

RAMP Tools & Infrastructure Breakout Report Winter 2008

Greg Gibeling
1/17/2008

Group: Andrew Putnam, Mark Oskin, Michael Papamichael, Angshuman Parashar, Mike Kishinevsky

1/17/2008 RAMP Tools & Infrastructure 1

RDF vs RDL

- RDF: RAMP Design Framework**
 - RAMP requires RDF
 - RDF != RDL, Capital mistake to confuse the two
 - Sharing, timing require only RDF
- RDL: RAMP Design Language & RDL**
 - RDL != Greg, Capital mistake to confuse the two
 - RDL will provide benefits, but not required for first pass
 - Could generate shells now
 - Wanted a looser interface

1/17/2008 RAMP Tools & Infrastructure 2

FSL Link

Implementation varies based on Latency, Bitwidth, FIFO Depth

1/17/2008 RAMP Tools & Infrastructure 3

RDF Interfaces (1)

- Write Side (Unit to Channel)**
 - Forward: WriteValid, WriteInvalid, Nop
 - Backward: Full, NotFull, Stall
- Read Side (Channel to Unit)**
 - Forward: Empty, NotEmpty, Stall
 - Backward: ReadValid, ReadInvalid, Nop
- Bottom Line**
 - Need three value logic to separate host & target
 - Probably two wires
 - Independent of actual signalling convention
 - Allows unit & timing model development absent RDL

1/17/2008 RAMP Tools & Infrastructure 4

RDF Interfaces (2)

1/17/2008 RAMP Tools & Infrastructure 5

Teamwork

- Design Sharing**
 - Many independent, identical implementations
 - RDF & RDL are ways to share work
 - Units (of course)
 - Link/channel implementations
 - RAMP & RDL are intended to be shared projects
- Need more regular communication**
 - Desperate need for constructive criticism
 - Open source mentality, **contribute your work back**
 - Central CVS/SVN?
 - More often mini-retreats?
 - Phone Conferences?

1/17/2008 RAMP Tools & Infrastructure 6