Software-Defined Systems: Using SDR Technology to Solve a New Problem Beyond Radio

Research.
Design.
SDR - WInnComm
Wireless Innovation Forum Conference on Wireless Communications Technologies and Software Defined Radio
Manufacture.
Deploy.

Connecting technical, business and regulatory leaders ~ Defining the future of radio communications

11-13 March 2014 ~ Schaumburg, Illinois

Manuel Uhm

Vice President, Marketing – Coherent Logix

Chief Marketing Officer, Board of Directors - Wireless Innovation Forum

2014-03-11

Coherent Logix™

Enabling Low Power Software Defined Systems

Confidential Business Information and Proprietary Data

Photo Courtesy NASA

Coherent Logix Profile

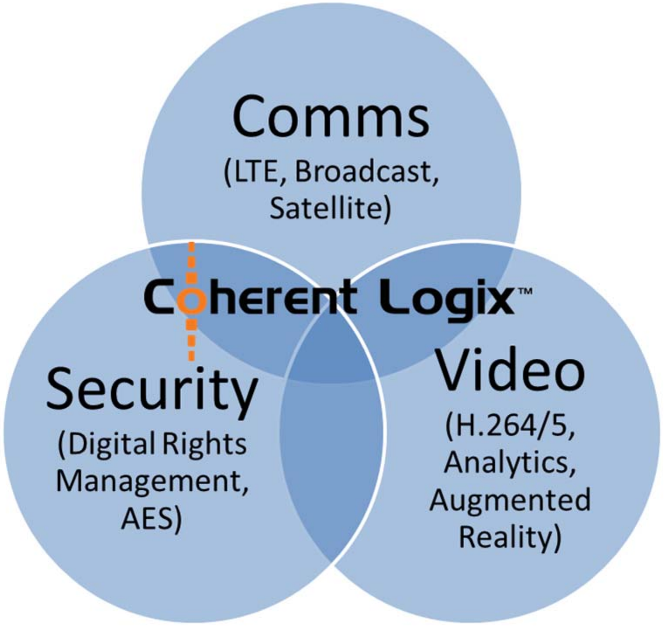
Maker of low-power, high performance, C-programmable processors (HyperX™) and RF chipsets (*rf*X™) for the embedded systems market
– enabling low-power, real-time software defined systems.

Portland, OR
San Jose, CA
Austin, TX
Boston, MA
Shanghai, China
Tokyo, Japan
Seoul, Korea
Atlanta, GA

Wireless
Image / Video
Mil / Aero
High-Rel

Coherent Logix™
Enabling Low Power Software Defined Systems™

The Coherent Logix Purpose



Comms
(LTE, Broadcast, Satellite)

Coherent Logix™

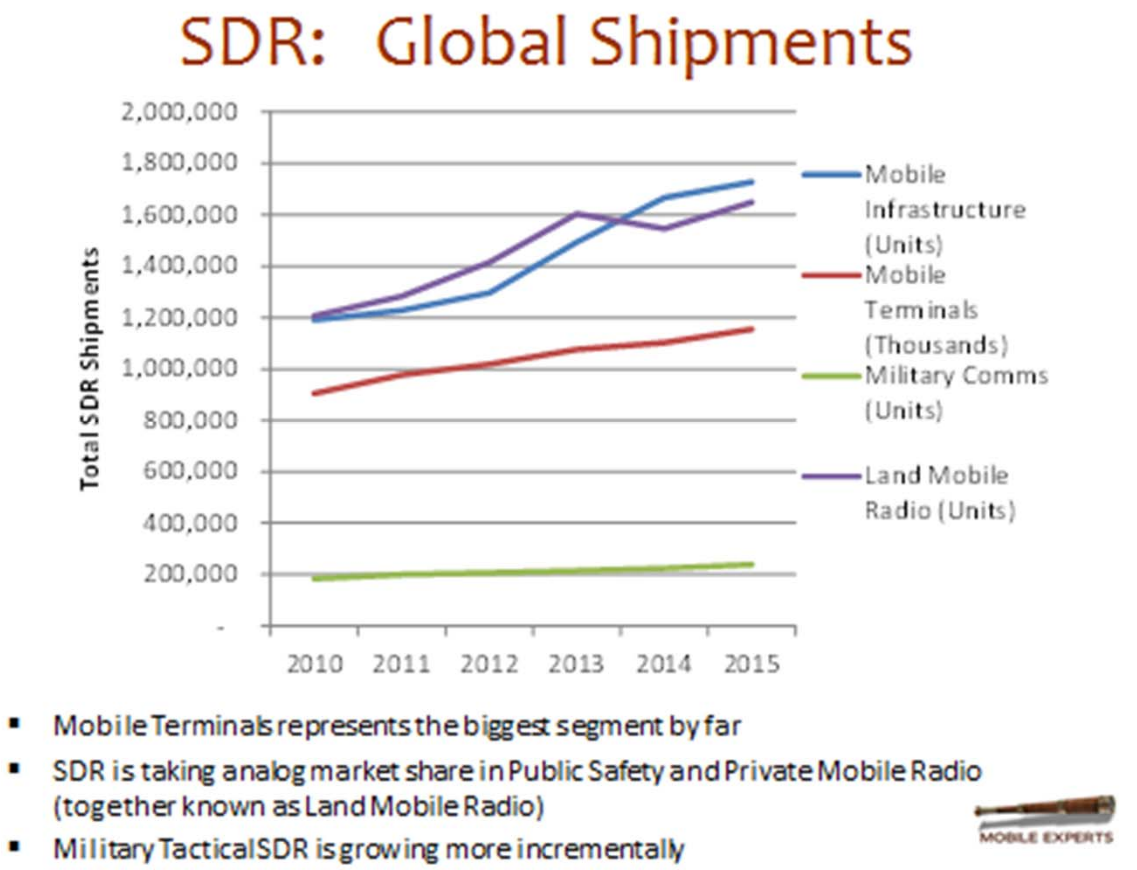
Security
(Digital Rights Management, AES)

Video
(H.264/5, Analytics, Augmented Reality)

Coherent Logix is enabling the convergence of voice, video and image processing being securely transmitted wirelessly at very low power.

The world is changing too fast for traditional ASIC or even FPGA development cycles. Software Defined Systems are required since one can no longer anticipate every possible feature or requirement in advance, necessitating maximum flexibility at the lowest possible power consumption.

4G: The Success of SDR



SDR: Global Shipments

- 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

Most 4G eNodeBs and UEs are based on SDR technology!

Coherent Logix™
Enabling Low Power Software Defined Systems™

Confidential Business Information and Proprietary Data

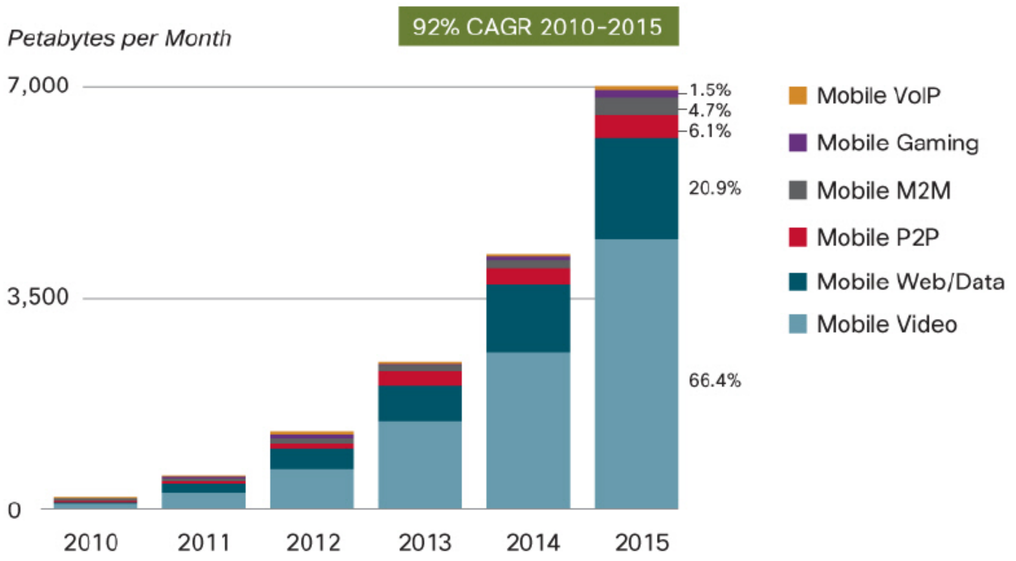
Why Has SDR Been So Successful for 4G?

- **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 Beyond 4G…

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

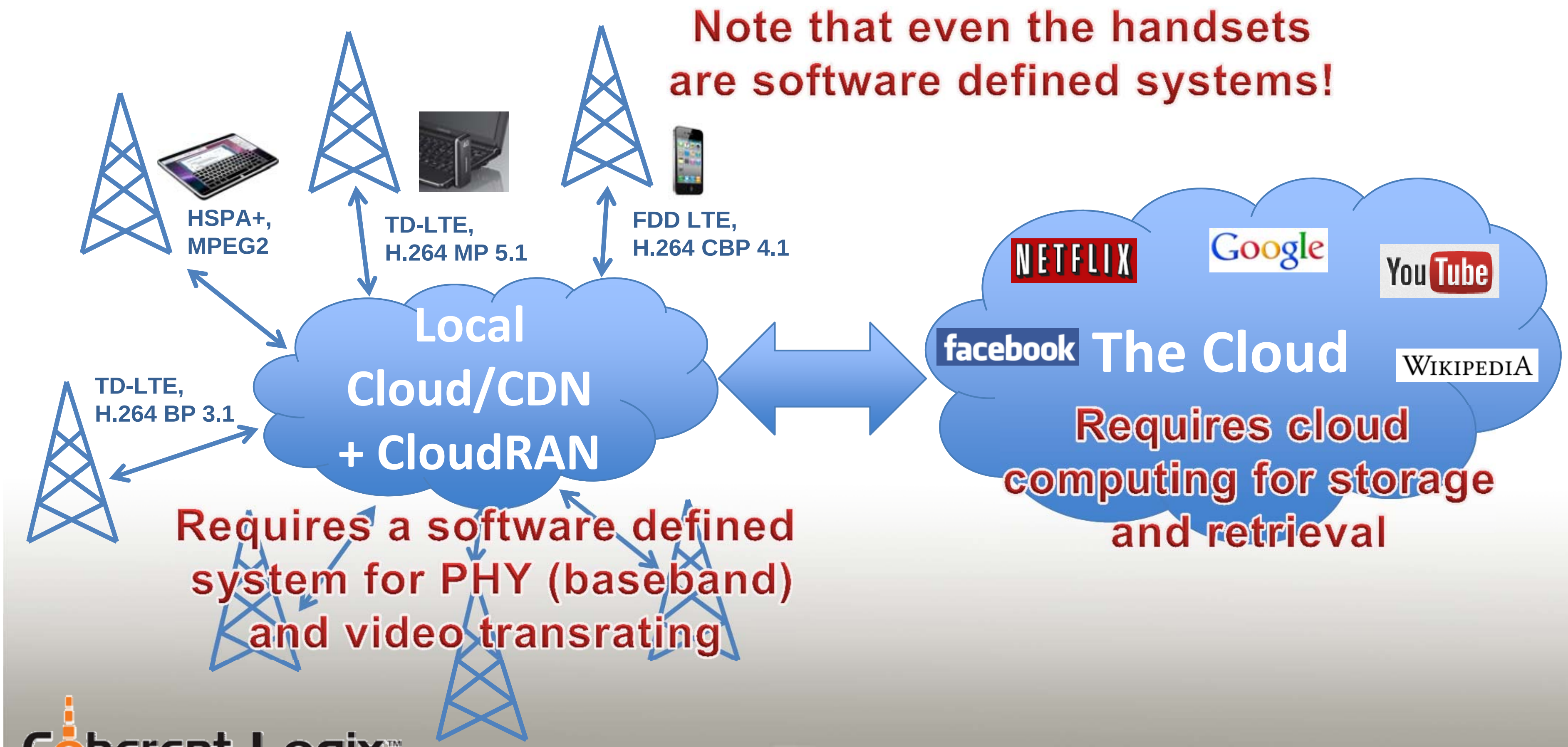
- **Video is driving 4G wireless data rates**
- **Future wireless systems will:**
  - Couple video tightly to reduce latency, maximize bandwidth usage, and improve the user experience,
  - Integrate computing, location awareness, and sensor fusion with communications to provide intelligent communications 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.**
- **By co-locating real-time video and PHY processing, latency can be significantly reduced.**
- **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).**

SDS: Enabling Real-Time Video Processing at the Edge

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



Note that even the handsets are software defined systems!

Requires a software defined system for PHY (baseband) and video transrating

Requires cloud computing for storage and retrieval

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.**

NAB’13 (National Association of Broadcasters) Demo: Network Edge Transcoder with LTE Base Station

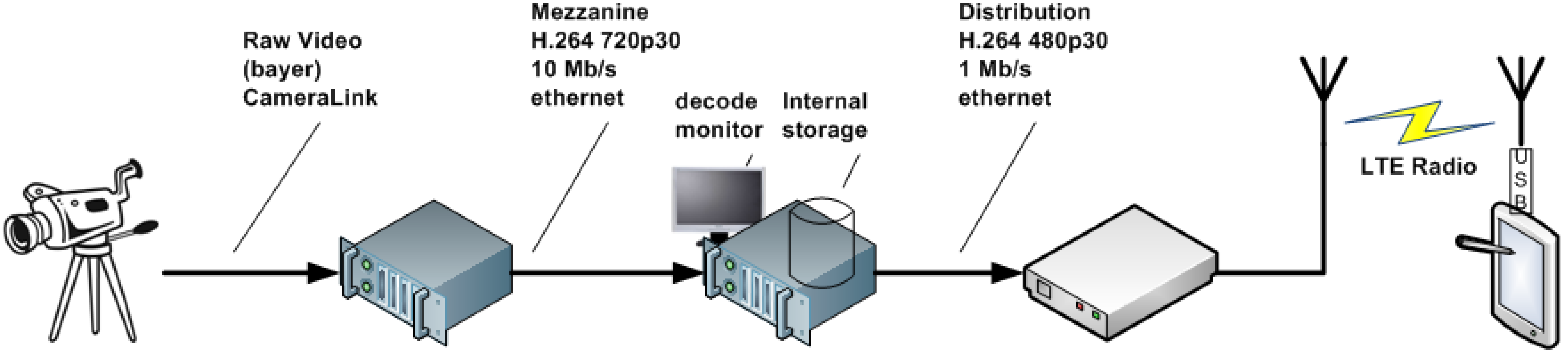
Video Capture

Mezzanine Encoder

Network Edge Transcoder

Public Wireless LTE Base Station

Handheld Device



All software defined
- Demosaic (bayer)
- Color correction
- Colorspace conversion
- H.264 HP encoder

Platform
- TCP/IP stream out

All software defined
- HQ downscale
- H.264 HP transcoder
- Integrated software configurable storage

Platform
- TCP/IP stream in
- H.264 decode + monitor
- TCP/IP stream out

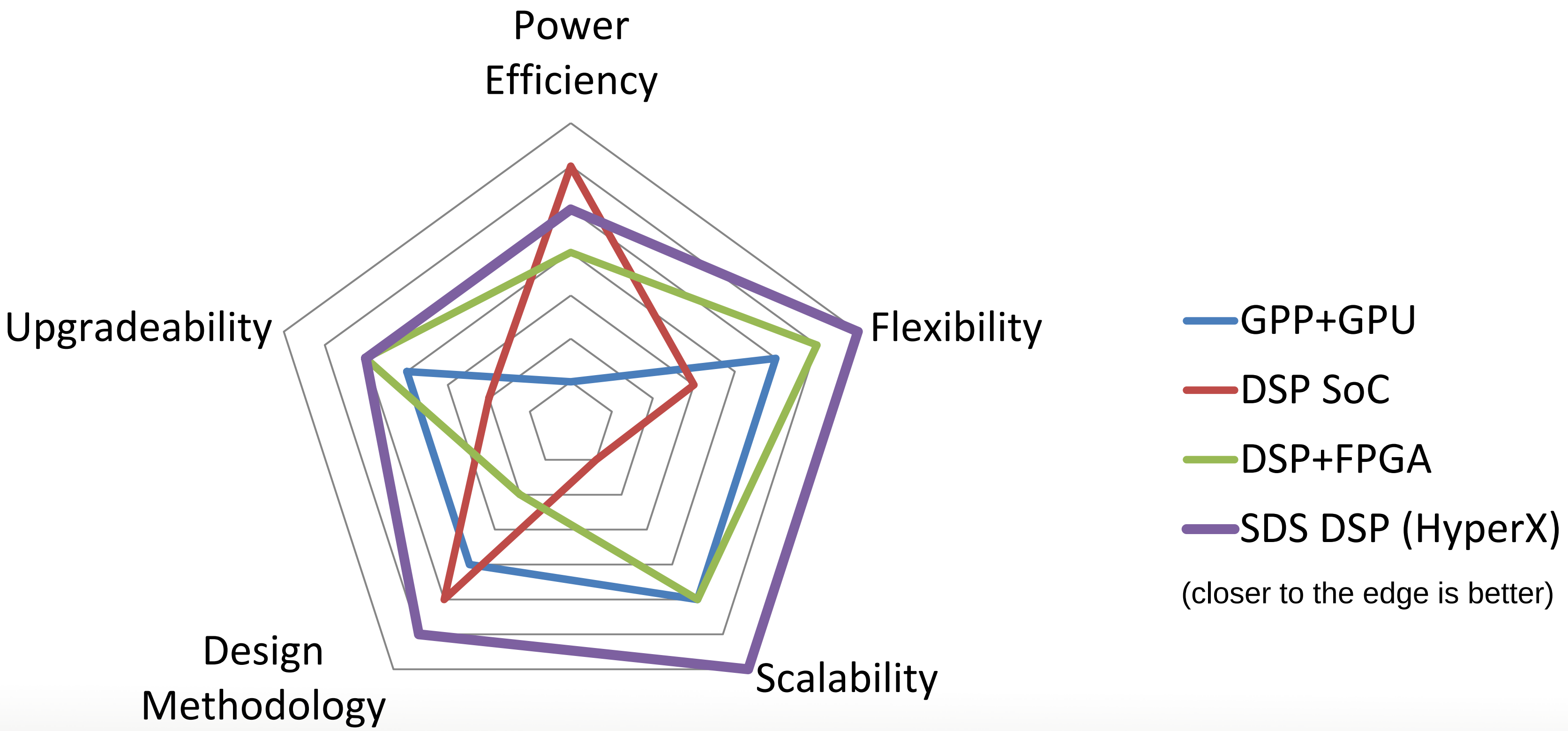
All software defined
- LTE PHY

Future 28nm HyperX Processor

Enabling Low Power Software Defined Systems™

Confidential Business Information and Proprietary Data

Processor Options for Software Defined Systems



(closer to the edge is better)

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 is 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.

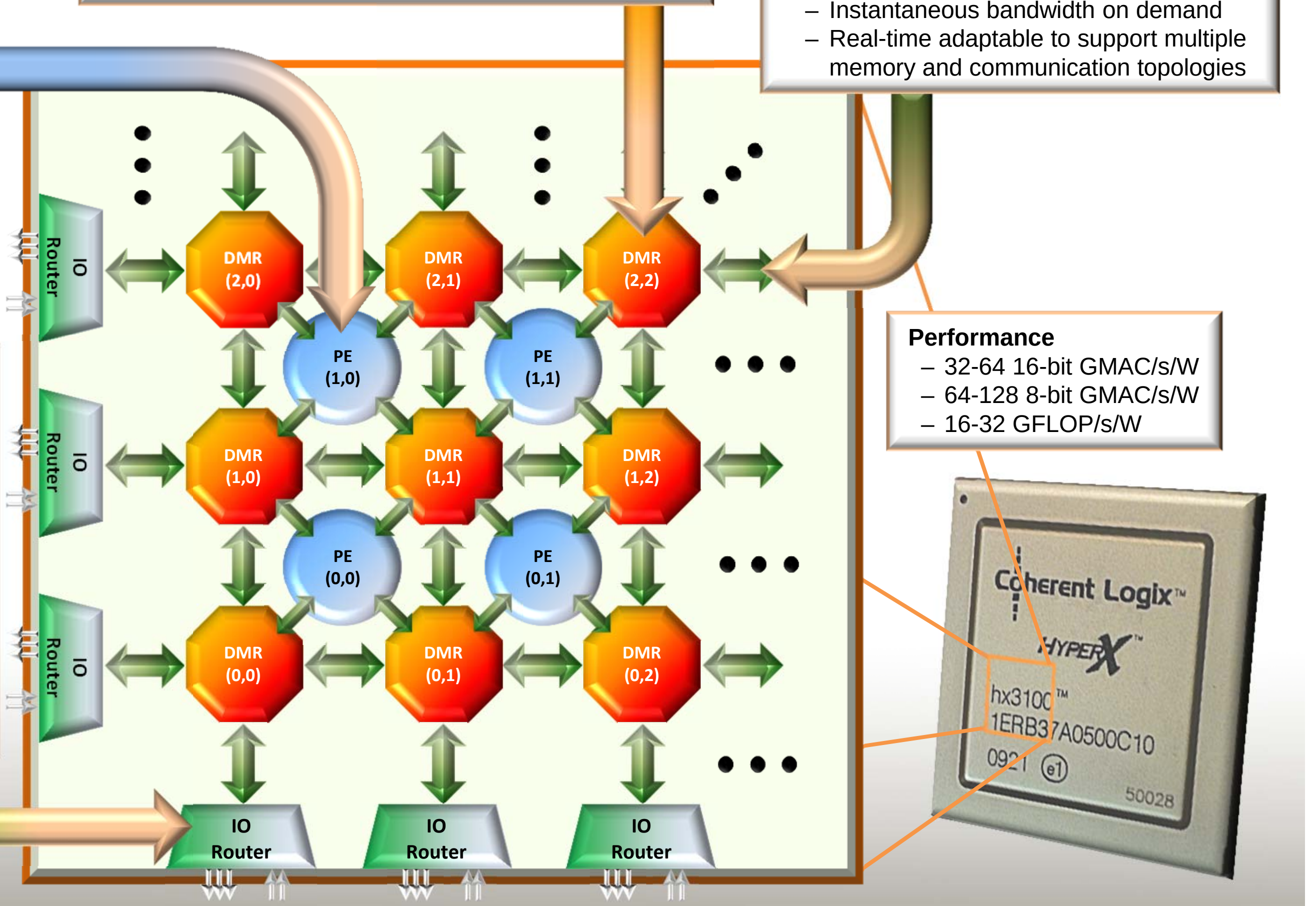
Too much power

Lacks flexibility

Hard to program

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

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

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

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

The hx3100 Processor Performance

Performance Throughput

- @ 500 MHz
- 50,000 MIPS
- 50 16-bit GMACS
- 100 8-bit GMACS
- 25 GFLOPS (32-bit)

DIE IO Bandwidth:

- 96 Gbps of LVDS IO
- 24 Gbps of CMOS IO
- 64 Gbps of DDR2 IO

General Purpose Package IO Bandwidth:

- 64 Gbps of LVDS IO
- 12 Gbps of CMOS IO
- 32 Gbps of DDR2 IO

Scalability

Performance

Power

Programmability™

25 mW to 2.5 W

(algorithm dependent)

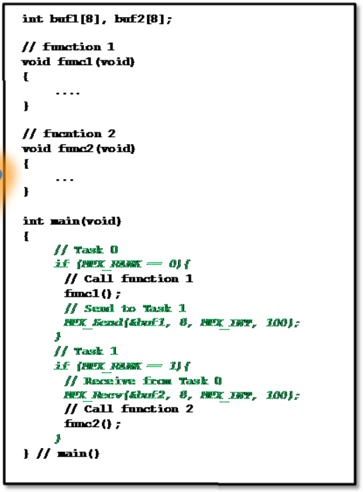
Performance Efficiency

- @ 500 MHz
- 64 16-bit GMAC/s/W
- 128 8-bit GMAC/s/W
- 32 GFLOP/s/W (32-bit)
- > 2.4 TOP/s/W (16-bit RISC equivalent)

(*) General Purpose Use Package Shown.
Die Intended To Be Packaged To Application, e.g. below.

Software Development to Hardware

Language – ANSI C

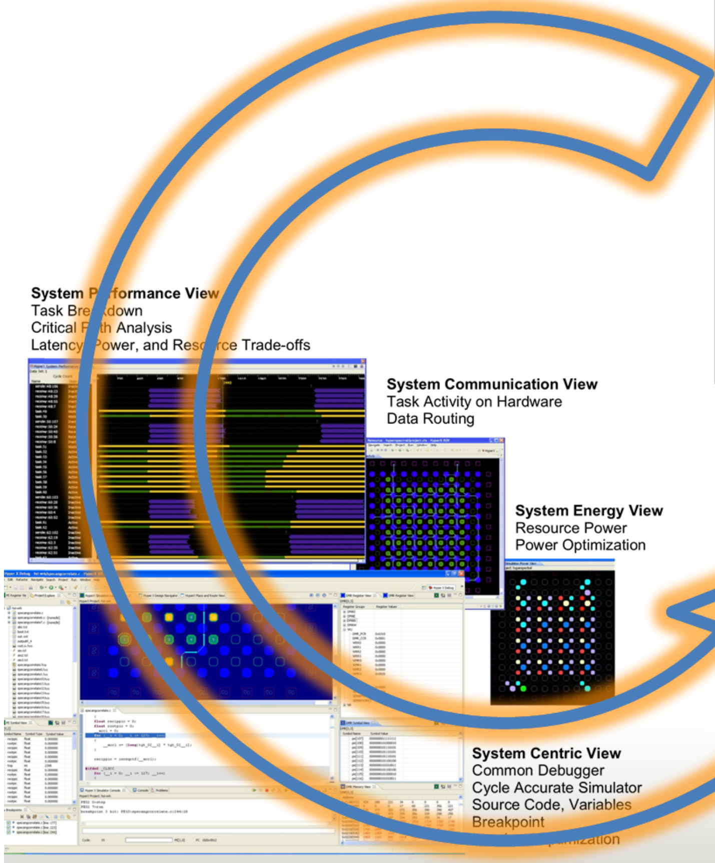


```
int buf1[8], buf2[8];

// function 1
void func1(void)
{
     ....
}

// fucntion 2
void func2(void)
{
     ...
}

int main(void)
{
     // Task 0
     if (MPX_RANK == 0){
       // Call function 1
       func1();
       // Send to Task 1
       MPX_Send(&buf1, 8, MPX_INT, 100);
     }
     // Task 1
     if (MPX_RANK == 1){
       // Receive from Task 0
       MPX_Recv(&buf2, 8, MPX_INT, 100);
       // Call function 2
       func2();
     }
} // main()
```

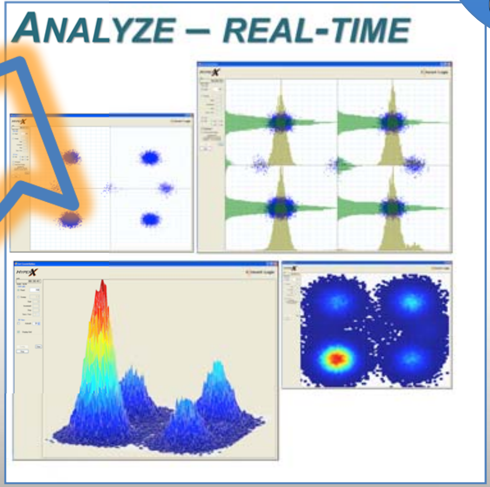


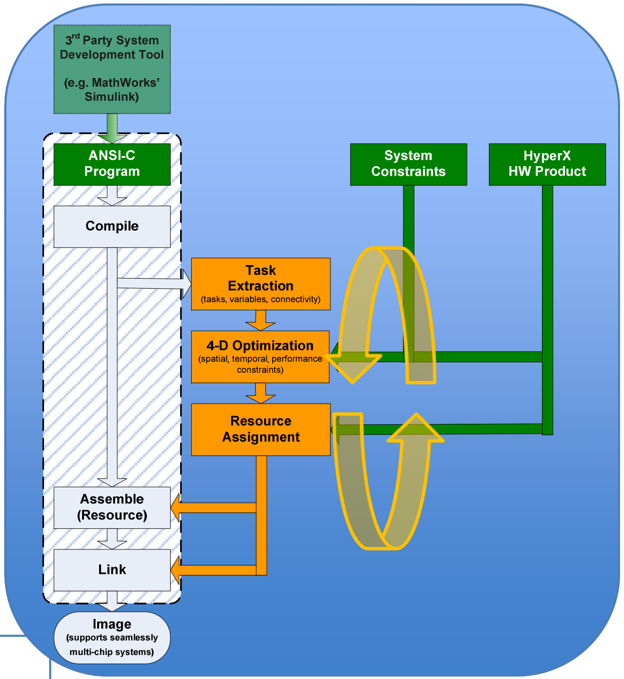
System Performance View
Task Breakdown
Critical Path Analysis
Latency, Power, and Resource Trade-offs

System Communication View
Task Activity on Hardware
Data Routing

System Energy View
Resource Power
Power Optimization

System Centric View
Common Debugger
Cycle Accurate Simulator
Source Code, Variables
Breakpoint





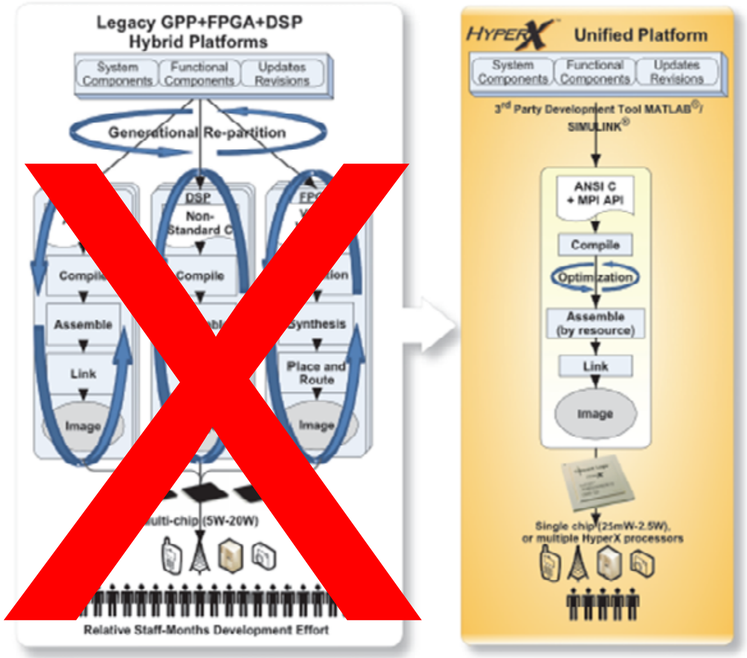
3rd Party System Development Tool
(e.g. MathWorks' Simulink)
ANSI-C Program
Compile
Task Extraction
(tasks, variables, connectivity)
4-D Optimization
(spatial, temporal, performance constraints)
Resource Assignment
System Constraints
HyperX HW Product
Assemble (Resource)
Link
Image
(supports seamlessly multi-chip systems)

- Source [code] platform independent until compile time
- > 10x reduction in development time over current (DSP/FPGA) architectures
- Unified development and verification flow
- Design optimizations (e.g. performance, power, and latency) performed without changing C code
- Leverages model-based design concepts

Unified Design Methodology Advantages

The design methodology is more important than the processor!

- **Program in ANSI C**
- **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



Simplification of System Partition and Implementation

Coherent Logix™

Enabling Low Power Software Defined Systems™

Conclusions

- **SDR has been a key enabling technology for 4G eNBs and UEs**
- **With video driving 4G wireless bandwidth, there is a need to go beyond SDR to SDS in order to reduce system-wide latency**
- **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 to reduce latency**
- **But an SDS requires a new model of processing**
- **The HyperX SDS processor is an enabling technology for true SDS and real-time video processing at the edge**

Thank you!

Manuel Uhm

Vice President, Marketing

manuel@coherentlogix.com

+1 (408) 600-1456

Photo Courtesy NASA