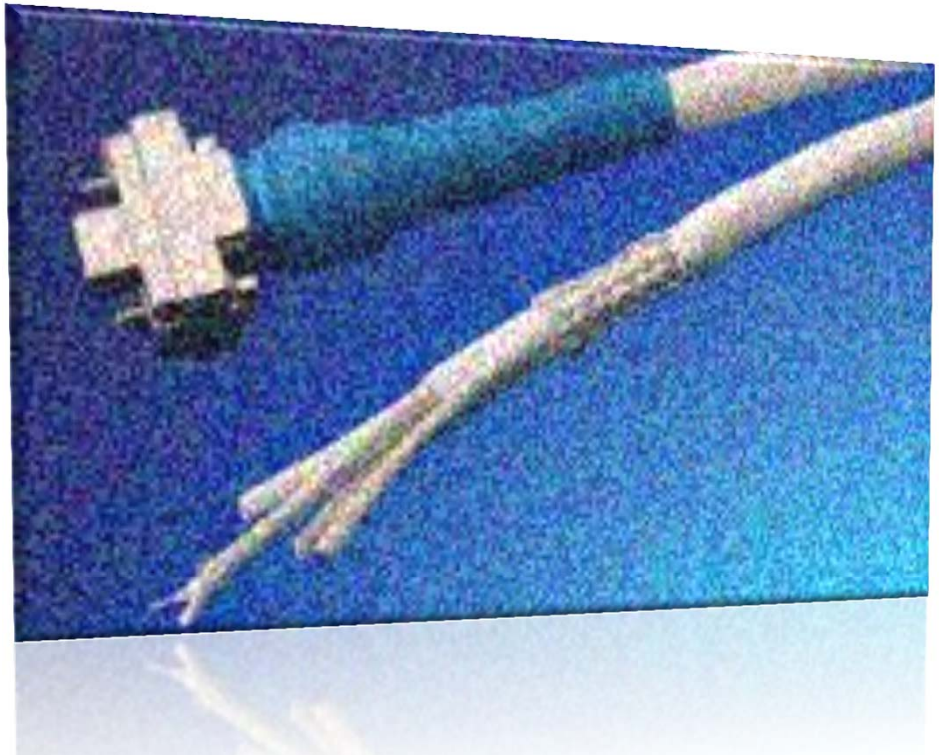


Findings on Half Duplex SpW and Interrupts Distribution through prototyping and validation

19th SpaceWire Working Group, 3rd October 2012
Paris, France

Project Overview

Current Status:

- Networking technology for building on-board communications in S/C, used for the interconnection of:
  - Mass-memory
  - OBC
  - Telemetry
  - …
- Designed by ESA and widely used on many ESA, NASA, JAXA, RKA space missions
- The standard specifies point-to-point full duplex links, with flow control mechanism which ensures that no data is lost due to receiver buffer overruns

The Problems:

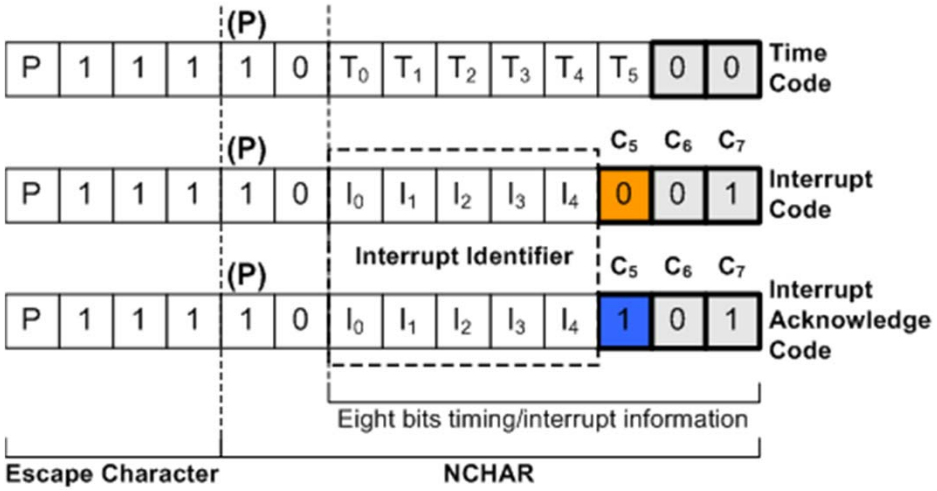
- Current SpW standard does not provide a mechanism for propagation of critical events (e.g. alarms) – **Interrupts distribution** mechanisms proposed by SUAI and refined by SUAI, ESA, Aeroflex-Gaisler, TELETEL were analyzed, implemented by 4Links and TELETEL and validated in the project
- Bidirectional flow of data is not always required (e.g. sensors, actuators) – **Half Duplex SpW** version was analyzed and studied in the project

SpaceWire Interrupts Distribution

SUAI Proposal: Time-Codes & Interrupts

The Time Code Characters:

- Time Code is a minimal latency character broadcasted throughout the entire SpW network
- Six bits of time information are held in the least significant six bits of the Time-Code (T0-T5)
- Two bits (T6, T7), assigned to “00”, contain control flags Time-Code
- The rest three T6, T7 combinations are reserved for future use



The SUAI Proposal:

- Define **Interrupt Codes (INTR)**, which is a signal representing a request to handle an event of high priority
- Define **Interrupt Acknowledge Codes (INTA)** which acknowledge Interrupt Code acceptance for processing by a handler
- Use the time-codes propagation mechanism to distribute interrupts/acknowledgements
- This ensures minimal propagation latency and propagation through blocked links
- Use one of the reserved T6, T7 combinations to define interrupts/acknowledgements
- Use on bit (C_5) to distinguish between Interrupt Code and Interrupt Acknowledge Code
- Use a five bits **interrupt identifier** (I_0-I_4) to define 32 Interrupt Codes and 32 Interrupt Acknowledgement Codes

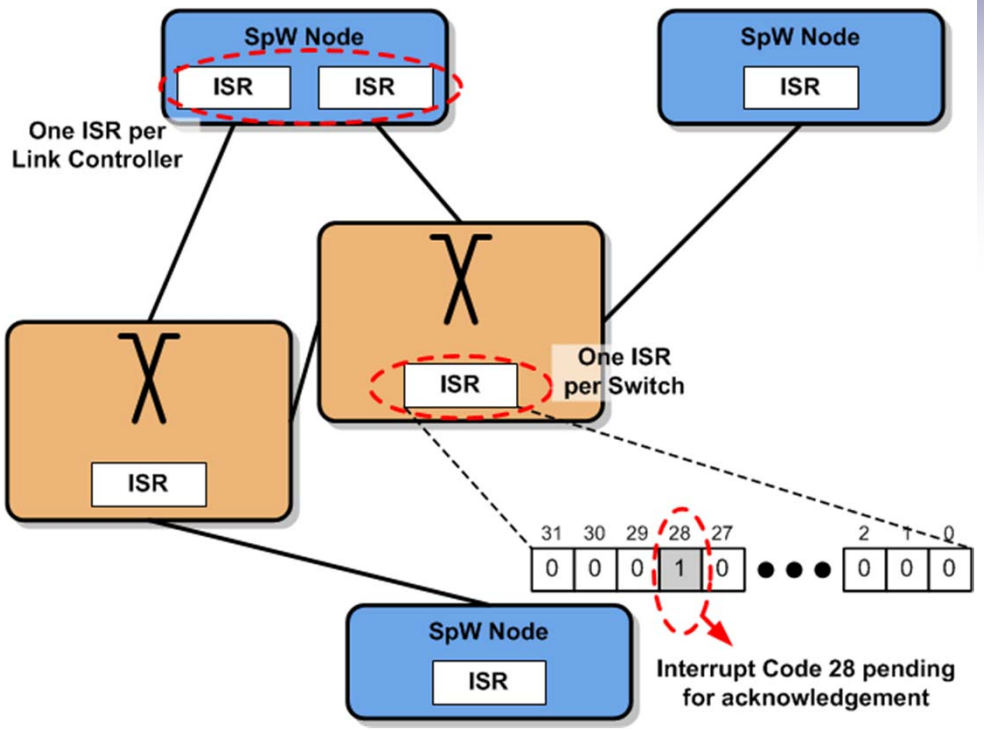
SUAI Proposal: Interrupt Status Register

The Interrupt Status Register

- Each Link Controller of a Node and each Switch contains one 32-bit Interrupt Status Register (ISR)
- Each ISR bit corresponds to one of 32 possible interrupt identifiers
- An ISR bit is set to ‘1’ upon the transmission or reception of an Interrupt Code with the corresponding Interrupt Source Identifier
- An ISR bit is cleared to ‘0’ upon the transmission or reception of an Interrupt Acknowledge Code with the corresponding Interrupt Source Identifier

Interrupt Status Register Functionality

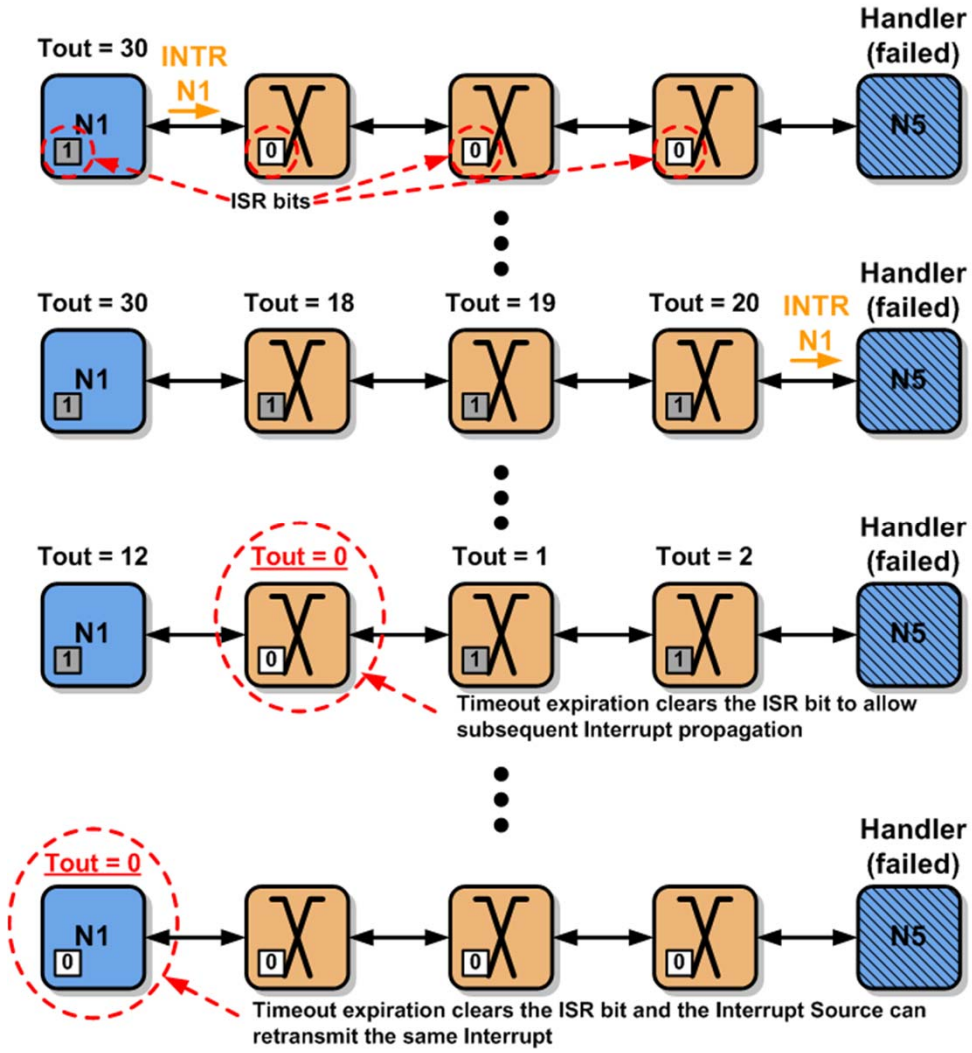
- Stores information about Interrupt Codes pending for Acknowledgement
- Initiates a time-out for each Interrupt Code broadcasted in the network
- Prevents repeated transmission of the same Interrupt Code or same Interrupt Acknowledgement in circular networks



SUAI Proposal: ISR and Timeouts

ISR Timeouts:

- An ISR bit set to ‘1’ in a Switch does not allow a received Interrupt to be broadcasted
- An ISR bit set to ‘1’ in an Interrupt Handler Node does not allow the respective Interrupt Code to be transmitted
- After the transmission of an Interrupt the respective ISR bits in Switches and Nodes will never be cleared if:
  - The INTR does not reach the Interrupt Handler
  - The Interrupt Handler does not respond
  - The INTA gets lost
- For this reason the SUAI proposal specifies:
  - A timeout is started upon transmission/reception of an Interrupt Code at the Nodes
  - A timeout is started upon reception of an Interrupt Code in Switches
  - Expiration of a timeout clears the respective ISR bit



ummary of updates and comments on the SUAI proposal (SpW WG #18)

- Acknowledgements are not necessary – Acknowledgement can be performed at system level **=> Interrupt mechanism without acknowledgements** is also specified in which timeout is the only means to clear the ISR and ISR timeouts
- Interrupt mechanism without acknowledgements can support up to 64 interrupts but the two mechanisms cannot coexist or configuration is required upon start-up in order to “inform” the devices how they shall treat Identifiers greater than 31 **=>** Both mechanisms support up to 32 Interrupt Identifiers
- Certain legacy devices cannot discriminate between Time-Codes and Interrupt Codes **=>** Switches shall be configurable not to send INTR/INTA over specific ports
- In multi-hop paths between a Source to a Handler calculation of the worst case propagation time shall not be based on the assumption that a single interrupt exists in the network **=>** Simultaneous arrival of INTR/INTA in the intermediate switches shall be taken into account for the calculation of the maximum INTR frequency
- In networks with redundant paths a transient link failure on one of the paths can cause switches to block INTR distribution for a successive number of transmission of the same identifier **=>** The appearance of transient failures shall be taken into account and the maximum transmission frequency shall be calculated in order to allow ISR timeouts in such cases
- The existence of a manager node clearing the non-acknowledged interrupts may cause loss of retransmitted INTR **=>** Time-out calculation at the Sources shall take into account Manager Node timeout plus the propagation time for the arrival of INTA from the manager node
- In circular networks transmission of INTA while the INTR coming through a second path has not reached the handler can cause problems **=>** Specify a minimum time for the Interrupt Handler response related to the network diameter (SUAI) and specify a minimum time for transmission of a INTR after the reception of INTA with the same identifier
- “Reset on disconnect” requires implementation of timeouts in the host SW **=>** ISR and timeouts are not affected by link disconnect
- The host is not aware of whether the INTR/INTA has been transmitted **=>** If the link is not in the RUN state upon host’s request for Interrupt Code transmission, the Link Controller shall notify the host and ignore the request
- Malfunctioning nodes may transmit INTR or respond to INTA **=>** Protection may be provided by **edge** switches in order to block INTR/INTA transmitted by babbling idiots

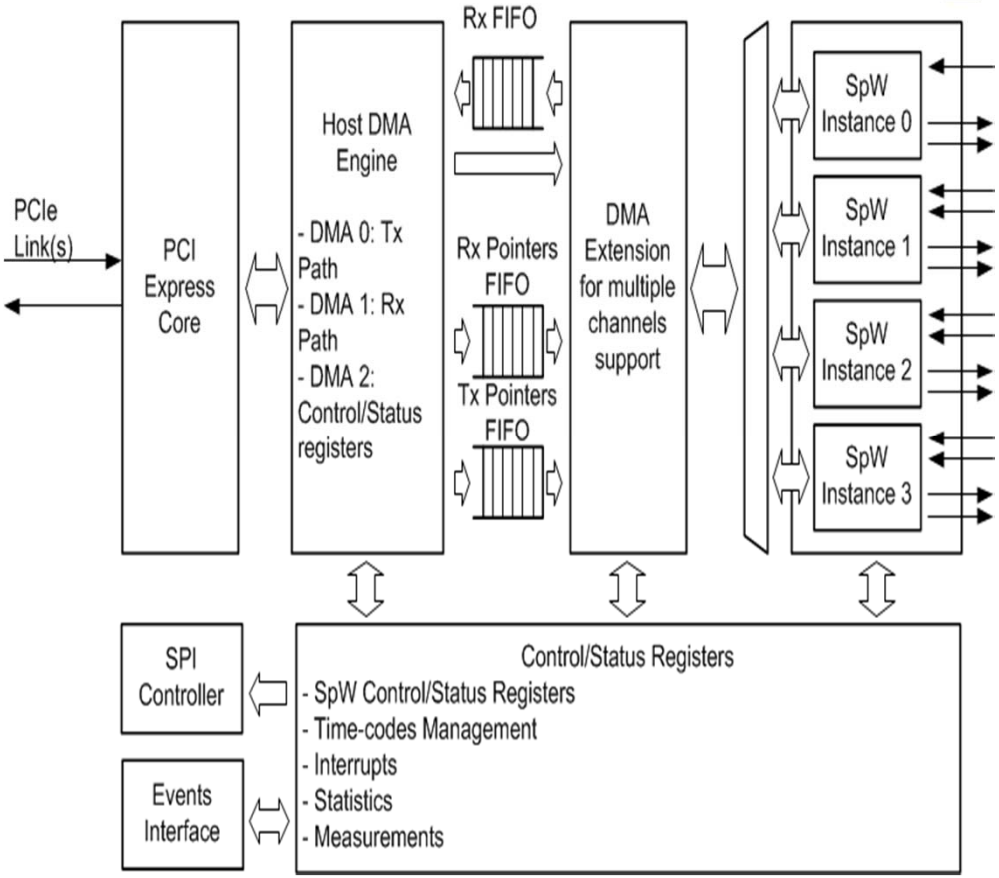
Implementation and validation (1/2)

HW:

- Interrupts block implemented in the PVS SpW FPGA
- PCIe host interface
- DMA capability for full line rate traffic injection
- Four SpW 1.1 ports implementing Interrupts distribution with and without acknowledgements
- Traffic generation/packet sinking for performance measurements – injection of INTR/INTA with traffic
- Inter-card communication interface for events distribution/ performance measurements on different cards
- Add on board specifically designed and developed for the SpW Evolutions project

SW:

- Interrupt Validation test suite developed on TELETEL's iSAFT TestRunner environment
- 4Links switches configuration by the PVS RMAP CODEC



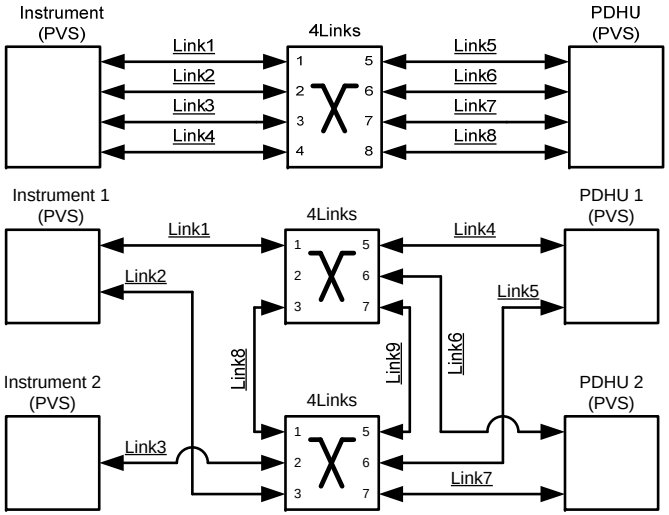
Implementation and validation (2/2)

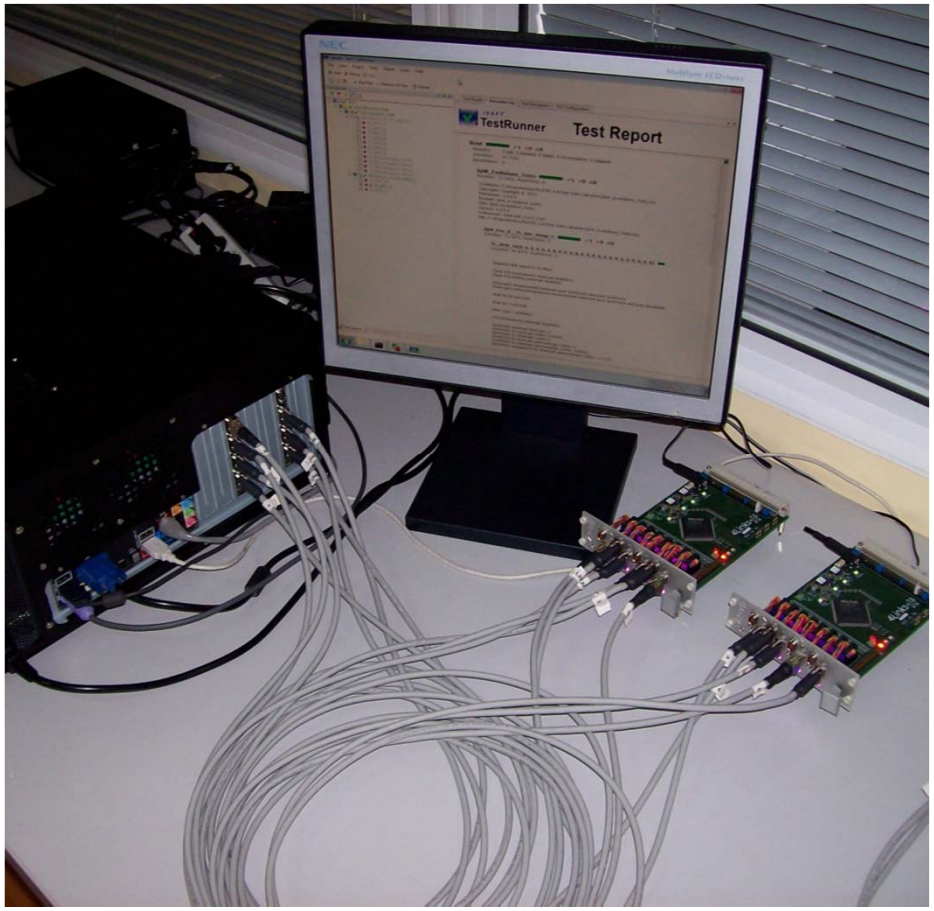
Three different set-ups:

- PVS point-to point: For the validation of the PVS functionality
- Single switch path: For the validation of the 4Links switches functionality and execution of Interrupt functional and performance tests
- Dual switch path: For validation tests involving paths with cascaded switches or redundant paths between a Source and a Handler

Equipment:

- Two PVS SpW cards were used
- Two 4Links SpW Evolutions switches were used





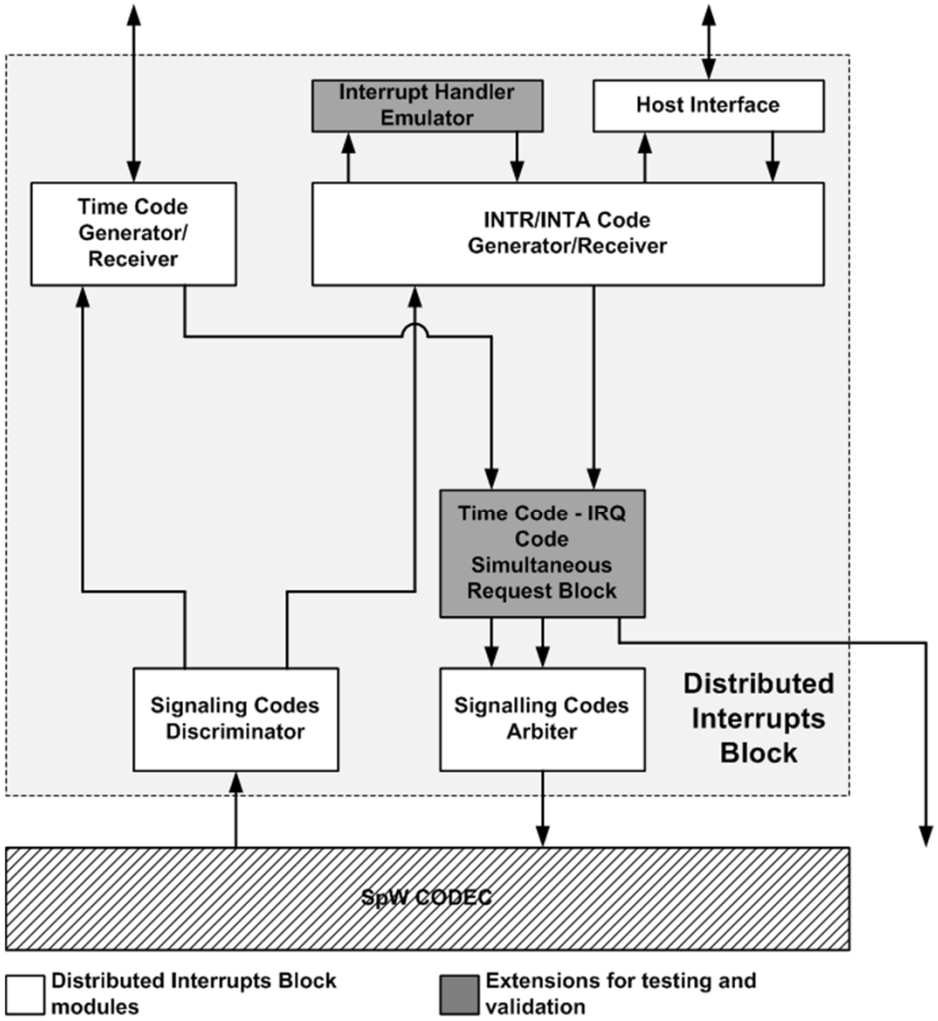
Implementation – SpW Node (1/3)

Interrupt Block modules:

- Signaling Codes arbiter: Block which resolves simultaneous requests for the transmission of Time-Codes, Interrupt-Codes, Interrupt-Acknowledge-Codes and ensures that they are transmitted with the correct priority
- Signaling codes discriminator: Discriminates among Time-Codes, Interrupt-Codes, Interrupt-Acknowledge-Codes passing them to the appropriate module
- INTR/INTA Transmitter receiver: Implements the Interrupts Distribution logic (ISR, ISR timeouts
- Host Interface: Interface to the host providing indication and control signals for transmission and reception of the Interrupt Codes & Acknowledgements

Extensions for testing and validation:

- Interrupt Handler Emulator: Block responding to programmable interrupts. Response time programmable in us
- Time-Code – INTR/INTA simultaneous transmission: Block which injects simultaneous Time-Codes and INTR/INTA or simultaneous INTR/INTA over two SpW ports for switch functional validation tests



Implementation – SpW Node (2/3)

Interrupt Transmitter:

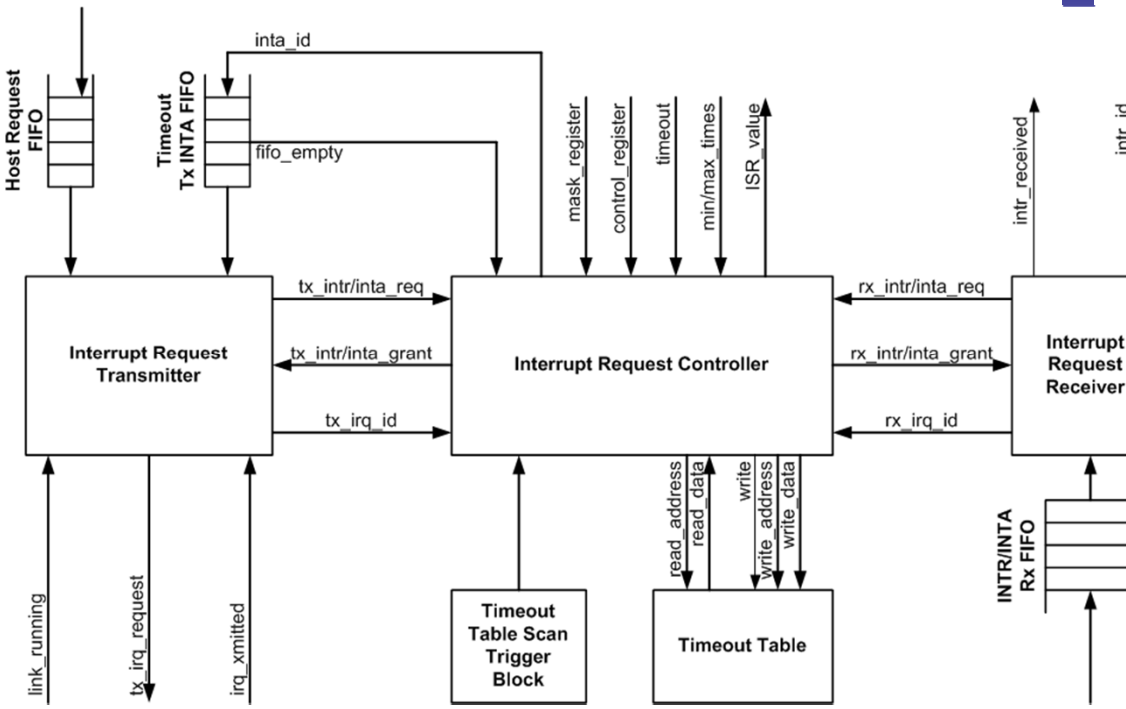
- Receives INTR/INTA Tx requests from Host
- Requests ISR search and update from the controller
- When Tx is granted, requests INTR/INTA transmission from Sig. Codes Arbiter if Link Is up

Interrupt Receiver:

- Receives INTR/INTA from discriminator
- Requests ISR search and update from the controller
- When request is granted it informs the host that a valid INTR has been received

Controller:

- Receives and resolves simultaneous requests from the Tx/Rx, performs ISR LUT and update and grants requests accordingly
- Receives “table scan” command from the Scan trigger block searches the Timeout table, invalidates ISR bits whose ISR timer has expired
- Requests INTA transmission in case the node is configures as Interrupt Source for a specific identifier and its timeout has expired



Implementation issues:

- Simultaneous requests may be issued to the controller by the Tx and Rx blocks for the same interrupt identifier and from the timeout trigger scan
- In Tx path from ISR check and update to actual transmission the link status may change. Transmitter locks the controller until INTR/INTA has been transmitted and all other requests are not served

Implementation – SpW Node (3/3)

Block	Module	Registers	LUTs	Memories
Interrupts Distribution Block				
Interrupts Transmitter/ Receiver	Transmitter	27	33	Host FIFO: 32x8 INTA FIFO: 32x8
	Receiver	19	34	32x8
	Controller	171	436	-
	Scan trigger	106	126	-
	Timeout Table	6	15	DPRAM: 32x35
Transmitter/ Receiver TOTAL		372	708	FIFOs: 3x32x8 DPRAMs: 32x35
Discriminator		0	2	-
Arbiter		31	29	-
Host I/F		89	225	-
Extensions for testing and validation				
Handler Emulator		105	129	-
Simultaneous Tx		2	4	-

Xilinx Virtex-5 LX implementation metrics

Validation – Tests

Category	Set up	PVS Tests	Switch tests
Functional tests	PVS Point to point	INTR/INTA generation, reset on disconnect, …	
		Validity of Statistics for valid/invalid events detection	
		Timeout Functionality	
	Single switch network		INTR/INTA distribution,
			Timeout functionality
		Propagation over blocked links	
		Periodic INTR transmission	
		Time-Code/INTR/INTA priority	
		Propagation over redundant paths	
	Dual switch network	Compliance with legacy devices	
Performance	Single switch network	INTR/INTA propagation latency	
		Maximum INTR generation frequency	
	Dual switch network	INTR/INTA propagation latency	
		Maximum INTR generation frequency	
		INTR/INTA propagation latency under heavy traffic load	
		Max INTR generation frequency comparative test for the two mechanisms	

Validation – results (1/4)

Set-up	Interrupt Identifier	Minimum Time (ns)	Maximum Time (ns)	MAX-MIN Difference in NULLs
Interrupt transmission to reception over two PVS ports – Link Speed = 50 Mbps				
Point to point	0	800	928	0,80
	16	768	896	0,80
	31	768	896	0,80
Single switch path	0	1312	2800	9,30
	16	1264	2800	9,60
	31	1280	2816	9,60
Dual switch path	0	1808	4592	17,40
	16	1840	4640	17,50
	31	1776	4640	17,90
Dual switch path with four links between the switches	0	1792	4512	17,00
	16	1824	4528	16,90
	31	1760	4544	17,40

Increases

Doubles

Decreases (slightly)

1) **Interrupt Processing Time in the switches independent from Interrupt Identifier**
2) **Latency is linearly proportional to the number of hops**
3) **Redundant paths between switches slightly decrease latency**

Validation – results (2/4)

Link Speed	Point to Point		Single switch path		Switch Interrupt Processing Time	
	Min (A)	Max (B)	Min (C)	Max (D)	Min = C – A – 16 bits (TC + 2 bits for parity)	Max = D – A – 32 bits (2 x NULLs + TC + 2 bits for parity
10 Mbps	3536	4144	5280	7936	144 ns	1200 ns
25 Mbps	1520	1760	2320	4064	160 ns	1264 ns
50 Mbps	800	928	1312	2800	192 ns	1360 ns
100 Mbps	432	496	816	2096	224 ns	1344 ns

Link Speed	Interrupt identifier	INTR Tx to INTA Rx on the same port (ns)	
		Min	Max
25 Mbps	**0**	3744	3984
	16	4096	4336
	31	4592	4976
50 Mbps	**0**	2128	2592
	16	2432	3504
	31	2608	3520
100 Mbps	**0**	1728	1936
	16	1904	2304
	31	1840	1968

Interrupt Processing Time at the Interrupt Handler depends on the Interrupt Identifier (implementation dependent)

Validation – results (3/4)

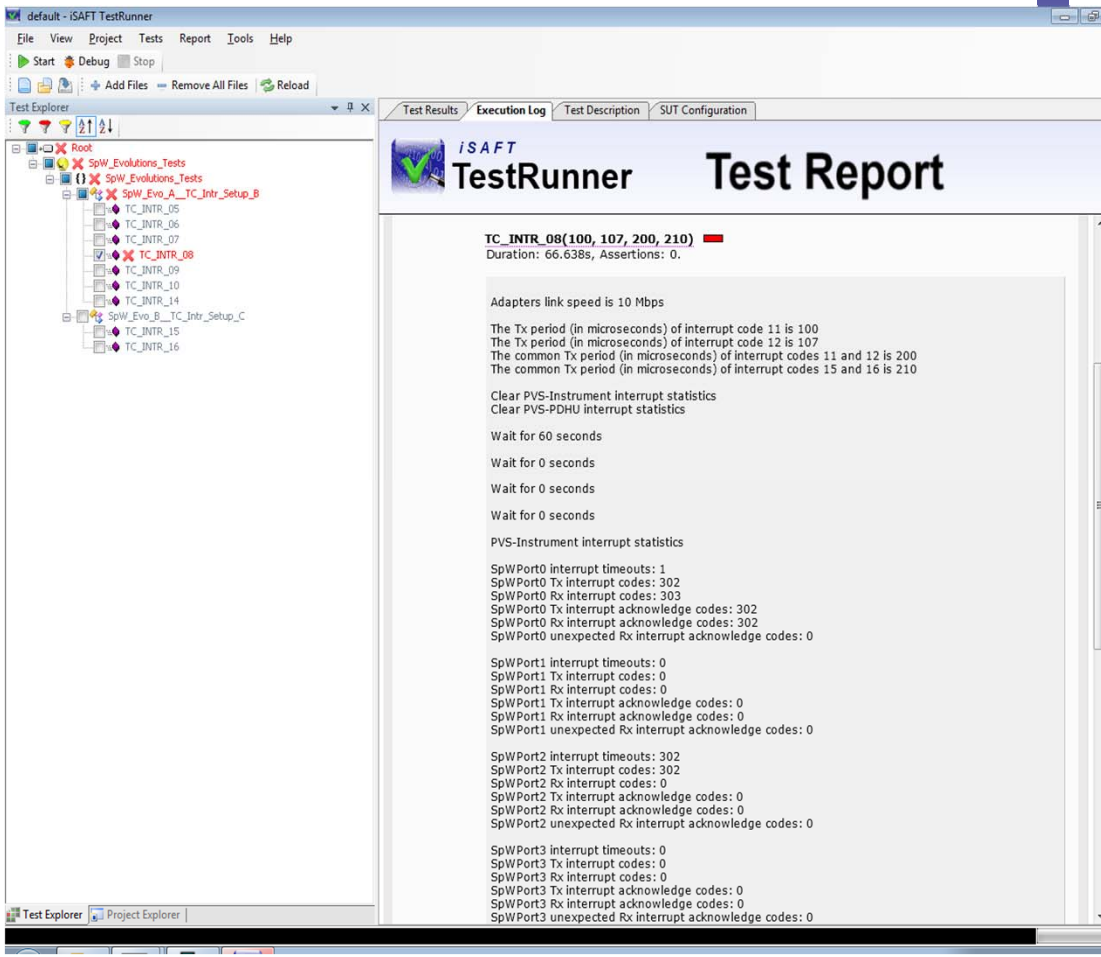
Link Speed	Topology	INTR0 Tx to INTA0 Rx on the same port (ns)		INTR0 Tx to INTR0 Rx on different port (ns)	
		Min	Max	Min	Max
25 Mbps	**Point to point**	3440	4288	1520	1760
	Single switch	6080	7888	2320	4064
	Dual switch	7088	12960	3120	6352
	Dual switch – redundant links	7056	12176	3120	6192
50 Mbps	**Point to point**	2160	2640	800	928
	Single switch	3856	5296	1312	2800
	Dual switch	4480	9488	1808	4592
	Dual switch – redundant links	4318	9120	1792	4512
100 Mbps	**Point to point**	1376	1456	432	496
	Single switch	2048	4688	816	2096
	Dual switch	3072	8000	1152	3760
	Dual switch – redundant links	3072	7936	1120	3744

Interrupts without Acknowledgements can operate at more than double frequency

NOTE: Interrupts without Acknowledgement measurements do not include Handler emulator processing time

Validation – results (4/4)

- All validation Test Cases except for SpW Interrupts Test Case 08, which involved the transmission of simultaneous INTR by two PVS ports, were successful
- Test Case 08 was successful up to 12,5 Mbps link speed
- For Link speeds 25Mbps or greater the result was non-deterministic or was failing
- Handshaking delays in the Time-Codes path was found to be the cause
- The time-Code path has been designed for Time-Codes with the assumption that the Time-Codes source is a single device in the SpW network and **arrival of back-to-back Time-Codes never occurs** – This cannot be ensured in the case of Interrupts



Back to back Interrupt Codes require the addition of a FIFO in the SpW CODEC Time-Codes path

Conclusions

- Low latency mechanism for propagation of significant events
- Interrupts without Acknowledgements can be transmitted at double maximum frequency than Interrupts with Acknowledgements
- Logic required at the nodes/switches is built outside the SpW CODEC using the Time-Codes interface
- Modifications in the Time-Codes path of SpW CODECs is required in order to support simultaneous existence of more than one Interrupts, or an Interrupt and a Time-Code
- Existing switches are not compatible with Interrupts since they interpret them as Time-Codes
- Depending on the implementation, the processing time at the switches and nodes cannot be calculated with us accuracy due to the existence of multiple requestors accessing the ISR timeout table
- Depending on the implementation, the response time for the transmission of a INTA may have a dependency on the Interrupt Identifier number

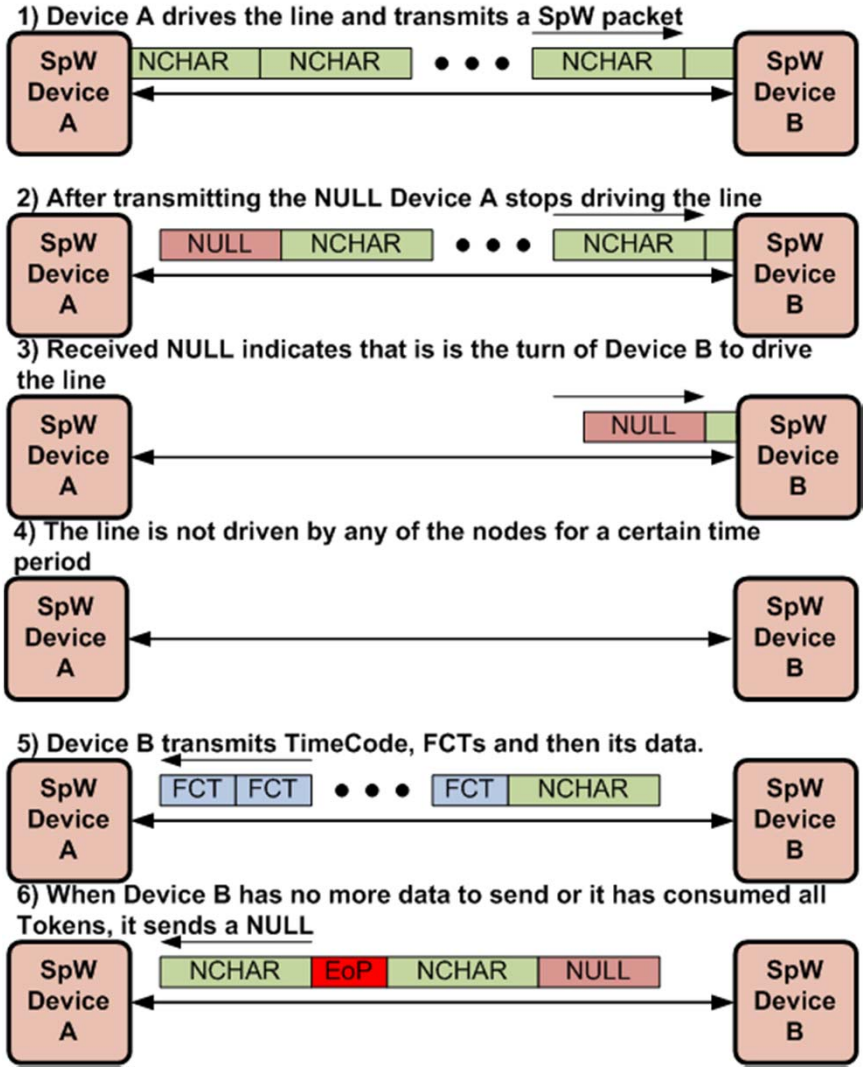
Half Duplex SpaceWire

Half Duplex SpW

Half Duplex SpW:

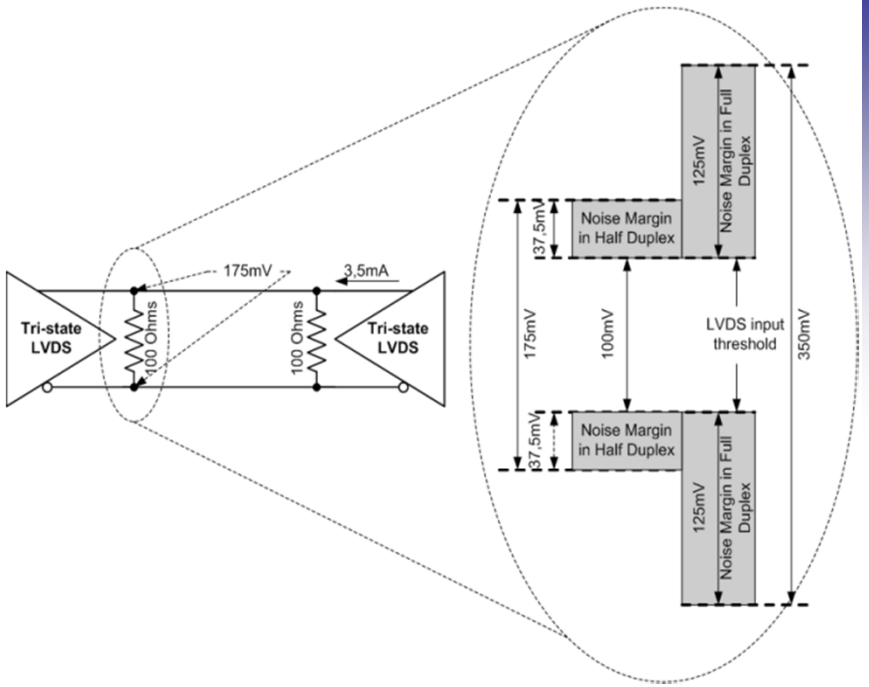
- One end transmits its data until its buffer is empty, or
- It has consumed all the FCTs it has received
- It then sends a NULL indicating its has ceased transmission
- Upon reception of a NULL the other end,
- Sends a NULL if there is nothing to send, or
- Transmits Time Codes, then
- FCTs, then
- NCHARs, EoP/EEP
- And finally a NULL to enable the other end to resume transmission

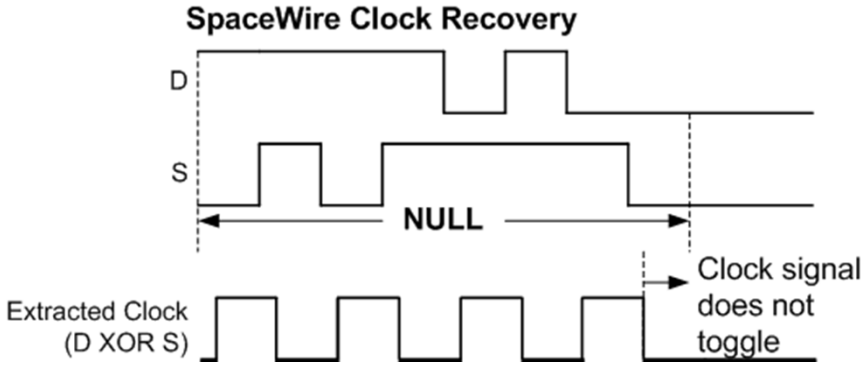
⇨ **Half Duplex SpW (initially seemed to) offer all Full Duplex SpW features**



Half Duplex SpW – Technical Issues

- The main challenges with Half Duplex SpW are the Link Initialization, the link direction reversal and the Signal and Physical Levels (Rx pairs are the same as Tx pairs)
- Link Initialization:
  - Full Duplex state machine ensures that the two ends pass through the same states concurrently
  - Not possible with Half Duplex since they will be both listening or driving the line at the same time
- Direction reversal:
  - SpW Receivers extract the remote end transmission clock by XORing the D and S signals
  - After the last NULL is received no mode clock pulses are generated and the logic generating the “NULL_received” cannot be generated (logic remains unclocked)
- Half Duplex Signal Level:
  - LVDS is point-to point and unidirectional
  - At some point in time both ends may be driving the line
  - Termination at both ends exceeds of the link causes the voltage at the termination resistors to be very close to the LVDS threshold
- Connector/cabling definition
  - One D-S pair wiring is not used





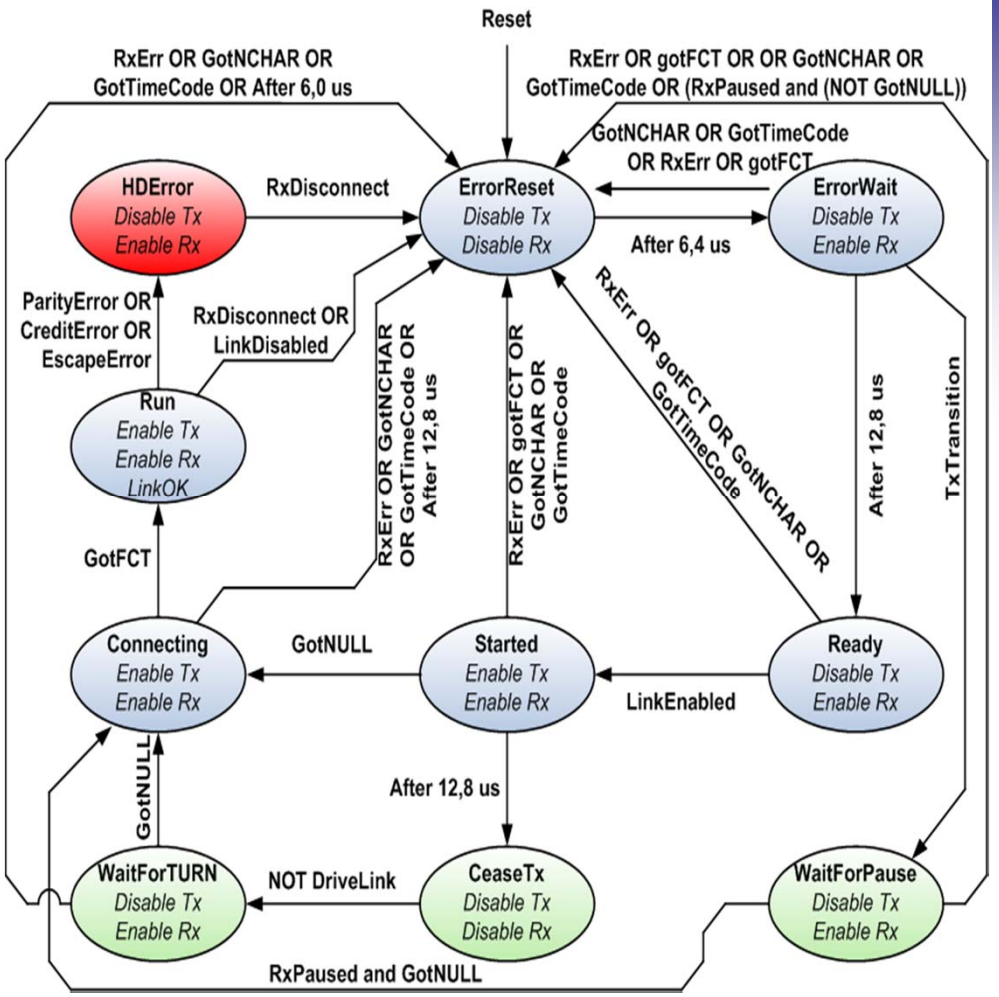
Technical Issues – Link Initialization

New signals:

- TxTransition: Indicates that a transition has been detected on the D-S pair
- RxPaused: Indicates that no transition has been detected on the D-S pair for 200 ns
- DriveLink: Indicates that the link is driven by the controller's transmitter. Deasserted after the last character has been transmitted and the transmitter does not drive the link

New states:

- ***WaitForPause***: Entered from the ***ErrorWait*** state if a transition has been detected on the transmit D-S pair. The state machine waits here until the remote end does not drive the line (rx_paused)
- ***CeaseTx***: Entered from ***Started*** state if no NULL has been received for 12,8 us a condition which may indicate that the remote end is a full duplex and is disabled, or it is half duplex and waits its turn for transmission
- ***WaitForTURN***: Entered from the ***CeaseTx*** state. At this state the transmitter is disabled to allow the remote end to transmit its NULLs and FCTs
- ***HDError***: Entered when the receiving side detects a SpW Error. Ensures that the "Exchange of Silence" mechanism is followed upon SpW error occurrence



Supports Full/Half Duplex auto detection

Need to insert asymmetry to avoid state machine live-lock

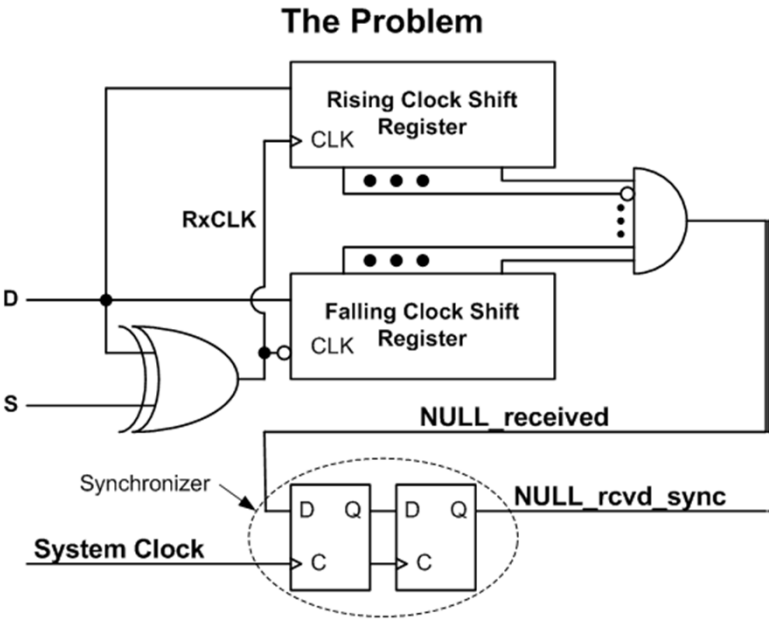
Technical Issues – Link Direction Reversal

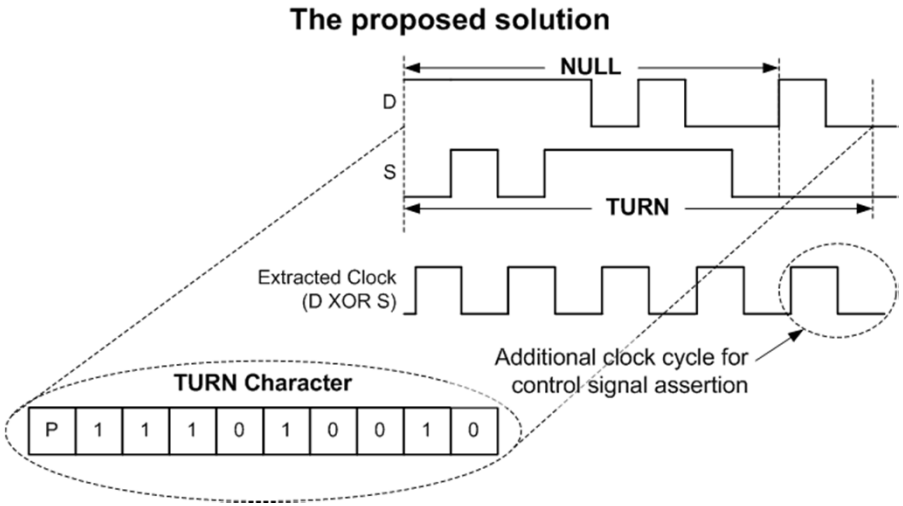
The Problem:

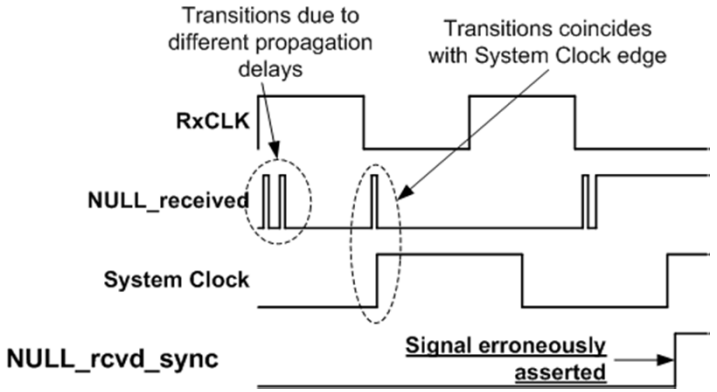
- The receiver's system clock and the reception clock are asynchronous
- The NULL_received signal will be generated by combinational logic
- The NULL decoder output has transitions due to changes of the logic levels at its inputs and differences in propagation delays
- These transitions may be patched by the system clock and erroneously cause link direction reversal

The initially proposed solution:

- Provide one more clock cycle to the receiver for decoder output latching by extending the NULL with a parity bit and one more zero (TURN character)







Technical Issues – Link Initialization & Direction Reversal

- The receiver requires at least one more clock cycle in order to decode the TURN character – parity is contained in bits following the “NULL” field

Alternative solution:

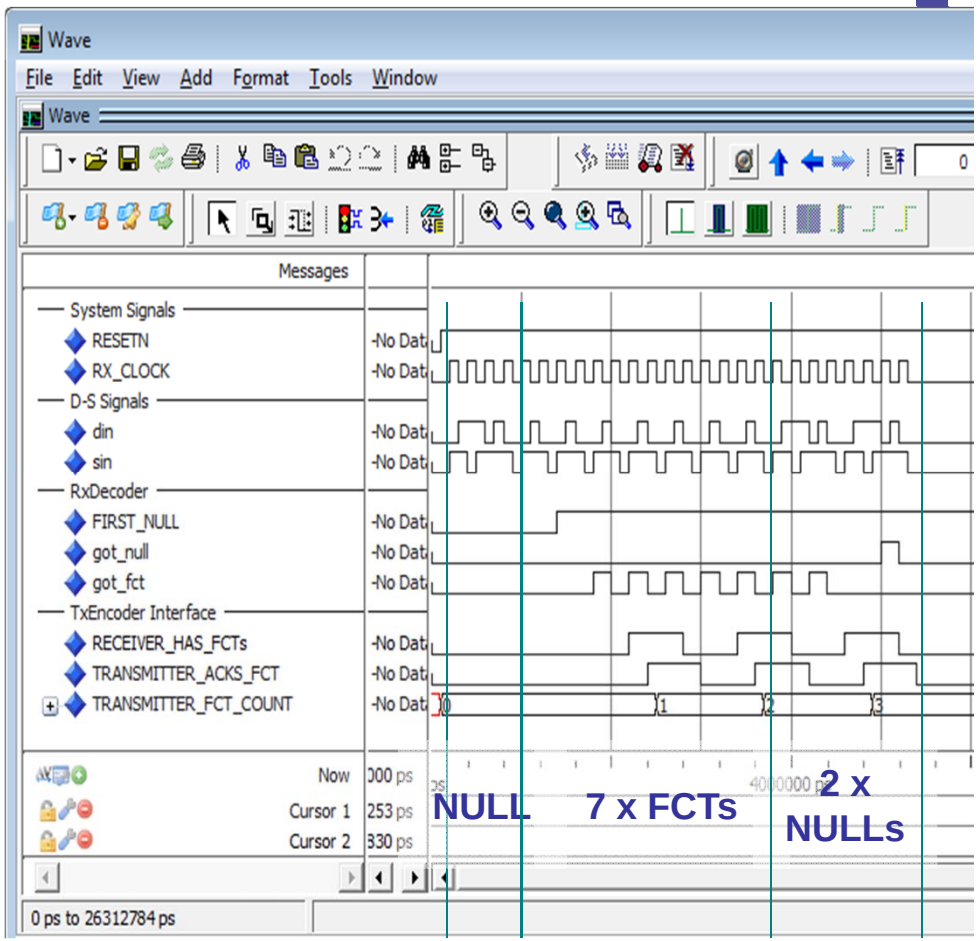
- Two NULLs were used to signal the Link Direction Reversal

but,

- The receiver cannot validate that the second character is a NULL indeed
- The parity of the second character is carried by a subsequent character which is received upon the next Link Direction Reversal
- **So parity errors on the second character go undetected!**

In addition,

- Receiver and transmitter are clocked by different sources
- Handshaking between the receiver and transmitter is required for passing the decoded FCTs
- Several clock cycles are required for handshaking
- This takes several cycles and,
- **Simulations have shown that the two NULLs are not adequate to decode a flow of continuous FCTs – performance degradation!**



Half Duplex SpW in mainstream implementations (D XOR S) cannot meet the performance requirements for the use case for which it was initially proposed

Technical Issues – Signal Level

Candidate Technologies			
Link Speed	**LVDS (-A) (TIA/EIA 644 (-A))**	**M-LVDS (TIA/EIA 899)**	**BLVDS (not standardized)**
Offset Voltage	**1,125 – 1,375 V**	**0,3 – 2,1 V**	**1,185 – 1,435 V**
Vout	**454 mV (100 Ohms)**	**565 mV (50 Ohms)**	**350 (50 Ohms)**
Transition time	**260 ps**	**1000 ps**	**350 – 1000 ps**
Driver strength	**3,5 mA**	**11,3 mA**	**7 – 11,1 mA**
Ground potential difference	**±1 V**	**±2 V**	**±1 V**
Input Voltage Range	**0 – 2,4 V**	**-1,4 – 3,8 V**	**0 – 2,4 V**
Input threshold	**±100 mV**	**±50 mV**	**±100 mV**
Max data rate (theoretical)	**1,923 Gbps**	**500 Mbps**	**800 Mbps**
Drivers contention	**Not supported**	**Output current control**	**Output current control**
Space Qualified Devices	**Exist**	**Aeroflex UT54LVDM055LV ?**	**Aeroflex UT54LVDM031LV**
Output voltage on 100 Ohms load	**350 mV**	**1130 mV**	**700 mV – 1110 mV**
Compatibility with LVDS	**Yes**	**Analysis per design is required. Current at LVDS termination resistor may cause a voltage of > 1 Vpp**	

LVDS not suitable for multi-point topologies. B/MLVDS inject more current

MLVDS has different Rx threshold – EMC/EMI characterization required

SpW WG 18 conclusions - UPDATED

☺ **Advantages:**

- Supports all SpW 1.0 functionality and does not infer the hazard of NCHAR loss as Simplex SpW
- Wormhole routing supported
- Fair bandwidth allocation between the two ends of the link – **low traffic sources "block" traffic having bulk traffic to send**
- ~~Requires simple functional changes in the SpW Cores logic since the functionality is almost identical~~
- Simpler and lighter cabling required – lighter to be confirmed after EMC characterization
- Lower cost solution for networks with few hops without inferring large jitter/latencies
- Proposed state machine allows for auto-detection of Full/Half Duplex

☹ **Drawbacks:**

- BLVDS and M-LVDS inject 50% more current than LVDS and therefore cable definition shall be re-examined for EMC issues – cannot yet evaluate throughput vs. mass performance
- ~~Character Level modification for the new "TURN" character is required~~
- An asymmetry in the two ends shall be introduced (e.g. RNG, different timers for switches/nodes) to avoid live-lock
- Latency and Jitter is introduced in Time-Code propagation and application packets – not suitable as backbone network in scheduled networks with complex topologies
- Cannot support precise time-distribution
- May present excessive jitter in hot redundant topologies
- Efficiency and Latency are factors driving to opposite directions. Trade-off analysis per application is required
- **Requires extensive modifications on the SpW CODECs which implement D-S clock extraction otherwise maximum throughput is severely decreased => Not suitable for the use case for which it was initially proposed**