

【システム解説】

鉄骨生産システムの自動運転システム

Automation System for Automated Manufacturing System
for Steel Frame Structures北島 彰夫*
*Akio KITAJIMA*松原 哲朗**
*Tetsuro MATSUBARA*上田 仁***
*Masashi UEDA*坪山 雄二****
*Yuji TSUBOYAMA*浦辺 裕二*****
Yuji URABE

1. まえがき

複数のパソコンから公衆電話回線によってジョブ転送をうけ、これを自動的に処理して、その結果をパソコン側に返すことを目的とする、ミニコン上の自動運転システムを開発したので報告する。これは鉄骨生産システムの外販（商品名：PROSSESS）にともない、センターミニコン側の運用上の必要から開発されたものである。

著者らはすでに、SおよびSRC構造のビル鉄骨を対象とし、梁伏図、軸組図などの一般図から、部材ごとの工作図、製作資料までを一貫して処理する鉄骨生産システムを開発し運用している。このシステムでは、パソコン上でデータ入力や図面、資料などの編集・出力の作業を行い、ミニコンで一部のバッチ処理をするという混合型のハードウェア構成をとっている。

そもそも、ミニコンに数個の端末を接続してタイムシェアリングで用いる形態よりも、比較的安価なパソコンを数多く用いてそれぞれ独立して物件を処理する形態のほうが、低コストのシステム構成をすることができる。実際の作業者にとっても、常に一定の時間で応答をえることができるうえに自分の責任で（任意に）、マシンの稼働、データの維持、管理などを行うことができるなど、鉄骨業界に適しているということができる。従来、パソコンに接続可能な周辺記憶装置の記憶容量があまり大きくなく、これに大量の図面情報などのデータを保存することができないという問題があったが、近頃40MB、150MBといった大容量のハードディスクがあらわれてこの問題はほぼ解消している。本鉄骨生産システムでは、パソコンで大部分の作業が処理されるが、残念ながら16ビットパソコンのメモリー空間が狭いために、一部のサブシス

テムはパソコンへの移植が困難で、ミニコン上で処理されている。図-1に、実際の処理のフローを示す。

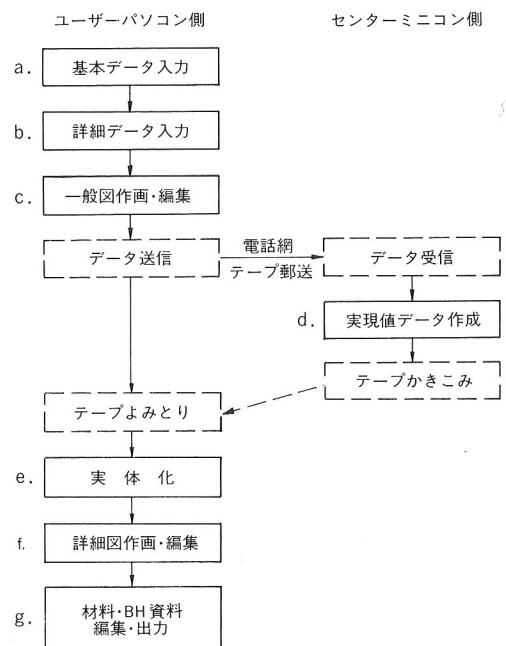


図-1 物件処理フロー

具体的には、作業者はまずパソコンでcまでの作業をしたあと、ミニコンにファイル転送する。このファイルをもとにミニコンでdの処理をして、その結果をカセットテープに収めてパソコン側に輸送する。こののち、作業者はパソコンでe以降の作業をする。従ってファイル転送の関係をのぞけば、ミニコンとパソコンのジョブはそれぞれ独立に実行されることになる。

このようにパソコンに比重のかかっている処理形態では、ミニコンが同時に数個の物件のジョブを並行処理す

*川田工業技術本部生産企画部部長 **川田工業技術本部生産企画部課長 ***川田工業技術本部生産企画部CAD/CAM課
****(株)システムエンジニアリング鉄構部CAD/CAM課 *****株)システムエンジニアリング技術部技術一課

る能力をもつこともあって、ミニコン1台である程度多数のパソコンの物件の処理をサポートすることができる。そこで処理センターを設置して、センター内にミニコンをおき、遠隔地のパソコンは公衆電話回線を通して、このセンターミニコンを共同利用している(図-2参照)。

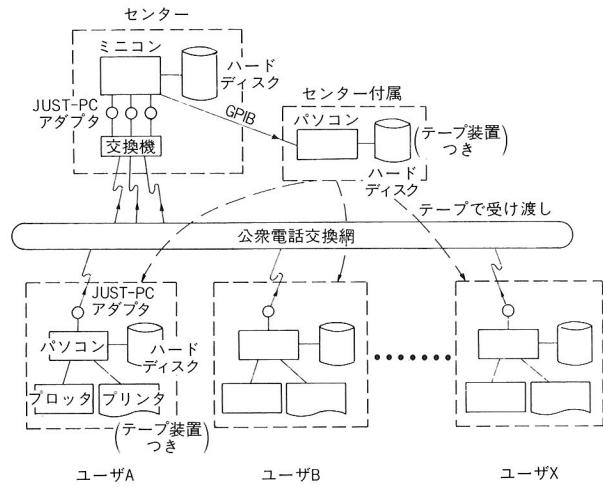


図-2 システム構成

処理センターにとっては、多数のユーザから不定期的に着信する回線を不斷に監視し、ファイル受信後手作業でジョブを起動するのでは、ジョブの前処理、実行状況の監視、後処理、ログ記録などで繁忙となり負担が大きいし、その場合、センターにオペレータがない時にはセンターミニコンを利用できないなどユーザにとっても不都合がある。そこで、ミニコン上で自動運転システムを常時動かしておき、このシステムが人間にかわって、前述の処理をすることにした。

なお、ユーザに送るデータは相当大きい(設計重量300tonの物件でも数MBになる)ので回線を使うと通話料金がかさんでしまうため、テープで受け渡しをする。

2. JUST-PCによるファイル転送

本システムでは、JUST-PCを使ってユーザからセンターハードウェアへファイル転送する。JUST-PCは、郵政省告示による公衆電話交換網上のプロトコルの国内実装仕様で、パソコン間の通信装置の推奨通信方式となっていて、次のような特徴をもつ。²⁾

- ① 2,400/4,800bps(半2重)と高速である。
- ② エラーフリーなプロトコルを実装している。
- ③ 通信処理をアダプタとして独立させ、コンピュータの負担を軽減した。
- ④ 低価格である。

自動発信、自動着信の機能もあり、RS-232C端子とアダプタつき電話機を接続して使用する。

また、パソコン側では作業者が画面上の送信メニュー

を選択するだけで、送信データの準備、センターのコール、送信、電話の切断など全ての処理を自動的に行う。図-3に通信処理の単純化したフローを示す。(ただし、ユーザ確認のための手順は省略している。)

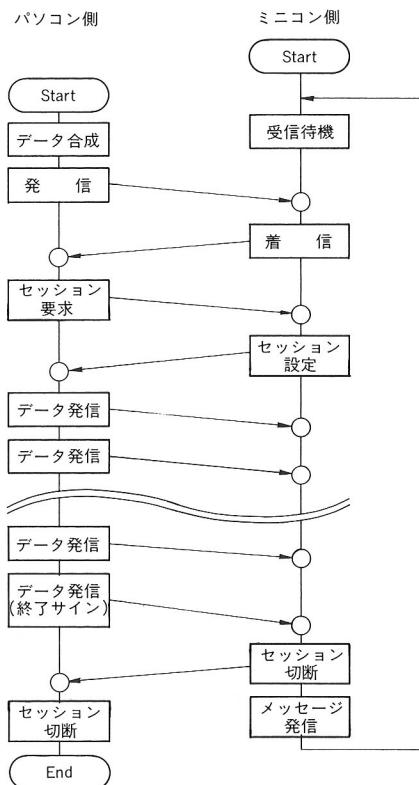


図-3 通信処理関連図



写真-1 センターミニコンとJUST-PCアダプタ

送信データはSYS,FRMの2つのファイルを主な構成要素とする。SYSは自動運転システム自体が使うデータ(ユーザ名、ユーザ工番名、パスワード等)を内容とする。FRMは鉄骨生産システムの全データファイルを合成したものである。送信データは他にREQ,RNGのファイルを含んでいてもよい。このうちREQはSYSの拡張の性格をもち、現在回線テスト目的で送信するデータを区別するために使っている。RNGは柱、梁の処理範囲節番号を内容とするきわめて小さいファイルで、本来はFRMの一

部をなすべきものだがこれを分けることによって、一度センターに送信したデータと全く同じデータで処理範囲だけをかえて実行する場合に、FRMを再送しないで済むようになる(図-4参照)。

またFRMを構成する各ファイルは、実は固定長レコードの表形式のテキストファイルで、実際には空白文字が多いので、空白文字を圧縮して送信する。これによって送信データを平均して約30%圧縮できた(図-5参照)。



図-4 送信データ構成

```
+ 1:1C1A :1:1-C1 :2:2:0 :0 :2:
+ 2:2C1A :2:2-C1 :2:3:0 :0 :2:
+ 3:3C1A :3:3-C1 :2:3:0 :0 :2:
+ 4:4C1A :4:4-C1 :2:4:0 :0 :2:
+ 5:1C2A :1:1-C2 :0:2:0 :0 :2:
+ 6:2C2A :2:2-C2 :0:3:0 :0 :2:
+ 7:3C2A :3:3-C2 :0:3:0 :0 :2:
+ 8:4C2A :4:4-C2 :0:4:0 :0 :2:
+ 1:1C1A :1:1-C1 :2:2:0 :0 :2:
+2:2C1A:2:2-C1:2:3:0:0:2:
+3:3C1A:3:3-C1:2:3:0:0:2:
+4:4C1A:4:4-C1:2:4:0:0:2:
+5:1C2A:1:1-C2:0:2:0:0:2:
+6:2C2A:2:2-C2:0:3:0:0:2:
+7:3C2A:3:3-C2:0:3:0:0:2:
+8:4C2A:4:4-C2:0:4:0:0:2:
```

図-5 データの圧縮前(上)と圧縮後(下)

このシステムでは送信データは全てテキストファイルだが、JUST-PCはバイナリファイルでも自由に転送することができる。テープやディスクのフォーマットに関する煩わしさもなく、異機種間転送に便利である。ただ、パソコン側のOS(MS-DOS)とミニコン側のOS(UNIX)との間でファイル仕様に若干の差異がありその調整が必要だが、処理効率を考えて、ミニコン側で調整している。

3. 自動運転システム

自動運転システムは、FRMを分解・復元して、これをデータとして鉄骨生産システムのジョブ(以降、単にAPとよぶことにする)を起動する。ジョブは、ユーザ名とユーザ工番名で識別される。

自動運転システムは、各ジョブに対して、ユーザ名とユーザ工番名とでユニークとなるような工番ディレクトリをわりつけ、そのディレクトリ下でジョブを実行させる(図-6参照)。APはカレントディレクトリの下位のファイルに対してのみ読み書きするので、これで各ジョブの独立が保障される。

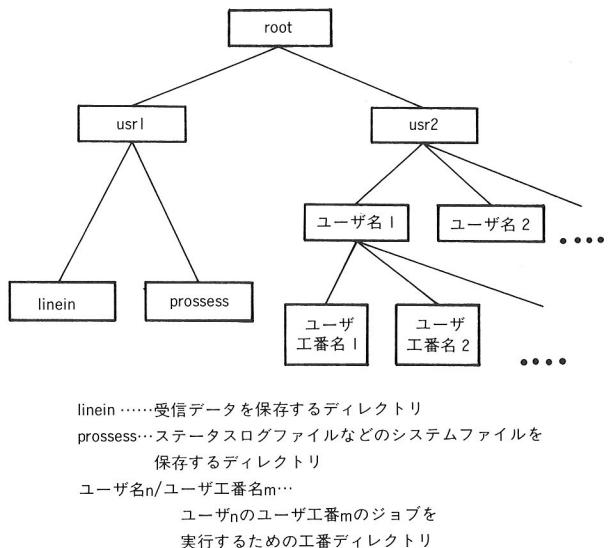


図-6 実行時のミニコン上のディレクトリ構造図

処理結果は高速のGPIBによってパソコンに転送し、パソコンでテープに書きこまれる。その後ミニコン上では、ログファイルを除いて、そのジョブに関連する全てのファイルを消去する。パソコン上で適当なテープオートローダーがないため、図-7でf以降の処理は自動化されていない。

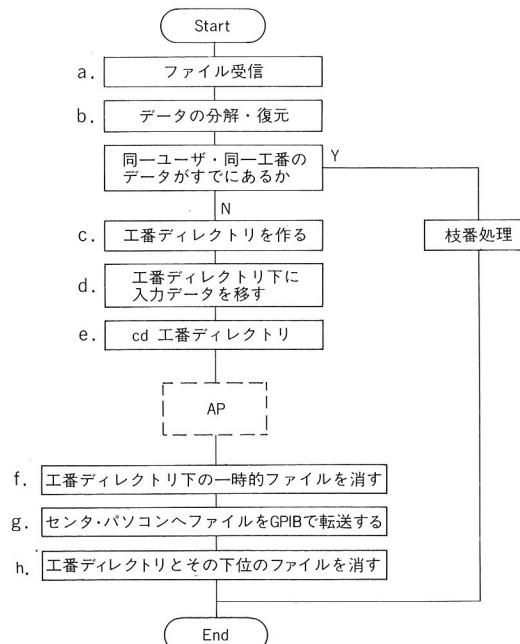


図-7 自動運転システムのジョブサイクル

受信データのユーザ名、ユーザ工番名が、現在ジョブサイクル内にあるジョブの1つと一致する場合、そのデータのジョブを自動的には起動せず、ログファイルに枝番のデータとして記録しておく。枝番処理されたジョブは、後述の再スタート機能を使って、ミニコン側のオペレータが起動することができる。このようにした理由は、同一工番のデータが送信されてくる理由が、処理範囲を分割したケース、データ修正による再実行のケース等さまざままで画一的に処理することができないためである。ただし、前のジョブのジョブサイクルが終了したあとの受信は通常の処理がなされる。

4. ミニコンのタスク編成

ミニコンには3台のJUST-PCアダプタを接続し、それに対応して3つの受信タスクを常駐させる。これによって同時に3ユーザからのデータ送信を受理することができる。受信タスクは、平常は受信待ちの状態にあり、ユーザからの送信を検知すると、ユーザ登録ファイル(pswd)を参照して正当なユーザからの送信であることを確認したあと、受信したデータをハードディスクに書きこむ。受信タスクは、受信終了後も消滅せず、再び受信待ちの状態になる。

ミニコンに制御タスクを常駐させる。制御タスクは受信タスクから1セッションのデータ受信の終了を知られると子タスクを通じてAPタスクを起動する。(APタスク終了時のログファイル更新などの処理は子タスクによって行われ、APタスク自体は自動運転システムに関連する処理は全くしない)。ただし、データ受信終了時に、すでに4つ以上のAPタスクが実行中である場合にはただちには子タスクを起動せず、いずれかのAPタスクが終了した時点で、制御タスクが子タスクを起動する。同時に

実行可能なAP個数を最大4つとする理由は、これをこえるとスループットが悪くなるのと、AP実行中にだけ作られる一時的ファイルの占める容量が大きくなりすぎること、同時にオープン可能なファイル数に関するOS上の制限をこえる恐れがあるためである。

3つのアダプタでの受信、4つ(以内)の子タスクの終了は非同期に生じる事象であるが、これらはメッセージ通信によって制御タスクに通知される。

APタスクを除く各タスクは、共有メモリ上のタスクコントロールテーブル(TCT)を参照・更新し、ステータスログファイルにログ記録する。TCTはジョブサイクル内にあるタスクの一覧表で、各ジョブのユーザ名、ユーザ工番名、どの実行状態にあるかを示すステータス等を内容とし、ステータスログファイルはこれを発生時刻順に記録したものである。複数のタスクが同時にテーブルを更新することのないよう、各タスクはテーブル更新時には、セマフォによって共有メモリをロックする。

メッセージ通信、共有メモリ、セマフォはUNIXのSYSTEM-Vで使うことのできるプロセス間通信・同期のための仕組みで、これを操作するためのC言語の関数が提供されている。

図-8において①、②等は標準的な場合に事象の生じる順序で、次のようなものとなっている(なお、n1、n2は共有メモリ上の数値で初期値は0である)。

① (受信タスク) 受信タスクは、データ受信開始時に、そのジョブのためにTCT上にエントリを確保し、ログを出力する。受信が終了すると、n1を1加算し、TCTを更新、ログ出力し、タイプ1のメッセージを送信する。その後、受信タスクは受信待ちとなる。

②、③ (制御タスク) 制御タスクはタイプ1また

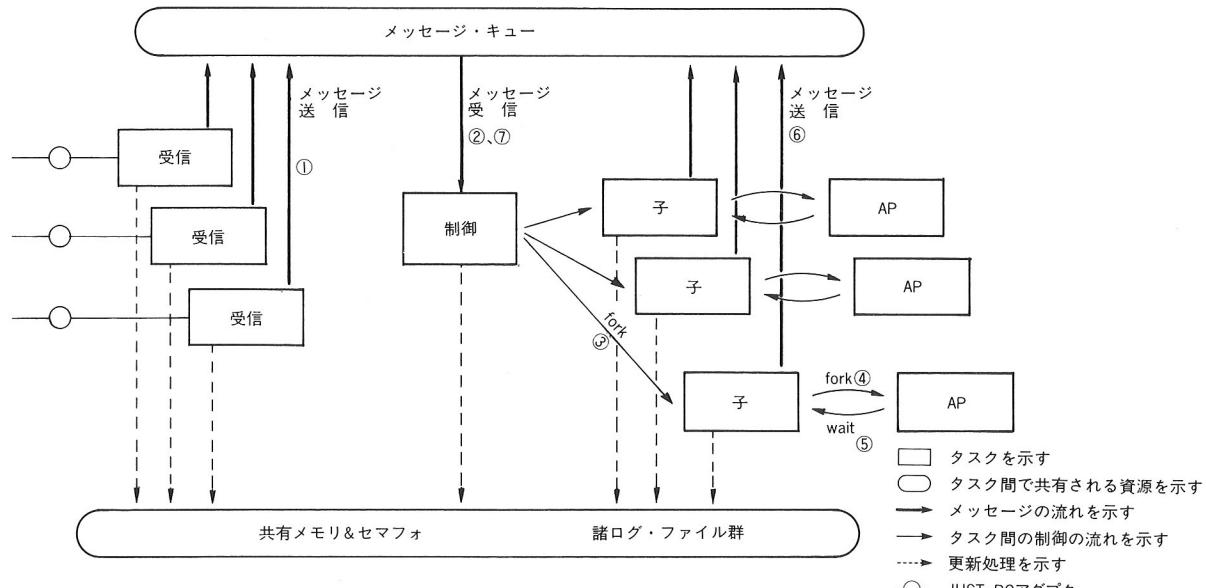


図-8 タスク相関図

は2のメッセージを受信すると、条件($n_1 > 0$ かつ $n_2 < 4$)が成立していれば、子タスクを1つ起動し、 n_1 を1減じ、 n_2 を1加算する。(条件不成立なら何もしない)。その後、制御タスクはメッセージ待ちとなる。

- ④—⑥ (子タスク) 子タスクはAPタスクを起動し、TCTを更新、ログ出力し、AP終了待ちとなる。APタスクが終了・消滅すると、子タスクは、 n_2 を1減じ、TCT上のジョブエントリを削除し、ログ出力、タイプ2のメッセージを出力した後、終了・消滅する。

5. その他の機能

鉄骨生産システムは、実行中に大きな作業ファイルを使うので、起動時にハードディスクの残容量が少ないと実行途中でアポートする恐れがある。アポートしても再実行などに不都合はないが、CPU時間を無駄にすることになるので、ディスク残量が一定量以下ならAPを起動しないことにした。

APタスクが作成されるファイルのうち、次の2つのファイルは特殊の目的に使用される。

① err. msg

ジョブの実行リストのファイルで、エラーがあった時などに、作業者またはオペレータが原因を探知する手段として使う。

② jobreport

ジョブのアカウント情報を保存するための数行程度の短いファイルで、この情報はシステム側で計測するアカウント情報とあわせて、ジョブごとにアカウントファイル(acct)にアペンドされる。鉄骨生産システムの場合、処理柱図面数、処理梁図面数などのデータになっている。またAPによって検知できたエラーのコード番号を記録する。

本システムは再スタート機能をもち、システムダウンした場合、再スタートオプションで立ち上げることによって以下の3種類のジョブは、継続処理することができる。

- ① ユーザから正常にファイルを受信した後ダウンしたジョブ
- ② AP実行中であったジョブ
- ③ ディスク領域不足のため待機中であったジョブ

これらのステータスはステータスログファイルに保存されており、再スタートオプション指定で起動されると制御タスクはまず、このファイルを調べる。

また枝番処理されたジョブは、ステータスログファイル中のステータスをエディタで上の①と同じものに変更することによって、再スタート時に実行させることができ

きる。

6. あとがき

著者らは自動運転システムを開発するにあたって、できるだけ鉄骨生産システムとは示様独立になるよう努め、効率性等の理由からこれを破る場合でも、その範囲を努めて局所化した。従ってこのシステムも、類似のシステムをミニコン上に開発する場合には、相当利用または参考にすることができるものと思う。パソコンとミニコンの間のJUST-PC通信プログラム、GPIBファイル転送プログラム等は単体としても利用価値があると思われる。

パソコンやワークステーションがますます高機能化し、多くのソフトがその上で稼働するようになると、多数のパソコンを使った分散処理が進む一方で、ミニコンや大型機などに蓄積されたデータベースや設計情報を利用したり、設計の流れ又は作業者の都合によって、パソコンまたはミニコン上のCADソフトウェアを選択使用したり、ミニコンでさまざまなソフトウェアによって実行された結果をパソコンでまとめて編集・印刷出力したり、いろいろな高度な利用形態があらわれてくるものと思われる。このシステムはその1つの形態である。

著者らにとっては、このシステムは、パソコン通信、マルチタスク制御という2つの点で、従来あまりなじみのないものであったが、このシステム開発で学んだ異種計算機利用のノウハウは、分散化とネットワーク化の進むこれから的情報化社会にむけて貴重なものとなろう。

なお、このシステムの開発にあたって多大な協力をいただいた日本ユニバックス情報システム株式会社の白枝邦夫氏はじめ多くの方々に感謝の意を表したい。

参考文献

- 1) 北島・松原・橋本・坪山・上田・猪瀬：鉄骨生産システム運用報告と近況、川田技報、Vol.6, 1987.
- 2) 石田ほか：コンピュータ・ネットワーク、1987.
- 3) 塩谷：実用UNIXシステムプログラミング、1986.
- 4) 山田：マルチタスク・プログラミング入門、インターフェース、No.121, 1987.