

车规级芯粒互联接口标准 – 草案

Road vehicles - Advanced Cost-driven Chiplet Interface (ACC_RV 1.0)

中国 Chiplet 产业联盟

2022 年 12 月

目 次

目 次	II
1 范围	1
2 规范性引用文件	1
3 术语和定义	1
4 总体概要	1
5 协议层 Protocol Layer	2
5.1 兼容 AXI4.0 模式	3
5.1.1 协议层包	3
5.1.2 包头定义	4
5.1.3 组包规则	5
5.2 兼容 CXL3.0 Standard 256B Flit 模式	9
5.2.1 协议层包	9
5.3 原生模式	9
5.3.1 协议层包定义及组包规则	9
5.4 协议层/链路层互联接口	10
6 链路层 Link Layer	11
6.1 ID 编号方法	12
6.2 CRC 计算及校验方法	12
6.3 链路层包	12
6.3.1 ACK/NAK 包定义	13
6.4 重传机制	13
6.4.1 重传规则	13
6.4.2 利用 NAK 包实现流控	14
6.5 链路适配功能	14
6.5.1 协议层/链路层包头尾标记方法	15
6.5.2 控制字符	15
6.5.3 多通道分发与合并	16
6.5.4 链路层/物理层互联接口	17
6.5.5 链路数据流举例	18
7 链路训练状态机 LTSM (Link Training State Machine)	19
7.1 概述	19
7.2 建链流程	19
7.3 NULL 码型	20
8 数字物理层 DPL (Digital Physical Layer)	21
8.1 数字物理层功能概述	21
8.2 加扰和随机码生成	23
8.3 编解码	23
8.4 数据对齐模块 (Alignment)	24
8.4.1 可信度计数器	24
8.5 链路交叉	25
8.6 通道极性反转	27

9 电气物理层 (Electrical Physical Layer)	28
9.1 数据率	28
9.2 电学参数	28
9.3 发送端	28
9.3.1 发送端的电学参数	28
9.3.2 驱动电路	29
9.3.3 发送端的前馈均衡	29
9.4 接收端	30
9.4.1 接收端的电学参数	31
9.4.2 连续时间线性均衡器 (CTLE)	31
9.4.3 时钟数据恢复电路	32
10 边带传输 Sideband (Optional)	32
11 接口定义	33
11.1 协议层与链路层接口信号定义	34
11.2 链路层与数字物理层接口信号定义	35
11.3 数字物理层与电气物理层接口信号定义	35
11.4 近端与远端接口信号定义	35
12 调试追踪 Debug&Trace	36
12.1 测试序列发送、检测和校验功能	36
12.2 环回功能	37
13 功能安全设计	37
13.1 存储保护	38
13.2 端到端总线保护	38
13.3 预警上报	39
13.4 存储器自测试及修复 (MBIST and Memory repair)	39
13.5 逻辑自测试 (LBIST)	40
13.6 安全岛控制	40
13.7 信道冗余及重映射设计	40
13.8 信道复用模式及快速测试	42
13.9 信道的 Loopback 测试	42
14 配置参数	43
15 封装 Package	44
15.1 封装类型	45
15.2 Bump 方案	45
15.3 绕线方案	45
15.4 信号走线规范	45
15.5 时钟方案	46
15.6 信号完整性	47

1 范围

本标准给出了芯粒互联场景下的协议层、链路层和物理层，并规定了各层的消息报格式、工作模式、传输方式等，以及调试追踪方法、接口配置参数、封装方案等。

本标准适用于各种芯粒在 32Gbps-128Gbps 速率互联互通场景下的接口通信。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

3 术语和定义

缩略语	英文全拼	中文示意
AMBA	Advanced Microcontroller Bus Architecture	高级微控制器总线架构
AXI	Advanced eXtensible Interface	高性能扩展总线接口
CXL	Compute Express Link	计算机高速互联总线
RAW	Raw Data	原始数据
TLP	Transaction Layer Packet	协议层包
DLP	Data Link Packet	链路层包
ACK	Acknowledge	正确接收协议层包的响应链路层包
NAK	Non Acknowledge	错误接收协议层包的响应链路层包
CRC	Cyclic Redundancy Check	循环冗余校验
STP	Start TLP	协议层包传输开始标记
SDP	Start DLP	链路层包传输开始标记
COM	Comma	间隙控制字符
IDL	Idle	空闲控制字符
PAD	Padding	填充控制字符
ISI	Inter-Symbol Interference	码间干扰

4 总体概要

本标准自顶向下，划分为协议层、链路层和物理层。如图4-1所示：

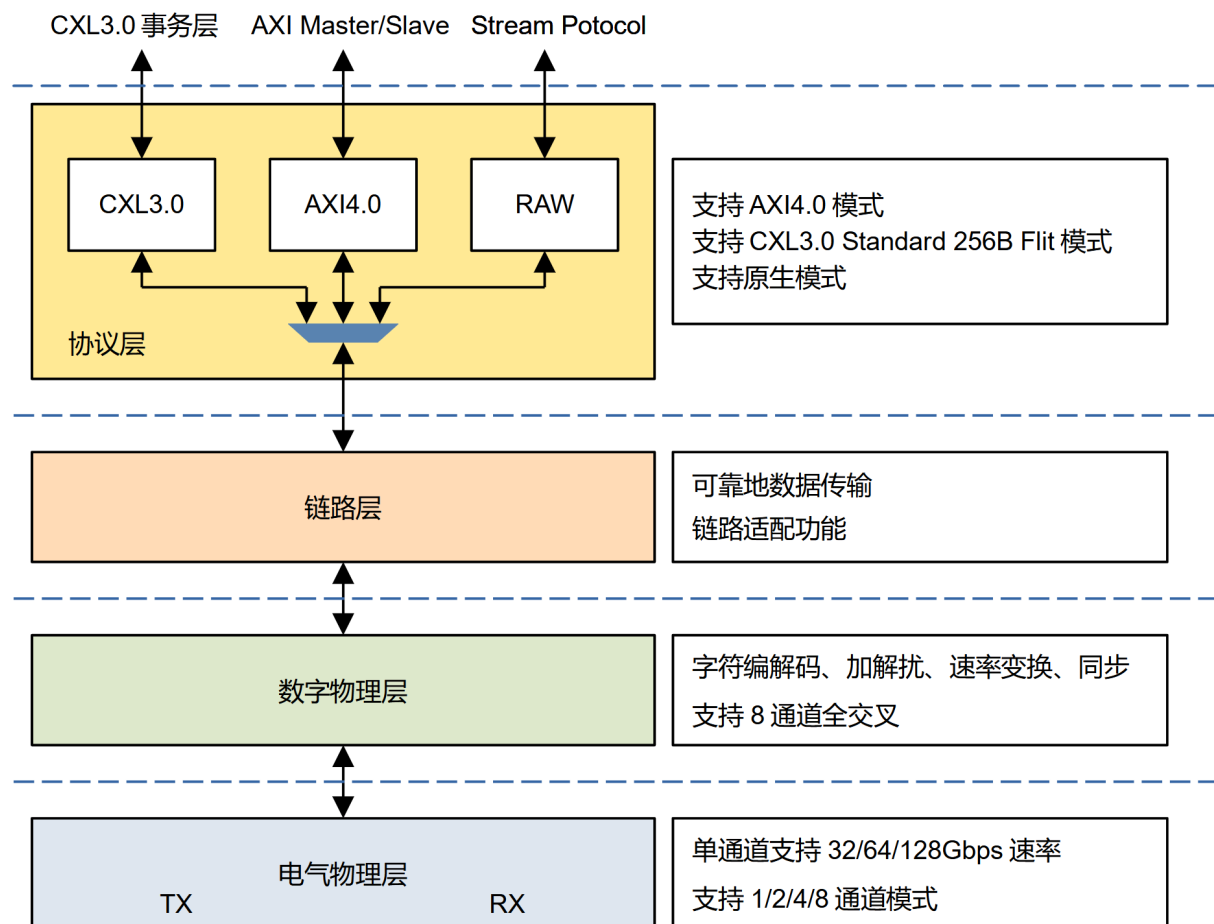


图4-1 协议总览

为了能够适用各类业务Die的Chiplet对接通信，只要遵循同链路层的接口（PLI）协议设计，便可以向上、灵活地对接多种系统协议。本标准默认以常见的高速总线系统AMBA AXI4为例，描述了协议层的具体实现。

链路层提供了快速的建链流程，同时支持自动检测和出错重传，以提供无差错传输。

物理层包含数字物理层和电气物理层。其中数字物理层承担高速数据传输必须的加解扰、编解码、数据对齐、链路交叉、极性反转等功能。而电气物理层采用超短距高速数据传输技术，满足高性能Die的高带宽、低延迟传输。

电气物理层的关键性能指标，如表4-1所示：

表4-1 关键性能指标

	标准封装		
数据率 (GT/s)	32	64	128
编码方式	NRZ	PAM4	PAM4
能效 ¹ (pJ/b)	2.5	2.5	2.5
延迟时间 (TX+RX) ² (ns)	6	6	6
带宽线密度(GT/s/mm)	128	256	512
误码率	10 ⁻¹⁵	10 ⁻¹²	10 ⁻⁹

注1：Power Efficiency Target包含所有电气物理层相关电路：TX，RX，PLL，Clock；

注2：Latency Target指的是电气物理层的延迟，包含TX和RX；

5 协议层 Protocol Layer

本标准支持AMBA AXI4.0协议和CXL3.0协议定义的Standard 256B Flit模式，此外还支持原生模式，即可将任意协议定义的数据流按照用户自定义的规则放置到本标准定义的协议层包中，协议层包做为载体将数据流传递到对端协议层。将其划分为如下3种模式：

- 兼容AXI4.0模式；
- 兼容CXL3.0 Standard 256B Flit模式；
- 原生模式。

设计人员可根据需求，在设计系统及上电初始化时配置所需的模式。

5.1、5.2、5.3章节详细介绍兼容AXI4.0模式、兼容CXL3.0 Standard 256B Flit模式和原生模式下的协议层包格式，5.4章节详细介绍协议层/链路层互联接口。

5.1 兼容 AXI4.0 模式

兼容AXI4.0模式是可选的，该模式支持AXI4.0协议定义的传输方式，AXI4.0总线位宽支持512bit。

5.1.1 协议层包

兼容AXI4.0模式协议层包格式定义如图5-2所示，其中STP字段预留2个字节，CRC字段预留8个字节，END字段预留6个字节由链路层填充，包长为128字节整数倍，最大长度是640字节，协议层负责将AXI4.0协议信号映射到包头（Head）和负载区域（Payload），映射方法参见5.1.2章节和5.1.3章节。协议层包如图5-1所示。

Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	STP		RSV					Head								Payload																
32	Payload																															
64	Payload																															
96	Payload																															
128	Payload																															
160	Payload																															
192	Payload																															
224	Payload																															
256	Payload																															
288	Payload																															
320	Payload																															
352	Payload																															
384	Payload																															
416	Payload																															
448	Payload																															
480	Payload																															
512	Payload																															
544	Payload																															
576	Payload																															

608	Payload	RSV	CRC	END
-----	---------	-----	-----	-----

图5-1 兼容AXI4.0模式下协议层包格式

5.1.2 包头定义

包头用于标识负载区域，包头内各字段位宽和位置如图5-2所示，其中Ax表示AW/AR/B通道数据组成的协议层包，W表示W通道数据组成的协议层包，R表示R通道数据组成的协议层包：

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31																															
Ax	T(3'b000)			R	C_0		C_1		RSV						CN		RSV																																														
W	T(3'b101)			RSV										TL			R	WL	RSV			ST	RSV																																								
R	T(3'b110)			RSV										TL			RSV																																														
Bit	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63																															
Ax	RSV																																																														
W	RSV																																																														
R	RSV																																																														

图5-2 AW/AR/B/W/R通道协议层包包头内各字段位宽和位置

包头内各字段含义如表5-1所示：
表5-1 包头字段列表

字段命名	字段功能描述
T[2:0]	用于标识协议层包类型，具体有如下 7 种，其余值保留： 3'b000: AW/AR/B 通道 3'b001: 中断拉高，不在标准中规定，由用户自定义实现 3'b010: 中断拉低，不在标准中规定，由用户自定义实现 3'b011: 清中断，不在标准中规定，由用户自定义实现 3'b100: PRBS 序列，不在标准中规定，由用户自定义实现 3'b101: W 通道 3'b110: R 通道
C_x[1:0]	AW/AR/B 通道 2 个命令识别码，优先使用 C_0，其次 C_1： 2'b00: AW 通道 2'b01: AR 通道 2'b10: B 通道
CN[0:0]	AW/AR/B 通道命令个数： 1'b0: 1 个命令 1'b1: 2 个命令
TL[2:0]	当前协议层包包含的 W/R 通道传输的个数： 3'b000: 1 个传输 3'b001: 2 个传输 3'b010: 3 个传输 3'b011: 4 个传输 3'b100: 5 个传输 3'b101: 6 个传输 3'b110: 7 个传输 3'b111: 8 个传输
WL[0:0]	当前协议层包的 W 通道的最后一个传输是 WLAST

ST[0:0]	当前协议层包的 W 通道的 WSTRB 信号压缩类型： 1'b0: 所有传输都带 WSTRB 信号 1'b1: 只有头尾两笔传输带 WSTRB 信号，中间位置自动补成全 1
R/RSV	保留字段

5.1.3 组包规则

5.1.3.1 AW/AR/B 通道组包规则

包头的类型字段为3'b000表示当前协议层包是AW/AR/B通道组成的协议层包，支持最多放置2个AW/AR/B通道命令到1个协议层包，优先放置到A_0，其次A_1，组包格式如图5-3所示：

Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	STP		RSV						Head								A_0[63:0]															
32	A_1[63:0]																RSV															
64	RSV																															
96	RSV																	CRC							END							

图5-3 兼容AXI4.0模式下AW/AR/B通道协议层包格式定义

A_0/A_1内各字段位宽和位置如图5-4所示：

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Ax	AL	R	ABST		ASIZE			APROT			ALEN					ACACHE				AREGION				AQOS			ADDR[0:3]					
B	BRSP		RSV																													
Bit	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
Ax	ADDR[4:35]																															
B	RSV																															
Bit	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
Ax	ADDR[36:63]																												AID[0:3]			
B	RSV																												BID[0:3]			
Bit	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
Ax	AID[4:7]				AUSER																RSV											
B	BID[4:7]				BUSER																RSV											

图5-4 兼容AXI4.0模式下协议层包A_0/A_1内各字段位宽和位置

A_0/A_1内各字段含义如表5-2所示：

表5-2 A_0/A_1字段列表

字段命名	字段功能描述
ADDR[63:0]	AW/AR 通道 ADDR 信号
ALEN[5:0]	AW/AR 通道 LEN 信号
AID[7:0]	AW/AR 通道 ID 信号

ASIZE[2:0]	AW/AR 通道 SIZE 信号
AQOS[3:0]	AW/AR 通道 QOS 信号
AUSER[15:0]	AW/AR 通道 USER 信号
AL[0:0]	AW/AR 通道 LOCK 信号
ABST[1:0]	AW/AR 通道 BURST 信号
APROT[2:0]	AW/AR 通道 PROT 信号
ACACHE[3:0]	AW/AR 通道 CACHE 信号
AREGION[3:0]	AW/AR 通道 REGION 信号
BID[7:0]	B 通道 ID 信号
BRSP[1:0]	B 通道 RESP 信号
BUSER[15:0]	B 通道 USER 信号
R/RSV	保留字段

5.1.3.2 W 通道组包规则

AXI4.0协议W通道主要传递的信号是WS

TRB/WDATA，为了节省链路带宽，本标准提出一种基于可变长有效数据动态排列的数据压缩方法，将W通道的带宽利用率提高到实际带宽的70%至95%。

链路层的重传和流控是以协议层包为最小单位进行控制的，若协议层包过大会影响链路控制精细度，且会占用更多的重传缓存，影响架构设计；若协议层包过小会导致开销和有效数据比例变大，影响链路传输效率。权衡之后确定W通道协议层包最多可以装8个传输，最少可以装1个传输，遇到WLAST则停止组包。

W通道拼接数据时，会根据情况附带WSTRB，若除首尾两个传输的WSTRB外，中间传输的WSTRB为全1，则省略中间传输的WSTRB，将包头的ST字段置为1；若中间传输的WSTRB不为全1，则全部传输都带WSTRB，将包头的ST字段置为0。

包头的类型字段为3'b101表示当前协议层包是W通道组成的协议层包，根据WSTRB指示的有效数据长度，将512bit的WDATA压缩之后放入协议层包，压缩方法是先将64bit的WSTRB视作8个字节，每个字节缩位或之后得到1bit，共计8bit。每1个bit对应64bit的WDATA，将缩位或之后，值为1所指示的WDATA放入协议层包，如图5-5所示。

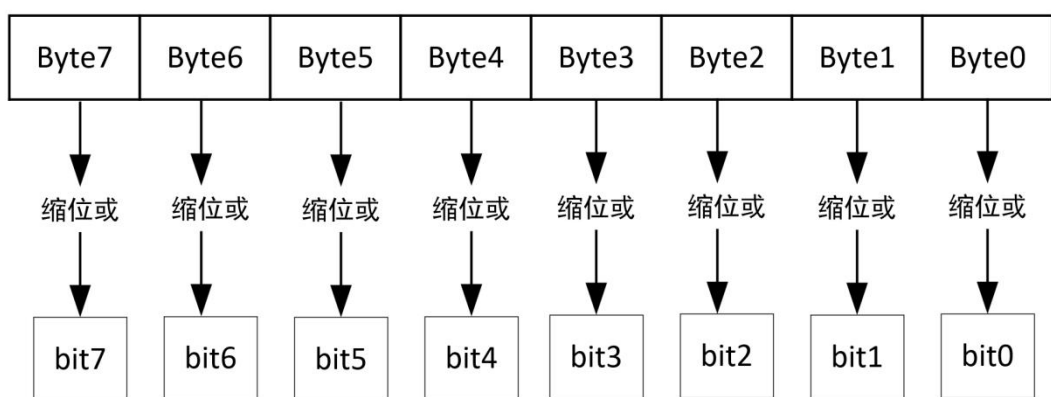


图5-5 WSTRB缩位或

注意：只支持缩位或之后值为1的区域是连续的情况，比如8'b00111000/8'b11111100等，当出现8'b10001000等值为1的区域不连续的情况时，需要将其补全为8'b11111000，并按照补全后的值来组包和解包。

组包格式举例如图5-6(a)至5-6(d)，其中PREFIX包含STP/Rev/Head字段，POSTFIX包含CRC/END字段：

Byte	0	8	16	24	32	40	48	56	64	72	80	88	96	104	112	120
------	---	---	----	----	----	----	----	----	----	----	----	----	----	-----	-----	-----

0	PREFIX	WA	S_0	WD_0(可变长)	WD_1(64 字节)
128	WD_1(64 字节)			WD_2(64 字节)	WD_3(64 字节)
256	WD_3(64 字节)			WD_4(64 字节)	WD_5(64 字节)
384	WD_5(64 字节)			WD_6(64 字节)	S_7 WD_7(可变长)
512	WD_7(可变长)			RSV	POSTFIX

图5-6(a) W通道协议层包举例（头尾带S标记，8个传输中第1个和第8个带S_0/S_7，用于标记WD_0/WD_7长度和位置）

Byte	0	8	16	24	32	40	48	56	64	72	80	88	96	104	112	120
0	PREFIX	WA	S_0	WD_0(可变长)	WD_1(64 字节)											
128	WD_2(64 字节)								WD_3(64 字节)							
256	S_4	WD_4(可变长)								RSV				POSTFIX		

图5-6(b) W通道协议层包举例（头尾带S标记，5个传输中第1个和第5个带S_0/S_4，用于标记WD_0/WD_4长度和位置）

Byte	0	8	16	24	32	40	48	56	64	72	80	88	96	104	112	120
0	PREFIX	WA	S_0	WD_0(可变长)	S_1	WD_1(可变长)										
128	S_2	WD_2(可变长)					S_3	WD_3(可变长)							S_4	
256	WD_4(可变长)					S_5	S_6	RSV					POSTFIX			

图5-6(c) W通道协议层包举例（全带S标记，7个传输，其中任意S_x可以是全0，表示该笔传输无有效数据）

Byte	0	8	16	24	32	40	48	56	64	72	80	88	96	104	112	120
0	PREFIX		WA		S_0	WD_0(可变长)			S_1	WD_1(可变长)						
128	S_2	WD_2(可变长)					S_3	WD_3(可变长)							S_4	
256	WD_4(可变长)							S_5	WD_5(可变长)							
384	S_6	WD_6(可变长)					S_7	WD_7(可变长)							RSV	
512	RSV														POSTFIX	

图5-6(d) W通道协议层包举例（全带S标记，8个传输，当最后128字节无法预留CRC/END时，需要将包长增加128字节）

WA内各字段位宽和位置如图5-7所示：

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	WUSER_0																WUSER_1															
Bit	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
	WUSER_2																WUSER_3															
Bit	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
	WUSER_4																WUSER_5															
Bit	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127

图5-9 R通道协议层包RA内各字段位宽和位置

RA内各字段含义如表5-4所示：

表5-4 RA字段列表

字段命名	字段功能描述
RUSER_x[15:0]	R 通道 USER 信号，RUSER_0 对应第 1 个传输，RUSER_7 对应第 8 个传输
RID_x[7:0]	R 通道 ID 信号，RID_0 对应第 1 个传输，RID_7 对应第 8 个传输
RL[7:0]	R 通道 RLAST 信号，RL[0]对应第 1 个传输，RL[7]对应第 8 个传输
RP_x[1:0]	R 通道 RRESP 信号，RP_0 对应第 1 个传输，RP_7 对应第 8 个传输

5.2 兼容 CXL3.0 Standard 256B Flit 模式

兼容CXL3.0 Standard 256B Flit模式是可选的。

5.2.1 协议层包

兼容CXL3.0 Standard 256B Flit模式协议层包格式定义如图5-10所示，其中STP字段预留2个字节，CRC 字段预留8个字节，END字段预留6个字节由链路层填充，包长固定为256字节，协议层负责将CXL3.0协议 Standard 256B Flit模式下定义的FlitHdr区域映射到FH区域，将FlitData区域映射到负载区域。协议层包如图5-10所示。

Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	STP		Payload																													
32	Payload																															
64	Payload																															
96	Payload																															
128	Payload																															
160	Payload																															
192	Payload																															
224	Payload																	CRC								FH		END				

图5-10 兼容CXL3.0 Standard 256B Flit模式下协议层包格式

5.3 原生模式

原生模式是可选的，当用户不需要承载AXI4.0协议或CXL3.0 Standard 256B Flit模式数据流时，可以选择原生模式，只要组包符合5.3.1章节描述的规则，仍然可以使用链路层重传功能。

5.3.1 协议层包定义及组包规则

原生模式协议层包格式定义如图5-11所示，其中STP字段预留2个字节，CRC字段预留8个字节，END字段预留6个字节，由链路层填充，包长为128字节整数倍，最大长度是640字节，协议层负责将任意协议信号映射到负载区域，映射方法由用户自定义。协议层包如图5-11所示。

Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	STP		Payload																													
32	Payload																															
64	Payload																															
96	Payload																															
128	Payload																															
160	Payload																															
192	Payload																															
224	Payload																															
256	Payload																															
288	Payload																															
320	Payload																															
352	Payload																															
384	Payload																															
416	Payload																															
448	Payload																															
480	Payload																															
512	Payload																															
544	Payload																															
576	Payload																															
608	Payload																	CRC								END						

图5-11 原生模式下协议层包格式

5.4 协议层/链路层互联接口

协议层/链路层接口如表5-5所示：
表5-5 协议层/链路层互联接口

信号命名	信号功能描述
prot2link_valid[0:0]	发送侧，协议层向链路层指示 prot2link_data/prot2link_tail 上的数据有效
link2prot_rdy[0:0]	发送侧，链路层向协议层指示可以接收数据，使用 valid-rdy 握手机制传递数据
prot2link_data[1023:0]	发送侧，协议层向链路层传递的协议层包的其中 1 拍数据
prot2link_tail[0:0]	发送侧，协议层向链路层传递的协议层包的尾标记
link2prot_valid[0:0]	接收侧，链路层向协议层指示 link2prot_data/link2prot_tail 上的数据有效
prot2link_rdy[0:0]	接收侧，协议层向链路层指示可以接收数据，使用 valid-rdy 握手机制传递数据
link2prot_data[1023:0]	接收侧，链路层向协议层传递的协议层包的其中 1 拍数据
link2prot_tail[0:0]	接收侧，链路层向协议层传递的协议层包的尾标记

协议层到链路层接口时序举例如图所示：

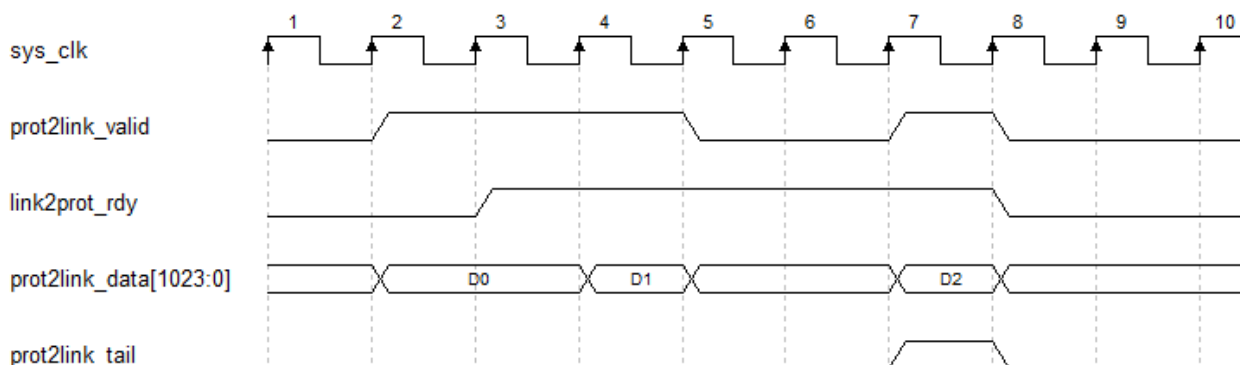


图5-13 协议层到链路层接口时序

链路层到协议层接口时序举例如图5-12所示：

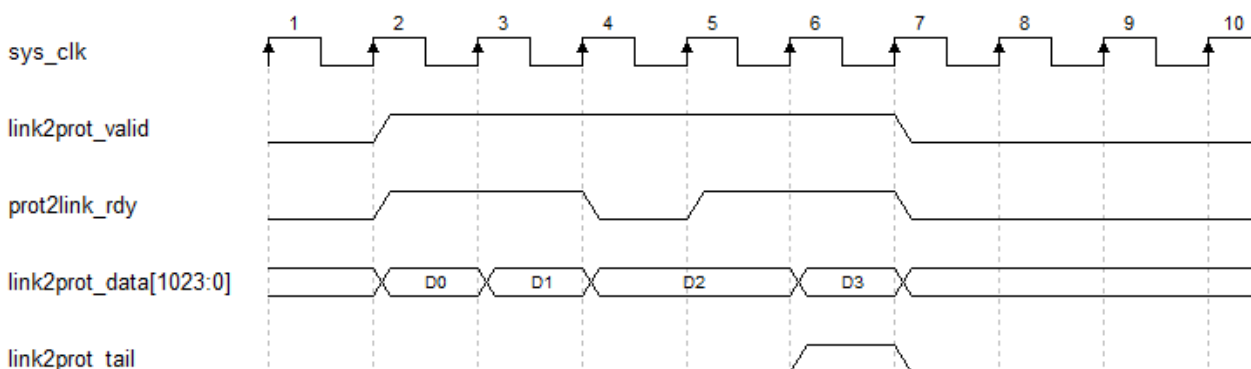


图5-12 链路层到协议层接口时序

6 链路层 Link Layer

协议层和链路层以协议层包为最小单元传输数据。链路层负责将协议层包可靠地传输到对端，利用 ACK/NAK 机制保证数据传输的可靠性。为此需要先对协议层包进行编号，然后对编号和负载区域做 CRC 校验位的计算，并将 CRC 计算结果放置到协议层包的 CRC 字段，由对端链路层对接收到的协议层包做 CRC 校验，决定是否需要重传。后续描述中涉及到控制字符的部分请参见 6.5.2 章节。

链路层发送端支持如下功能：

- 填充协议层包 ID 编号；
- 填充协议层包 CRC 校验位；
- 发送协议层包，并利用接收端反馈的 ACK/NAK+ID 编号指令释放/重传协议层包；
- 发送 ACK/NAK 包。

链路层接收端支持如下功能：

- 对链路层包进行 CRC 校验；
- 接收 ACK/NAK 包并控制 TX 重传缓存待重传的数据；
- 对协议层包进行 CRC 校验，并将校验结果整合到 ACK/NAK 包请求发送端发送给远端；
- 利用 NAK 包流控。

6.1 ID 编号方法

本标准规定协议层包中STP字段的bit[15:8]用于放置协议层包ID，ID编号0~255循环递增，255计满翻转到0，复位初始值为0，STP字段内ID编号位置如图6-1所示：

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	STP 字符的 bit[7:0]								ID							

图6-1 STP字段内ID编号位置

6.2 CRC 计算及校验方法

如图6-2所示，将协议层包每16字节划分为1个独立的CRC校验区域，共划分为8个独立的CRC校验区域。每个CRC校验区域对应1个CRC校验单元，每个CRC校验单元输入16字节数据，加上1字节的初始值或同一个协议层包上1拍该位置计算得到的CRC校验值，计算得到新的1字节CRC校验值，注意STP（不含ID）/CRC/END字段用0占位送入CRC计算单元。在协议层的包尾处，将总共8个字节的CRC校验值填充到CRC字段。

协议层包的CRC产生多项式为 $1+x^5+x^7+x^8$ ，verilog代码在附录中提供。

Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	CRC_0																CRC_1															
32	CRC_2																CRC_3															
64	CRC_4																CRC_5															
96	CRC_6																CRC_7															
128	CRC_0																CRC_1															
160	CRC_2																CRC_3															
192	CRC_4																CRC_5															
224	CRC_6																CRC_7															
256	CRC_0																CRC_1															
288	CRC_2																CRC_3															
320	CRC_4																CRC_5															
352	CRC_6																CRC_7															
384	CRC_0																CRC_1															
416	CRC_2																CRC_3															
448	CRC_4																CRC_5															
480	CRC_6																CRC_7															
512	CRC_0																CRC_1															
544	CRC_2																CRC_3															
576	CRC_4																CRC_5															
608	CRC_6																CRC_7															

图6-2 CRC校验区域划分

6.3 链路层包

链路层包（Data Link Packet，缩写为DLP）使用SDP字符标记头，END字符标记尾。链路层包和协议层包共享链路带宽，优先调度链路层包（本标准规定最小回复链路层包间隔为40ns，避免占用过多链路带宽），其次调度协议层包，链路层包定义如图6-3所示：

Byte	0	8	16	24	32	40	48	56	64	72	80	88	96	104	112	120
	SDP	DLP	END	PAD	PAD	PAD	PAD	PAD	PAD	PAD	PAD	PAD	PAD	PAD	PAD	PAD

图6-3 链路层包定义

6.3.1 ACK/NAK 包定义

ACK/NAK包格式定义如图6-4所示，通过bit[15]区分ACK/NAK。若接收端接收到的协议层包通过CRC校验且ID符合预期，则将协议层包的ID填充到ACK/NAK包的ID字段。

链路层包的CRC产生多项式为 $1+x^2+x^{15}+x^{16}$ ，输入为6字节（不含CRC字段），输出2字节CRC校验值填充到CRC字段，verilog代码在附录中提供。

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
ACK	8'ha5								RSV								0	ID								RSV							
NAK	8'ha5								RSV								1	ID								RSV							
Bit	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	
ACK	RSV																CRC																
NAK	RSV																CRC																

图6-4 链路层ACK/NAK包格式

6.4 重传机制

协议层包需要完整保存在链路层发送端的重传缓存中，当发送端收到ACK/NAK握手之后，则允许从重传缓存中释放已经握手完成的协议层包。设计电路时，需要将重传缓存的大小设计为大于2倍链路发送端链路层到接收端链路层缓存的数据拍数，从而允许接收端能适当放大回复ACK的间隔，提高链路传输效率，本标准规定最小回复ACK间隔为40ns。

6.4.1 重传规则

链路层发送端将所有协议层包缓存在重传缓存中，只在收到有效的ACK/NAK包之后，释放ACK/NAK包携带的ID之前（含）的协议层包，若收到NAK包，则重传NAK包携带的ID编号加1的协议层包；若长时间未收到有效的ACK/NAK包，则上报超时告警，并从上一次收到的有效的ACK/NAK包释放的协议层包ID加1处开始发送所有协议层包。注意上述NAK重传和超时重传需要保证当前正在发送的协议层包发送完毕之后，再去发送需要重传的协议层包。

链路层接收端对当前协议层包的判定流程如图6-5所示，先计算当前协议层包的CRC并和CRC校验位比较，注意计算区域为协议层包的ID/包头/负载，STP/CRC/END字段用0占位，若CRC校验正确，再去判断协议层包ID和期望ID是否相等。若相等，则上送协议层包给协议层，将期望ID加1，将NAK标记置为0，将ACK包放入发送队列并在适当间隔后交由链路层发送端发送；若CRC校验错误或者协议层包ID和期望ID不相等，则上报告警，然后判断NAK标记是否为1，若不为1，则触发重传，发送NAK包，将NAK标记置为1，之后丢弃当前协议层包，释放该协议层包占用的缓存；若NAK标记为1，则只丢弃当前协议层包，释放该协议层包占用的缓存。

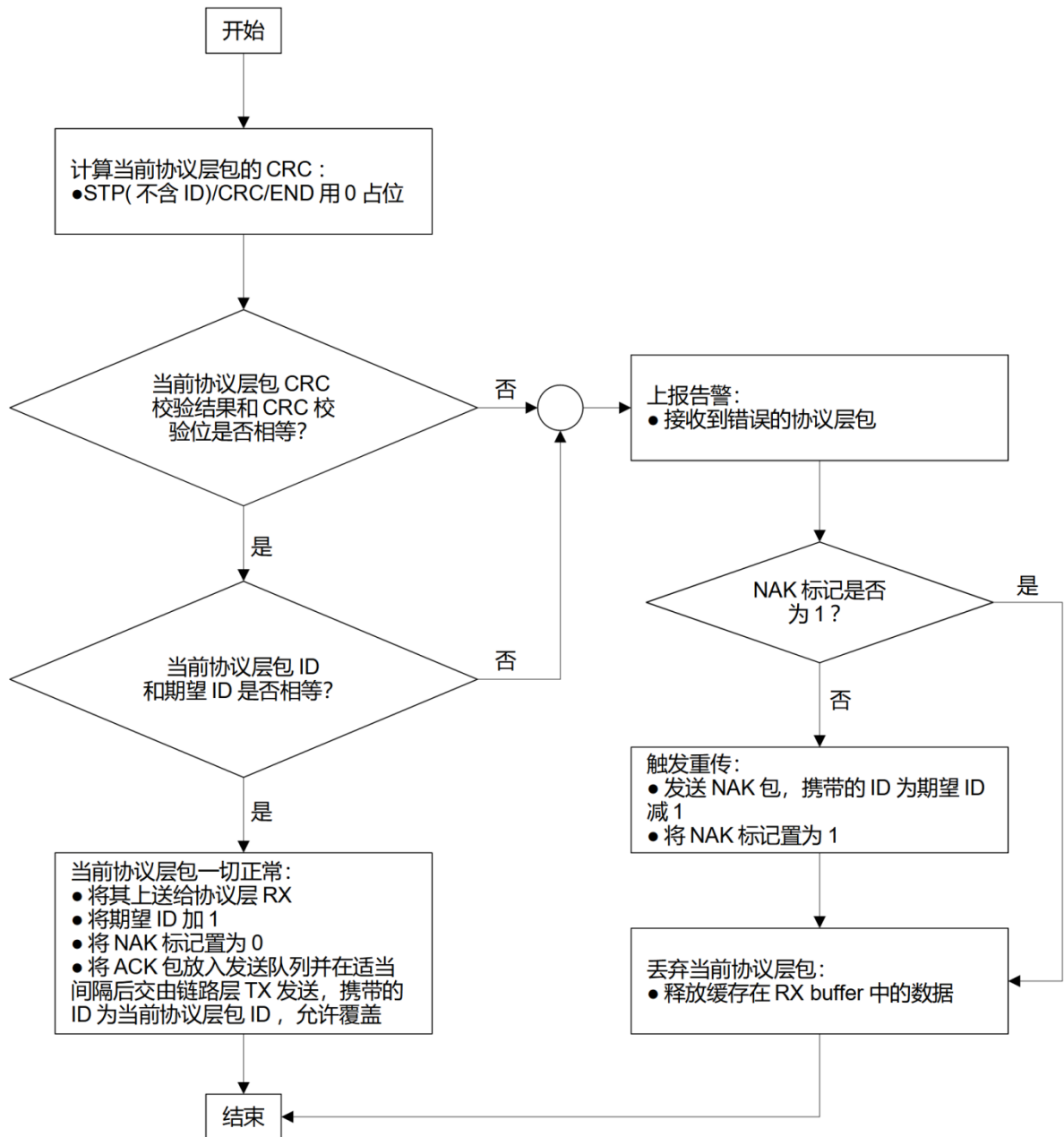


图6-5 链路层接收端触发重传判定流程

6.4.2 利用 NAK 包实现流控

当链路层接收端发现协议层接收端无法接收新的协议层包，并且自身缓存已满时，允许通过发送 NAK包的方式，让远端以协议层包为单位重传，直到协议层接收端畅通。

6.5 链路适配功能

链路适配功能用于将链路层组装好的协议层包/链路层包标记为数据/控制字符，并同COM字符/IDL字符三选一调度，调度1次得到8个数据/控制字符，再通过通道分发将8个字符分发到数字物理层的1/2/4/8个通道，接收端基于同样的原理将字符对齐，并还原为发送端发出时的排序之后，上送给链路层。

发送端支持如下功能：

- 将协议层/链路层包标记为数据字符并使用STP/SDP/END/PAD字符标记协议层/链路层包的头尾；
- 周期性发送COM字符；
- 空闲状态下发送IDL字符；

- 上述3种类别字符的调度，优先发送COM字符，其次发送协议层/链路层包编码之后的字符，再次发送IDL字符；

- 1/2/4/8通道分发。

接收端支持如下功能：

- 利用COM字符实现2/4/8通道去偏移；
- 1/2/4通道合并；
- 过滤控制字符，并将STP/SDP/END字符转换为协议层包/链路层包头尾信息。

6.5.1 协议层/链路层包头尾标记方法

协议层包头尾分别是 STP/END 字符，需要将其填充为 STP/END 字符的值，并将其所在的 128bit 码块标记为控制字符，其余部分标记为数据字符。规定 STP 字符只占用通道 0 的 bit[7:0]共 8bit，通道 0 的 bit[127:16]共 112bit 留给协议层，通道 0 的 bit[15:8]共 8bit 留给链路层使用，规定 END 字符只占用通道 7 的 bit[127:80]共 48bit，通道 7 的 bit[79:0]共 80bit 留给链路层使用，注意此处的通道 0/通道 7 为多通道分发前的通道编号。

链路层包头尾分别为 SDP/END 字符，需要将其填充为 SDP/END 字符的值，并将其所在的 128bit 码块标记为控制字符。规定 SDP 字符只占用通道 0 的 bit[63:0]共 64bit，通道 0 的 bit[127:64]共 64bit 留给链路层使用，规定 END 字符只占用通道 1 的 bit[63:0]共 64bit，通道 1 的 bit[127:64]填充 PAD 字符的 bit[127:64]。

6.5.2 控制字符

控制字符是指STP/SDP/END/PAD/COM/IDL字符，用于标记链路传输的数据流。

6.5.2.1 STP/SDP/END/PAD 字符

STP字符用于标记协议层包开始，位宽为16bit；SDP字符用于标记链路层包开始，位宽为64bit；END字符用于标记协议层包/链路层包结束，位宽为64bit，其中协议层包使用END字符的bit[63:16]；PAD字符用于将链路层包补足到8个字符的数据量。

STP字符的值为：{2{8'hFB}}；

SDP字符的值为：{8{8'h5C}}；

END字符的值为：{8{8'hFD}}；

PAD字符的值为：{16{8'h0}}；

6.5.2.2 COM 字符

COM字符位宽为128bit，发送周期可配，1/2/4/8通道模式下分别发送1/2/4/8个COM字符，使用IDL字符补满8个字符后，调度发送到链路上，以便每个通道都能获取到1个COM字符。

COM字符的值为：{15{8'hBC},8'h7D}；

6.5.2.2.1 同步数字物理层加解扰多项式

数字物理层使用COM字符同步加解扰多项式。

6.5.2.2.2 去偏移

链路层接收端工作在2/4/8通道模式时，需要利用COM字符实现去偏移，将COM字符对齐到同一拍。

6.5.2.2.3 多通道合并

链路层接收端工作在1/2/4通道模式时，需要利用COM字符实现多通道合并，将第1个COM字符挪到通道0，后续字符依次跟随挪动，从而还原为链路层发送端发出时的字符排列顺序。如图6-6所示，如果工作在2通道模式，发送端发送的COM字符位于通道0/1，接收端在接收到COM字符之后，需要将COM字符固定放置到通道0/1，不允许按照另外3种情况放置COM字符。

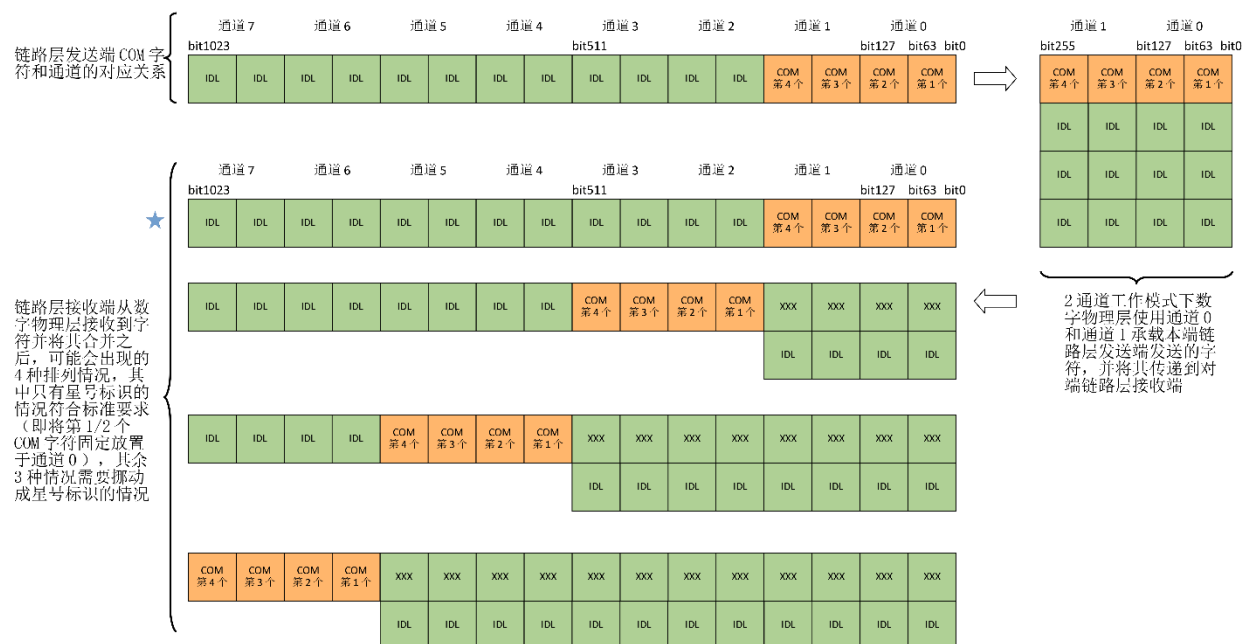


图6-6 多通道合并举例

6.5.2.3 IDL 字符

IDL字符位宽为128bit，用于填充链路空闲带宽，每调度1次IDL字符得到8个IDL字符，以便每个通道都能获取到1个IDL字符，需要注意IDL字符也需要加解扰，以确保链路在空闲时能维持在正常状态。

IDL字符的值为：{16{8'hDC}};

6.5.2.3.1 填充空闲带宽

IDL字符由链路适配发送端发出，用于填充链路空闲带宽，链路适配接收端将其丢弃。

6.5.3 多通道分发与合并

协议层包/链路层包、COM字符、IDL字符每调度1次获得的数据量都是8个字符，只允许在协议层包/链路层包结束之后，调度COM/IDL字符，从而保证链路上协议层包/链路层包的连续性。获取到8个字符之后，1/2/4通道工作模式需要将其分发8/4/2次给通道0/通道0~1/通道0~3，从而实现向下兼容1/2/4通道，8通道不需要做通道分发，每个通道发送1个字符即可。

通道合并是通道分发的逆过程，需要注意利用COM字符将字符序列还原为发送端发出时的字符的排列顺序。

6.5.4 链路层/物理层互联接口

链路层/物理层接口如表6-1所示：

表6-1 链路层/物理层互联接口

信号命名	信号功能描述
link2phy_valid[0:0]	发送侧，链路层向物理层指示 link2phy_data/link2phy_dk 上的数据有效
phy2link_rdy[0:0]	发送侧，物理层向链路层指示可以接收数据，使用 valid-rdy 握手机制传递数据
link2phy_data[1023:0]	发送侧，链路层向物理层传递的数据流，每个通道 128bit，共 8 个通道
link2phy_dk[7:0]	发送侧，链路层向物理层传递的数据流的控制字符/数据字符标记，每个通道 1bit，共 8 个通道 1'b0：控制字符 1'b1：数据字符
phy2link_valid[0:0]	接收侧，物理层向链路层指示 phy2link_data/phy2link_dk 上的数据有效
phy2link_data[1023:0]	接收侧，物理层向链路层传递的数据流，每个通道 128bit，共 8 个通道
phy2link_dk[7:0]	接收侧，物理层向链路层传递的数据流的控制字符/数据字符标记，每个通道 1bit，共 8 个通道 1'b0：控制字符 1'b1：数据字符

链路适配到物理层接口时序举例如图6-7所示：

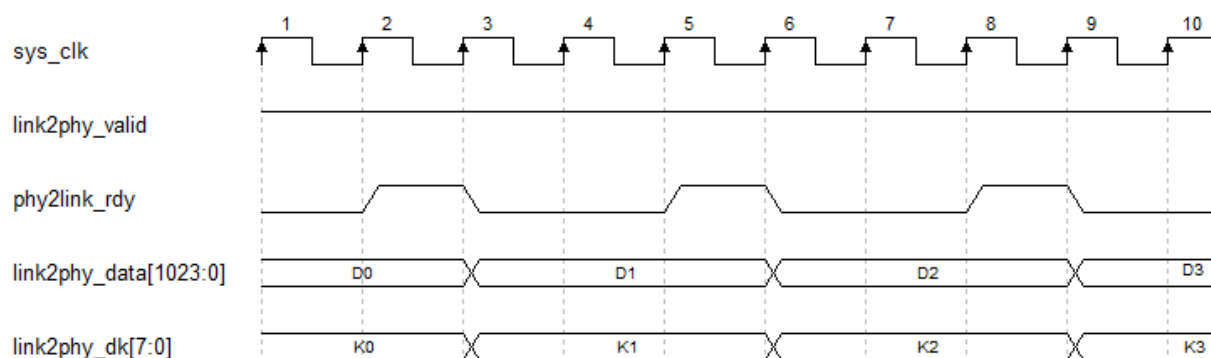


图6-7 链路适配到物理层接口时序

物理层到链路适配接口时序举例如图6-8所示：

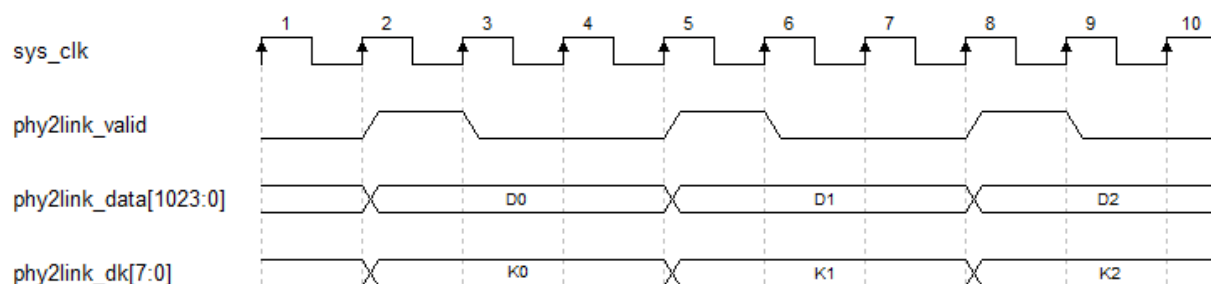


图6-8 物理层到链路适配接口时序

6.5.5 链路数据流举例

链路数据流可以是任意协议层包/链路层包/8个通道的IDL字符/8个通道的COM字符组合，若工作于兼容AXI4.0模式或原生模式，协议层包包长为128字节整数倍，最大长度是640字节。如图6-9中举例所示，协议层包A是640字节，协议层包B是256字节，协议层包C是128字节，协议层包D是384字节，协议层包E是512字节；若工作于兼容CXL3.0 Standard 256B Flit模式，协议层包包长固定为256字节。1个链路层包占用的带宽和1个包长为128字节的协议层包相等。COM字符发送周期可配，IDL字符用于填充链路空闲带宽。

[illegible]

图6-9 链路数据流举例（8通道模式）

7 链路训练状态机 LTSM（Link Training State Machine）

7.1 概述

LTSM主要用于D2D接口在建链过程中的状态控制，当发起建链后将进行状态的跳转，并在不同的状态下发出不同的控制信息，控制信息包括对链路层与物理层的控制，从而控制D2D接口整个建链与数据传输流程。

LTSM包括4个主要状态：空闲(Idle)、配置(Config)、训练(Traning)、正常通讯(Normal)。在配置状态下，近端D2D通过APB配置通道对其进行配置，同时近端D2D通过边带信号(Sideband signal)对远端D2D进行配置，配置内容包括需要使能的通道数、链路层与物理层的通道交叉配置、速率配置等等；训练状态下，近端与远端互相发送与检测NULL码；正常通讯状态在表示近端与远端可以通过D2D进行数据发送与接收。具体如下：

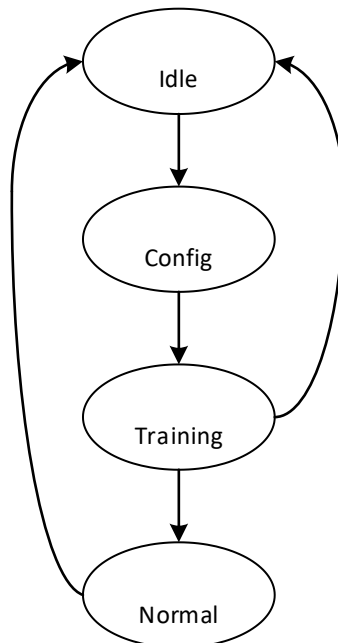


图 7-1 LTSM 状态示意图

7.2 建链流程

D2D 通过寄存器配置发起建链操作，本标准中将 D2D 接口的两端区分为近端和远端，主动发起建链的一端为近端，被动接收建链操作的一端为远端，具体连接关系如图 7-2 所示，远端与近端的数据交换，通过电气物理层 SerDes 的差分接口进行。寄存器参数的配置，则通过边带信号进行，并且只能由近端主动发起对远端的读写操作。

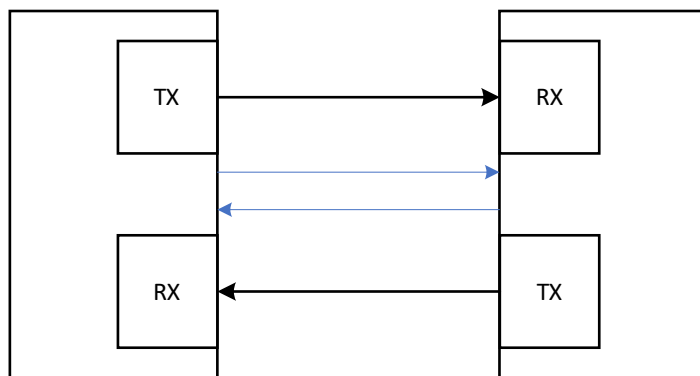


图 7-2 D2D 接口连接示意图

本标准的建链过程如下：

1. D2D 一端(近端)通过寄存器配置发起建链后，进入配置状态；
2. 近端通过边带信号对远端进行参数配置同步，完成配置后通过配置确定是否进行步骤 3，如果使能配置同步检查则进入步骤 3，否则进入步骤 4，状态机跳转至训练状态；
3. 当需要同步的寄存器配置（包括使能对应的通道以及 PLL）完成后，则进行配置确认，通过边带信号对远端寄存器进行读取并确认是否配置正确；
4. 近端发送端开始向远端发送 NULL 码（见章节 7.3），该状态下近端发送 NULL 码的数量可配（默认值为 2048 个 NULL 码）；
5. 远端接收端连续正确接收到足够数量（数量可配，默认值为 16）的 NULL 码后，则远端发送端开始向近端接收端发送 NULL 码（发送数量可配，默认值为 1024）；
6. 当近端接收端连续正确接收到足够数量（数量可配，默认值为 16）的 NULL 码且发送了足够数量（步骤 4 对应的数量）的 NULL 码后，则表示链路训练完成，结束训练状态，进入正常通讯状态，可以正常传数据。

流程如图 7-3 所示：

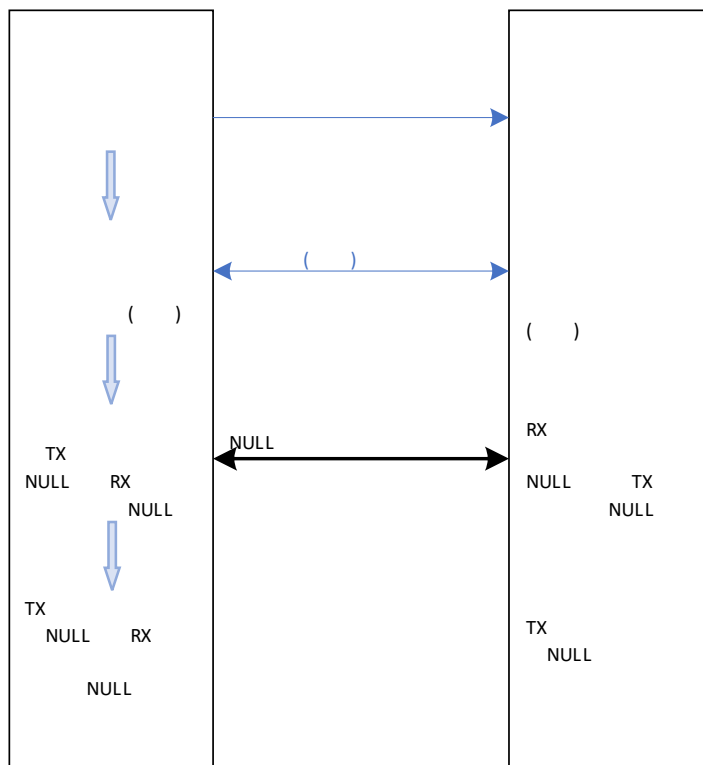


图 7-3 建链流程

7.3 NULL 码型

在训练状态下同时向所有使能的通道发送 NULL 码，具体组成为 1COM + 7*IDL，如图 7-4 所示：

COM	COM	COM	COM	COM	COM	COM	COM
IDL	IDL	IDL	IDL	IDL	IDL	IDL	IDL
IDL	IDL	IDL	IDL	IDL	IDL	IDL	IDL
IDL	IDL	IDL	IDL	IDL	IDL	IDL	IDL
IDL	IDL	IDL	IDL	IDL	IDL	IDL	IDL
IDL	IDL	IDL	IDL	IDL	IDL	IDL	IDL
IDL	IDL	IDL	IDL	IDL	IDL	IDL	IDL
IDL	IDL	IDL	IDL	IDL	IDL	IDL	IDL

图 7-4 Null 码示意图

8 数字物理层 DPL（Digital Physical Layer）

8.1 数字物理层功能概述

本标准定义了通过数字物理层发送和接收物理层传输的逻辑数据传输流。物理层分为数字物理层和电气物理层。数字物理层将链路层与 SerDes 结构的电气物理层衔接起来，并进行数据通路的编码隔离。

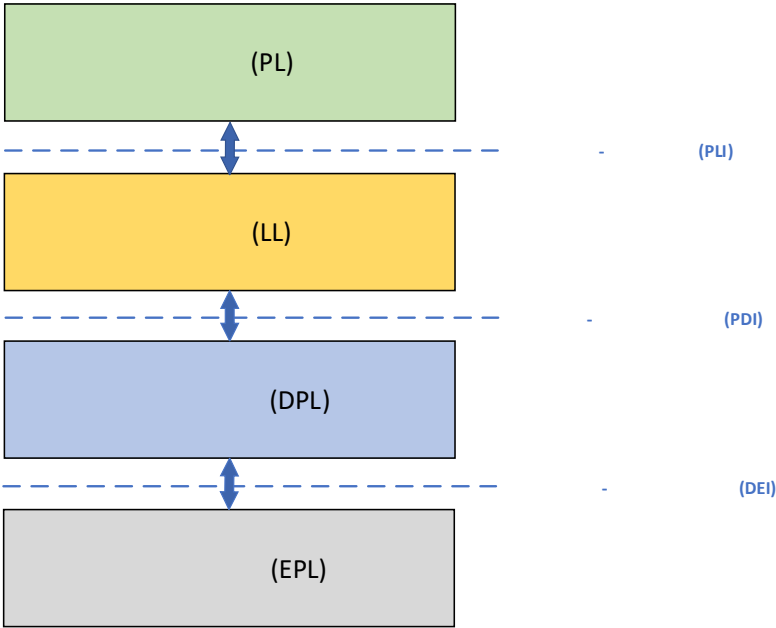


图8-1 物理层结构框图

数字物理层最多接收发送端链路层的1024bit数据。每一条通道内部结构相同，可以通过链路模式寄存器配置为不同的工作模式，如表8-1所示：

表8-1 物理层工作模式表

链路模式寄存器	工作模式
00	单通道模式
01	双通道模式
10	四通道模式
11	八通道模式

物理层最多允许8条通道同时工作，通道的使能由上层控制。

包括以下功能：

- 链路全交叉；
- 通道的极性反转；
- 链路的加解扰和编解码；
- 支持可信度的数据对齐模块（Block Alignment）；
- PRBS/FIX的生成和检测；
- 每条通道数据位宽为128bit。

性能指标要求：

表 8-2 数字物理层性能指标

时钟域	低速频率	中速频率	高速频率
数字物理层工作时钟	$\geq 300\text{MHz}$	$\geq 600\text{MHz}$	$\geq 1200\text{MHz}$
电气物理层读写时钟	250MHz	500MHz	1000MHz
每通道传输速率	32GHz	64GHz	128GHz
总传输速率	32/64/128/256GHz	64/128/256/512GHz	128/256/512/1024GHz

发送端数字物理层中的每条通道由加扰器、128b/130b编码器构成。接收端通道中由解扰器、128b/130b解码器以及对应数据对齐模块等模块构成。其整体传输流如图8-2所示：

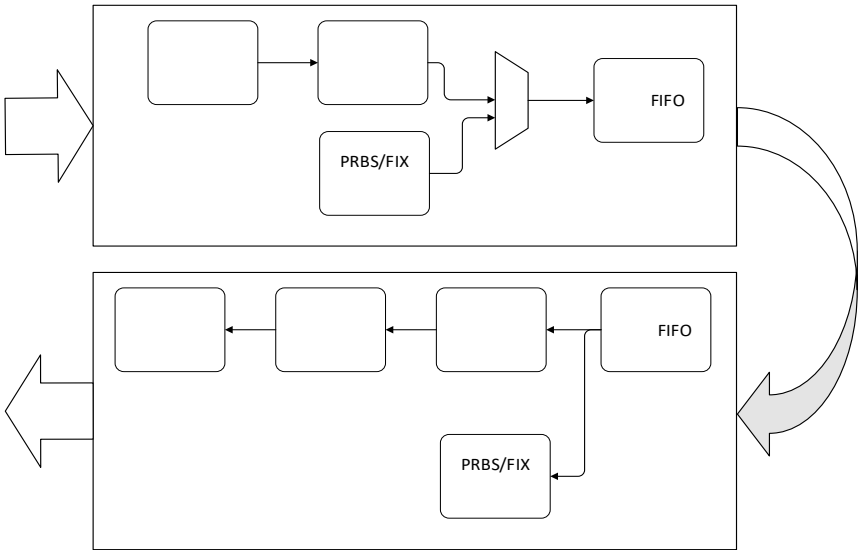


图8-2 数字物理层数据流框图

链路层发来的数据经过扰码器加扰后，送给128b/130b编码器。编码后的数据需要经过一个异步先进先出存储器（AsyncFIFO）来进行跨时钟域处理。之后即进入电气物理层中串行化。

在测试模式或者建链状态下，提供一个PRBS生成器，产生伪随机码流。

接收端通过数据对齐模块（Alignment）识别对比COM码来进行数据的定帧，再送给128b/130b解码器。解码后的数据进行128bit解扰处理，并进行基于编码规则的错误检测。

注意:接收端由于电气物理层在开启工作后会一直向数字物理层传输数据，因此接收端的数字物理层不能反压电气物理层的数据。

8.2 加扰和随机码生成

当数据重复传输时，能量就会集中在某一频率上，产生电磁干扰噪声。因此需要将集中的能量分散开来，让其几乎变成白噪声。并且一定程度上，更有利于时钟的提取，同时起到加密的效果。

对数据加扰，是一种将数据重新排列，或者进行编码，以使其随机化的方法，且能解扰恢复。目的是打乱长的连“0”和长的连“1”序列，将数据随机化。定义了一个线性反馈移位寄存器(LFSR)，用于加扰和产生随机码流的生成。目前选用的扰码器采用128bit位宽，多项式为：

$$G(x) = x^{23} + x^{21} + x^{16} + x^8 + x^5 + x^2 + 1$$

每条通道的种子是不同的，同一条通道接收端和发送端的种子必须相同。默认情况下的种子值如表8-3所示：

表8-3 加扰种子表

通道序列	0	1	2	3	4	5	6	7
种子值（h）	1DBFBC	0607BB	1EC760	18C0DB	010F12	19CFC9	0277CE	1BB807

不同的种子决定了线性反馈移位寄存器的周期性序列。若是链路进行了交叉，则内部需要根据交叉后的值进行种子的重新映射。

线性反馈移位寄存器实现：

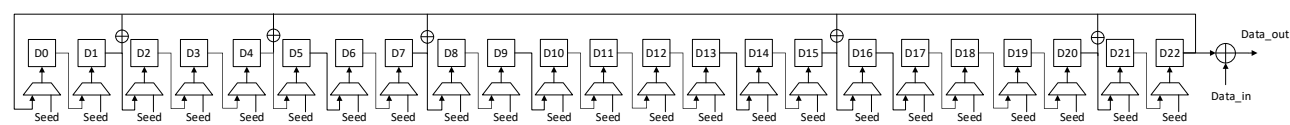


图8-3 加扰原理图

加扰规则：

- 数据和控制字符在正常的传输过程中需要进行加扰。
- 测试模式下支持不加扰。
- COM字符不加扰，且接收或发送COM字符时，线性反馈移位寄存器当前值重置为初始种子。

8.3 编解码

本标准采用128/130b编码，将128bit数据或控制字符编码为130bit块传输，bit[129:128]表示同步头，主要用于接收端的数据对齐。同步头有“01”和“10”两种，“01”表示bit[127:0]是数据字符，表示为D码，“10”表示bit[127:0]是控制字符，表示为K码。

K码可以根据协议层定义混合K码和D码的方式。需要注意的是，COM码作为特殊的控制码，用于做定帧和同步，因此不包含数据，也不能被加扰。128b/130b编码格式如图8-4所示：

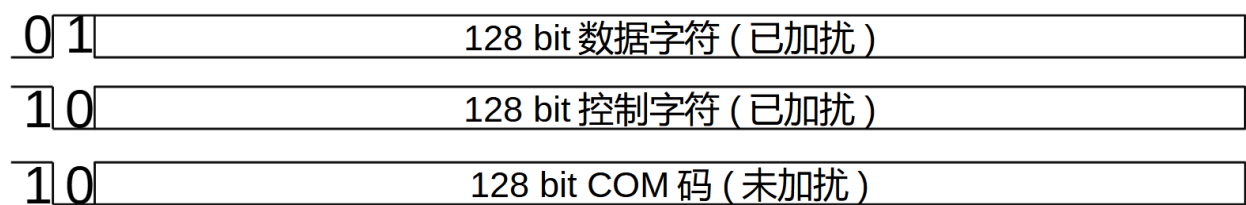


图8-4 128b/130b编码格式

接收到同步头为“11”或“00”的数据块是同步错误情况，并将此错误上报。

8.4 数据对齐模块（Alignment）

由于SerDes的发送端和接收端的启动时间不同，并且在发送数据的过程中，容易出现数据偏移，因此需要数据对齐模块进行数据同步，使得数字物理层能够准确找到字符块的起始位置。

链路层 发送端会周期性地发送COM码作为同步信息。数据对齐模块的目标是检测到COM码，并将数据对齐发送。因此需要在内部缓存一拍数据，并和下一拍的数据拼接成一个260bit的缓存器（Buffer）进行检测。检测到COM码时，将输出数据重新同步到正确的位置，并输出当前位置为位域坐标（Label），如图8-5所示：

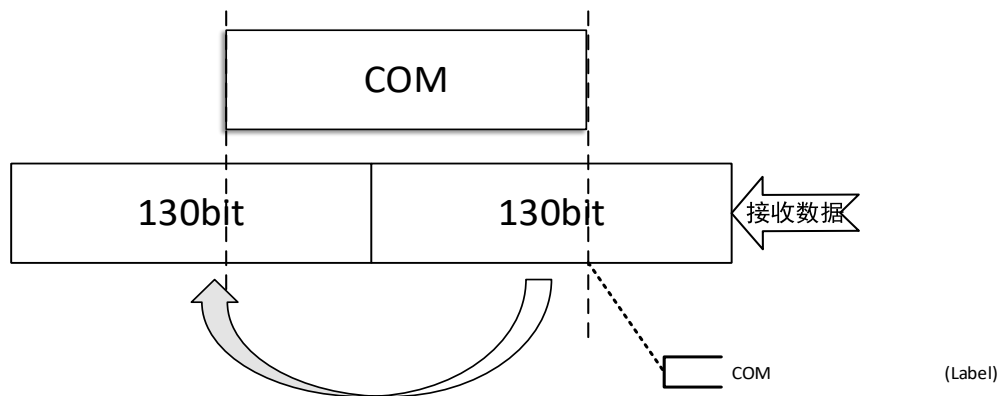


图8-5 数据对齐模块识别COM码示意图

8.4.1 可信度计数器

由于数据对齐模块功能是依靠识别COM码的固定码方式，有可能会存在着两个数据拼接中恰好和COM码相同的情况。因此在这里增加一个可信度计数器（credible_cnt）：每当收到一个COM码时，位域坐标没有发生改变，则可信度计数器加一。若收到COM码时，位域坐标发生了改变，则可信度计数器减一，若可信度计数器等于0，则可以更新最新的位域坐标值。如图8-6所示：

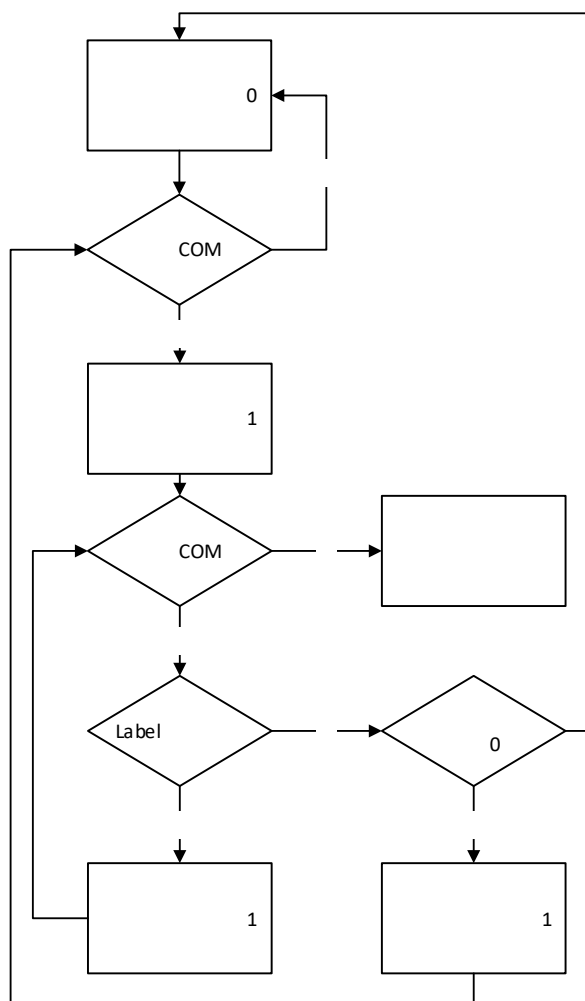


图8-6 可信度计数器流程图

最大可信度 (credible_max) 作为可配置使用，且最大可信度建议不要配置为0。

若是出现了由于时钟偏差而导致的累积错误，即链路可能会在运转一段时间后多出一个或少一个 bit，那么此时的位域坐标应该需要改变，才能够正确接收到数据。此时可信度计数器会在每接收一个 COM 后减一，直到减到0，位域坐标才会改变。因此最大可信度会影响出现这个问题后的恢复时间：最大可信度配置的越大，则需要收到COM码的数量越多。

数据对齐模块收到一个COM码，会上报已对齐(Align_done)信号。

如果位域坐标发生改变，会上报对齐位域改变(Align_change)信号。

在已对齐信号之前，所有传输的数据都是无效数据。

8.5 链路交叉

D2D在封装打线的时候，不一定会按照顺序连接。因此需要支持链路全交叉，即可以通过链路选择寄存器和链路模式寄存器配置通道的数量和物理交叉。

链路选择寄存器只作用在发送端，用于控制发送端的每一条通道应该取哪一个发送端的链路层通道的数据。这里默认定义链路层的数据是从小到大依次进行分发。即如果是配置的链路模式寄存器为2，则表示有四条通道进行工作，那么链路层下发的数据即是0号、1号、2号、3号数据。如图8-7所示：

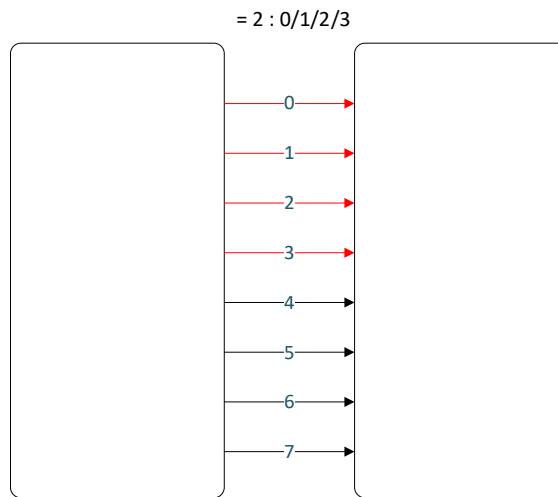


图8-7 发送端链路层到数字物理层数据有效通路

接收端会根据电气物理层输出的能量检测(signal_detect)信号来判断，是哪条通道上获取到了数据。这里默认定义了，接收端的数据会依通道序号的排列顺序，从小到大依次将接收端的数字物理层通道的数据发送给接收端的链路层。如果是配置的链路模式寄存器为2，则表示有四条通道进行工作，那么数字物理层接收的数据需要送给链路层的0号、1号、2号、3号数据通道。如图8-8所示：

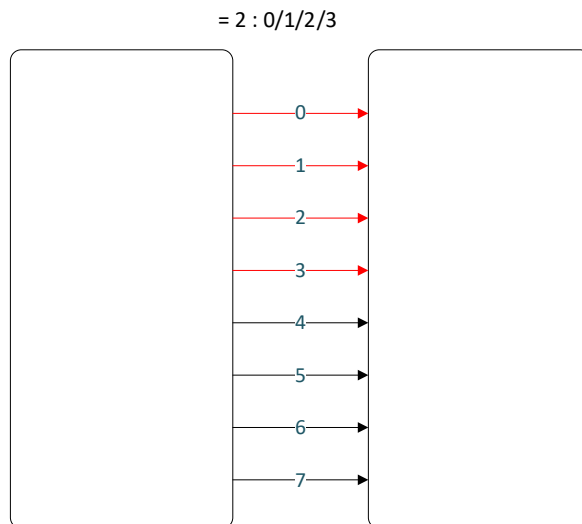


图8-8 接收端数字物理层到链路层数据有效通路

例如：图8-9为在链路模式寄存器等于2，即开启四条通道的示意图。此时物理上的连线如下：

- 发送端的电气物理层的0号通道和接收端的电气物理层的4号通道相连；
- 发送端的电气物理层的2号通道和接收端的电气物理层的7号通道相连；
- 发送端的电气物理层的3号通道和接收端的电气物理层的3号通道相连；
- 发送端的电气物理层的7号通道和接收端的电气物理层的1号通道相连。

由于接收端的数字物理层会将激活的通道重新排序，从小到大依次推入接收端的链路层。那么可以判断出来，发送端的数字物理层需要做如下操作才能够满足数据链路的正常通路：

- 发送端的数字物理层的0号通道和发送端的电气物理层的7号通道相连；
- 发送端的数字物理层的1号通道和发送端的电气物理层的3号通道相连；
- 发送端的数字物理层的2号通道和发送端的电气物理层的0号通道相连；
- 发送端的数字物理层的3号通道和发送端的电气物理层的2号通道相连。

因此需要根据得到的连接关系，配置发送端的数字物理层的各个通道的数据来源。如图8-9表示在上述连接关系中，数字物理层到电气物理层的链路交叉示例。

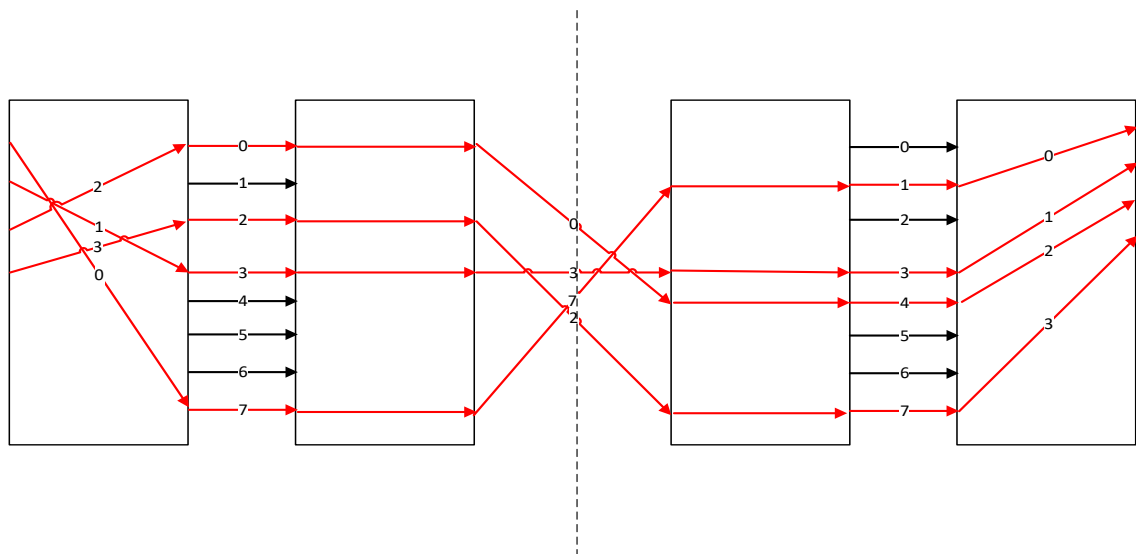


图8-9 链路交叉示例

全交叉功能亦可用于链路维修。若在使用的1/2/4条通道的情况下，其中一条通道出现物理性损坏，可以通过全交叉功能，将未被使用的通道利用起来。

注：交叉功能改变后，必须重新建链才可使用。

8.6 通道极性反转

设备需要支持通道的极性反转，以应对复杂多变的D2D封装情况。本标准定义了主设备（Master）和从设备（Slave），因此可以通过配置主设备的发送端和接收端进行单边的极性反转即可，如图8-10所示：

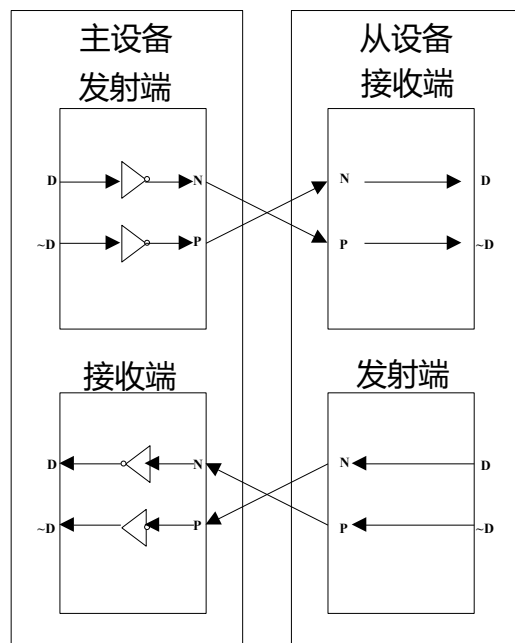


图8-10 链路极性反转示意图

9 电气物理层（Electrical Physical Layer）

电气规格的关键属性包括:

- 支持单通道 128/64/32GT/s 的数据率;
- 差分数据传输;
- 直流或者交流耦合的 Die-to-Die 互联;
- 数据中内嵌时钟,不需要单独的通道传送时钟信号。

9.1 数据率

最高支持 128GT/s 的数据传输，并向下兼容 64/32GT/s 的数据传输。

9.2 电学参数

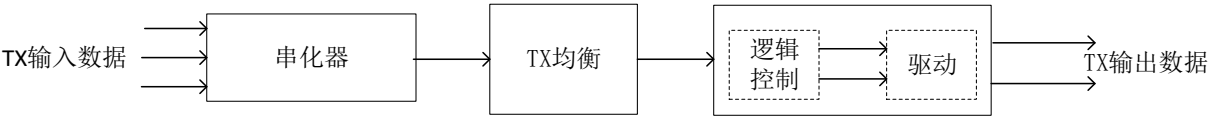
PHY 的典型电学参数如表 9-1 所示

表 9-1 PHY 的电学参数

	标准封装		
数据率 (GT/s)	32	64	128
能效 (pJ/b)	2.5	2.5	2.5
延迟时间 (TX+RX) (ns)	6	6	6
低功耗能量 (% 峰值能量)	10	10	10
通道长度 (mm)	1~50	1~50	1~50
带宽线密度 (GB/s/mm)	32	64	128
带宽面密度 (GB/mm^2)	40	80	160
PHY 宽度(um)	800	800	800
PHY 长度(um)	2000	2000	2000
静电防护 (充电设备模型)	100V	100V	100V

9.3 发送端

发送端的框架如图 9-1 所示，发送端包含串化器，TX 均衡，驱动电路等子模块。



9-1 发送器架构框图

串化器，将并行输入信号转化为串行输出信号，发送给接收端。

TX 均衡是前馈均衡器。由于通道良好，它的功能是可选的，只在 128GT/s 的应用中需要，64GT/s 和 32GT/s 的应用中则不需要。

9.3.1 发送端的电学参数

发送端的电学参数如表9-2所示。

表 9-2 发送端的电学参数

参数名称	最小值	典型值	最大值	单位	备注
发送信号摆幅		1		Vpp	NRZ

信号摆幅修调的范围	340		500	mV	
信号摆幅修调的步长		11		mV	
PAM-4 信令 00 对应的电平		0		V	PAM-4
PAM-4 信令 01 对应的电平		0.33		V	
PAM-4 信令 10 对应的电平		0.66		V	
PAM-4 信令 11 对应的电平		1		V	
差分输出阻抗	80	100	120	Ohm	
前坐标均衡范围			-9.6	dB	
后坐标均衡范围			-9.6	dB	
通路间的歪斜			0.1	UI	
时钟频率		16			GHz
输出总的抖动		6		ps	32GT/s
		6		ps	64GT/s
		3		ps	128GT/s

9.3.2 驱动电路

如图 9-2 所示，驱动电路由驱动逻辑电路，输出 NR 编码或者 PAM4 编码电路和 T-coil 组成：

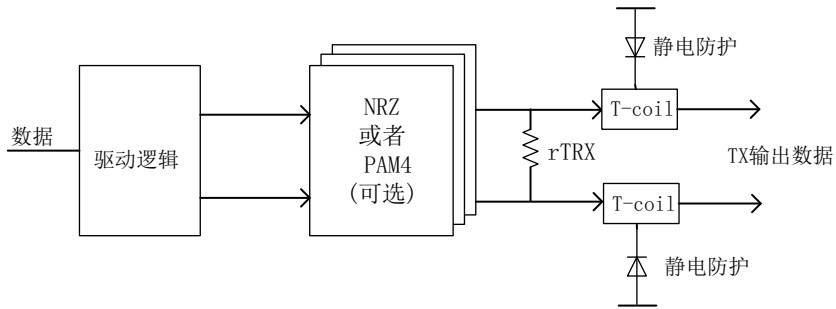


图 9-2 发送端驱动的框图

驱动电路针对 NRZ 和 PAM-4 两种编码方式进行了电路结构的优化，通过寄存器配置(OPTION)就可以切换编码输出方式，满足不同应用的需求。128/64GT/s 的应用采用 PAM-4 结构，而 32GT/s 的应用采用 NRZ 结构。

驱动电路支持减小输出信号摆幅；根据不同的通道特性，改变输出电阻网络的寄存器配置，可以实现对输出信号摆幅的控制。

9.3.3 发送端的前馈均衡

为了缓解接收端均衡的压力，在发送端使用 FFE(Feed Forward Equalizer)技术对信号进行预均衡。FFE 基于有限冲激响应滤波器设计实现，将延时的信号按照不同的权重 (C_{-1} , C_0 , ..., C_n) 相加。控制权重的大小即可调整均衡强度。实质是使用数字线性高通滤波器提高信号的高频分量，实现对信道的补偿，减小 ISI(Inter-Symbol Interference)效应的影响。

发射端的前馈均衡，基于 FIR 滤波器设计，2 阶 3TAP 的结构，电路框图如图 9-3 所示：

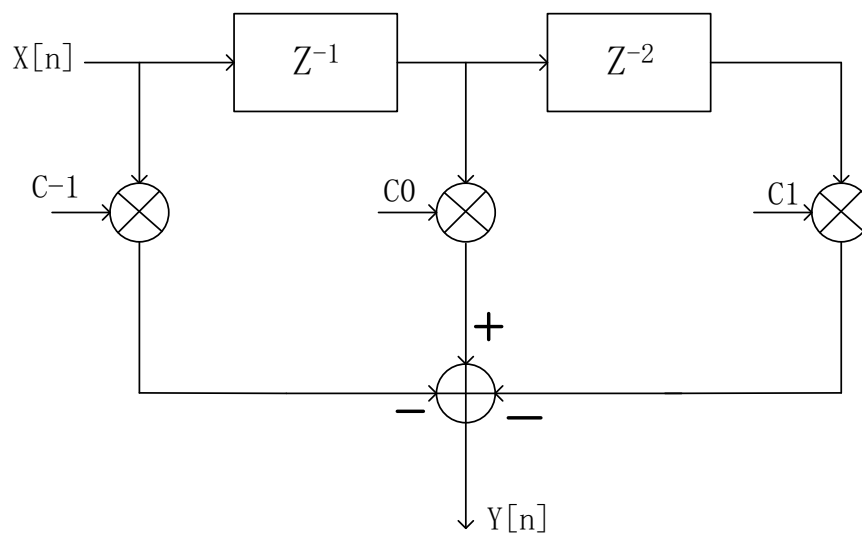


图 9-3 前馈均衡器的框图

$$Y[n] = C_0 * X[n - 1] + C_1 * X[n] + C_2 * X[n + 1]$$

$$|C_0| + |C_1| + |C_2| = 1$$

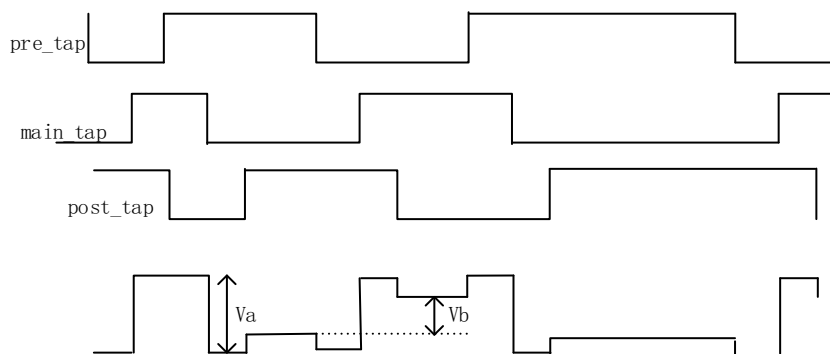


图 9-4 发送端均衡的原理图

$$\text{Pre_De_emphasis} = 20\log \left(\frac{C_0 - C_{-1} - C_1}{C_0 + C_{-1} - C_1} \right)$$

$$\text{Post_De_emphasis} = 20\log \left(\frac{C_0 - C_{-1} - C_1}{C_0 - C_{-1} + C_1} \right)$$

注 1: C_{-1} , C_0 , C_1 是发送端 pre-tap, main-tap, post-tap 的系数;

注 2: C_{-1} 和 C_1 表示的系数是负的;

注 3: 均衡的效果由 C_{-1} , C_0 , C_1 的比例决定。

9.4 接收端

接收端的框图如图 9-5 所示, 接收端包含端接电阻、采样器、串并转换、解串器、时钟数据恢复等子电路模块。

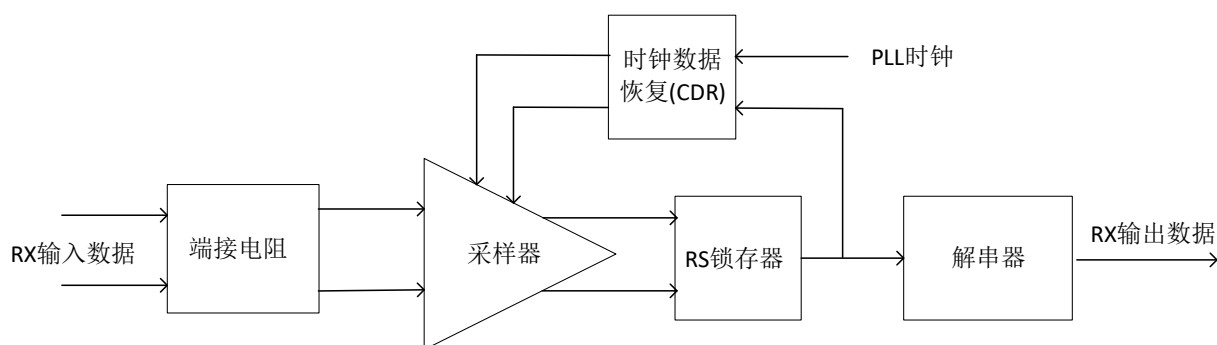


图 9-5 接收端的框图

由于通道的良好特性，连续时间线性均衡器（CTLE）的功能是可选的，只在 128GT/s 的应用中需要，64GT/s 和 32GT/s 的应用中则不需要。

时钟恢复电路主要从串行输入的信号中恢复出时钟信号。

解串器是将串型输入数据转为并行输出数据。

9.4.1 接收端的电学参数

接收端的电学参数如表 9-3 所示。

表 9-3 接收端的电学参数

参数名称	最小值	典型值	最大值	单位	注释
端接电阻的阻抗	80	100	120	Ohm	
端接电阻修调的范围	72		175	Ohm	
端接电阻修调的步长		3		Ohm	
CTLE 提升增益范围	11		17.8	dB	128GT/s
时钟恢复的频率		16		GHz	
接收端眼图的眼高	400			mV	NRZ
	133			mV	PAM-4
接收端眼图的眼宽	0.5			UI	NRZ
	0.17			UI	PAM-4

9.4.2 连续时间线性均衡器（CTLE）

对于传输 32/64GT/s 的数据率时，当合适的 TX 均衡设置后，在最坏的通道和封装模型下，传输至接收端的眼图仍然满足要求，所以在 32/64GT/s 中不需要在接收端进行均衡。

但是传输 128GT/s 的数据时，传输至接收端的眼图已不能满足要求，所以需要在接收端进行均衡，采用的均衡是一级的 CTLE。

CTLE 的原理图如图 9-6 所示：

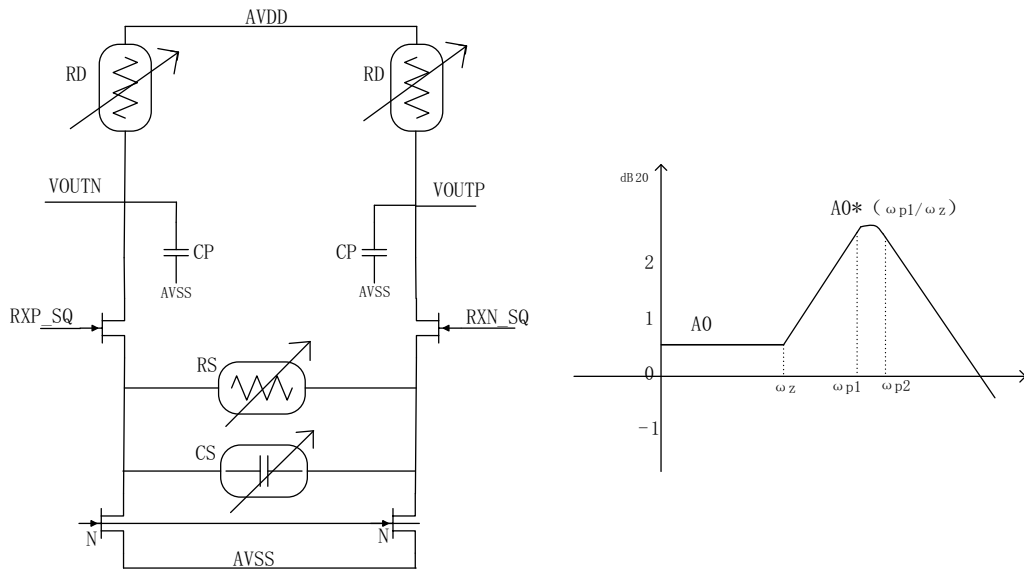


图 9-6 CTLE 电路原理图

CTLE 原理:

$$H(s) = \frac{G_m}{C_p} * \frac{S + \frac{1}{R_s * C_s}}{\left(S + \frac{1 + 0.5 * G_m * R_s}{R_s * C_s}\right) * \left(S + \frac{1}{R_D + C_p}\right)}$$

$$\text{直流增益: DC gain} = \frac{G_m * R_D}{1 + 0.5 * G_m * R_s}$$

$$\text{顶点增益: peak gain} = G_m * R_D$$

CTLE 的工作原理是直接通过线性模拟高通滤波器拟合信道的衰减，实现信道的补偿。

信道的频率响应等效为一个低通滤波器，而 CTLE 通过减小低频信号的方式补偿高低频的衰减差，从而补偿信道的衰减。

9.4.3 时钟数据恢复电路

时钟数据恢复电路主要完成两个工作，一个是时钟恢复，一个是数据重定时。即将嵌入在数据中的时钟信息提取出来，并通过恢复的时钟将接收到的数据重定时，采样含噪声的数据，从而消除数据在传输过程中引入的抖动，其性能决定了整个串行传输系统的性能。时钟恢复后的时钟采样时序如图 9-7 所示:

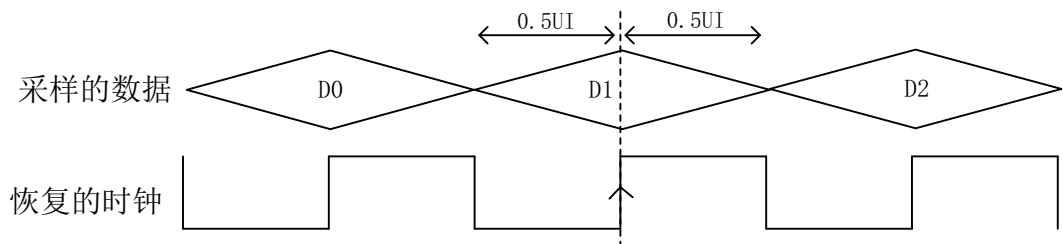


图 9-7 时钟恢复后的时钟采样图

10 边带传输 Sideband (Optional)

本标准中边带信号用于近端对远端读写寄存器。

边带信号包含2个引脚，一个是双向的串行时钟线，只由近端发送给远端，另一个是双向的串行数据线，当近端给远端写数据时，信号由近端发送给远端，当远端给近端返回数据时，信号由远端发送给近端。

边带信号串行时钟速度为100MHz，接收端根据时钟上升沿对数据线进行采样，先传输最高位，传输信号示意如图10-1所示：

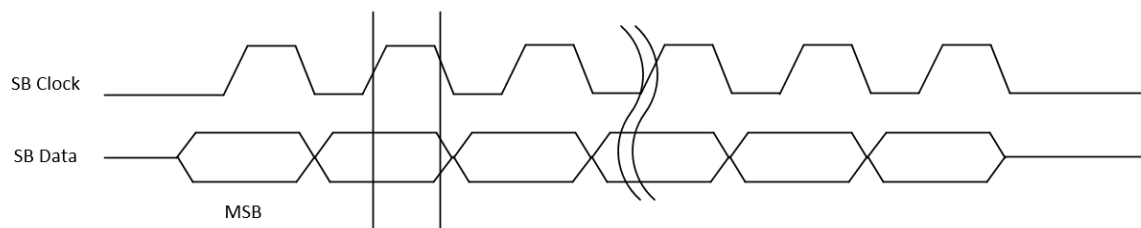


图10-1 边带信号

近端将对远端的寄存器访问打包并发送，远端接收到边带传输的协议包后解析并做出响应，如果是写寄存器，则立即执行，如果是读寄存器，则需要8个空拍才可回复寄存器值。

写寄存器包包长为64bit，包格式如图10-2所示：

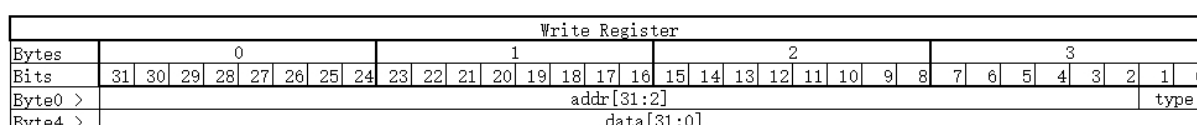


图10-2 写寄存器格式

写寄存器格式描述如表10-1所示：

表10-1 写寄存器格式描述

位域	描述
addr[31:2]	写寄存器地址的[31:2]，addr[1:0]固定为 0，由近端发送给远端。
type[1:0]	对于写寄存器来说，该值固定为 2'b11，由近端发送给远端
data[31:0]	写寄存器数据[31:0]，由近端发送给远端

读寄存器包包长为72bit，包格式如图10-3所示：

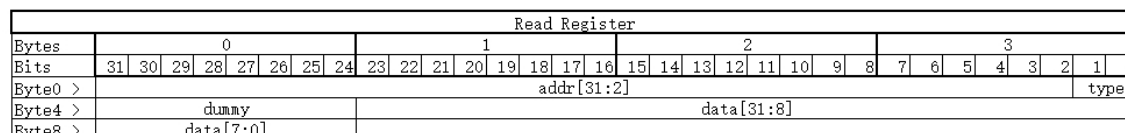


图10-3 读寄存器格式

读寄存器格式描述如表10-2所示：

表10-2 读寄存器格式描述

位域	描述
addr[31:2]	读寄存器地址的[31:2]，addr[1:0]固定为 2'b00，由近端发送给远端
type[1:0]	对于读寄存器来说，该值固定为 2'b00，由近端发送给远端
dummy[7:0]	空闲拍，近端只发送 8 拍时钟并释放数据线
data[31:0]	读寄存器数据[31:0]，由远端发送给近端

11 接口定义

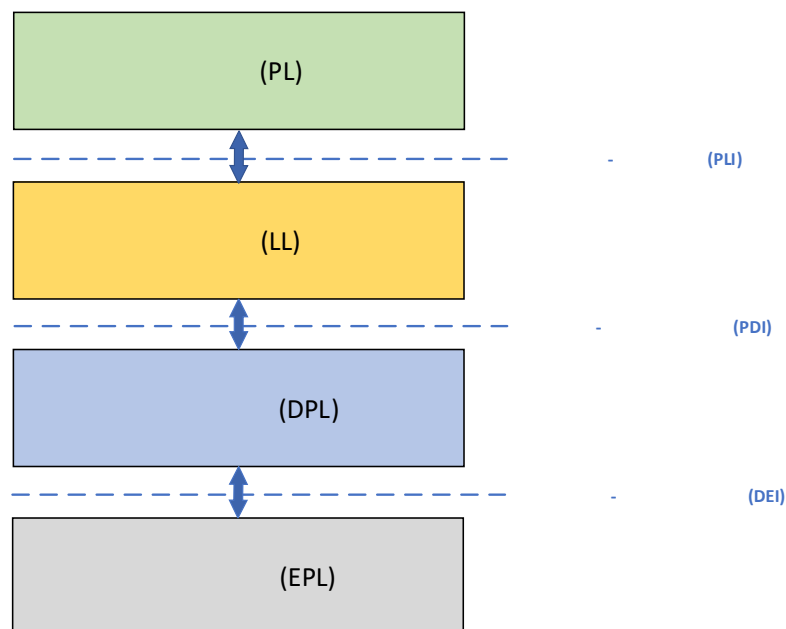


图 11-1 接口概述

如图 11-1 所示，本标准中将 D2D 接口划分为 4 个层次，分别为：

- 协议层（Protocol Layer，缩写为 PL）；
- 链路层（Link Layer，缩写为 LL）；
- 数字物理层（Digital Physic Layer，缩写为 DPL）；
- 电气物理层（Electrical Physic Layer，缩写为 EPL）。

在本标准中对于 D2D 的接口定义了如上 4 个层次之间的接口，分别为：

- 协议层与链路层之间的接口，即 PL-LL Interface（PLI）；
- 链路层与数字物理层之间的接口，即 LL-DPL Interface（LDI）；
- 数字物理层与电气物理层之间的接口，即 DPL-EPL Interface（DEI）；
- D2D 近端与远端的接口，通过近端与远端的 EPL 相连，即 EPL-EPL Interface（EEI）。

11.1 协议层与链路层接口信号定义

PLI 为协议层与链路层之间的接口。对于发送方向，协议层将来自上层系统的数据流与控制流打包，然后将其传输到链路层；而接收方向，链路层通过 CRC 检测，剥离链路层包等操作，将其中的协议层数据向上传输至协议层。具体接口如表 11-1 所示：

表 11-1 协议层与链路层接口

接口名	接口描述
prot2link_vld[0:0]	发送侧，协议层向链路层指示 prot2link_data/prot2link_tail 上的数据有效
link2prot_rdy[0:0]	发送侧，链路层向协议层指示可以接收数据，使用 valid-rdy 握手机制传递数据
prot2link_data[1023:0]	发送侧，协议层向链路层传递的协议层包的其中 1 拍数据
prot2link_tail[0:0]	发送侧，协议层向链路层传递的协议层包的尾标记
link2prot_vld[0:0]	接收侧，链路层向协议层指示 link2prot_data/link2prot_tail 上的数据有效
prot2link_rdy[0:0]	接收侧，协议层向链路层指示可以接收数据，使用 valid-rdy 握手机制传递数据
link2prot_data[1023:0]	接收侧，链路层向协议层传递的协议层包的其中 1 拍数据

link2prot_tail[0:0]	接收侧，链路层向协议层传递的协议层包的尾标记
---------------------	------------------------

11.2 链路层与数字物理层接口信号定义

本节定义了链路层与数字物理层之间的接口，在发送方向，链路层在接收到协议层发送的数据之后，对其进行添加 STP，END 等字符，进行 ID 编号，CRC 校验，插入链路层包等操作后将数据按照多个通道的方式发送到数字物理层，在接收方向上，数字物理层将电气物理层发送的数据进行定帧，解码，解扰等操作后按照多通道的方式将数据发送到链路层，具体接口定义如表 11-2 所示：

表 11-2 链路层与数字物理层接口

接口名	接口描述
link2phy_valid[0:0]	发送侧，链路层向物理层指示 link2phy_data/link2phy_dk 上的数据有效
phy2link_rdy[0:0]	发送侧，物理层向链路层指示可以接收数据，使用 valid-rdy 握手机制传递数据
link2phy_data[1023:0]	发送侧，链路层向物理层传递的数据流，每个通道 128bit，共 8 个通道
link2phy_dk[7:0]	发送侧，链路层向物理层传递的数据流的控制字符/数据字符标记，每个通道 1bit，共 8 个通道 1'b0：控制字符 1'b1：数据字符
phy2link_valid[0:0]	接收侧，物理层向链路层指示 phy2link_data/phy2link_dk 上的数据有效
phy2link_data[1023:0]	接收侧，物理层向链路层传递的数据流，每个通道 128bit，共 8 个通道
phy2link_dk[7:0]	接收侧，物理层向链路层传递的数据流的控制字符/数据字符标记，每个通道 1bit，共 8 个通道 1'b0：控制字符 1'b1：数据字符

11.3 数字物理层与电气物理层接口信号定义

本节定义了数字物理层与电气物理层之间的接口，在发送方向，数字物理层在接收到链路层发送到数字物理层各个通道上的数据进行加扰编码等操作后，按照一定的方式发送电气物理层，而接收方向上，电气物理层将对端发送的数据经过处理后按照一定的方式发送数字物理层，具体接口描述如表 11-3 所示：

表 11-3 数字物理层与电气物理层接口

接口名	接口描述
epl2dpl_tx_clk[7:0]	电气物理层发送数字物理层的发送时钟，数字物理层需要将数据同步到该时钟下发送到电气物理层
dpl2epl_tx_dat[127:0][7:0]	数字物理层发送到电气物理层的数据接口，分为 8 个通道，每个通道 128bit
epl2dpl_rx_clk[7:0]	电气物理层发送数字物理层的接收时钟，电气物理层需要通过该时钟对数据进行接收
dpl2dpl_rx_dat[127:0][7:0]	电气物理层发送到数字物理层的数据接口，分为 8 个通道，每个通道 128bit

11.4 近端与远端接口信号定义

本节定义 D2D 近端与远端的接口，即近端电气物理层与远端电气物理层的接口，这部分接口分为数据接口与边带传输接口，而边带传输接口为可选项（如第十章所述），对于数据接口，发送与接收方向

均为高速差分串行接口；边带传输信号（sideband）为 sb_clock 与 sb_data。具体接口描述如表 11-4 所示：

表 11-4 电气物理层与电气物理层接口

接口名	接口描述
tx_dat_p[7:0]	发送端串行数据线 p 分量
tx_dat_n[7:0]	发送端串行数据线 n 分量
rx_dat_p[7:0]	接收端串行数据线 p 分量
rx_dat_n[7:0]	接收端串行数据线 n 分量
sb_clock	边带传输时钟信号（可选）
sb_data	边带传输数据信号（可选）

12 调试追踪 Debug&Trace

本章描述各个层次提供的调试追踪功能，目的是方便从外部通过寄存器接口全方位了解各个层次内部的运行状态。主要包括如下6个功能类别：

- 测试序列发送、检测和校验功能；
- 环回功能；
- 统计计数功能，不在标准中规定，由用户自定义实现；
- 注入错误、暂停发包功能，不在标准中规定，由用户自定义实现；
- 告警/状态上报功能，不在标准中规定，由用户自定义实现；
- 跟踪功能，不在标准中规定，由用户自定义实现。

12.1 测试序列发送、检测和校验功能

发送端支持发送如下测试码流，用于接收端校验所接收数据的正确性：

- 发送端数字物理层支持发送 PRBS7、PRBS9、PRBS15、PRBS23、PRBS31 序列；
- 发送端数字物理层支持循环发送用户自定义的长度为 2048bit 的固定序列。

接收端支持检测如下测试码流，用于接收端校验所接收数据的正确性：

- 接收端数字物理层支持校验 PRBS7、PRBS9、PRBS15、PRBS23、PRBS31 序列，并单独上报每个通道的校验结果；
- 接收端数字物理层支持检测用户自定义的长度为 2048bit 的固定序列，并单独上报每个通道的校验结果。

每条通道皆需具备测试序列的发送和检测校验功能，接收端的错误校验器通过对比参考 PRBS 生成器生成的序列数据和接收到的测试序列数据（或是用户定义序列），来判断数据在通道中是否发生了错误。若发生了错误，则将错误信息上报。测试序列的发送、检测和校验如图 12-1 所示。

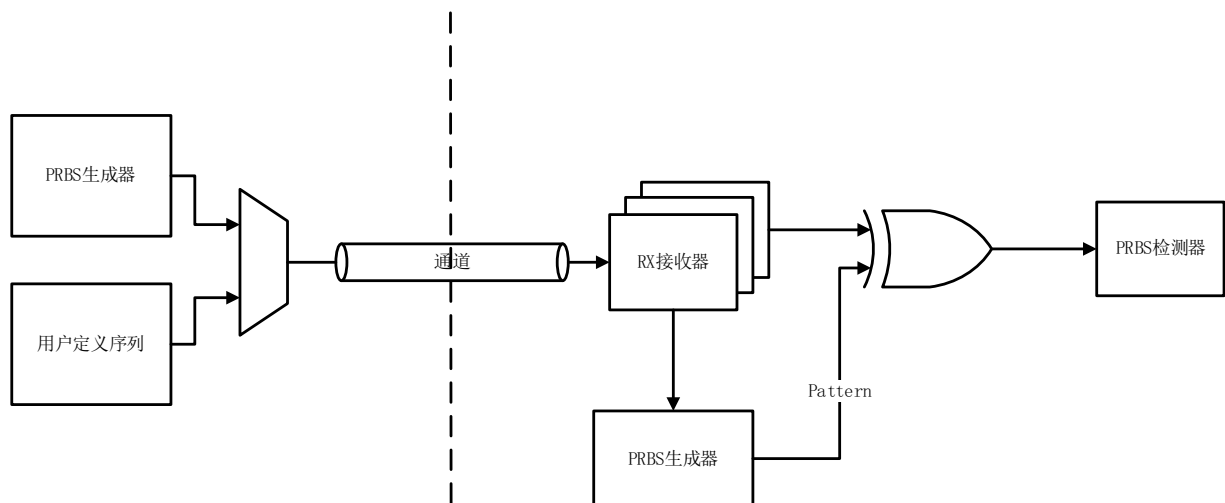


图12-1 测试序列的发送、检测和校验示意图

12.2 环回功能

环回功能主要用于测试过程中对链路通讯进行确认，对于D2D接口的环回功能，本标准定义了不同的环回点进行环回测试，如图12-2所示。

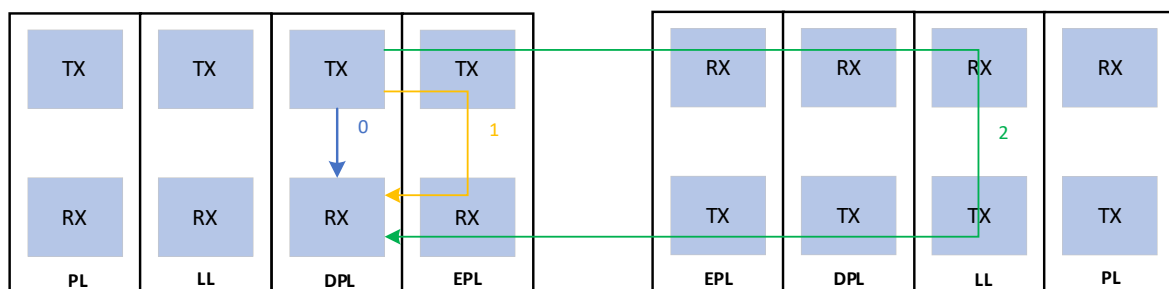


图12-2 环回路径示意图

本标准中定义的环回点如下：

- 环回点 0：从近端数字物理层的发送端环回到近端数字物理层的接收端；
- 环回点 1：从近端电气物理层的发送端环回到近端电气物理层的接收端；
- 环回点 2：从远端链路层的接收端环回到远端链路层的发送端；

13 功能安全设计

本标准用于道路车辆功能安全芯粒的互联，而对于车辆功能安全本标准中规定了相应的设计，如图13-1所示，本标准中的车辆功能安全主要基于存储保护，总线端到端保护，预警上报等措施。

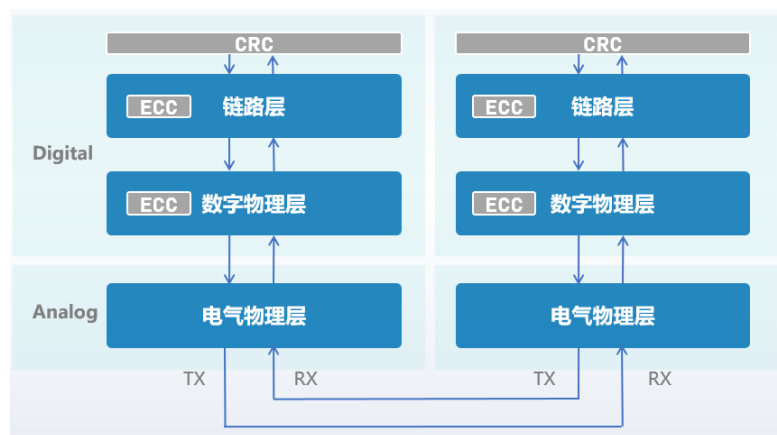


图13-1 车规级功能安全总图

13.1 存储保护

在标准的链路层与数字物理层中，存在基于sram的存储，该标准规定在存储上需要加入ECC校验保护，确保存储的数据保护，实现1bit纠错，2bit以上检测出错误，检测出无法纠正的错误时上报预警，从而以ECC为安全机制实现存储从单点失效，转换为双点失效，同时对ECC安全机制进行逻辑BIST，已检测到ECC逻辑的失效，如图13-2所示：

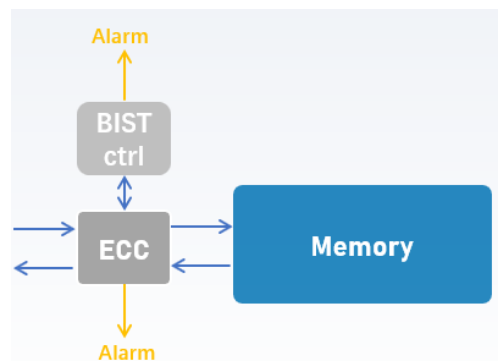


图13-2 存储保护

13.2 端到端总线保护

总线端到端保护，协议中数据将经过本端的链路层，数字物理层，电气物理层，经过差分线到对端的电气物理层，数字物理层，最后到对端的链路层。本标准中针对此数据路径采用端到端的数据保护方式实现功能安全。

在数据输入本端链路层之前，会通过CRC计算对数据生成冗余位，当数据传输到对端链路层后将数据进行CRC校验，从而避免数据传输中的单点失效，囊括数据在传输过程中的失效问题。同时对于CRC计算与校验逻辑进行BIST检测，对CRC计算与校验逻辑进行失效检测。

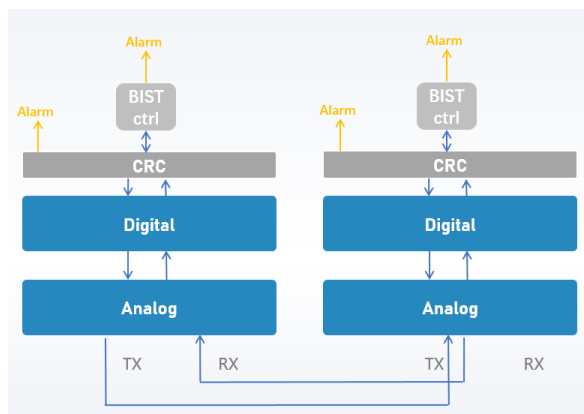


图13-3 端到端总线保护

13.3 预警上报

在本标准中规定了预警上报，对于影响D2D传输数据正确和稳定性的事件发生均需要进行预警上报，包括但不限于数据ECC，CRC校验出错，FIFO中空满标志，PLL失锁，电气物理层PLL失锁，信号探测模块signal_detect在传输过程中检测不到信号能量波动，CDR失锁等。以保证在传输数据过程中发生不稳定事件后进行预警，以便上一层做出最后的决策。

13.4 存储器自测试及修复（MBIST and Memory repair）

对嵌入式存储器需要采用内建自测试结构，目标是在车辆使用空闲时，能够启动存储器测试；且使用尽可能少的测试管脚，来完成激励输入，检测输出。存储器自测试结构需要内置激励产生单元，如图13-4所示，在启动测试信号拉高后产生测试激励对存储器进行读写操作，读取存储器内部值，在比较器内与期望值进行对比。对比结果通过一个标志位输出，通过标志位信号可以判断存储器是否存在缺陷。

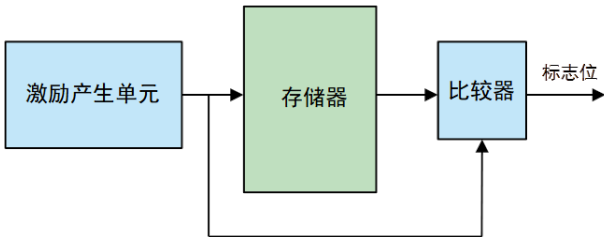


图13-4 存储器内建自测试示意图

上述对存储器的读写操作，即对连续地址空间升序或降序写入特定数值，并读出，这种读写操作称为测试算法。复杂算法带来更高的覆盖率，保障功能安全。

为了满足车规功能安全需求，存储器测试需要包含自修复功能。测试方案包括，必须选择带有冗余行/列逻辑的存储器，如图13-5所示，为了简化，这里存储器只标识三列存储单元，最右侧是冗余列，通过存储器下方选择器的选择端来决定是否选择冗余列。当存储器自测试发现存储器故障时，用冗余行替换故障行，并将这种替换信息，即选择器选择端的值，存入片上非易失性可编程存储器。

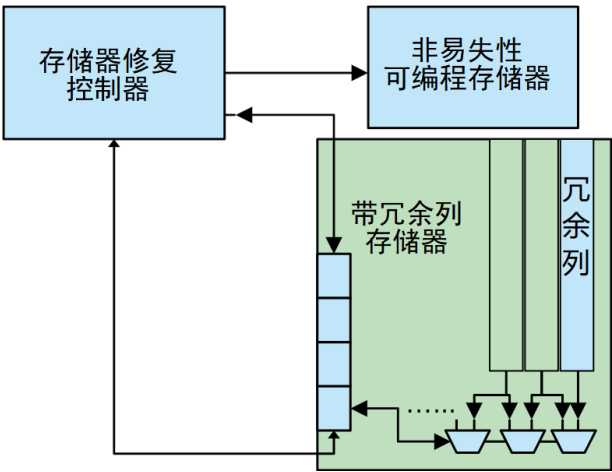


图13-5 存储器自修复示意图

存储器自测试与修复，很大程度上提高了芯片良率和可靠性，增强后期使用中发生故障时的处理能力，适用车规应用场景。

13.5 逻辑自测试 (LBIST)

车规芯片需要满足，车辆使用空闲时刻的逻辑自测试。对于片上逻辑需要加入自测试结构；同时考虑顶层管脚资源的限制，需要使用压缩逻辑，将数目庞大的测试扫描链，压缩为若干个测试通道；扫描链的输出经过压缩逻辑，压缩成若干输出。

如图13-6所示，启动逻辑自测试后，伪随机图形产生器产生测试图形，经过内置有线性反馈移位寄存器的解压缩逻辑解压后灌入扫描链中，通过一系列移位，捕获，再移位操作，抓取电路反馈值并且与期望值相比，判断电路是否存在故障。另外为了方便判断电路是否正确，扫描链的输出需要经过压缩逻辑进行压缩，压缩结果进入多输入签名寄存器电路，在这里和期望值进行对比得到一位结果标志位，这样做大大减少了所需管脚数目。

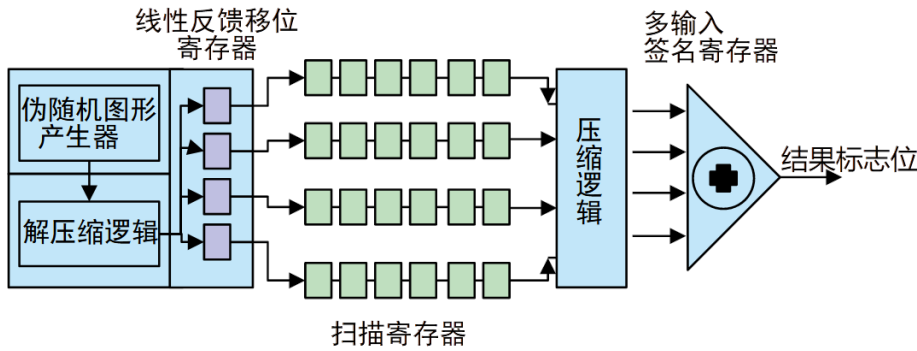


图13-6 逻辑自测试示意图

13.6 安全岛控制

存储器自测试和逻辑自测试必须受到来自安全岛的专用车规CPU的控制，在车辆上电和空闲时，启动测试。如图13-7所示，测试时，专用车规CPU发送测试指令，通过总线接口向系统内测试控制器下令或请求，由系统内测试控制器通过标准测试访问接口启动自测试，存储器自测试和逻辑自测试对电路测试后得到结果，最终给到系统内测试控制器收集结果，反馈给专用车规CPU。安全岛内CPU采用双锁步设计，为测试程序安全运行提供保障。

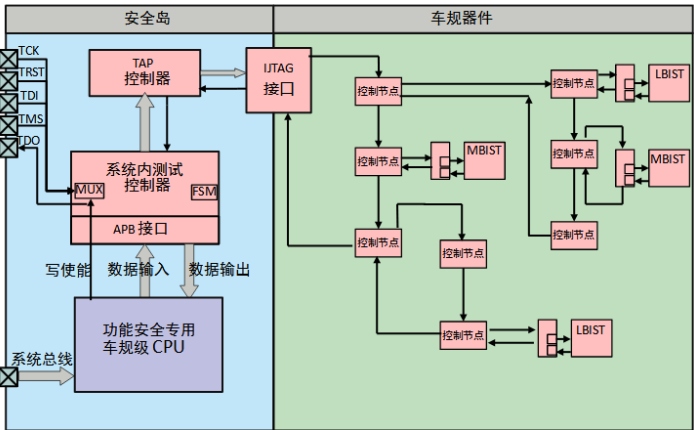


图13-7 带安全岛控制的DFT测试方案

13.7 信道冗余及重映射设计

为了提高可靠性，以及满足车规需求，在信道组之外，需要添加冗余信道。当某一信道发生故障时，用冗余信道去替换故障信道，这一冗余设计将提升良率和可靠性。

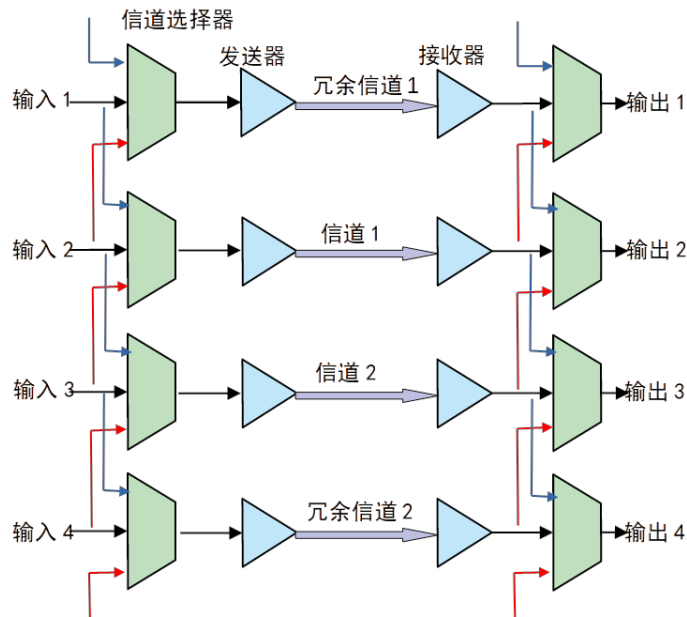


图13-8 带冗余信道的传输信道结构图

在信道的发送端和接收端分别加入一个3输入的信道选择器，如图13-8所示，信道上信号同时地，逐次地，送入相邻两边信道选择器。这样，如果信道1发生故障，通过控制选择器的选择端，将输入2映射到冗余信道1的选择器上，通过冗余信道的发送器发送输入2，同时冗余信道1的接收器输出，再映射回到原先信道1的输出2，这样输入2到输出2的传输就通过冗余信道1完成。同理，信道2故障时，输入3到输出3的传输，可以向下映射到冗余信道2上进行，如图13-9所示。

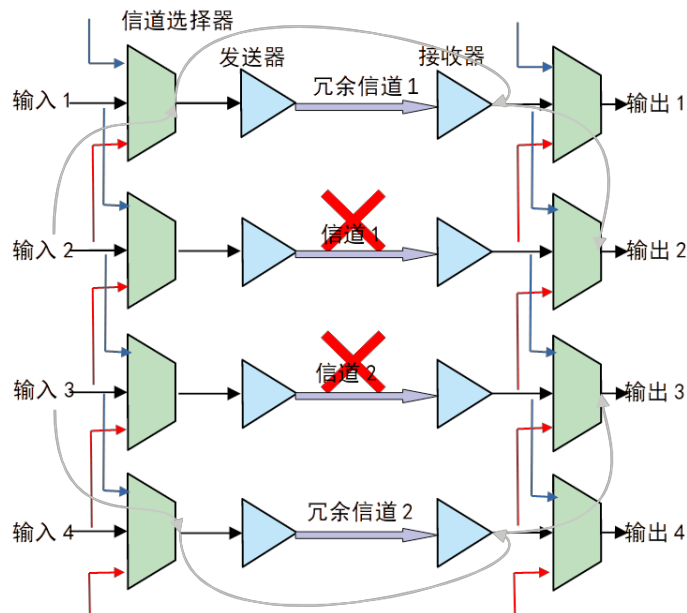


图13-9 带冗余信道的传输信道结构图

由于车规芯片测试对测试时间的需求，标准中特加入快速测试相关模式，具体方法如下：

13.8 信道复用模式及快速测试

信道设计上加入旁路结构，即复用信道做为快速测试通道。这个模式使差分输入端口可以被复用为测试通道输入端口，如图13-10所示，测试激励通过测试通道输入端口，进入片内进行测试；同时，差分输出端口可以被复用为测试通道输出端口，片内响应通过测试通道输出端口，输出片外观测。高速差分端口的复用为测试通道端口，可以加快测试速度。

旁路模式规定，差分信道输入可以直接被导入片内，输入信道被旁路；差分信道输出可以直接来自片内，输出信道被旁路。

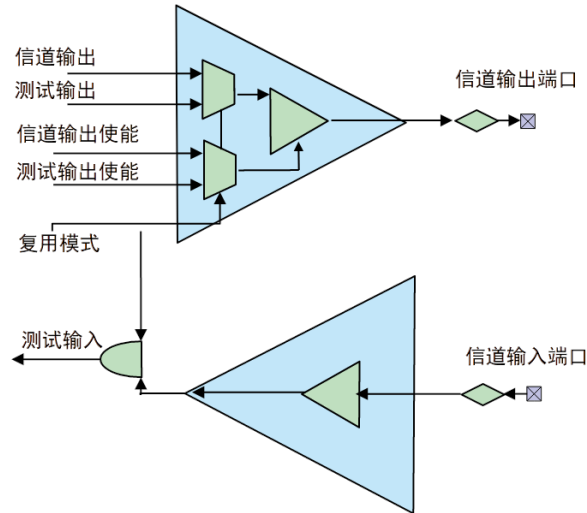


图13-10 传输信道端口复用结构图

13.9 信道的 Loopback 测试

loopback内回环测试模式中，如图13-11所示，输出侧输出预置测试图形，通过内部选择环回到输入侧，输入测采样数据和预置测试图形进行对比，判断内部信道通路是否正常。

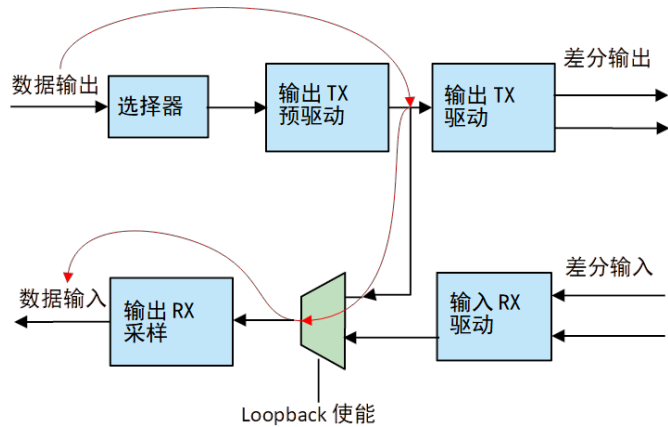


图13-11 内回环数据路径

loopback外回环测试模式中，如图13-12所示，输出侧输出预置测试图形，必须通过输出端口输出到片外，并且通过输入侧的选择器环回到输入侧，输入测采样数据和预置测试图形进行对比，判断内部信道通路是否正常。

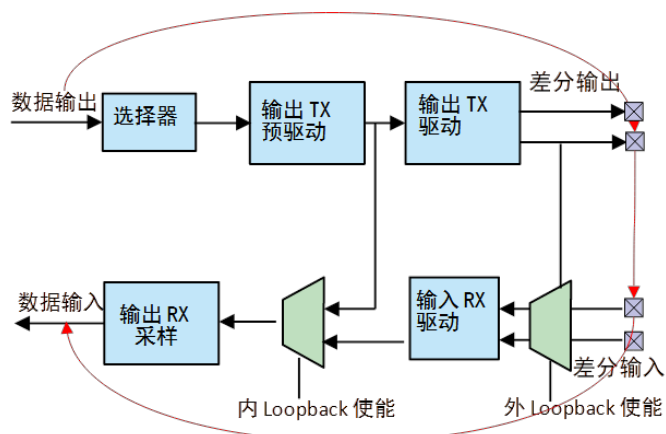


图13-12 外回环数据路径

Loopback测试能够满足封装之前信道通路的自检自测，可提前发现故障，加快测试速度。

14 配置参数

本标准中提出 D2D 必须的寄存器配置，配置寄存器如表 14-1 所示：

表 14-1 寄存器列表

寄存器名称	位宽	初始值	寄存器描述
code_stp	7:0	0xFB	STP 符号
code_sdp	7:0	0x5C	SDP 符号
code_end	7:0	0xFD	END 符号
code_com	31:0	0xBCBCBC7d	COM 符号
code_idl	7:0	0xDC	IDL 符号
code_pad	7:0	0x0	PAD 符号
idle	0:0	0x0	通过该配置可以将数据链路状态切换至 IDLE 状态，高有效
train_link_en	0:0	0x0	通过该配置信号向 controller 发起建链的请求，高有效
train_rate	1:0	0x3	EPL 通讯速率： 2'b00: 32GT/s 2'b01: 64GT/s 2'b10: 128GT/s 2'b11: reserve
lane_enable	7:0	0xFF	配置需要使用到的通道，8bit 对应 8 条通道，对应 bit 为 1 表示使能该通道
lane_mode	1:0	0x3	通道模式选择： 2'b00: 1 通道模式 2'b01: 2 通道模式 2'b10: 4 通道模式 2'b11: 8 通道模式

lane_link	23:0	0xFAC688	8 条通道的连接方式，对其配置可以实现各通道的全交叉连接，从低到高每 3bit 表示一条通道的交叉连接方式，例如[2:0]=0x7，则表示第 0 条通道交叉连接到第 7 条通道
loopback	1:0	0x0	loopback 模式： 2'b00: 非 loopback mode 2'b01: loopback0 mode，即本端 DPL 的发送与接收环回 2'b10: loopback1 mode，即本端 EPL 的发送与接收环回 2'b11: loopback2 mode，即从远端链路层的接收与发送端环回
data_sca_bypass	0:0	0x0	数据加扰旁路使能信号，为 1 时旁路掉加扰
training_time	4:0	0x2	training 状态下发生超时的时间限制，当超过该时间限制还未 training 完成则认为超时，每增加 1 则增加 500μs，例如当 training_time 为 5'd10 时，则时间为 5ms
null_send_len	15:0	0x3FF	TX LINK 链路训练一次使能最少发送的 NULL 序列个数，发送个数为配置值加 1，默认值为 16'd1023
acknak_lantency_time	15:0	0xFF	在接收到 TLP 的情况下，每隔多少个时钟周期产生 ACK 或 NAK
wait_expect_id_time	15:0	0x1FF	在产生 NAK 情况下，应该在多少个时钟周期内收到重发的包
crc_check_bypass	0:0	0x0	不检测 CRC 校验错误，高电平有效
null_det_len	15:0	0x10	RX LINK 检测 NULL 码个数
tx_dpl_polar_reverse	7:0	0x0	DPL 发送端 8 条通道的极性反转使能
rx_dpl_polar_reverse	7:0	0x0	DPL 接收端 8 条通道的极性反转使能

如上寄存器需要在建链之前，需要 D2D 近端与远端进行配置统一，然后分别配置 D2D 近端与远端的使能信号如表 14-2 所示：

表 14-2 EPL 使能配置

epl_pll_pu	0:0	0x0	EPL 的 PLL 使能
epl_tx_pu	7:0	0x0	EPL 8 条通道发送端的使能信号
epl_rx_pu	7:0	0x0	EPL 8 条通道接收端的使能信号

如表 14-1 与表 14-2 中的配置，本标准提出两种配置方式：边带传输配置方式与自定义传输配置：

- 边带传输方式见第 10 章；
- 自定义传输配置为 D2D 近端与远端本身就具有除了 D2D EPL 接口之外的其他可靠通讯接口，即可复用该接口进行表 14-1 的寄存器配置统一以及表 14-2 的使能操作。

15.1 封装类型

PHY设计应匹配bump pitch等于153um的Flip-Chip（FC）封装形式，如图15-1所示。

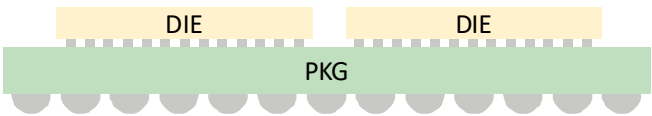


图15-1 标准封装

15.2 Bump 方案

Bump排列应如图15-2所示所示方案，旋转180°即可实现对应TX、RX的等长走线对接。

AVDDL		AVDDL		AVDDL		AVDDL		AVDDH PLL	AVDDL PLL	TP		AVDDL		AVDDL		AVDDL		AVDDL
	RXP0		RXP1		RXP2		RXP3		REXT		TXP3		TXP2		TXP1		TXP0	
RXN0		RXN1		RXN2		RXN3		REFCLK OUTP		REFCLK INP		TXN3		TXN2		TXN1		TXN0
	GND		GND		GND		GND		GND		GND		GND		GND		GND	
TXN7		TXN6		TXN5		TXN4		REFCLK OUTN		REFCLK INN		RXN4		RXN5		RXN6		RXN7
	TXP7		TXP6		TXP5		TXP4					RXP4		RXP5		RXP6		RXP7

图15-2 Bump排列

15.3 绕线方案

参考绕线方案至少需要用到4层无芯板的封装走线层：

- L1层为TX4~TX7与对端RX4~RX7信号互连层
- L2为GND互连层
- L3为TX0~TX3与对端RX0~RX3信号互连层
- L4层GND互连层

15.4 信号走线规范

封装信号走线应参照表15-1规范：

表15-1 封装内信号走线约束

	最小值	典型值	最大值
差分对内长度差			10um
差分对间长度差			100um
走线长度			10mm
走线阻抗	90Ω	100Ω	110Ω

PKG信号走线损耗及隔离度应满足表15-2规范：

表15-2 封装内信号损耗约束

	0-16GHz	16-32GHz
插入损耗	<1.5dB	<3dB
回波损耗	<-20dB	<-10dB
差模隔离度	<-30dB	

封装内信号线损耗指标约束如图15-3:

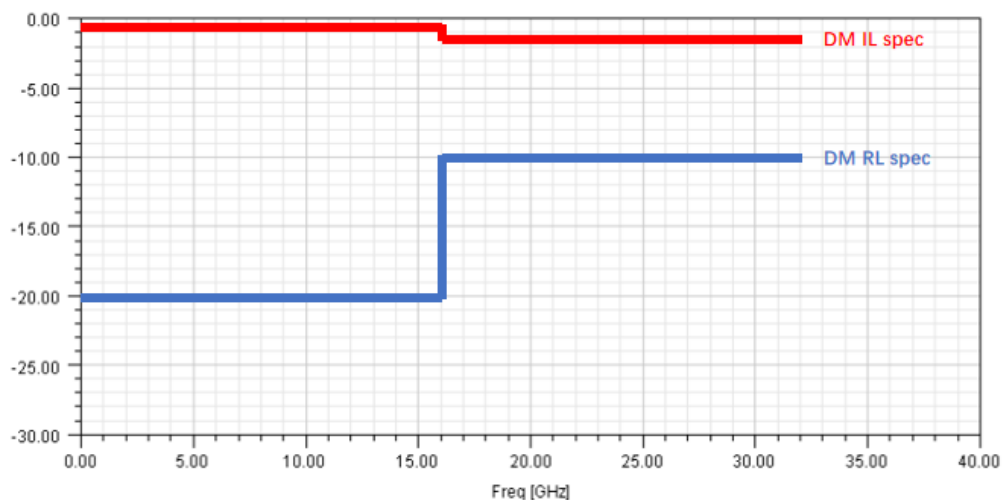


图15-3 信号线损耗指标约束

封装内信号线串扰指标约束如图15-4:

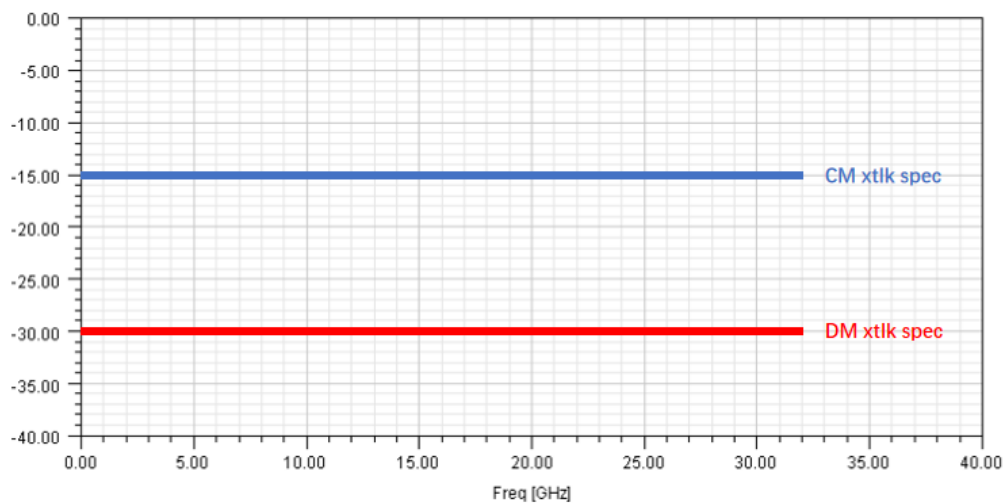


图15-4 信号线串扰指标约束

15.5 时钟方案

为使系统能够简化时钟方案，D2D phy需兼容参考时钟并联和参考时钟串联方案，因此必须有clkin接受参考时钟和clkout传递参考时钟，clkout输出时钟信号与clkin输入的时钟信号同频同相，如图15-5所示。

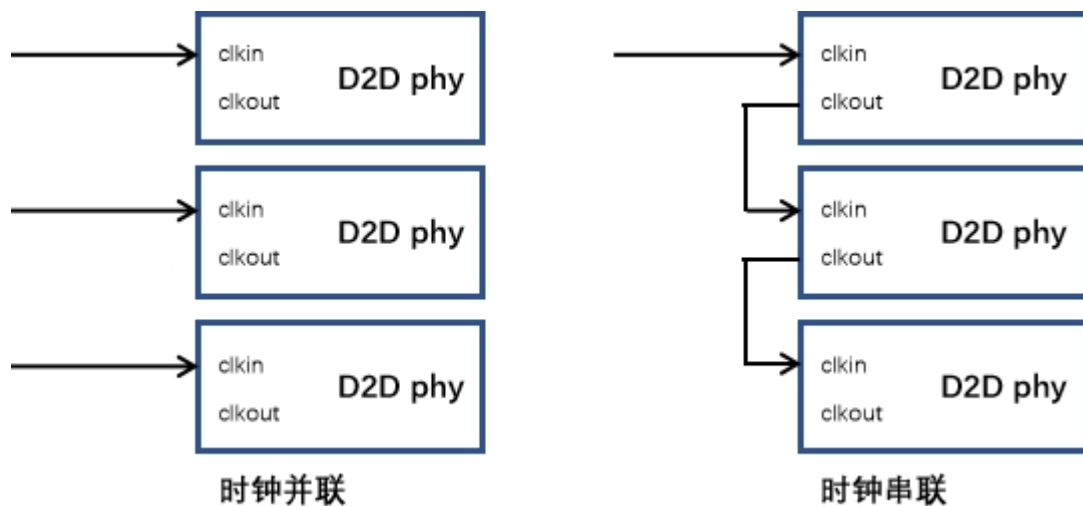


图15-5 时钟方案

15.6 信号完整性

为保证信号传输稳定可靠，RX端的1e-12误码率眼图需满足表15-3中的指标：

表15-3 眼图指标约束

	NRZ	PAM4
接收端最小眼宽	0.5UI	0.17UI
接收端最小眼高	400mV	133mV

NRZ眼图如图15-6所示，菱形区域为眼图指标约束，信号眼图的眼宽眼高应大于眼图指标约束：

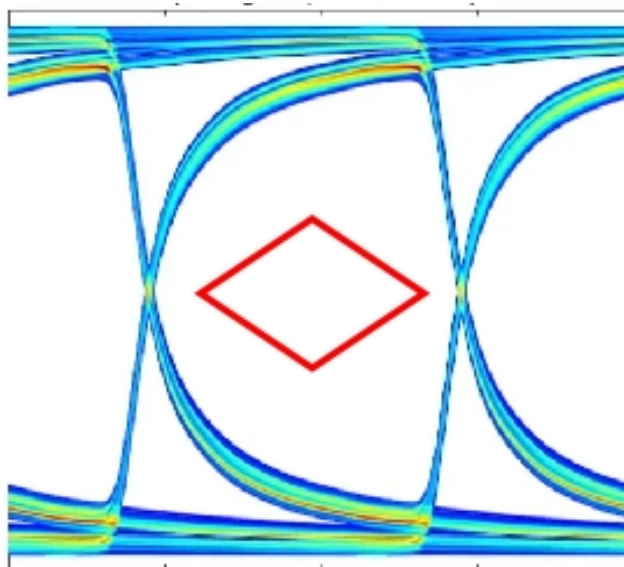


图15-6 NRZ眼图约束

PAM4眼图如图15-7所示，三个菱形区域为眼图指标约束，信号眼图的眼宽眼高应大于眼图指标约束：

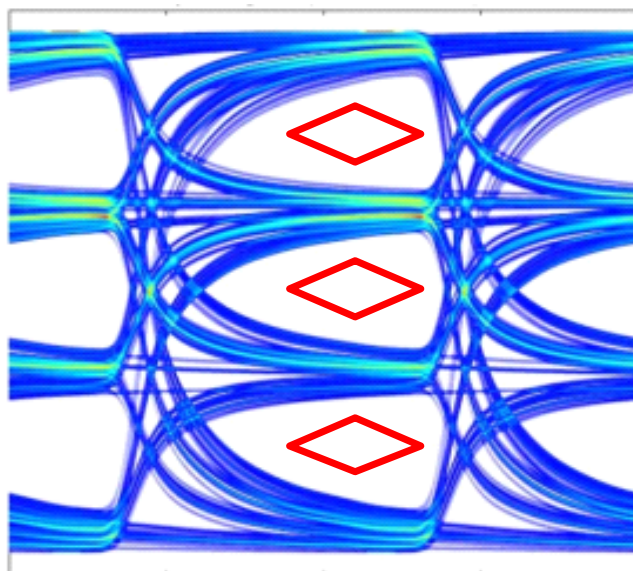


图15-7 PAM4眼图约束