(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2024-0028835
(43) 공개일자 2024년03월05일

(51) 국제특허분류(Int. Cl.)
H04L 1/16 (2023.01) *H04L 1/00* (2006.01)
H04L 1/18 (2023.01)
(52) CPC특허분류
H04L 1/1607 (2023.01)
H04L 1/004 (2013.01)
(21) 출원번호 10-2022-0107080
(22) 출원일자 2022년08월25일
심사청구일자 2022년08월25일

(71) 출원인
인하대학교 산학협력단
인천광역시 미추홀구 인하로 100(용현동, 인하대학교)
상명대학교 천안산학협력단
충청남도 천안시 동남구 상명대길 31, 상명대학교내 (안서동)
(72) 발명자
강진구
서울특별시 서초구 양재대로2길 109, 104동 1202호(우면동, 서초호반써밋)
신상웅
인천광역시 미추홀구 경인남길102번길 59-14, 205호(용현동)
김용우
서울특별시 송파구 백제고분로19길 3-22, 3층(잠실동)
(74) 대리인
양성보

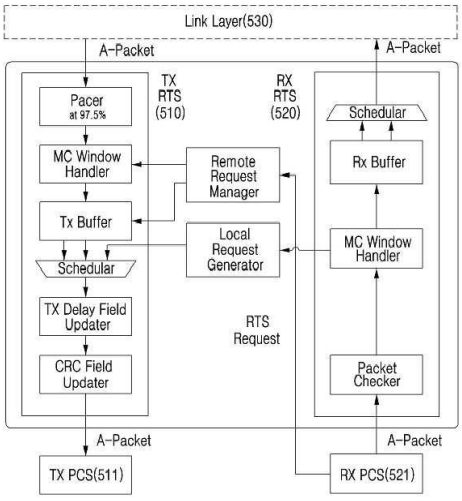
전체 청구항 수 : 총 8 항

(54) 발명의 명칭 **MIPI A-PHY RTS 계층 설계 및 구현**

(57) 요 약

MIPI A-PHY 기반 RTS 계층 구조 및 그 동작 방법이 제시된다. 본 발명에서 제안하는 MIPI A-PHY 기반 RTS(Retransmission) 계층 구조는 파서(Pacer)를 통해 물리 계층이 포화되지 않은 A-패킷을 링크 계층(Link Layer)으로부터 전달 받아 상기 A-패킷에 재전송을 위한 순차적인 값을 삽입하여 저장하는 TX RTS 및 상기 A-패킷을 수신하여 상태를 확인하여 상기 A-패킷의 상태에 따라 재전송 요청을 보내고 해당 A-패킷을 저장하는 RX RTS를 포함한다.

대 표 도 - 도5



(52) CPC특허분류
H04L 1/1858 (2021.08)
H04L 1/1874 (2013.01)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711152938
과제번호	2021-0-02052-002
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	대학ITRC센터육성지원사업(교육운영수익)
연구과제명	[Ezbaro][정부] 스마트 모빌리티를 위한 인공지능 시스템반도체 핵심 기술 개발 및 인력 양성(2차년도)
기 여 율	1/1
과제수행기관명	인하대학교 산학협력단
연구기간	2022.01.01 ~ 2022.12.31

명 세 서

청구범위

청구항 1

파서(Pacer)를 통해 물리 계층이 포화되지 않은 A-패킷을 링크 계층(Link Layer)으로부터 전달 받아 상기 A-패킷에 재전송을 위한 순차적인 값을 삽입하여 저장하는 TX RTS; 및

상기 A-패킷을 수신하여 상태를 확인하여 상기 A-패킷의 상태에 따라 재전송 요청을 보내고 해당 A-패킷을 저장하는 RX RTS

를 포함하는 RTS(Retransmission) 계층 구조.

청구항 2

제1항에 있어서,

상기 TX RTS는,

물리 계층이 포화되지 않은 A-패킷을 링크 계층으로부터 전달 받는 파서(Pacer);

순차적인 값인 MC를 상기 A-패킷 헤더에 삽입하는 MC 윈도우 핸들러(Window Handler);

상기 A-패킷을 저장하는 TX 버퍼(Buffer);

원본 A-패킷과 재전송되는 A-패킷 간의 전송 우선순위를 결정하는 스케쥴러(Schedular);

총 전송 지연시간 계산에 사용되는 딜레이(Delay)와 에러를 감지하기 위한 CRC값을 상기 재전송되는 A-패킷에 삽입하는 TX 딜레이 필드 업데이터(Delay Field Updater)와 CRC 필드 업데이터(Field Updater); 및

상기 CRC값이 삽입된 A-패킷을 전달 받는 TX PCS

를 포함하는 RTS 계층 구조.

청구항 3

제1항에 있어서,

상기 MC는 0부터 255까지의 8bit 값으로 각 A-패킷을 나타내는 순차적인 값이고, 상기 A-패킷의 순차전송 및 재전송을 가능하게 하는

RTS 계층 구조.

청구항 4

제1항에 있어서,

상기 RX RTS는,

상기 TX RTS로부터 A-패킷을 수신 받는 RX PCS;

상기 수신된 A-패킷의 CRC를 통해 오류 여부를 검사하고 상기 A-패킷의 헤더를 통해 MC, 재전송 여부, 지연시간을 포함하는 상기 A-패킷의 상태 확인하는 패킷 체커(Packet Checker);

상기 상태 확인된 A-패킷을 수신하여 상기 A-패킷의 상태에 따라 재전송 요청을 보내는 MC 윈도우 핸들러(Window Handler);

상기 A-패킷을 저장하는 RX 버퍼(Buffer);

상기 저장된 A-패킷을 원래의 순서에 맞춰 링크 계층에 전달하는 스케쥴러(Schedular); 및

상기 재전송 요청이 고속의 하향링크와 저속의 상향링크 사이를 지나도록 하는 요청 관리 및 생성기(Request

Manager/Generator)

를 포함하는 RTS 계층 구조.

청구항 5

파서(Pacer)를 통해 물리 계층이 포화되지 않은 A-패킷을 링크 계층(Link Layer)으로부터 전달 받아 상기 A-패킷에 재전송을 위한 순차적인 값을 삽입하여 저장하는 TX RTS 및 상기 A-패킷을 수신하여 상태를 확인하여 상기 A-패킷의 상태에 따라 재전송 요청을 보내고 해당 A-패킷을 저장하는 RX RTS를 포함하는 RTS(Retransmission) 계층 구조의 동작 방법에 있어서,

상기 RX RTS의 동작 방법은,

상기 A-패킷 헤더의 CRC8 오류를 감지하는 단계;

상기 A-패킷 전체의 CRC32 오류를 감지하고, 상기 CRC32와 수신된 A-패킷의 순서를 나타내는 MC를 검사하는 단계;

상기 CRC8 오류, 상기 CRC32 오류, 상기 MC 값에 따라 상기 A-패킷에 대한 단일 재전송 요청 또는 범위 재전송 요청을 전송하는 단계; 및

상기 재전송된 상기 A-패킷을 저장하는 단계

를 포함하는 방법.

청구항 6

제5항에 있어서,

상기 A-패킷 헤더의 CRC8 오류를 감지하는 단계는,

상기 CRC8에서 오류가 감지되는 경우 해당 A-패킷을 폐기하는

방법.

청구항 7

제5항에 있어서,

상기 A-패킷 전체의 CRC32 오류를 감지하고, 상기 CRC32와 수신된 A-패킷의 순서를 나타내는 MC를 검사하는 단계는,

상기 CRC8에서 오류가 감지되고, 상기 MC에서 오류가 감지되는 경우 해당 A-패킷을 폐기하는

방법.

청구항 8

제5항에 있어서,

상기 CRC8 오류, 상기 CRC32 오류, 상기 MC 값에 따라 상기 A-패킷에 대한 단일 재전송 요청 또는 범위 재전송 요청을 전송하는 단계는,

상기 CRC8에서 오류가 감지되고, 상기 MC에서 오류가 감지되지 않은 경우 해당 A-패킷에 대한 단일 재전송 요청을 전송하고,

상기 CRC8에서 오류가 감지되지 않고, 상기 MC에서 오류가 감지되는 경우 해당 A-패킷에 대한 범위 재전송 요청을 전송하는

방법.

발명의 설명

기 술 분 야

[0001] 본 발명은 MIPI A-PHY RTS 계층 설계 및 구현 방법에 관한 것이다.

배 경 기 술

[0002] 최근 자율주행 자동차 기술의 발전으로 카메라, 라이다, 레이더 센서 및 디스플레이 등 다양한 전자제어유닛(Electronic Control Unit; ECU)들이 차량 내 네트워크(In Vehicle Network; IVN)에 연결되고 있다. 특히 빠르게 발전하는 첨단 운전자 보조 시스템(Advanced Driver Assistance System; ADAS)과 자율주행 시스템(Autonomous Driving System; ADS)을 위해 요구되는 카메라와 디스플레이의 성능과 개수가 크게 증가하고 있으며, 이에 상응하여 데이터 전송 속도 또한 증가하고 있다. 현재 고해상도의 센서에서는 약 1.8Gbps에서 14.4Gbps까지의 데이터 전송속도가 요구되며, 2025년에 는 100Gbps까지 전송 속도 증가가 요구될 것으로 전망된다[1]. 그러나 기존 차량 내 네트워크(IVN)인 CAN, LIN, Flexray, MOST 등은 1~150 Mbps의 전송 속도 한계를 가져 증가하는 데이터 전송 속도를 감당할 수 없다[2]. 이를 해결하기 위해 차량용 이더넷과 SerDes(Serializer/Deserializer)가 제안되었다. 차량용 이더넷의 경우 기존 상용 이더넷의 상위 계층에 차량용 물리 계층만을 설계하여 사용 가능하며, 현재 10Gbps의 전송 속도를 지원하는 표준이 제정 되어있다[3]. 반면 SerDes는 송수신 간의 비대칭 데이터 전송에 최적화 되어있으며, 이더넷에 비해 상위 계층이 단순하다는 특징을 가진다. 따라서 점대점 비대칭 데이터 통신으로 구성되는 ADAS 센서 네트워크에는 차량용 이더넷 보다 SerDes가 적합하며, 차량용 이더넷은 그 외의 백본 네트워크에 적합하다[4-5]. SerDes를 차량에 적용하기 위해 ASA(Automotive SerDes Alliance)와 MIPI(Mobile Industry Processor Interface)의 A-PHY가 차량용 SerDes 표준으로 제안 되었다[6-7]. 두 표준 모두 최대 16Gbps의 전송속도를 제공하며 차량환경에 맞는 높은 안정성과 보안 그리고 최대 15m의 장거리 수용성 등을 제공한다. 두 표준은 유사하지만 MIPI A-PHY는 기존 카메라와 디스플레이 인터페이스 표준인 CSI(Camera Serial interface), DSI(Display Serial Interface) 및 VESA(Video Electronics Standards Association)의 DP 등 기존 프로토콜을 지원하는 장점을 가지며, 현재 A-PHY v1.0은 IEEE 2977-2021와 같이 표준으로 제정 되어있다[7]. MIPI A-PHY의 두 가지 큰 특징 중, 첫째는 NRZ(Non Return to Zero) 및 PAM(Pulse Amplitude Modulation)을 이용한 멀티레벨 시그널링(Multilevel Signaling) 기법이다. 수신단 신호의 상태에 따라 송신단의 시그널링 방식을 PAM4에서 NRZ로 또는 PAM8에서 PAM4등으로 조정 하여 전송 속도를 조절하고 전송 오류에 대응한다. 두 번째 특징은 오류를 감지하고 데이터의 재전송을 요청하는 RTS(Retransmission) 계층이다. 이 새로운 물리 계층은 데이터 재전송을 통해 잡음이 많은 차량 환경에서 높은 면역을 얻을 수 있다.

발명의 내용

해결하려는 과제

[0003] 본 발명이 이루고자 하는 기술적 과제는 차량용 SerDes 표준인 MIPI A-PHY 에서 새롭게 제시된 물리계층인 RTS 계층 설계 방안을 제공하는데 있다. 제안하는 RTS 계층은 재전송을 위해 패킷을 저장하는 버퍼, 재전송을 위해 A-패킷에 값을 삽입을 하는 모듈들, A-패킷의 상태를 읽고 필요에 따라 재전송을 요청하는 모듈들을 포함할 수 있다.

과제의 해결 수단

[0004] 일 측면에 있어서, 본 발명에서 제안하는 MIPI A-PHY 기반 RTS(Retransmission) 계층 구조는 파서(Pacer)를 통해 물리 계층이 포화되지 않은 A-패킷을 링크 계층(Link Layer)으로부터 전달 받아 상기 A-패킷에 재전송을 위한 순차적인 값을 삽입하여 저장하는 TX RTS 및 상기 A-패킷을 수신하여 상태를 확인하여 상기 A-패킷의 상태에 따라 재전송 요청을 보내고 해당 A-패킷을 저장하는 RX RTS를 포함한다.

[0005] 상기 TX RTS는 물리 계층이 포화되지 않은 A-패킷을 링크 계층으로부터 전달 받는 파서(Pacer), 순차적인 값인 MC를 상기 A-패킷 헤더에 삽입하는 MC 윈도우 핸들러(Window Handler), 상기 A-패킷을 저장하는 TX 버퍼(Buffer), 원본 A-패킷과 재전송되는 A-패킷 간의 전송 우선순위를 결정하는 스케줄러(Schedular), 총 전송 지연시간 계산에 사용되는 딜레이(Delay)와 에러를 감지하기 위한 CRC값을 상기 재전송되는 A-패킷에 삽입하는 TX 딜레이 필드 업데이터(Delay Field Updater)와 CRC 필드 업데이터(Field Updater) 및 상기 CRC값이 삽입된 A-패킷을 전달 받는 TX PCS를 포함한다.

[0006] 상기 MC는 0부터 255까지의 8bit 값으로 각 A-패킷을 나타내는 순차적인 값이고, 상기 A-패킷의 순차전송 및 재전송을 가능하게 한다.

[0007] 상기 RX RTS는 상기 TX RTS로부터 A-패킷을 수신 받는 RX PCS, 상기 수신된 A-패킷의 CRC를 통해 오류 여부를 검사하고 상기 A-패킷의 헤더를 통해 MC, 재전송 여부, 지연시간을 포함하는 상기 A-패킷의 상태 확인하는 패킷 체커(Packet Checker), 상기 상태 확인된 A-패킷을 수신하여 상기 A-패킷의 상태에 따라 재전송 요청을 보내는 MC 윈도우 핸들러(Window Handler), 상기 A-패킷을 저장하는 RX 버퍼(Buffer), 상기 저장된 A-패킷을 원래의 순서에 맞춰 링크 계층에 전달하는 스케줄러(Schedular) 및 상기 재전송 요청이 고속의 하향링크와 저속의 상향 링크 사이를 지나도록 하는 요청 관리 및 생성기(Request Manager/Generator)를 포함한다.

[0008] 또 다른 일 측면에 있어서, 본 발명에서 제안하는 파서(Pacer)를 통해 물리 계층이 포화되지 않은 A-패킷을 링크 계층(Link Layer)으로부터 전달 받아 상기 A-패킷에 재전송을 위한 순차적인 값을 삽입하여 저장하는 TX RTS 및 상기 A-패킷을 수신하여 상태를 확인하여 상기 A-패킷의 상태에 따라 재전송 요청을 보내고 해당 A-패킷을 저장하는 RX RTS를 포함하는 RTS(Retransmission) 계층 구조의 동작 방법에 있어서, 상기 RX RTS의 동작 방법은 상기 A-패킷 헤더의 CRC8 오류를 감지하는 단계, 상기 A-패킷 전체의 CRC32 오류를 감지하고, 상기 CRC32와 수신된 A-패킷의 순서를 나타내는 MC를 검사하는 단계, 상기 CRC8 오류, 상기 CRC32 오류, 상기 MC 값에 따라 상기 A-패킷에 대한 단일 재전송 요청 또는 범위 재전송 요청을 전송하는 단계 및 상기 재전송된 상기 A-패킷을 저장하는 단계를 포함한다.

발명의 효과

[0009] 본 발명의 실시예들에 따르면 새롭게 제안된 A-PHY 표준을 기반으로 한 RTS 계층 설계 및 구현을 통해 RTS 계층이 A-PHY 표준에서 제시된 재전송 동작을 만족하고, 설계된 RTS 계층을 포함한 A-PHY는 사용되는 LUT, 레지스터, 블록 메모리의 수를 줄일 수 있고, 최대 동작 속도를 200MHz까지 증가시킬 수 있다.

도면의 간단한 설명

[0010] 도 1은 본 발명의 일 실시예에 따른 A-PHY 연결 예시를 설명하기 위한 도면이다.

도 2는 본 발명의 일 실시예에 따른 A-Packet 구조를 설명하기 위한 도면이다.

도 3은 본 발명의 일 실시예에 따른 A-PHY의 계층 구조 블록도이다.

도 4는 본 발명의 일 실시예에 따른 A-PHY 시작 절차를 설명하기 위한 도면이다.

도 5는 본 발명의 일 실시예에 따른 A-PHY의 RTS 계층의 블록도이다.

도 6은 본 발명의 일 실시예에 따른 A-패킷 수신 시 RX RTS의 동작 순서를 설명하기 위한 흐름도이다.

도 7은 본 발명의 일 실시예에 따른 재전송 요청의 구조를 나타내는 도면이다.

도 8은 본 발명의 일 실시예에 따른 A-PHY RTS 검증을 위한 블록도이다.

도 9는 본 발명의 일 실시예에 따른 CRC 32 오류 시뮬레이션 결과를 나타내는 도면이다.

도 10은 본 발명의 일 실시예에 따른 단일 재전송 요청 과정 시뮬레이션 결과를 나타내는 도면이다.

도 11은 본 발명의 일 실시예에 따른 CRC 8 오류 시뮬레이션 결과를 나타내는 도면이다.

도 12는 본 발명의 일 실시예에 따른 범위 재전송 요청 과정 시뮬레이션 결과를 나타내는 도면이다.

도 13은 본 발명의 일 실시예에 따른 A-PHY RTS 검증을 위한 FPGA 개발 환경을 나타내는 도면이다.

도 14는 본 발명의 일 실시예에 따른 A-PHY RTS 검증 결과를 나타내는 도면이다.

발명을 실시하기 위한 구체적인 내용

[0011] 본 발명에서는 새롭게 제안된 A-PHY 표준을 기반으로 한 RTS 계층 설계 및 구현에 관하여 개시한다. 본 발명의 실시예에 따라 설계된 RTS 계층이 A-PHY 표준에서 제시된 재전송 동작을 만족함을 RTL 시뮬레이션을 통해 확인하였으며, 검증을 위해 Xilinx KC705 개발 보드와 FMC(FPGA Mezzanine Card) Loopback 모듈을 이용하여 통합 검증을 위한 SerDes 환경을 구성하였다. 설계된 RTS 계층을 포함한 A-PHY는 Kintex-7 FPGA에서 합성한 결과 3,924개의 LUT, 2,019개의 레지스터, 132개의 블록 메모리를 사용하고, 최대 동작 속도가 200MHz임을 확인하였다. 이하, 본 발명의 실시 예를 첨부된 도면을 참조하여 상세하게 설명한다.

[0013] 도 1은 본 발명의 일 실시예에 따른 A-PHY 연결 예시를 설명하기 위한 도면이다.

[0014] 도 1은 A-PHY의 연결 관계를 보여주고 있으며, A-PHY는 소스(Source)와 싱크(Sink)로 구성된다. 소스는 카메라(Camera)와 같이 병렬 데이터를 직렬화해 전송하는 모듈에 연결되며, 싱크는 ECU와 같은 모듈에 연결되어 고속 직렬 데이터를 병렬화해 처리될 수 있게 전달한다. 이때 소스에서 싱크로 대용량의 고속 데이터가 전송되는 채널을 하향링크(Downlink)이라 한다. 반대로 싱크에서 소스로 저속의 제어 정보를 전달하는 채널을 상향링크(Uplink)라고 한다.

[0015] <표 1>

[0016]

Gear	Data Rate	Modulation
1	2 Gbps	NRZ-8B10B
2	4 Gbps	NRZ-8B10B
3	8 Gbps	PAM4
4	12 Gbps	PAM8
5	16 Gbps	PAM16

[0017] 표 1은 A-PHY의 기어에 따른 하향링크의 전송 속도를 나타낸다. A-PHY는 5가지 하향링크 기어를 제공하며, 각 기어에 따라 2Gbps에서 16Gbps까지의 전송 속도를 제공한다. 이때 기어 1, 2는 8B/10B 인코딩(Encoding) 방식을 이용한 NRZ방식으로 데이터를 전송하며, 기어 3, 4, 5에서는 PAM4, 8, 16 변조방식을 이용하여 데이터를 고속으로 전송한다. 상향링크의 전송 속도는 모든 기어에서 동일하게 100Mbps이다.

[0018] 본 발명에서는 재전송 동작 확인을 위해 A-PHY의 소스와 싱크를 각각 설계하였으며, 전송채널 중 하향 채널은 기어 1과 같이 NRZ 8B10B 방식의 2Gbps, 상향 채널은 100Mbps로 구성하였다.

[0020] 도 2는 본 발명의 일 실시예에 따른 A-Packet 구조를 설명하기 위한 도면이다.

[0021] 도 2는 A-PHY에서 사용되는 데이터 형식인 A-패킷(A-Packet)의 구조이다. A-패킷은 헤더(Header), 페이로드(Payload), 테일(Tail) 세부분으로 구성되며, 하향링크와 상향링크에서 동일한 형식으로 사용된다. 헤더는 상위 프로토콜의 종류, 전송 우선순위 및 재전송을

[0022] 위한 정보 등 정상적 전송에 필요한 8 Byte로 구성된다. 이때 헤더의 마지막엔 헤더의 오류를 체크하기 위해 CRC8(Cyclic Redundancy Check 8)값이 계산되어 삽입된다. 페이로드에는 실제 전송되는 데이터로 구성되며 최대 길이는 하향링크에서는 380Byte, 상향링크에서는 32Byte이다. 테일에는 헤더를 포함한 A-패킷 전체의 오류를 검사하기 위한 CRC32값이 삽입된다. 이때 A-패킷의 Delay, MC(Message Counter), CRC8, CRC32 필드는 A-PHY에서 값이 삽입되고 사용되는 필드로, A-패킷의 전송상태를 확인하고 재전송을 위한 기준이 된다.

[0024] 도 3은 본 발명의 일 실시예에 따른 A-PHY의 계층 구조 블록도이다.

[0025] 도 3은 본 발명의 일 실시예에 따른 Link Layer, RTS, PCS(Physical Coding Sub Layer), PMD(Physical Media Dependent)로 구성된 A-PHY의 계층 구조 블록도를 나타낸다.

[0026] Link Layer는 A-패킷의 스케줄링, 우선순위 지정, 전달 등을 담당한다.

[0027] RTS는 데이터 재전송을 위해 구성된 계층으 로 A-패킷을 저장하고 전송된 A-패킷의 상태에 따라 재전송 동작을 실시한다.

[0028] PCS는 A-패킷과 PHY Symbol간 상호 변환을 담당하고 PMD는 CDR(Clock Data Recovery), CTLE(Continuous Time Linear Equalizer)등 전기 사양 및 물리적 매체가 정의된다.

[0029] 본 발명에서는 PCS와 RTS 계층만을 설계하여 A-PHY 소스와 싱크 사이의 A-패킷 전송 경로와 RTS 계층에 서의 오류 감지에 따른 재전송 동작을 확인한다. PCS 계층은 8B10B과 스크램블러, Byte Stream Controller, Link

Startup으로 구성된다. Byte Stream Controller는 데이터의 전송을 제어하며, CM(Control Mark), CN(Control Nibbles)이란 제어신호를 추가해 전 송 데이터의 시작과 끝을 표시한다.

[0031] 도 4는 본 발명의 일 실시예에 따른 A-PHY 시작 절차를 설명하기 위한 도면이다.

[0032] Link Startup은 도 4와 같은 A-PHY 채널의 연결을 위한 시작 절차를 제어한다. PCS는 훈련모드, 유효모드, 정상모드 3가지 모드로 동작하며, 소스와 싱크는 총 8 단계(Step 1 - Step 8)로 이뤄진 시작 절차동안 훈련 시퀀스를 서로 주고받으며 클럭 복원, 심볼 경계 정렬, 스크램블러 동기화 등을 진행한다.

[0034] 다시 도 3을 참조하면, RTS 계층은 하향링크와 상향링크에서 송수신을 하 기에 TX RTS와 RX RTS로 구성된다. 표준에 명시된 구조에 따라 TX PCS은 재전송을 위해 전송하는 A-패킷을 저장하고, 재전송에 필요한 정보를 A-패킷에 삽입한다. 반면 RX RTS는 TX RTS에서 A-패킷에 삽입한 값을 기준으로 A-패킷을 검사하고, 필요에 따라 재전송을 요청할 수 있는 구조로 설계되어야 한다.

[0036] 도 5는 본 발명의 일 실시예에 따른 A-PHY의 RTS 계층의 블록도이다.

[0037] 도 5는 제안하는 A-PHY의 RTS 계층의 블록도이다. TX RTS(510)는 파서(Pacer)를 통해 물리 계층이 포화되지 않을 정도의 A-패킷을 Link Layer(530)로부터 전달받는다. TX RTS(510)의 MC Window Handler를 통해 순차적인 값인 MC를 A-패킷 헤더에 삽입하고 A-패킷을 TX Buffer에 저장한다. 이때 MC는 0부터 255까지의 8bit 값으로 각 A-패킷들을 지칭하는 기준이 되며, A-패킷의 순차전송 및 재전송을 가능하게 한다. Schedular는 원본과 재전송되는 A-패킷 간의 전송 우선순위를 결정하는 역할을 한다. 이후 TX Delay Field Updater와 CRC Field Updater를 통해 총 전송 지연시간 계산에 사용되는 Delay와 에러를 감지하기 위한 CRC값을 A-패킷에 삽입한 후 TX PCS(511)에 전달된다. RX PCS(521)를 통해 RX RTS(520)에 A-패킷이 수신되면, Packet Checker를 통해 수신된 A-패킷의 상태를 확인한다. 여기서 A-패킷의 CRC를 통해 오류 여부를 검사하고 헤더를 통해 MC, 재전송 여부, 지연시간 등 A-패킷의 상태를 MC Window Handler에 전달한다. RX RTS(520)의 MC Window Handler는 A-패킷의 상태에 따라 재전송 요청을 보내고 A-패킷을 RX Buffer에 저장한다. Schedular는 RX Buffer에 저장된 A-패킷들을 원래의 순서에 맞춰 Link Layer(530)에 전달한다. 이때 Request Manager/Generator는 재전송 요청이 고속의 하향링크와 저속의 상향링크 사이를 적절히 지나게 도와주는 모듈이다.

[0039] 도 6은 본 발명의 일 실시예에 따른 A-패킷 수신 시 RX RTS의 동작 순서를 설명하기 위한 흐름도이다.

[0040] 본 발명의 실시예에 따른 파서(Pacer)를 통해 물리 계층이 포화되지 않은 A-패킷을 링크 계층(Link Layer)으로부터 전달 받아 상기 A-패킷에 재전송을 위한 순차적인 값을 삽입하여 저장하는 TX RTS 및 상기 A-패킷을 수신하여 상태를 확인하여 상기 A-패킷의 상태에 따라 재전송 요청을 보내고 해당 A-패킷을 저장하는 RX RTS를 포함하는 RTS(Retransmission) 계층 구조의 동작 방법에 있어서 상기 RX RTS의 동작 방법은 상기 A-패킷 헤더의 CRC8 오류를 감지하는 단계(610), 상기 A-패킷 전체의 CRC32 오류를 감지하고, 상기 CRC32와 수신된 A-패킷의 순서를 나타내는 MC를 검사하는 단계(620), 상기 CRC8 오류, 상기 CRC32 오류, 상기 MC 값에 따라 상기 A-패킷에 대한 단일 재전송 요청 또는 범위 재전송 요청을 전송하는 단계(630) 및 상기 재전송된 상기 A-패킷을 저장하는 단계(640)를 포함한다.

[0041] 단계(610)에서, A-패킷 헤더의 오류를 감지하는 CRC8에서 오류가 감지되었다면 해당 A-패킷은 바로 폐기된다(611).

[0042] 단계(620)에서, A-패킷 전체의 오류를 감지하는 CRC32와 수신된 A-패킷의 순서를 나타내는 MC를 검사하여, 재전송 요청 종류 및 A-패킷 저장 여부를 결정한다.

[0043] 상기 CRC8에서 오류(621)가 감지되고, 상기 MC에서 오류(622)가 감지되는 경우 해당 A-패킷을 폐기한다(611).

[0044] 단계(630)에서, 상기 CRC8 오류, 상기 CRC32 오류, 상기 MC 값에 따라 상기 A-패킷에 대한 단일 재전송 요청 또는 범위 재전송 요청을 전송한다.

[0045] 상기 CRC8에서 오류(621)가 감지되고, 상기 MC에서 오류(622)가 감지되지 않은 경우 해당 A-패킷에 대한 단일

재전송 요청을 전송한다(631).

[0046] 상기 CRC8에서 오류(622)가 감지되지 않고, 상기 MC에서 오류(623)가 감지되는 경우 해당 A-패킷에 대한 범위 재전송 요청을 전송한다(632).

[0047] 재전송 요청은 단일 재전송 요청과 범위 재전송 요청이 있다. 단일 재전송 요청은 CRC32를 통해 A-패킷의 페이로드에 오류가 발생함을 인지하였을 때 전송되며, 재전송을 요청하는 A-Packet의 MC와 재전송 요청의 오류를 감지하기 위한 CRC로 구성되어 있다. 범위 재 전송 요청은 CRC8 오류 발생 인해 A-패킷이 폐기되는 등, 수신되는 A-패킷들에 누락이 발생되었을 때 전송한다. 범위 재전송 요청은 누락 이전 마지막으로 확인된 A-패킷의 MC와 가장 최근에 수신된 A-패킷의 MC를 사용해 누락된 범위를 나타내고 단일 재전송과 같이 CRC로 구성된다.

[0048] 단계(640)에서, 상기 재전송된 A-패킷을 저장한다.

[0050] 도 7은 본 발명의 일 실시예에 따른 재전송 요청의 구조를 나타내는 도면이다.

[0051] 도 7(a)는 단일 재전송 요청(Single RTS Request)의 구조를 나타내고, 도 7(b)는 범위 재전송 요청(Gap RTS Request)의 구조를 나타낸다.

[0052] CM과 CN은 재전송 요청의 시작과 끝을 표시하기 위해 PCS에서 삽입되는 제어신호이다. 재전송 요청에 의해 TX RTS가 재전송을 할 때, A-패킷 헤더의 Delay 필드를 수정한다. 이는 RX RTS에서 재전송된 A-패킷에 대해 다시 재전송을 요청할 때, 해당 A-패킷의 재전송 횟수와 총 지연시간을 전달해 과도한 재전송을 막는다. 최소 재전송 보장 횟수는 3회 이며, 최대 재전송 지연시간은 기어에 따라 6us에서 12us까지이다.

[0053] 본 발명의 실시예에 따라 설계한 A-PHY RTS 계층과 재전송 기능을 시뮬레이션과 FPGA 구현을 통해 검증한다.

[0055] 도 8은 본 발명의 일 실시예에 따른 A-PHY RTS 검증을 위한 블록도이다.

[0056] 도 8을 참조하면, 재전송 기능 검증을 위한 블록도로 A-PHY의 소스, 싱크와 검증을 위한 모듈들로 구성된다. 검증은 하향링크를 통해 영상 데이터를 전송하고 그 결과를 모니터로 출력하여 확인한다. 검증을 위한 모듈 중 패턴 생성기는 두 가지 패턴의 VGA(640x480) 영상을 60FPS 속도로 생성하여 전송할 데이터를 생성한다. 이 데이터를 가지고 A-PHY에서 사용하는 A-패킷을 만들고 반대로 다시 영상 데이터로 바꾸는 변조 모듈이 구성되어 있으며, 이들은 A-PHY 물리 계층의 Link Layer를 대체한다. A-PHY를 통과한 영상 데이터는 HDMI 출력 모듈을 통해 모니터로 전달되어 데이터 전송 결과를 화면으로 확인할 수 있다. UART To AHB Bridge는 PCS의 Link Startup과 PC 에 연결되며, UART 터미널을 통해 PCS의 시작절차를 제어할 수 있게 한다. 실제 채널을 통해 고속 데이터의 이동 및 재전송 동작을 검증하기 위해, FPGA 보드의 GTX를 이용하여 하향링크의 고속 SerDes 송수신단을 구성했다[7]. 상향링크의 경우 저속에서도 동작하는 직/병렬화 모듈인 PISO(Parallel In Serial Out)와 SIPO(Serial In Parallel Out)을 사용해 구성하였다. 또한 재전송 동작에 의한 효과를 가시적으로 확인하기 위해 A-PHY 싱크의 수신단에 오류 생성 모듈을 추가하였다.

[0058] 도 9는 본 발명의 일 실시예에 따른 CRC 32 오류 시뮬레이션 결과를 나타내는 도면이다.

[0059] 도 9는 RX RTS에서 CRC32 오류가 발생하였을 때의 시뮬레이션이다. RX RTS는 CRC32 오류를 통해 MC가 18인 A-패킷의 페이로드에 문제가 발생함을 인 지하고 상향링크를 통해 단일 재전송 요청을 전송한다.

[0061] 도 10은 본 발명의 일 실시예에 따른 단일 재전송 요청 과정 시뮬레이션 결과를 나타내는 도면이다.

[0062] 도 10(a)는 단일 재전송 요청의 시뮬레이션으로 문 제가 생긴 A-패킷의 MC인 18과 에러를 확인하기 위한 CRC 값인 cd 그리고 전송 데이터의 경계를 표시하는 CM, CN 제어신호로 구성됨을 볼 수 있다. 도 10(b) 은 싱크 RX RTS에 수신된 A-패킷들과 그 상태를 보 여주는 시뮬레이션이다. 처음 입력된 MC가 18인 A-패킷에 오류가 발생했음을 Bad 신호를 통해 확인할 수 있고, 일정 시간 이후 MC가 18인 A-패킷이 재전송 되어 수신됨을 확인할 수 있다.

[0064] 도 11은 본 발명의 일 실시예에 따른 CRC 8 오류 시뮬레이션 결과를 나타내는 도면이다.

[0065] 도 11은 헤더의 오류를 감지하는 CRC8 오류로 인해 A-패킷이 폐기되어 수신된 A-패킷들에 누락이 발생한 시뮬레이션이며, CRC8 오류가 감지되어 MC가 b4인 A-패킷이 폐기된 것을 확인할 수 있다.

[0067] 도 12는 본 발명의 일 실시예에 따른 범위 재전송 요청 과정 시뮬레이션 결과를 나타내는 도면이다.

[0068] RX RTS 는 누락된 A-패킷을 복구하기 위해 도 12(a)와 같은 범위 재전송 요청을 전달한다. 범위 재전송 요청은 누락 이전에 확인된 A-Packet의 MC와 다음으로 수신된 A-Packet의 MC인 b3과 b5 및 CRC 값인 d0, 전송 데이터의 경계를 나타내는 CM, CN으로 구성된다. 누락되었던 MC가 b4인 A-패킷은 도 12(b)와 같이 재전송 되어 RX RTS에 수신된다. FPGA 검증은 Xilinx 사의 KC705 개발 보드[8]를 이 용하였고, 하나의 FPGA에 A-PHY의 소스와 싱크 그 리고 앞서 설명한 패턴 생성기, HDMI 출력 모듈 등의 검증 모듈을 담았다. KC705 보드의 FPGA GTX[9]와 IAM 사의 FMC Loopback 모듈[10]을 이용하여 하향 크와 상향링크 채널을 포함한 SerDes 환경을 구성하였다. FPGA와 PC를 UART를 통해 연결하고 UART 터미널을 통해 A-PHY 시작 절차를 직접 제어하였다. 마지막으로 모니터를 HDMI로 연결하여 A-PHY 출력 결과를 확인하였다.

[0070] 도 13은 본 발명의 일 실시예에 따른 A-PHY RTS 검증을 위한 FPGA 개발 환경을 나타내는 도면이다.

[0071] 도 13은 이러한 검증 환경을 나타낸다. 이때 오류 삽입 모듈의 동작과 재전송 동작을 별 도의 스위치로 제어 가능하게 구성하여 재전송 동작을 확인하였다.

[0073] 도 14는 본 발명의 일 실시예에 따른 A-PHY RTS 검증 결과를 나타내는 도면이다.

[0074] 도 14는 A-PHY를 통과한 영상 데이터가 모니터를 통해 출력되는 모습이다. 도 14(a)는 오류 삽입과 재전송 기능을 둘 다 켜지 않은 상태로 패턴 생성기에 서 만들어지는 영상이 모니터로 출력되는 것을 나타낸다. 도 14(b)는 오류 삽입 모듈을 동작시킨 상태로 삽입된 오류에 의해 출력 영상에 노이즈와 깨짐이 발생 함을 볼 수 있다. 이때 재전송 기능을 동작시키면 오류 가 복원되어 정상적인 영상이 출력됨을 도 14(c)을 통해 확인할 수 있다.

[0076] <표 2>

[0077]

		A-PHY Source	A-PHY Sink	Total
Logic Utilization	LUT	1,510 (<1%)	2,414 (1%)	3,924
	Logic Register	792 (<1%)	1,227 (<1%)	2,019
	Block Memory	66 (15%)	66 (15%)	132
	DSP	0 (0%)	0 (0%)	0
Maximum Frequency		200MHz		

[0078] 표 2는 RTS 계층을 포함한 A-PHY 소스와 싱크의 FPGA 합성 결과를 보여준다. FPGA 합성 결과 LUT는 3,924개, 레지스터는 2,019개로 1% 이하의 적은 사용량을 가지고, 블록메모리는 132개로 재전송 동작에서 A-패킷을 저장하기 위해 30% 정도의 자원이 사용됨을 확인할 수 있다. DSP는 사용되지 않았으며, 설계된 시스템은 최대 200MHz의 주파수에서 동작함을 확인하였다.

[0080] 본 발명의 실시예에 따른 차량용 SerDes 표준인 MIPI A-PHY 에서 새롭게 제시된 물리계층인 RTS 계층을 설계하고 검증하였다. 본 발명에서 설계한 RTS 계층은 재전송을 위해 패킷을 저장하는 버퍼, 재전송을 위해 A-패킷에

값을 삽입을 하는 모듈들, A-패킷의 상태를 읽고 필요에 따라 재전송을 요청하는 모듈들로 구성되어 있다. 설계한 RTS 계층이 A-PHY 표준의 요구사항에 따라 수신된 패킷의 상태에 맞춰 단일, 범위 재전송 동작이 가능하도록 설계하였고 시뮬레이션을 통해 동작을 확인 하였다. 설계된 RTS 계층의 검증을 위한 FPGA 통합 환경을 구축하였고 동작을 확인하였다.

[0082] 이상에서 설명된 장치는 하드웨어 구성요소, 소프트웨어 구성요소, 및/또는 하드웨어 구성요소 및 소프트웨어 구성요소의 조합으로 구현될 수 있다. 예를 들어, 실시예들에서 설명된 장치 및 구성요소는, 예를 들어, 프로세서, 콘트롤러, ALU(arithmetic logic unit), 디지털 신호 프로세서(digital signal processor), 마이크로컴퓨터, FPGA(field programmable gate array), PLU(programmable logic unit), 마이크로프로세서, 또는 명령(instruction)을 실행하고 응답할 수 있는 다른 어떠한 장치와 같이, 하나 이상의 범용 컴퓨터 또는 특수 목적 컴퓨터를 이용하여 구현될 수 있다. 처리 장치는 운영 체제(OS) 및 상기 운영 체제 상에서 수행되는 하나 이상의 소프트웨어 애플리케이션을 수행할 수 있다. 또한, 처리 장치는 소프트웨어의 실행에 응답하여, 데이터를 접근, 저장, 조작, 처리 및 생성할 수도 있다. 이해의 편의를 위하여, 처리 장치는 하나가 사용되는 것으로 설명된 경우도 있지만, 해당 기술분야에서 통상의 지식을 가진 자는, 처리 장치가 복수 개의 처리 요소(processing element) 및/또는 복수 유형의 처리 요소를 포함할 수 있음을 알 수 있다. 예를 들어, 처리 장치는 복수 개의 프로세서 또는 하나의 프로세서 및 하나의 콘트롤러를 포함할 수 있다. 또한, 병렬 프로세서(parallel processor)와 같은, 다른 처리 구성(processing configuration)도 가능하다.

[0083] 소프트웨어는 컴퓨터 프로그램(computer program), 코드(code), 명령(instruction), 또는 이들 중 하나 이상의 조합을 포함할 수 있으며, 원하는 대로 동작하도록 처리 장치를 구성하거나 독립적으로 또는 결합적으로(collectively) 처리 장치를 명령할 수 있다. 소프트웨어 및/또는 데이터는, 처리 장치에 의하여 해석되거나 처리 장치에 명령 또는 데이터를 제공하기 위하여, 어떤 유형의 기계, 구성요소(component), 물리적 장치, 가상 장치(virtual equipment), 컴퓨터 저장 매체 또는 장치에 구체화(embody)될 수 있다. 소프트웨어는 네트워크로 연결된 컴퓨터 시스템 상에 분산되어서, 분산된 방법으로 저장되거나 실행될 수도 있다. 소프트웨어 및 데이터는 하나 이상의 컴퓨터 판독 가능 기록 매체에 저장될 수 있다.

[0084] 실시예에 따른 방법은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 실시예를 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(magnetic media), CD-ROM, DVD와 같은 광기록 매체(optical media), 플롭티컬 디스크(floptical disk)와 같은 자기-광 매체(magneto-optical media), 및 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다.

[0085] 이상과 같이 실시예들이 비록 한정된 실시예와 도면에 의해 설명되었으나, 해당 기술분야에서 통상의 지식을 가진 자라면 상기의 기재로부터 다양한 수정 및 변형이 가능하다. 예를 들어, 설명된 기술들이 설명된 방법과 다른 순서로 수행되거나, 및/또는 설명된 시스템, 구조, 장치, 회로 등의 구성요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나, 다른 구성요소 또는 균등물에 의하여 대치되거나 치환되더라도 적절한 결과가 달성될 수 있다.

[0086] 그러므로, 다른 구현들, 다른 실시예들 및 특허청구범위와 균등한 것들도 후술하는 특허청구범위의 범위에 속한다.

[0088] <참고문헌>

[0089] [1] A. Legault, J. Rodriguez and J. Endo, "Meeting the Needs of Next-Generation Displays with a High-Performance MIPI DSI-2 Subsystem Solution", MIPI Devcon 2021, Virtual. https://www.mipi.org/devcon/2021/agenda/meeting-needs-next-generation-displays-high-performance-mipi-dsi-2-subsystem-solution, accessed 7 Feb 2022.

[0090] [2] W. Zeng, M. A. S. Khalid and S. Chowdhury,"In-Vehicle Networks Outlook: Achievements and Challenges," IEEE Communications survey&tutorials, vol.18, no.3, pp.1552-1571, 2016.

[0091] [3] IEEE Standard for Ethernet—Amendment 8:Physical Layer Specifications and Management Parameters for 2.5 Gb/s, 5 Gb/s, and 10 Gb/s Automotive Electrical Ethernet, IEEE Std 802.3ch-2020, June. 2020.

[0092] [4] H. Zinner, "Automotive Ethernet and SerDes in Competition", ATZ Electron Worldwide, vol.15, pp.40-43, 2020.

[0093] [5] A. A. Varghese, "Automotive In-Vehicle Networks - Ethernet or SERDES? Well....It Is Complicated", Keysight Blogs, Jan 2021. https://blogs.keysight.com/blogs/inds.entry.html/2021/01/28/automotive_in-vehicl-uj3D.html, accessed 11 Feb 2022.

[0094] [6] ASA, Automotive SerDes Alliance, 2022,https://auto-serdes.org, accessed 11 Feb 2022.

[0095] [7] IEEE Standard for Adoption of MIPI AllianceSpecification for A-PHY Interface (A-PHY)Version 1.0, IEEE Std 2977-2021, June 2021.

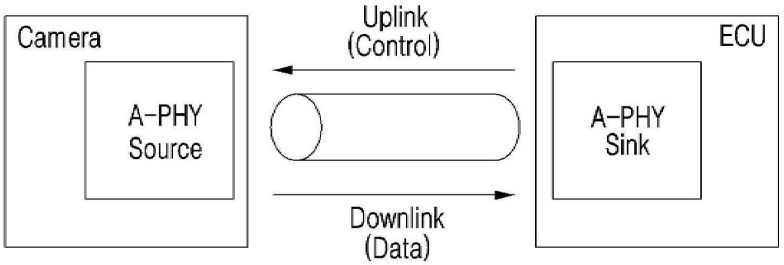
[0096] [8] Xilinx, "KC705 Evaluation Board for the Kintex-7 FPGA", Aug 2018, https://www.xilinx.com/support/documentation/boards_and_kits/kc705/ug810_KC705_Eval_Bd.pdf, accessed 22 June 2021.

[0097] [9] Xilinx, "Xilinx 7 Series FPGAs GTX/GTH Transceivers User Guide (UG476)", Aug 2018,https://www.xilinx.com/support/documentation/user_guides/ug476_7Series_Transceivers.pdf,accessed 20 Dec 2021.

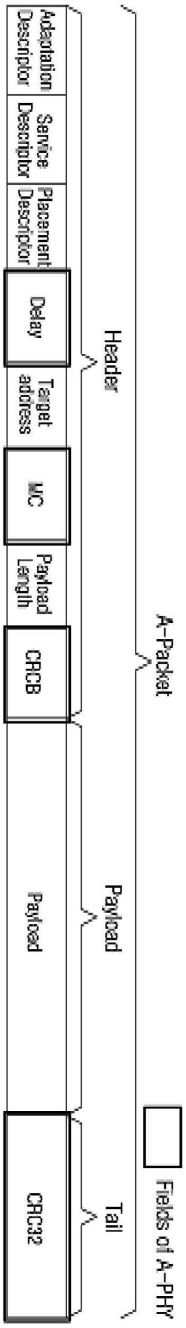
[0098] [10] IAM Electronic, "FMC Loopback Module", Apr2019, https://fmchub.github.io/projects/FMC_LOOPBACK/Datasheet/FMC_LOOPBACK_MODULE_datasheet.html, accessed 20 Dec 2021.

도면

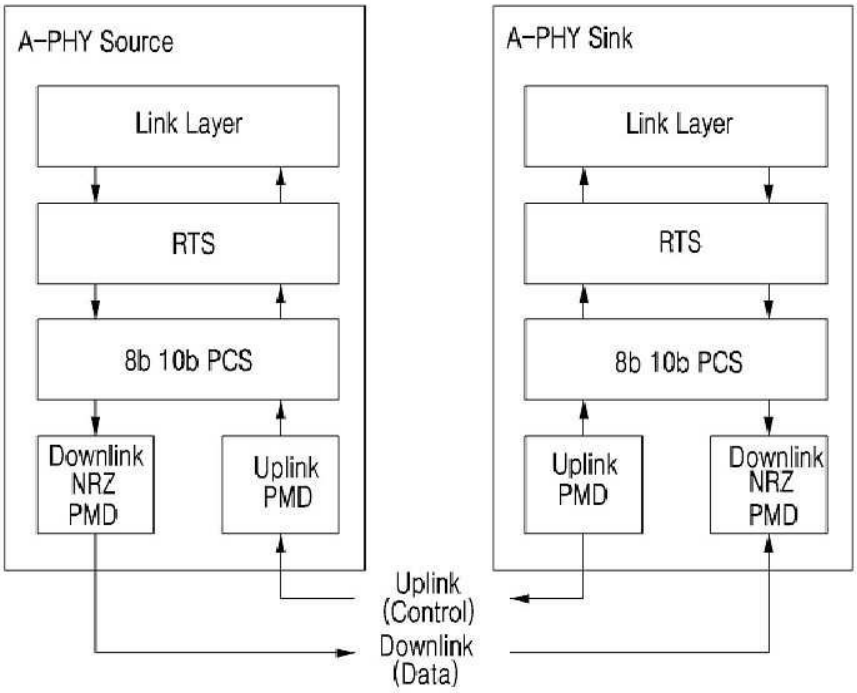
도면1



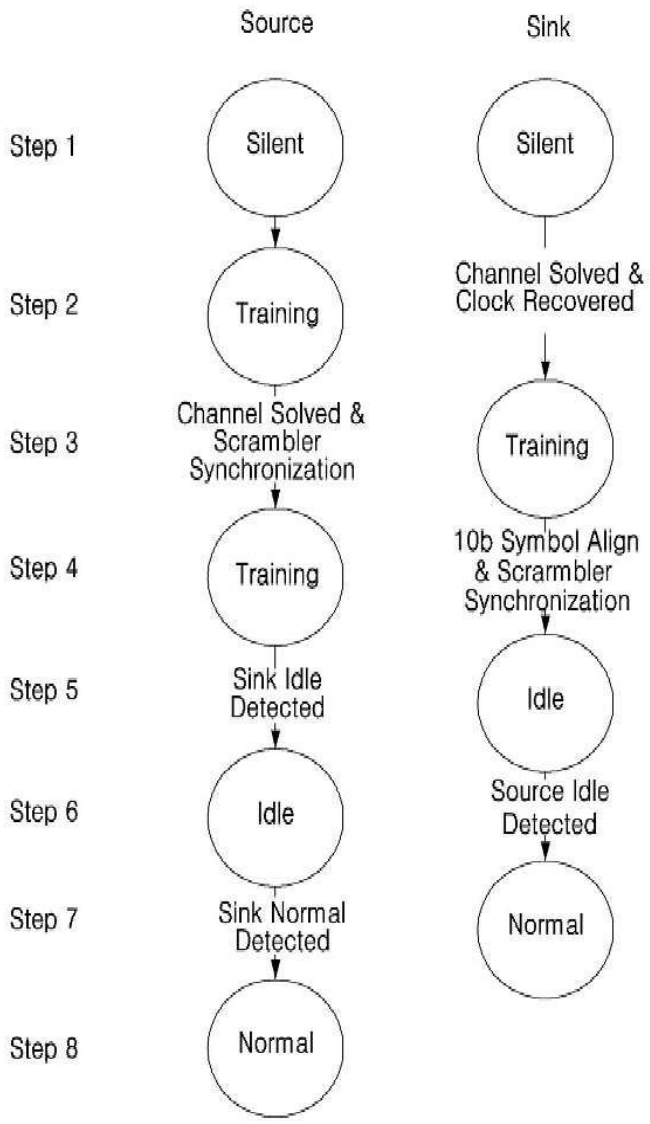
도면2


Adaptation Descriptor
Service Descriptor
Placement Descriptor
Delay
Target address
MC
Payload Length
CRCB
Payload
CRC32
Header
Payload
Tail
A-Packet
Fields of A-PHY

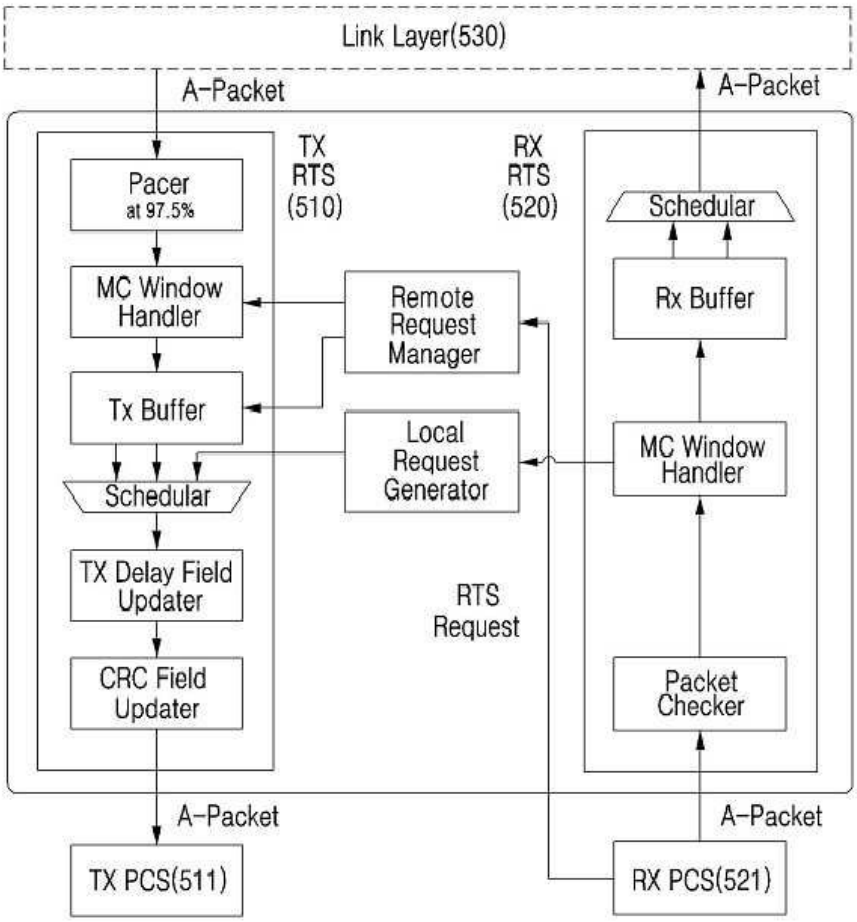
도면3


A-PHY Source
Link Layer
RTS
8b 10b PCS
Downlink NRZ PMD
Uplink PMD
A-PHY Sink
Link Layer
RTS
8b 10b PCS
Uplink PMD
Downlink NRZ PMD
Uplink (Control)
Downlink (Data)

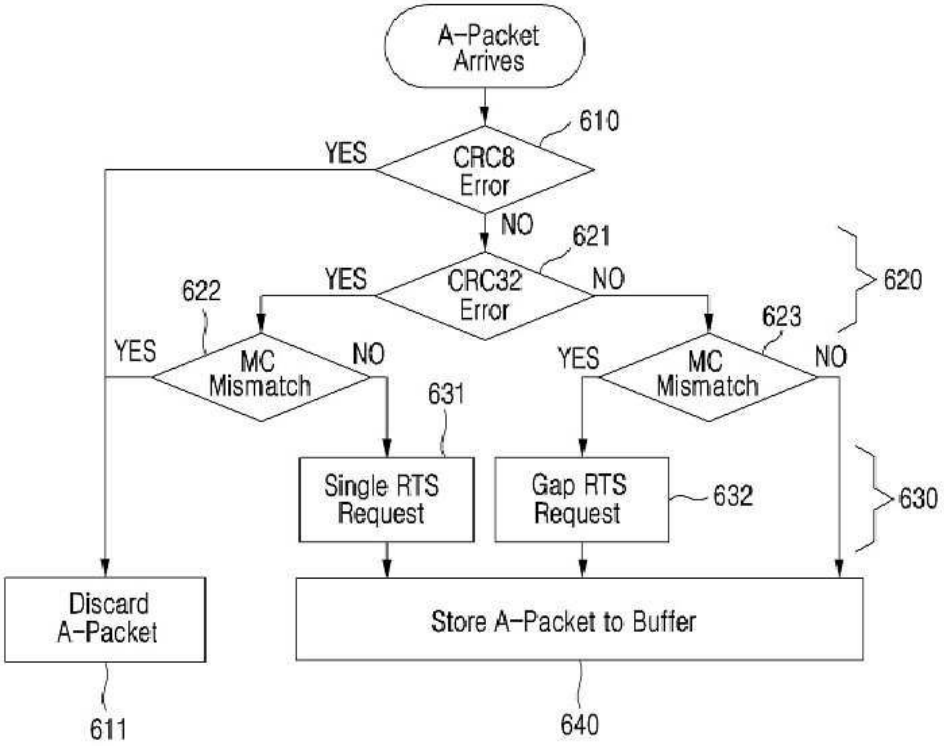
도면4


Source
Sink
Step 1
Silent
Silent
Step 2
Training
Channel Solved &
Clock Recovered
Step 3
Channel Solved &
Scrambler
Synchronization
Training
Step 4
Training
10b Symbol Align
& Scrarmbler
Synchronization
Step 5
Sink Idle
Detected
Idle
Step 6
Idle
Source Idle
Detected
Step 7
Sink Normal
Detected
Normal
Step 8
Normal

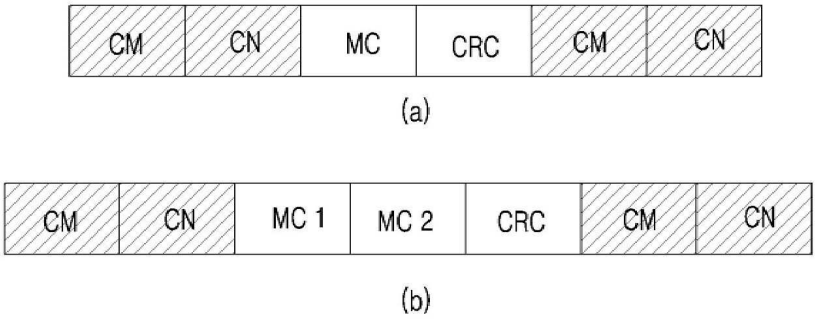
도면5


Link Layer(530)
A-Packet
A-Packet
TX
RTS
(510)
RX
RTS
(520)
Pacer
at 97.5%
MC Window
Handler
Tx Buffer
Schedular
TX Delay Field
Updater
CRC Field
Updater
Remote
Request
Manager
Local
Request
Generator
RTS
Request
Schedular
Rx Buffer
MC Window
Handler
Packet
Checker
A-Packet
A-Packet
TX PCS(511)
RX PCS(521)

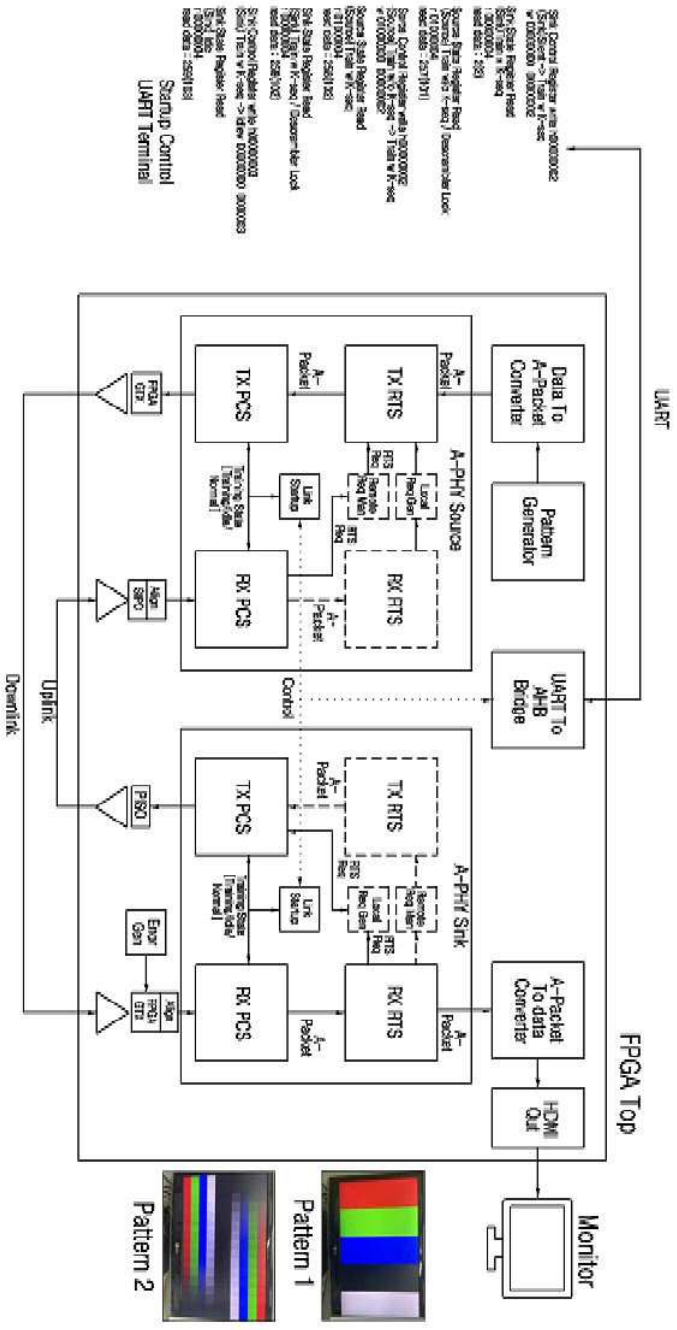
도면6


A-Packet Arrives
610
CRC8 Error
YES
NO
621
CRC32 Error
YES
NO
620
622
623
MC Mismatch
YES
NO
MC Mismatch
YES
NO
631
Single RTS Request
Gap RTS Request
632
630
Discard A-Packet
Store A-Packet to Buffer
611
640

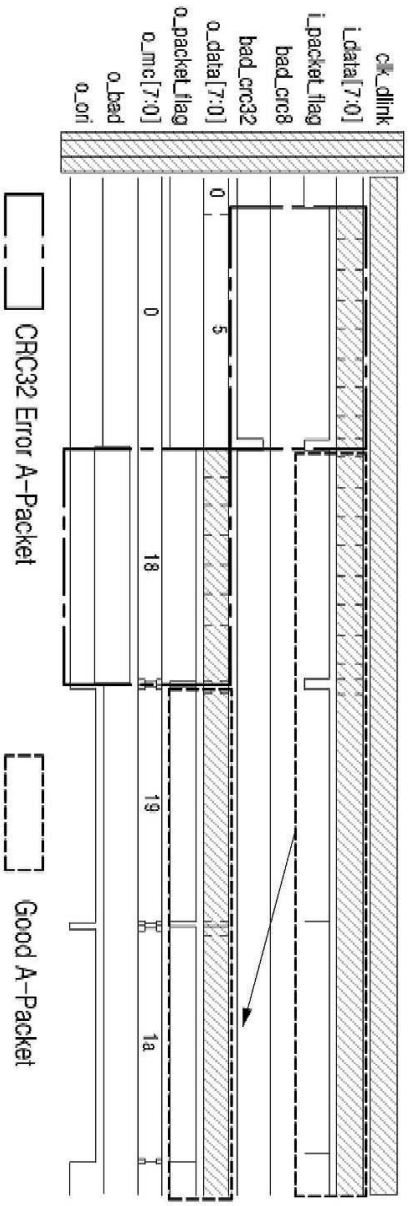
도면7


CM
CN
MC
CRC
CM
CN
(a)
CM
CN
MC 1
MC 2
CRC
CM
CN
(b)

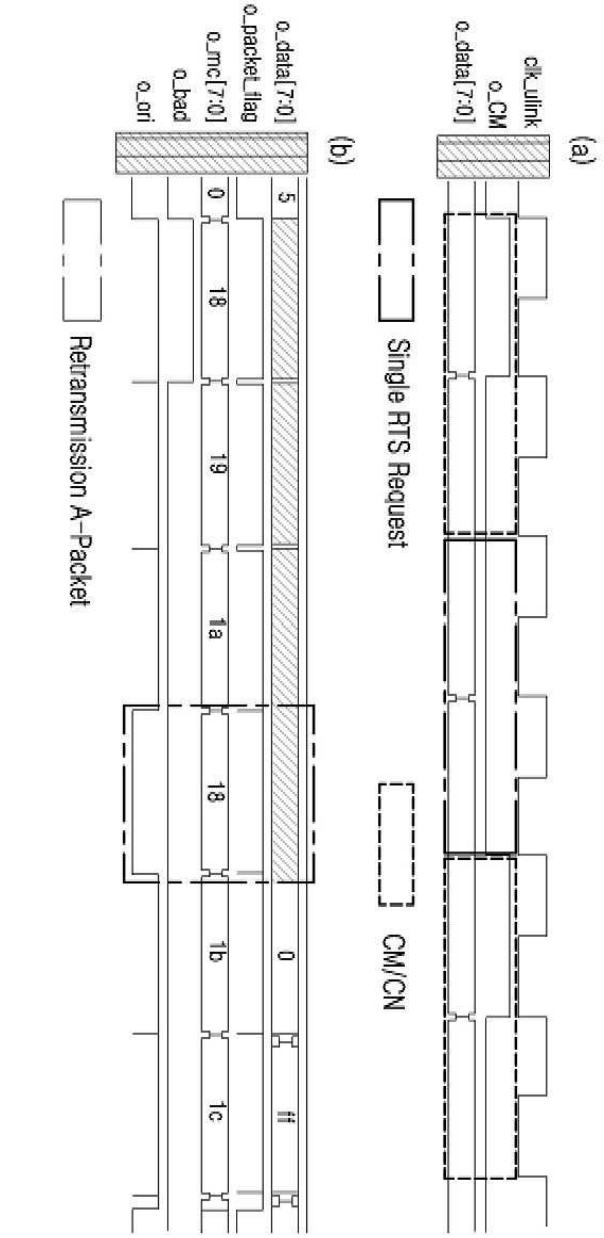
도면8


Startup Control
UART Terminal
FPGA Top
UART
Data To
A-Packet
Converter
Pattern
Generator
UART To
AHB
Bridge
A-PHY Source
TX RTS
RX RTS
TX PCS
RX PCS
A-PHY Sink
TX RTS
RX RTS
TX PCS
RX PCS
A-Packet
To data
Converter
HDMI
Out
Control
Uplink
Downlink
Error
Gen
Monitor
Pattern 1
Pattern 2

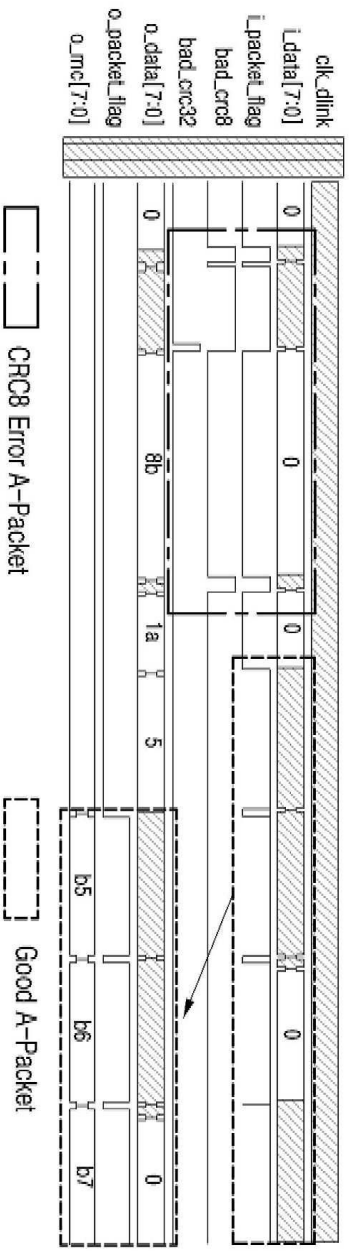
도면9


clk_dlink
i_data[7:0]
i_packet_flag
bad_crc8
bad_crc32
o_data[7:0]
o_packet_flag
o_mic[7:0]
o_bad
o_ori
0
5
0
18
19
1a
CRC32 Error A-Packet
Good A-Packet

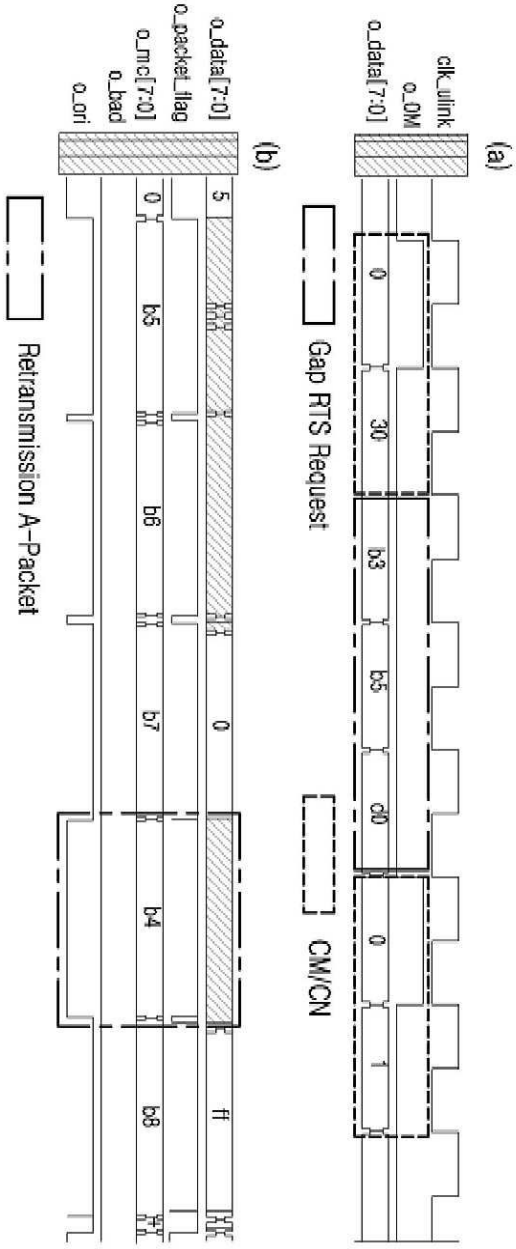
도면10


(a)
clk_ulink
o_CM
o_data[7:0]
Single RTS Request
CM/CN
(b)
o_data[7:0]
o_packet_flag
o_mc[7:0]
o_bad
o_ori
5
0
18
19
1a
18
0
1b
ff
1c
Retransmission A-Packet

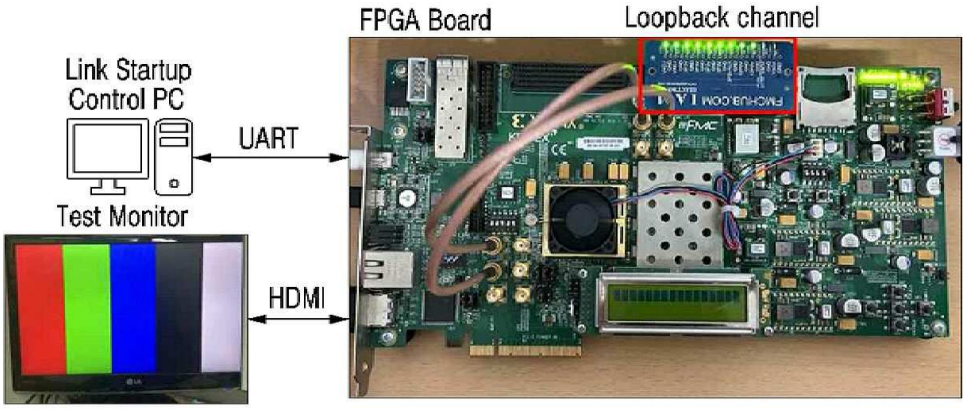
도면11


clk_dlink
i_data[7:0]
i_packet_flag
bad_crc8
bad_crc32
o_data[7:0]
o_packet_flag
o_mc[7:0]
0
0
0
0
8b
1a
5
0
b5
b6
b7
CRC8 Error A-Packet
Good A-Packet

도면12


(a)
clk_ulink
o_OM
o_data[7:0]
0
30
b3
b5
c0
0
1
Gap RTS Request
CM/CN
(b)
o_data[7:0]
o_packet_flag
o_mc[7:0]
o_bad
o_ori
5
0
b5
b6
b7
0
b4
ff
b8
Retransmission A-Packet

도면13


FPGA Board
Loopback channel
Link Startup
Control PC
UART
Test Monitor
HDMI

도면14


(a)
(b)
(c)