SpaceWire ECSS-E50-12A International SpaceWire Seminar 2003

Steve Parkes – University of Dundee
Josep Rosello – European Space Agency

Contents

- Introduction
- Physical Level
- Signal Level
- Character Level
- Exchange Level
- Packet Level

Introduction

- SpaceWire a standard for spacecraft onboard data-handling
  - ESA ECSS-E-50-12A
  - ISO ?
  - IEEE 1355.2 ?
- SpaceWire based on two commercial standards
  - IEEE-1355
  - LVDS (Low Voltage Differential Signalling)
- SpaceWire is the result of work by many people in ESA, NASA, the space industry and academia.

Introduction

- SpaceWire provides a unified high-speed data-handling infrastructure for connecting together

    - sensors,

    - processing elements,

    - mass-memory units,

    - downlink telemetry sub-systems, and

    - EGSE equipment.

Purpose

- The purpose of the SpaceWire standard is
  - to facilitate the construction of high-performance onboard data-handling systems
  - to help reduce system integration costs
  - to promote compatibility between data-handling equipment and sub-systems
  - to encourage re-use of data handling equipment across several different missions.

Compatibility and Reuse

- reduces development cost

- improves reliability

- eases integration and test

- increases "Science per Euro"

Scope

- **SpaceWire links are**
  - full-duplex, point-to-point, serial data communication links.

- **The scope of the SpaceWire standard is the**
  - physical connectors and cables,
  - electrical properties, and
  - logical protocols.

Scope

- The SpaceWire standard covers the following normative protocol levels

  - **Physical Level:** Connectors, cables, cable assemblies and PCB tracks

  - **Signal Level:** Signal encoding, voltage levels, noise margins, and data rates.

  - **Character Level:** Data and control characters used to manage the flow of data across a link.

  - **Exchange Level:** Protocol for link initialisation, flow control, link error detection and link error recovery.

  - **Packet Level:** How data is split up into packets for transfer across a SpaceWire link.

  - **Network Level:** Structure of a SpaceWire network, transfer of packets across a network, handling of errors.

SpaceWire Key Features

- High data rate (> 100 Mbps)
- Distance of 10m +
- Scalable
- Low error rate
- Good EMC performance
- Low-power
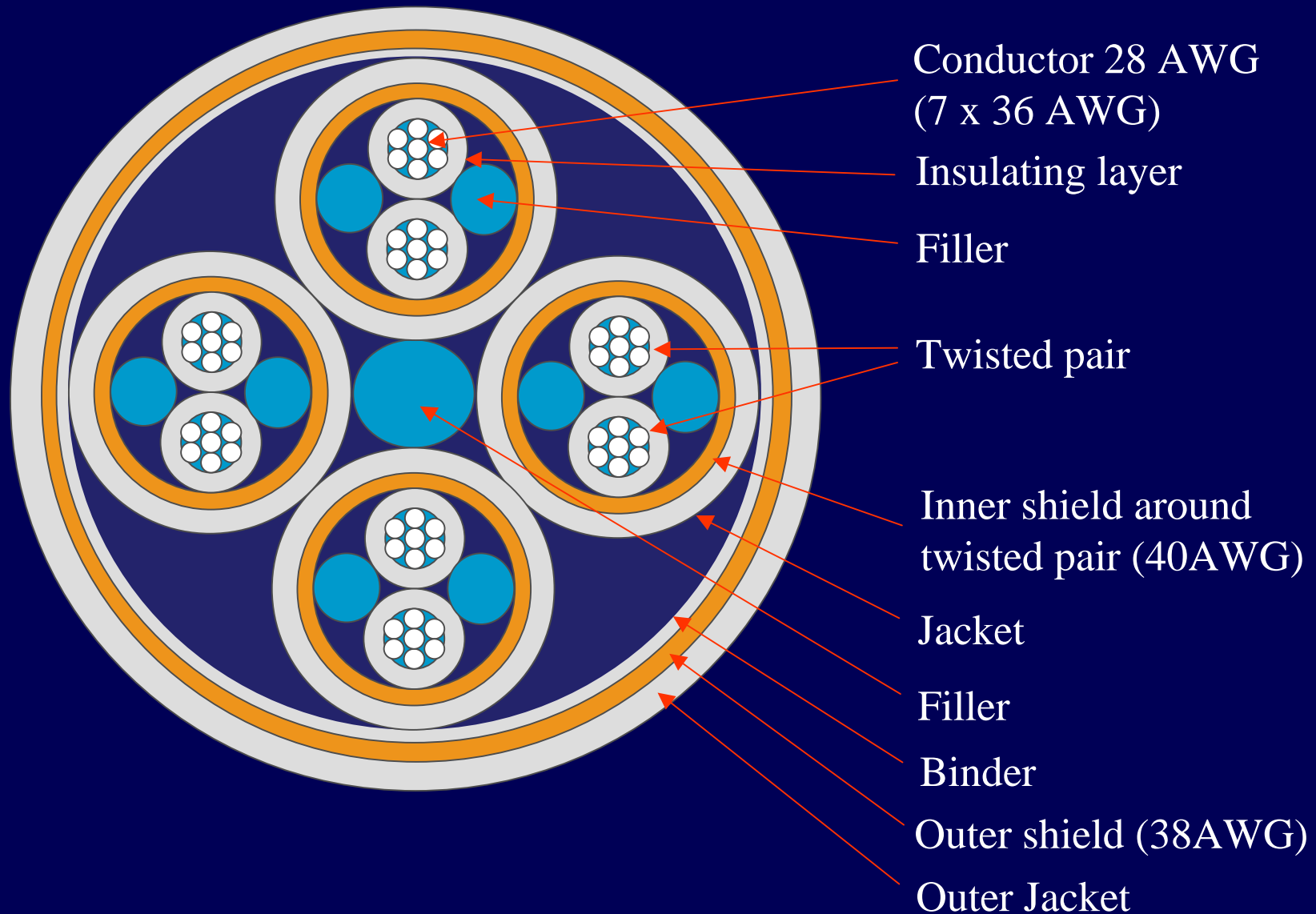- Fault tolerance support
- Radiation tolerant components

Physical Level

- Cables
- Connectors
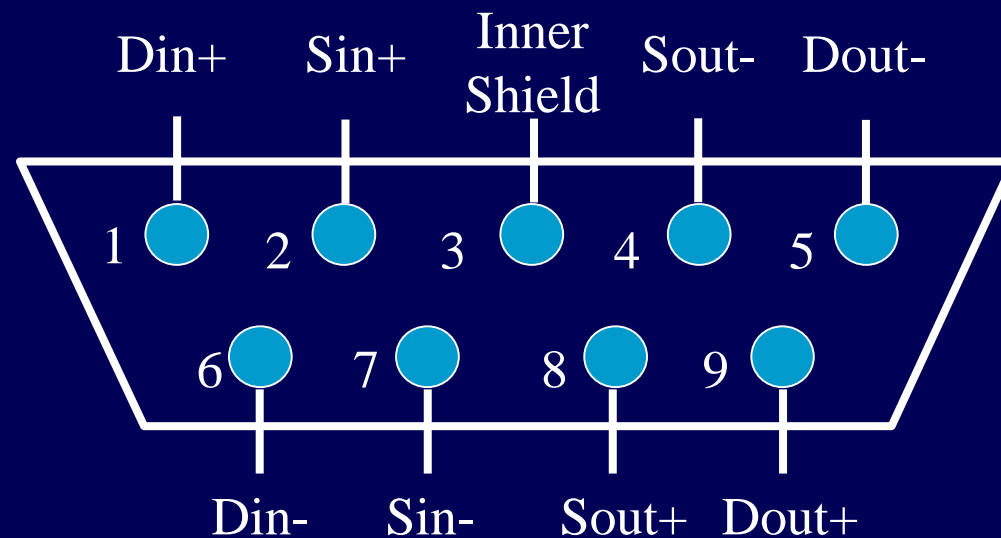- Cable Assemblies
- PCB traces

Cable Requirements

- Want high data-rate over distances of 10m plus
- Cable must have following characteristics
  - Characteristic impedance matched to line termination, 100 ohms
  - Low signal-signal skew
  - Low signal attenuation
  - Low cross-talk
  - Good EMC performance

Cable Construction

Conductor 28 AWG
(7 x 36 AWG)

Insulating layer

Filler

Twisted pair

Inner shield around
twisted pair (40AWG)

Jacket
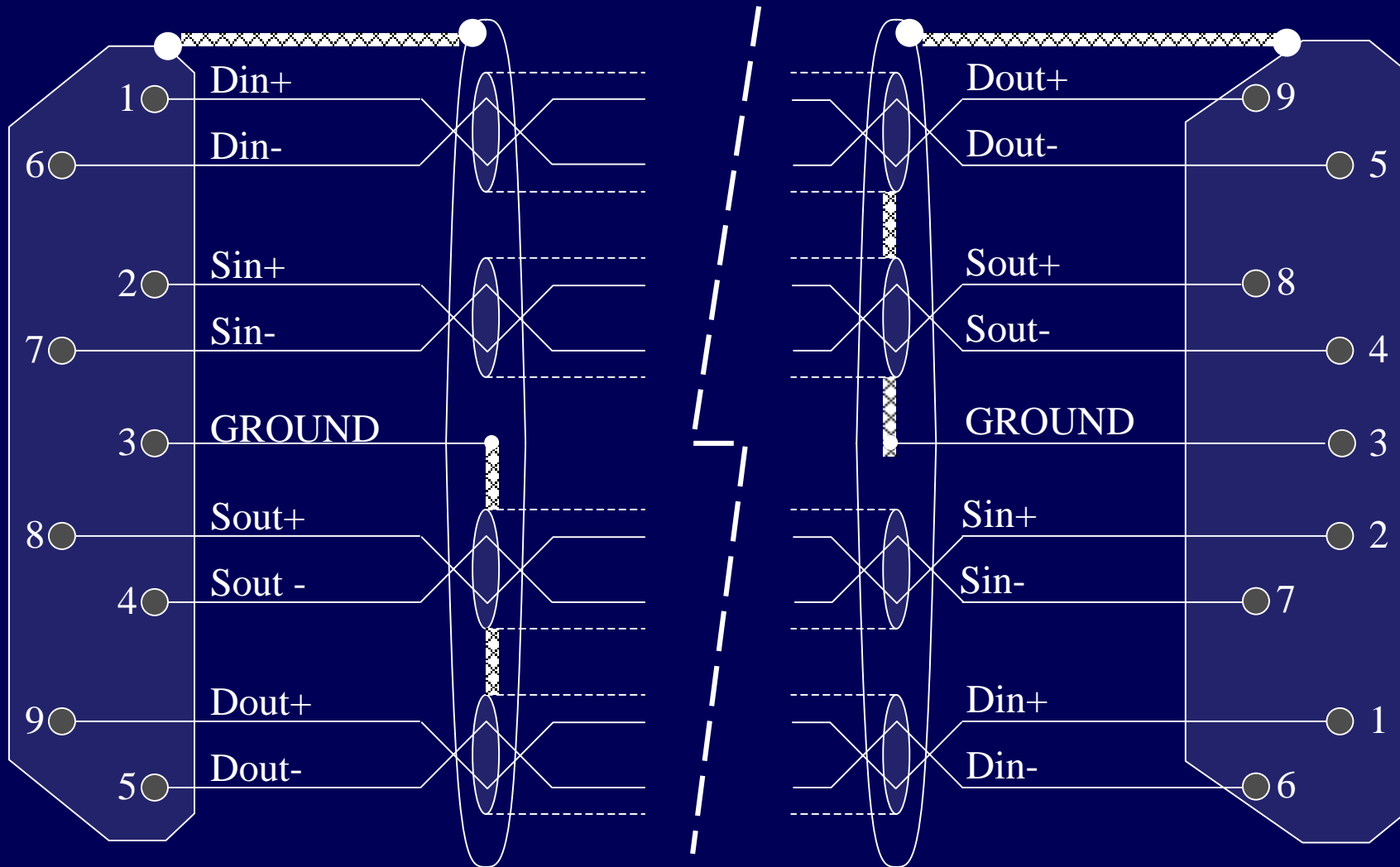
Filler

Binder

Outer shield (38AWG)

Outer Jacket

Connectors

- 8 signal wires
- Inner screen termination contact
- 9-pin micro-miniature D-type
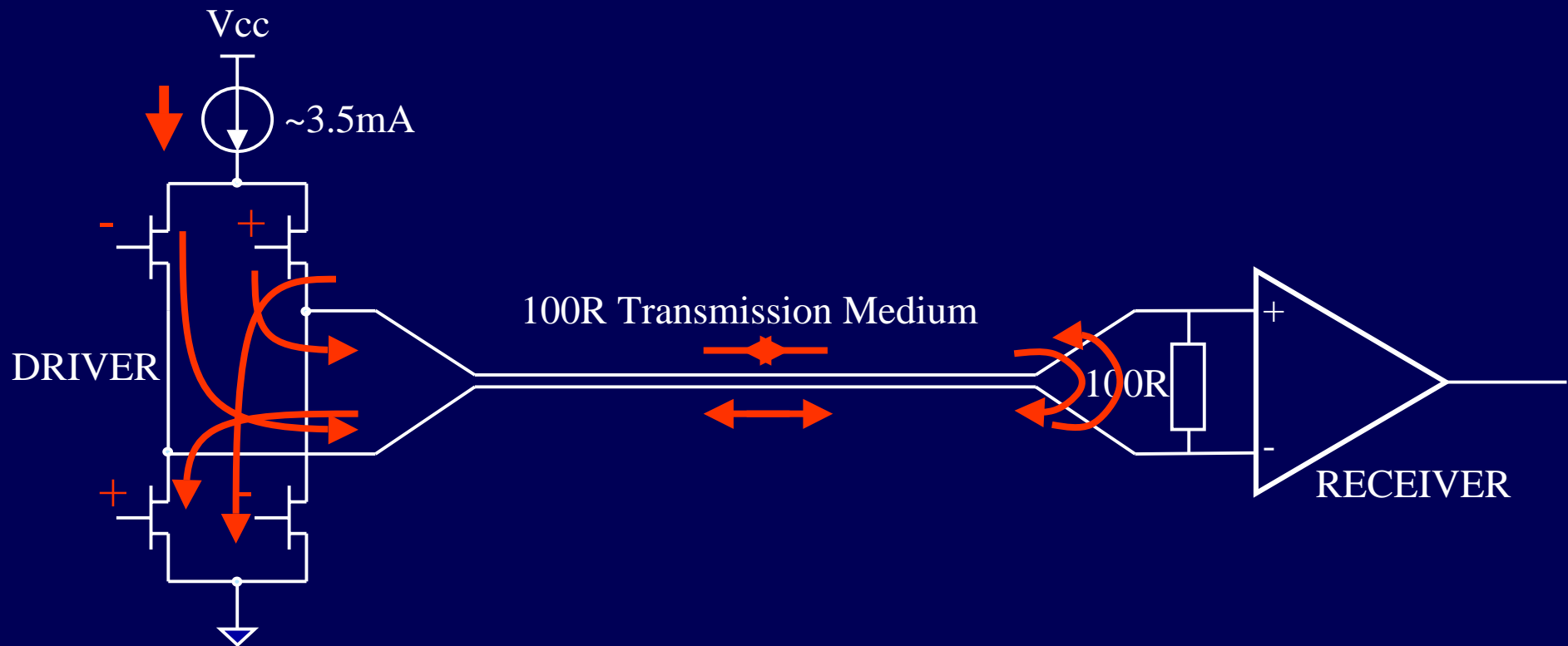
Cable Assembly

Cable Assembly

PCB Traces

- SpaceWire can also be run over PCB traces
- 100 ohm differential impedance must be maintained

Signal Level

- Low Voltage Differential Signalling (LVDS)
- Signal Encoding - Data/Strobe Encoding

LVDS Driver and Receiver

LVDS Features (1)

- Near constant total drive current
  - 3.5mA for logic 1 and 3.5mA for logic 0
  - decreases switching noise on power supplies.
- Good immunity to ground potential difference
  - between driver and receiver
  - can tolerate at least ±1V ground difference.
- High immunity to induced noise
  - because of differential signalling over twisted-pair cable.
- Low EM emission
  - because small equal and opposite currents create small electromagnetic fields which tend to cancel one another out.
- Not dependent upon particular device supply voltage(s).

LVDS Features (2)

- Simple 100 ohm termination at receiver.
- Failsafe operation -
  - the receiver output goes to the high state (inactive) whenever
    - the receiver is powered and the driver is not powered,
    - the inputs short together,
    - input wires are disconnected.
  - the transmitter output is current limited
    - preventing damage when transmitter outputs shorted together
    - or shorted to ground.
- Low power consumption
  - typically 50mW per driver/receiver pair for LVDS
  - 120mW for ECL/PECL

LVDS Signalling Levels

Driver Output Voltage
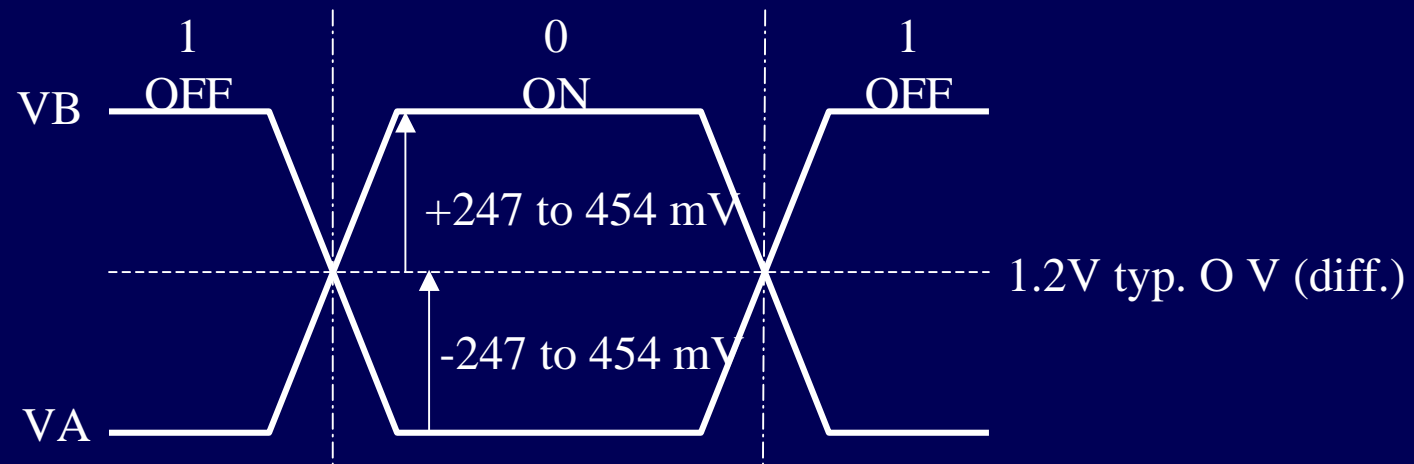
Receiver Input Thresholds
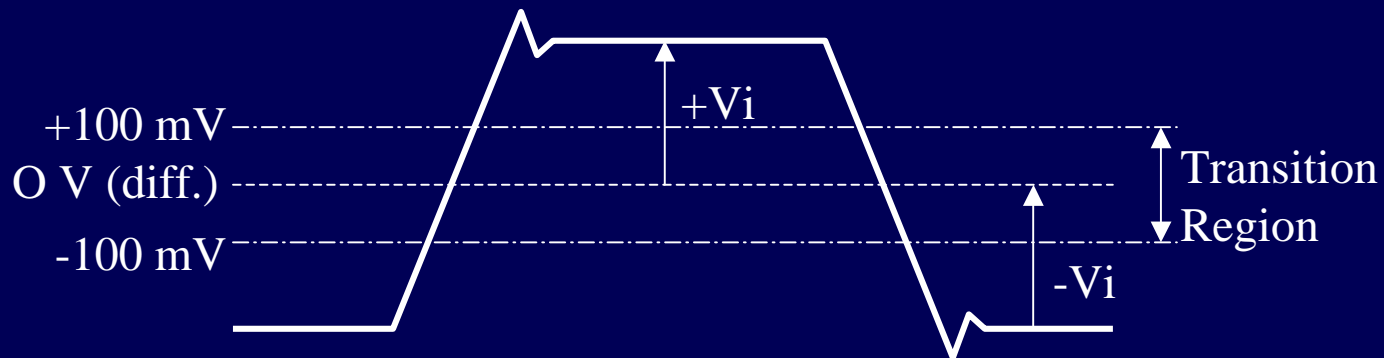
Jitter Analysis: Data TIE

H 1.00 ns/   0.0 s   0   T 0.0V
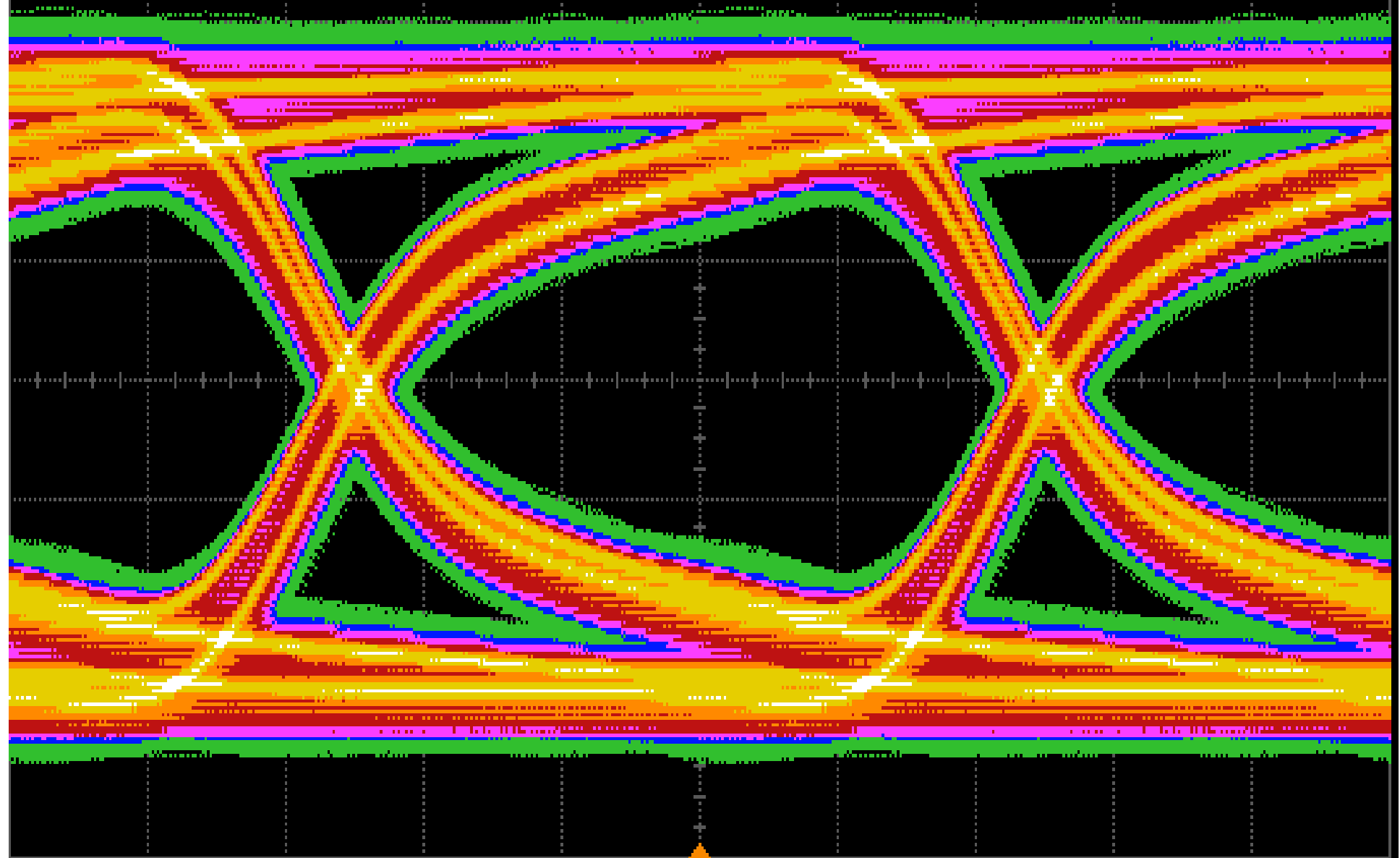
Data Encoding

- **SpaceWire uses Data-Strobe (DS) encoding**
  - Data values transmitted directly
  - Clock encoded with data to form strobe
  - XORing data and strobe recovers clock
  - provides improved jitter/skew tolerance compared to data/clock encoding
- **Used in IEEE-1355 and IEEE-1394 standards**

DATA-STROBE ENCODING

Data 0 1 0 0 1 1 0 1 1 0
D
S
CLK
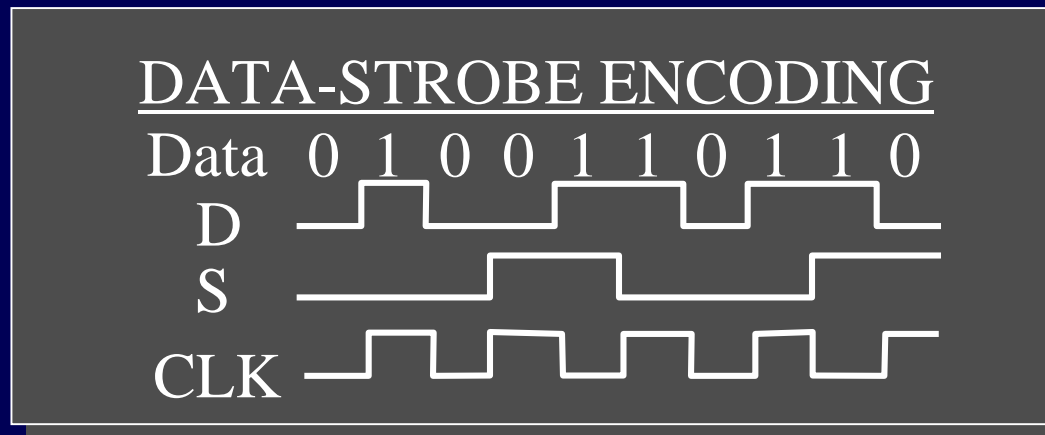
- **D and S signals transmitted differentially using LVDS**

Skew and Jitter
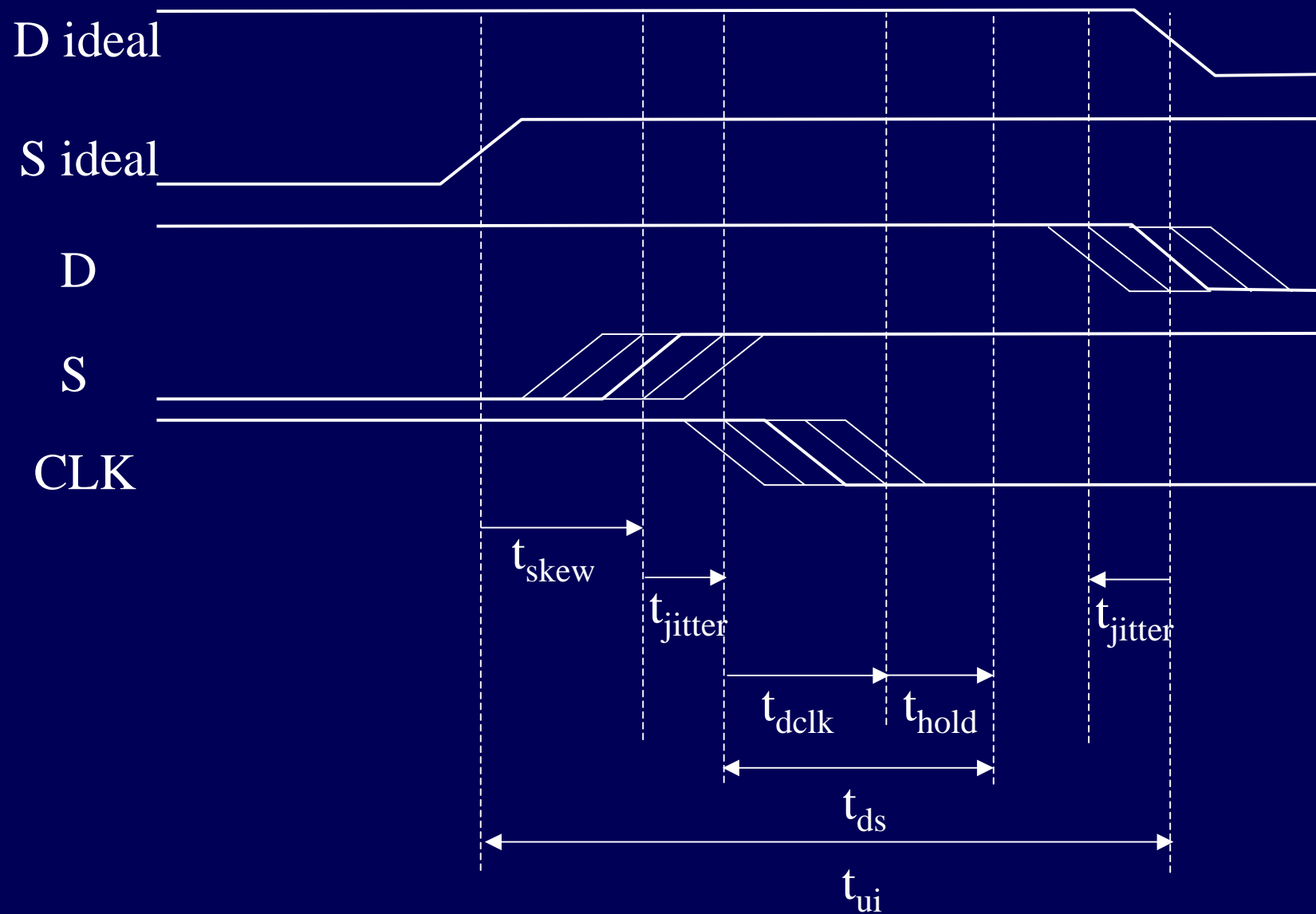
Data Signalling Rate

- Minimum data signalling rate
  - 2 Mbps
- Maximum data signalling rate
  - Depends upon system skew and jitter   >> 100 Mbps
- Operational data signalling rate
  - Any rate 2 Mbps to Maximum
  - Rx and Tx can operate at different rates
  - Tx rate may be altered at any time after link connection made
- Initial data signalling rate
  - 10 ± 1 Mbps

Character Level

- Character level defines:
  - data characters
    - used to hold data
  - control characters
    - used to manage flow of data across a link

Data Characters

P	0	X	X	X	X	X	X	X	X
		0	1	2	3	4	5	6	7

LSB   MSB

Data-Control Flag

Parity Bit
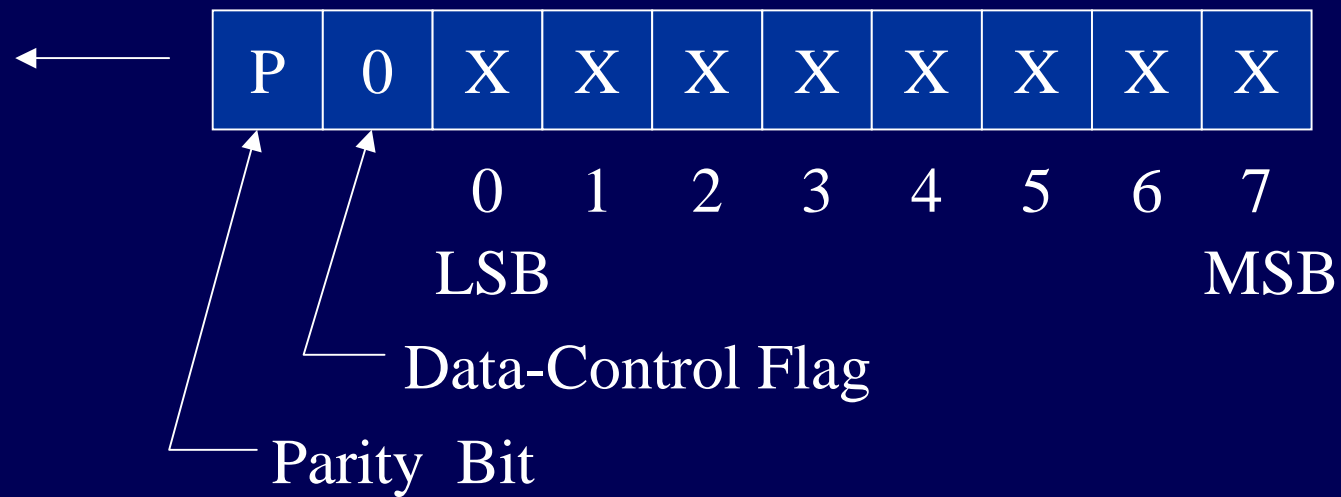
Control Characters

← | P | 1 | 0 | 0 |

FCT   Flow Control Token

← | P | 1 | 0 | 1 |

EOP   Normal End of Packet

← | P | 1 | 1 | 0 |

EEP   Error End of Packet

← | P | 1 | 1 | 1 |

ESC   Escape

Control Codes

ESC   FCT

| P | 1 | 1 | 1 | 0 | 1 | 0 | 0 | NULL

| P | 1 | 1 | 1 | 1 | 0 | T | T | T | T | T | T | T | T | Time-code

ESC   0   1   2   3   4   5   6   7

LSB   MSB

Parity Coverage

Data Character |—————— EOP —————| FCT

| P | 0 | X | X | X | X | X | X | X | P | 1 | 0 | 1 | P | 1 | 0 | 0 |
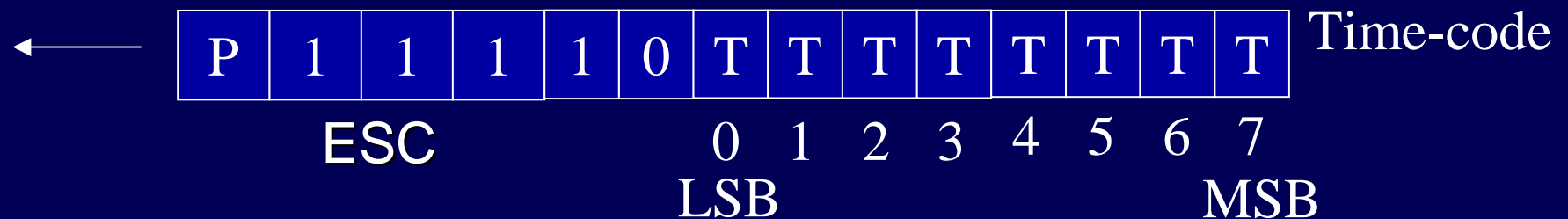
Parity
Coverage

Parity
Coverage
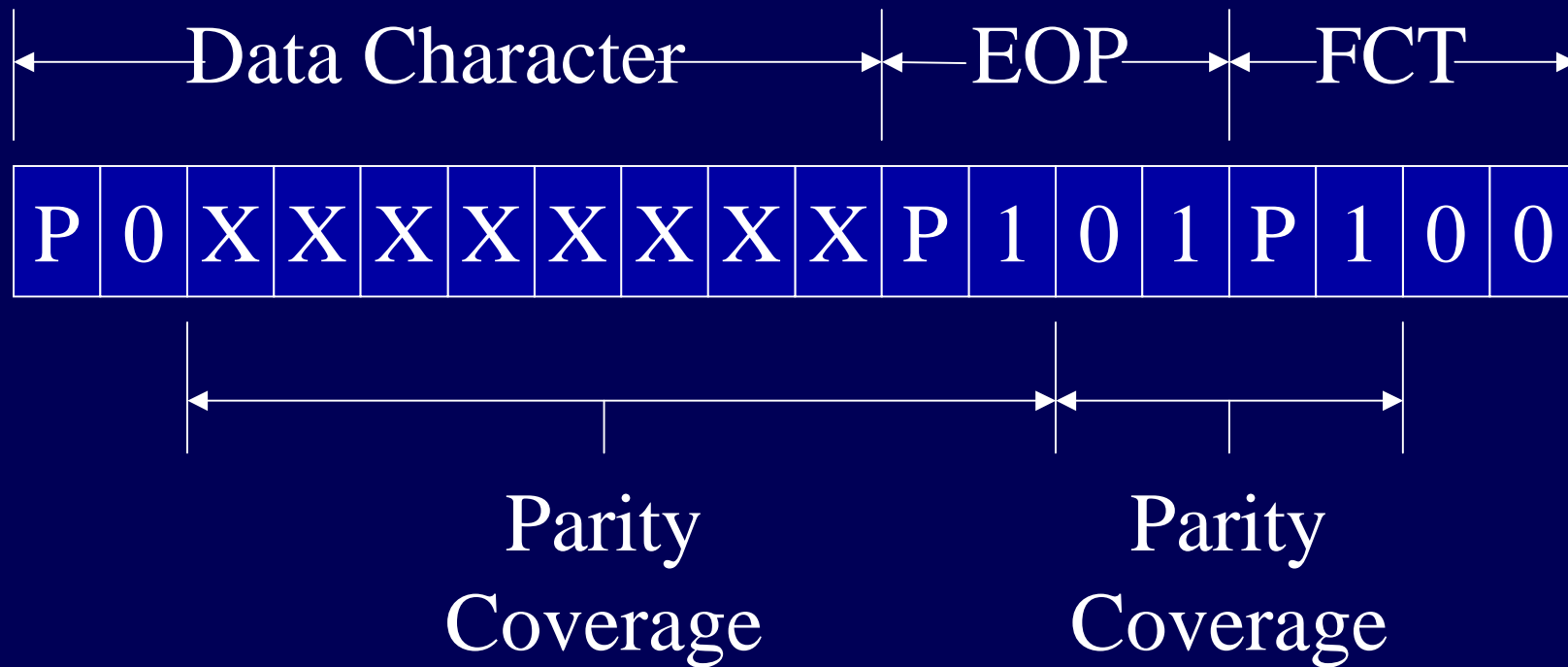
Data and Strobe on Link Start

First NULL

0	1	1	1	0	1	0	0

D

S
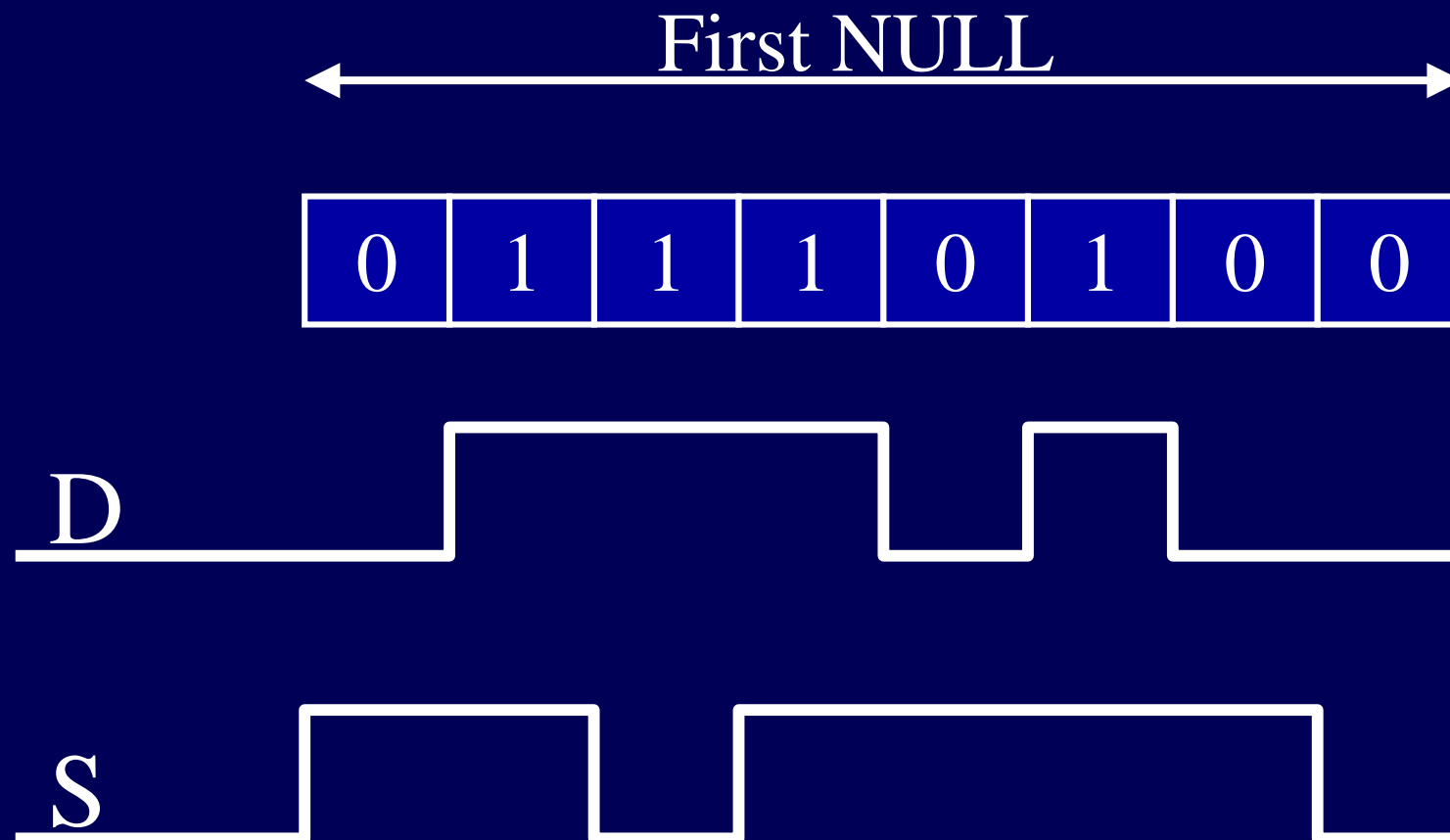
Exchange Level

- Exchange Level Protocol defines
  - Link Initialisation
  - Flow Control
  - Detection of Link Errors
  - Link Error Recovery
- Concepts from IEEE-1355 with significant improvements

SpaceWire Interface
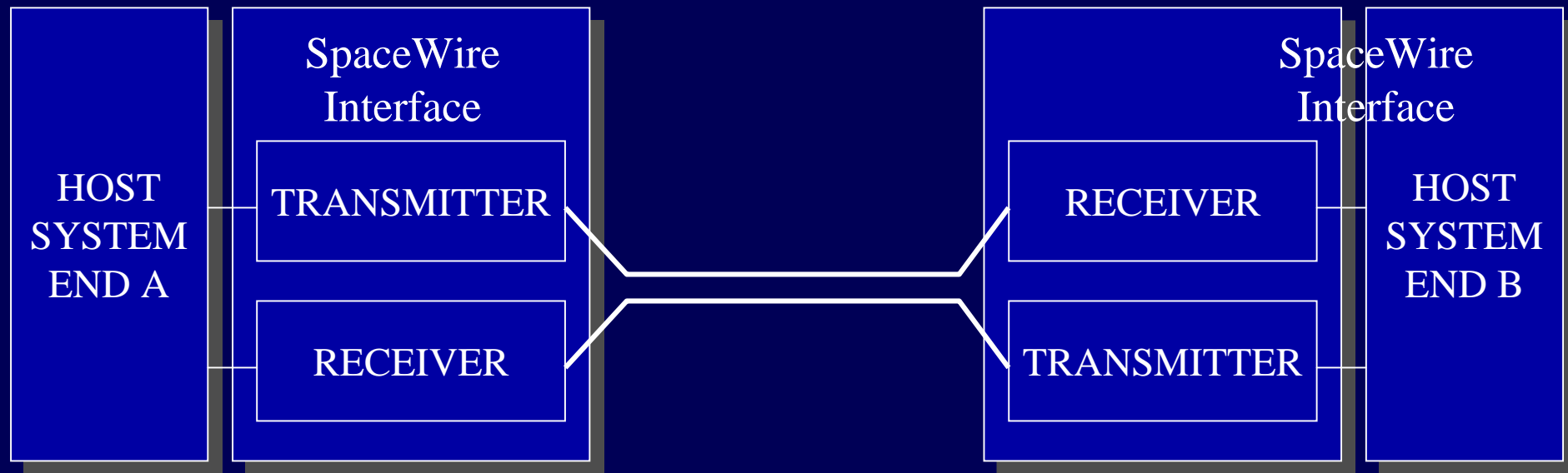
N-Chars and L-Chars

- L-Chars: Link characters (NULL, FCT)
  - used in exchange level
  - not passed on to higher level
- N-Chars: Normal characters
  - Data characters
  - End of packet markers (EOP, EEP)
  - Passed up to higher level
- Link interface interleaves L-Chars and N-Chars
- N-Chars from one packet NOT interleaved with N-Chars from another packet
- Received character must have parity checked before acted upon

Disconnect

- Disconnect detection is used for
  - monitoring physical connection
  - performing exchange of silence
- Receive clock is D XOR S
- Monitor time between one receive clock edge and next
- If > 850nsec then disconnect timeout
- Gives a lower limit to data signalling rate (2 Mbps)
- Disconnect detection enabled after first bit received
- i.e. after first receive clock edge

Transmission Priorities

- Character transmission is prioritised as follows:-
  - FCTs - highest priority
  - N-Chars
  - NULLs - lowest priority

Basic State Machine
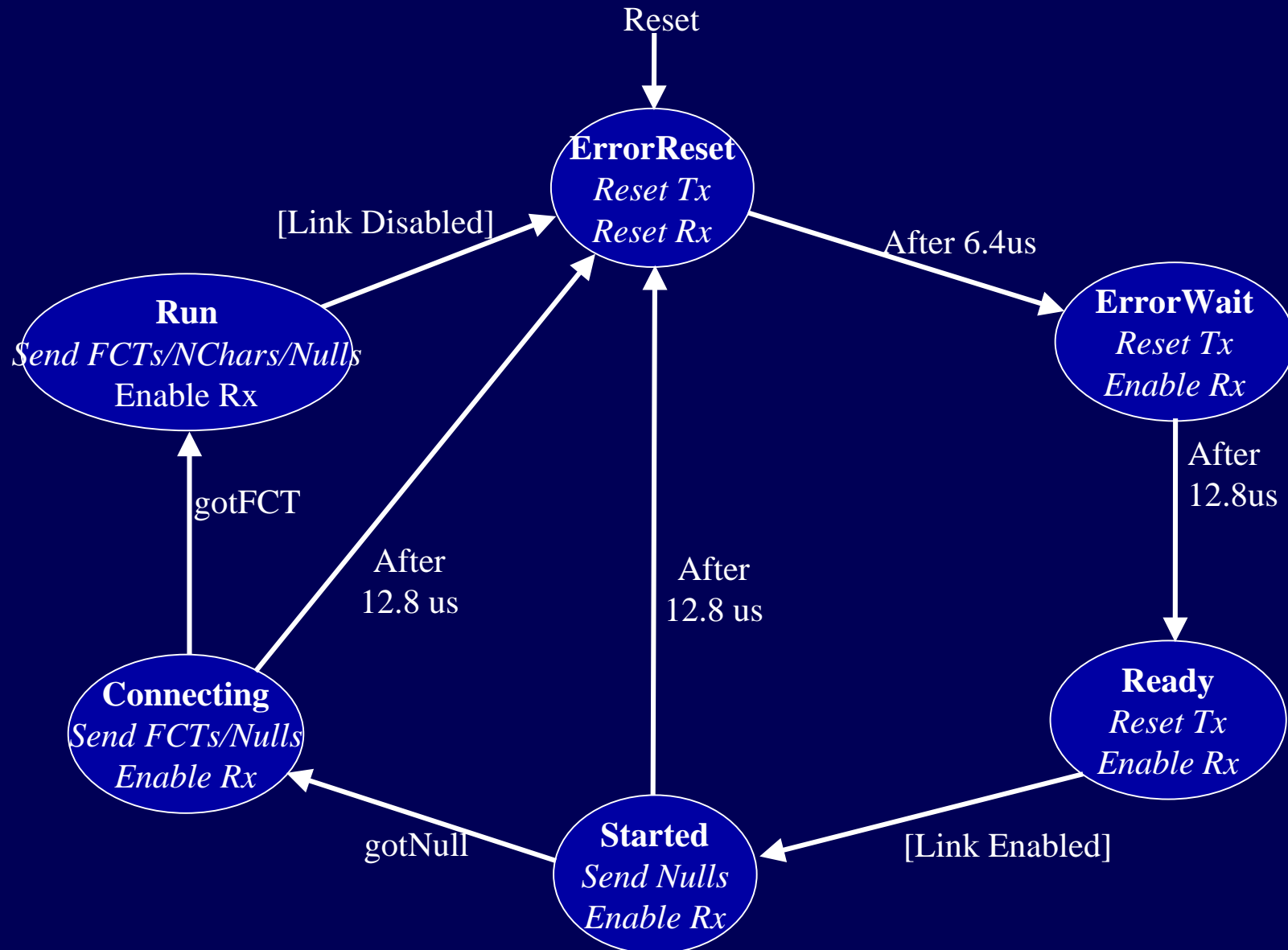
Link Initialisation (1)

End A

ErrorReset
Reset Tx, Reset Rx

After 6.4 us

ErrorWait
Reset Tx, Enable Rx

After 12.8 us

Ready
Reset Tx, Enable Rx

Link Enabled

Started
Send Nulls, Enable Rx

End B

ErrorReset
Reset Tx, Reset Rx

After 6.4 us

ErrorWait
Reset Tx, Enable Rx

After 12.8 us

Ready
Reset Tx, Enable Rx
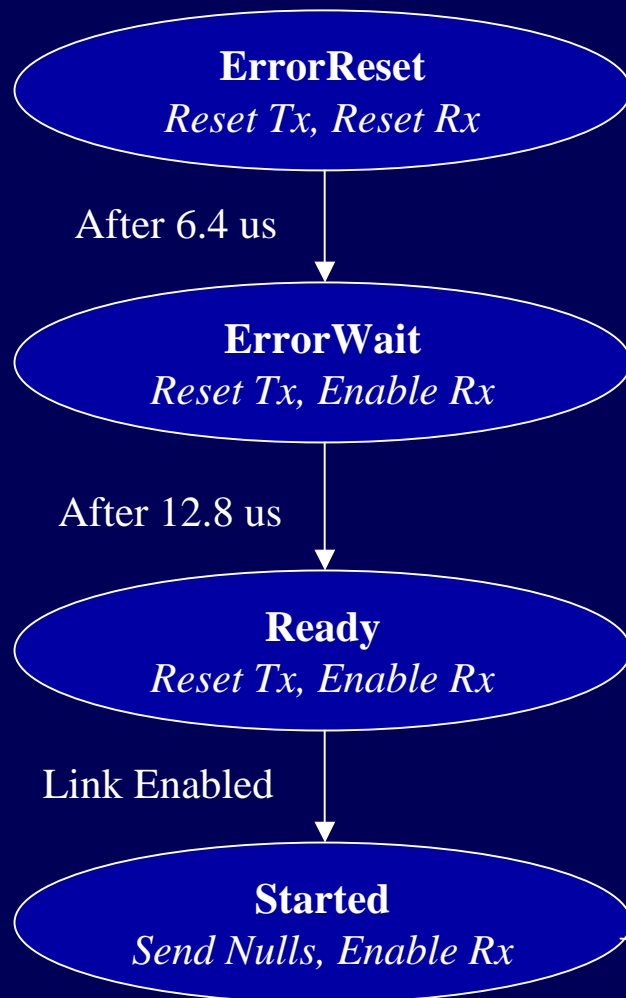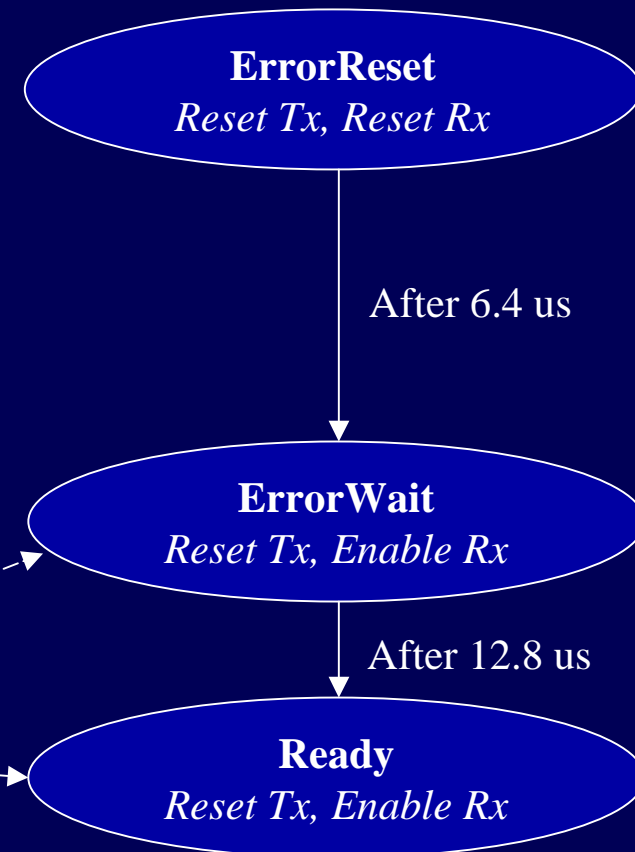
Send Nulls

Link Initialisation (2)

End A

End B
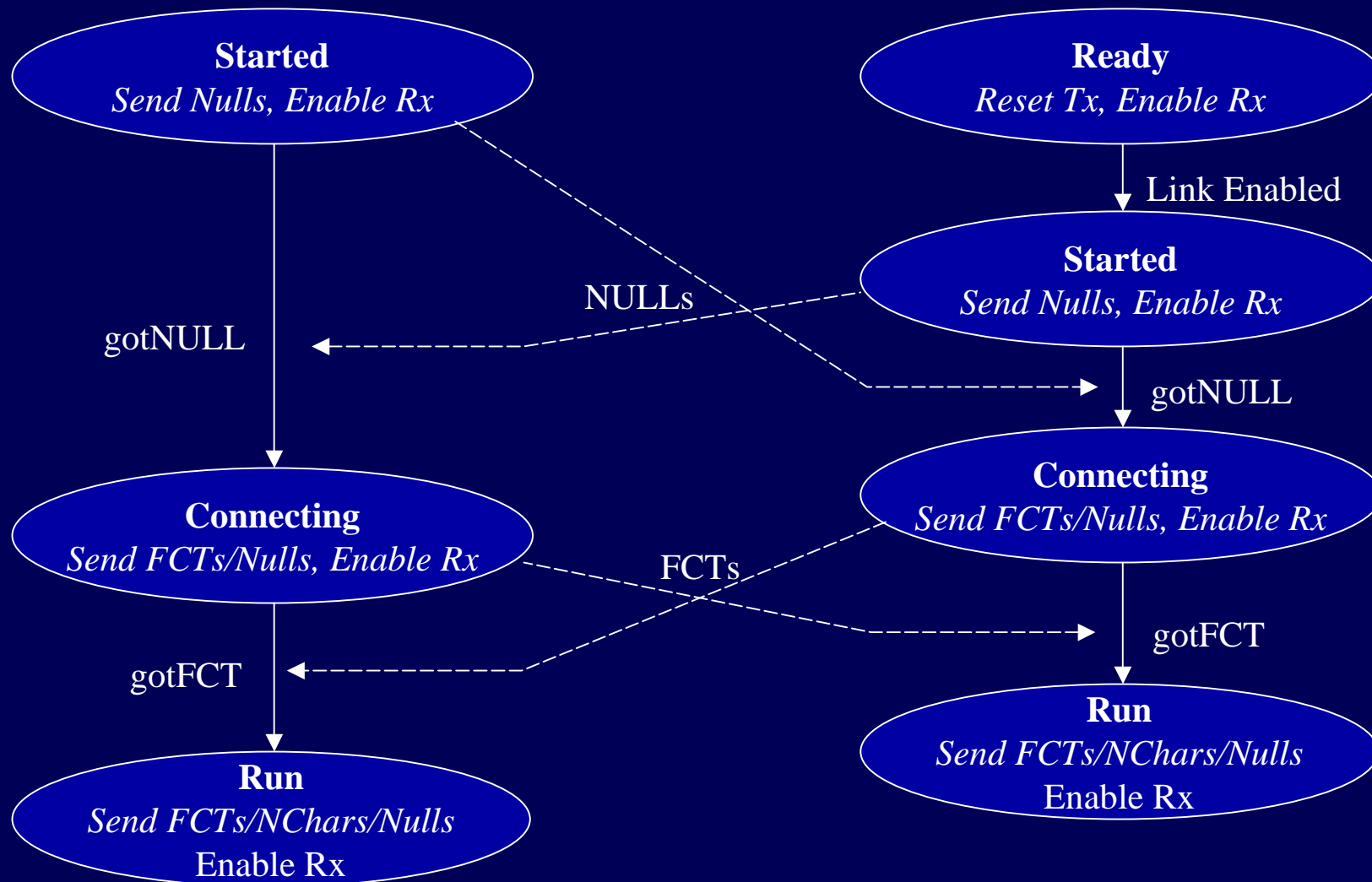
Started
Send Nulls, Enable Rx

Ready
Reset Tx, Enable Rx

Link Enabled

gotNULL

NULLs

Started
Send Nulls, Enable Rx

gotNULL

Connecting
Send FCTs/Nulls, Enable Rx

FCTs

Connecting
Send FCTs/Nulls, Enable Rx

gotFCT

gotFCT

Run
Send FCTs/NChars/Nulls
Enable Rx

Run
Send FCTs/NChars/Nulls
Enable Rx

Acquisition Time
07 Nov 2002 13:08:56

Label>	RXCHA1	RXCHA2	NCHAR1	NCHAR2	Time
Base>	Symbol	Symbol	Hex	Hex	Relativ
0		ESCAPE	000	000	
1		NULL	000	000	336
2		ESCAPE	000	000	344
3		NULL	000	000	336
4		ESCAPE	000	000	344
5	ESCAPE		000	000	176
6		NULL	000	000	160
7	NULL		000	000	184
8		ESCAPE	000	000	160
9	FCT		000	000	176
10		NULL	000	000	160
11	FCT	ESCAPE	000	000	184
12		NULL	000	000	16
13		FCT	000	000	24
14		FCT	000	000	16
15		FCT	000	000	24

Flow Control

- Flow control used to avoid overflow of host receive buffer
- Use flow control tokens (FCTs)
    - Send FCT from end A to end B
    - when end A can accept 8 more N-Chars.
    - This enables end B to send up to 8 more N-Chars.
    - Multiple FCTs can be sent if there is more room in buffer.
- FCT is exchanged for 8 N-Chars
- TX must keep a count of credit.

Normal Operation

1. Sends NULLs to maintain connection

2. Host System indicates it has room for 8 more chars

3. Transmitter sends FCT to request 8 more chars

4. FCT received enabling 8 more chars to be sent

HOST SYSTEM END A

SpaceWire Interface

TRANSMITTER

RECEIVER

SpaceWire Interface

RECEIVER

TRANSMITTER

HOST SYSTEM END B

6. N-Chars received and passed on to host system

5. Transmitter sends N-chars from host

AutoStart

- Link interface can start
  - on command from host system
  - automatically on receipt of a NULL
- [Link Enabled] = (NOT [Link Disabled]) AND ([LinkStart] OR ([AutoStart] AND gotNull))
  - **LinkDisabled** is the flag set by software or hardware to indicate that the link is disabled.
  - **LinkStart** is a flag set by software or hardware to start a link
  - **AutoStart** is a flag set by software or hardware to indicate that the link should start automatically on receipt of a NULL.
  - **gotNull** is the flag indicating that the link interface has received a NULL.
- LinkStart and AutoStart are only acted upon when the link interface is not disabled i.e. [LinkDisabled] = False.

Error Detection

- Disconnect Error
  - No RX clock transition for more that 850 ns
- Parity Error
  - Parity bit error
- Escape Error
  - ESC character should only be used to form a NULL (ESC, FCT) or time-code (ESC, Data)
  - ESC followed by any character other than FCT or data character is an error
- Credit Error
  - If there is no room in host RX buffer for data received then an error must have occurred which affected the FCTs
- Empty Packet Error
  - EOP or EEP followed by another EOP or EEP
  - Represents an empty packet
  - Not permitted

Exchange of Silence

One End
of Link

Other End
of Link

Error Detected

Reset Tx
Reset Rx

Disconnect Detected

Reset Tx
Reset Rx

After 6.4 usec

Reset Tx
Enable Rx

*Exchange
of Silence*

After 6.4 usec

Reset Tx
Enable Rx

After 12.8 usec

Send Nulls
Enable Rx

After 12.8 usec

Send Nulls
Enable Rx

Null Received

Send FCTs
Enable Rx

*NULL/FCT
Handshake*

Null Received

Send FCTs
Enable Rx

FCT Received

NORMAL
OPER.

FCT Received

NORMAL
OPER.

Full State Machine

Reset

ErrorReset
Reset Tx
Reset Rx

RxErr OR
CreditErr OR
EmptyPacketErr
[Link Disabled]

RxErr OR
gotFCT OR
gotNChar

After 6.4us

Run
Send FCTs/NChars/Nulls
Enable Rx

ErrorWait
Reset Tx
Enable Rx

gotFCT

RxErr OR
gotNChar OR
after 12.8 us

RxErr OR
gotFCT OR
gotNChar OR
after 12.8 us

RxErr OR
gotFCT OR
gotNChar

After
12.8us

Connecting
Send FCTs/Nulls
Enable Rx

Ready
Reset Tx
Enable Rx

gotNull

Started
Send Nulls
Enable Rx

[Link Enabled]

Packet Level

- **Packet Level Protocol**
  - defines packet structure
  - follows IEEE 1355 (with clarification)

SpaceWire Packets

- Packet Format

  ←   <DESTINATION> <CARGO> <END OF PACKET MARKER>

- Destination
  - represents either path to, or identity of destination node

- Cargo
  - data or message to be transferred from source to destination

- End of Packet Marker
  - indicates end of packet

Acquisition Time
07 Nov 2002 13:08:56

Label>	RXCHA1	RXCHA2	NCHAR1	NCHAR2	Time
Base>	Symbol	Symbol	Hex	Hex	Relativ
30	NULL		000	015	24
31	ESCAPE	NCHAR	000	016	16
32	NULL		000	016	24
33	ESCAPE		000	017	16
34		NCHAR	000	017	8
35	NULL		000	017	16
36	ESCAPE		000	017	16
37	NULL	NCHAR	000	018	24
38	ESCAPE		000	018	16
39	NULL		000	019	24
40		NCHAR	000	019	8
41	ESCAPE		000	019	8
42	NULL	EOP	000	100	24
43		ESCAPE	000	100	8
44	ESCAPE		000	100	8
45		NULL	000	100	16

Conclusion

- SpaceWire is an onboard communications standard designed specifically for spacecraft
- SpaceWire ECSS-E50-12A standard has been issued
- Many organisations and missions now using SpaceWire

- SpaceWire is a platform for the implementation of many onboard communication architectures

- Future work will focus on recommendations for
  - Application architectures
  - Transport layer protocols
  - etc