



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2020년07월02일
(11) 등록번호 10-2129552
(24) 등록일자 2020년06월26일

(51) 국제특허분류(Int. Cl.)
G09G 3/20 (2006.01)
(21) 출원번호 10-2014-0032474
(22) 출원일자 2014년03월20일
심사청구일자 2019년02월21일
(65) 공개번호 10-2015-0109569
(43) 공개일자 2015년10월02일
(56) 선행기술조사문헌
KR1020090049388 A
(뒷면에 계속)

(73) 특허권자
주식회사 실리콘웍스
대전광역시 유성구 테크노2로 222 (탑림동)
(72) 발명자
오광일
대전광역시 유성구 전민로46번길 15 203호
한윤택
경기도 안양시 동안구 관평로 257 한가람삼성아파트 207동 1103호
(뒷면에 계속)
(74) 대리인
이철희

전체 청구항 수 : 총 11 항

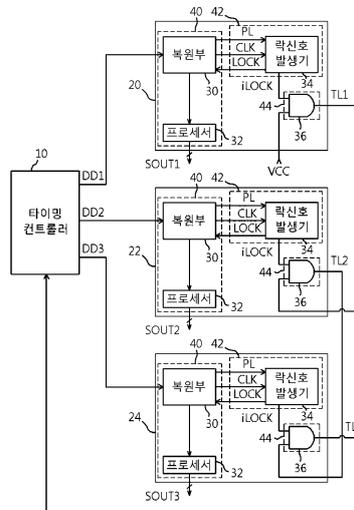
심사관 : 윤난영

(54) 발명의 명칭 소스 드라이버 및 디스플레이 장치

(57) 요약

본 발명은 소스 드라이버의 락 상태와 같은 상태 정보를 모니터링하는 것을 개선한 소스 드라이버 및 디스플레이 장치를 개시하며, 캐스케이드 경로를 형성하고 모니터링할 특정 위치를 선택하기 위한 테스트 신호를 제공함으로써 개별 소스 드라이버의 락 신호 또는 상태 정보를 모니터링할 수 있다.

대표도 - 도1



(72) 발명자

강광명

대전광역시 유성구 관들1길 35 202호

전현규

대전광역시 유성구 구즉로 16 한마을아파트104동
405호

김언영

대전광역시 대덕구 계족로761번길 24 현대아파트
106동 303호

(56) 선행기술조사문헌

KR1020130051182 A

KR1020130124096 A

KR1020120135805 A

KR1020100099927 A

KR1020120106397 A

명세서

청구범위

청구항 1

타이밍 컨트롤러에서 제공되는 디스플레이 데이터에 대응하는 소스 구동 신호를 출력하는 신호 처리부;

상기 디스플레이 데이터를 상기 소스 구동 신호로 처리하는데 필요한 제어 신호를 상기 신호 처리부에 제공하고, 외부에서 제공되는 테스트 신호에 대응하여 상기 제어 신호의 상태와 미리 설정된 값 중 하나를 표현하는 상태 정보를 제공하는 신호 제공부; 및

상기 상태 정보에 대응하는 전송 신호를 출력하는 전송 회로;를 포함함을 특징으로 하는 소스 드라이버.

청구항 2

제 1 항에 있어서,

상기 제어 신호는 로직 신호인 소스 드라이버.

청구항 3

제 1 항에 있어서,

상기 신호 제공부는 락신호 발생기를 포함하며, 상기 락신호 발생기는 상기 디스플레이 데이터에서 복원된 클럭 신호의 락 상태를 표현하는 락 신호를 생성하고 상기 제어 신호로서 상기 락 신호를 제공하는 소스 드라이버.

청구항 4

제1 항에 있어서,

상기 신호 처리부는 상기 디스플레이 데이터에 포함되어 전송된 상기 테스트 신호를 복원하고,

상기 신호 제공부는 상기 테스트 신호를 상기 신호 처리부에서 수신하는 소스 드라이버.

청구항 5

제1 항에 있어서,

상기 신호 제공부는 외부에서 제공되는 상기 테스트 신호를 핀 옵션 단자를 통하여 수신하는 소스 드라이버.

청구항 6

제1 항에 있어서,

상기 전송 회로는 상기 상태 정보와 다른 소스 드라이버의 상기 전송 신호를 비교한 결과를 출력하는 소스 드라이버.

청구항 7

제6 항에 있어서,

상기 전송 회로는 상기 상태 정보와 상기 다른 소스 드라이버의 상기 전송 신호를 앤드 조합한 결과를 출력하는 소스 드라이버.

청구항 8

디스플레이 데이터를 제공하는 타이밍 컨트롤러; 및

상기 디스플레이 데이터에 대응하는 소스 구동 신호를 출력하며, 외부에서 제공되는 테스트 신호에 대응하여 제어 신호와 미리 설정된 값 중 하나를 표현하는 상태 정보를 생성하고, 상기 상태 정보에 대응하는 전송 신호를 제공하는 복수 개의 소스 드라이버;를 포함하며,

첫째의 상기 소스 드라이버에서 생성되는 상기 전송 신호가 나머지 상기 소스 드라이버를 순차적으로 경유한 후 상기 타이밍 컨트롤러로 전송되고, 상기 첫째의 소스 드라이버에서 상기 전송 신호는 미리 설정된 값과 상기 상태 정보를 조합하여 생성되며, 상기 나머지 소스 드라이버에서 상기 전송 신호는 이전 단계의 상기 소스 드라이버의 상기 전송 신호와 현재 단계의 상기 소스 드라이버의 상기 상태 정보를 조합하여 생성됨을 특징으로 하는 디스플레이 장치.

청구항 9

제8 항에 있어서,

상기 소스 드라이버는 상기 테스트 신호의 상태에 의하여 모니터링 하는 것이 선택되고,

모니터링 하는 것이 선택되지 않은 상기 소스 드라이버들은 상기 미리 설정된 값에 대응하는 상기 상태 정보에 의하여 이전 단계의 상기 소스 드라이버의 상기 전송 신호를 그대로 출력하고, 모니터링하는 것으로 선택된 상기 소스 드라이버는 상기 제어 신호에 대응하는 상기 상태 정보에 대응하는 상기 전송 신호를 제공하는 디스플레이 장치.

청구항 10

제8 항에 있어서,

상기 전송 신호를 전송하는 캐스케이드 경로가 상기 복수 개의 소스 드라이버를 경유한 후 상기 타이밍 컨트롤러에 연결되도록 형성되며, 상기 제어 신호는 상기 디스플레이 데이터에서 복원된 클럭 신호의 락 상태를 표현하는 상기 각 소스 드라이버의 락 신호 또는 상기 각 소스 드라이버의 제어를 위한 로직 신호를 포함하는 디스플레이 장치.

청구항 11

제8 항에 있어서,

상기 테스트 신호는 상기 디스플레이 데이터에 포함되어 제공되거나 상기 각 소스 드라이버의 핀 옵션 단자를 통하여 제공되는 디스플레이 장치.

발명의 설명

기술 분야

[0001] 본 발명은 디스플레이 장치에 관한 것으로서, 보다 상세하게는 소스 드라이버의 락 상태와 같은 상태 정보를 모니터링하는 것을 개선한 소스 드라이버 및 디스플레이 장치에 관한 것이다.

배경 기술

[0002] 최근 디스플레이 장치는 액정 디스플레이(LCD) 장치, 발광 다이오드 디스플레이(LED) 장치 및 유기 발광 다이오드 디스플레이(OLED) 장치 등 다양하게 개발되고 있다.

[0003] 디스플레이 장치는 패널에 화상을 표현하기 위하여 게이트 드라이버, 소스 드라이버, 타이밍 컨트롤러 및 파워 회로 등 다양한 부품을 실장한다.

[0004] 타이밍 컨트롤러는 화상을 표현하기 위한 디스플레이 데이터를 소스 드라이버에 제공하며, 소스 드라이버는 디스플레이 데이터에 대응하는 소스 구동 신호를 생성하고 패널에 소스 구동 신호를 제공한다. 소스 드라이버는 패널의 크기나 해상도를 고려하여 패널에 대하여 복수 개가 실장될 수 있다.

[0005] 디스플레이 데이터를 전송하기 위한 타이밍 컨트롤러와 소스 드라이버 간의 신호 인터페이스는 다양한 방법으로 구현될 수 있다. 그 일례로, CEDS(Clock Embedded Data signaling) 방식이 제시될 수 있다. CEDS 방식은 데이터 신호 사이에 클럭 신호와 제어 데이터를 임베드(Embed)한 패킷을 전송하는 신호 인터페이스 방식이다.

[0006] 상기와 같이 패킷으로 디스플레이 데이터가 전송되는 경우, 디스플레이 데이터에 포함된 데이터 신호와 클럭 신호는 소스 드라이버에서 복원된다.

[0007] 타이밍 컨트롤러는 클럭 신호를 복원하기 위하여 소스 드라이버에 클럭 트레이닝을 위한 디스플레이 데이터를

제공한다. 클럭 트레이닝을 위하여, 소스 드라이버는 데이터 신호를 제외한 클럭 신호만 포함하는 클럭 트레이닝 패턴(Clock Training Pattern)을 갖는 디스플레이 데이터를 수신하며 클럭 신호를 복원하는 클럭 트레이닝 동작을 수행한다.

- [0008] 클럭 트레이닝 동작에 의하여 소스 드라이버에서 클럭 신호가 복원되면, 타이밍 컨트롤러는 데이터 신호와 클럭 신호를 포함하는 디스플레이 데이터를 전송한다. 그 후, 소스 드라이버는 복원된 클럭 신호를 이용하여 데이터 신호에 대응하는 소스 구동 신호를 생성한다.
- [0009] 상기한 과정에서, 소스 드라이버는 클럭 신호의 복원이 완료되지 않아서 위상이 언락(Unlock)된 경우에 대응하여 디스에이블 상태의 락 신호(LOCK)를 타이밍 컨트롤러에 피드백하고, 클럭 신호의 복원이 완료되어서 위상이 락(lock)된 경우에 대응하여 인에이블 상태의 락 신호를 타이밍 컨트롤러에 피드백한다.
- [0010] 대개의 경우, 디스플레이 장치는 복수 개의 소스 드라이버를 실장하며, 락 신호(LOCK)를 타이밍 컨트롤러에 전송하기 위한 캐스케이드(Cascade) 경로가 제공될 수 있다. 상기한 캐스케이드 경로는 이전 단계의 락 신호(LOCK)와 현재 단계의 락 신호(LOCK)를 비교하고 비교한 결과를 다음 단계로 전송하도록 형성될 수 있다.
- [0011] 상기와 같이 각 소스 드라이버들의 락 신호(LOCK)를 타이밍 컨트롤러에 전송하는 경우, 복수 개의 소스 드라이버들 중 어느 하나의 락 신호(LOCK)가 디스에이블 상태여도 결과적으로 타이밍 컨트롤러에 피드백되는 락 신호(LOCK)는 디스에이블 상태이다.
- [0012] 그러므로, 타이밍 컨트롤러는 소스 드라이버들 중 어느 하나에서 락 페일(Lock Fail)이 발생한 것을 판단할 수 있을 뿐 락 페일이 발생한 소스 드라이버를 분별하기 어렵다.
- [0013] 그러므로, 디스플레이 장치는 락 페일(lock fail)이 발생한 위치를 판단하기 어렵고 소스드라이버들의 개별적인 상태를 모니터링하는데 제한이 있다.
- [0014] 따라서, 타이밍 컨트롤러가 락 페일과 같은 각 소스 드라이버의 개별 상태 정보를 정확히 수집할 수 있는 기술이 제시될 필요성이 있다.

발명의 내용

해결하려는 과제

- [0015] 본 발명은 락 신호와 같은 로직 신호의 페일이 발생한 경우, 복수 개의 소스 드라이버들 중에서 락 페일이 발생한 위치를 타이밍 컨트롤러에서 확인할 수 있는 소스 드라이버 및 디스플레이 장치를 제공함을 목적으로 한다.
- [0016] 또한, 본 발명은 복수 개의 소스 드라이버를 경유하는 캐스케이드 경로를 제공하며, 상기 캐스케이드 경로를 이용하여 락 신호를 포함한 제어 신호의 상태를 모니터링할 수 있는 소스 드라이버 및 디스플레이 장치를 제공함을 다른 목적으로 한다.

과제의 해결 수단

- [0017] 상기 기술적 과제를 해결하기 위한 본 발명에 따른 소스 드라이버는, 타이밍 컨트롤러에서 제공되는 디스플레이 데이터에 대응하는 소스 구동 신호를 출력하는 신호 처리부; 상기 디스플레이 데이터를 상기 소스 구동 신호로 처리하는데 필요한 제어 신호를 상기 신호 처리부에 제공하고, 외부에서 제공되는 테스트 신호에 대응하여 상기 제어 신호의 상태와 미리 설정된 값 중 하나를 표현하는 상태 정보를 제공하는 신호 제공부; 및 상기 상태 정보에 대응하는 전송 신호를 출력하는 전송 회로;를 포함함을 특징으로 한다.
- [0018] 또한, 본 발명에 따른 디스플레이 장치는, 디스플레이 데이터를 제공하는 타이밍 컨트롤러; 및 상기 디스플레이 데이터에 대응하는 소스 구동 신호를 출력하며, 외부에서 제공되는 테스트 신호에 대응하여 상기 제어 신호의 상태와 미리 설정된 값 중 하나를 표현하는 상태 정보를 생성하고, 상기 상태 정보에 대응하는 전송 신호를 제공하는 복수 개의 소스 드라이버;를 포함하며, 첫째의 상기 소스 드라이버에서 생성되는 상기 전송 신호가 나머지 상기 소스 드라이버를 순차적으로 경유한 후 상기 타이밍 컨트롤러로 전송되고, 상기 첫째의 소스 드라이버에서 상기 전송 신호는 미리 설정된 값과 상기 상태 정보를 조합하여 생성되며, 상기 나머지 소스 드라이버에서 상기 전송 신호는 이전 단계의 상기 소스 드라이버의 상기 전송 신호와 현재 단계의 상기 소스 드라이버의 상기 상태 정보를 조합하여 생성됨을 특징으로 한다.

발명의 효과

[0019] 따라서, 본 발명에 의하면 원하는 소스 드라이버의 상태 정보를 모니터링할 수 있고, 복수 개의 소스 드라이버들 중에서 락 페일이 발생한 위치를 타이밍 컨트롤러에서 확인할 수 있다.

[0020] 또한, 본 발명에 의하면 각 소스 드라이버의 상태 정보를 수집할 수 있는 캐스캐이드 경로가 제공될 수 있으며, 상기 캐스캐이드 경로를 이용하여 락 신호를 포함한 제어 신호의 상태를 모니터링할 수 있다.

[0021] 그러므로, 본 발명은 복수 개의 소스 드라이버의 개별적인 상태를 모니터링할 수 있는 효율성을 제공할 수 있다.

도면의 간단한 설명

[0022] 도 1은 본 발명의 바람직한 실시예의 소스 드라이버 및 디스플레이 장치에 따른 실시예의 블록도.

도 2는 디스플레이 데이터의 패킷 구조도.

도 3은 본 발명의 실시예에 의한 테스트가 수행되는 것을 설명하는 타이밍 차트.

도 4는 본 발명의 소스 드라이버 및 디스플레이 장치에 따른 다른 실시예의 블록도.

발명을 실시하기 위한 구체적인 내용

[0023] 이하, 첨부된 도면을 참조하여 본 발명의 바람직한 실시예를 상세하게 설명한다. 본 명세서 및 특허청구범위에 사용된 용어는 통상적이거나 사전적 의미로 한정되어 해석되지 아니하며, 본 발명의 기술적 사항에 부합하는 의미와 개념으로 해석되어야 한다.

[0024] 본 명세서에 기재된 실시예와 도면에 도시된 구성은 본 발명의 바람직한 실시예이며, 본 발명의 기술적 사상을 모두 대변하는 것이 아니므로, 본 출원 시점에서 이들을 대체할 수 있는 다양한 균등물과 변형예들이 있을 수 있다.

[0025] 본 발명은 소스 드라이버와 디스플레이 장치를 개시하며, 소스 드라이버는 상태 정보를 개별적으로 피드백할 수 있으며, 디스플레이 장치의 타이밍 컨트롤러는 복수 개의 소스 드라이버 중 선택된 위치의 소스 드라이버의 상태 정보를 수신할 수 있다. 여기에서, 소스 드라이버는 집적 회로로 구현될 수 있다.

[0026] 본 발명에서 상태 정보는 디스플레이 데이터의 수신, 디스플레이 데이터에 포함된 데이터 신호 또는 클럭 신호의 복원, 데이터 신호에 대응하는 소스 구동 신호의 생성 또는 소스 구동 신호의 출력에 직간접으로 관련된 소스 드라이버 내부의 제어신호를 포함할 수 있다. 상기한 상태 정보는 각 제어 신호 또는 둘 이상의 제어 신호가 조합된 것으로 정의될 수 있다. 제어 신호는 로직 신호가 예시될 수 있으며, 본 발명의 실시예는 락 신호(LOCK)를 제어 신호의 일례로서 예시하여 설명한다.

[0027] 본 발명에서, 디스플레이 데이터는 데이터 신호 사이에 클럭 신호와 제어 데이터가 데이터 신호 사이에 임베드(Embed)된 CEDS 방식의 포맷을 갖는 신호로 정의될 수 있다. CEDS 방식의 디스플레이 데이터에 포함되는 데이터 신호와 클럭 신호 및 제어 신호는 동일 신호선을 통하여 전송된다. 그리고, CEDS 방식의 디스플레이 데이터의 데이터 신호와 클럭 신호는 동일한 레벨과 동일한 진폭을 갖도록 표현될 수 있다. 그리고, CEDS 방식의 디스플레이 데이터의 제어 데이터도 데이터 신호 및 클럭 신호와 동일한 레벨과 동일한 진폭을 갖도록 표현될 수 있다.

[0028] 디스플레이 데이터는 후술되는 도 2와 같이 디스플레이 데이터가 존재하는 데이터 신호 구간(DT), 클럭 신호가 존재하는 클럭 신호 구간(CT) 및 제어 데이터가 존재하는 제어 데이터 구간(P)으로 구분될 수 있다. 데이터 신호 구간(DT)에는 데이터 신호, 클럭 신호 및 제어 데이터가 각 화소 단위 또는 복수의 화소 단위로 주기적으로 존재할 수 있고, CEDS 방식의 포맷을 가질 수 있다. 즉, 클럭 신호 및 제어 데이터는 데이터 신호에 임베드된다. 클럭 신호 구간(T)에는 클럭 신호만 존재할 수 있고, 제어 데이터 구간(P)에는 제어 데이터만 존재할 수 있다. 본 발명에서 제어 데이터는 후술되는 테스트 신호(PL)를 포함할 수 있다.

[0029] 또한, 디스플레이 데이터에 의하여 패널에는 주기적인 프레임은 포함하는 화상이 형성되며, 각 프레임은 복수 개의 수평 라인을 포함한다. 여기에서, 각 수평 라인은 데이터 신호 구간(DT)에 대응될 수 있다. 그리고, 각 수평 라인의 사이에는 수평 블랭크 구간이 존재하고, 각 프레임 사이에는 수직 블랭크 구간이 존재한다. 수평 블랭크 구간과 수직 블랭크 구간에는 상기한 클럭 신호 구간(CT) 및 제어 데이터 구간(P)이 포함될 수 있고 클럭 트레이닝 패턴의 디스플레이 데이터를 이용한 클럭 트레이닝 동작이 수행될 수 있다.

- [0030] 또한, 본 발명에서, 각 소스 드라이버의 상태 정보를 수집하기 위하여, 소스 드라이버는 테스트 신호(PL)를 수신할 수 있으며, 테스트 신호(PL)가 인에이블 상태인 구간은 테스트 구간이라 정의할 수 있고, 테스트 신호(P1)가 디스에이블 상태인 구간은 노멀 구간이라 정의할 수 있다. 본 발명에서 인에이블 상태는 로직 하이 즉 하이(H)로 표시하고 디스에이블 상태는 로직 로우 즉 로우(L)로 표시한다.
- [0031] 도 1을 참조하면, 본 발명의 실시예에 따른 디스플레이 장치는 타이밍 컨트롤러(10)와 소스 드라이버들(20, 22, 24)을 포함한다.
- [0032] 타이밍 컨트롤러(10)는 소스 드라이버들(20, 22, 24)에 디스플레이 데이터들(DD1, DD2, DD3)을 제공하고, 소스 드라이버(24)에서 전송 신호(TL3)를 수신하도록 구성된다.
- [0033] 타이밍 컨트롤러(10)는 외부 디스플레이 소스(도시되지 않음)로부터 제공되는 데이터 신호와 클럭 신호 및 제어 데이터를 이용하여 도 2와 같은 포맷을 갖는 디스플레이 데이터들(DD1, DD2, DD3)을 구성하고, 디스플레이 데이터들(DD1, DD2, DD3)을 소스 드라이버들(20, 22, 24)에 각각 제공한다.
- [0034] 타이밍 컨트롤러(10)는 도 2와 같이 데이터 신호 구간(DT), 클럭 신호 구간(CT) 및 제어 데이터 구간(P)으로 구분되는 디스플레이 데이터들(DD1, DD2, DD3)을 생성하는 것으로 예시할 수 있다. 제어 데이터 구간(P)의 제어 데이터는 후술되는 테스트 신호(PL)를 포함할 수 있다.
- [0035] 타이밍 컨트롤러(10)는 테스트를 원하는 소스 드라이버에 하이(H) 상태의 테스트 신호(PL)를 제공하고 테스트를 원하지 않는 소스 드라이버에 로우(L) 상태의 테스트 신호(PL)를 제공할 수 있다.
- [0036] 그리고, 타이밍 컨트롤러(10)는 소스 드라이버(24)의 전송 신호(TL3)의 상태를 참조하여 선택된 소스 드라이버의 상태 정보를 모니터링 할 수 있다. 이에 대하여 소스 드라이버들(20, 22, 24)의 구성을 참조하여 후술한다.
- [0037] 각 소스 드라이버(20, 22, 24)는 각 디스플레이 데이터(DD1, DD2, DD3)에 대응하는 소스 구동 신호(SOUT1, SOUT2, SOUT3)를 각각 출력하며, 테스트 신호(PL)의 상태에 대응하여 락 신호(LOCK)와 미리 설정된 값 중 하나를 표현하는 상태 정보(iLOCK)를 생성하고, 상태 정보(iLOCK)를 참조한 전송 신호(TL1, TL2, TL3)를 각각 제공하도록 구성된다.
- [0038] 여기에서, 소스 드라이버(20)의 전송 신호(TL1)는 소스 드라이버들(22, 24)을 경유한 후 타이밍 컨트롤러(10)로 전송되고, 소스 드라이버(22)의 전송 신호(TL2)는 소스 드라이버(24)를 경유한 후 타이밍 컨트롤러(10)로 전송되며, 소스 드라이버(24)의 전송 신호(TL3)는 직접 컨트롤러(10)로 전송된다.
- [0039] 소스 드라이버(20)는 고정 값(VCC)과 상태 정보(iLOCK)를 조합하여 전송 신호(TL1)를 생성하고, 소스 드라이버(22)는 소스 드라이버(20)의 전송 신호(TL1)와 자신의 상태 정보(iLOCK)를 조합하여 전송 신호(TL2)를 생성하며, 소스 드라이버(24)는 소스 드라이버(22)의 전송 신호(TL2)와 자신의 상태 정보(iLOCK)를 조합하여 전송 신호(TL3)를 생성한다.
- [0040] 상기와 같이 본 발명의 실시예는 세 개의 소스 드라이버(20, 22, 24)를 포함한 것으로 예시된다. 이는 설명을 위한 예시일 뿐이며, 전송 신호를 전송하는 캐스케이드 경로가 네 개 이상의 소스 드라이버들을 경유하여 형성되도록 실시예가 구성될 수 있다. 도 1의 경우, 캐스케이드 경로는 전송 신호들(TL1, TL2, TL3)이 전송되는 경로를 포함하는 것으로 정의될 수 있다.
- [0041] 먼저, 소스 드라이버(20)의 구성을 설명한다. 소스 드라이버(20)는 신호 처리부(40)와, 신호 제공부(42) 및 전송 회로(44)를 포함하는 것으로 정의될 수 있다.
- [0042] 여기에서, 신호 처리부(40)는 타이밍 컨트롤러(10)에서 제공되는 디스플레이 데이터(DD1)에 대응하는 소스 구동 신호(SOUT1)를 출력하는 구성 요소를 포함할 수 있다. 도 1에서, 상기한 신호 처리부(40)는 복원부(30)와 프로세서(32)를 포함하는 것으로 정의될 수 있다.
- [0043] 그리고, 신호 제공부(42)는 디스플레이 데이터(DD1)를 소스 구동 신호(SOUT1)로 처리하는데 필요한 제어 신호를 제공하는 구성 요소를 포함할 수 있다. 도 1에서, 상기한 신호 제공부(42)는 락 신호(LOCK)를 신호 처리부(40)의 복원부(30)에 제공하고 테스트 신호(PL)에 대응하여 락 신호(LOCK)와 미리 설정된 값 중 하나를 표현하는 상태 정보(iLOCK)를 제공하는 락 신호 발생기(34)를 포함하는 것으로 정의될 수 있다.
- [0044] 그리고, 전송 회로(44)는 상태 정보(iLOCK)를 전송하는 구성요소를 포함할 수 있다. 도 1에서 상기한 전송 회로(44)는 앤드 게이트(36)를 포함하는 것으로 정의될 수 있다.

- [0045] 보다 구체적으로, 복원부(30)는 디스플레이 데이터(DD1)에 포함되어 전송된 클럭 신호(CLK), 데이터 신호(DATA) 및 테스트 신호(PL)를 복원한다. 복원부(30)에서 복원된 클럭 신호(CLK) 및 테스트 신호(PL)는 락 신호 발생기(34)에 제공된다. 또한, 복원부(30)에서 복원된 클럭 신호(CLK)는 데이터 신호(DATA) 복원에 이용되며, 복원부(30)는 복원된 데이터 신호(DATA)를 후술되는 프로세서(32)에 제공한다. 클럭 신호(CLK), 데이터 신호(DATA) 및 테스트 신호(PL)를 복원하는 복원부(30)의 구성 및 동작은 통상적으로 실시될 수 있는 것이므로 이에 대한 구체적인 설명은 생략한다.
- [0046] 프로세서(32)는 복원부(30)에서 제공된 데이터 신호(DATA)를 이용하여 소스 구동 신호(SOUT1)를 생성하고 생성된 소스 구동 신호(SOUT1)를 출력한다.
- [0047] 락신호 발생기(34)는 복원부(30)에서 테스트 신호(PL)와 클럭 신호(CLK)를 제공받고, 복원부(30)에 락 신호(LOCK)를 제공하며, 앤드 게이트(36)에 상태 정보(iLOCK)를 제공한다.
- [0048] 락 신호 발생기(34)는 클럭 신호(CLK)가 락 상태이면 하이(H) 상태의 락 신호(LOCK)를 복원부(30)에 제공하고 클럭 신호(CLK)가 언락(Unlock) 상태이면 로우(L) 상태의 락 신호(LOCK)를 복원부에 제공한다. 락 신호 발생기(34)는 기준 클럭 신호(미도시)를 이용하여 클럭 신호(CLK)의 락 상태를 판단할 수 있으며, 기준 클럭 신호는 내부에서 생성된 일정한 주파수를 갖는 신호 또는 복원부(30)에서 제공되는 복원되기 전의 클럭 신호가 이용될 수 있다. 즉, 락 신호 발생기(34)는 기준 클럭 신호와 클럭 신호(CLK)의 위상이 동기된 상태에 대응하여 하이(H) 상태 또는 로우(L) 상태의 락 신호(LOCK)를 제공한다.
- [0049] 또한, 락 신호 발생기(34)는 테스트 신호(PL)가 하이(H) 상태이면 제어 신호인 락 신호(LOCK)를 그대로 상태 정보(iLOCK)로 제공하고 테스트 신호(PL)가 로우(L) 상태이면 락 신호(LOCK)의 상태와 무관하게 미리 설정된 값을 갖도록 상태 정보(iLOCK)를 제공한다. 로우(L) 상태의 테스트 신호(PL)에 대응하여 상태 정보(iLOCK)로 제공되는 미리 설정된 값은 논리 레벨 하이로 제공됨이 바람직하다.
- [0050] 소스 드라이버(20)의 앤드 게이트(36)는 논리 레벨 하이에 대응하는 고정 값(VCC)과 락 신호 발생기(34)의 상태 정보(iLOCK)를 논리곱하고 논리곱 결과를 전송 신호(TL1)로서 출력한다.
- [0051] 앤드 게이트(36)는 상태 정보(iLOCK)를 전송하기 위하여 구성된 것이다. 소스 드라이버(20)에 구성된 앤드 게이트(36)는 고정 값(Vcc)과 상태 정보(iLOCK)를 논리곱한 결과를 전송 신호(TL1)로 출력하며, 그 결과 락 신호 발생기(34)의 상태 정보(iLOCK)가 그대로 전송 신호(TL1)로 출력된다. 이와 달리 소스 드라이버(22)의 앤드 게이트(36)는 락 신호 발생기(34)의 상태 정보(iLOCK)와 소스 드라이버(20)의 전송 신호(TL1)를 논리곱한 결과를 전송 신호(TL2)로 출력하도록 구성되며, 소스 드라이버(24)의 앤드 게이트(36)는 락 신호 발생기(34)의 상태 정보(iLOCK)와 소스 드라이버(22)의 전송 신호(TL2)를 논리곱한 결과를 전송 신호(TL3)로 출력하도록 구성된다.
- [0052] 본 발명의 실시예가 상술한 도 1과 같이 구성될 수 있으며, 도 1의 실시예를 이용하여 도 3과 같이 소스 드라이버를 모니터링하는 방법을 설명한다.
- [0053] 소스 드라이버들(20, 22, 24) 중 특정한 위치의 소스 드라이버에 대한 상태 정보를 얻기를 원하는 경우, 타이밍 컨트롤러(10)는 원하는 위치의 소스 드라이버에 하이(H) 상태의 테스트 신호(PL)를 제공할 수 있다. 타이밍 컨트롤러(10)에서 원하는 위치의 소스 드라이버에 하이(H) 상태의 테스트 신호(PL)를 제공하는 구간은 도 3과 같이 테스트 구간으로 정의할 수 있다.
- [0054] 보다 구체적으로, 소스 드라이버(20)의 상태 정보(iLOCK)를 얻기 원하는 경우, 소스 드라이버(22)의 상태 정보(iLOCK)를 얻기 원하는 경우 및 소스 드라이버(24)의 상태 정보(iLOCK)를 얻기 원하는 경우를 각각 설명한다.
- [0055] 먼저, 소스 드라이버(20)의 상태 정보(iLOCK)를 얻기 원하는 경우, 타이밍 컨트롤러(10)는 하이(H) 상태의 테스트 신호(PL)를 포함하는 디스플레이 데이터(DD1)를 소스 드라이버(20)에 제공한다. 또한, 타이밍 제어부(10)는 로우(L) 상태의 테스트 신호(PL)를 포함하는 디스플레이 데이터(DD2, DD3)를 소스 드라이버(22, 24)에 각각 제공한다. 여기에서, 테스트 신호(PL)은 수평 블랭크 구간 또는 수직 블랭크 구간의 제어 데이터 구간(P)에 제공될 수 있다.
- [0056] 하이(H) 상태의 테스트 신호(PL)를 제공받은 소스 드라이버(20)의 복원부(30)는 하이(H) 상태의 테스트 신호(PL)를 락 신호 발생기(34)에 제공한다.
- [0057] 락 신호 발생기(34)는 하이(H) 상태의 테스트 신호(PL)에 대응하여 현재 상태의 락 신호(LOCK)를 그대로 상태 정보(iLOCK)로 제공한다. 즉, 도 3과 같이 테스트 구간에 대응하여 상태 정보(iLOCK)는 락 신호(LOCK)의 변화를

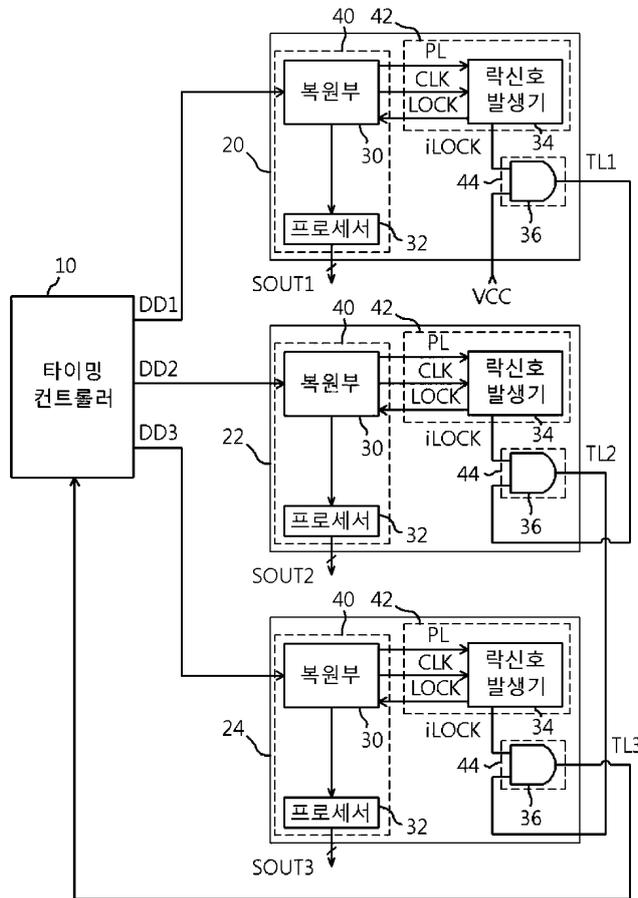
따도록 출력된다.

- [0058] 그 결과, 소스 드라이버(20)의 앤드 게이트(36)는 고정 값(VCC)과 상태 정보(iLOCK)를 논리곱한다. 고정 값(Vcc)는 논리적으로 하이 값에 해당하는 전압이며, 소스 드라이버(20)의 내부에서 생성되거나 또는 소스 드라이버(20)의 전압 인가 단자를 통하여 외부(일레로, 타이밍 컨트롤러(10))에서 제공될 수 있다. 결국, 소스 드라이버(20)의 앤드 게이트(36)는 락 신호 발생기(34)의 상태 정보(iLOCK)를 그대로 전송 신호(TL1)로 출력한다.
- [0059] 소스 드라이버(20)의 상태 정보(iLOCK)는 소스 드라이버(22)에 전달된다.
- [0060] 한편, 소스 드라이버들(22, 24)의 복원부(30)는 디스플레이 데이터(DD2)의 로우(L) 상태의 테스트 신호에 대응하여 락 신호 발생기(34)에 로우(L) 상태의 테스트 신호(PL)를 제공한다.
- [0061] 소스 드라이버들(22, 24)의 락 신호 발생기(34)는 로우(L) 상태의 테스트 신호(PL)에 대응하여 현재의 락 신호(LOCK)를 무시하고 논리적 하이 레벨에 대응하는 미리 고정된 값의 상태 정보(iLOCK)를 제공한다.
- [0062] 그 결과, 소스 드라이버(22)의 앤드 게이트(36)는 논리적 하이 레벨의 상태 정보(iLOCK)와 소스 드라이버(20)의 전송 신호(TL1)를 논리곱한다. 그리고, 소스 드라이버(24)의 앤드 게이트(36)는 논리적 하이 레벨의 상태 정보(iLOCK)와 소스 드라이버(22)의 전송 신호(TL2)를 논리곱한다. 결국, 소스 드라이버(22)의 앤드 게이트(36)는 수신된 전송 신호(TL1)를 그대로 전송 신호(TL2)로 출력하고, 소스 드라이버(24)의 앤드 게이트(36)는 수신된 전송 신호(TL2)를 그대로 전송 신호(TL3)로 출력한다.
- [0063] 상기와 같이 소스 드라이버(20)의 상태 정보(iLOCK)는 소스 드라이버들(22, 24)을 경유하는 캐스케이드 경로를 통하여 타이밍 컨트롤러(10)에 전송될 수 있다.
- [0064] 즉, 타이밍 컨트롤러(10)는 소스 드라이버(20)의 락 신호(LOCK)를 모니터링하기 위하여 하이(H) 상태의 테스트 신호(PL)를 소스 드라이버(20)에만 제공하고 그 결과 수신되는 전송 신호(TL3)를 이용하여 소스 드라이버(20)의 락 신호(LOCK)의 상태를 판단할 수 있다.
- [0065] 한편, 소스 드라이버(22)의 상태 정보(iLOCK)를 얻기 원하는 경우, 타이밍 컨트롤러(10)는 하이(H) 상태의 테스트 신호(PL)를 포함하는 디스플레이 데이터(DD2)를 소스 드라이버(22)에 제공한다. 또한, 타이밍 제어부(10)는 로우(L) 상태의 테스트 신호(PL)를 포함하는 디스플레이 데이터(DD1, DD3)를 소스 드라이버(20, 24)에 각각 제공한다.
- [0066] 하이(H) 상태의 테스트 신호(PL)를 제공받은 소스 드라이버(22)의 복원부(30)는 하이(H) 상태의 테스트 신호(PL)를 락 신호 발생기(34)에 제공한다.
- [0067] 소스 드라이버(22)의 락 신호 발생기(34)는 하이(H) 상태의 테스트 신호(PL)에 대응하여 현재 상태의 락 신호(LOCK)를 그대로 상태 정보(iLOCK)로 제공한다. 그리고, 소스 드라이버(22)의 앤드 게이트(36)는 소스 드라이버(22)에서 제공되는 전송 신호(TL1)와 상태 정보(iLOCK)를 논리곱한다. 여기에서, 전송 신호(TL1)는 후술되는 바와 같이 논리적으로 하이 값을 갖는다. 그러므로, 소스 드라이버(22)의 앤드 게이트(36)는 락 신호 발생기(34)의 상태 정보(iLOCK)를 그대로 전송 신호(TL2)로 출력한다.
- [0068] 소스 드라이버(22)의 상태 정보(iLOCK)는 소스 드라이버(24)에 전달된다.
- [0069] 소스 드라이버들(20, 24)의 락 신호 발생기(34)는 로우(L) 상태의 테스트 신호(PL)에 대응하여 현재의 락 신호(LOCK)를 무시하고 논리적 하이 레벨에 대응하는 미리 고정된 값의 상태 정보(iLOCK)를 제공한다.
- [0070] 그 결과, 소스 드라이버(20)의 앤드 게이트(36)는 논리적 하이 레벨의 상태 정보(iLOCK)와 논리적 하이 레벨 상태의 고정 값(Vcc)을 논리곱한다. 그리고, 소스 드라이버들(24)의 앤드 게이트(36)는 논리적 하이 레벨의 상태 정보(iLOCK)와 소스 드라이버(22)의 전송 신호(TL2)를 논리곱한다. 결국, 소스 드라이버(20)의 앤드 게이트(36)는 논리적 하이 레벨의 전송 신호(TL1)를 출력하고, 소스 드라이버(24)의 앤드 게이트(36)는 수신된 전송 신호(TL2)를 그대로 전송 신호(TL3)로 출력한다.
- [0071] 상기와 같이 소스 드라이버(22)의 상태 정보(iLOCK)는 소스 드라이버(24)를 경유하는 캐스케이드 경로를 통하여 타이밍 컨트롤러(10)에 전송될 수 있다.
- [0072] 즉, 타이밍 컨트롤러(10)는 소스 드라이버(22)의 락 신호(LOCK)를 모니터링하기 위하여 하이(H) 상태의 테스트 신호(PL)를 소스 드라이버(22)에만 제공하고 그 결과 수신되는 전송 신호(TL3)를 이용하여 소스 드라이버(22)의 락 신호(LOCK)의 상태를 판단할 수 있다.

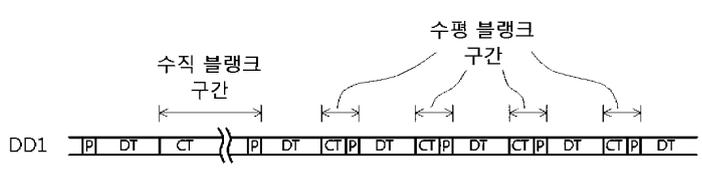
- 30 : 복원부
- 32 : 프로세서
- 34 : 락신호 발생기
- 36 : 앤드 게이트

도면

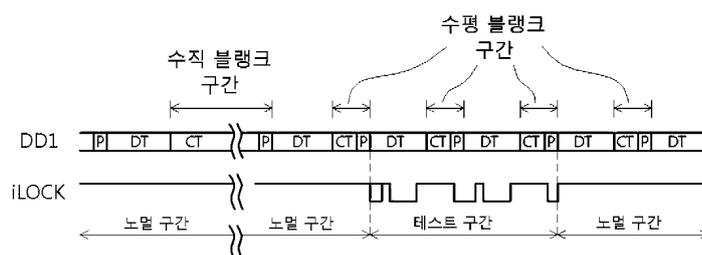
도면1



도면2



도면3



도면4

