Research Accelerator for Multiple Processors
Architecture, Language & Compiler

Introducing RAMP

• Research Accelerator for Multiple Processors. Originally envisioned as a cross platform architectural simulator. Designed to foster community research (Share & verify results).

• A distributed event simulation & message passing system framework. Orders of magnitude faster than existing solutions. Eases component re-use and integration.

• A modeling language (RDL) is a key step in the realization of RAMP. The “Target System,” the system being emulated, is captured in RDL and emulated on the “Host System” (an FPGA or CPU).

The “Counter” Example

```plaintext
unit {  
  input bit[1];       UpDown;
  output bit[32];    Count;
  }  

unit {  
  output bit[1];     Value;
  attribute C2LinX2; 
  attribute C2LinX2; 
  }  

platform {  
  language "verilog";
  "SerialLink";
  C2LinX2;
  IO::BooleanInput;
  default link  }

map {  
  C2LinX2 CounterExample BasePlatformInst;
  BaseUnitInst;
  C2LinX2Counter;
  }

unit {  
  instance IO::BooleanInput::BooleanInput;
  instance Counter;
  instance IO::Display8Seg::Display8Seg;
  channel fifoPipe[1, 1, 1];  
    InChannel;  
    {  
      BooleanInput.Value -> Counter [Counter];
      channel fifoPipe[32, 1, 1];  
    OutChannel;  
    {  
      Counter.Count -> Display8Seg.Value ;
      CounterExample;
    }
  }
```

The “Target System” Model

• Units communicate over point-to-point, unidirectional channels. A unit would be ~10,000 gates (Processor + L1 cache) Units are implemented in the “host” language, eg. Verilog

Existing message passing hardware/software can be ported easily

• Channels include a delay model

  Allows timing simulations

  Statically typed, variable size messages

  Bitwidth (Fragment)

  Latency Buffering

RDL Compiler Toolflow

• A simple unit

  Three ports

  Unstructured messages

• No functionality in RDL

  unit {
    input bit[1]; A;
    input bit[32]; B;
    output bit[12]; C;
    }  

  Unit (Shell)

  →

  RDLC (Map)

  →

  Xilinx XUP

  →

  Berkeley C2LinX2

  →

  Standard Compiler

  →

  Any computer

  →

  BEE2

• RDL Hierarchical Netlist

• Include channel model

• Flexible, extensible back end

• Support for multiple target languages

• Automatic cross platform implementation

• Complete host implementation