SpaceWire Network Packet Error Handling

Session: SpaceWire Networks and Protocol

Christopher T. Dailey
Dell Services Federal Government, Fairfax, Virginia, USA
E-mail: Christopher.T.Dailey@nasa.gov

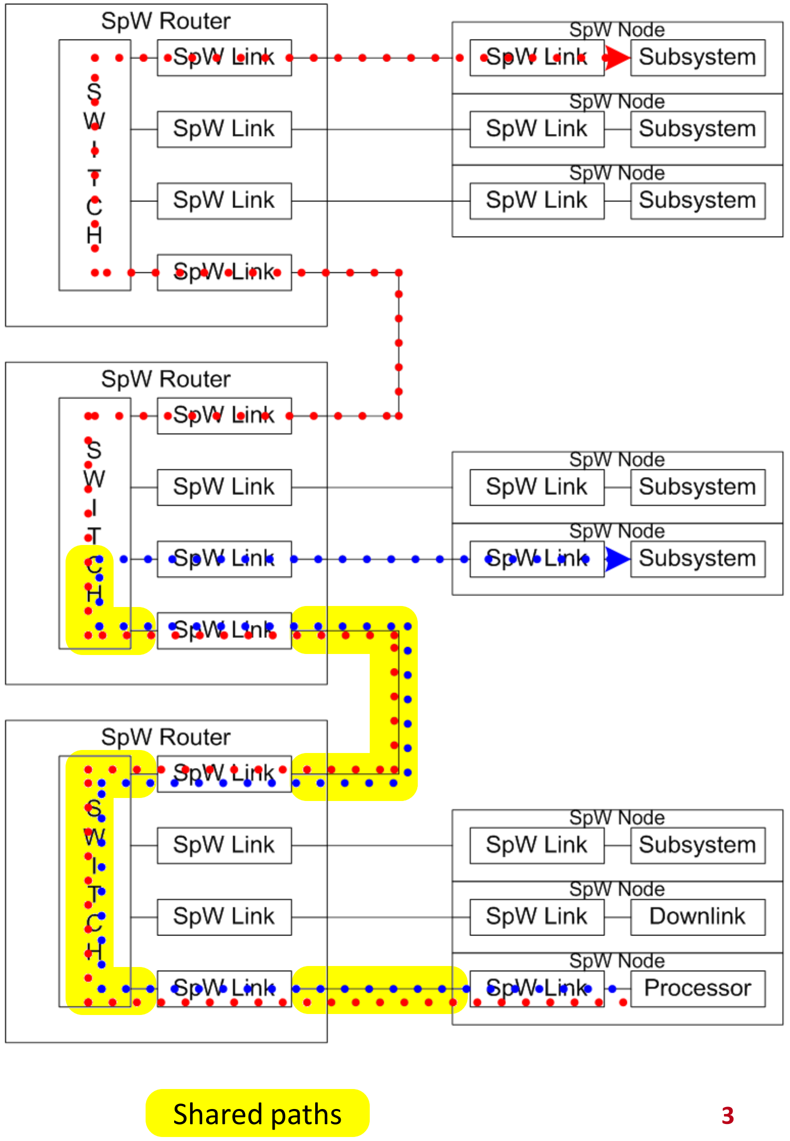
Michael W. Pagen
MEI Technologies, Inc., Seabrook, Maryland, USA
E-mail: Michael.W.Pagen@nasa.gov

Abstract

- SpaceWire (SpW) is becoming commonly used for communication networks between and within spacecraft subsystems
- Due to SpW's unbounded packet size and wormhole routing, some faults in a subsystem could propagate through a SpW network, disrupting other packets or possibly the entire network
- Packet error handling is an essential aspect for reliable, fault tolerant SpW networks
- These faults must be mitigated at the network level as these cannot always be mitigated at the transport or application levels
- The packet error handling logic was revised in the NASA GSFC developed SpW Router FPGA to automatically preclude packet fault propagation

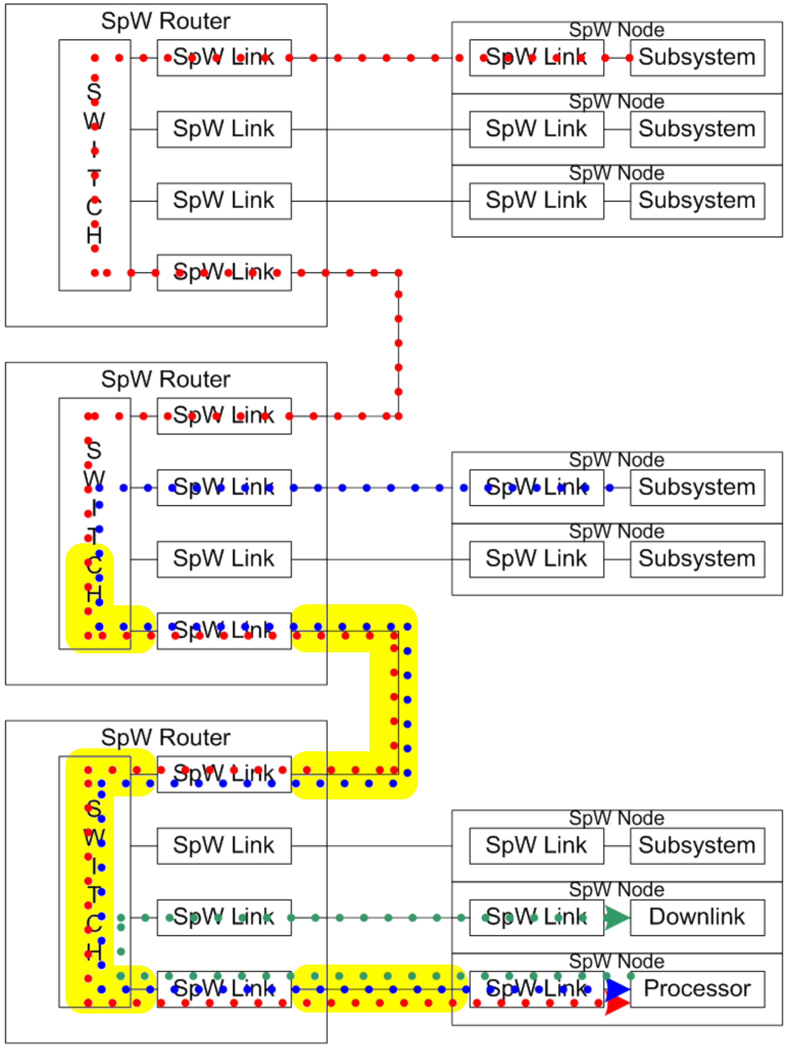
SpW Network Traffic

- **SpW networks can have any topological form**
  - Including loops
- **SpW packet traffic typically resembles funnel shapes**
  - One source flows packets out to multiple destinations such as
    - Onboard Processor sends command packets to multiple subsystems
  - Multiple packet streams merge to one destination such as
    - Science packets flowing from multiple instruments to an SSR
    - Telemetry packets flowing from multiple subsystems to the onboard processor
- **Shared paths are used in funnel-shaped traffic**



Shared Paths

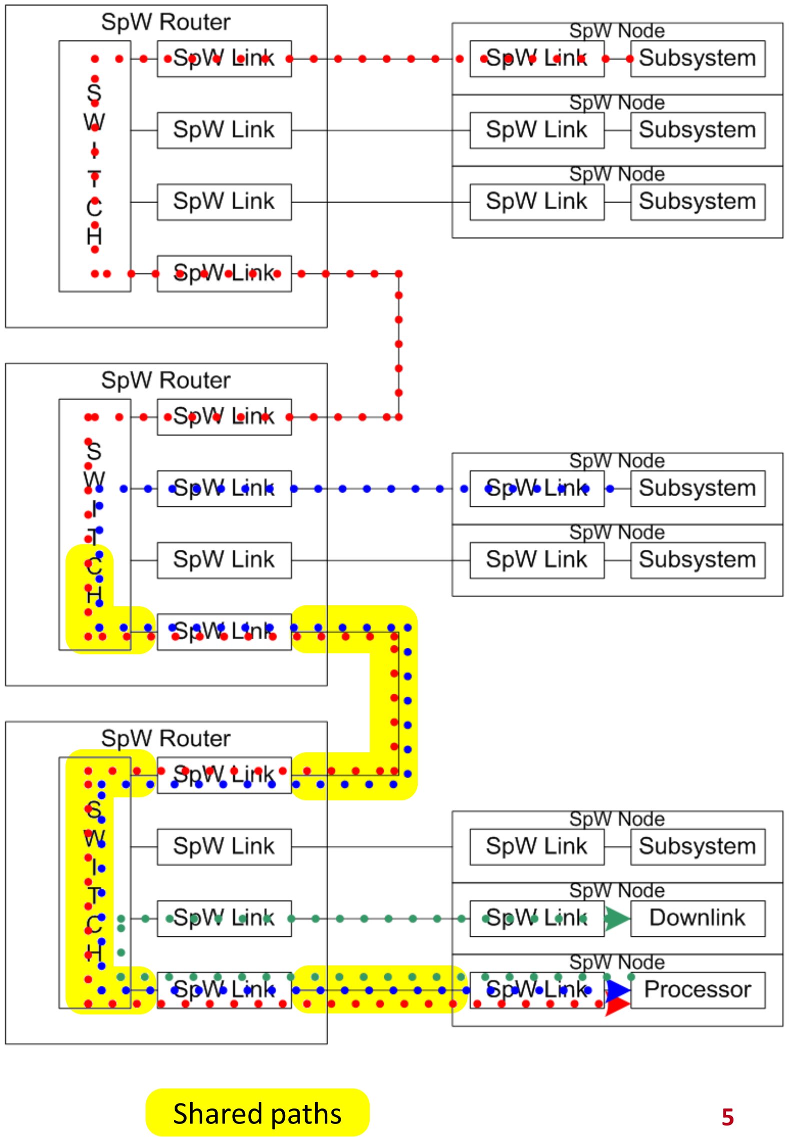
- **Shared paths can propagate faults**
  - If one source or destination fails such that a packet takes too long to wormhole through a shared path then
    - Brief stalls are normal consequences of packet funneling which contribute to packet latency through a network and are not faults
  - Other packets waiting to use the shared path(s) are precluded from doing so
    - Propagating the fault from one board or subsystem over the network to other boards within the subsystem and/or to other subsystems



Shared paths

Shared Path Fault Types and Detection

- **A packet can take too long to wormhole through a shared path due to**
  - A fault in the source that increases the packet size to be too long or infinite
  - A fault in a source that stops sending a packet for too long, without ending the packet with an EOP or EEP
  - A fault in a destination that starts receiving a packet then stops for too long
- **Result is a stuck-open wormhole**
- **Stuck-open wormhole detection**
  - Maximum packet size limit check
  - Packet timeout limit check

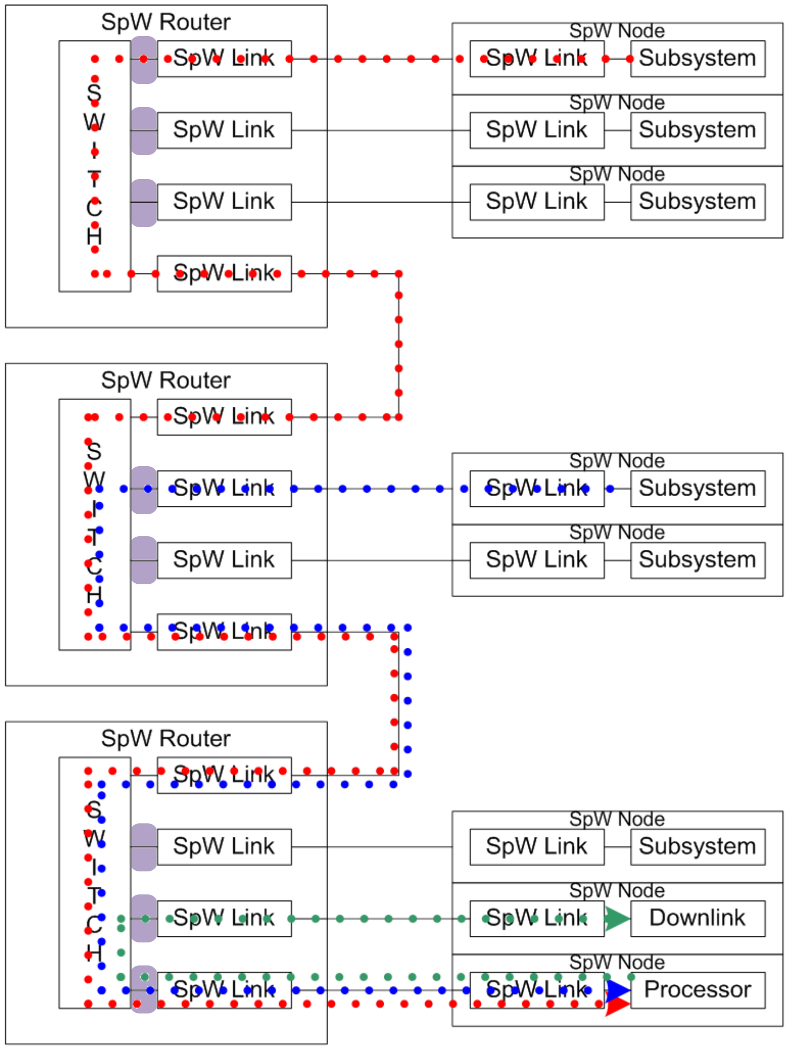


Packet Error Handling for Shared Path Faults

- **Disconnect the path between the source and destination switch ports inside the router**
  - Subsequent packets from other sources can then arbitrate for the destination port, thereby precluding fault propagation from faults in packet sources
- **Discard the remainder of the packet from the source port by draining the source FIFO until an EOP or EEP is found**
  - This may never complete if the source is sending an infinitely long packet or has stalled sending a packet
  - If the source completes sending the packet (with an EOP or EEP) then subsequent packets from this source can arbitrate for destination ports in the routing switch
    - This would be the case if the fault did not occur in the source or the source was able to recover from the fault
- **Truncate the packet at the destination port by appending an EEP**
  - If the destination port has failed to read the packet before the timeout value then the destination port is marked as failed and any subsequent packets requesting this port will be discarded
    - This allows any subsequent packets arriving through the source port to arbitrate for other destination ports, thereby precluding fault propagation from faults in packet destinations
    - If the fault in the destination is fixed such that its destination port FIFO in the router is read then the destination port's fail flag is automatically cleared and the destination port can resume receiving new packets
- **Set the appropriate error status**
  - To indicate which autonomous action was taken

Where to Apply Packet Error Handling

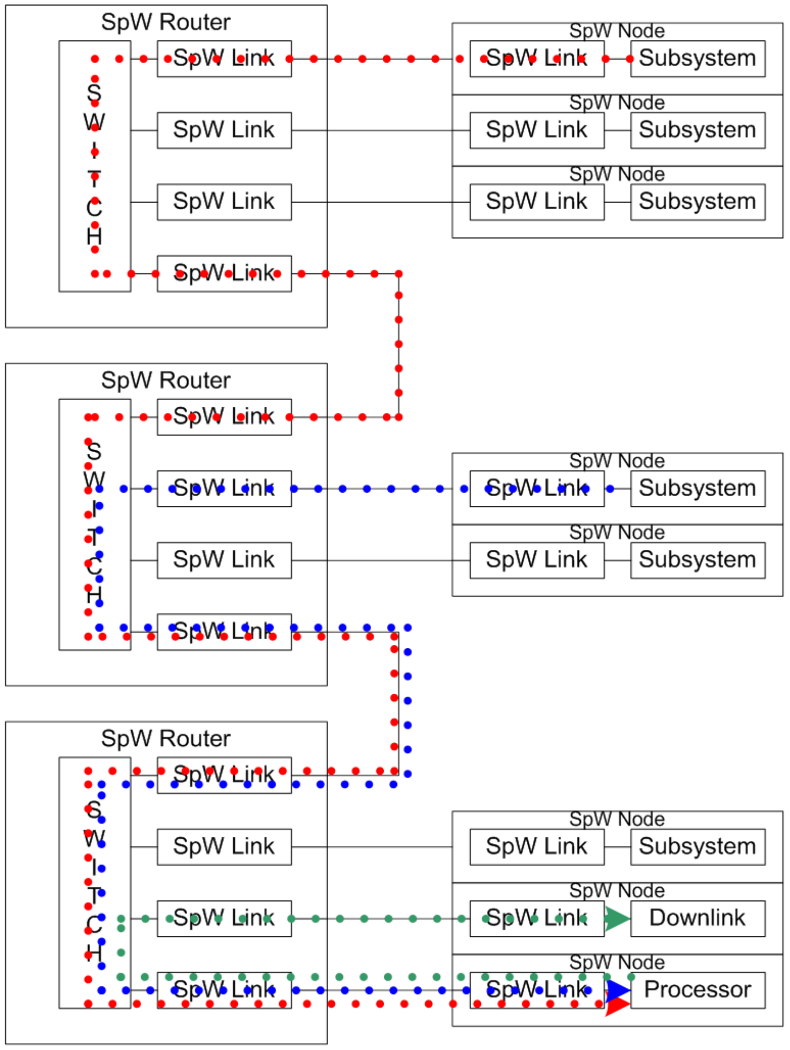
- **Packet error handling should be performed as close as possible to the cause of the fault**
  - At the network switch boundaries, where packets begin or end their wormhole paths through one or more shared paths in routers
- **Packet error handling could be performed at intermediate points inside the network switch boundaries but doing so**
  - May propagate faults as packets may be discarded that aren't from the errant source or destination



Network Switch Boundaries

Why Packet Error Handling at the Network Level

- **Since packets can be of any size, there is no time limit on wormhole routing in the SpW standard**
- **A wormhole that remains open does not violate any of the "rules" for the physical, signal, character, exchange, packet or network levels**
  - In fact the rules are followed in order for the wormhole to remain open
  - Links remain in the RUN state but only
    - Pass NULL characters (stalled packet) or
    - Pass bytes of data (infinite packet) but
    - Not an EOP or EEP.
- **Mitigating at the transport or application levels, such as automated (watchdog) or operator initiated resets or power cycles, may or may not stop fault propagation**
  - Depending on whether the transport or application level can affect the cause of the stuck-open wormhole



Conclusion

- Faults in a source or destination of a packet can result in stuck-open wormhole routes
- Stuck-open wormholes can disrupt other or all packet flows, propagating faults over a SpW network
- Stuck-open wormholes are best mitigated at the network level
- Configurable packet error handling logic was revised in the GSFC SpW Router FPGA to detect and autonomously mitigate stuck-open wormholes
- This logic has been simulated and tested, and works as intended to preclude packet fault propagation