

# GPSS による実時間システムの解析

吉川 和利\*・平田 威彦\*\*

## Analysis of a Real-Time Computer System for GPSS

Kazutoshi YOSHIKAWA and Takehiko HIRATA

### Abstract

In this paper, the analysis of the dynamic characteristics of an on-line real-time software system is described. An on-line real-time computer system on some assumptions is proposed, and the relation between the message processing and the system performance is analysed. The multi-thread method is introduced in the on-line system, where the disc device is mainly used. The message processing programs are controlled by the monitor, CCP (Communication Control Program), and an on-line control program. The message processed by the message processing program is classified into three types, i.e. the simple response message, the compound response message, and the branch response message.

The model is simulated in detail by GPSS/60. The main objects of this paper are (1) to be helpful to the system construction, (2) to be applicable to the modification of the proposed model flexibly, and (3) to be applicable to many job scheduling methods. Therefore it is clear that we can assign the priority, the routine to be usable or to be reusable, and that we can decide the polling intervals and the overlay structure in this model.

The simulation program was executed by FACOM 230-60, through which we have affirmed that the simulation is accurate and practical.

Finally we found that the modeling of an on-line real-time computer system for GPSS/60 improved the system performance.

### 1. 緒 言

計算機技術は、急速な進歩と応用範囲の拡大に伴って、複雑高度化の一途をたどっており、実時間システムも数多く実現され、各方面で稼動し、それなりの成果をあげている。しかし構成要素の相互関係が多様性をおびて、システム構成の最適化や、ソフトウェアの動特性の把握が困難になりつつあり、ボトルネックが発見されるたびに改良されてきた。そこでシステム設計に当って、与えられた仕様を最良の条件で満足させるには、システムの効率をあらかじめ予測し、最適なシステムを作成することが重要である。

現状の予測手段としては数学的解析<sup>1)</sup>の他にシミュレーション技術があり、ハードウェア・システムの動特性を求めたもの<sup>2)</sup>や操作システムをシミュレートし

たもの<sup>3)</sup>はあるが、ソフトウェア体系をも含めたシミュレーションは種々の不明確な点が多く、マクロなレベルで行なわれていたに過ぎない。

本稿では実時間システムの基礎概念<sup>4)</sup>を全て含むソフトウェア体系、およびそれをカバーするに十分な機器構成を仮定し、ロジカルなモデルを作成しGPSS/60<sup>5)</sup>を用いて、

- (1) 実時間システムの評価、および改良点の把握
  - (2) 最適な機器構成の把握
- を目的としてシミュレートした。

シミュレーション・プログラムに用いた GPSS/60 は次のような特長を有する。

- (1) 優れたタイミング・ルーチンを持つ。
- (2) 複雑な論理機能を有する。
- (3) 簡単に精巧な統計出力が得られる。

\* 大学院工学研究科電気工学専攻

\*\* 電気工学科

- (4) 待ち合せに関する問題に適する。
- (5) 特定の機能を持つ39種類のブロックを組み合わせてモデルを作るだけでよく、取扱いが簡単である
- (6) 汎用言語や他のシミュレーション言語に比べ、デバッグ機能が優れている。

## 2. 実時間システム

### 2.1 システム構成

実時間システムでは信頼性が重んじられ、デュプレックス・システムやデュアル・システムに構成されるか、あるいは衛星計算機を持ったものなどがあるが、通常デュプレックス・システムに構成され、ダウン時にインターフェイス・スイッチを用いて予備の計算機に切り換え、ジャーナル・ファイルを用いて、リラン処理を行なったのちオンライン・タスクをリスタートする。

しかしシミュレーションにおいては定常状態を考慮すればよく、上記のことと合せて決定した機器構成が Fig. 1 である。シミュレーション実行時には Fig. 1 に示した各構成要素の機能および性能が重要である。ここでは容量、転送速度、その他の仕様は標準的な値を用いるが、シミュレーション・モデルにおいては、これらの値を容易に変えることが可能である。次に各装置の仕様を示す。

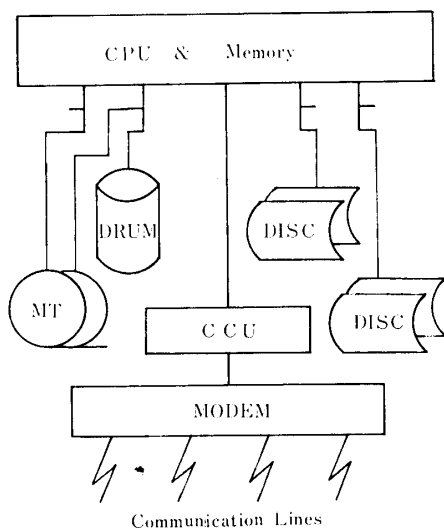


Fig. 1 System Configuration

#### (1) 端末装置および通信回線

シミュレーション・モデルでは端末装置が入力メッセージを発生した時点から、出力メッセージを受けとるまでを把握する。端末装置には種々の形態のものがあるが、分岐応答メッセージをも処理するので、出力装置だけのものが増え、また1通信回

線に対して端末装置2台、あるいは3台のマルチ・ドロップを考える。

通信回線には調歩同期式<sup>6)</sup>を用い、転送速度は50, 200, 400, 1200bpsを適用し、端末装置と通信回線の各転送速度を組み合わせ、入力体系を作成した。

#### (2) モデム

変復調装置で、データ伝送を行なうために端末装置と通信回線の特性を適合させる機能を持ったものである。

#### (3) 通信回線制御装置 (CCU)

モデムとマルチプレクサ・チャンネルとの間に位置し、各通信回線ごとにバッファを接続することができ、種々の端末装置に適応し、最適な通信システムを構成する。バッファには通信回線対応のデディケイト・バッファを用いる。このCCUでは通信回線の状態(すなわち、制御キャラクタの送信状態やエラー状況を通信回線制御プログラム(CCP)の回線状態語にセットし、かつ回線のデバイス・アドレスを設けてCPUに情報を転送する。

#### (4) 処理装置 (CPU)

容量はオペレーティング・システム、CCP、および最大のオンライン・プログラムが収容できる128KB以上を仮定する。実行速度はH-8400<sup>7)</sup>を仮定し、Gibson Mix<sup>8)</sup> 3.6 $\mu$ secとする。

セレクタ・チャンネルは各々コチャンネルを持ち、4チャンネル、8トラックのものとし、転送速度は磁気テープ制御装置、磁気ディスク制御装置、集団ディスク制御装置、あるいはドラム制御装置によるが、それらを十分に許容するものとする。

#### (5) ファイル用入出力装置

##### 1) 磁気テープ装置 (MT)

転送速度120KB/secで、読み書き立ち上り時間10msecである。ファイルの性質上シーケンシャル・ファイルのみであり、ダウン時のエラーリカバ用としてのジャーナル・ファイルとして用いる。

##### 2) 磁気ディスク装置 (Disc)

磁気ディスク装置は後述する集団ディスク装置に比べ転送速度が256KB/secと遅く、また容量が少ないので集団ディスクの補助として用いる。

##### 3) 集団ディスク装置 (Disc)

磁気ディスク装置に比べて転送速度は312KB/sec、記憶容量は29.2KBと有利であり、実時間処理に適しており、ユーザ・ファイルとして用い

る。しかし、シーク・タイムは Fig. 2 に示すように、またサーチ・タイムは回転角度に比例し、一周に25m sec 平均12.5m sec と磁気ディスク装置と同じである。

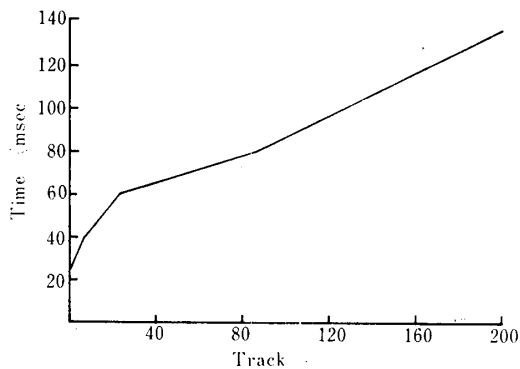


Fig. 2 Seek Time

#### 4) 磁気ドラム装置 (Drum)

転送速度は175KB/sec, 記憶容量は1.6MBで、アクセス・タイムは回転角度に比例し平均10msecである。

## 2.2 ソフトウェア・システム

この実時間システムは磁気ディスクを基本とし、タスクはマルチスレッド方式で処理される。タスクの処理の標準化を計るためタスク対応にインターフェイス・エリアを持ち、ユーザ・ファイルへタスク間の READ/WRITE 命令が競合したとき矛盾を生じないで処理する HOLD/FREE 機能を持ち、効率よく動作するため補助記憶装置への READ/WRITE 命令で最小時間セクタ・チャンネルを占有する OFF-LINE SEEK 機能を持ち、また端末装置のハング・アップを検知する機能を持つ。

システム・プログラムのコントロールのもとで CCP が働く。CCP は、端末装置のデータの発生を予知できないので、データの発生に対してすみやかにサービスを行なえるような中央起動方式を用い、所定時間を経過するたびに指定された端末装置にポーリングを行なう。ポーリングにより端末装置のデータを取得し、CCU のデディケイト・バッファに格納された入力メッセージは、オンライン・コントロール・プログラムのメッセージ読み込み (CMGET) 命令によってコントロールが CCP に移り、CCP が働き、入力メッセージがオンライン・コントロール・プログラムに渡

されインターフェイス・エリアに入力メッセージをセットし、処理を開始する。

オンライン・コントロール・プログラムは CCP, それよりも高位のシステム・プログラムのコントロールのもとで動作するもので、システムの柔軟性、拡張性、経済性などの関係から各機能ごとにモジュールに分割され、情報の交換はインターフェイス・エリアを介して行なう。ハウスキーピング機能を持つものや障害対策用のプログラムはオーバレイ構造にし、定常状態の処理ではターンアラウンド・タイムを考慮して、ほとんど常駐している使用頻度の高いモジュールは、リユーズブルにして、タスクの追い越しによる混乱を避ける。

メッセージ処理プログラムは、前述の高位の3つのプログラムのコントロールのもとでサブタスクとして、メッセージを処理する。これらのプログラムの関係を Fig. 3 に示す。

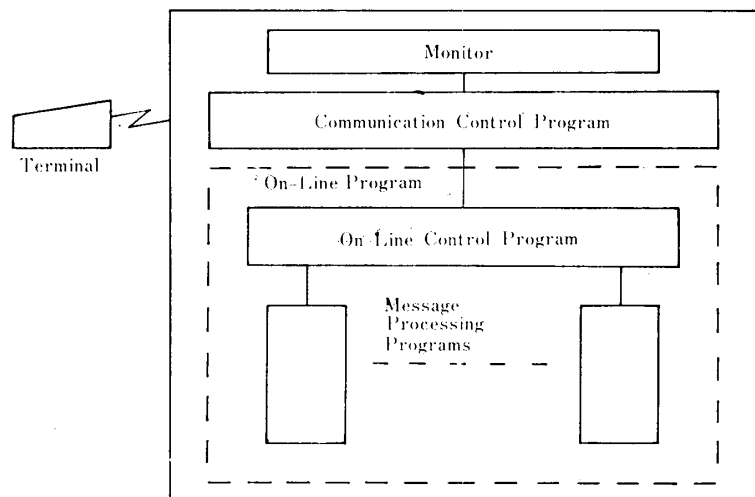
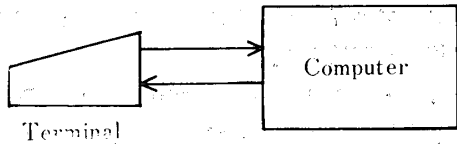
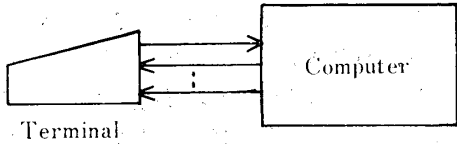


Fig. 3 Software System

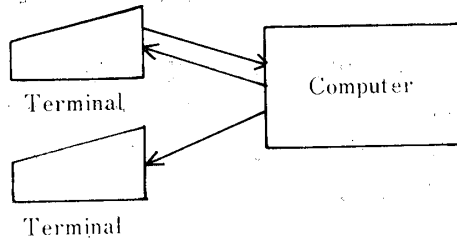
処理は、単応答メッセージ Fig. 4 (a), 複応答メッセージ Fig. 4 (b), 分岐応答メッセージ Fig. 4 (c)を対象にする。単応答メッセージは端末装置からの問い合わせに対して、一つのメッセージ・セグメントを応答するもので、コア上よりインターフェイス・エリアを介して送信する。複応答メッセージは複数個のメッセージ・セグメントを応答するもので、メッセージは出力待ちバッファに一時格納し、1メッセージ・セグメントごとに送信する。分岐応答メッセージは、入力端末装置には前二者の形態で応答し、その上、分岐端末装置にもメッセージを分岐する。当該端末装置が正常に1メッセージ・セグメントを受信するごとに ACK が



(a) Single Response Message



(b) Compound Response Message



(c) Branch Response Message

Fig. 4 Typical Messages

返送され、1メッセージ・セグメントの完了として出力待ちルーチンで処理される。

初期状態では、タスクは問い合わせモードであるが、出力待ちバッファを介するメッセージは指令モードにモード変換しなければならない。すなわち、単応答メッセージではモード変換は起こらないが、複応答メッセージ、分岐応答メッセージでは最低2回のモード変換が必要である。

ユーザ・ファイルには、即応性とデータの性質を考慮し、ランダム・アクセス・ファイルか、インデックス・シークエンシャル・ファイルを用いる。

### 2.3 オンライン・プログラム

実時間システムにおけるオンライン・コントロール・プログラムの位置は Fig. 2 に示すとおり、システムプログラム・CCP と、メッセージ処理プログラムの仲介のための制御部分を吸収することにより、メッセージの授受、ならびにメッセージ処理の多重化を実現したので、メッセージ処理プログラムが、データの処理に徹することができ、能率的である。またプログラムの作成および改造を容易にするため、機能により複数のモジュールに細分化する。そのモジュールの名称および機能の概略を以下に示す。

#### (1) メッセージ読み込みルーチン

CMGET 命令を発し、CCP より得たメッセージ

を分析、チェックし、待ち行列エリアに登録する。正常入力メッセージの場合は入力ジャーナル・ルーチンにコントロールが渡されるが、ACK では出力待ちルーチンにコントロールが渡される。

#### (2) タスク起動ルーチン

タスクへのプライオリティの割当てを行なう。すなわち、サブタスクとして入力ジャーナル・ルーチン、ランチ・ルーチン、メッセージ処理ルーチンで処理される場合は、メイン・タスクで処理されるよりは低いプライオリティが割付けられる。タスクへのプライオリティの割付けの方法には2種類ある。一つには、プライオリティの高い方から空いているものを割付ける方法と、割付けられている最低レベルのタスクが終了しない限り、それ以上の処理が終っても他のレベルを割付けずに待ちにする方法で、後者はタスクの洗み込みを防ぐのに有益である。

#### (3) 入力ジャーナル・ルーチン

タスク起動ルーチンによって割当てられたプライオリティをもったタスクが起動され、このルーチンにコントロールが渡されて処理される。このルーチンはダウン時のエラーリカバに対応するためのもので、テープ・ファイルに端末アドレス、回線アドレス、入力メッセージ内容等をジャーナル・ルーチンを介して記憶する。ジャーナル・ファイルには3つの磁気テープを用いるが、その内の1つは交替用である。

#### (4) ランチ・ルーチン

前ルーチンの処理が終り、コントロールがこのルーチンに渡ると、インターフェイス・エリアのメッセージの種類を判定し、対応したメッセージ処理プログラムへコントロールを渡す。

#### (5) メッセージ処理プログラム

メッセージ対応に作成されるもので、特異なメッセージの新設以外はこのプログラムのみを変更すればよく、新設によりオンライン・コントロール・プログラムに登録するだけでよい。メッセージの処理で、ユーザ・ファイルの更新を行なうとき、必ず読み込み前後のファイルの内容をジャーナル・ルーチンを介して記憶する。また出力メッセージを作成し、インターフェイス・エリアに格納する。長応答メッセージや分岐応答メッセージでは、SEND 命令を出すことにより、SEND ルーチンを介して出力待ち行列に登録され、出力メッセージは一時出力待ちバッファに格納される。同時に SEND ルーチ

ーンでは、ジャーナル・ルーチンを介して、出力メッセージを出力ジャーナルに記憶する。

(6) タスク終了チェック・ルーチン

メッセージ処理プログラムでの処理完了を検知したのち、後処理をし、完了チェック・ジャーナルをジャーナル・ルーチンを介して記憶する。

(7) タスク終了ルーチン

メッセージ出力 (CMPUT) 命令でメッセージを送出する。単応答メッセージでは CMPUT 命令によりインタフェース・エリアより当該端末装置に出力メッセージを送出し、正常に受信されれば ACK が返り、出力待ちルーチンを經由して、このルーチンで全ての処理を終了する。複応答メッセージでは、最初の CMPUT 命令により問い合わせモードより指令モードに変換され、ACK が返ってきたのち複数回メッセージ・セグメントを CMPUT 命令でインタフェース・エリアより送出し、最後の ACK を受信したのち、初期状態である問い合わせモードに変換する。分岐応答メッセージでは、分岐端末装置を起動し、モード変換を行なう。

(8) 出力待ちルーチン

メッセージ読み込みルーチンで ACK が読み込まれると、このルーチンにコントロールが渡り、処理される。単応答メッセージでは ACK ジャーナルをとり、コントロールをタスク終了ルーチンに渡せば処理は終了するが、複応答・分岐応答メッセージではメッセージ・セグメントを送信することに

ACK ジャーナルをとり、次にタスク終了ルーチンが端末装置へメッセージ・セグメントを送信可能なように出力待ち行列を見、磁気ディスク上の出力待ちバッファより、コア上のインタフェース・エリアに出力メッセージを転送する。最後の ACK を受けとった時は単応答メッセージと同様である。

(9) SEND ルーチン

複応答・分岐応答メッセージを処理中、メッセージ処理プログラムより SEND 命令が発行されると、このルーチンが呼ばれ、メッセージ処理プログラムで作られて、インタフェース・エリアに格納されている出力メッセージの出力ジャーナルを取り、次に磁気ディスク上の出力待ちバッファに蓄える。

(10) ジャーナル・ルーチン

入力ジャーナル、ファイル・ジャーナル、出力ジャーナル、タスク終了チェック・ジャーナル、ACK ジャーナルを取ろうとするとき、このルーチンにコントロールが渡され、磁気テープ、あるいは磁気ドラムにジャーナルが取られる。

これらのルーチンのほかに、実時間システム稼動時に必要なイニシャライズ処理ルーチン、オンライン系エラーに対するオンライン・エラー処理ルーチン、全ての処理の終了時に必要なクローズド・ルーチン、エラー処理ルーチンなどが考えられるが、定常状態では不要である。この各ルーチンの関係を Fig. 5 に示す。

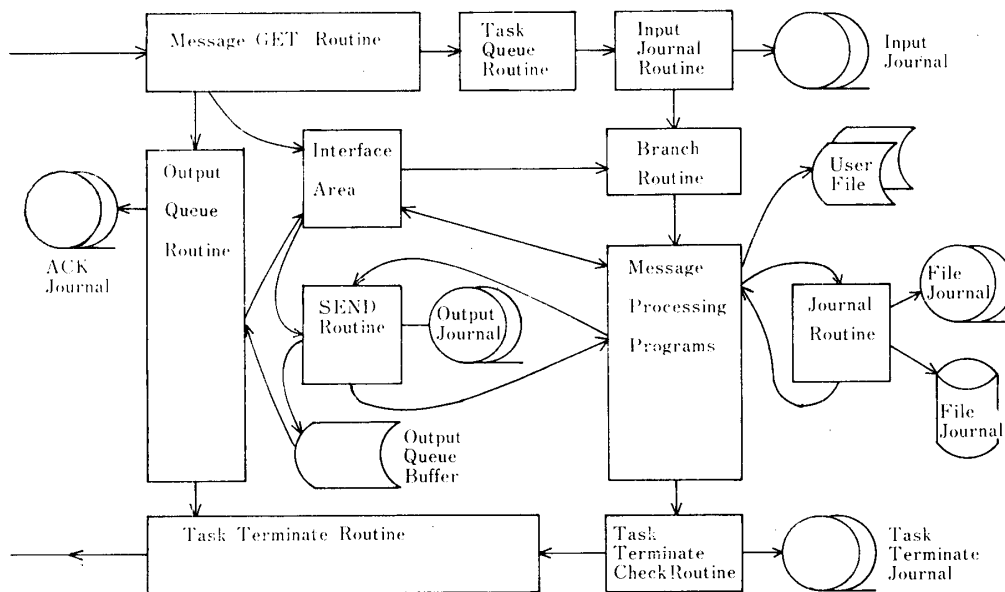


Fig. 5 On-Line Program

### 3. モデルの設定

#### 3.1 待ちを考慮したモデル

高トラフィックにも耐えるように各ルーチンをリユ-ザブルにし、またコア容量をより有効に利用するためオーバレイ構造を用いた。したがって、各ルーチンに待ち行列が構成され、各ルーチンの使用効率、使用時間、入力間隔、ひいてはターンアラウンド・タイムに大きな影響を与える。これらの要素が間接的にプログラム構造、処理内容、処理形態、機器構成、入出力端末装置台数、およびファイル構造の影響を与える。したがって、このシミュレーション・モデルでシミュレートすることによって、ボトルネックを発見し、それに影響を与える要因を見付け出し、変更していくことにより最適な構成が可能となる。

前述の各ルーチンと、これに待ちを考慮した概略的な関係を Fig. 6 に示す。端末装置に待ちをとらなかったのは、分岐応答メッセージによりハング・アップが起らないようにするためである。

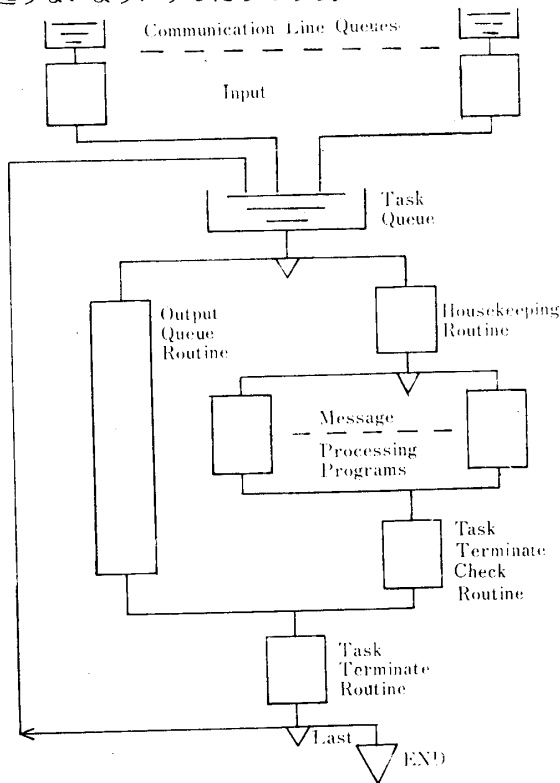


Fig. 6 Simulation Model

このモデルでは、各形態のメッセージが共通に処理される。この流れを以下に示す。

##### (1) 単応答メッセージ

端末装置 → 通信回線 → CCP → メッセージ読み込みルーチン → タスク起動ルーチン → 入力ジャーナル・ルーチン → ブランチ・ルーチン → メ

ッセージ処理プログラム → タスク終了チェック・ルーチン → タスク終了ルーチン → CCP → 端末装置 → CCP → メッセージ読み込みルーチン → 出力待ちルーチン → ACK ジャーナル → タスク終了ルーチン

##### (2) 複応答メッセージ

端末装置 → 通信回線 → CCP → メッセージ  
② ↑

読み込みルーチン (ACK か?) No. → 入力ジャーナル・ルーチン → タスク起動ルーチン → ブランチ・ルーチン → メッセージ処理プログラム → タスク終了チェック・ジャーナル → タスク終了ルーチン (最初か?) YES → 指令モード → ②  
No → 終わりか? No → メッセージ送出 → ②  
YES → 問い合わせモード → 終了

① → 出力待ちルーチン → ACK ジャーナル → 最終 ACK か? No. → 出力待ちバッファより出力 → ③  
YES → ③  
力メッセージ・セグメントをインターフェイス・エリアへ → ③

(3) 分岐応答メッセージ

分岐端末装置へ起動指令を出し、端末装置より ACK が返送されるだけで他は複応答メッセージの処理と同じである。

③ → 出力待ちバッファより出力 → ③

##### (3) 分岐応答メッセージ

分岐端末装置へ起動指令を出し、端末装置より ACK が返送されるだけで他は複応答メッセージの処理と同じである。

#### 3.2 GPSS モデル

GPSS/60 は他のシミュレーション言語と共通したエンティティ、アトリビュート、アクティビティという考えを持つ。エンティティはシミュレートするシステムの構成要素であり、アトリビュートはエンティティの属性である。またアクティビティはエンティティの動作である。特にシステムの構成要素に対応するエンティティは重要である。次に各エンティティの役割りを、本稿で述べるシミュレーション・モデルに対比して述べる。

##### (1) ファシリティ・エンティティ

サービス単位が1の設備を示し、一度に一つのトランザクション(メッセージ)が使用できる。このシミュレーションではファシリティ1から35までを用いて CPU, チャンネル, 各デバイスの平均使用率, 使用トランザクション数, 平均使用時間などの統計量をとる。

##### (2) ストレージ・エンティティ

ストレージは多数個のサービス可能な設備をシミュレートする。指定したキャパシティ以上にはサービスできず、待ちとなる。ストレージで通信回線、サブタスク、ユーザ・ファイル、出力待ちバッファへの READ/WRITE ルーチン、オーバレイ・リジョン、出力ジャーナル・ルーチンを表わし、これらの平均使用率、平均使用量、最大使用量、総使用量、平均使用時間の統計量をとる。

### (3) テーブル・エンティティ

ヒストグラム作成のため使用し、TABULATE ブロックと TABLE 定義カードにより、下限、間隔、個数が指定され、指定された値がどの範囲に落ちたかが計算される。テーブルは 1 から 62 まであり、メッセージの応答時間、サブタスクの処理時間、メッセージ処理プログラムの処理時間、出力待ちルーチンの処理時間、ターンアラウンド・タイム、およびメッセージの種類ごとの統計、メッセージの入出力間隔、出力待ちルーチンの入出力間隔などを得る。

### (4) バリアブル・エンティティ

GPSS には SNA (Standard Numerical Attribute) があり、各エンティティに関する情報や、システム情報の参照や更新を行なうとき利用するので、バリアブル・エンティティはこの SNA を使って演算子を用いて計算を行なう。すなわちメッセージ長、メッセージに対する端末装置番号の割当て、受信時間、入力ジャーナルに要する時間、送信回数、送信ブロック数、1 ブロックの送信時間、ファイル・ジャーナルの書き込み時間、出力ジャーナルの書き込み時間などを計算する。

### (5) ファンクション・エンティティ

ファンクションの機能は、2 変数間の関数関係を 5 つのタイプに示す。これにより種々の乱数の発生、メッセージ番号、端末装置番号、通信回線番号、入力メッセージ長、入力速度、オーバレイの有無、ファイル番号、ディスク番号、テーブルの割当てを行なう。

### (6) キュー・エンティティ

キューはモデル中にできる待ち行列についての統計をとるためのもので、QUEUE ブロックに入ったトランザクション総数、平均待ち時間、0 タイムで待ちを通過したトランザクション数などの統計量が TABULATE ブロックと QTABLE ブロック定義カードを利用して得られる。キュー番号は 1 から 116 まであり、ストレージ・エンティティ、ファ

シリティ・エンティティの各設備に対応している。

### (7) セイブバリュ・エンティティ

数値を蓄えるのに用い、サブタスクの優先度、ジャーナルのブロッキング・ファクタの換算、ポーリング回数、チャンネル番号、入出力速度、ポーリング回線数をセイブバリュ 1 から 30 までを用いて一時記憶する。

上述のエンティティにいくつかの命令ブロックが対応し、他にトランザクション・エンティティが多く命令ブロックを含む。これらを Fig. 7 に示すマッチング・フロー・チャートに組み込み、シミュレーションプログラムを作成した。メッセージに対応するものが、GENERATE ブロックより、あらかじめ調べた分布に従ってトランザクションとして発生され、実際の実時間システムに対応する各機能をもったブロックを通過し、最終的には TERMINATE ブロックで消滅し、1 つのメッセージの処理の終りを示す。

実時間システムの各種の時間的統計量をとるために MARK ブロックと MATCH ブロックを用いる。MARK ブロックでメッセージにあたるトランザクションの発生時からの経過時間、あるいは任意の時点からの経過時間を相対タイムを利用して計算することができる。MATCH ブロックにはトランザクションは無条件に入ることができるが、SPLIT ブロックによって新たに作られた対応するトランザクションが指定された MATCH ブロックに入るまでは待ちになる。従ってトランザクション間の同期がとれ、流れを簡明にし、かつコア容量を節約できる。また MATCH ブロックの次に TABULATE ブロックを入れれば、相対タイムを利用して、MARK ブロックからこのブロックに到達するまでの経過時間を測定でき、また一定時間に何個のトランザクションが入ったか、あるいは入力間隔についてのデータが得られる。この特性を活用して実時間システムの関係を示したものが Fig. 7 である。Fig. 7 の MATCH ブロック中のタイムの内容を次に示す。

$t_0$ ; サブタスクの処理時間

$t_1$ ; 出力待ちから最初の応答までの時間

$t_2$ ; 出力待ちルーチンでの処理時間

$t_3$ ; 端末装置からの応答を処理する時間

$t_4$ ; メッセージのレスポンス・タイム

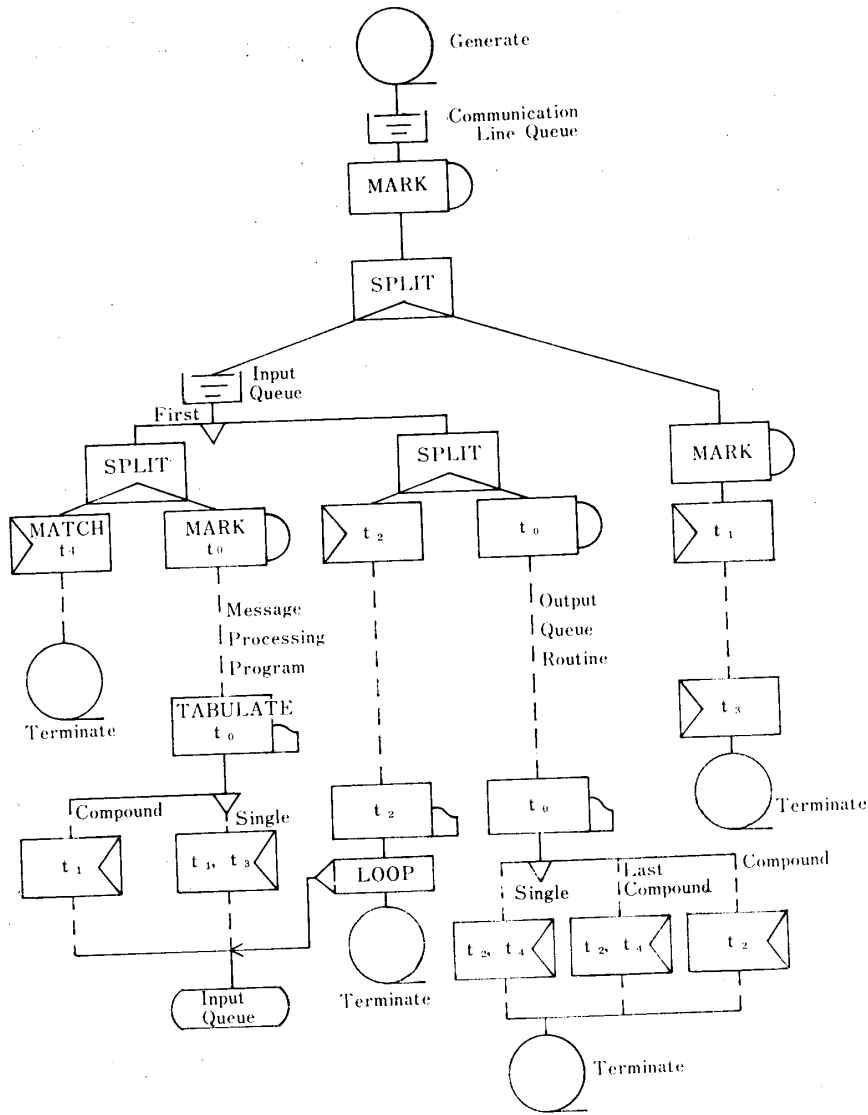


Fig. 7 Matching Flow-Chart of Model

3.3 ポーリング・モデル

1 通信回線に複数台の端末装置を接続するマルチ・ドロップを考えたが、マルチ・ドロップはトラフィック量がそれほど多くないときに適している。この構成では、端末装置に送信すべき情報があるか否かを中央装置が監視するために、CCPがCCPのテーブルに登録されている通信回線番号と端末装置番号で、各端末装置を順次一定時間間隔ごとにポーリングする。もし送信メッセージがあれば、テーブルにビジとして登録され、全処理が終ってフリーになるまでポーリングされない。ポーリング間隔は任意にパラメータ・カードで指定できるし、端末装置の構成の変化も同様にパラメータ・カードで変更できる。

固定メッセージ長での最適なポーリング間隔は求め

られている<sup>6)</sup>が、種々の長さのメッセージが混合しているばあいのポーリング間隔は、端末装置へのトラフィック量を考慮したシミュレーションによって求める以外にはない。すなわち、トラフィック量を測定して、このモデルでシミュレートすれば、最適の端末装置の構成が求まり、また各端末装置のポーリング間隔も決定できる。

ポーリング・モデルを Fig. 8 に示す。ここではテスト・ランのためポーリング間隔を1秒と2秒のものにしているが、ファンクション・エンティティにより任意に定めることができる。このモデルを Fig. 7 に示したモデルと同時にシミュレートすることにより、モニタ、CCPのコントロールを競合し、システム・オーバーヘッドとなって表われる。



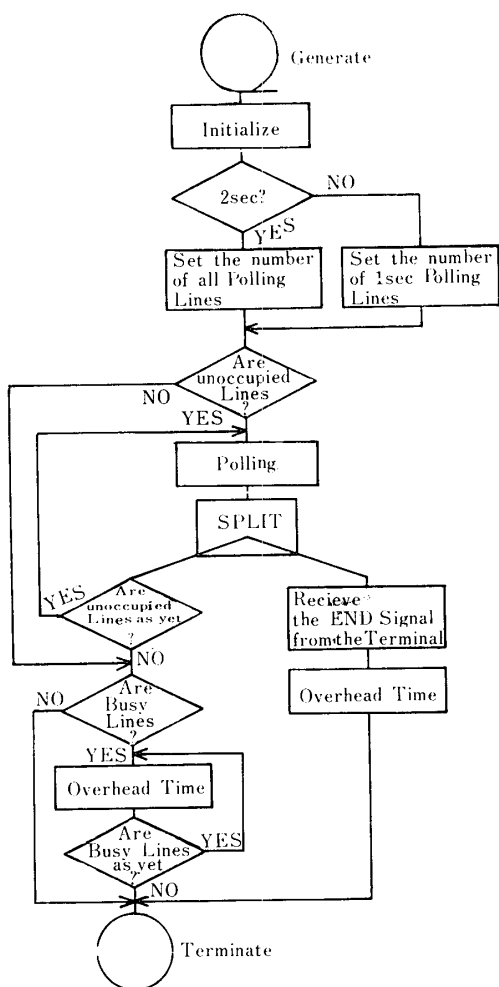


Fig. 8 Polling Model

#### 4. 結 言

実時間システムのシミュレーションについて研究し、ソフトウェア面からの動特性を評価し

- (1) システム構成のたすけとなる
- (2) あらたに変更されたものに対して柔軟に適應できる
- (3) 各種のジョブ・スケジューリングに適應できる

ことを目標として作成したものであり、GPSS/60の

特性を生かして、システム構成と一対一に対応させたので、モジュールの機能の変化、機器構成の変化、仕様の変化には対応したところだけを変更すればよく、柔軟性に富んでいる。しかし汎用性は少し欠ける点もある。スケジューリングでは優先度の割当て、ユーザブル・リユーザブルの割当て、オーバレイの構成、ポーリング間隔などが評価可能である。

GPSS/60は優先度を0から7までしか取れず、それ以上はPREEMPTブロックを利用しなければならず、プログラムが複雑になる。またソフトウェア・タイマを用いているし、ブロックの進行に伴ない各チェーンをサーチする形式であるために実行時間が長い。実行途中で打ち切っても中間結果が出力されない欠点もある。

シミュレーション・プログラムはFACOM 230-60を使用して実行し、プログラム動作を確認し、シミュレーションを終了した。この結果きわめて正確、かつ実用的なものであることが示された。

最後にプログラム作成にあたって協力をいただいた小林佑光（電気工学科助手）、溝尻由美夫（大学院電気工学専攻）、山崎敏明（昭和47年度卒論生）古谷宣子、二木和世（電子計算機室）、木野真澄（電気工学教室）の諸氏に謝意を表す。

#### 参 考 文 献

- 1) H. Kobayashi: Proc. 1st USA-Japan Computer Conference, 130 (1972)
- 2) 相磯秀夫: 電気試験所研究報告, 703 (1969)
- 3) 淵 一博: 電気試験所彙報, 34, 151 (1970)
- 4) J. Martin: "Design of Real-Time Computer Systems" Prentice-Hall INC. (1967)
- 5) FACOM 230-60 GPSS/60 解説書 (1), 富士通マニュアル
- 6) 大野 豊 "オンライン・リアルタイム・システムの設計" 産業図書 (1970) P. 37
- 7) H-8500/8400/8300処理装置, 日立マニュアル
- 8) 石田晴久: 情報処理, 13, 333 (1972)

(昭和47年12月15日受理)