CORDIC Background (2B)

- FPGA Architecture
-

Young Won Lim
08/11/2011

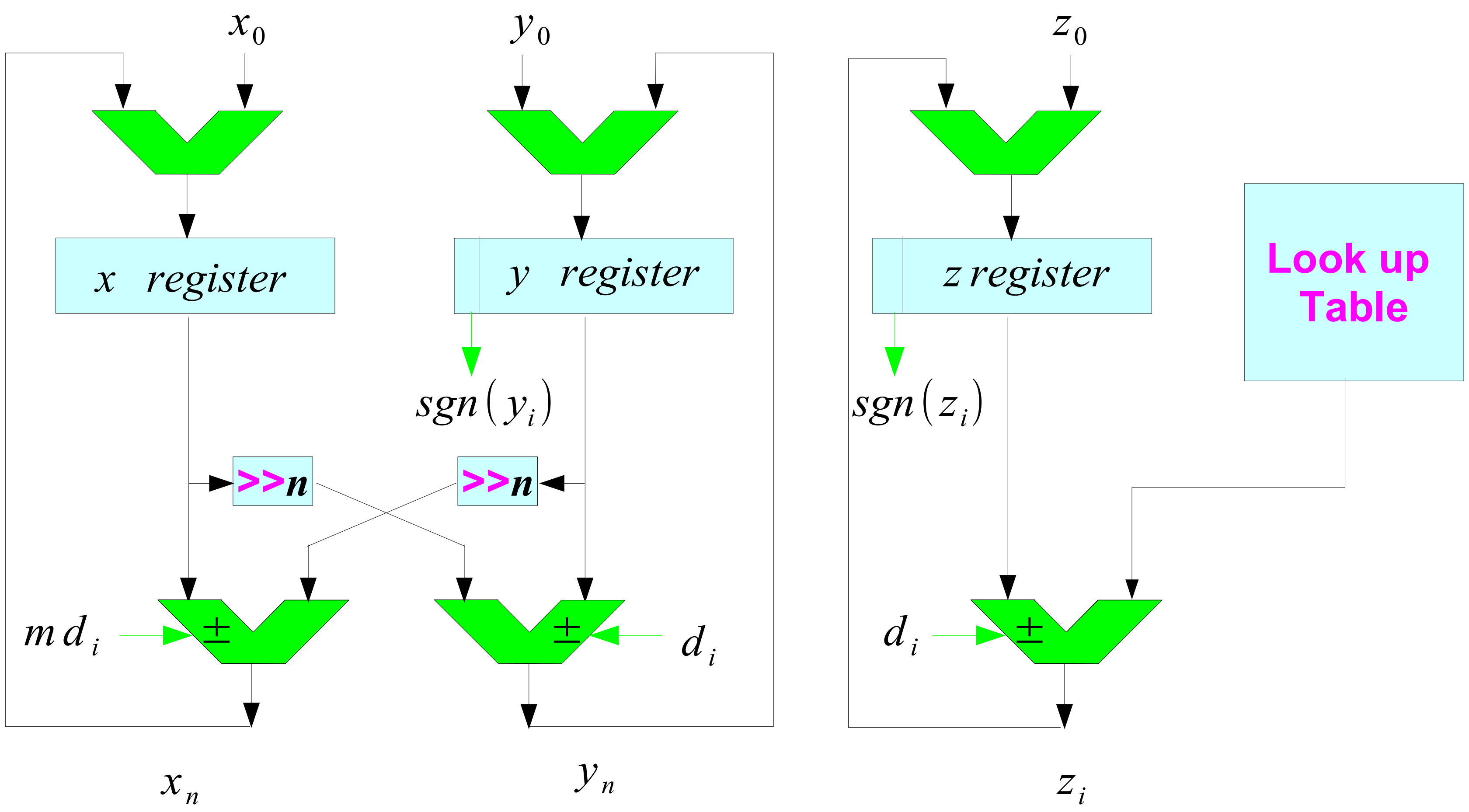
Please send corrections (or suggestions) to youngwlim@hotmail.com.

This document was produced by using OpenOffice and Octave.

CORDIC Background

1. A survey of CORDIC algorithms for FPGAs, Ray Andraka, www.andraka.com/cordic.htm

Bit-Parallel CORDIC: Structure



Bit-Parallel CORDIC: Decision Function

Decision Function d_i

In rotation mode

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

$$\begin{aligned} d_i &= -1 \quad \text{if} \quad z_i < 0 \\ d_i &= +1 \quad \text{otherwise} \end{aligned}$$

sign of z register

In vectoring mode

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

$$\begin{aligned} d_i &= +1 \quad \text{if} \quad y_i < 0 \\ d_i &= -1 \quad \text{otherwise} \end{aligned}$$

sign of y register

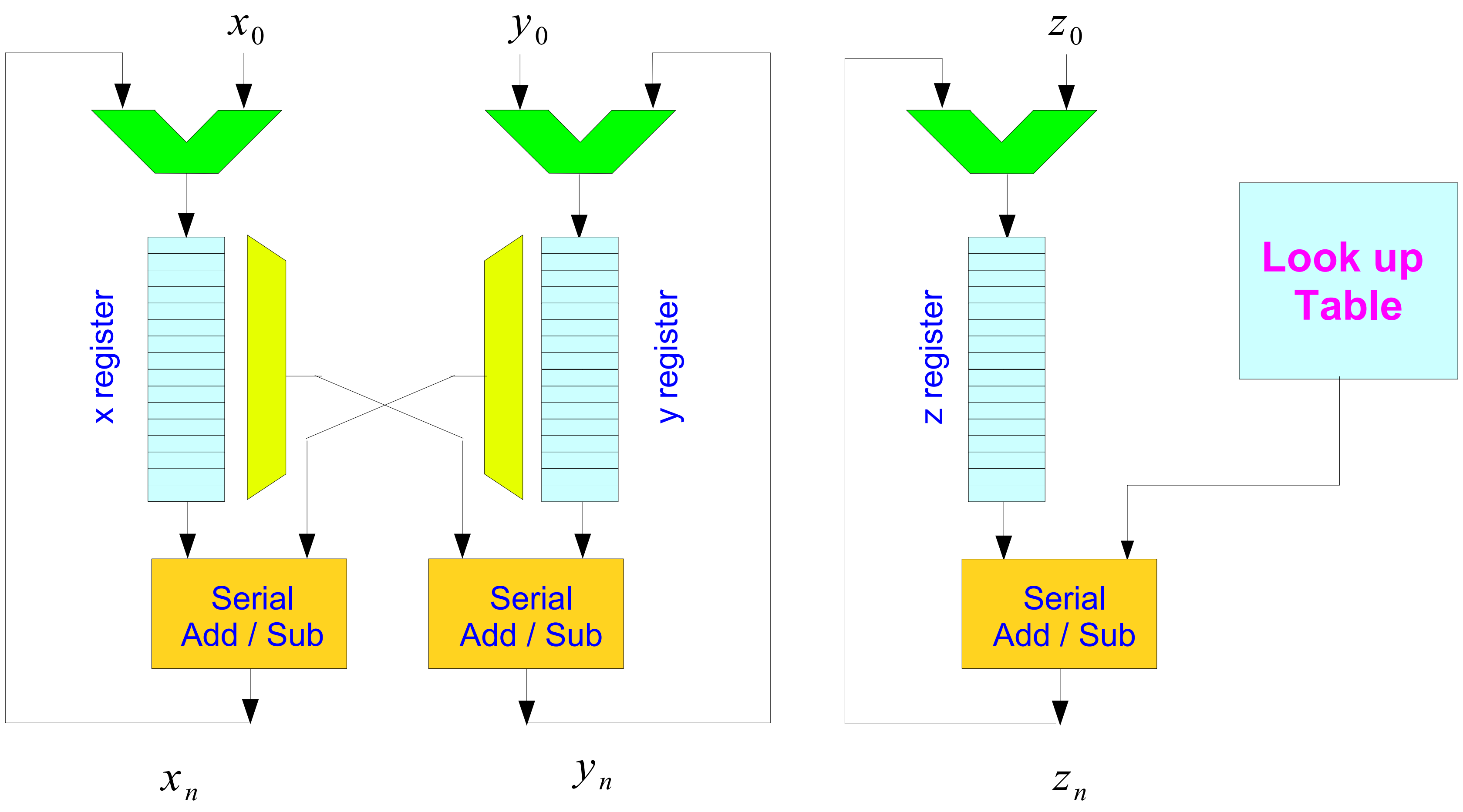
Bit-Parallel CORDIC: Characteristics

N iteration (clock cycles)
Variable Shifters
Adder- Subtractors
the amount of shift
the address of ROM LUT
the proper elementary angle
to z add/ subtractor

Bit-parallel variable shifter
Not good for FPGA
High Fan-in
→ several layers of logic
→ many cells to be traversed

- Slow design
- Large Area

Bit-Serial CORDIC: Structure



Bit-Serial CORDIC: Characteristics

Compact design
Simple interconnection →
Higher clock rate – up to the max rate of the FPGA
w :word width
w clocks per each iteration
3 bit-serial adder-subtractors
3 shifter registers
Shift tap multiplexers →

- Multiplexer tree: wiring problem
- Tri-state bus: ?
- Still simple interconn & logic

1 serial ROM
No shifting – using bit delays

During the nth iteration,
the result is read from the serial adders
While the next initialization data is shifted into the registers

Bit-Serial CORDIC: Shifter Register Implementation

Xilinx 4000E

Shift register

implemented in the CLB RAM

16x1 RAM emulates shift register

By incrementing read/write address

after each access

Dual port –

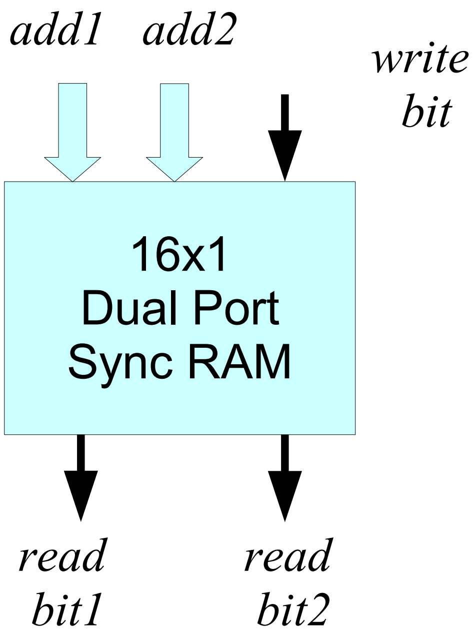
simultaneous reading 2 locations

By properly sequencing the 2nd address

→ shift tap multiplexer realized

w/o using actual multiplexer

→ upto 16 bit word length, 1 CLB is enough



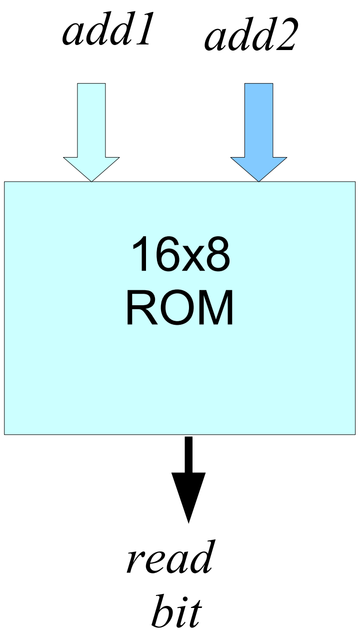
Bit-Serial CORDIC: ROM

Serial ROM

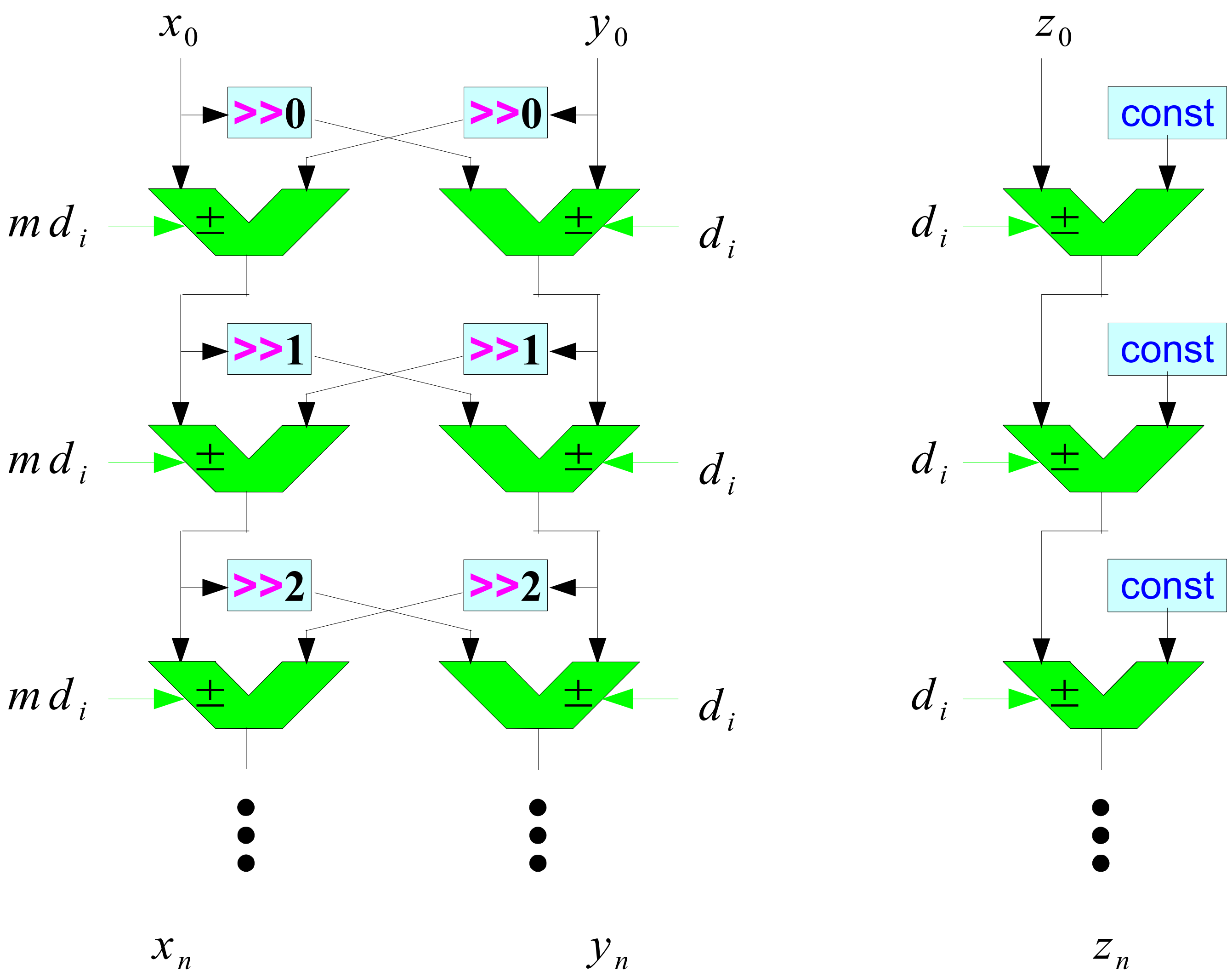
1 CLB is required for every 2 iteration

16-bit, 8 iteration CORDIC

16x8 ROM



Loop Unrolled CORDIC: Structure



Loop Unrolled CORDIC: Characteristics

design

Sims

Unified CORDIC Iteration Eq

References

[1] http://en.wikipedia.org/
[2] CORDIC FAQ, www.dspguru.com
[3] R. Andraka, A survey of CORDIC algorithms for FPGA based computers
[4] J. S. Walther, A Unified Algorithm for Elementary Functions