Beamforming

PETER ISHO

What is Beamforming?

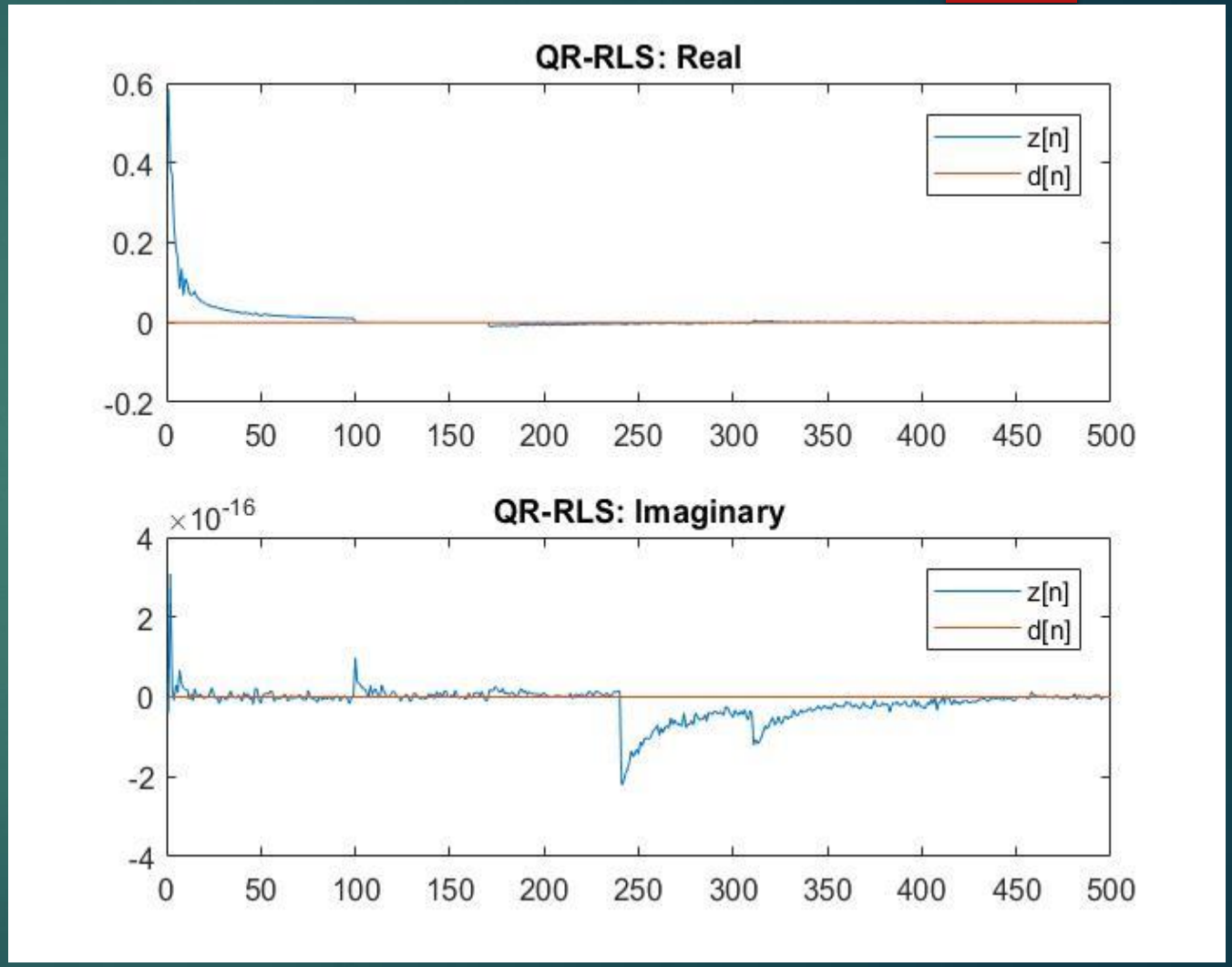
- Essentially a directed signal
- Signals output in all directions
- Faster, stronger, and more range

Broad Overview

- Input matrix of MxN data
- M sensors outputting signals from all directions
- One output direction after beamformer

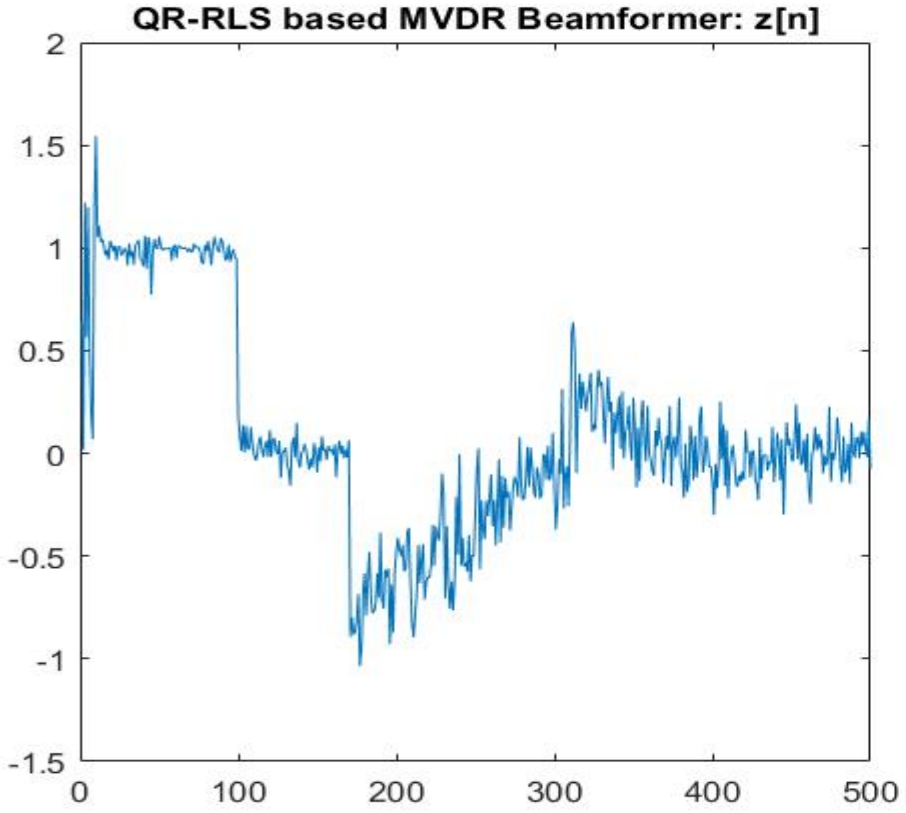
QR-RLS Algorithm

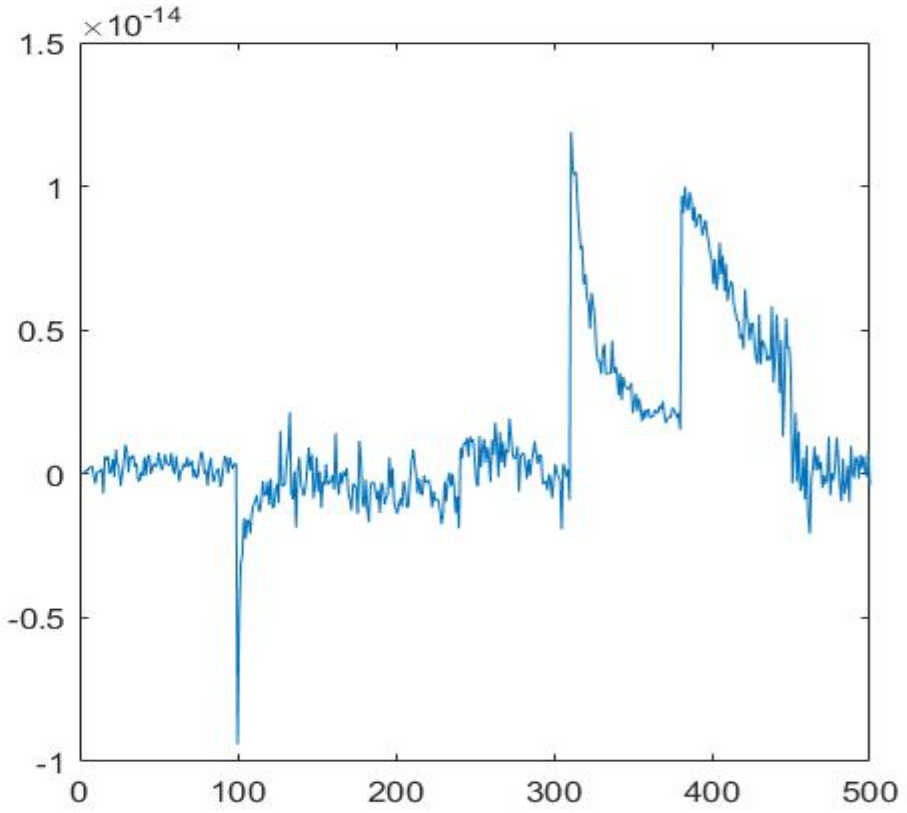
- Professor Llamocca helped with a MATLAB implementation
- Left Side – Real
- Right Side – Imaginary
- Target is at zero so output strives for zero



Output Signal

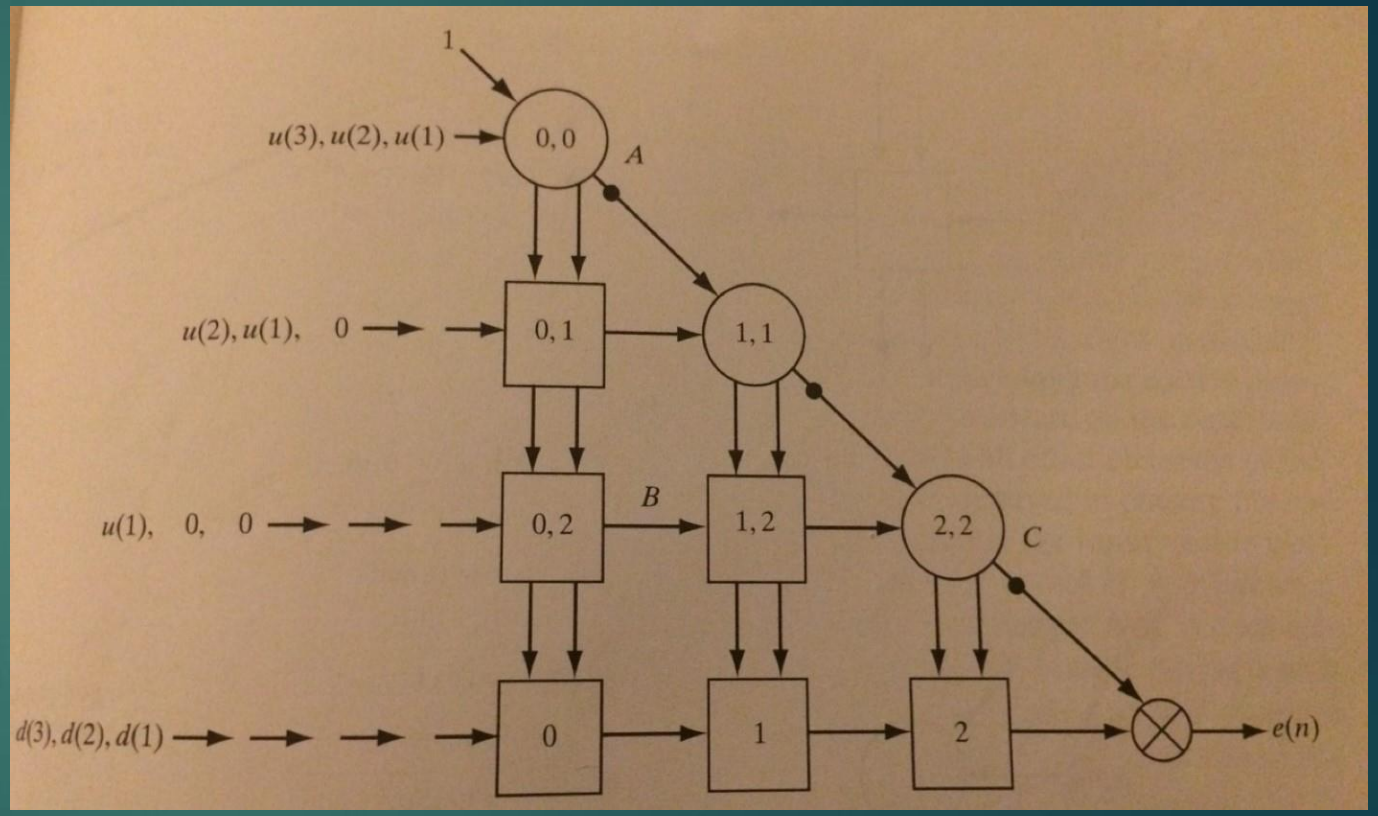
- This is the output signal after the beamformer





Hardware Implementation

- Systolic Array
- Initial Plan to design
- Timing would be hard

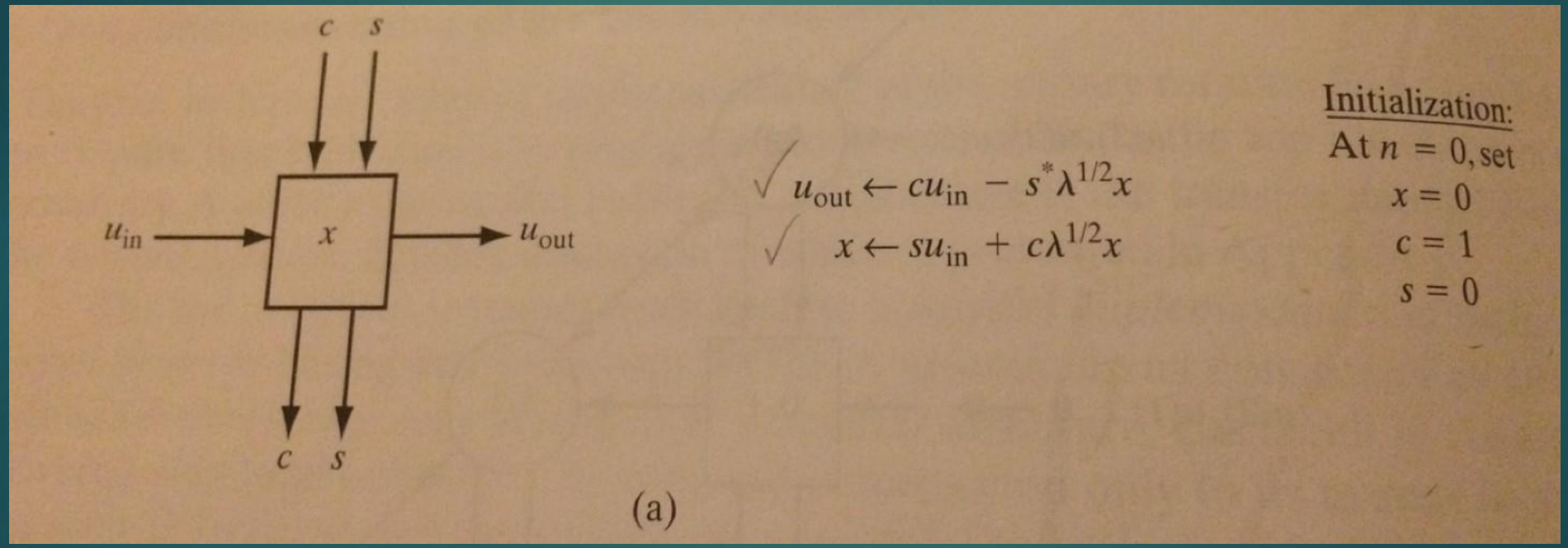


Complex Numbers

- Some of these signals are complex
- Therefore, a real and imaginary path are used
- Input data = .56 + .85i
- Split this into two signals: Real = .56 and Imaginary = .85

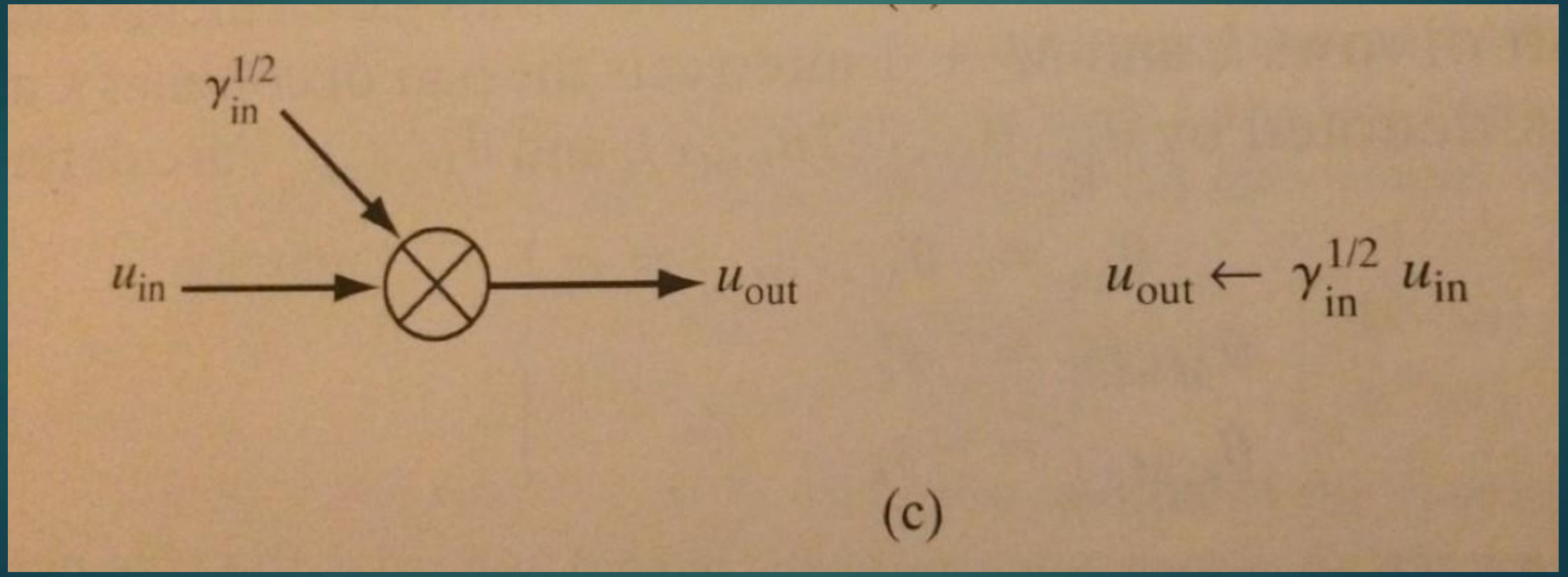
Internal Cell

- Two multiplications and two additions



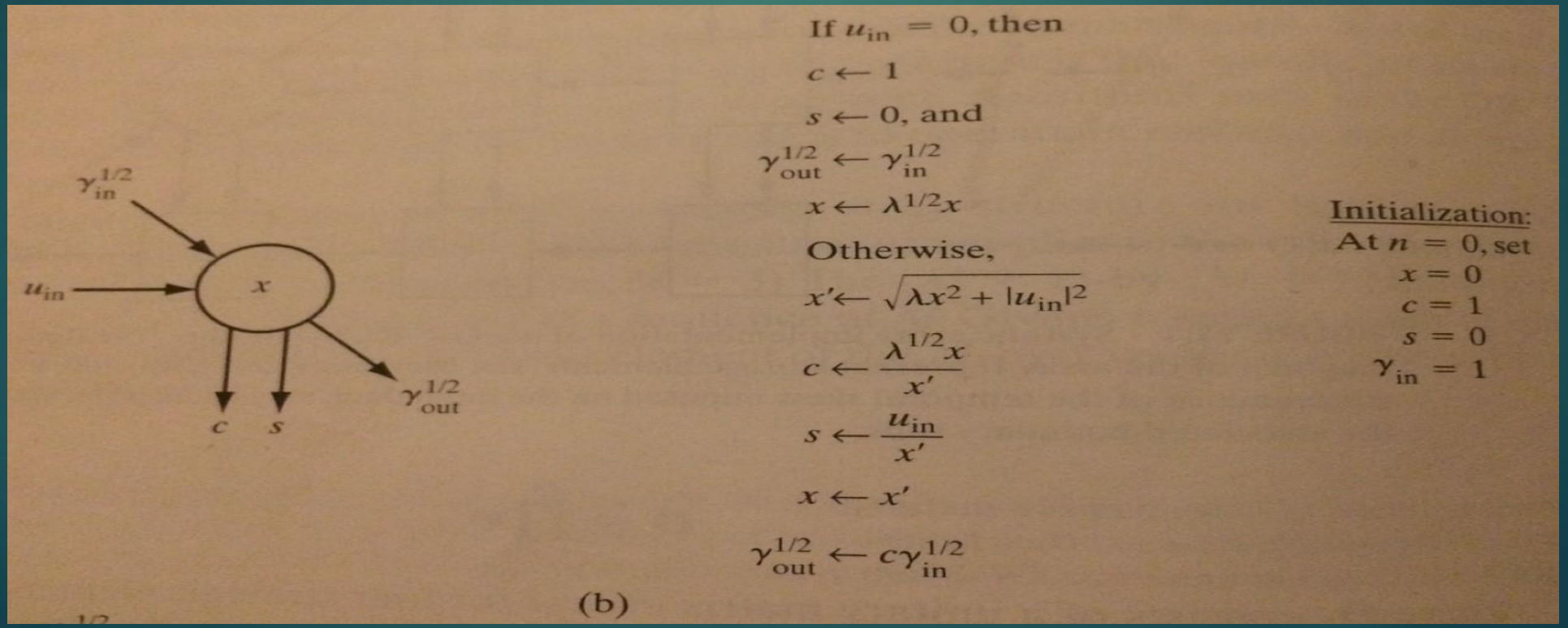
Final Processing Cell

- One multiplication for the output of the beamformer



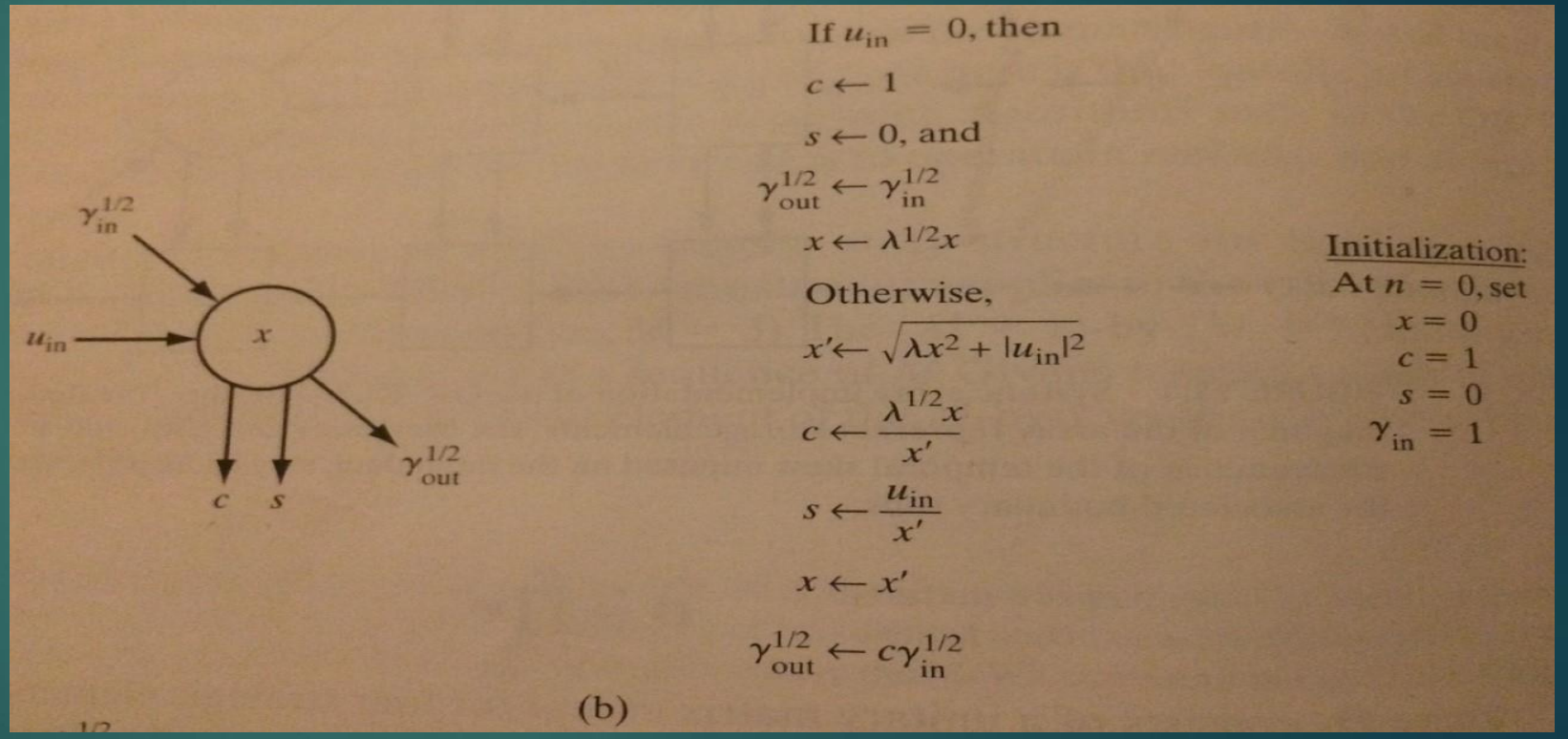
Edge Cell

- Contains Cordic, Division, Multiplication
- Entire thing Pipelined
- Right now I have the non-pipelined version complete



"FSM"

1. Calculate absU
2. Calculate x'
3. Calculate cO, sO, x
4. Calculate yO



Accuracy – Non-Pipelined

Inputs	Outputs Vivado	Outputs MATLAB
Y_in = 1.0	X = .309997	X = 0.3101
U = 1.28 + .2824i	C = 0.0	C = 0.0
X = 0.0	S = .41284 + .91119i	S = 0.4127 + 0.9109i
	Y = 0.0	Y = 0.0

Accuracy – Non-Pipelined

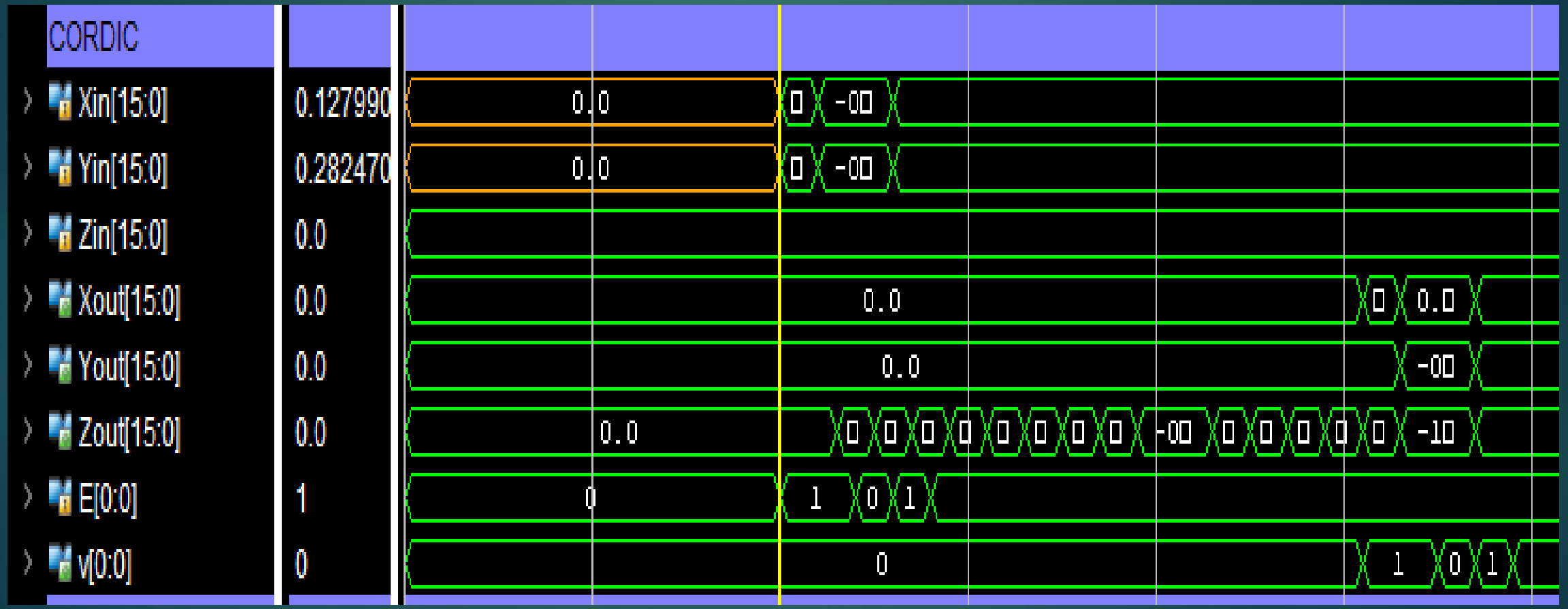
Inputs	Outputs Vivado	Outputs MATLAB
Y_in = 1.0	X = 0.328247	X = 0.3283
U = -0.03564 – 0.10211i	C = 0.944396	C = 0.9442
X = 0.309997	S = -0.10858 – 0.31103i	S = -0.1084 – 0.311i
	Y = 0.944396	Y = 0.9442

Accuracy – Non-Pipelined

Inputs	Outputs Vivado	Outputs MATLAB
Y_in = 0.6854	X = 0.34505	X = 0.9246
U = 0.2304 – 0.833i	C = 0.95556	C = 0.3552
X = 0.328247	S = 0.6707 + 1.575i	S = 0.2492 – 0.901i
	Y = 0.65496	Y = 0.2435

- Due to Cordic [16 14] FX format
- The real value can't be expressed due to limited number of integer bits

Can’t Pipeline Feedback Loop



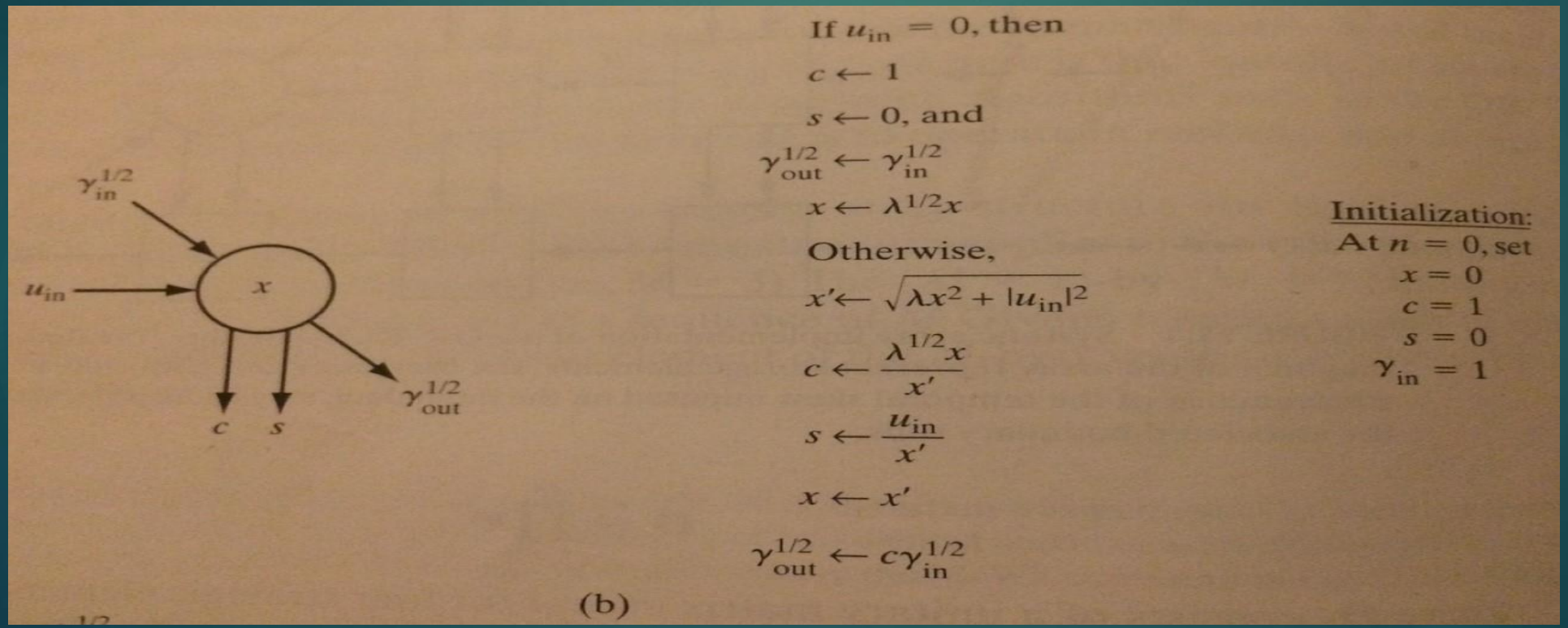
Cycle Time

- Theoretically takes 1 cycles
  - Actually takes 152 cycles PER boundary cell
    - Cordic
      - Cordic
        - 2 parallel divisions
          - 1 multiplication
- Can't Pipeline because of feedback loop
  - This makes it pointless to pipeline anything else
  - This is the cause of the increased cycle time

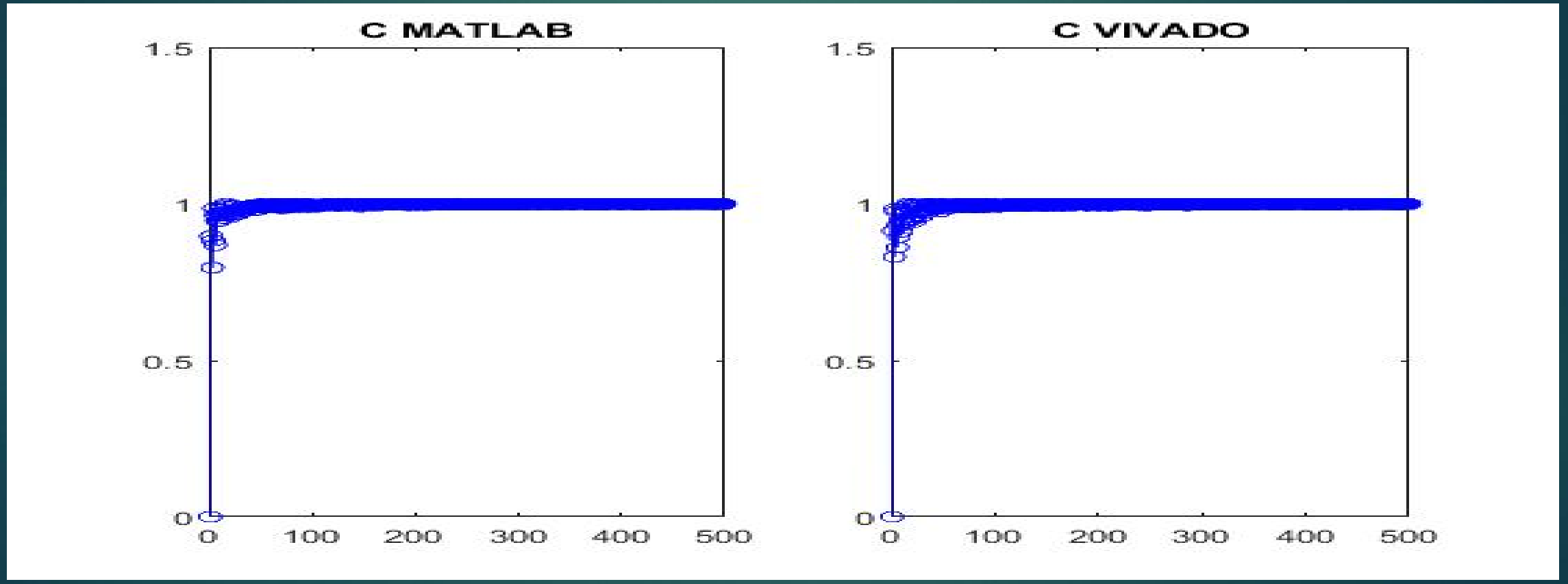
X outgrows [16 14]

- Theoretically takes 1 cycle
  - Actually takes 152 cycles

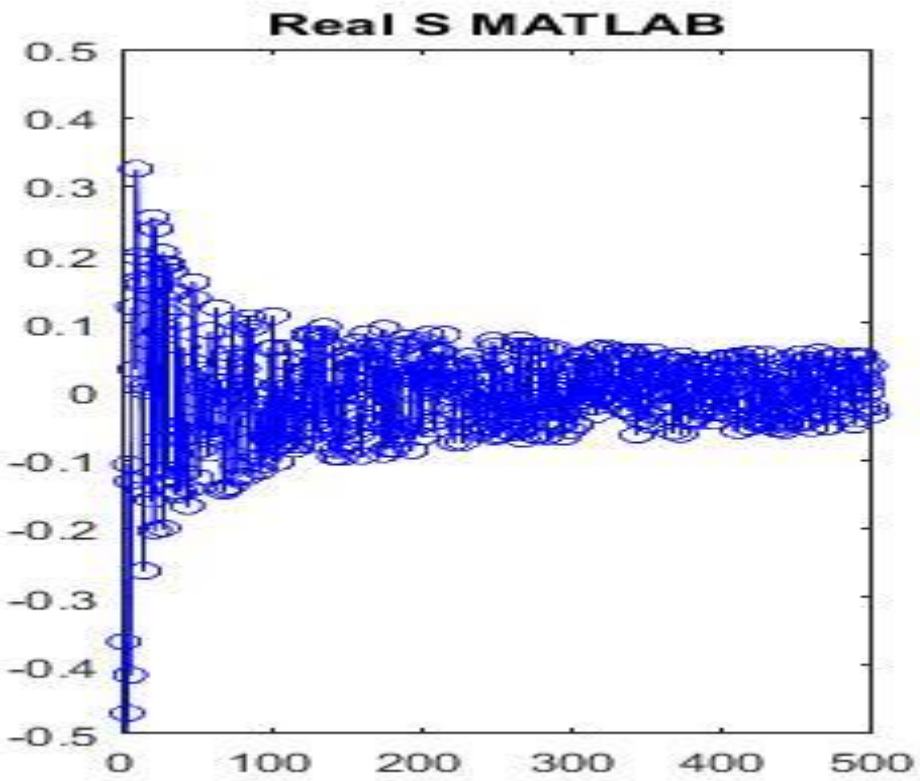
Iterations	x	Integer Bits
100	8.53	4
1000	25.32	5
10000	81,98	7

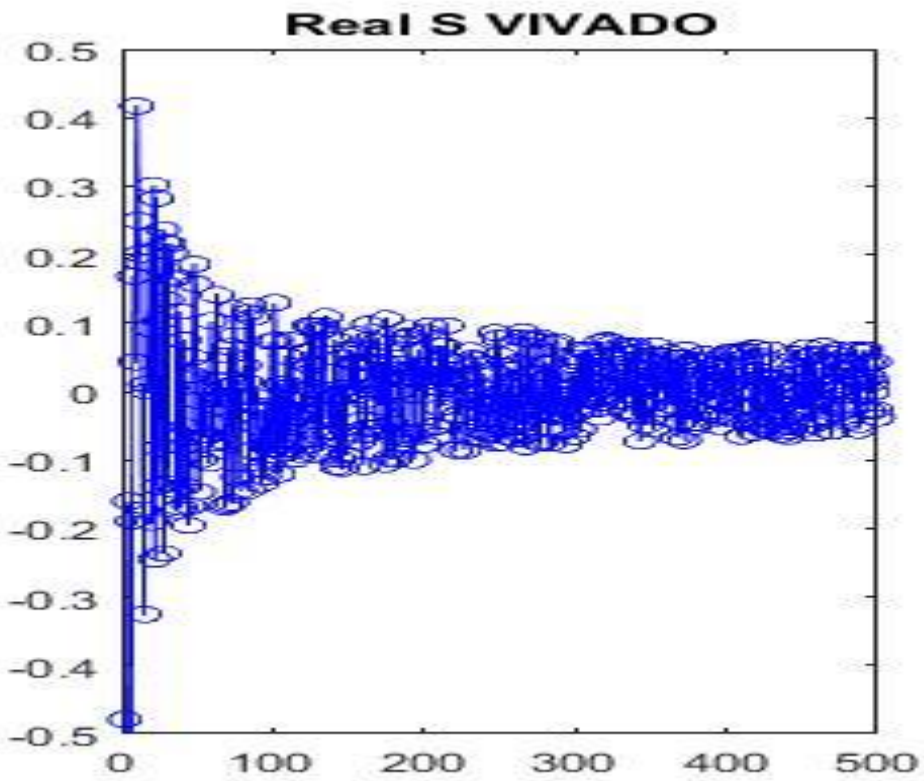


Simulation - 500 Inputs

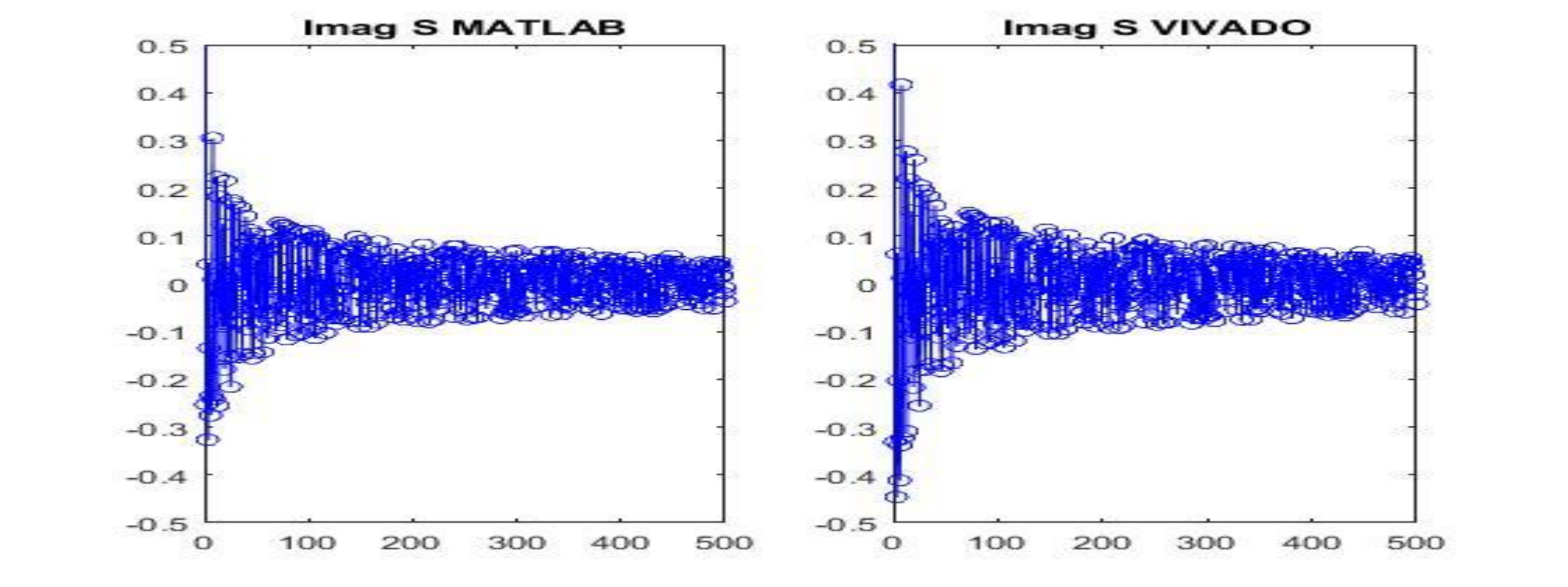


Simulation - 500 Inputs



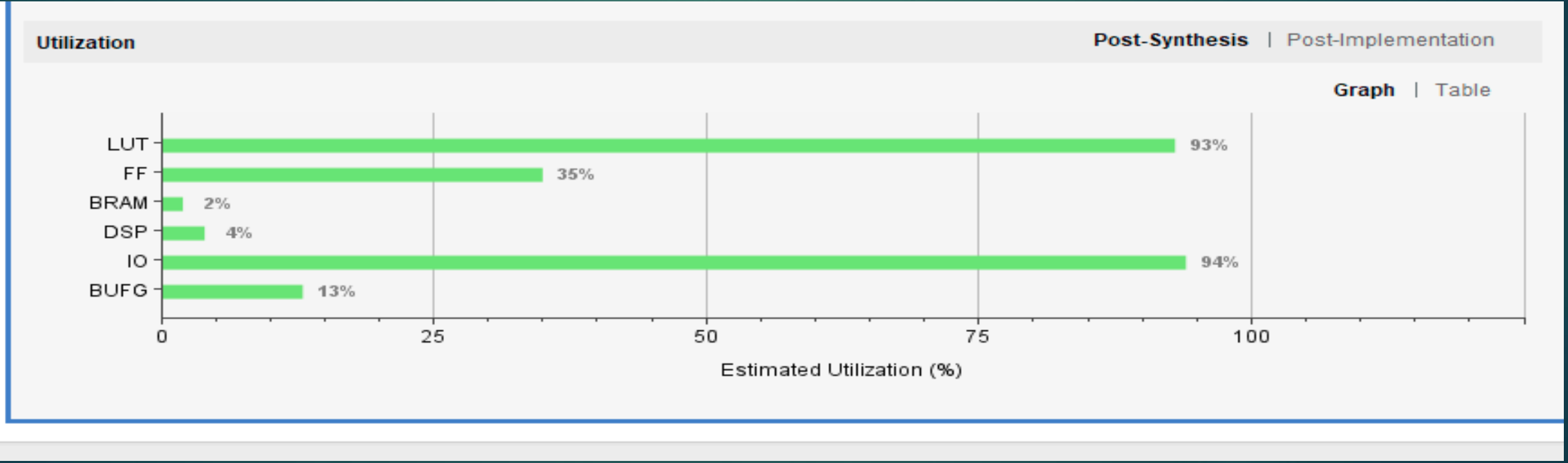


Simulation - 500 Inputs



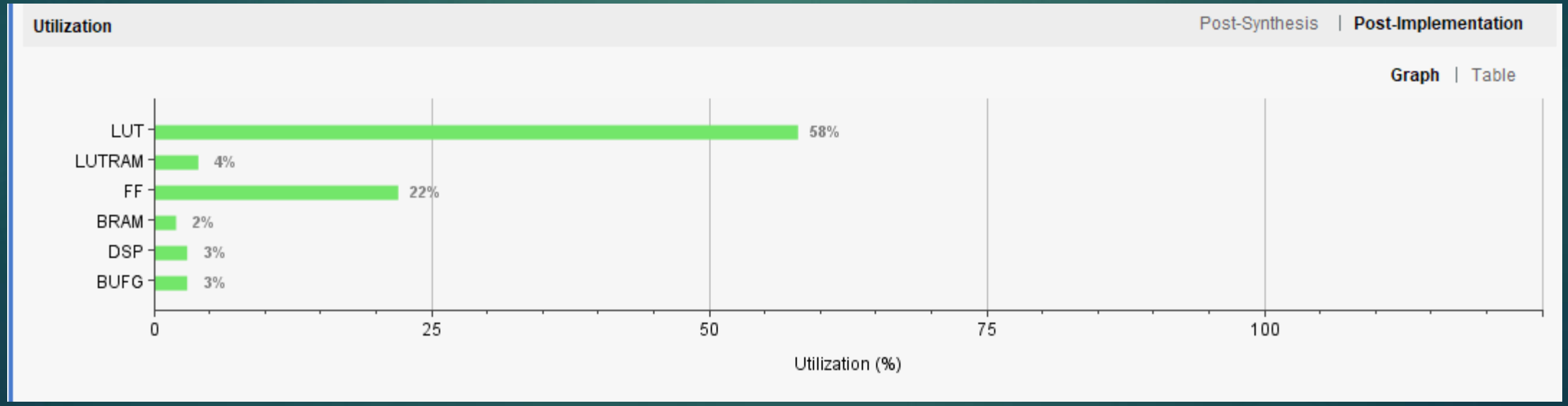
3 Antennas

- Surpassed the number of LUTs on the FPGA
- 93% only my hardware
- Had to move down to 2 antennas



2 Antennas

- Successfully used AXI-Full with 2 antenna beamformer
- Didn't have time to analyze results
- Pretty sure they were incorrect
- No demo for the 2 antenna beamformer



AXI-Full Interface

- Using iFIFO and oFIFO as before
- Develop FSM
- Write data on iFIFO one after another
- Write data from oFIFO when v signal is high

Future Development

- Estimation?
- Have a working Beamformer
- Eventually hook an antenna up to fpga and test

Questions?