All Programmable Devices: Not Just an FPGA Anymore

Vidya Rajagopalan
Corporate Vice President, Processing, Software, Systems and Applications
Xilinx

December 11, 2013

Then and Now

More Than Moore!

1985

- 100 Logic Cells
- 100 Flip Flops
- 74 IO
- Supply Voltage: 5V
- 2.5 um process

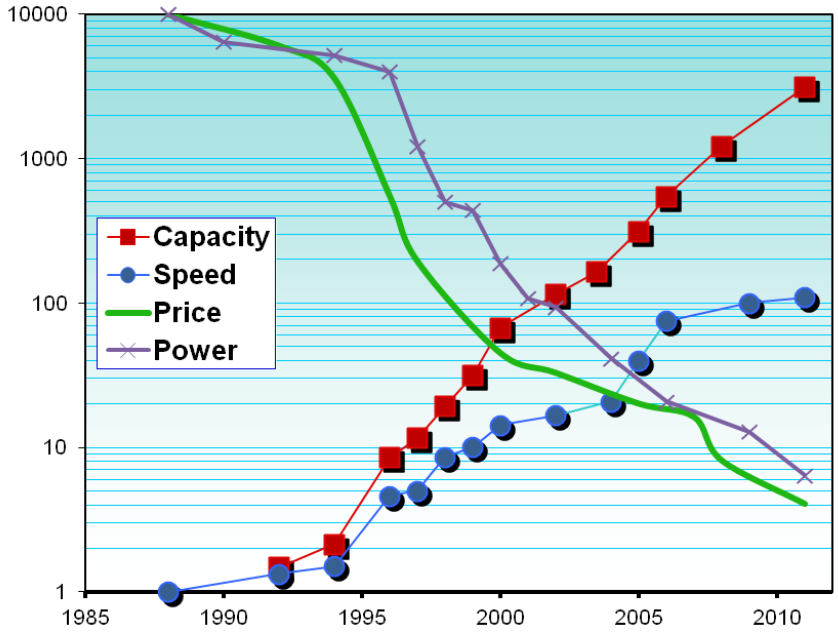
2012

- 2.15M Logic Cells (~21K x)
- 2.44M Flip Flops (~24K x)
- 1200 IO (16 x), 36 SERDES
- Supply Voltages: 8 supplies: 3.3V to 1V
- 28nm process (Scaling ~90x)

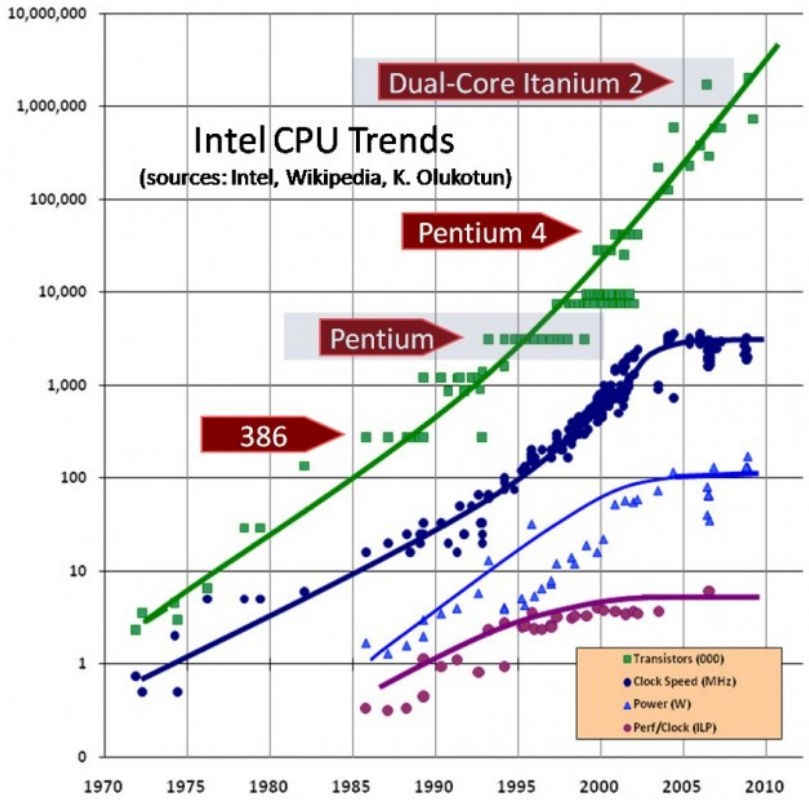
- 46Mb Block RAM, 21Mb LUT RAM
- 2160 DSP Blocks

Over Time: Moore's Law at Work

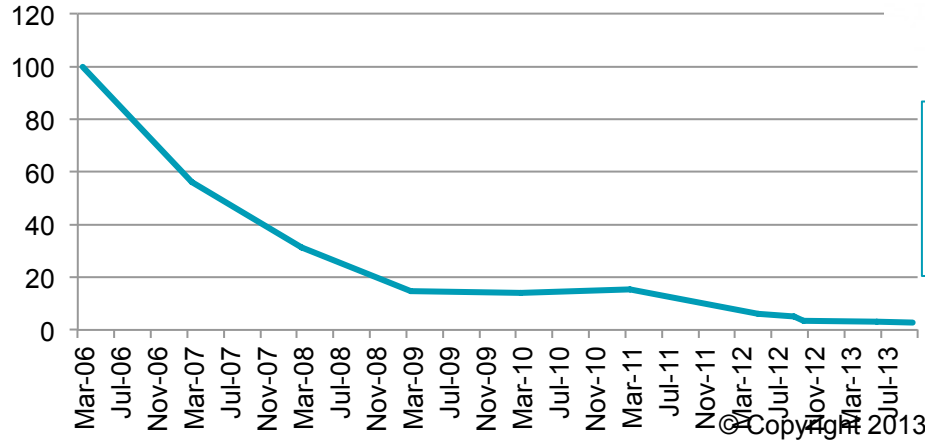
FPGA Trends over Time



CPU Trends over Time

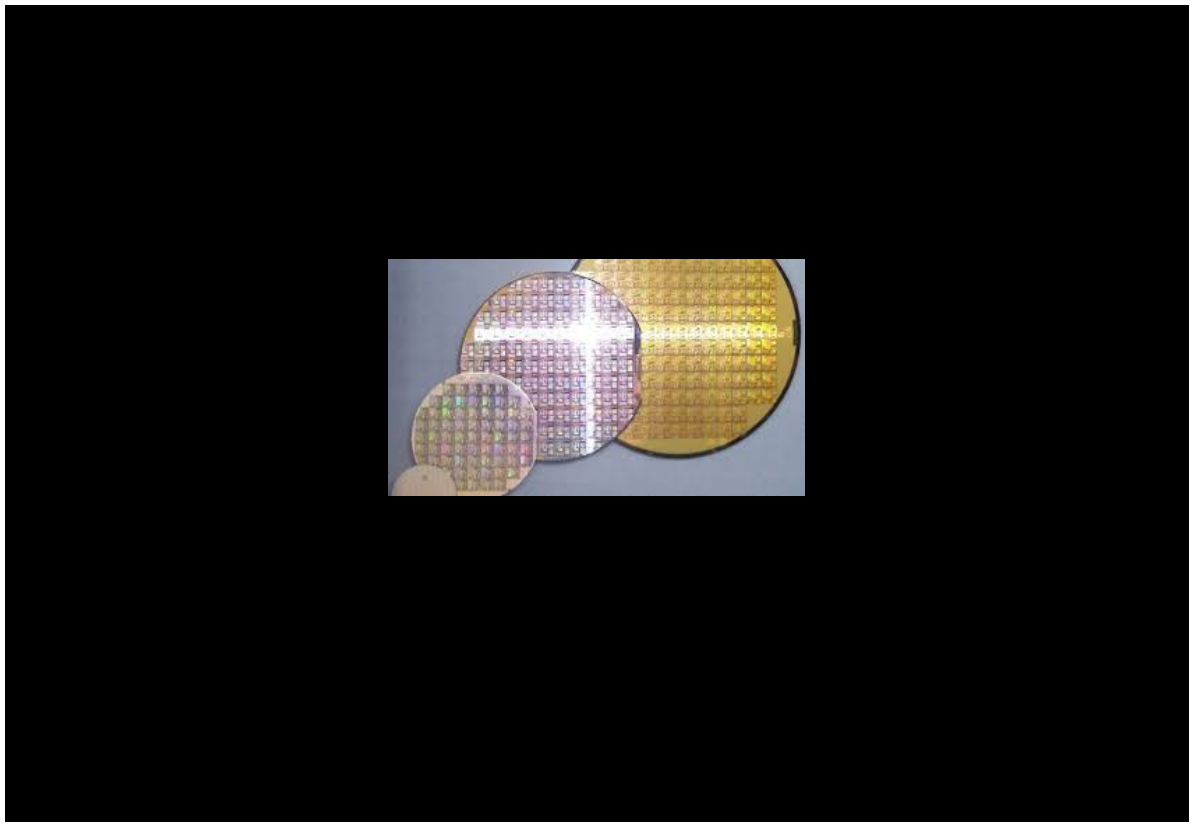


FPGA P&R Runtime



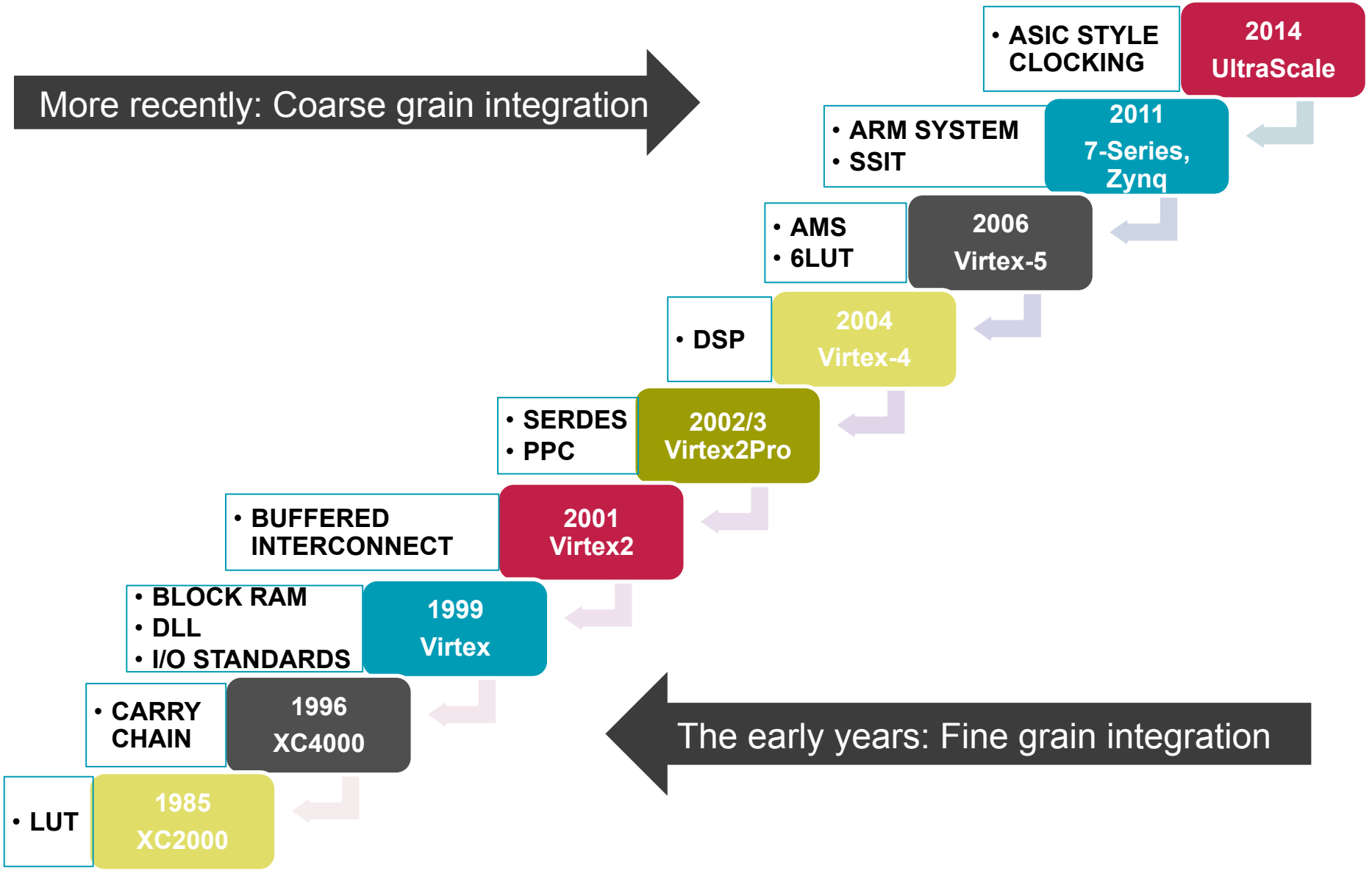
P&R runtime improved for equivalent design by 40x,
Device sizes increased by 6x

Wafer Prices Going Up

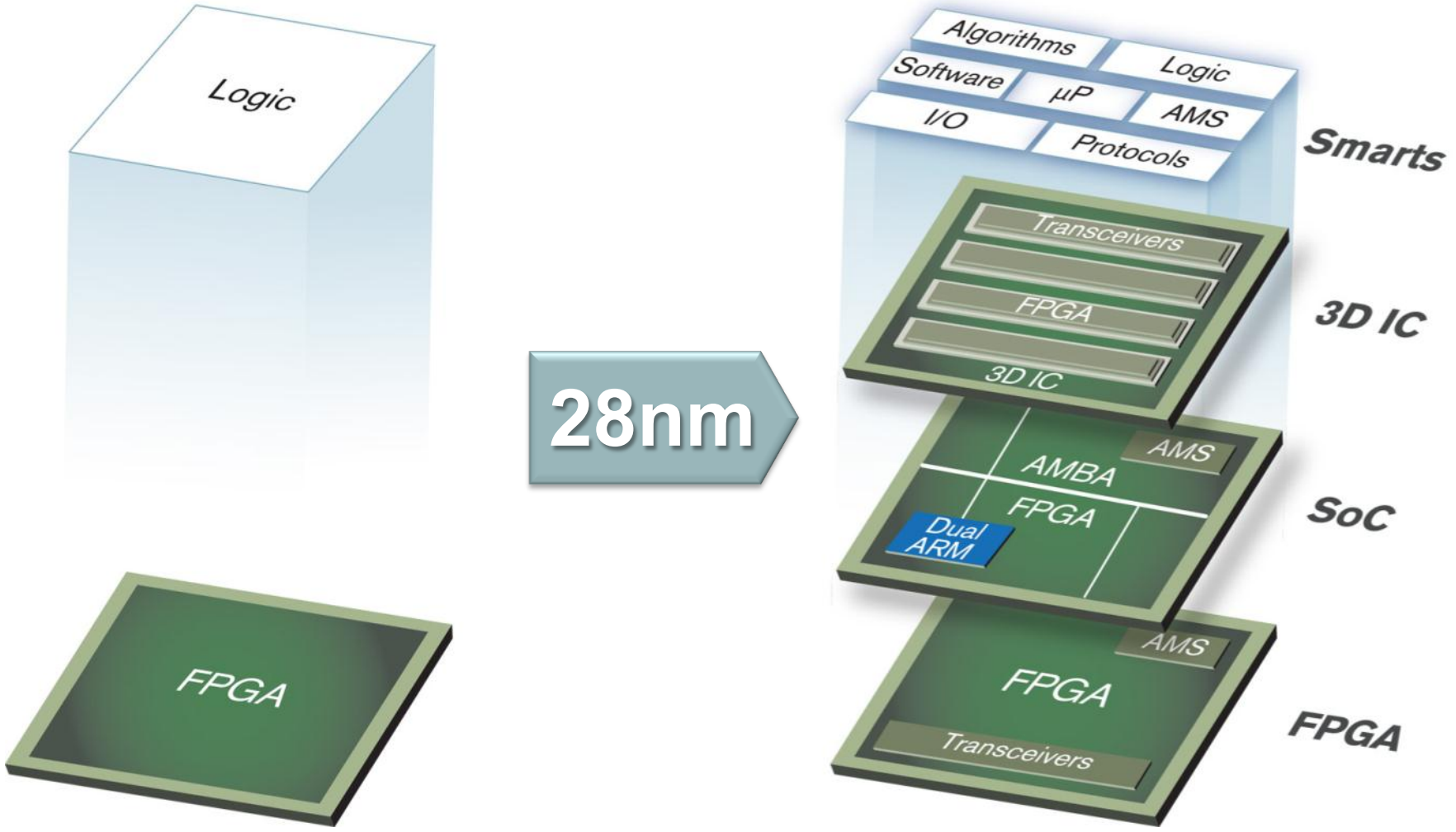


Fewer and fewer people can afford to build chips

More than Moore's Law Behind the Scenes



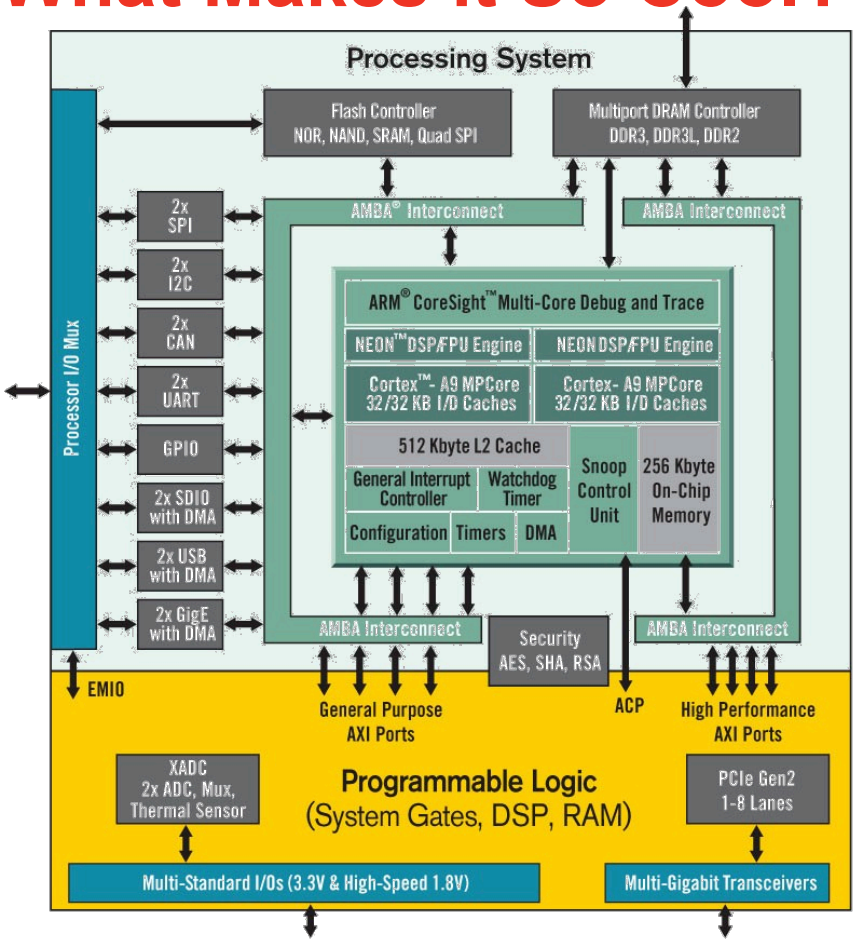
Evolving Towards Heterogeneous Multi-Core



Programmable Logic Devices
Enables Programmable "Logic"

All Programmable Devices
Enables Programmable "Systems Integration"

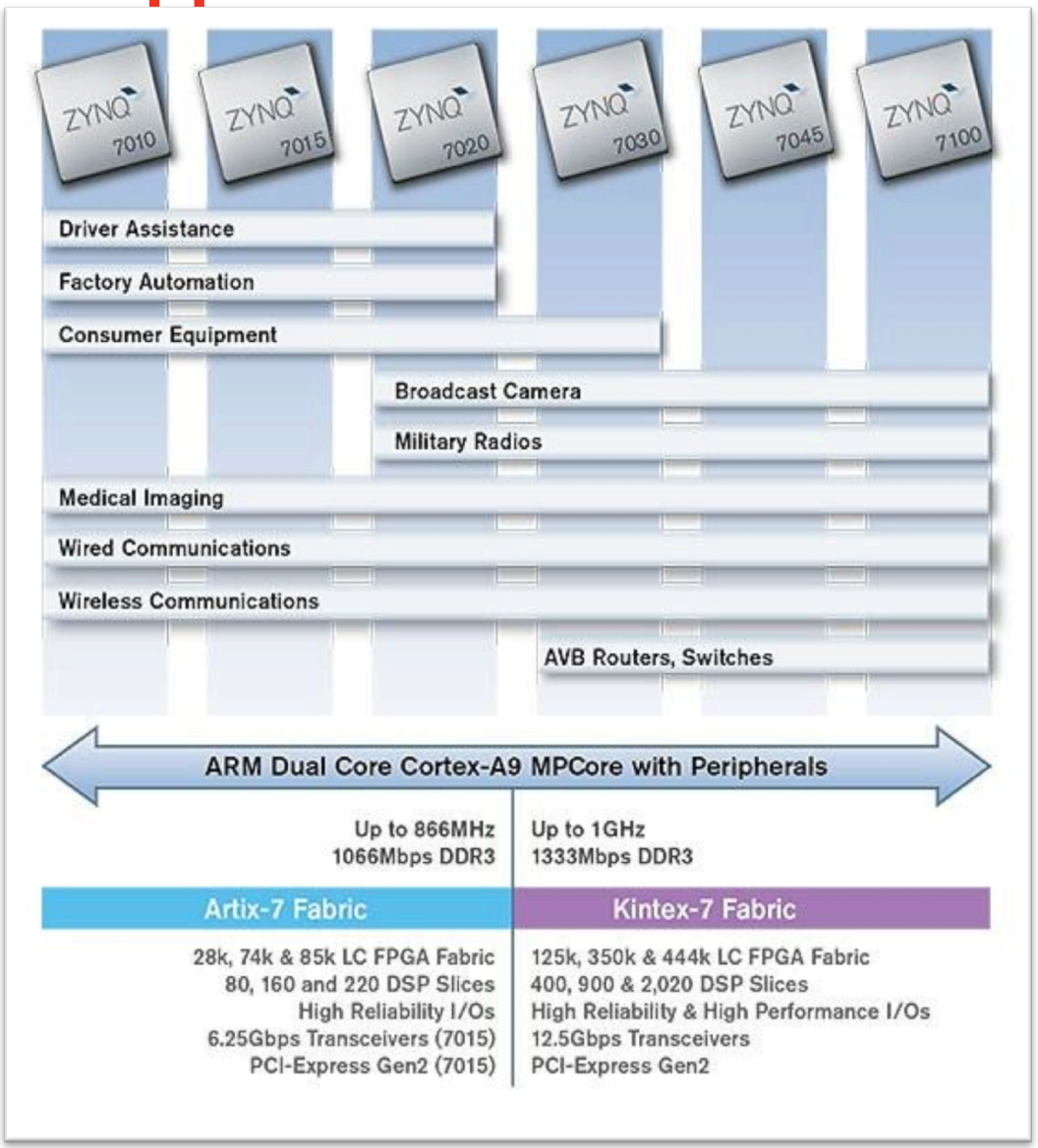
Zynq-7000 All Programmable SoC
What Makes it So Cool?



- **Boots up like a processor**
  - Runs software code
- **CPU master controls fabric**
  - Configuration/Reconfiguration
- **Ease of use of a processor**
- **2 P's of FPGAs**
  - Programmability
  - Parallelism
- **Making possible**
  - Near ASIC cost and performance
  - Build your own SoC
  - Software Acceleration Paradigms

Zynq® - Democratizing SoCs One Chip at a Time

Zynq-7000 All Programmable SoC Family Applications Served



- **Existing and emerging applications**
  - Image Processing
  - Signal Processing
  - Database Search (memcached)
  - etc.
- **Common themes include**
  - Data Parallel Applications
  - Signal Processing Applications
  - High Bandwidth Real Time Processing
  - Custom Interfaces

Target for Acceleration: Data Parallel Applications

Current Frame

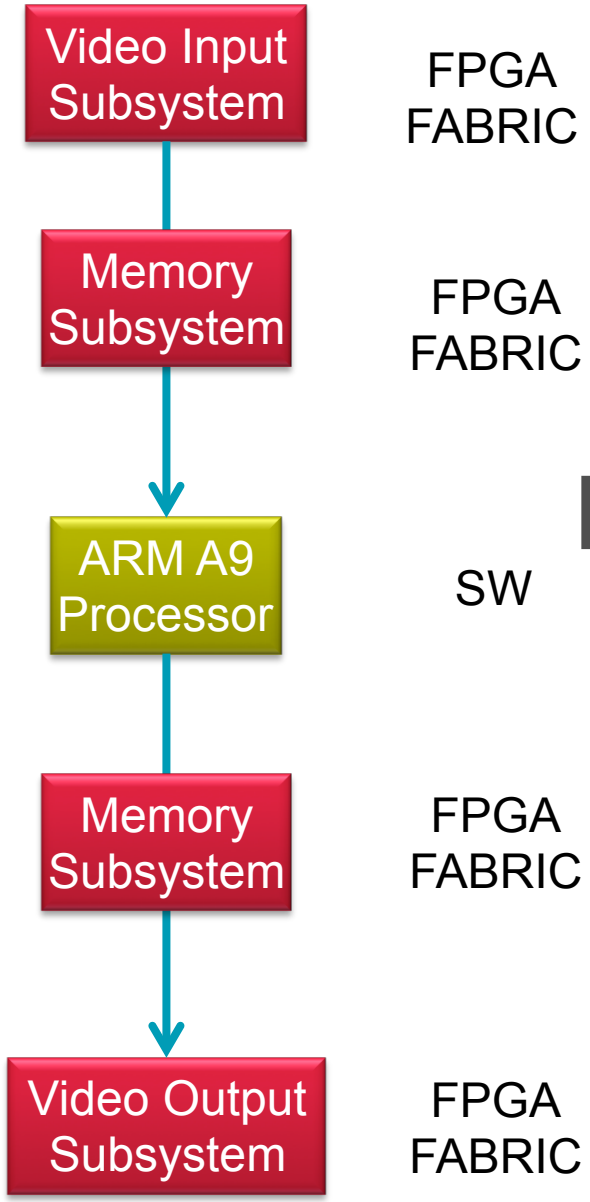
Previous Frame

OpenCV Motion Detection Algorithm

Output Frame

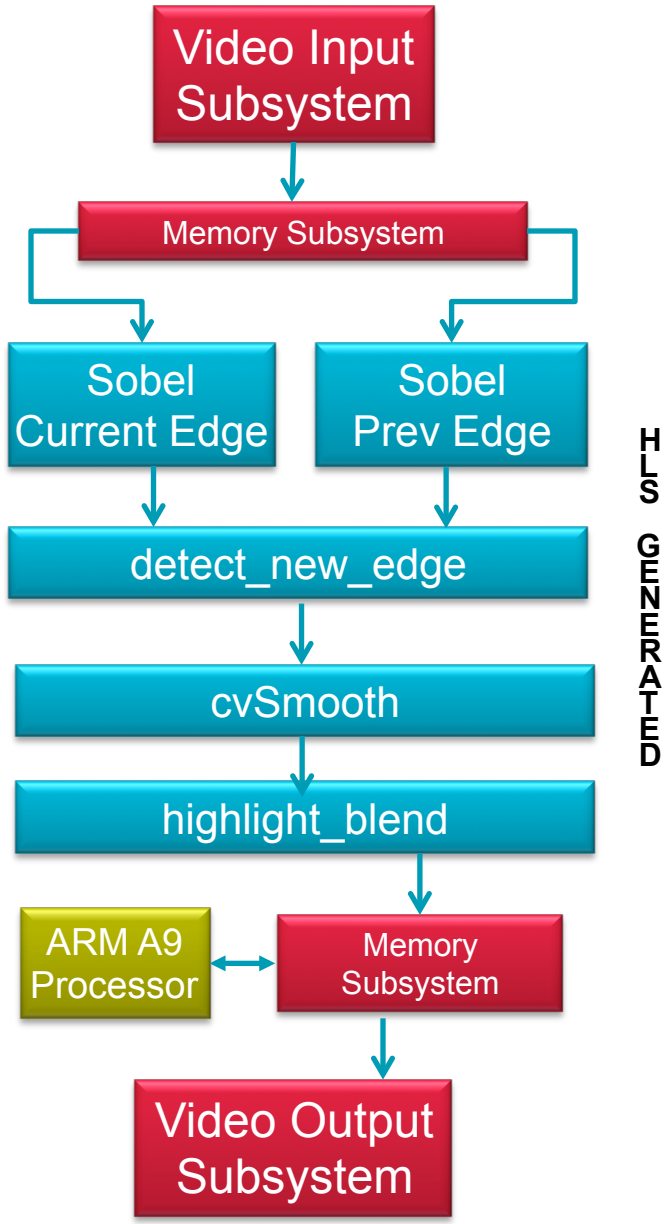
Detected New Car

Un-optmized Application on Zynq

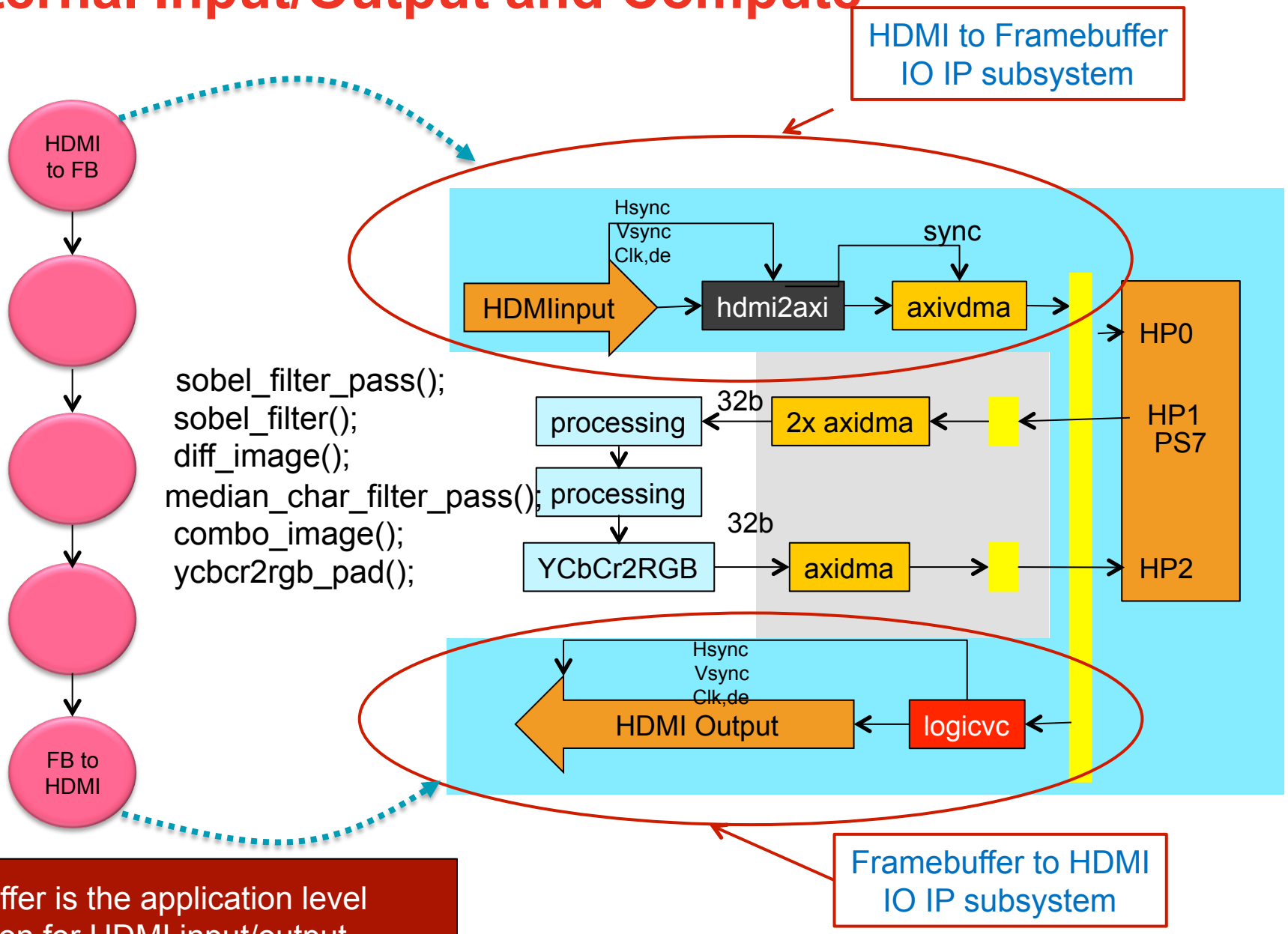


~180x speedup

Optimized Application on Zynq

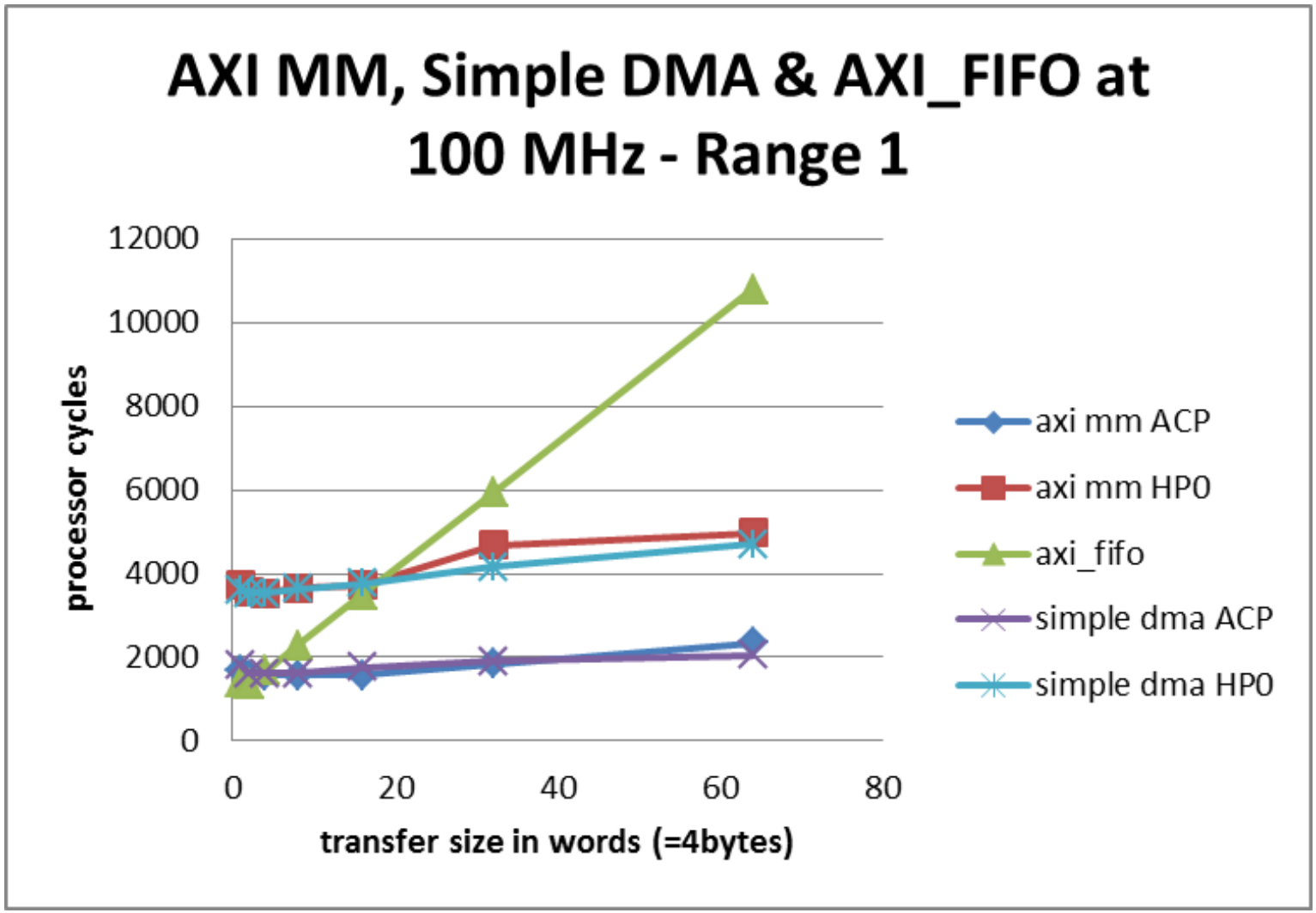


External Input/Output and Compute

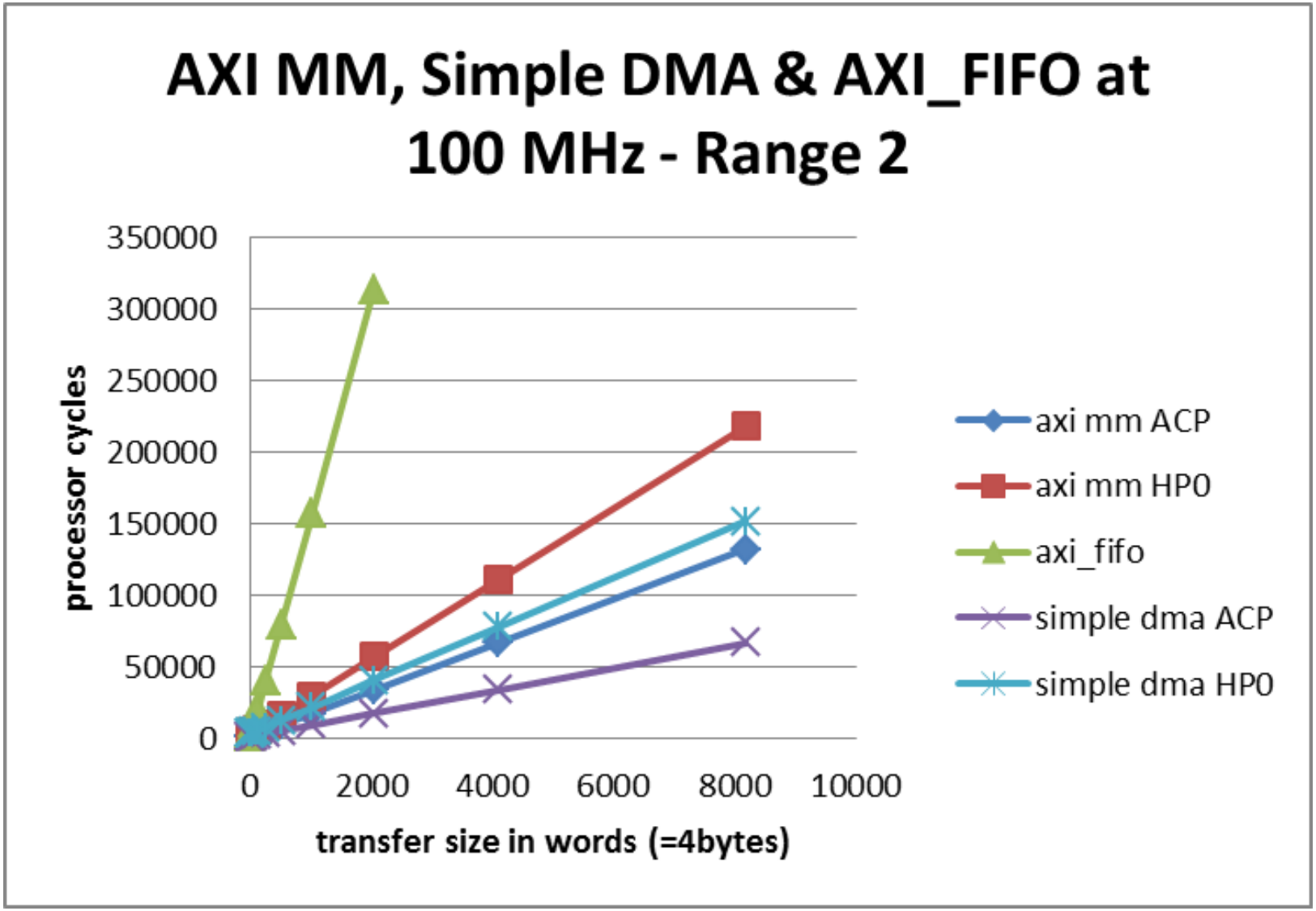


Frame Buffer is the application level abstraction for HDMI input/output

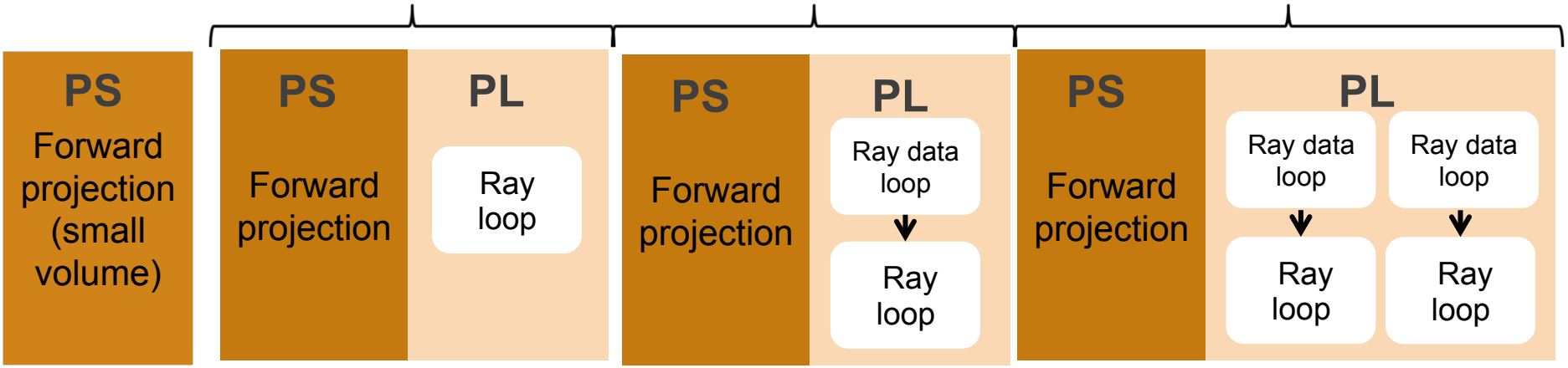
Effectiveness of Data Movers: Small Data



Effectiveness of Data Movers: Big Data



Accelerating Forward Projection (Medical Imaging)



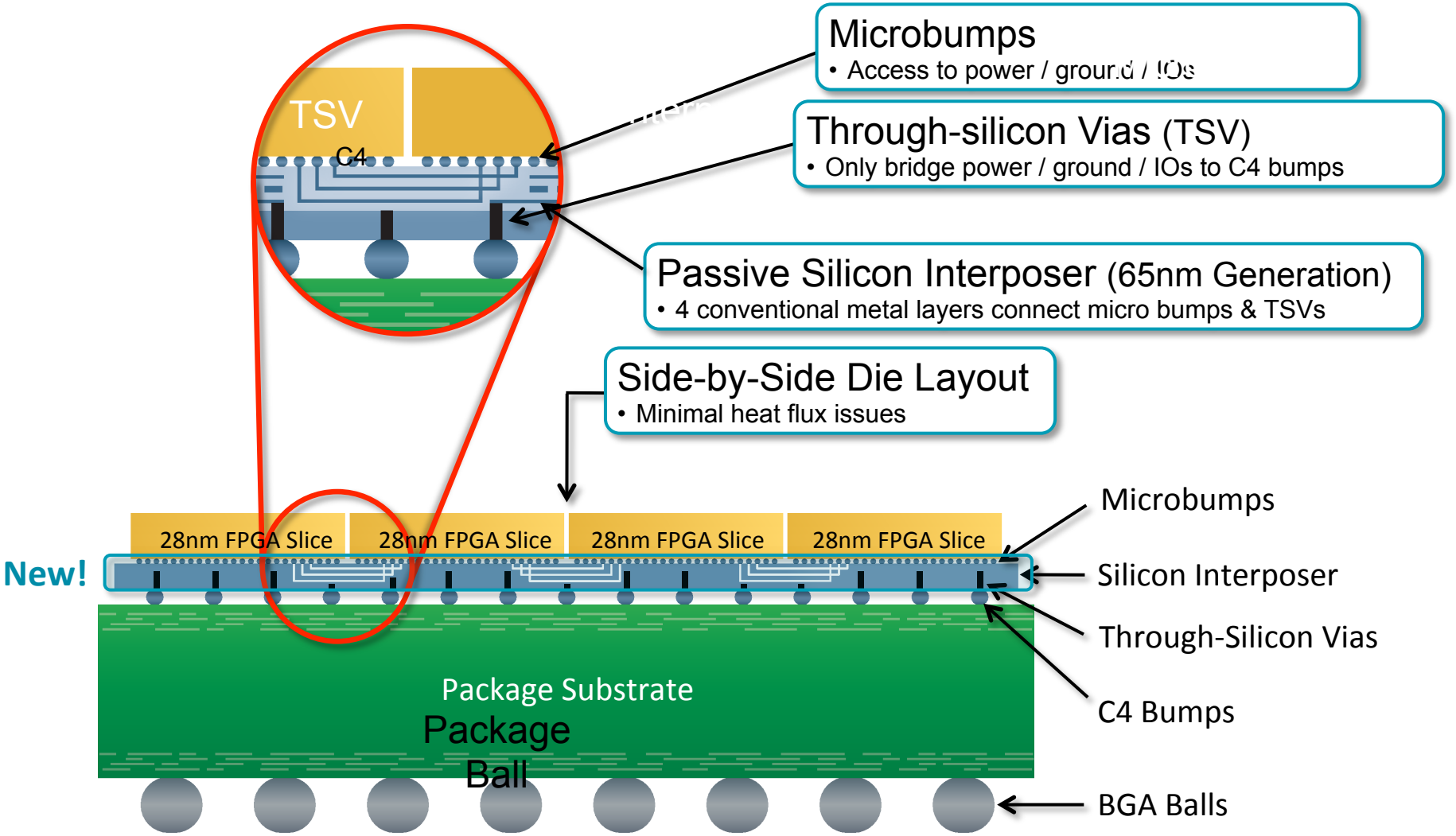
- Initial partitioning
- Move more computation to PL
- Reduce data motion overhead by direct connection
- Replicate accelerator for more performance

~0.9 sec/view → **~58x Speedup** → **65 views/sec**

ARM @ 800MHz, Accelerators @ 100MHZ

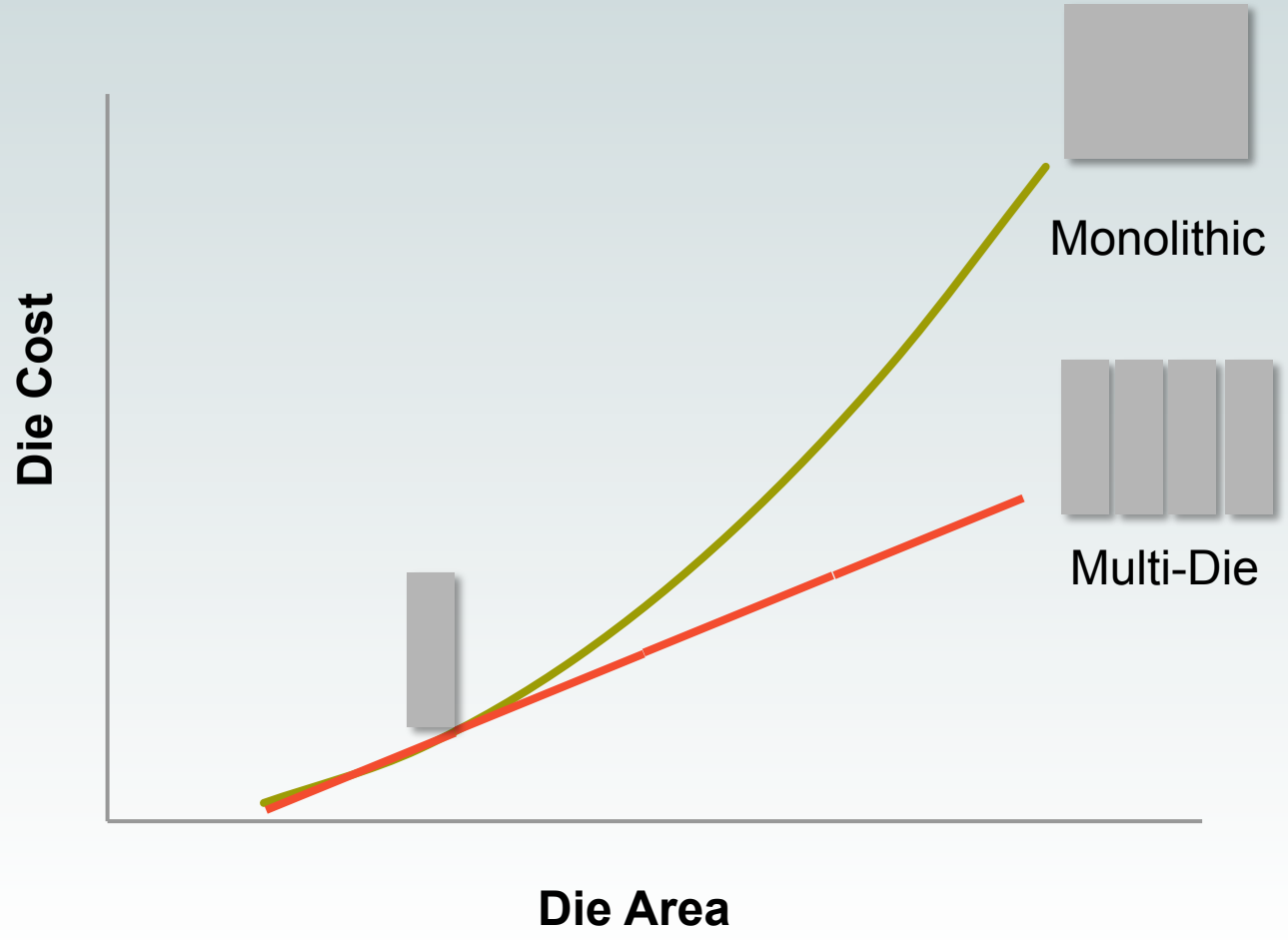
C source code ported to FPGA using Vivado-HLS

SSIT Makes Integration Possible

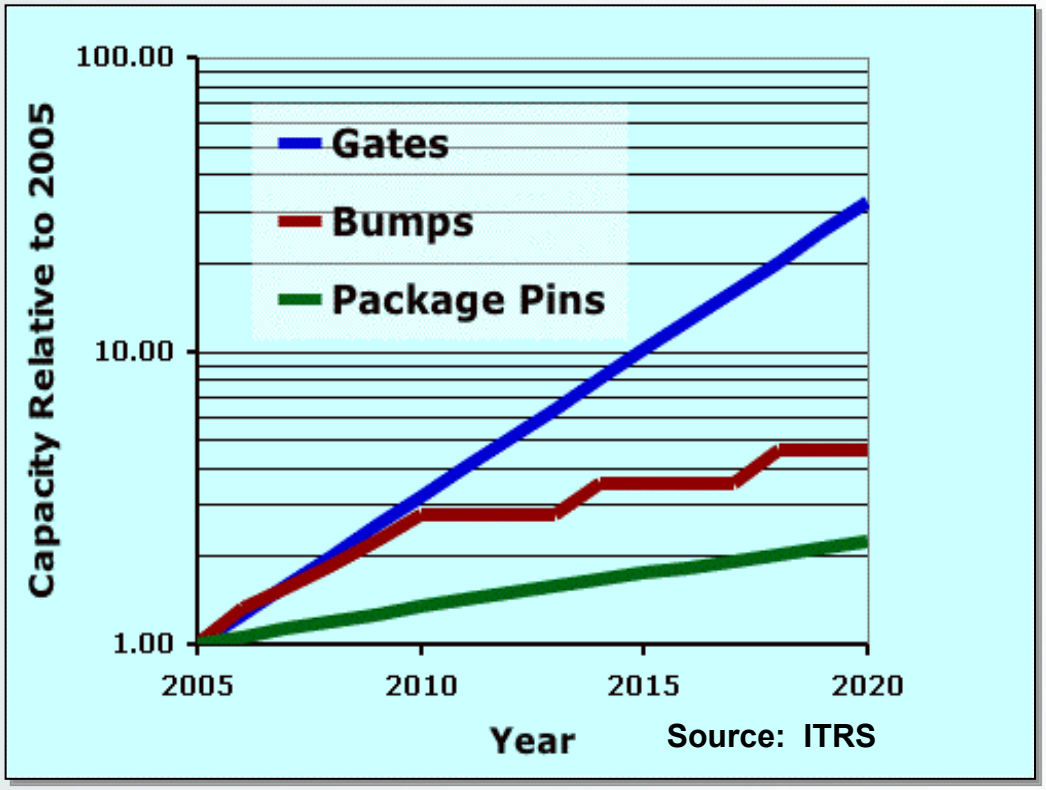


Cost Comparison: Monolithic vs. Multi-Die

"Moore's Law is really about economics" *– Gordon Moore*



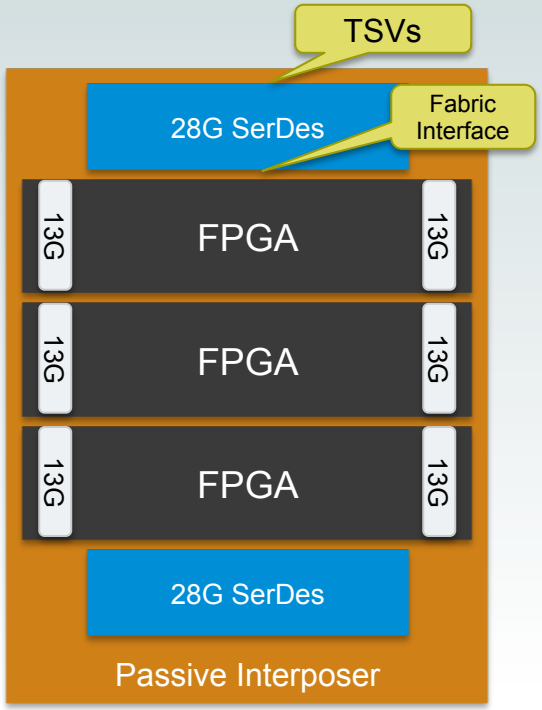
Chip Input/Output Bottleneck



15x Drop in I/O-to-logic Ratio by 2020

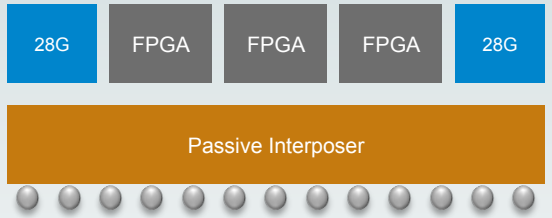
Virtex-7 HT: Heterogeneous SerDes Today

Top View



- 2.8Tb/s ~3X Monolithic
- 16 x 28G Transceivers
- 72 x 13G Transceivers
- 650 GPIO

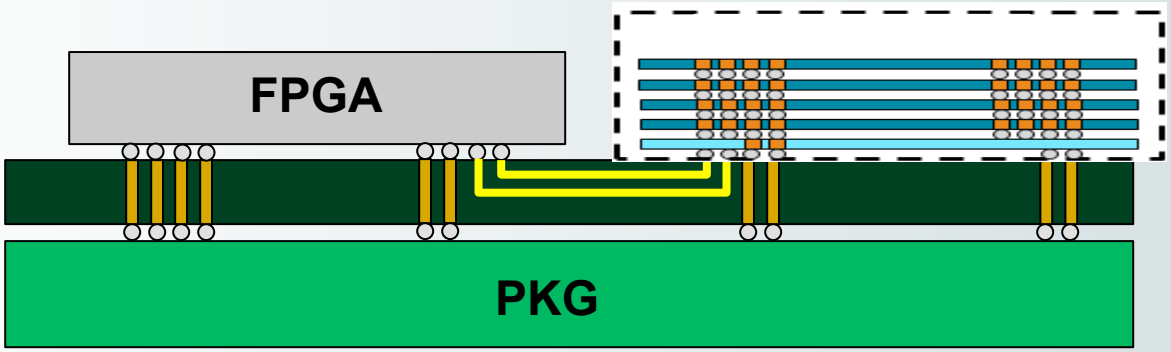
Cross Section



- Yield optimized
- Noise isolation
- 28G process optimized for performance
- FPGA process optimized for power

In the Future: Integrated Memory

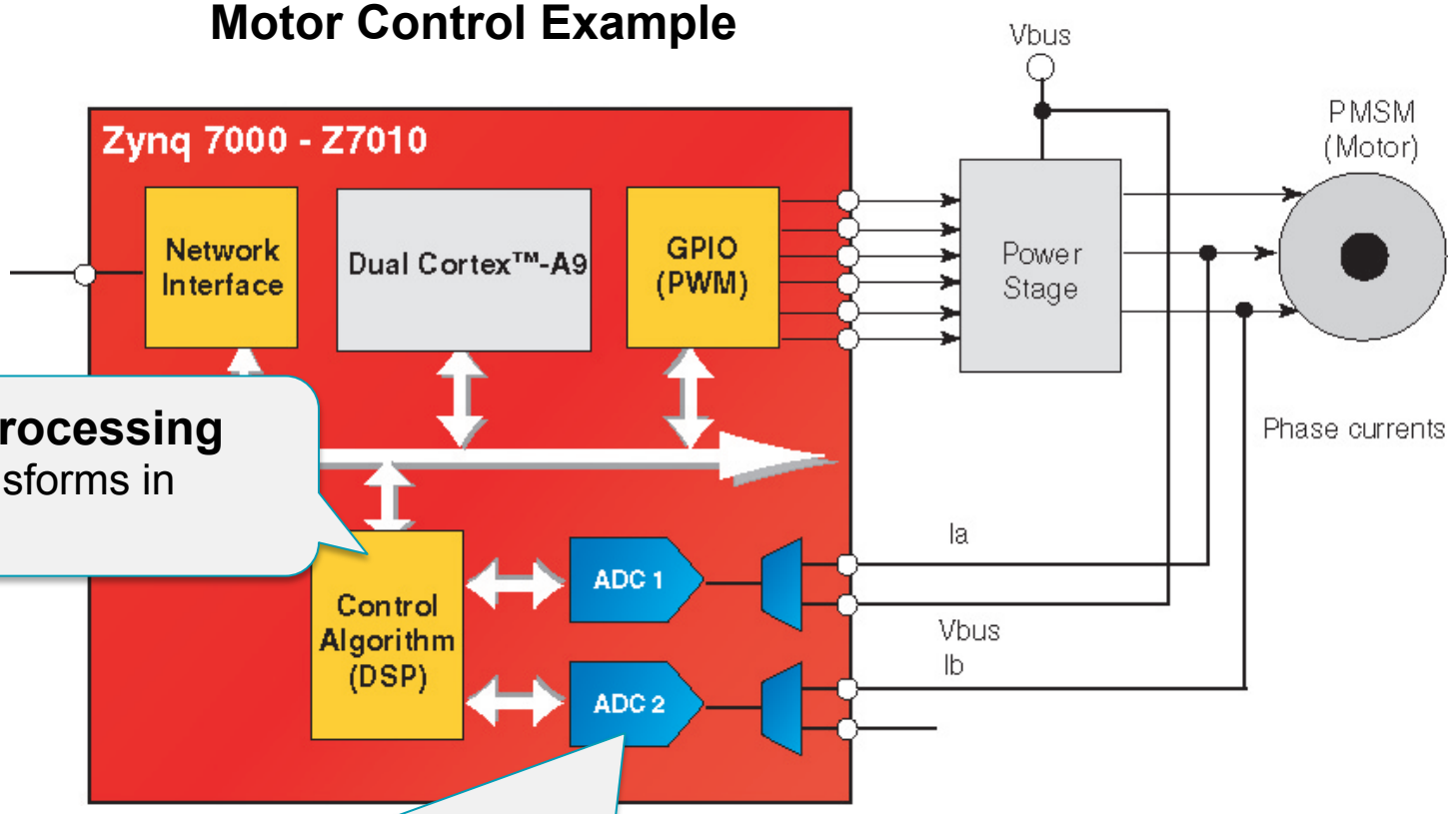
- Higher Memory Bandwidth at Lower Power
  - 1Tbps – 2Tbps
  - ~1Gb/s per interposer wire
  - Simple extension of existing work



Note: This does not reflect any current or proposed product offering from Xilinx

Analog Integration Further Enables Applications

Motor Control Example

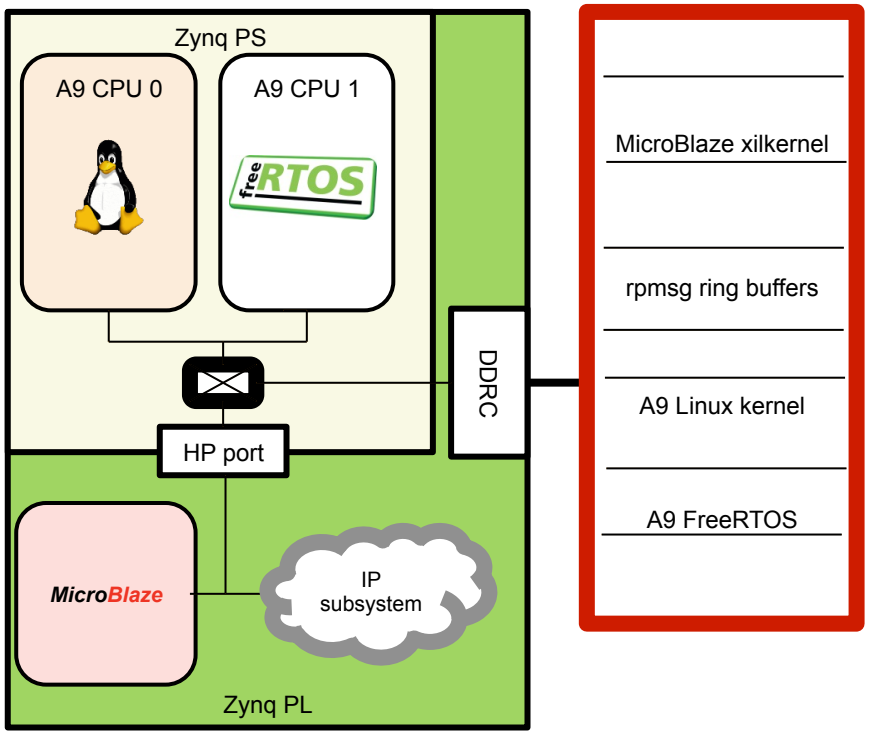


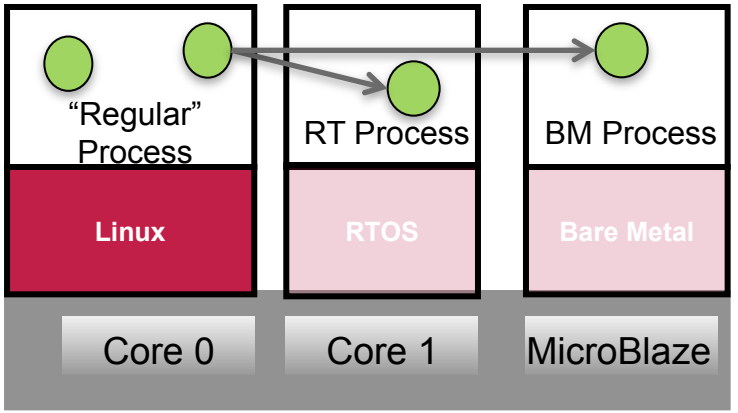
Custom Signal Processing
- Clarke & Park transforms in FPGA fabric

Simultaneous Sampling of Ia & Ib
- Accommodate current senor output unipolar / differential
- Synchronize ADC sampling to PMW

Programming Heterogeneous Multi-Cores: An AMP Framework

Uses rproc/rpmsg





- **Use Linux for Start/Stop**
  - Invoke app on AMP core like a process
    - Compile/link with different compiler depending on target core
    - Linux figures out where to put it

Many Areas to Explore

- **Fabric as accelerator paradigms**
  - Data movement
  - Fast design space exploration
  - FPGA enables easy prototyping
- **Programming models**
  - Need simple models
  - Enabling efficient real time
  - Low latency
- **Fast place and route**
  - So FPGA looks more “software-programmable”
- **Partial Reconfiguration**
  - So it looks like more like software
- **System monitoring capability opens up**
  - Avenues to explore energy efficiency

Summary

- Relentless trend towards integration
- Fabless model unattractive for all except the deepest pockets

- Many of today's applications are best addressed by a combination of Harvard architectures and massively parallel engines
- FPGAs have the key system building blocks plus the interconnect to put them together in a flexible way

Backup