Software-Defined Radio:
A Good Start, but is it Enough?

Manuel Uhm

Vice President, Marketing

Chief Marketing Officer, Treasurer, BoD - Wireless Innovation Forum

2013-06-11

Coherent Logix™

Enabling Low Power Software Defined Systems

Photo Courtesy NASA

Agenda

- Coherent Logix Profile
- The Success of SDR
- Why SDR isn't Enough
- What is a Software Defined System (SDS)
- Competing Processor Options for SDS
- The HyperX SDS Processor
- Conclusions

Coherent Logix Profile

Maker of low-power, high performance, C-programmable processors (HyperX™) and RF chipsets (*rf*X™) for the embedded systems market – enabling low-power, real-time software defined systems.

Wireless

Image / Video

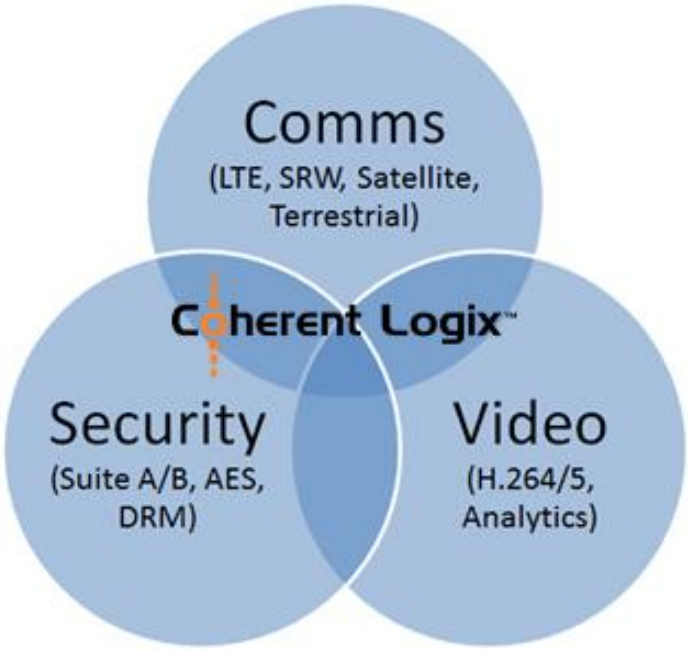
Mil / Aero

High-Rel / Rad-Tol

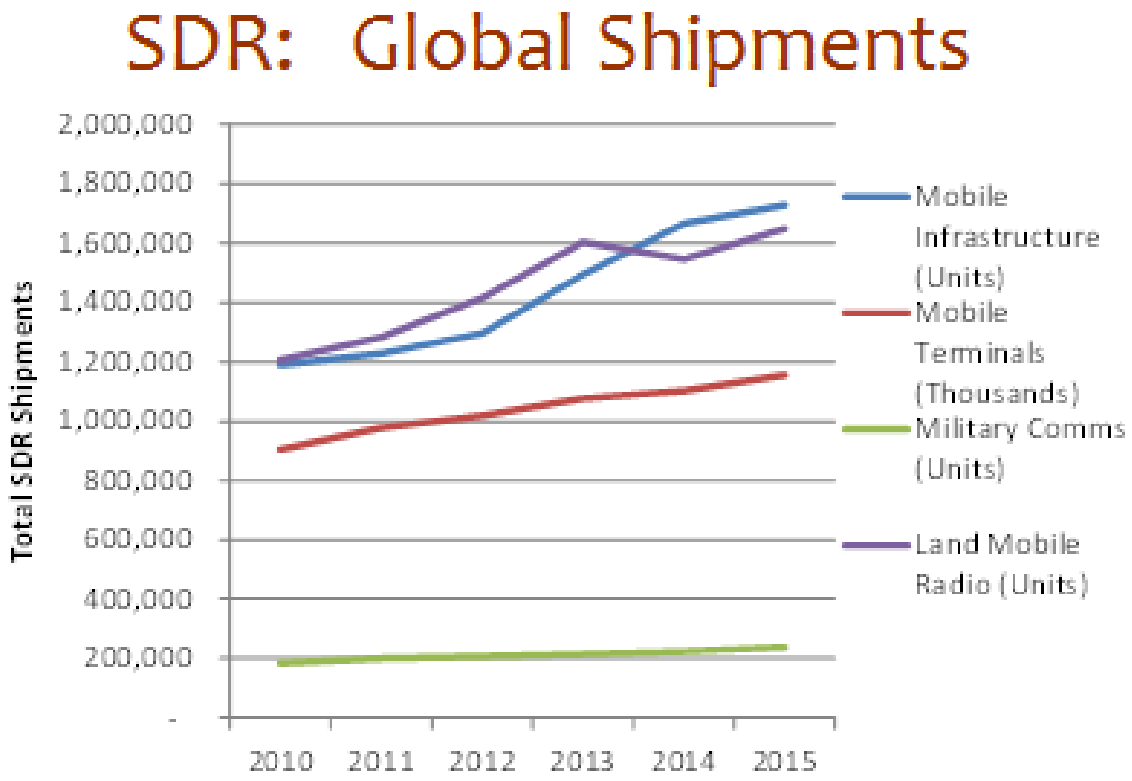
Coherent Logix™
Enabling Low Power Software Defined Systems™

Current Application Focus

- Dual-use technology to handle wireless data explosion
  - Convergence of wireless and video
  - Convergence of communication and computation
  - Secured delivery of wireless data
  - Low power requirement is everywhere
- Enables a disruptive change in mobile infrastructure, mobile devices, intelligent cameras, and networked sensors
- Scalable systems
  - From racks to watches
  - Emphasis on small form factor low power heterogeneous computing platform



The Success of SDR



- Mobile Terminals represents the biggest segment by far
- SDR is taking analog market share in Public Safety and Private Mobile Radio (together known as Land Mobile Radio)
- Military Tactical SDR is growing more incrementally

Source: Wireless Innovation Forum SDR Market Size Study, 2011

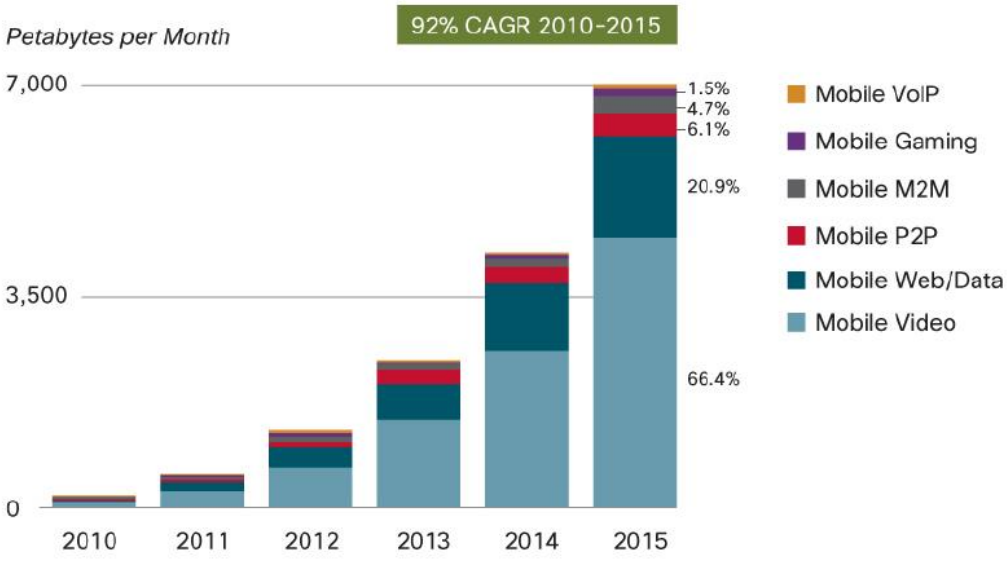
Why Has SDR Been So Successful?

- **Requirement for multi-mode support**
- **Flexibility to add features or adapt to ever evolving air interface protocols even after deployment, and do so in a manner that is CAPEX and OPEX friendly**
- **Development cost savings of code preservation**
- **Time-to-market benefit from software code reuse**

But it's not enough…

Why SDR Isn't Enough…

Figure 5. Mobile Video Will Generate 66 Percent of Mobile Data Traffic by 2015



VoIP traffic forecasted to be 0.4% of all mobile data traffic in 2015.
Source: Cisco VNI Mobile, 2011

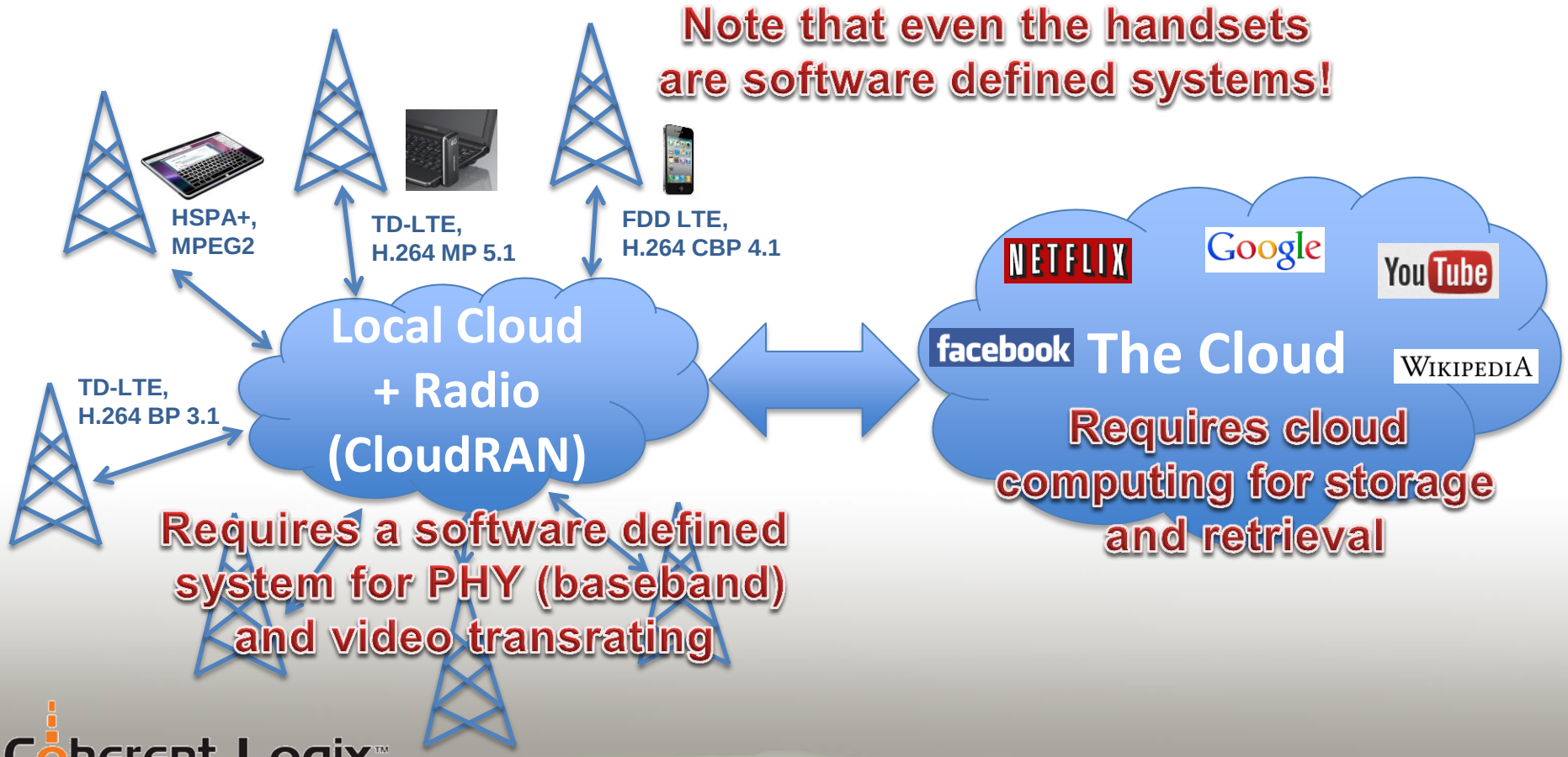
- **Video is driving wireless data rates**
- **Future wireless systems will:**
  - Couple video tightly to reduce latency, maximize bandwidth usage, and improve the user experience,
  - Integrate computing, location awareness, and sensor fusion with communications to provide intelligent communications and augmented reality
- **This requires a Software Defined System approach to do wireless, image, video and sensor processing for intelligent communications**

What is a “Software Defined System”?

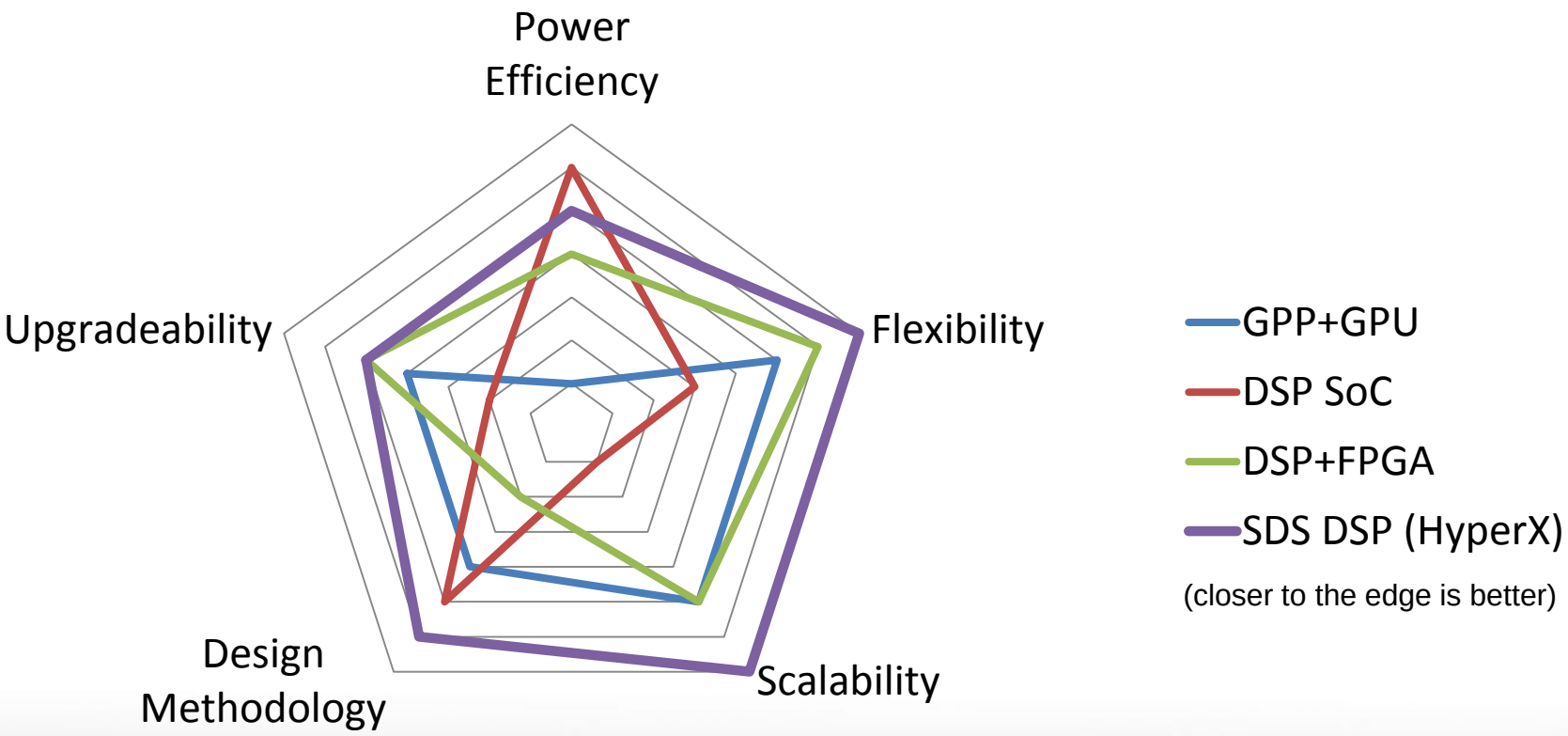
- **Software Defined Systems (SDS) can support both wireless (radio) and image/video functionality entirely in software, thereby virtualizing the signal processing.**
- **The ideal SDS is completely virtualized and ultimately flexible, meaning that all processing resources are available for wireless and/or video processing - it’s just a different software load (no hardware accelerators are needed).**

An Example of the Need for a Software Defined System

- **Bring the cloud to the edge by integrating it with a CloudRAN (large pool of baseband processing connected to Remote Radio Heads by fiber)**
  - Use video transrating to optimize bandwidth (based on device capability)
- **Results in efficient usage of available spectrum and bandwidth, and lower latency for mobile cloud computing.**



Processor Options for Software Defined Systems



Processor Competitive Comparisons by Requirements

	GPP + GPU	DSP w H/W Accelerators (SoC)	Hybrid DSP + FPGA	SDS HyperX DSP (Software Defined System)
Power Efficiency (Performance/W)	Very poor. GPUs not optimized for wireless or video processing and are very power hungry.	Very good, but only for a small, self-contained system. Requires different SoCs for wireless vs video processing.	Good (between options 1 and 2) since FPGAs consume more power than a hardware accelerator.	Good (between options 1 and 2) since HyperX consumes more power than a hardware accelerator.
Flexibility (multi-mode, profiles, levels, etc.)	Limited due to lack of optimization. Unable to perform compute intensive tasks like turbo decoding.	Severely limited due to H/W accelerators. No SoC available with both wireless and video accelerators.	Good flexibility but hardware/software co-design requires careful partitioning a priori.	Very flexible due to ability to use software acceleration for both wireless and video processing.
Upgradeability (future proof, able to support new algorithms or standards)	Limited due to lack of optimization. Unable to perform compute intensive tasks like turbo decoding.	Severely limited due to H/W accelerators. Obsolete with new air interface or video codec.	Possible, but difficult as it may require a total system repartitioning and rewrite.	Very upgradeable due to ability to use software acceleration for both wireless and video processing.
Scalability (adding more capability as required)	Scales but the right mix of devices difficult to determine a priori.	Severely limited due to I/O constraints designed to operate as a single chip.	Scales but the right mix of devices is difficult to determine a priori.	Very scalable. Multi-processor implementations scale with no glue logic.
Design Methodology (time-to-market, development time)	Very difficult due to lack of heterogeneous design/debug environment and lack of support for wireless/video processing.	C reprogrammability enables fast simulation and iterations, but little to no tools support for multi-chip designs (design or debug).	Very difficult due to lack of heterogeneous design/debug environment and need to use VHDL/RTL for FPGAs. Very slow iterations due to lengthy place and route	Very good due to homogeneous system-level design/ debug environment. C reprogrammability enables fast simulation and iterations.

Too much power

Lacks flexibility

Hard to program

An SDS DSP is the best choice for a fully software defined system.

Introducing…the HYPERX™ processor

A very high performance, ultra-low power multicore (100) processor that:

- **has comparable power efficiency to an ASIC - better than DSPs and FPGAs, and much better than GPPs and GPUs,**
- **has the processing performance of an FPGA to do tasks that normally require hardware accelerators completely in software, such as LTE turbo decoding and H.264 CABAC,**
- **has the ease-of-use and C programmability of a GPP, resulting in faster time-to-market**
- **may be software upgraded after deployment to support new air interfaces, codecs, advanced algorithms, or niche variants (i.e., LTE MBMS for broadcast or 4:2:2 chroma format) which other processors are not capable of supporting today,**
- **can scale from both a hardware (i.e., I/O) and software perspective (i.e., code reuse),**
- **is low latency and 100% deterministic,**
- **is highly secure with advanced digital rights management and security features.**

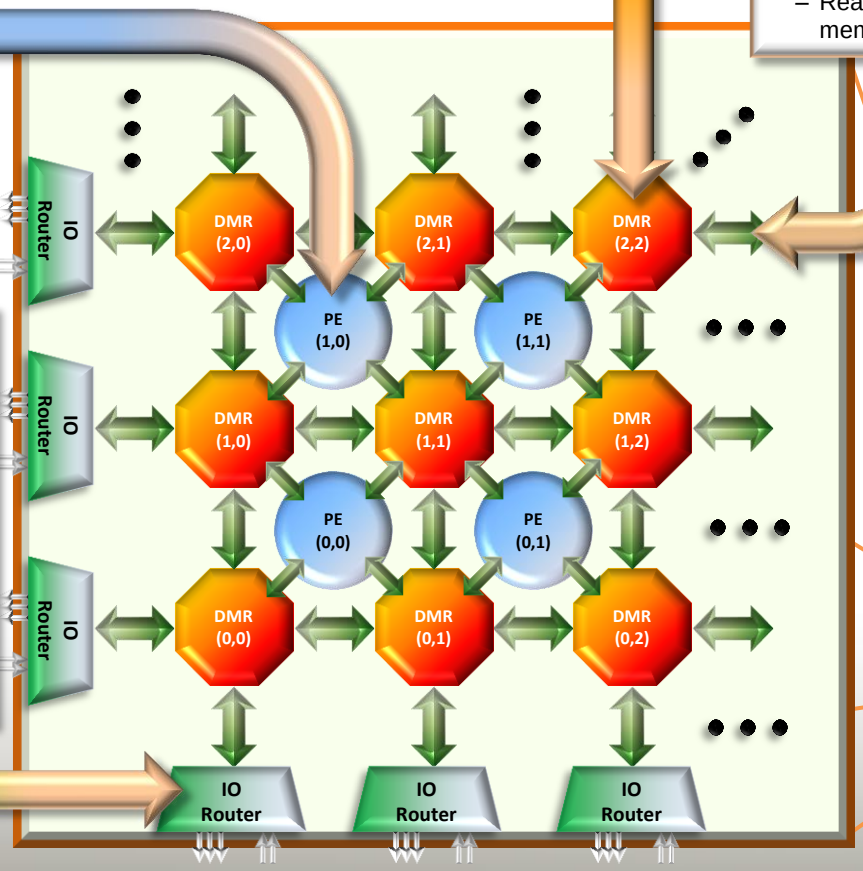
The Engine – What is the HyperX Processor: hx3100

100 Processing Resources (PEs)
- GPP/DSP w/ Variable clock to 600MHz +
- Supports data types: 8, 16, Nx16-bit integer, & 32-bit floating point
- 400KB of total on-chip program memory
  - Each PE supported directly by 4KB
- @ 500MHz
  - 50,000 MIPS
  - 50 16-bit GMACS
  - 100 8-bit GMACS
  - 25 GFLOPS

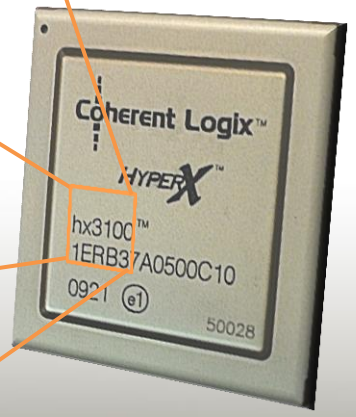
121 Data-Memory-Routers (DMRs)
- Memory Embedded in Network or Network Embedded in Memory Architecture
  - Hierarchical, Multi-Dimensional Communications
  - Physically Flat Memory
- 968KB of total on-chip data memory
  - 8KB data memory per DMR

Dynamic On-chip Memory-Network
- Autonomous data movement
- Instantaneous bandwidth on demand
- Real-time adaptable to support multiple memory and communication topologies

IO Routers
- 16 Multi-function General Purpose IO Channels
  - Physically Programmable
    - LVDS (EIA-644) and CMOS
  - Logically Programmable
    - GPIO, SYNC, ASYNC, & Multi-chip provides seamless chip-to-chip support *without* glue logic that would compromise performance or break the programming model
- 8 High-Speed External Memory IO Channels
  - 8 Programmable Controllers
  - Supports DDR2
  - Access up to 64 GB of total off-chip memory
- 24 programmable timers



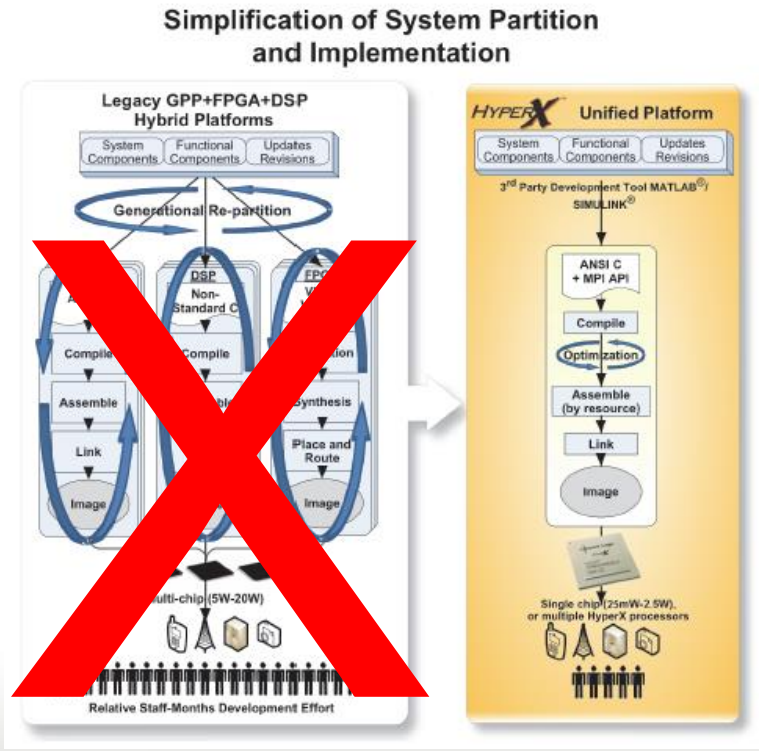
Performance
- 32-64 16-bit GMAC/s/W
- 64-128 8-bit GMAC/s/W
- 16-32 GFLOP/s/W



Unified Design Methodology Advantages

The design methodology is more important than the processor!

- **Program in ANSI C or Simulink**
- **Your C-based golden reference model is the basis for your design**
  - No “throw-away” code or need to redesign in VHDL
- **Very fast simulation in software or hardware enables rapid design iterations**
  - No need for behavioral synthesis or timing closure
- **Supports multi-chip designs**
  - Program and debug at the system level
  - No need to integrate and debug disparate designs on heterogeneous devices



Conclusions

- **Software Defined Systems (SDS) can support both wireless (radio) and image/video functionality entirely in software, thereby virtualizing the signal processing, and allowing video to be tightly coupled to a wireless system**
- **But an SDS requires a new model of processing**
- **Flexibility is paramount due to the numerous options/variants in target markets, features, etc.**
- **Upgradeability is a key differentiator that can have a huge impact on product longevity and lowering CAPEX**
- **Power efficiency is critical to reducing OPEX**
- **Scalability enables future growth**
- **Design methodology is significant in faster time-to-market and reducing total cost of ownership**
- **HyperX is an enabling technology for truly Software Defined Systems**

Thank you!

Manuel Uhm
Vice President, Marketing
manuel@coherentlogix.com
+1 (408) 600-1456

Coherent Logix™

Enabling Low Power Software Defined Systems

Photo Courtesy NASA