

Review of:
**Evaluation of the Streams-C C-to-FPGA Compiler:
An Applications Perspective**

- **Paper by:**
 - Jan Frigo, Maya Gokhale, and Dominique Lavenier
 - Los Alamos National Labs and IRISA - CNRS
- **Published in:**
 - FPGA 2001
 - Monterey, Feb 2001
- **Survey by:**
 - David V. Schuehler

The Challenge

- Reduce the time required to design reconfigurable applications
- Enable the mainstream use of FPGAs by simplifying development tools

Style of the Paper

- Conference paper
 - Evaluation of Streams-C
 - Analysis of Applications
- Brief overview of Streams-C
- Evaluate 4 applications developed with Streams-C
- Compare results versus traditional development
 - Development time
 - FPGA usage
 - Speed of design

Hardware Platform

- Annapolis Microsystems Wildforce
- Xilinx 4036



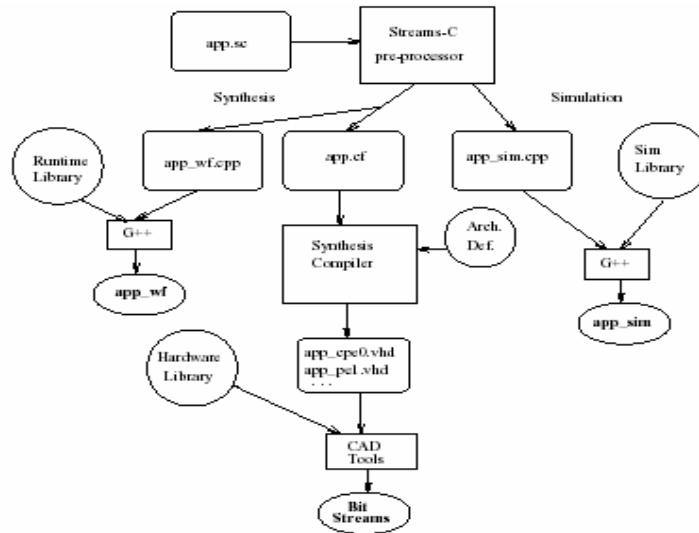
Streams-C Overview

- Stream based computation model supports
 - Streams of data
 - High data rates
 - Complex operations
- Extensions to C programming language
 - Syntax
 - Library functions
 - Hardware & software targets

Streams-C Function

```
/// PROCESS_FUN <function_name>
/// IN_STREAM <stream_element_data_type_name> <stream_name>
... other input streams ...
/// OUT_STREAM <stream_element_data_type_name> <stream_name>
... other output streams ...
/// IN_SIGNAL <signal_element_data_type_name> <signal_name>
... other input signals ...
/// OUT_SIGNAL <signal_element_data_type_name> <signal_name>
... other output signals ...
/// PARAM <parameter_type_name> <parameter_name>
... other parameter declarations ...
/// PROCESS_FUN_BODY
... C code ...
/// PROCESS_FUN_END
```

Development Process



Applications Analyzed

- Contrast Enhancement
 - Image grayscale adjustment
- Poly-phase Filter Bank
 - Multi-bank filter for signal detection
- Pixel Purity Index
 - Analyzing hyperspectral images
- K-means Clustering Algorithm
 - Image compression

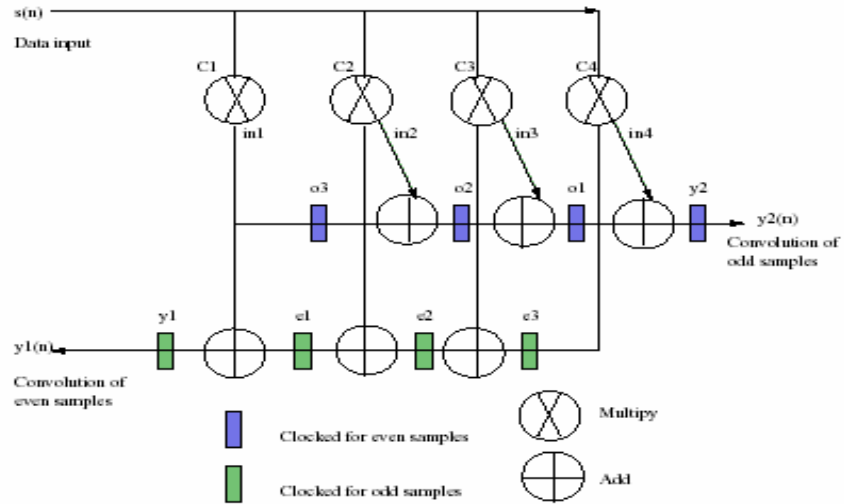
Contrast Enhancement

- Histogram based contrast enhancement
- Phases
 - Histogram generation (FPGA 1)
 - Contrast stretch generation (FPGA 1)
 - Image Remapping (FPGA 2)
- Hand coded phase 2
 - 18% of chip, 40MHz, 1 month
 - Results every clock cycle
- Streams-C
 - 57% of chip, 20MHz, 2 days
 - Results every other clock cycle (no CLB RAM)

Poly-phase Filter Bank

- Multi-rate filter banks help detect RF signals in noisy environments
- Utilizes FIR filter and FFT
- Hand coded
 - 27% of chip, 40MHz, 2 weeks
 - Results every clock cycle
- Streams-C
 - 37% of chip, 40MHz, 3 days
 - Results every clock cycle

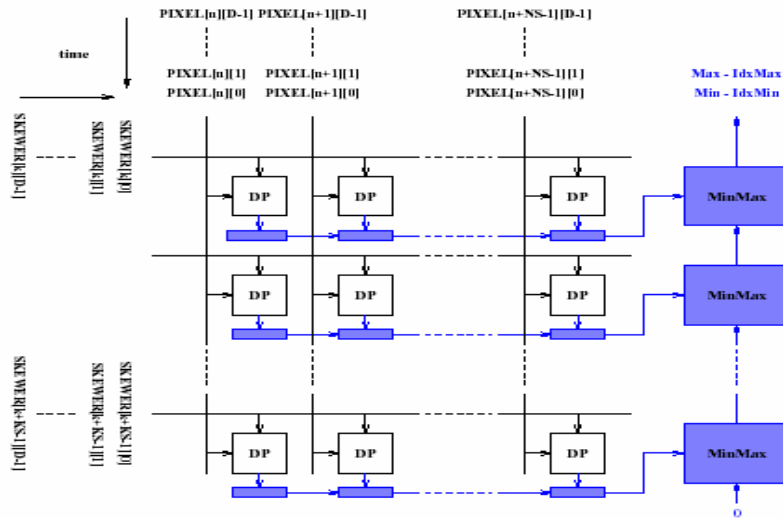
Poly Phase Filter Implementation



Pixel Purity Index

- Remote sensing application for analyzing hyperspectral images
- Low resolution imagery
- Single pixel covers several different materials
- Generate vectors through image
- Dot-product computations
- Hand coded
 - 22.5% of chip, 25MHz, 6 weeks
 - Results every clock cycle
- Streams-C
 - 100% of chip, 15MHz, 5 days
 - Results every clock cycle

PPI Algorithm Architecture



CS6813: Research Seminar on Reconfigurable Hardware



13

K-means Clustering Algorithm

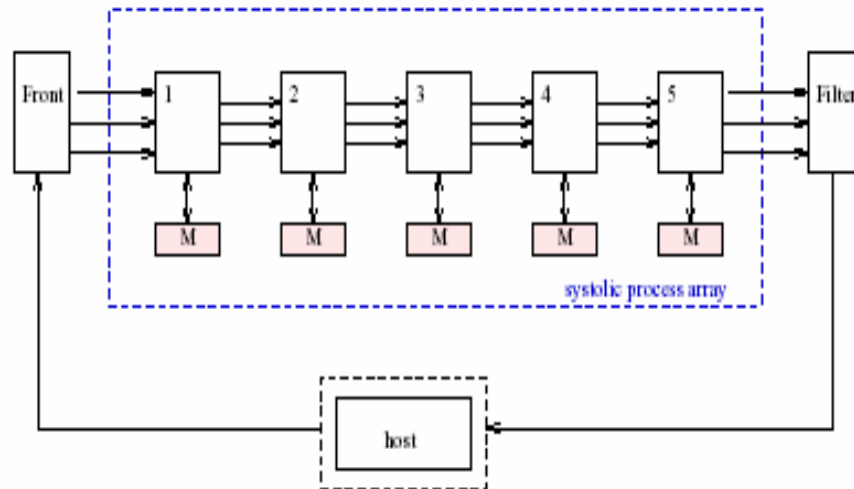
- Represent image using smaller number of pixels
- Cluster pixels together
- Phases of computation
 - Pixel distance computation to class center
 - Accumulator update
 - Center update
- Hand coded
 - 9.4% of chip, 20MHz, 2 weeks
 - Results every clock cycle
- Streams-C
 - 14% of chip, 20MHz, 1 day
 - Results every clock cycle

CS6813: Research Seminar on Reconfigurable Hardware



14

K-means Hardware Implementation



Results

Xilinx 4036 Architecture

	Streams-C VHDL			Handcoded VHDL		
	Area %	Speed MHz	Time wks	Area %	Speed MHz	Time wks
CE	55	20	1/2	18	40	4
PPF	37	40	1/2	27	40	2
PPI DPs 4x2	100	15	1	22.5	25	6
Kmeans	14	20	1/4	9	20	1-2

Xilinx FPGA Comparison

	Logic blocks	CLBs
Virtex 1000	27,648	96x64 = 6,144
4036	3,078	1,296

Virtex 1000

Virtex V1000 Architecture

Streams-C VHDL Handcoded VHDL

	Area %	Speed MHz	Area %	Speed MHz
CE	3	40	1	40
PPF	1	40	1	40
PPI DPs 4x2	6	40	2	45
Kmeans	< 1	40	< 1	40

Summary

- Streams-C versus VHDL
 - Two to three times increase in area
 - One half clock speed
 - Five to ten times the development time savings
 - Greatly dependant on algorithm
 - No support for on-chip SRAM

Concerns

- “An output every clock cycle is expected”
- Couldn’t find source listings to compare
- No information on the relative talents of the developers
- Did the same person develop all the HDL and another person develop the Streams-C?
- Did one person develop both implementations for a single application?
- If so, which implementation was developed first?
- Designs seem to be relatively small

Concerns (continued)

- Related work section weak
- C programmers are used to working with “unlimited” resources
- Did not convince me to switch to Streams-C