DSP in VLSI

Homework 4

Coordinate Rotation Digital Computer (CORDIC)

I. Purpose

In this homework, we will learn the principle of CORDIC operation to implement the conversion from Cartesian coordinate to polar coordinate. In other words, we can obtain the phase and magnitude of a complex value.

II. Principle of CORDIC

In digital signal processing or communication algorithms, we often need the phase or magnitude of a complex value. Given $Z = X + jY$, define

$$\measuredangle Z = \tan^{-1}\left(\frac{Y}{X}\right) \tag{1}$$

and

$$|Z| = \sqrt{X^2 + Y^2}. \tag{2}$$

Because the tangent and arctangent functions have $\pi/2$ symmetry as shown in Fig. 1, in hardware implementation, we usually consider the case in the first quadrant and then extend the result to the remaining quadrants according to the sign values of X and Y.

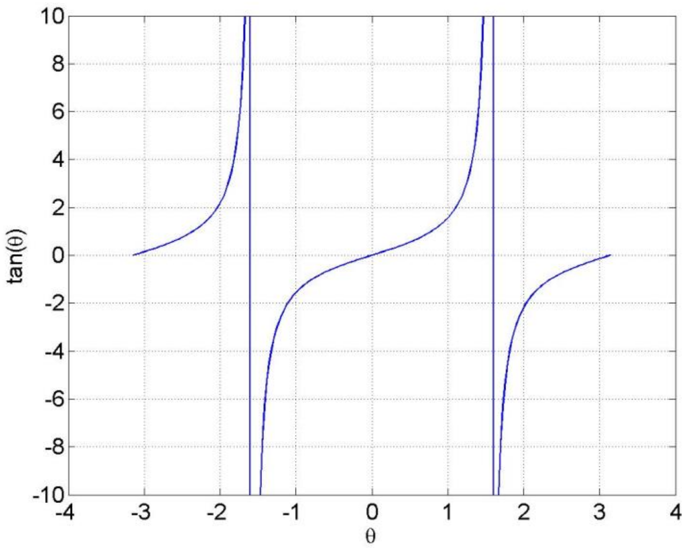


Fig. 1 The tangent function.

The CORDIC (COordinate Rotation DIgital Computer) operation is a good mean to convert values between the Cartesian coordinate and the polar coordinate. It is also widely used in QR decomposition, matrix triangularization, singular value decomposition, and Eigen-value decomposition.

The basic concept of CORDIC is to partition one rotation angle into the sum of several elementary angles. These elementary angles, denoted by $\theta_e(i)$ for $i = 0,1,...,$ are special angles, which can be accomplished by shift-and-add,

$$\theta = \sum_{i=0}^{N-1} \mu_i \theta_e(i), \tag{3}$$

$$\theta_e(i) = tan^{-1}(\frac{1}{2^i}) \tag{4}$$

where $\mu_i \in \{+1, -1\}$, and it determines the counter-clockwise rotation or clockwise rotation. At the ith micro-rotation step, vector $(X(i), Y(i))$ is converted to be

$$\begin{bmatrix} X(i+1) \\ Y(i+1) \end{bmatrix} = \begin{bmatrix} 1 & -\mu_i 2^{-i} \\ \mu_i 2^{-i} & 1 \end{bmatrix} \begin{bmatrix} X(i) \\ Y(i) \end{bmatrix}. \tag{5}$$

A. Phase in polar coordinate

If $Z = X + jY$ and the phase of Z is desired, we only need to rotate vector (X, Y) back to the x-axis and calculate the summation of all these angles, as shown in Fig. 2.

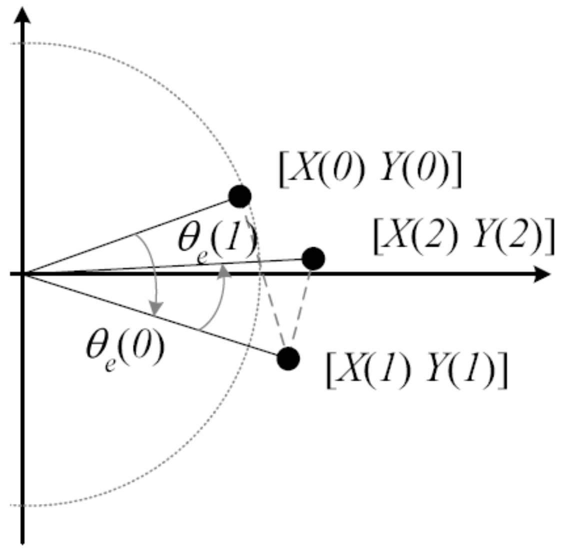


Fig. 2 CORDIC operation.

The procedure can be summarized as the following steps. Assume that $X > 0$.

1. Initialization

   $X(0) = X,\ Y(0) = Y,\ \hat{\theta}(0) = 0$

2. Determine direction

   $$\mu_i = -sgn(Y(i)). \tag{6}$$

3. Perform micro-rotation

$$\begin{bmatrix} X(i+1) \\ Y(i+1) \end{bmatrix} = \begin{bmatrix} 1 & -\mu_i 2^{-i} \\ \mu_i 2^{-i} & 1 \end{bmatrix} \begin{bmatrix} X(i) \\ Y(i) \end{bmatrix} \circ \tag{7}$$

4. Phase accumulation

$$\hat{\theta}(i+1) = \hat{\theta}(i) - \mu_i \tan^{-1}(2^{-i}) = \hat{\theta}(i) - \mu_i \theta_e(i) \tag{8}$$

5. Repeat step 2 to step 4 until $Y(i) \approx 0$.

In each micro-rotation steps, the length of vector $(X(i), Y(i))$ is changed [3]. However, if only the phase of vector (X, Y) is desired, it is not necessary to scale down the length of vector $(X(i+1), Y(i+1))$. The hardware architecture of the micro-rotation and phase accumulation is shown in Fig. 3.

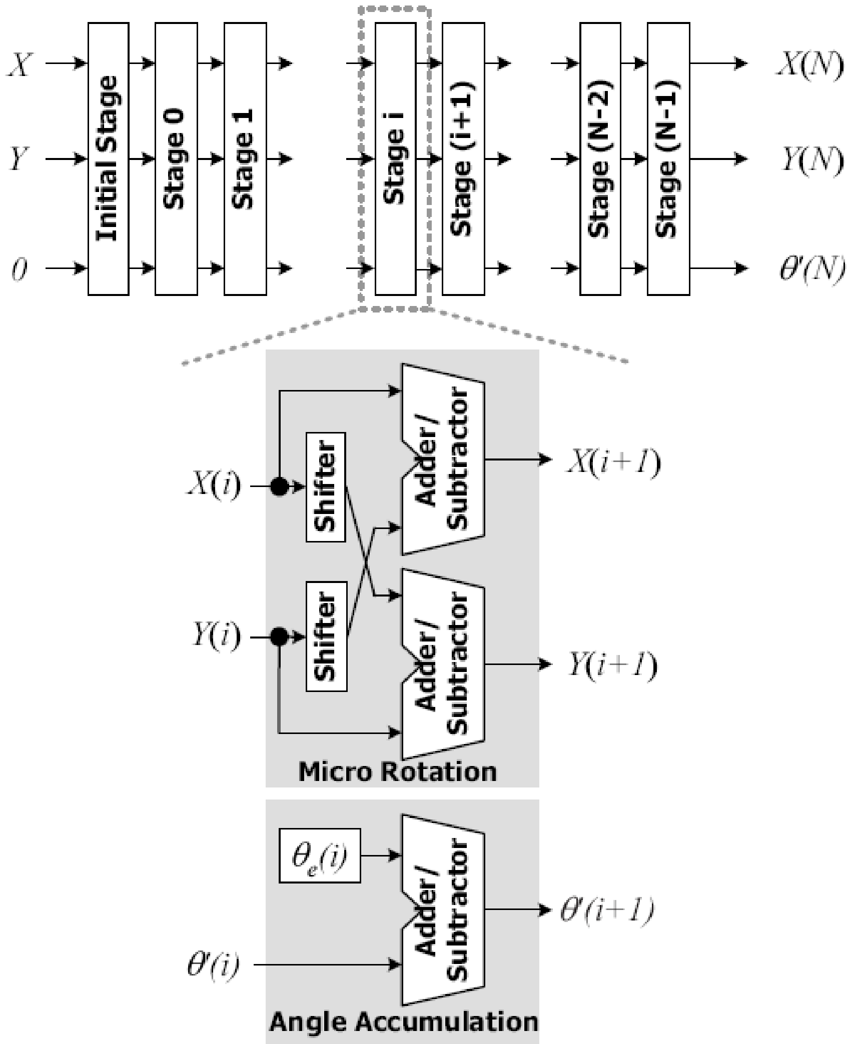


Fig. 3 Hardware architecture of CORDIC.

The convergence range of CORDIC can be computed as $\pm\sum_{i=0}^{N-1}\theta_e(i)$, which is $\pm 99^{\circ}$ approximately. Hence, we need to convert the vector in the second and the third quadrants into the first and the fourth quadrant. The initial stage in Fig. 3 deals with this operation. Subsequently, each micro-rotation is processed by each stage.

B. Magnitude in polar coordinate

From Fig. 2, if vector (X, Y) is rotated back to the x-axis with unchanged magnitude, the magnitude of vector (X, Y) is equal to the horizontal component along the x-axis. Observing Eq. (7), we can see that the vector length is increased by $\sqrt{1+2^{-2i}}$. Hence if $Y(N)=0$ after micro-rotations for N times, then

$$X(N)=\sqrt{X^2+Y^2}\prod_{i=0}^{N-1}\sqrt{1+2^{-2i}}. \quad (9)$$

As a result,

$$\sqrt{X^2+Y^2}=\frac{X(N)}{\prod_{i=0}^{N-1}\sqrt{1+2^{-2i}}}=S(N)X(N) \quad (10)$$

Usually, the number of micro-rotations N is determined according to the error tolerance. Consequently, $S(N)$ is a constant and can be represented by shift-and-add operation. The divider is not necessary for the scaling stage.

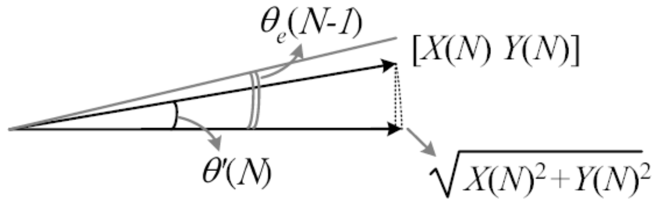


Fig. 4 The approximation error of magnitude function by using finite micro-rotations in CORDIC operation.

The error of using finite micro-rotations to approximate the magnitude function can be analyzed in the following.

$$\frac{|X+jY|-S(N)X(N)}{|X+jY|}=\frac{S(N)\sqrt{X(N)^2+Y(N)^2}-S(N)X(N)}{S(N)\sqrt{X(N)^2+Y(N)^2}}$$

$$=1-\frac{X(N)}{\sqrt{X(N)^2+Y(N)^2}}=1-\cos\left(\theta'(N)\right) \quad (11)$$

where the relationship for $X(N)$, $Y(N)$ and $\phi(N)$ is shown in Fig. 4. The residual phase $\phi(N)$ must be less than the rotation angle at the previous step, namely $\theta'(N)<\theta_e(N-1)=\tan^{-1}(\frac{1}{2^{N-1}})$. Therefore,

$$\frac{|X+jY|-S(N)X(N)}{|X+jY|}<1-\cos\bigl(\theta_e(N-1)\bigr)=1-\frac{1}{\sqrt{1+2^{-2(N-1)}}}. \quad (12)$$

From Eq. (12), we can derive the number of micro-rotations N if the CORDIC operation is adopted to approximate the magnitude function.

III. Design Procedures

1. Realize CORDIC properties

Although we do not perform vector scaling when using CORDIC to obtain the phase of a complex number, we still need to know the increase in magnitude after infinite micro-rotations for reserving sufficient dynamic range during implementation. Find out the scaling factor $S(N)$ for ≥ 23.

2. Determine the word-length of fixed-point representation

Assume that $X = \cos(\alpha_n)$, $Y = \sin(\alpha_n)$, where $\alpha_n = \frac{(4n+\beta)}{23}\pi$ for $n = 0,1,\ldots,10$ and $\beta = mod(I,3) + 1$, where I is the last digit in your student ID. Both X and Y are quantized into $(\mathrm{w}+2)$ bits including the sign bit and the w-bit fractional part. According to Q1, determine the integer word-length and the fractional wordlength w of $X(i)$ and $Y(i)$ at all the stages if they use the same format so that the average absolute phase error is less than 2^{-11} (Hint: Consider the possible growth of the input signal.)

3. Determine the number of micro rotations for arctangent function

Determine the number of required micro rotations and also determine the related elementary angles in fixed-point representation with proper word-length so that the average phase error of $\tan^{-1}(\frac{Y}{X})$ obtained by the CORDIC operation will be less than 2^{-11} radian.

4. Determine the number of micro rotations for magnitude function

Given 0.1% error tolerance of the magnitude function approximated by the CORDIC operation, determine the number of the required micro-rotations.

5. Determine the CSD of the scaling factor

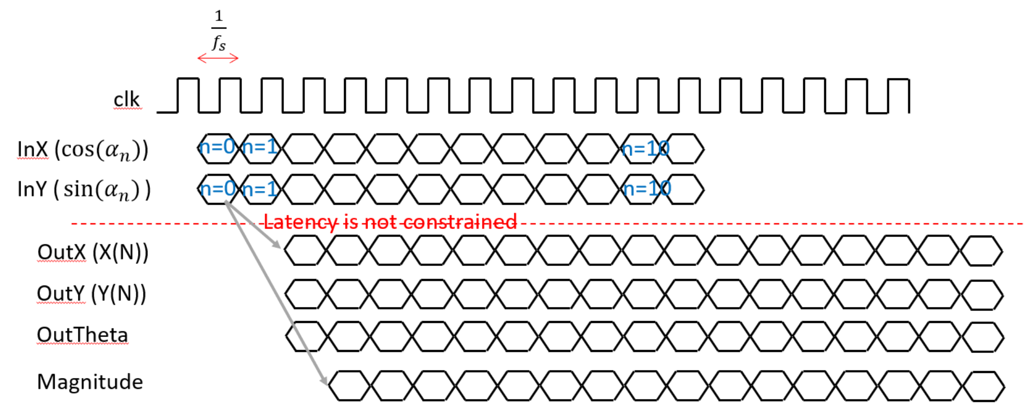
Use CSD to design the shift-and-add operation for the scaling factor $S(N)$ so that the magnitude function has an error less than 0.1%.

6. Pipeline architecture design

Design the complete architecture of the CORDIC block for the arctangent function $\measuredangle Z = \tan^{-1}(\frac{Y}{X})$ including the initial stage so that we can obtain the correct results when the 11 input sets distributed in four quandrants enter. Note that you need to use the number of micro-rotations that you have derived in Step 2 and Step 3.

7. Check operating frequency

Synthesize your design and insert pipeline registers to ensure the operating frequency (f_s) is higher than 75MHz for FPGA design or 130MHz for CMOS design. The timing diagram should be given as follows



8. Pipeline architecture design

Design the complete architecture of the CORDIC block for the magnitude function $|Z| = \sqrt{X^2 + Y^2}$ including the initial stage and the scaling stage so that we can obtain the correct results when 11 input sets enter. Note that you need to use the number of micro-rotations that you have derived in Step 4 and Step 5.

IV. Required Results

1. (Step 1) Please show how you calculate the scaling factor, write down the N value that you use and the result of $S(N)$. (5%)
2. (Step 2) Draw the figure of average absolute error versus fractional wordlength (10%) to show how you determine the setting of word-length of the fractional part. Write down the integer word-length of $X(i)$ and $Y(i)$ that you use all the stages. Please explain it. (5%)
3. (Step 3) Please draw a figure to denote the average phase errors of 11 **quantized** input pairs (X, Y) versus different numbers of micro-rotations N (10%) and draw a figure to show the resulted phase errors of 11 **quantized** input pairs versus the word-length of quantized elementary angles (10%). Explain how you determine it. Also list a table of the elementary angles (both in floating-point representation and binary fixed-point representation).(5%)
4. (Step 4) Please show how you decide the number of micro-rotations for the magnitude function with error tolerance of 0.1%. (10%)
5. (Step 5) Write a program to show the setting of fractional wordlength of CSD versus error. Draw the figure. (10%). Depict your design for the shift-and-add block according to your CSD representation. How many adders do you use? (10%)
6. (Step 6) Depict your design of the initial stage and the complete CORDIC

architecture for the arctangent function. Mark the word-length in the block diagram. (10%)

7. (Step 6) Implement your design with only DFFs inserted at the inputs and outputs. Show the timing diagram of behavior simulation. (20%) Compare the results with the arctangent function and draw the error versus index n to show that your implementation meets the precision requirements of error less than 2^{-11}. (10%)
8. (Step 6) Draw the critical path in your block diagram and synthesize your circuits to show the critical path and max delay (or operating frequency). (10%)
9. (Step 7) Calculate how to insert sufficient pipeline register to meet the requirement of operating frequency. (5%). Insert the pipeline registers in your design. Synthesize your implementation again to obtain the timing slack with the timing constraint of $(1/f_s)$ and show that the slack is positive. (5%) Check the critical path report from the synthesizer to show that your pipeline insertion is effective. (5%). Show the timing diagram of post-synthesis simulation with correct setting of clock period. (20%) Verified the error between the post-synthesis results and the floating-point arctangent results. Draw the error versus index n. (10%)
10. (Step 8) Based on the design in Step 7, change it to calculate the magnitude function. Show the timing diagram of behavior simulation and post-synthesis simulation with correct setting of clock period $(1/f_s)$. (20%). Verified the error between the post-synthesis results and the floating-point arctangent results. Draw the error versus index n. (10%)
11. Please upload your report containing the required results and your Verilog codes as well as Matlab/Python codes to NTUCool.

V. Reference

[1] R. Lyons, “Another contender in the arctangent race," IEEE Signal Processing Magazine, pp. 109-110, Jan. 2004.

[2] Y. H. Hu, “CORDIC-based VLSI architectures for digital signal processing," IEEE Signal Processing Magazine pp. 16-35, July 1992.

[3] Y. H. Hu, “The quantization effects of the CORDIC algorithm," IEEE Trans. Signal Processing, Vol. 40, No. 4, pp. 834-844, Apr. 1992.