

미세회전 예측 및 Table-Lookup을 이용한 CORDIC 방식 고속 삼각함수 연산기

조용권, 이문기
연세대학교 전기전자공학 VLSI & CAD 연구실
전화: 02-2123-4731

High Speed CORDIC Architecture with Pre-computed the Direction of Micro-rotation and Table-Lookup

Yongkwon Cho, Moonkey Lee
VLSI & CAD Laboratory Department of Electrics and Electronics
Yonsei University
E-mail : ykcho@yonsei.ac.kr

Abstract

The CORDIC algorithm can be implemented very simple H/W, but needs a lot of latency to compute trigonometric function. The RA(Redundant Arithmetic) resolves this problem, but also has difficulty to determine the directions of micro-rotations. The pre-computed direction of micro-rotation algorithm relieves the RA of this matter. In this paper, we proposed the modified the pre-computed algorithm adopted with a table-lookup. Instead of reducing H/W complexity, its performance and calculation errors are improved.

I. 서론

삼각함수의 값을 계산하기 위해서는 크게 Table-Lookup을 이용하는 방식과 CORDIC 알고리즘을 위한 방식이 사용된다[1][2]. Table-Lookup 방식은 고속 연산이 가능하지만 H/W가 복잡하고 CORDIC 방식은 덧셈과 쉬프트 연산으로만 구성된 간단한 H/W 갖지만 반복연산에 의한 처리속도 지연 문제가 발생한다.

CORDIC의 문제점을 해결하기 위해서 다양한 방법

들이 연구되어 왔다. 그 중 RA(Redundant Arithmetic)를 이용해 덧셈의 캐리(Carry) 전달 지연을 해결하는 방안이 제안되었다[3][4]. 그러나, RA는 수의 표현 체계상 부호 값 결정이 어렵다는 문제점을 가지고 있는데, 미세회전 처리 후 나머지 각의 부호 값으로 다음 미세회전의 방향을 결정하는 CORDIC 알고리즘의 특징 때문에 RA를 적용하기가 어려워진다.

RA의 단점을 해결하기 위한 방안으로는 회전 방향을 추측하고 보정하는 알고리즘[4], DCORDIC(Differential CORDIC) 알고리즘[5], 미세회전의 방향을 미리 계산하는 알고리즘[6] 등이 제안되었다.

II. 미세회전 방향 계산

CORDIC 알고리즘의 기본적인 공식은 식 (1)과 같다.

$$\begin{aligned} x_{i+1} &= x_i - \sigma_i 2^{-i} y_i \\ y_{i+1} &= y_i + \sigma_i 2^{-i} x_i \\ z_{i+1} &= z_i - \sigma_i \tan^{-1}(2^{-i}) \end{aligned} \quad (1)$$

여기서 σ_i 각 미세회전의 회전 방향을 나타내는데 $\sigma = 0, \dots, 0, -1$ 로 정의하면, 삼각함수 각도 θ 와 σ 사의 관계는 부분적인 선형특성을 가지며, 수식(2)로 표현된다.

$$\sigma = \theta + \epsilon_k$$

수식 (2)에서 ϵ_k 는 k번째 불연속점에서의 값의 차를

보정해주기 위한 상수 값이며, 그림 1은 θ 와 σ 사이의 불연속 특성을 보여준다.

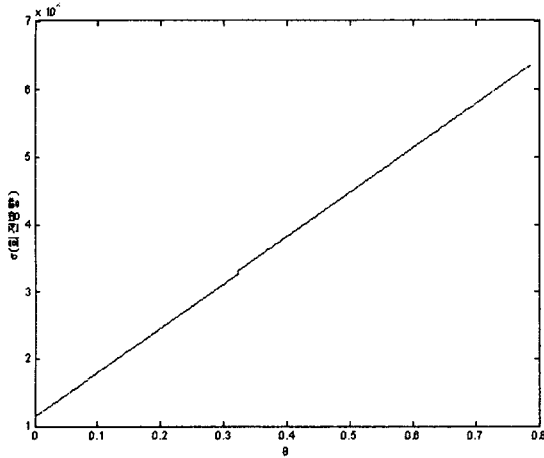


그림 1. 입력 각 θ 와 미세회전 σ 의 부분선형 관계 그래프

기본적인 CORDIC 알고리즘에서는 Z_{i-1} 값을 연산하기 위한 \tan^{-1} Table을 사용하지만 미세회전의 방향을 계산하는 알고리즘은 식(2)의 불연속점 값의 차를 보정해주기 위한 상수 Table을 사용한다. 16비트 삼각함수 연산의 Table의 값은 [6]에 예제로 나타나있다.

그림 2는 미세회전 방향 계산 알고리즘을 이용한 RA CORDIC 하드웨어를 보여준다.

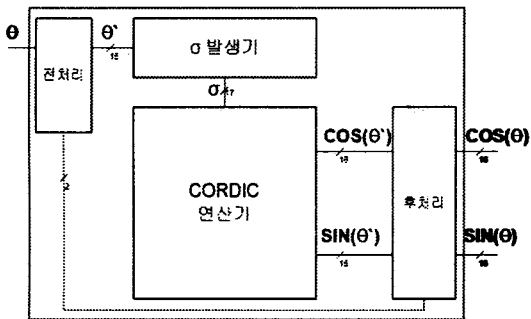


그림 2. 미세회전 방향 계산 CORDIC 구조

전처리기와 후처리기는 σ 발생기의 Table크기를 줄여주기 위해 θ 값을 $0 \sim \pi/4$ 의 영역으로 변환해주고, 결과값을 원래의 영역값으로 보정해주는 역할을 한다. CORDIC 연산기는 반복연산처리 속도를 향상시키기 위해서 RA를 이용한 비중첩(Unfolded) 구조가 사용하였다. 그림 3은 CORDIC 연산기의 구조를 보여준다.

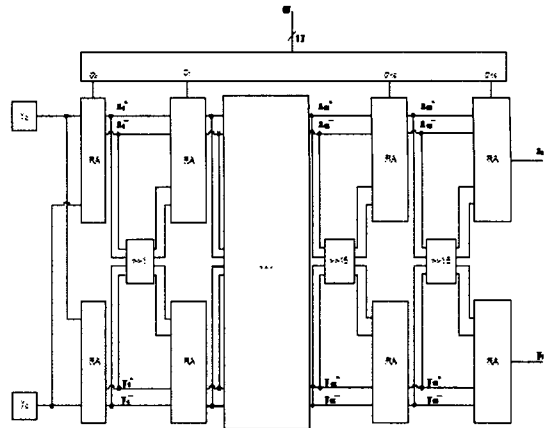


그림 3. RA를 이용한 CORDIC 연산기

본 논문에서는 C(Coarse)-angle을 구하기 위한 Table-Lookup 부분과 F(Fine)-angle을 구하기 위한 부분으로 나누어진 RA CORDIC 연산기 구조를 제안한다.

III. Table-Lookup CORDIC 연산기

본 논문에서는 16 비트의 삼각함수 값을 계산하기 위한 CORDIC 연산기를 설계하였다. 그림 4는 각 RA 미세회전 연산단의 비트수에 따른 CORDIC 연산기의 성능을 보여준다.

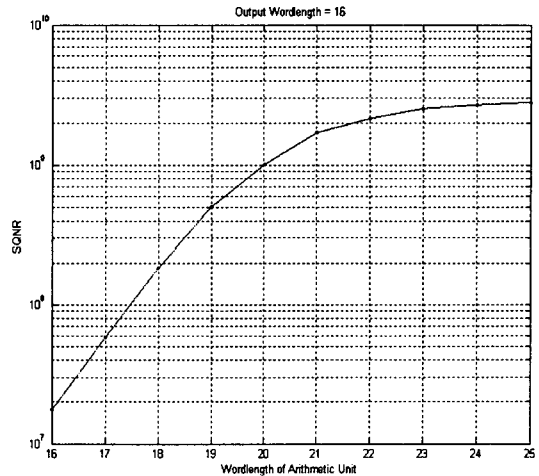


그림 4. RA 미세회전 연산단 비트수에 따른 성능

SQNR(Signal-to-Quantization Noise Ratio)는 삼각함수 연산기의 성능을 나타내는 지표로서 실제 삼각함수 파워값과 연산기 오차의 파워값의 비율로 나타낸다

[7]. 제안하는 시스템은 16비트의 삼각함수를 계산하기 위해 각각의 미세회전 연산단에 22비트 RA를 사용하였다.

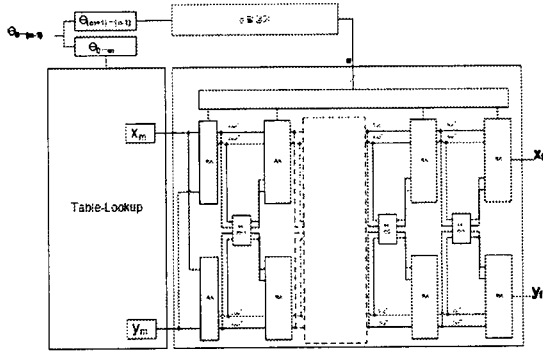


그림 5. 제안하는 Table-Lookup과 CORDIC 연산기의 구조

제안하는 구조는 기존의 CORDIC 연산기를 이용해 모든 미세회전 연산을 처리하던 방식과 다르게 C-angle 부분은 Table-Lookup으로 처리하고 F-angle 부분만 CORDIC 연산기로 처리한다. F-angle 부분 미세회전 연산의 단수 감소에 따른 하드웨어 크기 감소폭이 Table-Lookup 추가에 의한 하드웨어 증가폭보다 크기 때문에 제안하는 구조는 기존의 방식보다 하드웨어 크기가 줄어드는 장점을 가진다. 그림 6은 기존의 CORDIC 구조와 제안하는 구조의 하드웨어 크기를 비교한 것이다. 각각의 Table-Lookup과 RA는 0.25μm CMOS Standard Cell 라이브러리를 이용해 합성하여 게이트 수를 계산하였다[8]. Table-Lookup의 크기를 결정하는 입력비트 수는 C-angle의 비트수와 동일하다.

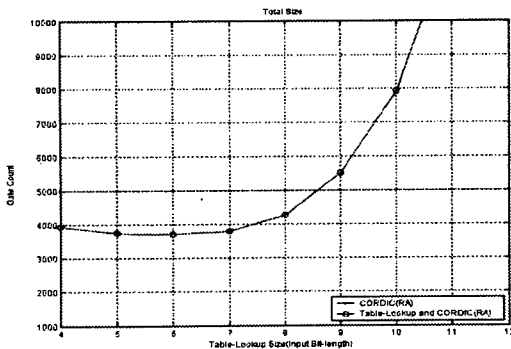


그림 6. 기존 CORDIC 연산기와 제안하는 연산기의 하드웨어 크기 비교

그림 6에서 Table-lookup 입력 비트수가 6보다 적을 경우, 전체 하드웨어 크기는 비트수가 증가할 수록 감소하고, 7보다 큰 경우는 증가하는 것을 볼 수 있다. 입력 비트수가 8보다 적은 경우까지는 여전히 기존의 CORDIC 연산기 보다는 전체 하드웨어 크기가 작다는 것을 알 수 있다. 이러한 결과는 Table-Lookup의 경우 C-angle의 비트수가 n인 경우 2^n으로 증가하고 CORDIC 연산기 부분은 C-angle 비트수 n에 따라 선형으로 감소하기 때문에 발생한다.

제안하는 구조의 CORDIC 연산기에서는 미세회전 연산단의 수가 줄었지만, F-angle가 여전히 수렴조건을 만족하기 때문에 나머지 미세회전 연산을 처리할 수 있다[9]. 수식 (3)은 CORDIC의 수렴조건을 보여준다.

$$\theta_i \leq \theta_{n-1} + \sum_{j=i+1}^{n-1} \theta_j \quad (3)$$

그림 7은 기존 CORDIC 연산기와 제안하는 연산기의 처리속도를 비교한 것이다.

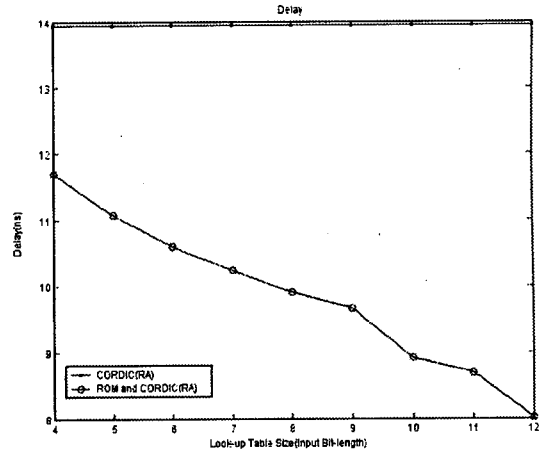


그림 7. 기존 CORDIC 연산기와 제안하는 연산기의 처리속도 비교

그림 7을 보면 Table-Lookup을 이용해 처리하는 C-angle의 비트 수가 증가할수록 전체 연산기의 처리속도 지연이 감소한다는 것을 알 수 있다.

그림 6과 그림 7의 결과에 따라 C-angle의 비트수를 8이나 9비트로 하면 기존 CORDIC 방식과 하드웨어 크기가 비슷하면서 전체 연산 처리속도 지연이 감소하는 구조를 얻을 수 있다.

그림 8은 16비트 삼각함수 값과 기존 CORDIC, 제안하는 연산기의 SQNR을 비교한 결과이다.

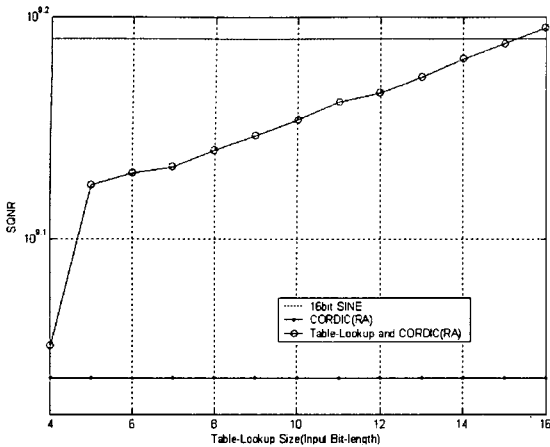


그림 8. SQNR 비교

Table-Lookup을 이용하면 C-angle구간 미세회전 연산의 오차를 줄일 수 있기 때문에, C-angle 비트가 증가할수록 제안하는 방식의 연산기 SQNR 값이 좋아지는 것을 알 수 있다.

IV. 결론

CORDIC 알고리즘을 이용해 삼각함수를 계산하는 방식중 RA구조 CORDIC 연산기는 반복연산에 의한 처리속도 지연 문제를 해결 할 수 있다. 본 논문에서는 16비트 삼각함수 값을 구하기 위해 RA 구조의 CORDIC 방식 중 미세회전 방향 계산 알고리즘을 사용하였다. 이 때, 제안하는 구조는 C-angle영역 미세회전을 처리할 때 Table-Lookup을 이용하였다. 그 결과 C-angle을 8~9비트로 할 경우 기존 RA 방식 CORDIC과 비슷한 하드웨어 복잡도를 갖지만, 처리속도 지연이 감소하고 SQNR이 향상되는 결과를 얻을 수 있다.

참고문헌

[1] P. T. P Tang, "Table-lookup algorithms for elementary functions and their error analysis," 10th IEEE Symposium on Computer Arithmetic, pp. 232-236, Jun., 1991.
 [2] J. E. Volder, "The CORDIC trigonometric computing technique," IRE Trans. Electron. Computers, vol. C-8, pp. 330-334, Sept. 1959.
 [3] H. X. Lin and H. J. Sips, "On-line CORDIC algorithms," IEEE Trans. On Computers, Vol. 39, Iss. 8, pp. 1038-1052, Aug. 1990.
 [4] N. Takagi, T. Asada, and S. Yajima, "Redundant CORDIC Methods with a constant Scale Factor for Sine and Cosine Computation," IEEE Trans. On

Computers, vol. 40, no. 9, 1991.

[5] H. Dawid and H. Meyr, "The Differential CORDIC Algorithm: Constant Scale Factor Redundant Implementation without Correcting Iterations," IEEE Transactions on Computers, vol. 45, no. 3, 1996.

[6] M. Kuhlmann and K.K. Parhi, "A High-Speed CORDIC Algorithm and Architecture for DPS Applications," in Proc. of the 1999 IEEE Workshop on Signal Processing Systems (SiPS'99), Taipei, Taiwan, Oct. 1999.

[7] C. S. Wu, A. Y. Wu, "Modified vector rotational CORDIC (MVR-CORDIC) algorithm and architecture", IEEE Trans. on Circuits and Systems Vol. 48, Iss. 6, pp. 548-511, June 2001.

[8] Samsung Electronics Co. Ltd., MDL110 0.25μm 2.5V CMOS Standard Cell Library DATABOOK, 1999

[9] X. Hu, R. G. Harber, S. C. Bass, "Expanding the range of convergence of the CORDIC algorithm," IEEE Trans. on Computers, Vol. 40, Iss. 1, pp. 13-21, Jan. 1991