

디스플레이 포트를 위한 고속 보조 채널 설계

Fast Auxiliary Channel Design for Display Port

진 현 배*, 문 용 환*, 장 지 훈*, 김 태 호*, 송 병 철*, 강 진 구**

Hyun-Bae Jin*, Yong-Hwan Moon*, Ji-Hoon Jang*, Tae-Ho Kim*, Byung-Cheol Song*,
Jin-Ku Kang**

Abstract

This paper presents the design of a fast auxiliary channel bus for DisplayPort 1.2 interface. The fast auxiliary channel supports Manchester transactions at 1Mbps and fast auxiliary transactions at 780Mbps. The Manchester transaction is used for managing the main link and auxiliary channel and the fast auxiliary transaction is for data transfer via the auxiliary channel. Simplified serial bus architecture is proposed to be implemented in fast auxiliary channel. The fast auxiliary channel transmitter and receiver are implemented with 7,648 LUTs and 6,020 slice register synthesized in Xilinx Vertex4 FPGA and can be operated at 72MHz to support 720Mbps.

요 약

본 논문은 디스플레이포트의 보조채널에서 고속 데이터 전송을 할 수 있는 고속 양방향 보조 채널을 구성하기 위한 새로운 송·수신기 구조를 제안하고 적용에 대해 서술하였다. 제안된 고속 보조 채널은 저속 전송에서 맨체스터 인코딩을 사용하여 1Mbps대역폭을, 고속 전송에서 8B/10B인코딩 방식을 사용하여 720Mbps의 대역폭을 지원한다. 맨체스터 전송을 사용하여 고속 보조채널 및 메인링크의 링크 서비스 및 디바이스 서비스를 위한 저속 보조 채널 블록을 제안하고, 8B/10B인코딩 방식을 통하여 보조채널을 통한 고속 데이터 전송을 위한 블록을 제안한다. 또한 데이터 패킷 구조와 데이터 전송방식에 대하여 정의하였다. 설계된 시스템은 Verilog HDL로 설계 되었으며, 고속 보조채널 송·수신기는 Xilinx Vertex4 FPGA를 사용하여 합성한 결과 7,648개의 LUTs와 6,020개의 registers를 사용 하였으며, 최대 동작 속도는 203MHz의 성능을 확인 하였다.

Key words : DisplayPort, 보조채널(auxiliary channel), Link training, DisplayPort Configuration Data(DPCD), Extended Display Identification Data(EDID)

1. 서론

최근 고해상도 영상 기술의 발달에 발맞추어 새로운 고속 디스플레이 인터페이스 표준이 발표되고 있다. 현재의 디스플레이에는 주로 LVDS, VGA, DVI, HDMI 인터페이스 방식이 사용되고 있다[1]-[3].

LVDS는 주로 노트북에 사용되고 VGA아날로그 인터페이스나 DVI 기술은 PC와 디스플레이 장치와의 영상 전송을 위해 사용되며, PC와 디스플레이 장치를 연결할 경우 음성 케이블의 별도 연결이 요구 된다. 이러한 현재의 기술들은 설치상의 불편함이 존재할 뿐만 아니라 영상의 훼손 및 확장성의 문제가 존재한다.

디스플레이에 필요한 영상 데이터 전송 외에, CAM(Camera Module), USB, Interface Unit (Keyboard, Mouse, Touch Screen)등의 전송요구가 증가하고 있다[4]. 모니터에 연결된 다양한 장치들의 데이터를 PC등의 단말기로 전송할 때 영상 데이터를 위한 케이블 외에 부수적인 데이터의 전송을 위한 케이블이 요구된다. 케이블의 증가로 인하여 많은 포트가 증가되며, 디스플레이 장치와 PC 및 단말기 사이의 복잡도가 증가된다. 이러한 문제를 해결하기 위해

* 인하대학교 전자공학부
(School of Electronics Engineering, Inha University)

★ 인하대학교 전자공학부 교수

※ 감사의 글 (Acknowledgment)

본 연구는 ETRI 시스템반도체 진흥센터의 SoC융합
인력양성사업 지원으로 수행됨.

接受日:2011年 06月 1日, 修正完了日: 2011年 06月 30日

최근 USB 혹은 HDMI의 CEC(Consumer Electronics Control)등의 방법이 해결책으로 나오고 있다[1],[4].

디스플레이포트는 차세대 디스플레이 인터페이스로서 PC, 모니터, 패널, 프로젝터 및 고해상도 콘텐츠 응용 프로그램 등에 광범위하게 사용되는 연결 솔루션으로 개발되었다[5]-[9]. 디스플레이 포트는 크게 소스 디바이스와 싱크 디바이스로 구성되어 있으며, 메인 링크, 보조 채널, HPD(Hot Plug Detect)신호 라인으로 소스 디바이스와 싱크 디바이스의 물리적 접촉을 가능하게 한다. 메인 링크는 영상 및 음성데이터를 소스 디바이스에서 싱크 디바이스로 전송시키며 대역폭이 높고 호출 시간이 짧은 채널이다. 메인링크의 최대 대역폭은 21.6Gbps를 지니며, 이는 3840x1600x30BPP(Bit Per Pixel)의 영상 및 2560x1600의 3D영상의 데이터를 전송 가능하게 한다. 또한, 디스플레이포트의 마이크로-패킷 구조를 사용하여 다수의 모니터의 영상 및 음성 데이터가 하나의 커넥터를 통해 전송 가능하다.

양방향 보조 채널(Auxiliary Channel)은 1Mbps와 720Mbps의 대역폭을 사용 가능하며, 링크 및 장치의 제어의 관리와 양방향 고속 데이터 전송을 가능하게 한다. 링크 트레이닝 과정을 통한 차동 전압 폭의 설정 및 등화기 최적화를 통하여 저 절전의 데이터 전송 및 데이터 전송 최적화가 가능하며, 고속의 데이터 전송을 통하여 하나의 디스플레이포트 케이블을 통한 USB2.0, CAM 및 Interface Unit, 혹은 제어신호의 전송이 가능하다.

본 논문은 디스플레이포트 인터페이스 기반의 보조 채널에서 고속 데이터 전송을 할 수 있는, 고속 양방향 보조 채널을 구성하기 위한 새로운 송·수신기 구조를 제안하고 적용 방법 대하여 설명한다. 고속 양방향 통신을 하기 위한 송·수신기는 하나의 직렬 버스 역할을 하기 때문에, 각각의 개별적인 모듈이 하나의 직렬 버스를 점유할 수 있는 중재 방식 및 스케줄링 방식에 대하여 설명한다. 각각의 중재를 하기 위한 중재기(arbiter), 대역폭 제어기(bandwidth controller) 및 고속 보조 채널을 위한 하나의 고속 직렬 버스 방식이 제안되었다. 또한 고속 보조채널과 메인링크 사이의 링크 관리 및 장치 제어를 위한 링크 서비스 및 디바이스 서비스를 위한 보조 채널 블록을 제안한다. 버스채널을 위한 데이터 패킷 구조와 데이터 전송방식에 대하여 정의하였다.

II. 고속 보조채널 설계

1. 고속 보조채널의 동작 절차

제안된 고속 보조채널은 Manchester II 인코딩을 사용하여 1Mbps를, 8B/10B 인코딩을 사용하여 720Mbps의 대역폭을 지닌다. ManchesterII 인코딩 방식은 디스플레이 포트를 통한 데이터 전송 여부 판단 및 링크의 관리를 위해 사용되고, 8B/10B 인코딩은 보조채널을 통한 고속 데이터 전송을 위해 사용된다 [10]. 모든 데이터 전송과정은 상위 계층에 해당되는 전송기(source)에 의해 제어되며 하위 계층에 해당되는 수신기(sink)는 전송기 상태를 따른다.

고속 보조채널의 동작 절차는 다음과 같다.

- (1) 장치 확인 상태
- (2) 메인링크 트레이닝 상태
- (3) 메인링크 정상 동작 상태
- (4) 고속 보조채널 트레이닝 상태
- (5) 고속 보조채널 정상 동작 상태

첫 번째 상태인 장치확인 상태에서 보조 채널은 수신기(sink)의 EDID로부터 모니터 장치에 관련된 정보를 읽어, 장치를 파악한다. 수신기의 장치를 확인하여, 메인링크를 통해 보내질 수 있는 데이터를 최적화 한다. 메인링크 트레이닝 상태에서 보조채널은 메인링크의 전송을 최적화하기 위한 링크 트레이닝을 진행한다. 메인링크 트레이닝 구간 동안 차동 전압폭 및 등화기의 상태가 최적화 된다. 고속 보조 채널의 트레이닝은 8B/10B 인코딩된 데이터가 보조채널을 통해 전송될 수 있도록 메인링크와 같은 트레이닝 과정을 거쳐 링크 트레이닝이 이루어지며, 양방향 통신을 위해 정 방향 및 역 방향 링크 트레이닝을 수행한다. 각각의 링크 트레이닝 상태가 성공적으로 끝나면 메인링크, 고속 보조채널을 통해 고속의 데이터가 전송된다.

2. 고속 보조 채널 송신기

그림 1은 제안된 고속 보조 채널의 송신기를 보여준다. 제안된 고속 보조채널 송신기는 Manchester encoding 방식을 사용하여 1Mbps대역폭을 가지는 저속 보조채널 블록과, 8B/10B encoding 방식을 사용하여 720Mbps의 대역폭을 가지는 고속 직렬 버스 블록으로 구성되어 있다. 한 개의 인코딩 방식이 하나의 보조채널에 사용될 수 있도록 고속 보조채널과 저속 보조채널을 선택적으로 사용할 수 있는 제어신호가 사용된다.

저속 보조채널 블록은 링크 폴리시 메이커(link policy maker), 스트림 폴리시 메이커(stream policy maker), source FSM, 인포-프레임 데이터 발생기(infoFrame data generator), 메인 스트림 속성 데이

터 발생기(main stream attribute generator), 맨체스터 인코더/디코더(manchester encoder/decoder)로 구성 되어 있다. 링크 폴리시 메이커는 메인 링크 및 보조채널의 구성 및 유지를 위한 링크 트레이닝 과정을 수행 하며, 스트림 폴리시 메이커는 EDID 읽기 동작을 통하여 수신부의 스트림 폴리시 메이커와 EDID로부터 디스플레이 장치에 관련된 정보를 읽어 온 후, EDID정보와 메인 스트림 속성 데이터, 인포-프레임 데이터 발생기의 정보를 비교하여 스트림 데이터의 전송여부를 판단하는 장치 서비스를 수행한다. 장치 서비스 이후 링크 폴리시 메이커가 동작하며 링크 폴리시 메이커는 DPCD의 데이터를 읽어, 링크 트레이닝을 시작한다. 링크 트레이닝을 통하여 메인링크의 상태를 최적화 한다.

고속 보조채널은 반이중 양방향 데이터 전송 채널이기 때문에, 송신기와 수신기 사이의 전송 및 수신 시간을 할당하여 서로간의 충돌을 방지하기 위해 source FSM블록이 설계되었다. Manchester encoder/decoder 블록은 저속 데이터 전송의 인코딩을 위해 사용된다. 고속 직렬 버스 블록은 FAUX serializer/deserializer, 8B/10B encoder/decoder, FAUX packer/unpacker, Master source 블록, 대역폭 제어기(bandwidth controller), 중재기(arbiter), FIFO 블록들로 구성되어 있다. 고속 직렬 버스 블록은 고속 데이터 전송을 위해 사용된다.

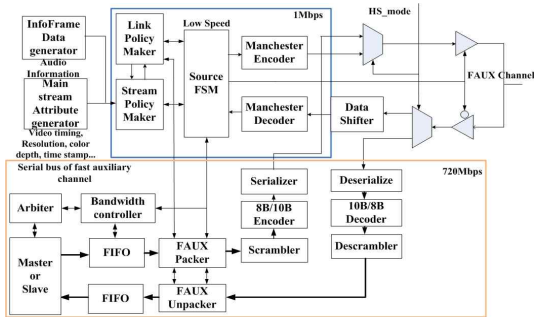


Fig. 1. The detail structure of transmitter for fast auxiliary channel

그림 1. 제안된 디스플레이포트 고속 보조 채널 송신기 세부구조

3. 고속 보조 채널 수신기

고속 보조채널을 위한 수신기의 구조는 그림 2와 같다. 구조는 보조채널의 송신기와 반대구조로 구성되어 있고, 인포-프레임 데이터 발생기(InfoFrame data generator), 메인 스트림 속성 데이터 발생기(Main stream attribute generator)대신 DPCD블록과 EDID 블록이 존재한다. DPCD블록은 하나의 RAM으로서,

수신기의 성능 및 디스플레이포트 링크의 상태 정보를 저장한다. EDID는 ROM으로서 사용되며, 모니터의 제조사, 제조년월일, 시리얼번호, 지원해상도를 저장한다. 수신기의 sink FSM블록은 송신기의 source FSM블록의 명령에 대응하는 슬레이브 역할을 수행한다.

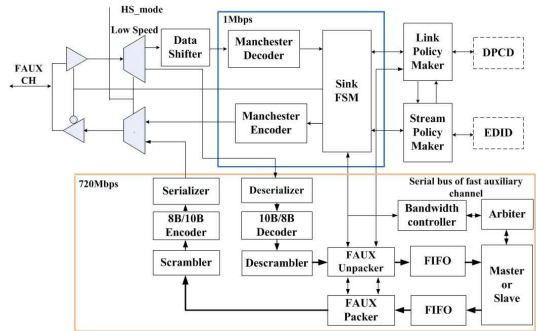


Fig. 2. The detail structure of receiver for fast auxiliary channel

그림 2. 제안된 디스플레이포트 고속 보조 채널 수신기 세부구조

III. 고속 보조채널을 위한 직렬 버스 구조

1. 고속 직렬 버스 송신기

그림 3은 제안된 양방향 고속 직렬 버스의 송신기의 구조로서 중재기(arbiter), source FSM, 대역폭 제어기(bandwidth controller), packer/unpacker, 8B/10 encoder/decoder, FIFO, 마스터(master)블록들로 구성된다. 제안된 고속 직렬 버스는 AMBA버스 구조와 유사한 구조로 구성되며 모든 동작은 송신기의 중재기(arbiter)에 의해 결정된다[11]-[17]. 중재기(arbiter)블록은 각각의 마스터들의 버스 점유 요청에 대한 중재역할을 한다. 중재 방식은 라운드 로빈(round robin)방식이 사용되었고, 대역폭 제어기와 연동되어 사용된다. 마스터(master)블록들은 자기 독립된 개체로서 시스템의 슬레이브(slave)들에게 읽기 혹은 쓰기 동작을 요청한다. 수신기의 마스터(master)블록은 마스터 소스(master source)블록, 마스터 싱크(master sink)블록들로 구성되어 있다. 송신기 내부의 마스터 싱크(master sink)블록들은 수신기에 해당하는 마스터 싱크(master sink)의 터미 블록으로써, 수신기(sink)의 master들의 제어신호를 데이터의 트랜잭션 때마다 받아 중재기(arbiter)로 수신기의 마스터들의 현재 상태를 전달해 주는 역할을 한다. 마스터 소스(master source)블록 들은 중재기(arbiter)로 현재의 상태를 즉각적으로 알려준다.

대역폭 제어기(bandwidth controller)블록은 FIFO에 저장되어 있는 데이터의 개수를 추정하여, 패킷화된 데이터를 스케줄링 및 전송될 데이터의 길이를 결정한다. 전송될 데이터의 길이 정보는 중재기(arbiter)와 packer로 전달되어 스케줄링 및 패킷화를 도와준다. source FSM블록은 패킷의 구조와 패킷의 방향을 결정하며 packer블록을 제어한다.

고속 직렬 버스의 동작은 버스의 점유를 원하는 마스터(master)블록이 요청(request)신호를 통해 중재기(arbiter)블록으로 요청하면, 요청한 마스터(master)블록은 중재기(arbiter)블록으로부터 허가(grant)신호를 기다리게 된다. 허가(grant)신호를 받은 마스터 블록은 데이터의 읽기 및 쓰기 동작을 시작한다.

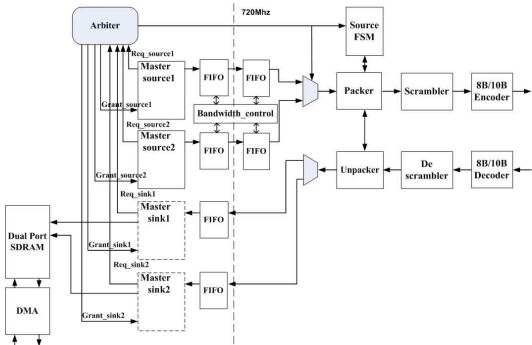


Fig. 3. Detail structure of transmitter for fast serial bus
그림 3. 고속 직렬 버스 송신기 구조

2. 고속 직렬 버스 수신기

그림 4는 고속 직렬 버스의 수신기(sink)의 블록을 보여준다. 고속 직렬 버스 수신기의 구조는 중재기(arbiter)블록을 제외하면 송신기와 유사한 구조를 지닌다. 수신기의 모든 마스터(master)들도 자기 독립된 개체로서 시스템의 슬레이브(slave)들에게 읽기 쓰기 동작을 요청할 수 있다. 각각의 마스터들은 현재 상태를 Sink FSM으로 전달하여 고속 보조채널을 통한 수신기에서 송신기로 패킷이 전송되는 트랜잭션 과정에서 패킷의 마지막에 각각의 마스터의 상태를 수신기의 마스터 싱크(master sink)블록으로 전달하여 각각의 수신기의 마스터들의 요청상태를 전달한다. 송신기의 각각의 마스터 싱크(master sink)블록들은 수신기의 마스터들의 요청신호를 중재기(arbiter)블록으로 전달하여 수신기의 마스터들의 요청상태를 알린다.

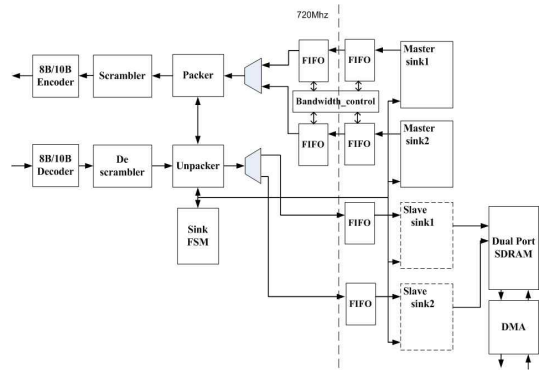


Fig. 4. Detail structure of receiver for fast serial bus
그림 4. 고속 직렬 버스 수신기 구조

3. 고속 보조채널을 위한 패킷 구조와 트랜잭션

그림 5는 보조채널이 고속 보조채널로 사용될 때의 패킷구조를 보여준다. 패킷은 프리앰블(preamble), 페이로드 데이터 영역, 시작, 종료기호로 구성된다. 프리앰블은 K28.7(0011111000)(2), D10.2(0101010101)(2), K28.5(0011111010)(2)의 특수 기호로 구성되어 있으며 12번 특수 기호가 반복된다. 처음 두 개의 K28.7기호는 신호가 없으면 데이터를 차단, 데이터가 전송되면 데이터를 통과 시켜 데이터가 없을 때의 잡음을 제거하는 수신기의 squelch detector를 동작시키는 용도로 사용된다. D10.2특수 코드는 클록과 같은 패턴으로써, CDR의 위상 고정(lock)을 유도한다. K28.5특수 코드는 데이터를 고정(lock)시키기 위한 콤파시호로 사용된다.

그림 5의 페이로드 데이터(payload data)영역은 그림 6과 같은 구조로 되어 있다. 페이로드 데이터 영역은 크게 데이터 패킷(data packet), 응답 패킷(handshake packet), 제어 패킷(token packet)으로 구성 되어 있다. 데이터 패킷은 수신기(sink)로부터 송신기(source)로 데이터의 이동이 요구 될 경우, 혹은 송신기(source)로부터 수신기(sink)로 데이터의 이동이 필요한 경우 사용된다. 데이터 패킷은 PID(Packet Identifier Field), end point, address, length 그리고 데이터 영역으로 구성되어 있다. PID는 현재의 트랜잭션이 제어(token), 데이터(data), 혹은 응답(handshake)패킷 여부를 전달해 주는 신호이며, end point신호는 패킷의 도착지점을 알려주는 신호이다. 데이터에 할당된 주소는 address영역에 패킷화되어 전송되며, 데이터의 길이를 알 수 있도록 대역폭 제어기(bandwidth controller)블록에서 결정된 데이터 길이가 패킷화된다. 전송 데이터는 데이터 영역에 패킷화된다.

응답 패킷(handshake packet)은 PID에 응답신호(ack, nak, stall)의 정보를 포함하여 송신기(source)로부터 수신기(sink)에 전달된 데이터의 이상 유무를 알려 주는 신호이다. 수신기(sink)에서 발생하는 데이터 패킷 및 응답 패킷 마지막 영역에는 sink request state 패킷이 삽입되어 수신기 내부 장치들의 요청 상태를 송신기(source)의 중재기(arbiter)로 각 트랜잭션마다 알려준다.

제어 패킷(token packet)은 수신기(sink)의 장치들로부터 데이터 요청, 혹은 장치 검색 및 수신기(sink)의 장치들의 상태를 파악할 때 사용된다.

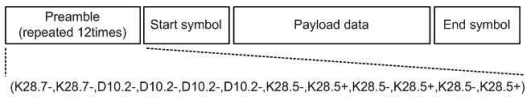


Fig. 5. Fast auxiliary channel packet structure
그림 5. 고속 보조 채널 패킷 구조

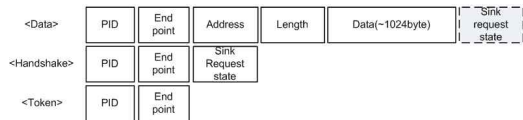


Fig. 6. Packet structure for fast auxiliary channel
그림 6. 고속 보조 채널을 위한 패킷 구조

그림 7과 같이 고속 보조 채널을 통해 데이터를 수신기(sink)에서 송신기(source)로 전송을 원할 경우, 우선 제어패킷이 송신기에서 수신기로 전송되며, 제어패킷 이외 수신기의 각 싱크 블록을 선택할 수 있는 end point 패킷과 주소정보를 패킷화해서 보낼 수 있다. end point와 주소 정보는 제어패킷이 제어를 할 수 있는 선택된 마스터 혹은 슬레이브 싱크 블록으로 전달될 수 있는 키(key)역할을 한다. 제어 심볼을 받은 수신기의 선택된 마스터 혹은 슬레이브 블록은 제어 신호에 대응되는 데이터를 수신기에서 송신기로 전송을 한다. 수신기에서 송신기로 이동하는 패킷의 구조는 PID, end point, address, length, data, sink request state 패킷으로 구성되어 있다. 가장 마지막 심볼인 sink request state는 현재 수신기의 마스터 싱크 및 슬레이브의 상태 정보를 삽입하여 보내준다. 수신기의 마스터 및 슬레이브 싱크의 요청 상태를 데이터의 트랜잭션 과정마다 송신기에 보냄으로서 데이터를 보내고자 하는 마스터 및 슬레이브 싱크의 상태를 중재기가 알 수 있게 한다.

그림 8은 송신기(source)에서 수신기(sink)로 데이터 전송을 원할 경우의 과정이며 제어패킷과 데이터를 패킷화하여 전송한다. 데이터 패킷은 제어정보 및

end point, 주소, 데이터 길이를 포함하고 있으며, 데이터를 받은 수신기(sink)는 응답 패킷을 보내준다. 이전의 경우와 마찬가지로 가장 마지막 패킷에 현재 수신부의 상태를 패킷화하여, 현재 수신단의 마스터 상태를 송신부에 알려준다.

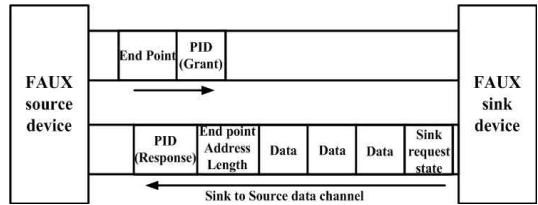


Fig. 7. Data transaction flow for read operation
그림 7. 읽기 동작을 위한 데이터 전송 순서

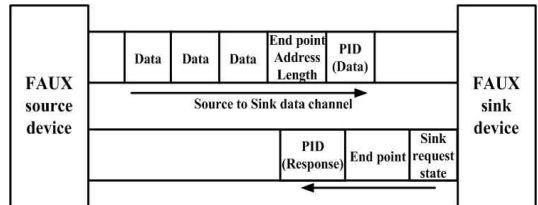


Fig. 8. Data transaction flow for write operation
그림 8. 쓰기 동작을 위한 데이터 전송 순서

IV. 고속 보조채널 검증

제안된 고속 보조채널은 디스플레이 포트의 메인링크 관리 및 고속 보조채널 관리를 하며, 고속 보조채널을 통하여 고속 양방향 데이터 전송을 할 수 있다. 링크의 관리는 링크 트레이닝이라는 과정을 통해 이루어지고, 링크 트레이닝을 통하여 차동 전압폭 설정, 동화기(equalizer)를 설정한다.

링크 트레이닝의 모든 과정은 링크 폴리시 메이커 블록이 통제 하며 그림 9와 10은 송신기(source)와 수신기(sink)의 링크 폴리시 메이커의 링크 트레이닝 모의실험 결과를 각각 보여준다. 스트림 전송 초기화 과정 후 링크 트레이닝은 stream ready 신호가 "set"되며 시작된다. 링크 트레이닝 첫 과정은 "클럭 복원 상태"이며 그림 9의 링크 트레이닝 상태(Link training state) 1에 해당한다. 이 과정 동안 메인링크 수신기의 클럭 및 데이터 복원회로가 클럭을 추출하고 데이터를 정상적으로 복원하는지를 판단하게 된다. 모의실험에서는 클럭 및 데이터 복원회로 없이 수신기의 링크 폴리시에 가상의 클럭 고정(lock) 신호를 링크트레이닝 구간에 "set"하여 송신기 및 수신기

의 링크 폴리시 메이커의 링크 트레이닝 상태가 정상적으로 동작하는 여부를 판단하였다. 두 번째 상태는 “채널 보상 상태”이고 링크 트레이닝 상태(Link training state) 2에 해당한다. 채널 보상 상태에서는 디스플레이 포트의 송신기에서 수신기까지 데이터가 메인링크를 통하여 손실 없이 전송되는지 파악하는 단계로서, 수신기의 링크 폴리시 메이커에 가상의 심볼 고정(symbol lock)신호를 링크트레이닝 구간에 “set”하여 링크 트레이닝과정이 정상적으로 동작하는지를 확인할 수 있다. 링크 트레이닝 상태(Link training state) 3과 4는 정 방향 및 역 방향 고속 보조 채널 링크 트레이닝 과정에 해당하며 고속 보조 채널에 대한 링크 트레이닝을 시작한다.

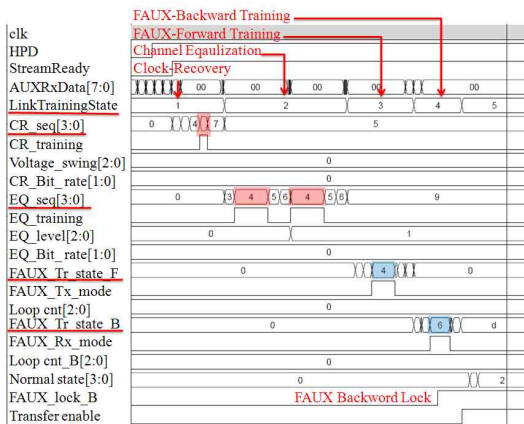


Fig. 9. Simulation result of link policy maker (source)
 그림 9. 링크 폴리시 메이커 모의실험 결과(source)

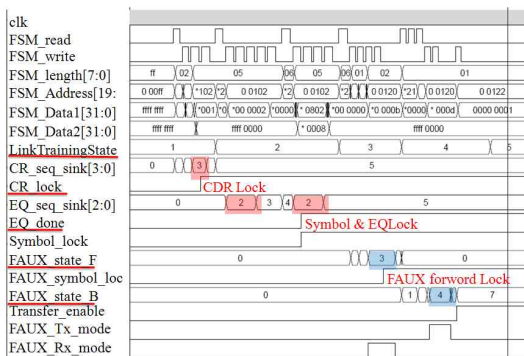


Fig. 10. Simulation result of link policy maker(sink)
 그림 10. 링크 폴리시 메이커 모의실험 결과(sink)

그림 11은 송신기 고속 보조채널 정방향 링크 트레이닝 과정을 보여주고 있다. 네 번째 고속 보조채널 정방향 링크 트레이닝 상태에서 고정된 것을 확인할 수 있고, 송신기가 수신기의 고정을 확인한 이후 다음 상태인 역방향 링크 트레이닝 과정(link training state) 4로 이동하는 것을 확인할 수 있다.

할 수 있고, 송신기가 수신기의 고정을 확인한 이후 다음 상태인 역방향 링크 트레이닝 과정(link training state) 4로 이동하는 것을 확인할 수 있다.



Fig. 11. Simulation result of FAUX forward training (source)
 그림 11. 고속 보조 채널 정방향 트레이닝 모의실험 결과(source)

그림 12는 데이터의 패킷화 과정을 나타내고 있다. 주소정보(add), 데이터 길이(length), 데이터(data)가 start 신호 이후에 패킷화 되는 것을 확인할 수 있다. 그림에서 시작 신호는 FE(16)에 해당하며 PID, 데이터 길이(length width) 0360(16), 주소(address)는 493C(16) 그리고 데이터가 성공적으로 패킷화 되는 것을 확인할 수 있다.

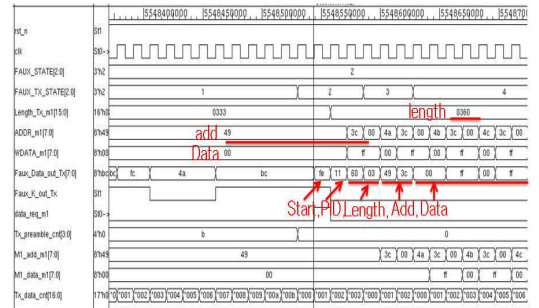


Fig. 12. Simulation result of packer/unpacker
 그림 12. Packer/unpacker 모의실험 결과

디스플레이 포트 고속 보조 채널 송·수신기의 검증을 위한 구성은 그림 13과 같다. 송신기와 수신기는 별도의 보드에 존재하여 직렬화기를 통해 보조 채널 링크를 통과 후 병렬화기를 거친 후 연결 되어야 하지만 검증을 용이하게 하기 위해 디스플레이 포트 고속 보조 채널 송·수신기의 검증은 하나의 FPGA보드 상에서 구현하였다. 채널은 이상적인 채널로, 직렬화기 및 병렬화기의 물리 층도 이상적으로 동작한다고 가정하여 병렬 데이터의 전송을 통해 데이터의 전송 과정을 검증하였다. 검증은 영상 데이터 전송을 통한 동시성 전송(isochronous data transfer)을 검증하였다. 그림 13과 같이, 고속 보조채널의 송신기(source)에는 두 개의 마스터 블록이 위치하여 수신기(sink)의 TFT LCD모듈을 위한 영상을 발생시킨다. 반대로,

수신기(sink)에 두 개의 마스터 블록이 위치되고 송신기(source)에 하나의 슬레이브 블록이 위치되어 TFT LCD 모듈을 위한 영상을 발생 시킨다. 송신기의 고속 보조 채널 블록은 두 개의 마스터 블록에서 발생시키는 영상을 중재하여 데이터를 고속 보조채널 포맷에 맞게 패킷화하는 역할을 한다. 패킷화된 데이터는 수신기의 고속보조채널 블록에서 다시 언패킷화되어 slave의 Dual Port Memory에 저장이 되며, 데이터는 DMA블록(Direct Memory Access Controller)을 통해 읽기 및 LCD Module전송 과정을 통해 TFT LCD모듈로 나오게 된다.

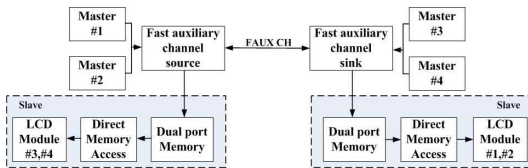


Fig. 13. Block diagram for bidirectional data transmission verification via FAUX

그림 13. 고속 보조 채널을 통한 양방향 데이터 검증을 위한 구성도

그림 14는 고속 보조 채널의 시뮬레이션 결과이다. 두 개의 마스터 모듈(module 1, module 2)로부터 발생된 데이터가 고속 보조채널의 송신기를 통해 중재 및 패킷화되어 고속 보조채널의 송신기에서 전달되고 언패킷화되어 메모리에 저장된다. 저장된 메모리의 주소를 DMA(Direct Memory Access)블록을 사용한 읽기과정을 통해 마스터 모듈(module 1, module 2)의 데이터가 출력되는 것을 확인 할 수 있다. 그림 15는 고속 보조채널의 보드테스트 결과를 나타내며 왼쪽과 오른쪽은 각각 module1, module2에서 발생된 패턴을 나타낸다.

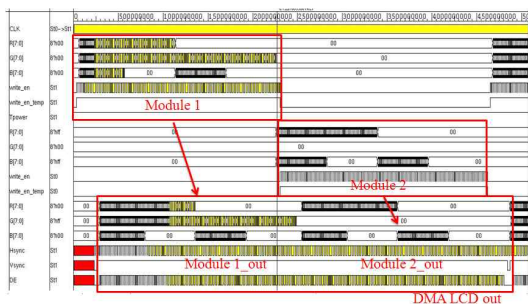


Fig. 14. Simulation result of fast auxiliary channel

그림 14. 고속 보조 채널 시뮬레이션 결과

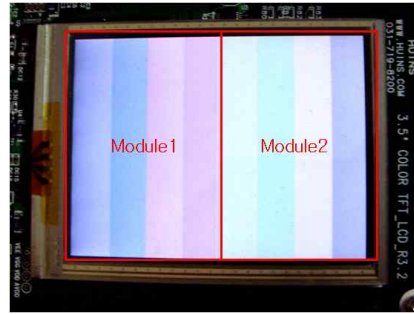


Fig. 15. Board test result

그림 15. 보드테스트 결과

V. 결론

고해상도 디스플레이 및 삼차원 영상 기술이 발전함에 따라 높은 대역폭의 디스플레이 인터페이스의 발전이 요구되고 있다. 디스플레이 장치가 발전하며 디스플레이와 연결되는 CAM(Camera Module), USB, Interface Module장치 등이 디스플레이에 추가되고 있다. 본 논문에서는 고 대역폭의 데이터를 전송할 수 있는 디스플레이포트의 보조 채널을 통한 양방향 고속 데이터 전송을 다뤘다. 제안된 고속 보조 채널을 통해 디스플레이포트의 하나의 케이블을 사용하여 메인링크를 통한 고속의 영상 데이터 전송 뿐 아니라 디스플레이 장치에 연결되어 있는 모듈의 데이터 전송이 가능할 것으로 예상된다.

본 논문을 통해 고속 양방향 보조 채널을 구성하기 위한 새로운 송·수신기 구조에 대한 적용과 고속 보조채널 및 메인링크 사이의 링크 관리 및 장치 제어를 위한 링크 서비스 및 디바이스 서비스를 위한 보조 채널 블록 적용 및 기능 검증을 수행하였다. 보조 채널이 720Mbps와 1Mbps의 대역폭을 사용할 경우에 대한 양방향 고속 데이터 전송에 대하여 검증을 수행하였다. 이는 USB2.0 및 CAM(Camera Module)데이터 전송을 만족시키는 대역폭이다.

Xilinx Vertex 4 FPGA를 사용하여 합성한 결과 7,648개의 LUTs와 6,020개의 registers가 사용되었으며, 최대 동작속도는 203MHz로서 고속 보조채널이 1byte로 동작하는 72MHz의 클럭 속도를 만족한다.

참고문헌

- [1] Hitachi, Ltd., Matsushita Electric Industrial Co., Ltd.Philips Consumer Electronics, International B.V., Silicon Image, Inc., Sony Corporation, Thomson Inc., Toshiba Corporation, "High-Definition Multimedia Interface Specification," Version 1.3, June 22, 2006
- [2] Digital Display Working Group, "Digital Visual Interface DVI," Revision 1.0, April 1999
- [3] National Semiconductor, "LVDS owner's manual, " 3rd Edition, Spring 2004
- [4] Compaq, Hewlett-Packard, Intel, Lucent, Microsoft, NEC, Philips, "Universal Serial Bus Specification," Revision 2.0, April 27, 2000
- [5] VESA, VESA DisplayPort Standard, Version 1, Revision 1a, January 11, 2008
- [6] VESA, VESA DisplayPort Standard, Version 1, Revision 2, January 5, 2010
- [7] VESA, DisplayPort Link Layer Compliance Test Standard, Version 1.0, September 14, 2007
- [8] Yoo-woo Kim, Seong-bok Cha, and Jin-Ku, "A Design of DisplayPort Link Layer," International SoC Design Conference, pp. 45-48, November, 2008
- [9] Seong-bok Cha, Yong-woo Kim, and Jin-Ku Kang, "A Design of DisplayPort AUX Channel," SoC Conference, Jeonju, Korea, May 2009
- [10] A. X. widmer, "A DC-balanced, Partitioned-Block, 8B/10B Transmission Code," IBM J. Res. Develop, vol.27 pp. 440-451, September 1983
- [11] ARM, AMBA Specification, Rev 2.0, 1999
- [12] A. Bystrov, D.J .Kinniment and A. Yakovlev, "Priority Arbiters", in Proc. IEEE 6th international Symp. ASYNC, pp.128-137, April. 2000
- [13] Mary K. Vernon and U. Manber, "Distributed Round-Robin and First-come First-Serve Protocols and Their Application to Multiprocessor Bus Arbitration", The ACM 15th Annual International Symposium on Computer Architecture, 1988
- [14] Y. Xu, L. Li, Ming-lun Gao, B.Zhand, Zhao-yu Jiand, Gao-ming Du and W.Zhang, "An Adaptive Dynamic Arbiter for Multi-Processor SoC", Solid-State and Integrated Circuit Technology International Conf., pp.1993-1996, 2006
- [15] S.Shimizu, T.Matsuoka, and K.Taniguchi, "Parallel bus systems using code-division multiple access technique",in Proc. IEEE Int. Symp. Circuits Syst., pp.II-240-II-243, 2003
- [16] K. Lahiri, A. Raghunathan and G. Lakshminarayana, "The LOTTERY BUS On-Chip Communication Architecture",IEEE Trans. VLSI Systems, vol.14, no.6, 2006
- [17] K. Kim, S. Lee, K. Lee, and H. Yoo, "an arbitration look-ahead scheme for reducing end-to-end latency in networks on chip,"IEEE Circuits and Systems, pp.2357 - 360, 2005

저 자 소 개

진 현 배 (학생회원)



2009년 : 인하대학교 전자과 졸업
(공학사)
2009년 ~ : 인하대학교 대학원 전자과 (공학석사)
<주관심분야> 고속 인터페이스 회로 설계, ADPLL

문 용 환 (비회원)



2002년 : 인하대학교 전자과 졸업
(공학사)
2004년 : 인하대학교 대학원 전자과 (공학석사)
2004년 3월~현재 : 인하대학교 대학원 전자과(박사과정)

<주관심분야> High speed transmitter and clock and data recovery circuit

장 지 훈 (비회원)



2009년 : 호서대학교 전자과 졸업
(공학사)
2009년 ~: 인하대학교 대학원 전자과 (공학석사)
<주관심분야> 고속 인터페이스 회로 설계

김 태 호 (학생회원)

2007년 : 인하대학교 전자과 졸업
(공학사)
2009년 : 인하대학교 대학원 전자
과 졸업(공학석사)
2009년 3월~현재 : 인하대학교
대학원 전자과(박사과정)

<주관심분야> High Speed CMOS Interface IC

송 병 철 (비회원)

1994년 : KAIST 전자과 졸업 (공
학사)
1996년 : KAIST 대학원 전자과
졸업(공학석사)
2001년 : KAIST 대학원 전자과
졸업(공학박사)
2001년~2008년 : 삼성전자

Senior engineer,

2008년~현재 : 인하대학교 전자과 부교수

<주관심분야> Multimedia signal processing,
Multimedia system

강 진 구 (정회원)

1983년 : 서울대학교 (공학사)
1990년 : New Jersey Institute
of Technology 전기 및 컴퓨터
공학(공학석사)
1996년 : North Carolina State
University 전기 및 컴퓨터 공학
(공학박사)

1983년~1988년 : 삼성전자(반도체)

1996년~1997년 : 미국 INTEL Senior Design
Engineer

1997년 3월~현재 : 인하대학교 전자과 교수

<주관심분야> 고속 CMOS 회로 설계, Mixed IC
설계, PLL/DLL/CDR, High Speed
Interface IC, Display IC