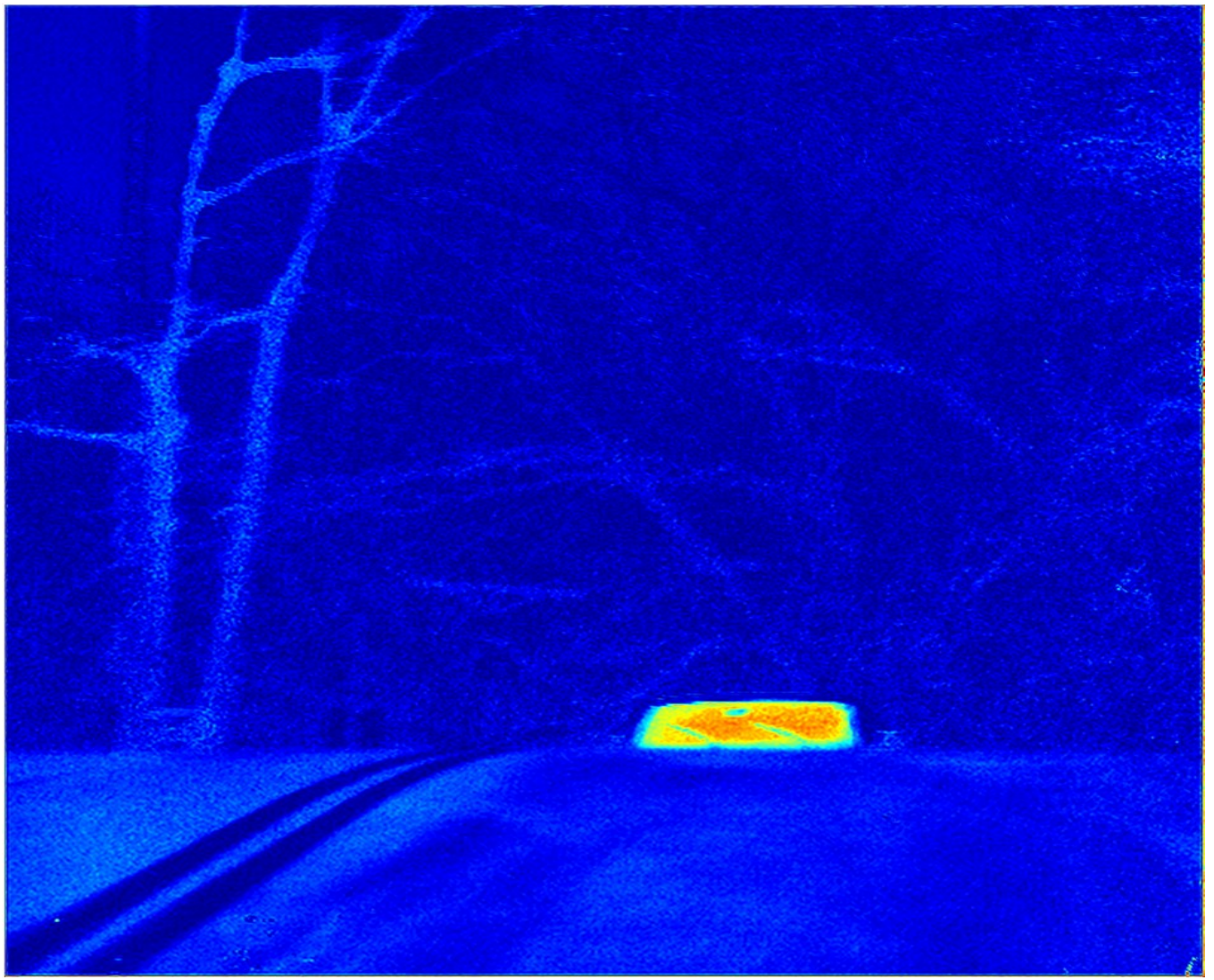
Polarization Pipeline

Filtermen:
Steve Berul
David D'Alessandro
Andrew Deck

Background



Viktor Gruev

Motivation

- Currently on CPU
  - Drains battery life
  - Battery runs out before air tanks

Where We Come In

- Our task:
  - Create polarized pipeline to be deployed on FPGA using ScalaPipe
  - Time trial individual kernels to determine bottlenecks in pipeline
  - Suggest best architecture for pipeline given bottlenecks

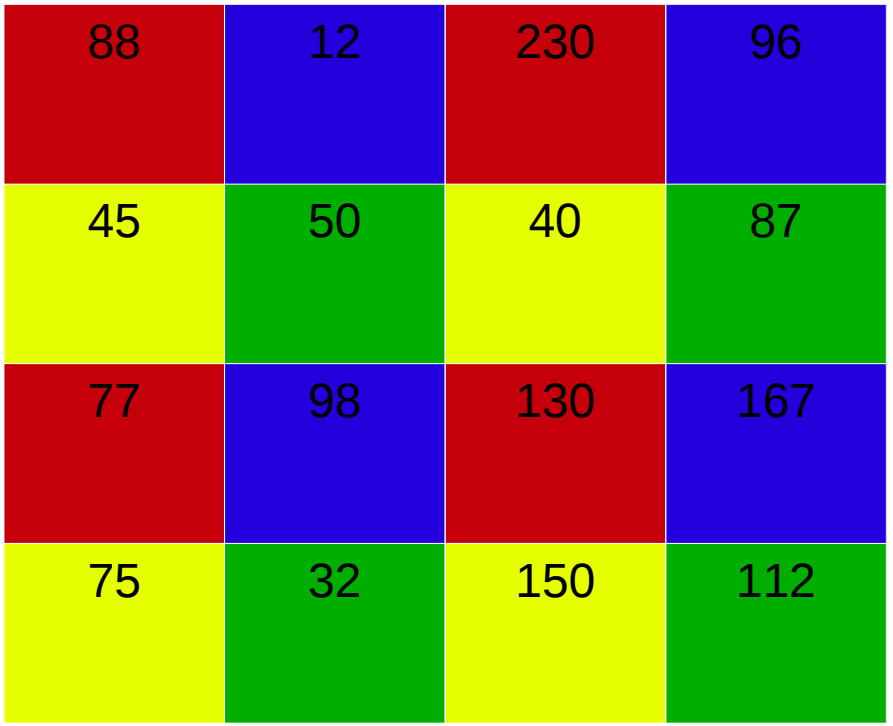
Where We Come In

- Data Throughput
  - 1920x1080 frames
  - 30 frames/second
  - 62.2 M pixels/second
- FPGA assumptions
  - Virtex 7
    - Best case clock rate: 700 Mhz

Pipeline Structure

- Border Extension
- Interpolation
- Stokes Parameter Calculation
- Degree of Linear Polarization
- Angle of Linear Polarization
- HSV Generator
- HSV->RGB Image Converter

Border Extension



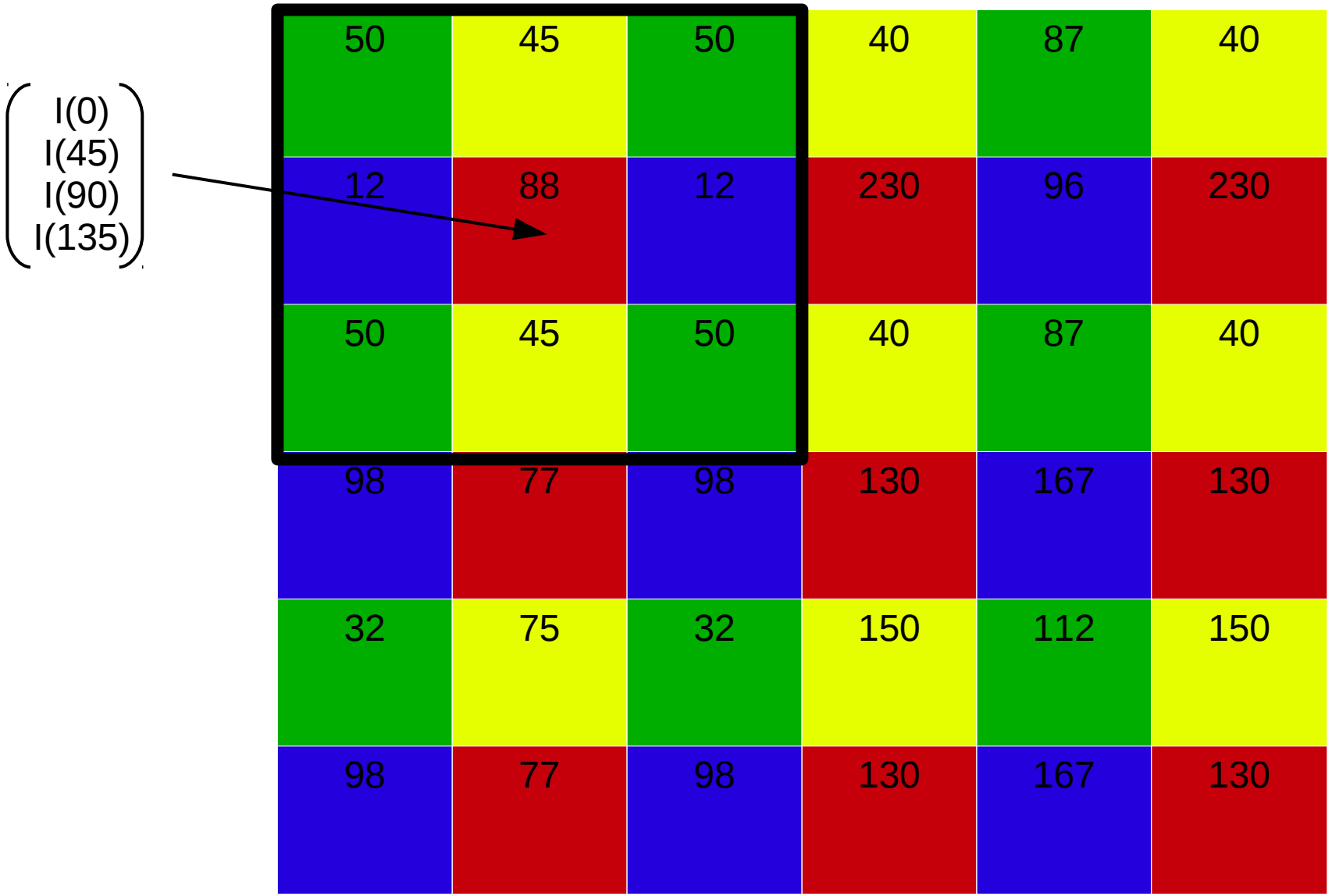
Border Extension

50	45	50	40	87	40
12	88	12	230	96	230
50	45	50	40	87	40
98	77	98	130	167	130
32	75	32	150	112	150
98	77	98	130	167	130

Border Extension

- Inputs:
  - Image Data – UNSIGNED32
- Outputs:
  - Image Data – UNSIGNED32
- Timing:
  - 7.8 cycles/pixel
  - 42.9 frames/second

Interpolation



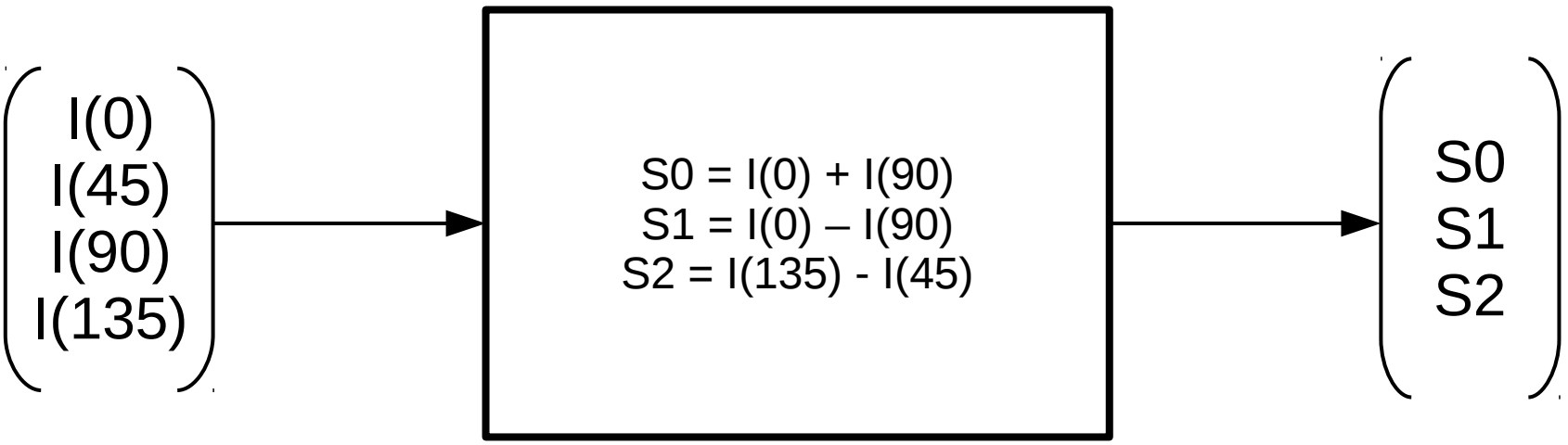
Interpolation

- Inputs:
  - Image Data – UNSIGNED32
- Outputs:
  - Interpolated Intensities – Vector(FLOAT32, 4)
- Timing:
  - 226.8 cycles/pixel
  - 1.5 frames/second

Interpolation

- Suggested Optimizations:
  - Parallelization
  - Integer interpolation
    - Loss of accuracy
    - Increase in speed

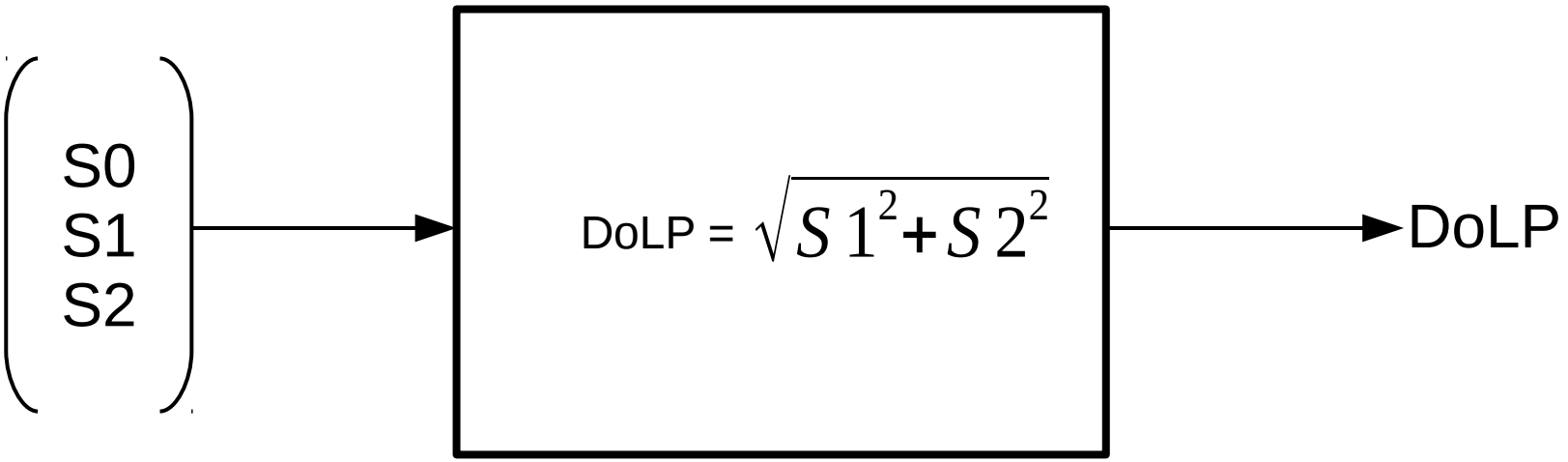
Stokes Parameters



Stokes Parameters

- Inputs:
  - Interpolated Intensities – Vector(FLOAT32, 4)
- Outputs:
  - Stokes Parameters – Vector(FLOAT32,3)
- Timing:
  - 23.9 cycles/pixel
  - 14.1 frames/second

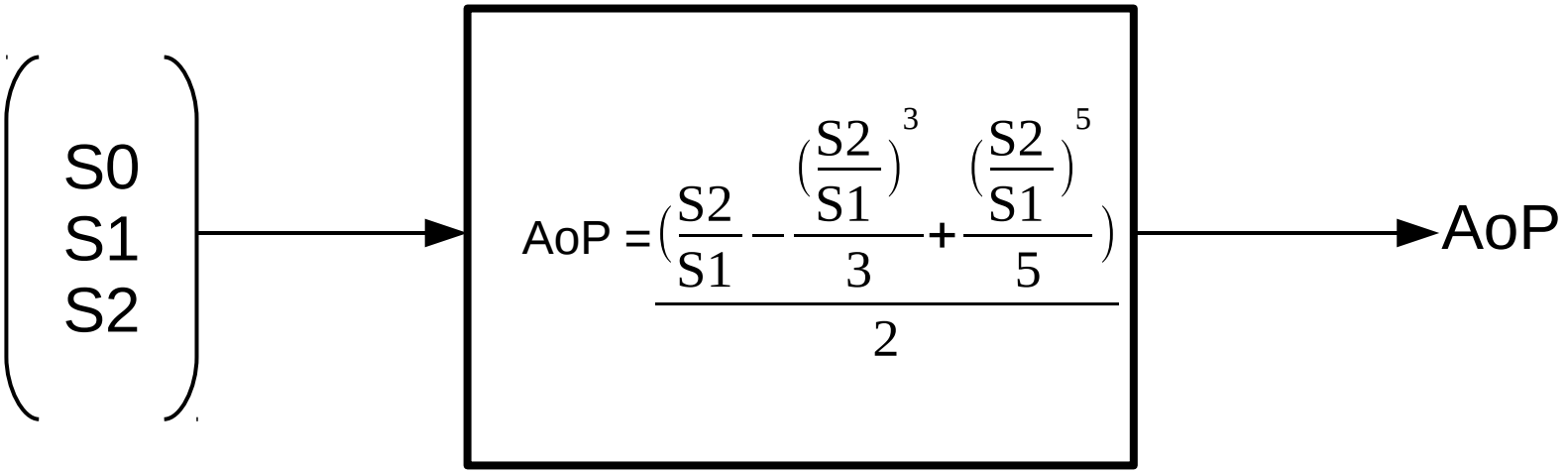
Degree of Linear Polarization



Degree of Linear Polarization

- Inputs:
  - Stokes Parameters – Vector(FLOAT32, 3)
- Outputs:
  - Degree of Linear Polarization – FLOAT32
- Timing:
  - 115.9 cycles/pixel
  - 2.9 frames/second

Angle of Polarization



Angle of Polarization

- Inputs:
  - Stokes Parameters – Vector(FLOAT32, 3)
- Outputs:
  - Angle of Polarization – FLOAT32
- Timing:
  - 236.5 cycles/pixel
  - 1.4 frames/second

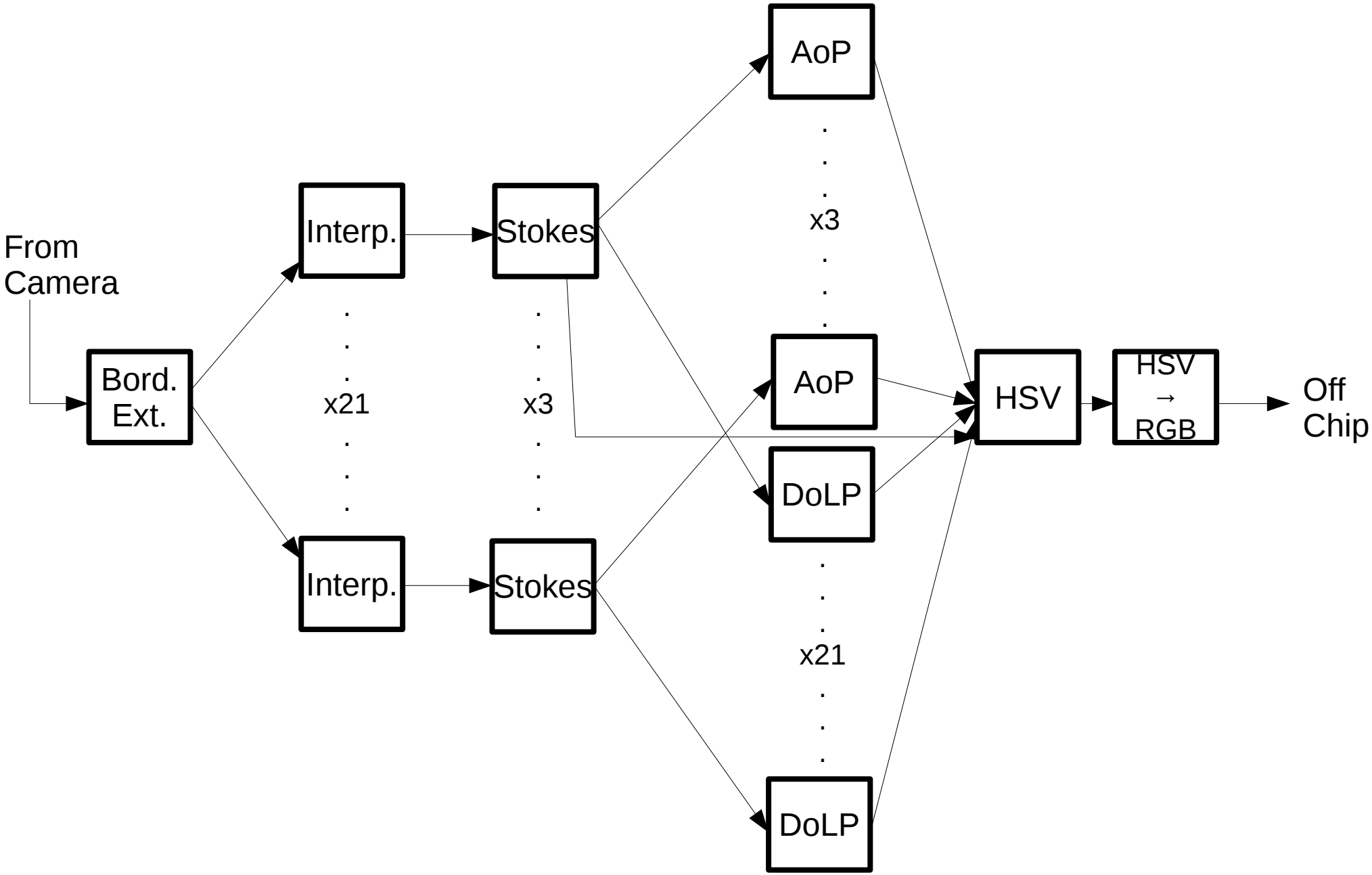
Generate HSV

- Inputs:
  - S0 – FLOAT32
  - Degree of Linear Polarization – FLOAT32
  - Angle of Polarization – FLOAT32
- Outputs:
  - HSV – Vector(FLOAT32, 3)
- Timing:
  - No results yet, but not a bottleneck

HSV → RGB

- Inputs:
  - HSV – Vector(FLOAT32, 3)
- Outputs:
  - RGB – Vector(UNSIGNED16, 3)
- Timing:
  - No results yet, but not a bottleneck

Suggested Architecture



Questions Still Seeking Answer

- Will this architecture fit on the FPGA?
  - Synthesis issues