

POPS キャラクタ・ディスプレイ・サブルーチン・ パッケージ (CSP)

POPS Character Display Subroutine Package (CSP)

柿 間 克 彦* 堀 田 敬 志*
Katsuhiko Kakima Takashi Horita

I. ま え が き

計算機制御の分野ではシステムの大規模かつ複雑化が進むにつれて、従来の“機械”重点のシステムに代わって、“人間”と“機械”との協調を図ったシステムが広く採用され始めている。このシステムは“人間”の思考力、判断力と“機械”の正確さ、高速性を自然な形で結びつけ、信頼性を向上させると共に、より高度な制御も行わせようとするものであり、一般的にマンマシンコミュニケーションシステムと呼ばれている。ディスプレイ装置は、その情報伝達における高速性、多量性あるいはインタラクティブ性などにより、このマンマシンコミュニケーションシステムを形成するための重要な装置の一つであり、今日では、いわゆるバッチ処理システム、オンラインシステムにとどまらず、プロセス用計算機制御システムにおいても広く採用されている。特にプロセス用計算機制御システムにおいてはプラント操業の現場に設置され、従来のスイッチ、ランプ、数値表示器などによって構成されていたプロセスオペレータズコンソール(POC)にとってかわろうとしているが、それは単なる置き換えではなく、新しい可能性をもたらすものであり、場合によっては、計算機制御システムの適用分野を広げるものとも考えられる。

当社においても早くから計算機制御システムにディスプレイ装置を導入し、マンマシンコミュニケーションシステムの実現化を推進してきた。そこで今回、当社の今までの経験をもとに FACOM/PANAFACOM U シリーズを対象としてプロセス用キャラクタディスプレイ表示システム CSP (Character Display Subroutine Package) を開発、実用化したのでここに報告する。

II. 開発の思想とねらい

現在もディスプレイ装置はハードウェア的にもソフトウェア的にもめまぐるしく進歩しつつあり、その使用法は一般の入出力機器のように固定化されるに至っていない。特に、計算機制御システムにおいては、記憶容量

や応答速度に対する制約が厳しいこともあって、広範囲なプログラム作成者を満足させるようなソフトウェアシステムの開発には困難がともなう。このためシステムにディスプレイ装置を組み込む場合、一般の入出力機器のように単機能的に用いるかあるいはソフトウェアの負担を大きくし高度な対話処理まで行わせるか、いずれにしてもユーザは妥協を強いられ、プログラム作成者に大きな負担がかかる結果になっている。

また、ディスプレイ装置自体についても、

- (1) ハードウェア的には単機能装置であるが、その表示機能とインタラクティブ機能を組み合わせソフトウェアで加工することにより機能あるいは使用範囲が大きく広がる。
 - (2) 常に人間と直接接触する機器であるため表示画面のフォーマットあるいは対話時における応答性、操作性など人間工学的見地からの配慮が必要である。
 - (3) 情報伝達時の高速性、多量性などの利点を損わせない。
 - (4) 装置の接続台数がシステムの大きさにより大幅に異なり、当然増設時の考慮も必要である。
- などの特性があるためプログラム作成者の負担を増長させることとなる。

CSPはこのような観点に基づいて開発されたキャラクタディスプレイ装置における汎用のサブルーチンパッケージである。

CSPの特長を以下に記す。

- (1) 画面の作成が容易である。
- (2) ディスプレイ装置へ多量のデータを効率よく転送するため、画面データ領域の縮小化を図っている。
- (3) オペレータあるいはプロセスからの即答性が要求されるため表示時間を重視している。
- (4) ディスプレイ装置の利点でもある豊富なインタラクティブ機能を容易に利用できるように設定データの読み込みなどのサポートをしている。
- (5) プロセス用計算機制御システムにおける多様なデータ、きめこまかな処理に対応させるため各種の表示機

* 情推・開発部

能を持つ。

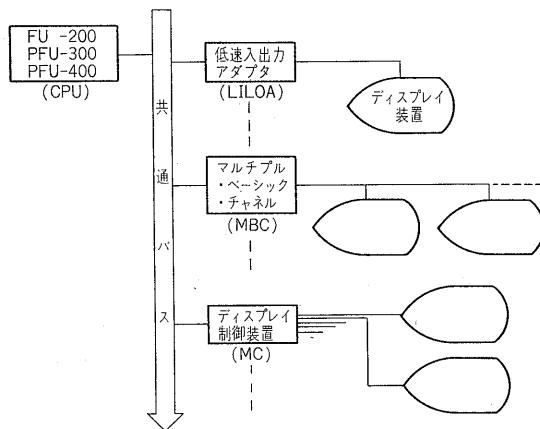
- (6) オンラインで画面あるいはディスプレイ装置（ソフト面）のメンテナンスが可能である。
- (7) この種のパッケージでとかく問題となる占有率についても可能な限り低くおさえている。
- (8) モジュール方式をとっているためシステム設計上の柔軟性を持っている。

また、CSPは当社で開発を進めてきた POPS⁽¹⁾ (Process Oriented Programming System) の中の一つであり、POPSの基本的な考えがその基に流れている。

III. システム構成

1. ハードウェアの構成

CSP は FACOM U-200, PANAFACOM U-300/400 のオペレーティングシステムUMOSのもとで動作し、計算機とディスプレイ装置は第1図のように接続される。



第1図 CPUとディスプレイ装置間の接続方式

Fig.1. Connection diagram between CPU and display

CSPではこれらのディスプレイ装置のうち、F2900シリーズ（F6226Aを含む）、F9520シリーズのすべてのキャラクタディスプレイ装置を取り扱い、ハードウェア上接続可能な台数を機種に関係なく一括して制御を行っている。

2. ソフトウェアの構成

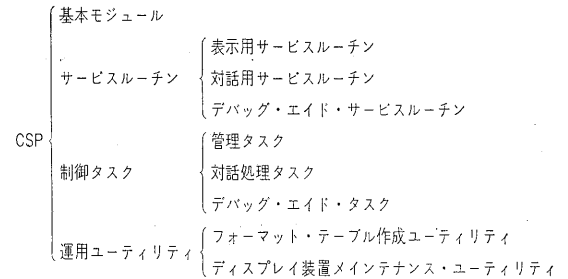
CSPはデータの作成から、表示、対話、運用面に到るまで広い範囲にわたってサポートしており、そのプログラム構成を第2図に示す。また、プログラム構造的にも第3図のようにモジュール方式をとっているため、柔軟性、信頼性、保守性に富んでいる。

以下にそれぞれのプログラムの概要を述べる。

1) 基本モジュール

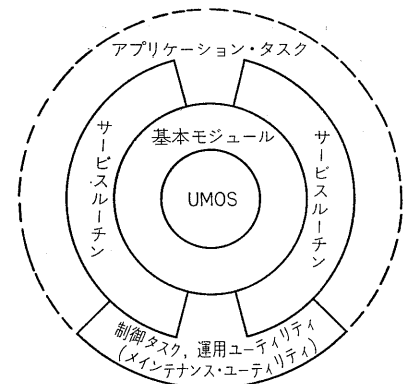
CSPの核となる内部サブルーチン

2) サービスルーチン



第2図 CSPのプログラム構成

Fig.2. Construction of CSP



第3図 CSPのプログラム構造

Fig.3. Structure of CSP

アプリケーションタスクより呼び出され、各種のサービスを行うサブルーチン

(1) 表示用サービスルーチン

表示データのハンドリング、編集および管理タスクへの表示要求、あるいは不要画面の解放要求などの処理を行う。

(2) 対話用サービスルーチン

計算機とオペレータの同期処理、オペレータからの設定データの読み込みなどを行う。

(3) デバッグエイドサービスルーチン

CSP用制御テーブルの内容の印字要求を行う。

3) 制御タスク

アプリケーションタスクと同期をとりながら、独立タスクとして管理、制御面において各種のサービスを行う。

(1) 管理タスク

ディスプレイ装置への出力、オペレータよりの割込みの解析、不要画面の解放などの処理を行い、CSPではモニタプログラムの存在である。

(2) 対話処理タスク

利用者により作成されたテーブルを参照しながら、オペレータとの対話を行う。

(3) デバッグエイドタスク

デバッグ時に利用され、CSP用制御テーブルの内容

を印字するタスク、ディスプレイ装置への表示データを印字するタスクがある。

4) 運用ユーティリティ

利用者より必要時に始動され、運用面における各種のサービスを行う。

(1) フォーマットテーブル作成ユーティリティ

第1表 CSPサービスルーチン一覧
Table 1. List of CSP service routines

分類	エントリ名	機能概要
表示用サブルーチン	PICT	画面データファイルを参照し、台画面を表示する。 プログラム内の EBCDIC コードデータを表示することもできる。
	SPICT	画面データファイルを参照し、画面上の指定位置にサブピクチャ表示を行う。 プログラム内の EBCDIC コードデータを表示することもできる。
	DMSG	プログラム内の EBCDIC コードデータを画面上の指定位置に表示する。サブピクチャ表示も可能。
	DMSGGE	プログラム内の ISO コードデータを画面上の指定位置に表示する。サブピクチャ表示も可能。
	GRAPH	画面データファイルを参照し、グラフパターン部を表示する。入力データをプログラム内に持つこともできる。サブピクチャ表示も可能。
	SCROLL	スクロールアップモード表示を行う。
	MPDLT	画面を解放する。対話終了処理も可能。
	SPDLT	サブピクチャを消去する。
	SCNDS	表示データファイルをコンデンスする。
対話用サブルーチン	DSPIOX	IOX レベルでディスプレイ装置をアクセスする。
	ATRCV	オペレータよりのインタラクション割込みを待ち、割込後設定データを読み込む。
	ATRCVW	対話処理の開始を宣言し、その画面での対話が終了するまでウェイト状態に入る。対話終了後ポストされる。
	ATRCVS	対話処理の開始を宣言する。対話終了後に始動するタスクを指定することもできる。
	ATRCVE	対話終了後に始動されるタスクの先頭で呼び出され、対話中に機器の異常が発生したかどうか調べる。
	FLREAD	オペレータから設定されたライトペン、ジョイスティックの画面上のアドレス、あるいはファンクションキーのキーコードを読み込む。
	CSREAD	オペレータから画面上に設定されたキーインデータ (SENDボタン) あるいは ID カードの内容を読み込む。
	DLGRST	対話処理を一時中断させる。
	DLGEND	対話終了処理を行う。
サービスユーティリティ	CSDSMP	CSP用制御テーブルの内容の印字要求を行う。

画面の修正、追加、削除などの保守を行う。

- (2) ディスプレイ装置メンテナンスユーティリティ
ディスプレイ装置をCSPの管理から切り離したり、あるいはCSPの管理へ復帰させたりする。

CSP利用者はプロセスの要求 (システムの規模、ディスプレイ装置の使用方法などにより決定される) に応じて、必要なプログラムを選択し、常に最適なCSPシステムを使用することができる。第1表にサービスルーチン、第2表に運用ユーティリティの一覧を示す。

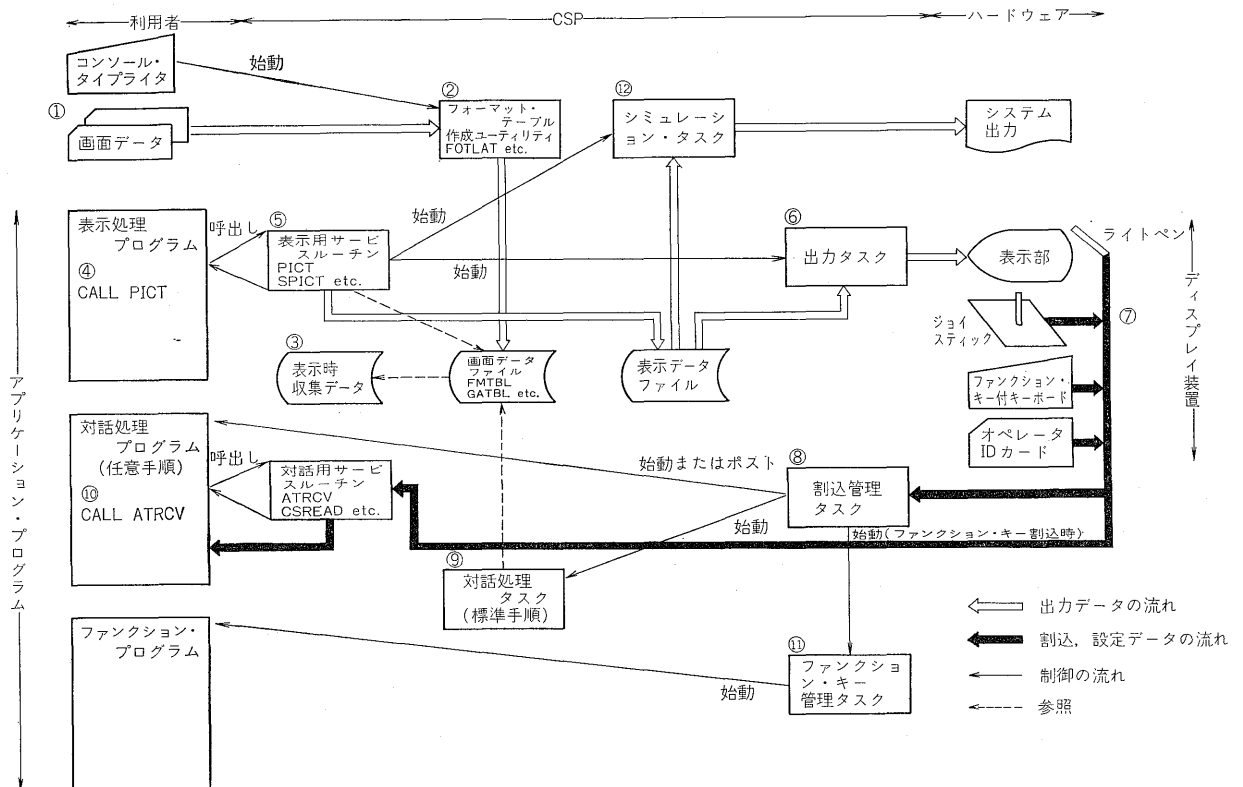
3. 処理の流れ

CSPの表示、対話時における標準的な処理の流れを第4図に示す。

- 画面データをカードにて作成する (利用者)。画面データとは、表示フォーマット、表示データ収集領域、対話内容などを規定するデータ群である。
- 画面データを翻訳しファイルに登録する (CSP)。画面データは登録後、テーブルとして管理され、CSPはこれらのテーブルをもとに表示、対話を行う。また、翻訳、登録以外にテーブルの削除、ダンプを行うユーティリティも用意されている。
- 表示時収集データの作成 (利用者)。表示用サービスルーチンを呼び出す前に、それと同一あるいは別タスクが(1)で規定される収集領域へデータを格納する。データはバイナリ形式、コード形式のいずれでも可能である。
- 表示用サービスルーチンを呼び出す (利用者)。

第2表 CSP運用ユーティリティ一覧
Table 2. List of CSP operational utilities

分類	エントリ名	機能概要
フォーマットテーブル作成ユーティリティ	FOTLAT	紙テープまたはカードにパンチされているフォーマット文をオブジェクトコードに変換する (FMTBL に適応)。
	GATLAT	紙テープまたはカードにパンチされているギャザリング文をオブジェクトコードに変換する (GATBL に適応)。
	FGDUMP	FOTLAT または GATLAT によって翻訳されたオブジェクトコードを紙テープにパンチアウトする (FMTBL, GATBL に適応)。
	DPDUMP	FASPによって翻訳され、PRBLにて作成されたオブジェクト・コードをテープにパンチアウトする (GPTBL, DETBL に適応)。
	PICTLD	FMTBL, GATBL, GPTBL, DETBL のそれぞれのオブジェクト・コードを画面データファイルへ登録する。
	GFLST	画面データファイル中の各種のテーブルの内容を印字する。
	PITDLT	画面データファイル中の各種のテーブルを削除する。
	DCHECK	ディスプレイ装置のメンテナンスを行う。



第 4 図 CSPにおける処理の流れ

Fig. 4. Process flow of CSP

- (5) 画面データファイルを参照しながら、表示データのハンドリング、画面の編集を行い、コード化された表示データをファイルに格納する。その後、出加タスクあるいはシミュレーションタスクに制御を渡す (CSP)。
- (6) 表示データをファイルより読み取り、ディスプレイ装置へ出力する (CSP)。
- (7) 計算機へのデータ設定およびインタラクション割込み (オペレータ)。
- (8) 割込内容の解析および該当の対話処理タスクの始動 (CSP)。
- (9) 対話処理 (CSP)。画面データファイル中のテーブルを参照しながら、オペレータとの対話を進める。
- (10) 対話用サービスルーチンを呼び出す (利用者)。対話処理における標準ルーチン ((8)) がCSPで用意されているが、利用者独自のきめのこまかい処理が要求される場合には別に作成することもできる。
- (11) ファンクションキーのキーのコードに対応したタスクを始動する (CSP)。
- (12) 表示データをファイルから読み取り、ディスプレイ装置をシミュレートし、その結果をシステム出力に印字する (CSP)。

IV. CSPの機能

画面データの作成

CSPは次のような各種の画面データテーブルにより表示、対話処理を行っているが、テーブルデータの作成方法も容易である。

(1) フォーマットテーブル (FMTBL)

一般画面の表示フォーマットを規定するテーブルである。フォートランの FORMAT ステートメントに類似した形式で記述し、フォーマットテーブル作成ユーティリティ FOTLAT にて翻訳する。

(2) ギャザリングテーブル (GATBL)

一般画面を表示する時のデータ (特に可変データに対して有効) の収集領域を規定するテーブルである。ステートメント形式で記述し、フォーマットテーブル作成ユーティリティ GATRAT にて翻訳する。

(3) グラフパターンテーブル (GPTBL)

グラフパターン部の表示フォーマット、入力データの領域などを規定するテーブルである。マクロ命令で記述し、アセンブラ FASP にて翻訳する。

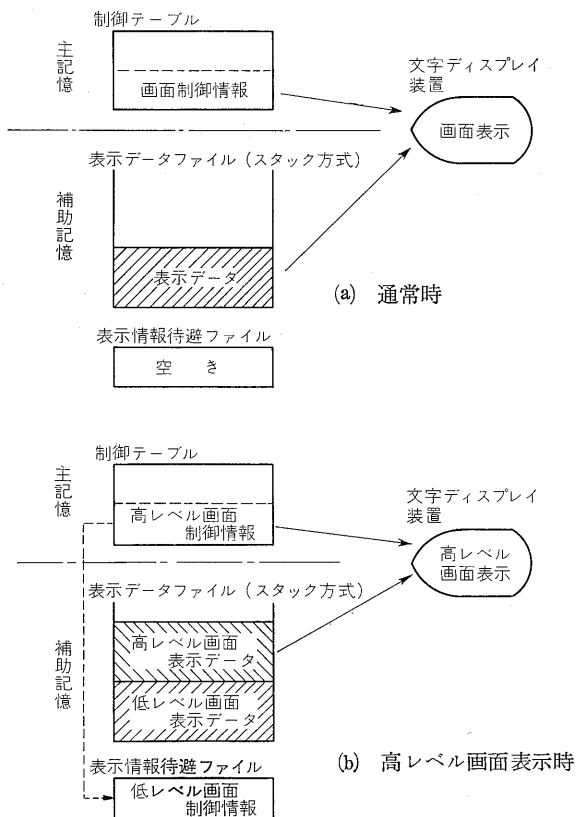
(4) データエントリテーブル (DETBL)

オペレータより表示画面に設定されるデータを規定するテーブルである。マクロ命令で記述し、アセンブラ FASP にて翻訳する。

また、これらのテーブルを管理するユーティリティが完備しているので、オンライン時においても表示画面の修正、追加、削除が可能である。

2) 画面の優先処理

リアルタイム処理では、タスクに優先度が必要であるように画面にもやはり優先度が必要になってくる。CSPでは同一ディスプレイ装置上において画面を低レベルと高レベルに分けて、リアルタイムシステムのタスクにおける優先処理のような機能をタスクのレベルとは独立して画面に持たせている。第5図にその原理を示す。CSPでは、常に表示データファイル中の最新データ（表示データファイルはスタック方式をとっている。最新データとは最後に格納されたデータを意味する）を表示し、その画面は制御テーブル中の画面制御情報（画面番号、画面レベルなど）によって管理されている。第5(a)図に通常時の様子を示す。第5(b)図のような状態で低レベル画面が表示されているところへさらに高レベル画面の表示要求があった場合には、第5(b)図に示すように、制御テーブル中にあった低レベル画面の制御情報が待避ファイルに待避され（制御テーブル中には高レベル画面の制御情報が作られる）、低レベル画面の表示データ上に高レベル画面の表示データがスタックされ、ディスプレイ装置上には高レベル画面が表示される。また、高レベル画面の解放要求があった場合には、低レベル画面の制御情報が制御テーブル中に復元され（制御テーブル中にあった高レベル画面の制御情報は壊される）、高レベル画面の表示データは棄てられ、第5(a)図の状態にもどり、低レベル画面が復元表示される。



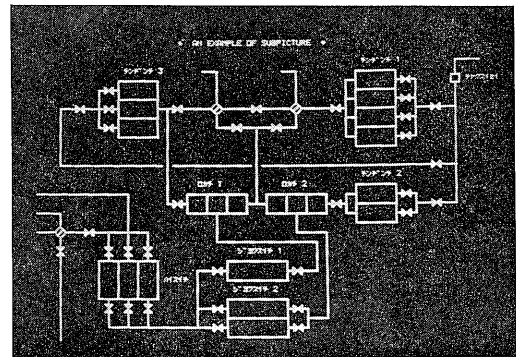
第5図 画面の優先処理

Fig. 5. Priority processing of pictures

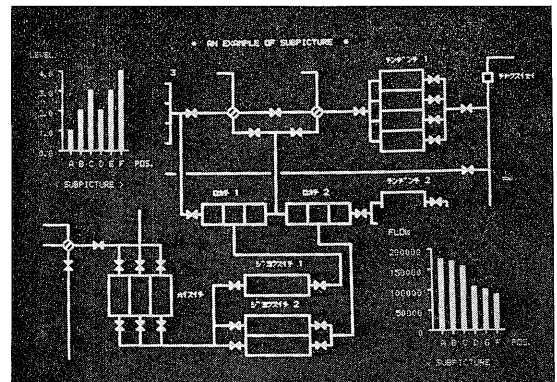
この機能により、たとえば通常のプロセスデータの表示中にプロセスよりの緊急データをいち早く操作中のオペレータに画面で知らせ、その緊急データが不要になった時点で元のプロセスデータ画面を再表示させるような処理が容易に行える。

3) サブピクチャ表示

表示中の台画面上に小画面をそう入したり、またその小画面を削除し元の台画面を復元する機能である。この小画面（サブピクチャ）を利用することによりデータの二元的表示が可能である。第6図に使用例を示す。



(a) 台画面



(b) 台画面と小画面

第6図 サブピクチャ表示例

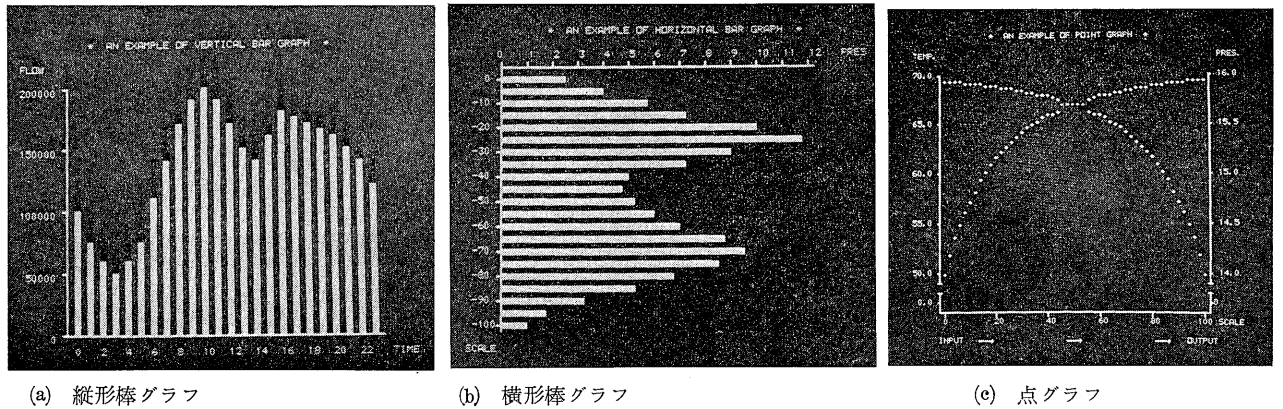
Fig. 6. Example of subpicture

4) グラフ表示

あらかじめグラフの形式、スケール値、データ収集領域などの情報を画面データファイル中に与えておくだけで、棒グラフ、点グラフを表示することができる。連続的に変化するプロセスデータのモニタリング表示も、3)のサブピクチャ表示機能を利用して効率よく行える。グラフの表示例を第7図に示す。

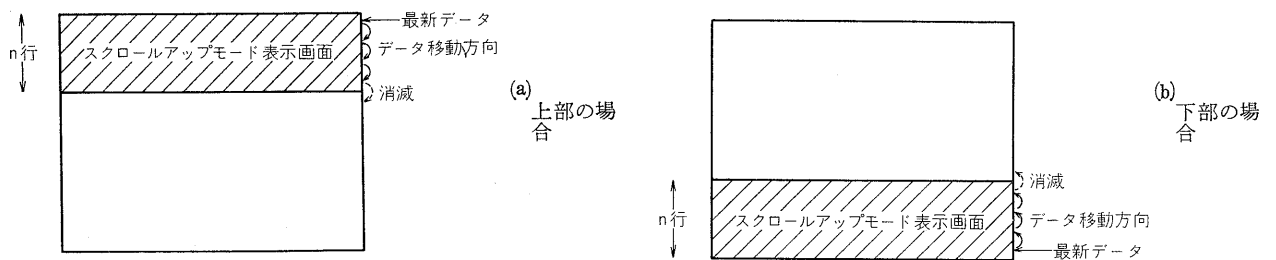
5) スクロールアップモード表示

画面の下部あるいは上部の任意の行のデータを行単位でロールアップ（上部の場合はロールダウン）しながら表示する機能である。一つの画面内にページモード画面（上部）とスクロールアップモード画面（下部）の双方を持たせる機能は、本来F9520シリーズのキャラクタデ



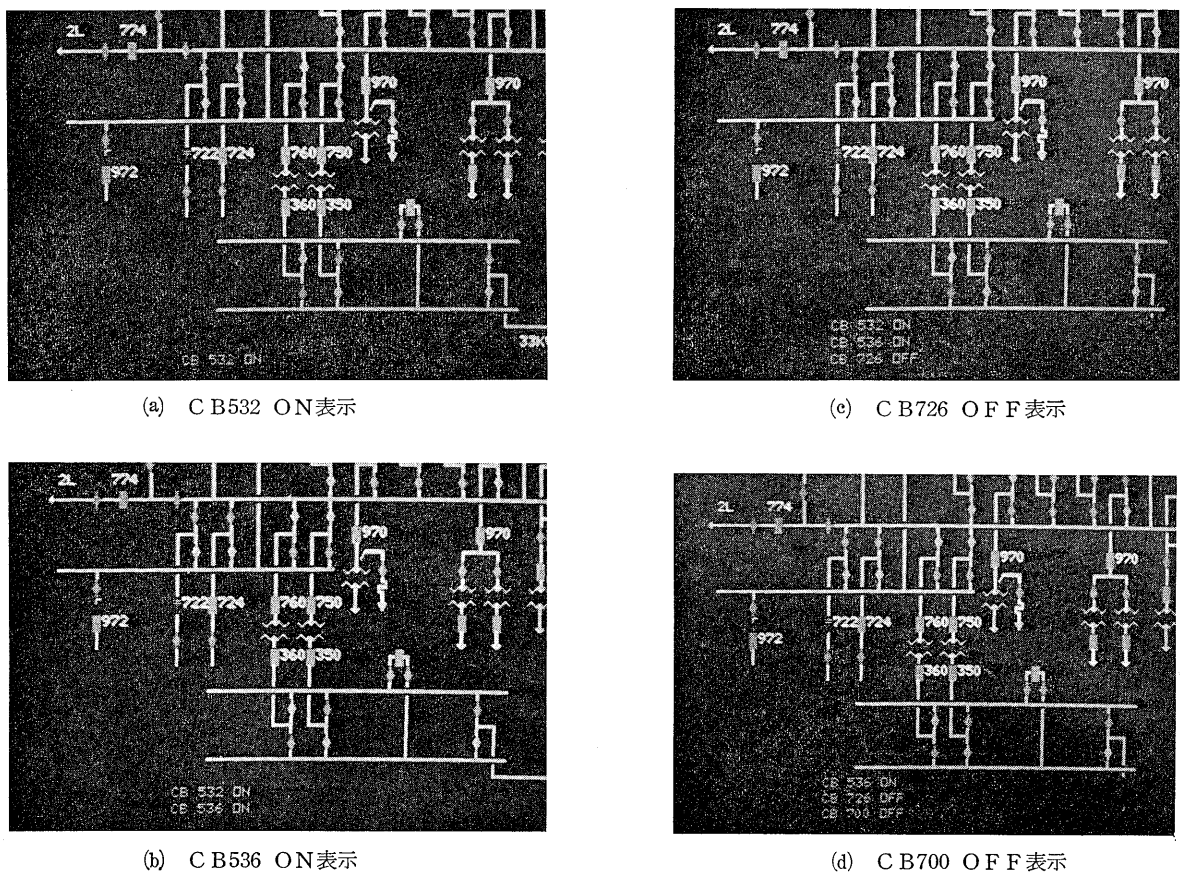
第 7 図 グラフ表示例

Fig. 7. Examples of graph



第 8 図 スクロールアップモード表示形式

Fig. 8. Display with scroll up mode



第 9 図 スクロールアップモード表示例

Fig. 9. Example of scroll up mode

ィスプレイ装置ではハード的に与えられたものであるが、CSPではこの機能をソフト的に他の装置でも利用可能とし、さらに上部でも表示できるようにしたものである。第8図に表示形式、第9図に表示例を示す。この機能はオペレータガイドに適している。

また、スクロールアップモードで表示されるデータを履歴としてファイルに保存しておくこともできるので、必要な時期にそれまでのデータをシステム出力印字することができる。

6) インタラクシオン機能のサポート

F9500/F2900 シリーズの各ディスプレイ装置には次のような豊富なインタクシオン機能が用意されている。

- ・文字キーボード：SEND機能（全機種）
- ・ファンクションキー（全機種）
- ・ライトペン（F9522R）
- ・ジョイスティック（F2905A）
- ・オペレータIDカード（F9522R）

CSPでは、これらの機能すべてについてサポートし、できるだけ利用者の負担を少なくしている。

7) ディスプレイ装置相互間の独立性

ディスプレイ装置相互間は独立しており、表示、対話時あるいは10)に述べるメンテナンス操作時においても他の装置に影響を与えたり、他の装置から影響を受けたりすることはない。

8) シミュレーション機能

デバッグ時の結果を残しておきたい時、またはディスプレイ装置が接続されていない時などに利用でき、ディスプレイ装置の動きをシミュレートし、その結果をシステム出力に印字する機能である。印字する種類は文字（シフトコードを含む）、カラー、プロテクションフィールドの3種類である。また、オンライン中においてディスプレイ装置との間で切り換えが可能である。第10図に印字例を示す。

X AXIS	Y AXIS	CHARACTER
0	0	0
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9
10	10	10
11	11	11
12	12	12
13	13	13
14	14	14
15	15	15
16	16	16
17	17	17
18	18	18
19	19	19
20	20	20
21	21	21
22	22	22
23	23	23

第10図 シミュレーション印字例

Fig. 10. Example of simulation list

6) 制御テーブル印字機能

CSP関係の制御テーブルの内容をシステム出力に印字するデバッグエイド機能である。印字例を第11図に示す。

TCDSP	DSWP	TSP	TSP
0001	0002	0003	0004
0005	0006	0007	0008
0009	0010	0011	0012
0013	0014	0015	0016
0017	0018	0019	0020
0021	0022	0023	0024
0025	0026	0027	0028
0029	0030	0031	0032
0033	0034	0035	0036
0037	0038	0039	0040
0041	0042	0043	0044
0045	0046	0047	0048
0049	0050	0051	0052
0053	0054	0055	0056
0057	0058	0059	0060
0061	0062	0063	0064
0065	0066	0067	0068
0069	0070	0071	0072
0073	0074	0075	0076
0077	0078	0079	0080
0081	0082	0083	0084
0085	0086	0087	0088
0089	0090	0091	0092
0093	0094	0095	0096
0097	0098	0099	0100

第11図 制御テーブル印字例

Fig. 11. Example of control block list

10) ディスプレイ装置メンテナンス機能

ハードウェアによるディスプレイ装置異常時の、その異常状態のチェック、装置のリジェクション（CSPの管理から切り離すこと）、また復旧時の装置のリカバリー（CSPの管理へ戻すこと）がオンラインで可能である。これらの操作はオペレータと計算機の間でコンソールタイプライタを介して対話形式で行われる。操作例を第12図に示す。

CHECK	PLEASE	CHECK
1100	1101	1102
1103	1104	1105
1106	1107	1108
1109	1110	1111
1112	1113	1114
1115	1116	1117
1118	1119	1120
1121	1122	1123
1124	1125	1126
1127	1128	1129
1130	1131	1132
1133	1134	1135
1136	1137	1138
1139	1140	1141
1142	1143	1144
1145	1146	1147
1148	1149	1150
1151	1152	1153
1154	1155	1156
1157	1158	1159
1160	1161	1162
1163	1164	1165
1166	1167	1168
1169	1170	1171
1172	1173	1174
1175	1176	1177
1178	1179	1180
1181	1182	1183
1184	1185	1186
1187	1188	1189
1190	1191	1192
1193	1194	1195
1196	1197	1198
1199	1200	1201

第12図 メインテナンスユーティリティによる操作例

Fig. 12. Example of operation by maintenance utility

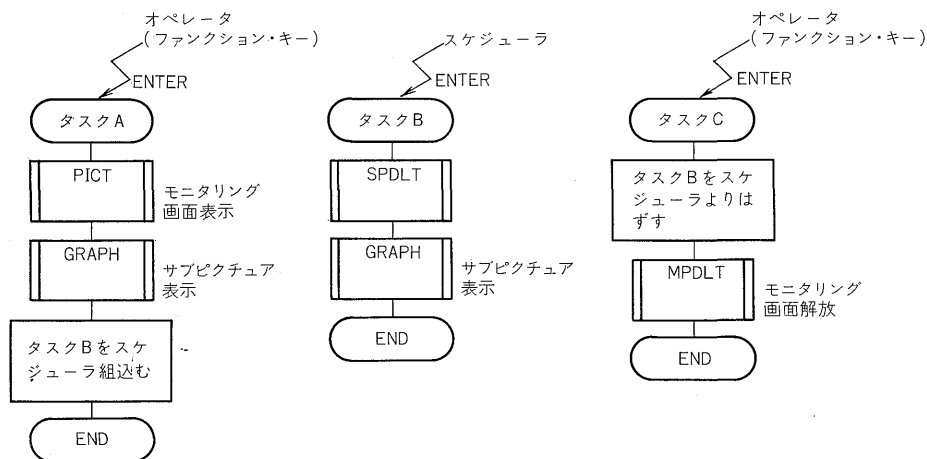
V. CSPの使用例

CSP利用者は第1表中の各種のサービスルーチンを適宜組み合わせたアプリケーションプログラム（システム）を作成することにより、多種多様のディスプレイ装置表示、対話システムを実現できる。以下に使用例を示す。

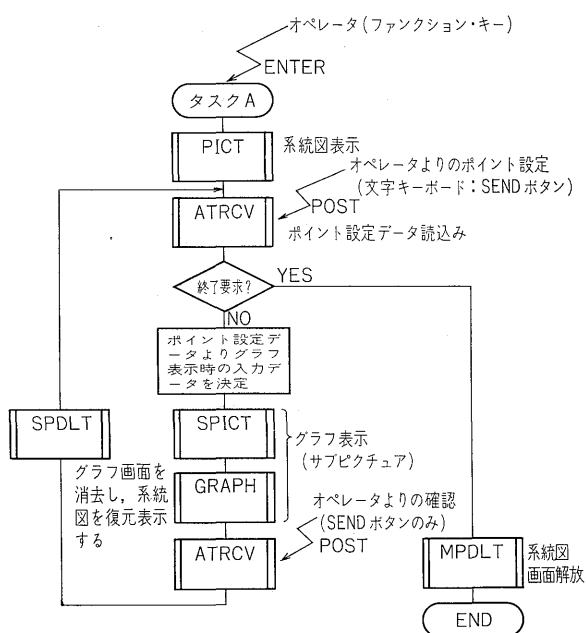
1) グラフによるモニタリング表示の例（第13図）

タスクA：モニタリング開始ファンクションキーが押されて始動される。

タスクB：スケジューラにより一定時間間隔でサイク



第13図 使用例 1
Fig. 13. Example of CSP 1



第14図 使用例 2
Fig. 14. Example of CSP 2

```

1  #
2  # EXAMPLE OF CODING FOR FIG.14
3  #
4  #
5  # PROG REI2
6  # VECT X'4000',AD
7  # DCSP PICT+ATRCV+SPIC+SPDLT+GRAPH+MPDLT
8  # ENTRY REI2
9  #
10 # REI2 PICT 1100,0,0,0,0,0,WORK,16,1
11 # CHKPOINT ATRCV 1100,0,4000,WORK,0,5
12 # IF WORK+2,END,DISPLAY,RELEASE,DISPLAY
13 # DISPLAY LAND =X'000F',WORK+2,GRAPH,0,4
14 # SPIC 1100,0,1,0,0,0,WORK,16,0,0
15 # GRAPH 1100,0,0,1000,0,WORK,16
16 # ATRCV 1100,0,4000,0
17 # SPDLT 1100,1,1
18 # GOTO CHKPOINT
19 # RELEASE MPDLT 1100,1
20 # STOP
21 #
22 # DATA X'C54E' CODE OF 'EN'
23 # DS 256D
24 # OS 256D
25 # END

```

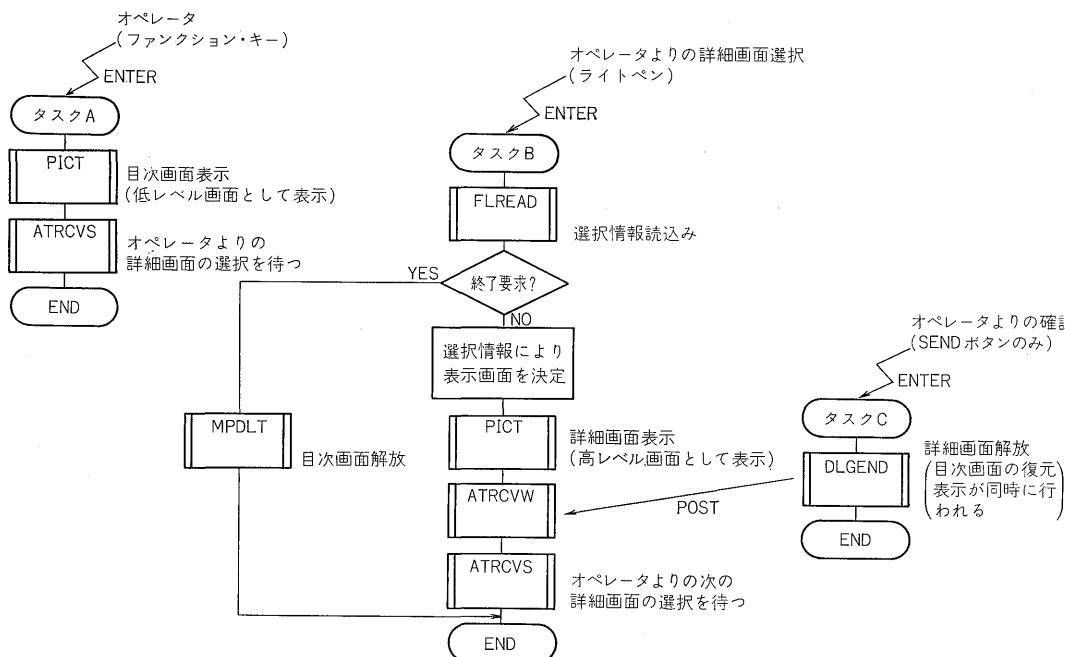
第15図 FPLによる使用例のコーディング例
Fig. 15. Example of coding by FPL for Fig. 14

リックに始動される。

タスク C：モニタリング終了用ファンクションキーが押されて始動される。

2) 系統図中のあるポイントの時系列データをグラフとしてサブピクチャ表示する例 (第14図)

タスク A：表示要求時にファンクションキーが押されて始動される。



第16図 使用例 3
Fig. 16. Example of CSP 3

FPLによるコーディング例を第15図に示す。

3) 画面レベルを用いた目次画面とそれに対する詳細画面の表示例 (第16図)

タスクA: 表示要求時にファンクションキーが押されて始動される。ATRCVSにて割込時の始動タスクをタスクBとする。

タスクB: ライトペンで参照したい画面が選択された時始動される。ATRCVWで割込時の始動タスクにタスクC, ATRCVSでタスクB自身をそれぞれ指定する。

タスクC: SENDボタンで詳細画面の参照終了が通知された時始動される。

VI. あとがき

以上、当社で開発したCSPについてその概要を述べた。CSPは鉄鋼、電力などの各分野ですでに稼動しており、現在も多数のプロセス用計算機制御システムにおいて組み込まれ稼動しつつある。これまでの結果、従来のSEの負担が軽減され、また表示時のオーバーヘッドタイムも平均0.5秒(640字表示用)～2秒(4,000字表示用)という実測値を得ている。CSPは現在のキャラク

タディスプレイ装置の持つ機能を最大限に生かすことを考えて作成されたものであるが、今後のハードウェアの動向、ユーザのニーズを常に見きわめながら、必要に応じて改良していきたいと考えている。

参考文献

- (1) 大星・小原: FACOM U-200 プロセス用プログラミング・システム (POPS) 富士時報 47 No.7 (昭49)
- (2) 柿間・堀田: プロセス用キャラクタ・ディスプレイ・パッケージ・システム (CSP) 昭50電気学会全国大会 p.1844
- (3) 大塚他: 荒川左岸流域下水道組合納入計算機装置 富士時報 45 No.11 (昭47)
- (4) 北浜他: 転炉工場の計算機制御 富士時報 46 No.10 (昭48)
- (5) 香川: CRTディスプレイ装置を用いたプロセス用汎用データ入出力操作盤 富士時報 47 No.4 (昭49)
- (6) 清水他: 静岡県駿豆水道用水給水事業ポンプ場中央集中管理システム 富士時報 47 No.9/10 (昭49)
- (7) 上本他: 川崎市長沢浄水場DDCシステム 富士時報 47 No.9/10 (昭49)
- (8) 齊藤他: 豊中市下水道部庄内下水道処理場計算機制御装置 富士時報 47 No.9/10 (昭49)
- (9) 山村他: 最近の製鋼プロセス計算機制御の動向 富士時報 47 No.11 (昭49)

発明の紹介

三相不平衡回路の力率測定方法

(特許第 690374 号)

この発明は、三相不平衡回路において、電流および電圧の位相角を検出し、この各相の位相角の和の平均を力率角として等価的に力率を測定するようにした力率測定方法に関する。

三相不平衡回路の等価力率は式(1)で定義されている。

$$\cos\phi = \frac{\text{有効電力}}{\sqrt{(\text{有効電力})^2 + (\text{無効電力})^2}} \quad \dots\dots\dots(1)$$

各相の電圧および電流を対称分で表示すると、等価力率 $\cos\phi$ は式(2)で表示される。

$$\cos\phi \doteq \cos\phi_1 + \frac{E_0}{E_1} \frac{I_0}{I_1} \sin\phi_1 \sin(\phi_1 - \phi_0) + \frac{E_2}{E_1} \frac{I_2}{I_1} \sin\phi_1 \sin(\phi_1 - \phi_2) \quad \dots\dots\dots(2)$$

ここで、 ϕ_1 は各相電圧および電流の正相電圧、正相電流の位相角であり、電流および電圧の双方とも不平衡が小さいかあるいは一方が平衡していれば、 $\frac{E_0}{E_1} \frac{I_0}{I_1} \ll 1, \frac{E_2}{E_1} \frac{I_2}{I_1} \ll 1$ となり、式(2)は式(3)に変形される。

$$\cos\phi \doteq \cos\phi_1 \quad \dots\dots\dots(3)$$

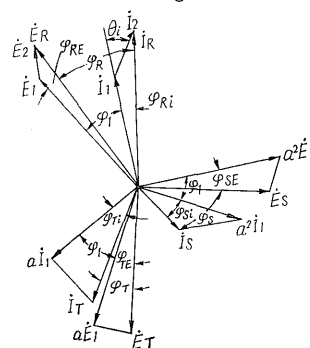
ところで、各相電圧および電流の位相角 $\varphi_R, \varphi_S, \varphi_T$ の平均 φ は式(4)で表示される。

$$\begin{aligned} \varphi &= -\frac{1}{3}(\varphi_R + \varphi_S + \varphi_T) \\ &= \phi_1 + \frac{1}{3}(\phi_{Ri} + \phi_{Si} - \phi_{Ti}) - \frac{1}{3}(\phi_{RE} + \phi_{SE} - \phi_{TE}) \quad \dots\dots\dots(4) \end{aligned}$$

電圧平衡であるから、 $\phi_{RE} = \phi_{SE} = \phi_{TE} = 0$ であり、また $\frac{1}{3}(\phi_{Ri} + \phi_{Si} - \phi_{Ti})$ は実用上無視し得るので、式(3)および式(4)から式(5)を得る。

$$\frac{1}{3}(\omega_R + \omega_S + \omega_T) \doteq \phi_1 \doteq \varphi \quad \dots\dots\dots(5)$$

したがって、この発明は、三相不平衡回路の電圧および電流の位相角 $\varphi_R, \varphi_S, \varphi_T$ を測定し、これらの位相角の和の平均値 $\frac{1}{3}(\varphi_R + \varphi_S + \varphi_T)$ から等価力率 $\cos\phi$ を測定するようにしたものである。





＊本誌に記載されている会社名および製品名は，それぞれの会社が所有する
商標または登録商標である場合があります。