

NAIS

イメージチェッカ

IMAGE CHECKER 30P・30RP リファレンスマニュアル コマンド

Ver 3P(2)/Ver 3RP(2)



松下電工の制御機器は
グローバルブランド **NAIS** に統一します。

はじめに

イメージチェッカ30P・30RPは、生産製造ラインにおいて要求される製品の外観、形状検査、寸法測定や種分け
また、対象物の面積・位置・傾き・個数などを非接触にて高速高精度で計測判断する装置です。

イメージチェッカ30P・30RPは専用言語を使用することで、より一層その応用範囲が広がり、今まで、困難であった検査・測定までが可能となっています。

本マニュアルは、専用言語についての説明を主体とし、サンプルプログラムを添付しております。

操作方法につきましては操作編を、また、品種モードにつきましてはイメージチェッカ30または、30Rマニュアル（ハード・品種）を参照願います。

目次

はじめに
イメージチェッカ30P・30RPマニュアル一覧
目次

第1章 イメージチェッカ30P・30RPプログラム仕様

1

1. プログラム仕様	2
[1] プログラムの構成	2
[2] 使用可能文字	3
[3] 命令文の構成	3
[4] コメント	3
[5] オペランド	4
[6] レジスタ	5
(1) vレジスタ	5
(2) aレジスタ (文字用レジスタ)	6
(3) 直接数値レジスタ	7
(4) チェッカ結果レジスタ	8
(5) Sレジスタ	15
(6) Bレジスタ	15
2. 補足	17
[1] 画面構成	17
[2] ポート構成	20
(1) パラレルポート	20
(2) シリアルポート	21
[3] プリンタとの接続	22
(1) セントロニクスプリンタ	22
(2) シリアルプリンタ	23
(3) 漢字キャラクタの制御コードについて	23
プログラムモードで回転補正を実行する際の注意	24
イメージチェッカ30P・30RPのプログラムのフローチャートの一例	25

ABS	30	NOT	94
ADD	31	OR	95
AND	32	OVOFF	96
ATAN	33	P-IN	97
CALL	34	P-OUT	99
CBIT	35	PRINT	101
CLRCRT	36	RBIT	103
CLRREG	37	READ	104
CODE	38	READF	106
COS	40	READY	108
C-OUT	41	REND	109
CSCAN	44	RET	110
CURSOR	45	RSCAN	111
DELAY	46	SBIT	112
DISP	47	SCAN	113
DISPD	49	SHOW	115
DIV	52	SIN	118
D-OUT	53	S-IN	119
DSCAN	57	SIOBR	120
ESCAN	58	SIOCL	121
FSCAN	60	SIOPB	122
GOSUB	62	SIOSB	123
GOTO	63	S-OUT	124
IF	64	SPRINT	126
IFNSTR	65	SQRT	127
IFSTR	66	START	128
ISCAN	67	STROB	129
KEYIN	69	SUB	130
KEYSNS	70	TAN	131
LEN	71	(TANG	132)
LOCATD	72	(TBIT	133)
LOCATE	74	(TCENT	134)
LSCAN	75	THRES	135
MLCLR	77	TMRD	136
MLOFF	78	TMWT	137
MLON	79	(TSCAN	138)
MONT	80	(TSHIFT	140)
MOVE	82	(TPRIN	141)
MOVEBL	85	(TWIDTH	142)
MOVEBW	86	TYPE	143
MOVELB	87	WHILE	144
MOVELW	88	WHILER	146
MOVEWB	89	WSCAN	147
MOVEWL	90	XOR	149
MSCAN	91		
MUL	93		

() は、30RPのみのコマンドです。

第3章 イメージチェッカ30P・30RPプログラム例

151

プログラムモードで回転補正を実行する際の注意	152
イメージチェッカ30P・30RPのプログラムのフローチャートの一例	153
三角比モードのプログラム例	154
[1] 検査物に合わせ、自動で品種切替え	155
[2] 2値化レベルを変えて画像を読み込み	157
[3] チェッカ判定結果により、検査チェッカを自動切替え	159
[4] 数値演算結果により、検査チェッカを自動切替え	161
[5] 2つのチェッカ判定結果により、検査チェッカ・判定出力の自動切り替え	163
[6] ウィンドウを任意に移動(特徴抽出での重心位置へ移動)	165
[7] ピッチ間隔の測定・パラレル出力	167
[8] 傾き角度の測定・パラレル出力	169
[9] 検査物の合格率計算	171
[10] 検査・測定の分解能の自動計算	174
[11] モニタ画面へのコメント表示	176
[12] シリアル通信例	178
[13] エラー発生時の画面表示	184

付録

187

[1] 漢字コード一覧表	188
[2] ASCIIコード一覧表	197
[3] パラレル入出力	198
イメージチェッカ30P・30RPのパラレル入出力ポート、RS232Cピン配置図	199
[4] プログラム入力シート	200
[5] エラーメッセージ一覧	201

マニュアル改訂履歴

203

1000

1000

C

C

イメージチェッカ30P・30RPプログラム仕様

1. プログラム仕様

- ・プログラムは最大64ステップ/1ファイルまで書き込めます。
- ・1ステップの文字入力は半角英数字で最大59文字までです。(但し、漢字1文字は2文字に担当します。)
- ・プログラムはイメージチェッカPタイプコントローラ本体に最大99ファイル(PM1~PM99)、ICカードに最大99ファイル(PI1~PI99)まで、登録できます。
- ・M. I. S. T. とパソコンを使用して、フロッピーにファイルとして登録する場合は、M. I. S. T. に付属していますマニュアルを参考ください。

[1] プログラムの構成

L 1 Δ MOVE Δ V 1, V 2
ラベル コマンド オペランド

- ・ラベル、コマンド、オペランドの順で構成され、各々1文字以上のスペース(Δで示す)で区切られています。
- ・また、オペランドが2つ以上ある場合、オペランド間はコンマ(,)で区切ります。

①ラベル……英大文字ではじまる2文字

②コマンド……英大文字6文字で構成

ラベルが書かれていなくても、前3文字分余白にしておいてください。

③オペランド…各コマンドへの引き数(最大4オペランド)

直接数値レジスタ等24種類

右図にプログラム表示エリア内

の各エリアを示します。

④ステップNo. 表示エリア

…キー入力により自動的

に次のステップを表示します。

⑤ラベル表示エリア

…… (2文字分)

⑥コマンド表示エリア

…… (6文字分)

⑦オペランド漢字エリア

…… (44文字分)

1:		CLRCRT	
2:	ST	MOVE	1, V1
3:		MOVE	2, V2
:			
:			
:			
	Ⓐ	Ⓑ	Ⓒ
			Ⓓ

[2] 使用可能文字

- (1) 英大文字
- (2) 英小文字 (コメントで使用可)
- (3) 数字
- (4) 特殊記号 (コメントで使用可)

[3] 命令文の構成

命令文は ラベル、コマンド、オペランド から構成されています。

A1 Δ MOVE Δ 10, V1
ラベル コマンド オペランド

ラベルとコマンド、コマンドとオペランド間は、空白 (上記ではΔで表現されている) が1字以上必要です。但し、コマンドは4文字目から書き始めます。

各部分の制約事項を述べます。

- (1) ラベル ジャンプ先を示します。英大文字で始まる2文字です。同じプログラムに2つ以上同じスタイルのものがないこと。
- (2) コマンド 英大文字6文字までで構成されています。
- (3) オペランド 各コマンドへの引数で、コマンドにより最大4オペランドまで指定できます。

[4] コメント

コメント文は、1文字目に「*」を入力してください。以降の文字列をコメントと判断します。

但し、入力文字数は最大30文字までです。

注意:

コマンドによっては、オペランドを省略することができますが以下の場合には、省略できません。

- ① オペランドを2つ以上使用する場合、1番目のオペランドを省略すること。
- ② オペランドを3つ以上使用する場合、中間のオペランドを省略すること。

省略しない場合には、'0' を書くなどの方法をとってください。

<文例> WSCAN 1, 10, 0, 100

[5] オペランド

オペランドには、次の表に示す様に16種類があり、各コマンド毎に使用できるオペランドには制限があります。 () は、30RPのみの機能です。

オペランド	内 容	使用例	制 約 事 項
NULL	オペランド無し RET	RET	
i	直接数値 (Immediate)	MOVE #23, V1	-32768~32767
h	直接数値 (Immediate-HEX)	MOVE 0XDE, V1	0X0000~0XFFFF
G	1文字	IF A1, EQ, +3	' ' でくくってください。
" "	文字列	DISP 3, 3, "HELLO", 6	文字数はMAX15文字
l	ラベル名	GOSUB #1	同じ名前のラベルが2つ以上ないこと 尚、大文字で始まる 2文字であること
f	ファイルNO.	CALL PM1	PM1~99 PI1~99
z	レジスタ間接指定	MOVE (V8), V2	vレジスタのみ
v	汎用数値レジスタ	ADD V1, V2, V3	V1~V1000
a	汎用文字レジスタ	MOVE "Y", A1, 1	A1~A1000
S	パラレル出力ポート読み込みレジスタ	MOVE DS1, V1	リードのみ
B	エラーレジスタ	MOVE DB2, V100	リードのみ
q	判定条件: I F	IF V1, V2, EQ, L2	EQ, NE, GT, GE, LT, LE,
b	判定条件: T B I T文	TBIT V1, V2, NZ, L2	NZ, ZR
j	チェッカ結果レジスタ		I, E, L, W, M, C, R, D, F, (G) (T, X I, Y I)
d	記号		ON, OF, SA, SB, A, B, etc.

[6] レジスタ

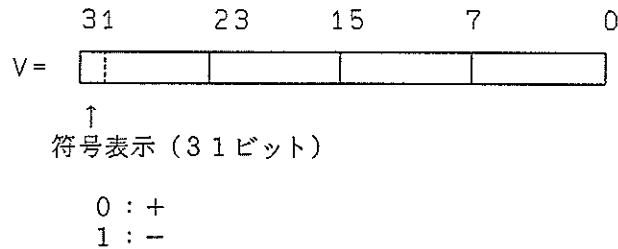
(1) vレジスタ

a. 符号付き32ビット構成です。 ($-2^{31} \sim 2^{31} - 1$)

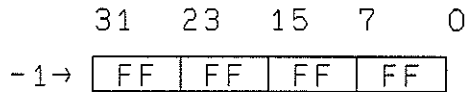
(参考) 負の数値の取扱いについて

- 10進数→2進数のとき符号をとって1を引き2進数表示し、ビット反転させる
- 2進数→10進数ビット反転し、10進数表示した数字に1を加え符号(-)を付ける

b. 指定は V1 ~V1000 です。



(例) -1の場合の格納の方法



c. () を付けて間接指定できます。

間接指定とは ----- 括弧をつけた vレジスタの内容を示す NO、の vレジスタを指定します。

vレジスタを使用しているほとんどのコマンドで使用できます。

(例) V1 = 10 のとき

(V1) = V10

(例外) チェッカ結果レジスタには使用できません。

CF [(V1)]、(V2)]

DW [(V1)]

d. vレジスタにより、aレジスタのNo. 指定が行えます。

(書式) A [v]

- ・ [] でくくられた vレジスタの内容を示す、AレジスタのNo. を指定します。
- ・ vレジスタの内容範囲は、1~1000です。

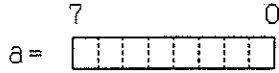
<文例> MOVE 10, V1 ; 10→V1
 MOVE A [V1], A1, 1 ; A [V1] = A10→A1

注意: MOVE A [(V1)], A1, 1 の様な指定は、できません。

(2) aレジスタ (文字用レジスタ)

a. 指定範囲は A1~A1000 です。

b. 符号無し8ビット構成です。(→1文字のみの格納です。)



c. 漢字を入力するとき

- ・ J I Sコードで入力し、内部でシフト J I Sコードに変換されます。
- ・ または、音読み入力で漢字入力を行い、内部では、シフト J I Sコードに変換されます。(V e r 3以降)
- ・ 変換されたシフト J I Sコードは、2個の aレジスタに格納されます。
- ・ 格納の方法は、シフト J I Sコードの1バイト目は若いNO. の aレジスタに、2バイト目は次のNO. の aレジスタに格納します。

(例) 「円」を入力するとき。

J I Sコードで「円」 = 3 1 5 F を入力します。

または、音読みで「E」を入力します。

内部でシフト J I Sコードに変換され、「円」 = 8 9 7 E になります。

すなわち、

```
MOVE Δ "円",A1,2
```

という命令を実行すると、

A 1 = 0 X 8 9

A 2 = 0 X 7 E

の様にそれぞれ格納されます。

- ・ 画面表示は、漢字1文字で半角英数字2文字分に相当します。

「DISP」又は「DISPD」で漢字表示するとき、文字数の指定に注意してください。

(3) 直接数値レジスタ

- a. 10進数表現「i」と16進数表現「h」があります。
10進数の場合：-32768～32767 の範囲で指定してください。
16進数の場合：0X0000～0XFFFF の範囲で指定してください。
↑
0Xは、Hex (16進数) の意味です。
- b. 符号付き16ビット構成です。

15 7 0



↑
符号表示 (15ビット)

0 : +
1 : -

(4) チェッカ結果レジスタ (j)

a. 下記の9種類あります。

グループ	レジスタ名	記号	結果内容
I	位置補正	I	数値・判定
	露出補正	E	
	ライン	L	
	ウィンドウ	W	
	パターンマッチング	M	
	特徴抽出	F	
	数値演算	C	
	※回転補正	T	
I I	判定出力 (D)	D	判定
	判定出力 (R)	R	
※ I I I	※位置補正(開始座標: X)	X I	数値
	※位置補正(開始座標: Y)	Y I	
	※特徴抽出(主軸角検出)	G	

※: 30RPのみの機能です。

b. 指定の方法

・チェッカのNO. とモードの指定は、イメージチェッカ30・30Rと同様です。

グループ I

レジスタの前に「C」、「D」を付けて指定します。

指定内容	前記する記号
数値結果	C
判定結果	D

- (例) MOVE Δ CL11,V1 : ラインNO. 1のカウンタ数をV1に格納します。
 MOVE Δ CT10,V2 : 主軸角: T10のデータを、V2に格納します。
 MOVE Δ DL11,V2 : ラインNO. 1のカウンタ数判定結果をV2に格納します。
 MOVE Δ DC1,V1 : C1の判定結果をV1に格納します。

グループ I I

レジスタの前に「D」を付けます。

(例) MOVE Δ DD1,V2 : D 1 の判定結果を V 2 に格納します。

(注) オールドデータの引用はできません。

Vレジスタに読み込んでレジスタ内に覚えておいてください。

グループ I I I (30RPのみの機能です。)

レジスタの前に「C」を付けて指定します。

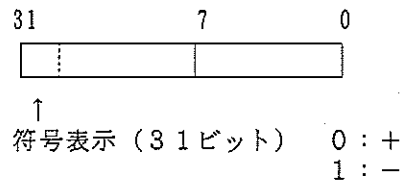
(例) MOVE Δ CX112,V1 : 位置補正No. 1の垂直方向チェッカの
開始点のX座標をV1に格納します。

MOVE Δ CG116,V2 : 特徴抽出(主軸角検出)チェッカNO. 1で
検出した主軸角をV2に格納します。

(注) グループ I I Iでは、オールドデータの引用はできません。

Vレジスタに読み込んでレジスタ内に覚えておいてください。

c. 符号付き32ビット構成です。



- d. vレジスタによるNo. の指定
- ・記号の後に付けるNo. をvレジスタで指定できます。(直接数値でも指定できます)
 - ・演算等を行う場合は直接使用せずに、「MOVE」でいったんvレジスタに読み込んでから行ってください。
 - ・vレジスタ間の間接指定はできません。
 - ・注意：vレジスタによるNo指定はMOVEコマンドのみ指定できます。

<書式>

レジスタ名	数値	判定
位置補正	CI [V(1),V(2)]	CI [V(1),V(2)]
露出補正	CE [V(1)]	CE [V(1)]
ライン	CL [V(1),V(2)]	DL [V(1),V(2)]
ウィンドウ	CW [V(1)]	DW [V(1)]
パターンマッチング	CM [V(1)]	DM [V(1)]
特徴抽出	CF [V(3),V(2)] ※1	DF [V(3),V(2)]
数値演算	CC [V(1)]	DC [V(1)]
●回転補正	CT [V(4)],V(2) ※2	DT [V(5)] ※3
判定出力 (D)	-----	DD [V(1)]
判定出力 (R)	-----	DR [V(1)]
●位置補正(開始座標：X)	CXI [V(1),2] ※5	-----
●位置補正(開始座標：Y)	CYI [V(1),1] ※5	-----
●特徴抽出(主軸角検出)	CG [V(3),V(2)] ※4	-----

●：30RPのみの機能です。

注) V(1)：チェックNo. V(3)：※1参照 V(5)：※3参照
V(2)：モード V(4)：メモリエリア

(文例書式) : 位置補正No. 3の位置補正水平エッジ検出データ(モード1)を
V3に読み込む場合、V1=3、V2=1と指定しますと
MOVE Δ CI [V1, V2], V3 となります。

※1 特徴抽出レジスタ (Fnbm) について

- ・特徴抽出チェック(特徴抽出チェックのNo1~9)の検査結果を引用することができるレジスタです。
- ・このレジスタは数値結果のみ引用することができます。

(1)指定方法

Fnbm n：チェックNo指定(1~9)
 b：対象No指定(チェックNoによります。下記①表参照)
 m：モード指定(0~6：下記②表参照)

①表：対象No指定：V3に相当します。

チェックNo	1~6	7~9
対象No	1~128	1~99

②表：モード指定：V2に相当します。：30RPのみの機能です。

モード	内容
0	特徴抽出測定個数
1	特徴抽出面積(対象No指定)
2	特徴抽出X重心座標(対象No指定)
3	特徴抽出Y重心座標(対象No指定)
4	特徴抽出X射影幅(対象No指定)
5	特徴抽出Y射影幅(対象No指定)
6	特徴抽出周囲長(対象No指定)

数値演算結果のみ引用可能ですので前に「C」を付けて引用可能です。 <例>CF116

(2)vレジスタによるNo指定

特徴抽出チェックと同様の指定で行えます。但し数値演算結果のみの引用です。

- ・書式 CF [V(3), V(2)]
 V(2)：モード
 V(3)：チェックNoを左から1桁目、対象Noを左から2桁目以降で指定します。

- ・文例： MOVE Δ 31, V1 ; 31→V1
MOVE Δ 4, V2 ; 4→V2
MOVE Δ CF [V1, V2], V3 ; CF314→V3

- ※2 回転補正結果レジスタ (Tnm) について：30RPのみの機能です。
- ・「設定モード」の「9. 回転補正」で回転補正後の数値結果を引用できます。
 - ・プログラム中、それぞれのパラメータの設定を行った時は設定された値が引用されます。
 - ・回転補正結果レジスタは数値結果のみ引用することができます。

(1) 指定方法

Tnm n : メモリ (1~4)
 メモリ : A→1 : B→2 : C→3 : D→4
 m : モード (0~9)
 下記表を参照ください。詳しい内容は30Rマニュアルを参照願います。

	三角比モード	主軸角モード
Tn0	検出角度 (Dθ)	主軸角 (Pθ)
Tn1	水平差分 (WX)	——
Tn2	垂直差分 (WY)	——
Tn3	回転角 (Tθ)	回転角 (Tθ)
Tn4	回転中心 (CX)	→
Tn5	回転中心 (CY)	→
Tn6	移動量 (SX)	〃
Tn7	移動量 (SY)	〃
Tn8	$\cos \theta \times 10000$	〃
Tn9	$\sin \theta \times 10000$	〃

- ・書式：数値演算結果レジスタですので、「C」を前に付けてください。
- ・使用可能コマンド：
MOVE, TWIDTH, TANG, TCENT, TPRIN, TSHIFT, IF
- ・文例：
MOVE Δ CT13, V1
TWIDTH Δ 1, CT13, CT14

(2) vレジスタによるNo指定

- ・記号の後に付けるメモリエリアを表わすNoとモード指定がvレジスタにより行えます。(MOVEコマンドのみ使用可能です。)
- ・書式： CT [V(4)、V(2)] V(4) : メモリエリア
 V(2) : モード
- ・文例： MOVE Δ 3, V1 ; 3→V1
 MOVE Δ 4, V2 ; 4→V2
 MOVE Δ CT [V1, V2], V3 ; CT34→V3

(3) チェッカ結果レジスタのvレジスタ指定

IF, TANG, TCENT, TSHIFT, TWIDTH, TPRINコマンドについては、オペランドでチェッカ結果レジスタのvレジスタ指定はできません。

- ・文例：TANG Δ 1, CT [V1, V2] 指定不可
- ・注意：チェッカ結果レジスタのvレジスタ指定はMOVEコマンドのみ使用できます。

※3 エラーレジスタとワーニングレジスタ：30RPのみの機能です。

- ・回転補正のレジスタです。
- ・このレジスタは判定結果のみ引用することができます。

(1) 指定方法

Tn n：レジスタNo（1～8）下記表を参照してください。

メモリエリア	エラーレジスタ				ワーニングレジスタ			
	A	B	C	D	A	B	C	D
記号	T1	T2	T3	T4	T5	T6	T7	T8
正常	1				1			
ワーニング	1				○			
エラー	○				○			

ワーニング：慣性主軸角が決定しなかった場合

エラー：検査対象物体がない場合

・書式：判定結果レジスタですので、「D」を前に付けてください。

・使用可能コマンド：

MOVE, IF, TPRIN

・文例：

MOVE Δ DT3, V1

IF Δ DT1, 1, EQ, L1

(2) vレジスタによるNo指定

・記号の後に付けるメモリエリア指定がvレジスタにより行えます。

(MOVEコマンドのみ)

・書式：DT[V(5)] V(5)：メモリエリア

・文例：MOVE Δ 3, V1 ; 3→V1

MOVE Δ DT[V1], V2 ; DT3→V2

※4 主軸角検出チェッカレジスタ (Gレジスタ) について: 30RPのみの機能です。

- ・主軸角検出チェッカ (特徴抽出チェッカのNo 501~509) の検査結果を引用することができるレジスタです。
- ・このレジスタは数値結果のみ引用することができます。

(1) 指定方法

G n b m $\left\{ \begin{array}{l} n: \text{チェッカNo指定 (501~509} \rightarrow 1 \sim 9 \text{ に対応)} \\ b: \text{対象No指定 (チェッカNoによります。下記①表参照)} \\ m: \text{モード指定 (0~6: 下記②表参照)} \end{array} \right.$

①表: 対象No指定: V3に相当します。

チェッカNo	501~506	507~509
対象No	1~128	1~99

②表: モード指定: V2に相当します。

モード	内容
0	主軸角測定個数
1	主軸角面積 (対象No指定)
2	主軸角X重心座標 (対象No指定)
3	主軸角Y重心座標 (対象No指定)
4	主軸角デルタ (対象No指定)
5	主軸角デルタ (対象No指定)
6	主軸角 (対象No指定)

数値演算結果のみ引用可能ですので前に「C」を付けて引用可能です。 <例>CG116

(2) vレジスタによるNo指定

特徴抽出チェッカと同様の指定で行えます。但し数値演算結果のみの引用です。

<書式>

CG [V (3), V (2)]

V (2): モード

V (3): チェッカNoを左から1桁目、対象Noを左から2桁目以降で指定します。

- ・文例: MOVE Δ 31, V1 ; 31→V1
- MOVE Δ 6, V2 ; 6→V2
- MOVE Δ CG [V1, V2], V3 ; CG316→V3

- ※5 位置補正設定値引用レジスタ：30RPのみの機能です。
- ・位置補正の設定座標値を引用するレジスタです。
 - ・このレジスタは数値のみ引用できます。

(1) 指定方法

- X I n 2 : 垂直方向位置補正の「開始点のX座標」を引用します。
この時、グループNoが他のNoを指定している時は
グループの補正量を加算して引用されます。
(n : 1 ~ 64)
- Y I n 1 : 水平方向位置補正の「開始点のY座標」を引用します。
この時、グループNoが他のNoを指定している時は
グループの補正量を加算して引用されます。
(n : 1 ~ 64)

<書式>

数値のみの引用ですので前に「C」を付けます。
例：CX I 1 2 CY I 3 1

(2) VレジスタによるNo指定

位置補正チェッカと同様の指定で行えます。但し数値演算
結果のみの引用です。

<書式>

CX I [V (1) , 2]
CY I [V (1) , 1] V (1) ; チェッカNo (1 ~ 64)

(5) Sレジスタ

- a. パラレル出力ポートNo. 1の状態を反映します。
- b. 範囲：S1～S8
- c. パラレル出力ポートNo. 1との対応表

ビット	7	6	5	4	3	2	1	0
パラレル出力 使用内容	BATRY	ERROR	READY	STROB	REND	OVFLG	PW-FAIL Ⓢ	FLASH
対応する レジスタ	S8	S7	S6	S5	S4	S3	S2	S1

Ⓢ Ver 3以降で対応。

- d. Sレジスタの値は、“1”または“0”です。また各ビットのON/OFFと同時にレジスタ内容が変化します。

出力ポートの状態	レジスタのビット内容
OFF	0
ON	1

- e. Sレジスタの使用できるコマンドは“MOVE”、“IF”、“CBIT”に限ります。それぞれの書式を下記にしめします。

<書式> MOVE Δ s, v
MOVE Δ s, a, $\begin{bmatrix} v \\ i \end{bmatrix}$

IF Δ $\begin{bmatrix} v \\ i \\ s \end{bmatrix}$, $\begin{bmatrix} v \\ i \\ s \end{bmatrix}$, q, $\begin{bmatrix} i \\ l \end{bmatrix}$

CBIT Δ $\begin{bmatrix} v \\ s \end{bmatrix}$, v, $\begin{bmatrix} v \\ i \end{bmatrix}$

- f. Sレジスタは、「判定結果」のみ取り扱い可能です。よって、レジスタの前には、「D」のみ伴います。

<文例>

MOVE Δ DS3, V1 ; S3→V1

IF Δ DS7, 1, EQ, L1; S7=1ならば、L1へジャンプ

(6) Bレジスタ

- a. 判定出力：B1～B4に対応し、エラーが発生したかどうかを知るためのレジスタです。従いまして、Bレジスタは読みだし専用です。
- b. Bレジスタの値は、“1”または“0”です。

エラー発生の状態	レジスタのビット内容
エラー発生なし	0
エラー発生	1

- c. Bレジスタは、エラーフラグですので「判定結果」のみ取り扱うことが可能です。
従いまして、レジスタの前には「D」のみ伴います。

<書式> DB1, DB2

- d. Bレジスタの使用できるコマンドは、"MOVE"、"IF"の2種類です。
注意：Ver 2では、IFコマンドは、使用できません。

<書式> MOVE Δ B, v
MOVE Δ B, a, $\begin{bmatrix} v \\ i \end{bmatrix}$

<文例> MOVE DB1, V1 ; B1→V1
MOVE DB3, A1, 1 ; B3→A1

(Bレジスタ内容)

レジスタ	内容	正常時	エラー発生時
B 1	位置補正エラー	0	1
※B 2	※回転補正エラー	0	1
B 3	露出補正エラー	0	1
B 4	数値演算エラー	0	1

※：30RPのみの機能です。

<文例> IF DB1, 1, EQ, ER : 位置補正エラーが発生した時は、
ラベルERへジャンプ

2. 補足

[1] 画面構成

(1) 全部で5画面用意されています。

名称		説明	SHOWでの指定記号
カメラから取り込んだ画像	生画像	処理していない画面	V
	2値化画像	2値化された画面	B
メモリ画像	イメージ	表示用	I
	スキャン	走査用	指定できません
	メッセージ*	文字・表等を描く画面	M
	カーソル *	カーソルを描く画面	C
	パターン *	チェッカを描く画面	P

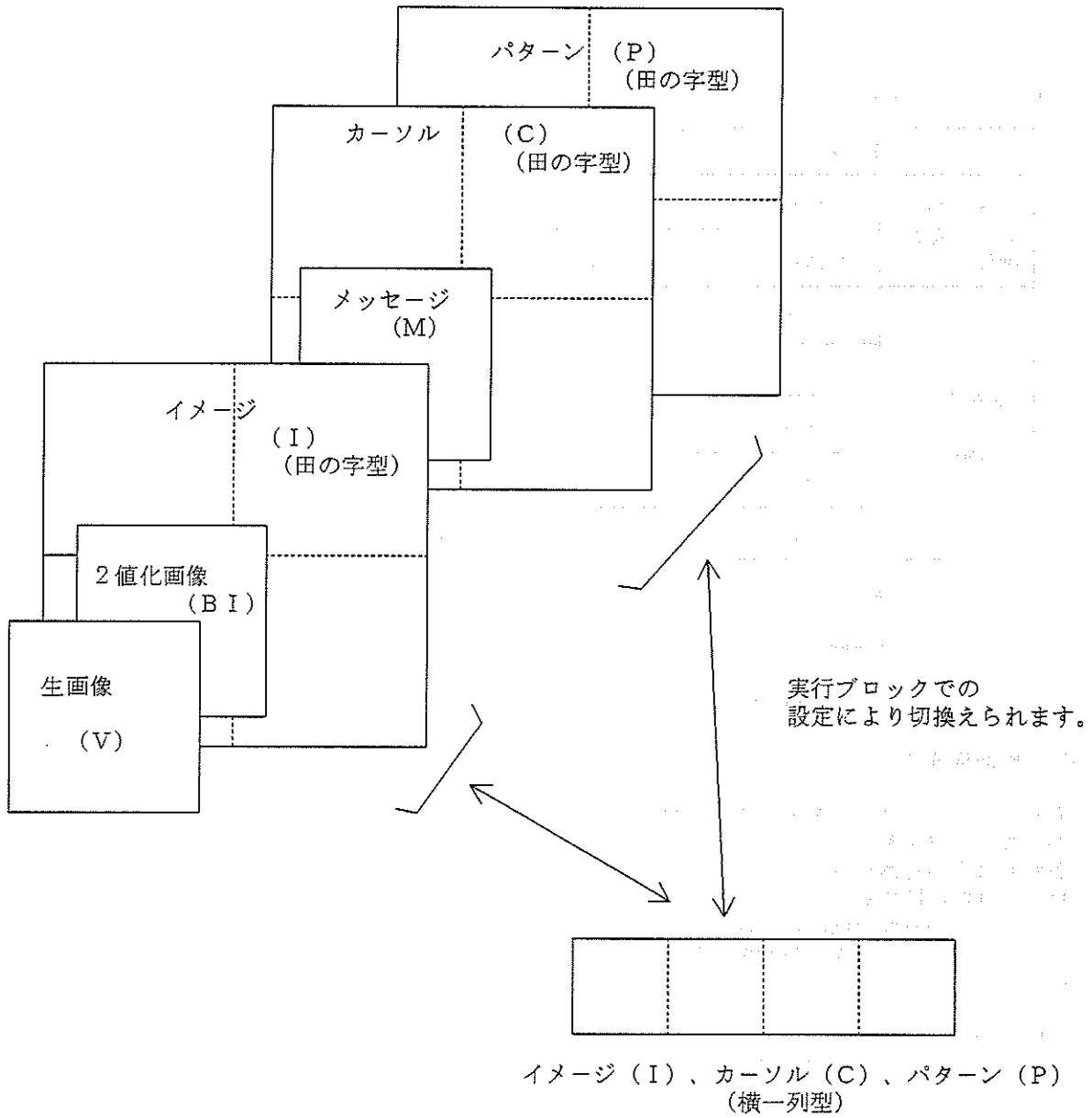
- a. *印はユーザーで描き込み可能な画面です。
- b. 生画像はカメラから直接取り込み、メモリとは関係なく表示されますので、表示を移動させることは出来ません。

(参考) 画面構成について

Pタイプのプログラムモードの' SHOW' コマンドで制御可能な画面は全部で6画面あります。(図参照)
 各画面は表示を選択されると、重ね書きされて画面表示します。
 但し、画像に関する 生画像 (V)、リアルタイム2値化画像 (BI)、イメージ (I) は2つ以上の画像を同時に画面表示することはできません。
 画像指定の優先順位は、生画像 (V)、リアルタイム2値化画像 (BI)、イメージ (I) の順です。

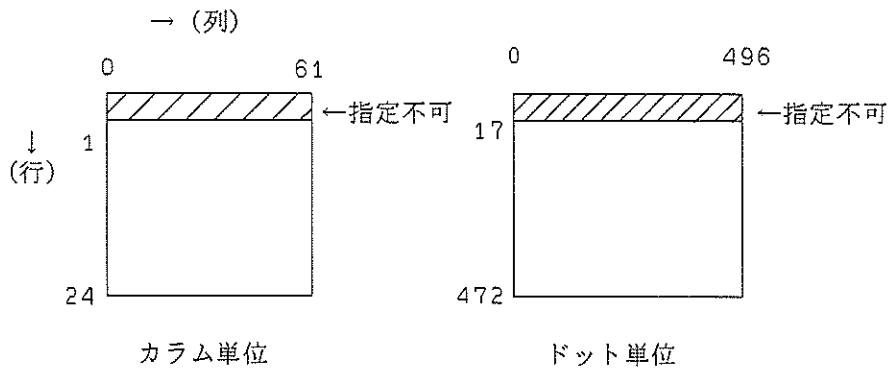
各画面の簡単な説明を下表に示します。

画面	' SHOW' での記号	説明	' MONT' による指定範囲
生画像	V	カメラから取込んだ画像をそのまま表示します。	A~D
リアルタイム2値化画像	BI	カメラから取込んだ画像を2値化し、リアルタイムで表示します。	A~D
イメージ	I	メモリに取込まれた、2値化画像を表示します。	田の字型 A~I 横一列型 A~G
メッセージ	M	" DISP", " DISPD", " MLON" コマンドより、設定された「文字・線」を表示します。	指定不可
カーソル	C	" CURSOR" コマンドにより設定された「カーソル」を表示します。	田の字型 A~I 横一列型 A~G
パターン	P	位置補正チェッカ、ウィンドウチェッカ等、チェッカを表示します。	



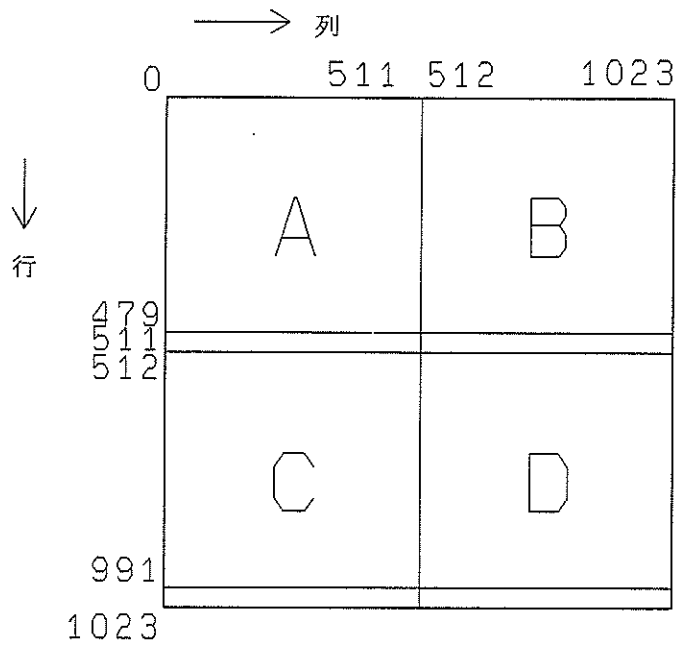
プログラムで制御可能な画面

(2) 表示範囲
メッセージプレーン

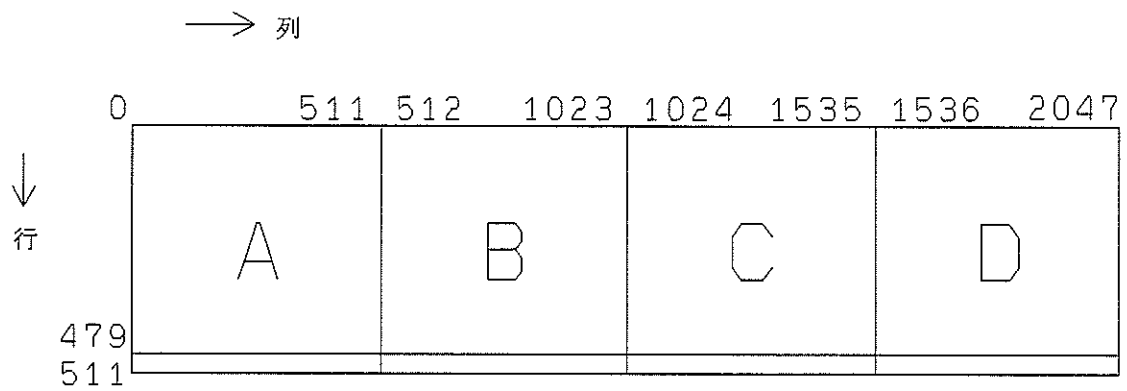


カーソル … 「品種モード」で画面構成の設定により表示範囲が変わります。

田の字型



横一列型



[2] ポート構成

(1) パラレルポート

a. 入力ポート … 各8ビットで、NO. 1～3まで用意されています。

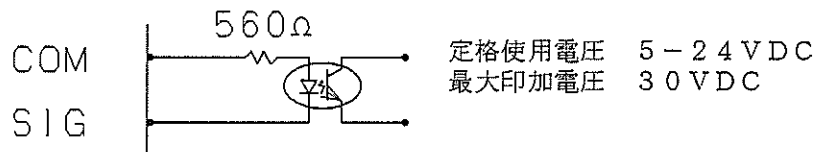
ポートNO.	7	6	5	4	3	2	1	ビット0
1								
	品種フラグ	位置補正フラグ	ソフトリセット	ACK	露出補正フラグ		電源リセット注1	スタート
2								
	← 品種切り替え番号 →							
3								
	品種復帰フラグ							

(下段はイメージチェッカ30・30Rでの使用内容)

・「P-IN」命令により実行します。
この時、Vレジスタに格納する内容を下表に示します。

入力ポート状態	レジスタのビット内容
ON	0
OFF	1

- ・イメージチェッカ30・30Rとは独立した動作をします。
- ・回路構成



b. 出力ポート … 各8ビットで、NO. 2～3まで用意されています。

ポートNO.	7	6	5	4	3	2	1	ビット0
1								
	BATRY ERROR	READY	STROB	REND	OVFLG	露出検出注1	フラッシュ同期番号	
2								
	D8	D7	D6	D5	D4	D3	D2	D1
3								
	← 予備 →							

← P-OUTにて指定できません

(下段はイメージチェッカ30・30Rでの使用内容)

注1: Ver 3以降での機能です。
「初期化ブロック」の「8. 瞬時停電検出時の復帰する条件」で「2. 復帰する」を設定時での内容です。

・「P-OUT」命令により実行されます。

この時、Vレジスタの内容がポートに出力される内容を下表に示します。

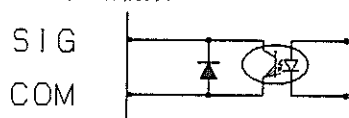
Vレジスタのビット内容	出力ポート状態
1	ON
0	OFF

・イメージチェッカ30・30Rとは独立した動作をします。

・ポートNO. 1 (//) は” P-OUT” コマンドで使用できません。

・注意：瞬時停電検出時に復帰させたい場合、入力ポートNo. 1のビット1と出力ポートNo. 1のビット1は、プログラム中で使用しないでください。
瞬時停電より復帰した場合に、電源ONと同じ動作をします。

・回路構成



定格使用電圧 5-24VDC
最大許可範囲 5-30VDC
最大負荷電流 30mA

(2) シリアルポート

仕様はイメージチェッカ30、30Rに準拠します。

形状	ピン	I/O	信号名	ピン	I/O	信号名	ピン	I/O	信号名
	1	-	FG	10	-	-	19	-	-
	2	O	TXD	11	-	-	20	O	DTR
	3	I	RXD	12	-	-	21	-	-
	4	O	RTS	13	-	-	22	-	-
	5	I	CTS	14	-	-	23	-	-
	6	I	DSR	15	-	-	24	-	-
	7	-	SG	16	-	-	25	-	-
	8	I	CD	17	-	-			
	9	-	-	18	-	-			

図は、RS232C本体側コネクタのピン配置です。

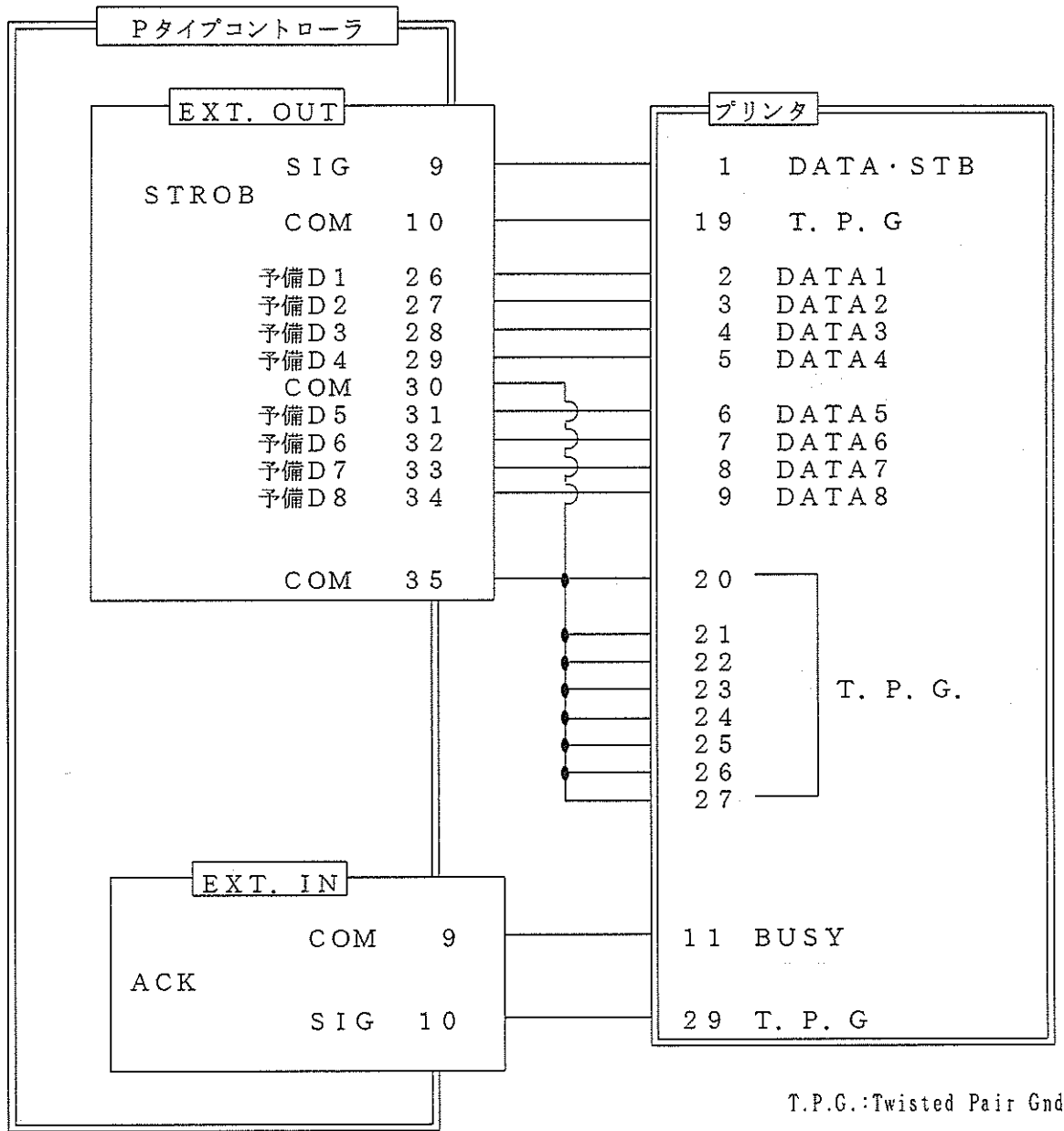
注意

”SCAN”または”*SCAN”を実行しますと、イメージチェッカ30・30Rで、実行する場合よりイメージチェッカ30P・30RPでは、1回実行する毎に、約7msecがかかります。

[3] プリンタとの接続

(1) セントロニクスプリンタ

プリンタとの接続は、「セントロニクス仕様」に準拠しています。図4-6-3に示す結線で接続を行ってください。イメージチェッカ30P・30RPと接続できるプリンタは”NEC:PC-PR201、EPSON:V P-2000、ブラザー:M-1924P”同等品です。



尚、プリントアウトに時間を要する場合は、プリンタバッファを使用することをお勧めいたします。

ケーブルは、ユーザ様にて自作願います。

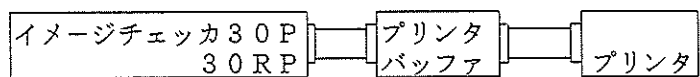


図4-6-3のケーブル

プリンタバッファ用ケーブル

<注意>

セントロニクス仕様のプリンタでしかプログラムリストのプリントアウトはできません。シリアル (RS232C) 仕様のプリンタでは、プログラムリストのプリントアウトはできません。

(2) シリアルプリンタ

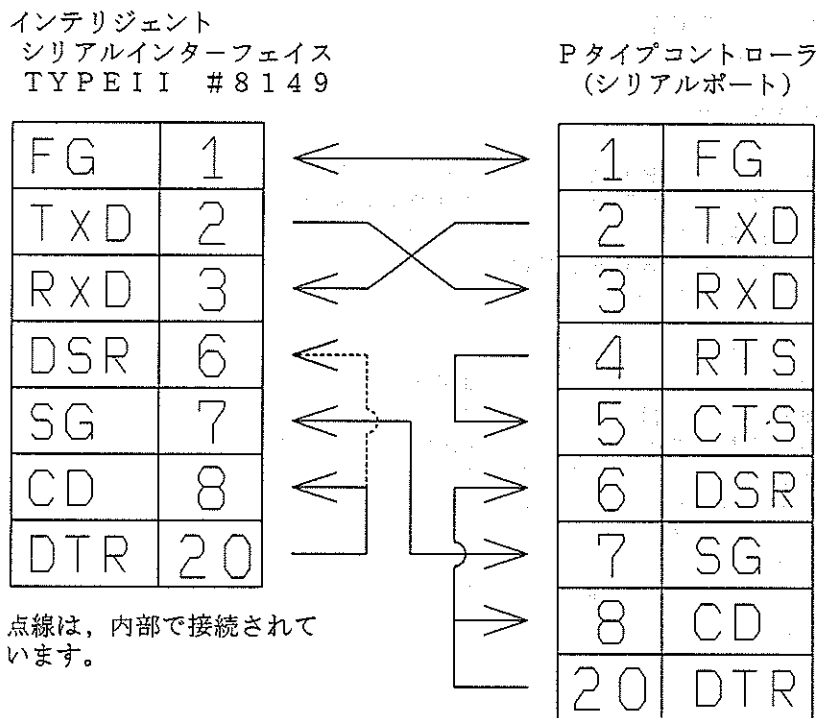
イメージチェッカ30P・30RPコントローラのシリアルポートから直接シリアルプリンタにプリントアウトをする場合は、シリアルプリンタの特性により、プリンタ毎に結線が異なりますので、プリンタケーブルを作成する場合は、プリンタのマニュアルを参考にして作成をしてください。

EPSON社から、シリアルインターフェイスボードがオプションで発売されています。ここでは、同社のインテリジェントシリアルインターフェイス type #8149との接続例を図4-6-4に示しますので参考にしてください。

注意：イメージチェッカ30P・30RPは、X-ON/X-OFFによる制御をおこないますので、プリンタの設定に注意してください。

また、I/Oの設定値（ボーレート、キャラクタ長、パリティビット、ストップビット）をプリンタとPタイプコントローラで合わせてください。

図4-6-4



(3) 漢字キャラクタの制御コードについて

イメージチェッカ30P・30RPで”PRINT”、”S-PRINT”コマンドを利用して、漢字キャラクタをプリンタ出力する場合、以下の制御コードで、制御を行います。接続するプリンタが、以下の制御コードになっていることを確認願います。

漢字in : ”ESC” + ”K”
漢字out : ”ESC” + ”H”

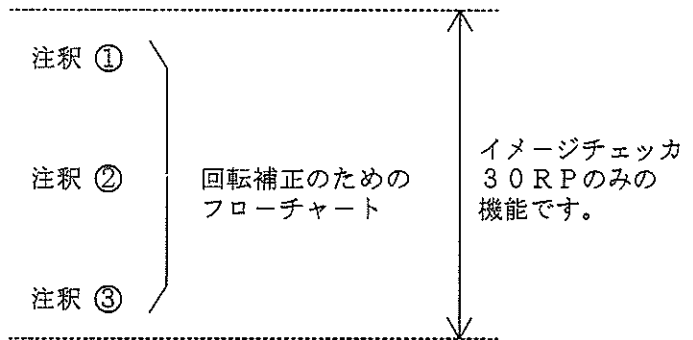
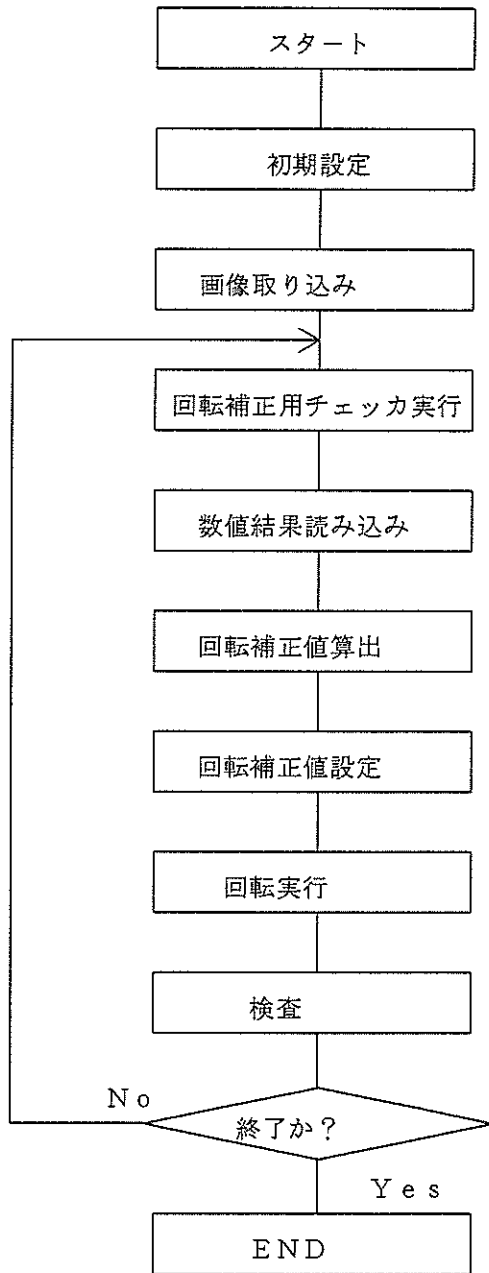
<注意>

イメージチェッカ30P・30RPのコントローラがサポートしていない文字、グラフィクス文字を出力しますと、コントローラの不具合の原因となりますので、プリントアウト出力は、行わないでください。漢字は、JIS第一水準をサポートしています。巻末付録を参考してください。

プログラムモードで回転補正を実行する際の注意：30RPのみ

- (1) プログラム中で回転補正を実行しようとする場合、回転補正用設定値を設定するためにはメモリエリアの確保が必要です。メモリエリアは「9. 回転補正」の「1. プログラム入力」で少なくとも「画面外エリア色」の設定を行うことにより確保されますので、忘れずに設定してください。
また、プログラム実行中、回転補正のモード設定「三角比モード」か「主軸角モード」かを変更することが可能です。
これは”TPRIN”または”TWIDTH” コマンドを実行することで行えます。
尚、”TSCAN” コマンドにより回転を実行する際、最後に設定されたモードにより回転を行いますのでご注意ください。
プログラム実行中回転補正のモードを変更したとしても品種モードでは回転補正を設定したときのモードで動作します。
- (2) 回転補正用設定値は、設定ブロックでの「9. 回転補正」の「1. プログラム入力」で設定された設定値を読み込んで、回転補正を実行します。
(イメージチェッカ30Rマニュアル参照)
プログラム中で回転補正用設定値をコマンド (TWIDTH、TANG、TPRIN、TCENT、TSHIFT) により設定されたときは、その設定値で回転を実行します。
- (3) ”SCAN” コマンドを実行すると、30Rと同様の動作をします。
- (4) ”READ”、”READF”、”ESCAN” コマンドによる画像取り込み後、”TSCAN” コマンドでの画像回転を回転補正用設定を変更して、2回以上連続して行う場合にご注意ください。
イメージチェッカ30RPでは画像メモリが検査実行用と回転補正用の2メモリが用意されています。モニタに表示されているのは検査実行用のメモリです。
”TSCAN” コマンドで回転を実行する場合、回転補正用画像メモリからデータを読み込んで回転を行い検査用画像メモリに書き込みます。この時、回転補正用メモリは保存されています。
”TSCAN” で回転実行後、回転補正用チェッカを実行し回転補正設定値を設定して再度、”TSCAN” を実行した場合、回転は検査実行用画像メモリのデータを使用せずに回転補正用画像メモリのデータを読み込んで行われるため、モニタに表示されている画像とは違う回転を実行しますのでご注意ください。
このような場合は、再度画像の読み込みを行うか回転補正用チェッカを最初に実行しておいてください。

イメージチェッカ30P・30RPのプログラムのフローチャートの一例



- 注釈 ① : 回転角 (Tθ)、水平・垂直差分 (WX、WY)、移動量 (SW、SY) の算出
- 注釈 ② : TANG、TWIDTH、TCENT、TSHIFT、TPRINコマンドによる設定
- 注釈 ③ : TSCANコマンドによる実行

イメージチェッカ30P・30RPコマンド

イメージチェッカ30P・30RP型命令一覧

分類	機能	コマンド	概略
プログラム 制御	ジャンプ命令	GOTO IF TBIT IFSTR IFNSTR WHILE ▷ WHILER	無条件ジャンプ 条件付ジャンプ (データによる判断) 条件付ジャンプ (ビットによる判断) 文字列の比較 (同じ場合ジャンプ) 文字列の比較 (違う場合ジャンプ) 区間内の繰り返し実行 (~GOTO) 区間内の繰り返し実行 (~GOTO)
	サブルーチンコール	GOSUB	サブルーチンコール
	ファイルNO.コール	CALL	ファイルNO. コール
	復帰命令	RET	CALL, GOSUBからの復帰
	遅延命令	DELAY	プログラムの遅延
数値演算	レジスタ制御	MOVE CODE SBIT RBIT CBIT LEN NOT MOVEWB MOVELB MOVELW MOVEBW MOVEBL MOVEWL CLRREG	レジスタ間転送 数値データと文字コードの変換 指定レジスタの指定ビットをセットする 指定レジスタの指定ビットをリセット (クリア) する 判定結果を指定の数値レジスタビットに反映する 数値レジスタの符合、桁数の読み込み 全ビットの反転 ワード→バイト 分割コマンド ロングワード→バイト 分割コマンド ロングワード→ワード 分割コマンド バイト→ワード 結合コマンド バイト→ロングワード 結合コマンド ワード→ロングワード 結合コマンド レジスタをクリアする
	算術演算	ABS ADD SUB MUL DIV	レジスタ絶対値 レジスタ間加算 レジスタ間減算 レジスタ間乗算 レジスタ間除算
	論理演算	AND OR XOR	レジスタ間論理積演算 レジスタ間論理和演算 レジスタ間排他的論理和演算
	三角関数	SIN COS TAN ATAN	三角関数 SIN 演算 三角関数 COS 演算 三角関数 TAN 演算 三角関数 TAN-1 演算
	平方根	SQRT	平方根の計算
	タイマー	TMWT TMRD	内部タイマーに初期値をセットする 内部タイマーの値を指定レジスタに読み込む
入出力制御	画面制御	DISP DISPD	イメージカウにキャラクタ単位制御の文字列を表示 イメージカウにドット単位制御の文字列を表示

▷ : Ver 3以降対応機能

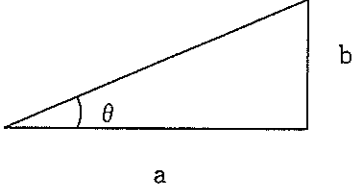
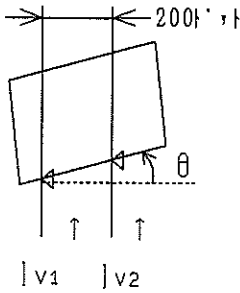
分類	機能	コマンド	概略
入出力制御	画面制御	CLRCRT SHOW MONT MLON MLOFF MLCLR CURSOR LOCATE LOCATD	メッセージプレーン上の全ての表示を消去する モニターの表示選択 モニターの表示エリアの選択 メッセージプレーンへのライン/ボックス描画 メッセージプレーンへのライン/ボックス消去 メッセージプレーンの部分消去 十字カーソルを表示する モニター上の文字表示の開始位置指定 (カラム単位) モニター上の文字表示の開始位置指定 (ドット単位)
	I/Oポート入出力制御	C-OUT D-OUT S-IN S-OUT P-IN P-OUT READY STROB REND OVOFF SIOBR SIOCL SIOPB SIOSB START	数値演算結果の平行出力 (C470~C512) 判定出力の平行出力 シリアル入力 シリアル出力 平行入力 平行出力 平行出力レディ信号の制御 平行出力ストロブ信号の制御 平行出力画像取り込み完了信号の制御 オーバーフローフラグのOFF ボーレイトの設定 キャラクタ長の設定 パリティビットの設定 ストップビットの設定 スタート・品種切換信号の読み込みと実行
	キーボード制御	KEYIN KEYSNS	キーボードからの文字入力 キーボードのセンス (キー入力がなくともステップは停止しない)
	プリンタ制御	PRINT SPRINT	プリンタへの出力 (平行) プリンタへの出力 (シリアル)
	カメラ制御	READ READF	メモリに画像取り込み メモリに画像取り込み (フラッシュ信号付)
画像処理条件設定		THRES TYPE ▶TANG ▶TWIDTH ▶TCENT ▶TSHIFT ▶TPRIN	2値化レベルの設定 品種切り替え 回転角 θ の設定 水平、垂直差分の設定 回転中心座標設定 水平、垂直移動量の設定 主軸角 ($P\theta$) の設定
画像処理実行		SCAN ISCAN LSCAN WSCAN MSCAN FSCAN ESCAN CSCAN RSCAN DSCAN ▶TSCAN	全チェックの実行 位置補正チェックの実行 ラインチェックの実行 ウインドウチェックの実行 パターンマッチングの実行 特徴抽出チェックの実行 露出補正チェックの実行 数値演算チェックの実行 判定出力チェックの実行 (Rタイプ) 判定出力チェックの実行 (Dタイプ) 画像回転と回転補正チェックの実行

▶: 30RPのみの機能です。

ABS		アブソリュート
機能	絶対値の算出	
書式	ABS Δ $\begin{pmatrix} i \\ v \end{pmatrix}, v$	
角解 説	<p>第1オペランドで指定された数値の絶対値を計算し、第2オペランドで指定されたvレジスタに格納します。</p> <p><文例></p> <pre> MOVE -2568, V10 ; -2568 → V10 ABS V10, V20 ; V10 = -2568 =2568 → V20 </pre> <p><文例></p> <pre> MOVE -2568, V10 ; -2568 → V10 ABS V10, V10 ; V10 = -2568 =2568 → V10 </pre>	

ADD		アッド
機能	レジスタ間加算	
書式	<p>1) 数値レジスタ間加算 ADD Δ $\begin{pmatrix} v \\ i \\ h \end{pmatrix}, \begin{pmatrix} v \\ i \\ h \end{pmatrix}, v$</p> <p>2) 文字レジスタ間加算 ADD Δ $\begin{pmatrix} a \\ G \end{pmatrix}, i, a$</p>	
角解 言説	<p>1) 数値レジスタ間加算 第1オペランドと第2オペランドの加算を行い、第3オペランドの指定するvレジスタに格納します。</p> <p>加算結果の値の範囲は、-2,147,483,648~2,147,483,647 です。 オーバーフローした時はエラーになり、オーバーフローフラグがONされます。 この時、第3オペランドの値は意味をなさない値となりますので、ご注意ください。</p> <p><文例> ADD V2, 200, V30 ; V2+200→V30 ADD V2, V50, V550 ; V2+V50→V550</p> <p><文例> MOVE 2, V9 ; 2→V9 MOVE 2568, V10 ; 2568→V10 ADD V9, V10, V9 ; V9+V10=2+2568=2570→V9</p> <p>2) 文字レジスタ間加算 aレジスタに格納されている文字のアスキーコードに数値を加算したアスキーコードを、第3オペランドで指定したaレジスタに格納します。</p> <p><文例> MOVE 'A', A1, 1 ; 'A'(41H)→A1 ADD A1, 1, A5 ; A1+1=(41H)+1=42H 42H('B')→A5</p> <p><注意> ADD "A", 1, A5 の様な文字列(" ")の指定はできません。</p>	

AND		アンド															
機能	論理積																
書式	$\text{AND } \Delta \left(\begin{array}{c} v \\ i \\ h \end{array} \right), \left(\begin{array}{c} v \\ i \\ h \end{array} \right), v$																
角平 言兌	<p>第1オペランドの内容と第2オペランドの内容との各ビット毎の論理積を行い、演算結果を第3オペランドで指定したvレジスタに格納します。</p> <p>論理積の演算を下表に示します。</p> <table border="1"> <thead> <tr> <th>第1オペランドの内容</th> <th>第2オペランドの内容</th> <th>第3オペランドに格納される内容</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table> <p><文例> AND V1, V2, V3 ; V1 and V2 → V3</p> <p><注意> AND : 論理積は、各レジスタの各ビット間の論理積となります。</p>		第1オペランドの内容	第2オペランドの内容	第3オペランドに格納される内容	0	0	0	0	1	0	1	0	0	1	1	1
第1オペランドの内容	第2オペランドの内容	第3オペランドに格納される内容															
0	0	0															
0	1	0															
1	0	0															
1	1	1															

A T A N	アークタンジェント
機 能	逆正接の計算
書 式	$\text{ATAN} \quad \Delta \quad \left[\begin{array}{c} v \\ i \end{array} \right], v$
角 度 説	<p>第1オペランドの内容の逆正接計算を行い、結果を第2オペランドで指定したvレジスタに格納します。 第1オペランドには、辺比 (b/a) × 10000を入力します。 計算結果は角度 (θ) × 100が第2オペランドで指定したvレジスタに格納されます。</p> <p><文例> <code>ATAN 10000, V1 ; tan⁻¹ 1.0の計算を行い</code> V1に結果を格納します。 <code>tan⁻¹ 1.0 = 45度</code> <code>45 × 100 = 4500 → V1</code></p>  $100 \theta = \text{ATAN} \left(\frac{b}{a} \times 10000 \right) \quad , \quad 100 \theta = \tan^{-1} \left(\frac{b}{a} \times 10000 \right)$ <p>角度検出例</p>  <p>注意 イメージチェッカ30RPは垂直方向、水平方向の分解能は、同一です。</p> <p>30RP</p> <pre> MOVE CI12, V1 ; I_{v1} → V1 MOVE CI22, V2 ; I_{v2} → V2 MOVE 200, V3 ; 200 → V3 SUB V1, V2, V4 ; V1 - V2 → V4 MUL V4, 10000, V5; V4 × 10000 → V5 DIV V5, V3, V6 ; V5 ÷ V3 → V6 ATAN V6, V7 ; (tan⁻¹ v6) × 10000 </pre> <p>30P イメージチェッカ30Pは垂直方向、水平方向の分解能が異なります。そのため、左図の様な傾き角度を求める場合は分解能を補正して求める必要があります。</p> <p>垂直方向分解能 = 100 μm/ドット 水平方向分解能 = 112 μm/ドット</p> <pre> MOVE CI12, V1 ; I_{v1} → V1 MOVE CI22, V2 ; I_{v2} → V2 MOVE 200, V3 ; 200 → V3 SUB V1, V2, V4 ; V1 - V2 → V4 MUL V3, 112, V5; V3 × 112 → V5 MUL V4, 100, V6; V4 × 100 → V6 MUL V6, 10000, V7 DIV V7, V5, V8 ATAN V8, V9 </pre> <p>垂直方向位置補正チェッカ No. 1 → I_{v1} No. 2 → I_{v2}</p>

CALL (～RET)	コール (～レット)
機能	ファイルNO. コール
書式	CALL Δ f
角解 言説	<p>ファイルNO. を指定することにより、他のファイルNO. をコールし、“RET”により復帰します。</p> <p>ファイルNO. は、コントローラ内部メモリの場合、PM1～PM99、ICカードの場合、PI1～PI99をコールすることができます。</p> <p><文例> CALL Δ PM9 ;ファイルNO. ”PM9” をコールし、“RET”により復帰します。</p> <p><注意> ネスティングは、“CALL”のみで合計32まで、ネスティングすることができます。</p> <pre> graph TD PM1[PM1] -- CALL PM2 --> PM2[PM2] PM2 -- CALL PM3 --> PM3[PM3] PM3 -- CALL PM4 --> PM4[PM4] PM4 -- RET --> PM3 PM3 -- RET --> PM2 PM2 -- RET --> PM1 </pre>

C B I T		コピービット
機能	判定結果を指定レジスタに反映	
書式	$CBIT \quad \Delta \quad \left[\begin{array}{c} v \\ S \end{array} \right], v, \left[\begin{array}{c} v \\ i \end{array} \right]$	
角解 言説	<p>第1オペランドは判定結果（0または1）が格納されたvレジスタを指定し、第2オペランドは判定結果をコピーする「vレジスタ」を、第3オペランドは判定結果をコピーする「ビット」をそれぞれ指定します。第1オペランドの値は”0”または”1”で、第3オペランドの値は、0～31です。それ以外の場合は、エラーとなり、第2オペランドの値は変化せずにオーバーフローフラグをONします。</p> <p>(参考) vレジスタの構成は、32ビットです。 (LSB=0ビット MSB=31ビット)</p> <p><文例> MOVE DW1, V1 ; ウィンドウNo.1の判定結果をV1に格納する CBIT V1, V10, 1 ; V1に格納された判定結果を、V10レジスタの1ビットにコピーする。</p> <p>CBIT DS3, V1, 1 ; S3の内容をV1の1ビットにコピーする。</p> <p><注意> 第3オペランドは、省略できません。</p> <p>このコマンドは、主に出力命令で、D-OUTを使用せずにP-OUTで任意のポートに出力したい場合に使用します。</p>	

CLRCRT		クリアシーアールティ
機 能	画面消去 (メッセージプレーン)	
書 式	CLRCRT	
角 説	<p>メッセージプレーンに書かれた文字の消去と、「MLON」により書かれたラインを消去します。</p> <p>注) 一部だけの消去を行う場合は「" "」(空白文字)を書き込んでください。</p> <p>例. DISP 10, 12, " ", 10; 12行10列から10文字分を消去します。 (空白文字を10文字書きます。)</p> <p>(参考) 1) 画面消去は、1行目(0行)を残して2行目(1行)から実行します。</p> <div style="text-align: center;"> <p>→列</p> <p>↓行</p> <p>←消去エリア</p> </div> <p>2) このコマンドは実行時間がかかるため(約190ms)、システムの立ち上げの時以外は実行しない方が望ましいと思われます。文字列の消去は、「" "」(空白)を上書きすることにより消去し、「MLON」によりメッセージプレーン描画されたラインは、「MLOFF」により消去を行ってください。</p> <p><文例> CLRCRT ; 画面消去を実行します。</p>	

CLRREG		クリアレジスタ
機能	レジスタのクリア	
書式	1) 数値レジスタ CLRREG Δ v, v 2) 文字レジスタ CLRREG Δ a, a	
角解 説	1) 数値レジスタ (vレジスタ) のクリア 第1オペランドで指定したvレジスタから、第2オペラ ンドで指定したvレジスタをクリア (=0) します。 <文例> CLRREG V1, V10 ; V1~V10 ← "0" 2) 文字レジスタ (aレジスタ) のクリア 第1オペランドで指定したaレジスタから、第2オペラ ンドで指定したaレジスタをクリア (0X00→) します。 <文例> CLRREG A1, A10 ; A1~A10 ← "0X00" (NUL)	

CODE		コード
機能	数値データと文字コードの変換 (アスキーコード)	
書式	<p>1) 数値レジスタ → 文字レジスタ (文字コード) CODE Δ v, a</p> <p>2) 数 値 → 文字レジスタ (文字コード) CODE Δ [h], a i</p> <p>3) 文字レジスタ → 数値レジスタ (文字コード) CODE Δ a, v</p> <p>4) 文 字 → 数値レジスタ (文字コード) CODE Δ G, v</p>	
角 注	<p>1) 数値レジスタ → 文字レジスタ (文字コード) 第1オペランドでは、アスキーコードを10進数表現で格納した vレジスタを指定し、第2オペランドで指定されたaレジスタに 格納します。</p> <p><文例> MOVE 0X25, V1 ; 0X25(hex)→37 (10進数) 37→V1</p> <p>CODE V1, A1 ; V1の内容で示されるアスキーコードの文字 をA1に格納します。アスキー (25 hまたは 37₍₁₀₎) → % ' %' → A1</p> <p>2) 数 値 → 文字レジスタ (文字コード) 第1オペランドでは、アスキーコードを指定し、第2オペラン ドで指定したaレジスタにその内容を格納します。</p> <p><文例> CODE 0X3F, A1 ; '??' の0X3F(?)をA1に格納します (?) ' ?' → A1</p>	

角 平 言 兌

3) 文字レジスタ → 数値レジスタ

(文字コード)

第1オペランドで指定したaレジスタの内容を、第2オペランドで指定したvレジスタに16進数のアスキーコードを10進数に変換した値で格納します。

<文例>

```
MOVE ' #' ,A1,1 ; ' #' をA1に格納します。
CODE A1, V1 ; A1の内容のアスキーコード
           ' #' =0X23を10進数に変換し
           0X23→35をV1に格納します。
```

4) 文字 → 数値レジスタ (文字コード)

第1オペランドで指定した文字を、第2オペランドで指定したvレジスタに16進数のアスキーコードを10進数に変換した値で格納します。

<文例>

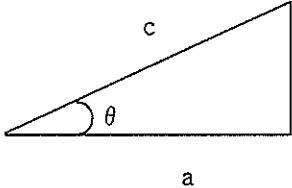
```
CODE ' &' ,V1 ; ' &' のアスキーコードである
               0X26(hex)を10進数表現して、
               0X26→38をV1に格納します。
```

(注意)

文字→文字レジスタの場合は、"MOVE" コマンドを用いてください。

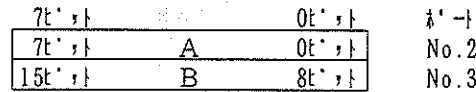
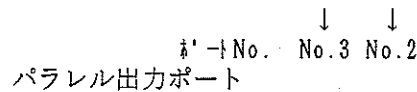
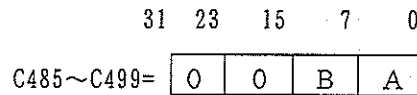
<文例>

```
MOVE ' &' ,A1,1 ; ' &' →A1
```

COS		コサイン
機能	余弦の計算	
書式	$\text{COS } \Delta \left[\begin{matrix} v \\ i \end{matrix} \right], v$	
角平 言兌	<p>第1オペランドの内容の余弦計算を行い、結果を第2オペランドで指定したvレジスタに格納します。</p> <p>第1オペランドには、角度(θ度)×100を入力します。計算結果は、(×10000)で第2オペランドで指定したvレジスタに格納されます。</p> <p>オーバーフローした時はエラーとなります。</p> <p><文例> COS 6000, V1 ; COS60度の計算を行い、V1に結果を格納します。 COS60度=0.5000 0.5×10000=5000 → V1</p> 	

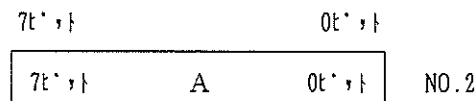
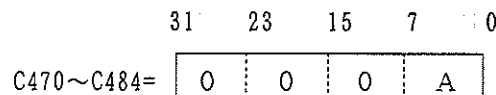
C-OUT		シーアウト																																																						
機能	数値演算結果の平行アウト																																																							
書式	C-OUT Δ $\left[\begin{array}{c} (v) \\ (i) \end{array} \right]$ ()は省略可																																																							
角解 説	<p>第1オペランド・・・コマンド実行時に”CSCAN 1、512”を実行するかどうかを選択します。</p> <table border="1"> <thead> <tr> <th>番号</th> <th>平行アウト</th> </tr> </thead> <tbody> <tr> <td>(省略), 0</td> <td>実行する</td> </tr> <tr> <td>1</td> <td>実行しない</td> </tr> </tbody> </table> <p><文例> C-OUT 1; ”CSCAN 1、512”を実行せずに数値演算結果を平行出力します。</p> <p>CSCANを実行した後 品種モードの設定ブロックで設定された、数値演算の数値結果を全て、平行出力します。「平行ハンドシェイクを行う一行わない」の設定は無視され平行ハンドシェイクを行います。</p> <p>1) データ出力の方法 データ出力にはCレジスタの設定により、3通りの方法がありますので、平行ハンドシェイクしながら外部機器にてデータを読み込んでください。</p> <p>イ) C500～C512を設定した場合 (32ビットデータの場合) C500から順番に、32ビットのデータをLSBから8ビットずつ4バイトに分割し、下記のポートNo.2、No.3から、16ビットずつ平行出力されます。</p> <table style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td style="text-align: center;">31</td> <td style="text-align: center;">23</td> <td style="text-align: center;">15</td> <td style="text-align: center;">7</td> <td style="text-align: center;">0</td> </tr> <tr> <td>C500～C512=</td> <td colspan="4" style="text-align: center;"> <table border="1" style="display: inline-table;"> <tr> <td style="width: 25px; height: 20px;">D</td> <td style="width: 25px; height: 20px;">C</td> <td style="width: 25px; height: 20px;">B</td> <td style="width: 25px; height: 20px;">A</td> </tr> </table> </td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">↓</td> <td style="text-align: center;">↓</td> <td style="text-align: center;">↓</td> <td style="text-align: center;">↓</td> <td></td> </tr> <tr> <td>※→No.</td> <td style="text-align: center;">No.3</td> <td style="text-align: center;">No.2</td> <td style="text-align: center;">No.3</td> <td style="text-align: center;">No.2</td> <td></td> </tr> </table> <p>平行出力ポート</p> <table style="margin-left: auto; margin-right: auto;"> <tr> <td>出力順番</td> <td style="text-align: center;">7ビット</td> <td style="text-align: center;">0ビット</td> <td>※→</td> </tr> <tr> <td>1</td> <td style="text-align: center;">7ビット</td> <td style="text-align: center;">A</td> <td style="text-align: center;">0ビット</td> </tr> <tr> <td>2</td> <td style="text-align: center;">23ビット</td> <td style="text-align: center;">C</td> <td style="text-align: center;">16ビット</td> </tr> <tr> <td>1</td> <td style="text-align: center;">15ビット</td> <td style="text-align: center;">B</td> <td style="text-align: center;">8ビット</td> </tr> <tr> <td>2</td> <td style="text-align: center;">31ビット</td> <td style="text-align: center;">D</td> <td style="text-align: center;">24ビット</td> </tr> </table> <p>まずA、Bが平行出力され、次にC、Dが平行出力されます。すなわち1つのCレジスタで16ビットのデータを2回平行出力することになります。</p>		番号	平行アウト	(省略), 0	実行する	1	実行しない		31	23	15	7	0	C500～C512=	<table border="1" style="display: inline-table;"> <tr> <td style="width: 25px; height: 20px;">D</td> <td style="width: 25px; height: 20px;">C</td> <td style="width: 25px; height: 20px;">B</td> <td style="width: 25px; height: 20px;">A</td> </tr> </table>				D	C	B	A			↓	↓	↓	↓		※→No.	No.3	No.2	No.3	No.2		出力順番	7ビット	0ビット	※→	1	7ビット	A	0ビット	2	23ビット	C	16ビット	1	15ビット	B	8ビット	2	31ビット	D	24ビット
番号	平行アウト																																																							
(省略), 0	実行する																																																							
1	実行しない																																																							
	31	23	15	7	0																																																			
C500～C512=	<table border="1" style="display: inline-table;"> <tr> <td style="width: 25px; height: 20px;">D</td> <td style="width: 25px; height: 20px;">C</td> <td style="width: 25px; height: 20px;">B</td> <td style="width: 25px; height: 20px;">A</td> </tr> </table>				D	C	B	A																																																
D	C	B	A																																																					
	↓	↓	↓	↓																																																				
※→No.	No.3	No.2	No.3	No.2																																																				
出力順番	7ビット	0ビット	※→																																																					
1	7ビット	A	0ビット																																																					
2	23ビット	C	16ビット																																																					
1	15ビット	B	8ビット																																																					
2	31ビット	D	24ビット																																																					

- ロ) C485~C499を設定した場合 (16ビットデータの場合)
 C485から順番に、16ビットのデータをLSBから8ビットずつ2バイトに分割し、下記のポートNo.2、No.3から、16ビットで平行出力されます。



A、Bのデータすなわち、16ビットのデータを1回平行出力することになります。

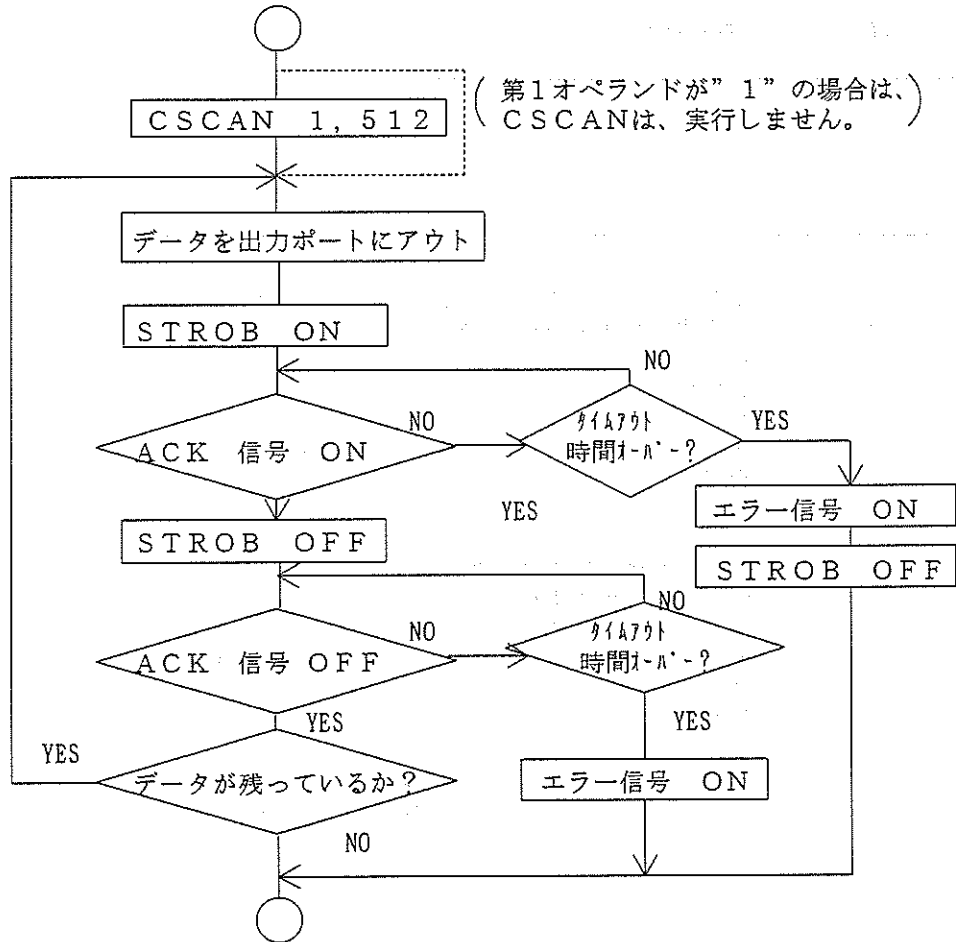
- ハ) C470~C484を設定した場合 (8ビットデータの場合)
 C470から順番に、LSBから8ビットのデータをポートNO.2から8ビットで平行出力します。



- 注) ・ ロ)、ハ) のとき、データが、8ビット又は16ビット以上になったり、負の値となるときは、エラーとなり、オーバーフローエラー信号をONします。このとき、データも平行出力しますので、外部機器にてこのデータを無視してください。 次のデータ出力の際エラーとならなければ、オーバーフローエラー信号をOFFにします。
 ・ C470~512を設定した場合、番号の若い順にイ)~ハ) の出力形式に応じて出力されます。(未設定の場合は、無視され前づめされます。)

2) データ出力の流れ

以下に動作の流れを、フローチャートに示します。



注) ・ ACK信号 ON-OFFの「タイムアウト時間」の設定値は「初期化ブロック-パラレルハンドシェイク-行う」で設定された設定値によります。初期化ブロックでのパラレルハンドシェイクを「行う-行なわない」の選択は無視されますが、忘れずに、「タイムアウト時間」を設定してください。

・ タイムアウトエラー発生時、エラー信号がONになりますがエラー信号OFFされるタイミングは” START” コマンドのスタート信号・品種切替信号入力時 ” READ”、” READF”、” ESCAN” 実行時です。

3) 結線方法

「イメージチェッカ30・30R導入マニュアル」の「8-1パラレル信号による通信」を参照してください。

注) 「C-OUT」を実行するときはパラレルハンドシェイクを行ないますので、ご注意ください。

<注意> プログラム上では、すべてのデータの出力が終了するまでかタイムアウトになるまで、このステップ上での処理を行います。

C S C A N		シー ス キ ャ ン
機 能	数値演算の実行	
書 式	CSCAN Δ $\begin{bmatrix} v \\ i \end{bmatrix}, \begin{bmatrix} v \\ i \end{bmatrix}$	
角 解 説	<p>第1オペランドで指定する数値演算No.から、第2オペランドで指定する数値演算No.までの演算を実行します。</p> <p>指定する数値演算のプログラムは、あらかじめ「数値演算」で設定しておいてください。</p> <p>もし、プログラムが設定されていない場合、無視して実行されますが、指定Cレジスタのメモリエリアが確保されていないため、「MOVE」等でCレジスタの内容を読み込もうとした場合、エラーとなります。</p> <p>また、数値演算のプログラム中、指定した数値結果レジスタに対応するチェッカの実行がされていない場合、数値結果レジスタにあらかじめ格納されている内容（ex.チェッカを設定したときに格納された内容、前回実行したときにバックアップされていた内容、別のプログラムで実行したときに格納されていた内容etc.）が読み込まれます。</p> <p>たとえば、C1=L11*100において ラインNo.1が実行（LSCAN）されていないとき L11に前回格納されていた内容が読み込まれます。</p> <p><文例> CSCAN 1,3 ;数値演算C1~C3を実行します。</p> <p><エラーについて> 以下の動作を実行した時、エラーランプがONになります。</p> <ul style="list-style-type: none"> ・数値演算プログラムに書かれているチェッカが未設定の時 ・演算結果が32ビットをオーバーした時 ・0除算の時 ・エラー発生したチェッカの数値結果を引用した時 	

CURSOR		カーソル																																	
機能	十字型カーソルの描き込み																																		
書式	CURSOR Δ [v], [v], [v]																																		
角解	<p>第1オペランドでは、カーソルの片側長をドット数で指定し、第2オペランドでは、カーソルの中心の「列」を、第3オペランドでは「行」をそれぞれドット数で指定します。片側長は自然数で指定してください。中心の座標は、下表の画面構成を参考にして設定してください。画面構成は品種モードでの設定によります。すでに描き込んだカーソルを消去する場合は、再度重ね描きを行うことにより消去できます。</p> <p>(参考) 画面構成 田の字型構成</p> <p>→ (列)</p> <p>0 511 512 1023</p> <table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td style="width: 40px;"></td><td style="width: 40px;">A</td><td style="width: 40px;">B</td></tr> <tr><td style="text-align: right;">479</td><td colspan="2" style="text-align: center;">//</td></tr> <tr><td style="text-align: right;">511</td><td colspan="2" style="text-align: center;">//</td></tr> <tr><td style="text-align: right;">512</td><td style="width: 40px;">C</td><td style="width: 40px;">D</td></tr> <tr><td style="text-align: right;">991</td><td colspan="2" style="text-align: center;">//</td></tr> <tr><td style="text-align: right;">1023</td><td colspan="2" style="text-align: center;">//</td></tr> </table> <p style="margin-left: 100px;">↓ (行)</p> <p style="margin-left: 100px;">(例) Bメモリーにカーソルを設定しようとするとき中心座標は(800, 300)などの様に設定します。</p> <p>横一列構成</p> <p>→ 列</p> <p>0 511 512 1023 1024 1536 2047</p> <table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td style="width: 40px;"></td><td style="width: 40px;">A</td><td style="width: 40px;">B</td><td style="width: 40px;">C</td><td style="width: 40px;">D</td></tr> <tr><td style="text-align: right;">479</td><td colspan="4" style="text-align: center;">//</td></tr> <tr><td style="text-align: right;">511</td><td colspan="4" style="text-align: center;">//</td></tr> </table> <p style="margin-left: 100px;">↓ 行</p> <p style="margin-left: 100px;">例: CURSOR 40, 100, 120</p> <div style="display: inline-block; vertical-align: middle;"> <p>(100, 120) の座標</p> </div> <p><文例></p> <p>CURSOR 40, 100, 120 ; 行120ドット列100ドットを中心に片側長40ドットのカーソルを描き込みます。</p> <p><注意></p> <p>CURSOR 60, 10, 30 のように、メモリがはみ出す場合は、はみ出した部分のみ表示されません。</p> <p>※カーソルの定義：画面構成中「メモリ画面-カーソル画面」に描ける「+」字のことです。"SHOW" コマンドで指定する記号では'C'で指定されます。</p>			A	B	479	//		511	//		512	C	D	991	//		1023	//			A	B	C	D	479	//				511	//			
	A	B																																	
479	//																																		
511	//																																		
512	C	D																																	
991	//																																		
1023	//																																		
	A	B	C	D																															
479	//																																		
511	//																																		

D E L A Y		デ イ レ イ
機 能	プログラム遅延 (一時停止)	
書 式	DELAY Δ $\left[\begin{matrix} v \\ i \end{matrix} \right]$	
角 解 説	<p>このコマンドにより、プログラムの実行を指定時間だけ遅延させることができます。 指定は、10msec単位で行います。</p> <p><文例></p> <p style="padding-left: 40px;">DELAY 30 ; プログラムを300msec遅延します。 (プログラムを300msec、このステップにて一時停止します。)</p>	

DISP		ディスプレイ
機能	メッセージプレーン上の文字表示 (カラム単位)	
書式	<p>1) 表示開始位置が前記されていないとき</p> <p>イ) aレジスタによる表示</p> $\text{DISP } \Delta \left[\begin{matrix} v \\ i \end{matrix} \right], \left[\begin{matrix} v \\ i \end{matrix} \right], a, \left[\begin{matrix} v \\ i \end{matrix} \right]$ <p>ロ) 文字列による表示</p> $\text{DISP } \Delta \left[\begin{matrix} v \\ i \end{matrix} \right], \left[\begin{matrix} v \\ i \end{matrix} \right], " " \left[\begin{matrix} (v) \\ (i) \end{matrix} \right]$	<p>2) 表示開始位置が前記されているとき</p> <p>イ) aレジスタによる表示</p> $\text{DISP } \Delta a, \left[\begin{matrix} v \\ i \end{matrix} \right]$ <p>ロ) 文字列による表示</p> $\text{DISP } \Delta " " \left[\begin{matrix} (v) \\ (i) \end{matrix} \right] \quad () \text{は省略可}$
角軍 言兌	<p>1) 表示開始位置が前記されていないとき</p> <p>イ) aレジスタによる表示</p> <p>第1・第2オペランドは、表示を開始する位置 (指定位置は<参考>を参照) をカラム単位で指定します。第1オペランドは「列」第2オペランドは「行」を指定します。第3オペランドは、表示を開始するaレジスタのNo.を、第4オペランドは表示するaレジスタの個数を自然数で指定します。</p> <p><文例></p> <p>DISP 3,3,A80,10 ;3行3列を先頭にA80からA89までの10個の桁を表示します。 3行3列~3行12列まで表示されます。</p> <p>ロ) 文字列による表示</p> <p>第1・第2オペランドは、表示を開始する位置をカラム単位で指定します。第1オペランドは「列」第2オペランドは「行」を指定します。第3オペランドは表示する文字列を、第4オペランドは表示する文字の個数を自然数で指定します。第3オペランドの文字数は最大15文字までです。第3オペランドに「"」で指定した内容を1回だけ表示するとき、第4オペランドの個数の指定は省略することが可能です。また、第4オペランドの数が第3オペランドの文字数より多い場合、第3オペランドで指定した文字列の最後の文字を、繰り返し残りの字数分表示します。なお、少ない場合は、第4オペランドの文字数分表示します。</p> <p><文例></p> <p>DISP 10,12," OK!!!" ,5 ;12行10列を先頭に" OK!!!" と表示します。</p> <p>DISP 10,12," #" ,10 ;12行10列を先頭に" #" を10個表示します。</p> <p>2) 表示開始位置が前記されているとき、または、[LOCATE]・[DISP]により表示を開始する位置をカラム単位で指定されていて、続けて表示する場合、前記の場合の第1・第2オペランドは省略できます。また、一度[LOCATE]・[DISP]で表示位置を指定した場合、再度「DISP」または「LOCATE」で表示位置を指定するか、表示の最後が61列になるまで、「DISP」で指定した表示内容を順番に表示します。61列を超えた文字は無視されます。</p> <p>イ) aレジスタによる表示</p> <p>表示を開始する位置は[LOCATE]・[DISP]で指定されています。第1オペランドは、表示を開始するaレジスタのNo.を、第2オペランドは、表示するaレジスタの個数を自然数で指定します。</p> <p><文例></p> <p>LOCATE 3, 5 DISP A 5, 5 ; 5行3列を先頭にA5からA9まで5個のキャラクターを表示します。</p>	

ロ) 文字列による表示

表示を開始する位置は[LOCATE]・[DISP]で指定されています

- ・第1オペランドは表示する文字列を、第2オペランドは表示する文字の個数を自然数で指定します。第1オペランドの文字数は最大15文字です。第1オペランドに「"」で指定した内容を1回だけ表示するとき、第2オペランドの個数の指定は省略することが可能です。
- ・第2オペランドの数が第1オペランドの文字数より多い場合、第1オペランドで指定した文字列の最後の文字を、繰り返し残りの文字数分表示します。なお少ない場合は、第2オペランドの文字数分表示します。

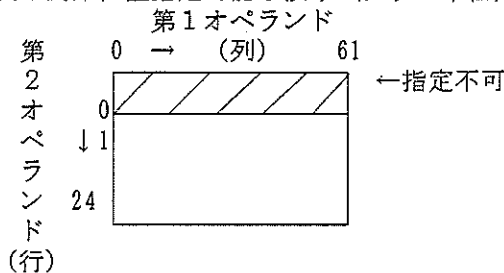
<文例>

```
LOCATE      8, 10  
DISP       " NG"
```

; 10行8列を先頭に" NG" と表示します。

<参考>

1) 表示開始位置指定可能な領域 (カラム単位)



(注意) aレジスタによる表示の場合、0X00 (Null) を表示の最後と判断するため表示するaレジスタの中に0X00 (Null) が格納されていると、それ以降は表示されませんので御注意ください。

(例) A1~A10まで表示する場合 A5=0X00 (Null) が格納されていると、A1~A4までしか表示されません。

(注意) キー入力、RS232C入力、"CODE" などにより aレジスタに格納された文字コードで、画面表示不可能な文字 (アスキーコード: 0X00~0X1F) を表示しようとすると無視して、前詰めされて、画面表示されますが、不具合の原因となりますので上記の様な文字は極力画面表示しないでください。

(注意) イメージチェッカ30P・30RPで対応しています漢字は、JIS第1水準です。これ以外の文字を画面表示しようとすると、不具合の原因となりますので、表示しないでください。

DISPD		ディスプレイ
機能	メッセージプレーン上の文字表示 (ドット単位)	
書式	<p>1) 表示開始位置が前記されていないとき</p> <p>イ) aレジスタによる表示</p> $\text{DISPD } \Delta \left[\begin{matrix} v \\ i \end{matrix} \right], \left[\begin{matrix} v \\ i \end{matrix} \right], a, \left[\begin{matrix} v \\ i \end{matrix} \right]$ <p>ロ) 文字列による表示</p> $\text{DISPD } \Delta \left[\begin{matrix} v \\ i \end{matrix} \right], \left[\begin{matrix} v \\ i \end{matrix} \right], " \quad " , \left[\begin{matrix} (v) \\ (i) \end{matrix} \right]$	<p>2) 表示開始位置が前記されているとき</p> <p>イ) aレジスタによる表示</p> $\text{DISPD } \Delta \quad a, \left[\begin{matrix} v \\ i \end{matrix} \right]$ <p>ロ) 文字列による表示</p> $\text{DISPD } \Delta " \quad " , \left[\begin{matrix} (v) \\ (i) \end{matrix} \right] \quad () \text{は省略可}$
角解	<p>1) 表示開始位置が前記されていないとき</p> <p>イ) aレジスタによる表示</p> <p>第1・第2オペランドは、表示を開始する位置 (指定位置は<参考>を参照) をドット単位で指定します。</p> <p>第1オペランドは「列」、第2オペランドは「行」を指定します。</p> <p>第3オペランドは、表示を開始する a レジスタの No. を、</p> <p>第4オペランドは、表示する a レジスタの個数を自然数で指定します。</p> <p><文例></p> <p>DISPD 100,120,A10,5 ;行120',列100',を先頭にA10からA14まで5個の文字を表示します。</p> <p>ロ) 文字列による表示</p> <p>第1・第2オペランドは、表示を開始する位置をドット単位指定します。第1オペランドは「列」、第2オペランドは「行」を指定します。第3オペランドは表示する文字列を、第4オペランドは表示する文字の個数を自然数で指定します。第3オペランドの文字数は最大15文字までです。第3オペランドに「"」で指定した内容を1回だけ表示するとき、第4オペランドの個数の指定は省略することが可能です。</p> <p>また、第4オペランドの数が第3オペランドの文字数より多い場合、第3オペランドで指定した文字列の最後の文字を、繰り返し残りの字数分表示します。なお、少ない場合は第4オペランドの字数分表示します。</p> <p><文例></p> <p>DISPD 200,150,"OK!!",5 ;行150',列200',を先頭に"OK!!"と表示します。</p> <p>2) 表示開始位置が前記されているとき、または、[LOCATD]・[DISPD]により表示を開始する位置をドット単位で指定されていて、続けて表示する場合、前記の場合の第1・第2オペランドは省略できます。また、一度 [LOCATD]・[DISPD] で表示位置を指定した場合、再度「DISPD」または「LOCATD」で表示位置を指定するか、表示の最後が489ドットになるまで、「DISPD」で指定した表示内容を順番に表示します。489'を越えた文字は無視されます。</p> <p>イ) aレジスタによる表示</p> <p>表示を開始する位置は [LOCATD]・[DISPD] で指定されています</p> <p>第1オペランドは、表示を開始する a レジスタの No. を</p> <p>第2オペランドは、表示する a レジスタの個数を自然数で指定します</p> <p><文例></p> <p>LOCATD 120,300</p> <p>DISPD A10,10</p> <p>;行300',列120',を先頭にA10からA19まで10個のキャラクターを表示します。</p>	

ロ) 文字列による表示

- ・表示を開始する位置は「LOCATD」・「DISPD」で指定されています。第1オペランドは表示する文字列を、第2オペランドは表示する文字の個数を自然数で指定します。第1オペランドの文字数は最大15文字です。第1オペランドに「"」で指定した内容を1回だけ表示するとき、第2オペランドの個数の指定は省略することが可能です。
- ・第2オペランドの数が第1オペランドの文字数より多い場合第1オペランドで指定した文字列の最後を繰り返し残りの文字数分表示します。なお、少ない場合は第2オペランドの文字数分表示します。

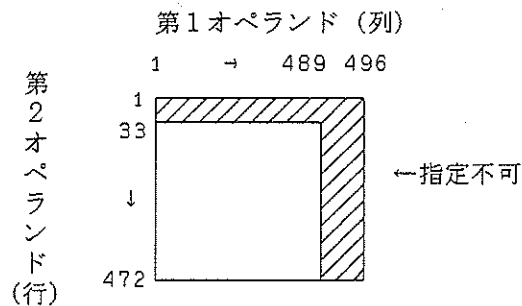
<文例>

```
LOCATD      150, 100
DISPD       " NG"
```

;行100, 列150, 列を先頭に" NG" と表示します。

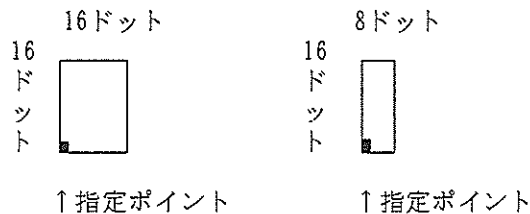
<参考>

1) 表示開始位置指定可能な領域 (ドット単位)



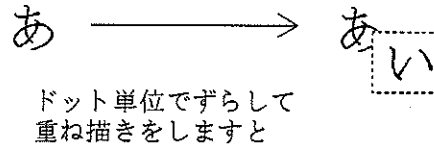
2) 文字の構成 (ドット単位)

イ) 漢字の場合 ロ) 漢字以外の場合



- 3) ドット単位で描き込まれている文字の表示位置とビット単位でずらして描き込んだ場合重ね描きされない部分は、残ります。

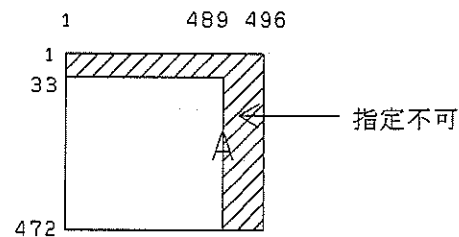
(例)



- 4) 表示される文字の一部が、指定範囲外にある場合は、指定された座標が領域内の場合は表示できます。

LOCATD 485,100

DISPD "A"



(注意)

aレジスタによる表示の場合0X00 (Null) を表示の最後と判断するため表示するaレジスタの中に0X00 (Null) が格納されていますと、それ以降は表示されませんのでご注意ください。

(例) A1~A10まで表示する場合 A5=0X00 (Null) が格納されていますと、A1~A4までしか表示されません。

(注意) キー入力、RS232C入力、"CODE" などにより aレジスタに格納された文字コードで、画面表示不可能な文字 (アスキーコード: 0X00~0X1F) を表示しようとすると無視して、前詰めされて、画面表示されますが、不具合の原因となりますので上記の様な文字は極力画面表示しないでください。

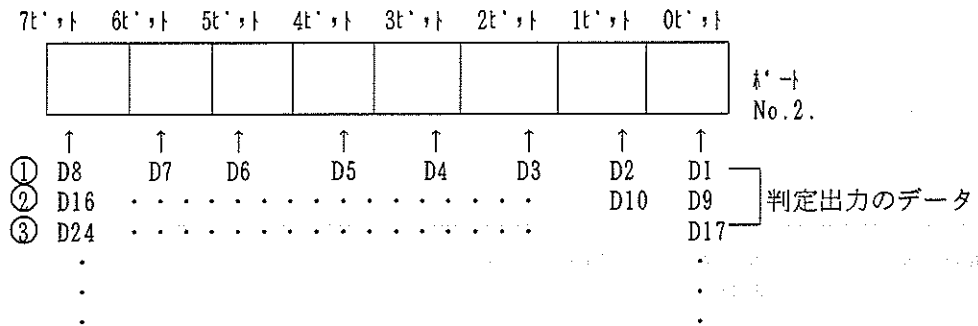
(注意) イメージチェッカ30P・30RPで対応しています漢字は、JIS第1水準です。これ以外の文字を表示しようとしますと、不具合の原因となりますので表示しないでください。

D I V	ディバイド
機能	レジスタ間除算
書式	$D I V \quad \Delta \quad \begin{bmatrix} v \\ i \end{bmatrix}, \begin{bmatrix} v \\ i \end{bmatrix}, v$
角解言兌	<p>第1オペランドの内容から第2オペランドの内容を除算し、第3オペランドの指定するvレジスタに格納します。 除算結果は、小数点以下を切り捨てて整数として格納されます。</p> <p>除数（第2オペランド）が0の場合や、オーバーフローした時はエラーとなり、オーバーフローフラグがONし、第3オペランドの値は変化しません。</p> <p><文例> D I V V20, V21, V200 ; V20÷V21 → V200</p>

D-OUT	ディーアウト																					
機能	判定出力の内容の平行アウト																					
書式	D-OUT Δ $\begin{bmatrix} v \\ i \end{bmatrix}$, $\begin{bmatrix} (v) \\ (i) \end{bmatrix}$, $\begin{bmatrix} (v) \\ (i) \end{bmatrix}$ () は省略可																					
角解言兌	<p>RSCAN、DSCANを実行した後 品種モードで設定された、判定出力の判定結果 (D1～) を全て、平行出力します。</p> <p>第1オペランド・・・使用するポートNo. を番号で指定します。 (「P-OUT」命令と同じです)</p> <table border="1" data-bbox="727 869 1251 969"> <thead> <tr> <th>指定する番号</th> <th>ポートNo.</th> <th>ビット数</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>2</td> <td>8</td> </tr> <tr> <td>3</td> <td>2、3</td> <td>16</td> </tr> </tbody> </table> <p>第2オペランド・・・平行ハンドシェイクの実行を指定します。</p> <table border="1" data-bbox="724 1081 1327 1182"> <thead> <tr> <th>番号</th> <th>平行ハンドシェイク</th> </tr> </thead> <tbody> <tr> <td>(省略)、0</td> <td>行わない</td> </tr> <tr> <td>1</td> <td>行う</td> </tr> </tbody> </table> <p>品種モードでの「平行ハンドシェイクを行うー行わない」の設定は無視されます。</p> <p>第3オペランド・・・コマンド実行時に”RSCAN 1, 512,” ”DSCAN 1, 512”を実行するかどうかを選択します。</p> <table border="1" data-bbox="727 1379 1390 1480"> <thead> <tr> <th>番号</th> <th>”RSCAN”、”DSCAN”</th> </tr> </thead> <tbody> <tr> <td>(省略)、0</td> <td>実行する</td> </tr> <tr> <td>1</td> <td>実行しない</td> </tr> </tbody> </table> <p>(注意) 第3オペランドを省略する場合、第2オペランドは、'0' を入れてください</p> <p><文例> D-OUT 1, 0, 1; ”RSCAN 1, 512”, ”DSCAN 1, 512”を実行した後、平行ハンドシェイクをしないで、判定出力の判定結果を平行ポートNo. 2より平行出力します。</p>	指定する番号	ポートNo.	ビット数	1	2	8	3	2、3	16	番号	平行ハンドシェイク	(省略)、0	行わない	1	行う	番号	”RSCAN”、”DSCAN”	(省略)、0	実行する	1	実行しない
指定する番号	ポートNo.	ビット数																				
1	2	8																				
3	2、3	16																				
番号	平行ハンドシェイク																					
(省略)、0	行わない																					
1	行う																					
番号	”RSCAN”、”DSCAN”																					
(省略)、0	実行する																					
1	実行しない																					

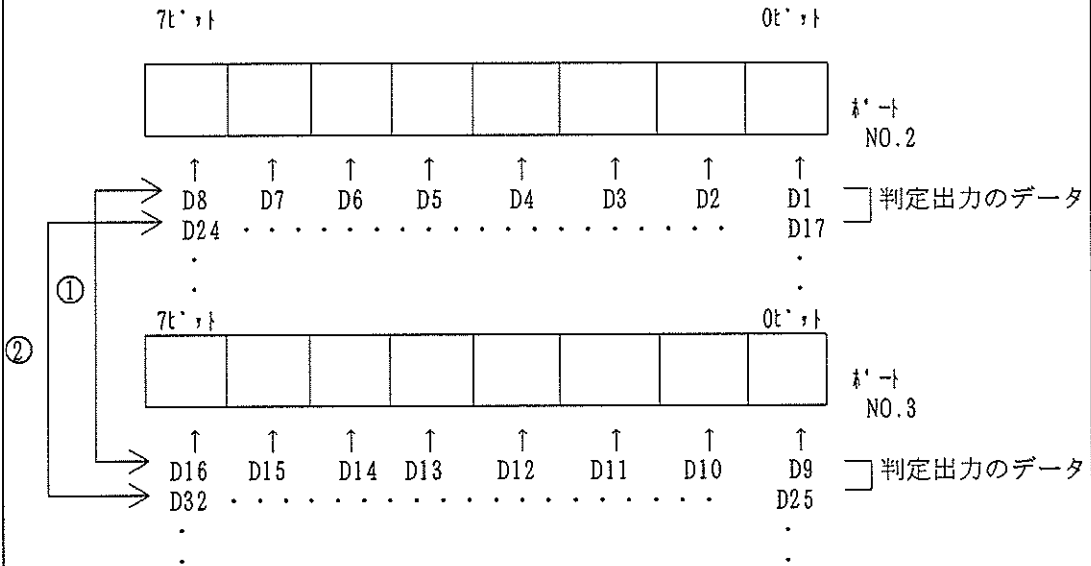
1) データ出力の方法

イ) 第1オペランドで「1」を指定した場合 (データを8個ずつ出力します)



1 D1~D8まで出力した後、順次② D9~D16、③ D17~D24 . . . とデータがなくなるまで、出力します。

ロ) 第1オペランドで「3」を指定した場合 (データを16個ずつ出力します)



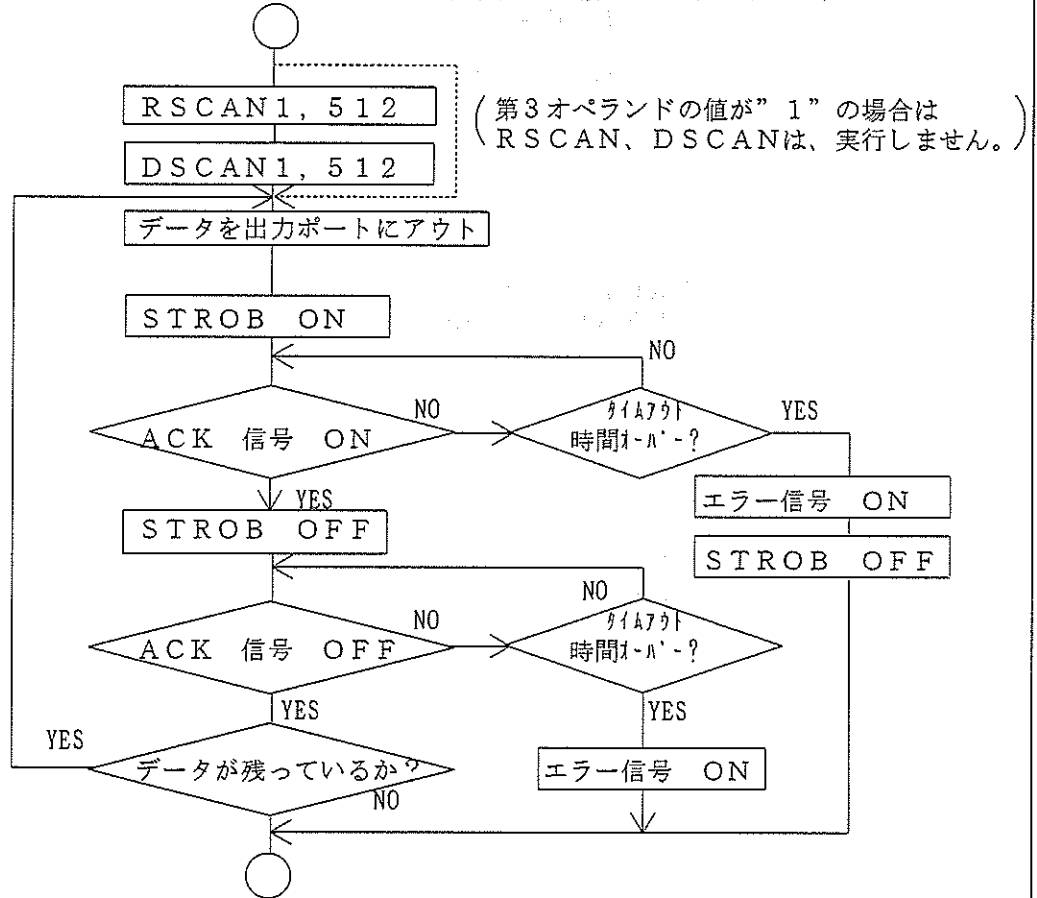
出力の順番

- ① D1~D8をポートNO.2、D9~D16をポートNO.3に同時に出力した後
- ② D17~D24をポートNO.2、D25~D32をポートNO.3に同時に出力という様に順次、データがなくなるまで、出力します。

2) データ出力の流れ

以下に、ハンドシェイクを行う場合と、行わない場合の動作の流れをフローチャートに示します。

イ) パラレルハンドシェイクを行う場合 (第2オペランド=1)

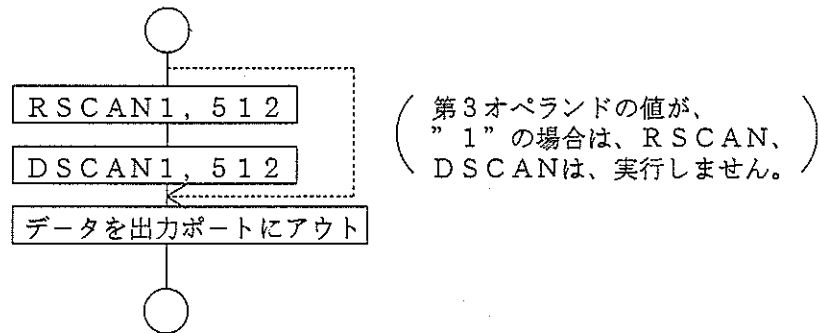


注) ・ ACK信号 ON-OFFの「タイムアウト時間」の設定値は「初期化ブロック-パラレルハンドシェイク-行う」で設定された設定値によります。初期化ブロックでのパラレルハンドシェイクを「行う-行わない」の選択は無視されますが、忘れずに「タイムアウト時間」を設定してください。

・ タイムアウトエラー発生時、エラー信号がONになりますがエラー信号OFFされるタイミングは、"START"コマンドのスタート信号・品種切替信号入力時、"READ"、"READF"、"ESCAN"実行時です。

角 評 言 兌

ロ) パラレルハンドシェイクを行わない場合)



STROB信号のON-OFFは実行されませんので、データの読み込みは、READY信号ONのタイミングで取り込んでください。

3) 結線方法

「イメージチェッカ30または、30R導入マニュアル」の「8-1パラレル信号による通信」を参照してください。

注) 「初期化ブロッカーパラレルハンドシェイカー、行う、行わない」の選択は無視されます。

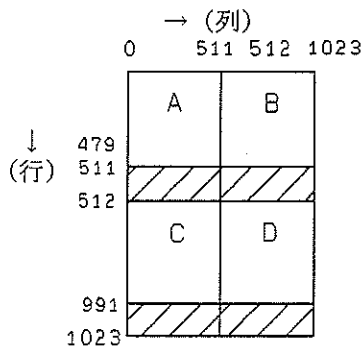
DSCAN	ディースキャン
機能	判定演算の実行 (Dレジスタ)
書式	DSCAN Δ $\begin{bmatrix} v \\ i \end{bmatrix}$, $\begin{bmatrix} v \\ i \end{bmatrix}$
角解 言説	<p>第1オペランドで指定する判定演算 (Dレジスタ) NO. から、第2オペランドで指定する判定演算 (Dレジスタ) NO. までの演算を実行します。</p> <p>指定する判定演算のプログラムはあらかじめ、「判定出力」で設定しておいてください。もし、プログラムが設定されていない場合、無視して実行されますが、指定Dレジスタのメモリエリアが確保されていないため、「MOVE」等でDレジスタの内容を読み込もうとした場合、エラーとなります。</p> <p>また、判定演算のプログラム中、指定した判定結果レジスタに対応するチェックの実行がされていない場合、判定結果レジスタにあらかじめ格納されている内容 (ex. チェックを設定したときに格納された内容、前回実行したときにバックアップされていた内容、別のプログラムで実行したときに格納されていた内容 etc.) が読み込まれます。</p> <p>たとえば、 判定演算 D1=L11*W1 において ラインNo. 1が実行 (LSCAN) されていないとき L11に前回格納されていた内容が読み込まれます。</p> <p><文例> DSCAN 2,5 ; 判定演算D2~D5を実行します。</p> <p><エラーについて> 以下の動作を実行した時、エラーランプがONします ・判定出力 (D) プログラムに書かれているチェックが未設定の時 ・エラーが発生したチェックの判定結果を引用した時</p>

E S C A N	イースキャン
機 能	露出補正チェッカの検査の実行
書 式	E S C A N Δ $\begin{bmatrix} v \\ i \end{bmatrix}$, $\begin{bmatrix} v \\ i \end{bmatrix}$, $\begin{bmatrix} (v) \\ (i) \end{bmatrix}$, $\begin{bmatrix} (v) \\ (i) \end{bmatrix}$ () は省略可です。
角 解	<p>第1オペランドで指定された露出補正チェッカNO. (メモリエリア) から第2オペランドで指定された露出補正チェッカNO. (メモリエリア) までの検査を実行します。</p> <p>露出補正チェッカは各メモリエリア毎に1個ずつ設定されますので、チェッカNO. はメモリエリアを示します。すなわち、メモリエリア (A, B, C, D) = チェッカNo (1, 2, 3, 4) に対応します。</p> <p>第3、第4オペランドでは、それぞれx方向、y方向の平行移動量を画素数で指定します。移動量が指定されているとき、移動量の値だけ露出補正チェッカを移動し、検査を実行します。</p> <p>第3、第4オペランドが「省略」または「0」が指定されているとき、チェッカを移動させずに検査を実行します。また、露出補正チェッカの検査条件 (判定条件) は、実行ブロックで設定された条件により実行されます。 また実行ブロックで「画面密着有り」を選択していますとコマンド実行と同時に実行されます。</p> <p>照明モードが「ノーマル光源」か「フラッシュ光源」かは、品種モードの実行ブロックで設定しておいてください。 露出補正チェッカは、「フラッシュ光源」に照明モードを設定しますと意味のない、検査となります。 したがって、このコマンド (E S C A N) は、「ノーマル光源」に照明モードを設定しておいてください。</p> <p><文例> E S C A N 1,2,30,50 ;露出補正チェッカNO.1 (メモリA) ~NO2 (メモリB) を (Δx, Δy) = (30,50) だけ平行移動し、検査を実行します。</p>

(参考)

- 1) 品種モードで「チェッカパターン表示選択」を「位置補正に追従して表示」を選択している場合、チェッカパターンの消去・描き込みに多少時間を要します。それでは問題のある場合、「固定位置に表示」又は「表示しない」を選択して時間を短縮してください。
- 2) 移動量を入力する際、画面座標は下図のようになっています。移動方向の符号に御注意ください。
画面構成は実行ブロックでの設定によります。

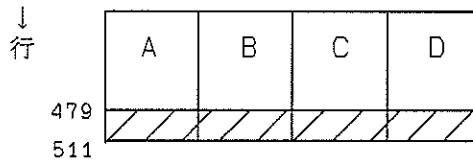
田の字型構成



→ 列

横一列構成 1535

0 511 512 1023 1024 1536 2047



- 3) 実行時、露出補正エラーが発生してもプログラムはストップしません。エラー処理したいときには、露出補正判定結果又はエラーレジスタ (DB 3) がエラーの時に「1」となることで判断してください。

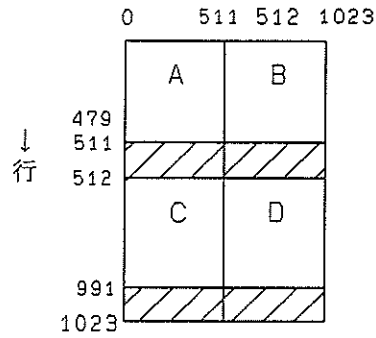
<エラーについて>

露出補正エラー発生時、エラーランプがONします。

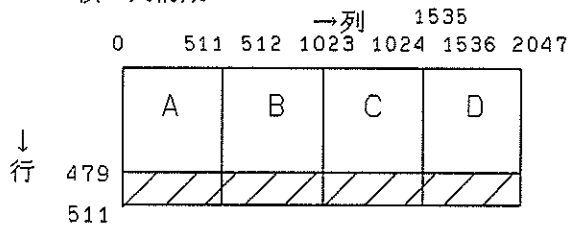
F S C A N		エフスキャン	
機 能	特徴抽出チェッカの検査の実行		
書 式	$F S C A N \quad \Delta \quad \begin{bmatrix} v \\ i \end{bmatrix}, \begin{bmatrix} v \\ i \end{bmatrix}, \begin{bmatrix} (v) \\ (i) \end{bmatrix}, \begin{bmatrix} (v) \\ (i) \end{bmatrix}$ <p style="text-align: right;">() は省略可</p>		
角 説	<p>第1オペランドで指定された特徴抽出チェッカNoから、第2オペランドで指定された特徴抽出チェッカNoまでの検査を実行します。</p> <p>このとき、第3、第4オペランドでは、それぞれx方向、y方向の平行移動量を画素数で指定します。移動量が指定されているとき、「グループ選択」で指定されたグループの補正量と移動量の和の値だけ特徴抽出チェッカを移動し、検査を実行します。第3、第4オペランドが「省略」または「0」が指定されているとき、「グループ選択」で指定されたグループの補正量だけ移動し、検査を実行します。また、特徴抽出チェッカの検査条件（判定条件、グループ選択）は、品種ブロックで設定された条件により実行されます。</p> <p>グループの補正量による移動は行なわず、数値データ等の移動量のみを移動を行なう場合、移動量の算出の際に位置補正の補正量の分を差引いておいてください。</p> <p>グループの補正量は、グループNO. で指定された位置補正の補正量によりますので、グループNO. を指定された位置補正が実行されないとき、補正量は電源を切らない限りメモリに残っているため、前回実行されたときの補正量により実行されます。</p> <p><文例> F S C A N 5,10,40,50 ;特徴抽出チェッカNo5~No10 をグループの補正量より補正後、 (Δx、Δy) = (40、50) だけ平行移動し、検査を実行します。</p> <p>(参考) 1) 品種モードで「チェッカパターン表示選択」を「位置補正に追従して表示」を選択している場合、チェッカパターンの消去・描き込みに多少時間を要します。それでは問題のある場合、「固定位置に表示」又は「表示しない」を選択して時間を短縮してください。</p>		

- 2) 移動量を入力する際、画面座標は下図のようになっています。移動方向の符号に御注意ください。
画面構成は実行ブロックでの設定によります。

田の字型構成
→列



横一列構成



- 3) 実行時、特徴抽出エラーが発生してもプログラムはストップしません。

<エラーについて>

ラベリング個数が、128個を越えた時、エラーランプがONします。

GOSUB (~RET)		ゴージャブ (~レット)	
機能	サブルーチンコール		
書式	GOSUB Δ $\begin{pmatrix} i \\ 1 \end{pmatrix}$		
角解 言説	<p>同一プログラム内に作成されたサブルーチンをコールします。 "RET"により復帰します。 オペランドは、ジャンプするステップ数またはラベル名を指定します。 ステップ数は-63~63(0を除く)の範囲で、最終行を越えない様に設定してください。</p> <p><文例> GOSUB 10 ;10ステップ先にジャンプし"RET"により復帰します。 GOSUB L1 ;ラベルL1のステップにジャンプし、"RET"により復帰します。</p> <p><注意> ネスティングは、"GOSUB"のみで合計32まで、ネスティングすることができます。</p> <p><参考> 'GOSUB' コマンドによるネスティングプログラム実行例</p> <pre> 1 : ST CLR CRT 10 : GOSUB L1 11 : 20 : GOTO L3 29 : L1 35 : GOSUB L2 38 : RET 41 : L2 47 : RET 48 : L3 61 : RET </pre> <p>この例では、ネスティングが2個設定されています。 ネスティング部でのプログラム実行順です。</p>		

GOTO		ゴートウ
機能	無条件ジャンプ	
書式	GOTO Δ $\left(\begin{array}{c} i \\ 1 \end{array} \right)$	
角解 言兌	<p>直接数値で指定した相対ステップ数、または、同一プログラム内にあるラベルによって指定したステップに無条件でジャンプします。</p> <p>ステップ数は-63~63（0を除く）の範囲で、最終行を越えない様に設定してください。</p> <p><文例></p> <p>GOTO 3 ; 3ステップ先へジャンプします。</p> <p>GOTO L1 ; ラベルL1のステップにジャンプします。</p>	

I F		イフ
機能	条件つきジャンプ	
書式	<p>1) 数値レジスタの比較</p> $IF \Delta \begin{pmatrix} v \\ i \\ j \\ s \end{pmatrix}, \begin{pmatrix} v \\ i \\ j \\ s \end{pmatrix}, q, \begin{pmatrix} i \\ 1 \end{pmatrix}$ <p>2) 文字レジスタの比較</p> $IF \Delta \begin{pmatrix} a \\ G \end{pmatrix}, \begin{pmatrix} a \\ G \end{pmatrix}, q, \begin{pmatrix} i \\ 1 \end{pmatrix}$	<p>第3オペq; 判定条件を示す 文字記号</p> <p>EQ → 第1オペ = 第2オペ : i → j NE → " ≠ " : j → i GE → " ≥ " : j → i GT → " > " : j → j LE → " ≤ " : i → i LT → " < " : i → j</p> <p>第4オペ; ジャンプ先 i; ステップ数 1; ラベル</p>
角解	<p>第1オペランドと第2オペランドの比較を行ない、条件が成立すれば指定した相対ステップ数、またはラベルへジャンプします。また、不成立ならば次ステップを実行します。 ステップ数は-63~63(0を除く)の範囲で、最終行を越えない様に設定してください。</p> <p>1) 数値レジスタの比較 (vとv), (iとv), (vとs), (iとs), (jとj), (jとi)の比較を行ないます。</p> <p><文例> IF V20,120,LT,3 ; V20 < 120ならば3ステップ先へ進みます。 IF V10,V20,EQ,L1 ; V10 = V20ならばL1へジャンプします。 IF DS3,1,EQ,ER ; S3 = 1ならばERへジャンプします。</p> <p><文例> IF CT13,60,EQ,L1;CT13=60ならばL1へジャンプします。 IF CT13,CT23,NE,L1;CT13≠CT23ならばL1へジャンプします。</p> <p>(注意) vレジスタによるメモリエリア及びモード指定はできません。 IF CT[V1,V2],60,EQ,L1 } この様な指定はできません。 IF DT[V1],1,NE,L1 }</p> <p>2) 文字レジスタの比較 (aとa)、もしくは(Gとa)の比較を行ないます。</p> <p><文例> IF A1,A120,NE,-10 ; A1≠A120ならば10ステップ戻ります。 IF A1,'Y',EQ,ST ; A1='Y'(0X59)ならばSTへジャンプします。</p>	

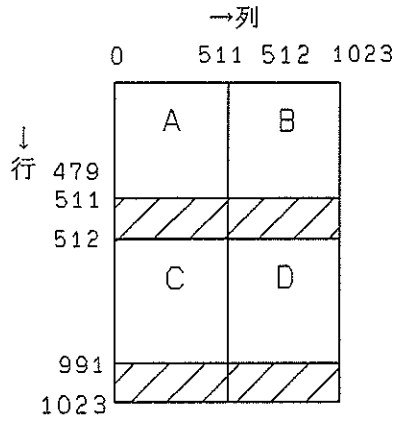
I F N S T R		イフノットストリングス
機 能	文字列の比較 (違う場合ジャンプ)	
書 式	I F N S T R Δ a , a , $\begin{bmatrix} v \\ i \end{bmatrix}$, $\begin{bmatrix} l \\ i \end{bmatrix}$	
角 注	<p>第1、第2オペランドでは比較の対象となる a レジスタの開始NO. を指定します。 第3オペランドでは比較する文字数を自然数で指定します。 第4オペランドでは比較の結果が違う場合にジャンプするラベル名又は、ジャンプするステップ数を指定します。 比較の結果、同じ場合は次ステップを実行します。 比較する a レジスタのNo. が A 1 0 0 0 を越えない様に、レジスタNo. と比較する個数を設定してください。 ジャンプするステップ数は - 6 3 ~ 6 3 (0 を除く) の範囲で、最終行を越えない様に設定してください。</p> <p><文例> I F N S T R A1, A5, 3, LL ; A1~A3とA5~A7までの 3文字づつを比較し違う場合は” LL”へジャンプします。</p>	

I F S T R		イフストリングス
機能	文字列の比較 (同じ場合ジャンプ)	
書式	$IFSTR \Delta a, a, \begin{pmatrix} v \\ i \end{pmatrix}, \begin{pmatrix} l \\ i \end{pmatrix}$	
角平 説	<p>第1、第2オペランドでは比較の対象となる a レジスタの開始NO. を指定します。</p> <p>第3オペランドでは比較する文字数を自然数で指定します。</p> <p>第4オペランドでは比較の結果が同じ場合にジャンプするラベル名又はジャンプするステップ数を指定します。</p> <p>比較の結果、違う場合は次ステップを実行します。</p> <p>比較する a レジスタのNo. が A1000 を越えない様に、レジスタNo. を比較する個数を設定してください。</p> <p>ジャンプするステップ数は、-63 ~ 63 (0を除く) の範囲で最終行を越えない様に設定してください。</p> <p><文例> IFSTR A10, A20, 5, OK ; A10~A14とA20~A24 までの5文字ずつを比較し、 同じであれば“OK”へ ジャンプします。</p>	

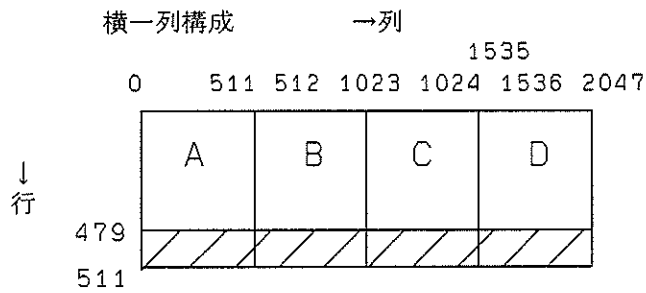
角卒 言兌

2) 移動量を入力する際、画面座標は
下図のようになっていますので、
移動方向の符号に御注意ください。
画面構成は実行ブロックでの構成によります。

田の字型構成



横一列構成



KEYIN	キーイン
機能	キーボードからの入力
書式	KEYIN Δ a, $\left[\begin{matrix} (v) \\ (i) \end{matrix} \right], (v), \left[\begin{matrix} (a) \\ (v) \\ (i) \\ (h) \end{matrix} \right]$ () は省略可
	<p>キーボードから設定された文字数、または、区切り記号が入力されるまで a レジスタに文字を格納します。</p> <p>第1オペランド 格納を開始する a レジスタを指定</p> <p>第2オペランド 格納する文字数を指定 (1、2、3 . . .) 省略した場合は、文字数 = 1 となります。</p> <p>第3オペランド 「区切り記号」を含めた入力文字数を格納する v レジスタを指定</p> <p>第4オペランド 「区切り記号」を「アスキーコード」で指定 (0X0 ~ 0X7F) 省略されますと 0XD = " CR" となります。</p> <p>このコマンドは以下の場合に終了し次のステップに進みます。</p> <p>1. 「区切り記号」の指定がある場合は、 ① 指定個数の文字入力がある ② 「区切り記号」が入力される のどちらかの場合</p> <p>2. 「区切り記号」の指定がない場合は、 ① 指定個数の文字入力がある ② " CR" が入力される のどちらかの場合</p> <p><文例> KEYIN A10,10,V10,0X41 ; 10文字または"A" (=0X41)がキー入力するまで入力し続けます。</p>

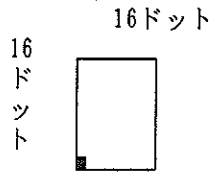
KEYSNS		キーセンス
機 能	キー入力 (入力がなくても次ステップへ)	
書 式	KEYSNS Δ a	
角 注	<p>キー入力がなくても次ステップを実行します。キー入力があると、指定された a レジスタに格納して次ステップを実行します。 (Break in機能)</p> <p>キー入力がなかった場合、a レジスタは 'Null'(0X00) が格納されます。</p> <p><文例> KEYSNS A10 ; キー入力があると、A10に格納するが、キー入力がなくても次のステップを実行します。</p> <p><注意> <u>プログラムブロック</u>で、"KEYSNS" コマンドを実行した場合、キーボードからの入力のタイミングにより、キー入力が格納されない場合があります。 しかし、<u>実行ブロック</u>で、プログラム実行 (実際の実行中) では、上記の様な問題はありません。</p>	

L E N	レングス						
機 能	数値レジスタの符号、桁数の読み込み						
書 式	LEN Δ v, v, v						
角 平 言 句	<p>数値レジスタ（vレジスタ）の符号、桁数（符号を含まない）を読み込みます。</p> <p>第1オペランド.....対象となるvレジスタを指定</p> <p>第2オペランド.....符号を格納するvレジスタを指定</p> <table border="1" data-bbox="778 891 1098 1066"> <thead> <tr> <th>符号</th> <th>レジスタ格納内容</th> </tr> </thead> <tbody> <tr> <td>+</td> <td>0</td> </tr> <tr> <td>-</td> <td>1</td> </tr> </tbody> </table> <p>第3オペランド.....桁数を格納するvレジスタを指定 符号は含みません。</p> <p><文例></p> <pre> MOVE -123,V1 ; - 1 2 3 → V 1 LEN V1,V2,V3 ; V 1 = - 1 2 3 1 → V 2 3 → V 3 </pre>	符号	レジスタ格納内容	+	0	-	1
符号	レジスタ格納内容						
+	0						
-	1						

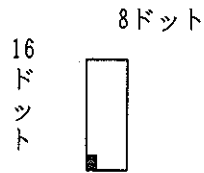
L O C A T D		ローケイトディ
機能	メッセージプレーン上の文字表示の開始位置指定 (ドット単位)	
書式	LOCATD Δ $\begin{bmatrix} v \\ i \end{bmatrix}$, $\begin{bmatrix} v \\ i \end{bmatrix}$	
角平 言兌	<p>第1オペランドの内容は「列」を、第2オペランドの内容は「行」をドット単位で指定し、以降のステップで「DISPD」命令により指定される、文字の表示開始位置を示します。</p> <p>一度、この命令で表示開始位置を指定しますと、「LOCATD」又は「DISPD」命令で再度表示開始位置を指定するか、最後の表示位置が489ドットになるまで、「DISPD」で指定された文字の内容を順番に表示します。489ドットを超えた文字は無視されます。</p> <p><文例> LOCATD V20,V30 ;行V30の内容のドット,列V20の内容のドットから文字表示を開始します。 DISPD "NG!!" ;前ステップの「LOCATD」で指定した表示位置から"NG!!"と表示します。</p> <p><参考> 1) 表示開始位置指定可能な領域 (ドット単位)</p> <p>第1オペランド 1 → 489 496</p> <p>第2オペランド 1 33 ↓ 472</p>	

2) 文字の構成

字以外の場合



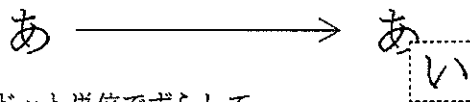
↑ 指定ポイント



↑ 指定ポイント

3) 描き込まれている文字の表示位置とドット単位でずらして描き込んだ場合、重ね描きされない部分は残ります。

例)



ドット単位でずらして
重ね描きをしますと

L O C A T E		ローケイト
機 能	メッセージプレーン上の文字表示の開始位置指定 (カラム単位)	
書 式	LOCATE Δ [v] , [v]	
角 注	<p>第1オペランドの内容は「列」を、第2オペランドの内容は「行」を、カラム単位で指定し、以降のステップで「DISP」命令により、指定される文字の表示開始位置を示します。</p> <p>一度この命令で表示開始位置を指定しますと、「LOCATE」又は「DISP」命令で再度表示開始位置を指定するか、表示の最後が61列を超えるまで、「DISP」で指定された表示内容を順番に表示します。</p> <p><文例> LOCATE V10,V15 ;行V15の内容、列V10の内容から文字表示を開始します。 DISP " OK!!" ;前ステップの「LOCATE」で指定した表示位置から、" OK!!" と表示します。</p> <p><参考> 1) 表示開始位置指定可能な領域 (カラム単位)</p> <p style="text-align: center;">第1オペランド (列)</p>	

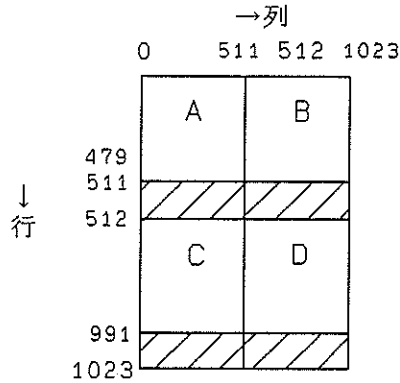
L S C A N	エ ル ス キ ャ ン
機 能	ラインチェッカの検査の実行
書 式	L S C A N Δ $\begin{bmatrix} v \\ i \end{bmatrix}$, $\begin{bmatrix} v \\ i \end{bmatrix}$, $\begin{bmatrix} (v) \\ (i) \end{bmatrix}$, $\begin{bmatrix} (v) \\ (i) \end{bmatrix}$ () は省略可です。
角 準 言 兌	<p>第1オペランドで指定されたラインチェッカNoから、第2オペランドで指定されたラインチェッカNoまでの検査を実行します。</p> <p>このとき、第3、第4オペランドでは、それぞれx方向、y方向の平行移動量を画素数で指定します。移動量が指定されているとき、「グループ選択」で指定されたグループの補正量と移動量の和の値だけラインチェッカを移動し、検査を実行します。</p> <p>第3、第4オペランドが「省略」または「0」が指定されているとき、「グループ選択」で指定されたグループの補正量だけ移動し、検査を実行します。また、ラインチェッカの検査条件は、品種モードで設定された条件（判定条件、グループ選択）により実行されます。</p> <p>グループの補正量による移動は行なわず、数値データ等の移動量のみを移動を行う場合、移動量の算出の際に位置補正の補正量の分を差引いておいてください。</p> <p>グループの補正量は、グループNO. で指定された位置補正の補正量によりますので、グループNO. を指定された位置補正が実行されないとき、補正量は電源を切らない限りメモリに残っているため、前回実行されたときの補正量により実行されます。</p> <p><文例> L S C A N 1,5,30,20 ;ラインチェッカNo1~No5をグループの補正量により補正後、(Δx,Δy)=(30,20)だけ平行移動し、検査を実行します。</p>

角 評 言 兌

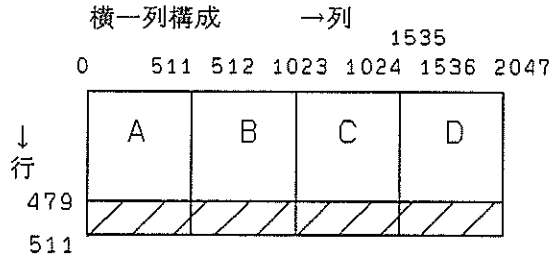
<参考>

- 1) 品種モードで「チェッカパターン表示選択」を「位置補正に追従して表示」を選択している場合、チェッカパターンの消去・描き込みに多少時間を要します。それでは問題のある場合、「固定位置に表示」又は「表示しない」を選択して時間を短縮してください。
- 2) 移動量を入力する際、画面座標は右図のようになっていますので、移動方向の符号に御注意ください。
画面構成は実行ブロックでの設定によります。

田の字型構成



横一列構成



<例> 位置補正NO.1 (水平設定時) エッジ位置 : 100 補正量 : $\Delta x = 0$
 ラインNO.1 (110,10) - (110,50) の直線

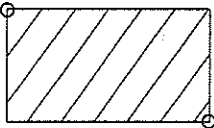
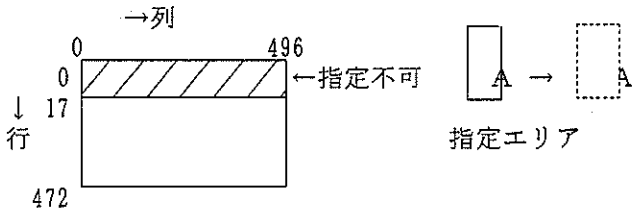
↓

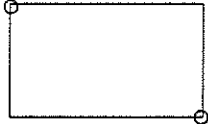
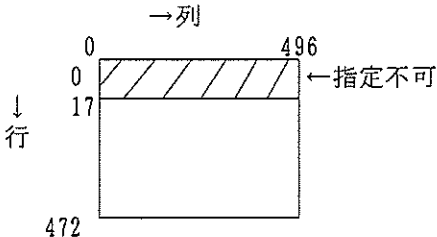
ワーク移動後画像取込み ISCAN 1,1
 LSCAN 1,1
 を実行

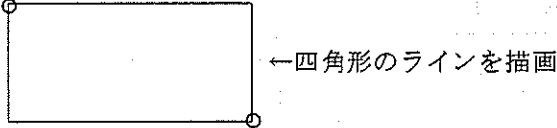
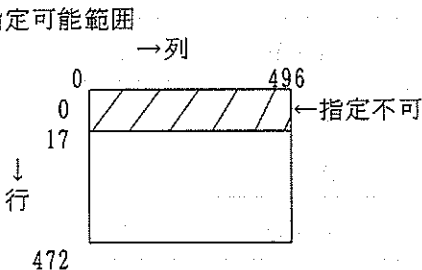
エッジ位置 : 110 補正量 : $\Delta x = 10$
 ラインNO.1 (120,10) - (120,50) に移動
 ↓ その後 LSCAN 1,1,10を実施
 ラインNO.1は, (130,10) - (130,50) に移動

$$\text{※座標計算} \quad \underline{110} + \underline{10} + \underline{10} = \underline{130}$$

↑ ↑ ↑ ↑ ↑
 (この例の場合 : x 座標) 設定値 補正量 移動量 新座標

MLCLR	メッセージ クリア
機 能	メッセージプレーンの部分消去
書 式	MLCLR Δ $\left[\begin{smallmatrix} v \\ i \end{smallmatrix} \right], \left[\begin{smallmatrix} v \\ i \end{smallmatrix} \right], \left[\begin{smallmatrix} v \\ i \end{smallmatrix} \right], \left[\begin{smallmatrix} v \\ i \end{smallmatrix} \right]$
角 解 説	<p>メッセージプレーンに描かれている内容で、始点[第1オペランド(列)、第2オペランド(行)]、終点[第3オペランド(列)、第4オペランド(行)]で指定された四角形の範囲内を消去します。座標の指定は、画素単位で行ないます。画面構成は下記の通りですので、座標の指定は、MIN(0,17),MAX(496,472)です。</p> <p>始点 [第1オペランド(列)、第2オペランド(行)]</p>  <p>この範囲を消去します。</p> <p>終点 [第3オペランド(列)、第4オペランド(行)]</p> <p>注) 指定したエリア内に文字がある場合、同時に消去されます。(文字もメッセージプレーンに描画されるため。)</p> <p>指定可能範囲</p>  <p><文例> MLCLR 100,120,300,350 ;始点(100,120)、終点(300,350)で指定された四角形の範囲内に描かれている内容を消去します。</p> <p>(参考) メッセージプレーンとは 本システムのメモリ表示用の画面は全部で4面(I, M, C, P)用意されていますが、そのうちで主に文字を描画する画面で、輝度が高い画面を示します。”SHOW”コマンドで指定する記号では’M’で指定されます。</p>

M L O F F	メッセージラインオフ
機能	メッセージプレーンのラインの消去
書式	MLOFF Δ $\left[\begin{smallmatrix} v \\ i \end{smallmatrix} \right], \left[\begin{smallmatrix} v \\ i \end{smallmatrix} \right], \left[\begin{smallmatrix} v \\ i \end{smallmatrix} \right], \left[\begin{smallmatrix} v \\ i \end{smallmatrix} \right]$
角解言兌	<p>メッセージプレーン上の始点 [第1オペランド (列)、第2オペランド (行)] 終点 [第3オペランド (列)、第4オペランド (行)] で指定された四角形のラインを消去します。 座標の指定は画素単位で行ないます。画面構成は下記の通りですので座標の指定は、MIN(0,17)、MAX(496,472) です。</p> <p>始点 [第1オペランド (列)、第2オペランド (行)]</p>  <p>←四角形のラインを消去</p> <p>終点 [第3オペランド (列)、第4オペランド (行)]</p> <p>注) 指定したライン上に文字がある場合ライン上の部分が消去されます。(文字もメッセージプレーンに描画されるため)</p> <p>指定可能範囲</p>  <p>←指定不可</p> <p><文例> MLOFF 100,120,300,350 ;始点(100,120)、終点(300,350) で指定された四角形のラインを消去します。</p> <p><参考>メッセージプレーンとは 本システムのメモリ表示用の画面は全部で4面 (I, M, C, P) 用意されていますが、そのうちで主に文字を描画する画面で、輝度が高い画面を示します。”SHOW”コマンドで指定する記号では’M’で指定されます。</p>

MLON	メッセージラインオン
機能	メッセージプレーンへのラインの描画
書式	MLON Δ $\begin{bmatrix} v \\ i \end{bmatrix}$, $\begin{bmatrix} v \\ i \end{bmatrix}$, $\begin{bmatrix} v \\ i \end{bmatrix}$, $\begin{bmatrix} v \\ i \end{bmatrix}$
角解言兌	<p>メッセージプレーンに、始点 [第1オペランド (列)、第2オペランド (行)]、終点 [第3オペランド (列)、第4オペランド (行)] で指定された四角形のラインを描画します。座標の指定はドット単位で行ないます。画面構成は下記の通りですので座標の指定は、MIN(0,17), MAX(496,472) です。</p> <p>始点 [第1オペランド (列)、第2オペランド (行)]</p>  <p>終点 [第3オペランド (列)、第4オペランド (行)]</p> <p>指定可能範囲</p>  <p><文例> MLON 100,120,300,350 ;始点(100,120)、終点(300,350)で指定された四角形のラインを描画します。</p> <p>(参考) メッセージプレーンとは 本システムのモニター表示用の画面は全部で4面 (I, M, C, P) 用意されていますが、そのうちで主に文字を描画する画面で、輝度が高い画面を示します。”SHOW” コマンドで指定する記号では 'M' で指定されます。</p>

MONT

モント

機能

画面表示メモリエリアの指定

書式

MONT Δ $\left[\begin{array}{l} a \\ d \end{array} \right]$

角字 記号

画面表示のメモリエリアを記号 (d) で指定します。画面構成には「田の字型構成」と「横一列構成」がありますのでそれぞれ下表により記号で指定してください。この記号以外を指定すると、「エラー」になりますのでご注意ください。

但し、生画像表示、リアルタイム2値化画像表示の場合、画面メモリを2つ以上またがって表示することはできません。

1) 田の字型構成

A	B
C	D

メモリエリア	記号: d
A	A
B	B
C	C
D	D
A-B	E
C-D	F
A-C	G
B-D	H
A-D, B-C	I

注) 「A-B」とはAとBの中間を示します。

角 解 説

2) 横一列型構成

A	B	C	D
---	---	---	---

メモリエリア	記号 : d
A	A
B	B
C	C
D	D
A-B	E
B-C	F
C-D	G

<文例> 田の字型構成の場合

MONT "B" ; Bメモリエリアを画面表示します。

MOVE "F",A1,1 ; 'F'をA1に格納します。

MONT A1 ; F(C-D間)のメモリエリアを画面表示します

MOVE		ムーブ
機能	レジスタ間転送	
書式	<p>1) 数値レジスタ→数値レジスタ 直接数値 数値結果 判定結果</p> $\text{MOVE } \Delta \begin{pmatrix} v \\ i \\ j \\ h \\ B \\ S \end{pmatrix}, v$ <p>2) 数値レジスタ→文字レジスタ 数値結果 判定結果</p> $\text{MOVE } \Delta \begin{pmatrix} v \\ i \\ j \\ h \\ B \\ S \end{pmatrix}, a, \begin{pmatrix} v \\ i \end{pmatrix}$ <p>3) 文字レジスタ→文字レジスタ 文字列</p> $\text{MOVE } \Delta \begin{pmatrix} a \\ G \\ \text{" " } \end{pmatrix}, a, \begin{pmatrix} v \\ i \end{pmatrix}$ <p>4) 文字レジスタ→数値レジスタ (数値)</p> $\text{MOVE } \Delta a, v, \begin{pmatrix} v \\ i \end{pmatrix}$	
角評 言兌	<p>1) 数値レジスタ→数値レジスタ 直接数値 数値結果 判定結果</p> <p>第1オペランドで指定された内容 (v又はi) を第2オペランドで指定された v レジスタに転送します。</p> <p><文例> MOVE 100, V1 ; 100→V1 MOVE V1, V2 ; V1→V2 MOVE DW1, V3 ; (W1の判定結果) →V3 MOVE OX25, V4 ; 25(Hex)→37 (10進数) →V4 MOVE DS3, V5 ; S3→V5 MOVE DB1, V6 ; B1→V6</p> <p>2) 数値レジスタ→文字レジスタ 数値結果 判定結果</p> <p>第1オペランドで指定されたレジスタの内容を第3オペランドで指定された桁数に相当する個数だけ、第2オペランドで指定された a レジスタを先頭に転送します。</p> $\text{MOVE } \Delta V, A_n, m; (1) \rightarrow A[n], (2) \rightarrow A[n+1], (3) \rightarrow A[n+2],$ $(4) \rightarrow A[n+3] \cdots (m) \rightarrow A[n+m-1]$ <p>()の数字は v レジスタの内容で桁数の大きい方からの順番をさします。 []は格納される a レジスタのNO. をさします。</p>	

4) 第1オペランドの内容(数値)を第3オペランドで指定した桁数だけ、第2オペランドで指定するvレジスタに転送します。

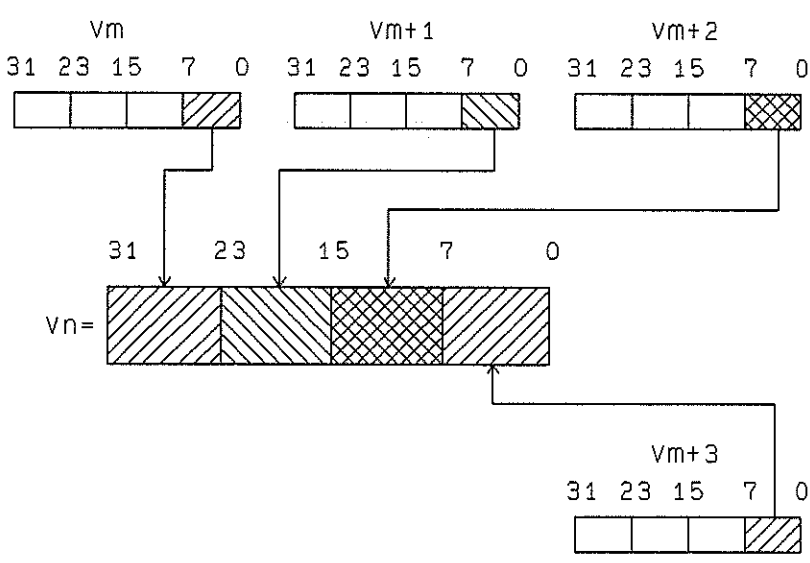
```
<文例> MOVE "128",A1,3 ;'1'(31H)→A1
          ;'2'(32H)→A2
          ;'8'(38H)→A3
          MOVE A1,V1,3 ;128→V1
```

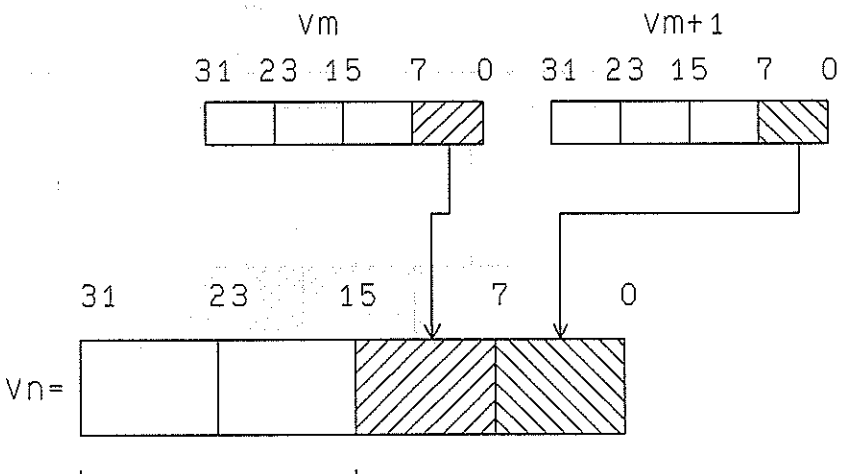
(注意) 転送するaレジスタの最初の文字は符合(+、-)または、数字、空白文字のみ判断し転送されます。
空白文字は、最初から何文字あっても、判断され無視されます
初めて符合(+、-)または、数字が判断され以降の文字を数字と判断し転送されます。この時、数字以外の文字があると、その文字以降は、無視されます。

```
<文例①> MOVE "123A4",A1,5; 1 →A1
          ; 2 →A2
          ; 3 →A3
          ; A →A4
          ; 4 →A5
          MOVE A1,V1,5 ; 123 →V1
```

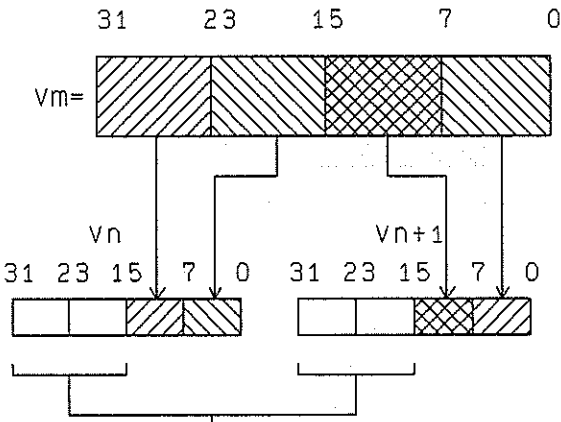
```
<文例②> MOVE -123,V1 ; -123→V1
          MOVE V1,A1,6 ; -123→A1~A6
          (空白)→A1
          (空白)→A2
          "-"→A3
          "1"→A4
          "2"→A5
          "3"→A6
          MOVE A1,V1,6 ; -123→V1
```

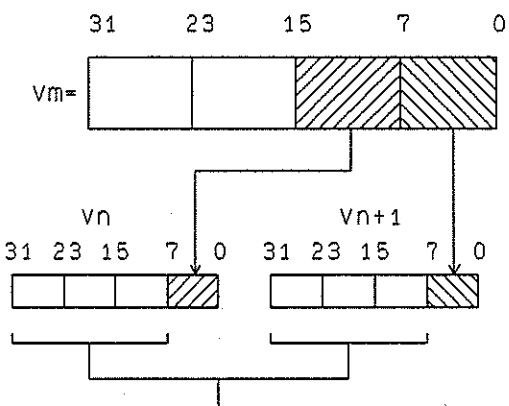
```
<文例③> MOVE "-123A4",A1,6; "-123A4"→A1~A6
          MOVE A1,V1,6 ; -123→V1
```

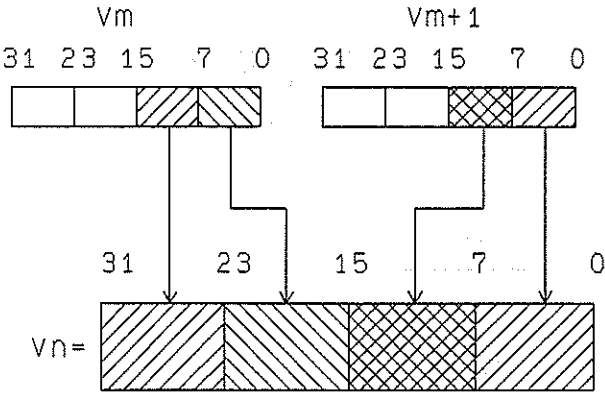
MOVEBL	ムーブ バイト ロングワード
機能	レジスタ結合 (byte → long word)
書式	MOVEBL Δ v, v
角解 言	<p>第1オペランドで指定したvレジスタを先頭に、4つのvレジスタの下位8ビットを、第2オペランドで指定したvレジスタに格納します。</p> <p>格納は、上位8ビットから順に行ないます。</p> <p><文例> MOVEBL V40, V50 ;V40を先頭に4つのvレジスタの下位8ビットをV50に格納します。</p> <p>注) この実行は符号なしの実行となりますので、「負の整数」を取扱う場合は、御注意ください。 MOVEBL Δ Vm, Vn ↓</p>  <p>この時、Vm~Vm+3の値は、保持されています。</p>

MOVEBW	ムーブ バイト / ワード
機能	レジスタ結合 (byte → word)
書式	MOVEBW Δ v, v
角解 言	<p>第1オペランドで指定したvレジスタを先頭に、2つのvレジスタの下位8ビットを、第2オペランドで指定したvレジスタの下位16ビットに格納します。このとき、上位16ビットは'0'にクリアされます。</p> <p>格納は、16ビットのうち上位8ビットから順に行ないます。</p> <p><文例> MOVEBW V30,V40 ; V30を先頭に2つのvレジスタの下位8ビットをV40の下位16ビットに格納します。</p> <p>注) この実行は符号なしの実行となりますので、「負の整数」を取り扱う場合は御注意ください。 MOVEBW Δ Vm,Vn ↓</p>  <p>この部分はクリアされます</p> <p>この時、Vm~Vm+1の値は、保持されます。</p>

MOVELB	ムーブ ロングワード / バイト
機能	レジスタ分割 (long word → byte)
書式	MOVELB Δ v, v
角軍 説	<p>第1オペランドで指定したvレジスタを8ビットずつ4分割し、上位から順に第2オペランドで指定したvレジスタを先頭に4つのレジスタに格納します。格納されるのは、指定レジスタの下位8ビットです。このとき、上位24ビットは'0'にクリアされます。</p> <p><文例> MOVELB V100,V150 ; V100を8ビットずつ4分割し、V150を先頭に4つのvレジスタに、上位8ビットから順に格納します。</p> <p>注) この実行は符号なしの実行となりますので、「負の整数」を取り扱う場合は、御注意ください。 MOVELB Δ Vm,Vn ↓</p> <p style="text-align: right;">この時、Vmの値は、保持されます。</p>

MOVELW	ムーブ ロングワード ワード
機能	レジスタ分割 (long word → word)
書式	MOVELW Δ v, v
角解 言兌	<p>第1オペランドで指定したvレジスタを16ビットずつ2分割し、上位から順に第2オペランドで指定したvレジスタを先頭に、2つのレジスタにそれぞれ格納します。格納されるのは、指定レジスタの下位16ビットです。 このとき上位16ビットは'0'にクリアされます。</p> <p><文例> MOVELW V10,V20 ; V10を16ビットずつ2分割し上位から順に、V20を先頭に2つのvレジスタに格納します。</p> <p>注) この実行は符号なしの実行となりますので、「負の整数」を取扱う場合は、御注意ください。 MOVELW Δ Vm,Vn ↓</p>  <p>'0'にクリアされます</p> <p>この時、Vmの値は、保持されます。</p>

MOVEWB	ムーブ ワード バイト
機能	レジスタ分割 (word → byte)
書式	MOVEWB Δ v, v
角解 言説	<p>第1オペランドで指定したvレジスタの下位16ビットを2分割し、上位8ビットを第2オペランドで指定したvレジスタに、下位8ビットを次の番号のvレジスタにそれぞれ格納します。格納されるのは、指定レジスタの下位8ビットです。</p> <p>このとき上位24ビットは'0'にクリアされます。</p> <p><文例> MOVEWB V10,V5 ; V10の下位16ビットを2つに分割し 上位8ビットをV5に、下位8ビット をV6に格納します。</p> <p>注) この実行は符号なしの実行となりますので、「負の整数」を取扱う場合は、御注意ください。 MOVEWB Δ Vm,Vn ↓</p>  <p>'0'にクリアされます</p> <p>この時、Vmの値は保持されます</p>

MOVEWL	ムーブ ワード ロングワード
機 能	レジスタ結合 (word → long word)
書 式	MOVEWL Δ v, v
角 解 説	<p>第1オペランドで指定したvレジスタを先頭に、2つのvレジスタの下位16ビットを、第2オペランドで指定したvレジスタに格納します。</p> <p>格納は、上位16ビットから順に行ないます。</p> <p><文例> MOVEWL V20, V40 ; V20を先頭に2つのvレジスタ(V20とV21)の下位16ビットをV40に格納します。</p> <p>注) この実行は符号なしの実行となりますので、「負の整数」を取扱う場合は、御注意ください。 MOVEWL Δ Vm, Vn ↓</p>  <p>この時、Vm, Vm+1は、保持されます。</p>

MSCAN	エム スキャン
機能	パターンマッチングの検査の実行
書式	MSCAN $\Delta \begin{pmatrix} v \\ i \end{pmatrix}, \begin{pmatrix} v \\ i \end{pmatrix}, \begin{pmatrix} (v) \\ (i) \end{pmatrix}, \begin{pmatrix} (v) \\ (i) \end{pmatrix}$ () は省略可です。
角解 言兌	<p>第1オペランドで指定されたパターンマッチングチェッカNO.から、第2オペランドで指定されたパターンマッチングチェッカNO.までの検査を実行します。</p> <p>このとき、第3、第4オペランドでは、それぞれx方向、y方向の平行移動量を画素数で指定します。移動量が指定されているとき、「グループ選択」で指定されたグループの補正量と移動量の和の値だけパターンマッチングチェッカを移動し、検査を実行します。</p> <p>第3、第4オペランドが「省略」又は「0」が指定されているとき「グループ選択」で指定されたグループの補正量だけ移動し、検査を実行します。また、パターンマッチングチェッカの検査条件（判定条件、グループ選択）は品種モードで設定された条件により実行されます。</p> <p>グループの補正量による移動は行なわず、数値データ等の移動量のみを移動を行なう場合、移動量の算出の際に位置補正の補正量の分を差引いておいてください。</p> <p>グループの補正量は、グループNO.で指定された位置補正の補正量によりますので、グループNO.を指定された位置補正が実行されないとき、補正量はメモリに電源を切らない限り残っているため、前回実行されたときの補正量により実行されます。</p> <p><文例> MSCAN 3,6,120,150; パターンマッチングのチェッカ No3~No6を位置補正で補正後、 $(\Delta x, \Delta y)=(120,150)$だけ平行移動し、検査を実行します。</p>

MUL		マルチプライ
機 能	レジスタ間乗算	
書 式	$\text{MUL } \Delta \quad \left[\begin{array}{c} v \\ i \end{array} \right], \left[\begin{array}{c} v \\ i \end{array} \right], v$	
角 注	<p>第1オペランドの内容と第2オペランドの内容を乗算し、第3オペランドの指定するvレジスタに格納します。</p> <p>乗算結果の値の範囲は、-2,147,483,648~2,147,483,647 です。オーバーフローした時はエラーとなり、オーバーフローフラグがONされます。この時、第3オペランドの値は、意味をなさない値となりますので、ご注意ください。</p> <p><文例> MUL V20,V21,V100 ; V20×V21→V100</p>	

NOT	ノット
機能	ビット反転
書式	NOT Δ v
角解 言兌	<p>第1オペランドで指定されたvレジスタのビットを全て反転します。</p> <p><文例> NOT V100 ; V100レジスタのビットを全て反転します。</p> <p>(参考) 反転とは? 「0」を「1」、「1」を「0」にすることです。</p> <pre> [NOT] 0 ----> 1 1 ----> 0 </pre> <p>(例) MOVE Δ 0X80, V1 NOT Δ V1 という命令を実行するとき、レジスタの内容を4ビットずつ16進数表現すると、</p> <pre> 31 23 15 7 0 V1= [0 0 0 0 0 0 8 0] [NOT] V1= [F F F F F F 7 F] </pre>

OR	オア															
機能	論理和															
書式	$\text{OR } \Delta \left[\begin{array}{c} v \\ i \\ h \end{array} \right], \left[\begin{array}{c} v \\ i \\ h \end{array} \right], v$															
角解 言	<p>第1オペランドの内容と、第2オペランドの内容との各ビット毎の論理和を行ない、演算結果を第3オペランドで指定したvレジスタに格納します。</p> <p>論理和の演算を下表に示します。</p> <table border="1" data-bbox="507 913 1171 1240"> <thead> <tr> <th>第1オペランドの内容</th> <th>第2オペランドの内容</th> <th>第3オペランドに格納される内容</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table> <p><文例></p> <pre> MOVE 1,V1 ; 1→V1 MOVE 0,V2 ; 0→V2 OR V1,V2,V3 ; V1 or V2 → V3 1 or 0 → 1 → V3 </pre> <p>OR：論理和は、各レジスタの各ビット間の論理和となります。</p>	第1オペランドの内容	第2オペランドの内容	第3オペランドに格納される内容	0	0	0	0	1	1	1	0	1	1	1	1
第1オペランドの内容	第2オペランドの内容	第3オペランドに格納される内容														
0	0	0														
0	1	1														
1	0	1														
1	1	1														

OVOFF

オーバーフローフラグオフ

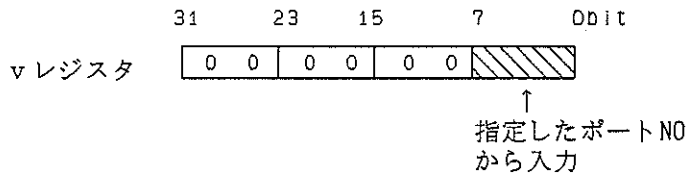
機能	OVFLG (オーバーフローフラグ) のOFF																		
書式	OVOFF (オペランドなし)																		
角解 説	<p>OVFLG (オーバーフローフラグ) をOFFにします。</p> <p>OVFLGがONになるのは、</p> <ul style="list-style-type: none"> ① プログラムエラーが発生した時 ② "C-OUT" 命令実行時、データエラーが発生した時の2通りがあります。 <p>このコマンドは、①の場合に"OVFLG"がON/OFFであるかを外部機器などに読み込んだ時に"OVFLG"を"OFF"にするためのコマンドです。</p> <p>②の場合には、"C-OUT" 命令実行時、"OVFLG" は一度"OFF" され実行前の状態(ON/OFF)は記憶されています。</p> <p>"C-OUT" 実行中、数値演算結果にエラーがあるかどうか("OVFLG" がON/OFFか)は、外部機器で数値演算データを読み込む時に同時に読み込んでください。"C-OUT" 命令実行後、"OVFLG" は実行前の状態に戻ります。</p> <p>(参考) "OVFLG" はパラレル出力ポートNo. 1のビットです。</p> <table border="1" data-bbox="383 1272 1141 1422"> <tr> <td>ビット</td> <td>7</td> <td>6</td> <td>5</td> <td>4</td> <td>3</td> <td>2</td> <td>1</td> <td>0</td> </tr> <tr> <td>使用内容</td> <td>BATRY</td> <td>ERROR</td> <td>READY</td> <td>STROB</td> <td>REND</td> <td>OVFLG</td> <td>PW-FAIL 注1</td> <td>FLASH</td> </tr> </table> <p><文例></p> <p>OVOFF ; "OVFLG" を"OFF" します。</p> <p>注1: PW-FAILは、Ver 3以降のみ対応です。</p>	ビット	7	6	5	4	3	2	1	0	使用内容	BATRY	ERROR	READY	STROB	REND	OVFLG	PW-FAIL 注1	FLASH
ビット	7	6	5	4	3	2	1	0											
使用内容	BATRY	ERROR	READY	STROB	REND	OVFLG	PW-FAIL 注1	FLASH											

P - I N		パラレルイン																					
機 能	パラレル入力																						
書 式	P - I N Δ $\left(\begin{matrix} v \\ i \end{matrix} \right), v$																						
角 平 言 兌	<p>第1オペランドではパラレル入力するポートNoを示す番号を指定し、第2オペランドで指定するvレジスタに格納します。</p> <p>パラレル入力用のポートは、1ポート8ビット構成で、合計3ポート用意されています。</p> <p>第1オペランドで指定する番号とポートNoとの関係を下表に示します。</p> <p>第1オペランドで「1」、「2」または「3」を指定したときの入力方法はvレジスタの下位8ビットに格納されます。</p> <p>第1オペランドで「4」または「5」を指定したときの入力方法はvレジスタの下位16ビットのうち、上位8ビット、下位8ビットに分けると、上位8ビットにはポートNoの大きい方が、下位8ビットにはポートNoの小さい方が格納されます。</p> <p>また、第1オペランドで「6」を指定したときの入力方法は、vレジスタの下位24ビットのうち、8ビットずつに分けると、上位8ビットから順に、ポートNo 3、2、1の順に格納されます。</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>第1オペランドで指定する番号</th> <th>使用するポートNo</th> <th>パラレル入力する総ビット数</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">8ビット</td> </tr> <tr> <td style="text-align: center;">2</td> <td style="text-align: center;">2</td> <td style="text-align: center;">8ビット</td> </tr> <tr> <td style="text-align: center;">3</td> <td style="text-align: center;">3</td> <td style="text-align: center;">8ビット</td> </tr> <tr> <td style="text-align: center;">4</td> <td style="text-align: center;">1,2</td> <td style="text-align: center;">16ビット</td> </tr> <tr> <td style="text-align: center;">5</td> <td style="text-align: center;">2,3</td> <td style="text-align: center;">16ビット</td> </tr> <tr> <td style="text-align: center;">6</td> <td style="text-align: center;">1,2,3</td> <td style="text-align: center;">24ビット</td> </tr> </tbody> </table>		第1オペランドで指定する番号	使用するポートNo	パラレル入力する総ビット数	1	1	8ビット	2	2	8ビット	3	3	8ビット	4	1,2	16ビット	5	2,3	16ビット	6	1,2,3	24ビット
第1オペランドで指定する番号	使用するポートNo	パラレル入力する総ビット数																					
1	1	8ビット																					
2	2	8ビット																					
3	3	8ビット																					
4	1,2	16ビット																					
5	2,3	16ビット																					
6	1,2,3	24ビット																					

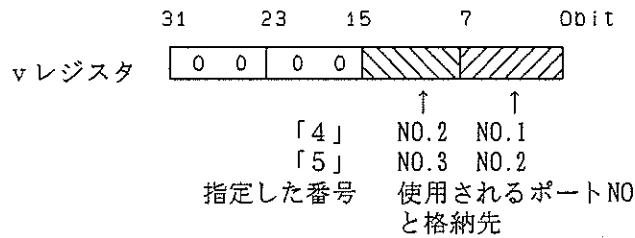
角 評 言 兌

<文例> P-IN 4,V1 ; ポートNO1,2からパラレル入力し、V1に格納します。

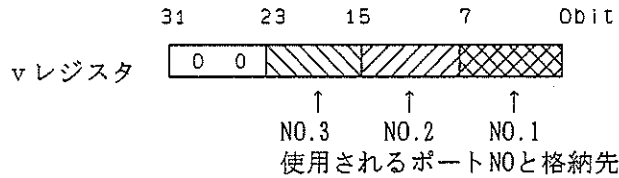
<参考> 1) 第1オペランドで「1」「2」または「3」を指定したときの入力方法



2) 第1オペランドで「4」または「5」を指定したときの入力方法



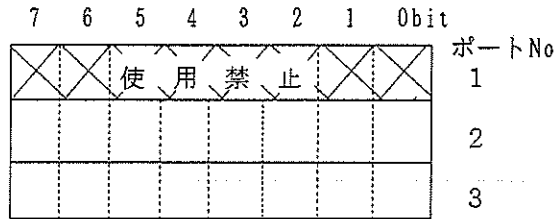
3) 第1オペランドで「6」を指定したときの入力方法



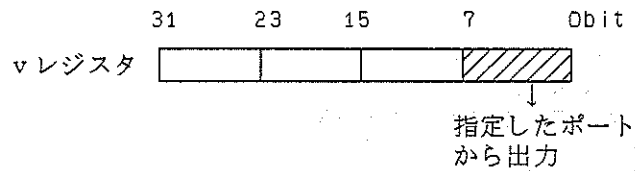
P-OUT		パラレルアウト												
機能	パラレル出力													
書式	P-OUT Δ $\left[\begin{matrix} v \\ i \end{matrix} \right], v$													
解説	<p>第1オペランドではパラレル出力するポートNoを示す番号を指定し、第2オペランドで指定したvレジスタの内容をパラレル出力します。パラレル出力用のポートは、1ポート8ビット構成で、合計3ポート用意されていますが、ポートNo1は固定出力用のため使用禁止になっています。第1オペランドで指定する番号とポートNoとの関係を下表に示します。第1オペランドで「1」又は「2」を指定したときの出力方法はvレジスタの32ビットのうち、下位8ビットを指定したポートから出力されます。第1オペランドで「3」を指定したときの出力方法はvレジスタの下位16ビットのうち、上位8ビットがNo3から、下位8ビットがNo2から出力されます。vレジスタの32ビットを出力させたいときは、「MOVELW」命令で16ビットのvレジスタに分けて出力してください。</p> <table border="1" data-bbox="475 1126 1027 1388"> <thead> <tr> <th>第1オペランドで指定する番号</th> <th>使用するポートNo</th> <th>パラレル出力する総ビット数</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>2</td> <td>8ビット</td> </tr> <tr> <td>2</td> <td>3</td> <td>8ビット</td> </tr> <tr> <td>3</td> <td>2、3</td> <td>16ビット</td> </tr> </tbody> </table> <p><文例> P-OUT 2,V100 ;ポートNO3からV100の内容をパラレルアウトします。</p>		第1オペランドで指定する番号	使用するポートNo	パラレル出力する総ビット数	1	2	8ビット	2	3	8ビット	3	2、3	16ビット
第1オペランドで指定する番号	使用するポートNo	パラレル出力する総ビット数												
1	2	8ビット												
2	3	8ビット												
3	2、3	16ビット												

角 評 言

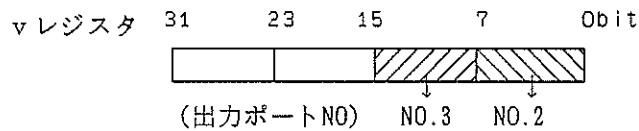
<参考> 1) パラレル出力用ポート構成



2) 第1オペランドで「1」または「2」を指定したときの出力方法

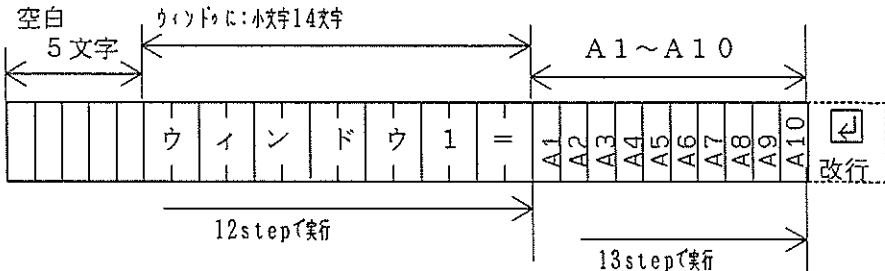


3) 第1オペランドで「3」を指定したときの出力方法



(注意) 負の数値を出力する際の注意事項

3 2ビットを出力しない場合に注意が必要です。
3 1ビット目が '1' (負の値) であるかをイメージチェッカ
内で判断しその結果を別に出力して、外部機器側で読み込んで
ください。

P R I N T	プリント						
機能	プリンタへの出力 (パラレル出力ポート)						
書式	$\text{PRINT } \Delta \left[\begin{matrix} v \\ i \end{matrix} \right], \left[\begin{matrix} v \\ i \end{matrix} \right], \left[\begin{matrix} a \\ \text{" " } \end{matrix} \right], \left[\begin{matrix} v \\ i \end{matrix} \right]$						
角解 言説	<p>パラレル入出力ポート (EXT-IN, EXT-OUT) からセントロニクスプリンタへ出力します。</p> <p>第1オペランド.....プリントアウトモードを指定</p> <table border="1" data-bbox="577 806 1256 974"> <thead> <tr> <th>オペランド内容</th> <th>プリントアウトモード</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>改行コードを付けずに出力</td> </tr> <tr> <td>1</td> <td>改行コード (OXD="CR") を付けて出力</td> </tr> </tbody> </table> <p>第2オペランド.....前置するスペース (空白) の数を指定 (0,1,2,3.....)</p> <p>第3オペランド.....プリントアウトする a レジスタの開始No か文字列を指定</p> <p>第4オペランド.....プリントアウトする a レジスタまたは文字列の個数を指定 (1,2,3.....)</p> <p>実際にプリンタに文字が打ち出されるのは改行コードを受け取るまでプリンタ内部のバッファに蓄積され続け、改行コードを受け取って文字が打ち出されます。また、プリンタ内部のバッファに1行分 (約80文字: 通常) の文字が蓄積されると、自動的に文字がプリントアウトされます。</p> <p><文例> ウィンドウ1の計測データのプリントアウト</p> <pre> 10: MOVE CW1,V1 ウィンドウ1のデータをV1に格納 V1←CW1 11: MOVE V1,A1,10 V1をA1~A10に格納 12: PRINT 0.5,"ウィンドウ1=",14 プリンタに"ウィンドウ1="とカウントした後 13: PRINT 1,0,A1,10 計測値をカウントし、改行します。 </pre> 	オペランド内容	プリントアウトモード	0	改行コードを付けずに出力	1	改行コード (OXD="CR") を付けて出力
オペランド内容	プリントアウトモード						
0	改行コードを付けずに出力						
1	改行コード (OXD="CR") を付けて出力						

角 平 言 兌

注) 本システムのプリンタに関する入出力方法は「セントロニクス仕様」に準拠しています。

注) 本システムにより正常に動作するプリンタは

NEC PC-PR201V
 エプソン VP-1000
 ブラザー M-1924P

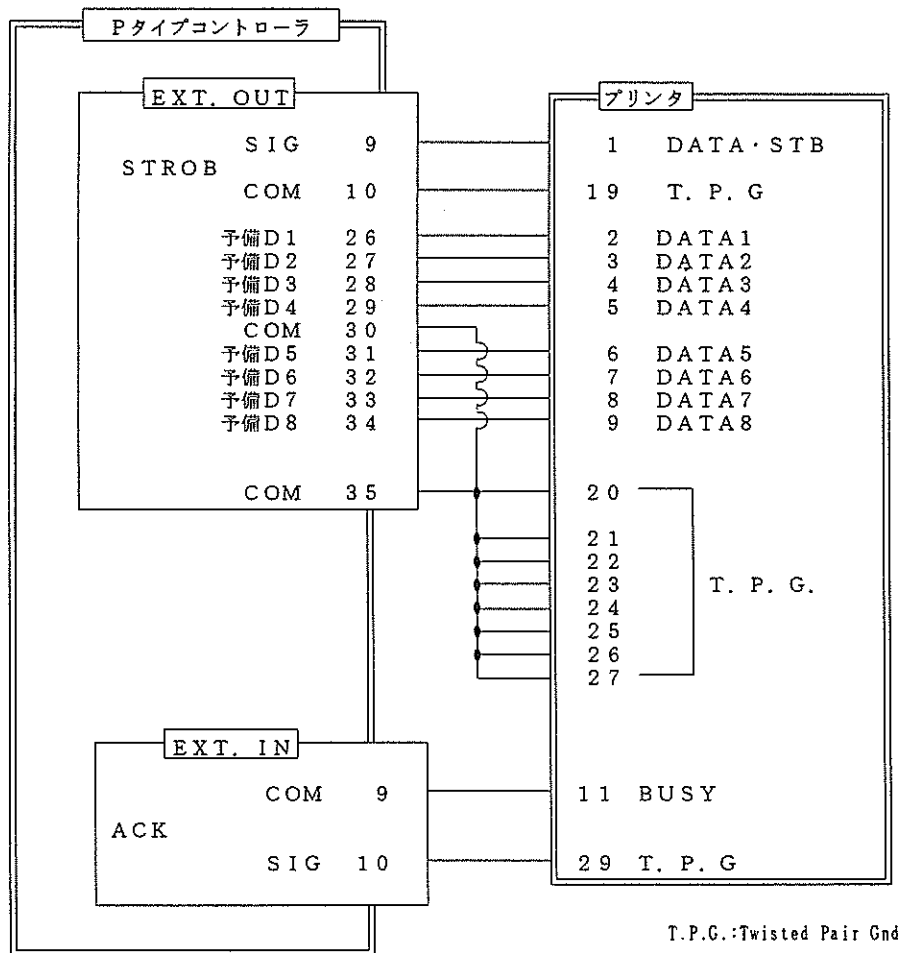
と同等の機能を有するプリンタに限ります。

但し、グラフィック文字、コントローラがサポートしていないJIS漢字第1水準以外をプリントアウトする場合は、保障しかねますので、ご了承ください。

注) 漢字キャラクターの制御コードが、以下であることをプリンタにて確認ください。

- ・漢字 in : "ESC" + "K"
- ・漢字 out : "ESC" + "H"

注) プリンタとの結線は、下図を参照ください。



尚、プリントアウトに時間を要する場合は、プリンタバッファを使用することをお勧めいたします。

ケーブルは、ユーザーにて自作願います。

<注意>

セントロニクス仕様のプリンタでしかプログラムリストのプリントアウトはできません。シリアル (RS232C) 仕様のプリンタでは、プログラムリストのプリントアウトはできません。

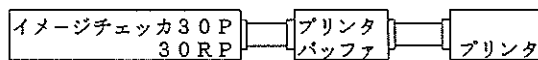


図4-6-3のケーブル

プリンタバッファ用ケーブル

R B I T	リセットビット
機 能	ビットリセット
書 式	RBIT Δ $\left[\begin{matrix} v \\ i \end{matrix} \right], v$
角 評 言 兌	<p>指定レジスタ（第2オペランド）の指定ビット（第1オペランド）をリセット（=0）します。</p> <p>vレジスタは32ビット構成です。 （LSB→0ビット、MSB→31ビット）</p> <p><文例> RBIT 3,V2 ; V2の3ビットを”0”とします。</p> <p>なお、第1オペランドが”0～31”以外の場合は、エラーとなり、オーバーフローフラグがONされます。 この時、第2オペランドの値は、変化しません。</p>

READ		リード						
機能	メモリに画像の取り込み							
書式	READ Δ $\left[\begin{array}{c} a \\ \text{" ABCD"} \end{array} \right], \left[\begin{array}{c} (v) \\ (i) \\ (h) \end{array} \right]$ ……()は省略可							
角解 言説	<p>第1オペランド……取り込みメモリ画像を指定 第2オペランド……画面密着を実行するか、しないかを選択。</p> <table border="1"> <thead> <tr> <th>オペランド内容</th> <th>画面密着モード</th> </tr> </thead> <tbody> <tr> <td>0 (省略)</td> <td>画面密着を実行する</td> </tr> <tr> <td>1</td> <td>画面密着を実行しない</td> </tr> </tbody> </table> <p>・画面密着を実行するを選択した場合の動作は以下の順で動作します。 1. 画像取り込みの実行 2. 画面密着の実行</p> <p>第1オペランドで指定したメモリに画像を取り込みます。指定の方法は、画像を取り込むメモリの記号を指定して、取り込まないメモリは、「*」を指定します。また、指定の順番は、A、B、C、Dの順に指定します。複数のカメラから画像を取り込む際、後面パネルの「カメラ切替えスイッチ」も切替えておいてください。このとき、メモリ構成はイメージチェッカ30と同じです。又、「画面密着有り」を設定されていますと、同時に実行されます。(この例としまして、「画面密着有り」を設定し、READ "A***" を実行しますと、指定メモリに画像を取り込んで画面密着を行うこととなります。) aレジスタで指定した場合は、指定された番号から4個目までを無条件に読み込んで実行します。 この命令を実行するときの2値化レベルはこの命令の前に"THRES" コマンドで設定した値、もしくは"ESCAN" コマンドを実行した時に決定した値のうち、最後に実行された値になります。 また、いずれの実行もされていない場合は、「2値化レベルのマニュアル設定」の設定値が採用されます。 (READ Δ An ; READ Δ "An An+1 An+2 An+3") ↑ ↑ ↑ ↑ メモリ A B C D</p> <p><注意> 実行ブロックの「5. 画面密着」で「画面密着あり」を指定している時のみ、第2オペランドで指定した"画面密着を実行する"が有効になります。</p>		オペランド内容	画面密着モード	0 (省略)	画面密着を実行する	1	画面密着を実行しない
オペランド内容	画面密着モード							
0 (省略)	画面密着を実行する							
1	画面密着を実行しない							

角 注 言 究

<文例> READ "A***",1 ; メモリ A に画像を画像密着を実行せずに取り込みます。
READ A10 ; A10 から A13 までの内容のメモリに画像密着を実行
して画像を取り込みます。
(Ex A10="*", A11='B', A12='C', A13='*' のとき READ△A10 と READ△"*BC*" は同じ内容です。)

- (注意)
- ・ 品種モードで選択した「照明モード」は無視されます。
 - ・ 品種モードにて「2 値化レベルの自動設定」を設定しましても、このコマンドでは実行されません。
但し、「THRES」コマンドを設定していて「ESCAN」コマンドを実行した場合は、「2 値化レベルの自動設定」で最後に決定された 2 値化レベルで、また 2 値化レベルを指定していない場合は、「2 値化レベルのマニュアル指定」の設定値で画像をメモリに取り込みますので、ご注意ください。

READF		リード	エフ						
機能	メモリに画像取り込み (フラッシュ同期信号付)								
書式	$\text{READF } \Delta \left(\begin{array}{c} a \\ \text{" ABCD" } \end{array} \right), \left(\begin{array}{c} (v) \\ (i) \\ (h) \end{array} \right) \quad () \text{ は省略可}$								
角解	<p>第1オペランド・・・取り込みメモリ画像を指定 第2オペランド・・・画面密着を実行するか、しないかを選択。</p> <table border="1"> <thead> <tr> <th>オペランド内容</th> <th>画面密着モード</th> </tr> </thead> <tbody> <tr> <td>0 (省略)</td> <td>画面密着を実行する</td> </tr> <tr> <td>1</td> <td>画面密着を実行しない</td> </tr> </tbody> </table> <p>・画面密着を実行するを選択した場合の動作は以下の順で動作します。 1. 画像取り込みの実行 2. 画面密着の実行</p> <p>第1オペランドで指定したメモリにフラッシュ同期信号を出力して画像を取り込みます。指定の方法は、画像を取り込むメモリの記号を指定して、取り込まないメモリは「*」を指定します。また、指定の順番は、A、B、C、Dの順に指定します。複数台のカメラから画像を取り込む際、後面パネルのカメラ切替えスイッチも切替えておいてください。このとき、メモリ構成はイメージチェッカ30と同じです。又、「画面密着-有り」を設定していますと、同時に実行されます。(この例としまして、「画面密着あり」を設定し、READF "A***" を実行しますと、指定メモリに画像を取り込んで画面密着を行うこととなります。</p> <p>aレジスタで指定した場合は、指定された番号から4個目までを無条件に読み込んで実行します。 この命令を実行するときの2値化レベルはこの命令の前に"THRES" コマンドで設定した値、もしくは"ESCAN" コマンドを実行した時に決定した値のうち、最後に実行された値になります。 また、いずれの実行もされていない場合は、「2値化レベルのマニュアル設定」の設定値が採用されます。</p> <p>(READF Δ An ; READF Δ "An An+1 An+2 An+3")</p> <p style="text-align: center;"> ↑ ↑ ↑ ↑ メモリ A B C D </p> <p><注意> 実行ブロックの「5. 画面密着」で「画面密着あり」を指定している時のみ、第2オペランドで指定した「画面密着を実行する」が有効になります。</p>			オペランド内容	画面密着モード	0 (省略)	画面密着を実行する	1	画面密着を実行しない
オペランド内容	画面密着モード								
0 (省略)	画面密着を実行する								
1	画面密着を実行しない								

<p>角 平 言 兌</p>	<p><文例> READF "*B**",1 ;フラッシュ同期信号を出力してメモリBに画像を画像密着を実行せずに取り込みます。</p> <p>READF A10 ;フラッシュ付でA10からA13までの内容で指定したメモリに画像を画像密着を実行して取り込みます。 (Ex A10 = "A", A11='*', A12='*', A13='D' のとき READF△A10 と READF△"A**D" は同じ内容です。)</p> <p>(注意)</p> <ul style="list-style-type: none"> ・品種モードで選択された「照明モード」は、無視されます。 ・フラッシュ同期信号は、出力ポートNO. 1の"0"ビット(ピンNO. 1~2)で出力されます。) ・品種モードにて「2値化レベルの自動設定」を設定しましても、このコマンドでは、実行されません。但し、"THRES"コマンドを設定していて、"ESCAN"コマンドを実行した場合は、「2値化レベルの自動設定」で、最後に決定された2値化レベルで、また、2値化レベルを指定していない場合は、「2値化レベルのマニュアル設定」の設定値で、画像をメモリに取り込みますので、ご注意ください。
----------------	---

READY	レディ
機能	レディ信号の制御
書式	READY Δ d
角解 説	<p>パラレルハンドシェイク用のレディー信号を制御します。 オペランドの指定 (d) は レディー信号を出力するとき 「ON」 レディー信号を出力しないとき 「OF」</p> <p>「ON」を実行したとき「電源を切る」か”START”コマンド実行中に”START信号”が入力されるか「OF」を実行するまで、「ON」の状態を保持し続けますので、注意してください。</p> <p>信号出力時の出力ポートと出力ビットはイメージチェッカ30・30Rと同じで、固定されています。</p> <p><文例> READY ON ; レディ信号をONにします。</p>

R E N D	レンド (リードエンド)				
機 能	画像取り込み完了信号				
書 式	R E N D Δ d				
角 注 言 究	<p>パラレルハンドシェイク用の画像取り込み完了信号を制御します。</p> <p>オペランドの指定 (d) は</p> <table border="0" style="margin-left: 40px;"> <tr> <td>画像取り込み完了信号を出力するとき</td> <td>「ON」</td> </tr> <tr> <td>画像取り込み完了信号を出力しないとき</td> <td>「OF」</td> </tr> </table> <p>「ON」を実行したとき「電源を切る」か「OF」を実行するまで、「ON」の状態を保持し続けますので、注意してください。</p> <p>信号出力時の出力ポートと出力ビットはイメージチェッカ30・30Rと同じで、固定されています。</p> <p><文例></p> <p>R E N D ON ; 画像取り込み完了信号をONにします。</p>	画像取り込み完了信号を出力するとき	「ON」	画像取り込み完了信号を出力しないとき	「OF」
画像取り込み完了信号を出力するとき	「ON」				
画像取り込み完了信号を出力しないとき	「OF」				

R E T		レ ッ ト
機 能	CALL、GOSUBからの復帰 プログラムの終結	
書 式	R E T	
角 準 言 兌	<p>” CALL” 又は” GOSUB” から復帰します。 登録されるプログラムは全て” R E T” で終結する事とします。</p> <p><文例>(CALL～) R E T ; ” CALL” から復帰します。</p> <p>(GOSUB～) R E T ; ” GOSUB” から復帰します</p> <p>R E T ; プログラムが終結します。</p>	

RSCAN	ルール スキャン
機能	判定演算の実行 (Rレジスタ)
書式	$RSCAN \Delta \begin{pmatrix} v \\ i \end{pmatrix}, \begin{pmatrix} v \\ i \end{pmatrix}$
角解 言説	<p>第1オペランドで指定する判定演算 (Rレジスタ) Noから第2オペランドで指定する判定演算 (Rレジスタ) Noまでの演算を実行します。</p> <p>指定する判定演算のプログラムはあらかじめ「判定出力」で設定しておいてください。もし、プログラムが設定されていない場合、無視して実行されますが、指定Rレジスタのメモリエリアが確保されていないため、「MOVE」等でRレジスタの内容を読み込もうとした場合、エラーとなります。</p> <p>また、判定演算のプログラム中、指定した判定結果レジスタに対応するチェッカの実行がされていない場合、判定結果レジスタにあらかじめ格納されている内容 (ex. チェッカを設定したときに格納された内容、前回実行したときにバックアップされていた内容、別のプログラムで実行したときに格納されていた内容etc.) が読み込まれます。</p> <p>たとえば、判定演算 $R1 = L11 * W1$ において ラインNo. 1が実行 (RSCAN) されていないとき L11に格納されていた内容が読み込まれます。</p> <p><文例> RSCAN 1,5 ; 判定演算R1~R5を実行します。</p> <p><エラーについて> 以下の動作を実行した時、エラーランプがONします。</p> <ul style="list-style-type: none"> ・判定出力 (R) プログラムに書かれているチェッカが未設定の時 ・エラーが発生したチェッカの判定結果を引用した時

S B I T		セットビット
機 能	ビットセット	
書 式	SBIT Δ $\left[\begin{array}{c} v \\ i \end{array} \right], v$	
解 説	<p>指定レジスタ（第2オペランド）の指定ビット（第1オペランド）をセット（=1）します。</p> <p>vレジスタは、32ビット構成です。 (LSB →0ビット、MSB →31ビット)</p> <p><文例> SBIT 3, V2 ; V2の3ビットを”1”（セット）とする。</p> <p>なお、第1オペランドが”0”～”31”以外の場合は、エラーとなりオーバーフローフラグがONされます。この時、第2オペランドの値は、変化しません。</p>	

SCAN		スキャン																								
機能	全チェックの検査と演算の実行																									
書式	SCAN (オペランドなし)																									
角解 言兌	<p>設定された全チェック (位置補正、ライン、ウィンドウ、パターンマッチング、特徴抽出) の検査と数値演算・判定の実行をします。なお、この"SCAN" コマンドでは、"ESCAN" コマンドは実行されません。また、判定結果も出力されません。</p> <p>(参考) 1) 品種モードで「チェックパターン表示選択」を「位置補正に追従して表示」を選択している場合、チェックパターンの消去・書き込みに多少時間を要します。それでは問題ある場合「固定位置に表示」又は、「表示しない」を選択して時間を短縮してください。</p> <p><文例> SCAN ; 全チェックの検査と演算の実行を行います</p> <p>このコマンドを実行しますと、イメージチェック30の検査と同じ内容を実行します。その内容をコマンドで書きますと(参考)の様になります。以下のプログラムの様にこのコマンドの後に、チェックを移動する様にプログラムを書いて、実行し(例:LSCAN 1, 1, V1, V2) 2回目に、"SCAN" を実行した場合</p> <pre> ST, START READ "A***" SCAN LSCAN 1, 1, V1, V2 GOTO ST </pre> <p>"LSCAN" により移動した、チェックが残ります、これは"SCAN" 実行した時、"LSCAN" で移動した移動先を読み込みに行けないためチェックが消せないことにより、但し表示だけの問題であり、実際の検査には問題がありません。</p> <p>[対策] ① "SCAN" を使用せずに"*SCAN" でプログラムを書く</p> <p>② 表示選択で、チェックパターンの「固定位置に表示」又は「表示しない」を選択する。</p> <p>(参考)</p> <table> <tr> <td>"SCAN" 中の実行内容</td> <td></td> <td></td> </tr> <tr> <td>ISCAN</td> <td>1, 64</td> <td></td> </tr> <tr> <td>LSCAN</td> <td>1, 512</td> <td rowspan="7">※TSCANは、30RPのみの機能です。</td> </tr> <tr> <td>WSCAN</td> <td>1, 512</td> </tr> <tr> <td>MSCAN</td> <td>1, 512</td> </tr> <tr> <td>FSCAN</td> <td>1, 512</td> </tr> <tr> <td>CSCAN</td> <td>1, 512</td> </tr> <tr> <td>RSCAN</td> <td>1, 512</td> </tr> <tr> <td>DSCAN</td> <td>1, 512</td> </tr> <tr> <td>※ TSCAN</td> <td>1</td> <td></td> </tr> </table>		"SCAN" 中の実行内容			ISCAN	1, 64		LSCAN	1, 512	※TSCANは、30RPのみの機能です。	WSCAN	1, 512	MSCAN	1, 512	FSCAN	1, 512	CSCAN	1, 512	RSCAN	1, 512	DSCAN	1, 512	※ TSCAN	1	
"SCAN" 中の実行内容																										
ISCAN	1, 64																									
LSCAN	1, 512	※TSCANは、30RPのみの機能です。																								
WSCAN	1, 512																									
MSCAN	1, 512																									
FSCAN	1, 512																									
CSCAN	1, 512																									
RSCAN	1, 512																									
DSCAN	1, 512																									
※ TSCAN	1																									

角率 言兌

<エラーについて>

以下の動作を実行した時、エラーランプがONします。
 "SCAN" コマンドは、ISCAN、LSCAN、WSCAN、MSCAN、
 FSCAN、CSCAN、DSCAN、RSCAN、TSCAN、ESCANを
 同時に実行しますので、上記チェックでエラーが発生する項目でエラー発生した場合、
 同様にしてエラーが発生します。

チェック	コマンド	内容
位置補正	ISCAN	初期化ブロックで「7. エラー信号をONする条件」で「1. エラー信号をONする」を選択している時、位置補正検出エッジが検出出来なかった場合エラーランプがONします。
特徴抽出	FSCAN	ラベリング処理個数が128個を越えた場合、エラーランプがONします。
主軸角検出	FSCAN	ラベリング処理個数が128個を越えた場合、エラーランプがONします。
数値演算	CSCAN	数値演算で引用しているチェックが未設定 数値演算結果が32ビットを越えた 0除算 引用しているチェックでエラーが発生した
判定出力	DSCAN RSCAN	判定出力 (R, D) プログラムで引用しているチェックが未設定 引用しているチェックでエラーが発生した
回転補正	TSCAN	検査対象物体がない 回転補正用位置検出エッジで検出エラー発生
露出補正	ESCAN	露出補正ができなかった場合

SHOW		シ ョ ウ
機 能	画面の表示選択	
書 式	$\text{SHOW } \Delta \left(\begin{array}{c} a \\ \text{" VBIMCP"} \end{array} \right)$	
角 度 説 明	<p>画面にオペランドで指定された表示をします。 指定方法は、" VBIMCP" の順番で、表示をしない場合は「*」を指定します。 それぞれ記号の意味は</p> <p>V : 生画像 (カメラから取込んだ処理をしていない画像) B : 2値化画像 (2値化された、リアルタイムの画像) I : イメージ (メモリ内容の表示) M : メッセージ (メッセージプレーンに描かれている内容の表示) C : カーソル (カーソルの表示) P : チェッカパターン (設定されたチェッカパターンの表示)</p> <p>a レジスタで指定する場合、指定 a レジスタから 6 個目までを無条件に読み込みNOの小さい順に左側から対応します。 すなわち $\text{SHOW } \Delta A_n$ $= \text{SHOW "A}_n \text{ A}_{n+1} \text{ A}_{n+2} \text{ A}_{n+3} \text{ A}_{n+4} \text{ A}_{n+5}"$</p> <p>各指定の文字以外を指定した場合は、「エラー」になりますのでご注意ください。 Bを単独では設定できません。必ず、' I ' と同時に指定してください。(例 $\text{SHOW " * B I M * * "}$)</p> <p>(参考) 1)リアルタイムの2値化画像を選択する際は、B、Iを指定してください。 (例) $\text{SHOW } \Delta \text{" * B I * * * "}$</p> <p>2)このコマンドを指定しない場合は、画像を取り込んだとき、(READ、READF、ESCANを実行させたとき) 2値化メモリ画像 ($\text{SHOW } \Delta \text{" * * I M C P "}$) を表示します。</p> <p><注意> キーボードより" ALT" キーと同時に" I"、" M"、" P"、" C" が入力されていたり、品種モードの「チェッカパターンを表示しない」を選択されていますと、これらの命令が、" SHOW" コマンドより優先されますので、ご注意ください。</p> <p>この" SHOW" コマンドで設定せずに、プログラムを作成、実行した場合は、$\text{SHOW " * * I M C P "}$ の設定に自動的になります。</p> <p><文例> $\text{SHOW " V * * M C P "}$; 画面に生画像(V)、メッセージ(M)カーソル(C)、チェッカパターン(P)の表示をします。 $\text{SHOW } A_{100}$; A100~A105の内容を画面に表示します。 $\text{SHOW " A}_{100}, A_{101}, A_{102}, A_{103}, A_{104}, A_{105} "$</p>	

(参考) 画面構成について

イメージチェッカ30P・30RPのプログラムモードの' SHOW' コマンドで制御可能な画面は全部で6画面あります。(図参照)

各画面は表示を選択されると、重ね書きされて画面表示します。

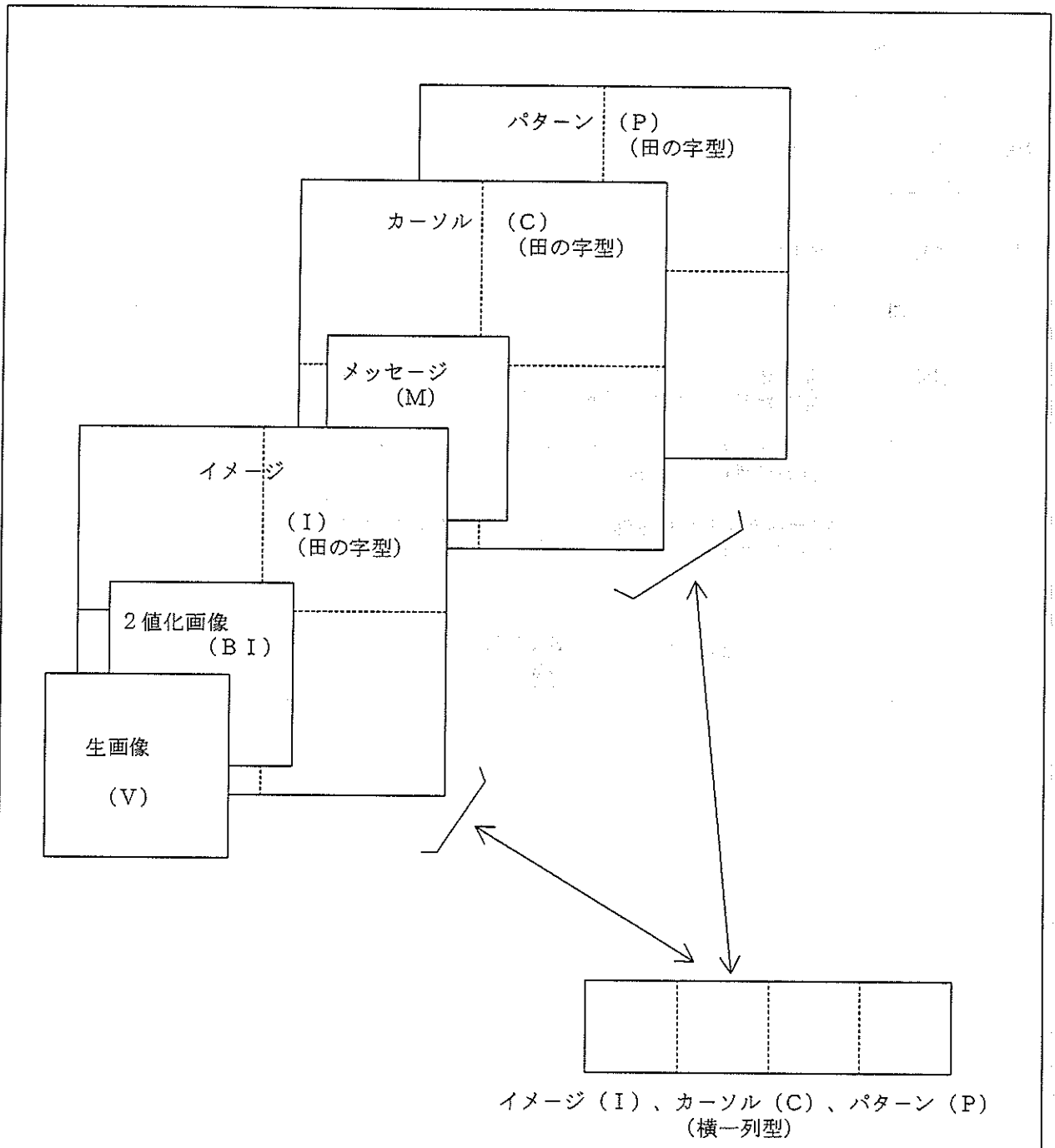
但し、画像に関する 生画像 (V)、リアルタイム2値化画像 (BI)、イメージ (I) は2つ以上の画像を同時に画面表示することはできません。

画像指定の優先順位は、生画面 (V)、リアルタイム2値化画像 (BI)、イメージ (I) の順です。

各画面の簡単な説明を下表に示します。

画面	' SHOW' での記号	説明	" MONT" による指定範囲※
生画像	V	カメラから取込んだ画像をそのまま表示します。	A~D
リアルタイム 2値化画像	BI	カメラから取込んだ画像を2値化し、リアルタイムで表示します。	A~D
イメージ	I	メモリに取込まれた2値化画像を表示します。	田の字型 A~I 横一列型 A~G
メッセージ	M	" DISP"、" DISPD"、" MLON" コマンドで指定された「文字・線」を表示します。 イメージチェッカ30Rでのメッセージとは、異なります。	指定 不可
カーソル	C	" CURSOR" コマンドにより設定された「カーソル」を表示します。	田の字型 A~I 横一列型 A~G
パターン	P	位置補正チェッカ、ウィンドウチェッカ等、チェッカを表示します。	

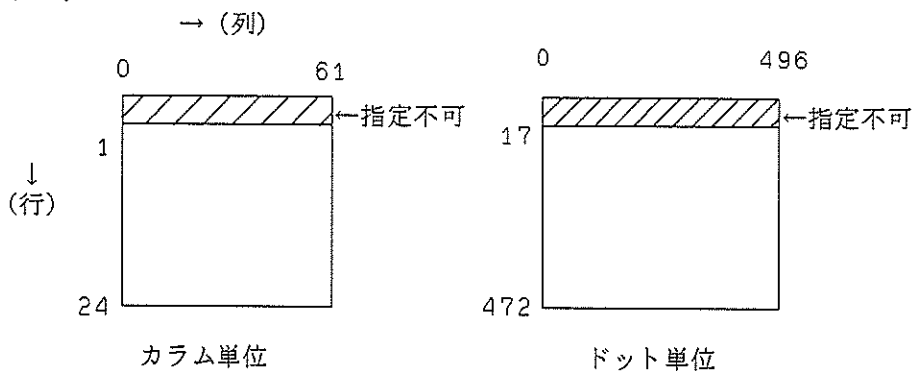
※ MONTによる指定範囲はV、BI、C、Pが全て同一となります。
V、または、BIを表示している場合は、A~Dのみが指定できます。

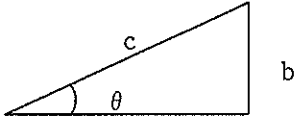


プログラムで制御可能な画面

表示範囲

メッセージプレーン



S I N		サイン
機 能	正弦の計算	
書 式	S I N Δ $\left[\begin{matrix} v \\ i \end{matrix} \right], v$	
角 度 説 明	<p>第1オペランドの内容の正弦計算を行い、結果を第2オペランドで指定したVレジスタに格納します。</p> <p>第1オペランドには角度(θ度)×100の値を入力します。 計算結果は(×10000)で、Vレジスタに格納されます。</p> <p>第1オペランドの指定範囲は、-2,147,483,648~2,147,483,647です。 オーバーフローした時にはエラーとなります。</p> <p><文例> S I N 3000, V 1 ; S I N 30度の計算を行いV 1 に結果を格納 します。 S I N 30度=0.5 → V 1 = 5000</p> <div style="text-align: center;">  </div>	

S - I N		シリアル イン
機 能	シリアル入力	
書 式	$S - I N \quad \Delta \quad a, \quad \begin{pmatrix} v \\ i \end{pmatrix}, \quad (v), \quad \begin{pmatrix} (a) \\ (v) \\ (i) \\ (h) \end{pmatrix}$ <p style="text-align: right;">() は省略可</p>	
角 準 言 兌	<p>第2オペランドで指定した文字列の個数をシリアル入力して、第1オペランドで指定した a レジスタから順番に、文字列の個数分格納します。また第4オペランドでは「区切り記号」をアスキーコードで指定し、第3オペランドでは「区切り記号」を含めた入力文字数を格納する v レジスタを指定します。</p> <ol style="list-style-type: none"> 1. 「区切り記号」の指定がある場合 <ol style="list-style-type: none"> ① 指定個数の文字入力がある ② 「区切り記号」が入力される のどちらかの場合に終了します。 2. 「区切り記号」の指定がない場合 <ol style="list-style-type: none"> ① 指定個数の文字が入力される。 ② CR”が入力される。 のどちらかの場合に終了します。②の場合、“CR”も1字とみなされ、a レジスタに格納されます。 <p>使用する a レジスタのNo. が A 1 0 0 0 を越えない様に、レジスタNo. と個数を設定してください。</p> <p><文例></p> <p>S - I N A30,5 ; 5個の文字列をシリアル入力し、A30からA34へ格納します。</p> <p>S - I N A10,10,V5,0X23 ; 10個の文字列か、0X23(=' #')の文字が入力するまでシリアル入力しA10~A19へ格納します。</p> <p><注意></p> <p>” S - I N ” コマンドが書かれているプログラムステップで、シリアルポートからの入力が入ってくるまでは、入力待ち状態となりプログラムは、そのステップで、停止しています。</p> <p>(参考)</p> <p>シリアル入力 (R S - 232 C) 用のコネクタのピン配列については、「イメージチェッカ30・30R導入マニュアル」の「R S - 232 Cの使い方」の項を参照してください。</p> <p>なお実行したとき、この” S - I N ” コマンドがあるステップではシリアル入力待ち状態となり、指定された文字数・区切り記号が入力されるまで、実行を停止している様に見えます。</p>	

S I O B R

S I O ボーレート

機 能

ボーレートの設定

書 式

SIOBR Δ $\begin{pmatrix} v \\ i \end{pmatrix}$

解 説

ボーレートの設定を下表に示す番号で指定します。
指定されていない場合は、初期設定値（9600ボー）に設定されます。

ボーレート (ボー)	番号
300	1
600	2
1200	3
2400	4
4800	5
9600	6
19200	7
38400	8

(注) プログラムの実行が終了すると、システム設定値に戻ります。

<文例>

SIOBR 6 ; ボーレートを9600ボーに設定します。

SIOCL		SIOキャラクタレングス										
機能	キャラクタ長の設定											
書式	SIOCL Δ $\begin{pmatrix} v \\ i \end{pmatrix}$											
角解 言説	<p>キャラクタ長の設定を、下表に示す番号で指定します。 指定されていない場合は、初期設定（8ビット）に設定されます。</p> <table border="1"> <thead> <tr> <th>キャラクタ長(ビット)</th> <th>番号</th> </tr> </thead> <tbody> <tr> <td>5</td> <td>1</td> </tr> <tr> <td>6</td> <td>2</td> </tr> <tr> <td>7</td> <td>3</td> </tr> <tr> <td>8</td> <td>4</td> </tr> </tbody> </table> <p>(注) プログラムの実行が終了すると、システム設定値に戻ります。</p> <p><文例> SIOCL 3 ; キャラクタ長を7ビットに設定します。</p>		キャラクタ長(ビット)	番号	5	1	6	2	7	3	8	4
キャラクタ長(ビット)	番号											
5	1											
6	2											
7	3											
8	4											

S I O P B	S I O パリティビット								
機 能	パリティビットの設定								
書 式	$S I O P B \Delta \begin{pmatrix} v \\ i \end{pmatrix}$								
角 説	<p>パリティビットの設定を、下表に示す番号で指定します。 指定されていない場合は、初期設定(パリティ無し)に設定されます。</p> <table border="1" data-bbox="566 884 869 1120"> <thead> <tr> <th>パリティビット</th> <th>番号</th> </tr> </thead> <tbody> <tr> <td>パリティ無し</td> <td>1</td> </tr> <tr> <td>偶数</td> <td>2</td> </tr> <tr> <td>奇数</td> <td>3</td> </tr> </tbody> </table> <p>(注) プログラムの実行が終了すると、システム設定値に戻ります。</p> <p><文例> S I O P B 2 ; パリティビットを「偶数」に設定します。</p>	パリティビット	番号	パリティ無し	1	偶数	2	奇数	3
パリティビット	番号								
パリティ無し	1								
偶数	2								
奇数	3								

S I O S B		S I O ストップビット										
機 能	ストップビットの設定											
書 式	S I O S B Δ $\begin{pmatrix} v \\ i \end{pmatrix}$											
解 説	<p>ストップビットの設定を、下表に示す番号で指定します。 指定されていない場合は、初期設定値（2ビット）に設定されます。</p> <table border="1"> <thead> <tr> <th>ストップビット（ビット）</th> <th>番号</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1</td> </tr> <tr> <td>1.5</td> <td>2</td> </tr> <tr> <td>2</td> <td>3</td> </tr> <tr> <td>無効</td> <td>4</td> </tr> </tbody> </table> <p>(注) プログラムの実行が終了すると、システム設定値に戻ります。</p> <p><文例> S I O S B 1 ;ストップビットを1ビットに設定します。</p>		ストップビット（ビット）	番号	1	1	1.5	2	2	3	無効	4
ストップビット（ビット）	番号											
1	1											
1.5	2											
2	3											
無効	4											

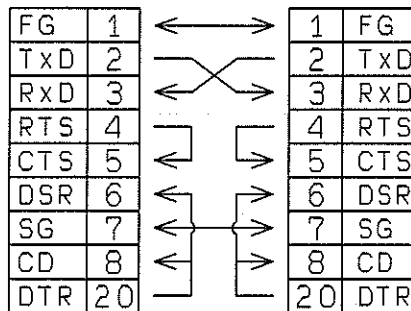
S - O U T		シリアルアウト
機 能	シリアル出力	
書 式	S-OUT Δ a , $\left[\begin{matrix} v \\ i \end{matrix} \right]$	
角 平 言 説	<p>第1オペランドで指定したaレジスタから第2オペランドで指定した個数分の文字列を、シリアル出力します。</p> <p>シリアル出力 (RS232C) 用のコネクタのピン配列については「イメージチェッカ30または30R導入マニュアル」の「RS232Cの使い方」の項を参照してください。</p> <p>使用するaレジスタのNo. がA1000を越えない様に、レジスタNo. と個数を設定してください。</p> <p><文例>S-OUT A10,10 ; A10からA19まで10個の文字列をシリアル出力します。</p>	

S P R I N T		シリアルプリント																		
機 能	プリンタへの出力 (シリアル出力ポート)																			
書 式	SPRINT Δ $\left[\begin{smallmatrix} v \\ i \end{smallmatrix} \right], \left[\begin{smallmatrix} v \\ i \end{smallmatrix} \right], \left[\begin{smallmatrix} a \\ n \end{smallmatrix} \right], \left[\begin{smallmatrix} v \\ i \end{smallmatrix} \right]$																			
角 説	<p>シリアル (RS 232C) からシリアルプリンタへデータ出力します。</p> <p>第1オペランド. プリントアウトモードを指定</p> <table border="1"> <thead> <tr> <th>オペランド内容</th> <th>プリントアウトモード</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>改行コードを付けずに出力</td> </tr> <tr> <td>1</td> <td>改行コード (OXD="CR") を付けて出力</td> </tr> </tbody> </table> <p>第2オペランド. 前置するスペース (空白) の数を指定 (0,1,2,3.)</p> <p>第3オペランド. プリントアウトする a レジスタの開始No か文字列を指定</p> <p>第4オペランド. プリントアウトする a レジスタまたは文字列の個数を指定 (1,2,3.)</p> <p>実際にプリンタに文字が打ち出されるのは改行コードを受け取るまでプリンタ内部のバッファに蓄積され続け、改行コードを受け取って文字が打ち出されます。また、プリンタ内部のバッファに1行分 (約80文字:通常) の文字が蓄積されると、自動的に文字がプリントアウトされます。</p> <p>(結線方法) 多くのプリンタはセントロニクス仕様となっていますのでシリアルポートから出力されたデータをシリアルプリンタに入力しても文字を打ち出すことはできません。この場合は、イメージチェッカ30P・30RPのコントローラとプリンタとの間にシリアル/パラレル変換器 (シリアル→パラレル) を設置してプリンタにパラレルで入力されるようにしてください。</p> <p>(プリンタモード) 尚、イメージチェッカ30P・30RPのコントローラは、X-ON/OFFによる制御を行いますので、プリンタの設定に注意してください。また、I/Oの設定値 (ボーレート、キャラクタ長、パリティビット、ストップビット) を同じ設定にしてください。</p> <p><文例></p> <table> <tbody> <tr> <td>MOVE</td> <td>CW1,V1</td> <td>カウンタ1の値をA1-A10C</td> </tr> <tr> <td>MOVE</td> <td>V1,A1,10</td> <td>格納</td> </tr> <tr> <td>SPRINT</td> <td>0,5,"ウィンドウ1=",14</td> <td>"ウィンドウ1="とカウント7を行った後</td> </tr> <tr> <td>SPRINT</td> <td>1,0,A1,10</td> <td>計測値をカウント7行し、改行します。</td> </tr> </tbody> </table>		オペランド内容	プリントアウトモード	0	改行コードを付けずに出力	1	改行コード (OXD="CR") を付けて出力	MOVE	CW1,V1	カウンタ1の値をA1-A10C	MOVE	V1,A1,10	格納	SPRINT	0,5,"ウィンドウ1=",14	"ウィンドウ1="とカウント7を行った後	SPRINT	1,0,A1,10	計測値をカウント7行し、改行します。
オペランド内容	プリントアウトモード																			
0	改行コードを付けずに出力																			
1	改行コード (OXD="CR") を付けて出力																			
MOVE	CW1,V1	カウンタ1の値をA1-A10C																		
MOVE	V1,A1,10	格納																		
SPRINT	0,5,"ウィンドウ1=",14	"ウィンドウ1="とカウント7を行った後																		
SPRINT	1,0,A1,10	計測値をカウント7行し、改行します。																		

シリアル/パラレル変換器とプリンタとの接続については、メーカーによりコネクタ、結線等が異なりますので、シリアル/パラレル変換器を購入される際には、各メーカーの専用ケーブルを併せて購入ください。

① セントロニクス使用のプリンタを使用する場合
シリアル/パラレル変換器を使用した例

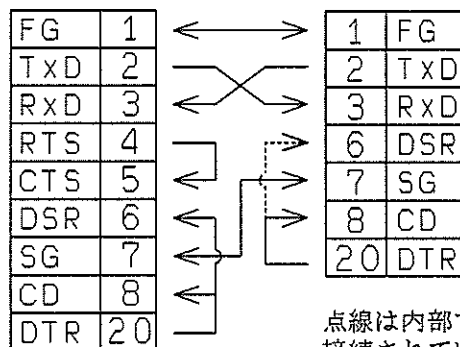
Pタイプコントローラ シリアル/パラレル変換器



② シリアルプリンタを使用する場合

イメージチェッカ30P・30Pコントローラのシリアルポートから直接シリアルプリンタにプリントアウトをする場合は、シリアルプリンタの特性により、プリンタ毎に結線が異なりますので、プリンタケーブルを作成する場合は、プリンタのマニュアルを参考にして作成をしてください。EPSON社から、シリアルインターフェイスボードがオプションで発売されています。ここでは、同社のインテリジェントシリアルインターフェイス type II #8149 との接続例を下記に示します。

Pタイプコントローラ インテリジェントシリアル
インターフェイス type II #8149



点線は内部で
接続されています。

(参考)

”PRINT” 命令を利用してプリンタに出力する時パラレル出力ならびにパラレル入力をプリンタ制御用として使用することになります。この時、外部機器とパラレル入出力を使用していますと同時にプリンタが使用できませんので、”SPRINT” 命令を利用することでプリンタも同時に接続することができます。この場合、シリアルプリンタの入手し難いまたは、既にセントロニクス仕様のプリンタが有る場合は、①の例に示しました様に、シリアル/パラレル変換器を使用することで、対応することができます。

<注意>

漢字キャラクタのプリンタへの出力時の制御コードは、以下の様になっています。接続されるプリンタが、以下の様になっていることを確認ください。JIS第1水準以外の文字は、出力できません。

漢字 in : ”ESC” + ”K”
漢字 out : ”ESC” + ”H”

S Q R T		スクエアルート
機 能	平方根の計算	
書 式	S Q R T Δ $\left[\begin{matrix} v \\ i \end{matrix} \right], v$	
角 平 言 兌	<p>第1オペランドの内容の平方根計算を行ない結果を第2オペランドで指定したvレジスタに格納します。 計算結果は(×10000)でvレジスタに格納されます。</p> <p>オーバーフローした時はエラーとなります。</p> <p><文例> S Q R T 2,V100 ; $\sqrt{2}$の計算結果をV100に格納します。 $\sqrt{2}=1.4142$ $1.4142 \times 10000=14142 \rightarrow v 100$</p>	

START	スタート
機能	スタート・品種切替の信号（パラレル入力）の読み込みと実行
書式	START
解説	<p>イメージチェッカ30・30Rのパラレル入力のうち、スタート信号読み込み又は、品種切替信号の読み込みと実行を行ないません。 スタート信号、品種切替信号は信号の“ON”するエッジで読み込まれます。 （スタート信号、品種切替信号は、微分入力型です。） 品種切替を実行したとき、品種設定が行なわれていない等、 品種切替エラーが発生した場合、エラー信号を「ON」にします。</p> <p>1) データ出力の流れ 以下に、コマンドを実行したときの流れをフローチャートに示します。</p> <pre> graph TD Start(()) --> ReadyON[READY ON] ReadyON --> OVFLG[OVFLG OFF] OVFLG --> StartSignal{スタート信号 ON?} StartSignal -- YES --> ErrorOff1[エラー信号 OFF] ErrorOff1 --> ReadyOff1[READY OFF] ReadyOff1 --> Circle1(()) StartSignal -- NO --> SpeciesSignal{品種切替信号 ON?} SpeciesSignal -- YES --> ErrorOff2[エラー信号 OFF] ErrorOff2 --> ReadyOff2[READY OFF] ReadyOff2 --> SpeciesExec[品種切替実行] SpeciesExec --> SpeciesError{品種切替エラー?} SpeciesError -- YES --> ErrorOn[エラー信号 ON] ErrorOn --> Circle2(()) SpeciesError -- NO --> Right(()) Right --> Circle3(()) Circle3 --> ReadyON </pre> <p>このコマンドを実行したとき、実行モード中、「品種切替の有無」の設定は無視され、品種切替が行なわれます。</p> <p>2) 結線方法 パラレル入出力の使用ポートはイメージチェッカ30・30Rと同じです。 結線については「イメージチェッカ30・30R導入マニュアル」の「8-1 パラレル信号による通信」を参照してください。</p>

S T R O B		ス ト ロ ー ブ
機 能	ストロブ信号の制御	
書 式	STROB Δ d	
角 弁 説	<p>パラレルハンドシェイク用のストロブ信号を制御します。 オペランドの指定 (d) は、 ストロブ信号を出力するとき 「ON」 ストロブ信号を出力しないとき 「OF」</p> <p>「ON」 を実行したとき、「電源を切る」か「OF」 を実行するまで 「ON」 の状態を保持し続けますのでご注意ください。</p> <p>信号出力時の出力ポートと出力ビットはイメージチェッカ30・30Rと同じで、 固定されています。</p> <p><文例> STROB ON ; ストロブ信号をONします。</p>	

SUB		サブ
機能	レジスタ間減算	
書式	<p>1) 数値レジスタ間減算</p> $\text{SUB } \Delta \quad \begin{bmatrix} v \\ i \end{bmatrix}, \begin{bmatrix} v \\ i \end{bmatrix}, v$ <p>2) 文字レジスタ間減算</p> $\text{SUB } \Delta \quad \begin{bmatrix} a \\ G \end{bmatrix}, i, a$	
角平 言兑	<p>1) 数値レジスタ間減算</p> <p>第1オペランドの内容から第2オペランドの内容を減算し、第3オペランドの指定するvレジスタに格納します。 〔(第1オペランド) - (第2オペランド)〕 → (第3オペランド)</p> <p>減算結果の値の範囲は、-2,147,483,648~2,147,483,647 です。 オーバーフローした時はエラーとなりオーバーフローフラグがONされます。 この時、第3オペランドの値は、意味をもたない値となりますのでご注意ください。</p> <p><文例> SUB V20, V30, V40 ; (V20-V30→V40)</p> <p>2) 文字レジスタ間減算</p> <p>第1オペランドの示すアスキーコードから、数値(第2オペランド)を減算したアスキーコードを、第3オペランドで指定したaレジスタに格納します。</p> <p><文例> MOVE 'B', A1, 1 ; 'B'(42H)→A1 SUB A1, 1, A2 ; A1-1=(42H)-1=(41H) 41H('A')→A2</p> <p><注意> SUB 'B', 1, A1 左のような、文字列(" ")の指定はできません。</p>	

TAN		タンジェント
機能	正接の計算	
書式	TAN Δ $\left[\begin{matrix} v \\ i \end{matrix} \right], v$	
角単 言説	<p>第1オペランドの内容の正接計算を行ない、結果を第2オペランドで指定したvレジスタに格納します。</p> <p>第1オペランドには、角度(θ)×100の値を入力します。 計算結果は、(×10,000)でvレジスタに格納されます。</p> <p>オーバーフローした時はエラーとなります。</p> <p><文例> TAN 4500, V1 ; TAN45度の計算を行ない、V1に結果を格納します。 TAN45度=1 1×10000=10000→ V1</p>	

TANG		ティーングル
機能	回転角 $T\theta$ の設定 (回転補正用)	
書式	$\text{TANG } \Delta \begin{pmatrix} v \\ i \\ h \end{pmatrix}, \begin{pmatrix} v \\ j \\ i \\ h \end{pmatrix}$	
角準 言兌	<p>回転補正の設定値設定専用コマンドです。</p> <p>第1オペランド・・・第2オペランドで指定した回転角を実行するメモリモードを指定します。 0：全メモリ (メモリA, B, C, D) 1：メモリA 2：メモリB 3：メモリC 4：メモリD</p> <p>第2オペランド・・・回転角 ($T\theta$) を整数で設定します。 ・回転角とは、回転補正後、画像がx軸となす角度です。</p> <p>このコマンドにより設定された回転角 ($T\theta$) の設定値は、次のTANGコマンドにより設定されるまでもしくは、次回プログラム実行まで有効です。</p> <p><文例> TANG 1,30:Aメモリの回転角 ($T\theta$) を30度に設定します。</p> <p>(注意) vレジスタによるメモリエリア、モード指定はできません。 TANG Δ 1,C1[V1,V2] この様な指定はできません。</p>	

T B I T		テストビット
機能	条件付きジャンプ	
書式	$TBIT \quad \Delta \quad \left[\begin{matrix} v \\ i \end{matrix} \right], v, b, \left[\begin{matrix} 1 \\ i \end{matrix} \right]$ <p style="text-align: right;">判定条件を示す記号：第3オペ b NZ → ≠ 0 (=1) ZR → = 0</p>	
角解 言説	<p>指定レジスタ（第2オペランド）の指定ビット（第1オペランド）のビットテストを行ない、判定条件に従って指定ステップ又はラベルへジャンプします。</p> <p>また、不成立ならば、次のステップを実行します。</p> <p>（vレジスタは32ビット構成 LBS→0ビット, MSB→31ビット）</p> <p>ステップ数は-63～63（0を除く）の範囲で、最終行を越えない様に設定してください。</p> <p>尚、第1オペランドが”0”～”31”以外の場合は、エラーとなり、オーバーフローフラグがONされ、次のステップを実行します。</p> <p>また、第3オペランド：bは、以下の判定条件を示します。</p> <p style="text-align: center;">NZ → ≠ 0 (=1) ZR → = 0</p> <p><文例> TBIT 0, V10, ZR, OK ; V10の第0ビットをテストし、0ならばOKへジャンプします。</p>	

TCENT		ティーンセンター
機能	回転中心座標設定 (回転補正用)	
書式	$\text{TANG } \Delta \quad \begin{pmatrix} v \\ i \\ h \end{pmatrix}, \begin{pmatrix} v \\ j \\ i \\ h \end{pmatrix}, \begin{pmatrix} v \\ j \\ i \\ h \end{pmatrix}$	
角評 言兌	<p>回転補正の設定値設定専用のコマンドです。</p> <p>第1オペランド・・・回転中心座標を設定するメモリモードを指定します。</p> <p style="padding-left: 40px;">0：全メモリ (メモリ A, B, C, D) 1：メモリ A 2：メモリ B 3：メモリ C 4：メモリ D</p> <p>第2オペランド・・・第1オペランドで指定されたメモリの 回転中心 x 座標 (CX) を設定します。</p> <p>第3オペランド・・・第1オペランドで指定されたメモリの 回転中心 y 座標 (CY) を設定します。</p> <p>このコマンドは次回 TCENT コマンドもしくは、次回プログラム 実行まで有効です。</p> <p><文例> TCENT 3,G112,G113:Cメモリの回転中心を CX=G112 (主軸角検出チェッカ No. 501 の対象 No. 1 の重心 x 座標) CY=G113 (" " " " の重心 x 座標) に設定します。</p> <p>(注意) vレジスタによるメモリエリア, モード指定はできません。 TCENT Δ 1,C1[V1,V2],C1[V3,V4] この様な指定はできません。</p>	

THRES		スレッシュ
機能	2 値化レベルの設定	
書式	$\text{THRES } \Delta \begin{pmatrix} a \\ d \end{pmatrix}, \begin{pmatrix} v \\ i \\ d \end{pmatrix}$ 第1オペランドのd ; (A, B, C, D) 第2オペランドのd ; (SA, SB, SC, SD)	
角平 言兌	<p>第1オペランドでは2 値化レベルを設定するメモリ (A, B, C, D) を指定し、第2オペランドでは2 値化レベル (0~63) を指定します。</p> <p>第2オペランドでは、他メモリの設定値を読み込むことが可能です。「SA, SB, SC, SD」を指定することにより、それぞれAメモリからDメモリの設定値を読み込めます。a レジスタによる指定は出来ません。</p> <p>SA→Aメモリの設定値。</p> <p><文例：1> THRES A, 32 ; Aメモリの2 値化レベルを32に設定します。</p> <p><文例：2> ESCAN 1, 1 ; 露出補正チェッカNo. 1 (Aメモリ) を実行します。</p> <p>THRES B, SA ; Bメモリの2 値化レベルをAメモリの設定値 (露出補正チェッカにより決定した値) に設定します。</p> <p>一度” THRES” で2 値化レベルを設定するとそれ以降は設定された2 値化レベルとなります。</p>	

TMRD		タイマリード
機能	内部タイマ値の読み込み	
書式	TMRD Δ v	
角平 言兌	<p>内部タイマの値をオペランドで指定したvレジスタに格納します。</p> <p>内部タイマは、10 msecに1回カウントアップします。 従って、(読み出された値)×10によってタイマの経過値(msec)が得られます。</p> <p><文例> TMRD V100 ; 内部タイマの値をV100に格納します。</p>	

TMWT	タイマライト
機能	内部タイマの初期値セット
書式	$TMWT \quad \Delta \quad \begin{pmatrix} v \\ i \end{pmatrix}$
角解 言兌	<p>内部タイマの初期値をオペランドで指定された値にセットします。</p> <p>設定は0以上で10msec単位です。</p> <p>内部タイマは、10msecに1回カウントアップします。</p> <p><文例></p> <p>TMWT 0 ; タイマの初期値を0msecにリセットします。</p>

TSCAN		ティースキャン
機能	回転補正チェッカと回転補正の実行（回転補正用）	
書式	$TSCAN \quad \Delta \quad \left[\begin{array}{c} "ABCD" \\ a \end{array} \right], \quad \left[\begin{array}{c} (v) \\ (i) \\ (h) \end{array} \right]$ () は省略可です。	
角率 言説	<p>回転補正実行専用のコマンドです。</p> <p>第1オペランド・・・回転補正を実行する画像メモリを指定します。</p> <ul style="list-style-type: none"> ・指定の方法は、回転補正させるメモリの記号を指定して、回転させないメモリは「*」を指定します。また、指定の順番はA, B, C, Dの順に指定します。 ・aレジスタを指定した場合は、指定された番号から4個目までを無条件に読み込んで実行します。 <p>第2オペランド・・・設定ブロックの「9、回転補正」で指定された回転角検出チェッカの実行を回転補正実行前に行うか、行なわないかを'0'（省略）、'1'で指定します。</p> <p>1：回転角検出チェッカを実行しない。 0、(省略)：回転角検出チェッカを実行する。</p> <ul style="list-style-type: none"> ・設定ブロックの「9、回転補正」のプログラムで設定された差分(WX, WY) 回転角(Tθ)、回転中心(CX, CY)、移動量(SX, SY)の値により、回転を実行します。'TPRIN' 'TWIDTH', 'TANG', 'TCENT', 'TSHIFT'のコマンドにより、回転補正の条件が設定されていた場合は、その設定値で回転を実行します。 ・「画面密着」を選択していた場合、「回転補正」実行後、「画面密着」を実行します。 ・この命令により回転補正する画像は、このコマンドを実行する直前の画像により行います。つまりこのコマンドより前に'READ', 'READF'コマンドにより画像取込みをしなければ、前回取込んだ画像により、回転補正を実行します。 $(TSCAN \quad \Delta \quad A_n : TSCAN \quad \Delta \quad " \begin{array}{cccc} A_n & A_{n+1} & A_{n+2} & A_{n+3} \end{array} ")$ $\begin{array}{cccc} \uparrow & \uparrow & \uparrow & \uparrow \\ \text{メモリ} & A & B & C & D \end{array}$ <p>TSCANコマンド実行時、以下の場合</p> <ul style="list-style-type: none"> ・エラー信号を"ON"。(S7=1) ・回転補正を実行しない。 ・B2=1, T1~T4=0 <p>になります。</p> <p>回転補正設定コマンドにより設定値を設定していない時 (イメージチェッカ30Rとして動作します。)</p> <ul style="list-style-type: none"> ・検査物体がない。 ・指定チェッカがない。 ・プログラム中で数値演算結果を引用したチェッカがエラー発生 <p>回転補正設定コマンドにより設定値を設定している時</p> <ul style="list-style-type: none"> ・設定値が範囲外("TPRIN", "TWIDTH"コマンド) 	

角 評 言

注意：・回転補正設定コマンドとは” TWIDTH” , ” TPRIN” , ” TANG” , ” TCENT” , ” TSHIFT” のことを示します。

・チェックを実行してエラーが発生した時演算の方法によっては設定値が範囲外とならない場合があります。このとき” TSCAN” 実行時にエラーかどうか判断できないためエラーとならずに回転補正を実行しますのでプログラム中でチェックの判定結果を読み込んで、エラーかどうかを判断してください。

<文例> TSCAN Δ ” **C*” : メモリCの回転補正を行います。
 TSCAN A10 : A10からA13までの内容のメモリを回転補正します。

(Ex A10= ” *” , A11= ” B” , A12= ” C” , A13= ” *” のとき READ Δ A10= READ Δ ” *BC*” です。)

(注意) ・設定ブロックの「9. 回転補正」のプログラムを設定していないメモリは回転補正せず、無視しますので、ご注意ください。
 このとき、「9. 回転補正」のプログラムを設定せずにTPRIN, TANG, TCENT, TSHIFT, TWIDTHにより回転補正の設定を行っても、回転補正は実行されません。

・少なくとも「画像外エリア色」の色（白または黒）の指定を行うことにより、回転補正のプログラムが設定してあるものと見なし回転補正が実行されます。

<エラーについて>

検査対象物がない場合

回転補正用位置検出エッジで検出エラーが発生した場合。

エラーランプが、ONします。

T S H I F T		テ ィ ー シ フ ト
機 能	移動量 (SX, SY) 設定 (回転補正用)	
書 式	$TSHIFT \quad \Delta \quad \begin{pmatrix} v \\ i \\ h \end{pmatrix}, \begin{pmatrix} v \\ j \\ i \\ h \end{pmatrix}, \begin{pmatrix} v \\ j \\ i \\ h \end{pmatrix}$	
角 解 説	<p>回転補正設定値設定専用コマンドです。</p> <p>第1オペランド・・・移動量を設定するメモリモードを指定します。 0：全メモリ (メモリ A, B, C, D) 1：メモリ A 2：メモリ B 3：メモリ C 4：メモリ D</p> <p>第2オペランド・・・第1オペランドで指定されたメモリの x方向移動量 (SX) を設定します。</p> <p>第3オペランド・・・第1オペランドで指定されたメモリの y方向移動量 (SY) を設定します。</p> <p>このコマンドは次回のTSHIFTコマンドもしくは、 次回プログラム実行まで有効です。</p> <p><文例> TSHIFT 4, 100, 120 : Dメモリの移動量SX=100, SY=120に設定します。</p> <p>(注意) vレジスタによるメモリエリア, モード指定はできません。 TSHIFT Δ 1, CI [V1, V2], CI [V3, V4] この様な指定はできません。</p>	

TPRIN	テープリンシプル
機能	主軸角 (Pθ) 設定 (回転補正用)
書式	$\text{TPRIN } \Delta \left(\begin{array}{c} v \\ i \\ h \end{array} \right), \left(\begin{array}{c} v \\ j \\ i \\ h \end{array} \right)$
角弁 言説	<p>回転補正設定値の設定値設定のコマンドです。</p> <p>第1オペランド・・・主軸角を設定するメモリモードを指定します。 0：全メモリ (メモリA, B, C, D) 1：メモリA 2：メモリB 3：メモリC 4：メモリD</p> <p>第2オペランド・・・第1オペランドで指定されたメモリの主軸角 (Pθ) を設定します。</p> <p>設定範囲：-90 ≤ Pθ ≤ 90</p> <p>このコマンドにより設定された主軸角 (Pθ) の値は次回、TPRIN コマンドにより設定するかもしくはRETコマンドでプログラム実行を終了し次回、プログラムを実行するまで有効です。</p> <p><文例> TPRIN 1, CG116 ; Aメモリの主軸角 (Pθ) を Pθ = G116 (主軸角検出チェックNo. 501の対象No. 1の主軸角) に設定します。</p> <p>(注意) このコマンドは、主軸角の設定を行う機能のみ持っています。第2オペランドで、G116等チェック結果レジスタ (j) によりチェックの検査結果を引用しようとする場合は、'FSCAN' 等により検査結果を引用しようとするチェックをあらかじめ実行しておいてください。</p> <div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 10px 0;"> <p>回転補正モードの切換え</p> </div> <p>プログラム実行中、回転補正のモード「主軸角モード」「三角比モード」を切換え可能です。</p> <ul style="list-style-type: none"> ・切換えは”TPRIN”又は”TWIDTH”コマンドを実行した時です。 <p style="margin-left: 40px;">TPRIN・・・主軸角モード</p> <p style="margin-left: 40px;">TWIDTH・・・三角比モード</p> <ul style="list-style-type: none"> ・”TSCAN”実行時、最後に設定されたモードで回転を実行します。 ・品種モードでは、回転補正を設定したときのモードで実行します。 ・TPRIN、TWIDTH実行時、エラーが発生した場合、'OVFLG' がONされ、モード切換えは行ないません。(この後、TSCANを実行すると、エラーが発生する以前のモードで実行されます)

T W I D T H	テ ィ ー ウ ィ ド ス
機 能	水平差分 (WX)、垂直差分 (WY) の設定 (回転補正用)
書 式	$T W I D T H \quad \Delta \quad \begin{pmatrix} v \\ i \\ h \end{pmatrix}, \begin{pmatrix} v \\ j \\ i \\ h \end{pmatrix}, \begin{pmatrix} v \\ j \\ i \\ h \end{pmatrix}$
角 解 説	<p>回転補正設定値の設定専用のコマンドです。</p> <p>第1オペランド・・・差分値を設定するメモリモードを指定します。 0:全メモリ (メモリA, B, C, D) 1:メモリA 2:メモリB 3:メモリC 4:メモリD</p> <p>第2オペランド・・・第1オペランドで指定されたメモリの水平差分 (WX) を設定します。</p> <p>第3オペランド・・・第1オペランドで指定されたメモリの垂直差分 (WY) を設定します。</p> <p>このコマンドは次回のTWIDTHコマンドもしくは、次回プログラム実行まで有効です。</p> <p>第2オペランドの指定範囲 $-511 \leq WX \leq 511$ 第2オペランドの指定範囲 $-479 \leq WY \leq 479$</p> <p><文例> TWIDTH 2, 100, 150 : Bメモリの水平差分 (WX) = 100 垂直差分 (WY) = 150に設定します。</p> <p>(注意) vレジスタによるメモリエリア, モード指定はできません。 TWIDTH Δ 1, CI [V1, V2], CI [V3, V4] この様な指定はできません。</p> <div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 10px 0;"> <p>回転補正モードの切換え</p> </div> <p>プログラム実行中、回転補正のモード、「主軸角モード」「三角比モード」を切換え可能です。</p> <ul style="list-style-type: none"> ・切換えは”TPRIN”又は”TWIDTH”コマンドを実行した時です。 <p style="margin-left: 40px;">TPRIN 主軸角モード</p> <p style="margin-left: 40px;">TWIDTH 三角比モード</p> <ul style="list-style-type: none"> ・”TSCAN”実行時、最後に設定された、モードで回転を実行します。 ・品種モードでは、回転補正を設定したときのモードで実行します。 ・TPRIN、TWIDTH実行時、エラーが発生した場合、’OVFLG’がONされ、モード切換えは行ないません。(この後、TSCANを実行すると、エラーが発生する以前のモードで実行されます)

TYPE		タイプ
機能	品種切替	
書式	TYPE Δ $\left[\begin{array}{c} d \\ a \end{array} \right], \left[\begin{array}{c} v \\ i \end{array} \right]$	第1オペランドのd M: 内部メモリ I: ICカードのメモリ
解説	<p>第1オペランドはメモリモード (M: 内部、I: ICカード) を指定し、第2オペランドは品種NO. を指定します。</p> <p><文例> TYPE M, 10 ; 内部メモリーのNo. 10に品種切替えします。</p> <p>(注意)</p> <p>1) 品種モードで、「チェッカパターン表示選択」を「位置補正に追従して表示」または「固定位置に表示」を選択している場合、チェッカパターンの消去・描き込みに多少時間を要します。それでは問題のある場合、「表示しない」を選択して時間を短縮してください。</p>	

WHILE (～GOTO)		フォワイル (～ゴートウ)
機能	WHILEからGOTOまでの区間中にある命令を繰り返して実行	
書式	$\text{WHILE } \Delta \left[\begin{array}{c} v \\ i \end{array} \right], \left[\begin{array}{c} v \\ i \end{array} \right], \left[\begin{array}{c} v \\ i \end{array} \right], \left[\begin{array}{c} l \\ i \end{array} \right]$	
角評 言兌	<p>” WHILE” から” GOTO” までの区間中にある命令を「初期値」から「増分」だけ加減算しながら実行し、「終値」を超えると指定されたラベルまたは、ステップへジャンプします。</p> <p>第1オペランド. 「初期値」を設定します。</p> <p>第2オペランド. 「終値」を設定します。</p> <p>第3オペランド. 「増分」を指定します。(0は指定不可)</p> <p>第4オペランド. 終了したときのジャンプ先を指定します。 ラベル : 英大文字で始まる2文字の英数字 ステップ数: -63～63(0を除く)</p> <p>「終値」を越えるというのは、</p> <ul style="list-style-type: none"> ・「増分」が正の時、「終値」より大きくなる時 ・「増分」が負の時、「終値」より小さくなる時 <p>を意味します。</p> <p>実行した時、以下の場合には実行せずにエラーとなり、指定されたラベル、またはステップへジャンプします。</p> <ul style="list-style-type: none"> ・「増分」が正の値で、「初期値」が「終値」より大きい場合 ・「増分」が負の値で、「初期値」が「終値」より小さい場合 	

角 評

<文例>

```
MOVE 0,V1 ;0→V1
L2 WHILE 0,10,1,L1
ADD 1,V1,V1 ;V1→1+V1
GOTO L2
L1
```

初期値”0”から増分”1”を加算しながら
”GOTO”までの実行を繰り返し終値”10”を
越えると”L1”へジャンプします。
”WHILE”文を終了した時、”V1=11”
となります。

(注意)

”WHILE~GOTO”の中で”WHILE~GOTO”を挿入することはできません。
つまり、このコマンドを2重使用することはできません。2重使用
した場合は、2回目の”WHILE”コマンドのステップでエラーが発生し
以降は実行されずに、2回目の”WHILE”で指定されたジャンプ先へジ
ャンプします。そして”GOTO”で1回目の”WHILE”文に戻ったときは、
以降のステップは、実行されずに指定されたラベルまたは、ステップ
へジャンプします。

この様な2重使用をしなければならない場合は、2回目の”WHILE”文
のかわりに”IF”,”ADD”コマンドでカウントアップするプログラムを
設定してください。もしくは、WHILERコマンドを使用ください。
前記<文例>を例にすると以下の様になります。

<文例>

```
MOVE 0,V1 ;0→V1
MOVE 0,V2 ;0→V2
L2 IF V2,10,GT,L1
ADD 1,V1,V1 ;V1→V1+1
ADD 1,V2,V2 ;V2→V2+1
GOTO L2
L1
```

(コマンド制約事項)

また”WHILE~GOTO”の中で”CALL”,”GOTO”,”GOSUB”,”IF”,”TBIT”等の
ジャンプコマンドを書いて他のファイル、他のルーチンにジャンプ
した場合、プログラムの動作に関しては保証しかねますので、この
様なプログラムを作成しないでください。

上記の様なプログラムを作成する場合は、”WHILER”コマンドを使用
してください。

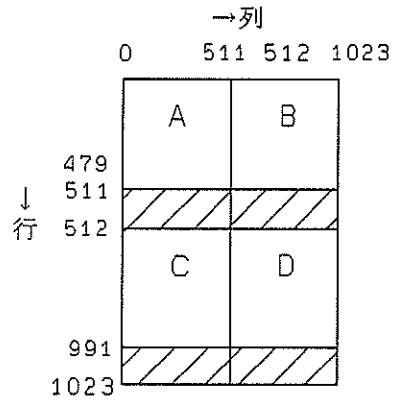
WHILER (～GOTO)		フォワイル リード (～ゴートウ)
機能	WHILERからGOTOまでの区間中にある命令を繰り返して実行しカウントアップを行う	
書式	WHILER Δ v , $\begin{bmatrix} v \\ i \end{bmatrix}$, $\begin{bmatrix} v \\ i \end{bmatrix}$, $\begin{bmatrix} 1 \\ i \end{bmatrix}$	
角平 言兌	<p>” WHILER” から” GOTO” までの区間中にある命令を「初期値」を指定されたvレジスタの値を「増分」だけ加減算し、このときvレジスタの値は加減算された値になります。「終値」を超えると指定されたラベルまたはステップへジャンプします。</p> <p>第1オペランド. 「初期値」をvレジスタで設定します。</p> <p>第2オペランド. 「終値」を設定します。</p> <p>第3オペランド. 「増分」を指定します。(0は指定不可)</p> <p>第4オペランド. 終了したときのジャンプ先を指定します。 ラベル : 英大文字で始まる2文字の英数字 ステップ数: -63～63(0を除く)</p> <p>「終値」を越えるというのは、</p> <ul style="list-style-type: none"> ・「増分」が正の時、「終値」より大きくなる時 ・「増分」が負の時、「終値」より小さくなる時 <p>を意味します。</p> <p>実行した時、以下の場合には実行せずに、指定されたラベル、またはステップへジャンプします。</p> <ul style="list-style-type: none"> ・「増分」が正の値で、「初期値」が「終値」より大きい場合 ・「増分」が負の値で、「初期値」が「終値」より小さい場合 <p><文例></p> <pre> 10: MOVE 0,V1 ;0→V1 11: L1 WHILER V1,10,1,NS ;初期値を"0"に設定。増分"1"を"V1"に加算しながら ;30ｽﾚｯﾄﾞ'まで実施 ;V1の値が"10"になりますと31ｽﾚｯﾄﾞ'目の"NS"に ;ジャンプします。 20: MOVE V1,V2 ;V1→V2 ;ここでは11ｽﾚｯﾄﾞ'～30ｽﾚｯﾄﾞ'までのループを何回 ;ループしているかをV2に格納することができます。 30: GOTO L1 31: NS </pre> <p><注意> このコマンドは、Ver 3. 0 P, 3. 0 R P以降に有効なコマンドです。</p>	

W S C A N		ダブルスキャン
機 能	ウィンドウチェッカの検査の実行	
書 式	$W S C A N \quad \Delta \quad \begin{bmatrix} v \\ i \end{bmatrix}, \begin{bmatrix} v \\ i \end{bmatrix}, \begin{bmatrix} (v) \\ (i) \end{bmatrix}, \begin{bmatrix} (v) \\ (i) \end{bmatrix}$ <p style="text-align: right;">() は省略可です。</p>	
角 準 言 説	<p>第1オペランドで指定されたウィンドウチェッカNO. から第2オペランドで指定されたウィンドウチェッカNO. までの検査を実行します。</p> <p>このとき、第3・第4オペランドでは、それぞれx方向、y方向の平行移動量を画素数で指定します。移動量が指定されているとき、「グループ選択」で指定されたグループの補正量と移動量の和の値だけウィンドウチェッカを移動し、検査を実行します。</p> <p>第3・第4オペランドが「省略」または「0」が指定されているとき、「グループ選択」で指定されたグループの補正量だけ移動し、検査を実行します。また、ウィンドウチェッカの検査条件（判定条件、グループ選択）は、品種モードで設定された条件により実行されます。</p> <p>グループの補正量による移動は行なわず、数値データ等の移動量のみ移動を行なう場合、移動量の算出の際に位置補正の補正量の分を差引いておいてください。</p> <p>グループの補正量は、グループNO. で指定された位置補正の補正量によりますので、グループNO. を指定された位置補正が実行されないとき、補正量は電源を切らない限りメモリに残っているため、前回実行されたときの補正量により実行されます。</p> <p>(参考)</p> <p>1) 品種モードで、「チェッカパターン表示選択」を「位置補正に追従して表示」を選択している場合、チェッカパターンの消去・描き込みに多少時間を要します。それでは問題のある場合、「固定位置に表示」又は「表示しない」を選択して時間を短縮してください。</p>	

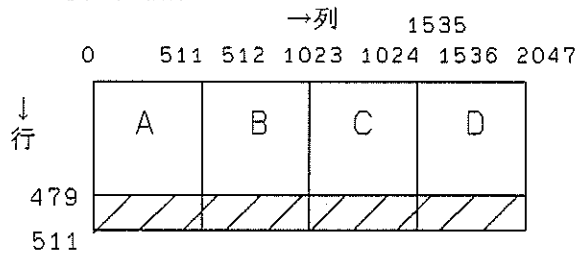
角 説

2) 移動量を入力する際、画面座標は右図のようになっていますので、移動方向の符号に御注意ください。
画面構成は実行ブロックでの設定によります。

田の字型構成



横一列構成



<文例>

WSCAN 5,8,100,200 ; ウィンドウチェッカNO.5~NO.8を
グループの補正量により補正後、
(Δx , Δy) = (100,200) だけ
平行移動し、検査を実行します。

X O R	イクスクルーシブオア															
機 能	排他的論理和															
書 式	$\text{XOR } \Delta \begin{pmatrix} v \\ i \\ h \end{pmatrix}, \begin{pmatrix} v \\ i \\ h \end{pmatrix}, v$															
角 説	<p>第1オペランドの内容と、第2オペランドの内容との各ビット毎の排他的論理和を行ない、演算結果を第3オペランドで指定したvレジスタに格納します。</p> <p>排他的論理和の演算を下表に示します。</p> <table border="1" data-bbox="497 945 1174 1272"> <thead> <tr> <th>第1オペランドの内容</th> <th>第2オペランドの内容</th> <th>第3オペランドに格納される内容</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> </tr> </tbody> </table> <p><文例> XOR V1, V2, V3 ; (V1 exculsive or V2) →V3</p> <p>XOR：排他的論理和は、各レジスタの各ビット間の排他的論理和となります。</p>	第1オペランドの内容	第2オペランドの内容	第3オペランドに格納される内容	0	0	0	0	1	1	1	0	1	1	1	0
第1オペランドの内容	第2オペランドの内容	第3オペランドに格納される内容														
0	0	0														
0	1	1														
1	0	1														
1	1	0														

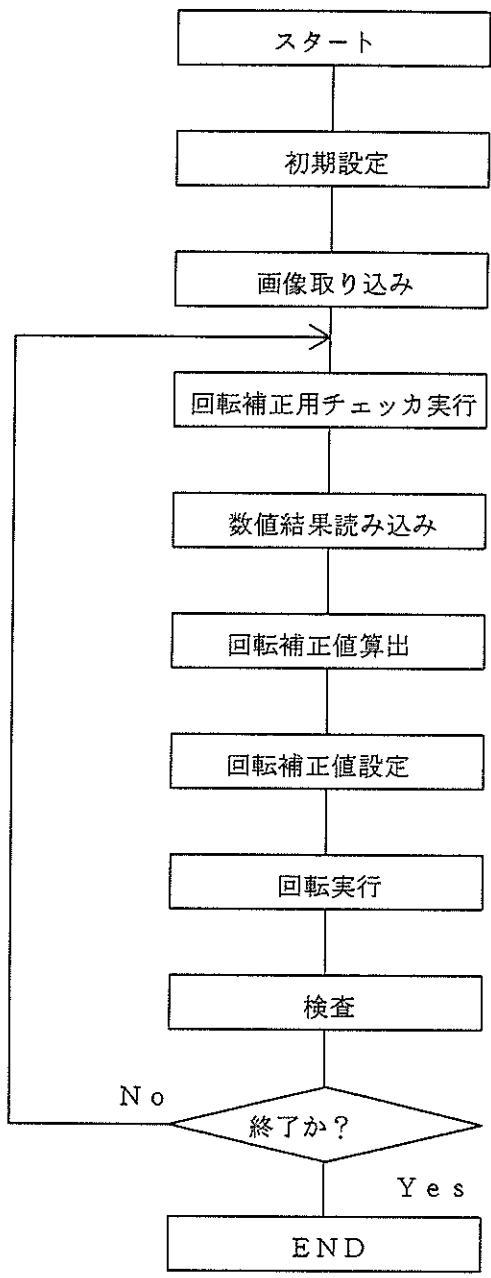
イメージチェッカ30P・30RPプログラム例

このプログラム例では、30Pのコントローラに必要な分解能補正（水平、垂直分解能補正）を行っていません。補正例は、「ATAN」の文例を参照してください。30RPは、水平・垂直の分解能は、1：1ですので分解能補正を行う必要はありません。

プログラムモードで回転補正を実行する際の注意：30RPのみ

- (1) プログラム中で回転補正を実行しようとする場合、回転補正用設定値を設定するためにはメモリエリアの確保が必要です。メモリエリアは「9. 回転補正」の「1. プログラム入力」で少なくとも「画面外エリア色」の設定を行うことにより確保されますので、忘れずに設定してください。
また、プログラム実行中、回転補正のモード設定「三角比モード」か「主軸角モード」かを変更することが可能です。
これは”TPRIN”または”TWIDTH” コマンドを実行することで行えます。
尚、”TSCAN” コマンドにより回転を実行する際、最後に設定されたモードにより回転を行いますのでご注意ください。
プログラム実行中回転補正のモードを変更したとしても品種モードでは回転補正を設定したときのモードで動作します。
- (2) 回転補正用設定値は、設定ブロックでの「9. 回転補正」の「1. プログラム入力」で設定された設定値を読み込んで、回転補正を実行します。
(イメージチェッカ30Rマニュアル参照)
プログラム中で回転補正用設定値をコマンド(TWIDTH、TANG、TPRIN、TCENT、TSHIFT)により設定されたときは、その設定値で回転を実行します。
- (3) 'SCAN' コマンドで実行すると、30Rと同様の動作をします。
- (4) ”READ”、”READF”、”ESCAN” コマンドによる画像取り込み後、”TSCAN” コマンドでの回転補正を回転補正用設定値を変更して、2回以上連続して行う場合にご注意ください。
イメージチェッカ30RPでは画像メモリが検査実行用と回転補正用の2メモリが用意されています。モニタに表示されているのは検査実行用のメモリです。
”TSCAN” コマンドで回転を実行する場合、回転補正用画像メモリからデータを読み込んで回転を行い検査用画像メモリに書き込みます。この時、回転補正用メモリは保存されています。
”TSCAN” で回転実行後、回転用補正チェッカを実行し回転補正設定値を設定して再度、”TSCAN” を実行した場合、回転は検査実行用画像メモリのデータを使用せずに回転補正用画像メモリのデータを読み込んで行われるため、モニタに表示されている画像とは違う回転を実行しますのでご注意ください。
この様な場合は、再度画像の読み込みを行うか回転補正用チェッカを最初に実行しておいてください。

イメージチェッカ 30P・30RPのプログラムのフローチャートの一例



注釈 ①
注釈 ②
注釈 ③

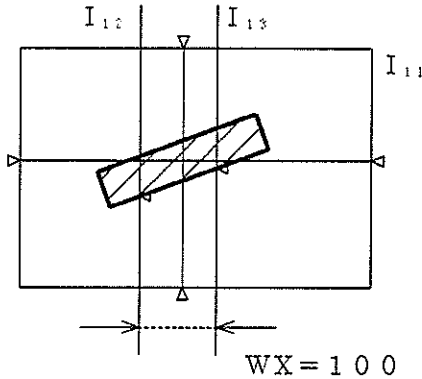
回転補正のための
フローチャート

イメージチェッカ
30RPのみの
機能です。

- 注釈 ①：回転角 ($T\theta$)、水平・垂直差分 (WX、WY)、移動量 (SW、SY) の算出
- 注釈 ②：TANG、TWIDTH、TCENT、TSHIFT、TPRINコマンドによる設定
- 注釈 ③：TSCANコマンドによる実行

三角比モードのプログラム例：30RPのみ

30RPで、30Rと同じ動作を行うプログラムです。
 位置補正チェックは、No11, No12, No13の設定を行っています。
 水平差分：WX=100での設定例です。



I₁₁ : 水平垂直位置補正No. 11
 (センターエッジ検出)

I₁₂ : 垂直位置補正No. 12

I₁₃ : 垂直位置補正No. 13

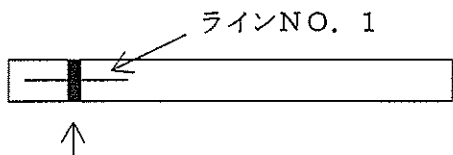
【プログラム】

- | | | | |
|------|-----------------|---|--|
| 1 : | CLRCRT | : | 画面消去 |
| 2 : | CLRREG V1,V1000 | : | V1~V1000レジスタをクリア (=0) します。 |
| 3 : | CLRREG A1,A1000 | : | A1~A1000レジスタをクリア (=Null) します。 |
| 4 : | L1 READ "A***" | : | Aメモリに画像取り込みをします。 |
| 5 : | ISCAN 11,13 | : | 位置補正チェックをチェックNo11~No13を実行します。 |
| 6 : | MOVE CI111,V1 | : | 位置補正チェックNo11のセンタ検出結果のx座標をV1に転送します。 |
| 7 : | MOVE CI112,V2 | : | 位置補正チェックNo11のセンタ検出結果のy座標をV2に転送します。 |
| 8 : | MOVE CI122,V3 | : | 位置補正No12の垂直エッジ検出位置データをV3に転送します。 |
| 9 : | MOVE CI132,V4 | : | 位置補正No13の垂直エッジ検出位置データをV4に転送します。 |
| 10 : | SUB V4,V3,V5 | : | 垂直差分WYを算出し、V5に転送します。
(V4-V3→V5) |
| 11 : | TWIDTH 1,100,V5 | : | Aメモリの水平差分 (WX) = 100、垂直差分 (WY) = V5に設定します。 |
| 12 : | TANG 1,20 | : | Aメモリの回転角 (Tθ) = 20に設定します。 |
| 13 : | TCENT 1,V1,V2 | : | Aメモリの回転中心 (CX) = V1、回転中心 (CY) = V2に設定します。 |
| 14 : | TSHIFT 1,30,40 | : | Aメモリの移動量 (SX) = 30、移動量 (SY) = 40に設定します。 |
| 15 : | TSCAN "A***",1 | : | 回転補正用チェックを実行せずに回転を実行します。 |
| 16 : | L2 KEYIN A1 | : | キーボードから1文字読み込み、A1に格納します。 |
| 17 : | IF A1,'Y',EQ,RE | : | A1 = 'Y' ならば 'RE' へジャンプします。 |
| 18 : | IF A1,'N',EQ,L1 | : | A1 = 'N' ならば 'L1' へジャンプします。 |
| 19 : | GOTO L2 | : | L2へジャンプします。 |
| 20 : | RE RET | : | プログラムを終了します。 |

[プログラム例. 1]

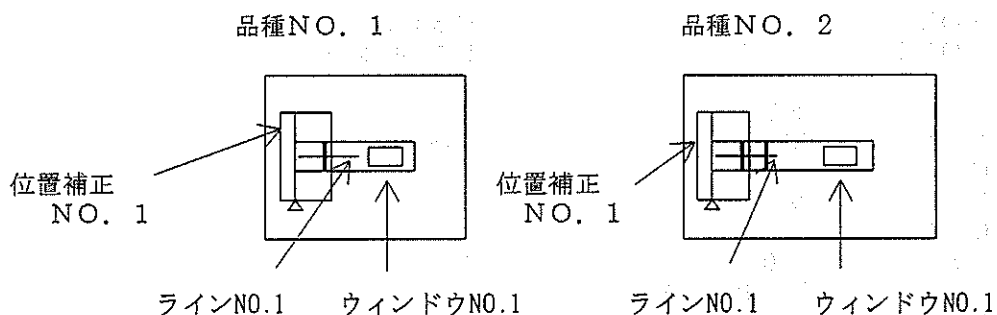
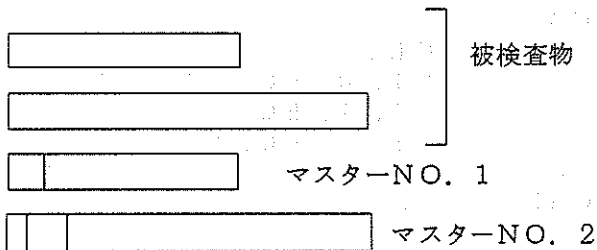
検査物に合わせ、自動で品種切替え

帯数により被検査物、マスターNO. 1、マスターNO. 2、を見分け、マスターなら、品種切替えを行ないます。検査はウィンドウNO. 1のカウンタ値の判定結果により行います。

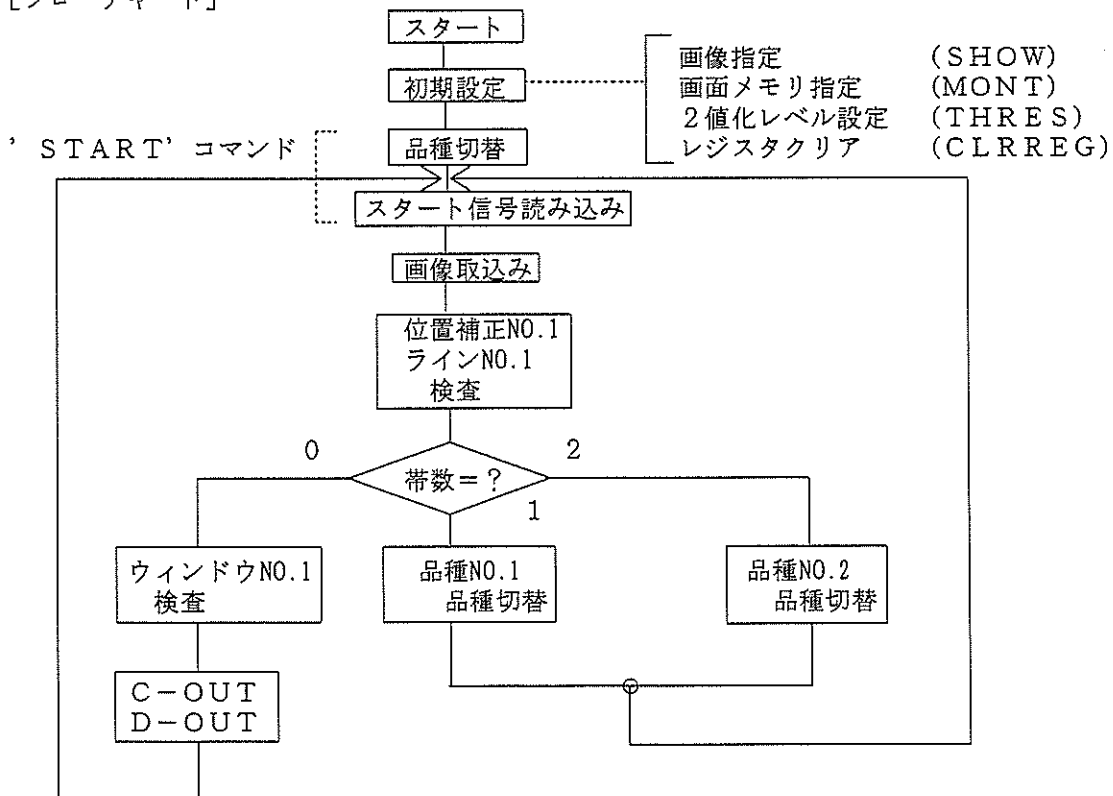


マスターはこの位置にライン（黒帯）が引かれています。ラインNO. 1で帯数をカウントします。

帯数	品種
0	被検査物
1	NO. M1
2	NO. M2



[フローチャート]



数値演算 C500=W1

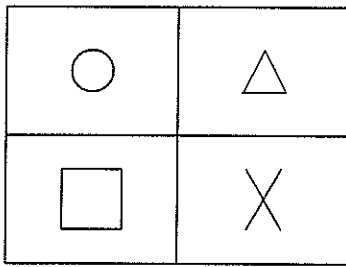
判定出力 D1 =W1

はあらかじめ設定ブロックで設定しておきます。

```
1:      CLRCRT          ; 画面消去します。
2:      SHOW           " *BIMCP" ; リアルタイム2値化画像・メッセージ・カーソル・
                                     パターン
                                     を表示します。
3:      MONT           " A"      ; Aメモリを表示します。
4:      THRES          A, 32     ; Aメモリの2値化レベルを32にします。
5:      CLRREG         V1, V1    ; V1をクリアします。
6: ST START          ; スタート信号・品種切替信号の読み込み
7:      READ           " A***"   ; Aメモリに画像取込みをします。
8:      ISCAN          1, 1      ; 位置補正NO. 1を実行します。
9:      LSCAN          1, 1      ; ラインNO. 1を実行します。
10:     MOVE           CL12, V1   ; ラインNO. 1の帯数のカウント値→V1
11:     IF              V1, 0, EQ, L1 ; V1=0ならば, L1へジャンプ
12:     IF              V1, 1, EQ, L2 ; V1=1ならば " L2" "
13:     IF              V1, 2, EQ, L3 ; V1=2ならば " L3" "
14:     GOTO            ER        ; 'ER'へジャンプ
15: L1 WSCAN          1, 1      ; ウィンドウNO. 1を実行
16:
17:
18:     C-OUT           ; C500をパラレルアウト
19:     D-OUT           1, 1     ; D1の結果をポートNO. 2からハンドシェーク
                                     しながら、パラレルアウト
20:     GOTO            ST        ; 'ST'へジャンプ
21: L2 TYPE           M, 1      ; 品種NO. M1に品種切替
22:     GOTO            ST
23: L3 TYPE           M, 2
24:     GOTO            ST
25: ER DISP           10, 10,    ; 10列、10行を先頭に、" ERROR"と表示
                                     " ERROR"
26:     KEYIN           A1       ; キー入力待ち、キー入力があるとA1に格納し、
                                     次のステップへ
27:     DISP            10, 10,   ; 10列、10行を先頭に" " (空白)を5文字
                                     " ", 5
                                     表示
28:     GOTO            ST
```

[プログラム例. 2]

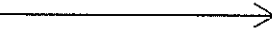
2値化レベルを変えて画像を読み込み



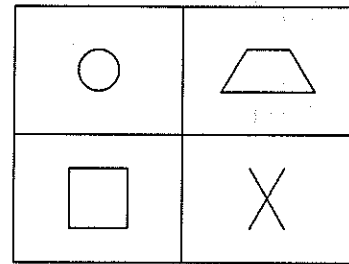
A, B, C, Dメモリに
画像取込み

メモリ	2値化レベル
A	3 2
B	3 0
C	2 5
D	3 5

Bメモリのみ読み込み



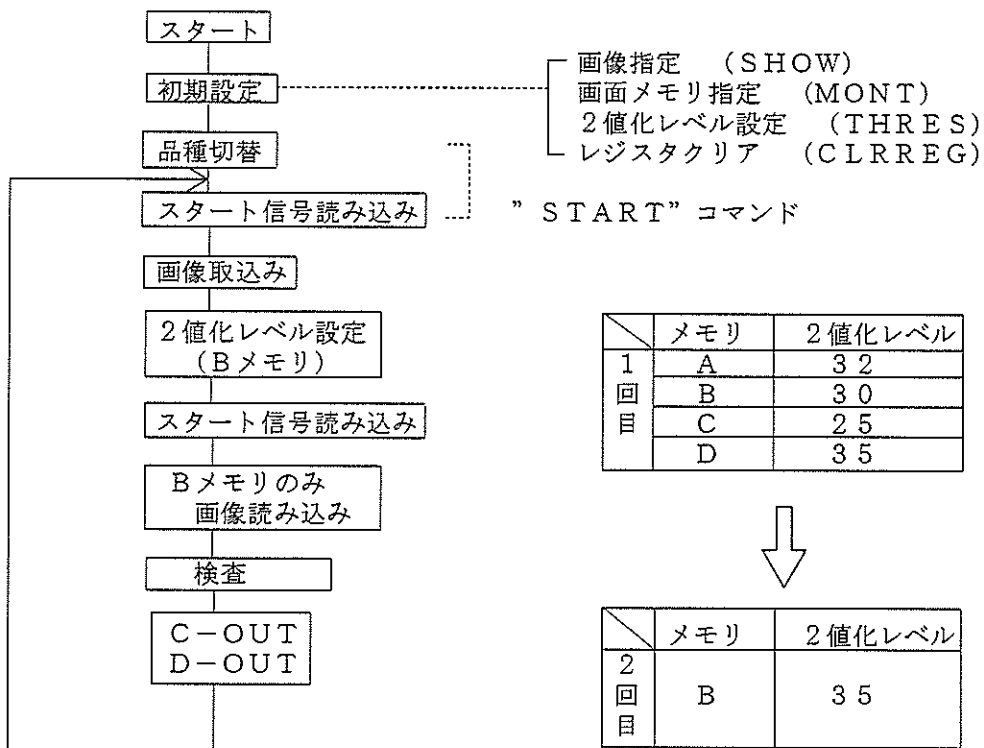
2値化レベル
を変えることが
できます。



メモリA, C, Dは、前の2値化レベルで
の画像が残っています。

メモリ	2値化レベル
A	3 2
B	3 5
C	2 5
D	3 5

メモリBの2値化レベル
を3 5に変更

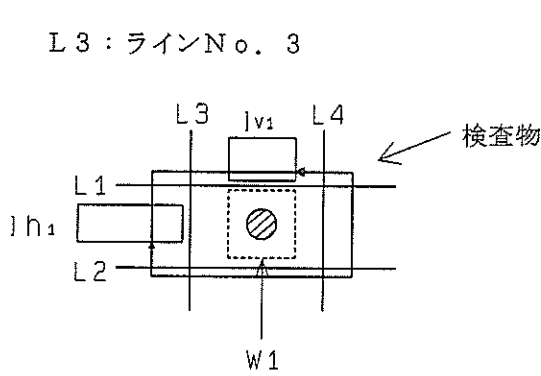


各チェック、数値演算式、判定式は、あらかじめ設定ブロックにて、設定してください。

```
1 :      CLR CRT          ; 画面をクリア
2 :      MONT             " A"          ; Aメモリを画面表示
3 :      SHOW             " *BIMCP"
4 :      THRES            A, 32         ; Aメモリの2値化レベルを32にします。
5 :      THRES            B, 30         ; B " " " 30 "
6 :      THRES            C, 25         ; C " " " 25 "
7 :      THRES            D, 35         ; D " " " 35 "
8 : ST   START           ; スタート信号、品種切替信号の読み込み
9 :      READ             " ABCD"       ; A, B, C, Dメモリに画像取り込みをします。
10 :     THRES            B, 35         ; Bメモリの2値化レベルを35にします。
11 :     START
12 :     READ             " *B**"       ; Bメモリから画像を取り込みます。
13 :     SCAN
14 :     C-OUT
15 :     D-OUT            1, 1          ; 数値演算結果をパラレルアウトします。
16 :     GOTO             ST            ; 判定出力結果をパラレルアウトします。
```

[プログラム例. 3]

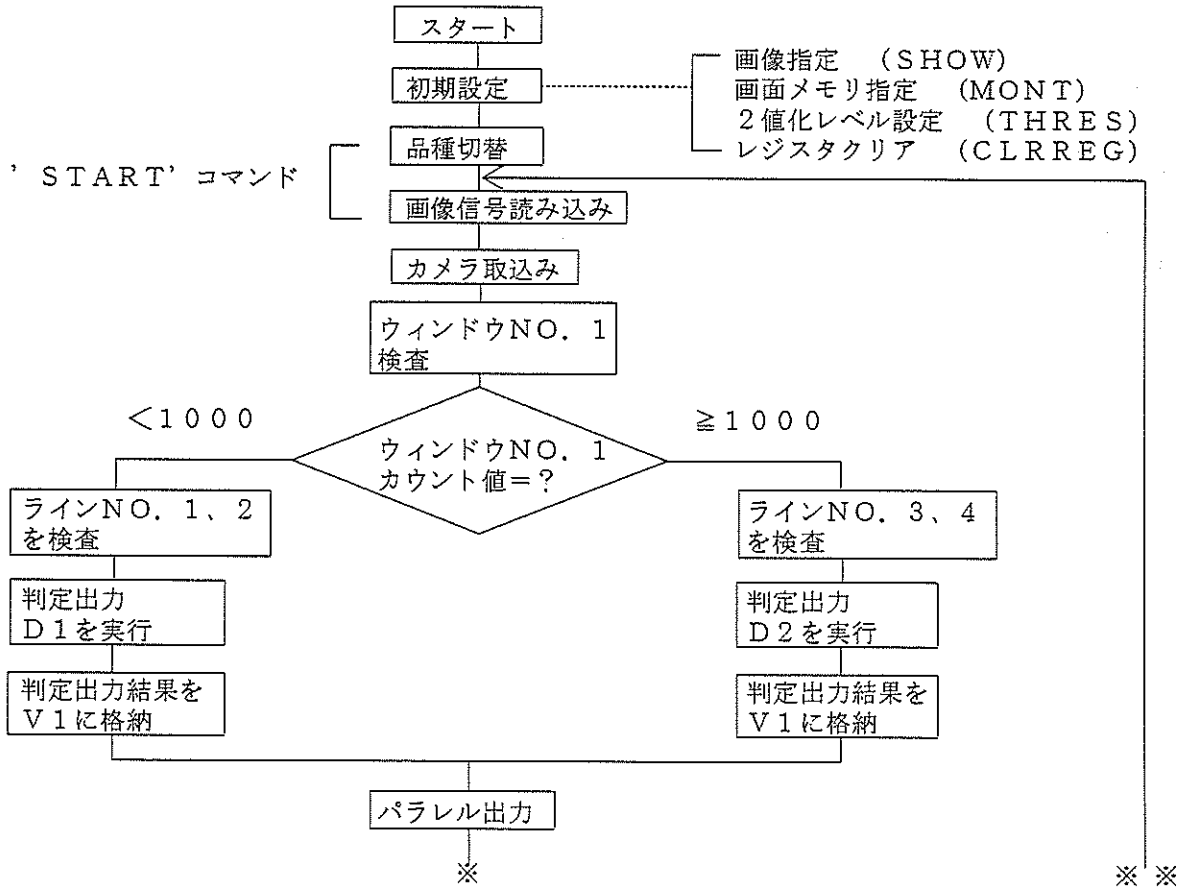
チェッカ判定結果により、検査チェッカを自動切替え

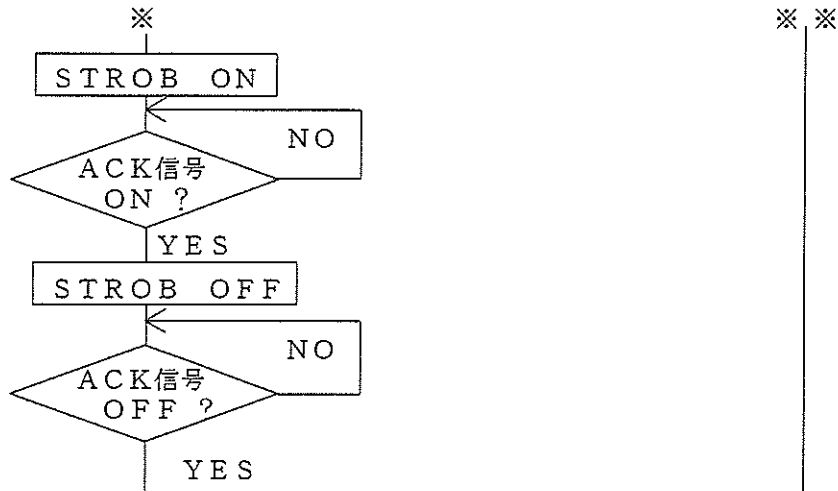


- L1 : ラインNo. 1
- L2 : ラインNo. 2
- L3 : ラインNo. 3
- L4 : ラインNo. 4
- W1 : ウィンドウNo. 1
- lh1 : 水平位置補正No. 1
- lv1 : 垂直位置補正No. 2

条件	W1のマークが ▲ の時 (W1<1000)	W1のマークが ● の時 (W1≥1000)
検査内容	L1、L2を検査します。 判定出力: D1=L11*L21 D1をパラレル出力します。	L3、L4を検査します。 判定出力: D2=L31*L41 D2をパラレル出力します。

[フローチャート]

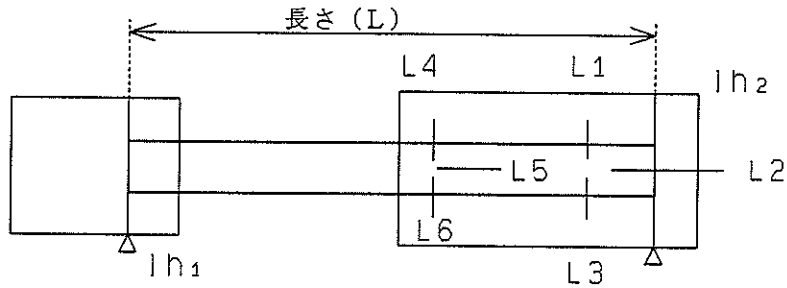




1 :	CLRCRT		
2 :	SHOW	" **IMCP"	; 2値化画像・メッセージ・カーソル・パターンを ; 表示
3 :	MONT	" A"	; Aメモリを画面表示
4 :	THRES	A, 30	; Aメモリの2値化レベルを30に設定
5 :	CLRREG	V1, V4	; V1~V4をクリア
6 : ST	START		
7 :	READ	" A***"	
8 :	ISCAN	1, 1	; 位置補正NO. 1を実行
9 :	WSCAN	1, 1	; ウィンドウNO. 1を実行
10 :	MOVE	CW1, V2	; ウィンドウNO. 1のカウンタ値→V2
11 :	IF	V2, 1000, LT, M1	; V2<1000ならば, M1へジャンプ
12 :	IF	V2, 1000, GE, M2	; V2≥1000ならば, M2へジャンプ
13 : M1	LSCAN	1, 2	; ライン NO. 1~NO. 2を実行
14 :	DSCAN	1, 1	; 判定出力 NO. 1を実行
15 :	MOVE	DD1, V1	; 判定出力 NO. 1の結果→V1
16 :	GOTO	L1	
17 : M2	LSCAN	3, 4	; ライン NO. 3~NO. 4を実行
18 :	DSCAN	2, 2	; 判定出力 NO. 2を実行
19 :	MOVE	DD2, V1	; 判定出力 NO. 2の結果→V1
20 : L1	P-OUT	1, V1	; ポート NO. 2からV1の値をパラレルアウト
21 :	STROB	ON	; 同期信号をONにします。
22 : L3	P-IN	1, V3	; ポート NO. 1からパラレルインし, V3に格納
23 :	TBIT	4, V3, ZR, L2	; V3の4ビットが' 0' (ON) ならL2へ ; ジャンプ
24 :	GOTO	L3	
25 : L2	STROB	OF	; 同期信号をOFFにします。
26 : L4	P-IN	1, V4	; ポート NO. 1からパラレルインし, V4に格納 ; します。
27 :	TBIT	4, V4, NZ, ST	; V4の4ビットが' 1' (OFF) ならSTへ ; ジャンプ
28 :	GOTO	L4	

[プログラム例, 4]

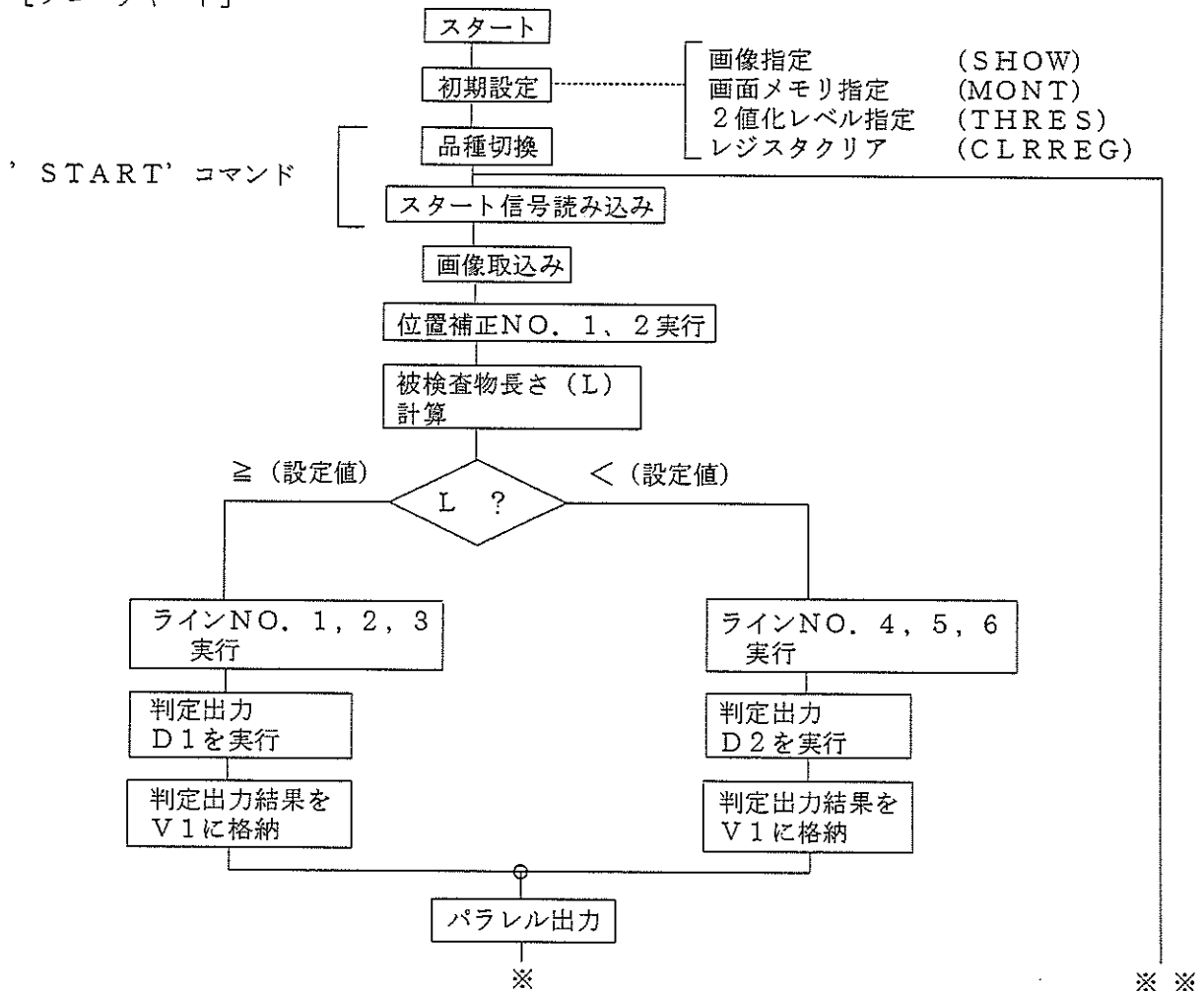
数値演算結果により、検査チェッカを自動切替え

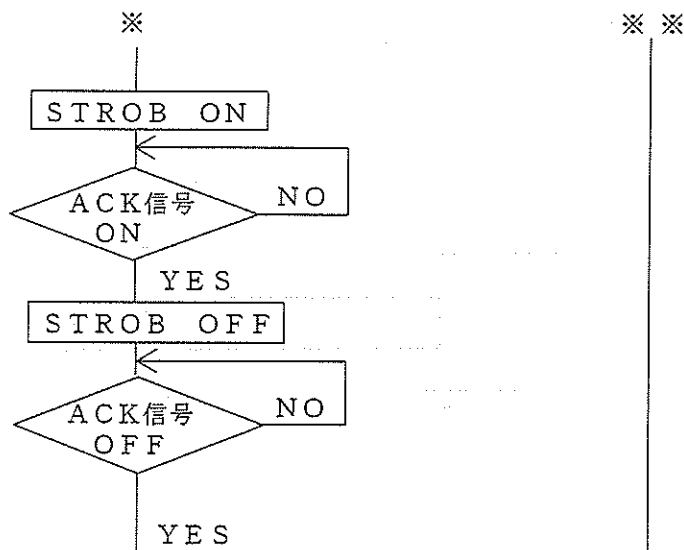


検査内容

1. 位置補正NO. 1、NO. 2の水平エッジ検出結果の差を求めることにより、被検査物の長さを求める。] $L = I21 - I11$
2. [
 - $L \geq (\text{設定値}) \rightarrow$ ラインNO. 1, 2, 3を検査
 - $L < (\text{設定値}) \rightarrow$ ラインNO. 4, 5, 6を検査
3. 判定結果を出力

[フローチャート]



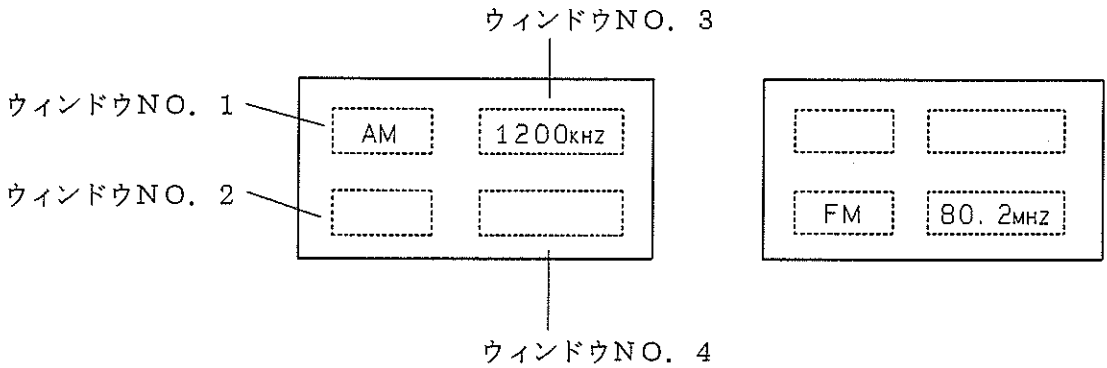


```

1: CLRCRT
2: SHOW      " *BIMCP"
3: MONT      " A"
4: THRES     A, 30
5: CLRREG    V1, V4      ; V1~V4をクリアします。
6: ST START
7: READ      " A***"
8: ISCAN     1, 2
9: MOVE      CI11, V2    ; 位置補正NO. 1の水平エッジ検出値→V2
10: MOVE     CI21, V3    ; " NO. 2 " " →V3
11: SUB      V3, V2, V4  ; V3-V2→V4
12: IF       V4, 100, GE, M1 ; V4≥100→M1へジャンプ
13: IF       V4, 100, LT, M2 ; V4<100→M2へジャンプ
14: M1 LSCAN 1, 3      ; ラインNO. 1~NO. 3を実行
15: DSCAN   1, 1      ; 判定出力NO. 1を実行
16: MOVE    DD1, V1    ; 判定出力NO. 1の結果→V1
17: GOTO    L1         ; 'L1'へジャンプ
18: M2 LSCAN 4, 6      ; ラインNO. 4~NO. 6を実行
19: DSCAN   2, 2      ; 判定出力NO. 2を実行
20: MOVE    DD2, V1    ; 判定出力NO. 2の結果→V1
21: L1 P-OUT 1, V1    ; ポートNO. 2からV1の値をパラレルアウト
                       ; します。
22: STROB   ON        ; 同期信号を'ON'します。
23: L3 P-IN  1, V3    ; ポートNO. 1からパラレルインし、V3に格納
                       ; します。
24: TBIT    4, V3, ZR, L2 ; V3の4ビットが、'0' (ON)ならば
                       ; 'L2'へジャンプ
25: GOTO    L3
26: L2 STROB OF       ; 同期信号を'OFF'します。
27: L4 P-IN  1, V4
28: TBIT    4, V4, NZ, ST ; V4の4ビットが、'1' (OFF)ならば
                       ; 'ST'へジャンプ
29: GOTO    L4
  
```


[プログラム例. 5]

2つのチェッカ判定結果により、検査チェッカ・判定出力の自動切替え



(W1の判定結果) = 1 → W3を検査

(W2の判定結果) = 1 → W4を検査

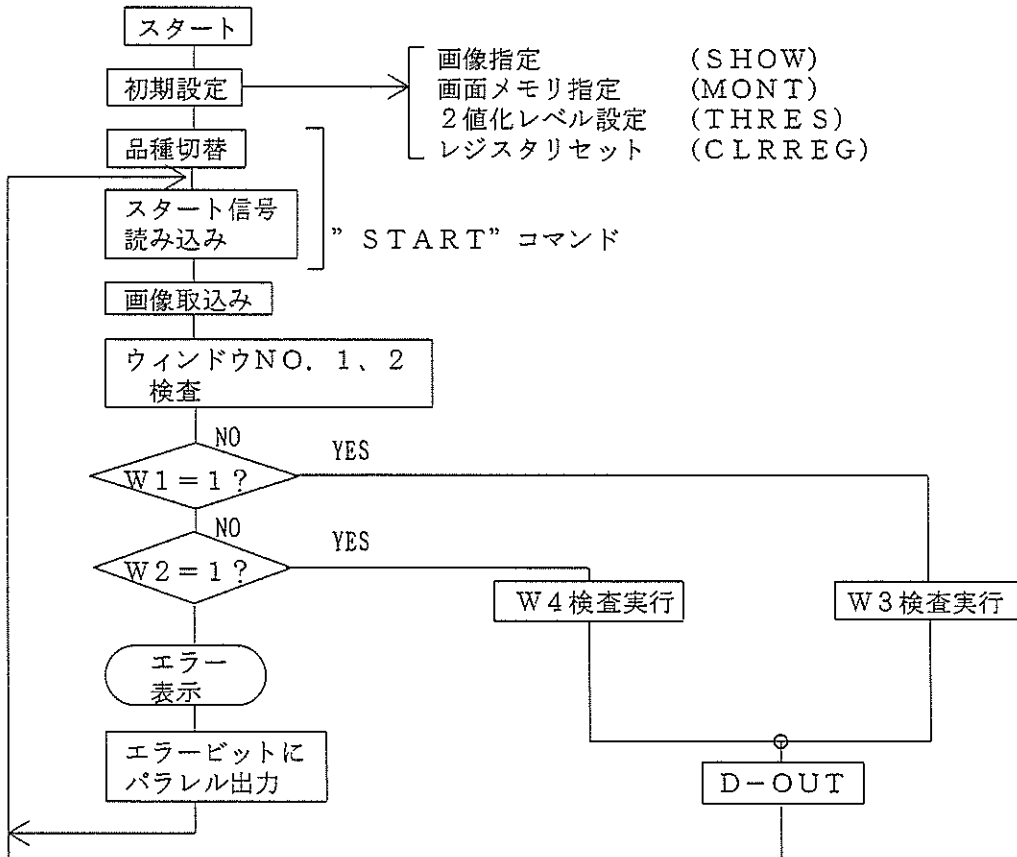
W1 = 0, W2 = 0 のとき
 ↓ エラー表示
 エラーです。(Vレジスタの4ビット(エラービットとします)を1にしてパラレル出力します。)

判定出力をパラレル出力します。

D1 = W1	}	" AM" のとき
D2 = W3		
D3 = W2	}	" FM" のとき
D4 = W4		

[出力の組み合わせ]

D1 = 1 → D2が判定結果
 D3 = 1 → D4が判定結果



判定出力

D1=W1
D2=W3
D3=W2
D4=W4

はあらかじめ設定ブロックにて設定しておきます。

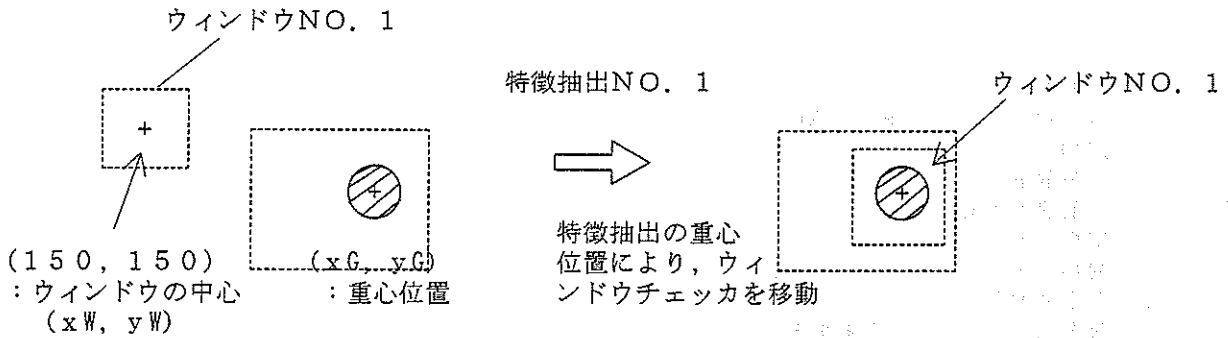
```

1:   CLRCRT
2:   SHOW   " *BIMCP"
3:   MONT   " A"
4:   THRES  A, 30
5:   CLRREG V1, V4
6: ST START
7:   READ   " A***"
8:   WSCAN  1, 2           ; ウィンドウNO. 1~NO. 2を実行します。
9:   MOVE   DW1, V1       ; ウィンドウNO. 1の判定結果 → V1
10:  MOVE   DW2, V2       ; " NO. 2 " → V2
11:  IF     V1, 1, EQ, M1  ; V1=1ならばM1へジャンプ
12:  IF     V2, 1, EQ, M2  ; V2=1ならばM2へジャンプ
13:  DISP   10, 10, " ERROR" ; 10列10行を先頭に " ERROR" と表示
14:  SBIT   4, V3         ; V3の4ビットをセット (=1) します。
15:  P-OUT  1, V3         ; ポートNO. 1からV3の値をパラレルアウトします。
16:  STROB  ON           ; 同期信号を ' ON' します。
17: L2 P-IN  1, V4       ; ポートNO. 1からパラレルインしV4に格納し
                           ; ます。
18:  TBIT   4, V4, ZR, L1 ; V4の4ビットが ' 0' (ON) なら、' L1' へ
                           ; ジャンプ
19:  GOTO   L2           ; ' L2' へジャンプ
20: L1 STROB OF         ; 同期信号を ' OFF' します。
21:  MOVE   0, V4       ; 0 → V4
22:  DISP   10, 10, " ", 5 ; 10列10行を先頭に、" " (空白) を5文字表示
23:  GOTO   ST         ; ' ST' へジャンプ
24: M1 WSCAN  3, 3     ; ウィンドウNO. 3を実行
25:
26:  GOTO   DO         ; DOへジャンプ
27: M2 WSCAN  4, 4     ; ウィンドウNO. 4を実行
28:
29: DO D-OUT  1, 1     ; ポートNO. 1から、判定出力の結果をハンドシェー
                           ; クしながらパラレルアウトします。
30:  GOTO   ST

```

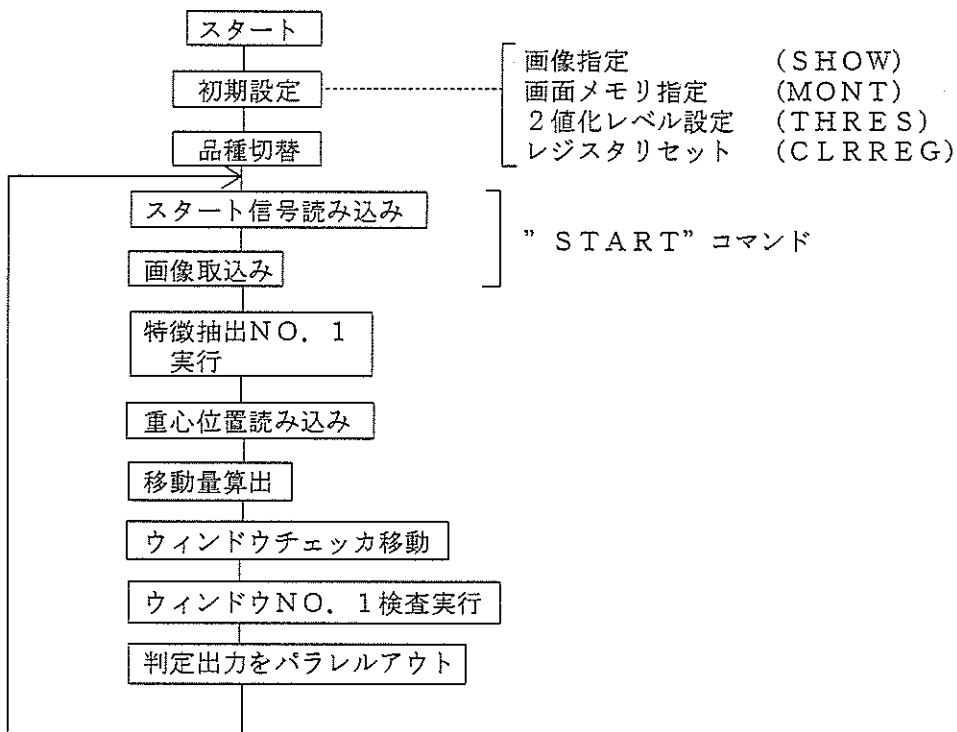
[プログラム例. 6]

ウィンドウを任意に移動 (特徴抽出での重心位置へ移動)

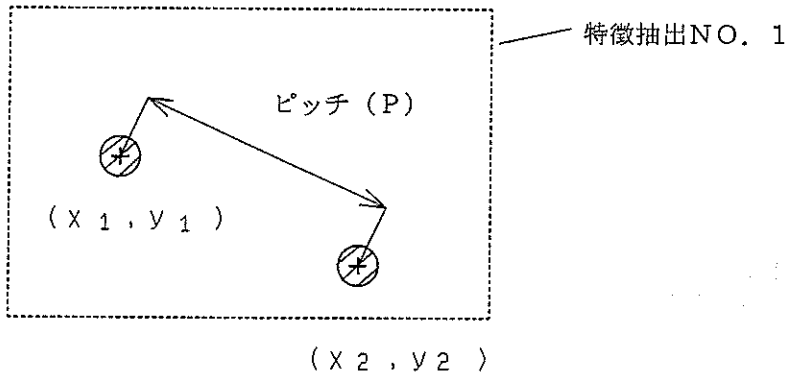


移動量の算出式 $(\Delta x, \Delta y) = ((xG - xW), (yG - yW))$
 但し、位置補正による補正がされないとき
 位置補正による補正がある場合は、 Δx 、 Δy ともに、補正量を
 差引いて下さい。

[フローチャート]



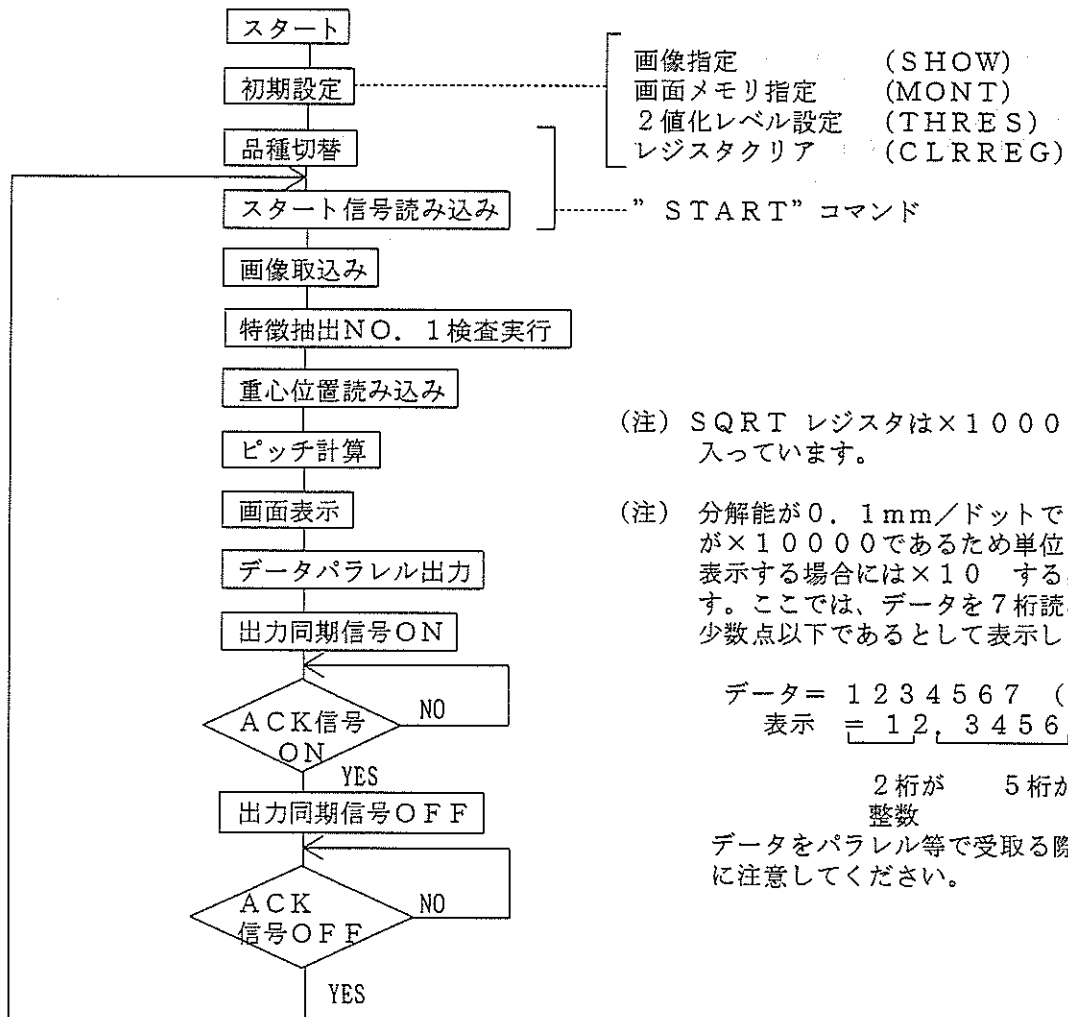
[プログラム例. 7]
ピッチ間隔の測定・パラレル出力



$$\text{ピッチ (P)} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

分解能 = 0.1 mm / ドットとする。

[フローチャート]



```

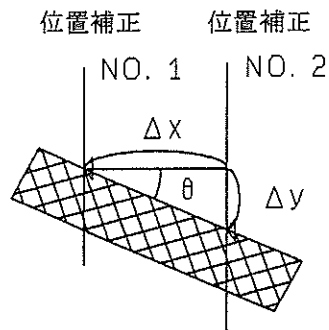
1 : CLRCRT
2 : SHOW " *BIMCP"
3 : MONT " A"
4 : THRES A, 32
5 : CLRREG V1, V10
6 : ST START
7 : READ " A***"
8 : SCAN
9 : MOVE CF112, V1 ; 特徴抽出NO. 1、対象NO. 1のx座標の値 → V1
10 : MOVE CF113, V2 ; " " " y座標の値 → V2
11 : MOVE CF122, V3 ; " " " NO. 2 x座標の値 → V3
12 : MOVE CF123, V4 ; " " " y座標の値 → V4
13 : SUB V3, V1, V5 ; V3-V1 → V5
14 : SUB V4, V2, V6 ; V4-V2 → V6
15 : MUL V5, V5, V7 ; V5×V5 → V7
16 : MUL V6, V6, V8 ; V6×V6 → V8
17 : ADD V7, V8, V9 ; V7+V8 → V9
18 : SQRT V9, V10 ; √V9 → V10
19 : MOVE V10, A1, 7 ; V10の値7個 → A1~A7
20 : DISP 20, 10, ; 20列10行を先頭に"ピッチ="と表示
    "ピッチ="

21 : DISP A1, 2 ; A1~A2を表示
22 : DISP ". " ; ". "を表示
23 : DISP A3, 5 ; A3~A7を表示
24 : DISP " mm" ; " mm"と表示
25 : P-OUT 3, V10 ; ポートNO. 1, 2からV10の内容をパラレルアウト
26 : STROB ON ; 同期信号を'ON'します。
27 : L1 P-IN 1, V101 ; ポートNO. 1からパラレルインしV101に格納
    します。
28 : TBIT 4, V101, ZR ; V101の4ビットが'0' (ON)なら'L2'へ
    , L2 ジャンプ
29 : GOTO L1 ; 'L1'へジャンプ
30 : L2 STROB OF ; 同期信号を'OFF'します。
31 : L4 P-IN 1, V102 ; ポートNO. 1からパラレルインし、V102に格納
    します。
32 : TBIT 4, V102, NZ ; V102の4ビットが'1' (OFF)なら'L3'
    , L3 へジャンプ
34 : L3 DISP 20, 10, " ", 18 ; 20列10行を先頭に" " (空白)を18個表示
35 : GOTO ST ; 'ST'へジャンプ

```

[プログラム例. 8]

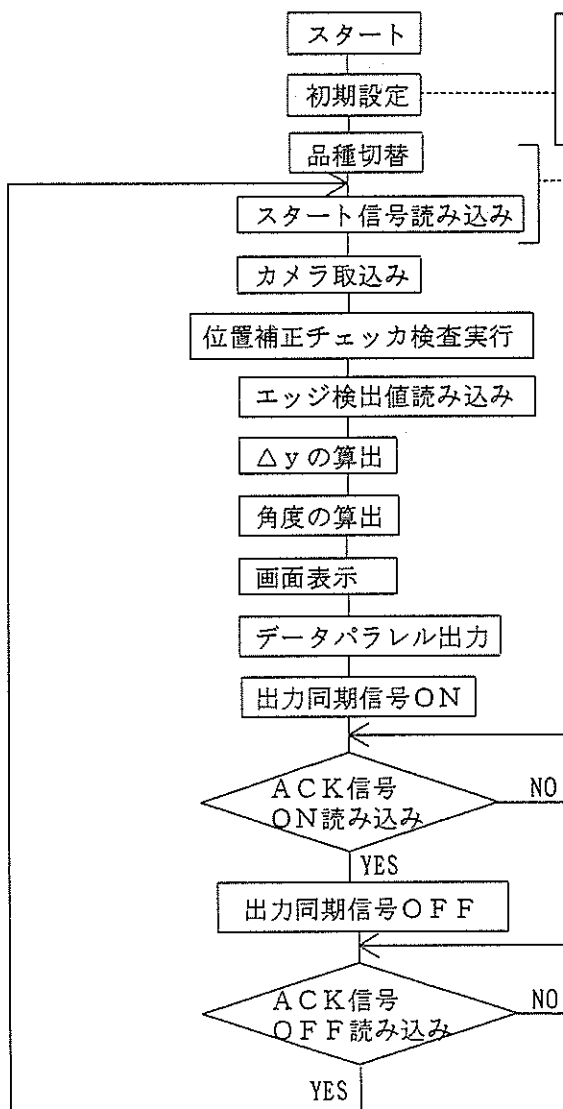
傾き角度の測定・パラレル出力



Δx : チェッカ設定のときに計算できます。
(例題では 200)
Δy : エッジ検出結果の差
($I_{22} - I_{12}$)

$$\text{角度}(\theta) \tan^{-1} \frac{\Delta y}{\Delta x}$$

[フローチャート]



画像指定 (SHOW)
画面メモリ指定 (MONT)
2値化レベル設定 (THRES)
レジスタクリア (CLRREG)

"START" コマンド

(注) "ATAN" コマンドの入力は、×10000
結果は×100 (度) で算出されます。

(注) 角度の算出結果を、5桁とり、上3桁を整数
部分下2桁を小数点以下とします。

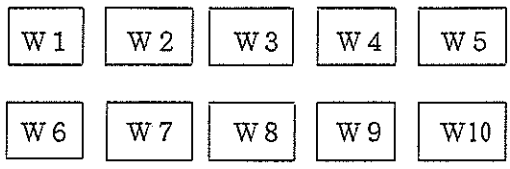
```

1: CLRCRT
2: SHOW " *BIMCP"
3: MONT " A"
4: THRES A, 32
5: CLRREG V1, V10
6: ST START
7: READ " A***"
8: SCAN
9: MOVE CI12, V1 ; 位置補正NO. 1の垂直エッジ検出値 → V1
10: MOVE CI22, V2 ; " NO. 2の " → V2
11: SUB V2, V1, V3 ; V2-V1 → V3
12: MUL V3, 10000, V4 ; V3×10000 → V4
13: DIV V4, 200, V5 ; V4÷200 → V5
14: ATAN V5, V6 ; tan-1(V5) → V6
15: MOVE V6, A1, 6 ; V6の値を6個 → A1~A6
16: DISP 20, 10, " 角度=" ; 20列10行を先頭に" 角度=" と表示
17: DISP A1, 4 ; A1~A4を表示
18: DISP ". " ; ". " を表示
19: DISP A5, 2 ; A5~A6を表示
20: DISP " 度" ; " 度" と表示
21: P-OUT 3, V6 ; ポートNO. 1、2からV6の内容をパラレルアウト
22: STROB ON ; 同期信号を' ON' します。
23: L1 P-IN 1, V7 ; ポートNO. 1からパラレルインし、V7に格納します。
24: TBIT 4, V7, ZR, L2 ; V7の4ビットが' 0' (ON) ならば' L2' へ
    ジャンプ
25: GOTO L1 ; ' L1' へジャンプ
26: L2 STROB OFF ; 同期信号を' OFF' します。
27: L4 P-IN 1, V8 ; ポートNO. 1からパラレルインし、V8に格納します。
28: TBIT 4, V8, NZ, L3 ; V8の4ビットが' 1' (OFF) ならば' L3' へ
    ジャンプ
29: GOTO L4 ; ' L4' へジャンプ
30: L3 DISP 20, 10, " ", 15 ; 20列10行を先頭に" " (空白) を15個表示
31: GOTO ST ; ' ST' へジャンプ

```

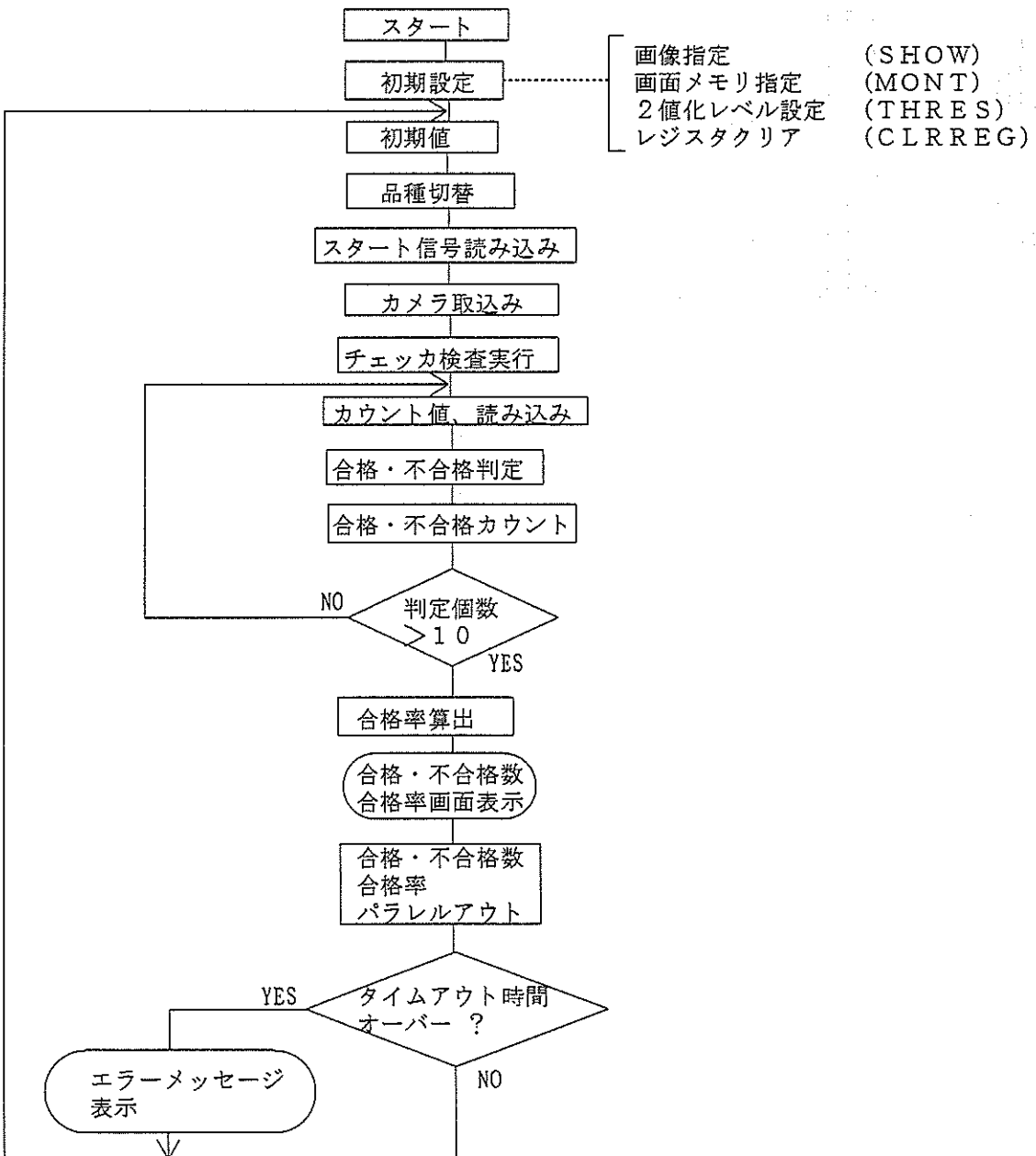

[プログラム例. 9]
検査物の合格率計算

10個のウィンドウのカウント数により、合否を判定します。
合格 \geq 1000、不合格 $<$ 1000、としそれぞれの個数をカウントします。
また、合格率を計算します。



$$\text{合格率} = \frac{\text{合格数}}{\text{全数}} \times 100 \quad (\%)$$

[フローチャート]



```

1:      CLRCRT
2:      SHOW      " *BIMCP"
3:      MONT      " A"
4:      THRES     A, 32
5:      CLRREG    V1, V10
6: ST MOVE      1, V101          ; 1 → V101
7:      CLRREG    V102, V105    ; V102、V105をクリア
8:      START
9:      DISP      20, 18, " ", 26
10:     READ      " A***"
11:     SCAN
12: L2 MOVE      CW[V101], (V101) ; V101の示すウィンドウNO. のカウント
                                ; 結果をV101の内容のNO. のvレジスタ
                                ; で格納します。
13:     ADD        V101, 1, V101  ; V101+1 → V101
14:     IF         V101, 10, GT, L1 ; V101>10 → ' L1' ヘジャンプ
15:     GOTO       L2             ; ' L2' ヘジャンプ
16: L1 MOVE      1, V101          ; 1 → V101
17: L4 IF        (V101), 1000, GE, OK ; (V101の内容のNO. のvレジスタ) ≥
                                ; 1000 → ' OK' ヘジャンプ
18:     ADD        1, V103, V103  ; 1+V103 → V103
19:     ADD        1, V101, V101  ; 1+V101 → V101
20:     IF         V101, 10, GT, L3 ; V101>10 → ' L3' ヘジャンプ
21:     GOTO       L4             ; ' L4' ヘジャンプ
22: OK ADD       1, V102, V102    ; 1+V102 → V102
23:     ADD        1, V101, V101  ; 1+V101 → V101
24:     IF         V101, 10, GT, L3 ; V101>10 → ' L3' ヘジャンプ
25:     GOTO       L4             ; ' L4' ヘジャンプ
26: L3 MUL       V102, 1000, V105 ; V102×1000 → V105
27:     DIV        V105, 10, V104  ; V105÷10 → V104
28:     MOVE       V102, A1, 2     ; V102の値を2個 → A1~A2
29:     MOVE       V103, A3, 2     ; V103の値を2個 → A3~A4
30:     MOVE       V104, A5, 4     ; V104の値を4個 → A5~A8
31:     DISP      20, 8, " 合格数=" ; 20列8行を先頭に" 合格数=" と表示
32:     DISP      A1, 2           ; A1~A2を表示
33:     DISP      20, 10, " 不合格数=" ; 20列10行を先頭に" 不合格数=" と表示
34:     DISP      A3, 2           ; A3~A4を表示
35:     DISP      20, 12, " 合格率=" ; 20列12行を先頭に" 合格率=" と表示
36:     DISP      A5, 3           ; A5~A7を表示
37:     DISP      " . "          ; " . " を表示
38:     DISP      A8, 1           ; A8を表示
39:     DISP      " %"           ; " %" を表示
                                MOVEWB V104, V104 ; V104の下位16ビットのうち上位8ビッ
                                ; ットをV104に下位8ビットをV105
                                ; に格納します。
                                ; 102 → V106
40:     MOVE       102, V106      ; ポートNO. 2からV106の内容のNO.
41: B5 P-OUT     1, (V106)        ; のvレジスタをパラレルアウト
42:     STROB      ON            ; スローブ ON
43:     TMWT       0             ; タイマーを0msecにセット
44: B2 P-IN     1, V107          ; ポートNO. 1からパラレルインしV107
                                ; に格納します。
45:     TBIT       4, V107, ZR, B1 ; V107の4ビットが' 0' (ON) ならば
                                ; ' B1' ヘジャンプ
46:     TMRD       V109          ; タイマーの値をV109に格納します。
47:     IF         V109, 100, GT, ER ; V109>100 → ' ER' ヘジャンプ
48:     GOTO       B2            ; ' B2' ヘジャンプ
49: B1 STROB     OF              ; 同期信号を' OFF' します。
50:     TMWT       0             ; タイマーを0msecにセット
51: B4 P-IN     1, V108          ; ポートNO. 1からパラレルインし、V10
                                ; 8に格納します。

```

```

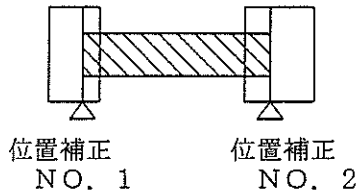
52:    TBIT      4, V108, NZ, B3      ; V108の4ビットが' 1' ならば' B3'
                                           ヘジャンプ
53:    TMRD      V109                  ; タイマーの値をV109に格納します。
54:    IF        V109, 100, GT, ER    ; V109>100 → ' ER' ヘジャンプ
55:    GOTO      B4                    ; B4ヘジャンプ
56: B3  ADD      V106, 1, V106        ; V106+1 → V106
57:    IF        V106, 105, GT, ST    ; V106>105 → ' ST' ヘジャンプ
58:    GOTO      B5                    ; ' B5' ヘジャンプ
59: ER  STROB    OF                    ; 同期信号を' OFF' します。
60:    DISP      20, 18, " タイムアウト" ; 20列18行を先頭に" タイムアウト" と
                                           表示
61:    DISP      " オーバーエラー"      ; " オーバーエラー" と表示
62:    GOTO      ST                    ; ' ST' ヘジャンプ

```

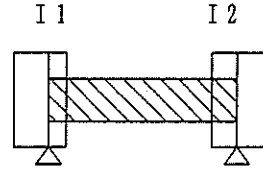
[プログラム例. 10]

検査・測定の分解能の自動計算

マスタ



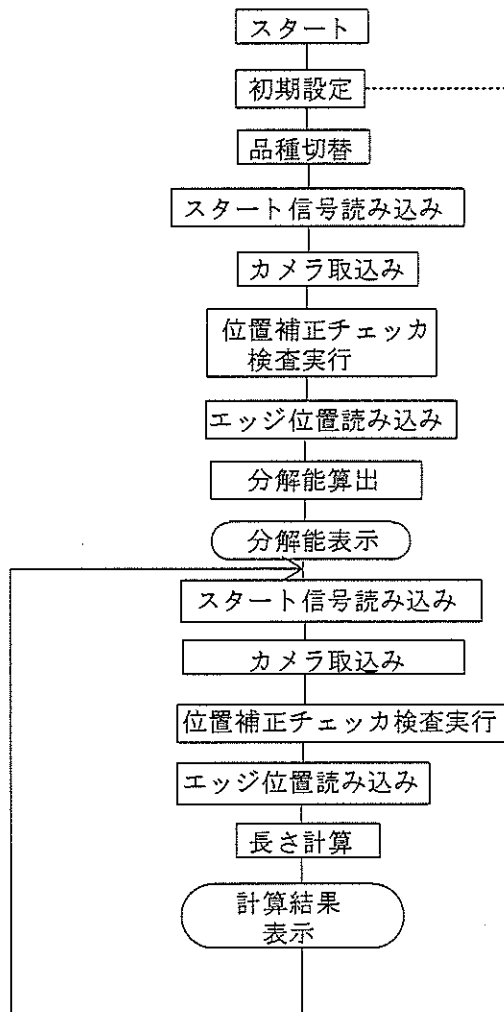
検査



マスタの全長 = 8 mm
ドット数 = I 21 - I 11

ドット数 I 21 - I 11
長さ = (I 21 - I 11) × (分解能)

$$\text{分解能} = \frac{8}{I 21 - I 11} \quad (\text{mm/ドット})$$



- 画像指定 (SHOW)
- 画面メモリ (MONT)
- 2値化レベル設定 (THRES)
- レジスタクリア (CLRREG)

```

1: CLRCRT
2: SHOW " *BIMCP"
3: MONT " A"
4: THRES A, 32
5: CLRREG V1, V10
6: START
7: READ " A***"
8: SCAN
9: MOVE CI11, V1 ; 位置補正NO. 1の水平エッジ検出値→V1
10: MOVE CI21, V2 ; " NO. 2 " →V2
11: SUB V2, V1, V3 ; V2-V1→V3
12: MOVE 8000, V11 ; 8000→V11
13: MUL V11, 10, V11 ; V11×10→V11
14: DIV V11, V3, V4 ; V11÷V3→V4
15: MOVE V4, A1, 3 ; V4の値3個→A1~A3
16: DISP 20, 10; " 分解能=" ; 20列10行を先頭に" 分解能=" と表示
17: DISP " 0. 00" ; " 0. 00" と表示
18: DISP A1, 3 ; A1~A3を表示
19: DISP " (mm/dot) " ; " (mm/dot) " と表示
20: ST START
21: READ " A***"
22: SCAN
23: MOVE CI11, V5
24: MOVE CI21, V6
25: SUB V6, V5, V7 ; V6-V5→V7
26: MUL V7, V4, V8 ; V7×V4→V8
27: MOVE V8, A11, 5 ; V8の値5個 →A11~A15
28: DISP 20, 12, " 長さ="
29: DISP A11, 1
30: DISP " ."
31: DISP A12, 4
32: DISP " (mm) "
33: GOTO ST

```

[プログラム例. 11]
モニタ画面へのコメント表示

1. 判定結果の表示例

DISP 20、10、”合格数=”

合格数=

DISP 20、10、”不合格数=”

不合格数=

2. 演算結果の表示

DISP 20、10、”合格率=”

合格率=

DISP 20、10、”ピッチ”

ピッチ

3. メッセージ

DISP 20、10、”マスターをセッ”
DISP ”トしてください”

マスターをセットしてください

DISP 20、10、”検査を終了しま”
DISP ”した”

検査を終了しました。

プログラム例9で、判定結果の画面表示の後に以下の様なプログラムを追加します。

```
PRINT 1, 0, " ", 1
PRINT 0, 2, " 合格数=", 8
PRINT 1, 0, A1, 2
PRINT 0, 2, " 不合格数=", 10
PRINT 1, 0, A3, 2
PRINT 0, 2, " 合格率=", 10
PRINT 0, 0, A5, 3
PRINT 0, 0, ". ", 1
PRINT 0, 0, A8, 1
PRINT 1, 0, "% ", 2
```

この時のプリントアウトは以下の様になります。

```
合格数 = 5
不合格数 = 5
合格率 = 50.0%
```

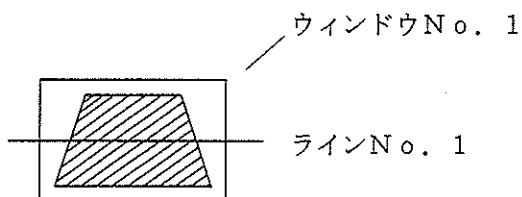
```
合格数 = 7
不合格数 = 3
合格率 = 70.0%
```

[プログラム例. 12]
シリアル通信例

通信プロトコルは、イメージチェッカ30、30Rを参考にしていますので、
詳細は「イメージチェッカ30・30Rマニュアル、8-2 RS232Cに
よる通信」を参照してください。

(検査内容)

画像取込み後、ラインNo. 1、ウィンドウNo. 1の検査を行ない、それぞれの
カウント値の数値結果・判定結果をシリアルで出力します。



(プロトコルの簡単な説明)
(ホスト側) →

(イメージチェッカ側)

- %SCR : 検査スタート
- %DL11CR : ラインNo. 1 カウント値の判定結果の要求
- %DW1CR : ウィンドウNo. 1 カウント値の判定結果の要求
- %VL11CR : ラインNo. 1 カウント値の数値結果の要求
- %VW1CR : ウィンドウNo. 1 カウント値の数値結果の要求
- %RCR : 再検査
- %XM1CR : 品種M1 に品種切替え

(イメージチェッカ側) → (ホスト側)

- %RCR : 画像取込み完了
- %ECR : 検査完了
- %D...CR : 判定結果データの応答
- %V...CR : 数値結果データの応答
- %YCR : 品種切替え完了
- %ZCR : 未登録コードエラー
- %UCR : データコードエラー

(注) 'S-I-N' コマンド実行時パリティチェック (垂直パリティチェック) は実行されません。
 パリティエラーが発生すると、別のコードになるため、入力されるべき文字であるかをチェックすることによりパリティチェックすることと同様の結果が得られます。

(レジスタ使用内容)

< vレジスタ >

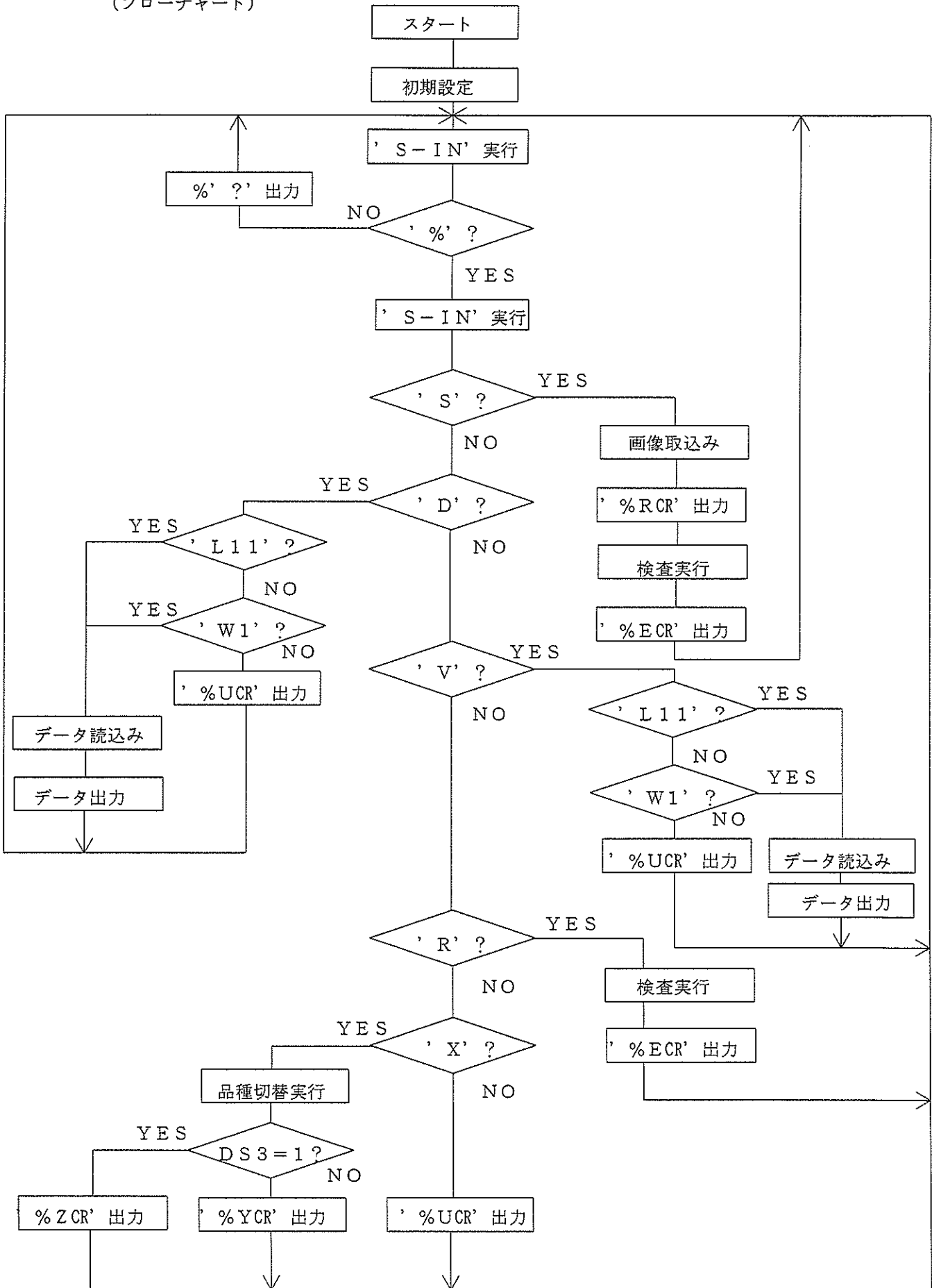
V 1 : データ読み込み
 V 3 : 桁数読み込み
 V 5 : 出力個数

< aレジスタ >

A 1 }
 | } : シリアル入力データ読み込み
 A 5 }

A 1 0 1	:	%	A 1 3 1	:	%
A 1 0 2	:	S	A 1 3 2	:	Y
A 1 0 3	:	D	A 1 3 3	:	CR
A 1 0 4	:	V	A 1 3 4	:	%
A 1 0 5	:	R	A 1 3 5	:	Z
A 1 0 6	:	L	A 1 3 6	:	CR
A 1 0 7	:	1			
A 1 0 8	:	1			
A 1 0 9	:	W	A 5 1	:	%
A 1 1 0	:	1	A 5 2	:	D
A 1 1 1	:	%	A 5 3	:	判定結果データ
A 1 1 2	:	?	A 5 4	:	CR
A 1 1 3	:	CR			
A 1 1 4	:	%			
A 1 1 5	:	R	A 6 1	:	%
A 1 1 6	:	CR	A 6 2	:	V
A 1 1 7	:	%	A 6 3	:] 数値結果データ
A 1 1 8	:	E			
A 1 1 9	:	CR	A [V 4]	:	CR
A 1 2 0	:	%			
A 1 2 1	:	U			
A 1 2 2	:	CR			

(フローチャート)



Matsushita Image Checker Language: (PM1) [MAIN]

```

1: CLRCRT
2: CLRREG V1, V200: V1~V200をクリア (=0) します。
3: CLRREG A1, A200: A1~A200をクリア (=Null) します。
4: CALL PM7 : PM7へジャンプします。
5: ST
6: S-IN A1, 1 : シリアル入力し, A1に格納します。
7: IFSTR A1, A101, 1, IN: A1=A101('%')ならば'IN'へジャンプします。
8: S-OUT A111, 3 : A111から3個の文字をシリアル出力します。
9: GOTO ST : 'ST'へジャンプします。
10: IN
11: S-IN A1, 5, V101 : 5文字になるまで, 又は'CR'が入力されるまで
シリアル入力し, A1~A5に格納し, 入力文字数を
V101に格納します。
12: IFSTR A1, A102, 1, RE: A1=A102('S')ならば'RE'へジャンプします。
13: IFSTR A1, A103, 1, DO: A1=A103('D')ならば'DO'へジャンプします。
14: IFSTR A1, A104, 1, VO: A1=A104('V')ならば'VO'へジャンプします。
15: IFSTR A1, A105, 1, RR: A1=A105('R')ならば'RR'へジャンプします。
16: IFSTR A1, A124, 1, TY: A1=A124('X')ならば'TY'へジャンプします。
17: S-OUT A120, 3 : A120~A122(%UCR)をシリアル出力します。
18: GOTO ST
19: RE
20: CALL PM2
21: GOTO ST
22: DO
23: CALL PM3
24: GOTO ST
25: VO
26: CALL PM4
27: GOTO ST
28: RR
29: CALL PM5
30: GOTO ST
31: TY
32: CALL PM6
33: GOTO ST

```

Matsushita Image Checker Language: (PM2) [TORIKOMI-KENSA]

```

1: READ "A***" : Aメモリに画像取込みをします。
2: S-OUT A114, 3 : A114~A116 (%RCR)をシリアル出力します。
3: SCAN : 全チェックの検査を実行します。
4: S-OUT A117, 3 : A117~A119 (%ECR)をシリアル出力します。
5: RET

```

Matsushita Image Checker Language: (PM3) [D-OUT]

```

1: IFSTR A2, A106, 3, LO: A2~A4=A106~A108(L11)ならば'LO'へジャンプ
します。
2: IFSTR A2, A109, 2, WO: A2~A3=A109~A110(W1)ならば'WO'へジャンプ
します。
3: S-OUT A120, 3 : A120~A122(%UCR)をシリアル出力します。
4: RET
5: LO
6: MOVE DL11, A53, 1 : ういNo.1カウント値の判定結果をA53に格納します。
7: GOTO SO
8: WO
9: MOVE DW1, A53, 1 : ういNo.1カウント値の判定結果をA53に格納します。
10: SO
11: S-OUT A51, 4 : A51~A54をシリアル出力します。
12: RET

```

```

Matsushita Image Checker Language: (PM4) [V-OUT ]
1: IFSTR A2, A106, 3, LO: A2~A4=A106~A108(L11)ならば'LO'へジャンプ
    します。
2: IFSTR A2, A109, 2, WO: A2~A3=A109~A110(W1)ならば'WO'へジャンプ
    します。
3: S-OUT A120, 3
4: RET
5: LO
6: MOVE CL11, V1 : 11番No.1のかけ値の数値結果をV1に格納します。
7: GOTO SO
8: WO
9: MOVE CW1, V1 : 11番No.1のかけ値の数値結果をV1に格納します。
10: SO
11: LEN V1, V2, V3 : V1の符号・桁数をV2、V3に格納します。
12: MOVE V1, A63, V3
13: ADD 63, V3, V4 : 63+V3→V4
14: CODE 0XD, A[V4] : 0XD(=CR) →V4の内容が示すNo.のaレジスタ
15: ADD V3, 3, V5
16: S-OUT A61, V5
17: RET

```

```

Matsushita Image Checker Language: (PM5) [SAI-KENSA]
1: SCAN : 全チェックの検査を実行します。
2: S-OUT A117, 3 : A117~A119(%ECR)をシリアル出力します。
3: RET

```

```

Matsushita Image Checker Language: (PM6) [TYPE ]
1: MOVE A3, V11, 1
2: TYPE A2, V11 : A2の内容の11でV11の内容のNo.に品種切替えます。
3: IF DS3, 1, EQ, ER: DS3('OVFLG')=1ならば'ER'へジャンプします。
4: S-OUT A131, 3
5: RET
6: ER
7: S-OUT A134, 3
8: OVOFF : 'OVFLG'信号を'OFF'にします。
9: RET

```

```

Matsushita Image Checker Language: (PM7) [MOVE ]
 1: MOVE      "%D", A51, 2
 2: CODE      0XD, A54
 3:
 4: MOVE      "%V", A61, 2
 5:
 6:
 7: MOVE      "%SDVR", A101, 5
 8: MOVE      "L11W1", A106, 5
 9:
10: MOVE      "%?" , A111, 2
11: CODE      0XD, A113
12:
13: MOVE      "%R", A114, 2
14: CODE      0XD, A116
15:
16: MOVE      "%E", A117, 2
17: CODE      0XD, A119
18:
19: MOVE      "%U", A120, 2
20: CODE      0XD, A122
21:
22: MOVE      "X", A124, 1
23:
24: MOVE      "%Y", A131, 2
25: CODE      0XD, A133
26:
27: MOVE      "%Z", A134, 2
28: CODE      0XD, A136
29:
30: MOVE      "%D", A1, 2
31: MOVE      "%V", A11, 2

```



```

1:      CLRCRT                ;画面をクリアします。
2:      CLRREG  V1, V100      ;V1~V100をクリア (=0) に
                               ;します。
3:      CLRREG  A1, A100      ;A1~A100をクリア (=0X00) に
                               ;します。
4: ST
   |
   |
   |
   |
52:     IF          DS3, 1, EQ, ER ; S3=1 ("OVFLG" =1)ならば"ER"へ
                               ;ジャンプ
53:     IF          V100, 1, EQ, OK ; V100=1ならば,"OK"へジャンプ
54:     GOTO        ST          ; "ST"へジャンプします。
55: ER                                     ; ラベル名"ER"
                               ; ("エラー"を表示します。)
56:     IF          V100, 1, EQ, ST ; V100=1ならば,"ST"へジャンプ
57:     DISP        54, 24, "エラー" ; 24行,54列に"エラー"と表示します。
58:     OVOFF                                     ; "OVFLG"信号をOFFします。
59:     MOVE        1, V100          ; 1→V100
60:     GOTO        ST          ; "ST"へジャンプします。
61: OK                                     ; ラベル名"OK"
                               ; ("エラー"表示を消去します。)
62:     DISP        54, 24, " ", 6 ; 24行,54列を先頭に" " (空白文字)を
                               ; 6文字分表示します。
63:     MOVE        0, V100         ; 0→V100
64:     GOTO        ST          ; "ST"へジャンプします。

```


付 録

[1] 漢字コード一覧表 (JIS第1水準)

		0 1 2 3	4 5 6 7	8 9 A B	C D E F
記号	2120	注1 ㊦ , .	, . . :	; ? ! .	° - ` ~ ..
	2130	^ _ _ ,	ゞ 〉 ゞ "	全 々 へ ○	- - - /
	2140	\ ~ ' '	" " ()	[] []
	2150	{ } < >	《 》 「 」	『 』 【 】	+ - ± ×
	2160	÷ = ≠ <	> ≤ ≥ ∞	∴ † ‡ °	' " ° ‰
	2170	\$ € £ %	# & * @	§ ☆ ★ ○	● ◎ ◇
	2220	◆ □ ■	△ ▲ ▽ ▼	※ 〒 → ←	↑ ↓ =
英数字	2330	0 1 2 3	4 5 6 7	8 9	
	2340	A B C	D E F G	H I J K	L M N O
	2350	P Q R S	T U V W	X Y Z	
	2360	a b c	d e f g	h i j k	l m n o
	2370	p q r s	t u v w	x y z	
ひらがな	2420	あ あ い	い う え	え お お か	が き ぎ く
	2430	ぐ け げ こ	ご さ ざ し	じ す ず せ	ぜ そ ぞ た
	2440	だ ち ぢ っ	つ づ て で	と ど な に	ぬ ね の は
	2450	ば ば ひ び	び ふ ぶ ぶ	へ べ ぺ ほ	ぼ ぼ ま み
	2460	む め も や	や ゆ ゆ よ	よ ら り る	れ ろ わ わ
	2470	ゐ ゑ を ん			
カタカナ	2520	ア ア イ	イ ウ エ	エ オ オ カ	ガ キ ギ ク
	2530	グ ケ ゲ コ	ゴ サ ザ シ	ジ ス ズ セ	ゼ ソ ゾ タ
	2540	ダ チ デ ッ	ツ ヅ テ デ	ト ド ナ ニ	ヌ ネ ノ ハ
	2550	バ パ ヒ ビ	ピ フ ブ プ	ヘ ベ ペ ホ	ボ ポ マ ミ
	2560	ム メ モ ヤ	ヤ ユ ユ ヨ	ヨ ラ リ ル	レ ロ ヲ ワ
	2570	キ エ ラ ン	ヴ カ ケ		
ギリシア文字	2620	A B Γ	Δ E Z H	Θ I K Λ	M N Ξ O
	2630	Π P Σ T	Υ Φ X Ψ	Ω	
	2640	α β γ	δ ε ζ η	θ ι κ λ	μ ν ξ ο
	2650	π ρ σ τ	υ φ χ φ	ω	
		0 1 2 3	4 5 6 7	8 9 A B	C D E F

表の見方

コードは16進で表現されています。例えば" B "のコードは2340+2=2342となります。

注意 1

2120は日本語コードとして定義されていません。
2121の㊦は空白(スペース)コードを示します。

		0 1 2 3	4 5 6 7	8 9 A B	C D E F
ロ シ ア 文 字	2720	А Б В	Г Д Е Ё	Ж З И Й	К Л М Н
	2730	О П Р С	Т У Ф Х	Ц Ч Ш Щ	Ь Ы Ь Э
	2740	Ю Я			
	2750	а б в	г д е ё	ж з и й	к л м н
	2760	о п р с	т у ф х	ц ч ш щ	ь ы ь э
	2770	ю я			
ア	3020	厶 咿 娃	阿 哀 愛 挨	始 逢 葵 茜	穉 悉 握 渥
	3030	旭 葦 芦 鯨	梓 庄 幹 扱	宛 姐 虻 飴	絢 綾 鮎 或
	3040	粟 裕 安 庵	按 暗 案 闇	鞍 杏	
イ	3040			以 伊	位 依 偉 匪
	3050	夷 委 威 尉	惟 意 慰 易	椅 為 畏 異	移 維 緯 胃
	3060	萎 衣 謂 違	遺 医 井 亥	域 育 郁 磯	一 壹 溢 逸
	3070	稻 茨 芋 鱒	允 印 咽 員	因 姻 引 飲	淫 胤 陰
	3120	院 陰 隱	韻 吋		
ウ	3120		右 宇	烏 羽 迂 雨	卯 鵝 窺 丑
	3130	確 白 渦 嘘	唄 蔚 蔚 鰻	姥 厥 浦 瓜	閩 噂 云 運
	3140	雲			
エ	3140	荏 餌 叡	當 嬰 影 映	曳 榮 永 泳	洩 瑛 盈 顛
	3150	穎 英 衛 詠	銳 液 疫 益	馱 悅 謁 越	閱 榎 厭 円
	3160	園 堰 奄 宴	延 怨 掩 援	浴 演 炎 焰	煙 燕 猿 縁
	3170	艶 苑 菌 遠	鉛 鴛 塩		
オ	3170		於	汚 甥 凹 央	奧 往 応
	3220	押 旺 横	欧 殴 王 翁	襖 鶯 鷗 黃	岡 沖 萩 億
	3230	屋 憶 臆 桶	牡 乙 俺 卸	恩 温 穩 音	
カ	3230				下 化 仮 何
	3240	伽 伽 佳 加	可 嘉 夏 嫁	家 寡 科 暇	果 架 歌 河
	3250	火 珂 禍 禾	稼 箇 花 苛	茄 荷 華 菓	蝦 課 嘩 貨
	3260	迦 過 霞 蚊	俄 峨 我 牙	画 臥 芽 蛾	賀 雅 餓 駕
	3270	介 会 解 回	塊 壞 廻 快	怪 悔 恢 懷	戒 拐 改
	3320	魁 誨 械	海 灰 界 皆	繪 芥 蟹 開	階 貝 凱 劾
	3330	外 咳 害 崖	慨 概 涯 碍	蓋 街 該 鎧	骸 湮 馨 蛙
	3340	垣 柿 蠣 鈎	割 嚇 各 廓	拈 攪 格 核	殼 獲 確 穫
		0 1 2 3	4 5 6 7	8 9 A B	C D E F

		0 1 2 3	4 5 6 7	8 9 A B	C D E F
力	3 3 5 0	覺 角 赫 較	郭 闊 隔 革	学 岳 樂 額	顎 掛 笠 檉
	3 3 6 0	樞 梶 歛 渴	割 喝 恰 括	活 渴 滑 葛	褐 轄 且 鏗
	3 3 7 0	叶 柁 樺 鞫	株 兜 竈 蒲	釜 鎌 嚙 鴨	栢 茅 萱
	3 4 2 0	粥 刈 苻	瓦 乾 侃 冠	寒 刊 勘 勸	卷 喚 堪 姦
	3 4 3 0	完 官 寬 干	幹 患 感 慣	憾 換 敢 柑	桓 棺 款 歛
	3 4 4 0	汗 漢 澗 灌	環 甘 監 看	竿 管 簡 緩	缶 翰 肝 艦
	3 4 5 0	莞 觀 諫 貫	還 鑑 間 閑	闕 陷 韓 館	館 丸 含 岸
	3 4 6 0	巖 玩 癌 眼	岩 翫 贗 雁	頑 顏 願	
キ	3 4 6 0			企	伎 危 喜 器
	3 4 7 0	基 奇 嬉 寄	岐 希 幾 忌	揮 机 旗 既	期 棋 棄
	3 5 2 0	機 埽 毅	氣 汽 畿 祈	季 稀 紀 微	規 記 貴 起
	3 5 3 0	軌 輝 飢 騎	鬼 龜 偽 儀	妓 宜 戲 技	擬 欺 犧 疑
	3 5 4 0	祇 義 蟻 誼	議 掬 菊 鞠	吉 吃 喫 桔	橘 詰 砧 杵
	3 5 5 0	黍 却 客 脚	虐 逆 丘 久	仇 休 及 吸	宮 弓 急 救
	3 5 6 0	朽 求 汲 泣	灸 球 究 窮	笈 級 糾 給	旧 牛 去 居
	3 5 7 0	巨 拒 扞 拳	渠 虛 許 距	鋸 漁 禦 魚	亨 享 京
	3 6 2 0	供 夾 僑	兇 競 共 凶	協 匡 卿 叫	喬 境 峽 強
	3 6 3 0	疆 怯 恐 恭	挾 教 橋 況	狂 狹 矯 胸	脅 興 蕎 鄉
	3 6 4 0	鏡 響 饗 驚	仰 凝 堯 曉	業 局 曲 極	玉 桐 籽 僅
	3 6 5 0	勤 均 巾 錦	斤 欣 欽 琴	禁 禽 筋 緊	芹 菌 衿 襟
	3 6 6 0	謹 近 金 吟	銀		
ク	3 6 6 0		九 俱 句	区 狗 玖 矩	苦 軀 驅 駟
	3 6 7 0	駒 具 愚 虞	喰 空 偶 寓	遇 隅 串 櫛	鉤 屑 屈
	3 7 2 0	掘 窟 沓	靴 轡 窪 熊	隈 叅 栗 繰	桑 鍬 勳 君
	3 7 3 0	薰 訓 群 軍	郡		
ケ	3 7 3 0		卦 袞 祁	係 傾 刑 兄	啓 圭 珪 型
	3 7 4 0	契 形 徑 恵	慶 慧 憩 揭	携 敬 景 桂	溪 畦 稽 系
	3 7 5 0	経 繼 繫 野	莖 荊 蚩 計	詣 警 輕 頸	鷄 芸 迎 鯨
	3 7 6 0	劇 戟 擊 激	隙 桁 傑 欠	決 潔 穴 結	血 訣 月 件
	3 7 7 0	儉 倦 健 兼	券 劍 喧 圈	堅 嫌 建 憲	懸 拳 捲
	3 8 2 0	檢 權 牽	犬 猷 研 硯	絹 梟 肩 見	謙 賢 軒 遣
	3 8 3 0	鍵 險 頭 驗	齧 元 原 蔽	幻 弦 滅 源	玄 現 絃 舷
		0 1 2 3	4 5 6 7	8 9 A B	C D E F

		0 1 2 3	4 5 6 7	8 9 A B	C D E F	
ケ	3840	言 諺 限				
二	3840		乎	個 古 呼 固	姑 孤 己 庫	弧 戸 故 枯
	3850	湖 狐 糊 袴	股 胡 菰 虎	誇 跨 鈷 雇	顧 鼓 五 互	
	3860	伍 午 吳 吾	娛 後 御 悟	梧 檣 瑚 碁	語 誤 護 醐	
	3870	乞 鯉 交 佼	侯 候 倅 光	公 功 効 勾	厚 口 向	
	3920		后 喉 坑	垢 好 孔 孝	宏 工 巧 巷	幸 広 庚 康
	3930	弘 恒 慌 抗	拘 控 攻 昂	晃 更 杭 校	梗 構 江 洪	
	3940	浩 港 溝 甲	皇 硬 稿 糠	紅 紘 絞 綱	耕 考 肯 肱	
	3950	腔 膏 航 荒	行 衡 講 貢	購 郊 醇 鉞	砧 鋼 閤 降	
	3960	項 香 高 鴻	剛 劫 号 合	壕 拷 濠 豪	轟 麴 克 刻	
	3970	告 国 穀 酷	鵠 黒 獄 漉	腰 甌 忽 惚	骨 狛 込	
3A20		此 頃 今	困 坤 墾 婚	恨 懇 昏 昆	根 梱 混 痕	
3A30	紺 良 魂					
サ	3A30		些	佐 又 唆 嗟	左 差 查 沙	蹉 砂 詐 鎖
	3A40	裳 坐 座 挫	債 催 再 最	哉 塞 妻 宰	彩 才 採 栽	
	3A50	歲 濟 災 采	屨 碎 砦 祭	斉 細 菜 裁	載 際 劑 在	
	3A60	材 罪 財 冴	坂 阪 堺 柵	肴 咲 崎 埼	碕 鷺 作 削	
	3A70	咋 搾 昨 朔	柵 窄 策 索	錯 桜 鮭 笹	匙 冊 刷	
	3B20		察 撈 撮	擦 札 殺 薩	雜 臯 鯖 捌	鯖 鮫 皿 晒
	3B30	三 傘 參 山	慘 撒 散 棧	燦 珊 産 算	纂 蚕 讚 贊	
3B40	酸 餐 斬 暫	殘				
シ	3B40		仕 仔 伺	使 刺 司 史	嗣 四 士 始	
	3B50	姉 姿 子 屍	市 師 志 思	指 支 攷 斯	施 旨 枝 止	
	3B60	死 氏 獅 祉	私 糸 紙 紫	肢 脂 至 視	詞 詩 試 誌	
	3B70	諮 資 賜 雌	飼 齒 事 似	侍 児 字 寺	慈 持 時	
	3C20		次 滋 治	爾 璽 痔 磁	示 而 耳 自	蒔 辭 汐 鹿
	3C30	式 識 鳴 竺	軸 穴 零 七	叱 執 失 嫉	室 悉 湿 漆	
	3C40	疾 質 実 蔀	篠 惲 柴 芝	屢 蕊 縞 舎	写 射 捨 赦	
	3C50	斜 煮 社 紗	者 謝 車 遮	蛇 邪 借 勺	尺 杓 灼 爵	
	3C60	酌 积 錫 若	寂 弱 惹 主	取 守 手 朱	殊 狩 珠 種	
	3C70	腫 趣 酒 首	儒 受 呪 寿	授 樹 綬 需	囚 収 周	
3D20		宗 就 州	修 愁 拾 洲	秀 秋 終 繡	習 臭 舟 蒐	
		0 1 2 3	4 5 6 7	8 9 A B	C D E F	

		0 1 2 3	4 5 6 7	8 9 A B	C D E F
シ	3D30	衆 襲 讐 蹴	輯 週 酋 酬	集 醜 什 住	充 十 從 戎
	3D40	柔 汁 洩 獸	縱 重 銃 叔	夙 宿 淑 祝	縮 肅 塾 熟
	3D50	出 術 述 俊	峻 春 瞬 竣	舜 駿 准 盾	旬 楯 殉 淳
	3D60	準 潤 盾 純	巡 遵 醇 順	処 初 所 暑	曙 渚 庶 緒
	3D70	署 書 薯 蓆	諸 助 叙 女	序 徐 恕 鋤	除 傷 償
	3E20	勝 匠 升	召 哨 商 唱	嘗 獎 妾 娼	宵 將 小 少
	3E30	尚 庄 床 廠	彰 承 抄 招	掌 捷 昇 昌	昭 晶 松 梢
	3E40	樟 樵 沼 消	涉 湘 燒 焦	照 症 省 硝	礁 祥 称 章
	3E50	笑 粧 紹 肖	蒿 蔣 蕉 衝	裳 訟 証 詔	詳 象 賞 醬
	3E60	鉦 鍾 鐘 障	鞘 上 丈 丞	乘 冗 剩 城	場 壞 嬾 常
	3E70	情 擾 条 杖	淨 狀 疊 穰	蒸 讓 釀 錠	囑 壇 飾
	3F20	拭 植 殖	燭 織 職 色	触 食 蝕 辱	尻 伸 信 侵
	3F30	唇 娠 寢 審	心 慎 振 新	晋 森 榛 浸	深 申 疹 真
	3F40	神 秦 紳 臣	芯 薪 親 診	身 辛 進 針	震 人 仁 刃
	3F50	塵 壬 尋 甚	尽 腎 訊 迅	陣 韌	
ス	3F50			筍 諏	須 醉 囟 厨
	3F60	逗 吹 垂 帥	推 水 炊 睡	粹 翠 衰 遂	醉 錐 錘 隨
	3F70	瑞 髓 崇 高	数 枢 趨 雛	据 杉 楯 菅	頗 雀 裾
	4020	澄 摺 寸			
セ	4020		世 瀬 畝 是	凄 制 勢 姓	征 性 成 政
	4030	整 星 晴 棲	栖 正 清 牲	生 盛 精 聖	声 製 西 誠
	4040	誓 請 逝 醒	青 静 齐 税	脆 隻 席 惜	戚 斥 昔 析
	4050	石 積 籍 績	脊 責 赤 跡	躓 碩 切 拙	接 撰 折 設
	4060	窃 節 説 雪	絶 舌 蟬 仙	先 千 占 宣	專 尖 川 戰
	4070	扇 撰 栓 梅	泉 淺 洗 染	潜 煎 煽 旋	穿 箭 線
	4120	織 羨 腺	舛 船 薦 詮	賤 踐 選 遷	錢 銃 閃 鮮
	4130	前 善 漸 然	全 禅 繕 膳	糲	
ソ	4130			噌 塑 岨	措 曾 曾 楚
	4140	狙 疏 疎 礎	祖 租 粗 素	組 蘇 訴 阻	遡 鼠 僧 創
	4150	双 叢 倉 喪	壯 奏 爽 宋	層 匝 惣 想	搜 掃 挿 搔
	4160	操 早 曹 巢	槍 槽 漕 燥	争 瘦 相 窓	糟 総 綜 聡
	4170	草 莊 葬 蒼	藻 装 走 送	遭 鎗 霜 騷	像 増 憎
		0 1 2 3	4 5 6 7	8 9 A B	C D E F

		0 1 2 3	4 5 6 7	8 9 A B	C D E F
ン	4 2 2 0 4 2 3 0	臟 藏 贈 屬 賊 族 統	造 促 側 則 卒 袖 其 揃	即 息 捉 束 存 孫 尊 損	測 足 速 俗 村 遜
夕	4 2 3 0 4 2 4 0 4 2 5 0 4 2 6 0 4 2 7 0 4 3 2 0 4 3 3 0 4 3 4 0	太 汰 訖 唾 対 耐 岱 帶 退 逮 隊 黛 宅 托 扱 拓 叩 但 達 丹 單 嘆 坦 胆 蛋 誕 鍛	隨 妥 惰 打 待 怠 態 戴 鯛 代 台 大 沢 濯 琢 託 辰 奪 脱 巽 担 探 且 歎 団 壇 彈 断	柁 舵 楫 陀 替 泰 滯 胎 第 醜 題 鷹 鐸 濁 諾 茸 豎 迪 棚 谷 淡 湛 炭 短 暖 檀 段 男	他 多 馱 驢 体 堆 腿 苔 袋 貸 滝 瀧 卓 啄 胤 蛸 只 狸 鱒 樽 誰 端 箒 綻 耽 談
チ	4 3 4 0 4 3 5 0 4 3 6 0 4 3 7 0 4 4 2 0 4 4 3 0 4 4 4 0	弛 恥 智 池 逐 秩 窒 茶 註 耐 鑄 駐 帖 帳 庁 聰 脹 腸 蝶 沈 珍 賃 鎮	痴 稚 置 致 嫡 着 中 仲 禱 瀦 猪 苧 弔 張 彫 微 調 諜 超 跳 陳	蜘蛛 遲 馳 築 宙 忠 抽 昼 著 貯 丁 兆 懲 挑 暢 朝 銚 長 頂 鳥	值 知 地 畜 竹 筑 蓄 柱 注 虫 衷 凋 喋 寵 潮 牒 町 眺 勅 抄 直 朕
ツ	4 4 4 0 4 4 5 0 4 4 6 0	規 佃 漬 柘 釣 鶴	津 墜 椎 辻 蔦 綴 鏝	槌 追 鎚 痛 椿 漬 坪 壺	通 塚 柁 摺 孀 紬 爪 吊
テ	4 4 6 0 4 4 7 0 4 5 2 0 4 5 3 0 4 5 4 0	亭 低 梯 抵 挺 提 邸 鄭 釘 徹 撤 輟 迭 点 伝 殿 澱	停 偵 剃 貞 梯 汀 碇 禎 鼎 泥 摘 擻 鉄 典 填 天 田 電	呈 堤 定 帝 程 締 艇 訂 敵 滴 的 笛 展 店 添 纏	底 庭 廷 弟 諦 蹄 遞 適 鑄 溺 哲 甜 貼 転 顛
ト	4 5 4 0 4 5 5 0 4 5 6 0 4 5 7 0 4 6 2 0 4 6 3 0 4 6 4 0	登 菟 賭 途 凍 刀 唐 塔 盜 淘 湯 涛 董 蕩 藤 動 同 堂 導 得 德 浣 特	兔 吐 都 鍍 砥 砺 塘 套 宕 島 灯 燈 当 痘 討 騰 豆 踏 懂 撞 洞 瞳 督 秃 篤 毒	堵 塗 妬 屠 努 度 土 奴 嶋 悼 投 搭 禱 等 答 筒 逃 透 鑑 陶 童 胴 苟 道 独 読 栲 椽	徒 斗 杜 渡 怒 倒 党 冬 東 桃 袴 棟 糖 統 到 頭 騰 闕 働 銅 峠 鴉 匿 凸 突 椽 届
		0 1 2 3	4 5 6 7	8 9 A B	C D E F

		0 1 2 3	4 5 6 7	8 9 A B	C D E F
ト	4650	鳶 苦 寅 酉	滯 噸 屯 惇	敦 沌 豚 遁	頓 吞 曇 鈍
ナ	4660	奈 那 内 乍	風 薙 謎 灘	捺 鍋 樞 馴	繩 暇 南 楠
	4670	軟 難 汝			
ニ	4670		尼 忒 迓 匂	賑 肉 虹 廿	日 乳 入
	4720	如 尿 菲	任 妊 忍 認		
ヌ	4720			濡	
ネ	4720			襦 祢 寧	葱 猫 熱 年
	4730	念 捻 撚 燃	粘		
ノ	4730		乃 迺 之	埜 囊 惱 濃	納 能 腦 膿
	4740	農 覗 蚤			
ハ	4740		芭 播 霸 杷	波 派 琶 破	婆 罵 芭 馬
	4750	俳 糜 捩 排	敗 杯 盃 牌	背 肺 輩 配	倍 培 媒 梅
	4760	煤 煤 狠 買	壳 賠 陪 遄	蠅 秤 矧 菽	伯 剥 博 拍
	4770	柏 泊 白 箔	粕 舶 薄 迫	曝 漠 爆 縛	莫 駁 麥
	4820	函 箱 砧	箸 肇 筍 櫨	幡 肌 畑 阜	八 鉢 滌 癸
	4830	醜 髮 伐 罰	拔 筏 闕 鳩	嘶 塙 蛤 隼	伴 判 半 反
	4840	叛 帆 搬 斑	板 汜 汎 版	犯 班 畔 繁	般 藩 販 範
	4850	采 煩 頌 飯	挽 晚 番 盤	磐 蕃 蠻	
ヒ	4850			匪	卑 否 妃 庇
	4860	彼 悲 扉 批	披 斐 比 泌	疲 皮 碑 秘	緋 罷 肥 被
	4870	誹 費 避 非	飛 樋 簸 備	尾 微 枇 毘	琵琶 眉 美
	4920	鼻 柎 稗	匹 疋 髭 彥	膝 菱 肘 弼	必 畢 筆 逼
	4930	桧 姬 媛 紐	百 謬 倭 彪	標 冰 漂 瓢	票 表 評 豹
	4940	廟 描 病 秒	苗 錨 鋌 蒜	蛭 鱒 品 彬	斌 浜 瀕 貧
	4950	賓 頻 敏 瓶			
フ	4950		不 付 埠 夫	婦 富 富 布	府 怖 扶 敷
	4960	斧 普 浮 父	符 腐 膚 芙	譜 負 賦 赴	阜 附 侮 撫
	4970	武 舞 葡 蕪	部 封 楓 風	葦 蕩 伏 副	復 幅 服
	4A20	福 腹 複	覆 淵 弗 弘	沸 仏 物 鮒	分 吻 噴 墳
	4A30	憤 扮 焚 奮	粉 糞 紛 霏	文 聞	
フ	4A30			丙 併	兵 塤 幣 平
	4A40	弊 柄 並 蔽	閉 陞 米 頁	僻 壁 癖 碧	別 瞥 蔑 篋
		0 1 2 3	4 5 6 7	8 9 A B	C D E F

		0 1 2 3	4 5 6 7	8 9 A B	C D E F
へ	4A50	偏 変 片 篇	編 辺 返 遍	便 勉 婉 弁	鞭
ホ	4A50				保 舗 舗
	4A60	圃 捕 歩 甫	補 輔 穂 募	墓 慕 戊 暮	母 簿 菩 倣
	4A71	俸 包 呆 報	奉 宝 峰 峯	崩 庖 抱 捧	放 方 朋
	4B20	法 泡 烹	砲 縫 胞 芳	萌 蓬 蜂 褒	訪 豊 邦 鋒
	4B30	飽 鳳 鵬 乏	亡 傍 剖 坊	妨 帽 忘 忙	房 暴 望 某
	4B40	棒 冒 紡 肪	膨 謀 貌 貿	鉉 防 吠 類	北 僕 卜 墨
	4B50	撲 朴 牧 睦	穆 鉤 勃 沒	殆 堀 幌 奔	本 翻 凡 盆
マ	4B60	摩 磨 魔 麻	埋 妹 味 枚	每 哩 禩 幕	膜 枕 鯖 枉
	4B70	鱗 榭 亦 俣	又 抹 末 沫	迄 俛 繭 鷹	万 慢 滿
	4C20	漫 蔓			
ミ	4C20	味	未 魅 巳 箕	岬 密 蜜 湊	蓑 稔 脈 妙
	4C30	耗 民 眠			
ム	4C30	務	夢 無 牟 矛	霧 鷓 掠 婿	娘
メ	4C30				冥 名 命
	4C40	明 盟 迷 銘	鳴 姪 牝 滅	免 棉 綿 緬	面 麵
モ	4C40				摸 模
	4C50	茂 妄 孟 毛	猛 盲 網 耗	蒙 儲 木 默	目 杳 勿 餅
	4C60	尤 戾 糊 貫	問 悶 紋 門	匆	
ヤ	4C60			也 冶 夜	爺 耶 野 弥
	4C70	矢 厄 役 約	粟 訳 躍 靖	柳 藪 鍵	
ユ	4C70			愉	愈 油 癒
	4D20	論 輪 唯	佑 優 勇 友	宥 幽 悠 憂	揖 有 柚 湧
	4D30	涌 猶 猷 由	祐 裕 誘 遊	邑 郵 雄 融	夕
ヨ	4D30				予 余 与
	4D40	誉 輿 預 備	幼 妖 容 庸	揚 搖 擁 曜	楊 樣 洋 溶
	4D50	熔 用 窯 羊	耀 葉 蓉 要	謡 踊 遙 陽	養 慾 抑 欲
	4D60	沃 浴 翌 翼	淀		
ラ	4D60		羅 螺 裸	来 萊 賴 雷	洛 絡 落 酪
	4D70	乱 卵 嵐 欄	濫 藍 蘭 覽		
リ	4D70			利 吏 履 李	梨 理 璃
	4E20	痢 裏 裡	里 離 陸 律	率 立 葎 掠	略 劉 流 溜
		0 1 2 3	4 5 6 7	8 9 A B	C D E F

		0 1 2 3	4 5 6 7	8 9 A B	C D E F
リ	4E30	琉 留 硫 粒	隆 竜 龍 侶	慮 旅 虜 了	亮 僚 兩 凌
	4E40	寮 料 梁 涼	獵 療 瞭 稜	糧 良 諒 遼	量 陵 領 力
	4E50	綠 倫 厘 林	淋 燐 琳 臨	輪 隣 鱗 麟	
ル	4E50				瑠 罌 淚 累
	4E60	類			
レ	4E60	令 伶 例	冷 勵 嶺 伶	玲 礼 苓 鈴	隸 零 靈 麗
	4E70	齡 曆 歷 列	劣 烈 裂 廉	恋 憐 漣 煉	簾 練 聯
	4F20	蓮 連 鍊			
ロ	4F20		呂 魯 櫓 炉	賂 路 露 勞	婁 廊 弄 朗
	4F30	樓 榔 浪 漏	牢 狼 籠 老	聾 蠅 郎 六	麓 祿 肋 録
	4F40	論			
ワ	4F40	倭 和 話	歪 賄 脇 惑	粹 鷺 瓦 亘	鰐 詫 蕈 蕨
	4F50	椀 湾 碗 腕			
		0 1 2 3	4 5 6 7	8 9 A B	C D E F

[2] ASCIIコード一覧表

上位4ビット→

下位4ビット↓

	0	1	2	3	4	5	6	7
0	Null	DLE	SP	0		P		p
1	SOH	DC1	!	1	A	Q	a	q
2	STX	DC2	"	2	B	R	b	r
3	ETX	DC3	#	3	C	S	c	s
4	EOT	DC4	\$	4	D	T	d	t
5	ENQ	NAK	%	5	E	U	e	u
6	ACK	SYN	&	6	F	V	f	v
7	BEL	ETB	'	7	G	W	g	w
8	BS	CAN	(8	H	X	h	x
9	HT	EM)	9	I	Y	i	y
A	LF	SUB	*	:	J	Z	j	z
B	VT	ESC	+	;	K		k	
C	FF	FS	,	<	L		l	
D	CR	GS	-	=	M		m	
E	SO	RS	.	>	N	^	n	
F	SI	VS	/	?	O	_	o	DEL

画面表示できるASCIIコードは、"0X20" ~ "0X7F" までのASCIIコードです。

また、"0X00" ~ "0X1F" のASCIIコードは、制御用コードとなります。

0X00="Null"、0X20のSPは、スペースを示します。

[3] パラレル入出力

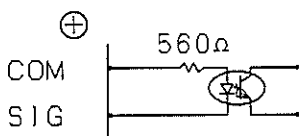
<パラレル入力: EXT-IN>

ポートNo.	ビット	ピンNo.
1	0	1: COM
		2
	1	3: COM
		4
	2	5: COM
		6
	3	7: COM
		8
	4	9: COM
		10
	5	11: COM
		12
	6	13: COM
		14
	7	15: COM
		16
2	-----	17: COM
	0	18
	1	19
	2	20
	3	21
	4	22
	5	23
	6	24
7	25	
3	-----	26: COM
	0	27
	1	28
	2	29
	3	30
	-----	31: COM
	4	32
	5	33
6	34	
7	35	

<パラレル出力: EXT-OUT>

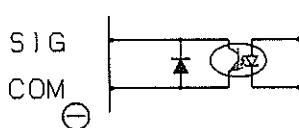
ポートNo.	ビット	ピンNo.
1	0	1
		2: COM
	1	3
		4: COM
	2	5
		6: COM
	3	7
		8: COM
	4	9
		10: COM
	5	11
		12: COM
	6	13
		14: COM
	7	15
		16: COM
2	0	17
	1	18
	2	19
	3	20
	4	21
	5	22
	6	23
	7	24
-----	25: COM	
3	0	26
	1	27
	2	28
	3	29
	-----	30: COM
	4	31
	5	32
	6	33
7	34	
-----	35: COM	

入力部



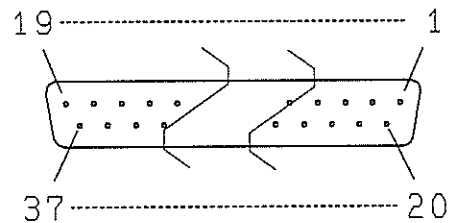
定格使用電圧 5-24VDC
最大印加電圧 30VDC

出力部

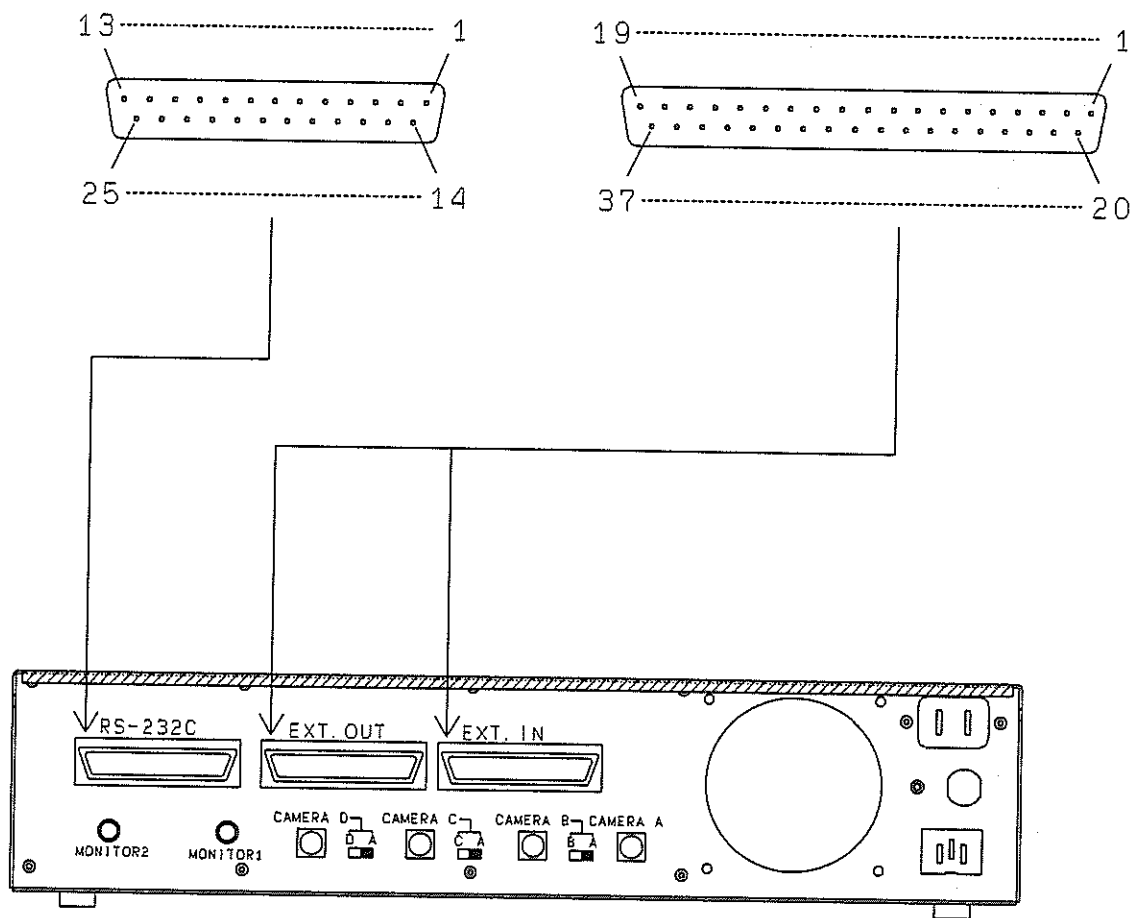


定格使用電圧 5-24VDC
最大許可能範囲 5-30VDC
最大負荷電流 30mA

D-SUBコネクタピン配置: 本体側
(入力、出力共用)



イメージチェッカ30P・30RPの平行入出力ポート、RS232Cピン配置図



図は、コントローラ本体側ピン配置です。

[4] プログラム入力シート (コピーして、ご使用ください)

[プログラム：PM, P I] T I T L E :

No	ラベル	コマンド	オペランド	解説
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				
21				
22				
23				
24				
25				
26				
27				
28				
29				
30				
31				
32				
33				
34				
35				
36				
37				
38				
39				
40				
41				
42				
43				
44				
45				
46				
47				
48				
49				
50				
51				
52				
53				
54				
55				
56				
57				
58				
59				
60				
61				
62				
63				
64				

[5] エラーメッセージ一覧

e 1 0 0	ファイル名の長さは15文字までです
e 1 0 1	ファイル番号の指定範囲は' 1' ~' 99' です
e 1 0 2	ステップ番号の指定範囲は' 1' ~' 64' です
e 1 0 3	コピー元ファイルが存在しません
e 1 0 4	マージ元ファイルが存在しません
e 1 0 5	ステップ数がオーバーする為マージ出来ません
e 1 0 6	プロテクトされています。修正/削除は出来ません
e 1 0 7	ジャンプ先にプログラムが書かれていません。
e 1 0 8	プログラムフォーマットが間違っています
e 1 0 9	Vレジスタ番号の指定範囲は' 1' ~' 1000' です
e 1 1 0	Aレジスタ番号の指定範囲は' 1' ~' 1000' です
e 1 1 1	ビットの指定範囲は' 0' ~' 31' です
e 1 1 2	ポートアドレス指定が間違っています
e 1 1 3	カラムの指定範囲は (0, 1) - (61, 24) の間です
e 1 1 4	ドットの指定範囲は (1, 33) - (489, 472) の間です
e 1 1 5	品種切り替えの指定が間違っています
e 1 1 6	メモリの指定が間違っています
e 1 1 7	プリントモードの指定が間違っています
e 1 1 8	演算エラー：SQRTは正の数でのみ演算可能です
e 1 1 9	0除算エラー：0では割れません
e 1 2 0	演算エラー：レジスタ容量をオーバーしました
e 1 2 1	品種プログラムが設定されていません
e 1 2 2	チェッカが設定されていません
e 1 2 3	プログラムフォーマットが間違っています。
e 1 2 4	プログラムの長さは59文字までです。
e 1 2 5	コメントの長さは30文字までです
e 1 2 6	プログラム中にエラーあり
e 1 2 7	コマンドが存在しません
e 1 2 8	オペランドの使用法が間違っています
e 1 2 9	ラベルが存在しないか2重定義されています
e 1 3 0	Vレジスタ番号の指定範囲は' 1' ~' 991' です
e 1 3 1	Aレジスタ番号の指定範囲は' 1' ~' 841' です
e 1 3 2	直接数値の入力範囲は-32768~32767です
e 1 3 3	J I Sコードの入力範囲は0x2121~0x415fです
e 1 3 4	1文字の指定が' ' ' で囲まれていません
e 1 3 5	文字列の指定が' " ' で囲まれていません
e 1 3 6	文字列の長さが15文字を越えています
e 1 3 7	ファイル名の長さが15文字を越えています
e 1 3 8	Vレジスタ番号の指定範囲は' 1' ~' 1000' です
e 1 3 9	Aレジスタ番号の指定範囲は' 1' ~' 1000' です

e 1 4 0	チェッカナンバーが範囲を越えています
e 1 4 1	2値化レベルの入力範囲は' 0' ~' 63' です
e 1 4 2	CURSORの指定範囲は(0,0)-(2047,479)の間です
e 1 4 3	CURSORの指定範囲は(0,0)-(1023,991)の間です
e 1 4 4	CALL/GOSUBのネストは3 2レベルまでです
e 1 4 5	プログラムの書き込みは6 4ステップまでです
e 1 4 6	プログラムが書かれていません
e 1 4 7	MLxxの指定範囲は(0,17)-(496,472)の間です
e 1 4 8	チェッカ引用時の第二パラメータはVnのみです
e 1 4 9	強制的にPM1に変更しました
e 1 5 0	No.の指定が間違っています
e 1 5 1	アスキーコードの指定が間違っています
e 1 5 2	座標の指定が間違っています
e 1 5 3	文字数の指定は自然数です
e 1 5 4	ジャンプ先のステップ数指定が間違っています
e 1 5 5	チェッカNo.モード指定が範囲を越えています
e 1 5 6	MOVExxの指定がV1000を越えています
e 1 5 7	文字数の指定は' 1' ~' 1000' の範囲です
e 1 5 8	Aレジスタ番号の指定がA1000を越えています
e 1 5 9	タイマーの設定値が間違っています
e 1 6 0	チェッカNo.の指定が間違っています
e 1 6 1	S I Oモードの設定値の指定が間違っています
e 1 6 2	文字数の指定が間違っています
e 1 6 3	J I Sコードの指定が間違っています
e 1 6 4	プログラムの長さは3 3文字までです
e 1 6 5	Sレジスタ番号の指定範囲は' 1' ~' 8' です
e 1 6 6	HEXの直接数値の入力範囲は 0x0~0 x f f f fです
e 1 6 7	W H I L E文が入れ子になっています。
e 1 6 8	選択したエラー発生ステップが見つかりません
e 1 6 9	ラベルの書式が間違っています
e 1 7 0	増分は0を指定出来ません
e 1 7 1	プロテクトフォーマットエラー
e 1 7 2	タイムアウトの指定範囲は' 2 0' ~' 2 0 0 0 0' です。
e 1 7 3	コマンド実行後のジャンプ先がありません
e 1 7 4	このコマンドは実行できません
e 1 7 5	オペランドの設定が範囲を越えています。

■ マニュアル改訂履歴

マニュアルバージョン: Version DP (P/RP) -1K
 イメージチェッカコントローラ: Version 3.0P/3.0RP

発行日付	マニュアルバージョン	コントローラバージョン	改訂内容
910301	Version DP 1RP	1.1P (1.0)	初版
910701	Version DP (P/RP) -1K	Ver 3.0P (2.0) Ver 3.0RP (2.0)	① 30P・30RPを1冊化 ② CPU Ver Upに対応 ・WHILER (~RET) 追加 ・瞬時停電ビット追加 ③ 漢字制御コマンド (プリントアウト) 時の注意追加 ④ エラー処理プログラム追加

本品は、”外国為替及び外国貿易管理法”で定められた戦略物資に該当します。

本品を輸出するとき、または国外に持ち出すときは、日本国政府の輸出許可が必要です。

マニュアル作成に際しまして細心の注意を行っておりますが、万一誤り等がございましたら下記までご連絡を頂きましたら幸いです。

〒571-71 大阪府門真市門真1048 松下電工(株) FAシステム機器事業部
 イメージチェッカマニュアル係