

## プリスタック方式による入カシステムについて

伴, 直幸  
九州大学大型計算機センター

二村, 祥一  
九州大学大型計算機センター

景川, 耕宇  
九州大学大型計算機センター

宇津宮, 孝一  
九州大学工学部

<https://doi.org/10.15017/1468031>

---

出版情報 : 九州大学大型計算機センター広報. 6 (6), pp.4-15, 1973-12-22. 九州大学大型計算機センター

バージョン :

権利関係 :





め、装置の速度で読込まれるのはデータカードの時だけとなり、操作員のいらいらの大きな原因になります。プリスタックにすることによりこの問題が解決する理由は3で述べることにします。

以上のような背景から、MVI/VII (FACOM 230-60/75 オペレーティングシステム)でも、プリスタック方式を採用しているようです。第3の目的(これは副次的要素が強いのですが)は、第1の目的が達成されると、読込みが済んだジョブデックは、再度入力する必要がなくなり、利用者がすぐに持って帰れるようになるため、カードリーダーの利用者へのオープンが可能となります。

### 3. スタック方式とプリスタック方式の相違

#### 3.1 スタック方式

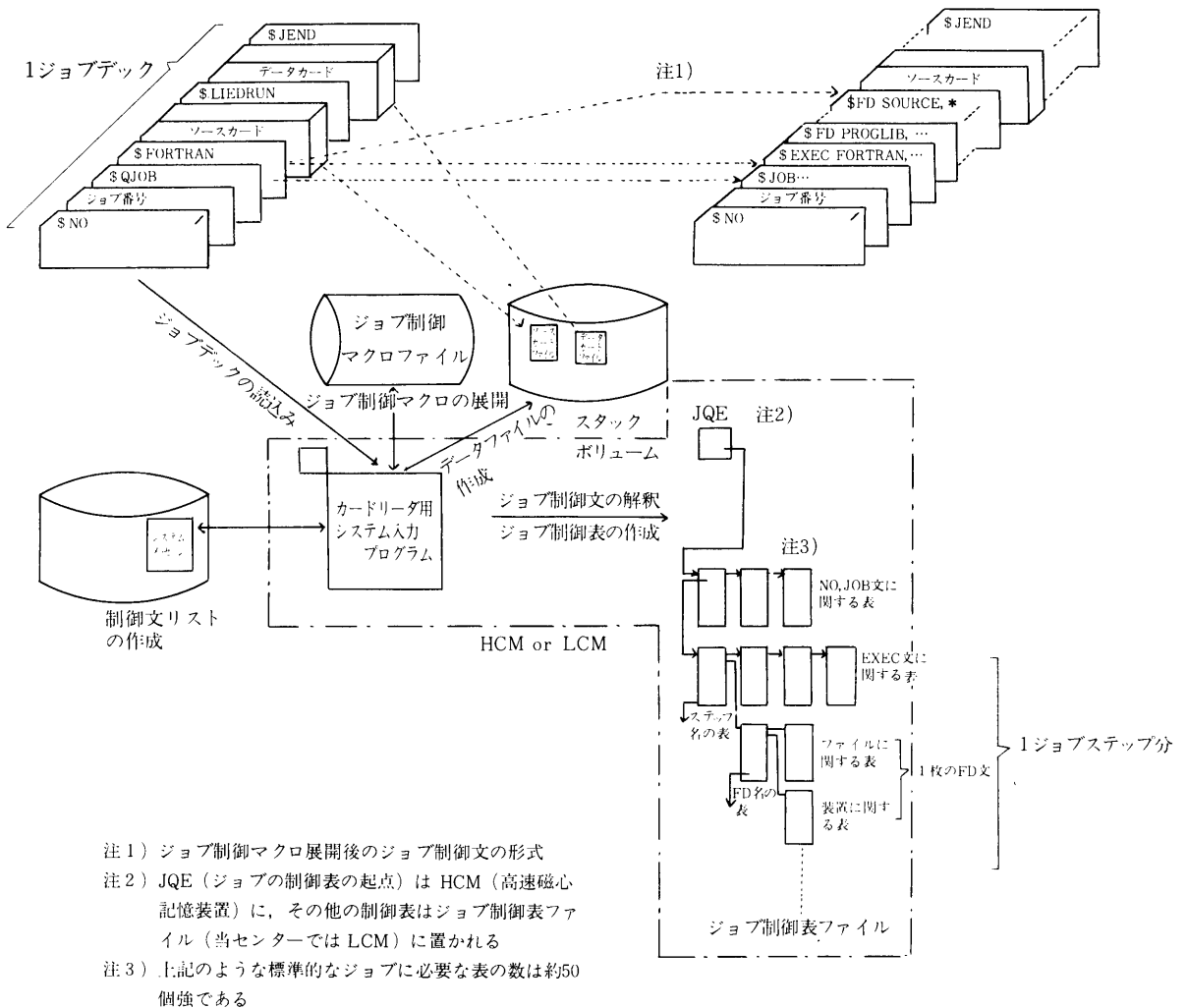


図 3.1 スタック方式概念図

システム入力プログラムは、ジョブデックの読込み、制御文の解釈、制御表の作成の3つの処理を行ないながらシステム入力を続けていきます。このような入力の方式を MV では、スタック方式の

入力と呼んでいます。スタック方式では、基本制御文、ジョブ制御マクロ文、データのカードによって以下のように3つの異なった処理をします。

- (1) 基本制御文カード (\$NO, \$JOB, \$EXEC, \$FD, \$JEND, \$CONV, \$NOCONV, \$SET, \$IF, \$REND, \$\*, \$\_) の場合  
 \$CONV などのように制御表の作成が不要な制御文は、解釈後次のカードの処理に移ります。制御表の作成が必要なものは、制御文を解釈しその内容を制御表にまとめてジョブ制御表ファイルに書出します。
- (2) ジョブ制御マクロ文カードの場合  
 基本制御文およびデータカード以外で第1文字が\$で始まるものはジョブ制御マクロ文と見なされ、システム入力プログラムはマクロ名をキーにしてジョブ制御マクロファイルの中を検索し、その内容を主記憶に読込んで来て、ジョブ制御マクロ文のオペランド部で指定されたパラメータの値をうめこみながら基本制御文列に展開していきます (図3.1 参照)。この処理にかなりの時間がかかるわけです。基本制御文列に展開された後の処理は(1)と同様です。
- (3) データカードの場合  
 \$FD\_ FD名, { \* DATA } の基本制御文が解釈されると、大記憶にこのジョブの中でのみ有効となる一時ファイルが確保されます。この文に続くデータ区切り文までの間の内容はデータとして、そのファイルに書き出されていきます。これをデータファイルと言います。

以上のような処理の後、結局ジョブデックはジョブに関する制御表とデータファイルに変換されて、初期のイメージは全くなってしまいますから、ジョブ制御表ファイルが書替えられると、もはやジョブデックとの対応づけはできなくなってしまいます。

### 3.2 プリスタック方式

プリスタック方式のシステム入力は、読み込み部と解釈・ジョブ制御表作成部とを機能的に分けてしまったもので、図3.2 のようになっています。

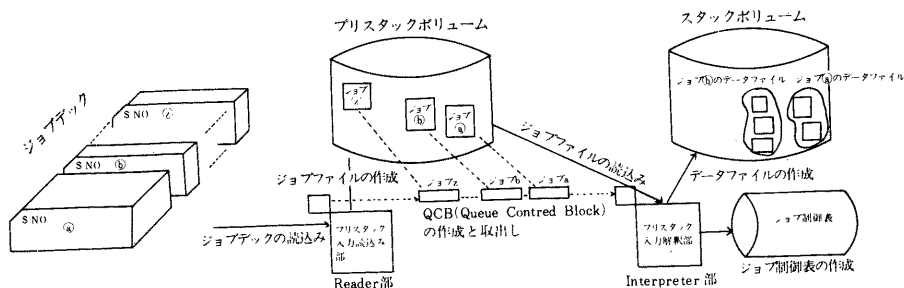


図 3.2 プリスタック方式概念図



## 4.2 機 能

### (1) 初期設定部 (Qdai SYS in INI tialize module)

- ① プリスタック入力済～出力中までのジョブファイルの復元と制御部への処理依頼
- ② プリスタック入力システム用制御表類の初期設定
- ③ 頭10文字が SYS1JOB.F.S (一時ファイル) のファイルとプリスタック入力途中のジョブファイルのスクラッチ処理
- ④ 大記憶システム入力プログラム (DAIN) の起動

### (2) 起動部 (Qdai Input action module)

- ① プリスタック入力部の起動

### (3) 操作部 (Qdai Input Operation module)

- ① プリスタック入力部および制御部への制御情報の設定
- ② プリスタック入力部の一時停止および終了指示
- ③ ジョブの情報のタイプライタへの表示

### (4) プリスタック入力部 (Qdai SYStem INput module)

- ① カードリーダからジョブデックの読み込みと大記憶 あるいは磁気テープへのジョブファイル あるいは集団ジョブファイルの作成と制御部への処理依頼
- ② 大記憶に作成されたジョブファイル群 あるいは集団ジョブファイルの磁気テープへの退避
- ③ 磁気テープ あるいは大記憶から集団ジョブファイルを読み込み、大記憶へのジョブファイル あるいは集団ジョブファイルの作成と制御部への処理依頼

### (5) 制御部 (Qdai Input Control Program)

- ① 初期設定部およびプリスタック入力部より依頼されたジョブの登録と一定のアルゴリズムにより DAIN への処理依頼
- ② ジョブの状態の管理
- ③ 仕分情報作成部および操作部より依頼されたジョブファイルのスクラッチ処理
- ④ プリスタック入力部へ優先権あるいは機番別にジョブの情報の通知

### (6) 仕分情報作成部 (CLoSe Line Printer output module)

- ① 制御部へのジョブファイルのスクラッチ処理依頼

### (7) リモートバッチ入力部 (Remote INput module)

- ① 端局からのジョブの入力と、大記憶へのジョブファイルの作成と制御部への処理依頼

### (8) 大記憶システム入力プログラム (DA INput module)

- ① 大記憶のジョブファイルを読み、解釈とジョブの制御表およびデータファイルの作成
- ② バッチジョブ制御プログラムへ実行依頼

## 5. 制御語と制御表

以下に述べる制御語と制御表はプリスタック入力システムで特に使用するものです。



プリスタック入力部は、この装置表から使用可の装置機番を捜し、Q.QIUK からの相対アドレスを求めます。制御部は、DAIN に処理依頼する時相対アドレスより機番を求めることになります。

(4) プリスタックジョブ優先権表とジョブ番号表

プリスタックジョブ優先権表は、各優先権ごとに DAIN に依頼できるジョブ数の最大値、現在依頼しているジョブ数、また当該優先権にジョブがある場合には、ジョブ番号表の先頭アドレスが記入されています。ジョブ番号表はジョブの情報と状態が設定されています。プリスタックジョブは、この2つの表で管理されています。ジョブ番号表のことを以後 JNLIST と呼びます。

		0	5 6	11	16	35	
Q.QIJPTBLKEY	この表の変更・参照時のキー						
Q.QIJPTBL	+0	処理依頼最大値	現在処理依頼数	0	優先権0のジョブ番号表のアドレス		ジョブ番号表がない場合には0が設定されています
(グローバルな名札)	+1	"	"	0	" 1	"	
	+2	"	"	0	" 2	"	
	+3	"	"	0	" 3	"	
	+4	"	"	0	" 4	"	
	+5	"	"	0	" 5	"	
	+6	"	"	0	" 6	"	
	+7	"	"	0	" 7	"	

表 5.2 ジョブ優先権表

JNLIST	+0	①pointer	②count	0	③JNLISTNXT	
(この表は必要になった時に確保され不要になったら返却されます)	1					①pointer 次に登録する位置 ②count 登録されているジョブ数 ③JNLISTNXT 次のJNLISTのアドレス ない時は0が設定されています。 ④JOBINF (2)のジョブの情報 JOBINFのない箇所は0 が設定されています。
	2					
	3					
	4					
	5	④JOBINF				
	≈					
	≈					
	15					

表 5.3 ジョブ番号表

DAIN へ処理依頼する場合には、優先権の高い方から（優先権7→0）、同一優先権のものは古い JNLIST（優先権表へ近い方につながっている）の古いジョブ（JNLIST の頭の方）から行なわれます。

(5) プリスタック入力プログラム接続表とプログラム表

プリスタック入力プログラム接続表は、プリスタック入力部のプログラム表を接続するためのものです。同一のプログラム名のもものが接続されないようにしています。プログラム表は、プリスタック入力部が処理するのに必要となる制御情報を表にまとめたものです。

Q.QIODLKEY	この表変更・参照時のキー		
Q.QIODL	0	プログラム表先頭アドレス	プログラム表が定義されていない時は0が設定されています。
(グローバルな名札)			

	0	15	16	35			
QIDCB	0		① QIDCBNXT		①QIDCBNXT QIDCBの次のアドレス、 ない場合0		
(プログラム表、 1 起動前に作成さ 2 れ、終了時に領 3 域が返却されま す)	QIDCBNM (プリスタック入力名)				②QIDCBTYP タイプライタ機番 ③QIDCBS		
	8	9	空	17 18	30	35	9 : 1...入力1時停止 10 : 1...入力終了
	② QIDCBTYP		③ QIDCBS		0	④QIDCBPRTY	④QIDCBPRTY
	0		⑤ QIDCBIJS		アドレス		31 : 1...Xジョブを集団ジョブファ イルに移す
	0		⑥ QIDCBIOB		アドレス		32 : 1...Aジョブ #
	0		⑦ QIDCBOJS0		アドレス		33 : 1...Bジョブ #
	0		⑧ QIDCBOJS1		アドレス		34 : 1...Cジョブ #
	0		⑨ QIDCBUCB		アドレス		35 : 1...Dジョブ #
	QIDCBSJN (開始ジョブ番号)				⑤QIDCBIJS 入力JSIOCB先頭アドレ ス		⑥QIDCBIOB 入力がカードリーダの場 合IOB先頭アドレス
	QIDCBEJN (最終ジョブ番号)				⑦QIDCBOJS0 集団ジョブファイル用 JSIOCB先頭アドレス		⑧QIDCBOJS1 ジョブファイル用 JSIOCB先頭アドレス
	空				⑨QIDCBUCB 入力がカードリーダの場合 UCB先頭アドレス		
	空						

表 5.4 プログラム表

注5)  
(5) QCB

	0	8	9	16	35
Q.QIQUEUE (グローバルな 名札)	個 数		0		
	0	1	0	待ち行列先頭番地	
	0		待ち行列最終番地		

表 5.5 QCB

これは、QICP が DAIN に情報を与えるための QCB です。QICP は、DAIN に処理依頼が可能なジョブが発生すると、ジョブファイルや装置に関する情報を表にまとめた SIDCB (System Input Define Control Block) のアドレスをキーにしてこの QCB に接続します。DAIN は、モニターよりこのことを通知されると起動され、SIDCB より必要な情報を取り出してジョブファイルの処理をします。

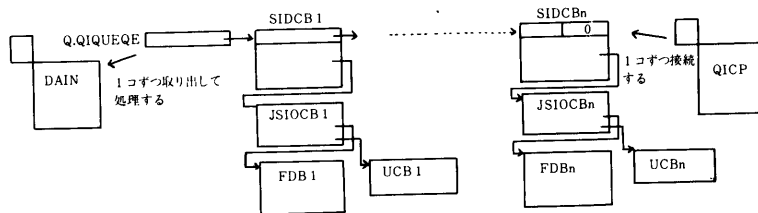


図 5.1 QCB 接続図

JSIOCB : 入出力に関する制御情報をまとめたもの  
FDB : ファイルに関する制御情報をまとめたもの  
UCB : 装置に関する制御情報をまとめたもの

注5) 「FACOM230-60 システムマクロ文法編」参照

6. モジュール詳説

6.1 QI モジュール

これは、操作員がプリスタック入力部を起動したい時にタイプライタより以下の形式でタイプインします。

(1) タイプイン形式

	指 令 名	オ ペ ラ ン ド
RQ	QI-	① プリスタック入力名, [ ② 入力装置機番 ], [ ③ 入力装置ボリューム通番 ], [ ④ 入力ファイル名 ], [ ⑤ ジョブ種別 ], [ ⑥ 出力装置機番 ], [ ⑦ 出力装置ボリューム通番 ], [ ⑧ 出力ファイル名 ] [ , ⑨ ジョブ番号列 ]

[ ] のパラメータは省略可

RQ タイプライタのRQキーを押すことを示します

表 6.1 QI タイプイン形式

(2) オペランドの説明

- ① プリスタック入力名      英字で始まり8文字以内の英数字とピリオドで入力名を与えます。頭文字がLで始まる時には、プリスタック入力部(QSYSIN)をLCMにロードします。
  - ② 入力装置機番
  - ③ 入力装置ボリューム通番
  - ④ 入力ファイル名
  - ⑤ ジョブ種別
  - ⑥ 出力装置機番
  - ⑦ 出力装置ボリューム通番
  - ⑧ 出力ファイル名
  - ⑨ ジョブ番号列
- ② ③ } 入力装置の機番とボリューム通番を指定します。
- ④ } 入力が大記憶および磁気テープの集団ジョブファイルの時、そのファイル名を与えます。
- ⑤ } ⑥ ⑦ ⑧ のパラメータの指定がある場合、BCD(ジョブ種別B, C, D)というふうにジョブ種別のつながりで指定すると指定された種別のジョブは⑥ ⑦ ⑧ の出力ボリュームのファイルに集団ジョブファイルとして作成され、それ以外のジョブはジョブファイルとして大記憶に作成されます。
- ⑥ } 出力が大記憶のジョブファイルのみの場合は指定する必要はないが、集団ジョブファイルも作成したい場合には装置機番・ボリューム通番、集団ジョブファイル名を指定します。
- ⑨ } [開始ジョブ番号] [, 最終ジョブ番号] という形式で指定します。入力が集団ジョブファイルの時のみ指定可能で、指定された開始ジョブ番号から最終ジョブ番号までのジョブが処理の対象となります。

(3) 処理内容

以上のオペランドのパラメータを解釈し QIDCB と付属の制御表を作成して、誤まりがなければ、

EXEC マクロ命令<sup>注5)</sup>によりプリスタック入力部 (QSYSIN) を主記憶にロードして実行を依頼します。パラメータの指定とプリスタック入力部の機能は QSYSIN の項を参照してください。

## 6.2 QIO モジュール

これは、以下の形式でタイプライクよりタイプインすることにより起動されます。

### (1) タイプイン形式

	指 令 名	オ ペ ラ ン ド
RQ	QIO	① プリスタック入力名, { YAMEYO TOMARE HAZIME } ② DSPLY, { JOB, ジョブ番号 } { PRTY, 優先権 } { UNIT, 機番 } ③ SCRH, ジョブ番号 ④ CHAP, ジョブ番号, 優先権 ⑤ INIT, 優先権, 個数 ⑥ { MNT } { DMNT }, 機番

{ } のパラメータはこの中のどれかを必ず指定します。

表 6.2 QIO タイプイン形式

### (2) オペランドの説明

#### ① プリスタック入力名

, YAMEYO プリスタック入力部は入力をやめ、プログラムの領域を解放します。

, TOMARE プリスタック入力を一時停止します。HAZIMEにより再開します。

② DSPLY , JOB, ジョブ番号 指定されたジョブ番号の優先権, 機番, ジョブの状態をタイプライタに表示します。

, PRTY, 優先権 指定された優先権に属するすべてのジョブのジョブ番号, 機番, 状態を表示します。

, UNIT, 機番 指定された機番の装置に属するすべてのジョブのジョブ番号, 優先権, 状態を表示します。

③ SCRH, ジョブ番号 指定されたジョブ番号のジョブを JNLIST から削除し、大記憶のジョブファイルをスクラッチします。

④ CHAP, ジョブ番号, 優先権 指定されたジョブ番号のジョブを指定された優先権の JNLIST に移します。

注5) 「FACOM230-00 システムマクロ文法編」参照



- ① ジョブの種類 00：ローカルバッチジョブ 10：リモートバッチジョブ  
11：デマンドジョブ
- ② 処理形状
- 100000：QON JNLIST にジョブを登録し，DAIN への処理を依頼します。
- 110000：ALCT JNLIST にジョブは登録するが，ジョブの状態はアロケイトのみとし，DAIN への処理依頼はしない。
- 010000：QOFF DAIN での処理が済んだのでジョブの状態を DAIN 処理済とします。
- 001000：SCRH JNLIST からジョブを削除し，大記憶のジョブファイルをスクラッチします。
- 001100：GETJ1 アロケイト済のジョブの中で，指定された機番に属するジョブ番号の通知をします。
- 001101：GETJ2 アロケイト済のジョブの中で，指定された優先権に属するジョブ番号の通知をします。
- 001110：GETJ3 アロケイト済のジョブの中で，ジョブ番号が一致するジョブの情報を通知をします。

## 7. おわりに

このシステム設計時の企画にたずさわっていただいた竹下<sup>※</sup>節子・大石<sup>※</sup>一夫・石田<sup>※</sup>いつ子およびこの計画実現に協力いただいたセンター職員の方々ならびに富士通 S E ・ C E 各位に感謝します。このシステムは運用面での考慮やリモートバッチのインターフェイスが未修正ということもあって実質的運用まではなおしばらく時間がかゝると思われまゝ。私達を含めて，センター開所以来一連の運用管理システムの開発の任にたずさわってきた者は，当センター設立時の目標でありました<sup>※注6)</sup>

- (1) 出来るだけ多数の人が気軽に，便利にセンターを利用できること。
- (2) センター職員が働きやすいこと。
- (3) 運用結果が将来の設計，運用にフィードバックできること。

の達成を目指して頑張ってきたつもりであります。また計算機センターのオペレーティングシステムのあり方に我々なりの一石を投じてきたと自負しておるものであります。MV 中での当センターのシステム開発（制御プログラム部分）はこれをもって最後になると思われまゝですが，(1)～(3)の目標を達成するために我々は今後より一層努力を続け，メーカー側にも厳しく要望を出していく考えです。最後にシステムの開発に際して，当初より貴重な御意見と温い御支援を寄せてくださった利用者の皆様ならびに大野克郎<sup>※※</sup>前センター長，高田勝現<sup>※</sup>センター長に感謝の意を表します。（宇津宮記）

※ 九州大学大型計算機センター

※※ 九州大学工学部

注6) 九州大学大型計算機センタ広報研究開発の欄「vol 1, No. 1, No. 2, vol 5, No. 2, No. 3, No. 4, vol 6, No. 3, No. 4」参照