

JAERI-M

5657

FACOM 230-60システムの
CPUアイドルの解析

1974年4月

石黒美佐子・山田 孝行

この報告書は、日本原子力研究所が JAERI-M レポートとして、不定期に刊行している研究報告書です。入手、複製などのお問い合わせは、日本原子力研究所技術情報部（茨城県那珂郡東海村）あて、お申しこしてください。

JAERI-M reports, issued irregularly, describe the results of research works carried out in JAERI. Inquiries about the availability of reports and their reproduction should be addressed to Division of Technical Information, Japan Atomic Energy Research Institute, Tokai-mura, Naka-gun, Ibaraki-ken, Japan.

FACOM 230-60 システムのCPUアイドルの解析

日本原子力研究所東海研究所原子炉工学部

石黒美佐子・山田孝行

(1974年3月4日受理)

日本原子力研究所(以下原研と書く)における電子計算機システムのCPU利用状況を調査研究するために、CPUアイドルが測定され解析された。

CPU利用状況の詳細は、(1) ユーザCPU使用時間、(2) システム・オーバヘッド、(3) I/O待時間、(4) アイドル時間 に分けて示され論じられる。CPUアイドル率が約35%に達していることがわかる。CPUアイドルの原因を知るために、関連するデータが、ソフトウェア測定や会計情報から集められ、回帰分析によってシステムの一連の事象が統計的に解析された。

さらに、周辺機器の使用率を、スケーラを使用頻度の高い機器に結合して、ハードウェア的に測定することを試みた。

全体を通して多くの図表が示されているので、原研における最近の電子計算機の利用状況がわかる。

Analysis of the CPU Idle Time of Computer FACOM 230-60

Misako ISHIGURO and Takayuki YAMADA

Division of Reactor Engineering, Tokai, JAERI

(Received March 4, 1974)

CPU idle was measured and analyzed to examine the CPU utilization of JAERI computer systems. The CPU utilization is described in detail, dividing it in four states: (1) user CPU time, (2) system overhead, (3) I/O wait time, and (4) idle time. The CPU idle time is seen to be about 35%. To find out the causes of CPU idle, the concerned data are gathered by software measurement and accounting information. And a series of events in computer system are statistically analyzed by the regression method. Hardware measurement is also attempted to count the device utility by connecting the scalars to the computer devices frequently used. Many figures and tables are given to facilitate the review of computer usage in JAERI.

目 次 な し

はじめに

現在、計算センタには、F 230-60 システムが 2 システム導入されていて、各システムが 2 つの CPU と、主メモリ 256 K 語 (+ 131 K 語の大容量記憶装置)、さらに磁気ドラム 7.5 M バイト、磁気ディスク 460 M バイトと周辺機器を持ち、各システムは、周辺機器を除いて同一の機器構成を持っている。

一方のシステムを 1 システムと呼び、オープン・バッチ処理で運転され、もう一方のシステムを 2 システムと呼び、オペレータによるクローズ・バッチ処理で運転されている。

これらのシステムは、多重ジョブ処理方式のオペレーティング・システム M-V のもとで動いている。このような処理方式は、CPU をはじめ高価な計算機の機器を効率よく使う上で優れているとされている。しかしながら、原研システムでは、CPU が効率よく利用されているかどうかについて以前から疑問とされているところであった。今回、CPU のアイドル状況を測定した結果、1 システム、2 システム共に 35% 程度のアイドルを確認した。

CPU アイドルをもたらす要因は多く、それらが互いにかみあって、データを収集してもそれらを取りまとめて結論を導き出すのは難しいことである。

計算機システムを評価するのによく使われる方法の一つに、解析的方法と呼ばれるものがある。計算機システムを統計的にモデル化して、待行列の理論により解析的な解を得る方法である。この方法の欠点は、単純なモデルに対してしか適用できないので、得られた結果が現実を反映しない場合も起り得ることである。

他によく使われる方法は、シミュレーションである。この場合は複雑なモデルを取扱える点ですぐれているが、有効に利用できるシミュレーション言語がなかったり、計算時間がかかりすぎて実行不可能な場合も起る。

これらの方法は、新しく開発するシステムの性能予測によく使われている。

著者がここで用いたのは、多変量の回帰分析による方法で、この方法は手軽に実行できる上に、収集したデータに対して全体的な結論を下せる点ですぐれている。

原研システムのジョブ・パターンとシステムの運用状況を把握するために、ユーザ CPU 使用時間、システム・オーバヘッド、CPU アイドル時間、ジョブのサーヴィス分布、到着分布、ターンアラウンド・タイム、ジョブの実行多重度、ロールアウト率、データ・チャンネル使用率、磁気ドラムの使用状況、さらにジョブごとのデータとして、出力量、メモリ使用容量等の統計資料を得た。これらは、CPU アイドル時間の測定、装置の使用回数、使用時間のハードウェア的測定とアカウンティング・ファイルからのデータ収集によって得られたデータを整理したものである。これらの基礎資料から、回帰分析の方法を用いて、CPU アイドルとシステム・オーバヘッドの原因を明確にしていく。さらに、現状 (1973 年前期) のシステム運転状況とジョブ・パターンを前提として、CPU アイドル、入出力等待時間、システム・オーバヘッドがユーザ CPU 使用時間に対してどのように変化するかを図によって示す。

1. CPUアイドル原因についての考察

まず、CPUアイドルの原因となることを順にあげる。

- (1) 実行すべきジョブがない。
- (2) ユーザがシステムの処理能力に見合った利用をしていない。
- (3) 同じ傾向のジョブばかり流すので装置の奪い合いが起る。
- (4) オペレーティング・システムのジョブ・スケジューリング、タスク・スケジューリング、資材の割当等が悪い。
- (5) ハードウェアの機器構成に難点がある。

以上の事項を確認するために、既に述べた方法でデータ収集し、統計整理を行った。その結果についての詳しい説明に先立って、現在のシステムに対する結論を先に述べる。

(1)に関しては、まず $\#1$ システムにおいては、オープン・バッチ処理を行っているので、ジョブ入力がユーザ任せとなり、昼食時や、システム開始時は、ジョブの入力が少くなる。10時から12時までと14時から16時までが混雑する時間帯である。

$\#2$ システムでは、クローズ・バッチ処理でオペレータによる24時間運転をしている。ジョブの受付時間は9時から20時となっている。ジョブを受付次第、オペレータはシステムに入力する。翌朝までに受けたジョブの処理をすべて完了してしまう場合が多い。

(2)については、 $\#1$ システムの運用が短いジョブを対象としていて、負荷が少なすぎる傾向にはあるが、オープン・バッチ処理においては、ターンアラウンドを早くすることも、システムを評価する上での重要な項目であるから、そのためにCPUアイドルが多くても止むをえないという考え方もある。

$\#2$ システムでは、夜間は多重度を下げて、大きなジョブばかりを流しているので、CPUアイドル率が高い。これは(3)および(5)で述べるメモリの不足と入出力機器使用の競合により、多重度を上げられない事情が一方では存在する。

(3)は、原研においてCPUアイドルを招く大きな原因となっている。 $\#1$ システムでは、短時間ジョブが対象なので、プログラム作成・ディバックに使用されることが多い。そのため、FORTRAN コンパイルや、プログラム・リンケージ等のジョブ・ステップの比率がかなり高くなる。特にFORTRAN ライブラリの参照回数が多く、同一装置、すなわちドラムへのアクセスの競合が起る。その他に、コントロール・カードの処理、ジョブ・ステップの起動/停止処理のようなシステム介入の機会が増え、いわゆるI/Oバウンドの状態におちいることが多い。著しいのは、ドラムへのアクセスで、ドラムが結合されているセクタ・チャンネルとファイル・コントローラが一つしかないために、これらの使用率がきわめて高くなってくる。このこともすでに報告されている。

$\#2$ システムにおいては、長時間ジョブが多いので、メモリ使用容量が大きく、ロールアウト率が高くなっている。一方、磁気テープの台数が不足して、実行準備を完了したジョブが装置待ちの状態で待たされたり、XYプロッタを使用するジョブの競合が起ったりする。これらの理由により実行多重度が上らず、CPUアイドルをもたらすことになる。

(4)はソフトウェア上の問題である。ジョブ・スケジューリング上の問題点は、システムの資料を使わないジョブ程早いターンアラウンドを与えるという、原研におけるプライオリティ設定基準が、もし短時間ジョブと、長時間ジョブを同一のシステムで処理するとしたら、保証されなくなる。ロールアウト・ファイル領域は有限なので、短いジョブが入力されるたびに多重度をどんどん上げていくことは現実には不可能である。結局、多重度を一定に保つことになる。その場合、長時間ジョブがいったん開始されると、短時間ジョブの実行開始が妨げられ、短時間ジョブのターンアラウンドを乱すことになる。ジョブ・クラスという考え方を導入することにより、最近の大型計算機では、これを解決している。現状では、2つのシステムがそれぞれ分担を分けた運用をするしか方法がない。

メモリの空領域の管理がダイナミックに行われないので、こまぎれの空領域が増えるのもメモリ有効利用の上で問題がある。現在のシステムを多少改良して、空領域管理をダイナミックに行おうとすれば、システム・オーバヘッドが大きくなり、適用基準に検討を要することが京大で報告されている〔1〕。原研でも№1システムに対して、ある基準を設けて空領域の一掃を試みたがあまり効果が上らなかった。その理由は、ジョブ・ステップごとのユーザCPU使用時間が少ないので、システム内での事象の変化が激しく、たえず主メモリと補助記憶間をプログラムが移動することになり、オーバヘッドを増加させてしまうからである。№2システムでは、事象の変化がそれ程激しくないので、空領域を一掃して新しく一まとまりの領域を得ることは効果がある。現在おこなっている方法は、空領域管理プログラム(1K語)によって、一定時間(約30秒)ごとに、領域使用状況をコンソールに打出し、オペレータが入力されているジョブのパターンを見て空領域の有効利用が可能と判断される場合のみ、コンソールからその由指令を入れるようになっている。

空領域管理プログラムは、実行多重度を他のジョブより低くし、通常は定時間間隔で起動がかけられている。オペレータから空領域一掃の指令が入力されると、実行多重度を高くすると共に、ダイナミックに全領域(№2システムでは200K語)を確保する。そうすることにより、実行中のジョブがすべてロールアウトされる。その後直ちに、空領域管理プログラムの実行多重度をもとどおり低くし、メモリも開放する。そうすれば、いったんロールアウトされたジョブが、メモリの上部から順序よく配置され、一まとまりの空領域が確保される。

この方法適用後の効果についてまだとりまとめていないが、このような方法をいくら試みてもメモリの管理が根本的によくなる。これらを解消するために最近の計算機では、仮想記憶方式が採用されるようになった。この方法によれば、プログラマは、見せかけの大きな空間を使用して、実装されているメモリの大きさを意識しないでプログラムを書くことができる。

XYプロッタの処理方式を磁気テープ経由ではなしに、システム出力ファイル式に大記憶を経由するように変更すれば、XYプロッタを使用するジョブを2つ以上共存させて流すことが可能となる。

(5)については、№1システムでは、ドラムに結合されるデータ・チャネルを高速化すべきである。〔*注1〕

№2システムでは、メモリ不足の原因は、F230-60システムが実装可能なメモリ容量の上限が256K語であるということである。2-CPU構成のシステムでは、CPUの処理能

力が高すぎて、メモリ容量との釣合がとれないと思われる。システム常駐部が60K~80K語も占めるのは、メーカ側の善処をうながしたいところである。

導入時に、両システムを対称にすることに重点を置きすぎて、各システムの使用目的、運用の差違についての考慮が不足していたと著者らは反省している。CPU使用率から見れば、1システム構成で、変化のあるジョブ・パターンを処理する方が有利であると考ええる。

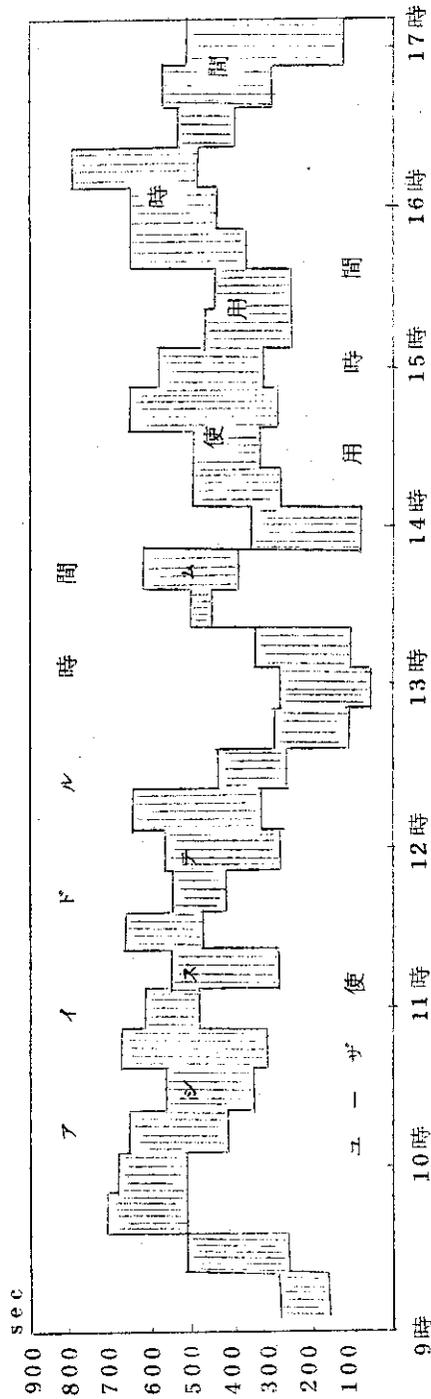
注1 49年2月の機器交換によって、ドラムは高速セクタ・チャンネルに接続されることとなり、ドラムもアクセスの速いものに交換される。

2. CPUのアイドル状況

第1図は、1972年12月15日(金)に、 $\#1$ システムのCPUのアイドル状況を測定した結果を図示したものである。図の下側は、ユーザがCPUを使用した時間、上側がアイドル時間、中間の斜線部分は、オペレーティング・システムが使用した時間でありシステムのオーバヘッドと考えられる。一見してわかるように、計算機の稼働時間のうち、実際にユーザがCPUを使用している比率はかなり低いことがわかる。この日は、どちらかと言えばジョブの少ない日であるが、CPUのアイドル率が40%を越えていることに注目するべきであろう。これは、9時から午後5時迄を、15分間隔で測定し、約15分毎のCPUアイドルの変動を示している。システムの初期設定時と昼食時間はジョブが少ないので、CPUのアイドル率が高くなることも図からわかる。

CPUのアイドル状況をもっとマクロに測定した結果が第2図である。第1図と同じく下がユーザ使用時間、上がアイドル時間、中間がシステムの使用時間である。1973年6月14日(木)の9時25分から11時40分までの間を約40秒間隔で測定したものである。CPUアイドル等の状態の変化がかなり激しいことがわかる。下側の○印は、メモリ内の実行タスクの多重度を示している。5多重で運用しているが平均値は4.2となっている。ユーザが入力したジョブ以外にシステム・タスクが1つ加わるので、実際は6になる場合もあり、逆に、ジョブがロールアウトされている場合は、実行多重度を含めないで4以下となる場合もある。ソフトウェア的測定法を用いたので測定プログラムが常駐しているが多重度からは取除いている。後に示すように $\#1$ システムのCPUアイドル率の平均は38%なので、第2図の結果は平均よりCPUのアイドルが少ないが、混雑する時間帯を選んで測定したことによるものである。

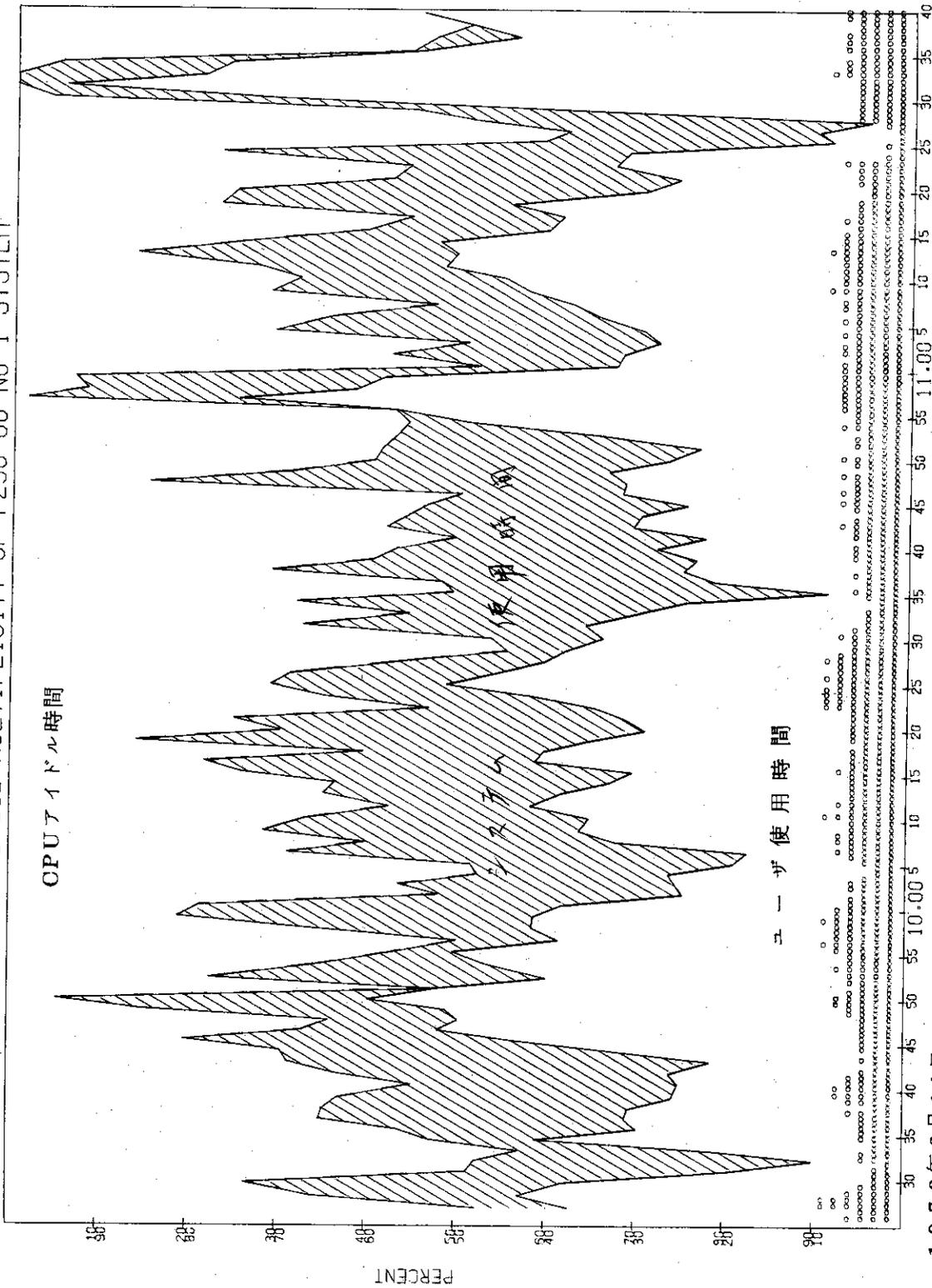
CPUのアイドル状況をもっとマイクロに測定した結果が第1表と第2表である。第1表は $\#1$ システムに対するもので、1973年6月から7月にかけて、2時間から8時間を測定間隔としてデータを得たものである。第2表は $\#2$ システムに対するもので、7月と8月に測定した結果である。これらの表から、両システムのCPUアイドル率を大局的に知ることができる。いずれのシステムでも、アイドル率が全体の3分の1以上を占める。



ユーザ使用時間平均 33.8 %
 システム使用時間平均 25.6 %
 アイドル時間平均 40.6 %

第1図 CPUアイドル状況 (1)
 (1972年12月15日測定)

CPU UTILIZATION AND JOB MULTIPLICITY OF F230-60 NO 1 SYSTEM



ユーザ使用時間平均 37.7%
 システム使用時間平均 26.7%
 アイドル時間平均 35.6%

第2図 CPUアイドル状況(2)

1973年6月14日

No.1 システム

DATA 1973

DATE	START	END	INTERVAL	USER TIME	0.43%	SOFTWARE TIME	0.22%	IDLE TIME	JOBS	INPUT	MACRO	CPU USED	OUTPUT	JOBSTEP	
1	0625	1135	0	1531	0	12090.23	0.43%	5017.72	0.33%	198	1625	12090.23	225155	572	
2	0625	1531	0	1712	0	4873.76	0.23	1415.13	0.36	92	817	4873.76	90649	300	
3	0626	919	3	12	014	6497.63	0.37	2763.34	0.39	168	1530	6497.63	186598	484	
4	0627	92334	1832	6	23712.00	18776.59	0.26	9580.07	0.37	401	3372	18776.59	448331	1268	
5	0628	9	939	114719	9460.00	6123.72	0.32	4345.17	0.46	128	1098	6123.72	106245	331	
6	0628	114719	13228	6309.00	5035.85	0.40	1011.83	2779.24	0.44	35	25049	5035.85	39827	126	
7	0628	133228	16	240	9012.00	7204.94	0.40	3050.20	0.34	149	1368	7204.94	176820	443	
8	0629	947	9	171519	26890.00	2319.73	0.44	8789.15	0.33	400	3638	2319.73	319856	1108	
9	0702	101216	114828	5772.00	3633.17	0.31	1462.88	2492.53	0.43	100	826	3633.17	78132	276	
10	0702	114828	131930	5462.00	2722.19	0.75	1248.17	2852.73	0.52	196	296	2722.19	58550	87	
11	0702	131930	17	045	13275.00	6077.61	0.23	7866.36	0.51	156	1354	6077.61	110770	382	
12	0703	10	250	172637	26627.00	15893.92	0.30	11876.90	0.45	312	2716	15893.92	329731	944	
13	0706	941	3	164237	25294.00	17775.10	0.35	10208.19	0.40	333	3232	17775.10	332639	913	
14	0707	92619	113710	7971.00	5960.68	0.37	2013.65	7277.01	0.37	128	1181	5960.68	113353	340	
15	0709	102311	162468	2177.00	19365.86	0.45	4667.58	2828.69	0.34	286	89243	2610	14365.46	287534	798
16	0710	93648	162339	24411.00	27481.83	0.26	1603.05	5712.45	0.23	346	156996	3167	27401.43	393875	932
17	0711	93320	114013	5416.00	9163.68	0.60	1263.27	1428.11	0.19	156	1542	9163.68	128782	337	
18	0711	114013	131029	7643.00	2883.29	0.27	1603.05	2111.09	0.50	149	514	2883.29	41415	200	
19	0711	131029	162442	11653.00	11134.86	0.48	2508.21	3577.36	0.31	173	1667	11134.86	136210	420	
20	0712	93916	121653	9457.75	7559.75	0.40	2224.72	3432.40	0.37	148	1598	7559.75	127464	366	
21	0712	121653	131335	3402.00	1116.31	0.16	941.28	1802.57	0.56	23	11289	1116.31	58340	74	
22	0712	131335	1619	9	11134.00	9167.70	0.41	2518.14	0.23	164	1490	9167.70	140990	444	
23	0713	1022	0	115333	5893.00	0.41	1085.34	1030.30	0.19	107	965	6754.71	87055	266	
24	0716	93054	12	533	9279.00	7505.84	0.40	2327.15	0.25	133	1165	7505.84	134904	384	
25	0716	12	333	13534	6001.00	3664.27	0.31	1540.00	0.26	65	688	3664.27	95876	176	
26	0716	134534	152730	6016.00	6864.87	0.26	1213.08	1470.47	0.24	96	885	6864.87	59709	316	
27	0717	94838	114326	6888.00	6320.79	0.46	1594.87	2132.73	0.31	118	30802	6320.79	93251	337	
28	0717	114326	131010	5204.00	2063.28	0.20	1361.97	2807.90	0.54	44	418	2063.28	44554	137	

第1表 CPUのアイドル状況(4)
(1973年6月, 7月測定)

No.1 システム

ユーザー使用时间平均 38.5%
システム使用时间平均 23.5%
アイドル時間平均 38.0%

FILE	CORE	TURNAROUND(MEAN,SD)
1	112329	0.15E 07 617.7 643.8
2	161345	0.64E 06 809.5 836.4
3	124317	0.11E 07 737.7 770.2
4	317863	0.29E 07 634.8 594.1
5	136993	0.87E 06 731.5 864.1
6	46008	0.67E 06 603.4 596.1
7	121490	0.12E 07 804.4 1081.4
8	240457	0.27E 07 609.5 633.5
9	51279	0.48E 06 453.4 408.2
10	34436	0.34E 06 411.3 514.1
11	70037	0.98E 06 411.3 514.1
12	217801	0.21E 07 439.1 345.1
13	245129	0.26E 07 746.8 968.7
14	95857	0.84E 06 418.6 389.6
15	266957	0.23E 07 1094.9 1152.2
16	225829	0.31E 07 840.5 1808.4
17	66043	0.92E 06 1042.5 1660.9
18	59889	0.11E 07 534.5 389.9
19	115315	0.11E 07 1158.6 1420.8
20	88025	0.93E 06 493.1 436.1
21	35271	0.15E 06 372.3 183.4
22	127153	0.98E 06 523.9 456.7
23	47848	0.85E 06 928.5 1137.9
24	118487	0.82E 06 438.8 376.7
25	36562	0.44E 06 523.4 437.6
26	67046	0.79E 06 1148.0 1261.3
27	83477	0.95E 06 772.7 792.5
28	36497	0.35E 06 370.1 303.4

DATA 1973 No.2 システム

FILE	DATE	START	END	INTERVAL	USER TIME	SOFTWARE TIME	IDLE TIME	JOB TIME	JOB	INPUT	MACRO	CPU USED	OUTPUT	JOBSTEP
1	0717	91625	131045	14030.00	16066.63	0.57	4922.18	0.35	39	25304	495	16066.63	53481	108
2	0717	131045	154623	9338.00	11370.95	0.61	1997.56	0.21	26	13618	182	11370.95	35663	62
3	0718	92110	134830	16040.00	14471.70	0.65	6000.13	0.37	63	26948	1007	14471.70	99000	150
4	0718	134830	162042	1432.00	12501.75	0.66	2120.40	0.23	33	9926	278	12501.75	67454	68
5	0719	150300	172140	5470.00	7319.01	0.67	1493.18	0.27	29	8912	273	7319.61	61977	65
6	0720	92318	131920	14162.00	14864.71	0.52	5030.73	0.36	48	11810	557	14844.71	71230	103
7	0721	93924	119521	8157.00	6769.07	0.54	2888.06	0.35	36	8922	425	8769.02	54805	110
8	0725	1015 0	1642 0	23220.00	25081.40	0.54	8213.85	0.35	86	21774	980	25081.40	134887	208
9	0727	934 0	1137 0	7380.00	8235.74	0.56	2437.42	0.33	47	12938	435	8235.94	46215	90
10	0806	112712	13 749	6837.00	7462.23	0.62	1945.94	0.32	12	5044	121	7462.23	48223	41
11	0806	112712	1443 1	5137.00	5749.36	0.50	2121.96	0.37	34	9534	467	5749.36	48858	122
12	0806	1443 1	155141	4120.00	2966.26	0.39	1780.72	0.43	14	6765	150	2966.26	45350	36
13	0807	105026	165013	21587.00	28765.54	0.67	5334.16	0.25	57	38751	730	28765.54	139989	157
14	0808	93717	131744	15227.00	17213.34	0.65	3836.27	0.26	36	19285	503	17213.35	96762	125
15	0808	131744	144039	9875.00	4443.23	0.63	974.87	0.20	30	17627	404	4443.23	76956	61
16	0827	91294	113920	8786.00	6738.33	0.38	1171.75	0.13	34	11725	464	6738.33	40262	138
17	0827	113920	134050	7490.00	4315.65	0.30	1202.07	0.16	14	3366	230	4315.45	37972	48
18	0827	134050	154647	7557.00	8074.42	0.53	3930.20	0.54	21	13950	324	8074.82	45991	93
19	0828	91125	114329	9136.00	10386.36	0.57	1317.38	0.14	36	17809	520	10384.36	36928	111
20	0829	92639	114951	8592.00	8009.96	0.47	2624.44	0.29	37	22858	345	8009.99	44710	98
21	0829	114951	148023	10232.00	11725.35	0.57	3294.29	0.32	32	18996	345	11725.35	45000	87
22	0830	132621	17 235	12852.00	12269.35	0.48	4506.88	0.35	57	27754	776	12289.35	129141	206
23	0831	95340	112615	8673.00	6673.00	0.55	975.70	0.11	29	18586	375	9470.82	44780	87
24	0831	115813	142622	8869.00	7270.01	0.41	3946.68	0.44	42	17482	404	7270.01	59280	114
25	0831	142622	165243	8781.00	11957.56	0.66	1911.39	0.22	41	16134	342	11957.56	64057	94

第2表 CPUのアイドル状況(4)
(1973年7月, 8月測定)
No.2システム

ユーザ使用時間平均 53.3%
システム使用時間平均 12.5%
アイドル時間平均 34.2%

FILE	CORE	TURNAROUND(MEAN,SD)
1	163351	4135.2
2	63019	7505.0
3	96073	4866.4
4	57620	2558.2
5	46588	2652.0
6	120797	3874.9
7	108192	2894.2
8	176496	5184.8
9	64063	4777.3
10	14753	9104.5
11	77444	4428.8
12	43105	2950.1
13	223479	5967.5
14	224934	6465.2
15	119281	4281.4
16	67065	3042.7
17	21763	2278.4
18	58958	807.2
19	50117	505.6
20	98455	5215.7
21	95221	2802.1
22	210447	4397.5
23	64081	1694.3
24	43959	1424.5
25	51078	3511.5
		4058.2
		5406.1
		1993.0
		2376.3
		2228.9
		1044.1
		4607.6

3. CPU使用時間、システム・オーバヘッドの測定方法

CPUは、システムに電源が投入され、STARTキを押されると休みなく働き続ける。CPUに電源が投入されている間をCPUの稼働時間と呼ぶことにする。

原研のシステムは、 μ 1システム、 μ 2システム共に2つのCPUで構成されている。2つのCPUがユーザのための仕事のみをすると仮定すれば、CPUの稼働時間の2倍がユーザ使用時間となる。しかしながら既に述べたようにその何割かの部分が、システム・オーバヘッドで、残る何割かはアイドル状態となっている。F230-60のM-Vモニタでは、システムの仕事として数えられるものの中に、ジョブの入力処理、ラインプリンタ等への出力処理、コントロール・カードの処理、ジョブ・ステップの起動/停止、実行時の入出力処理等ユーザの仕事としてカウントしてよいものまでが含まれている。これらの時間をユーザ使用時間として取扱うには、オペレーティング・システム設計上で難しい点もあろうが、多重タスク処理方式のシステムでは、同じ仕事に対する処理時間が一定でないので、アカウントできない面もあるのではないかと考えられる。システムの仕事としてこの他にジョブ・スケジュール、タスク・スケジュール、メモリ割付等純粋なシステム・オーバヘッドといえるものも含まれている。

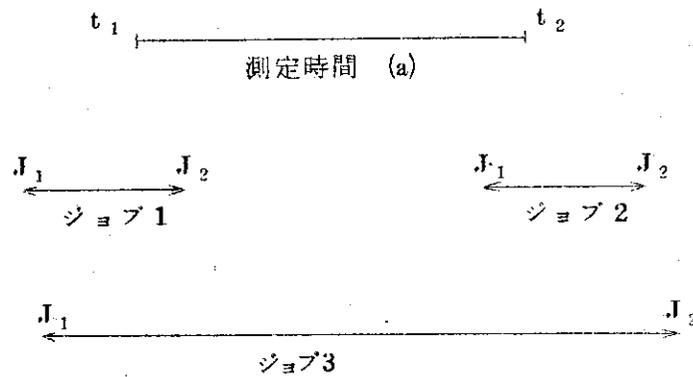
F230-60のM-Vモニタにおける多重タスク処理によれば、タスクは0から34までの実行時のプライオリティをつけることができる。システム・タスクは高いプライオリティを持ち、ユーザ・タスクは比較的低いプライオリティを持っている。原研システムでは、ユーザ・タスクの実行プライオリティは0から7のランクを持っている。CPUが空いたときは、高いプライオリティを持つタスクが次の処理の対象となり、残りが待たされる。CPUは2つしかないわけだから、実行可能なタスクが2つを越えるときは、どれかがCPU待ちの状態となる。

システム内で最もプライオリティの低いタスクとしてアイドル・タスクがある。CPUは仕事があってもなくても動いている。仕事があるときはアイドルタスク以外のタスクを実行している。仕事があっても入出力処理等の事象待ちで処理を先に進められないときは、アイドル・タスクを実行し時間つぶしをする。仕事のないときは、アイドル・タスクをくり返し実行している。

アイドル・タスクの1回あたりの実行時間は $320\mu\text{s}$ である。何回アイドル・タスクを実行したかの累積数がCPU別にメモリの特定番地にカウントされている。CPUは平等に使用されているので、両者ほぼ同じ値となっている。これらの番地はシステム初期設定によってクリアされる。

測定開始時刻(t_1)と終了時刻(t_2)におけるアイドル回数の差が測定時間内のアイドル回数である。それに $320\mu\text{s}$ を乗じるとアイドル時間(y)が得られる。

一方、ユーザ使用時間は、アカウント情報からジョブごとにCPU使用時間とジョブの開始時刻、終了時刻が得られるので、測定時間内に実行されたジョブを集計することによりわかる。しかしながら測定区間の両端にまたがるジョブに対しては、正確なCPU使用時間が求められない場合がある。ここでは、次に示す計算方式でユーザCPU使用時間を出すことにする。



ジョブ開始時刻を (J_1) , 終了時刻 (J_2) とすれば,

$$\text{ジョブ1の場合} \quad \frac{J_2 - t_1}{J_2 - J_1} \times \text{CPU使用時間},$$

$$\text{ジョブ2の場合} \quad \frac{t_2 - J_1}{J_2 - J_1} \times \text{CPU使用時間},$$

$$\text{ジョブ3の場合} \quad \frac{t_2 - t_1}{J_2 - J_1} \times \text{CPU使用時間}$$

を測定区間内の使用時間だと仮定する。

2つのCPUが平等に仕事を分担していることから, 1-CPUに対するユーザ使用時間 (x) は, 集計値を2で割ったものである。 x , y がわかるとこれらから次式を得る。

$$a = t_2 - t_1$$

$$Z = a - x - y$$

- ただし a ; 測定時間
 t_1 ; 測定開始時刻
 t_2 ; 測定終了時刻
 x ; ユーザCPU使用時間
 y ; アイドル時間
 Z ; システム使用時間

ユーザ使用時間の計算は, この方法では, No.1システムの場合は1%程度, No.2システムの場合は, 数%程度の誤差を含むことになる。したがって, システム使用時間も同程度の誤差を含んでいる。アカウント情報からジョブごとのユーザ使用時間を集計するしか方法がないので誤差があるのは止むをえない。測定時間が長い程誤差が少なくなる。

4. CPU アイドルの原因

4.1 機器の使用率とCPUアイドル

第2図，第3図，第4図は， $\mu 1$ システムに対して1973年6月14日の10時25分から11時40分までの135分間にわたって，約40秒間隔で，CPUアイドル，メモリ内の実行タスク数，メモリ使用率，C-チャンネル使用率を測定した結果を図示したものである。縦軸に使用率を%で示し，横軸に測定時刻を示している。アイドル率は上から下への目盛を見ればわかる。

第2図は，CPUアイドル状況と，実行タスクの多重度を示している。詳細はすでに，2で述べたとおりである。

第3図は，ユーザCPU使用率とメモリ使用率を示している。 $\mu 1$ システムでは，199K語がユーザに開放されている。平均のメモリ使用率は，58.5%（115K語）である。図で見るかぎりでは，これら両者の相関はあまり認められない。

第4図は，CPU使用率とC-チャンネル使用率について示している。平均のC-チャンネル使用率が72.3%となっている。C-チャンネルの使用率だけを取り上げたのは，既に述べたように， $\mu 1$ システムでは，他のチャンネルに比べて，C-チャンネルの使用率がきわめて高いので，CPUアイドルの原因の1つと考えられるからである。C-チャンネルは，ユーザ・プログラムよりは，むしろシステム・タスクで使用される場合が多いので，この図ではCPU使用率の中に，システムが使用する時間も含めている。CPU，C-チャンネル共に使用率の時間による変動は激しく，それらの間の相関は，一見ないといえる。

第2図，第3図，第4図からジョブの実行多重度，メモリ使用率，C-チャンネル使用率は，CPUアイドル率にそれ程影響しないことがわかる。相関係数表によってこれをたしかめる。測定点が121あるうち前後を省いて90点を選んで相関係数を出したのが第3表である。相関の有意水準表（ t -分布）から，

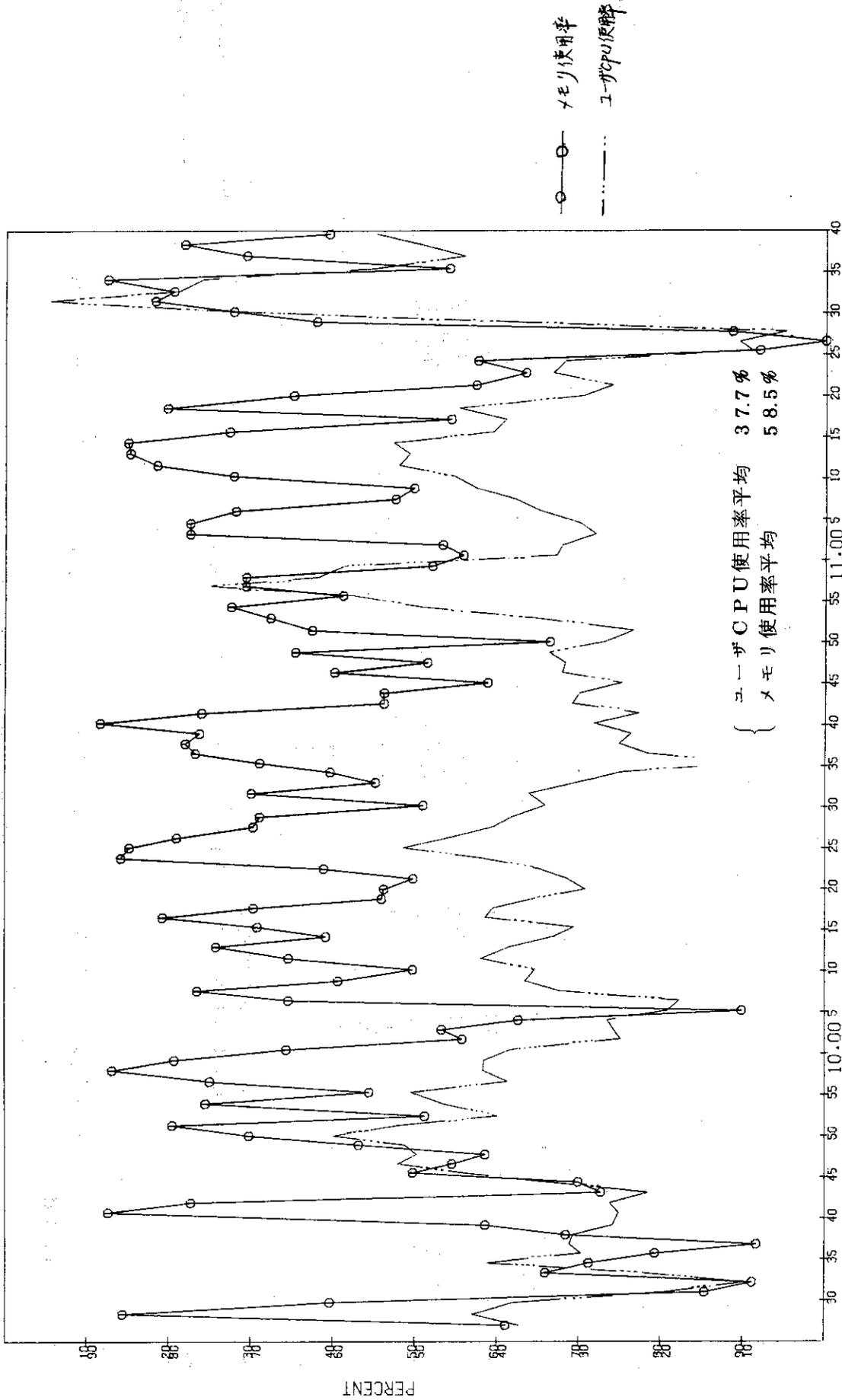
$$r(90, 0.01) = 0.269, \quad r(90, 0.05) = 0.206。$$

CPUアイドル率との相関が認められるためには，相関係数の絶対値が有意水準を上まわる必要がある。結局，第3表からC-チャンネル使用率との相関が認められる。重相関係数の2乗（ R^2 ）が0.38であることも相関のなさを示している。

CPUアイドル率が50%を越える場合が，121点中42点あり，第4表はその場合におけるジョブの実行多重度，メモリ使用率，C-チャンネル使用率との因果関係を示している。原因がこれら3者でない場合が11点もあることがわかる。第4表のA，B，Cは，第5，6，7，8図に示したそれぞれの累積度数図をもとにして設定した。CPUアイドル率とC-チャンネルの使用率は，80%の近辺を接点とし，メモリ内実行タスク数とメモリ使用率は，20%の近辺を接点として定めた。

以上の結果から装置の使用率とCPUアイドル率とは，C-チャンネル使用率を除いて顕著でないといえる。しかしながら，測定時間間隔が短かすぎて，C-チャンネルの使用が極限に達し，しばらくしてCPUのアイドル率が増加するといった時間のずれを考慮していないので，これ

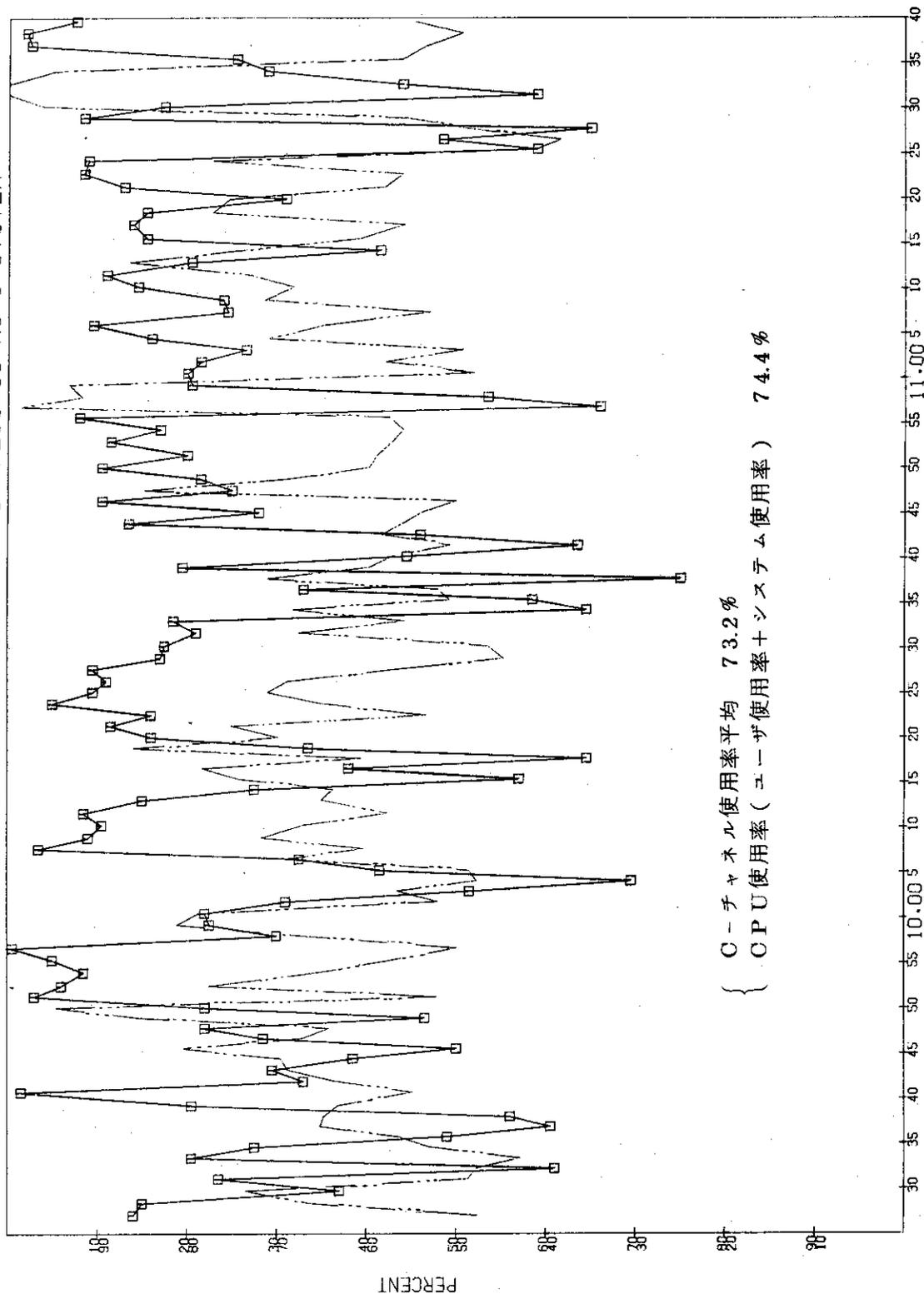
CPU AND CORE USED BY USER PROGRAM OF F230-60 NO.1 SYSTEM



第3図 コアメモリ使用率

1973年6月14日

CPU UTILIZATION AND C-CHANNEL UTILIZATION OF F230-60 NO 1 SYSTEM



1973年6月14日

第4図 C-チャネルの使用率

第3表 相関係数表1 (1973年6月14日)

	CPUアイドル率	コア内実行タスク数	コア使用率	C-チャンネル使用率
CPUアイドル率	1.00	-0.11	-0.12	0.29 **
コア内実行タスク数	-0.11	1.00	0.72 **	0.42 **
コア使用率	-0.12	0.72 **	1.00	0.21 **
C-チャンネル使用率	0.29 **	0.42 **	0.21 **	1.00

$R^2 = 0.381$

$r(90, 0.01) = 0.269$

$r(90, 0.05) = 0.206$

** 高度に有意

* 有意

第4表 CPUアイドルの原因(アイドル率が50%以上の場合)

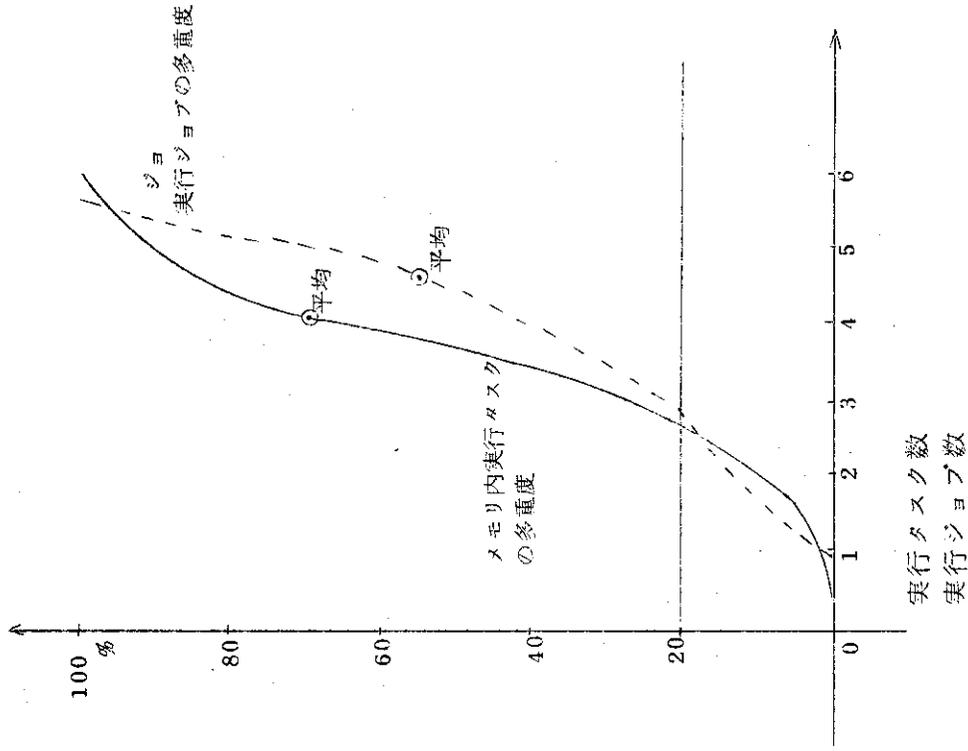
アイドル率50%以上の件数	42
A	18
B	16
C	12
A, B	14
B, C	1
C, A	1
A, B, C	1
いずれでもない。	11

A メモリ内実行タスク数が3以下

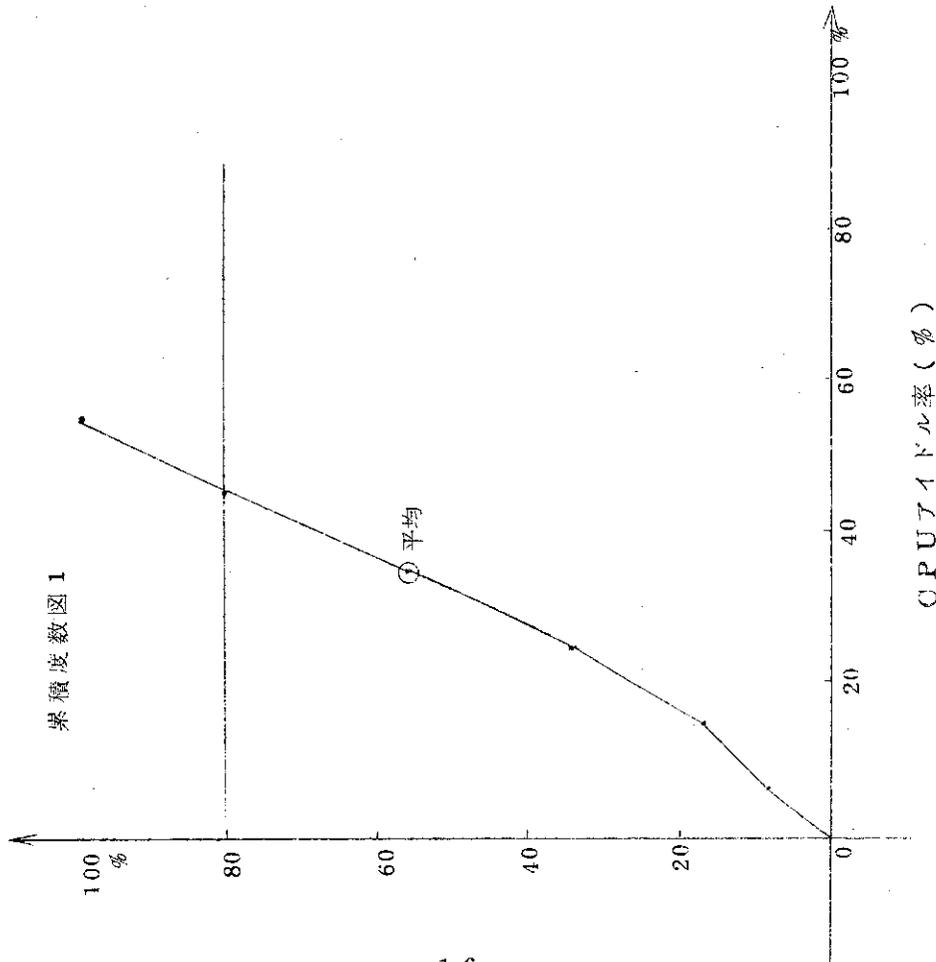
B メモリ使用率40%以下

C C-チャンネル使用率90%以上

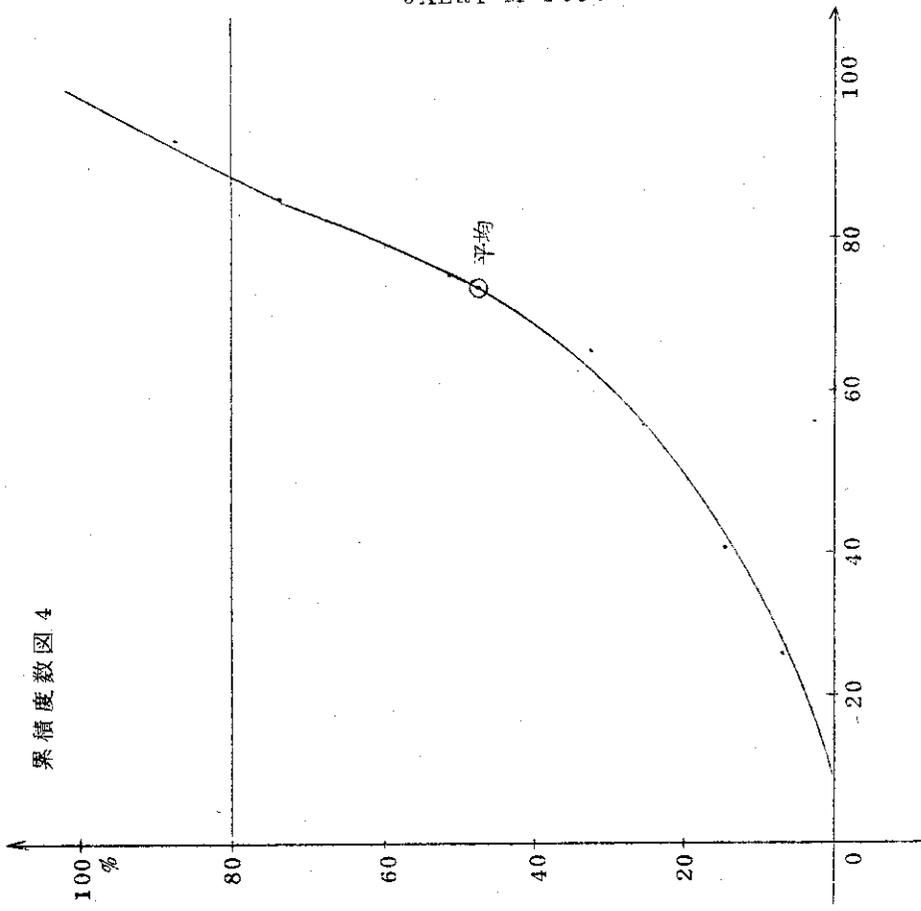
(1973年6月14日)



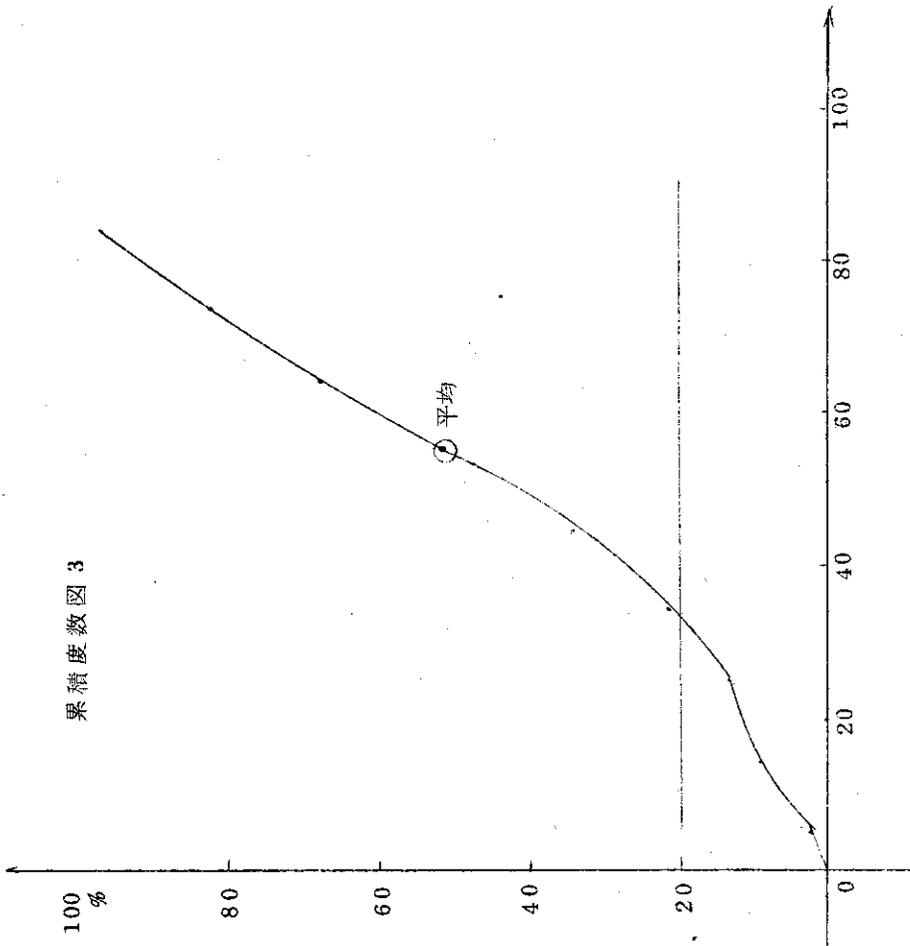
第6図 実行タスク数 (1973年6月14日)



第5図 CPUアイドル率 (1973年6月14日)



C-チャネル使用率



メモリ使用率

第 8 図 C-チャネル使用率 (1973年6月14日)

第 7 図 メモリ使用率 (1973年6月14日)

だけでは結論がでない。実際は、もう少し相関が深いはずであるが、瞬間的に見た場合、因果関係が顕著でなかったということである。

4.2 アカウンティング情報とCPU アイドル時間

アカウンティング情報としてF230-60 M-V モニタでは、ジョブごとの情報として、利用者氏名、ジョブ実行開始、終了時刻、CPU、メモリ使用量等が取られる他に、ジョブ・ステップ情報として、CPU、メモリ使用量、完了状態等が取られ、1ジョブあたり約70種の情報がディスクまたは磁気テープに取りられ保存されている。通常は、利用料金を算定したり、ジョブの統計を取るのに利用される。

CPUアイドルは実行するジョブ・パターンに大きく依存する。それを数字で示すことを試してみる。CPUアイドル(y)に影響を与える要因としてアカウンティング情報から取れるものに次のものがある。

- (1) 入力ジョブ数 (x_1)
- (2) 入力カード枚数 (x_2)
- (3) コントロール・カード枚数 (x_3)
- (4) ユーザCPU使用時間 (x_4)
- (5) ラインプリンタ出力行数
+カード・パンチ枚数 (x_5)
- (6) ジョブ・ステップ数 (x_6)
- (7) ファイル使用回数(順繰成ファイルのみ) (x_7)
- (8) メモリ使用容量(\sum CORE-TIME \times CORE使用容量) (x_8)
- (9) ターンアラウンド・タイム (x_9)

既に表示した第1表と第2表は、上記のデータを記録している。第8欄より右から順に上記の項目の(1)から(9)が示されている。

さらに、第5表と第6表は、項目(1)~(8)を測定時間で割って、単位時間(秒)あたりの値を示したものである。第5表と第6表を基礎データとして、 $\#1$ システム、 $\#2$ システムのCPUアイドルについての回帰分析を行っていく。

なお $\#2$ システムの測定は、昼間の時間帯で行なった。夜間を入れれば、もっとCPUアイドル率が上がることが予想される。

以上のように、多くの要因を持つ場合は、多変量をパラメータとした重回帰分析の手法が用いられる。その場合、変数が互いに独立であるという仮定に立っている。しかしながら、実際のデータでは、変数の従属性が除去されていない「汚いデータ」が常である。回帰分析を完行する前に、変数が互いにどのように関連し合っているかを知っておく必要がある。そのために相関係数表を作る。

第5表 CPUアイドルの要因についてのサンプル・データ
 №1 システム
 (アカウテンディング情報から)

DATA/INTERVAL		DATE	START	END	INTERVAL	USER	SOFT	IDLE	JOB5	INPUT	MACRO	OUTPUT	JOBSTEP	FILE	CORE	TURNAROUND
1	0625	1135	0	1531	0	0.427	0.219	0.334	0.014	3.967	0.115	15.901	0.040	7.947	106.1	617.7
2	0625	1531	0	1712	0	0.427	0.234	0.364	0.015	4.398	0.135	14.959	0.050	10.123	105.4	809.5
3	0626	919	3	12	014	0.336	0.275	0.389	0.017	4.681	0.158	19.295	0.050	12.875	117.3	770.2
4	0627	92324	1632	6	25712	0.00	0.262	0.373	0.014	6.055	0.131	17.437	0.049	12.356	111.9	634.8
5	0628	9	939	114719	9460	0.00	0.324	0.459	0.014	5.376	0.116	11.231	0.035	14.441	91.9	864.3
6	0628	114719	133228	6309	0.00	0.399	0.160	0.441	0.006	3.970	0.068	6.313	0.020	7.292	108.5	596.1
7	0628	133228	16	240	9012	0.00	0.400	0.262	0.338	5.171	0.152	18.621	0.049	13.441	123.1	1081.4
8	0629	947	9	171519	26890	0.00	0.443	0.230	0.377	4.709	0.135	11.895	0.041	8.942	804.4	1081.4
9	0702	102126	114828	5772	0.00	0.315	0.253	0.432	0.017	5.101	0.143	13.556	0.048	8.884	809.5	633.5
10	0702	114828	131930	5467	0.00	0.249	0.229	0.522	0.007	3.468	0.034	9.310	0.016	6.305	353.4	435.4
11	0702	131930	17	045	13275	0.00	0.260	0.511	0.012	5.027	0.102	8.344	0.029	5.276	453.4	408.2
12	0703	10	230	172637	26627	0.00	0.298	0.446	0.013	5.244	0.102	12.383	0.035	8.165	74.1	514.1
13	0706	941	3	164237	25294	0.00	0.351	0.245	0.404	5.203	0.128	13.151	0.036	9.691	101.1	345.1
14	0709	92419	113710	7971	0.00	0.374	0.253	0.373	0.013	6.218	0.148	14.221	0.043	9.691	78.0	439.1
15	0710	93648	162339	2441	0.00	0.449	0.216	0.335	0.013	4.136	0.121	13.326	0.037	12.355	102.0	418.6
16	0711	93648	114013	7613	0.00	0.561	0.205	0.234	0.014	6.515	0.130	16.195	0.038	9.241	129.9	840.3
17	0711	114013	131029	5416	0.00	0.526	0.233	0.301	0.009	2.992	0.203	16.916	0.044	8.673	1084.4	1660.9
18	0711	131029	162442	11635	0.00	0.478	0.215	0.307	0.009	4.889	0.143	11.689	0.036	9.896	1084.4	1660.9
19	0712	93916	121653	9457	0.00	0.400	0.235	0.365	0.015	5.624	0.148	13.510	0.041	9.308	90.1	1138.6
20	0712	121653	131335	3402	0.00	0.164	0.277	0.559	0.007	3.318	0.071	11.329	0.022	10.368	483.1	1420.8
21	0712	131335	1619	11134	0.00	0.412	0.226	0.362	0.015	4.678	0.134	12.663	0.040	11.470	44.4	372.3
22	0716	93054	12	533	5493	0.00	0.615	0.198	0.168	6.181	0.176	15.848	0.048	8.711	88.4	523.9
23	0716	93054	12	533	9274	0.00	0.404	0.251	0.345	6.058	0.126	14.539	0.041	12.769	124.8	929.5
24	0716	12	533	134534	6001	0.00	0.305	0.257	0.438	6.149	0.115	15.977	0.029	9.425	87.9	459.8
25	0716	134534	152730	6116	0.00	0.361	0.198	0.240	0.011	4.423	0.145	9.763	0.052	10.962	73.3	376.7
26	0717	94838	114326	6888	0.00	0.459	0.232	0.310	0.014	4.423	0.145	9.763	0.052	10.962	128.7	437.6
27	0717	14326	131010	5284	0.00	0.199	0.262	0.340	0.017	4.472	0.130	13.538	0.052	12.119	118.0	1261.3
28	0717	14326	131010	8781	0.00	0.199	0.262	0.340	0.008	2.426	0.080	8.561	0.026	7.013	158.0	792.5

第6表 CPUアイドルの要因についてのサンプル・データ
 №2 システム
 (アカウテンディング情報から)

DATA/INTERVAL		DATE	START	END	INTERVAL	USER	SOFT	IDLE	JOB5	INPUT	MACRO	OUTPUT	JOBSTEP	FILE	CORE	TURNAROUND
1	0717	91625	131045	14030	0.00	0.573	0.077	0.351	0.003	1.404	0.035	3.412	0.005	11.643	134.7	435.2
2	0717	131045	154623	4938	0.00	0.609	0.177	0.214	0.003	1.458	0.019	4.247	0.007	6.749	131.5	5358.2
3	0718	92110	134830	16040	0.00	0.451	0.175	0.374	0.004	1.680	0.063	6.172	0.009	5.990	100.5	2842.0
4	0718	134830	162042	9132	0.00	0.685	0.093	0.232	0.004	1.087	0.030	7.387	0.010	8.517	134.6	3479.6
5	0719	135030	172140	5470	0.00	0.669	0.060	0.271	0.005	1.629	0.050	11.330	0.012	8.517	133.7	2849.1
6	0720	92318	131920	14162	0.00	0.574	0.121	0.355	0.003	0.834	0.039	5.030	0.007	8.530	134.9	5184.8
7	0721	93924	119521	8157	0.00	0.536	0.108	0.354	0.004	1.094	0.052	6.719	0.013	13.264	134.9	4773.4
8	0725	1015	0	1642	0	0.540	0.106	0.354	0.004	0.938	0.042	5.809	0.009	7.601	112.8	4428.8
9	0727	934	0	1137	0	0.558	0.112	0.330	0.006	1.753	0.069	6.292	0.012	8.681	124.1	5962.5
10	0806	112712	13	749	6037	0.00	0.518	0.060	0.002	0.836	0.020	4.673	0.007	2.444	167.7	4865.2
11	0806	13	749	1443	1	0.503	0.125	0.371	0.006	1.569	0.087	8.534	0.021	13.565	16.7	5042.7
12	0806	1443	1	15814	4120	0.00	0.360	0.208	0.321	1.157	0.036	11.002	0.009	10.462	102.6	2278.4
13	0807	102026	162013	21287	0.00	0.666	0.087	0.247	0.003	1.795	0.034	6.485	0.007	10.352	138.4	807.2
14	0808	93717	131744	13227	0.00	0.451	0.090	0.260	0.003	1.458	0.038	7.415	0.009	17.006	143.1	5826.1
15	0808	131744	144039	4975	0.00	0.447	0.196	0.357	0.006	3.543	0.081	15.469	0.012	23.976	120.1	3205.6
16	0827	91254	113920	8786	0.00	0.383	0.133	0.483	0.004	1.338	0.053	4.583	0.016	9.910	4199.4	5715.7
17	0827	113920	134050	7490	0.00	0.296	0.165	0.359	0.002	0.462	0.032	6.209	0.007	7.985	182.0	4397.5
18	0827	134050	154647	7957	0.00	0.534	0.112	0.354	0.003	1.714	0.043	5.086	0.012	2.802	77.3	1694.3
19	0828	91125	114939	9134	0.00	0.568	0.184	0.281	0.004	1.950	0.057	4.093	0.012	7.802	113.1	1866.5
20	0829	92639	114951	8592	0.00	0.466	0.147	0.387	0.004	2.660	0.062	5.204	0.011	11.459	101.4	3406.5
21	0829	114951	144023	10232	0.00	0.575	0.105	0.322	0.003	1.560	0.035	6.398	0.009	9.306	101.4	4058.2
22	0830	134821	17	233	12852	0.00	0.474	0.111	0.341	2.160	0.060	10.095	0.016	16.375	123.2	2448.7
23	0831	93340	113615	12852	0.00	0.546	0.112	0.342	0.003	2.160	0.060	10.095	0.016	16.375	140.6	2419.5
24	0831	113615	142622	8899	0.00	0.546	0.112	0.342	0.003	2.160	0.060	10.095	0.016	16.375	140.6	2419.5
25	0831	142622	163243	8781	0.00	0.409	0.147	0.444	0.005	1.967	0.045	6.669	0.013	4.943	102.9	1226.8
26	0831	142622	163243	8781	0.00	0.409	0.147	0.444	0.005	1.837	0.045	7.295	0.011	5.817	102.9	4607.6

4.2.1 №1システムのCPUアイドル

第7表は、第5表をサンプルデータとした場合の各変数間の相関関係を示す。

CPUアイドルとの相関は、ファイル使用回数を除いて負の相関を持ち、いづれも高度に有意である。ファイル使用回数に対して相関が低い理由は、現在のところ順編成ファイルに対する使用回数のみしか使用回数としてカウントされないため、主としてシステムが使用する他の編成のファイル使用回数が無視され、現実を反映していないためかと思われる。

№1システムでは、CPUアイドルは、ユーザのCPU使用時間の他に、入出力量、ジョブ・ステップ数などにも影響される。ジョブ・ステップに要するCPU時間が比較的短いので(第16表)、ジョブ・ステップ数が増えれば、ユーザCPU使用時間も増えるが、ジョブ・ステップの起動/停止処理のようなシステム・オーバーヘッドも多くなり、結局、CPU使用率が上昇し、アイドルが減少する。このことは、第9図からわかる。

第10図は、№1システムにおけるCPUアイドルとターンアラウンド・タイムの関係を図示している。CPUアイドル率が高いということは、システムが混雑していないことになり、ターンアラウンドが短くなると解釈できる。

4.2.2 №2システムのCPUアイドル

第8表は、同じく№2システムの相関係数表である。№2システムは、クローズ・バッチ処理で、ジョブの多重度は4を基準とし、オペレータにより、その時の事情により変更される。深夜から朝にかけては、長時間でメモリ使用量の大きいジョブが集中するので多重度は、2か3を採用する場合が多い。実際のジョブの多重度、つまり実行多重度の平均値は第14表よりわかる。

№2システムは、クローズ・バッチ処理を行っているのにCPUのアイドル率が高いのは、大いに問題がある。アイドル理由は、既に1.で述べられているように、CPUの速さに比してメモリや周辺装置の不足が影響している。相関表(第8表)によると、CPUアイドル率と相関が強いのは、ユーザCPU使用時間と、メモリ使用率である。メモリ使用率とユーザCPU使用率はもともと相関が強いことから、№2システムでは、CPUアイドルは、ユーザCPU使用率で決ってしまう。

№1システムで有意であった入出力量との相関は、№2システムでは有意でない。入出力に要するCPU時間は№2システムでは無視できる。

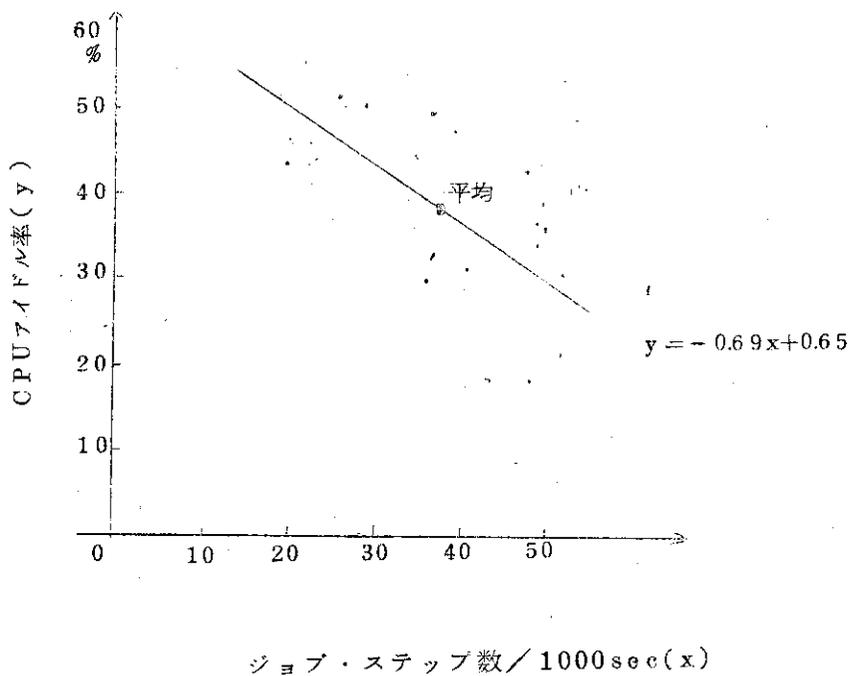
ジョブ・ステップ数との間に相関がないのは、ジョブ・ステップに要する時間の95%がユーザ・プログラムの実行に対するものであるから(第14表)、一定時間内に実行されるジョブ・ステップ数が多いということは、比較的短時間のジョブ・ステップ、たとえば、FORTRAN, LIED, LIBE 等が多いことになり、ユーザ・プログラムを実行する場合の比率が下がり、ユーザCPU使用時間が減る傾向となる。

第7表 相 関 係 数 表 2

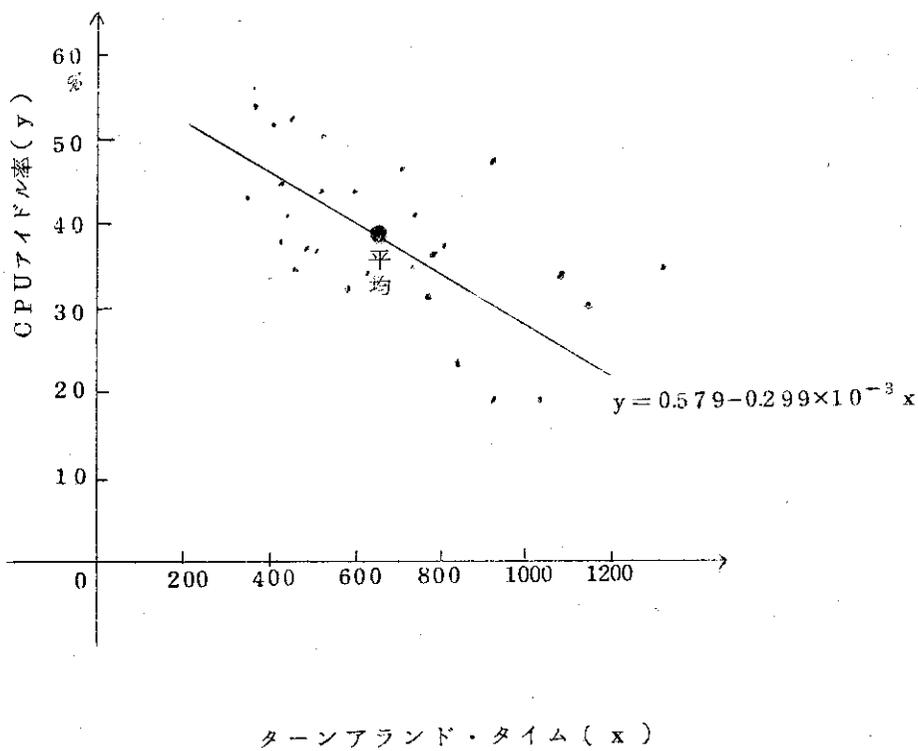
(1973年6-7月測定, 機1システム)

項 目	入力ジョブ数	入力カード枚数	コントロールカード枚数	ユーザCPU使用率	プリンタ出力行数 (+出力カード枚数)	実行ジョブ・ステップ数	ファイル使用回数	メモリ使用容量	ターンアラウンド・タイム	CPUアイドル率
入力ジョブ数	1.0 **	0.62 **	0.96 **	0.68 **	0.70 **	0.89 **	0.38 *	0.36	0.48 **	-0.79 **
入力カード枚数	0.62 **	1.0	0.64 **	0.50 **	0.60 **	0.41 *	0.22	-0.00	0.19	-0.57 **
コントロールカードの枚数	0.96 **	0.64 **	1.0	0.72 **	0.68 **	0.81 **	0.35	0.38 *	0.54 **	-0.81 **
ユーザCPU使用率	0.68 **	0.50 **	0.72 **	1.0	0.39 *	0.58 **	0.16	0.49 **	0.77 **	-0.98 **
プリンタ出力行数	0.70 **	0.60 **	0.68 **	0.39 *	1.0	0.64 **	0.44 *	0.15	0.26	-0.53 **
実行ジョブ・ステップ数	0.89 **	0.41 *	0.81 **	0.58 **	0.64 **	1.0	0.53 **	0.54 **	0.43 *	-0.68 **
ファイル使用回数	0.38 *	0.22	0.35 *	0.16	0.44 *	0.53 **	1.0	0.34	0.29	-0.23
メモリ使用容量	0.36	-0.00*	0.38 *	0.49 **	0.18	0.54 **	0.34	1.0	0.45 *	-0.47 **
ターンアラウンド・タイム	0.48 **	0.19	0.54 **	0.77 **	0.26	0.43 *	0.29	0.45 *	1.0	-0.74 **
CPUアイドル率	-0.79 **	-0.57 **	-0.81 **	-0.98 **	-0.53 **	-0.68 **	-0.23	-0.47 **	-0.74 **	1.0

$r(28, 0.01) = 0.46$, $r(28, 0.05) = 0.36$



第9図 アイドル率とジョブ・ステップ数との関係
(1973年6,7月 機1システム)



第10図 CPUアイドル率とターンアラウンド・タイムの関係
(1973年6,7月 機1システム)

第8表 相関係数表 3

(1973年7-8月測定; 系2システム)

項目	入力ジョブ数	入力カード枚数	コントロールカード枚数	ユーザCPU使用率	プリンタ出力行数	実行ジョブ・ステップ数	ファイル使用回数	メモリ容量	CPUアイドル率
入力ジョブ数	1.0	0.54**	0.86**	-0.05	0.54**	0.69**	0.40*	0.18	-0.02
入力カード枚数	0.54**	1.0	0.61**	-0.01	0.45*	0.35	0.59**	0.24	-0.14
コントロールカード枚数	0.86**	0.61**	1.0	-0.23	0.49**	0.78**	0.52**	0.08	0.15
ユーザCPU使用率	-0.05	-0.01	-0.23	1.0	-0.11	-0.20	-0.07	0.58**	-0.93**
プリンタ出力行数 (+出力カード枚数)	0.54**	0.45*	0.49**	-0.11	1.0	0.40*	0.64**	0.20	-0.02
実行ジョブ・ステップ数	0.69**	0.35	0.78**	-0.20	0.40*	1.0	0.40*	0.03	0.19
ファイル使用回数	0.40*	0.59**	0.52**	-0.07	0.64**	0.40*	1.0	0.37	-0.03
メモリ使用容量	0.18	0.24	0.08	0.58**	0.20	0.03	0.37	1.0	-0.62**
CPUアイドル率	-0.02	-0.14	0.15	-0.93**	-0.02	0.19	-0.03	-0.62**	1.0

 $r(25, 0.01) = 0.49$ $r(25, 0.05) = 0.38$

5. CPU アイドルの回帰分析

5.1 分散分析表

第9表, 第10表はNo.1システムに対する分散分析表で, 第11表はNo.2システムに対するものである。分散分析表は, 推定回帰式の精密度を表わすのによく用いられる。

サンプル・データを (X, Y) とし, 推定回帰式 $Y = b_0 + b \cdot X$ を得たとする。ここで, X, Y, \hat{Y}, \bar{y} は n 組のデータに対して,

$$X = (x_1, x_2, \dots, x_n), \quad Y = (y_1, y_2, \dots, y_n),$$

$$\hat{Y} = (\hat{y}_1, \hat{y}_2, \dots, \hat{y}_n), \quad \bar{y} = \frac{1}{n} \sum_{i=1}^n y_i \quad \text{とおけば,}$$

次の関係が成立する。

$$\sum_{i=1}^n (y_i - \bar{y})^2 = \sum_{i=1}^n (y_i - \hat{y}_i)^2 + \sum_{i=1}^n (\hat{y}_i - \bar{y})^2 .$$

この式は, 次のように解釈される。

(平均よりの残差平方和) = (回帰よりの残差平方和) + (回帰による残差平方和)。

分散分析表では, 上から順に, 回帰による残差平方和 (単に回帰と略す), 回帰よりの残差平方和 (残差と略す), 平均よりの残差平方和 (TOTALと示す) について記入される。第2列目は自由度, 第3列目は残差平方和, 第4列目は, 残差平方和を自由度で割ったもの: 不偏分散を示す。F値は, 回帰による不偏分散と残差不偏分散の比である。分散分析表の下の半分は, 追加部分で, 各変数の偏F値を示している。

回帰式の各係数は, 多変量模型に対しては, 相関係数表から計算される。ここでは, 変数を見易くするために, 適当な定数を乗じて変数の単位の変更を行っている。

回帰式が模型に適合しているかどうかを測る尺度として, F値と R^2 値がある。 R^2 値は変数の数を減らすと減少する。

F-検定は, 実測F値: (回帰による不偏分散) / (残差不偏分散) が, F-分布の対応する百分率点の値を越えているかどうかにより, 回帰式が有意であるかどうかを判定することである。

$$R^2 = \frac{\sum_{i=1}^n (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^n (y_i - \bar{y})^2} ,$$

R^2 ; 重相関係数の2乗は, 回帰による残差平方和と平均よりの残差平方和の比によって示され, どの程度回帰式がデータの変動を表現できているかをパーセントで示したものである。

偏F-検定は, 模型に変数を追加したり, 取除いたりすることに対する有効な判定基準を与えるものである。変数を追加した場合としない場合の不偏分散比が, F-分布の対応する百分率点の値を越えているかどうかで判定する。

5.2 推定回帰式

第9表は, 全変数をパラメータとした場合の分散分析表と回帰分析の結果を示している。変

数間に従属関係があるので、偏F値が小さい値を持つものが多い。第10表は、第9表において、偏F値の小さい変数を除去し、残る4変数に対し、回帰分析を試みた結果について示している。R²値の減少は微々たるもので、逆に偏F値が増加することから、第10表の式の方が推定回帰式として優れていることがわかる。

2式を較べてみると、ジョブ・ステップ数に対応する係数に変化が見られる。変数の除去における影響が出ていると考えられる。変数の組合せをいろいろ換えて計算してみたところ、ユーザCPU使用時間のみを変数とした場合に第12表で見られるように、F値が第10表の場合よりやや大きくなる。いずれの場合もR²値の減少を伴う。結局、第10表の4変数をパラメータとした回帰式か、ユーザCPU使用時間のみをパラメータとした式のいずれかが良い推定式となる。

第11表は、全変数をパラメータとして μ 2システムに対して回帰分析を行った結果を示す。 μ 1システムの場合に比して係数が少し異なることがわかる。これは偏F値が示すように、ユーザCPU使用時間以外は、変数が有意でないと考えられるからである。F値、R²値共に μ 1システムの場合より小さく近似が悪いことがわかる。

5.3 CPUアイドル率とCPU使用率の関係

第12表は、ユーザCPU使用時間のみを変数として回帰式を出したものである。定数項は、両システムがほぼ同じ値となるが、勾配は、 μ 1システムの方が、やや大きい。これは、式

$$\begin{aligned} \text{CPU稼働時間} &= \text{ユーザCPU使用時間} \\ &+ \text{システム使用時間} \\ &+ \text{アイドル} \end{aligned}$$

より、 μ 1システムは、システム使用時間が大きいのでそれが式に現われているといえる。回帰式を、測定区間の範囲を越えて外挿するのは意味のないことである。ユーザのジョブが全くない状態における(アイドル状態)CPUのアイドル時間は、約90%であるという実測結果を得ていることから、ユーザCPU使用時間がゼロの場合には、第12表に示す式のとおりではないことがわかる。定数項が、両システムが近い値をとることは、興味ある結果である。

第11図は、CPUアイドルとユーザ使用時間、さらにシステム使用時間の関係を図示したものである。ユーザ使用率を横軸に、CPUアイドル率を縦軸にとっている。全体を1としたときのユーザ使用率、システム使用率、アイドル率の比が、上から順に示される。ユーザCPU使用率を与えれば、それに対応するアイドル率、システム使用率を知ることができる。

ユーザ使用率が高くなるにしたがって、システム使用率は減る傾向にある。これは測定区間内に対する解析の結果に過ぎないので、ユーザ使用率がゼロに近い場合には、この限りではないことをことわっておく。

μ 1システムと μ 2システムとの勾配の差が、短時間ジョブを処理する μ 1システムのシステム・オーバヘッドの増加分であると解釈できる。

第9表 分散分析表1 (CPUアイドル)

No.1 システム

変動要因	自由度	平方和	不偏分散	F値, 偏F値
回帰 (回帰による残差)	9	0.2627	0.0292	
残差 (回帰よりの残差)	18	0.0024	0.0001	219.5**
TOTAL	27	0.2651		
x_1 (入力ジョブ数/秒)				0.094
x_2 (入力カード数/秒) $\times 10^{-2}$				0.099
x_3 (コントロールカード枚数/秒)				10.144**
x_4 (ユーザCPU使用率)				258.100**
x_5 (プリンタ出力行数/秒) $\times 10^{-2}$				7.550**
x_6 (実行ジョブステップ数/秒)				2.092
x_7 (ファイル使用回数/秒) $\times 10^{-3}$				0.250
x_8 (コア使用量(K)/秒)				1.100
x_9 (ターンアラウンドタイム(秒) $\times 10^{-5}$)				0.040

$$R^2 = 0.991$$

$$F_{0.01}(18, 9) = 3.6$$

回帰式 (y; CPUアイドル)

$$y = 0.75 + 1.0x_1 - 0.11x_2 - 0.29x_3 - 0.73x_4 - 0.29x_5 - 1.08x_6 \\ + 0.68x_7 + 0.13x_8 - 0.34x_9$$

第10表 分散分析表2 (CPUアイドル)

No.1 システム

変動要因	自由度	平方和	不偏分散	F値, 偏F値
回帰	4	0.2625	0.0656	
残差	23	0.0026	0.0001	572.1**
TOTAL	27	0.2651		
x_3 (コントロールカード枚数/秒)				3.926
x_4 (ユーザCPU使用率)				759.189**
x_5 (プリンタ出力行数/秒) $\times 10^{-2}$				12.95**
x_6 (ジョブステップ数/秒)				2.065

$$R^2 = 0.990$$

$$回帰式 \quad y = 0.76 - 0.28x_3 - 0.73x_4 - 0.31x_5 - 0.53x_6$$

第11表 分散分析表3 (CPUアイドル)

#2システム

変動要因	自出度	平方和	不偏分散	F値, 偏F値
回帰	8	0.1373	0.0172	20.33**
残差	16	0.0135	0.0008	
TOTAL	24	0.1508		
x_1 (入力ジョブ数/秒)				0.001
x_2 (入力カード数/秒) 10^{-2}				1.398
x_3 (コントロールカード枚数/秒)				0.008
x_4 (ユーザCPU使用率)				74.039**
x_5 (プリンタ出力行数/秒) $\times 10^{-2}$				0.551
x_6 (実行ジョブステップ数/秒)				0.637
x_7 (ファイル使用回数/秒) $\times 10^{-3}$				0.051
x_8 (コア使用量(K)/秒)				0.219

$$R^2 = 0.91$$

$$F_{0.01}(16, 8) = 3.9$$

回帰式 (y ; CPUアイドル)

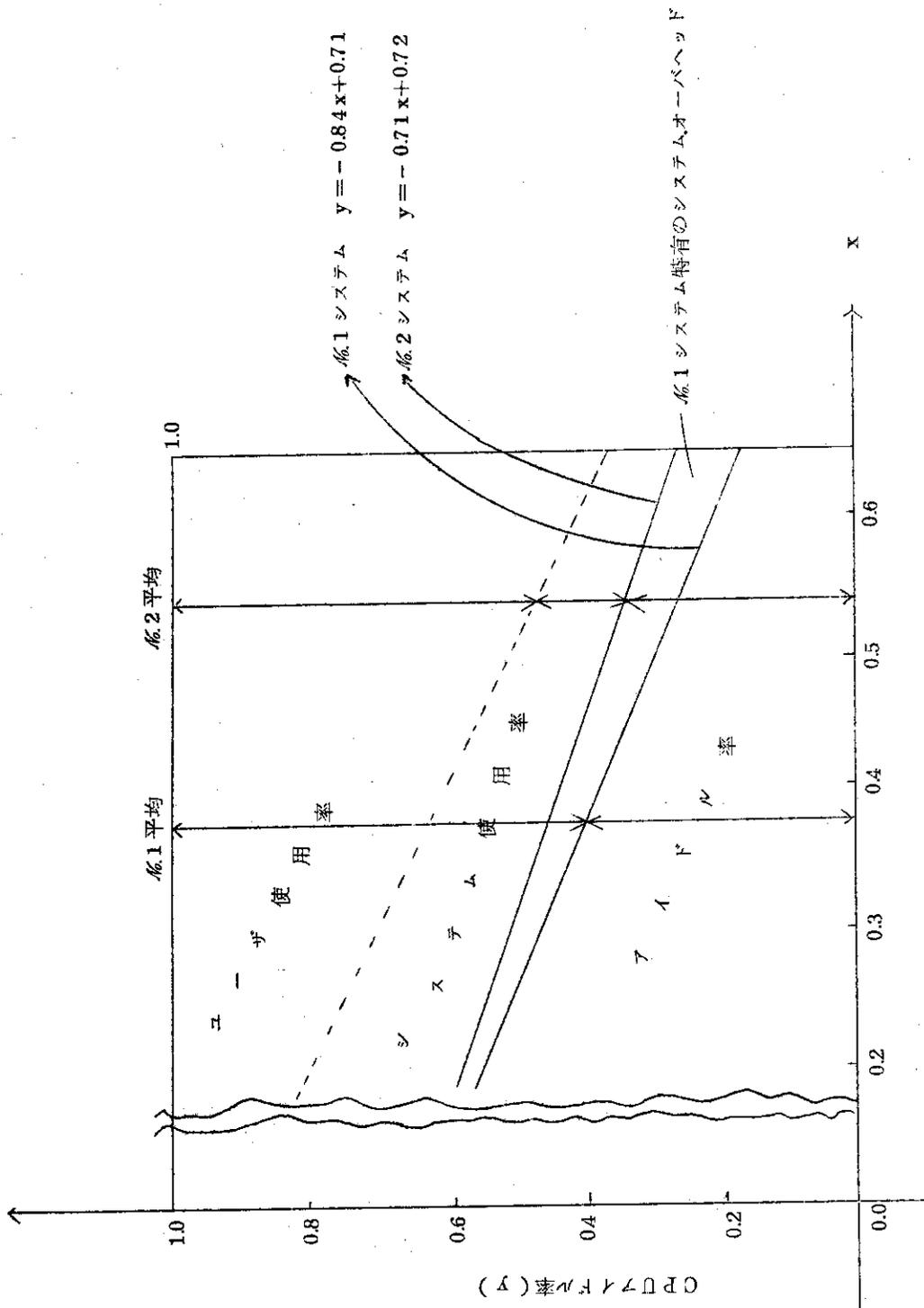
$$y = 0.75 - 1.01x_1 - 0.01x_2 - 0.02x_3 - 0.69x_4 - 0.23x_5 + 2.35x_6 \\ + 0.47x_7 - 0.17x_8$$

第12表 CPUアイドル率とユーザCPU使用率

	回帰式	F	R ²
#1システム	$y = -0.84x + 0.71$	596.7	0.958
#2システム	$y = -0.71x + 0.72$	165.2	0.878

y ; CPUアイドル率

x ; ユーザCPU使用率



第11図 CPUアアイドルの分析

6. システム・オーバヘッド

CPUの全稼働時間から、アイドル時間とユーザ使用時間をひいた残りをシステムの使用時間とし、システム・オーバヘッドと考える。全稼働時間を1とした場合に、ユーザCPU使用時間(x)、アイドル時間(y)、システム・オーバヘッド(z)とすれば、

$$x + y + z = 1 \quad (1)$$

を得る。

アイドル時間に対する推定回帰式が

$$y = a_1 x_1 + a_2 x_2 + \dots + a_9 x_9 + a_0 \quad (2)$$

で与えられたとする。ただし

$$x_4 = x \quad (\text{ユーザCPU使用時間}) \quad (3) \quad \text{とする。}$$

(1), (2), (3)から

$$\begin{aligned} Z &= 1 - y - x \\ &= -a_1 x_1 - a_2 x_2 - a_3 x_3 - (1+a_4)x - a_5 x_5 - a_6 x_6 \\ &\quad - a_7 x_7 - a_8 x_8 - a_9 x_9 + (1-a_0) \end{aligned} \quad (4)$$

№1システムの場合の回帰式の結果を代入すると、

第9表から、

$$\begin{aligned} Z &= 0.25 - 1.0 x_1 + 0.11 x_2 + 0.29 x_3 - 0.27 x + 0.29 x_5 + 1.08 x_6 \\ &\quad - 0.68 x_7 - 0.13 x_8 + 0.34 x_9 \end{aligned} \quad (5)$$

となる。

変数を少なくした場合は、第10表から、

$$Z = 0.24 + 0.28 x_3 - 0.27 x + 0.31 x_5 + 0.53 x_6 \quad (6)$$

ユーザCPU使用時間のみを変数とした場合には、

$$\text{№1システム} \quad Z = -0.16 x + 0.29 \quad (7)$$

$$\text{№2システム} \quad Z = -0.29 x + 0.28 \quad (8)$$

となる。(7), (8)式を、ユーザCPU使用時間を横軸にとって図示すれば、第12図が得られる。既に記述したように、システム・オーバヘッドは、平均的にみて、№1システムでは23~25%、№2システムでは11~15%である。

CPUアイドル率(y)を推定する回帰式から、システム・オーバヘッド率(z)を(4)式のように推定してもよいことを次に示そう。

サンプル・データを次のようにきめる。

n 組のデータに対して、

$$\mathbf{x} = (x_1^i, x_2^i, \dots, x_n^i) \quad , \quad \mathbf{x} = \mathbf{x}^i$$

$$\mathbf{y} = (y_1, y_2, \dots, y_n)$$

$$\mathbf{z} = (z_1, z_2, \dots, z_n)$$

さらに、変動要因 $\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^n$, $\mathbf{x}^i = (x_1^i, x_2^i, \dots, x_n^i)$

より、マトリックスXを次のように示す。

$$\mathbf{X} = \begin{pmatrix} 1 & x_1^1 & \cdots & x_1^g \\ 1 & x_2^1 & & x_2^g \\ \vdots & \vdots & & \vdots \\ 1 & x_n^1 & & x_n^g \end{pmatrix}$$

$y = \mathbf{X} \cdot \alpha + \epsilon$ なる関係が成立し、 α の推定値として a を仮定する。よく知られているように (5) ,

$$a = (\mathbf{X}' \cdot \mathbf{X})^{-1} \cdot \mathbf{X}' \cdot y \quad (9)$$

が成立する。同じく

$z = \mathbf{X} \cdot \beta + \epsilon$ なる関係が成立し、 β の推定値として b を仮定する。ここで ϵ は誤差を示す。

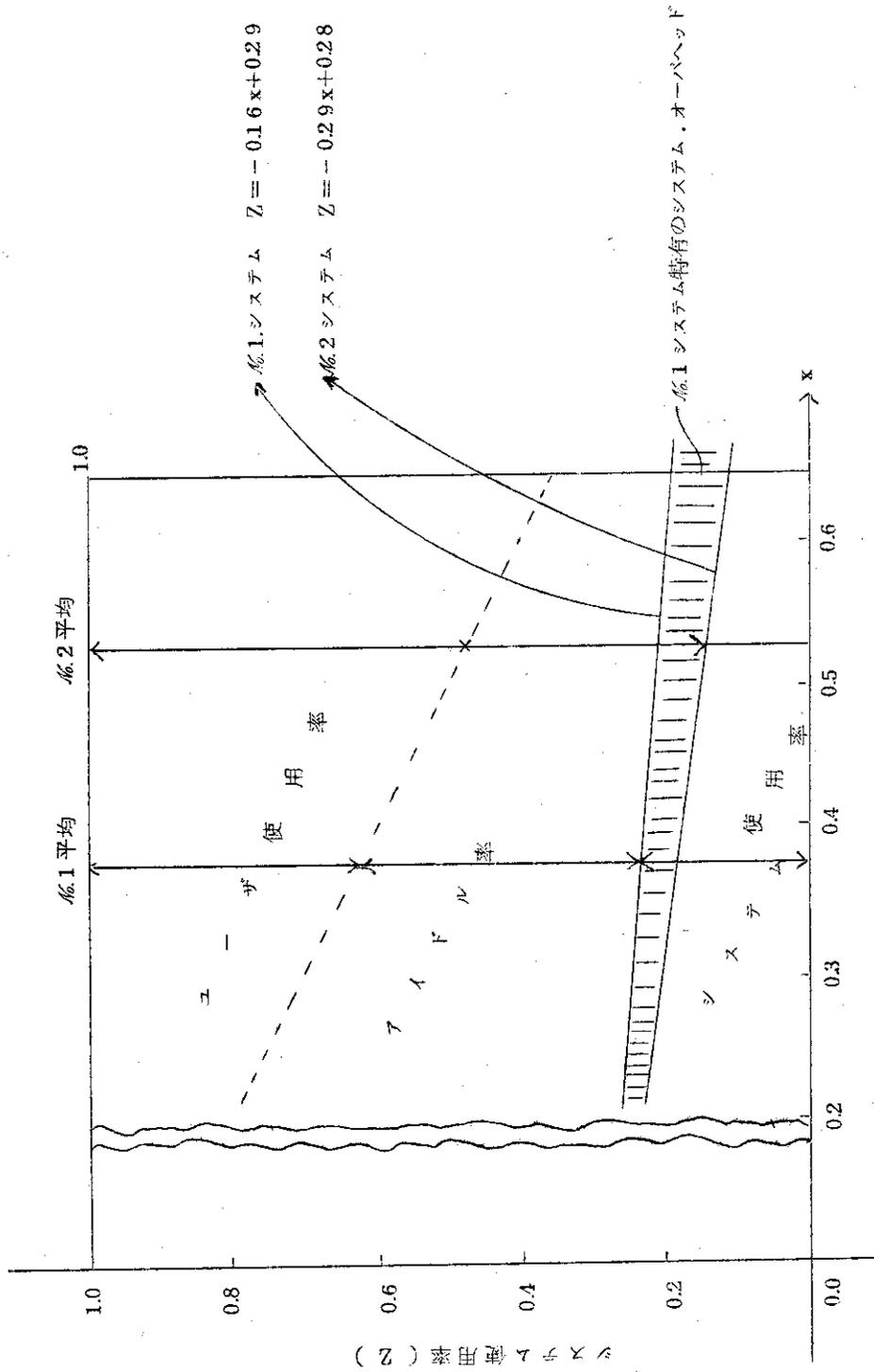
$$b = (\mathbf{X}' \cdot \mathbf{X})^{-1} \cdot \mathbf{X}' \cdot z \quad (10)$$

$$\begin{aligned} z &= \mathbf{X} \cdot b \\ &= \mathbf{X} \cdot (\mathbf{X}' \cdot \mathbf{X})^{-1} \cdot \mathbf{X}' \cdot (1 - y - x) \\ &= 1 - \mathbf{X} \cdot a - x \end{aligned} \quad (11)$$

z の推定式としては、

$$z = 1 - (1, x_1, x_2, \dots, x_g) \cdot a - x \quad (12)$$

となり(4)式と同じ結果を得る。



第12図 システム・オーバーヘッドの分析

7. 入出力等の待時間

原研システムではCPUアイドルが比較的多いことは、既に述べたとおりである。さて、CPUアイドル時間のうち、どの程度が入出力処理等の待時間として費されているのであろうか。CPUアイドルのように、実測する手段が今のところないので、他のデータを利用して判断するしかない。

CPUと入出力機器は、データ・チャネルを通して、非同期で働いているので、CPUがアイドル状態でも、入出力処理が実行されている場合があり、システムとしてアイドル状態とは言えない場合がある。しかしながら、このような場合においてもCPUアイドルを少なくすることが多重タスク処理をする上で重要な項目である。一方、入出力バウンドのジョブが重なり合ったり、同じ入出力機器への使用が重なったりすることは、現実には避けられないことであり、入出力作業待ちのためにCPUアイドル状態をゼロにすることは不可能である。

CPUアイドル時間のうちどの程度が入出力処理や、メモリ待ちに起因するかを定式化するために、次の仮定をおく。

「CPUアイドル時間のうち、入出力処理等の待時間は、CPU使用率(ユーザ+システム)に比例する。」

この仮定により、入出力等の待時間(w)は、次式のように表わせる。

$$w = b \cdot y \cdot \frac{a - y}{a} \quad (1)$$

$$a = 0.71$$

0.71は第12表でユーザCPU使用率が理論上ゼロになったときの値で、この場合の入出力等の待時間をゼロとするために、きめたものである。bは、入出力等の待ちに対する負荷を示す。この値をどのように定めるかについての根拠は、現在のところはっきりしたものがないので、 $b=1$ としておく。

$$(1)式より, \quad w = -\frac{1}{0.71} \left(y - \frac{0.71}{2} \right)^2 + \frac{0.71}{4},$$

系1システムでは、 $y = -0.84x + 0.71$ だから、これを代入すると、

$$w = -0.99(x - 0.42)^2 + 0.18 \quad (2)$$

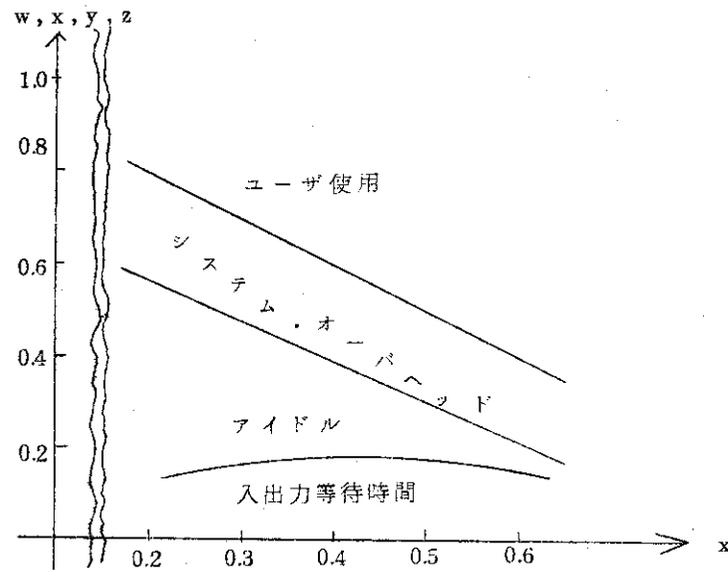
同じく系2システムでは、

$$w = -0.71(x - 0.5)^2 + 0.18 \quad (3)$$

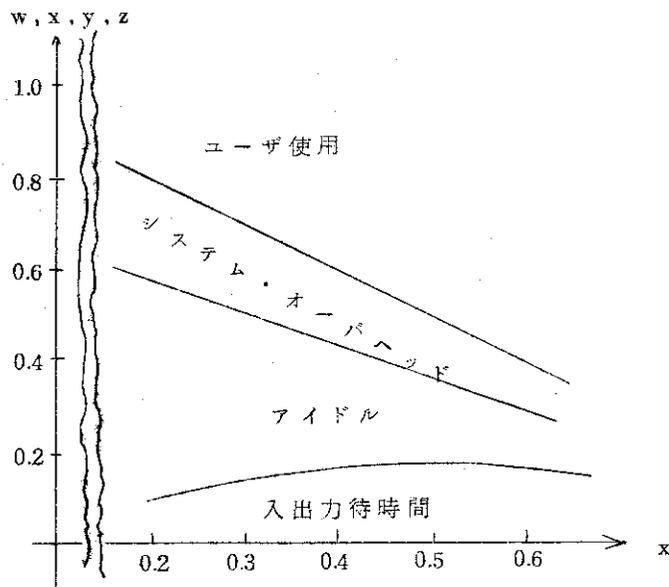
となる。

(1)、(2)式を図示したのが第13、14図である。結局系1システムは、平均的に見て

ユーザ使用率	38.5%
システム・オーバーヘッド	23.5%
入出力等の待時間	17.7%



第13図 入出力待ち時間の推定
(No.1システム)



第14図 入出力待ち時間の推定
(No.2システム)

アイドル時間	20.3%
同じく #2 システムでは、	
ユーザ使用率	53.3%
システム・オーバヘッド	12.5%
入出力等の待時間	17.6%
アイドル時間	16.6%

入出力処理等の待時間を正確に測定する試みには、今のところまだ、どこでもなされていないことがらである。この推定が正しいかどうかは、大がかりなハードウェア測定と収集データの解析を伴うことになり、将来にゆずることとする。

8. №1システムと№2システムの比較

№1システムと№2システムの運用の違いは、既に述べられている。原研独自の基準を設けて、プライオリティが設定されている。№1システムは、プライオリティの高いジョブ、№2システムでは、低いジョブと、自然にそれぞれが処理するジョブのパターンが定まり、計算センターが導入当初意図した方向でシステムの使い分けがなされている。ほぼ同じハードウェア構成を持つシステムが、運用の違いでどれ程異なったジョブ・パターンを処理することになるかを明確に示すために、№1システムと№2システムの比較を行なう。

8.1 多重度、CPU使用時間、メモリ使用容量等の比較

第13表は、№1システムにおけるジョブの平均多重度、集計時間内の処理件数、平均CPU使用時間、平均メモリ使用容量、平均出力量を示している。第14表は同じく№2システムに関するものである。

№1システムは、5多重で実行されているが、ジョブが必ずしも充足せず、平均すれば4以下となっている。時に昼食時は、入力ジョブが少なくなる。1日の処理件数の平均は350程度である。第13表、14表の集計では、システム・ダウンその他、ジョブの入力が妨げられる時間を取除いている。№1システムでは、平均CPU使用時間が1分強、平均メモリ使用容量は30K程度、出力量はラインプリンタで1000~1500行(約25ページ)となっている。

№2システムでは、長時間ジョブが処理されるが、午前中は比較的短いもの、午後は中程度のもの、夜間は超長時間ジョブが実行されるような運用がなされている。オペレータによりジョブの入力が調整され、午前中に長時間ジョブを入力しないことにもよるが、原研のプライオリティのつけ方から、長時間ジョブ程、後に残ることになる。多重度、平均CPU使用時間、メモリ使用容量は、午前、午後、夜間とで異なった値となっている(第14表)。多重度は、原則として4で、ロールアウトされているジョブが多いときは、下げたり、短時間ジョブが入力され何らかの理由で処理を危ぐ時は上げたりオペレータにより指示される。実際の多重度は4以下となっていることから、夜間の処理能力に余力があると判断される。平均CPU使用時間とメモリ使用容量は、夜間、特に早朝に大きくなる。出力量は、№1システムより多い。

CPU使用時間に限った比較が第15表でなされている。平均CPU使用時間は、№2システムは№1システムに較べて約6.5倍となっている。№1システムでは約80%が1分以内のジョブである。№2システムにおいても、エラーで早く終るジョブもあるので、短いジョブも多いが10分を越えるものが20%程度あり、これらのジョブによって使用されるCPU時間が大きい。

8.2 ジョブ・ステップの比較

各システムでどのようなジョブ・ステップが重点的に実行されているかを調べて表にしたのが、第16表である。

CPU使用時間別に見ると、 $\#2$ システムの方が、圧倒的にユーザプログラムの実行比率が高い。ジョブ・ステップ比(数)では、 $\#1$ システムの方がFORTRANの比率が高くなっている。時間比では、FORTRAN, LIED, LIBEの比率が $\#1$ システムでは30%もあり、 $\#2$ システムでは5%未満なのに比べると大きい。 $\#1$ システムがディバック用に使用されているのがわかる。

8.3 ロールアウト・時間の比較

ジョブの実行開始から終了までを、ジョブの実行時間とすると、そのうち何パーセントかが、ロールアウトされ大記憶(ドラムまたはディスク)に掃出された状態となる。メモリ内に滞在中のジョブでも、多重ジョブ処理のシステムだから、同時にいくつかのジョブが開設されているので、CPU待ちとなることもある。入出力等の待ちにより、そのジョブ自身の理由により、CPUのサービスを延期されていることもある。第17表から、 $\#1$ システムの平均は、ロールアウト率が19%、CPU使用待ち時間+入出力待ち時間が54.5%、残る26.5%がCPU使用時間となっている。 $\#2$ システムでは、ロールアウト率が29.3%とやや大きく、CPU使用待ち時間+入出力待ち時間が40.4%、CPU使用時間が30.3%となっている。ロールアウト率が大きく、CPU使用率がやや高い。

この表から単純に判断するにすぎず、 $\#1$ システムでは、5多重、 $\#2$ システムでは4多重で実行するのは、処理能力と見合った運用方式であるといえる。長時間ジョブの場合でも多重度がこれより落ちると、ロールアウト率は多少下がるが、やはりメモリ・バウンドとなり、CPUアイドルを増加させる。たとえば、 $\#2$ システムで、多重度が3以下で、長時間ジョブが処理されている場合、ロールアウト率30%を考慮すると、平均2ジョブしかメモリ内にないわけだから、2-CPUのシステム構成では、CPUのアイドルが当然のことながら大きくなる。一方、多重度を、これ以上上げると、 $\#1$ システムでは短時間ジョブのターンアラウンドを阻害することになり、 $\#2$ システムでは、ロールアウト率が増加して、オーバヘッドが大きくなり、実行多重度は上らない。

8.4 プライオリティとターンアラウンド・タイムの比較

第18表は、プライオリティの分布と、ターンアラウンド・タイムを示したものである。 $\#2$ システムは、クローズ方式だから、ターンアラウンド・タイムというよりは、システム滞在時間、すなわちジョブ入力開始から出力終了までを示すことにする。受付後ジョブ入力までと、出力後資料配布までのオペレータの介入時間はこの中に含まれていない。

$\#1$ システムでは、約半数のジョブのプライオリティが7となっている。7, 6, 5で90%程度を占める。一方 $\#2$ システムでは、7, 6, 5をあわせて64%、4, 3, 2, 1, 0のジョブの比率もかなり大きい。 $\#1$ システムのターンアラウンド・タイムは平均7.6分と良好で、プライオリティが7の時は4.4分とかなり早い。 $\#2$ システムでは、プライオリティに比例して、システム滞在時間は大きくなる。平均40分程度、大きいジョブでも100分以内であるから、オペレータが努力すればかなり早いターンアラウンドを期待できる。混雑時にオペレータによジョブ入力を制限することを考慮するとしても、システム滞在時間の倍程度のターンアラウンド・タイムが望ましい。

第13表 平均多重度・平均CPU使用時間

No.1 システム

(1973年8月)

日時	ジョブの多重度	処理件数	平均CPU 使用時間	平均CORE 使用量	平均出力量
8月1日(水) 10.42~17.00	3.7	278	79.9S	29.1K	1369.1行
8月2日(木) 9.12~13.42	2.8	176	59.5秒	30.9K	1006.8行
14.22~17.10	4.5	118	113.0秒	32.0K	1461.8行
8月3日(金) 9.22~17.00	2.8	347	59.8秒	32.4K	1133.2行
8月4日(土) 9.08~11.42	2.8	124	55.2秒	32.6K	1005.3行

(システム・ダウン時, メンテナンス時, システムの閉鎖時を除く)

第14表 平均多重度・平均CPU使用時間等

No.2 システム

(1973年8月)

日時	ジョブの多重度	処理件数	平均CPU 使用時間	平均CORE 使用時間	平均出力量 (プリンタ・カード)
8月1日 9.54~17.00	3.2	58	10分12秒	50.7K	2769.8行
17.00~23.00	2.9	45	8分	50.0K	2795.4行
8月2日 0.00~5.08	2.1	12	23分39秒	93.2K	1419.4行
12.48~17.00	3.2	49	5分23秒	45.4K	1649.8行
17.00~23.30	3.4	67	8分43秒	50.2K	2620.2行
8月3日 0.00~5.40	2.0	8	26分20秒	109.4K	4352.6行
9.06~17.00	3.6	83	5分26秒	38.5K	2359.0行
17.00~23.40	2.1	73	5分45秒	42.9K	1466.3行
8月4日 0.00~8.30	3.0	29	22分47秒	79.2K	2989.6行
9.00~14.00	3.6	39	10分3秒	47.3K	2530.1行

空領域管理プログラムは多重度には入っていない。

(システムダウン時, メンテナンス時, 日付変更時, システムの閉鎖時を除く)

第15表 CPU使用時間詳細

(1973年8月・9.00~12.30, 13.30~16.30の全ジョブ)

CPU使用時間	No.1 システム		No.2 システム	
	ジョブ数(%)		ジョブ数(%)	
0.1分まで	2180 (28.8)			
0.3	1745 (23.1)		761 (34.5)	
0.6	1079 (14.3)		197 (8.9)	
1.0	802 (10.6)		224 (10.2)	
2.0	705 (9.3)		275 (12.5)	
5.0	679 (9.0)		273 (12.4)	
10.0	265 (3.5)		171 (7.8)	
20.0	113 (1.5)		154 (7.0)	
30.0			67 (3.0)	
160.0			63 (2.9)	
120.0			20 (0.9)	
合計	7586		2205	
平均	6分2秒		6分39秒	

第16表 ジョブ・ステップの分析

(1973年8月全ジョブ)

比率 ジョブステップ名	No.1 システム		No.2 システム	
	ジョブステップ・数比	処理時間比	ジョブ・ステップ数比	処理時間比
FORTRAN	25.8%	14.8%	17.0%	2.0%
LIED	27.4	0.9	22.4	1.9
LIBE	10.2	7.4	18.2	0.7
FSUPROGM (独立ユーティリティ)	4.7	0.0	6.3	0.0
FASP	0.1	0.6	0.1	0.0
COBOL	0.9	0.3	1.5	0.2
NONANE ユーザ	26.9	67.9	19.9	69.1
その他 プログラム	4.0 } 30.9	8.1 } 76.0	14.6 } 34.5	26.1 } 95.2
全ジョブ・ステップ数	22,840		11,719	
全処理時間	498,739.1(s)		1,619,047.9(s)	

表17 ロールアウト率

(1973年8月全ジョブ)

No.1 システム (オープン・バッチ システム)	コア・タイム (81.0%)		ロールアウト タイム(19.0%)	ジョブ処理件数 8079 9.00~17.00まで オープン
	CPUタイム (26.5%)	ウエイト・タイム (54.5%)		

No.2 システム (クローズ・バッチ システム)	コア・タイム (70.7%)		ロールアウト・ タイム(29.3%)	ジョブ処理件数4058 平日24時間運転
	CPUタイム (30.3%)	ウエイト・タイム (40.4%)		

(全ジョブ)

第18表 プライオリティとターンアラウンド・タイム

(1973年8月, 9.00時~12.30分, 13.30~14.30分の全ジョブ)

プライオリティ	No.1 システム		No.2 システム	
	ジョブ数(%)	ターンアラウンド・タイム分	ジョブ数(%)	システム滞在時間
7	3789 (50.1)	4.4	695 (31.5)	20.5
6	1272 (16.8)	7.2	505 (22.9)	47.3
5	1589 (21.0)	9.6	224 (10.2)	29.0
4	477 (6.3)	15.4	143 (6.5)	28.8
3	227 (3.0)	21.3	130 (5.9)	40.6
2	86 (1.1)	21.4	125 (5.7)	53.2
1	120 (1.6)	20.1	178 (8.1)	58.2
0	68 (0.1)	20.7	205 (9.3)	91.9
合計	7568		2205	
平均		7.6分		40.8分
標準偏差		8.9分		50.3分

9. ジョブの到着分布、サービス分布、システム負荷

9.1 ジョブの到着分布

サービスを提供する窓口に来客が1人づつくるものとして、ある時点から n 番目の客が窓口に着する時刻を t_n とする。 $t_{n+1} - t_n$ すなわち客の到着時間間隔は、同一の分布をもつ確率変数とみなし、その分布を到着分布という。 M_1 システムは、オープン・バッチ処理だから、計算機に入力するジョブにあてはめてみる。このような到着分布は指数分布となることが多い。到着分布が指数分布であるとき、ある時間間隔 $(a, a+t)$ 内に到着する客数(ジョブ数)は、ポアソン分布にしたがう。時間間隔内に到着するジョブ数についての分布は、 M_1 システムでは、ポアソン分布よりはむしろ正規分布に近いことはすでに述べた。

さて、 M_1 システムの到着分布について調べてみたところ、累積分布関数: $F(t)$ は第15図のとおりで、指数分布で近似できる。到着分布: $f(x)$ は、

$$f(t) = F'(t)$$

となり、やはり指数分布となる。ジョブの平均到着間隔は72.5秒となっている。

第15図より、ジョブの到着分布は、次のように近似できる。

$$F(t) = 1 - e^{-0.013t} \quad (1)$$

$$f(t) = 0.013 e^{-0.013t} \quad (2)$$

9.2 ジョブのサービス分布

ジョブのサービス分布は、1つのジョブに提供されるサービス時間の分布であり、指数分布や他の一般分布にしたがうことが知られている。

計算機システムの場合、ジョブのサービス時間の定義は難しい。ジョブがシステムに滞在している時間のうちどれ程がシステムからのサービスを受けているかを定式化できない現状である。多重ジョブ処理方式で、入力、出力処理との並行処理も行っているわけだから、ジョブは、いろいろなステージでのサービスを受けているはずである。しかしながら、ここでは、データ収集が容易なことから、CPU使用時間をサービス時間と考えることにする。

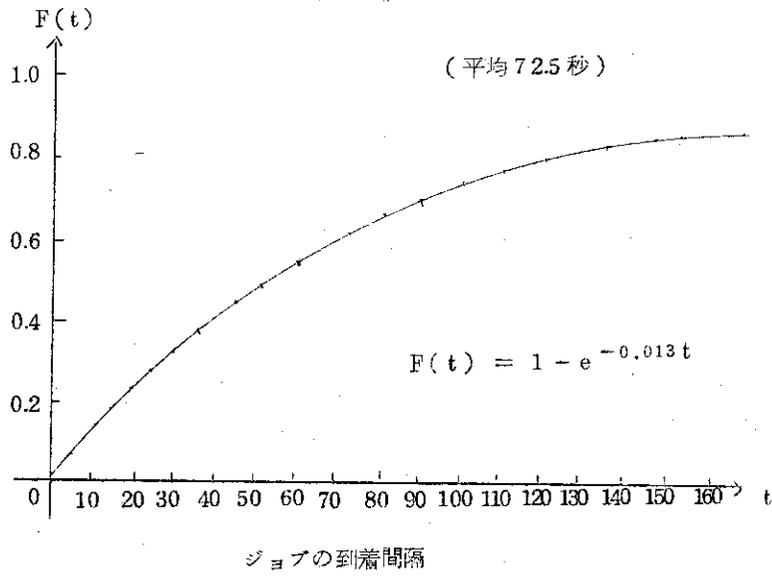
サービス分布は、 M_1 システムと M_2 システムでは、当然のことながら異った分布となる。累積度数分布は、第16図で示す。次式のように指数分布を変形した式で近似できる。

$$M_1 \text{ システム } F_1(t) = 1 - e^{-0.2\sqrt{t}} \quad (1),$$

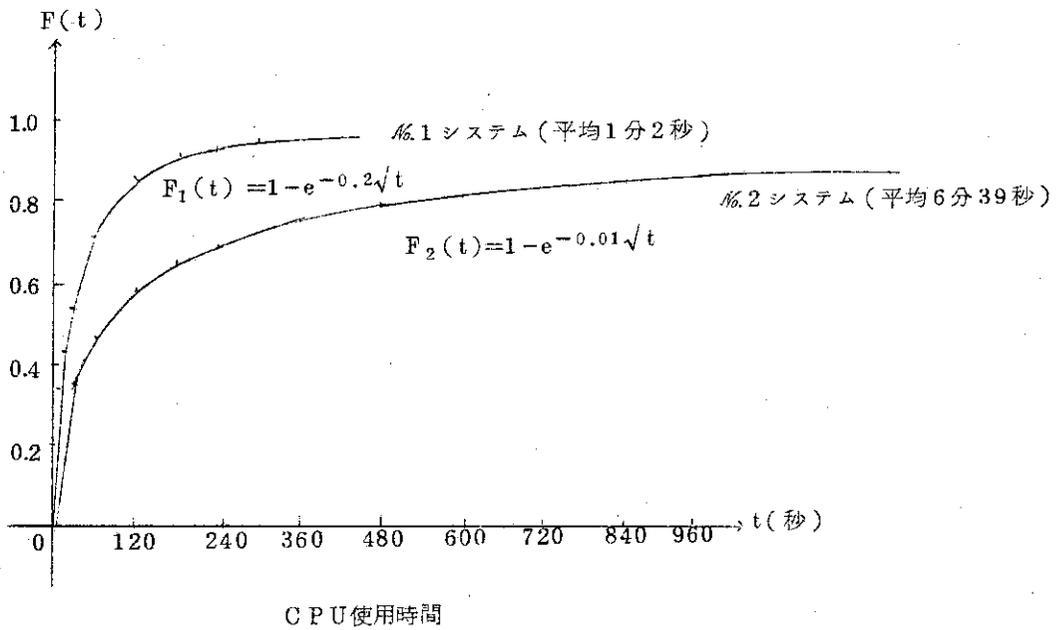
$$f_1(t) = \frac{0.1}{\sqrt{t}} e^{-0.2\sqrt{t}} \quad (2),$$

$$M_2 \text{ システム } F_2(t) = 1 - e^{-0.07\sqrt{t}} \quad (3),$$

$$f_2(t) = \frac{0.035}{\sqrt{t}} e^{-0.07\sqrt{t}} \quad (4),$$



第15図 ジョブの到着分布—累積分布関数—



第16図 ジョブのサービス分布—累積分布関数—

$F(t)$ をサーヴィス分布の累積分布関数とすると、平均残存時間 (mean residual life time), すなわち、 t 時間のサーヴィスを受けた後、残り何時間サーヴィスを受けるか、は次式のように計算される (2)。

$$G(t) = \frac{\int_t^{\infty} (1 - F(x)) dx}{1 - F(t)} \quad (5)$$

今、 $\#1$ システム、 $\#2$ システム共に $F(t)$ は、

$$F(t) = 1 - e^{-b\sqrt{t}}$$

の形をしているから、

$$H(x) = \int \{1 - F(x)\} dx = \int e^{-b\sqrt{x}} dx \quad (6)$$

$$= -\frac{2}{b} e^{-b\sqrt{x}} \left(\sqrt{x} + \frac{1}{b}\right) \quad (6)$$

$\lim_{x \rightarrow \infty} e^{-b\sqrt{x}} \cdot \sqrt{x} = 0$ だから

$$G(t) = \frac{H(\infty) - H(t)}{e^{-b\sqrt{t}}} = \frac{2}{b} (\sqrt{t} + b) \quad (7)$$

$G(t)$ は増加関数となるから、現在少ししかサーヴィスを受けていないジョブ程、早く処理が終るといえる。システム中に滞在するジョブ数を少なくすることを、ジョブ・スケジュール上の目安とするならば、短時間ジョブほど先に処理するという原研方式のプライオリティのつけ方は、意味があると考えられる。

システムがもっと単純なモデルなら、到着分布とサーヴィス分布がわかれば、待ち時間等を待ち行列の理論により理論的に計算することが可能である。原研システムに適用するには、もっと基礎データを取ると共に、詳しい解析が必要となる。

9.3 システム負荷

ジョブの到着分布とサーヴィス分布の関係、さらに CPU 利用状況から、 $\#1$ システムに対して、以下の推測ができる。

ジョブの平均到着時間間隔 (t)

ジョブの平均 CPU 占有時間 (s)

とする。システム・オーバヘッド、入出力処理等に対して必要な CPU のアイドル時間を考慮すると、7. で述べた原研における CPU 利用状況から、ジョブの平均 CPU 占有時間は、次のように考えられる。

$$s = \text{平均 CPU 使用時間} \times \left\{ 1 + \frac{\text{平均システム・オーバヘッド率} + \text{平均入出力等待時間率}}{\text{平均 CPU 使用率}} \right\}$$

7. の結果から、 $\#1$ システムでは、

$$s = 60.2 \left(1 + \frac{23.5 + 17.7}{3.85} \right) = 124.6 \quad (\text{秒}) \quad .$$

第15図から

$$t = 72.5$$

システムへの負荷は、平均CPU占有時間と平均到着時間の比で示されるが、システムが、2-CPU構成であることを考慮すると、

$$\frac{s}{2} \cdot \frac{1}{t} = 0.86 \quad .$$

この結果から、1システムのシステム負荷率は、約86%と考えられる。14%の余裕がある状態で使用されているといえる。ただし、ジョブの到着分布は比較的忙しい時期を選んで出しているので、1日の平均では、 $t = 85$ となり、負荷率は、その場合、73%程度に下がる。

10. データ・チャンネル、ファイル・コントローラ、ドラムの使用状況についてのハードウェア的測定

既に述べたように、 $\mu 1$ システムは、ドラムの入出力処理を分担するセクタ・チャンネル（C-チャンネル）が低速なために、C-チャンネルの使用率がきわめて高くなっていることは、4.でソフトウェア測定結果により明らかである。

ドラムに関連する入出力機器（第17図）の使用率を、もう少し詳しく調査することになり、今回、ハードウェアの制御信号を、ユニバーサル・スケラで測定し、機器の使用率等を出したのでその結果を報告する。

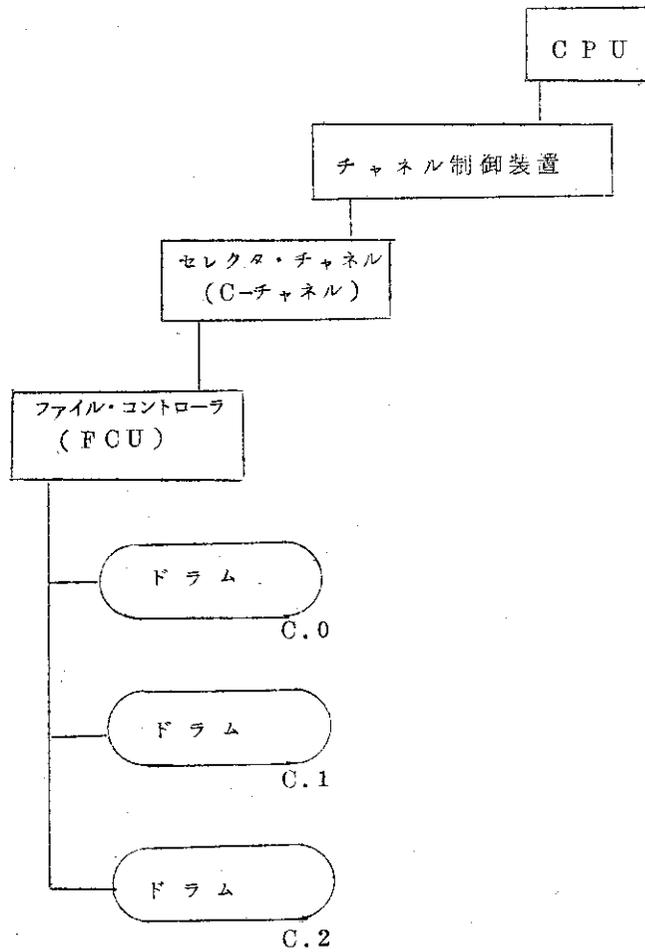
第19表は、データ・チャンネルのアクセス回数について示している。A、Bチャンネルには、ディスク、C-チャンネルはドラムと磁気テープ、F-チャンネルには、ラインプリンタ、カードリーダーとタイプライタが接続されている。A、B、Cは、セクタ・チャンネルで、Fはマルチプレキサ・チャンネルである。（第18図）。第19表には、各チャンネル毎のアクセス数と、チャンネル使用合計（T）を100とした場合の使用比率が示されている。C-チャンネルが約50%で、アクセス回数の大半を占める。ドラムのアクセス時間や転送時間は他のものに比して小さいとはいえ、C-チャンネルの比率が高すぎると考えられる。F-チャンネルは、マルチプレキサなので、時分割モードであることから、他のチャンネルとの比較は難しいが、この表では連続モードに換算した値を用いている。

第20表は、ファイル・コントローラのアクセス回数とリジェクト回数を示している。リジェクト率は思ったより小さいが、これは、ハードウェアのリジェクトのみを示すもので、ドラムへのアクセス要求が出されたとき、ソフトウェア的にドラムへのアクセスを拒否されるような場合を含んでいない。リジェクト率は10%以下である。

第21表は、ドラムのリザーブ回数を示す。接続されている3台のドラムに対して、各々のリザーブ回数が示されている。使用比率からすれば、C・0が約半数を占め、ドラムの利用法に一考を要することがわかる。C・0は、非常駐のシステム・タスクとエラーメッセージ・ファイル等に使用されている。

第22表と第23表は、ドラムのリザーブ時間を示すものである。時間比では、C・0のみならずC・2も大きいことがわかる。C・2は、FORTRANライブラリとして主に使用されている。1回のアクセスに対するブロック長が、他のファイルに比して大きいと考えられる。第23表は、同じくドラムのリザーブ率を示す。第22表は、全リザーブ率を100としたときの個々のドラムに対する百分率であったが、第23表は、単位時間あたりのC-チャンネルのリザーブ時間、いわゆる使用率について示している。測定データが少ないので、断言できないが、使用率は50%程度と考えられる。以前に、ソフトウェア的にC-チャンネルの使用率を測定した場合と比較すると小さい値となっている。これは、夏休み休暇中でシステムが空いている時間帯でハードウェア測定を行ったこともあるが、ドラム間で並行に処理されている作業の一部が、ソフトウェア測定では、直列にカウントされていることにもよる。いづれにしてもCPUの使用率に比較してC-チャンネルの使用率は、高すぎるといえる。

今回の測定は、ドラムを中心としたものに限定されるが、もっと多くの測定、たとえば、CPUとデータ・チャンネルのオーバラップなどを行なうと、もっと興味ある評価が得られるものと考えられる。

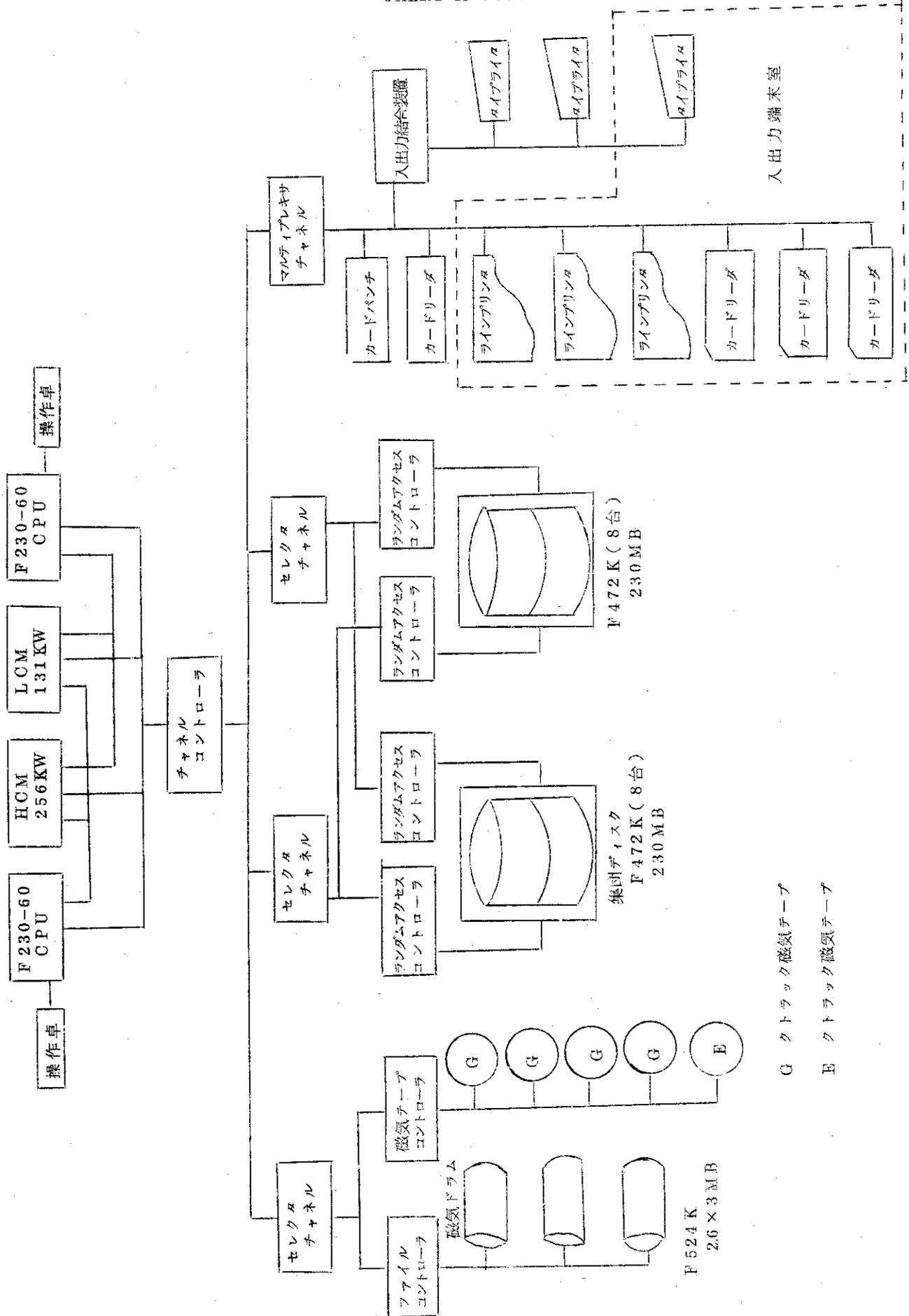


ドラム 1台につき 2.5Mバイト

平均アクセスタイム 17m秒

情報転送速度 156Kバイト/秒

第17図 ドラムに関連する入出力機器



G クトラック磁気テープ
E クトラック磁気テープ

第18図 系1 システムのハードウェア構成

第19表 チャンネルのアクセス回数

48.9.1~9.4

測定時間	A·ch	A/T%	B·ch	B/T%	C·ch	C/T%	F·ch	E/T%
50分	233630	26.0	208322	23.2	435155	48.4	21980	2.4
60	208137	24.3	211452	24.7	404366	45.4	32743	5.6
65	236266	21.3	254355	23.0	569313	51.4	47750	4.3
85	389593	24.4	377036	23.7	723462	45.4	104089	6.5
45	230527	24.2	229766	24.1	427563	44.8	65614	6.9
55	287912	25.7	261518	23.3	494754	44.2	76289	6.8
60	330391	23.1	310139	21.7	683267	47.7	107654	7.5
85	321390	19.4	324640	19.6	898157	54.1	113814	6.9

$$T = (A+B+C+F) \text{ ch.}$$

第20表 ファイル・コントローラへのアクセス回数とリジェクト回数

測定月日	アクセス回数	REJ.回数	REJ/ACCE %
S48. 8. 8	4160090	311241	7.5
8.10	3968038	297223	7.5
8.13	4627032	342262	7.4
8.14	4911866	401245	8.2
8.15	4712030	377169	8.0
8.17	4317128	346066	8.0
8.20	4278623	370000	8.6
8.21	3454657	270960	7.8

第21表 ドラムのリザーブ回数

測定月日	開始時刻	終了時刻	測定時間	C.0回数	C.0/T	C.1	C.1/T	C.2	C.2/T
7.27	9:45	10:20	0. ^H 35 ^M	20544	47.1	8190	18.8	144881	34.1
#	10:35	11:35	1.00	20802	44.8	8775	18.9	166853	36.3
#	11:36	13:32	1.56	28169	45.2	11464	18.4	222689	36.4
#	13:33	14:49	1.16	36180	46.3	15211	19.4	266820	34.3
7.30	9:30	17:30	8.00	152922	46.1	61994	18.7	1166686	35.2
7.31	9:30	8 ¹ / ₉ :00	23.30	429780	65.3	75138	11.4	1533648	23.3
8.1	10:39	17:27	6:48	178070	44.8	71189	17.9	1488079	37.3
8.2	9:20	17:38	8:18	185592	44.4	74184	17.8	1577802	37.8

$$T=C.0+C.1+C.2$$

第22表 ドラムのリザーブ時間

測定月日	開始時刻	終了時刻	測定時間	C.0時間	C.0/T	C.1	C.1/T	C.2	C.2/T
8.3	10:27	17:30	7.03	5691S				6309S	
8.6	9:30	16:07	6.37	4353	45.3	1247	13.0	4015	41.7
8.7	9:30	17:00	7.30	6900	41.9	2186	13.3	7382	44.8

第23表 ドラムのリザーブ時間比

測定月日	開始時刻	終了時刻	測定時間	ジョブ数	平均多重度	Cチャンネル/システム×100	(時間比)
8.1	10:39	17:27	6:48	285	3.5		
8.2	9:20	17:38	8:18	294	3.0		
8.6	9:30	16:07	6:37	209	1.8	40.4	
8.7	9:30	17:00	7:30	353	3.3	61.0	

(磁気テープは含まず)

11. おわりに

48年度前期に行ったF230-60のシステム効率に関する測定データを取りまとめた。データを収集するために、会計情報を利用したり、ソフトウェア、ハードウェア測定を行う。会計情報ファイルは、ジョブ毎の情報として、とりまとめられているので、多重ジョブ処理を行う際の、ジョブ間の競合についてとか、機器の全般的な使用状況とか、時間の経過に対する事象の変化を動的にとらえることができない。

ソフトウェア測定は、すでに述べられているように、SPIEマクロにより、割込処理を横道に引込み、ユーザ側からシステム内のファイルを覗く方法である。ユーザ・プログラムで測定を実行できる点ですぐれているが、オーバヘッドが多すぎて困る場合がある。たとえば、データ・チャンネルA, B, C, Fの利用状況を同時に測定するとすれば、1回の測定に数秒を要する。これは、ユーザ側からモニタ領域を覗くとき、領域読込侵害となり、モニタ・エラーの扱いとなりその処理に時間を費すためである。測定プログラムをモニタ・タスクとして使用する方法をとるべきであろう。モニタ内では、モニタ・タスクは、常駐タスク(RT)、非常駐タスク(TT)、ユーザ領域内の非常駐タスク(TTU)のいずれかの資格で動くことになる。測定プログラムがTTUとして組込まれるべきである。

ハードウェア測定は、測定値の正確さにおいて申し分ない。我々の行ったのは、ユニバーサル・スケーラを測定対象物に1つ(使用回数を調査するとき)、または2つ以上(使用時間を調査するとき)接続して、定時間内の使用回数または使用時間をカウンターから読取る方法である。システムの主たる入出力機器の使用率を得ようとすれば、数10台のユニバーサル・スケーラを必要とし、原研内で調達するのが難しくなる。部分的にしか測定を実行できないので同一事象を同時に追跡できない。また測定したデータを自動的に記録するデータ・レコーダ付のスケーラがあれば、もっとハードウェア測定が便利になる。

CPUの使用率のみがシステム評価の問題とは考えられないが、いろいろな項目のうち最も解り易く、影響力が大きなものとして、今回とりあげてみた。

参考文献

- (1) 北川一, 他; オペレーティング・システムのパフォーマンス・モニタリング, 情報処理学会誌13-11(1972)。
- (2) W. Freiberger; Statistical Computer Performance, Academic Press(1972) Press (1972)。
- (3) N.R. Draper, H. Smith (中村慶一訳); 応用回帰分析, 森北出版(1967)。
- (4) 石黒, 山田; CPUアイドルの解析, 情報処理学会大会(1973, 東京)。