

NEC μ PD7220/72120 Related Public Documents

- [ISSCC](#) (International Solid-State Circuit Conference)
- [\$\mu\$ PD7220A User's Manual](#)
- Nikkei Electronics Magazine ([\$\mu\$ PD7220](#))
- Transistor Gijutsu (Technology) Magazine ([\$\mu\$ PD7220](#))
- Transistor Gijutsu (Technology) Magazine ([\$\mu\$ PD7220A](#))
- Nikkei Electronics Magazine ([\$\mu\$ PD72120](#))
- [\$\mu\$ PD72120 User's Manual](#)

Go to <https://www.oguchi-rd.com/LSI%20products.php> to get more detailed NEC μ PD7220/72120 related information such as;

"Logic Schematics", "Design Notes", "Evaluation Board Schematics", "Evaluation Software", "Silicon Die Photos", "Newspaper", "Magazine", and so forth.

Go to <https://www.oguchi-rd.com/patents.php> to get patent information including NEC μ PD7220/72120 related patents.

μPD7220Aを使った グラフィック・ディスプレイ・コントローラの製作

小口哲司 / 高橋欣悟 / 樋口三左男

本誌 '83年1月号で、グラフィック・ディスプレイ・コントローラ μPD7220を紹介しました。その後このLSIを使用したボードや応用装置が、いろいろ発表されています。

今回は、μPD7220の機能強化バージョンμPD7220Aの内容も含めて、μPD7220を上手に使いこなすためのテクニックについて、具体的な応用例を混じえて紹介します。

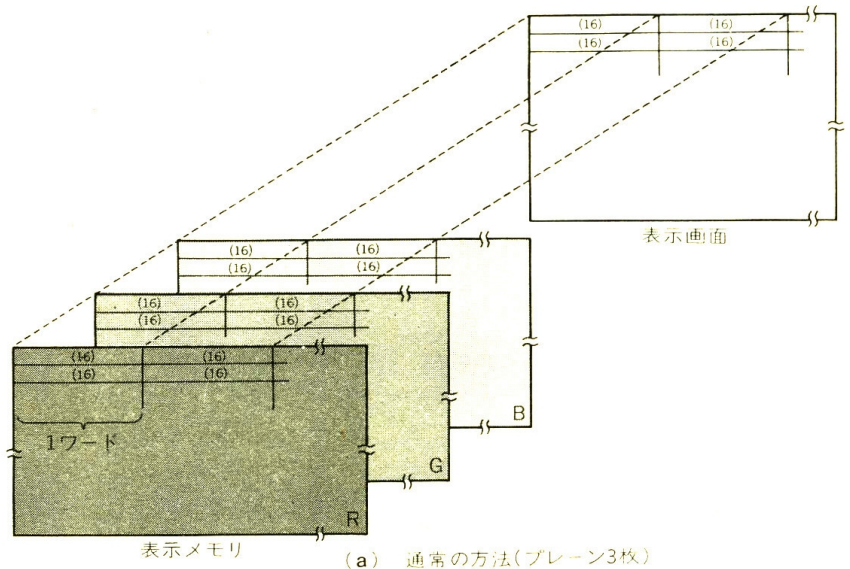
まず基本的なテクニックとして、RGB同時描画を紹介し、つぎに色選択法、塗りつぶし法、表示メモリのサイクル・スチール法について述べます。

なお、μPD7220Aの各種コマンドについては p.357より付録としてまとめて掲載してありますので、それを参照してください。

市販されているグラフィックス表示/描画機能をもつパーソナル・コンピュータのなかで、RGBの各プレーンに対して同時に描画動作に入れるものは見あたりません。同時描画を行わせるには、ハードウェアとソ

フトウェアにまたがるなんらかの特殊な細工が必要となるからです。

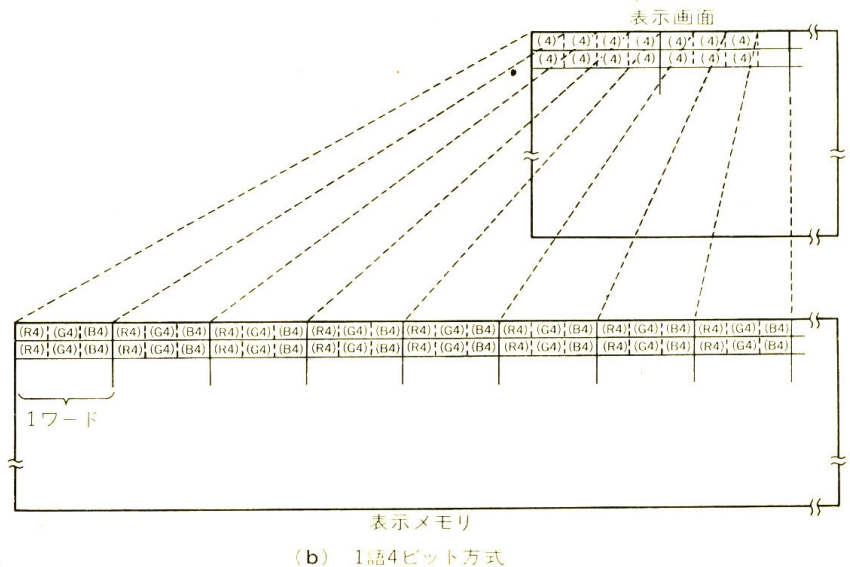
描画用プロセッサとして、μPD7220Aを使用した場合についても同様であり、一般的なメモリ構成法ととり「白」を描画するには、どのよ



RGB同時描画

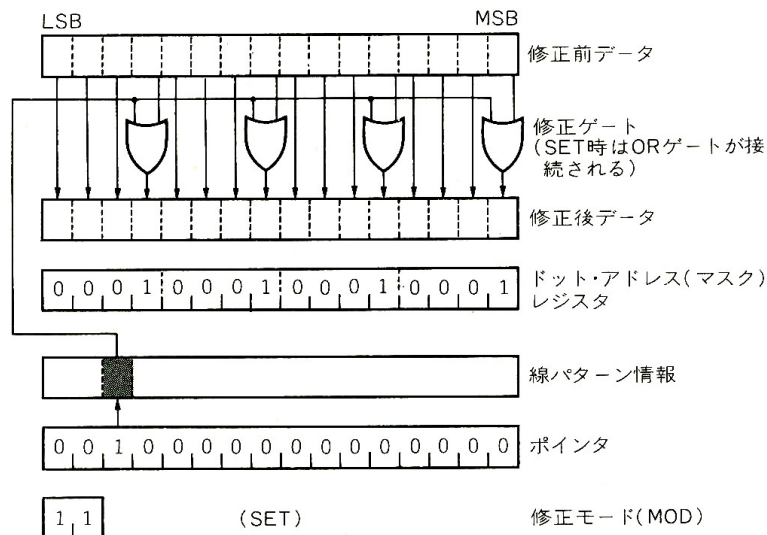
●カラーRGB同時描画の方法

カラー・グラフィックス表示では、通常、RGB計3枚のメモリ・プレーンをもち、表示時にはRGBのプレーンを同時に読み出して映像直列信号に変換したのち、モニタ・テレビに送出します(図1)。現在、一般に



〈図1〉
表示メモリと画面との対応づけ

〈図3〉 μ PD7220Aの描画実行回路

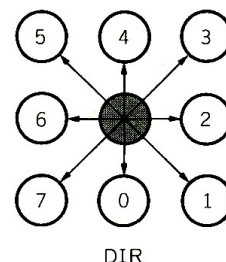


うな色選択法(色選択法の項を参照)の場合であっても、RGB各色プレーンに対して同一の描画を計3回も繰り返す必要があります。

表示メモリに対する書き込み動作であるRGB同時描画ができなければ、当然のことながら、表示メモリの読み出しについてもRGB同時に

実行できないことになります。すなわち、同一画面位置に表示されるデータであっても、その格納アドレスは異なるため、任意閉領域の塗りつぶしの際に行う境界点サーチなどを行うときには、異なるRGB 3種類のアドレスのデータを読み出す必要が生じます。

〈図4〉 描画方向の定義



しかし、RGBの3種類のデータを同一アドレス内に格納するメモリ構成をとれば、1回の読み出しで、例えば、4画素分のデータを得ることができ、連続したアドレスに対するデータ読み出しという単純な操作だけでよいことになります。さらに、ハードウェア上、各16ビットの入出力端子をもつRGB 3面のプレーンに分割する必要がなくなるため、64Kビットや256Kビットなどの高集積度のDRAMを使用でき、その結果、高密度実装が実現できることになります。

μ PD7220のアップ・バージョン μ PD7220A

μ PD7220と μ PD7220Aとの相違点について述べます。

μ PD7220Aは μ PD7220のアップ・バージョンのため、そのまま差し替えができます。

- (1)スレープ・モード時での表示制御、外部同期制御の自由度が高くなった。
- (2)PITCHレジスタが9ビットに拡張された。

(3)つぎに示す5種類のフラグを「1」にすることにより、以下の動作が可能。

- ①DS; ライト・ペン入力端子を最低4クロック間、高レベルに保つことにより、 μ PD7220Aが描画実行中の場合であっても、その描画アドレス出力を一時的に停止させ、表示アドレスを出力させることができる。 μ PD7220Aの描画動作の優先順位を

落として使用する応用や、ライン・バッファ内蔵DRAMとのインターフェースを容易にする。

- ②VH; 垂直消去時間であることを示すVBLANKタイミング・ステータスをDB₆端子から検出できる。

③VL; インタレース・シュリンク設定時の1フレームあたりの表示に、使用される走査線数を偶数本に設定することができる。

- ④WG; グラフィック・モード、または、混在モードにおいて、「DGD=1」の場合であっても、文字描画時と同様に、WRITE₀コマンドにより、パラメータ設定値をそのままの形で表示メモリに書き込むことができるため、画面がバイトの境界で区切られているように構成されたビット・マップ・メモリであれば文

字描画をバイト単位で実行できる。塗りつぶし描画や描画方向を変化させる必要のないグラフィックス文字描画などが高速化できる。

- ⑤CE; スレープ動作時、外部同期入力信号(EX. SYNC)の立ち下がり時に、かならず、内部同期信号発生回路を初期化する。外部同期信号が不安定であるテレテキストなどへの応用に対して効果を発揮する。

(4)文字表示におけるカーサ位置検出を、表示アドレス16ビットすべてについて行っている。

(5)文字表示におけるカーサ形状定義に関する制約条件がすべて除去された。

(6)アドレス出力A₁₆とA₁₇はステイック信号を出力する。

(7)拡大表示時の $\overline{\text{RAS}}$ 信号の高レベ

DIR	描画アドレスの演算
0	EAD+P→EAD
1	EAD+P→EAD
	dAD(MSB)=1:EAD+1→EAD dAD→LR
2	dAD(MSB)=1:EAD+1→EAD
	dAD→LR
3	EAD-P→EAD
	dAD(MSB)=1:EAD+1→EAD dAD→LR
4	EAD-P→EAD
5	EAD-P→EAD
	dAD(LSB)=1:EAD-1→EAD dAD→RR
6	dAD(LSB)=1:EAD-1→EAD
	dAD→RR
7	EAD+P→EAD
	dAD(LSB)=1:EAD-1→EAD dAD→RR

●μPD7220を使ったRGB同時描画ボード

RGB同時描画を行う一番簡単なアプローチは、各色プレーンごとに1個のμPD7220を割り当てる方法ですが、コストに敏感な分野での応用では、この方法は顧みられること

〈表A〉 μPD7220Aのスピード区分

品名	最高動作周波数
μPD7220A	6 MHz
μPD7220A-1	7 MHz
μPD7220A-2	8 MHz

ルが、HBLANKの立ち上がり時に、半クロックだけ通常時と比較して伸びる現象は起きない⁽⁶⁾。

(8)1画面表示終了後のHFP(画面右下隅)においても、また、垂直帰線消去期間においても、表示アドレスのインクリメントを実行する。

(9)VS_{SYNC}, DACK端子にプルアップ抵抗, LPEN端子にプルダウン抵抗を内蔵している。

(10)最高動作周波数は8 MHzであり、3種類のスピード区分がある(表A)。

〈表1〉◀
描画アドレス演算

ビット・アドレス	MASKコマンドのパラメータ値								
	P1	0	0	0	1	0	0	0	1
0	P1	0	0	0	1	0	0	0	1
	P2	0	0	0	1	0	0	0	1
1	P1	0	0	1	0	0	0	1	0
	P2	0	0	1	0	0	0	1	0
2	P1	0	1	0	0	0	1	0	0
	P2	0	1	0	0	0	1	0	0
3	P1	1	0	0	0	1	0	0	0
	P2	1	0	0	0	1	0	0	0

◀〈表2〉▶
一語4ビット構成時のJADとMASKコマンドのパラメータとの関係

はないかもしれません。そこで、μPD7220Aを1個だけ使用して、RGB同時描画を実行できるボードの回路図を図2に示します。

μPD7220Aは「REPLACE/COMPLEMENT/SET/CLEAR」の4種類のドット修正モードをもっており、16ビットのデータ・バスにそれぞれ独立した修正ゲートがあります。すなわち16ビットまでのデータであれば、同時修正が可能であり、かつこの修正機能はグラフィックス制御にとって不可欠な機能です。この機能を損なわずにRGB同時描画を可能とするために、表示メモリのデータ・バス幅を各面に対して4ビット(最大4面まで可)に縮小し、1アドレスにRGB各4ビットが分散して格納される形をとります(図1)。

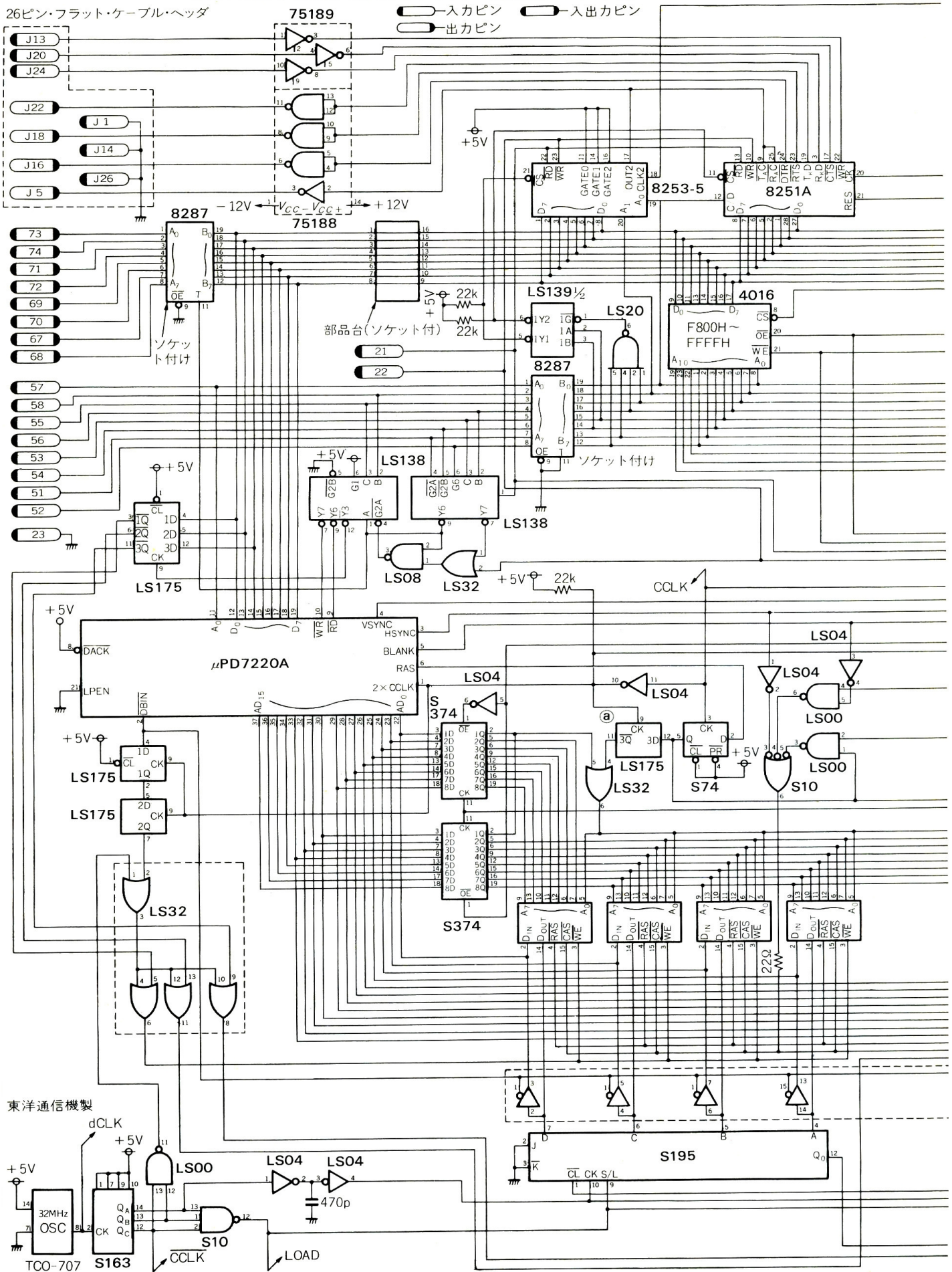
実際にμPD7220A内では、図3のような回路構成によって、グラフィックス描画を実行しています。描画サイクルに入ると、まず、修正前のデータが表示メモリから読み出され、μPD7220A内の一時記憶レジスタに書き込まれます。ドット・アドレス・レジスタ(dAD マスク・レジスタ兼用)の値が1であるビットに対してのみ修正ゲートが接続され、REPLACE選択時を除き、0のビットについては修正は実行されず、元のデータがそのまま書き込みされ

ることになります。さらに、16ビットの線パターン情報のうちの1ビットだけをポインタによって選択し修正を加えます。したがって、図3の例では、16ビットの入力信号中、マスク・レジスタに1が立っているビット位置にある4ビットのデータが修正されることとなります。

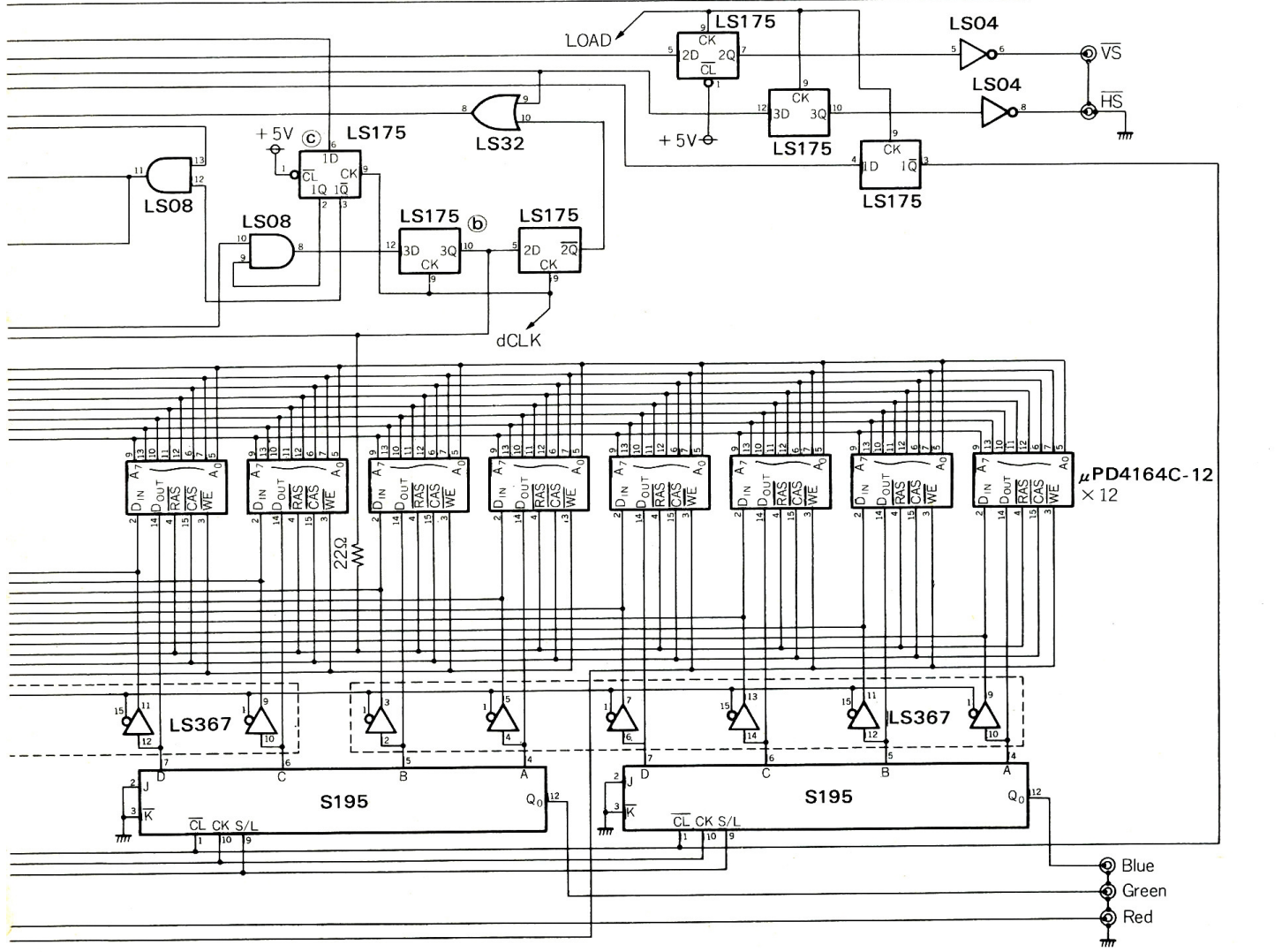
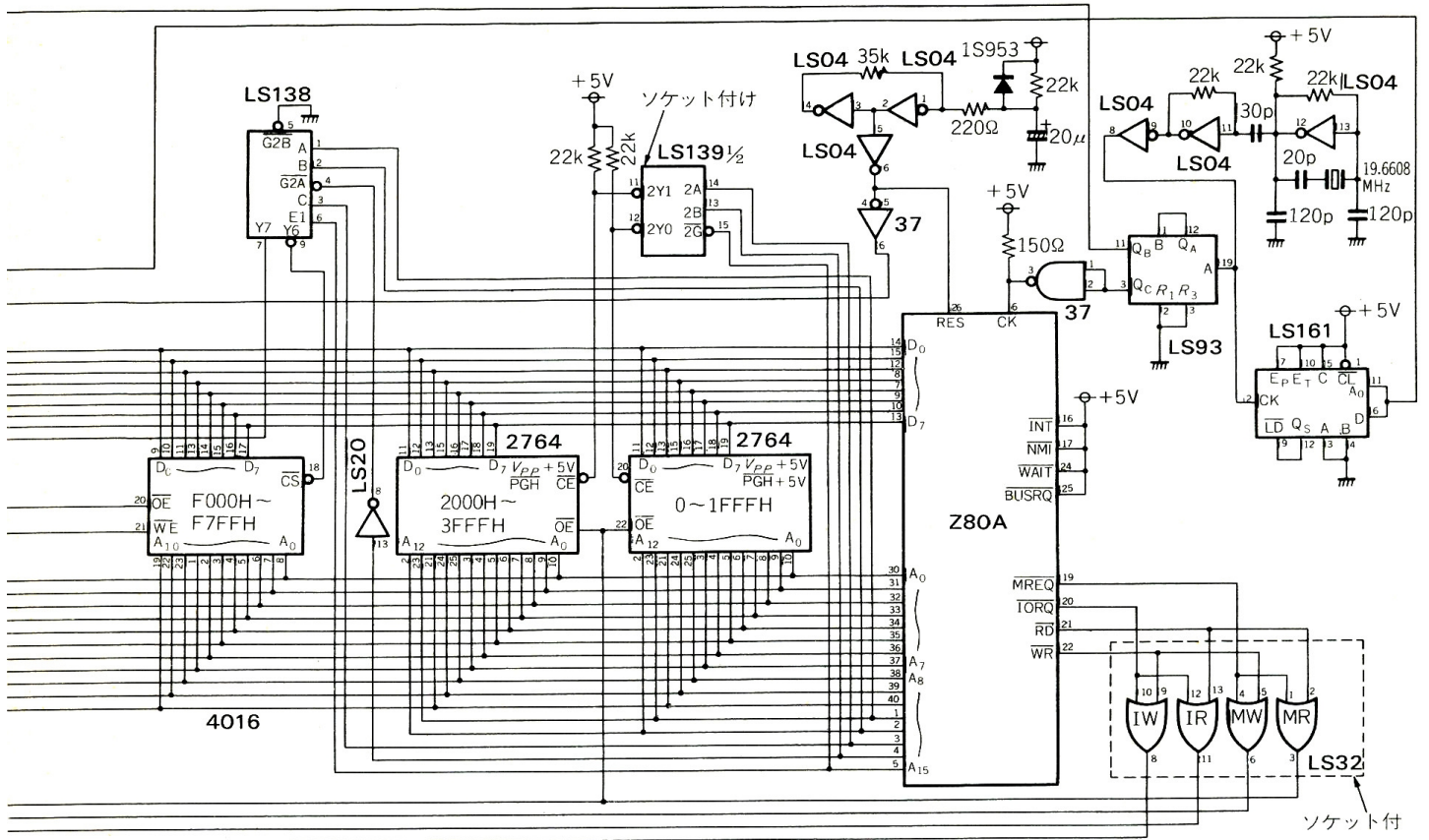
さらに、描画終了後、μPD7220Aは自分自身で描画アドレス(EAD, dAD)の更新を行います。その際、D, D1などのレジスタ間演算により、描画方向(DIR, 図4)の変化をまず決定し、その結果を基にして表1に示したような描画アドレス演算を実行します。この時、dADのLSBまたはMSBが1となることを判別し、EADとPITCHレジスタ(P)との演算やdADの左右への回転動作(LR, RR)を行っています。したがって、1語を4個の4ビットに分割して構成したとき、各4ビットのフィールドに同一値のdADを設定しておけば、μPD7220Aがもつ各種の高速グラフィックス描画機能は有効に動作することがわかります。

ドット・アドレスの設定はCSR Wコマンドによらず、表2にしたがってMASKコマンドによって直接設定します。このとき、パラメータ値は2バイトとも同一値となるため、テーブル参照による作成は容易です。つぎに、どの色プレーンに対して書き込み信号を供給し、修正済みデータの書き込みをするのかを選択するため、実際の描画動作に先立って、CPUは色指定値を外付けのフリッ

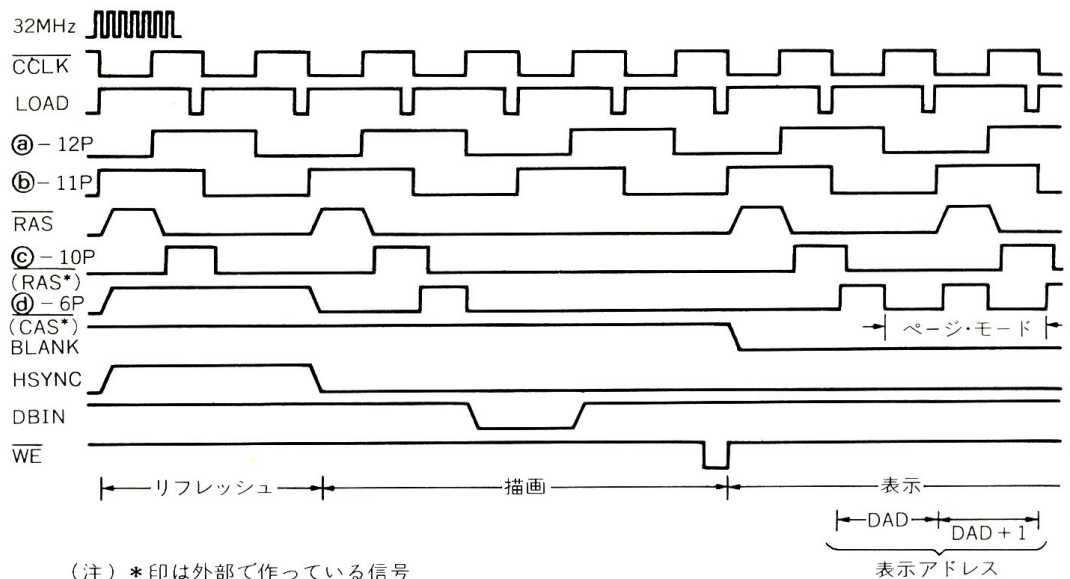
〈図2〉 RGB同時描画が可能なグラフィックス機能拡張ボードの回路



東洋通信機製



〈図5〉
RGB同時描画ボード
のタイミング



プフロップ(LS175)に記憶させておきます。この色指定の設定と描画の実行とはタイミングを合わせる必要があるため、描画が終了し、以前に μ PD7220Aに送出したコマンド/パラメータが内部に読み取られ、FIFOがEMPTY状態になったことを確認した後に、新しい描画実行コマンドとともに色設定することになります。

このボードは、512×512ドットの8色グラフィックス表示制御回路のほかに、Z80-CPU、4KB RAM、16KB PROMやCRT端末などとの接続用として、RS-232Cシリアル・ポートとそのボーレートを発生する

ためのタイマなどが、標準インテル(マルチバス)ボード1枚にすべて実装されています(写真1)。ボードの左半分に位置しているチップ・セレクト回路も含めたグラフィックス制御関係回路は、 μ PD7220A、メモリなども含めて総計36個です。

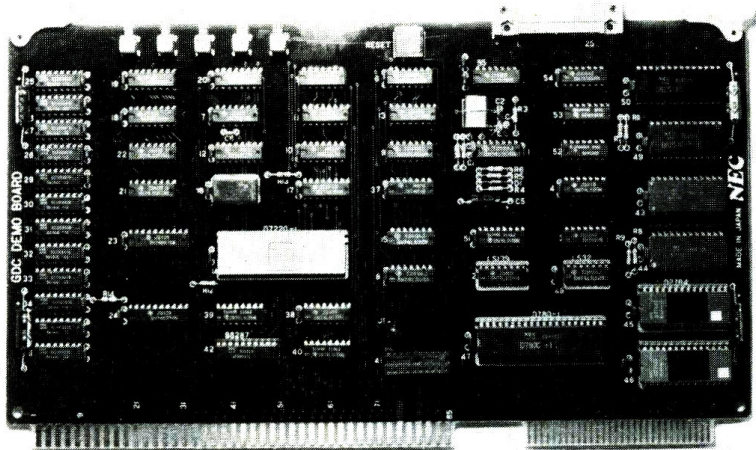
● 4ビット/語データ時のタイミング

1語を16ビットから4ビットに縮小すると、その比率だけ μ PD7220Aに供給するクロック周波数を高める必要が生じます。640×400ドット程度のノンインタレース表示では、使用するモニタ・テレビの仕様に依存しますが、 μ PD7220Aに供給するク

ロックは、2.5MHz近辺となります。なんらの処置もせず、そのままの形で4ビットに縮小すると、同一の表示をするためには、4倍のアドレスをアクセスする必要があることから、約10MHzのクロックを供給しなければなりません。そこで、表示アドレスが1アドレス・サイクルごとに2ずつ増加していくDAD+2の動作モードの選択を行い、1クロックごとに異なるアドレスを発生するようにし、通常使用時の2倍の表示データを読み出すことによって、最高動作周波数以下で μ PD7220Aを動作させるようにします。

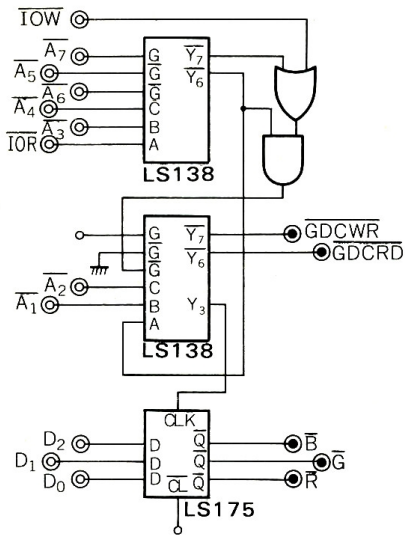
このDAD+2の動作モードを選択したとき、 μ PD7220Aから出力される表示アドレスのサイクルは図5のようになります。表示ブランク時(描画時)やリフレッシュ時を除いて、2クロックのアドレス・サイクルの始めの1クロックでは、 μ PD7220Aが出力するアドレスをそのまま使い、つぎのクロックでは最下位アドレスを強制的に1にし、アドレスを変更する外付け回路を用いています。

このとき、最小で260ns程度しか保証されていないDRAMのサイクル・タイムを満足するほど余裕時間を作り出すことができなくなるため、第2のアドレス・サイクルではRASを低レベルにしたまま、CASだけを信号変化させるページ・モード動



〈写真1〉RGB同時描画ができるグラフィックス機能拡張ボード

〈図6〉CPUインターフェース部



I/Oアドレス	READ	WRITE
60,61H	GDC	GDC
62,63H	アキ	アキ
64,65H	アキ	色指定
66,67H	アキ	アキ

I/Oマップ

作により、サイクル・タイムの不足を補っています。DRAMのサイクル・タイムを速めるには他にも数種の方法がありますが、これらについては後述します。

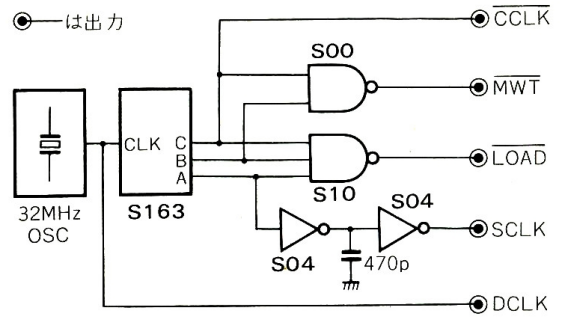
●各部の構成

本ボードにおけるグラフィックス制御回路は非常に簡単な構成となっていますが、以下、これをブロックに分解して説明を加えます。

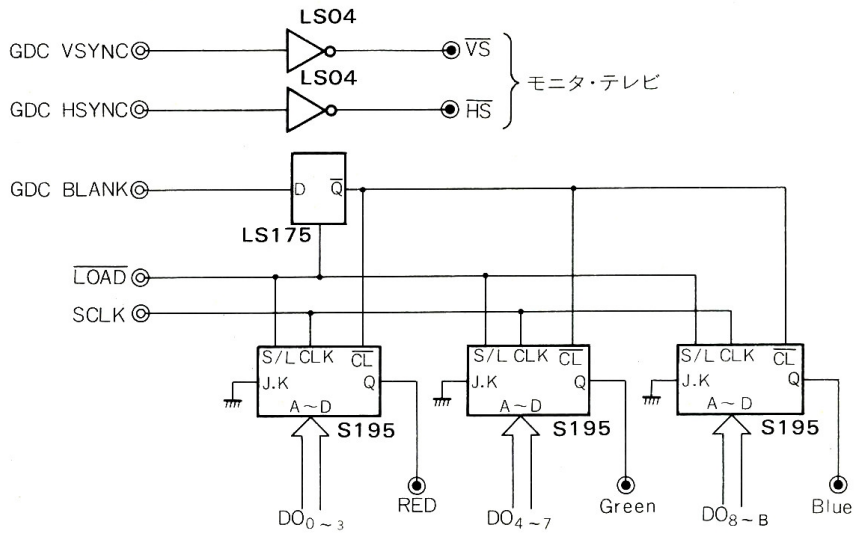
(1)CPU インターフェース部

Z80から出力されたアドレスは、マルチ・バス仕様に合わせるため、信号反転型双方向性バス・ドライバ8287によって負論理信号としてデコーダLS138に接続します。μPD7220Aは独立したチップ選択端子をもっていないため、外部回路により、RD, WR信号にチップ選択信号を組み合わせる必要があります。しかし、図6の回路を用いると、少ないゲート使用量でポート選択数を多く取ることができるようになります。LS175は、RGB同時描画を行う際の色プレーン選択スイッチとして使

〈図7〉ドット・クロック発振/分周タイミング発生部



〈図8〉CRTインターフェース部



(2)ドット・クロック発振/分周タイミング発生部

14ピンのTTLと同じ大きさのパッケージに水晶発振器とドライバを内蔵したハイブリッドICを使用し、32MHz(dCLK)を発振させています。このdCLKは、約30ns刻みでデジタル的遅延をつくりだす基準クロックとしても使用されます。SCLKは、表示メモリから並列に読み出した表示データを一時的に記憶し、直列信号として読み出す働きをもつ並列-直列変換レジスタS195に供給する16MHzのシフト・クロックです。LOADは低レベル期間のSCLKの立ち上がり時に表示データをS195に書き込むためのタイミング信号であり、約60nsの幅をもたせています。MWTは表示メモリへのデータ書き込みタイミング信号であり、幅は約120nsあります。CCLKは1サイクルがSCLK 4クロックに相当するクロックであり、反転後、μPD7220Aに供給されます(図7)。

(3)CRT インターフェース部

垂直同期(VS)、水平同期(HS)、色信号(R, G, B)の各端子をもつカラー・モニタテレビに接続できます。全回路図(図2)では、VSYNC, HSYNCともにLS175を介して信号を遅延させていますが、これはなくてもかまいません。また、BLANK信号を受けるLS175のクロックは、CCLKに変更してもよいでしょう(図8)。

(4)表示メモリ制御-RAS/CASアドレス部

μPD7220Aが出力するRAS信号が制御回路に対する基準信号となります。RASが発生したサイクルのCCLKの立ち下がりから、1クロック間だけ能動的になる信号RAS1/2dを作成します。この信号を60nsだけ遅延させ、表示メモリに供給するRAS信号およびアドレス・ラッチS374のラッチ信号とします。RAS信号立ち下がり後30nsまでは、μPD7220Aのアドレス出力上位8ビット

トの信号がS374を経由して表示メモリに与えられます。その後、CASが立ち上がり、アドレス下位ビットが供給されます。表示サイクル中(BLANK=0)には、CCLK 2クロック間に2回、CASが発生するようにし、2回目のCASの立ち下がり時には、アドレス0を強制的に1とします。リフレッシュ期間中(H SYNC=1)には、μPD7220Aのアドレス下位8ビットに出力されるリフレッシュ・アドレスのみを表示メモリに供給するとともに、CASを強制的に1としRASオンリ・リフレッシュを実行させます(図9)。

メモリに供給するとともに、CASを強制的に1としRASオンリ・リフレッシュを実行させます(図9)。

(5)表示メモリ制御-DBIN, ME部

μPD7220Aが出力するDBIN信号発生期間には、表示メモリ出力信号をμPD7220Aのアドレス/データ・バスに接続します。さらに、DBIN信号を2クロック遅延させた後、MWTおよび色プレーン選択信号と組み合わせ、表示メモリ書き込み信号WEを生成します(図10)。

色選択法

カラー・グラフィックス機能をもっている一般的なパーソナル・コンピュータは、以下に示す2種類の色描画における色選択法を採用しています。

①色彩選択法

②プレーン選択法

色彩選択法は、各色を表現するときに、例えば、RGBの3枚のメモリ・プレーンが存在していることをユーザーが意識しなくともすむように考えられています。指定された色が無条件に描画されます。その結果、描画の順番の新しい色が優先表示されます。例えば、交差する2本の実線を「白」、「赤」の順に描画したとすると、交差点の色は「赤」となります。

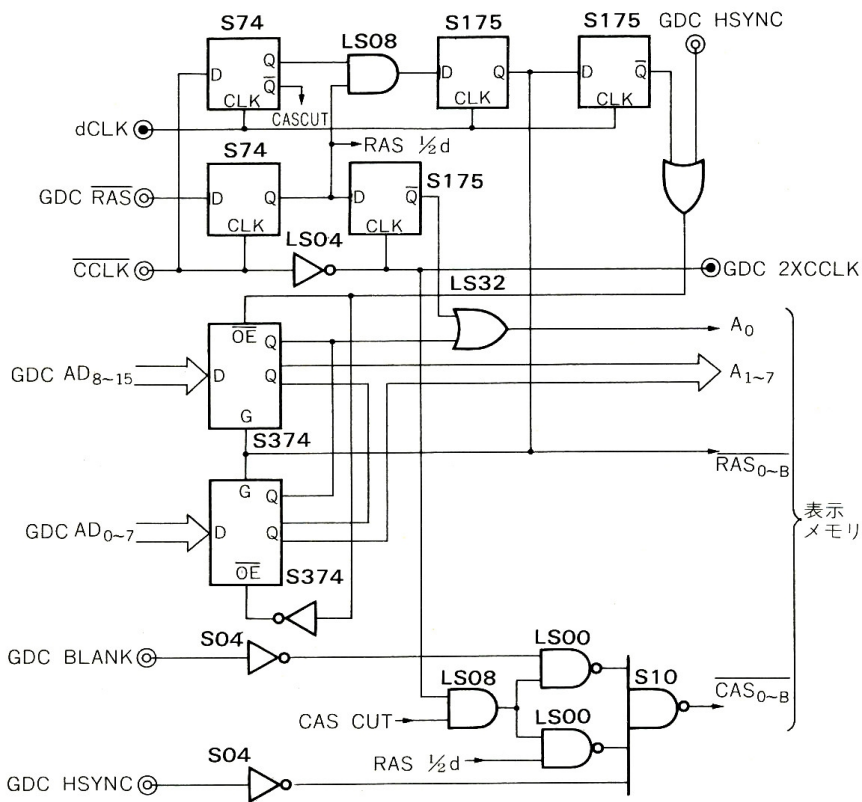
「黄」を描画する場合には、RとGのプレーンに対してSETによる描画を、Bのプレーンに対してはCLEARによる描画を行うため、この選択法では、RGBの3枚のプレーンに対してかならず1回、計3回の描画を実行する必要があります。

プレーン選択法の場合には、ユーザーはRGBに色付けされる別個の3枚のプレーンをそれぞれ独立して使用していることを意識する必要があります。指定された色プレーンだけに対して描画を実行します。

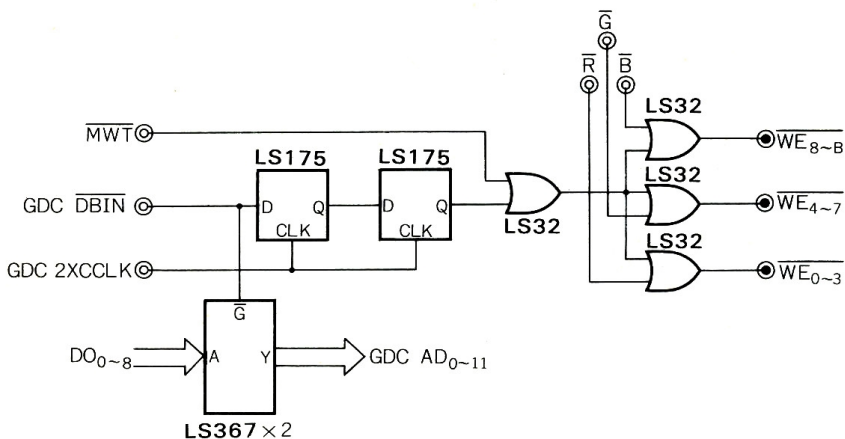
例えば、「黄」を色選択した場合には、RとGのプレーンに対してのみ「REPLACE/COMPLEMENT/SET/CLEAR」の描画修正モードのうちの一つを選択して描画を実行することになります。このプレーン選択法は、多くの層に分かれているプリント基板などの図面を重ね合わせて表示し、色彩効果を非常に高めて使用する応用分野(CADなど; Computer Aided Design)で頻繁に使用されています。

以上、述べたような描画時における2種類の色選択法と同様に、任意閉領域内の塗りつぶしなどの際に必

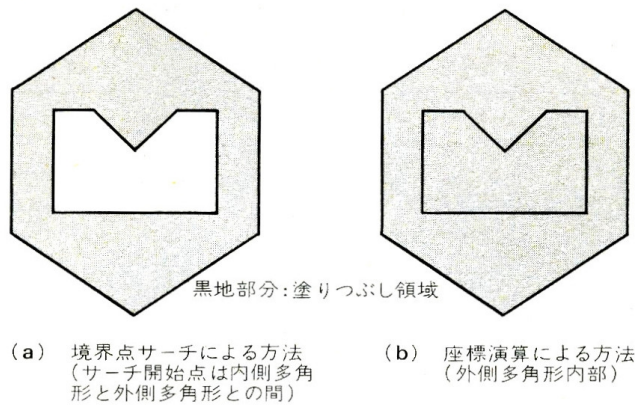
〈図9〉表示メモリ・アドレス RAS / CAS部



〈図10〉表示メモリDBIN, MW部



〈図11〉
閉領域検出法の違い
による塗りつぶし結
果の相違



要となる表示メモリの境界点サーチにおいても、どの色プレーンに対しても境界点サーチを実行するかどうかの選択が必要となります。この色選択においても、上記の①および②の2種類の方法があります。

色彩選択法では、定義した色彩によって境界点をサーチします。すなわち、「黄」を選択した場合には、RGBの3種類の表示メモリの内容を読み出した後、RとG、および、Bの反転内容との論理積をとり、その結果、1である点を境界点とみなします。したがって、RGB計3プレーンに対する読み出し動作を、常に必要とします。

プレーン選択法では、定義されたプレーンに対してのみ境界点サーチを実行します。「黄」を選択した場合には、RとGの表示メモリ内容の論理積結果が「1」である点を境界点とみなします。

RGB同時描画が可能なグラフィックス機能拡張ボードでは、 μ PD 7220Aに供給するクロック周波数を同一としたとき、RGB同時描画ができない通常的设计手法によって設計されたボードと比較して、色彩選択法で1.5倍、プレーン選択法で最高3倍も描画速度が向上します。

塗りつぶし法

任意不定形閉領域の塗りつぶしを実行する前処理ともいえる閉領域検出方法としては、以下の2種類があります。

①境界点サーチによる方法

②CPUの座標演算による方法

①の方法は、実際に表示メモリ上に描画されているデータを読み出して境界点を検出する方法であり、閉領域を形成する図形の形状に依存しない万能型ですが、例えば、円内の塗りつぶしを行おうとしたとき、円と交差する直線が存在していると、その直線によって閉領域が形成されてしまい、所期の結果が得られないことがあります。また、閉領域形成が不完全であると、水が漏れるように他の領域まで塗りつぶしてしまいます。さらに、通常、塗りつぶしの開始点の座標を与える必要があります。

②の方法では、CPUは表示メモリの内容について注意を払わず、座標演算やテーブル参照のみにより塗りつぶし領域を特定する方法であり、多角形や円弧の内部の塗りつぶしに限定されます。そのため、塗りつぶし領域内に他の図形が存在していたとしても、その図形を無視して塗りつぶしてしまいます。その結果、図11のように、内側の多角形と外側の多角形との間を塗りつぶすことはできません。

このように、①、②の方法ともに互いを補間し合うような長所/短所をもちあわせているため、①、②の2種類の塗りつぶしをユーザが選択できる仕様となっている装置があれば、多目的な用途に使用できるようになるでしょう。さらに、①の方法を採る場合には、表示属性用のメモ

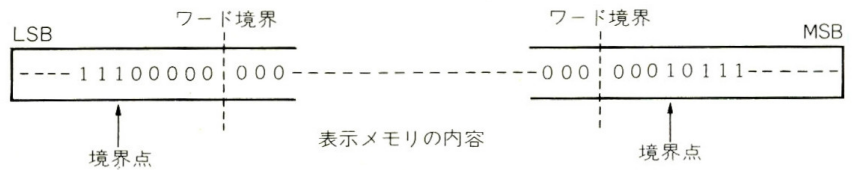
リ・プレーン(表示のプリンクなどに使用する補助プレーン)を用意し、このプレーンに対して境界点サーチを行えば、より自由度の高い塗りつぶし機能を提供できるようになります。

①, ②の場合ともに、閉領域の検出後の塗りつぶし描画は水平方向への直線描画によって実行します。このように描画方向が特定された直線描画であれば、任意方向への直線描画のようにドット単位での描画によらず、ワード単位での描画をさせたほうが高速に塗りつぶしを実行できるはずですが、1語が16ビット構成である場合、単純に計算して16倍も高速になります。

μ PD7220Aはグラフィック・モードであっても、CSRWコマンドの第3パラメータによりWG=1とすることにより、WRITEコマンドに続くパラメータ値をワード単位で描画することができます。境界点検出を終了したのち、図12のように、ワード単位での描画のみによって塗りつぶしを実行できます。

図中、CSRWコマンドのアドレス(\$\$)として、左方の境界点が存在するワード・アドレスを設定します。第3パラメータでWG=1としますが、このときdAD=0とすると、 μ PD7220A内でのパラメータ処理が速くなります。表示メモリ内容が0の部分にのみ描画を実行するように、MASKコマンドのパラメー

〈図12〉 μ PD7220Aによる高速塗りつぶし描画



<CSRW>	-49-	\$\$	\$\$	04	(WG=1)
<MASK>	-4A-	00	F8		
<VECTW>	-4C-	02			(DIR=2)
<WRITE:W>	-23-	**	**		(SET)
<VECTW>	-4C-	02	¥¥ ¥¥		(DIR=2)
<MASK>	-4A-	FF	FF		
<WRITE:W>	-23-	**	**		(SET)
<MASK>	-4A-	07	00		
<WRITE:W>	-23-	**	**		(SET)

タ値を決定します。WRITEコマンドに続くパラメータ値(**)を種々に変化させると、塗りつぶしの紋様を多種多様なものとすることができます。中央部の塗りつぶし開始アドレスは、 μ PD7220Aが自動的に発生するため、特にCSRWコマンドで設定する必要はありません。塗りつぶしワード数をVECTWコマンドの第2, 第3パラメータで設定します。

表示メモリのサイクル・スチール法

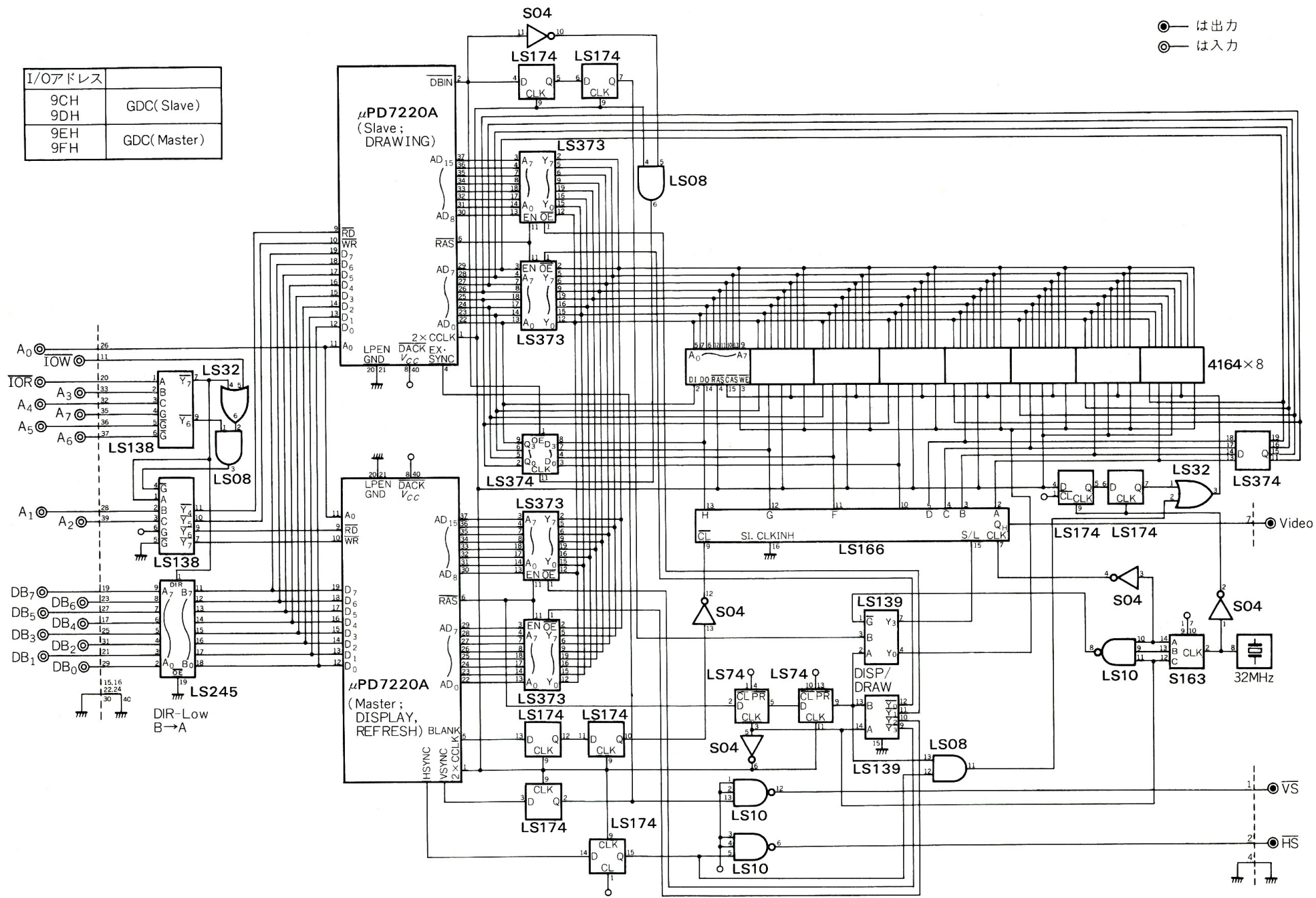
表示消去期間中にしか描画に入れないフラッシュ・レス描画モードでの応用が多く、「カタログ上での描画速度は速いのに損をしている」とか、「文献1に示してあるサイク

ル・スチール法の具体例について知りたい」という言葉がよく聞かれます。表示期間中であっても、メモリ・アクセスに余裕が生じた空きサイクルを盗んで、描画が実行できれば、描画が間断なく連続して実行されていると仮定した場合には、その描画速度は約4~5倍に向上するはずですが、この時、表示にフラッシュなどのチラツキは生じないことは言うまでもありません。以下に、このように描画速度を著しく向上できるメモリ・サイクル・スチール法の具体例を示します(図13)。

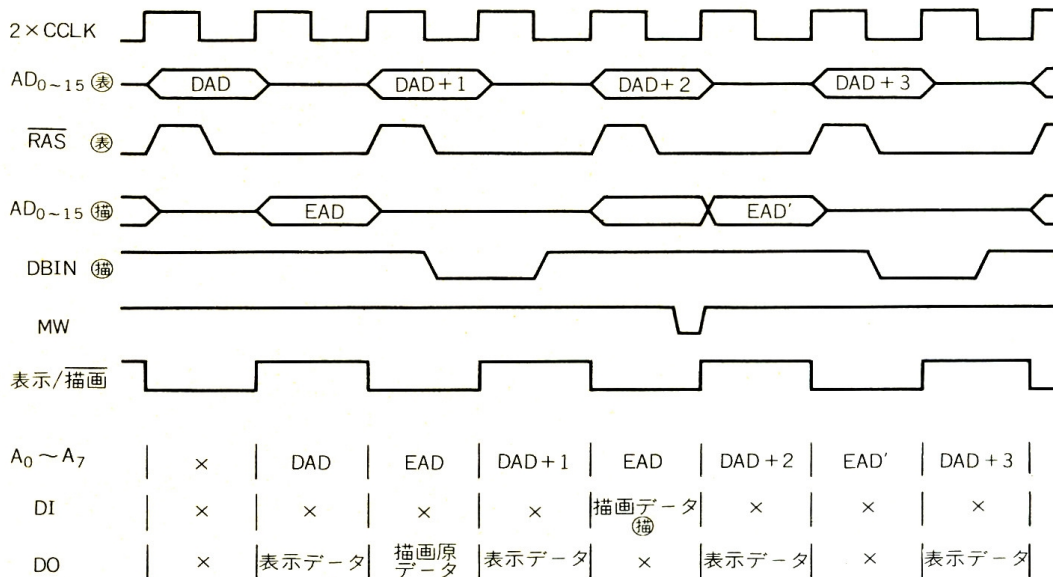
●メモリ・サイクル・スチールの構成

ハードウェア構成上、表示とリフレッシュのアドレスを出力する表示用と描画アドレスを出力する描画用の2個の μ PD7220Aを使用し、表示

〈図13〉 メモリ・サイクル・スチール可能なグラフィック・ボード図



〈図14〉
メモリ・サイクル・
スチール・タイミン
グ図



アドレスと描画アドレスのサイクルと交互に切り替えます。この方式を採ると、バスの切り替え信号やメモリ制御信号の発生が容易になります(図14)。

表示用 μ PD7220Aは他の一般的なCRTCで代用することができますが、実装効率を高めるため、本具体例では、 μ PD7220Aを使用しています。表示用と描画用ともに同一周波数のクロックを供給します。

μ PD7220Aが出力する表示アドレスのサイクルは2クロックですが、サイクル・スチールを実行する应用の場合には、その2クロックのうち1クロックを描画可能なアドレス・サイクルとして割り当てます。したがって、表示および描画のアドレス・サイクルは、共に1クロック期間となります。

描画用 μ PD7220Aは描画時、連続した4クロックのアドレス・サイクル中でR/M/W動作をしますが、上記したようにメモリ・サイクルを1クロックに分割し縮小したため、リードとライトを各々異なるメモリ・サイクルに分けて、実行させることとなります。

μ PD7220Aから出力されるアドレスやメモリから読み出されるデータは、いったん、外付けのラッチやフリップフロップに記憶させ、1クロック後に使用するようにタイミング

設計します。このとき使用するラッチなどは、サイクル・スチールを必要としない应用であっても、バッファやセクタとして付加しなければならない外付け回路であるため、とりたてて外付け回路が増加するわけではありません。むしろ、メモリ・サイクルの区切りが単純であるため、RASやCASなどのメモリ制御信号は μ PD7220Aのクロックに同期させて発生させることができ、タイミング設計が容易になります。図13に示すように、コンパクトな回路でメモリ・サイクル・スチールを実現できます。

描画用 μ PD7220Aに対しては、フラッシュ描画(F=0)、リフレッシュ・アドレス発生なし(D=0)、および表示停止状態の動作モードの設定をします。これらの設定をすると、DMA転送や描画を連続して実行でき、かつ、DMA転送時の制約事項に注意を払う必要がなくなります。

●CPUが直接表示メモリの内容を読み書きする応用

描画可能なメモリ・サイクルにおいて、描画用 μ PD7220Aが描画動作を起こしていない時に、CPUのアドレス/データ・バスを描画用 μ PD7220Aの代わりに接続し、CPUが直接、表示メモリの内容を読み書きできるような应用到に発展させることも簡単にできます。CPUがアクセス

できるタイミングが1クロックおきに発生するため、その抽出が容易であるからです。 μ PD7220Aに付加された描画ウェイト機能(CE=1)を使用すれば、 μ PD7220AとCPUとの並行動作も可能となります。

1024×512ドットの領域に対して、ドット単位描画によるグラフィックス文字描画を実行したところ、フラッシュ・レス描画でリフレッシュ・アドレス発生ありの設定をした場合に、約1.8秒であった描画時間が、上記設定をすると約0.5秒となりました(描画条件：4 MHz μ PD7220A)。

ダイナミックRAMの開発動向からみたグラフィックス装置の展望

DRAMは、記憶容量に対する消費電力が少なくすみ、さらに、実装効率が高いことからグラフィックス表示用メモリ素子として一般に用いられています。しかしながら、定常電流を低く抑え消費電力の減少を図ることを目的として、論理ゲート動作とは無関係なプリチャージ動作を必要とするダイナミック回路構成を採っているため、サイクル・タイムの高速化にはおのずから限界を生じています。

●メモリのサイクル・タイムの高速化

一方、表示のドット数や1ピクセル(画素)当たりのビット数が増加するにつれて、表示メモリとして使用されるダイナミックRAMのサイクル・タイムの高速化が望まれてきており、以下に示す方法が提案されています。

- ①ページ・モード
- ②ニブル・モード
- ③スタティック・コラム
- ④ライン・バッファ内蔵デュアル・ポート

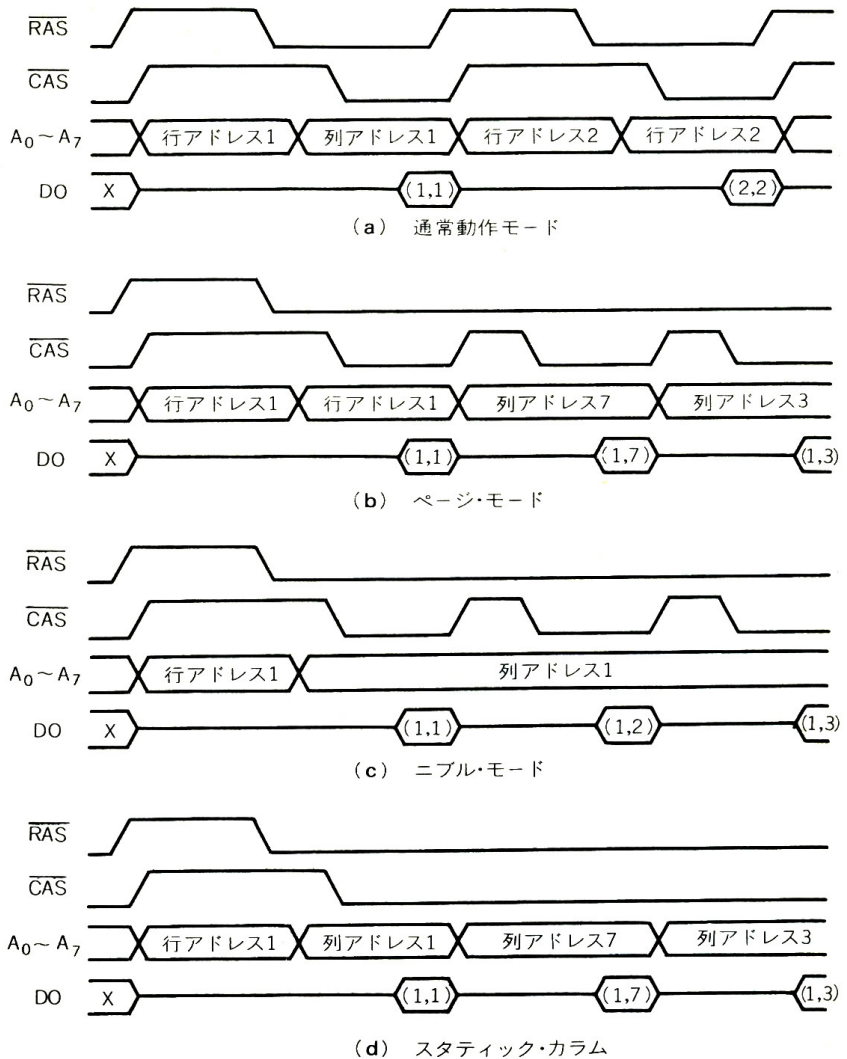
ページ・モードは一般的なDRAMがもっている機能であり、RAS信号を立ち下げ、1行分の記憶データを一時的にセンス・アンプに保持した後は、異なった列アドレスを伴ったCAS信号を立ち下げるサイクルを繰り返すだけで、同一行内の異なったアドレスに格納されているデータのアクセスを実行できます。ページ・モード動作中のサイクル・タイムは、RASのプリチャージに要する時間だけ短縮されると考えて良いでしょう。

ニブル・モードは、CAS立ち下がり時に与えられた列アドレスの下位2ビットが、「00, 01, 10, 11」である4個(ニブル)の連続したアドレスに格納されているデータを保持し、CASの立ち下がりによって歩進するポイントによって、そのデータを選択します。ページ・モードのように、列アドレスをCAS立ち下がりごとに与える必要はありませんが、サイクル・タイムの高速化は連続した4アドレス内のアクセスに限定されるため、自由度が高いとは言えません。

スタティック・コラム方式は、列アドレス関係の回路をスタティック回路によって実現しているため、消費電力は増大しますが、CAS信号は不要となり、1行中の記憶データのアクセスであれば、スタティックRAMと同程度のサイクル・タイムを得ることができます。

以上、掲げた方法は、いずれもデ

〈図15〉 DRAMのサイクル・タイム高速化の方法



ータのアクセスをする際に、メモリ素子に対し必ずアドレスを与えなければなりません。しかしながら、連続したアドレスに格納されているデータを読み出すということに限定すれば、DRAM内にライン・バッファと、読み出し信号の到来ごとに読み出し位置を変更する回路を内蔵することにより、アドレスを不要とすることができます。このアプローチが可能であれば、デュアル・ポート化、すなわち、1個のメモリを2種類のポートから制御できるようになります。

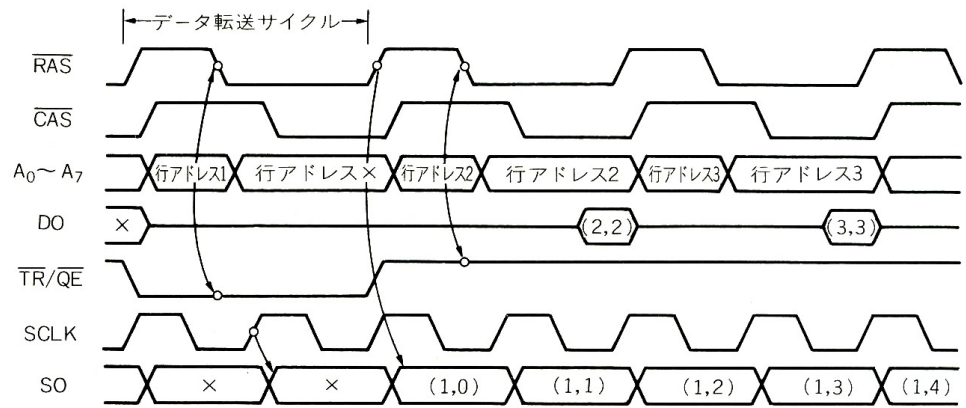
テキサス・インスツルメント(TI)からライン・バッファ内蔵DRAMが発表されました⁽⁶⁾。このTI4161は、アドレスやRAS, CASなどが接続されるポートAには64K×1ビット

の記憶セルを、シリアル・ポートBには256ビットのライン・バッファを内蔵しています。1ラインの表示の開始直前にポートAより表示アドレスを供給し、1行分のデータをライン・バッファに蓄積した後(この動作をデータ転送サイクルと呼ぶ)は、ポートBに対し転送クロック(SCLK)を供給するだけで、ポートAとは独立して表示データを取り出せるようになっています(図16)。

●DRAMの効果的使い方

従来型のDRAMを使用した場合、表示期間中は表示アドレスを優先して供給しなければならないため、描画動作など表示以外の目的でDRAMをアクセスできる時間に大きな制約がありました。このため、描画などに使用可能な時間を多く確保する

〈図16〉
ライン・バッファ内蔵
デュアル・ポートDR
AM TI4161のタイミ
ング図



ため、次のようなアプローチをして
いました。

- ①メモリ・サイクル・スチール
- ②外付けライン・バッファへの表示
データの高速転送

①は、低から中程度の解像度をも
つ表示装置に適用することができま
したが、640×400ドットを超える表
示装置に対しては、DRAMの速度
の限界から応用することができず、
②の方式を採らざるを得ませんでした。
②では、並列読み出しを行うデー
タの幅を64ビット程度に広げて、
メモリのアクセス回数を減らすなど
の措置を講じないと効果が得られな
いことから、メモリ素子数が増加し、
さらには、集積度の高いメモリ素子
を使用するとメモリの使用効率が落

ちるため、むしろ、集積度の低い素
子を使用しなければならないなど装
置設計上、矛盾が生ずる結果を招い
ていました。

ライン・バッファ内蔵デュアル・
ポートDRAMは、上記した問題点
をすべて一掃する画期的なメモリ素
子であり、今後、グラフィックス装
置に表示用記憶素子として大量に使
用されることが予想されています。
しかし、現時点で量産されている唯
一の製品であるTI4161には、例え
ば、水平方向の表示ドット数を任意
値に選択できないなど実使用上、ま
だまだ改良すべき点がいくつかあり、
今後、集積回路メーカーから各社工
夫をこらした製品が続々と発表され
ることでしょう。

◆参考文献◆

- (1)小口哲司, 南野宏樹, 樋口三左男;
グラフィック・ディスプレイ・コン
ローラ, トランジスタ技術, 1983年
1月号, p.320
- (2)日本電気, μPD7220ユーザーズ・マ
ニュアル, IEM-734
- (3)日本電気, μPD7220アプリケーシ
ョン・マニュアル, IEM-855
- (4)日本電気, μPD7220A開発速報, IP-
5289
- (5)浅田善己, 村上知広; μPD7220(GD
C)を用いた電子ビーム描画露光装置
の製作, インターフェース, Sept.
1983, p.186
- (6)Ray Pinkham, Mark Novak, K
arl Guttag; Video RAM excels
at fast graphics, Electronic
Design, Aug.18, 1983, p.161

ここで、μPD7220Aのコマンド内容と、その設定法についてまとめておきます。

コマンドのパラメータの設定を表Aに示します。以下、各コマンドについて説明します。

【動作制御】

●RESETコマンド

RESET1, RESET2, RESET3の3種類が用意されており、同期信号発生回路や描画動作の停止などの初期動作の実行を指示する(表B)。μPD7220AはReset入力端子をもたないため、電源を入れた直後では、内部レジスタや出力端子の状態は不定となっている。したがって、この場合には、FIFOの状態を検出することなく、無条件にこのコマンドを送出する必要がある。

VECTWコマンドのパラメータ

によって設定するDC, D, D1, D2, DMを除き、コマンドに付随するパラメータによって設定する種々のレジスタ値に対する初期化は、本コマンドによって実行されないため、初期化フローにおいて個々のコマンドを与え、初期化しなければならない。

このコマンドに続いてパラメータを送出した場合には、SYNCコマンドと同一の動作をする。

●SYNCコマンド

ノンインタレース、インタレース、インタレース・シュリンク(I, S)の選択、文字、文字/グラフィック混在、グラフィック(CHR, G)の選択、描画タイミング(F), リフレッシュ・アドレス発生有無(D)などの表示動作モードや、同期信号発生期間(HFP, HS, HBP, C/R, VFP, VS, VBP, L/F)の定義などを指示する(図A)。D=1とした

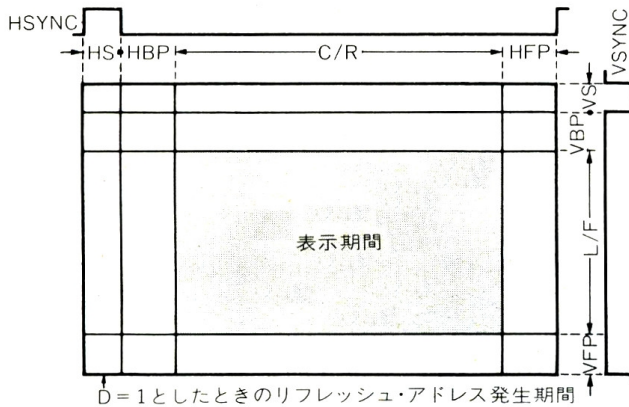
とき、水平同期信号発生期間中に2クロック周期で連続したDRAMに対するリフレッシュ・アドレスを、アドレス出力端子の低位8ビットに出力する。

水平関係のレジスタ設定値に関して、例えば、『インタレース・シュリンクのモードにおいて拡大表示を行うとき、「HFP」の設定値は3文字以上』などという制約条件があり、これらの条件を一つでも満足しない設定をした場合には、ユーザが期待する動作が保証されないので、特に注意が必要である(図B)。

●MASTER/SLAVEコマンド

マスタ動作かスレーブ動作かの選択をする。スレーブ動作の選択をした場合(M=0)には、外部同期信号受け付け可能期間中(例えば、STOP1コマンドからSTARTコマンドまで)の外部同期信号の立ち下

〈図A〉 SYNCコマンドによる表示フォーマットの定義

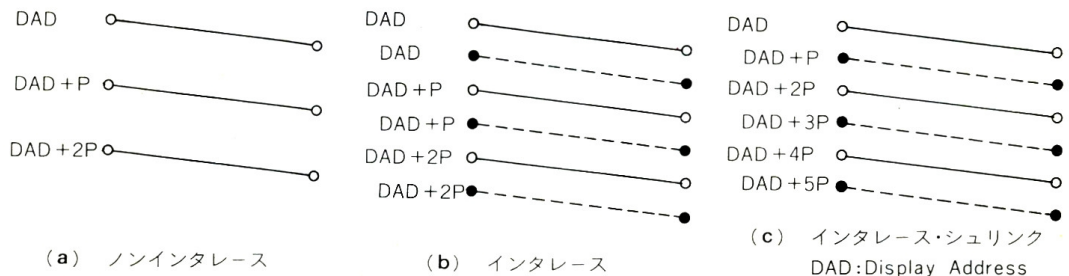


〈表B〉 RESET, SYNC, STOP, STARTの動作

コマンド名	初期化動作	表示の停止/開始	外部同期信号の受け付け停止/開始
RESET1	有	停止	開始
RESET2	有	停止	停止
RESET3	有	開始	停止
SYNC	DE=0	無	開始
	DE=1	無	停止
STOP1	無	停止	開始
STOP2	無	停止	停止
START	無	開始	停止

- (1) 外部信号受け付けは、スレーブ動作設定時(M=0)にのみ有効となる。
- (2) M=0, CE=1の場合には、常時、外部信号受け付け開始状態となる。

〈図B〉 各表示ラインにおける表示開始アドレスの推移(L/R=1)



〈表A〉 コマンド・パラメータの設定

コマンド名	C/P	コード							
		DB ₇	DB ₆	DB ₅	DB ₄	DB ₃	DB ₂	DB ₁	DB ₀
RESET1	C	0	0	0	0	0	0	0	0
RESET2	C	0	0	0	0	0	0	0	1
RESET3	C	0	0	0	0	1	0	0	1
SYNC	C	0	0	0	0	1	1	1	DE
	P1	0	0	CHR	F	I	D	G	S
	P2	C/R							
	P3	VS _L				HS			
	P4	HFP				VS _H			
	P5	DS	P _H	HBP					
	P6	VH	VL	VFP					
	P7	L/F _L							
	P8	VBP				L/F _H			
MASTER/SLAVE	C	0	1	1	0	1	1	1	M
START	C	0	1	1	0	1	0	1	1
	C	0	0	0	0	1	1	0	1
STOP1	C	0	0	0	0	1	1	0	0
STOP2	C	0	0	0	0	0	1	0	1
ZOOM	C	0	1	0	0	0	1	1	0
	P1	ZR				ZW			
SCROLL	C	0	1	1	1	RA			
	P1 } P16	任意アドレスから任意数のパラメータ							

コマンド名	C/P	コード							
		DB ₇	DB ₆	DB ₅	DB ₄	DB ₃	DB ₂	DB ₁	DB ₀
CSRFORM	C	0	1	0	0	1	0	1	1
	P1	CS	CE	0	L/R				
	P2	BL _L		BD	CST				
	P3	CFI				BL _H			
PITCH	C	0	1	0	0	0	1	1	1
	P1	P _L							
VECTW	C	0	1	0	0	1	1	0	0
	P1	SL	R	C	T	L	DIR		
	P2	DC _L							
	P3	X	DGD	DC _H					
	P4	D _L							
	P5	X	X	D _H					
	P6	D2 _L							
	P7	X	X	D2 _H					
	P8	D1 _L							
	P9	X	X	D1 _H					
	P10	DM _L							
	P11	X	X	DM _H					
VECTE	C	0	1	1	0	1	1	0	0
TEXTW	C	0	1	1	1	1	RA'		
	P1 } P8	"8"以上の任意アドレスから任意数のパラメータ							
TEXTE	C	0	1	1	0	1	0	0	0

- C/P = Command / Parameter
- DE = Display Enable
- CHR = Character mode
- F = Flash
- I = Interlace
- D = Dynamic RAM mode
- G = Graphics mode
- S = Shrink
- C/R = Character / Row
- VS = Vertical Sync
- HS = Horizontal Sync
- HFP = Horizontal Front Porch
- DS = Draw Stop
- P_H = Pitch Register の最上位ビット
- HBP = Horizontal Back Porch
- VH = Vertical / Horizontal
- VL = Vertical Line
- VFP = Vertical Front Porch
- L/F = Lines / Frame
- VBP = Vertical Back Porch
- M = Master
- ZR = Zoom Read
- ZW = Zoom Write
- RA = RAM Address
- CS = Display Corsor

- CE = Continuous External Sync Enable
- L/R = Lines / Row
- BL = Blinking Rate
- BD = Blinking Disable
- CST = Corsor Start
- CFI = Corsor Finish
- P = Pitch Register
- SL = Scan Line
- R = Rectangle
- C = Cercle
- T = Text
- L = Line
- DIR = Direction
- DC = DC Register
- DGD = Dynamic Graphics Drawing
- D = Display Address
- DM = DM Register
- RA' = RAM Address 4ビットのうち最上位ビットが1
- EAD = Execution Address
- dAD = dot Address
- MASK = MASK Register
- WG = Write in Graphics Mode
- MOD = Modification Mode
- LB = Low Byte
- HB = High Byte

コマンド名	C/P	コード							
		DB ₇	DB ₆	DB ₅	DB ₄	DB ₃	DB ₂	DB ₁	DB ₀
CSRW	C	0	1	0	0	1	0	0	1
	P1	←EAD _L →							
	P2	←EAD _M →							
	P3	dAD	WG	0	←EAD _H →				
MASK	C	0	1	0	0	1	0	1	0
	P1	←MASK _L →							
	P2	←MASK _H →							
LPEN	C	1	1	0	0	0	0	0	0
	(1)	←LAD _L →							
	(2)	←LAD _M →							
	(3)	X	X	X	X	X	X	←LAD _H →	
CSRR	C	1	1	1	0	0	0	0	0
	(1)	←EAD _L →							
	(2)	←EAD _M →							
	(3)	X	X	X	X	X	X	←EAD _H →	
	(4)	←dAD _L →							
(5)	←dAD _H →								
WRITE:LB	C	0	0	1	1	0	0	←MOD→	
	P1	←LB1→							
	P2	パラメータとして下位バイト・データを連続送出可							
WRITE:HB	C	0	0	1	1	1	0	←MOD→	
	P1	←HB1→							
	P2	パラメータとして上位バイト・データを連続送出可							

コマンド名	C/P	コード							
		DB ₇	DB ₆	DB ₅	DB ₄	DB ₃	DB ₂	DB ₁	DB ₀
WRITE:WORD	C	0	0	1	0	0	0	←MOD→	
	P1	←LB1→							
	P2	←HB1→							
P3	パラメータとして下位→上位バイトの順で連続送出可								
DMAW:LB	C	0	0	1	1	0	1	←MOD→	
DMAW:HB	C	0	0	1	1	1	1	←MOD→	
DMAW:WORD	C	0	0	1	0	0	1	←MOD→	
READ:LB	C	1	0	1	1	0	0	←MOD→	
	(1)	←LB1→							
	(2)	DCレジスタに設定したバイト数だけ連続読み出し可							
READ:HB	C	1	0	1	1	1	0	←MOD→	
	(1)	←HB1→							
	(2)	DCレジスタに設定したバイト数だけ連続読み出し可							
READ:WORD	C	1	0	1	0	0	0	←MOD→	
	(1)	←LB1→							
	(2)	←HB1→							
	(3)	DCレジスタに設定したワード数だけ連続読み出し可							
DMAR:LB	C	1	0	1	1	0	1	←MOD→	
DMAR:HB	C	1	0	1	1	1	1	←MOD→	
DMAR:WORD	C	1	0	1	0	0	1	←MOD→	

()内の数字は読み出しコマンドにおける読み出し順を示す
Xは不定または1/0どちらを設定してもかまわないことを表す

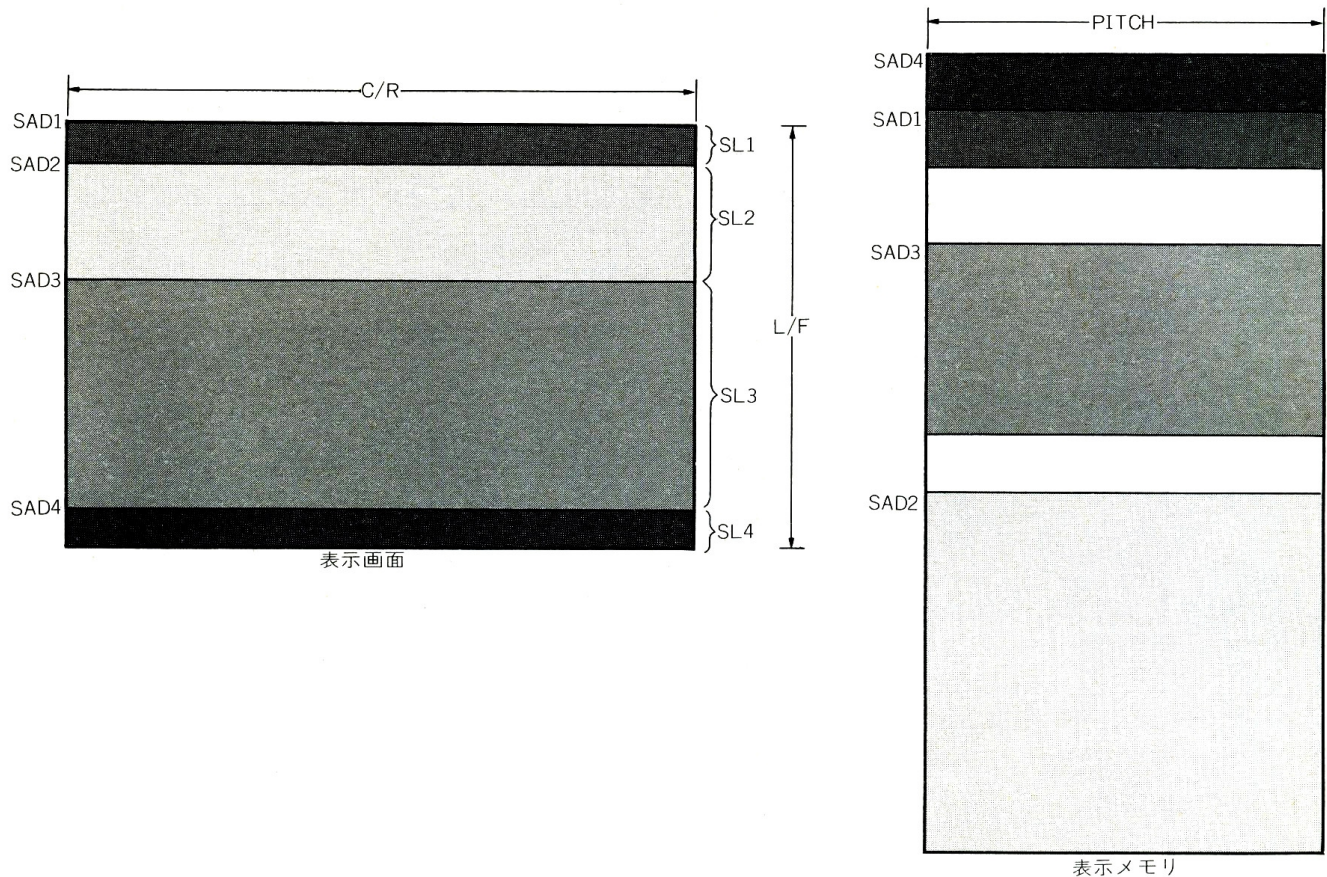
RA	DB ₇	DB ₆	DB ₅	DB ₄	DB ₃	DB ₂	DB ₁	DB ₀
0	←SAD1 _L →							
1	←SAD1 _M →							
2	←SL1 _L →				0	0	←SAD1 _H →	
3	DAD+2	IM	←SL1 _H →					
4	←SAD2 _L →							
5	←SAD2 _M →							
6	←SL2 _L →				0	0	←SAD2 _H →	
7	DAD+2	IM	←SL2 _H →					
8	←SAD3 _L →							
9	←SAD3 _M →							
A	←SL3 _L →				0	0	←SAD3 _H →	
B	DAD+2	IM	←SL3 _H →					
C	←SAD4 _L →							
D	←SAD4 _M →							
E	←SL4 _L →				0	0	←SAD4 _H →	
F	DAD+2	IM	←SL4 _H →					

〈表C〉 内蔵データRAMの内容

直線グラフィックス文字描画時

RA	DB ₇	DB ₆	DB ₅	DB ₄	DB ₃	DB ₂	DB ₁	DB ₀
8	←TX8 or PTN _L →							
9	←TX7 or PTN _H →							
A	←TX6→							
B	←TX5→							
C	←TX4→							
D	←TX3→							
E	←TX2→							
F	←TX1→							

SAD = Scroll Address TX = Text
SLI = Scan Line PTN = Pattern
IM = Image



がり時に、内蔵している同期信号発生回路などの初期化を実行する。この結果、多数の μ PD7220Aを互いに同期が取れた状態で使用したり、ほかのビデオ機器との同期合わせなどが容易になる。

NTSC基準にしたがっている外部同期信号との同期合わせについては、参考文献(1)に触れられている。

【表示制御】

●STARTコマンド

表示の開始を指示する(表B)。

●STOPコマンド

STOP1, STOP2の2種類のコマンドが用意されており、BLANK出力信号を高レベルとして、表示を停止させる(表B)。

●ZOOMコマンド

それぞれ1から16倍までの表示時の拡大係数(ZR)、および、グラフィックス文字描画時の拡大係数(ZW)を設定する。

拡大機能を使用しない場合であっ

ても、初期化フローでかならず設定する必要がある。

●SCROLLコマンド

16バイトの容量をもつ内蔵データRAMの、任意のRAMアドレス(RA)から、パラメータ値を順次、書き込む指示をする。このRAMのデータは4バイトごとに区切られ、表示開始アドレス(SAD)と分割表示画面領域の縦方向の大きさを表すライン数(SL)とで構成される(表C, 図C)。グラフィックス描画を実行する際には、データRAMの上位8バイトは、直線のパターン・フォーマットやグラフィックス文字のフォントを一時的に記憶するために使用される(TEXTWコマンド参照)。したがって、グラフィックス描画を行う場合には、画面の分割表示は最大2画面までに制限されることになる。

●CSRFORMコマンド

おもに、文字表示モード時に使用される1行中のライン数(L/R)、ブ

リンク・レート(BL)やプリンキングの有無(BD)、カーサの表示フォーマット(CST, CFI)や表示の有無(CS)などの指示をする。

グラフィックス表示モードで使用する場合であっても、初期化フローにおいてこのコマンド/パラメータを発行しなければならない。プリンキング・レート設定値として0を設定した場合には、インタレースを正確に実行しない場合があるので注意が必要。

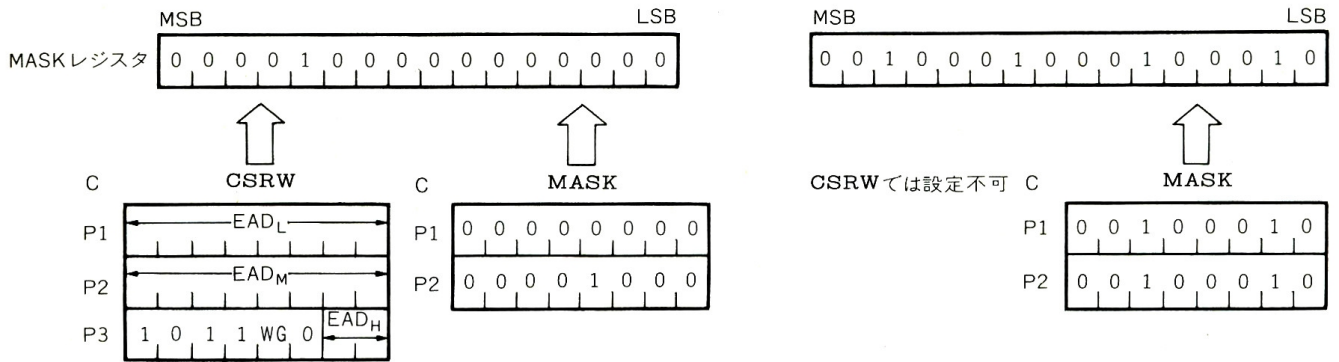
●PITCHコマンド

SYNCコマンドの第5パラメータのPH値と組み合わせられて、表示メモリの水平方向の総アドレス数(0から511)を定義する(図C)。表示や描画のアドレス演算時に、この値が参照される。

●LPENコマンド

画面表示開始時から、第一回目に発生するライト・ペン入力信号の立ち上がり時の表示アドレスを保持するレジスタ(LAD)の内容の読み出

〈図D〉 MASKレジスタの設定法



しを指示する。

ライト・ペンによる位置検出の精度を上げるため、連続した2画面間、同じ表示アドレス位置でライト・ペン信号が立ち上がることを確認した後、LPEN DETECTステータス・フラグがセットされる。通常は、このステータス・フラグが能動的になったことを確認してからアドレスを読みに行くことになるが、フラグが能動的にならなくともLADは信号立ち上がり時の値に更新されており、読み取ることができる。

DAD+2=1とした状態で、 μ PD7220Aを使用している場合には、LADとして偶数アドレスのみを出力することに注意する必要があるが、ライト・ペン自身の検出誤差の範囲内であると考えて差し支えない。

【描画制御】

●VECTWコマンド

描画やDMA転送を行う際に使用される重要なコマンド。描画の種類(SL, R, C, T, L)と描画方向(DIR)および5種類の14ビット・レジ

スタ(DC, D, D2, D1, DM)値の設定をする。

描画の種類にしたがって、レジスタの使用目的が異なっている。負数は「2の補数」表現をする。上位2ビット(×印)は負数であったとしても、強制的に0にする必要はない。

●VECTEコマンド

直線、四辺形、円、弧、1ドット描画の開始を指示する。データRAMのアドレス8と9にTEXTWコマンドによってあらかじめ格納しておく直線、破線、一点鎖線などの線種パターンを参照しながら1ドットずつ描画する。

●TEXTWコマンド

SCROLLコマンドと同一種のコマンド。グラフィックス描画時に必要となる、線種パターンやグラフィックス文字のフォント情報を記憶させるため、データRAMの内容を書き換える(表C)。

●TEXTEコマンド

グラフィックス文字描画の実行開始を指示する。データRAMの上位8アドレスに、あらかじめ格納して

おく文字フォント情報を参照しながら描画する。

●CSRWコマンド

描画実行ワード・アドレス(EAD)とドット位置を示すアドレス(dAD)とを設定する。この描画アドレスは、SCROLLコマンドで設定する表示アドレスとは別個のハードウェアで構成されている。

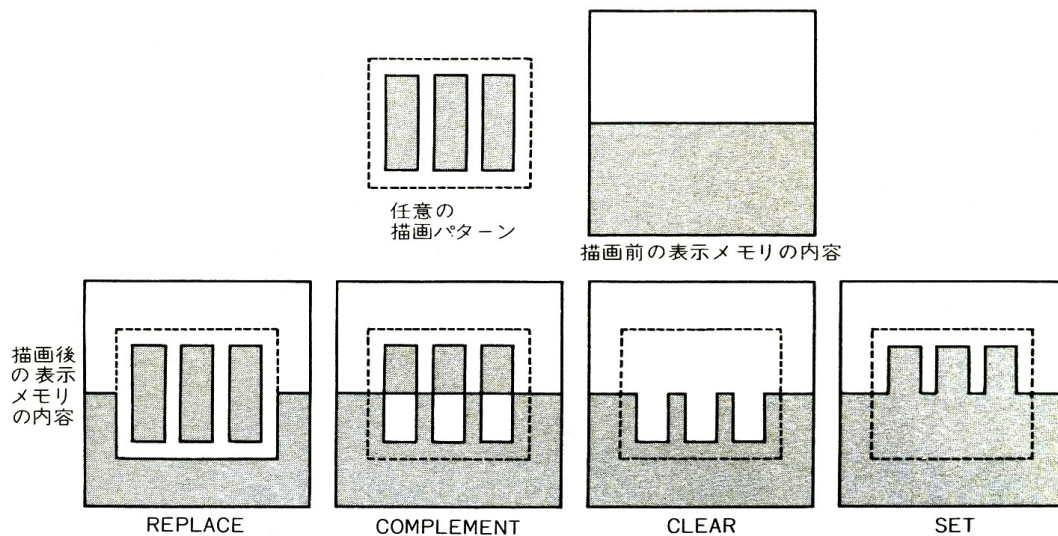
●CSRRコマンド

描画実行ワード・アドレス(EAD)およびドット・アドレス(dAD)の読み出し開始の指示をする。グラフィックス描画実行後にアドレスの読み出しを行う場合には、最終描画を行った次に描画すべきアドレスが読み出されることに注意する必要がある。また、dADは16ビットに展開されたデータとして読み出される。

●MASKコマンド

どのドット位置に対して、描画を実行するのかわを示す16ビットのマスク・レジスタの内容を直接書き換える。このレジスタは、CSRWコマンドによる4ビットにパックされた形で表現されているドット・アドレ

〈図E〉
ドット修正機能



ス(dAD)設定によっても行えるが(図D), この場合には, 16ビットのうちいずれか1ビットのみが1である形式しか選択することができない。MASKコマンドを使用すると任意ビットに1を立てることができ, 描画データ操作の自由度が高まる。

【映像メモリ制御】

●WRITEコマンド

後述するREAD, DMAW, DMARコマンドと同様に, 中位の2ビットによって, ワード/下位バイト/上位バイトのうち, いずれかの選択ができるようになっており, 表示メモリへパラメータ値を書き込む

指示をする。さらに, コマンド・コードの下位2ビットによってREPLACE/COMPLEMENT/SET/CLEARの4種類のドット修正モード(MOD, 図E)のうちから1種類のモードを選択する。

コマンドに続けてパラメータを送出した場合には, パラメータ値が表示メモリに書き込まれていく。このとき, 描画アドレスは自動的に μ PD7220Aによって演算されるため, いちいち, CSRWコマンドによって設定し直してやる必要はない。

●READコマンド

表示メモリのデータをFIFOを経由して読み出す指示をする。アドレ

スが連続している部分のデータの読み出しであれば, μ PD7220Aが自動的に読み出しアドレスを更新し, そのデータをFIFOに蓄積するため, 表示メモリ・データのチェックや移送は容易である。

●DMAWコマンド

DMA転送によって主記憶に格納されているデータを, 表示メモリに書き込む指示をする。連続した一次元アドレスだけでなく, 二次元アドレス領域に対する選択的なDMA転送も設定できる。

●DMARコマンド

DMAWコマンドとは転送方向が逆の同様な動作の実行を指示する。