SAND2011-5935C

NEW TECHNIQUE FOR SPACEWIRE NETWORK DISCOVERY

Session: Networks and Protocols

Long Paper

Kody D. Mason, Justin W. Enderle
Sandia National Laboratories, PO Box 5800, Albuquerque, NM 87185-0968

E-mail: kmason@sandia.gov, jwender@sandia.gov

ABSTRACT

Early techniques used to discover the topology of a dynamic SpaceWire network have typically relied on prior knowledge of some protocol implementation. Systematically generated request messages, when responded to by each routing switch or end-node, facilitated discovery. The challenge today, however, is to discover and map network topology without relying on any one protocol implementation - or even any SpaceWire protocol. By exploiting the design of SpaceWire routing switches, discovery is possible on dynamic, heterogeneous SpaceWire networks using the concept and technique of looping messages back to oneself. Exploring the advantages and implications of such a viable technique may lead to a new standard for network discovery.

1 NETWORKS AND NODES

Using the terms and definitions from the European Cooperation for Space Standardization (ECSS) Glossary, and building upon the SpaceWire foundation [1], the notion of a dynamic SpaceWire network is one in which the links between routing switches and nodes can be added or removed in a *Plug-n-Play* like fashion. When links between routing switches are manipulated, the topology of the SpaceWire network changes. When links between nodes and routing switches are manipulated, packet sources and destinations appear and disappear.

This paper will begin by differentiating between *Network Discovery* and *Node Discovery*. The former involves the systematic probing for SpaceWire routing switches, and the latter involves polling switches for links to nodes, and then identifying such.

When probing for routing switches, early network discovery techniques typically relied on each routing switch's configuration port to respond to identification requests to confirm the routing switch's presence. A request packet was typically dispatched to the configuration port, and a response packet provided confirmation of existence.

This same request/response approach was generally used for node identification as well. Dispatching one or more requests to an active link (which might be node or another routing switch) could produce a response if a node was present and it understood the protocol.

Recent proposals, such as the SpaceWire PnP Protocol Definition Draft [2], put forth basic *Service Definitions* for *device identification*, *network management*, and *link* and *router* configuration. This paper will blur the boundaries between Network Discovery and routing switch configuration. Link configuration (particularly speed) is assumed to be automatic or take place prior to physical link connection.

2 NEW PROBING TECHNIQUE

Per the SpaceWire PnP Draft [2], "SpaceWire does not offer a standard mechanism for detecting the topology of a network." One aim of this paper is to propose such a standard.

The new probing technique involves a shift away from the request/response model. Rather than dispatching a request to some possible physical-path-address on the network, and awaiting a response from a packet receiving/processing/replying entity, a single packet is addressed with a round-trip physical-path-address that will essentially "loop" through a possible routing switch and be returned to the originator with all path-addressing bytes removed along the way out and back.

Perhaps the best way to visualize this technique is to think of the SpaceWire routing switch as a "roundabout" intersection with a vehicle (packet) both entering and exiting the roundabout at the same point.

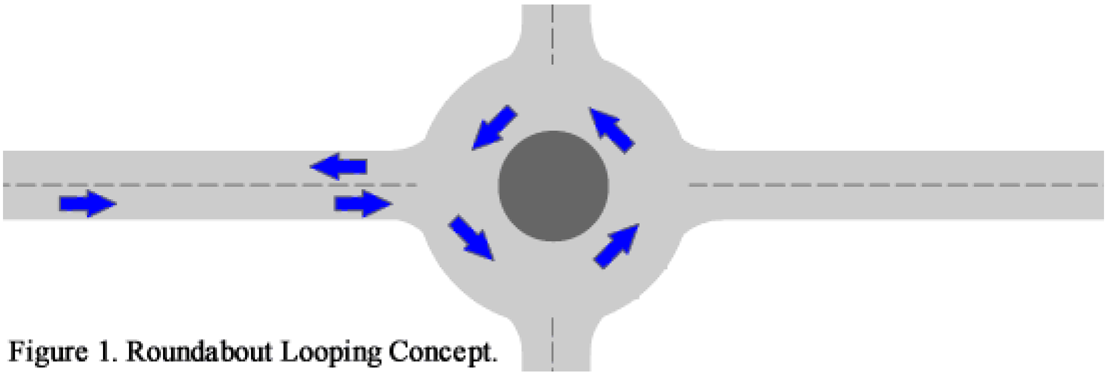

Figure 1. Roundabout Looping Concept.

The significance to the probing entity is that if it receives the recognizable payload portion of a packet back, then that round-trip physical address is valid in most cases.

To more explicitly reiterate this technique, consider a node acting as a probing entity connected to routing switch A's port five. Switch A's port three links to switch B's port two, and switch B's port four links to switch C's port one. Therefore, the physical-path-address from the probe to switch C is "34", and the return path is "125".

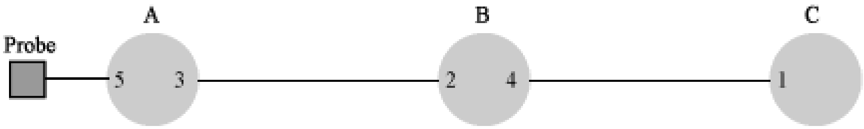

Figure 2. Example of Round-Trip Physical-Path *Addressing*: "*34125*"

By addressing a probe packet, [PACKET], with “34125”, then the probe node will receive back [PACKET] after it *loops* through switch C. Notionally, switch C’s port one (1) is the “turn-around point” or the “turn-around port.”

The Network Discovery process is typically breadth-first. General practice is to begin probing one link (or “hop”) from the probing node, then as routing switches are discovered, a new list of potentially viable physical-paths is generated for one hop beyond that. Probing can be stopped when the hop count reaches a point where the new potentially viable list yields no results.

2.1 BREADTH-FIRST PROBING

Recall that SpaceWire physical-path-addressing uses addresses in the range of one to thirty-one (1-31.) A probing entity can discover its own port number on its routing switch with a single-byte physical address preceding its probe packet payload. From Figure 2, the packet containing “5[PACKET]”, when written, will cause “[PACKET]” to be read back.

Round-trip physical-path-addresses are always an odd number of bytes. The iteration technique, when generating the potential list of addresses for the next hop count, involves inserting different pairs of port numbers just before the turn-around point of each known round-trip-address at the previous hop.

For example, if the list of known round-trip-addresses for hop number two (hop #2) was simply “325”, then the initial potential list for hop number three (hop #3) would be:

a) 31125
b) 31225
c) 31325
d) 31425
e) 31525
f) 31625
g) 31725
h) 31825
i) ...
lll) 38825

where the maximum port to be probed is either thirty-one (31) or an implementation-defined maximum. From the list above, the maximum port to be probed for is eight (8.) However, generating possible round-trip-addresses is subject to certain pitfalls (see section 2.3.)

2.2 BASIC ROUTER IDENTIFICATION REQUIREMENT

Discovering physical-path-addresses that indicate a potentially valid round-trip path through a routing switch is the first step in mapping a network topology. In order to be able to accurately create a topology map, some unique indicator must be available to identify routing switch instances in order to distinguish newly discovered switches from ones previously discovered through other physical paths.

Since the SpaceWire routing switch design has a configuration component, a request for a router ID is one method of routing switch identification. A potential *best practice* for hardware designers is to allow a hardware component to be used to set a unique default ID per router (not unlike the purpose of a "MAC" Address for an Ethernet "PHY".)

Another option for identification involves using the Remote Memory Access Protocol (RMAP) [3] to read an identification number or string from a non-volatile memory location. The SpaceWire PnP Draft [2] proposes something even more advanced.

2.3 POTENTIAL PITFALLS

Probing in the manner described above is subject to several pitfalls. These pitfalls fall into three basic categories: discovery logic, routing switch design, and node robustness.

2.3.1 COINCIDENTAL RETURN PATHS (DISCOVERY LOGIC PITFALL)

As mentioned above, the receipt of a recognizable probe *payload* does not guarantee that the round-trip physical-path-address actually looped at a turn-around point in a routing switch. There is a chance that, by coincidence, two separate return paths coming back from the routing switch are identical except for the turn-around port, itself. In this case, the fact that two probe packets (with identical outbound paths) successfully made their way back to the probing entity is the clue necessary to identify this situation and trigger further analysis. One return path completes the loop-back through the routing switch, but the other return path flows through different links back to the probing entity.

Referencing Figure 3, two probe packets addressed as "12115" and "12415" will both be returned to the probe entity. Likewise, two others addressed as "42115" and "42415" will also. When only the turn-around port is different in the round-trip path-addresses, the coincidental path should be discarded. Determining which one should be discarded requires confirming the identity of the switch one hop prior to the suspected turn-around point. In the case of the "12115" and "12415" pair, confirming that the identity returned by addressing "1" (switch B) matches that returned by "124" (also switch B) is required to know that "12415" is the one to keep, and "12115" is the one to discard.

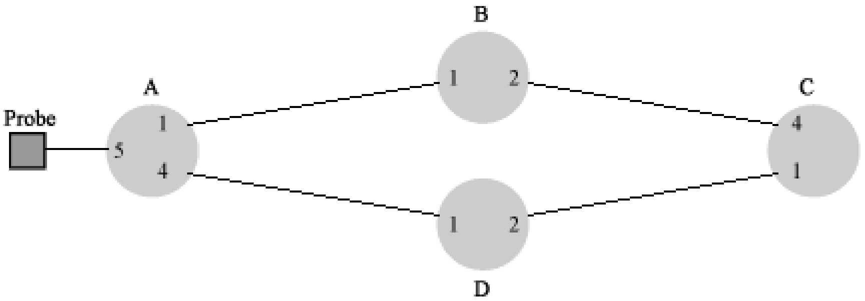

Figure 3. Coincidental Return Paths.

2.3.2 ECHOING (DISCOVERY LOGIC PITFALL)

In the course of generating the list of potentially viable round-trip paths at the next hop count, care must be taken not to "echo" back and forth between two routing switches. For example, as depicted in Figure 4, if a discovered round-trip path is “34125”, then the temptation to probe for “34**14**125” should be avoided.

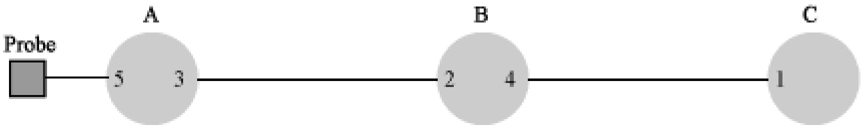

Figure 4. Example of Echoing: “3414125”

2.3.3 NEVER BACKWARDS (DISCOVERY LOGIC PITFALL)

Even more general than the echoing pitfall is the condition when generating the list produces any next hop round-trip path-address where the next outbound port (at the next hop) matches the previous turn-around point. As an example, consider the new potential path of “34**1x**125” (where ‘x’ is anything.) As long as the bolded ‘1’ matches the ‘1’ in “125”, the route will bring the packet backwards (closer to the probe.)

2.3.4 INACTIVE OR NON-EXISTENT PORTS (ROUTING SWITCH DESIGN PITFALL)

As a probing entity transmits its discovery packets across the network, routing switches will invariably receive packets physically addressed to ports that are not active, or do not even exist. Depending on the routing switch design, an attempt to remove the next physical-address-byte and write the remaining packet to such a port could cause a router lockup. One *best practice* for a SpaceWire routing switch design is to always silently drop packets destined for inactive or non-existent ports.

2.3.5 BUFFER LIMITATIONS (ROUTING SWITCH DESIGN PITFALL)

So far, little has been mentioned regarding the contents of the probe packet payload – the bytes that find their way back to the probing entity indicating that a potentially valid round-trip address was discovered. The issue at hand is not so much what the probe packet *payload* contents is, but rather how large it is.

Using the roundabout analogy presented earlier, suppose that a large truck is pulling three large trailers as it attempts to circum-navigate the roundabout. Before the third trailer enters the roundabout from the side street, suppose the front of the truck runs into it. The SpaceWire routing switch design may limit the number of bytes that can be buffered while a packet is retrieved from a port and then written back to it. To minimize the likelihood of such an occurrence, very small *payloads* should be used in the probe packets.

Note: Since the number of bytes which have to loop through the routing switch include both the return-path portion of the address and the *payload*, then the buffer size used in the routing switch design is the key to determining the maximum number of “hops” that can be discovered with this technique.

2.3.6 PACKET PARSING ERRORS (NODE ROBUSTNESS PITFALL)

This new technique for Network Discovery can create a manageable “storm” of probe packets on the SpaceWire network. The *blast intervals* and delays between packet transmissions are easily configurable within the probing entity; however, the effects of all these physical-path-addressed probe packets on nodes could be problematic.

As potentially viable probe packets find their way across links from routing switches to nodes, the nodes may encounter bytes from the physical-path-address or from the probe packet *payload* contents. These bytes may fall where a SpaceWire protocol byte is expected. Nodes have the potential of misinterpreting these packets (if they appear to be a recognized SpaceWire protocol), or in other cases, nodes may fail to disregard these packets (if they appear to be an unknown or unsupported protocol.)

Although on the surface, this new Network Discovery technique appears to introduce the risk of node failures, it actually can have the opposite effect. By requiring this discovery technique to be used during the design and testing of routing switches and nodes, the entire network can be tested for a higher level of reliability and robustness before final implementation.

2.4 COMPLETING NETWORK DISCOVERY

When the probing process is completed, a *results table* will contain all valid round-trip physical-path-addresses and corresponding router identities. Multiple rows may be found for any router identity signifying multiple paths to the router. At this point, a logical addressing scheme can be used to compile route tables. These tables can be generated with any desired regional addressing supplement. Note that section 2.6 contains a method for consistent logical address assignment based on the concept of affinity.

Routing switches may be partially configured now. Specifically, switch-to-switch logical address routes may be inserted into all *route tables*. Node Discovery is now possible using either physical-path or (routing switch level) logical addressing combined with (node level) physical addressing.

Finally, the *results table* can be used to dynamically visualize the network. Depicted are the probing entity (blue), and routing switches from two separate vendors (red, and green.) Presumably, the identification of routing switches may have involved more than one technique (per section 2.2.)

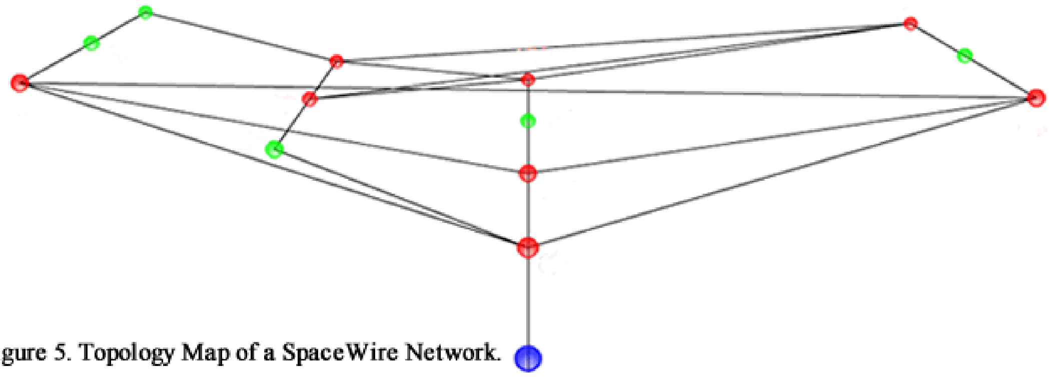

Figure 5. Topology Map of a SpaceWire Network.

2.5 POLLING FOR NODES

The process of *Node Discovery* involves the systematic polling of nodes for management information. Node Discovery requires that each node receive and process a request packet, then respond.

As of the writing of this paper, the authors are unaware of an adopted standard in the SpaceWire community to address Node Discovery in a multi-vendor, heterogeneous SpaceWire network.

A proposal to adopt an Internet standard, such as the Simple Network Management Protocol (SNMP), could remedy the situation. Specifically, adoption of SNMPv1 [4] as a SpaceWire-supported protocol with a minimal required implementation of the "System" group from RFC-1213 [5] could enable standardized Node Discovery as well as provide a single technique for routing switch and end-node identification. Such adoption may be consistent with one of the aims of the SpaceWire PnP Draft [2] to "leverage existing technologies as much as possible."

2.6 Logical Address Assignments – Affinity

The notion of *affinity* (of a SpaceWire logical address to a particular switch or node) can be borrowed from the *plug-n-play* behaviour of many computers and personal computing devices. Consider how portable storage devices or serial communications devices are often managed when they are attached to a computer:

For example, upon the first attachment of a USB modem to a personal computer (PC), the USB *plug-n-play* device manager will determine the device type and serial number of the modem. If this specific device is not listed within a registry, then it is assigned the next unused "COM" port and added to the registry. In the future, each time the device is subsequently attached, its registry information is used to re-assign the same "COM" port as before, so the device has an affinity to a particular port number. The rationale for this behaviour is that humans will naturally remember which COM port is which over time, and humans will want consistency in assignments.

Another example of affinity is the manner in which Dynamic Host Control Program (DHCP) servers typically assign Internet Protocol (IP) addresses. When a request for an IP address is made, most DHCP servers will attempt to re-assign one that was last used by the requesting MAC if that IP address is not already in use.

This same notion applies to dynamic *plug-n-play* SpaceWire networks. When a new routing switch or node is discovered, the probe entity can assign the next unused logical address for the region. If the probe has a means to persistently save the identity of the discovered switch or node, along with its newly assigned logical address, then subsequent re-discoveries of the same entity can result in consistent logical address re-assignment.

3 Summary

The techniques described above for *Network Discovery* and *Node Discovery* are indeed different. While the request/response type of discovery technique is required for node discovery, the benefits of using round-trip physical-path-addressed SpaceWire packets to discover routing switches are many. Chief among them is not relying on packet processing entities to support (understand) one or more SpaceWire protocols. Essentially, if a routing switch has active links on the network, and it is functioning with a unique identity, then it can be discovered and mapped through its switch-to-switch links.

4 REFERENCES

1 ECSS-E-ST-50-12C, European Cooperation for Space Standardization, “SpaceWire – Links, nodes, routers and networks”, 31 July 2008, 15-22.

2 SpW-PnP-PD, Space Technology Centre, School of Computing, University of Dundee, “SpaceWire PnP Protocol Definition, Draft A Issue 21” 16 September 2009, 16-17, 41.

3 ECSS-E-ST-50-52C, European Cooperation for Space Standardization, “SpaceWire – Remote memory access protocol”, 5 February 2010, 13-15.

4 J. Case et al., RFC-1157, “A Simple Network Management Protocol (SNMP)”, May 1990, 2-33.

5 K. McCloghrie et al., RFC-1213, “Management Information Base for Network Management of TCP/IP-based internets: MIB-II”, March 1991, 10-13.

Unclassified Unlimited Release (UUR) SAND 2011-5935C

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.