

(19) 日本国特許庁(JP)

(12) 公表特許公報(A)

(11) 特許出願公表番号

特表2004-506254
(P2004-506254A)

(43) 公表日 平成16年2月26日(2004.2.26)

(51) Int. Cl. ⁷	F I	テーマコード (参考)
G06F 3/12	G06F 3/12 W	2C187
B41J 21/00	B41J 21/00 Z	5B021
G06F 3/03	G06F 3/03 330J	5B068

審査請求 未請求 予備審査請求 有 (全 222 頁)

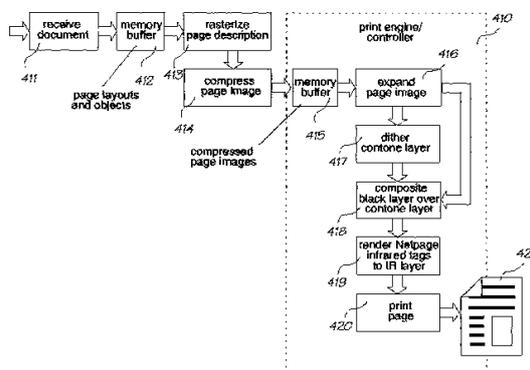
(21) 出願番号	特願2001-586056 (P2001-586056)	(71) 出願人	303024600 シルバーブルック リサーチ ピーティワイ イ リミテッド オーストラリア ニューサウスウェールズ 2041, バルメイン, ダーリング ス トリート 393
(86) (22) 出願日	平成12年5月24日 (2000.5.24)	(74) 代理人	100094318 弁理士 山田 行一
(85) 翻訳文提出日	平成14年11月25日 (2002.11.25)	(74) 代理人	100104282 弁理士 鈴木 康仁
(86) 国際出願番号	PCT/AU2000/000517	(72) 発明者	ラブソン, ポール オーストラリア, ニュー サウス ウェ ールズ州, ロッド ポイント, デュー ク アヴェニュー 13
(87) 国際公開番号	WO2001/089838		
(87) 国際公開日	平成13年11月29日 (2001.11.29)		
(81) 指定国	AP (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), EA (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), EP (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), O A (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG), AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW		

最終頁に続く

(54) 【発明の名称】 印刷ページタグエンコーダ

(57) 【要約】

印刷ページに組み込まれるタグを生産するタグ エンコーダを開示する。タグ エンコーダはタグ構造テンプレートを受信する入力部と、固定データ ビットを受信する入力部と、可変データ ビット記録を受信する入力部と、タグ構造テンプレートと固定及び可変データ ビットにより定義されたタグ内の位置に依り単一ビットを出力するタグ ジェネレータとを有する。エンコーダはリード - ソロモン符号化を利用する冗長形エンコーダを有する。タグ エンコーダはプリンタ内にある。タグ エンコーダに加えて、プリンタは圧縮連続調画像面を復号化する連続調画像デコーダと、圧縮データ内の任意の圧縮二水準画像面を復号化する二水準デコーダとを有する。これらの復号化画像面はハーフトナー/コンポジットにおいてタグ エンコーダの出力と組み合わせられ、タグ付きエリアを持つ印刷ページを生産する。タグは人間の目には見えないインクで印刷される。かかるインクはIR又はUV吸収性であり得る。



【特許請求の範囲】

【請求項 1】

タグ構造テンプレートを受信する入力部と、
 固定データ ビットを受信する入力部と、
 可変データ ビット記録を受信する入力部と、及び
 タグ構造テンプレートと前記固定及び前記可変データとにより定義されたタグ内の位置に
 依って単一のビットを出力するタグ ドット ジェネレータ、
 とを含む印刷ページ タグ エンコーダ。

【請求項 2】

任意に前記固定及び / 又は前記可変データを符号化する冗長形エンコーダを更に含む請求
 項 1 に記載の印刷ページ タグ エンコーダ。 10

【請求項 3】

冗長形エンコーダがリード - ソロモン符号化を利用する請求項 2 に記載の印刷ページ タ
 グ エンコーダ。

【請求項 4】

タグが規則的にページ上に配置される請求項 1 に記載の印刷ページ タグ エンコーダ。

【請求項 5】

タグが三角形グリッドに配置される請求項 4 に記載の印刷ページ タグ エンコーダ。

【請求項 6】

印刷されるページに対する固定データを、特定のタグ データ値と共に、被定義タグ構造
 フォーマットに従ってフォーマットするステップ、及び
 ページ上に規則的にタグを位置付けするステップ
 を含む印刷ページ タグ設定方法。 20

【請求項 7】

固定及び / 又は特定タグ データを冗帳符号化するステップを更に含む請求項 6 に記載の
 印刷ページ タグの印刷方法。

【請求項 8】

受信圧縮ページ データ内の任意の圧縮連続調画像面を復号化する連続調画像デコーダと

、
 受信圧縮ページ データ内の任意の圧縮二水準画像面を復号化する二水準デコーダと、
 タグ画像面を生産するタグ エンコーダと、及び
 画像面とタグ データ面との集積化を達成するカラー マスクにより制御されたドット合併
 ユニットの有するハーフトナー / コンポジタとを含むインク ドロップ印刷ヘッドを駆動
 するプリント エンジン / コントローラ。 30

【請求項 9】

タグが、三角形グリッド上のタグ画像面に配置される請求項 8 に記載のプリント エンジ
 ン / コントローラ。

【請求項 10】

タグ画像面が、印刷ページに赤外線インクで印刷されたタグを配置する赤外線インク チ
 ャネルに接続される請求項 8 に記載のプリント エンジン / コントローラ。 40

【請求項 11】

タグ エンコーダが、前記タグ画像面向けのタグ データを冗長符号化する請求項 8 に記載
 のプリント エンジン / コントローラ。

【請求項 12】

冗長符号化は、リード - ソロモン符号化を使用する請求項 11 に記載のプリント エンジ
 ン / コントローラ。

【請求項 13】

プリント エンジン / コントローラが、タグ データの固定及び可変部分の両方を符号化す
 る請求項 11 に記載のプリント エンジン / コントローラ。

【請求項 14】

圧縮ページ データを受信するインタフェースと、
 タグ画像面を出力するタグ エンコーダと、
 受信圧縮ページデータ内の任意の圧縮連続調画像面を復号化する連続調画像デコーダと、
 受信圧縮ページ データ内の任意の二水準画像面を復号化する二水準デコーダと、
 任意の連続調画像面又はタグ画像面を覆って任意の二水準画像面を合成するハーフトーン
 /コンポジットと、及び
 合成体を印刷ヘッドに出力する印刷ヘッド ドライバとを含む、タグ付き印刷ページを生
 成するインク ドロップ印刷ヘッドとインタフェースするプリント エンジン/コントロー
 ラ チップ。

【請求項 15】

10

圧縮ページ データを受信するインタフェースと、
 タグ画像面を生成するタグ エンコーダと、
 受信圧縮ページ データ内の任意の圧縮連続調画像面を復号化する連続調画像デコーダと
 、
 受信圧縮ページ データ内の任意の二水準画像面を復号化する二水準デコーダと、
 任意の連続調画像面又はタグ画像面を覆って任意の二水準画像面を合成するハーフトーン
 /コンポジットと、
 合成体を印刷ヘッドに出力する印刷ヘッド ドライバと、及び
 印刷ヘッドとを含む、プリント エンジン/コントローラにより駆動された、タグ付きペ
 ージを生成するインクドロップ プリンタ。

20

【発明の詳細な説明】

【0001】

本発明は、一般的にはコンピュータと対話する方法、システム、及び装置に関する。

【0002】

より具体的には、本発明は、符号化タグのフォーマットと、ページの印刷中に印刷
 ページに付加されるタグと、フォーマットに従ってタグの生産を達成するタグ エンコー
 ダとに関する。タグ エンコーダは、他のグラフィックやテキストの印刷物と一緒にタグ
 を組み入れる印刷ページを生産するプリント エンジン/コントローラ内で特に実行され
 ることが可能である。

【0003】

30

本発明は、多数の分散した利用者が、印刷物や光学センサを介してネットワークで結ばれ
 た情報と対話し、それにより対話型の印刷物をネットワークで結ばれた高速のカラー プ
 リンタを介してオンデマンドに得ることができるように主として開発された。たとえ本発
 明はこの使用に関連してここに大部分が説明されるにしても、本発明はこの分野での使用
 に限定されるものではないことは理解されるであろう。

【0004】

【同時出願】

本発明に係る種々の方法、システム、及び装置が、本発明の出願人又は譲受人が同時に
 出願した下記の同時係属出願に開示されている：

40

P C T / A U 0 0 / 0 0 5 1 8 , P C T / A U 0 0 / 0 0 5 1 9 , P C T / A U 0 0 / 0

0 5 2 0 ,
 P C T / A U 0 0 / 0 0 5 2 1 , P C T / A U 0 0 / 0 0 5 2 2 , P C T / A U 0 0 / 0

0 5 2 3 ,
 P C T / A U 0 0 / 0 0 5 2 4 , P C T / A U 0 0 / 0 0 5 2 5 , P C T / A U 0 0 / 0

0 5 2 6 ,
 P C T / A U 0 0 / 0 0 5 2 7 , P C T / A U 0 0 / 0 0 5 2 8 , P C T / A U 0 0 / 0

0 5 2 9 ,
 P C T / A U 0 0 / 0 0 5 3 0 , P C T / A U 0 0 / 0 0 5 3 1 , P C T / A U 0 0 / 0

0 5 3 2
 P C T / A U 0 0 / 0 0 5 3 3 , P C T / A U 0 0 / 0 0 5 3 4 , P C T / A U 0 0 / 0

50

0 5 3 5 ,
P C T / A U 0 0 / 0 0 5 3 6 , P C T / A U 0 0 / 0 0 5 3 7 , P C T / A U 0 0 / 0
0 5 3 8 、
P C T / A U 0 0 / 0 0 5 3 9 , P C T / A U 0 0 / 0 0 5 4 0 , P C T / A U 0 0 / 0
0 5 4 1 、
P C T / A U 0 0 / 0 0 5 4 2 , P C T / A U 0 0 / 0 0 5 4 3 , P C T / A U 0 0 / 0
0 5 4 4 、
P C T / A U 0 0 / 0 0 5 4 5 , P C T / A U 0 0 / 0 0 5 4 7 , P C T / A U 0 0 / 0
0 5 4 6 、
P C T / A U 0 0 / 0 0 5 5 4 , P C T / A U 0 0 / 0 0 5 5 6 , P C T / A U 0 0 / 0 10
0 5 5 7 、
P C T / A U 0 0 / 0 0 5 5 8 , P C T / A U 0 0 / 0 0 5 5 9 , P C T / A U 0 0 / 0
0 5 6 0 、
P C T / A U 0 0 / 0 0 5 6 1 , P C T / A U 0 0 / 0 0 5 6 2 , P C T / A U 0 0 / 0
0 5 6 3 、
P C T / A U 0 0 / 0 0 5 6 4 , P C T / A U 0 0 / 0 0 5 6 5 , P C T / A U 0 0 / 0
0 5 6 6 、
P C T / A U 0 0 / 0 0 5 6 7 , P C T / A U 0 0 / 0 0 5 6 8 , P C T / A U 0 0 / 0
0 5 6 9 、
P C T / A U 0 0 / 0 0 5 7 0 , P C T / A U 0 0 / 0 0 5 7 1 , P C T / A U 0 0 / 0 20
0 5 7 2 、
P C T / A U 0 0 / 0 0 5 7 3 , P C T / A U 0 0 / 0 0 5 7 4 , P C T / A U 0 0 / 0
0 5 7 5 、
P C T / A U 0 0 / 0 0 5 7 6 , P C T / A U 0 0 / 0 0 5 7 7 , P C T / A U 0 0 / 0
0 5 7 8 、
P C T / A U 0 0 / 0 0 5 7 9 , P C T / A U 0 0 / 0 0 5 8 1 , P C T / A U 0 0 / 0
0 5 8 0 、
P C T / A U 0 0 / 0 0 5 8 2 , P C T / A U 0 0 / 0 0 5 8 7 , P C T / A U 0 0 / 0
0 5 8 8 、
P C T / A U 0 0 / 0 0 5 8 9 , P C T / A U 0 0 / 0 0 5 8 3 , P C T / A U 0 0 / 0 30
0 5 9 3 、
P C T / A U 0 0 / 0 0 5 9 0 , P C T / A U 0 0 / 0 0 5 9 1 , P C T / A U 0 0 / 0
0 5 9 2 、
P C T / A U 0 0 / 0 0 5 8 4 , P C T / A U 0 0 / 0 0 5 8 5 , P C T / A U 0 0 / 0
0 5 8 6 、
P C T / A U 0 0 / 0 0 5 9 4 , P C T / A U 0 0 / 0 0 5 9 5 , P C T / A U 0 0 / 0
0 5 9 6 、
P C T / A U 0 0 / 0 0 5 9 7 , P C T / A U 0 0 / 0 0 5 9 8 , P C T / A U 0 0 / 0
0 5 1 6 、
P C T / A U 0 0 / 0 0 5 1 1 , P C T / A U 0 0 / 0 0 5 0 1 , P C T / A U 0 0 / 0 40
0 5 0 2 、
P C T / A U 0 0 / 0 0 5 0 3 , P C T / A U 0 0 / 0 0 5 0 4 , P C T / A U 0 0 / 0
0 5 0 5 、
P C T / A U 0 0 / 0 0 5 0 6 , P C T / A U 0 0 / 0 0 5 0 7 , P C T / A U 0 0 / 0
0 5 0 8 、
P C T / A U 0 0 / 0 0 5 0 9 , P C T / A U 0 0 / 0 0 5 1 0 , P C T / A U 0 0 / 0
0 5 1 2 、
P C T / A U 0 0 / 0 0 5 1 3 , P C T / A U 0 0 / 0 0 5 1 4 , P C T / A U 0 0 / 0
0 5 1 5

これらの同時係属出願の開示は、相互参照によりここに組み入れられる。

【 0 0 0 5 】

【 発明の背景 】

紙は情報を表示し、記録するのに広く使用されている。印刷情報はコンピュータ画面に表示された情報よりも読み易い。手書き図面や肉筆はコンピュータ キーボードやマウスを介しての入力より、より豊かな表現をもたらす。更に、紙はバッテリーでは動かず、明るい光のもとで読むことができ、コーヒーがこぼれても平気であり、持ち運びができ且つ使い捨てができる。

【 0 0 0 6 】

オンライン パブリケーションは従来の紙をベースしたパブリケーションより多くの有利な点がある。消費者の観点からすると、情報はオンデマンドで利用でき、ハイパーテキスト リンクを介して操縦され、探索され、自動的に個人の専有とすることができる。 10

【 0 0 0 7 】

出版者の観点からすると、印刷及び物理的な配達のコストが削除され、パブリケーションは、パブリケーションにお金を払う広告者にとって、パブリケーションを具体的なデモグラフィックスの対象とし、且つ商品サイトに関連付けることができるのでより魅力的となる。

【 0 0 0 8 】

オンライン パブリケーションはまた不利な点も有する。コンピュータ画面は紙より劣る。雑誌ページと同じ品質では、S V G A コンピュータ画面は約5分の1の情報だけを表示するのにすぎない。特に周囲光が強いと、C R T 及び L C D の両方とも輝度及びコントラストの問題がある、一方紙上のインクは、放射性であるよりもむしろ反射性であり、周囲光のなかでは明るく且つシャープの両方である。 20

【 0 0 0 9 】

【 発明の概要 】

1つの形態において本発明は、
タグ構造テンプレートを受信する入力部と
固定データ ビットを受信する入力部と、
可変データ ビット記録を受信する入力部と、及び
タグ構造テンプレートと前記固定及び前記可変データとにより定義されたタグ内の位置によって単一のビットを出力するタグ ドット ジェネレータと、
を含む印刷ページ タグ エンコーダに存在する。 30

【 0 0 1 0 】

本タグ エンコーダを含むプリント エンジン / コントローラは、圧縮ページ データを受信する高速シリアル インタフェースを使用することが好ましい。ページ データは、J P E G デコーダが復号する連続調画像面を含むことができ、その画像面は余白ユニットの制御によりハーフトナ / コンポジタで拡大 / 縮小できる。二水準画像面を G 4 ファクシミリ デコーダにより復号でき、また余白ユニットの制御によりハーフトナ / コンポジタで拡大 / 縮小もできる。プリント エンジン / コントローラ内の好ましい赤外線タグ エンコーダは、赤外線インクで印刷されたタグを印刷ページに配置するように画像面のプロセッシングに合わせて、ライン毎に赤外線データを生産する働きをする。 40

【 0 0 1 1 】

タグ エンコーダの目的は、タグがしばらく後で適切なペン又はそれと同等なデバイスで読み取ることができるように、タグを、印刷ページを覆って配置することである。各タグは、読み取られ、後で読むことができる2Dのデータ パッケージ（たとえタグは任意の形状の表面に印刷されてもよいが）であり得る。たとえ時には単にデータのパッケージ（我々のタグ）の存在がそれ自体情報であっても、ページに書かれることになるパッケージに何らかのデータが記憶されることが一般的である。本タグ エンコーダでは、これらの多くのデータの packets をページ上に全体にわたり書き出すことが望まれている。これらの packets の発生は、サイズや構造のいずれか又はすべてにおいて、及びどのようにデータが内側に記憶されるかに関して制御される。タグ エンコーダ及びタグ フォーマット構 50

造（以下に説明される）によりこの制御がなされる。

【0012】

タグフォーマット構造により、タグ設計者が一定のタグに対して物理的な印刷タグ構造の一部としてどのドットを印刷して、どのドットをデータから抜粋すべきかを指定することができる。タグのデータ部分は可変及び固定部に分割される。固定部はページ上のタグの各々及び全てに対して同じデータであるが、一方可変部は各タグに対して指定される。1つの限定された場合には、データが全て可変であるが偶然同じ値を有し、かくしてデータが事実上固定されることになる。プリントエンジン/コントローラ(PEC)利用者に、各々及び全てのタグに対してデータを常に供給することを強いるよりも、むしろ、各タグに対して固定データを有する可能性を許容する。正確には、なんのデータがタグにあるかは完全にアプリケーション固有のことである。あるページは、可変データとしてタグのX/Y座標と、固定データ構成要素としてページIDとを含むタグを有してよい。ページ対話型ペン(又はその相当物)は、ページ上のタグからこれらの座標をその後読み戻し、ページ上の位置により行動を遂行できるであろう。異なるページはページ全体にわたり固定データを有してよく、その結果、ペンがページ上のどこをクリックしても同じデータが戻ってくるであろう。他のページはページ上に特大のタグをすかしの形式で持っているだけでよい。タグの単なる存在で十分である。読み取りアプリケーションが読み取りタグからデータを抜粋でき、その時、そのデータが役に立つと解釈する限り、固定データ及び可変データはなんでもよい。

10

【0013】

タグの構造は利用者定義可能なものであり、種々のアプリケーションが適切な構造を構築しそれらのデータを保持することを可能にする。タグは(ペン内の)位置付けソフトウェアがタグ検出するのを助ける何らかの構造と、データビットが正確に抜粋されることを可能にするいくつかのオリエンテーション特性とを有することが理想的である。最後に、タグに埋め込まれたデータは、読み取り装置(ペン)が、埃、汚れ、泥、読み取りノイズ等による誤りを修正できるように、冗長符号化すべきである。

20

【0014】

タグは1600dpiで定義され、精密な形状をしたタグ構造が可能になる。しかしながら、現在は、各データドットが単一の印刷ドットにより表現されるページ上にデータドットを印刷することは有効ではない。読み取り環境にエラーが導入されことはあまりにも厳しすぎる結果となる。1600dpiドットを再度取り戻すことができる少なくとも3200dpi以上のスキャナをペン内に必要とする。その結果として、タグ設計者は多くの物理的な印刷ドットをページ上にクラスタ化させ、単一のデータロッドを表現することが一般的である。この印刷ドットのクラスタは、単一の論理データを表すのでマクロドットと言われ、一緒に纏められ、読み取り装置におけるドット検出及び復号化アルゴリズムを容易にする。タグフォーマット構造により、タグ内の任意の出力ドットが任意のデータビットから出ることができるので、マクロドットのサイズ及び形状は全くの任意である。タグ設計者は、ペンの読み取り及び光学的な能力をもとにしてマクロドットを設計する。

30

【0015】

タグエンコーダは、タグを風景画書式及び肖像画書式モードで印刷できるのが理想的である。タグエンコーダにより内部的に回転する単一のタグフォーマット構造は、それを行う1つのやり方であるが、我々のタグエンコーダでは、単にタグエンコーダに事前回転のタグフォーマット構造を読み取らせて、我々自身がそれを回転させる面倒をなくしている。

40

【0016】

最後に、ページ上のタグの配置に関して、タグを三角形グリッドに配置することは、インク使用の点から長方形グリッドに置くよりも良い。我々の特殊なタグエンコーダは長方形面にのみに即応するが、タグを任意の曲面に配置する時には、三角形グリッドも便利である。かくして、同じタグ対話型ペンは他の表面に印刷されたタグを読み取ることができ

50

る。

Kなどの他のインクは限られた環境でタグに使用してよいが、タグ エンコーダには、一般的にはIRインクが印刷ヘッドに存在することが必要である。

【0017】

タグ エンコーダは、どんな他の画像面が作られつつあっても、それに合わせた速度でタグ生成を実行するように作業する。タグ エンコーダは、ドット毎にタグを生成するためにタグの固定及び可変コンポーネントが供給される事前に定義されたタグ フォーマットを扱い且つ、生成画像面が合成されるにつれ、それらのタグをライン毎にコンポジットに渡すことにより速度を達成する。タグ エンコーダは印刷されるページに対する固定データを、特定の可変タグ データ値と共に誤り修正可能な符号化タグに符号化し、それはその後ページ上に通常は赤外線インク、時には黒インクで印刷される。タグ エンコーダは定期的にページ上でタグの位置付けをするのが理想的であり、タグを好ましい三角形グリッド上に配置することが理想的である。当業者は、三角形に加えて他のタグ アレイを使用することができることを認めるであろう。タグ エンコーダにより、風景画書式及び肖像画書式の両方の姿勢が可能となる。基本タグ構造は1600 dpiで描画され、一方タグ データは任意の形状のマクロドット(1600 dpiで1ドットの最小サイズを有する)として符号化される。たとえ当業者は他の方法も導き出されることを理解するにしても、出力ドットのストリームは、特殊なプリンタに合わすように設定された出力順序で作成され得る。更に、当業者は、目には見えないが適切なセンサにより検出可能な赤外線インクの使用も有利性を理解するであろうし、他のインクは時には用途があることに気付くであろう。

10

20

【0018】

既に符号化されたデータ パッケージをプリント エンジン/コントローラ(PEC)に送信する代わりに、PECに冗長符号化を行わせることにより、PECへの帯域幅は減少する。具体的説明はリード-ソロモン符号化の使用だが、PECは同様に他の任意のエンコーダでもよい。PECはタグ データの固定及び可変部の両方を符号化することが好ましい。

【0019】

本発明は、常にオフであり、常にオンであり、そして符号化データから抜粋されるドットを含む包括的データ パッケージを与えるテンプレートを定義する。これにより、種々のサイズのマクロドットと、位置付けを助ける大きなオブジェクト等を含む任意の範囲のデータ パッケージ定義が開発できることになる。タグ構造は、実行してもすべての包含するチップを組立てることを必要としない関連のDRAMに記憶してもよい。外部DRAMに代えてオンチップのタグ構造を有することが僅かな拡張にすぎない。

30

【0020】

【好ましい実施例の詳細な説明】

好ましい、及び他の本発明の実施例を、非限定的な例としてのみ、添付図面と共に説明する。

【0021】

注記: MemjetTMはシルバブルック リサーチ Pty Ltd.、オーストラリアの商標である。

40

【0022】

好適な実施形態において、本発明はネットページ ネットワークで結ばれたコンピュータシステムを取り扱うように構成され、その詳細の概観は以下に説明する。実行を全て行っても、基本システムとの関連において以下に議論された特定の詳細や拡張の全て又は大部分すらも必ずしも体现されるわけではないことは理解されよう。しかしながら、本発明の好適な実施例や態様が動作する脈絡を理解しようとする時、外部参照を行う必要を減らすためシステムを最も完全なフォームで説明する。

【0023】

概要において、好適なネットページ システムのフォームはマップ面のフォームをしたコ

50

ンピュータ インタフェース、即ち、コンピュータ システム内に維持された面のマップへの参照を含む物理面を用いる。マップ参照は適切な感知デバイスにより問いただすことができる。個々の実行によるが、マップ面上でのローカルな問合せにより明白なマップ参照がマップ内及び種々のマップ間の両方で生み出されるような方法で、マップ参照が目に見えるように又は目にみえないように符号化され、定義することができる。コンピュータ システムはマップ面に関する特徴についての情報を含み、かかる情報は、マップ面で使用された感知デバイスが供給したマップ参照をもとにして検索できる。かくして検索情報は、オペレータの面特徴との対話に応じて、オペレータに代わってコンピュータ システムが始動する行動のフォームを取ることができる。

【0024】

10

その好適なフォームにおいて、ネットページ システムはネットページの生産やネットページと人間との対話に依存する。これらは、通常の紙に印刷されたテキスト、グラフィック及び画像のページであるが、対話型ウェブ ページのように作動する。情報は、実質的に人間の肉眼では見えないインクを使用して各ページに符号化される。しかしながら、インク、及びそれによるコード化データは光学的なイメージング ペンにより感知でき、ネットページ システムに送信することができる。

【0025】

好適なフォームにおいて、各ページ上のアクティブ ボタンやハイパーリンクをペンでクリックしてネットワークからの情報を要求し、又はネットワーク サーバへ選好を信号で送ることができる。一実施例において、ネットページに手書きされたテキストはネットページ システムで自動的に認識され、コンピュータ テキストに変換されて、フォームに書き入れることができる。他の実施例において、ネットページに記録された署名は自動的に検証され、Eコマース取引を確実に認可することができる。

20

【0026】

図1に示すように、印刷ページ1は、利用者が印刷ページ上で物理的及び「電子的」の両方でペンとネットページ システムとの間の通信を介して書き入れることができる対話型フォームを表現できる。例は、名称及びアドレス フィールドと、送信ボタンを含む“要求”フォームを示す。ネットページは、目に見えるインクを使用して印刷されたグラフィック データ2と、タグ4の収集として目に見えないインクを使用し印刷されたコード化データ3とからなる。該当のページ記述5はネットページ ネットワークに記憶されているが、ネットページの個々の要素を記述する。特に、ページ記述は各対話型要素（即ち、例におけるテキスト フィールド又はボタン）のタイプ及び空間的な限界（ゾーン）を記述し、ネットページ システムがネットページを介して入力を正しく解釈できるようにする。送信ボタン6は、例えば、該当のグラフィック8の空間的な限界に対応するゾーン7を有する。

30

【0027】

図2に示すように、ネットページ ペン101は、その好適なフォームは図8及び9に示され、且つ以下により詳細に説明されるが、ネットページ プリンタ601と、家庭、オフィス又は移動使用用のインターネット接続された印刷機器との関連で作業する。ペンはワイヤレスで、短距離無線リンク9を介してネットページ プリンタと確実に通信する。

40

【0028】

ネットページ プリンタ601は、その好適なフォームは図11から13に示され、且つ以下により詳細に説明されるが、定期的に、又はオンデマンドで、個人専有となった新聞、雑誌、カタログ、パンフレットや他の出版物を、それら全ては対話ネットページとして高品質で印刷されるが、引渡すことができる。パーソナル コンピュータと違って、ネットページ プリンタは、例えば、利用者のキッチン、朝食テーブルの近く、又は家族がその日出発するポイント近くなどの、朝のニュースが最初に撮取される近辺の壁に装着できる機器である。ネットページ プリンタはまたテーブルトップ、デスクトップ及び小型バージョンでも導入される。

【0029】

50

撮取されるポイントで印刷されたネットページは、使いやすい紙を対話型媒体の即時性及び対話性と結びつけるものである。

【0030】

図2に示すように、ネットページペン101は印刷ネットページ1上でコード化データと対話し、短距離無線リンク9を介して対話をネットページプリンタに通信する。プリンタ601は対話を関連のネットページサーバ10に送り、解釈させる。適切な環境において、ページサーバは該当のメッセージをネットページアプリケーションサーバ13上で作動するアプリケーションコンピュータソフトウェアに送信する。アプリケーションサーバは、順番に、発信プリンタ上に印刷される応答を送信できる。

【0031】

ネットページシステムは、好適な実施例において、高速なマイクロエレクトロメカニカルシステム(MEMS)をベースにしたインクジェット(MemjetTM)プリンタと共に使用することにより、相当程度より便利なものになる。この技術の好適なフォームにおいて、比較的高速で且つ高品質な印刷を消費者が手に入れること可能にされる。その好適なフォームにおいて、ネットページ出版物は、操縦が容易であること及び取り扱いが快適であることの両方を一緒にした、両面をフルカラーで印刷した書簡サイズの美しいページのセット等の従来のニュース雑誌の物理的な特徴を有する。

【0032】

ネットページプリンタは、ブロードバンドインターネットアクセスの可用性が増大していることを利用する。ケーブルサービスは米国において95%の家庭で利用でき、ブロードバンドインターネットアクセスを提供するケーブルモデムサービスは既に20%の家庭で利用できる。ネットページプリンタはより遅い接続でも作動できるが、配信時間が長くなり、画質が低下する。実際に、ネットページシステムは、システムがより遅く作動し、それ故、消費者の観点からすると受け入れ難いが、現存する民生のインクジェット及びレーザープリンタを使用して可能である。他の実施例において、ネットページシステムは私設イントラネット上で主宰される。更なる他の実施例において、ネットページシステムは単一のコンピュータ又はプリンタ等のコンピュータ使用可能なデバイス上で主宰される。

【0033】

ネットページネットワーク上のネットページパブリケーションサーバ14は、印刷品質の出版物をネットページプリンタに引渡すように構成される。定期出版物は自動的にポイントキャスト及びマルチキャストインターネットプロトコルを介して予約購読ネットページプリンタに引渡される。個人の専有になった出版物は個々の利用者プロフィールに従ってフィルタをかけられ、フォーマットされる。

【0034】

ネットページプリンタは任意の数のペンをサポートするように構成でき、一のペンは任意の数のネットページプリンタで作業できる。好適な実行において、各ネットページペンはユニークな識別子を有する。一家庭はカラー付きネットページペンの収集物を有してもよく、ファミリーの各メンバーに一本のペンを割り当てる。これにより、各利用者はネットページパブリケーションサーバ又はアプリケーションサーバに関して固有のプロファイルを維持できることになる。

【0035】

ネットページペンはまたネットページ登録サーバ11に登録でき、1つ以上の決済カード口座にリンクできる。これにより、Eコマース決済がネットページを使用して確実に認可される。ネットページ登録サーバはネットページペンが捕獲した署名と事前登録の署名とを比較し、登録サーバはEコマースサーバに対して利用者の同一性を認証することができる。他のバイオメトリックスも同一性を検証するために使用できる。あるバージョンのネットページペンは指紋スキャニングを含み、それはネットページ登録サーバにより同様な方法で検証される。

【0036】

10

20

30

40

50

ネットページ プリントは利用者の介在なしで朝刊等の定期刊行物を配達可能だが、たのまれもせずにジャンクメールを絶対に配信しないように構成することができる。その好適なフォームにおいて、ネットページ プリントは予約購読した、又は別途認可されたソースからの定期刊行物を配信するのみである。この点で、ネットページ プリントは、電話番号やEメールアドレスを知っている任意のジャンクメール差出人が見ることができるファックス機、又はEメール口座と同じではない。

【0037】

1 ネットページ システム アーキテクチャ

統一モデリング言語 (UML) クラス線図を使用してシステム内の各オブジェクト モデルを説明する。クラス線図は、関係により結びつけられたオブジェクト クラスのセットからなり、2種類の関係がここでは重要である：アソシエーションと般化。アソシエーションはオブジェクト間、即ちクラスの事例間のある種類の関係を表現する。般化は現実のクラスを関連させ、次のやり方で理解することができる：あるクラスをそのクラスのオブジェクト全てのセットとして考え、クラスAがクラスBの般化であるとする、その時、Bは単にAのサブセットである。UMLは二次モデリング 即ちクラスのクラスを直接サポートしない。

10

【0038】

各クラスはクラスの名称でラベルを貼った長方形として線描される。それは水平線により名称から分離したクラスの属性のリストと、水平線により属性リストと分離したクラスの動作のリストとを含む。しかしながら、あとに続くクラス線図において、動作は決してモデル化されない。

20

【0039】

アソシエーションは2つのクラスを結合する線として描かれ、その線はいずれかの線端においてアソシエーションの多重度で任意にラベルを貼られる。初期設定多重度はその1つである。星印 (?) は“多数”の多重度、即ち零以上を表示する。各アソシエーションは任意にその名称でラベルを貼られ、また任意にいずれかの端において該当のクラスの役割でラベルを貼られる。オープン ダイヤモンドは集合アソシエーション (“の1部である”) を表示し、アソシエーション線の集積者端に描かれる。

【0040】

般化関係 (“である”) は2つのクラスを結合する実線として描かれ、般化端に矢印 (オープン三角形の形状で) が付いている。

30

【0041】

クラス線図が多数の線図に割れると、複製された任意のクラスは、クラス線図を定義するメイン線図を除く全てが破線の輪郭で示される。クラス線図は被定義場所にのみ属性と共に示される。

【0042】

1.1 ネットページ

ネットページはネットページ ネットワークが構築される基礎である。ネットページは出版情報及び対話型サービスへの紙ベースの利用者インタフェースを提供する。

【0043】

ネットページは、ページのオンライン記述への参照が目に見えないようにタグ付けされた印刷ページ (又は他の表面領域) からなる。オンライン記述はネットページ ページ サーバにより一貫して維持される。ページ記述は、テキストと、グラフィックスと、画像とを含む目に見えるページのレイアウト及び内容を記述する。ページ記述はまたボタンと、ハイパーリンクと、入力フィールドとを含むページ上の入力要素を記述する。ネットページにより、ネットページ面上にネットページ ペンで作られたマーキングをして、ネットページ システムにより同時に捕獲し、プロセスすることが可能になる。

40

【0044】

多数のネットページは同じページ記述を共有できる。しかしながら、別途同一のページを通しての入力が識別できるように、各ネットページには固有の識別子が割り当てられる。

50

このページIDの正確性は十分であり、非常に多くのネットページ間で識別される。

【0045】

ページ記述への各参照は印刷タグ内で符号化される。タグはタグ自身が現れる固有のページを識別し、それによりページ記述を間接的に識別する。タグはページ上の自分自身の位置も識別する。タグの特徴は以下により詳細に説明する。

【0046】

タグは、普通紙等の赤外線反射性のある任意の基板上に赤外線吸収性インクで印刷される。近赤外線波長は人間の目には見えないが、適切なフィルタ付きの固体画像センサにより容易に感知される。

【0047】

タグはネットページ ペン内のエリア画像センサにより感知され、タグ データは最も近いネットページ プリンタを介してネットページ システムに送信される。ペンはワイヤレスで、短距離無線リンクを介してネットページ プリンタと通信する。タグは十分に小さく、密に配置され、ペンは単一のクリックでさえも、少なくとも1つのタグをページ上に確実に描くことができる。対話は無国籍であるため、ペンがページとの対話毎にページID及び位置を認識することは重要である。タグは誤り修正可能に符号化され、タグは表面損傷には部分的には耐えられる。

10

【0048】

ネットページ ページ サーバは各印刷ページに対するユニークなページ事例を維持し、それにより、ネットページ ページ サーバが入力フィールドに対して利用者が供給した値の

20

【0049】

ページ記述と、ページ事例と、印刷ページとの間の関係は図4に示す。ページ事例は、ページ事例を印刷したネットページ プリンタと、ページ事例を要求したネットページ利用者、もし知られていれば、との両方に関連する。

【0050】

1.2 ネットページ タグ

1.2.1 タグ データ内容

好適なフォームにおいて、各タグはタグ自身が現れる領域と、領域内のそのタグの場所とを識別する。タグはまた全体として領域、又はタグに関係するフラッグを含むことができる。1つ以上のフラッグ ビットは、例えば、タグ感知デバイスに信号を送信し、感知デバイスが領域の記述を参照する必要がないように、タグの隣接エリアに関連した機能を表示するフィードバックを提供する。ネットページ ペンは、例えば、ハイパーリンクのゾーンにあると、“アクティブ エリア”LEDを点灯する。

30

【0051】

以下に、より明確に説明するが、好適な実施例において、各タグは、容易に認識される不変構造を含み、それは初期の検出を助け、且つ表面又は感知プロセスにより誘発されたいかなるゆがみの影響も最小限にすることを助ける。タグがページ全体をタイルすることが好ましく、タグは十分に小さく、密に配置され、ペンは単一のクリックでも少なくとも1つのタグをページ上に確実に描画できる。対話は無国籍であるため、ペンが、ページとの

40

【0052】

好適な実施例において、タグが参照する領域はページ全体と一致し、タグ内でコード化された領域IDは、それ故、タグが現れるページのページIDと同義である。他の実施例において、タグが参照する領域は、ページ又は他の表面の任意のサブ領域であり得る。例えば、その領域は対話型要素のゾーンと一致し、その場合、領域IDは対話型要素を直接識別できる。

表1 - タグ データ

【0053】

【表1】

50

表1

フィールド	精度 (ビット)
領域ID	100
タグID	16
フラッグ	4
合計	120

10

【0054】

各タグは120ビットの情報を含み、表1に示したように配分されるのが一般的である。平方インチあたり64の最大タグ密度と仮定すると、16ビットのタグIDは1024平方インチまでの領域サイズをサポートする。単に隣接する領域やマップを使用することにより、タグ精度を増大せずに、より大きい領域を連続的にマッピングすることができる。100ビットの領域IDにより、 2^{100} ($\sim 10^{30}$ 又は100万京)の異なる領域をユニークに識別することができる。

【0055】

1.2.2 タグ データ符号化

120ビットのタグ データは(15、5)リード-ソロモン符号を使用して冗長的に符号化される。これは各15の4ビットの記号の6コードワードからなる360符号化ビットを生み出す。(15、5)符号により、5つまでの記号誤りがコードワード当たり修正することができる、即ち、コードワード当たり33%までの記号誤り率を許容する。

20

【0056】

各4ビットの記号はタグ内に空間コヒーレント方法で表示され、6つのコードワードの記号は空間的にタグ内で差し挟まれる。これにより、バースト誤り(多数の空間的に隣接したビットに影響を与える)が全体で最小の数の記号、及び任意の1つのコードワードにおいても最小の数の記号を損傷させるにとどまり、かくしてバースト誤りを完全に修正することができる可能性を最大限にする。

【0057】

1.2.3 物理的なタグ構造

図5に示したタグの物理的な表現は、固定目標構造15、16、及び17と、可変データエリア18とを含む。固定目標構造により、ネットページ ペン等の感知デバイスがタグを検知し、その3次元のオリエンテーションをセンサとの関連において推定することができるようになる。データ エリアは符号化タグ データの個々のビットの表現を含む適切なタグ再生産を達成するため、タグは 256×256 ビットの解像力で描画される。インチ当たり1600ドットで印刷されると、これにより約4mmの直径をもつタグが生み出される。この解像力で、タグは、半径16ドットの“静かなエリア”により囲まれるように設計される。静かなエリアはまた隣接のタグにより与えられるので、タグは16ドットをタグの実効径に付加するのみである。

30

40

【0058】

タグは6つの目標構造を含む。検出リング15により、感知デバイスが最初にタグを検出できる。リングの検出は、それが回転的には不変であるため、且つそのアスペクト比を単に修正することにより透視歪の影響の大部分を除くため、容易である。オリエンテーション軸心16により、感知デバイスがセンサのヨーによりタグの近似プレーナオリエンテーションを判別できるようになる。オリエンテーション軸心は斜めになりユニークなオリエンテーションを生み出す。4つの透視目標17により、感知デバイスがタグの正確な2次元の透視変形を推定し、これによりセンサに関連したタグの正確な3次元位置及びオリエンテーションを推定できるようになる。

【0059】

50

全ての目標構造は冗長的に大きく、目標構造のノイズに対する免疫性を向上させている。

【0060】

全体的なタグ形状は円形である。このことは、なかんずく、不規則な三角形グリッド上で最適なタグパッキングをサポートする。このことは、円形検出リングとの組合せにより、タグ内のデータビットの円形配列を最適にする。データビットのサイズを最大にするため、各データビットは、2つの放射状線と、2つの同心の円形アークとに接するエリアの形状で放射状の楔により表現される。各楔は1600dpiで8ドットの最小ディメンジョンを有し、楔のベース(楔の内部アーク)が少なくともこの最小ディメンジョンと同じであるように設計される。放射状方向の楔の高さは常に最小ディメンジョンと同じである。各4ビットのデータ記号は2x2楔のアレイにより表現される。

10

【0061】

6つのコードワードの各々の15個の4ビットのデータ記号は、インタリーブリング様式で4つの同心円記号リング18aから18dに配分される。記号はタグの廻りに円形状の進行で交互に配分される。

【0062】

インタリーブリングは同じコードワードの任意の2つの間の平均空間距離を最大にするように設計される。

【0063】

タグ付き領域との“単一クリック”の対話を感知デバイスを介してサポートするため、感知デバイスは、タグが領域のどこにあってはまたいかなるオリエンテーションで位置しても、その視野内で少なくとも1つのタグ全体を見ることができなければならない。感知デバイスの所要の視野径は、それ故、タグのサイズ及びスペーシングの関数である。

20

【0064】

円形タグ形状を仮定すると、図6に示したように、センサ視野の最小径は、タグが等辺三角形グリッド上でタイルされる時に得られる。

【0065】

1.2.5 タグ画像プロセッシング及び復号化

ネットページ等の感知デバイスにより実行されたタグ画像プロセッシング及び復号化を図7に示す。捕獲画像は画像センサから取得されつつあり、一方その画像のダイナミックレンジは(20で)判別される。その時、レンジの中心は画像21用の2進閾値として選択される。その時、画像は閾値化され、接続画素領域(即ち、形状23)に分割される(22において)。小さすぎてタグ目標構造を表現できない形状は捨てられる。各形状のサイズ及び重心もまた算出される。

30

【0066】

2進形状モーメント25が、その時、各形状に対して算出され(24において)、これらは以降に目標構造の位置付けをする基礎を提供する。中心形状モーメントは、本来は、位置については不変であり、拡大/縮小と、アスペクト比、回転とについて容易に不変にすることができる。

【0067】

リング目標構造15は最初に位置付けされるものである(26において)。リングは、透視歪みが生じると、非常に良好に行動するという有利な点を有する。マッチングは、各形状のモーメントをアスペクト正規化し、且つ回転正規化することにより進行する。一旦その2次モーメントが正規化されると、たとえ透視歪みが大きくなっていても、リングを認識することは容易である。リングの本来のアスペクト及び回転27はともに透視変形を有効的に近似値化する。

40

【0068】

軸心目標構造16は次に位置決めされるものである(28において)。マッチングは、リングの正規化を各形状のモーメントに、回転正規化をその結果生じるモーメントにそれぞれ適用することにより進行する。一旦その2次モーメントが正規化されると、軸心目標は容易に認識される。軸心の2つの可能性のあるオリエンテーションを明確にするために、

50

三分の一次元モーメントが必要であることに注目せよ。形状は、このことを可能にするため、故意に片側に傾斜する。透視歪みは軸心目標の軸心を隠すことができるので、軸心目標を回転正規化することは、軸心目標にリングの正規化が適用された後にのみ可能であることにもまた注目せよ。軸心目標の本来の回転は、ペンのヨー29によるタグの回転を有効的に近似値化する。

【0069】

4つの透視目標構造17は最後に位置付けされるものである(30において)。それらの位置の良好な推定は、リング及び軸心目標と、リングのアスペクト及び回転と、軸心の回転との既知の空間的關係をもとにして算出される。マッチングは、リングの正規化を各形状のモーメントに適用することにより進行する。一旦、それらの2次モーメントが正規化されると、円形の透視目標を認識することは容易であり、各推定位置に最接近した目標をマッチとする。4つの透視目標の本来の重心は、その時、タグ空間内の、既知のサイズの四角形の透視歪みが生じたコーナー31であるとし、4つのタグスペースポイントと画像スペースポイントとの対に係る十分に理解された式を解くことにより、8つの自由度を持つ透視変形が推定される(32において)。

10

【0070】

タグスペースから画像スペースへの推定された透視変形を使用して、タグスペース内の各既知のデータビット位置を画像スペースに投影し(36において)、その画像スペースでは実数値の位置を使用して、入力画像で4つの関連する隣接画素の補間が双一次的に行われる(36において)。事前算出の画像閾値21を使用して、その結果を閾値化し最終ビット値37を生産する。

20

【0071】

一旦360のデータビット37全てがこの方法で得られると、6つの60ビットのリードソロモンコードワードの各々を復号化し(38において)、全部で20個の復号化ビット39、又は120個の復号化ビットを生み出す。コードワード記号はコードワード順にサンプリングされ、その結果コードワードはサンプリングプロセス中に暗黙にインタープリングが解かれることに注目せよ。

【0072】

リング目標15は、画像のサブエリアにおいてのみ求められ、サブエリアの画像との関係によりリングは、もし見つければ、完全なタグの一部であることが保証される。完全なタグが見つからず、且つ首尾よく復号化されることがなければ、その時、ペン位置は現状フレームに対して記録されない。十分なプロセッシングパワー及び理想的には非最小視野193が与えられれば、代替の戦術において現状画像の他のタグの捜索に関与する。

30

【0073】

取得タグデータは、タグ含む領域及び領域内のタグの位置の識別を表示する。領域内のペン先の正確な位置35、並びにペンの全体的なオリエンテーション35が、その時、タグ上で観察された透視変形33と、ペンの物理的な軸心とペンの光学的な軸心との既知の空間的關係から推定される(34において)。

【0074】

1.2.6 タグマップ

タグの復号化は、領域IDと、タグIDと、タグ関連のペン変形とをもたらし。領域内のタグの場所は、タグID及びタグ関連のペン場所をタグ付きの領域内の絶対場所に翻訳する前に、知らなければならない。これはタグマップにより与えられ、タグ付き領域の各タグIDを該当の場所にマッピングする関数である。タグマップクラス線図は、ネットページプリンタクラス線図の一部として図22に示される。

40

【0075】

タグマップは、表面領域をタグでタイルするのに使用されたスキームを反映し、これは表面タイプに従って変わることができる。多数のタグ付き領域が同じタイリングスキーム及び同じタグナンバリングスキームを共有する時、それらの領域はまた同じタグマップも共有できる。

50

【0076】

領域に対するタグ マップは領域 ID を介して検索可能でなければならない。かくして、領域 ID と、タグ ID と、ペン変形とが与えられれば、タグ マップは検索でき、タグ ID は領域内の絶対的なタグ場所に翻訳でき、タグ関連のペン場所はタグ場所に付加されて、領域内の絶対的なペン場所を生み出すことができる。

【0077】

1.2.7 タグ付けスキーム

2つの別個の表面コード化スキームが重要であり、その両方はこのセクションで先に説明したタグ構造を使用する。好適なコード化スキームは既に論議したように“場所表示”タグを使用する。代替のコード化スキームはオブジェクト表示タグを使用する。

10

【0078】

場所表示タグはタグ ID を含み、そのタグ ID はタグ付き領域に関連したタグ マップを通して翻訳されると、領域内でユニークなタグ場所を生み出す。ペンのタグ関連の場所をこのタグ場所に付加し、領域内のペンの場所を生み出す。これを順番に使用して、領域に関連したページ記述において利用者インタフェース要素に係るペンの場所を判別する。利用者インタフェース要素自体が識別されるのみならず、利用者インタフェース要素に係る場所も識別される。それ故、場所表示タグが特殊な利用者インタフェース要素のゾーンにおいて絶対ペン経路の捕獲をサポートすることは些細なことである。

【0079】

オブジェクト表示タグはタグ ID を含み、そのタグ ID は領域に関連したページ記述において利用者インタフェース要素を直接識別する。利用者インタフェース要素のゾーンにあるタグ全てが利用者インタフェース要素を識別し、それらのインタフェース要素が全て同一になり、それ故、区別がつかなくする。それ故、オブジェクト表示タグは絶対ペン経路の捕獲はサポートしない。しかしながら、オブジェクト表示タグは相対ペン経路の捕獲はサポートする。位置サンプリング頻度が出あいタグ頻度の倍を超える限り、一方のサンプリングされたペン位置から次のペン位置へのストローク内の置換は明白に判別できる。

20

【0080】

いずれのタグ付けスキームでも、タグは、感知デバイスがタグ データを読み取るために且つネットページ システムにおいて適切な応答を生成するために、利用者が適切な感知デバイスを使用して印刷ページと対話できるという点において、利用者インタフェース要素としてネットページ上で関連の目視要素と協力して機能する。

30

【0081】

1.3 ドキュメント及びページ記述

ドキュメント及びページ記述クラス線図の好適な実施例を図 25 及び 26 に示す。

【0082】

ネットページ システムにおいて、ドキュメントは3つのレベルで記述される。最も抽象的レベルでは、ドキュメント 836 は階層的な構造を有し、その端末要素 839 はテキスト オブジェクト、テキスト スタイル オブジェクト、画像オブジェクト等の内容オブジェクト 840 に関連する。一旦ドキュメントが特定のページサイズで且つ特定の利用者の拡大/縮小率の選好に従ってプリンタで印刷されると、ドキュメントにはページ数を付けられ、別途フォーマットされる。フォーマット端末要素 835 は、ある場合には、それらの該当する端末要素と関連したオブジェクトと異なる他の内容オブジェクトと関連し、特にそこでは内容オブジェクトがスタイル関連である。ドキュメントやページの各印刷事例はまた別個に記述され、特殊なページ事例 830 を通して捕獲された入力と同じページ記述の他の事例を通して捕獲された入力とは別個に記録することができる。

40

【0083】

最も抽象的なドキュメント記述がサーバ上に存在していることにより、利用者はソースドキュメント固有のフォーマットの受理を強いられずにドキュメントのコピーを要求できる。例えば、利用者は異なるページサイズのプリンタを通してドキュメントのコピーを要求しているかもしれない。逆に、フォーマット ドキュメント記述がサーバ上に存在して

50

いることにより、ページサーバが特殊な印刷ページ上の利用者行動を効率的に解釈できることになる。

【0084】

フォーマットドキュメント834は、各々がフォーマット化端末要素835のセットからなるフォーマットドキュメント記述5のセットからなる。各フォーマット化要素は空間的な広がり、即ちゾーン58をページ上で有する。これによりハイパーリンクや入力フィールドなどの入力要素のアクティブエリアが定義される。

【0085】

ドキュメント事例831はフォーマット化ドキュメント834に対応する。ドキュメント事例は、各々がフォーマット化ドキュメントのページ記述5に対応するページ事例830のセットからなる。各ページ事例830は単一のユニークな印刷ネットページ1を記述し、ネットページのページID50を記録する。ページ事例は、分離して要求されたページのコピーを表現しても、ドキュメント事例の一部ではない。

10

【0086】

ページ事例は端末要素事例832のセットからなる。要素事例は、要素事例が事例固有の情報を記録する場合のみに存在する。かくして、ハイパーリンク事例は、ハイパーリンク要素がページ事例に特有の取引ID55を記録するので、ハイパーリンク要素に対して存在し、フィールド事例は、フィールド要素がページ事例に固有の入力を記録するので、フィールド要素に対して存在する。しかしながら、要素事例はテキストフローなどの静的要素に対しては存在しない。

20

【0087】

図27に示したように、端末要素は、静的要素843、ハイパーリンク要素844、フィールド要素845、又はページサーバコマンド要素846であり得る。図28に示したように、静的要素843は、関連のスタイルオブジェクト854付きのスタイル要素847、関連のスタイルテキストオブジェクト855付きのテキストフロー要素848、関連の画像オブジェクト856付きの画像要素849、関連のグラフィックオブジェクト857付きのグラフィック要素850、関連のビデオクリップオブジェクト858付きのビデオクリップ要素851、関連のオーディオクリップオブジェクト859付きのオーディオクリップ要素852、又は関連のスクリプトオブジェクト860付きのスクリプト要素853であり得る。

30

【0088】

ページ事例は背景フィールド833を有し、特定の入力要素に適用されないページ上に捕獲された任意のデジタルリンクを記録する。

【0089】

発明の好適な形態において、タグマップ811は各ページ事例に関連して、ページ上のタグをページ上の場所に翻訳することができる。

【0090】

1.4 ネットページネットワーク

図3に示したように、好適な実施例において、ネットページネットワークは、インターネットなどのネットワーク19を介して接続された、ネットページサーバ10と、ネットページ登録サーバ11と、ネットページIDサーバ12と、ネットページアプリケーションサーバ13と、ネットページパブリケーションサーバ14と、ネットページプリンタ601との分散セットからなる。

40

【0091】

ネットページ登録サーバ11は、利用者と、ペンと、プリンタと、アプリケーションと、パブリケーションとの関係を記録し、それにより種々のネットワー活動を認可するサーバである。ネットページ登録サーバは、利用者を認証し、アプリケーション取引において認証利用者に代わって署名代理人として活動する。ネットページ登録サーバはまた肉筆認識サービスを提供する。上に記載したように、ネットページサーバ10はページ記述やページ事例についての一貫した情報を維持する。ネットページネットワークは各々がペー

50

ジ事例のサブセットを取り扱う任意の数のページサーバを含む。ページサーバはまた各ページ事例に対する利用者入力値を維持するので、ネットページプリンタなどのクライアントはネットページ入力を適切なページサーバに直接送信する。ページサーバは該当のページの記述に係る任意の入力を解釈する。

【0092】

ネットページIDサーバ12は、複数のドキュメントID51をオンデマンドで割り当て、そのID割り当てスキームを介して複数のページサーバの負荷がバランスするようにする。

【0093】

ネットページプリンタは、インターネット分散ネームシステム(DNS)、又は類似のものを使用し、ネットページページID50を該当のページ事例を取り扱うネットページページサーバのネットワークアドレスに分解する。

10

【0094】

ネットページアプリケーションサーバ13は、対話型ネットページアプリケーションをホストするサーバである。ネットページパブリケーションサーバ14はネットページドキュメントをネットページプリンタに出版するアプリケーションサーバである。これらはセクション2で詳細に説明する。

【0095】

ネットページサーバは、IBM、ヒューレットパッカード、サンなどの製造者からの種々のネットワークサーバプラットフォーム上でホストされることが可能である。多数のネットページサーバは同時に単一のホスト上で作動でき、単一のサーバを多くのホスト上に分散することが可能である。ネットページサーバにより提供された機能性のいくらか又は全てを、及び、特にIDサーバやページサーバにより提供された機能性もまた、コンピュータワークステーションにおいて、又はローカルネットワーク上で、ネットページプリンタなどのネットページ機器で直接提供することが可能である。

20

【0096】

1.5 ネットページプリンタ

ネットページプリンタ601はネットページシステムで登録され、ネットページドキュメントをオンデマンドで且つ予約購読を介して印刷する機器である。各プリンタは固有のプリンタID62を有し、インターネット等のネットワークを介して、理想的にはブロードバンド接続を介してネットページネットワークに接続される。

30

【0097】

不揮発性メモリ内での識別やセキュリティ設定を除いて、ネットページプリンタは永続性の記憶を含まない。利用者に関する限り、“ネットワークはコンピュータである”。ネットページは分散ネットページページサーバ10の助けを借りて、特定のネットページプリンタとは無関係に、スペース及び時間を横断して対話的に機能する。

【0098】

ネットページプリンタは予約購読のネットページドキュメントをネットページパブリケーションサーバ14から受信する。各ドキュメントは2つの部分に配分される：ページレイアウト及びページに居住する実際のテキスト及び画像オブジェクト。個人の専有化のため、ページレイアウトは特定の加入者に固有なものになるのが一般的であり、それで、適切なページサーバを介して加入者のプリンタにポイントキャストされる。一方、テキスト及び画像オブジェクトは他の加入者と共有されるのが一般的であり、それで、全ての加入者のプリンタ及び適切なページサーバにマルチキャストされる。

40

【0099】

ネットページパブリケーションサーバは、ドキュメント内容をポイントキャスト及びマルチキャストに最適に分割する。ドキュメントのページレイアウトのポイントキャストを受信後、プリンタはどのマルチキャストに、もしあれば、聞くべきかを知る。

【0100】

プリンタは、印刷すべきドキュメントを定義する完全なページレイアウト及びオブジェ

50

クトを一旦、受信すると、プリンタはそのドキュメントを印刷できる。

【0101】

プリンタはシートの両面で奇数及び偶数のページを同時にラスタ化し、印刷する。プリンタは複式のプリント エンジン コントローラ760と、この目的のためのMemjetTM印刷ヘッド350を利用するプリント エンジンとを含む。

【0102】

印刷プロセスは2つの分離したステージからなる：ページ記述のラスタ化及びページ画像の拡大及び印刷。ラスタ画像プロセッサ(RIP)は並列に作動する1つ以上の標準DSP757からなる。複式のプリント エンジン コントローラは、プリント エンジン内の印刷ヘッドの動作に同期する、リアルタイムでページ画像を拡大し、ディザリングし、印刷するカスタム プロセッサからなる。

10

【0103】

IR印刷が可能ではないプリンタは、IR吸収性の黒インクを使用して、さもなければページのエンブティ エリアにタグを制限するにしても、タグを印刷する選択肢を有する。かかるページはIR印刷ページより機能性がより制限されるが、依然ネットページとして区分される。

【0104】

通常のネットページ プリンタは紙のシート上にネットページを印刷する。より特殊化されたネットページ プリンタは、球状のものなどのより特殊化された表面上に印刷できる。各プリンタは少なくとも1つの表面タイプをサポートし、且つ各表面タイプに対して少なくとも1つのタグ タイリング スキームを、それによりタグ マップを、サポートする。ドキュメントを印刷するのに実際に使用されたタグ タイリング スキームを記述するタグ マップ811は、そのドキュメントと関連するようになり、その結果、ドキュメントのタグは正しく解釈される。

20

【0105】

図2はネットページ プリンタ クラス線図を示し、ネットページ ネットワーク上の登録サーバ11により維持されたプリンタ関連の情報を反映する。

【0106】

ネットページ プリンタの好適な実施例については、図11から16を参照して、以下のセクション6でより詳細に説明する。

30

【0107】

15.1.1 MemjetTM 印刷ヘッド

ネットページ システムは、サーマル インクジェット、圧電インクジェット、レーザ電子写真等を含む広い範囲のデジタル印刷技術で作られたプリンタを使用して作動できる。しかしながら、幅広い消費者に受け入れられるには、ネットページ プリンタが下記の特徴を有することが望ましい：

- ・ 写真品質のカラー印刷
- ・ 高品質のテキスト印刷
- ・ 高信頼性
- ・ 低プリンタ コスト
- ・ 低インク コスト
- ・ 低紙コスト
- ・ 簡単な操作
- ・ 殆ど無音に近い印刷
- ・ 高印刷速度
- ・ 同時両面印刷
- ・ コンパクトな形状係数
- ・ 低消費電力

40

商業的に利用できる印刷技術でこれらの特徴の全てを有しているものはない。

【0108】

50

これらの特徴を持つプリンタの生産を可能にするため、本出願人は、MemjetTM技術と称する新しいプリント技術を発明した。MemjetTMは、マイクロエレクトロメカニカルシステム(MEMS)技術を使用して製造されたページ幅の印刷ヘッドを組み込むドロップオンデマンドインクジェット技術である。図17はMemjetTM印刷ヘッドの単一の印刷要素300を示す。ネットページ壁プリンタは168960個の印刷要素300を組み込み、1600dpiページ幅複式プリンタを形成する。このプリンタはシアンインクと、マゼンタインクと、黄インクと、黒インクと赤外線インクと、並びに紙コンディショナ及び色留め剤を同時に印刷することができる。

【0109】

印刷要素300は約長さ110ミクロン、幅32ミクロンである。これらの印刷要素のレイはCMOSロジックと、データトランスファと、タイミングと、駆動回路とを組み入れるシリコン基板上に形成される(図示せず)。

【0110】

印刷要素300の主要な要素はノズル302と、ノズルリム303と、ノズルチャンバ-304と、流体シール305と、インクチャンネルリム306と、レバーアーム307と、能動アクチュエータビーム対308と、受動アクチュエータビーム対309と、能動アクチュエータアンカー310と、受動アクチュエータアンカー311と、インクインレット312とである。

【0111】

能動アクチュエータビーム対308は、ジョイント319において受動アクチュエータビーム対309に機械的に結合される。両ビーム対ともビーム対のそれぞれのアンカーポイント310及び311で固定される。要素308、309、310、311、及び319の組合せにより片持ち梁式電熱バンドアクチュエータ320が形成される。

【0112】

図18は、印刷要素300の断面部315を含む、印刷要素300のレイの小部分を示す。

【0113】

断面部315は、シリコンウエーハ301を通過するインクインレット312を明確に示すためにインク無しで示される。

【0114】

図19(a)、19(b)、19(c)はMemjetTM印刷要素300の動作サイクルを示す。

【0115】

図19(a)はインクドロップレットを印刷する前のインクメニスカス316の静止した位置を示す。インクは、インクメニスカス316における、及びノズルチャンバ-とインクチャンネルリム306との間で形成された流体シール305における表面張力により、ノズルチャンバ-内に保持される。

【0116】

印刷時、印刷ヘッドCMOS回路はデータをプリントエンジンコントローラから正しい印刷要素へ分配し、データをラッチし、データをバッファして能動アクチュエータビーム対308の電極318を駆動する。これにより電流がビーム対308を約1マイクロ秒間流れ、ジュール加熱をもたらす。ジュール加熱に起因する温度上昇により、ビーム対308が拡大する。受動アクチュエータビーム対309は加熱されないため、拡大せず、2つのビーム対間で応力の差が生じる。この応力の差は、基板301の方に曲がっている電熱バンドアクチュエータ320の片持ち梁端により部分的に解決される。レバーアーム307はこの動きをノズルチャンバ-304に伝達する。ノズルチャンバ-304は約2μm、図19(b)に示した位置に移動する。これによりインク圧が増加し、インク321をノズル302から強制的に排出し、且つインクメニスカス316が膨れることになる。ノズルリム303は、インクメニスカス316がノズルチャンバ-304の表面を横切って広がるのを防ぐ。

【0117】

ビーム対308及び309の温度が均等になるにつれ、アクチュエータ320はその本来の位置に戻る。これにより、図19(c)に示したように、インクドロップレット317がノズルチャンバー内のインク321から絶縁するのを助ける。ノズルチャンバーはインクメニスカス316における表面張力の作用により再充填される。

【0118】

図20は印刷ヘッド350のセグメントを示す。ネットページプリンタにおいて、印刷ヘッドの長さは方向351における紙の全幅(210mmが一般的)である。示したセグメントは長さ0.4mm(完全な印刷ヘッドの約0.2%)である。印刷時、紙は方向352で固定印刷ヘッドを通過する。印刷ヘッドは6列の嵌合した印刷要素300を有し、

10

【0119】

作動中の脆弱な印刷ヘッドの表面を保護するため、ノズルガードウエーハ330が印刷ヘッド基板301に装着される。各ノズル302に対して、インクドロップレットが発射される該当のノズルガード穴331がある。ノズルガード穴331が紙繊維又は他の破片によりブロックされるのを防ぐため、印刷中、濾過空気が、空気インレット332を通してノズルガード穴から吸い出される。インク321の乾燥を防ぐため、プリンタが空運転されている時、ノズルガードがシールされる。

【0120】

1.6 ネットページ ペン

ネットページシステムのアクティブ感知デバイスはペン101であるのが一般的であり、ペンは埋め込みコントローラ134を使用して、画像センサを介してページからのIR位置タグを捕獲し、復号化できる。画像センサは、近赤外線波長でのみ感知を可能にする適切なフィルタ付きで提供された固体デバイスである。以下により詳細に説明したように、システムは先端が表面と接触している時に感知でき、ペンはタグを十分な速さで感知し、人間の肉筆(即ち、200dpi以上、及び100Hz以上)を捕獲することができる。ペンにより捕獲された情報は暗号化され、無線でプリンタ(又は基地局)に送信され、プリンタ又は基地局は(既知の)ページ構造に関してデータを解釈する。

20

【0121】

ネットページペンの好適な実施例は、通常のマーキングインクペン及び非マーキング針の両方として作動する。しかしながら、マーキング態様は、ペンをインターネットインタフェースとして使用する時など、ブラウジングシステムとしてネットページシステムを使用するのに必要ではない。各ネットページペンはネットページシステムで登録され、固有のペンID61を有する。図23はネットページペンクラス線図を示し、ネットページネットワーク上の登録サーバ11で維持されたペン関連の情報を反映する。

30

【0122】

いずれかの先端がネットページと接触していると、ペンはページとの関連においてその位置及びオリエンテーションを判別する。先端は力センサに装着され、先端の力が、ペンが“アップ”又は“ダウン”かどうか表示する閾値との関連において解釈される。これにより、例えて言えば、ネットワークからの情報を要求するために、ペン先で押すことにより

40

【0123】

ペンは、その先端近辺にあるページのエリア193を赤外線スペクトルでイメージングすることにより、ネットページ上でその先端の位置及びオリエンテーションを判別する。ペンは最も近いタグを復号化し、タグに係る先端の位置をイメージングされたタグ上で観察された透視歪み及びペン光学の既知の幾何学から算出する。ページ上のタグ密度がタグサイズと逆比例するので、タグの位置解像力は低いかもしれないが、調整された位置解像力はきわめて高く、正確な肉筆認識に必要な最小解像力を超えている。

【0124】

50

ネットページに係るペンの行動は一連のストロークとして捕獲される。ストロークはページ上の時刻印のペン位置のシーケンスからなり、ペンダウン事象から始まり、以降のペンアップ事象により完了する。タグIDが、通常の状態ではストロークの開始で変わるが、変わる毎にストロークにはまたネットページのページID 50でタグ付けされる。

【0125】

各ネットページ ペンは、それに関連した現状選択 826を有し、利用者がコピーやペー
スト操作等を行うことができるようになる。選択は、システムが被定義の期間後選択を捨
てることができるように時刻印される。現状選択はページ事例の領域を記述する。現状
選択は、ページの背景エリアに係るペンを通して捕獲された最新のデジタル インク ス
トロークからなる。一旦現状選択が選択ハイパーリンク起動を介してアプリケーションに
配信されると、現状選択はアプリケーション固有の方法で解釈される。

10

【0126】

各ペンは現状先端 824を有する。これはペンにより最後にシステムに通知された先端で
ある。上記したデフォルト ネットページ ペンの場合、マーキング黒インク先端又は非マ
ーキング 針先端のいずれかが現状のものである。各ペンはまた現状先端スタイル 825
を有する。これはアプリケーションにより、例えば、パレットからカラーを選択する利用
者に対応してペンに最後に関連した先端スタイルである。初期設定 先端スタイルは現状
先端に関連した先端スタイルである。ペンを通して捕獲されたストロークは、現状先端
スタイルでタグ付けされる。ストロークが以降再生産されると、ストロークは、ストローク
がタグ付けされる先端スタイルで再生産される。

20

【0127】

ペンが通信できるプリンタの範囲内にあるときはいつでも、ペンはゆっくりとその“オン
ライン”LEDを点滅する。ペンがページに係るストロークを復号化するのに失敗すると
、ペンはその“誤り”LEDを瞬間に起動させる。ペンがページに係るストロークを復号
化するのに成功すると、ペンはその“OK”LEDを瞬間に起動させる。

【0128】

一連の捕獲されたストロークはデジタル インクと称する。デジタル インクは、肉筆
のオンライン認識のため、及び署名のオンライン検証のための、線描や肉筆をデジタル
交換する基礎を形成する。

【0129】

ペンはワイヤレスで、短距離無線リンクを介してデジタル インクをネットページ プリ
ンタに送信する。送信されたデジタル インクは、プライバシー及びセキュリティのため
暗号化され、効率的な送信のためパケット化されているが、プリンタ内の取扱を時宣を得
て保証するため、ペンアップ事象では常に流される。

30

【0130】

ペンがプリンタの範囲外であれば、ペンはデジタル インクを10分を超える連続した
手書きの容量を有する内部メモリにバッファする。ペンが再度プリンタの範囲内に入れば
、ペンは任意のバッファリングされたデジタル インクを転送する。

【0131】

ペンは任意の数のプリンタに登録することができるが、状態データ全てが紙上及びネット
ワーク両方の上でネットページ内にあるので、ペンが任意の特定の時間にどのプリンタと
通信するかは全く重要ではない。

40

【0132】

ペンの好適な実施例については、図8から10を参照して、以下のセクション6で更に詳
細に説明する。

【0133】

1.7 ネットページ対話

ペンを使用してネットページ1と対話すると、ネットページ プリンタ601はペン10
1からのストロークに係るデータを受信する。ペンを使用してストロークなどの動きを実
行すると、タグ4のコード化データはペンにより読み出される。データにより、特定のペ

50

ージ及び関連の対話型要素の身元を判別し、且つページに係るペンの相対位置の表示を得ることができる。表示データはプリンタに送信され、そこでプリンタはストロークのページID 50を、DNSを介して、該当のページ事例830を維持するネットページサーバ10のネットワークアドレスに分解する。その時、プリンタはストロークをページサーバに送信する。ページがより早いストロークで最近識別されていれば、その時には、プリンタは関連のページサーバのアドレスをそのキャッシュに既に有するであろう。各ネットページはネットページサーバ(下記参照)により一貫して維持されたコンパクトなページレイアウトからなる。ページレイアウトは、一般的にはネットページ上の他のどこかに記憶されている画像、フォント、テキストのピースなどのオブジェクトを参照する。

10

【0134】

ページサーバがペンからストロークを受信すると、ページサーバはストロークが適用されるページ記述を検索し、ストロークがページ記述のどの要素に交差するかを判別する。その時、ページサーバは関連要素のタイプの文脈においてストロークを解釈できる。

【0135】

“クリック”はストロークであり、ここではペンドアウン位置と以降のペンアップ位置との間の距離及び時間が両方ともいくらかの小さい最大値未満である。クリックにより起動されるオブジェクトには一般的にクリックを起動することが必要であり、それ故、長いストロークは無視される。“いいかげんな”クリックなどの、ペン活動の登録の失敗はペンの“OK”LEDからの応答の欠如により表示される。

20

【0136】

ネットページページ記述には2種類の入力要素がある：ハイパーリンク及びフォームフィールド。フォームフィールドを通しての入力はまた関連のハイパーリンクの起動をトリガーできる。

【0137】

1.7.1 ハイパーリンク

ハイパーリンクはメッセージをリモートアプリケーションに送る手段で、一般的にはネットページシステム内の印刷された応答を引き出す。

【0138】

ハイパーリンク要素844はハイパーリンクの起動を取り扱うアプリケーション71と、アプリケーションへのハイパーリンクを識別するリンクID54と、ハイパーリンク起動において利用者のアプリケーションエイリアスID65を含むようにシステムに要求する“エイリアス要求”フラッグと、ハイパーリンクがお気に入りとして記録されると又は利用者の履歴に現れると使用される記述とを識別する。ハイパーリンク要素クラス線図は図29に示される。

30

【0139】

ハイパーリンクが起動すると、ページサーバはネットワーク上のどこかのアプリケーションに要求を送信する。アプリケーションはアプリケーションID64により識別され、アプリケーションIDはDNSを介して通常の方法で分解される。図30に示すように、3つのタイプのハイパーリンクがある：一般ハイパーリンク863、フォームハイパーリンク865、及び選択ハイパーリンク864。一般ハイパーリンクはリンクされたドキュメントに対する要求を実行できる、又はサーバへ選好を信号で単に送ってもよい。フォームハイパーリンクは該当のフォームをアプリケーションに配信する。選択ハイパーリンクは現状選択をアプリケーションに配信する。現状選択がテキストの単一ワードの断片を含めば、例えば、アプリケーションは単一ワードの断片が現れる文脈内のワードの意味、又は異なる言語への翻訳を与える単一ページのドキュメントを返却してもよい。各ハイパーリンクタイプは、なんの情報もアプリケーションに配信されるかにより特徴付けられる。

40

【0140】

該当のハイパーリンク事例862は取引ID55を記録し、それはハイパーリンク事例が

50

現れるページ事例に固有であり得る。取引IDはアプリケーションに対する利用者固有のデータ、例えば購入アプリケーションにより維持された保留の購入という“ショッピングカート”を利用者に代わって識別できる。

【0141】

システムは選択ハイパーリンク起動においてペンの現状選択826を含む。

たとえハイパーリンクが、最後のフォーム配信が含まれるので、入力されただけの“配信デルタ”属性セットを持っていても、システムはフォームハイパーリンク起動において関連のフォーム事例868の内容を含む。システムは全てのハイパーリンク起動において効果的な戻り路を含む。

【0142】

図31に示したように、

ハイパーリンクされたグループ866は、関連のハイパーリンクを有するグループ要素838である。グループ内で任意のフィールド要素を通して入力が生じると、そのグループに関連したハイパーリンク844は起動する。ハイパーリンクされたグループを使用してハイパーリンク行動をチェックボックス等のフィールドに関連付けることができる。ハイパーリンクされたグループはまた、フォームハイパーリンクの“配信デルタ”属性と一緒に使用され、連続した入力をアプリケーションに提供できる。それ故、ハイパーリンクされたグループを使用して“黒板”対話モデルをサポートできる、即ちそこでは入力が生じやすいや、入力が捕獲され、それ故、共有される。

【0143】

1.7.2 フォーム

フォームは、印刷されたネットページを通して関係する入力のセットを捕獲するために使用された関係する入力フィールドの収集物を定義する。フォームにより、利用者が、サーバで作動するアプリケーションソフトウェアプログラムに1つ以上のパラメータを配信することができる。

【0144】

フォーム867はドキュメント階層内のグループ要素838である。フォームは端末フィールド要素839のセットを最終的に含む。フォーム事例868はフォームの印刷事例を表現する。フォーム事例はフォームのフィールド要素845に対応するフィールド事例870のセットからなる。各フィールド事例は関連値871を有し、そのタイプは該当のフィールド要素のタイプによる。各フィールド値は特定の印刷フォーム事例を通った、即ち、1つ以上の印刷ネットページを通った入力を記録する。フォームクラス線図は図32に示される。

【0145】

各フォーム事例は、フォームが活動状態であるか、フリーズされているか、配信されているか、無効であるか、又は終了しているかを表示する状況872を有する。フォームは、最初に印刷された時は活動状態である。一旦フォームが署名されると、フォームはフリーズされる。ハイパーリンクがその“配信デルタ”属性セットを有していなければ、一旦フォームの配信ハイパーリンクの1つが起動すると、フォームが配信された状態になる。利用者がボイドフォーム、リセットフォームページコマンド、又は複製フォームページコマンドを呼び起こすと、フォームは無効になる。フォームが活動状態であった時間がフォームの規定された寿命を超えると、フォームは終了する。フォームが活動状態である間フォーム入力は許容される。活動状態ではないフォームを通しての入力は、代わりに、関連のページ事例の背景フィールド833で捕獲される。

【0146】

フォームが活動状態、又はフリーズされている状態の時、フォーム配信は許容される。フォームを配信するいかなる企ても、フォームが活動状態ではない、又はフリーズされていない時は拒絶され、代わりにフォーム状況レポートを引き出すことになる。

【0147】

各フォーム事例は、各フォーム事例から派生された任意のフォーム事例に関連し(59に

10

20

30

40

50

において)、かくしてバージョン履歴を提供する。これにより、特定の時間期間における最新のフォーム バージョンを除く全てを検索から除外することができる。

【0148】

入力全てはデジタル インクとして捕獲される。デジタル インク 873 は各々がスタイル化ストローク 875 のセットからなる時間刻印のストローク グループのセットからなる。各ストロークは、各々がまたペンオリエンテーション及び先端力を含む時間刻印されたペン位置 876 のセットからなる。デジタル インク クラス線図は図 33 に示される。

【0149】

フィールド要素 845 は、チェックボックス フィールド 877、テキスト フィールド 878、線描フィールド 879、又は署名フィールド 880 であり得る。フィールド要素 クラス線図は図 34 に示される。フィールドのゾーン 58 で捕獲された任意のデジタル インクはフィールドに割り当てられる。

【0150】

図 35 に示したように、チェックボックス フィールドは、関連のブール値 881 を有する。チェックボックス フィールドのゾーン内で捕獲された任意のマーク(合印、十字形、ストローク、横 Z 字形など)により、真の値がフィールドの値に割り当てられることになる。

【0151】

図 36 に示したように、テキスト フィールドは関連のテキスト値 882 を有する。テキスト フィールドのゾーン内で捕獲された任意のデジタル インクはオンライン肉筆認識を介してテキストに自動的に変換され、そのテキストはフィールドの値に割り当てられる。オンライン肉筆認識はよく知られている(例えば、タパート シー、シー ワイ スーエン、及びティ 若原による「オンライン肉筆認識の最新技術」、パターン解析及び機械知能に関する I E E E 報告書, V o l . 1 2、N o . 8、1990 年 8 月を参照)。

【0152】

図 37 に示したように、署名フィールドは、関連のデジタル署名値 883 を有する。署名フィールドのゾーン内で捕獲された任意のデジタル インクはペンの所有者の識別に関して自動的に検証され、フィールドがそのフォームの一部分であるフォームの内容のデジタル署名が生成され、フィールドの値に割り当てられる。デジタル署名はフォームを所有するアプリケーションに固有のペン利用者の秘密署名鍵を使用して生成される。オンライン署名検証はよく知られている(例えば、プラモンドン アール及びギー ロレットによる「自動署名検証及び筆者識別 - 最新技術」、パターン認識、V o l . 2 2、N o . 2、1989 を参照)。

【0153】

フィールド要素は、その " 隠された " 属性を設定すると、隠される。隠されたフィールド要素はページ上で入力ゾーンを有せず、入力を受け入れない。隠されたフィールド要素は、フィールドを含むフォームが配信されると、フォーム データに含まれる関連のフィールド値を有することができる。削除を示すストライクスルーなどの " 編集 " コマンドもまたフォーム フィールドで認識することができる。

【0154】

肉筆認識アルゴリズムは " オフライン " (即ち、ペン マーキングのビット マップのみにアクセス付き)よりもむしろ " オンライン " (即ち、ペン動作のダイナミクスへのアクセス付き)で作業するので、そのアルゴリズムは係属する、不連続に書かれた文字を相対的に高い正確性で、筆者依存の訓練相なしで認識できる。筆者依存の肉筆のモデルは自動的に時間の経過で生成されるが、しかしながら、必要なら、先行して生成できる。

【0155】

デジタル インクは、既に述べたように、ストロークのシーケンスからなる。特定の要素のゾーンから始まる任意のストロークは要素のデジタル インク ストリームに付加され、直ぐに解釈できる。オブジェクトのデジタル インク ストリームに付加されなかつ

10

20

30

40

50

た任意のストロークは背景フィールドのデジタル インク ストリームに付加される。

【 0 1 5 6 】

背景フィールドで捕獲されたデジタル インクは選択ジェスチャとして解釈される。1 つ以上のオブジェクトを外接することは、実際の解釈はアプリケーション固有であるけれども、外接されたオブジェクトの選択として一般的には解釈される。

表 2 はこれらの種々なペンのネットページとの対話を要約する。

表 2 ネットページとペン対話の概要

【 0 1 5 7 】

【表 2】

表 2

オブジェクト	タイプ	ペン入力	行動
ハイパーリンク	一般	クリック	行動をアプリケーションへ配信
	フォーム	クリック	フォームをアプリケーションへ配信
	選択	クリック	選択をアプリケーションへ配信
フォームフィールド	チェックボックス	任意のマーク	フィールドに真を割り当てる
	テキスト	肉筆	デジタル インクをテキストに変換する；テキストをフィールドに割り当てる
	図面	デジタル インク	デジタル インクをフィールドに割り当てる
	署名	署名	デジタル インク署名を検証する；フォームのデジタル署名を生成する；デジタル署名をフィールドに割り当てる
なし		外接	デジタル インクを現状選択に割り当てる

10

20

30

【 0 1 5 8 】

システムは各ペンに対する現行の選択を維持する。選択は背景フィールドで捕獲された最新のストロークから単になる。選択は不活動のタイムアウト後、クリアされ、予測可能な行為を保証する。

【 0 1 5 9 】

全てのフィールドで捕獲された生のデジタル インクはネットページ ページ サーバ上で保持され、フォームがアプリケーションに配信されると、フォーム データと共に任意に送信される。これにより、アプリケーションが、肉筆テキストの変換などの元の変換を疑えば、生のデジタルインクを問いただすことができる。これにより、例えば、あるアプリケーション固有の整合チェックに失敗するフォームに対してアプリケーション レベルでの人間の介入を含むことができる。これの拡張として、フォームの背景エリア全体を線描フィールドとして指名できる。その時、アプリケーションが、フォームの明示的なフィールドの外側にあるデジタルインクの存在に基づいて決定し、利用者がこれらのフィールドの外側にあるフィルイン フィールドに修正を指示したと言う仮定で、フォームを人間のオペレータに発送することができる。

40

【 0 1 6 0 】

図 3 8 はネットページに係るペン入力を扱うプロセスのフローチャートを示す。プロセスは、ペンからストロークを受信する（ 8 8 4 において ）ステップと、ストローク内のページ ID 5 0 が参照するページ事例 8 3 0 を識別する（ 8 8 5 において ）ステップと、ペー

50

ジ記述 5 を検索する (8 8 6 において) ステップと、フォーマット要素 8 3 9 を、ストロークがそのゾーンと交差するが、識別する (8 8 7 において) ステップと、フォーマット要素がフォールド要素と対応するかどうか判別し (8 8 8 において)、もしそうであれば受信したストロークをフィールド値 8 7 1 のデジタルインクに付加し (8 9 2 において)、累積されたフィールドのデジタル インクを解釈し (8 9 3 において)、フィールドがハイパーリンクされたグループ 8 6 6 の一部分であるかどうかを判別し (8 9 4 において)、もしそうであれば関連のハイパーリンクを起動させる (8 9 5 において) ステップと、代替として、フォーマットされた要素がハイパーリンク要素に対応するかどうかを判別し (8 9 9 において)、もしそうであれば該当のハイパーリンクを起動させる (8 9 5 において) ステップと、代替として、入力フィールド又はハイパーリンクがない時は受信ストロークを背景フィールド 8 3 3 のデジタル インクに付加する (8 9 0 において) ステップと、受信したストロークを、登録サーバにより維持されたように、現状ペンの現状選択 8 2 6 にコピーするステップとからなる。

10

20

30

40

50

【 0 1 6 1 】

図 3 8 a は、図 3 8 に示したプロセスにおけるステップ 8 9 3 の詳細なフローチャートを示し、そのステップでは累積されたフィールドのデジタル インクがフィールドのタイプに従って解釈される。プロセスは、フィールドがチェックボックスであるかどうかを判別し (8 9 6 において)、且つデジタル インクがチェックマークを表現しているかどうかを判別し (8 9 7 において)、もしそうであれば、真の値をフィールド値に割り当てる (8 9 8 において) サブステップと、代替として、フィールドがテキスト フィールドであるかどうかを判別し (8 9 9 において)、もしそうであれば、適切な登録サーバの助けを借りてデジタル インクをコンピュータ テキストに変換し (9 0 0 において)、且つ変換コンピュータ テキストをフィールド値に割り当てる (9 0 1 において) サブステップと、代替として、フィールドが署名フィールドであるかどうかを判別し (9 0 2 において)、もしそうであれば、適切な登録サーバの助けを借りてデジタル インクをペンの所有者の署名として検証し (9 0 3 において)、適切な登録サーバの助けをまた借り、且つ該当アプリケーションに係るペン所有者の秘密署名鍵を使用して、該当フォームの内容のデジタル署名を形成し (9 0 4 において)、デジタル署名をフィールド値に割り当てる (9 0 5 において) サブステップとからなる。

【 0 1 6 2 】

1 . 7 . 3 ページ サーバ コマンド

ページ サーバ コマンドはページ サーバによりローカルに扱われるコマンドである。ページ サーバ コマンドはフォームと、ページと、ドキュメント事例に直接作用を及ぼす。

【 0 1 6 3 】

図 3 9 に示したように、ページ サーバ コマンド 9 0 7 は、ボイド フォーム コマンド 9 0 8、複製フォーム コマンド 9 0 9、リセット フォーム コマンド 9 1 0、ゲット フォーム状況コマンド 9 1 1、複製ページ コマンド 9 1 2、リセット ページ コマンド 9 1 3、ゲット ページ状況コマンド 9 1 4、複製ドキュメント コマンド 9 1 5、リセットドキュメント コマンド 9 1 6、又はゲット ドキュメント状況コマンド 9 1 7 であり得る。

【 0 1 6 4 】

ボイド フォーム コマンドは該当のフォーム事例を無効にする。複製フォーム コマンドは該当のフォーム事例を無効にし、次いで、フィールド値を保存した状態で、現状フォーム事例のアクティブ印刷コピーを生産する。コピーはオリジナルと同じハイパーリンク取引 ID を含み、それでコピーはアプリケーションにとってオリジナルと見分けがつかない。リセット フォーム コマンドは該当のフォーム事例を無効にし、次いで、フィールド値を捨てた状態でフォーム事例のアクティブ印刷コピーを生産する。ゲット フォーム状況コマンドは、誰が発行したか、いつ印刷されたか、誰のために印刷されたか、そしてフォーム事例のフォーム状況を含む、該当のフォーム事例の状況についての印刷レポートを生産する。

【0165】

フォーム ハイパーリンク事例は取引IDを含むので、アプリケーションは新しいフォーム事例を作ることに従事しなければならない。新しいフォーム事例を要求するボタンは、それ故、ハイパーリンクとして実行されるのが一般的である。

【0166】

複製ページ コマンドは、背景フィールド値を保存した状態で該当ページ事例の印刷コピーを生産する。もしページがフォームを含む、又はページがフォームの一部であれば、その時、複製ページ コマンドは複製フォーム コマンドとして解釈される。リセット ページ コマンドは背景フィールド値を捨てた状態で該当ページ事例の印刷コピーを生産する。もしページがフォームを含む、又はページがフォームの一部であれば、その時、リセット ページ コマンドはリセット フォーム コマンドとして解釈される。ゲット ページ状況コマンドは、レポートは誰が発行したか、いつ印刷されたか、誰のために印刷されたか、を含む該当のページ事例の状況についての、及びページが含有するあるいはページがその一部である任意のフォームの状況についての印刷レポートを生産する。

10

【0167】

すべてのページに現れるネットページ ロゴは通常複製ページ要素と関連する。

【0168】

ページ事例がフィールド値を保存した状態で複製されると、フィールド値はフィールド値の本来のフォームで印刷される、即ちチェックマークは標準チェックマーク グラフィックとして現れ、テキストは植字されたテキストとして現れる。線描及び署名のみが本来のフォームで現れ、署名検証の成功を表示する標準グラフィックが署名に伴う。

20

【0169】

複製ドキュメント コマンドは、背景フィールド値を保存した状態で該当ドキュメント事例の印刷コピーを生産する。もしドキュメントが任意のフォームを含めば、その時、複製ドキュメント コマンドは、複製フォーム コマンドが行うと同じ方法でフォームを複製する。リセット ドキュメント コマンドは、背景フィールド値を捨てた状態で該当ドキュメント事例の印刷コピーを生産する。もしドキュメントが任意のフォームを含めば、その時、リセット ドキュメント コマンドは、リセット フォーム コマンドが行うと同じ方法でフォームをリセットする。ゲット ドキュメント状況コマンドは、レポートは、誰が発行したか、いつ印刷されたか、誰のために印刷されたかを含む該当のドキュメント事例の状況、及びそれが含む任意のフォームの状況、についての印刷レポートを生産する。

30

【0170】

もしページ サーバ コマンドの“オンセレクトド”属性が設定されると、その時、コマンドは、コマンドを含むページではなく、ペンの現状選択により識別されたページに対して作動する。これにより、ページ サーバ コマンドのメニューを印刷することができる。もし目標ページが指定ページ サーバ コマンドに対するページ サーバ コマンド要素を含まないと、その時、コマンドは無視される。

【0171】

アプリケーションは、関連のページ サーバ コマンド要素をハイパーリンク グループに埋め込むことによりアプリケーション固有の取扱を提供できる。ページ サーバは、ページ サーバ コマンドを実行させるのではなく、ハイパーリンクされたグループに関連したハイパーリンクを起動させる。

40

【0172】

ページ サーバ コマンド要素は、その“隠された”属性が設定されると、隠される。隠されたコマンド要素はページ上に入力ゾーンを有せず、それで、利用者により直接起動させることはできない。しかしながら、隠されたコマンド要素は、異なるページに埋め込まれたページ サーバ コマンドを介して、もしそのページ サーバ コマンドがその“オンセレクトド”属性をセットさせれば、起動させることができる。

【0173】

1.8 ネットページの標準的な特徴

50

好適なフォームにおいて、各ネットページには、それはネットページであり、それ故、対話型特徴を有していることを示すネットページ ロゴが下端に印刷される。ロゴはコピーボタンとしても活動する。多くの場合、ロゴを押すことによりページのコピーが生産される。フォームの場合、ボタンはフォーム全体のコピーを生産する。そしてチケットやクーポンなどの確かなドキュメントの場合、ボタンにより、説明的な注記又は広告ページが引き出される。

【0174】

初期設定単一ページのコピー機能は関連のネットページ ページ サーバにより直接扱われる。特殊なコピー機能はロゴ ボタンをアプリケーションにリンクすることにより扱われる。

10

【0175】

1.9 利用者ヘルプ システム

好適な実施例において、ネットページ プリンタは“ヘルプ”とラベルを貼られた単一ボタンを有する。ボタンを押すと、ボタンは単一のページの情報を引き出し、その情報は下記を含む：

- ・ プリンタ接続の状況
- ・ プリンタ消耗品の状況
- ・ 上層のヘルプ メニュー
- ・ ドキュメント機能メニュー
- ・ 上層のネットページ ネットワーク ディレクトリ

20

ヘルプ メニューはネットページ システムの使用方法についての階層的なマニュアルを提供する

ドキュメント機能メニューは下記の機能を含む：

- ・ ドキュメントのコピーを印刷する
- ・ フォームのクリーン コピーを印刷する
- ・ ドキュメントの状況を印刷する

ドキュメント機能はボタンを単に押し、次いで、ドキュメントの任意のページに触れることにより開始される。ドキュメントの状況は、ドキュメントを誰が発行したか、いつ、誰に引渡したか、誰に及びいつフォームとして以降配信されたかを表示する。

【0176】

30

ネットページ ネットワーク ディレクトリにより、利用者はネットワーク上のパブリケーション及びサービスの階層を移動できる。代替として、利用者はネットページ ネットワーク “900” ナンバー “黄ページ” を呼び、人間のオペレータと話すことができる。オペレータは所要のドキュメントを位置付けし、それを利用者プリンタに送ることができる。ドキュメント タイプによるが、出版者又は利用者は少額の“黄ページ” サービス フィーを払う。

【0177】

プリンタが印刷不可能であれば、ヘルプ ページは利用できないことは明白である。この場合 “誤り” 灯が点灯し、利用者はネットワークを超えて遠隔診断を要求できる。

2 個人専有化パブリケーション モデル

40

下記の記述において、ニュースを標準的パブリケーション例として使用してネットページ システムにおける個人専有化メカニズムを示す。ニュースは、新聞やニュース雑誌の限られた意味でしばしば使用されるが、本文脈において意図する範囲はより広い。

【0178】

ネットページ システムにおいて、ニュース パブリケーションの編集内容や広告内容は、種々のメカニズムを使用して個人専有化される。編集内容は、読者が明示的に述べ、暗黙に捕獲した興味プロフィールに従って個人専有化される。広告内容は、読者の局地性や人口統計に従って個人専有化される。

【0179】

2.1 編集の個人専有化

50

加入者は、2種類のニュースソースを利用することができる：ニュースパブリケーションを引渡すニュースソース及びニュースストリームを引渡すニュースソース。ニュースパブリケーションが出版者により集められ、編集される一方、ニュースストリームはニュース出版者又は専門化されたニュースアグリゲータのいずれかにより集められる。ニュースパブリケーションは、一般的に従来の新聞やニュース雑誌に対応し、一方ニュースストリームは数多く且つ多彩であり得る：ニュースサービスから供給された“生の”ニュース、続き漫画、フリーランサのコラム、友人のブリテンボード、又は読者自身のEメール。

【0180】

ネットページパブリケーションサーバは編集されたニュースパブリケーションの出版並びに多数のニュースストリームの集合をサポートする。読者により直接選択されたニュースストリームの集合及びそれによるフォーマットを扱うことにより、サーバは、そうでないとサーバは編集上の制御を持たないページ上に広告を配置することができる。

10

【0181】

加入者は、1つ以上の寄稿しているニュースパブリケーションを選択し、各々の個人専有化されたバージョンを作成することにより、毎日の新聞を構築する。その結果の毎日の編集が印刷され、単一の新聞と一緒に束ねられる。家庭の種々のメンバーは、異なる毎日のパブリケーションを選択し、次いで、それらをカスタマイズすることにより、メンバーの異なる興味や趣味を表現するのが一般的である。

20

【0182】

各パブリケーションに対して、読者は特定のセクションを任意に選択する。あるセクションは毎日現れるが、一方、他のセクションは毎週現れる。ニューヨークタイムズオンラインから利用できる毎日のセクションは、例えば、“ページワンプラス”、“ナショナル”、“インターナショナル”、“オピニオン”、“ビジネス”、“アートノリピング”、“テクノロジー”、“スポーツ”を含む。利用可能セクションのセットは、デフォルトサブセットと同様、パブリケーションに固有である。

【0183】

読者はカスタムセクションを作成することにより日刊の新聞を拡張でき、各々は任意の数のニュースストリームを利用する。カスタムセクションはEメール及び友達のアナウンスメント(“個人的”)のため、又は特定のトピック(“アラート”又は“クリッピング”)のためのニュース提供を監視するために作成されるかもしれない。

30

【0184】

各セクションに対して、読者はそのサイズを定性的に(例えば、短い、中間、長い)又は数的に(例えば、そのページ数の制限として)のいずれかで、及び所要の広告比率を定性的に(例えば、高い、普通、低い、無し)又は数的に(例えば、パーセンテージとして)のいずれかで任意に指定する。

【0185】

読者はまた多数のより短い記事又は少数のより長い記事に対する選好を任意に表明する。各記事を短い及び長いフォームの両方で書き(編集し)、この選好をサポートするのが理想的である。

40

【0186】

任意の記事を種々のバージョンで書き(編集し)、予想された読者の知的素養に合わせ、例えば子供や大人のバージョンを提供してもよい。適切なバージョンは読者の年齢に従って選択される。読者は、生物学的な年齢より優先される“読書年齢”を指定できる。

【0187】

各セクションを構成する記事は編集者により選択され、優先順位がきめられ、各々には利用寿命が割り当てられる。初期設定により、それらは関連の加入者の全てに優先順位により引渡され、加入者の編集におけるスペース制約の対象となる。

【0188】

50

セクションにおいて、適当であれば、読者は提携的なフィルタリングを任意に行うことができる。その時、これは十分に長い寿命を有する記事に適用される。提携的なフィルタリングの対象となる各記事には、格付けボタンが記事の終わりに印刷される。ボタンは容易な選択を（例えば、“好き”及び“好まない”）提供でき、読者があえて記事の格付けをすることを起こり易くする。

【0189】

高い優先度及び短い寿命の記事は、それ故、編集者により、重要な読み物であると効率的に見なされ、多くの関連の加入者に引渡される。

【0190】

読者は、掘り出し物係数を定性的に（例えば、私を驚かす、又は私を驚かさない）又は数的のいずれかで任意に指定する。高い掘り出し物係数は、提携的なフィルタリング中のマッチングに使用された閾値を下げる。高い係数により、該当のセクションが読者の指定した容量一杯になることが多くなる。異なる掘り出し物係数を週の別の日に指定できる。

【0191】

読者はまた、特に興味のあるトピックスをセクション内で任意に指定し、これは編集者により割り当てられた優先度を修正することになる。

【0192】

読者のインターネット接続の速度が、画像を引渡すことができる品質に影響を与える。読者はより少ない画像又はより小さい画像、又はその両方に対する選好を任意に指定する。画像の数又はサイズを減らさなければ、その時は、画像を低品質で（例えば、低解像力又は大きい圧縮で）引渡すかもしれない。

【0193】

地球規模のレベルでは、読者は量、日時、時間、お金の価値をどのようにして現地化するかを指定する。これにより、単位が英本国の標準か又はメートル法か、現地時間帯及び時間フォーマット、現地の通貨、そして現地化はその場翻訳か又は註釈からなるかを指定する。これらの選好は初期設定により読者の地域から抜粋される。

【0194】

視力が弱いことが原因の読書困難を減らすため、読者はより大きい表現に対する選好を地球規模で任意に指定する。テキスト及び画像の両方がそれに応じて拡大され、各ページ上により少ない情報が収納される。

【0195】

ニュース パブリケーションが出版される言語、及びその該当のテキスト符号化はパブリケーションの特性であり、読者により表明された選好ではない。しかしながら、ネットページ システムは種々な装いで自動翻訳サービスを提供するように構成できる。

【0196】

2.2 広告の現地化及び目的化

広告は編集文脈をうまく利用して配置されるのが一般的であるので、編集内容の個人専有化は広告内容に直接影響する。旅行広告は、例えば、他のセクションよりも旅行セクションに現れがちである。広告主にとって（それ故、出版者にとって）の編集内容の価値は、正しい人口統計で数多くの読者の引きつけるその能力にある。

【0197】

効果的な広告は局地性及び人口統計を基礎にして配置される。局地性により特別のサービス、小売業者等への接近と、地方コミュニティや環境に関連した特別な興味や関心事が判別される。人口統計は一般的な興味や先入観、並びに有望な消費パターンを判別する。

【0198】

ニュース出版者の最も利益のある商品は広告“スペース”であり、パブリケーションの地理的被覆率と、その読者層のサイズと、その読者層人口統計と、広告に使用できるページ エリアとによりにより決定された多次元実体が広告に利用できる。

【0199】

ネットページ システムでは、ネットページ パブリケーション サーバは、パブリケーシ

ヨンの地理的被覆率と、セクションの読者層と、各読者のセクション編集のサイズと、各読者の広告比率と、各読者の人口統計とを考慮しながら、セクション毎を基礎にして、近似的な多次元サイズのパブリケーションの販売可能広告スペースを算出する。

【0200】

他の媒体と比較すると、ネットページ システムにより広告スペースを非常に詳細に定義することができ、スペースのより小さい断片を個別に販売することができる。それ故、広告スペースを真の値により近い値で販売できることになる。

【0201】

例えば、同じ広告“スロット”を幾つかの広告主に変動する率で販売でき、個々の読者のページは手当たり次第にある広告主又は他の広告主の広告を受けるが、全体としては各広告主に販売されたスペースの率を保持する。 10

【0202】

ネットページ システムにより、広告が詳細な商品情報やオンライン購入と直接リンクすることができる。それ故、ネットページ システムは広告スペースの本来の価値を高めることになる。

【0203】

個人専有化や局地化はネットページ パブリケーション サーバにより自動的に取り扱われるので、広告集積者は、地理及び人口統計の両方の広範囲にわたる被覆率を任意に提供できる。以降の分解は、自動的に行われるので、効率的である。これにより、出版者が直接広告を捕獲するより広告集積者と取引することがよりコスト効率的になる。たとえ広告集積者は広告収入のある比率を取っても、出版者は集積が非常に効率的であるのでその、変更が利益中立であることに気付くであろう。広告集積者は広告主と出版者との仲介者として活動し、同じ広告を多数のパブリケーションに配置することができる。 20

【0204】

ネットページ パブリケーションでの広告配置は、パブリケーションの広告スペースがより複雑なので、パブリケーションの従来に対応物での広告配置より複雑であることは注目に値する。広告主と、広告集積者と、出版者との間の協議の十分な複雑さを無視して、ネットページ システムの好適なフォームは、これらの協議に対して、広告スペースの自動オークションに対するサポートを含む何らかの自動的なサポートを提供する。自動化は、僅かの又は高度に局地化された広告のような、少額の収入を生み出す広告の配置にとって特に望ましい。 30

【0205】

一旦配置が協議されると、集積者は広告を集め、編集し、それをネットページ広告サーバ上に記録する。それに応じて、出版者は広告配置を関連のネットページ パブリケーション サーバ上に記録する。ネットページ パブリケーション サーバが各利用者の個人専有化されたパブリケーションをレイアウトすると、ネットページ パブリケーション サーバは関連の広告をネットページ広告サーバから取り出す。

【0206】

2.3 利用者プロフィール

2.3.1 情報フィルタリング

ニュースや他のパブリケーションの個人専有化は、下記を含む利用者固有のプロファイル情報の区分けに依存する。 40

- ・ パブリケーションの特注生産
- ・ 協同的なフィルタリング ベクトル
- ・ 接触細部
- ・ 表現選好

パブリケーションの特注生産は一般的にはパブリケーション固有のものであり、それで特注生産情報は関連のネットページ パブリケーション サーバにより維持される。

【0207】

協同的なフィルタリング ベクトルは多くのニュース項目に対する利用者の格付けからな 50

る。協同的なフィルタリング ベクトルは、推薦状を作成する目的で種々の利用者の興味を関連付けるために使用される。単一の協同的なフィルタリング ベクトルを任意の特別なパブリケーションと無関係に維持する利益はあるけれども、各パブリケーションに対して別個のベクトルを維持することがより実際的である 2 つの理由がある：同じパブリケーションへの加入者のベクトル間の重複が、異なるパブリケーションへの加入者のベクトル間より発生しがちである；及びパブリケーションがその利用者の協同的なフィルタリング ベクトルを、他においては見られないそのブランドの価値の一部として示すことを欲しがちである。協同的なフィルタリング ベクトルは、それ故、関連のネットページ パブリケーション サーバによっても維持される。

【0208】

名前、街路住所、郵便番号、州、国、電話番号を含む接触細部は、本来は地球規模であり、ネットページ登録サーバによって維持される。

【0209】

量と、日付と、時間とに対する選好を含む表現選好は、同様に地球規模であり、同じやり方で維持される。

【0210】

広告の局地化は利用者の接触細部で示した局地性に依存し、一方広告の目標化は誕生日、性、結婚状況、収入、職業、教育などの個人情報、又は年齢範囲や収入範囲などの定性的な派生物に依存する。

【0211】

広告目的のため個人情報を明かすことを選ぶ利用者に対して、情報は関連のネットページ登録サーバにより維持される。かかる情報がない場合、利用者の Z I P 又は Z I P + 4 コードに関連した人口統計を基礎にして広告の目標を定めることができる。

【0212】

図 2 1、2 2、2 3 及び 2 4 に示したように、利用者、ペン、プリンタ、アプリケーション プロバイダ、及びアプリケーションの各々には、それら自体の固有の識別子が割り当てられ、ネットページ登録サーバがそれらの間の関係を維持する。登録目的のため、出版者は特殊な種類のアプリケーション プロバイダであり、パブリケーションは特殊な種類のアプリケーションである。

【0213】

各利用者 8 0 0 は任意の数のプリンタ 8 0 2 の使用を認可され、各プリンタを任意の数の利用者が使用できる。各利用者は単一の初期設定プリンタを有し (6 6 において)、そのプリンタに定期パブリケーションが初期設定により引渡される、一方オンデマンドで印刷されたページは利用者対話しているプリンタに引渡される。サーバはどの出版者に利用者が利用者の初期設定プリンタに印刷することを認可したかを絶えず注意している。出版者は任意の特別のプリンタの I D を記録せず、代わりに要求されると I D を決定する。

【0214】

利用者がパブリケーション 8 0 7 へ予約購読 8 0 8 すると、出版者 8 0 6 (即ち、アプリケーション プロバイダ 8 0 3) は特定されたプリンタ又は利用者の初期設定プリンタに印刷するように認可される。この認可は利用者によりいかなる時にも取り消すことができる。各利用者が幾つかのペン 8 0 1 を有してもよいが、ペンは単一利用者に固有である。もし利用者が認可されて特別のプリンタを使用すれば、その時、そのプリンタは利用者のペンのどれでも認識する。

【0215】

ペン I D を使用して、通常な方法で D N S を介して特別のネットページ登録サーバにより維持された該当の利用者プロフィールを位置付けする。

【0216】

ウェブ端末 8 0 9 が特別のネットページ プリンタ上に印刷されることが認可され、ウェブ ブラウジング中に会ったウェブページ及びネットページドキュメントを最も近いネットページ プリンタ上で便宜よく印刷することが可能となる。

10

20

30

40

50

【0217】

ネットページ システムは、プリンタ プロバイダに代わって、プロバイダのプリンタ上に印刷されたパブリケーションを通して得られた収入についてのフィー及び手数料を集金することができる。かかる収入は広告フィーと、クリックスルー フィーと、Eコマース手数料と、取引フィーとを含む。プリンタが利用者により保有されていれば、その時、利用者はプリンタ プロバイダである。

【0218】

各利用者はまたネットページ口座 820 を有し、その口座は、少額借方及び貸方と（前の章で説明したものなど）、名前、住所、電話番号を含む接触細部 815 と、プライバシー、配信、局地化設定を含む地球規模選好 816 と、利用者の符号化署名 818、指紋 819 等を含む任意の数の生物形質記録 817 と、システムにより自動的に維持された肉筆モデル 819 と、Eコマースの決済ができる SET 決済カード口座 821 とを累積するのに使用される。

10

【0219】

2.3.2 お気に入りリスト

ネットページ利用者は、ネットページネットワーク上に“お気に入り” - 役に立つドキュメント等へのリンクのリスト 922 を維持できる。リストはシステムにより利用者に代わって維持される。リストはフォルダ 924 の階層として組織化され、その好適な実施形態が図 41 のクラス線図に示される。

【0220】

2.3.3 履歴リスト

システムは各利用者に代わって、利用者がネットページ システムを通してアクセスしたドキュメント等へのリンクを含む履歴リスト 929 を維持する。履歴リストは日時順のリストとして組織化され、その好適な実施形態が図 42 のクラス線図に示される。

20

【0221】

2.4 知的なページ レイアウト

ネットページ パブリケーション サーバは、セクション毎のベースで各利用者の個人専有化されたパブリケーションのページを自動的にレイアウトする。大部分の広告が事前にフォーマットされた長方形のフォームであるので、それらをページ上で編集内容の前に配置する。

30

セクションに対する広告率は、セクション内の個々のページ上で広告率を広く変えて達成でき、広告レイアウト アルゴリズムがこれを利用する。アルゴリズムは、ドイト (do-it-yourself) の屋根ふき修理についての特集記事のために、屋根ふき材料の広告をパブリケーション内に特に配置するなど、密接に結びついた編集及び広告内容を特に一緒に位置付けすることを企てるように構成される。

【0222】

テキストと、関連の画像及びグラフィックスとを含む利用者のために選択した編集内容は、次いで、種々の審美的な法則に従ってレイアウトされる。

【0223】

広告の選択や編集内容の選択を含むプロセス全体は、利用者が述べたセクション サイズ選好がより密接に達成されることを意図するために、一旦レイアウトが纏まって、繰り返さなければならぬ。しかしながら、セクション サイズ選好は時間平均で一致させることができ、日々のかんりの変化が可能となる。

40

【0224】

2.5 ドキュメント フォーマット

一旦ドキュメントがレイアウトされると、それはネットページ ネットワーク上での効率的な配布及び永続的な記憶のため符号化される。

【0225】

主たる効率メカニズムは、単一利用者の編集に固有の情報と多数の利用者の編集間での共有情報との分離である。固有の情報はページレイアウトからなる。共有情報はページレイ

50

アウトが参照するオブジェクトからなり、それは画像、グラフィックス、テキストの断片を含む。

【0226】

テキスト オブジェクトは、拡張可能スタイルシート言語 (XSL) を使用する、拡張可能マークアップ言語 (XML) で表現された、完全にフォーマットされたテキストを含む。XSLは、テキストが設定されつつある領域、この場合にはレイアウトにより提供されている、とは無関係にテキスト フォーマットを正確に制御する。テキスト オブジェクトは自動翻訳を可能にする埋め込み言語コードと、パラグラフ フォーマットを助ける埋め込みハイフンネーション ヒントとを含む。

【0227】

画像オブジェクトは、JPEG 2000 ウェブレット ベースの圧縮画像フォーマットで画像を符号化する。グラフィック オブジェクトはスケーラブル ベクター グラフィックス (SVG) フォーマットで二次元のグラフィックを符号化する。

【0228】

レイアウト自体は、一連の配置された画像及びグラフィック オブジェクト、テキスト オブジェクトが流れるリンクされたテキスト フロー オブジェクト、上記したハイパーリンク及び入力フィールド、及びすかし領域からなる。これらのレイアウト オブジェクトを表3に要約する。レイアウトは効率的な配布及び記憶に適したコンパクトなフォーマットを使用する。

表3 ネットページ レイアウト オブジェクト

【0229】

【表3】

表3

レイアウト オブジェクト	属性	リンクされたオブジェクトのフォーマット
画像	位置	—
	画像オブジェクトID	JPEG2000
グラフィック	位置	—
	画像オブジェクトID	SVG
テキストフロー	テキストフローID	—
	ゾーン	—
	任意のテキスト オブジェクトID	XML/XSL
ハイパーリンク	タイプ	—
	ゾーン	—
	アプリケーションID等	—
フィールド	タイプ	—
	意味	—
	ゾーン	—
すかし	ゾーン	—

【0230】

2.6 ドキュメント配布

上記したように、ネットページ ネットワーク上での効率的な配布及び永続的に保存する目的のため、利用者固有のページ レイアウトは、それが参照する共有オブジェクトから分離される。

【0231】

10

20

30

40

50

予約購読のパブリケーションの配布準備ができると、ネットページパブリケーションサーバは、ネットページIDサーバ12の助けを借りて、ページ、ページ事例、ドキュメント、及びドキュメント事例に各々固有のIDを割り当てる。

【0232】

サーバは、共有内容が最適化されたサブセットのセットを算出し、各サブセットに対してマルチキャストチャンネルを作成し、次いで、各利用者固有のレイアウトを、そのレイアウトにより使用された共有内容を運ぶであろうマルチキャストチャンネルの名称でタグ付けする。サーバは、次いで、各利用者のレイアウトを、適切なページサーバを介してその利用者のプリンタにポイントキャストし、ポイントキャストが完了すると、共有内容を指定チャンネル上でマルチキャストする。そのポイントキャストを受信後、ページサーバ及びプリンタの各々は、ページレイアウトで指定されたマルチキャストチャンネルに予約購読する。マルチキャスト中、ページサーバ及びプリンタの各々は、マルチキャストストリームからそのページレイアウトにより参照されたオブジェクトを抜粋する。ページサーバは受信ページレイアウト及び共有内容を永続的に記録保管する。

10

【0233】

一旦プリンタがそのページレイアウトが参照する全てのオブジェクトを受信すると、プリンタは完全に配置されたレイアウトを再作成し、次いで、それをラスタ化し、印刷する。

【0234】

通常的环境下では、プリンタはページを配信することができるよりも早く印刷する。各ページの四分の一が画像で占められると仮定すると、平均ページは400KB未満のサイズを有する。プリンタは、それ故、その内部64MBメモリ内にかかるページを100枚を越えて保持でき、一時的なバッファ等を可能にする。プリンタは毎秒1ページの割合で印刷する。これは毎秒400KB又は約3Mビットのページデータに相当し、それはブロードバンドネットワーク上で予想される最高のページデータ配信率と同様である。

20

【0235】

プリンタが紙切れを生じる時などの、たとえ異常な環境においても、プリンタの100ページ相当の内部記憶容量が使い尽くされる前に、利用者は紙を補給することができると思われる。

【0236】

しかしながら、プリンタの内部容量が実際にいっぱいになってしまうと、それが最初に生じると、その時、プリンタはマルチキャストを使用できない。それ故、ネットページパブリケーションサーバは、プリンタが再マルチキャストに対する要求を配信することができるようにする。臨界の数の要求が受信されると、又はタイムアウトが発生すると、サーバは該当の共有オブジェクトを再マルチキャストする。

30

【0237】

一旦ドキュメントが印刷されると、プリンタは、そのページレイアウト及び内容を関連のページサーバから検索することにより、いかなる時でも正確な複製を生産することができる。

【0238】

2.7 オンデマンドドキュメント

ネットページドキュメントがオンデマンドで要求されると、ネットページドキュメントは、定期刊行物と非常に類似したやり方で個人専有化され、引渡されることができる。しかしながら、共有内容がないので、マルチキャストを使用しないで、要求しているプリンタに直接引渡される。

40

【0239】

非ネットページドキュメントがオンデマンドで要求されると、それは個人専有化されず、それをネットページドキュメントとして再フォーマットする指定ネットページフォーマットサーバを介して引渡される。ネットページフォーマットサーバはネットページパブリケーションサーバの特殊な事例である。ネットページフォーマッ

50

ティング サーバは、アドビ社のポータブル ドキュメント フォーマット (P D F)、及びハイパーテキスト マーク付け言語 (H T M L) を含む、種々のインターネット ドキュメント フォーマットの知識を有する。 H T M L の場合、それは印刷ページのより高い解像力を利用して、内容の表付きで、ウェブ ページをマルチコラム フォーマットで表示できる。 H T M L は、要求されたページに直接リンクされた全てのウェブ ページを自動的に含むことができる。利用者は、選好を介してこの挙動を調整することができる。

【 0 2 4 0 】

ネットページ フォーマット サーバにより、対話性及び永続性を含む標準のネットページ挙動が、その起源やフォーマットが何であれ、任意のインターネット ドキュメント上で利用できるようになる。ネットページ フォーマット サーバは、ネットページ プリンタ及びネットページ サーバの両方からの種々のドキュメント フォーマットの知識を隠し、ウェブ サーバからのネットページ システムの知識を隠す。

10

【 0 2 4 1 】

3 セキュアリティ

3 . 1 暗号法

暗号法を使用して、機密扱いの情報を記憶及びトランシットの両方において保護し、取引に係るパーティを認証する。広く使用される 2 つのクラスの暗号法がある：秘密キー暗号法及び公開キー暗号法。ネットページ ネットワークは両方のクラスの暗号法を使用する。

【 0 2 4 2 】

秘密キー暗号法は、対称暗号法とも称されるが、同じキーを使用しメッセージを暗号化し解読する。メッセージを交換することを希望する 2 つのパーティは秘密キーを安全に交換するため最初に手筈をつけなければならない。

20

【 0 2 4 3 】

公開キー暗号法は、非対称暗号法とも称されるが、2 つの暗号化キーを使用する。2 つのキーは、一方のキーを使用して暗号化された任意のメッセージは、他のキーを使用してのみ解読できる様に、数学的に関係している。その時、これらのうち 1 つのキーは公開され、一方他のキーは秘密のままである。公開キーを使用して、秘密キーの所有者用に意図した任意のメッセージを暗号化する。一旦公開キーを使用して暗号化されると、メッセージは秘密キーを使用してのみ解読できる。かくして、最初に秘密キーを交換する必要がなく、2 つのパーティはメッセージを安全に交換することができる。秘密キーが確実であることを保障するため、秘密キーの所有者はキー対を生成するのが普通である。

30

【 0 2 4 4 】

公開キー暗号法を使用して、デジタル署名を作成することができる。公開キーの所有者は既知のメッセージのハッシュを作成し、次いで、秘密キーを使用してそのハッシュを暗号化できる。次いで、公開キーを使用して暗号化されたハッシュを解読し、且つそのメッセージに対してハッシュを検証することにより、誰でも暗号化されたハッシュが、その特別のメッセージに関して秘密キーの所有者の“署名”を構成していることを検証できる。もし署名がメッセージに付加されれば、その時、メッセージの受信者はメッセージが本物であり、且つメッセージがトランシットにおいて変更されていないことの両方を検証できる。

40

【 0 2 4 5 】

公開キー暗号法が機能するすためには、偽造を防ぐ公開キーを配布する方法が必要である。これは通常の認証及び認証局を用いて行われる。認証局は、公開キーといずれかの人の識別との間の接続を認証する委託された第三者機関である。認証局は、識別ドキュメントを調べることにより人の識別を検証し、そして人の識別の詳細及び公開キーを含むデジタル証書を作り、署名する。認証局を信頼する誰でも証書内の公開キーをこれは本物であるという高い確実性で使用できる。彼らは証書が実際に、公開キーが周知の認証局により、署名されていることを実際に検証しなければならない。

【 0 2 4 6 】

50

大部分の取引環境において、公開キー暗号法はデジタル署名を作成し、安全に秘密セッションキーを交換するためにのみ使用される。秘密キー暗号法は他の全ての目的のために使用される。

【0247】

下記の議論において、ネットページプリンタとサーバとの間の情報の確実な送信への参照がなされると、実際に発生することは、プリンタはサーバの証書を得、認証局に参照することによりそれを認証し、証書内の公開キー交換キーを使用して秘密セッションキーをサーバと交換し、次いで秘密セッションキーを使用してメッセージデータを暗号化する。セッションキーは、定義上、任意の短い寿命であり得る。

【0248】

10

3.2 ネットページプリンタセキュリティ

各ネットページプリンタには、プリンタ内の読み出し専用メモリ及びネットページ登録サーバデータベースに記憶される、一对の固有の識別子が製造時割り当てられる。第1のID62は公開で、ネットページネットワーク上でプリンタを固有に識別する。第2のIDは秘密であり、プリンタが最初にネットワーク上で登録される時に使用される。

【0249】

プリンタは、設置後、最初にネットページネットワークに接続すると、一对の署名公開/秘密キーを作成する。プリンタは秘密ID及び公開キーを安全にネットページ登録サーバに送信する。サーバは、秘密IDをそのデータベースに記録されているプリンタの秘密IDに対して比較し、それらのIDが一致すれば、登録を受け入れる。次いで、サーバはプリンタの公開ID及び公開署名キーを含む証書を作成し、それに署名し、証書を登録データベース内に記憶する。

20

【0250】

ネットページ登録サーバは、秘密情報へのアクセスを有し、プリンタの識別を検証することができるので、ネットページプリンタに対する認証局として活動する。

【0251】

利用者がパブリケーションを予約購読すると、出版者に利用者の初期設定プリンタ又は指定プリンタにパブリケーションを印刷することを認可する、ネットページ登録サーバデータベース内に、記録が作成される。ページサーバを介してプリンタに送られた各ドキュメントは、特定の利用者にアドレス指定され、出版者の秘密署名キーを使用して出版者により署名される。ページサーバは、登録データベースを介して、出版者がパブリケーションを指定利用者に引渡すことが認可されていることを検証する。ページサーバは、登録データベースに記憶されている出版者の証書から得られる出版者の公開キーを使用して、署名を検証する。

30

【0252】

ネットページ登録サーバは、印刷認可をデータベースに加える要求を、それらの要求がプリンタに登録されたペンを介して始動されている限り受け入れる。

【0253】

3.3 ネットページペンセキュリティ

各ネットページペンには、ペン内の読み出し専用メモリ及びネットページ登録サーバデータベースに記憶される、固有の識別子が製造時に割り当てられる。ペンID61はネットページネットワーク上でペンを固有に識別する。

40

【0254】

ネットページペンは多くのネットページプリンタを“知る”ことができ、プリンタは多くのペンを“知る”ことができる。ペンは、ペンがプリンタの範囲内にあればいつでも、無線周波信号を介してプリンタと通信する。一旦、ペン及びプリンタが登録されると、ペン及びプリンタは定期的にセッションキーを交換する。ペンがデジタルインクをプリンタに送信する毎に、デジタルインクは適切なセッションキーを使用して常に暗号化される。デジタルインクは平文では決して送信されない。

【0255】

50

ペンは、ペンが知っている各プリンタに対する、プリンタIDによる索引付きセッションキーを記憶し、プリンタは、プリンタが知っている各ペンに対する、ペンIDによる索引付きセッションキーを記憶する。両方ともセッションキー用の大きい、有限の記憶容量を有し、必要なら、最低使用頻度ベースでセッションキーを忘れるであろう。

【0256】

ペンがプリンタの範囲内に入ると、ペン及びプリンタは、それらが互いに知っているかどうかを発見する。それらが互いに知っていなければ、次いで、プリンタはペンを知っているかどうかを判別する。これは、例えば、ペンが、プリンタを使用するように登録されている利用者に属しているかもしれないためである。もしプリンタがペンを知らなければならぬが、知っていなければ、次いで、プリンタは自動ペン登録手順を始動する。もしプリンタがペンを知る必要がなければ、次いで、プリンタは、ペンが充填カップに配置されるまで、その時プリンタは登録手順を始めるが、ペンを無視することについてペンと同意する。

10

【0257】

ペンの公開IDの他に、ペンは秘密のキー交換キーを含む。キー交換キーはまた製造時にネットワーク登録サーバデータベースに記録される。登録中に、ペンはそのペンIDをプリンタに送信し、プリンタはペンIDをネットワーク登録サーバに送信する。サーバはプリンタ及びペンが使用するセッションキーを生成し、安全にセッションキーをプリンタに送信する。サーバはまたペンのキー交換キーで暗号化されたセッションキーのコピーを送信する。プリンタはペンIDによる索引付きセッションキーを内部記憶し、暗号化されたセッションキーをペンに送信する。ペンはプリンタIDによる索引付きセッションキーを内部記憶する。

20

【0258】

にせペンがペン登録プロトコールでペンを偽装できても、実のペンのみがプリンタにより送信されたセッションキーを解読できる。

【0259】

事前に登録されなかったペンが最初に登録されると、そのペンが利用者にリンクされるまで、使用が制限される。登録されたが“所有されていない”ペンは、ネットワーク利用者及びペン登録フォームを要求し、それに所要の書き入れをし、新しいペンが自動的にリンクされる新しい利用者を登録し、又は新しいペンを既存の利用者に加えてのみ使用するの

30

【0260】

ペンは、ハードウェア性能がペンにおいて束縛されるので、公開キー暗号法よりもむしろ秘密キー暗号法を使用する。

【0261】

3.4 セキュア ドキュメント

ネットワークシステムは、チケットやクーポン等のセキュア ドキュメントの引渡しをサポートする。ネットワーク プリンタはすかしを印刷する設備を含むが、適切に認可される出版者からの要求でのみそれを行う。出版者はその当局に証書内に、プリンタが認証できるすかしを、印刷するように指示する

40

“すかし”印刷プロセスは、ページの指定“すかし”領域において代替のディザマトリックスを使用する。背面併せのページは、印刷された時に一致するミラー画像すかし領域を含む。奇数及び偶数ページのすかし領域で使用されたディザマトリックスは、領域と一緒に、印刷シートを通して見ることにより得られる干渉効果を生産するように設計される。

【0262】

効果は、すかしがページの片面でのみで見ると見えず、且つ通常的手段でページをコピーすると失われる点においてすかしに類似している。

【0263】

セキュア ドキュメントのページは、上記のセクション1.9で説明した組み込み式ネッ

50

トページ コピー メカニズムを使用してはコピーできない。これは、ネットページ認識の
写真複写機上でのネットページの印刷に拡張することになる。

【0264】

セキュア ドキュメントは一般的にはEコマース取引の一部として生成される。セキュ
ア ドキュメントは、それ故、セクション2で説明したように、利用者が生物形質情報を
ネットページ登録サーバに登録した時に捕獲した利用者の写真を含む。

【0265】

セキュア ネットページ ドキュメントを提示されると、受信者は通常の方法でその状況を
要求することによりその真正を検証できる。セキュア ドキュメントの固有IDはドキュ
メントの寿命の間にのみ有効であり、セキュア ドキュメントIDは、便宜主義的な偽造
者によりそのIDが予測されないように、不連続に配分される。セキュア ドキュメント
検証ペンは、検証失敗についての組込みフィードバック方式で開発でき、容易な提示時点
ドキュメント検証をサポートする。

10

【0266】

明らかに、すかしも利用者の写真もいずれも暗号的な意味において安全ではない。それら
は思いつきの偽造に対する大きな障害を単に提供するだけである。特に検証ペンを使用す
る、オンライン ドキュメント検証は、必要とされる場合、追加的なレベルのセキュリテ
ィをもたらすが、なお完全に偽造からまぬがれるものではない。

【0267】

3.5 非拒絶

20

ネットページ システムにおいて、利用者が配信したフォームは確実にフォーム取扱者に
引渡され、ネットページ ページ サーバに永続的に記録保存される。それ故、受領者は配
信を拒絶することは不可能である。

【0268】

システムを通して行われたEコマース決済はまた、セクション4で説明したように、受取
人が拒絶することは不可能である。

【0269】

4 電子コマース モデル

4.1 セキュア電子取引 (SET)

ネットページ システムは、セキュア電子取引 (SET) をその決済システムの1つとし
て使用する。マスターカード及びヴィザにより開発されたSETは、決済カードを中心
にして組織され、このことは用語に反映される。しかしながら、システムの大部分は使用さ
れる口座の種類には無関係である。

30

【0270】

SETにおいて、カード所有者及び商人は認証局に登録し、彼等の公開署名キーを含む証
書が発行される。認証局は、カード所有者のカード発行人への登録の詳細を適切かどうか
検証し、商人の取得者への登録の詳細を適切かどうか検証する。カード所有者及び商人は
、それぞれの秘密署名キーを彼等のコンピュータに安全に記憶する。決済プロセス中、こ
れらの証書は、商人及びカード所有者を相互に認証するために、また彼等の両方が決済ゲ
ートウェイに入ることを認証するために使用される。

40

【0271】

一部には、キー及び証書をカード所有者が維持することがわずらわしいと見なされている
ので、SETはまだ広く採用されていない。カード所有者のキー及び証書をサーバ上に維
持し、カード所有者にパスワードを介してアクセスを与える暫定的な解決により、いくら
かの成功が得られている。

【0272】

4.2 SET決済

ネットページ システムにおいて、ネットページ登録サーバは、SET決済取引において
、ネットページ利用者(即ち、カード所有者)に対する代理人として行動する。

【0273】

50

ネットページ システムは生物測定法を使用して利用者を認証し、S E T 決済を認可する。システムがペンをベースのものであるので、使用される生物形質は利用者のオンライン署名であり、それは時間変動するペンの位置及び圧力からなる。ペンに指紋センサを設計することにより、高コストであるが、指紋生物形質も使用することができる。使用された生物形質のタイプは生物形質の捕獲にのみ影響し、システムの認可態様には影響しない。

【 0 2 7 4 】

S E T 決済を可能にする第 1 のステップは、利用者の生物形質をネットページ登録サーバに登録することである。これは制御された環境、例えば、銀行でなされ、そこでは利用者の識別を検証すると同時に生物形質を捕獲することができる。生物形質は捕獲され、登録データベースに記憶され、それは利用者の記録にリンクされる。利用者の写真もまた任意に捕獲され、記録にリンクされる。S E T カード保有者登録プロセスが完了し、その結果の秘密署名キー及び証書がデータベースに記憶される。利用者の決済カード情報もまた記憶され、任意の S E T 決済取引において利用者の代理人として行動するためにネットページ登録サーバに十分な情報を与える。

10

【 0 2 7 5 】

利用者が最終的に生物形質を提供し、例えば、ネットページ注文フォームに署名することにより、決済を完了すると、プリンタが、注文情報、ペン I D、及び生物形質データをネットページ登録サーバに安全に送信する。サーバは、ペン I D により識別された利用者に関して生物形質を検証し、それから以降 S E T 決済取引を完了する際に利用者の代理人として行動する。

20

【 0 2 7 6 】

4 . 3 少額決済

ネットページ システムは、少額決済に対するメカニズムを含み、低価格のドキュメントをオンデマンドで印刷し、及び著作権のあるドキュメントをコピーすることに対して利用者に便宜よく請求することが可能であり、恐らく、広告材料を印刷する時に発生した支出に対して利用者が返済することもできる。後者は利用者に既に提供した補助金のレベルによる。

【 0 2 7 7 】

利用者が E コマースに対して登録すると、少額決済を集めるネットワーク口座が開設される。利用者は定期的にステートメントを受け取り、標準の決済メカニズムを使用して、任意の未払いの借方残高を清算することができる。

30

【 0 2 7 8 】

ネットワーク口座を拡大して定期刊行物の予約購読フィーを集めることができ、これらも、さもなければ、個々のステートメントのフォームで利用者に提示される筈である。

【 0 2 7 9 】

4 . 4 取引

利用者が特殊なアプリケーション文脈においてネットページを要求すると、アプリケーションは、利用者固有の取引 I D 5 5 をページに埋め込むことができる。それ以降のこのページを通しての入力は取引 I D でタグ付けされ、アプリケーションはそれにより利用者の入力に対する適切な文脈を設定できる。

40

【 0 2 8 0 】

しかしながら、入力が利用者固有ではないページを通して発生すると、アプリケーションは利用者に固有の識別を使用して文脈を設定しなければならない。一般的な例では事前印刷のカタログページからの品目を利用者の仮想の“ショッピングカート”に加えることに伴って生じる。しかしながら、利用者のプライバシーを保護するため、ネットページ システムに知られた固有の利用者 I D はアプリケーションには漏らされない。これは、異なるアプリケーション プロバイダが、別個に蓄積された行動データを容易に関連付けるのを防ぐためである。

【 0 2 8 1 】

50

図 2 4 に示したように、ネットページ登録サーバは、この代わりに、ユニーク エイリアス ID 6 5 を介して利用者とアプリケーションとの匿名の関係を維持する。利用者が“登録済み”の属性でタグ付けしたハイパーリンクを起動させるといつでも、ネットページサーバは、ペン ID 6 1 と共に関連のアプリケーション ID 6 4 を、エイリアス ID 6 5 に翻訳するようにネットページ登録サーバに要請する。それにより、エイリアス ID 6 5 はハイパーリンクのアプリケーションに配信される。

【 0 2 8 2 】

アプリケーションはエイリアス ID による索引付き状態情報を維持し、利用者の地球規模の識別を知らないで利用者固有の状態情報を検索できる。

【 0 2 8 3 】

システムはまた、利用者のアプリケーションの各々に対する独立した証書及び秘密署名キーを維持し、システムが、アプリケーション固有の情報のみを使用して、利用者に代わってアプリケーション取引に署名することが可能になる。

【 0 2 8 4 】

システムが商品バーコード (UPC) “ハイパーリンク” 起動を発送するのを助けるため、システムは任意の数の商品タイプに対して、お気に入りアプリケーションを利用者の代わりに記録する。

【 0 2 8 5 】

各アプリケーションはアプリケーション プロバイダに関連し、システムは各アプリケーション プロバイダの代わりに口座を維持し、アプリケーションがクリックスルー フィー

10

20

30

40

50

等をプロバイダの貸方及び借方に記入することが可能になる。

【 0 2 8 6 】

アプリケーション プロバイダは定期的な予約購読された内容の出版者であり得る。システムは、予約購読されたパブリケーションを受領する利用者の意欲、並びに予想パブリケーション頻度を記録する。

【 0 2 8 7 】

4 . 5 リソース定義及び著作権

リソース記述クラス線図の好適な実施例は図 4 0 に示される。

【 0 2 8 8 】

各ドキュメント及び内容オブジェクトを 1 つ以上のリソース記述 8 4 2 により記述してもよい。リソース記述は、電子リソースの発見を容易にするように設計されるダブリン コア メタデータ要素セットを使用する。ダブリン コア メタデータはワールド ワイド ウェブ コンソーシアム (W3C) リソース定義フレームワーク (RDF) に適合する。

【 0 2 8 9 】

リソース記述は権利保持者 9 2 0 を識別できる。ネットページ システムは、利用者が著作権内容を印刷すると、著作権フィーを利用者から権利保持者に自動的に発送する。

【 0 2 9 0 】

5 交信プロトコール

交信プロトコールは実体間の順序付きメッセージの交換を定義する。ネットページ システムにおいて、ペン、プリンタ、及びサーバなどの実体は被定義のプロトコールのセットを利用して、利用者とネットページ システムとのやりとりを協調的に扱う。

【 0 2 9 1 】

各プロトコールはシーケンス線図により示され、水平次元を使用してメッセージ フローを表現し、垂直次元を使用して時間を表現する。各実体は実体の名称を含む四辺形と、実体の生命線を表示する垂直コラムとにより表現される。実体が存在する時間中、生命線は破線で示される。実体が活動状態である時間中、生命線は 2 重線で示される。ここで考慮したプロトコールは実体を作らない、又は破壊しないので、生命線は、実体がプロトコールに参加するのを停止すると、直ちに切りつめられるのが一般的である。

【 0 2 9 2 】

5 . 1 定期購読配信プロトコール

定期購読配信プロトコールの好適な実施例は図 4 3 に示される。

【 0 2 9 3 】

多数の利用者が定期パブリケーションに予約購読できる。各利用者の編集は異なってレイアウトされるかもしれないが、多くの利用者の編集はテキストオブジェクト及び画像オブジェクトなどの共通内容を共有する。それ故、定期購読配信プロトコールは、ドキュメント構造を、ポイントキャストを介して個人のプリンタに引渡すが、共有内容オブジェクトはマルチキャストを介してを引渡す。

【 0 2 9 4 】

アプリケーション（即ち、出版者）は各ドキュメントに対するドキュメント ID 5 1 を ID サーバ 1 2 から最初に得る。次いで、アプリケーションは、そのドキュメント ID 及びページ記述を含む各ドキュメント構造を、ドキュメントの新たに割り振られた ID を配分することに責任を持つページサーバ 1 0 に送信する。アプリケーションはそれ自体のアプリケーション ID 6 4、加入者のエイリアス ID 6 5、及び関連のマルチキャストチャンネル名のセットを含む。アプリケーションは、その秘密署名キーを使用してメッセージに署名する。

10

【 0 2 9 5 】

ページサーバはアプリケーション ID 及びエイリアス ID を使用して、登録サーバから該当の利用者 ID、利用者の選択プリンタ ID 6 2（それはアプリケーションに対して明示的に選択されるかもしれない、又は利用者の初期設定プリンタであるかもしれない）、及びアプリケーションの証書を得る。

20

【 0 2 9 6 】

アプリケーションの証書により、ページサーバはメッセージ署名を検証することができる。登録サーバへのページサーバの要求は、アプリケーション ID 及びエイリアス ID が定期購読 8 0 8 を一緒に識別しなければ、失敗する。

【 0 2 9 7 】

次いで、ページサーバはドキュメント ID 及びページ事例 ID を配分し、ページ ID 5 0 を含むページ記述をプリンタへ発送する。ページサーバは、プリンタが聞くべき関連のマルチキャストチャンネル名のセットを含む。

【 0 2 9 8 】

次いで、ページサーバは新たに配分されたページ ID を将来の参照のためアプリケーションに返却する。

30

【 0 2 9 9 】

一旦アプリケーションが、関連のページサーバを介して、ドキュメント構造の全てを加入者の選択プリンタに配布し終わると、アプリケーションは共有オブジェクトの種々のサブセットを事前選択のマルチキャストチャンネル上でマルチキャストする。ページサーバ及びプリンタの両方とも適切なマルチキャストチャンネルを監視し、それらの要求された内容オブジェクトを受信する。次いで、ページサーバ及びプリンタは、事前にポイントキャストされたドキュメント構造を配置できる。これにより、ページサーバは完全なドキュメントを彼らのデータベースに加えることができ、それによりプリンタはドキュメントを印刷できることになる。

40

【 0 3 0 0 】

5 . 2 ハイパーリンク起動プロトコール

ハイパーリンク起動プロトコールの好適な実施例を図 4 5 に示す。

【 0 3 0 1 】

利用者がネットページ上でネットページ ペンを使いクリックすると、ペンはそのクリックを最も近いネットページ プリンタ 6 0 1 に通信する。クリックはページ及びページ上の場所を識別する。プリンタはペンの ID 6 1 をペン接続プロトコールから既に知っている。

【 0 3 0 2 】

プリンタは、DNS を介して、特別のページ ID 5 0 を扱うページサーバ 1 0 a のネッ

50

トワーク アドレスを判別する。利用者が同じページと最近対話済みであれば、アドレスはそのキャッシュに既にあるかもしれない。次いで、プリンタはペンID、それ自体のプリンタID 6 2、ページID、及びクリック場所をページサーバに発送する。

【0303】

ページサーバは、ページIDにより識別されたページ記述5を取り込み、そのクリックが、もしあれば、どの入力要素のゾーン58に在るかを判別する。関連の入力要素がハイパーリンク要素844であると仮定すると、その時、ページサーバは関連のアプリケーションID 6 4及びリンクID 5 4を取得し、DNSを介して、アプリケーション71をホストするアプリケーションサーバのネットワークアドレスを判別する。

10

【0304】

ページサーバは、ペンID 6 1を使用して該当の利用者ID 6 0を登録サーバ11から取得し、次いで、地球規模で固有のハイパーリンク要求ID 5 2を配分し、ハイパーリンク要求934を構築する。ハイパーリンク要求クラス線図は図44に示される。ハイパーリンク要求は、要求している利用者及びプリンタのIDを記録し、クリックされたハイパーリンク事例862を識別する。次いで、ページサーバはそれ自体のサーバID 5 3、ハイパーリンク要求ID、及びリンクIDをアプリケーションに送信する。

【0305】

アプリケーションは、アプリケーション固有のロジックに従って応答ドキュメントを生産し、ドキュメントID 5 1をIDサーバ12から取得する。次いで、アプリケーションはドキュメントを、要求しているページサーバのID及びハイパーリンク要求IDと共に、ドキュメントの新たに配分されたIDに責任を負うページサーバ10bに送る。

20

【0306】

第2のページサーバは、ハイパーリンク要求ID及びアプリケーションIDを第1のページサーバに送り、該当の利用者ID及びプリンタID 6 2を取得する。第1のページサーバは、ハイパーリンク要求が既に終了していれば又は異なるアプリケーション向けであれば、リクエストを拒絶する。

【0307】

第2のページサーバは、ドキュメント事例及びページID 5 0を配分し、新たに配分されたページIDをアプリケーションに返却し、完全なドキュメントをそれ自体のデータベースに加え、そして最後にページ記述を、要求しているプリンタに送信する。

30

【0308】

ハイパーリンク事例は、意味のある取引ID 5 5を含むことができ、その場合第1のページサーバはアプリケーションに送信したメッセージ内に取引IDを含む。これにより、アプリケーションはハイパーリンク起動に対する取引固有の文脈を設定することができる。

【0309】

ハイパーリンクが利用者エイリアスを要求していれば、即ち“エイリアス要求”属性が設定されれば、その時、第1のページサーバはペンID 6 1及びハイパーリンクのアプリケーションID 6 4の両方を登録サーバ11に送り、ペンIDに対応する利用者IDだけでなく、アプリケーションID及び利用者IDに対応するエイリアスID 6 5もまた取得する。ページサーバは、アプリケーションに送信したメッセージ内にエイリアスIDを含み、アプリケーションがハイパーリンク起動に対する利用者固有の文脈を設定することが可能になる。

40

【0310】

5.3 肉筆認識プロトコール

利用者がネットページ上でネットページペンを使いストロークを描くと、ペンはそのストロークを最も近いネットページプリンタに通信する。ストロークはページ及びページ上のパスを識別する。

【0311】

50

プリンタはペン I D 6 1、それ自体のプリンタ I D 6 2、ページ I D 5 0、及びストロークパスを通常の方法でページサーバ 1 0 に発送する。

【 0 3 1 2 】

ページサーバは、ページ I D により識別されたページ記述 5 を取り込み、どの入力要素のゾーン 5 8 に、もしあれば、ストロークが交差しているかを判別する。関連の入力要素がテキストフィールド 8 7 8 であると仮定すると、ページサーバはストロークをテキストフィールドのデジタルインクに添える。

【 0 3 1 3 】

テキストフィールドのゾーンにおける不活動の期間後、ページサーバはペン I D 及び懸案のストロークを登録サーバ 1 1 に送信し、解釈させる。登録サーバはペンに対応する利用者を識別し、利用者の累積された肉筆モデル 8 2 2 を使用して、ストロークを肉筆のテキストとして解釈する。一旦登録サーバがストロークをテキストに変換すると、登録サーバは、要求しているページサーバに、テキストを返却する。ページサーバはテキストをテキストフィールドのテキスト値に添える。

【 0 3 1 4 】

5 . 4 署名検証プロトコール

入力要素が、そのゾーンにストロークが交差しているが、署名フィールド 8 8 0 であると仮定すると、ページサーバ 1 0 はストロークを署名フィールドのデジタルインクに添える。

【 0 3 1 5 】

署名フィールドのゾーンにおける不活動の期間後、ページサーバはペン I D 及び懸案のストロークを登録サーバ 1 1 に送り、検証させる。ページサーバはまた、署名フィールドがその一部であるフォームに関連したアプリケーション I D 6 4、並びにフォーム I D 5 6 及びフォームの現状データ内容を送る。登録サーバは、ペンに対応する利用者を識別し、利用者の動的な署名生物形質 8 1 8 を使用して、ストロークを利用者の署名として検証する。一旦登録サーバが署名を検証し終わると、登録サーバはアプリケーション I D 6 4 及び利用者 I D 6 0 を使用して利用者のアプリケーション固有の秘密署名キーを識別する。次いで、登録サーバはそのキーを使用してフォームデータのデジタル署名を生成し、要求しているページサーバにデジタル署名を返却する。ページサーバはデジタル署名を署名フィールドに割り当て、関連のフォームの状況をフリーズされている状態に設定する。

【 0 3 1 6 】

デジタル署名は該当の利用者のエイリアス I D 6 5 を含む。これにより、単一のフォームが多数の利用者の署名を捕獲することができる。

【 0 3 1 7 】

5 . 5 フォーム送信プロトコール

フォーム送信プロトコールの好適な実施例を図 4 6 に示す。

【 0 3 1 8 】

フォーム送信はフォームハイパーリンク起動を介して生じる。かくして、フォーム配信はセクション 5 . 2 で定義されたプロトコールに従い、いくらかのフォーム固有のものが追加される。

【 0 3 1 9 】

フォームハイパーリンクの場合、ページサーバ 1 0 によりアプリケーション 7 1 に送信されたハイパーリンク起動メッセージもフォーム I D 5 6 及びフォームの現状データ内容を含む。もしフォームが任意の署名フィールドを含むと、その時、アプリケーションは、該当のデジタル署名に関連したエイリアス I D 6 5 を抜粋し、及び登録サーバ 1 1 から該当の証書を取得することにより、各署名フィールドを検証する。

【 0 3 2 0 】

5 . 6 手数料決済プロトコール

手数料決済プロトコールの好適な実施例を図 4 7 に示す。

10

20

30

40

50

【0321】

Eコマース環境において、クリックスルー、取引及び販売についてのフィー及び手数料は、アプリケーション プロバイダから出版者に決済可能である。フィーについての手数料及び手数料についての手数料も出版者からプリンタのプロバイダに決済可能である。

【0322】

ハイパーリンク要求ID52を使用して、フィー又は手数料貸方を、目標アプリケーション プロバイダ70a(例えば、商人)からソースアプリケーション プロバイダ70b(例えば、出版者)へ、及びソース アプリケーション プロバイダ70bからプリンタ プロバイダ72へ発送する。

【0323】

セクション5.2で説明したように、ハイパーリンクが最初に起動すると、目標アプリケーションは、ページ サーバ10からハイパーリンク要求IDを受信する。目標アプリケーションが、ソース アプリケーション プロバイダの貸方に記入する必要がある時、目標アプリケーションはアプリケーション プロバイダ貸方を本来のページ サーバにハイパーリンク要求IDと共に送る。ページ サーバはハイパーリンク要求IDを使用してソースアプリケーションを識別し、貸方を、ソース アプリケーションID64、それ自体のサーバID53、及びハイパーリンク要求IDと共に関連の登録サーバ11に送る。登録サーバは該当のソース アプリケーション プロバイダの口座827の貸方に記入する。登録サーバはアプリケーション プロバイダにも通知する。

【0324】

アプリケーション プロバイダは、もしプリンタ プロバイダの貸方に記入することが必要であれば、アプリケーション プロバイダは、プリンタ プロバイダ貸方をハイパーリンク要求IDと共に、本来のページ サーバに送る。ページ サーバはハイパーリンク要求IDを使用してプリンタを識別し、プリンタIDと共に、貸方を関連の登録サーバに送る。登録サーバは該当のプリンタ プロバイダの口座814の貸方に記入する。

【0325】

ソース アプリケーション プロバイダは目標アプリケーション プロバイダの識別を任意に通知され、プリンタ プロバイダはソース アプリケーション プロバイダの識別を任意に通知される。

【0326】

6 ネットページ ペン記述

6.1 ペン メカニズム

図8及び9を参照して、通常参照数字101により指定されるペンは、ペン コンポーネントを装着するための内部スペース104の輪郭を規定する壁103を有する、プラスチック モールドイングの形状をしたハウジング102を含む。ペン上部105は、ハウジング102の一端106において、操作時に回転自在になるように装着される。半透明のカバー107は、ハウジング102の反対側の端108に固定される。カバー107もモールド プラスチックで、利用者がハウジング102内に装着されたLEDの状況を見ることができるように、半透明材料で形成される。カバー107は、ハウジング102の端108を実質的に取り囲む主部109と、主部109から後方に突出し、ハウジング102の壁103に形成された該当のスロット111内に嵌る突出部110とを含む。無線アンテナ112は、突出部110の背後で、ハウジング102内に装着される。カバー107上でアパーチャ113Aを取り囲むネジ山113は、金属末端部114を受容するように配置され、該当のネジ山115を含む。金属末端部114は脱着自在で、インク カートリッジの交換が可能である。

【0327】

屈曲するPCB117上の3色状況LED116も、カバー107内に装着される。アンテナ112も屈曲するPCB117上に装着される。状況LED116は、良好な全周視界のためペン101の上部に装着される。

【0328】

10

20

30

40

50

ペンは、通常のマーキング インク ペン及び非マーキング針の両方として作動できる。先端 1 1 9 付きインク ペン カートリッジ 1 1 8 及び針先端 1 2 1 付き針 1 2 0 は、ハウジング 1 0 2 内に並んで装着される。ペン上部 1 0 5 を回転させることにより、インク カートリッジ先端 1 1 9 又は針先端 1 2 1 のいずれかが、金属末端部 1 1 4 の開口端 1 2 2 を通って前方に来ることができる。それぞれのスライダ ブロック 1 2 3 及び 1 2 4 はインク カートリッジ 1 1 8 及び針 1 2 0 にそれぞれ装着される。回転可能なカムバレル 1 2 5 は、操作時ペン上部 1 0 5 に固定され、ペン上部とともに回転するように配置される。カム バレル 1 2 5 は、カム バレルの壁 1 8 1 内に、スロットの形状をしたカム 1 2 6 を含む。スライダ ブロック 1 2 3 及び 1 2 4 から突き出るカム フォロア 1 2 7 及び 1 2 8 は、カム スロット 1 2 6 内に嵌る。カム バレル 1 2 5 の回転につれて、スライダ ブ

10

- ・ 針 1 2 0 先端 1 2 1 外、
- ・ インク カートリッジ 1 1 8 先端 1 1 9 外、及び
- ・ インク カートリッジ 1 1 8 先端 1 1 9、針 1 2 0 先端 1 2 1 のいずれも外ではない

【 0 3 2 9 】

第 2 の屈曲 P C B 1 2 9 は、ハウジング 1 0 2 内に据えられる電子シャーシ 1 3 0 上に装着される。第 2 の屈曲 P C B 1 2 9 は、表面に赤外線を投射する、赤外線 L E D 1 3 1 を装着する。画像センサ 1 3 2 は、屈曲 P C B 1 2 9 に装着されて、表面からの反射線を受容する。第 2 の屈曲 P C B 1 2 9 はまた、R F 発信器及び R F 受信機を含む無線周波チップ 1 3 3、及びペン 1 0 1 の動作を制御するコントローラ チップ 1 3 4 を装着する。光学ブロック 1 3 5 (モールド透明プラスチックから形成された) はカバー 1 0 7 内に据えられ、赤外線ビームを表面に投射し、画像を画像センサ 1 3 2 上で受信する。電力供給線 1 3 6 は、第 2 の屈曲 P C B 1 2 9 上のコンポーネントを、カム バレル 1 2 5 内に組み込まれている電池端子 1 3 7 に接続する。ターミナル 1 3 8 は、電池端子 1 3 7 とカム

20

30

【 0 3 3 0 】

ラバー グリップ パッド 1 4 1 及び 1 4 2 がハウジング 1 0 2 の端 1 0 8 に向かって備えられ、ペン 1 0 1 を握るのを助け、上部 1 0 5 もペン 1 0 1 をポケットに留めるクリップ

40

【 0 3 3 1 】

6 . 2 ペン コントローラ

ペン 1 0 1 は、その先端近辺の表面のエリアを赤外線スペクトルで画像化することにより、その先端 (針先端 1 2 1 又はインク カートリッジ先端 1 1 9) の位置を判別するように配置される。ペンは、最も近い場所タグから場所データを記録し、光学ブロック 1 3 5 及びコントローラ チップ 1 3 4 を利用して、先端 1 2 1 又は 1 1 9 の場所タブからの距離を算出するよう配置される。コントローラ チップ 1 3 4 は、画像タグ上で観察された透視歪みから、ペンの方位及び先端からタグまでの距離を算出する。

【 0 3 3 2 】

50

R Fチップ133及びアンテナ112を利用して、ペン101は、デジタル インク データ(セキュリティのため暗号化され、効率的な送信のためパッケージ化される)を算出システムに送信できる。

【0333】

ペンが受信器の範囲内であれば、デジタル インク データは形成されるにつれて、送信される。ペン101が範囲外へ動くと、デジタル インク データはペン101内でバッファされ(ペン101回路はデジタル インク データを約12分間の表面上でのペン動作を記憶するように準備されたバッファを含む)、後で送信できる。

【0334】

コントローラ チップ134は、ペン101内の第2の屈曲PCB129に装着される。図10は、コントローラ チップ134のアーキテクチャをより詳細に示すブロック図である。図10はまた、R Fチップ133、画像センサ132、3色状況LED116、IR照明LED131、IR力センサLED143、及び力センサ フォトダイオード144の表現を示す。

10

【0335】

ペン コントローラ チップ134は制御プロッセサ145を含む。バス146は、コントローラ チップ134のコンポーネント間におけるデータの交換を可能にする。フラッシュメモリ147及び512KB DRAM148も含まれる。アナログ デジタル変換器149は、力センサ フォトダイオード144からのアナログ信号をデジタル信号に変換するように配置される。

20

【0336】

画像センサ インタフェース152は、画像センサ132とインタフェースする。トランシーバ コントローラ153及びベースバンド回路154はまた、R F回路155と、アンテナ112に接続されたR F共振子及び誘導子156とを含むR Fチップ133とインタフェースするように含まれる。

【0337】

制御プロッセサ145は、画像センサ132を介して、タグからの場所データを表面から捕獲し、復号化し、力センサ フォトダイオード144を監視し、LED116と、131と、143とを制御し、無線トランシーバ153を介して短距離無線通信を扱う。制御プロッセサは中間性能(~40MHz)の汎用RISCプロッセサである。

30

【0338】

プロッセサ145、デジタル トランシーバ コンポーネント(トランシーバ コントローラ153及びベースバンド回路154)、画像センサ インタフェース152、フラッシュメモリ147、及び512KB DRAM148は、単一コントローラASIC内に一体化される。アナログR Fコンポーネント(R F回路155及びR F共振子及び誘導子156)は、別のR Fチップ内に備えられる。

【0339】

画像センサは、IRフィルタ付きの215x215画素CCDである(かかるセンサは松下電器により生産され、板倉、ケイ ティ 信定、エヌ 奥千也、アール 永吉、エム 尾崎による論文「小型カメラシステム用1mmの50K画素IT CCD画像センサ」、電子デバイスに関するIEEE報告、Vol. 47、ナンバー1、2000年1月に説明されており、それはここに引用して援用する)。

40

【0340】

ペン101が表面に接触していないと、コントローラASIC134は不活動の期間後、休止状態に入る。コントローラASICは、力センサ フォトダイオード144を監視し、パワー マネージャ151を介してペン ダウン事象でコントローラ134を起動する専用回路150を組み入れる。

【0341】

無線トランシーバは、コードレス電話により通常使用されている非ライセンスの900MHz帯で、又は代替として非ライセンスの2.4GHz産業、科学及び医療(ISM)帯

50

で通信し、周波数ホッピング及び衝突検出を使用して干渉フリー通信を提供する。

【0342】

代替の実施例において、ペンは基地局又はネットページプリンタとの短距離通信用に、赤外線データ通信標準化協会 (IrDA) インタフェースを組み入れる。

【0343】

更なる実施例において、ペン101はペン101軸心の垂直面に装着された一对の直交加速度計を含む。加速度計190は図9及び10にゴースト線で示される。

【0344】

加速度計の装備により、ペン101のこの実施例では、表面場所タグを参照しないで動きを感知することが可能になり、場所タグをより低い率でサンプリングしてもすむようになる。その時、各場所タグIDは表面上の位置よりもむしろ興味の対象を識別できる。例えば、もしオブジェクトが利用者インタフェース入力要素 (例えばコマンドボタン) であれば、その時、入力要素のエリア内の各位置付けタグのタグIDは、入力要素を直接識別できる。

10

【0345】

x及びy方向の各々で加速度計により測定された加速は、時間に関して積分され瞬間的な速度及び位置を生産する。

【0346】

ストロークの開始位置は未知なので、ストローク内の相対的位置のみを算出する。位置の積分化により感知された加速の誤差は累積されるが、加速度計は一般的には高解像力を有し、誤差が累積するストロークの持続時間は短い。

20

【0347】

7 ネットページプリンタ記述

7.1 プリンタメカニズム

垂直装着のネットページ壁プリンタ601は、図11に完全に組立てられた状態で示される。それは、図12及び12aに示すように、複式8 1/2" MemjetTM プリントエンジン602及び603を使用して、レター/A4サイズの媒体上にネットページを印刷する。それは、真っ直ぐな紙路を使用し、紙604はフルカラー及び完全な裁ち幅状態でシートの両面を同時に印刷する、複式プリントエンジン602及び603を通過する。

30

【0348】

一体式製本アセンブリ605は、糊の帯を各印刷シートの一方の端に沿って供給し、印刷シートが、前のシートに対して押された時、そのシートと接着できるようになる。これにより、厚さが1シートから数百枚のシートに及ぶ最終の装丁ドキュメント618が作成される。

【0349】

複式プリントエンジンに結合した、図13に示した交換可能なインクカートリッジ627は、色留め剤、糊、シアン、マゼンタ、黄、黒、及び赤外線インクを貯蔵する袋又は室を有する。カートリッジはまたベースモルディング内に超小型空気フィルタを有する。超小型空気フィルタは、プリンタの内部でホース639を介して空気ポンプ638に繋ぎ合わさる。これにより、濾過空気が印刷ヘッドに提供され、空気を濾過しないと印刷ヘッドノズルを詰まらすである超小型粒子のMemjetTM印刷ヘッド350への浸入を防ぐ。空気フィルタをカートリッジ内に組み入れることにより、フィルタの操作寿命がカートリッジの寿命と効果的にリンクされる。インクカートリッジは3000ページ (1500枚) を印刷し、接着する容量を持つ、完全にリサイクル可能な商品である。

40

【0350】

図12を参照すると、動力付き媒体取り出しローラアセンブリ626は、最上部シートを、媒体トレイから第1のプリントエンジン602上の紙センサを通過して、複式MemjetTM印刷ヘッドアセンブリ内に直接押す。2つのMemjetTMプリントエンジン602及び603は、真っ直ぐな紙路に沿って一線に向かい合う逐次的な構成で装

50

着される。紙 604 は、一体の動力付き取り出しローラ 626 により第 1 のプリント エンジン 602 に引き込まれる。紙 604 の位置及びサイズは感知され、裁ち幅一杯の印刷が開始される。色留め剤が同時に印刷され、可能な最短時間の乾燥を助ける。

【0351】

紙は、ゴムを引いたローラに対抗して作動する、動力付き出口スパイク ホイールのセット（一直線の紙路に沿って配列した）を通過して、第 1 の MemjetTM プリント エンジン 602 を出る。これらのスパイク ホイールは“湿った”印刷表面に接触し、シート 604 を第 2 の MemjetTM プリント エンジン 603 に供給し続ける。

【0352】

図 12 及び 12a を参照して、紙 604 は複式プリント エンジン 602 及び 603 から製本アセンブリ 605 へと通過する。印刷されたページは、繊維状のサポート ローラを持つ動力付きスパイク ホイール回転軸 670 と、スパイク ホイール及び瞬間的に動くグルー ホイールを持つ他の可動回転軸との間を通過する。可動回転軸/グルーアセンブリ 673 は、金属のサポート ブラケットに装着され、前方に搬送され、ギアを介してカムシャフトの動きにより動力付き回転軸 670 と繋ぎ合わさる。別のモータがこのカムシャフトを駆動する。

10

【0353】

グルー ホイール アセンブリ 673 は、インク カートリッジ 627 からの糊供給ホース用の回転カップリング付きの部分的に窪んだ回転軸 679 からなる。この回転軸 679 は、放射状の穴を通して毛管作用により接着剤を吸収するグルー ホイールに接続する。鑄造ハウジング 682 は、開口部を前にしてグルー ホイールを取り囲む。旋回側鑄造物及びスプリング付き外側ドアは、金属ブラケットに取り付けられ、アセンブリ 673 の残りが前方に押されると、横方向に蝶番で動く。この作用により、グルー ホイールが鑄造ハウジング 682 の前部を通して露出する。張力スプリングがアセンブリを閉じ、不活動の期間中効果的にグルー ホイールに蓋をかぶせる。

20

【0354】

シート 604 がグルー ホイール アセンブリ 673 に進むと、製本アセンブリ 603 内の下方に搬送されるにつれて、接着剤が表側の（ドキュメントの最初のシートから離れた）一方の垂直端に適用される。

【0355】

7.2 プリンタ コントローラ アーキテクチャ

図 14 に示したように、ネットページ コントローラは、制御プロセッサ 750、工場設置又はフィールド設置のネットワーク インタフェース モジュール 625、無線レシーバ（トランシーバ コントローラ 753、ベースバンド回路 754、RF 回路 755、並びに RF 共振子及び誘導子 756）、二重ラスト画像プロセッサ（RIP）DSP 757、複式プリント エンジン コントローラ 760a 及び 760b、フラッシュメモリ 658、及び 64MB の DRAM 657 からなる。

30

【0356】

制御プロセッサは、ネットワーク 19、及びローカル ワイヤレス ネットページ ペン 101 との通信を扱い、ヘルプ ボタン 617 を感知し、利用者インタフェース LED 613 ~ 616 を制御し、RIP DSP 757 及びプリント エンジン/コントローラ 760 にフィードし、同期させる。これは中程度性能の汎用マイクロプロセッサからなる。制御プロセッサ 750 は高速シリアル バス 659 を介してプリント エンジン コントローラ 760 と通信する。

40

【0357】

RIP DSP は、ページ記述をネットページ プリンタの圧縮ページフォーマットにラスタ化し、圧縮する。各プリント エンジン コントローラはリアルタイムで（即ち、毎分 30 ページを越える）、ページ画像をその関連した MemjetTM 印刷ヘッド 350 に拡大し、ディザリングし、印刷する。複式プリント エンジン コントローラはシートの両面を同時に印刷する。

50

【0358】

マスター プリント エンジン コントローラ760aは紙搬送を制御し、マスター チップ665及びインク カートリッジQAチップ761と共にインクの使用を監視する。

【0359】

プリント コントローラのフラッシュ メモリ658は、プロセッサ750及びDSP用のソフトウェアの両方、並びに構成データを保持する。これはブート時にメイン メモリ657にコピーされる。

【0360】

プロセッサ750、DSP757、及びデジタル トランシーバ コンポーネント(トランシーバ コントローラ753及びベースバンド回路754)は、単一コントローラASIC656で一体化される。アナログRFコンポーネント(RF回路755、及びRF共振子及び誘導子756)は別のRFチップ762で提供される。ネットページ プリンタにより、ネットワーク接続を工場選択にする又はフィールド選択にすることが可能になるので、ネットワーク インタフェース モジュール625は別個のものとなる。フラッシュメモリ658及び2x256Mビット(64MB)DRAM657もオフチップである。プリント エンジンコントローラ760は別個のASICに装備される。

【0361】

各々がネットページ ネットワーク インタフェース751を提供し、及びローカル コンピュータ又はネットワーク インタフェース752を任意に提供する、様々なネットワーク インタフェース モジュール625が提供される。ネットページ ネットワーク インタフェースは、POTSモデム、混成ファイバ・同軸(HFC)ケーブル モデム、ISDNモデム、DSLモデム、衛星トランシーバ、現及び次世代セルラ電話トランシーバ、及び無線ローカル ループ(WLL)トランシーバを含む。ローカル インタフェースはIEEE1284(パラレルポート)、10ベースT及び100ベースTのイーサネット、USB及びUSB2.0、IEEE1394(ファイアワイヤ)、及び種々の新興ホームネットワーク インタフェースを含む。インターネット接続がローカル ネットワーク上で利用できれば、その時、ローカル ネットワーク インタフェースをネットページ ネットワーク インタフェースとして使用できる。

【0362】

無線トランシーバ753は、コードレス電話により通常使用されている非ライセンスの900MHz帯で、又は代替として非ライセンスの2.4GHz産業科学医療(ISM)帯で通信し、周波数ホッピング及び衝突検出を使用して干渉フリー通信を提供する。

【0363】

プリンタ コントローラは、ネットページ カメラなどのデバイスから“噴出した”データを受信する赤外線データ通信標準化団体(IrDA)インタフェースを任意に組み入れる。代替の実施形態において、プリンタは適切に構成されたネットページ ペンとの短距離通信用のIrDAインタフェースを使用する。

【0364】

7.2.1 ラスタ化及び印刷

一旦メイン プロセッサ750がドキュメントのページ レイアウト及びページ オブジェクトを受信し、検証し終わると、それは適切なRIPソフトウェアをDSP757上で動かす。

【0365】

DSP757は各ページ記述をラスタ化し、ラスタ化ページ画像を圧縮する。メイン プロセッサは各圧縮ページ画像をメモリに記憶する。多数のDSPの負荷バランスを取る最も簡単な方法は、各DSPに個々のページをラスタ化させることである。任意の数のラスタ化ページを、一般に、メモリに記憶できるので、DSPを常にビジーに保つことができる。この戦略により、短いドキュメントをラスタ化する時、DSPの利用が貧弱になる可能性が生じるにすぎない。

【0366】

ページ記述内のすかし領域は、連続調解像力の二水準ビット マップにラスタ化され、それは損失なくごく僅かなサイズに圧縮され、且つ圧縮ページ画像の一部を形成する。

【0367】

印刷ページの赤外線 (I R) 層は、インチ当たり約 6 つの密度でコード化ネットページ タグを含む。各タグは、ページ I D、タグ I D、及び制御ビットを符号化し、各タグのデータ内容がラスタ化中に生成され、圧縮ページ画像に記憶される。

【0368】

メイン プロセッサ 7 5 0 は背中合せのページ画像を複式プリント エンジン コントローラ 7 6 0 に渡す。各プリント エンジン コントローラ 7 6 0 は、圧縮ページ画像をそのローカル メモリに記憶し、ページ拡大及び印刷パイプラインを始める。ページ拡大及び印刷は、1 1 4 M B の二水準 C M Y K + I R ページ画像全体をメモリ内に記憶することは実際的ではないので、パイプラインで送信される。

10

【0369】

7 . 2 . 2 プリント エンジン コントローラ

プリント エンジン コントローラ 7 6 0 のページ拡大及び印刷パイプラインは、高速 I E E 1 3 9 4 シリアル インタフェース 6 5 9、標準 J P E G デコーダ 7 6 3、標準 G 4 ファックス デコーダ 7 6 4、カスタム ハーフトーナ / コンポジタ ユニット 7 6 5 と、カスタム タグ エンコーダ 7 6 6 と、ライン ローダ / フォーマッタ ユニット 7 6 7、及び M e m j e t ^{T M} 印刷ヘッド 3 5 0 へのカスタム インタフェース 7 6 8 からなる。

【0370】

20

プリント エンジン コントローラ 3 6 0 はダブル バッファリング方式で作動する。一のページが高速シリアル インタフェース 6 5 9 を介して D R A M 7 6 9 に取り込まれる一方、前に取り込まれたページは D R A M 7 6 9 から読み出され、プリント エンジン コントローラ パイプラインを通過する。一旦ページが印刷を終了すると、取り込まれたばかりのページは印刷され、一方他のページが取り込まれる。

【0371】

パイプラインの第 1 のステージは J P E G 圧縮の連続調 C M Y K 層を拡大し (7 6 3 において)、G 4 ファックス圧縮の二水準の黒色層を拡大し (7 6 4 において)、二水準ネットページ タグ層をセクション 1、2 で定義されたタグ フォーマットに従い描画し (7 6 6 において)、これら全ては並列に行われる。第 2 のステージは、連続調 C M Y K 層のディザリングを行い (7 6 5 において)、その結果生じる二水準 C M Y K 層を覆って二水準黒色層を合成する (7 6 5 において)。その結果生じる二水準 C M Y K + I R 層ドットデータは、ライン バッファのセットを介して、バッファされ、M e m j e t ^{T M} 印刷ヘッド 3 5 0 上での印刷のためフォーマットされる (7 6 7 において)。これらのライン バッファの大部分はオフチップ D R A M に記憶される。最終ステージは印刷インタフェース 7 6 8 を介して、6 チャンネルの二水準ドット データ (色留め剤を含む) を M e m j e t ^{T M} 印刷ヘッドに印刷する。

30

【0372】

いくつかのプリント エンジン コントローラ 7 6 0 を、複式構成におけるが如く、一緒に使用する時、それらは共有ライン同期信号 7 7 0 を介して同期される。外部マスター / スレーブピン 7 7 1 を介して選択された一方のプリント エンジン 7 6 0 のみがライン同期信号 7 7 0 を共有ライン上に生成する。

40

【0373】

プリント エンジン コントローラ 7 6 0 は、ページ拡大及び描画パイプラインを同期化し、低速のシリアル バス 7 7 3 を介して印刷ヘッド 3 5 0 を構成し、ステッパ モータ 6 7 5 及び 6 7 6 を制御する低速のプロセッサ 7 7 2 を含む。

【0374】

ネットページ プリンタの 8 1 / 2 " バージョンにおいて、2 つのプリント エンジンの各々は、ページの長手方向 (1 1 ") に沿って、毎分 3 0 枚のレターページを印刷し、1 6 0 0 d p i で 8 . 8 k H Z のライン率を与える。ネットページ プリンタの 1 2 " バージョン

50

ジョンにおいて、2つのプリントエンジンの各々は、ページの短方向（8 1 / 2 "）に沿って、毎分45枚のレターページを印刷し、10.2 kHzのライン率を与える。これらのライン率は、現状の設計において30 kHzを越えるMemjetTM印刷ヘッドの操作周波数内に十分入る。

8 プリントエンジンコントローラ及びタグエンコーダ

一般的な12インチ印刷ヘッド幅は、以下に説明するように、1つ以上のPECにより制御され、A4及びレターページの両方の完全な全幅印刷を可能にする。6つのチャンネルのカラーインクは現在の印刷環境において予想される最大なものであり、これらは：

- ・ CMY, 通常の色印刷用
- ・ K, 黒色テキスト及び他の黒色印刷用
- ・ IR (赤外線)、ネットページ有効アプリケーション用
- ・ F (色留め剤)、高速印刷を可能にする

10

プリンタは高速印刷が可能であるべきなので、より高速で、次のページの印刷が完了し終える前にインクの乾燥が可能になるようにする色留め剤が必要である。そうでないと、ページはお互いに混じる可能性がある。より低速な印刷環境において色留め剤は必要ではないと思われる。

【0375】

PECは単一チップに組み込まれて印刷ヘッドとインタフェースすることができる。PECは4つの基本レベルの機能性を含む：

- ・ IEEE 1394などのシリアルインタフェースを介して圧縮ページを受信
- ・ 圧縮フォームからページを生産するプリントエンジン。プリントエンジン機能性はページ画像を拡大すること、連続調層のディザリングを行うこと、黒色層を連続調層上に合成すること、及びその結果生じる画像を印刷ヘッドに送信することを含む。
- ・ 印刷ヘッド及びステッパモータを制御するプリントコントローラ
- ・ 2つのQAチップとの通信用の2つの標準低速度シリアルポート。認証手順中、強力なセキュリティを保証するため、単一ポートではなく、2つのポートが必須であることに注目せよ。

20

【0376】

図48にはドキュメントをコンピュータシステムから印刷ページに送信するデータの流が見られる。ドキュメントは411で受信され、メモリバッファ412に取り込まれ、そこではページレイアウトの実行が可能で、任意の所要のオブジェクトが加えられるかもしれない。メモリ412からのページは、プリントエンジンコントローラ410に送信する前に、413でラスタ化され、414で圧縮される。ページは、プリントエンジンコントローラ410内で圧縮された二水準ページ画像として受信されて、メモリバッファ415に入り、そこからページ画像は、ページ画像が検索されるページ拡大器416に供給される。任意の必須のディザリングが、417で任意の連続調層に適用されるかもしれない。任意の黒色二水準の層が、419における任意の赤外線タグと共に連続調層上に418において合成することができるであろう。合成ページデータは420において印刷され、ページ421を生産する。

30

【0377】

プリントエンジン/コントローラは圧縮ページ画像を取り入れ、ページ拡大及び印刷をパイプライン様式で開始する。大きさを決められる二水準CMYK+IRページ画像をメモリ内に記憶することは実際的ではないので、ページ拡大及び印刷はパイプラインで送信することが好ましい。

40

【0378】

パイプラインの第1のステージはJPEG圧縮の連続調CMYK層を拡大し（以下参照）、G4ファックス圧縮の二水準ディザマトリックス選択マップを拡大し（以下参照）、G4ファックス圧縮の二水準黒色層を拡大し（以下参照）、これら全ては並列に行われる。第2のステージは、ディザマトリックス選択マップにより選択されたディザマトリックスを使用して、連続調CMYK層のディザリングを行い、その結果生じる二水準のK層

50

上に二水準の黒色層を合成する。これに平行して、タグ エンコーダは二水準 I R タグ データを圧縮ページ画像から符号化する。色留め剤層もまた、C、M、Y、K又はI Rチャンネルのいずれかで必要があれば、各ドット位置において生成される。最終ステージは印刷ヘッド インタフェースを介して印刷ヘッドを通して二水準 C M Y K + I R データを印刷する(以下参照)。

【0379】

図49では、プリント エンジン/コントローラ410が全体のプリンタ システム アーキテクチャ内にどのようにして嵌るかが見られる。種々のプリンタ システムのコンポーネントは下記を含むであろう。

・ プリント エンジン/コントローラ(P E C) . P E Cチップ410、又は複数のチップは圧縮ページ画像を受信してメモリ バッファ424に記憶し、ページ拡大を行い、黒色層合成をし、ドット データを印刷ヘッド423に送信する責任がある。それはまたQ Aチップ425及び426と通信を行い、印刷ヘッドの特徴を検索しする手段を提供して、最適な印刷を保証する。P E Cはこの明細書の主題である。

・ メモリ バッファ . メモリ バッファ424は圧縮ページ画像を記憶し、所定のページの印刷中のスクラッチ使用のためのものである。メモリ バッファの製造及び作業は当業者に既知であり、それらを使用するための様々な標準チップ及び技術は、本発明のP E Cの使用において利用できるであろう。

・ マスターQ Aチップ . マスター チップ425は交換可能なインク カートリッジQ Aチップ426に合致する。Q Aユニットの構造及び作業は当業者に既知であり、様々な既知のQ Aプロセスは、本発明のP E Cの使用において利用できるであろう。例えば、Q Aチップは同時係属の米国特許出願に記載されている。

【0380】

【表4】

U S S N	整理番号	名称
T B A	A U T H 0 1	バリデーション プロトコール及びシステム
09/112,763	A U T H 0 2	I D Dゆらぎ攻撃に対するチップの保護回路
09/112,737	A U T H 0 4	オンチップ メモリ (フラッシュ及びR A M) の保護方法
09/112,761	A U T H 0 5	チップ改ざん防止方法
09/113,223	A U T H 0 6	物理的なオブジェクトの認証システム
T B A	A U T H 0 7	バリデーション プロトコール及びシステム
T B A	A U T H 0 8	バリデーション プロトコール及びシステム
09/505,003	A U T H 0 9	消費可能認証プロトコール及びシステム
09/517,608	A U T H 1 0	消費可能認証プロトコール及びシステム
09/505,147	A U T H 1 1	消費可能認証プロトコール及びシステム
09/505,952	A U T H 1 2	フラッシュ メモリに記憶されている値の非認可修正
T B A	A U T H 1 3	セキュア データの操作システム
09/516,874	A U T H 1 4	パワー サプライ攻撃からの保護付き認証チップ
T B A	A U T H 1 5	秘密データの遮蔽操作

【0381】

P E Cチップは画像を拡大し、かつ

物理的な印刷ヘッドを動かす役割を有しているので、Q Aチップ通信はP E Cチップの全体的な機能性内に最適に含まれるものである。Q Aチップ通信をそこに位置付けすることにより、ページを印刷するのに十分なインクがあることを保証できる。印刷ヘッドアセン

プリに埋め込まれた Q A が認証チップを使用して実行されることが好ましい。それはマスター Q A チップなので、認証キーのみを有し、利用者データを含まない。しかしながら、それはインク カートリッジの Q A チップに合致しなければならない。インク カートリッジ内の Q A チップは考え得る最良の印刷品質を維持するのに必要な情報を含み、認証チップを使用して実行される。

【 0 3 8 2 】

6 4 M ビット (8 M バイト) メモリ バッファを使用して圧縮ページ画像を記憶することが好ましい。一のページをバッファに書き込みながら、他のページが読み出される (ダブル バッファリング)。更に、 P E C は、ページを印刷中、メモリを使用して算出ドット情報をバッファする。ページ N を印刷中、バッファは下記に使用される：

10

- ・ 圧縮ページ N を読み出す
- ・ ページ N に対して二水準ドット情報を読み出し、書き込む
- ・ 圧縮ページ N + 1 を書き込む

P E C チップは単一のマイクロコントローラ C P U コア 4 3 5 を組み入れて、下記の機能を遂行することが好ましい。

- ・ 印刷ページ間の Q A チップ認証プロトコールを遂行する
- ・ 印刷中、ステップ モータをパラレル インタフェース 5 8 9 で動かす (ステップ モータは 5 K H z プロセスが必要である)
- ・ 印刷中、 P E C チップの種々の部分を同期化する
- ・ 外部のデータ要求 (プログラミング登録等) とインタフェースする手段を提供する
- ・ 印刷ヘッド セグメントの低速データ要求 (特徴づけベクトルの読み出しやパルス プロファイルの書き込みなど) とインタフェースする手段を提供する
- ・ 肖像画書式及び風景画書式タグ構造を外部 D R A M に書き込む手段を提供する。

20

【 0 3 8 3 】

画像プロセッシングは全て専用ハードウェアにより遂行されるので、 C P U は画素をプロセスする必要がない。その結果、 C P U は極端に単純になり得る。 C P U が既知の幅広い種類のコアが適切である：それは要求される算出及び制御機能を十分早く遂行するための十分なプロセッシング パワーを持つ、任意のプロセッサ コアであり得る。適切なコアの例は約 1 M H z で作動するフィリップス 8 0 5 1 マイクロコントローラである。プログラム R O M 及びスモール プログラム スクラッチ R A M が C P U コア 4 3 5 に関連する。 C P U はメモリマップ I / O を介して P E C チップ内の他のユニットと通信する。特別のアドレス範囲が特別のユニットにマッピングされ、各範囲内で、その特別のユニット内の、特別の範囲にマッピングされる。これはシリアル及びパラレル インタフェースを含む。スモール プログラム フラッシュ R O M を P E C チップに組み入れることができる。そのサイズは選択した C P U によるが、 8 K B を越えないことが望ましい。同様に、スモール スクラッチ R A M エリアを P E C チップに組み入れることができる。プログラム コードは画像を操作する必要がないので、大きなスクラッチ エリアの必要性はない。 R A M サイズは選択した C P U (スタック メカニズム、サブルーチン呼び込み規定、レジスタ サイズ等) によるが、約 2 K B を越えないことが望ましい。

30

【 0 3 8 4 】

上記に言及したセグメント ベースのページ幅印刷ヘッドを使用する P E C チップは、黒色を完全なドット解像力 (一般的には、 1 6 0 0 d p i) で再現できるが、網点化を使用して連続調 カラーをいくらか低い解像力で再現する。それ故、ページ記述は黒色二水準の層と連続調層とに区分される。黒色二水準の層は連続調層を覆って合成するように定義される。黒色二水準の層は各画素に対して 1 ビットの不透明度を含むビットマップからなる。この黒色層マップは、プリンタのドット解像力の整数倍である解像力を有する。最高にサポートされた解像力は 1 6 0 0 d p i、即ちプリンタのフルドット解像力である。連続調層は、各画素に対して 3 2 ビットの C M Y K カラーを含むビットマップからなり、ここでは K は随意である。この連続調画像は、プリンタのドット解像力の整数倍である解像力を有する。最高にサポートされた解像力は各単一 E P C に対して 1 2 インチにわたって

40

50

320 dpi、即ちプリンタのドット解像力の5分の1である。より高い連続調解像力に対しては多数のPECが必要であり、各PECは出力ページの帯を生産する。連続調解像力も、一般的には黒色二水準解像力の整数倍であり、RIPにおける算出が簡単になる。しかしながら、このことは必要条件ではない。黒色二水準の層及び連続調層は、両方ともプリンタの内部メモリにおいては、効率的な記憶のため圧縮フォームである。

【0385】

図50に、プリントエンジンアーキテクチャが見られる。プリントエンジンのページ拡大及び印刷パイプラインは、高速シリアルインタフェース427（標準IEEE1394インタフェースなど）、標準JPEGデコーダ428、標準G4ファックスデコーダ、カスタムハーフトーナ/コンポジタユニット429、カスタムタグエンコーダ430、ラインローダ/フォーマッタユニット431、及び印刷ヘッド433へのカスタムインタフェース432からなる。デコーダ428及び588とエンコーダ430は、ハーフトーナ/コンポジタ429にバッファされる。

【0386】

タグエンコーダ430は、ページになにが使用されるかにより決まるプロトコールに従って、赤外線タグ又は複数のタグをページに設定し、そしてタグの実際の内容は本発明の主題ではない。

【0387】

プリントエンジンはダブルバッファ手法で作業する。一のページがバス586上のDRAMインタフェース587、及び高速シリアルインタフェース27を介してDRAM34に取り込まれ、一方前に取り込まれたページはDRAM434から読み出され、プリントエンジンパイプラインを通る。一旦ページが印刷を終了すると、次いで、取り込まれたばかりのページは印刷されるページとなり、新しいページが高速シリアルインタフェース427を介して取り込まれる。第1のステージにおいて、パイプラインは任意のJPEG圧縮の連続調(CMYK)層を拡大し、2つのG4ファックス圧縮の二水準データストリームのいずれも拡大する。2つのストリームは、黒色層(PECは、実際はカラー不可知論的であり、この二水準の層は出力インクのいずれにも向けることができるけれども)、及び連続調ディザリングのためディザマトリックス間で選択するマットである(下記参照)。第2のステージにおいて、第1と平行して、後でIR又は黒色インクのいずれかで描画するため任意のタグが符号化される。最後に、第3のステージは連続調層のディザリングを行い、その結果生じる二水準ディザリング層にわたって、位置タグ及び二水準のスポット1層を合成する。データストリームは、印刷ヘッド内の重複セグメントを横切って円滑な遷移を創作するように調整されるのが理想的であり、また印刷ヘッド内の故障したノズルを補償するように調整されるのが理想的である。6つのチャンネルまでの二水準データがこのステージから生産される。6つのチャンネル全部は印刷ヘッド上にないかもしれないことに注目せよ。例えば、印刷ヘッドは、KをCMYチャンネルに押し入れ、IRを無視して、CMYのみでもよい。代替として、もしIRインクが利用できなければ(又は試験目的のため)、位置タグをKで印刷してもよい。その結果生じる二水準CMYK-IRドットデータは、ラインバッファのセットを介して印刷ヘッド33上での印刷のためバッファされ、フォーマットされる(下記参照)。これらのラインバッファの大部分はオフチップDRAM434上で記憶されるのが理想的であろう。最終ステージは印刷ヘッドインタフェース432を介して、6つのチャンネルの二水準ドットデータを印刷する。

【0388】

PECを用いる印刷システムでは圧縮を使用する。これはデータフローが、一定の速度で作動する印刷ヘッドより、先行することを可能にするためである。267 dpiでは、連続調CMYKデータのレターページは、25MBのサイズを有する。JPEGなどの損失の多い連続調圧縮アルゴリズムを使用して(下記参照)、連続調画像は際立つ品質の損失がなく10:1までの率で圧縮し、2.5MBの圧縮ページサイズが得られる。800 dpiでは、二水準データの書簡ページは、7MBのサイズを有する。テキストなどのコ

ヒーレント データは十分に圧縮される。G4ファクシミリなどの損失のない二水準圧縮アルゴリズムを使用して(下記参照)、10ポイントのテキストは約10:1率で圧縮され、0.8MBの圧縮ページサイズが得られる。一旦ディザリングされると、CMYK連続調画像データのページは114MBの二水準データからなる。以下に説明した二層圧縮ページ画像フォーマットは、損失の多いPEG連続調画像圧縮と、損失のない二水準テキスト圧縮との相対強さを利用する。フォーマットは十分にコンパクトで記憶効率的であり、十分にシンプルで、印刷中に直接的なリアルタイム拡大が可能となる。テキスト及び画像は通常重複しないので、通常の最悪ケースのページ画像サイズは2.5MBである(即ち、画像のみ)、一方、通常の最良ケースのページ画像サイズは0.8MBである(即ち、テキストのみ)。絶対的な最悪ケースのページ画像サイズは3.3MBである(即ち、画像を覆うテキスト)。平均ページの4分の1は画像を含むと仮定すると、平均画像サイズは1.2MBである。

【0389】

G4ファックス デコーダは、二水準データを復元(デコンプレッション)する働きをする。二水準データは単一スポット カラー(一般的には、テキスト及び線画に対して黒色)及び以降の連続調データ(JPEGデコーダにより復元された)のディザリングに使用されるディザマトリックス選択ビットマップに制限される。G4ファックス デコーダへの入力、外部DRAMから読み込まれた二水準データの2つの面である。G4ファックス デコーダの出力は、復元された二水準データの2つの面である。復元された二水準データは、印刷パイプラインにおける次のステージに向けて、ハーフトーナ/コンポジットユニット(HCU)に送られる。2つの二水準バッファは、二水準データをG4ファックス デコーダとHCUとの間で転送する手段を提供する。各復元された二水準層は、2つのライン バッファに出力される。各バッファは、予想最大解像力で、完全な12インチラインのドットを保持できる。2つのライン バッファを有することにより、一方のラインがHCUにより読み出され、一方他のラインはG4ファックス デコーダにより書き込まれることができる。これは、単一の二水準ラインは一般的には1600dpi未満であり、それ故ドット及びライン寸法の両方で拡大しなければならないので、重要である。バッファリングが1つのフルライン未満であれば、G4ファックス デコーダは同一ラインを多数回 - 各出力600dpiドットラインに対して一回、復号化する必要があるだろう。

【0390】

スポット カラー1は、出力画像の単一カラー面に対して高解像力ドット データを許容するように設計される。連続調層が画像に対して十分な解像力を提供する一方、スポット カラー1はテキスト及び線画(一般的には黒色)などのアプリケーションにおいて目標となる。テキスト及び線画として使用されると、一般的な圧縮率は10:1を越える。スポット カラー1により、最高印刷品質に対して1600dpiまでの可変解像力が可能となる。2つのライン バッファの各々は、それ故、合計2400バイト(12インチ×1600dpi=19,200ビット)である。

【0391】

ディザマトリックス選択マップの解像力は、理想的には連続調解像力に合致すべきである。その結果、2つのライン バッファの各々は、それ故、480バイト(3840ビット)であり、12インチを320dpiで記憶できる。マップが連続調解像力に合致すると、一般的な圧縮率は50:1を超える。

【0392】

下記に対するサポートを提供するため

- ・ 800dpiスポット カラー1層(一般的には黒)
- ・ 320dpiディザマトリックス選択層

復元バンド幅必要条件は、毎秒1ページの性能に対して9.05MB/秒で(ページ幅が12インチか又は8.5インチかに関わらず)、最高印刷速度性能の間(毎秒30,000ライン)、12インチ及び8.5インチページ幅に対して、それぞれ20MB/秒及び

14.2 MB / 秒である。復元されたデータがライン バッファに出力されるとすると、G4 ファックス デコーダは、出力の各々からのラインを一度に1つ、直ぐに復元することができる。

【0393】

G4 ファックス デコーダは、DRAM インタフェースを介してメインメモリから直接リードされる。圧縮の量により外部DRAMへのバンド幅必要条件が決まる。G4 ファックスは損失無しなので、画像の複雑性はデータの量、それによりバンド幅に強い影響をあたえる。800 dpi 黒色テキスト/グラフィックス層は、一般的には10:1 圧縮を越え、それ故毎秒1ページを印刷するのに必要なバンド幅は0.78 MB / 秒である。同様に、一般的な320 dpi デザインマトリックスは50:1 を越えて圧縮され、その結果0.025 MB / 秒バンド幅となる。デザインマトリックス用の320 dpi 及びスポットカラー1用の800 dpi の最速印刷速度設定は、1.72 MB / 秒及び0.056 MB / 秒をそれぞれ必要とする。それ故、2 MB / 秒の全バンド幅はDRAM バンド幅に対して十二分である。

10

【0394】

G4 ファックス復号化機能性は、G4 ファックス デコーダ コア的手段により実行される。さまざまなG4 ファックス デコーダ コアが適している：それは所要の算出及び制御機能を十分に早く遂行するのに十分なプロセッシング パワーを持つ、任意のコアであり得る。それは400 dpi ファクシミリ アプリケーションで通常出会うランレングスを超えるランレングスを取り扱うことができなければならない、それで修正が必要かもしれない。

20

【0395】

CMYK (又はCMY) 連続調層は、平面カラー JPEG バイトストリームに圧縮される。もし輝度/クロミナス分離が、表共有の目的又はクロミナス サブサンプリングのいずれかのために必要とみなされれば、その時、CMYKをYCrCbに変換し、CrとCbは正当にサブサンプリングされる。JPEG バイトストリームは完成し、自己完結している。それは、量子化及びハフマン表を含む、復元に必要な全てのデータを包含する。

【0396】

JPEG デコーダは、連続調データ層のオンザフライ復元を遂行する責任を負う。JPEG デコーダへの入力は、連続調データの4つの面までである。これは一般的には3つの面であり、CMY 連続調画像を表現し、又は4つの面で、CMYK 連続調画像を表現する。一般的には全てのカラー面は同じ解像力であろうが、各カラー面は異なる解像力であり得る。連続調層は外部DRAMから読み出される。JPEG デコーダの出力は復元された連続調データであり、面に分離されている。復元された連続調画像はハーフトーン/コンボジタ ユニット (HCU) 429 に送信され、印刷パイプラインにおける次のステージ向かう。4面の連続調バッファにより、連続調データをJPEG デコーダとHCU 429との間で転送する手段が提供される。

30

【0397】

復元された連続調データの各カラー面は、2つのライン バッファのセットに出力される (下記参照)。各ライン バッファは3840 バイトで、それ故、12インチの単一カラー面の画素を320 dpi で保持できる。ライン バッファリングにより、一のライン バッファがHCUにより読み出され、一方他のライン バッファはJPEG デコーダにより書き込まれることができる。これは、単一の連続調ラインは一般的には1600 dpi 未満であり、それ故ドット及びライン寸法ンの両方で拡大しなければならないので、重要である。バッファリングがフルライン未満であれば、JPEG デコーダは同一ラインを多数回 - 各出力600 dpi ドット ラインに対して一回、復号化する必要があるだろう。種々の解像力をサポートしても、解像力と利用可能なバンド幅との間で相殺がある。解像力及びカラーの数が増加するにつれて、バンド幅必要条件も増加する。更に、PECチップの標的となるセグメントの数もバンド幅に影響し、恐らく解像力にも影響する。連続調画像は平面フォーマットでプロセスされるので、各カラー面は異なる解像力で (例えば、CMYはK面より高解像力であるかもしれない) 記憶できることに注目せよ。最高にサポート

40

50

された連続調解像力は1600 dpi (プリンタのフルドット解像力に合致)である。しかしながら、長さ12インチの320 dpiラインに対して十分な連続調画素を保持するに十分な出力ラインバッファメモリがあるにすぎない。もし完全な12インチの出力がより高い連続調解像力で要求されたら、多数のPECチップが必要となるだろうが、しかし、プリンタ上の最終出力は依然として二水準のみであることに注目すべきであろう。320 dpiで4つのカラーをサポートした状態では、復元出力バンド幅必要条件は、毎秒1ページの性能に対して40MB/秒であり(ページ幅が12インチか又は8.5インチであるかにかかわらず)、最高プリンタ速度性能(毎秒30,000ライン)の間は、12インチ及び8.5インチページ幅に対してそれぞれ88MB/秒及び64MB/秒である。表5を使用して種々の解像力/カラー面/ページ幅の組合せに必要なバンド幅を決定できる。

10

【0398】

JPEGデコーダは、DRAMインタフェースを介してメインメモリから直接フィールドされる。圧縮の量により、外部DRAMへのバンド幅必要条件が決まる。圧縮のレベルが上がるにつれて、バンド幅は減少するが、最終出力画像の品質もまた減少し得る。単一のカラー面に対するDRAMバンド幅は、圧縮率を表5に示した出力バンド幅に適用することにより、容易に算出できる。例えば、320 dpiで10:1の圧縮率を持つ単一のカラー面には、毎秒一のページを生産するためにはDRAMへの1MB/秒アクセスが必要である。

【0399】

JPEG機能性はJPEGコアにより実行される。幅広い種類のJPEGコアが適切である:それは、所要の算出及び制御機能を十分早く遂行するための十分なプロセッシングパワーを持つ、任意のJPEGコアであり得る。例えば、BTGXマッチのコアは、140Mバイト/秒までの復元速度を有し、最高プリンタ速度(1600 dpiで毎秒30,000ライン)に対して、400 dpiまでの連続調解像力で、及び1ページ/秒のプリンタ速度に対しては800 dpiまでの連続調解像力で、4つのカラー面を復元可能にする。コアは復元をサポートするのみに必要であり、より汎用のJPEG圧縮/復元コアにより課せられる必要条件を下げることに注目せよ。コアのサイズは100,000ゲート以下であると予想される。復元データをラインバッファに出力するとすると、JPEGデコーダはカラー面の各々に対して1ライン全体を、一度に1つ、容易に復元でき、かくしてライン中の文脈切り換えを節約し、且つJPEGデコーダの制御を簡単にする。4つの文脈は保持しなげならず(各カラー面に対して1つの文脈)、且つ、適切なJPEG復号化パラメータとともに、外部DRAM内の現状アドレスを含む。

20

30

【0400】

図51において、ハーフトナー/コンポジットユニット(HCU)429は、連続調(一般的にはCMYK)層を、同二水準バージョンにハーフトナーリングする機能と、適切にハーフトナーリングされた連続調層(単数又は複数)上にスポット1二水準の層を合成する機能とを組み合わせる。もしもプリンタ内にKインクがなければ、HCU429はKをCMYに適切にマッピングすることができる。HCUはまた、画素の2つのディザマトリックス間で画素毎に、ディザマトリックス選択マップ内の該当の値を基にして、選択する。HCU429への入力は、バッファ437を通った(JPEGデコーダユニットからの)拡大された連続調層、バッファ438を通った拡大された二水準のスポット1層、バッファ439を通った、連続調層と一般的には同じ解像力の、拡大されたディザマトリックス選択ビットマップ、及びバッファ440を通った、フルドット解像力でのタグデータである。HCU429は2つまでのディザマトリックスを使用し、それは外部DRAMから読み出される。441でのHCU429からラインローダ/フォーマットユニット(LLFU)への出力は、6つまでのカラー面内の、プリンタ解像力二水準画像ラインのセットである。一般的には、連続調層はCMYK又はCMYで、二水準のスポット1層はKである。

40

【0401】

50

図52に、HCUがより詳細に見られる。一旦始動すると、HCUは、ページ終了条件を検出するまで、又はその制御レジスタを介して明示的に停止されるまで進む。HCUの第1のタスクは、442などのバッファ面で受信したデータ全てを、縮小/拡大ユニット443のようなそれぞれの縮小/拡大ユニットで、プリンタ解像力に、水平及び垂直の両方において、縮小/拡大する。縮小/拡大ユニットにより、連続調又は二水準データを、プリンタ解像力に、水平及び垂直の両方において、縮小/拡大する手段が提供される。縮小/拡大は、データ値を両方のディメンションで整数回複製することにより、達成される。データを縮小/拡大するプロセスは当業者にお馴染みである。連続調層の各々は異なる解像力であり得るので、連続調層は個別に縮小/拡大される。バッファ445における二水準スポット1層及びバッファ446におけるディザマトリックス選択層も、縮小/拡大する必要がある。バッファ447における二水準タグデータは、正しい解像力で設定されており、縮小/拡大する必要がない。拡大されたディザマトリックス選択ビット、はディザマトリックスアクセスユニット448により使用されて、2つのディザマトリックスから単一の8ビット値を選択する。8ビット値は、それを特定の8ビット連続調値と単に比較する、4つの比較器444及び449~451に出力される。実際のディザマトリックスの生成は印刷ヘッド構造により決まり、ディザマトリックス生成する一般的なプロセスは当業者にお馴染みである。もし連続調値が8ビットのディザマトリックス値以上であれば、1が出力される。そうでなければ、その時、0が出力される。次いで、これらのビットは452~456において、余白ユニット457からのインページビットとともに、全てAND処理される(特別のドットがページの印刷可能エリア内部に存在してもしなくても)。HCU内における最終ステージは合成ステージである。6つの出力層の各々に対して、ユニット458のような、各々が6つの入力を持つ、単一のドット合併ユニットがある。各ドット合併ユニットからの単一の出力ビットは、任意の又は全ての入力ビットの組合せである。これにより、スポットカラーは任意の出力カラー面に配置され(試験目的の赤外線を含む)、黒がシアン、マゼンタ、及び黄に混ざり(もし黒インクが印刷ヘッドになければ)、かつ、タグドットデータが目に見える面に配置されることが可能になる。色留め剤カラー面も容易に生成することができる。ドット再組織ユニット(DRU)459は、所定のカラー面に対して生成されたドットストリームを取り入れること、及び、出力がセグメント順になり、且つセグメント内ではドット順になるように、それを32ビットの量に組織することに、責任がある。重複セグメントに対するドットはセグメント順に生成されないので、最小の再順序化が必要である。

【0402】

2つの制御ビットが余白ユニット457により縮小/拡大ユニット457に提供される：アドバンスドット及びアドバンスライン。アドバンスドットビットにより、状態マシーンが同じドットデータの多数の事例(ページ余白、及びMemjetTM印刷ヘッド内で重複セグメントに対するドットデータを形成するのに有用)を生成できる。アドバンスラインビットにより、状態マシーンが特別のドットのラインがいつ終了したかを制御でき、それによりプリンタ余白に従うデータの打ち切りが可能になる。アドバンスラインビットはまた、縮小/拡大ユニットが特殊な終線ロジックを要求しなくてもよいようにする。

【0403】

比較器ユニットはシンプルな単一の8ビット“以上”比較器を含む。その比較器を使用して、8ビットの連続調値が8ビットのディザマトリックス値以上であるかどうか判別する。だから、比較器ユニットは2つの8ビット入力を取り入れ、単一の1ビット出力を生産する。

【0404】

図53に、ドット合併ユニットのより詳細が見られる。ドット合併ユニットは二水準のディザリングされたデータ、スポット1カラー、及びタグデータをマッピングし、実際の印刷ヘッド内のインクを出力する手段を提供する。各ドット合併ユニットは6つの1ビット入力を取り入れ、そのカラー面に対する出力ドットを表現する単一のビット出力を生産

する。460における出力ビットは、任意の又は全ての入力ビットの組合せである。これにより、スポット カラーは任意の出力カラー面に配置され（試験目的の赤外線を含む）、黒がシアン、マゼンタ、及び黄に混ざり（黒インクが印刷ヘッドにない場合）、かつタグ ドット データが目に見える面に配置されることが可能になる。色留め剤に対する出力は、入力ビットの全てを単に組み合わせることにより容易に生成できる。ドット合併ユニットは、6つの入力ビットに対するマスクとして使用される、6ビットのカラーマスクレジスタ461を含む。入力ビットの各々は、該当のカラーマスク レジスタ ビットとともにAND処理され、次いで、その結果生じる6ビットと一緒にOR処理され最終出力ビットを形成する。

【0405】

図54にドット再組織ユニット(DRU)が見られ、それは所定のカラー面に対して生成されたドット ストリームを取り入れることに、また、それを、出力がセグメント順になり、且つセグメント内ではドット順になるように、32ビットの量に組織することに、責任がある。重複セグメントに対するドットはセグメント順に生成されないことから、最小の再順序化が必要である。DRUは32ビットのシフト レジスタ、規則的な32ビットのシフト レジスタ、及び規則的な16ビットのシフト レジスタを含む。5ビットのカウンタは、それまでにプロセスされたビット数に絶えず注意する。ディザ マトリックス アクセス ユニット(DMAU)からのドット アドバンス信号を使用して、DRUにどのビットを出力すべきかについて指示する。

【0406】

図54において、レジスタ(A)462はサイクル毎に計時される。それは、ドット合併ユニット(DMU)により生産された最も新しい32のドットを含む。完全な32ビット値は、DRU状態マシン464により生産された書き込み可能信号によりシンプルな5ビットのカウンタを介して、レジスタ(B)463に32サイクル毎にコピーされる。レジスタ(B)463からの16の奇数ビット(ビット1、3、5、7等)は、レジスタ(C)465に同じ書き込み可能パルスでコピーされる。その時、32ビットのマルチプレックサ466は、状態マシンからの2つのビットに基づいて下記の3つの出力間で選択する：

- ・ レジスタBからの完全な32ビット
- ・ レジスタAの16の偶数ビット(ビット0、2、4、6等)及びレジスタBの16の偶数ビットから構成された32ビットの値。レジスタAからの16の偶数ビットはビット0~15を形成し、一方レジスタBからの16の偶数ビットはビット16~31を形成する。
- ・ レジスタBの16の奇数ビット(ビット1、3、5、7等)及びレジスタCの16のビットから構成された32ビットの値。レジスタCビットはビット0~15を形成し、一方レジスタBからの奇数ビットはビット16~31を形成する。

【0407】

DRUに対する状態マシンは、表1で見ることができる。それは状態0で開始する。それは32サイクル毎に状態を変える。32サイクル中、単一の非重複ビットは、それらの32サイクルに対するドット アドバンス ビットの全てのANDを集める(サイクル0に対して非重複=ドット アドバンス、サイクル1~31に対して非重複=非重複ANDドット アドバンス)。

表1 DRUに対する状態マシン

【0408】

【表5】

10

20

30

40

表 1

状態	非重複	出力	出力有効	コメント	次の状態
0	X	B	0	開始状態	1
1	1	B	1	レギュラ非重複	1
1	0	B	1	Aは第1の重複を含む	2
2	X	偶数A、偶数B	1	Aは第2の重複を含む Bは第1の重複を含む	3
3	X	C、奇数B	1	Cは第1の重複を含む Bは第2の重複を含む	1

10

【0409】

図52において、余白ユニット(MU)457は、現状ページのページ余白を基にして、ディザマトリックスアクセスユニット(DMAU)448からのアドバンスドット及びアドバンスライン信号を、一般の制御信号に変えることに責任を持つ。それはまた、ページ終了条件を生成することに責任を持つ。MUはページをまたがるドット及びラインのカウンタを保持する。両者は、ページの初めで0に設定される。ドットカウンタは、MUがドットアドバンス信号をDMAUから受信する毎に、1進む。MUがラインアドバンス信号をDMAUから受信すると、ラインカウンタは増加し、ドットカウンタは0にリセットされる。サイクル毎に、現状ライン及びドット値がページの余白と比較され、適切な出力ドットアドバンス、ラインアドバンス、及び余白内信号が、これらの余白を基にして与えられる。DMAUはHCUに対する実質的なメモリ必要条件のみを含む。

20

【0410】

印刷可能ページエリアに関連して明示的に定義されることとは別に、各ページ記述は完成し、自己完結している。ページ記述が参照する、ページ記述とは別に記憶されているデータは存在しない。PECは既にセットアップされたディザマトリックス及びタグ構造に依存するが、これらは一般的なページフォーマットの一部分とはみなされない。

30

【0411】

ページ記述は、ページのサイズ及び解像力を記述するページヘッダからなり、実際のページ内容を記述する1つ以上のページバンドが続く。

【0412】

表2はページヘッダのフォーマットを示す。

表2 ページヘッダフォーマット

【0413】

【表6】

40

表2

フィールド	フォーマット	記述
署名	16ビットの整数	ページ ヘッダ フォーマット署名
バージョン	16ビットの整数	ページ ヘッダ フォーマット バージョン番号
構造サイズ	16ビットの整数	ページ ヘッダのサイズ
目標解像力 (dpi)	16ビットの整数	目標ページの解像力。これは現プリンタに対して常に1600
目標ページ幅	16ビットの整数	目標ページの幅、ドットで
目標ページ高さ	16ビットの整数	目標ページの高さ、ドットで
目標左余白	16ビットの整数	目標左余白の幅、ドットで
目標上部余白	16ビットの整数	目標上部余白の高さ、ドットで
タグ フラグ	16ビットの整数	ビット0はこのページに対してタグを生成するかどうかを指定する (0=いいえ、1=はい) ビット1はタグ オリエンテーションを指定する (0=肖像画書式、1=風景画書式) ビット2は固定タグ データがPECにより冗長的に符号化されるべきか、または直接的に使用されるべきかを指定する (0=直接に使用、1=符号化) ビット3は可変タグ データがPECにより冗長的に符号化されるべきか、または直接的に使用されるべきかを指定する (0=直接に使用、1=符号化) 残りビットは予備
固定タグ データ	128ビットの整数	これは生成されたタグ フラグが設定される場合にのみ有効 (タグ フラグのビット0) タグ フラグのビット1がクリアされれば、その時、固定タグ データのより低位の120ビットは事前符号化の固定データを含む。 タグ フラグのビット1が設定されると、その時、より低位の40ビットが、PECにより符号化されることになっている非符号化固定データを含む。 上位の8つのビットは予備
黒色拡大/縮	16ビットの整数	黒色二水準解像力から目標解像力への拡

10

20

30

40

小係数		大／縮小係数（1以上のこと）
黒色ページ幅	16ビットの整数	黒色ページの幅、黒色画素で
黒色ページ高さ	16ビットの整数	黒色ページの高さ、黒色画素で
連続調カラー空間	16ビットの整数	連続調J P E Gチャンネルの数を定義する。一般的にはC M Y対C M Y Kに対して3又は4
連続調1拡大／縮小係数	16ビットの整数	連続調チャンネル1解像力から目標解像力への拡大／縮小係数（1以上のこと）
連続調1ページ幅	16ビットの整数	連続調ページの幅、連続調1画素で
連続調1ページ高さ	16ビットの整数	連続調ページの高さ、連続調1画素で
連続調2拡大／縮小係数	16ビットの整数	連続調チャンネル2解像力から目標解像力への拡大／縮小係数（1以上のこと）
連続調2ページ幅	16ビットの整数	連続調ページの幅、連続調2画素で
連続調2ページ高さ	16ビットの整数	連続調ページの高さ、連続調2画素で
連続調3拡大／縮小係数	16ビットの整数	連続調チャンネル3解像力から目標解像力への拡大／縮小係数（1以上のこと）
連続調3ページ幅	16ビットの整数	連続調ページの幅、連続調3画素で
連続調3ページ高さ	16ビットの整数	連続調ページの高さ、連続調3画素で
連続調4拡大／縮小係数	16ビットの整数	連続調チャンネル4解像力から目標解像力への拡大／縮小係数（1以上のこと）
連続調4ページ幅	16ビットの整数	連続調ページの幅、連続調4画素で
連続調3ページ高さ	16ビットの整数	連続調ページの高さ、連続調4画素で

10

20

30

【0414】

ページヘッダは、プリントエンジンがページヘッドフォーマットを識別することを可能にする署名及びバージョンを含む。署名及び/又はバージョンが無く、又はプリントエンジンと不適合性であると、その時、プリントエンジンはページを拒絶できる。連続調カラー空間は、連続調層がどのくらいあるかを定義し、連続調層がC M Y又はC M Y Kのいずれであるかを定義するために一般的に使用される。ページヘッダは目標ページの解像力及びサイズを定義する。黒色及び連続調層は、必要なら目標ページにクリップで留める。これは、黒色又は連続調の拡大／縮小係数が目標ページ幅又は高さの係数ではない時は、必ず起こる。目標左及び上部余白は、印刷可能ページエリア内に目標ページの位置決めを定義する。

40

【0415】

タグパラメータは、ネットページタグがこのページに対して生産されるべきか否か、及びどんなオリエンテーション（風景画書式又は肖像画書式モード）でタグを生産すべきかを指定する。固定タグデータも提供される。

【0416】

黒色層パラメータは、二水準の黒色層の画素サイズ、及びその目標解像力に対する整数の

50

拡大 / 縮小係数を定義する。連続調層パラメータは、4つの連続調層の各々の画素サイズ、及びそれらの目標解像力に対する整数の拡大 / 縮小係数を定義する。

【0417】

表3はページバンドヘッダのフォーマットを示す。

表3 ページバンドヘッダフォーマット

【0418】

【表7】

表3

フィールド	フォーマット	記述
署名	16ビットの整数	ページバンドヘッダフォーマット署名
バージョン	16ビットの整数	ページバンドヘッダフォーマットバージョン番号
構造サイズ	16ビットの整数	ページバンドヘッダのサイズ
黒色バンド高さ	16ビットの整数	黒色バンドの高さ、黒色画素で
黒色バンドデータサイズ	32ビットの整数	黒色バンドデータのサイズ、バイトで
連続調バンド高さ	16ビットの整数	連続調バンドの高さ、連続調画素で
連続調バンドデータサイズ	32ビットの整数	連続調バンドデータのサイズ、バイトで
ディザマトリックス選択マップバンドデータサイズ	32ビットの整数	ディザマトリックス選択マップバンドデータのサイズ、バイトで。サイズ=0であれば、1つのディザマトリックスのみを使用する
タグバンドデータサイズ	32ビットの整数	非符号化タグデータバンドのサイズ、バイトで。タグデータが提供されないことを表示する0であり得る。

10

20

30

【0419】

黒色(二水準)層パラメータは、黒色バンドの高さ、及びその圧縮バンドデータのサイズを定義する。可変サイズの黒色データがページバンドヘッダに続く。連続調層パラメータは、連続調バンドの高さ、及びその圧縮ページデータのサイズを定義し、連続調カラーデータ及び関連の二水準のディザマトリックス選択マップからなる。可変サイズ連続調データが黒色データに続く。可変サイズの二水準ディザマトリックス選択マップデータが連続調データに続く。

【0420】

タグバンドデータは、タグエンコーダにより要求されたタグデータ半ラインのセットである。タグデータのフォーマットは以下に見られる。タグバンドデータはディザマトリックス選択マップに続く。

40

【0421】

表4はページバンドヘッダに続く可変サイズの圧縮バンドデータのフォーマットを示す。

表4 ページバンドデータフォーマット

【0422】

【表8】

表 4

フィールド	フォーマット	記述
黒色データ	G 4 ファックス バイストリーム	圧縮二水準黒色データ
連続調データ	J P E G バイストリ ーム	圧縮連続調 C M Y K 又は C M Y デ ータ
ディザ マトリック ス選択マップ	G 4 ファックス バイストリーム	圧縮二水準ディザ マトリックス選 択マップ データ
タグ データ マッ プ	バットマップ	タグ データ フォーマット。セクシ ョン 9. 2. 2 参照

10

【 0 4 2 3 】

バンド データの各可変サイズのセグメントは、 8 バイトの境界に配列する。

【 0 4 2 4 】

図 5 0 のタグ エンコーダ (T E) 4 3 0 は、タグ可能のアプリケーションに対する機能性を提供し、一般的には、それは I R インクが印刷ヘッドに存在することを要求する (限られた環境において、 K インク又は他のインクをタグに対して使用してもよいが)。 T E は、印刷されるページに対する固定データを、特定のタグ データ値と共に、赤外線又は黒インクでページ上に以降印刷される誤り修正可能な符号化タグに符号化する。 T E は三角形グリッド (図 5 5 参照) 上にタグを配置し、風景画書式及び肖像画書式オリエンテーションの両方が可能になる。基本タグ構造は 1 6 0 0 d p i で描画される、一方、タグデータは任意の形状のマクロドット (1 6 0 0 d p i で 1 ドットの最小サイズを持つ) として符号化してもよい。

20

【 0 4 2 5 】

T E は入力として下記を取る入れる：

- ・ 風景画 / 肖像画フラグ
- ・ 単一のタグの構造を定義するテンプレート
- ・ 多くの固定データ ビット (ページ用に固定された)
- ・ 冗長的に固定データ ビットを符号化するか否かを、又は既に符号化されているとしてビットを扱うかどうかを定義するフラグ
- ・ 各々が、所定のタグのライン上においてタグ用の可変データ ビットを含む多くの可変データ ビット記録
- ・ 冗長的に可変データを符号化するか否かを、又は既に符号化されいてるとしてビットを扱うかどうかを定義するフラグ

30

タグ エンコーダ (T E) からの出力は、タグ データが印刷されるべきである 1 6 0 0 d p i の二水準の層である。出力は、 1 ビットのワイド F I F O 4 4 7 (図 5 0 及び 5 2 の) を介してなされ、その出力は、順繰りに、図 5 0 の H C U 4 2 9 により入力として使用される。タグは、以降タグ感知デバイスにより読み取り可能な赤外線吸収性のあるインクで印刷されるのが好ましい。黒インクは I R 吸収性があり得るので、黒インクを使用して、オフセット印刷ページの、さもなければ空白になるエリアに、限られた機能性、例えばボタンを符号化する機能性を提供できる。代替として、目に見えない赤外線インクを使用して、位置タグを通常ページの上部に印刷できる。しかしながら、目に見えない I R インクを使用すると、黒インクは赤外線タグを隠すであろうから、ページ上の任意の他の印刷情報が赤外線透明の C M Y インクで印刷されることを確実にするために注意を払わなければならない。単色のスキームは、ぼやけた読み取り環境においてダイナミック レンジを最大限にするために好まれる。

40

【 0 4 2 6 】

ページの同じ面を印刷するのに多数の P E C チップを使用する時、 2 つの P E C チップに

50

より単一のタグを作ることが可能である。これは、タグ エンコーダは複数の部分タグ印刷することができなければならないことを意味する。

【0427】

タグ エンコーダ (TE) は 1600 dpi の二水準データを出力するので、TE の内部作業はハーフトーナ / コンポジタ ユニット (タグ データの利用者) から完全に隠される。

【0428】

タグ エンコーダ (TE) の概念実行により、タグが可変構造並びに固定及び可変データコンポーネントを有することができても、このTEの実行によりある符号化パラメータには範囲制限が課せられる。表5は符号化パラメータ並びに範囲制限を挙げる。しかしながら、これらの制限は、最も起こりそうな符号化シナリオに対して選択された、バッファサイズ及びアドレッシングビット数の直接の結果である。バッファサイズ及び該当のアドレッシングを調整して任意の符号化パラメータを他の実行で可能にすることは簡単な事柄である。

10

表5 符号化パラメータ

【0429】

【表9】

表5

名称	定義	TEにより課せられた最大値
W	ページ幅	12インチ
S	タグサイズ	最小は2mm×2mm
N	タグの各デメンションにおけるドットの数	384ドット (Eに与えられる92ドットが最小)
E	タグデータに対する冗長符号化	5:10でのリード-ソロモンGF(2 ⁴)
D _F	固定データのサイズ (非符号化)	40ビット
R _F	冗長符号化された固定データのサイズ	120ビット
D _V	可変データのサイズ (非符号化)	120ビット
R _V	冗長符号化可変データのサイズ	360ビット
T	ページ幅当たりのタグ	152 (2mm×2mmのパック化タグが可能)
M	マクロドットサイズ	最小は1ドット

20

30

【0430】

各タグにおける固定及び可変データコンポーネントは特に注目される。固定データコンポーネントは、変化しないタグデータの一部である(変化するタグ構造の一部とは異なる)。固定データは、その非符号化フォームでPECチップより読み出されてPEC内で一旦符号化されるか、又は読み出されてそのまま使用されるかのいずれかである(固定データは、それ故、外部的に冗長符号化されるべきである)。可変データビットは各タグに対して可変のデータビットであり、且つ、固定データと同様に、必要ならTE内部で冗長符号化され、又はそのまま使用される。

40

【0431】

データビット(固定及び可変の両方)を冗長符号化ビットにマッピングすることは、用いられた冗長符号化方法に大きく依存する。リード-ソロモン符号化が、最小冗長を使用してバースト誤りを取り扱い、及び誤りを効果的に検出し、修正する能力のため、選ばれた。リード-ソロモン符号化は、ライペンズ、エッチによる「リード-ソロモン誤り修正

50

」、ドップ博士のジャーナル (Dr. Dobb's Journal) Vol. 22、No. 1、1997年1月と；ローラボ、シーによる「誤りコード化 クックブック」、マクグロー-ヒル 1996年と；ウィッカー、エス及びバーガバ、ヴイによる「リード-ソロモンコード及びそれらの応用」、IEEEプレス1994年とに記述されている。

【0432】

タグ エンコーダ (TE) の本実行に際して、リード-ソロモン符号化はガロア フィールド GF (2^4) 上で使用される。記号サイズは4ビットである。各コードワードは60ビットのコードワード長さに対して15の4ビット記号を含む。15の記号のうち、5つは本来のデータ (20ビット) で、10は冗長ビット (40ビット) である。10の冗長記号は、誤っている5つの記号まで修正できることを意味する。

10

【0433】

タグ当たりの本来のデータの合計量は160ビット (40固定、120可変) である。これは冗長符号化され、下記の480ビット (120固定、360可変) の合計量が与えられる：

- ・ 各タグは40ビットまでの本来の固定データを含む。それ故、2つのコードワードが固定データに必要であり、合計符号化データ サイズは120ビットになる。この固定データのみを毎ページ回符号化する必要があることに注目せよ。
- ・ 各タグは120ビットまでの本来の可変データを含む。それ故、6つのコードワードが可変データに必要であり、合計符号化データ サイズは360ビットになる。

20

【0434】

TE は二水準タグ ストリームを二水準タグ FIFO に書き込む。TE は、符号化タグ データを基本タグ構造と合併し、且つ引き続く印刷のために、出力 FIFO 内にドットを正しい順序で配置する責任を有する。符号化タグ データはバッファ スペースを最小にするため、進行中の本来のデータ ビットから生成される。

【0435】

図55に、肖像画書式及び風景画書式印刷に対するタグの配置が見られる。TE がタグ 488をページ上に三角形グリッド アレンジメント、488、489、及び490で配置することが好ましく、風景画オリエンテーション492及び肖像画オリエンテーション491の両方を引き受ける。僅か2つの印刷オリエンテーション (風景画及び肖像画) の制限、及びコラム又はタグの列が重複しないことが組み合わせられた三角形メッシュ488、489、及び490は、タグ配置のプロセスが非常に簡素化されることを意味する。

30

【0436】

図56にはタグを配置する一般的な場合が見られ、それは、それ故、多くのパラメータに依存する。所定のドットのラインに対して、そのライン上のタグ全ては一般のタグ構造の同じ部分に対応する。三角形配置は代替のタグのラインと見なすことができ、そこではタグの一のラインがドット寸法のある量により差し込まれ、他のドットのラインは異なる量により差し込まれる。

ドット タグ間ギャップ493は両方のタグのラインにおいて同じであり、ライン タグ間ギャップ494とは異なる。

40

【0437】

パラメータは、表6及び7においてより形式的に説明する。パラメータの1セットのみ - 肖像画書式印刷用パラメータのセットが必要であることに注目せよ。オリエンテーションが肖像画から風景画に変われば、その時、タグ高さ及びタグ幅パラメータと、一般のドット及びライン パラメータとは簡単に交換できる。表6 タグ配置パラメータ

【0438】

【表10】

表6

パラメータ	記述	制限
タグ高さ	タグのバウンディング ボックス内のドット ライン数	最小1
タグ幅	タグのバウンディング ボックスの単一ライン内のドットの数。タグ自体内のドット数はタグの形状により変わるが、バウンディング ボックス内のドット数は一定である（定義により）。	最小1
ドットのタグ間ギャップ	1つのタグのバウンディング ボックス端から次のタグのバウンディング ボックスの始まりまでのドット数、ドット方向において	最小0
ラインのタグ間ギャップ	1つのタグのバウンディング ボックス端から次のタグのバウンディング ボックスの始まりまでのドットライン数、ライン方向において	最小0
開始位置	ページ（又は複数のP E Cが使用されていれば、ストリップ）上の最初のドット列の開始及び最初のタグ列に対する現状位置記録（表10参照）。非タグ付けのセクションのサイズをタグ間ギャップのサイズを越えて増大するには、非印刷タグを使用せよ。	表10参照
代替タグ ライン位置	代替タグ列の開始に対する現状位置記録。肖像画モード印刷にはドット パラメータを使用し、風景画モード印刷にはライン パラメータを使用する。	表10参照

10

20

表7 現状位置記録

【0439】

【表11】

表7

名称	記述
タグ状態ドット	0 = タグ間ギャップに 1 = タグに
タグ状態ライン	0 = タグ間ギャップに 1 = タグに
ローカル オフセット ドット	タグ間ギャップ又はタグ内の現状ドット位置、最小=0
ローカル オフセット ライン	タグ間ギャップ又はタグ内の現状ライン位置、最小=0

30

40

T E は幾つかの特定のデータ構造を利用する：

- ・ T E オリエンテーション フラグ、それはページが肖像画書式又は風景画書式タグ配置規則を使用して印刷されつつあるかどうかを判別する。
- ・ タグ フォーマット構造、固定タグ構造、可変データ ビット、及び固定データ ビットの視点から総称タグの構成を詳述するテンプレート。それは多数のタグ ライン構造で構成され、タグ内の各 1 6 0 0 d p i に対して1つある。2つのタグ フォーマット構造 - 1つは肖像画書式印刷用、及びもう1つは風景画書式印刷用、がある。
- ・ 固定タグ データ バッファ、ページ（又は多数のP E Cチップが使用される時、ページの一部）上のタグ全てに対する冗長符号化固定データ コンポーネントを含む。
- ・ 「タグ印刷」のフラグ、特定のタグが印刷されるべきか否かを指定する。タグ フォ

50

ーマット構造を無視し、且つタグを単に出力しないかどうかをエンコーダに命令する。

- ・ 半ラインのタグ データ バッファ、非符号化データ及びタグに対する「タグ印刷」のフラグを所定のタグのラインの半分を含む（ラインはこの P E C チップにより印刷されたストリップの幅である）。もしタグの一部のみがこの P E C チップにより印刷される場合、その時、全体タグのデータは存在しなければならない。

- ・ 可変タグ データ バッファ、単一のタグに対する冗長符号化可変データを含む。

【 0 4 4 0 】

データ構造は以下に更に詳細に説明する。種々の構造のサイズは表 5 に挙げたタグ符号化パラメータに基づくことに注目せよ。符号化パラメータの種々のセットに対して、構造のサイズ及び対応するアドレス ビットの数は適切に変化すべきである。

【 0 4 4 1 】

T E は風景画書式及び肖像画書式印刷の両方をサポートする。モードは P E C に接続した印刷ヘッドの長さとは完全に無関係である。給紙が正しければ、12 インチ印刷ヘッドはレター及び A 4 ページを風景画及び肖像画の両方の書式で印刷でき、多数の P E C チップを結合して任意の大きさのページを生産することができる。その結果、T E はタグのオリエンテーションを決定するフラグを含む。

【 0 4 4 2 】

T E オリエンテーションは、それ故、表 8 に示したような値を持つ 1 ビットのフラグである。

表 8 T E オリエンテーション レジスタ値

【 0 4 4 3 】

【表 1 2】

表 8

値	記述
0	風景画書式
1	肖像画書式

【 0 4 4 4 】

各 10 ビットのエントリは、表 9 に説明したように個別に解釈され、状態情報には信頼を置いていない。特に（2つの P E C を跨って広がった）部分タグの片側を描画中、エントリへのランダム アクセスが可能である故に、これは重要である。

表 9 タグ ライン構造における 10 ビットのエントリの解釈

【 0 4 4 5 】

【表 1 3】

10

20

30

表9

ビット9	解釈
0	このドットは基本タグ構造の一部である。ビット8はドット出力値を含む。残りの8ビットは予備であり、0に設定すべきである。
1	このドットはタグのデータ部分から抜粋される。 より下位の9ビットを使用して、使用する実際のデータビットを決定する。もしアドレスの上位2ビットを設定すれば、残りの7ビットを使用して、符号化固定データの120ビットをページに対してアドレス指名する。もしアドレスの上位2ビットを両方とも設定しなければ、完全な9ビットのアドレスを使用して、符号化可変データの360ビットをタグに対してアドレス指名する。

10

【0446】

タグフォーマット構造(TFS)はラインベースなので、我々にかかる2つの構造を外部DRAMに記憶させている - 1つは肖像画オリエンテーション用、もう1つは風景画オリエンテーション用。TEオリエンテーションフラグは、2つのうちのどれを使用するかを決定する。2つのタグフォーマット構造は外部プロセスにより供給され、外部DRAM内で記憶され、それ故、任意に異なることがありあり得る。しかしながら、実際は、それらは90°回転した同じタグである。単一のTFSにより要求されるメモリの合計は3840×タグ高さビットである。所要の最大メモリ量は、高さのタグ384に対するものであり、180Kバイトになる。それ故、最大合計360Kバイトが2つのオリエンテーションに必要である。

20

【0447】

図55に示したように、所定のドットのラインに対して、そのライン上のタグ全ては同じタグライン構造に対応する。その結果、所定の出力ドットのラインに対して、単一のタグライン構造が必要であり、TFS全体ではない。ダブルバッファリングにより、次のタグライン構造をDRAM内のTFSから取り出すことができる、一方現存するタグライン構造を使用して、現状タグラインを描画する。タグ構造データのラインを読み出すと、その結果、オリエンテーションに関わらず同じDRAMバンド幅を消費する。TFS全体はPESチップ上に記憶でき、その場合には回転はオンザフライで行われる。TFSに対するメモリ必要条件は、それ故、チップ上のダブルバッファリングされたタグライン構造で(合計では3840ビット×2=7680ビット、又は960バイト)、肖像画書式TFS及び風景画書式TFSに対して外部DRAM内の360Kバイトまでである。バンド幅については、肖像画書式TFS及び風景画書式TFSの書き込みは一度行う必要があるのみであり、それは問題点ではない。しかしながら、印刷中適切なTFSを読み出すことが問題点である。最悪ケースである隣接タグを仮定すると、出力ライン毎にタグライン構造を読み出す必要がある。各タグライン構造は480バイトである。毎秒30,000ラインの最高印刷速度に対して、TFSアクセスは13.8MB/秒に達する。

30

40

【0448】

固定タグデータバッファは120ビットのデータバッファであり、7ビットによりアドレス指名される。バッファはページに対するタグデータの符号化固定コンポーネントを保持する。固定タグデータバッファは、120ビットの本来の固定データ入力から直接、又は本来の固定データのより下位40ビットがリード-ソロモン符号化された後のいずれかに、ページ当たり一度書き込まれる。

【0449】

「タグ印刷」のフラグは、特別のタグを印刷すべきか否かを指定する。単一のビットのみ

50

で、このフラグは合計 2 ビットに対してダブル バッファリングされる。ダブル バッファリングにより、現状のタグを描画しながら、次のタグに対する「タグ印刷」のフラグを決定することができる。「タグ印刷」は、それ故、表 10 に示した値を持つ 1 ビットのフラグである。

表 10 「タグ印刷」レジスタ値

【 0 4 5 0 】

【表 1 4】

表 10

値	記述
0	タグを印刷するな。T F S 並びに固定されたタグ及び可変データ値を無視せよ。タグ バウンディング ボックス内の各ドットに対して 0 を出力せよ。
1	種々のタグ構造により指定されたようにタグを印刷せよ。

10

【 0 4 5 1 】

半ライン タグ データ バッファは、ライン上のタグの半分までに対する非符号化可変タグ データを含む。各ラインは最大 1 5 2 タグを含むことができるので (1 2 インチの長さ
にわたって 2 mm × 2 mm の密にパックされたタグ サイズ)、各半ライン タグ バッ
ファはせいぜい 7 6 タグを含むにすぎない。図 5 5 に示したように、1 2 8 ビットは、各
タグ 4 9 5、4 9 6、4 9 7 などに配分される：1 2 0 ビットの非符号化データ 5 0 1、
1 ビットの「タグ印刷」のフラグ 4 9 8、1 ビットの半ライン最終タグのフラグ 4 9 9、
及び 6 つの予備ビット (0 に設定された) 5 0 0 である。単一のバッファのサイズは、そ
れ故、9 7 2 8 ビット (1 2 1 6 バイト) である。非符号化データに対するマジック値 (
例えば、0) を有する代わりに、1 ビットの「タグ印刷」のフラグ 4 9 8 へ配分すること
は、非符号化データが 1 2 0 ビットの完全に非制限のデータであることを意味する。

20

【 0 4 5 2 】

タグ データのライン全体をダブル バッファリングするよりもむしろ、我々はタグ デー
タの半ラインをトリプル バッファリングする。これにより、(ダブル バッファリングさ
れた完全なタグ ラインに比較して) 1 2 1 6 バイトが節約されるが、完全なタグ ライン
を読み出すライン時間全体を有する代わりに、半ライン タグ データをドット ラインの
半分を読み出さなければならというタイミング制限が生じる。2 つだけではなく、3 つの
半ライン バッファを有することが重要であることに注目せよ。2 つだけの半ライン バッ
ファでは、所定のタグのセットが多数のドット ラインにわたって延びときに、同じタグ
データを再度読み出す必要がある。トリプル バッファリングにより、D R A M から再読
み出す必要なく、同じ 2 つの半ライン タグ バッファを多数回 (各タグのラインに対して
1 回) 使用できる。第 3 の半ライン タグ バッファを使用して、現状のタグのセットをプ
ロセッシング中、次のタグ ラインのデータの前半を取り込み、且つ、次のタグ ラインの前
半をプロセッシング中、次のタグ ラインのデータの後半を取り込む。全体の印刷プロセス
中、所定のタグ ラインのデータが一度だけ読み出されることに注目せよ。その結果、1
ビットの「最初にプロセス」のフラグは、この半ライン上のタグが以前にプロセスされて
いるかを指定する各半ライン バッファに関連する。所定の半ラインをプロセスする最
初の時に、次の半ライン バッファを D R A M から取り込む。

30

40

【 0 4 5 3 】

タグ データは D R A M 内に半ラインに関して配置される。所定のライン上に N 個のタグ
があれば、D R A M 内に記憶されている各半ラインは N / 2 のタグに対するデータを含む
。N が奇数であれば、半ラインのうち的一方は他の半ラインより 1 つ少ないタグを含む。
「半ライン最後タグ」のフラグは一方の半ラインに対してタグ N / 2 内に、他の半ライン
に対してタグ (N / 2 - 1) 内に設定される。それにも拘わらず、一方のタグ半ラインか

50

ら次のタグ半ラインへのズレは両方の場合とも同じである。肖像画書式及び風景画書式ページは、タグの合計数に関してお互いにバランスを取る。最悪のケースとして、隣接2mm×2mmタグを仮定すると、半ライン当たり76タグあり、長さ8.5インチのページに対して107のタグがライン寸法中にある。DRAM中の全データのサイズは、それ故、 $1216 \times 2 \times 108 = 255 \text{ K}$ バイトである。毎秒1ページの印刷速度に対して、DRAMへのバンド幅は、それ故、 255 KB /秒である。毎秒30,000ラインの最高印刷速度に対して、TFSアクセスは約561Kバイト/秒に達する。

【0454】

可変タグ データ バッファは単一のタグに対して360ビットの符号化可変データを保持する。TEは合計720ビットに対して可変タグ データ バッファをダブル バッファリングする。ダブル バッファリングにより、次のタグに対する生の120ビットの可変データを(必要なら)冗長符号化し、一の可変タグ データ バッファに記憶することができ、一方他の可変タグ データ バッファを使用しながら状タグに対するドットを生成することができる。もし可変タグ データがPECにより符号化されなければ、360の可変データ ビットのうち最初の120ビットのみが有効であること、及び120ビットの可変タグ データが適切な冗長符号化を既に適用されていることを保証するのは外部ページ プロバイダの責任であることに注目せよ。

10

【0455】

図58に示した可変タグ データ バッファは、現状タグの可変データである。現状のタグに対するドットが生産されつつある時、図59に示したように、次のタグに対する可変データは第2の可変タグ データ バッファに符号化されつつある。

20

【0456】

タグ フォーマット構造の全体、又は全タグに対する可変タグ データを記憶するよりもむしろ、データはジャスト イン タイム様式で外部DRAMから取り込まれる。適切なトレードオフがバッファ サイズと転送バンド幅との間でおこなわれる。プロセッシングがドット及びライン方向の両方で起きてから、データがジャスト イン タイム作業で利用できることが保証される。

- ・ 一方のタグに対するドットがドット方向で生成されつつある時に、次のタグに対する可変データ コンポーネントが第2の可変データ バッファに冗長符号化されつつあり、次タグの「タグ印刷」のフラグが決定されつつある。これらのタスクの両方共半ライン タグ データ バッファからの読み出しを伴い、外部DRAMへのアクセスは伴わない。

30

- ・ 半ライン タグ データ バッファが使用される最初の時に、非符号化タグ データの次の半ラインがDRAMから取り出される。タグ データの半ラインが再度使用される時はDRAMからなにも読み出されない。3つの半ライン タグ バッファがあるので、2つのバッファを単一のタグのラインに対して多数回使用できる、一方、タグの次の半ラインに対するデータが準備されている。これにより、各タグの非符号化タグ データが一度のみDRAMから読み出すことができることに注目せよ。

- ・ タグの一方のラインが生産されつつある時、タグ フォーマット構造の次のラインが外部DRAMから読み出される。これは、現状の出力ラインが実際にタグの一部である場合のみに必要である。タグの最終ラインの場合、タグの最初のラインが再度読み出される。タグ間ラインをプロセッシング中はなにも読み出されない。

40

【0457】

表11はTEに対するメモリ必要条件をオンチップ及びオフチップ(外部DRAM)の両方に関して要約する。

表11 TEメモリ必要条件

【0458】

【表15】

表 1 1

名称	オンチップ合計必要 条件	オフチップ最悪ケース (外部DRAM)
TEオリエンテーション	1ビット	—
タグフォーマット構造	960バイト	360Kバイト(合計)
固定タグデータバッファ	120ビット	—
「タグ印刷」のフラグ	2ビット	—
半ラインタグデータバッファ	3648バイト	255Kバイト(ページ毎に)
可変タグデータバッファ	720ビット	—
合計	5018バイト	

10

【0459】

最高のレベルにおいて、TE内の状態マシンは一度に1ラインずつページの出力ラインを進み、開始位置はタグ間ギャップ又はタグ内である(単一のラインを印刷している多数のPECのため、1つのPECはタグの一部を印刷しているにすぎないかもしれない)。もし現状位置がタグ間ギャップ内であれば、0の出力が生成される。もし現状位置がタグ内であれば、タグフォーマット構造を使用して、出力ドットの値を決定し、必要ならば固定又は可変データバッファからの適切な符号化データビットを使用する。次いで、TEはドットのラインに沿って進み、タグ配置パラメータに従いタグ及びタグ間ギャップを通過する。一旦出力ドットのライン全体が生産されたならば、TEは次のドットのラインに進み、ライン方向に対するタグ配置規則に従いタグ及びタグ間ギャップを通過する。PEC内の他のドット生成プロセスに遅れずについて行くために、出力ドットはサイクル毎に生成しなければならない。擬似コードにおいて、プロセスは次の通りである。DRAMにアクセスするためのロジックは示されないことに注意せよ。

20

【0460】

イフ(TEオリエンテーション = 肖像画書式)

最大タグコンポーネントライン[0] = ラインタグ間ギャップ

最大タグコンポーネントライン[1] = タグ高さ

最大タグコンポーネントドット[0] = ドットタグ間ギャップ

最大タグコンポーネントドット[1] = タグ幅

開始ドットオフセット[0] = 開始位置。ローカルオフセットドット

開始ドット状態[0] = 開始位置。タグ状態ドット

開始ドットオフセット[1] = 代替タグライン位置。ローカルオフセットドット

開始ドット状態[1] = 代替タグライン位置。タグ状態ドット

現状位置タグ状態ライン = 開始位置。タグ状態ライン

現状位置ローカルオフセットライン = 開始位置。ローカルオフセットライン

エルス

最大タグコンポーネントライン[0] = ドットタグ間ギャップ

最大タグコンポーネントライン[1] = タグ幅

最大タグコンポーネントドット[0] = ラインタグ間ギャップ

最大タグコンポーネントライン[1] = タグ高さ

開始ドットオフセット[0] = 開始位置。ローカルオフセットライン

開始ドット状態[0] = 開始位置。タグ状態ライン

開始ドットオフセット[1] = 代替タグライン位置。ローカルオフセットライン

開始ドット状態[1] = 代替タグライン位置。タグ状態ライン

現状位置タグ状態ライン = 開始位置。タグ状態ドット

30

40

50

現状位置ローカルオフセットライン = 開始位置。ローカルオフセットドット エンドイフ
 R S エンコーダの出力タグ用意のフラグが設定されるまでストールせよ
 R S エンコーダからドットジェネレータへ「タグ印刷」のフラグを転送せよ
 R S エンコーダからドットジェネレータへ可変タグ データ バッファを転送せよ
 アドバンス タグ信号を次のタグの符号化を始める R S エンコーダに送信せよ
 タグラインタイプ = 0
 ラインカウント = 0
 ホワイル (ラインカウント < 最大ライン)
 ドウ
 現状位置タグ状態ドット = ドット状態を開始 [ライン タイプをタグ付けせよ] 10
 現状位置ローカルオフセットドット = ドットオフセットを開始 [ライン タイプをタグ付
 けせよ]
 ドットカウント = 0
 ホワイル (ドットカウント < 最大ドット)
 ドウ
 イフ (現状位置タグ状態ドット = タグ間ギャップに)
 0 を F I F O に書き込む
 エルス
 イフ (現状位置タグ状態ドット = タグに)
 (タグライン構造を復号化せよ [現状位置ローカルオフセットドット]) を F I F O に書 20
 き込む
 エルス
 0 を F I F O に書き込む
 エンドイフ
 現状位置ローカルオフセットドットを進める
 イフ (現状位置ローカルオフセットドット > 最大タグコンポーネントドット [現状位置タ
 グ状態ドット])
 現状位置ローカルオフセットドット = 0
 現状位置タグ状態ドット = ((~ 現状位置タグ状態ドット) 又は (ドットタグ間ギャップ
 = 0)) 30
 イフ (現状位置タグ状態ドット = タグに)
 R S エンコーダからドットジェネレータへ「タグ印刷」のフラグを転送せよ
 R S エンコーダからドットジェネレータへ可変タグ データ バッファを転送せよ
 アドバンス タグ信号を次のタグの符号化を始める R S エンコーダに送信せよ
 エンドイフ
 エンドイフ
 エンドイフ
 ドットカウントを進めよ
 エンドドウ
 ラインカウントを進めよ 40
 現状位置ローカルオフセットラインを進めよ
 イフ (現状位置ローカルオフセットライン > 最大タグコンポーネントライン [現状位置タ
 グ状態ライン])
 現状位置ローカル状態ライン = ((~ 現状位置タグ状態ライン) 又は (ラインタグ間ギャ
 ップ = 0))
 現状位置ローカルオフセットライン = 0
 イフ (現状位置タグ状態ライン = タグに)
 タグラインタイプ = ~ タグラインタイプ
 エンドイフ
 エンドイフ 50

エンドドゥ

タグ内の位置に基づいた単一のビットを出力することは、適切なタグ ライン構造、現状タグに対する符号化固定及び可変タグ データ、及び現状タグに対する「タグ印刷」のフラグへのアクセスを持つことに依存する。これらが適切に取り込まれたと仮定し、且つ表 5 の符号化パラメータを仮定すると、単一タグドットの生成はブロック図のフォームで図 5 8 に見ることができる。

【0461】

図 5 9 においてエンコーダのブロック図が見られる。TE は、シンボル アト ア タイムの GF (2^4) リード - ソロモン エンコーダ 5 9 0 を含む。記号サイズは 4 ビットである。各コードワードは、60 ビットのコードワード長さに対して 15 の 4 ビット記号を含む。15 の記号のうち、5 つは本来のデータ (20 ビット) で、10 個は冗長ビット (40 ビット) である。各タグは 120 ビットの可変の本来のデータを含むので、合計 360 ビットの符号化データ サイズには 6 つのコードワードが必要である。固定タグ データも、同じリード - ソロモン エンコーダを使用して符号化される。固定タグ データも、同じリード - ソロモン エンコーダを使用して符号化される。固定データは、所定のページ (又は多数の PEC が使用されていれば、ページのストリップ) 用のタグ全てにわたり一定であり、それで、印刷 (又は印刷のセット) 前に一度セットアップする必要があるのみである。非符号化固定データは長さが 40 ビットである。これらの 40 ビットは符号化されて 120 ビットを生産する。固定データを符号化するため、CPU は固定データを非符号化タグ データ バッファの最初の 40 ビットに取り込み、次いで、状態マシンを始動させて 2 つのコードワードを符号化する。その結果生じる可変タグ データ内の 120 ビットは、次いで、固定タグ データ バッファに転送され、そこでそれらのビットは少なくとも 1 ページの印刷のために、多くの場合は多数ページの印刷のために、留まる。もし固定データが PEC により符号化されないことになっていれば、その時、120 ビットの固定データ全ては直接固定タグ データ バッファにコピーされる。状態マシン 5 9 1 は、タグ データを符号化するアドレッシング及び制御信号を生産する責任がある。表 1 3 は状態マシン 5 9 1 をプログラミングするのに使用されるレジスタを示す。

【0462】

「タグ準備済み」のフラグは、始動時、5 9 2 において、また、5 9 3 においてアドバンス タグ信号が受信された後はいつでも、状態マシン 5 9 1 によりクリアされる。一旦コードワードのセット全体が適切にリード - ソロモン符号化されると、フラグは設定される。「タグ準備済み」のフラグにより、符号化データの外部利用者が適切にストールすることができる。

【0463】

5 : 10 記号の符号化を生産するために、状態マシン 5 9 1 は、適切な半ライン タグ バッファ 5 9 4 から記号幅リード - ソロモン デコーダ 5 9 0 への 4 ビットのデータを、5 9 5 においてゲートする。クロックデータの信号は最初の 5 つのクロックに対して供給され、そのインバースが次の 10 個のクロックに供給される。これがコードワード数の回数繰り返される。90 個のクロックが、それ故、タグ データ全体 (6 コードワード \times 15 クロック) を符号化するのに必要である。更なる 2 つのクロックが、残りの 8 ビットを飛び越えるのに必要で、かくして合計を 92 サイクルにする。状態マシン 5 9 1 は、「タグ準備済み」のフラグを 5 9 2 で設定し、5 9 3 上のアドバンス タグ信号が TE の高レベルプロセスから与えられるまで (この信号が与えられるのに要する時間はタグの幅による。92 のタグ サイズは最小遅延を与える) ストールする。これら最後の 2 つのクロックの最初のクロックについて、書き込み可能信号が 5 9 6 で生成され、その結果、「タグ印刷」のフラグ 5 9 7 が、非符号化タグ データ バッファ 5 9 4 (タグ データのビット 121) から読み出された 4 つのうちの最初ビットに設定される。同じクロック中、4 つのうちの第 2 のビットは状態マシンに渡される。この第 2 のビットは、半ライン内最終タグと呼ばれ、プロセスされたばかりのタグが、実際に半ライン バッファでプロセスされる予定の最終タグであるかどうかを決別する。半ライン タグ バッファ 5 9 4 に対

して状態マシンにより生成されたアドレスは、14ビットである。高位の2ビットが、3つのデータバッファのうちのいずれかがアドレス指名されるかを選択する。次の9ビットは、バッファからどの32ビットの量を読み出すべきかを判別し、より下位の3ビットを使用して4ビットの8つのセットのどれが選択されるべきかを決定する。14のアドレスビットのうち、下位の12ビットのアドレスは0で始動し、32回進むまで、各サイクルを進める。その時、カウンタは、593上のアドバスタグ信号が高レベルの符号化プロセスから来るまでストールする。しかしながら、もし「半ライン内最終タグ」のフラグが設定されれば（最も新しくプロセスされたタグからビット122として読み出されれば）、その時、下位の12ビットのアドレスは0にクリアされ、タグ半ラインバッファの2ビットのインデックスが更新され、DRAMからの次のタグデータの半ラインに対する取り込みプロセスが潜在的に開始される。

10

【0464】

状態マシン591はこの完全なタグラインに対してプロセスされた半ラインの数に対する10ビットのタグラインカウンタを保持する。タグラインカウンタは、始動時クリアされ、次いで、状態マシンが「半ライン内最終タグ」のフラグが設定されているタグの符号化を終了する毎に、増加する。タグラインカウンタが増加すると、タグ半ラインインデックスに対する新しい値を決定し、並びにもしかするとタグラインカウンタ自体の再設定に、10ビットを使用する。ラインの前半（タグラインカウンタの最下位ビット=0）のとき、次の半ラインバッファは常に同じタグラインの後半の半ラインであろう。これは2ビットのインデックスを更新することを単に意味する。ラインの後半（タグラインカウンタの最下位ビット=1）のとき、次の半ラインは我々がこのタグラインのプロセッシングを終了しているか否かによる。もし我々がこのタグラインをプロセスしていなければ（TEOオリエンテーション内の値により、タグラインカウンタの最上位9ビットが、タグ高さ又はタグ幅のいずれにも合致しない）、次の半ラインは前の半ラインと同じである。もし我々がこのタグラインを終了していれば、次の半ラインは次のラインから来るので、それ故、我々は次のタグラインの半ラインバッファを使用する。我々は新しいタグラインを始めつつあるので、カウンタは0にまたクリアされる。表12は古い及び新しいカウンタと、半ラインバッファインデックスとの関係を示す。

20

表12 半ライン内最終タグが設定される時に何をすべきか

【0465】

30

【表16】

表 1 2 及び脚注

タグライン カウンタの下 位ビット	タグライン カウンタの上 位ビット=タグ 高さ	現状インデッ クス値	次インデック ス値	カウンタをク リア？
0 ¹⁰	x ¹¹	0	1	いいえ
0	X	1	2	いいえ
0	X	2	0	いいえ
1 ¹²	0	0	2	いいえ
1	0	1	0	いいえ
1	0	2	1	いいえ
1	1	0	1	はい
1	1	1	2	はい
1	1	2	0	はい

- 1 0 ラインの前半を意味する
 1 1 状態に構うな
 1 2 ラインの後半を意味する

10

20

【 0 4 6 6 】

インデックス値が変わる時はいつでも、古いインデックスを保持し、新しいインデックスに関連した半ライン バッファに対する「最初にプロセス」のフラグをチェックする。「最初にプロセス」のフラグがクリアされていれば、それ以上なにもしない。しかしながら、「最初にプロセス」のフラグが設定されていれば、それはクリアされ、DRAMからの次の半ラインに対する次のデータのセットを、古いインデックスにより指定された半ラインに読み出すプロセスが始動する。古いインデックスに関連した半ラインに対する「最初にプロセス」のフラグが、次いで設定される。DRAMから読み出される32ビットのワード数は、表13に記載したように、半ラインサイズのレジスタにより指定される。タグ半ラインを読み出す現状アドレスは、その時、そのアドレスが読み出される次の半ラインを指すように、半ラインサイズにより増加する。このスキームのため、単一の半ラインがページの終わりに早手回しに読み出される。データはページに送信されないため、これは問題ではない。

30

表 1 3 タグの可変データを操作するレジスタ

【 0 4 6 7 】

【 表 1 7 】

表13

パラメータ	記述	代表値
データ記号	出力コードワード内のデータ記号の数	5
冗長記号	出力コードワード内の冗長記号数	10
コードワード数	符号化するコードワード数	6
半ラインサイズ	DRAMから読み込まれる可変タグデータの半ライン内の32ビットの量の数	304
符号化選択	このビットが設定されていれば、その時、データはリード-ソロモン符号化される。クリアされていれば、データは単にコピーされる。	1

10

【0468】

結論

本発明は、好適な実施例及び多数の特定の代替実施例に関して説明した。しかしながら、当業者により、これらの特に説明した実施例とは異なる多くの他の実施例も、本発明の精神及び範囲に入ることは理解されるであろう。従って、本発明は、関連出願の説明により適切に援用されたドキュメントを含む、本明細書で記載した特定の実施例に限定する意図はないことは理解されるであろう。本発明の範囲は添付の特許請求の範囲によってのみ限定される。

20

【図面の簡単な説明】

【図1】サンプル印刷ネットページとそのオンライン ページ記述との関係の概略図である。

【図2】ネットページ ペンと、ネットページ プリンタと、ネットページ ページ サーバと、ネットページ アプリケーション サーバとの間の対話の概略図である。

【図3】ネットワークを介して相互接続されたネットページ及びプリンタの情報収集を示す。

30

【図4】印刷ネットページの高レベル構造及びそのオンライン ページ記述の概略図である。

【図5】ネットページタグの構造を示す平面図である。

【図6】図5に示したタグのセット間の関係及びネットページ ペンの形態をしたネットページ感知デバイスの視野を示す平面図である。

【図7】タグ画像プロセッシング及び復号化アルゴリズムのフローチャートである。

【図8】ネットページ ペン及びその関連のタグを感知する視野錐体の透視図である。

【図9】図8に示したネットページ ペンの透視分解組立図である；

【図10】図8及び9に示したネットページ用のペン コントローラの概略ブロック図である。

40

【図11】壁装着のネットページ プリンタの透視図である。

【図12】図11のネットページ ペンプリンタの長さ方向に沿った断面である。

【図12a】複式プリント エンジン及びグルー ホイール組立品の断面を示す図12の拡大部分である。

【図13】図11及び12のネットページ プリンタのインク カートリッジ、インク、空気及びグルー通路、及びプリント エンジンの詳細図である。

【図14】図11及12に示したネットページ プリンタ用のプリンタ コントローラの概略ブロック図である。

【図15】図14に示した複式プリント エンジン コントローラ及びプリンタ コントロ

50

ーラに関連した MenjetTM 印刷ヘッドの概略ブロック図である。

【図16】図14及15に示したプリントエンジンコントローラの概略ブロック図である。

【図17】例えば、図10から12のネットページプリンタで使用された単一の MenjetTM 印刷要素の透視図である。

【図18】MenjetTM 印刷要素のアレイの小部分の透視図である。

【図19】図13に示した MenjetTM 印刷要素の動作サイクルを示す一連の透視図である。

【図20】ページ幅 MenjetTM プリントヘッドの短セグメントの透視図である。

【図21】利用者クラス線図の概略図である。

【図22】プリンタクラス線図の概略図である。

【図23】ペンクラス線図の概略図である。

【図24】アプリケーションクラス線図の概略図である。

【図25】ドキュメント及びページ記述クラス線図の概略図である。

【図26】ドキュメント及びページ所有権クラス線図の概略図である。

【図27】端末要素特殊化クラス線図の概略図である。

【図28】静的要素特殊化クラス線図の概略図である。

【図29】ハイパーリンク要素クラス線図の概略図である。

【図30】ハイパーリンク要素特殊化クラス線図の概略図である。

【図31】ハイパーリンクグループクラス線図の概略図である。

【図32】フォームクラス線図の概略図である。

【図33】デジタルインククラス線図の概略図である。

【図34】フィールド要素特殊化クラス線図の概略図である。

【図35】チェックボックスフィールドクラス線図の概略図である。

【図36】テキストフィールドクラス線図の概略図である。

【図37】署名フィールドクラス線図の概略図である。

【図38】入力プロセッシングアルゴリズムのフローチャートである。

【図38a】図38のフローチャートの1ステップの詳細なフローチャートである。

【図39】ページサーバコマンド要素クラス線図の概略図である。

【図40】リソース記述クラス線図の概略図である。

【図41】お気に入りリストクラス線図の概略図である。

【図42】履歴リストクラス線図の概略図である。

【図43】定期購読配信プロトコルの概略図である。

【図44】ハイパーリンク要求クラス線図の概略図である。

【図45】ハイパーリンク起動プロトコルの概略図である。

【図46】フォーム配信プロトコルの概略図である。

【図47】手数料決済プロトコルの概略図である。

【図48】データの流れ及びプリントエンジンコントローラにより遂行された機能を示す線図である。

【図49】全体のプリンタシステムアーキテクチャの文脈におけるプリントエンジンコントローラを示す。

【図50】プリントエンジンコントローラアーキテクチャを示す。

【図51】図50のハーフトナー/コンポジットユニット(HCU)への外部インタフェースを示す。

【図52】図51のHCPへの内部回路を示す線図である。

【図53】図52のドット合併ユニット内のプロセスを示すブロック図を示す。

【図54】図52のドット再組織化ユニット内のプロセスを示す線図を示す。

【図55】肖像画書式及び風景画書式モードにおけるタグの配置を示す。

【図56】タグ配置を定義するために使用されたパラメータを表す。

【図57】半ラインタグデータバッファ構造を表す。

10

20

30

40

50

【図58】単一のタグ ドットを生成する回路を示す。

【図59】リード - サーモンをベースにした回路符号化タグ データを示す。

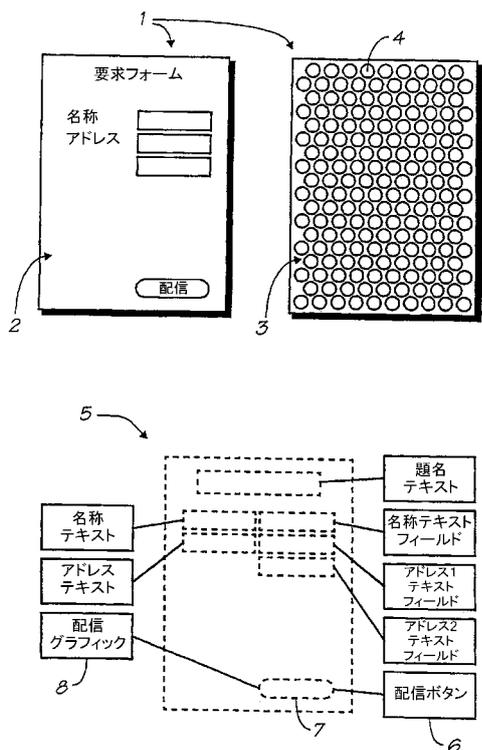
【符号の説明】

1 : 印刷ページ、2 : 印刷されたグラフィック データ、3 : 印刷されたコード化データ、4 : 印刷されたタグ、5 : ページ記述、7 : ゾーン、9 : 短距離無線リンク、15 : 検出リング、16 : オリエンテーション軸心、17 : 固定目標構造、18 : 可変データエリア、18 a から 18 d : 同心円記号リング、101 : ペン、102 : ハウジング、103 : 壁、104 : 内部スペース、105 : ペン上部、106 : 端、107 : カバー、108 : 端、109 : 主部、110 : 突出部、111 : スロット、112 : アンテナ、113 : ネジ山、113 A : アパーチャ、114 : 金属末端部、115 : ネジ山、119 : ペン先端、120 : 針、121 : 針先端、122 : 開口端、123 及び 124 : スライダ ブロック、125 : カムバレル、126 : カム、127 及び 128 : カムフォロア、129 : 屈曲 PCB、130 : 電子シャーシ、131 : 赤外線 LED、132 : 画像センサ、133 : 無線周波チップ、134 : コントローラ チップ、135 : 光学ブロック、136 : 電力供給線、137 : 電池端子、138 : ターミナル、139 : バッテリ、140 : 誘導充電コイル、141 及び 142 : ラバー グリップ パッド、143 : LED、181 : 壁、190 : 加速度計、193 : 非最小視野、300 : 印刷要素、301 : シリコン ウエーハ、302 : ノズル、303 : ノズル リム、304 : ノズル チャンバー、305 : 流体シール、306 : インク チャンネル リム、307 : レバー アーム、308 : 能動アクチュエータ ビーム対、309 : 受動アクチュエータ ビーム対、310 : 能動アクチュエータ アンカー、311 : 受動アクチュエータアンカー、312 : インク インレット、315 : 断面部、316 : インク メニスカス、317 : インク ドロップレット、320 : 電熱ベンド アクチュエータ、321 : インク、330 : ノズル ガード ウエーハ、331 : ノズル ガード穴、332 : 空気インレット、350 : MEMJET 印刷TM ヘッド、601 : ネットページ 壁プリンタ、602 及び 603 : MemjetTM プリントエンジン、604 : 紙、605 : 製本アセンブリ、613 から 616 : 利用者インタフェース LED、617 : ヘルプ ボタン、626 : ローラ アセンブリ、627 : インク カートリッジ、638 : 空気ポンプ、639 : ホース、656 : プリンタ コントローラ。

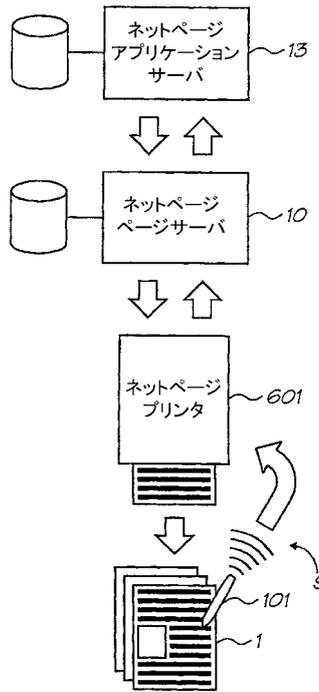
10

20

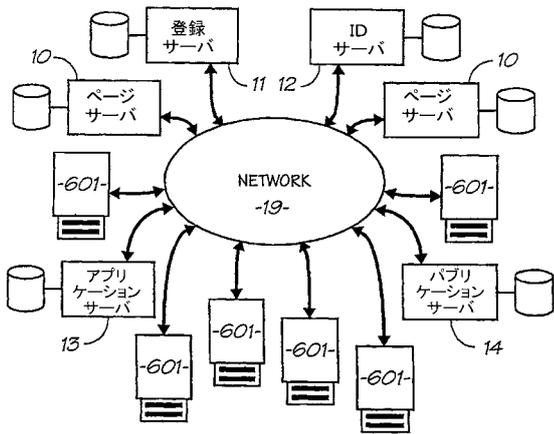
【 図 1 】



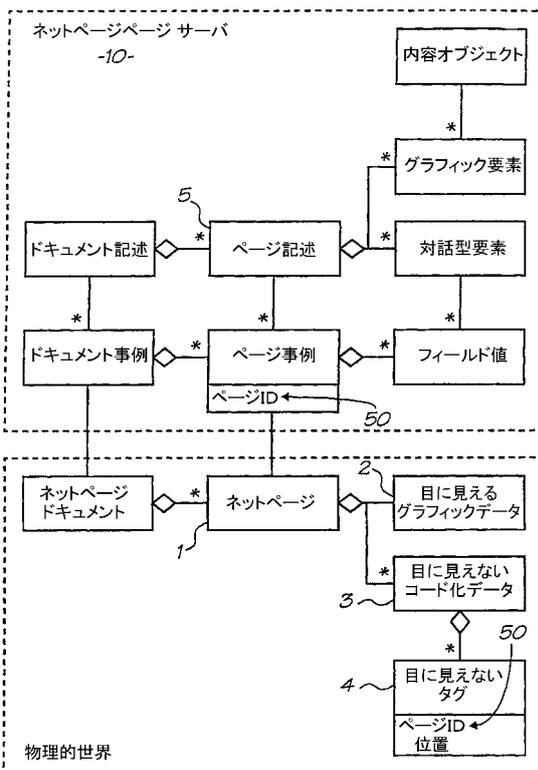
【 図 2 】



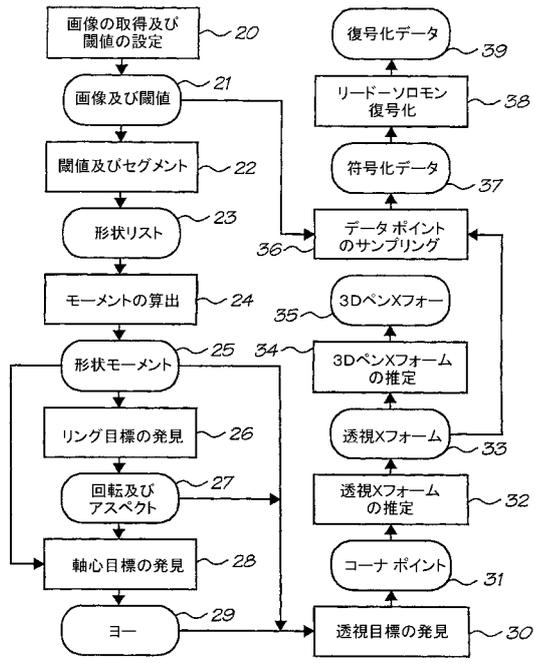
【 図 3 】



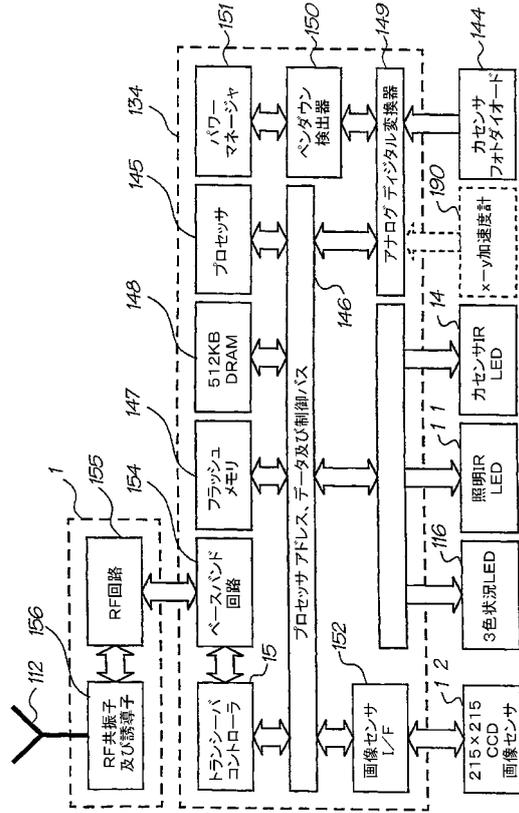
【 図 4 】



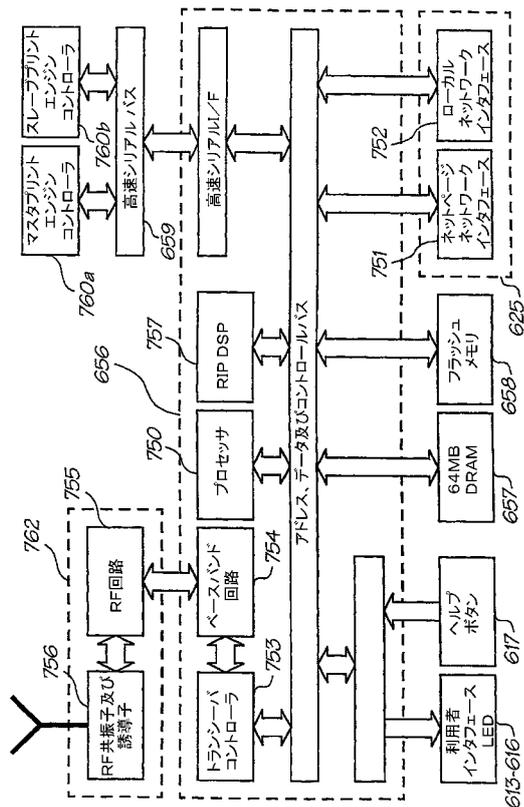
【 図 7 】



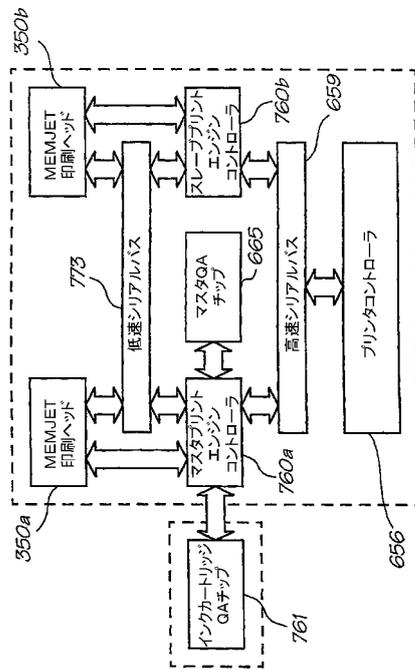
【 図 10 】



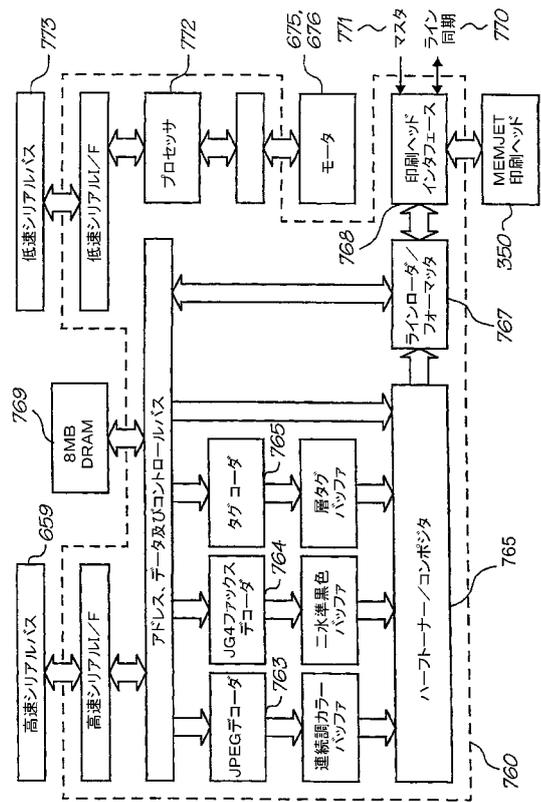
【 図 14 】



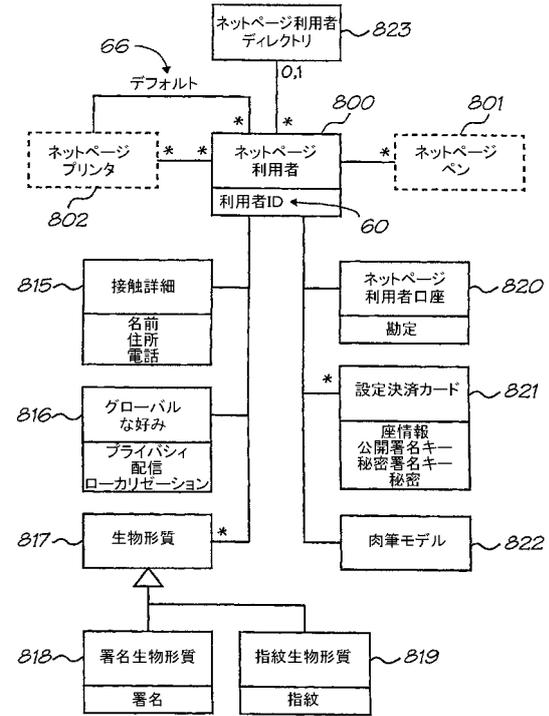
【 図 15 】



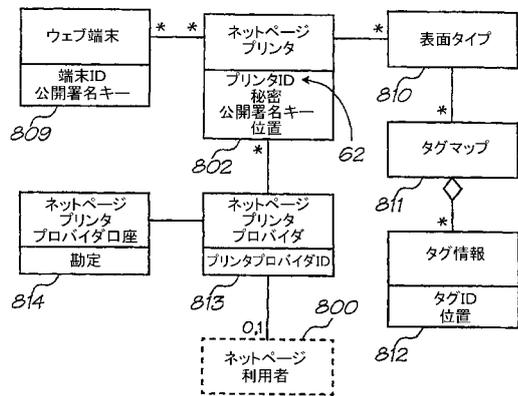
【 図 1 6 】



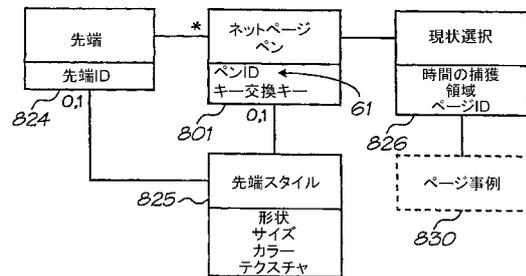
【 図 2 1 】



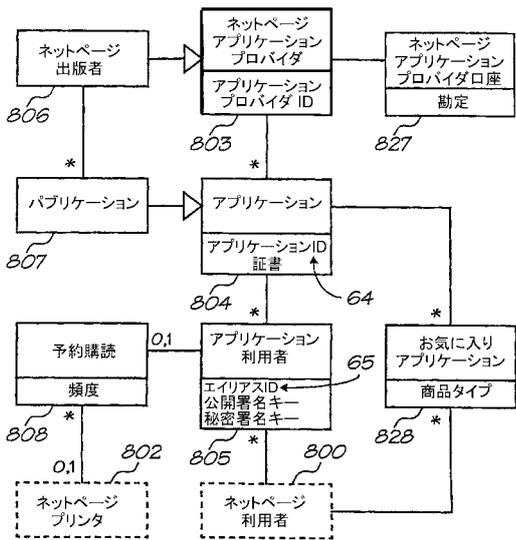
【 図 2 2 】



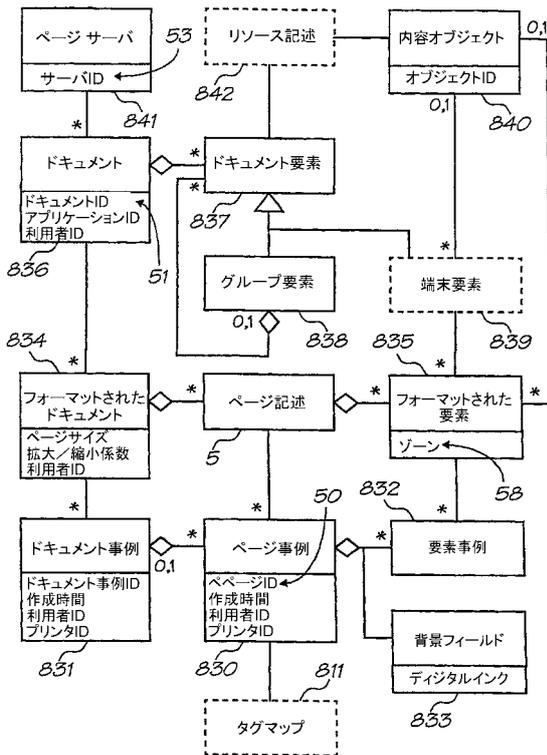
【 図 2 3 】



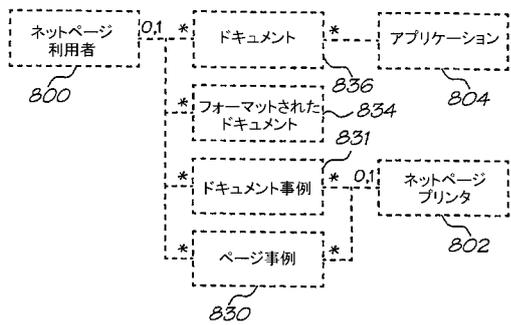
【 図 2 4 】



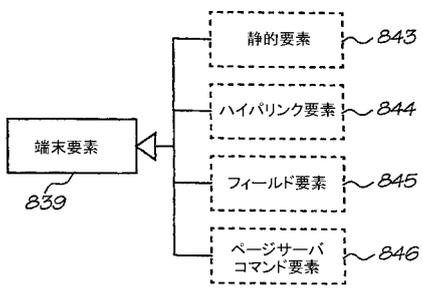
【 図 2 5 】



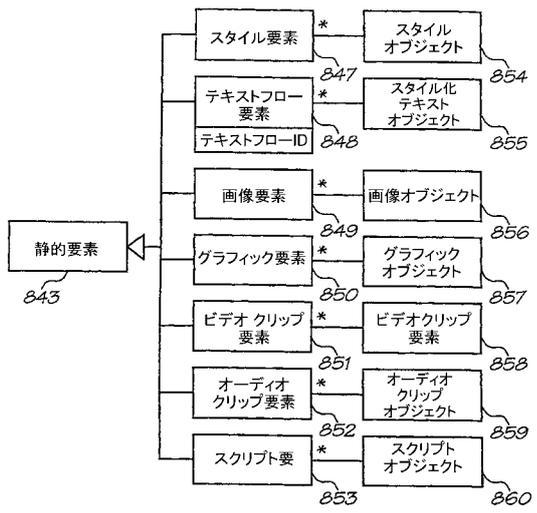
【 図 2 6 】



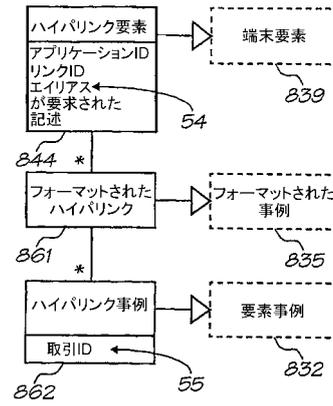
【 図 2 7 】



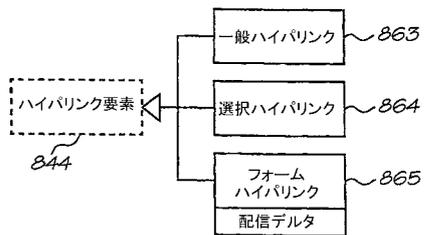
【図 28】



【図 29】



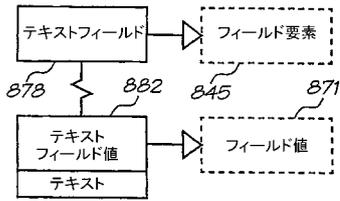
【図 30】



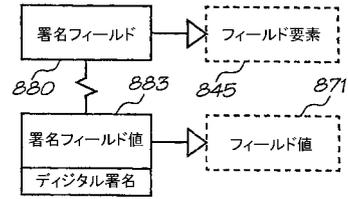
【図 31】



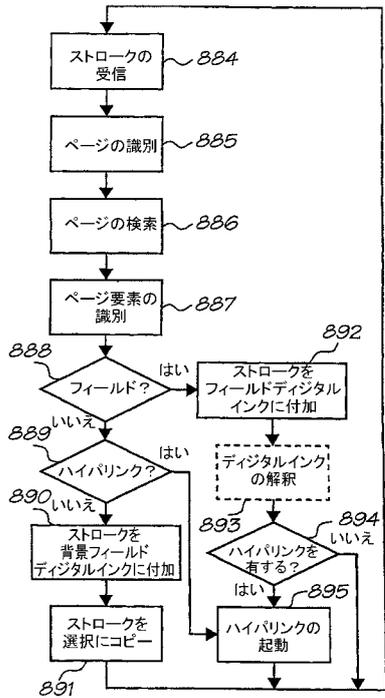
【図36】



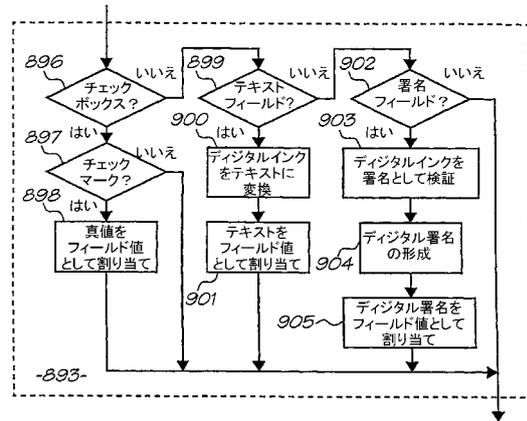
【図37】



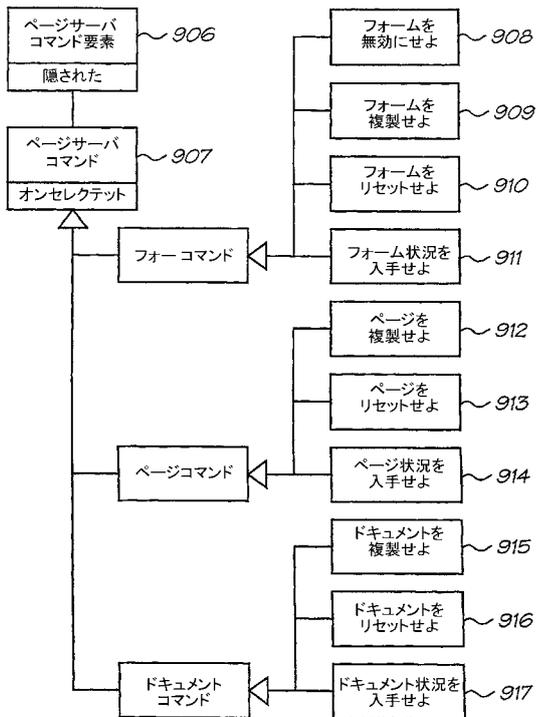
【図38】



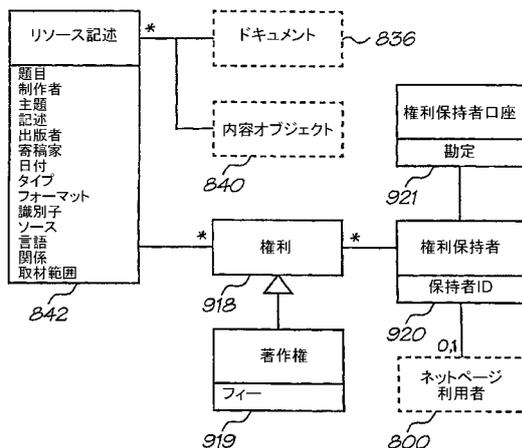
【図38a】



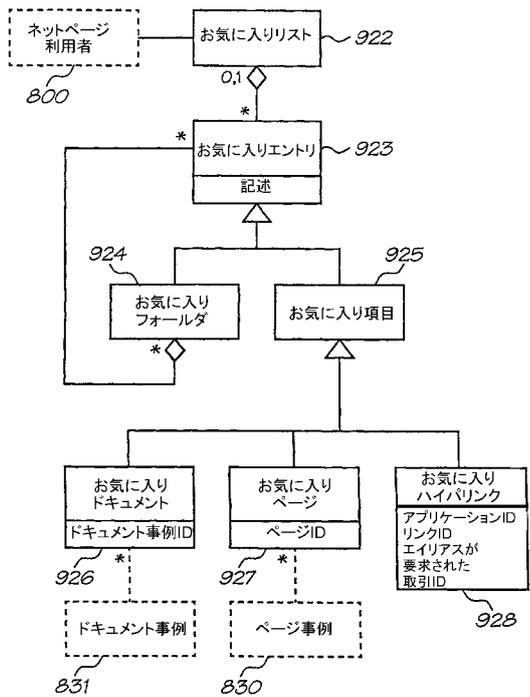
【 図 3 9 】



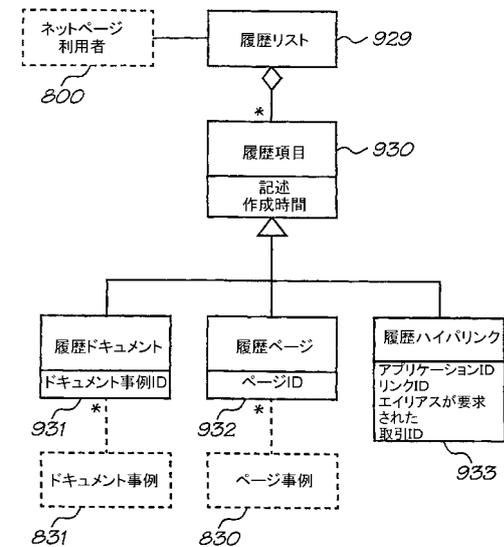
【 図 4 0 】



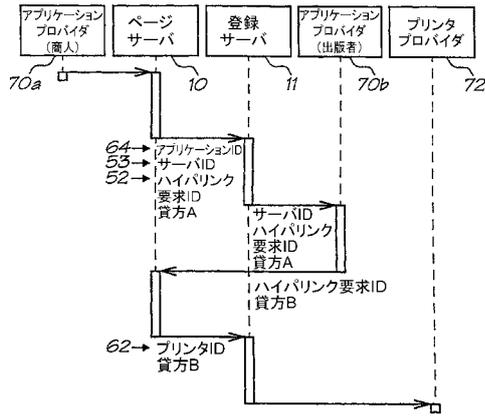
【 図 4 1 】



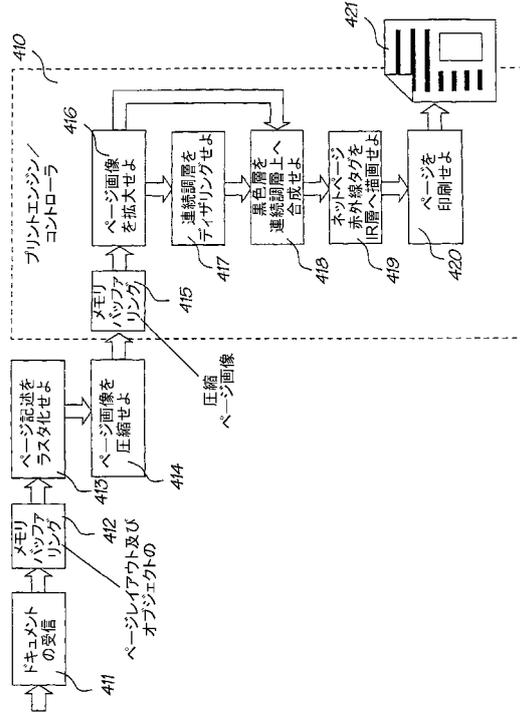
【 図 4 2 】



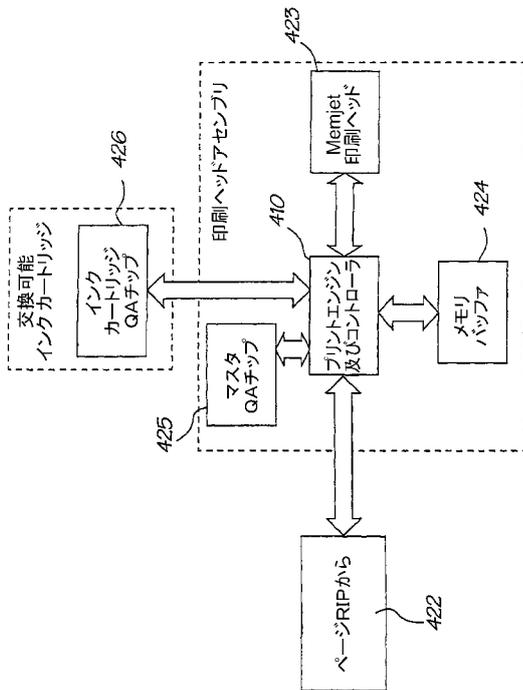
【 図 4 7 】



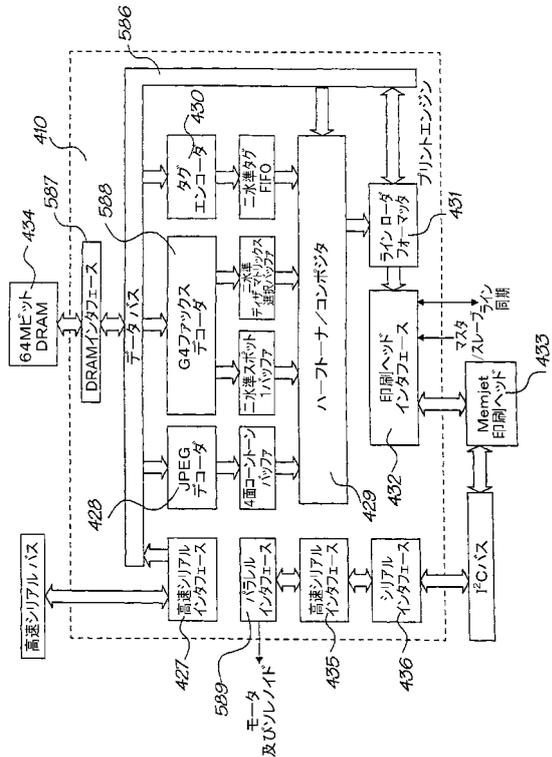
【 図 4 8 】



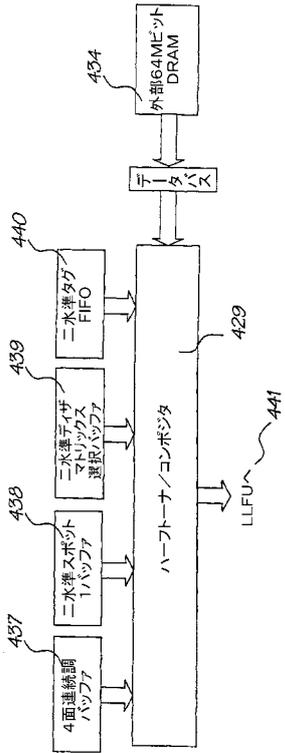
【 図 4 9 】



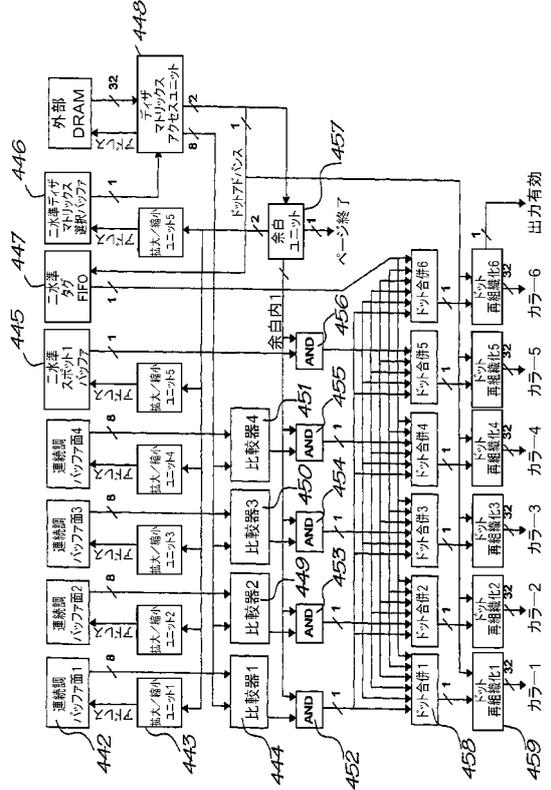
【 図 5 0 】



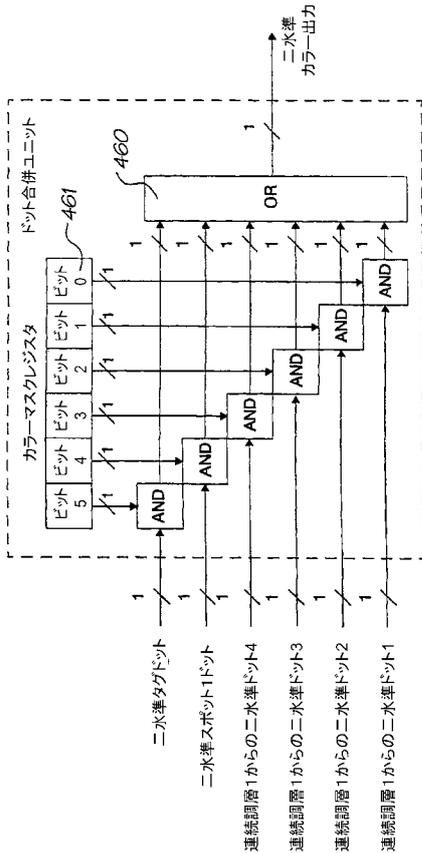
【 図 5 1 】



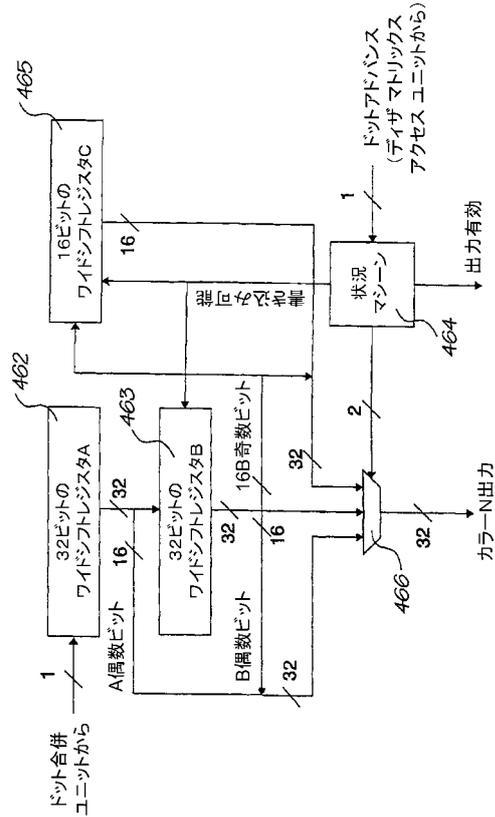
【 図 5 2 】



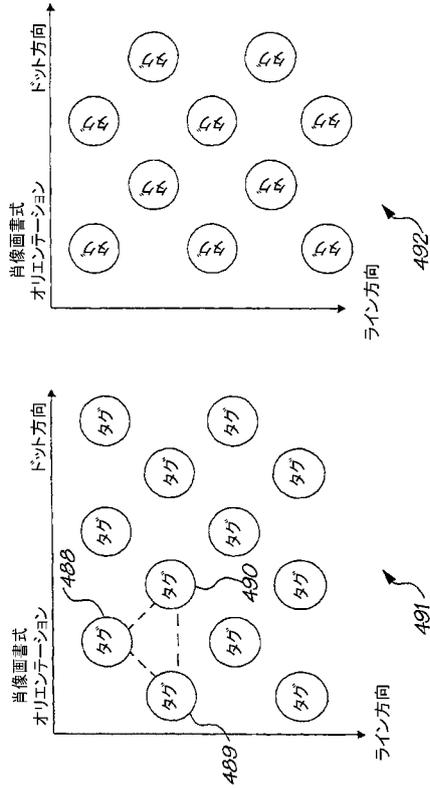
【 図 5 3 】



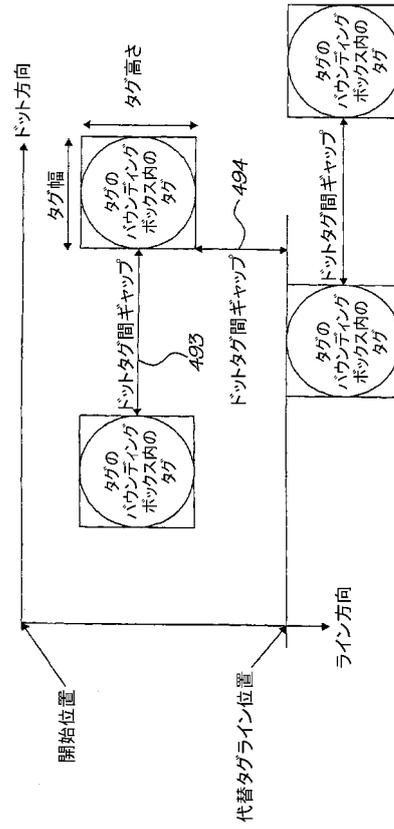
【 図 5 4 】



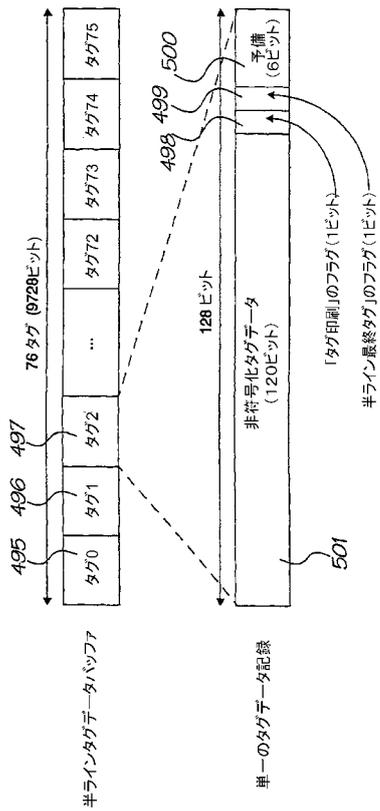
【 図 5 5 】



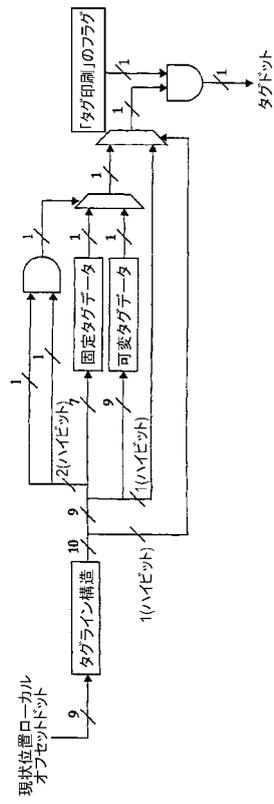
【 図 5 6 】



【 図 5 7 】



【 図 5 8 】



【国際公開パンフレット】

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
29 November 2001 (29.11.2001)

PCT

(10) International Publication Number
WO 01/89838 A1

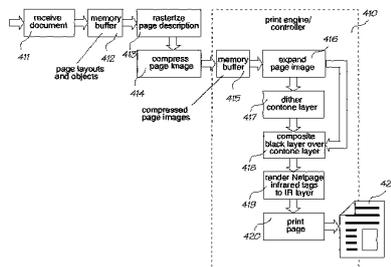
- (51) International Patent Classification: **B41J 2/04**, H04N 1/40, G06K 19/06
- (21) International Application Number: PCT/AU00/00517
- (22) International Filing Date: 24 May 2000 (24.05.2000)
- (25) Filing Language: English
- (26) Publication Language: English
- (71) Applicant (for all designated States except US): **SILVERBROOK RESEARCH PTY. LTD.** [AU/AU]; 393 Darling Street, Balmain, NSW 2041 (AU).
- (72) Inventors; and
(75) Inventors/Applicants (for US only): **LAPSTUN, Paul** [NO/AU]; 13 Duke Avenue, Rodd Point, NSW 2046 (AU). **WALMSLEY, Simon, Robert** [AU/AU]; Unit 3, 9 Pembroke Street, Epping, NSW 2121 (AU).
- (74) Agent: **SILVERBROOK RESEARCH PTY. LTD.**; 393 Darling Street, Balmain, NSW 2041 (AU).
- (81) Designated States (national): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.
- (84) Designated States (regional): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BI, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published: — with international search report

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.



(54) Title: PRINTED PAGE TAG ENCODER



(57) Abstract: A tag encoder for producing tags to be incorporated into a printed page is disclosed. The tag encoder has an input to receive a tag structure template, an input to receive fixed data bits, an input to receive variable data bit records and a tag generator outputting single bits depending on the position in the tag defined by the tag structure template and the fixed and variable data bits. The encoder has a redundancy encoder which utilises Reed-Solomon encoding. The tag encoder is present in a printer. In addition to the tag encoder, the printer has a contone image decoder to decode compressed continuous tone image planes and a bi-level decoder to decode any compressed bi-level image plane in the compressed data. These decoded image planes are combined with the output of the tag encoder in a halftone/compositor to produce a printed page carrying tagged areas. The tags are printed in ink which is invisible to the human eye. Such inks, could be IR or UV absorptive.

WO 01/89838 A1

WO 01/89838

PCT/AU00/00517

- 1 -

PRINTED PAGE TAG ENCODER**FIELD OF INVENTION**

The present invention relates generally to methods, systems and apparatus for interacting with computers.

5 In more specific terms, the present invention relates to the formatting of a coded tag, a tag to be added to a printed page during the printing of the page, and a tag encoder to effect the production of tags in accordance with the format. The tag encoder is particularly able to be implemented in a print engine/controller by which to produce printed pages incorporating tags, along with other graphic and textual matter.

10 The invention has been developed primarily to allow a large number of distributed users to interact with networked information via printed matter and optical sensors, thereby to obtain interactive printed matter on demand via high-speed networked color printers. Although the invention will largely be described herein with reference to this use, it will be appreciated that the invention is not limited to use in this field.

CO-PENDING APPLICATIONS

15 Various methods, systems and apparatus relating to the present invention are disclosed in the following co-pending applications filed by the applicant or assignee of the present invention simultaneously with the present application:

20 PCT/AU00/00518, PCT/AU00/00519, PCT/AU00/00520, PCT/AU00/00521,
PCT/AU00/00522, PCT/AU00/00523, PCT/AU00/00524, PCT/AU00/00525,
PCT/AU00/00526, PCT/AU00/00527, PCT/AU00/00528, PCT/AU00/00529,
PCT/AU00/00530, PCT/AU00/00531, PCT/AU00/00532, PCT/AU00/00533,
PCT/AU00/00534, PCT/AU00/00535, PCT/AU00/00536, PCT/AU00/00537,
PCT/AU00/00538, PCT/AU00/00539, PCT/AU00/00540, PCT/AU00/00541,
PCT/AU00/00542, PCT/AU00/00543, PCT/AU00/00544, PCT/AU00/00545,
25 PCT/AU00/00547, PCT/AU00/00546, PCT/AU00/00554, PCT/AU00/00556,
PCT/AU00/00557, PCT/AU00/00558, PCT/AU00/00559, PCT/AU00/00560,
PCT/AU00/00561, PCT/AU00/00562, PCT/AU00/00563, PCT/AU00/00564,
PCT/AU00/00565, PCT/AU00/00566, PCT/AU00/00567, PCT/AU00/00568,
PCT/AU00/00569, PCT/AU00/00570, PCT/AU00/00571, PCT/AU00/00572,
PCT/AU00/00573, PCT/AU00/00574, PCT/AU00/00575, PCT/AU00/00576,
30 PCT/AU00/00577, PCT/AU00/00578, PCT/AU00/00579, PCT/AU00/00581,
PCT/AU00/00580, PCT/AU00/00582, PCT/AU00/00587, PCT/AU00/00588,
PCT/AU00/00589, PCT/AU00/00583, PCT/AU00/00593, PCT/AU00/00590,
PCT/AU00/00591, PCT/AU00/00592, PCT/AU00/00584, PCT/AU00/00585,
PCT/AU00/00586, PCT/AU00/00594, PCT/AU00/00595, PCT/AU00/00596,
35 PCT/AU00/00597, PCT/AU00/00598, PCT/AU00/00516, PCT/AU00/00511,
PCT/AU00/00501, PCT/AU00/00502, PCT/AU00/00503, PCT/AU00/00504,
PCT/AU00/00505, PCT/AU00/00506, PCT/AU00/00507, PCT/AU00/00508,
PCT/AU00/00509, PCT/AU00/00510, PCT/AU00/00512, PCT/AU00/00513,
PCT/AU00/00514, PCT/AU00/00515

40 The disclosures of these co-pending applications are incorporated herein by cross-reference.

RECTIFIED SHEET (RULE 91)

WO 01/89838

PCT/AU00/00517

- 1a -

BACKGROUND

Paper is widely used to display and record information. Printed information is easier to read than information displayed on a computer screen. Hand-drawing and handwriting afford greater richness of expression than input via a computer keyboard and mouse. Moreover, paper doesn't run on batteries, can be read in bright light, more robustly
5 accepts coffee spills, and is portable and disposable.

Online publication has many advantages over traditional paper-based publication. From a consumer's point of view, information is available on demand, information can be navigated via hypertext links, searched and automatically personalized.

10 From the publisher's point of view, the costs of printing and physical distribution are eliminated, and the publication becomes more attractive to the advertisers who pay for it because it can be targeted to specific demographics and linked to product sites.

Online publication also has disadvantages. Computer screens are inferior to paper. At the same quality as a magazine page, an SVGA computer screen displays only about a fifth as much information. Both CRTs and LCDs have brightness and contrast problems, particularly when ambient light is strong, while ink on paper, being reflective rather

RECTIFIED SHEET (RULE 91)

WO 01/89838

PCT/AU00/00517

- 2 -

than emissive, is both bright and sharp in ambient light.

SUMMARY OF THE INVENTION

In one form the invention resides in a printed page tag encoder comprising:

- 5 an input at which to receive a tag structure template;
- an input at which to receive fixed data bits;
- an input at which to receive variable data bit records;
- and
- a tag dot generator outputting single bits depending on position in the tag defined by the tag structure template and said fixed and said variable data.

- 10 A print engine/controller that includes the present tag encoder preferably uses a high speed serial interface at which to receive compressed page data. Page data may include contone image planes that are decoded by a JPEG decoder and they may be scaled in the halftoner/compositor under control of a margin unit. A bi-level image plane may be decoded by a Group 4 facsimile decoder and it also can be scaled in the halftoner/compositor under control of the margin unit. A preferably infrared tag encoder within the print engine/controller serves to produce infrared data line by line in step with
- 15 processing of the image planes so as to place infrared ink printed tags into a printed page.

- The purpose of the tag encoder is to place tags over the printed page in such a way that they can be read at some later time by a suitable pen or equivalent device. Each tag can be a 2D package of data (although the tag may be printed on an arbitrary shaped surface) that is to be written out and able to read later. Typically there will some data to be stored in the package that is written to the page, although sometimes the mere presence of the package of data (our tag) is information in itself. With the present tag encoder it is desired to write out lots of these packets of data all over the page. The generation of these packets is controlled in any or all of size, structure, and how the data is stored inside. The tag encoder and the Tag Format Structure (described below) gives this control.
- 20

- The Tag Format Structure allows the tag designer to specify for a given tag which dots are printed as part of the physical printed tag structure and which dots are to be derived from the data. The data part of a tag is broken into variable and fixed portions. The fixed portions are the same data for each and every tag on the page, while the variable portions are specified for each tag. One limit case might be that all the data is variable, but just happens to contain the same value, thus making the data effectively fixed. Rather than force the user of the print engine/controller (PEC) to supply data for each and every tag always we allow the possibility of having fixed data for each tag. Exactly what data is in the tag will be completely application specific. One page may have tags that contain X/Y coordinate of the tag as the variable data, and a page id as the fixed data component. A page interactive pen (or equivalent) could subsequently read those coordinates back from tags on the page and perform actions depending on the position on the page. A different page may have fixed data over the entire page so that no matter where the pen clicks on the page, the same data will be returned. Another page may simply have giant tags over the page as a form of watermark – the mere presence of the tag is enough. The fixed data and variable data can be anything – as long as the reading application can extract the data from the read tag and then interpret the data it is useful.
- 25
- 30
- 35

- The structure of the tag is user-definable to enable different applications build appropriate structures to hold their data. Ideally a tag has some structure to help the locating software (in the pen) detect it, and some orientation features to enable the data bits to be extracted correctly. Finally, the data embedded in the tag should be redundantly encoded to allow the reading equipment (the pen) to correct errors due to dust, grime, dirt, reading noise etc.

- 40 The tag is defined in terms of 1600 dpi dots to enable nicely shaped tag structures. However it is not useful at present to print data dots on to a page where each data dot is represented by a single printed dot. The error introduced in

WO 01/89838

PCT/AU00/00517

- 3 -

the reading environment would be too severe. You would need at least a 3200 dpi scanner in the pen to be able to get the 1600 dpi dots back again. Consequently a tag designer will typically cluster a number of physical printed dots on the page to represent a single data dot. This cluster of printed dots is referred to as a macrodot since it represents a single logical dot, and is clustered together to ease dot detection and decoding algorithms in the reading device. Since the Tag Format Structure allows any output dot within a tag to come from any data bit the size and shape of a macrodot is completely arbitrary. The tag designer will design the macrodot based upon the reading and optical capabilities of the pen.

5 A Tag Encoder should ideally be capable of printing tags in landscape and portrait modes. A single Tag Format Structure that is internally rotated by the Tag Encoder is one way of doing it, but in our Tag Encoder we simply have the Tag Encoder read a pre-rotated Tag Format Structure to save the bother of rotating it ourselves.

10 Finally, in terms of placement of tags on a page, placing tags in a triangular grid is better than on a rectangular grid in terms of ink usage. Triangular grids are also convenient when placing tags on an arbitrarily curved surface, although our particular tag encoder only copes with rectangular planes. Thus the same tag interactive pen can read tags printed on other surfaces.

15 The tag encoder typically requires the presence of IR ink at the print head although other inks such as K might be used for tags in limited circumstances.

The tag encoder works to effect tag generation at speed, in step with whatever other image planes are being worked. It achieves speed by working with a predefined tag format into which fixed and variable components of a tag are fed to generate tags dot by dot delivering them line by line to a compositor as image planes are composited. It can encode fixed data for the page being printed, together with specific variable tag data values, into an error-correctable encoded tag which is subsequently printed, usually in infrared, or sometimes in black ink on the page. The tag encoder ideally regularly locates tags on a page, ideally placing tags on a preferably triangular grid. Those skilled in the art will recognise that other tag arrays beside triangular might be used. The tag encoder allows for both landscape and portrait orientations. Basic tag structures are rendered at 1600 dpi, while tag data is encoded as arbitrarily shaped macrodots (with a minimum size of 1 dot at 1600 dpi). The output dot stream might be created in an output order set to match a particular printer, although those skilled in the art will appreciate that other regimes might be evolved. Further, those skilled in the art will appreciate the advantages of use of infrared ink, not visible to the eye but detectable by appropriate sensors, and will realise that other inks may sometimes have a use.

20 Instead of sending data packages to the print engine/controller (PEC) already encoded, bandwidth to the PEC is reduced by having the PEC do the redundancy encoding. Specifically described is use of Reed-Solomon encoding, but it could equally be any other encoder. The PEC preferably encodes both the fixed and variable parts of the tag data.

25 The invention defines a template that gives a generic data package that includes dots that are always off, always on, and derived from the encoded data. This allows for development of any of a range of data package definitions, including macrodots of different sizes, large objects to help location, and so forth. Tag structures might be stored in associated DRAM where implementation does not involve fabrication of an all embracing chip. A trivial extension is to have the tag structure on-chip instead of in the extremal DRAM.

BRIEF DESCRIPTION OF THE DRAWINGS

30 Preferred and other embodiments of the invention will now be described, by way of non-limiting example only, with reference to the accompanying drawings, in which:

40 Figure 1 is a schematic of a the relationship between a sample printed netpage and its online page description;

Figure 2 is a schematic view of an interaction between a netpage pen, a netpage printer, a netpage page server, and

WO 01/89838

PCT/AU00/00517

- 4 -

- a netpage application server;
- Figure 3 illustrates a collection of netpage servers and printers interconnected via a network;
- Figure 4 is a schematic view of a high-level structure of a printed netpage and its online page description;
- Figure 5 is a plan view showing a structure of a netpage tag;
- 5 Figure 6 is a plan view showing a relationship between a set of the tags shown in Figure 5 and a field of view of a netpage sensing device in the form of a netpage pen;
- Figure 7 is a flowchart of a tag image processing and decoding algorithm;
- Figure 8 is a perspective view of a netpage pen and its associated tag-sensing field-of-view cone;
- Figure 9 is a perspective exploded view of the netpage pen shown in Figure 8;
- 10 Figure 10 is a schematic block diagram of a pen controller for the netpage pen shown in Figures 8 and 9;
- Figure 11 is a perspective view of a wall-mounted netpage printer;
- Figure 12 is a section through the length of the netpage printer of Figure 11;
- Figure 12a is an enlarged portion of Figure 12 showing a section of the duplexed print engines and glue wheel assembly;
- 15 Figure 13 is a detailed view of the ink cartridge, ink, air and glue paths, and print engines of the netpage printer of Figures 11 and 12;
- Figure 14 is a schematic block diagram of a printer controller for the netpage printer shown in Figures 11 and 12;
- Figure 15 is a schematic block diagram of duplexed print engine controllers and Memjet[®] printheads associated with the printer controller shown in Figure 14;
- 20 Figure 16 is a schematic block diagram of the print engine controller shown in Figures 14 and 15;
- Figure 17 is a perspective view of a single Memjet[®] printing element, as used in, for example, the netpage printer of Figures 10 to 12;
- Figure 18 is a perspective view of a small part of an array of Memjet[®] printing elements;
- 25 Figure 19 is a series of perspective views illustrating the operating cycle of the Memjet[®] printing element shown in Figure 13;
- Figure 20 is a perspective view of a short segment of a pagewidth Memjet[®] printhead;
- Figure 21 is a schematic view of a user class diagram;
- Figure 22 is a schematic view of a printer class diagram;
- Figure 23 is a schematic view of a pen class diagram;
- 30 Figure 24 is a schematic view of an application class diagram;
- Figure 25 is a schematic view of a document and page description class diagram;
- Figure 26 is a schematic view of a document and page ownership class diagram;
- Figure 27 is a schematic view of a terminal element specialization class diagram;
- Figure 28 is a schematic view of a static element specialization class diagram;
- 35 Figure 29 is a schematic view of a hyperlink element class diagram;
- Figure 30 is a schematic view of a hyperlink element specialization class diagram;
- Figure 31 is a schematic view of a hyperlinked group class diagram;
- Figure 32 is a schematic view of a form class diagram;
- Figure 33 is a schematic view of a digital ink class diagram;
- 40 Figure 34 is a schematic view of a field element specialization class diagram;
- Figure 35 is a schematic view of a checkbox field class diagram;

WO 01/89838

PCT/AU00/00517

- 5 -

- Figure 36 is a schematic view of a text field class diagram;
 Figure 37 is a schematic view of a signature field class diagram;
 Figure 38 is a flowchart of an input processing algorithm;
 Figure 38a is a detailed flowchart of one step of the flowchart of Figure 38;
 5 Figure 39 is a schematic view of a page server command element class diagram;
 Figure 40 is a schematic view of a resource description class diagram;
 Figure 41 is a schematic view of a favorites list class diagram;
 Figure 42 is a schematic view of a history list class diagram;
 Figure 43 is a schematic view of a subscription delivery protocol;
 10 Figure 44 is a schematic view of a hyperlink request class diagram;
 Figure 45 is a schematic view of a hyperlink activation protocol;
 Figure 46 is a schematic view of a form submission protocol;
 Figure 47 is a schematic view of a commission payment protocol;
 Figure 48 is a diagram illustrating data flow and the functions performed by the print engine controller.
 15 Figure 49 shows the print engine controller in the context of the overall printer system architecture.
 Figure 50 illustrates the print engine controller architecture.
 Figure 51 illustrates the external interfaces to the halftoner/compositor unit (HCU) of Figure 50.
 Figure 52 is a diagram showing internal circuitry to the HCU of Figure 51.
 Figure 53 shows a block diagram illustrating the process within the dot merger unit of Figure 52.
 20 Figure 54 shows a diagram illustrating the process within the dot reorganization unit of Figure 52.
 Figure 55 illustrates a placement of tags in portrait and landscape modes.
 Figure 56 represents the parameters used to define tag placement.
 Figure 57 indicates a half line tag data buffer structure.
 Figure 58 shows a circuit by which to generate a single tag dot.
 25 Figure 59 shows a Reed-Solomon based circuit encoding tag data.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

Note: Menjet™ is a trade mark of Silverbrook Research Pty Ltd, Australia.

- 30 In the preferred embodiment, the invention is configured to work with the netpage networked computer system, a detailed overview of which follows. It will be appreciated that not every implementation will necessarily embody all or even most of the specific details and extensions discussed below in relation to the basic system. However, the system is described in its most complete form to reduce the need for external reference when attempting to understand the context in which the preferred embodiments and aspects of the present invention operate.

- 35 In brief summary, the preferred form of the netpage system employs a computer interface in the form of a mapped surface, that is, a physical surface which contains references to a map of the surface maintained in a computer system. The map references can be queried by an appropriate sensing device. Depending upon the specific implementation, the map references may be encoded visibly or invisibly, and defined in such a way that a local query on the mapped surface yields an unambiguous map reference both within the map and among different maps. The computer system can contain information about features on the mapped surface, and such information can be retrieved based on map references
 40 supplied by a sensing device used with the mapped surface. The information thus retrieved can take the form of actions which are initiated by the computer system on behalf of the operator in response to the operator's interaction with the

WO 01/89838

PCT/AU00/00517

- 6 -

surface features.

In its preferred form, the netpage system relies on the production of, and human interaction with, netpages. These are pages of text, graphics and images printed on ordinary paper, but which work like interactive web pages. Information is encoded on each page using ink which is substantially invisible to the unaided human eye. The ink, however, and
5 thereby the coded data, can be sensed by an optically imaging pen and transmitted to the netpage system.

In the preferred form, active buttons and hyperlinks on each page can be clicked with the pen to request information from the network or to signal preferences to a network server. In one embodiment, text written by hand on a netpage is automatically recognized and converted to computer text in the netpage system, allowing forms to be filled in. In other embodiments, signatures recorded on a netpage are automatically verified, allowing e-commerce transactions to
10 be securely authorized.

As illustrated in Figure 1, a printed netpage 1 can represent a interactive form which can be filled in by the user both physically, on the printed page, and "electronically", via communication between the pen and the netpage system. The example shows a "Request" form containing name and address fields and a submit button. The netpage consists of graphic data 2 printed using visible ink, and coded data 3 printed as a collection of tags 4 using invisible ink. The
15 corresponding page description 5, stored on the netpage network, describes the individual elements of the netpage. In particular it describes the type and spatial extent (zone) of each interactive element (i.e. text field or button in the example), to allow the netpage system to correctly interpret input via the netpage. The submit button 6, for example, has a zone 7 which corresponds to the spatial extent of the corresponding graphic 8.

As illustrated in Figure 2, the netpage pen 101, a preferred form of which is shown in Figures 8 and 9 and described in more detail below, works in conjunction with a netpage printer 601, an Internet-connected printing appliance for home, office or mobile use. The pen is wireless and communicates securely with the netpage printer via a short-range
20 radio link 9.

The netpage printer 601, a preferred form of which is shown in Figures 11 to 13 and described in more detail below, is able to deliver, periodically or on demand, personalized newspapers, magazines, catalogs, brochures and other publications, all printed at high quality as interactive netpages. Unlike a personal computer, the netpage printer is an appliance which can be, for example, wall-mounted adjacent to an area where the morning news is first consumed, such as
25 in a user's kitchen, near a breakfast table, or near the household's point of departure for the day. It also comes in tabletop, desktop, portable and miniature versions.

Netpages printed at their point of consumption combine the ease-of-use of paper with the timeliness and interactivity of an interactive medium.
30

As shown in Figure 2, the netpage pen 101 interacts with the coded data on a printed netpage 1 and communicates, via a short-range radio link 9, the interaction to a netpage printer. The printer 601 sends the interaction to the relevant netpage page server 10 for interpretation. In appropriate circumstances, the page server sends a corresponding message to application computer software running on a netpage application server 13. The application server may in turn send a
35 response which is printed on the originating printer.

The netpage system is made considerably more convenient in the preferred embodiment by being used in conjunction with high-speed microelectromechanical system (MEMS) based inkjet (Memjet™) printers. In the preferred form of this technology, relatively high-speed and high-quality printing is made more affordable to consumers. In its preferred form, a netpage publication has the physical characteristics of a traditional newsmagazine, such as a set of letter-size glossy pages printed in full color on both sides, bound together for easy navigation and comfortable handling.
40

The netpage printer exploits the growing availability of broadband Internet access. Cable service is available to

WO 01/89838

PCT/AU00/00517

- 7 -

95% of households in the United States, and cable modem service offering broadband Internet access is already available to 20% of these. The netpage printer can also operate with slower connections, but with longer delivery times and lower image quality. Indeed, the netpage system can be enabled using existing consumer inkjet and laser printers, although the system will operate more slowly and will therefore be less acceptable from a consumer's point of view. In other embodiments, the netpage system is hosted on a private intranet. In still other embodiments, the netpage system is hosted on a single computer or computer-enabled device, such as a printer.

Netpage publication servers 14 on the netpage network are configured to deliver print-quality publications to netpage printers. Periodical publications are delivered automatically to subscribing netpage printers via pointcasting and multicasting Internet protocols. Personalized publications are filtered and formatted according to individual user profiles.

10 A netpage printer can be configured to support any number of pens, and a pen can work with any number of netpage printers. In the preferred implementation, each netpage pen has a unique identifier. A household may have a collection of colored netpage pens, one assigned to each member of the family. This allows each user to maintain a distinct profile with respect to a netpage publication server or application server.

15 A netpage pen can also be registered with a netpage registration server 11 and linked to one or more payment card accounts. This allows e-commerce payments to be securely authorized using the netpage pen. The netpage registration server compares the signature captured by the netpage pen with a previously registered signature, allowing it to authenticate the user's identity to an e-commerce server. Other biometrics can also be used to verify identity. A version of the netpage pen includes fingerprint scanning, verified in a similar way by the netpage registration server.

20 Although a netpage printer may deliver periodicals such as the morning newspaper without user intervention, it can be configured never to deliver unsolicited junk mail. In its preferred form, it only delivers periodicals from subscribed or otherwise authorized sources. In this respect, the netpage printer is unlike a fax machine or e-mail account which is visible to any junk mailer who knows the telephone number or email address.

1 Netpage System Architecture

25 Each object model in the system is described using a Unified Modeling Language (UML) class diagram. A class diagram consists of a set of object classes connected by relationships, and two kinds of relationships are of interest here: associations and generalizations. An association represents some kind of relationship between objects, i.e. between instances of classes. A generalization relates actual classes, and can be understood in the following way: if a class is thought of as the set of all objects of that class, and class A is a generalization of class B, then B is simply a subset of A. The UML does not directly support second-order modelling - i.e. classes of classes.

30 Each class is drawn as a rectangle labelled with the name of the class. It contains a list of the attributes of the class, separated from the name by a horizontal line, and a list of the operations of the class, separated from the attribute list by a horizontal line. In the class diagrams which follow, however, operations are never modelled.

35 An association is drawn as a line joining two classes, optionally labelled at either end with the multiplicity of the association. The default multiplicity is one. An asterisk (*) indicates a multiplicity of "many", i.e. zero or more. Each association is optionally labelled with its name, and is also optionally labelled at either end with the role of the corresponding class. An open diamond indicates an aggregation association ("is-part-of"), and is drawn at the aggregator end of the association line.

A generalization relationship ("is-a") is drawn as a solid line joining two classes, with an arrow (in the form of an open triangle) at the generalization end.

40 When a class diagram is broken up into multiple diagrams, any class which is duplicated is shown with a dashed outline in all but the main diagram which defines it. It is shown with attributes only where it is defined.

WO 01/89838

PCT/AU00/00517

- 8 -

1.1 NETPAGES

Netpages are the foundation on which a netpage network is built. They provide a paper-based user interface to published information and interactive services.

5 A netpage consists of a printed page (or other surface region) invisibly tagged with references to an online description of the page. The online page description is maintained persistently by a netpage page server. The page description describes the visible layout and content of the page, including text, graphics and images. It also describes the input elements on the page, including buttons, hyperlinks, and input fields. A netpage allows markings made with a netpage pen on its surface to be simultaneously captured and processed by the netpage system.

10 Multiple netpages can share the same page description. However, to allow input through otherwise identical pages to be distinguished, each netpage is assigned a unique page identifier. This page ID has sufficient precision to distinguish between a very large number of netpages.

Each reference to the page description is encoded in a printed tag. The tag identifies the unique page on which it appears, and thereby indirectly identifies the page description. The tag also identifies its own position on the page. Characteristics of the tags are described in more detail below.

15 Tags are printed in infrared-absorptive ink on any substrate which is infrared-reflective, such as ordinary paper. Near-infrared wavelengths are invisible to the human eye but are easily sensed by a solid-state image sensor with an appropriate filter.

20 A tag is sensed by an area image sensor in the netpage pen, and the tag data is transmitted to the netpage system via the nearest netpage printer. The pen is wireless and communicates with the netpage printer via a short-range radio link. Tags are sufficiently small and densely arranged that the pen can reliably image at least one tag even on a single click on the page. It is important that the pen recognize the page ID and position on every interaction with the page, since the interaction is stateless. Tags are error-correctably encoded to make them partially tolerant to surface damage.

The netpage page server maintains a unique page instance for each printed netpage, allowing it to maintain a distinct set of user-supplied values for input fields in the page description for each printed netpage.

25 The relationship between the page description, the page instance, and the printed netpage is shown in Figure 4. The page instance is associated with both the netpage printer which printed it and, if known, the netpage user who requested it.

1.2 NETPAGE TAGS

1.2.1 Tag Data Content

30 In a preferred form, each tag identifies the region in which it appears, and the location of that tag within the region. A tag may also contain flags which relate to the region as a whole or to the tag. One or more flag bits may, for example, signal a tag sensing device to provide feedback indicative of a function associated with the immediate area of the tag, without the sensing device having to refer to a description of the region. A netpage pen may, for example, illuminate an "active area" LED when in the zone of a hyperlink.

35 As will be more clearly explained below, in a preferred embodiment, each tag contains an easily recognized invariant structure which aids initial detection, and which assists in minimizing the effect of any warp induced by the surface or by the sensing process. The tags preferably tile the entire page, and are sufficiently small and densely arranged that the pen can reliably image at least one tag even on a single click on the page. It is important that the pen recognize the page ID and position on every interaction with the page, since the interaction is stateless.

40 In a preferred embodiment, the region to which a tag refers coincides with an entire page, and the region ID encoded in the tag is therefore synonymous with the page ID of the page on which the tag appears. In other embodiments,

WO 01/89838

PCT/AU00/00517

- 9 -

the region to which a tag refers can be an arbitrary subregion of a page or other surface. For example, it can coincide with the zone of an interactive element, in which case the region ID can directly identify the interactive element.

Table 1 - Tag data

Field	Precision (bits)
Region ID	100
Tag ID	16
Flags	4
Total	120

5 Each tag contains 120 bits of information, typically allocated as shown in Table 1. Assuming a maximum tag density of 64 per square inch, a 16-bit tag ID supports a region size of up to 1024 square inches. Larger regions can be mapped continuously without increasing the tag ID precision simply by using abutting regions and maps. The 100-bit region ID allows 2^{100} ($\sim 10^{30}$ or a million trillion trillion) different regions to be uniquely identified.

1.2.2 Tag Data Encoding

10 The 120 bits of tag data are redundantly encoded using a (15, 5) Reed-Solomon code. This yields 360 encoded bits consisting of 6 codewords of 15 4-bit symbols each. The (15, 5) code allows up to 5 symbol errors to be corrected per codeword, i.e. it is tolerant of a symbol error rate of up to 33% per codeword.

15 Each 4-bit symbol is represented in a spatially coherent way in the tag, and the symbols of the six codewords are interleaved spatially within the tag. This ensures that a burst error (an error affecting multiple spatially adjacent bits) damages a minimum number of symbols overall and a minimum number of symbols in any one codeword, thus maximising the likelihood that the burst error can be fully corrected.

1.2.3 Physical Tag Structure

20 The physical representation of the tag, shown in Figure 5, includes fixed target structures 15, 16, 17 and variable data areas 18. The fixed target structures allow a sensing device such as the netpage pen to detect the tag and infer its three-dimensional orientation relative to the sensor. The data areas contain representations of the individual bits of the encoded tag data.

25 To achieve proper tag reproduction, the tag is rendered at a resolution of 256x256 dots. When printed at 1600 dots per inch this yields a tag with a diameter of about 4 mm. At this resolution the tag is designed to be surrounded by a "quiet area" of radius 16 dots. Since the quiet area is also contributed by adjacent tags, it only adds 16 dots to the effective diameter of the tag.

30 The tag includes six target structures. A detection ring 15 allows the sensing device to initially detect the tag. The ring is easy to detect because it is rotationally invariant and because a simple correction of its aspect ratio removes most of the effects of perspective distortion. An orientation axis 16 allows the sensing device to determine the approximate planar orientation of the tag due to the yaw of the sensor. The orientation axis is skewed to yield a unique orientation. Four perspective targets 17 allow the sensing device to infer an accurate two-dimensional perspective transform of the tag and hence an accurate three-dimensional position and orientation of the tag relative to the sensor.

All target structures are redundantly large to improve their immunity to noise.

35 The overall tag shape is circular. This supports, amongst other things, optimal tag packing on an irregular triangular grid. In combination with the circular detection ring, this makes a circular arrangement of data bits within the tag optimal. To maximise its size, each data bit is represented by a radial wedge in the form of an area bounded by two

WO 01/89838

PCT/AU00/00517

- 10 -

radial lines and two concentric circular arcs. Each wedge has a minimum dimension of 8 dots at 1600 dpi and is designed so that its base (its inner arc), is at least equal to this minimum dimension. The height of the wedge in the radial direction is always equal to the minimum dimension. Each 4-bit data symbol is represented by an array of 2x2 wedges.

5 The 15 4-bit data symbols of each of the six codewords are allocated to the four concentric symbol rings 18a to 18d in interleaved fashion. Symbols are allocated alternately in circular progression around the tag.

The interleaving is designed to maximise the average spatial distance between any two symbols of the same codeword.

10 In order to support "single-click" interaction with a tagged region via a sensing device, the sensing device must be able to see at least one entire tag in its field of view no matter where in the region or at what orientation it is positioned. The required diameter of the field of view of the sensing device is therefore a function of the size and spacing of the tags.

Assuming a circular tag shape, the minimum diameter of the sensor field of view is obtained when the tags are tiled on an equilateral triangular grid, as shown in Figure 6.

1.2.5 Tag Image Processing and Decoding

15 The tag image processing and decoding performed by a sensing device such as the netpage pen is shown in Figure 7. While a captured image is being acquired from the image sensor, the dynamic range of the image is determined (at 20). The center of the range is then chosen as the binary threshold for the image 21. The image is then thresholded and segmented into connected pixel regions (i.e. shapes 23) (at 22). Shapes which are too small to represent tag target structures are discarded. The size and centroid of each shape is also computed.

20 Binary shape moments 25 are then computed (at 24) for each shape, and these provide the basis for subsequently locating target structures. Central shape moments are by their nature invariant of position, and can be easily made invariant of scale, aspect ratio and rotation.

25 The ring target structure 15 is the first to be located (at 26). A ring has the advantage of being very well behaved when perspective-distorted. Matching proceeds by aspect-normalizing and rotation-normalizing each shape's moments. Once its second-order moments are normalized the ring is easy to recognize even if the perspective distortion was significant. The ring's original aspect and rotation 27 together provide a useful approximation of the perspective transform.

30 The axis target structure 16 is the next to be located (at 28). Matching proceeds by applying the ring's normalizations to each shape's moments, and rotation-normalizing the resulting moments. Once its second-order moments are normalized the axis target is easily recognized. Note that one third order moment is required to disambiguate the two possible orientations of the axis. The shape is deliberately skewed to one side to make this possible. Note also that it is only possible to rotation-normalize the axis target after it has had the ring's normalizations applied, since the perspective distortion can hide the axis target's axis. The axis target's original rotation provides a useful approximation of the tag's rotation due to pen yaw 29.

40 The four perspective target structures 17 are the last to be located (at 30). Good estimates of their positions are computed based on their known spatial relationships to the ring and axis targets, the aspect and rotation of the ring, and the rotation of the axis. Matching proceeds by applying the ring's normalizations to each shape's moments. Once their second-order moments are normalized the circular perspective targets are easy to recognize, and the target closest to each estimated position is taken as a match. The original centroids of the four perspective targets are then taken to be the

WO 01/89838

PCT/AU00/00517

- 11 -

perspective-distorted corners 31 of a square of known size in tag space, and an eight-degree-of-freedom perspective transform 33 is inferred (at 32) based on solving the well-understood equations relating the four tag-space and image-space point pairs.

5 The inferred tag-space to image-space perspective transform is used to project (at 36) each known data bit position in tag space into image space where the real-valued position is used to bilinearly interpolate (at 36) the four relevant adjacent pixels in the input image. The previously computed image threshold 21 is used to threshold the result to produce the final bit value 37.

10 Once all 360 data bits 37 have been obtained in this way, each of the six 60-bit Reed-Solomon codewords is decoded (at 38) to yield 20 decoded bits 39, or 120 decoded bits in total. Note that the codeword symbols are sampled in codeword order, so that codewords are implicitly de-interleaved during the sampling process.

15 The ring target 15 is only sought in a subarea of the image whose relationship to the image guarantees that the ring, if found, is part of a complete tag. If a complete tag is not found and successfully decoded, then no pen position is recorded for the current frame. Given adequate processing power and ideally a non-minimal field of view 193, an alternative strategy involves seeking another tag in the current image.

20 The obtained tag data indicates the identity of the region containing the tag and the position of the tag within the region. An accurate position 35 of the pen nib in the region, as well as the overall orientation 35 of the pen, is then inferred (at 34) from the perspective transform 33 observed on the tag and the known spatial relationship between the pen's physical axis and the pen's optical axis.

1.2.6 Tag Map

25 Decoding a tag results in a region ID, a tag ID, and a tag-relative pen transform. Before the tag ID and the tag-relative pen location can be translated into an absolute location within the tagged region, the location of the tag within the region must be known. This is given by a tag map, a function which maps each tag ID in a tagged region to a corresponding location. The tag map class diagram is shown in Figure 22, as part of the netpage printer class diagram.

30 A tag map reflects the scheme used to tile the surface region with tags, and this can vary according to surface type. When multiple tagged regions share the same tiling scheme and the same tag numbering scheme, they can also share the same tag map.

The tag map for a region must be retrievable via the region ID. Thus, given a region ID, a tag ID and a pen transform, the tag map can be retrieved, the tag ID can be translated into an absolute tag location within the region, and the tag-relative pen location can be added to the tag location to yield an absolute pen location within the region.

35 1.2.7 Tagging Schemes

Two distinct surface coding schemes are of interest, both of which use the tag structure described earlier in this section. The preferred coding scheme uses "location-indicating" tags as already discussed. An alternative coding scheme uses object-indicating tags.

40 A location-indicating tag contains a tag ID which, when translated through the tag map associated with the tagged

WO 01/89838

PCT/AU00/00517

- 12 -

region, yields a unique tag location within the region. The tag-relative location of the pen is added to this tag location to yield the location of the pen within the region. This in turn is used to determine the location of the pen relative to a user interface element in the page description associated with the region. Not only is the user interface element itself identified, but a location relative to the user interface element is identified. Location-indicating tags therefore trivially support the capture of an absolute pen path in the zone of a particular user interface element.

An object-indicating tag contains a tag ID which directly identifies a user interface element in the page description associated with the region. All the tags in the zone of the user interface element identify the user interface element, making them all identical and therefore indistinguishable. Object-indicating tags do not, therefore, support the capture of an absolute pen path. They do, however, support the capture of a relative pen path. So long as the position sampling frequency exceeds twice the encountered tag frequency, the displacement from one sampled pen position to the next within a stroke can be unambiguously determined.

With either tagging scheme, the tags function in cooperation with associated visual elements on the netpage as user interactive elements in that a user can interact with the printed page using an appropriate sensing device in order for tag data to be read by the sensing device and for an appropriate response to be generated in the netpage system.

1.3 DOCUMENT AND PAGE DESCRIPTIONS

A preferred embodiment of a document and page description class diagram is shown in Figures 25 and 26.

In the netpage system a document is described at three levels. At the most abstract level the document 836 has a hierarchical structure whose terminal elements 839 are associated with content objects 840 such as text objects, text style objects, image objects, etc. Once the document is printed on a printer with a particular page size and according to a particular user's scale factor preference, the document is paginated and otherwise formatted. Formatted terminal elements 835 will in some cases be associated with content objects which are different from those associated with their corresponding terminal elements, particularly where the content objects are style-related. Each printed instance of a document and page is also described separately, to allow input captured through a particular page instance 830 to be recorded separately from input captured through other instances of the same page description.

The presence of the most abstract document description on the page server allows a user to request a copy of a document without being forced to accept the source document's specific format. The user may be requesting a copy through a printer with a different page size, for example. Conversely, the presence of the formatted document description on the page server allows the page server to efficiently interpret user actions on a particular printed page.

A formatted document 834 consists of a set of formatted page descriptions 5, each of which consists of a set of formatted terminal elements 835. Each formatted element has a spatial extent or zone 58 on the page. This defines the active area of input elements such as hyperlinks and input fields.

A document instance 831 corresponds to a formatted document 834. It consists of a set of page instances 830, each of which corresponds to a page description 5 of the formatted document. Each page instance 830 describes a single unique printed netpage 1, and records the page ID 50 of the netpage. A page instance is not part of a document instance if it represents a copy of a page requested in isolation.

WO 01/89838

PCT/AU00/00517

- 13 -

5 A page instance consists of a set of terminal element instances 832. An element instance only exists if it records instance-specific information. Thus, a hyperlink instance exists for a hyperlink element because it records a transaction ID 55 which is specific to the page instance, and a field instance exists for a field element because it records input specific to the page instance. An element instance does not exist, however, for static elements such as textflows.

10 A terminal element can be a static element 843, a hyperlink element 844, a field element 845 or a page server command element 846, as shown in Figure 27. A static element 843 can be a style element 847 with an associated style object 854, a textflow element 848 with an associated styled text object 855, an image element 849 with an associated image element 856, a graphic element 850 with an associated graphic object 857, a video clip element 851 with an associated video clip object 858, an audio clip element 852 with an associated audio clip object 859, or a script element 853 with an associated script object 860, as shown in Figure 28.

15 A page instance has a background field 833 which is used to record any digital ink captured on the page which does not apply to a specific input element.

In the preferred form of the invention, a tag map 811 is associated with each page instance to allow tags on the page to be translated into locations on the page.

1.4 THE NETPAGE NETWORK

20 In a preferred embodiment, a netpage network consists of a distributed set of netpage page servers 10, netpage registration servers 11, netpage ID servers 12, netpage application servers 13, netpage publication servers 14, and netpage printers 601 connected via a network 19 such as the Internet, as shown in Figure 3.

25 The netpage registration server 11 is a server which records relationships between users, pens, printers, applications and publications, and thereby authorizes various network activities. It authenticates users and acts as a signing proxy on behalf of authenticated users in application transactions. It also provides handwriting recognition services. As described above, a netpage page server 10 maintains persistent information about page descriptions and page instances. The netpage network includes any number of page servers, each handling a subset of page instances. Since a page server also maintains user input values for each page instance, clients such as netpage printers send netpage input directly to the appropriate page server. The page server interprets any such input relative to the description of the corresponding page.

30 A netpage ID server 12 allocates document IDs 51 on demand, and provides load-balancing of page servers via its ID allocation scheme.

35 A netpage printer uses the Internet Distributed Name System (DNS), or similar, to resolve a netpage page ID 50 into the network address of the netpage page server handling the corresponding page instance.

A netpage application server 13 is a server which hosts interactive netpage applications. A netpage publication server 14 is an application server which publishes netpage documents to netpage printers. They are described in detail in Section 2.

40 Netpage servers can be hosted on a variety of network server platforms from manufacturers such as IBM, Hewlett-Packard, and Sun. Multiple netpage servers can run concurrently on a single host, and a single server can be distributed

WO 01/89838

PCT/AU00/00517

- 14 -

over a number of hosts. Some or all of the functionality provided by netpage servers, and in particular the functionality provided by the ID server and the page server, can also be provided directly in a netpage appliance such as a netpage printer, in a computer workstation, or on a local network.

1.5 THE NETPAGE PRINTER

5 The netpage printer 601 is an appliance which is registered with the netpage system and prints netpage documents on demand and via subscription. Each printer has a unique printer ID 62, and is connected to the netpage network via a network such as the Internet, ideally via a broadband connection.

10 Apart from identity and security settings in non-volatile memory, the netpage printer contains no persistent storage. As far as a user is concerned, "the network is the computer". Netpages function interactively across space and time with the help of the distributed netpage page servers 10, independently of particular netpage printers.

15 The netpage printer receives subscribed netpage documents from netpage publication servers 14. Each document is distributed in two parts: the page layouts, and the actual text and image objects which populate the pages. Because of personalization, page layouts are typically specific to a particular subscriber and so are pointcast to the subscriber's printer via the appropriate page server. Text and image objects, on the other hand, are typically shared with other subscribers, and so are multicast to all subscribers' printers and the appropriate page servers.

20 The netpage publication server optimizes the segmentation of document content into pointcasts and multicasts. After receiving the pointcast of a document's page layouts, the printer knows which multicasts, if any, to listen to.

Once the printer has received the complete page layouts and objects that define the document to be printed, it can print the document.

25 The printer rasterizes and prints odd and even pages simultaneously on both sides of the sheet. It contains duplexed print engine controllers 760 and print engines utilizing Menjet™ printheads 350 for this purpose.

30 The printing process consists of two decoupled stages: rasterization of page descriptions, and expansion and printing of page images. The raster image processor (RIP) consists of one or more standard DSPs 757 running in parallel. The duplexed print engine controllers consist of custom processors which expand, dither and print page images in real time, synchronized with the operation of the printheads in the print engines.

Printers not enabled for IR printing have the option to print tags using IR-absorptive black ink, although this restricts tags to otherwise empty areas of the page. Although such pages have more limited functionality than IR-printed pages, they are still classed as netpages.

35 A normal netpage printer prints netpages on sheets of paper. More specialised netpage printers may print onto more specialised surfaces, such as globes. Each printer supports at least one surface type, and supports at least one tag tiling scheme, and hence tag map, for each surface type. The tag map 811 which describes the tag tiling scheme actually used to print a document becomes associated with that document so that the document's tags can be correctly interpreted.

40 Figure 2 shows the netpage printer class diagram, reflecting printer-related information maintained by a registration

WO 01/89838

PCT/AU00/00517

- 15 -

server 11 on the netpage network.

A preferred embodiment of the netpage printer is described in greater detail in Section 6 below, with reference to Figures 11 to 16.

5 **1.5.1 Memjet™ Printheads**

The netpage system can operate using printers made with a wide range of digital printing technologies, including thermal inkjet, piezoelectric inkjet, laser electrophotographic, and others. However, for wide consumer acceptance, it is desirable that a netpage printer have the following characteristics:

- 10 • photographic quality color printing
- high quality text printing
- high reliability
- low printer cost
- low ink cost
- low paper cost
- 15 • simple operation
- nearly silent printing
- high printing speed
- simultaneous double sided printing
- compact form factor
- 20 • low power consumption

No commercially available printing technology has all of these characteristics.

To enable to production of printers with these characteristics, the present applicant has invented a new print technology, referred to as Memjet™ technology. Memjet™ is a drop-on-demand inkjet technology that incorporates pagewidth
25 printheads fabricated using microelectromechanical systems (MEMS) technology. Figure 17 shows a single printing element 300 of a Memjet™ printhead. The netpage wallprinter incorporates 168960 printing elements 300 to form a 1600 dpi pagewidth duplex printer. This printer simultaneously prints cyan, magenta, yellow, black, and infrared inks as well as paper conditioner and ink fixative.

30 The printing element 300 is approximately 110 microns long by 32 microns wide. Arrays of these printing elements are formed on a silicon substrate 301 that incorporates CMOS logic, data transfer, timing, and drive circuits (not shown).

Major elements of the printing element 300 are the nozzle 302, the nozzle rim 303, the nozzle chamber 304, the fluidic seal 305, the ink channel rim 306, the lever arm 307, the active actuator beam pair 308, the passive actuator beam pair
35 309, the active actuator anchor 310, the passive actuator anchor 311, and the ink inlet 312.

The active actuator beam pair 308 is mechanically joined to the passive actuator beam pair 309 at the join 319. Both beams pairs are anchored at their respective anchor points 310 and 311. The combination of elements 308, 309, 310, 311, and 319 form a cantilevered electrothermal bend actuator 320.
40

Figure 18 shows a small part of an array of printing elements 300, including a cross section 315 of a printing element 300.

WO 01/89838

PCT/AU00/00517

- 16 -

The cross section 315 is shown without ink, to clearly show the ink inlet 312 that passes through the silicon wafer 301.

Figures 19(a), 19(b) and 19(c) show the operating cycle of a Memjet™ printing element 300.

5 Figure 19(a) shows the quiescent position of the ink meniscus 316 prior to printing an ink droplet. Ink is retained in the nozzle chamber by surface tension at the ink meniscus 316 and at the fluidic seal 305 formed between the nozzle chamber 304 and the ink channel rim 306.

10 While printing, the printhead CMOS circuitry distributes data from the print engine controller to the correct printing element, latches the data, and buffers the data to drive the electrodes 318 of the active actuator beam pair 308. This causes an electrical current to pass through the beam pair 308 for about one microsecond, resulting in Joule heating. The temperature increase resulting from Joule heating causes the beam pair 308 to expand. As the passive actuator beam pair 309 is not heated, it does not expand, resulting in a stress difference between the two beam pairs. This stress difference is partially resolved by the cantilevered end of the electrothermal bend actuator 320 bending towards the substrate 301. The lever arm 307 transmits this movement to the nozzle chamber 304. The nozzle chamber 304 moves about two microns to the position shown in Figure 19(b). This increases the ink pressure, forcing ink 321 out of the nozzle 302, and causing the ink meniscus 316 to bulge. The nozzle rim 303 prevents the ink meniscus 316 from spreading across the surface of the nozzle chamber 304.

20 As the temperature of the beam pairs 308 and 309 equalizes, the actuator 320 returns to its original position. This aids in the break-off of the ink droplet 317 from the ink 321 in the nozzle chamber, as shown in Figure 19(c). The nozzle chamber is refilled by the action of the surface tension at the meniscus 316.

25 Figure 20 shows a segment of a printhead 350. In a netpage printer, the length of the printhead is the full width of the paper (typically 210 mm) in the direction 351. The segment shown is 0.4 mm long (about 0.2% of a complete printhead). When printing, the paper is moved past the fixed printhead in the direction 352. The printhead has 6 rows of interdigitated printing elements 300, printing the six colors or types of ink supplied by the ink inlets 312.

30 To protect the fragile surface of the printhead during operation, a nozzle guard wafer 330 is attached to the printhead substrate 301. For each nozzle 302 there is a corresponding nozzle guard hole 331 through which the ink droplets are fired. To prevent the nozzle guard holes 331 from becoming blocked by paper fibers or other debris, filtered air is pumped through the air inlets 332 and out of the nozzle guard holes during printing. To prevent ink 321 from drying, the nozzle guard is sealed while the printer is idle.

35 1.6 The Netpage Pen

40 The active sensing device of the netpage system is typically a pen 101, which, using its embedded controller 134, is able to capture and decode IR position tags from a page via an image sensor. The image sensor is a solid-state device provided with an appropriate filter to permit sensing at only near-infrared wavelengths. As described in more detail below, the system is able to sense when the nib is in contact with the surface, and the pen is able to sense tags at a sufficient rate to capture human handwriting (i.e. at 200 dpi or greater and 100 Hz or faster). Information captured by the pen is encrypted and wirelessly transmitted to the printer (or base station), the printer or base station interpreting the data with respect to

WO 01/89838

PCT/AU00/00517

- 17 -

the (known) page structure.

5 The preferred embodiment of the netpage pen operates both as a normal marking ink pen and as a non-marking stylus. The marking aspect, however, is not necessary for using the netpage system as a browsing system, such as when it is used as an Internet interface. Each netpage pen is registered with the netpage system and has a unique pen ID 61. Figure 23 shows the netpage pen class diagram, reflecting pen-related information maintained by a registration server 11 on the netpage network.

10 When either nib is in contact with a netpage, the pen determines its position and orientation relative to the page. The nib is attached to a force sensor, and the force on the nib is interpreted relative to a threshold to indicate whether the pen is "up" or "down". This allows an interactive element on the page to be "clicked" by pressing with the pen nib, in order to request, say, information from a network. Furthermore, the force is captured as a continuous value to allow, say, the full dynamics of a signature to be verified.

15 The pen determines the position and orientation of its nib on the netpage by imaging, in the infrared spectrum, an area 193 of the page in the vicinity of the nib. It decodes the nearest tag and computes the position of the nib relative to the tag from the observed perspective distortion on the imaged tag and the known geometry of the pen optics. Although the position resolution of the tag may be low, because the tag density on the page is inversely proportional to the tag size, the adjusted position resolution is quite high, exceeding the minimum resolution required for accurate handwriting recognition.

20 Pen actions relative to a netpage are captured as a series of strokes. A stroke consists of a sequence of time-stamped pen positions on the page, initiated by a pen-down event and completed by the subsequent pen-up event. A stroke is also tagged with the page ID 50 of the netpage whenever the page ID changes, which, under normal circumstances, is at the commencement of the stroke.

25 Each netpage pen has a current selection 826 associated with it, allowing the user to perform copy and paste operations etc. The selection is timestamped to allow the system to discard it after a defined time period. The current selection describes a region of a page instance. It consists of the most recent digital ink stroke captured through the pen relative to the background area of the page. It is interpreted in an application-specific manner once it is submitted to an application

30 via a selection hyperlink activation.

Each pen has a current nib 824. This is the nib last notified by the pen to the system. In the case of the default netpage pen described above, either the marking black ink nib or the non-marking stylus nib is current. Each pen also has a current nib style 825. This is the nib style last associated with the pen by an application, e.g. in response to the user selecting a color from a palette. The default nib style is the nib style associated with the current nib. Strokes captured through a pen are tagged with the current nib style. When the strokes are subsequently reproduced, they are reproduced in the nib style with which they are tagged.

40 Whenever the pen is within range of a printer with which it can communicate, the pen slowly flashes its "online" LED. When the pen fails to decode a stroke relative to the page, it momentarily activates its "error" LED. When the pen succeeds in decoding a stroke relative to the page, it momentarily activates its "ok" LED.

WO 01/89838

PCT/AU00/00517

- 18 -

A sequence of captured strokes is referred to as digital ink. Digital ink forms the basis for the digital exchange of drawings and handwriting, for online recognition of handwriting, and for online verification of signatures.

5 The pen is wireless and transmits digital ink to the netpage printer via a short-range radio link. The transmitted digital ink is encrypted for privacy and security and packetized for efficient transmission, but is always flushed on a pen-up event to ensure timely handling in the printer.

10 When the pen is out-of-range of a printer it buffers digital ink in internal memory, which has a capacity of over ten minutes of continuous handwriting. When the pen is once again within range of a printer, it transfers any buffered digital ink.

15 A pen can be registered with any number of printers, but because all state data resides in netpages both on paper and on the network, it is largely immaterial which printer a pen is communicating with at any particular time.

A preferred embodiment of the pen is described in greater detail in Section 6 below, with reference to Figures 8 to 10.

1.7 NETPAGE INTERACTION

20 The netpage printer 601 receives data relating to a stroke from the pen 101 when the pen is used to interact with a netpage. The coded data 3 of the tags 4 is read by the pen when it is used to execute a movement, such as a stroke. The data allows the identity of the particular page and associated interactive element to be determined and an indication of the relative positioning of the pen relative to the page to be obtained. The indicating data is transmitted to the printer, where it resolves, via the DNS, the page ID 50 of the stroke into the network address of the netpage page server 10 which maintains the corresponding page instance 830. It then transmits the stroke to the page server. If the page was recently identified in an earlier stroke, then the printer may already have the address of the relevant page server in its cache. Each netpage consists of a compact page layout maintained persistently by a netpage page server (see below). The page layout refers to objects such as images, fonts and pieces of text, typically stored elsewhere on the netpage network.

25 When the page server receives the stroke from the pen, it retrieves the page description to which the stroke applies, and determines which element of the page description the stroke intersects. It is then able to interpret the stroke in the context of the type of the relevant element.

30 A "click" is a stroke where the distance and time between the pen down position and the subsequent pen up position are both less than some small maximum. An object which is activated by a click typically requires a click to be activated, and accordingly, a longer stroke is ignored. The failure of a pen action, such as a "sloppy" click, to register is indicated by the lack of response from the pen's "ok" LED.

35 There are two kinds of input elements in a netpage page description: hyperlinks and form fields. Input through a form field can also trigger the activation of an associated hyperlink.

1.7.1 Hyperlinks

40 A hyperlink is a means of sending a message to a remote application, and typically elicits a printed response in the netpage system.

WO 01/89838

PCT/AU00/00517

- 19 -

5 A hyperlink element 844 identifies the application 71 which handles activation of the hyperlink, a link ID 54 which identifies the hyperlink to the application, an "alias required" flag which asks the system to include the user's application alias ID 65 in the hyperlink activation, and a description which is used when the hyperlink is recorded as a favorite or appears in the user's history. The hyperlink element class diagram is shown in Figure 29.

10 When a hyperlink is activated, the page server sends a request to an application somewhere on the network. The application is identified by an application ID 64, and the application ID is resolved in the normal way via the DNS. There are three types of hyperlinks: general hyperlinks 863, form hyperlinks 865, and selection hyperlinks 864, as shown in Figure 30. A general hyperlink can implement a request for a linked document, or may simply signal a preference to a server. A form hyperlink submits the corresponding form to the application. A selection hyperlink submits the current selection to the application. If the current selection contains a single-word piece of text, for example, the application may return a single-page document giving the word's meaning within the context in which it appears, or a translation into a different language. Each hyperlink type is characterized by what information is submitted to the application.

15 The corresponding hyperlink instance 862 records a transaction ID 55 which can be specific to the page instance on which the hyperlink instance appears. The transaction ID can identify user-specific data to the application, for example a "shopping cart" of pending purchases maintained by a purchasing application on behalf of the user.

20 The system includes the pen's current selection 826 in a selection hyperlink activation. The system includes the content of the associated form instance 868 in a form hyperlink activation, although if the hyperlink has its "submit delta" attribute set, only input since the last form submission is included. The system includes an effective return path in all hyperlink activations.

25 A hyperlinked group 866 is a group element 838 which has an associated hyperlink, as shown in Figure 31. When input occurs through any field element in the group, the hyperlink 844 associated with the group is activated. A hyperlinked group can be used to associate hyperlink behavior with a field such as a checkbox. It can also be used, in conjunction with the "submit delta" attribute of a form hyperlink, to provide continuous input to an application. It can therefore be used to support a "blackboard" interaction model, i.e. where input is captured and therefore shared as soon as it occurs.

30 1.7.2 Forms

A form defines a collection of related input fields used to capture a related set of inputs through a printed netpage. A form allows a user to submit one or more parameters to an application software program running on a server.

35 A form 867 is a group element 838 in the document hierarchy. It ultimately contains a set of terminal field elements 839. A form instance 868 represents a printed instance of a form. It consists of a set of field instances 870 which correspond to the field elements 845 of the form. Each field instance has an associated value 871, whose type depends on the type of the corresponding field element. Each field value records input through a particular printed form instance, i.e. through one or more printed netpages. The form class diagram is shown in Figure 32.

40 Each form instance has a status 872 which indicates whether the form is active, frozen, submitted, void or expired. A form is active when first printed. A form becomes frozen once it is signed. A form becomes submitted once one of its

WO 01/89838

PCT/AU00/00517

-20-

submission hyperlinks has been activated, unless the hyperlink has its "submit delta" attribute set. A form becomes void when the user invokes a void form, reset form or duplicate form page command. A form expires when the time the form has been active exceeds the form's specified lifetime. While the form is active, form input is allowed. Input through a form which is not active is instead captured in the background field 833 of the relevant page instance.

5 When the form is active or frozen, form submission is allowed. Any attempt to submit a form when the form is not active or frozen is rejected, and instead elicits an form status report.

Each form instance is associated (at 59) with any form instances derived from it, thus providing a version history. This allows all but the latest version of a form in a particular time period to be excluded from a search.

10

All input is captured as digital ink. Digital ink 873 consists of a set of timestamped stroke groups 874, each of which consists of a set of styled strokes 875. Each stroke consists of a set of timestamped pen positions 876, each of which also includes pen orientation and nib force. The digital ink class diagram is shown in Figure 33.

15 A field element 845 can be a checkbox field 877, a text field 878, a drawing field 879, or a signature field 880. The field element class diagram is shown in Figure 34. Any digital ink captured in a field's zone 58 is assigned to the field.

A checkbox field has an associated boolean value 881, as shown in Figure 35. Any mark (a tick, a cross, a stroke, a fill zigzag, etc.) captured in a checkbox field's zone causes a true value to be assigned to the field's value.

20

A text field has an associated text value 882, as shown in Figure 36. Any digital ink captured in a text field's zone is automatically converted to text via *online handwriting recognition*, and the text is assigned to the field's value. *Online handwriting recognition* is well-understood (see for example Tappert, C., C.Y. Suen and T. Wakahara, "The State of the Art in On-Line Handwriting Recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.12, No.8, August 1990).

25

A signature field has an associated digital signature value 883, as shown in Figure 37. Any digital ink captured in a signature field's zone is automatically verified with respect to the identity of the owner of the pen, and a digital signature of the content of the form of which the field is part is generated and assigned to the field's value. The digital signature is generated using the pen user's private signature key specific to the application which owns the form. *Online signature verification* is well-understood (see for example Plamondon, R. and G. Lorette, "Automatic Signature Verification and Writer Identification - The State of the Art", Pattern Recognition, Vol.22, No.2, 1989).

30

A field element is hidden if its "hidden" attribute is set. A hidden field element does not have an input zone on a page and does not accept input. It can have an associated field value which is included in the form data when the form containing the field is submitted. "Editing" commands, such as strike-throughs indicating deletion, can also be recognized in form fields.

35

Because the handwriting recognition algorithm works "online" (i.e. with access to the dynamics of the pen movement), rather than "offline" (i.e. with access only to a bitmap of pen markings), it can recognize run-on discretely-written characters with relatively high accuracy, without a writer-dependent training phase. A writer-dependent model of

40

WO 01/89838

PCT/AU00/00517

- 21 -

handwriting is automatically generated over time, however, and can be generated up-front if necessary,

5 Digital ink, as already stated, consists of a sequence of strokes. Any stroke which starts in a particular element's zone is appended to that element's digital ink stream, ready for interpretation. Any stroke not appended to an object's digital ink stream is appended to the background field's digital ink stream.

Digital ink captured in the background field is interpreted as a selection gesture. Circumscription of one or more objects is generally interpreted as a selection of the circumscribed objects, although the actual interpretation is application-specific.

10 Table 2 summarizes these various pen interactions with a netpage.

Table 2 - Summary of pen interactions with a netpage

Object	Type	Pen input	Action
Hyperlink	General	Click	Submit action to application
	Form	Click	Submit form to application
	Selection	Click	Submit selection to application
Form field	Checkbox	Any mark	Assign true to field
	Text	Handwriting	Convert digital ink to text; assign text to field
	Drawing	Digital ink	Assign digital ink to field
	Signature	Signature	Verify digital ink signature; generate digital signature of form; assign digital signature to field
None	-	Circumscription	Assign digital ink to current selection

15 The system maintains a current selection for each pen. The selection consists simply of the most recent stroke captured in the background field. The selection is cleared after an inactivity timeout to ensure predictable behavior.

20 The raw digital ink captured in every field is retained on the netpage page server and is optionally transmitted with the form data when the form is submitted to the application. This allows the application to interrogate the raw digital ink should it suspect the original conversion, such as the conversion of handwritten text. This can, for example, involve human intervention at the application level for forms that fail certain application-specific consistency checks. As an extension to this, the entire background area of a form can be designated as a drawing field. The application can then decide, on the basis of the presence of digital ink outside the explicit fields of the form, to route the form to a human operator, on the assumption that the user may have indicated amendments to the filled-in fields outside of those fields.

25 Figure 38 shows a flowchart of the process of handling pen input relative to a netpage. The process consists of receiving (at 884) a stroke from the pen; identifying (at 885) the page instance 830 to which the page ID 50 in the stroke refers; retrieving (at 886) the page description 5; identifying (at 887) a formatted element 839 whose zone 58 the stroke intersects; determining (at 888) whether the formatted element corresponds to a field element, and if so appending (at 892) the received stroke to the digital ink of the field value 871; interpreting (at 893) the accumulated digital ink of the field, and determining (at 894) whether the field is part of a hyperlinked group 866 and if so activating (at 895) the associated

30

WO 01/89838

PCT/AU00/00517

- 22 -

hyperlink; alternatively determining (at 889) whether the formatted element corresponds to a hyperlink element and if so activating (at 895) the corresponding hyperlink; alternatively, in the absence of an input field or hyperlink, appending (at 890) the received stroke to the digital ink of the background field 833; and copying (at 891) the received stroke to the current selection 826 of the current pen, as maintained by the registration server.

- 5 Figure 38a shows a detailed flowchart of step 893 in the process shown in Figure 38, where the accumulated digital ink of a field is interpreted according to the type of the field. The process consists of determining (at 896) whether the field is a checkbox and (at 897) whether the digital ink represents a checkmark, and if so assigning (at 898) a true value to the field value; alternatively determining (at 899) whether the field is a text field and if so converting (at 900) the digital ink to
10 computer text, with the help of the appropriate registration server, and assigning (at 901) the converted computer text to the field value; alternatively determining (at 902) whether the field is a signature field and if so verifying (at 903) the digital ink as the signature of the pen's owner, with the help of the appropriate registration server, creating (at 904) a digital signature of the contents of the corresponding form, also with the help of the registration server and using the pen owner's private signature key relating to the corresponding application, and assigning (at 905) the digital signature to the
15 field value.

1.7.3 Page Server Commands

A page server command is a command which is handled locally by the page server. It operates directly on form, page and document instances.

- 20 A page server command 907 can be a void form command 908, a duplicate form command 909, a reset form command 910, a get form status command 911, a duplicate page command 912, a reset page command 913, a get page status command 914, a duplicate document command 915, a reset document command 916, or a get document status command 917, as shown in Figure 39.

- 25 A void form command voids the corresponding form instance. A duplicate form command voids the corresponding form instance and then produces an active printed copy of the current form instance with field values preserved. The copy contains the same hyperlink transaction IDs as the original, and so is indistinguishable from the original to an application. A reset form command voids the corresponding form instance and then produces an active printed copy of the form
30 instance with field values discarded. A get form status command produces a printed report on the status of the corresponding form instance, including who published it, when it was printed, for whom it was printed, and the form status of the form instance.

Since a form hyperlink instance contains a transaction ID, the application has to be involved in producing a new form instance. A button requesting a new form instance is therefore typically implemented as a hyperlink.

- 35 A duplicate page command produces a printed copy of the corresponding page instance with the background field value preserved. If the page contains a form or is part of a form, then the duplicate page command is interpreted as a duplicate form command. A reset page command produces a printed copy of the corresponding page instance with the background field value discarded. If the page contains a form or is part of a form, then the reset page command is interpreted as a reset
40 form command. A get page status command produces a printed report on the status of the corresponding page instance, including who published it, when it was printed, for whom it was printed, and the status of any forms it contains or is part

WO 01/89838

PCT/AU00/00517

-23-

of.

The netpage logo which appears on every netpage is usually associated with a duplicate page element.

- 5 When a page instance is duplicated with field values preserved, field values are printed in their native form, i.e. a checkmark appears as a standard checkmark graphic, and text appears as typeset text. Only drawings and signatures appear in their original form, with a signature accompanied by a standard graphic indicating successful signature verification.
- 10 A duplicate document command produces a printed copy of the corresponding document instance with background field values preserved. If the document contains any forms, then the duplicate document command duplicates the forms in the same way a duplicate form command does. A reset document command produces a printed copy of the corresponding document instance with background field values discarded. If the document contains any forms, then the reset document command resets the forms in the same way a reset form command does. A get document status command produces a printed report on the status of the corresponding document instance, including who published it, when it was printed, for whom it was printed, and the status of any forms it contains.
- 15 If the page server command's "on selected" attribute is set, then the command operates on the page identified by the pen's current selection rather than on the page containing the command. This allows a menu of page server commands to be printed. If the target page doesn't contain a page server command element for the designated page server command, then the command is ignored.
- 20 An application can provide application-specific handling by embedding the relevant page server command element in a hyperlinked group. The page server activates the hyperlink associated with the hyperlinked group rather than executing the page server command.
- 25 A page server command element is hidden if its "hidden" attribute is set. A hidden command element does not have an input zone on a page and so cannot be activated directly by a user. It can, however, be activated via a page server command embedded in a different page, if that page server command has its "on selected" attribute set.
- 1.8 STANDARD FEATURES OF NETPAGES**
- 30 In the preferred form, each netpage is printed with the netpage logo at the bottom to indicate that it is a netpage and therefore has interactive properties. The logo also acts as a copy button. In most cases pressing the logo produces a copy of the page. In the case of a form, the button produces a copy of the entire form. And in the case of a secure document, such as a ticket or coupon, the button elicits an explanatory note or advertising page.
- 35 The default single-page copy function is handled directly by the relevant netpage page server. Special copy functions are handled by linking the logo button to an application.
- 1.9 USER HELP SYSTEM**
- In a preferred embodiment, the netpage printer has a single button labelled "Help". When pressed it elicits a single page of information, including:
- 40
- status of printer connection
 - status of printer consumables

WO 01/89838

PCT/AU00/00517

- 24 -

- top-level help menu
- document function menu
- top-level netpage network directory

The help menu provides a hierarchical manual on how to use the netpage system.

5 The document function menu includes the following functions:

- print a copy of a document
- print a clean copy of a form
- print the status of a document

10 A document function is initiated by simply pressing the button and then touching any page of the document. The status of a document indicates who published it and when, to whom it was delivered, and to whom and when it was subsequently submitted as a form.

The netpage network directory allows the user to navigate the hierarchy of publications and services on the network. As an alternative, the user can call the netpage network "900" number "yellow pages" and speak to a human operator. The operator can locate the desired document and route it to the user's printer. Depending on the document type, the publisher or the user pays the small "yellow pages" service fee.

15

The help page is obviously unavailable if the printer is unable to print. In this case the "error" light is lit and the user can request remote diagnosis over the network.

2 Personalized Publication Model

20 In the following description, news is used as a canonical publication example to illustrate personalization mechanisms in the netpage system. Although news is often used in the limited sense of newspaper and newsmagazine news, the intended scope in the present context is wider.

25 In the netpage system, the editorial content and the advertising content of a news publication are personalized using different mechanisms. The editorial content is personalized according to the reader's explicitly stated and implicitly captured interest profile. The advertising content is personalized according to the reader's locality and demographic.

2.1 EDITORIAL PERSONALIZATION

30 A subscriber can draw on two kinds of news sources: those that deliver news publications, and those that deliver news streams. While news publications are aggregated and edited by the publisher, news streams are aggregated either by a news publisher or by a specialized news aggregator. News publications typically correspond to traditional newspapers and newsmagazines, while news streams can be many and varied: a "raw" news feed from a news service, a cartoon strip, a freelance writer's column, a friend's bulletin board, or the reader's own e-mail.

35 The netpage publication server supports the publication of edited news publications as well as the aggregation of multiple news streams. By handling the aggregation and hence the formatting of news streams selected directly by the reader, the server is able to place advertising on pages over which it otherwise has no editorial control.

40 The subscriber builds a daily newspaper by selecting one or more contributing news publications, and creating a personalized version of each. The resulting daily editions are printed and bound together into a single newspaper. The various members of a household typically express their different interests and tastes by selecting different daily publications and then customizing them.

WO 01/89838

PCT/AU00/00517

- 25 -

- For each publication, the reader optionally selects specific sections. Some sections appear daily, while others appear weekly. The daily sections available from The New York Times online, for example, include "Page One Plus", "National", "International", "Opinion", "Business", "Arts/Living", "Technology", and "Sports". The set of available sections is specific to a publication, as is the default subset.
- 5 The reader can extend the daily newspaper by creating custom sections, each one drawing on any number of news streams. Custom sections might be created for e-mail and friends' announcements ("Personal"), or for monitoring news feeds for specific topics ("Alerts" or "Clippings").
- 10 For each section, the reader optionally specifies its size, either qualitatively (e.g. short, medium, or long), or numerically (i.e. as a limit on its number of pages), and the desired proportion of advertising, either qualitatively (e.g. high, normal, low, none), or numerically (i.e. as a percentage).
- 15 The reader also optionally expresses a preference for a large number of shorter articles or a small number of longer articles. Each article is ideally written (or edited) in both short and long forms to support this preference.
- An article may also be written (or edited) in different versions to match the expected sophistication of the reader, for example to provide children's and adults' versions. The appropriate version is selected according to the reader's age. The reader can specify a "reading age" which takes precedence over their biological age.
- 20 The articles which make up each section are selected and prioritized by the editors, and each is assigned a useful lifetime. By default they are delivered to all relevant subscribers, in priority order, subject to space constraints in the subscribers' editions.
- 25 In sections where it is appropriate, the reader may optionally enable collaborative filtering. This is then applied to articles which have a sufficiently long lifetime. Each article which qualifies for collaborative filtering is printed with rating buttons at the end of the article. The buttons can provide an easy choice (e.g. "liked" and "disliked"), making it more likely that readers will bother to rate the article.
- 30 Articles with high priorities and short lifetimes are therefore effectively considered essential reading by the editors and are delivered to most relevant subscribers.
- The reader optionally specifies a serendipity factor, either qualitatively (e.g. do or don't surprise me), or numerically. A high serendipity factor lowers the threshold used for matching during collaborative filtering. A high factor makes it more likely that the corresponding section will be filled to the reader's specified capacity. A different serendipity factor can be specified for different days of the week.
- 35 The reader also optionally specifies topics of particular interest within a section, and this modifies the priorities assigned by the editors.
- 40 The speed of the reader's Internet connection affects the quality at which images can be delivered. The reader optionally specifies a preference for fewer images or smaller images or both. If the number or size of images is not reduced, then

WO 01/89838

PCT/AU00/00517

- 28 -

images may be delivered at lower quality (i.e. at lower resolution or with greater compression).

5 At a global level, the reader specifies how quantities, dates, times and monetary values are localized. This involves specifying whether units are imperial or metric, a local timezone and time format, and a local currency, and whether the localization consist of *in situ* translation or annotation. These preferences are derived from the reader's locality by default.

To reduce reading difficulties caused by poor eyesight, the reader optionally specifies a global preference for a larger presentation. Both text and images are scaled accordingly, and less information is accommodated on each page.

10 The language in which a news publication is published, and its corresponding text encoding, is a property of the publication and not a preference expressed by the user. However, the netpage system can be configured to provide automatic translation services in various guises.

2.2 ADVERTISING LOCALIZATION AND TARGETING

15 The personalization of the editorial content directly affects the advertising content, because advertising is typically placed to exploit the editorial context. Travel ads, for example, are more likely to appear in a travel section than elsewhere. The value of the editorial content to an advertiser (and therefore to the publisher) lies in its ability to attract large numbers of readers with the right demographics.

20 Effective advertising is placed on the basis of locality and demographics. Locality determines proximity to particular services, retailers etc., and particular interests and concerns associated with the local community and environment. Demographics determine general interests and preoccupations as well as likely spending patterns.

25 A news publisher's most profitable product is advertising "space", a multi-dimensional entity determined by the publication's geographic coverage, the size of its readership, its readership demographics, and the page area available for advertising.

30 In the netpage system, the netpage publication server computes the approximate multi-dimensional size of a publication's saleable advertising space on a per-section basis, taking into account the publication's geographic coverage, the section's readership, the size of each reader's section edition, each reader's advertising proportion, and each reader's demographic.

In comparison with other media, the netpage system allows the advertising space to be defined in greater detail, and allows smaller pieces of it to be sold separately. It therefore allows it to be sold at closer to its true value.

35 For example, the same advertising "slot" can be sold in varying proportions to several advertisers, with individual readers' pages randomly receiving the advertisement of one advertiser or another, overall preserving the proportion of space sold to each advertiser.

The netpage system allows advertising to be linked directly to detailed product information and online purchasing. It therefore raises the intrinsic value of the advertising space.

40 Because personalization and localization are handled automatically by netpage publication servers, an advertising aggregator can provide arbitrarily broad coverage of both geography and demographics. The subsequent disaggregation is

WO 01/89838

PCT/AU00/00517

- 27 -

efficient because it is automatic. This makes it more cost-effective for publishers to deal with advertising aggregators than to directly capture advertising. Even though the advertising aggregator is taking a proportion of advertising revenue, publishers may find the change profit-neutral because of the greater efficiency of aggregation. The advertising aggregator acts as an intermediary between advertisers and publishers, and may place the same advertisement in multiple publications.

It is worth noting that ad placement in a netpage publication can be more complex than ad placement in the publication's traditional counterpart, because the publication's advertising space is more complex. While ignoring the full complexities of negotiations between advertisers, advertising aggregators and publishers, the preferred form of the netpage system provides some automated support for these negotiations, including support for automated auctions of advertising space. Automation is particularly desirable for the placement of advertisements which generate small amounts of income, such as small or highly localized advertisements.

Once placement has been negotiated, the aggregator captures and edits the advertisement and records it on a netpage ad server. Correspondingly, the publisher records the ad placement on the relevant netpage publication server. When the netpage publication server lays out each user's personalized publication, it picks the relevant advertisements from the netpage ad server.

2.3 USER PROFILES

2.3.1 Information Filtering

The personalization of news and other publications relies on an assortment of user-specific profile information, including:

- publication customizations
- collaborative filtering vectors
- contact details
- presentation preferences

The customization of a publication is typically publication-specific, and so the customization information is maintained by the relevant netpage publication server.

A collaborative filtering vector consists of the user's ratings of a number of news items. It is used to correlate different users' interests for the purposes of making recommendations. Although there are benefits to maintaining a single collaborative filtering vector independently of any particular publication, there are two reasons why it is more practical to maintain a separate vector for each publication: there is likely to be more overlap between the vectors of subscribers to the same publication than between those of subscribers to different publications; and a publication is likely to want to present its users' collaborative filtering vectors as part of the value of its brand, not to be found elsewhere. Collaborative filtering vectors are therefore also maintained by the relevant netpage publication server.

Contact details, including name, street address, ZIP Code, state, country, telephone numbers, are global by nature, and are maintained by a netpage registration server.

Presentation preferences, including those for quantities, dates and times, are likewise global and maintained in the same way.

WO 01/89838

PCT/AU00/00517

- 28 -

The localization of advertising relies on the locality indicated in the user's contact details, while the targeting of advertising relies on personal information such as date of birth, gender, marital status, income, profession, education, or qualitative derivatives such as age range and income range.

5 For those users who choose to reveal personal information for advertising purposes, the information is maintained by the relevant netpage registration server. In the absence of such information, advertising can be targeted on the basis of the demographic associated with the user's ZIP or ZIP+4 Code.

10 Each user, pen, printer, application provider and application is assigned its own unique identifier, and the netpage registration server maintains the relationships between them, as shown in Figures 21, 22, 23 and 24. For registration purposes, a publisher is a special kind of application provider, and a publication is a special kind of application.

15 Each user 800 may be authorized to use any number of printers 802, and each printer may allow any number of users to use it. Each user has a single default printer (at 66), to which periodical publications are delivered by default, whilst pages printed on demand are delivered to the printer through which the user is interacting. The server keeps track of which publishers a user has authorized to print to the user's default printer. A publisher does not record the ID of any particular printer, but instead resolves the ID when it is required.

20 When a user subscribes 808 to a publication 807, the publisher 806 (i.e. application provider 803) is authorized to print to a specified printer or the user's default printer. This authorization can be revoked at any time by the user. Each user may have several pens 801, but a pen is specific to a single user. If a user is authorized to use a particular printer, then that printer recognizes any of the user's pens.

25 The pen ID is used to locate the corresponding user profile maintained by a particular netpage registration server, via the DNS in the usual way.

A Web terminal 809 can be authorized to print on a particular netpage printer, allowing Web pages and netpage documents encountered during Web browsing to be conveniently printed on the nearest netpage printer.

30 The netpage system can collect, on behalf of a printer provider, fees and commissions on income earned through publications printed on the provider's printers. Such income can include advertising fees, click-through fees, e-commerce commissions, and transaction fees. If the printer is owned by the user, then the user is the printer provider.

35 Each user also has a netpage account 820 which is used to accumulate micro-debits and credits (such as those described in the preceding paragraph); contact details 815, including name, address and telephone numbers; global preferences 816, including privacy, delivery and localization settings; any number of biometric records 817, containing the user's encoded signature 818, fingerprint 819 etc; a handwriting model 819 automatically maintained by the system; and SET payment card accounts 821 with which e-commerce payments can be made.

2.3.2 Favorites List

40 A netpage user can maintain a list 922 of "favorites" - links to useful documents etc. on the netpage network. The list is maintained by the system on the user's behalf. It is organized as a hierarchy of folders 924, a preferred embodiment of

WO 01/89838

PCT/AU00/00517

- 29 -

which is shown in the class diagram in Figure 41.

2.3.3 History List

5 The system maintains a history list 929 on each user's behalf, containing links to documents etc. accessed by the user through the netpage system. It is organized as a date-ordered list, a preferred embodiment of which is shown in the class diagram in Figure 42.

2.4 INTELLIGENT PAGE LAYOUT

10 The netpage publication server automatically lays out the pages of each user's personalized publication on a section-by-section basis. Since most advertisements are in the form of pre-formatted rectangles, they are placed on the page before the editorial content.

15 The advertising ratio for a section can be achieved with wildly varying advertising ratios on individual pages within the section, and the ad layout algorithm exploits this. The algorithm is configured to attempt to co-locate closely tied editorial and advertising content, such as placing ads for roofing material specifically within the publication because of a special feature on do-it-yourself roofing repairs.

The editorial content selected for the user, including text and associated images and graphics, is then laid out according to various aesthetic rules.

20 The entire process, including the selection of ads and the selection of editorial content, must be iterated once the layout has converged, to attempt to more closely achieve the user's stated section size preference. The section size preference can, however, be matched on average over time, allowing significant day-to-day variations.

2.5 DOCUMENT FORMAT

Once the document is laid out, it is encoded for efficient distribution and persistent storage on the netpage network.

25 The primary efficiency mechanism is the separation of information specific to a single user's edition and information shared between multiple users' editions. The specific information consists of the page layout. The shared information consists of the objects to which the page layout refers, including images, graphics, and pieces of text.

30 A text object contains fully-formatted text represented in the Extensible Markup Language (XML) using the Extensible Stylesheet Language (XSL). XSL provides precise control over text formatting independently of the region into which the text is being set, which in this case is being provided by the layout. The text object contains embedded language codes to enable automatic translation, and embedded hyphenation hints to aid with paragraph formatting.

35 An image object encodes an image in the JPEG 2000 wavelet-based compressed image format. A graphic object encodes a 2D graphic in Scalable Vector Graphics (SVG) format.

40 The layout itself consists of a series of placed image and graphic objects, linked textflow objects through which text objects flow, hyperlinks and input fields as described above, and watermark regions. These layout objects are summarized in Table 3. The layout uses a compact format suitable for efficient distribution and storage.

Table 3 - netpage layout objects

WO 01/89838

PCT/AU00/00517

- 30 -

Layout object	Attribute	Format of linked object
Image	Position	-
	Image object ID	JPEG 2000
Graphic	Position	-
	Graphic object ID	SVG
Textflow	Textflow ID	-
	Zone	-
	Optional text object ID	XML/XSL
Hyperlink	Type	-
	Zone	-
	Application ID, etc.	-
Field	Type	-
	Meaning	-
	Zone	-
Watermark	Zone	-

2.6 DOCUMENT DISTRIBUTION

As described above, for purposes of efficient distribution and persistent storage on the netpage network, a user-specific page layout is separated from the shared objects to which it refers.

5

When a subscribed publication is ready to be distributed, the netpage publication server allocates, with the help of the netpage ID server 12, a unique ID for each page, page instance, document, and document instance.

10

The server computes a set of optimized subsets of the shared content and creates a multicast channel for each subset, and then tags each user-specific layout with the names of the multicast channels which will carry the shared content used by that layout. The server then pointcasts each user's layouts to that user's printer via the appropriate page server, and when the pointcasting is complete, multicasts the shared content on the specified channels. After receiving its pointcast, each page server and printer subscribes to the multicast channels specified in the page layouts. During the multicasts, each page server and printer extracts from the multicast streams those objects referred to by its page layouts. The page servers

15

persistently archive the received page layouts and shared content.

Once a printer has received all the objects to which its page layouts refer, the printer re-creates the fully-populated layout and then rasterizes and prints it.

20

Under normal circumstances, the printer prints pages faster than they can be delivered. Assuming a quarter of each page is covered with images, the average page has a size of less than 400KB. The printer can therefore hold in excess of 100 such pages in its internal 64MB memory, allowing for temporary buffers etc. The printer prints at a rate of one page per second. This is equivalent to 400KB or about 3Mbit of page data per second, which is similar to the highest expected rate of page data delivery over a broadband network.

25

WO 01/89838

PCT/AU00/00517

- 31 -

Even under abnormal circumstances, such as when the printer runs out of paper, it is likely that the user will be able to replenish the paper supply before the printer's 100-page internal storage capacity is exhausted.

5 However, if the printer's internal memory does fill up, then the printer will be unable to make use of a multicast when it first occurs. The netpage publication server therefore allows printers to submit requests for re-multicasts. When a critical number of requests is received or a timeout occurs, the server re-multicasts the corresponding shared objects.

Once a document is printed, a printer can produce an exact duplicate at any time by retrieving its page layouts and contents from the relevant page server.

2.7 ON-DEMAND DOCUMENTS

10 When a netpage document is requested on demand, it can be personalized and delivered in much the same way as a periodical. However, since there is no shared content, delivery is made directly to the requesting printer without the use of multicast.

15 When a non-netpage document is requested on demand, it is not personalized, and it is delivered via a designated netpage formatting server which reformats it as a netpage document. A netpage formatting server is a special instance of a netpage publication server. The netpage formatting server has knowledge of various Internet document formats, including Adobe's Portable Document Format (PDF), and Hypertext Markup Language (HTML). In the case of HTML, it can make use of the higher resolution of the printed page to present Web pages in a multi-column format, with a table of contents. It can automatically include all Web pages directly linked to the requested page. The user can tune this behavior via a preference.

20 The netpage formatting server makes standard netpage behavior, including interactivity and persistence, available on any Internet document, no matter what its origin and format. It hides knowledge of different document formats from both the netpage printer and the netpage page server, and hides knowledge of the netpage system from Web servers.

25 3 Security

3.1 CRYPTOGRAPHY

Cryptography is used to protect sensitive information, both in storage and in transit, and to authenticate parties to a transaction. There are two classes of cryptography in widespread use: secret-key cryptography and public-key cryptography. The netpage network uses both classes of cryptography.

30 Secret-key cryptography, also referred to as symmetric cryptography, uses the same key to encrypt and decrypt a message. Two parties wishing to exchange messages must first arrange to securely exchange the secret key.

35 Public-key cryptography, also referred to as asymmetric cryptography, uses two encryption keys. The two keys are mathematically related in such a way that any message encrypted using one key can only be decrypted using the other key. One of these keys is then published, while the other is kept private. The public key is used to encrypt any message intended for the holder of the private key. Once encrypted using the public key, a message can only be decrypted using the private key. Thus two parties can securely exchange messages without first having to exchange a secret key. To ensure that the private key is secure, it is normal for the holder of the private key to generate the key pair.

40 Public-key cryptography can be used to create a digital signature. The holder of the private key can create a known hash

WO 01/89838

PCT/AU00/00517

- 32 -

of a message and then encrypt the hash using the private key. Anyone can then verify that the encrypted hash constitutes the "signature" of the holder of the private key with respect to that particular message by decrypting the encrypted hash using the public key and verifying the hash against the message. If the signature is appended to the message, then the recipient of the message can verify both that the message is genuine and that it has not been altered in transit.

5

To make public-key cryptography work, there has to be a way to distribute public keys which prevents impersonation. This is normally done using certificates and certificate authorities. A certificate authority is a trusted third party which authenticates the connection between a public key and someone's identity. The certificate authority verifies the person's identity by examining identity documents, and then creates and signs a digital certificate containing the person's identity details and public key. Anyone who trusts the certificate authority can use the public key in the certificate with a high degree of certainty that it is genuine. They just have to verify that the certificate has indeed been signed by the certificate authority, whose public key is well-known.

10

In most transaction environments, public-key cryptography is only used to create digital signatures and to securely exchange secret session keys. Secret-key cryptography is used for all other purposes.

15

In the following discussion, when reference is made to the *secure* transmission of information between a netpage printer and a server, what actually happens is that the printer obtains the server's certificate, authenticates it with reference to the certificate authority, uses the public key-exchange key in the certificate to exchange a secret session key with the server, and then uses the secret session key to encrypt the message data. A *session* key, by definition, can have an arbitrarily short lifetime.

20

3.2 NETPAGE PRINTER SECURITY

Each netpage printer is assigned a pair of unique identifiers at time of manufacture which are stored in read-only memory in the printer and in the netpage registration server database. The first ID is public and uniquely identifies the printer on the netpage network. The second ID is secret and is used when the printer is first registered on the network.

25

When the printer connects to the netpage network for the first time after installation, it creates a signature public/private key pair. It transmits the secret ID and the public key securely to the netpage registration server. The server compares the secret ID against the printer's secret ID recorded in its database, and accepts the registration if the IDs match. It then creates and signs a certificate containing the printer's public ID and public signature key, and stores the certificate in the registration database.

30

The netpage registration server acts as a certificate authority for netpage printers, since it has access to secret information allowing it to verify printer identity.

35

When a user subscribes to a publication, a record is created in the netpage registration server database authorizing the publisher to print the publication to the user's default printer or a specified printer. Every document sent to a printer via a page server is addressed to a particular user and is signed by the publisher using the publisher's private signature key. The page server verifies, via the registration database, that the publisher is authorized to deliver the publication to the specified user. The page server verifies the signature using the publisher's public key, obtained from the publisher's certificate stored in the registration database.

40

WO 01/89838

PCT/AU00/00517

- 33 -

The netpage registration server accepts requests to add printing authorizations to the database, so long as those requests are initiated via a pen registered to the printer.

3.3 NETPAGE PEN SECURITY

5 Each netpage pen is assigned a unique identifier at time of manufacture which is stored in read-only memory in the pen and in the netpage registration server database. The pen ID 61 uniquely identifies the pen on the netpage network.

10 A netpage pen can "know" a number of netpage printers, and a printer can "know" a number of pens. A pen communicates with a printer via a radio frequency signal whenever it is within range of the printer. Once a pen and printer are registered, they regularly exchange session keys. Whenever the pen transmits digital ink to the printer, the digital ink is always encrypted using the appropriate session key. Digital ink is never transmitted in the clear.

15 A pen stores a session key for every printer it knows, indexed by printer ID, and a printer stores a session key for every pen it knows, indexed by pen ID. Both have a large but finite storage capacity for session keys, and will forget a session key on a least-recently-used basis if necessary.

20 When a pen comes within range of a printer, the pen and printer discover whether they know each other. If they don't know each other, then the printer determines whether it is supposed to know the pen. This might be, for example, because the pen belongs to a user who is registered to use the printer. If the printer is meant to know the pen but doesn't, then it initiates the automatic pen registration procedure. If the printer isn't meant to know the pen, then it agrees with the pen to ignore it until the pen is placed in a charging cup, at which time it initiates the registration procedure.

25 In addition to its public ID, the pen contains a secret key-exchange key. The key-exchange key is also recorded in the netpage registration server database at time of manufacture. During registration, the pen transmits its pen ID to the printer, and the printer transmits the pen ID to the netpage registration server. The server generates a session key for the printer and pen to use, and securely transmits the session key to the printer. It also transmits a copy of the session key encrypted with the pen's key-exchange key. The printer stores the session key internally, indexed by the pen ID, and transmits the encrypted session key to the pen. The pen stores the session key internally, indexed by the printer ID.

30 Although a fake pen can impersonate a pen in the pen registration protocol, only a real pen can decrypt the session key transmitted by the printer.

35 When a previously unregistered pen is first registered, it is of limited use until it is linked to a user. A registered but "un-owned" pen is only allowed to be used to request and fill in netpage user and pen registration forms, to register a new user to which the new pen is automatically linked, or to add a new pen to an existing user.

The pen uses secret-key rather than public-key encryption because of hardware performance constraints in the pen.

3.4 SECURE DOCUMENTS

40 The netpage system supports the delivery of secure documents such as tickets and coupons. The netpage printer includes a facility to print watermarks, but will only do so on request from publishers who are suitably authorized. The publisher indicates its authority to print watermarks in its certificate, which the printer is able to authenticate.

WO 01/89838

PCT/AU00/00517

- 34 -

- The "watermark" printing process uses an alternative dither matrix in specified "watermark" regions of the page. Back-to-back pages contain mirror-image watermark regions which coincide when printed. The dither matrices used in odd and even pages' watermark regions are designed to produce an interference effect when the regions are viewed together, achieved by looking *through* the printed sheet.
- 5 The effect is similar to a watermark in that it is not visible when looking at only one side of the page, and is lost when the page is copied by normal means.
- 10 Pages of secure documents cannot be copied using the built-in netpage copy mechanism described in Section 1.9 above. This extends to copying netpages on netpage-aware photocopiers.
- 15 Secure documents are typically generated as part of e-commerce transactions. They can therefore include the user's photograph which was captured when the user registered biometric information with the netpage registration server, as described in Section 2.
- 20 When presented with a secure netpage document, the recipient can verify its authenticity by requesting its status in the usual way. The unique ID of a secure document is only valid for the lifetime of the document, and secure document IDs are allocated non-contiguously to prevent their prediction by opportunistic forgers. A secure document verification pen can be developed with built-in feedback on verification failure, to support easy point-of-presentation document verification.
- Clearly neither the watermark nor the user's photograph are secure in a cryptographic sense. They simply provide a significant obstacle to casual forgery. Online document verification, particularly using a verification pen, provides an added level of security where it is needed, but is still not entirely immune to forgeries.
- 25 **3.5 NON-REPUDIATION**
- In the netpage system, forms submitted by users are delivered reliably to forms handlers and are persistently archived on netpage page servers. It is therefore impossible for recipients to repudiate delivery.
- E-commerce payments made through the system, as described in Section 4, are also impossible for the payee to repudiate.
- 30 **4 ELECTRONIC COMMERCE MODEL**
- 4.1 SECURE ELECTRONIC TRANSACTION (SET)**
- The netpage system uses the Secure Electronic Transaction (SET) system as one of its payment systems. SET, having been developed by MasterCard and Visa, is organized around payment cards, and this is reflected in the terminology. However, much of the system is independent of the type of accounts being used.
- 35 In SET, cardholders and merchants register with a certificate authority and are issued with certificates containing their public signature keys. The certificate authority verifies a cardholder's registration details with the card issuer as appropriate, and verifies a merchant's registration details with the acquirer as appropriate. Cardholders and merchants store their respective private signature keys securely on their computers. During the payment process, these certificates are used to mutually authenticate a merchant and cardholder, and to authenticate them both to the payment gateway.
- 40 SET has not yet been adopted widely, partly because cardholder maintenance of keys and certificates is considered burdensome. Interim solutions which maintain cardholder keys and certificates on a server and give the cardholder access

WO 01/89838

PCT/AU00/00517

- 35 -

via a password have met with some success.

4.2 SET PAYMENTS

In the netpage system the netpage registration server acts as a proxy for the netpage user (i.e. the cardholder) in SET payment transactions.

5

The netpage system uses biometrics to authenticate the user and authorize SET payments. Because the system is pen-based, the biometric used is the user's on-line signature, consisting of time-varying pen position and pressure. A fingerprint biometric can also be used by designing a fingerprint sensor into the pen, although at a higher cost. The type of biometric used only affects the capture of the biometric, not the authorization aspects of the system.

10

The first step to being able to make SET payments is to register the user's biometric with the netpage registration server. This is done in a controlled environment, for example a bank, where the biometric can be captured at the same time as the user's identity is verified. The biometric is captured and stored in the registration database, linked to the user's record. The user's photograph is also optionally captured and linked to the record. The SET cardholder registration process is completed, and the resulting private signature key and certificate are stored in the database. The user's payment card information is also stored, giving the netpage registration server enough information to act as the user's proxy in any SET payment transaction.

15

20

When the user eventually supplies the biometric to complete a payment, for example by signing a netpage order form, the printer securely transmits the order information, the pen ID and the biometric data to the netpage registration server. The server verifies the biometric with respect to the user identified by the pen ID, and from then on acts as the user's proxy in completing the SET payment transaction.

4.3 MICRO-PAYMENTS

The netpage system includes a mechanism for micro-payments, to allow the user to be conveniently charged for printing low-cost documents on demand and for copying copyright documents, and possibly also to allow the user to be reimbursed for expenses incurred in printing advertising material. The latter depends on the level of subsidy already provided to the user.

25

30

When the user registers for e-commerce, a network account is established which aggregates micro-payments. The user receives a statement on a regular basis, and can settle any outstanding debit balance using the standard payment mechanism.

The network account can be extended to aggregate subscription fees for periodicals, which would also otherwise be presented to the user in the form of individual statements.

4.4 TRANSACTIONS

35

When a user requests a netpage in a particular application context, the application is able to embed a user-specific transaction ID 55 in the page. Subsequent input through the page is tagged with the transaction ID, and the application is thereby able to establish an appropriate context for the user's input.

40

When input occurs through a page which is not user-specific, however, the application must use the user's unique identity to establish a context. A typical example involves adding items from a pre-printed catalog page to the user's virtual "shopping cart". To protect the user's privacy, however, the unique user ID 60 known to the netpage system is not

WO 01/89838

PCT/AU00/00517

- 36 -

divulged to applications. This is to prevent different application providers from easily correlating independently accumulated behavioral data.

5 The netpage registration server instead maintains an anonymous relationship between a user and an application via a unique alias ID 65, as shown in Figure 24. Whenever the user activates a hyperlink tagged with the "registered" attribute, the netpage page server asks the netpage registration server to translate the associated application ID 64, together with the pen ID 61, into an alias ID 65. The alias ID is then submitted to the hyperlink's application.

10 The application maintains state information indexed by alias ID, and is able to retrieve user-specific state information without knowledge of the global identity of the user.

The system also maintains an independent certificate and private signature key for each of a user's applications, to allow it to sign application transactions on behalf of the user using only application-specific information.

15 To assist the system in routing product bar code (UPC) "hyperlink" activations, the system records a favorite application on behalf of the user for any number of product types.

Each application is associated with an application provider, and the system maintains an account on behalf of each application provider, to allow it to credit and debit the provider for click-through fees etc.

20 An application provider can be a publisher of periodical subscribed content. The system records the user's willingness to receive the subscribed publication, as well as the expected frequency of publication.

4.5 RESOURCE DESCRIPTIONS AND COPYRIGHT

A preferred embodiment of a resource description class diagram is shown in Figure 40.

25 Each document and content object may be described by one or more resource descriptions 842. Resource descriptions use the Dublin Core metadata element set, which is designed to facilitate discovery of electronic resources. Dublin Core metadata conforms to the World Wide Web Consortium (W3C) Resource Description Framework (RDF).

A resource description may identify rights holders 920. The netpage system automatically transfers copyright fees from users to rights holders when users print copyright content.

5 Communications Protocols

30 A communications protocol defines an ordered exchange of messages between entities. In the netpage system, entities such as pens, printers and servers utilise a set of defined protocols to cooperatively handle user interaction with the netpage system.

35 Each protocol is illustrated by way of a sequence diagram in which the horizontal dimension is used to represent message flow and the vertical dimension is used to represent time. Each entity is represented by a rectangle containing the name of the entity and a vertical column representing the lifeline of the entity. During the time an entity exists, the lifeline is shown as a dashed line. During the time an entity is active, the lifeline is shown as a double line. Because the protocols considered here do not create or destroy entities, lifelines are generally cut short as soon as an entity ceases to participate in a protocol.

40 5.1 SUBSCRIPTION DELIVERY PROTOCOL

A preferred embodiment of a subscription delivery protocol is shown in Figure 43.

WO 01/89838

PCT/AU00/00517

- 37 -

5 A large number of users may subscribe to a periodical publication. Each user's edition may be laid out differently, but many users' editions will share common content such as text objects and image objects. The subscription delivery protocol therefore delivers document structures to individual printers via pointcast, but delivers shared content objects via multicast.

10 The application (i.e. publisher) first obtains a document ID 51 for each document from an ID server 12. It then sends each document structure, including its document ID and page descriptions, to the page server 10 responsible for the document's newly allocated ID. It includes its own application ID 64, the subscriber's alias ID 65, and the relevant set of multicast channel names. It signs the message using its private signature key.

15 The page server uses the application ID and alias ID to obtain from the registration server the corresponding user ID 60, the user's selected printer ID 62 (which may be explicitly selected for the application, or may be the user's default printer), and the application's certificate.

The application's certificate allows the page server to verify the message signature. The page server's request to the registration server fails if the application ID and alias ID don't together identify a subscription 808.

20 The page server then allocates document and page instance IDs and forwards the page descriptions, including page IDs 50, to the printer. It includes the relevant set of multicast channel names for the printer to listen to.

It then returns the newly allocated page IDs to the application for future reference.

25 Once the application has distributed all of the document structures to the subscribers' selected printers via the relevant page servers, it multicasts the various subsets of the shared objects on the previously selected multicast channels. Both page servers and printers monitor the appropriate multicast channels and receive their required content objects. They are then able to populate the previously pointcast document structures. This allows the page servers to add complete documents to their databases, and it allows the printers to print the documents.

5.2 HYPERLINK ACTIVATION PROTOCOL

30 A preferred embodiment of a hyperlink activation protocol is shown in Figure 45.

35 When a user clicks on a netpage with a netpage pen, the pen communicates the click to the nearest netpage printer 601. The click identifies the page and a location on the page. The printer already knows the ID 61 of the pen from the pen connection protocol.

The printer determines, via the DNS, the network address of the page server 10a handling the particular page ID 50. The address may already be in its cache if the user has recently interacted with the same page. The printer then forwards the pen ID, its own printer ID 62, the page ID and click location to the page server.

40 The page server loads the page description 5 identified by the page ID and determines which input element's zone 58, if any, the click lies in. Assuming the relevant input element is a hyperlink element 844, the page server then obtains the

WO 01/89838

PCT/AU00/00517

- 38 -

- associated application ID 64 and link ID 54, and determines, via the DNS, the network address of the application server hosting the application 71.
- 5 The page server uses the pen ID 61 to obtain the corresponding user ID 60 from the registration server 11, and then allocates a globally unique hyperlink request ID 52 and builds a hyperlink request 934. The hyperlink request class diagram is shown in Figure 44. The hyperlink request records the IDs of the requesting user and printer, and identifies the clicked hyperlink instance 862. The page server then sends its own server ID 53, the hyperlink request ID, and the link ID to the application.
- 10 The application produces a response document according to application-specific logic, and obtains a document ID 51 from an ID server 12. It then sends the document to the page server 10b responsible for the document's newly allocated ID, together with the requesting page server's ID and the hyperlink request ID.
- 15 The second page server sends the hyperlink request ID and application ID to the first page server to obtain the corresponding user ID and printer ID 62. The first page server rejects the request if the hyperlink request has expired or is for a different application.
- The second page server allocates document instance and page IDs 50, returns the newly allocated page IDs to the application, adds the complete document to its own database, and finally sends the page descriptions to the requesting printer.
- 20 The hyperlink instance may include a meaningful transaction ID 55, in which case the first page server includes the transaction ID in the message sent to the application. This allows the application to establish a transaction-specific context for the hyperlink activation.
- 25 If the hyperlink requires a user alias, i.e. its "alias required" attribute is set, then the first page server sends both the pen ID 61 and the hyperlink's application ID 64 to the registration server 11 to obtain not just the user ID corresponding to the pen ID but also the alias ID 65 corresponding to the application ID and the user ID. It includes the alias ID in the message sent to the application, allowing the application to establish a user-specific context for the hyperlink activation.
- 30 **5.3 HANDWRITING RECOGNITION PROTOCOL**
- When a user draws a stroke on a netpage with a netpage pen, the pen communicates the stroke to the nearest netpage printer. The stroke identifies the page and a path on the page.
- 35 The printer forwards the pen ID 61, its own printer ID 62, the page ID 50 and stroke path to the page server 10 in the usual way.
- The page server loads the page description 5 identified by the page ID and determines which input element's zone 58, if any, the stroke intersects. Assuming the relevant input element is a text field 878, the page server appends the stroke to the text field's digital ink.
- 40 After a period of inactivity in the zone of the text field, the page server sends the pen ID and the pending strokes to the

WO 01/89838

PCT/AU00/00517

- 39 -

registration server 11 for interpretation. The registration server identifies the user corresponding to the pen, and uses the user's accumulated handwriting model 822 to interpret the strokes as handwritten text. Once it has converted the strokes to text, the registration server returns the text to the requesting page server. The page server appends the text to the text value of the text field.

5 **5.4 SIGNATURE VERIFICATION PROTOCOL**

Assuming the input element whose zone the stroke intersects is a signature field 880, the page server 10 appends the stroke to the signature field's digital ink.

10 After a period of inactivity in the zone of the signature field, the page server sends the pen ID 61 and the pending strokes to the registration server 11 for verification. It also sends the application ID 64 associated with the form of which the signature field is part, as well as the form ID 56 and the current data content of the form. The registration server identifies the user corresponding to the pen, and uses the user's dynamic signature biometric 818 to verify the strokes as the user's signature. Once it has verified the signature, the registration server uses the application ID 64 and user ID 60 to identify the user's application-specific private signature key. It then uses the key to generate a digital signature of the form data, and returns the digital signature to the requesting page server. The page server assigns the digital signature to the signature field and sets the associated form's status to frozen.

The digital signature includes the alias ID 65 of the corresponding user. This allows a single form to capture multiple users' signatures.

20 **5.5 FORM SUBMISSION PROTOCOL**

A preferred embodiment of a form submission protocol is shown in Figure 46.

Form submission occurs via a form hyperlink activation. It thus follows the protocol defined in Section 5.2, with some form-specific additions.

25 In the case of a form hyperlink, the hyperlink activation message sent by the page server 10 to the application 71 also contains the form ID 56 and the current data content of the form. If the form contains any signature fields, then the application verifies each one by extracting the alias ID 65 associated with the corresponding digital signature and obtaining the corresponding certificate from the registration server 11.

30 **5.6 COMMISSION PAYMENT PROTOCOL**

A preferred embodiment of a commission payment protocol is shown in Figure 47.

In an e-commerce environment, fees and commissions may be payable from an application provider to a publisher on click-throughs, transactions and sales. Commissions on fees and commissions may also be payable from the publisher to the provider of the printer.

35 The hyperlink request ID 52 is used to route a fee or commission credit from the target application provider 70a (e.g. merchant) to the source application provider 70b (i.e. publisher), and from the source application provider 70b to the printer provider 72.

40 The target application receives the hyperlink request ID from the page server 10 when the hyperlink is first activated, as described in Section 5.2. When the target application needs to credit the source application provider, it sends the

WO 01/89838

PCT/AU00/00517

- 40 -

application provider credit to the original page server together with the hyperlink request ID. The page server uses the hyperlink request ID to identify the source application, and sends the credit on to the relevant registration server 11 together with the source application ID 64, its own server ID 53, and the hyperlink request ID. The registration server credits the corresponding application provider's account 827. It also notifies the application provider.

5 If the application provider needs to credit the printer provider, it sends the printer provider credit to the original page server together with the hyperlink request ID. The page server uses the hyperlink request ID to identify the printer, and sends the credit on to the relevant registration server together with the printer ID. The registration server credits the corresponding printer provider account 814.

10 The source application provider is optionally notified of the identity of the target application provider, and the printer provider of the identity of the source application provider.

6. Netpage Pen Description

6.1 PEN MECHANICS

15 Referring to Figures 8 and 9, the pen, generally designated by reference numeral 101, includes a housing 102 in the form of a plastics moulding having walls 103 defining an interior space 104 for mounting the pen components. The pen top 105 is in operation rotatably mounted at one end 106 of the housing 102. A semi-transparent cover 107 is secured to the opposite end 108 of the housing 102. The cover 107 is also of moulded plastics, and is formed from semi-transparent material in order to enable the user to view the status of the LED mounted within the housing 102. The cover 107 includes

20 a main part 109 which substantially surrounds the end 108 of the housing 102 and a projecting portion 110 which projects back from the main part 109 and fits within a corresponding slot 111 formed in the walls 103 of the housing 102. A radio antenna 112 is mounted behind the projecting portion 110, within the housing 102. Screw threads 113 surrounding an aperture 113A on the cover 107 are arranged to receive a metal end piece 114, including corresponding screw threads 115. The metal end piece 114 is removable to enable ink cartridge replacement.

25 Also mounted within the cover 107 is a tri-color status LED 116 on a flex PCB 117. The antenna 112 is also mounted on the flex PCB 117. The status LED 116 is mounted at the top of the pen 101 for good all-around visibility.

30 The pen can operate both as a normal marking ink pen and as a non-marking stylus. An ink pen cartridge 118 with nib 119 and a stylus 120 with stylus nib 121 are mounted side by side within the housing 102. Either the ink cartridge nib 119 or the stylus nib 121 can be brought forward through open end 122 of the metal end piece 114, by rotation of the pen top 105. Respective slider blocks 123 and 124 are mounted to the ink cartridge 118 and stylus 120, respectively. A rotatable cam barrel 125 is secured to the pen top 105 in operation and arranged to rotate therewith. The cam barrel 125 includes a cam 126 in the form of a slot within the walls 181 of the cam barrel. Cam followers 127 and 128 projecting from slider blocks 123 and 124 fit within the cam slot 126. On rotation of the cam barrel 125, the slider blocks 123 or 124 move relative to each other to project either the pen nib 119 or stylus nib 121 out through the hole 122 in the metal end piece 114. The pen 101 has three states of operation. By turning the top 105 through 90° steps, the three states are:

- 35 •Stylus 120 nib 121 out;
- Ink cartridge 118 nib 119 out; and
- 40 •Neither ink cartridge 118 nib 119 out nor stylus 120 nib 121 out.

WO 01/89838

PCT/AU00/00517

- 41 -

A second flex PCB 129, is mounted on an electronics chassis 130 which sits within the housing 102. The second flex PCB 129 mounts an infrared LED 131 for providing infrared radiation for projection onto the surface. An image sensor 132 is provided mounted on the second flex PCB 129 for receiving reflected radiation from the surface. The second flex PCB 129 also mounts a radio frequency chip 133, which includes an RF transmitter and RF receiver, and a controller chip 134 for controlling operation of the pen 101. An optics block 135 (formed from moulded clear plastics) sits within the cover 107 and projects an infrared beam onto the surface and receives images onto the image sensor 132. Power supply wires 136 connect the components on the second flex PCB 129 to battery contacts 137 which are mounted within the cam barrel 125. A terminal 138 connects to the battery contacts 137 and the cam barrel 125. A three volt rechargeable battery 139 sits within the cam barrel 125 in contact with the battery contacts. An induction charging coil 140 is mounted about the second flex PCB 129 to enable recharging of the battery 139 via induction. The second flex PCB 129 also mounts an infrared LED 143 and infrared photodiode 144 for detecting displacement in the cam barrel 125 when either the stylus 120 or the ink cartridge 118 is used for writing, in order to enable a determination of the force being applied to the surface by the pen nib 119 or stylus nib 121. The IR photodiode 144 detects light from the IR LED 143 via reflectors (not shown) mounted on the slider blocks 123 and 124.

Rubber grip pads 141 and 142 are provided towards the end 108 of the housing 102 to assist gripping the pen 101, and top 105 also includes a clip 142 for clipping the pen 101 to a pocket.

6.2 PEN CONTROLLER

The pen 101 is arranged to determine the position of its nib (stylus nib 121 or ink cartridge nib 119) by imaging, in the infrared spectrum, an area of the surface in the vicinity of the nib. It records the location data from the nearest location tag, and is arranged to calculate the distance of the nib 121 or 119 from the location tag utilising optics 135 and controller chip 134. The controller chip 134 calculates the orientation of the pen and the nib-to-tag distance from the perspective distortion observed on the imaged tag.

Utilising the RF chip 133 and antenna 112 the pen 101 can transmit the digital ink data (which is encrypted for security and packaged for efficient transmission) to the computing system.

When the pen is in range of a receiver, the digital ink data is transmitted as it is formed. When the pen 101 moves out of range, digital ink data is buffered within the pen 101 (the pen 101 circuitry includes a buffer arranged to store digital ink data for approximately 10 minutes of the pen motion on the surface) and can be transmitted later.

WO 01/89838

PCT/AU00/00517

- 42 -

154 are also included to interface with the RF chip 133 which includes an RF circuit 155 and RF resonators and inductors 156 connected to the antenna 112.

5 The controlling processor 145 captures and decodes location data from tags from the surface via the image sensor 132, monitors the force sensor photodiode 144, controls the LEDs 116, 131 and 143, and handles short-range radio communication via the radio transceiver 153. It is a medium-performance (~40MHz) general-purpose RISC processor.

The processor 145, digital transceiver components (transceiver controller 153 and baseband circuit 154), image sensor interface 152, flash memory 147 and 512KB DRAM 148 are integrated in a single controller ASIC. Analog RF components (RF circuit 155 and RF resonators and inductors 156) are provided in the separate RF chip.

10 The image sensor is a 215x215 pixel CCD (such a sensor is produced by Matsushita Electronic Corporation, and is described in a paper by Itakura, K T Nobusada, N Okusenya, R Nagayoshi, and M Ozaki, "A 1mm 50k-Pixel IT CCD Image Sensor for Miniature Camera System", IEEE Transactions on Electronic Devices, Volt 47, number 1, January 2000, which is incorporated herein by reference) with an IR filter.

15 The controller ASIC 134 enters a quiescent state after a period of inactivity when the pen 101 is not in contact with a surface. It incorporates a dedicated circuit 130 which monitors the force sensor photodiode 144 and wakes up the controller 134 via the power manager 151 on a pen-down event.

20 The radio transceiver communicates in the unlicensed 900MHz band normally used by cordless telephones, or alternatively in the unlicensed 2.4GHz industrial, scientific and medical (ISM) band, and uses frequency hopping and collision detection to provide interference-free communication.

25 In an alternative embodiment, the pen incorporates an Infrared Data Association (IrDA) interface for short-range communication with a base station or netpage printer.

In a further embodiment, the pen 101 includes a pair of orthogonal accelerometers mounted in the normal plane of the pen 101 axis. The accelerometers 190 are shown in Figures 9 and 10 in ghost outline.

30 The provision of the accelerometers enables this embodiment of the pen 101 to sense motion without reference to surface location tags, allowing the location tags to be sampled at a lower rate. Each location tag ID can then identify an object of interest rather than a position on the surface. For example, if the object is a user interface input element (e.g. a command button), then the tag ID of each location tag within the area of the input element can directly identify the input element.

35 The acceleration measured by the accelerometers in each of the x and y directions is integrated with respect to time to produce an instantaneous velocity and position.

40 Since the starting position of the stroke is not known, only relative positions within a stroke are calculated. Although position integration accumulates errors in the sensed acceleration, accelerometers typically have high resolution, and the time duration of a stroke, over which errors accumulate, is short.

WO 01/89838

PCT/AU00/00517

- 43 -

7. Netpage Printer Description

7.1 PRINTER MECHANICS

5 The vertically-mounted netpage wallprinter 601 is shown fully assembled in Figure 11. It prints netpages on Letter/A4 sized media using duplexed 8 1/2" Memjet™ print engines 602 and 603, as shown in Figures 12 and 12a. It uses a straight paper path with the paper 604 passing through the duplexed print engines 602 and 603 which print both sides of a sheet simultaneously, in full color and with full bleed.

10 An integral binding assembly 605 applies a strip of glue along one edge of each printed sheet, allowing it to adhere to the previous sheet when pressed against it. This creates a final bound document 618 which can range in thickness from one sheet to several hundred sheets.

15 The replaceable ink cartridge 627, shown in Figure 13 coupled with the duplexed print engines, has bladders or chambers for storing fixative, adhesive, and cyan, magenta, yellow, black and infrared inks. The cartridge also contains a micro air filter in a base molding. The micro air filter interfaces with an air pump 638 inside the printer via a hose 639. This provides filtered air to the printheads to prevent ingress of micro particles into the Memjet™ printheads 350 which might otherwise clog the printhead nozzles. By incorporating the air filter within the cartridge, the operational life of the filter is effectively linked to the life of the cartridge. The ink cartridge is a fully recyclable product with a capacity for printing and gluing 3000 pages (1500 sheets).

20 Referring to Figure 12, the motorized media pick-up roller assembly 626 pushes the top sheet directly from the media tray past a paper sensor on the first print engine 602 into the duplexed Memjet™ printhead assembly. The two Memjet™ print engines 602 and 603 are mounted in an opposing in-line sequential configuration along the straight paper path. The paper 604 is drawn into the first print engine 602 by integral, powered pick-up rollers 626. The position and size of the paper 604 is sensed and full bleed printing commences. Fixative is printed simultaneously to aid drying in the shortest possible time.

25 The paper exits the first Memjet™ print engine 602 through a set of powered exit spike wheels (aligned along the straight paper path), which act against a rubberized roller. These spike wheels contact the "wet" printed surface and continue to feed the sheet 604 into the second Memjet™ print engine 603.

30 Referring to Figures 12 and 12a, the paper 604 passes from the duplexed print engines 602 and 603 into the binder assembly 605. The printed page passes between a powered spike wheel axle 670 with a fibrous support roller and another movable axle with spike wheels and a momentary action glue wheel. The movable axle/glue assembly 673 is mounted to a metal support bracket and it is transported forward to interface with the powered axle 670 via gears by action of a camshaft. A separate motor powers this camshaft.

35 The glue wheel assembly 673 consists of a partially hollow axle 679 with a rotating coupling for the glue supply hose 641 from the ink cartridge 627. This axle 679 connects to a glue wheel, which absorbs adhesive by capillary action through radial holes. A molded housing 682 surrounds the glue wheel, with an opening at the front. Pivoting side moldings and sprung outer doors are attached to the metal bracket and hinge out sideways when the rest of the assembly 673 is thrust forward. This action exposes the glue wheel through the front of the molded housing 682. Tension springs close the assembly and effectively cap the glue wheel during periods of inactivity.

40

WO 01/89838

PCT/AU00/00517

- 44 -

As the sheet 604 passes into the glue wheel assembly 673, adhesive is applied to one vertical edge on the front side (apart from the first sheet of a document) as it is transported down into the binding assembly 605.

7.2 PRINTER CONTROLLER ARCHITECTURE

- 5 The netpage printer controller consists of a controlling processor 750, a factory-installed or field-installed network interface module 625, a radio transceiver (transceiver controller 753, baseband circuit 754, RF circuit 755, and RF resonators and inductors 756), dual raster image processor (RIP) DSPs 757, duplexed print engine controllers 760a and 760b, flash memory 658, and 64MB of DRAM 657, as illustrated in Figure 14.
- 10 The controlling processor handles communication with the network 19 and with local wireless netpage pens 101, senses the help button 617, controls the user interface LEDs 613-616, and feeds and synchronizes the RIP DSPs 757 and print engine controllers 760. It consists of a medium-performance general-purpose microprocessor. The controlling processor 750 communicates with the print engine controllers 760 via a high-speed serial bus 659.
- 15 The RIP DSPs rasterize and compress page descriptions to the netpage printer's compressed page format. Each print engine controller expands, dithers and prints page images to its associated Menjet™ printhead 350 in real time (i.e. at over 30 pages per minute). The duplexed print engine controllers print both sides of a sheet simultaneously. The master print engine controller 760a controls the paper transport and monitors ink usage in conjunction with the master QA chip 665 and the ink cartridge QA chip 761.
- 20 The printer controller's flash memory 658 holds the software for both the processor 750 and the DSPs 757, as well as configuration data. This is copied to main memory 657 at boot time.
- 25 The processor 750, DSPs 757, and digital transceiver components (transceiver controller 753 and baseband circuit 754) are integrated in a single controller ASIC 656. Analog RF components (RF circuit 755 and RF resonators and inductors 756) are provided in a separate RF chip 762. The network interface module 625 is separate, since netpage printers allow the network connection to be factory-selected or field-selected. Flash memory 658 and the 2×256Mbit (64MB) DRAM 657 is also off-chip. The print engine controllers 760 are provided in separate ASICs.
- 30 A variety of network interface modules 625 are provided, each providing a netpage network interface 751 and optionally a local computer or network interface 752. Netpage network Internet interfaces include POTS modems, Hybrid Fiber-Coax (HFC) cable modems, ISDN modems, DSL modems, satellite transceivers, current and next-generation cellular telephone transceivers, and wireless local loop (WLL) transceivers. Local interfaces include IEEE 1284 (parallel port), 10Base-T and 100Base-T Ethernet, USB and USB 2.0, IEEE 1394 (Firewire), and various emerging home networking interfaces. If
- 35 an Internet connection is available on the local network, then the local network interface can be used as the netpage network interface.
- 40 The radio transceiver 753 communicates in the unlicensed 900MHz band normally used by cordless telephones, or alternatively in the unlicensed 2.4GHz industrial, scientific and medical (ISM) band, and uses frequency hopping and collision detection to provide interference-free communication.

WO 01/89838

PCT/AU00/00517

- 45 -

The printer controller optionally incorporates an Infrared Data Association (IrDA) interface for receiving data "squirted" from devices such as netpage cameras. In an alternative embodiment, the printer uses the IrDA interface for short-range communication with suitably configured netpage pens.

7.2.1 RASTERIZATION AND PRINTING

5 Once the main processor 750 has received and verified the document's page layouts and page objects, it runs the appropriate RIP software on the DSPs 757.

10 The DSPs 757 rasterize each page description and compress the rasterized page image. The main processor stores each compressed page image in memory. The simplest way to load-balance multiple DSPs is to let each DSP rasterize a separate page. The DSPs can always be kept busy since an arbitrary number of rasterized pages can, in general, be stored in memory. This strategy only leads to potentially poor DSP utilization when rasterizing short documents.

15 Watermark regions in the page description are rasterized to a contone-resolution bi-level bitmap which is losslessly compressed to negligible size and which forms part of the compressed page image.

The infrared (IR) layer of the printed page contains coded netpage tags at a density of about six per inch. Each tag encodes the page ID, tag ID, and control bits, and the data content of each tag is generated during rasterization and stored in the compressed page image.

20 The main processor 750 passes back-to-back page images to the duplexed print engine controllers 760. Each print engine controller 760 stores the compressed page image in its local memory, and starts the page expansion and printing pipeline. Page expansion and printing is pipelined because it is impractical to store an entire 114MB bi-level CMYK+IR page image in memory.

7.2.2 PRINT ENGINE CONTROLLER

25 The page expansion and printing pipeline of the print engine controller 760 consists of a high speed IEEE 1394 serial interface 659, a standard JPEG decoder 763, a standard Group 4 Fax decoder 764, a custom halftoner/compositor unit 765, a custom tag encoder 766, a line loader/formatter unit 767, and a custom interface 768 to the Memjet™ printhead 350.

30 The print engine controller 360 operates in a double buffered manner. While one page is loaded into DRAM 769 via the high speed serial interface 659, the previously loaded page is read from DRAM 769 and passed through the print engine controller pipeline. Once the page has finished printing, the page just loaded is printed while another page is loaded.

35 The first stage of the pipeline expands (at 763) the JPEG-compressed contone CMYK layer, expands (at 764) the Group 4 Fax-compressed bi-level black layer, and renders (at 766) the bi-level netpage tag layer according to the tag format defined in section 1.2, all in parallel. The second stage dithers (at 765) the contone CMYK layer and composites (at 765) the bi-level black layer over the resulting bi-level CMYK layer. The resultant bi-level CMYK+IR dot data is buffered and formatted (at 767) for printing on the Memjet™ printhead 350 via a set of line buffers. Most of these line buffers are stored in the off-chip DRAM. The final stage prints the six channels of bi-level dot data (including fixative) to the Memjet™ printhead 350 via the printhead interface 768.

40 When several print engine controllers 760 are used in unison, such as in a duplexed configuration, they are synchronized

WO 01/89838

PCT/AU00/00517

- 46 -

via a shared line sync signal 770. Only one print engine 760, selected via the external master/slave pin 771, generates the line sync signal 770 onto the shared line.

5 The print engine controller 760 contains a low-speed processor 772 for synchronizing the page expansion and rendering pipeline, configuring the printhead 350 via a low-speed serial bus 773, and controlling the stepper motors 675, 676.

10 In the 8 1/2" versions of the netpage printer, the two print engines each prints 30 Letter pages per minute along the long dimension of the page (11"), giving a line rate of 8.8 kHz at 1600 dpi. In the 12" versions of the netpage printer, the two print engines each prints 45 Letter pages per minute along the short dimension of the page (8 1/2"), giving a line rate of 10.2 kHz. These line rates are well within the operating frequency of the Menjet™ printhead, which in the current design exceeds 30 kHz.

8. Print Engine Controller and Tag Encoder

15 A typically 12 inch printhead width is controlled by one or more PECs, as described below, to allow full-bleed printing of both A4 and Letter pages. Six channels of colored ink are the expected maximum in the present printing environment, these being:

- CMY, for regular color printing.
- K, for black text and other black printing.
- IR (infrared), for Netpage-enabled applications.
- 20 • F (fixative), to enable printing at high speed.

Because the printer is to be capable of fast printing, a fixative will be required to enable the ink to dry before the next page has completed printing at higher speeds. Otherwise the pages might bleed on each other. In lower speed printing environments the fixative will not be not required.

25 A PEC might be built in a single chip to interface with a printhead. It will contain four basic levels of functionality:

- receiving compressed pages via a serial interface such as IEEE 1394
- a print engine for producing a page from a compressed form. The print engine functionality includes expanding the page image, dithering the contone layer, compositing the black layer over the contone layer, and sending the resultant image to the printhead.
- 30 • a print controller for controlling the printhead and stepper motors.
- two standard low-speed serial ports for communication with the two QA chips. Note that there must be two ports and not a single port to ensure strong security during the authentication procedure.

35 In Figure 48 is seen the flow of data to send a document from computer system to printed page. A document is received at 411 and loaded to memory buffer 412 wherein page layouts may be effected and any required objects might be added. Pages from memory 412 are rasterized at 413 and compressed at 414 prior to transmission to the print engine controller 410. Pages are received as compressed two-layer page images within the print engine controller 410 into a memory buffer 415, from which they are fed to a page expander 416 wherein page images are retrieved. Any requisite dither might be applied to any contone layer at 417. Any black bi-level layer might be composited over the contone layer at 418 together with any infrared tags at 419. The composited page data is printed at 420 to produce page 421.

40 The print engine/controller takes the compressed page image and starts the page expansion and printing in pipeline

fashion. Page expansion and printing is preferably pipelined because it is impractical to store a sizable bi-level CMYK+IR page image in memory.

5 The first stage of the pipeline expands a JPEG-compressed contone CMYK layer (see below), expands a Group 4 Fax-compressed bi-level dither matrix selection map (see below), and expands a Group 4 Fax-compressed bi-level black layer (see below), all in parallel. The second stage dithers the contone CMYK layer using a dither matrix selected by the dither matrix select map, and composites the bi-level black layer over the resulting bi-level K layer. In parallel with this, the tag encoder encodes bi-level IR tag data from the compressed page image. A fixative layer is also generated at each dot position wherever there is a need in any of C, M, Y, K, or IR channels. The last stage prints the bi-level CMYK+IR data through the printhead via a printhead interface (see below).

10 In FIG. 49 is seen how the print engine/controller 410 fits within the overall printer system architecture. The various components of the printer system might include

- a *Print Engine/Controller (PEC)*. A PEC chip 410, or chips, is responsible for receiving the compressed page images for storage in a memory buffer 424, performing the page expansion, black layer compositing and sending the dot data to the printhead 423. It may also communicate with QA chips 425, 426 and provides a means of retrieving printhead characteristics to ensure optimum printing. The PEC is the subject of this specification.
- a *memory buffer*. The memory buffer 424 is for storing the compressed page image and for scratch use during the printing of a given page. The construction and working of memory buffers is known to those skilled in the art and a range of standard chips and techniques for their use might be utilised in use of the PEC of the invention.
- a *master QA chip*. The master chip 425 is matched to replaceable ink cartridge QA chips 426. The construction and working of QA units is known to those skilled in the art and a range of known QA processes might be utilised in use of the PEC of the invention. For example, a QA chip is described in co-pending United States Patent Applications:

USSN	Our Docket Number	Our Title
TBA	AUTH01	Validation Protocol and System
09/112,763	AUTH02	Circuit for Protecting Chips Against IDD Fluctuation Attacks
09/112,737	AUTH04	Method for Protecting On-Chip Memory (Flash and RAM)
09/112,761	AUTH05	Method for Making a Chip Tamper-Resistant
09/113,223	AUTH06	A system for authenticating physical objects
TBA	AUTH07	Validation Protocol and System
TBA	AUTH08	Validation Protocol and System
09/505,003	AUTH09	Consumable Authentication Protocol and System
09/517,608	AUTH10	Consumable Authentication Protocol and System
09/505,147	AUTH11	Consumable Authentication Protocol and System
09/505,952	AUTH12	Unauthorized Modification of Values Stored in Flash Memory
TBA	AUTH13	A System for the Manipulation of Secure Data
09/516,874	AUTH14	An Authentication Chip with Protection from Power Supply Attacks
TBA	AUTH15	Shielding Manipulations of Secret Data

25 QA chip communication may be best included within the overall functionality of the PEC chip since it has a role in the expansion of the image as well as running the physical printhead. By locating QA chip communication there it can be ensured that there is enough ink to print the page. Preferably the QA embedded in the printhead assembly is implemented

WO 01/89838

PCT/AU00/00517

- 48 -

using an authentication chip. Since it is a master QA chip, it only contains authentication keys, and does not contain user data. However, it must match the ink cartridge's QA chip. The QA chip in the ink cartridge contains information required for maintaining the best possible print quality, and is implemented using an authentication chip.

5 Preferably a 64 MBit (8 MByte) memory buffer is used to store the compressed page image. While one page is being written to the buffer another is being read (double buffering). In addition, the PEC uses the memory to buffer the calculated dot information during the printing of a page. During the printing of page *N*, the buffer is used for:

- Reading compressed page *N*
- Reading and writing the bi-level dot information for page *N*
- Writing compressed page *N+1*

10

Preferably a PEC chip will incorporate a simple micro-controller CPU core 435 to perform the following functions:

- perform QA chip authentication protocols between print pages
- run the stepper motor at parallel interface 589 during a print (the stepper motor requires a 5 KHz process)
- synchronize the various portions of the PEC chip during a print
- 15 • provide a means of interfacing with external data requests (programming registers etc.)
- provide a means of interfacing with printhead segment low-speed data requests (such as reading the characterization vectors and writing pulse profiles)
- provide a means of writing the portrait and landscape tag structures to external DRAM

20

Since all of the image processing is performed by dedicated hardware, the CPU does not have to process pixels. As a result, the CPU can be extremely simple. A wide variety of CPU known cores are suitable: it can be any processor core with sufficient processing power to perform the required calculations and control functions fast enough. An example of a suitable core is a Philips 8051 micro-controller running at about 1 MHz. Associated with the CPU core 435 may be a program ROM and a small program scratch RAM. The CPU communicates with the other units within the PEC chip via memory-mapped I/O. Particular address ranges may map to particular units, and within each range, to particular registers within that particular unit. This includes the serial and parallel interfaces. A small program flash ROM may be incorporated into the PEC chip. Its size depends on the CPU chosen, but should not be more than 8KB. Likewise, a small scratch RAM area can be incorporated into the PEC chip. Since the program code does not have to manipulate images, there is no need for a large scratch area. The RAM size depends on the CPU chosen (e.g. stack mechanisms, subroutine calling conventions, register sizes etc.), but should not be more than about 2 KB.

25

30

A PEC chip using the above referenced segment based page wide printhead can reproduce black at a full dot resolution (typically 1600 dpi), but reproduces contone color at a somewhat lower resolution using halftoning. The page description is therefore divided into a black bi-level layer and a contone layer. The black bi-level layer is defined to composite over the contone layer. The black bi-level layer consists of a bitmap containing a 1-bit *opacity* for each pixel. This black layer *matte* has a resolution which is an integer factor of the printer's dot resolution. The highest supported resolution is 1600 dpi, i.e. the printer's full dot resolution. The contone layer consists of a bitmap containing a 32-bit CMYK *color* for each pixel, where *K* is optional. This contone image has a resolution which is an integer factor of the printer's dot resolution. The highest supported resolution is 320 ppi over 12 inches for a single PEC, i.e. one-fifth the printer's dot resolution. For higher contone resolutions multiple PECs are required, with each PEC producing an strip of the output page. The contone resolution is also typically an integer factor of the black bi-level resolution, to simplify calculations in the RIPs. This is not a requirement, however. The black bi-level layer and the contone layer are both in compressed form for efficient storage in the printer's internal memory.

35

40

WO 01/89838

PCT/AU00/00517

- 49 -

In FIG. 50 is seen the print engine architecture. The print engine's page expansion and printing pipeline consists of a high speed serial interface 427 (such as a standard IEEE 1394 interface), a standard JPEG decoder 428, a standard Group 4 Fax decoder, a custom halftoner/compositor unit 429, a custom tag encoder 430, a line loader/formatter unit 431, and a custom interface 432 to the printhead 433. The decoders 428, 588 and encoder 430 are buffered to the halftoner/compositor 429.

5 The tag encoder 430 establishes an infrared tag or tags to a page according to protocols dependent on what uses might be made of the page and the actual content of a tag is not the subject of the present invention.

The print engine works in a double buffered way. One page is loaded into DRAM 34 via a DRAM interface 587 on bus 586 and the high speed serial interface 27 while the previously loaded page is read from DRAM 434 and passed through

10 the print engine pipeline. Once the page has finished printing, then the page just loaded becomes the page being printed, and a new page is loaded via the high speed serial interface 427. At the first stage the pipeline expands any JPEG-compressed contone (CMYK) layer, and expands any of two Group 4 Fax-compressed bi-level data streams. The two streams are the black layer (although the PEC is actually color agnostic and this bi-level layer can be directed to any of the output inks), and a matte for selecting between dither matrices for contone dithering (see below). At the second stage, in

15 parallel with the first, is encoded any tags for later rendering in either IR or black ink. Finally the third stage dithers the contone layer, and composites position tags and the bi-level spot1 layer over the resulting bi-level dithered layer. The datastream is ideally adjusted to create smooth transitions across overlapping segments in the printhead and ideally it is adjusted to compensate for dead nozzles in the printhead. Up to 6 channels of bi-level data are produced from this stage. Note that not all 6 channels may be present on the printhead. For example, the printhead may be CMY only, with K

20 pushed into the CMY channels and IR ignored. Alternatively, the position tags may be printed in K if IR ink is not available (or for testing purposes). The resultant bi-level CMYK-IR dot-data is buffered and formatted for printing on the printhead 33 via a set of line buffers (see below). The majority of these line buffers might be ideally stored on the off-chip DRAM 434. The final stage prints the 6 channels of bi-level dot data via the printhead interface 432.

Compression is used in a printing system that employs the PEC. This is to enable the data flow to keep ahead of the

25 printhead that is run at a constant speed. At 267 ppi, a Letter page of contone CMYK data has a size of 25MB. Using lossy contone compression algorithms such as JPEG (see below), contone images compress with a ratio up to 10:1 without noticeable loss of quality, giving a compressed page size of 2.5MB. At 800 dpi, a Letter page of bi-level data has a size of 7MB. Coherent data such as text compresses very well. Using lossless bi-level compression algorithms such as Group 4 Facsimile (see below), ten-point text compresses with a ratio of about 10:1, giving a compressed page size of 0.8MB.

30 Once dithered, a page of CMYK contone image data consists of 114MB of bi-level data. The two-layer compressed page image format described below exploits the relative strengths of lossy JPEG contone image compression and lossless bi-level text compression. The format is compact enough to be storage-efficient, and simple enough to allow straightforward real-time expansion during printing. Since text and images normally don't overlap, the normal worst-case page image size is 2.5MB (i.e. image only), while the normal best-case page image size is 0.8MB (i.e. text only). The absolute worst-case

35 page image size is 3.3MB (i.e. text over image). Assuming a quarter of an average page contains images, the average page image size is 1.2MB.

The Group 4 Fax (G4 Fax) decoder is responsible for decompressing bi-level data. Bi-level data is limited to a single spot color (typically black for text and line graphics), and a dither matrix select bit-map for use in subsequent dithering of the contone data (decompressed by the JPEG decoder). The input to the G4 Fax decoder is 2 planes of bi-level data, read

40 from the external DRAM. The output of the G4 Fax decoder is 2 planes of decompressed bi-level data. The decompressed

WO 01/89838

PCT/AU00/00517

- 50 -

bi-level data is sent to the Halftoner/Compositor Unit (HCU) for the next stage in the printing pipeline. Two bi-level buffers provides the means for transferring the bi-level data between the G4 Fax decoder and the HCU. Each decompressed bi-level layer is output to two line buffers. Each buffer is capable of holding a full 12 inch line of dots at the expected maximum resolution. Having two line buffers allows one line to be read by the HCU while the other line is being written to by the G4 Fax decoder. This is important because a single bi-level line is typically less than 1600 dpi, and must therefore be expanded in both the dot and line dimensions. If the buffering were less than a full line, the G4 Fax decoder would have to decode the same line multiple times - once for each output 600dpi dotline.

Spot color 1 is designed to allow high resolution dot data for a single color plane of the output image. While the contone layers provide adequate resolution for images, spot color 1 is targeted at applications such as text and line graphics (typically black). When used as text and line graphics, the typical compression ratio exceeds 10:1. Spot color 1 allows variable resolution up to 1600dpi for maximum print quality. Each of the two line buffers is therefore total 2400 bytes (12 inches \times 1600 dpi = 19,200 bits).

The resolution of the dither matrix select map should ideally match the contone resolution. Consequently each of the two line buffers is therefore 480 bytes (3840 bits), capable of storing 12 inches at 320 dpi. When the map matches the contone resolution, the typical compression ratio exceeds 50:1.

In order to provide support for:

- 800 dpi spot color 1 layer (typically black)
- 320 dpi dither matrix select layer

the decompression bandwidth requirements are 9.05 MB/sec for 1 page per second performance (regardless of whether the page width is 12 inches or 8.5 inches), and 20 MB/sec and 14.2 MB/sec for 12 inch and 8.5 inch page widths respectively during maximum printer speed performance (30,000 lines per second). Given that the decompressed data is output to a line buffer, the G4 Fax decoder can readily decompress a line from each of the outputs one at a time.

The G4 Fax decoder is fed directly from the main memory via the DRAM interface. The amount of compression determines the bandwidth requirements to the external DRAM. Since G4 Fax is lossless, the complexity of the image impacts on the amount of data and hence the bandwidth. typically an 800 dpi black text/graphics layer exceeds 10:1 compression, so the bandwidth required to print 1 page per second is 0.78 MB/sec. Similarly, a typical 320 dpi dither select matrix compresses at more than 50:1, resulting in a 0.025 MB/sec bandwidth. The fastest printing speed configuration of 320 dpi for dither select matrix and 800 dpi for spot color 1 requires bandwidth of 1.72 MB/sec and 0.056 MB/sec respectively. A total bandwidth of 2 MB/sec should therefore be more than enough for the DRAM bandwidth.

The G4 Fax decoding functionality is implemented by means of a G4 Fax Decoder core. A wide variety of G4Fax Decoder cores are suitable: it can be any core with sufficient processing power to perform the required calculations and control functions fast enough. It must be capable of handling runlengths exceeding those normally encountered in 400 dpi facsimile applications, and so may require modification.

The CMYK (or CMY) contone layer is compressed to a planar color JPEG bytestream. If luminance/chrominance separation is deemed necessary, either for the purposes of table sharing or for chrominance subsampling, then CMYK is converted to YCrCb and Cr and Cb are duly subsampled. The JPEG bytestream is complete and self-contained. It contains all data required for decompression, including quantization and Huffman tables.

The JPEG decoder is responsible for performing the on-the-fly decompression of the contone data layer. The input to the JPEG decoder is up to 4 planes of contone data. This will typically be 3 planes, representing a CMY contone image, or 4 planes representing a CMYK contone image. Each color plane can be in a different resolution, although typically all color

WO 01/89838

PCT/AU00/00517

- 51 -

planes will be the same resolution. The contone layers are read from the external DRAM. The output of the JPEG decoder is the decompressed contone data, separated into planes. The decompressed contone image is sent to the halftoner/compositor unit (HCU) 429 for the next stage in the printing pipeline. The 4-plane contone buffer provides the means for transferring the contone data between the JPEG decoder and the HCU 429.

5 Each color plane of the decompressed contone data is output to a set of two line buffers (see below). Each line buffer is 3840 bytes, and is therefore capable of holding 12 inches of a single color plane's pixels at 320 ppi. The line buffering allows one line buffer to be read by the HCU while the other line buffer is being written to by the JPEG decoder. This is important because a single contone line is typically less than 1600 ppi, and must therefore be expanded in both the dot and line dimensions. If the buffering were less than a full line, the JPEG decoder would have to decode the same line multiple times - once for each output 600dpi dotline. Although a variety of resolutions is supported, there is a tradeoff between the resolution and available bandwidth. As resolution and number of colors increase, bandwidth requirements also increase. In addition, the number of segments being targeted by the PEC chip also affects the bandwidth and possible resolutions. Note that since the contone image is processed in a planar format, each color plane can be stored at a different resolution (for example CMY may be a higher resolution than the K plane). The highest supported contone resolution is 1600ppi (matching the printer's full dot resolution). However there is only enough output line buffer memory to hold enough contone pixels for a 320ppi line of length 12 inches. If the full 12 inches of output was required at higher contone resolution, multiple PEC chips would be required, although it should be noted that the final output on the printer will still only be bi-level. With support for 4 colors at 320ppi, the decompression output bandwidth requirements are 40 MB/sec for 1 page per second performance (regardless of whether the page width is 12 inches or 8.5 inches), and 88 MB/sec and 64 MB/sec for 12 inch and 8.5 inch page widths respectively during maximum printer speed performance (30,000 lines per second). Table 5 can be used to determine the bandwidth required for different resolution / color plane / page width combinations.

25 The JPEG decoder is fed directly from the main memory via the DRAM interface. The amount of compression determines the bandwidth requirements to the external DRAM. As the level of compression increases, the bandwidth decreases, but the quality of the final output image can also decrease. The DRAM bandwidth for a single color plane can be readily calculated by applying the compression factor to the output bandwidth shown in Table 5. For example, a single color plane at 320 ppi with a compression factor of 10:1 requires 1MB/sec access to DRAM to produce a single page per second.

30 The JPEG functionality is implemented by means of a JPEG core. A wide variety of JPEG cores are suitable: it can be any JPEG core with sufficient processing power to perform the required calculations and control functions fast enough. For example, the BTG X-Match core has decompression speeds up to 140 MBytes/sec, which allows decompression of 4 color planes at contone resolutions up to 400ppi for the maximum printer speed (30,000 lines at 1600dpi per second), and 800ppi for 1 page/sec printer speed. Note that the core needs to only support decompression, reducing the requirements that are imposed by more generalized JPEG compression/decompression cores. The size of the core is expected to be no more than 100,000 gates. Given that the decompressed data is output to a line buffer, the JPEG decoder can readily decompress an entire line for each of the color planes one at a time, thus saving on context switching during a line and simplifying the control of the JPEG decoder. 4 contexts must be kept (1 context for each color plane), and includes current address in the external DRAM as well as appropriate JPEG decoding parameters

40 In FIG. 51 the halftoner/compositor unit (HCU) 429 combines the functions of halftoning the contone (typically CMYK) layer to a bi-level version of the same, and compositing the spot1 bi-level layer over the appropriate halftoned contone

WO 01/89838

PCT/AU00/00517

- 52 -

layer(s). If there is no K ink in the printer, the HCU 429 is able to map K to CMY dots as appropriate. It also selects between two dither matrices on a pixel by pixel basis, based on the corresponding value in the dither matrix select map. The input to the HCU 429 is an expanded contone layer (from the JPEG decoder unit) through buffer 437, an expanded bi-level spot1 layer through buffer 438, an expanded dither-matrix-select bitmap at typically the same resolution as the contone layer through buffer 439, and tag data at full dot resolution through buffer 440. The HCU 429 uses up to two dither matrices, read from the external DRAM 434. The output from the HCU 429 to the line loader/format unit (LLFU) at 441 is a set of printer resolution bi-level image lines in up to 6 color planes. Typically, the contone layer is CMYK or CMY, and the bi-level spot1 layer is K.

In FIG. 52 is seen the HCU in greater detail. Once started, the HCU proceeds until it detects an *end-of-page* condition, or until it is explicitly stopped via its control register. The first task of the HCU is to scale, in the respective scale units such as the scale unit 443, all data, received in the buffer planes such as 442, to printer resolution both horizontally and vertically. The scale unit provides a means of scaling contone or bi-level data to printer resolution both horizontally and vertically. Scaling is achieved by replicating a data value an integer number of times in both dimensions. Processes by which to scale data will be familiar to those skilled in the art. Since each of the contone layers can be a different resolution, they are scaled independently. The bi-level spot1 layer at buffer 445 and the dither matrix select layer at buffer 446 also need to be scaled. The bi-level tag data at buffer 447 is established at the correct resolution and does not need to be scaled. The scaled-up dither matrix select bit is used by the dither matrix access unit 448 to select a single 8-bit value from the two dither matrices. The 8-bit value is output to the 4 comparators 444, and 449 to 451, which simply compare it to the specific 8-bit contone value. The generation of an actual dither matrix is dependent on the structure of the printhead and the general processes by which to generate one will be familiar to those skilled in the art. If the contone value is greater than or equal to the 8-bit dither matrix value a 1 is output. If not, then a 0 is output. These bits are then all ANDed at 452 to 456 with an *inPage* bit from the margin unit 457 (whether or not the particular dot is inside the printable area of the page). The final stage in the HCU is the compositing stage. For each of the 6 output layers there is a single dot merger unit, such as unit 458, each with 6 inputs. The single output bit from each dot merger unit is a combination of any or all of the input bits. This allows the spot color to be placed in any output color plane (including infrared for testing purposes), black to be merged into cyan, magenta and yellow (if no black ink is present in the printhead), and tag dot data to be placed in a visible plane. A fixative color plane can also be readily generated. The dot reorg unit (DRU) 459 is responsible for taking the generated dot stream for a given color plane and organizing it into 32-bit quantities so that the output is in segment order, and in dot order within segments. Minimal reordering is required due to the fact that dots for overlapping segments are not generated in segment order.

Two control bits are provided to the scale units by the margin unit 457: *advance dot* and *advance line*. The *advance dot* bit allows the state machine to generate multiple instances of the same dot data (useful for page margins and creating dot data for overlapping segments in the Memjet printhead). The *advance line* bit allows the state machine to control when a particular line of dots has been finished, thereby allowing truncation of data according to printer margins. It also saves the scale unit from requiring special end-of-line logic.

The comparator unit contains a simple 8-bit "greater-than-or-equal" comparator. It is used to determine whether the 8-bit contone value is greater than or equal to the 8-bit dither matrix value. As such, the comparator unit takes two 8-bit inputs and produces a single 1-bit output.

In FIG. 53 is seen more detail of the dot merger unit. It provides a means of mapping the bi-level dithered data, the spot1 color, and the tag data to output inks in the actual printhead. Each dot merger unit takes 6 1-bit inputs and produces a single bit output that represents the output dot for that color plane. The output bit at 460 is a combination of any or all of

WO 01/89838

PCT/AU00/00517

- 53 -

the input bits. This allows the spot color to be placed in any output color plane (including infrared for testing purposes), black to be merged into cyan, magenta and yellow (in the case of no black ink in the printhead), and tag dot data to be placed in a visible plane. An output for fixative can readily be generated by simply combining all of the input bits. The dot merger unit contains a 6-bit *ColorMask* register 461 that is used as a mask against the 6 input bits. Each of the input bits is ANDed with the corresponding *ColorMask* register bit, and the resultant 6 bits are then ORed together to form the final output bit.

In FIG. 54 is seen the dot reorg unit (DRU) which is responsible for taking the generated dot stream for a given color plane and organizing it into 32-bit quantities so that the output is in segment order, and in dot order within segments. Minimal reordering is required due to the fact that dots for overlapping segments are not generated in segment order. The DRU contains a 32-bit shift register, a regular 32-bit register, and a regular 16-bit register. A 5-bit counter keeps track of the number of bits processed so far. The dot advance signal from the dither matrix access unit (DMAU) is used to instruct the DRU as to which bits should be output.

In FIG. 54 register(A) 462 is clocked every cycle. It contains the 32 most recent dots produced by the dot merger unit (DMU). The full 32-bit value is copied to register(B) 463 every 32 cycles by means of a *WriteEnable* signal produced by the DRU state machine 464 via a simple 5-bit counter. The 16 odd bits (bits 1, 3, 5, 7 etc.) from register(B) 463 are copied to register(C) 465 with the same *WriteEnable* pulse. A 32-bit multiplexor 466 then selects between the following 3 outputs based upon 2 bits from the state machine:

- The full 32 bits from register B
- A 32-bit value made up from the 16 even bits of register A (bits 0, 2, 4, 6 etc.) and the 16 even bits of register B. The 16 even bits from register A form bits 0 to 15, while the 16 even bits from register B form bits 16-31.
- A 32-bit value made up from the 16 odd bits of register B (bits 1, 3, 5, 7 etc.) and the 16 bits of register C. The bits of register C form bits 0 to 15, while the odd bits from register B form bits 16-31.

The state machine for the DRU can be seen in Table 1. It starts in state 0. It changes state every 32 cycles. During the 32 cycles a single *noOverlap* bit collects the AND of all the *dot advance* bits for those 32 cycles (*noOverlap* = *dot advance* for cycle 0, and *noOverlap* = *noOverlap* AND *dot advance* for cycles 1 to 31).

Table 1. State machine for DRU

state	NoOverlap	Output	output Valid	Comment	Next state
0	X	B	0	Startup state	1
1	1	B	1	Regular non-overlap	1
1	0	B	1	A contains first overlap	2
2	X	Even A, even B	1	A contains second overlap B contains first overlap	3
3	X	C, odd B	1	C contains first overlap B contains second overlap	1

The margin unit (MU) 457, in FIG. 52, is responsible for turning *advance dot* and *advance line* signals from the dither matrix access unit (DMAU) 448 into general control signals based on the page margins of the current page. It is also responsible for generating the *end of page* condition. The MU keeps a counter of dot and line across the page. Both are set to 0 at the beginning of the page. The dot counter is advanced by 1 each time the MU receives a *dot advance* signal from the DMAU. When the MU receives a *line advance* signal from the DMAU, the line counter is incremented and the

WO 01/89838

PCT/AU00/00517

- 54 -

dot counter is reset to 0. Each cycle, the current line and dot values are compared to the margins of the page, and appropriate output *dot advance*, *line advance* and *within margin* signals are given based on these margins. The DMAU contains the only substantial memory requirements for the HCU.

- 5 Apart from being implicitly defined in relation to the printable page area, each page description is complete and self-contained. There is no data stored separately from the page description to which the page description refers. PEC relies on diher matrices and tag structures to have already been set up, but these are not considered to be part of a general page format.

- 10 The page description consists of a page header which describes the size and resolution of the page, followed by one or more page bands which describe the actual page content.

Table 2 shows the format of the page header.

Table 2. Page header format

Field	format	Description
Signature	16-bit integer	Page header format signature.
Version	16-bit integer	Page header format version number.
structure size	16-bit integer	Size of page header.
target resolution (dpi)	16-bit integer	Resolution of target page. This is always 1600 for the present printer.
target page width	16-bit integer	Width of target page, in dots.
target page height	16-bit integer	Height of target page, in dots.
target left margin	16-bit integer	Width of target left margin, in dots.
target top margin	16-bit integer	Height of target top margin, in dots.
tag flags	16-bit integer	Bit 0 specifies whether to generate tags for this page (0 = no, 1 = yes). Bit 1 specifies the tag orientation (0 = portrait, 1 = landscape). Bit 2 specifies whether the fixed tag data should be redundantly encoded by PEC or directly used (0=directly use, 1 = encode). Bit 3 specifies whether the variable tag data should be redundantly encoded by PEC or directly used (0=directly use, 1 = encode). The remaining bits are reserved.
fixed tag data	128-bit integer	This is only valid if the generate tags flag is set (bit 0 of tag flags). If bit 1 of tag flags is clear, then the lower 120 bits of fixed tag data contain the pre-encoded fixed data. If bit 1 of tag flags is set, then the lower 40

- 55 -

		bits contain the unencoded fixed data that is to be encoded by PEC. The upper 8 bits are reserved.
black scale factor	16-bit integer	Scale factor from black bi-level resolution to target resolution (must be 1 or greater).
black page width	16-bit integer	Width of black page, in black pixels.
black page height	16-bit integer	Height of black page, in black pixels.
contone color space	16-bit integer	Defines the number of contone JPBG channels. Typically 3 or 4 for CMY vs CMYK.
contone1 scale factor	16-bit integer	Scale factor from contone channel 1 resolution to target resolution (must be 1 or greater)
contone1 page width	16-bit integer	Width of contone page, in contone1 pixels.
contone1 page height	16-bit integer	Height of contone page, in contone1 pixels.
contone2 scale factor	16-bit integer	Scale factor from contone channel 2 resolution to target resolution (must be 1 or greater)
contone2 page width	16-bit integer	Width of contone page, in contone2 pixels.
contone2 page height	16-bit integer	Height of contone page, in contone2 pixels.
contone3 scale factor	16-bit integer	Scale factor from contone channel 3 resolution to target resolution (must be 1 or greater)
contone3 page width	16-bit integer	Width of contone page, in contone3 pixels.
contone3 page height	16-bit integer	Height of contone page, in contone3 pixels.
contone4 scale factor	16-bit integer	Scale factor from contone channel 4 resolution to target resolution (must be 1 or greater)
contone4 page width	16-bit integer	Width of contone page, in contone4 pixels.
contone4 page height	16-bit integer	Height of contone page, in contone4 pixels.

- The page header contains a signature and version which allow the print engine to identify the page header format. If the signature and/or version are missing or incompatible with the print engine, then the print engine can reject the page. The contone color space defines how many contone layers are present, which typically is used for defining whether the contone layer is CMY or CMYK. The page header defines the resolution and size of the target page. The black and contone layers are clipped to the target page if necessary. This happens whenever the black or contone scale factors are not factors of the target page width or height. The target left and top margins define the positioning of the target page within the printable page area.
- 5 The tag parameters specify whether or not Netpage tags should be produced for this page and what orientation the tags should be produced at (landscape or portrait mode). The fixed tag data is also provided.
- 10 The black layer parameters define the pixel size of the bi-level black layer, and its integer scale factor to the target resolution. The contone layer parameters define the pixel size of each of the four contone layers and their integer scale factor to the target resolution.
- Table 3 shows the format of the page band header.

WO 01/89838

PCT/AU00/00517

- 56 -

Table 3. Page band header format.

Field	format	description
Signature	16-bit integer	Page band header format signature.
Version	16-bit integer	Page band header format version number.
Structure size	16-bit integer	Size of page band header.
Black band height	16-bit integer	Height of black band, in black pixels.
Black band data size	32-bit integer	Size of black band data, in bytes.
Contone band height	16-bit integer	Height of contone band, in contone pixels.
contone band data size	32-bit integer	Size of contone band data, in bytes.
dither matrix select map band data size	32-bit integer	Size of dither matrix select map band data, in bytes. If the size = 0 only one dither matrix is used.
Tag band data size	32-bit integer	Size of unencoded tag data band, in bytes. Can be 0 which indicates that no tag data is provided.

5 The black (bi-level) layer parameters define the height of the black band, and the size of its compressed band data. The variable-size black data follows the page band header. The contone layer parameters define the height of the contone band, and the size of its compressed page data, consisting of the contone color data and the associated bi-level dither matrix select map. The variable-size contone data follows the black data. The variable-size bi-level dither matrix select map data follows the contone data.

10 The tag band data is the set of tag data half-lines as required by the tag encoder. The format of the tag data is found below. The tag band data follows the dither matrix select map.

Table 4 shows the format of the variable-size compressed band data which follows the page band header.

Table 4. Page band data format

Field	format	description
black data	G4Fax bytestream	Compressed bi-level black data.
contone data	JPEG bytestream	Compressed contone CMYK or CMY data.
dither matrix select map	G4Fax bytestream	Compressed bi-level dither matrix select map data.
tag data map	bitmap	Tag data format. See Section 9.2.2.

15 Each variable-size segment of band data is aligned to an 8-byte boundary.

The tag encoder (TE) 430 in FIG. 50, provides functionality for tag-enabled applications, and it typically requires the presence of IR ink at the print head (although K ink or other might be used for tags in limited circumstances). The TE encodes fixed data for the page being printed, together with specific tag data values into an error-correctable encoded tag which is subsequently printed in infrared or black ink on the page. The TE might place tags on a triangular grid (see FIG.

20 55), allowing for both landscape and portrait orientations. Basic tag structures are rendered at 1600 dpi, while tag data

WO 01/89838

PCT/AU00/00517

- 57 -

might be encoded as arbitrarily shaped macrodots (with a minimum size of 1 dot at 1600 dpi).

The TE takes the following as input:

- A portrait/landscape flag
- A template defining the structure of a single tag
- 5 • A number of fixed data bits (fixed for the page)
- A flag that defines whether or not to redundantly encode the fixed data bits or whether to treat the bits as already having been encoded
- A number of variable data bit records, where each record contains the variable data bits for the tags on a given line of tags
- 10 • A flag that defines whether or not to redundantly encode the variable data bits or whether to treat the bits as already having been encoded.

The output from the tag encoder (TE) is a 1600 dpi bi-level layer of where tag data should be printed. The output is via a 1-bit wide FIFO 447 (in FIGs 50 and 52) which is in turn used as input by the HCU 429 in FIG. 50. The tags are subsequently preferably printed with an infrared-absorptive ink that can be read by a tag sensing device. Since black ink can be IR absorptive, limited functionality can be provided on offset-printed pages using black ink on otherwise blank areas of the page - for example to encode buttons. Alternatively an invisible infrared ink can be used to print the position tags over the top of a regular page. However, if invisible IR ink is used, care must be taken to ensure that any other printed information on the page is printed in infrared-transparent CMY ink, for black ink will obscure the infrared tags.

20 The monochromatic scheme is preferred to maximize dynamic range in blurry reading environments.

When multiple PEC chips are used for printing the same side of a page, it is possible that a single tag will be produced by two PEC chips. This implies that the tag encoder must be able to print partial tags.

Since the tag encoder (TE) outputs 1600 dpi bi-level data, the internal workings of the TE are completely hidden from the half-toner/compositor unit (the user of tag data).

25 Even though the conceptual implementation of the tag encoder (TE) allows tags to have a variable structure as well as fixed and variable data components, this implementation of the TE does impose range restrictions on certain encoding parameters. Table 5 lists the encoding parameters as well as the range restrictions. However, these restrictions are a direct result of buffer sizes and the number of addressing bits, chosen for the most likely encoding scenarios. It is a simple matter to adjust the buffer sizes and corresponding addressing to allow arbitrary encoding parameters in other implementations.

30

Table 5. Encoding parameters

Name	Definition	maximum value imposed by TE
W	page width	12 inches
S	tag size	minimum is 2mm \times 2mm
N	number of dots in each dimension of the tag	384 dots (minimum of 92 dots given E)
E	Redundancy encoding for tag data	Reed-Solomon GF(2 ⁸) at 5:10
D_F	size of fixed data (unencoded)	40 bits
R_F	size of redundancy encoded fixed data	120 bits

WO 01/89838

PCT/AU00/00517

- 58 -

D_V	size of variable data (unencoded)	120 bits
R_V	size of redundancy encoded variable data	360 bits
T	tags per page width	152 (allows for packed 2mm x 2mm tags)
M	Macrodot size	Minimum is 1 dot

Of particular note is the fixed and variable data component in each tag. The fixed data component is the part of tag *data* that does not change (different to the part of the tag *structure* that does change). The fixed data is either read by the PEC chip in its unencoded form and encoded once within PEC, or it can be read and used as-is (the fixed data should therefore be redundantly encoded externally). The variable data bits are those data bits that are variable for each tag, and are as with fixed data, are redundancy encoded inside the TE as required or used as-is.

The mapping of data bits (both fixed and variable) to redundancy encoded bits relies heavily on the method of redundancy encoding employed. Reed-Solomon encoding was chosen for its ability to deal with burst errors and effectively detect and correct errors using a minimum of redundancy. Reed Solomon encoding is discussed in Lypkens, H., "Reed-Solomon Error Correction", *Dr. Dobbs's Journal* Vol.22, No.1, January 1997, Rorabaugh, C, *Error Coding Cookbook*, McGraw-Hill 1996, and Wicker, S., and Bhargava, V., *Reed-Solomon Codes and their Applications*, IEEE Press 1994.

In the present implementation of the tag encoder (TE) is used Reed-Solomon encoding over the Galois Field $GF(2^4)$. Symbol size is 4 bits. Each codeword contains 15 4-bit symbols for a codeword length of 60 bits. Of the 15 symbols, 5 are original data (20 bits), and 10 are redundancy bits (40 bits). The 10 redundancy symbols mean that we can correct up to 5 symbols in error.

The total amount of original data per tag is 160 bits (40 fixed, 120 variable). This is redundancy encoded to give a total amount of 480 bits (120 fixed, 360 variable) as follows:

- Each tag contains up to 40 bits of fixed original data. Therefore 2 codewords are required for the fixed data, giving a total encoded data size of 120 bits. Note that this fixed data only needs to be encoded once per page.
- Each tag contains up to 120 bits of variable original data. Therefore 6 codewords are required for the variable data, giving a total encoded data size of 360 bits.

The TE writes a bi-level tag bitstream to the bi-level tag FIFO. The TE is responsible for merging the encoded tag data with the basic tag structure, and placing the dots in the output FIFO in the correct order for subsequent printing. The encoded tag data is generated from the original data bits on-the-fly to minimize buffer space.

In FIG. 55 is seen the placement of tags for portrait and landscape printing. The TE preferably places tags 488 on the page in a triangular grid arrangement, 488,489,490, accounting for both landscape 492 and portrait orientations 491. The triangular mesh of tags 488,489,490 combined with the restriction of only two printing orientations (landscape and portrait) and no overlap of columns or rows of tags means that the process of tag placement is greatly simplified.

In FIG. 56 is seen the general case for placement of tags therefore relies on a number of parameters. For a given line of dots, all the tags on that line correspond to the same part of the general tag structure. The triangular placement can be considered as alternative lines of tags, where one line of tags is inset by one amount in the dot dimension, and the other line of dots is inset by a different amount. The dot inter-tag gap 493 is the same in both lines of tag, and is different from the line inter-tag gap 494.

WO 01/89838

PCT/AU00/00517

- 59 -

The parameters are more formally described in Table 6 and Table 7. Note that only one set of parameters are required - those for portrait printing. If the orientation changes from portrait to landscape, then tag height and tag width parameters, and general dot and line parameters are simply interchanged.

5 Table 6. Tag placement parameters

Parameter	Description	Restrictions
TagHeight	The number of dot lines in a tag's bounding box.	minimum 1
TagWidth	The number of dots in a single line of the tag's bounding box. The number of dots in the tag itself will vary depending on the shape of the tag, but the number of dots in the bounding box will be constant (by definition).	minimum 1
Dot inter-tag gap	The number of dots from the edge of one tag's bounding box to the start of the next tag's bounding box, in the dot direction.	minimum = 0
Line inter-tag gap	The number of dot lines from the edge of one tag's bounding box to the start of the next tag's bounding box, in the line direction.	minimum = 0
Start Position	The Current Position Record (see Table 10) for the start of the first row of dots on the page (or strip if multiple PECs are used) and the first row of tags. To increase the size of the non-tagged section beyond the size of the inter-tag gap, use non-printed tags.	see Table 10
AllTagLinePosition	The Current Position record for the start of the alternate row of tags. The dot parameter is used for portrait mode printing, and the line parameter is used for landscape mode printing.	see Table 10

Table 7. Current position record

Name	Description
TagStateDot	0 = in inter-tag gap 1 = in tag
TagStateLine	0 = in inter-tag gap 1 = in tag
LocalOffsetDot	Current dot position within the inter-tag gap or tag, minimum = 0
LocalOffsetLine	Current line position within the inter-tag gap or tag, minimum = 0

10

The TE makes use of several specific data structures:

- a *TEOrientation* flag, that determines whether the page is being printed using portrait or landscape tag placement

WO 01/89838

PCT/AU00/00517

- 60 -

rules.

- a *tag format structure*. A template detailing the composition of a generic tag in terms of fixed tag structure, variable data bits and fixed data bits. It is composed of a number of *tag line structures*, one for each 1600 dpi line in the tag. There are two tag format structures - one for portrait and one for landscape printing.
- 5 • a *fixed tag data* buffer. Contains the redundancy encoded fixed data component for all tags on the page (or part of a page when multiple PEC chips are used).
- a *TagsPrinted* flag. Specifies whether a particular tag is to be printed or not. Directs the encoder whether to ignore the tag format structure and simply output no tag.
- 10 • a *half-line tag data* buffer. Contains the unencoded data and TagsPrinted flags for the tags in half of a given line of tags (a line is the width of the strip printed by this PEC chip). If only part of a tag is printed by this PEC then the whole tag's data must be present.
- a *variable tag data* buffer. Contains the redundancy encoded variable data for a single tag.

The data structures are described in more detail below. Note that the sizes of the various structures are based on the tag encoding parameters as listed in Table 5. For different sets of encoding parameters the sizes of the structures and the corresponding number of address bits should be appropriately changed.

20 The TE supports both landscape and portrait printing. The mode is completely independent of the length of the printhead connected to the PEC. Given correct paper feed, a 12 inch printhead can print Letter and A4 pages in both landscape and portrait, and multiple PEC chips can be combined to produce arbitrary sized pages. As a result the TE contains a flag to determine the orientation of the tags.

TEOrientation is therefore a 1 bit flag with values as shown in Table 8.

Table 8. TEOrientation Register Values

Value	Description
0	Landscape
1	Portrait

25 Each 10-bit entry is interpreted independently as described by Table 9, and has no reliance on state information. This is important so that random access to the entries is possible, especially during the rendering of one side of a partial tag (spread over 2 PECs).

Table 9. Interpretation of 10-bit entry in Tag Line Structure

bit 9	Interpretation
0	This dot is part of the basic tag structure. Bit 8 contains the dot output value. The remaining 8 bits are reserved and should be set to 0.
1	This dot is derived from the data part of the tag. The lower 9 bits are used to determine the actual data bit to use. If the upper 2 bits of the address are set, the remaining 7 bits are used to address the 120 bits of encoded fixed data for the page. If the upper 2 bits of the address are not both set, the full 9 bit address is used to address the 360 bits of encoded variable data for the tag.

30

Since the Tag Format Structure (TFS) is line based, we have two such structures stored in the external DRAM - one for portrait orientation, and one for landscape orientation printing. The TBOrientation flag determines which of the two will be used. The two tag format structures are supplied by an external process, are stored in the external DRAM, and therefore can be arbitrarily different. In practice however, they are the same tag rotated through 90 degrees. The total memory required by a single TFS is $3840 \times \text{TagHeight}$ bits. The maximum amount of memory required is for a tag of height 384, and totals 180 KBytes. A maximum total of 360 KBytes is therefore required for the two orientations.

As seen in Figure 55, for a given line of dots, all the tags on that line correspond to the same tag line structure. Consequently, for a given line of output dots, a single tag line structure is required, and not the entire TFS. Double buffering allows the next tag line structure to be fetched from the TFS in DRAM while the existing tag line structure is used to render the current tag line. Reading a line of tag structure data consequently consumes the same DRAM bandwidth regardless of the orientation. The entire TFS might be stored on the PEC chip, in which case the rotation could be performed on-the-fly. The memory requirements for the TFS is therefore a double buffered tag line structure on chip (totalling $3840 \text{ bits} \times 2 = 7,680 \text{ bits}$, or 960 bytes), and up to 360 KBytes in the external DRAM for the portrait TFS and landscape TFS. In terms of bandwidth, the writing of the portrait TFS and landscape TFS only has to be done once, so is not an issue. Reading the appropriate TFS during printing however, is an issue. Assuming a worst case of adjacent tags, there is a need to read a tag line structure each output line. Each tag line structure is 480 bytes. For a maximum print speed of 30,000 lines per second, the TFS access amounts to 13.8 MB/sec.

The fixed tag data buffer is a 120-bit data buffer, addressed by 7 bits. The buffer holds the encoded fixed component of the tag data for the page. The fixed tag data buffer is written to once per page either directly from the 120 bits of original fixed data input or after the lower 40 bits of original fixed data has been Reed-Solomon encoded.

A *TagsPrinted* flag specifies whether or not a particular tag should be printed. Only a single bit, this flag is double buffered for a total of 2 bits. Double buffering allows the *TagsPrinted* flag for the next tag to be determined while the current tag is being rendered. *TagsPrinted* is therefore a 1 bit flag with values as shown in Table 10.

Table 10. *TagsPrinted* register values

25

Value	Description
0	Don't print the tag. Ignore the TFS as well as tag fixed and variable data values. Output 0 for each dot within the tag bounding box.
1	Print the tag as specified by the various tag structures.

A Half-Line Tag Data Buffer contains the unencoded variable tag data for up to a half the tags on a line. Since each line can contain a maximum of 152 tags (a tag size of $2\text{mm} \times 2\text{mm}$ closely packed over a length of 12 inches), each half-line tag buffer contains at most 76 tags. 128 bits are allocated to each tag 495, 496, 497 and so on as shown in Figure 55: 120 bits of unencoded data 501, a 1 bit *TagsPrinted* flag 498, a 1 bit *LastTagOfHalfLine* flag 499, and 6 reserved bits (set to 0) 500. The size of a single buffer is therefore 9728 bits (1216 bytes). The allocation of 1 bit to the *TagsPrinted* flag 498, instead of having a magic value for the unencoded data (eg 0) means that the unencoded data is 120 bits of completely unrestricted data.

Rather than double buffering an entire line of tag data, we triple buffer a half line of tag data. This saves 1216 bytes (compared to the double buffered full tag line), but comes with a timing restriction in that instead of having an entire line time to read the full tag line, the half-line tag data must be read in half a dotline. Note that it is important to have three half-line buffers and not just two. With only two half-line buffers the same tag data needs to be re-read as a given set of

35

WO 01/89838

PCT/AU00/00517

- 62 -

tags extends over multiple dot lines. The triple buffer allows the same two half-line tag buffers to be used multiple times (once for each line of the tag) without having to be re-read from DRAM. The third half-line tag buffer is used to load the first half of the next tag-line's data, during the processing of the current set of tags, and is used to load the second half of the next tag-line's data during the processing of the first half of the next tag-line. Note that a given tag-line's data is only read once during the entire print process. Consequently a 1-bit FirstTimeProcessed flag is associated with each half-line buffer to specify if the tags on this half-line have been processed before. The first time a given half-line is processed, the next half-line buffer is loaded from DRAM.

The tag data is arranged in DRAM in terms of half-lines. If there are N tags on a given line, each half-line stored in DRAM contains the data for N/2 tags. If N is odd, one of the half-lines will contain 1 less tag than the other. The LastTagOffHalfLine flag will be set in tag N/2 for one half line, and tag (N/2 -1) for the other. Regardless, the offset from one tag half-line to the next is the same in both cases. Portrait and landscape pages balance each other out in terms of total number of tags. Assuming a worst case of adjacent 2mm x 2mm tags, there are 76 tags per half-line, and for an 8.5 inch long page there are 107 tags in the line dimension. The size of the entire data in DRAM is therefore $1216 \times 2 \times 108 = 255$ KBytes. For a print speed of 1 page per second, the bandwidth to DRAM is therefore 255 KB/sec. For a maximum print speed of 30,000 lines per second, the TFS access amounts to approximately 561 KBytes/sec.

The variable tag data buffer holds the 360 bits of encoded variable data for a single tag. The TE double buffers the variable tag data buffer for a total of 720 bits. Double buffering allows the raw 120 bits of variable data for the next tag to be redundancy encoded (if required) and stored in one variable tag data buffer while the other is being used to generate dots for the current tag. Note that if the variable tag data is not encoded by PEC, only the first 120 bits of the 360 variable data bits are valid, and it is the responsibility of the external page provider to ensure that the 120 bits of variable tag data have appropriate redundancy encoding already applied.

The variable tag data buffer shown in FIG. 58 is the current tag's variable data. While the dots for the current tag are being produced, the variable data for the next tag is being encoded to a second variable tag data buffer, as shown in FIG. 59.

Rather than store the entire tag format structure, or the variable tag data for all tags, the data is loaded from the external DRAM in a just-in-time way. Appropriate trade-offs are made between buffer sizes and transfer bandwidth. Processing ahead to ensure data is available just in time works occurs in both the dot and line directions.

- As the dots for one tag are being generated in the dot direction, the variable data component for the next tag are being redundancy encoded into a second variable data buffer, and the next tag's TagIsPrinted flag is being determined. Both of these tasks involve reading the from the half-line tag data buffer, and does not involve an access to the external DRAM.

- The first time a half-line tag data buffer is used, the next half-line of unencoded tag data is fetched from DRAM. Nothing is read from DRAM when a half-line of tag data is used again. Since there are 3 half-line tag buffers, two buffers can be used multiple times for a single line of tags while the data for the next half-line of tags is ready. Note that this allows each tag's unencoded data to be read from DRAM only once.

- While the dots for one line of tags are being produced, the next line of the tag format structure is read from external DRAM. This is only required if the current output line is actually part of a tag. In the case of the last line of a tag the first line of the tag is reread. Nothing is read while processing an inter-tag line.

Table 11 summarizes the memory requirements for the TE, both in terms of on-chip and off-chip (external DRAM) requirements.

WO 01/89838

PCT/AU00/00517

- 63 -

Table 11. TE memory requirements

Name	On Chip Total Requirements	Off Chip worst case (external DRAM)
TEOrientation	1 bit	-
Tag Format Structure	360 bytes	360 KBytes (total)
Fixed tag data buffer	120 bits	-
TagsPrinted flag	2 bits	-
Half-line tag data buffers	3648 bytes	255 KBytes (per page)
Variable tag data buffer	720 bits	-
TOTAL	5018 bytes	-

At the highest level, a state machine in the TE steps through the output lines of a page one line at a time, with the starting position either in an inter-tag gap or in a tag (a PEC may be only printing part of a tag due to multiple PECs printing a single line). If the current position is within an inter-tag gap, an output of 0 is generated. If the current position is within a tag, the tag format structure is used to determine the value of the output dot, using the appropriate encoded data bit from the fixed or variable data buffers as necessary. The TE then advances along the line of dots, moving through tags and inter-tag gaps according to the tag placement parameters. Once the entire line of output dots has been produced, the TE advances to the next line of dots, moving through tags and inter-tag gaps according to the tag placement rules for the line direction. An output dot must be generated each cycle in order to keep up with other dot generating processes in the PEC. In pseudocode, the process is as follows. Note that the logic for accessing DRAM is not shown.

```

5      If (TEOrientation = Portrait)
15         maxTagComponentLine[0] = LineInterTagGap
            maxTagComponentLine[1] = TagHeight
            maxTagComponentDot[0] = DotInterTagGap
            maxTagComponentDot[1] = TagWidth
            startDotOffset[0] = StartPosition.LocalOffsetDot
            startDotState[0] = StartPosition.TagStateDot
20         startDotOffset[1] = AltTagLinePosition.LocalOffsetDot
            startDotState[1] = AltTagLinePosition.TagStateDot
            CurrPos.TagStateLine = StartPosition.TagStateLine
            CurrPos.LocalOffsetLine = StartPosition.LocalOffsetLine
25         Else
            maxTagComponentLine[0] = DotInterTagGap
            maxTagComponentLine[1] = TagWidth
            maxTagComponentDot[0] = LineInterTagGap
            maxTagComponentDot[1] = TagHeight
            startDotOffset[0] = StartPosition.LocalOffsetLine
            startDotState[0] = StartPosition.TagStateLine
30         startDotOffset[1] = AltTagLinePosition.LocalOffsetLine

```

WO 01/89838

PCT/AU00/00517

- 64 -

```

startDotState[1] = AllTagLinePosition.TagStateLine
CurrPos.TagStateLine = StartPosition.TagStateDot
CurrPos.LocalOffsetLine = StartPosition.LocalOffsetDot
EndIf
5
Stall until the RSEncoder's output TagReady flag is set
transfer TagIsPrinted flag from RSEncoder to DotGenerator
transfer variable tag data buffer from RSEncoder to DotGenerator
send AdvanceTag signal to RSEncoder to begin encoding the next tag
10
tagLineType = 0
LineCount = 0
While (LineCount < MaxLine)
Do
    CurrPos.TagStateDot = startDotState[tagLineType]
    CurrPos.LocalOffsetDot = startDotOffset[tagLineType]
    DotCount = 0
    While (DotCount < MaxDot)
    Do
        15
        If (CurrPos.TagStateLine == inInterTagGap)
            Write 0 to FIFO
        Else
            If (CurrPos.TagStateDot == inTag)
                Write (Decode TagLineStructure[CurrPos.LocalOffsetDot]) to FIFO
            Else
                20
                Write 0 to FIFO
            EndIf
            increment CurrPos.LocalOffsetDot
            If (CurrPos.LocalOffsetDot > maxTagComponentDot[CurrPos.TagStateDot])
                CurrPos.LocalOffsetDot = 0
                CurrPos.TagStateDot = (~currPos.TagStateDot) OR
                25
                (dotInterTagGap == 0)
            If (CurrPos.TagStateDot == inTag)
                transfer TagIsPrinted flag from RSEncoder to DotGenerator
                transfer variable tag data buffer from RSEncoder to DotGenerator
                send AdvanceTag signal to RSEncoder to begin encoding the next tag
                30
                EndIf
            EndIf
            increment DotCount
        EndDo
    EndDo
    35
    increment lineCount
    40

```

WO 01/89838

PCT/AU00/00517

- 65 -

```

increment CurrPos.LocalOffsetLine
If (CurrPos.LocalOffsetLine > maxTagComponentLine(CurrPos.TagStateLine))
  CurrPos.TagStateLine = ((-currPos.TagStateLine) OR (lineInterTagGap == 0))
  CurrPos.LocalOffsetLine = 0
5   If (CurrPos.TagStateLine == inTag)
      tagLineType = ~tagLineType
    EndIf
  EndIf
EndDo
10

```

The outputting of a single bit based upon the position within the tag depends on having access to the appropriate tag line structure, the encoded fixed and variable tag data for the current tag, and the TagsPrinted flag for the current tag. Assuming that these have been appropriately loaded, and assuming the encoding parameters of Table 5, the generation of a single tag dot can be seen in FIG. 58 in block diagram form.

15 In FIG. 59 is seen a block diagram of the encoder. The TE contains a symbol-at-a-time GF(2⁴) Reed-Solomon encoder 590. Symbol size is 4 bits. Each codeword contains 15 4-bit symbols for a codeword length of 60 bits. Of the 15 symbols, 5 are original data (20 bits), and 10 are redundancy bits (40 bits). Since each tag contains 120 bits of variable original data, 6 codewords are required for a total encoded data size of 360 bits. The fixed tag data is also encoded using the same Reed-Solomon encoder. The fixed tag data is also encoded using the same Reed-Solomon encoder. The fixed data is constant over all tags for a given page (or strip of a page if multiple PECs are used), so only needs to be set up once before a print (or set of prints). The unencoded fixed data is 40 bits in length. These 40 bits are encoded to produce 120 bits. To encode the fixed data, the CPU loads the fixed data into the first 40 bits of the unencoded tag data buffer and then starts the state machine to encode two codewords. The resultant 120 bits in the variable tag data are then transferred to the fixed tag data buffer where they will stay for the printing of at least one page, and in most cases many pages. If the fixed data is not to be encoded by PEC then all 120 bits of the fixed data are copied directly to the fixed tag data buffer. The state machine 591 is responsible for producing the addressing and control signals for encoding the tag data. Table 13 shows the registers used to program the state machine 591.

The TagReady flag is cleared at 592 by the state machine 591 at startup, and subsequently whenever the AdvanceTag signal is received at 593. The flag is set once the entire set of codewords has been appropriately Reed-Solomon encoded.

30 The TagReady flag allows external users of the encoded data to stall appropriately.

To produce an encoding of 5:10 symbols, the state machine 591 gates 4-bit data at 595 from the appropriate half-line tag buffer 594 into the symbol-width Reed-Solomon decoder 590. A clock-data signal is supplied for the first 5 clocks, and the inverse of that is supplied for the next 10. This is repeated *NumberOfCodewords* times. 90 clocks are therefore required to encode the entire tag data (6 codewords × 15 clocks). A further 2 clocks are required to skip over the remaining 8 bits, thus taking the total to 92 cycles. The state machine 591 sets the TagReady flag at 592, and stalls until the Advance signal on 593 is given from the TE's high level process (the time taken for this signal to be given will depend on the width of the tag. A tag size of 92 gives a minimum delay). On the first of these last 2 clocks, a WriteEnable signal is generated on 596 so that the TagsPrinted flag 597 is set to the 1st bit of 4 read from the unencoded tag data buffer 594 (bit 121 of tag data). During the same clock, the 2nd bit of 4 is passed to the state machine. This 2nd bit, called

35 LastTagInHalfLine determines whether the tag just processed is in fact the last tag to be processed in the half-line buffer. The address generated by the state machine 591 for the half-line tag buffer 594 is 14 bits. The high 2 bits select which of

40

the 3 data buffers are addressed. The next 9 bits determine which 32-bit quantity to read from the buffer, and the lower 3 bits are used to determine which of the 8 sets of 4-bits should be selected. Of the 14 address bits, the lower 12-bit address starts at 0, and increments each cycle until it has advanced 32 times. The counter then stalls until the AdvanceTag signal on 593 comes in from the high level encoding process. If however, the LastTagInHalfLine flag is set (read as bit 122 from the latest processed tag), then the lower 12-bit address is cleared to 0, the tag half-line buffer 2-bit index is updated, and the load process for the next half-line of tag data from DRAM is potentially started.

The state machine 591 keeps a 10-bit TagLineCounter for the number of half-lines processed for this full tag line. The TagLineCounter is cleared at startup, and then incremented each time the state machine finishes encoding a tag whose LastTagInHalfLine flag is set. When the TagLineCounter is incremented, the 10 bits are used to determine the new value for the half-line index as well as potentially resetting the TagLineCounter itself. When in the first half of the line (the lowest bit of TagLineCounter = 0), the next half-line buffer will always be the second half-line of the same tag line. This simply means updating the 2-bit index. When in the second half of the line (the lowest bit of TagLineCounter = 1), the next half-line depends on whether we have finished processing this tag line or not. If we have not finished processing the tag line (the 9 highest bits of TagLineCounter don't match either TagHeight or TagWidth, depending on the value in TEOrientation), the next half-line is the same as the previous half-line. If we have finished the tag line, the next half-line comes from the next line, and therefore we use the next tag line's half-line buffer. Since we are starting a new tag line, the counter is cleared to 0 as well. Table 12 shows the relationship between old and new counter and half-line buffer indexes.

Table 12. What to do when LastTagInHalfLine is set

Lo bit of TagLineCounter	Upper bits of TagLineCounter = tag height	Current Index Value	Next Index Value	Clear Counter?
0 ¹⁰	x ¹¹	0	1	No
0	X	1	2	No
0	X	2	0	No
1 ¹²	0	0	2	No
1	0	1	0	No
1	0	2	1	No
1	1	0	1	Yes
1	1	1	2	Yes
1	1	2	0	Yes

Whenever the index value changes, the old index is kept and the FirstTimeProcessed flag for the half-line buffer associated with the new index is checked. If the FirstTimeProcessed flag is clear, nothing more is done. However, if the FirstTimeProcessed flag is set, it is cleared and the process of reading the next set of data for the next half-line from DRAM into the half-line specified by the old index is started. The FirstTimeProcessed flag for the half-line associated

¹⁰ signifies first half of line

¹¹ don't care state

¹² signifies second half of line

WO 01/89838

PCT/AU00/00517

- 67 -

with the old index is then set. The number of 32-bit words to be read from DRAM is specified by the *HalfLineSize* register, as described in Table 13. The current address for reading tag half lines is then incremented by *HalfLineSize* so that it is pointing at the next half-line to be read. This scheme causes a single half-line to be read in anticipation at the end of the page. Since the data is not sent to the page, it does not matter.

5 Table 13. Registers for manipulating tag's variable data

Parameter	Description	Typical Value
DataSymbols	The number of data symbols in an output codeword	5
RedundancySymbols	The number of redundancy symbols in an output codeword	10
NumberOfCodeWords	The number of codewords to encode	6
HalfLineSize	The number of 32-bit quantities in a half-line of variable tag data to be loaded from DRAM	304
EncodeSelect	If this bit is set, then data is Reed-Solomon encoded. If clear, the data is merely copied.	1

Conclusion

10 The present invention has been described with reference to a preferred embodiment and number of specific alternative embodiments. However, it will be appreciated by those skilled in the relevant fields that a number of other embodiments, differing from those specifically described, will also fall within the spirit and scope of the present invention. Accordingly, it will be understood that the invention is not intended to be limited to the specific embodiments described in the present specification, including documents incorporated by cross-reference as appropriate. The scope of the invention is only limited by the attached claims.

WO 01/89838

PCT/AU00/00517

- 68 -

CLAIMS

1. A printed page tag encoder comprising:
 - an input at which to receive a tag structure template;
 - an input at which to receive fixed data bits;
 - 5 an input at which to receive variable data bit records; and
 - a tag dot generator outputting single bits depending on position in the tag defined by the tag structure template and said fixed and said variable data.
2. A printed page tag encoder as claimed in claim 1 additionally comprising a redundancy encoder for optionally encoding said fixed and/or said variable data.
- 10 3. A printed page tag encoder as claimed in claim 2 wherein the redundancy encoder utilises Reed-Solomon encoding.
4. A printed page tag encoder as claimed in claim 1 wherein tags are placed regularly on a page.
5. A printed page tag encoder as claimed in claim 4 wherein tags are placed in a triangular grid.
6. A method of establishing printed page tags comprising:
 - 15 formatting fixed data for the page being printed together with specific tag data values according to a defined tag structure format; and
 - regularly locating tags on a page.
7. A method of printing a printed page tag as claimed in claim 6 further comprising redundancy encoding the fixed and/or specific tag data.
- 20 8. A print engine/controller to drive an ink drop printhead comprising:
 - a contone image decoder to decode any compressed continuous tone image planes in the received compressed page data;
 - a bi-level decoder to decode any compressed bi-level image plane in the received compressed page data;
 - a tag encoder to produce a tag image plane; and
 - 25 a halftoner/compositor including a dot merger unit controlled by a color mask to effect integration of the image planes and tag data plane.
9. A print engine/controller as claimed in claim 8 wherein tags are placed in the tag image plane on a triangular grid.
10. A print engine/controller as claimed in claim 8 wherein the tag image plane is connected to an infrared ink channel by which to place infrared ink printed tags into a printed page.
- 30 11. A print engine/controller as claimed in claim 8 wherein the tag encoder redundancy encodes the tag data destined for said tag image plane.
12. A print engine/controller as claimed in claim 11 wherein redundancy encoding uses Reed-Solomon encoding.
13. A print engine/controller as claimed in claim 11 wherein the print engine/controller encodes both the fixed and variable parts of the tag data.
- 35 14. A print engine/controller chip to interface with an ink drop printhead by which to generate tagged printed page comprising:
 - an interface at which to receive compressed page data;
 - a tag encoder to output a tag image plane;
 - a contone image decoder to decode any continuous tone image planes in the received compressed page data;
 - 40 a bi-level decoder to decode any bi-level image planes in the received compressed page data;

WO 01/89838

PCT/AU00/00517

- 69 -

- a half-toner/compositer to composite any bi-level image plane over any continuous tone image plane or tag image plane; and
a printhead driver to output the composite to a printhead.
- 5 15. An inkdrop printer to generate tagged pages driven by a print engine/controller comprising:
an interface at which to receive compressed page data;
a tag encoder to generate a tag image plane;
a contone image decoder to decode any continuous tone image planes in the received compressed page data;
a bi-level decoder to decode any bi-level image planes in the received compressed page data;
10 a half-toner/compositer to composite any bi-level image plane over any continuous tone image plane or tag image plane;
a printhead driver to output the composite to a printhead; and
a printhead.

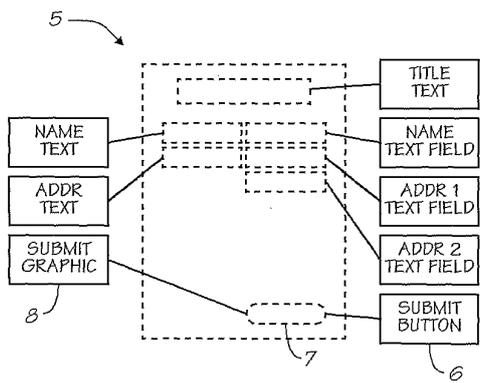
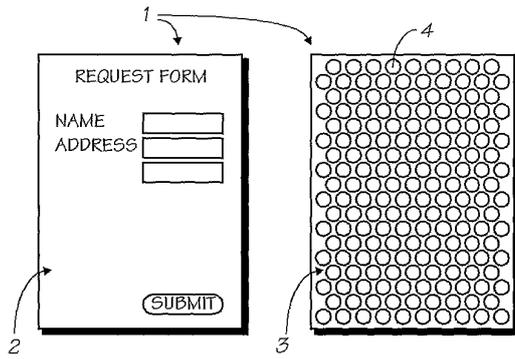


FIG. 1

2/52

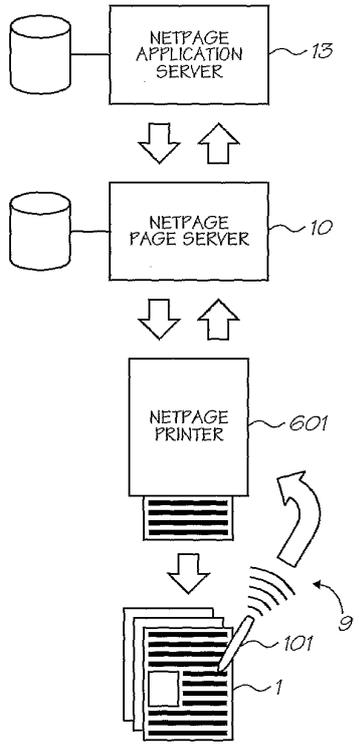


FIG. 2

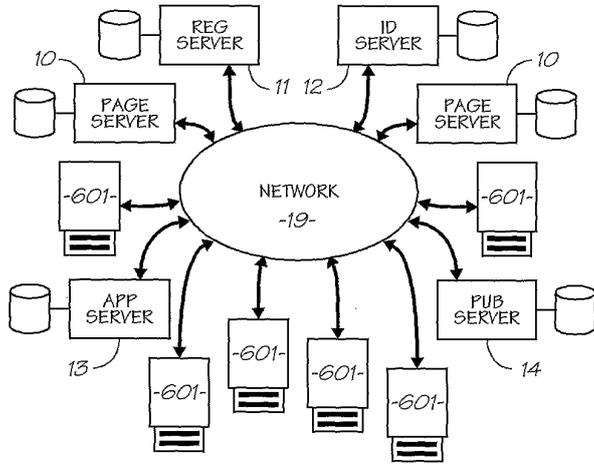


FIG. 3

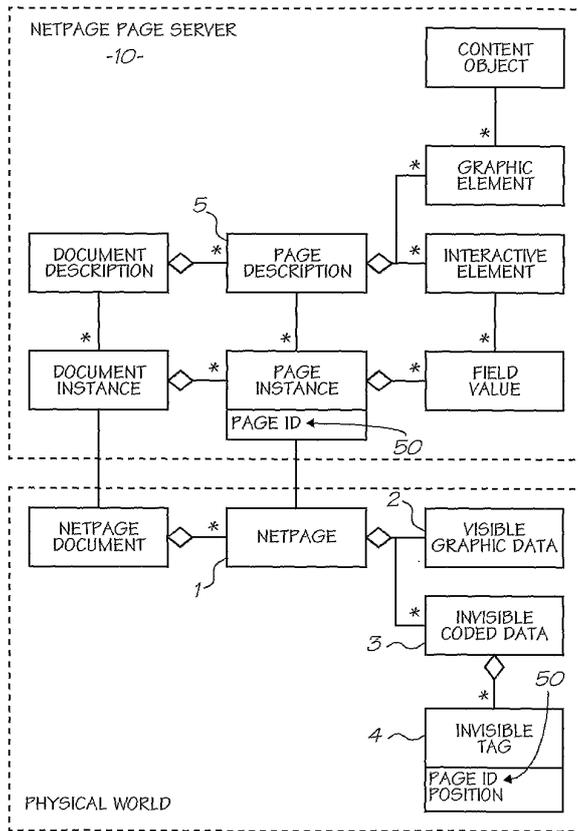


FIG. 4

WO 01/89838

PCT/AU00/00517

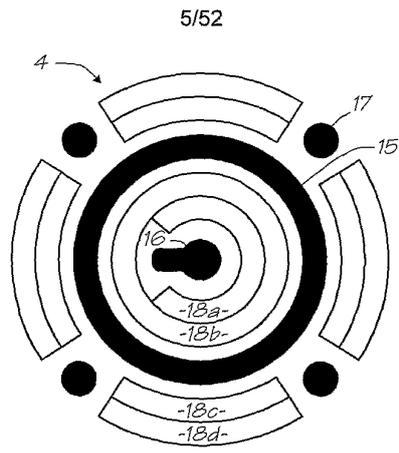


FIG. 5

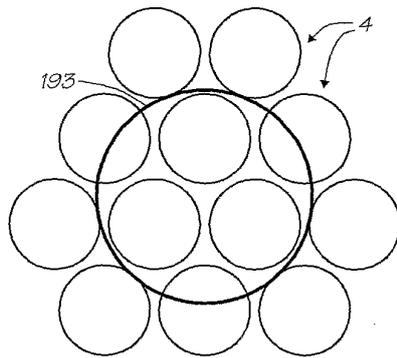


FIG. 6

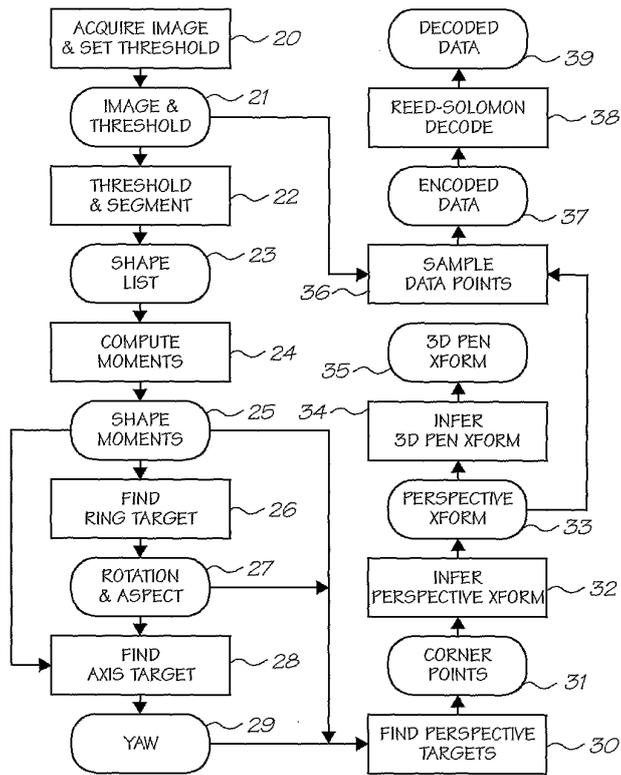


FIG. 7

7/52

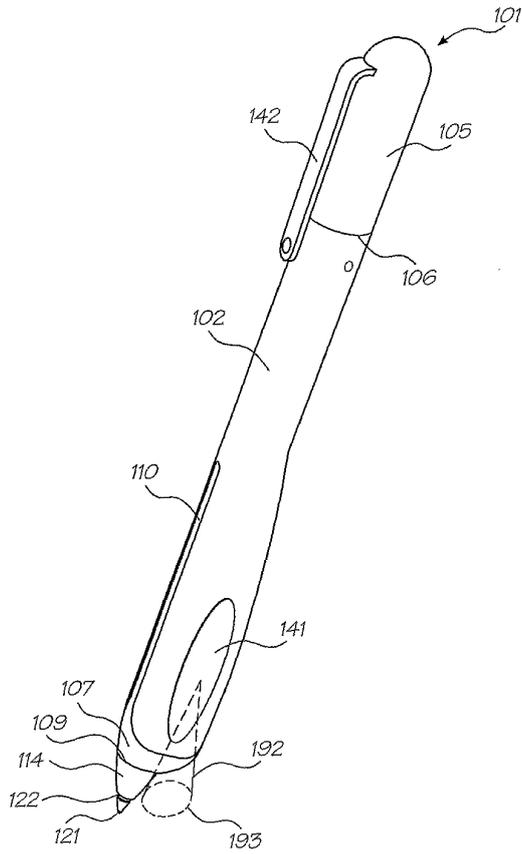


FIG. 8

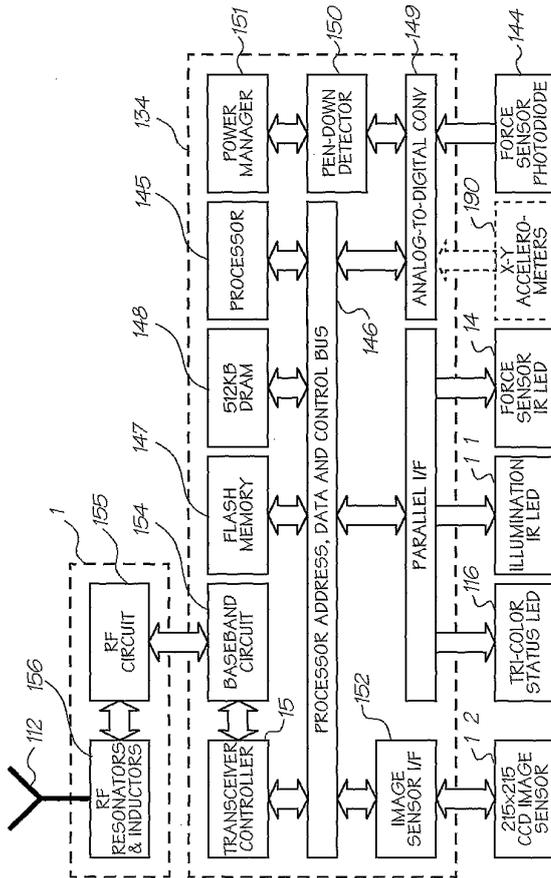


FIG. 10

WO 01/89838

PCT/AU00/00517

10/52

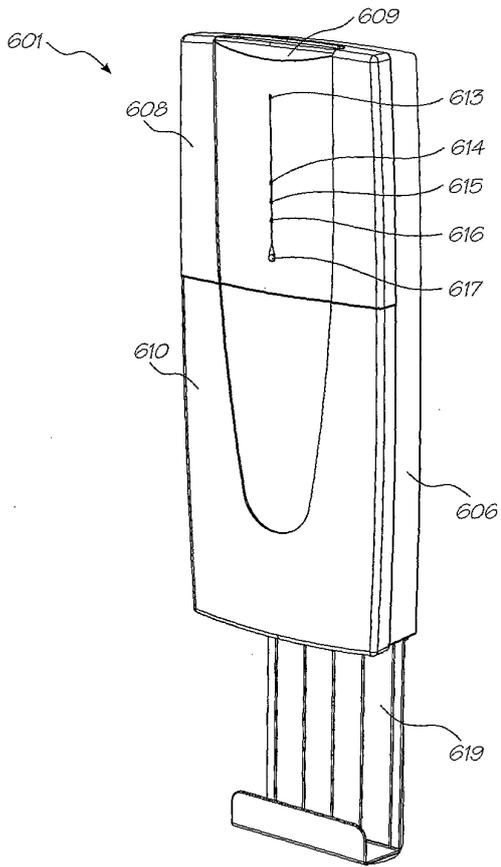


FIG. 11

11/52

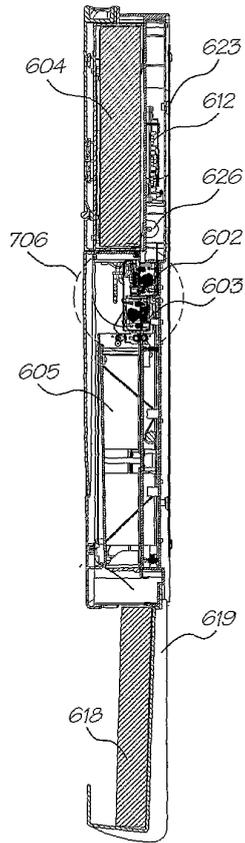


FIG. 12

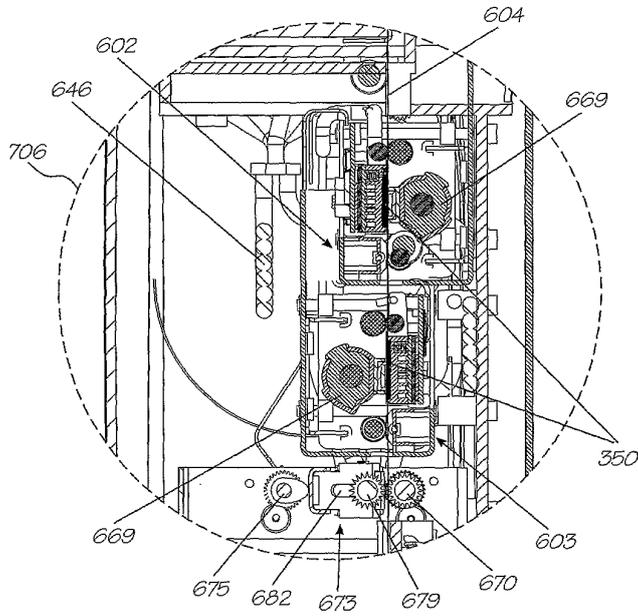


FIG. 12a

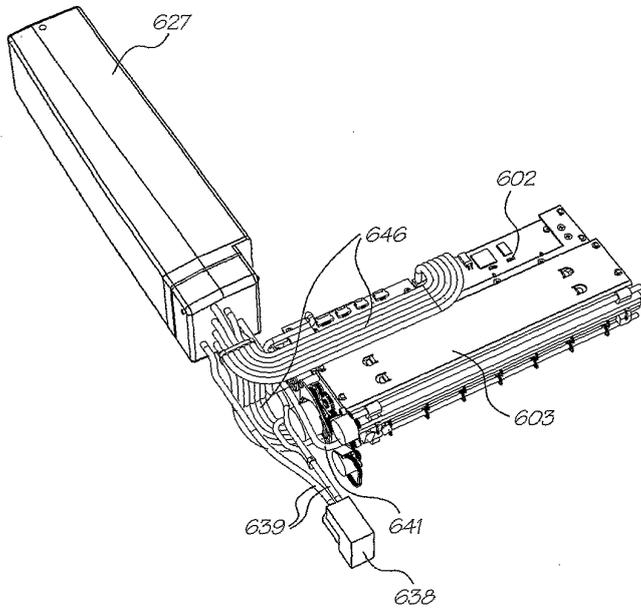


FIG. 13

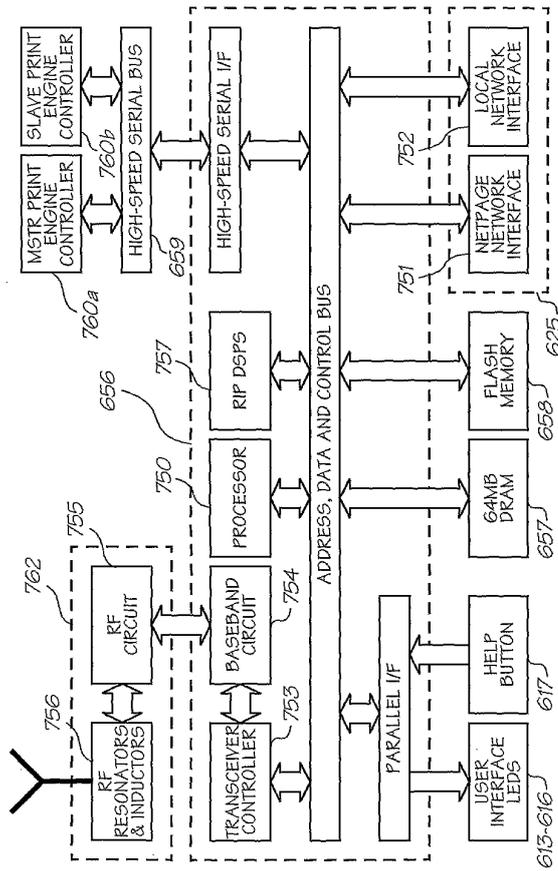


FIG. 14

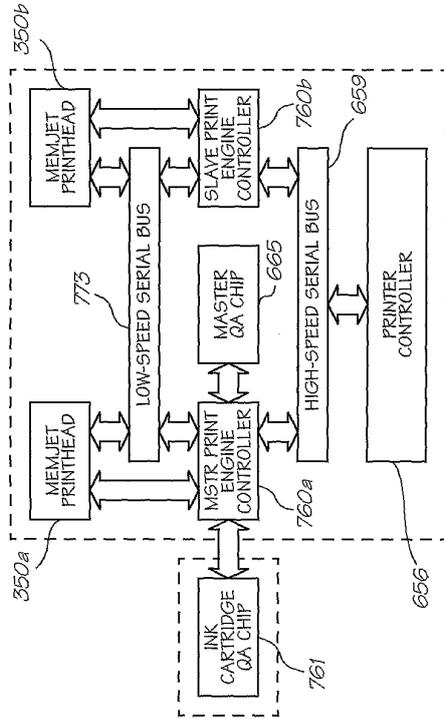


FIG. 15

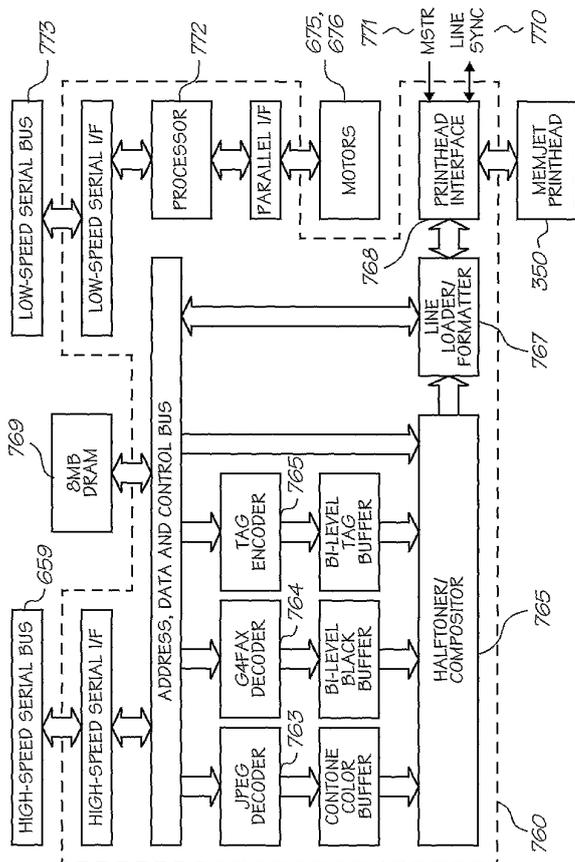


FIG. 16

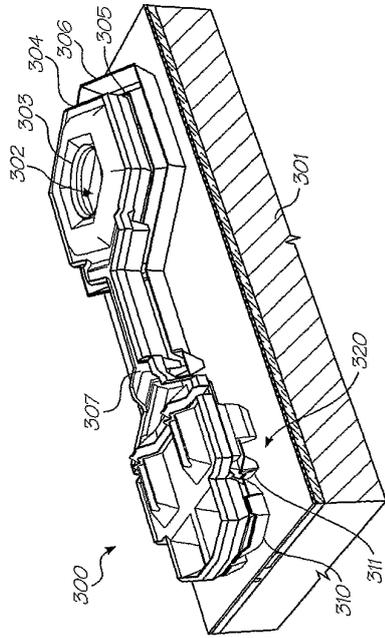


FIG. 17

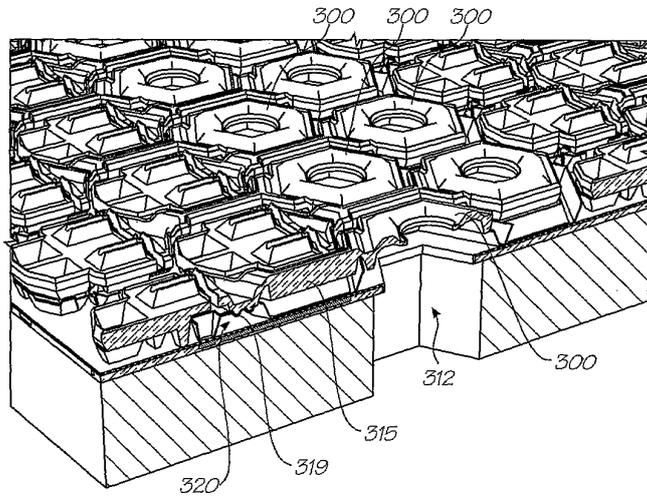


FIG. 18

19/52

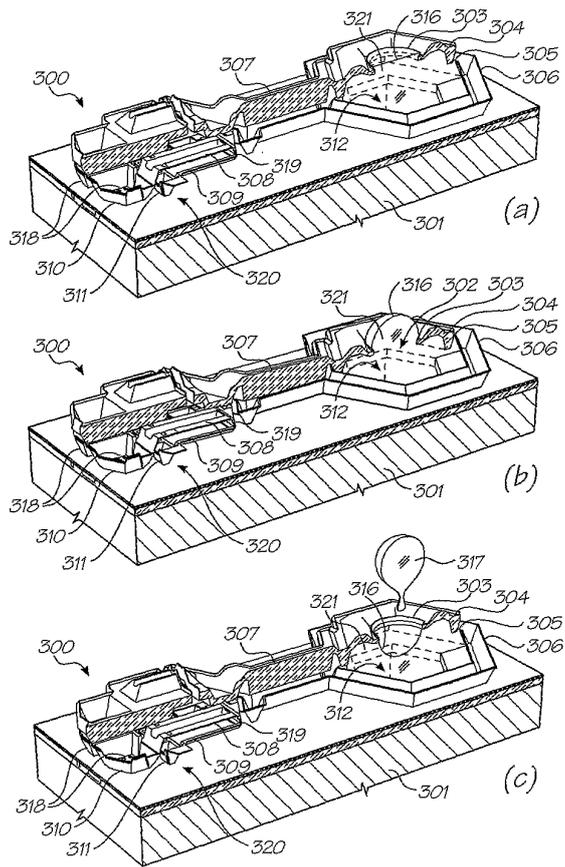


FIG. 19

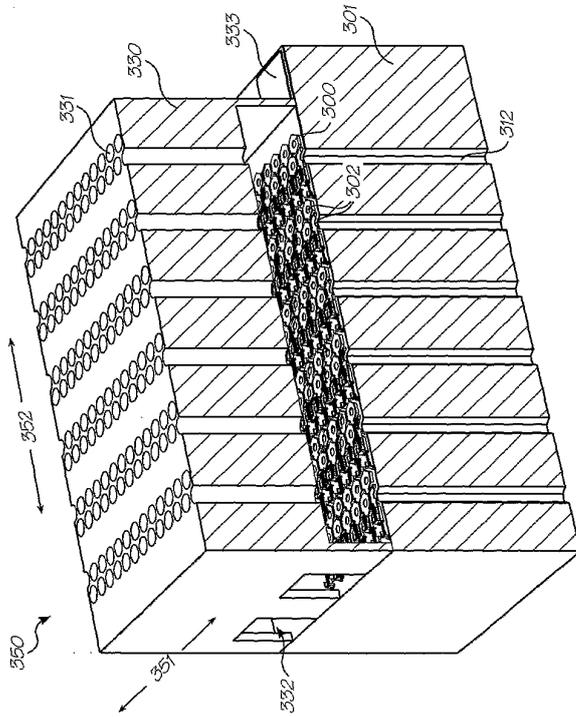


FIG. 20

21/52

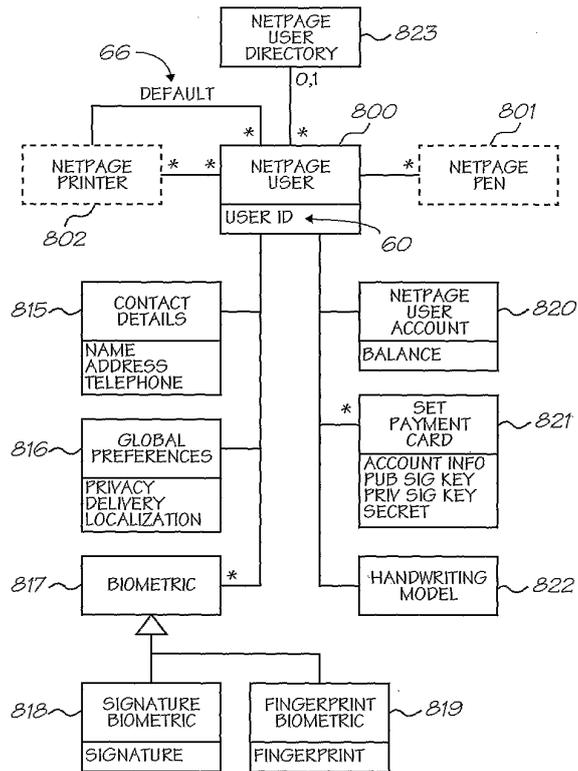


FIG. 21

WO 01/89838

PCT/AU00/00517

23/52

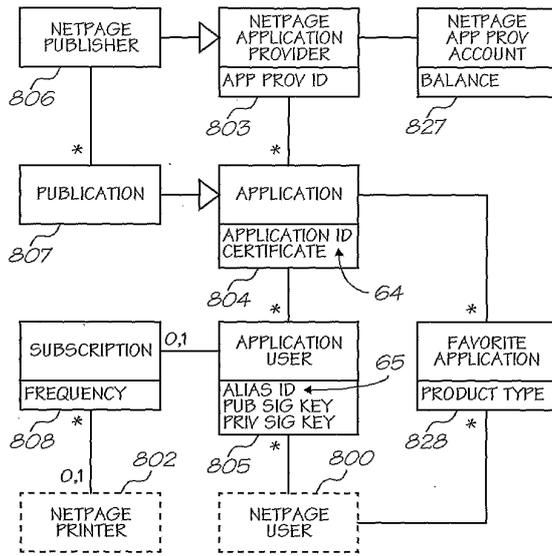


FIG. 24

WO 01/89838

PCT/AU00/00517

24/52

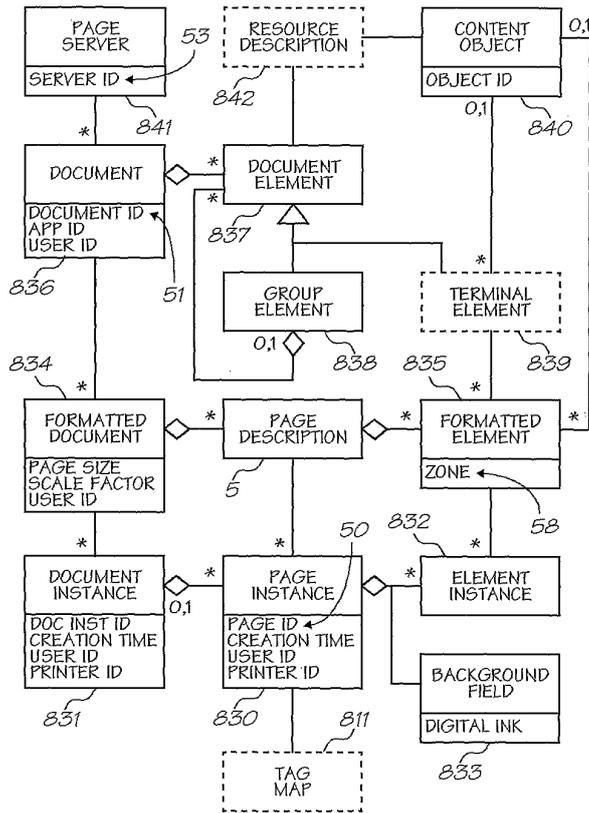


FIG. 25

25/52

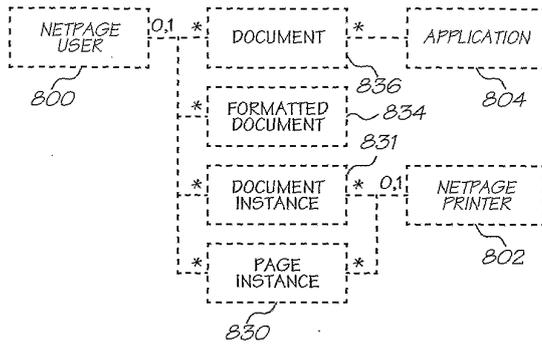


FIG. 26

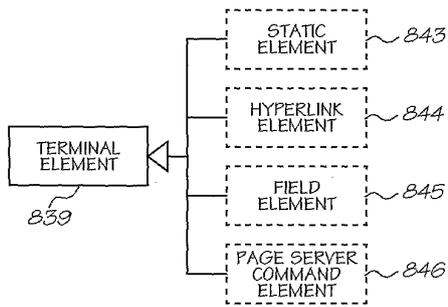


FIG. 27

26/52

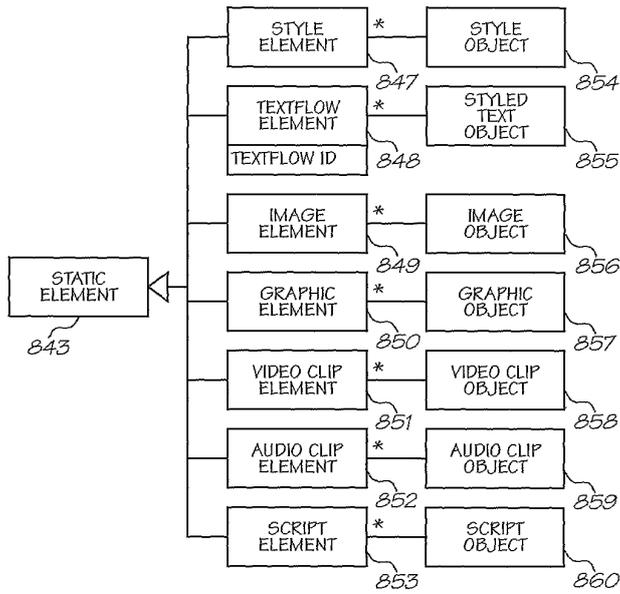


FIG. 28

27/52

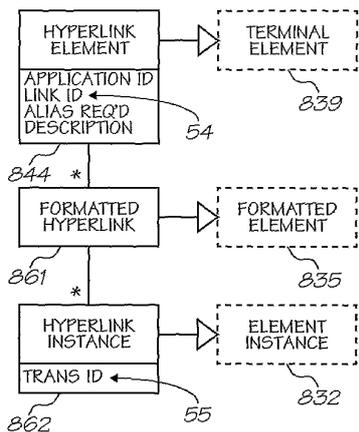


FIG. 29

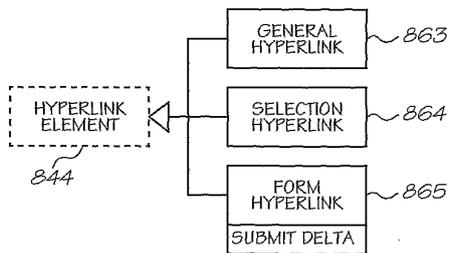


FIG. 30

WO 01/89838

PCT/AU00/00517

29/52

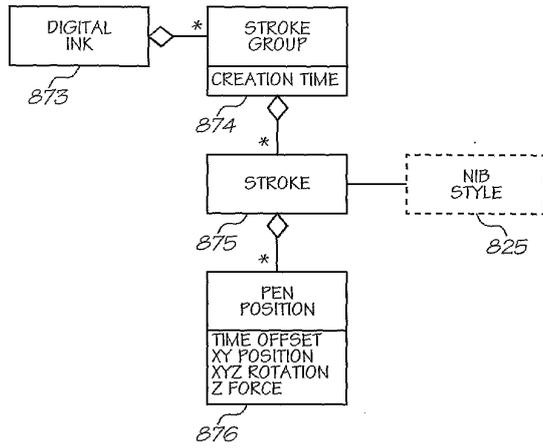


FIG. 33

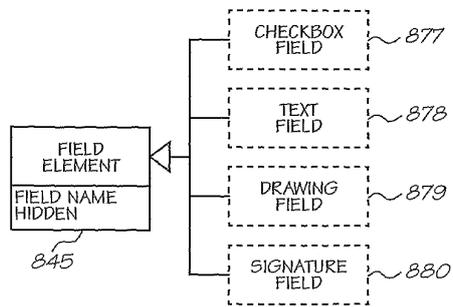


FIG. 34

30/52

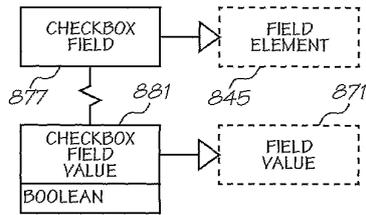


FIG. 35

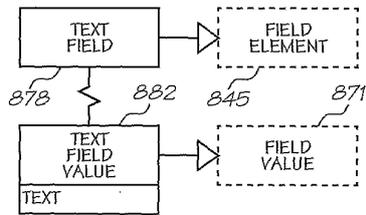


FIG. 36

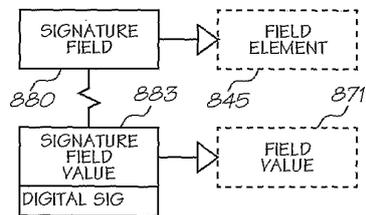


FIG. 37

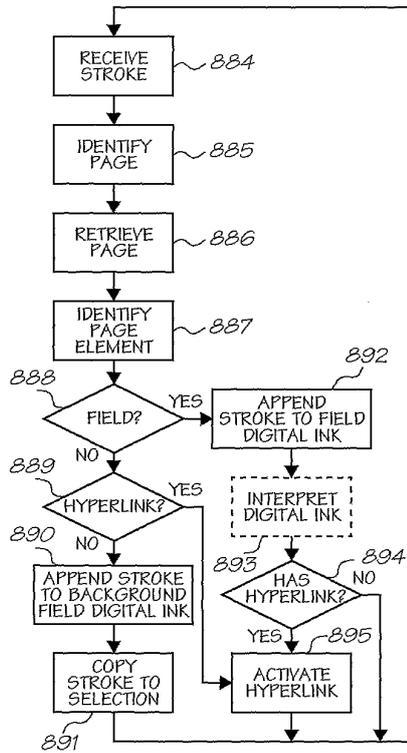


FIG. 38

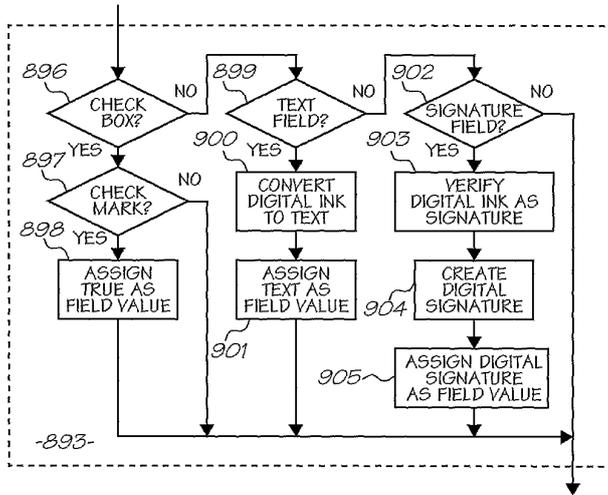


FIG. 38a

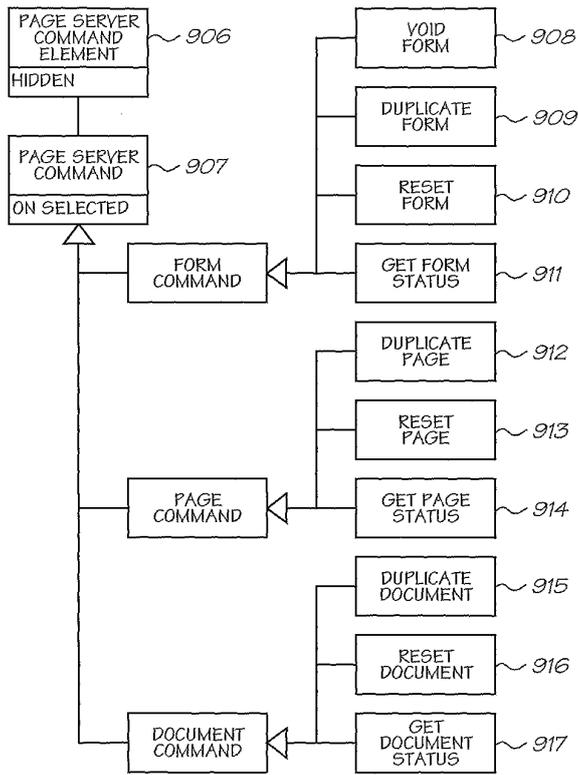


FIG. 39

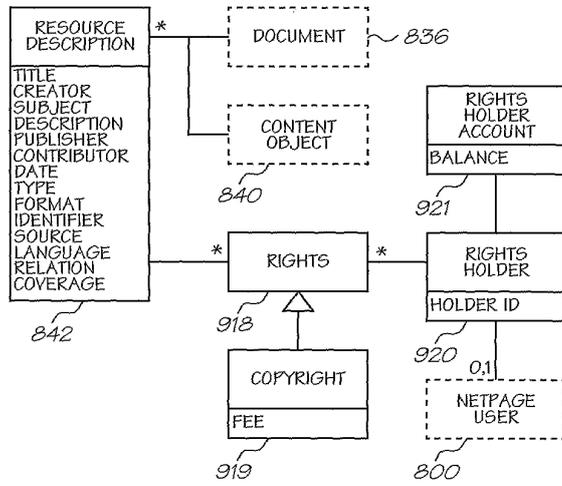


FIG. 40

WO 01/89838

PCT/AU00/00517

35/52

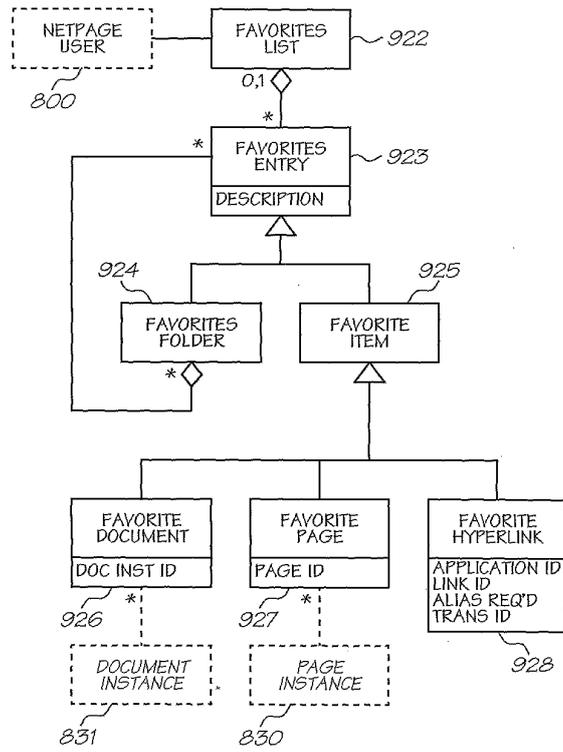


FIG. 41

WO 01/89838

PCT/AU00/00517

36/52

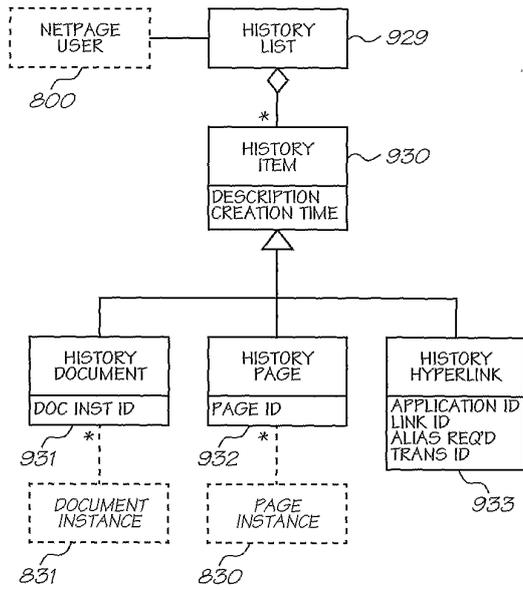


FIG. 42

37/52

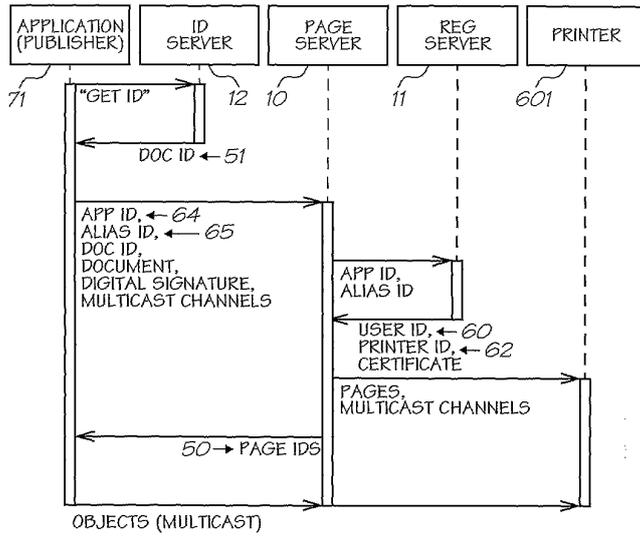


FIG. 43

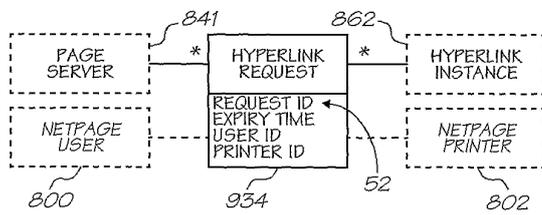


FIG. 44

38/52

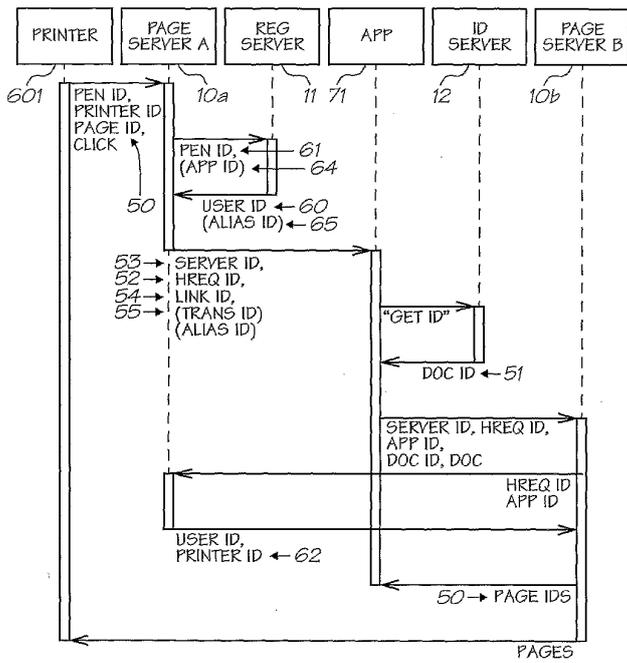


FIG. 45

39/52

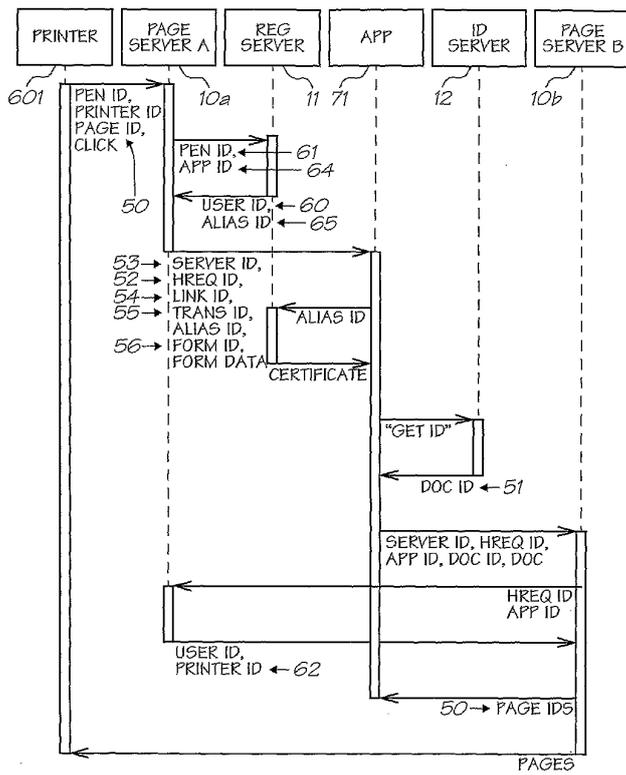


FIG. 46

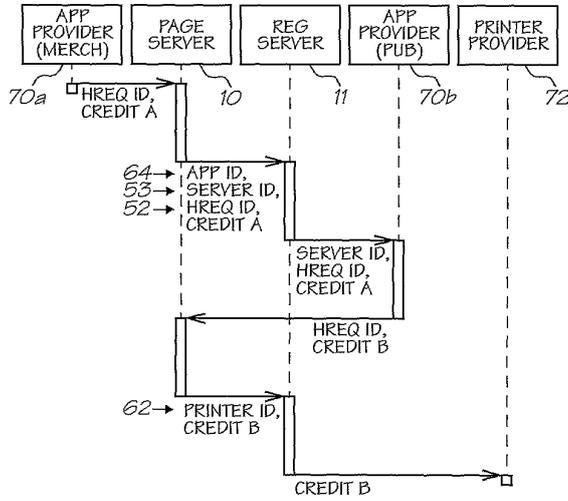


FIG. 47

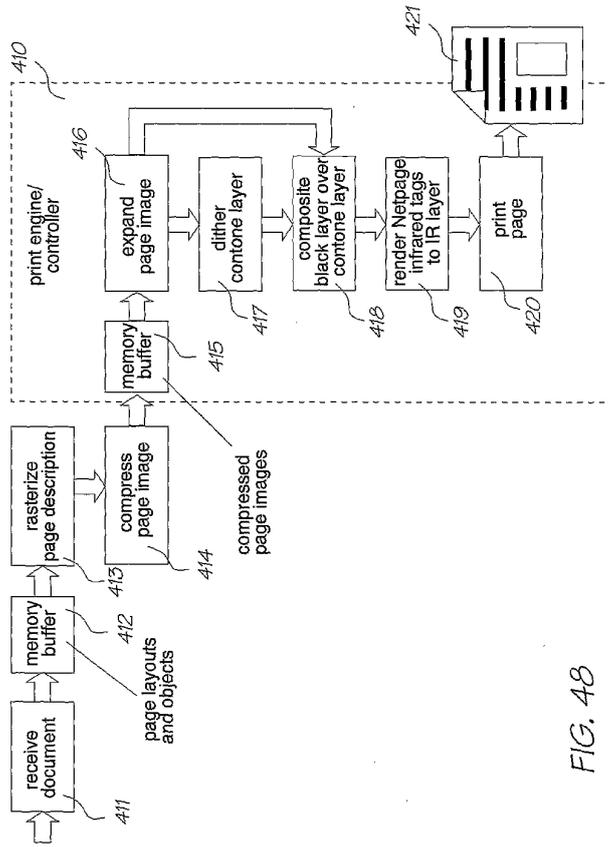


FIG. 48

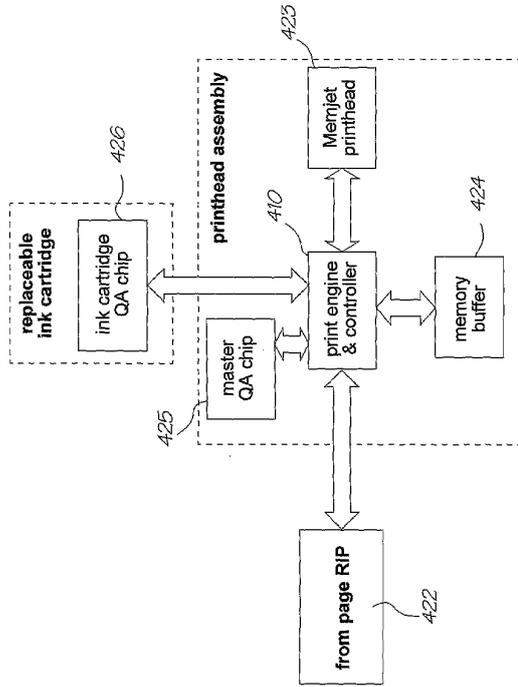


FIG. 49

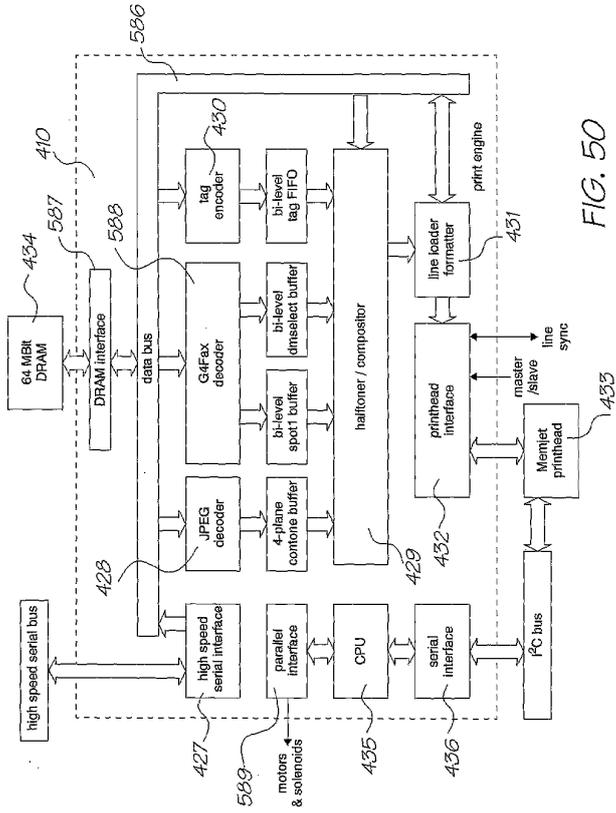


FIG. 50

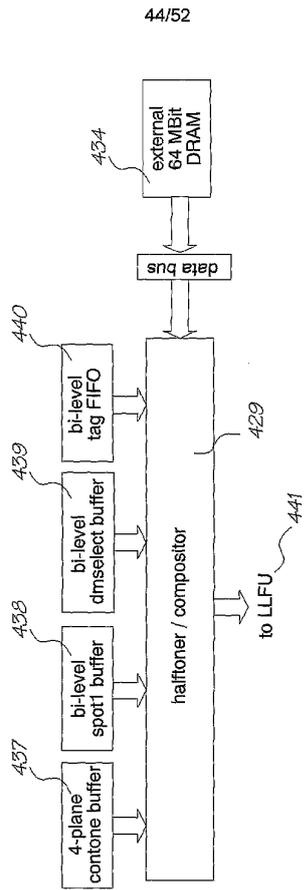


FIG. 51

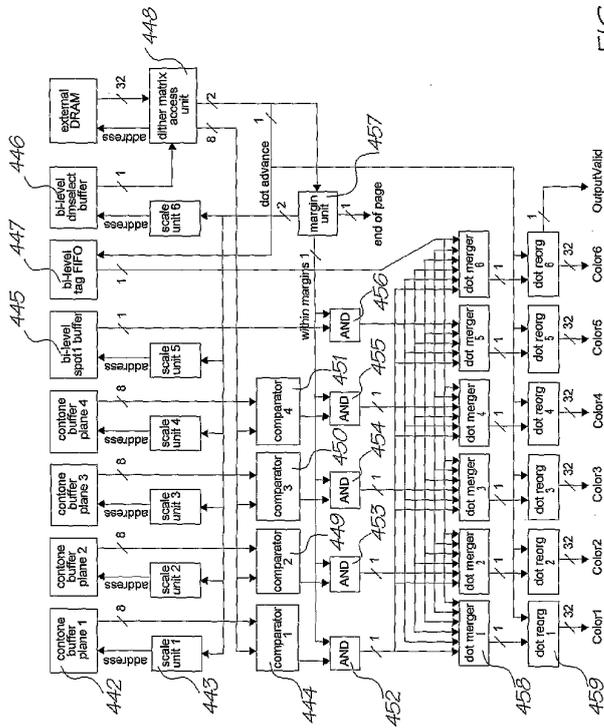


FIG. 52

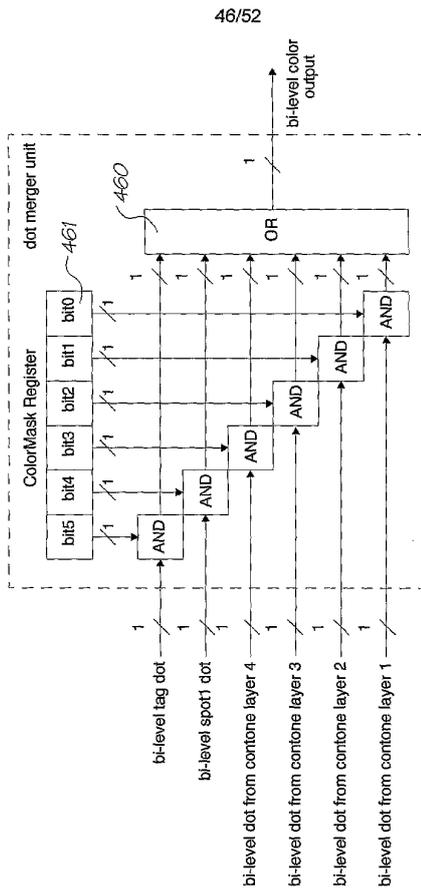


FIG. 53

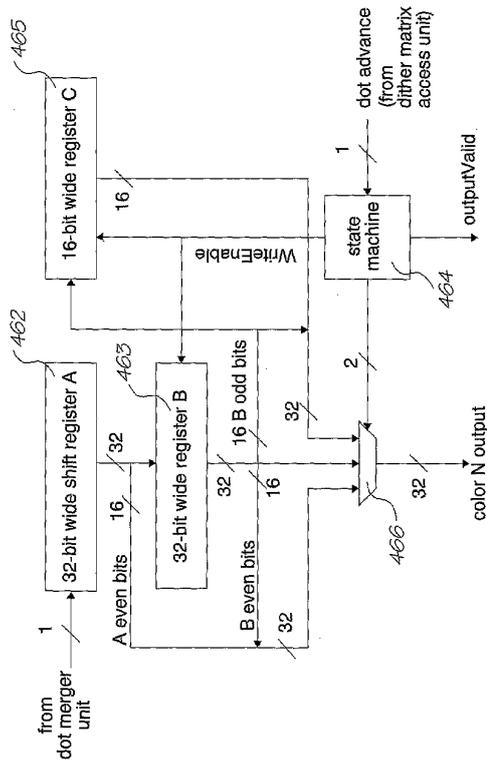


FIG. 54

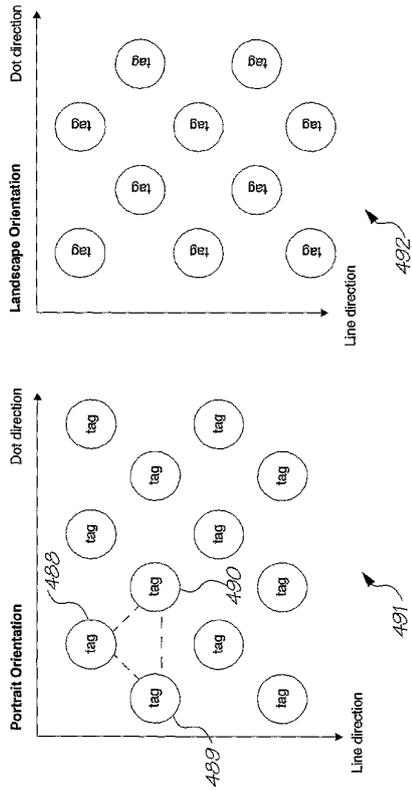


FIG. 55

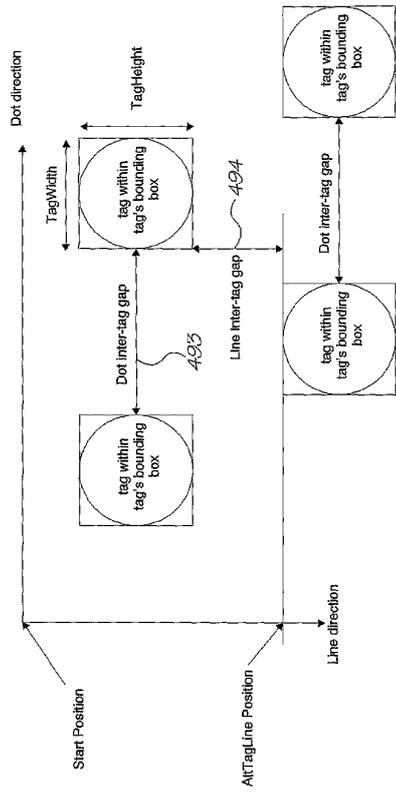


FIG. 56

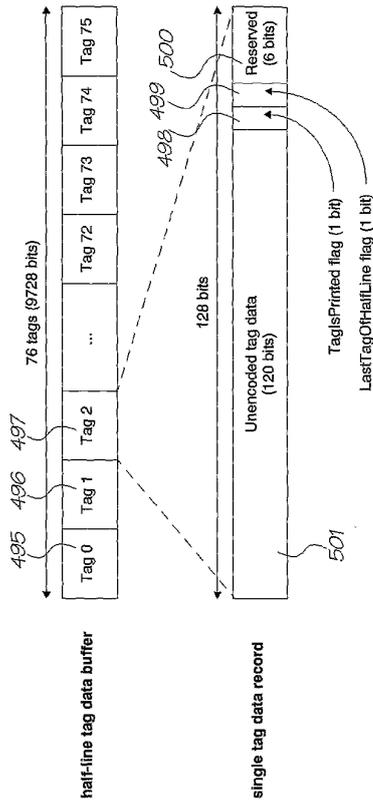


FIG. 57

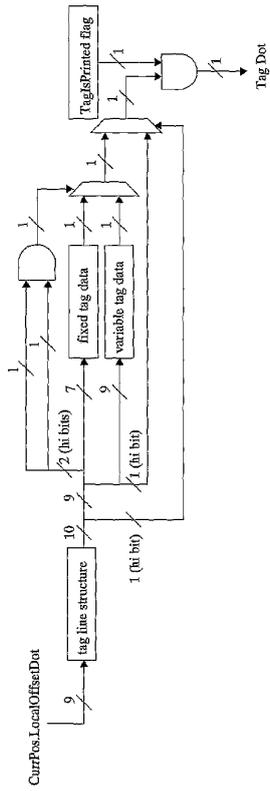


FIG. 58

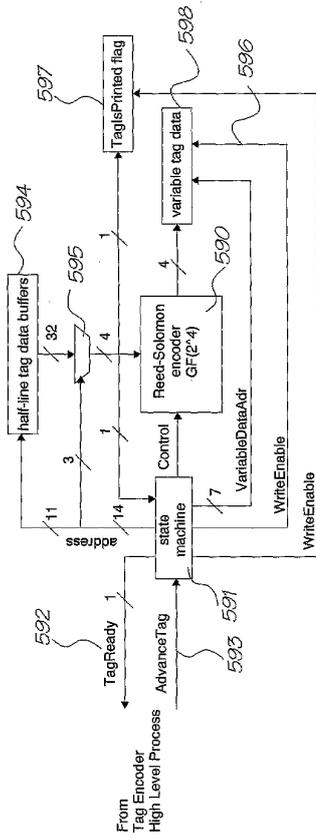


FIG. 59

【 国際調査報告 】

INTERNATIONAL SEARCH REPORT		International application No. PCT/AU00/00517
A. CLASSIFICATION OF SUBJECT MATTER		
Int. Cl. ⁷ : B41J 2/04, H04N 1/40, G06K 19/06		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED		
Minimum documentation searched (classification system followed by classification symbols) IPC: B41J, G06, H04		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) DWPI, USPTO, JAPIO		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 4754127 A (BRASS et al) 28 June 1988 . Whole document	1-7
Y	Whole document	8-15
X	WO 99/50787 A (XEROX CORPORATION) 7 October 1999 . Whole document	1-7
Y	Whole document	8-15
X	US 5661506 A (LAZZOUNI et al) 26 August 1997 . Whole document	1-7
Y	Whole document	8-15
<input checked="" type="checkbox"/> Further documents are listed in the continuation of Box C <input checked="" type="checkbox"/> See patent family annex		
* Special categories of cited documents:		
"A" document defining the general state of the art which is not considered to be of particular relevance	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention	
"E" earlier application or patent but published on or after the international filing date	"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone	
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art	
"O" document referring to an oral disclosure, use, exhibition or other means	"&" document member of the same patent family	
"P" document published prior to the international filing date but later than the priority date claimed		
Date of the actual completion of the international search 1 August 2000	Date of mailing of the international search report	
Name and mailing address of the ISA/AU AUSTRALIAN PATENT OFFICE PO BOX 200, WODEN ACT 2606, AUSTRALIA E-mail address: pct@ipaustralia.gov.au Facsimile No. (02) 6283 3929	Authorized officer GREG POWELL Telephone No: (02) 6283 2308	

INTERNATIONAL SEARCH REPORT		International application No. PCT/AU00/00517
C (Continuation), DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5652412 A (LAZZOUNI et al) 29 July 1997 · Whole document	1-7
Y	Whole document	8-15
X	US 5477012 A (SEKENDUR) 19 December 1995 · Whole document	1-7
Y	Whole document	8-15
Y	EP 932299 A (ADOBE SYSTEMS INC) 28 July 1999 · Whole document	8-15
Y	AU 94090/98 A (CANON INC et al) 17 June 1999 · Whole document	8-15
Y	WO 96/32261 A (EASTMAN KODAK CO) 17 October 1996 · Whole document	8-15
Y	WO 96/32813 A (EASTMAN KODAK CO) 17 October 1996 · Whole document	8-15

Form PCT/ISA/210 (continuation of Box C) (July 1998)

INTERNATIONAL SEARCH REPORT		International application No. PCT/AU00/00517
Box I Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)		
This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:		
1.	<input type="checkbox"/>	Claims Nos : because they relate to subject matter not required to be searched by this Authority, namely:
2.	<input type="checkbox"/>	Claims Nos : because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:
3.	<input type="checkbox"/>	Claims Nos : because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a)
Box II Observations where unity of invention is lacking (Continuation of item 3 of first sheet)		
This International Searching Authority found multiple inventions in this international application, as follows:		
1) Claims 1-7 directed to a tag encoder 2) Claims 8-15 directed to a printer		
As reasoned on the supplementary sheet.		
1.	<input checked="" type="checkbox"/>	As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims
2.	<input type="checkbox"/>	As all searchable claims could be searched without effort justifying an additional fee, this Authority did not invite payment of any additional fee.
3.	<input type="checkbox"/>	As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:
4.	<input type="checkbox"/>	No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:
Remark on Protest	<input type="checkbox"/>	The additional search fees were accompanied by the applicant's protest.
	<input type="checkbox"/>	No protest accompanied the payment of additional search fees.

INTERNATIONAL SEARCH REPORT

International application No.
PCT/AU00/00517

Supplemental Box

(To be used when the space in any of Boxes I to VIII is not sufficient)

Continuation of Box No: II

The international application does not comply with the requirements of unity of invention because it does not relate to one invention or to a group of inventions so linked as to form a single general inventive concept. In coming to this conclusion the International Searching Authority has found that there are different inventions as follows:

1. Claims 1-7 are directed to a tag encoder and method using this tag encoder wherein the tag encoder has inputs for tag structure, fixed data and variable data. The tag is generated and regularly located on a page. It is considered that this encoder comprises a first "special technical feature".
2. Claims 8-15 are directed to a print engine/controller/printer having a contone image decoder, a bi-level decoder, a tag encoder and a halftoner/compositor to merge image and tags onto a printed page. It is considered that this print engine/controller/printer comprises a second "special technical feature".

These groups are not so linked as to form a single general inventive concept. That is, they do not have any common inventive features which define a contribution over the prior art. The common feature linking together these groups of claims is a tag encoder. However, while each set of claims contains a tag encoder, the tag encoder of claims 8-15 is not necessarily the tag encoder of claims 1-7. In addition, tag encoders are found in many documents, of which US 4754127 (cited in the ISR) is an example. Therefore these claims lack unity a posteriori.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/AU00/00517

This Annex lists the known "A" publication level patent family members relating to the patent documents cited in the above-mentioned international search report. The Australian Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

Patent Document Cited in Search Report		Patent Family Member			
US	4754127	US	4728783		
WO	9950787	WO	9950736	WO	9950751
US	5661506	NONE			
US	5652412	NONE			
US	5477012	AU	45298/96	WO	9722959
		EP	892971	US	5852434
EP	932299	JP	11263044		
AU	94090/98	NONE			
WO	9632261	EP	765222	US	5914737
WO	9632813	EP	765570	US	5909227
END OF ANNEX					

フロントページの続き

(特許庁注：以下のものは登録商標)

イーサネット

(72)発明者 ウォルムズリー， シモン， ロバート

オーストラリア， ニュー サウス ウェールズ州， エッピング， ペンブローク ストリート
9， ユニット 3

Fターム(参考) 2C187 AC07 AC08 AE13 AF03 DB04 DD03 FB08 FC01

5B021 AA01 LG00

5B068 BB18 BC05 BD09 BD16 BE08 CC06 CD02