# Dual Fixed-Point CORDIC Processor: Architecture and FPGA Implementation

**Andres Jacoby and Daniel Llamocca**

Electrical and Computer Engineering Department

Oakland University

Rochester, MI, USA

# Main Contributions

- ***Parameterized architecture validated on an FPGA:*** The hardware, developed in fully parametric VHDL code, can be implemented on any existing technology (e.g. FPGA, Programmable SoC, ASIC).
- ***Design Space Exploration:*** Trade-offs among resources, accuracy, and hardware design parameters are explored. Pareto-optimal realizations are also generated.
- ***Comparisons among DFX, FX, and FP architectures:*** The proposed DFX CORDIC was compared with similar FX and FP realizations to assess if the resource increase as well as accuracy.

# Dual Fixed-Point Representation

- An *n*-bit Dual Fixed-Point (DFX) number is composed of a (*n-1*)-bit signed significand ($X$) and an exponent bit ($E$). The exponent determines the scaling for the significand:
- $D = \begin{cases} num0: \ X.2^{-p_0}, & if\ E = 0 \\ num1: \ X.2^{-p_1}, & if\ E = 1 \end{cases}, \ p_0 > p_1$

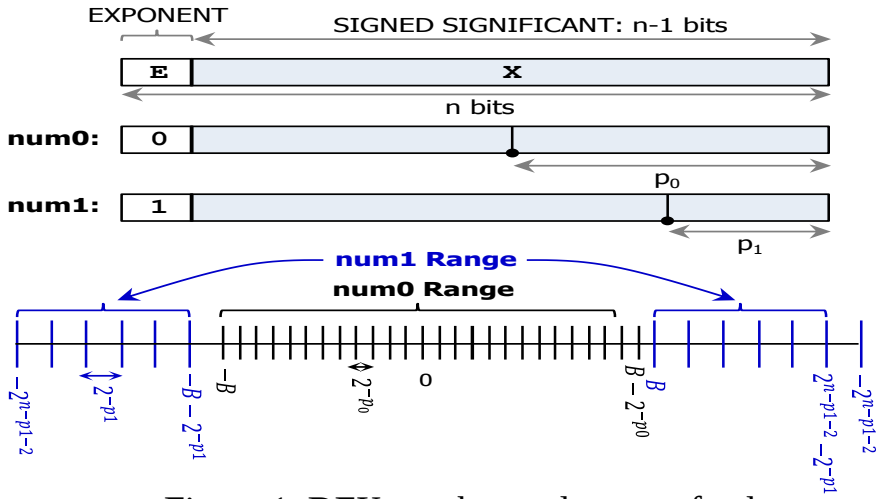


Figure 1. DFX number and range of values.

# Hardware Architectures

- The hardware developed in this work is fully parametric VHDL, below is a list of the cores:
  - DFX Basic Units:
    - DFX Adder/subtractor
    - DFX Multiplier
    - DFX Barrel shifter
  - DFX Expanded Circular CORDIC
  - DFX Expanded Hyperbolic CORDIC
  - DFX Logarithm
  - DFX Powering

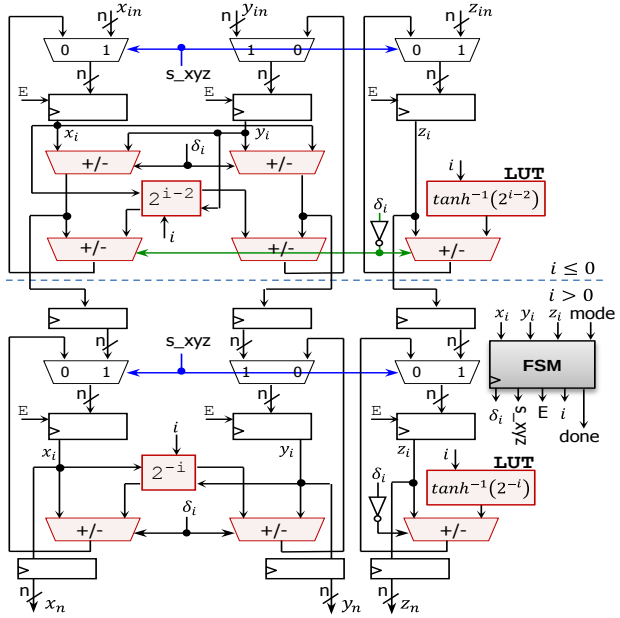


Figure 2. Expanded DFX Hyperbolic CORDIC.

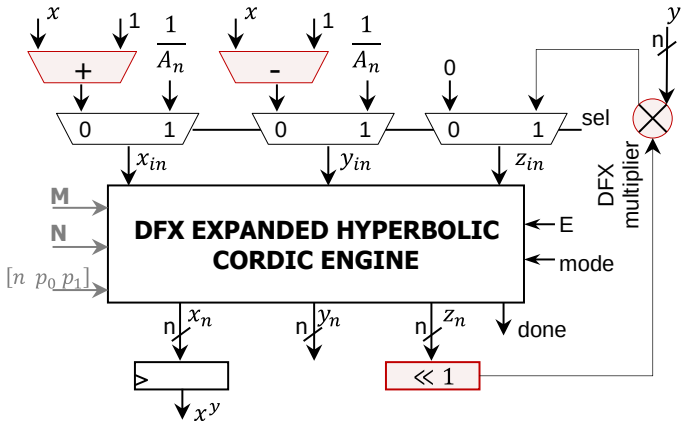


Figure 3. Fully parameterized DFX Powering.

# Setup and Design Space Exploration

- By varying $n$, $p_0$, $p_1$ (the DFX format), we create a design space of hardware configurations for every function to be tested. This also requires careful selection of the domain of the inputs.
- Some functions were only explored for a subset of the design space; this is due to intrinsic limitations such as convergence or CORDIC algorithm, scaling factor.

Table 1. Testing domain for the CORDIC-based functions.

| FUNCTION | INPUT DOMAIN FOR TESTING | CORDIC MODULE | M |
|---|---|---|---|
| $\sin(x)$, $\cos(x)$ | $-\pi \le x \le \pi$ | CIRCULAR: ROTATION. $z_{-M+1} = x$ $x_{-M+1} = 1/A_n$, $y_{-M+1} = 0$ | 2 |
| $\operatorname{atan}(x)$ | $0 \le x \le 20$ | CIRCULAR: VECTORING. $y_{-M+1} = x$ $x_{-M+1} = 1$, $z_{-M+1} = 0$ | 2 |
| $\sinh(x)$, $\cosh(x)$ | $0 \le x \le 4$ | HYPERBOLIC: ROTATION. $z_{-M} = x$, $x_{-M} = 1/A_n$, $y_{-M} = 0$ | 4 |
| $e^x$ | $-2 \le x \le 2$ | HYPERBOLIC: ROTATION. $z_{-M} = x$, $y_{-M} = x_{-M} = 1/A_n$ | 4 |
| $\operatorname{atanh}(x)$ | $\lvert x \rvert \le 0.9995$ | HYPERBOLIC: VECTORING. $y_{-M} = x$ $x_{-M} = 1$, $z_{-M} = 0$ | 5 |
| $\sqrt{x}$ | $0 \le x \le 36$ | HYPERBOLIC: VECTORING. $z_{-M} = 0$, $x_{-M} = x + 1/(4A_n^2)$, $y_{-M} = x - 1/(4A_n^2)$ | 3 |
| $\ln(x)$ | $0.0005 \le x < 15$ | HYPERBOLIC: VECTORING. $z_{-M} = 0$, $x_{-M} = x + 1$, $y_{-M} = x - 1$ | 5 |
| $x^y$ | $0.135 \le x \le 7.39$ $-2 \le y \le 2$ | HYPERBOLIC: VECTORING AND ROTATION | 4 |

# Results

- The Pareto-optimal realizations for $atan/lnx$ allows us to only consider optimal hardware realizations while simultaneously satisfying accuracy and resource constraints.
- Table 2 depicts how DFX compares to FX in terms of resources and accuracy. For $x^y$ on average, resources increased by 55% while accuracy improved 61.45dB.

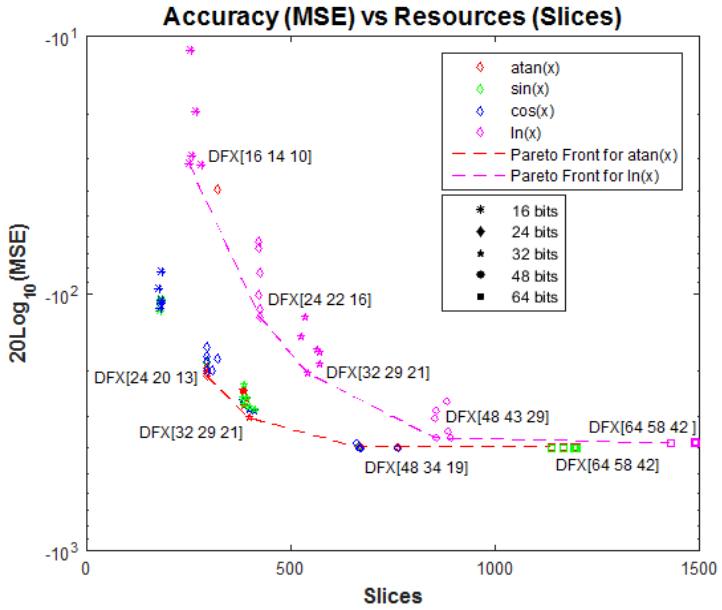


Figure 4. Accuracy-Resources design space for atan/sin/cos/lnx.

Table 2. DFX vs FX. Resources and accuracy.

| Fn. | FX | | DFX | avg resource accuracy inc. (DFX/FX) | FP EW:24 FW: 16 |
|---|---|---|---|---|---|
| $x^y$ | [24 15] | [24 9] | [24 15 9] | | |
| | 343<br>115.78 dB | 326<br>100.42dB | 518<br>46.65 dB | 55%<br>61.45 dB | 769<br>7.61 dB |
| lnx | [24 10] | [24 20] | [24 20 10] | | |
| | 198<br>-34.70dB | 200<br>28.49 dB | 439<br>-104.61dB | 120%<br>101.5 dB | 718<br>-135.2dB |
| sinh | [24 15] | [24 10] | [24 15 10] | | |
| | 201<br>71.62 dB | 197<br>-11.17 dB | 399<br>-35.29 dB | 100%<br>65.52 dB | 605<br>-37.92dB |

# Thank You