An Overview of IEEE 1394 Simulation in ns2

Firewire & ns2

Glendon Holst
gholst@cs.sfu.ca

Spring 2002
Final Project Presentation
CMPT 883-3: Special Topics: High-Performance Networks
Outline

Summary
- Protocols and Agent/IEEE1394, LL/IEEE1394
- Arbitration and MAC/IEEE1394
- Declaring Topologies

Overview of IEEE1394 in ns2
- Timers and Simulating Events
- Nodes, Classifiers, Links, MAC, LANNodes, Agents...

Introduction to ns2 Simulation Concepts
- Packets
- Protocols (Arbitration, Asynchronous, Isochronous)
- Topology and Configuration
- Specifications

Introduction to IEEE1394 (Firewire, iLink)
Introduction to Firewire

Topology and Configuration

- IEEE1394-1995, IEEE1394a, IEEE1394b (based on IEEE1212)
- Serial-Bus (Packet based and network-like)
- Tree Topology (Identify root node)
- Node Identification on Bus-Reset (Sticky ID)
- Hot-Plugable (Plugging produces a bus-reset)
- Highspeed (Nodes may add or use power to the bus)
- Proposed (IEEE1394b): 800 Mbps, 1.6 Gbps, 3.2 Gbps (OC ~ 100 M)
- Currently: 100 Mbps, 200 Mbps, 400 Mbps
- Experimental IP over IEEE1394
- Well suited for video, hard-drives, cameras, printers, scanners.
- Primarily Connects Peripheral Devices
- Asynchronous and Isochronous Transfers modes.
- Peer-to-Peer (Unlike USB which requires a host CPU)
- Powered (Nodes may add or use power to the bus)
- Topology and Configuration

Summary

Primarily Connects Peripheral Devices
- Well suited for video, hard-drives, cameras, printers, scanners.
- Primarily Connects Peripheral Devices
- Asynchronous and Isochronous Transfers modes.
- Peer-to-Peer (Unlike USB which requires a host CPU)
- Hot-Plugable (Plugging produces a bus-reset)
- Powered (Nodes may add or use power to the bus)
- Topology and Configuration

Experimental IP over IEEE1394

Currently: 100 Mbps, 200 Mbps, 400 Mbps

IEEE1394-1995, IEEE1394a, IEEE1394b (based on IEEE1212)
Introduction to Firewire Protocol Overviews


 Arbitration During the Cycle: Nodes request access to bus and the Root


Isochronous packets are NOT acknowledged. 2x packet size

Asyncchronous packets require Ack.

Transfers.

dallowed. Up to 80% of the cycle (100 µsec) can be reserved to isochronous

Asyncchronous packets require Ack.

Subactions. [arb] [data-packet] [ack-gap] [ack] [subaction-gap] [arb] [arb]

Grant's access.

Protocol Overviews
Introduction to Firewire

Arbitration

The root node must win arbitration to send Cycle_Start.

Nodes can only win arbitration once per cycle (arb-won flag), necessarily over all priority.

Nodes closer to the root have positional priority, but not internal. A Grant packet notifies the winners.

The root grants the first request to arrive at its ports (or
and passes the request to its parent.

Node chooses first request to arrive at its ports (or internal),

After time-out (either subaction or bus-idle) send Request to

Arbitration
Introduction to Firewire

Asynchronous Transfer

For Guaranteed Packet Delivery

Transaction Based

Read, Write, Lock

Must arbitrate to initiate a transfer.

Acknowledged: lock, read, write

For Guaranteed Packet Delivery

Asyncchronous Transfer

Max payload size:

- 512 bytes for 100 Mbps
- 1024 bytes for 200 Mbps
- 2048 bytes for 400 Mbps
- 4096 bytes for 800 Mbps, etc.

Requested data

Requested node must then arbitrate for access before sending the requested data. The Ack must return within subaction time. The requested node can then return the result.

Split Transfer: The requested node cannot return the result in time, so an Ack is sent indicating a delayed response. The requested node can then return the result. The Ack must return within subaction time.

Concatenated Transfer: The requested node can return the result fast enough, so the data is appended to the Ack.

For read, there are two possible results:

- Ack must return within subaction time.
- Must arbitrate to initiate a transfer.

For write, only an Ack is required.

Transaction Based

For Guaranteed Packet Delivery
Introduction to Firewire

Isochronous Transfer

For Guaranteed Time-Slice (bandwidth).

Max payload size:

- Isochronous arbitration phase of the cycle begins.
- After these packets are transmitted (in order of channel number), the
  channels identify target nodes. 64 channels.
  Packets transmitted after the Cycle_Start packet.
  Channels identify target nodes. 64 channels.
  Resource Manager (root).

- Must obtain channel and bandwidth from Isochronous
  Transfer.

- For Guaranteed Time-Slice (bandwidth).

Mbps, 8192 for 800 Mbps, etc.
- 1024 bytes for 100 Mbps, 2048 bytes for 200 Mbps, 4096 bytes for 400
Introduction to Firewire

Packet

Header

- Last Quad-Word is Data-CRC.
- Data Payload Length is limited by port speed.
- Packet is padded mod 4 bytes (quad-word).
- Header is 20 bytes

Packet

Addressing

- Header CRC (4 bytes).
- Packet Type (4 bytes).
- Destination Offset (6 bytes).
- Source ID (2 bytes).
- It (trans label), It (retry), Trans Code (trans code), Priority (2 bytes).
- Destination ID (2 bytes).

- Node ID: 10 bits for Bus, 6 bits for Node on bus.
- Node ID: 10 bits for Bus, 6 bits for Node on bus.
- Destination: 48 bits, memory address location on target node.
Introduction to ns2

Simulation Concepts

Need to understand so we can later map IEEE 1394 to ns2.

Nodes, Classifiers, Links, and Agents.

Outgoing link to send the packet along.

○ If the packet is for another node, the classifier determines which
  ○ This is done by another classifier.
  ○ If the packet is for this node, it must be sent to the appropriate agent.
  ○ Another node, a packet classifier determines this.
  ○ When packets arrive at nodes, they are either for that node or for
    ○ A node is like a vertex, a link is like an edge.

Simulate Concepts
Introduction to ns2
Introduction to ns2

- Lan Nodes
  - Nodes are suited for point to point communication.
  - We need to simulate issues of bus contention and arbitration.
  - LanNodes let us capture Link Layer, MAC layer, and even PHYs layer behaviours.
  - Nodes remain as the higher level network elements.
Introduction to ns2
Introduction to ns2

Packets

All packets supported by the system are sent as a Packet.

Packets only need to contain sufficient state to simulate the behaviour of real packets.

Packets supported by the system are sent as a Packet.

例：

- Simulating Link delays, etc.

- When the timer expires, the event occurs and the method is invoked.

- To simulate events: acquire timer, register event (class and method to invoke).

Timers and Events

We use address information from the IP packets to simulate IEEE 1394 addresses.

Packet data is simulated via a size variable.

They don't need to exactly match real packets.

包

Packets supported by the system are sent as a Packet.
Firewire in ns2

Topology

Require user to set up network with the following attributes:

Example Tcl Simulation Script:

```tcl
set lan0 [ns make-lan "$n0 $n1" 400Mb 1ms LL/ieee1394 Queue/DropTail Mac/ieee1394]
set lan1 [ns make-lan "$n1 $n2" 400Mb 1ms LL/ieee1394 Queue/DropTail Mac/ieee1394]
set lan0 [ns make-lan "$n0 $n3" 400Mb 1ms LL/ieee1394 Queue/DropTail Mac/ieee1394]
set lan1 [ns make-lan "$n1 $n4" 400Mb 1ms LL/ieee1394 Queue/DropTail Mac/ieee1394]
```

*Tree Structure*
Parents have lower id numbers than their children (i.e., root is lowest numbered node).

Example:
Require the following attributes:
Firewire in ns2

Arbitration and Mac/IEEE 1394
Wait for end of Subaction timeout.
If Arb Won is NOT set, then Request Arb from Parent.
Arbitration and Mac/IEEE 1394

Protocols and Agent/IEEE 1394, LL/IEEE 1394
Agent creates packets (determines size, transfer type, destination).
Agent handles Ack.
Agent ignores Split-transactions.
If granted Arb, then set Arb Won flag.
Listen for Cycle Start, and reset Arb Won flag.
If Guaranteed Arb, then set Arb Won flag.
Arb may be for Asynchronous or Isochronous transfer.
If Arb Won is NOT set, then Request Arb from Parent.

Protocols and Agent/IEEE 1394, LL/IEEE 1394

Arbitration and Mac/IEEE 1394

Firewire in ns2
Summary

Introduction to basic operation of IEEE1394

Introduction to simulation concepts and classes in ns2.

Original Goals - See Demo

Mapping IEEE 1394 protocols to ns2.

Comparing IEEE 802.3 vs. IEEE 802.5 (Ethernet vs. Token-
ring) would be more informative (but I wanted to get "under-the-
hood") of ns2 and learn more about Firewire.

The whole point of the project was to evaluate bandwidth allocation of asynchronous vs. isochronous transfer modes.

The mode to use is primarily determined by the application.

Original Goals - See Demo

Mapping IEEE 1394 protocols to ns2.

Comparing IEEE 802.3 vs. IEEE 802.5 (Ethernet vs. Token-ring) would be more informative (but I wanted to get "under-the-
hood") of ns2 and learn more about Firewire.

The whole point of the project was to evaluate bandwidth allocation of asynchronous vs. isochronous transfer modes.

The mode to use is primarily determined by the application.
References

- IEEE 1394-1995 High Performance Serial Bus (Overview):
- New Technology for 1394 (Overview): 1394 rehabreview.pdf
- Isochronous Resource Management br062r00.pdf
- What's New About 1394b: ppt1.pdf
- IEEE 1394-1995 High Performance Serial Bus (Overview):
- ISO-2 Document Implementation (For Implementation Purposes): ns-doc.pdf