

お客様各位

---

## カタログ等資料中の旧社名の扱いについて

---

2010年4月1日を以ってNECエレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願ひ申し上げます。

ルネサスエレクトロニクス ホームページ (<http://www.renesas.com>)

2010年4月1日  
ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社 (<http://www.renesas.com>)

【問い合わせ先】<http://japan.renesas.com/inquiry>

## ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
3. 当社製品を改造、改変、複製等しないでください。
4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。  
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット  
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）  
特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

注 1. 本資料において使用されている「当社」とは、ルネサスエレクトロニクス株式会社およびルネサスエレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

保守/廃止

# AGDC Q&A集

$\mu$ PD72120

$\mu$ PD72123

保守/廃止

# AGDC Q&A集

μ PD72120

μ PD72123

**保守 / 廃止**

mc68000は、モトローラ社の製品です。

V25, V33, V40, V50, V53は、日本電気株式会社の商標です。

- 本書に記載されている内容は1994年9月現在の資料にもとづいたもので、今後、予告なく変更することがあります。量産設計の際には最新の個別データ・シート等をご参照ください。
  - 文書による当社の承諾なしに本資料の転載複製を禁じます。
  - この製品を使用したことにより、第三者の工業所有権等にかかわる問題が発生した場合、当社製品の構造製法に直接かかわるもの以外につきましては、当社はその責を負いませんのでご了承ください。
  - 当社は品質、信頼性の向上に努めていますが、半導体製品はある確率で故障が発生します。当社半導体製品の故障により結果として、人身事故、火災事故、社会的な損害等を生じさせない冗長設計、延焼対策設計、誤動作防止設計等安全設計に十分ご注意願います。
  - 当社は、当社製品の品質水準を「標準水準」、「特別水準」およびお客様に品質保証プログラムを指定して頂く「特定水準」に分類しております。また、各品質水準は以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認の上ご使用願います。
    - 標準水準：コンピュータ、OA機器、通信機器、計測機器、AV機器、家電、工作機械、パーソナル機器、産業用ロボット
    - 特別水準：輸送機器（自動車、列車、船舶等）、交通用信号機器、防災／防犯装置、各種安全装置、生命維持を直接の目的としない医療機器
    - 特定水準：航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器、生命維持のための装置またはシステム等
- 当社製品のデータ・シート／データ・ブック等の資料で、特に品質水準の表示がない場合は標準水準製品であることを表します。当社製品を上記の「標準水準」の用途以外でご使用をお考えのお客様は、必ず事前に当社販売窓口までご相談頂きますようお願い致します。
- この製品は耐放射線設計をしておりません。

M8 94.6

本資料に掲載の応用回路および回路定数は、例示的に示したものであり、量産設計を対象とするものではありません。

なお、AGDC ( $\mu$ PD72120,  $\mu$ PD72123) を他の部品と組み合わせて実現するアプリケーション機能の一部が、米国CADTRAK社の米国特許4,197,590およびRe.31,200等ならびにそれらの対応各国特許に関するおそれがあります。このような特許は、他のグラフィック表示コントローラを用いても、あるいはディスプレイ回路を用いても問題になり得るもので、AGDC ( $\mu$ PD72120,  $\mu$ PD72123) 単独では解決できませんので、お客様の責任において対応策をご検討の上、アプリケーション・システムを設計していただきますようお願いいたします。

本版で改訂された主な箇所

箇所	内容
p.4	Q1.7を追加
p.15	Q3.4を追加
p.29	Q3.24を追加
p.56	Q4.16を追加
p.92	Q5.34を追加
p.93	Q5.35を追加
p.93	Q5.36を追加
p.93	Q5.37を追加
p.94	Q5.38を追加
p.111	Q6.25を追加
p.115	Q6.30を追加
p.119	Q6.36を追加
p.125	Q6.42を追加
p.141	Q6.56を追加
p.146	Q6.63を追加

本文欄外の★印は、本版で改訂された主な箇所を示しています。

巻末にアンケート・コーナーを設けております。このドキュメントに対するご意見をお気軽にお寄せください。

## はじめに

### ○対象者

このインフォメーションは、AGDCの使用を検討されている、または応用システムを設計されているユーザの技術者を対象としています。

### ○目的

このインフォメーションは、ユーザの方からご質問をいただいた内容をまとめたものです。製品の使用をご検討中、または応用システムの設計中に不明な点が出た場合にご参照いただくことを目的としています。

### ○構成

このインフォメーションは、AGDC編です。Q&Aは、目次要約に示す分類に従ってまとめてあります。

### ○読み方

不明な内容を目次より索引して参照してください。

このインフォメーションを使用する場合、必ず最新のユーザズ・マニュアル、データ・シートをあわせてご参照ください。

このインフォメーションを読むにあたっては、論理回路やマイクロコンピュータに関する一般的知識を必要とします。

特に断らないかぎり、AGDCは $\mu$ PD72120、72123の両方を示します。

### ○関連資料

#### $\mu$ PD72120

- ユーザズ・マニュアル (IEM-5038)
- データ・シート (IC-7528)
- アプリケーション・ノート (I) (IEA-616)
- アプリケーション・ノート (II) (IEA-632)

#### $\mu$ PD72123

- ユーザズ・マニュアル (IEU-758)
- データ・シート (IC-7967)
- アプリケーション・ノート (ハードウェア編) (IEA-678)
- アプリケーション・ノート (ソフトウェア編) (IEA-711)
- アプリケーション・ノート (ウィンドウ編) (IEA-712)

**保守 / 廃止**

{メ モ}

## 目 次 要 約

第1章	基本システム構成	…	1
第2章	リセット	…	5
第3章	ホストCPUとのインタフェース	…	7
第4章	メモリ制御とバス・アービトレーション	…	40
第5章	表示制御	…	60
第6章	描画制御	…	96

[メ 毛]

## 目 次

第1章	基本システム構成	… 1
Q1.1	プレーン構成とピクセル構成のどちらがAGDC応用システムにマッチする？	… 1
Q1.2	プレーン構成の利点は？	… 2
Q1.3	AGDCをバケット・ピクセル・モードで使用すると、コマンドに制約がある？	… 2
Q1.4	$\mu$ PD72120と $\mu$ PD72123との違いは？	… 3
Q1.5	CLK, SCLKについて	
	(1) 動作周波数と消費電力とは関係ある？	… 3
	(2) 表示プロセッサを使わなければSCLKを供給しなくてもいい？	… 3
	(3) 動作中に動作周波数を変更してもいい？	… 3
Q1.6	IC端子の扱いは、どうすればいい？	… 4
Q1.7	SCLK=dotCLKでAGDCは使用可能か？	… 4
第2章	リセット	… 5
Q2.1	電源電圧安定後におけるRESET信号の立ち上がり/立ち下がり時間の最大値は？	… 5
Q2.2	リセット中にはクロックを供給しなければならない？また、供給するとすれば、それはCLK, SCLKのどちら？	… 5
Q2.3	リセット中 (RESET入力がハイ・レベル) の各入力端子の状態は？	… 6
Q2.4	リセット時、入力信号 $\overline{CSIR}$ , $\overline{CSDM}$ , $\overline{DMAAK}$ をどうすればいい？	… 6
第3章	ホストCPUとのインタフェース	… 7
Q3.1	AGDCとホストCPUとの接続例は？	… 7
Q3.2	$\overline{RD}$ , $\overline{WR}$ 入力がインアクティブのとき、 $\overline{CSIR}$ , $\overline{CSDM}$ 端子にグリッジを入力したらどうなる？	… 14
Q3.3	$\overline{CSIR}$ , $\overline{CSDM}$ と $\overline{ASTB}$ のタイミングに規定がないが、 $\overline{ASTB}$ より $\overline{CSIR}$ , $\overline{CSDM}$ が遅れてアクティブ (ロウ・レベル) になってよいか？	… 14
Q3.4	$\overline{RD}$ , $\overline{WR}$ 信号がアクティブでチップ・セレクト入力信号 ( $\overline{CSIR}$ , $\overline{CSDM}$ , $\overline{DMAAK}$ ) がインアクティブの場合、AGDCはどのような動作をするのか？	… 15
Q3.5	ホストCPUがAGDCをアクセスしないとき、 $\overline{ASTB}$ を入力していい？	… 15
Q3.6	システム・バスのリード/ライト・サイクルで、 $\overline{RD}$ , $\overline{WR}$ と $\overline{ASTB}$ のタイミングはどうなる？	… 15
Q3.7	システム・バス上でのDMA転送時、8/16ビットの切り替え方法は？	… 16
Q3.8	表示メモリ・アクセスのための $\overline{DLBE}$ 出力は、表示メモリへのアドレス出力DAD0の値と同じ？	… 17
Q3.9	システム・バス側を8ビット・データ・バス ( $\overline{UBE}=\text{H}$ ) にして、DMA転送する場合、表示メモリ側のD0-D15とシステム・バス側のAD0-AD7の関係は？	… 17
Q3.10	描画レジスタの内容を読む場合、READY信号は必ずインアクティブになる？	… 18
Q3.11	プリプロセッサ動作中にステータス・レジスタを読み出そうとしても、READYがインアクティブ状態のままになってしまい、ステータス・レジスタを読み出せないのでは？	… 18
Q3.12	$\overline{WR}$ 信号とREADY信号との関係は？	… 18

- Q3.13 プリプロセッサが、ノンビジィのときに描画パラメータを書き込むと、READY信号はど  
うなる？ … 22
- Q3.14 PAINTコマンドを実行するとREADYが長い間ロウ・レベルになる？ … 22
- Q3.15  $t_{DRRY}$ ,  $t_{DWRV}$ はMAX.30 ns ( $\mu$ PD72120(A)では40 ns)となっているが、これは、そのサ  
イクルでAGDCがビジィ状態であれば、リード/ライト後必ず30 ns以内にREADY信号は  
ロウ・レベルに移行するということ？ … 23
- Q3.16 ホストDMA時のREADY出力と、システム・バス上の優先順位の関係は？ … 23
- Q3.17 描画プロセッサが動作中にCSDMによりメモリ・アクセスができる？ … 25
- Q3.18 AGDCが描画中にDISPLAY\_CTRLレジスタの内容を書き換えてもいい？ … 25
- Q3.19 AGDCが複雑な描画などを行っている場合、ホストCPUからの表示メモリ・アクセス(CSDM  
による)に対するREADY=Lの期間は？ … 25
- Q3.20 AGDCに描画コマンドを発行しても描画動作が行われない場合、原因は？ … 26
- Q3.21 ホストCPUがAGDCをアクセスしても、AGDCのREADY信号がアクティブにならない場  
合、原因は？ … 26
- Q3.22 V50のDMAで表示メモリの内容をメイン・メモリに転送する場合、AGDCとV50の各READY  
端子のインタフェースは？ … 28
- Q3.23 INT信号がアクティブ、インアクティブになるタイミングは？ … 29
- Q3.24 AGDCによるINT割り込みの発生と同時にステータス・リードを行うと、INT信号はクリ  
アされてしまうのか？ … 29
- Q3.25 AGDCのステータス・レジスタについて  
(1) ステータス・レジスタのビジィ・フラグは、同時にアクティブになることがある？ … 30  
(2) ステータス・レジスタでエラーを検出したらどういう処理をすればいい？ … 30
- Q3.26 CTRLレジスタ内のPBIEをイネーブルするタイミングと、プリプロセッサがビジィから  
ノンビジィへと変化するタイミングがあった場合の、INT信号との関係について  
(1) ノンビジィとなるタイミングが早い場合、INT信号はONになる？ … 30  
(2) ノンビジィとなるタイミングが遅い場合、INT信号はONになる？ … 30
- Q3.27 AGDCのPBIE, DBIEの基本的な使い方は？ … 31
- Q3.28 V40(8ビット・データ・バス)で16×16ビット構成のPGPORTにDMAを行う場合、外付  
け回路は不要？ … 32
- Q3.29 PUT/GETコマンド実行時のフラグ・チェックは、ステータス・レジスタのPPBSYとPGRDY  
だけでいい？ … 33
- Q3.30 8ビット・ホストCPUをAGDCに接続する場合の注意点は？ … 35
- Q3.31 モトローラ系の16ビットCPUとの接続のとき  
(1) データ・バス接続方法は？ … 36  
(2) バイト・アクセスもできる？ … 36
- Q3.32 マルチAGDCシステム(マスタAGDCでスレーブAGDCを制御)でのバスの制御方法  
は？ … 37
- Q3.33 コマンド終了をDPBSYのハードウェアINTで検出できる？ … 39
- 第4章 メモリ制御とバス・アービトレーション … 40**
- Q4.1 表示画面上で、LSBとMSBはそれぞれ左右どちら？ … 40
- Q4.2 表示メモリ用のリフレッシュ機能で、システム・バス上のメモリも同時にリフレッシュ  
できる？ … 41
- Q4.3 表示メモリをアクセスしていないとき、DLBE, DUBE信号はどうなる？ … 41
- Q4.4  $\mu$ PD24C2000-X11/-X12との接続について  
(1) 接続例は？ … 42

- (2) 複数の書体を持たせたいが、どうすればいい? ... 42
- Q4.5 漢字CGを表示メモリ領域に割り当てた場合、CLK = 8 MHzでは、CGのアクセス・スピードが不足? ... 45
- Q4.6 表示メモリのアクセスを8ビットにできる? ... 47
- Q4.7 表示メモリ・バスにバイト単位かワード単位のCGを割り当てたとき  
 (1) コピー・コマンドでのデータ転送方法は? ... 47  
 (2) ホストからデータを読み出す方法は? ... 47
- Q4.8 表示メモリ・バス上のHLDRQを最高優先度にする方法は? ... 51
- Q4.9 描画サイクルの起動を待たせるフラグ/コマンドはある? ... 51
- Q4.10 リフレッシュ・サイクル直前にHLDRQを入力した場合どうなる? ... 52
- Q4.11 HLDRQ入力直後にWAIT(リフレッシュ要求)がアクティブになると、 $\overline{\text{HLDAK}}$ はどうなる? ... 53
- Q4.12 データ・トランスファ・モードにおいて、DTサイクル終了から描画サイクル開始までの最小時間は? ... 54
- Q4.13 WAIT信号は、 $\overline{\text{DT}}$ 出カタイミングの4SCLK前にアクティブとなる? ... 54
- Q4.14 WAIT信号がインアクティブとなるタイミングは? ... 55
- Q4.15 表示メモリをAGDC以外のモジュールでリフレッシュ動作を行う場合、AGDCの描画サイクルとのアービトレーションを行う方法は? ... 55
- Q4.16 描画サイクル中にHLDRQ信号がアクティブになったときのタイミングは? ... 56
- Q4.17  $\mu\text{PD72123}$ を描画アクセラレータとして使用する場合、DWAITのみで、 $\mu\text{PD72123}$ に通常グラフィクス・バスを占有させないで、バス・アービトレーションを行うことは可能? ... 57
- Q4.18 HLDRQ入力信号がアクティブのときに、CPUよりホスト・ダイレクト・アクセスがなされた場合の $\mu\text{PD72123}$ のグラフィクス・バス・アービトレーションはどのようなタイミングになる? ... 58
- Q4.19  $\overline{\text{HLDAK}}$ 出力時の各端子のレベルは? ... 59
- Q4.20 CSモードでも、ウエイト・サイクルを挿入できる? ... 59

## 第5章 表示制御 ... 60

- Q5.1 DTモードでのSCLK, DTT, HD, WC, DAD+の設定方法は? ... 60
- Q5.2 WC, DAD+, HDの設定方法は? ... 63
- Q5.3 ソース側とデスティネーション側のPITCHが違うとき、DISPLAY\_PITCH, WC, DADの設定はどうする? ... 65
- Q5.4 DAD+フラグを「DAD+1からDAD+2」、SCLKを従来の2倍遅い周波数に変更した場合、WCレジスタ設定値はどう変わる? ... 65
- Q5.5 表示画面には、プレーン間の変位量を設定するレジスタはある? ... 66
- Q5.6 表示アドレスの更新タイミングは? ... 67
- Q5.7 PPBSY=1のときに、表示プロセッサのレジスタを書き換えても動作上問題はない? ... 68
- Q5.8 表示中 (BLANK=L) にDADレジスタの内容を書き換えると  
 (1) いつ表示が変わる? ... 69  
 (2) 表示が乱れる? ... 69
- Q5.9 表示制御もリフレッシュ制御もAGDCにさせないとき、表示パラメータは設定しなくてもいい? ... 69
- Q5.10 水平表示ドット数1024で $\mu\text{PD42264}$ を使用する場合、DAD+レジスタの設定は? ... 70
- Q5.11 サイクル・スチール・モード時、DAD+とリフレッシュ・サイクル/表示サイクル/描画サイクルの関係は? ... 70
- Q5.12 VS, またはVFPの期間でもリフレッシュをする? ... 75

- Q5.13 DTモードにおいて、CLKとSCLKが非同期でいい？ … 76
- Q5.14 AGDCをインタレース・モードに設定し、転送タイミングのDTTを1に設定した場合、画面上の各走査線の表示開始時（BLANK信号立ち下がり時）は、絶対に転送は行われない？ … 76
- Q5.15 サイクル・スチール・モード（SCLK=CLK）の表示では、クロック（CLKまたはSCLK）のデューティが50%でなくなることがあるけど、問題ない？ … 76
- Q5.16 水平走査（ドット・クロック）の遅いシステムへの対応について  
 (1) SCLK, CLKの動作範囲（600 nsMAX.）を越えたらどうなる（ただし、 $\mu$ PD72123のSCLKは2000 nsMAX.）？ … 77  
 (2) 対策は？ … 77
- Q5.17 AGDC外部同期について  
 (1) ソースが同じSCLKで、2つのAGDCを動かしてスーパーインポーズする場合の注意点は？ … 78  
 (2) ソースが異なるSCLKで、2つのAGDCを動かしてスーパーインポーズする場合の注意点は？ … 78
- Q5.18 外部同期モード（スレーブ・モード）において、同期信号のずれを補正するため、補正期間だけSCLK信号を停止させていい？ … 79
- Q5.19 外部同期モード時、EXVS, EXHS入力のタイミング規定は？ … 80
- Q5.20 外部同期モード設定後、EXVS, EXHSに従って動作するタイミングは？ … 80
- Q5.21 外部同期モード時、表示サイクルのD1, D2を同期させるには？ … 81
- Q5.22 AGDC間の同期は、ドット・クロック単位であわせられる？ … 82
- Q5.23 外部同期モード時、SCLKの立ち上がりで動作すると説明されているけど、立ち下がりは無視される？ … 83
- Q5.24 内部同期-外部同期（マスター-スレーブ）間の切り替えを行うと、描画メモリの内容に影響する？ … 83
- Q5.25 プリンタ・スレーブ・モードにおけるメモリ・リフレッシュは、どうなる？ … 84
- Q5.26 AGDCにバイト・アクセスする場合、同期パラメータの設定手順は？ … 84
- Q5.27 AGDCの同期パラメータの設定において、バイト書き込みでパラメータを設定したが、表示画面の同期がとれない。原因は？ … 85
- Q5.28 インタレース・モードでの同期パラメータの設定は？ … 85
- Q5.29 表示開始の設定について  
 (1) CRT同期信号設定レジスタ（HS, HBS, HH, HFP）の設定を4クロック以上にするのはなぜ？ … 88  
 (2) DTモード時、BLANK信号の立ち上がりでDT信号が発生して表示サイクルになるため、CRTが表示を開始するのは、HSYNC信号の立ち上がりから6クロック以上となるのでは？ … 88
- Q5.30 高解像度のCRTにも対応できる？ … 88
- Q5.31 LCDの表示制御はできる？ … 89
- Q5.32 表示画面の設定（プレーン・モード）において、表示アドレス変更のみでスクロールはできる？ … 89
- Q5.33 AGDCの水平スクロールは、サイクル・スチール・モードでも使用できる？ … 91
- Q5.34 SYNCパラメータ（HS, HBP, HH, HD, HFP, VS, VBP, L/F, VFP）の設定条件は？ … 92
- Q5.35 スレーブ・モードのとき、HSパラメータの設定値を0003H以下にしたらどうなる？ … 93
- Q5.36 ノン・インタレース・モードのとき、SYNCレジスタのHHパラメータはどのように設定したらいい？ … 93

- Q5.37 グラフィック・カーサを表示する場合、カーサの形状や大きさはどのようにして決めるのか？ … 93
- Q5.38 NTSCコンポジット信号とAGDCの表示画面をスーパーインポーズしてテレビ画面に表示させる方法は？ … 94

## 第6章 描画制御 … 96

- Q6.1 使用する予定のないレジスタの設定はしなくてもいい？ … 96
- Q6.2 EADORG, dADORG(座標原点)を画面左上にセットしたとき、Y座標の設定は？ … 96
- Q6.3 描画領域と表示領域を独立に設定できる？ … 97
- Q6.4 プレーンの合成表示はできる？ … 97
- Q6.5 AGDCに対し、インバリッド・コマンドを発行するとAGDCはどうなる？また、このときの状態を知る方法は？ … 97
- Q6.6 直線描画を行ったときに、穴空きやゴミが表示されてしまう。表示メモリにもそのようなデータが書き込まれているが、考えられる原因は？ … 98
- Q6.7 表画面と裏画面の大きさを変えられる？ … 98
- Q6.8 書き込みを行うプレーン数は、P<sub>MAX</sub>レジスタだけで制限される？ … 98
- Q6.9 コマンド・パラメータとして規定外の値を設定してもハングアップしない？ … 99
- Q6.10 三角形の塗りつぶしを行う場合に、短辺をy軸に平行すると描画されない部分が発生してしまうのはどうして？ … 99
- Q6.11 横方向に1ドットずつスクロールする方法は？ … 100
- Q6.12 拡大／縮小を任意の倍率で行う方法は？ … 100
- Q6.13 拡大／縮小コピーについて変倍後のサイズの算出方法は？ … 101
- Q6.14 文字フォントの拡大+傾斜を行いたいが、処理速度、ジャギーなどを考慮すると、拡大コピーと傾斜コピーのどちらを先に実行すべき？ … 101
- Q6.15 4点ORによる1/2縮小の方法は？ … 102
- Q6.16 スタック領域について
- (1) 文字のアウトライン内を塗りつぶすには、どのくらい必要？ … 105
  - (2) PAINTコマンドを実行するには、どのくらい必要？ … 105
  - (3) 塗りつぶしパターンによって、スタックの消費量は変化する？ … 105
  - (4) プレーン数を増やすと、スタック領域も広げる必要がある？ … 105
- Q6.17 直線描画について
- (1) 始点と終点が同一座標のとき、描画を行う？もし行わないなら、レジスタの変化は通常と同じ？ … 106
  - (2) そのとき、線幅を拡張しているとうなる？ … 106
- Q6.18 ライン描画コマンドのパラメータ設定で
- (1) A<sub>LINE</sub><sub>D0</sub>からA<sub>LINE</sub><sub>D2</sub>で「(X#, Y#) = (X, Y) であること」とあるが、(X#, Y#) ≠ (X, Y) のとき、どうなる？ … 107
  - (2) R<sub>LINE</sub><sub>D0</sub>からR<sub>LINE</sub><sub>D2</sub>では、(X#, Y#) = (X, Y) でなければならない？ … 107
- Q6.19 直線描画コマンドでプリプロセッサ・エラーは発生する？ … 107
- Q6.20 32ビット線種パターンの選択方法は？ … 108
- Q6.21 円、楕円描画時の線幅拡大方法は？ … 108
- Q6.22 太線（8ビット、16ビットなど）で直線以外の線図形（円、楕円など）を描画する方法は？ … 109

- Q6.23 1点から3方向に直線を描画する場合、A\_LINE\_M1(D1)では命令実行後の描画ポイント(X#, Y#)が描画終了座標となり、1回の直線描画で(X, Y)≠(X#, Y#)のため、描画不可能と思えるけど、どうすればいい? … 109
- Q6.24 右上がりの直線をコマンドA\_LINE\_Mで描画した場合、直線中央部に段差ができることがあるのはなぜ? … 110
- Q6.25 直線は最大何ドットまで描画できるのか? … 111
- Q6.26 円弧描画について
- (1) (xs, ys) = (-α, β)という座標を指定したにもかかわらず、AGDCが(α, β)という座標より描画を開始することがある? … 112
  - (2) 半径rであるような円弧描画を行うときその円弧内に描画開始座標(xs, ys)を指定した場合、AGDCはどのような判定を行いながら円弧を描画する? … 112
- Q6.27 クリッピングが使用できないコマンドは? … 113
- Q6.28 クリッピングとFASTフラグの関係は? … 114
- Q6.29 任意閉領域塗りつぶし(PAINT)を実行する場合、クリッピング領域を指定していると、クリッピング境界線の塗りつぶしはどうか? … 115
- Q6.30 PAINTコマンドでクリッピング領域と円弧描画コマンドで囲まれた閉領域を塗りつぶすことが可能? … 115
- Q6.31 クリッピングの際の描画無効領域に対して
- (1) 読み出したビット・データをそのまま同じ番地書き戻す? … 116
  - (2) 描画ライト・サイクルを起動しないかを外部から指定できる? … 116
- Q6.32 AGDCは、メモリが実装されているかどうか意識して描画サイクルを起動する? … 116
- Q6.33 タイリング(パターンによる塗りつぶし)にPTNCNTレジスタを使用しないとき、外部定義パターンを表示メモリのどこに設定すればいい? … 117
- Q6.34 PTNCNTレジスタを塗りつぶしパターンとして使用する場合、パターンの連続性は? … 117
- Q6.35 塗りつぶしパターンを表示メモリ参照方式にした場合
- (1) タイリング・パターンを32×32ドットで展開する方法は? … 118
  - (2) どの位置から塗りつぶしても、パターンの連続性は保たれる? … 118
- Q6.36 図形をTRI\_FILLコマンドとA\_REC\_FILLコマンドを用いて描画し、境界線上での塗りつぶしパターンの連続性は保てるか? … 119
- Q6.37 3オペランド・コピーの際のマスク領域のメモリ幅(ピッチ)はどのように設定すればいい? … 120
- Q6.38 コピー・コマンド受け付け後、メモリ・プレーン・アクセスまでのクロック数は? … 121
- Q6.39 回転コピーと塗りつぶしについて、次の方法で楕円内部を塗りつぶせる? … 122
- ① 線幅1ドットで楕円を描画
  - ② 45°の回転コピーを実行(境界点検索をする塗りつぶしコマンド)
  - ③ 回転した楕円にPAINTコマンドを実行
- Q6.40 コピー・コマンドにおいて
- (1) ソース・データの右下を読み出し開始点(ESE=1)にしたらどうなる? … 123
  - (2) 表示メモリの同一アドレスに同一データをコピーできる? … 123
  - (3) 表示メモリの同一アドレスに空白データをコピーできる? … 123
- Q6.41 コピー・コマンドでデータの鏡像反転(REV=1)を選択し、転送源と転送先の位置を等しくしてコピーを実行した場合、コピーされたデータの値は保証される? … 124
- Q6.42 転送源矩形領域と転送先矩形領域が重なるときの単純コピーは実行できる? … 125
- Q6.43 縮小コピーで、ソースとデスティネーションを同じ領域にすることはできる? … 125
- Q6.44 コピー・コマンドの際、転送先メモリプレーンの横幅が10ワードのとき、そこへ横幅20ワードのデータを転送しようとした場合、クリッピングで横幅を10ワードに制限しておけば正常なコピー動作が行われる? … 126

- Q6.45 任意角回転コピーにおいて、FSフラグを1にすると、どの点を描画する？ … 127
- Q6.46 拡大／縮小コピーを行うときに、任意角回転・拡大／縮小コピーを用いても拡大／縮小できる？ … 130
- Q6.47 ソース領域にデータがあったとして、●のデータ領域分をデスティネーション領域にコピーしたい場合、AGDCのワード単位のコピーはどのように行う？ … 131
- Q6.48 実装メモリ空間を越えて三角形を描画するとどうなる？ … 132
- Q6.49 漢字ROMについて
- (1) 容量、アドレスなどには、制限がある？ … 133
  - (2) アプリケーション・ノートで、漢字ROMの配置を工夫している？ … 133
  - (3) 24×24ドットのフォント展開はできる？ … 133
- Q6.50 漢字ROMから文字を表示メモリに転送した後、ソフトで表示をON、OFFさせるのではなく、AGDCの機能でCRTに点減させて表示できる？ … 133
- Q6.51 文字データのあるバック・グランド上へ転送する方法は？ … 134
- Q6.52 キャラクタへのアトリビュート(下線、白黒反転、ブリンクなど)の付け方は？ … 136
- Q6.53 90°回転コピーとデスティネーションのアドレス発生について
- (1) 読み出し開始位置が左上の場合の90°回転コピーはどうなる？ … 138
  - (2) 読み出し開始位置が右下の場合の90°回転コピーはどうなる？ … 139
- Q6.54 90°回転と1/2縮小をするとき
- (1) 最も速い処理方法は？ … 140
  - (2) そのときのスピードは？ … 140
  - (3) そのときのメモリ領域の取り方は？ … 140
- Q6.55 ショート・ベクタ(線幅4ビット、X軸に対し45°方向へ40ドットの線分)の描画速度は？ … 141
- Q6.56 ピクセル単位の互換性について
- (1) 同一のコマンド、かつ同一のパラメータを用いてAGDCに描画をさせる場合には、いつ描画させても実行結果は同じ？ … 141
  - (2) 直線描画コマンドで描画開始点と描画終了点を入れ替えても描画結果は同じ？ … 141
- Q6.57 代表的な描画コマンドの描画速度は？ … 142
- Q6.58 システム・メモリ上にあるデータを表示メモリに取り込むには？ … 143
- Q6.59 PUT, GETコマンドを使うときの注意点は？ … 144
- Q6.60 AGDCのPUT, GETの転送スピードは？ … 144
- Q6.61 描画コマンドのスピードについて
- (1) 描画コマンドでCSECなどは、CARC+LINEという処理速度？ … 145
  - (2) パラメータ設定などにより描画速度が変わるということだが、どのくらいの差がある？ … 145
- Q6.62 線描画コマンドは1ドットずつ描画する？ … 145
- Q6.63 GETコマンドにおいて複数プレーンのデータを論理演算してシステム・メモリにブロック転送するときのSD\_SEL, MODO, MOD1, PLANESの設定方法は？ … 146
- Q6.64 ソフトウェア・カーサの実現方法は？ … 147
- Q6.65 ピクセル構成で使用する場合の描画方法は？ … 148

(× ㊦)

図 の 目 次

3-1 リードの立ち下がりタイミング … 27

3-2 連続でPUTコマンドを発行した場合のシーケンス … 34

4-1 ROMアドレスの制御 … 43

4-2 AGDCへの接続例 … 44

4-3 表示サイクル（リフレッシュ・サイクル）の直前にHLDRQが入った場合のタイミング … 52

4-4 リード・サイクルの場合のタイミング … 58

5-1 1 HSYNC周期でのn表示サイクル … 63

5-2 表示アドレス … 64

5-3 サイクル・スチール・モード時の表示タイミング … 71

5-4 表示タイミングの画面との対応 … 71

5-5 サイクル・スチール・モード時のリフレッシュ・タイミング … 72

5-6 DAD+1/2のときの表示タイミング … 73

5-7 DAD+1/4のときの表示タイミング … 74

5-8 ノン・インタレース・モード時の表示アドレスの進み方 … 74

5-9 インタレース・モード時の表示アドレスの進み方 … 75

5-10 SCLK EXVSおよびEXHSの関係 … 81

5-11 EXVSによるD1/D2カウンタの遷移 … 82

5-12 水平方向ドット単位スクロール用回路例 … 90

5-13 NTSCコンポジット信号インタフェース・ブロック図 … 95

6-1 TL=0時の塗りつぶし実行例（論理演算はREPLACE） … 117

6-2 32×32ドットのパターンをタイリング・パターンとしての使用例（COPYを5×6=30実行） … 118

6-3 45°回転コピーと塗りつぶし … 122

6-4 ESE=1としたときのコピー・スキャン・アドレス … 123

6-5 DDAによる直線描画 … 127

6-6 長方形を任意角回転コピーした例 … 128

6-7 FSフラグを1にしたときの直線描画 … 129

6-8 図6-6(a)の図をFS=1として任意角回転コピーを行った場合 … 130

6-9 折り返しが生じる場合 … 132

6-10 AGDCのクリッピング機能を使用した場合 … 132

6-11 バック・グラウンド・データヘフォア・グラウンド・データをコピーするときの論理演算 … 134

6-12 PLANESレジスタへ黄色を設定した例 … 135

6-13 読み出し開始位置が左上の場合の90°回転コピー … 138

6-14 読み出し開始位置が右下の場合の90°回転コピー … 139

表 の 目 次

5-1 DAD+とDADDの関係 … 73  
5-2 SCLKの電気的特性 … 79

## 第 1 章 基本システム構成

<b>Q1.1</b>	内容は？	<input checked="" type="checkbox"/> 機能・動作 <input type="checkbox"/> 特性 <input type="checkbox"/> 応用
プレーン構成とピクセル構成のどちらがAGDC応用システムにマッチする？		

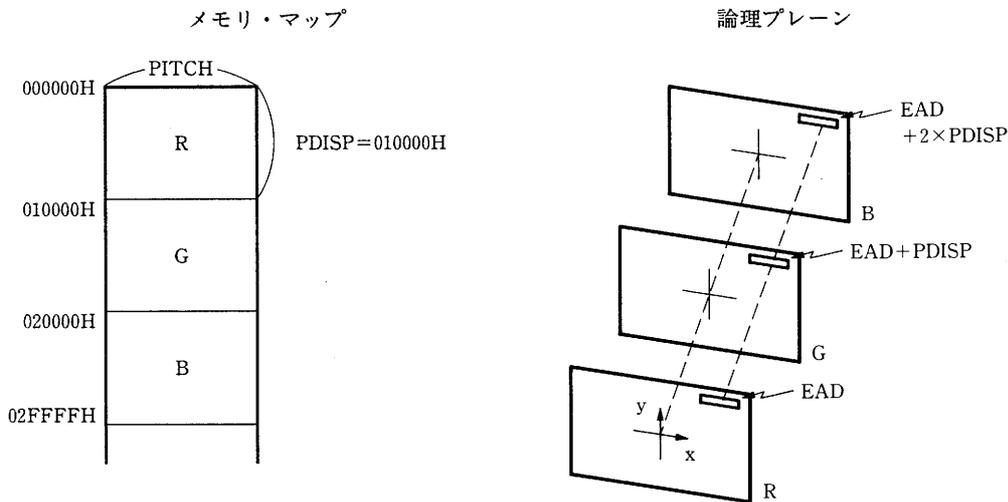
**A1.1**

AGDCはプレーン描画を基本とするシステムに主眼をおいた設計となっています。  
 ピクセル構成については、円、直線のみを描画であればプレーン構成より有効ですが、塗りつぶし、コピー等がシステム上、必要であればプレーン構成としてください。  
 プレーン構成におけるAGDCの描画シーケンスを下図に示します。

1. プレーン描画シーケンス

(条件)

メモリ・プレーン R, G, B 各1枚



2. プレーン描画のアドレス発生

- 第1プレーンの描画アドレス = EAD
- 第2           "           = EAD + PDISP
- 第3           "           = EAD + 2 × PDISP

AGDCの描画レジスタは次のような意味をもっています。

内 容	ソース	デスティネーション
論理プレーン横幅	PITCHS	PITCHD
論理プレーン・ディスプレイメント (論理プレーン間距離)	PDISPS	PDISPD

### Q1.2

内容は？

機能・動作 特性 応用

プレーン構成の利点は？

### A1.2

1. 二次元データの処理（塗りつぶし、コピー、回転、拡大、縮小）を1メモリ・アクセスで16ビット処理できるので、描画スピードが速い。
2. プレーンごとにデータをアクセスすることができるので、マスク処理が簡単にできる。

### Q1.3

内容は？

機能・動作 特性 応用

AGDCをパケット・ピクセル・モードで使用すると、コマンドに制約がある？

### A1.3

制約があります。PXEN, BPPXフラグがないコマンドは、使用できません。

可能なコマンドとしては、点描画コマンド、長方形描画コマンド、円描画コマンド、楕円描画コマンドがあります。塗りつぶし、コピー、GET/PUT等のコマンドは使用できません。

また、ラスタ・オペレーション（論理演算）は、パケット・ピクセル・モードではPMAX=1の選択となるため、プレーン別に論理演算を定義することができません。

## Q1.4

内容は？

■機能・動作 □特性 □応用

μPD72120とμPD72123との違いは？

## A1.4

μPD72123のμPD72120に対する拡張機能は、次のとおりです。

- 1) ハードウェア面
  - ① 動作（クロック）周波数を8 MHz → 10 MHzに変更
  - ② 全アドレス空間（000000H-FFFFFFH）をホスト・ダイレクト・アクセス可能
  - ③ ビット・リバース機能の追加
  - ④ 描画サイクル中にウェイト・サイクルを挿入可能
- 2) ソフトウェア面
  - ① 3オペランド・コピー・コマンドの追加  
μPD72120では、任意形状のくりぬきのCOPYは、COPYコマンドが複数回必要でしたが、μPD72123では、このコマンドを1回発行するだけでCOPYができます。
  - ② グラフィクス・ペン機能の追加  
μPD72120では、幅のある円は1コマンドでは描画できませんでしたが、μPD72123では、このコマンドを発行するだけで描画できます。

備考 詳細は、「μPD72123のデータ・シート、ユーザーズ・マニュアル」を参照してください。

## Q1.5

内容は？

■機能・動作 □特性 □応用

CLK, SCLKについて

- (1) 動作周波数と消費電力とは関係ある？
- (2) 表示プロセッサを使わなければSCLKを供給しなくてもいい？
- (3) 動作中に動作周波数を変更してもいい？

## A1.5

- (1) 動作周波数が高いほど、消費電力は大きくなります。
- (2) いけません。  
表示プロセッサを使用しない場合でも、SCLK端子にはクロックを供給してください。
- (3) 途中で周波数を変更することは避けてください。動作途中で動作周波数を変更すると、グラフィクス・バス側において、データの読み出し/書き込みが正常にできなくなる可能性があります。

## Q1.6

内容は？

■機能・動作 □特性 □応用

IC端子の扱いは、どうすればいい？

## A1.6

デバイスの動作保証の面から、データ・シート、ユーザズ・マニュアルなどに記載のとおり、IC端子はオープン状態で使用してください。

ただし、AGDCのIC端子は、デバイス内部において、パッドとは接続されていません。

したがって、外部においてその端子をプルアップ、またはプルダウンしても特に問題はありません。

★

## Q1.7

内容は？

■機能・動作 □特性 □応用

SCLK=dotCLKでAGDCは使用可能か？

## A1.7

使用可能です。ただし次の制限事項がありますので注意してください。

- ・CSモードであること (DTモードでは使用できません)。
- ・SCLK=CLKであること。

## 第2章 リセット

### Q2.1

内容は？ 機能・動作 特性 応用

電源電圧安定後におけるRESET信号の立ち上がり／立ち下がり時間の最大値は？

### A2.1

RESET信号の立ち上がり／立ち下がり時間の最大値は50 ns (10 MHz動作時) を目安にしてください。CLKの立ち上がりでRESET信号をサンプリングしますので、この時のRESET信号を0か1に固定しなければなりません。AGDCはリセット入力ハイ・レベル保持時間、ロウ・レベル設定時間がそれぞれ30 ns、20 ns必要ですので、CLK=100 ns (10 MHz動作時) の場合、この時間は50 ns以内であれば問題ないと思われます。

### Q2.2

内容は？ 機能・動作 特性 応用

リセット中にはクロックを供給しなければならない？また、供給するとすれば、それはCLK, SCLKのどちら？

### A2.2

CLK, SCLKともに常時供給してください。

## Q2.3

内容は？

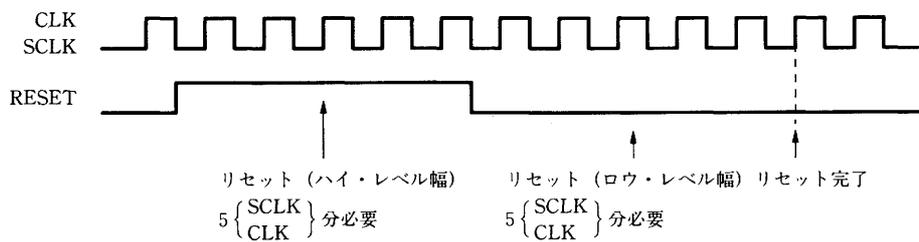
機能・動作
 特性
 応用

リセット中 (RESET入力がハイ・レベル) の各入力端子の状態は？

## A2.3

READY,  $\overline{\text{DRD}}$ ,  $\overline{\text{DWR}}$ ,  $\overline{\text{HLDAK}}$ はハイ・レベル, INT, DMARQ,  $\overline{\text{DUBE}}$ ,  $\overline{\text{DLBE}}$ はロウ・レベルとなります。また、これらの信号以外は不定です (ハードウェア設計上、注意してください)。

下図のようにリセット動作が実行され、このあとはじめてリセット完了となります (完了以前は不定です)。



## Q2.4

内容は？

機能・動作
 特性
 応用

リセット時、入力信号 $\overline{\text{CSIR}}$ ,  $\overline{\text{CSDM}}$ ,  $\overline{\text{DMAAK}}$ をどうすればいい？

## A2.4

いずれの入力信号も、インアクティブ ( $>V_{IH}$  (MIN.)) としてください。フローティング (未処置) 状態の場合、READYがインアクティブ状態を続けることがあります。

### 第3章 ホストCPUとのインタフェース

#### Q3.1

内容は？

機能・動作 特性 応用

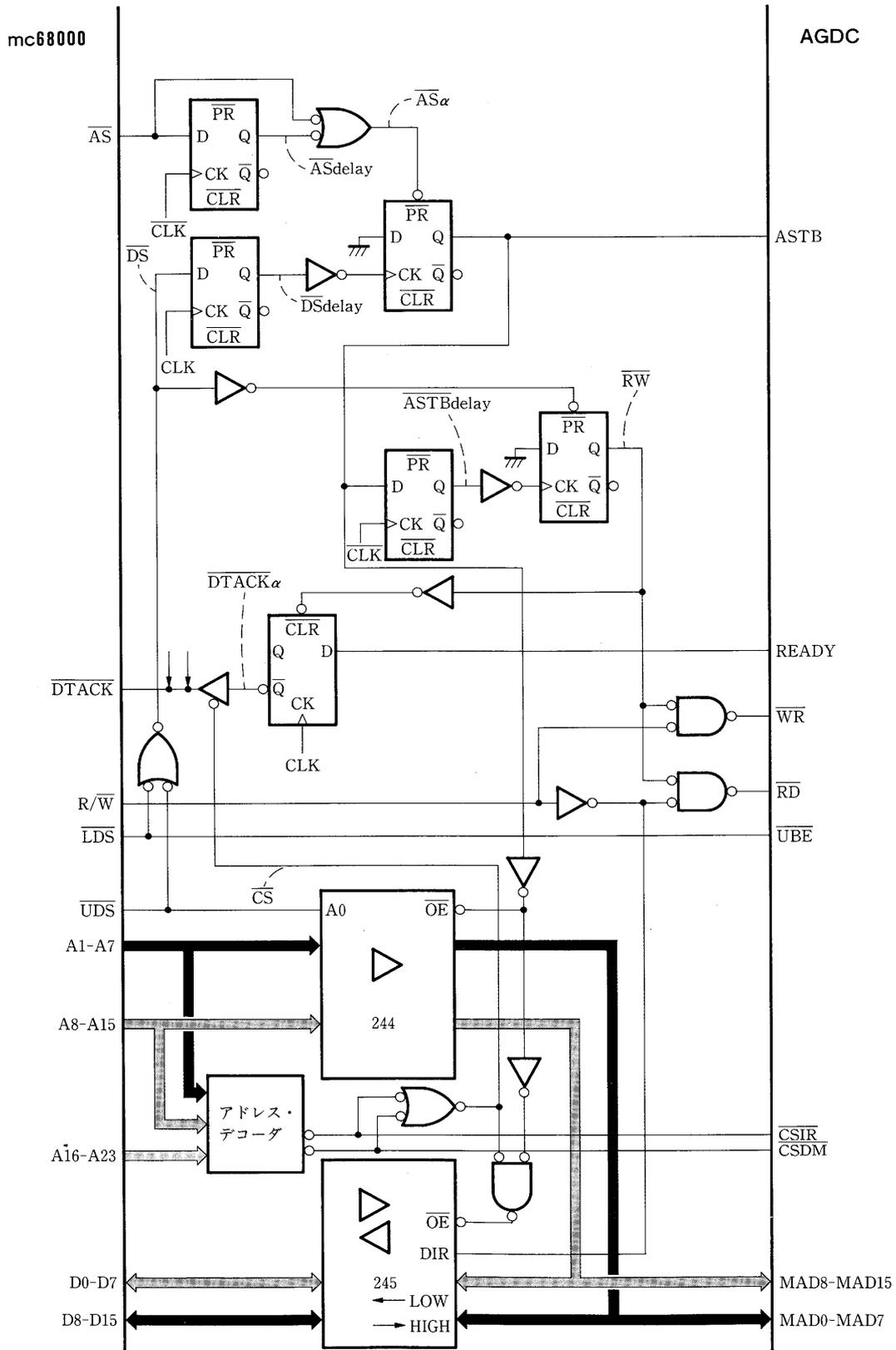
AGDCとホストCPU (①～⑥の場合) との接続例は？

- ① mc68000
- ② V25™
- ③ V33™
- ④ V40™
- ⑤ V50™
- ⑥ V53™

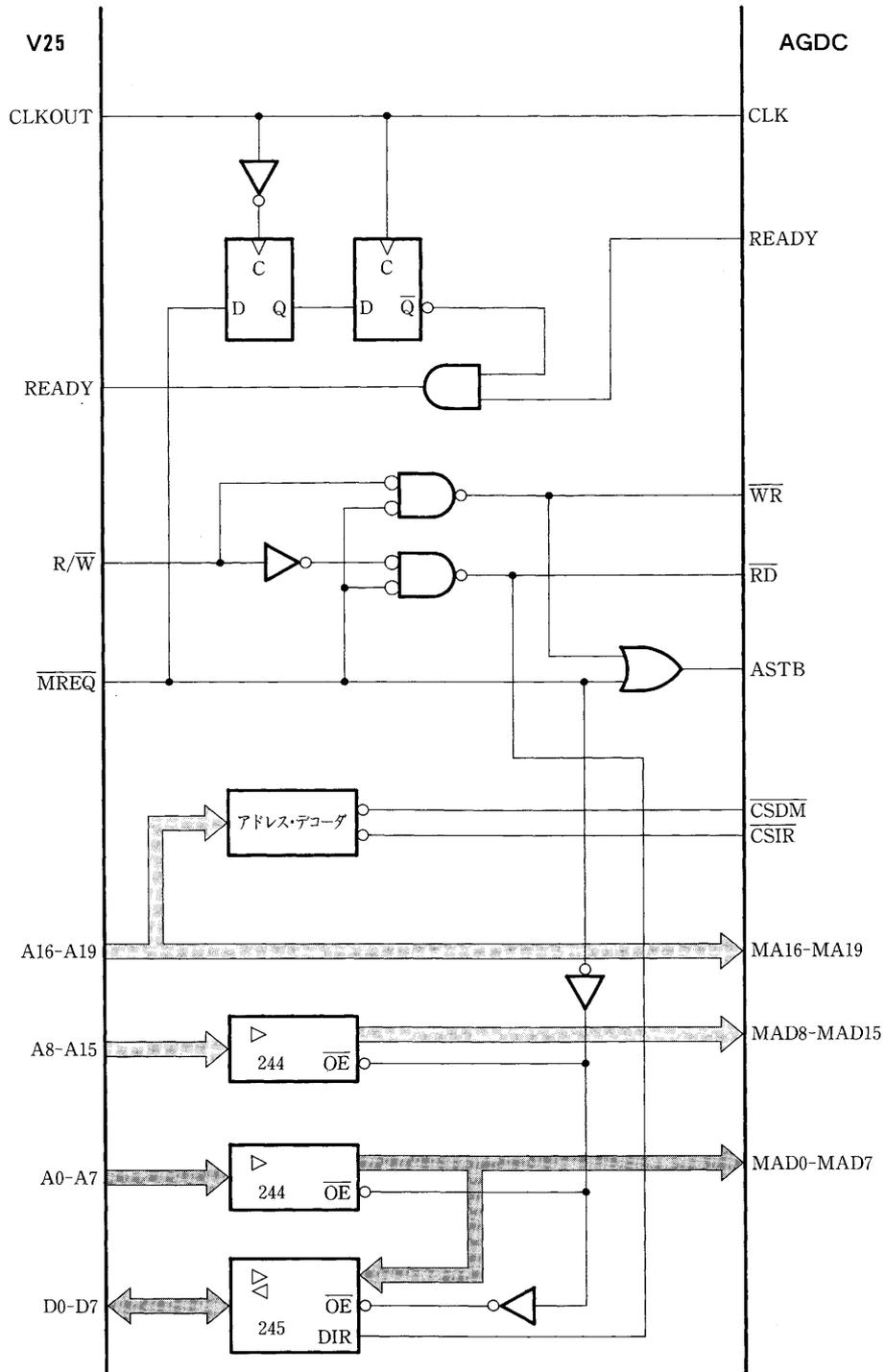
#### A3.1

上記 (①～⑥) の順で、接続例を次ページ以降に示します。

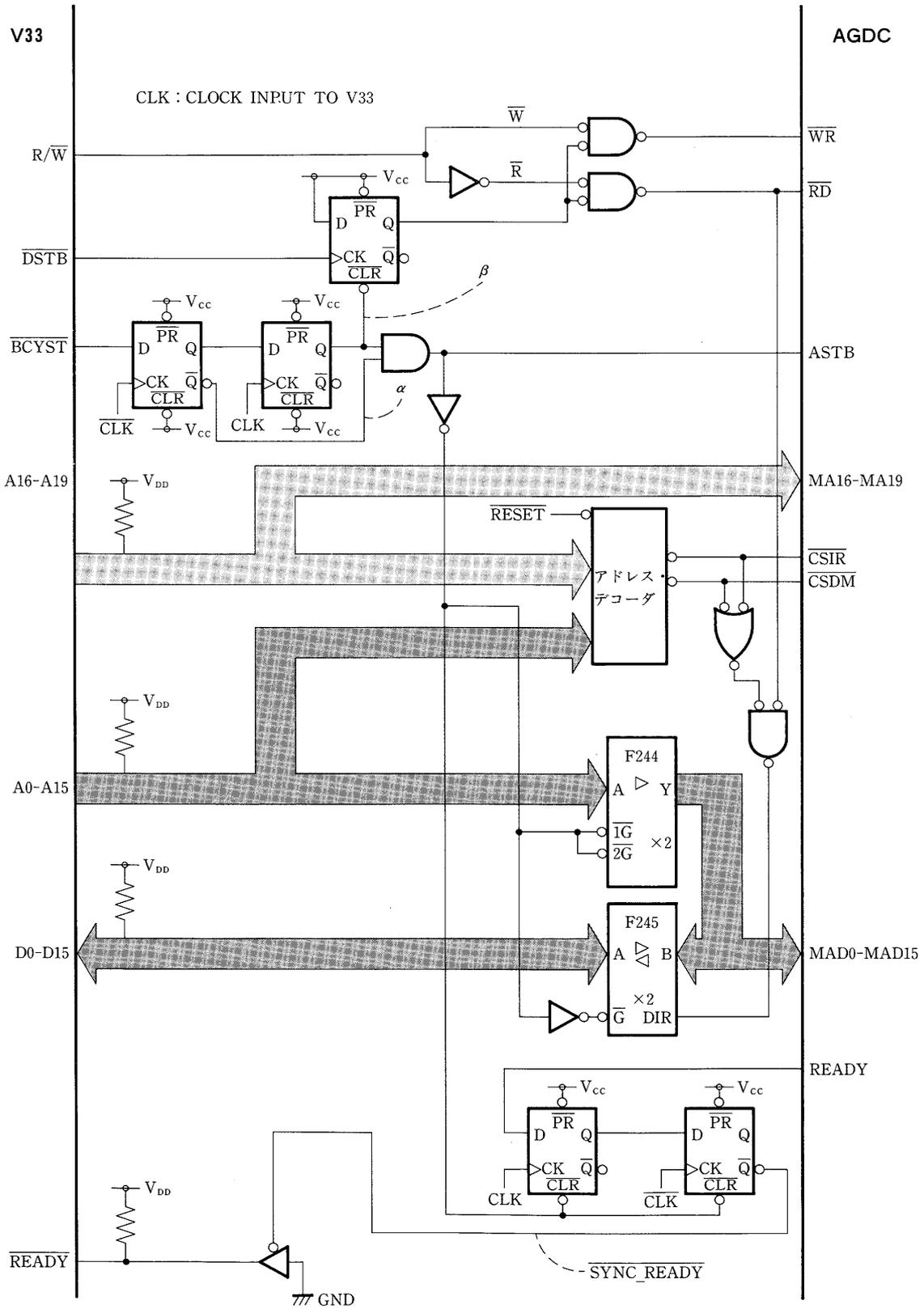
① mc68000とAGDCの接続例



② V25とAGDCの接続例

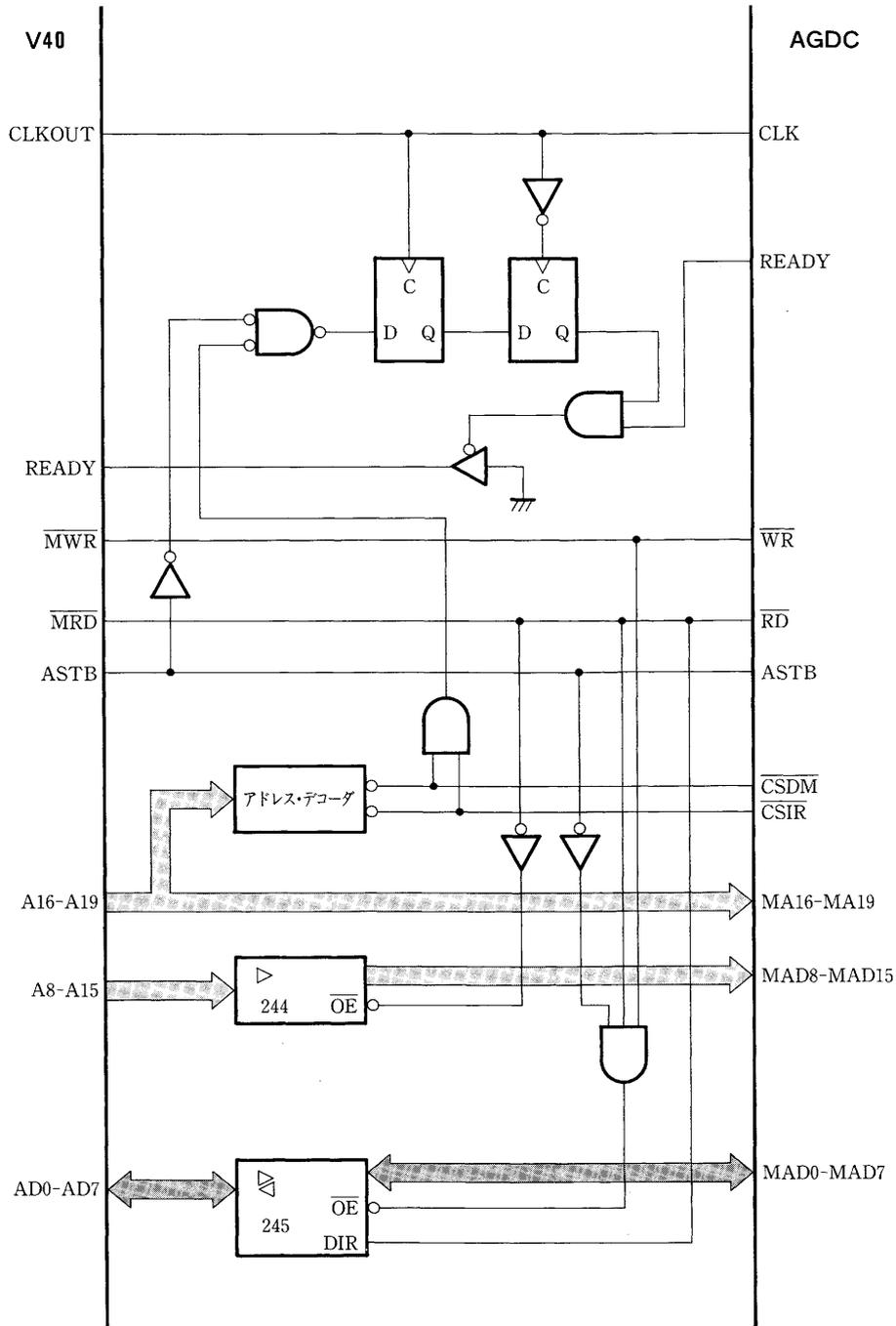


③ V33とAGDCの接続例

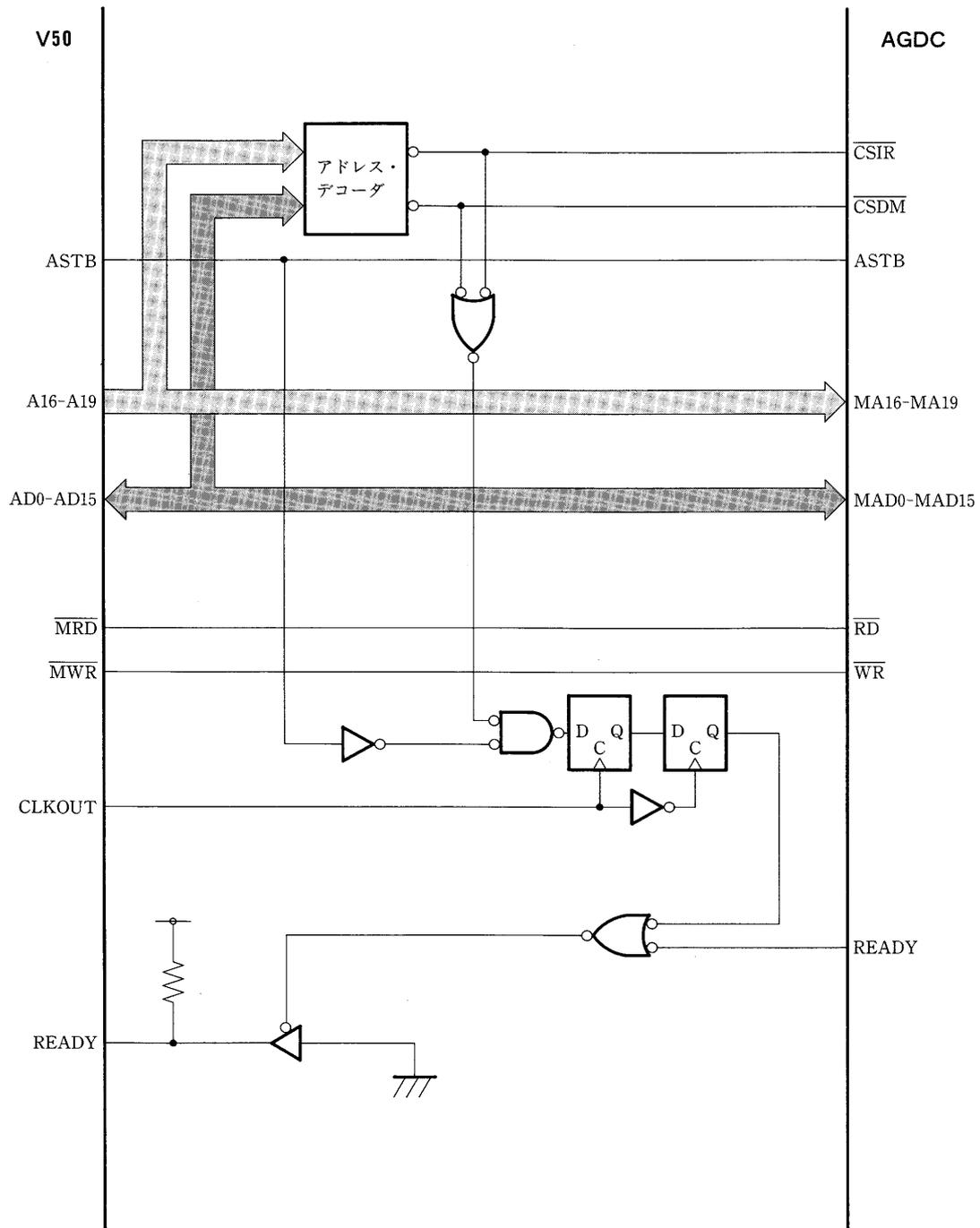


④ V40とAGDCの接続例

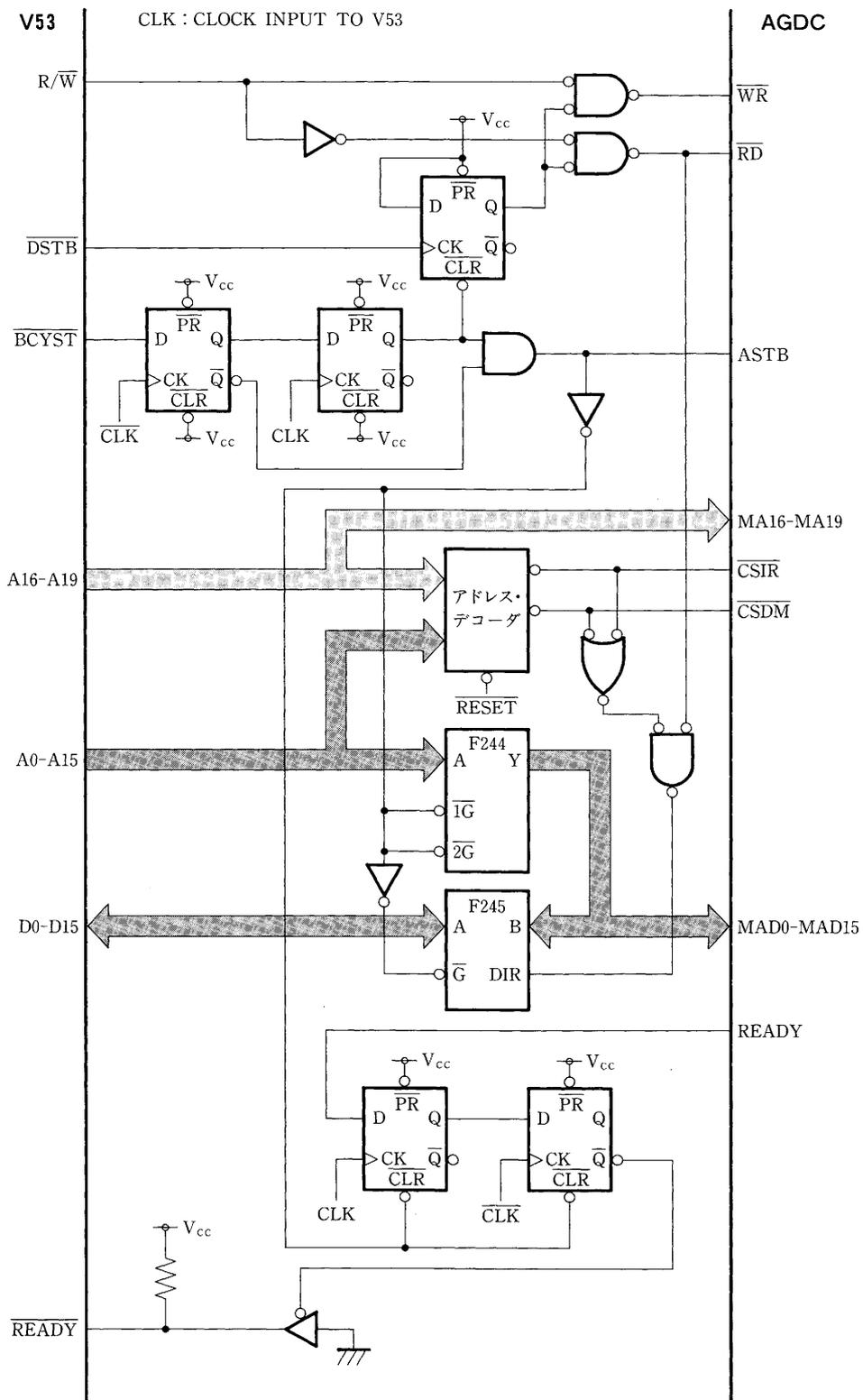
★



⑤ V50とAGDCの接続例



⑥ V53とAGDCの接続例



Q3.2

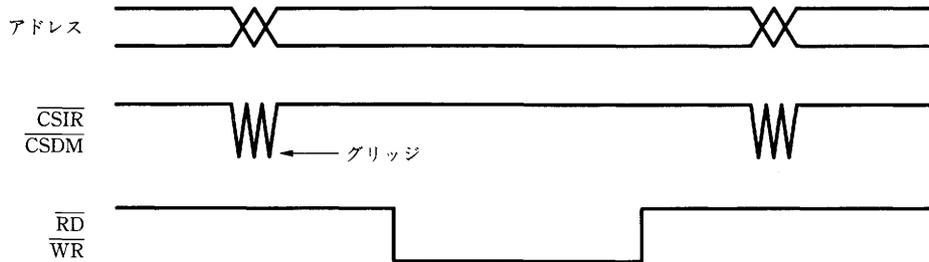
内容は？

機能・動作

特性

応用

$\overline{RD}$ ,  $\overline{WR}$ 入力がインアクティブのとき,  $\overline{CSIR}$ ,  $\overline{CSDM}$ 端子にグリッジを入力したらどうなる？



**A3.2**

特に問題はありません。

$\overline{RD}$ 信号および $\overline{WR}$ 信号ともにインアクティブ状態 (ハイ・レベル) であれば $\overline{CSIR}$ ,  $\overline{CSDM}$ 信号にグリッジが入力されても影響ありません。

Q3.3

内容は？

機能・動作

特性

応用

$\overline{CSIR}$ ,  $\overline{CSDM}$ と $\overline{ASTB}$ のタイミングに規定がないが,  $\overline{ASTB}$ ↓より $\overline{CSIR}$ ,  $\overline{CSDM}$ が遅れてアクティブ (ロウ・レベル) になってよいか？

**A3.3**

特に問題はありません。

$\overline{CSDM}$ ,  $\overline{CSIR}$ ,  $\overline{DMAAK}$ がすべてインアクティブ状態でも $\overline{ASTB}$ ↓があれば $\overline{AGDC}$ はその時点でアドレスをラッチします。

**Q3.4**

内容は？ 機能・動作 特性 応用

★

$\overline{RD}$ ,  $\overline{WR}$ 信号がアクティブでチップ・セレクト入力信号 ( $\overline{CSIR}$ ,  $\overline{CSDM}$ ,  $\overline{DMAAK}$ ) がインアクティブの場合、AGDCはどのような動作をするのか？

**A3.4**

動作はしません。すなわちAGDC以外のほかのモジュールに対する $\overline{RD}$ ,  $\overline{WR}$ 信号をAGDCに直接入力してもかまいません。ただし、この状態においてチップ・セレクト入力信号にノイズが乗らないように注意してください。AGDCが誤動作する可能性があります。

**Q3.5**

内容は？ 機能・動作 特性 応用

ホストCPUがAGDCをアクセスしないとき、ASTBを入力していい？

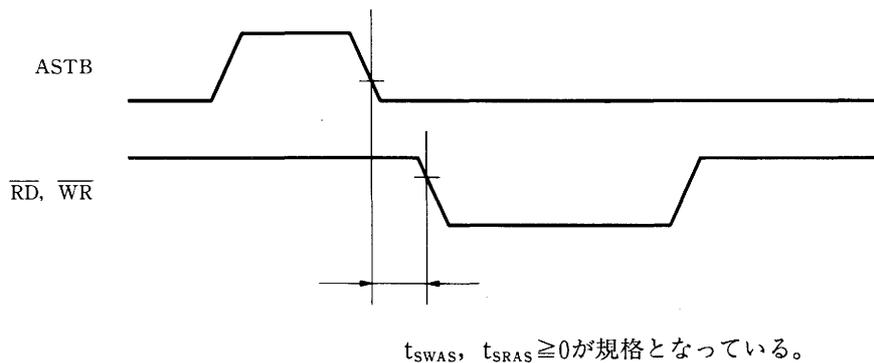
**A3.5**

$\overline{RD}$ ,  $\overline{WR}$ ともにインアクティブ状態であれば、ASTB↓が入力されてもAGDCに影響ありません。

**Q3.6**

内容は？ 機能・動作 特性 応用

システム・バスのリード/ライト・サイクルで、 $\overline{RD}$ ,  $\overline{WR}$ とASTBのタイミングはどうなる？



**A3.6**

リード・サイクルについては、 $t_{SRAS} \geq 0$ を満足する必要があります。ただしライト・サイクルについては、 $\overline{WR}$ 信号の立ち上がりで入力データのセット・アップ、ホールド時間が満足されていれば、 $t_{SWAS} < 0$ となっても、問題ありません。

## Q3.7

内容は？

機能・動作 特性 応用

システム・バス上でのDMA転送時、8/16ビットの切り替え方法は？

## A3.7

8/16ビット転送の識別信号としてMAD0,  $\overline{UBE}$ を用意していますので、下表に基づいて切り替えてください。

データ・バスが8ビットのCPUを接続する場合はプルアップしてください ( $\overline{UBE}$ )。

MAD0	$\overline{UBE}$	データ・アクセス形式
0	0	偶数アドレス・ワード
0	1	偶数アドレス・バイト
1	0	奇数アドレス・バイト
1	1	奇数アドレス・バイト

これらの信号をラッチするためには、DMA要求デバイス(たとえば8ビットCPU)よりASTBを供給してください。

逆に、ASTB信号を供給しなくても、DMA転送(ワード転送)を行う1つ前の処理において、ASTB↓にラッチしたアドレス・データが、MAD0=0,  $\overline{UBE}$ =0であったならば、AGDCは内部のテンポラリ・アドレス・データにその値を保持し続けます。

したがって、その状態でDMAを起動する際には、ASTB信号を供給する必要はありません。

また、MAD0,  $\overline{UBE}$ のレベルも“0”, “1”のどちらでもかまいません。

**Q3.8**

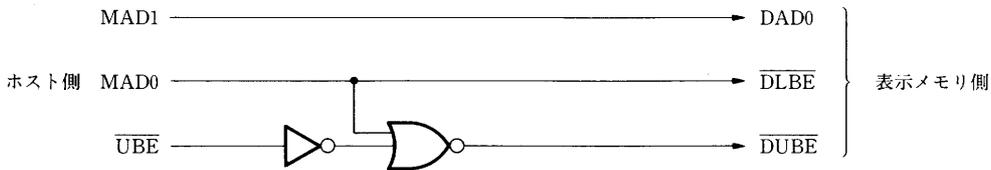
内容は? 機能・動作 特性 応用

表示メモリ・アクセスのための $\overline{DLBE}$ 出力は、表示メモリへのアドレス出力DAD0の値と同じ?

**A3.8**

同一ではありません。

$\overline{DLBE}$ は、下図の回路に従って生成されます。



したがって、ホスト側MAD1, 0が等しいときのみ、DAD0と $\overline{DLBE}$ は等しくなります。

また、AGDCがメモリ・サイクルを起動しない場合、 $\overline{DLBE}$ 、 $\overline{DUBE}$ はともにロウ・レベルとなっています。

**Q3.9**

内容は? 機能・動作 特性 応用

システム・バス側を8ビット・データ・バス ( $\overline{UBE}=H$ ) にして、DMA転送する場合、表示メモリ側のD0-D15とシステム・バス側のAD0-AD7の関係は?

**A3.9**

8ビットCPUのアクセス・アドレスによって、以下のように異なります。

- ・ 奇数アドレス： $\overline{DLBE}=1$ 、 $\overline{DUBE}=0$ となり、MAD0-MAD7はDAD8-DAD15に接続されます (DAD0-DAD7のレベルは不定です)。
- ・ 偶数アドレス： $\overline{DLBE}=0$ 、 $\overline{DUBE}=1$ となり、MAD0-MAD7はDAD0-DAD7に接続されます (DAD8-DAD15のレベルは不定です)。

## Q3.10

内容は？

■機能・動作 □特性 □応用

描画レジスタの内容を読む場合、READY信号は必ずインアクティブになる？

## A3.10

必ず、READYはインアクティブとなります。

したがって、このREADY信号を必ずホストCPUに渡し、ウェイトをかけてください。

## Q3.11

内容は？

■機能・動作 □特性 □応用

プリプロセッサ動作中にステータス・レジスタを読み出そうとしても、READYがインアクティブ状態のままになってしまい、ステータス・レジスタを読み出せないのでは？

## A3.11

ステータス・レジスタはプリプロセッサの外にあります。ステータス・レジスタ読み出しの際には、READY信号は、いったんインアクティブ状態（ロウ・レベル）になります。ただし、すぐにアクティブ状態に戻ります。

また、READY信号出力は、 $\overline{WR}$ または $\overline{RD}$ が入力された時点ではじめてインアクティブ状態になる（またはアクティブ状態のままの）信号です。 $\overline{WR}$ または $\overline{RD}$ が入力されるまでは、常にアクティブ状態（ハイ・レベル）になっています。

## Q3.12

内容は？

■機能・動作 □特性 □応用

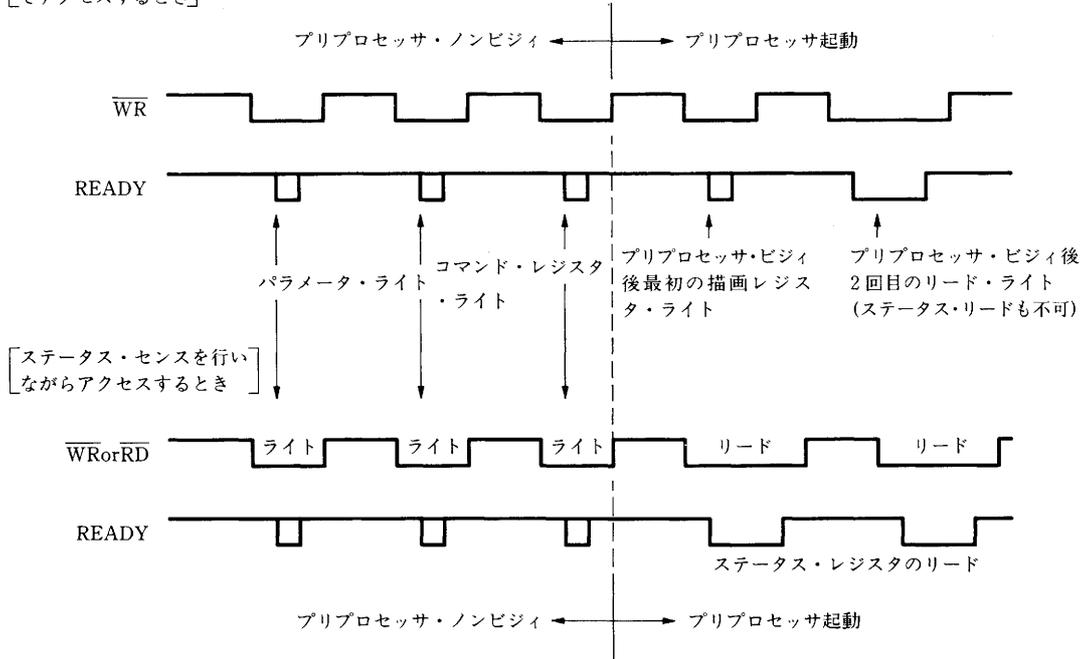
$\overline{WR}$ 信号とREADY信号との関係は？

## A3.12

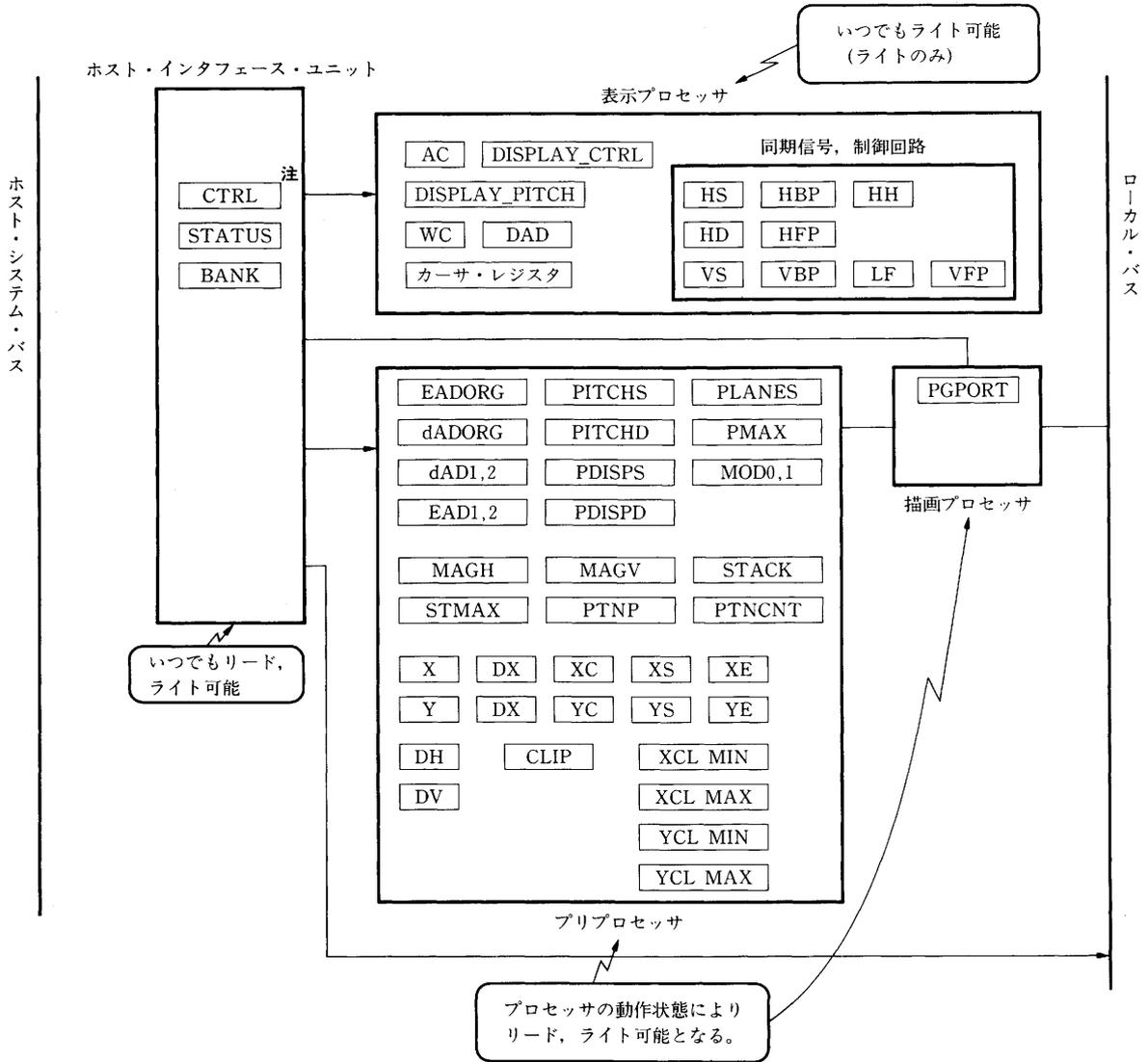
プリプロセッサは描画プロセッサに受け渡すデータを前処理します。そのとき、描画レジスタをアクセスすると、このREADYが出力されます。特に、このプリプロセッサがビジィ（コマンド・パラメータ書き込み直後）状態で次のパラメータを書き込むと、ビジィのまま外部のホストは待たされることになります。この期間はホストに接続されるメイン・メモリ（一般にはDRAM）のリフレッシュ期間より長く（>4 ms, 8 ms）になることもありますので、必ずステータス・ポーリングを行ってからライトしてください。次ページにその詳細を示します。

1.  $\overline{WR}$ , READYの関係について

[READYのみの使用]  
でアクセスするとき



2. AGDC内部ブロックと各レジスタの関係



注 CTRLおよびBANKレジスタはライトのみ、STATUSレジスタはリードのみが可能です。

## ポイント

前ページの各レジスタとプロセッサの関係を参考にしながら下表を見てください。

- 1) コマンド・レジスタ・ライト時のREADYは、すぐにアクティブになります。
- 2) プリプロセッサがビジイであっても、基本的に次にプリプロセッサ内のパラメータ（描画レジスタ）へのライトが実行されるまでの間のレジスタ・リードは下表に示すように可能です。

リード種類	プリプロセッサ・ビジイ中	描画プロセッサ・ビジイ中
プリプロセッサ・パラメータ	プリプロセッサを1CLKウエイト <sup>注1</sup>	影響なし
【STATUS】	影響なし	影響なし
DMリード <sup>注2</sup>	プリプロセッサを4CLKウエイト	描画プロセッサを4CLKウエイト
GET	プリプロセッサを2CLKウエイト	描画プロセッサを2CLKウエイト

注1. 使用上、この状態となるアクセスはしないと考えられます。

2. AGDCスルーの表示メモリ・リード

- 3) プリプロセッサ・ビジイであってもCTRLレジスタ、BANKレジスタへのライトは可能です（下表）。

ライト種類	プリプロセッサ・ビジイ中	描画プロセッサ・ビジイ中
表示・同期パラメータ	影響なし	影響なし
プリプロセッサ・パラメータ	アクセス不可状態になる	影響なし
【CTRL】【BANK】	影響なし	影響なし
DMライト <sup>注</sup>	プリプロセッサを3CLKウエイト	描画プロセッサを3CLKウエイト
PUT	プリプロセッサを2CLKウエイト	描画プロセッサを2CLKウエイト

注 AGDCスルーの表示メモリ・ライト

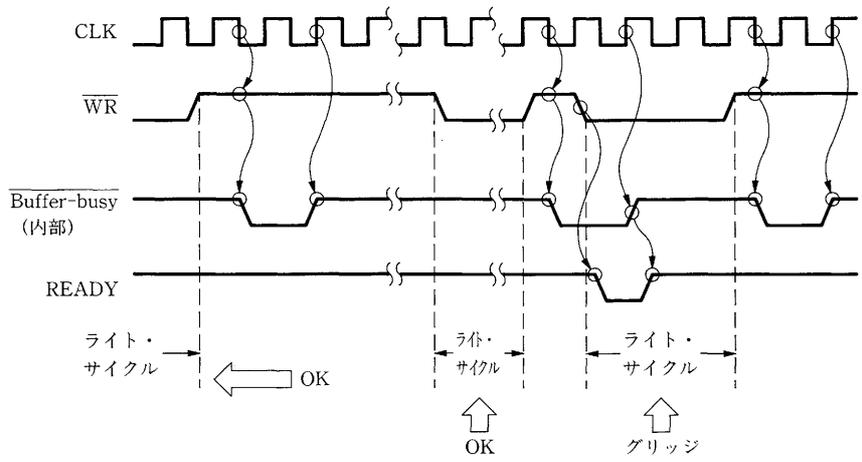
**Q3.13**

内容は？	<input type="checkbox"/> 機能・動作 <input checked="" type="checkbox"/> 特性 <input type="checkbox"/> 応用
------	---------------------------------------------------------------------------------------------------

プリプロセッサが、ノンビジィのときに描画パラメータを書き込むと、READY信号はどうなる？

**A3.13**

ライト・サイクルを連続する場合、READYにグリッジが出る可能性があります（ $\neg \nabla$  となりますが、 $\neg \wedge$  とはなりません）。



- ①  $\overline{WR}$ ↑をCLK↓でサンプリングします。
- ②  $\overline{WR}$ が  $\neg \wedge$  となったことを検出したCLK↓から数えて、1.5CLK分だけAGDC内部の $\overline{Buffer-busy}$ 信号がアクティブ状態になります。
- ③ READY信号は $\overline{WR}$ ↓によりインアクティブ状態となり、 $\overline{Buffer-busy}$ 信号↑によりアクティブ状態となります。
- ④ したがって、 $\overline{Buffer-busy}$ がアクティブ状態の間に次のライト・サイクルを起動すると、READY信号が  $\neg \nabla$  となる可能性があります。

**Q3.14**

内容は？	<input type="checkbox"/> 機能・動作 <input type="checkbox"/> 特性 <input checked="" type="checkbox"/> 応用
------	---------------------------------------------------------------------------------------------------

PAINTコマンドを実行するとREADYが長い間ロウ・レベルになる？

**A3.14**

PPBSYフラグ=0をステータス・ポーリングすれば、READYが長時間ロウ・レベルになることはありません。ただし、PAINT, PUT, GETのコマンドはPPBSY=DPBSY=0をステータス・ポーリングしてください。

**Q3.15**

内容は？ 機能・動作 特性 応用

$t_{\text{DRRY}}, t_{\text{DWRy}}$ はMAX.30 ns( $\mu$ PD72120(A)では40 ns)となっているが、これは、そのサイクルでAGDCがビジー状態であれば、リード/ライト後必ず30 ns以内にREADY信号はロウ・レベルに移行するということ？

**A3.15**

移行します。リード時は、30 ns以内で必ずロウ・レベルになります。ただし、ライト時は、30 ns以内でロウ・レベルにならなければ、そのサイクル中ロウ・レベルにはなりません。

注意1.  $\overline{\text{RD}}\downarrow$ , または $\overline{\text{WR}}\downarrow$ ののち、30 ns経過してREADY信号を参照してください。そのとき、ハイ・レベルであれば、そのサイクルを終了してかまいません。

2. READY信号の立ち上がりエッジで動作するロジックにしないでください。

**Q3.16**

内容は？ 機能・動作 特性 応用

ホストDMA時のREADY出力と、システム・バス上の優先順位の関係は？

**A3.16**

DTモード（デュアルポート・グラフィクス・バッファ制御モード）を使用するものとして説明します。ホスト・ダイレクト・メモリ・アクセスは、 $\overline{\text{CSDM}}$ , と $\overline{\text{RD}}$ ,  $\overline{\text{WR}}$ いずれかをアクティブすることで、ホストCPUが表示メモリをダイレクトにアクセスできるモードです。

このときのバスの優先順位は、

- ① 表示プロセッサが $\overline{\text{DT}}$ サイクルまたは、リフレッシュ・サイクルを起動
- ② 外部よりのHLDRQ入力
- ③  $\overline{\text{CSDM}}$ によるアクセス
- ④ 描画サイクル

} となります。

したがって、 $\overline{\text{CSDM}}$ によるホストCPUからのアクセスが行われても、①, ②による高優先の要求が発生した場合には、①, ②による処理を実行後、③が実行されます。そのため、ホストCPUは、READYをインアクティブとしたまま、停止していることとなります。

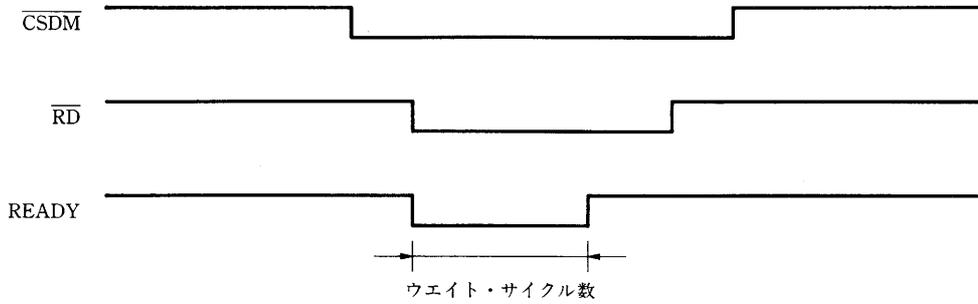
このときの、READY信号がロウ・レベルとなり、ホストCPUが待ち状態となる期間を以下に示します。

サイクル	MIN.	MAX.
ライト	1.5CLK	5.5CLK+A <sup>注</sup>
リード	4.5CLK	2.5CLK+A <sup>注</sup>

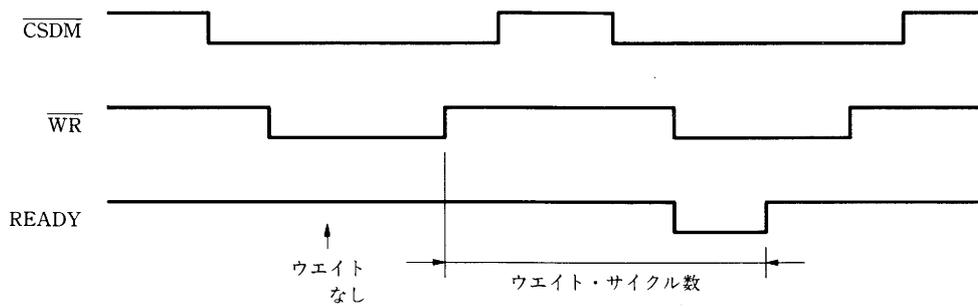
注 表示サイクル、リフレッシュ・サイクルを競合した場合、その分ウエイトとなるサイクル数

また、ウエイト期間開始のタイミングは、以下のとおりです。

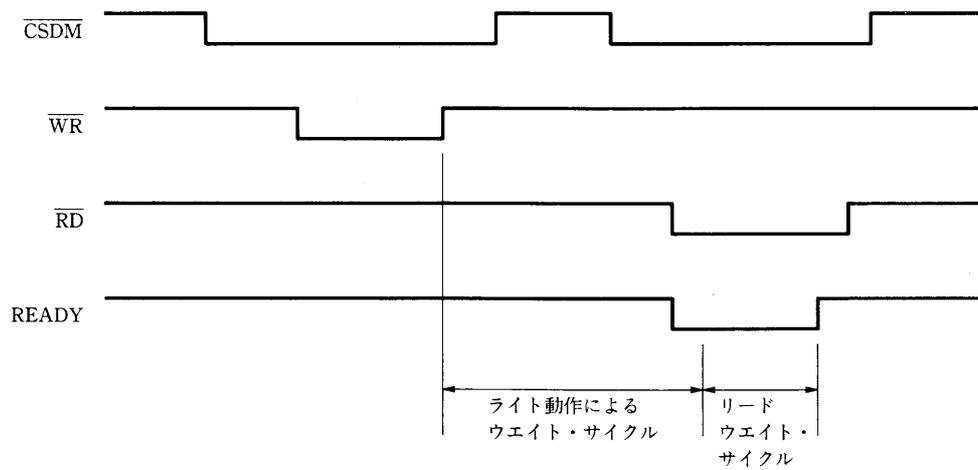
(リード・サイクル)



(ライト・サイクル1)



(ライト・サイクル2)



## Q3.17

内容は？

機能・動作
 特性
 応用

描画プロセッサが動作中にCSDMによりメモリ・アクセスができる？

## A3.17

できます。表示メモリ・バス・アクセスの優先順位からCSDMによるアクセスは、描画プロセッサによる描画サイクルより優先されます（ユーザーズ・マニュアル参照）。したがって、上記の場合はその期間中、描画プロセッサは待たされCSDMによるアクセスが実行されます。

## Q3.18

内容は？

機能・動作
 特性
 応用

AGDCが描画中にDISPLAY\_CTRLレジスタの内容を書き換えてもいい？

## A3.18

AGDCが描画の最中に、DISPLAY\_CTRLレジスタの内容を書き換えても描画には影響はありません。AGDCがグラフィクス・バス側において描画サイクル（書き込み、または読み出し）を起動している場合でも、CPUよりAGDCのDISPLAY\_CTRLレジスタの内容を書き換えても（ただし、プリプロセッサがノンビジィのとき）、VRAM上に描画されるデータが壊れてしまうことはありません。

## Q3.19

内容は？

機能・動作
 特性
 応用

AGDCが複雑な描画などを行っている場合、ホストCPUからの表示メモリ・アクセス（CSDMによる）に対するREADY=Lの期間は？

## A3.19

READY=Lの期間は、最大でもリフレッシュ期間（HS期間）程度です。

つまり、10  $\mu$ s程度です。

描画サイクル実行中の場合は描画を中断し、ホスト・ダイレクト・アクセスを実行します。

**Q3.20**

内容は? 機能・動作 特性 応用

AGDCに描画コマンドを発行しても描画動作が行われない場合、原因は?

**A3.20**

ソフトウェア、ハードウェアそれぞれの面から次のような原因が考えられます。

- ソフト：・描画コマンドの発行が行われていない。  
 ・描画されたメモリが実装空間ではない。  
 ・描画された領域が表示領域内でない。

ハード：・HLDRQ端子のアクティブが続き、描画プロセッサがバスを獲得できないので描画が行われない。

**Q3.21**

内容は? 機能・動作 特性 応用

ホストCPUがAGDCをアクセスしても、AGDCのREADY信号がアクティブにならない場合、原因は?

**A3.21**

原因として、ハード、ソフトの両面が考えられます。

次ページ以降に原因と対策を示します。

1) ハード的な要因 (1/2)

チェック内容	対策
1. $\overline{CSIR}$ , $\overline{CSDM}$ , $\overline{DMAAK}$ を同時に2つ以上アクティブ状態にしていないか? (入力ハイ・インピーダンスも入力アクティブ状態となりえます)	1. $\overline{CSIR}$ , $\overline{CSDM}$ , $\overline{DMAAK}$ を同時に2つ以上アクティブ状態にしないでください。
2. HLDRQが常にハイ・レベル(アクティブ状態)になっていないか? (HLDRQにハイ・レベルが供給されているとREADYはインアクティブ状態が続く場合があります)	2. 通常HLDRQはロウ・レベルにしてください。

1) ハード的な要因 (2/2)

チェック内容	対策
<p>3. READYがアクティブ状態になったりならなかったりし、その現象に電圧および温度依存性がないか？ この場合は、ASTBに対するアドレスのセット・アップ／ホールドに関するスペックが満足されているかどうかチェックしてください（たとえばアドレスのセット・アップ／ホールドが満足されず、誤ってPGPORTが選択されるとアクセスを行うため、READYはインアクティブ状態になります）。</p>	<p>3. アドレスのセット・アップ／ホールドを満足するようにしてください。</p>
<p>4. <math>\overline{\text{DMAAK}}</math>信号にノイズが乗って、PGPORTレジスタにアクセスした状態になっていないか？</p>	<p>4. オシロスコープなどで信号線を確認してください。</p>
<p>5. <math>\overline{\text{CSDM}}</math>信号にノイズが乗り、ホスト・ダイレクト・アクセスを行ったような状態になっていないか？</p>	<p>5. オシロスコープなどで信号線を確認してください。 ★</p>
<p>6. システム・バス・ライト・サイクルは動作するが、システム・バス・リード・サイクルの動作がおかしい。 この場合は、ASTBと<math>\overline{\text{RD}}</math>のタイミングがスペックを守っているかどうかをチェックしてください（<math>\overline{\text{RD}}\downarrow</math>がASTB<math>\downarrow</math>より先に行われた場合、<math>\overline{\text{RD}}\downarrow</math>によりAD15-0は出力になり、アドレスをラッチできなくなります。このためREADYがインアクティブ状態になる場合があります）。</p>	<p>6. ASTB<math>\downarrow</math>と<math>\overline{\text{RD}}\downarrow</math>の関係を図3-1に示します。</p> <p style="text-align: center;">図3-1 リードの立ち下がりタイミング</p> <p>(a) ASTBの立ち下がり以前に行われた場合 (スペックを守っていない)</p> <div style="text-align: center;"> </div> <p>(b) ASTBの立ち下がり以降に行われた場合 (タイミング・スペック<math>t_{\text{SRC}}</math>を守る)</p> <div style="text-align: center;"> </div>

2) ソフト的な要因

チェック内容	対策
1. プリプロセッサ・ビジィ (PPBSY=1) 中, 描画レジスタ (PGPORTを除く) にライト動作を行っているか?	1. ステータス・レジスタ (PPBSY=0) を確認してからライト動作を行ってください。
2. PUT/GETコマンド実行時でPGPORTをバイト・アクセスしている場合, 3E, 3F番地を連続アクセスしているか?	2. 3E, 3F番地の間に他の番地アクセスを行わないでください。  <div style="display: flex; flex-direction: column; align-items: center;"> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <div style="text-align: center; margin-right: 10px;">3EH バイト・リード/ライト</div> <div style="border-bottom: 1px solid black; width: 100px;"></div> <div style="margin-left: 10px;">← 他の番地アクセス不可</div> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <div style="text-align: center; margin-right: 10px;">3FH バイト・リード/ライト</div> <div style="border-bottom: 1px solid black; width: 100px;"></div> <div style="margin-left: 10px;">← 他の番地アクセス可能</div> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <div style="text-align: center; margin-right: 10px;">3EH バイト・リード/ライト</div> <div style="border-bottom: 1px solid black; width: 100px;"></div> <div style="margin-left: 10px;">← 他の番地アクセス不可</div> </div> <div style="display: flex; align-items: center;"> <div style="text-align: center; margin-right: 10px;">3FH バイト・リード/ライト</div> <div style="border-bottom: 1px solid black; width: 100px;"></div> </div> </div>

**Q3.22**

内容は?    機能・動作    特性    応用

V50のDMAで表示メモリの内容をメイン・メモリに転送する場合, AGDCとV50の各READY端子のインタフェースは?

**A3.22**

AGDCはDMAサイクルで,  $\overline{RD}$ ↓または,  $\overline{WR}$ ↓に同期してREADYをホストCPU (V50) に返します。V50はT2サイクルのCLKOUTの立ち上がりでREADY信号を受けます。ただし, AGDCのREADY信号をそのまま用いると遅れてしまいます。

したがって, AGDCが $\overline{RD}$ ↓,  $\overline{WR}$ ↓を送出後,  $t_{DRRY}$ ,  $t_{DWRV}$  (30 ns)の間, 疑似的にNOT-READYの信号を作ってください。

## Q3.23

内容は？

機能・動作 特性 応用

INT信号がアクティブ、インアクティブになるタイミングは？

## A3.23

INT信号出力は、以下の場合にCLKの立ち上がりエッジに同期し50 ns以内にアクティブとなります。その後、ステータス・リード時の $\overline{RD}$ 信号をアクティブとしたあと、 $t_{DRI}$ 以内 (MIN.  $2 \cdot t_{CYK}$ , MAX.  $3 \cdot t_{CYK} + 50$  ns) にインアクティブとなります。

INT信号がアクティブになる条件

- ①コントロール・レジスタ内のCIE, PBIE, DBIEフラグ=1のときにフラグに相当する要因が発生すると、INT信号がアクティブになる。
- ②プリプロセッサ、描画プロセッサが処理実行中にエラー状態を検出すると、INT信号がアクティブになる。

## Q3.24

内容は？

機能・動作 特性 応用

AGDCによるINT割り込みの発生と同時にステータス・リードを行うと、INT信号はクリアされてしまうのか？

## A3.24

クリアされることはありません。INT出力端子は最小1CLK幅で出力します。

★

## Q3.25

内容は？

■機能・動作 □特性 □応用

AGDCのステータス・レジスタについて

- (1) ステータス・レジスタのビジィ・フラグは、同時にアクティブになることがある？
- (2) ステータス・レジスタでエラーを検出したらどういふ処理をすればいい？

## A3.25

- (1) AGDCは、プリプロセッサ処理の途中でも、描画プロセッサに渡すパラメータが準備できると描画プロセッサにそのパラメータを引き渡します。したがって、両方のプロセッサがアクティブになることがあります (PPBSY=DPBSY=1の状態になる)。

PPBSY=DPBSY=1のアクティブ状態のままになる条件として考えられることは、クリッピング領域を設定しないでPAINTコマンドを発行し、指定した内部座標点が塗りつぶす図形の外部の点を指定したなどが考えられます。

- (2) ① エラー割り込みは、処理する必要はありません。
- ② ステータス・レジスタを読み出してください。読み出すことでエラーを特定することができます。また、エラー・フラグ (PPER, DPERR) が0クリアされます。
- ③ エラーを起こしたあとは、プリプロセッサ、描画プロセッサともコマンドを受け付ける状態になっています。したがってRESET, ABORTなどする必要はありません。

## Q3.26

内容は？

□機能・動作 □特性 ■応用

CTRLレジスタ内のPBIEをイネーブルするタイミングと、プリプロセッサがビジィからノンビジィへと変化するタイミングがあった場合の、INT信号との関係について

- (1) ノンビジィとなるタイミングが早い場合、INT信号はONになる？
- (2) ノンビジィとなるタイミングが遅い場合、INT信号はONになる？

## A3.26

- (1) ONになりません。
- (2) ONになります。

## Q3.27

内容は？

機能・動作特性応用

AGDCのPBIE, DBIEの基本的な使い方は？

(これらの制御ビットをどのような使い方を前提として設けたか？)

## A3.27

基本的な動作は次のようになります。(H=ハイ・レベル, L=ロウ・レベル)

AGDC内部ではINT出力はエッジ<sup>注</sup>・トリガになっています。

- ① INT出力がHのとき：PBIEにいかなる値を書き込んでも、INT出力はHのままです。
- ② INT出力がLのとき：プリプロセッサ・ビジィ中にPBIEに0を書き込むと、INT出力がHになります。プリプロセッサ・ノンビジィ中にPBIEに0を書き込んだ場合はLのままです。

注 ビジィ→ノンビジィというエッジ

## Q3.28

内容は？

機能・動作 特性 応用

V40(8ビット・データ・バス)で16×16ビット構成のPGPORTにDMAを行う場合、外付け回路は不要？

## A3.28

外付け回路なしでの転送は可能です。

AGDCのシステム・バスは、次のようにアクセスされます。

MAD0	$\overline{UBE}$	データ・アクセス形式
0	0	偶数アドレス・ワード
0	1	偶数アドレス・バイト
1	0	奇数アドレス・バイト
1	1	奇数アドレス・バイト

したがって、8ビット・システム・バスを持つV40では、 $\overline{UBE}$ をHとしてMAD0に最下位アドレスを供給すればよいことになります。

このときのセットアップ/ホールド時間は、システム・バス・リード/ライト時と同一です。

また、転送バイト数は偶数としてください。

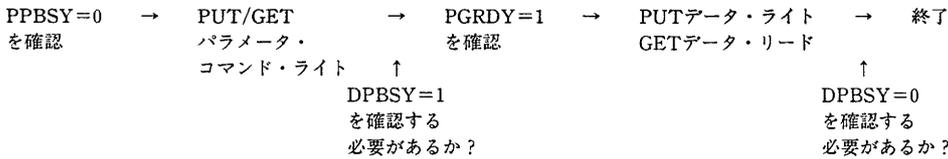
$$\text{転送バイト数} = \left\{ \frac{DH+1}{8} + \alpha \right\} \times (DV+1)$$

$$\text{このとき} \left\{ \begin{array}{ll} \frac{DH+1}{8} = \text{奇数} & \text{なら } \alpha = 1 \\ \frac{DH+1}{8} = \text{偶数} & \text{なら } \alpha = 0 \end{array} \right\} \text{となります。}$$

**Q3.29**

内容は? 機能・動作 特性 応用

PUT/GETコマンド実行時のフラグ・チェックは、ステータス・レジスタのPPBSYとPGRDYだけでいい?



**A3.29**

PUT/GETコマンドを発行して以下のような処理を行う場合、コマンド発行終了後、DPBSY=0の確認をしてください。

PUTコマンドを連続で発行する場合を以下に示します。

今、1回目のPUTコマンドを発行し、PUTデータの最終ワードをPGPORTに書き込んだとします。ところが、表示メモリ・バス側でHLDRQを受け付けていると、PGPORTに書き込まれたデータは表示メモリへ書き込みできません。

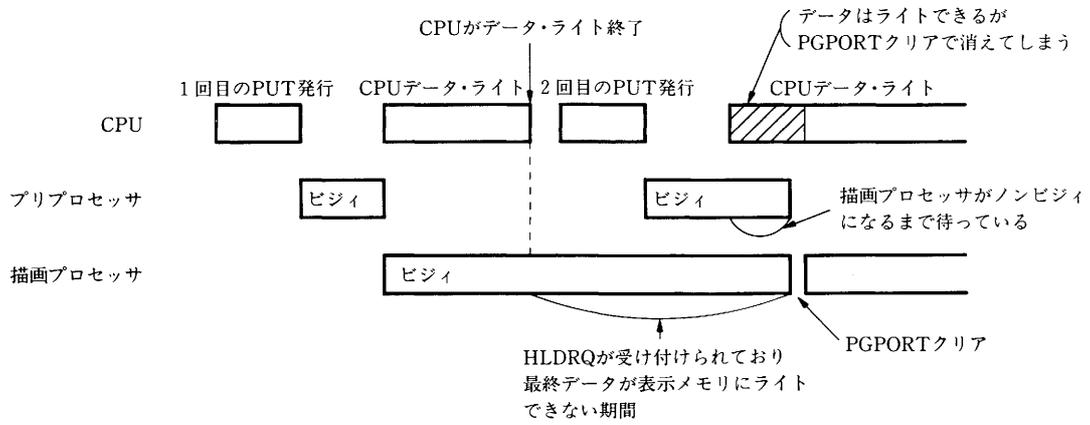
このときに2回目のPUTコマンドを発行しても、描画プロセッサが1回目のPUTコマンドを実行中(ビジー)のため、プリプロセッサでは2回目のPUTコマンド動作が停止した状態になります。2回目のPUTコマンド動作が停止した状態であっても、PGPORTへは書き込めます。

ここで、PGPORTへデータ(2回目のPUTコマンドの最初のデータ)を書き込んだとします。このとき、偶然HLDRQの受け付けが終了し、1回目のPUTデータが表示メモリに書き込まれたとすると、描画プロセッサは1回目のPUTコマンドの動作を終了し、2回目のPUTコマンドを実行しようとしています。

このとき、PGPORTのデータがクリア(フラッシュ・アウト)され、2回目のPUTコマンドの最初のデータが消えてしまうことがあります。

このようなことを防ぐため、2回目のPUTコマンドを発行する前に、前回のPUTコマンドの描画プロセッサの動作が、終了したかどうかをDPBSYで確認してください。上記のシーケンスを図3-2に示します。

図3-2 連続でPUTコマンドを発行した場合のシーケンス

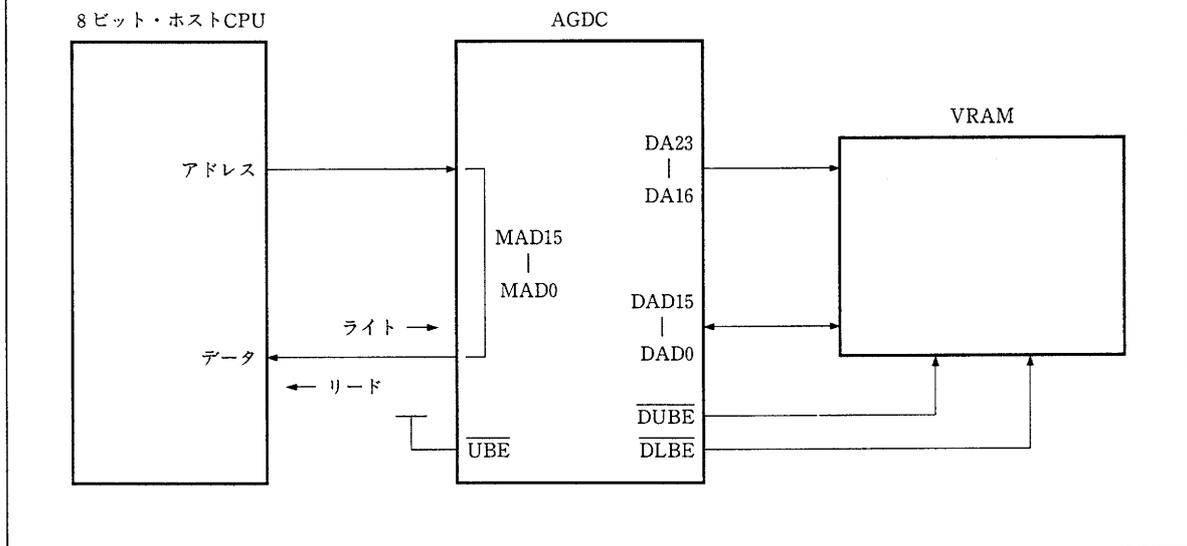


## Q3.30

内容は？

機能・動作特性応用

8ビット・ホストCPUをAGDCに接続する場合の注意点は？



## A3.30

8ビット・ホストCPUとAGDC（16ビットの入出力アドレス/データ・バス）との接続上の注意点は、以下のとおりです。

## 1) ライト・サイクル時（CPU→AGDC）

8ビット・ホストCPUとの接続では、UBEをV<sub>DD</sub>にプルアップすること以外、特にハードウェア上の注意点はありませぬ。

## 2) リード・サイクル時（AGDC→CPU）

AGDCのデータ出力MAD8-MAD15ビットは、UBEをハイ・レベルにしても、16ビット・ホストCPUとの接続が前提で設計されているため、ハイ・インピーダンスとなりませぬ（何らかの確定データが出力します）。このため、8ビット・ホストCPUの上位アドレスとAGDCのMAD8-MAD15ビットが同一のバス上で衝突します。この状態を回避するには、バッファによる入出力制御が必要です。

## Q3.31

内容は？

■機能・動作 □特性 □応用

モトローラ系の16ビットCPUとの接続のとき、

- (1) データ・バス接続方法は？
- (2) バイト・アクセスもできる？

## A3.31

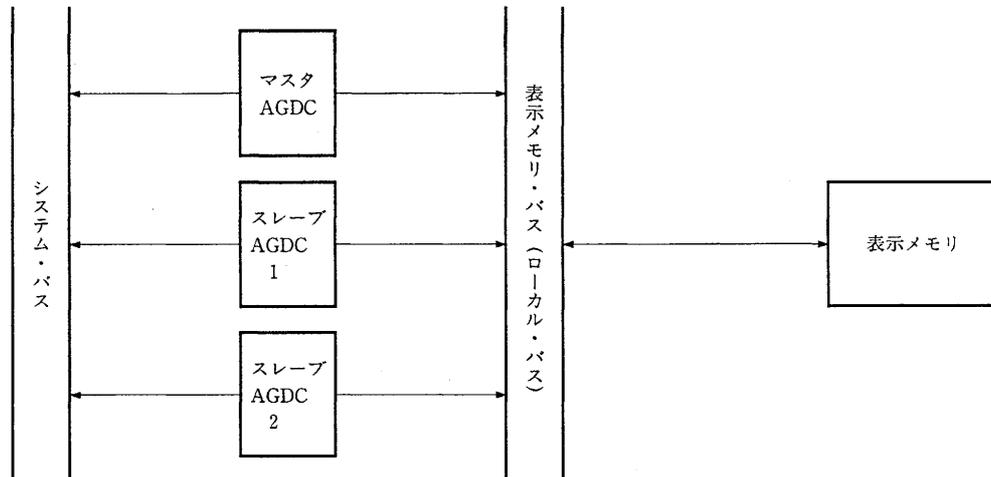
- (1) AGDCのアドレス割り付けは、モトローラ系の16ビットCPU（たとえば、mc68000など）とは、バイト単位で逆になっています。したがって、ワード単位でアクセスする場合には、外部システム・バスをバイト単位で逆にして接続してください(mc68000のD0-D7をAGDCのMAD8-MAD15と接続し、D8-D15をMAD0-MAD7と接続します)。これにより、ワード単位でのアクセスが可能になります(「Q3.1の接続例」を参照)。
- (2) バイトのみのアクセスは可能です。  
ただし、ステータス・レジスタ（3DH, 3CH番地）を読み出す際、3CH番地のデータを読み出すとそのときに、3DH番地のデータも同時にクリアされてしまいますので注意してください。また、(1)のようにすればワード単位でのアクセスも可能になります。

## Q3.32

内容は？

機能・動作特性応用

マルチAGDCシステム（マスタAGDCでスレーブAGDCを制御）でのバスの制御方法は？



## A3.32

以下に、4ビット/ピクセル構成のマルチAGDCシステム例を示します。

## マルチAGDCシステム構成上のポイント

## 1. 構成上のポイント

- 1) 各メモリ・プレーンへは同時描画を行う。
- 2) 第1プレーン描画用のAGDCをマスタとし、第2, 3…プレーン描画用のAGDCをスレーブとする。  
このとき、表示タイミング生成、データ転送<sup>注</sup>、リフレッシュ動作等は、マスタAGDCが制御する。したがって、スレーブAGDCは、純粋な“描画プロセッサ”として機能する。
- 3) マスタAGDCをすべてのメモリ・プレーンに対して、アクティブ（バスを保持し続ける）とし、プレーン同時描画をさせるときのみ、スレーブがメモリ・プレーンをアクセスできるように、動作モード選択信号を制御する。

このとき、スレーブAGDCは2)のようにリフレッシュ非実行(REFフラグ=0)、データ転送非実行(SDフラグ=1)となる動作に設定されているため、ホストAGDCよりのHLDRQ入力がスレーブAGDCの最高優先の表示メモリ・バス・サイクルとなる。

**注** デュアルポート・グラフィクス・バッファがシリアル出力を行うためのシフト・レジスタへ、データをロードする動作です。

## 2. システム構成上のAGDCに関する制限

次のコマンドは同時描画には使用できません。

- ・ PAINT
- ・ SD\_SELフラグが00, 01, 10のときのCOPYとPUT/GET

## 3. システム構成例の説明

## 1) ハードウェア上の制御

マスタAGDCがバスを保持するのは、以下の期間となります。

- 動作モード選択信号がハイ・レベルとなる期間
- HSがアクティブであるDRAMのリフレッシュ期間
- $\overline{DT}$ がアクティブであるデュアルポート・グラフィクス・バッファのデータ転送サイクル期間

## 2) ソフトウェア上の制御 (コマンド, パラメータの発行方法)

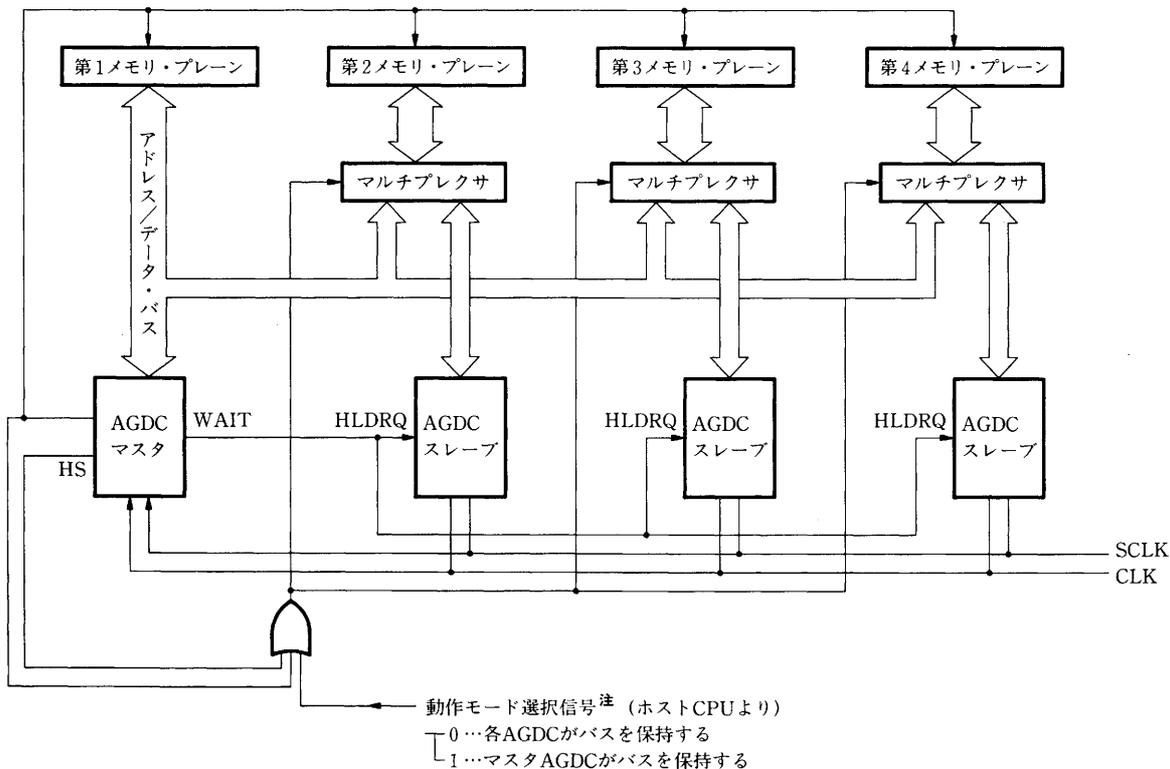
①マスタAGDCがすべてのメモリ・プレーンに対してアクティブであるときには、次の2つのコマンド・パラメータの発行方法があります。

- プリプロセッサ・ビジィ (PPBSY) フラグの状態をポーリングし、ビジィではなければ、コマンド・パラメータを発行する。
- プリプロセッサ・インタラプト・イネーブル (PBIE) フラグを1にセットしプリプロセッサがレディ状態となった時点で、出力されるINT信号をホストCPUに出力する。

②スレーブAGDCがバスを必要とするときのコマンド, パラメータの発行も①と同様に2通りあります。

- PPBSYおよびドローイング・プロセッサ・ビジィ (DPBSY) フラグがともに0であることをポーリングし、コマンド, パラメータを発行する。
- PBIE, ドローイング・プロセッサ・インタラプト・イネーブル (DPIE) フラグをともに1とし、プリプロセッサ, ドローイング・プロセッサがレディ状態となった時点で出力されるINT信号をホストCPUに出力する。

(マルチAGDCシステム構成例)



注 ホストCPUのアドレス空間上にマッピングします。

**Q3.33**

内容は? 機能・動作 特性 応用

コマンド終了をDPBSYのハードウェアINTで検出できる?

**A3.33**

できません。

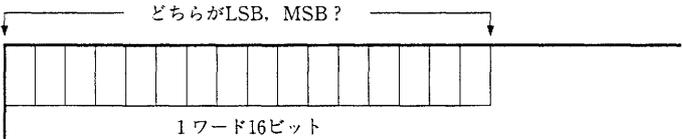
同一コマンドにて、DPBSYの割り込み信号が複数回発生することがありますが、このDPBSYの割り込み信号の幅の違いによって、最終割り込みをハードで識別することは不可能です。割り込み処理ルーチン内で「PPBSY=DPBSY=0」という条件判定をしてください。

第4章 メモリ制御とバス・アービトレーション

Q4.1

内容は？ 機能・動作 特性 応用

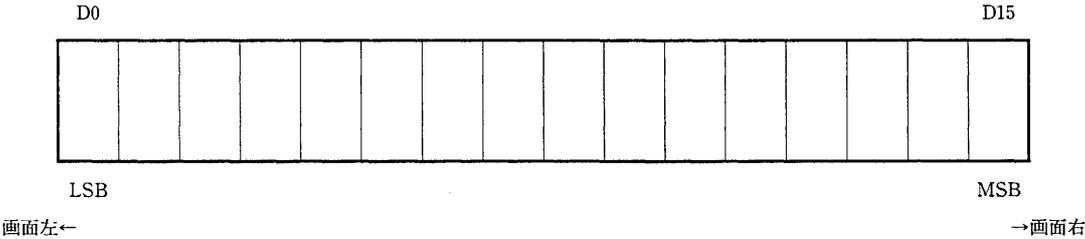
表示画面上で、LSB<sup>注</sup>とMSB<sup>注</sup>はそれぞれ左右どちら？



注 LSB…Least Significant Bit  
MSB…Most Significant Bit

**A4.1**

下図のとおりです。



## Q4.2

内容は？

機能・動作 特性 応用

表示メモリ用のリフレッシュ機能で、システム・バス上のメモリも同時にリフレッシュできる？

## A4.2

基本的には、システム・バス上のメモリのリフレッシュをサポートすることは可能です。

注意 AGDCがリフレッシュ実行に入るタイミング（リフレッシュ・サイクルが起動される4SCLKサイクル前にWAIT信号がアクティブ状態となる）では、必ず、AGDCにバスの制御権を渡してください。たとえば、AGDCをDTモード（デュアルポート・グラフィクス・バッファ制御モード）で使う場合、BLANK信号が0→1となったあと、最初に起動されるWAIT信号出力が0→1となるタイミングを捕まえて、ホスト・プロセッサへ、BUSRQ, HLDRQ, HALT等の信号を、供給してください。

## Q4.3

内容は？

機能・動作 特性 応用

表示メモリをアクセスしていないとき、 $\overline{DLBE}$ ,  $\overline{DUBE}$ 信号はどうなる？

## A4.3

ホスト・ダイレクト・アクセスによりバイト・アクセスを行ったときのみ、 $\overline{DUBE}$ ,  $\overline{DLBE}$ はハイ・レベルに変化します。それ以外では、 $\overline{DUBE}$ ,  $\overline{DLBE}$ はともに“0”となっています。したがって、ユーザーズ・マニュアルにも、説明しているように、ともに“1”の状態は存在しません。

## Q4.4

内容は？

■機能・動作 □特性 ■応用

$\mu$ PD24C2000-X11/-X12との接続について

- (1) 接続例は？
- (2) 複数の書体を持たせたいが、どうすればいい？

## A4.4

- (1) ROMアドレスの制御を図4-1、また、 $\mu$ PD24C2000-X11/-X12を用いた接続例を図4-2に示します。
- (2) カスタムするのであれば別ですが、通常は、明朝体のROM、ゴシック体のROMなど、フォントごとにROMを載せます。

その際には、同じアドレスを2つのROMに対して割り付けておき、チップ・イネーブル信号でどちらのROMをアクセスするかを決めてください。その方が、デバッグが簡単になります。

4種類あっても同様で、それぞれのROMには同じアドレスを割り付けて、チップ・イネーブル信号で切り替えてください。

図4-1 ROMアドレスの制御

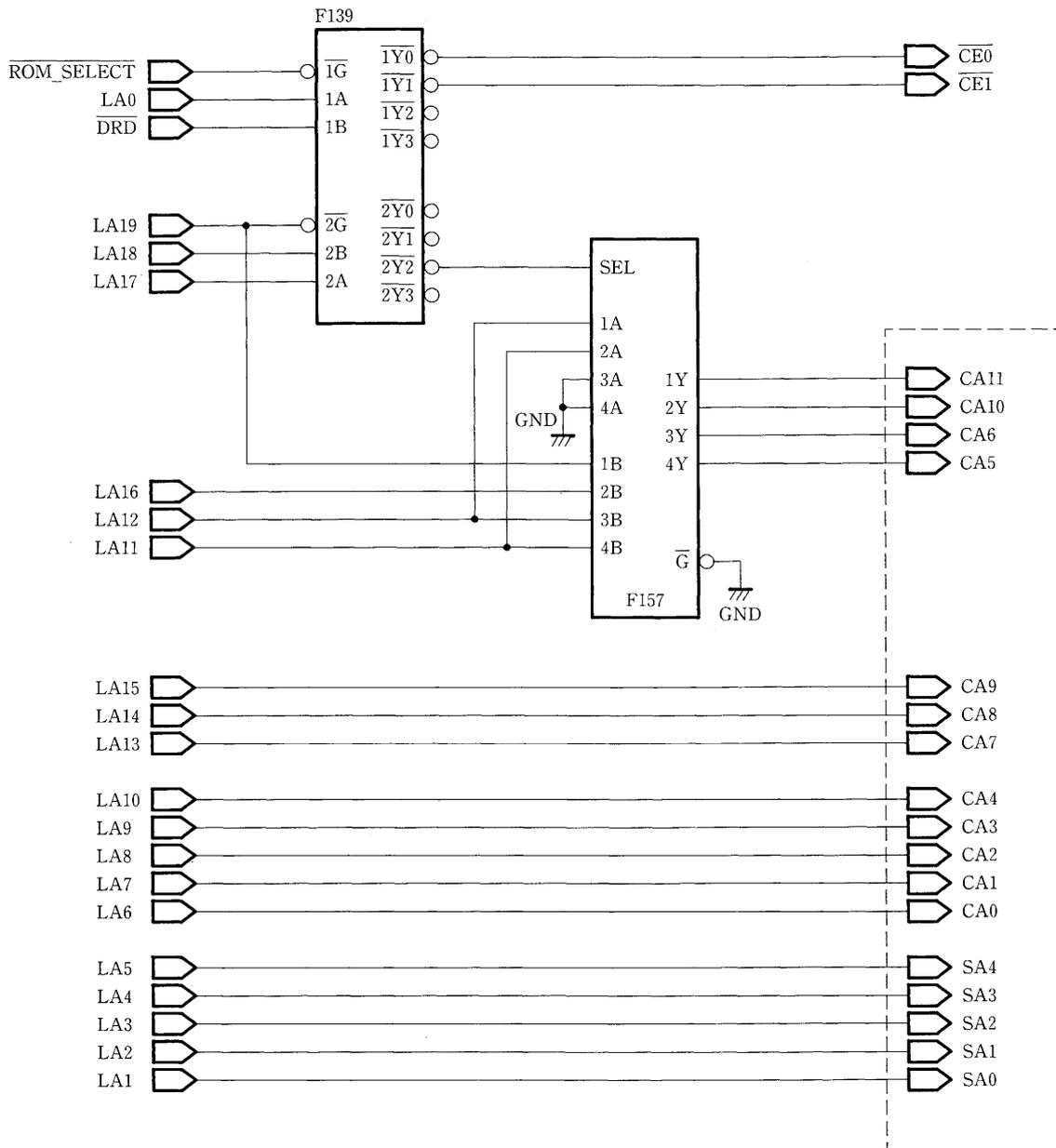
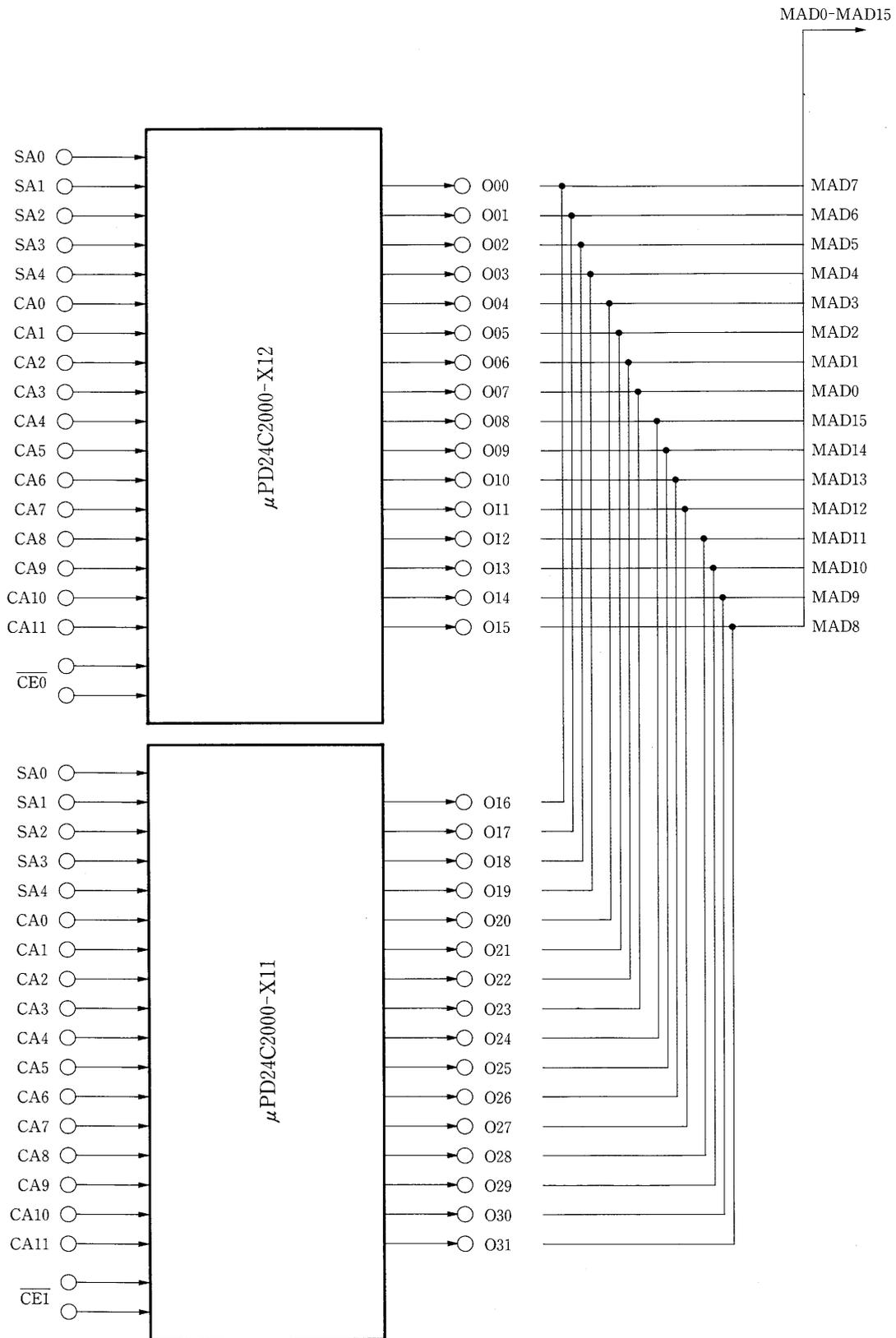


図 4-2 AGDCへの接続例



**Q4.5**

内容は? 機能・動作 特性 応用

漢字CGを表示メモリ領域に割り当てた場合、CLK=8 MHzでは、CGのアクセス・スピードが不足?

**A4.5**

不足することがあります。

CGのアクセス時間は、一般的に200~250 nsです。μPD72120のCLK=8 MHzとするとAGDCのメモリ・アクセス時間は $2 \times \text{CLK} - (t_{\text{DKA}} + t_{\text{SKD}}) \geq 200 \text{ ns}$ となります。

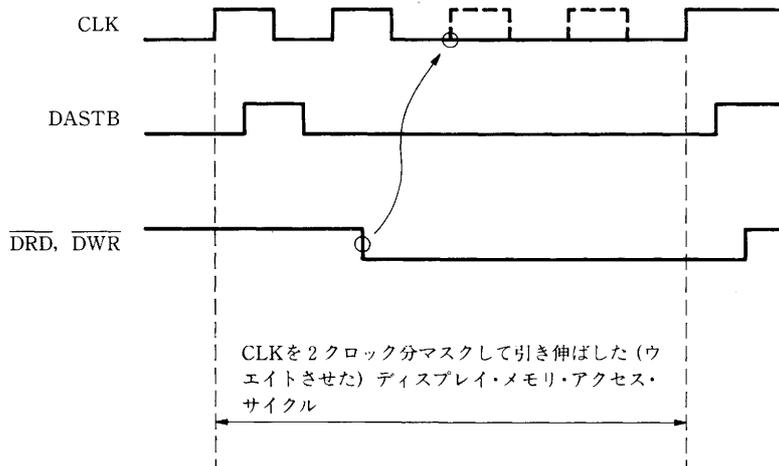
したがって、μPD72120の動作クロックを止める必要が生じます。

ただし、μPD72120側には、汎用CPUのようなウエイトまたはレディに相当する制御信号がないため、以下のような手段でμPD72120の動作クロックを止めることができます。

また、μPD72123には、DWAIT端子がありますので、これによりCGのアクセス時間を制御することができます。

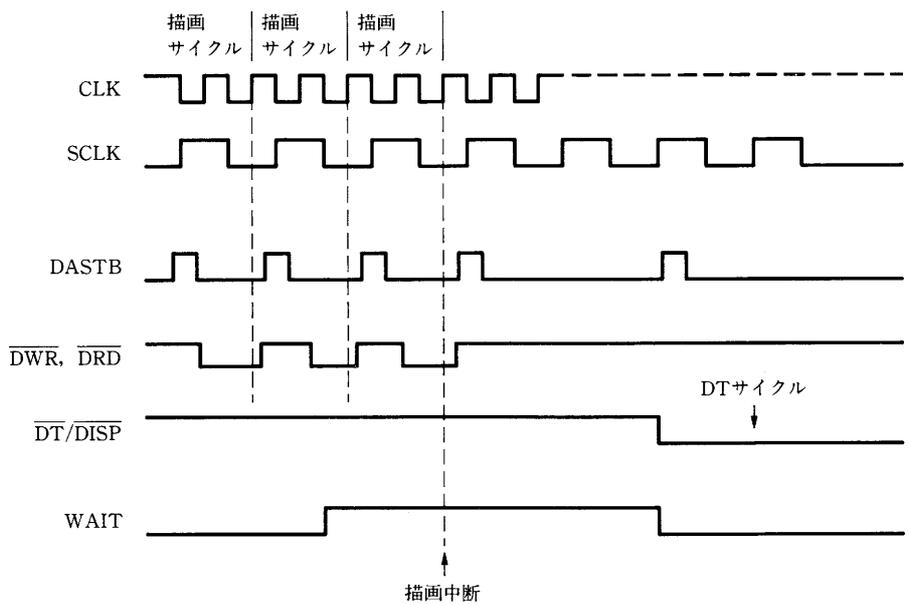
○CLK制御による描画サイクル時間の引き伸ばしについて

AGDCがメモリ・アクセスする際に出力する $\overline{\text{DRD}}$ 、 $\overline{\text{DWR}}$ 信号がアクティブ状態となったあとのCLKをマスクすること（ロウ・レベルを保持し続ける）で、見かけ上AGDCはウエイトが入ったのと同じような状態を保ちます。



このCLKマスクを行う場合、AGDCがアクセスを一時的に中断することを示すWAIT信号がハイ・レベルになったあと、データ転送サイクル ( $\overline{\text{DT}} \downarrow$ ) までの時間に最大3SCLKで終了する描画サイクル+ $2\alpha$ <sup>注1</sup>のサイクルがどこまで入るかを計算すればよいことになります。

ただし、このとき、描画サイクルの拡張がこのあとに続くDTサイクル、リフレッシュ・サイクルをディスターブすることのない範囲であることに注意する必要があります。



これらの条件により、以下の計算式が得られます。

$$4 \cdot t_{SCLK} - t_{DSKWT} \geq \{(1 + \alpha) + (2 + \alpha)\} \times t_{CLK} + (\text{ウェイト信号サンプリングのためのセットアップ}^{\text{注2}})$$

CGメモリ使用の場合には、 $\alpha = 1$ 、すなわち1回のウェイト挿入で十分と考えられるため、 $t_{CLK} = 125 \text{ ns}$  (8 MHz) としても、 $t_{SCLK} \geq 192.5 \text{ ns}$  ( $\approx 5.2 \text{ MHz}$ ) となります。

したがって、 $CLK = 8 \text{ MHz}$ 、 $SCLK = 5.2 \text{ MHz}$ 以下であれば、このようなクロック・マスクが可能となります。

- 注1. 1回の描画サイクルで挿入されるウェイト数です。
- 2. セットアップ =  $t_{WSKL} + 20 \text{ ns}$  (内部でのディレイ)

**備考** 描画中断は、WAIT信号をCLKの立ち上がりでサンプリングしWAIT信号がハイ・レベルであれば最大3CLK (DCサイクル起動時、3CLK、D1サイクル起動時2CLK) 後となります。

## Q4.6

内容は？

機能・動作
 特性
 応用

表示メモリのアクセスを8ビットにできる？

## A4.6

AGDCは、画像メモリ側のデータ・バスは16ビット固定で扱います。画像メモリ側のデータ・バス幅を8ビット幅にすることは、普通できません。

やむを得ず、8ビット幅にする場合は、16ビットの接続において、上位、または下位の8ビット分すべて取ります。この場合、AGDCからのアドレッシングは架空の8ビットを含めた16ビットに対して1つのアドレスが割り付けられることとなります。

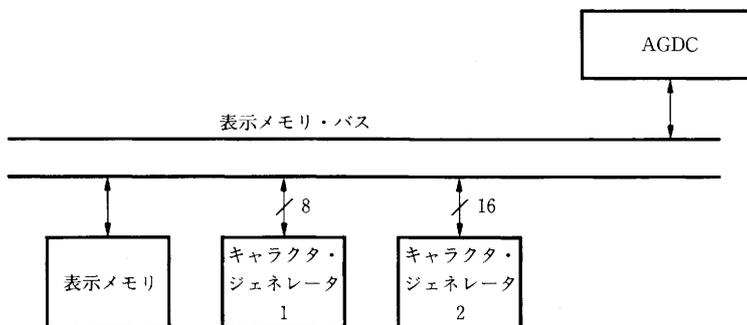
## Q4.7

内容は？

機能・動作
 特性
 応用

表示メモリ・バスにバイト単位かワード単位のCGを割り当てたとき

- (1) コピー・コマンドでのデータ転送方法は？
- (2) ホストからデータを読み出す方法は？



## A4.7

- (1) ハード、ソフト両方にまたがる制御が必要です。

詳細を以下に示します。

○キャラクタ・ジェネレータ使用例

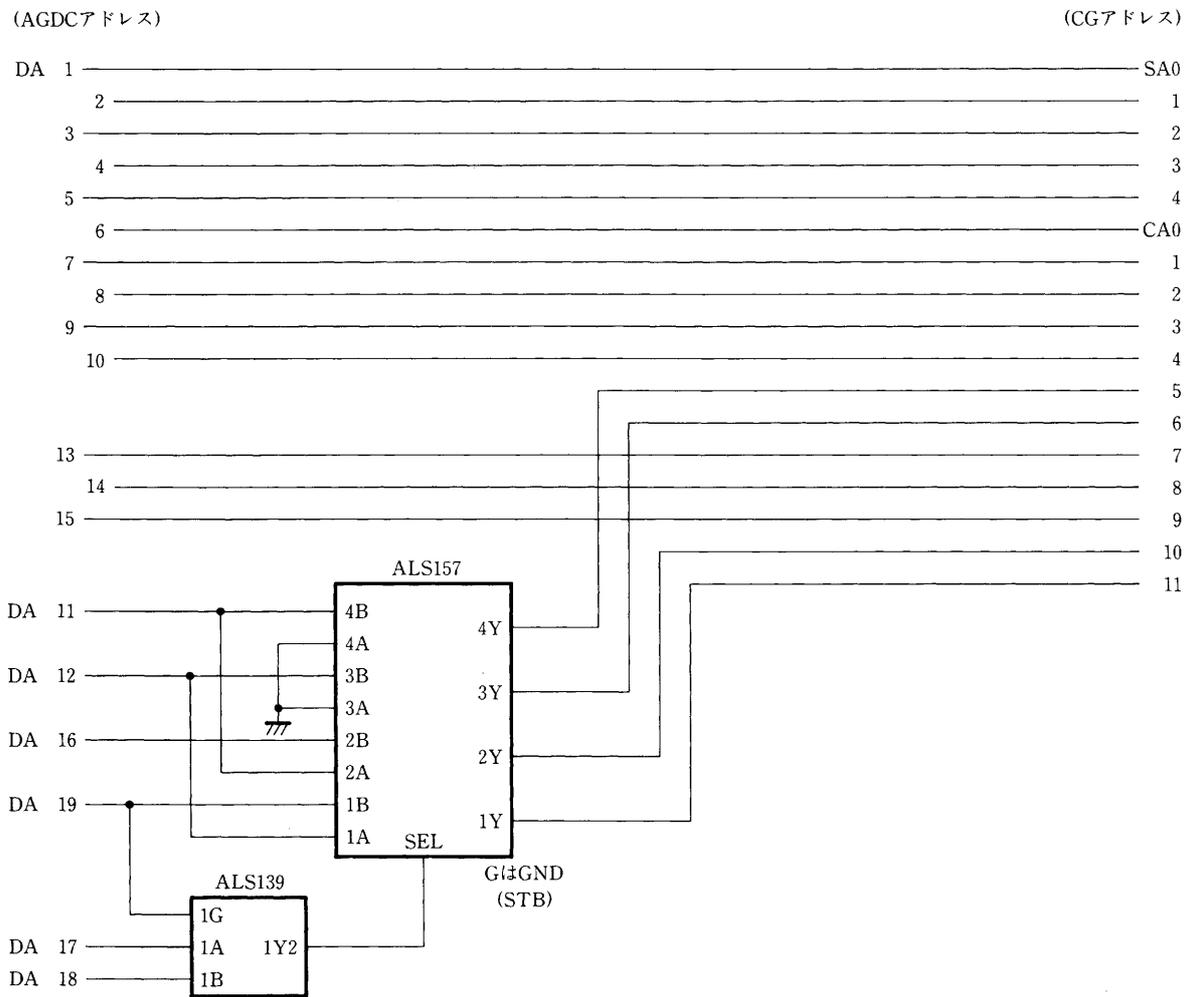
1. 24×24ドットCG使用 (ロウ・スキャン・タイプ)

(条件)

CGとして、 $\mu$ PD24C1000C/D-Y01~Y03を使用する。

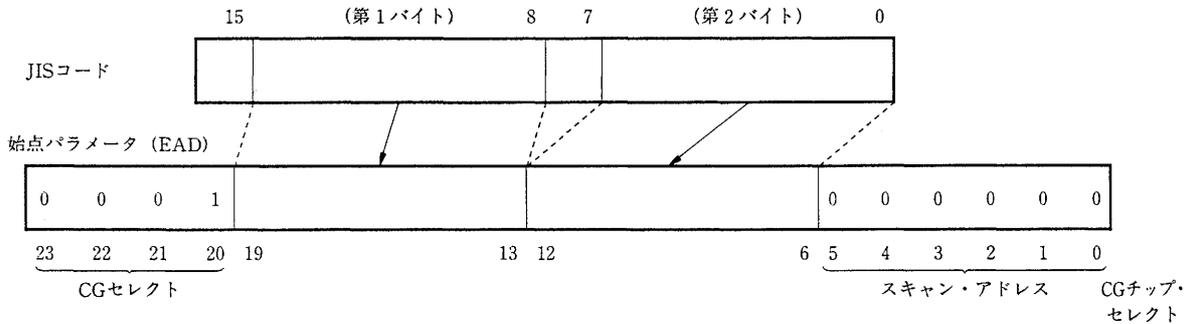
1) ハード構成法

- ① CGでは、非漢字領域と漢字領域で文字コードとデバイスのアドレス対応が異なります(ICメモリのデータ・ブック等を参照してください)。このため、コードのb17, b16, b15に対応するAGDCのアドレス出力をデコードしてマルチプレクスしてください。
- ② キャラクタ・アドレスには、AGDCのDA6~DA19を接続します。
- ③ スキャン・アドレスには、AGDCのDA1~DA5を接続します。
- ④ AGDCのDA0は“0”でフォントの左側16ビット, “1”で右側8ビットを選択します。
- ⑤ CGの選択は、AGDCの最上位側のアドレスを使用します。
- ⑥ CGフォント・データのLSB/MSBと画面左右のLSB/MSBを一致させます。



2) ソフト例

- ① コピー・コマンドは、ソースの始点を絶対アドレスで与える“COPY\_AA”または“COPY\_AC”を使用します。
- ② 始点パラメータは、JISコードをもとに下図のとおり再編成します。



dAD = “0” (EAD1で選択されたワード内のドット位置)

- ③ PITCHS (表示メモリ内の転送源メモリ・プレーンなどの横幅をワード数で設定したもの)  
PITCHS = “2” → ソースの横幅は2ワード (24×24ドット)

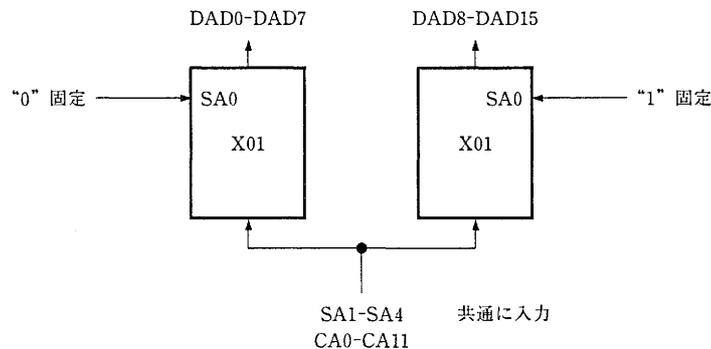
2. 16×16ドットCG使用 (ロウ・スキャン・タイプ)

(条件)

CGとして、μPD24C1000C/D-1-X01を使用する。

1) CGを2チップ使用し、16ビット・データを同時出力する方法

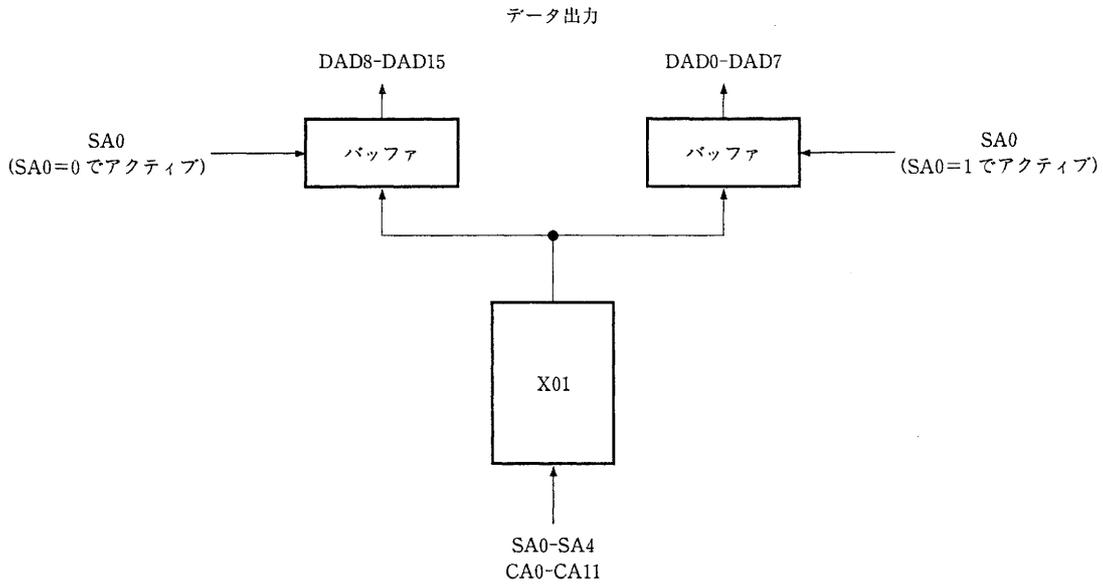
μPD24C1000C/D-1-X01は、1カラム分16ビットをスキャン・アドレスSA0の“H”, “L”により2回に分けて出力します。したがって、あらかじめ下図のように2チップ用意することで高速コピーが可能となります。



(ソフト)

24×24ドットの場合とほぼ同一です。

2) CGを1チップ使用し、8ビットずつ2回に分けてリードする方法



(ソフト)

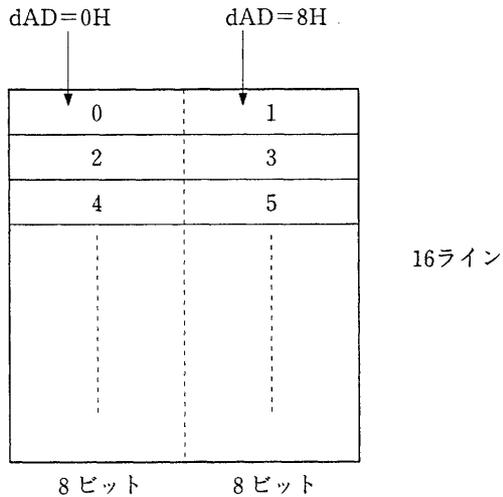
始点パラメータEADの生成方法は24×24ドットの場合と同一です。

次に概要を示します。

3. 16×16ドットCG使用時のコピー・コマンド使用法

2回のコピー・コマンド発行で1文字を描画させます。

- ① 左側の8×16ドットのブロックをdAD=0Hでコピーします。
- ② 右側の8×16ドットのブロックをdAD=8Hでコピーします。



(2) 次の2通りの方法が考えられます。

- 1) GETコマンドによるリード
- 2) AGDCスルーのDMリード

コピー・コマンドで与えた始点パラメータを2倍(左1ビット・シフト)したアドレスでDMリードします。このとき横方向に2回リードが必要です。さらに、CGのスキャン・アドレスをインクリメントしつつ、フォントの縦幅ドット数のリードが必要です。

#### Q4.8

内容は？

機能・動作 特性 応用

表示メモリ・バス上のHLDRQを最高優先度にする方法は？

#### A4.8

AGDCが内蔵するバス・アービタの優先順位は、次の4通りです。

- ① リフレッシュ・サイクルの起動, 表示サイクルの起動
- ② HLDRQ要求(表示メモリ・バス上のデバイスからの表示メモリ・バス使用要求)
- ③  $\overline{\text{CSDM}}$ 要求(システム・バス上のデバイスからの表示メモリ・バス使用要求)
- ④ 描画サイクルの起動

したがって、AGDCがリフレッシュ・サイクル, 表示サイクルの起動をしないように設定することで、HLDRQ要求を最優先順位にできます。

##### リフレッシュ・サイクル起動停止の設定方法

DISPLAY\_CTRLレジスタのRE (Refresh Enable) フラグを0にセットします。

##### 表示サイクル起動停止の設定方法

DISPLAY\_CTRLレジスタのSD (Stop Display) フラグを1にセットします。

#### Q4.9

内容は？

機能・動作 特性 応用

描画サイクルの起動を待たせるフラグ/コマンドはある？

#### A4.9

AGDCでは、描画サイクルの起動を待たせるフラグ/コマンドはありません。ただし、表示プロセッサを使用しないという条件であれば、外部からのHLDRQ信号の入力により、描画サイクルを待たせることができます。

**Q4.10**

内容は？ 機能・動作 特性 応用

リフレッシュ・サイクル直前にHLDRQを入力した場合どうなる？

**A4.10**

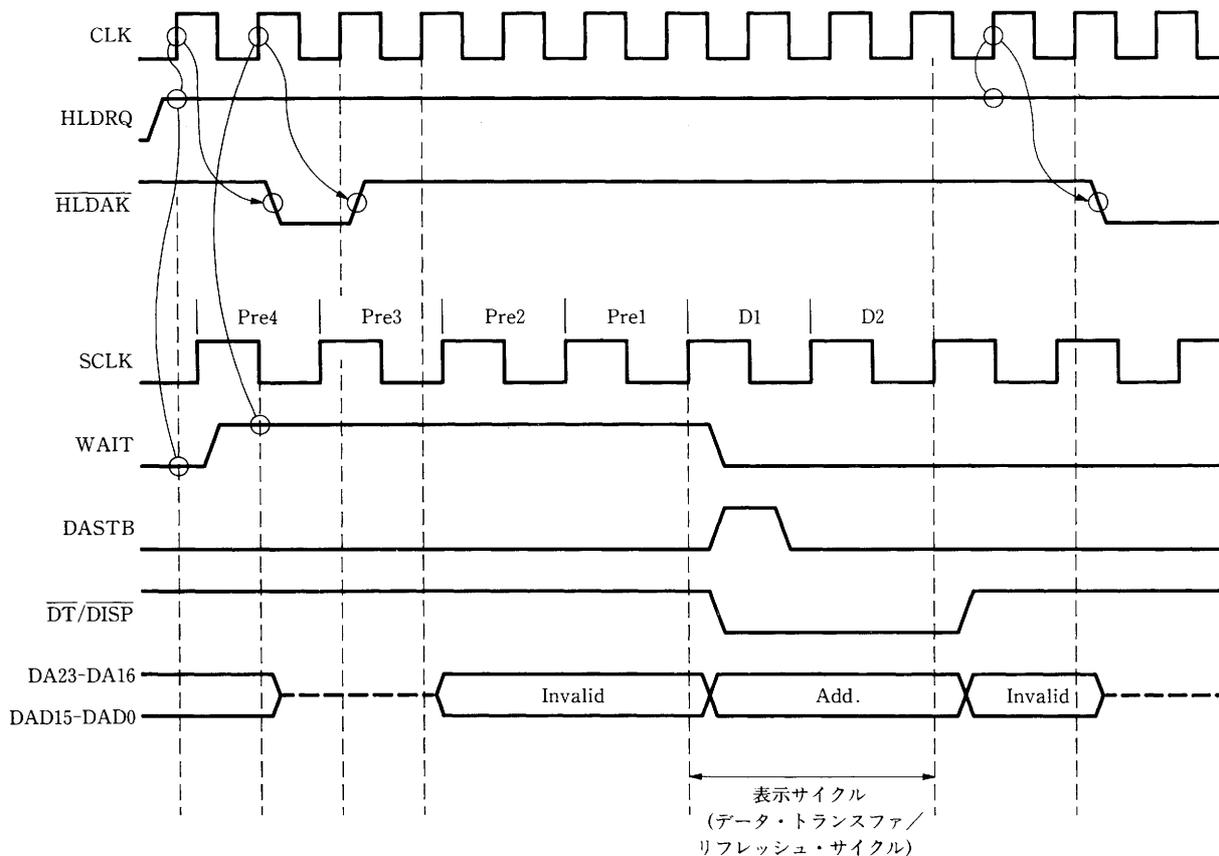
AGDCが表示／リフレッシュ・サイクルを起動するとき、サイクルが起動される4クロック前(SCLK)にWAIT出力信号はアクティブとなり、サイクルが起動される最短1クロック前(CKL)に $\overline{\text{HLDAK}}$ 出力信号はインアクティブとなります。

WAIT信号がアクティブになってから、HLDRQが入力された場合は、表示／リフレッシュ・サイクルが終了するまで $\overline{\text{HLDAK}}$ はアクティブ状態になりません。

図4-3に表示サイクル起動時にHLDRQがアクティブとなるタイミング例を示します。

なお、表示サイクル部は、リフレッシュ・サイクルを含みます。

図4-3 表示サイクル(リフレッシュ・サイクル)の直前にHLDRQが入った場合のタイミング

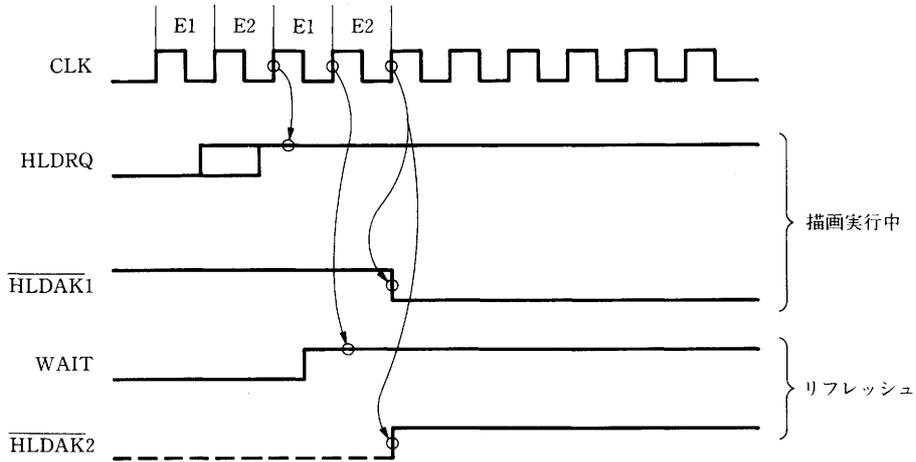


備考 リフレッシュ・サイクルの場合、 $\overline{\text{DT}}$ はハイ・レベルになります。

Q4.11

内容は？ 機能・動作 特性 応用

HLDRQ入力直後にWAIT (リフレッシュ要求) がアクティブになると、 $\overline{\text{HLDAK}}$ はどうか？



備考  $\overline{\text{HLDAK1}}$ ……HLDRQで起動，ロウ・レベルになる。  
 $\overline{\text{HLDAK2}}$ ……WAITで起動，ハイ・レベルになる。

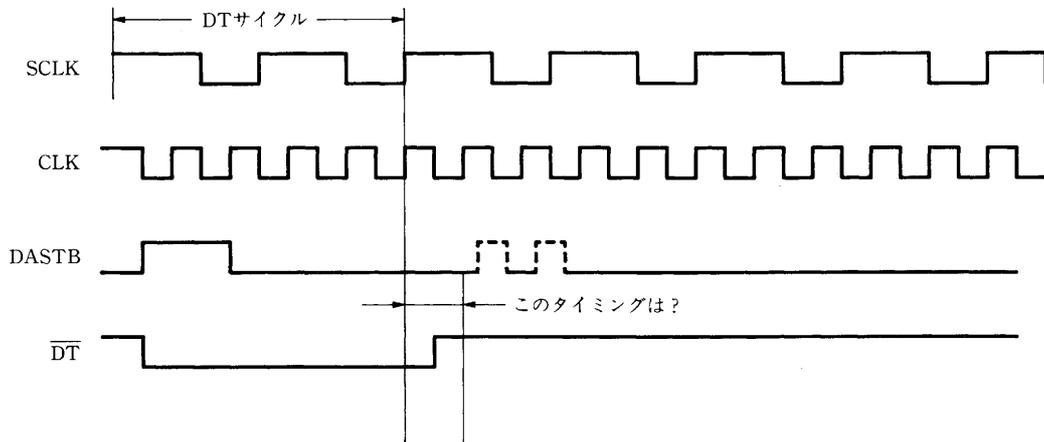
**A4.11**

このタイミングでは、 $\overline{\text{HLDAK}}$ はハイ・レベルのままです。

**Q4.12**

内容は？ 機能・動作 特性 応用

データ・トランスファ・モードにおいて、DTサイクル終了から描画サイクル開始までの最小時間は？



**A4.12**

$t_{SVK}$  (CLK=8 MHzの場合125 ns) です。

**Q4.13**

内容は？ 機能・動作 特性 応用

WAIT信号は、 $\overline{DT}$ 出力タイミングの4SCLK前にアクティブとなる？

**A4.13**

アクティブとなります。

## Q4.14

内容は？

機能・動作 特性 応用

WAIT信号がインアクティブとなるタイミングは？

## A4.14

WAIT信号は表示プロセッサが起動するDTサイクルおよびリフレッシュ・サイクルが終了する1メモリ・サイクル前（SCLK2クロック分前）のSCLK立ち上がりに同期して、立ち下がります。

DTサイクル、リフレッシュ・サイクル別のWAIT信号インアクティブ・タイミングは以下のとおりです。

- 1) DTサイクルが2回連続する場合 ……2回目のDTサイクル開始時のSCLK立ち上がり
- 2) DTサイクル1回のみの場合 ……DTサイクル開始時のSCLK立ち上がり
- 3) リフレッシュ・サイクルが数回連続する場合……最終リフレッシュ・サイクル開始時のSCLK立ち上がり

## Q4.15

内容は？

機能・動作 特性 応用

表示メモリをAGDC以外のモジュールでリフレッシュ動作を行う場合、AGDCの描画サイクルとのアービトレーションを行う方法は？

## A4.15

AGDCのDISPLAY\_CTRLレジスタのREビットを“0”にすることにより、AGDCは表示メモリに対してリフレッシュ・サイクルを起動しません（AGDCは、表示メモリに対して描画、および表示を行うだけのモジュールとなります）。

その際、ほかのモジュールがリフレッシュ・サイクル機能を起動するためにAGDCの描画サイクルを停止させたいときは、AGDCのHLDRQ端子を制御してください。AGDCはHLDRQアクティブ（ハイ・レベル）を入力されると、そのとき実行していた描画サイクルを終了後、 $\overline{\text{HLDAK}}$ を返してグラフィクス・バスをあげわたします。AGDCがグラフィクス・バスをあげわたしている間にリフレッシュ動作を行ってください。

★

Q4.16

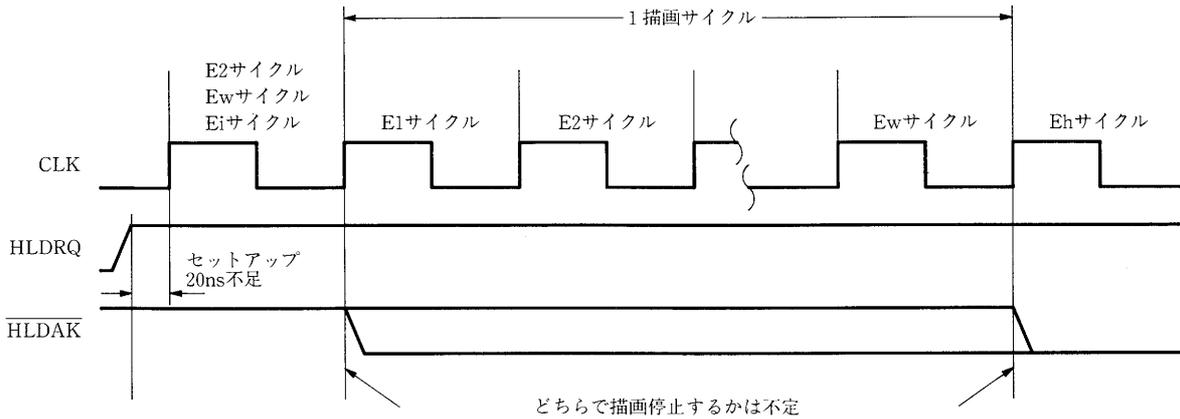
内容は? 機能・動作 特性 応用

描画サイクル中にHLDRQ信号がアクティブになったときのタイミングは?

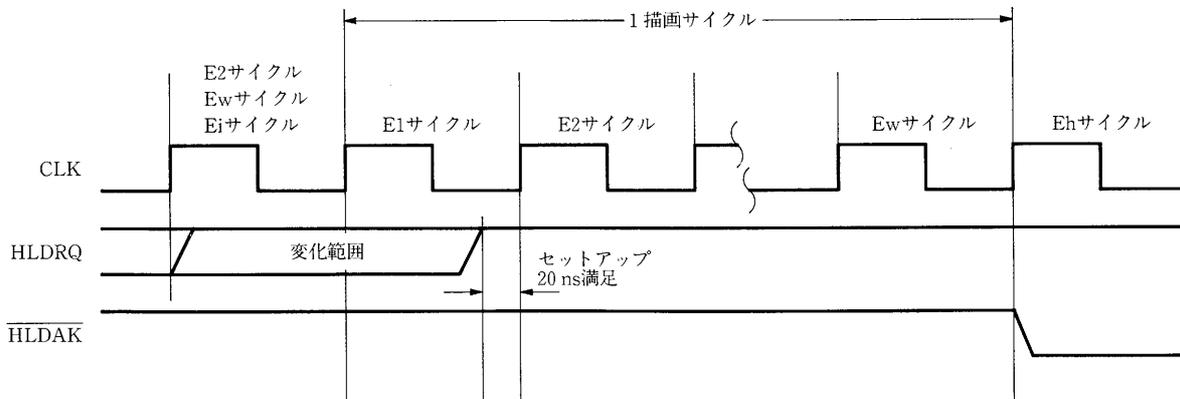
**A4.16**

HLDRQ信号がアクティブになると、描画プロセッサは現在実行中の描画サイクル（ホスト・ダイレクト・アクセス・サイクルを含む）の終了後、グラフィクス・バスを解放します。AGDCはHLDRQ信号をE2, Ei, Ehの立ち上がりでサンプリングしています。以下にCLK非同期でHLDRQ信号を入力する場合とCLK同期でHLDRQを入力する場合のHLDAK出力信号がアクティブになるまでのタイミングを示します。

○CLK非同期でHLDRQ信号を入力する場合



○CLK同期でHLDRQ信号を入力する場合



**Q4.17**

内容は？ 機能・動作 特性 応用

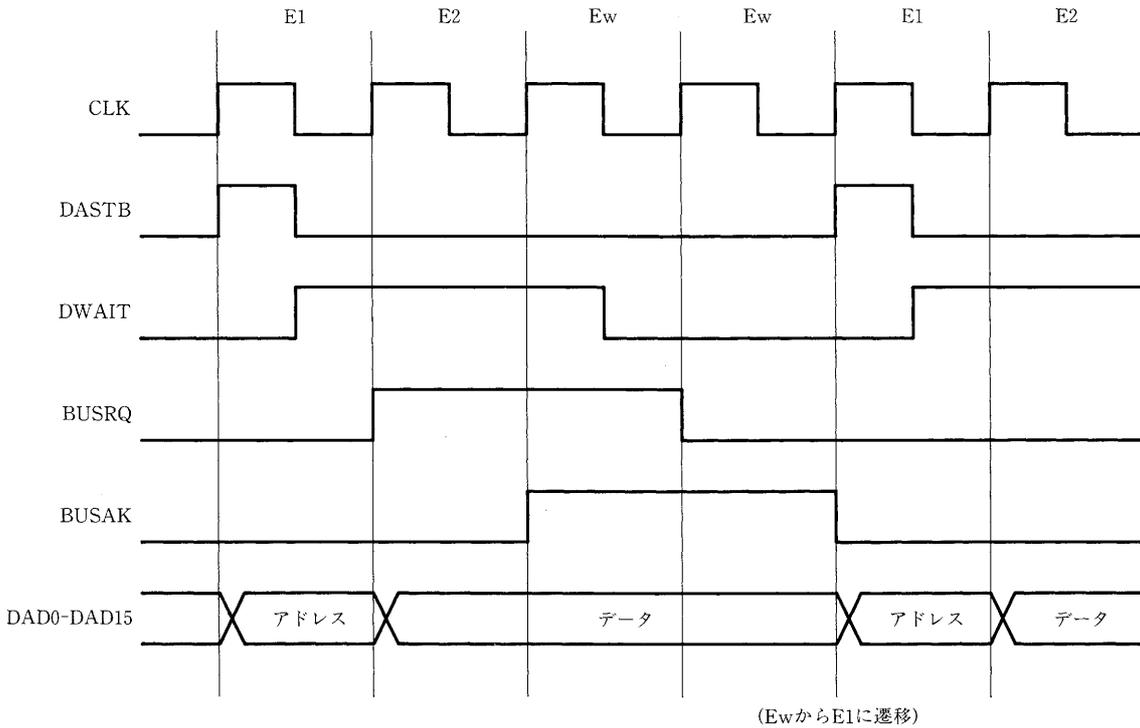
$\mu$ PD72123を描画アクセラレータとして使用する場合、DWAITのみで、 $\mu$ PD72123に通常グラフィクス・バスを占有させないで、バス・アービトレーションを行うことは可能？

**A4.17**

可能です。ただし、次の条件が満足されていることが前提条件となっています。

[条 件]

$\mu$ PD72123のDASTB出力信号がアクティブになったことを受けて、外部回路でバス・アービタに対してBUSRQ信号を入力し、バス・アービタがBUSAKを返すことにより、バス・アービタが $\mu$ PD72123に対してバス使用权を与えることを示すシステムにおいて、 $\mu$ PD72123は、DWAIT入力信号がインアクティブになるとEwサイクルからE1サイクルに遷移する場合があります。このとき再びDASTB信号がアクティブとなります。その際、外部のBUSRQ信号、およびBUSAK信号は、DASTB信号がアクティブになる前にインアクティブの状態ではなりません。



BUSRQ：バス使用权要求信号

BUSAK：バス使用权許可信号

備考 詳細は $\mu$ PD72123のユーザーズ・マニュアル「7.4.5  $\mu$ PD72123に通常グラフィクス・バスを占有させない方法」を参照してください。

**Q4.18**

内容は？ 機能・動作 特性 応用

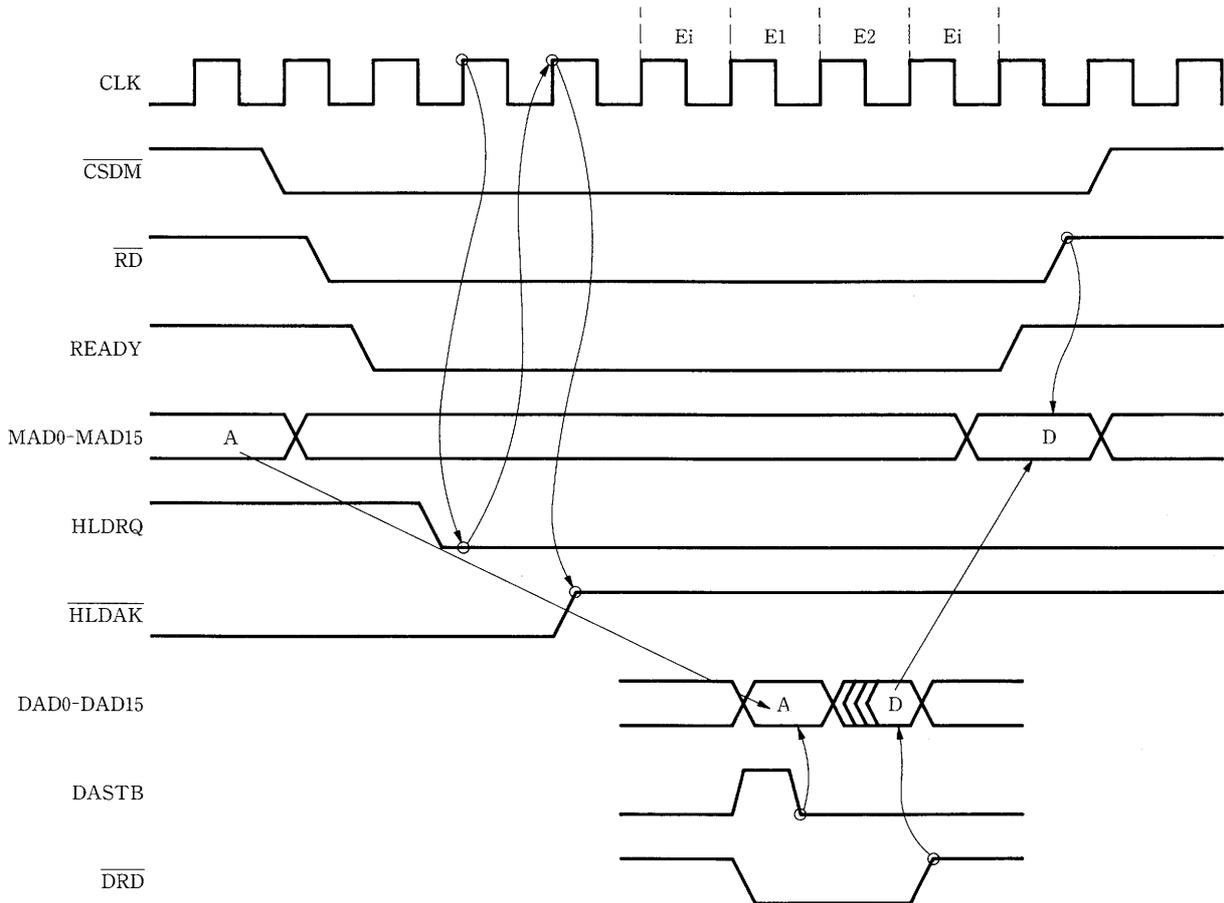
HLDRQ入力信号がアクティブのときに、CPUよりホスト・ダイレクト・アクセスがなされた場合の $\mu$ PD72123のグラフィクス・バス・アービトレーションはどのようなタイミングになる？

**A4.18**

HLDRQ入力信号がアクティブのとき、CPUがホスト・ダイレクト・アクセスを起動したときのタイミングを以下に示します。

メモリは、ノンウエイットの2クロック・サイクルでアクセスできるものと仮定します。

図4-4 リード・サイクルの場合のタイミング



## Q4.19

内容は？

機能・動作
 特性
 応用

$\overline{\text{HLDAK}}$ 出力時の各端子のレベルは？

## A4.19

$\overline{\text{HLDAK}}$ アクティブ時のイメージ・バス側の各端子のレベルは次のとおりです。

DUBE：ロウ・レベル（ワード・アクセスを想定）

DLBE：ロウ・レベル（ワード・アクセスを想定）

$\overline{\text{DWR}}$ ：ハイ・レベル

$\overline{\text{DRD}}$ ：ハイ・レベル

DASTB：ロウ・レベル

$\overline{\text{DT}}/\overline{\text{DISP}}$ ：ハイ・レベル

GCSR：ロウ・レベル

## Q4.20

内容は？

機能・動作
 特性
 応用

CSモードでも、ウェイト・サイクルを挿入できる？

## A4.20

$\mu\text{PD72123}$ をCSモードで使用した場合、非表示区間においてはウェイト・サイクルを挿入することは可能です。ただし、表示区間では描画と表示が2クロック単位で切り替わるために、ウェイト・サイクルを挿入することはできません。

したがって、アプリケーション・プログラムでは、ウェイト・サイクルを挿入しなければならないタイミングがどこで現れるかわからないため、CSモードではウェイト・サイクルを挿入することはできません。「 $\mu\text{PD72123}$ のユーザーズ・マニュアル「図7-14」」を参照してください。

また、DTモードで $\text{CLK}=\text{SCLK}$ にした場合には、1ウェイト挿入可能です。

## 第5章 表示制御

<b>Q5.1</b>	内容は？	<input type="checkbox"/> 機能・動作 <input type="checkbox"/> 特性 <input checked="" type="checkbox"/> 応用
DTモードでのSCLK, DTT, HD, WC, DAD+の設定方法は？		

### A5.1

以下に $\mu$ PD72120でのDTT, SCLK, HD, DAD+, WCの設定方法を説明します。  
 $\mu$ PD72123でも同様にして考えることができます。

#### DTT

メモリ・プレーン横幅が表示画面横幅と等しくないときに必ず“0”としてください（それ以外では，“0”，“1”どちらでもかまいません）。

#### SCLK

SCLKとして使用可能な周波数は1.66…MHz- 8 MHzまでです（上限は回路の速度そのものに、下限については、ダイナミック回路により構成されるレジスタのリフレッシュ周期により決定されます）。

ドット・クロック（DOTCLK）をn分周したものをSCLKとします。

nは、 $1.666\cdots\text{MHz} \leq \text{SCLK} \leq 8\text{ MHz}$ の条件より決定します。

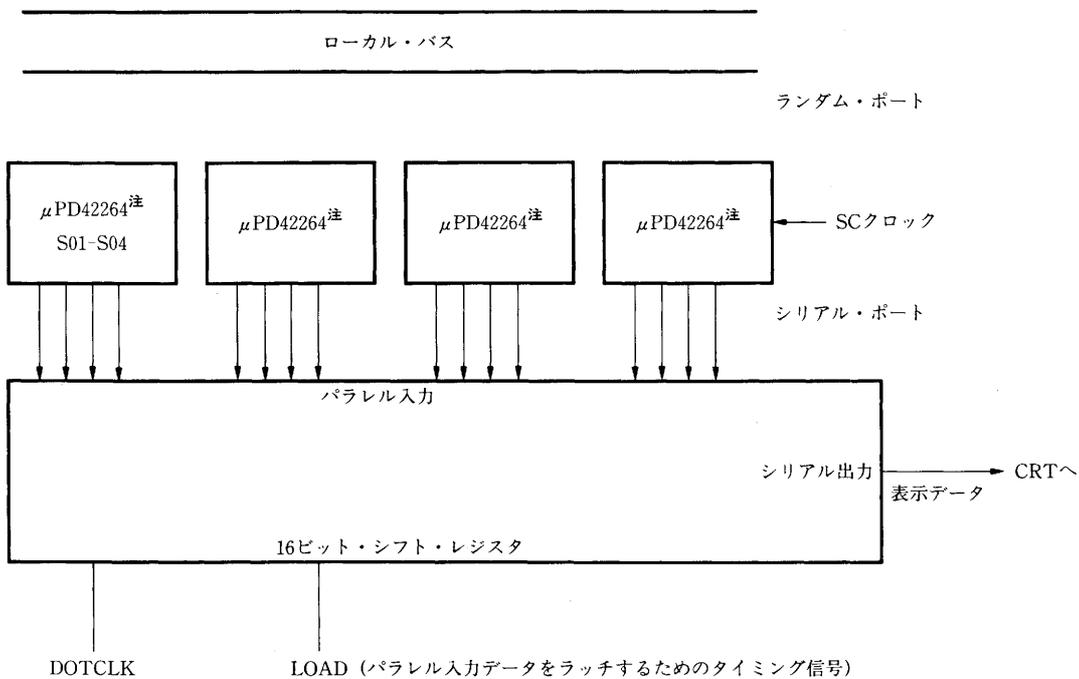
たとえば、1120×750の解像度で、DOTCLK=48 MHzならば、

- SCLK=48 MHzの4分周=12 MHz … > 8 MHz
  - SCLK=48 MHzの8分周= 6 MHz …
  - SCLK=48 MHzの16分周= 3 MHz …
  - SCLK=48 MHzの32分周=1.5 MHz… < 1.6 MHz
- } 使用可能

SCLKとしては、3 MHzと6 MHzの2通りがあります。

下記のようなハードウェア構成として考えます。

- ・ 64 Kワード× 4 ビットのデュアルポート・グラフィクス・バッファ 4 個を 1 モジュールとする。
- ・ 16 ビットのシフト・レジスタを用いる。

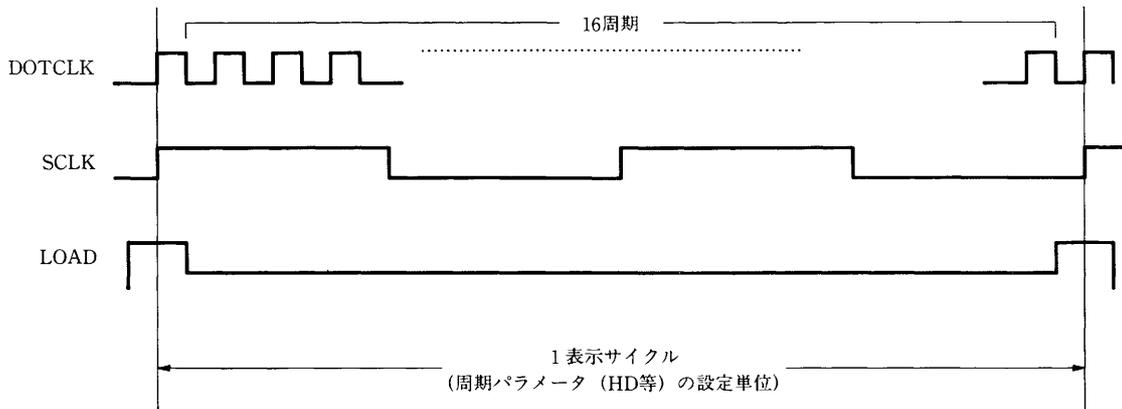


注 当社のデュアルポート・グラフィクス・バッファ (64 Kワード× 4 ビット構成) です。

① SCLK = 6 MHz の場合の HD, DAD+, WC 設定例

**HD**

この場合、DOTCLK 16 周期で 1 表示サイクル (SCLK 2 周期) となりますので、タイミングは下図のようになります。



横方向1120ドットの画面ならば、 $HD=1120/16=70$  [表示サイクル] となります。

### DAD+

$\mu$ PD42264は256ビットのシリアル・データ出力を行うデータ・レジスタを4本内蔵していますが、1本のデータ・レジスタに着目して考えます。1表示サイクル(2SCLK期間)内では1本のデータ・レジスタから1ビットだけデータが出力されます。したがって、256表示サイクルでデータ・レジスタが空になります。正確にはSCクロックを257クロック入力すると1クロック目と同一のデータが出力されます。したがって、256表示サイクルに1回だけ、データ・トランスファ・サイクルを起動してください。

AGDCは表示アドレス・レジスタの下位8ビット(またはACレジスタで規定された8ビット)がすべて0であるという条件で、データ・トランスファ・サイクルを起動します。256表示サイクルに1回という周期で、データ・トランスファ・サイクルを起動するためには、DAD+=+1のインクリメント・モードを選択します。

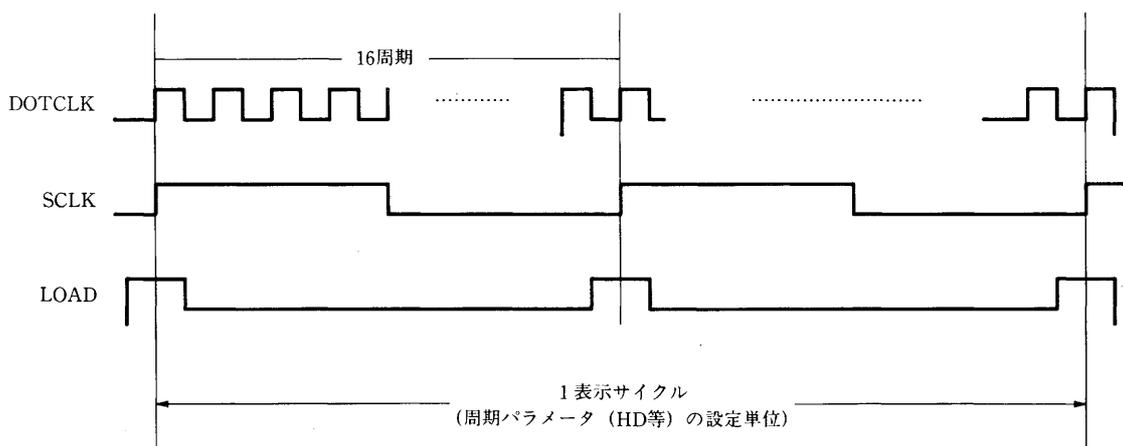
### WC

1表示サイクル期間に16ビット(1ワード)表示するので、横方向1120ドットならば、 $WC=1120/16=70$  [ワード] となります。

## ② SCLK = 3 MHzの場合のHD, DAD+, WC設定例

### HD

この場合、DOTCLK32周期で1表示サイクル(SCLK 2周期)となりますのでタイミングは下図のようになります。



横方向1120ドットの画面ならば、 $HD=1120/32=35$  [表示サイクル] となります。

## DAD+

SCLK = 3 MHzの場合には、128表示サイクルに1回データ・トランスファ・サイクルを起動する必要があるため、DAD+=+2のインクリメント・モードを選択します。

## WC

1表示サイクル期間に32ビット(2ワード)表示するので、横方向1120ドットならば、 $WC=1120/32=35$ [ワード]となります。

## Q5.2

内容は？

機能・動作
 特性
 応用

WC, DAD+, HDの設定方法は？

## A5.2

DTモードでの設定方法を説明します。

HDには表示期間中の表示サイクル (SCLK 2クロック分が1サイクル) 数-1の値を設定します。

図5-1 1 HSYNC周期でのn表示サイクル

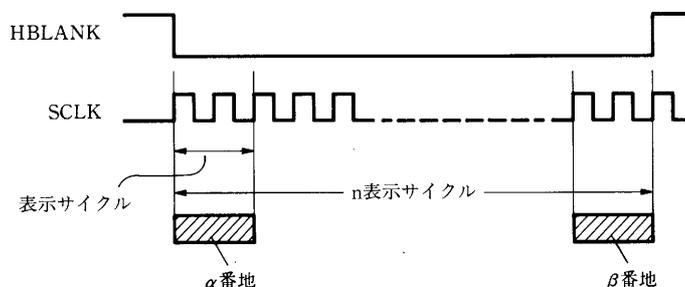


図5-1は、1HSYNC周期に $n$ 表示サイクルに入るため、HDには、 $n-1$ の値を設定します。

たとえば、横方向1024ドットあり、1表示サイクルで16ドット出力すれば $1024/16=64$ 表示サイクルとなり、HDレジスタには、 $HD=3FH$ を設定します。

WC (Word Count) の設定は、表示サイクルでのアドレス・インクリメント数DAD+の設定によって異なります。ここでは、2インクリメント・モード ( $m=2$ ) とします (このときDAD+には001を設定します)。この場合、WCには、 $WC=m \times (n-1) = 2(n-1)$  を設定します。

$n$ : 表示サイクル数

$m$ : 1表示サイクルごとのアドレス・インクリメント値

たとえば、 $HD=3FH$ の場合には、 $WC=2 \times 3FH=7EH$ を設定します。

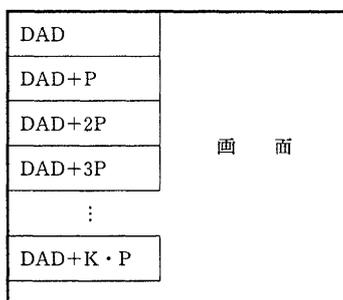
1 インクリメント・モードでは、 $HD=WC$ となります。

以上のように設定する理由は、表示アドレスの計算のためです。

表示アドレスのインクリメントは、HBLANKがロウ・レベルの期間内のみ行われ、HBLANKがハイ・レベルになった時点（たとえば図5-1の $\beta$ 番地）での表示アドレスDADDは、 $DADD=HD \times m + DAD$ となっています。

次のHSYNCの立ち下がり時に $DADD \leftarrow DADD - WC + DISPLAY\_PITCH$ の演算が実行されます。このため1ラスタ下のラインの表示先頭アドレスを1ラスタ上のラインの表示先頭アドレスに比べてDISPLAY\_PITCHだけ増えているようにするためには（図5-2参照） $WC=HD \times m$ の関係を満たしてください。

図5-2 表示アドレス

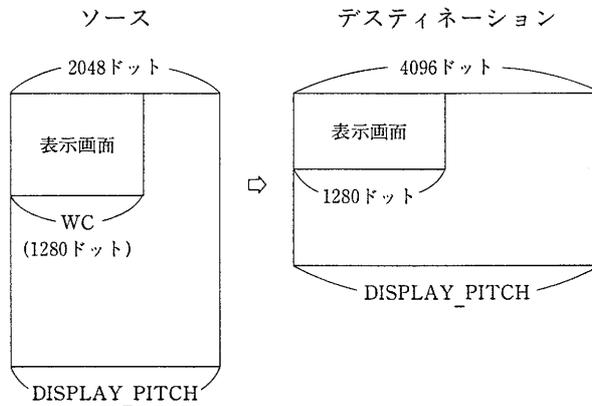


**Q5.3**

内容は？

機能・動作 特性 応用

ソース側とデスティネーション側のPITCHが違うとき、DISPLAY\_PITCH, WC, DADの設定はどうする？



**A5.3**

ソース側、デスティネーション側をおのおの表示したいとき、メモリの構成（ピッチ幅）が異なる場合には、表示したい側のDISPLAY\_PITCHおよびDADの値で設定してください(WCは、表示領域に依存します。したがって、同期信号のHDや表示アドレスの進み方に変化がなければ、変更不要です)。

つまり、ソース側を表示したいときは、ソース領域にあわせたDISPLAY\_PITCHとDADを、デスティネーション側を表示したいときは、デスティネーション領域にあわせたDISPLAY\_PITCHとDADを再設定してください。

**Q5.4**

内容は？

機能・動作 特性 応用

DAD+フラグを「DAD+1からDAD+2」, SCLKを従来の2倍遅い周波数に変更した場合、WCレジスタ設定値はどう変わる？

**A5.4**

WC設定値は、**A5.2** で示したように、 $WC = m \times (n - 1)$  で与えられます。

n：表示サイクル数

m：1表示サイクルごとのアドレス・インクリメント値

SCLKを2倍遅い周期にしても、水平同期周波数や横方向表示ドット数を変更しないのなら、1表示サイクルに表示すべきドット数は変更前の倍になり、表示サイクル数は変更前の半分になります。変更前の表示サイクル数をNとすれば、変更後の表示サイクル数はN/2、また、m=2 (DAD+を2にする) となります。したがって、変更後のWC設定値は、 $WC = 2 \times (N/2 - 1)$  となります。

## Q5.5

内容は？

■機能・動作 特性 応用

表示画面には、プレーン間の変位量を設定するレジスタはある？

## A5.5

ありません。

表示画面が3プレーン (R, G, B) あるとすると、AGDCが表示メモリに対して描画を行う場合、まずRのプレーンに描画し、次にGのプレーンに描画し、最後にBのプレーンに描画します。メモリ・プレーンが、3プレーンある場合には3回描画動作を繰り返します。

したがって、アドレス計算を行う上で、各プレーン間の変位量 (1プレーン当たりのワード数) を示すレジスタ (PDISPSレジスタ) が必要になります。

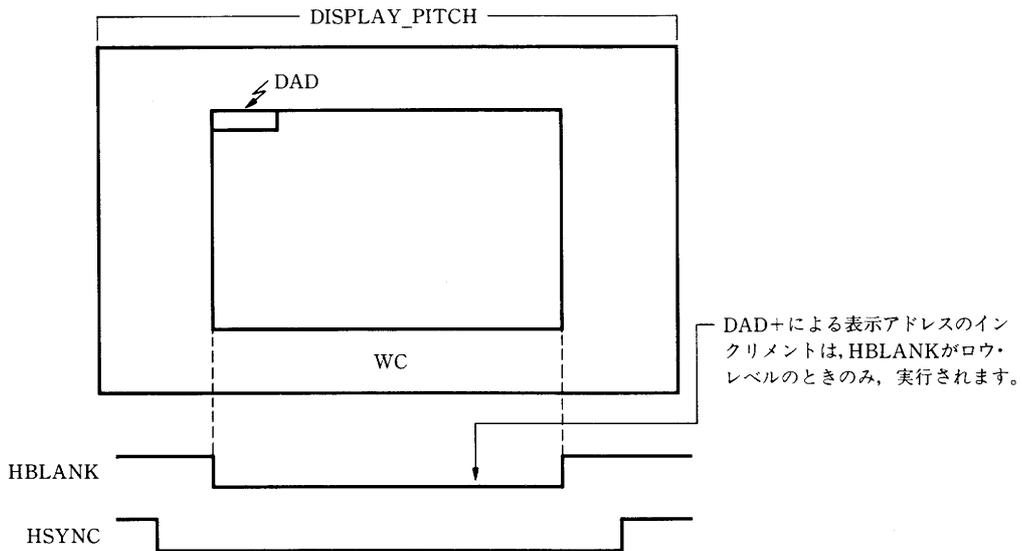
ただし、表示の場合は、3プレーン同時にデータを読み出すようにシステムを設計するので、そのようなレジスタは不要です。

**Q5.6** 内容は? 機能・動作 特性 応用

表示アドレスの更新タイミングは?

**A5.6**

下図に、表示タイミングとDAD、WC、DISPLAY\_PITCHの関係を示します。



次に、AGDCの表示アドレスの演算方法について示します。

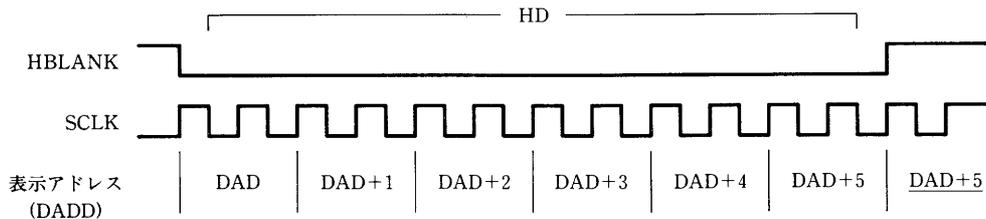
1表示期間中の表示アドレスは、

$$DADD = DAD + DAD+1 + \dots \text{で算出されます。}$$

↑                    ↑

表示アドレス    先頭アドレス

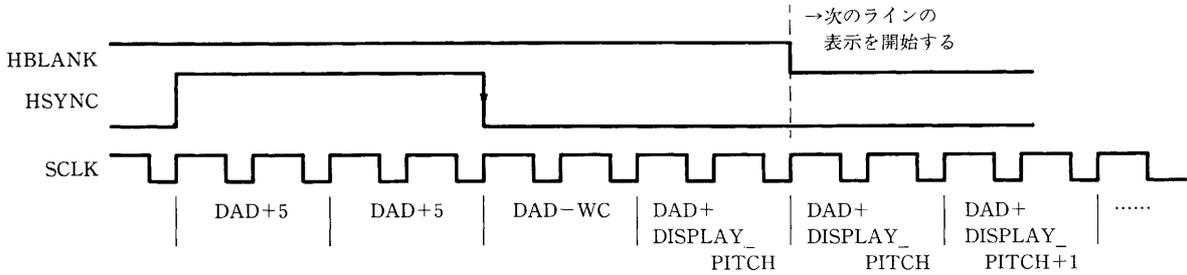
このとき、1水平走査期間中にDAD+による表示アドレスのインクリメントを何回行うかについては、HDにより決定されます。



DAD+レジスタ値=000H (+1のインクリメント)

HD = 005H (6表示サイクル)

HBLANKがロウ・レベルである期間にのみ、表示アドレスはインクリメントされ、HBLANKが立ち上がった後は、1 スキャン・ラインの最終表示アドレス（上図ではDAD+5）を次のHSYNCが立ち下がるまで保持します。このときの動作を次に示します。



WCの演算 DISPLAY\_PITCH  
演算

以上より、WCを005Hと設定した場合、各水平走査ラインの表示先頭アドレスは、 $DAD + DISPLAY\_PITCH \times (K^{注-1})$  となります。

注 K=1, 2, 3..., L/F

**Q5.7**

内容は？	<input checked="" type="checkbox"/> 機能・動作	<input type="checkbox"/> 特性	<input type="checkbox"/> 応用
------	-------------------------------------------	-----------------------------	-----------------------------

PPBSY=1のときに、表示プロセッサのレジスタを書き換えても動作上問題はない？

**A5.7**

DADやGCSRX, GCSRYS, GCSRYSレジスタは、いつ書き換えても問題はありません。しかし、他の表示プロセッサのレジスタ（特に表示制御を行っているSYNCレジスタ、WCレジスタ）の値をプリプロセッサおよび描画プロセッサがビジー中に書き換えてしまうと、AGDC内部で表示と描画のアービトレーションが崩れてしまい、バスが衝突する恐れがあります。

初期化以外でレジスタを書き換えたい場合はPPBSY=DPBSY=0を確認してから行ってください。

## Q5.8

内容は？

機能・動作
 特性
 応用

表示中 (BLANK=L) にDADレジスタの内容を書き換えると

- (1) いつ表示が変わる？
- (2) 表示が乱れる？

## A5.8

- (1) 表示中にDADレジスタを書き換えた場合、次のVBLANK信号<sup>注</sup>が立ち上がった時点より変更した内容による表示が開始されます。



注 VBLANK信号は、AGDCの内部信号であり、外部に出力されているのはBLANK (HBLANKとVBLANKの論理和) 信号です。

- (2) DADレジスタにセットされた値を表示プロセッサが参照するのは、1画面の先頭のみです。したがって、表示画面途中でDADレジスタを書き換えても表示が乱れることはありません。

## Q5.9

内容は？

機能・動作
 特性
 応用

表示制御もリフレッシュ制御もAGDCにさせないとき、表示パラメータは設定しなくてもいい？

## A5.9

DISPLAY\_CTRLレジスタのREビット、およびSDビットに、RE=0, SD=1を設定してください。  
DISPLAY\_CTRLレジスタのREビット、およびSDビットに、RE=0, SD=1を設定した場合には、表示プロセッサのレジスタ72H~7FHの設定は不要です。

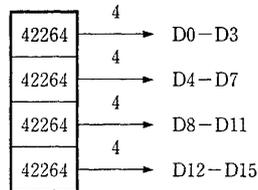
## Q5.10

内容は？

機能・動作特性応用

水平表示ドット数1024で $\mu$ PD42264を使用する場合、DAD+レジスタの設定は？

16ビット構成デバイス



## A5.10

DAD+に1を設定してください。

DAD+レジスタは1表示サイクル(2SCLK分)でどれだけアドレスを進めるかを設定するものであり、あとは上位、下位バイト(上図のD0-D7, D8-D15)を別々のSCクロック( $\mu$ PD42264へ入力),  $\overline{SOE}$ クロックで出力させればよいことになります。したがって16×nビットを1表示サイクル中に表示する場合、+nのインクリメントを設定することになります。

なお、この構成の場合、WC=63 (1024/16-1)を設定してください。

## Q5.11

内容は？

機能・動作特性応用

サイクル・スチール・モード時、DAD+とりフレッシュ・サイクル/表示サイクル/描画サイクルの関係は？

## A5.11

サイクル・スチール・モードは通常のDRAMを使用するとき用いるモードであり、CLK=SCLK(同位相, 同周波数)という条件が必須となります。

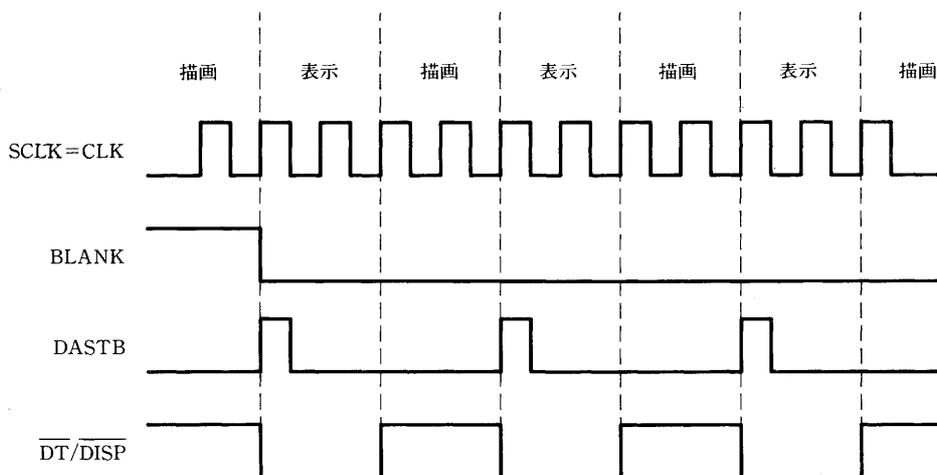
サイクル・スチール・モードを使用する場合は、DTM=0とし、DTT=0, SC=0としてください。また、すべてのサイクルでWAITは意味をもちません。

次に、各サイクルについて説明します。

## 1) 表示サイクル

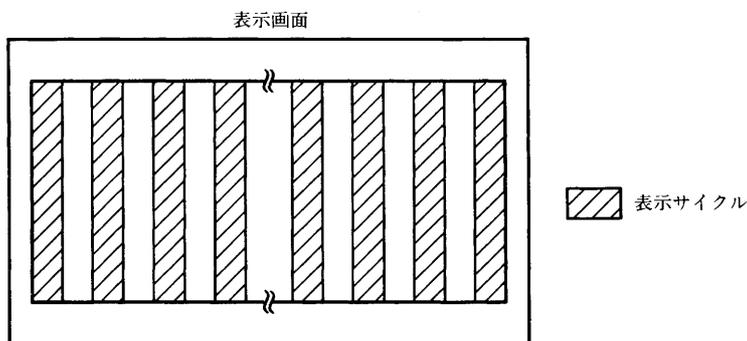
HSがロウ・レベルでかつ、 $\overline{DISP}$ がアクティブになるサイクルが表示サイクルです。

図5-3 サイクル・スチール・モード時の表示タイミング



BLANKがロウ・レベルになった直後のサイクルは必ず表示サイクルであり、以降2サイクルごとに表示サイクルとなります。BLANKがロウ・レベルの期間が奇数サイクル数であれば、最後のサイクルは表示サイクルで終了します。

図5-4 表示タイミングの画面との対応

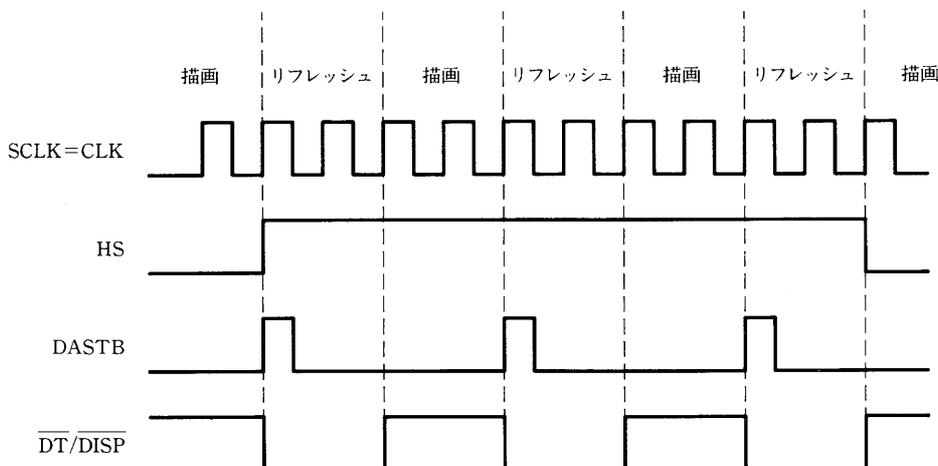


備考 なお、SD=1とすると、表示サイクルは現れず、代わりに描画サイクルを実行できます。

2) リフレッシュ・サイクル

HSがハイ・レベルでかつDISPがロウ・レベルになるサイクルがリフレッシュ・サイクルです。HSがハイ・レベルになった直後の2SCLK (CLK) は必ずリフレッシュ・サイクルで、以降描画可能サイクル、リフレッシュ・サイクルが交互に現れます。

図 5-5 サイクル・スチール・モード時のリフレッシュ・タイミング



リフレッシュ・アドレスは毎リフレッシュ・サイクルごとに+1されます。HSのハイ・レベル幅が奇数サイクルのときは、最後のサイクルがリフレッシュ・サイクルで終了します。なお、RE=0とすると、リフレッシュ・サイクルは現れず、代わりに描画可能期間として使用できます。

### 3) 描画可能サイクル

$\overline{\text{DISP}}$ がハイ・レベルのサイクルが描画可能サイクルです。

### 4) DAD+, ACパラメータとWCレジスタ

サイクル・スチール・モードでは、アドレスの更新は各表示サイクルに行われます。したがって、2SCLKごとに更新するDTモードと異なり、4SCLKごとに1回更新されます。

[DAD+による表示アドレスの更新のタイミング]

- DAD+1, DAD+2, DAD+4, DAD+8, DAD+16, DAD+32のとき図5-3の各表示サイクルで、+1-+32のインクリメントが行われます。

DAD+2以上の指定をした場合のDAD+パラメータとDADの関係は、表5-1を参照してください。WC= $n \times \text{HD}/2$  (小数点以下切捨て) としてください。

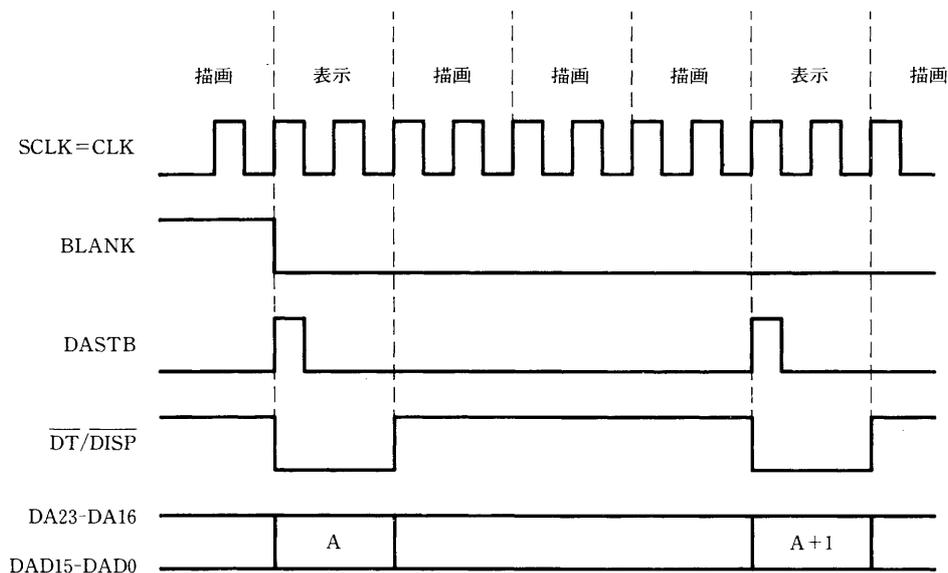
表 5-1 DAD+とDADDの関係

DAD+パラメータ	DADD
DAD+1	制約なし
DAD+2	2の倍数
DAD+4	4の倍数
DAD+8	8の倍数
DAD+16	16の倍数
DAD+32	32の倍数
DAD+1/2	制約なし
DAD+1/4	制約なし

●DAD+1/2のとき

図 5-6 のように、表示サイクルが通常の場合と異なり、4 サイクルに 1 回が表示サイクルとなります。WC=HD/4 (小数点以下切捨て) としてください。

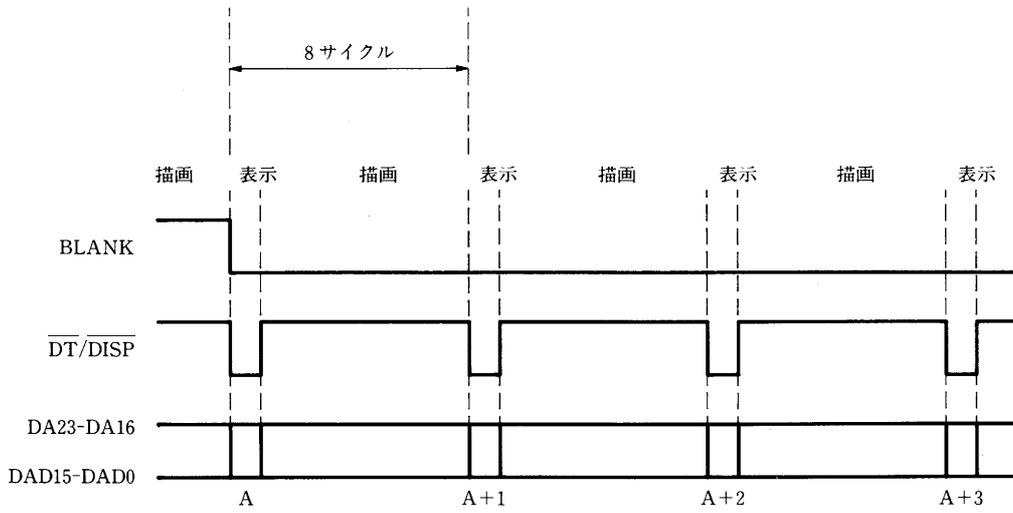
図 5-6 DAD+ 1 / 2 のときの表示タイミング



●DAD+1/4のとき

図 5-7 のように、表示サイクルが通常の場合と異なり、8 サイクルに 1 回が表示サイクルとなります。WC=HD/8 (小数点以下切捨て) としてください。

図5-7 DAD+1/4のときの表示タイミング



[ACパラメータ]

- サイクル・スチール・モードでは常にAC=000Bとしてください。

5) インタレース・モード時のアドレスの進み方

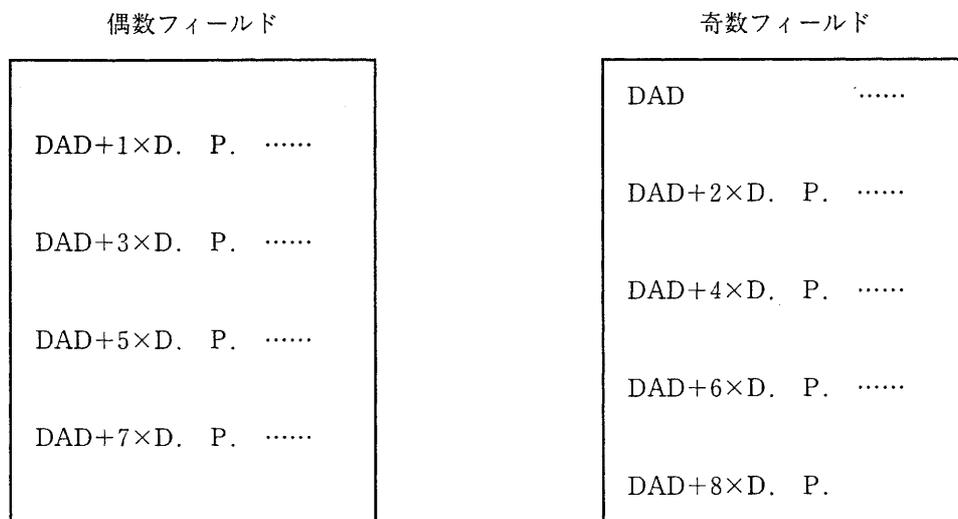
- DTモードのときと同じになります。図5-8を参照してください。

図5-8 ノン・インタレース・モード時の表示アドレスの進み方

DAD	.....
DAD+1×D.	P.注 .....
DAD+2×D.	P. ....
DAD+3×D.	P. ....
DAD+4×D.	P. ....
DAD+5×D.	P. ....

注 DISPLAY\_PITCHのことです。

図5-9 インタレース・モード時の表示アドレスの進み方



**Q5.12**

内容は? 機能・動作 特性 応用

VS, またはVFPの期間でもリフレッシュをする?

**A5.12**

VS, またはVFPの期間においても, HS期間のみリフレッシュ動作を行います。VS, またはVFPの期間においても, HS以外の期間では, リフレッシュ動作を行いません。

## Q5.13

内容は？

機能・動作
 特性
 応用

DTモードにおいて、CLKとSCLKが非同期でいい？

## A5.13

DISPLAY\_CTRLレジスタ内のSCフラグを以下のとおりの設定とすれば問題ありません。

DTM=1…DTモードとなります。このとき、DT/DISP端子はDT信号をDTTフラグの設定に従い発生します。

SC=0 …SCLK≠CLKの場合に、0を設定します（このSCLK≠CLKは周波数、位相の少なくともどちらかが異なることを示しています）。

## Q5.14

内容は？

機能・動作
 特性
 応用

AGDCをインタレース・モードに設定し、転送タイミングのDTTを1に設定した場合、画面上の各走査線の表示開始時（BLANK信号立ち下がり時）は、絶対に転送は行われない？

## A5.14

DTT=1とした場合でも、BLANK信号立ち下がり時に、偶然下位8ビットの表示アドレスがすべて0になった場合には、DTサイクルが起動されます。

## Q5.15

内容は？

機能・動作
 特性
 応用

サイクル・スチール・モード（SCLK=CLK）の表示では、クロック（CLKまたはSCLK）のデューティが50%でなくなることがあるけど、問題ない？

## A5.15

パルス幅のMIN.値μPD72120なら52 ns、μPD72123ならハイ・レベル幅42 ns、ロウ・レベル幅44 nsを満足していれば問題ありません。

**Q5.16**

内容は？

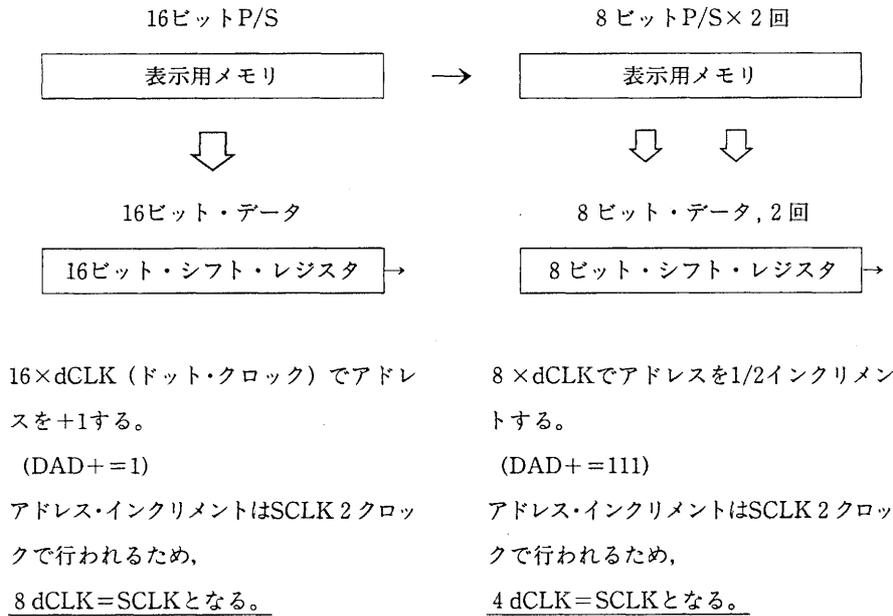
機能・動作 特性 応用

水平走査（ドット・クロック）の遅いシステムへの対応について

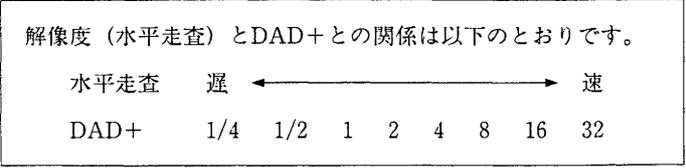
- (1) SCLK, CLKの動作範囲(600 nsMAX.)を越えたらどうなる(ただし,  $\mu$ PD72123のSCLKは2000 nsMAX.)?
- (2) 対策は？

**A5.16**

- (1) レジスタなどの内部回路は、ダイナミック回路により構成されている部分があるため、リフレッシュが必要となります。このリフレッシュはSCLKをもとに起動されているため、この周期を伸ばした場合、レジスタ内容が不定になり正常動作ができなくなることが考えられます。
- (2) 表示メモリよりの出力をシフト・レジスタを使って直列（シリアル）データに変換する際のデータ幅を変更してください。さらに、これに応じてDAD+レジスタでの水平表示によるアドレスのインクリメントの度合いを調整することで対応してください。次にその一例を示します。



**注意** 表示メモリからの出力をインタリーブ(時分割)して、シフト・レジスタに与えるハードウェア上の対策が必要です。



**Q5.17**

内容は？ 機能・動作 特性 応用

AGDC外部同期について

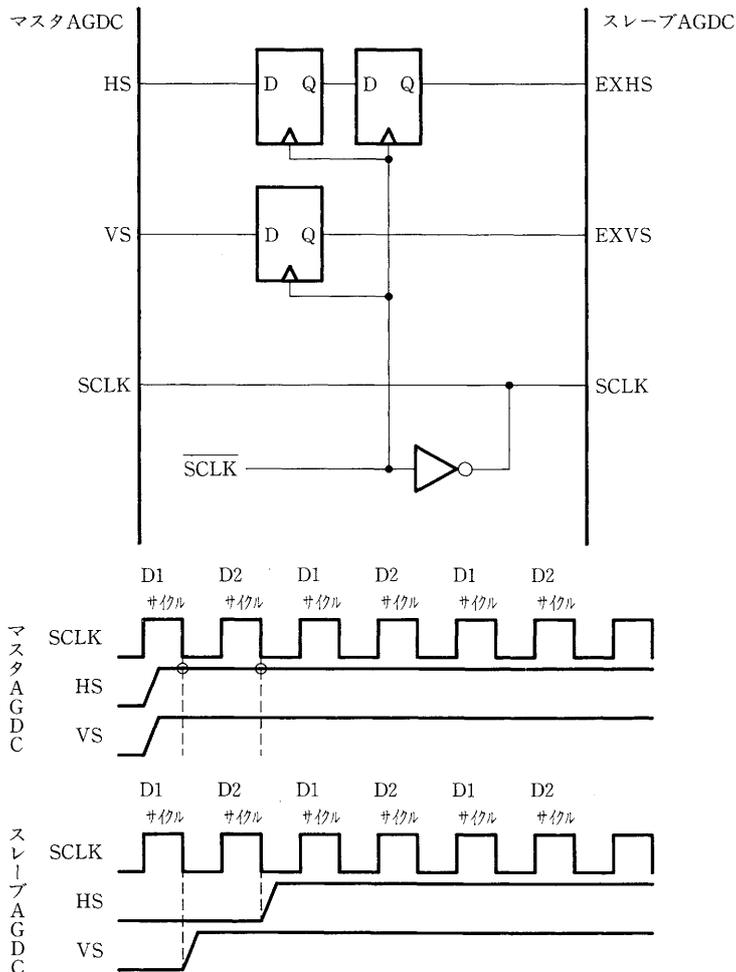
- (1) ソースが同じSCLKで、2つのAGDCを動かしてスーパー・インポーズする場合の注意点は？
- (2) ソースが異なるSCLKで、2つのAGDCを動かしてスーパー・インポーズする場合の注意点は？

**A5.17**

(1) AGDC間（マスタとスレーブ）の接続

条件：同一のクロック発生源からのSCLKを両AGDCに供給します。

マスタAGDCが出力するHS, VSをSCLK立ち上がり同期でスレーブAGDCに入力してください。

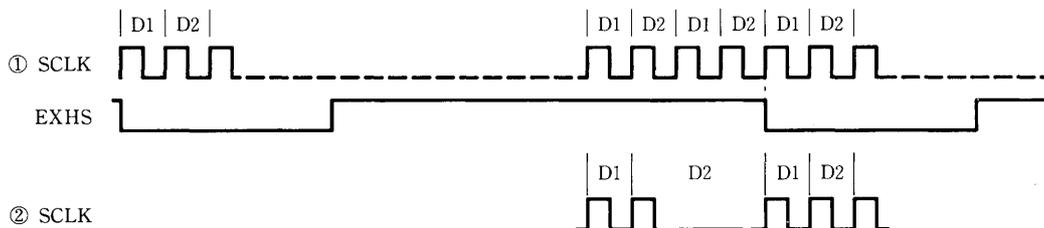


(2) DOTCLKレベルでスーパー・インポーズする場合には、PLLを使ってphase lockさせてください。

**Q5.18**

内容は？ 機能・動作 特性 応用

外部同期モード（スレーブ・モード）において、同期信号のずれを補正するため、補正期間だけSCLK信号を停止させていい？



- ① SCLK 通常のタイミング
- ② SCLK 補正するために一部SCLKを伸ばしたタイミング

**A5.18**

SCLKには、クロック周期として、最低周期数が規定されています。  
この規定内に入る周波数でSCLK信号を停止させることができます。 $\mu$ PD72120のSCLKのスペックを表5-2に示します。

表5-2 SCLKの電気的特性

項目	略号	条件	規格値		単位
			MIN.	MAX.	
クロック周期	$t_{CYSK}$	$t_{CYK}^{\text{注}} \leq t_{CYSK}$	125	600	ns
ハイ・レベル・クロック幅	$t_{WSKH}$		52		ns
ロウ・レベル・クロック幅	$t_{WSKL}$		52		ns
クロック立ち上がり時間	$t_{SKR}$			15	ns
クロック立ち下がり時間	$t_{SKF}$			15	ns

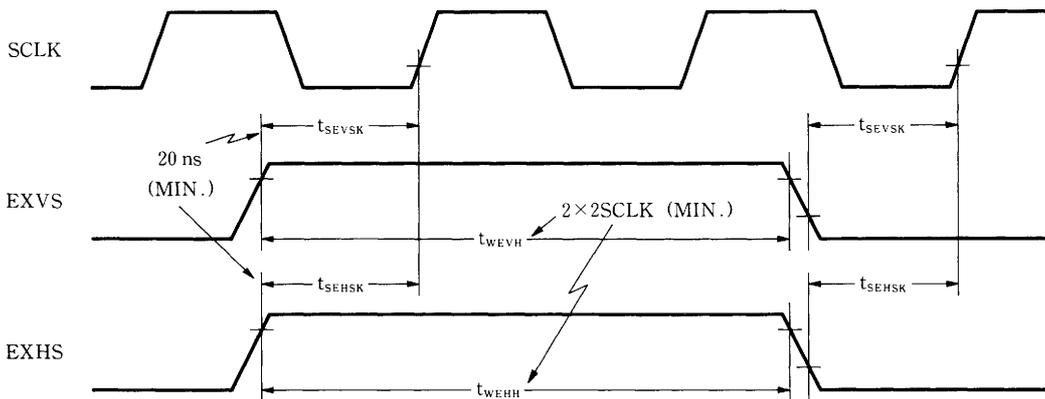
注  $t_{CYK}$ はクロック周期を表します。

**Q5.19** 内容は? 機能・動作 特性 応用

外部同期モード時、EXVS, EXHS入力のタイミング規定は?

**A5.19**

以下のとおりとなっています。



**Q5.20** 内容は? 機能・動作 特性 応用

外部同期モード設定後、EXVS, EXHSに従って動作するタイミングは?

**A5.20**

スリープ・モードに設定したAGDCに対して、EXHS, EXVS信号を入力しない(両信号ともにインアクティブ・レベルを供給する)場合、そのAGDCは内部レジスタに設定されたパラメータ (HS, HBP, HH, HD, HFP, VS, VBP, L/F, VFP) に従って動作します。つまり、あたかもマスタ・モードに設定されたAGDCのように動作します。ただし、水平同期信号HS, 垂直同期信号VSは出力しません。

スリープ・モードに設定したAGDCに対して、EXHS, EXVS信号を入力する場合、EXHS, EXVS信号の立ち上がりで、水平カウンタおよび、垂直カウンタがリセットされます。つまり、EXHS (EXVS) 信号の立ち上がり時点をHS (VS) の立ち上がりタイミングと認識します。以後、次のEXHS (EXVS) 信号が立ち上がるまでは、内部レジスタに設定されたパラメータに従って動作します。

## Q5.21

内容は？

機能・動作
 特性
 応用

外部同期モード時、表示サイクルのD1, D2を同期させるには？

## A5.21

1表示サイクルは、SCLK2クロック分で、最初のサイクルをD1サイクル、次のサイクルをD2サイクルといいます。

AGDCをスレーブ・モードで使用する場合、システム立ち上げ時に外部の垂直同期信号/水平同期信号により、D1サイクル、D2サイクルを同期させます。特にマスタAGDCがある場合は、マスタとスレーブのAGDCの同期をとる必要があります。

同期をとる場合は、DISPLAY\_CTRLレジスタのMASKフラグを“0”に、TCCLフラグを“1”にセットします。D1/D2を決定するD1/D2カウンタはEXVSの立ち上がりで初期化されます。また、EXVSはD2サイクルの立ち上がりで、EXHSはD1サイクルの立ち上がりでサンプリングされます。このため、SCLK, EXVS, EXHSの関係は図5-10のようにしてください。

図5-10 SCLK EXVSおよびEXHSの関係

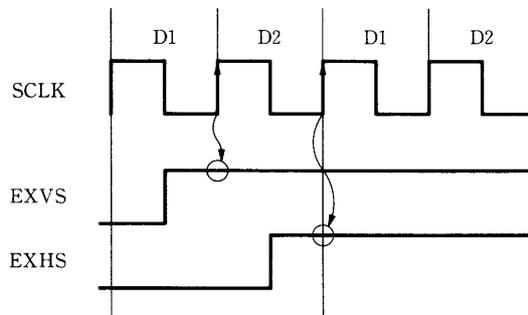
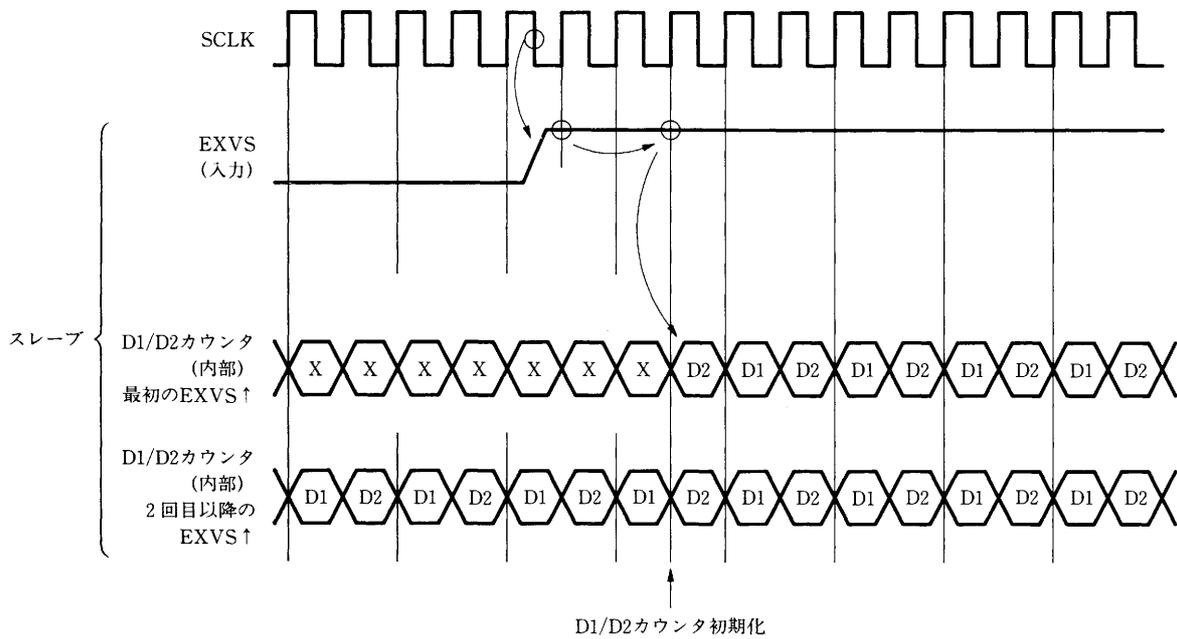


図5-10のようなタイミングでEXVS, EXHSを入力すると、図5-11のようにD1, D2サイクルが決定します。つまり、一番最初SCLKの立ち上がりでEXVS=Hを受けた2SCLK後のサイクルがD2サイクルとなります。

図 5-11 EXVSによるD1/D2カウンタの遷移



条件：スレープAGDCのMASK=0, TCCL=1 (DISPLAY\_CTRLレジスタ内)

**Q5.22**

内容は？ 機能・動作 特性 応用

AGDC間の同期は、ドット・クロック単位であわせられる？

**A5.22**

AGDCは、SCLK単位でのみ同期をとることができます。したがって、ドット単位のずれは外部で対応してください。

SCLK単位でずれるのであれば、AGDC (スレープ側) のDISPLAY\_CTRLレジスタのTCCLフラグが“1”になっているかを確認してください。

## Q5.23

内容は？

■機能・動作 □特性 □応用

外部同期モード時，SCLKの立ち上がりで動作すると説明されているけど，立ち下がりは無視される？

## A5.23

SCLKの立ち下がりについては，チップ内部で参照されていません。

ただし，SCLKで同期信号をとり込むときのデータ保持時間としてハイ・レベルのMIN.幅 $2 \times t_{\text{CYSL}}$ <sup>注</sup>が必要です。

注 実際に入力されるSCLK2クロック分以上の幅を意味しています。

## Q5.24

内容は？

■機能・動作 □特性 □応用

内部同期-外部同期（マスタースレーブ）間の切り替えを行うと，描画メモリの内容に影響する？

## A5.24

どのような条件下でマスタースレーブ・モードの切り替えを行うかにもよりますが，以下の条件下であればディスプレイ・コントロール・レジスタ（DISPLAY\_CTRLレジスタ）のM/Sフラグを0→1と切り替えても，メモリ内容に支障はありません。

- ① AGDCは描画を実行していない。
- ② スレーブとなっているとき，MASKフラグを1とする（MASK=1のときEXCS，HSを無効とするフラグです）。

なお，注意事項として，スレーブ・モード時にAGDCはHSを出力しないので，リフレッシュを定期的に，WAIT信号を利用して行うようにしてください。μPD72123ではIHSを出力することができます。

## Q5.25

内容は？

機能・動作
 特性
 応用

プリンタ・スレーブ・モードにおけるメモリ・リフレッシュは、どうなる？

## A5.25

μPD72123をDTモードで使用した場合、IHS信号がアクティブの期間にメモリ・リフレッシュが行われます。これは、プリンタ・スレーブ・モードで動作させる場合も同様です。

メモリ・リフレッシュをμPD72123に実行させる場合は、メモリ・リフレッシュが間に合う周期でEXVS信号、およびEXHS信号を入力してください。

ただし、プリンタのように、EXVS信号、およびEXHS信号が不定期に入力されるものは、μPD72123にメモリ・リフレッシュを行わせないようにした方がよいと思います。

## Q5.26

内容は？

機能・動作
 特性
 応用

AGDCにバイト・アクセスする場合、同期パラメータの設定手順は？

(DISPLAY\_CTRLレジスタ内のSPSTフラグの処理は？)

## A5.26

DISPLAY\_CTRLレジスタ内のSPSTフラグは、同期パラメータ (HS, HBP, HH, HD, VS, VBP, L/F, VFP) を設定するとき以外は、“0” にしてください。

また、SPST=1のとき7EH, 7FH以外はアクセス禁止です。

以下にその例を示します。ただし、レジスタへのアクセスは7EH→7FHの順序で行ってください。

70H番地 ライト (SPST=1とする) →71Hには上述のようにライトしてはいけません。

↓

7EH番地 ライト

7FH番地 ライト

↓

⋮

↓

70H番地 ライト (SPST=0とする)

備考 71Hをライトしているためのトラブルもよくあるケースの1つです。

**Q5.27**

内容は? 機能・動作 特性 応用

AGDCの同期パラメータの設定において、バイト書き込みでパラメータを設定したが、表示画面の同期がとれない。原因は?

**A5.27**

同期パラメータが与えられたレジスタに設定されていないことが原因として考えられます。

AGDCでは、DISPLAY\_CTRLレジスタ (70H, 71H番地) のSPSTフラグを1に設定すると、レジスタ・ポインタがHSレジスタ (7EH番地) に移ります。バイト書き込みでSPST=1と設定した時点で、レジスタ・ポインタは7EH番地に移っていますから、SPST=1に設定した後に続けて71H番地にパラメータを設定したつもりでも、実際には7EH番地にそのパラメータが書き込まれてしまいます。

表示画面の同期がとれない原因は、このような同期パラメータの設定のずれによると考えられます。

バイト・アクセスのときは、70Hをアクセス (SPST=1) した後、必ず7EH番地をアクセスしてください。

**Q5.28**

内容は? 機能・動作 特性 応用

インタレース・モードでの同期パラメータの設定は?

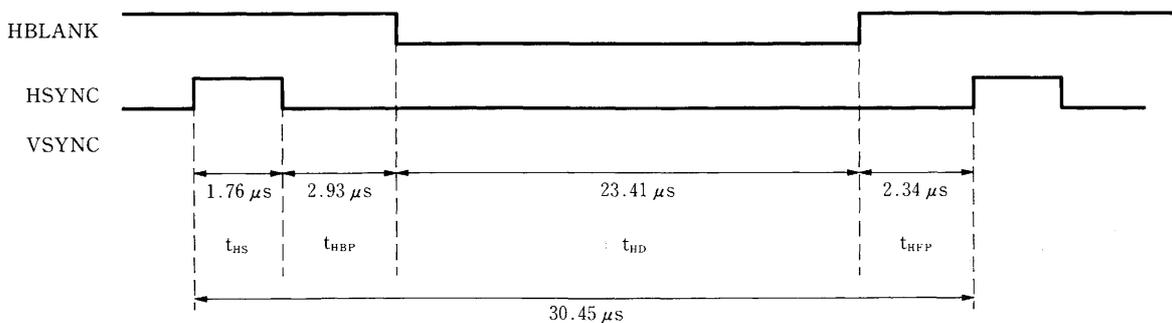
**A5.28**

水平走査周波数32.84 KHz, 解像度1120×750ドットにおける設定例を以下に示します。このとき、水平走査に関するパラメータ (HS, HBP, HH, HD, HFP) は1表示サイクル (2SCLK) を1設定単位として、垂直走査に関するパラメータ (VS, VBP, L/F, VFP) はHSYNC1周期が設定単位となります。

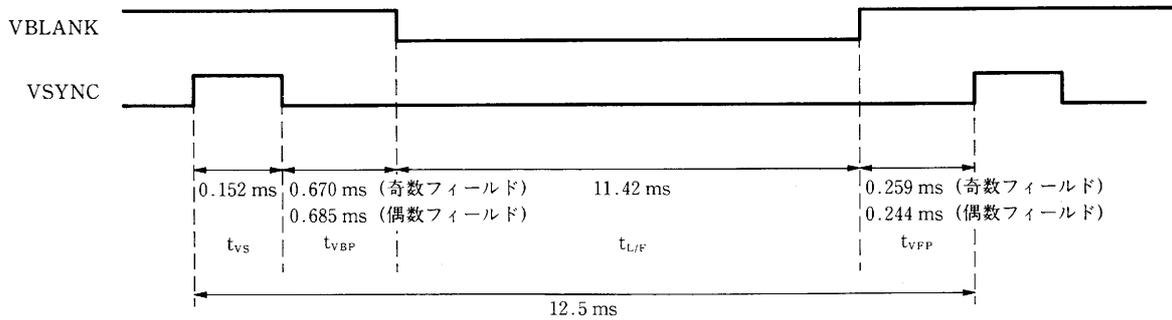
同期パラメータの算出方法

① CRT仕様

[HSYNC]



[VSYNC]



- ・ HSYNC周期=32.84 [KHz]
- ・ インタレース表示
- ・ 解像度 1120×750ドット

② ドット・クロックの決定

水平表示期間=23.41 [ $\mu$ s]

水平解像度 =1120 [ドット]

したがって

$DOTCLK = 23.41 / 1120 \approx 20.9$  [ns]  $\approx 48$  [MHz]

③ SCLKの決定

1.666... [MHz]  $\leq$  SCLK  $\leq$  8 [MHz] の制限がありますので, SCLK=6 [MHz] (DOTCLKの8分周) として使用します。

つまり, 2SCLK期間内に16ドット表示するので, DAD+フラグ(DISPLAY\_CTRLレジスタ)は, DAD+1のインクリメント・モードに設定します。

備考 SCLK=3 [MHz] (DOTCLKの16分周) としてもかまいませんが, この場合, DAD+2のインクリメント・モードになります。

④ DISPLAY\_CTRLレジスタの設定

DTM = 1 .....DTモード

DTM = 0 .....メモリ・プレーン横幅と画面横幅が等しいドット数ですので0を設定し, 描画効率のアップを図ります。

DAD+= 0 0 0 .....+1インクリメント

IN = 1 .....インタレース表示

RE = 1 .....リフレッシュ実行

SC = 0 .....SCLK  $\neq$  CLK

MASK= 1 .....通常のVSYNC出力

M/S = 1 .....マスタ・モード

## ⑤ HS, HBP, HH, HD, DFPの算出

1 表示サイクル (SCLKで2周期分) がHS, HBP, HH, HD, HFPの設定単位となります。

1 サイクルは,

$$t_c = 2/6 \times 10^6 \text{ [Hz]} = 0.333 \text{ [\mu s]}$$

ですので以下のようになります。

$$\frac{t_{HS}}{t_c} = \frac{1.76 \text{ [\mu s]}}{0.333 \text{ [\mu s]}} = 5.28 \doteq 5 \quad \text{HS} \rightarrow 4 \text{ (H) (設定値)}$$

$$\frac{t_{HBP}}{t_c} = \frac{2.93 \text{ [\mu s]}}{0.333 \text{ [\mu s]}} = 8.79 \doteq 9 \quad \text{HBP} \rightarrow 8 \text{ (H)}$$

$$\frac{t_{HD}}{t_c} = \frac{23.41 \text{ [\mu s]}}{0.333 \text{ [\mu s]}} = 70.23 \doteq 70 \quad \text{HD} \rightarrow 69 \rightarrow 45 \text{ (H)}$$

$$\frac{t_{HFP}}{t_c} = \frac{2.34 \text{ [\mu s]}}{0.333 \text{ [\mu s]}} = 7.02 \doteq 7 \quad \text{HBP} \rightarrow 6 \text{ (H)}$$

$$\text{HH} = \frac{(5+9+70+7)}{2} - (5+9) = 31.5 \doteq 31 \quad \text{HH} \rightarrow 30 \rightarrow \text{IE (H)}$$

## ⑥ VS, VBP, L/F, VFPの算出

HSYNC1周期が設定単位となります。

$$t_L = 30.45 \text{ [\mu s]}$$

$$\frac{t_{VS}}{t_L} = \frac{0.152 \text{ [ms]}}{30.45 \text{ [\mu s]}} = 4.99 \doteq 5 \quad \text{VS} \rightarrow 5 \text{ (H)}$$

$$\frac{t_{VBP}}{t_L} = \frac{0.685 \text{ [ms]}}{30.45 \text{ [\mu s]}} = 22.00 \doteq 22 \quad \text{VBP} \rightarrow 22 \rightarrow 16 \text{ (H)}$$

$$\frac{t_{L/F}}{t_L} = \frac{11.42 \text{ [ms]}}{30.45 \text{ [\mu s]}} = 375.04 \doteq 375 \quad \text{L/F} \rightarrow 375 \rightarrow 177 \text{ (H)}$$

$$\frac{t_{VFP}}{t_L} = \frac{0.244 \text{ [ms]}}{30.45 \text{ [\mu s]}} = 8.01 \doteq 8 \quad \text{VFP} \rightarrow 8 \text{ (H)}$$

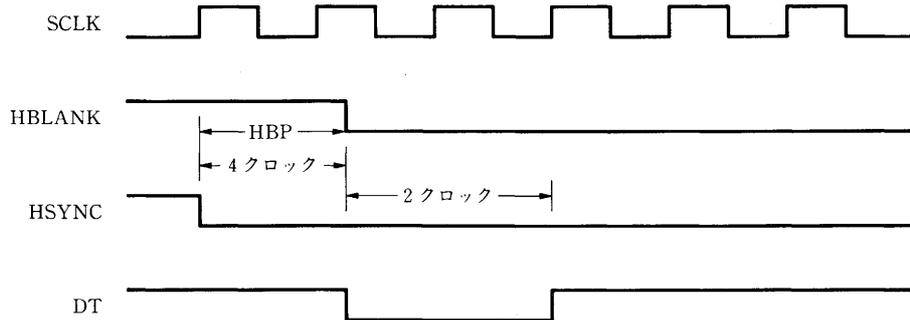
## Q5.29

内容は？

機能・動作
 特性
 応用

表示開始の設定について

- (1) CRT同期信号設定レジスタ (HS, HBS, HH, HFP) の設定を4クロック以上にするのはなぜ？
- (2) DTモード時、BLANK信号の立ち上がりでDT信号が発生して表示サイクルになるため、CRTが表示を開始するのは、HSYNC信号の立ち上がりから6クロック以上となるのでは？



## A5.29

- (1) AGDC内部の表示アドレス・レジスタの値を正常にインクリメントするために4SCLK必要となります。したがって、AGDCの表示機能を使用しないのであれば(描画プロセッサとしてのみ使用), 2SCLK以上で問題ありません。
- (2) そのとおりです。  
したがって、AGDCのHBLANKをクロックで同期をとり3~4SCLK程度遅らせ、 $\mu$ PD42264より出力されるドット・データ(このデータは一度シフト・レジスタで受け、このあとシリアル・データとされる場合が多い)をマスクしてください。

**備考** 通常は、HS, VS, BLANK信号等は、3から4SCLK分遅らせてCRTに供給されています。

## Q5.30

内容は？

機能・動作
 特性
 応用

高解像度のCRTにも対応できる？

## A5.30

対応できます。

ただし、HFP, VFPの期間についてはCRT仕様を詳細に検討してください。

## Q5.31

内容は？

■機能・動作 □特性 ■応用

LCDの表示制御はできる？

## A5.31

AGDCの表示プロセッサは、LCD対応ができません。したがって、HBP、HFPには最小値を設定し、疑似的にHBP、HFPの部分を作成してください。そして、HSYNC、VSYNCを外部のLCDコントローラに入力してください。

## Q5.32

内容は？

■機能・動作 ■特性 □応用

表示画面の設定（プレーン・モード）において、表示アドレス変更のみでスクロールはできる？

## A5.32

横方向スクロールは、16ドット単位であれば表示アドレス変更のみで行えます。

縦方向スクロールは、表示アドレス変更のみで行えます。

ハードウェアでは、垂直方向には1ドットずつスクロールできます。ただし、水平方向にはAGDC単体ではできません（ソフトウェア的には、コピーを行うことで実現できます）。

## [垂直方向にスクロールする場合]

先頭の表示アドレスを変更することで実現できます。

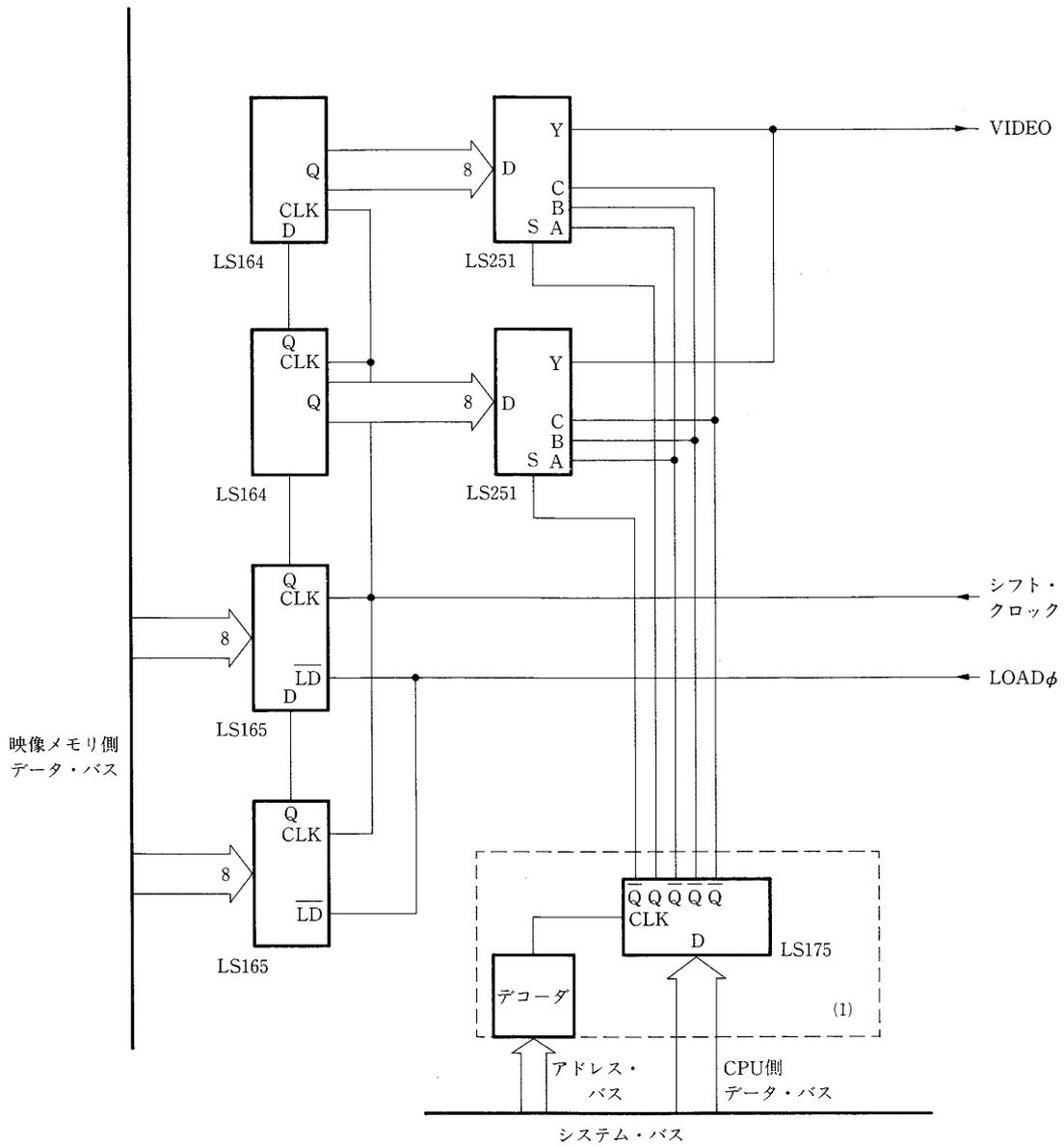
メモリが1画面分しかないときは、スクロールして黒くなる場所に1ライン・コピーをします。

## [水平方向にスクロールする場合]

外部回路が必要です。

外部回路は、ドット・シフト量を記憶する4ビットのフリップフロップとその内容に従って、ドット遅延を生じさせる16ビットのシフトレジスタの信号出力端子を選択する1/8セクタ2個によって構成される回路です（「図5-12」を参照）。

図5-12 水平方向ドット単位スクロール用回路例



注意 この回路はシステム・バス上にあるCPUが直接制御します。

**Q5.33**

内容は？

■機能・動作 特性 応用

AGDCの水平スクロールは、サイクル・スチール・モードでも使用できる？

**A5.33**

サイクル・スチール・モードでも使用は可能です。

ただし、SYNCパラメータのHDの値とWCの値を実際の設定値よりも+1多く設定します。実際の表示画面より大きな領域になっています。したがって、BLANK信号より、遅延信号を作成して実際の表示画面に合わせ込んでください。

★

**Q5.34**

内容は？ 機能・動作 特性 応用

SYNCパラメータ (HS, HBP, HH, HD, HFP, VS, VBP, L/F, VFP) の設定条件は？

**A5.34**

以下にSYNCパラメータの設定条件を示します。

SYNCパラメータ		インタレース・モード		ノン・インタレース・モード	
		スレーブ・モード	マスタ・モード	スレーブ・モード	マスタ・モード
HS	設定値	0004H << FFFFH	0001H << FFFFH	0004H << FFFFH	
	同期サイクル数	5 << 65535	2 << 65535	5 << 65535	
	SCLKクロック数	10 << 13170	4 << 13170	10 << 13170	
HBP	設定値	0002H << FFFFH		0001H << FFFFH	
	同期サイクル数	3 << 65535		2 << 65535	
	SCLKクロック数	6 << 131070		4 << 1310.70	
HH	設定値			0001H << (HD設定値) - 1	
	同期サイクル数	$\frac{-(HS+1) - (HBP+1) + (HD+1) + (HFP+1)}{2}$		2	
	SCLKクロック数			4	
HD	設定値				
	同期サイクル数 SCLKクロック数	0002H << FFFFH 3 << 65535 4 << 131070			
HFP	設定値				
	同期サイクル数 SCLKクロック数	0001H << FFFFH 1 << 65535			
VS					
VBP	設定値	0001H << FFFFH			
L/F	走査線数	1 << 65535			
VFP					

注意 VS, VBP, L/F, VFPでは、設定値0000Hのとき走査線数は65536ラインです。

## Q5.35

内容は？

機能・動作
 特性
 応用

スレープ・モードのとき、HSパラメータの設定値を0003H以下にしたらどうなる？

★

## A5.35

表示することができません。

スレープ・モードで動作するAGDCはEXVS入力信号(DISP\_CTRLレジスタ内SVSフラグ=1)、あるいはEXHS入力信号(DISP\_CTRLレジスタ内SVSフラグ=0のとき)がインアクティブからアクティブに変化したことをサンプリングしたSCLKの立ち上がりから2クロック後のSCLKの立ち上がりでAGDC内部のHC(水平カウンタ)を“4”に初期化します。そして1同期サイクル(2SCLK)ごとにインクリメントし、HCと各同期パラメータ(HS, HBP, HH, HD, HFP)を比較して合致した場合に、IHSと内部HBLANK信号をアクティブからインアクティブへ、インアクティブからアクティブに変化させます。したがって、HSを“3”以下に設定するとHCとHSの設定値を比較しても合致することはなくなり、IHS, HBLANKがアクティブになり続け表示をすることができません。

## Q5.36

内容は？

機能・動作
 特性
 応用

ノン・インタレース・モードのとき、SYNCレジスタのHHパラメータはどのように設定したらいい？

★

## A5.36

HHパラメータはインタレース・モードのみ意味を持つパラメータで、ノン・インタレース・モードでは動作に影響しないダミー・パラメータとなります。したがって、設定範囲内の任意の値(推奨値:0001H)を設定することができます。ただし、省略することはできません。

## Q5.37

内容は？

機能・動作
 特性
 応用

グラフィック・カーサを表示する場合、カーサの形状や大きさはどのようにして決めるのか？

★

## A5.37

AGDCはグラフィック・カーサをハードウェアで構成する場合、GCSRX, GCSRY, GCSRYSレジスタにより、カーソルの位置を制御するタイミング信号GCSRを出力するにすぎません。よってグラフィック・カーサを実現する場合、各R, G, B信号とマスクをする外付け回路を作成してください。

★

## Q5.38

内容は？

機能・動作特性応用

NTSCコンポジット信号とAGDCの表示画面をスーパーインポーズしてテレビ画面に表示させる方法は？

## A5.38

NTSCコンポジット信号をマスタとし、AGDCをスレーブ・デバイスとして外部同期をとることによって実現できます。したがって、NTSCコンポジット信号をY/C分離回路でクロマ信号R、G、Bと水平、垂直同期信号HD、VDに分離してください。

ドット・クロック生成は、分離したHDとAGDCの水平同期信号（IHS）からPLLを使用してドット・クロックを作成します。一般的にドット・クロックを作成するためにはPLL（Phase Locked Loop）回路を使用します。水平方向の同期合わせはPLLで制御します。

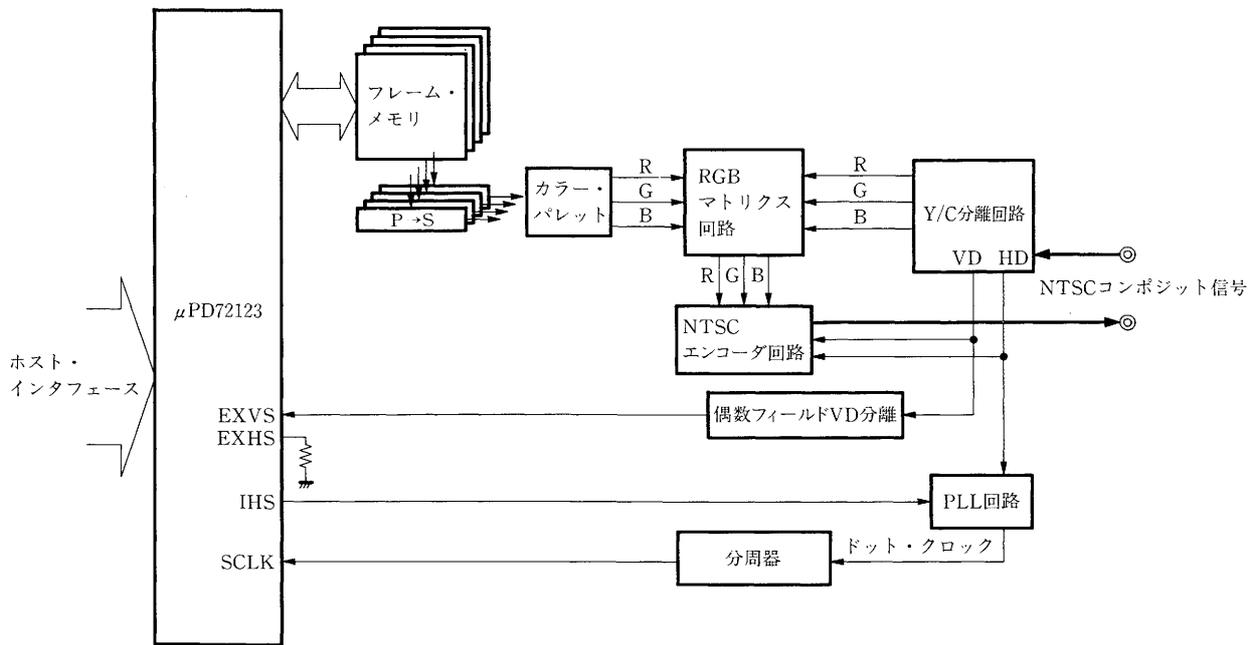
垂直同期合わせはNTSCコンポジット信号から分離されたVDをAGDCのVS/EXVS端子に供給します。ただし、この場合テレビ画面はインタレース表示なので、AGDCもインタレース表示(DISP\_CTRLレジスタ内のINフラグを“1”)に設定し、AGDCに供給するVDは、偶数フィールドのみのVDを供給してください。

NTSCコンポジット信号から分離されたR、G、Bクロマ信号はAGDCが表示するR、G、B信号とRGBマトリクス回路で合成してください。これを実現するにはアナログ・スイッチ（74HC4066など）を使用します。

NTSCエンコーダ回路は、合成したクロマ信号とNTSC信号からのHD、VDからNTSCコンポジット信号に変換して出力します。

図5-13にNTSCコンポジット信号とのインタフェース・ブロック図を示します。

図5-13 NTSCコンポジット信号インタフェース・ブロック図



NTSC基準のうち、垂直関係、等価パルス、カラー関係を除いた各項目は次のとおりです。

走査線数/フレーム	525本
フィールド周波数	60 Hz
水平周波数	15734.26 Hz
公称周期	63.5555 $\mu$ s
水平ブランキング期間	10.5~11.4 $\mu$ s
水平同期パルス幅	4.19~5.08 $\mu$ s
水平フロントポーチ HFP	1.27~2.22 $\mu$ s
水平バックポーチ+水平パルス幅	9.2~10.3 $\mu$ s

ドット・クロック周波数を決定するには表示画面サイズが720ドット×480ドットの場合、次のようになります。

$$\text{水平表示期間} = 63.5555 \mu\text{s} - 11.4 \mu\text{s} = 52.1555 \mu\text{s}$$

$$f_{\text{dot}} = \frac{720 \text{ dots}}{52.1555 \mu\text{s}} = 13.80 \text{ MHz}$$

$f_{\text{dot}}$ : ドット・クロック周波数

このドット・クロックを分周することにより表示用クロック (SCLK) を決定し、NTSC基準にあったSYNCパラメータをAGDCに設定してください。設定の際にはSYNCパラメータ設定条件 (特に最小値) に注意してください。

## 第6章 描画制御

### Q6.1

内容は？

機能・動作 特性 応用

使用する予定のないレジスタの設定はしなくてもいい？

### A6.1

画面構成に必要なレジスタ (EADORG, PITCHD, PMAX, PDISPD, CLIPなど) はパラメータを必ず設定してください。例えば、クリッピングを使用しないからと言って、CLIPレジスタに何も設定しないということは避けてください。また、描画関係レジスタの場合は、あるコマンドを発行する際に、そのコマンドで使用しないレジスタの設定をする必要はありません。

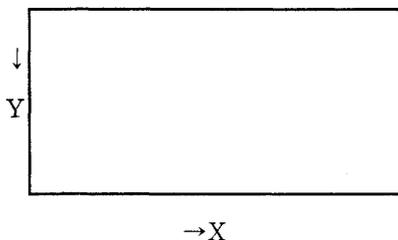
### Q6.2

内容は？

機能・動作 特性 応用

EADORG, dADORG (座標原点) を画面左上にセットしたとき、Y座標の設定は？

EADORG, dADORG



### A6.2

座標原点に対して、負の座標設定時には2の補数を用いてください。

このときの設定値に対する定義域は、次のとおりです。

$$-32768 \leq X, Y \leq 32767$$

(8000H)                      (7FFFH)

たとえば、Y方向に10Hであれば、7FF0Hの設定となります。

## Q6.3

内容は？

機能・動作
 特性
 応用

描画領域と表示領域を独立に設定できる？

## A6.3

描画領域は

EADORG, dADORG, PDISPD, PITCHD

表示領域は

DISPLAY\_PITCH, DAD, WC

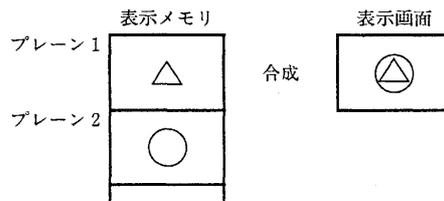
で定義されるため独立に設定できます。

## Q6.4

内容は？

機能・動作
 特性
 応用

下記のようなプレーンの合成表示はできる？



## A6.4

プレーン1をソース・データ、プレーン2をデスティネーション・データとし、ソースのデータをデスティネーションにコピーすれば、このような表示ができます。

## Q6.5

内容は？

機能・動作
 特性
 応用

AGDCに対し、インバリッド・コマンドを発行するとAGDCはどうなる？また、このときの状態を知る方法は？

## A6.5

AGDCに対してインバリッド・コマンドを発行した場合、AGDCの動作は保証できません。また、このときの状態を知る方法もありません。インバリッド・コマンドは発行しないようにしてください。

## Q6.6

内容は？

■機能・動作 特性 応用

直線描画を行ったときに、穴空きやゴミが表示されてしまう。

表示メモリにもそのようなデータが書き込まれているが、考えられる原因は？

## A6.6

データの書き込まれるアドレスをラッチするタイミングが遅く、不定のアドレスをラッチしてしまい、別のアドレスに描画していると思われます。

描画サイクルのアドレス・ラッチに関する規格が守られているか、確認してください。

## Q6.7

内容は？

■機能・動作 特性 応用

表画面と裏画面の大きさを変えられる？

## A6.7

表画面と裏画面の大きさを変えても、支障なく使用できます。

ただし、PITCHSとPITCHDの値に注意してください。

## Q6.8

内容は？

■機能・動作 特性 応用

書き込みを行うプレーン数は、PMAXレジスタだけで制限される？

## A6.8

制限されます。

AGDCはプレーン間の描画をPDISP (Plane Displacement: プレーン間変位) を用いて各プレーンの描画アドレスを算出します。このときの加算回数がPMAXとして定義されています。

## Q6.9

内容は？

機能・動作 特性 応用

コマンド・パラメータとして規定外の値を設定してもハングアップしない？

## A6.9

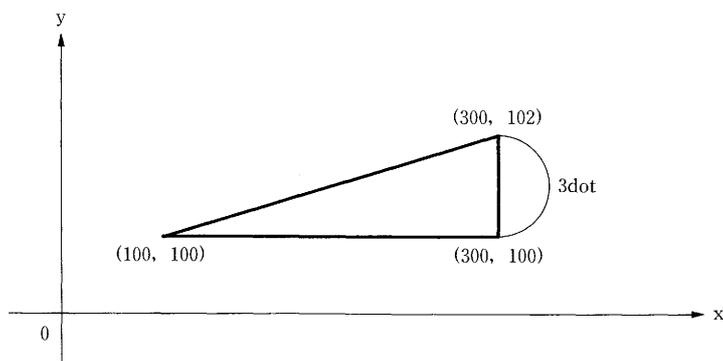
コマンド・コードに規定外の値を設定するとハングアップする場合があります。また、PAINTコマンドのみは、パラメータ値によってもハングアップすることがあります。

## Q6.10

内容は？

機能・動作 特性 応用

下図のような三角形の塗りつぶしを行う場合に、短辺 (3dot) をy軸に平行すると描画されない部分が発生してしまうのはどうして？

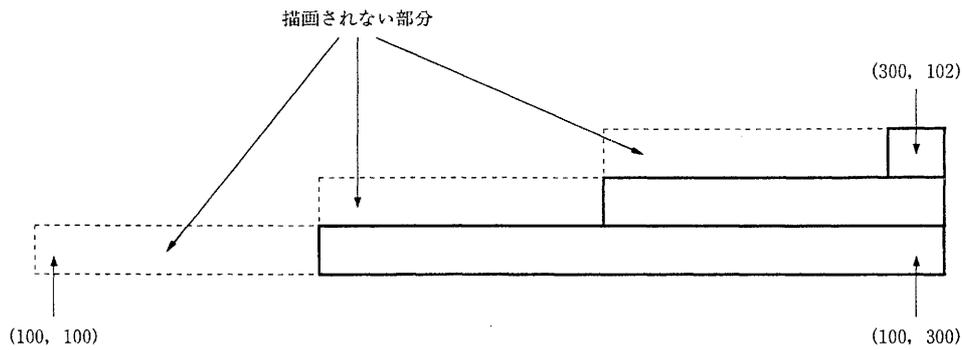


## A6.10

AGDCでは、三角形塗りつぶし (TRI\_FILL) の場合、オペレーション・フラグのWL, WRフラグで斜辺の塗りつぶしの制御を行います。

左右の斜辺の傾きがx軸に対して45°未満の場合には、これらのフラグで制御できるピクセルとできないピクセルが存在してしまいます (『μPD72123のユーザーズ・マニュアル「付録H ピクセル単位の互換性」』を参照してください)。

したがって、上記のような三角形の塗りつぶしの場合には、斜辺にあたる部分が描画されませんので、うまく塗りつぶしが実行されていないように見えます。



## Q6.11

内容は？

機能・動作
 特性
 応用

横方向に1ドットずつスクロールする方法は？

## A6.11

横方向にドット単位のスクロールを実現するために、コピー・コマンドを使用します。そのとき、スクロールの方向（右か左か）に注意してください。方向によっては、元のデータを壊してしまう可能性があります。それを防止するために、COPYコマンドのESE, ROTフラグで制御して、コピーを行ってください（左にスクロールする場合は、ESE=0, ROT=0で、右にスクロールする場合は、ESE=1, ROT=1で実行）。

備考  $\mu$ PD72123のユーザーズ・マニュアル「10.8.5 実アドレスまたは座標の設定 (C)オペレーション・フラグ (ESE, ROT) の操作」を参照してください。

## Q6.12

内容は？

機能・動作
 特性
 応用

拡大/縮小を任意の倍率で行う方法は？

## A6.12

AGDCでは、拡大の場合 $16/n$ 倍、縮小の場合 $n/16$ 倍 ( $n=1-16$ ) のおのおの16通り選ぶことができます。任意の拡大/縮小は、この $16/n$ と $n/16$ の組み合わせを行うしかありません。画像劣化を緩和するには、拡大してから縮小してください。

## Q6.13

内容は？

機能・動作
 特性
 応用

拡大／縮小コピーについて変倍後のサイズの算出方法は？

## A6.13

計算式は以下のようになります。

デスティネーション横ドット数 =  $\{(DH+1) \times \text{倍率}\}$

デスティネーション縦ドット数 =  $\{(DV+1) \times \text{倍率}\}$

備考 { } は四捨五入を表します。

- 例1. 15ドットを1/16倍するとき： $(15 \times 1) / 16$ を四捨五入します。  
 2. 5ドットを7/16倍するとき： $(5 \times 7) / 16$ を四捨五入します。  
 3. 13ドットを16/3倍するとき： $(13 \times 16) / 3$ を四捨五入します。

## Q6.14

内容は？

機能・動作
 特性
 応用

文字フォントの拡大+傾斜を行いたいが、処理速度、ジャギーなどを考慮すると、拡大コピーと傾斜コピーのどちらを先に実行すべき？

## A6.14

拡大コピー→傾斜コピーの方がジャギーは軽減されます。ただし、処理速度は変わりません。

また、拡大+傾斜であれば任意角回転&拡大／縮小コピーでパラメータを適切に設定すれば1コマンドで実行できます。ただし、1ドットごとの描画となるため、処理速度は遅くなります。

## Q6.15

内容は？

機能・動作 特性 応用

4点ORによる1/2縮小の方法は？

## A6.15

AGDCの縮小方法は、単純間引きですが、次のような方法で4点ORによる1/2縮小ができます。原理的には1/2縮小コマンドを4回発行することです。最初のコマンドのみ論理演算モードを“REPLACE”で行い、残り3回は“OR”で行います。絶対座標指定の縮小コピーでは、次のようにレジスタに設定します。

- ・ソース・データのリード開始点                    : (XS, YS)
- ・デスティネーション・データのライト開始点: (XD, YD)
- ・ソース側コピー領域                                 : (DH, DV)

また、縮小率1/2をMAGH (=7), MAGV (=7) におおの設定します。これで縮小を行います。ただし、このとき、ソース・データのリード開始点 (XS, YS) のドット情報は必ず描画します。

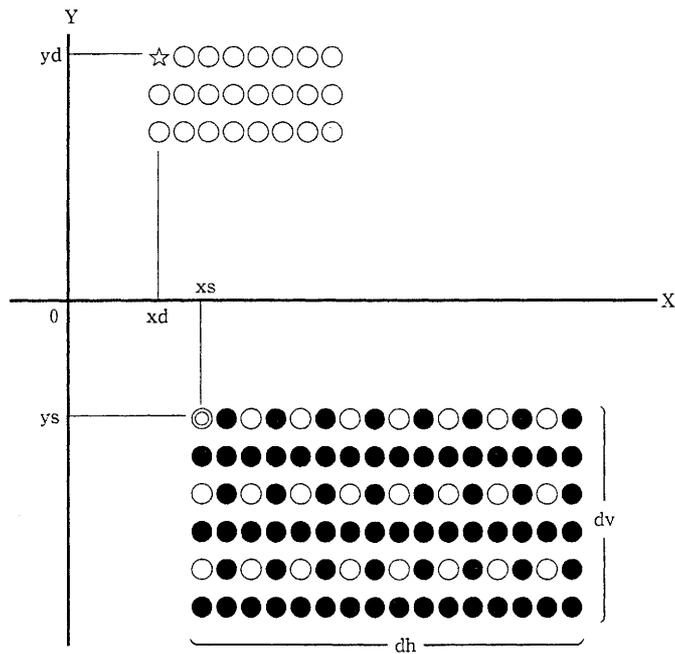
つまり、(XS+2m, YS-2n) の座標位置に存在するデータは間引かれません。

ただし、m=0, 1, 2, ……., DH/2

n=0, 1, 2, ……., DV/2

これはソース・データのリード開始点… (XS, YS) がソース矩形領域の左上にある場合 (コピー・コマンド内オペレーション・フラグのESE=0) です。

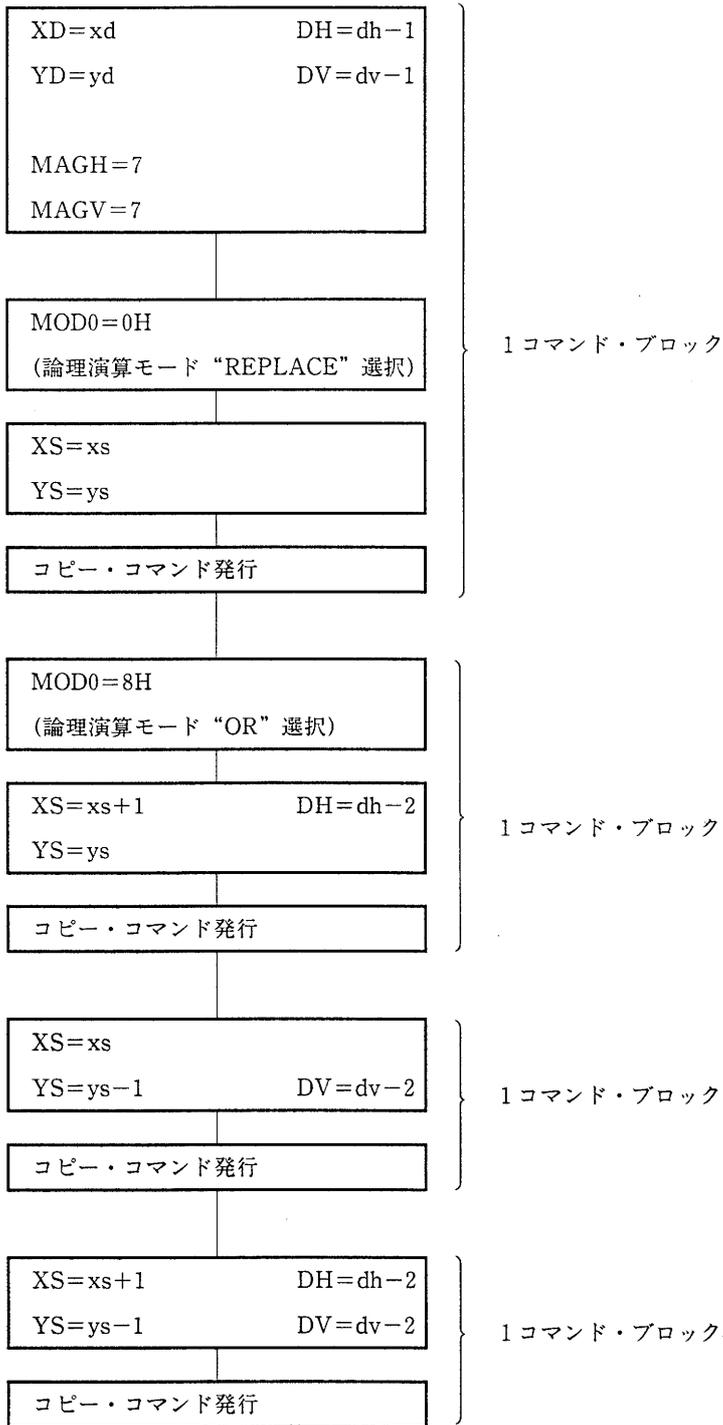
この動作を利用して次のように4点ORを実現します。次ページに、具体例 (フロー・チャート) を示します。



- ◎…ソース・データのリード開始点
- ☆…デスティネーション・データのライト開始点
- …ソース・データの間引き点

誤解を招かないように、パラメータ値を小文字で、レジスタ名を大文字で説明します。PLANESレジスタは全ビット“0”になっており、MOD0レジスタで設定されている論理演算モードが選択されると仮定します。

★



## Q6.16

内容は？

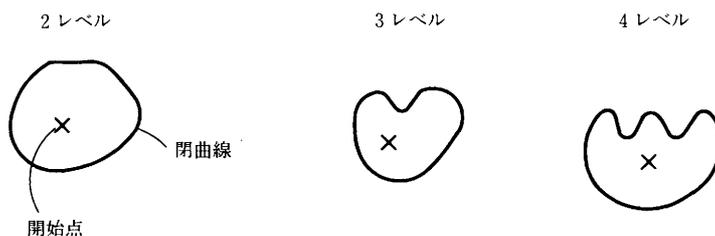
機能・動作特性応用

スタック領域について

- (1) 文字のアウトライン内を塗りつぶすには、どのくらい必要？
- (2) PAINTコマンドを実行するには、どのくらい必要？
- (3) 塗りつぶしパターンによって、スタックの消費量は変化する？
- (4) プレーン数を増やすと、スタック領域も広げる必要がある？

## A6.16

- (1) スタック領域をどれだけ確保すればよいかは、明確な定義はできません。目安として、1120×750の画面いっぱいに24×24ドットの文字を40文字×25文字並べてバック・グラウンドをPAINTすると1000レベル程度のスタック領域の消費量となります。
- (2) スタック領域をどれだけ確保すればよいかは、明確な定義はできません。スタック領域は塗りつぶす対象となる閉領域の形状に依存します。  
たとえば、円、長方形、台形はそれぞれ2レベル必要となります(1レベルは、6ワード=6×16ビットです)。また、円が下図のように変形された場合、大略すれば、1箇所凹が増すごとにスタック領域は1レベルずつ増加します。ただし、開始点位置や、へこみぐあい等にも依存してスタック領域は増減します。  
なお、スタック領域をオーバーした場合、復帰→塗りつぶしの継続は行われませんので、スタック領域は十分余裕を持つことをお勧めします。



- (3) スタックの消費量は塗りつぶしパターンに依存しません。
- (4) 必要ありません。スタックの消費量はプレーン数に依存しません。

## Q6.17

内容は？

機能・動作特性応用

直線描画について

- (1) 始点と終点が同一座標のとき、描画を行う？もし行わないなら、レジスタの変化は通常と同じ？
- (2) そのとき、線幅を拡張しているとうなる？

## A6.17

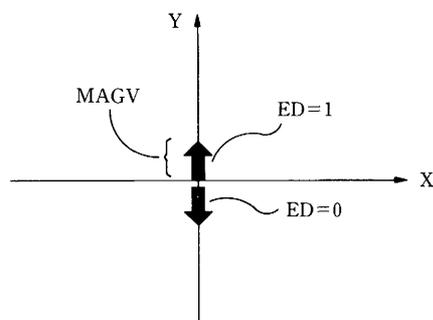
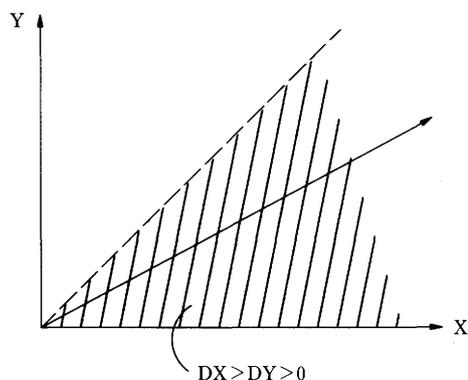
- (1) AGDCは、次に描画すべき点が終点かどうかを判断してから描画を行いますので、WEPフラグにより以下の動作となります。

{ WEPフラグ=1であれば点が描画されます。  
 { WEPフラグ=0であれば描画されません。

また、レジスタの値は通常描画と同様に変化します。

- (2) 線幅が拡大されている場合は、Y軸方向に太くなります。

長さが1ドットの直線はすべて左図の斜線領域の方向ベクトルを持っているものとして処理されます。したがって太い直線を描画した場合、Y軸方向に太くなります。その方向は、右図に示すようにEDフラグによって決定されます。



## Q6.18

内容は？

■機能・動作 特性 応用

ライン描画コマンドのパラメータ設定で

- (1) A\_LINE\_D0からA\_LINE\_D2で「(X#, Y#) = (X, Y) であること」とあるが、(X#, Y#) ≠ (X, Y) のとき、どうなる？
- (2) R\_LINE\_D0からR\_LINE\_D2では、(X#, Y#) = (X, Y) でなければならない？

## A6.18

- (1) 正常に描画できません。
- (2) そのとおりです。  
必ず (X#, Y#) = (X, Y) にしてください。

## Q6.19

内容は？

■機能・動作 特性 応用

直線描画コマンドでプリプロセッサ・エラーは発生する？

## A6.19

A\_LINE\_M0で、プリプロセッサ・エラーは発生しません（AGDCの内部には、そのような要因はありません）。

エラーが起きた場合には、枠内の点に注意し、A\_LINE\_M0に対するプリプロセッサ・エラーかどうかを確認してください。

A\_LINE\_M0コマンドのプリプロセッサ処理が正常に実行されると、(x, y)に設定した座標を、実アドレスに変換したビット位置が、EAD1, dAD1レジスタにセットされている。

**Q6.20**

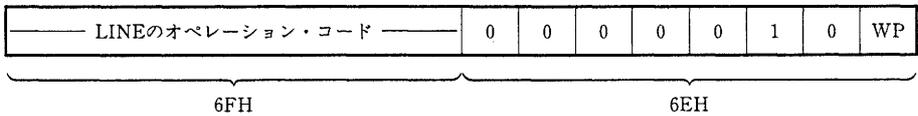
内容は？	<input type="checkbox"/> 機能・動作 <input type="checkbox"/> 特性 <input checked="" type="checkbox"/> 応用
------	---------------------------------------------------------------------------------------------------

32ビット線種パターンを選択方法は？

**A6.20**

LINE命令のオペレーション・フラグ部を下図のように設定してください。ビット2のPLフラグを1にすることにより32ビットの線種パターンを選択できます。32ビットの線種パターンはPTNCNTレジスタ、およびDHレジスタを用います。

また、32ビット線種の場合は、IPフラグ、ESフラグはありません。IP=1, ES=0の動作を行います。



**Q6.21**

内容は？	<input type="checkbox"/> 機能・動作 <input type="checkbox"/> 特性 <input checked="" type="checkbox"/> 応用
------	---------------------------------------------------------------------------------------------------

円、楕円描画時の線幅拡大方法は？

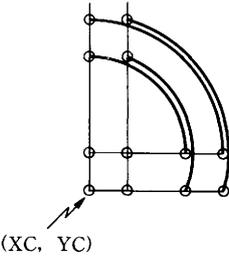
**A6.21**

μPD72120では、円、楕円、描画時に線幅を1回のコマンドにより拡大することはできません。ただし、コマンドを数回行うことにより線幅を拡大することができます。

以下に線幅を拡大する一例を説明します。

2ビット幅の弧を描画する場合は (XC, YC), (XC+1, YC), (XC, YC+1), (XC+1, YC+1) の4点を設けこの点を中心として半径DXの弧を4回描画します。3ビットの線幅の場合は同様に9ビットの中心に対して、9回弧を描画します。

μPD72123では、グラフィクス・ペンの機能が使えます。



## Q6.22

内容は？

機能・動作 特性 応用

太線（8ビット、16ビットなど）で直線以外の線図形（円、楕円など）を描画する方法は？

## A6.22

方法としては、ペン先のビット・パターンをメモリに定義しておき、コピー・コマンドで1ポイントずつ描画する方法があります。

μPD72123では、グラフィクス・ペンの機能が使えます。

## Q6.23

内容は？

機能・動作 特性 応用

1点から3方向に直線を描画する場合、A\_LINE\_M1(D1)では命令実行後の描画ポイント(X#, Y#)が描画終了座標となり、1回の直線描画で(X, Y) ≠ (X#, Y#)のため、描画不可能と思えるけど、どうすればいい？

## A6.23

A\_LINE\_M1コマンドの場合は描画できます。

A\_LINE\_M1コマンドを発行すると、プリプロセッサは、X, Yレジスタ値を描画プロセッサのX#, Y#レジスタに転送します。

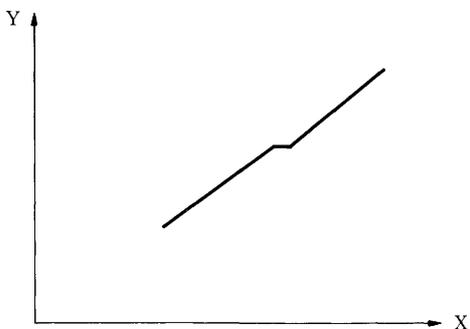
したがって(X#, Y#)がコマンド発行前にどこに位置していても、X, Yレジスタ値がA点にあれば3方向に直線を描画するのは可能です。

ただし、A\_LINE\_D1コマンドはご質問のとおりです。

**Q6.24**

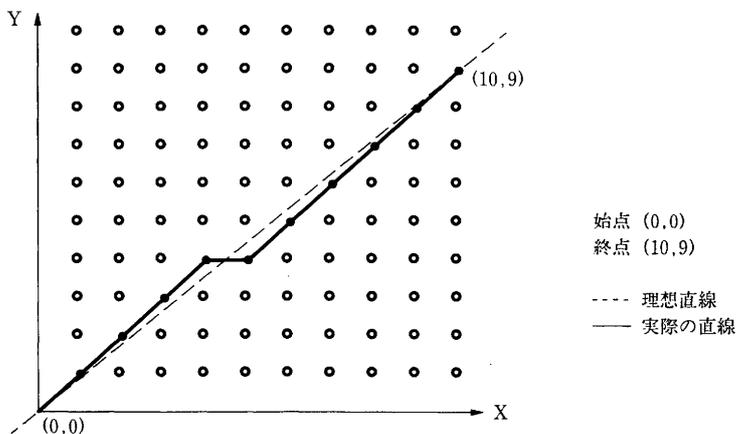
内容は? 機能・動作 特性 応用

右上がりの直線をコマンドA\_LINE\_Mで描画した場合、直線中央部に段差ができることがあるのはなぜ?



**A6.24**

傾きが45°に近い場合に起こります。たとえば、始点が(0,0)で終点が(10,9)のような場合AGDCは、理想直線に近い点(ビット・マップ・メモリ)を描画していきます。このため、図のように直線の中央付近に段差が生じます。ただし、この現象は直線アルゴリズム上生じるものでAGDC固有の問題ではありません。



## Q6.25

内容は？

■機能・動作 □特性 □応用

★

直線は最大何ドットまで描画できるのか？

## A6.25

最大32768 (8000H) ドットまでの直線を描画できます。次に示す絶対座標指定の直線描画コマンドのときは特に注意してください。

A\_LINE\_M0

A\_LINE\_M1

A\_LINE\_M2

A\_LINE\_D0

A\_LINE\_D1

A\_LINE\_D2

A\_LINE\_D3

また、AGDC内部でこれらのコマンドを使用している次のコマンドの場合も同様です。

A\_REC

A\_REC\_FILL\_C

TRI\_FILL

A\_TRA\_FILL

このようなコマンドを使用する場合はソフトウェアでドット数判別ルーチンを設けることをお奨めします。なお、相対座標指定の直線描画コマンドは描画終了点を描画開始座標からの変位量で示し、DX, DYは2の補数で設定するので32769ドット以上の直線描画を実行することはありません。

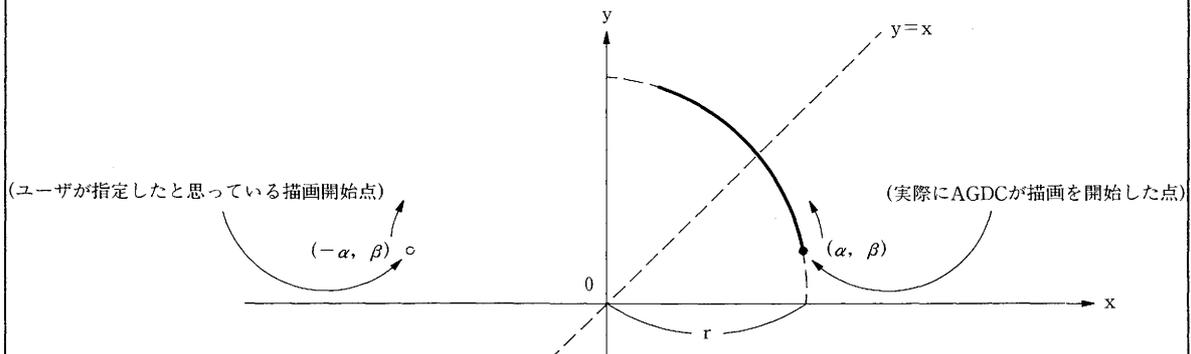
## Q6.26

内容は？

■機能・動作 □特性 □応用

円弧描画について、

- (1)  $(x_s, y_s) = (-\alpha, \beta)$  という座標を指定したにもかかわらず、AGDCが  $(\alpha, \beta)$  という座標より描画を開始することがある？
- (2) 半径  $r$  であるような円弧描画を行うときその円弧内に描画開始座標  $(x_s, y_s)$  を指定した場合、AGDC はどのような判定を行いながら円弧を描画する？



## A6.26

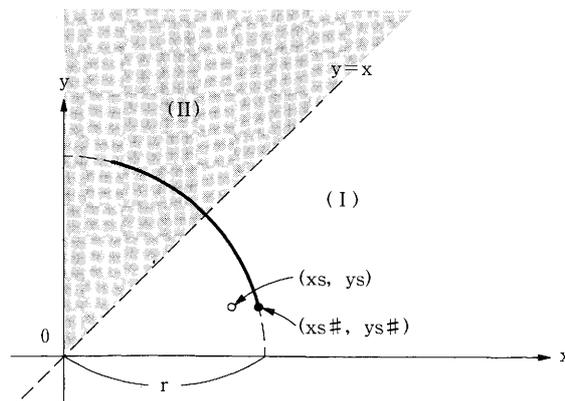
- (1) ありません。

EADORG, dADORGで設定されたところを原点として論理上のX-Y座標空間を形成し、与えられた座標から実アドレスに変換します。したがって、ほかの象限の点を開始点にはありません。

- (2) 描画開始点の判定は第一象限を例にとると、傾きが45°の直線 ( $y=x$ ) を境界として行っています (下図参照)。

(I)の領域に描画開始点  $(x_s, y_s)$  が与えられた場合は、半径  $r$  と  $y_s$  の値を基にして、実際AGDCが描画を開始する座標  $(x_s\#, y_s\#)$  を求めます。

(II)の領域に描画開始点  $(x_s, y_s)$  が与えられた場合は、半径  $r$  と  $x_s$  の値を基にして、実際AGDCが描画を開始する座標  $(x_s\#, y_s\#)$  を求めます。



## Q6.27

内容は？

■機能・動作 □特性 □応用

クリッピングが使用できないコマンドは？

## A6.27

クリッピングが使用できないコマンドを、次に示します。

- ・ COPY\_AA
- ・ COPY\_CA
- ・ COPY\_AC (高速コピー)
- ・ COPY\_CC (高速コピー)
- ・ PUT\_A
- ・ A\_REC\_FILL\_A
- ・ A\_REC\_FILL\_C (高速塗りつぶし)
- ・ R\_REC\_FILL (高速塗りつぶし)

上記のコピー・コマンドは、単純、傾斜、90°回転、拡大／縮小、任意角回転・拡大／縮小すべてに共通しています。

これらすべてのコピー・コマンドでは、クリッピングを使用できません。

μPD72123では、3オペランド・コピーでも上記コピー・コマンドと同様にクリッピングを使用できません。上記に示すように、デスティネーション描画位置を実アドレスで指定するコマンドに関しては、クリッピング機能を使用しないでください。

また、PAINTコマンドを発行する場合は、クリッピングを設定してから行ってください。

**Q6.28**

内容は？

■機能・動作 □特性 □応用

クリッピングとFASTフラグの関係は？

**A6.28**

FAST=1と指定してコマンドを発行すると、2クロック単位でライト・サイクルを起動します。

この関係で、クリッピング機能を使用することができません。

したがって、オペレーション・フラグにFASTフラグがあるコマンドは、CLIPレジスタを01bとして発行してください。

該当するコマンドを次に示します。

- ・ R\_REC\_FILL\_C
- ・ R\_REC\_FILL
- ・ A\_COPY\_AA
- ・ A\_COPY\_AC
- ・ A\_COPY\_CA
- ・ A\_COPY\_CC

また、次に示すコマンドのデスティネーションの座標を絶対番地で指定するコマンドについてもクリッピング機能が使用できません。

- ・ R\_REC\_FILL\_A
- ・ A\_COPY\_AA
- ・ A\_COPY\_CA
- ・ PUT\_A

備考 「Q6.27」を参照してください。

## Q6.29

内容は？

機能・動作
 特性
 応用

任意閉領域塗りつぶし (PAINT) を実行する場合、クリッピング領域を指定していると、クリッピング境界線の塗りつぶしはどうか？

## A6.29

クリッピング領域として指定するレジスタ (XCLMIN, XCLMAX, YCLMIN, YCLMAX, CLIP) は、PAINT コマンド発行時は境界点検索領域となります。

このときの境界線はクリッピング・レクタングルの内側としてみなされます。

つまり、塗りつぶしがクリッピング・レクタングル内にあり、境界線まで塗りつぶしがあれば境界線は塗りつぶされます。クリッピング・レクタングルのほかの場合は境界点検索領域外になり、PAINT コマンドは発行されません。

注意 PAINT コマンド実行時には CLIP=00 にしてください。

## Q6.30

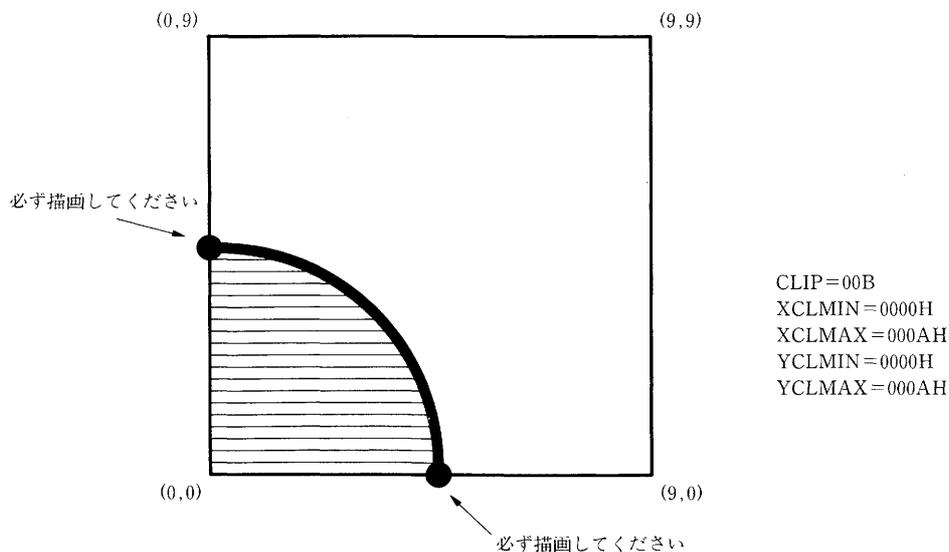
内容は？

機能・動作
 特性
 応用

PAINT コマンドでクリッピング領域と円弧描画コマンドで囲まれた閉領域を塗りつぶすことが可能？

## A6.30

可能です。クリッピング境界線上も境界点検索領域であり、描画可能対象点になるので必ず描画してください。



**Q6.31**

内容は？

■機能・動作 特性 応用

クリッピングの際の描画無効領域に対して、

- (1) 読み出したビット・データをそのまま同じ番地書き戻す？
- (2) 描画ライト・サイクルを起動しないかを外部から指定できる？

**A6.31**

(1), (2)の選択に関しては、AGDCの内部で自動的に判断して行っています。  
したがって、外部から指定するものではありません。

**Q6.32**

内容は？

■機能・動作 特性 応用

AGDCは、メモリが実装されているかどうか意識して描画サイクルを起動する？

**A6.32**

まったく意識せずに描画処理を行います。  
したがって、メモリ未実装エリアをAGDCがアクセスした場合は、パリティ・エラーが発生することが考えられます。

**Q6.33**

内容は？ 機能・動作 特性 応用

タイリング(パターンによる塗りつぶし)にPTNCNTレジスタを使用しないとき、外部定義パターンを表示メモリのどこに設定すればいい？

**A6.33**

PTNPレジスタに、塗りつぶしパターン格納エリアの最低位アドレスを設定してください。

このとき、PTNCNTレジスタは、PTNPレジスタに設定されたアドレスから何ワード目までを塗りつぶしパターンに使用するかを定義します。

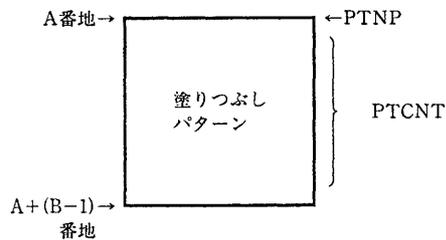
また、Y (Y方向の座標値) に書かれるパターンは、PTNP=A, PTNCNT=Bとすると

$$A + (B - 1) - \{Y \text{ を } B \text{ で 割 っ た も の の 余 り}\}$$

で指定されたデータになります。

A + (B - 1) 番地に指定されたデータがY=0の

ラインに描画されます。



**Q6.34**

内容は？ 機能・動作 特性 応用

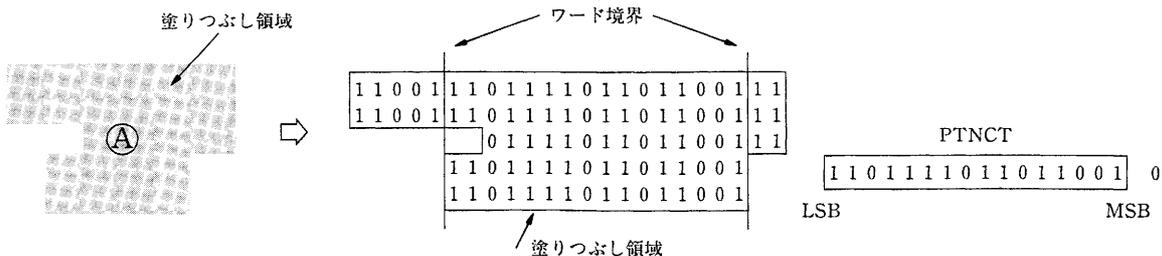
PTNCNTレジスタを塗りつぶしパターンとして使用する場合、パターンの連続性は？

**A6.34**

連続性は保たれます。

AGDC内蔵のレジスタ (PTNCNT) を塗りつぶしパターンとして使用する場合には、パターンは表示メモリのワード境界で与えられます。したがって、垂直方向には同じ値が出てくることになります。

図 6-1 TL=0 時の塗りつぶし実行例 (論理演算はREPLACE)



## Q6.35

内容は？

■機能・動作

□特性

■応用

塗りつぶしパターンを表示メモリ参照方式にした場合

- (1) タイリング・パターンを32×32ドットで展開する方法は？
- (2) どの位置から塗りつぶしても、パターンの連続性は保たれる？

## A6.35

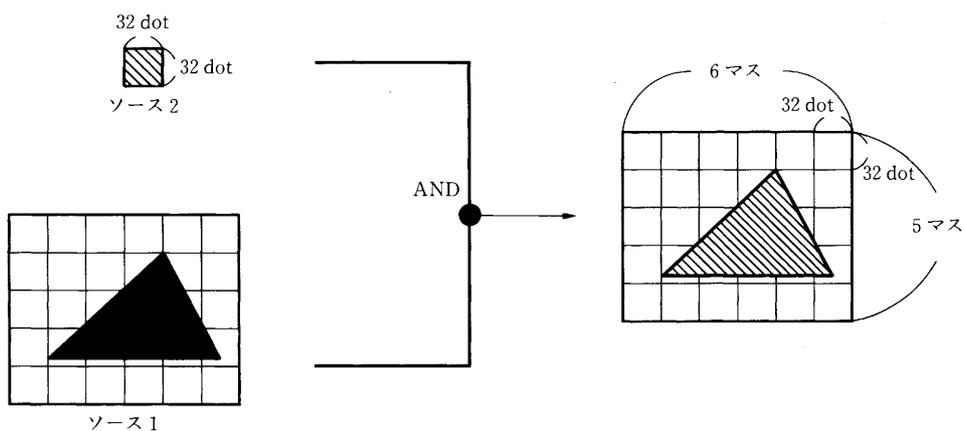
- (1)  $\mu$ PD72120では、直接32×32ドットのタイリング・パターンを扱うことはできません。

一例として表示メモリに余裕がある場合（ワーク用のビット・マップ空間がある場合）以下の方法を参照してください。

32×32ドットのパターンをタイリング・パターンとして使用する場合の一例

- ①塗る対象の図形を単色で塗る。
- ②塗った図形をワーク・プレーンにコピーする。
- ③ワーク・プレーンにコピーした図形をソース1（マスク・パターン）、塗りつぶしパターンをソース2として、マルチソース&シングル・デスティネーション・コピーを実行する。  
このとき論理演算はANDとします。
- ④③をワーク・プレーンに切り取った矩形領域分だけコピーを実行します（図6-2を参照してください）。

図6-2 32×32ドットのパターンをタイリング・パターンとしての使用例  
(COPYを5×6=30実行)



また、 $\mu$ PD72123では、32×32ドットのタイリング・パターンを指定できます。

- (2) 保たれます。ただし、PTNP, PTNCNT, EADORGレジスタ値を変更しないことが前提です。

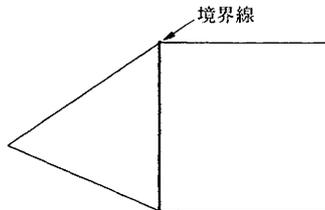
## Q6.36

内容は？

機能・動作特性応用

★

下図のような図形をTRI\_FILLコマンドとA\_REC\_FILLコマンドを用いて描画し、境界線上での塗りつぶしパターンの連続性は保てるか？



## A6.36

塗りつぶしパターンの連続性は保てません。

塗りつぶしパターンの連続性を保つためには直線コマンドでポリゴン描画し、PAINTコマンドで塗りつぶしをしてください。

## Q6.37

内容は？

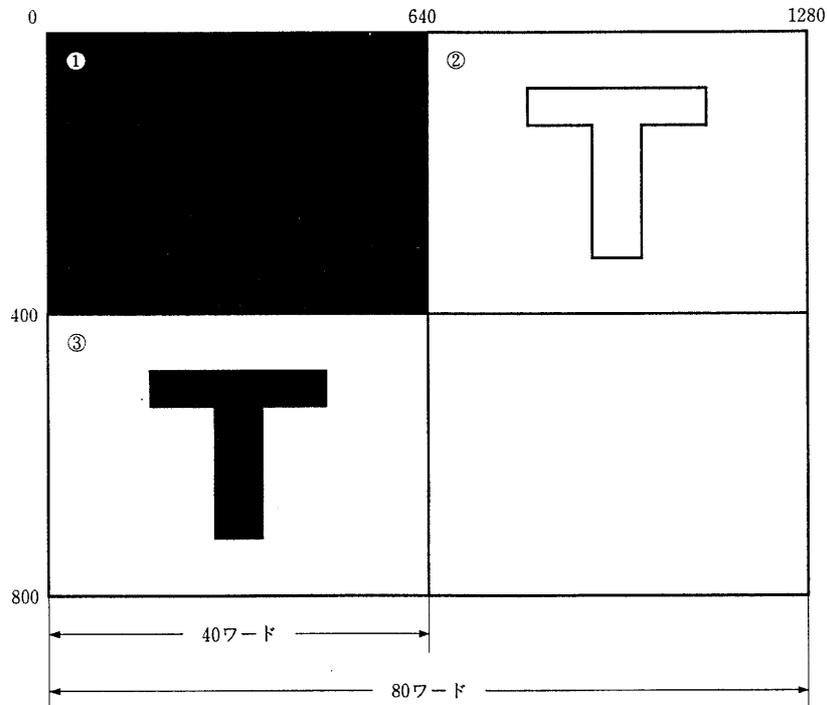
機能・動作
 特性
 応用

3オペランド・コピーの際のマスク領域のメモリ幅（ピッチ）はどのように設定すればいい？

## A6.37

3オペランド・コピーの際のマスク領域のメモリ幅は、PITCHS、またはPITCHDレジスタの値で代用します。どちらのレジスタの値を参照するかは、3オペランド・コピーのオペレーション・フラグのPSELフラグで指定します。

例



- ① マスク領域
- ② ソース領域
- ③ デスティネーション領域

この場合、PITCHSレジスタに40Hを設定してPSEL=1として3OP\_COPYコマンドを発行すると、①のマスク領域のメモリ幅は、PITCHSレジスタの値が参照され、③のデスティネーション領域にコピーされます。

## Q6.38

内容は？

機能・動作
 特性
 応用

コピー・コマンド受け付け後，メモリ・プレーン・アクセスまでのクロック数は？

## A6.38

以下に各コピー・コマンド発行後，メモリ・プレーン・アクセス開始までのクロック数を示します。ただし，コピー自体は，プリプロセッサ，描画プロセッサによる処理であり，CLK入力により実行されます。

COPY\_AA..... $\alpha$  (16 $\leq\alpha\leq$ 31)

COPY\_AC..... $\alpha + PS^{\text{注1}} + PD^{\text{注2}}$  (35 $\leq\alpha\leq$ 50)

COPY\_CA..... $\alpha + PS + PD$  (36 $\leq\alpha\leq$ 51)

COPY\_CC..... $\alpha + PS + PD$  (55 $\leq\alpha\leq$ 70)

注1. PS=PITCHSの有効ビット数

2. PD=PITCHDの有効ビット数

## Q6.39

内容は？

機能・動作特性応用

回転コピーと塗りつぶしについて、次の方法で楕円内部を塗りつぶせる？

- ①線幅1ドットで楕円を描画
- ②45°の回転コピーを実行(境界点検索をする塗りつぶしコマンド)
- ③回転した楕円にPAINTコマンドを実行

## A6.39

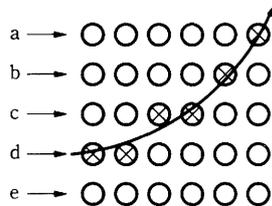
塗りつぶすことはできません。

FR\_ES\_COPYコマンドで、+45°の回転を行った例を下図に示します。

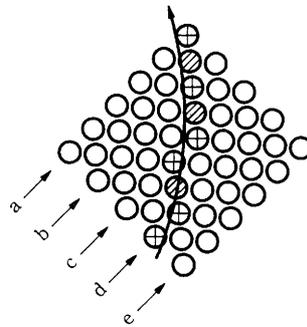
この場合の○部分が塗りつぶされず、線がとぎれることがあります。このため、楕円内を塗りつぶすことができません。

図6-3 45°回転コピーと塗りつぶし

(a) 楕円弧



(b) FR\_ES\_COPYコマンドにより+45°  
回転させた楕円弧



**Q6.40**

内容は? 機能・動作 特性 応用

コピー・コマンドにおいて

- (1) ソース・データの右下を読み出し開始点 (ESE=1) にしたらどうなる?
- (2) 表示メモリの同一アドレスに同一データをコピーできる?
- (3) 表示メモリの同一アドレスに空白データをコピーできる?

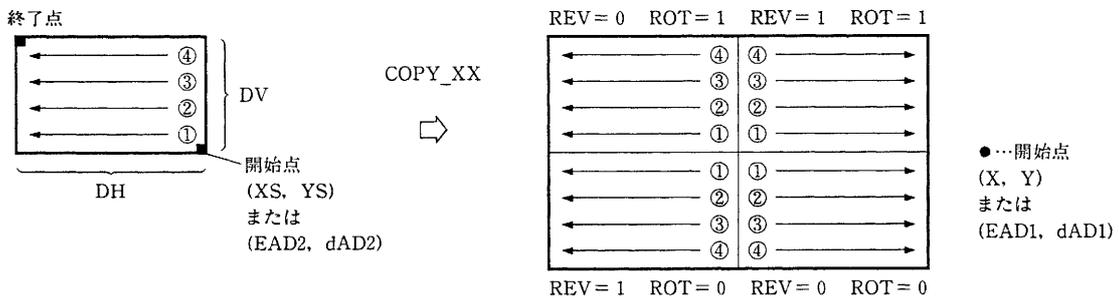
**A6.40**

(1) ESEフラグを1にし、ソース・データの右下を読み出し開始点とした場合、REV, ROTフラグのセットの仕方により転送の方法が異なります。

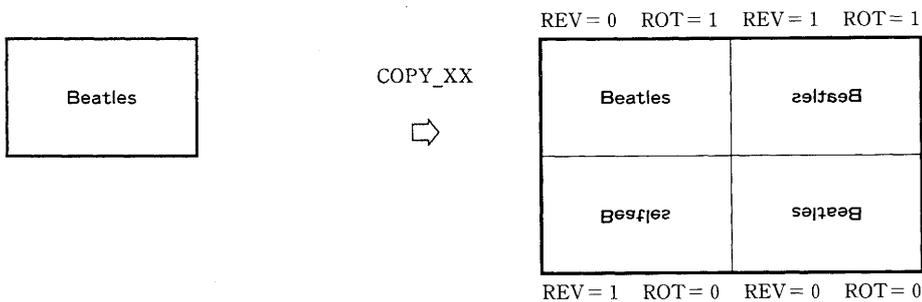
図6-4にREV=0, ROT=0-REV=1, ROT=1の4種類の場合のアドレス・スキャン方向について示します。

図6-4 ESE=1としたときのコピー・スキャン・アドレス

- (a) ソース・データ・アドレス・シーケンス      (b) デスティネーション・データ・アドレス・シーケンス



Beatlesの文字を転送した場合、次のようになります。



- (2) コピーできます。
- (3) コピーできます。

描画時の論理演算モードをDestination←0にすればコピーできます。

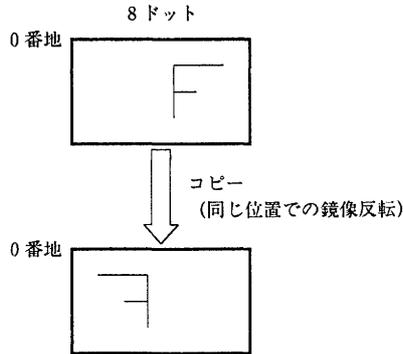
Q6.41

内容は? 機能・動作 特性 応用

コピー・コマンドでデータの鏡像反転 (REV=1) を選択し、転送源と転送先の位置を等しくしてコピーを実行した場合、コピーされたデータの値は保証される?

設定例

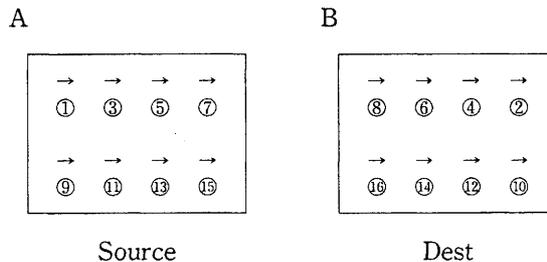
PITCHS =1H  
 PITCHD =1H  
 EAD1 =0H  
 dAD1 =7H  
 EAD2 =0H  
 dAD2 =0H  
 DH =7H  
 DV =7fffH  
 COMMAND=784CH



備考 コピー幅は8ビットのみです。

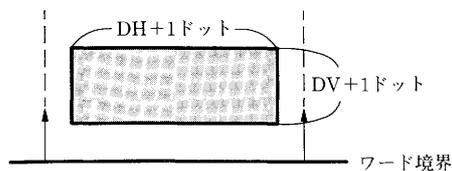
**A6.41**

通常は保証されません。読み書き順序は、下図のようになります。



(AとBは同じ位置にあるピクセルとします。)

ただし、この場合のパラメータにかぎって言えば、(転送矩形領域の1ワード内に納まっているならば)保証されます。

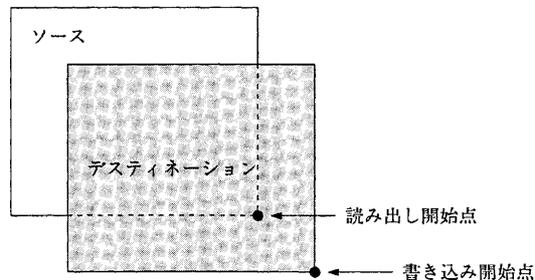


## Q6.42

内容は？

機能・動作
 特性
 応用

転送源矩形領域と転送先矩形領域が下図のように重なるときの単純コピーは実行できる？



## A6.42

実行できます。ソース矩形領域の読み出し開始点（画面左上/右下）と走査方向を決めるESEフラグ，デスティネーション矩形領域の書き込み開始点と書き込み走査方向を決めるROTフラグの設定に注意してください。

上図の例では，ソース矩形領域の読み出し開始点（X，Y）を右下に設定するのでESE=1，走査方向は右下から左上なのでROT=1になります。

## Q6.43

内容は？

機能・動作
 特性
 応用

縮小コピーで，ソースとデスティネーションを同じ領域にすることはできる？

## A6.43

できます。

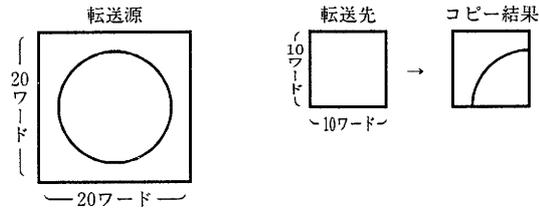
COPY\_AAを例にとって考えた場合，単純の縮小コピーを行うときに，EAD1=EAD2，dAD1=dAD2としてコマンドを発行してもコピーは可能です。

## Q6.44

内容は？

機能・動作
 特性
 応用

コピー・コマンドの際、転送先メモリアドレスの横幅が10ワードのとき、そこへ横幅20ワードのデータを転送しようとした場合、クリッピングで横幅を10ワードに制限しておけば正常なコピー動作が行われる？



## A6.44

コピー動作は正常に行われます。

コピー・コマンドを発行する際、実際の転送先のメモリアドレスの横幅が10ワードしかないところに横幅20ワード分のデータを転送するには、転送先のメモリ領域に対してクリッピングをかければ、メモリのPITCHが20ワードのままでデータを転送することができます。また、転送源の読み出し開始位置、転送先の書き込み開始位置がそれぞれ左上端からであれば、上の図のようなコピー結果が得られます。

## Q6.45

内容は？

■機能・動作 □特性 □応用

任意角回転コピーにおいて、FSフラグを1にすると、どの点を描画する？

## A6.45

任意角回転コピー (FR\_COPY<sup>注</sup>) のアルゴリズムを以下に示します。

FR\_COPYを実行した場合、AGDCは、DDA (Digital Differential Analyzer) により直線描画を行います。直線描画はX-軸に対してその角度が $0 \leq \theta \leq 45^\circ$ のときと $45^\circ < \theta < 90^\circ$ のときの2つに分けられます。例を図6-5に示します。

注 正式には、任意角回転、拡大/縮小コピー (FR\_ES\_COPY) ですが、ここでは任意角回転のみを表すため、FR\_COPYとしています。

図6-5 DDAによる直線描画

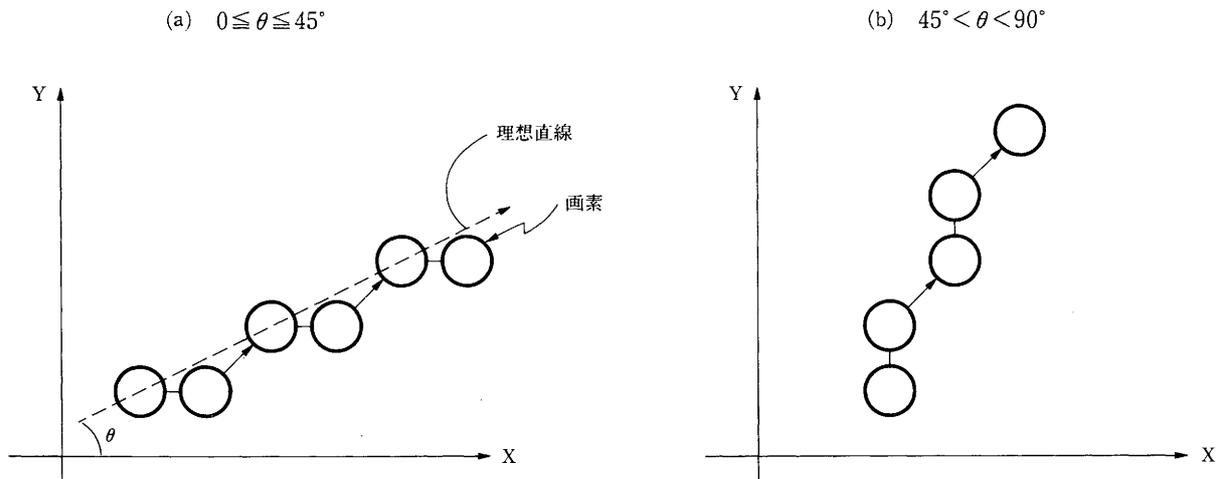


図6-6に長方形を回転コピーする例を示します。AGDCは2つのDDAを持っており、まず最初の第1ライン a1 - b1 - c1 - d1 - e1 - f1をA1 - B1 - C1 - D1 - E1 - F1に転送します。この転送を1つのDDAで行います。第2ラインa2 - b2 - c2 - d2 - e2 - f2をA2 - B2 - C2 - D2 - E2 - F2へ転送します。この転送をもう1つのDDAで行います。

この繰り返しにより第3, 第4, 第5ラインを転送します。

図6-6(b)を見て分かるようにこの場合●の描画しない点が生じます。

この描画しない点を描画したい場合、FS (Fill Shortage) フラグを1にします。

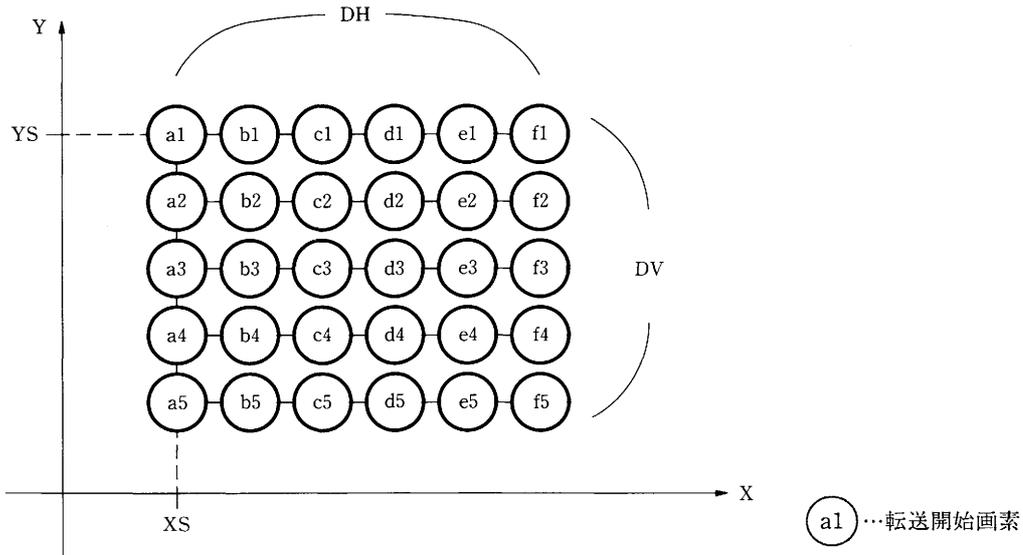
FS=1とした場合の直線描画を図6-7に示します。

また、図6-6(a)をFS=1で任意角回転コピーを実行した結果を図6-8に示します。

このとき、X1のデータはA1のデータと同一に、X2のデータはC1のデータを同一にX3のデータはE1と同一に  
 …となります。

図6-6 長方形を任意角回転コピーした例

(a) 転送する前の絵



(b) 任意角回転コピーを行った絵

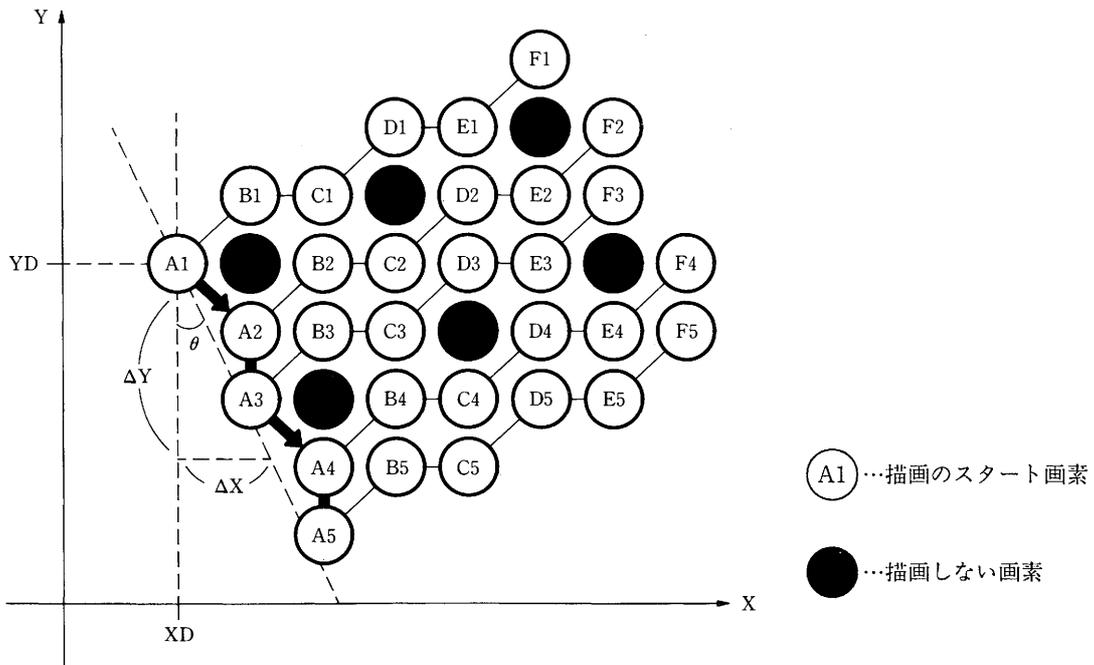
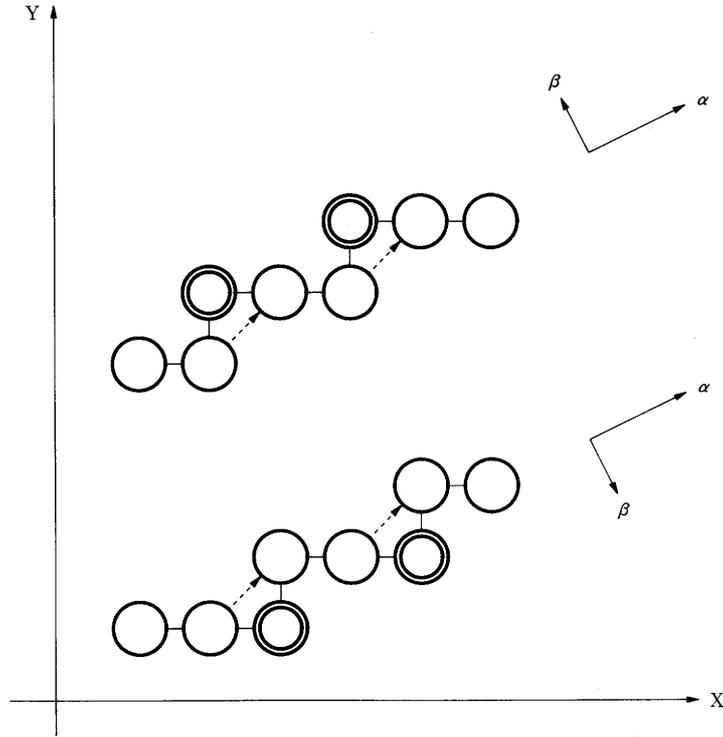


図 6-7 FSフラグを1にしたときの直線描画

(a)  $\alpha$ の方向の傾き $\theta$ が $0^\circ \leq \theta \leq 45^\circ$ のときの直線描画◎の位置は、次に描画する直線の位置方向 $\beta$ の向きによって異なります。



(b)  $\alpha$ の方向の傾き $\theta$ が $45^\circ < \theta < 90^\circ$ の場合

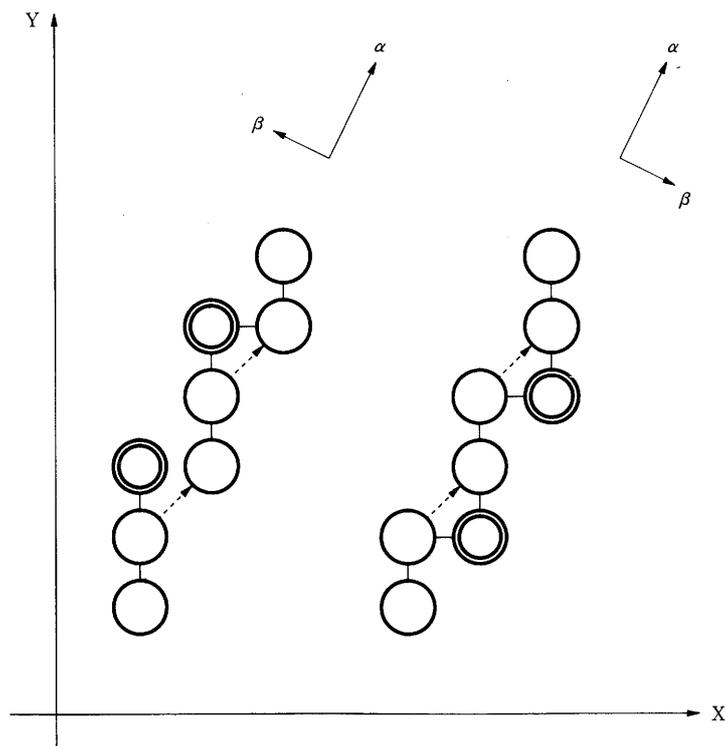


図 6-8 図 6-6(a)の図をFS=1として任意角回転コピーを行った場合

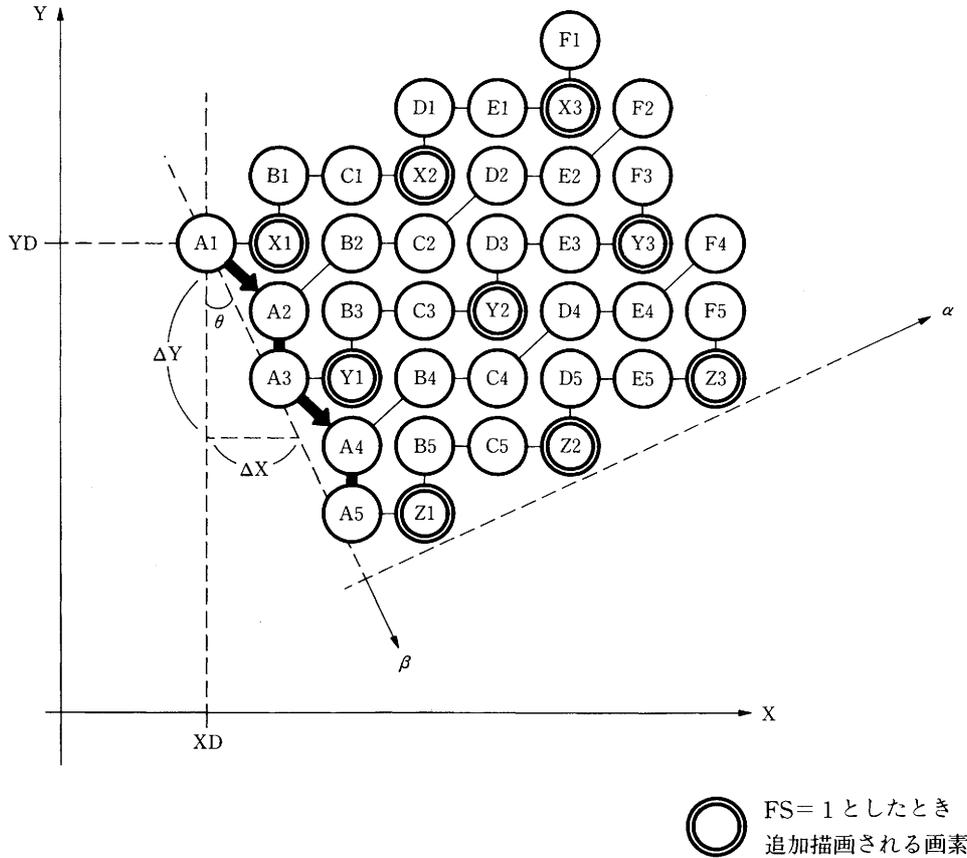


図 6-6(b)の●部が描画されている。

**Q6.46**

	内容は？ <input checked="" type="checkbox"/> 機能・動作 <input type="checkbox"/> 特性 <input type="checkbox"/> 応用
拡大／縮小コピーを行うときに、任意角回転・拡大／縮小コピーを用いても拡大／縮小できる？	

**A6.46**

できます。

ただし、任意角回転・拡大／縮小コピー・コマンドは、ドット単位で処理を行います。したがって、コピーの速度が遅くなります。

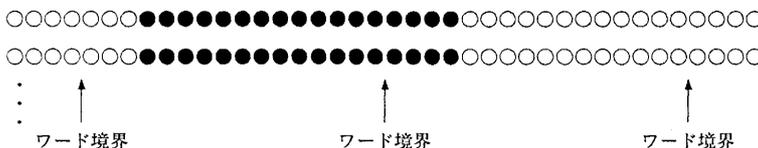
回転を伴わない拡大／縮小コピーの場合は、通常の拡大／縮小コピー・コマンドで行うことを奨めます。また、任意角回転・拡大／縮小 (COPY\_CA) の際の0度のコピーは、DX=1, DY=0, XE=0, YE=-1を設定します。

Q6.47

内容は？

機能・動作 特性 応用

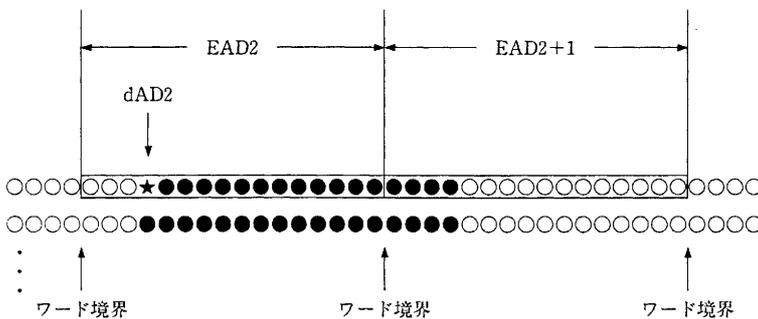
下図のようにソース領域にデータがあったとして、●のデータ領域分をデスティネーション領域にコピーしたい場合、AGDCのワード単位のコピーはどのように行う？



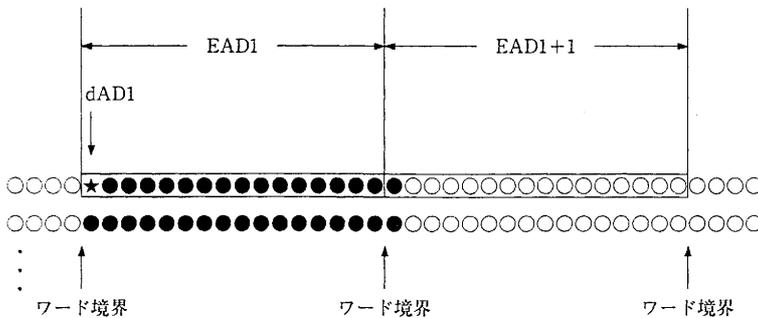
**A6.47**

COPY\_AAを例にとって考えた場合、下図のようにEAD2, dAD2レジスタにデータを設定したとします。AGDCは、EAD2番地のデータと、EAD2+1番地のデータを連続して読み出して16ビット（1ワード）のデータを作成します。その後、EAD1, dAD1レジスタに設定されたメモリ領域にデータを書き込みます。

[ソース領域]



[デスティネーション領域]



Q6.48

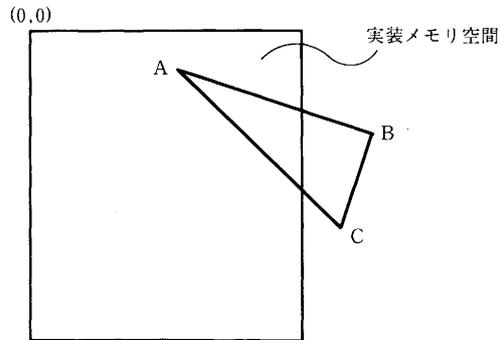
内容は？

機能・動作

特性

応用

実装メモリ空間を越えて三角形を描画するとどうなる？

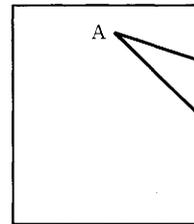
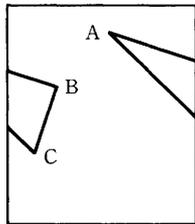


**A6.48**

描画点の演算は、PITCHS, PITCHDで行われますので、実装メモリ空間を越えて描画した場合、メモリ空間に折り返しが出ます(図6-9参照)。また、AGDCのクリッピング機能を使用すれば図6-10のようになります。

図6-9 折り返しが生じる場合

図6-10 AGDCのクリッピング機能を使用した場合



## Q6.49

内容は？

機能・動作
 特性
 応用

漢字ROMについて

- (1) 容量, アドレスなどには, 制限がある？
- (2) アプリケーション・ノートで, 漢字ROMの配置を工夫している？
- (3) 24×24ドットのフォント展開はできる？

## A6.49

- (1) 容量, アドレスなどには, 特に制限はありません。
- (2) JIS規格では文字の配置は14ビットのJISコードで示されますが, このJIS符号表で文字の配置されている領域は不連続です。したがって, 文字を再配置して連続した領域とすることで, 漢字キャラクタ・ジェネレータの容量を小さくするために行っています。
- (3) 24×24ドットのフォントの展開を考えた場合, AGDCはデータを16ビット単位で扱います。したがって, フォントの横幅としては, 2ワード設定してください。  
フォントの大きさは, DH, DVレジスタに設定します。これらのレジスタに32を設定すれば, 32×32ドットのフォントでも対応できます。

## Q6.50

内容は？

機能・動作
 特性
 応用

漢字ROMから文字を表示メモリに転送した後, ソフトで表示をON, OFFさせるのではなく, AGDCの機能でCRTに点滅させて表示できる？

## A6.50

AGDCは, ハードウェア・カーサを実現するためのステータス信号を出力することができます。GCSR端子から出力されるカーサ信号を利用して外部回路を設計して頂ければ, ブリンキングをハードウェアで実現することができます。

## Q6.51

内容は？

機能・動作特性応用

文字データをおあるバック・グラウンド上へ転送する方法は？

## A6.51

文字データを単純に論理演算REPLACEで転送（コピー・コマンド）すると図6-11のようになります。これをバック・グラウンド上に文字部のみを出したい、つまり、トランスペアレンシ（Transparency）を実行したい場合は次のように行います。

図6-11 バック・グラウンド・データへフォア・グラウンド・データをコピーするときの論理演算

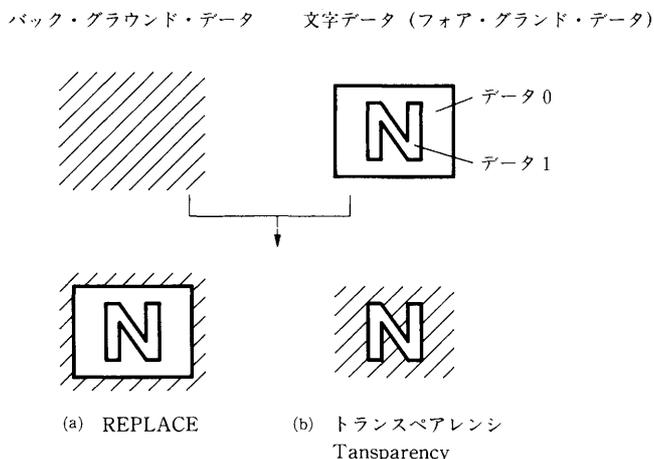


図6-11(b)のトランスペアレンシを実行する方法

文字データをフォア・グラウンドとした場合

文字データをソース、バック・グラウンド・データをデスティネーションとしてソース中の1のデータは、 $D \leftarrow S$ 、0のデータは、 $D \leftarrow D$ を実行することにより行えます。

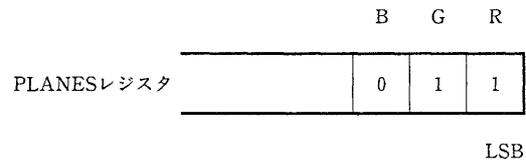
このためには、

MOD0レジスタに $D \leftarrow \bar{S} \cdot D$ MOD1レジスタに $D \leftarrow S + D$ 

を設定し、PLANESレジスタに描画したい色を設定することによって行えます。

PLANESには、たとえば、黄色で描画したければ、システムがR.G.Bの3プレーンを持つと仮定すれば、Bに0、GとRに1を設定してください。

図 6-12 PLANESレジスタへ黄色を設定した例



また、 $\mu$ PD72123では、3オペランド・コピーにより、簡単に実現することができます。

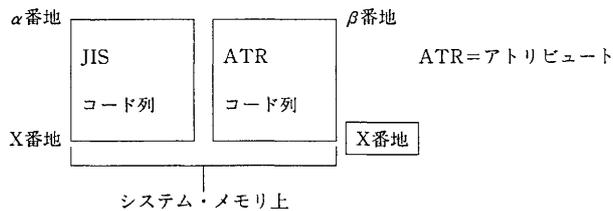
Q6.52

内容は? 機能・動作 特性 応用

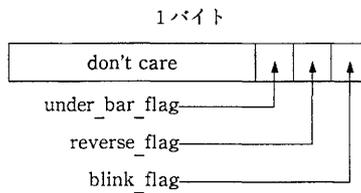
キャラクタへのアトリビュート（下線，白黒反転，ブリンクなど）の付け方は？

**A6.52**

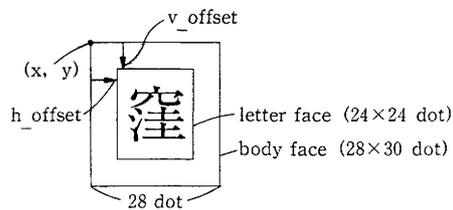
「白黒反転」との記述から、「文字はモノクロ表示である」という前提で回答します。  
文字のデータ構造をどうするかにもよりますが、一例をあげます。



- (1) 上記のエリアとスクリーン上の文字表示位置 (xy座標) が、1対1対応をしているものとします。  
α+n番地に“窪”という文字のJISコードを書くと、β+nのATRコードを参照しながら、対応する (Xn, Yn) を左上の点として「窪」のフォント・データをコピーします。
- (2) ATRコードは例えば次のようなフォーマットになっているものとします。



- (3) 文字は、次のような定数で制御されるものとします。



**下線**

ATRを参照し、`under_bar_flag=1`ならば、

スタート=(X, Y+29)      として直線を引きます。

エンド=(X+27, Y+29)

**白黒反転**

ボディ・フェース分 (28×30 dot) のすべて「1」のビット・ブロック・データをワーク・メモリに用意します。

このデータをソースとし、また反転させたい文字のボディ・フェースをデスティネーションとします。そして、論理演算モードをEXORにしCOPYを実行します (カラーの場合には補色になります)。

**ブリンク**

タイマ割り込みで一定のブリンク・タイミングを作り、割り込みがかかるごとに上記の「EXORのCOPY」を実行します。

**Q6.53**

内容は？ 機能・動作 特性 応用

90°回転コピーのソースとデスティネーションのアドレス発生について

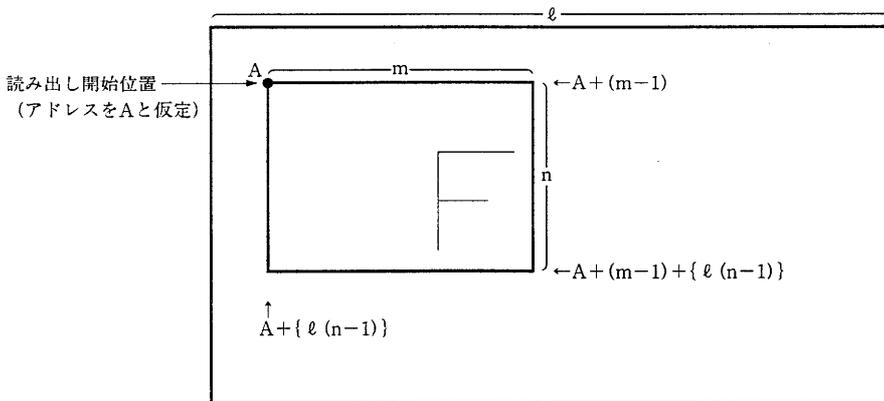
- (1) 読み出し開始位置が左上の場合の90°回転コピーはどうなる？
- (2) 読み出し開始位置が右下の場合の90°回転コピーはどうなる？

**A6.53**

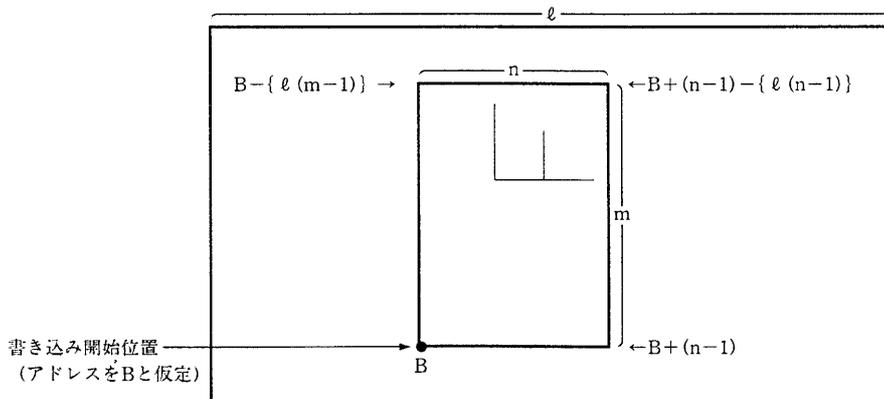
- (1) 読み出し開始アドレスAから、 $A, A + \ell, A + 2\ell, A + 3\ell, \dots$ と16ワードずつ読み出しては、書き込み開始アドレスBより、 $B, B - \ell, B - 2\ell, \dots$ と16ワードずつ書き込んで行きます。また、次の16ワードは、 $A + 1, (A + 1) + \ell, (A + 1) + 2\ell, (A + 1) + 3\ell, \dots$ とデータを読み出し、 $B - 16\ell, B - 17\ell, \dots$ と書き込んで行きます。

図 6-13 読み出し開始位置が左上の場合の90°回転コピー

ソース矩形領域



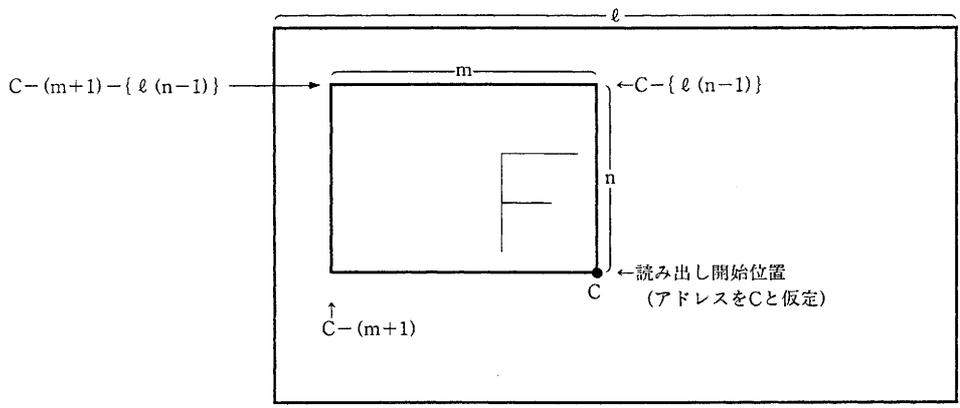
デスティネーション矩形領域



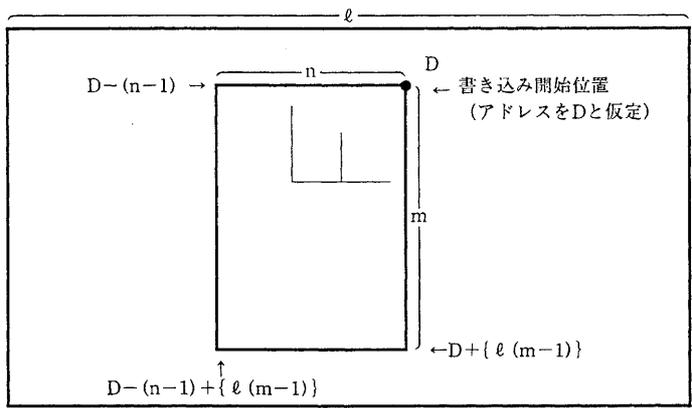
- (2) 読み出し開始アドレスCから、 $C, C - \ell, C - 2\ell, C - 3\ell, \dots$ と16ワードずつ読み出しては、書き込み開始アドレスDより、 $D, D + \ell, D + 2\ell, \dots$ と16ワードずつ書き込んで行きます。  
 また、次の16ワードは、 $C - 1, (C - 1) - \ell, (C - 1) - 2\ell, (C - 1) - 3\ell, \dots$ とデータを読み出し、 $D + 16\ell, D + 17\ell, \dots$ と書き込んで行きます。

図6-14 読み出し開始位置が右下の場合の90°回転コピー

ソース矩形領域



デスティネーション矩形領域



備考 90°回転コピーの場合は、上記のような処理をAGDC内部で行っています。

## Q6.54

内容は？

■機能・動作 □特性 □応用

90°回転と1/2縮小をするとき

- (1) 最も速い処理方法は？
- (2) そのときのスピードは？
- (3) そのときのメモリ領域の取り方は？

## A6.54

- (1) 1/2縮小 → 90°回転が最も速くできます。  
(ES\_COPY) (90\_COPY)
- (2) 以下の計算式からES\_COPYと90\_COPY速度を算出してください。

	コマンド	プリプロセッサ	描画プロセッサ
1/2縮小	ES_COPY_CC	110+PS+PD	$((2+7\times\alpha)\times((H/16)+1)\times L+2\times 16/MV)\times V\times MV/16+4$
90°回転	90_COPY_CC	90+PS+PD	$(3\times WH\times V)+(4\times WV\times H)+((4\times WH+4)\times WV)+5$

注意1. 処理ステップ数 (CLKをベースとした概算値)

2. 90\_COPYは、単純コピーに比べて1.1~1.2倍の速度低下になります。

★ 備考 PS : PITCHSレジスタ値の有効ビット数

PD : PITCHDレジスタ値の有効ビット数

MV : MAGVレジスタ値+1

 $\alpha$  : 0, または1

L : 描画対象となるプレーン数

WH : 読み出し対象となる水平方向ワード数 ...  $((DHレジスタ値+1)/16)+1$ 

V : 読み出し対象となる垂直方向画素数 ... DVレジスタ値+1

WV : 読み出し対象となる垂直方向ワード数 ...  $((DVレジスタ値+1)/16)+1$ 

H : 読み出し対象となる水平方向画素数 ... DHレジスタ値+1

- (3) 90\_COPYは、ソース領域とデスティネーション領域とがオーバーラップしないようにしてください。

AGDC自身がワーク用としてメモリを使用するのは、PAINTコマンドのみです。

## Q6.55

内容は? 機能・動作 特性 応用

ショート・ベクタ (線幅4ビット, X軸に対し45°方向へ40ドットの線分) の描画速度は?

## A6.55

実行ステップ数は,

太直線コマンド… プリプロセッサ=120 CLK

描画プロセッサ=(2W+8)P+15

W=太さドット数 P=直線長さドット数

∴120+(2×4×8)+40+15=775 [CLK]

8 MHz時125 [ns]×775=97 [μs]=0.1 [ms]

□10 K ベクタ/秒 (600 k ベクタ/分)

## Q6.56

内容は? 機能・動作 特性 応用

ピクセル単位の互換性について

- (1) 同一のコマンド, かつ同一のパラメータを用いてAGDCに描画をさせる場合には, いつ描画させても実行結果は同じ?
- (2) 直線描画コマンドで描画開始点と描画終了点を入れ替えても描画結果は同じ?

## A6.56

(1) ピクセル単位で必ず同じになります。

(2) 異なる場合があります。詳しくはμPD72123のユーザーズ・マニュアル「付録H ピクセル単位の互換性」を参照してください。

## Q6.57

内容は？

■機能・動作

■特性

□応用

代表的な描画コマンドの描画速度は？

- ・図形描画 ; A\_LINE\_M0  
A\_LINE\_D0  
A\_REC  
CRL  
CARC
- ・塗りつぶし ; A\_REC\_FILL\_C
- ・コピー ; COPY\_AC  
COPY\_CC  
ES\_COPY (Enlarge)
- ・システム 表示メモリ ; PUT\_C  
PUT\_A  
GET\_A

★

## A6.57

計算式を以下に示します。

処理ステップ数 (CLKをベースとした概算値)

コマンド	プリプロセッサ	描画プロセッサ
A_LINE_M0	110+PD	$4 \times P \times L + 4$
A_LINE_D0	90+PD	$4 \times P \times L + 4$
A_REC	70+PD	$(4 \times P \times L + 4) \times 4$
CRL	140+PD+DV	$(4 \times P + 2) \times L + 6$
CARC	150+PD+DV	$(4 \times P + 2) \times L + 6$
A_REC_FILL_C	90+PD	$4 \times ((H/16) + 1) \times L + 5 \times V + 2$
COPY_AC	40+PS+PD	$((6 \times ((H/16) + 1) + 5) \times L + 3) \times V + 2$
COPY_CC	60+PS+PD	$((6 \times ((H/16) + 1) + 5) \times L + 3) \times V + 2$
ES_COPY_CC	110+PS+PD	縮小時 $((2 + 7 \times \alpha) \times ((H/16) + 1) \times L + 2 \times 16/MV) \times V \times MV/16 + 4$
		拡大時 $((2 + (4 \times 16/MV + 4) \times \alpha \times 16/MH) \times ((H/16) + 1) \times L + 2) \times V + 4$
PUT_A	70+PD	$((5 \times H/16) + 1) \times P + 5 \times V + 7$
PUT_C	90+PD	$((5 \times H/16) + 1) \times P + 5 \times V + 7$
GET_C	60+PD	$((10 \times H/16) + 1) + 4 \times P \times V + 9$

- 備考 PS : PITCHSレジスタ値の有効ビット数  
 PD : PITCHDレジスタ値の有効ビット数
- PS, PDにおける有効ビット数とは、レジスタ値を2進数にしたときの、有効桁数のことです。
- 例 PITCHSまたはPITCHDが0028Hのとき、  
 0028H=0000000000101000B  
 有効桁数は101000の6桁で、PSまたはPDは6となります。
- DV : DVレジスタ値の有効ビット数  
 P : 描画対象となる画素数  
 L : 描画対象となるプレーン数  
 H : 読み出し対象となる水平方向画素数  
 V : 読み出し対象となる垂直方向画素数  
 MV : MAGVレジスタ値+1  
 MH : MAGHレジスタ値+1  
 $\alpha$  : 1, または2
- 転送先矩形領域に1ワードのデータを書き込む際に、ワード境界にまたがらない場合は“1”またがる場合は“2”となります。
- COPY, PUT, GET : SD\_SEL=11B (マルチソース&マルチデスティネーションの前提で計算)

## Q6.58

内容は？

機能・動作
 特性
 応用

システム・メモリ上にあるデータを表示メモリに取り込むには？

## A6.58

AGDCが管理するメモリに、システム・メモリ上にあるデータを転送することは、AGDCのPUTコマンドを使えばできます。

AGDCはシステム・メモリのデータを直接アクセスすることができないので、CPUまたはDMACがデータをAGDCに転送します。AGDCは転送されたデータを、CPUがあらかじめ設定した表示メモリの領域に書き込みます。

## Q6.59

内容は？

■機能・動作 □特性 □応用

PUT, GETコマンドを使うときの注意点は？

## A6.59

次の2点に注意してください。

- ① オペレーション・フラグのSD\_SEL=11bと設定してPUT, GETコマンドを発行すれば、うまくデータ転送が行われます。
- ただし、PUT\_A, GET\_Aコマンドを使用している場合は、CLIP=01bに設定してクリッピングをはずしてください。
- GET\_Cを使用する場合は、PITCHSレジスタとPITCHDレジスタには同じ値を設定してください。
- ② ステータス・レジスタのPGRDYフラグを確認しながらデータを転送すれば、AGDCがロックすることはありません。
- PGPORTを読み出すとき (GETコマンド時) は、READY信号の制御も行ってください。

## Q6.60

内容は？

■機能・動作 ■特性 □応用

AGDCのPUT, GETの転送スピードは？

(条件1.で動作させた場合、1ワード転送にどのくらい時間がかかる？)

条件1. AGDC 7.5 MHz  
 マスタCPU 80386 16 MHz  
 DMA未使用

## A6.60

AGDCが表示メモリ・バス100%占有できるという仮定のもとでは下表のようになります。

	プリプロセッサ処理	描画プロセッサ処理
PUT	98+PITCHDの有効ビット数	1ワード当たり8サイクル
GET	88+PITCHDの有効ビット数	1ワード当たり11サイクル

備考 単位はAGDCのCLKのサイクル数

**Q6.61**

内容は？

■機能・動作 特性 応用

描画コマンドのスピードについて

- (1) 描画コマンドでCSECなどは、CARC+LINEという処理速度？
- (2) パラメータ設定などにより描画速度が変わるということだが、どのくらいの差がある？

**A6.61**

- (1) そのとおりです。
- (2) 10~20ステップ程度です。

**Q6.62**

内容は？

■機能・動作 特性 応用

線描画コマンドは1ドットずつ描画する？

**A6.62**

1ドットずつ描画します。

★

## Q6.63

内容は？

■機能・動作 □特性 □応用

GETコマンドにおいて複数プレーンのデータを論理演算してシステム・メモリにブロック転送するときのSD\_SEL, MOD0, MOD1, PLANESの設定方法は？

## A6.63

MOD0, MOD1を、どの演算に適用させるかを指定する方法は2種類あります。これはSD\_SELの設定により選択できます。

## ①SD\_SEL=00Bのとき

合成ソース・データを作成するための各プレーン間で行う論理演算をMOD1で設定します。また、合成ソース・データとデスティネーション間で行う論理演算をMOD0で設定します。このときPLANESはプレーン・マスク用のレジスタになり、PLANES内の“0”が設定されているビットに対応するプレーンは演算に関与しません。ただし、第1プレーンは例外で必ず関与します。

## ②SD\_SEL=01Bのとき

合成ソース・データを作成するための各プレーン間で行う論理演算をMOD0, MOD1で設定します。また、合成ソース・データとデスティネーション間で行う論理演算は、リプレース演算 (D→S) に固定されます。このときPLANESはソース・データ間で行う論理演算を指定するレジスタになり、PLANES内の“0”が設定されているビットに対応するプレーンはMOD0, “1”が設定されているビットに対応するプレーンはMOD1の論理演算を実行します。

備考 詳細は $\mu$ PD72123のユーザーズ・マニュアル「10.8.3(3) マルチソース&シングル・デスティネーション (カラー縮小)」を参照してください。

## Q6.64

内容は？

■機能・動作 □特性 □応用

ソフトウェア・カーサの実現方法は？

## A6.64

ホスト・ダイレクト・アクセス，またはPUTコマンドで，画面上に表示させたいカーサ形状のデータ（矢印など）を表示メモリ上の連続した番地に定義します。

その定義した形状を，AGDCのコピー・コマンド（たとえば，COPY\_AC）を用いて表示領域にコピーすることで実現します。

例 カーサ形状が16×16ドットの形状であったとすると，そのデータを表示メモリ上の連続した番地にホストより書き込みます。

先頭番地が $\alpha$ 番地であったとすると，この場合 $\alpha$ 番地から $\alpha+15$ 番地に形状を定義します。

これを，COPY\_ACコマンドを用いて表示領域にコピーします。その場合の設定例は，以下のようになります。

[設定例]	EAD2	$\alpha$ H
	dAD2	0H
	X	コピー先の座標
	Y	
	DH	FH
	DV	FH
	COMMAND	800CH

## Q6.65

内容は？

■機能・動作 □特性 □応用

ピクセル構成で使用する場合の描画方法は？

## A6.65

8ビット／ピクセルの場合の1ワードのデータ構成は次のようになります。

1ワード RRGGBBIIIRRGGBBII

たとえば、黄色の実線を描画する場合は、PTNCNTレジスタには次のような値を設定します。

PTNCNTレジスタ 1111001111110011b

ただし、8ビット／ピクセル・モードで使用すると、1プレーン当たり2ビットの自由度しかありません。したがって、実線、または1ドットおきの点線しか描画することができません。

直線や円の描画は、通常にレジスタ設定を行いコマンドを発行してください。

8ビット／ピクセルのとき、dADORGレジスタの4ビットに関しては、0, 1のどちらでもかまいません。どのようなときに0, または1にするという規定はありません。

AGDCをピクセル・モードで使用することができるコマンド群すべて、CPUから与えるパラメータは、XY座標で与えます。

すなわち、絶対番地の変換はAGDCのプリプロセッサが行います。したがって、プログラム上で作成することはできません。

**アンケート記入のお願い**

お手数ですが、このドキュメントに対するご意見をお寄せください。今後のドキュメント作成の参考にさせていただきます。

[ドキュメント名] AGDC Q&A集, インフォメーション (IEI-647C (第4版))

[お名前など] (さしつかえのない範囲で)

御社名 (学校名, その他) ( )  
ご住所 ( )  
お電話番号 ( )  
お仕事の内容 ( )  
お名前 ( )

1. ご評価 (各欄に○をご記入ください)

項 目	大変良い	良 い	普 通	悪 い	大変悪い
全体の構成					
説明内容					
用語解説					
調べやすさ					
デザイン, 字の大きさなど					
そ の 他 ( )					
( )					

2. わかりやすい所 (第 章, 第 章, 第 章, 第 章, その他 )  
理由 [ ]

3. わかりにくい所 (第 章, 第 章, 第 章, 第 章, その他 )  
理由 [ ]

4. ご意見, ご要望

5. このドキュメントをお届けしたのは

NEC 販売員, 特約店販売員, NEC 半導体ソリューション技術本部員,  
その他 ( )

ご協力ありがとうございました。

下記あてにFAXで送信いただくか, 最寄りの販売員にコピーをお渡ししてください。

キ  
リ  
ト  
リ

**保守 / 廃止**

— お問い合わせは、最寄りのNECへ —

**【営業関係お問い合わせ先】**

半導体第一販売事業部 半導体第二販売事業部 半導体第三販売事業部	〒108-01 東京都港区芝五丁目7番1号 (NEC本社ビル)	東京 (03)3454-1111 (大代表)
中部支社 半導体販売部	〒460 名古屋市中区栄四丁目14番5号 (松下中日ビル)	名古屋 (052)242-2755
関西支社 半導体第一販売部 半導体第二販売部 半導体第三販売部	〒540 大阪市中央区城見一丁目4番24号 (NEC関西ビル)	大阪 (06) 945-3178 大阪 (06) 945-3200 大阪 (06) 945-3208
北海道支社 東北支社 北支社 山形支社 山支社 いわき支社 長岡支社 土浦支社 水戸支社 神奈川支社 群馬支社 茨城支社 栃木支社 宇都宮支社	札幌 (011)231-0161 仙台 (022)261-5511 盛岡 (0196)51-4344 山形 (0236)23-5511 山支店 (0249)23-5511 いわき支店 (0246)21-5511 長岡支店 (0258)36-2155 土浦支店 (0298)23-6161 水戸支店 (0292)26-1717 神奈川支店 (045)324-5511 群馬支店 (0273)26-1255 茨城支店 (0276)46-4011 栃木支店 (0286)21-2281	小長野支社 (0285)24-5011 山形支店 (0262)35-1444 松本支店 (0263)35-1666 上野原支店 (0266)53-5350 早稲田支店 (0552)24-4141 埼玉支店 (048)641-1411 立川支店 (0425)26-5981 千葉支店 (043)238-8116 静岡支店 (054)255-2211 沼津支店 (0559)63-4455 浜松支店 (053)452-2711 北陸支店 (0762)23-1621 福井支店 (0776)22-1866
富山支社 石川支社 福井支社 山梨支社 長野支社 新潟支社 富山支社 石川支社 福井支社 山梨支社 長野支社 新潟支社	富山支店 (0764)31-8461 石川支店 (0592)25-7341 福井支店 (075)344-7824 山梨支店 (078)332-3311 長野支店 (0857)27-5311 新潟支店 (086)225-4455 富山支店 (0878)36-1200 石川支店 (0897)32-5001 福井支店 (0899)45-4111 山梨支店 (092)271-7700 長野支店 (093)541-2887 新潟支店	

**【本資料に関する技術お問い合わせ先】**

半導体ソリューション技術本部 第一システム技術部	〒210 川崎市幸区塚越三丁目484番地	川崎 (044)548-8884	半導体 インフォメーションセンター FAX(044)548-7900 (FAXにてお願い致します)
半導体販売技術本部 東日本販売技術部	〒108-01 東京都港区芝五丁目7番1号 (NEC本社ビル)	東京 (03)3798-9619	
半導体販売技術本部 中部販売技術部	〒460 名古屋市中区栄四丁目14番5号 (松下中日ビル)	名古屋 (052)242-2762	
半導体販売技術本部 西日本販売技術部	〒540 大阪市中央区城見一丁目4番24号 (NEC関西ビル)	大阪 (06) 945-3383	