; z_0 = \varphi;$$

$$P_c = \prod_{i=1}^{m} cos\left(arctan(2^{-i})\right) = \prod_{i=1}^{m} \frac{1}{\sqrt{1+2^{-2i}}},$$

де P_c −масштабуючий коефіцієнт CORDIC – методу.

Основним вузьким місцем традиційного CORDIC – методу, яке не дозволяє виключити з ітераційних рівнянь дві змінні z_i та σ_i і таким чином спростити архітектуру пристрою, є послідовне обчислення σ_i, яке не піддається попередньому передбаченню.

Тепер розглянемо MKW-метод. Тут кут мікрообертання складає $\alpha_i = 2^{-i}$,$\mathrm{atan}(\sigma_i \alpha_i) = \tan(\sigma_i 2^{-i})$. Крім того, у даному методі вхідний кут φ перекодовується у кут φ_r

$$\varphi_r = \varphi_{const} + \varphi,$$

де

$$\varphi_{const} = 2^{-1} - 2^{-m-1}, \varphi_r = \sum_{i=2}^{m+1} b_i 2^{-i}; b_i = 2a_{i-1} - 1 \quad (5)$$

У цьому випадку $\sigma_i = b_i = \{+1, -1\}$,а ітераційні рівняння мають вигляд

$$x_i = x_{i-1} - b_i y_{i-1} \cdot tan(2^{-i}) \quad (6)$$

$$y_i = y_{i-1} - b_i x_{i-1} \cdot tan(2^{-i})$$

i=2,…, m+1.

Початкові умови є такими:

$$x_1 = P_m \cdot cos(\varphi_{const});$$

$$y_1 = P_m \cdot sin(\varphi_{const});$$

$P_m = \prod_{i=2}^{m+1} cos(2^{-i})$,(7)

P_m - масштабуючий коефіцієнт MKW-методу.

Тут враховано, що $\tan(\sigma_i 2^{-i}) = \sigma_i \tan(2^{-i})$. Як бачимо, з ітераційного процесу (6) вилучено обчислення z_i та σ_i (у порівнянні з формулами (3) традиційного CORDICa) – і у цьому основна перевага MKW – методу. Як і раніше $\sigma_i = b_i$ визначає тип операції – додавання чи віднімання. Однак, є серйозна проблема - множення на $tan(2^{-i})$, яке дуже складно реалізувати апаратно.

Для того, щоб усунути цей недолік на практиці застосовують два прийоми – використання переглядових таблиць (LUT – lookuptable) на ранніх стадіях ітераційного (6), коли $tan(2^{-i}) \neq 2^{-i}$ [4,10], і об'єднання двох методів – CORDIC та MKW в один з наступним розкладом кутів мікрообертань 2^{-i} на суми арктангенсів [5,6,7,9]. Розглянемо окремо кожен з цих прийомів.

MKW – метод з використанням LUT.

У роботах [4, 10] доведено, що для $i \geq \lfloor\frac{m}{3}\rfloor, tan(2^{-i}) \approx 2^{-i}$,тобто у цьому випадку справджується умова –

$\left|tan(2^{-i}) - 2^{-i}\right| < 2^{-m}$. А для $i < \lfloor\frac{m}{3}\rfloor$ результати обчислень на кожній з ітерацій об'єднуються і заміняються на значення синусів та косинусів, які зчитуються з LUT. Таким чином, ітераційні рівняння (6) наберуть такого вигляду

$$x_i = x_{i-1} - b_i y_{i-1} 2^{-i} (8)$$

$$y_i = y_{i-1} + b_i x_{i-1} 2^{-i}$$

$$i = \lfloor\frac{m}{3}\rfloor + 1, \dots m + 1; b_i = 2a_{i-1} - 1;$$

Початкові умови будуть такими:

$$x\left[\lfloor\frac{m}{3}\rfloor\right] = P_m \cdot cos(\varphi_{const} + \varphi_1);$$

$$y\left[\lfloor\frac{m}{3}\rfloor\right] = P_m \cdot sin(\varphi_{const} + \varphi_1);$$

$$\varphi_{const} = 2^{-\lfloor\frac{m}{3}\rfloor-1} - 2^{-m-1};$$
$$\varphi_1 = \sum_{i=1}^{\lfloor\frac{m}{3}\rfloor} a_i 2^{-i};$$

де P_m :
$$P_m = \prod_{i=\lfloor\frac{m}{3}\rfloor+1}^{m+1} cos(2^{-i}). \quad (9)$$

Основним недоліком такого підходу є великі об'єми пам'яті таблиць типу LUTпри великих значеннях m.

Об'єднання MKW та CORDIC - методів.

Як випливає з раніше викладеного матеріалу, основною перевагою MKW-методу є наперед визначене значення σ_i ітераційного процесу (6), що досягається з допомогою перекодування (5) вхідного кута φ. Якщо використати таке ж перекодування в методі CORDIC, то отримаємо наступну систему рекурентних рівнянь:

$$x_i = x_{i-1} - b_i y_{i-1} 2^{-i}. \quad (10)$$
$$y_i = y_{i-1} + b_i x_{i-1} 2^{-i}$$
$$i = 2, \dots, m+1; b_i = 2a_{i-1} - 1;$$

при початкових умовах:

$$x_1 = P_c \cdot cos(\varphi_{const});$$
$$y_1 = P_c \cdot sin(\varphi_{const}); \quad (11)$$
$$P_c = \prod_{i=2}^{m+1} \frac{1}{\sqrt{1+2^{-2i}}}.$$

Очевидно, що на кожній і-й ітерації кут α_i повороту (мікрообертання) вектора з координатами $\{x_{i-1}; y_{i-1}\}$буде складати $\alpha_i = arctan(2^{-i})$, у той час як згідно з (6) він повинен би дорівнювати 2^{-i}. Таким чином, на кожній ітерації буде відставання кута повороту на величину

$$d_i = b_i[2^{-i} - arctan(2^{-i})]. \quad (12)$$

У результаті проведення усіх m -ітерацій (і=2…m +1) сумарний кут відставання становитиме

$$\Delta = \sum_{i=2}^{m+1} d_i = \sum_{i=2}^{m+1} b_i[2^{-i} - arctan(2^{-i})] \quad (13)$$

Отже, якщо провести ітерації за формулами (10) з початковими умовами (11), то після завершення цих ітерацій необхідно додатково повернути вектор $\{x_{i+1}; y_{i+1}\}$ на кут Δдля того, щоб мати повний поворот початкового вектора на кут φ і таким чином отримати шукані значення $sin(\varphi)$та $cos(\varphi)$. Тут можуть бути використані два підходи для додаткових обертань:

1) розклад d_iна суму $arctan\left(2^{-i}\right)$ і проведення додаткових ітерацій (без використання пам'яті типу LUT). Саме на такому принципі працюють алгоритми [6]. Однак додаткові ітерації створюють додаткові затримки і вимагають додаткових апаратних затрат. Наприклад, для $m = 64$потрібно додатково провести 39 ітерацій.
2) спроби об'єднати деякі мікрообертання [7] хоч і підвищують швидкодію, однак, не дають суттєвого покращення результатів. Дещо інші підходи можна знайти у [5,9], але вони не міняють кардинально суті справи.

4. ОСНОВНІ РЕЗУЛЬТАТИ ДОСЛІДЖЕНЬ

Пропонований метод.

У даній роботі пропонується зменшення об'ємів переглядових таблиць при обчисленні функцій синуса та косинуса. Це досягається за рахунок застосування гібридного CORDIC–методу обчислень, описаного у [3, 8, 14], із використанням принципу перекодування кута, який детально розглянутий вище. Вхідний кут φ розбивається на три кути :

$$\varphi = \varphi_1 + \varphi_2 + \varphi_3, \tag{14}$$

причому

$$\varphi_1 = \sum_{i=1}^{m_{LUT}} a_i 2^{-i}, \tag{15}$$

$$\varphi_2 = \sum_{i=m_{LUT}+1}^{m_{CORDIC}} a_i 2^{-i}, \tag{16}$$

$$\varphi_3 = \sum_{i=m_{CORDIC}+1}^{m} a_i 2^{-i}, \tag{17}$$

Тут m_{LUT} – число старших бітів кута φ, що подаються на переглядову таблицю LUT, яка виконує функцію:

$$x_{m_{LUT}+1} = P_c \cdot cos(\varphi_1 + \varphi_{2const}), \tag{18}$$

$$y_{m_{LUT}+1} = P_c \cdot sin(\varphi_1 + \varphi_{2const}), \tag{19}$$

$$\varphi_{2const} = 2^{-(m_{LUT}+1)} - 2^{-(m_{CORDIC}+1)}, \tag{20}$$

m_{CORDIC} – число середніх бітів ($i_{CORDIC} = m_{LUT} + 2 \dots m_{CORDIC} + 1$), які обробляються за методом CORDIC (біти кута φ_2). Значення m_{CORDIC} вибирається з умови [3,12, 13]:

$$m_{CORDIC} = \left\lceil \frac{m}{2} \right\rceil. \tag{21}$$

Причому саме в CORDICу кут φ_2 перекодовується в кут φ_{2r}

$$\varphi_{2r} = \sum_{i=m_{LUT}+2}^{m_{CORDIC}} b_i 2^{-i} = \sum_{i=m_{LUT}+2}^{m_{CORDIC}} (2a_{i-1} - 1) \cdot 2^{-i}, \tag{22}$$

де φ_{2const}визначається за формулою (20) , так що

$$\varphi_2 = \varphi_{2r} - \varphi_{2const}. \tag{23}$$

Після завершення ітерацій за формулами

$$x_i = x_{i-1} - b_i y_{i-1} 2^{-i};$$
$$y_i = y_{i-1} + b_i x_{i-1} 2^{-i};$$
$$i = m_{LUT} + 2, \dots, m_{CORDIC} + 1.$$

отримаємо вектор з координатами $\{x_{m_{CORDIC}+1}; y_{m_{CORDIC}+1}\}$. Цей вектор здійснює додатковий поворот на залишковий кут$\varphi_{зал}$, який дорівнює скоригованому куту φ_3 на кут Δ:

$$\varphi_{зал} = \varphi_3 + \Delta, \tag{24}$$

де Δ визначається за формулою, подібною до формули (13):

$$\Delta = \sum_{m_{LUT}+2}^{m_3} d_i = \sum_{m_{LUT}+2}^{m_3} b_i [2^{-i} - arctan(2^{-i})]$$

Тут

$$m_3 = \lfloor \frac{m}{3} \rfloor - 1.$$

Далі виконуються дві дії, подібно до алгоритму, описаному у [8] – множення на кут $\varphi_{зал}$:

$$x_{m_{CORDIC}+2} = x_{m_{CORDIC}+1} - \varphi_{зал} \cdot y_{m_{CORDIC}+1}, \tag{25}$$
$$y_{m_{CORDIC}+2} = y_{m_{CORDIC}+1} + \varphi_{зал} \cdot y_{m_{CORDIC}+1}.$$

Однак існує обмеження на значення кута відставання Δ для того, щоб мати можливість виконати поворот (25) з відповідною точністю обчислень. Необхідною умовою виконання (25) є згідно з (1) набуття значень $\cos(\varphi_{зал}) \approx 1, \sin(\varphi_{зал}) \approx \varphi_{зал}$ з точністю до m бітів. Кут φ_3 відповідає цій умові, оскільки:

$$cos\varphi_3 \approx 1 - \frac{\varphi_3^2}{2};$$

$$sin\varphi_3 \approx \varphi_3 - \frac{\varphi_3^3}{6}.$$

Для $\varphi_3 \le 2^{-\frac{m}{2}}$складова $\frac{\varphi_3^2}{2}$у найгіршому випадку становитиме 2^{-m-1}, що буде вдвічі меншим за прийняту похибку обчислень в m двійкових розрядів. Таким чином, необхідно обмежити кут відставання Δ до цього ж значення. Розкладемо функцію arctan(2^{-i})ряд Тейлора з виразу (12)

$$d_i = b_i[2^{-i} - arctan(2^{-i})];$$
$$arctan(2^{-i}) \approx 2^{-i} - 2^{-3i}/3$$

і висунемо умову

$$\left|2^{-i} - (2^{-i} - 2^{-3i}/3)\right| \le 2^{\frac{-m}{2}}.$$

Звідси знайдемо, що

$$i = \lceil \frac{m - 2log_2(3)}{6} \rceil$$

Очевидно, що знайдене значення i визначає мінімальне значення $m_{LUT_{min}}$

$$m_{LUT_{min}} = i - 1,$$

яке і обмежить найбільше значення кута Δ.

В таблиці 1 наведені діапазони значень m_{LUT}для різних значень m.

Таблиця 1

Число старших бітів m_{LUT} вхідного кута φ, поданих на адресний вхід переглядової таблиці LUT

m	m_{LUT}-пропоноване	m_{LUT}[4,10]
16	2,3,4	4
24	3,4,5,6,7	7
32	4,5,6,7,8,9,10	10
48	5,6,7,8,9,10,11, 12,13,14,15	15
54	7,8,9,10,11,12,13,14,15,16,17	17
64	9,10,11,12,13,14,15,16,17,18,19,20	20

Як випливає з поданої таблиці 1 запропонований метод значно розширює межі значень m_{LUT}у порівнянні з відомими результатами робіт [4,10].

5. ВИСНОВКИ

Описаний метод гібридного CORDICa із використанням принципу перекодування кута дає можливість розробнику гнучко вибирати оптимальні об’єм пам’яті LUT та число ітерацій CORDICa для побудови апаратного обчислювача функцій синуса та косинуса.

1.J. E. Volder, "The CORDIC Trigonometric Computing Technique," IEEE Transactions on Electronic Computers, vol. EC-8, no. 3, pp. 330–334, Sep. 1959. 2.J.S. Walther, "A unified algorithm for elementary functions", in Proc. AFIPS Conf., vol. 38, 1971, pp. 385-389. 3.Л.В.Мороз, Я.І.Грабовський, Т.М.Микитів, Т.Р.Борецький, Ю.М.Костів, С.С.Войтусік. Швидкодіючий гібридний CORDIC-обчислювач тригонометричних функцій. Науковий вісник НЛТУ України. – 2014. – Вип. 24.8, с.352-357. 4.A. Madisetti, A.Y.Kwentus, A.N.Willson. A 100 MHz, 16-b, direct digital frequency synthesier with 100-dBc supurious-free dynamic range. IEEE Journal of Solid-State Circuits. – vol.34, – № 8, – 1999, – pp. 1034 – 1043. 5.M. Kuhlmann and K. K. Parhi, "P-CORDIC : A Precomputation Based Rotation," EURASIP Journal on Applied Signal Processing, vol. 2002, no. 1, pp. 936–943, 2002. Available:http://dx.doi.org/10.1155/S1110865702205028. 6. T.-B. Juang, S.-F. Hsiao, and M.-Y. Tsai, "Para-CORDIC: Parallel CORDIC Rotation Algorithm," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 51, no. 8, pp. 1515–1524, Aug. 2004. 7. T. Juang, "Low Latency Angle Recoding Methods for the Higher Bit-Width Parallel CORDIC Rotator Implementations," IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 55, no. 11, pp. 1139–1143, Nov. 2008. 8.Leonid Moroz, Shinobu Nagayama, Taras Mykytiv, Ihor Kirenko, Taras Boretskyy / "Simple Hybrid Scaling-Free CORDIC Solution for FPGAs " // International Journal of Reconfigurable Computing, vol. 2014, Article ID 615472, 4 pages, 2014. http://dx.doi.org/10.1155/2014/615472. 9.Francisco AGUIRRE-RAMOS, Alicia MORALES-REYES,Rene CUMPLIDO,Claudia FEREGRINO-URIBE. An Area Efficient Composed CORDIC Architecture. Advances in Electrical and Computer Engineering,Volume 14, Number 2, 2014, pp. 113–116. 10. Shoab, A. K. Digital design of signal processing systems: A practical approach (1st ed). New York, NY: John Wiley, 2011. 11.P. K. Meher, J. Valls, T.-B. Juang, K. Sridharan, andK. Maharatna. 50 years of cordic: Algorithms, architectures, and applications. IEEE Transactions on Circuits and Systems I : Regular papers,56(9):1893–1907, Sept. 2009. 12.D. Timmermann, H. Hahn, and B. Hostika. Modified CORDIC algorithm with reduced iterations.Electronics Letters, 25(15):950–951, 1989. 13. F. de Dinechin, M. Istoan, and G. Sergent, "Fixed-point trigonometric functions on FPGAs," in Highly-Efficient Accelerators and Reconfigurable Technologies, Mar. 2013. 14. Л.В.Мороз. Теорія та швидкодіючі апаратно-програмні засоби ітераційних методів обчислення функцій. Автореф. дис. д.т.н.,Львів,2013.