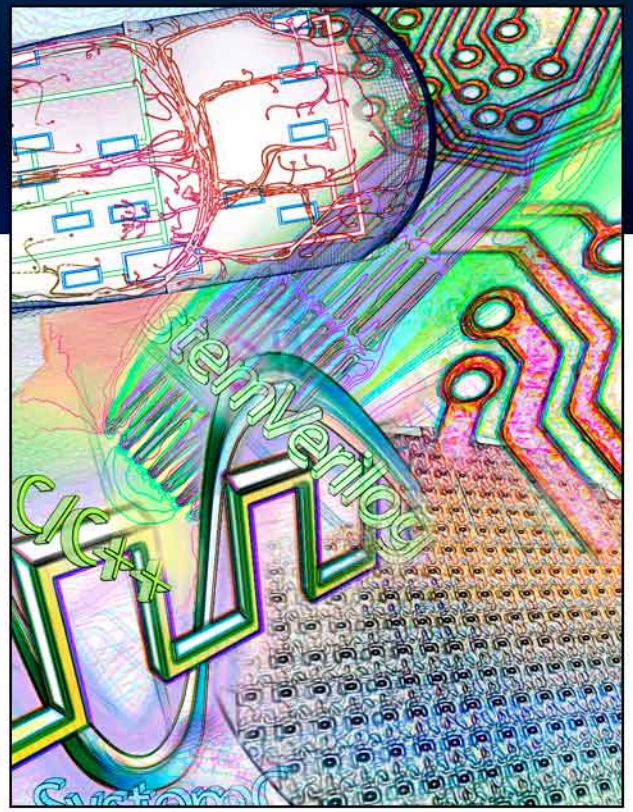


Complete SoC Validation from Device Drivers to Peripherals

Jim Kenney

Director of Product Marketing

Mentor Emulation Division

Mentor Graphics®

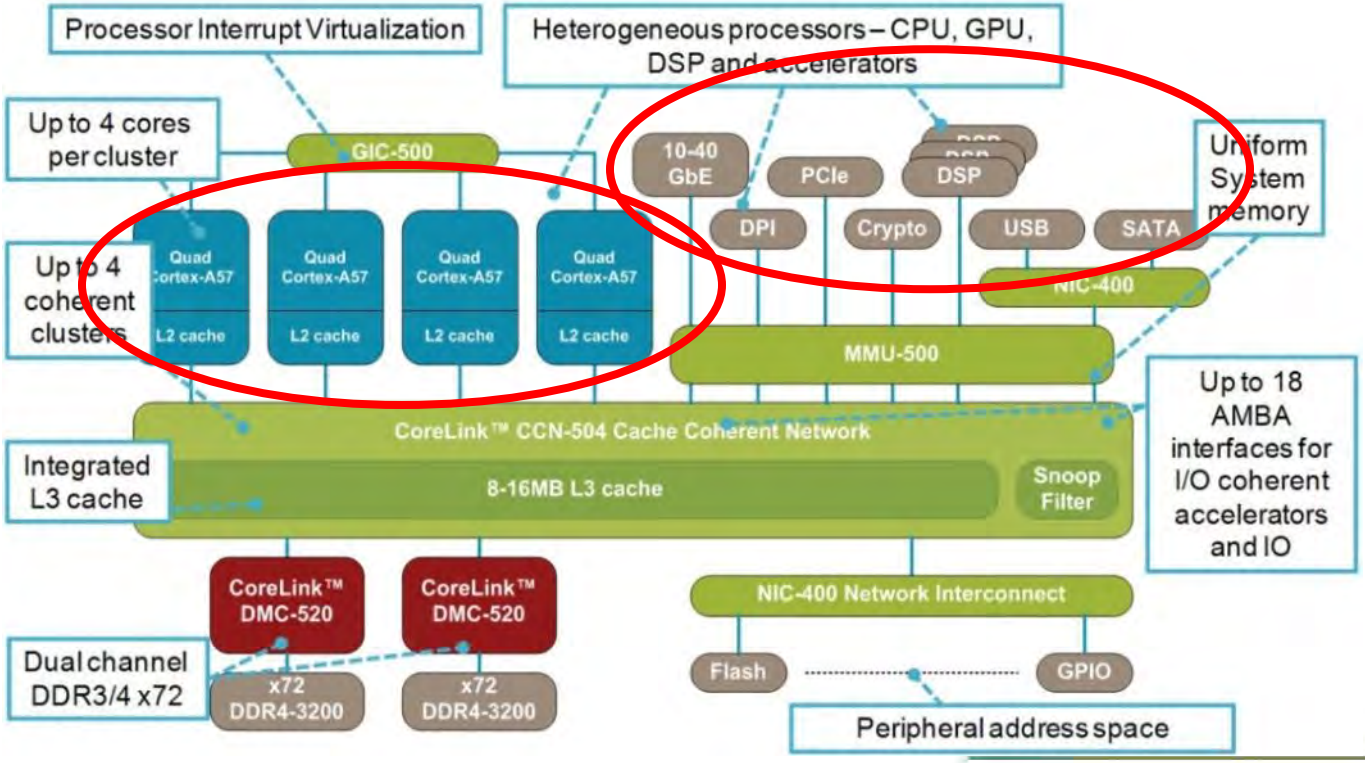
We start with 1 paradigm
That serves us well at the time

We stretch the
Paradigm to fit
Our growing
needs

Until the paradigm is
Stretched to breaking point
Unable to meet needs it was
Never designed to achieve

Advance SoCs Have Changed Everything

- Where to run device drivers and how to debug them?
- How to model peripheral devices?



Where to Execute RTOS and Drivers

- RTL processor models on hardware emulator
  - Clock-cycle hardware accurate
  - Validates path through bus fabric to peripheral devices
  - Executes code at emulation clock speed of a few MHz
- Fast ISS or virtual machine on Linux workstation
  - Instruction accurate but not hardware accurate
  - Executes code at ~ 100 MHz
  - Assumes bus fabric has been verified

SW Developers Demand Fast Debug

- JTAG probe is the traditional method of live-target debug

"If I had asked people what they wanted, they would have said faster horses."
-Henry Ford

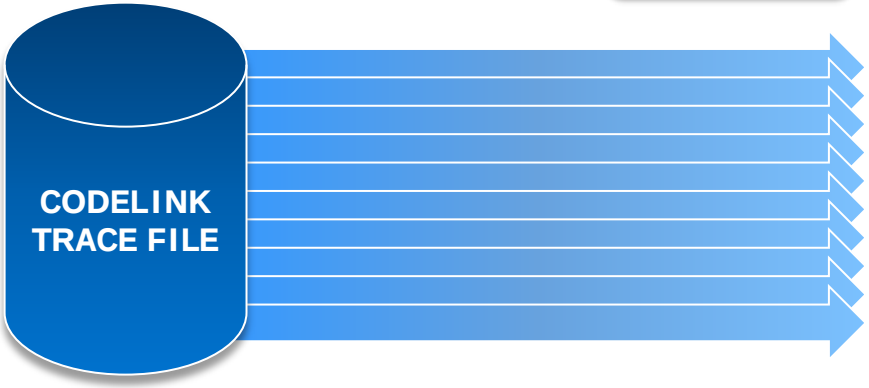
JTAG Emulation is Slow, Intrusive, Expensive

JTAG
1 MHz

One user at a time

Post-Emulation

Codelink
100 MHz



~10 Simultaneous users

JTAG Probes Can Inject Heisenbugs

- Bugs which only manifest themselves when NOT attached to a debugger
- A problem which occurs only when software is run at “full speed”
  - Usually a race condition between hardware and software or two software processes
  - When a JTAG probe debugger is attached to the software it inserts 100,000s of clocks between events – forcing software to “lose” the race
- May be intermittent

Werner Heisenberg

Peripheral Model Targets for Device Drivers

Physical peripherals

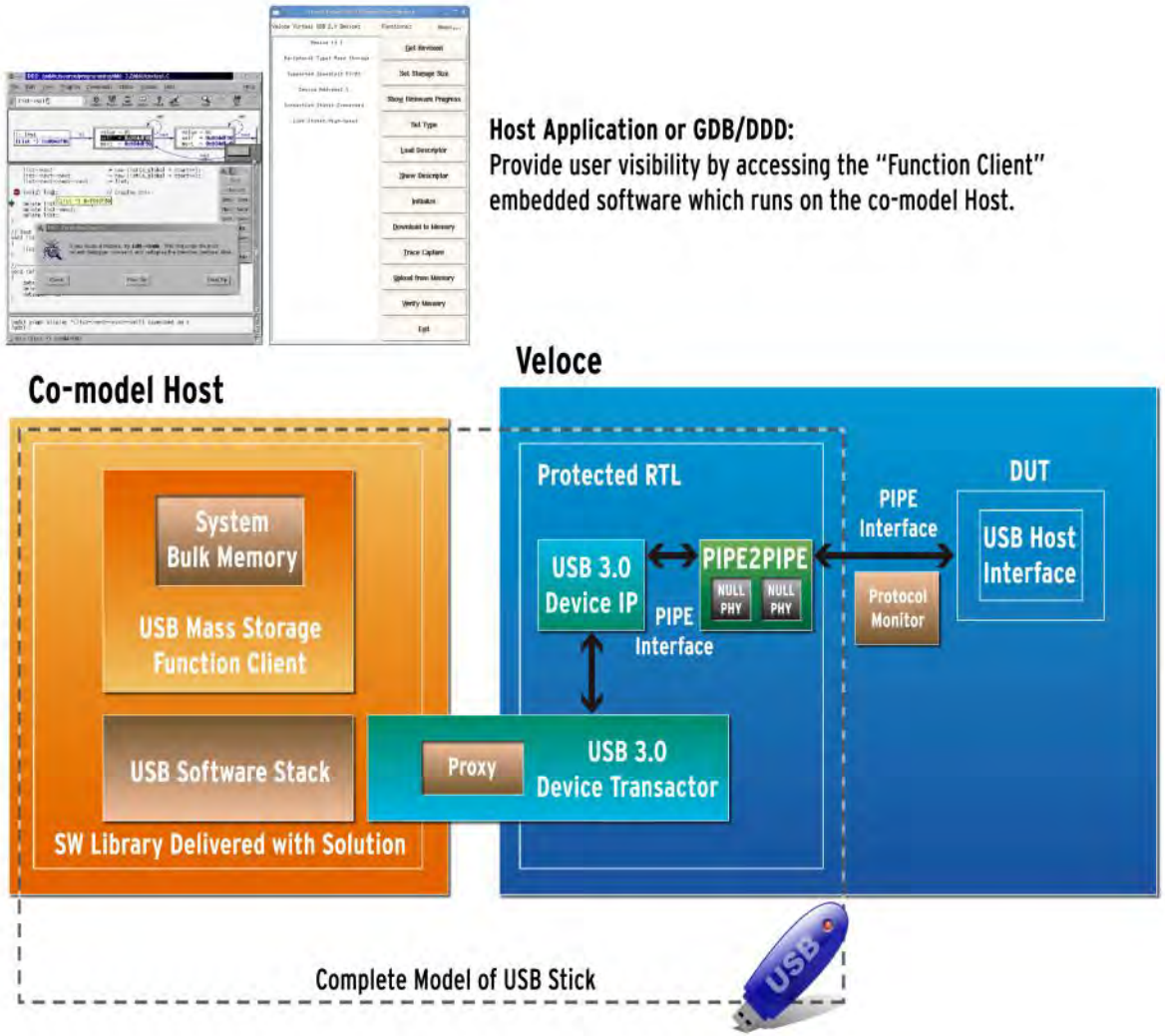
Virtual peripherals

Linux Workstation

- ICE is the traditional approach
- Many peripheral devices available
- Reliability hampered by external HW and cables
- Expensive to replicate for multi-user
- Restricts multi-project access

- Virtual peripherals use the same design IP as ICE solutions, delivering the same speed and accuracy
- Efficient multi-user support, especially useful for many SW engineers
- Quickly and easily reconfigured

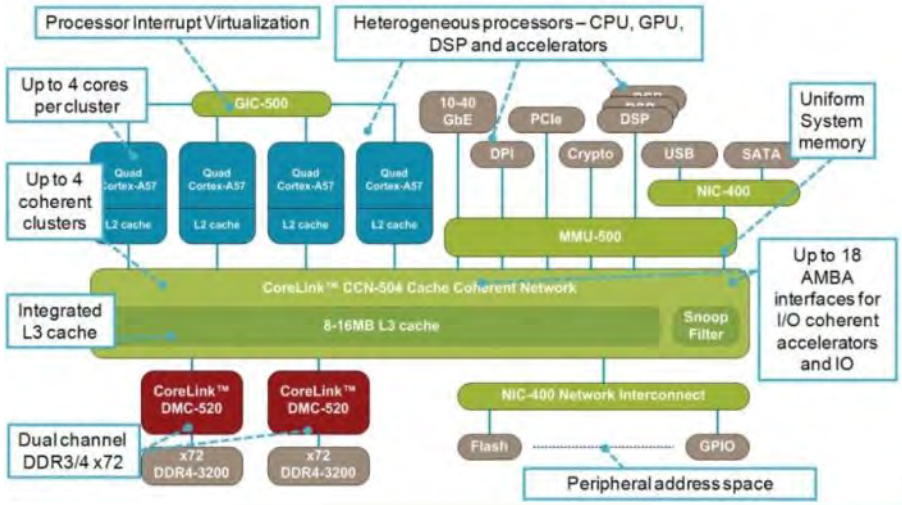
Virtual USB 3.0 Example
Mass storage peripheral device



- Models standard USB 3.0 SuperSpeed mass storage device
- USB mass storage modelled on Linux PC for high-capacity needs
- Supports USB 3.0 PIPE interface to deliver flexible host controller connectivity
- Host application interface provides control:
  - Set-up
  - Initialization
  - Data upload/download
  - Configuration of USB characteristics, e.g.
    - Device speed
    - Descriptors

SoC Validation Considerations

- Software execution target
  - RTL CPU model
  - Fast instruction-set simulator
- Software debug method
  - JTAG on emulator
  - Off line
- Peripheral models as target for device drivers
  - External hardware (ICE)
  - Virtual device models



THANK YOU