AI and SDR: Software Meets Hardware Again…

Manuel Uhm
Director, Silicon Marketing
Chair of the Board, Wireless Innovation Forum (SDR Forum v2.0)

Jason Vidmar
Sr. System Architect – MILCOM / SATCOM / Machine Learning

© Copyright 2019 Xilinx
Figure 1: How successive generations of SDRs have come to dominate the radio industry and will continue to evolve.

Key semiconductor technology drivers:
- Moore’s Law
- FPGAs
- RFICs
- Analog/Digital Integration

Source: Manuel Uhm, Software-Defined Radio: To Infinity and Beyond, Military Embedded Systems, October 2016
Key semiconductor technology drivers:
- Moore’s Law
- GPUs
- FPGAs
- ASICs
SDR & AI Payload Convergence

Multi-Mission
Situationally
Aware
Payload:
Enabled by SDR and AI Technology
**End of the Line for Processor Performance?**

**DENNARD SCALING**
Power Density Rises

**MOORE’S LAW**
End of “PPA” Improvement

**AMDAHL’S LAW**
Multicore Hits Limit

---

**40 Years of Processor Performance**

- **CISC**
  - 2x / 3.5yrs
  - (22%/yr)

- **RISC**
  - 2x / 1.5yrs
  - (52%/yr)

- **End of Dennard Scaling**
  - 2x / 6yrs
  - (12%/yr)

- **End of Amdahl’s Law**
  - 2x / 20yrs
  - (3%/yr)

---

**Moving Forward: Domain-Specific Architectures (DSAs)**

Source: John Hennessy and David Patterson, *Computer Architecture: A Quantitative Approach*, 6/e. 2018
Evolving Processor Landscape

- **FPGAs**
- **ASICs**
- **ASSPs**
- **ACAPs (Domain Specific Architecture)**
- **General Purpose Processors**

**Why ACAP?**

- **Performance/Power Efficiency**

Number of Applications
The Adaptive Compute Acceleration Platform

**ADAPTIVE**
- Diverse Workloads in Milliseconds
- Future-Proof for New Algorithms

**COMPUTE ACCELERATION**
- Multi-core Processing System
- Programmable Logic
- DSP (Vector-based & Fabric-based)

- Scalar Engines
- Adaptable Engines
- Intelligent Engines

**PLATFORM**
- Development Tools
- HW/SW Libraries
- Run-time Stack
- SW Programmable Silicon Infrastructure

Enabling Data Scientists, SW Developers, HW Developers
Hardware Adaptable: Accelerating the Whole Application

Scalar, Sequential & App Processing
Flexible Parallel Compute, Data manipulation
ML & Signal Processing Vector, Compute Intensive

Heterogeneous Processing For Tactical Edge Systems
(Example Applications)

- Adaptive Beamforming
- AJ
- Tactical Networking
- SAR Backprojection
- Spectrum Processing
- Machine Learning

Robust Device & Run-time Security

Delivering Deterministic Performance & Low Latency
Versal ACAP: A Platform for Software *and* Hardware Developers

Fully Software Programmable
with Hardware Design Path

- User Application
  - C, C++, Python
- Frameworks
- Runtime
- OS • Drivers
- IP • Libraries

### Software Platform
- Scout
- Vivado

### Hardware Platform
- Evaluation & Deployment Boards
- Versal ACAP Device & Integrated Shell

© Copyright 2019 Xilinx
Possible Platform Example: Multi-Mission Situationally Aware UAV Payload with Versal ACAP

<table>
<thead>
<tr>
<th>Multi-Mission Applications: Comms, Radar, SIGINT, EW</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frameworks</td>
</tr>
<tr>
<td>TensorFlow</td>
</tr>
<tr>
<td>PyTorch</td>
</tr>
<tr>
<td>Xilinx Runtime (XRT)</td>
</tr>
<tr>
<td>VxWorks</td>
</tr>
<tr>
<td>Integrity</td>
</tr>
<tr>
<td>Cytos</td>
</tr>
<tr>
<td>xfopenCV</td>
</tr>
<tr>
<td>DSPlib</td>
</tr>
<tr>
<td>ML Overlay</td>
</tr>
<tr>
<td>Scalar Engines</td>
</tr>
<tr>
<td>Adaptable Engines</td>
</tr>
<tr>
<td>AI Engines</td>
</tr>
</tbody>
</table>

Versal ACAP Eval Board

VERSAL ACAP
Versal ACAP Roadmap

AI Core
AI Inference Throughout

Prime
Broadest Application

AI Edge
Lowest power AI

Premium
112G SerDes
600G Cores

AI RF
AI with Integrated RF

HBM
Memory Integration
Advanced SDR: Technologies and Challenges
Trends in SDR Pushing the Compute Boundary

[CAPACITY]
5G 100X Complexity vs. 4G

- 100X Peak Data Rate
- 100X User Experienced Data Rate
- 10X Area Traffic Capacity
- 10X Network Energy Efficiency
- 10X Massive IoT
- 10X Ultra-reliable & Low Latency
- 3X Spectrum Efficiency
- 100X Mobility

Rise of Deep Learning (Dawn of Next Wave of AI)

- Rise from AlexNet to AlphaGo Zero

[RESILIENCY]
Operations in Contested Spectrum


Enabling Technologies

> Direct-RF / High-IF Sampling Data Converters
> Array Antennas
> Compute Optimizations for Deep Learning

Deep Learning Classification

Image Input

Non-image Input (RF)

Classification Result

“Cat”
“Dog”
“Bird”...

“QPSK”
“BPSK”
“8PSK”...

Controlled Reception Pattern Array (CRPA)
beam patterns.
(source: gpsworld.com)

Animation credit: Philip Leone, Univ. of Sydney. Presentation.
Advanced SDR: Compute Comparisons

Space Time Adaptive Processing
Application Example: Beamforming/Nulling (Comms / Anti-Jam)

Deep Learning Inference (Conv. Nets)
Application: Modulation Recognition, Waveform Classification

References: "Implementing a Real-Time Beamformer on an FPGA Platform." Xilinx.
Xcell Journal 60.
See also: Xilinx WP452 “Adaptive Beamforming for Radar: Floating-Point QRD+WBS in an FPGA”

References: "Applied Deep Learning - Part 4: Convolutional Neural Networks",
Towards Data Science (blog).

\[ W = R_{xx}^{-1} \cdot b \]
Steering Vector
Covariance Matrix Decomposition: QR, Cholesky, etc.

\[ Y = X \cdot K \]

Complex-valued
Higher Precision Desirable (e.g., SPFP32)
Typical FLOPS: up to MFLOPS per Decomposition

Real-valued
Lower Precision Desirable (e.g., INT8)
Typical OPS: 7.6 GOPS (Resnet-50 unpruned)
### Real Data Types

<table>
<thead>
<tr>
<th>Data Type</th>
<th>MACs / Cycle (per core)</th>
</tr>
</thead>
<tbody>
<tr>
<td>32x32 SPFP</td>
<td>8</td>
</tr>
<tr>
<td>32x32 Real</td>
<td>8</td>
</tr>
<tr>
<td>32x16 Real</td>
<td>16</td>
</tr>
<tr>
<td>16x16 Real</td>
<td>32</td>
</tr>
<tr>
<td>16x8 Real</td>
<td>64</td>
</tr>
<tr>
<td>8x8 Real</td>
<td>128</td>
</tr>
</tbody>
</table>

### Complex Data Types

<table>
<thead>
<tr>
<th>Data Type</th>
<th>MACs / Cycle (per core)</th>
</tr>
</thead>
<tbody>
<tr>
<td>32x32 Complex</td>
<td>2</td>
</tr>
<tr>
<td>32x16 Complex</td>
<td>4</td>
</tr>
<tr>
<td>16x16 Complex</td>
<td>8</td>
</tr>
<tr>
<td>16 Complex x 16 Real</td>
<td>16</td>
</tr>
</tbody>
</table>

### AI Engine: Multi-Precision Math Support

#### Optimized For:

- **Linear Algebra**
  - Matrix-Matrix Mult
  - Matrix-Vector Mult

- **Convolution**
  - FIR Filters
  - 2-D Filters

- **Transforms**
  - FFTs/IFFTs
  - DCT, etc
AI Engine: Scalar Unit, Vector Unit, Load Units and Memory

32-bit Scalar RISC Processor

Local, Shareable Memory
• 32KB Local, 128KB Addressable

Instruction Parallelism: VLIW
7+ operations / clock cycle
• 2 Vector Loads / 1 Mult / 1 Store
• 2 Scalar Ops / Stream Access

Highly Parallel

Data Parallelism: SIMD
Multiple vector lanes
• Vector Datapath
• 8 / 16 / 32-bit & SPFP operands

Up to 128 MACs / Clock Cycle per Core (INT 8)
8 FLOPs / Clock Cycle (32SPFP)
AI Engine: Terminology

Versal ACAP

Scalar Engines
- Arm Dual-Core Cortex™-A72 Application Processor
- Arm Dual-Core Cortex-R5 Real-Time Processor
- Processing System
- Platform Management Controller

Adaptable Engines
- Versal Adaptable Hardware
  - DSP Engines
  - Block RAM
  - UltraRAM
  - Accelerator/ROM

Intelligent Engines
- 1GHz+ VLIW / SIMD vector processor

AI Engine Array

AI Engine Tile

AI Engine

Interconnect

ISA-based Vector Processor
- AI Vector Extensions
- 5G Vector Extensions

Local Memory

Data Mover

Scalar Unit
- Scalar Register File
- Scalar ALU
- Non-linear Functions
- Load Unit A
- Load Unit B
- Store Unit
- Memory Interface

Vector Unit
- Vector Register File
- Fixed-Point Vector Unit
- Floating-Point Vector Unit
- Instruction Fetch & Decode Unit
- Stream Interface

Stream Interface

1GHz+ VLIW / SIMD vector processor

© Copyright 2019 Xilinx
AI Inference Mapping on Versal™ ACAP
Program Directly From High-level ML Frameworks

> Custom memory hierarchy
  > Buffer on-chip vs off-chip: Reduce latency and power

> Stream Multi-cast on AI interconnect
  > Weights and Activations
  > Read once: reduce memory bandwidth

> AI-optimized vector instructions (128 INT8 mults/cycle)
AI Engine Delivers High Compute Efficiency

> Adaptable, non-blocking interconnect
  >> Flexible data movement architecture
  >> Avoids interconnect “bottlenecks”

> Adaptable memory hierarchy
  >> Local, distributed, shareable = extreme bandwidth
  >> No cache misses or data replication
  >> Extend to PL memory (BRAM, URAM)

> Transfer data while AI Engine Computes

Vector Processor Efficiency

<table>
<thead>
<tr>
<th></th>
<th>ML Convolutions</th>
<th>FFT</th>
<th>DPD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Block-based</td>
<td></td>
<td></td>
<td>Volterra-based</td>
</tr>
<tr>
<td>Matrix Multiplication</td>
<td>(32×64) × (64×32)</td>
<td></td>
<td>forward-path DPD</td>
</tr>
<tr>
<td></td>
<td>95%</td>
<td>80%</td>
<td>98%</td>
</tr>
</tbody>
</table>

© Copyright 2019 Xilinx
Summary

> The evolution of processing for AI is following a similar track to SDR where hardware and software need to be tightly coupled

> The drive for more Capacity, Autonomy and Resiliency in advanced SDRs carry high compute demands and mixed precision processing capabilities

> Moore’s Law is running out of steam which means the goal of a SWaP-friendly multi-mission situationally aware payload requires advancements in processing beyond just process technology

> ACAPs are a response to this new reality

Adaptable. Intelligent.

THANK YOU!

Contact Info:
manuelu@xilinx.com
jasonv@xilinx.com