ビ デ オ ゲ ー ム

東京芝浦電気株式会社 今 村 誠
竹 内 久 晴
大 須 賀 英 巳

テレビ画面でゲームを行うビデオゲーム機が普及してきた。本文では、将来のホームコンピュータにまで発展すると期待されている家庭用のマイクロコンピュータ使用ビデオゲーム機について、ハードウェアとソフトウェアの概略を述べる。

1. ま え が き

ビデオゲームが最初に商品化されたのは 1972 年¹⁾の ATARI 社の Pong と Magnavox 社の Odyssey といわれ業務用を中心に発展して来たが、家庭用は 1975 年頃よりランダムロジックを中心とした専用 IC を用いたゲーム機が生産されるようになり、爆発的に売れた。これらはいずれもパドルゲーム*的なもので、同じ種類のゲーム機であった。

その後、同一ゲーム機で異なった種類のゲームを行う目的で、回路ブロックを組合せることによって変化を持たせたゲーム用 IC が市販されだした。

この方式には2つの流れがあり、第1の方式はランダムロジックを機能ブロック毎に IC 化し必要な組合せによってゲームの種類を変更するもので、音響発生部、画像発生部、入力装置等にそれぞれ数種類の異なった IC を製作し、目的とするゲームに必要な IC を用いてゲームを組立てる方式である。

第2の方式はマイクロコンピュータを用いたストアードプログラム方式を採用し、同じハードウェアを用いたプログラムを内蔵した ROM のみを交換することによって、異なったゲームを実現する方式である。

このストアードプログラム方式はマイクロコンピュータの発達によって出現した方式で、ハードウェアの変更なしでただゲームプログラムのみ変更することによって新しいゲームに変身するし、マイクロプロセッサの演算機能を生かしてゲームの動作にランダム(偶然)な機能を出したり、人間とゲーム機とが対戦することができ

本 体 カートリッジ

ジョイスティック

写真1 チャンネル F (FCI)

る1人用ゲーム機を実現することができた。

これらはゲーム機として当然要求される機能であり、マイクロコンピュータはこの要求に充分応えられると期待されているため、この種のゲーム機を中心に述べて行く。

国内においては専用 IC を用いたパドルゲームがいまだに多いが、米国においてはマイクロプロセッサを用いた各種のゲーム機が話題の中心となっている。マイクロコンピュータを用いた家庭用ビデオゲームは 1976 年発売の“Channel F”(FCI, 写真1), 1977 年春発売の“STUDIO-II”(RCA), 同年のクリスマス頃に発売された“Video Computer System”(ATARI), 1978 年春発売の“ビジコン”(東芝), “ARCADE”(Bally), “Video Brain”(VideoBrain) 等があり表1にまとめておく。

これらのビデオゲーム機はすべて汎用8ビットマイク

¹⁾“Video Games” by Makoto Imamura, Hisaharu Takeuchi and Hidemi Ohsuka (Toshiba Corporation, Kawasaki)

* パドル(楯)状のラケットでボール等を打つテニスなどのゲーム。

表 1 ビデオゲームの一覧

	Channel F	STUDIO-II	Video Computer System	ビ ジ コ ン	ARCADE	VideoBrain
メーカ名	FCI	RCA	ATARI	東芝	Bally	VideoBrain
CPU	F 8 3850	COSMAC 1802	6507	COSMAC 1802	Z 80 3880	F 8 3850
画素数	64×64	64×32	128×128	64×32	160×100	128×128
カラー, B/W	カラー	B/W	カラー	カラー	カラー	カラー
内蔵メモリー容量 ROM	2 kB	1 kB		2 kB	8 kB	4 kB
内蔵メモリー容量 RAM	2 kB	0.5 kB	128B	0.75 kB	4 kB	1 kB
カートリッジメモリー容量	1 kB	1 kB	2 kB	2 kB	2 kB	2 kB
内蔵ゲーム数	2	5	0	5	4	4
入力装置	キーボード, ジョイスティック	キーボード	ジョイスティック, ADC	キーボード, ジョイスティック	キーボード, ジョイスティック	キーボード, ジョイスティック
表示方式	グラフィック	グラフィック	キャラクター	グラフィック	グラフィック	
備考	世界最初のマイコン利用ビデオゲーム	CMOS	ジョイスティックはデジタル入力とアナログ入力	国内最初のマイコン利用ビデオゲーム	最大4個のジョイスティックが使用できる	フルキーボード

図 1 ビデオゲームの基本構成図

ロプロセッサを内蔵し、ゲームソフトウェアは外部より挿入されるカートリッジ内の ROM にプログラムされている。

図 1 はストアードプログラム方式ビデオゲーム機の基本構成図である。ゲームソフトウェアが入っているプログラムメモリーの内容に従って制御演算部は入力装置、すなわちオペレーターによってコントロールされるキーボードあるいはジョイスティック*に連動する AD 変換器の状態を読み込み、この情報に基づき必要な演算を行い音響、画像情報を音響発生部、画像発生部に与える。こ

の両情報は音響信号、画像信号に変換され RF 変換部にて、FM 変調波としての音響信号、NTSC テレビ信号としての画像信号に変換し、汎用テレビ受像機にて受信できるように RF 変調される。オペレーターは受像機の画面を見てゲームの進行状況を知り（フィードバックされ）ゲームを続行することができる。

2. ゲーム機のハードウェア

2.1 制御演算部

ビデオゲームに用いられているマイクロプロセッサは 8 ビット CPU が最も多く、しかも要求される性能も当然ながら高速で多機能であるが、具体的に重要なことは次の 4 点である。

(a) ビット処理が容易であること

テレビ画面上にアニメーション表示するため画像処理に必要な処理時間が、ゲーム全処理時間の 80% 以上を占めている例がある。この画像処理はアキュムレーター内の個々のビットに対するシフト命令、理論演算が中心である。すなわちビット処理が容易になるとゲーム機は高速動作ができるようになる。

(b) データ転送速度が速いこと

画像を表示するリフレッシュメモリーに CPU より画像情報を転送するが、任意の位置に、しかも CPU の語長までしか転送できないため大型の画像を表示する場合、長時間必要となる。このため修飾アドレス方式の種類が多いとリフレッシュメモリーのアドレッシングが容

* 操縦桿

易になり、同じクロック周波数でも転送速度が上がる。

(c) インタラプト機能を持っていること

テレビ画面上に画像を表示するため CPU より任意の時間にリフレッシュメモリー上にデータを書込むと、読出しタイミングと一致してフリッカを起こす場合があり、これを防ぐため読出しタイミングに関連した水平同期信号、垂直同期信号のタイミンギングを CPU 側が知らねばならず、インタラプトによって容易に同期をとることができる。

(d) ゲームソフトウェアの開発が容易であること

ゲーム機の生命はゲームソフトウェアであるといっても過言ではなく、最も労力を必要とする部分である。このためハードウェアにもソフトウェア開発が容易なシステム設計であることは当然であるが、ソフトウェア開発においてモニター、アセンブラプログラムが利用でき、しかもゲームのシミュレーションによりデバッキングし欠陥を未然に防ぐ必要がある。

マイクロコンピュータを選択するときハード的な面よりのみ検討し、開発用ソフトウェアについて見落としがちであるが、ゲーム機の場合、ソフトウェア主導であるため、この点を特に考慮すべきである。

2.2 プログラムメモリー

プログラムメモリーは、すべてのゲームに共通な OS プログラムと個々のゲームに必要なゲームプログラムに分類される。OS プログラムは文字発生、乱数発生、音響発生、入力読込み等、ゲームを維持するのに必要な共通なプログラムであるため、ゲーム機内部に組込むことがシステムの的に効率が良い。

一方、ゲームプログラムはピンポン、ブラックジャック等、個々のゲームに特有なプログラムで、一般にメモリーはカートリッジ形式の基板に取付けられ、ゲーム機本体に外付けされる。

2.3 入力装置

ゲーム選択用とオペレーターのアクションを入力する装置に分類できる。ゲーム選択には簡単なスイッチで良いが、アクション入力にはアナログ入力を可能にした AD 変換器とスイッチの組合せがある。アナログ入力方式では、ジョイスティックに直結した可変抵抗器の出力電圧を AD 変換器でディジタル量に変換している。変換器に要求される精度は制御対象によって異なるが、表示

図2 グラフィックディスプレイ

位置の場合7ビット以上、表示速度の場合4ビット以上が妥当である。スイッチを用いた方式には演算を中心としたゲーム向きの10キーボードと、アクションを容易にしたジョイスティック付8ポジションスイッチがある。写真1は前、後、左、右、右回り、左回り、上、下の8ポジションを可能にしたジョイスティックである。

2.4 画像発生部

CPU の処理能力はリアルタイムで画像表示するまでに至っていないため、1フィールドのリフレッシュメモリーを用いたグラフィックディスプレイ方式かキャラクタージェネレーター方式を採用している。前者は任意の画像表示が可能であるがリフレッシュメモリー容量が大きくなる。一方、後者は画像表示スピードは数倍上がるが、多種類のパターンを表示する場合、パターン用メモリー容量が増大するため、ゲームの目的に応じて決定する。

図2はグラフィックディスプレイ方式で、1ワード8ビットのメモリーを2枚使用し、4色のカラー表示を行っている。画像情報はCPUのデータバスを通してリフレッシュメモリーに8ビット並列に書込まれ、8ビット並列に読出され直列データに変換される。このため1画素のアクセス時間はメモリーのアクセス時間の1/8でよく、遅いメモリーでも細かい表示ができるが、同一番地の8ビットは同じ処理を行うため、画像を平行移動する場合不連続になり、ソフトウェアで連続にする必要がある。

キャラクタージェネレーター方式では、前者が8画素を同時に処理するのにくらべて、8×8画素を同時に処

図3 キャラクタジェネレータ方式の変形例

図4 音響発生部

理するため高速表示が可能である。ただ、この方式は8×8画素を単位とし、表示可能位置が決まっているためスムーズな移動が必要な場合、多くのパターンを必要とする。

この欠点を改良し、1種類のパターンを任意の位置に表示できる方式²⁾が図3である。CPUより転送されたデータは水平・垂直同期信号に同期して水平位置、垂直位置レジスタにストアされカウント開始位置を設定する。この値はパターンの表示位置に相当し(正確には画面の中心に対する点对称の位置)、任意の位置に任意の大きさのパターンをCPUの負担なしで表示できる。このため画像の移動を中心とし、しかもパターンの種類が少ないゲームに最適である。

2.5 音響発生部

効果音は画像情報と同じくらい重要な要素で複雑化する傾向にあり、楽音を発生するための発振器とプログラマブルカウンターの組合せ、擬音源となる白色ノイズが用いられる。図4は音響発生部の例でCPUの制御に従って分周比、振幅、デューティレシオを決定し、ゲーム

の進行に必要な効果音を流すことができる。信号源が発振器の場合出力は楽音となり、白色ノイズの場合リミッターにより零クロス点のみの情報を用いて白色ノイズの実用的な代用とし、衝突音、摺動音、爆発音等を発生することができる。

2.6 RF変調部

ここでは効果音、画像情報がベースバンドであるため、汎用テレビ受像機で受信できるように変換する。第1の方式は複数の独立したリフレッシュメモリ出力の論理和によりY信号、デコードした信号によってカラーサブキャリヤを移相した信号を選択し、合成しNTSC信号を作る。第2の方式は、赤、緑、青の3色のメモリを持ちY、I、Q信号を作る。I、Q信号はさらにカラーサブキャリヤを直角二重変調し、Y信号と合成しNTSC信号を作る。

このNTSC信号と4.5MHzをFM変調した効果音を合成し、テレビチャンネル周波数の残留側波帯信号に変換し家庭用ビデオゲーム機の出力となる。

3. ゲーム機のソフトウェア

3.1 基本構成

ビデオゲームのソフトウェアの構成は、使用されているマイクロプロセッサの種類やハードウェアの方式に応じて異なるが、ここでは比較的汎用性の高い方式と考えられる図2のグラフィックディスプレイ方式においてCPU、TMP9080Aを使用する場合の例を示す。

図5にソフトウェアの基本構成を示す。初期条件設定ルーチンはリセット入力が増えられた後にアドレス0000Hより実行開始される。このルーチンには一般に次の動作の実行が含まれる。

(a) レジスタ、ワーキングメモリ、入出力ポートの初期設定

図5 ソフトウェアの基本構成

- (b) リフレッシュメモリのクリア
- (c) ゲームの種類、難易度、人数、時間、開始などの指定の入力要求の表示、およびそれらの受付
- (d) ゲーム背景画面の表示

これらの初期条件設定ルーチンの中で、ゲーム開始指定入力を受付けてゲーム実行ルーチンに入る。ゲーム実行ルーチンにおいては、CPU の演算時間が長く、ゲームの時間進行を支配する例（例えば CPU 対人間のチェスゲーム）もあるが、一般にアニメーション表示が中心となるゲームでは、インタラプト入力に加えられる垂直同期信号によってゲーム進行のタイミングを得る方式が多い。

この場合には図5のゲーム実行ルーチンとして示す部分は次のインタラプト入力を受付けるまでの待ち時間の生成を目的としたループとして構成し、実際のゲーム進行は割込ルーチンが行う形とする。割込ルーチンは TMP 9080A と TDP 8228 を使用する場合には RST 7 命令がハード的に生成されるよう構成できるのでこれを利用し、その実行開始アドレスは 0038H となる。この割込ルーチンでは一般に次の動作のタイミングを得る。

- (a) 時間計測
- (b) 移動パターンの表示
- (c) 音響出力の制御
- (d) キーボードやジョイスティック等の入力受付

ゲーム時間経過、得点数等によるゲーム終了条件が満足された場合にはゲーム終了ルーチンへ実行が移ることになる。

以上の各ルーチン実行上で必要に応じて OS プログラムがサブルーチン的にコールされている。

3.2 画像表示ソフトウェア

ビデオゲームのソフトウェア中で画像表示用の部分は大きな割合を占めているわけであるが、リフレッシュメモリが CPU 側から見て通常のメモリと同様なアクセス方式で画像データを書込むだけで良いようにハードウェアが構成されている例が多く、この場合には画像表示用ソフトウェアの基本的な考え方は非常に簡単となる。すなわちメモリへデータを転送する命令（たとえば MOV M, A 命令など）がそのまま1ワード分の画像表示用のプログラムとして使える。

縦横8ビットの文字や図形などのパターンを表示するサブルーチンをフローチャート図6に示す。H-L レジスタに表示位置の上辺のアドレスを、D-E レジスタにパターンメモリ中の連続8ワードにストアされているパターンの最初のアドレスを設定してからコールする。

次に運動体のパターンを1画素単位の任意の位置に表

図6 ブロックパターン表示サブルーチンフローチャート

図7 移動パターン表示サブルーチンフローチャート

示する際には、横方向に連続する複数ワード内でパターンをシフトして表示する必要がある。縦横8ビットのパターンを扱う場合のサブルーチンのフローチャートを図7に示す。Bレジスタにワード内でのシフトするビット数を、H-L レジスタに表示位置の左上辺のアドレス

を、D-E レジスタにパターンメモリー中の連続8ワードにストアされているパターンの最初のアドレスを設定してからコールする。パターンは縦8ビット×横16ビット(2ワード)のブロックとしてリフレッシュメモリーへ転送されることになる。

3.3 音響出力ソフトウェア

音響出力を制御するため、マイクロコンピュータは実時間でパラメーターを出力する。簡単には発振器のON-OFF信号であり、図5の例では信号源、分周比、振幅、デューティレシオの各種パラメーターを指定できる。

信号源として発振器を指定すると、分周比を適当に選択することにより近似的に音階を発生でき、パラメーターをメモリーから必要なタイミングで出力することにより簡単なメロディーを演奏することができる。

信号源として白色ノイズを指定すると、振幅を時間に応じて制御することによりエンベロープを各種形成することができ、デューティレシオ制御と合せて、衝突音、スキークラッシュ音、エンジン音などを発生できる。

例えば、衝突音の場合にはエンベロープのアタックが鋭く最後は減衰する波形を合成すれば良い。また、スキークラッシュ音の場合はエンベロープを比較的なめらかに変化させる。エンジン音の場合はエンベロープやデューティレシオを周期的に変化させて連続音とする。

3.4 乱数発生ソフトウェア

ゲームの重要な構成要素のひとつにランダム性がある。ブラックジャック、競馬ゲームなど程度の差はあってもランダム性の関係するゲームは多い。そこで乱数発生ソフトウェアはゲーム中で重要な位置を占める。

マイクロコンピュータを使用して算術乱数を発生する方法として、大別して次の2種類が利用できる³⁾。

(a) 平方採中法

第 n 番目の乱数 x_n を二乗して x_n^2 を作り(ただし x_n は $2a$ 桁とする)中央部の $2a$ 桁を x_{n+1} とする。以下同様の手順で乱数列を得る。

(b) 合同法

$x_{n-j}, x_{n-j+1}, \dots, x_n$ の1次結合を P で割った剰余を x_{n+1} とする。すなわち、

$$x_{n+1} \equiv a_0 x_n + a_1 x_{n-1} + \dots + a_j x_{n-j} + b \pmod{P}$$

以上の方法で初期値や定数を適当に選ぶと良好な乱数を得られる。その際の初期値の設定はゲームの各種指定入力のタイミングで指定する方法があり、ゲーム機の動作に適している。

なお、実際のゲームにおいてはあまり厳密な乱数性は要求されない場合も多く、一部レジスタ等の数値をもとに乱数の代用とすることもある。

3.5 ソフトウェア開発の方法

一般のマイクロコンピュータ応用機器のソフトウェア開発手法はビデオゲームの場合にも適用できるわけであるが、全体の開発工程のうち大きな割合を占めるデバッグの際のゲームソフトウェア開発における特徴的な点について触れる。

マイクロコンピュータのソフトウェアのデバッグの手段としては、

- (a) 上位コンピュータによるソフトウェアシミュレーター
- (b) 同種マイクロコンピュータによるハードウェアシミュレーター
- (c) ゲーム機と EPROM の組合せ
- (d) ゲーム機と ROM シミュレーターの組合せ
- (e) ゲーム機とインサーキットエミュレーターの組合せ

が利用できる。このうち、クロスソフトウェアである(a)は複雑な演算処理が必要なゲームの場合や短いルーチンのデバッグには有効であるが、アニメーション表示中心のゲームのデバッグには効率が良くない。それはゲーム機の次のような特徴による。

- (i) 画像表示出力まで含めると、出力データが膨大な量となる。
- (ii) アニメーション表示出力や音響出力やコントロール入力や垂直・水平同期信号による割込みは実時間でのチェックが必要である。
- (iii) 画像出力によれば動作状態が一目瞭然となることが多い。

(b)のハードウェアシミュレーターによる方法は、画像発生器、音響発生器、入力装置など、ゲーム機固有のハード部分が多いので、(c)～(e)のゲーム機自身を利用する方法の方がむしろ便利である。(c)のEPROMを使用する方法は最終システムに近い形でチェックが可能であるが、その書込み時間の損失が大きい。(d)のROMシミュレーターとして、マイクロコンピュータ開発システム側からはRAMとしてプログラムの書込みが可能であり、ゲーム機側からはプログラムROMとしてそれを使用できるよう構成することが考えられ、デバッグの効率が高められる。

(e)のインサーキットエミュレーターは実時間の実行を行いながらデバッグできる物であれば、プログラムの誤りの追跡等に有効である。

4. 実 例

前章まではビデオゲームのハードウェア、ソフトウェアについて述べたが、ここでは実例として“ビジコン”(写真2)の動作について簡単に触れることにする。

図 8 “ピジョン”の系統図

写真 2 ピジョン (東芝)

図 8 はブロック図で CPU を軸として本体側 ROM、カートリッジ側 ROM、ワーキングメモリー、リフレッシュメモリーでマイクロコンピューターシステムをなしている。CPU は DMA* 機能を持っているため、DMA 出力要求でリフレッシュメモリーを読み出し、並-直列変換にて画像信号を作り出す。この信号は横 64、縦 32 に分けられており、1 画面 2048 画素構成となっている。各画素は 2 ビット構成となっているため、背景色を含めて 4 色表示が可能である。

本体側の ROM は 2 kB で、その 1 kB は OS プログラムとしてのインタプリタ、基本サブルーチンで、残り 1 kB が内蔵ゲームである。カートリッジ ROM は 2 kB で内容はすべてゲームソフトウェアである。

RAM は演算の途中データ等を蓄えるワーキングメモリー、表示画素に 1 対 1 で対応するリフレッシュメモリー

* Direct Memory Access の略。メモリーと他の部分の間で、CPU のプログラム実行動作によらずに、直接データを転送すること。一般にプログラム実行法と比較して、高速のデータ転送が可能となる。

ー 1, 2 が置かれていて、CPU 側より自由に読み書きできる。

DMA コントローラは画像信号の同期信号を作り出すとともに、画像期間に CPU に DMA 出力の要求信号を出し、この信号に応じて CPU とバスセパレーターは DMA モードに切り、CPU は RAM に対してアドレス信号およびリード信

号を送り、バスセパレーターはリフレッシュメモリーのデータバスを分離し、おのの出力信号を同時に並-直列変換 1, 2 に送出する。

DMA によって得られた画像信号は 2 系統の 8 ビット並列データで、これを並-直列変換器にて 2 系統 (2 ビット) の直列データにする。

この信号はデコーダ部において 4 ラインに分離し 3.58 MHz を移相した信号を選択しカラー信号を作るとともに、論理和によって Y 信号を作る。ここでカラー信号、Y 信号および DMA コントローラ回路よりの同期信号を合成し、準 NTSC 信号を作り、さらに RF 変調回路にて音声 FM 波を加え汎用テレビ受像機で受信できる 1 ch, 2 ch 信号とする。

図 9 メモリーマップ

以上はハードウェアの動作であるが、ここでソフトウェアについて述べる。図9はメモリーマップで管理ルーチンと実行ルーチンよりなるニューメリックインタプリター*と基本サブルーチンは本体側のROMに含まれている。このニューメリックインタプリターは基本16種、変数の組合せの場合40種類のビジコンランゲージ(マクロ命令)よりなり、等価的には16個のゼネラルレジスターを持つことになる。

このゼネラルレジスターはタイマー、音声用 F/F に直結したものもあり、ロード命令のみでタイマー制御、音声制御が可能である。

画像表示は基本的にはグラフィックディスプレイ方式で、アドレス 1100~11 FF, 1300~13 FF の任意の位置に表示できるが、加えて最大8×8ドットパターンをいったんバッファメモリーに登録し、その内容をリフレッシュメモリーに転送し表示する間接法がある。

この間接法による表示は8種類のパターンまで同時に登録できる。この登録には表示場所、色、パターンの大きさ等をパターンの登録とともに登録し、それにはソフトに2~3ステップ必要だが、ゲームのイニシャライズ時に指定し、ゲームプレー中ではパターンの表示、消去、移動がそれぞれ1ステップでプログラム可能なため、プログラムが効率的である。

このように、ビジコンランゲージを用いることによってゲームに必要な基本機能、すなわち画面をクリアする、得点を表示する、キーボードをセンスする等を少ないプログラムステップで行える。しかも、ゲームに必要なほとんどの機能を含んでいるため、プログラムはゲームのルール of アルゴリズムに従って記述することで組立てることができ、ソフトウェアの開発に負担が掛らない。

最後にゲームソフトウェアとして“落書き”、“ボーリング”、“模様書き”、“カーレース”、“加算ゲーム”がゲーム本体に内蔵され、2kBのROM内蔵カートリッジとして“算数ドリル”、“スペースコマンド”、“スポーツファン”、“靈感”、“ギャングラ(I),(II)”がある。

5. む す び

マイクロコンピュータを用いた家庭用ビデオゲームを中心に述べて来たが、システムとしてメモリー容量が4kB以上にも達することが多く、現在家庭で用いられるマイクロコンピュータシステムとして最大規模であり、半導体市場として無視できない存在である。

* ソースプログラムをその構成単位ごとに解釈しつつ実行する言語処理プログラムがインタプリターである。ニューメリックインタプリターは、ソースプログラムが数値すなわちニューメリックコードの形で表記されるビジコンランゲージ用のインタプリターである。

また、話題のホームコンピューターへの最短距離にあり、このことは単にゲームの分野にとどまらず、教育用、ホームマネジメント用として用いられることが期待され、その傾向が現われて来ている。ただ現在は技術的な面ではなく価格的な面、ソフトウェアの面より期待される姿にないが、半導体の進歩を考えると、近い将来充分われわれの期待に応えられる製品が現われると思う。

しかしながら、将来の発展を期待するならば、現在早急に改良すべき点があり、このことが改善されない限り、特に以下の(a)項が解決しない限り大きな期待は持てないこととなる。

(a) ビデオゲーム向きのゲームソフトウェア

現在すでにあるゲームのシミュレーションでは本物のゲームより優れていることは絶対なく、ビデオゲームでなければ実現できないゲームで、しかも魅力的でなければならない。この傾向としてゲーム機が人間と対戦するブラックジャックや、機械の調子に関係なく完全な等確率のルーレット等がある。

(b) 処理速度が遅い

現在のハードウェアでは画像表示に必要な処理速度が充分でなく画素が粗い。このためアニメーションの表現が未熟で迫力がないが、マイクロコンピューターの語長の増加、メモリー容量の増加および素子の動作速度アップを考えると、ここ数年以内に解決されると思われる。

(c) 入出力装置が不十分

現在使用されている入出力装置はゲーム機のために開発された装置ではなく、現在すでに存在する装置を用いている。このため入力とは単純なキーボードが多く、人間の意志を入力するには充分ではなく、また、出力としての表示装置は汎用テレビ受像機を用いているため色表現が充分でない。そのため現在存在する入出力装置のみを考えるのではなく、ゲーム機を中心に考え、必要な入出力装置を開発すべきである。

(a)項はゲーム人口が増加すると解決すると考えられるが、その方向は残念ながら模索中で不明である。(b)、(c)項はすでに解決の方向に向っていて、価格と妥協できる時期が何時かである。

(昭和53年8月24日受付)

【参 考 文 献】

- 1) 八木ほか：民生機器、テレビ誌、**32**, 7 (1978) 597-601
- 2) Super Bug Operation, Maintenance and Service Manual, TM-098, ATARI
- 3) 津田孝夫：モンテカルロ法とシュミレーション、培風館(1969)
- 4) S. Mazor: Applying Microcomputers in Video Games, IEEE Trans., **CE-23**, 3 (1977) 258-261
- 5) Joseph A. Weisbecker: COSMAC VIP, the RCA Fun Machine, BYTE, **2**, 8 (1977) 30-32, 94-97