CORDIC-UART

Artix-7 FPGA Implementation

By Grant Yu

Contents

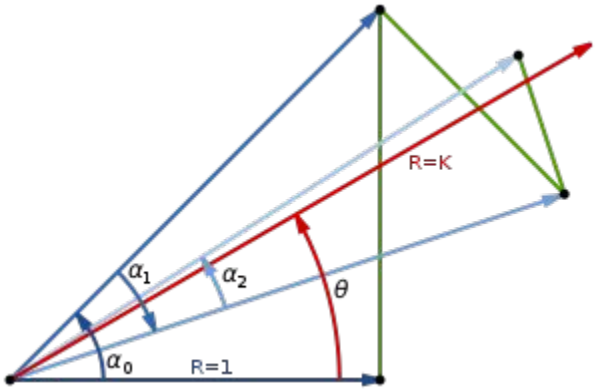
- Overview of the CORDIC algorithm
- Calculation of Sine and Cosine using CORDIC
- Top-level FPGA block diagram
- Design, Verification, and Implementation
- Demonstration using custom client PC program

CORDIC Overview

- CORDIC is a collection of iterative algorithms using shifts and adds in lieu of multiplications in order to compute a number of vector rotations

- At each step, rotate vector by a predetermined angle so as to eventually achieve a target angle

- Larger # of steps yields a better precision, but at the cost of higher area or latency in HW



Source: https://en.wikibooks.org/wiki/Digital_Circuits/CORDIC

CORDIC Cosine and Sine Calculation

- Applying the rotation transformation matrix on the unit *i*-direction vector results in the cosine and sine of the rotation angle

$$x' = xcos\theta - ysin\theta$$
$$y' = ycos\theta + xsin\theta$$

$$x' = cos\theta(x - ytan\theta)$$
$$y' = cos\theta(y + xtan\theta)$$

- How can we eliminate all multiplications?

CORDIC Cosine and Sine Calculation (cont.)

- We remove the multiplication with the tangent terms
- Now onto the cosine terms...

$$x_{n+1} = cos\theta(x_n - y_n tan\theta)$$
$$y_{n+1} = cos\theta(y_n + x_n tan\theta)$$

$$\theta = tan^{-1}(2^{-n})$$

$$x_{n+1} = cos\theta(x_n - y_n(2^{-n}))$$
$$y_{n+1} = cos\theta(y_n + x_n(2^{-n}))$$

CORDIC Cosine and Sine Calculation (cont.)

- The cosine term over a fixed # of steps is predetermined
- Therefore, the magnitude of the initial vector can be scaled down to compensate for the gain
- Keep track of angle to determine if a rotation should be clockwise/counter

$$x_0 = \frac{1}{cos(\theta_0) * cos(\theta_1) * \ldots * cos(\theta_n)} \approx 0.60725\ldots$$

$$y_0 = 0$$

$$\longrightarrow \quad \begin{aligned} x_{n+1} &= x_n - y_n(2^{-n}) \\ y_{n+1} &= y_n + x_n(2^{-n}) \end{aligned}$$

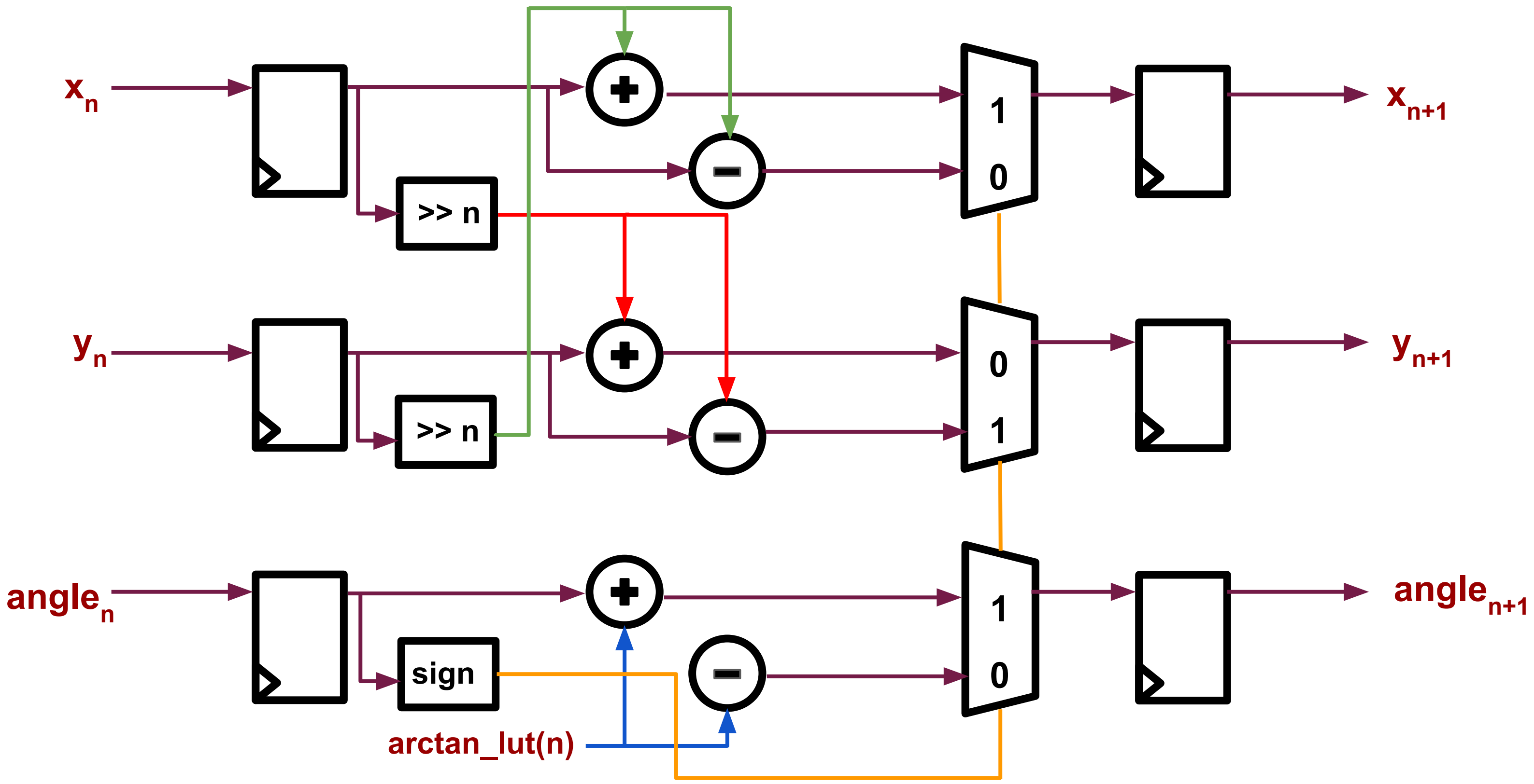
$$angle_0 = angle_{in} \quad \longrightarrow \quad \begin{aligned} x_{n+1} &= x_n - y_n(2^{-n})(sign(angle_n)) \\ y_{n+1} &= y_n + x_n(2^{-n})(sign(angle_n)) \\ angle_{n+1} &= angle_n - \theta_n(sign(angle_n)) \end{aligned}$$

CORDIC Cosine and Sine - Other Considerations

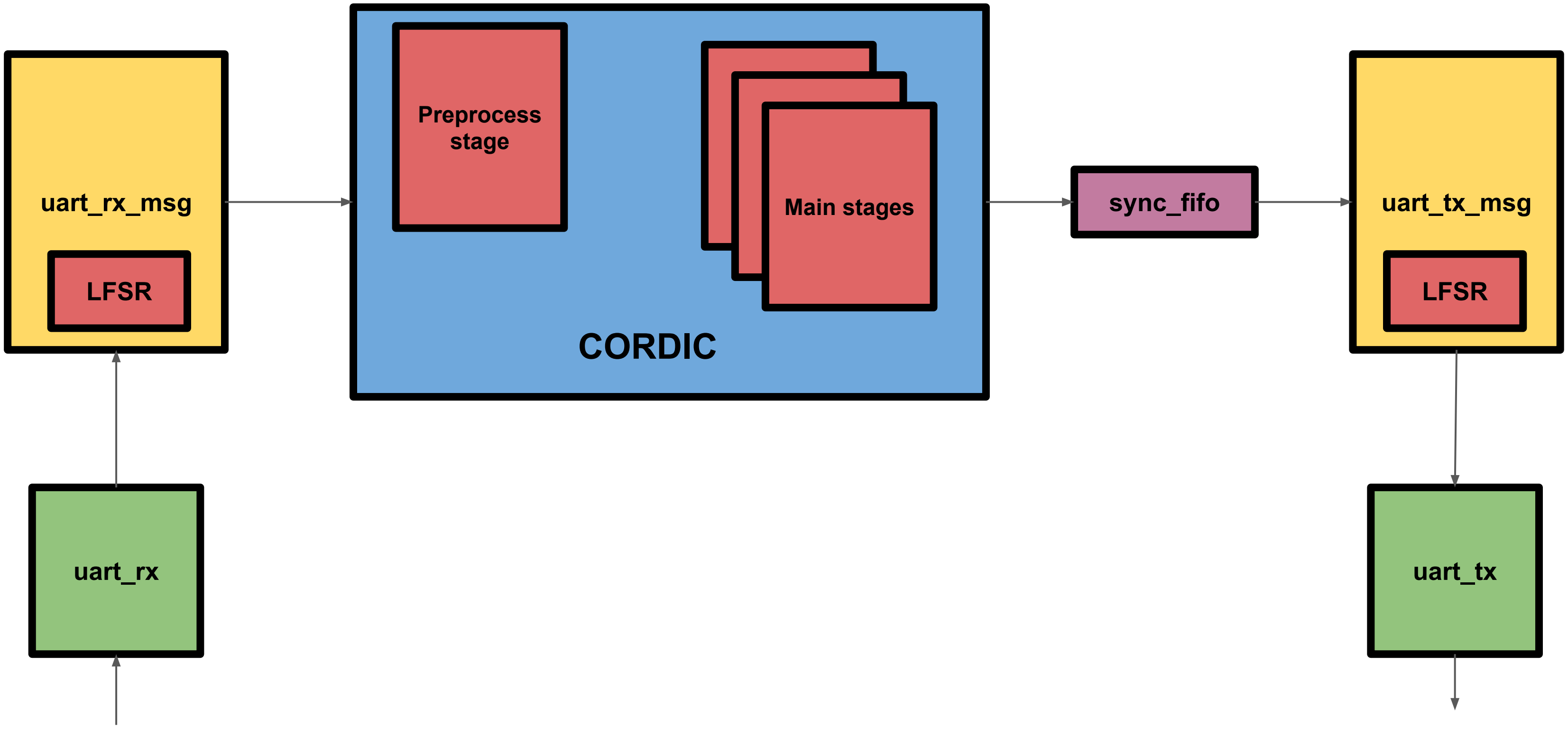
- The CORDIC is limited to input angles in the range [-pi/2, pi/2]
- Perform angle corrections if outside this range
    - Assuming $angle_{in}$ is in range [-2pi, 2pi]:
    - If $angle_{in}$ is in range[-2pi, -3pi/2], add 2pi to $angle_{in}$
    - If $angle_{in}$ is in range[-3pi/2, -pi/2], add pi to angle and invert the final cos/sin results
    - If $angle_{in}$ is in range[pi/2, 3pi/2], subtract pi from the angle and invert the final cos/sin results
    - If $angle_{in}$ is in range[3pi/2, 2pi], subtract 2pi from $angle_{in}$

Block Diagram - CORDIC Pipeline Stage

$$x_{n+1} = x_n - y_n(2^{-n})(sign(angle_n))$$
$$y_{n+1} = y_n + x_n(2^{-n})(sign(angle_n))$$
$$angle_{n+1} = angle_n - \theta_n(sign(angle_n))$$



Block Diagram - CORDIC FPGA Top Level



Design, Verification, and Implementation

- All HW components designed using SystemVerilog
  - CORDIC preprocessing and main pipeline blocks
  - Receiver and sender instruction messaging engines
  - Custom CRC-8 protocol implementation using Galois LFSR
  - UART RX/TX physical layer implementation
  - Reset synchronizer
  - Synchronous FIFO
- Verified using a custom UVM testbench (UVM 1.1d)
  - Developed golden reference DPI-C CORDIC model
  - Ran on QuestaSim 10.7c and VCS 2020.03, achieving 100% success rate and coverage
- Implemented on Arty A7 board (Artix-7 FPGA)
  - Achieved a 100MHz frequency
  - Using on-board USB-UART bridge (3MBaud) to connect with client PC program

Appendix

- GitHub repository: https://github.com/grant4001/CORDIC-UART-Artix-7
- EDA Playground TB simulation: https://edaplayground.com/x/9xNx

References

- Andraka, Ray. “A Survey of CORDIC algorithms for FPGA based computers.” http://www.andraka.com/files/crdcsrvy.pdf
- Arar, Steve. “An Introduction to the CORDIC algorithm.” https://www.allaboutcircuits.com/technical-articles/an-introduction-to-the-cordic-algorithm/
- Wikipedia, The Free Encyclopedia. “CORDIC.” https://en.wikipedia.org/wiki/CORDIC