AFPEfD & IoH PR&SARPF

Gary J. Minden
The University of Kansas
Information and Telecommunications Technology Center
2291 Irving Hill Road
Lawrence, KS  66045
gminden@ittc.ukans.edu
• A — We’re only doing one, it’s Engineering, not Science
• FPE — Functional Programming Environment
  • Do it right, do it with rigor
• D & I — Design and Implementation
  • We’re concerned with how engineers design and produce product
• HP — High Performance
  • We’re speed freaks
• R — Radio
  • We do like to communicate (even though once an engineer always a nerd)
• SAR — Synthetic Aperture Radar
  • A reasonably hard problem
• PF — Processing Functions
  • We want to do real work
Motivation

- **Defense Environment**
  - 30-year military vs. 18-month commercial product cycles
  - New platforms equipped with 10-year old technology at time of deployment
  - Numerous information technology subsystems improvement cycles
  - Proprietary, stovepiped systems
  - Highly volatile COTS marketplace with little interest in defense problems

Provide the Best!
Adaptive Computational Systems

- **Micro-Programmable Computers**
  - Emulate target machines
  - “Configurable”
    - User-level instructions
    - Control flow
  - Fixed hardware functions
  - Fixed interconnect topology
  - Tough Problem!

- **Module-level Configuration**
  - DEC M-Series Modules
  - Military standard modules
    - 10s of gates/module
    - Mix and Match
    - Fixed interconnect
    - No reconfiguration

  - ~10^6 gates/chip
  - “Sea of Gates”
  - Config hardware architecture
  - Config interconnect
  - Definable instructions
  - Definable system function
  - Dynamically reconfigurable functionality (per cycle)
  - Fine grained performance allocation (computing by the square inch)
ITTC — Approach

- **Given**
  - Need for multiple and flexible radio communications systems
  - Need for “see-thru” surveillance of imaging radars
- **Implement a robust design language for FPGAs design**
- **Implement and evaluate radio functions on FPGAs**
  - High Order Modulation/Demodulation
  - Beam-steered antenna systems
  - Forward Error Correction
  - Compression/Decompression
- **Implement and evaluate SAR functions on FPGAs**
  - Not a “flow-thru” algorithm
  - Significant memory required
  - Memory access derived from ephemeris data
The Functional Programming Language

- Application-oriented language, compiler, and verifier
- Dataflow applications at high abstraction levels
- The Functional nature of the language is important
  - Problems in the application areas are easily expressed by users at a high level of abstraction and utilizing dataflow paradigms
  - Functional languages are amenable to symbolic and automatic manipulation to:
    - Convert from high abstraction levels to implementation levels
    - Optimize conversions based on different criteria and target architectures
    - Elicit through formal methods properties concerning an application and their implementation

- Formal specifications of program properties
  - Use formal specifications during the compilation process to meet user established criteria, and the ability to make specific statements about the properties of programs and implementations
Radio Communications

- Involves significant digital signal processing.
- Wide bandwidth intermediate frequency (IF) for high capacity systems
- Digital beam forming for receivers and transmitters for interferer cancelation and directionality requires high computation rates
- Spread spectrum for anti-jamming, low probability of intercept and multiple access applications
- Multiple coding mechanisms for error control, privacy, and compression
- Multiple modulation mechanisms
- Channel adaptation with rapid time-varying behavior
Synthetic Aperture Radar Processing

- A second class of applications
- High data rates
- Large memories to store temporary return information
- In-line processing decisions based on platform behavior and imaged terrain
Single Channel Radio Receiver

Today

• Single, processing specific integrated circuits
• Functionality determined by largest market
• Chips not necessarily designed for compatibility
• Difficult configuration steps

With FPE....

• Processing tuned to application
• Parallel execution when needed
• Configuration at design time
• Computation tuned as a system
Today's Design Paths

- SPW
- VHDL
- XNF

- Raw Bits
- Load Device

Today
- Long chain from design change to hardware
- Tools implement How not What
- Difficult to achieve speed, density
Functional Programming Environment

Criteria
- Power
- Speed
- Density
- Precision
- Architecture

Functional Description
- What (Specification)
- High-level How

Optimizations
- Algorithms
- End-product criteria

Objectives
- Faster
- Better algorithms
- More flexibility
- Multiple implementations
- “Run-time support”

Intermediate Form

Load Device
ITTC — Adaptive Computational Systems

• New Ideas
  • A Functional Programming Environment to express and implement significant radio and synthetic aperture radar processing functions at a high level of abstraction.
  • Formal specifications included for high assurance program transformations and optimizations.
  • Advanced radio and SAR processing functions implemented on FPGA based systems.
  • A run-time environment for managing allocation of FPGA resources.

• Impact
  • High level functional languages increase programmer productivity and enable automatic transforms, optimizations, and mapping to multiple FPGA architectures.
  • Formal specifications track the engineer’s intent through compilation, transformation, and optimization to implement robust systems.
  • Advanced radio and SAR processing functions used for high performance defense communications and image processing tasks.