

(19)대한민국특허청(KR)
(12) 등록특허공보(B1)

(51) 。 Int. Cl.	(45) 공고일자	2006년05월08일
B41J 2/07 (2006.01)	(11) 등록번호	10-0577357
B41J 29/38 (2006.01)	(24) 등록일자	2006년04월28일

(21) 출원번호	10-2004-7017987(분할)	(65) 공개번호	10-2004-0104720
(22) 출원일자	2004년11월08일	(43) 공개일자	2004년12월10일
(62) 원출원	특허10-2001-7005870		
	원출원일자 : 2001년05월09일	심사청구일자	2004년11월08일
번역문 제출일자	2004년11월08일		
(86) 국제출원번호	PCT/AU1999/000984	(87) 국제공개번호	WO 2000/27640
국제출원일자	1999년11월09일	국제공개일자	2000년05월18일

(81) 지정국

국내특허 : 아랍에미리트, 알바니아, 아르메니아, 오스트리아, 오스트레일리아, 아제르바이잔, 보스니아 헤르체고비나, 바르바도스, 불가리아, 브라질, 벨라루스, 캐나다, 스위스, 중국, 코스타리카, 쿠바, 체코, 독일, 덴마크, 도미니카, 에스토니아, 스페인, 핀란드, 영국, 그라나다, 그루지야, 가나, 감비아, 크로아티아, 헝가리, 인도네시아, 이스라엘, 인도, 아이슬란드, 일본, 케냐, 키르기스스탄, 북한, 대한민국, 카자흐스탄, 세인트루시아, 스리랑카, 리베이라, 레소토, 리투아니아, 룩셈부르크, 라트비아, 모로코, 몰도바, 마다가스카르, 마케도니아공화국, 몽고, 말라위, 멕시코, 노르웨이, 뉴질랜드, 폴란드, 포르투갈, 루마니아, 러시아, 수단, 스웨덴, 싱가포르, 슬로베니아, 슬로바키아, 시에라리온, 타지키스탄, 투르크멘, 터키, 트리니다드토바고, 탄자니아, 우크라이나, 우간다, 미국, 우즈베키스탄, 베트남, 세르비아 앤 몬테네그로, 남아프리카, 짐바브웨,

AP ARIPO특허 : 가나, 감비아, 케냐, 레소토, 말라위, 수단, 시에라리온, 스와질랜드, 탄자니아, 우간다, 짐바브웨,

EA 유라시아특허 : 아르메니아, 아제르바이잔, 벨라루스, 키르기스스탄, 카자흐스탄, 몰도바, 러시아, 타지키스탄, 투르크멘,

EP 유럽특허 : 오스트리아, 벨기에, 스위스, 사이프러스, 독일, 덴마크, 스페인, 핀란드, 프랑스, 영국, 그리스, 아일랜드, 이탈리아, 룩셈부르크, 모나코, 네덜란드, 포르투갈, 스웨덴,

OA OAPI특허 : 부르키나파소, 베닌, 중앙아프리카, 콩고, 코트디부아르, 카메룬, 가봉, 기니, 기니 비사우, 말리, 모리타니, 니제르, 세네갈, 차드, 토고,

(30) 우선권주장	PP7024	1998년11월09일	오스트레일리아(AU)
	PP7025	1998년11월09일	오스트레일리아(AU)

(73) 특허권자

실버브룩 리서치 피티와이 리미티드
오스트레일리아 뉴 사우스 웨일즈 2041 발메인 달링 스트리트 393

(72) 발명자

실버브룩키아
오스트레일리아 뉴 사우스 웨일즈 2041 발메인 달링 스트리트 393

랩스툼폴
오스트레일리아 뉴 사우스 웨일즈 2041 발메인 달링 스트리트 393

왓슬레이시몬로버트
오스트레일리아 뉴 사우스 웨일즈 2041 발메인 달링 스트리트 393

(74) 대리인 정진상
 박중혁

심사관 : 김태수

(54) 프린터용 파이어링 펄스 지속시간 제어 시스템

요약

본 발명은, 프린터용 파이어링 펄스 지속시간 제어 시스템으로서, 이용될 수 있는 전압을 표시하는 신호를 수신하는 제 1 입력 포트; 파이어링 펄스에 대하여 소정 지속시간을 표현하는 제어 신호를 발생시키기 위해서, 제 1 입력 포트에서 수신된 신호에 의해서 인덱싱된, 프로그래밍 가능한 펄스 지속시간 테이블; 및 제어 신호를 송신하는 출력 포트를 포함한다.

대표도

도 2

색인어

연속톤, 하프토너, 합성, 파이어링, 해상도, 레지스터, 2-레벨, 압축

명세서

도면의 간단한 설명

- 도 1은 프린터에서 2중-버퍼링으로써 성취 가능한 지속 인쇄 속도를 예시하는 테이블,
- 도 2는 애플리케이션으로부터 인쇄된 페이지로의 개념적인 데이터 흐름을 도시하는 흐름도,
- 도 3은 닫힌 구성에서의 아이프린트(iPrint) 프린터의 회화도,
- 도 4는 열린 구성에서의 아이프린트 프린터의 회화도,
- 도 5는 프린터를 통하는 용지 통로를 도시하는 절단도,
- 도 6은 멤젯(Memjet) 프린트헤드 카트리지 및 프린트헤드 캡핑(capping) 메카니즘의 절단 회화도,
- 도 7은 도 6의 멤젯 프린트헤드 카트리지 및 프린트헤드 캡핑 메카니즘의 단면도,
- 도 8은 프린터 제어기의 회화도,
- 도 9는 단순 블랙 및 화이트 이미지를 코딩하는 예시도,
- 도 10은 파이어링 순서로 번호 매겨진 10개의 인쇄 노즐의 파드의 개략도,
- 도 11은 로드 순서로 번호 매겨진 10개의 인쇄 노즐의 동일 파드의 개략도,

- 도 12는 크로마파드의 개략도,
- 도 13은 5개의 크로마파드의 파드그룹의 개략도,
- 도 14는 2개의 파드그룹의 페이지그룹의 개략도,
- 도 15는 세그먼트, 파이어그룹, 페이지그룹, 파드그룹 및 크로마파드 사이의 관계를 도시하는 개략도,
- 도 16은 전형적 인쇄 사이클 동안 A이네이블 및 B이네이블 라인의 페이지도,
- 도 17은 프린터 제어기 아키텍처의 도,
- 도 18은 페이지 확장 및 인쇄 데이터 흐름을 요약한 흐름도,
- 도 19는 EDRL 확장기 유닛의 블록도,
- 도 20은 EDRL 스트림 디코더의 블록도,
- 도 21은 런길이 디코더의 블록도,
- 도 22는 런길이 인코더의 블록도,
- 도 23은 JPEG 디코더의 블록도,
- 도 24는 하프토너/합성기 유닛의 블록도,
- 도 25는 페이지 폭과 여백 사이의 관계를 도시하는 일련의 페이지 라인,
- 도 26은 다중-임계값 디더의 블록도,
- 도 27은 3중-임계값 유닛의 로직의 블록도,
- 도 28은 프린트헤드 인터페이스의 내부 구조의 블록도,
- 도 29는 인쇄 라인 N 및 N+ 1 동안에 2중 버퍼링의 개념적 개략도,
- 도 30은 LLFU의 구조의 블록도,
- 도 31은 버퍼의 개념적 구조도,
- 도 32는 버퍼의 논리적 구조도,
- 도 33은 A이네이블 및 B이네이블 펄스 폭의 발생의 블록도,
- 도 34는 도트 카운트 로직의 도,
- 도 35는 스피커 인터페이스의 블록도,
- 도 36은 2개-층 페이지 버퍼의 도,
- 도 37은 화이트 이미지상으로 블랙 대상물의 합성을 도시하는 일련의 도,
- 도 38은 화이트 이미지상으로 연속톤 대상물의 합성을 도시하는 일련의 도,

도 39는 연속톤 대상물을 포함하는 이미지상으로 블랙 대상물의 합성을 도시하는 일련의 도,

도 40은 블랙 대상물을 포함하는 이미지상으로 불투명한 연속톤 대상물의 합성을 도시하는 일련의 도,

도 41은 블랙 대상물을 포함하는 이미지상으로 투명한 연속톤 대상물의 합성을 도시하는 일련의 도, 및

도 42는 프린터 드라이버 구성요소를 갖는 윈도우즈 9x/NT 인쇄 시스템의 블록도.

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

(기술분야)

본 발명은 페이지-폭 드롭-온-디맨드형 잉크젯 프린트헤드를 사용하는 프린터와 같은 고성능 컬러 프린터용 프린터 드라이버에 관한 것이다. 또 다른 태양에서, 본 발명은 인쇄 방법에 관한 것이다.

본 발명은 컬러 프린터용 모듈러 프린트헤드에 관한 것이다. 더 나아가 태양에서, 본 발명은 프린트헤드를 사용하여 로딩 및 인쇄하는 방법에 관한 것이다.

본 발명은 컬러 프린터용 모듈러 프린트헤드에 관한 것이다. 더 나아가 태양에서, 본 발명은 프린트헤드를 사용하여 로딩 및 인쇄하는 방법에 관한 것이다.

본 발명은 전형적으로 페이지-폭 드롭-온-디맨드형 잉크젯 프린트헤드를 사용하는 고성능 컬러 프린터에 관한 것이다. 특히, 본 발명은 연속톤(contone) 컬러 층을 2-레벨로 디지털식으로 하프토닝하고 블랙 층을 하프토닝된 연속톤 층 위로 합성하기 위한 하프토닝/합성기 유닛에 관한 것이다. 또 다른 태양에서, 본 발명은 하프토닝 및 합성 방법에 관한 것이다.

본 발명은 연속톤 컬러 화소값의 어레이 형태를 갖는 연속톤 컬러 이미지를 2-레벨 도트로 하프토닝하기 위한 디더 유닛에 관한 것이다. 또 다른 태양에서, 본 발명은 디더 유닛을 동작시키는 방법에 관한 것이다.

본 발명은 프린트헤드용 예열 사이클에 관한 것이다. 본 발명은, 예를 들어, 드롭-온-디맨드형 잉크젯 프린트헤드를 사용하는 고성능 컬러 프린터에서의 유틸리티를 갖는다.

본 발명은 컴퓨터 메모리에 유지된 자원 및 그 자원으로의 동시 액세스를 요구하는 다중 병렬 프로세서에 관한 것이다. 자원은 연속톤 컬러 화소값의 어레이 형태를 갖는 연속톤 컬러 이미지를 2-레벨 도트로 디지털식으로 하프토닝하기 위해 사용되는 디더 볼륨 또는 디더 행렬일 수 있고, 이것은 다른 임계 유닛에 의해 병렬로 액세스되도록 요구될 수 있다. 또 다른 태양에서, 본 발명은 그러한 자원에 액세스하는 방법이다.

본 발명은 프린터용 잉크통의 잉크가 바닥날 때를 예측하기 위한 방법 및 프린터 시스템에 관한 것이다. 프린터는, 예를 들어, 드롭-온-디맨드형 잉크젯 프린트헤드를 사용할 수 있다.

본 발명은 고성능 컬러 프린터와 같은 프린터용 프린트헤드에서 파이어링(firing; 분사) 펄스 지속시간의 제어에 관한 것이다. 본 발명은 특히 제어 시스템 및 방법에 관한 것이다. 프린터는, 예를 들어, 페이지폭 드롭-온-디맨드형 잉크젯 프린트헤드를 사용할 수 있다.

(발명의 배경)

범용 256-레벨 디더 볼륨은 다른 강도 레벨을 감결합함으로써, 디더 셀 디자인에서 현저한 유연성을 제공한다. 그것은 최적 확률적(stochastic) 디더를 디자인할 때 디더 행렬에 대하여 여분의 자유도를 제공한다[12]. 범용 디더 볼륨은 클 수 있다 - 64×64×256 디더 볼륨은, 예를 들어, 128KB의 크기를 갖는다. 또한 각각의 컬러 구성요소가 볼륨과 다른 비트의 검색을 요구할 수 있기 때문에 그것에 액세스하는 것이 비능률적일 수 있다.

드롭-온-디맨드형 잉크젯 프린트헤드에서, 인쇄된 도트의 크기는 잉크 온도에 따라 변한다. 잉크 방울을 분출하기 위해 에너지가 사용될 때, 잉크 온도가 상승한다. 잉크 방울이 실제로 분출될 때, 그것은 잉크 온도를 다소 가져가 버린다. 다수의 방울을 분출하는 과정 동안 평형 온도에 도달한다. 일단 평형 온도에 도달하면 인쇄 프로세스는 평형 온도에 머무르는 경향이 강하다.

본 발명은 프린트헤드의 온도를 표시하는 신호를 산출하는 온도 센서, 및 잉크통으로부터 잉크를 받는 잉크 공급 포트, 잉크 도트를 분출하는 잉크 증착(deposition) 포트, 및 프린트헤드의 파이어링 사이클 동안에 증착 포트로부터 잉크 도트가 증착되어지게 하는 신호를 수신하는 파이어링 제어 포트를 각각 갖는 복수의 노즐을 포함한 유형의 프린트헤드용 프린트헤드 제어기를 구비한 프린터이다. 여기서, 프린트헤드 제어기는 온도 센서로부터 프린트헤드의 온도를 표시하는 신호를 수신하는 수단, 및 노즐의 파이어링 제어 포트에 파이어링 신호를 공급하는 수단을 포함하고, 프린트헤드 제어기는, 각각의 인쇄 작업 전에, 모든 노즐을 파이어링으로 설정하고, 각각의 노즐에 일련의 짧은 파이어링 펄스를 제공하기 위해 일련의 변형 파이어링 사이클을 실행하도록 동작 가능하고, 프린트헤드의 온도를 표시하는 신호가 동작 평형 온도에 도달했음을 표시할 때까지 각각의 펄스의 지속시간은 방울을 파이어링하기에는 불충분하지만 잉크를 가열시키기에는 충분하다.

예열 모드 동안의 피드백이 T센스(Tsense; 아래에 정의됨)에 의해서 제공될 수 있고, 실온보다 높은 약 30℃의 온도에 도달할 때까지 계속된다. 시간 또는 온도는 잉크 구성물에 의존하여 변할 수 있기 때문에, 평형 온도에 도달하는 때에 관하여 온도 정보가 피드백되게 하는 것이 중요하다.

일 예에서, 각각의 노즐에 대해서 약 200 펄스가 요구된다.

예열 모드의 지속시간은 약 50 밀리초일 수 있고, 잉크 구성물에 의존한다.

예열은 데이터가 프린터에 전송되고 있는 동안에 이루어지기 때문에 성능에 영향을 미치지 않는다.

예열 사이클은 1을 갖는 모든 노즐로의 단일 로드 사이클(즉, 모든 노즐을 파이어링으로 설정), 및 각각의 노즐로의 많은 짧은 파이어링 펄스를 수반한다. 펄스의 지속시간은 방울을 파이어링하기에는 불충분하지만 잉크를 가열시키기에는 충분해야 한다. 펄스의 지속시간은 프린터에서 사용되는 잉크의 각각에 대해서 변화할 수 있다. 전체적으로 각각의 노즐에 대해서 약 200 펄스가 요구되고, 표준 인쇄 사이클로서 동일 시퀀스에서 사이클링 스루한다.

다중 병렬 프로세서는 컴퓨터 메모리에 유지된 자원으로서의 동시 액세스를 요구하는 경우에, 몇 가지 방법이 가능하다. 첫 번째로, 프로세서가 교대로 자원에 액세스할 수 있지만, 그러나 이것은 프로세서의 성능을 감소시킨다. 두 번째로, 다중-포트 메모리가 사용될 수 있고, 세 번째로, 전체 자원이 다른 메모리 뱅크에 복제될 수 있지만 이 두가지 옵션은 비용이 많이 든다.

컴퓨터 메모리에 유지된 자원의 특별한 예는 연속톤 컬러 이미지를 디지털식으로 하프토닝하기 위해 사용되는 디더 볼륨 또는 디더 행렬이다. 디더 셀 일치(registration)가 이미지의 다른 컬러 평면(color plane) 사이에서 소망되지 않을 때, 개개의 컬러 구성요소의 디더링을 처리하는 세트 임계 유닛은 다른 디더 셀 위치로의 동시 액세스를 요구할 수 있다.

프린트헤드로의 파이어링 펄스의 지속시간은 잉크의 점성에 의존하는데, 잉크의 점성은 온도 및 잉크 특성 및 프린트헤드에 이용될 수 있는 전력량에 의존한다. 전형적인 펄스 지속시간 범위는 1.3 내지 1.8 ms이다.

발명이 이루고자 하는 기술적 과제

본 발명의 목적은, 프린터용 파이어링 펄스 지속시간 제어 시스템으로서, 이용될 수 있는 전압을 표시하는 신호를 수신하는 제 1 입력 포트; 파이어링 펄스에 대하여 소정 지속시간을 표현하는 제어 신호를 발생시키기 위해서, 제 1 입력 포트에서 수신된 신호에 의해서 인덱싱된(색인이 생성된), 프로그래밍 가능한 펄스 지속시간 테이블; 및 제어 신호를 송신하는 출력 포트를 포함하는, 프린터용 파이어링 펄스 지속시간 제어 시스템을 제공하는 것이다.

발명의 구성 및 작용

본 발명은 고성능 프린터용 프린터 드라이버인데, 프린터 드라이버는 인쇄될 페이지를 표현하는 2개의 층 페이지 버퍼를 관리한다. 버퍼의 제 1 층은 배경 연속톤 데이터를 포함하고, 반면에 제 2 층은 전경 2-레벨 데이터를 포함한다. 프린터 드라이버는 페이지의 완료시에 버퍼를 프린터에 송신하고, 프린터는 연속톤 층을 하프토닝한 후에 2개의 층을 합성한다. 프

린터 드라이버는 연속톤 층과 합성되는 연속톤 데이터가 2-레벨 층의 데이터를 가리는(obscure) 것을 결정할 때, 가려진 2-레벨 데이터는 2-레벨 층으로부터 제거되고 버려지거나, 또는 가리는 연속톤 데이터에 의해서 표현된 이미지와 가려진 2-레벨 데이터에 의해서 표현되는 이미지 사이에 특정 상호작용이 존재하는 경우에 그 때 가리는 연속톤 데이터가 연속톤 층과 합성되기 전에 가려진 2-레벨 데이터가 연속톤 층과 합성되도록 동작한다.

연속톤 데이터는 2-레벨 데이터보다 더 낮은 해상도를 가질 수 있다.

2개의 층 페이지 버퍼는 연속톤 데이터의 제 3 층으로써 증대될 수 있는데, 제 3 층은 제 1 층의 해상도로 서브샘플링된 제 2 층의 2-레벨 데이터의 연속톤 버전을 포함하고, 프린터 드라이버는 제 1 층과 합성되는 연속톤 데이터가 제 3 층의 연속톤 데이터를 가린다는 것을 결정할 때, 가리는 연속톤 데이터가 연속톤 층과 합성되기 전에 가려진 연속톤 데이터가 제 3 층으로부터 제거되고 연속톤 층과 합성되고, 제 3 층의 가려진 연속톤 데이터에 대응하는 제 2 층의 2-레벨 데이터가 제 2 층으로부터 제거되도록 동작한다.

전형적으로 상호작용은 연속톤 데이터가 불투명하지 않은 이미지 대상물을 표현하는 경우에 가려진 2-레벨 데이터와 가리는 연속톤 데이터에 의해서 표현된 이미지 사이에서 일어난다.

연속톤 데이터는 전형적으로 컬러 데이터이고, 반면에 2-레벨 데이터는, 또 다른 컬러일 수 있기는 하지만, 통상 블랙 데이터일 것이다. 부가 컬러의 부가 전경 2-레벨 층을 수용하기 위해서 부가 층이 제공될 수 있다.

프린터 드라이버는 다른 그래픽스 및 이미징 오퍼레이션, 특히 합성 오퍼레이션 및 텍스트 오퍼레이션에 대하여 디바이스-특정 처리를 제공할 수 있도록, 통상 호스트 그래픽스 시스템과 밀접하게 연결될 것이다.

디바이스-독립 컬러가 표준 방식으로 프린터-특정 컬러로 변환될 수 있도록, 호스트는 컬러 관리에 대한 지원을 제공할 것이다. 프린터에 전송된 페이지 기술(description)은 일반적으로 디바이스-특정 컬러를 포함한다.

호스트 그래픽스 시스템은 프린터 드라이버에 의해서 지정된 공칭 해상도로 이미지 및 그래픽스를 렌더링하지만, 그러나 그것은 프린터 드라이버가 텍스트 렌더링을 제어하게 한다. 특히, 그래픽스 시스템은 프린터 드라이버가 공칭 디바이스 해상도보다 더 높은 해상도에서 텍스트를 렌더링 및 위치결정하게 하도록 프린터 드라이버에 충분한 정보를 제공한다.

호스트 그래픽스 시스템은 공칭 디바이스 해상도에서 연속톤 페이지 버퍼로의 랜덤 액세스를 요구하고, 그 안으로 그래픽스 및 이미징 대상물을 합성하지만, 그러나 프린터 드라이버가 페이지 버퍼를 관리하기를 기대하는 점에서 프린터 드라이버가 실제의 합성을 제어하게 한다.

프린터의 페이지 기술은 267 ppi 연속톤 층 및 800 dpi 블랙 층을 포함할 수 있다. 프린터 드라이버는 그래픽스 시스템에 267 ppi의 공칭 페이지 해상도를 지정할 수 있다. 가능한 경우에 프린터 드라이버는, 블랙 텍스트를 제외하고, 이미지 및 그래픽스 대상물을 267 ppi에서 화소 레벨로 렌더링하기 위해서 그래픽스 시스템에 의존한다. 프린터 드라이버는 모든 텍스트 렌더링 요청을 처리하고, 800 dpi에서 블랙 텍스트를 검출 및 렌더링하지만, 그러나 267 ppi에서 렌더링을 위해 블랙-아닌 텍스트 렌더링 요청을 그래픽스 시스템으로 반환한다.

프린터 드라이버에 의해서 이하 규칙이 구현될 수 있다.

블랙 대상물은 페이지 버퍼와 합성될 때 2-레벨 블랙 층과 합성된다. 블랙 층은 대상물 불투명도(opacity)와 블랙 층 불투명도를 단순히 OR함으로써 업데이트되고, 중해상도 연속톤 블랙 층의 대응 부분은 고해상도 블랙 층으로부터 재계산된다.

연속톤 컬러 대상물은 페이지 버퍼와 합성될 때 연속톤 층과 합성된다. 연속톤 층 및 블랙 층은 다음과 같이 업데이트된다:

연속톤 대상물이 블랙 층을 가리는 경우마다, 완전히 불투명하게는 아니더라도, 영향받은 블랙 층 화소는 블랙 층으로부터 연속톤 층으로 밀려난다, 즉, 연속톤 층과 합성되고 블랙 층으로부터 제거된다. 그 후 연속톤 대상물은 연속톤 층과 합성된다.

연속톤 대상물 화소가 완전히 불투명하다면, 배경 연속톤 화소는 전경 연속톤 화소에 의해서 후속적으로 완벽히 말소될 것이기 때문에, 그 때 해당 블랙 화소를 배경 연속톤 층으로 밀어 넣을 필요는 없다.

2-레벨 블랙 층 데이터는 압축된 형태로 프린터에 송신될 수 있다. 그룹 4 픽시밀리 코딩은 이러한 목적으로 사용될 수 있으나, 수평 런(run) 길이의 변형 허프만 코딩이 더 높은 해상도를 위해 제거되거나 또는 튜닝된다면 더 좋다. 대안으로 EDRL 압축이 사용될 수 있다.

연속톤 층 데이터는 또한 압축된 형태로 프린터에 송신될 수 있다. JPEG 또는 웨이브렛 압축이 이러한 목적으로 사용될 수 있다.

또 다른 태양에서 본 발명은 고성능 프린터용 프린터 드라이버를 동작시키는 방법인데, 프린터 드라이버는 2개의 층 페이지 버퍼를 관리한다; 제 1 층은 연속톤 데이터를 위한 것이고, 제 2 층은 2-레벨 데이터를 위한 것인데 2-레벨 데이터는 프린터에 의해서 연속톤 데이터와 합성될 것이다; 상기 방법은

연속톤 층과 합성되는 연속톤 데이터가 2-레벨 층의 데이터를 가리는 때를 결정하는 단계, 가려진 2-레벨 데이터를 2-레벨 층으로부터 제거하는 단계, 및 그것을 버리는 단계, 또는 가려진 2-레벨 데이터에 의해서 표현된 이미지와 가리는 연속톤 데이터에 의해서 표현된 이미지 사이에 특정 상호작용이 존재하는 경우에 연속톤 데이터와 연속톤 층을 합성하기 전에 가려진 2-레벨 데이터와 연속톤 층을 합성하는 단계를 포함한다. 본 발명은 잉크 공급 포트, 잉크 증착 포트 및 프린트헤드의 파이어링 사이클 동안에 증착 포트로부터 잉크 도트가 증착되어지게 하는(파이어링) 신호를 수신하는 파이어링 제어 포트를 각각 갖는 복수의 노즐을 포함하는, 컬러 프린터용 모듈러 프린트헤드이다. 프린트헤드의 노즐은 그룹 또는 파드에 배치되고, 각각의 파드(pod)의 노즐의 잉크 공급 포트는 공통 잉크 공급 라인에 접속되어 있다. 각각의 파드의 노즐은 행(행은 인쇄될 페이지를 가로지르는 방향으로 뻗어 있다)에 배치되고, 노즐의 행은 각각의 행의 노즐이 동시에 파이어링된다면 도트를 다른 라인상에 증착시키도록 서로로부터 오프셋된다. 선택된 상호 배제적인 노즐 서브-그룹이 파이어링 사이클의 소정 페이지에서 그들이 동시에 파이어링되게 하도록 게이팅된 그들의 파이어링 제어 포트를 갖는 동작 그룹내로 각각의 다른 컬러의 파드가 함께 배치된다.

일 예에서 각각의 파드에 노즐의 2개의 행이 존재하고 하나의 행의 노즐은 페이지상의 하나의 라인을 따라 짝수 도트를 증착시키고, 다른 행의 노즐은 페이지상의 인접 라인을 따라 홀수 도트를 증착시킨다.

노즐의 행 사이의 오프셋의 양은 노즐 아래의 용지의 흐름과 매칭하도록 디자인된다.

각각의 파드의 노즐은 제 1 측으로부터 시작하는 제 1 행을 따라서, 그리고 그 후 다른 측에서 끝나는 동일한 방향으로의 다른 행을 따라서 순서대로 파이어링될 수 있다.

단일 파드는 공통 잉크통을 공유하는 10개의 노즐로 구성될 수 있다. 5개의 노즐이 하나의 행에 있고, 5개가 또 다른 행에 있다. 각각의 노즐은 15.875mm 격자상에 일정 간격으로 이격된 직경 22.5mm 도트를 산출할 수 있다.

하나의 컬러의 노즐에 의해서 인쇄된 도트는 동시에 다른 컬러에 의해서 인쇄된 것과는 다른 라인을 위한 것이나, 크로마 파드(chromapod)의 각각의 파드가 차례로 도트의 동일 그룹을 인쇄하도록 다른 컬러의 파드가 배치되는 크로마파드를 형성하기 위해서 각각의 다른 컬러의 하나의 파드는 함께 그룹화될 수 있다.

시안, 마젠타, 옐로우 및 블랙의 하나의 파드는 크로마파드로 그룹화될 수 있다. 크로마파드는 다른 라인상의 10개의 도트의 동일 수평 세트의 다른 컬러 구성요소를 표현한다. 다른 컬러 파드 사이의 정확한 거리는 일정수의 도트-폭이고, 따라서 인쇄할 때 고려되어야 한다: 인쇄 알고리즘은 컬러 사이의 도트-폭의 가변 거리에 대해서 허용하여야 한다.

이네이블된 파드그룹에서의 노즐의 그룹이 일정 파이어링 페이즈 동안에 동시에 파이어링되는 페이즈그룹으로 하나 이상의 크로마파드가 형성될 수 있다. 페이즈그룹의 파드그룹의 하나 이상은 요구되는 인쇄 속도에 의존해서 동시에 이네이블된다.

5개의 크로마파드가 단일 파드그룹으로 편성될 수 있다. 각각의 크로마파드는 40개의 노즐을 포함할 수 있기 때문에, 각각의 파드그룹은 200개의 노즐(50개의 시안, 50개의 마젠타, 50개의 옐로우, 및 50개의 블랙 노즐)을 포함할 수 있다.

2개의 파드그룹이 단일 페이즈그룹으로 편성될 수 있다. 페이즈그룹은 페이즈그룹내의 노즐의 그룹이 일정 파이어링 페이즈 동안에 동시에 파이어링되기 때문에 그렇게 명명된다. 2개의 파드그룹으로부터 페이즈그룹의 형성은 2개의 파드그룹 이네이블 라인을 통해서 저속 및 고속 인쇄를 전부 허용한다.

2개의 페이지그룹은 단일 파이어그룹(firegroup)으로 편성될 수 있고, 각각의 세그먼트에는 4개의 파이어그룹이 있다. 파이어그룹은 그들이 모두 동일 노즐을 동시에 파이어링하기 때문에 그렇게 명명된다. 2개의 이네이블 라인은 다른 파이어링 페이지로서 독립적으로 페이지그룹의 노즐의 파이어링을 허용한다.

4-인치 프린트헤드는 전형적으로 8개의 세그먼트로 나란히 구성될 것이고, 각각의 세그먼트는 4개의 파이어그룹을 가질 것이다.

더 넓은 프린트헤드는 프린트헤드의 2개를 함께 조립함으로써 만들어질 수 있다. 그래서 8-인치 프린트헤드는 총 51,200개의 노즐에 대해서 2개의 4-인치 프린트헤드로 구성된다.

노즐 계층은 고른 전력 소비를 유지하면서 오버래핑 페이지 및 다중 속도를 허용한다. 더하여, 노즐 그룹화 파드는 물리적 안정성을 제공한다.

전력 소비의 견지에서, 노즐 그룹화는 속도/전력 소비가 다른 제품 구성에서 교환(trade-off)되게 하기 위해서 저속 및 고속 인쇄 모드를 이네이블한다.

단일 4-인치 프린트헤드는 총 25,600개의 노즐을 포함할 수 있다. 인쇄 사이클은 인쇄될 정보에 의존하여 모든 이들 노즐까지의 파이어링을 수반한다. 그들을 모두 동시에 파이어링하는 것은 너무 많은 전력을 소비하고 잉크 리필 및 노즐 간섭의 견지에서 문제일 것이다. 더 나아가, 노즐의 파이어링은 또한 그 노즐 파드의 공통 잉크통내에서 제한된 시간 동안 음향 섭동을 일으킨다. 섭동은 동일 파드내의 또 다른 노즐의 파이어링과 간섭할 수 있다. 결과적으로, 파드내의 노즐의 파이어링은 가능한 서로로부터 오프셋되어야 한다.

이것을 다루기 위해서, 컬러 당 하나의 노즐이 크로마파드로부터 파이어링될 수 있고 그 후 파드그룹내의 다음 크로마파드로부터 노즐이 파이어링될 수 있다.

2개의 파이어링 모드(저속 인쇄 모드 및 고속 인쇄 모드)가 정의될 수 있다.

저속 인쇄 동안에, 각각의 페이지그룹의 하나의 파드그룹만이 파이어링 펄스를 수신해서 2개중 하나의 파드그룹만이 노즐을 파이어링한다. 저속 인쇄 모드에서 2개의 파드그룹내의 크로마파드는 처음 크로마파드가 다시 파이어링하기 전에 모두 파이어링해야 한다.

저속 인쇄 모드에서, 128개의 노즐이 각각의 4-인치 프린트헤드로부터 동시에 파이어링될 수 있다. 파이어링된 노즐은 최대한으로 멀어야 하고, 그래서 16개의 노즐이 각각의 8개의 세그먼트로부터 파이어링된다. 모든 25,600개의 노즐을 파이어링하기 위해서, 128개의 노즐의 200개의 다른 세트가 파이어링되어야 한다.

고속 인쇄 동안에, 2개의 파드그룹이 설정되어, 2개의 파드그룹이 노즐을 파이어링한다. 고속 인쇄 모드에서 단일 파드그룹내의 크로마파드는 처음 크로마파드가 다시 파이어링하기 전에 모두 파이어링해야 한다.

고속 인쇄 모드에서, 256개의 노즐이 각각의 4-인치 프린트헤드로부터 동시에 파이어링될 수 있다. 파이어링된 노즐은 최대한으로 멀어야 하고, 그래서 32개의 노즐이 각각의 세그먼트로부터 파이어링된다. 모든 25,600개의 노즐을 파이어링하기 위해서, 256개의 노즐의 100개의 다른 세트가 파이어링되어야 한다.

고속 인쇄가 동시에 두 배의 노즐을 파이어링하기 때문에 결과적으로 저속 인쇄는 고속 인쇄보다 두 배의 시간이 걸린다. 저속 모드에서의 전력 소비는 고속 모드의 전력 소비의 반이다. 그러나, 페이지를 인쇄하는데 소비되는 에너지는 두가지 경우에서 동일함을 주목하라.

프린트헤드는 파이어링 펄스의 타이밍을 조절하기 위해서 피드백의 수개의 라인을 산출한다. 하나의 피드백 신호는 프린트헤드가 얼마나 뜨거운지를 제어기에게 알려준다. 이것은 온도가 잉크의 점성에 영향을 미치기 때문에, 제어기가 파이어링 펄스의 타이밍을 조절하게 한다. 제 2 피드백 신호는 얼마나 많은 전압이 액추에이터에 이용될 수 있는지를 제어기에게 알려준다. 이것은 펄스 폭을 조절함으로써 제어기가 플랫폼 배터리 또는 고전압원에 대해 보상하게 한다. 제 3 피드백 신호는 액추에이터 히터의 저항률(스퀘어 당 옴)을 제어기에게 알려준다. 이것은 히터 저항률에 관계없이 일정한 에너지를 유지하기 위해서 제어기가 펄스 폭을 조절하게 한다. 제 4 피드백 신호는 리소그래픽 및 에칭 편차에 기인하여 $\pm 5\%$ 까지 변할 수 있는 히터의 임계부의 폭을 제어기에게 알려준다. 이것은 제어기가 펄스 폭을 적절히 조절하게 한다.

로드 사이클은 후속 인쇄 사이클 동안에 인쇄될 정보로써 프린트헤드의 로딩 업을 수반한다. 각각의 노즐의 파이어링 제어 포트는 노즐이 인쇄 사이클 동안에 파이어링할 것인지 아닌지를 결정하는 연관 노즐이네이블 비트를 가질 수 있다. 노즐이네이블 비트는 시프트 레지스터의 세트를 통해서 로드 사이클 동안 로딩된다. 모든 시프트 레지스터가 완전히 로딩되면, 모든 비트는 적합한 노즐이네이블 비트에 병렬로 전송된다. 전송이 일어난 직후, 인쇄 사이클이 시작될 수 있다. 인쇄 사이클 및 로드 사이클은 모든 노즐이네이블 비트의 병렬 로드와 인쇄 사이클의 끝에 일어나는 한 동시에 일어날 수 있다.

인쇄 프로세스는 프린트헤드에 대하여 정확한 순서로 데이터를 산출해야 한다. 예로서, 제 1 클록 펄스는 다음 인쇄 사이클의 도트 0, 800, 1600, 2400, 3200, 4000, 4800, 및 5600에 대하여 CMYK 비트를 전송할 수 있다. 제 2 클록 펄스는 다음 인쇄 사이클의 도트 1, 801, 1601, 2401, 3201, 4001, 4801, 및 5601에 대하여 CMYK 비트를 전송할 수 있다. 800 SR클록 펄스 후에, 전송 펄스가 주어질 수 있다.

물론, 800 SR클록 펄스내에서, 시프트 레지스터는 노즐이네이블 비트로의 최종 전송과의 통신에 따라 로딩되어야 하고, 여기서 많은 다른 배선 가능성이 존재한다. 하나의 로딩(따라서 배선) 가능성은 비트를 파드 순서대로 로딩하는 것인데, 각각의 파드내에서 비트는 파드의 하나의 측으로부터 다른 측으로의 각각의 노즐을 표현한다(제 1 행에서의 제 2 노즐상로의 이동 전에 제 1 행으로부터 마지막 행으로까지 제 1 노즐을 효과적으로 로딩). 2개의 행 파드에서 이것은 외견상 지그-재그 방식으로 노즐을 로딩하는 것을 의미한다. 또 다른 가능성은 비트를 파드 순서대로 로딩하는 것인데, 각각의 파드내에서 비트는 각각의 행을 표현하고, 각각의 행내에서 파드의 하나의 측으로부터 다른 측으로 노즐에서부터 시작한다.

홀수 및 짝수 CMYK 출력은, 동일 인쇄 사이클 동안에 인쇄되더라도, 동일한 물리적 출력 라인상에 나타나지 않는다는 것을 주목하는 것은 중요하다. 다른 컬러의 노즐 사이의 분리뿐만 아니라 프린트헤드내에서의 홀수 및 짝수 노즐의 물리적 분리는 그들이 페이지의 다른 라인상에 도트를 산출할 것을 보장한다. 이러한 상대적 차이는 데이터를 프린트헤드내로 로딩할 때에 대해서 고려되어야 한다. 라인에서의 실제 차이는 프린트헤드에서 사용된 잉크젯 메카니즘의 특성에 의존한다. 그 차이는 다른 컬러의 노즐 사이의 거리 및 동일 컬러의 노즐 사이의 거리를 표현하는 변수에 의해서 정해될 수 있다.

본 발명은 잉크 공급 포트, 잉크 증착 포트 및 프린트헤드의 파이어링 사이클 동안에 증착 포트로부터 잉크 도트가 증착되어지게 하는(파이어링) 신호를 수신하는 파이어링 제어 포트를 각각 갖는 복수의 노즐을 포함하는, 컬러 프린터용 모듈러 프린트헤드이다. 프린트헤드의 노즐은 그룹 또는 파드에 배치되고, 각각의 파드의 노즐의 잉크 공급 포트는 공통 잉크 공급 라인에 접속되어 있다. 각각의 파드의 노즐은 행(행은 인쇄될 페이지를 가로지르는 방향으로 뻗어 있다)에 배치되고, 각각의 행의 노즐이 동시에 파이어링된다면 도트를 다른 라인상으로 증착시키도록 노즐의 행은 서로로부터 오프셋된다. 선택된 상호 배제적인 노즐 서브-그룹이 파이어링 사이클의 소정 페이지에서 그들이 동시에 파이어링되게 하도록 게이팅된 그들의 파이어링 제어 포트를 갖는 동작 그룹내로 각각의 다른 컬러의 파드가 함께 배치된다.

일 예에서 각각의 파드에 노즐의 2개의 행이 존재하고 하나의 행의 노즐은 페이지상의 하나의 라인을 따라 짝수 도트를 증착시키고, 다른 행의 노즐은 페이지상의 인접 라인을 따라 홀수 도트를 증착시킨다.

노즐의 행 사이의 오프셋의 양은 노즐 아래의 용지의 흐름과 매칭하도록 디자인된다.

각각의 파드의 노즐은 제 1 측으로부터 시작하는 제 1 행을 따라서, 그리고 그 후 다른 측에서 끝나는 동일한 방향으로의 다른 행을 따라서 순서대로 파이어링될 수 있다.

단일 파드는 공통 잉크통을 공유하는 10개의 노즐로 구성될 수 있다. 5개의 노즐은 하나의 행에 있고, 5개는 또 다른 행에 있다. 각각의 노즐은 15.875mm 격자상에 일정 간격으로 이격된 직경 22.5mm 도트를 산출할 수 있다.

하나의 컬러의 노즐에 의해서 인쇄된 도트가 동시에 다른 컬러에 의해서 인쇄된 것과는 다른 라인을 위한 것이나, 크로마 파드의 각각의 파드가 차례로 도트의 동일 그룹을 인쇄하도록 다른 컬러의 파드가 배치되는 크로마파드를 형성하기 위해서 각각의 다른 컬러의 하나의 파드는 함께 그룹화될 수 있다.

시안, 마젠타, 옐로우 및 블랙의 하나의 파드는 크로마파드로 그룹화될 수 있다. 크로마파드는 다른 라인상의 10개의 도트의 동일 수평 세트의 다른 컬러 구성요소를 표현한다. 다른 컬러 파드 사이의 정확한 거리는 일정수의 도트-폭이고, 따라서 인쇄할 때 고려되어야 한다: 인쇄 알고리즘은 컬러 사이의 도트-폭의 가변 거리에 대해서 허용하여야 한다.

이네이블된 파드그룹에서의 노즐의 그룹이 일정 파이어링 페이스 동안에 동시에 파이어링되는 페이스그룹으로 하나 이상의 크로마파드가 형성될 수 있다. 페이스그룹의 파드그룹의 하나 이상은 요구되는 인쇄 속도에 의존해서 동시에 이네이블된다.

5개의 크로마파드가 단일 파드그룹으로 편성될 수 있다. 각각의 크로마파드는 40개의 노즐을 포함할 수 있기 때문에, 각각의 파드그룹은 200개의 노즐(50개의 시안, 50개의 마젠타, 50개의 옐로우, 및 50개의 블랙 노즐)을 포함할 수 있다.

2개의 파드그룹이 단일 페이스그룹으로 편성될 수 있다. 페이스그룹은 페이스그룹내의 노즐의 그룹이 일정 파이어링 페이스 동안에 동시에 파이어링되기 때문에 그렇게 명명된다. 2개의 파드그룹으로부터 페이스그룹의 형성은 2개의 파드그룹 이네이블 라인을 통해서 저속 및 고속 인쇄를 전부 허용한다.

2개의 페이스그룹은 단일 파이어그룹으로 편성될 수 있고, 각각의 세그먼트에는 4개의 파이어그룹이 있다. 파이어그룹은 그들이 모두 동일 노즐을 동시에 파이어링하기 때문에 그렇게 명명된다. 2개의 이네이블 라인은 다른 파이어링 페이스로서 독립적으로 페이스그룹의 노즐의 파이어링을 허용한다.

4-인치 프린트헤드는 전형적으로 8개의 세그먼트로 나란히 구성될 것이고, 각각의 세그먼트는 4개의 파이어그룹을 가질 것이다.

더 넓은 프린트헤드는 프린트헤드의 2개를 함께 조립함으로써 만들어질 수 있다. 그래서 8-인치 프린트헤드는 총 51,200개의 노즐에 대해서 2개의 4-인치 프린트헤드로 구성된다.

노즐 계층은 고른 전력 소비를 유지하면서 오버래핑 페이스 및 다중 속도를 허용한다. 더하여, 노즐 그룹화 파드는 물리적 안정성을 제공한다.

전력 소비의 견지에서, 노즐 그룹화는 속도/전력 소비가 다른 제품 구성에서 교환되게 하기 위해서 저속 및 고속 인쇄 모드를 이네이블한다.

단일 4-인치 프린트헤드는 총 25,600개의 노즐을 포함할 수 있다. 인쇄 사이클은 인쇄될 정보에 의존하여 모든 이들 노즐까지의 파이어링을 수반한다. 그들을 모두 동시에 파이어링하는 것은 너무 많은 전력을 소비하고 잉크 리필 및 노즐 간섭의 견지에서 문제일 것이다. 더 나아가, 노즐의 파이어링은 또한 그 노즐 파드의 공통 잉크통내에서 제한된 시간 동안 음향 섭동을 일으킨다. 섭동은 동일 파드내의 또 다른 노즐의 파이어링과 간섭할 수 있다. 결과적으로, 파드내의 노즐의 파이어링은 가능한 서로로부터 오프셋되어야 한다.

이것을 다루기 위해서, 컬러 당 하나의 노즐이 크로마파드로부터 파이어링될 수 있고 그 후 파드그룹내의 다음 크로마파드로부터 노즐이 파이어링될 수 있다.

2개의 파이어링 모드(저속 인쇄 모드 및 고속 인쇄 모드)가 정의될 수 있다.

저속 인쇄 동안에, 각각의 페이스그룹의 하나의 파드그룹만이 파이어링 펄스를 수신해서 2개중 하나의 파드그룹만이 노즐을 파이어링한다. 저속 인쇄 모드에서 2개의 파드그룹내의 크로마파드는 처음 크로마파드가 다시 파이어링하기 전에 모두 파이어링해야 한다.

저속 인쇄 모드에서, 128개의 노즐이 각각의 4-인치 프린트헤드로부터 동시에 파이어링될 수 있다. 파이어링된 노즐은 최대로 멀어야 하고, 그래서 16개의 노즐이 각각의 8개의 세그먼트로부터 파이어링된다. 모든 25,600개의 노즐을 파이어링하기 위해서, 128개의 노즐의 200개의 다른 세트가 파이어링되어야 한다.

고속 인쇄 동안에, 2개의 파드그룹이 설정되어서, 2개의 파드그룹이 노즐을 파이어링한다. 고속 인쇄 모드에서 단일 파드그룹내의 크로마파드는 처음 크로마파드가 다시 파이어링하기 전에 모두 파이어링해야 한다.

고속 인쇄 모드에서, 256개의 노즐이 각각의 4-인치 프린트헤드로부터 동시에 파이어링될 수 있다. 파이어링된 노즐은 최대로 멀어야 하고, 그래서 32개의 노즐이 각각의 세그먼트로부터 파이어링된다. 모든 25,600개의 노즐을 파이어링하기 위해서, 256개의 노즐의 100개의 다른 세트가 파이어링되어야 한다.

고속 인쇄가 동시에 두 배의 노즐을 파이어링하기 때문에 결과적으로 저속 인쇄는 고속 인쇄보다 두 배의 시간이 걸린다. 저속 모드에서의 전력 소비는 고속 모드의 전력 소비의 반이다. 그러나, 페이지를 인쇄하는데 소비되는 에너지는 두가지 경우에서 동일함을 주목하라.

프린트헤드는 파이어링 펄스의 타이밍을 조절하기 위해서 피드백의 수개의 라인을 산출한다. 하나의 피드백 신호는 프린트헤드가 얼마나 뜨거운지를 제어기에게 알려준다. 이것은 온도가 잉크의 점성에 영향을 미치기 때문에, 제어기가 파이어링 펄스의 타이밍을 조절하게 한다. 제 2 피드백 신호는 얼마나 많은 전압이 액추에이터에 이용될 수 있는지를 제어기에게 알려준다. 이것은 펄스 폭을 조절함으로써 제어기가 플랫 배터리 또는 고전압원에 대해 보상하게 한다. 제 3 피드백 신호는 액추에이터 히터의 저항률(스퀘어 당 옴)을 제어기에게 알려준다. 이것은 히터 저항률에 관계없이 일정한 에너지를 유지하기 위해서 제어기가 펄스 폭을 조절하게 한다. 제 4 피드백 신호는 리소그래픽 및 에칭 편차에 기인하여 $\pm 5\%$ 까지 변할 수 있는, 히터의 임계부의 폭을 제어기에게 알려준다. 이것은 제어기가 펄스 폭을 적절히 조절하게 한다.

로드 사이클은 후속 인쇄 사이클 동안에 인쇄될 정보로써 프린트헤드의 로딩 업을 수반한다. 각각의 노즐의 파이어링 제어 포트는 노즐이 인쇄 사이클 동안에 파이어링할 것인지 아닌지를 결정하는 연관 노즐이네이블 비트를 가질 수 있다. 노즐이네이블 비트는 시프트 레지스터의 세트를 통해서 로드 사이클 동안 로딩된다. 모든 시프트 레지스터가 완전히 로딩되면, 모든 비트는 적합한 노즐이네이블 비트에 병렬로 전송된다. 전송이 일어난 직후, 인쇄 사이클이 시작될 수 있다. 인쇄 사이클 및 로드 사이클은 모든 노즐이네이블 비트의 병렬 로드와 인쇄 사이클의 끝에 일어나는 한 동시에 일어날 수 있다.

인쇄 프로세스는 프린트헤드에 대하여 정확한 순서로 데이터를 산출해야 한다. 예로서, 제 1 클록 펄스는 다음 인쇄 사이클의 도트 0, 800, 1600, 2400, 3200, 4000, 4800, 및 5600에 대하여 CMYK 비트를 전송할 수 있다. 제 2 클록 펄스는 다음 인쇄 사이클의 도트 1, 801, 1601, 2401, 3201, 4001, 4801, 및 5601에 대하여 CMYK 비트를 전송할 수 있다. 800 SR클록 펄스 후에, 전송 펄스가 주어질 수 있다.

물론, 800 SR클록 펄스내에서, 시프트 레지스터는 노즐이네이블 비트로의 최종 전송과의 통신에 따라 로딩되어야 하고, 여기서 많은 다른 배선 가능성이 존재한다. 하나의 로딩(따라서 배선) 가능성은 비트를 파드 순서대로 로딩하는 것인데, 각각의 파드내에서 비트는 파드의 하나의 측으로부터 다른 측으로의 각각의 노즐을 표현한다(제 1 행에서의 제 2 노즐상로의 이동 전에 제 1 행으로부터 마지막 행으로까지 제 1 노즐을 효과적으로 로딩). 2개의 행 파드에서 이것은 외견상 지그-재그 방식으로 노즐을 로딩하는 것을 의미한다. 또 다른 가능성은 비트를 파드 순서대로 로딩하는 것인데, 각각의 파드내에서 비트는 각각의 행을 표현하고, 각각의 행내에서 파드의 하나의 측으로부터 다른 측으로 노즐에서부터 시작한다.

홀수 및 짝수 CMYK 출력은, 동일 인쇄 사이클 동안에 인쇄되더라도, 동일한 물리적 출력 라인상에 나타나지 않는다는 것을 주목하는 것은 중요하다. 다른 컬러의 노즐 사이의 분리뿐만 아니라 프린트헤드내에서의 홀수 및 짝수 노즐의 물리적 분리는 그들이 페이지의 다른 라인상에 도트를 산출할 것을 보장한다. 이러한 상대적 차이는 데이터를 프린트헤드내로 로딩할 때에 대해서 고려되어야 한다. 라인에서의 실제 차이는 프린트헤드에서 사용된 잉크젯 메카니즘의 특성에 의존한다. 그 차이는 다른 컬러의 노즐 사이의 거리 및 동일 컬러의 노즐 사이의 거리를 표현하는 변수에 의해서 정해질 수 있다.

본 발명은 연속톤 컬러 층을 2-레벨로 하프토닝하고 블랙 층을 하프토닝된 연속톤 층 위로 합성시키기 위한 하프토너/합성기 유닛인데:

일련의 연속톤 컬러 화소값의 형태를 갖는 확장된 연속톤 컬러 층 및 일련의 블랙 도트값의 형태를 갖는 확장된 블랙 층을 수신하는 입력 포트,

입력 포트에 수신된 각각의 연속톤 컬러 화소값을 디터링하고 각각의 컬러 구성요소에 대해서 2-레벨 출력 도트의 값을 결정하는 디터 유닛,

디터 유닛으로부터의 2-레벨 출력 도트의 값 및 입력 포트로부터의 블랙 도트값을 수신하고, 예를 들어, 블랙 도트의 값이 완전한 불투명도를 표현할 때 그 때 각각의 컬러에 대해 하프토닝된 도트의 값이 컬러를 표현하지 않게 설정되도록 블랙 층을 하프토닝된 층 위로 합성하는 합성 유닛,

연속톤 컬러 화소 입력, 블랙 도트 입력, 및 도트 출력을 클록킹하기 위한 이네이블 신호를 발생시키는 클록 이네이블 발생기,

분리된 컬러 평면을 가질 수 있는 일련의 2-레벨 도트의 형태를 갖는 2-레벨 이미지 라인의 세트를 배달하는 출력 포트를 포함한다.

출력은 1600 dpi 2-레벨 이미지 라인의 세트일 수 있다.

컬러 연속톤 층은 CMYK 연속톤 층일 수 있다.

입력 연속톤 CMYK FIFO는 완전한 8KB 라인 버퍼를 포함할 수 있다. 각각의 라인은 한번 판독되고 그 후 라인 복제를 통해서 수직 업-스케일링을 성취하기 위해 연속톤 환산 계수(scale factor) 회 사용된다. FIFO 기록 번지 래핑은 라인의 최종 복제된 사용의 시작 때까지 디스에이블되는데, 그 때 클록 이네이블 발생기는 래핑(wrapping)을 이네이블하는 연속톤 라인 전진 이네이블 신호를 발생시킨다.

대안은 주 메모리 연속톤 환산 계수 회로부터 라인을 판독하는 것인데, 메모리 트래픽을 65MB/s 만큼 증가시키지만 온-칩 8KB 라인 버퍼에 대한 필요를 피한다.

유닛에 의해서 산출된 데이터의 소비자는 전형적으로 프린트헤드 인터페이스이다. 프린트헤드 인터페이스는 플래너 포맷, 즉, 분리된 컬러 평면을 갖는 2-레벨 이미지 데이터를 요구하기만 할 수는 없다. 그러나, 또한 그것은 짝수 및 홀수 화소가 분리되도록 요구할 수 있다. 컬러 연속톤 층이 CMYK 연속톤 층일 때 유닛의 출력 단계는 짝수 시안, 홀수 시안, 짝수 마젠타, 홀수 마젠타, 짝수 옐로우, 홀수 옐로우, 짝수 블랙, 및 홀수 블랙에 대하여 각각 하나씩 8 병렬 화소 FIFO를 사용할 수 있다.

이러한 목적으로 클록 이네이블 발생기는 또한 출력 도트 FIFO의 짝수 또는 홀수 세트를 선택하는데 사용되는 짝수 신호를 발생시킬 수 있다.

시작된 후, 유닛은 페이지 끝 조건을 검출할 때까지, 또는 그 제어 레지스터를 통해서 명시적으로 중지될 때까지 계속할 수 있다.

유닛은 지정된 폭 및 길이의 도트의 페이지를 발생시키고 많은 레지스터는 페이지 구조 및 매개 변수를 제어하기 위해서 데이터를 제공하는데 사용될 수 있다.

페이지 폭 레지스터는 프린트헤드의 폭에 대응할 수 있는 페이지 폭 데이터를 수신한다.

페이지 길이 레지스터는 타겟 페이지의 길이에 대응할 수 있는 페이지 길이 데이터를 수신한다.

좌측 여백 레지스터는 좌측 여백의 위치를 기술하는 데이터를 수신한다.

우측 여백 레지스터는 우측 여백의 위치를 기술하는 데이터를 수신한다.

좌측 여백으로부터 우측 여백으로까지의 거리는 타겟 페이지 폭에 대응한다. 하프토너/합성기 유닛은 페이지 폭에 관하여 지정된 좌측 및 우측 여백 사이에 타겟 페이지 데이터를 발생시킨다.

블랙 페이지 폭 레지스터는 블랙 페이지 폭을 기술하는 데이터를 수신한다.

연속톤 페이지 폭 레지스터는 연속톤 페이지 폭을 기술하는 데이터를 수신한다.

하프토너/합성기 유닛은 지정된 블랙 및 연속톤 페이지 폭에 따라 블랙 및 연속톤 데이터를 소비한다.

하프토너/합성기 유닛은 블랙 및 연속톤 데이터를 타겟 페이지 폭에 클리핑한다. 이것은 입력 FIFO 레벨에서 임의의 특별한 라인 끝 로직을 요구함이 없이 블랙 및 연속톤 페이지 폭이 타겟 페이지 폭을 초과하게 한다.

이러한 목적으로 클록 이네이블 발생기는 또한 현재 도트 위치가 페이지의 좌측 또는 우측 여백에 있을 때 화이트 도트를 발생시키는데 사용되는 여백 신호를 발생시킬 수 있다.

하프토너/합성기 유닛은 지정된 환산 계수에 수평 및 수직적으로 기초하여 프린터 해상도로 연속톤 데이터를 스케일링한다. 연속톤 환산 계수 레지스터는 연속톤 환산 계수를 수신하기 위해 제공될 수 있다. 이러한 환산 계수는 하프토너/합성기 유닛을 시작하기 이전에 연속톤 환산 계수 레지스터에 기록되어야 한다.

하프토너/합성기 제어 및 구성 레지스터는 이하의 표에 따라 요약될 수 있다:

하프토너/합성기 제어 및 구성 레지스터		
레지스터	폭	설명
시작	1	하프토너/합성기를 시작
중지	1	하프토너/합성기를 중지
페이지 폭	14	도트에서, 인쇄된 페이지의 페이지 폭. 이것은 각각의 라인에 대해서 발생되어야 하는 도트의 수이다.
좌측 여백	14	도트에서, 좌측 여백의 위치
우측 여백	14	도트에서, 우측 여백의 위치
페이지 길이	15	도트에서, 인쇄된 페이지의 페이지 길이. 이것은 각각의 페이지에 대해서 발생되어야 하는 라인의 수이다.
블랙 페이지 폭	14	도트에서, 블랙 층의 페이지 폭. 블랙 라인의 끝을 검출하는데 사용된다.
연속톤 페이지 폭	14	도트에서, 연속톤 층의 페이지 폭. 연속톤 라인의 끝을 검출하는데 사용된다.
연속톤 환산 계수	4	연속톤 데이터를 2-레벨 해상도로 스케일링하는데 사용되는 환산 계수

디더 셀에서 볼륨의 각각의 도트 열은 256 별개의 비트로서 구현될 수 있다.

대안으로, 볼륨의 각각의 도트 열은 임계값의 고정 세트로서 구현될 수 있다. 3개의 8-비트 임계값을 사용하는 것은, 예를 들어, 24 비트를 소비할 뿐이다. 이제, n 임계값은 $n+1$ 강도 인터발을 정하는데, 그 인터발내에서 대응 디더 셀 위치는 교대로 설정되지 않거나 설정되거나 한다. 디더링된 연속톤 화소값은 $n+1$ 인터발 중 하나를 유일하게 선택하고, 이것은 대응 출력 도트의 값을 결정한다.

연속톤 데이터는 3중-임계값 $64 \times 64 \times 3 \times 8$ -비트(12KB) 디더 볼륨을 사용하여 디더링될 수 있다. 3개의 임계값은 하나의 사이클에서 디더 셀 ROM으로부터 검색될 수 있는 편리한 24-비트 값을 형성한다.

디더 셀 일치가 컬러 평면 사이에 소망된다면, 그 때 동일 3중-임계값이 한번 검색되어 각각의 컬러 구성요소를 디더링하는데 사용될 수 있다.

디더 셀 일치가 소망되지 않는다면, 그 때 디더 셀은 4개의 서브셀로 분할되어 4개의 개별적으로 번지 지정 가능한 ROM에 저장될 수 있는데, 그 ROM으로부터 4개의 다른 3중-임계값이 하나의 사이클에서 병렬로 검색될 수 있다. 4개의 컬러 평면은 서로로부터의 32 도트의 수직 및/또는 수평 오프셋에서 동일 디더 셀을 공유한다.

다중-임계값 디더 유닛이 사용될 수 있다. 예를 들어, 3중-임계값 유닛은 3중-임계값 및 강도값을 인터발 및 일 또는 제로 비트로 변환한다. 3중-임계 규칙은 이하 표에 나타내어진다.

3중-임계 규칙	
인터발	출력
$V \leq T_1$	0
$T_1 < V \leq T_2$	1
$T_2 < V \leq T_3$	0
$T_3 < V$	1

합성 유닛은 블랙 층 도트를 하프토닝된 CMYK 층 도트 위로 합성한다. 블랙 층 불투명도가 일이라면, 그 때 하프토닝된 CMY는 제로로 설정된다.

4-비트 하프토닝된 컬러 $C_cM_cY_cK_c$ 및 1-비트 블랙 층 불투명도 K_b 가 주어질 때, 합성 로직은 이하 표에서 정의되는 바와 같다:

합성 로직	
컬러 채널	조건
C	$C_c \wedge \neg K_b$
M	$M_c \wedge \neg K_b$
Y	$Y_c \wedge \neg K_b$
K	$K_c \vee K_b$

클록 이네이블 발생기는 카운터의 세트를 사용한다. 카운터의 내부 로직은 이하 표에서 정의된다:

클록 이네이블 발생기 카운터 로직					
카운터	약자	폭	데이터	로드 조건	감분 조건
도트	D	14	페이지 폭	$RP^a \vee EOL^b$	$(D > 0) \wedge clk$
라인	L	15	페이지 길이	RP	$(L > 0) \wedge EOL$
좌측 여백	LM	14	좌측 여백	$RP \vee EOL$	$(LM > 0) \wedge clk$
우측 여백	RM	14	우측 여백	$RP \vee EOL$	$(RM > 0) \wedge clk$
짝수/홀수 도트	E	1	0	$RP \vee EOL$	clk
블랙 도트	BD	14	블랙 폭	$RP \vee EOL$	$(LM=0) \wedge (BD > 0) \wedge clk$
연속톤 도트	CD	14	연속톤 폭	$RP \vee EOL$	$(LM=0) \wedge (CD > 0) \wedge clk$
연속톤 서브-화소	CSP	4	연속톤 환산 계수	$RP \vee EOL \vee (CSP=0)$	$(LM=0) \wedge clk$
연속톤 서브-라인	CSL	4	연속톤 환산 계수	$RP \vee (CSL=0)$	$EOL \wedge clk$

^a RP(리셋 페이지) 조건: 외부 신호
^b EOL(라인 끝) 조건: $(D=0) \wedge (BD=0) \wedge (CD=0)$

클록 이네이블 신호의 로직은 이하 표에서 정의된다:

클록 이네이블 발생기 출력 신호 로직	
출력 신호	조건
출력 도트 클록 이네이블	$(D > 0) \wedge \neg EOP^a$
블랙 도트 클록 이네이블	$(LM=0) \wedge (BD > 0) \wedge \neg EOP$
연속톤 화소 클록 이네이블	$(LM=0) \wedge (CD > 0) \wedge (CSP=0) \wedge \neg EOP$
연속톤 라인 전진 이네이블	$(CSL=0) \wedge \neg EOP$
짝수	$E=0$
여백	$(LM=0) \vee (RM=0)$

^a EOP(페이지 끝) 조건: $L=0$

본 발명은 연속톤 컬러 화소값의 어레이의 형태를 갖는 연속톤 컬러 이미지를 2-레벨 도트로 디지털식으로 하프토닝하기 위한 디더 유닛이다. 디더 유닛은 디더 셀 위치가 교대로 설정되지 않고 설정되고 하도록 정해진 $n+1$ 강도 인터발을 정하는 n 임계값의 고정 세트를, 각각의 디더 셀 위치에 대해서, 포함하는 디더 볼륨으로 구성된다.

그러한 디더 유닛은 디더 볼륨을 탐색함으로써 각각의 컬러 구성요소에 대한 2-레벨 출력 도트의 값을 결정함으로써 입력 포트에서 수신된 각각의 연속톤 컬러 화소값을 디더링하도록 동작할 수 있다.

임계값의 세트는 디더 볼륨의 비트 열을 효과적으로 런길이-인코딩하도록, 즉 비트 열을 압축하도록 사용될 수 있다. 이것은 열에서의 인접 비트 사이에 전형적으로 코히어런스가 존재한다는 사실에 의존한다. 임계값의 수가 제한 또는 고정될 때, 그 때 범용 디더 볼륨은 더 이상 반드시 표현될 수 있는 것이 아니고, 그래서 임계값의 제한된 수는 디더 볼륨이 생성될 때 고려될 필요가 있다. 임계값의 수가 하나로 제한될 때, 전통적인 디더 행렬이 얻어진다.

다중-임계값 유닛은 이미지의 각각의 컬러 구성요소에 대해서 제공될 수 있다. 모든 다중-임계값 유닛은 디터 볼륨에 동작적으로 연결되고, 각각의 다중-임계값 유닛은 출력 도트의 위치에 대응하는 디터 셀 위치가 연속톤 값이 유일하게 선택하는 강도 인터발내에 설정되도록 정해지는지를 결정함으로써 연속톤 컬러 화소 구성요소값에 대응하는 출력 도트의 값을 결정한다.

3개의 임계값이 사용될 수 있고, 그것들은 8-비트 임계값일 수 있다.

디터 셀 위치가 컬러 평면 사이에 소망된다면, 그 때 임계값은 한번 검색되어 각각의 컬러 구성요소를 디터링하는데 사용될 수 있다.

디터 셀 위치가 컬러 평면 사이에 소망되지 않는다면, 그 때 디터 셀은 서브셀로 분할되고, 개별적으로 번지 지정 가능한 메모리에 저장될 수 있는데, 그 메모리로부터 다른 다중-임계값이 병렬로 검색될 수 있다.

4개의 컬러 구성요소 연속톤 층이 하프토닝되려는 경우에, 4개의 별개의 3중-임계값 유닛은 각각의 컬러 구성요소에 대하여 일련의 연속톤 컬러 화소값을 각각 수신할 수 있고, 디터 셀 번지 발생기는 디터 유닛의 서브셀로부터 4개의 다른 3중-임계값의 검색을 제어하기 위해서, 각각의 3중-임계값 유닛에 대해, 4개의 4-웨이 멀티플렉서와 관련하여 동작할 수 있다.

3중-임계값 유닛은 이하의 표에 나타내어진 규칙에 따라서 3중-임계값 T_1 , T_2 및 T_3 및 강도 값 V 를 인터발 및 일 또는 제로 비트로 변환할 수 있다.

3중-임계 규칙	
인터발	출력
$V \leq T_1$	0
$T_1 < V \leq T_2$	1
$T_2 < V \leq T_3$	0
$T_3 < V$	1

더 나아간 태양에서 본 발명은 연속톤 컬러 화소값의 어레이의 형태를 갖는 연속톤 컬러 이미지를 2-레벨 도트로 디지털 식으로 하프토닝하기 위한 디터 유닛에 서, 상기 디터 유닛은 디터 볼륨을 포함하고,

상기 디터 셀 위치가 교대로 설정되지 않고 설정되고 하도록 정해진 $n+1$ 강도 인터발을 정하는 n 임계값의 고정 세트에 의해서 상기 디터 볼륨의 각각의 디터 셀 위치를 표현하는 단계, 및

상기 연속톤 화소 구성요소 값에 대응하는 출력 도트의 값을 결정하기 위해서, 상기 $n+1$ 강도 인터발 중 하나를 유일하게 선택하도록 디터링되는 연속톤 화소 구성요소의 값을 사용하는 단계를 포함하는 하프토닝 방법에 있어서,

디터 유닛은, 이미지의 각각의 컬러 구성요소에 대해서, 다중-임계값 유닛을 더 포함하고, 그 모든 다중-임계값 유닛은 디터 볼륨에 동작적으로 연결되고,

상기 출력 도트의 위치에 대응하는 디터 셀 위치가 상기 연속톤값이 유일하게 선택하는 강도 인터발내에 설정되도록 정해지는지를 결정함으로써 연속톤 컬러 화소 구성요소 값에 대응하는 출력 도트의 값을 각각의 상기 다중-임계값 유닛에서 결정하는 단계를 더 포함하는 방법이다.

본 발명은 프린트헤드의 온도를 표시하는 신호를 산출하는 온도 센서, 및 잉크통으로부터 잉크를 받는 잉크 공급 포트, 잉크 도트를 분출하는 잉크 증착 포트, 및 프린트헤드의 파이어링 사이클 동안에 증착 포트로부터 잉크 도트가 증착되어지게 하는 신호를 수신하는 파이어링 제어 포트를 각각 갖는 복수의 노즐을 포함한 유형의 프린트헤드용 프린트헤드 제어기를 구비한 프린터이다. 여기서, 프린트헤드 제어기는 온도 센서로부터 프린트헤드의 온도를 표시하는 신호를 수신하는 수단, 및 노즐의 파이어링 제어 포트에 파이어링 신호를 공급하는 수단을 포함하고, 프린트헤드 제어기는, 각각의 인쇄 작업 전

에, 모든 노즐을 파이어링으로 설정하고, 각각의 노즐에 일련의 짧은 파이어링 펄스를 제공하기 위해 일련의 변형 파이어링 사이클을 실행하도록 동작 가능하고, 프린트헤드의 온도를 표시하는 신호가 동작 평형 온도에 도달했음을 표시할 때까지 각각의 펄스의 지속시간은 방울을 파이어링하기에는 불충분하지만 잉크를 가열시키기에는 충분하다.

예열 모드 동안의 피드백이 T센스(아래에 정의됨)에 의해서 제공될 수 있고, 실온보다 높은 약 30℃의 온도에 도달할 때까지 계속된다. 시간 또는 온도는 잉크 구성물에 의존하여 변할 수 있기 때문에, 평형 온도에 도달하는 때에 관하여 온도 정보가 피드백되게 하는 것이 중요하다.

일 예에서, 각각의 노즐에 대해서 약 200 펄스가 요구된다.

예열 모드의 지속시간은 약 50 밀리초일 수 있고, 잉크 구성물에 의존한다.

예열은 데이터가 프린터에 전송되고 있는 동안에 이루어지기 때문에 성능에 영향을 미치지 않는다.

예열 사이클은 1을 갖는 모든 노즐로의 단일 로드 사이클(즉, 모든 노즐을 파이어링으로 설정), 및 각각의 노즐로의 많은 짧은 파이어링 펄스를 수반한다. 펄스의 지속시간은 방울을 파이어링하기에는 불충분하지만 잉크를 가열시키기에는 충분해야 한다. 펄스의 지속시간은 프린터에서 사용되는 각각의 잉크에 대해서 변화할 수 있다. 전체적으로 각각의 노즐에 대해서 약 200 펄스가 요구되고, 표준 인쇄 사이클로서 동일 시퀀스에서 사이클링 스루한다.

본 발명은 컴퓨터 메모리에 유지된 자원, 그 자원으로의 동시 액세스를 요구하는 다중 병렬 프로세서, 및 그 자원내에 좌표를 발생시키는 자원 번지 발생기인데, 자원은 다른 파트로 분할되고 각각의 파트는 다른 메모리 बैं크에 저장되고, 자원 번지 발생기는 각각의 프로세서에 의해서 사용되는 자원의 파트를 선택하는데 사용되는 좌표를 발생시키고, 그 선택은 각각의 파트가 한번에 하나의 프로세서에 의해서 사용되지만 하는 것을 보장하도록 배치된다. 또한 선택은 파트가 자원에 나타나는 순서대로 각각의 프로세서가 파트를 사용하는 것을 보장할 수 있다.

자원은 연속톤 컬러 화소값의 어레이의 형태를 갖는 연속톤 컬러 이미지를 2-레벨 도트로 디지털식으로 하프토닝하기 위한 다중-임계값 디터 행렬일 수 있다. 상기 디터 행렬은 디터 셀 위치가 교대로 설정되지 않고 설정되고 하도록 정해진 $n+1$ 강도 인터발을 정하는 n 임계값의 고정 세트를, 각각의 디터 셀 위치에 대해서, 포함할 수 있다.

자원으로의 동시 액세스를 요구하는 다중 병렬 프로세서는 이미지의 각각의 컬러 구성요소에 대해서 제공되는 다중-임계값 유닛일 수 있다. 모든 다중-임계값 유닛은 다중-임계값 디터 행렬에 동작적으로 연결되고, 각각의 다중-임계값 유닛은 상기 출력 도트의 위치에 대응하는 디터 셀 위치가 연속톤값이 유일하게 선택하는 강도 인터발내에 설정되도록 정해지는 지를 결정함으로써 연속톤 컬러 화소 구성요소 값에 대응하는 출력 도트의 값을 결정한다.

그러한 디터 유닛은 다중-임계값 디터 행렬을 탐색함으로써 각각의 컬러 구성요소에 대한 2-레벨 출력 도트의 값을 결정함으로써 입력 포트에서 수신된 각각의 연속톤 컬러 화소값을 디터링하도록 동작할 수 있다.

디터 셀은 서브셀로 분할되고, 개별적으로 번지 지정 가능한 메모리에 저장될 수 있는데, 그 메모리로부터 다른 다중-임계값이 병렬로 검색될 수 있다.

4개의 컬러 구성요소 연속톤 층이 하프토닝되려는 경우에, 4개의 별개의 3중-임계값 유닛은 각각의 컬러 구성요소에 대하여 일련의 연속톤 컬러 화소값을 각각 수신할 수 있고, 디터 셀 번지 발생기는 디터 행렬의 4개의 다른 서브셀로부터 4개의 다른 3중 임계값의 검색을 제어하기 위해서, 각각의 임계값 유닛에 대해, 4개의 4-웨이 멀티플렉서와 관련하여 동작한다.

번지 발생기는, 컬러 평면 사이의 디터 셀 위치가 소망되지 않는다면, 어떠한 2개의 임계값 유닛도 동일 서브셀로의 동시 액세스를 요구하지 않도록 쉽게 배치될 수 있다.

더 나아가 태양에서, 본 발명은 컴퓨터 메모리에 유지된 자원에 액세스하는 방법에 있어서, 다중 병렬 프로세서는 자원으로의 동시 액세스를 요구하고, 그 자원내에 좌표를 발생시키는 자원 번지 발생기가 존재하고,

자원을 다른 파트로 분할하는 단계;

각각의 파트를 다른 메모리 बैं크에 저장하는 단계; 및

각각의 프로세서에 의해서 사용되는 자원의 파트를 선택하는데 사용되는 좌표를 발생시키기 위해서 자원 번지 발생기를 동작시키는 단계를 포함하는 방법인데, 선택은 각각의 파트가 한번에 하나의 프로세서에 의해서 사용되지만 하는 것을 보장하도록 배치된다.

또한 선택은 자원에 파트가 나타나는 순서대로 각각의 프로세서가 파트를 사용하는 것을 보장할 수 있다.

자원은 연속톤 컬러 화소값의 어레이의 형태를 갖는 연속톤 컬러 이미지를 2-레벨 도트로 디지털식으로 하프토닝하기 위한 다중-임계값 디더 행렬일 수 있다. 상기 디더 행렬은 디더 셀 위치가 교대로 설정되지 않고 설정되고 하도록 정해진 $n+1$ 강도 인터발을 정하는 n 임계값의 고정 세트를, 각각의 디더 셀 위치에 대해서, 포함할 수 있다.

자원으로의 동시 액세스를 요구하는 다중 병렬 프로세서는 이미지의 각각의 컬러 구성요소에 대해서 제공되는 다중-임계값 유닛일 수 있다. 모든 다중-임계값 유닛은 다중-임계값 디더 행렬에 동작적으로 연결되고, 각각의 다중-임계값 유닛은 출력 도트의 위치에 대응하는 디더 셀 위치가 상기 연속톤값이 유일하게 선택하는 강도 인터발내에 설정되도록 정해지는 지를 결정함으로써 연속톤 컬러 화소 구성요소 값에 대응하는 출력 도트의 값을 결정한다.

디더 셀은 서브셀로 분할되고, 개별적으로 번지 지정 가능한 메모리에 저장될 수 있는데, 그 메모리로부터 다른 다중-임계값이 병렬로 검색된다.

4개의 컬러 구성요소 연속톤 층이 하프토닝되려는 경우에, 4개의 별개의 3중-임계값 유닛은 각각의 컬러 구성요소에 대하여 일련의 연속톤 컬러 화소값을 각각 수신할 수 있고, 디더 셀 번지 발생기는 디더 행렬의 4개의 다른 서브셀로부터 4개의 다른 3중 임계값의 검색을 제어하기 위해서, 각각의 임계값 유닛에 대해, 4개의 4-웨이 멀티플렉서와 관련하여 동작한다.

디더 셀 번지 발생기는 디더 셀의 크기를 법으로(modulo) 하여 현재 도트 좌표를 반영하는 디더 셀 좌표를 발생시킬 수 있다. 디더 셀 좌표의 하위 비트는 각각의 디더 서브셀내에 위치를 번지 지정하는데 사용될 수 있고, 디더 셀 좌표의 상위 비트는 각각의 다중-임계값 유닛에 의해서 어느 디더 서브셀이 사용되는지를 선택하는데 사용될 수 있고, 선택은 각각의 서브셀이 한번에 하나의 다중-임계값 유닛에 의해서 사용되지만 하는 것을 보장하고, 디더 서브셀이 디더 셀에 나타나는 순서대로 각각의 다중-임계값 유닛이 디더 서브셀을 사용하는 것을 보장하도록 배치된다.

본 발명은 잉크 공급 포트, 및 잉크 증착 포트 및 프린트헤드의 파이어링 사이클 동안에 증착 포트로부터 잉크 도트가 증착되어지게 하는 신호를 수신하는 파이어링 제어 포트를 각각 갖는 복수의 노즐을 포함한 유형의 프린트헤드를 갖는 프린터에서 사용하기 위한 프린터 시스템이다. 또한 프린터는 상기 노즐의 잉크 공급 포트에 잉크를 공급하기 위한 (카트리지와 같은) 잉크통을 포함하고, 상기 프린터 시스템은 상기 잉크통의 잉크가 바닥날 때를 예측하도록 동작한다. 프린터 시스템은 프린터에 의해서 인쇄될 또는 인쇄된 잉크의 도트의 수를 표시하는 신호를 수신하는 입력 포트, 상기 잉크통에 남아있는 잉크의 도트의 수를 표현하는 데이터가 저장되는 메모리, 및 상기 입력 포트로부터 신호를 수신하고 인쇄가 일어날 때 상기 메모리를 업데이트하는 프로세서를 포함한다. 프린터에 의해서 인쇄된 잉크의 도트의 수를 표시하는 신호는 상기 노즐의 파이어링 제어 포트에서 수신된 신호와 관련된다.

페이지의 소정 수의 끝에서, 또 다른 페이지를 인쇄하기에 충분한 잉크가 잉크통에 남아있는지를 표시하는 것이 발생할 수 있다.

충분한 잉크가 남아있지 않다면 새로운 페이지는 인쇄될 수 없다.

인쇄될 또는 인쇄된 도트의 수를 표시하는 신호는 잉크통, 프린트헤드 그 자체, 또는 프린트헤드를 위한 제어기로부터 유도될 수 있다.

도트 카운터는 프린트헤드로부터 파이어링된 잉크의 도트의 수의 카운트를 보전할 수 있다. 도트 카운트는 주기적으로 클리어링될 수 있다. 도트 카운터는 도트가 인쇄될 때마다 신호를 수신하는 입력 포트, 클록 및 클리어 입력 터미널을 갖는 비트 카운터, 카운터 주위의 포지티브 피드백 접속부, 및 현재 카운트를 표시하는 카운터로부터의 출력을 가질 수 있다.

인코더는 수신된 신호를 변환하기 위해서 도트 카운터로의 입력 라인에 놓일 수 있다.

하나 이상의 잉크통을 포함하는 프린터의 경우에, 프린터 시스템은 프린터에 의해서 인쇄될 또는 인쇄된 그 잉크통으로부터 잉크의 도트의 수를 표시하는 신호를 수신하기 위해서 각각의 잉크통에 각각인 입력 포트를 포함할 수 있고, 메모리는 각각의 잉크통에 남아있는 잉크의 도트의 수를 표현하는 데이터를 저장할 수 있다.

또 다른 태양에서 본 발명은 상기 잉크통의 잉크가 바닥날 때를 예측하기 위한 프린터 시스템을 동작시키는 방법에 있어서,

프린터에 의해서 인쇄될 또는 인쇄된 잉크의 도트의 수를 표시하는 신호를 수신하는 단계;

잉크통에 남아있는 잉크의 도트의 수를 표현하는 데이터를 저장하는 단계; 및

인쇄가 일어날 때 메모리를 업데이트하는 단계를 포함하는 방법인데, 프린터에 의해서 인쇄될 또는 인쇄된 잉크의 도트의 수를 표시하는 신호는 상기 노즐의 파이어링 제어 포트에서 수신된 신호와 관련된다.

페이지의 소정 수의 끝에서, 또 다른 페이지를 인쇄하기에 충분한 잉크가 잉크통에 남아있는지에 관하여 표시하는 것이 발생될 수 있다. 충분한 잉크가 남아있지 않다면 새로운 페이지는 인쇄될 수 없다.

본 발명은:

이용될 수 있는 전압을 표시하는 신호를 수신하는 제 1 입력 포트,

파이어링 펄스에 대하여 소정 지속시간을 표현하는 제어 신호를 발생시키기 위해서, 제 1 입력 포트에서 수신된 신호에 의해 인덱싱된, 프로그래밍 가능한 펄스 지속시간 테이블, 및

제어 신호를 송신하는 출력 포트를 포함하는, 프린터용 파이어링 펄스 지속시간 제어 시스템이다.

제 2 입력 포트가 프린터에서의 프린트헤드의 온도를 표시하는 신호를 수신하기 위해서 제공될 수 있다. 이 경우에 프로그래밍 가능한 펄스 지속시간 테이블은 제어 신호를 발생시키기 위해 제 1 및 제 2 입력 포트로부터 수신된 신호에 의해서 인덱싱된다.

펄스 지속시간의 테이블은 더 낮은 비용의 전원공급 장치의 사용을 허용하고, 더 정확한 방울 분출을 보전하는데 도움이 된다.

테이블에서의 엔트리(entry; 항목)는 0-4 μ s의 범위에서의 값을 표현할 수 있고 제어 출력은 인덱싱된 엔트리에 따라서 발생된다.

펄스 폭 발생기는, 사용시, 제어 출력을 수신하고 프린트헤드용 파이어링 펄스를 발생시키기 위해서 제어 시스템의 흐름의 아래에 놓일 것이다.

테이블은 인쇄 런의 제 1 페이지를 인쇄하기 전에 기록될 수 있다. 소망된다면, 테이블은 인쇄 런의 페이지 사이에서 업데이트될 수 있다.

테이블에서의 각각의 엔트리는 이하의 특징의 하나 이상으로부터의 가중치를 포함할 수 있다:

- 사용자 휘도 설정
- 잉크의 점성 커브
- T센스는 프린트헤드가 얼마나 뜨거운지를 제어기에 알려준다. 이것은 온도가 잉크의 점성에 영향을 미치기 때문에, 제어기가 파이어링 펄스의 타이밍을 조절하게 한다.
- V센스(Vsense)는 얼마나 많은 전압이 액추에이터에 이용될 수 있는지를 제어기에 알려준다. 이것은 제어기가 펄스 폭을 조절함으로써 플랫 배터리 또는 고전압원에 대해 보상하게 한다.

· R센스(Rsense)는 액추에이터 히터의 저항률(스퀘어 당 옴)을 제어기에 알려준다. 이것은 히터 저항률에 관계없이 일정 에너지를 보전하기 위해서 제어기가 펄스 폭을 조절하게 한다.

· W센스(Wsense)는 리소그래픽 및 에칭 편차에 기인하여 $\pm 5\%$ 까지 변할 수 있는, 히터의 임계부의 폭을 제어기에 알려준다. 이것은 제어기가 펄스 폭을 적절히 조절하게 한다.

펄스 지속시간 테이블은 256 엔트리를 가질 수 있고, 각각의 엔트리는 8-비트일 수 있다. 전압을 표시하는 신호는 V센스로부터 올 수 있고, 온도를 표시하는 신호는 T센스로부터 올 수 있다. 이 경우에 입력 포트에서 수신된 신호는 테이블을 인덱싱하는데 사용하기 위해서 변환될 수 있다. 테이블로의 8-비트 엔트리는 2개의 4-비트 넘버에 의해서 인덱싱될 수 있고, 그래서 상위 4-비트는 V센스로부터 올 수 있고, 하위 4-비트는 T센스로부터 올 수 있다.

더 나아간 태양에서 본 발명은:

제 1 입력 포트;

프로그래밍 가능한 펄스 지속시간 테이블; 및

출력 포트를 포함하는, 프린터용 파이어링 펄스 지속시간 제어 시스템에 관한 것이고:

이용될 수 있는 전압을 표시하는 제 1 신호를 수신하는 단계;

파이어링 펄스에 대하여 소정 지속시간을 표현하는 제어 신호를 발생시키기 위해서 제 1 신호에 의해 프로그래밍 가능한 펄스 지속시간 테이블을 인덱싱하는 단계; 및

제어 신호를 송신하는 단계를 포함하는, 파이어링 펄스 지속시간 제어 신호를 발생시키는 방법에 관한 것이다.

(실시예)

본 발명을 구체화하는 프린터의 예가 첨부 도면을 참조하여 이제 설명될 것이다.

1 서설

본 발명은 사진같은-양질의 이미지 재현과 잡지같은-양질의 텍스트 재현을 조합한 고성능 컬러 프린터와 관련하여 설명될 것이다. 프린터는 인치 당 1600 도트(dpi) 2-레벨 CMYK(시안, 마젠타, 옐로우, 블랙)를 산출하는 8" 페이지-폭 드롭-온-디맨드형 마이크로전자기계식 잉크젯("멤젯(Memjet)") 프린트헤드를 이용한다. 그것은 분 당 30 완전-컬러 A4 또는 레터 페이지를 인쇄하고, 엔트리-레벨 데스크톱 프린터로서 의도된다. 프린터는 아이프린트로서 일컬어져왔고 이하 설명에서 그 이름으로 나타내어질 것이다.

1.1 동작상의 개관

아이프린트는 2-레벨 블랙을 직접 사용하여 블랙 텍스트 및 그래픽스를, 디터링된 2-레벨 CMYK를 사용하여 연속-톤(contone) 이미지 및 그래픽스를 재현한다. 실제적으로, 아이프린트는 800 dpi의 블랙 해상도, 및 인치 당 267 화소(ppi)의 연속톤 해상도를 지원한다.

아이프린트는, 사용시, 비교적 저속(1.5MBytes/s) 범용 직렬 버스(USB) 접속부를 통해서 워크스테이션 또는 퍼스널 컴퓨터(PC)에 부속된다[14]. 아이프린트는 연속톤 화소 및 블랙 도트의 레벨로 각각의 페이지를 렌더링하기 위해서 PC에 의존한다. PC는 프린터로의 2초 이하의 배달에 대하여 3MB보다 더 작게 각각의 렌더링된 페이지를 압축한다. 아이프린트는 페이지를 압축해제하고 멤젯 프린트헤드의 속도로 한 라인씩 페이지를 인쇄한다. 아이프린트는 2개의 압축된 페이지(6MB)에 대해서 충분한 버퍼 메모리를 포함하고, 그것이 다음 페이지를 받는 동안 하나의 페이지를 인쇄하게 하지만, 그러나 압축되지 않은 페이지(119MB)에 대해서는 단일 페이지이더라도 충분한 버퍼 메모리를 포함하지 않는다.

1.2 페이지 폭

표준 맷젯 노즐 레이아웃은 반-인치 유닛 셀을 갖고, 그래서 반인치의 배수인 페이지 폭에 보통 적응될 수 있다. 임의의 페이지 폭은 그러한 특수화를 정당화하는 시장에서, 고객 노즐 레이아웃으로써 성취될 수 있다. 초기 맷젯 빌딩 블록은 6-인치 실리콘 웨이퍼를 효과적으로 사용하는 널리 유용한 4-인치 프린트헤드이다. 따라서 아이프린트 디자인은 함께 결합된 2개의 4-인치 프린트헤드로 구성된 8-인치 맷젯 프린트헤드를 가정한다. A4/레터 페이지상의 완전한 블리드(bleed)를 달성하기 위한 더 넓은 프린트헤드의 사용은 아이프린트 디자인의 몇 가지 면(특히 정확한 기계적 디자인 및 프린트헤드 인터페이스의 로직)에만 영향을 미친다.

2 맷젯-기반 인쇄

맷젯 프린트헤드는 1600 dpi 2-레벨 CMYK를 산출한다. 저-확산 용지상에서, 각각의 분출된 방울은 거의 완벽히 22.5 미크론 직경 원형 도트를 형성한다. 도트는 격리되어 쉽게 산출되고, 분산-도트 디터링이 그 최대한으로 활용되게 한다. 맷젯 프린트헤드는 페이지-폭이고 일정 용지 속도로 동작하기 때문에, 4개의 컬러 평면은 완벽히 일치되어 인쇄되고, 이상적 도트-온-도트 인쇄를 허용한다. 결과적으로 컬러 평면 사이에 어떠한 공간 상호작용도 존재하지 않기 때문에, 각각의 컬러 평면에 대해서 동일 디터 행렬이 사용된다.

페이지 레이아웃은 이미지, 그래픽스 및 텍스트의 혼합물을 포함할 수 있다. 연속톤 이미지 및 그래픽스는 확률적 분산-도트 디터를 사용하여 재현된다. 클러스터-도트(또는 진폭-변조) 디터와는 다르게, 분산-도트(또는 주파수-변조) 디터는 도트 해상도의 거의 한계까지로 높은 공간 주파수(즉, 이미지 상세)를 재현하고, 동시에 그 완전한 컬러 깊이까지로 더 낮은 공간 주파수를 재현한다. 확률적 디터 행렬은 이미지를 가로질러 타일링될 때 불쾌한 저-주파수 패턴이 없도록 신중히 디자인된다. 그러한 것으로서 전형적으로 그 크기는 많은 강도 레벨을 지원하기 위해서 요구되는 최소 크기(즉, 257 강도 레벨에 대하여 $16 \times 16 \times 8$ 비트)를 초과한다. 아이프린트는 $64 \times 64 \times 3 \times 8$ 비트의 디터 볼륨을 사용한다. 그 볼륨은 도트가 강도 범위를 통해서 상태를 (종래 디터 행렬에서와 같이 단 한번보다는) 다수 번 변화시키게 함으로써 디터의 디자인 동안에 여분의 자유도를 제공한다.

인간 대비 감도는 시각 필드의 정도 당 약 3 사이클의 공간 주파수에서 최고를 이루고 그 후 대수적으로 떨어지고, 100배 만큼 감소하고 정도 당 약 40 사이클 이상에서는 측정하기가 어려워진다[2]. 400mm 및 250mm 사이의 통상 시거리에서, 이것은 대략 인쇄된 페이지상의 인치 당 150-250 사이클(cpi), 또는 나이퀴스트(Nyquist) 이론에 따른 인치 당 300-500 샘플로 번역된다. 컬러 감도가 그레이스케일 감도보다 덜 예리하다는 사실을 고려할 때, 인치 당 약 400 화소(ppi) 이상의 연속톤 해상도는 따라서 제한된 유틸리티이고, 실제로 디터를 통해서 컬러 에러에 다소 기여한다.

블랙 텍스트 및 그래픽스는 2-레벨 블랙 도트를 직접 사용하여 재현되고, 따라서 인쇄되기 전에 안티에일리어싱(즉, 로-패스 필터링)되지 않는다. 따라서, 텍스트는 공간적으로 적분될 때 매끄러운 에지를 산출하기 위해서, 상기 시각 한계이상에서 슈퍼샘플링된다. 약 1200 dpi까지의 텍스트 해상도는 지각되는 텍스트 선명도에 계속 기여한다(물론, 저-확산 용지를 가정).

3.1 제약

USB(범용 직렬 버스)는 새로운 PC상의 표준 저속 주변장치 접속부이다[4]. 표준 고속 주변장치 접속부인 IEEE 1394가 추천되지만 불행히도 PC 99 명세에서는 여전히 선택적이고[5], 그래서 아이프린트가 처음 나올 때 널리 보급되어 사용될 수 없다. 따라서 아이프린트는 USB를 통해서 퍼스널 컴퓨터(PC) 또는 워크스테이션에 연결되고, 따라서 USB 접속부의 속도는 아이프린트 시스템의 아키텍처에 가장 중요한 제약을 부과한다. 30 페이지/분의 지속 인쇄 속도에서, 1.5MByte/s에서의 USB는 3MB/페이지의 평균 한계를 부과한다. 페이지의 인쇄 동안에 맷젯-기반 프린터를 인터럽팅하는 행위는 가시적인 불연속을 산출하기 때문에, 버퍼 언더런(underrun)의 가능성을 제거하기 위해서, 프린터가 인쇄를 개시하기 전에 전체 페이지를 받는 것이 유리하다. 프린터는 제한된 버퍼 메모리(즉, 2 페이지 분량 또는 6MB)만을 포함할 수 있기 때문에, 그 때 3MB/페이지 한계는 절대적으로 고려되어야 한다.

도 1은 프린터에서 2중-버퍼링으로써 달성 가능한 지속 인쇄 속도를 예시한다. 제 1 단계(1)는 제 1 페이지가 PC에서 렌더링되도록 요구하고, 이것은 2초까지 걸린다. 제 2 단계(2) 동안에 다음 페이지가 렌더링되고 제 1 페이지는 프린터에 전송되고, 다시 이것은 2초까지 걸린다. 제 3 단계(3)에서 제 1 페이지는 인쇄되고, 제 2 페이지는 전송되고 제 3 페이지는 렌더링되고, 이것은 2초가 걸린다. 결과로서 제 1 페이지가 인쇄되는데 6초까지 걸리고 그 이후 2초마다 일 페이지가 인쇄될 수 있다.

다른 데스크톱 접속부 옵션은 USB에 마찬가지로 대역폭을 제공하고, 그래서 아키텍처에 마찬가지로 제약을 부과한다. 이들은 2MB/s에서 병렬 포트, 및 약 1MB/s에서 10베이스-T 이더넷을 포함한다.

3.2 페이지 렌더링 및 압축

페이지 렌더링(또는 래스터화)은 다양한 방식으로 PC 및 프린터 사이에서 분할될 수 있다. 특정 프린터는 포스트스크립트와 같은 완전한 페이지 기술 언어(PDL)를 지원하고, 대응되게 정교한 렌더러(renderer)를 포함한다. 다른 프린터는 높은 텍스트 해상도를 달성하기 위해서, 텍스트를 렌더링하는데만 전문적 지원을 제공한다. 이것은 통상 내장 또는 다운로드 가능한 폰트에 대한 지원을 포함한다. 각각의 경우에서 내장된 렌더러의 사용은 PC에 대해 렌더링 부담을 감소시키고 PC로부터 프린터로 송신되는 데이터의 양을 감소시킨다. 그러나, 이것은 값이 비싸진다. 이들 프린터는 보통보다 더 복잡하고, PC의 그래픽스 시스템에 대한 완전한 지원을 종종 제공할 수 없으며, 그것을 통하여 애플리케이션 프로그램은 페이지를 작성하고, 렌더링하고 인쇄한다. 종종 그들은 현재 PC의 고성능을 활용하지 못하고, PC 성능에서의 계획된 지수적 성장을 레버리징할(leverage) 수 없다.

아이프린트는 페이지, 즉, 연속톤 이미지와 그래픽스를 화소 레벨로, 블랙 텍스트와 그래픽스를 도트 레벨로 렌더링하기 위해서 PC(4)에 의존한다. 아이프린트(5)는 연속톤 데이터를 디터링하고 그 결과와 임의의 전경 2-레벨 블랙 텍스트 및 그래픽스를 조합하는 단순 렌더링 엔진만을 포함한다. 이러한 전략은 프린터가 단순하고, 임의의 페이지 기술 언어 또는 그래픽스 시스템으로부터 독립되게 유지한다. 그것은 현재 PC의 고성능을 완전히 활용한다. 이러한 전략의 다운사이드는 PC로부터 프린터로 송신되어야 하는 잠재적으로 많은 양의 데이터이다. 결과적으로 이러한 데이터를 30페이지/분의 지속 인쇄 속도를 허용하는데 요구되는 3MB/페이지로 감소시키기 위해서 압축이 사용된다.

도 2는 애플리케이션(6)으로부터 인쇄된 페이지(7)로의 개념적 데이터 흐름을 예시하는 흐름도이다.

8"×11.7" A4 페이지는 1600 dpi에서 114.3MBytes의 2-레벨 CMYK 페이지크기 및 300 ppi에서 32.1MB의 연속톤 CMYK 페이지크기를 갖는다.

프린터 드라이버(8)에서, 연속톤 데이터를 압축하기 위해서 JPEG 압축(9)이 사용된다. JPEG는 고유 손실이 있기는 하지만, 10:1 이하의 압축비에 대해서 그 손실은 통상 무시할 수 있다[16]. 연속톤 대 2-레벨 정수비를 얻고 약간의 압축 여유를 제공하기 위해서, 267 ppi의 연속톤 해상도가 선택된다. 이것은 25.5MB의 연속톤 CMYK 페이지크기, 3MB/페이지 한계내에 맞는 8.5:1의 대응 압축비, 및 각각의 디멘션에서 1:6의 연속톤 대 2-레벨비를 산출한다.

프린터 해상도(1600 dpi)에서 래스터화된 블랙 텍스트(및/또는 그래픽스)의 완전한 페이지는 28.6MB의 2-레벨 이미지를 산출한다. 1600 dpi에서 텍스트를 래스터화하는 것은 적은 이득에 대해서 PC에 무거운 부담을 주기 때문에, 완전히 받아들일 수 있는 800 dpi에서 텍스트를 래스터화하도록 선택한다. 이것은 7.1MB의 2-레벨 이미지를 산출하고, 3MB/페이지 한계내에 맞게 하기 위해서 2.5:1보다 더 적은 무손실 압축비를 요구한다. 일반적으로 10에서 모두 표시된, 그룹 4 팩시밀리로부터 적응된 2-디멘셔널 압축 방법으로써 이것을 달성한다.

페이지의 이미지 및 텍스트 영역이 오버래핑하지 않는 한, 2개의 임의의 조합은 3MB 한계내에 맞는다. 텍스트가 배경 이미지의 상부에 놓여 있다면, 그 때 최악의 경우는 압축된 페이지크기가 (실제의 텍스트 압축비에 의존해서) 6MB에 근접하는 것이다. 이것은 프린터의 페이지 버퍼 메모리내에 맞지만, 프린터에서 페이지의 2중-버퍼링을 방지하고, 따라서 프린터의 페이지 속도를 2/3만큼, 즉 10페이지/분으로 감소시킨다.

3.3 페이지 확장 및 인쇄

상기된 바와 같이, PC는 연속톤 이미지와 그래픽스를 화소 레벨로, 및 블랙 텍스트와 그래픽스를 도트 레벨로 렌더링한다. 이들은 다른 수단에 의해서 압축되고(11) 프린터에 함께 송신된다.

프린터는 2개의 3MB 페이지 버퍼 - PC로부터 수신되는 페이지에 대하여 1개(12), 및 인쇄되는 페이지에 대하여 1개(13)를 포함한다. 프린터는 압축된 페이지를 인쇄되고 있을 때 확장한다. 이러한 확장은 267 ppi 연속톤 CMYK 이미지 데이터를 압축해제(14), 결과적 연속톤 화소를 1600 dpi 2-레벨 CMYK 도트로 하프토닝(15), 800 dpi 2-레벨 블랙 텍스트 데이터를 압축해제(16), 및 결과적 2-레벨 블랙 텍스트 도트를 대응하는 2-레벨 CMYK 이미지 도트 위로 합성(17)하는 것으로 구성된다.

애플리케이션으로부터 인쇄된 페이지로의 개념적 데이터 흐름이 도 2에 예시된다.

4 프린터 하드웨어

페이지 폭 멤젯 프린트헤드의 단순함으로 인해, 아이프린트는 매우 소형이다. 그것은 닫혔을 때 겨우 폭 270mm×깊이 85mm×높이 77mm이다. 도 3은 닫혔을 때 아이프린트(21)의 회화도이다.

커버(22)는, 도 4에 도시된 바와 같이, 열려서 용지 트레이의 파트를 형성한다. 제 2 파트(23)는 커버(22)내에 경첩식으로 움직여지고, 열려서 용지 트레이를 연장한다. 용지 출구 트레이(24)는 프린터의 정면으로부터 슬라이딩 가능하게 연장될 수 있다.

커버(22)가 열릴 때 보이는 정면 패널(25)은 사용자 인터페이스 - 사용자 버튼(26) 및 전력 표시기 LED(27), 용지 공급 버튼(28), 및 용지 없음(29) 및 잉크 부족(30) LED를 포함한다.

4.1 용지 통로

아이프린트는 표준 용지 이송 메카니즘을 사용한다. 용지 통로(50)가 도 5에서 예시되는데, 단일 스테퍼 모터(51)가 낱장 공급 롤러(52) 및 용지 이송을 모두 구동시킨다. 정방향으로 움직일 때 스테퍼 모터는 액티브 용지 통로의 시작 및 끝에서 용지 드라이브 롤러(53) 및 핀치 휠(54)을 각각 구동시킨다. 역으로 될 때, 스테퍼 모터는 낱장 공급기로부터 최상부의 낱장을 붙잡고 있는 낱장 공급 롤러(52)를 구동시키고 그 낱장을 용지 드라이브 롤러(53)에 짧은 거리로 이송하는데, 낱장은 기계적 매체 센서(55)에 의해서 검출된다.

용지 센터링 슬라이더(56)는 용지가 센터링되는 것을 보장한다. 이것은 단일 센터링된 매체 센서가 낱장을 검출하는 것을 보장하고, 또한 프린트헤드보다 더 넓은 낱장이 균형잡힌 여백으로써 인쇄되는 것을 보장한다.

4.1.1 멤젯 프린트헤드

교체 가능한 멤젯 프린트헤드 카트리지(60)가 또한 도 5에 도시된다. 이것은 아이프린트와 같은 제품에서 잉크 카트리지와 관련하여 프린트헤드를 배치하기 위한 4개의 가능한 방법 중 하나를 표현한다:

- 영구 프린트헤드, 교체 가능한 잉크 카트리지(여기서 도시된 바와 같음)
- 분리된 교체 가능한 프린트헤드 및 잉크 카트리지
- 리필 가능한 조합된 프린트헤드 및 잉크 카트리지
- 처분 가능한 조합된 프린트헤드 및 잉크 카트리지

도 6에서의 절단 회화도 및 도 7에서의 단면도에서 예시된, 프린트헤드 캡핑 메카니즘(62) 및 프린트헤드 어셈블리(61)가 프린트헤드 카트리지(60) 아래에 있다. 사용되지 않을 때, 멤젯 프린트헤드(63)는 잉크로 채워져 있고, 그래서 노즐을 통한 잉크의 증발을 방지하기 위해서 캡핑되어야 한다. 잉크 증발은 노즐 동작을 손상시킬 수 있는 잉크 구성요소의 점진적 증착을 초래할 수 있다.

아이프린트는 엘라스토메릭 시일(65) 및 스폰지(66)를 갖는 캡핑 몰딩(64)으로 구성되는 기계적 페이지 폭 캡핑 메카니즘(62)을 포함한다. 프린트헤드가 사용되지 않고 있을 때, 캡핑 몰딩(64)은 프린트헤드 어셈블리(61)의 면에 대어서 스프링에 의해 유지되고, 엘라스토메릭 시일(65)은 프린트헤드 어셈블리의 면에 합치하고 프린트헤드(63) 주위에 밀폐된 시일을 생성한다. 스폰지(66)는 프린트헤드 클리닝 사이클 동안에 분출된 방울을 붙잡는데 사용된다. 프린트헤드가 사용되고 있을 때, 캡핑 몰딩(64)은 프린트헤드 어셈블리(61)로부터 떨어져서 용지 통로로부터 떨어져서 유지된다.

캡핑 몰딩(64)은 봉(69)으로부터 가요성 암(68)의 세트에 의해서 오프셋 된다. 캡핑 몰딩(64) 및 암(68)은 그 축에 대해서 봉(69)과 피벗팅(pivoting; 피벗식 결합)한다. 슬립 휠(70)은 봉(69)의 끝에 탑재된다. 슬립 휠(70)은 드라이브 휠(71)과 접촉한다. 인쇄가 일어나고 있을 때, 드라이브 휠(71)은 용지 이송 모터에 연결되고 언캡핑(uncapping) 방향(72)으로 구동된다. 이것은 슬립 휠(70) 및 봉(69)이 그 축에 관하여 회전하게 하고 캡핑 몰딩(64)을 프린트헤드로부터 멀리 스윙한다. 슬

립 휠이 언캐핑 슬립 포인트(73)까지 회전한 후, 슬립 휠 및 캐핑 몰딩은 회전을 중지한다. 인쇄가 완료되었을 때, 드라이브 휠은 역으로 되고 캐핑 방향(74)으로 구동된다. 슬립 휠이 캐핑 슬립 포인트(75)까지 회전한 후, 슬립 휠 및 캐핑 몰딩은 회전을 중지하고, 캐핑 스프링은 캐핑 플레이트를 프린트헤드 어셈블리의 면에 댄 자리에 유지한다. 가요성 암(68)은 캐핑 플레이트(67)가 프린트헤드 어셈블리(61)의 면에 합치하도록 돕는다.

4.2 프린터 제어기

프린터 제어기(80)가 도 8에 예시되고, 단지 몇 가지 구성요소 - 64Mbit RDRAM(82), 아이프린트 중앙 프로세서(ICP) 칩(83), 사용자에게 에러 조건을 통지하기 위한 스피커(84), QA 칩(85), 외부 3V DC 전력 접속부(86), 외부 USB 접속부(87), 용지 이송 스테퍼 모터로 접속부(88), 및 매체 센서(55), LED(7, 9 및 10), 버튼(6 및 8), 및 프린트헤드(63)로의 링크(90)에 접속되는 플렉스 PCB(89)를 갖는 작은 PCB(81)로 구성된다.

4.3 잉크 카트리지와 잉크 통로

잉크 카트리지의 2개의 버전이 존재한다 - 하나는 크고, 하나는 작다. 2개 모두 아이프린트 유닛의 뒷면에서 동일 잉크 카트리지를 슬롯에 맞는다.

5 프린터 제어 프로토콜

이 섹션은 호스트와 아이프린트 사이에 사용되는 프린터 제어 프로토콜을 설명한다. 이것은 실제의 페이지 기술뿐만 아니라 제어 및 상태 처리를 포함한다.

5.1 제어 및 상태

프린터에 대한 USB 디바이스 클래스 정의[15]는 단방향성 및 양방향성 IEEE 1284 병렬 포트의 에뮬레이션에 대해 제공한다[3]. 그 가장 기초 레벨에서, 이것은 호스트가 (GET_DEVICE_ID를 통해서) 프린터 능력(capability)을 결정하고, (GET_PORT_STATUS를 통해서) 프린터 상태를 획득하고, (SOFT_RESET을 통해서) 프린터를 리셋하게 한다. 센트로닉스/IEEE 1284 프린터 상태 필드가 이하 테이블 1에서 설명된다.

(테이블 1)

센트로닉스/IEEE 1284 프린터 상태	
필드	설명
선택	프린터가 선택되고 데이터 전송에 대해 이용 가능하다.
용지 없음	용지 없음 조건이 프린터에 존재한다.
결함	결함 조건이 프린터에 존재한다(용지 없음 및 선택하지 않음을 포함)

전형적으로 퍼스널 컴퓨터 인쇄 서브시스템은 IEEE 1284 지원의 특정 레벨을 제공한다. 따라서 프린터에서 IEEE 1284와의 호환성은 대응하는 프린터 드라이버의 개발을 단순화한다. 프린터에 대한 USB 디바이스 클래스 정의는 이러한 동일 호환성을 레버리징하려 한다.

아이프린트는 프린터에 대한 USB 디바이스 클래스 정의 이상의 어떠한 제어 프로토콜도 지원하지 않는다. 더 높은-레벨 제어 프로토콜이 정의된다면, 그 때 잉크 없음과 같은 조건이 또한 (프린터의 잉크 없음 LED를 통해서만 보다는) 사용자에게 보고될 수 있음을 주목하라.

아이프린트는 페이지 기술을 가공되지 않은 전송, 즉 임의의 더 높은-레벨 제어 프로토콜에서 캡슐에 넣어지지 않은 것으로서 수신된다.

5.2 페이지 기술

아이프린트는 완전 도트 해상도(1600 dpi)에서 블랙을 재현하지만, 하프토닝을 사용하여 다소 더 낮은 해상도에서 연속톤 컬러를 재현한다. 따라서 페이지 기술은 블랙 층과 연속톤 층으로 분할된다. 블랙 층은 연속톤 층 위로 합성하도록 정해진다.

블랙 층은 각각의 화소에 대해서 1-비트 불투명도를 포함하는 비트맵으로 구성된다. 이러한 블랙 층 매트(matte)는 프린터의 도트 해상도의 정수 인수인 해상도를 갖는다. 지원되는 가장 높은 해상도는 1600 dpi, 즉 프린터의 완전 도트 해상도이다.

연속톤 층은 각각의 화소에 대해서 32-비트 CMYK 컬러를 포함하는 비트맵으로 구성된다. 이러한 연속톤 이미지는 프린터의 도트 해상도의 정수 인수인 해상도를 갖는다. 지원되는 가장 높은 해상도는 267 ppi, 즉 프린터의 도트 해상도의 1/6이다.

연속톤 해상도는 프린터 드라이버에서의 계산을 단순화하기 위해서 또한 전형적으로 블랙 해상도의 정수 인수이다. 그러나, 이것은 필요조건은 아니다.

블랙 층 및 연속톤 층은 모두 저속 USB 접속부를 통한 프린터로의 효과적인 송신을 위해서 압축된 형태이다.

5.2.1 페이지 구조

아이프린트는 그 프린트헤드의 폭, 그 용지 통로의 특성, 및 현재 선택된 인쇄 매체의 크기에 의해서 결정되는 인쇄 가능한 페이지 영역을 갖는다.

인쇄 가능한 페이지 영역은 8"의 최대 폭을 갖는다. 물리적 페이지 폭이 8"를 초과하면, 그 때 대칭적인 좌측 및 우측 여백이 암시적으로 생성된다. 물리적 페이지 폭이 8"보다 작다면, 그 때 인쇄 가능한 페이지 폭은 그에 따라서 감소된다. 인쇄 가능한 페이지 영역은 최대 길이를 갖지 않는다. 그것은 용지 통로의 특성에 의해 부과된 상부 및 하부 여백을 뺀 물리적 페이지 길이일 뿐이다.

타겟 페이지 크기는 페이지 기술에서 지정된 명시적인 (타겟) 좌측 및 상부 여백을 뺀, 인쇄 가능한 페이지 영역에 의해서 제약을 받는다.

이론적으로 아이프린트는 상부 또는 하부 여백을 부과하지 않는다 - 즉, 그것은 수직 방향으로 완전한 블리드를 허용한다. 그러나, 실제로는 아이프린트가 8" 프린트헤드를 사용하기 때문에 완전-블리드 A4/레터 프린터로서 디자인되지 않으므로, 규칙적으로 에지를 벗어나는 인쇄에 대처하기에 충분히 큰 스폰지를 포함해야 하는 것을 피하기 위해서 의사 상부 및 하부 여백이 부과된다.

5.2.2 페이지 기술 포맷

테이블 2는 아이프린트에 의해서 기대되는 페이지 기술의 포맷을 나타낸다.

(테이블 2)

페이지 기술 포맷		
필드	포맷	설명
서명	16-비트 정수	페이지 기술 포맷 서명
버전	16-비트 정수	페이지 기술 포맷 버전 넘버
구조 크기	16-비트 정수	페이지 기술의 고정-크기 파트의 크기
타겟 해상도 (dpi)	16-비트 정수	타겟 페이지의 해상도. 이것은 아이프린트에 대해서 항상 1600 이다.
타겟 페이지 폭	16-비트 정수	도트에서, 타겟 페이지의 폭
타겟 페이지 높이	16-비트 정수	도트에서, 타겟 페이지의 높이
타겟 좌측 여백	16-비트 정수	도트에서, 타겟 좌측 여백의 폭
타겟 상부 여백	16-비트 정수	도트에서, 타겟 상부 여백의 높이
블랙 환산 계수	16-비트 정수	블랙 해상도로부터 타겟 해상도로의 환산 계수(2 또는 더 커야한다)
블랙 페이지 폭	16-비트 정수	블랙 화소에서, 블랙 페이지의 폭
블랙 페이지 높이	16-비트 정수	블랙 화소에서, 블랙 페이지의 높이
블랙 페이지 데이터 크기	32-비트 정수	바이트에서, 블랙 페이지 데이터의 크기
연속톤 환산 계수	16-비트 정수	연속톤 해상도로부터 타겟 해상도로의

		환산 계수(6 또는 더 커야한다)
연속톤 페이지 폭	16-비트 정수	연속톤 화소에서, 연속톤 페이지의 폭
연속톤 페이지 높이	16-비트 정수	연속톤 화소에서, 연속톤 페이지의 높이
연속톤 페이지 데이터 크기	32-비트 정수	바이트에서, 연속톤 페이지 데이터의 크기
블랙 페이지 데이터	EDRL 바이트스트림	압축된 2-레벨 블랙 페이지 데이터
연속톤 페이지 데이터	JPEG 바이트스트림	압축된 연속톤 CMYK 페이지 데이터

인쇄 가능한 페이지 영역에 관하여 암시적으로 정해지는 것과는 별개로, 각각의 페이지 기술은 완벽하고 다 갖추어져 있다. 페이지 기술이 설명하는 페이지 기술과 분리되어 프린터에 송신되는 데이터는 없다.

페이지 기술은 프린터가 페이지 기술 포맷을 식별하게 하는 서명 및 버전을 포함한다. 서명 및/또는 버전이 없거나 또는 프린터와 모순된다면, 그 때 프린터는 페이지를 거절할 수 있다.

페이지 기술은 타겟 페이지의 크기 및 해상도를 정한다. 블랙 및 연속톤 층은 필요하다면 타겟 페이지에 클리핑된다. 이것은 블랙 및 연속톤 환산 계수가 타겟 페이지 폭 또는 높이의 인수가 아닐 때마다 일어난다.

타겟 좌측 및 상부 여백은 인쇄 가능한 페이지 영역내에서 타겟 페이지의 위치결정을 정한다.

블랙 층 매개 변수는 블랙 층의 화소 크기, 타겟 해상도로의 그 정수 환산 계수, 및 그 압축된 페이지 데이터의 크기를 정한다. 가변-크기 블랙 페이지 데이터는 페이지 기술의 고정-크기 파트를 따른다.

연속톤 층 매개 변수는 연속 톤 층의 화소 크기, 타겟 해상도로의 그 정수 환산 계수, 및 그 압축된 페이지 데이터의 크기를 정한다. 가변-크기 연속톤 페이지 데이터는 가변-크기 블랙 페이지 데이터를 따른다.

페이지 기술에서의 모든 정수는 빅-엔디언(big-endian) 바이트 순서로 저장된다.

가변-크기 블랙 페이지 데이터 및 가변-크기 연속톤 페이지 데이터는 8-바이트 경계로 정렬된다. 요구되는 패딩의 크기는 가변-크기 블랙 데이터 및 페이지 기술 구조의 고정-크기 파트의 크기에 포함된다.

전체 페이지 기술은 프린터에서의 페이지 버퍼 메모리에 따라서, 3MB보다 작은 타겟 크기 및 6MB의 최대 크기를 갖는다.

이하 섹션은 압축된 블랙 층 및 압축된 연속톤 층의 포맷을 설명한다.

5.2.3 2-레벨 블랙 층 압축

5.2.3.1 그룹 3 및 4 픽시밀리 압축

그룹 3 픽시밀리 압축 알고리즘[1]은 느리고 잡음 섞인 전화 라인을 통한 전송을 위해서 2-레벨 데이터를 손실없이 압축한다. 2-레벨 데이터는 화이트 배경상에 스캐닝된 블랙 텍스트 및 그래픽스를 표현하고, 알고리즘은 이미지의 이러한 클래스에 대해서 튜닝된다(그것은, 예를 들어, 하프토닝된 2-레벨 이미지에 대해서 명시적으로 튜닝되지는 않는다). 1D 그룹 3 알고리즘은 각각의 스캔라인을 런길이-인코딩하고 그 후 결과 런길이를 허프만-인코딩한다. 범위 0에서 63에서의 런길이는 종료(terminating) 코드와 코딩된다. 범위 64에서 2623에서의 런길이는 종료 코드를 수반하는 구성 코드와 코딩되고, 각각은 64의 배수를 표현한다. 2623을 초과하는 런길이는 종료 코드를 수반하는 다중 구성 코드와 코딩된다. 허프만 테이블은 고정되지만, 블랙 및 화이트 런에 대해서 분리되어 튜닝된다(공통인, 1728 이상의 구성 코드는 제외). 가능할 때, 2D 그룹 3 알고리즘은 이전 스캔라인을 참조하여 스캔라인을 짧은 에지 델타의 세트(0, ± 1 , ± 2 , ± 3)로서 인코딩한다. 델타 심볼은 (제로 델타 심볼이 하나의 비트 길이만을 가지기 위해서 등) 엔트로피-인코딩된다. 델타-인코딩될 수 없는 2D-인코딩된 라인내의 에지는 런길이-인코딩되고, 프리픽스에 의해서 식별된다. 1D- 및 2D-인코딩된 라인은 다르게 마킹된다. 1D-인코딩된 라인은 디코더가 최소 이미지 저하로 라인 잡음으로부터 회복할 수 있는 것을 보장하기 위해서, 실제적으로 요구되던 아닌딘간에, 규칙적 인터발에서 발생된다. 2D 그룹 3은 6:1까지의 압축비를 성취한다[13].

그룹 4 픽시밀리 알고리즘[1]은 에러-없는 통신 라인을 통한 전송을 위해서 2-레벨 데이터를 손실없이 압축한다(즉, 라인은 정말로 에러가 없거나, 또는 더 낮은 프로토콜 레벨에서 에러-보정이 이루어진다). 그룹 4 알고리즘은 송신이 에러가

없다고 가정되기 때문에 1D-인코딩된 라인이 에러-회복에 도움으로서 규칙적 인터발에서 더 이상 발생되지 않게 본질적으로 변형되어 2D 그룹 3 알고리즘에 기초한다. 그룹 4는 테스트 이미지의 CCITT 세트에 대하여 20:1 내지 60:1 범위의 압축비를 달성한다[13].

그룹 4 압축 알고리즘의 성능 및 디자인 목표는 그것에 2-레벨 블랙 층에 대한 압축 알고리즘으로서 자격을 부여하는 것이다. 그러나 그 허프만 테이블은 더 낮은 스캐닝 해상도(100-400 dpi)로 튜닝되고, 그것은 2623을 초과하는 런길이를 서툴게 인코딩한다. 800 dpi에서, 최대 런길이는 현재 6400이다. 그룹 4 디코더 코어는 프린터 제어기 칩에서의 사용을 위해 이용 가능하기는 하지만(섹션 7), 그것은 400 dpi 팩시밀리 애플리케이션에서 통상 부딪히게 되는 것들을 초과하는 런길이를 처리할 수 없고, 그래서 변형을 요구할 것이다.

그룹 4의 이득의 대부분은 델타-인코딩으로부터 오기 때문에, 델타-인코딩만에 기초한 더 단순한 알고리즘은 우리의 요구를 충족시킬 것도 같다. 이러한 접근은 이하 상세히 설명된다.

5.2.3.2 2-레벨 에지 델타 및 런길이(EDRL) 압축 포맷

에지 델타 및 런길이(EDRL) 압축 포맷은 막연히 그룹 4 압축 포맷 및 그 선구자에 기초한다[1][17].

EDRL은 적절히 엔트로피-코딩된 3가지 종류의 심볼을 사용한다. 이들은 생성 에지, 소거 에지, 및 에지 델타이다. 각각의 라인은 그 선행자를 참조하여 코딩된다. 제 1 라인의 선행자는 화이트 라인으로 정해진다. 각각의 라인은 화이트로부터 시작하도록 정해진다. 라인이 실제로 블랙으로 시작한다면(덜 있을 법한 상황), 그 때 그것은 오프셋 제로에서 블랙 에지를 정해야 한다. 각각의 라인은 그 좌측 끝, 즉 오프셋 페이지 폭에서 에지를 정해야 한다.

동일 센스(화이트-블랙 또는 블랙-화이트)를 갖는 최대 델타 범위내의 에지가 존재하지 않는다면, 에지는 이전 라인에서의 에지를 참조하여 코딩될 수 있다. 이것은 에지 델타 코드 중 하나를 사용한다. 더 짧고 더 있을법한 델타는 더 짧은 코드를 갖는다. 최대 델타 범위(± 2)는 전형적 글리프(glyph) 에지에 대하여 델타의 분포를 매칭시키기 위해 선택된다. 이러한 분포는 대부분 포인트 크기로부터 독립적이다. 전형적인 예가 테이블 3에 주어진다.

(테이블 3)

800 dpi에서 10 포인트 타임즈에 대한 에지 델타 분포	
델타	확률
0	65%
1	23%
2	7%
≥ 3	5%

에지는 또한 동일 라인에서의 이전 에지로부터 런의 길이를 사용하여 코딩될 수 있다. 이것은 짧은(7-비트) 및 긴(13-비트) 런길이에 대해서 생성 에지 코드 중 하나를 사용한다. 간소화를 위해서, 그룹 4와 다르게, 런길이는 엔트로피-코딩되지 않는다. 에지 델타를 이전 라인에서의 에지와 암시적으로 동기화되게 유지하게 위해서, 이전 라인에서의 각각의 사용되지 않은 에지는 현재 라인에서 통과될 때 "소거"된다. 이것은 소거 에지 코드를 사용한다. 페이지 끝 코드는 페이지의 끝을 디코더에 신호 보낸다.

7-비트 및 13-비트 런길이는 800 dpi A4/레터 페이지를 지원하기 위해서 특별히 선택된다. 더 긴 런길이는 압축 성능상에 상당한 충격없이 지원될 수 있다. 예를 들어, 1600 dpi 압축을 지원한다면, 런길이는 각각 적어도 8-비트 및 14-비트이어야 한다. 범용-목적의 선택은 8-비트 및 16-비트일 수 있고, 따라서 40" 와이드 1600 dpi 페이지까지 지원한다.

코드의 완전한 세트가 테이블 4에 정의된다. 라인 끝 코드가 존재하지 않음을 주목하라. 디코더는 라인의 끝을 검출하기 위해서 페이지 폭을 사용한다. 코드의 길이는 코드 발생의 상대 확률에 의해서 오더링된다.

(테이블 4)

EDRL 코드워드			
코드	인코딩	서픽스	설명
$\Delta 0$	1	-	다음 에지를 이동시키지 않음

$\Delta+1$	010	-	대응 에지를 +1 이동시킴
$\Delta-1$	011	-	대응 에지를 -1 이동시킴
$\Delta+2$	00010	-	대응 에지를 +2 이동시킴
$\Delta-2$	00011	-	대응 에지를 -2 이동시킴
소거 에지	0010	-	대응 에지를 소거
가까운 에지 생성	0011	7-비트 RL	짧은 런길이(RL)로부터 에지를 생성
먼 에지 생성	00001	13-비트 RL	긴 런길이(RL)로부터 에지를 생성
페이지 끝(EOP)	000001	-	페이지 끝 마커

도 9는 단순한 블랙 및 화이트 이미지(90)를 코딩하는 예를 도시한다. 이미지는 화소(92)의 라인(91)으로서 배치된다. 제 1 라인(91)은 화이트라 가정되고, 그렇기 때문에, $\Delta 0$ 으로서 코딩된다. 또 다른 전부-화이트 라인에 수반되는 전부-화이트 라인의 공통 상황은 단일 비트($\Delta 0$)를 사용하여 코딩되고, 또 다른 전부-블랙 라인에 수반되는 전부-블랙 라인은 2개의 비트($\Delta 0, \Delta 0$)를 사용하여 코딩됨을 주목하라. 제 4 라인(93)과 같이, 라인에서 에지가 일어나는 경우에, 에지를 정하기 위해서 생성 코드가 사용된다. 다음 라인(94)에서, $\Delta-1$ 및 $\Delta+1$ 코드가 에지를 이동시키기 위해서 사용된다. 다음 라인(95)에서, 에지를 이동시키는 거보다 새로운 에지를 생성하고 오래된 에지를 소거하는 것이 더 편리하다.

EDRL 인코딩 예

상기의 것은 압축 알고리즘 그 자체(per se)를 설명하는 것이 아니라, 압축 포맷을 설명하는 것임을 주목하라. 다양한 등가 인코딩이 다른 것들보다 다소 더 소형인, 동일 이미지에 대해서 산출될 수 있다. 예를 들어, 순수한 런길이 인코딩은 압축 포맷에 합치한다. 압축 알고리즘의 목표는 일정 이미지에 대해서, 최상은 아니더라도, 훌륭한 인코딩을 발견하는 것이다.

이하의 것은 선행자를 참조하여 라인의 EDRL 인코딩을 산출하기 위한 단순한 알고리즘이다.

```
#define SHORT_RUN_PRECISION7 // 짧은 런의 정밀도
```

```
#define LONG_RUN_PRECISION13 // 긴 런의 정밀도
```

```
EDRL_CompressLine
```

```
(
```

```
Byte prevLine[], // 이전 (기준) 2-레벨 라인
```

```
Byte currLine[], // 현재 (코딩) 2-레벨 라인
```

```
int lineLen, // 라인 길이
```

```
BITSTREAMs // 출력 (압축된) 비트스트림
```

```
)
```

```
int prevEdge = 0 // 이전 라인에서 현재 에지 오프셋
```

```
int currEdge = 0 // 현재 라인에서 현재 에지 오프셋
```

```
int codedEdge = currEdge // 가장 최근 코딩된 (출력) 에지
```

```
int prevColor = 0 // 이전 라인에서 현재 컬러 (0 = 화이트)
```

```
int currColor = 0 // 현재 라인에서 현재 컬러
```

```
int prevRun // 이전 라인에서 현재 런
```

```

int currRun // 현재 라인에서 현재 런

bool bUpdatePrevEdge = true // 처음 에지 업데이트 강제

bool bUpdateCurrEdge = true // 처음 에지 업데이트 강제

while (codedEdge < lineLen)

// 이전 라인에서 현재 에지를 업데이트

if (bUpdatePrevEdge)

if (prevEdge < lineLen)

prevRun = GetRun(prevLine, prevEdge, lineLen, prevColor)

else

prevRun = 0

prevEdge += prevRun

prevColor = !prevColor

bUpdatePrevEdge = false

// 현재 라인에서 현재 에지를 업데이트

if (bUpdateCurrEdge)

if (currEdge < lineLen)

currRun = GetRun(currLine, currEdge, lineLen, currColor)

else

currRun = 0

currEdge += currRun

currColor = !currColor

bUpdateCurrEdge = false

// 가능할 때마다 델타를 출력, 즉

// 에지가 매치를 감지하고, 델타가 충분히 작을 때

if (prevColor == currColor)

delta = currEdge - prevEdge

if (abs(delta) <= MAX_DELTA)

```

```
PutCode(s, EDGE_DELTA0 + delta)
```

```
codedEdge = currEdge
```

```
bUpdatePrevEdge = true
```

```
bUpdateCurrEdge = true
```

```
continue
```

```
// 이전 라인에서 매칭되지 않은 에지를 소거
```

```
if (prevEdge <= currEdge)
```

```
PutCode(s, KILL_EDGE)
```

```
bUpdatePrevEdge = true
```

```
// 현재 라인에서 매칭되지 않은 에지를 생성
```

```
if (currEdge <= prevEdge)
```

```
PutCode(s, CREATE_EDGE)
```

```
if (currRun < 128)
```

```
PutCode(s, CREATE_NEAR_EDGE)
```

```
PutBits(currRun, SHORT_RUN_PRECISION)
```

```
else
```

```
PutCode(s, CREATE_FAR_EDGE)
```

```
PutBits(currRun, LONG_RUN_PRECISION)
```

```
codedEdge = currEdge
```

```
bUpdateCurrEdge = true
```

알고리즘은 라인 사이의 실제의 에지 연속성을 분간 못하고, 실제로 2개의 라인 사이에 "틀린" 에지와 매칭할 수 있음을 주목하라. 다행히도 압축 포맷은 옳게 디코딩하고, "틀린" 매치가 압축비에 불리한 효과를 주는 것은 어렵기 때문에 압축 포맷은 이것에 대해서 말할 것이 없다.

완결을 위해 대응 압축해제 알고리즘이 이하 주어진다. 그것은 프린터 제어기 칩에서 EDRL 확장기 유닛의 코어를 형성한다(섹션 7).

```
EDRL-DecompressLine
```

```
(
```

```
BITSTREAM s, // (압축된) 비트스트림을 입력
```

```

Byte prevLine[], // 이전 (기준) 2-레벨 라인

Byte currLine[], // 현재 (코딩된) 2-레벨 라인

int lineLen // 라인 길이

)

int prevEdge = 0 // 이전 라인에서 현재 에지 오프셋

int currEdge = 0 // 현재 라인에서 현재 에지 오프셋

int prevColor = 0 //이전 라인에서 현재 컬러(0=화이트)

int currColor = 0 // 현재 라인에서 현재 컬러

while (currEdge < lineLen)

code = GetCode(s)

switch (code)

case EDGE_DELTA_MINUS2:

case EDGE_DELTA_MINUS1:

case EDGE_DELTA_0:

case EDGE_DELTA_PLUS1:

case EDGE_DELTA_PLUS2:

// 델타로부터 에지를 생성

int delta = code - EDGE_DELTA_0

int run = prevEdge + delta - currEdge

FillBitRun(currLine, currEdge, currColor, run)

currEdge += run

currColor = !currColor

prevEdge+=GetRun(prevLine,prevEdge,lineLen,prevColor)

prevColor = !prevColor

case KILL_EDGE:

// 사용되지 않는 기준 에지를 버린다

prevEdge+=GetRun(prevLine,prevEdge,lineLen,prevColor)

```

```
prevColor = !prevColor

case CREATE_NEAR_EDGE:

case CREATE_FAR_EDGE:

// 에지를 명시적으로 생성

int run

if (code == CREATE_NEAR_EDGE)

run = GetBits(s, SHORT_RUN_PRECISION)

else

run = GetBits(s, LONG_RUN_PRECISION)

FillBitRun(currLine, currEdge, currColor, run)

currColor = !currColor

currEdge += run
```

5.2.3.3 EDRL 압축 성능

테이블 5는 그룹 4 알고리즘을 선택하기 위해서 사용되는 CCITT 테스트 문서상의 EDRL 및 그룹 4의 압축 성능을 나타낸다. 각각의 문서는 400 dpi에서 스캐닝된 단일 페이지를 표현한다. 그룹 4의 우수한 성능은 400 dpi 특징으로 튜닝된, 그 엔트로피-코딩된 런길이에 기인한다.

(테이블 5)

400 dpi에서 표준 CCITT 문서상의 그룹 4 및 EDRL 압축 성능		
CCITT 문서 넘버	그룹 4 압축비	EDRL 압축비
1	29.1	21.6
2	49.9	41.3
3	17.9	14.1
4	7.3	5.5
5	15.8	12.4
6	31.0	25.5
7	7.4	5.3
8	26.7	23.4

전형적으로 잡지 텍스트는 10의 포인트 크기에서 (타임즈(Times)와 같은) 세리프를 갖는 서체에서 조판된다. 이러한 크기에서, 전형적 잡지 페이지는 약 7,000 캐릭터만을 유지하기는 하지만, A4/레터 페이지는 14,000 캐릭터까지 유지한다. 5보다 더 작은 포인트 크기에서 텍스트는 거의 조판되지 않는다. 800 dpi에서, 텍스트는 표준 서체를 사용하여 2보다 더 낮은 포인트 크기에서 의미있게 렌더링될 수 없다. 테이블 6은 다양한 포인트 크기의 읽기 쉬움을 예시한다.

(테이블 6)

다른 포인트 크기에서 텍스트	
포인트 크기	샘플 텍스트(타임즈에서)
8	The quick brown fox jumps over the lazy dog.

9	The quick brown fox jumps over the lazy dog.
10	The quick brown fox jumps over the lazy dog.

테이블 7은 800 dpi에서 렌더링된, 변하는 포인트 크기의 텍스트의 페이지상의 그룹 4 및 EDRL 압축 성능을 나타낸다. EDRL은 3의 포인트 크기에서 조판된 텍스트의 전체 페이지에 대하여 2.5의 요구되는 압축비를 달성함을 주목하라. 테스트 페이지상의 캐릭터의 분포는 영국-언어 통계에 기초된다[12].

(테이블 7)

800 dpi에서 텍스트상의 그룹 4 및 EDRL 압축 성능			
포인트 크기	캐릭터/A4 페이지	그룹 4 압축비	EDRL 압축비
2	340,000	2.3	1.7
3	170,000	3.2	2.5
4	86,000	4.7	3.8
5	59,000	5.5	4.9
6	41,000	6.5	6.1
7	28,000	7.7	7.4
8	21,000	9.1	9.0
9	17,000	10.2	10.4
10	14,000	10.9	11.3
11	12,000	11.5	12.4
12	8,900	13.5	14.8
13	8,200	13.5	15.0
14	7,000	14.6	16.6
15	5,800	16.1	18.5
20	3,400	19.8	23.9

9 또는 더 큰 포인트 크기에 대해서, 그룹 4의 런길이 코드는 400 dpi로 튜닝되기 때문만으로 EDRL은 그룹 4보다 약간 성능이 낫다.

이러한 압축 결과는 매우 작은 캐릭터의 경우에서와 같이 데이터가 수직으로 불완전하게 상관되지 않는다면 엔트로피-인코딩된 런길이가 2D 인코딩보다 압축에 덜 기여한다는 경험을 확증한다.

5.2.4 연속톤 총 압축

5.2.4.1 JPEG 압축

JPEG 압축 알고리즘[6]은 지정된 품질 레벨에서 연속톤 이미지를 손실있게 압축한다. 그것은 5:1 이하의 압축비에서 미세한 이미지 저하 및 10:1 이하의 압축비에서 무시할 수 있는 이미지 저하를 도입한다[16].

전형적으로 처음에 JPEG는 휘도 및 색차를 분리된 컬러 채널로 분리하는 컬러 공간으로 이미지를 변환한다. 이것은 색차보다 휘도에 대한 인간 가시 시스템의 비교적 더 큰 감도 때문에 색차 채널이 감지할 수 있는 손실없이 서브샘플링되게 한다. 이 첫번째 단계 후에, 각각의 컬러 채널은 분리되어 압축된다.

이미지는 8×8 화소 블록으로 분할된다. 각각의 블록은 그 후 불연속 코사인 변환(DCT)을 통해서 주파수 도메인으로 변환된다. 이러한 변환은 더 높은-주파수 계수가 더 있는 그대로 양자화되게 하는 상대적으로 더 낮은-주파수 계수에서 이미지의 에너지를 집중시키는 효과를 갖는다. 이러한 양자화는 JPEG에서 압축의 주요 소스이다. 더 나아간 압축은 인접 제로 계수의 가능성을 최대화하기 위한 주파수에 의해서 계수를 오더링하고, 그 후 제로의 런을 런길이-인코딩함으로써 달성된다. 최종적으로, 런길이 및 제로-아닌 주파수 계수는 엔트로피 코딩된다. 압축해제는 압축의 역 프로세스이다.

5.2.4.2 CMYK 연속톤 JPEG 압축 포맷

CMYK 연속톤 층은 인터리빙 컬러 JPEG 바이트스트림으로 압축된다. 프린터에서 공간-효율적인 압축해제를 위해서 인터리빙이 요구되지만, 디코더를 4개보다는 2개의 세트의 허프만 테이블로 한정할 수 있다(즉, 컬러 채널 당 하나)[16]. 휘도와 색차가 분리된다면, 그 때 휘도 채널은 테이블의 하나의 세트를 공유하고 색차 채널은 다른 세트를 공유할 수 있다.

휘도/색차 분리가 필요하다고 생각되면, 색차 서브샘플링 또는 테이블 공유의 목적을 위해서, 그 때 CMY가 YCrCb로 변환되고 Cr 및 Cb가 때에 알맞게 서브샘플링된다. K는 휘도 채널로서 다루어지고 서브샘플링되지 않는다.

JPEG 바이트스트림은 완벽하고 다 갖추어져 있다. 그것은 양자화 및 허프만 테이블을 포함하는, 압축해제를 위해 요구되는 모든 데이터를 포함한다.

6 멤젯 프린트헤드

8-인치 멤젯 프린트헤드는 나란히 함께 결합된 2개의 표준 4-인치 멤젯 프린트헤드로 구성된다.

2개의 4-인치 프린트헤드는 아이프린트에서 사용하기 위한 특정 방식으로 함께 배선된다. 배선은 4-인치 프린트헤드의 지식을 요구하기 때문에, 4-인치 프린트헤드의 개략이 여기서 제공된다.

6.1 4-인치 프린트헤드의 합성

각각의 4-인치 프린트헤드는 8 세그먼트로 구성되고, 각각의 세그먼트는 길이가 1/2 인치이다. 세그먼트의 각각은 최종 이미지를 산출하기 위해서 2-레벨 시안, 마젠타, 옐로우, 및 블랙 도트를 페이지의 다른 파트 위로 인쇄한다.

프린트헤드는 1600 dpi에서 도트를 인쇄하기 때문에, 각각의 도트는 직경이 대략 22.5 미크론이고, 15.875 미트론 떨어져 일정 간격으로 이격된다. 따라서, 각각의 반-인치 세그먼트는 800 도트를 인쇄하고, 8 세그먼트는 테이블 8에 나타내어진 위치에 대응한다.

(테이블 8)

각각의 세그먼트에 의해서 번지 지정된 최종 이미지 도트				
세그먼트	프린트헤드 1		프린트헤드 2	
	처음 도트	마지막 도트	처음 도트	마지막 도트
0	0	799	6,400	7,199
1	800	1,599	7,200	7,999
2	1,600	2,399	8,000	8,799
3	2,400	3,199	8,800	9,599
4	3,200	3,999	9,600	10,399
5	4,000	4,799	10,400	11,199
6	4,800	5,599	11,200	11,999
7	5,600	6,399	12,000	12,799

각각의 세그먼트는 최종 이미지의 800 도트를 산출하기는 하지만, 각각의 도트는 2-레벨 시안, 마젠타, 옐로우 및 블랙 잉크의 조합에 의해서 표현된다. 인쇄는 2-레벨이기 때문에, 입력 이미지는 최상의 결과를 위해서 에러-확산되거나 디터링되어야 한다.

그 때 각각의 세그먼트는 3,200개의 노즐을 포함한다: 시안, 마젠타, 옐로우 및 블랙의 각각이 800개씩이다. 4-인치 프린트헤드는 총 25,600개의 노즐에 대해 8개의 그러한 세그먼트를 포함한다.

6.1.1 세그먼트내의 노즐의 그룹화

단일 세그먼트내의 노즐은 인쇄 동안의 전력 소비의 최소화뿐만 아니라 물리적 안정성의 이유로 그룹화된다. 물리적 안정성의 견지에서, 총 10개의 노즐이 동일 잉크통을 공유한다. 전력 소비의 견지에서, 저속 및 고속 인쇄 모드를 이네이블하도록 그룹화가 이루어진다.

프린트헤드는 속도/전력 소비가 다른 제품 구성에서 교환되게 하기 위해서 2개의 인쇄 속도를 지원한다.

저속 인쇄 모드에서, 128개의 노즐이 각각의 4-인치 프린트헤드로부터 동시에 파이어링된다. 파이어링된 노즐은 최대로 멀어야하고, 그래서 16개의 노즐이 각각의 세그먼트로부터 파이어링된다. 모든 25,600개의 노즐을 파이어링하기 위해서, 128개의 노즐의 200개의 다른 세트가 파이어링되어야 한다.

고속 인쇄 모드에서, 256개의 노즐이 각각의 4-인치 프린트헤드로부터 동시에 파이어링된다. 파이어링된 노즐은 최대로 멀어야 하고, 그래서 32개의 노즐이 각각의 세그먼트로부터 파이어링된다. 모든 25,600개의 노즐을 파이어링하기 위해서, 256개의 노즐의 100개의 다른 세트가 파이어링되어야 한다.

저속 모드에서의 전력 소비는 고속 모드에서의 전력소비의 반이다. 그러나, 페이지를 인쇄하기 위해 소비되는 에너지는 둘 다의 경우에서 동일함을 주목하라.

6.1.1.1 10개의 노즐이 파드를 만든다

단일 파드(100)는 공통 잉크통을 공유하는 10개의 노즐(101)로 구성된다. 5개의 노즐이 하나의 행에 있고, 5개는 또 다른 행에 있다. 각각의 노즐은 15.875 미크론 격자상에 일정 간격으로 이격된 직경 22.5 미크론 도트를 산출한다. 도 10은 단일 파드(100)의 배치를 도시하고, 노즐(101)은 그들이 파이어링되어야 하는 순서에 따라 번호가 매겨져 있다.

노즐은 이 순서대로 파이어링되기는 하지만, 노즐의 관계와 인쇄된 페이지상의 도트의 물리적 배치는 다르다. 하나의 행으로부터의 노즐은 페이지상의 하나의 라인으로부터의 짝수 도트를 표현하고, 다른 행상의 노즐은 페이지상의 인접 라인으로부터의 홀수 도트를 표현한다. 도 11은 노즐이 로딩되어야 하는 순서에 따라 번호 매겨진 노즐을 갖는 동일 파드(100)를 도시한다.

따라서 파드내의 노즐은 1 도트의 폭에 의해서 논리적으로 분리된다. 노즐 사이의 정확한 거리는 맴젯 파이어링 메카니즘의 특성에 의존할 것이다. 프린트헤드는 용지의 흐름을 매칭하기 위해서 디자인된 스테거된 노즐로써 디자인된다.

6.1.1.2 각각의 컬러의 하나의 파드는 크로마파드를 만든다

각각의 컬러, 즉 시안(121), 마젠타(122), 옐로우(123) 및 블랙(124)의 하나의 파드는 크로마파드(125)로 그룹화된다. 크로마파드는 다른 라인상에서 10 도트의 동일 수평 세트의 다른 컬러 구성요소를 표현한다. 다른 컬러 파드 사이의 정확한 거리는 맴젯 동작 매개 변수에 의존하고, 하나의 맴젯 디자인으로부터 또 다른 것으로 변할 수 있다. 거리는 도트-폭의 일정 수인 것으로 고려되고 따라서 인쇄할 때 고려되어야 한다: 시안 노즐에 의해서 인쇄된 도트는 마젠타, 옐로우 또는 블랙 노즐에 의해서 인쇄된 것보다 다른 라인에 대한 것일 것이다. 인쇄 알고리즘은 컬러 사이에 약 8 도트-폭까지의 가변 거리에 대해서 허용하여야 한다. 도 12는 단일 크로마파드를 예시한다.

6.1.1.3 5개의 크로마파드가 파드그룹을 만든다

5개의 크로마파드(125)는 단일 파드그룹(126)으로 편성된다. 각각의 크로마파드는 40개의 노즐을 포함하기 때문에, 각각의 파드그룹은 200개의 노즐(50개의 시안, 50개의 마젠타, 50개의 옐로우, 및 50개의 블랙 노즐)을 포함한다. 배열이 도 13에 도시되고, 크로마파드는 0-4로 번호 매겨진다. 인접 크로마파드 사이의 거리는 명료함을 위해서 과장됨을 주목하라.

6.1.1.4 2개의 파드그룹은 페이지그룹을 만든다

2개의 파드그룹(126)은 단일 페이지그룹(127)으로 편성된다. 페이지그룹은 일정 파이어링 페이지 동안에 페이지그룹내의 노즐의 그룹이 동시에 파이어링되기 때문에 그렇게 명명된다(이것은 이하 더 상세히 설명된다). 2개의 파드그룹으로부터 페이지그룹의 형성은 전적으로 2개의 파드그룹이네이블 라인을 통하여 저속 및 고속 인쇄의 목적을 위해서이다.

저속 인쇄 동안, 2개의 파드그룹이네이블 라인 중 하나만이 일정 파이어링 펄스에서 세팅되고, 그래서 2개 중 하나의 파드그룹만이 노즐을 파이어링한다. 고속 인쇄 동안에, 2개의 파드그룹이네이블 라인이 모두 세팅되고, 그래서 2개의 파드그룹이 노즐을 파이어링한다. 결과적으로 고속 인쇄는 한번에 두배 많은 노즐을 파이어링하기 때문에, 저속 인쇄는 고속 인쇄보다 두배 많은 시간이 걸린다.

도 14는 페이지그룹의 합성을 예시한다. 인접 파드그룹 사이의 거리는 명료함을 위해서 과장된다.

6.1.1.5 2개의 페이지그룹은 파이어그룹을 만든다

2개의 페이지그룹(127; 페이지그룹A 및 페이지그룹B)은 단일 파이어그룹(128)으로 편성되고, 4개의 파이어그룹은 각각의 세그먼트(129)내에 있다. 파이어그룹은 그것이 모두 동시에 동일 노즐을 파이어링하기 때문에 그렇게 명명된다. 2개의 이네이블 라인, A이네이블 및 B이네이블은 페이지그룹A 노즐 및 페이지그룹B 노즐의 파이어링을 다른 파이어링 페이지로서 독립적으로 허용한다. 배열이 도 15에 도시된다. 인접 그룹화 사이의 거리는 명료함을 위해서 과장된다.

6.1.1.6 노즐 그룹화 요약

테이블 9는 프린트헤드에서 노즐 그룹화의 요약이다.

(테이블 9)

단일 4-인치 프린트헤드에 대한 노즐 그룹화			
그룹화의 이름	합성	복제비	노즐 카운트
노즐	베이스 유닛	1:1	1
파드	파드 당 노즐	10:1	10
크로마파드	CMYK 크로마파드 당 파드	4:1	40
파드그룹	파드그룹 당 크로마파드	5:1	200
페이지그룹	페이지그룹 당 파드그룹	2:1	400
파이어그룹	파이어그룹 당 페이지그룹	2:1	800
세그먼트	세그먼트 당 파이어그룹	4:1	3,200
4-인치 프린트헤드	4-인치 프린트헤드 당 세그먼트	8:1	25,600

8-인치 프린트헤드는 총 51,200개의 노즐에 대하여 2개의 4-인치 프린트헤드로 구성된다.

6.1.2 로드 및 인쇄 사이클

단일 4-인치 프린트헤드는 총 25,600개의 노즐을 포함한다. 인쇄 사이클은 인쇄될 정보에 의존한, 이들 노즐의 전부까지의 파이어링을 포함한다. 로드 사이클은 후속 인쇄 사이클 동안에 인쇄될 정보로써 프린트헤드의 로딩 업을 포함한다.

각각의 노즐은 인쇄 사이클 동안 노즐이 파이어링할 것인지 아닌지를 결정하는 연관 노즐이네이블 비트를 갖는다. 노즐이네이블 비트(노즐 당 하나)는 시프트 레지스터의 세트를 통해서 로딩된다.

논리적으로 세그먼트 당 4개의 시프트 레지스터(컬러 당 하나)가 존재하고, 각각은 800 디프(deep)이다. 일정 컬러에 대한 시프트 레지스터내로 비트가 시프팅할 때 그들은 교번 펄스상에서 하부 및 상부 노즐로 향해진다. 내부적으로, 각각의 800-디프 시프트 레지스터는 2개의 400-디프 시프트 레지스터(상부 노즐에 대하여 하나, 및 하부 노즐에 대하여 하나)로 구성된다. 교번 비트는 교번 내부 레지스터내로 시프팅된다. 외부 인터페이스가 관계되는 한 그러나, 단일 800 디프 시프트 레지스터가 존재한다.

모든 시프트 레지스터가 완전히 로딩되면(800 로드 펄스), 모든 비트는 적합한 노즐이네이블 비트에 병렬로 전송된다. 이것은 25,600 비트의 단일 병렬 전송과 같다. 전송이 일어난 직후, 인쇄 사이클은 시작할 수 있다. 모든 노즐이네이블 비트의 병렬 로드가 인쇄 사이클의 끝에서 일어나는 한 인쇄 사이클 및 로드 사이클이 동시에 일어날 수 있다.

6.1.2.1 로드 사이클

로드 사이클은 다음 인쇄 사이클의 노즐이네이블 비트를 갖는 프린트헤드의 시프트 레지스터를 로딩하는 것과 관계된다.

각각의 세그먼트는 시안, 마젠타, 옐로우 및 블랙 시프트 레지스터에 직접 관련된 4개의 입력을 갖는다. 이들 입력은 C데이터인, M데이터인, Y데이터인 및 K데이터인으로 불린다. 8개의 세그먼트가 존재하기 때문에, 4-인치 프린트헤드 당 총 32개의 컬러 입력 라인이 존재한다. SR클록 라인(모든 8개의 세그먼트 사이에 공유됨)상의 단일 펄스는 32 비트를 적합한

시프트 레지스터내로 전송한다. 교번 펄스는 하부 및 상부 노즐로 각각 비트를 전송한다. 25,600개의 노즐이 존재하기 때문에, 전송을 위해서 총 800 펄스가 요구된다. 모든 25,600 비트가 전송되고 나면, 공유된 P트랜스퍼 라인상의 단일 펄스가 시프트 레지스터로부터 적합한 노즐이네이블 비트로 데이터의 병렬 전송을 일으킨다.

P트랜스퍼상의 펄스를 통한 병렬 전송은 인쇄 사이클이 완료된 후에 일어나야 한다. 그렇지 않으면 인쇄될 라인에 대한 노즐이네이블 비트가 부정확해질 것이다.

모든 8개의 세그먼트가 단일 SR클록 펄스로써 로딩되기 때문에, 임의의 인쇄 프로세스는 프린트헤드에 대하여 옳은 시퀀스에서 데이터를 산출해야 한다. 예로서, 제 1 SR클록 펄스는 다음 인쇄 사이클의 도트 0, 800, 1600, 2400, 3200, 4000, 4800, 및 5600에 대하여 CMYK 비트를 전송할 것이다. 제 2 SR클록 펄스는 다음 인쇄 사이클의 도트 1, 801, 1601, 2401, 3201, 4001, 4801, 및 5601에 대하여 CMYK 비트를 전송할 것이고, P트랜스퍼 펄스는 주어질 수 있다.

홀수 및 짝수 CMYK 출력은, 동일 인쇄 사이클 동안에 인쇄되기는 하지만, 동일한 물리적 출력 라인상에 나타나지는 않는다는 것을 주목하는 것이 중요하다. 다른 컬러의 노즐 사이의 분리뿐만 아니라, 프린트헤드내의 홀수 및 짝수 노즐의 물리적 분리는 그들이 페이지의 다른 라인상에 도트를 산출할 것을 보장한다. 이러한 상대적 차이는 데이터를 프린트헤드내로 로딩할 때 고려되어야 한다. 라인에서의 실제 차이는 프린트헤드에서 사용되는 잉크젯 메카니즘의 특성에 의존한다. 차이는 변수(D_1 및 D_2)에 의해서 정해질 수 있는데, 여기서 D_1 은 다른 컬러의 노즐 사이의 거리이고 D_2 는 동일 컬러의 노즐 사이의 거리이다. 테이블 10은 처음 4개의 펄스상에서 프린트헤드의 세그먼트 n에 전송된 도트를 나타낸다.

(테이블 10)

4-인치 프린트헤드로 전송된 도트의 오더					
펄스	도트	블랙 라인	옐로우 라인	마젠타 라인	시안 라인
1	800S ^a	N	N+D ₁ ^b	N+2D ₁	N+3D ₁
2	800S+1	N+D ₂ ^c	N+D ₁ +D ₂	N+2D ₁ +D ₂	N+3D ₁ +D ₂
3	800S+2	N	N+D ₁	N+2D ₁	N+3D ₁
4	800S+3	N+D ₂	N+D ₁ +D ₂	N+2D ₁ +D ₂	N+3D ₁ +D ₂

^a S = 세그먼트 넘버(0-7)
^b D₁ = 하나의 컬러와 다음것의 노즐 사이의 라인의 수 (likely = 4 - 8)
^c D₂ = 동일 컬러의 노즐의 2개의 행 사이의 라인의 수 (likely = 1)

모든 800개의 펄스에 대해서 그러한 방식이다.

데이터는 20 MHz의 최대 비율로 프린트헤드내로 클럭킹될 수 있는데, 그것은 다음 라인에 대한 전체 데이터를 40μs에 로딩할 것이다.

6.1.2.2. 인쇄 사이클

4-인치 프린트헤드는 25,600개의 노즐을 포함한다. 그들을 모두 한번에 파이어링하는 것은 너무 많은 전력을 소비하고 잉크 리필 및 노즐 간섭의 견지에서 문제가 될 것이다. 결과적으로 2개의 파이어링 모드(저속 인쇄 모드 및 고속 인쇄 모드)가 정의된다:

- 저속 인쇄 모드에서, 200개의 페이지가 존재하고, 각각의 페이지는 128개의 노즐을 파이어링한다. 이것은 세그먼트 당 16개의 노즐 또는 파이어그룹 당 4개의 노즐과 동등하다.
- 고속 인쇄 모드에서, 100개의 페이지가 존재하고, 각각의 페이지는 256개의 노즐을 파이어링한다. 이것은 세그먼트 당 32개의 노즐 또는 파이어그룹 당 8개의 노즐과 동등하다.

일정 파이어링 펄스에서 파이어링되려는 노즐은

- 3 비트 크로마파드셀렉트(파이어그룹으로부터 5개의 크로마파드 중 1개를 선택)

- 4 비트 노즐선택트(파드로부터 10개의 노즐 중 1개를 선택)
- 2 비트의 파드그룹이네이블 라인(파이어링할 0, 1, 또는 2 파드그룹을 선택)에 의해서 결정된다.

파드그룹이네이블 라인 중 하나가 세팅될 때, 크로마파드선택트 및 노즐선택트에 의해서 결정된 바와 같이 지정된 파드그룹의 4개의 노즐만이 파이어링할 것이다. 2개의 파드그룹이네이블 라인이 모두 세팅될 때, 2개의 파드그룹은 그들의 노즐을 파이어링할 것이다. 저속 모드에 대해서, 2개의 파이어링 펄스가 요구되고, 파드그룹이네이블은 각각 10 및 01이다. 고속 모드에 대해서, 1개의 파이어링 펄스만이 요구되고, 파드그룹이네이블=11이다.

파이어링 펄스의 지속시간은 모든 파이어그룹으로부터 페이즈그룹A 및 페이즈그룹B 노즐을 각각 파이어링하는 A이네이블 및 B이네이블 라인에 의해서 주어진다. 파이어링 펄스의 전형적인 지속시간은 1.3 - 1.8 μ s이다. 펄스의 지속시간은 (온도 및 잉크 특성에 의존한) 잉크의 점성 및 프린트헤드에 이용 가능한 전력량에 의존한다. 온도 변화를 보상하기 위한 프린트헤드로부터의 피드백에 관한 상세 설명에 대해서 섹션 6.1.3을 보라.

A이네이블 및 B이네이블은 파이어링 펄스가 오버래핑할 수 있기 위해서 분리된 라인이다. 따라서 저속 인쇄 사이클의 200개의 페이즈는 100개의 A 페이즈 및 100개의 B 페이즈로 구성되고, 페이즈 A 및 페이즈 B의 100개의 세트를 효과적으로 준다. 마찬가지로, 고속 인쇄 사이클의 100개의 페이즈는 50개의 A 페이즈 및 50개의 B 페이즈로 구성되고, 페이즈 A 및 페이즈 B의 50개의 페이즈를 효과적으로 준다.

도 16은 전형적인 인쇄 사이클 동안 A이네이블(130) 및 B이네이블(131) 라인을 도시한다. 고속 인쇄에서는 각각 2 μ s의 50 사이클이 존재하고, 반면에 저속 인쇄에서는 각각 2 μ s의 100 사이클이 존재한다. 도면에 도시된 바와 같이, 공칭에 대하여 최소 및 최대 반 사이클 시간에서 약간의 변화가 용인 가능하다.

고속 인쇄 모드에 대해서, 파이어링 순서는:

- 크로마파드선택트 0, 노즐선택트 0, 파드그룹이네이블 11(페이즈 A 및 B)
- 크로마파드선택트 1, 노즐선택트 0, 파드그룹이네이블 11(페이즈 A 및 B)
- 크로마파드선택트 2, 노즐선택트 0, 파드그룹이네이블 11(페이즈 A 및 B)
- 크로마파드선택트 3, 노즐선택트 0, 파드그룹이네이블 11(페이즈 A 및 B)
- 크로마파드선택트 4, 노즐선택트 0, 파드그룹이네이블 11(페이즈 A 및 B)
- 크로마파드선택트 0, 노즐선택트 1, 파드그룹이네이블 11(페이즈 A 및 B)
- ...
- 크로마파드선택트 3, 노즐선택트 9, 파드그룹이네이블 11(페이즈 A 및 B)
- 크로마파드선택트 4, 노즐선택트 9, 파드그룹이네이블 11(페이즈 A 및 B)이다.

저속 인쇄 모드에 대해서, 파이어링 순서는 유사하다. 파드그룹이네이블이 11인 고속 모드의 각각의 페이즈에 대해서, 파드그룹이네이블 = 01 및 10의 2개의 페이즈로 다음과 같이 대체된다:

- 크로마파드선택트 0, 노즐선택트 0, 파드그룹이네이블 01(페이즈 A 및 B)
- 크로마파드선택트 0, 노즐선택트 0, 파드그룹이네이블 10(페이즈 A 및 B)
- 크로마파드선택트 1, 노즐선택트 0, 파드그룹이네이블 01(페이즈 A 및 B)
- 크로마파드선택트 1, 노즐선택트 0, 파드그룹이네이블 10(페이즈 A 및 B)

· ...

- 크로마파드셀렉트 3, 노즐셀렉트 9, 파드그룹이네이블 01(페이즈 A 및 B)
- 크로마파드셀렉트 3, 노즐셀렉트 9, 파드그룹이네이블 10(페이즈 A 및 B)
- 크로마파드셀렉트 4, 노즐셀렉트 9, 파드그룹이네이블 01(페이즈 A 및 B)
- 크로마파드셀렉트 4, 노즐셀렉트 9, 파드그룹이네이블 10(페이즈 A 및 B)

노즐이 파이어링할 때, 리필하는데 대략 100 μ s가 걸린다. 노즐은 이 리필 시간이 경과하기 전에는 파이어링될 수 없다. 이것은 가장 빠른 인쇄 속도를 라인 당 100 μ s로 제한한다. 고속 인쇄 모드에서, 라인을 인쇄하는 시간은 100 μ s이고, 그래서 하나의 라인으로부터 다음까지 노즐을 파이어링하는 사이의 시간은 리필 시간과 매칭한다. 저속 인쇄 모드는 이것보다 더 느리고, 그 또한 용인 가능하다.

노즐의 파이어링은 또한 그 노즐의 파드의 공통 잉크통내에 제한된 시간에 대해서 음향 섭동을 일으킨다. 그 섭동은 동일 파드내의 또 다른 노즐의 파이어링과 간섭할 수 있다. 결과적으로, 파드내의 노즐의 파이어링은 가능한 서로로부터 오프셋되어야 한다. 따라서 본 발명은 크로마파드로부터 4개의 노즐(컬러 당 1개의 노즐)을 파이어링하고 그 후 파드그룹내의 다음 크로마파드상으로 이동한다.

저속 인쇄 모드에서 파드그룹은 분리되어 파이어링된다. 따라서 2개의 파드그룹내의 5개의 크로마파드가 제 1 크로마파드가 다시 파이어링하기 전에 모두 파이어링해야 한다, 총 10 $\times$ 2 μ 사이클. 결과적으로 각각의 파드는 20 μ s 당 한번 파이어링된다.

고속 인쇄 모드에서, 파드그룹은 함께 파이어링된다. 따라서 단일 파드그룹내의 5개의 크로마파드는 제 1 크로마파드가 다시 파이어링하기 전에 모두 파이어링해야 한다, 총 5 $\times$ 2 μ 사이클. 결과적으로 각각의 파드는 10 μ s 당 한번 파이어링된다.

잉크 채널은 300 미크론 길이이고 잉크에서 음속은 약 1500m/s이기 때문에, 잉크 채널의 공진 주파수는 2.5MHz이다. 따라서 음향 펄스가 감쇠하기 위해서 저속 모드는 50 공진 사이클을 허용하고, 고속 모드는 25 공진 사이클을 허용한다. 결과적으로, 2가지 경우에서 임의의 음향 간섭이 최소화이다.

6.1.3 프린트헤드로부터 피드백

프린트헤드는 (8개의 세그먼트로부터 축적된) 피드백의 수개의 라인을 산출한다. 피드백 라인은 파이어링 펄스의 타이밍을 조절하는데 사용된다. 각각의 세그먼트는 동일 피드백을 산출하기는 하지만, 모든 세그먼트로부터의 피드백은 동일 3-상태 버스 라인을 공유한다. 결과적으로 한번에 1개의 세그먼트만이 피드백을 제공할 수 있다.

시안상의 데이터와 AND된 센스세그셀렉트 라인상의 펄스는 어느 세그먼트가 피드백을 제공할 것인지를 선택한다. 피드백 센스 라인은 다음 센스세그셀렉트 펄스 때까지 선택된 세그먼트로부터 올 것이다. 피드백 센스 라인은 다음과 같다:

- T센스는 프린트헤드가 얼마나 뜨거운지를 제어기에 알려준다. 이것은 온도가 잉크의 점성에 영향을 미치기 때문에, 제어기가 파이어링 펄스의 타이밍을 조절하게 한다.
- V센스는 얼마나 많은 전압이 액추에이터에 이용될 수 있는지를 제어기에 알려준다. 이것은 제어기가 펄스 폭을 조절함으로써 플랫 배터리 또는 고전압원에 대해 보상하게 한다.
- R센스는 액추에이터 히터의 저항률(스퀘어 당 옴)을 제어기에 알려준다. 이것은 히터 저항률에 관계없이 일정 에너지를 보전하기 위해서 제어기가 펄스 폭을 조절하게 한다.
- W센스는 리소그래픽 및 에칭 편차에 기인하여 $\pm 5\%$ 까지 변할 수 있는 히터의 임계부의 폭을 제어기에 알려준다. 이것은 제어기가 펄스 폭을 적절히 조절하게 한다.

6.1.4 예열 사이클

인쇄 프로세스는 평형 온도에 머무르려는 강한 경향을 갖는다. 인쇄된 포토그래프의 처음 섹션이 일관된 도트 크기를 갖는 것을 보장하기 위해서, 임의의 도트가 인쇄되기 전에 평형 온도가 충족되어야 한다. 이것은 예열 사이클을 통해서 성취된다.

예열 사이클은 1을 갖는 모든 노즐로의 단일 로드 사이클(즉, 모든 노즐을 파이어링으로 세팅), 및 각각의 노즐로의 많은 짧은 파이어링 펄스를 포함한다. 펄스의 지속시간은 방울을 파이어링하기에는 불충분하지만 잉크를 가열하기에는 충분해야 한다. 전체적으로 각각의 노즐에 대해서 약 200 펄스가 요구되고, 표준 인쇄 사이클로서 동일 시퀀스에서 사이클링 스루한다.

T센스에 의해서 예열 모드 동안 피드백이 제공되고, 평형 온도에 도달할 때(실온보다 높은 약 30℃)까지 계속된다. 예열 모드의 지속시간은 약 50 밀리초이고, 잉크 구성물에 의존한다.

예열은 각각의 인쇄 작업 전에 수행된다. 이것은 데이터가 프린터에 전송되고 있는 동안 그것이 행해질 때 성능에 영향을 미치지 않는다.

6.1.5 클리닝 사이클

노즐이 막히게 되는 가능성을 감소시키기 위해서, 각각의 인쇄 작업 전에 클리닝 사이클이 착수될 수 있다. 각각의 노즐은 흡수성 스폰지내로 많은 횡수 파이어링된다.

클리닝 사이클은 1을 갖는 모든 노즐로의 단일 로드 사이클(즉, 모든 노즐을 파이어링으로 세팅), 및 각각의 노즐로의 많은 파이어링 펄스를 포함한다. 노즐은 표준 인쇄 사이클로서 동일 노즐 파이어링 시퀀스를 통하여 클리닝된다. 각각의 노즐이 파이어링되는 횡수는 잉크 구성물, 및 프린터가 휴지하였던 시간에 의존한다. 예열과 같이, 클리닝 사이클은 프린터 성능에 어떠한 영향도 미치지 않는다.

6.1.6 프린트헤드 인터페이스 개요

단일 4-인치 프린트헤드는 테이블 11에 나타내어진 접속부를 갖는다.

(테이블 11)

이름	#핀	4-인치 프린트헤드 접속부 설명
크로마패드셀렉트	3	어느 크로마패드가 파이어링할지를 선택(0-4)
노즐셀렉트	4	패드로부터 어느 노즐이 파이어링할지를 선택(0-9)
패드그룹이네이블	2	패드그룹을 파이어링하게 함(초이스:01, 10, 11)
A이네이블	1	페이지그룹 A에 대하여 펄스를 파이어링
B이네이블	1	페이지그룹 B에 대하여 펄스를 파이어링
C데이터인[0-7]	8	세그먼트 0-7의 시안 시프트 레지스터로 시안 입력
M데이터인[0-7]	8	세그먼트 0-7의 마젠타 시프트 레지스터로 마젠타 입력
Y데이터인[0-7]	8	세그먼트 0-7의 옐로우 시프트 레지스터로 옐로우 입력
K데이터인[0-7]	8	세그먼트 0-7의 블랙 시프트 레지스터로 블랙 입력
SR클록	1	SR클록(시프트레지스터클록)상의 펄스가 C데이터인[0-7], M데이터인[0-7], Y데이터인[0-7], 및 K데이터인[0-7]으로부터의 현재값을 32 시프트 레지스터내로 로딩한다.
P트랜스퍼	1	시프트 레지스터로부터 내부 노즐이네이블 비트(노즐 당 하나)로 데이터의 병렬 전송
센스세그셀렉트	1	C데이터인[n]상의 데이터와 AND된 센스세그셀렉트상의 펄스가 세그먼트 n에 대하여 센스 라인을 선택한다.
T센스	1	온도 감지
V센스	1	전압 감지
R센스	1	저항률 감지

W센스	1	폭 감지
로직GND	1	로직 그라운드
로직PWR	1	로직 전력
V-	버스 바	액추에이터 그라운드
V+		액추에이터 전력
총	52	

4-인치 프린트헤드로의 내부에서, 각각의 세그먼트는 테이블 12에 나타내어진 집합 패드로의 접속부를 갖는다.

(테이블 12)

4 인치 프린트헤드 내부 세그먼트 접속부		
이름	#핀	설명
크로마패드셀렉트	3	어느 크로마패드가 파이어링할지를 선택(0-4)
노즐셀렉트	4	패드로부터 어느 노즐이 파이어링할지를 선택(0-9)
패드그룹이네이블	2	패드그룹을 파이어링하게 함(초이스:01, 10, 11)
A이네이블	1	패드그룹 A에 대하여 펄스를 파이어링
B이네이블	1	패드그룹 B에 대하여 펄스를 파이어링
C데이터인	1	시안 시프트 레지스터로 시안 입력
M데이터인	1	마젠타 시프트 레지스터로 마젠타 입력
Y데이터인	1	옐로우 시프트 레지스터로 옐로우 입력
K데이터인	1	블랙 시프트 레지스터로 블랙 입력
SR클록	1	SR클록(시프트레지스터클록)상의 펄스가 C데이터인, M데이터인, Y데이터인 및 K데이터인으로부터의 현재값을 4 시프트 레지스터내로 로딩한다.
P트랜스퍼	1	시프트 레지스터로부터 내부 노즐이네이블 비트(노즐 당 하나)로 데이터의 병렬 전송
센스세그셀렉트	1	C데이터인상의 데이터와 AND된 센스세그셀렉트상의 펄스가 이 세그먼트에 대하여 센스 라인을 선택한다.
T센스	1	온도 감지
V센스	1	전압 감지
R센스	1	저항률 감지
W센스	1	폭 감지
로직GND	1	로직 그라운드
로직PWR	1	로직 전력
V-	21	액추에이터 그라운드
V+	21	액추에이터 전력
총	66	모든 세그먼트에 대하여 66 × 8 세그먼트 = 528

6.2 8-인치 프린트헤드 고찰

8-인치 멤젯 프린트헤드는 단순히 물리적으로 함께 놓인 2개의 4-인치 프린트헤드이다. 제어 칩으로부터의 핀의 수가 감소되고 2개의 프린트헤드가 동시에 인쇄할 수 있기 위해서 프린트헤드는 함께 배선되고 많은 공통 접속부를 공유한다. 이 때문에 많은 세부사항이 고려되어야 한다.

6.2.1 접속부

2개의 프린트헤드로부터의 노즐의 파이어링은 동시에 일어나기 때문에, 크로마패드셀렉트, 노즐셀렉트, A이네이블 및 B이네이블 라인이 공유된다. 데이터로써 프린트헤드를 로딩하기 위해서, C데이터인, M데이터인, Y데이터인, K데이터인의 32 라인이 공유되고, 2개의 다른 SR클록 라인이 2개의 프린트헤드 중 어느 것이 로딩될 것인지를 결정하는데 사용된다. 단일 P트랜스퍼 펄스는 로딩된 데이터를 2개의 프린트헤드에 대한 노즐이네이블 비트내로 전송하는데 사용된다. 마찬가지로, T센스, V센스, R센스, 및 W센스 라인이 공유되고, 2개의 센스이네이블 라인은 2개의 프린트헤드 사이를 구별한다.

따라서 2개의 4-인치 프린트헤드는 SR클록 및 센스이네이블을 제외하고 모든 접속부를 공유한다. 이들 2개의 접속부는 각각의 프린트헤드에 대하여 한번 반복된다. 실제의 접속부가 테이블 13에 나타내어진다.

(테이블 13)

8-인치 프린트헤드 접속부		
이름	#핀	설명
크로마파드셀렉트	3	어느 크로마파드가 파이어링할지를 선택(0-4)
노즐셀렉트	4	파드로부터 어느 노즐이 파이어링할지를 선택(0-9)
파드그룹이네이블	2	파드그룹을 파이어링하게 함(초이스:01, 10, 11)
A이네이블	1	파드그룹 A에 대하여 펄스를 파이어링
B이네이블	1	파드그룹 B에 대하여 펄스를 파이어링
C데이터인[0-7]	8	세그먼트 0-7의 시안 시프트 레지스터로 시안 입력
M데이터인[0-7]	8	세그먼트 0-7의 마젠타 시프트 레지스터로 마젠타 입력
Y데이터인[0-7]	8	세그먼트 0-7의 옐로우 시프트 레지스터로 옐로우 입력
K데이터인[0-7]	8	세그먼트 0-7의 블랙 시프트 레지스터로 블랙 입력
SR클록1	1	SR클록(시프트레지스터클록)상의 펄스가 C데이터인[0-7], M데이터인[0-7], Y데이터인[0-7] 및 K데이터인[0-7] 으로부터의 현재값을 4-인치 프린트헤드 1에 대한 32 시프트 레지스터내로 로딩한다.
SR클록2	1	SR클록(시프트레지스터클록)상의 펄스가 C데이터인[0-7], M데이터인[0-7], Y데이터인[0-7] 및 K데이터인[0-7] 으로부터의 현재값을 4-인치 프린트헤드 2에 대한 32 시프트 레지스터내로 로딩한다.
P트랜스퍼	1	시프트 레지스터로부터 내부 노즐이네이블 비트(노즐 당 하나)로 데이터의 병렬 전송
센스세그셀렉트1	1	C데이터인[n]상의 데이터와 AND된 4-인치 프린트헤드 1의 센스세그셀렉트 라인상의 펄스가 세그먼트 n에 대하여 센스 라인을 선택한다.
센스세그셀렉트2	1	C데이터인[n]상의 데이터와 AND된 4-인치 프린트헤드 2의 센스세그셀렉트 라인상의 펄스가 세그먼트 n에 대하여 센스 라인을 선택한다.
T센스	1	온도 감지
V센스	1	전압 감지
R센스	1	저항률 감지
W센스	1	폭 감지
로직GND	1	로직 그라운드
로직PWR	1	로직 전력
V-	버스 바	액추에이터 그라운드
V+		액추에이터 전력
총	54	

6.2.2 타이밍

2개의 4-인치 프린트헤드의 결합 및 적합한 접속부의 배선은 8-인치 와이드 이미지가 4-인치 와이드 이미지만큼 빠르게 인쇄되게 한다. 그러나, 다음 라인이 인쇄될 수 있기 전에 2개의 프린트헤드로 전송되는 데이터는 두배 많다. 출력 이미지가 인쇄되는 소망 속도에 의존해서, 지속하기 위한 적합한 속도로 데이터가 발생되고 전송되어야 한다.

6.2.2.1 예

예로서, 8"×12" 페이지를 2초에 인쇄하는 타이밍을 고려하자. 이 페이지를 2초에 인쇄하기 위해서, 8-인치 프린트헤드는 19,200 라인(12×1600)을 인쇄해야 한다. 나머지 올림하여 2초에 20,000 라인으로 생각하면 100μs의 라인 시간을 산출한다. 단일 인쇄 사이클 및 단일 로드 사이클은 모두 이 시간내에 완료되어야 한다. 더하여, 프린트헤드의 외부의 물리적 프로세스는 용지를 적합한 양 이동시켜야 한다.

인쇄 관점으로부터, 고속 인쇄 모드는 4-인치 프린트헤드가 전체 라인을 100μs에 인쇄하게 한다. 따라서 4-인치 프린트헤드는 동시에 인쇄하기 위해서 둘 다 고속 인쇄 모드에서 러닝되어야 한다. 따라서 파이어링 펄스 당 512개의 노즐이 파이어링함으로써, 8-인치 라인의 인쇄가 지정된 시간내에 가능하게 한다.

2개의 4-인치 프린트헤드로의 800 SR클럭 펄스(각각의 클럭 펄스는 32 비트를 전송)는 또한 100 μ s 라인 시간내에 일어나야 한다. 2개의 프린트헤드가 동시에 로딩된다면(64 데이터 라인), SR클럭 펄스의 길이는 $100\mu\text{s}/800 = 125$ 나노초를 초과할 수 없으며, 프린트헤드가 8MHz에서 클럭킹되어야함을 나타낸다. 2개의 프린트헤드가 한번에 1개 로딩된다면(32 공유 데이터 라인), SR클럭 펄스의 길이는 $100\mu\text{s}/1600 = 62.5$ 나노초를 초과할 수 없다. 따라서 프린트헤드는 16MHz에서 클럭킹되어야 한다. 두 경우에서, (51,200개의 노즐의 각각에 대해서) 각각의 비트 값을 계산하기 위한 평균 시간은 $100\mu\text{s}/51,200 = 2$ 나노초를 초과하지 않아야 한다. 이것은 도트 발생기가 다음의 속도 중 하나에서 러닝할 것을 요구한다:

- 사이클 당 1 비트(도트)를 발생시키는 500 MHz
- 사이클 당 2 비트(도트)를 발생시키는 250 MHz
- 사이클 당 4 비트(도트)를 발생시키는 125 MHz

7 프린터 제어기

7.1 프린터 제어기 아키텍처

프린터 제어기는 도 8에 도시된 바와 같이, 아이프린트 중앙 프로세서(ICP) 칩(83), 64MBit RDRAM(82), 및 마스터 QA 칩(85)으로 구성된다.

ICP(83)는 도 17에 도시된 바와 같이, 범용-목적 프로세서(139) 및 프로세서 버스를 통해서 프로세서에 의해서 제어되는 목적-특정 기능 유닛의 세트를 포함한다. 3개의 기능 유닛-EDRL 확장기(140), 하프토너/합성기(141), 및 맵젯 프린트헤드를 제어하는 프린트헤드 인터페이스(142)만이 비-표준이다.

프로세서상에서 러닝하는 소프트웨어는 다양한 기능 유닛이 페이지를 수신, 확장 및 인쇄하도록 코디네이팅한다. 이것은 다음 섹션에서 설명된다.

ICP의 다양한 기능 유닛이 후속 섹션에서 설명된다.

7.2 페이지 확장 및 인쇄

페이지 확장 및 인쇄는 다음과 같이 진행된다. 페이지 기술이 USB 인터페이스(146)를 통해서 호스트로부터 수신되고 주 메모리에 저장된다. 주 메모리의 6MB는 페이지 저장에 전용된다. 이것은 2개의 페이지가 각각 3MB를 초과하지 않게 유지하거나 또는 1개의 페이지를 6MB까지 유지할 수 있다. 호스트가 3MB를 초과하지 않는 페이지를 발생시키면, 그 때 프린터는 스트리밍 모드에서 동작한다 - 즉, 프린터는 다음 페이지를 수신하는 동안 하나의 페이지를 인쇄한다. 호스트가 3MB를 초과하는 페이지를 발생시키면, 그 때 프린터는 단일-페이지 모드에서 동작한다 - 즉, 프린터는 다음 페이지를 수신하기 전에 각각의 페이지를 수신하여 인쇄한다. 호스트가 6MB를 초과하는 페이지를 발생시키면 그 때 그것은 프린터에 의해서 거절된다. 실제로 프린터 드라이버는 이것이 일어나는 것을 방지한다.

페이지는 2개의 파트 - 2-레벨 블랙 층, 및 연속톤 층으로 구성된다. 이들은 별개의 포맷 - 2-레벨 블랙 층은 EDRL 포맷, 연속톤 층은 JPEG 포맷으로 압축된다. 페이지 확장의 제 1 단계는 2개의 층을 병렬로 압축해제하는 것으로 구성된다. 2-레벨 층은 EDRL 확장기 유닛(140)에 의해서 압축해제되고(16), 연속톤 층은 JPEG 디코더(143)에 의해서 압축해제된다(14).

페이지 확장의 제 2 단계는 연속톤 CMYK 데이터를 2-레벨 CMYK로 하프토닝하고(15), 그 후 2-레벨 블랙 층을 2-레벨 CMYK 층 위로 합성하는(17) 것으로 구성된다. 하프토닝 및 합성은 하프토너/합성기 유닛(141)에 의해서 수행된다.

최종적으로, 합성된 2-레벨 CMYK 이미지는 맵젯 프린트헤드를 제어하는 프린트헤드 인터페이스 유닛(142)을 통해서 인쇄된다(18).

멤젯 프린트헤드는 고속으로 인쇄하기 때문에, 용지는 일정 속도로 프린트헤드를 지나 이동해야 한다. 데이터가 충분히 빠르게 프린트헤드에 공급될 수 없기 때문에 용지가 중지한다면, 그 때 눈에 보이는 인쇄 불규칙성이 일어날 것이다. 따라서 요구되는 비율로 2-레벨 CMYK 데이터를 프린트헤드에 전송하는 것이 중요하다.

완전히-확장된 1600 dpi 2-레벨 CMYK 페이지는 114.3MB의 크기를 갖는다. 확장된 페이지를 프린터 메모리에 저장하는 것은 비실용적이기 때문에, 각각의 페이지는 인쇄 동안 실시간으로 확장된다. 따라서 페이지 확장 및 인쇄의 다양한 단계가 파이프라인된다. 페이지 확장 및 인쇄 데이터 흐름이 테이블 14에 설명된다. 주 메모리로부터/로의 총 트래픽 174MB/s는 램버스와 같은 현재 기술의 능력내에 만족된다.

(테이블 14)

페이지 확장 및 인쇄 데이터 흐름						
프로세스	입력	입력 윈도우	출력	출력 윈도우	입력 비율	출력 비율
연속톤 수신	-	-	JPEG 스트림	1	-	1.5MB/s 3.3Mp/s
2-레벨 수신	-	-	EDRL 스트림	1	-	1.5MB/s 30Mp/s
연속톤 압축해제	JPEG 스트림	-	32-비트 CMYK	8	1.5MB/s 3.3Mp/s	13MB/s 3.3Mp/s
2-레벨 압축해제	EDRL 스트림	-	1-비트 K	1	1.5MB/s 30Mp/s ^a	14MB/s 120Mp/s
하프톤	32-비트 CMYK	1	_b	-	13MB/s 3.3Mp/s ^c	-
합성	1-비트 K	1	4-비트 CMYK	1	14MB/s 120Mp/s	57MB/s 120Mp/s
인쇄	4-비트 CMYK	24, 1 ^d	-	-	57MB/s 120Mp/s	- -
					87MB/s	87MB/s
						174MB/s

^a 800 dpi => 1600 dpi (2×2 확장)
^b 하프톤이 합성과 조합, 그래서 그들 사이에 어떠한 외부 데이터 흐름도 존재하지 않는다
^c 267 ppi => 1600 dpi (6×6 확장)
^d 24라인의 윈도우가 필요하지만, 1 라인만 전진한다

각각의 단계는 주 메모리에서의 공유 FIFO를 통해서 다음 단계와 통신한다. 각각의 FIFO는 라인으로 편성되고 각각의 FIFO의 (라인에서의) 최소 크기는 생산자의 (라인에서의) 출력 윈도우 및 소비자의 (라인에서의) 입력 윈도우를 수용하도록 디자인된다. 단계-간 주 메모리 버퍼가 테이블 15에서 설명된다. 6.3MB의 총 버퍼 공간 이용은 (이용 가능한 8MB로부터) 프로그램 코드 및 스크래치 메모리에 대하여 1.7MB를 자유로이 남겨둔다.

(테이블 15)

페이지 확장 및 인쇄 주 메모리 버퍼			
버퍼	편성 및 라인 크기	라인의 수	버퍼 크기
압축된 페이지 버퍼(146)	바이트 스트림 (1개 또는 2개의 페이지)	-	6MB
연속톤 CMYK 버퍼 (147)	32-비트 인터리빙된 CMYK (267 ppi × 8" × 32 = 8.3KB)	8 × 2 = 16	134KB
2-레벨 K 버퍼 (148)	1-비트 K (800 dpi × 8" × 1 = 1.5KB)	1 × 2 = 2	3KB
2-레벨 CMYK 버퍼 (149)	4-비트 플래너 홀수/짝수 CMYK (1600 dpi × 8" × 4 = 6.3KB)	24 + 1 = 25	156KB
			6.3MB

FIFO를 포함한 전체 데이터 흐름이 도 18에 예시된다.

연속된 페이지 압축해제는 JPEG 디코더(143)에 의해서 수행된다. 2-레벨 페이지 압축해제는 EDRL 확장기(140)에 의해서 수행된다. 하프토닝 및 합성은 하프토너/합성기 유닛(141)에 의해서 수행된다. 이들 기능 유닛이 이하 섹션에서 설명된다.

7.2.1 DMA 접근법

각각의 기능 유닛은 하나 이상의 온-칩 입력 및/또는 출력 FIFO를 포함한다. 각각의 FIFO는 다중-채널 DMA 제어기(144)에서의 분리된 채널에 할당된다. DMA 제어기(144)는 이중-번지 전송보다는 단일-번지를 처리하고, 그래서 각각의 채널에 대해 분리된 요청/긍정응답 인터페이스를 제공한다.

각각의 기능 유닛은 입력 FIFO가 소진되거나 출력 FIFO가 채워질 때마다 살며시 기능정지된다.

프로세서(139)는 각각의 DMA 전송을 프로그래밍한다. DMA 제어기(144)는 채널에 접속된 기능 유닛으로부터의 요청시 전송의 각각의 워드에 대하여 번지를 발생시킨다. 기능 유닛은 그 요청이 DMA 제어기(144)에 의해서 긍정응답될 때 데이터 버스(145)상으로 또는 떨어져 워드를 래칭한다. DMA 제어기(144)는 전송이 완료될 때 프로세서(139)를 인터럽팅하고, 따라서 프로세서(139)가 시기적절하게 동일 채널상에서 또 다른 전송을 프로그래밍하게 한다.

일반적으로 프로세서(139)는 대응 주 메모리 FIFO가 이용 가능하자마자(즉, 판독에 대해서 비어있지 않고, 기록에 대해서 꽉 차 있지 않음) 채널상에서 또 다른 전송을 프로그래밍할 것이다.

DMA 제어기(144)에서 구현된 채널 서비스의 입상(granularity)은 주 메모리의 대기시간(latency)에 다소 의존한다.

7.2.2 EDRL 확장기

도 19에 도시된 EDRL 확장기 유닛(EEU; 140)은 EDRL-압축된 2-레벨 이미지를 압축해제한다.

EEU로의 입력은 EDRL 비트스트림(150)이다. EEU로부터의 출력은 정수 환산 계수에 의해서 확장된 2-레벨 이미지의 해상도로부터 1600 dpi로 수평으로 스케일링된 2-레벨 이미지 라인의 세트(151)이다.

시작된 직후, EEU는 그것이 EDRL 비트스트림에서 페이지 끝 코드를 검출할 때까지 또는 그것이 그 제어 레지스터를 통해서 명시적으로 중지될 때까지 진행한다.

EEU는 비트스트림을 디코딩하기 위해서 명시적인 페이지 폭에 의존한다. 이것은 EEU를 시작하기 이전에 페이지 폭 레지스터(152)에 기록되어야 한다.

확장된 2-레벨 이미지의 스케일링은 명시적인 환산 계수에 의존한다. 이것은 EEU를 시작하기 이전에 환산 계수 레지스터(153)에 기록되어야 한다.

(테이블 16)

EDRL 확장기 제어 및 구성 레지스터		
레지스터	폭	설명
시작	1	EEU를 시작
중지	1	EEU를 중지
페이지 폭	13	라인 끝을 검출하기 위해서 디코딩 동안에 사용된 페이지 폭
환산 계수	4	확장된 이미지의 스케일링 동안 사용된 환산 계수

EDRL 압축 포맷이 섹션 5.2.3에서 설명된다. 그것은 2-레벨 이미지를 그 예지의 견지에서 표현한다. 각각의 라인에서의 각각의 예지는 이전 라인의 예지에 관하여 또는 동일 라인의 이전 예지에 관하여 코딩된다. 그것이 어떻게 코딩되던간에, 결국 각각의 예지는 동일 라인에서의 이전 예지로부터의 그 거리로 디코딩된다. 그 후 이 거리, 또는 런길이는 이미지의 대응 파트를 표현하는 제로 비트 또는 일 비트의 스트링으로 디코딩된다. 섹션 5.2.3.2에서 압축해제 알고리즘이 또한 정의된다.

EEU는 비트스트림 디코더(154), 상태기(155), 에지 계산 로직(156), 2개의 런길이 디코더(157 및 158), 및 런길이 (리)인 코더(159)로 구성된다.

비트스트림 디코더(154)는 비트스트림으로부터 엔트로피-코딩된 코드워드를 디코딩하고 그것을 상태기(155)로 넘겨준다. 상태기(155)는 코드워드의 크기를 비트스트림 디코더(154)로 반환하는데, 이것은 디코더(154)가 다음 코드워드로 전진하게 한다. 생성 에지 코드의 경우에서, 상태기(155)는 비트스트림으로부터 대응 런길이를 끌어내기 위해서 비트스트림 디코더를 사용한다. 상태기는 테이블 18에 정의된 바와 같이 런길이 디코딩/인코딩 및 에지 계산 로직을 제어한다.

에지 계산 로직은 매우 간단하다. 이전 (기준) 및 현재 (코딩) 라인에서의 현재 에지 오프셋이 기준 에지 레지스터(160) 및 에지 레지스터(161)에 각각 유지된다. 생성 에지 코드와 연관된 런길이는 런길이 디코더로 직접 출력되고, 현재 에지에 추가된다. 델타 코드는 연관된 델타를 기준 에지에 추가하고 현재 에지를 공제함으로써 런길이를 번역된다. 발생된 런길이 트 런길이 디코더로 출력되고, 현재 에지에 추가된다. 다음 런길이는 런길이 인코더(159)로부터 끌어내어지고 기준 에지 (160)에 추가된다. 소거 에지 코드는 현재 기준 에지가 스킵핑되게 할뿐이다. 다시 다음 런길이가 런길이 인코더로부터 끌어내어지고 기준 에지에 추가된다.

에지 계산 로직(156)이 에지를 표현하는 런길이를 발생시킬 때마다, 그것은 런길이 디코더로 넘겨진다. 런길이 디코더가 그 런을 디코딩하는 동안 그것은 기능정지 신호를 상태기에 발생시킨다. 런길이 디코더(157)가 에지 계산 로직보다 더 느리기 때문에, 그것을 감결합할 때 그다지 많은 포인트가 존재하는 것은 아니다. 확장된 라인은 8" 800 dpi 라인(800 바이트)을 유지하기에 충분히 큰 라인 버퍼(162)에 축적된다.

이전에 확장된 라인은 또한 버퍼링된다(163). 그것은 현재 라인의 디코딩에 대하여 기준으로서 행동한다. 이전 라인은 런길이 온 디맨드로서 재-인코딩된다. 이것은 최악의 경우가 각각의 화소에 대해 하나의 13-비트 런길이이기 때문에(1600dpi에서 20KB), 이전 라인의 디코딩된 런길이를 버퍼링하는 것보다 값이 덜 든다. 런길이 인코더(159)가 그 런을 인코딩하는 동안 그것은 기능정지 신호를 상태기에 발생시킨다. 런길이 인코더는 라인 끝을 검출하기 위해서 페이지 폭(152)을 사용한다. (현재) 라인 버퍼(162) 및 이전 라인 버퍼(163)는 런길이 인코더(159)를 단순화하기 위해서 단일 FIFO로서 결부되어 관리된다.

런길이 디코더(158)는 8" 1600 dpi 라인(1600 바이트)을 유지하기에 충분히 큰 라인 버퍼(164)로 출력 런길이를 디코딩한다. 이 출력 런길이 디코더로 넘겨지는 런길이는 환산 계수(153)만큼 곱해지고, 그래서 이 디코더는 1600 dpi 라인을 산출한다. 라인은 출력 화소 FIFO(165)를 통해서 환산 계수 횟수번 출력된다. 이것은 간단한 라인 복제에 의해서 요구되는 수직 스케일링을 달성한다. EEU는 그 이미지 스케일링내로 집적된 에지 스무딩(smoothing)으로써 디자인될 수 있다. 템플레이트-매칭에 기초한 단순한 스무딩 방법은 매우 효과적인 수 있다[10]. 이것은 저해상도 런길이 디코더 및 스무드 스케일링 유닛 사이에 다중-라인 버퍼를 요구할 것이지만, 그러나 고해상도 런길이 디코더를 제거할 것이다.

7.2.2.1 EDRL 스트림 디코더

도 20에 예시된 EDRL 스트림 디코더(154)는 입력 비트스트림에서 엔트로피-코딩된 EDRL 코드워드를 디코딩한다. 그것은 그 좌측 (가장 중요한) 에지가 비트스트림에서 코드워드 경계로 항상 정렬되는 16-비트 배럴 시프터(168)를 통하여 뷰잉된 2개의-바이트 입력 버퍼(167)를 사용한다. 배럴 시프터(168)에 접속된 디코더(169)는 테이블 17에 따라 코드워드를 디코딩하고, 상태기(155)에 대응 코드를 공급한다.

(테이블 17)

EDRL 스트림 코드워드 디코딩 테이블		
입력 코드워드 비트 패턴 ^a	출력 코드	출력 코드 비트 패턴
1xxx xxxx	$\Delta 0$	1 0000 0000
010x xxxx	$\Delta +1$	0 1000 0000
011x xxxx	$\Delta -1$	0 0100 0000
0010 xxxx	소거 에지	0 0010 0000
0011 xxxx	가까운 에지 생성	0 0001 0000
0001 0xxx	$\Delta +2$	0 0000 1000
0001 1xxx	$\Delta -2$	0 0000 0100
0000 1xxx	먼 에지 생성	0 0000 0010

0000 01xx	페이지 끝(EOP)	0 0000 0001
^a x = 무시		

상태기(155)는 차례로 코드의 길이를 출력한다. 이것은 다음 코드워드 비트 오프셋을 산출하기 위해서 현재 코드워드 비트 오프셋에 부가된다(170; 모듈로-8). 비트 오프셋은 차례로 배럴 시프터(168)를 제어한다. 코드워드 비트 오프셋이 랩핑한다면, 그 때 캐리 비트는 입력 FIFO(166)로부터의 다음 바이트의 래칭을 제어한다. 이번에 바이트 2가 바이트 1로 래칭되고, FIFO 출력은 바이트 2로 래칭된다. 입력 버퍼를 채우는데 길이 8의 두 사이클이 걸린다. 이것은 상태기(155)에서 상태를 시작함으로써 처리된다.

7.2.2.2 EDRL 확장기 상태기

EDRL 확장기 상태기(155)는 EDRL 스트림 디코더(154)에 의해서 공급된 코드에 응답하여 예지 계산 및 런길이 확장 로직을 제어한다. 그것은 EDRL 스트림 디코더에 현재 코드워드의 길이를 공급하고 예지 계산 로직에 현재 델타 코드와 연관된 델타값을 공급한다. 상태기는 또한 제어 레지스터로부터의 시작 및 중지 제어 신호, 및 예지 계산 로직으로부터의 라인 끝(EOL) 신호에 응답한다.

또한 상태기는 생성 예지 코드와 연관된 런길이의 다중-사이클 인출(fetch)를 제어한다.

(테이블 18)

EDRL 확장기 상태기						
입력 신호	입력 코드	현재 상태	다음 상태	코드 길이	델타	조치
시작	-	중지됨	시작함	8	-	-
-	-	시작함	휴지	8	-	-
중지	-	-	중지됨	0	-	RL 디코더 및 FIFO를 리셋
EOL	-	-	EOL 1	0	-	RL 인코더를 리셋; RL 디코더를 리셋; 기준 예지 및 예지를 리셋
-	-	EOL 1	휴지			RL 인코더=>기준 RL; 기준 예지+=기준 RL
-	00	휴지	휴지	1	0	RL=예지-기준 예지+델타; 예지+=RL; RL=>RL 디코더; RL 인코더=>기준 RL; 기준 예지+=기준 RL
-	$\Delta+1$	휴지	휴지	2	+1	"
-	$\Delta-1$	휴지	휴지	3	-1	"
-	$\Delta+2$	휴지	휴지	4	+2	"
-	$\Delta-2$	휴지	휴지	5	-2	"
-	소거 예지	휴지	휴지	6	-	RL 인코더=>기준 RL; 기준 예지+=기준 RL
-	가까운 예지 생성	휴지	RL 로 7 생성	7	-	생성 RL 리셋
-	먼 예지 생성	휴지	RL 하이 6 생성	8	-	-
-	EOP	휴지	중지됨	8	-	-
-	-	RL 하이 6 생성	RL 로 7 생성	6	-	생성 RL 하이 6 래치
-	-	RL 로 7 생성	예지 생성	7	-	생성 RL 로 7 래치
-	-	예지 생성	휴지	0	-	RL=RL 생성; 예지+=RL; RL=>RL 인코더

7.2.2.3 런길이 디코더

도 21에 도시된 런길이 디코더(157/158)는 런길이를 출력 스트림에서의 대응 길이의 일 비트 또는 제로 비트의 시퀀스로 확장한다. 라인에서의 제 1 런은 화이트(컬러0)라 가정된다. 각각의 런은 그 선행자에 반대하는 컬러라 가정된다. 제 1 런이 실제로 블랙(컬러1)이라면, 그 때 그것은 제로-길이 화이트 런의 뒤에 와야만 한다. 런길이 디코더는 내부적으로 현재 컬러를 기억하고 있다.

런길이 디코더는 매 클록마다 최대 8 비트를 출력 스트림에 첨부한다. 전형적으로 런길이는 8의 정수 배수가 아니고, 그래서 이미지에서의 제 1의 것 외의 런은 전형적으로 바이트-정렬되지 않는다. 런 디코더는, 바이트 공간 레지스터(180)에서, 현재 지어진 바이트에서 이용 가능한 비트의 수를 유지한다. 이것은 디코딩의 시작에서, 및 모든 바이트의 출력상에서 8로 초기화된다.

디코더는 다음 런 라인이 제로-아닌 값을 런길이 레지스터(181)내로 래칭하자마자 비트의 런을 출력하는 것을 시작한다. 디코더는 런길이 레지스터가 제로로 갈 때 효과적으로 기능정지된다.

현재 컬러의 많은 비트는 각각의 클록마다 출력 바이트 레지스터(182)내로 시프팅된다. 현재 컬러는 1-비트 컬러 레지스터(183)에 유지관리된다. 실제로 출력되는 비트의 수는 런길이에 남겨진 비트의 수, 및 출력 바이트에 남겨진 스페어 비트의 수에 의해서 제한된다. 출력되는 비트의 수는 런길이 및 바이트 공간으로부터 공제된다. 런의 트레일링 비트가 아직 출력 바이트 레지스터, 펜딩 출력에 있을 수 있기는 하지만, 런길이가 제로로 갈 때 그것은 완벽히 디코딩되었다. 바이트 공간이 제로로 갈 때 출력 바이트는 완전하고 출력 스트림에 첨부된다.

16-비트 배럴 시프터(184), 출력 바이트 레지스터(182) 및 컬러 레지스터(183)는 매 클록마다 다수 비트 위치로 시프팅될 수 있고 컬러를 직렬 입력으로서 갖는 8-비트 시프트 레지스터를 함께 구현한다.

외부 리셋 라인은 라인의 시작에서 런길이 디코더를 리셋하는데 사용된다. 외부 다음 런 라인은 새로운 런길이의 디코딩을 요청하는데 사용된다. 그것은 외부 런길이 라인상의 런길이에 수반된다. 다음 런 라인은 리셋 라인으로서 동일 클록상에 세팅되지 않아야 한다. 다음 런은 현재 컬러를 인버팅하기 때문에, 컬러의 리셋은 그것을 제로가 아닌 일로 세팅한다. 외부 플러시(flush) 라인은, 불완전하다면, 런의 마지막 바이트를 플러싱하는데 사용된다. 그것은 바이트-정렬된 라인을 산출하기 위해서 라인-라인 기초 또는 바이트-정렬된 이미지를 산출하기 위해서 이미지 기초상에서 사용될 수 있다.

외부 준비 라인은 런길이 디코더가 런길이를 디코딩할 준비가 되어 있는지를 나타낸다. 그것은 외부 로직을 기능정지하게 하는데 사용될 수 있다.

7.2.2.4 런길이 인코더

도 22에 도시된 런길이 인코더(159)는 입력 스트림에서의 제로 또는 일 비트의 런을 검출한다. 라인에서의 제 1 런은 화이트(컬러0)라 가정된다. 각각의 런은 그 선행자에 반대하는 컬러라 가정된다. 제 1 런이 실제로 블랙(컬러1)이라면, 그 후 런길이 인코더는 라인의 시작에서 제로-길이 화이트 런을 발생시킨다. 런길이 디코더는 현재 컬러를 내부적으로 기억하고 있다.

런길이 인코더는 매 클록마다 입력 스트림으로부터 최대 8 비트를 판독한다. 그것은 그 좌측 (가장 중요한) 에지가 비트 스트림에서의 현재 위치에 항상 정렬되는 16-비트 배럴 시프터(191)를 통해서 뷰잉된 2개의-바이트 입력 버퍼(190)를 사용한다. 배럴 시프터에 접속된 인코더(192)는 테이블 19에 따라서 8-비트 (부분) 런길이를 인코딩한다. 인코더(192)는 적합한 컬러의 런을 인식하기 위해서 현재 컬러를 사용한다.

8-비트 런길이 인코더에 의해서 발생된 8-비트 런길이는 런길이 레지스터(193)에서의 값에 부가된다. 8-비트 런길이 인코더가 현재 런의 끝을 인식할 때 그것은 준비 레지스터(194)에 의해서 래칭되는 런 끝 신호를 발생시킨다. 준비 레지스터(194)의 출력은 인코더가 런길이 레지스터(193)에 축적된 현재 런길이를 인코딩하는 것을 완료했음을 나타낸다. 또한 준비 레지스터(194)의 출력은 8-비트 런길이 인코더(192)를 기능정지하게 하는데 사용된다. 기능정지될 때 8-비트 런길이 인코더(192)는 제로-길이 런 및 제로 런 끝 신호를 출력하고, 전체 런길이 인코더를 효과적으로 기능정지하게 한다.

(테이블 19)

8-비트 런길이 인코더 테이블			
컬러	입력	길이	런 끝
0	0000 0000	8	0
0	0000 0001	7	1
0	0000 001x	6	1
0	0000 01xx	5	1
0	0000 1xxx	4	1
0	0001 xxxx	3	1
0	001x xxxx	2	1
0	01xx xxxx	1	1
0	1xxx xxxx	0	1
1	1111 1111	8	0
1	1111 1110	7	1
1	1111 110x	6	1
1	1111 10xx	5	1
1	1111 0xxx	4	1
1	1110 xxxx	3	1
1	110x xxxx	2	1
1	10xx xxxx	1	1
1	0xxx xxxx	0	1

8-비트 런길이 인코더(192)의 출력은 남아있는 페이지 폭에 의해서 제한된다. 실제의 8-비트 런길이는 남아있는 페이지 폭으로부터 공제되며, 배럴 시프터(191)를 제어하고 바이트 스트림 입력을 클록킹하기 위해서 사용되는 모듈로-8 비트 위치에 부가된다(195).

외부 리셋 라인은 라인의 시작에서 런길이 인코더를 리셋하기 위해서 사용된다. 그것은 현재 컬러를 리셋하고 페이지 폭을 페이지 폭 레지스터내로 래칭한다. 외부 다음 런 라인은 런길이 인코더로부터 또 다른 런길이를 요청하기 위해서 사용된다. 그것은 현재 컬러를 인버팅하고, 런길이 레지스터 및 준비 레지스터를 리셋한다. 외부 플러시 라인은, 불완전하다면, 런의 마지막 바이트를 플러싱하기 위해서 사용된다. 그것은 바이트-정렬된 라인을 프로세싱하기 위해서 라인-라인 기초, 또는 바이트-정렬된 이미지를 프로세싱하기 위해서 이미지 기초상에서 사용될 수 있다.

외부 준비 라인은 런길이 인코더가 런길이를 인코딩할 준비가 되어 있고 현재 런길이가 런길이 라인상에서 이용 가능하다는 것을 나타낸다. 그것은 외부 로직을 기능정지하게 하는데 사용될 수 있다.

7.2.3 JPEG 디코더

도 23에 도시된 JPEG 디코더(143)는 JPEG-압축된 CMYK 연속톤 이미지를 압축해제한다.

JPEG 디코더로의 입력은 JPEG 비트스트림이다. JPEG 디코더로부터의 출력은 연속톤 CMYK 이미지 라인의 세트이다.

압축해제할 때, JPEG 디코더는 그 출력을 8×8 화소 블록의 형태로 기록한다. 때때로 이들은 코덱과 밀접하게 연결된 페이지 폭×8 스트림 버퍼를 통해서 완전-폭 라인으로 변환된다. 이것은 67KB 버퍼를 요구할 것이다. 대신에 본 발명은 도 23에 도시된 바와 같이, 공유 버스 액세스 및 8 대응 DMA 채널을 갖는 8 병렬 화소 FIFO를 사용한다.

7.2.4 하프토너/합성기

도 24에 도시된 하프토너/합성기 유닛(HCU; 141)은 연속톤 CMYK 층을 2-레벨 CMYK로 하프토닝하는 기능과 블랙 층을 하프토닝된 연속톤 층 위로 합성하는 기능을 조합한다.

HCU로의 입력은 확장된 267 ppi CMYK 연속톤 층(200), 및 확장된 1600 dpi 블랙 층(201)이다. HCU로부터의 출력은 1600 dpi 2-레벨 CMYK 이미지 라인의 세트(202)이다.

시작된 직후, HCU는 그것이 페이지 끝 조건을 검출할 때까지, 또는 그것이 그 제어 레지스터를 통해서 명시적으로 중지될 때까지 진행한다.

HCU는 지정된 폭 및 길이의 도트의 페이지를 발생시킨다. 폭 및 길이는 HCU를 시작시키기 이전에 페이지 폭 및 페이지 길이 레지스터에 기록되어야 한다. 페이지 폭은 프린트헤드의 폭(171)에 대응한다. 페이지 길이는 타겟 페이지의 길이에 대응한다.

HCU는 페이지 폭에 관한 지정된 좌측 및 우측 여백 사이의 타겟 페이지 데이터를 발생시킨다. 좌측 및 우측 여백의 위치는 HCU를 시작시키기 이전에 좌측 여백 및 우측 여백 레지스터에 기록되어야 한다. 좌측 여백으로부터 우측 여백으로까지의 거리는 타겟 페이지 폭에 대응한다.

HCU는 지정된 블랙(172) 및 연속톤(173) 페이지 폭에 따른 블랙 및 연속톤 데이터를 소비한다. 이들 페이지 폭은 HCU를 시작시키기 이전에 블랙 페이지 폭 및 연속톤 페이지 폭 레지스터에 기록되어야 한다. HCU는 블랙 및 연속톤 데이터를 타겟 페이지 폭(174)에 클리핑한다. 이것은 입력 FIFO 레벨에서 임의의 특별한 라인 끝 로직을 요구함이 없이 블랙 및 연속톤 페이지 폭이 타겟 페이지 폭을 초과하게 한다.

페이지 폭(171), 블랙(172) 및 연속톤(173) 페이지 폭, 및 여백 사이의 관계가 도 25에 예시된다.

HCU는 지정된 환산 계수에 수평 및 수직적으로 둘 다 기초한 프린터 해상도로 연속톤 데이터를 스케일링한다. 이러한 환산 계수는 HCU를 시작시키기 이전에 연속톤 환산 계수 레지스터에 기록되어야 한다.

(테이블 20)

하프토너/합성기 제어 및 구성 레지스터		
레지스터	폭	설명
시작	1	HCU를 시작
중지	1	HCU를 중지
페이지 폭	14	도트에서, 인쇄된 페이지의 페이지 폭. 이것은 각각의 라인에 대해서 발생되어야 하는 도트의 수이다.
좌측 여백	14	도트에서, 좌측 여백의 위치
우측 여백	14	도트에서, 우측 여백의 위치
페이지 길이	15	도트에서, 인쇄된 페이지의 페이지 길이. 이것은 각각의 페이지에 대해서 발생되어야 하는 라인의 수이다.
블랙 페이지 폭	14	도트에서, 블랙 층의 페이지 폭. 블랙 라인의 끝을 검출하기 위해서 사용된다.
연속톤 페이지 폭	14	도트에서, 연속톤 층의 페이지 폭. 연속톤 라인의 끝을 검출하기 위해서 사용된다.
연속톤 환산 계수	4	연속톤 데이터를 2-레벨 해상도로 스케일링하기 위해서 사용되는 환산 계수

HCU에 의해서 산출된 데이터의 소비자는 프린트헤드 인터페이스이다. 프린트헤드 인터페이스는 2-레벨 CMYK 이미지 데이터를 플레너 포맷으로 요구한다, 즉 컬러 평면이 분리된다. 더 나아가, 그것은 또한 짝수 및 홀수 화소가 분리되는 것을 요구한다. 따라서, HCU의 출력 단계는 짝수 시안, 홀수 시안, 짝수 마젠타, 홀수 마젠타, 짝수 옐로우, 홀수 옐로우, 짝수 블랙, 및 홀수 블랙에 대하여 각각 하나씩 8 병렬 화소 FIFO를 사용한다.

입력 연속톤 CMYK FIFO는 완전한 8KB 라인 버퍼이다. 라인은 라인 복제를 통해서 수직 업-스케일링을 이루기 위해 연속톤 환산 계수 횡수 사용된다. FIFO 기록 번지 래핑은 라인의 마지막 사용의 시작 때까지 디스에이블된다. 대안은 주 메모리로부터 라인을 연속톤 환산 계수 횡수 판독하는 것인데, 메모리 트래픽을 65MB/s 만큼 증가시키지만, 온-칩 8KB 라인 버퍼에 대한 필요를 피한다.

7.2.4.1 다중-임계값 디더

범용 256-층 디더 볼륨은 다른 강도 레벨을 감결함함으로써, 디더 셀 디자인에서 상당한 유연성을 제공한다. 범용 디더 볼륨은 클 수 있다 - 64×64×256 디더 볼륨은, 예를 들어, 128KB의 크기를 갖는다. 또한 그들은 각각의 컬러 구성요소가 볼륨으로부터 다른 비트의 검색을 요구하기 때문에 액세스하기에 비효율적이다. 실제로, 디더 볼륨의 각각의 층을 완전히 감결합할 필요는 없다. 볼륨의 각각의 도트 열은 256 별개의 비트보다는 임계값의 고정 세트로서 구현될 수 있다. 3개의 8-

비트 임계값을 사용하는 것은, 예를 들어, 24 비트를 소비하기만 한다. 이제, n 임계값은 대응 디터 셀 위치가 교대로 설정되지 않거나 설정되는 $n+1$ 강도 인터발을 정한다. 디터링되는 연속톤 화소 값은 $n+1$ 인터발 중 하나를 유일하게 선택하고, 이것은 대응 출력 도트의 값을 결정한다.

본 발명은 3중-임계값 $64 \times 64 \times 3 \times 8$ -비트(12KB) 디터 볼륨을 사용하여 연속톤 데이터를 디터링한다. 3개의 임계값은 하나의 사이클에서 디터 셀 ROM으로부터 검색될 수 있는 편리한 24-비트 값을 형성한다. 디터 셀 일치는 컬러 평면 사이에 소망되고, 그 때 동일 3중-임계값은 한번 검색되어 각각의 컬러 구성요소를 디터링하는데 사용될 수 있다. 디터 셀 일치가 소망되지 않는다면, 그 때 디터 셀은 4개의 서브-셀로 분할되어 4개의 개별적으로 번지 지정 가능한 ROM에 저장될 수 있는데, 그 ROM으로부터 4개의 다른 3중-임계값이 하나의 사이클에서 병렬로 검색될 수 있다. 도 26에 도시된 번지 지정안을 사용하여, 4개의 컬러 평면은 서로로부터 32 도트의 수직 및/또는 수평 오프셋에서 동일 디터 셀을 공유한다.

다중-임계값 디터(203)가 도 26에 도시된다. 3중-임계값 유닛(204)은 3중-임계값 및 강도 값을 인터발 및 일 또는 제로 비트로 변환한다. 3중-임계값 규칙이 테이블 21에 나타내어진다. 대응 로직(208)은 도 27에 도시된다.

도 26을 더 상세히 설명하면, 일반적으로 204에 나타내어진 4개의 별개의 3중 임계값 유닛은 각각 CMYK 신호의 각각의 컬러 구성요소에 대하여 일련의 연속톤 컬러 화소 값을 수신한다. 디터 볼륨은 일반적으로 205에 나타내어진 4개의 디터 서브셀(A, B, C, 및 D)로 분할된다. 디터 셀 번지 발생기(206) 및 일반적으로 207에 표시된 4개의 게이트는 각각의 다른 컬러에 대하여 하나의 사이클에서 병렬로 검색될 수 있는 4개의 다른 3중 임계값의 검색을 제어한다.

(테이블 21)

3중-임계 규칙	
인터발	출력
$V \leq T_1$	0
$T_1 < V \leq T_2$	1
$T_2 < V \leq T_3$	0
$T_3 < V$	1

7.2.4.2 합성

합성 유닛(205)은 블랙 층 도트를 하프토닝된 CMYK 층 도트 위로 합성한다. 블랙 층 불투명도가 일이라면, 그 때 하프토닝된 CMY는 제로로 세팅된다.

4-비트 하프토닝된 컬러 $C_c M_c Y_c K_c$ 및 1-비트 블랙 층 불투명도 K_b 가 주어질 때, 합성 및 클립 로직은 테이블 22에 정의된 바와 같다.

(테이블 22)

합성 로직	
컬러 채널	조건
C	$C_c \wedge \neg K_b$
M	$M_c \wedge \neg K_b$
Y	$Y_c \wedge \neg K_b$
K	$K_c \vee K_b$

7.2.4.3 클록 이네이블 발생기

클록 이네이블 발생기(206)는 연속톤 CMYK 화소 입력, 블랙 도트 입력, 및 CMYK 도트 출력을 클록킹하기 위한 이네이블 신호를 발생시킨다.

일찍이 설명된 바와 같이, 연속톤 화소 입력 버퍼는 라인 버퍼 및 FIFO 둘 다로서 사용된다. 각각의 라인은 한번 판독되고 그 후 연속톤 환산 계수 횡수 사용된다. FIFO 기록 번지 래핑은 라인의 최종 복제된 사용의 시작 때까지 디스에이블되는 데, 그 시간에 클록 이네이블 발생기는 래핑을 이네이블하는 연속톤 라인 전진 이네이블 신호를 발생시킨다.

또한 클록 이네이블 발생기는 출력 도트 FIFO의 짝수 또는 홀수 세트를 선택하기 위해서 사용되는 짝수 신호, 및 현재 도트 위치가 페이지의 좌측 또는 우측 여백내에 있을 때 화이트 도트를 발생시키기 위해서 사용되는 여백 신호를 발생시킨다.

클록 이네이블 발생기는 카운터의 세트를 사용한다. 카운터의 내부 로직은 테이블 23에 정의된다. 클록 이네이블 신호의 로직은 테이블 24에 정의된다.

(테이블 23)

클록 이네이블 발생기 카운터 로직					
카운터	약자	폭	데이터	로드 조건	감분 조건
도트	D	14	페이지 폭	$RP^a \vee EOL^b$	$(D > 0) \wedge clk$
라인	L	15	페이지 길이	RP	$(L > 0) \wedge EOL$
좌측 여백	LM	14	좌측 여백	$RP \vee EOL$	$(LM > 0) \wedge clk$
우측 여백	RM	14	우측 여백	$RP \vee EOL$	$(RM > 0) \wedge clk$
짝수/홀수 도트	E	1	0	$RP \vee EOL$	clk
블랙 도트	BD	14	블랙 폭	$RP \vee EOL$	$(LM=0) \wedge (BD > 0) \wedge clk$
연속톤 도트	CD	14	연속톤 폭	$RP \vee EOL$	$(LM=0) \wedge (CD > 0) \wedge clk$
연속톤 서브-화소	CSP	4	연속톤 환산 계수	$RP \vee EOL \vee (CSP=0)$	$(LM=0) \wedge clk$
연속톤 서브-라인	CSL	4	연속톤 환산 계수	$RP \vee (CSL=0)$	$EOL \wedge clk$

^a RP(리셋 페이지) 조건: 외부 신호
^b EOL(라인 끝) 조건: $(D=0) \wedge (BD=0) \wedge (CD=0)$

(테이블 24)

클록 이네이블 발생기 출력 신호 로직	
출력 신호	조건
출력 도트 클록 이네이블	$(D > 0) \wedge \neg EOP^a$
블랙 도트 클록 이네이블	$(LM=0) \wedge (BD > 0) \wedge \neg EOP$
연속톤 화소 클록 이네이블	$(LM=0) \wedge (CD > 0) \wedge (CSP=0) \wedge \neg EOP$
연속톤 라인 전진 이네이블	$(CSL=0) \wedge \neg EOP$
짝수	$E=0$
여백	$(LM=0) \vee (RM=0)$

^a EOP(페이지 끝) 조건: $L=0$

7.3 프린트헤드 인터페이스

프린트헤드 인터페이스(PHI; 142)는 프로세서가 맴젯 프린트헤드를 인쇄될 도트로써 로딩하는 수단이고, 실제 도트 인쇄 프로세스를 제어한다. PHI는:

- 일정 인쇄 라인에 대한 도트를 로컬 버퍼 스토리지내로 로딩하고 그들을 맴젯 프린트헤드에 대해 요구되는 오더로 포맷팅하는 라인 로더/포맷 유닛(LLFU; 209).
- 데이터를 맴젯 프린트헤드(63)로 전송하고, 인쇄 동안 노즐 파이어링 시퀀스를 제어하는 맴젯 인터페이스(MJI; 210)를 포함한다.

PHI내의 유닛은 프로세서(139)에 의해서 프로그래밍된 많은 레지스터에 의해서 제어된다. 더하여, 프로세서는 메모리로부터 LLFU로의 전송을 위해 DMA 제어기(144)에서 적합한 매개 변수를 세팅 업하는 것을 책임진다. 이것은 페이지가 깨끗한 에지를 갖기 위해서 페이지의 시작 및 끝 동안 화이트(전부 0)를 적합한 컬러내로 로딩하는 것을 포함한다. 프린트헤드 인터페이스(142)의 내부 구조가 도 28에 도시된다.

7.3.1 라인 로더/포맷 유닛

라인 로더/포맷 유닛(LLFU; 209)은 일정 인쇄 라인에 대하여 도트를 로컬 버퍼 스토리지내로 로딩하고 그들을 뿔렛 프린트헤드에 대해 요구되는 오더로 포맷팅한다. 그것은 페이지의 최후의 인쇄를 위해서 사전-계산된 노즐이네이블 비트를 뿔렛 인터페이스로 공급하는 것을 책임진다.

8-인치 프린트헤드에서의 단일 라인은 12,800개의 4-컬러 도트로 구성된다. 컬러 당 1 비트에서, 단일 인쇄 라인은 51,200 비트로 구성된다. 이들 비트는 프린트헤드상에 보내지기 위해서 정확한 순서대로 공급되어야 한다. 로드 사이클 도트 로딩 오더에 관련한 더 많은 정보에 대해서는 섹션 6.1.2.1을 보라, 그러나 요약해서, 32 비트는 2개의 4-인치 프린트헤드의 각각에 한번에 전송되고, 32 비트는 8 세그먼트의 각각에 대하여 4 도트를 표현한다.

인쇄는 도트-비트 정보를 준비 및 액세스하기 위해서 2중 버퍼링 방법을 사용한다. 하나의 라인이 제 1 버퍼(213)내로 로딩되고 있는 동안, 제 2 버퍼(214)에서의 사전-로딩된 라인은 뿔렛 도트 오더에서 판독되고 있다. 전체 라인이 뿔렛 인터페이스를 통해서 제 2 버퍼(214)로부터 프린트헤드로 전송된 직후, 판독 및 기록 프로세스가 버퍼를 교환한다. 이제 제 1 버퍼(213)는 판독되고 제 2 버퍼는 데이터의 새로운 라인으로써 로딩 업된다. 이것은 도 29의 개념적 개관에서 보여질 수 있는 바와 같이, 인쇄 프로세스 동안에 걸쳐 반복된다.

LLFU의 실제 구현이 도 30에 도시된다. 다른 것에 기록되고 있는 동안 하나의 버퍼로부터 판독되고 있기 때문에, 번지 라인의 2개의 세트가 사용되어야 한다. 공통 데이터 버스로부터의 32-비트 데이터인은, DMA 긍정응답에 응답하여 상태기에 의해서 발생하는 기록 이네이블에 의존해서 로딩된다.

멀티플렉서(215)는 버퍼 0(213) 및 버퍼 1(214)의 2개의 4-비트 출력 사이에서 선택하고 그 결과를 4-비트 시프트 레지스터(216)에 의해서 8-엔트리에 보낸다. 처음 8 판독이 사이클링한 후에, 그리고 전진 펄스가 MJJ로부터 올 때마다, 시프트 레지스터로부터의 현재 32-비트 값은 32-비트 전송 레지스터(217)내로 게이팅되는데, 거기서 그것은 MJJ에 의해서 사용될 수 있다.

7.3.1.1 버퍼

2개의 버퍼(213 및 214)의 각각은 컬러 당 1개씩 4개의 서브-버퍼(220, 221, 222 및 223)로 쪼개진다. 도 31에 도시된 바와 같이, 모든 짝수 도트는 각각의 컬러의 버퍼에 홀수 도트 전에 놓인다.

인쇄될 다음 라인에서 도트를 표현하는 51,200 비트는 컬러 버퍼 당 12,800 비트, 400개의 32-비트 워드로서 저장된다. 첫번째의 200개의 32-비트 워드(6400 비트)는 컬러에 대하여 짝수 도트를 표현하고, 반면에 두번째의 200개의 32-비트 워드(6400 비트)는 컬러에 대하여 홀수 도트를 표현한다.

번지 지정 디코딩 회로설계는 일정 사이클에서 단일 32-비트 액세스가 모든 4 서브-버퍼로 될 수 있는 그러한 것이다 - 모든 4로부터 판독 또는 4 중 하나에 기록. 각각의 컬러 버퍼로부터 판독된 32-비트 중 하나의 비트만이 총 4 출력 비트에 대해서 선택된다. 프로세스가 도 32에 도시된다. 번지의 13 비트는 32 비트를 선택하기 위해 사용되고 있는 번지의 8-비트에 의해서 특별한 비트의 판독을 허용하고, 번지의 5-비트는 그들 32로부터 1-비트를 선택한다. 모든 컬러 버퍼는 이 로직을 공유하기 때문에, 단일 13-비트 번지는 컬러 당 하나씩 총 4 비트를 내놓는다. 각각의 버퍼는 단일 32-비트 값이 일정 사이클에서 특별한 컬러 버퍼에 기록되게 하기 위해서 그 자신의 기록이네이블 라인을 갖는다. 하나의 버퍼만이 실제로 데이터를 클럭 인 할것이기 때문에, 데이터인의 32-비트는 공유된다.

7.3.1.2 번지 발생

7.3.1.2.1 판독

관독을 위한 번지 발생은 간단하다. 각각의 사이클에서 본 발명은 특별한 세그먼트에 대해서 컬러 당 1-비트를 표현하는 4 비트를 인출하기 위해서 사용되는 비트 번지를 발생시킨다. 400을 현재 비트 번지에 부가함으로써, 본 발명은 다음 세그먼트의 등가 도트로 전진한다. 홀수 및 짝수 도트가 버퍼에서 분리되기 때문에 본 발명은 (800이 아니라) 400을 부가한다. 본 발명은 이것을 짝수 도트를 표현하는 8 세그먼트의 2개의 세트에 대해서 32 비트의 2개의 세트를 검색하기 위해서 16 번 행하고(결과적 데이터는 한번에 32 비트 MJ에 전송된다) 홀수 도트를 로딩하기 위해서 또 다른 16번 행한다. 이러한 32-사이클 프로세스는 400번 반복되고, 매번 시작 번지를 증분시킨다. 따라서 400×32 사이클에서, 총 400×32×4 (51,200) 도트 값이 프린트헤드에 의해서 요구되는 순서대로 전송된다.

더하여, 본 발명은 전송기록이네이블 제어 신호를 발생시킨다. LLFU는 MJ전제 시작하기 때문에, 본 발명은 MJ로부터의 전진 펄스 전에 제 1 값을 전송해야 한다. 또한 본 발명은 제 1 전진 펄스에 대해서 준비를 갖추고 다음 32-비트 값을 발생 시켜야 한다. 해법은 제 1 32-비트 값을 8 사이클 후에 전송 레지스터로 전송하고, 그 후 전진 펄스가 다음 8-사이클 그룹을 시작시키는 것을 대기 하여 나중에 8-사이클을 기능정지하게 하는 것이다. 제 1 전진 펄스가 도착한 직후, LLFU는 MJ에 동기화된다. 그러나, MJ는 초기 전송 값이 유효하고 다음 32-비트 값이 전송 레지스터내로 로딩될 준비가 되어 있기 위해서 적어도 LLFU 16 사이클 후에 시작되어야 한다.

관독 프로세스가 이하 의사코드에 나타내어진다:

DotCount = 0

For DotInSegment0 = 0 to 400

CurrAdr = DotInSegment0

Do

V1 = (CurrAdr = 0) OR (CurrAdr = 3200)

V2 = Low 3 bits of DotCount = 0

TransferWriteEnable = V1 OR ADVANCE

Stall = V2 AND (NOT TransferWriteEnable)

If (NOT Stall)

Shift Register = Fetch 4-bits from CurrReadBuffer: CurrAdr

CurrAdr = CurrAdr + 400

DotCount = (DotCount + 1) MOD 32 (odd & even, printheads 1&2, segments 0-7)

EndIf

Until (DotCount = 0) AND (NOT Stall)

EndFor

라인이 완료된 직후, CurrReadBuffer 값이 프로세서에 의해서 토글링되어야 한다.

7.3.1.2.2 기록

기록 프로세스 또한 간단하다. 4 DMA 요청 라인이 DMA 제어기에 출력된다. 요청이 반환 DMA 긍정응답 라인에 의해서 만족될 때, 적합한 8-비트 수신지 번지가 선택되고(13-비트 출력 번지의 하부 5 비트는 무시 값이다) 긍정응답 신호가 옴

은 버퍼의 기록이네이블 제어 라인에 넘겨진다(현재 기록 버퍼는 현재판독버퍼이다). 8-비트 수신지 번지가 컬러 당 하나의 번지씩 4 현재 번지로부터 선택된다. DMA 요청이 만족될 때 적합한 수신지 번지는 증분되고, 대응하는 트랜스퍼리메이닝(TransfersRemaining) 카운터는 감분된다. DMA 요청 라인은 그 컬러에 대하여 남아있는 전송의 수가 제로가 아닐 때 세팅되기만 한다.

이하 의사코드가 기록 프로세스를 예시한다:

CurrentAdr[0-3] = 0

While (TransfersRemaining[0-3] are all non-zero)

DMARequest[0-3] = TransfersRemaining[0-3] != 0

If DMAAcknowledge[N]

CurrWriteBuffer:CurrentAdr[N] = Fetch 32-bits from data bus

CurrentAdr[N] = CurrentAdr[N] + 1

TransfersRemaining[N] = TransfersRemaining[N] - 1 (floor 0)

EndIf

EndWhile

7.3.1.3 레지스터

LLFU에 이하 레지스터가 포함된다:

(테이블 25)

라인 로드/포맷 유닛 레지스터	
레지스터 이름	설명
현재판독버퍼	현재 버퍼로부터 판독된다. 버퍼0으로부터 판독되고 있을 때 버퍼1에 기록되고, 그 역으로. MJ1로부터 각각의 전진 라인 펄스로써 토글링되어야 한다.
고	비트 0 및 1이 각각 판독 및 기록 프로세스의 시작을 제어한다. 적합한 비트로의 제로-아닌 기록은 프로세스를 시작시킨다.
스톱	비트 0 및 1이 각각 판독 및 기록 프로세스의 종지를 제어한다. 적합한 비트로의 제로-아닌 기록은 프로세스를 중지한다.
트랜스퍼리메이닝C	시안 버퍼내로 판독될 남아있는 32-비트 전송의 수
트랜스퍼리메이닝M	마젠타 버퍼내로 판독될 남아있는 32-비트 전송의 수
트랜스퍼리메이닝Y	옐로우 버퍼내로 판독될 남아있는 32-비트 전송의 수
트랜스퍼리메이닝K	블랙 버퍼내로 판독될 남아있는 32-비트 전송의 수

7.3.2 램젯 인터페이스

램젯 인터페이스(MJI; 211)는 데이터를 램젯 프런트헤드(63)로 전송하고, 인쇄 동안 노즐 파이어링 시퀀스를 제어한다.

MJI는 섹션 6.1.2에 설명된 파이어링 순서 및 프런트헤드 로딩을 따르는 상태기일 뿐이고(도 28 참조), 섹션 6.1.4 및 섹션 6.1.5에 설명된 바와 같은 예열 사이클 및 클리닝 사이클의 기능성을 포함한다. 고속 및 저속 인쇄 모드가 둘 다 이용 가능하다. 각각의 컬러에 대한 도트 카운트는 또한 MJ1에 의해서 유지된다.

MJI는 2 데이터 소스의 선택으로부터 프런트헤드내로 데이터를 로딩한다:

전부 1. 이것은 모든 노즐이 후속 인쇄 사이클 동안 파이어링할 것임을 의미하고, 예열 또는 클리닝 사이클에 대하여 프린트헤드를 로딩하기 위한 표준 메카니즘이다.

LLFU의 전송 레지스터에 유지된 32-비트 입력으로부터이다. 이것은 이미지를 인쇄하는 표준 수단이다. LLFU로부터의 32-비트 값은 프린트헤드에 직접 보내지고 1-비트 '전진' 제어 펄스는 LLFU에 보내진다. 각각의 라인의 끝에서, 1-비트 '전진라인' 펄스가 또한 이용 가능하다.

MJI는 LLFU가 제 1 32-비트 전송 값을 이미 준비한 후에 시작되어야 한다. 이것은 그러해서 32-비트 데이터 입력은 프린트헤드로의 제 1 전송에 대해서 유효할 것이다.

따라서 MJI는 LLFU 및 외부 메모리 프린트헤드에 직접 접속된다.

7.3.2.1 프린트헤드로의 접속부

MJI(211)는 MJI에 관하여 입력 및 출력의 센스로써, 프린트헤드(63)로의 이하 접속부를 갖는다. 이름은 프린트헤드상의 핀 접속부와 매칭한다(8-인치 프린트헤드가 배선되는 방식의 설명에 대한 섹션 6.2.1 참조).

(테이블 26)

메모리 인터페이스 접속부			
이름	#핀	I/O	설명
크로마패드셀렉트	3	0	어느 크로마패드가 파이어링할지를 선택(0-4)
노즐셀렉트	4	0	패드로부터 어느 노즐이 파이어링할지를 선택(0-9)
패드그룹이네이블	2	0	패드그룹을 파이어링하게 함(초이스:01, 10, 11)
AOI네이블	1	0	패드그룹 A에 대하여 펄스를 파이어링
BOI네이블	1	0	패드그룹 B에 대하여 펄스를 파이어링
C데이터인[0-7]	8	0	세그먼트 0-7의 시안 시프트 레지스터로 시안 출력
M데이터인[0-7]	8	0	세그먼트 0-7의 마젠타 시프트 레지스터로 마젠타 입력
Y데이터인[0-7]	8	0	세그먼트 0-7의 옐로우 시프트 레지스터로 옐로우 입력
K데이터인[0-7]	8	0	세그먼트 0-7의 블랙 시프트 레지스터로 블랙 입력
SR클록1	1	0	SR클록1(시프트레지스터클록1)상의 펄스가 C데이터인[0-7], M데이터인[0-7], Y데이터인[0-7] 및 K데이터인[0-7]으로부터의 현재값을 4-인치 프린트헤드1의 32 시프트 레지스터내로 로딩한다.
SR클록2	1	0	SR클록2(시프트레지스터클록2)상의 펄스가 C데이터인[0-7], M데이터인[0-7], Y데이터인[0-7] 및 K데이터인[0-7]으로부터의 현재값을 4-인치 프린트헤드2의 32 시프트 레지스터내로 로딩한다.
P트랜스퍼	1	0	시프트 레지스터로부터 프린트헤드의 내부 노즐이네이블 비트(노즐 당 하나)로 데이터의 병렬 전송
센스세그셀렉트1	1	0	C데이터인[n]상의 데이터와 AND된 센스세그이네이블1상의 펄스가 4-인치 프린트헤드1에서의 세그먼트 n에 대하여 센스 라인을 이네이블한다.
센스세그셀렉트2	1	0	C데이터인[n]상의 데이터와 AND된 센스세그이네이블2상의 펄스가 4-인치 프린트헤드2에서의 세그먼트 n에 대하여 센스 라인을 이네이블한다.
T센스	1	I	온도 감지
V센스	1	I	전압 감지
R센스	1	I	저항률 감지
W센스	1	I	폭 감지
총	52		

7.3.2.2 파이어링 펄스 지속시간

A이네이블 및 B이네이블 라인상의 파이어링 펄스 지속시간은 (온도 및 잉크 특성에 의존하는) 잉크의 점성 및 프린트헤드에 이용될 수 있는 전력량에 의존한다. 전형적인 펄스 지속시간 범위는 1.3 내지 1.8 μ s이다. 따라서 MJI는 프린트헤드로부터의 피드백에 의해서 인덱싱된, 프로그래밍 펄스 지속시간 테이블(230)을 포함한다. 펄스 지속시간의 테이블은 더 낮은 비용의 전원 공급장치의 사용을 허용하고, 더 정확한 방울 분출을 유지하는데 도움이 된다.

펄스 지속시간 테이블은 256 엔트리를 갖고, 현재 V센스(231) 및 T센스(232) 세팅에 의해서 인덱싱된다. 번지의 상위 4-비트는 V센스로부터 오고, 번지의 하위 4-비트는 T센스로부터 온다. 각각의 엔트리는 8-비트이고 0-4 μ s의 범위에서 고정 포인트 값을 표현한다. A이네이블 및 B이네이블 라인을 발생시키는 프로세스가 도 33에 도시된다. 아날로그 V센스(231) 및 T센스(232) 신호는 각각의 샘플 및 유지 회로(233 및 234)에 의해서 수신되고, 그 후 펄스 지속시간 테이블(230)에 인가되기 전에 각각의 변환기(235 및 236)에서 디지털 워드로 변환된다. 펄스 지속시간 테이블(230)의 출력은 파이어링 펄스를 발생시키기 위해서 펄스 폭 발생기(237)에 인가된다.

256-바이트 테이블은 제 1 페이지를 인쇄하기 전에 CPU에 의해서 기록된다. 테이블은 소망된다면 페이지 사이에서 업데이트될 수 있다. 테이블에서의 각각의 8-비트 펄스 지속시간 엔트리는:

- (페이지 기술로부터) 사용자 휘도 세팅
- (QA 칩으로부터) 잉크의 점성 커브
- R센스
- W센스
- V센스
- T센스를 조합한다.

7.3.2.3 도트 카운트

MJI(211)는 도트 카운트 레지스터(240)에서 프린트헤드로부터 파이어링된 각각의 컬러의 도트의 수의 카운트를 유지한다. 각각의 컬러에 대한 도트 카운트는 프로세서 제어 하에, 신호(241)에 의해서, 개별적으로 클리어링된, 32-비트 값이다. 전형적인 사용예에서, 도트 카운트는 판독되고 각각의 페이지 후에 클리어링될 것이기는 하지만, 32-비트 길이에서 각각의 도트 카운트는 17 12-인치 페이지의 최대 커버리지 도트 카운트를 유지할 수 있다.

도트 카운트는 잉크 카트리지의 잉크가 바닥날 때를 예측하기 위해서 QA 칩(85)을 업데이트하는 프로세서에 의해서 사용된다(섹션 7.5.4.1). 프로세서는 QA 칩으로부터 C, M, Y, 및 K의 각각에 대하여 카트리지에서의 잉크의 볼륨을 안다. 방울의 수를 카운팅하는 것은 잉크 센서에 대한 필요를 제거하고, 잉크 채널이 마른 상태가 되는 것을 방지한다. 업데이트된 방울 카운트는 각각의 페이지 후에 QA 칩에 기록된다. 새로운 페이지는 충분한 잉크가 남아있지 않다면 인쇄되지 않을 것이고, 사용자가 재인쇄되어야 하는 불발 하프-인쇄된 페이지를 얻음이 없이 잉크를 교환하게 한다.

시안에 대한 도트 카운터의 레이아웃이 도 34에 도시된다. 남아있는 3 도트 카운터(마젠타, 옐로우, 및 블랙 각각에 대한 M도트카운트, Y도트카운트, 및 K도트카운트)는 구조에서 동일하다.

7.3.2.4 레지스터

프로세서(139)는 레지스터 세트를 통해서 MJI(211)와 통신한다. 레지스터는 프로세서가 인쇄 프로세스에 관해 피드백을 수신할 뿐만 아니라 인쇄를 매개 변수화하게 한다.

다음의 레지스터가 MJI에 포함된다:

(테이블 27)

레지스터 이름	설명
인쇄 매개 변수	
넘트랜스퍼 (NumTransfers)	프린트헤드를 로딩하기 위해 요구되는 전송 수(통상 1600). 이것은 일정 라인에 대해 전송하기 위한 32-비트 데이터 값의 총 수 및 양 SR클록 라인에 대한 펄스의 수이다.
인쇄속도	저속 또는 고속에서 인쇄 (인쇄 동안 파드그룹이네이블 라인상에서 값을 결정)
넘라인(NumLines)	수행하기위한 로드/인쇄 사이클의 수
프린트를 모니터링	
상태	멤젯 인터페이스의 상태 레지스터
라인리메이닝 (LinesRemaining)	남아있는 인쇄될 라인의 수. 고=1 동안 유효할 뿐. 시작 값은 넘라인이다.
트랜스퍼리메이닝	프린트헤드가 현재 라인에 대해 로딩되도록 고려되기 전에 남아있는 전송의 수. 고=1 동안 유효할 뿐.
센스세그먼트	후속 피드백 센스세그먼트 펄스 동안 시안 데이터 라인상에 놓기 위한 8-비트 값. 8 세그먼트 중 하나에 대응하는, 8 비트 중 1만이 세팅되어야 한다. 2개의 4-인치 프린트헤드 중 어느 것을 센스할지 어떻게 결정하는가에 대해서는 센스세그먼트를 참조.
세트올노즐 (SetAllNozzles)	제로가 아니라면, 모든 노즐이 후속 인쇄도트 프로세스 동안 파이어링되도록, 로드도트 프로세스 동안 프린트헤드에 기록된 32-비트 값은 모두 1s이다. 이것 은 예열 및 클리닝 사이클 동안 사용된다. 00이라면, 프린트헤드에 기록된 32-비트 값은 LLFU로부터 온다. 이것은 레귤러 이미지의 실제 인쇄 동안 경우이다.
조치	
리셋	이 레지스터로의 기록은 MJ1을 리셋하고, 임의의 로딩 또는 인쇄 프로세스를 중지하고, 모든 레지스터를 0으로써 로딩한다.
센스세그먼트	임의의 값을 갖는 이 레지스터로의 기록은 상태 레지스터의 피드백밸리드 비트를 클리어링하고, 로-오더 비트에 의존해서, 로딩도트 및 인쇄도트 상태 비트가 모두 00이라면 센스이네이블1 또는 센스이네이블2 라인상에 펄스를 보낸다. 임의의 상태 비트가 세팅된다면, 피드백 비트는 클리어링되고 더 이상 아무것도 되지 않는다. 다양한 센스 라인이 테스트된 직후, 값은 T센스, V센스, R센스, 및 W센스 레지스터에 놓이고, 그 후 상태 레지스터 의 피드백 비트가 세팅된다.
고	이 비트로의 1의 기록은 로드도트/인쇄도트 사이클을 시작시킨다. 총 넘라인 라인은 인쇄되고, 각각은 넘트랜스퍼 32 비트 전송을 포함한다. 각각의 라인이 인쇄 될 때, 라인리메이닝이 감분되고, 트랜스퍼리메이닝이 넘트랜스퍼로써 다시 재로딩된다. 상태 레지스터는 인쇄 상태 정보를 포함 한다. 넘라인의 완료시, 로딩/인쇄 프로세스는 중지하고 고 비트는 클리어링된다. 최종 인쇄 사이클 동안, 프린트헤드내로 아무것도 로딩되지 않는다. 이 비트로의 0의 기록은 인쇄 프로세스를 중지하지 만 임의의 다른 레지스터를 클리어링하지 않는다.
클리어카운트	이 레지스터로의 기록은 비트 0, 1, 2 또는 3이 각각 세팅된다면 C도트카운트, M도트카운트, Y도트카운트, 및 K도트카운트 레지스터를 클리어링한다. 결과적으로 0의 기록은 어떠한 영향도 미치지 않는다.

피드백	
T센스	세그먼트 센스세그먼트에 보내진 마지막 센스세그먼트 펄스로부터 T센스의 피드백만을 판독. 상태 레지스터의 피드백밸리드 비트가 세팅된다면 유효할 뿐.
V센스	세그먼트 센스세그먼트에 보내진 마지막 센스세그먼트 펄스로부터 V센스의 피드백만을 판독. 상태 레지스터의 피드백밸리드 비트가 세팅된다면 유효할 뿐.
R센스	세그먼트 센스세그먼트에 보내진 마지막 센스세그먼트 펄스로부터 R센스의 피드백만을 판독. 상태 레지스터의 피드백밸리드 비트가 세팅된다면 유효할 뿐.
W센스	세그먼트 센스세그먼트에 보내진 마지막 센스세그먼트 펄스로부터 W센스의 피드백만을 판독. 상태 레지스터의 피드백밸리드 비트가 세팅된다면 유효할 뿐.
C도트카운트	프린트헤드에 보내진 시안 도트의 32-비트 카운트만을 판독
M도트카운트	프린트헤드에 보내진 마젠타 도트의 32-비트 카운트만을 판독

Y도트카운트	프린트헤드에 보내진 옐로우 도트의 32-비트 카운트만을 판독
K도트카운트	프린트헤드에 보내진 블랙 도트의 32-비트 카운트만을 판독

MJI의 상태 레지스터는 다음과 같은 비트 해석을 갖는 16-비트 레지스터이다.

(테이블 28)

MJI 상태 레지스터		
이름	비트	설명
로딩도트	1	세팅한다면, MJI는 도트를 현재 로딩하고 있고, 트랜스퍼리메이닝에서 남아있는 전송될 도트의 수를 갖는다. 클리어한다면, MJI는 도트를 현재 로딩하고 있지 않다.
인쇄도트	1	세팅한다면, MJI는 도트를 현재 인쇄하고 있다. 클리어한다면, MJI는 도트를 현재 인쇄하고 있지 않다.
인쇄A	1	이 비트는 A이네이블 라인상에 펄스가 존재하는 동안 세팅된다.
인쇄B	1	이 비트는 B이네이블 라인상에 펄스가 존재하는 동안 세팅된다.
피드백밸리드	1	이 비트는 피드백 값 T센스, V센스, R센스, 및 W센스가 유효한 동안 세팅된다.
예치(Reserved)	3	-
인쇄크로마파드	4	이것은 인쇄도트 상태 비트가 세팅된 동안 파이어링되는 현재 크로마파드를 유지한다.
인쇄노즐	4	이것은 인쇄도트 상태 비트가 세팅된 동안 파이어링되는 현재 노즐을 유지한다.

7.3.2.5 예열 및 클리닝 사이클

클리닝 및 예열 사이클은 적합한 레지스터를 세팅함으로써 간단히 성취될 수 있다:

- 세트올노즐 = 1
- 펄스지속시간 레지스터를 (예열 모드의 경우에서의) 낮은 지속시간 또는 클리닝 모드에 대한 적합한 방울 분출 지속시간으로 세팅
- 넘라인을 노즐이 파이어링되어야 하는 횟수로 세팅
- 고 비트를 세팅하고 그 후 인쇄 사이클이 완료되었을 때 클리어링될 고 비트를 대기.

7.4 프로세서 및 메모리

7.4.1 프로세서

프로세서(139)는 페이지 수신, 확장 및 인쇄 동안 다른 기능 유닛을 동기화시키는 제어 프로그램을 러닝한다. 또한 다양한 외부 인터페이스에 대한 디바이스 드라이버를 러닝하고, 사용자 인터페이스를 통해서 사용자 조치에 응답한다.

그것은, 효율적 DMA 관리를 제공하기 위해서, 낮은 인터럽트 대기시간을 가져야 하지만, 그렇지 않다면 특별히 고성능 DMA 제어기일 필요는 없다.

DMA 제어기는 27 채널상에서 단일-번지 전송을 지원한다(테이블 29 참조). 그것은 전송 완료시 프로세서로의 벡터링된 인터럽트를 발생시킨다.

(테이블 29)

DMA 채널 사용예		
기능 유닛	입력 채널	출력 채널
USB 인터페이스	-	1

EDRL 확장기	1	1
JPEG 디코더	1	8
하프토너/합성기	2	8
스피커 인터페이스	1	-
프린트헤드 인터페이스	4	-
	8	19
		27

7.4.3 프로그램 ROM

프로그램 ROM은 시스템 부팅 동안 주 메모리내로 로딩되는 ICP 제어 프로그램을 유지한다.

7.4.4 램버스 인터페이스

램버스 인터페이스는 외부 8MB (64Mbit) 램버스 DRAM (RDRAM)으로의 고속 인터페이스를 제공한다.

7.5 외부 인터페이스

7.5.1 USB 인터페이스

범용 직렬 버스(USB) 인터페이스는 표준 USB 디바이스 인터페이스를 제공한다.

7.5.2 스피커 인터페이스

스피커 인터페이스(250)(도 35)는 주 메모리로부터 사운드 클립의 DMA-중재된 전송을 위해 사용되는 작은 FIFO(251), 각각의 8-비트 샘플 값을 전압으로 변환하는 8-비트 디지털-아날로그 변환기(DAC; 252), 및 외부 스피커에 공급하는 증폭기(253)를 포함한다. FIFO가 비어 있을 때, 그것은 제로 값을 출력한다.

스피커 인터페이스는 사운드 클립의 주파수에서 클로킹된다.

프로세서는 스피커 인터페이스의 DMA 채널을 프로그래밍하기만 함으로써 사운드 클립을 스피커에 출력한다.

7.5.3 병렬 인터페이스

병렬 인터페이스(231)는 많은 병렬 외부 신호 라인상에서 I/O를 제공한다. 이것은 프로세서가 테이블 30에 리스팅된 디바이스를 감지 또는 제어하게 한다.

(테이블 30)

병렬 인터페이스 디바이스
병렬 인터페이스 디바이스
파워 버튼
용지 공급 버튼
파워 LED
용지 없음 LED
잉크 부족 LED
매체 센서
용지 이송 스테퍼 모터

7.5.4 직렬 인터페이스

직렬 인터페이스(232)는 2개의 표준 저속 직렬 포트를 제공한다.

하나의 포트는 마스터 QA 칩(85)에 접속하기 위해서 사용된다. 다른 하나는 잉크 카트리지를(233)에서 QA 칩에 접속하기 위해서 사용된다. 2개 사이의 프로세서-중재된 프로토콜은 잉크 카트리지를 인증하기 위해서 사용된다. 그 때 프로세서는 각각의 잉크의 남아있는 볼륨 뿐만 아니라, QA 칩으로부터 잉크 특성을 검색할 수 있다. 프로세서는 멤젯 프린트헤드를 적당히 컨피규어링하기 위해서 잉크 특성을 사용한다. 그것은 프린트헤드가 마른 상태에 의해서 손상되게 하지 않는 것을 보장하기 위해서, 프린트헤드 인터페이스에 의해 축적된 잉크 소비 정보로써 페이지-페이지 기초에서 업데이트된, 남아있는 잉크 볼륨을 사용한다.

7.5.4.1 잉크 카트리지 QA 칩

잉크 카트리지에서 QA 칩(233)은 최상의 가능한 인쇄 품질을 유지하기 위해서 요구되는 정보를 포함하고, 인증 칩을 사용하여 구현된다. 인증 칩에서의 데이터의 256 비트는 다음과 같이 할당된다:

(테이블 31)

잉크 카트리지의 256 비트(16-비트의 16 엔트리)			
M[n]	엑세스	폭	설명
0	R0 ^a	16	기초 헤더, 플래그 등
1	R0	16	직렬 넘버
2	R0	16	배치 넘버
3	R0	16	미래 확장을 위해 예치. 0이어야 한다.
4	R0	16	시안 잉크 특성
5	R0	16	마젠타 잉크 특성
6	R0	16	옐로우 잉크 특성
7	R0	16	블랙 잉크 특성
8-9	D0 ^b	32	나노리터에서, 남아있는 시안 잉크
10-11	D0	32	나노리터에서, 남아있는 마젠타 잉크
12-13	D0	32	나노리터에서, 남아있는 옐로우 잉크
14-15	D0	32	나노리터에서, 남아있는 블랙 잉크
^a 판독 전용(R0)			
^b 감분 전용(D0)			

각각의 페이지가 인쇄되기 전에, 프로세서는 전체의 최악의 경우도 고려한 페이지에 대하여 충분한 것을 보장하기 위해서 남아있는 잉크의 양을 체크하여야 한다. 페이지가 인쇄된 직후, 프로세서는 (프린트헤드 인터페이스로부터 얻어진) 각각의 컬러의 방울의 총 수를 방울 볼륨만큼 곱한다. 인쇄된 잉크의 양은 남아있는 잉크의 양으로부터 공제된다. 남아있는 잉크에 대한 측정의 단위는 나노리터이고, 그래서 32 비트는 잉크의 4리터 이상 표현할 수 있다. 페이지에 대해서 사용된 잉크의 양은 가장 가까운 나노리터(즉, 대략 1000 인쇄된 도트)로 나머지를림되어야 한다.

7.5.5 JTAG 인터페이스

표준 JTAG(조인트 테스트 액션 그룹) 인터페이스는 테스트 목적으로 포함된다. 칩의 복잡성에 기인하여, BIST(내장 자체 테스트) 및 기능 블록 격리를 포함하는 다양한 테스트 기술이 요구된다. 칩 영역에서 총경비의 10%가 전체 칩 테스트 회로 설계에 대하여 가정된다.

8 제네릭 프린터 드라이버

이 섹션은 아이프린트용 임의의 호스트-기반 프린터 드라이버의 제네릭 태양을 설명한다.

8.1 그래픽스 및 이미징 모델

본 발명은 프린터 드라이버가 다른 그래픽스 및 이미징 동작, 특히 합성 동작 및 텍스트 동작에 대해서 디바이스-특정 처리를 제공할 수 있기 위해서 프린터 드라이버가 호스트 그래픽스 시스템과 밀접하게 연결됨을 가정한다.

본 발명은 사용자-선택된 아이프린트-특정 ICC(국제 컬러 컨소시엄) 컬러 프로필에 기초해서, 표준 방식으로 디바이스-독립 컬러가 아이프린트-특정 CMYK 컬러로 변환될 수 있기 위해서, 호스트가 컬러 관리에 대한 지원을 제공함을 가정한다. 컬러 프로필은 사용자가 프린터에서 출력 매체를 지정할 때 사용자에게 의해서 암시적으로 통상 선택된다(즉, 평범한 용지, 코팅된 용지, 투명, 등). 프린터에 보내진 페이지 기술은 디바이스-특정 CMYK 컬러를 항상 포함한다.

본 발명은 호스트 그래픽스 시스템이 이미지 및 그래픽스를 프린터 드라이버에 의해서 지정된 공칭 해상도로 렌더링하지만 그것은 프린터 드라이버가 텍스트 렌더링을 제어하게 함을 가정한다. 특히, 그래픽스 시스템은 충분한 정보를 프린터 드라이버에 제공하여 그것이 공칭 디바이스 해상도보다 더 높은 해상도에서 텍스트를 렌더링 및 위치결정하게 한다.

본 발명은 호스트 그래픽스 시스템이 그것이 그래픽스 및 이미징 대상물을 합성하는 공칭 디바이스 해상도에서 연속톤 페이지 버퍼로의 랜덤 액세스를 요구하지만, 그것은 프린터 드라이버가 실제의 합성을 제어하게 함-즉, 그것은 프린터 드라이버가 페이지 버퍼를 관리하기를 기대함을 가정한다.

8.2 2개의-층 페이지 버퍼

프린터의 페이지 기술은 267 ppi 연속톤 층 및 800 dpi 블랙 층을 포함한다. 블랙 층은 개념적으로 연속톤 층 위에 있다 - 즉 블랙 층이 프린터에 의해서 연속톤 층 위로 합성된다. 따라서 프린터 드라이버는 중해상도 연속톤 층(261) 및 고해상도 블랙 층(262)을 대응되게 포함하는 페이지 버퍼(260)를 유지한다.

그래픽스 시스템은 대상물을 페이지 버퍼내로 밑으로부터 위로 렌더링 및 합성한다 - 즉 더 늦은 대상물이 더 이른 대상물을 가린다. 이것은 후에 합성될 2개의 층이 존재할 때가 아니라 단일 층만이 존재할 때 자연스럽게 작동한다. 따라서 연속톤 층상에 놓이는 대상물이 블랙 층상의 어떤 것을 가리는 때를 검출하는 것이 필요하다.

가림이 검출될 때, 가려진 블랙 화소는 연속톤 층과 합성되고 블랙 층으로부터 제거된다. 그 때 가리는 대상물은 연속톤 층상에서 아래로 놓이고, 특정 방식으로 블랙 화소와 상호작용할 수도 있다. 가리는 대상물의 합성 모드가 배경과의 상호작용이 가능하지 않은 그러한 것이라면, 그 때 블랙 화소는 연속톤 층과 합성됨이 없이 단지 버려질 수 있다. 실제로, 물론, 연속톤 층과 블랙 층 사이에 상호작용은 거의 없다.

프린터 드라이버는 그래픽스 시스템으로 267 ppi의 공칭 페이지 해상도를 지정한다. 가능한 경우에 프린터 드라이버는 이미지 및 그래픽스 대상물을, 블랙 텍스트를 제외하고, 267 ppi에서 화소 레벨로 렌더링하기 위해서 그래픽스 시스템에 의존한다. 프린터 드라이버는 모든 텍스트 렌더링 요청을 처리하고, 800 dpi에서 블랙 텍스트를 검출 및 렌더링하지만, 267 ppi에서 렌더링을 위해 블랙-아닌 텍스트 렌더링 요청을 그래픽스 시스템에 반환한다.

이상적으로 그래픽스 시스템 및 프린터 드라이버는 디바이스-독립 RGB에서 컬러를 조작하고, 페이지가 프린터에 보내지도록 완료되고 준비될 때까지 디바이스-특정 CMYK로의 변환을 연기한다. 이것은 페이지 버퍼 요구를 감소시키고 합성이 더 합리적이게 한다. CMYK 컬러 공간에서의 합성은 이상적이지 않다.

궁극적으로 그래픽스 시스템은 프린터 드라이버가 각각의 렌더링된 대상물을 프린터 드라이버의 페이지 버퍼내로 합성하도록 요구한다. 각각의 그러한 대상물은 24-비트 연속톤 RGB를 사용하고, 명시적인 (또는 암시적으로 불투명) 불투명도 채널을 갖는다.

프린터 드라이버는 3개의 파트에서 2-층 페이지 버퍼(260)를 유지한다. 제 1 파트는 중해상도(267 ppi) 연속톤 층(261)이다. 이것은 24-비트 RGB 비트맵으로 구성된다. 제 2 파트는 중해상도 블랙 층(263)이다. 이것은 8-비트 불투명도 비트맵으로 구성된다. 제 3 파트는 고해상도(800dpi) 블랙 층(262)이다. 이것은 1-비트 불투명도 비트맵으로 구성된다. 중해상도 블랙 층은 고해상도 불투명도 층의 서브샘플링된 버전이다. 실제로, 중해상도가 고해상도의 정수 인수 n 이라 가정하면(예를 들어 $n = 800/267 = 3$), 각각의 중해상도 불투명도 값은 대응 $n \times n$ 고해상도 불투명도 값을 평균함으로써 얻어진다. 이것은 박스-필터링된 서브샘플링에 대응한다. 블랙 화소의 서브샘플링은 고해상도 블랙 층에서 에지를 효과적으로 안티에일리어싱하고, 그로써 연속톤 층이 후속적으로 JPEG-압축 및 압축해제될 때 링잉(ringing) 인공물을 감소시킨다.

페이지 버퍼의 구조 및 크기가 도 36에 예시된다.

8.3 합성 모델

페이지 버퍼 합성 모델을 논의할 목적으로, 본 발명은 다음 변수를 정의한다.

(테이블 32)

합성 변수			
변수	설명	해상도	포맷
n	고해상도 환산 계수로의 매체	-	-
C_{BgM}	배경 연속톤 총 컬러	중	8-비트 컬러 구성요소
C_{ObM}	연속톤 대상물 컬러	중	8-비트 컬러 구성요소
a_{ObM}	연속톤 대상물 불투명도	중	8-비트 불투명도
a_{FgM}	중해상도 전경 블랙 총 불투명도	중	8-비트 불투명도
a_{FgH}	전경 블랙 총 불투명도	고	1-비트 불투명도
a_{TxH}	블랙 대상물 불투명도	고	1-비트 불투명도

불투명도 a_{TxH} 의 블랙 대상물이 블랙 층과 합성될 때, 블랙 층은 다음과 같이 업데이트된다:

$$a_{FgH}[x,y] \leftarrow a_{FgH}[x,y] \vee a_{TxH}[x,y] \quad (\text{규칙 1})$$

$$a_{FgM}[x,y] \leftarrow \frac{1}{n^2} \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} 255 a_{FgH}[nx+i, ny+j] \quad (\text{규칙 2})$$

대상물 불투명도는 블랙 층 불투명도와 단지 OR되고(규칙 1), 중해상도 블랙 층의 대응 파트는 고해상도 블랙 층으로부터 재계산된다(규칙 2).

컬러 C_{ObM} 및 불투명도 a_{ObM} 의 연속톤 대상물은 연속톤 층과 합성될 때, 연속톤 층 및 블랙 층이 다음과 같이 업데이트된다:

$$a_{ObM}[x,y] > 0 \text{ 이면 } C_{BgM}[x,y] \leftarrow C_{BgM}[x,y](1 - a_{FgM}[x,y]) \quad (\text{규칙 3})$$

$$a_{ObM}[x,y] > 0 \text{ 이면 } a_{FgM}[x,y] \leftarrow 0 \quad (\text{규칙 4})$$

$$a_{ObM}[x/n, y/n] > 0 \text{ 이면 } a_{FgH}[x,y] \leftarrow 0 \quad (\text{규칙 5})$$

$$C_{BgM}[x,y] \leftarrow C_{BgM}[x,y](1 - a_{ObM}[x,y]) + C_{ObM}[x,y]a_{ObM}[x,y] \quad (\text{규칙 6})$$

연속톤 대상물이 블랙 층을 가리는 곳 어디든지, 완전히 불투명하게는 아닐지라도, 영향받은 블랙 층 화소는 블랙 층으로부터 연속톤 층으로 밀리고, 즉 연속톤 층(규칙 3)과 합성되고 블랙 층으로부터 제거된다(규칙 4 및 규칙 5). 그 때 연속톤 대상물은 연속톤 층과 합성된다(규칙 6).

연속톤 대상물 화소가 완전히 불투명하다면(즉, $a_{ObM}[x,y] = 255$), 그 때 배경 연속톤 화소는 전경 연속톤 화소에 의해서 후속적으로 완벽히 말소될 것이기 때문에(규칙 6), 대응 블랙 화소를 배경 연속톤 층으로 밀어넣을 필요가 없다(규칙 3).

도 37 내지 도 41은 다양한 타입의 대상물을 2개 층에 의해서 표현된 이미지상으로 합성하는 배경 연속톤 층 및 전경 블랙 층상의 효과를 예시한다. 각각의 경우에서 2개 층의 상태는 대상물이 합성되기 전 및 후에 도시된다. 전경 및 배경 층의 다른 해상도는 층의 다른 화소 격자 밀도에 의해서 표시된다.

이미지의 실제 렌더링은 여기 논의의 포커스가 아니기 때문에, 2개 층에 표현된 출력 이미지는 화소 격자 없이 도시된다.

중해상도 전경 블랙 층은 예시되지 않지만, 암시적으로 존재한다. 규칙 1이 고해상도 전경 블랙 층에 적용될 때마다, 규칙 2는 중해상도 전경 블랙 층에 암시적으로 적용된다. 규칙 4가 적용될 때마다, 규칙 5는 또한 암시적으로 적용된다.

도 37은 블랙 대상물(270)을 화이트 이미지상으로 합성하는 효과를 예시한다. 블랙 대상물은 전경 블랙 층(271)내로 간단히 합성된다(규칙 1). 배경 연속톤 층(272)은 영향받지 않고, 출력 이미지(273)는 블랙 대상물이다.

도 38은 연속톤 대상물(280)을 화이트 이미지상으로 합성하는 효과를 예시한다. 연속톤 대상물(280)은 배경 연속톤 층(282)내로 간단히 합성된다(규칙 6). 전경 블랙 층(281)은 영향받지 않고, 출력 이미지(283)는 연속톤 대상물이다.

도 39는 연속톤 대상물(292)을 이미 포함하는 이미지상으로 블랙 대상물(290)을 합성하는 효과를 예시한다. 다시 블랙 대상물은 전경 블랙 층(291)내로 간단히 합성된다(규칙 1). 배경 연속톤 층은 영향받지 않고, 출력 이미지(293)는 연속톤 대상물(292) 위로 블랙 대상물(290)을 갖는다.

도 40은 블랙 대상물(301)을 이미 포함하는 이미지상으로 불투명한 연속톤 대상물(300)을 합성하는 효과를 예시한다. 연속톤 대상물은 존재하는 블랙 대상물의 파트를 가리기 때문에, 존재하는 2-레벨 대상물의 영향받은 파트가 전경 블랙 층으로부터 제거된다(302; 규칙 4). 연속톤 대상물은 완전히 불투명하기 때문에 영향받은 파트를 연속톤 층내로 합성할 필요가 없고, 따라서 규칙 3은 스킵된다. 연속톤 대상물은 어느때와 같이 배경 연속톤 층내로 합성되고(303; 규칙 6), 출력 이미지(304)는 블랙 대상물 위로 가리는 연속톤 대상물(300)을 보여준다.

도 41은 블랙 대상물(311)을 이미 포함하는 이미지상으로 부분적으로 투명한 연속톤 대상물(310)을 합성하는 효과를 예시한다. 연속톤 대상물은 존재하는 블랙 대상물의 파트를 부분적으로 투명하게 가리기 때문에, 블랙 대상물의 영향받은 파트는 연속톤 층내로 합성되고(312; 규칙 3), 그 때 전경 블랙 층으로부터 제거된다(313; 규칙 4). 그 후 연속톤 대상물은 어느 때와 같이 배경 연속톤 층내로 합성된다(314; 규칙 6).

최종 이미지(315)는 존재하는 블랙 대상물의 파트를 투명하게 가리는 그들 연속톤 화소의 어둡게 함을 보여준다.

8.4 페이지 압축 및 배달

페이지 렌더링이 완성된 직후, 프린터 드라이버는 그래픽스 시스템에서의 컬러 관리 기능의 도움으로 연속톤 층을 아이프린트-특정 CMYK로 변환한다.

그 때 프린터 드라이버는 블랙 층 및 연속톤 층을 섹션 5.2에서 설명된 바와 같은 아이프린트 페이지 기술내로 압축 및 패키징한다. 이러한 페이지 기술은 표준 스포러를 통해서 프린터에 배달된다.

블랙 층은 1-비트 불투명도 값의 세트로서 조작되지만, 1-비트 블랙 값의 세트로서 프린터에 배달됨을 주목하라. 이들 2개의 해석은 다르기는 하지만, 그들은 동일 표현을 공유하고, 그래서 어떠한 데이터 변환도 요구되지 않는다.

9 윈도우즈 9X/NT 프린터 드라이버

9.1 윈도우즈 9x/NT 인쇄 시스템

윈도우즈 9x/NT 인쇄 시스템에서[8][9], 프린터(320)는 그래픽스 디바이스이고, 애플리케이션(321)은 그래픽스 디바이스 인터페이스(322; GDI)를 통해서 그것과 통신한다. 프린터 드라이버 그래픽스 DLL(323; 동적 링크 라이브러리)은 GDI에 의해서 제공되는 다양한 그래픽스 기능의 디바이스-의존 태양을 구현한다.

스폴러(333)는 프린터에 페이지의 배달을 처리하고, 애플리케이션 요청 인쇄으로의 다른 머신상에 거주할 수 있다. 그것은 프린터로의 물리적 접속을 처리하는 포트 모니터(334)를 통해서 프린터에 페이지를 배달한다. 선택적 언어 모니터(335)는 프린터와의 통신에 부가적 프로토콜을 부과하는 프린터 드라이버의 파트이고, 특히 스포러를 위하여 프린터로부터의 상태 응답을 디코딩한다.

프린터 드라이버 사용자 인터페이스 DLL(336)은 프린터-특정 특성을 편집하고 프린터-특정 이벤트를 보고하기 위한 사용자 인터페이스를 구현한다.

윈도우즈 9x/NT 인쇄 시스템의 구조는 도 42에 예시된다.

아이프린트는 USB IEEE-1284 에뮬레이션을 사용하기 때문에, 아이프린트용 언어 모니터를 구현할 필요가 없다.

이 섹션의 나머지는 프린터 드라이버 그래픽스 DLL의 디자인을 설명한다. 적합한 윈도우즈 9x/NT DDK 문서 자료와 관련해서 읽어야 한다.

9.2 윈도우즈 9x/NT 그래픽스 디바이스 인터페이스(GDI)

GDI는 애플리케이션이 디바이스 표면(surface)상에 드로잉하게 하는 기능, 즉 전형적으로 디스플레이 스크린 또는 인쇄된 페이지의 추출을 제공한다. 래스터 디바이스에 대해서, 디바이스 표면은 개념적으로 컬러 비트맵이다. 애플리케이션은 디바이스-독립 방식으로, 즉 디바이스의 컬러 특성 및 해상도로부터 독립되어 표면에 드로잉할 수 있다.

애플리케이션은 전체 디바이스 표면으로의 랜덤 액세스를 갖는다. 이것은 메모리-제한 프린터 디바이스가 밴딩(banded) 출력을 요구한다면, 그 때 GDI는 전체 페이지의 GDI 커맨드를 버퍼링하고 윈도우잉된 그들을 각각의 밴드내로 차례로 재생하여야 한다는 것을 의미한다. 이것은 애플리케이션에 대단한 유연성을 제공하기는 하지만, 그것은 성능에 역으로 영향을 미칠 수 있다.

GDI는 컬러 관리를 지원함으로써, 애플리케이션에 의해서 제공된 디바이스-독립 컬러는 디바이스의 표준 ICC(국제 컬러 컨소시엄) 컬러 프로필에 따라 디바이스-의존 컬러로 투명하게 번역된다. 프린터 드라이버는, 예를 들어, 드라이버-관리된 프린터 속성 낱장상의 용지 타입의 사용자의 선택에 의존하는 다른 컬러 프로필을 활성화시킬 수 있다.

GDI는 라인 및 스플라인(spline) 윤곽선 그래픽스 (경로), 이미지, 및 텍스트를 지원한다. 윤곽선 폰트 글리프를 포함하는 윤곽선 그래픽스는 스트로킹되어 비트-맵핑된 브러시 패턴으로 채워질 수 있다. 그래픽스 및 이미지는 기하학적으로 변환되어 디바이스 표면의 콘텐츠와 합성될 수 있다. 윈도우즈 95/NT4는 불린 합성 연산만을 제공하는 반면에, 윈도우즈 98/NT5는 적당한 알파-블렌딩을 제공한다[9].

9.3 프린터 드라이버 그래픽스 DLL

래스터 프린터는, 이론적으로, 윈도우즈 9x/NT 하에 표준 프린터 드라이버 구성요소를 이용하고 이것은 프린터 드라이버 트리비얼을 개발하는 작업을 만들 수 있다. 이것은 디바이스 표면을 단일 비트맵으로서 모델링할 수 있음에 의존한다. 이것을 갖는 문제는 텍스트 및 이미지가 동일 해상도에서 렌더링되어야 하는 것이다. 이것은 텍스트 해상도를 타협하거나, 또는 너무 많은 출력 데이터를 발생시키는데, 성능을 타협한다.

일찍이 설명된 바와 같이, 아이프린트의 접근법은 각각의 재현을 최적화하기 위해서 블랙 텍스트 및 이미지를 다른 해상도에서 렌더링하는 것이다. 따라서 프린터 드라이버는 섹션 8에서 설명된 제네릭 디자인에 따라서 구현된다.

따라서 드라이버는 섹션 8.2에 설명된 바와 같이 2-층 3-파트 페이지 버퍼를 유지관리하고, 이것은 프린터 드라이버가 디바이스 표면을 관리하는 것을 떠맡아야 함을 의미하고, 차례로 그것이 디바이스 표면로의 모든 GDI 액세스를 중재하여야 함을 의미한다.

9.3.1 디바이스 표면을 관리

프린터 드라이버는 다음을 포함하는 많은 표준 함수를 지원해야 한다:

(테이블 33)

표준 그래픽스 드라이버 인터페이스 함수	
함수	설명
DrvEnableDriver	드라이버 그래픽스 DLL내로의 초기 엔트리 포인트. 드라이버에 의해서 지원되는 함수의 번지를 반환
DrvEnablePDEV	그것으로써 드라이버가 드로잉 표면을 연관시킬 수 있는 물리적 디바이스의 논리적 표현을 생성

DrvEnableSurface	일정 PDEV와 연관되어, 드로잉 온 되는 표면을 생성
------------------	--------------------------------

DrvEnablePDEV는 GDI로, 반환된 DEVINFO 구조의 flGraphicsCaps 멤버를 통해서, 드라이버의 그래픽스 렌더링 능력을 표시한다. 이것은 이하 더 논의된다.

DrvEnableSurface는 2개의 개념적 층 및 3개의 파트로 구성된 디바이스 표면을 생성한다(267 ppi 연속톤 층 24-비트 RGB 컬러, 267 ppi 블랙 층 8-비트 불투명도, 및 800 dpi 블랙 층 1-비트 불투명도). 이들 2개의 층을 캡슐에 넣는 가상 디바이스 표면은 267 ppi의 공칭 해상도를 갖고, 그래서 이것은 GDI 동작이 일어나는 해상도이다.

총계 페이지 버퍼는 약 33MB의 메모리를 요구하기는 하지만, PC 99 오피스 표준[5]은 최소 64MB를 지정한다.

실제로, 디바이스 표면을 관리하고 그것으로의 GDI 액세스를 중재하는 것은 프린터 드라이버가 다음의 부가적 함수를 지원해야 함을 의미한다.

(테이블 34)

디바이스-관리된 표면에 대하여 요구되는 그래픽스 드라이버 함수	
함수	설명
DrvCopyBits	디바이스-관리된 래스터 표면과 GDI-관리된 표준-포맷 비트맵 사이를 번역
DrvStrokePath	경로를 스트로크
DrvPaint	지정된 영역을 페인트
DrvTextOut	지정된 위치에서 글리프의 세트를 렌더링

이미지를 카피하는 것, 패스를 스트로킹하는 것, 및 영역을 채우는 것은 모두 연속톤 층 상에서 일어나고, 반면에 솔리드 블랙 텍스트를 렌더링하는 것은 2-레벨 블랙 층 상에서 일어난다. 더욱이, 블랙-아닌 텍스트를 렌더링하는 것은 또한 그것이 블랙 층상에서 지원되지 않기 때문에 연속톤 층상에서 일어난다. 역으로, 솔리드 블랙으로 스트로킹 또는 채우는 것은 블랙 층상에서 일어날 수 있다(우리가 그렇게 선택한다면).

프린터 드라이버는 상기 함수를 호킹하도록 강요되기는 하지만, 그것은 연속톤 층이 표준-포맷 비트맵이기 때문에, 연속톤 층에 적용되는 함수 호출을 함수의 대응 GDI 구현에 다시 펀트(punt)할 수 있다. 모든 DrvXxx 함수에 대해서 GDI에 의해 제공되는 대응 EngXxx 함수가 존재한다.

섹션 8.2에서 설명된 바와 같이, 연속톤 층에 대하여 예정된 대상물이 블랙 층상의 화소를 가릴 때, 가려진 블랙 화소는 연속톤 대상물이 연속톤 층과 합성되기 전에 블랙 층으로부터 연속톤 층으로 전송되어야 한다. 이 프로세스 동작의 핵심은 가리는 것이 그것이 GDI로 다시 펀트되기 전에 검출되어 호킹된 호출에서 처리된다는 것이다. 이것은 연속톤 대상물의 화소-화소 불투명도를 그 기하학으로부터 결정하는 것, 및 섹션 8.2에서 설명된 바와 같이 블랙 화소를 블랙 층으로부터 연속톤 층으로 선택적으로 전송하기 위해서 이 불투명도를 사용하는 것을 포함한다.

9.3.2 연속톤 대상물 기하학을 결정

각각의 연속톤 대상물의 기하학을 그것이 렌더링되기 전에 결정하고 따라서 그것이 어느 블랙 화소를 가리는지를 효율적으로 결정하는 것이 가능하다. DrvCopyBits 및 DrvPaint의 경우에서, 기하학은 직사각형의 세트로서 낱낱이 세어질 수 있는 클립 대상물(CLIPOBJ)에 의해서 결정된다.

DrvStrokePath의 경우에서, 일은 더 복잡하다. DrvStrokePath는 스트레이트-라인 및 베지에르-스플라인 커브 세그먼트 둘 다, 및 단일-화소-와이드 라인 및 기하학적-와이드 라인을 지원한다. 제 1 단계는 드라이버의 DEVINFO 구조의 flGraphicsCaps 멤버에서 대응하는 능력 플래그(GCAPS_BEZIER 및 GCAPS_GEOMETRICWIDE)를 클리어링함으로써 베지에르-스플라인 커브 세그먼트 및 기하학적-와이드 라인의 복잡성을 전적으로 피하는 것이다. 이것은 GDI가 DrvPaint로의 그러한 호출을 더 단순한 호출의 세트로서 재-포물레이팅하게 한다. 일반적으로, GDI는 고-레벨 능력을 가속화시키는 기회를 드라이버에 주지만, 드라이버에 의해 제공되지 않는 임의의 능력을 시뮬레이팅한다.

남은 것은 단일-화소-와이드 스트레이트 라인의 기하학을 결정하는 것 뿐이다. 그러한 라인은 솔리드이거나 코스메틱일 수 있다. 후자의 경우에서, 라인 스타일은 지정된 라인 애트리뷰트(LINEATTRS)에서의 스타일링 어레이에 의해서 결정된다. 스타일링 어레이는 라인이 불투명 및 투명 사이에서 그 길이를 따라서 어떻게 교대할 것인지를 지정하고, 그래서 다양한 대시트(dashed) 라인 효과 등을 지원한다.

브러시가 솔리드 블랙일 때, 800 dpi 해상도에 의해 함축된 증가된 폭으로써이기는 하지만, 스트레이트 라인은 또한 블랙 층으로 유용하게 렌더링될 수 있다.

9.3.3 텍스트를 렌더링

DrvTextOut의 경우에서, 일은 또한 더 복잡하다. 첫째로, 불투명한 배경은, 만약 있다면, 연속톤 층상의 임의의 다른 채움과 같이 처리된다(DrvPaint 참조). 전경 브러시가 블랙이 아니거나, 또는 믹스 모드가 효과적으로 불투명하지 않거나, 또는 폰트가 스케일 가능하지 않거나, 또는 폰트가 윤곽선 스트로킹을 표시하면, 그 때 호출은 연속톤 층에 적용되기 위해서 EngTextOut에 편트된다. 그러나, 호출이 편트되기 전에, 드라이버는 각각의 글리프의 기하학을 (FONTOBJ_cGetGlyphs를 통해서) 그 비트맵을 얻음으로써 결정하고 블랙 층에 대하여 통상의 가림 체크를 한다.

DrvTextOut 호출을 편팅하는 것이 허용되지 않으면(문서 자료가 모호하면), 그 때 드라이버는 복잡한 텍스트 동작을 금하여야 한다. 이것은 (GCAPS_VECTOR_FONT 능력 플래그를 클리어링함으로써) 윤곽선 스트로킹을 금하는 것, 및 (GCAPS_ARBMIXTXT 능력 플래그를 클리어링함으로써) 복잡한 믹스 모드를 금하는 것을 포함한다.

전경 브러시가 블랙 및 불투명하고 폰트가 스케일링 가능하고 스트로킹되지 않는다면, 그 때 글리프는 블랙 층상에서 렌더링된다. 이 경우에서 드라이버는 (다시 FONTOBJ_cGetGlyphs를 통해서, 그러나 PATHOBJ로서) 그 윤곽선을 얻음으로써 각각의 글리프의 기하학을 결정한다. 그 때 드라이버는 800 dpi에서 그 윤곽선으로부터 각각의 글리프를 렌더링하고 그것을 블랙 층에 기록한다. 윤곽선 기하학이 디바이스 좌표를 사용하기는 하지만(즉, 267 ppi에서), 좌표는 더 높은-해상도 렌더링에 대하여 많은 분수 정밀도를 갖는 고정 포인트 포맷이다.

스트라이크스루 및 언더라인 직사각형이, 지정된다면, 글리프 기하학에 추가된다는 것을 주목하라.

드라이버는 (다시 267 ppi 디바이스 좌표에서의) 글리프 위치가, 800 dpi에서 정확한 위치결정을 허용하기 위해서, 높은-정밀도 고정-포인트 포맷으로 GDI에 의해서 공급되는 것을 요청하기 위해서 DEVINFO에서의 GCAPS_HIGHRESTEXT 플래그를 세팅하여야 한다. 드라이버는 또한 DrvGetGlyphMode 함수의 구현을 제공해야 하는데, 그것이 글리프가 비트맵보다는 윤곽선으로서 캐싱되어야 한다는 것을 GDI에 표시할 수 있기 위함이다. 이상적으로 드라이버는 메모리가 허용하는 효율에 대해서 렌더링된 글리프 비트맵을 캐싱하여야 한다. 특정 포인트 크기 이하의 글리프만이 캐싱되어야 한다.

9.3.4 연속톤 층을 압축

일찍이 설명된 바와 같이, 연속톤 층은 JPEG를 사용하여 압축된다. 포워드 불연속 코사인 변환(DCT)은 JPEG 압축의 가장 비용이 드는 파트이다. 현재 고-품질 소프트웨어 구현에서, 각각의 8×8 블록의 포워드 DCT는 12 정수 승산 및 32 정수 가산을 요구한다[7]. 펜티엄 프로세서상에서, 정수 승산은 10 사이클을 요구하고 정수 가산은 2 사이클을 요구한다[11]. 이것은 184 사이클의 블록 당 총 코스트에 동등하다.

25.5MB 연속톤 층은 약 77M사이클의 전체 포워드 DCT 코스트를 주는 417,588 JPEG 블록으로 구성된다. 300MHz, PC 99 데스크톱 표준에서[5], 이것은 페이지 당 2초 한계내에 만족되는 0.26초에 동등하다.

10 참조

[1] ANSI/EIA 538-1988, *Facsimile Coding Schemes and Coding Control Functions for Group 4 Facsimile Equipment*, August 1988

[2] Humphreys, G.W., and V. Bruce, *Visual Cognition*, Lawrence Erlbaum Associates, 1989, p.15

[3] IEEE Std 1284-1994, *IEEE Standard Signaling Method for a Bidirectional Parallel Peripheral Interface for Personal Computers*, 2 December 1994

- [4] Intel Corp. and Microsoft Corp., *PC 98 System Design Guide*, 1997
- [5] Intel Corp. and Microsoft Corp., *PC 99 System Design Guide*, 1998
- [6] ISO/IEC 19018-1:1994, *Information technology – Digital compression and coding of continuous-tone still images: Requirements and guidelines*, 1994
- [7] Loeffler, C., A. Ligtenberg and G. Moschytz, "Practical Fast 1-D DCT Algorithms with 11 Multiplications", Proceedings of the International Conference on Acoustics, Speech, and Signal Processing 1989 (ICASSP '89), pp.988-991
- [8] Microsoft Corp., *Microsoft Windows NT 4.0 Device Driver Kit*, 1997
- [9] Microsoft Corp., *Microsoft Windows NT 5.0 Device Driver Kit*, 1998
- [10] Olsen, J. "Smoothing Enlarged Monochrome Images", in Glassner, A.S. (ed.), *Graphics Gems*, AP Professional, 1990
- [11] Schmit, M.L., *Pentium Processor Optimization Tools*, AP Professional, 1995
- [12] Thompson, H.S., *Multilingual Corpus 1* CD-ROM, European Corpus Initiative
- [13] Urban, S.J., "Review of standards for electronic imaging for facsimile systems", *Journal of Electronic Imaging*, Vol.1(1), January 1992, pp.5-21
- [14] USB Implementers Forum, *Universal Serial Bus Specification*, Revision 1.0, 1996
- [15] USB Implementers Forum, *Universal Serial Bus Device Class Definition for Printer Devices*, Version 1.07 Draft, 1998
- [16] Wallace, G.K., "The JPEG Still Picture Compression Standard", *Communications of the ACM*, 34(4), April 1991, pp.30-44
- [17] Yasuda, Y., "Overview of Digital Facsimile Coding Techniques in Japan", Proceedings of the IEEE, Vol. 68 (7), July 1980, pp.830-845

넓게 설명된 바와 같은 본 발명의 범위 또는 취지로부터 벗어남이 없이 특정 실시예에서 보여진 바와 같이 수많은 변형 및/또는 수정이 본 발명에 이루어질 수 있다는 것을 당업자는 인식할 것이다. 따라서, 본 실시예는 모든 고려에서 한정적이지 아니라 예시적으로 생각될 것이다.

발명의 효과

본 발명에 의하면, 프린터용 파이어링 펄스 지속시간 제어 시스템으로서, 이용될 수 있는 전압을 표시하는 신호를 수신하는 제 1 입력 포트; 파이어링 펄스에 대하여 소정 지속시간을 표현하는 제어 신호를 발생시키기 위해서, 제 1 입력 포트에서 수신된 신호에 의해서 인덱싱된, 프로그래밍 가능한 펄스 지속시간 테이블; 및 제어 신호를 송신하는 출력 포트를 포함하는, 프린터용 파이어링 펄스 지속시간 제어 시스템이 제공된다.

(57) 청구의 범위

청구항 1.

프린터용 파이어링 펄스 지속시간 제어 시스템에 있어서,

이용될 수 있는 전압을 표시하는 신호를 수신하는 제 1 입력 포트;

파이어링 펄스에 대하여 소정 지속시간을 표현하는 제어 신호를 발생시키기 위해서, 제 1 입력 포트에서 수신된 신호에 의해서 인덱싱된, 프로그래밍 가능한 펄스 지속시간 테이블; 및

제어 신호를 송신하는 출력 포트를 포함하는 것을 특징으로 하는 제어 시스템.

청구항 2.

제 1 항에 있어서, 프린터에서의 프린트헤드의 온도를 표시하는 신호를 수신하는 제 2 입력 포트를 더 포함하고, 프로그래밍 가능한 펄스 지속시간 테이블은 제 1 및 제 2 입력 포트에서 수신된 신호에 의해서 인덱싱되는 것을 특징으로 하는 제어 시스템.

청구항 3.

제 1 항에 있어서, 테이블에서의 엔트리는 0-4 μ s의 범위에서의 값을 표현하고 제어 출력은 인덱싱된 엔트리에 따라 발생하는 것을 특징으로 하는 제어 시스템.

청구항 4.

제 1 항에 있어서, 펄스 폭 발생기는 그 제어 출력을 수신하고 프린트헤드용 파이어링 펄스를 발생시키기 위해서 제어 시스템의 흐름의 아래에 놓인 것을 특징으로 하는 제어 시스템.

청구항 5.

제 1 항에 있어서, 테이블은 인쇄 런의 제 1 페이지를 인쇄하기 전에 기록되는 것을 특징으로 하는 제어 시스템.

청구항 6.

제 1 항에 있어서, 테이블은 인쇄 런의 페이지 사이에서 업데이트되는 것을 특징으로 하는 제어 시스템.

청구항 7.

제 1 항에 있어서, 테이블에서의 각각의 엔트리는

사용자 휘도 설정;

잉크의 점성 커브;

프린트헤드가 얼마나 뜨거운지를 제어기에 알려주는 T센스;

얼마나 많은 전압이 액추에이터에 이용될 수 있는지를 제어기에 알려주는 V센스;

액추에이터 히터의 저항률(스퀘어 당 옴)을 제어기에 알려주는 R센스; 및

히터의 임계부의 폭을 제어기에 알려주는 W센스 중 하나 이상으로부터의 가중치를 포함하는 것을 특징으로 하는 제어 시스템.

청구항 8.

제 1 항에 있어서, 펄스 지속시간 테이블은 256 엔트리를 갖고, 각각의 엔트리는 8-비트인 것을 특징으로 하는 제어 시스템.

청구항 9.

제 2 항에 있어서, 전압을 표시하는 신호는 V센스로부터 오고, 온도를 표시하는 신호는 T센스로부터 오는 것을 특징으로 하는 제어 시스템.

청구항 10.

제 9 항에 있어서, 입력 포트에서 수신된 신호는 테이블을 인덱싱하는데 사용하기 위해서 변환되는 것을 특징으로 하는 제어 시스템.

청구항 11.

제 10 항에 있어서, 테이블로의 8-비트 엔트리는 V센스로부터의 상위 4-비트 및 T센스로부터의 하위 4-비트로 구성된 2개의 4-비트 수에 의해서 인덱싱되는 것을 특징으로 하는 제어 시스템.

청구항 12.

제 1 입력 포트;

프로그래밍 가능한 펄스 지속시간 테이블; 및

출력 포트를 포함하는, 프린터용 파이어링 펄스 지속시간 제어 시스템에서, 파이어링 펄스 지속시간 제어 신호를 발생시키는 방법에 있어서,

이용될 수 있는 전압을 표시하는 제 1 신호를 수신하는 단계;

파이어링 펄스에 대해서 소정 지속시간을 표현하는 제어 신호를 발생시키기 위해 제 1 신호에 의해서 프로그래밍 가능한 펄스 지속시간 테이블을 인덱싱하는 단계; 및

제어 신호를 송신하는 단계를 포함하는 것을 특징으로 하는 파이어링 펄스 지속시간 제어 신호 발생 방법.

청구항 13.

제 12 항에 있어서, 제 2 입력이 제공되고,

프린터에서의 프린트헤드의 온도를 표시하는 제 2 신호를 수신하는 단계; 및

제 1 및 제 2 신호에 의해서 프로그래밍 가능한 펄스 지속시간 테이블을 인덱싱하는 단계를 더 포함하는 것을 특징으로 하는 파이어링 펄스 지속시간 제어 신호 발생 방법.

청구항 14.

제 12 항에 있어서, 인쇄 런의 제 1 페이지를 인쇄하기 전에 테이블을 기록하는 단계가 선행하는 것을 특징으로 하는 파이어링 펄스 지속시간 제어 신호 발생 방법.

청구항 15.

제 14 항에 있어서, 인쇄 런의 페이지 사이에서 테이블을 업데이트하는 단계를 더 포함하는 것을 특징으로 하는 파이어링 펄스 지속시간 제어 신호 발생 방법.

청구항 16.

제 12 항에 있어서,

사용자 휘도 설정;

잉크의 점성 커브;

프린트헤드가 얼마나 뜨거운지를 제어기에 알려주는 T센스;

얼마나 많은 전압이 액추에이터에 이용될 수 있는지를 제어기에 알려주는 V센스;

액추에이터 히터의 저항률(스퀘어 당 옴)을 제어기에 알려주는 R센스; 및

히터의 임계부의 폭을 제어기에 알려주는 W센스 중 하나 이상으로써 테이블에서의 각각의 엔트리에 가중치를 부여하는 단계를 더 포함하는 것을 특징으로 하는 파이어링 펄스 지속시간 제어 신호 발생 방법.

청구항 17.

제 13 항에 있어서, 전압을 표시하는 신호에 대해 V센스를 사용하는 단계 및 온도를 표시하는 신호에 대해 T센스를 사용하는 단계를 더 포함하는 것을 특징으로 하는 파이어링 펄스 지속시간 제어 신호 발생 방법.

청구항 18.

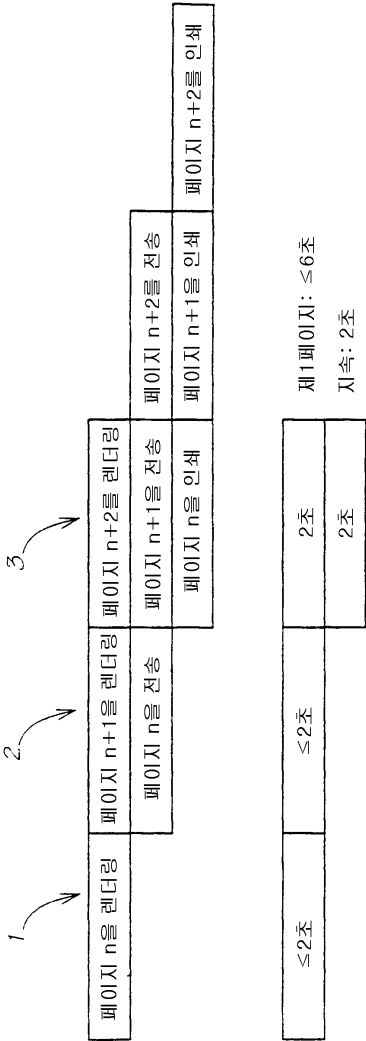
제 17 항에 있어서, 테이블을 인덱싱하는데 사용하기 위해서 입력 포트에서 수신된 신호를 변환하는 단계를 더 포함하는 것을 특징으로 하는 파이어링 펄스 지속시간 제어 신호 발생 방법.

청구항 19.

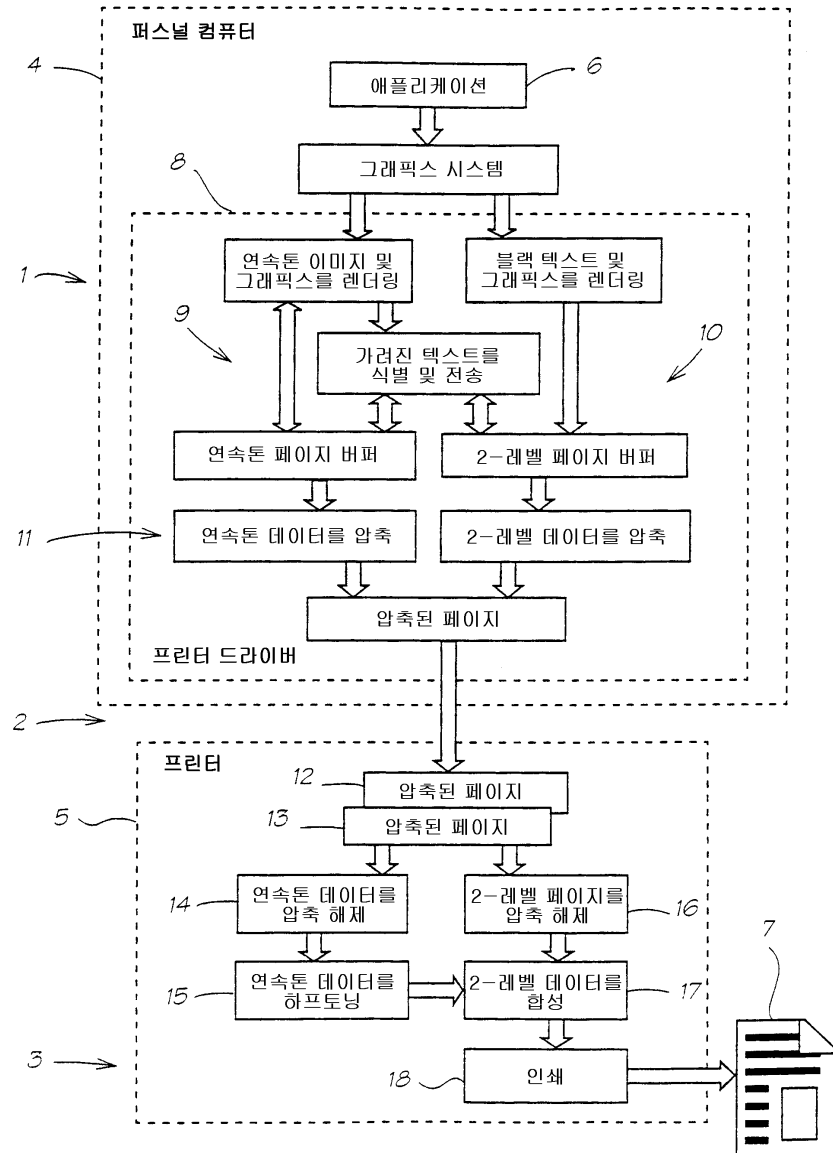
제 18 항에 있어서, V센스로부터의 상위 4-비트 및 T센스로부터의 하위 4-비트로 구성된 2개의 4-비트 수에 의해서 8-비트 엔트리를 테이블에 인덱싱하는 단계를 더 포함하는 것을 특징으로 하는 파이어링 펄스 지속시간 제어 신호 발생 방법.

도면

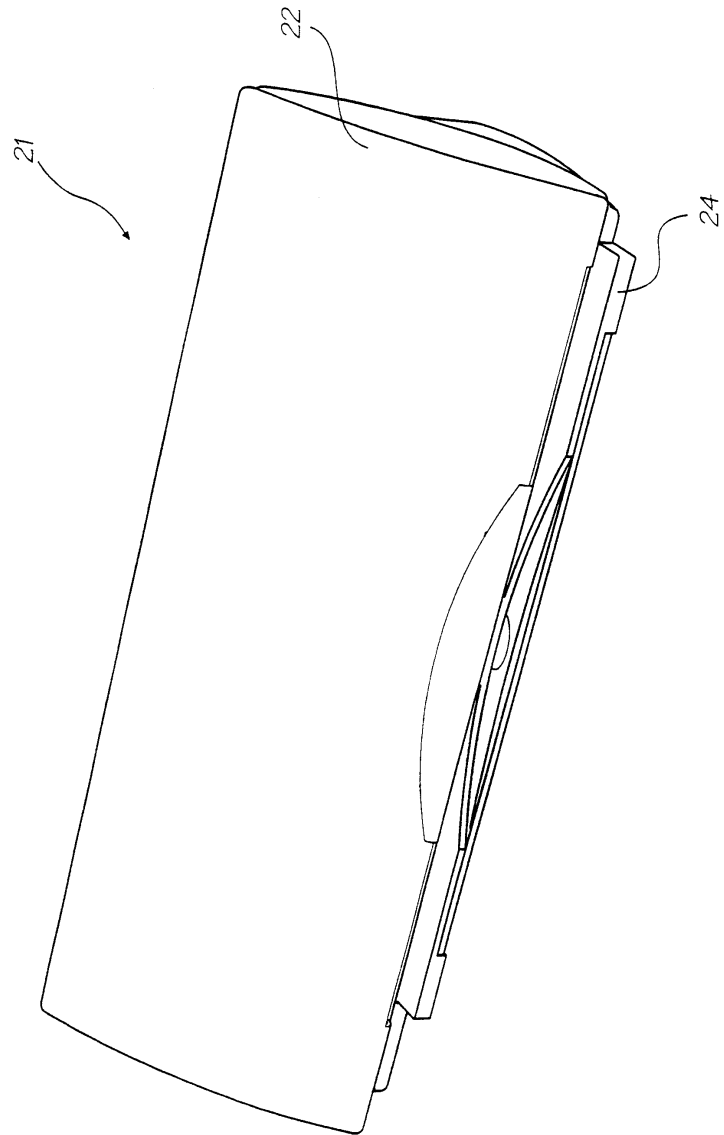
도면1



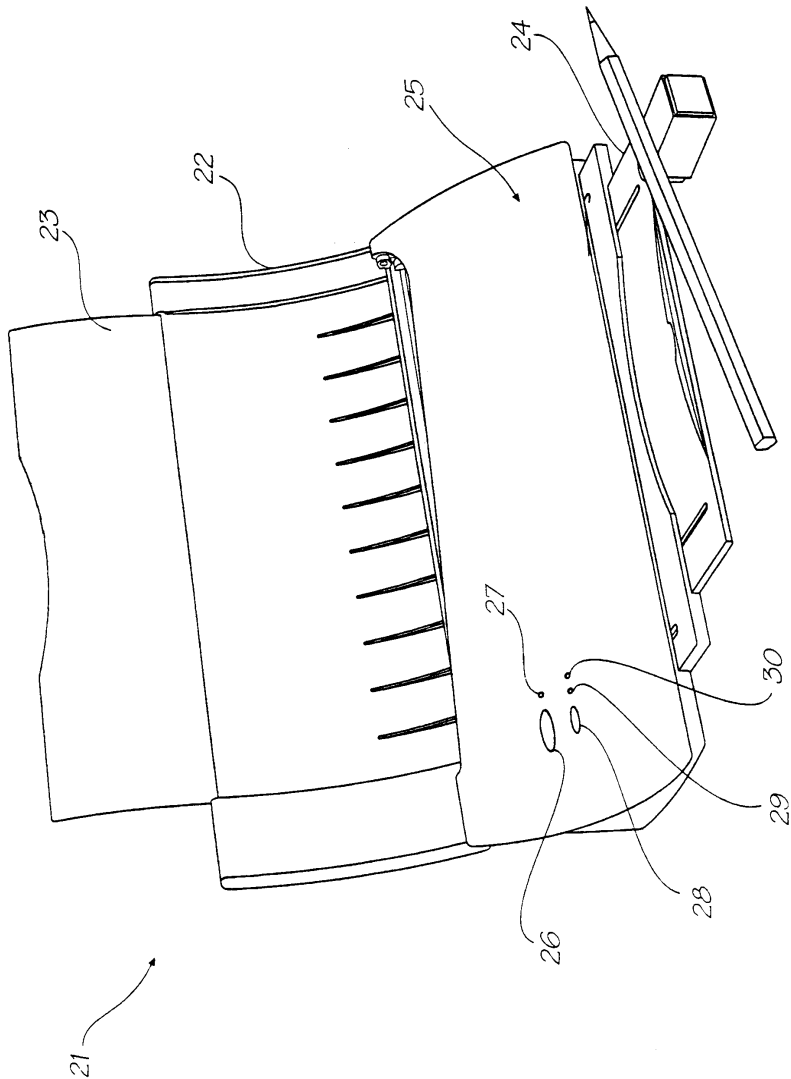
도면2



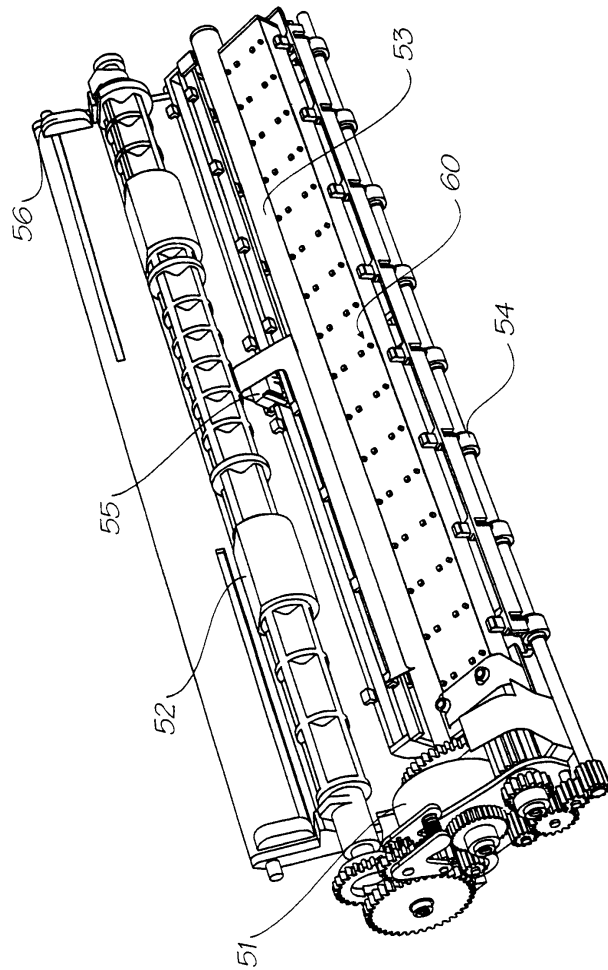
도면3



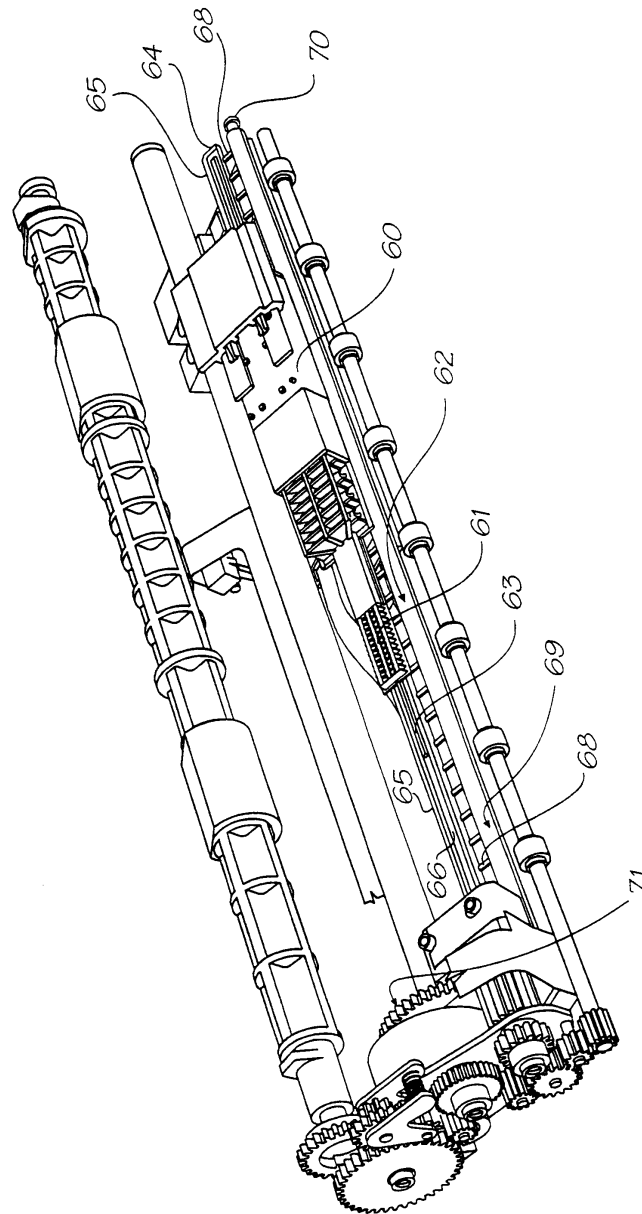
도면4



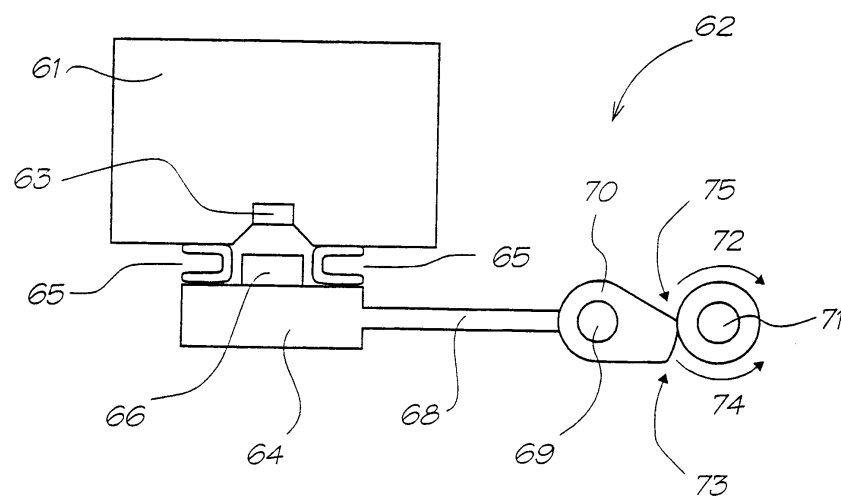
도면5



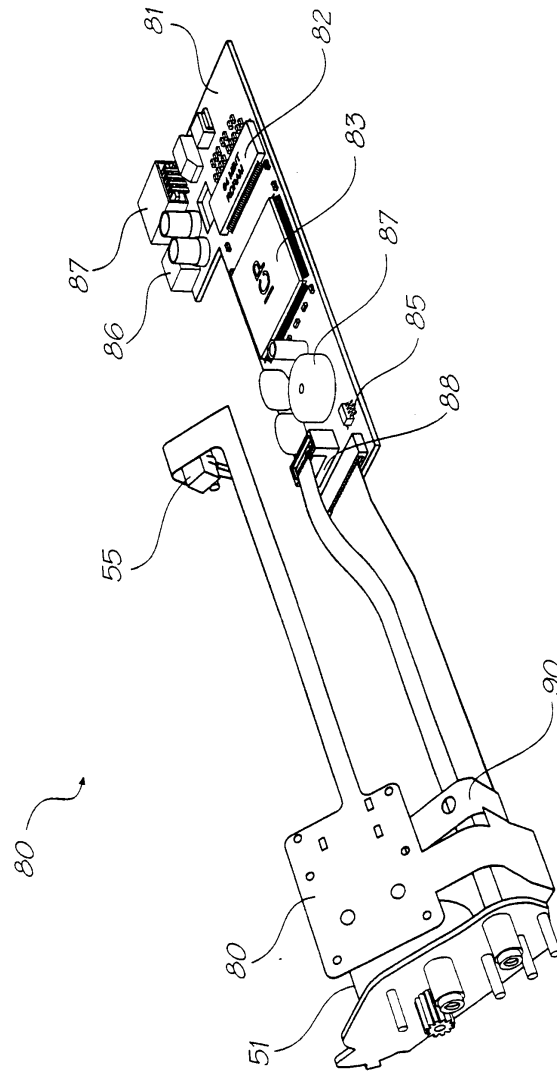
도면6



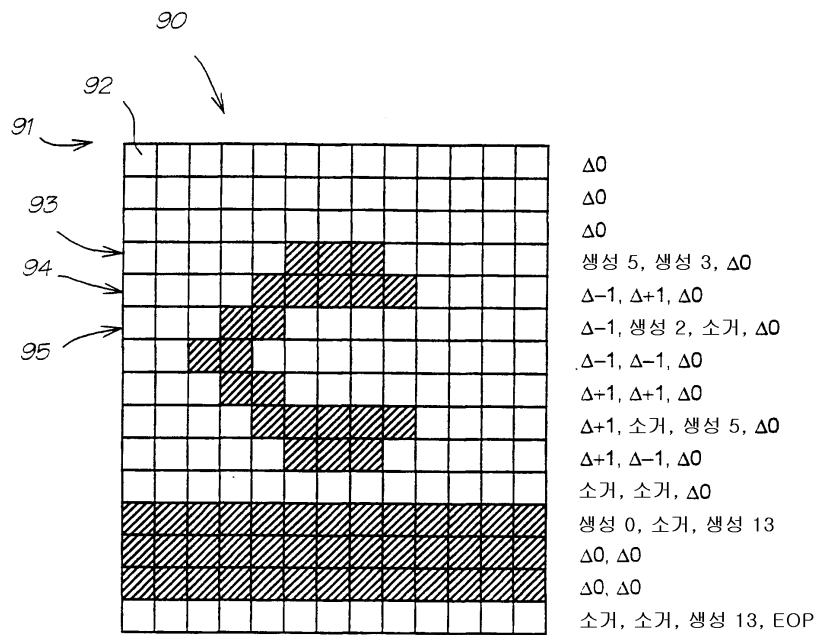
도면7



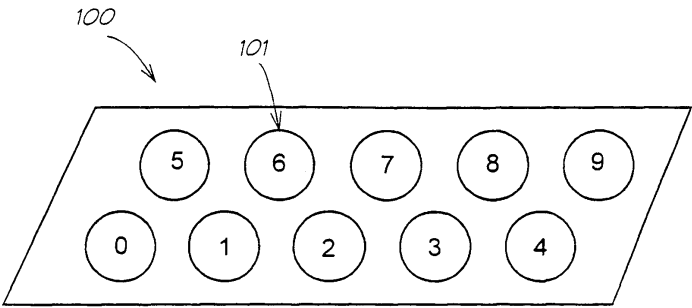
도면8



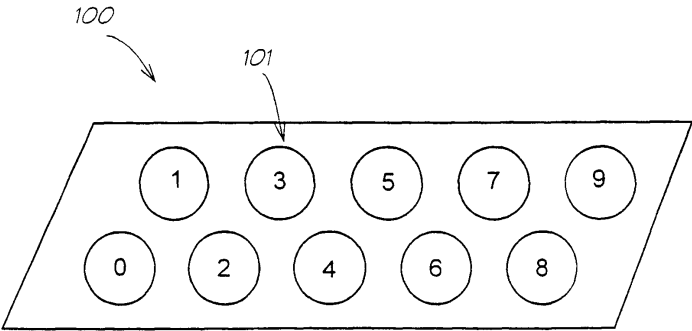
도면9



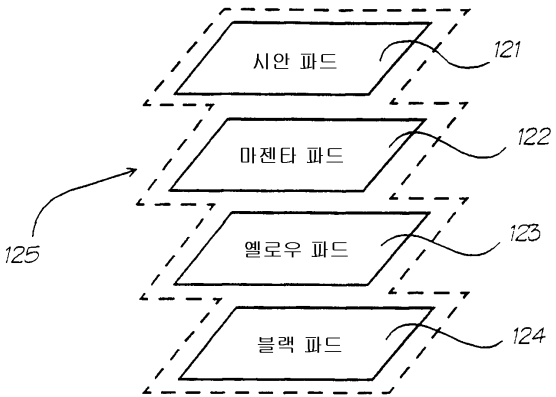
도면10



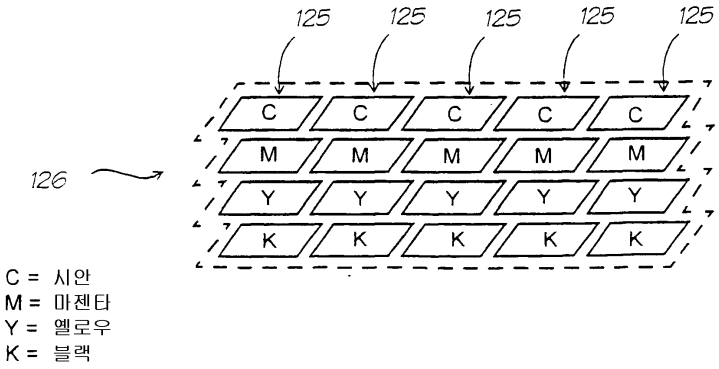
도면11



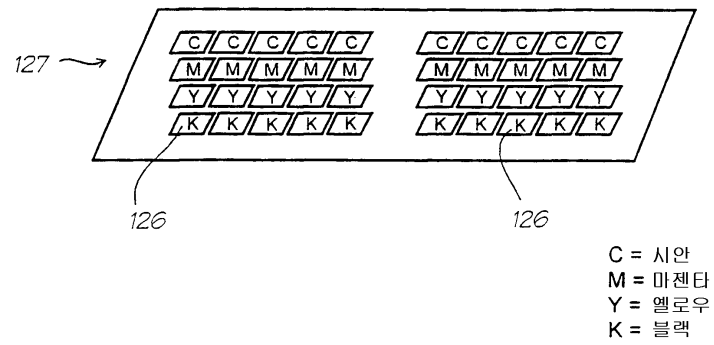
도면12



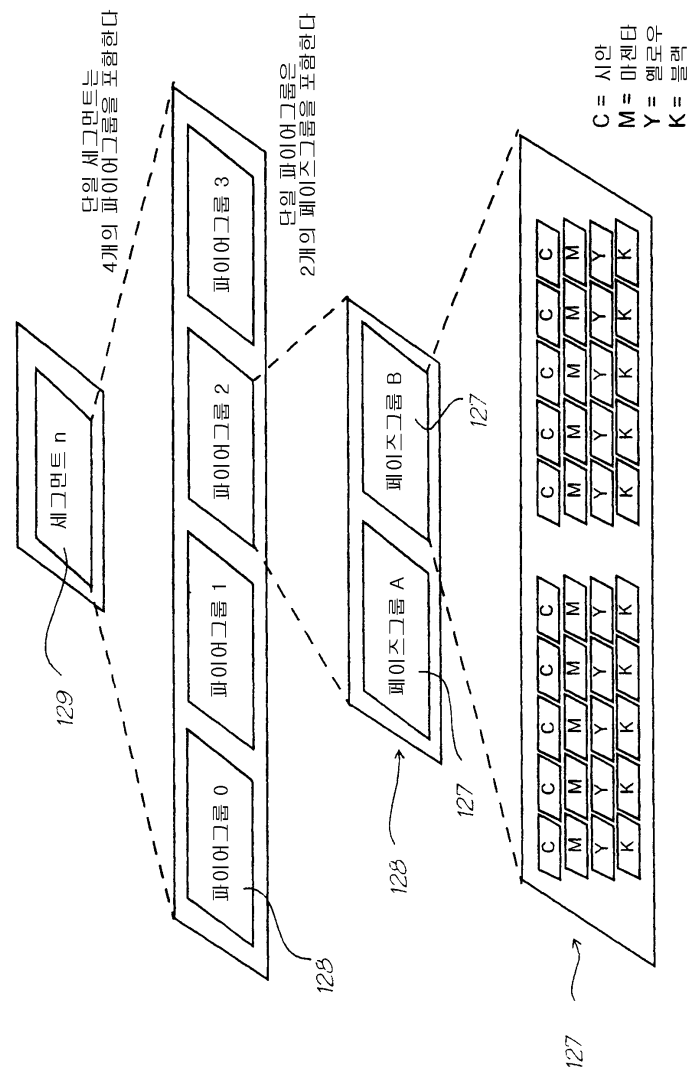
도면13



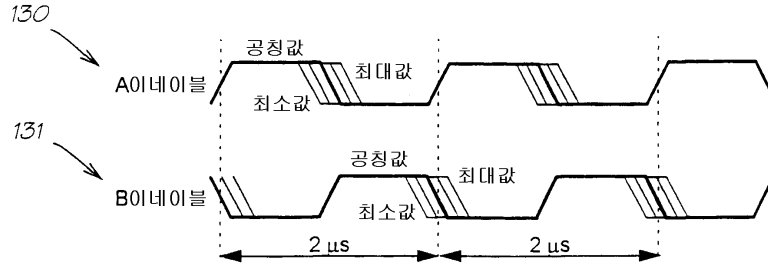
도면14



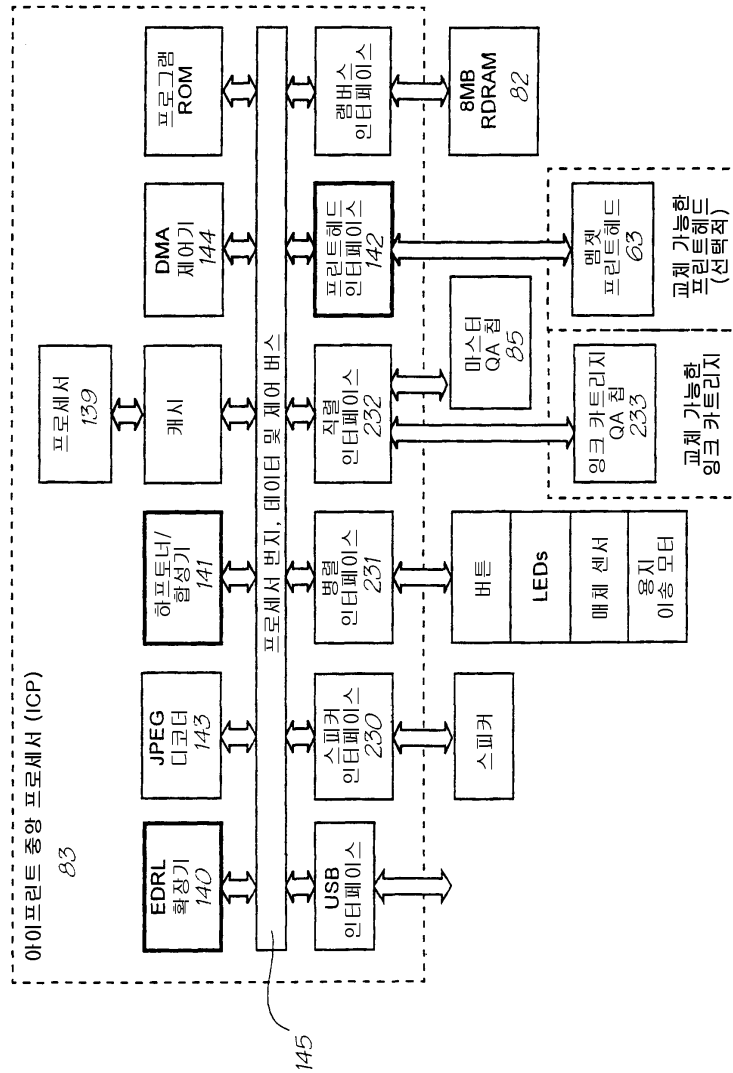
도면15



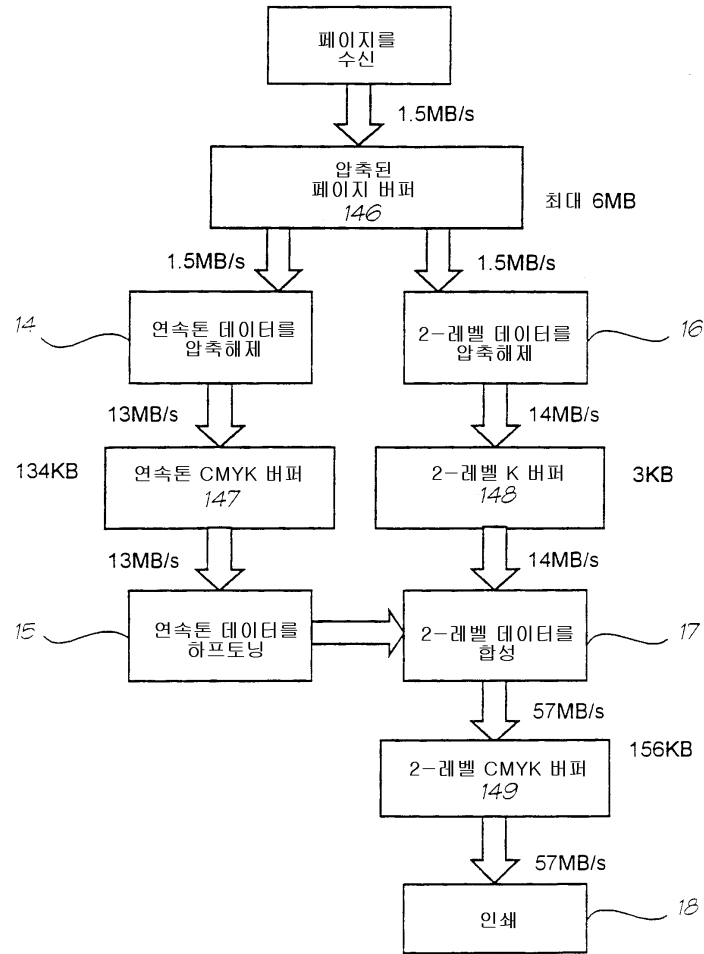
도면16



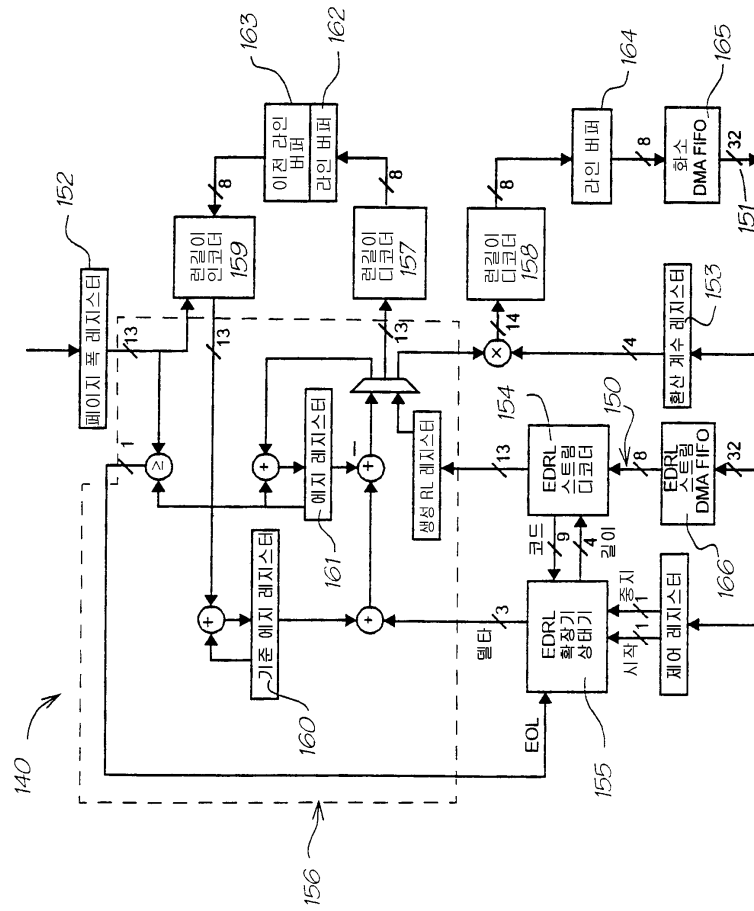
도면17



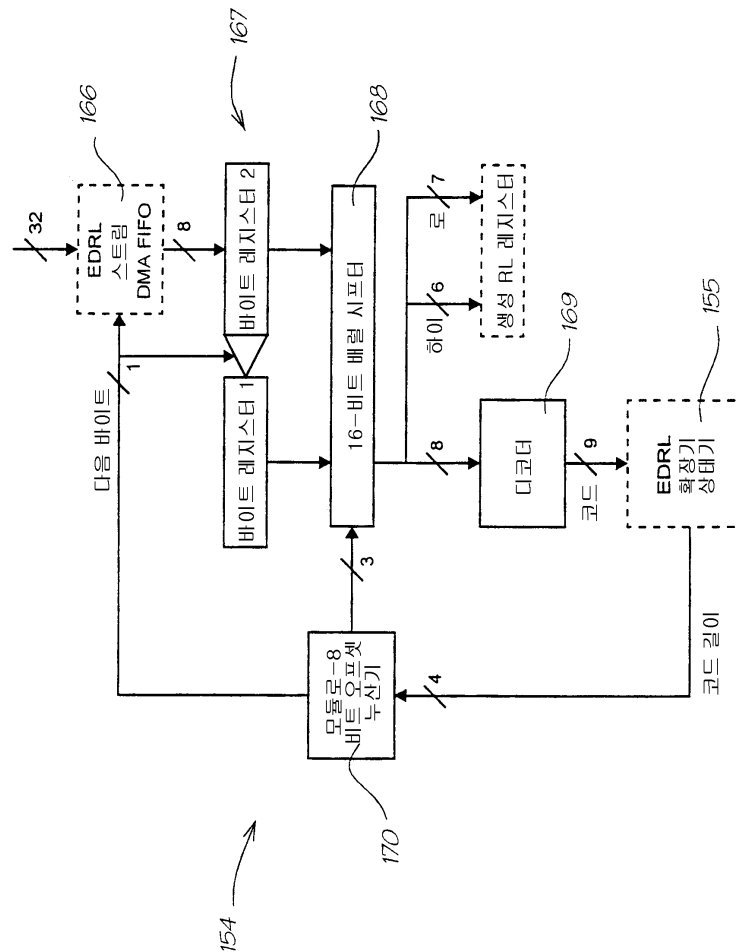
도면18



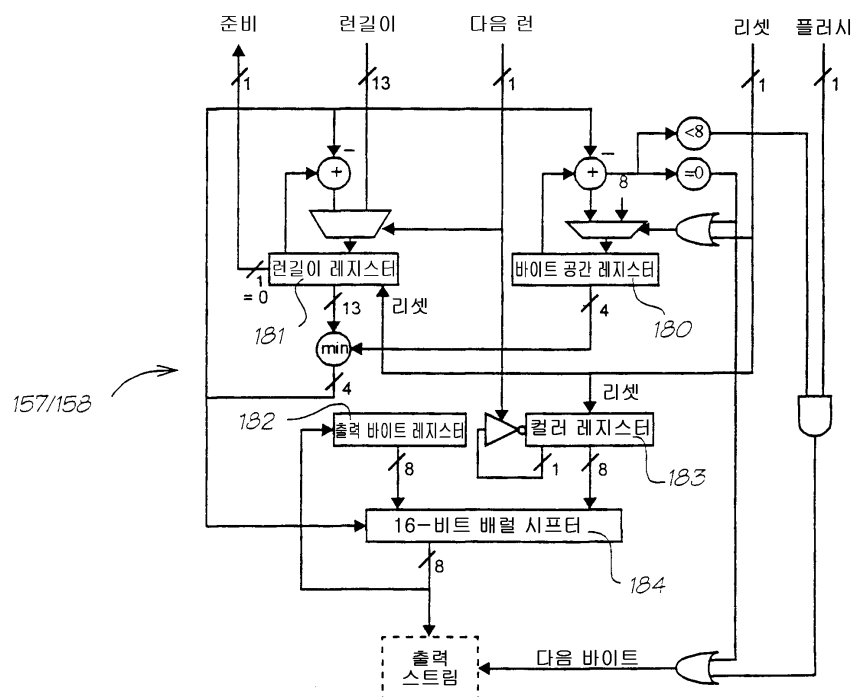
도면19



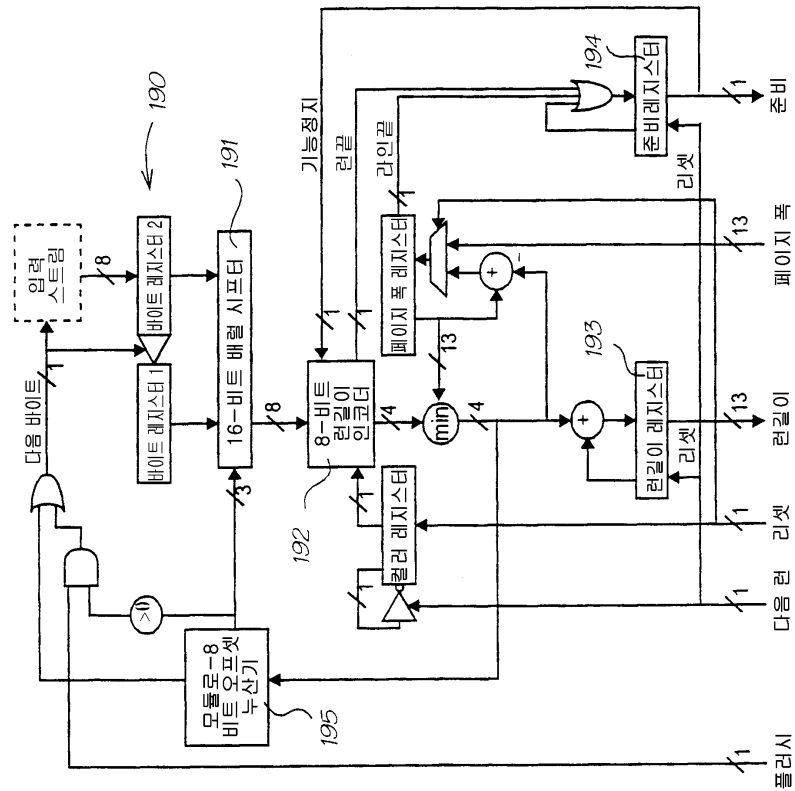
도면20



도면21

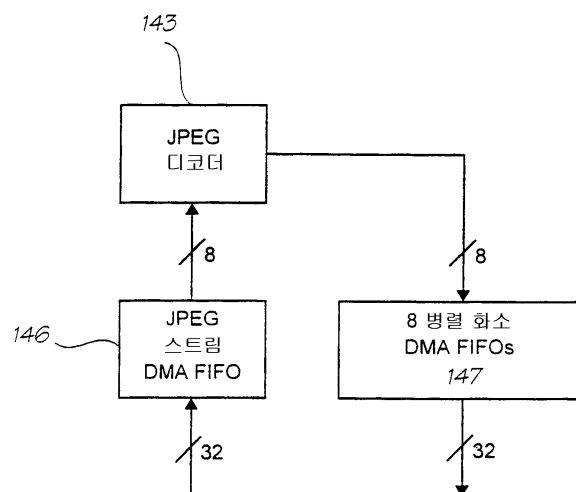


도면22

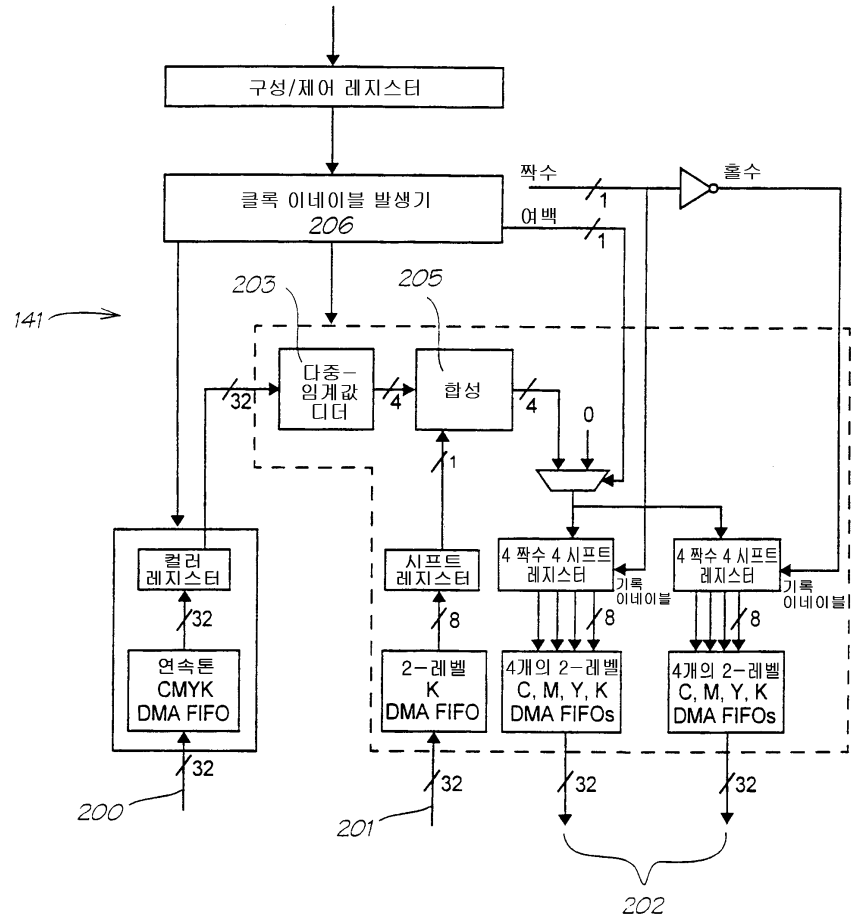


159

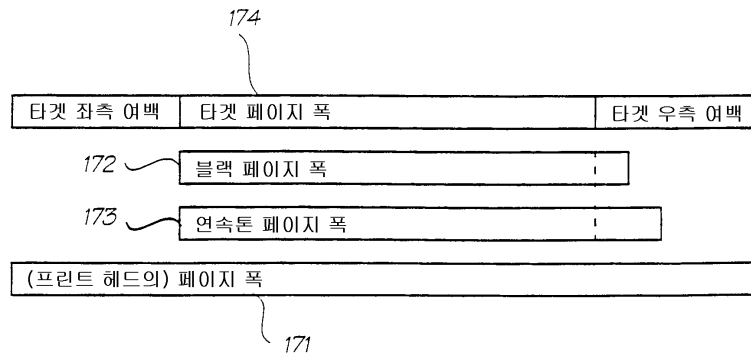
도면23



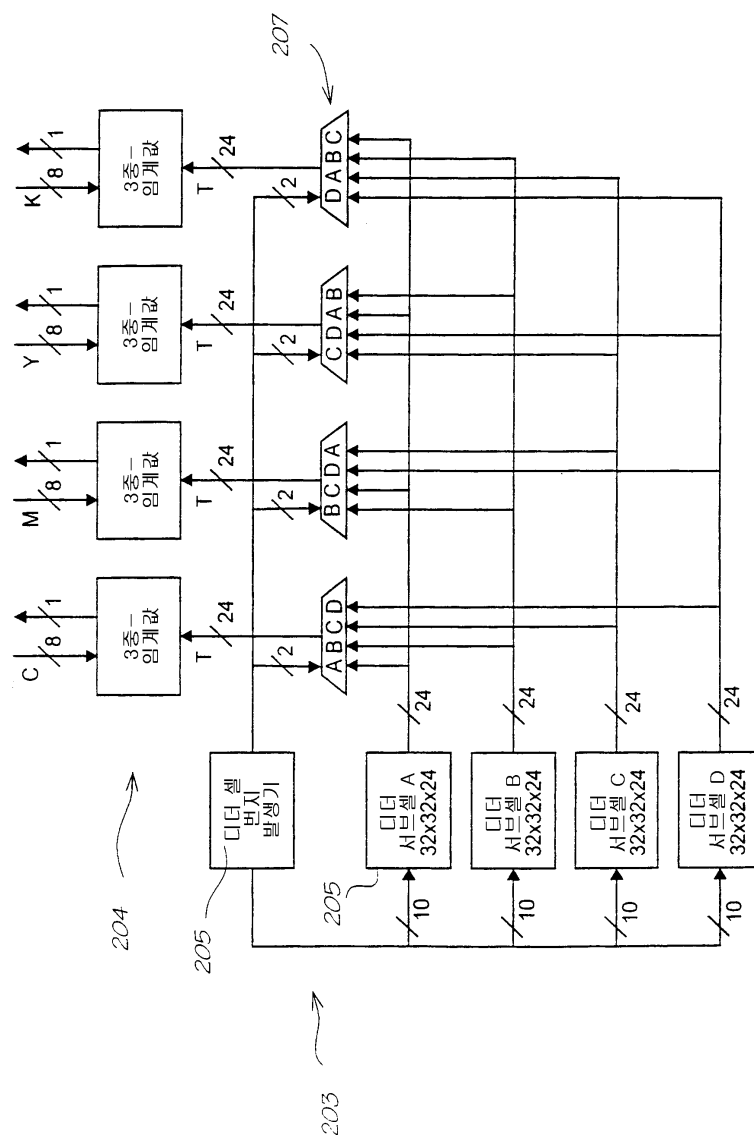
도면24



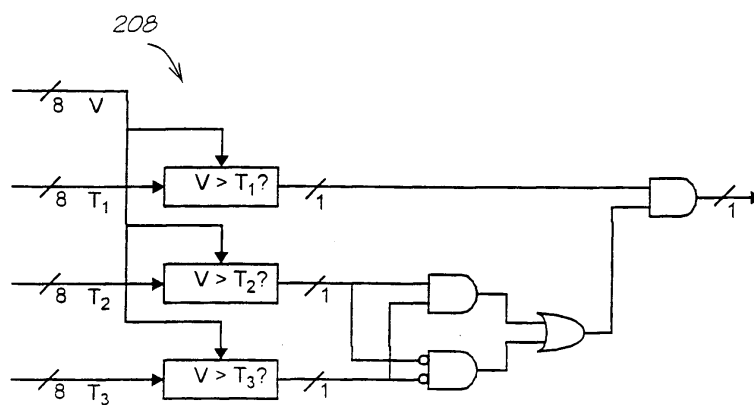
도면25



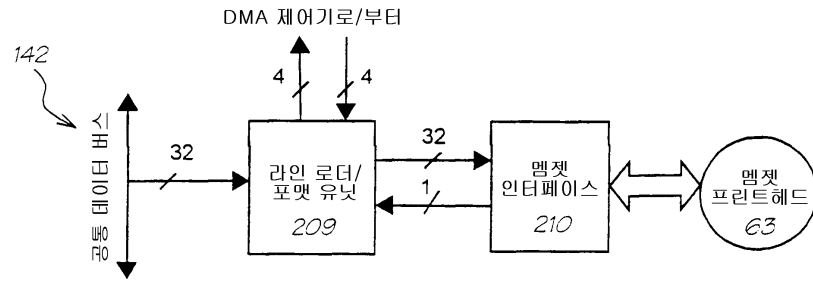
도면26



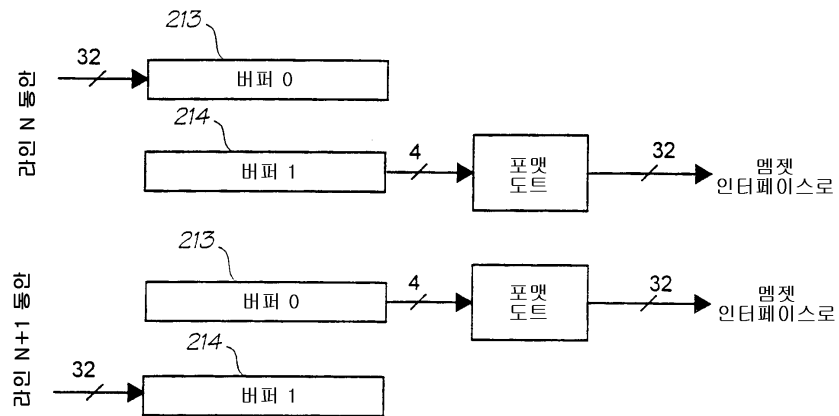
도면27



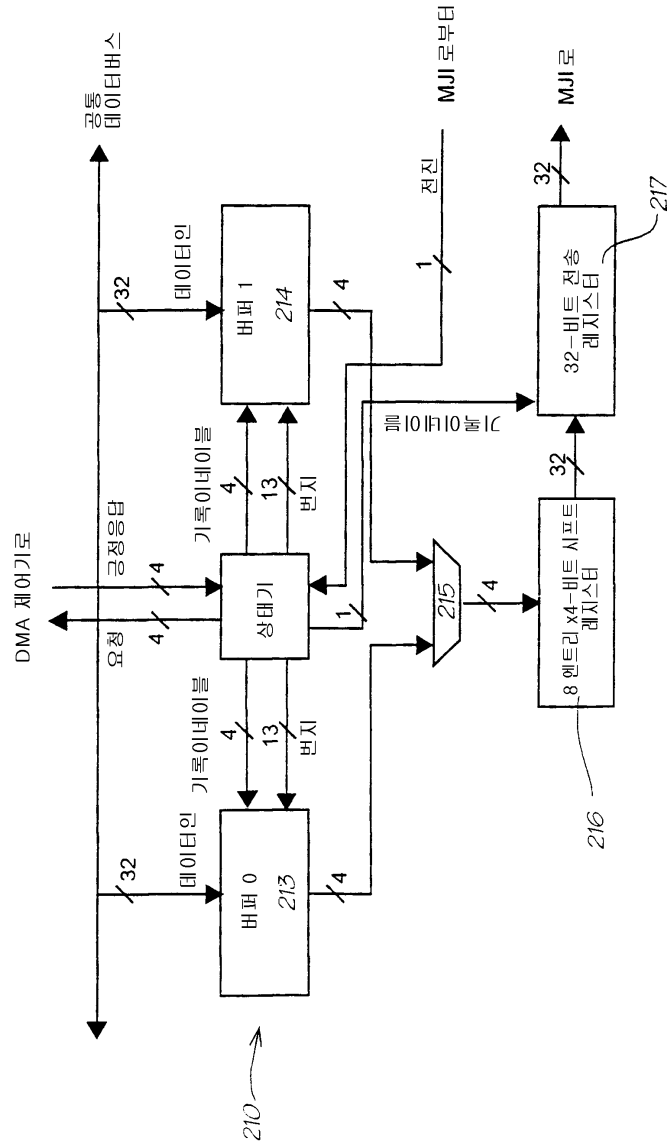
도면28



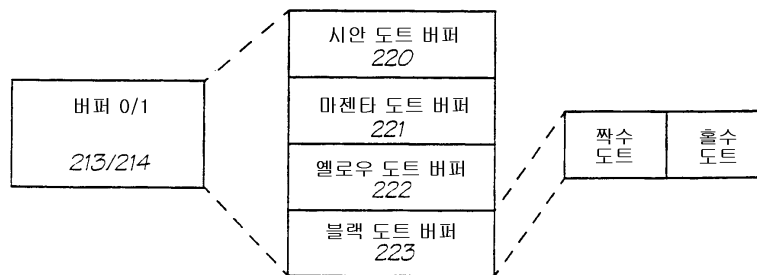
도면29



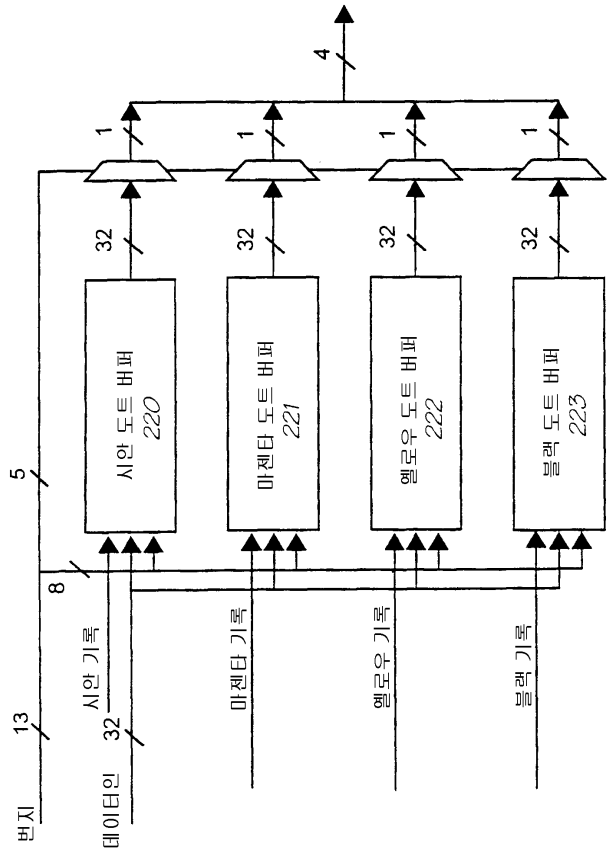
도면30



도면31

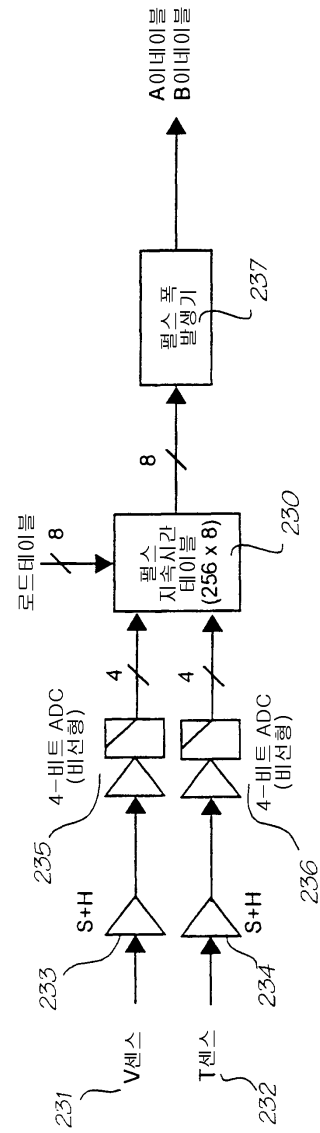


도면32

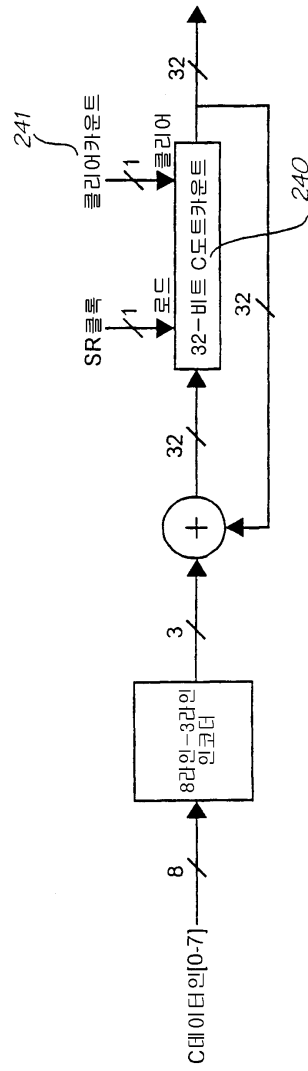


213/214

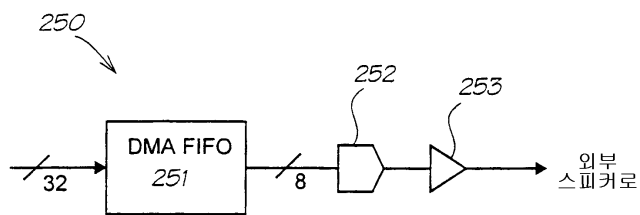
도면33



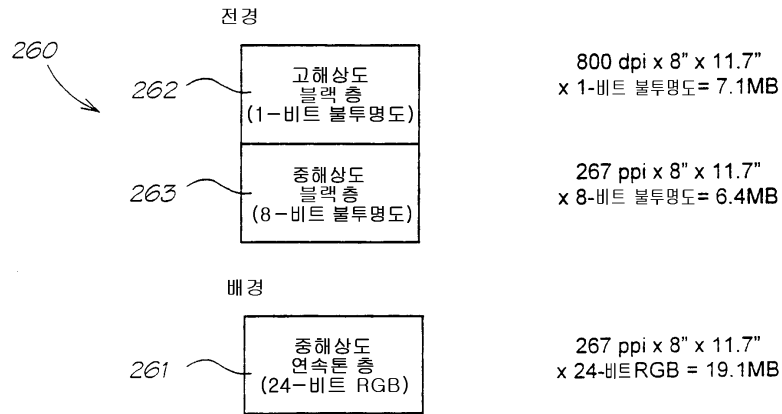
도면34



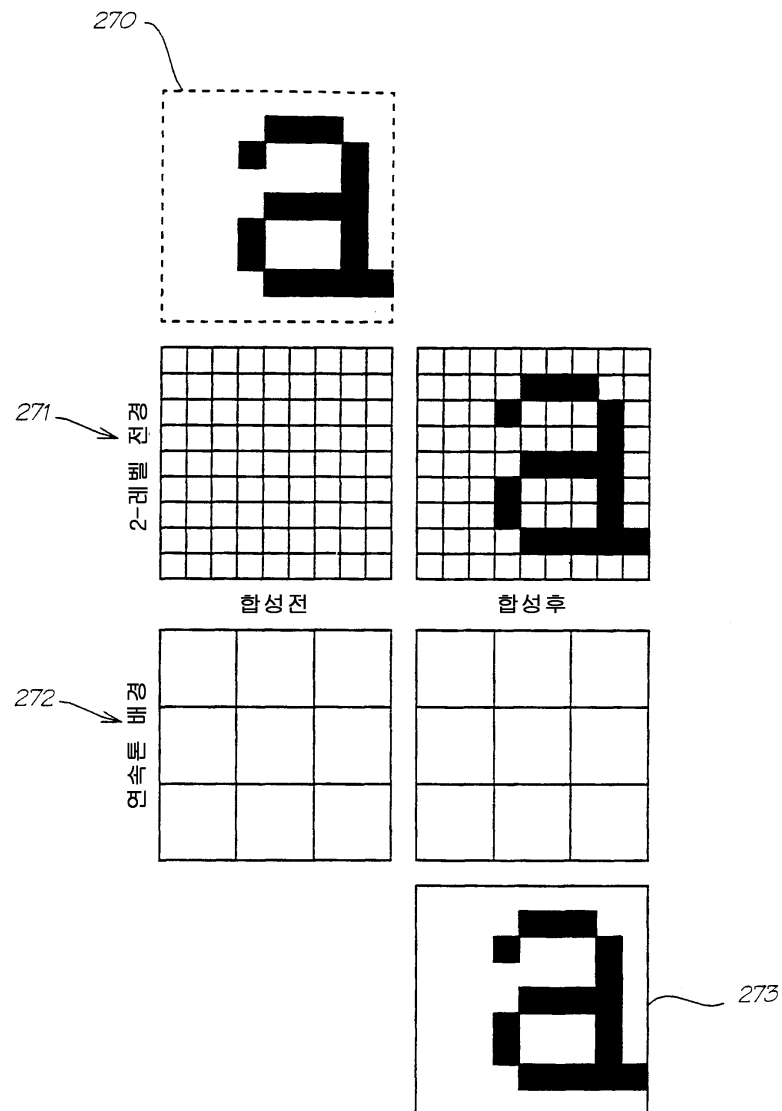
도면35



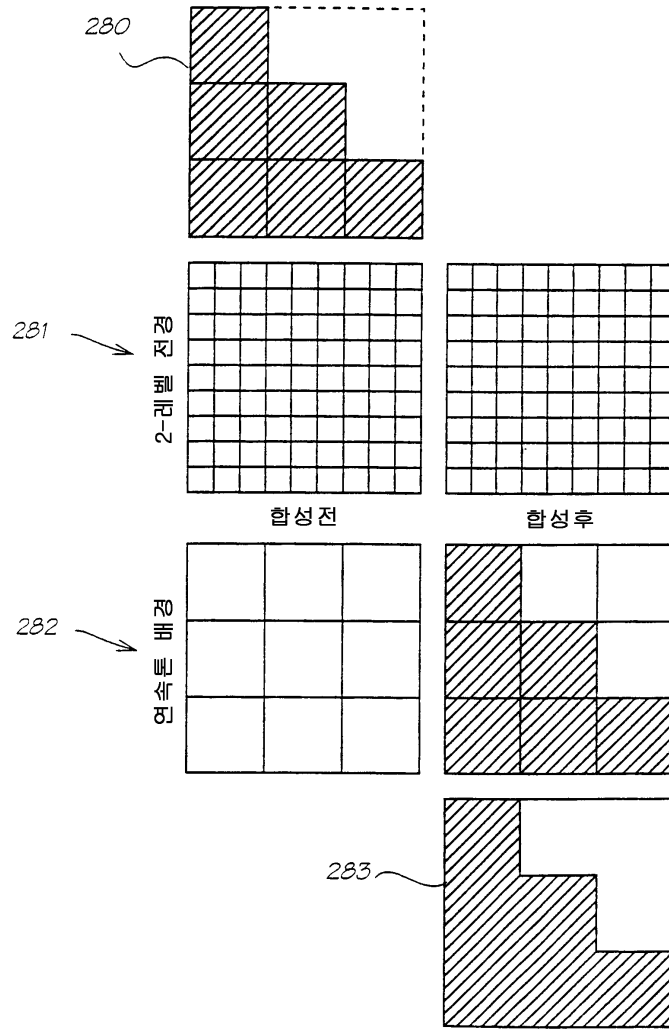
도면36



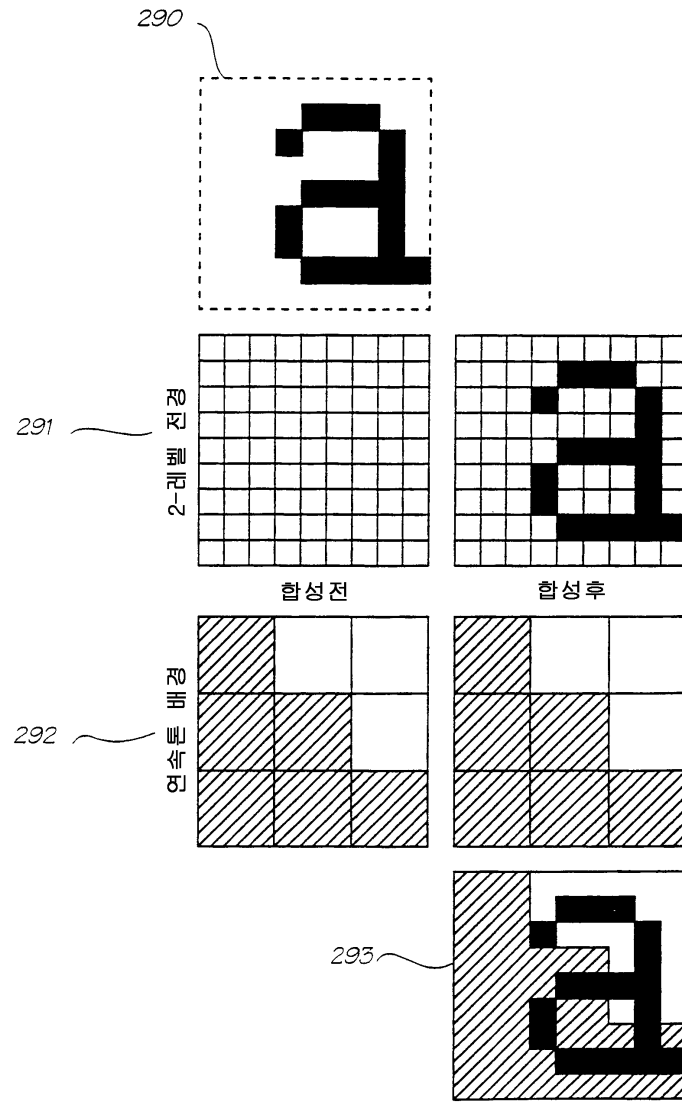
도면37



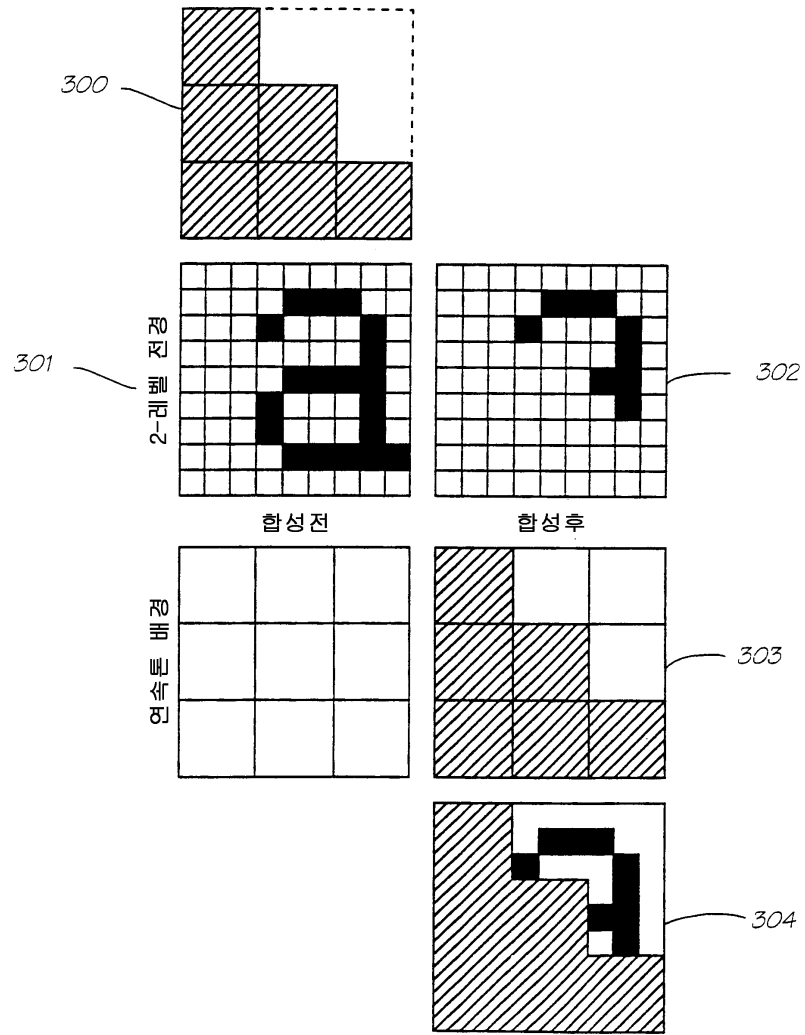
도면38



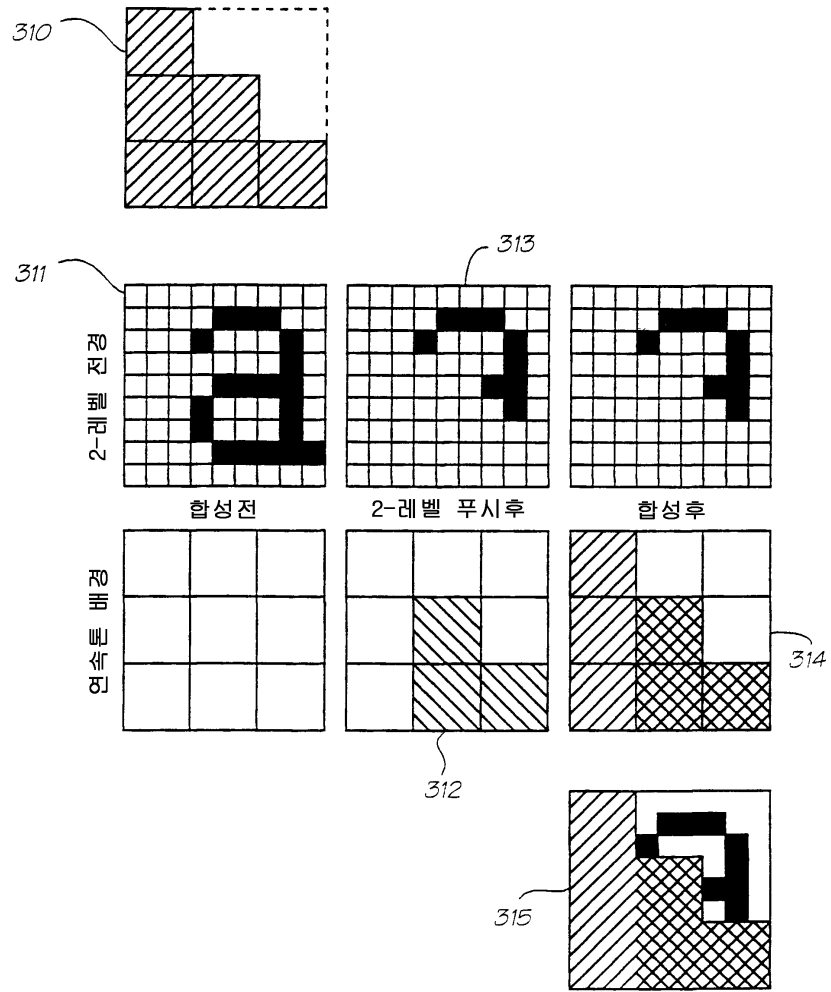
도면39



도면40



도면41



도면42

