

TOSHIBA

汎用プログラマブルコントローラ

PROSEC **T** **SERIES**

命令語説明書(ラダー, SFC編)

TS SERIES

はじめに

本書の目的

この「Tシリーズ命令語説明書（ラダー、SFC編）」はプログラマブルコントローラPROSEC Tシリーズ（T3H/T3/T2N/T2E/T2/T1S/T1）がサポートするプログラミング言語のラダー言語、SFC言語の仕様の詳細を、例題とともに説明したものです。

説明書体系

Tシリーズシステムの説明書として以下の種類の説明書を準備しています。

T3/T3H本体ハードウェア説明書 (UM-TS03***-J002)

プログラマブルコントローラT3H
T3の基本部ハードウェアについて
構成、仕様、取付・配線方法、保守
・保全方法が説明されています。

T3/T3H本体機能説明書 (UM-TS03***-J003)

T3H T3のCPUが持っている
機能とその使用方法、ユーザプログラ
ムの作成に必要な情報について説
明されています。

T2N製品説明書 (UM-TS02N**-J001)

プログラマブルコントローラT2N
のハードウェア（基本部、基本I/
O）および本体機能について説明さ
れています。

T2E製品説明書 (UM-TS02E**-J001)

プログラマブルコントローラT2E
のハードウェア（基本部、基本I/
O）および本体機能について説明さ
れています。

T 2 製品説明書 (UM-TS02***-J001)	プログラマブルコントローラ T 2 の基本的ハードウェアについて構成、使用、取付、配線方法、保守・保安方法が説明されています。 また、入出力モジュールの取扱いについても説明しています。 T 2 が持っている機能とその使用方法、ユーザープログラムの作成に必要な情報について説明されています。
T 1 / T 1 S 製品説明書 (UM-TS01***-J001)	プログラマブルコントローラ T 1 のハードウェアと機能、使用方法について説明されています。
T シリーズ命令語説明書<ラダー、S F C 編> (UM-TS03***-J004)	本書 T シリーズがサポートするプログラム言語のラダーと S F C について、各命令語の仕様詳細が説明されています。
T - P D S for Windows 操作説明書 (UM-TS03***-J038)	入門編 プログラムの起動からプログラムの作成、保管、モニタリングなどの基本的な操作手順が、例題に沿って説明されています。
T シリーズプログラマ操作説明書 (UM-TS03***-J006)	入門編 プログラムの起動からプログラムの作成、保管、モニタリングなどの基本的な操作手順が、例題に沿って説明されています。
T シリーズプログラマ操作説明書 (UM-TS03***-J007)	応用編 プログラムの作成や保管、データの設定などのプログラムのコマンド操作が機能ごとに説明されています。
T シリーズコンピュータリンク機能説明書 (UM-TS03***-J008)	T シリーズの C P U が内蔵しているコンピュータリンク機能について、仕様、取り扱い方法が説明されています。

目次

はじめに

目次

概要	1
1. 数値	2
1. 1 数値データの大きさ	2
1. 2 単長データの表現の種類	4
1. 3 倍長データの表現の種類	8
2. オーバーフロー、アンダーフロー、キャリーフラグ、エラーフラグ	11
2. 1 オーバーフロー、アンダーフロー	11
2. 2 キャリーフラグ	13
2. 3 エラーフラグ	16
3. 命令語の修飾	19
3. 1 インデックス修飾	19
3. 2 桁指定	21
4. 命令語のパルス化	22
5. 命令語の書式	23
6. ステップ数	24
7. 命令語ページの見方	26
8. プログラミング上の留意点	27
8. 1 命令のパルス化機能使用時の注意	27
8. 2 縦型命令の入力	28
8. 3 カウンタ (C N T) 命令出力	29
8. 4 サブルーチンコール (C A L L) 命令	29
8. 5 ステップシーケンス命令	29
8. 6 S F C (シーケンシャル・ファンクション・チャート)	30
8. 7 タイマレジスタ	30
8. 8 ファイルレジスタ F 1 0 0 0 0 以降のプログラミング	30
9. 命令語一覧	31
9. 1 ラダー言語	31
9. 2 S F C 言語	38
10. 各機種種の相違点について	40
10. 1 レジスタ／デバイスの範囲	40
10. 2 桁指定	40

命令語仕様	41
ラダー言語	43
S F C 言語	323
さくいん	336

概 要

この説明書では、各命令語の書式、取扱いなどについて説明します。また次ページより本書をご理解いただくにあたり、必要な知識について説明していますので、よくご覧下さい。

ここでは、次の事柄について説明します。

1. Tシリーズで取り扱える数値データについて
2. オーバーフロー、アンダーフロー、キャリーフラグ、エラーフラグについて
3. 命令語の修飾（インデックス、桁指定）について
4. 命令語のパルス化について
5. 命令語の書式について
6. ステップ数の計算方法について
7. 命令語ページの見方
8. プログラミング上の注意について
9. ラダー、SFC言語の一覧
10. 各機種の相違点について

では、各々について説明を行います。

なお、キーワードはアンダーラインで示しています。

1. 数值

1. 1 数値データの大きさ

各命令語で扱われる数値データは、16ビット単位の集合体である1ワードというサイズを基本にしています。これを単長データと呼び、10進で0～65535のデータを扱うことができます。Tシリーズはこのデータを16進で表示することができ、単長データでは0000～FFFFまでとなります。実際に表示される時には10進データと区別する意味で、16進データの先頭にはH(Hexadecimalの頭文字)が付きH0000～HFFFFの範囲で表現されます。

単長データ

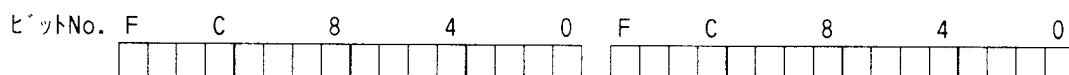


この図で小さな箱（0～F）の中には、ONまたはOFFの情報が格納されています。つまり16桁の2進数とみなすことができます。この2進数を10進数に換算すると16桁すべてOFFの時は0、16桁すべてONの時には65535になります。

したがって単長データでは0～65535のデータを扱うことができます。

さらに 6 5 5 3 5 よりも大きな数値データを扱うことができます。これは単長データを 2 つ連結して 3 2 ビット (2 ワード) のデータにしたもので、倍長データと呼びます。倍長データは 1 0 進で 0 ~ 4 2 9 4 9 6 7 2 9 5 までのデータを扱うことができます。

倍長データ



基本的な考え方は単長データと同じで、桁数が倍になっただけです。その分大きな数値を扱うことができるようになり、その範囲は0～4 294 967 295となります。16進で表示される場合H 00000000～H FFFFFFFFとなります。

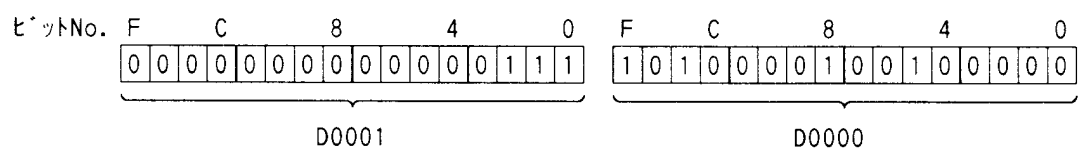
補足

単長データ、倍長データなどの数値データが実際にTシリーズで扱われる時には、メモリ領域のレジスタと呼ばれるところに格納されます。このレジスタのサイズは1ワード（16ビット）単位ですので、単長データを扱う時にはレジスタ1個を使用すればよいのですが、倍長データを扱う時にはレジスタ2個を必要とします。

倍長データの場合の例をデータレジスタ D 0 0 0 0 と D 0 0 0 1 を用いて説明します。

(例)

仮に 5 0 0 0 0 0 というデータを D 0 0 0 0、D 0 0 0 1 を連結したレジスタに転送した場合、次のようになります。



このとき D 0 0 0 0 を 下位レジスタ (または 下位ワード)、D 0 0 0 1 を 上位レジスタ (または 上位ワード) と呼びます。

T シリーズで扱える数値データは基本的にこの単長データと倍長データだけです。この 2 種類のデータに符号、B C D、浮動小数点のような表現上の付加価値機能が付いています。

次に単長データと倍長データに分けてこの表現について説明します。

1. 2 単長データの表現の種類

単長データには、次の表現があります。

- 1 符号無しデータ
- 2 符号付きデータ
- 3 B C D データ

1. 2. 1 符号無しデータ

1. 1 項の単長データの説明と同じです。16ビット全ビットが数値データとなり、データ範囲は0～65535となります。

1. 2. 2 符号付きデータ

コンピュータの内部では負の数値そのものを扱うことはできませんが、便宜的に表現する方法があります。Tシリーズでは一般的によく用いられる2の補数表現という方法をとっています。

2の補数表現

(例)

仮に-10000という10進データを16ビットの2の補数表現に変換するには次のような方法で行います。

まず正数10000を2進数に変換すると次のようになります。

ビットNo.	F	C				8				4				0			
	0	0	1	0	0	1	1	1	0	0	0	1	0	0	0	0	

次に全ビットを反転します(0と1を入れ換える)。

ビットNo.	F	C		8		4		0							
	1	1	0	1	1	0	0	0	1	1	1	0	1	1	1

最後に1を加えると-10000の2の補数表現になります。

ビットNo.	F	C			8			4			0					
	1	1	0	1	1	0	0	0	1	1	1	1	0	0	0	0

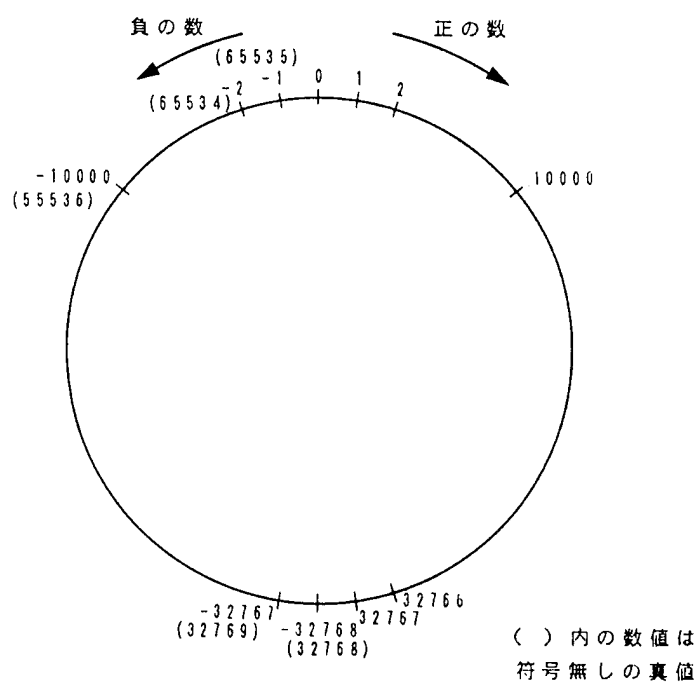
↑ 符号ビット

単長データでの2の補数表現ではFビット目は必ず符号ビットとなります。

F = 0 のとき……正の数

F = 1 のとき……負の数

したがって数値データとしては0～Eまでの15ビットを使うことになり、範囲は－32768～32767となります。下図に概念図を示します。



1. 2. 3 B C D データ

B C D は Binary Coded Decimal (2 進化 1 0 進) のことで、4 ビットの 2 進数 1 組で 1 桁の 1 0 進数を表現する方法です。
次に 0 ～ 2 0 までの対応例を示します。

1 0 進数	B C D データ	1 0 進数	B C D データ
0	0 0 0 0 0 0 0 0	1 0	0 0 0 1 0 0 0 0
1	0 0 0 0 0 0 0 1	1 1	0 0 0 1 0 0 0 1
2	0 0 0 0 0 0 1 0	1 2	0 0 0 1 0 0 1 0
3	0 0 0 0 0 0 1 1	1 3	0 0 0 1 0 0 1 1
4	0 0 0 0 0 1 0 0	1 4	0 0 0 1 0 1 0 0
5	0 0 0 0 0 1 0 1	1 5	0 0 0 1 0 1 0 1
6	0 0 0 0 0 1 1 0	1 6	0 0 0 1 0 1 1 0
7	0 0 0 0 0 1 1 1	1 7	0 0 0 1 0 1 1 1
8	0 0 0 0 1 0 0 0	1 8	0 0 0 1 1 0 0 0
9	0 0 0 0 1 0 0 1	1 9	0 0 0 1 1 0 0 1
		2 0	0 0 1 0 0 0 0 0

単長データのサイズは 1 6 ビットのため、B C D データの容量は 4 桁となり、H 0 0 0 0 ～ H 9 9 9 9 まで扱うことができます。

次に加算を例に違いを説明します。

(例)

B C D データの加算 (B C D 加算)

H 3 7 6 5 と H 4 8 4 9 を加算すると

ビットNo.	F		C			8			4			0
	0	0	1	1	0	1	1	1	0	1	1	0
			3			7			6			5
	0	1	0	0	1	0	0	0	0	1	0	0
十)			4			8			4			9
	1	0	0	0	0	1	1	0	0	0	1	0
			8			6			1			4

10進のため4ビット毎
に9で桁上がりする。

16進表現の加算 (通常の加算)

H 3 7 6 5 と H 4 8 4 9 を加算すると

ビットNo.	F		C			8			4			0
	0	0	1	1	0	1	1	1	0	1	1	0
			3			7			6			5
	0	1	0	0	1	0	0	0	0	1	0	0
十)			4			8			4			9
	0	1	1	1	1	1	1	1	0	1	0	1
			7			F			A			E

16進のため4ビット毎
にFで桁上がりする。
F = (1 1 1 1) のため
結果として2進の加算と
等しくなる。

1. 3 倍長データの表現の種類

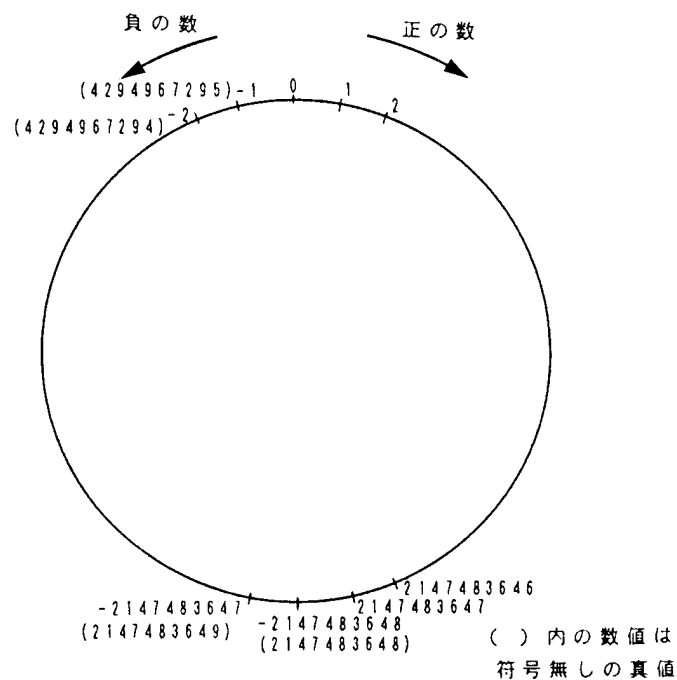
倍長データには次の表現があります。

- 1 符号付きデータ
- 2 B C D データ
- 3 浮動小数点表現データ (T 2 E / T 2 N / T 3 / T 3 H)

1. 3. 1 符号付きデータ

基本的には単長データの符号付きと同じで、2の補数表現で表します。桁数が16ビットから32ビットになった分、大きな数値データを扱うことができます。

数値データの範囲は-2147483648～2147483647となります。下図に概念図を示します。



1. 3. 2 BCDデータ

BCDデータも単長データのBCDデータと同じです。倍長データの場合32ビットのため、BCDデータの容量は8桁となり、0～99999999まで扱うことができます。

1. 3. 3 浮動小数点データ (T2E/T2N/T3/T3H)

今まで説明してきた2進数演算機能 (BCDデータは除く) は、単長、倍長ともにすべて固定小数点データと呼ばれるものです。これらの方式では整数データしか扱うことができませんでした。

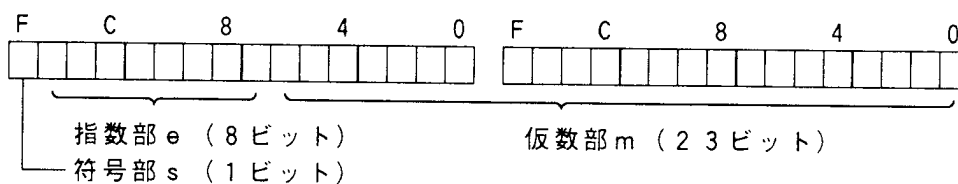
T2E/T2N/T3/T3Hでは小数点以下の実数を扱うことができる浮動小数点データの演算機能を持っています。(浮動小数点データはT1/T2では扱えません。)

浮動小数点データ

任意の大きさの実数を浮動小数点形式で表す場合、次のように表現します。

$$\text{実数値} = (\overset{\text{符号}}{S}) (\overset{\text{仮数}}{M} \times \overset{\text{指数}}{2^E})$$

ここでSは符号、mは1より小さい小数、Eは-127～128となります。実際にT3で扱われる時のデータフォーマットは次のようになります。



先ほどの公式にて、実数値を得るには、

$$S = s$$

$$M = 1 + \frac{m}{2^{23}}$$

$$E = e - 127$$

より、

$$\text{実数値} = (s) \left(\left(1 + \frac{m}{2^{23}} \right) \times 2^{e-127} \right)$$

となります。

補足

- ・ T2E/T2N/T3/T3Hの浮動小数点のデータフォーマットはIEEE754規格に準じています。
- ・ 演算は必ず浮動小数点データ同士で行います。

次ページに変換フォーマットを示しますのでご参照下さい。

浮動小数点表現			倍長 1 6 進数	真 値	命令語での扱い	備考		
符号	指数部	仮数部						
0	HFF	H7FFFFFF } H000000	H7FFF FFFF } H7F80 0000	$+6.80565 \times 10^{-38}$: 	$+3.40282 \times 10^{-38}$ にてリミット			
	HFE	H7FFFFFF } H000000	H7F7F FFFF } H7F00 0000	$+3.40282 \times 10^{-38}$ } $+1.70141 \times 10^{-38}$	正の最大値			
	:							
	H96	H7FFFFFF } H000000	H4B7F FFFF } H4B00 0000	$+16777215$ } $+8388608$		※		
	:							
	H7F	H7FFFFFF } H000000	H3FFF FFFF } H3F80 0000	$+1.99999988$ } $+1$				
	:							
	H01	H7FFFFFF } H000000	H00FF FFFF } H0080 0000	$+2.35099 \times 10^{-38}$ } $+1.17549 \times 10^{-38}$	正の最小値			
	H00	H7FFFFFF } H000000	H007F FFFF } H0000 0000	$+5.87747 \times 10^{-39}$	0			
	1	H00	H000000 } H7FFFFFF	H8000 0000 } H807F FFFF			-5.87747×10^{-39} : 	
		H01	H000000 } H7FFFFFF	H8080 0000 } H80FF FFFF			-1.17549×10^{-38} } -2.35099×10^{-38}	負の最小値
:								
H7F		H000000 } H7FFFFFF	HBFF80 0000 } HBFFF FFFF	-1 } -1.99999988				
:								
H96		H000000 } H7FFFFFF	HCB00 0000 } HCB7F FFFF	-8388608 } -16777215				
:								
HFE		H000000 } H7FFFFFF	HFF00 0000 } HFF7F FFFF	-1.70141×10^{-38} } -3.40282×10^{-38}	負の最大値			
HFF		H000000 } H7FFFFFF	HFF80 0000 } HFFFF FFFF	-6.80565×10^{-38}	-3.40282×10^{-38} にてリミット			

※ -16777215 から 16777215 までの範囲内の整数は、浮動小数点変換を行っても誤差を生じません。

2. オーバーフロー、アンダーフロー、キャリーフラグ、エラーフラグ

2. 1 オーバーフロー、アンダーフロー

1項で説明した通り、負の整数を表すためには2の補数表現を用いています。この方法では、単長あるいは倍長の数値データを正の領域と負の領域に分けて使用しているため、演算によっては自身の領域からあふれ出てしまい他方の領域内で解を得ることがあります。

単長データの時3 2 7 6 7、倍長データの時2 1 4 7 4 8 3 6 4 7からあふれて負の領域に入ることをオーバーフローと呼びます。

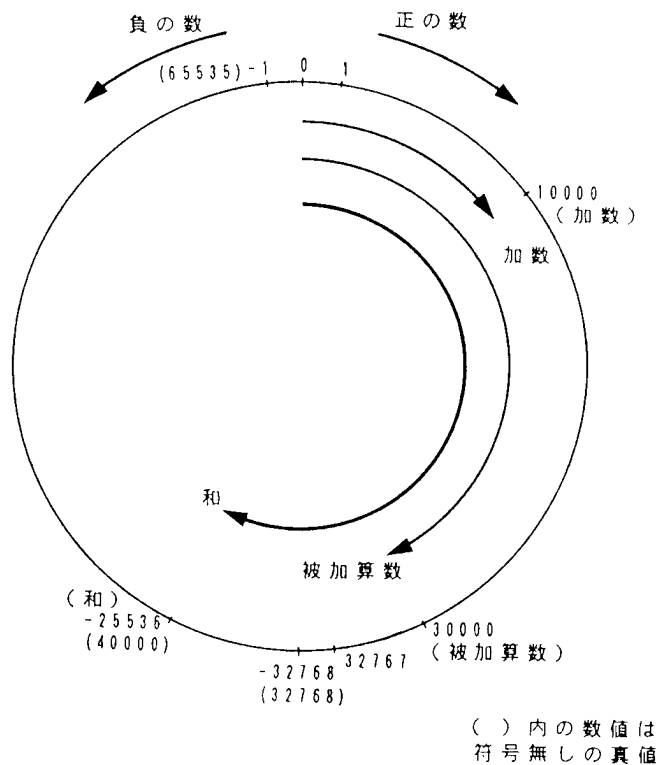
単長データの時-3 2 7 6 8、倍長データの時-2 1 4 7 4 8 3 6 4 8からあふれて正の領域に入ることをアンダーフローと呼びます。

次にオーバーフローが発生する計算例と、アンダーフローが発生する計算例を示します。

(例1) オーバーフロー例(単長)

$$30000 + 10000 = 40000$$

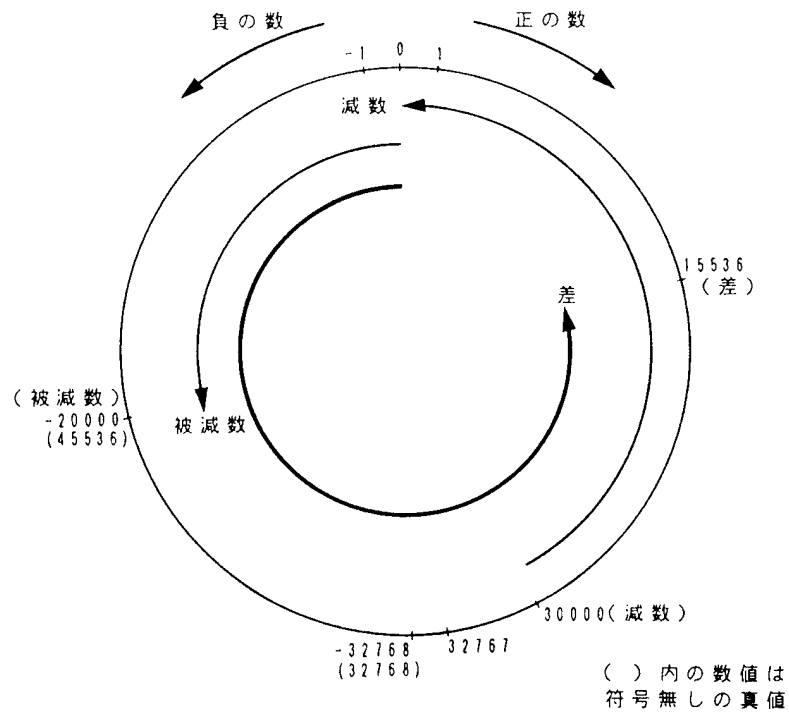
単長データの最大値3 2 7 6 7を超え、負の領域に入るため、オーバーフローが発生します。次に概念図を示します。



(例 2) アンダーフロー例 (単長)

$$-20000 - 30000 = -50000$$

単長データの最小値 -32768 を超え、正の領域に入るため、アンダーフローが発生します。次に概念図を示します。



2. 2 キャリーフラグ (C F)

Tシリーズでは基本的に単長データあるいは倍長データによる演算しかありません。このような固定長データでの加・減算では、Fビット目からの桁上がり、もしくはFビットへの桁下がり（桁借り）が発生することがあります。この現象を総称してキャリーが発生するといいます。Tシリーズではキャリー付きの加・減算命令を持っており、キャリーが発生するとデバイスS0050をセットします。

このS 0 0 5 0をキャリーフラグと呼びます。

次にキャリーが発生する単長データでの演算例を示します。

(例 1) 桁上がりによるキャリーの発生

$$-1\ 0\ 0\ 0\ 0 + 2\ 0\ 0\ 0\ 0 = 1\ 0\ 0\ 0\ 0$$

ビットNo. F C 8 4 0

1	1	0	1	1	0	0	0	1	1	1	1	0	0	0	0

— 1 0 0 0 0 (5 5 5 3 6)

0	1	0	0	1	1	1	0	0	0	1	0	0	0	0

2 0 0 0 0

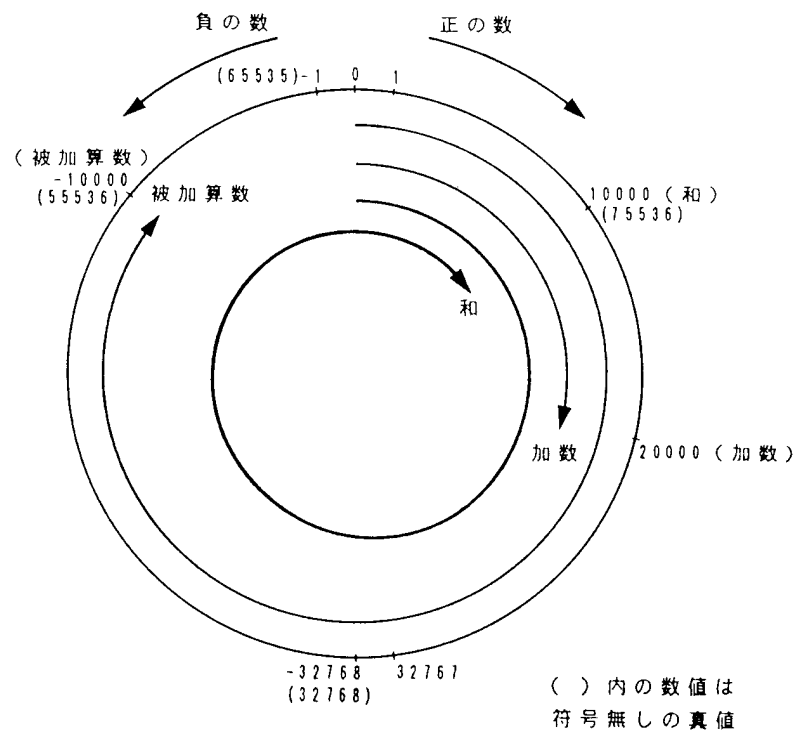
+))

キャリーフラグ (S0050) 1
0
0
1
0
0
1
1
1
0
0
0
1
0
0
0
0

1 0 0 0 0 (7 5 5 3 6)
 () 内は符号無しの真値

桁上がり

次に概念図を示します。端的に言えばこのように解が0点を通過した時キャリーが発生します。



(例2) 桁下がりによるキャリーの発生

$$10000 - (-30000) = 40000$$

ビットNo. F C 8 4 0

0010011100010000 10000

1000101011010000 - 30000 (35536)

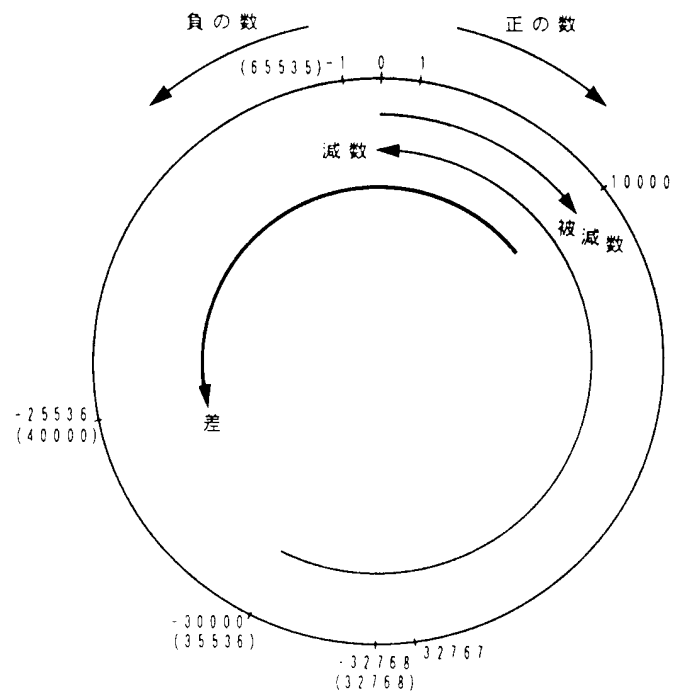
—)

1 1001110001000000 - 25536 (40000)

() 内は符号無しの真値

桁下がり

次に概念図を示します。



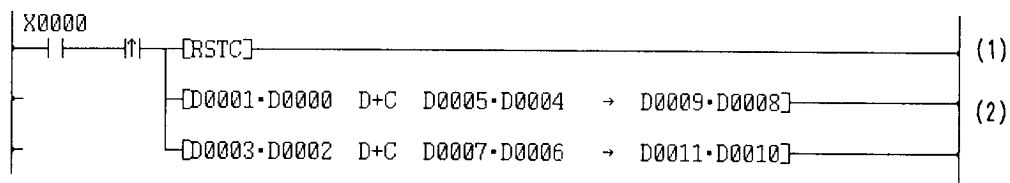
() 内の数値は
符号無しの真値

キャリー付き加減算命令を連結させることにより倍長より大きな数値データの加・減算を行うことができます。

次に4倍長（4ワード）の加算のプログラム例を示します。

（例3）

A接点X0000がONの時D0000～D0003の4倍長データとD0004～D0007の4倍長データを加算してD0008～D0011に4倍長データとして格納します。



(1) キャリーフラグをリセットします。

(2) 2つのキャリー付き加算命令を連結することによって自動的に4倍長データの和が求められます。

2. 3 エラーフラグ (E R F)

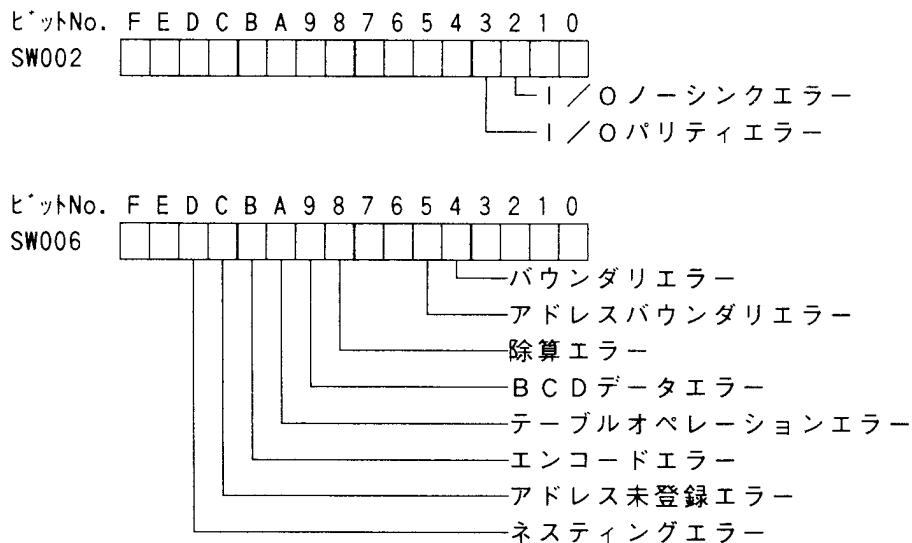
演算エラーが発生すると、デバイス S 0 0 5 1 を O N にセットします。この S 0 0 5 1 をエラーフラグと呼びます。

このフラグは 1 度セットされると、R S T 命令 (F U N 1 1 5) により S 0 0 5 1 をリセットしない限り、セット状態を保持します。

演算エラーには次の種類のものがあります。

- (1) I / O ノーシンクエラー
- (2) I / O パリティエラー
 - a) I / O リードパリティエラー
 - b) I / O ライトパリティエラー
- (3) バウンダリエラー
- (4) アドレスバウンダリエラー
- (5) 除算エラー
- (6) B C D データエラー
- (7) テーブルオペレーションエラー
- (8) エンコードエラー
- (9) アドレス未登録エラー
- (10) ネスティングエラー

これらのエラーが発生すると、エラー要因を判別するため、レジスタ S W 0 0 2 及び S W 0 0 6 に要因識別フラグがエラーフラグと同時にセットされます。



次表に各々のエラーについて説明を行います。

エラー一覧表

エラー名称	識別 フラグ	説明	対処	対応する命令語
I/O ノーシンク エラー	S0022	I/Oからの応答がない。I/O アクセスのリトライを行う が、リトライが4回を越え るとエラーダウン。	I/Oが挿着されているかど うか。また、正確に挿着さ れているかチェックして下 さい。	デバイスI,0またはレジ スタIW,OWをオペランド として使用できる命令語
I/O パリティ エラー	S0023	I/Oリードパリティエラー: I/Oからのデータに異常が ある。 I/Oライトパリティエラー: I/Oのデータに異常がある。 I/Oアクセスのリトライを 行うがリトライが4回を越 えるとエラーダウン。	I/Oが確実に挿着されてい るかどうか。拡張ケーブル に異常がないかチェックし て下さい。	デバイスI,0またはレジ スタIW,OWをオペランド として使用できる命令語
バウンダリ エラー	S0064	ビットオペランド、ワード オペランドまたはテーブル がレジスタ領域を越えた。	オペランド範囲が指定され た領域内におさまるように して下さい。	オペランドにレジスタを 扱うことができる命令語
アドレス バウンダリ エラー	S0065	サブルーチンコール命令、 ジャンプ命令において示さ れるエントリーNo.が0～ 255を越えた。	サブルーチンコール命令、 ジャンプ命令においてエン トリーNo.を0～255の範囲 内で指定して下さい。	CALL, JUMP
除算エラー	S0068	1. 除算命令において除数＝ 0となった。	1. 除数が0以外となるよう にして下さい。	/, D/, U/, DIV, B/, DB/, F/
		2. 符号無し倍長／単長除算 命令で除数≠0だが商がオー バーフローした。	2. 商が65535以下となるよ うに演算を行って下さい。	DIV
BCD データエラー	S0069	BCDの関連命令で、BCDデー タが0～9以外の数値となっ た。	BCDデータの各桁が0～9に なるようにして下さい。	BIN, DBIN, BCD, DBCD B+, B-, B*, B/, DB+, DB-, DB*, DB/, B+C, B-C, DB+C, DB-C
テーブルオペ レーション エラー (T1は除く)	S006A	1. mビットファイルnビット シフト／ローテート命令に おいて ①シフト／ローテート数 がテーブルサイズを越え た。 ②シフト／ローテート数 が1～16以外となった。	1. テーブルサイズ、シフト ／ローテート数の確認を行 って下さい。	TSHR, TSHL TRTR, TRTL TRRC, TRLC
		2. テーブルビットセット／ リセット命令において、 ビットポインタがテーブ ルサイズをオーバーした。	2. テーブルサイズの確認を 行って、ビットポインタ がその範囲内におさま るようにして下さい。	TSET, TRST
		3. プッシュ／ポップ命令に おいてデータ総数がテー ブルサイズをオーバーし た。	3. テーブルサイズの確認を 行ってデータ総数がその 範囲内におさまるよう にして下さい。	PUSH, POPL, POPF

エラー名称	識別 フラグ	説 明	対 処	対応する命令語
エンコード エラー	S006B	エンコード命令において対象テーブルにONビットがない。	ONビットを含むデータを入力して下さい。	ENC
アドレス 未登録エラー	S006C	サブルーチンコール命令、ジャンプ命令において示されるエントリーNo.に対応するプログラムが見つからない。	サブルーチンコール命令、ジャンプ命令で指定されるエントリーNo.と実際のプログラムの対応を確認して下さい。	CALL, JUMP
ネスティング エラー	S006D	1. サブルーチン、FOR～NEXT命令でネスティングが6重を越えた。	1. サブルーチン、FOR～NEXT命令でのネスティングを6重までとして下さい。	CALL, FOR～NEXT
		2. マスタコントロールのネスティングが8重を越えた。	2. マスタコントロールでのネスティングを8重までとして下さい。	MCS, MCR MCSn, MCRn

— 補 足 —

T 1 / T 1 S においては、ネスティング可能数は次の通りです。

- ・ サブルーチン及び F O R ～ N E X T

T 1 : ネスティングなし

T 1 S : 3 重まで

- ・ マスタコントロール

T 1 / T 1 S : ネスティングなし

3. 命令語の修飾

命令語中のレジスタやデバイスを指定する時に、次の2種類の修飾を使用することができます。

1 インデックス修飾

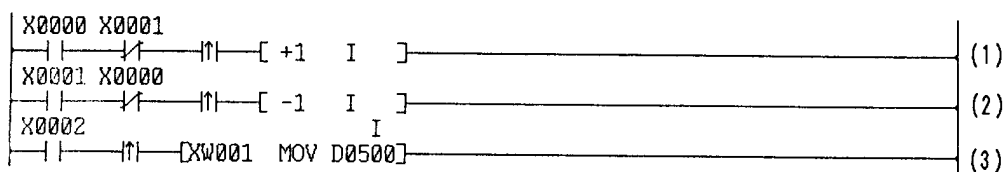
2 桁指定

各々について説明します。

3. 1 インデックス修飾

インデックスレジスタ (I、J、K) を使用することにより、あるレジスタ、デバイス、数値を基準として任意にレジスタ、デバイス、数値の間接指定を行うことができます。I、J、Kの範囲はそれぞれ-32768～32767までです。実際にどのように使用するか、データ転送命令と倍長加算命令によるプログラム例を次に示します。

(例1) データ転送命令

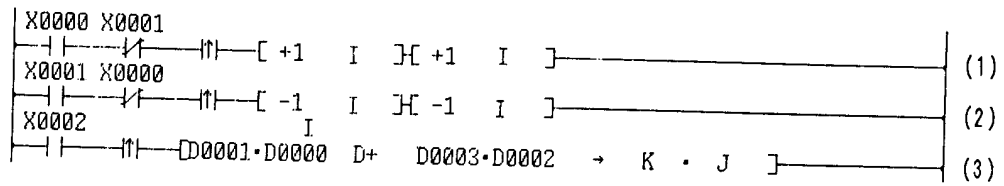


- (1) X0000の立ち上がりでインデックスレジスタIの内容をインクリメントします。
(2) X0001の立ち上がりでインデックスレジスタIの内容をデクリメントします。
(3) X0002の立ち上がりで、XW001の内容をD0500にIの内容を足した番地(たとえばI=500の時はD1000)に転送します。

補足

Iは-32768～32767まで使用できますが、データレジスタの範囲はD0000～D8191のため、D0500を基準とした場合Iの範囲は-500～7691までとなります。この範囲を超えた場合、バウンダリーエラー(領域エラー)となりエラーフラグ(S0051)をONにします。

(例 2) 倍長加算命令



- (1) X 0 0 0 0 の立ち上がりで、インデックスレジスタ I の内容を 2 回 (倍長データのため) インクリメントします。
- (2) X 0 0 0 1 の立ち上がりで、インデックスレジスタ I の内容を 2 回 (倍長データのため) デクリメントします。
- (3) X 0 0 0 2 の立ち上がりで、D 0 0 0 1 · D 0 0 0 0 に I の内容を足した番地 (たとえば I = 5 0 0 の時には D 0 5 0 1 · D 0 5 0 0) の倍長データに D 0 0 0 3 · D 0 0 0 2 の倍長データを加算し、倍長のインデックスレジスタ K · J に格納します。

補足

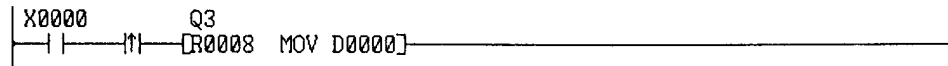
インデックスレジスタを (他のレジスタと同じように) 倍長で使用する時には、必ず「J · I」または「K · J」の組み合わせとなります。「I · K」の組み合わせはできません。

3. 2 桁指定 (T1 / T1Sは除く)

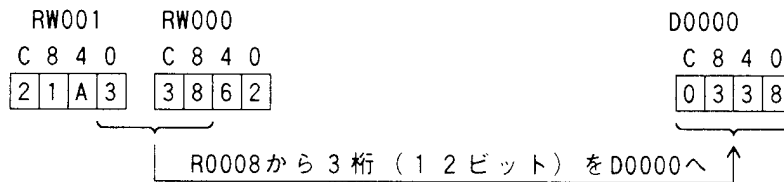
データの転送元や転送先において4ビット単位での桁数の指定を行うことができます。BCDデータなどのマスク処理などで活用することができます。データ転送命令によるプログラム例を示します。(T2 / T2E / T2NではXCHG (FUN18)、MOV (FUN22)の2命令に対してのみ、桁指定が可能です。)

(例1) 桁指定ビットデバイス→レジスタ

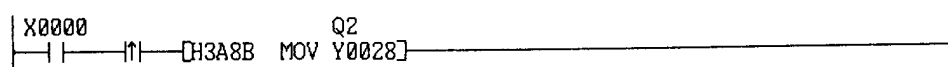
RW000にはH3862、RW001にはH21A3が格納されているとします。



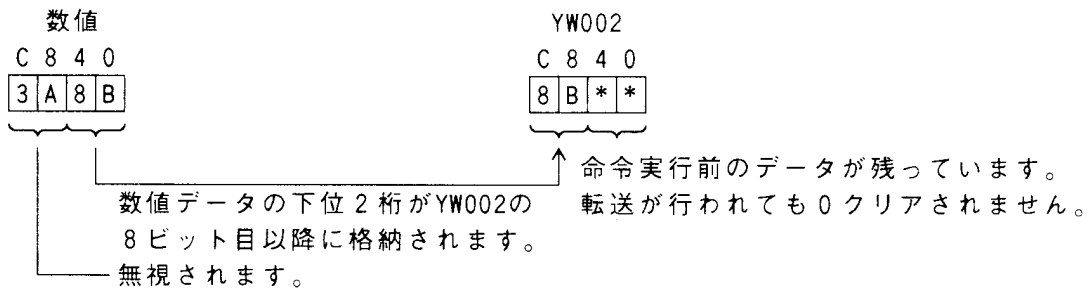
X0000の立ち上がりでR0008から連続した3桁 (= 12ビット) のデータをD0000に転送します。



(例2) 数値→桁指定ビットデバイス



X0000の立ち上がりでH3A8Bのうち、下位2桁 (8ビット) のデータをY0028から連続した2桁 (8ビット) に格納します。



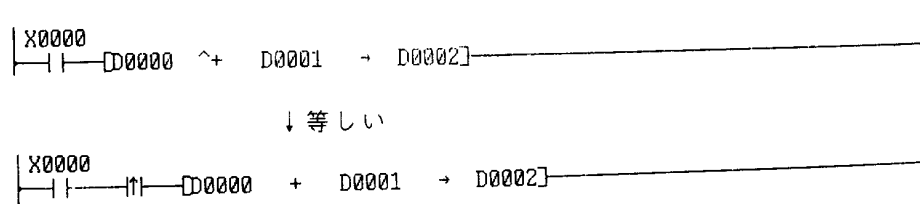
補足

1. 桁指定でQ0を指定すると、1ビット指定となります。
同様に、Q1を指定すると、4ビット指定
Q2を指定すると、8ビット指定
Q3を指定すると、12ビット指定となります。
以降の指定はT3 / T3Hで可能です。
Q4を指定すると、16ビット指定
Q5を指定すると、20ビット指定
Q6を指定すると、24ビット指定
Q7を指定すると、28ビット指定
Q8を指定すると、32ビット指定となります。
2. 桁指定を行うことができるオペランドは、
X, Y, S, L, R, Z, I, O
のうち、使用する命令語で許可されたものです。

4. 命令語のパルス化 (T1/T1Sは除く)

指定された命令語のパルス化（立ち上がり実行化）を行うことができます。パルス化された命令語は、パルス化されていない命令語の直前に立ち上がり微分接点を配置したものと、同じ意味を持ちます。次に加算命令のプログラム例を示します。

(例)



どちらもX0000の立ち上がりで加算命令を実行します。

5. 命令語の書式

四則演算命令、論理演算命令、比較命令などで、キャリー付き、浮動小数点演算のような付加価値機能を使用するときには、次表のような表記方法のきまりがあります。

機 能	表 記	名 称		凡 例
倍長	D [命令語]	倍長加算		$[\textcircled{A}+1\cdot\textcircled{A} \text{ D}+ \textcircled{B}+1\cdot\textcircled{B} \rightarrow \textcircled{C}+1\cdot\textcircled{C}]$
キャリー付き	[命令語] C	キャリー付き加算		$[\textcircled{A} +\text{C} \textcircled{B} \rightarrow \textcircled{C}]$
キャリー付き倍長 ※	D [命令語] C	キャリー付き 倍長加算		$[\textcircled{A}+1\cdot\textcircled{A} \text{ D}+\text{C} \textcircled{B}+1\cdot\textcircled{B} \rightarrow \textcircled{C}+1\cdot\textcircled{C}]$
B C D	B [命令語]	B C D 加算		$[\textcircled{A} \text{ B}+ \textcircled{B} \rightarrow \textcircled{C}]$
倍長 B C D ※	D B [命令語]	倍長 B C D 加算		$[\textcircled{A}+1\cdot\textcircled{A} \text{ DB}+ \textcircled{B}+1\cdot\textcircled{B} \rightarrow \textcircled{C}+1\cdot\textcircled{C}]$
キャリー付き B C D 倍長 ※	D B [命令語] C	キャリー付き B C D 倍長加算		$[\textcircled{A}+1\cdot\textcircled{A} \text{ DB}+\text{C} \textcircled{B}+1\cdot\textcircled{B} \rightarrow \textcircled{C}+1\cdot\textcircled{C}]$
浮動小数点 (T2E/T2N/T3/T3H)	F [命令語]	浮動小数点加算		$[\textcircled{A}+1\cdot\textcircled{A} \text{ F}+ \textcircled{B}+1\cdot\textcircled{B} \rightarrow \textcircled{C}+1\cdot\textcircled{C}]$
テーブル ※	T [命令語]	テーブル転送		$[\textcircled{A} \text{ TMOV}[\text{n}] \textcircled{B}]$
符号無し ※	U [命令語]	比較・符号無し より大きい		$[\textcircled{A} \text{ U} > \textcircled{B}]$

※ T 1 では対応しておりません。

6. ステップ数

各命令語の使用ステップ数は命令語一覧表の中に記載してあります。その中で、命令語によっては、“3～5”といった具合に命令語の使い方によってステップ数が変化するものがあります。ここではその考え方について説明します。

仮にステップ数が” 3 ～ 5 ” という命令語があった時、その命令語の基本使用ステップ数は 3 ステップとなります。

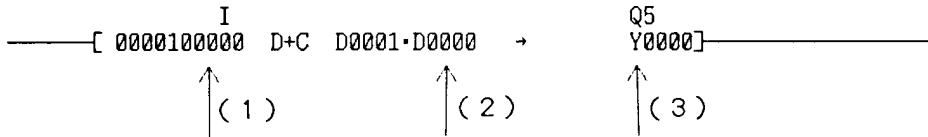
そこに命令語の修飾などがあると、1 アイテムにつき1 ステップ増加します。ステップ数が増える要素には次の3つのものがあります。

- (1) オペランドで倍長の数値データを使用した時、そのデータが 6 5 5 3 5 を超えた場合。
- (2) 数値データにインデックス修飾をつけた場合。
- (3) 桁指定修飾を行った場合。

具体例を次に示します。

(例 1) キャリー付き倍長加算

この命令語の基本使用ステップ数は4ステップです。では次のようにオペランドを設定するとどうなるのかを説明します。



- (1) 倍長数値データで、65535を超えて使用しているので、1ステップ増加します。
数値データにインデックス修飾をつけているのでさらに1ステップ増加します。
- (2) 該当する項目がないので、このオペランドではステップ数は増加しません。
- (3) 桁指定修飾を行っているので1ステップ増加します。

したがって基本使用ステップ数の4ステップに増加分3ステップを加えて計7ステップとなります。

(例2) ASCII変換

この命令語の基本使用ステップ数は3ステップです(変換文字が1文字または2文字までの時)。以降2文字単位に1ステップずつ増加します。

———[アスキー・エン]——— ASC D1000]———

全部で9文字ありますが、最初の2文字は基本使用ステップに含まれます。

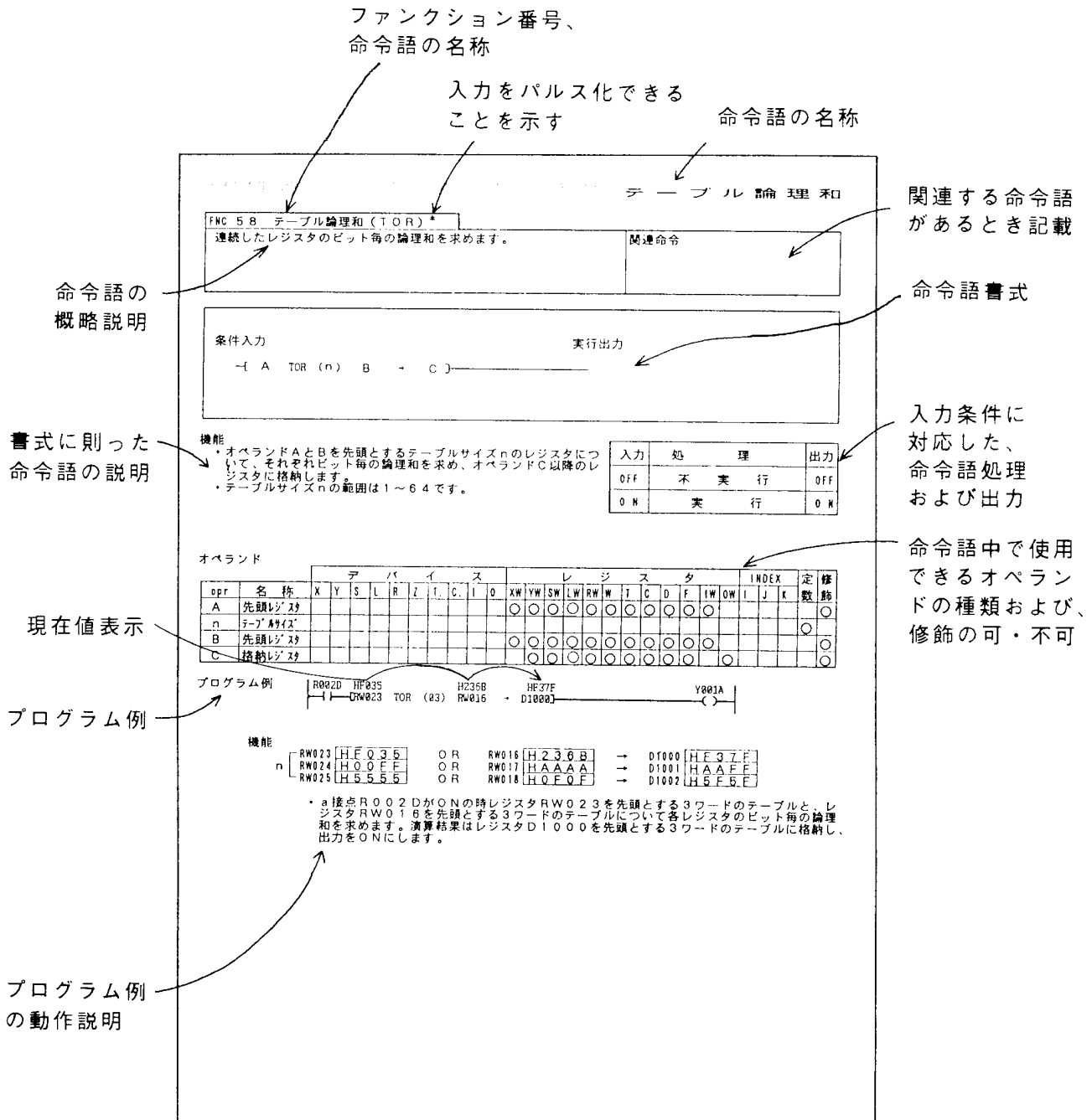
ア ス キ ー ・ エ ン カ ン

↑
基本使用ステップ 計4ステップ
に含まれる。

したがって基本使用ステップ数の3ステップに増加分4ステップを加えて計7ステップとなります。

7. 命令語ページの見方

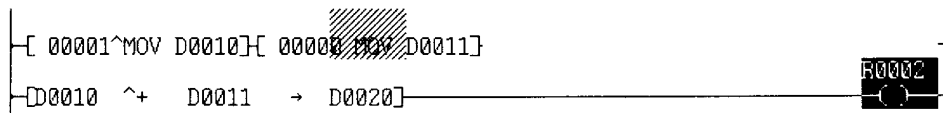
この項では各命令語ページに書かれている情報について説明します。



8. プログラミング上の注意

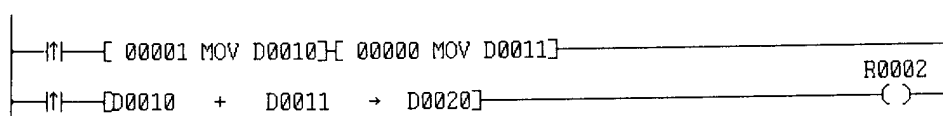
8. 1 命令のパルス化機能使用時の注意

パルス化することができる応用命令を使用する時には、次の点に注意して下さい。
パルス化した命令語の後に命令語を書くことはできません。



パルス化命令に対する接続に誤りがあります

この場合には $\uparrow$ 命令を追加して次のように変更して下さい。



補足

特に出力が条件出力（例えば加算命令のオーバーフロー出力）の場合、出力を使用するときに注意して下さい。

8. 2 縦型命令の入力

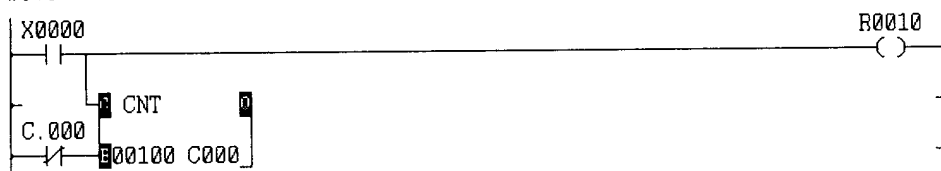
縦型命令には命令語に2つ以上の入力があります。この入力のうちいちばん下の入力を除いた入力には、分流あるいは合流をした後の入力を接続することはできません。

このような回路では分流あるいは合流後に、特殊リレー常時ON (S004F)などのダミー接点を使用して接続して下さい。

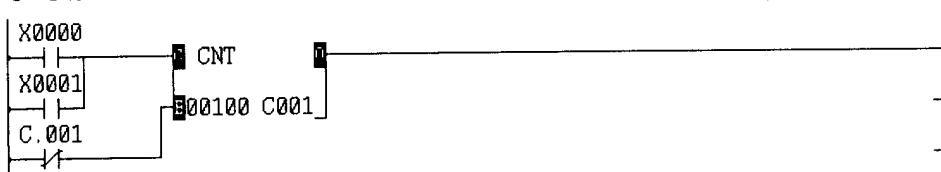
縦型命令としては次に示す5つの命令語があります。

命令語	記号	FUN番号	分流／合流接続	
			不可の入力	可能な入力
カウンタ	C N T	—	C	E
シフトレジスタ	S R	74	D、S	E
双方向シフトレジスタ	D S R	75	D、S、E	L
フリップフロップ	F / F	147	S	R
アップダウンカウンタ	U / D	149	U、C	E

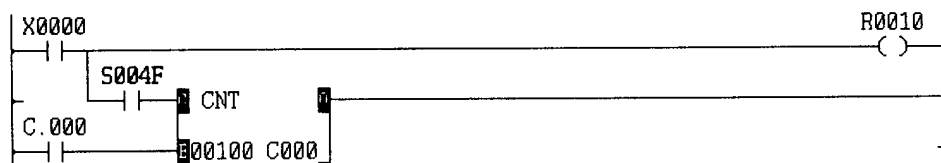
例えば



または



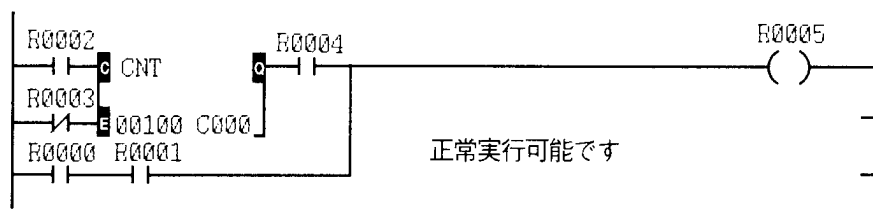
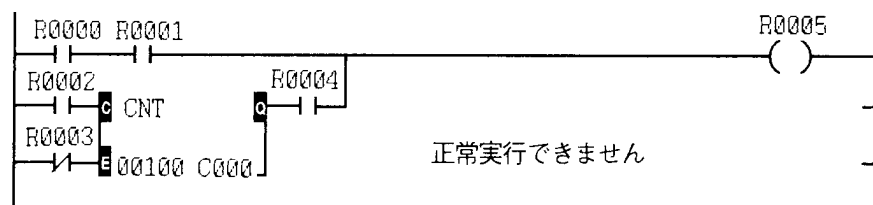
これらの場合には



8. 3 カウンタ（CNT）命令出力

注意事項適用機種：T 2，T 2 E，T 2 N，T 3

CNT 命令の出力が、その上のラインにOR接続されるような回路は作成しないで下さい。そのような回路がある場合、T 2 / T 2 E / T 2 N / T 3 は正常に作動しなくなります。

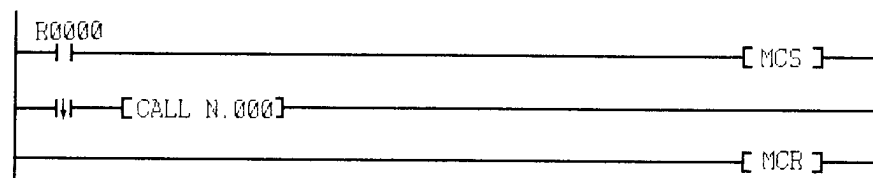


8. 4 サブルーチンコール（CALL）命令

注意事項適用機種：全機種

母線がOFFの状態ですブルーチンがコールされるような回路は作成しないで下さい。母線OFFの状態ですCALL命令が実行された場合はエラーダウンしてしまいます。

例えば下記の回路は正常実行できません。また下記の回路で立ち下がり微分接点 $\neg \downarrow$ トではなく、インバート接点 $\neg$ トを使用した場合も、同様に実行不能です。



8. 5 ステップシーケンス命令

注意事項適用機種：T 2，T 2 E，T 2 N，T 3

ステップシーケンスの階層構成はできません。つまり、一連のステップシーケンス回路中からサブルーチンをコールし、さらに呼び出されたサブルーチン内でステップシーケンスを使用している場合、T 2 / T 2 E / T 2 N / T 3 の動作が異常となります。

またこれと同じ理由で、並列実行される複数のプログラムでステップシーケンスを使用することはできません。ステップシーケンスはメインプログラムでのみ使用して下さい。

8. 6 S F C (シーケンシャル・ファンクション・チャート) 注意事項適用機種: T 2, T 2 E, T 2 N, T 3

並列実行される複数のプログラムで S F C を使用することはできません。S F C はメインプログラムでのみ使用して下さい。

8. 7 タイマレジスタ 注意事項適用機種: T 3 H

T 3 H のタイマレジスタ T 5 1 2 ~ T 9 9 9 は、T 3 H 内部では T 0 0 0 ~ T 5 1 1 とは異なった種別のレジスタとして扱われます。

従って、インデックスやテーブル単位での命令で T 5 1 1 ~ T 5 1 2 をまたぐような使い方はできませんので留意願います。

例えば、[T 4 5 0 T M O V (0 1 0 0) D 0 1 0 0] とプログラミングした場合、T 4 5 0 ~ T 5 1 1 は D 1 0 0 ~ D 1 6 1 に転送されますが、バウンダリーエラーが発生し、それ以降のレジスタは転送が実行されません。

このような場合は、命令を T 5 1 1 と T 5 1 2 を境に 2 つに分けてプログラミングして下さい。

8. 8 ファイルレジスタ F 1 0 0 0 0 以降のプログラミング 注意事項適用機種: T 3 H

T 3 H のファイルレジスタの内、F 1 0 0 0 0 以降のレジスタをプログラム中で使用する場合、[D 1 0 0 0 M O V F 3 0 0 0 0] のような記述は行えません。

プログラム中で F 1 0 0 0 0 以降をアクセスする場合には、インデックス I ・ J ・ K を使用してアドレスの指定を行います。

I ← I の値が 3 0 0 0 0 の場合 F 3 0 0 0 0 となる
[D 1 0 0 0 M O V F 0 0 0 0]

9. 命令語一覧

9. 1 ラダー言語

命令語	記号	○	FUN 番号	機種 ^{*1}					ステップ	ページ
				T3H	T3	T2N/T2E	T2	T1S/T1		

パルス化命令
(○:可)

*1:命令
インデックス
桁指定
(○:可) 但し△:T1Sのみ可
(|:可)
(Q:可)

基本命令										
a 接点	— —			○		○		○		1 43
b 接点	— /—			○		○		○		1 44
ON時微分接点	— ↑—			○		○		○		1 45
OFF時微分接点	— ↓—			○		○		○		1 46
コイル	—()—			○		○		○		1 47
フォーストコイル	—()—			○		○		○		1 48
インバータ	— /—			○		○		○		1 49
インバートコイル	—()—			○		○		○		1 50
ポジティブパルス接点	— P—			○		○		△		1 51
ネガティブパルス接点	— N—			○		○		△		1 52
ポジティブパルスコイル	—(P)—			○		○		△		1 53
ネガティブパルスコイル	—(N)—			○		○		△		1 54
ジャンプコントロールセット	JCS			○		○		○		1 55
ジャンプコントロールリセット	JCR			○		○		○		1 55
終了	END			○		○		○		1 56
オンディレイタイマ	TON			○	○	○	○	○		2 57
オフディレイタイマ	TOF			○	○	○	○	○		2 58
シングルショット	SS			○	○	○		○		2 59
カウンタ	CNT			○	○	○		○		2 60
マスターコントロールセット	MCS			○		○		○		1 61
マスターコントロールリセット	MCR			○		○		○		1 61
マスターコントロールセット n	MCSn	134	○		○		○			2 193
" " リセット n	MCRn	135	○		○		○			2 193
タイマトリガ	TRG	148	○		○		○			2 203

転送命令										
データ転送	MOV	○ 18	○ Q	○ Q	○ Q	○ Q	○ Q	○	3 to 5	62
倍長データ転送	DMOV	○ 19	○ Q	○ Q	○	○	○	○	3 to 6	63
データ否定転送	NOT	○ 20	○ Q	○ Q	○	○	○	○	3 to 5	64
倍長データ否定転送	DNOT	○ 21	○ Q	○ Q	○	○			3 to 6	65
データ交換	XCHG	○ 22	○ Q	○ Q	○ Q	○ Q	○ Q	○	3 to 5	66
倍長データ交換	DXCH	○ 23	○ Q	○ Q	○	○			3 to 5	67
テーブル初期化	TINZ	○ 24	○ Q	○ Q	○	○		△	4 to 6	68
テーブル転送	TMOV	○ 25	○ Q	○ Q	○	○		△	4 to 6	69
テーブル否定転送	TNOT	○ 26	○ Q	○ Q	○	○		△	4 to 6	71

T 1 はパルス化は不可です。

命令語	記号	○	FUN 番号	機種 ^{*)}					ステップ	ページ
				T3H	T3	T2N/T2E	T2	T1S/T1		

パルス化命令
(○:可)

*1:命令
インデックス
桁指定
(○:可) 但し△:T1Sのみ可
(|:可)
(Q:可)

四則演算																			
加算	+	○	27	○		Q	○		Q	○		○		○		4 to 7	73		
減算	-	○	28	○		Q	○		Q	○		○		○		4 to 7	74		
乗算	*	○	29	○		Q	○		Q	○		○		○		4 to 7	75		
除算	/	○	30	○		Q	○		Q	○		○		○		4 to 7	76		
倍長加算	D+	○	31	○		Q	○		Q	○		○		○		4 to 9	77		
倍長減算	D-	○	32	○		Q	○		Q	○		○		○		4 to 9	78		
倍長乗算	D*	○	33	○		Q	○		Q	○		○				4 to 9	79		
倍長除算	D/	○	34	○		Q	○		Q	○		○				4 to 9	80		
キャリー付き加算	+C	○	35	○		Q	○		Q	○		○		○		4 to 7	81		
キャリー付き減算	-C	○	36	○		Q	○		Q	○		○		○		4 to 7	83		
キャリー付き倍長加算	D+C	○	37	○		Q	○		Q	○		○				4 to 9	85		
キャリー付き倍長減算	D-C	○	38	○		Q	○		Q	○		○				4 to 9	86		
符号無し乗算	U*	○	39	○		Q	○		Q	○		○		△		4 to 7	87		
符号無し除算	U/	○	40	○		Q	○		Q	○		○		△		4 to 7	88		
符号無し倍長単長除算	DIV	○	41	○		Q	○		Q	○		○		○		4 to 8	89		
倍長乗除算	D'/	○	42	○												4 to 8	90		
インクリメント	+1	○	43	○		Q	○		Q	○		○		○		2 to 3	92		
倍長インクリメント	D+1	○	44	○		Q	○		Q	○		○				2 to 3	93		
デクリメント	-1	○	45	○		Q	○		Q	○		○		○		2 to 3	94		
倍長デクリメント	D-1	○	46	○		Q	○		Q	○		○				2 to 3	95		
浮動小数点加算	F+	○	208	○			○			○						4	278		
浮動小数点減算	F-	○	209	○			○			○						4	279		
浮動小数点乗算	F*	○	210	○			○			○						4	280		
浮動小数点除算	F/	○	211	○			○			○						4	281		

T 1 はパルス化は不可です。

論理演算																			
論理積	AND	○	48	○		Q	○		Q	○		○		○		4 to 7	96		
倍長論理積	DAND	○	49	○		Q	○		Q	○		○				4 to 9	97		
論理和	OR	○	50	○		Q	○		Q	○		○		○		4 to 7	98		
倍長論理和	DOR	○	51	○		Q	○		Q	○		○				4 to 9	99		
排他的論理和	EOR	○	52	○		Q	○		Q	○		○		○		4 to 7	100		
倍長排他的論理和	DEOR	○	53	○		Q	○		Q	○		○				4 to 9	101		
否定排他的論理和	ENR	○	54	○		Q	○		Q	○		○				4 to 7	102		
倍長否定排他的論理和	DENR	○	55	○		Q	○		Q	○		○				4 to 9	103		
テーブル論理積	TAND	○	57	○			○			○		○				5	106		
テーブル論理和	TOR	○	58	○			○			○		○				5	107		
テーブル排他的論理和	TEOR	○	59	○			○			○		○				5	108		
テーブル否定排他的論理和	TENR	○	60	○			○			○		○				5	109		
テスト	TEST	○	64	○		Q	○		Q	○		○		○		3 to 5	114		
倍長テスト	DTST		65	○		Q	○		Q	○		○				3 to 7	115		
ビットファイル・ビットテスト	TTST		66	○			○			○		○				4 to 5	116		

T 1 はパルス化は不可です。

命令語	記号	○	FUN 番号	機種 ^{#1}					ステップ	ページ
				T3H	T3	T2N/T2E	T2	T1S/T1		

パルス化命令
(○:可)

*1:命令
インデックス
桁指定

(○:可) 但し△:T1Sのみ可
(I:可)
(Q:可)

シフト																				
1ビット右シフト	SHR1	○	68	○	I	Q	○	I	Q	○			○	I		○	I		2 to 3	117
1ビット左シフト	SHL1	○	69	○	I	Q	○	I	Q	○			○	I		○	I		2 to 3	118
nビット右シフト	SHR	○	70	○	I	Q	○	I	Q	○			○	I		○	I		4 to 6	119
nビット左シフト	SHL	○	71	○	I	Q	○	I	Q	○			○	I		○	I		4 to 6	121
mビットファイルnビット右シフト	TSHR	○	72	○	I		○	I		○			○	I					4 to 5	123
mビットファイルnビット左シフト	TSHL	○	73	○	I		○	I		○			○	I					4 to 5	125
シフトレジスタ	SR		74	○	I		○	I		○			○	I		○			3	127
双方向シフトレジスタ	DSR		75	○	I		○	I		○			○	I		○			3	128
デバイスシフト	SFT	○	76	○	I		○	I		○			○	I					2	130

T1はパルス化は不可です。

ローテート												
1ビット右ローテート	RTR1	○	78	○		Q	○	I	Q	○	2 to 3	131
1ビット左ローテート	RTL1	○	79	○		Q	○	I	Q	○	2 to 3	132
nビット右ローテート	RTR	○	80	○		Q	○	I	Q	○	4 to 6	133
nビット左ローテート	RTL	○	81	○		Q	○	I	Q	○	4 to 6	134
mビットファイル nビット右ローテート	TRTR	○	82	○			○	I		○	4 to 5	135
mビットファイル nビット左ローテート	TRTL	○	83	○			○	I		○	4 to 5	137
キャリー付き 1ビット右ローテート	RR1	○	84	○	I	Q	○	I	Q	○	2 to 3	139
キャリー付き 1ビット左ローテート	RL1	○	85	○	I	Q	○	I	Q	○	2 to 3	140
キャリー付き nビット右ローテート	RR	○	86	○	I	Q	○	I	Q	○	4 to 6	141
キャリー付き nビット左ローテート	RL	○	87	○	I	Q	○	I	Q	○	4 to 6	142
キャリー付きmビットファイル nビット右ローテート	TRRC	○	88	○	I		○	I		○	4 to 5	143
キャリー付きmビットファイル nビット左ローテート	TRL	○	89	○	I		○	I		○	4 to 5	145
マルチプレクサ	MPX	○	90	○	I		○	I		○	5 to 6	147
デマルチプレクサ	DPX	○	91	○	I		○	I		○	5 to 6	148
テーブル→ビット転送	TBM	○	92	○	I		○	I		○	5 to 6	149
ビット→テーブル転送	BTM	○	93	○	I		○	I		○	5 to 6	150

T1はパルス化は不可です。

パルス化命令 (○:可)

*1: 命令 (○:可) 但し△:T1Sのみ可
 インデックス (|:可)
 桁指定 (Q:可)

データ処理																			
デバイス／レジスタセット	SET	○	114	○	┃	Q	○	┃	Q	○	┃		○	┃		○		2 to 3	170
デバイス／レジスタリセット	RST	○	115	○	┃	Q	○	┃	Q	○	┃		○	┃		○		2 to 3	172
テーブルビットセット	TSET	○	116	○	┃		○	┃		○	┃		○	┃				4 to 5	174
テーブルビットリセット	TRST	○	117	○	┃		○	┃		○	┃		○	┃				4 to 5	176
キャリーセット	SETC		118	○			○			○			○			○		1	178
キャリーリセット	RSTC		119	○			○			○			○			○		1	179
エンコード	ENC	○	120	○	┃		○	┃		○	┃		○	┃		○		3 to 4	180
デコード	DEC	○	121	○	┃		○	┃		○	┃		○	┃		○		3 to 4	181
データビットカウンタ	BC	○	122	○	┃	Q	○	┃	Q	○	┃		○	┃		○		3 to 5	182
倍長ビットカウンタ	DBC	○	123	○	┃	Q	○	┃	Q	○	┃		○	┃		△		3 to 6	183
データサーチ	SCH	○	124	○	┃		○	┃		○	┃		○	┃				5 to 6	184
プッシュ	PUSH	○	125	○	┃	Q	○	┃	Q	○	┃		○	┃				5 to 6	185
ポップラスト	POPL	○	126	○	┃	Q	○	┃	Q	○	┃		○	┃				5 to 6	187
ポップファースト	POPF	○	127	○	┃	Q	○	┃	Q	○	┃		○	┃				5 to 6	188
フリップフロップ	F / F		147	○	┃		○	┃		○	┃		○	┃		○		2	202
アップダウンカウンタ	U / D		149	○	┃		○	┃		○	┃		○	┃		○		2	204

- 34 -

命令語	記号	○	FUN 番号	機種 ^{※1}					ステップ	ページ
				T3H	T3	T2N/T2E	T2	T1S/T1		

パルス化命令
(○:可)

*1:命令
インデックス
桁指定
(○:可) 但し△:T1Sのみ可
(|:可)
(Q:可)

P R G制御																				
サブルーチンコール	CALL	128	○			○			○			○			○			2 to 3	189	
サブルーチンリターン	RET	129	○			○			○			○			○			1	190	
条件ジャンプ	JUMP	130	○			○			○			○						2 to 3	191	
ジャンプラベル	LBL	136	○			○			○			○						2	191	
繰り返しFOR	FOR	132	○			○			○			○			○			2	192	
繰り返しNEXT	NEXT	133	○			○			○			○			○			1	192	
サブルーチンエントリー	SUBR	137	○			○			○			○			○			2	194	
割り込みプログラム許可	EI	140	○			○			○			○			○			1	195	
割り込みプログラム禁止	DI	141	○			○			○			○			○			1	196	
割り込みプログラム終了	IRET	142	○			○			○			○			○			1	197	
ウォッチドッグタイマリセット	WDT	143	○			○			○			○			○			2	198	
ステップシーケンスイニシャライズ	STIZ	144	○						○			○			○			3	199	
ステップシーケンス入力	STIN	145	○			○			○			○			○			2	200	
ステップシーケンス出力	STOT	146	○			○			○			○			○			2	201	
S F Cイニシャライズ	SFIZ	241	○			○			○			○						3	334	

R A S																				
診断表示	DIAG		150	○			○			○			○						3 to 4	205
診断表示リセット	DIAR		151	○			○			○			○						2 to 3	207
ステータスラッチセット	STLS		152	○			○			○			○						1	209
ステータスラッチリセット	STLR		153	○			○			○			○						1	211
カレンダー設定	CLND	○	154	○			○			○			○			△			2	212
カレンダー演算	CLDS	○	155	○			○			○			○			△			3	213
ドラムシーケンサ	DRUM		158	○			○			○			○						6	220
カムシーケンサ	CAM		159	○			○			○			○						5	222

命令語	記号	○	FUN 番号	機種 ^{*1}					ステップ	ページ
				T3H	T3	T2N/T2E	T2	T1S/T1		

パルス化命令
(○:可)

*1:命令
インデックス
桁指定
(○:可) 但し△:T1Sのみ可
(|:可)
(Q:可)

関数										
移動平均	MAVE	○	56					○		△
デジタル フィルタ	DFL	○	61					○		△
上限リミット	UL	○	160	○		Q	○		Q	△
下限リミット	LL	○	161	○		Q	○		Q	△
最大値	MAX	○	162	○			○			△
最小値	MIN	○	163	○			○			△
平均値	AVE	○	164	○			○			△
関数発生器	FG	○	165	○		Q	○		Q	○
デッドバンド	DB	○	166	○		Q	○		Q	○
開平	RT	○	167	○		Q	○		Q	○
積分	INTG	○	168	○		Q	○		Q	○
ランプ関数	RAMP	○	169	○		Q	○		Q	○
速度型PID	PID		170	○			○			○
偏差2乗型PID	PID2		171	○			○			○
本質継承型PID	PID3	○	156	○			○			○
正弦関数	SIN	○	172	○		Q	○		Q	○
余弦関数	COS	○	173	○		Q	○		Q	○
正接関数	TAN	○	174	○		Q	○		Q	○
逆正接関数	ASIN	○	175	○		Q	○		Q	○
逆余弦関数	ACOS	○	176	○		Q	○		Q	○
逆正接関数	ATAN	○	177	○		Q	○		Q	○
指数関数	EXP	○	178	○		Q	○		Q	○
常用対数関数	LOG	○	179	○		Q	○		Q	○

T1はパルス化は不可です。

データ変換										
HEX → ASCII 変換	HTOA	○	62					○		△
ASCII → HEX 変換	ATOH	○	63					○		△
絶対値	ABS	○	180	○		Q	○		Q	○
倍長絶対値	DABS	○	181	○		Q	○		Q	○
2の補数	NEG	○	182	○		Q	○		Q	○
倍長2の補数	DNEG	○	183	○		Q	○		Q	○
ダブルワード変換	DW	○	184	○		Q	○		Q	○
7セグメントデコード	7SEG	○	185	○		Q	○		Q	○
ASCII 変換	ASC	○	186	○			○			○
バイナリ変換	BIN	○	188	○		Q	○		Q	○
倍長バイナリ変換	DBIN	○	189	○		Q	○		Q	○
BCD変換	BCD	○	190	○		Q	○		Q	○
倍長BCD変換	DBCD	○	191	○		Q	○		Q	○
浮動小数点変換	FLT	○	204	○			○			○
固定小数点変換	FIX	○	205	○			○			○
浮動小数点絶対値	FABS	○	206	○						○
浮動小数点符号反転	FNEG	○	207	○						○

T1はパルス化は不可です。

命令語	記号	○	FUN 番号	機種 ^{*1}					ステップ	ページ
				T3H	T3	T2N/T2E	T2	T1S/T1		

パルス化命令
(○:可)

*1:命令 (○:可) 但し△:T1Sのみ可
インデックス (|:可)
桁指定 (Q:可)
*2:T2Nのみ可

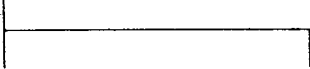
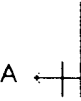
B C D 演算																						
B C D 加算	B+	○	192	○		Q	○		Q	○			○								4 to 7	262
B C D 減算	B-	○	193	○		Q	○		Q	○			○								4 to 7	263
B C D 乗算	B*	○	194	○		Q	○		Q	○			○								4 to 7	264
B C D 除算	B/	○	195	○		Q	○		Q	○			○								4 to 7	265
倍長 B C D 加算	DB+	○	196	○		Q	○		Q	○			○								4 to 9	266
倍長 B C D 減算	DB-	○	197	○		Q	○		Q	○			○								4 to 9	267
倍長 B C D 乗算	DB*	○	198	○		Q	○		Q	○			○								4 to 9	268
倍長 B C D 除算	DB/	○	199	○		Q	○		Q	○			○								4 to 9	269
キャリー付き B C D 加算	B+C	○	200	○		Q	○		Q	○			○								4 to 7	270
キャリー付き B C D 減算	B-C	○	201	○		Q	○		Q	○			○								4 to 7	271
キャリー付き倍長 B C D 加算	DB+C	○	202	○		Q	○		Q	○			○								4 to 9	272
キャリー付き倍長 B C D 減算	DB-C	○	203	○		Q	○		Q	○			○								4 to 9	273

実数関数																							
浮動小数点 上限リミット	FUL	○	218	○			○														4	288	
浮動小数点 下限リミット	FLL	○	219	○			○														4	289	
浮動小数点 デッドバンド	FDB	○	220	○			○														4	290	
浮動小数点 開平	FRT	○	221	○			○														3	291	
浮動小数点 速度型PID	FPID	○	222	○			○														4	292	
浮動小数点 偏差2乗型PID	FPID2	○	223	○			○														4	294	
浮動小数点 本質継承型PID	FPID3	○	232	○																	4	304	
浮動小数点 正弦関数	FSIN	○	224	○			○														3	296	
浮動小数点 余弦関数	FCOS	○	225	○			○														3	297	
浮動小数点 正接関数	FTAN	○	226	○			○														3	298	
浮動小数点 逆正弦関数	FASIN	○	227	○			○														3	299	
浮動小数点 逆余弦関数	FACOS	○	228	○			○														3	300	
浮動小数点 逆正接関数	FATAN	○	229	○			○														3	301	
浮動小数点 指数関数	FEXP	○	230	○			○														3	302	
浮動小数点 常用対数関数	FLOG	○	231	○			○														3	303	

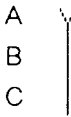
入出力																							
直接入出力	I/O	○	235	○			○			○			○			○			○			3	309
拡張データ転送	XFER	○	236	○			○			○			○			○			○			4	310
特殊モジュールデータ入力	READ	○	237	○			○			○			○			○			○			4 to 5	318
特殊モジュールデータ出力	WRITE	○	238	○			○			○			○			○			○			4 to 5	320
メッセージ伝送命令 (送信)	SEND	○	239	○						○ ^{*2}													322
メッセージ伝送命令 (受信)	RECV	○	240	○						○ ^{*2}													328

T1はパルス化は不可です。

9. 2 S F C (T1/T1Sは使用不可)

シンボル名称	シンボル	ステップ数	詳細部	ページ
イニシャル ステップ	A 	A: ステップNO. (0~4095) 2	あり	335
エンドステップ	A 	A: ステップNO. (0~4095) ※イニシャルステップと 同一であること。 2	あり	336
ステップ	A 	A: ステップNO. (0~4095) 1	あり	337
待ち時間付き ステップ	A  B C	A: ステップNO. (0~4095) B: タイマNO. (0~511) C: 設定時間 (タイマT0~T63 × 10ms) (T64~T511×100ms) 4	あり	338
波滯検出付き ステップ	A  B C D	A: ステップNO. (0~4095) B: タイマNO. (0~511) C: 監視時間 (タイマT0~T63 × 10ms) (T64~T511×100ms) D: アラームデバイス 5	あり	339
遷移		1	あり	340
条件分岐		$2 \times n - 1$ nは 分岐列数		341
並列実行		$n + 3$ nは 分岐列数		342
ジャンプ	A 	A: ラベルNO. (0~1023) 5	あり	343

シンボル名称	シンボル	ステップ数	詳細部	ページ

ラベル		A: ラベルNO. (0~1023) ※ジャンプと同一である こと。	2		343
SFC イニシャライズ		A: プログラムNO. (0~63) B: デバイスNO. C: ステップ個数	4		344
マクロステップ		A: ステップNO. (0~4095) B: マクロNO. (0~127)	3		345
マクロラベル		A: マクロNO. (0~127)	1		346
マクロエンド			2	あり	346
エンド		A: ラベルNO. (0~1023)	2	あり	347

10. 各機種の相違点について

本書で説明されている命令語を使用する上で、T1、T1S、T2、T2E、T2N、T3、T3Hでは次の5点が異なりますので留意して下さい。

- 1 命令語のオペランドに記述できるレジスタ／デバイスの範囲が異なります。
- 2 T1／T1S、T2では浮動小数点データを扱うことはできません。
- 3 T2／T2E／T2Nでは桁指定のできる命令語に制限があります。
- 4 T1／T1Sではインデックス指定のできる命令語に制限があります。また桁指定・命令語パルス化はできません。
- 5 T1／T1Sではリンク関連のレジスタ（W、LW）、ファイルレジスタ（F）、直接入出力レジスタ（I、O、IW、OW）は使用できません。

10.1 レジスタ／デバイスの範囲

各機種のレジスタ／デバイス範囲は次の通りです。命令語のオペランドでレジスタ／デバイスを使用する際は、それぞれのレジスタ／デバイスの範囲で指定して下さい。

種 別	レジスタ／デバイス範囲							
		T1	T1S	T2	T2E	T2N	T3	T3H
X／Y	0000～	031F	031F	063F	063F	127F	255F	511F
XW／YW	000～	031	031	063	063	127	255	511
R	0000～	063F	255F	127F	127F	255F	511F	999F
RW	000～	063	255	127	127	255	511	999
L	0000～	—	—	255F	255F	255F	255F	255F
LW	000～	—	—	255	255	255	255	255F
Z	0000～	—	—	511F	511F	999F	511F	999F
W	0000～	—	—	1023	1023	2047	1023	2047
T	000～	063	255	255	255	511	511	999
C	000～	063	255	255	255	511	511	511
S	0000～	063F	063F	255F	255F	255F	255F	255F
SW	000～	063	063	255	255	255	255	255
D	0000～	1023	4095	4095	4095	8191	8191	8191
F	0000～	—	—	1023	1023	1023	8191	32767*

※オペランドの記述はF9999までとなっていますが、インデックス修飾によってF32767まで使用できます

10.2 桁指定

T2／T2E／T2Nで桁指定のできる命令語はデータ転送（FUN 18: MOV）とデータ交換（FUN 22: XCHG）の2命令語のみです。他の命令語で桁指定をすることはできません。なお、データ転送、データ交換で桁指定のできるオペランドはT3、T3Hと同じです。また、T1／T1Sでは桁指定を使用できません。

命令語仕様

ラダー言語	43
S F C 言語	335

a 接点

シーケンス命令	a 接点
リレーコイルの a 接点です。	関連命令 b 接点

A — —

機能

- オペランド A に相当するリレーコイルの a 接点（常時開接点）です。

入力	処 理	出力
OFF	デバイスの内容にかかわらず	OFF
ON	デバイス = 0	OFF
	デバイス = 1	ON

オペランド

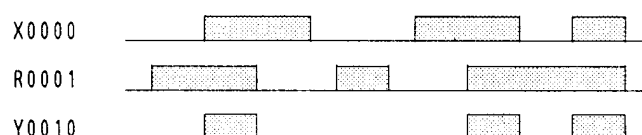
		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾	
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J
A	デバイス番号	○	○	○	○	○	○	○	○	○																	

プログラム例

X0000	R0001	Y0010
		()

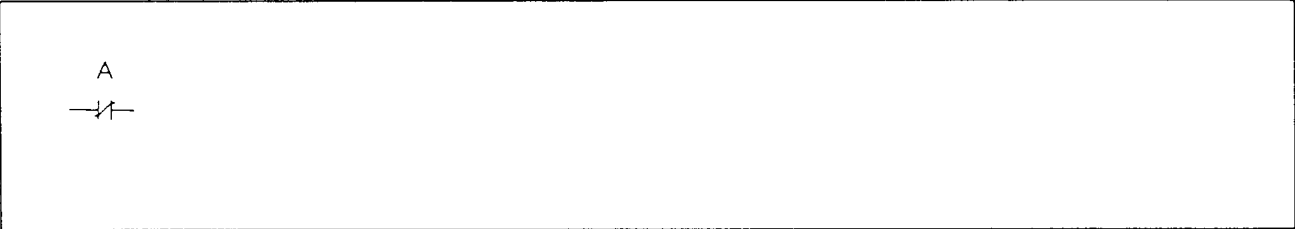
動作

- デバイス X0000、R0001 の内容が共に 1（ON）の時のみ、コイル Y0010 が ON になります。



b 接点

シーケンス命令	b 接点	
リレーコイルの b 接点です。		関連命令 a 接点



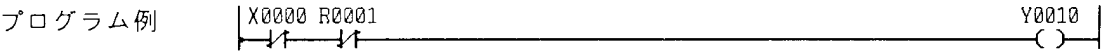
機能

- オペランド A に相当するリレーコイルの b 接点（常時閉接点）です。

入力	処 理	出力
OFF	デバイスの内容にかかわらず	OFF
O N	デバイス = 0	O N
	デバイス = 1	OFF

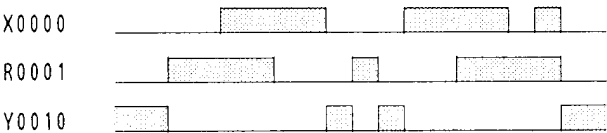
オペランド

		デ バ イ ス										レ ジ ス タ												INDEX			定 数	修 飾
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K		
A	デバイス番号	○	○	○	○	○	○	○	○	○																		



動作

- デバイス X0000、R0001 の内容が共に 0（OFF）の時のみコイル Y0010 が ON になります。



N 時微分接点

シーケンス命令 ON時微分接点

入力のOFFからONへの変化を検出します。

関連命令

—|T|—

機能

- 入力がOFFからONに切り換わると、出力を1スキanning時間ONにします。OFF→ONになる変化を取り出してパルス信号を作るのに使用します。

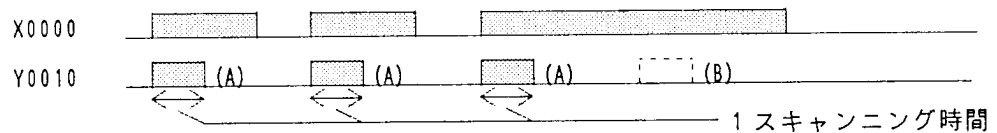
入力	処 理	出力
OFF	以前の状態に関わりなく	OFF
ON	前回スキanningの入力がOFF	ON
	前回スキanningの入力がON	OFF

プログラム例



動作

- a接点X0000がOFF→ONになった時から、1スキanning時間だけ出力をONにします。(A)
 前回スキanning時の入力がONのときや、今回の入力がOFF（a接点X0000がOFF）の時は出力はOFFになります。(B)

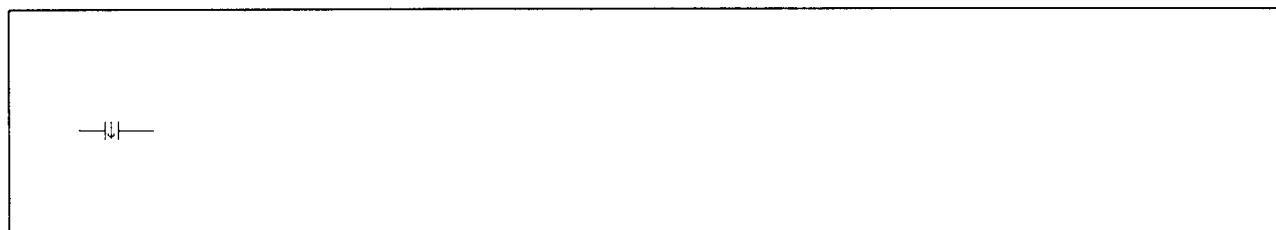


補足

T1 / T1Sでは、ON時 / OFF時微分接点は合計で512個まで使用できます。

OFF 時微分接点

シーケンス命令	OFF 時微分接点	関連命令
入力のONからOFFへの変化を検出します。		

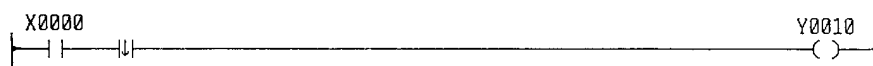


機能

- 入力がONからOFFに切り換わると、出力を1スキャンニングだけONにします。ON→OFFになる変化を取り出してパルス信号を作るのに使用します。

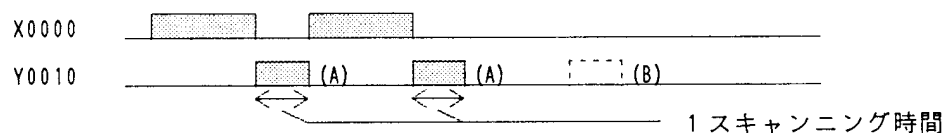
入力	処 理	出力
OFF	前回スキャンの入力がOFF	OFF
	前回スキャンの入力がON	ON
ON	以前の状態に関わりなく	OFF

プログラム例



動作

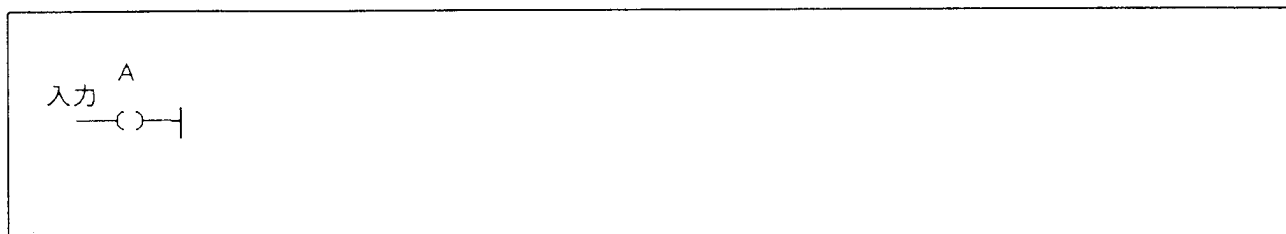
- a 接点X0000がON→OFFになった時から、1スキャンニング時間だけ出力をONにします。(A)
 前回スキャン時の入力がOFFであった場合や、今回の入力がON (a 接点X0000がON) の時は出力をOFFにします。(B)



補足

T1 / T1Sでは、ON時 / OFF時微分接点は合計で512個まで使用できます。

シーケンス命令	コイル
デバイスAのリレーコイルです。	関連命令



機能

- オペランドAで示されるリレーコイルです。

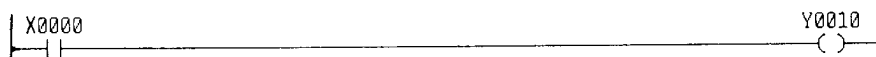
入力	処 理	出力
OFF	デバイスAに0を格納	
ON	デバイスAに1を格納	

オペランド

オペランド

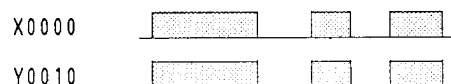
		デバイス										レジスタ										INDEX			定数	修飾			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	デバイス番号		○	○	○	○	○				○																		

プログラム例



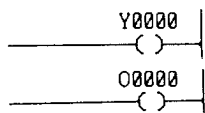
動作

- a接点X0000がONするとデバイスY0010に1が格納されます。OFFすると0が格納されます。



補足

オペランドAには、次のような表記が可能です。

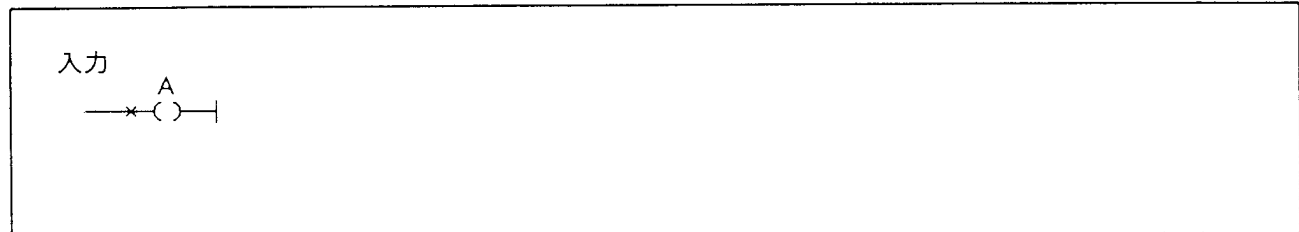


… デバイスY0000に格納。

… デバイスY0000を格納。YW000データを対応する各モジュールに出力。Y0000以外の出力ビットは変化しません。

フォーストコイル

シーケンス命令	フォーストコイル	関連命令
出力を保持するコイルです。		



機能

- ・入力がONであってもOFFであっても、前の状態を保持するコイルです。
- ・複数のコイルにデバイスAが使用されている場合は、フォーストコイルとしての動作は保証されません。

入力	処 理	出力
OFF	前の状態を保持	
ON	前の状態を保持	

オペランド

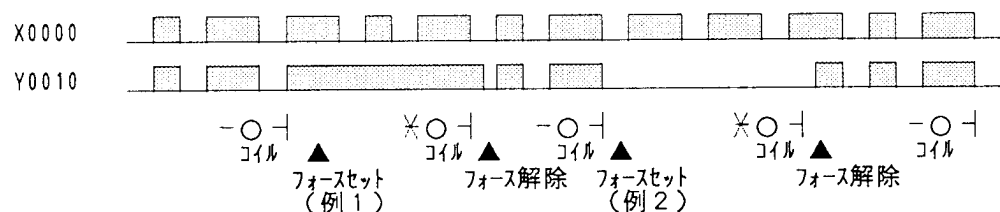
		デ バ イ ス										レ ジ ス タ										INDEX			定	修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
A	デバイス番号		○		○	○	○																						

プログラム例



動作

- ・フォースされたコイル（フォーストコイル）は、a 接点X0000のON／OFF状態にかかわらずフォーストされる直前のデバイスY0010の内容を保持します。フォースセットされる。直前のY0010の内容がONであればフォーストコイルY0010はONを保持します（例1）直前のY0010の内容がOFFであればフォーストコイルY0010はOFFを保持します。（例2）



インバータ

シーケンス命令	インバータ	関連命令
入力状態を反転して出力します。		

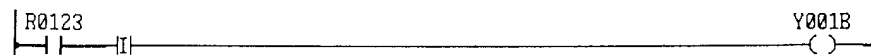
入力	出力
— I —	

機能

- ・入力状態がOFFの時はONに、ONの時はOFFにします。

入力	処 理	出力
OFF	反転	ON
ON	反転	OFF

プログラム例



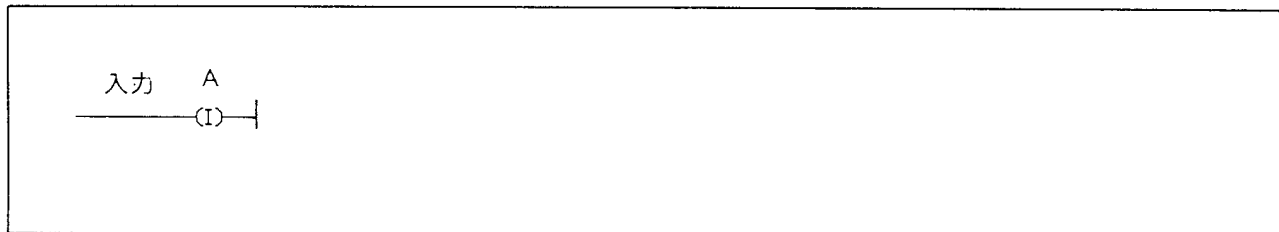
動作

- ・a接点R0123がOFFの時、Y001BはONになります。a接点R0123がONの時、Y001BはOFFになります。



インバートコイル

シーケンス命令	インバートコイル	
演算結果を反転して出力します。		関連命令



機能

- ・入力がONの時、オペランドAにOFFを格納します。
- ・入力がOFFの時、デバイスAにONを格納します。

入力	処 理	出力
OFF	デバイスAに1を格納	/
ON	デバイスAに0を格納	/

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定	修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
A	デバイス番号		○	○	○	○	○				○																		



動作

- ・a 接点X0003がONの時、デバイスR023EにOFFを格納します。
- ・a 接点X0003がOFFの時、デバイスR023EにONを格納します。



ポジティブパルス接点

シーケンス命令	ポジティブパルス接点	関連命令
	デバイスのOFFからONへの変化を検出します。	

入力 A 出力
—P—

機能

- 入力がONの時、デバイスAがOFF→ONへと変化すると、出力を1スキャンニング時間だけONにします。

入力	処 理	出力
OFF	不実行	OFF
ON	A : OFF	OFF
	A : ON→OFF	OFF
	A : OFF→ON	ON
	A : ON	OFF

オペランド

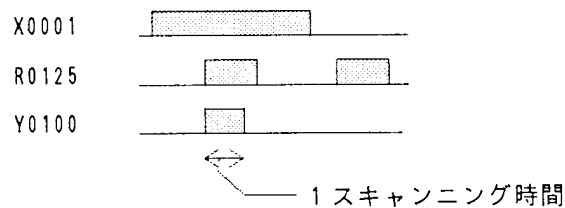
		デ バ イ ス										レ ジ ス タ										INDEX			定	修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
A	デバイス番号	○	○	○	○	○	○	○	○	○																			

プログラム例



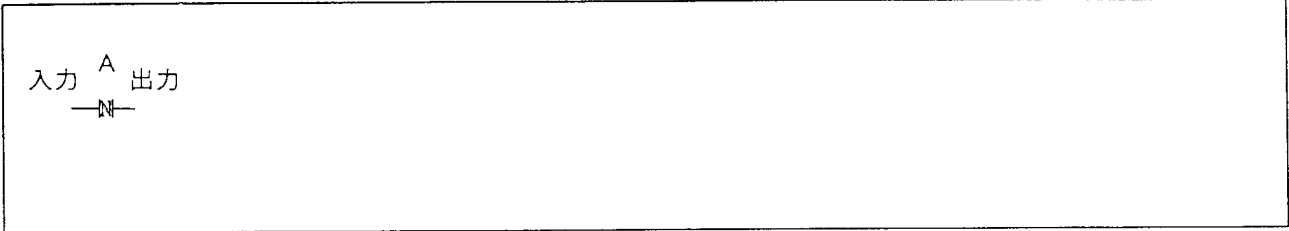
動作

- 入力条件接点X0001がONで、デバイスR0125がOFF→ONに切り換わった時、出力を1スキャンニング時間ONにします。それ以外の状態では、出力はOFFになります。



ネガティブパルス接点

シーケンス命令	ネガティブパルス接点	関連命令
デバイスのONからOFFへの変化を検出します。		



機能

- 入力がONの時、デバイスAがON→OFFへ変化すると、出力を1スキャンニング時間だけONにします。

入力	処 理	出力
OFF	不実行	OFF
ON	A : ON	OFF
	A : OFF→ON	OFF
	A : ON→OFF	ON
	A : OFF	OFF

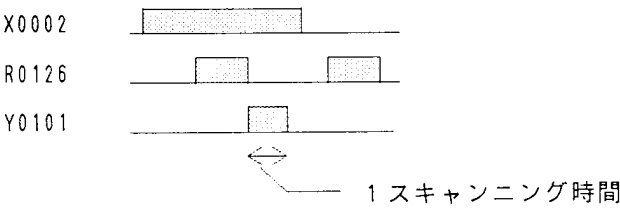
オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定	修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
A	デバイス番号	○	○	○	○	○	○	○	○	○																			



動作

- a接点X0002がONでデバイスR0126がON→OFFの時のみ出力Y0101を1スキャンニング時間ONにします。それ以外の状態では、出力はOFFになります。

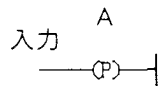


ジ テ ィ ブ パ ル ス コ イ ル

シーケンス命令 | ポジティブパルスコイル

入力のOFF→ON変化により出力するコイルです。

関連命令



機能

- ・入力がOFF→ONに切り換わった時、出力デバイスを1スキャンニング時間ONにします。

入力	処 理	出力
OFF	デバイスAに0を格納	/
ON	前回スキャン時の入力がONの時 →デバイスAに0を格納	/
	前回スキャン時の入力がOFFの時 →デバイスAに1を格納	/

オペランド

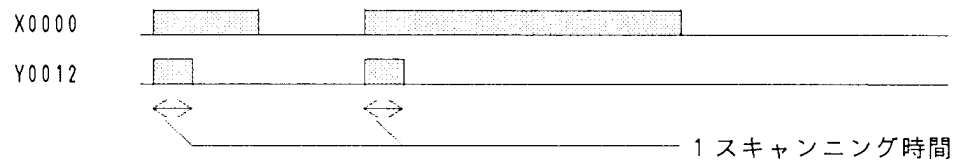
		デバイス										レジスタ										INDEX			定数	修飾			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	デバイス番号		○	○	○	○	○				○																		

プログラム例

X0000	Y0012
	(P)

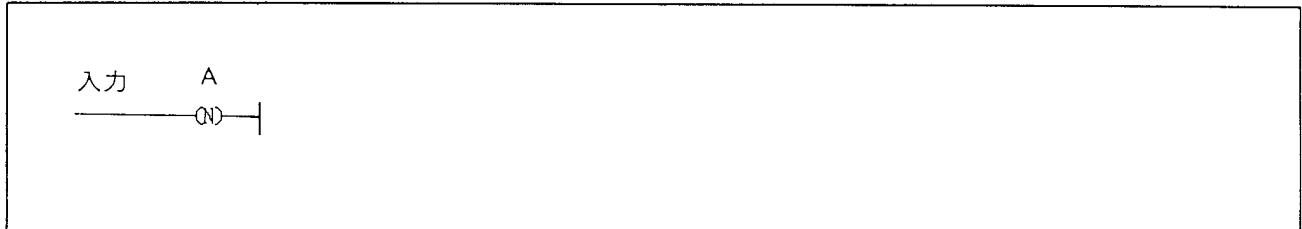
動作

- ・a接点X0000がOFF→ONに切り換わった時、デバイスY0012を1スキャンニング時間ONにします。それ以外の状態では、出力はOFFになります。



ネガティブパルスコイル

シーケンス命令	ネガティブパルスコイル
入力のON→OFF変化により出力するコイルです。	関連命令



機能

- 入力がON→OFFに切り換わった時、出力デバイスを1スキャン時間ONにします。

入力	処 理	出力
OFF	前回スキャン時の入力がOFFの時 →デバイスAに0を格納	/
	前回スキャン時の入力がONの時 →デバイスAに1を格納	/
ON	デバイスAに0を格納	/

オペランド

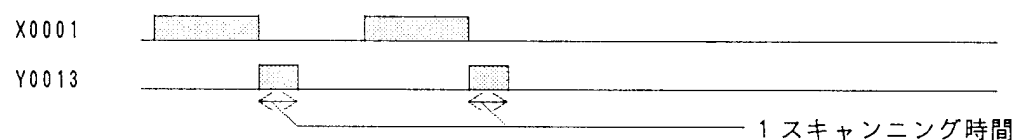
		デバイス										レジスタ										INDEX			定数	修飾	
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J
A	デバイス番号		○	○	○	○	○				○																

プログラム例



動作

- a 接点X0001がON→OFFに切り換わった時、デバイスY0013を1スキャン時間ONにします。それ以外の状態では、出力はOFFになります。



ジャンプコントロールセット・リセット

シーケンス命令	ジャンプコントロールセット・リセット	関連命令
指定回路をジャンプしてプログラムを実行します。		

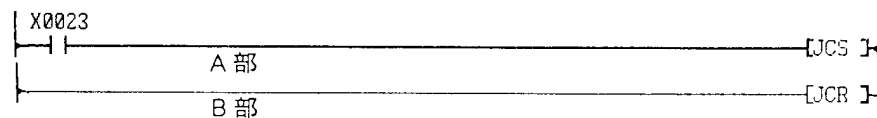
条件入力

機能

- ・ JCS 命令の入力条件が ON になった時、JCS～JCR 間の命令は高速読み飛ばしを行い、JCR 以降の命令から実行を始めます。
- ・ JCS、JCR は必ずペアで使用します。
- ・ JCS と JCR の順序が逆になるとエラーになります。
- ・ JCS は連続で 2 回以上プログラミングできません。

入力	処 理	出力
OFF	不実行	
ON	ジャンプ	

プログラム例



動作

- ・ a 接点 X0023 が ON の時、A 部の命令は実行せず B 部の命令まで高速読み飛ばしを行い、B 部の命令から実行します。
- ・ a 接点 X0023 が OFF の時、JCS 命令、JCR 命令を無視して A 部、B 部の命令を実行します。

終了

シーケンス命令	終了
プログラムの終了命令です。	関連命令

—END —

機能

- ・メインプログラム、サブプログラムの終了命令です。
- ・END命令以降にプログラムを追加してもかまいませんが実行はされません。ただし、使用ステップ数には含まれますので注意してください。
- ・END命令は、使用しているプログラムの中に必ず1個は入れて下さい。

プログラム例

—END —

[illegible]

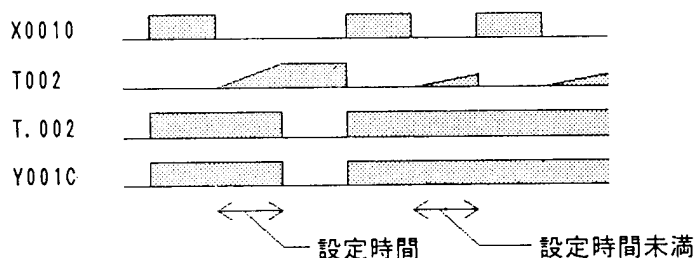
条件入力	タイマ出力
—[A TOF B]—	

入力値がOFFの時、設定時間経過後に出力をOFFにします。
経過時間（現在値）は加算方式です。
オペランドAの定数範囲は0～32767です。
オペランドAで定数を指定した場合は、その定数にはインデック
修飾できません。

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	設定時間											○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
B	現在値																		○									

X0010 00100 00088 Y001C
 D1234 TOF T002

- ・ a 接点X0010がON→OFFになってからレジスタD1234のデータ00100（設定時間1秒）が経過するとT0Fの出力（タイマデバイスT.002）がOFFになります。
経過時間（現在値）はT002に格納されています。
- ・ a 接点X0010がOFF→ONになると、レジスタT002のデータを0にし、タイマデバイスT.002及びコイルY001CはONになります。
- ・ 現在値が設定時間に達した後の現在値内容は入力（a 接点X0010）がOFFになっている限り設定時間の内容を保持しています。



T 1 S / T 2 / T 2 E / T 2 N / T 3 : T000～T063=0.01秒単位
T064～最終=0.1 秒単位

T 3 H : 0.01秒単位のタイマの範囲は設定可能です。
初期設定では T000～T063=0.01秒単位
T064～T999=0.1 秒単位

T 1 : T001～T031=0.01秒単位
T032～T063=0.1 秒単位

シングル ショット

シーケンス命令 シングル ショット (SS)

設定時間の間、出力をONにします。

関連命令

条件入力

タイマ出力

—[A SS B]—

機能

- ・入力ONの時、設定時間が経過するまで出力をONにします。
- ・経過時間(現在値)は加算方式です。
- ・オペランドAの定数範囲は0~32767です。
- ・オペランドAで定数を指定した場合は、その定数にはインデックス修飾できません。

入力	処 理		出力
OFF	不実行		OFF
ON	実行	現在値<設定時間	ON
		現在値=設定時間	OFF

オペランド

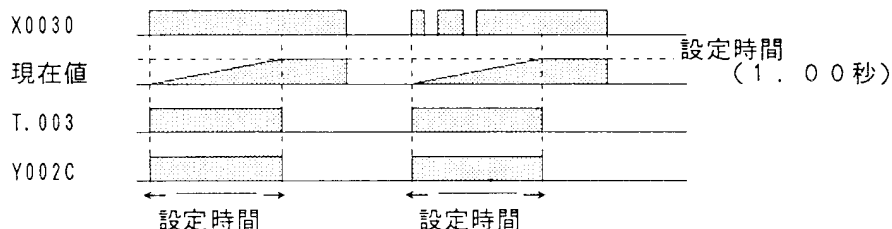
		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	設定時間											○	○	○	○	○	○	○	○	○	○	○			○	○	○	○
B	現在値																		○									

プログラム例

X0030 00050 Y001A
| | [00100 SS T003] ()

動作

- ・a接点X0030がOFF→ONになってからオペランドAで指定されたデータ00100(設定時間1秒)が経過するまでSSの出力(タイマデバイスT.003)及びコイルY001AをONにします。なお、経過時間(現在値)はT003に格納されています。
- ・現在値が設定時間に達する前に、入力(a接点X0030)がOFFになっても現在値は引き続き更新されます。
- ・現在値が設定時間に達した後の現在値内容は入力(a接点X0030)がONになっている限り設定値の内容を保持しています。また、T.003およびY001Aの内容は、現在値=設定値となった時点でOFFとなります。そして、入力(a接点X0030)がOFFになれば現在値T003の内容を0にします。



補足

T1S / T2 / T2E / T2N / T3 : T000~T063=0.01秒単位

T064~最終=0.1 秒単位

T3H

: 0.01秒単位のタイマの範囲は設定可能です。

初期設定では T000~T063=0.01秒単位

T064~T999=0.1 秒単位

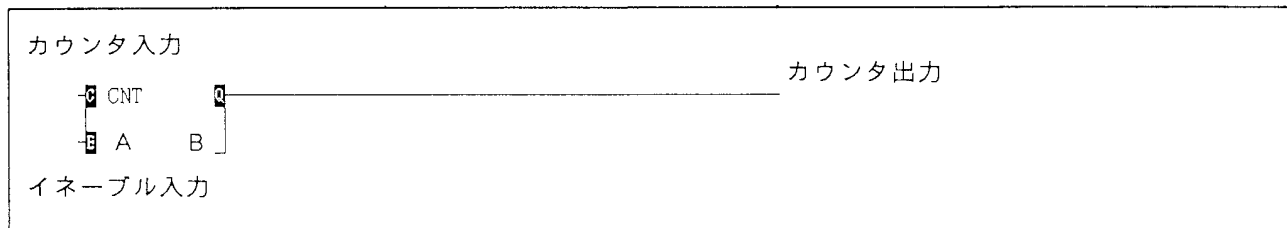
T1

: T001~T031=0.01秒単位

T032~T063=0.1 秒単位

カウンタ

シーケンス命令	カウンタ (CNT)
OFF→ONで動作するアップカウンタです。	
関連命令	



機能

- イネーブル入力がある時、カウンタ入力のOFF→ONの変化回数をカウントしカウント値Bが設定値Aに達すると、出力をONにします。以後イネーブル入力があるうちはカウント入力があっても、カウント値が更新されず、出力はONのままです。現在値は加算方式です。
- オペランドAの定数範囲は0～65535です。
- オペランドAで定数を指定した場合は、その定数にはインデックス修飾できません。

入力	処 理	出力
E=OFF	不実行(カウント値7)	OFF
E=ON	実行	
	カウント値<設定値	OFF
	カウント値≥設定値	ON

オペランド

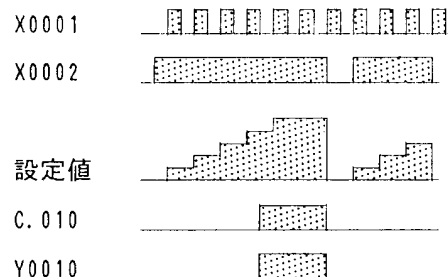
		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	設定時間											○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
B	現在値																		○										

プログラム例



動作

- a 接点X0001がOFF→ONの変化回数をカウントします。カウント値はオペランドBで示されるC010に格納されます。カウント値がオペランドAで示される設定値00005に達するとCNTの出力(カウントデバイスC.010)およびコイルY0010がONになります。
- a 接点X0002 (イネーブル入力) がOFFになるとカウント値の内容を0にし、出力をOFFにします。(カウント中にイネーブル入力がある時も同様です)



補足

- カウンタ命令の前に微分接点をおく必要はありません。(入力のOFFからONへの立ち上がりはCNTが検出します。)

マスターコントロールセット／リセット

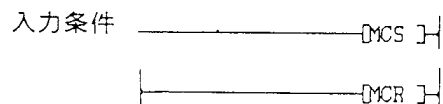
シーケンス命令 マスターコントロールセット／リセット (MCS/MCR)

指定回路間の母線入力をON/OFFします。

関連命令

マスターコントロールセット n

マスターコントロールリセット n

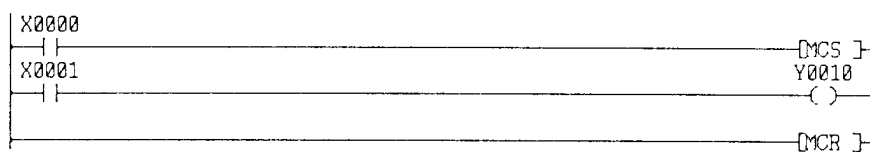


機能

- ・MCSの入力がOFFの時MCRまでの母線入力をOFFにします。
- ・MCSの入力がONの時は通常動作を行います。

入力	処 理	出力
OFF	母線入力 OFF	
ON	通常動作	

プログラム例



動作

- ・a 接点X0000がONの時は通常動作を行います。つまりX0001がONの時はY0010がONとなり、X0001がOFFの時はY0010はOFFになります。
- ・a 接点X0000がOFFの時はMCSの入力がOFFになるので、MCS-MCR間の母線入力がOFFとなり、X0001がONであってもOFFであってもY0010はOFFとなります。

補足

- ・MCS、MCRは必ずペアで使用する必要があります。
- ・本命令ではネスティングすることはできません。ネスティングが必要なときには、マスターコントロールセット n、マスターコントロールリセット n (FUN 134, 135) を使用して下さい。

データ転送

FUN18 データ転送 (MOV) *

レジスタデータを他のレジスタに転送します。

関連命令

条件入力

実行出力

← [A MOV B] →

機能

- ・オペランドAの内容をオペランドBに転送します。
- ・オペランドAの定数範囲は-32768～32767です。
- ・レジスタ間転送時は高速動作を行います。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

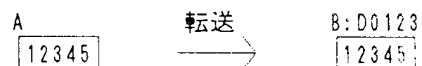
オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	転送元	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
B	転送先		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○	○	○	○

プログラム例

R002F 12345


- 動作
- ・a接点R002FがONの時、オペランドAの定数(12345)をレジスタD0123に格納し、出力をONにします。



データ否定転送

FUN2 0 データ否定転送 (NOT) *

データのビットを反転して他のレジスタに転送します。

関連命令

条件入力

実行出力

[A NOT B]

機能

- ・オペランドAの16ビットデータを反転しデータBに転送します。
- ・オペランドの定数範囲は-32768～32767です。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定	修				
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾		
A	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○	○
B	反転結果		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○	○			○

プログラム例

```
Z0023 H4321 HBCDE
[ RW002 NOT RW041 ]
```

動作

- ・a 接点Z0023がONの時、オペランドRW002のデータH4321を全ビット反転させて結果HBCDEをRW041に格納し出力をONにします。

		F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	
A: XW023		0	1	0	0	0	0	1	1	0	0	1	0	0	0	0	1	(H4321)
		↓ 全ビットの反転																
B: RW041		1	0	1	1	1	1	0	0	1	1	0	1	1	1	1	0	(HBCDE)

倍長データ否定転送

FUN2 1 倍長データ否定転送 (DNOT) *

倍長データのビットを反転し、他の倍長レジスタに転送します。

関連命令

条件入力

実行出力

$\neg [A+1 \cdot A \text{ DNOT } B+1 \cdot B]$

機能

- オペランド $A+1 \cdot A$ の32ビットデータを反転し、オペランド $B+1 \cdot B$ に転送します。
- オペランドの定数範囲は-2147483648~2147483647です。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デバイス										レジスタ										INDEX			定数	修飾			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A+1・A	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○		○	○
B+1・B	反転結果		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○			○

プログラム例

R0033 H13579BDF HECA86420
 $\neg [RW031 \cdot RW030 \text{ DNOT } D2344 \cdot D2343]$

動作

- a 接点R0033がONの時、RW031・RW030のデータH13579BDFを全ビット反転し、レジスタD2344・D2343に演算結果HECA86420を格納し出力をONにします。

A+1:RW031 A:RW030
 上位ワード 下位ワード (H13579BDF)

↓ 全ビットの反転

B+1:D2344 B:D2343
 上位ワード 下位ワード (HECA86420)

UN22 データ交換 (XCHG) ※

2つのレジスタのデータを相互に交換します。

関連命令

実行出力

条件入力	実行出力
[A XCHG B]	

機能

- ・入力条件がONの時、オペランドA及びBの16ビットデータをそれぞれ交換し、出力をONにします。
- ・入力条件がOFFの時は、交換を行わずに出力をOFFにします。
- ・オペランドA、Bのデータ範囲は-32768～32767です。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

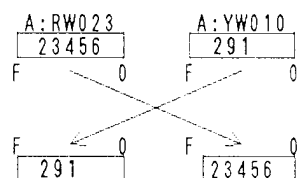
オペランド

		デ バ イ ス										レ ジ ス タ														INDEX		定数	修飾
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	J	K				
A	変換データ		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○	○		○
B	変換データ		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○	○		○

プログラム例

動作

- ・ a 接点R0044がONの時、レジスタRW023のデータ23456をレジスタYW010に、レジスタYW010のデータ(291)をRW023にそれぞれ格納し出力をONにします。



© 2006 The Authors
Journal compilation © 2006 Blackwell Publishing Ltd

FUN2 3 倍長データ交換 (DXCH) *

2つの倍長レジスタを相互に交換します。

関連命令

条件入力

実行出力

$$- [A+1 \cdot A \quad DXCH \quad B+1 \cdot B]$$

機能

- 入力条件が ON の時、オペランド A と B の倍長データをそれぞれ交換し、出力を ON にします。
- 入力条件が OFF の時は、交換を行わず出力を OFF にします。
- オペランド A + 1 ・ A、B + 1 ・ B のデータ範囲は -2147483648 ~ 2147483647 です。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

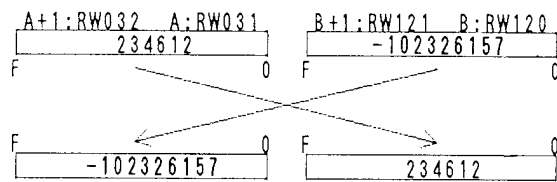
		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A+1・A	変換データ		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○		○	○			○
B+1・B	変換データ		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○		○	○			○

プログラム例 R0055 0000234612 -0102326157

R0055 0000234612 -0102326157
 [RW032·RW031 DXCH RW121·RW120]

動作 ・ a 接点R0055がONの時、レジスタRW032・RW031のデータ(234612)をレジスタ RW121・RW120に

- ・ a 接点R0055がONの時、レジスタRW032・RW031のデータ(234612)をレジスタRW121・RW120にレジスタRW121・RW120のデータ(-102326157)をRW032・RW031にそれぞれ格納し出力をONにします。



テーブル初期化

FUN2 4 テーブル初期化 (T I N Z) *

連続したレジスタを一括初期化します。

関連命令

条件入力

実行出力

← [A TINZ(n) B]

機能

- ・入力条件が O N の時、指定したレジスタ以降のテーブルサイズ n のレジスタを、全てオペランド A のデータにイニシャライズします。
- ・入力条件が O F F の時は、実行せずに出力を O F F にします。
- ・テーブルサイズ (オペランド n) の定数範囲は、1~1024 です。
- ・オペランド A の定数範囲は-32768~32767 です。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

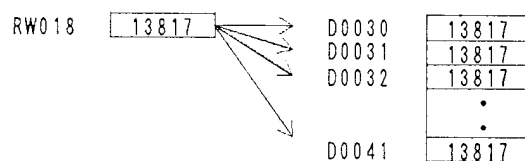
		デバイス										レジスタ										INDEX			定数	修飾		
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	イニシャライズデータ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○
n	テーブルサイズ																											○
B	テーブル先頭			○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○					○

プログラム例 1

R0173 13817 13817
[RW018 TINZ(0012)D0030]

動作

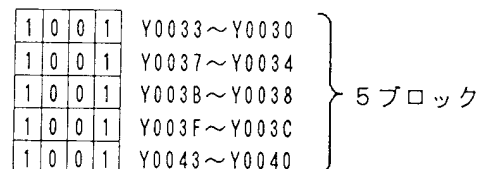
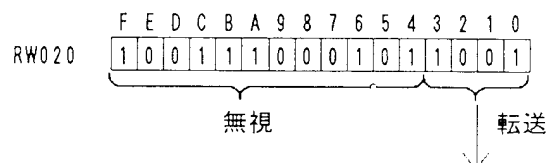
- ・ a 接点 R0173 が O N の時、レジスタ RW018 のデータ (13817) をレジスタ D0030 を先頭としたサイズ 12 のテーブル (D0030~D0041) に転送し、出力を O N にします。



プログラム例 2

- ・オペランド B を桁指定した場合は次のようになります。

R0173 Q1
[RW020 TINZ(0005)Y0030]



FUN25 テーブル転送 (TMOV) *

連続したレジスタのデータを他のレジスタにブロック転送します。

関連命令

条件入力

実行出力

[A TMOV (n) B]

機能

- ・入力条件がONの時、オペランドAを先頭とするテーブルサイズnのデータをオペランドB以降にブロック転送します。
- ・転送元及び転送先のテーブルは、各レジスタ空間の有効範囲である必要があります。
- ・転送元と転送先のテーブル同士が重なってもテーブル転送は可能です。
- ・テーブルサイズ (オペランドn) の範囲は、1~1024です。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

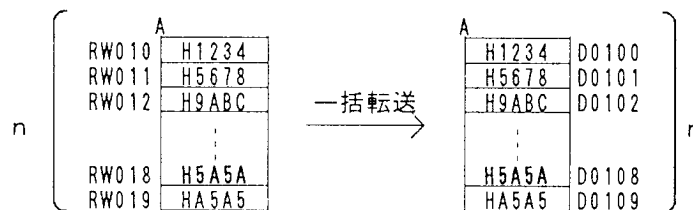
		デバイス										レジスタ										INDEX			定数	修飾	
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J
A	転送元データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○					
n	テーブルサイズ																										○
B	転送先データ		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○			○

プログラム例 1

X000E 04660 04660
[RW010 TMOV (0010)D0100]

動作

- ・a接点X000EがONの時、レジスタRW010を先頭とするテーブルサイズ10ワード分のレジスタのデータを、D0100以降にブロック転送し出力をONにします。
- ・a接点X000EがOFFの時は、転送を行わず出力をOFFにします。



プログラム例 2

- ・オペランドBを桁指定した場合は次のようになります。

X000E Q1
[YW050 TMOV (0004)R0020]



YW050から4ワード分のデータのうち下位4ビットデータをRW002へ4ビット毎に転送します。

プログラム例 3

- オペランド A を桁指定した場合は次のようになります。

X000E Q2
|-----| R0010 TMOV (0005) YW020 |-----|

RW001	H1234
RW002	H5678
RW003	H9ABC
RW004	HDEF0



H0034	YW020
H0012	YW021
H0078	YW022
H0056	YW023
H00BC	YW024

→ 上位 8 ビットには 0 が格納されます。
RW001 の下位 8 ビットデータを先頭とする 5 バイト分のデータを
YW020 を先頭とする 5 ワード分に転送します。

テーブル否定転送

FUN2 6 テーブル否定転送 (TNOT) *

連続したレジスタのデータを反転し、他のレジスタにブロック転送します。

関連命令

条件入力

実行出力

—[A TNOT (n) B]—

機能

- ・オペランドAを先頭とするテーブルサイズnのデータを反転し、オペランドB以降にブロック転送します。
- ・テーブルサイズ（オペランドn）の範囲は、1～1024です。
- ・オペランドA及びBは桁指定できます。
- ・転送元及び転送先のテーブルは、各レジスタ空間の有効範囲であることが必要です。
- ・転送元と転送先のテーブル同士が重なってもテーブル転送は可能です。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

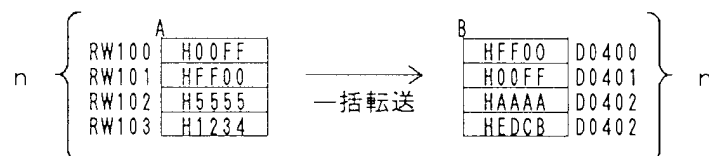
		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	転送元データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○						○
n	テーブルサイズ																											○	
B	転送先データ		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○					○

プログラム例

|X001E H00FF HFF00
|—[RW100 TNOT (0004)D0400]—

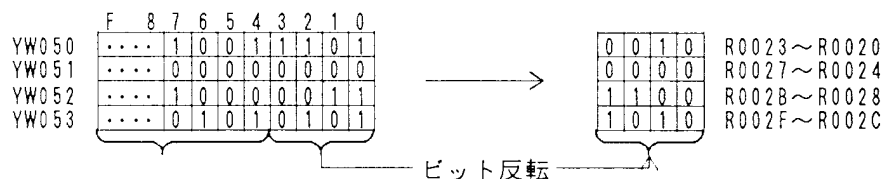
動作

- ・a 接点X001EがONの時、レジスタRW100を先頭とするテーブルサイズ4ワード分のレジスタデータを各ビット毎に反転してレジスタD0400以降にブロック転送し出力をONにします。
- ・A接点X001EがOFFの時は、転送を行わず出力をOFFにします。



プログラム例 2

|X001E Q1
|—[YW050 TNOT (0004)R0030]—



YW050から4ワード分のデータのうち下位4ビットデータをビット毎に反転しRW002へ4ビット毎に転送します。

プログラム例 3

X001E Q2
R0010 TNOT (0005)YW030

- オペランド A を桁指定した場合は次のようになります。

RW001	H1234		H00CB	YW030
RW002	H5678	→	H00ED	YW031
RW003	H9ABC		H0087	YW032
RW004	HDEF0		H00A9	YW033
			H0043	YW034

→ 上位 8 ビットには 0 が格納されます。

RW001 の下位 8 ビットデータを先頭とする 5 バイト分のデータをビット毎に反転し、YW030 を先頭とする 5 ワード分に転送します。

加算

FUN2 7 加算 (+) *

2つのレジスタデータを加算します。

関連命令
キャリー付き加算

条件入力

[A + B → C]

実行出力

オーバーフロー/
アンダーフロー出力

機能

- オペランドAのデータとオペランドBの16ビットデータを加算し、オペランドCに格納します。
- オペランドA、B、Cのデータ範囲は-32768～32767です。これを超える（アンダーフロー／オーバーフロー）とオペランドCにはリミット値を格納し、出力をONにします。
（符号付き16ビットキャリーなし加算）
- レジスタ間の加算時は高速動作を行います。

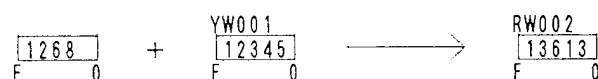
入力	処 理	出力
OFF	不実行	OFF
ON	実行：正常実行	OFF
	実行：オーバーフロー／アンダーフロー	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾				
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K				
A	被加算数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○	
B	加数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
C	和		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○

プログラム例 1

- 動作
- a 接点R011FがONの時、オペランドAのデータ(1268)とレジスタYW001のデータ(12345)の和13613を求め、レジスタRW002に格納し、出力をOFFにします。



プログラム例 2

- 動作
- a 接点R011FがONの時、オペランドAのデータ(10000)とレジスタYW001のデータ(26024)の和を求めます。演算結果がリミット値を超えるため、レジスタRW002には32767格納し、出力をONにします。



減算

FUN28 減算 (-) *

2つのオペランドデータの間で減算を行います。

関連命令
キャリー付き減算

条件入力

{ A - B → C } _____ オーバーフロー／
アンダーフロー出力

機能

- ・オペランドAのデータからオペランドBのデータを減算し、オペランドCに格納します。
- ・オペランドA、B、Cの定数範囲は-32768～32767です。これを超える（アンダーフロー／オーバーフロー）と演算結果（オペランドC）にはリミット値を格納し出力をONにします。
- ・（符号付き16ビットキャリーなし減算）
- ・レジスタ間の減算時は高速動作を行います。

入力	処 理		出力
OFF	不実行		OFF
0 N	実行	正常	OFF
		オーバーフロー/アンダーフロー	0 N

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定	修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
A	被減数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
B	減数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
C	差		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○		○	○	○	○	○

プログラム例 1

R021F -15684 -18232 Y035A
└─┬─┐ └─┬─┐ └─┬─┐
RW002 - 02548 → RW005 ()

動作

- ・a接点R021FがONの時、レジスタRW002のデータ(-15684)とオペランドBのデータ(2548)の差を求め、RW005に(-18232)を格納し、出力をOFFにします。

A:RW002 B C:RW005
┌─┐ ┌─┐ ┌─┐
-15684 - 2548 → -18232
└─┐ └─┐ └─┐
F 0 F 0 F 0

プログラム例 2

R021F -25684 -32768 Y035A
└─┬─┐ └─┬─┐ └─┬─┐
RW002 - 22548 → RW005 ()

- ・レジスタRW002のデータが-25684、オペランドBのデータが22548の時は、演算の結果アンダーフローが発生し、RW005にはリミット値の-32768を格納し、出力をONにします。

A:RW002 B C:RW005
┌─┐ ┌─┐ ┌─┐
-25684 - 22548 → -32768
└─┐ └─┐ └─┐
F 0 F 0 F 0

FUN 2 9 乗算 (*) *

2つのレジスタの間で乗算を行います。

関連命令

条件入力

実行出力

$$\neg [A * B \rightarrow C + 1 \cdot C]$$

機能

- ・オペランドAのデータとオペランドBのデータの積を求めオペランドC + 1・Cに格納します。
- ・オペランドA、Bの定数範囲は-32768~32767です。
(符号付き16ビット乗算)
- ・レジスタ間の乗算時は高速動作を行います。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A	被乗数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
B	乗数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
C+1・C	積		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○			○

プログラム例

```

R034F      12693      0298222035
| | | 23495 * RW031 → RW052·RW051 |_____

```

動作

- 動作
- ・ a 接点R034FがONの時、オペランドAのデータ(23495)とレジスタRW031のデータ(12693)の積(298222035)を求め、倍長レジスタRW052・RW051に格納し、出力をONにします。

$$\begin{array}{|c|c|c|c|c|} \hline A & & & & \\ \hline 23495 & & & & \\ \hline F & & & & 0 \\ \hline \end{array} \times \begin{array}{|c|c|c|c|c|} \hline B:RW031 & & & & \\ \hline 12693 & & & & \\ \hline F & & & & 0 \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|c|c|} \hline C+1:C:RW052 \cdot RW051 & & & & \\ \hline 298222035 & & & & \\ \hline F & & & & 0 \\ \hline \end{array}$$

- ・ a 接点R034FがO F Fの時は、演算を行わず出力をO F Fにします。

除算

FUN 3 0 除算 (/) *

2つのレジスタデータの間で除算を行います。

関連命令

条件入力

実行出力

[A / B → C]

機能

- オペランドAのデータをオペランドBのデータで除し、商をC、剰余をC+1に格納します。
- オペランドA、Bの定数範囲は-32768~32767です。
- オペランドBが0の場合は、エラーフラグをセットし（除算エラー）出力をOFFにします。
- （符号付き16ビット除算）
- レジスタ間の除算時は高速動作を行います。

入力	処 理	出力	ERF
OFF	不実行	OFF	
ON	実行	B ≠ 0: 正常	ON
		B = 0: 除算エラー、不実行	OFF ○

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾				
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K		
A	被除数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○	
B	除数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
C	商		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○		○	○	○		○

プログラム例

R031E -26854 / 00325 → RW027 Y035A ()

動作

- a接点R031EがONの時、レジスタRW022のデータ(-26854)をオペランドBのデータ(325)で除し、商(-82)をレジスタRW027に、剰余(-204)をレジスタRW028に格納し出力をONにします。

A:RW022
 -26854
 F 0

÷

B:
 325
 F 0

→

C:RW027
 -82
 F 0
 -204
 C+1:RW028

補足

- オペランドCにインデックスKを指定すると、剰余データ(C+1)は切り捨てられますのでご注意ください。

倍長加算

FUN3 1 倍長加算 (D+) *

2つの倍長データを加算します。

関連命令

キャリー付き倍長加算

条件入力

オーバーフロー／アンダーフロー出力

-[A+1・A D+ B+1・B → C+1・C]-

機能

- オペランドA+1・AのデータとB+1・Bのデータを加算し、C+1・Cに格納します。
- オペランドA+1・AとB+1・B及びC+1・Cのデータ範囲は-2147483648～2147483647です。
- 演算結果がデータ範囲を越えた（オーバーフロー／アンダーフロー）時は、オペランドC+1・Cにはリミット値を格納し出力をONにします。
（符号付き32ビットキャリーなし加算）

入力	処 理	出力
OFF	不実行	OFF
ON	正常	OFF
	オーバーフロー／アンダーフロー	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾				
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K				
A+1・A	被加算数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○			○	○	
B+1・B	加数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○			○	○
C+1・C	和		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○			○	○

プログラム例 1

R0102 0358426434 0379794886 Y035A
┌─┴─┐ [0021368452 D+ YW001・YW000 → RW031・RW030] ─┬─┐ () ─┴─┐

動作

- a接点R0102がONの時、オペランドA+1・Aのデータ(21368452)と、倍長レジスタYW001・YW000のデータ(358426434)の和(379794886)を求め、レジスタRW031・RW030に格納し、出力をOFFにします。

A+1: A:
021368452
358426434
B+1: YW001 B: YW000
→ + → C+1: RW031 C: RW030
379794886 出力OFF

プログラム例 2

R0102 1358426434 2147483647 Y035A
┌─┴─┐ [1121368452 D+ YW001・YW000 → RW031・RW030] ─┬─┐ () ─┴─┐

動作

- オペランドA+1・Aのデータが(1121368452)、倍長レジスタYW001・YW000のデータが(1358426434)の時、演算結果(2479794886)がオーバーフローするため、レジスタRW031・RW030にはリミット値(2147483647)を格納し、出力をONにします。

A+1: A:
1121368452
1358426434
B+1: YW001 B: YW000
→ + → C+1: RW031 B: RW030
2147483647 出力ON

倍長減算

FUN3 2 倍長減算 (D-) *

2つの倍長データの間で減算します。

関連命令

キャリー付き倍長減算

条件入力

オーバーフロー／アンダーフロー出力

$-(A+1 \cdot A \quad D- \quad B+1 \cdot B \quad + \quad C+1 \cdot C)$

機能

- オペランド $A+1 \cdot A$ のデータから $B+1 \cdot B$ のデータを減算し、 $C+1 \cdot C$ に格納します。
- オペランド $A+1 \cdot A$ と $B+1 \cdot B$ 及び $C+1 \cdot C$ のデータ範囲は -2147483648 ~ 2147483647 です。
- 演算結果がこの範囲を越えた (アンダーフロー／オーバーフロー) 時は、オペランド $C+1 \cdot C$ にリミット値を格納し出力を ON にします。
(符号付き 32 ビットキャリー無し減算)

入力	処	理	出力
OFF	不実行		OFF
ON	実行	正常	OFF
		オーバーフロー／アンダーフロー	ON

オペランド

opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	定数	修飾
A+1・A	被減算数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
B+1・B	減数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
C+1・C	差		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○

プログラム例

R0112 -0583254795 1153681537 -1736936332 Y035A
└─┬─┐ └─┬─┐ └─┬─┐ └─┬─┐
RW053・RW052 D- RW050・RW049 → RW048・RW047 ()

動作

- a 接点 R0112 が ON の時、倍長レジスタ RW053・RW052 のデータ (-583254795) と、倍長レジスタ RW050・RW049 のデータ (1153681537) の差 (-1736936332) を求め、倍長レジスタ RW048・RW047 に格納し、出力を OFF にします。

$$\begin{array}{l} \text{A+1:RW053} \quad \text{A:RW052} \\ \boxed{-583254795} \\ \text{B+1:RW050} \quad \text{B:RW049} \\ \boxed{1153681537} \end{array} \quad \begin{array}{c} \rightarrow \\ + \\ \rightarrow \end{array} \quad \begin{array}{l} \text{C+1:RW048} \quad \text{C:RW047} \\ \boxed{-1736936332} \end{array} \quad \text{出力 OFF}$$

プログラム例 2

R0112 -1583264795 1153681537 -2147483648 Y035A
└─┬─┐ └─┬─┐ └─┬─┐ └─┬─┐
RW053・RW052 D- RW050・RW049 → RW048・RW047 ()

動作

- 倍長レジスタ RW053・RW052 のデータが (-1583264795)、倍長レジスタ RW050・RW049 のデータが (1153681537) の場合、演算結果 (-2147483648) がアンダーフローするため倍長レジスタ RW048・RW047 にはリミット値 (-2147483648) を格納し、出力を ON にします。

$$\begin{array}{l} \text{A+1:RW053} \quad \text{A:RW052} \\ \boxed{-1583264795} \\ \text{B+1:RW050} \quad \text{B:RW049} \\ \boxed{1153681537} \end{array} \quad \begin{array}{c} \rightarrow \\ + \\ \rightarrow \end{array} \quad \begin{array}{l} \text{C+1:RW048} \quad \text{C:RW047} \\ \boxed{-2147483648} \end{array} \quad \text{出力 ON}$$

[illegible]

FUN 3 3 倍長乗算 (D*) *

2つの倍長データを乗算します。

関連命令

条件入力 実

$\neg [A+1 \cdot A \quad D* \quad B+1 \cdot B \quad \rightarrow \quad C+1 \cdot C]$ _____

実行出力

機能

- ・オペランド $A + 1 \cdot A$ のデータと $B + 1 \cdot B$ のデータの積を求め、 $C + 3 \cdot C + 2 \cdot C + 1 \cdot C$ に格納します。
- ・オペランド $A + 1 \cdot A$ と $B + 1 \cdot B$ のデータ範囲は -2147483648 ~ 2147483647 です。

(符号付き 32 ビットキャリー無し乗算)

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K		
A+1・A	被乗数	○	○	○	○	○	○			○		XW	○	○	○	○	○	○	○	○	○	○		○	○		○	○
B+1・B	乗数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
C+1・C	積		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○	○	○			

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A+1	A 被乗数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○		○	○	
B+1	B 乗数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
C+1	C 積		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○		○	○

プログラム例

R0002	0000010203	1644933620
RW048-RW047	D*	0000582172 → RW030-RW029

R0002 0000010203 1644933620
RW048·RW047 D* 0000582172 → RW030·RW029

動作

- ・ a 接点R0002がONの時、倍長レジスタRW048・RW047のデータ(10203)と、オペランドB + 1
- ・ Bのデータ(582172)の積(5939900916)を求め、4倍長レジスタRW032・RW031・RW030・RW029に格納し、出力をONにします。

- ・ a 接点R0002がONの時、倍長レジスタRW048・RW047のデータ(10203)と、オペランドB + 1
- ・ Bのデータ(582172)の積(5939900916)を求め、4倍長レジスタRW032・RW031・RW030・RW029に格納し、出力をONにします。

A+1:RW048 A:RW047
10203
582172
B+1: B:

$\times \longrightarrow$

C+3:RW032 C+2:RW031 C+1:RW030 C:RW029
5939900916

- ・画面には指定した2ワードのデータが表示されます。

補足

- ・オペランドC+1・Cにインデックスを指定すると積データ (C+3・C+2) は切り捨てられますのでご注意ください。

- ・オペランドC+1・Cにインデックスを指定すると積データ (C+3・C+2) は切り捨てられますのでご注意ください。

倍長除算

FUN3 4 倍長除算 (D/) *

2つの倍長データの間で除算します。

関連命令

条件入力

実行出力

[A+1・A D/ B+1・B → C+1・C]

機能

- オペランドA+1・AのデータとB+1・Bのデータを除し、商をC+1・Cへ、剰余をC+3・C+2に格納します。
- オペランドA+1・AとB+1・Bのデータ範囲は-2147483648~2147483647です。
(符号付き32ビットキャリー無し除算)

入力	処 理	出力	ERF
OFF	不実行	OFF	
ON	B+1・B≠0 : 正常	ON	
	B+1・B=0 : 不実行	OFF	○

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾				
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K		
A+1・A	被除数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○		○	○	
B+1・B	除数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○		○	○	
C+1・C	商		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○			○	○

プログラム例

R032E [1784961564 D/ -0001398245 -0000001276 Y035A
RW024・RW023 → RW015・RW014] ()

動作

- a 接点R032EがONの時、オペランドA+1・Aのデータ(1784961564)を倍長レジスタRW024・RW023のデータ(-1398245)で除し、倍長レジスタC+1・Cに商(-1276)、倍長レジスタRW017・RW016に剰余(800944)を格納し、出力をONにします。

A+1: A:RW047
 1784961564
 B+1:RW024 B:RW023
 -1398245
 C+1:RW015 C:RW014
 -1276
 C+3:RW017 C+2:RW016
 800944

→ ÷ →

- A 接点R032EがOFFの時は、演算を行わず出力をOFFにします。

補足

- オペランドC+1・Cにインデックスレジスタを指定すると剰余データ(C+3・C+2)は切り捨てられますのでご注意ください。

キャリー付き加算

FUN 3 5 キャリー付き加算 (+C)*

2つのレジスタデータを加算します。(キャリー付き)

関連命令

条件入力

オーバーフロー／アンダーフロー出力

[A +C B → C]

機能

- オペランドAとオペランドBをキャリーフラグを含め加算を行いオペランドCに格納します。演算結果に桁上りがあればキャリーフラグをセットします。(符号付き16ビットキャリー付き加算)
- オペランドA、B、Cのデータ範囲は-32768～32767です。
- 演算結果がデータ範囲を越えると(オーバーフロー／アンダーフロー)出力をONにします。

入力	処 理	出力	CF
OFF	不実行	OFF	
ON	実行	正常	桁上り無し
		正常	桁上り有
		オーバーフロー／アンダーフロー: 桁上り無し	
		オーバーフロー／アンダーフロー: 桁上り有	
		OFF	リセット
		OFF	セット
		ON	リセット
		ON	セット

オペランド

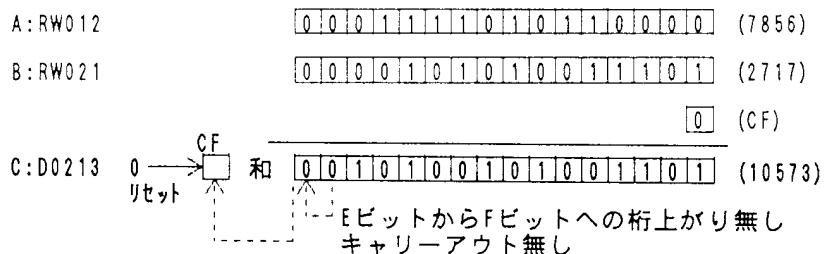
		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A	被加算数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○	
B	加数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
C	和		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○	○	○	○

プログラム例 1

R0103 07856 02717 10573 Y035A
[RW012 +C RW021 → D0213] ()

動作

- a接点R0103がONの時、レジスタRW012のデータ(7856)とレジスタRW021のデータ(2717)の和(10573)を求めます。桁上り(キャリアウト)はないのでキャリーフラグをリセットし、レジスタD0213に演算結果(10573)を格納します。



R0103 -32768 -10000 22768 Y035A
[] [RW012] +C RW021 → D0213 [] ————— ()

- レジスタRW012が-32768、レジスタRW021が-10000の場合は、Fビットからの桁上がり（キャリアアウト）を生じキャリアフラグをセットします。D0213には22768が格納されます。

CF 1 0 1 0 1 1 0 0 0 1 1 1 1 0 0 0 0 (22768) (704* -704)

R0103 20000 30000 -15536 Y035A
└─┬──┴─┐ ┌─┬──┴─┐ +C ─┬──┴─┐ → D0213 ─────────────────────────────────── ()

- レジスタRW012が20000、レジスタRW021が30000の場合は、EビットからFビットへの桁上がりによりオーバーフローになります。また、Fビットからのキャリーアウトはないため、キャリーフラグをリセットします。D0213には-15536が格納されます。

CF $\rightarrow$

1	1	0	0	0	0	1	1	0	1	0	1	0	0	0	0

 (-15536) (オ-ハ-70-)

R0103 -00025 -10000 09975 Y035A
 R0012 +C R0021 → D0213 ()

- レジスタRW012が-00025、レジスタRW021が-10000の場合は、Fビットからの桁上がり（キャリーアウト）とEビットからFビットへの桁上がりの両方を生じます。この場合はキャリーフラグをセットし正常計算し、演算結果は9975になります。

CF 1 0 0 1 0 0 1 1 0 1 1 1 0 1 1 1 (9975)

- ・ オーバーフロー・アンダーフローは、キャリアアウトあるいはEビットからFビットへの桁上りのどちらか一方があったときに発生します。

FUN3 6 キャリー付き減算 (ーC) *

2つのレジスタデータ間で減算します。(キャリー付き)

関連命令

条件入力

オーバーフロー／アンダーフロー出力

$$\neg [A \rightarrow C \wedge B \rightarrow C]$$

機能

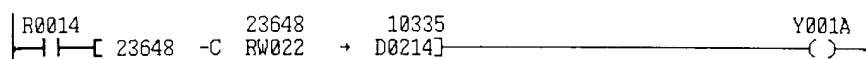
- [illegible]

入力	処 理		出力	CRF	
OFF	不実行		OFF		
O N	実行	正常	桁下がり無	OFF	リセット
			桁下がり有	OFF	セット
		オーバ・フロー: 桁下がり有		O N	セット
		アンダ・フロー: 桁下がり無		O N	リセット

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A	被減数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
B	減数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
C	差		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○

プログラム例 1



動作

- ・ a 接点R0104がONの時、オペランドAのデータ(23648)とレジスタRW022のデータ(10335)及びキャリーフラグ0の差(13313)を求めます。ポローイン(16ビットデータ外からの桁下がり)はないのでキャリーフラグをリセットします。レジスタD0214には演算結果(13313)を格納し、正常実行します。出力はOFFになります。

A:

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
0	1	0	1	1	1	0	0	0	1	1	0	0	0	0	0	0

 (23648)

B: RW022

0	0	1	0	1	0	0	0	0	1	0	1	1	1	1	1	1

 (10335)

0 (CF)

→

C: D0214

0	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	1

 (13313)

0 →

CF

 差

リセット

△△ EビットからFビットへの桁下がり無し
---- ボローイン無し

プログラム例 2

R0104 [10000 -C RW022 → D0214] Y035A

動作

- オペランドAが-20000、レジスタRW022が30000の場合は、FビットからEビットへの桁下がりし、アンダーフローになります。キャリーフラグをリセットし、オペランドCに15536を格納します。

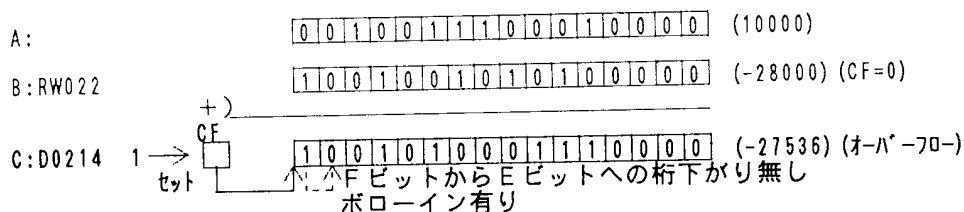


プログラム例 3

R0104 [10000 -C RW022 → D0214] Y035A

動作

- オペランドAが10000、レジスタRW022が-28000の場合は、ボローインを生じオーバーフローになります。キャリーフラグはセットされます。D0214には-27536が格納されます。

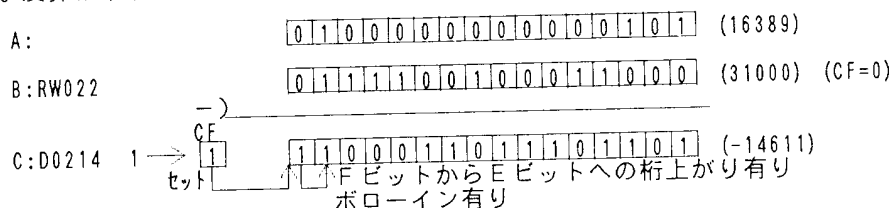


プログラム例 4

R0104 [16389 -C RW022 → D0214] Y035A

動作

- オペランドAが16389、レジスタRW022が31000の場合は、ボローインとFビットからEビットへの桁下がりの両方を生じます。この場合キャリーフラグをセットのうえ正常計算します。演算結果(-14611)はD0214に格納されます。



※キャリーフラグはボローインがあったときセットされます。

- オーバーフロー、アンダーフローは、FビットからEビットへの桁下がりあるいはボローインのどちらかがあったとき発生します。

[illegible]

FUN3 7 キャリー付き倍長加算 (D + C) *

2つの倍長レジスタ間で加算します。(キャリー付き)

關連命令

条件入力

オーバーフロー／アンダーフロー出力

$$- [A+1 \cdot A \quad D+C \quad B+1 \cdot B \quad \rightarrow \quad C+1 \cdot C]$$

機能

- ・倍長データ $A + 1 \cdot A$ と $B + 1 \cdot B$ 及びキャリーフラグを加算し $C + 1 \cdot C$ に格納します。桁上がりがあればキャリーフラグをリセットします。
 - ・オペランド $A + 1 \cdot A$ 、 $B + 1 \cdot B$ の定数範囲は -2147483648 ~ 2147483647 です。
 - ・オペランド $C + 1 \cdot C$ のデータ範囲は -2147483648 ~ 2147483647 で、演算結果がこの範囲を超えるとオーバーフロー・アンダーフローとなります。
- (符号付き 32 ビット、キャリー付き加算)

入力	処 理		出力	CRF	
OFF	不実行		OFF		
O N	実行	正常	桁上がり無	OFF	リセット
			桁上がり有	OFF	セット
		オーバーフロー/アンダーフロー	桁上がり無	O N	リセット
		オーバーフロー/アンダーフロー	桁上がり有	O N	セット

オペランド

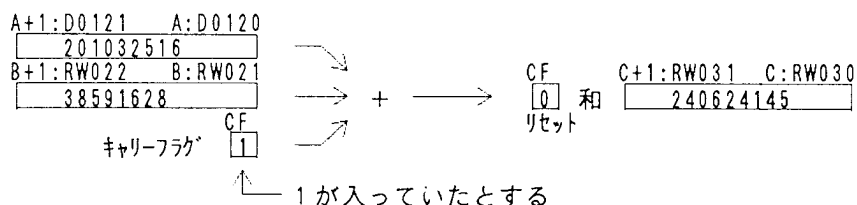
		デバイス										レジスタ										INDEX			定数	修飾					
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K			
A+1・A	被加算数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○		○	○		
B+1・B	加数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○			○	○		○	○	
C+1・C	和		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○		○	○			○	○

プログラム例 1

R0326 0201032516 0038591628 0240624145 Y035A
D0121-D0120 D+C RW022-RW021 + RW031-RW030

動作

- ・ a 接点R0326がONの時、倍長レジスタD0121・D0120のデータ(201032516)と倍長レジスタW022・RW021のデータ(38591628)及びキャリーフラグ(1)の和を求め、倍長レジスタRW031・W030に240624145を格納します。この時Fビットからの桁上がりは生じないのでキャリーフラグをリセットします。同様にオーバーフロー／アンダーフローも発生していないので出力をOFFにします。



- ・ a 接点R0326がOFFの時は演算を行わず出力をOFFにします。

© 2006 The Authors
Journal compilation © 2006 Blackwell Publishing Ltd

FUN3 8 キャリー付き倍長減算 (D-C) *

2つの倍長レジスタ間で減算します。

関連命令

条件入力 オーバーフロー／アンダーフロー出力

オーバーフロー／アンダーフロー出力

$$- [A+1 \cdot A \quad D-C \quad B+1 \cdot B \rightarrow C+1 \cdot C] \text{-----}$$

機能

- ・倍長データ $A + 1 \cdot A$ から $B + 1 \cdot B$ 及びキャリーフラグを減算し $C + 1 \cdot C$ に格納します。桁上がりがあればキャリーフラグをリセットします。
- ・オペランド $A + 1 \cdot A$ 、 $B + 1 \cdot B$ の定数範囲は -2147483648 ~ 2147483647 です。
- ・オペランド $C + 1 \cdot C$ のデータ範囲は -2147483648 ~ 2147483647 で、演算結果がこの範囲を超えるとオーバーフロー・アンダーフローとなります。
(符号付き32ビット、キャリー付き減算)

入力		処 理		出力	CRF
OFF		不実行		OFF	
O N	実行	正常	桁上がり無	OFF	リセット
			桁上がり有	OFF	セット
		オーバーフロー/ アンダーフロー: 桁上がり無		O N	セット
		オーバーフロー/ アンダーフロー: 桁上がり有		O N	リセット

オペランド

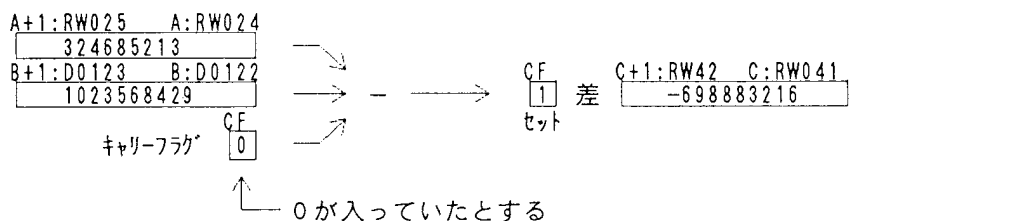
		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A+1	A 被減算数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
B+1	B 減数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
C+1	C 差		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○			○

プログラム例 | R0327 0324685213 1023568429 -0698883216 Y035A |

R0327 0324685213 1023568429 -0698883216 Y035A
 [RW025-RW024 D-C D0123-D0122 + RW042-RW041] ()

動作 • a 接点R0327がONの時、倍長レジスタRW025・RW024のデータ(324685312)から倍長レジスタ

- ・ a 接点R0327がONの時、倍長レジスタRW025・RW024のデータ(324685312)から倍長レジスタD0123・D0122のデータ(1023568429)及びキャリーフラグ(1)を減算し、倍長レジスタRW042・RW041に-698883216を格納します。この時Fビットからの桁下上がりが生じているためキャリーフラグをセットします。またオーバーフロー／アンダーフローは発生していないので出力をOEにします。



- ・ a 接点R0327がOFFの時は演算を行わず出力をOFFにします。

符号無し乗算

FUN3 9 符号無し乗算 (U*) *

レジスタデータの乗算を行います。(符号無し)

関連命令

条件入力	実行出力
$- [\quad A \quad U \times \quad B \quad + \quad C + 1 \cdot C \quad]$	

条件入力	実行出力
$- [\quad A \quad U \times \quad B \quad + \quad C + 1 \cdot C \quad]$	

条件入力

実行出力

$\neg [A \quad U \times \quad B \quad \rightarrow \quad C + 1 \cdot C]$ —————

機能

- ・オペランドA、オペランドBの符号無し16ビットデータを乗算し、オペランドCに格納します。
- ・オペランドA、Bの定数範囲は0～65535です。
(符号無し、16ビット乗算)

- 機能
- ・オペランドA、オペランドBの符号無し16ビットデータを乗算し、オペランドCに格納します。
 - ・オペランドA、Bの定数範囲は0～65535です。
(符号無し、16ビット乗算)

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	被乗数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○
B	乗数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○
C+1・C	積		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○	○	

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾				
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K		
A	被乗数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○	
B	乗数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
C+1	C 積		○	○	○	○	○			○			○	○	○	○	○	○	○	○	○	○		○		○	○			○

プログラム例

R0001	62312	59426	3702952912
D0125	U*	RW032	→ RW087・RW086

プログラム例

R0001	62312	59426	3702952912
D0125	U*	RW032	→ RW087・RW086

動作

- ・ a 接点R0001がONの時、レジスタD0125のデータ(62312)とレジスタRW032のデータ(59426)の積(3702952912)を求め倍長レジスタRW087・RW086に格納し、出力をONにします。

- 動作
- ・ a 接点R0001がONの時、レジスタD0125のデータ(62312)とレジスタRW032のデータ(59426)の積(3702952912)を求め倍長レジスタRW087・RW086に格納し、出力をONにします。

A: D0125 B: RW032 C+1: RW087 C: RW086

62312

 ×

59426

 →

3702952912

F 0 F 0 F 0 0

- ・ a 接点R0001がOFFの時は、演算を行わず出力をOFFにします。

[illegible]

条件入力

実行出力

A B C

- ・オペランドAをオペランドBで除し、商をCへ、剰余をC+1へ格納します。
- ・オペランドA、Bの定数範囲は0~65535です。
(符号無し16ビット除算)

オペランド

プログラム例

動作

- ・ a 接点R0002がONの時、レジスタRW033のデータ(54321)とオペランドBのデータ(127)で除し商(427)をレジスタD0123に、剰余(92)をレジスタD0124にそれぞれ格納し、出力をONにします。

$$\begin{array}{c} \text{A: RW033} \\ \boxed{54321} \\ \text{F} \qquad \text{O} \end{array} \div \begin{array}{c} \boxed{127} \\ \text{F} \qquad \text{O} \end{array} \longrightarrow \begin{array}{c} \text{C: D0123} \\ \boxed{427} \\ \text{F} \qquad \text{O} \\ \boxed{92} \\ \text{C+1: D0124} \end{array}$$

- ・オペランドBが0の場合は、エラーフラグのみをセットし（除算エラー）出力をOFFにします。
- ・a接点R0002がOFFの時は、演算を行わず出力をOFFにします。

- ・オペランドCにインデックスKを指定すると、剰余データ(C+1)は切り捨てられますのでご注意ください。

符号無し倍長単長除算

FUN4 1 符号無し倍長単長除算 (DIV)*

倍長レジスタのデータを、単長レジスタのデータで除算し、商と剰余を
各々格納します。(符号無し)

関連命令

条件入力

実行出力

$[A+1 \cdot A \text{ DIV } B \rightarrow C]$

機能

- オペランド $A+1 \cdot A$ をオペランド B で除し、商を C へ、剰余を $C+1$ へ格納します。
- オペランド $A+1 \cdot A$ の定数範囲は、0~4294967295 です。
- オペランド B の定数範囲は、0~65535 です。
- オペランド B が 0 の場合は除算エラーになります。(出力 OFF)
- 演算結果 (オペランド C) のデータ範囲は 0~65535 で、演算結果が 65535 を超えると商に 65535、剰余に 0 を格納します。但し、出力は ON になります。(符号無し 32 ビット)

オペランド

入力	処 理	出力	ERF
OFF	不実行	OFF	
ON	B ≠ 0: オペランド無し 正常	ON	
	B ≠ 0: オペランド有り リミット	ON	○
	B = 0: 除算エラー 不実行	OFF	○

		デ バ イ ス										レ ジ ス タ										INDEX			定	修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
A+1・A	被除数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
B	除数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
C	商		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○	○		○

プログラム例

R0100 3109876542 60877
└─┬─┤ D0314・D0313 DIV 51084 → RW051 ─┬─┘

動作

- a 接点 R0100 が ON の時、倍長レジスタ D0314・D0313 のデータ (3109876542) とレジスタのデータ (51084) で除し、商 (60877) をレジスタ RW051 に、剰余のデータ 35874 をレジスタ RW052 にそれぞれ格納し、出力を ON にします。

$$\begin{array}{|c|c|} \hline A+1:D0314 & A:D0313 \\ \hline \hline 3109876542 & \\ \hline F & 0 & F & 0 \\ \hline \end{array} \div 51084 \longrightarrow \begin{array}{|c|c|} \hline C:RW051 & \\ \hline \hline 60877 & 427 \\ \hline F & 0 \\ \hline \hline 35874 & \\ \hline C+1:RW052 & \\ \hline \end{array}$$

- a 接点 R0100 が OFF の時は、演算を行わず出力を OFF にします。

補足

- オペランド C にインデックス K を指定すると剰余データ (C+1) は切り捨てられますのでご注意ください。

[T 3 Hのみ使用可能]

命令連関

实行出力

100% 99% 98% 97% 96% 95% 94% 93% 92% 91% 90% 89% 88% 87% 86% 85% 84% 83% 82% 81% 80% 79% 78% 77% 76% 75% 74% 73% 72% 71% 70% 69% 68% 67% 66% 65% 64% 63% 62% 61% 60% 59% 58% 57% 56% 55% 54% 53% 52% 51% 50% 49% 48% 47% 46% 45% 44% 43% 42% 41% 40% 39% 38% 37% 36% 35% 34% 33% 32% 31% 30% 29% 28% 27% 26% 25% 24% 23% 22% 21% 20% 19% 18% 17% 16% 15% 14% 13% 12% 11% 10% 9% 8% 7% 6% 5% 4% 3% 2% 1% 0%

機能

- ・オペランド $A+1 \cdot A$ のデータと $B+1 \cdot B$ のデータの積をとり、その値を $B+3 \cdot B+2$ のデータで除算し、商を $C+1 \cdot C$ へ、剰余を $C+3 \cdot C+2$ に格納します。
- ・オペランド $A+1 \cdot A$ と $B+1 \cdot B$ と $B+3 \cdot B+2$ のデータ範囲は -2147483648～2147483647 です。
- ・オペランド $B+3 \cdot B+2$ が 0 の場合は除算エラーになります。（出力 OFF）
- ・演算結果（オペランド $C+1 \cdot C$ ）のデータ範囲は -2147483648～2147483647 で、演算結果がこの範囲を超えるとエラーフラグを ON にし、次の演算結果を格納します。ただし、出力は、ON になります。

入力	処 理		出力	ERF
OFF	不実行		OFF	
0 N	B+3・B+2≠1、オーバーフローなし	正常	0 N	
	B+3・B+2≠1、オーバーフローあり	リミット	0 N	0 N
	B+3・B+2=0	不実行	OFF	0 N

演算結果

正にオハ ⁺ 70-:商=2147483647	剰余=0
負にオハ ⁻ 70-:商=-2147483647	剰余=0

負にオハ-7口-：商=-2147483647 剰余=0

オペランド		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K		
A+1・A	被乗算データ											○	○	○	○	○	○	○	○	○	○						○	○
B+1・B	乗算・除算データ											○	○	○	○	○	○	○	○	○	○							○
C+1・C	商・剰余データ												○	○	○	○	○	○	○	○	○							○

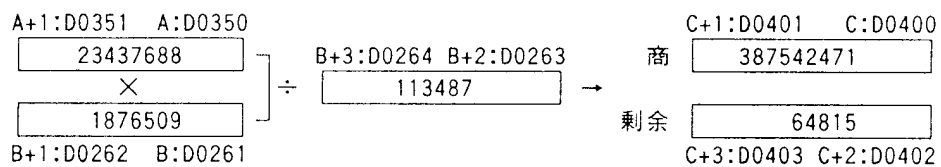
桁指定はサポートしません。

プログラム例

R0200 0023437688 0001876509 0387542471
┌───┴───┐ ┌───┴───┐ ┌───┴───┐
D0351・D0350 D*/ D0262・D0261 → D0401・D0400

動作

- ・ a 接点R0200がO Nの時、倍長レジスタD0351・D0350のデータ(23437688)と倍長レジスタD0262・D0261のデータ(1876509)の積を倍長レジスタD0264・D0263のデータ(113487)で除し、商(387542471)を倍長レジスタD0401・D0400に、剰余(64815)を倍長レジスタD0403・D0402に格納し、出力をO Nにします。



- ・ a 接点R0200がO F Fの時は、演算を行わず出力をO F Fにします。

1. *Journal of the American Medical Association*, 2000; 284: 2689-2695.

FUN43 インクリメント (+1) *

レジスタデータを+1します。

関連命令

実行出力

Diagram illustrating a control system with a feedback loop. The input is labeled "条件入力" (Conditional Input). The output is labeled "実行出力" (Execution Output). The feedback path is labeled "-[+1 A]".

機能

- ・オペランドAのデータを1インクリメントします。
- ・データは符号付き16ビットデータです。
- ・演算結果のデータ範囲は、-32768～32767です。
- ・演算結果がデータ範囲を超えた場合でも、オーバーフローにはなりません。

- ・オペランドAのデータを1インクリメントします。
- ・データは符号付き16ビットデータです。
- ・演算結果のデータ範囲は、-32768~32767です。
- ・演算結果がデータ範囲を超えた場合でも、オーバーフローにはなりません。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	演算データ		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○	○	○	○			○

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾	
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J
A	演算データ		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○	○	○	○	

プログラム例

X0101 00750
[+1 RW023]

- ・ a 接点X0101がONの時、レジスタRW023のデータ(750)をインクリメントし、演算結果(751)をレジスタRW023に再格納し出力をONにします。

A:RW023 + 1 → A:RW023
 [00750] [00751]

- オペランドAのデータが32767の場合は、-32768がオペランドAに格納されます。

$$\begin{array}{|c|} \hline A:RW023 \\ \hline 32767 \\ \hline \end{array} + 1 \longrightarrow \begin{array}{|c|} \hline A:RW023 \\ \hline -32768 \\ \hline \end{array}$$

- ・ a 接点X0101がOFFの時は、演算を行わず出力をOFFにします。

© 2006 The Authors
Journal compilation © 2006 Blackwell Publishing Ltd

FUN 4 4 倍長インクリメント (D+1) *

倍長レジスタのデータを+1します。

關連命令

条件入力

実行出力

$$- [D+1 \quad A+1 \cdot A]$$

機能

- ・オペランド A + 1 ・ A のデータを 1 インクリメントします。
- ・データは符号付き 32 ビットデータです。
- ・演算結果のデータ範囲は、-2147483648 ~ 2147483647 です。
- ・演算結果がデータ範囲を超えた場合でも、オーバーフローにはなりません。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定	修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
A+1	A 演算データ		○	○	○	○	○			1	○		○	○	○	○	○	○	○	○	○		○	○	○				○

プログラム例



動作

- ・ a 接点X0102がONの時、倍長レジスタRW028・RW027のデータ(123456789)をインクリメントし、演算結果(1234567890)を倍長レジスタRW028・RW027に再格納し出力をONにします。



- ・オペランド A + 1 ・ A のデータ (2147483647) をインクリメントしたときは、-2147483648 が再格納されます。



- ・ a 接点X0102がOFFの時は、演算を行わず出力をOFFにします。

Diagram illustrating the function block structure:

- 条件入力** (Condition Input): Inputs are `-1` and `A`.
- 実行出力** (Execution Output): The output is `_____`.

- ・オペランドAのデータを1デクリメントします。
- ・データは符号付き16ビットデータです。
- ・演算結果のデータ範囲は、-32768～32767です。
- ・演算結果がデータ範囲を超えた場合でも、アンダーフローにはなりません。

オペランド

プログラム例

- ・ a 接点X0103がONの時、レジスタRW025のデータ(3665)をデクリメントし、演算結果(3664)をレジスタRW025に再格納し出力をONにします。

A:RW025 - 1 → A:RW025

-32768

32767

- 94 -

倍長デクリメント

FUN 4 6 倍長デクリメント (D-1) *

倍長レジスタデータを-1します。

関連命令

条件入力

実行出力

-[D-1 A+1・A]

機能

- ・オペランドA+1・Aのデータを1デクリメントします。
- ・データは符号付き32ビットデータです。
- ・演算結果のデータ範囲は、-2147483648~2147483647です。
- ・演算結果がデータ範囲を超えた場合でもアンダーフローにはなりません。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾	
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J
A+1・A	演算データ		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○	○	○		○

プログラム例

X0104 2109876544
┌──┴──[D-1 RW035・RW034]──┐

動作

- ・a接点X0104がONの時、倍長レジスタRW035・RW034のデータ(2109876544)をデクリメントし、演算結果(2109876543)を倍長レジスタRW035・RW034に再格納し出力をONにします。

A+1:RW035 A:RW034 A+1:RW035 A:RW034
┌──────────┴──────────┐ 1 ──────────> ┌──────────┴──────────┐
2109876544 2109876543

- ・オペランドA+1・Aのデータ(-2147483648)をデクリメントしたときは2147483647が再格納されます。

A+1:RW035 A:RW034 A+1:RW035 A:RW034
┌──────────┴──────────┐ 1 ──────────> ┌──────────┴──────────┐
-2147483648 2147483647

- ・a接点X0104がOFFの時は、演算を行わず出力をOFFにします。

論理積

FUN 48 論理積 (AND) *	
2つのレジスタデータのビット毎の論理積を求めます。	関連命令 倍長論理積

条件入力	実行出力
[A AND B - C]	

機能	入力	処 理	出力
・オペランドAの内容とオペランドBの内容とのビット毎の論理積を求めオペランドCに格納します。	OFF	不実行	OFF
・オペランドA, B, Cのデータ範囲はH0000~HFFFFです。	ON	実行	ON

オペランド

		デバイス										レジスタ										INDEX			定数	修飾		
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K		
A	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
B	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
C	演算結果		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○	○	○	○		○

プログラム例

R0112	H4321	H8765	H0321
[R012	AND	R003	- D1231]

動作	
RW012	F E D C B A 9 8 7 6 5 4 3 2 1 0 0 1 0 0 0 0 0 1 1 0 0 1 0 0 0 1
	(ビット毎の論理積)
RW003	1 0 0 0 0 0 1 1 1 0 1 1 0 0 1 0 1
	↓
D1231	0 0 0 0 0 0 0 1 1 0 0 1 0 0 0 0 1

- ・a接点R0112がONの時、レジスタRW012の内容(H4321)とレジスタRW003の内容(H8765)のビット毎の論理積を求めます。結果(H0321)をレジスタD1231に格納し、出力をONにします。

倍長論理積

FUN 4 9 倍長論理積 (DAND) *

2つの倍長レジスタデータのビット毎の論理積を求めます。

関連命令
論理積

条件入力

実行出力

—[A+1・A DAND B+1・B → C+1・C]—

機能

- ・オペランドA+1・Aの内容とオペランドB+1・Bの内容のビット毎の論理積を求めオペランドC+1・Cに格納します。
- ・オペランドA+1・A, B+1・Bのデータ範囲はH00000000~HFFFFFFFです。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A+1・A	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○		○	○
B+1・B	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○		○	○
C+1・C	演算結果		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○			○

プログラム例

R0011 H2014B762 H1268A5B3 H0000A522
—[RW022・RW021 DAND D0213・D0212 → RW033・RW032]—

動作

RW022

F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
0	0	1	0	0	0	0	0	0	0	0	1	0	1	0	0

 • RW021

F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
1	0	1	1	0	1	1	1	0	1	1	0	0	0	1	0

(ビット毎の論理積)

D0213

0	0	0	1	0	0	1	0	0	1	1	0	1	0	0	0

 • D0212

1	0	1	0	0	1	0	1	1	0	1	1	0	0	1	1

↓

RW033

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

 • RW032

1	0	1	0	0	1	0	1	0	0	1	0	0	0	1	0

- ・a接点R0011がONの時、倍長レジスタRW022・RW021の内容(H2014B762)と倍長レジスタD0213・D0212の内容(H1268A5B3)のビット毎の論理積を求めます。結果(H0000A522)を倍長レジスタRW033・RW032に格納し出力をONにします。

論理和

FUN 5 0 論理和 (OR) *	
2つのレジスタデータのビット毎の論理和を求めます。	関連命令 倍長論理和

条件入力	実行出力
-[A OR B → C]	

機能

- ・オペランドAの内容とオペランドBの内容とのビット毎の論理和を求めオペランドCに格納します。
- ・オペランドA, B, Cのデータ範囲はH0000～HFFFFです。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K		
A	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
B	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
C	演算結果		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○	○	○	○		○

プログラム例	R0113 H432A H8765 HC76F └─┬─┐ └─┬─┐ OR └─┬─┐ → D1232

動作	
RW024	F E D C B A 9 8 7 6 5 4 3 2 1 0 0 1 0 0 0 0 0 1 1 0 0 1 0 1 0 1 0
	(ビット毎の論理和)
RW004	1 0 0 0 0 0 1 1 1 0 1 1 0 0 1 0 1
	↓
D1232	1 1 0 0 0 0 1 1 1 0 1 1 0 1 1 1 1

- ・ a 接点R0113がONの時、レジスタRW024の内容(H432A)とレジスタRW004の内容(H8765)のビット毎の論理和を求めます。結果(HC76F)をレジスタD1232に格納し、出力をONにします。

倍長論理和

FUN 5 1 倍長論理和 (DOR) *

2つの倍長レジスタデータのビット毎の論理和を求めます。

関連命令
論理和

条件入力

実行出力

—[A+1・A DOR B+1・B → C+1・C]—

機能

- ・オペランドA+1・Aの内容とオペランドB+1・Bの内容のビット毎の論理和を求めオペランドC+1・Cに格納します。
- ・オペランドA+1・A, B+1・Bのデータ範囲はH00000000~HFFFFFFFです。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デバイス										レジスタ										INDEX			定数	修飾				
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K		
A+1・A	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○		○	○	
B+1・B	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○			○	○		○	○
C+1・C	演算結果		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○			○	

プログラム例

R0100 HD9B6837E HFBFEBF7F
H2A5CBD67 DOR D0181・D0180 → RW033・RW032

動作

H2A5C F E D C B A 9 8 7 6 5 4 3 2 1 0 0 0 1 0 1 1 1 0 0 0 1 0 1 1 1 0 0 • HBD67 F E D C B A 9 8 7 6 5 4 3 2 1 0 1 0 1 1 1 1 0 1 0 1 1 0 0 1 1 1 1

(ビット毎の論理積)

D0181 1 0 1 1 1 1 0 0 1 1 0 1 1 0 1 1 0 • D0180 1 0 0 0 0 0 0 1 1 0 1 1 1 1 1 1 0

↓

RW033 1 0 1 1 1 1 1 0 1 1 1 1 1 1 1 0 • RW032 1 0 1 1 1 1 1 1 0 1 1 1 1 1 1 1

- ・ a 接点R0100がONの時、オペランドA+1・Aの定数(H2A5CBD67)と倍長レジスタD0181・D0180の内容(HD9B6837E)のビット毎の論理和を求めます。結果(HFBFEBF7F)を倍長レジスタRW033・RW032に格納し、出力をONにします。

排 他 的 論 理 和

FUN 5 2 排他的論理和 (E O R)※

2つのレジスタデータのビット毎の排他的論理和を求めます。

関連命令
倍長排他的論理和

条件入力

実行出力

-[A EOR B → C]

機能

- オペランドAの内容とオペランドBの内容とのビット毎の排他的論理和を求めオペランドCに格納します。
- オペランドA, B, Cのデータ範囲はH0000~HFFFFです。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デバイス										レジスタ										INDEX			定数	修飾		
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K		
A	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
B	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
C	演算結果		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○	○	○	○		○

プログラム例

R0114 H23F2 H175A H34A8
[RW021 EOR RW030 → RW025]

動作

RW021

F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
0	0	1	0	0	0	1	1	1	1	1	1	0	0	1	0

(ビット毎の排他的論理和)

RW030

0	0	0	1	0	1	1	1	0	1	0	1	1	0	1	0

↓

RW025

0	0	1	1	0	1	0	0	1	0	1	0	1	0	0	0

- a接点R0114がONの時、レジスタRW021の内容(H23F2)とレジスタRW030の内容(H175A)のビット毎の排他的論理和を求めます。結果(H34A8)をレジスタRW025に格納し、出力をONにします。

倍長排他的論理和

FUN 53 倍長排他的論理和 (DEOR)*

2つの倍長レジスタデータのビット毎の排他的論理和を求めます。

関連命令
排他的論理和

条件入力

実行出力

—[A+1・A DEOR B+1・B → C+1・C]——

機能

- オペランドA+1・Aの内容とオペランドB+1・Bの内容のビット毎の排他的論理和を求め、オペランドC+1・Cに格納します。
- オペランドA+1・A、B+1・Bのデータ範囲はH00000000～HFFFFFFFです。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K		
A+1・A	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
B+1・B	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
C+1・C	演算結果		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○	○	○			○

プログラム例

R0102 H1234CDEF H2D1F0888
—[RW048・RW047 DEOR H3F2BC567 → D0253・D0252]——

動作

RW048

F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0

 ・ RW047

F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1

(ビット毎の論理積)

H3F2B

0	0	1	1	1	1	1	1	0	0	1	0	1	0	1	1

 ・ HC567

1	0	1	1	0	1	0	1	0	1	1	0	0	1	1	1

↓

D0253

0	0	1	0	1	1	0	1	0	0	0	1	1	1	1	1

 ・ D0252

0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0

- a接点R0102がONの時、倍長レジスタRW048・RW047の内容(H1234CDEF)とオペランドB+1・Bの定数(H3F2BC567)のビット毎の排他的論理和を求めます。結果(H2D1F0888)を倍長レジスタD0253・D0252に格納し、出力をONにします。

FUN 5 4 否定排他的論理和 (ENR) *

2つのレジスタデータのビット毎の否定排他的論理和を求めます。

倍長否定排他的論理和

実行出力

— [A ENR B → C] —

機能

- オペランドAの内容とオペランドBの内容とのビット毎の否定排他的論理和を求めオペランドCに格納します。
- オペランドA, Bのデータ範囲はH0000~HFFFFです。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾				
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K		
A	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○	
B	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
C	演算結果		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○		○	○	○	○	○

プログラム例

R0115 H23F8 H3528 HE92F
| | | RW031 ENR RW030 → RW026 |

動作

[illegible]

(ビット毎の否定排他的論理和)

RW030	0	0	1	1	0	1	0	1	0	0	1	0	1	0	0	0

↓

RW026 1 1 1 0 1 0 0 1 0 0 1 0 1 1 1 1

- a 接点 R0115 が ON の時、レジスタ RW031 の内容 (H23F8) と レジスタ RW030 の内容 (H3528) のビット毎の否定排他的論理和を求めます。結果 (HE92F) をレジスタ RW026 に格納し、出力を ON にします。

倍長否定排他的論理和

FUN 55 倍長否定排他的論理和 (DENR) *

2つの倍長レジスタデータのビット毎の否定排他的論理和を求めます。

関連命令
否定排他的論理和

条件入力

実行出力

—[A+1・A DENR B+1・B → C+1・C]—

機能

- ・オペランドA+1・Aの内容とオペランドB+1・Bの内容のビット毎の否定排他的論理和を求めオペランドC+1・Cに格納します。
- ・オペランドA+1・A, B+1・Bのデータ範囲はH00000000~HFFFFFFFです。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デバイス										レジスタ										INDEX			定数	修飾		
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A+1・A	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
B+1・B	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
C+1・C	演算結果		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○	○	○			○

プログラム例

R0104
H23625B7E DENR H7F86B93D → HA31B1DEC
RW022・RW021

動作

H2362

F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
0	0	1	0	0	0	1	1	0	1	1	0	0	0	1	0

 • H5B7E

F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
0	1	0	1	1	0	1	1	0	1	1	1	1	1	1	0

(ビット毎の否定排他的論理和)

H7F86

0	1	1	1	1	1	1	1	0	0	0	0	1	1	0

 • HB93D

1	0	1	1	1	0	0	1	0	0	1	1	1	1	0	1

↓

RW022

1	0	1	0	0	0	1	1	0	0	0	1	1	0	1	1

 • RW021

0	0	0	1	1	1	0	1	1	0	1	1	1	1	0	0

- ・ a 接点R0104がONの時、オペランドA+1・Aの定数(H23625B7E)とオペランドB+1・Bの定数(H7F86B93D)のビット毎の否定排他的論理和を求めます。結果(HA31B1DEC)を倍長レジスタRW022・RW021に格納し、出力をONにします。

移動平均

FUN 5 6	移動平均 (MAVE)	
時間的に連続するN個の入力データの平均値を算出します。		関連命令

入力	—[A MAVE (n) B → C]—	出力

機能

入力が ON のとき、A のレジスタのデータを B を先頭とする n ワードのレジスタに移動し、その平均値をレジスタ C に格納します。つまり、B を先頭とする n ワードのレジスタには n スキャン分の A レジスタのデータが格納されています。C+1 のレジスタには、その格納された中で最も新しいデータの位置を示すポインタが格納されています。ワード数 n の有効範囲は 1 ~ 64 です。

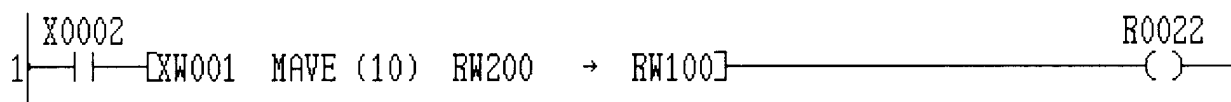
実行条件

入力	動作	出力
OFF	不実行	OFF
ON	実行	ON

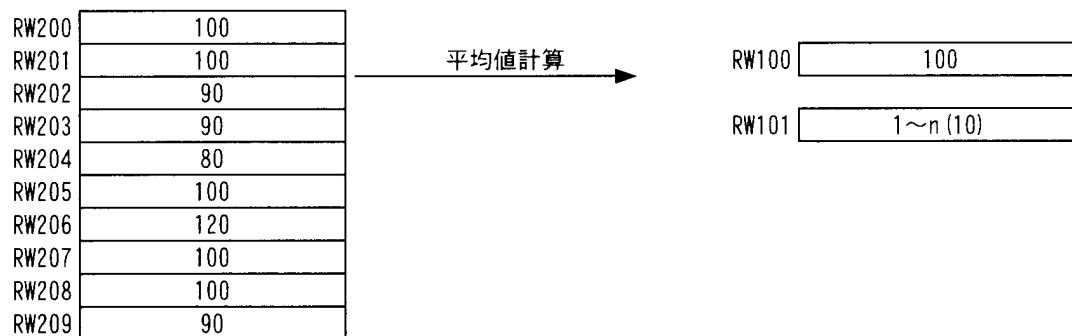
オペランド

		デ バ イ ス										レ ジ ス タ													定数	インデックス		
	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	入力データ											○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○
n	サイズ																											1 - 64
B	ワークテーブル											○	○	○	○	○	○	○	○	○	○	○	○					○
C	格納先												○	○	○	○	○	○	○	○	○	○		○	○	○		○

実行例



X0002 が ON のとき、RW200 を先頭とする 10 個のレジスタに最新の 10 スキャン分の XW001 のデータが格納され、その平均値が RW100 に、最も新しいデータの位置を示すポインタが RW101 に格納されます。



* オペランド C にインデックス K を使用するとバウンダリーエラーとなり実行しません。(出力 OFF)

* オペランド C + 1 (* インタ) 値は、n がテーブルの先頭、1 がテーブルの最後のレジスタを示します。

* T 1 S ではオペランド C にインデックスレジスタは使用できません。

[REDACTED]

FUN 57 テーブル論理積 (TAND) *

連続したレジスタのビット毎の論理積を求めます。

関連命令

Figure 1 illustrates a rule structure. It consists of a horizontal line representing the rule. On the left side, under the label '条件入力' (Condition Input), is the expression $[A \text{ TAND } (n) B \rightarrow C]$. On the right side, under the label '実行出力' (Execution Output), is the variable C .

Diagram illustrating a TAND gate (AND gate) configuration:

- 条件入力 (Condition Input):** Inputs A and B.
- 実行出力 (Execution Output):** Output C.
- Logic:** The output C is the result of the TAND operation on inputs A and B, represented as $A \text{ TAND } B \rightarrow C$.

機能

- ・オペランドAとBを先頭とするテーブルサイズnのレジスタについて、それぞれビット毎の論理積を求め、オペランドC以降のレジスタに格納します。
- ・テーブルサイズnの範囲は1～64です。

- 機能
- オペランド A と B を先頭とするテーブルサイズ n のレジスタについて、それぞれビット毎の論理積を求め、オペランド C 以降のレジスタに格納します。
 - テーブルサイズ n の範囲は 1 ～ 64 です。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	先頭レジスタ											○	○	○	○	○	○	○	○	○	○	○						○
n	テーブルサイズ																											○
B	先頭レジスタ											○	○	○	○	○	○	○	○	○	○	○						○
C	格納レジスタ												○	○		○	○	○	○	○	○		○					○

プログラム例

プログラム例

R0020	HF005	H235B	H2021
		RW023	TAND (03)
		RW016	- D1000]

動作

n	RW023	H F 0 3 5	AND	RW016	H 2 3 6 B	→	D1000	H 2 0 2 1
	RW024	H 0 0 F F	AND	RW017	H A A A A	→	D1001	H 0 0 A A
	RW025	H 5 5 5 5	AND	RW018	H 0 F 0 F	→	D1002	H 0 5 0 5

- ・ a 接点 R 0 0 2 C が ON の時、レジスタ R W 0 2 3 を先頭とする 3 ワードのテーブルと、レジスタ R W 0 1 6 を先頭とする 3 ワードのテーブルについて各レジスタのビット毎の論理積を求めます。演算結果はレジスタ D 1 0 0 0 を先頭とする 3 ワードのテーブルに格納し、出力を ON にします。

[illegible]

FUN 59 テーブル排他的論理和 (TEOR) *

連続するレジスタのビット毎の排他的論理和を求めます。

関連命令

条件入力	実行出力
$\neg [A \quad \text{TEOR} (n) \quad B \quad \rightarrow \quad C]$	_____

条件入力

実行出力

$\neg [A \text{ XOR } (n) \ B \rightarrow C]$ _____

条件入力

実行出力

$\neg [A \text{ XOR } (n) \ B \rightarrow C]$ _____

機能

- ・オペランドAとBを先頭とするテーブルサイズnのレジスタについて、それぞれビット毎の排他的論理和を求め、オペランドC以降のレジスタに格納します。
- ・テーブルサイズnの範囲は1～64です。

- 機能
- ・オペランドAとBを先頭とするテーブルサイズnのレジスタについて、それぞれビット毎の排他的論理和を求め、オペランドC以降のレジスタに格納します。
 - ・テーブルサイズnの範囲は1～64です。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K		
A	先頭レジスタ											○	○	○	○	○	○	○	○	○	○	○						○
n	テーブルサイズ																										○	
B	先頭レジスタ											○	○	○	○	○	○	○	○	○	○	○						○
C	格納レジスタ												○	○	○	○	○	○	○	○	○		○					○

オペランド		デバイス										レジスタ										INDEX			定数	修飾			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	先頭レジスタ											○	○	○	○	○	○	○	○	○	○	○							○
n	テーブルサイズ																											○	
B	先頭レジスタ											○	○	○	○	○	○	○	○	○	○	○							○
C	格納レジスタ												○	○	○	○	○	○	○	○	○		○						○

プログラム例

R002E	HF035	H236B	HD35E
└─┐	└─┐	└─┐	└─┐
CRW023	TEOR (03)	RW016	D1000

→

プログラム例

R002E	HF035	H236B	HD35E
└─┐	└─┐	└─┐	└─┐
CRW023	TEOR (03)	RW016	D1000

→

動作

n	RW023	H F 0 3 5	E O R	RW016	H 2 3 6 B	→	D1000	H D 3 5 E
	RW024	H 0 0 F F	E O R	RW017	H A A A A	→	D1001	H A A 5 5
	RW025	H 5 5 5 5	E O R	RW018	H 0 F 0 F	→	D1002	H 5 A 5 A

動作

n	RW023	H F 0 3 5	E O R	RW016	H 2 3 6 B	→	D1000	H D 3 5 E
	RW024	H 0 0 F F	E O R	RW017	H A A A A	→	D1001	H A A 5 5
	RW025	H 5 5 5 5	E O R	RW018	H 0 F 0 F	→	D1002	H 5 A 5 A

- 動作

n	RW023	H F 0 3 5	E O R	RW016	H 2 3 6 B	→	D1000	H D 3 5 E
	RW024	H 0 0 F F	E O R	RW017	H A A A A	→	D1001	H A A 5 5
	RW025	H 5 5 5 5	E O R	RW018	H 0 F 0 F	→	D1002	H 5 A 5 A

・ a 接点 R 0 0 2 E が O N の時、レジスタ R W 0 2 3 を先頭とする 3 ワードのテーブルと、レジスタ R W 0 1 6 を先頭とする 3 ワードのテーブルについて各レジスタのビット毎の排他的論理和を求めます。演算結果はレジスタ D 1 0 0 0 を先頭とする 3 ワードのテーブルに格納し、出力を O N にします。

FUN 60 テーブル否定排他的論理和 (TENR) *

連続するレジスタのビット毎の否定排他的論理和を求めます。

関連命令

条件入力

実行出力

$$-[A \text{ TENR } (n) \ B \ \rightarrow \ C] \text{-----}$$

機能

- ・オペランド A と B を先頭とするテーブルサイズ n のレジスタについて、それぞれビット毎の否定排他的論理和を求め、オペランド C 以降のレジスタに格納します。
- ・テーブルサイズ n の範囲は 1 ～ 64 です。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	先頭レジスタ											○	○	○	○	○	○	○	○	○	○	○	○					○
n	テーブルサイズ																											○
B	先頭レジスタ											○	○	○	○	○	○	○	○	○	○	○	○					○
C	格納レジスタ												○	○	○	○	○	○	○	○	○	○	○					○

プログラム例

```

| R002F HF035          H236B      H2CA1
| | | RW023 TENR (03)  RW016  → D1000]

```

動作

n	RW023	H F 0 3 5	E N R	RW016	H 2 3 6 B	→	D1000	H 2 C A 1
	RW024	H 0 0 F F	E N R	RW017	H A A A A	→	D1001	H 5 5 A A
	RW025	H 5 5 5 5	E N R	RW018	H 0 F 0 F	→	D1002	H A 5 A 5

- ・ a 接点 R002F が ON の時、レジスタ RW023 を先頭とする 3 ワードのテーブルと、レジスタ RW016 を先頭とする 3 ワードのテーブルについて各レジスタのビット毎の否定排他的論理和を求めます。演算結果はレジスタ D1000 を先頭とする 3 ワードのテーブルに格納し、出力を ON にします。

ディジタルフィルタ

FUN 6 1 デジタルフィルタ (DFL)

フィルタ定数により、入力データのフィルタリングを行います。

関連命令

入力 —[A DFL B → C]— 出力

機能

入力が ON のとき、A のレジスタのデータを B で指定するレジスタ値によりフィルタリングを行い、C のレジスタに格納します。B レジスタ値の有効範囲は 0 ~ 9999 です。C+1 のレジスタは内部演算用に使用され、初回実行時は 0 で実行後 1 が格納されます。C+1 レジスタ値が 0 の時、入力データがそのまま出力データとなり、0 以外の時はフィルタリングが実行されます。

実行条件

入力	動作	出力
OFF	不実行	OFF
ON	フィルタリング実行 ($0 \leq FL \leq 9999$)	ON
	入力値をそのまま出力する (レジスタ C+1=0 の時)	ON

オペランド

		デ バ イ ス										レ ジ ス タ													定数	インデックス			
	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	入力データ											○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
B	フィルタ定数											○	○	○	○	○	○	○	○	○	○	○	○			○	○		○
C	出力データ												○	○	○	○	○	○	○	○	○	○	○						○

実行例

1 | X0003 | R0022 |
 1 | | D0220 DFL D0100 → RW100 | () |

X0003 が ON のとき、下記の演算を行います。

$$Y0 = (1 - FL) * X0 + FL * Y-1$$

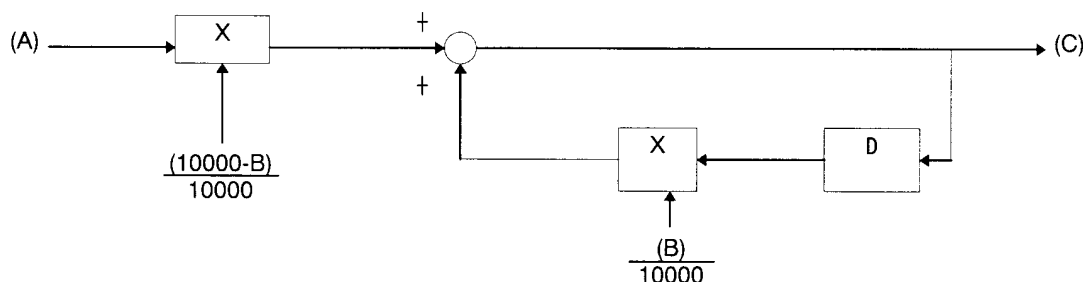
X0 : オペランド A で指定される入力データ

FL : オペランド B で指定されるフィルタ定数の 1/10000 の値

Y0 : オペランド C で指定される関数値 (出力データ)

Y-1 : 前回の関数値 (出力データ)

例えば、D0100 が 5000 の時、入力データ D0220 が 20 で前回の関数値が 10 の時、得られる関数値は 15 となり、RW100 には 15 が格納されます。尚、演算結果の小数点以下は切り捨てられます。



HEX → ASCII 変換

FUN 6 2 HEX → ASCII 変換 (HTOA)

16 進表現の数値をアスキーコードに変換します。

関連命令

ASCII → HEX 変換

入力 —[A HTOA (n) B]— 出力

機能

入力が ON のとき、A から始まる n ワードの 16 進データを ASCII 文字に変換し、B 以降に格納します。
変換は A の最上位桁から開始され、A の最上位桁が B の下位バイトに、という順番で行われます。
ワード数 n の有効範囲は 1 ~ 32 です。

実行条件

入力	動作	出力
OFF	不実行	OFF
ON	正常実行	ON

オペランド

		デ バ イ ス										レ ジ ス タ														定数	インデックス	
	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J			K
A	変換元											○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
n	サイズ																											1 - 32
B	格納先												○	○	○	○	○	○	○	○	○	○		○				○

実行例



R0010 が ON のとき、RW100 から RW103 の 4 ワードの 16 進データが、ASCII 文字に変換され、D0220 以降の 8 ワードに格納されます。

	F	0		F	8	7	0
RW100		H0123	→ HEX → ASCII 変換	D0220	"1" (H31)	"0" (H30)	
RW101		H4567		D0221	"3" (H33)	"2" (H32)	
RW102		H89AB		D0222	"5" (H35)	"4" (H34)	
RW103		HCDEF		D0223	"7" (H37)	"6" (H36)	
				D0224	"9" (H39)	"8" (H38)	
				D0225	"B" (H42)	"A" (H41)	
				D0226	"D" (H44)	"C" (H43)	
				D0227	"F" (H46)	"E" (H45)	

* オペランド A にインデックスレジスタまたは定数を使用した場合は n = 1 のみ実行できます。
(n > 1 のときはバウンダリーエラーとなり、実行しません。)

* T1S はオペランド A に I, J, K, 定数は使用できません。

ASCII → HEX 変換

FUN 63	ASCII → HEX 変換 (ATOH)	関連命令 HEX → ASCII 変換
アスキーコードの数字を 16 進の数値に変換します。		

入力 —[A ATOH (n) B]— 出力

機能

入力が ON のとき、A から始まる n ワードのレジスタに保存された ASCII 文字を 16 進データに変換し、B 以降に格納します。変換は A の下位バイトから開始され、A の下位バイトが B の最上位桁に、という順番で行われます。変換元として有効な ASCII 文字は、“0” (H30) ~ “9” (H39) 及び “A” (H41) ~ “F” (H46) の範囲です。ワード数 n の有効範囲は 1 ~ 64 です。

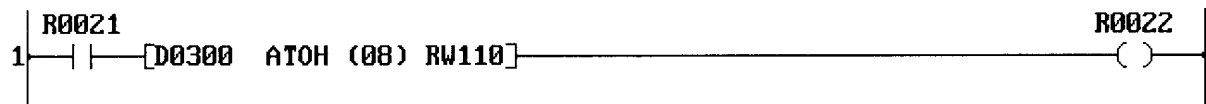
実行条件

入力	動作	出力	ERF
OFF	不実行	OFF	-
ON	正常実行	ON	-
	変換データエラー (変換不実行)	OFF	ON

オペランド

		デ バ イ ス										レ ジ ス タ															定数	インデックス	
	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	変換元											○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○	
n	サイズ																											1 - 64	
B	格納先												○	○	○	○	○	○	○	○	○		○						○

実行例



R0021 が ON のとき、D0300 から D0307 の 8 ワードのレジスタに保存された ASCII 文字 (16 文字) が、4 ワードの 16 進データに変換され、RW110 以降に格納されます。

	F	8	7	0		F	0
D0300	"1" (H31)			"0" (H30)	ASCII → HEX 変換 →	RW110	H0123
D0301	"3" (H33)			"2" (H32)		RW111	H4567
D0302	"5" (H35)			"4" (H34)		RW112	H89AB
D0303	"7" (H37)			"6" (H36)		RW113	HCDEF
D0304	"9" (H39)			"8" (H38)			
D0305	"B" (H42)			"A" (H41)			
D0306	"D" (H44)			"C" (H43)			
D0307	"F" (H46)			"E" (H45)			

* 変換元に無効データ (“0” ~ “F” 以外) がある場合には、変換は実行されず、出力 (R0022) OFF となります。また、変換ワード数が奇数のときには、変換後の最終データの下位 2 桁は不定となります。

	F	8	7	0		F	0
D0300	"1" (H31)			"0" (H30)	ASCII → HEX 変換 1ワード →	RW110	H01XX (X は不定)

必要に応じて偶数ワード数変換して下さい。

* オペランド A にインデックスレジスタまたは定数を使用した場合は n = 1 のみ実行できます。
(n > 1 のときはバウンダリーエラーとなり、実行しません。)

* T 1 S はオペランド A に I, J, K, 定数は使用できません。

テスト

FUN 64 テスト (TEST) *

2つのレジスタデータのビット毎の論理積を求め0かどうか判定します。

関連命令
倍長テスト

条件入力

判定出力

—[A TEST B]—

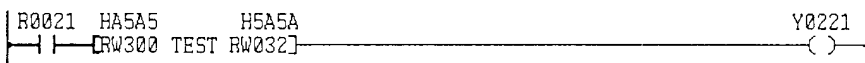
機能

- オペランドAの内容とオペランドBの内容とのビット毎の論理積を求め、結果が0以外のときは出力をONにし、0のときは出力をOFFにします。
- 演算データ及びテストデータの内容は本命令を実行しても変化しません。
- オペランドA・Bのデータ範囲はH0000~HFFFFです。

入力	処 理	出力
OFF	不実行	OFF
ON	実行：演算結果≠0	ON
	実行：演算結果=0	OFF

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定	修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
A	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
B	テストデータ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○

プログラム例 1 

動作

RW300

F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1

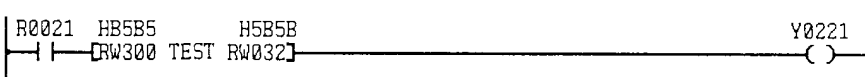
(ビット毎の論理積)

RW032

0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0

出力OFF (結果がH0000のため)

- a接点R0021がONの時、レジスタRW300の内容(HA5A5)とレジスタRW032の内容(H5A5A)のビット毎の論理積を求めます。結果がH0000になるので出力(Y0221)をOFFにします。

プログラム例 2 

動作

RW300

F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
1	0	1	1	0	1	0	1	1	0	1	1	0	1	0	1

(ビット毎の論理積)

RW032

0	1	0	1	1	0	1	1	0	1	0	1	1	0	1	1

出力ON (結果がH1111のため)

- a接点R0021がONの時レジスタRW300の内容(HB5B5)とレジスタRW032の内容(H5B5B)のビット毎の論理積を求めます。結果がH1111になるので出力(Y0221)をONにします。

© 2006 The Authors
Journal compilation © 2006 Blackwell Publishing Ltd

FUN 66 ビットファイル・ビットテスト (TTST)

ビットテーブルの特定の1ビットのON、OFF状態を調べます。

関連命令

条件入力

判定出力

— [A TTST (n) B] —

機能

- ・オペランドBを先頭データとするテーブルサイズnのデータの
中から、オペランドA（ビットポイント）で示すビットのON
／OFF状態を調べます。ビットが1の時は出力をONに、0
の時は出力をOFFにします。
- ・ビットポイントがテーブル範囲外の時はテストは行わず、出力
をOFFにします。
- ・テーブルサイズは、最大64ワードまでです。
($1 \leq n \leq 64$)

入力	処 理		出力
OFF	不実行		OFF
0 N	ビットポインタ 正常時（テーブル 範囲内）	ビット = 1	0 N
		ビット = 0	OFF
	ビットポインタ オーバー時	不実行	OFF

オペランド

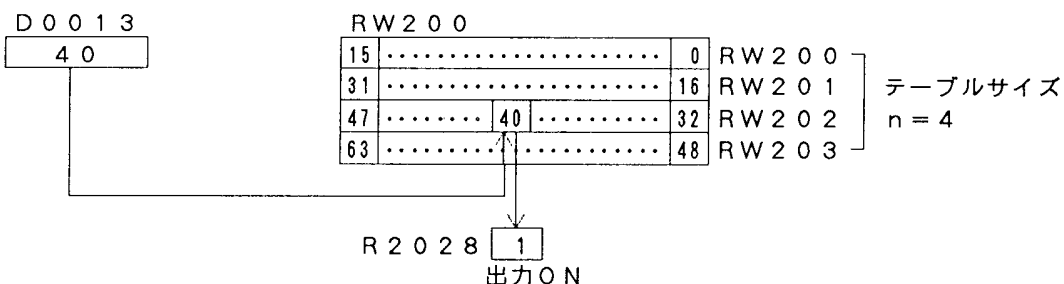
		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A	ビットホ・インタ											○	○		○	○	○	○	○	○	○	○	○		○	○	○	○	○
n	テープ・ルサイズ																											○	○
B	先頭レジスタ												○	○	○	○	○	○	○	○	○	○	○						○

プログラム例



動作

- ・ a 接点 R0208 が ON の時、RW200 を先頭とする 4 ワードのテーブルサイズのデータの中から、D0013 (0040) で示されるビットポインタ NO. 40 のビットの状態を調べます。対象となるビット (R2028) が 1 のため、出力 R0218 を ON にします。
- ・ a 接点 R0208 が OFF の時はテストを行わず出力を OFF にします。



補足

- データテーブルの先頭ビットはビットポインタNO. 0で表すものとします ($0 \leq \text{ビットポインタNO.} \leq n \times 16 - 1$)。例題のようにビットポインタNO. 40のビットは、テーブルの先頭から41番目のビットになります。また、例題でビットポインタNO. が64以上になるとビットポインタオーバーとなります。

1 ビット左シフト

FUN 6 9 1ビット左シフト (SHL 1) *

レジスタデータを1ビット左にシフトします。

関連命令

1ビット右シフト

条件入力

最上位ビット判定出力

—[SHL1 A]—

機能

- オペランドAのレジスタデータを、キャリーフラグを含め1ビット左にシフトします。
- シフト前の最上位ビットはキャリーフラグに格納されます。
- シフト後の最下位ビットには0が格納されます。
- シフト後の最上位ビットが1のときは出力をONにし、0のときは出力をOFFにします。
- シフト結果はオペランドAに再格納されます。

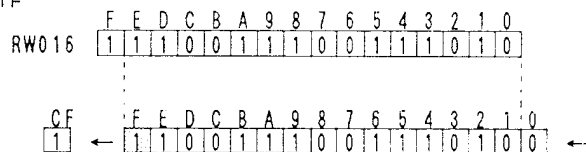
入力	処 理	出力
OFF	不実行	OFF
ON	実行：最上位ビット=1	ON
	実行：最上位ビット≠1	OFF

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定	修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	G	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
A	演算データ		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○	○	○	○			○

プログラム例 1 | x0202 | HCE74 | Y001A |
 |-----|-----|-----|
 |-----[SHL1RW016]-----|

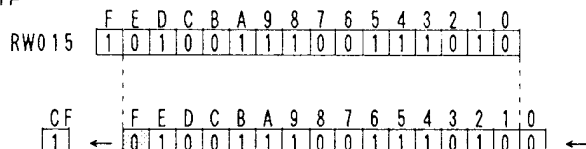
動作



- a接点X0202がONの時、レジスタRW016の内容(HE73A)を1ビット左にシフトします。結果(HCE74)をレジスタRW016に再格納します。
- シフト結果の最上位ビットが1なので出力をONにします。

プログラム例 2 | X0202 | H4E74 | Y001A |
 |-----|-----|-----|
 |-----[SHL1RW016]-----|

動作



- a接点X0202がONの時、レジスタRW016の内容(HA73A)を1ビット左にシフトします。結果(H4E74)をレジスタRW016に再格納します。
- シフト結果の最上位ビットが0なので出力をOFFにします。

n ビット右シフト

FUN 7 0 nビット右シフト (SHR) *

レジスタデータをnビット右にシフトします。

関連命令

nビット左シフト

条件入力

最下位ビット判定出力

—[A SHR n → B]—

機能

- ・オペランドAのレジスタデータを、キャリーフラグを含めnビット右にシフトします。
- ・シフト前の最下位からnビット目をキャリーフラグに格納します。
- ・シフト後の最上位ビットからnビット分には0を格納します。
- ・シフト後の最下位ビットが1であれば出力をONにし、0であれば出力をOFFにします。
- ・シフト結果はオペランドBに格納されます。
- ・オペランドAのデータ範囲はH0000~HFFFFです。
- ・オペランドnの範囲は1~16です。

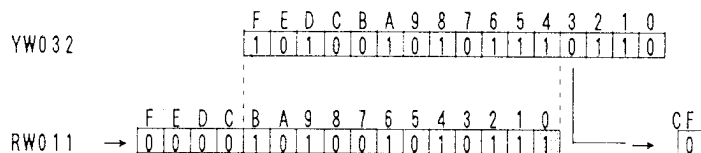
入力	処 理	出力
OFF	不実行	OFF
ON	実行：最下位ビット=1	ON
	実行：最下位ビット≠1	OFF

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定	修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
A	シフトソースデータ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○	○
n	シフトビット数																											○	
B	演算データ		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○	○	○	○	○	○		○

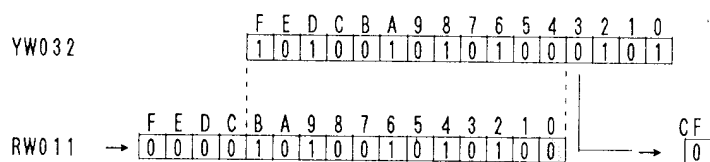
プログラム例 1 R023F HA576 YW032 SHR 04 → H0A57 RW011 Y001A ()

動作



- ・a接点R023FがONの時、レジスタYW032の内容(HA576)を4ビット右にシフトします。結果(H0A57)をレジスタRW011に格納します。
- ・シフト結果の最下位ビットが1なので出力をONにします。

動作



- ・ a 接点 R 0 2 3 F が O N の時、レジスタ Y W 0 3 2 の内容 (H A 5 4 5) を 4 ビット右にシフトします。演算結果 (H 0 A 5 4) はレジスタ R W 0 1 1 に格納します。
- ・ シフト結果の最下位ビットが 0 なので出力を O F F にします。

- ・シフト結果の最下位ビットが0なので出力を0FFにします。

n ビット左シフト

FUN 7 1 nビット左シフト (SHL*)

レジスタデータをnビット左にシフトします。

関連命令
nビット右シフト

条件入力

最上位ビット判定出力

[A SHL n → B]

機能

- オペランドAのレジスタデータを、キャリーフラグを含めnビット左にシフトします。
- シフト前の最上位からnビット目をキャリーフラグに格納します。
- シフト後の最下位ビットからnビット分には0を格納します。
- シフト後の最上位ビットが1のときは出力をONにし、0のときは出力をOFFにします。
- シフト結果はオペランドBに格納されます。
- オペランドAのデータ範囲はH0000~HFFFFです。
- オペランドnの範囲は1~16です。

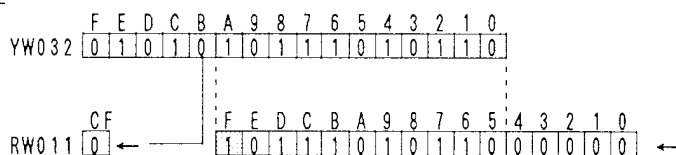
入力	処 理	出力
OFF	不実行	OFF
ON	実行：最下位ビット=1	ON
	実行：最下位ビット≠1	OFF

オペランド

		デバイス										レジスタ										INDEX			定数	修飾			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	シフトソースデータ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○		○
n	シフトビット数																											○	
B	演算データ		○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○		○	○	○	○			○

プログラム例 1 | X0203 H65D6 | HBAC0 | Y001A |
| | YW033 SHL 05 → RW012 | () |

動作

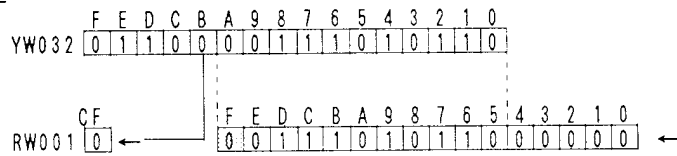


- a 接点 X0203 が ON の時、レジスタ YW033 の内容 (H65D6) を 5 ビット左にシフトします。結果 (HBAC0) をレジスタ RW012 に格納します。
- シフト結果の最上位ビットが 1 なので出力を ON にします。

プログラム例 2

X0203	H61D6	SHL 05	→ RW012	H3AC0	Y001A

動作



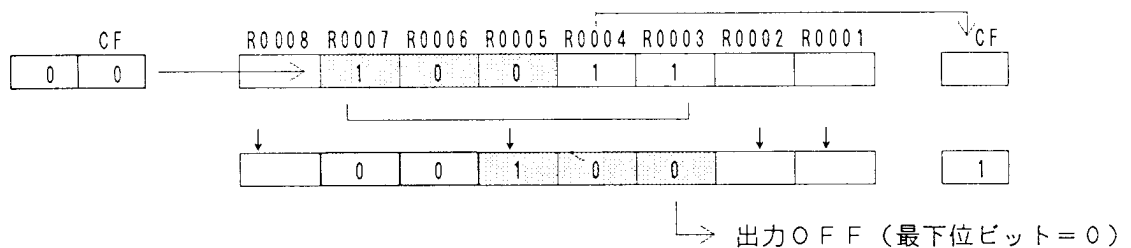
- ・ a 接点 X 0 2 0 3 が O N の時、レジスタ Y W 0 3 3 の内容 (H61D6) を 5 ビット左にシフトします。
結果 (H3AC0) をレジスタ R W 0 1 2 に格納します。
- ・ シフト結果の最上位ビットが 0 なので出力を O F F にします。

プログラム例 2
オペランド B をデバイスで指定した場合



動作

- a 接点 X 0 1 0 2 が O N の時、デバイス R 0 0 0 3 を先頭とするテーブルサイズ 5 のデバイスファイルをレジスタ Y W 0 0 3 の内容である 2 ビット分右にシフトします。
- シフトした最下位ビット R 0 0 0 3 が 0 になるため出力を O F F にします。
- シフト実行により最後にシフトアウトされるデバイス R 0 0 0 4 の内容 (1) はキャリーフラグに格納されます。



補足

- シフトの対象となるワード/ビット数がテーブルの有効範囲を越えた場合は、エラーフラグをセットします。(テーブルオペレーションエラー)
- 本命令はテーブル先頭番地にたいして桁指定ができません。
- ワード単位のシフトのときは、命令実行によりキャリーフラグは変化しません。
- シフト後の最上位ビットから n ビット分には 0 を格納します。

mビットファイルnビット左シフト

FUN 73 mビットファイルnビット左シフト (TSHL)*

連続したレジスタあるいはデバイスをnワードあるいはnビット左シフトします。

関連命令
mビットファイルnビット右シフト

条件入力

最上位ビット判定出力

[A TSHL (m) B]

機能

- オペランドBを先頭とするテーブルサイズmのワード/ビットファイルをオペランドAで示すワード/ビット数だけシフトします。シフト後、最上位ビットが1のときは出力をONにし、0のときは出力をOFFにします。
- テーブルサイズmの単位は、オペランドBがレジスタの場合はワードファイル、デバイスの場合はビットファイルになります。
- オペランドAの定数範囲は1～16、オペランドmの定数範囲は1～256です。

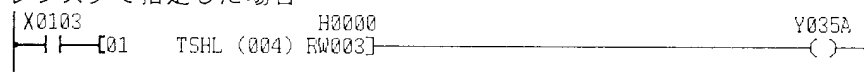
オペランド

入力	処 理	出力	CF	ERF
OFF	不実行	OFF		
ON	実行：最上位ビット=1	正	ON	
	実行：最上位ビット=0	常	OFF	
	シフト数>テーブルサイズ	NOP	OFF	

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A	シフト数											○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
m	テーブルサイズ																												○
B	先頭番地		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○					○

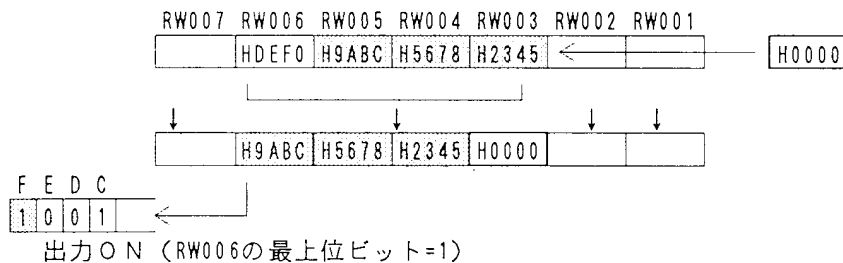
プログラム例 1

オペランドBをレジスタで指定した場合



動作

- a接点X0103がONの時、レジスタRW003を先頭とするテーブルサイズ4のワードファイルをオペランドAの内容である1ワード分左にシフトします。
- シフトの結果、最上位ワードRW006の内容(H7765)の最上位ビットが0になるので出力をOFFにします。



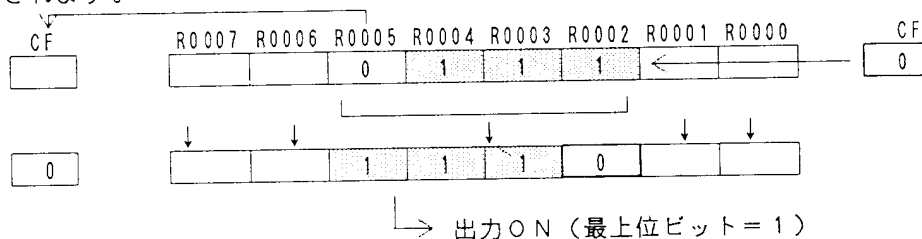
プログラム例 2

オペランド B をデバイスで指定した場合



動作

- ・ a 接点 X 0 1 0 4 が O N の時、デバイス R 0 0 0 2 を先頭とするテーブルサイズ 4 のデバイスファイルをオペランド A の内容である 1 ビット分左にシフトします。
- ・ シフトの結果、最上位ビット R 0 0 0 5 が 0 になるため出力を O F F にします。
- ・ シフト実行により最後にシフトアウトされるデバイス R 0 0 0 5 の内容 (0) はキャリーフラグに格納されます。



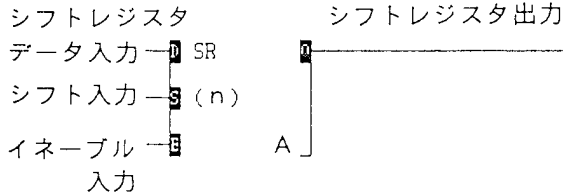
補足

- ・ オペランド A が 1 6 を超えた場合、またはシフト対象となるワード/ビット数がテーブルの有効範囲を超えた場合は、エラーフラグをセットします。(テーブルオペレーションエラー)
- ・ 本命令はテーブル先頭番地にたいして桁指定ができません。
- ・ ワード単位のシフトのときは、命令実行によりキャリーフラグは変化しません。
- ・ シフト後の最下位ビットから n ビット分には 0 を格納します。

FUN 7 4 シフトレジスタ (SR)

連続したデバイス (ビット) を 1 ビット左にシフトし、最上位ビットを出力します。

関連命令
双方向シフトレジスタ



機能

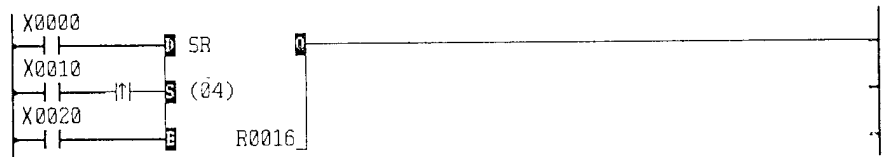
- ・イネーブル入力が ON のとき、シフト入力毎にオペランド A を先頭とした n 個のデバイス内容を 1 ビット左にシフトし、最上位ビットを出力します。
- ・イネーブル入力 OFF により n 個のデバイスの内容をクリアし、キャリーフラグもリセットします。
- ・シフト前の最上位ビットはキャリーフラグに格納されます。
- ・オペランド n の範囲は 1 ~ 64 です。

入 力	処 理	出 力	CF
E=OFF	対象となるデバイスをすべてクリア	OFF	○
E=ON	S=ON: シフト実行	デバイス (A+n-1)	○
	S=OFF: シフト不実行		

オペランド

		デバイス										レジスタ										INDEX			定数	修飾			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	先頭デバイス		○	○	○	○	○																						○
n	デバイスサイズ																												○

プログラム例



		R0019	R0018	R0017	R0016	
		1	1	0	1	← d
1 回目のシフト結果	CF	1	1	0	1	d1
2 回目のシフト結果		1	0	1	d1	d2

出力 = 1 (R0019)
出力 = 0 (R0019)

動作

- ・イネーブルが ON のとき、シフト入力毎に R0016 ~ R0019 と 1 ビットづつ左シフトします。
- ・1 回目のシフトでは出力は 1、キャリーフラグは 1 になります。
- ・2 回目のシフトでは出力は 0、キャリーフラグは 1 になります。
- ・d1、d2 にはシフト入力時のデータ入力の状態 X0000 が格納されます。

補足

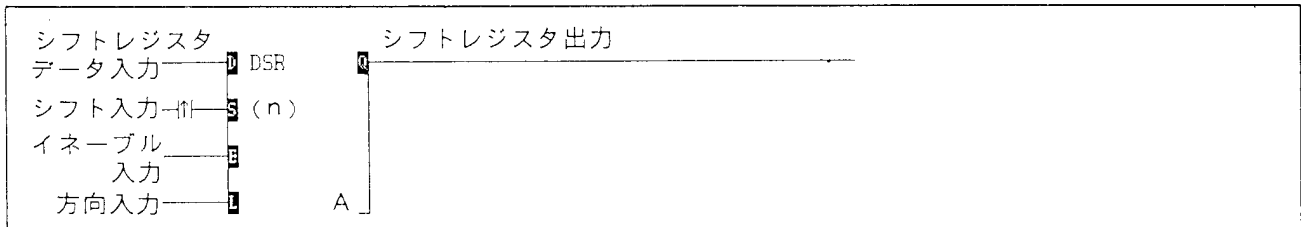
- ・シフト入力には微分接点をいれないとスキャン毎にシフトしてしまいますので注意してください。

双方向シフトレジスタ

FUN 75 双方向シフトレジスタ (DSR)

連結したデバイス（ビット）を1ビット左または右にシフトし、最上位ビットまたは最下位ビットを出力します。

関連命令
シフトレジスタ



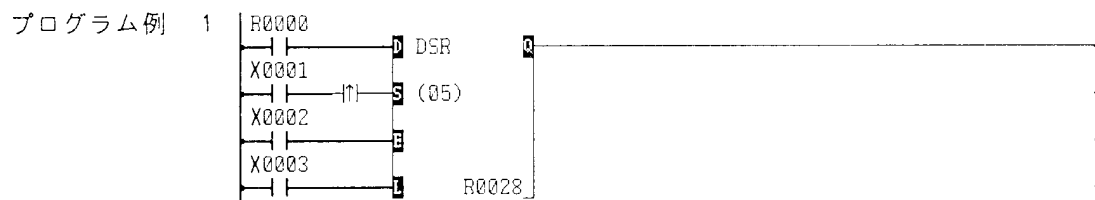
機能

- イネーブル入力 ON のとき、シフト入力毎にオペランド A を先頭とする n 個のデバイス（ビット）を1ビット右または左にシフトし、最上位ビット（左シフト時）あるいは最下位ビット（右シフト時）を出力します。
- 方向入力 ON で左シフト、OFF で右シフトします。
- イネーブル入力 OFF により、n 個のデバイスの内容をすべてクリアし、キャリーフラグもリセットします。
- オペランド n の範囲は 1 ～ 64 です。

入力	処 理	出 力	CF
E=OFF	シフト対象のデバイスをすべてクリア	OFF	○
E=ON	S=ON L=ON	1ビット左シフト実行	A+n-1 ○
	S=ON L=OFF	1ビット右シフト実行	A ○
	S=OFF	シフト不実行	A+n-1

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定	修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
A	先頭デバイス		○	○	○	○	○																						○
n	デバイスサイズ																												○



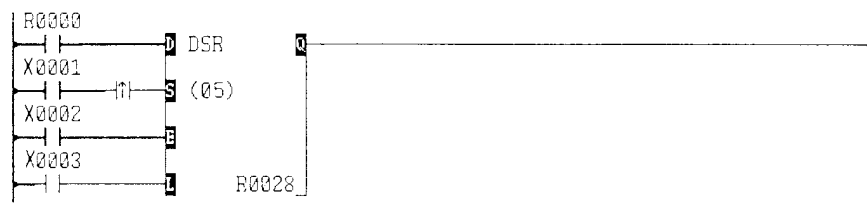
動作 (1 回目のシフト : L = 1 E = 1)
R002C R002B R002A R0029 R0028

CF	1	0	1	1	1	d1

出力 = 0 (R002C)

- シフト入力 ON により1ビット左シフトし、0を出力します。キャリーフラグには1が格納されます。
- R0028にはデータ入力の状態(d1)が格納されます。

プログラム例 2



動作 (2回目のシフト : L = 0 E = 1)

→

d2	0	1	1	1

CF
d1

 出力 = 1 (R 0 0 2 8)

- シフト入力 ONにより1ビット右シフトし、1を出力します。キャリーフラグにはd 1が格納されます。
- R 0 0 2 Cにはデータ入力の状態(d 2)が格納されます。

補足

- シフト入力には微分接点をいれないとスキャン毎にシフトしてしまいますので注意してください。

デバイスシフト

FUN 76 デバイスシフト (SFT)*

連続したデバイスの中から、特定の1デバイスだけを左シフトします。関連命令
(キャリーなし)

条件入力

実行出力

[SFT A]

機能

- ・オペランドA-1を1ビット左にシフトしオペランドAに格納します。
- ・オペランドA-1には0が格納されます。
- ・オペランドA-1が存在しない場合（オペランドAがR0000などの場合等）は、領域エラーとなりシフトは行われません。
- ・A、A-1以外のデバイスは変化しません。

入力	処 理	出力
OFF	不実行	OFF
ON	A-1 存在する : 実行	ON
	A-1 存在しない : 不実行	OFF

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	対象デバイス		○	○	○	○	○																					○

プログラム例



動作

R0102	R0101	R0100	R009F	R009E	
1	1	0	1	0	
		↑			
1	1	1	0	0	

実行前

実行後

- ・ a 接点R0002がONの時、R009F(1)を1ビット左にシフトしてR0100に格納します。R009Fには0を格納します。

ビット右ローテート

FUN 7 8 1ビット右ローテート (RTR1) *

レジスタのデータを1ビット右にローテートします。(キャリーなし)

関連命令

1ビット左ローテート

条件入力

最下位ビット判定出力

[RTR1 A] _____

機能

- オペランドAのレジスタをキャリーフラグを含めずに1ビット右にローテートします。ローテート前の最下位ビットが最上位ビットとキャリーフラグに格納されます。
- ローテートの結果、最下位ビットが1のときは出力をONにし、0のときは出力をOFFにします。

入力	処 理	出力	CF
OFF	不実行	OFF	
ON	実行：最下位ビット=1	ON	○
	実行：最下位ビット=0	OFF	○

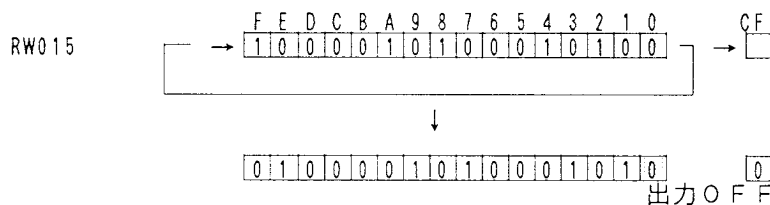
オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	L	RW	W	T	C	D	F	IW	OW	I			J	K
A	演算データ		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○	○	○

プログラム例

R0301 H428A Y001A
[RTR1RW015] ()

動作



- a 接点X0301がONの時、レジスタRW015 (H8514)を1ビット右ローテートします。
- ローテート前の最下位ビット(0)が最上位ビットとキャリーフラグに格納されます。
- ローテート結果 (H428A) の最下位ビットが0のため、出力をOFFにします。

1 ビット左ローテート

FUN 7 9 1 ビット左ローテート (RTL1) *

レジスタのデータを1ビット左にローテートします。(キャリーなし)

関連命令

1 ビット右ローテート

条件入力

最下位ビット判定出力

[RTL1 A]

機能

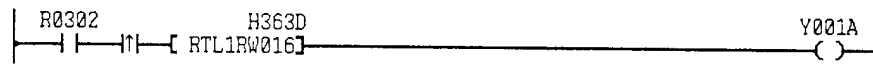
- オペランドAのレジスタをキャリーフラグを含めないで1ビット左にローテートします。ローテート前の最上位ビットが最下位ビットとキャリーフラグに格納されます。
- ローテートの結果、最下位ビットが1のときは出力をONにし、0のときは出力をOFFにします。

入力	処 理	出力	CF
OFF	不実行	OFF	
ON	実行：最下位ビット=1	ON	○
	実行：最下位ビット=0	OFF	○

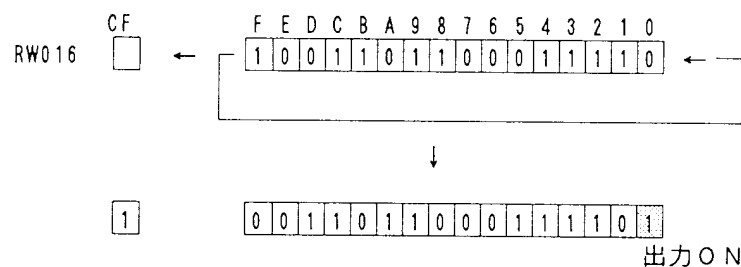
オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾	
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J
A	演算データ		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○	○	○	○	○

プログラム例



動作



- a 接点X0302がONの時、レジスタRW016 (H9B1E) を1ビット左ローテートします。
- ローテート前の最上位ビット(1)が最下位ビットとキャリーフラグに格納されます。
- ローテート結果 (H363D) の最下位ビットが1のため、出力をONにします。

© 2006 The Authors
Journal compilation © 2006 Blackwell Publishing Ltd

FUN 80 nビット右ローテート (RTR) *

レジスタのデータをnビット右にローテートします。(キャリーなし)	関連命令

条件入力 最下位ビット判定出力

$$- [A \text{ RTR } n \rightarrow B] \text{_____}$$

機能

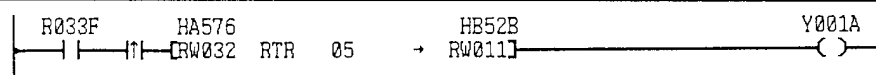
- 機能
- オペランドのアドレスを右ビットに格納し、右ビットに格納したデータを出力する。
 - ロード時に、アドレスの範囲内にあるデータは、オペランドの範囲内に格納される。

入力	処 理	出力	CF
OFF	不実行	OFF	
ON	実行：最下位ビット = 1	ON	○
	実行：最下位ビット = 0	OFF	○

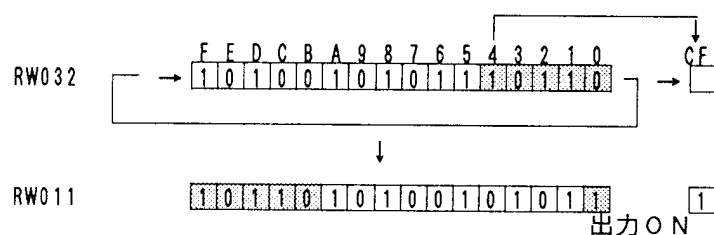
オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A	ソースデータ	○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
n	ローテート数																												○
B	演算結果		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○	○		

プログラム例



動作



- ・ a 接点 R 0 3 3 F が ON の時、レジスタ R W 0 3 2 (HA576) を 5 ビット右ローテートします。
ローテート前の 4 ビット目をキャリーフラグに格納します。
- ・ ローテート結果 (HB52B) を R W 0 1 1 に格納します。
- ・ ローテート結果の最下位ビットが 1 のため、出力を ON にします。

n ビット左ローテート

FUN 8 1 nビット左ローテート (RTL) *

レジスタのデータをnビット左にローテートします。(キャリーなし)

関連命令

nビット右ローテート

条件入力

最下位ビット判定出力

[A RTL n → B]

機能

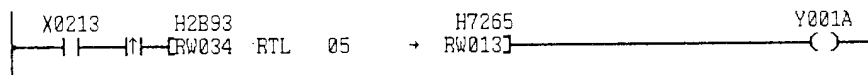
- オペランドAのレジスタデータをキャリーフラグを含めないでnビット左にローテートします。ローテート前の最上位ビットからnビット分をローテート後の最下位ビットからnビット分の位置に格納し、ローテート前の最上位からnビット目をキャリーフラグに格納します。
- ローテートの結果、最下位ビットが1のときは出力をONにし、0のときは出力をOFFにします。
- オペランドAの定数範囲はH0000~HFFFF、オペランドnの定数範囲は1~16です。
- ローテート結果は、オペランドBに格納されます。

入力	処 理	出力	CF
OFF	不実行	OFF	
ON	実行：最下位ビット=1	ON	○
	実行：最下位ビット=0	OFF	○

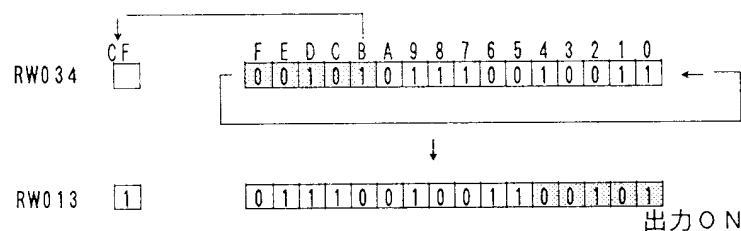
オペランド

		デ バ イ ス										レ ジ ス タ												INDEX			定 数	修 飾	
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	ソースデータ	○	○	○	○	○	○				○	○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
n	ローテート数																											○	
B	演算結果		○	○	○	○	○				○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○		○

プログラム例



動作



- a接点X0213がONの時、レジスタRW034(H2B93)を5ビット左ローテートします。ローテート前のBビット目をキャリーフラグに格納します。
- ローテート結果(H7265)をRW013に格納します。
- ローテート結果の最下位ビットが1のため、出力をONにします。

n ビット右ローテート

FNC 82 mビットファイルnビット右ローテート (TRTR)*

連続したレジスタあるいはデバイスをnワードあるいはnビット右にローテートします。(キャリーなし)

関連命令
mビット/ファイルn左シフト

条件入力

最下位ビット判定出力

←[A TRTR (m) B]

機能

- オペランドBを先頭とするテーブルサイズmのファイルをオペランドAで示す数だけキャリーフラグを含めないで右にローテートします。出力は 最下位ビットが1であればONにし、0であればOFFにします。
- テーブルサイズmの単位は、オペランドBがレジスタの場合はワードファイル、デバイスの場合はビットファイルになります。
- オペランドAの定数範囲は1～16、オペランドmの定数範囲は1～256です。

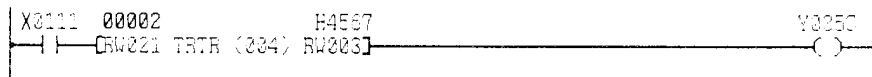
入力	処 理	出力	CF	ERF
OFF	不実行	OFF		
ON	実行：最下位ビット=1 正	ON	○	
	実行：最下位ビット=0 常	OFF		
	シフト数>テーブルサイズ	NOP	OFF	○

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定	修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
A	ローテート数											○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
m	テーブルサイズ																											○	
B	先頭番地		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○					○

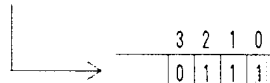
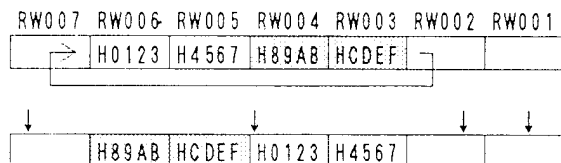
プログラム例 1

オペランドBをレジスタで指定した場合



機能

- a接点X0111がONの時、レジスタRW003を先頭とするテーブルサイズ4のワードファイルをレジスタRW021の内容である2ワード分を右にローテートします。
- ローテートした最下位ワードRW003の内容(H4567)の最下位ビットが1なので、出力をONにします。



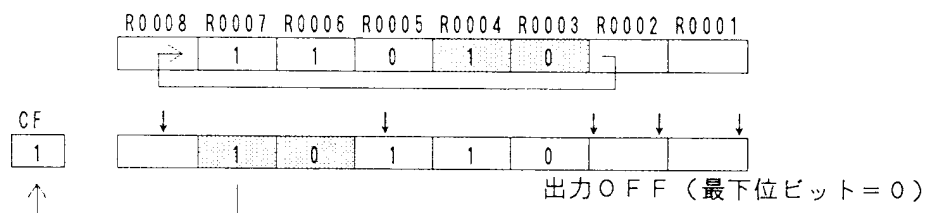
出力ON (RW003の最下位ビット=1)

プログラム例 2
オペランド B をデバイスで指定した場合



機能

- a 接点 X 0 1 1 2 が O N の時、デバイス R 0 0 0 3 を先頭とするテーブルサイズ 5 のデバイスファイルをレジスタ R W 0 0 4 の内容である 2 ビット分右にローテートします。
- ローテートした最下位ビット R 0 0 0 3 が 0 なので、出力を O F F にします。
- 最上位ビットにローテートされるデバイス R 0 0 0 4 (1) はキャリーフラグへ格納されます。



補足

- オペランド A が 1 6 を超えた場合、またはローテート対象となるワード/ビット数がテーブルの有効範囲を超えた場合は、エラーフラグをセットします。(テーブルオペレーションエラー)
- 本命令はテーブル先頭番地に対して桁指定ができません。
- ワード単位のローテートでは、命令実行によりキャリーフラグは変化しません。

n ビット左ローテート

FUN 83 mビットファイルnビット左ローテート (TRTL) *

連続したレジスタあるいはデバイスをnワードあるいはnビット左にローテートします。(キャリーなし)

関連命令

mビットファイルnビット右シフト

条件入力

最下位ビット判定出力

—[A TRTL (m) B]—

機能

- オペランドBを先頭とするテーブルサイズmのファイルをオペランドAで示す数だけキャリーフラグを含めないで左にローテートします。命令実行後、最下位ビットが1であれば出力をONにし、0であればOFFにします。
- テーブルサイズmの単位にはオペランドBがレジスタの場合はワードファイル、デバイスの場合はビットファイルになります。
- オペランドAの定数範囲は1～16、mの定数範囲は1～256です。

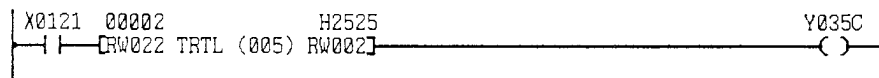
オペランド

入力	処 理	出力	CF	ERF
OFF	不実行	OFF		
ON	実行：最下位ビット = 1	正	ON	
	実行：最下位ビット = 0	常	OFF	
	シフト数 > テーブルサイズ	NOP	OFF	

		デ バ イ ス										レ ジ ス タ										INDEX			定	修				
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾		
A	ローテート数											○	○	○		○	○	○	○	○	○	○		○	○	○		○	○	
m	テーブルサイズ																											○		
B	先頭番地		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○					○	

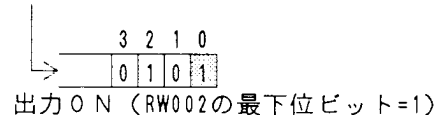
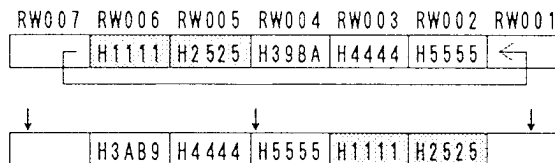
プログラム例 1

オペランドBをレジスタで指定した場合



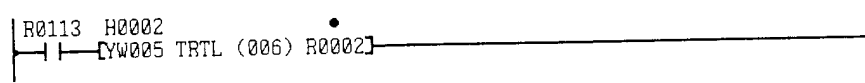
機能

- a接点X0121がONの時、レジスタRW002を先頭とするテーブルサイズ5のワードファイルをレジスタRW022の内容である2ワード分左にローテートします。
- ローテートの結果、最下位ワードRW002の内容(H2525)の最下位ビットが1となり、出力をONにします。



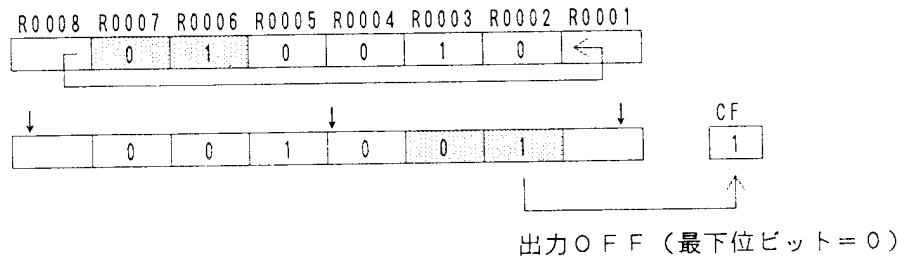
プログラム例 2

オペランド B をデバイスで指定した場合



機能

- ・ a 接点 X 0 1 1 3 が ON の時、デバイス R 0 0 0 2 を先頭とするテーブルサイズ 6 のデバイスファイルをレジスタ Y W 0 0 5 の内容である 2 ビット分左にローテートします。
- ・ ローテートした結果、最下位ビット R 0 0 0 2 が 1 となり、出力を ON にします。
- ・ 最下位ビットにローテートされるデバイス R 0 0 0 6 (1) はキャリーフラグに格納されます。



補定

- オペランド A が 16 を超えた場合、またはローテート対象となるワード/ビット数がテーブルの有効範囲を超えた場合は、エラーフラグをセットします。(テーブルオペレーションエラー)
- 本命令はテーブル先頭番地に対して桁指定ができません。
- ワード単位のローテートでは、命令実行によりキャリーフラグは変化しません。

キャリー付き 1 ビット左ローテート

FUN 85 キャリー付き 1 ビット左ローテート (RLC1)*

レジスタデータを 1 ビット左にローテートします。(キャリー付き)	関連命令 1 ビット右ローテート (キャリー付き)

条件入力	最下位ビット判定出力
[RLC1 A]	

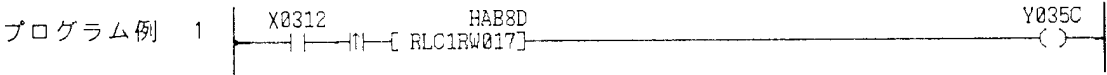
機能

- オペランド A のレジスタをキャリーフラグ付きで 1 ビット左にローテートします。ローテート前の最上位ビットがキャリーフラグに、ローテート前のキャリーフラグが最下位ビットにそれぞれ格納されます。
- ローテートの結果、最下位ビットが 1 のときは出力を ON、0 のときは出力を OFF にします。

入力	処 理	出力	CF
OFF	不実行	OFF	
ON	実行：最下位ビット = 1	ON	○
	実行：最下位ビット = 0	OFF	○

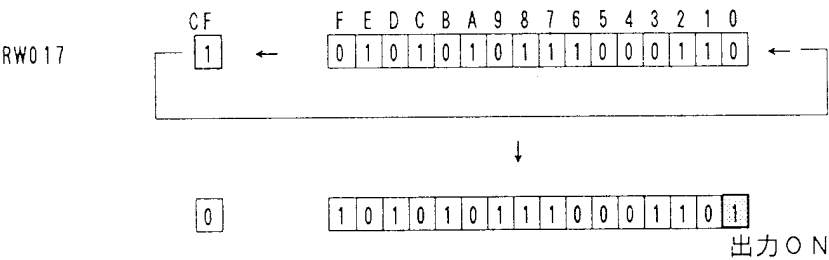
オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 修 数 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I		J	K
A	演算データ		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○	○	○	○	○



動作

- a 接点 X 0 3 1 2 が ON の時、レジスタ RW 0 1 7 (H55C6) を 1 ビット左ローテートします。
- ローテート前の最上位ビット (0) がキャリーフラグに、ローテート前のキャリーフラグ (1) が最下位ビットにそれぞれ格納されます。
- ローテート結果 (HAB8D) の最下位ビットが 1 になるため出力を ON にします。



関連命令
nビット左ローテート（キャリア付き）

最下位ビット判定出力

$$-[A \text{ RRC } n \rightarrow B] \text{-----}$$

- ・オペランドAをキャリーフラグ付きでnビット右にローテートします。
- ・オペランドAのデータ範囲はH0000~HFFFF、オペランドnのデータ範囲は1~16です。
- ・ローテート結果はオペランドBに格納されます。

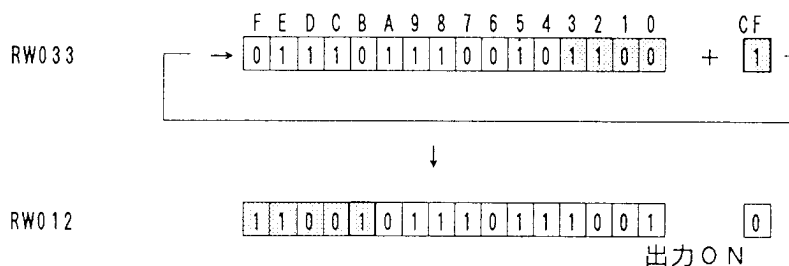
入力	処 理	出力	CF
OFF	不実行	OFF	
ON	実行：最下位ビット = 1	ON	○
	実行：最下位ビット = 0	OFF	○

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	ソースデータ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
n	ローテート数																										○	
B	演算結果		○	○	○	○	○				○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○

プログラム例 1

R0340 H772C RW033 RRC 05 HCB89 Y001A

- ・ a 接点 R 0 3 4 0 が ON の時、レジスタ R W 0 3 3 (H772C) をキャリーフラグを含めて 5 ビット右ローテートします。
ローテート前の 3 ~ 0 ビット分をローテート後の F ~ C ビット分に、ローテート前の 4 ビット目をローテート後のキャリーフラグに、ローテート前のキャリーフラグをローテート後の B ビット目に、それぞれ格納します。
- ・ ローテート結果 (HCB9) を R W 0 1 2 に格納します。
- ・ ローテート結果の最下位ビットが 1 になるため出力を ON にします。



キャリー付き n ビット左ローテート

FUN 87 キャリー付き n ビット左ローテート (RLC) *

レジスタデータを n ビット左にローテートします。
(キャリー付き)

関連命令

n ビット右ローテート (キャリー付き)

条件入力

最下位ビット判定出力

[A RLC n → B]

機能

- オペランド A のレジスタデータをキャリーフラグ付きで n ビット左にローテートします。
- オペランド A のデータ範囲は H 0 0 0 0 ~ H F F F F、オペランド n のデータ範囲は 1 ~ 1 6 です。
- ローテート結果はオペランド B に格納されます。

入力	処 理	出力	CF
OFF	不実行	OFF	
ON	実行：最下位ビット = 1	ON	○
	実行：最下位ビット = 0	OFF	○

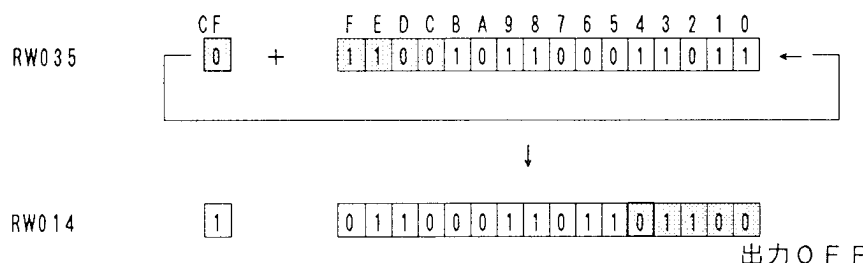
オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	ソースデータ	○	○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○
n	ローテート数																											○
B	演算結果		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○	○	○

プログラム例 1 $\xrightarrow{X0214} \xrightarrow{HCB1B} [RW035] \text{ RLC } 05 \rightarrow [RW014] \xrightarrow{H636C} \xrightarrow{Y035C} ()$

動作

- a 接点 R 0 2 1 4 が ON の時、レジスタ RW 0 3 5 (HCB1B) をキャリーフラグを含めて 5 ビット左ローテートします。
- ローテート前の F ~ C ビット分をローテート後の 3 ~ 0 ビット分に、ローテート前の B ビット目をローテート後のキャリーフラグに、ローテート前のキャリーフラグをローテート後の 4 ビット目に、それぞれ格納します。
- ローテート結果 (H636C) を RW 0 1 4 に格納します。
- ローテート結果の最下位ビットが 0 になるため出力を OFF にします。



■ キャリー付きmビットファイルnビット右ローテート

FUN 88 キャリー付きmビットファイルnビット右ローテート (TRRC) *

連続したレジスタあるいはデバイスをnワードあるいはnビット右にローテートします。(キャリー付き)

関連命令

mビットファイルn左シフト
(キャリー付き)

条件入力

最下位ビット判定出力

[A TRRC (m) B]

機能

- オペランドBを先頭とするテーブルサイズmのファイルをオペランドAで示す数だけキャリーフラグ付きで右にローテートします。命令実行後、最下位ビットが1のときは出力をONにし、0のときはOFFにします。
- テーブルサイズmの単位には、オペランドBがレジスタの場合はワードファイル、デバイスの場合はビットファイルになります。
- オペランドAのデータ範囲は1～16、オペランドmのデータ範囲は1～256です。

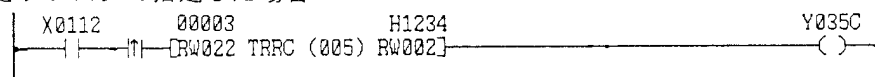
入力	処 理	出力	CF	ERF
OFF	不実行	OFF		
ON	実行：最下位ビット = 1	正	ON	
	実行：最下位ビット = 0	常	OFF	
	ローテート数 > テーブルサイズ	NOP	OFF	

オペランド

		デバイス										レジスタ										INDEX			定数	修飾			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	ローテート数											○	○	○	○	○	○	○	○	○	○	○			○	○	○		○
m	テーブルサイズ																											○	
B	先頭番地		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○					○

プログラム例 1

オペランドBをレジスタで指定した場合



動作

- a接点X0112がONの時、レジスタRW002を先頭とするテーブルサイズ5のワードファイルをレジスタRW022の内容である3ワード分右にローテートします。
- ローテートした最下位ワードRW002の内容(H1234)の最下位ビットが0になるため出力をOFFにします。

RW007 RW006 RW005 RW004 RW003 RW002 RW001

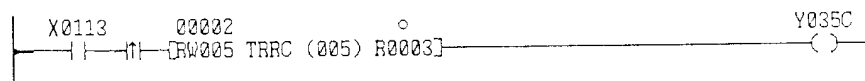
→ H0123 H1234 H2345 H3456 H4567

↓
H2345 H3456 H4567 H0123 H1234

3 2 1 0
→ 0 1 0 0

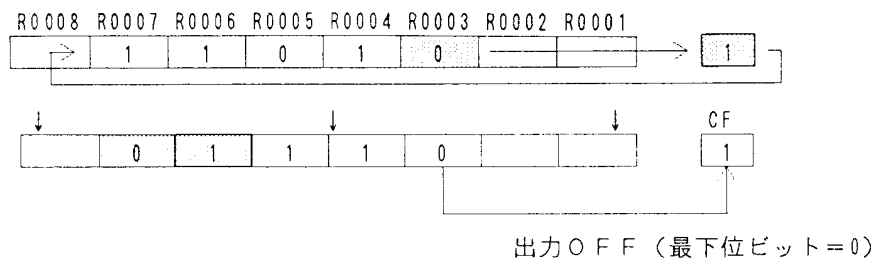
出力OFF (RW002の最下位ビット=0)

プログラム例 2
オペランド B をデバイスで指定した場合



動作

- a 接点 X 0 1 1 3 が O N の時、デバイス R 0 0 0 3 を先頭とするテーブルサイズ 5 のデバイスファイルレジスタ R W 0 0 5 の内容である 2 ビット分キャリーフラグを含めて右にローテートします。
- ローテート前のデバイス R 0 0 0 3 をローテート後のデバイス R 0 0 0 7 に、ローテート前のキャリーフラグをローテート後のデバイス R 0 0 0 6 に、ローテート前のデバイス R 0 0 0 4 をローテート後のキャリーフラグに、ローテート前のデバイス R 0 0 0 7 ~ R 0 0 0 5 をローテート後のデバイス R 0 0 0 5 ~ R 0 0 0 3 に格納します。
- ローテート後、ローテートした最下位ビット R 0 0 0 3 が 0 になるため出力を O F F にします。



補足

- オペランド A が 1 6 を超えた場合、またはローテート対象となるワード/ビット数がテーブルの有効範囲を超えた場合は、エラーフラグをセットします。(テーブルオペレーションエラー)
- 本命令はテーブル先頭番地に対して桁指定ができません。
- ワード単位のローテートでは、命令実行によりキャリーフラグは変化しません。

11

FUN 89 キャリー付きmビットファイルnビット左ローテート (TRL C)*

連続したレジスタあるいはデバイスを n ワードあるいは n ビット左にローテートします。(キャリー付き)

關連命令

mビットファイル n右シフト
(キャリー付き)

条件入力

最下位ビット判定出力

— [A TRLC (m) B] —

機能

- [illegible]

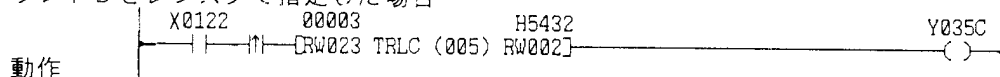
入力	処 理		出力	CF	ERF
OFF	不実行		OFF		
0 N	実行：最下位ビット = 1	正	0 N	○	
	実行：最下位ビット = 0	常	OFF		
	ローテート数 > テーブルサイズ		NOP		OFF

オペランド

		デ バ イ ス										レ ジ ス タ												INDEX			定	修	
opr	名 称	X	Y	S	L	R	Z	T	O	I	O	KW	W	SW	W	RW	W	T	C	D	F	W	OW	I	J	K	数	飾	
A	ローテート数											○	○	○	○	○	○	○	○	○	○	○		○	○	○		○	
m	テーブルサイズ																											○	
B	先頭番地		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○	○						○

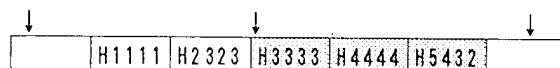
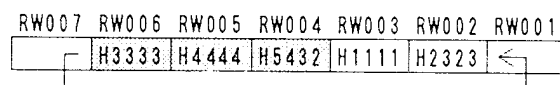
プログラム例 1

オペランドBをレジスタで指定した場合



動作

- ・ a 接点 X0122 が ON の時、レジスタ RW002 を先頭とするテーブルサイズ 5 のワードファイルレジスタ RW023 の内容である 3 ワード分左にローテートします。
- ・ ローテートした最下位ワード RW002 の内容 (H5432) の最下位ビットが 0 になるため出力を OFF にします。

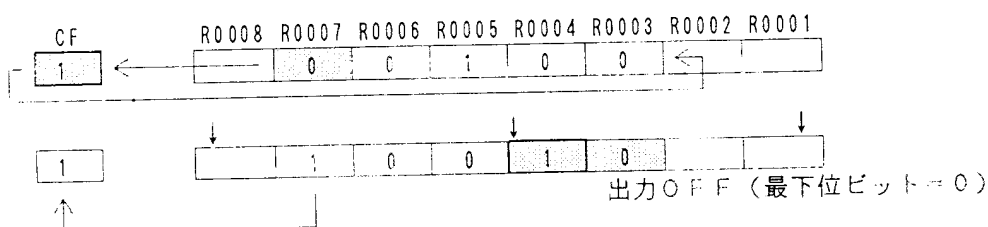


プログラム例 2
オペランド B をデバイスで指定した場合

機能



- a 接点 X 0 1 2 3 O N により、デバイス R 0 0 0 3 を先頭とするテーブルサイズ 5 のデバイスファイルをレジスタ YW 0 0 6 の内容である 2 ビット分キャリーフラグを含めて左にローテートします。
- ローテート前のデバイス R 0 0 0 7 をローテート後のデバイス R 0 0 0 3 に、ローテート前のキャリーフラグをローテート後のデバイス R 0 0 0 4 に、ローテート前のデバイス R 0 0 0 6 をローテート後のキャリーフラグに、ローテート前のデバイス R 0 0 0 5 ~ R 0 0 0 3 をローテート後のデバイス R 0 0 0 7 ~ R 0 0 0 5 に格納します。
- ローテート後、ローテートした最下位ビット R 0 0 0 3 が 0 になるため出力を O F F にします。



補足

- オペランド A が 1 6 を超えた場合、またはローテート対象となるワード/ビット数がテーブルの有効範囲を超えた場合は、エラーフラグをセットします。(テーブルオペレーションエラー)
- 本命令はテーブル先頭番地に対して桁指定ができません。
- ワード単位のローテートでは、命令実行によりキャリーフラグは変化しません。

マルチプレクサ

FUN90 マルチプレクサ (MPX)*

連続するレジスタデータの中から任意のレジスタのデータを特定のレジスタに転送します。マルチプレクサとして使用できます。

関連命令
デマルチプレクサ

条件入力

テーブルポインタオーバー出力

—[A MPX (n) B → C]——

機能

- オペランドAを先頭とするテーブルサイズnのレジスタデータの
中から、オペランドB (テーブルポインタNO.) で示されるレ
ジスタの内容をレジスタCに格納します。
- テーブルポインタがテーブルの範囲外の時は転送は行わず、出力
をONにします。
- テーブルサイズは、最大64ワードまでです。(1 ≤ n ≤ 64)

入力	処 理		出力
OFF	不実行		OFF
ON	正常時	実行	OFF
	テーブルポインタ オーバー時	不実行	ON

オペランド

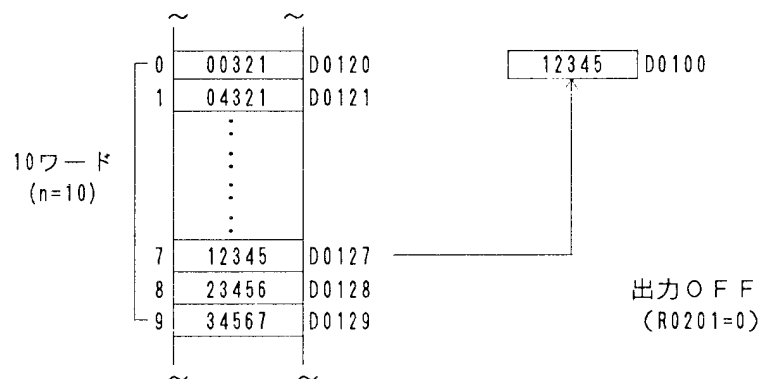
		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K		
A	先頭レジスタ											○	○	○	○	○	○	○	○	○	○	○						○
n	テーブルサイズ																											○
B	テーブルポインタ											○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
C	転送先レジスタ												○	○	○	○	○	○	○	○	○	○		○	○	○	○	○

プログラム例

X0070 00321 00007 12345 R0201
└─┬─┐ └─┬─┐ └─┬─┐ └─┬─┐ └─┬─┐
D0120 MPX (10) RW008 → D0100 ()

動作

- a接点X0070がONの時、レジスタD0120を先頭とする10ワードのデータの中
から、XW008 (0007) で示されるテーブルポインタNO. 7のレジスタD012
7の内容 (12345) をD0100に格納します。テーブルポインタがテーブル範囲内
で正常にデータ転送が行われるため、出力R0201はOFFになります。
- a接点X0070がOFFの時はデータ転送は行わず、出力をOFFにします。



補足

- テーブルの先頭レジスタは、テーブルポインタNO. 0で表すものとします (0 ≤ テーブルポインタNO. ≤ n-1)。例題のようにテーブルポインタNO. 7のレジスタはテーブルの先頭から8番目のレジスタになります。また、例題でテーブルポインタNO. が10以上になるとテーブルポインタオーバーとなります。
- テーブルは各レジスタ空間内の有効範囲内である必要があります。

デマルチプレクサ

FUN9 1 デマルチプレクサ (DPX) *

特定のレジスタのデータをデータテーブルの任意のレジスタに転送します。デマルチプレクサとして使用できます。

関連命令
マルチプレクサ

条件入力

テーブルポインタオーバー出力

— [A DPX (n) B → C] —

機能

- レジスタAの内容をレジスタCを先頭とするテーブルサイズnの中のオペランドB（テーブルポインタ）で示されるレジスタに格納します。
- テーブルポインタがテーブルの範囲外の時は転送は行わず、出力をOFFにします。
- テーブルサイズは、最大64ワードまでです。（ $1 \leq n \leq 64$ ）

入力	処 理	出力
OFF	不実行	OFF
ON	正常時	実行
	テーブルポインタオーバー時	不実行
		ON

オペランド

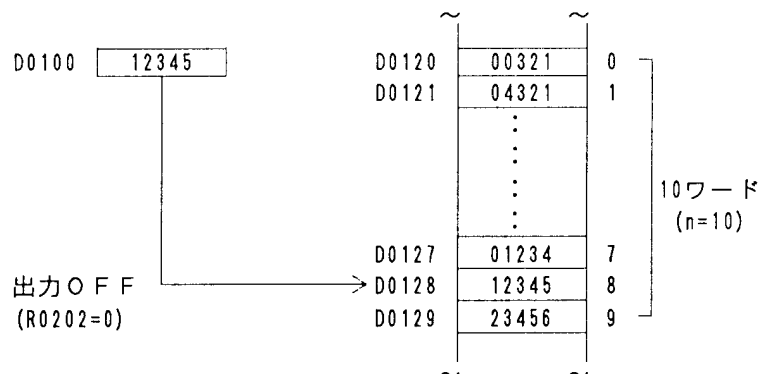
		デ バ イ ス										レ ジ ス タ										INDEX			定 修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数 飾	
A	転送レジスタ											○	○		○	○	○	○	○	○	○	○		○	○	○	○	○
n	テーブルサイズ																											○
B	テーブルポインタ											○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
C	先頭レジスタ												○	○	○	○	○	○	○	○	○		○					○

プログラム例

X0071 12345 00008 00321 R0202
D0100 DPX (10) XW008 → D0120 ()

動作

- a接点X0071がONの時、レジスタD0100の内容（12345）をレジスタD0120を先頭とするテーブルサイズ10のテーブルの中のXW008（0008）で示されるテーブルポインタNO. 8のレジスタD0128に格納します。テーブルポインタがテーブル範囲内で正常にデータ転送が行われるため出力R0202はOFFになります。
- a接点X0071がOFFの時はデータ転送は行わず、出力をOFFにします。



補足

- テーブルの先頭レジスタは、テーブルポインタNO. 0で表すものとします（ $0 \leq \text{テーブルポインタNO.} \leq n-1$ ）。例題のようにテーブルポインタNO. 8のレジスタはテーブルの先頭から9番目のレジスタになります。また、例題でテーブルポインタNO. が10以上になるとテーブルポインタオーバーとなります。
- テーブルは、各レジスタ空間内の有効範囲内である必要があります。

FUN 9 3 ビット→テーブル転送 (BTM)*

デバイスの内容を、レジスタテーブルの中の特定の1ビットに転送します。

関連命令
テーブル→ビット転送

条件入力

ビットポインタオーバー出力

—[A BTM (n) B → C]————

機能

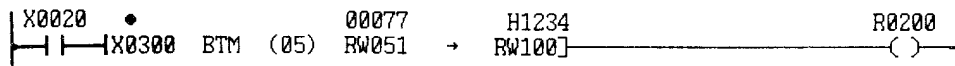
- オペランド A の内容をオペランド C を先頭とするテーブルサイズ n のデータ内の、オペランド B (ビットポイント) で示されるビットに転送します。
- ビットポイントがテーブルの範囲外の時は転送を行わず、出力 ON にします。
- テーブルサイズは、最大 64 ワードまでです。
($1 \leq n \leq 64$)

入力	処 理		出力
OFF	不実行		OFF
O N	正常時	実行	OFF
	ビットポイント オーバー時	不実行	O N

オペランド

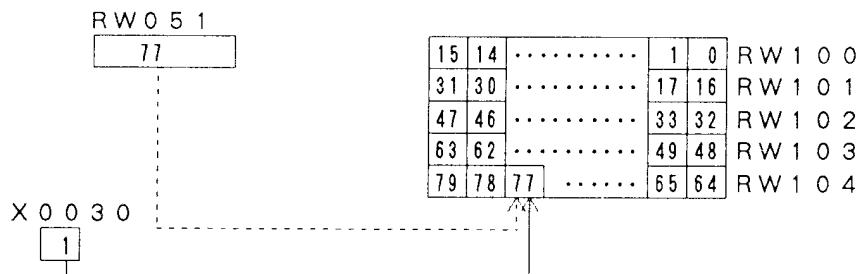
		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	転送デバイス	○	○	○	○	○	○			○																		○
n	テープルサイズ																											○
B	ビットホインタ											○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○
C	先頭レジスタ												○	○	○	○	○	○	○	○	○	○	○	○				○

プログラム例



動作

- ・ a 接点X0020がONの時、レジスタRW100を先頭としたテーブルサイズ5ワードのデータの中の、RW051で示されるビットポインタNO.77のビット(R104Dに、X0300の内容(1)を転送します。ビットポインタがテーブル範囲内(RW051の内容が79以下)で正常にビット転送が行われるため出力はOFFになります。
- ・ a 接点X0020がOFFの時はビット転送は行わず、出力をOFFにします。



補足

- ・テーブルの先頭ビットは、ビットポインタ NO. 0 で表すものとします ($0 \leq \text{ビットポインタ NO.} \leq n \times 16 - 1$)。例題のようにビットポインタ NO. 77 のビットはテーブルの先頭から 78 番目のビットになります。

FUN95 ビットファイル比較 (TCMP)

2組のレジスタテーブルのデータを比較し、不一致（箇所）ビットを示します。

関連命令

条件入力

不一致検出力

$$\neg [A \text{ TEMP}(n) \quad B \rightarrow C]$$

機能

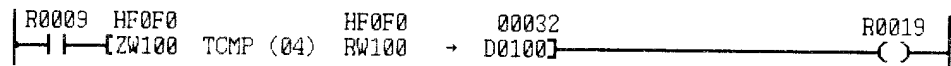
- ・オペランド A・B で指定される 2 組のテーブルサイズ n のデータを比較し、不一致のビットポイント NO. をオペランド C に格納します。オペランド C の内容 (ビットポイント) は実行ごとに更新されます (不一致箇所がない時は不変)。
- ・実行開始前にビットポイントがテーブル範囲外の時は、0 ビットから比較を行います。
- ・テーブルサイズは、最大 64 ワードまでです。
($1 \leq n \leq 64$)

入力	処 理		出力
OFF	不実行		OFF
ON	実行	不一致ビット有	ON
		不一致ビット無	OFF

オペランド

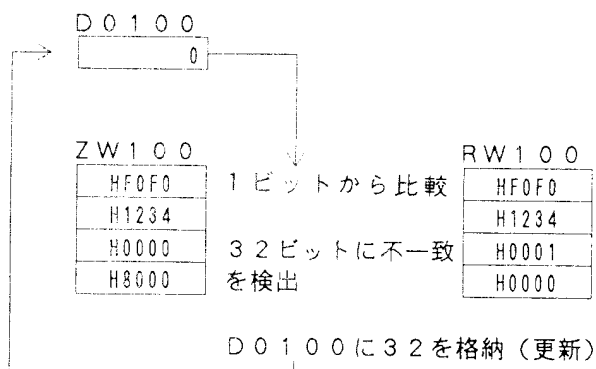
		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	先頭レジスタ											○	○	○	○	○	○	○	○	○	○							○
n	テーブルサイズ																											
B	先頭レジスタ											○	○	○	○	○	○	○	○	○	○							○
C	不一致 ビットポイント												○	○	○	○	○	○	○	○	○				○	○	○	○

プログラム例



動作

- ・a 接点 R 0 0 0 9 が ON の時、ZW 1 0 0 と RW 1 0 0 の各々を先頭とするテーブルサイズ 4 ワードのテーブルを比較します。不一致ビットポイントが 0 (初期値) のため、1 ビット目 (ビット NO. 0) から比較を始め、3 3 ビット目 (ビット NO. 3 2) のビットが不一致のため、3 2 を不一致ビットポイントとして D 0 1 0 0 に格納し、出力を ON にします。
- ・a 接点 R 0 0 0 9 が再び ON になると、3 3 ビット目 (ビット NO. 3 3) からテーブルを比較し、テーブルの最後まで比較したら先頭にもどって比較を行います。
- ・もとのポイントまで比較し (テーブルの全ビットを比較し)、不一致が見つからなかったときは実行を停止し出力を OFF にします。この時はビットポイントは変化しません。
- ・a 接点 R 0 0 0 9 が OFF の時は比較は行わず出力を OFF にします。



一 補足

- 不一致ビットポインタNO. はテーブルの先頭ビットを不一致ビットポインタNO. 0として表します ($0 \leq \text{不一致ビットポインタNO.} \leq n \times 16 - 1$)。例題のように不一致ビットポインタNO. 32は先頭から33ビット目のビットになります。

比較：より大きい

FUN9 6 比較：より大きい(>)

データを比較し>を検出します。

関連命令

条件入力

判定出力

-[A > B]-

機能

- ・オペランドA>オペランドBのとき出力をONにします。
- ・オペランドA、Bのデータ範囲は-32768~32767です。
- ・オペランドA及びBのデータはこの比較により変化しません。
(符号付きデータ比較)

入力	処 理		出力
OFF	不実行		OFF
ON	実行	$A > B$	ON
		$A \leq B$	OFF

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	比較値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○	○
B	基準値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○	○

プログラム例

X0123 23456
└─┬─┐ D0125 > 13495 ─┬─┐ R0001
└────────────────────────────────┘

動作

- ・a接点X0123がONの時、レジスタD0125のデータ(23456)とオペランドBのデータ(13495)を比較します。比較結果はオペランドA>オペランドBになるため出力の出力をONにします。

A:D0125
23456
B:
13495

→ 比較 : > → A > B

比較：より大きいまたは等しい

FUN 9 7 比較：より大きいまたは等しい ($\geq$)

データを比較し $\geq$ を検出します。

関連命令

判定出力

条件入力	判定出力
$\neg [A \geq B]$	

機能

- ・オペランドA $\geq$ オペランドB のとき出力を ON にします。
- ・オペランドA、B のデータ範囲は -32768 ~ 32767 です。
- ・オペランドA、及びB のデータはこの比較により変化しません。
(符号付きデータ比較)

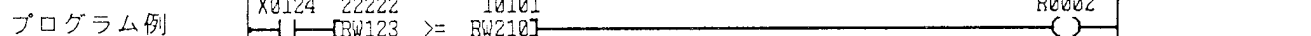
- 機能
- ・オペランドA $\geq$ オペランドB のとき出力を ON にします。
 - ・オペランドA、B のデータ範囲は -32768 ~ 32767 です。
 - ・オペランドA、及びB のデータはこの比較により変化しません。
(符号付きデータ比較)

入力	処 理	出力
OFF	不実行	OFF
ON	実行	$A \geq B$
		$A < B$

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A	比較値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
B	基準値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○

プログラム例



動作

- ・ a 接点X0124がONの時、レジスタRW0123のデータ(22222)とレジスタRW210のデータ(10101)を比較します。比較結果はオペランドA $\geq$ オペランドBとなるため出力をONにします。

- 動作
- ・ a 接点X0124がONの時、レジスタRW0123のデータ(22222)とレジスタRW210のデータ(10101)を比較します。比較結果はオペランドA $\geq$ オペランドBとなるため出力をONにします。

A: RW123
22222
B: RW210
10101

比較： $> = \longrightarrow A \geq B$

条件入力

判定出力

$\neg [A = B]$

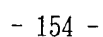
- ・オペランド A = オペランド B のとき出力を ON にします。
- ・オペランド A、B のデータ範囲は -32768 ~ 32767 です。
- ・オペランド A 及び B のデータはこの比較により変化しません。
(符号付きデータ比較)

オペランド

プログラム例



- ・ a 接点X0125がONの時、レジスタRW125のデータ(20234)とレジスタRW001のデータ(20234)を比較します。比較結果はオペランドA=オペランドBとなるため出力をONにします。



比較：等しくない

FUN 9 9 比較：等しくない (< >)

データを比較し≠を検出します。

関連命令

条件入力

判定出力

[A < > B]

機能

- オペランドA≠オペランドBのとき出力をONにします。
- オペランドA、Bのデータ範囲は-32768～32767です。
- オペランドA及びBのデータはこの比較により変化しません。
(符号付きデータ比較)

入力	処 理	出力
OFF	不実行	OFF
ON	A ≠ B	ON
	A = B	OFF

オペランド

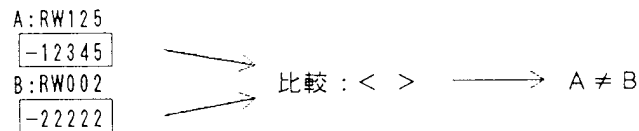
		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	比較値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
B	基準値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○

プログラム例

X0126	-12345	-22222	R0004
[X0126 <> R0002]			()

動作

- a 接点X0126がONの時、レジスタRW127のデータ(-12345)とレジスタRW002のデータ(-22222)を比較します。比較結果はオペランドA≠オペランドBとなるため出力をONにします。



比較：より小さい

FUN1 0 0 比較：より小さい(<)	
データを比較しくを検出します。	関連命令

条件入力	判定出力
{ A < B }	

- 機能
- ・オペランドA<オペランドBのとき出力をONにします。
 - ・オペランドA、Bのデータ範囲は-32768~32767です。
 - ・オペランドA及びBのデータはこの比較により変化しません。
(符号付きデータ比較)

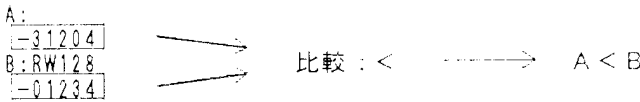
入力	処 理		出力
OFF	不実行		OFF
ON	実行	$A < B$	ON
		$A \geq B$	OFF

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	比較値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
B	基準値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○

プログラム例	X0127	-01234	R0005
	{-31204 < RW128}		

- 動作
- ・a 接点X0127がONの時、オペランドAのデータ(-31204)とレジスタRW128のデータ(-01234)を比較します。比較結果はオペランドA<オペランドBとなるため出力をONにします。



FUN 1 0 1 比較：より小さいまたは等しい (<=)	
データを比較し ≤ を検出します。	関連命令

Diagram illustrating a basic conditional statement structure:

```
graph LR; A[条件入力] --> B{判定出力}; B --> C[ ]; C --> D[ ];
```

The diagram shows a flow from a box labeled "条件入力" (Condition Input) to a box labeled "判定出力" (Decision Output). Below the "判定出力" box, there is a horizontal line with a bracket on the left side, indicating a loop or continuation point.

機能

- ・オペランド A ≤ オペランド B のとき出力を ON にします。
 - ・オペランド A、B のデータ範囲は -32768 ~ 32767 です。
 - ・オペランド A 及び B のデータはこの比較により変化しません。
- (符号付きデータ比較)

入力	処 理		出力
OFF	不実行		OFF
ON	実行	$A \leq B$	ON
		$A > B$	OFF

オペランド

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A	比較値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
B	基準値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○

プログラム例



Diagram illustrating a program example (プログラム例) using a ladder logic structure. The diagram shows a normally open contact labeled X0128 in series with a coil labeled Y0102. The coil is connected to a power rail labeled R0006. The diagram is enclosed in a rectangular frame.

動作

- ・ a 接点X0128がONの時、レジスタRW102のデータ(17326)とレジスタRW098のデータ(20129)を比較します。比較結果はオペランドA≦オペランドBとなるため出力をONにします。

A: RW102
17326
B: RW098
20129

比較： $< = \longrightarrow A \leq B$

倍長比較：より大きい

FUN 1 0 2 倍長比較：より大きい (D>)

倍長データを比較し>を検出します。

関連命令

条件入力

判定出力

[A+1・A D> B+1・B]

機能

- 2つの倍長データ $A+1 \cdot A$ 、 $B+1 \cdot B$ が $A+1 \cdot 1 > B+1 \cdot B$ のとき出力を ON にします。
- データ A、B のデータ範囲は -2147483648 ~ 2147483647 です。
- データ A、B のデータはこの比較により変化しません。
(符号付きデータ比較)

入力	処 理	出力
OFF	不実行	OFF
ON	実行	A > B
		A ≤ B
		ON
		OFF

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A+1・A	比較値	○	○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
B+1・B	基準値	○	○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○	○		○	○		○	○

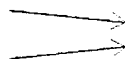
プログラム例

X0143 0385678521 0012345678 R0007
 ┌──┴──┐ ┌──┴──┐ D1234・D1233 D> RW025・RW024 ───────────┐ ┌──┴──┐

動作

- a 接点 X0143 が ON の時、倍長レジスタ D1234・D1233 のデータ (385678521) と倍長レジスタ RW025・RW024 のデータ (12345678) を比較します。比較結果はオペランド A > オペランド B となるため出力を ON にします。

A+1:D1234 A:D1233
 385678521
 F 0 F 0
 12345678
 B+1:RW025 B:RW024



比較：D> → A>B

倍長比較：より大きいまたは等しい

FUN 1 0 3 倍長比較：より大きいまたは等しい ($D \geq$)

倍長データを比較し $\geq$ を検出します。

関連命令

条件入力 判

判定出力

$$- [A+1 \cdot A \quad D] \geq B+1 \cdot B \quad]$$

機能

- ・ 2 つの倍長データが $A + 1 \cdot 1 \geq B + 1 \cdot B$ のとき出力を ON にします。
- ・ データ $A + 1 \cdot A$ 、 $B + 1 \cdot B$ のデータ範囲は -2147483648 ~ 2147483647 です。
- ・ データ A、B のデータはこの比較により変化しません。
(符号付きデータ比較)

入力	処 理		出力
OFF	不実行		OFF
ON	実行	$A + 1 \cdot A \geq B + 1 \cdot B$	ON
		$A + 1 \cdot A < B + 1 \cdot B$	OFF

オペランド

		デバイス										レジスタ												INDEX			定数	修飾	
opr	名称	X	Y	S	L	R	Z	T	C.	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A+1・A	比較値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
B+1・B	基準値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○		○	○

プログラム例

X0144	Z109876543	R0008
	[RW027・RW026 D]>= 1234567890]	()

X0144 2109876543 R0008
| |—[RW027·RW026 D]>= 1234567890]—————()

動作 • a 接点X0144がONの時、倍長レジスタRW027・RW026のデータ(2109876543)とオペランドB

- ・ a 接点X0144がONの時、倍長レジスタRW027・RW026のデータ(2109876543)とオペランドB + 1・Bの倍長データ(1234567890)を比較します。比較結果はオペランドA ≥ オペランドBとなるため出力をONにします。

A+1:RW027 A:RW026
2109876543
F 0 F 0
1234567890
B+1: B:

比較： $D \geq \rightarrow A + 1 \cdot A \geq B + 1 \cdot B$

© 2006 The Authors
Journal compilation © 2006 Blackwell Publishing Ltd

条件入力

判定出力

$D = [A+1 \cdot A \quad B+1 \cdot B]$

- ・2つの倍長データが $A + 1 \cdot 1 = B + 1 \cdot B$ のとき出力をONにします。
- ・データ $A + 1 \cdot A$ 、 $B + 1 \cdot B$ のデータ範囲は-2147483648～2147483647です。
- ・データA、Bのデータはこの比較により変化しません。
(符号付きデータ比較)

オペランド

プログラム例

R0145	-2012345678	-2012345678	R0009
	D0001・D0000	D= RW032・RW031	

- ・ a 接点X0145がONの時、オペランドA + 1・Aの倍長データ(-2012345678)と倍長レジスタRW032・RW031の倍長データ(-2012345678)を比較します。比較結果はオペランドA = オペランドBとなるため出力をONにします。

比較： $\Rightarrow A + 1 \cdot A = B + 1 \cdot B$

© 2006 The Authors
Journal compilation © 2006 Blackwell Publishing Ltd

FUN 1 0 5 倍長比較 : 等しくない (D<>)

倍長データを比較し≠を検出します。

関連命令

条件入力

判定出力

$$- [A+1 \cdot A \quad D \langle \rangle \quad B+1 \cdot B]$$

機能

- ・ 2つの倍長データが $A + 1 \cdot 1 \neq B + 1 \cdot B$ のとき出力を ON にします。(符号付きデータ比較)
- ・ データ $A + 1 \cdot A$ 、 $B + 1 \cdot B$ の定数範囲は -2147483648 ~ 2147483647 です。
- ・ データ A、B のデータはこの比較により変化しません。
- ・ 入力が OFF の場合は比較をおこないません。

入力	処 理		出力
OFF	不実行		OFF
ON	実行	$A + 1 \cdot A \neq B + 1 \cdot B$	ON
		$A + 1 \cdot A = B + 1 \cdot B$	OFF

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾				
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K		
A+1・A	比較値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○		○	○	
B+1・B	基準値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○			○	○		○	○

プログラム例

X0146 0027854167 -0027854167 R000A
 [RW035·RW034 D<> RW038·RW037] ————— ()

動作

- ・ a 接点X0146がONの時、レジスタRW035・RW034の倍長データ(27854167)とオペランドB + 1・Bの倍長データ(-27854167)を比較します。比較結果はオペランドA≠オペランドBとなるため出力をONにします。

A+1:RW035 A:RW034
27854167
F 0 F 0
-27854167
B+1: B:

比較： $D < > \longrightarrow A + 1 \cdot A \neq B + 1 \cdot B$

条件入力	判定出力
$\neg [A+1 \cdot A \quad D < \quad B+1 \cdot B]$	

- ・ 2つの倍長データが $A + 1 \cdot 1 < B + 1 \cdot B$ のとき出力を ON にします。
- ・ データ $A + 1 \cdot A$ 、 $B + 1 \cdot B$ のデータ範囲は -2147483648 ~ 2147483647 です。
- ・ データ A 、 B のデータはこの比較により変化しません。
- ・ (符号付きデータ比較)

オペランド

プログラム例

X0147	0087654321	0123456789	R000B
	[RW040·RW039]	D< D1009·D1008]	()

- ・ a 接点X0147がONの時、レジスタRW040・RW039の倍長データ(87654321)とレジスタD1009 D1008の倍長データ(123456789)を比較します。比較結果はオペランドA<オペランドBとなるため出力をONにします。

A+1:RW040 A:RW039
87654321
F 0 F 0
123456789
B+1:D1009 B:D1008

比較： $D < \longrightarrow A < B$

比較：より小さいまたは等しい

FUN1 0 7 倍長比較：より小さいまたは等しい ($D \leq$)

倍長データを比較し $\leq$ を検出します。(符号付き)

関連命令

条件入力

判定出力

$\neg [A+1 \cdot A \quad D \leq B+1 \cdot B]$

機能

- 2つの倍長データが $A+1 \cdot 1 \leq B+1 \cdot B$ のとき出力をONにします。
- データ $A+1 \cdot A$ 、 $B+1 \cdot B$ のデータ範囲は-2147483648～2147483647です。
- データA、Bのデータはこの比較により変化しません。(符号付き、データ比較)

入力	処 理	出力
OFF	不実行	OFF
ON	実行	$A+1 \cdot A \leq B+1 \cdot B$ ON
		$A+1 \cdot A > B+1 \cdot B$ OFF

オペランド

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A+1・A	比較値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
B+1・B	基準値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○

プログラム例 $X0148 \quad -0052684912 \quad R000C$
 $\neg [-2015326489 \quad D \leq RW043 \cdot RW042]$

動作

- a接点X0148がONの時、オペランド $A+1 \cdot A$ の倍長データ(-2015326489)とレジスタRW043・RW042の倍長データ(-52684912)を比較します。比較結果はオペランド $A \leq$ オペランドBとなるため出力をONにします。

A+1: A:
 -2015326489
 F 0 F 0
 -52684912
 B+1:RW043 B:RW042

比較: $D \leq \rightarrow A+1 \cdot A \leq B+1 \cdot B$

Figure 1 is a logic diagram. On the left, under the label '条件入力' (Condition Input), there is a logic gate with two inputs labeled 'A' and 'U', and one output labeled 'B'. A horizontal line connects the output 'B' to the label '判定出力' (Judgment Output) on the right.

- ・オペランドA > オペランドB のとき出力をONにします。
- ・オペランドA、Bのデータ範囲は0~65535です。
- ・オペランドA及びBのデータはこの比較により変化しません。

(符号無し、データ比較)

オペランド動作

- ・ a 接点X0133がONの時、レジスタD0126のデータ(58263)とオペランドBのデータ(42532)を比較します。比較結果はオペランドA>オペランドBとなるため出力をONにします。

- 164 -

条件入力	判定出力
$\neg [A \cup \geq B]$	

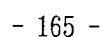
- ・オペランド A $\geq$ オペランド B のとき出力を ON にします。
- ・オペランド A、B のデータ範囲は 0 ~ 65535 です。
- ・オペランド A 及び B のデータはこの比較により変化しません。
(符号無し、データ比較)

オペランド

プログラム例

X0134	58263	11028	R000E
		U>= RW211	()

- ・ a 接点X0134がONの時、レジスタRW124のデータ(58263)とレジスタRW211のデータ(11028)を比較します。比較結果はオペランドA>オペランドBとなるため出力をONにします。



符号無し比較：等しい

FUN 1 1 0 符号無し比較：等しい (U=)	
データを比較し=を検出します。(符号無し)	関連命令

条件入力	判定出力
[A U= B] _____	

- 機能
- オペランドA = オペランドB のとき出力をONにします。
 - オペランドA、Bの定数範囲は0~65535です。
 - オペランドA及びBのデータはこの比較により変化しません。
(符号無し、データ比較)

入力	処 理	出力
OFF	不実行	OFF
ON	実行	A = B ON
		A ≠ B OFF

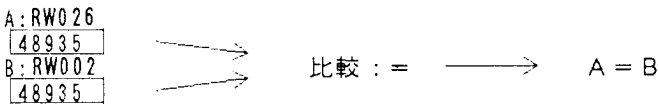
オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K		
A	比較値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
B	基準値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○

プログラム例

X0135	48935	48935	R000F
		[RW026 U= RW002]	()

- 動作
- a 接点X0135がONの時、レジスタRW026のデータ(48935)とレジスタRW002のデータ(48935)を比較します。比較結果はオペランドA = オペランドBとなるため出力をONにします。



符号無し比較：等しくない

FUN1 1 1 符号無し比較：等しくない (U<>)

データを比較し≠を検出します。(符号無し)

関連命令

条件入力

判定出力

[A U<> B]

機能

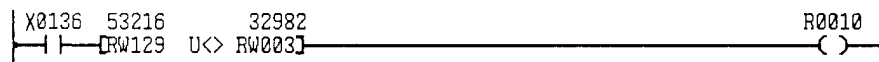
- ・オペランドA≠オペランドBのとき出力をONにします。
- ・オペランドA、Bの定数範囲は0~65535です。
- ・オペランドA及びBのデータはこの比較により変化しません。
(符号無し、データ比較)

入力	処 理		出力
OFF	NOP (不実行)		OFF
ON	実行	A ≠ B	ON
		A = B	OFF

オペランド

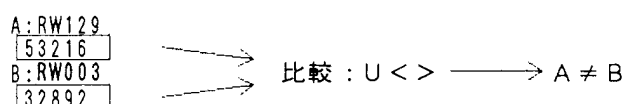
		デバイス										レジスタ												INDEX			定数	修飾	
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	比較値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
B	基準値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○

プログラム例



動作

- ・ a 接点X0136がONの時、レジスタRW129のデータ(53216)とレジスタRW003のデータ(32982)を比較します。比較結果はオペランドA≠オペランドBとなるため出力をONにします。



© 2006 The Authors
Journal compilation © 2006 Blackwell Publishing Ltd

データを比較しくを検出します。

関連命令

— [A U K B] —————

- ・オペランドA<オペランドBの場合に出力をONにします。
- ・データは符号無しデータです。
- ・オペランドA、Bのデータ範囲は0~65535です。
- ・オペランドA及びBのデータはこの比較により変化しません。
(符号無し、データ比較)

入力	処	理	出力
OFF	不実行		OFF
ON	実行	$A < B$	ON
		$A \geq B$	OFF

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A	比較値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
B	基準値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○

X0137 65201 R0011
[42587 UK RW129]

- ・ a 接点X0137がONの時、オペランドAのデータ(42587)とレジスタRW129のデータ(65201)を比較します。比較結果はオペランドA<オペランドBとなり出力をONにします。

42587
B:RW129
65201

65201

比較： $U < \longrightarrow A < B$

条件入力	判定出力
$\neg [A \vee \leq B]$	

- オペランドA ≦ オペランドB のとき出力を ON にします。
- オペランドA、B のデータ範囲は 0 ~ 65535 です。
- オペランドA 及び B のデータはこの比較により変化しません。
(符号無し、データ比較)

オペランド

$$A \subseteq B$$

デバイス／レジスタセット

FUN 1 1 4 デバイス／レジスタセット (SET) *

レジスタ、デバイスをセットします。

関連命令

デバイス／レジスタリセット

条件入力

実行出力

—[SET A]—

機能

- オペランドAがデバイスの場合は、デバイスAをONにします。
- オペランドAがレジスタの場合は、レジスタAにHFFFFを格納します。
- 命令実行後に入力条件がOFFになると、出力はOFFになりますが、レジスタ、デバイスの状態は変化しません。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定	修		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾
A	デバイス・レジスタ	○	○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○	○	○	○		○

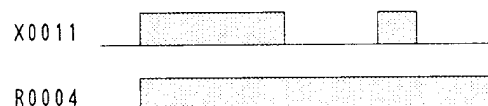
プログラム例 1

オペランドAがデバイスのとき



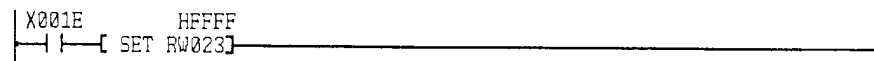
動作

- a 接点X0011がONになるとデバイスR0004が1にセットされ、出力がONになります。その後X0011がOFFになってもR0004の状態は変化しません。ただし出力はOFFになります。



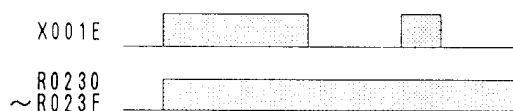
プログラム例 2

オペランド A がレジスタのとき



動作

- ・ a 接点X001EがONになるとレジスタRW023がHFFFFにセットされ、出力がONになります。その後X001EがOFFになってもRW023の状態は変化しません。ただし出力はOFFになります。



デバイス／レジスタリセット

FUN1 1 5 デバイス／レジスタリセット (RST) *	
レジスタ、デバイスのデータを0にリセットします。	関連命令 デバイス／レジスタセット

条件入力	実行出力
—[RST A]—	

機能

- ・オペランドAがデバイスの場合は、デバイスAをリセットします。
- ・オペランドAがレジスタの場合は、レジスタAにH0000を格納します。
- ・命令実行後に入力条件がOFFになると、出力はOFFになりますが、レジスタ、デバイスの状態は変化しません。

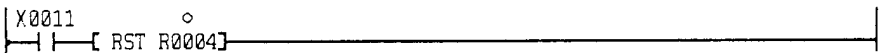
入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デバイス										レジスタ										INDEX			定修			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数飾	
A	デバイス・レジスタ	○	○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○	○	○	○		○

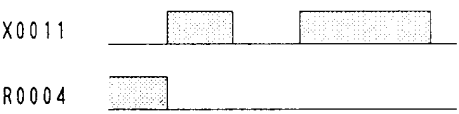
プログラム例 1

オペランドAがデバイスの場合



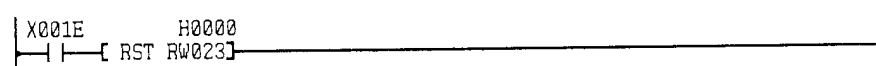
動作

- ・a 接点X0011がONの時、デバイスR0004にOFFをリセットし出力をONにします。



プログラム例 2

オペランド A がレジスタの場合



動作

- ・ a 接点X001EがONの時、レジスタRW023にデータH0000をリセットし出力をONにします。

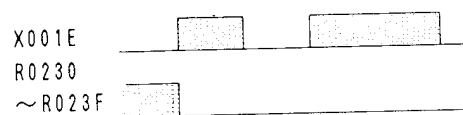


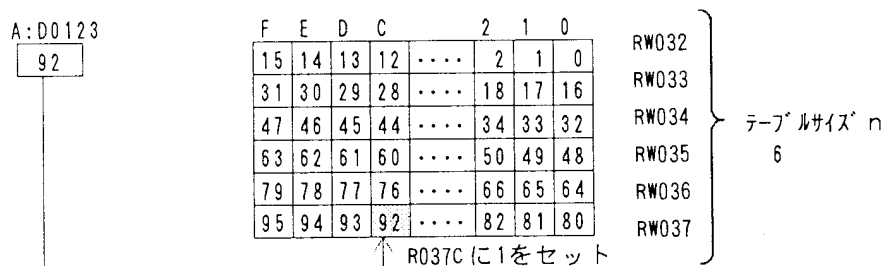
Figure 1. The effect of the number of trials on the number of correct responses. The number of correct responses was plotted against the number of trials for each condition. The number of correct responses increased with the number of trials for all conditions. The number of correct responses was highest for the condition with the highest number of trials (10 trials) and lowest for the condition with the lowest number of trials (2 trials).

条件入力	実行出力
—[A TSET (n) B]—————	

- レジスタBを先頭とするテーブルサイズnのレジスタファイルの中の、オペランドA（ビットポインタ）で示す任意の1ビットをONにセットします。
- ビットポインタがテーブル範囲外の時は、セットを行わず出力をOFFにします。（テーブルオペレーションエラー）
- テーブルサイズ（オペランドn）はレジスタ（ワード）単位で1~64の範囲で設定できます（1024ビット）。

オペランド動作

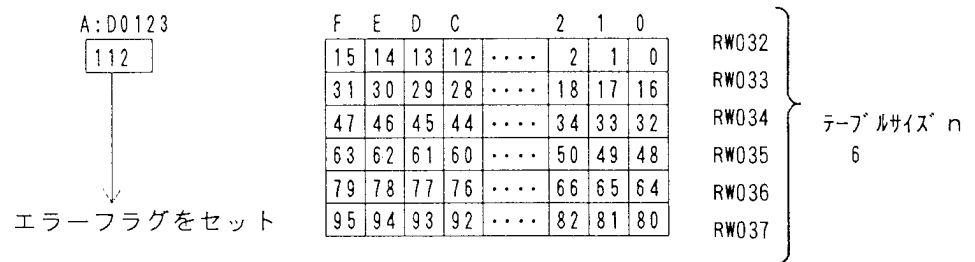
- ・ a 接点X012AがONの時、レジスタRW032を先頭とする6ワードのテーブルにおいて、レジスタD0123のデータ(92)に該当するビットポインタ92番のデバイスをONにセットします。



プログラム例 2

R012A	00112	H1005
D0123 TSET (06) RW032		

- 動作
- レジスタ D0123 のデータ (112) に該当するビットポインタはテーブル範囲外です。この場合はエラーフラグをセットし、出力を O F F にします。



条件入力	実行出力
$\neg [A \text{ TRST } (n) \ B]$	

機能

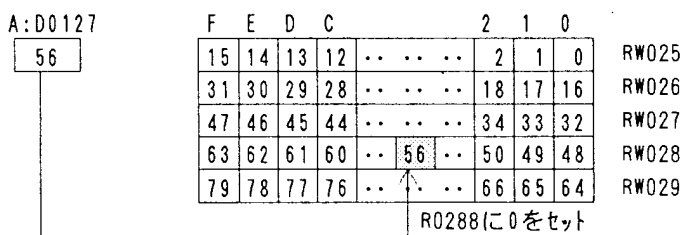
- レジスタ B を先頭とするテーブルサイズ n のレジスタファイルの中の、オペランド A (ビットポインタ) で示す任意の 1 ビットを 0 リセットします。
- ビットポインタがテーブル範囲外の時は、リセットを行わず出力を OFF にします。(テーブルオペレーションエラー)
- テーブルサイズ (オペランド n) の単位は、レジスタ (ワード) 単位で 1~64 の範囲で設定できます (1024 ビット)。

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	リセットビットP											○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
n	テープルサイズ																										○	
B	T先頭番地											○	○	○	○	○	○	○	○	○	○		○					○

```
|R012E 00056          H367E  
| | |D0127 TRST (05) RW025]
```

動作

- ・ a 接点X012EがONの時、レジスタRW025を先頭とする5ワードのテーブルにおいて、レジスタD0127のデータ(56)に該当するビットポイントa番のデバイスをリセット(0)し出力をONにします。

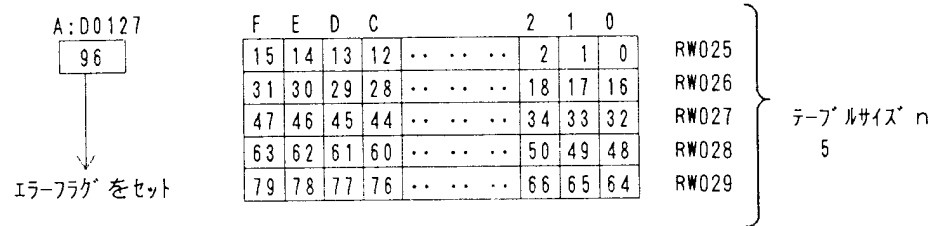


プログラム例 2 テーブルサイズオーバーの場合

```
|R012E 00096 H367E  
| | |D0127 TRST (05) RW025]
```

動作

- レジスタD0127のデータ(96)に該当するビットポインタはテーブル範囲外です。
この場合はエラーフラグをセットし、出力をOFFにします。



キャリーセット

FUN1 1 8 キャリーセット (SETC)

キャリーフラグをセットします。

関連命令

キャリーリセット

条件入力

--[SETC]

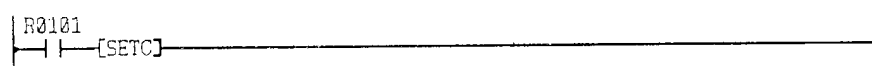
実行出力

機能

- ・ 入力がONの時、キャリーフラグをセットし出力をONにします。
- ・ 入力がOFFの時は、キャリーフラグをセットせず出力をOFFにします。

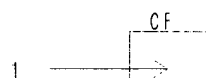
入力	処 理	出力	C F
OFF	不実行	OFF	
ON	実行	ON	○

プログラム例



動作

- ・ a 接点R0101がONの時、キャリーフラグをセットし出力をONにします。



キャリーリセット

FUN1 1 9 キャリーリセット (RSTC)

キャリーフラグをリセットします。

関連命令

キャリーセット

条件入力

—[RSTC]—

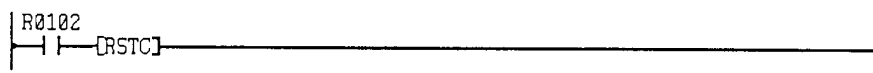
実行出力

機能

- 入力がONの時は、キャリーフラグをリセットし出力をONにします。
- 入力がOFFの時は、キャリーフラグをリセットせず出力をOFFにします。

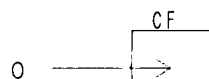
入力	処 理	出力	C F
OFF	不実行	OFF	
ON	実行	ON	リセット

プログラム例



動作

- a 接点R0102がONの時、キャリーフラグをリセットし出力をONにします。



連続したビットデータの中の最上位ONビットを検出し、ファイルを作成します。	関連命令 デコード

条件入力	実行出力
-[A ENC (n) B]-	

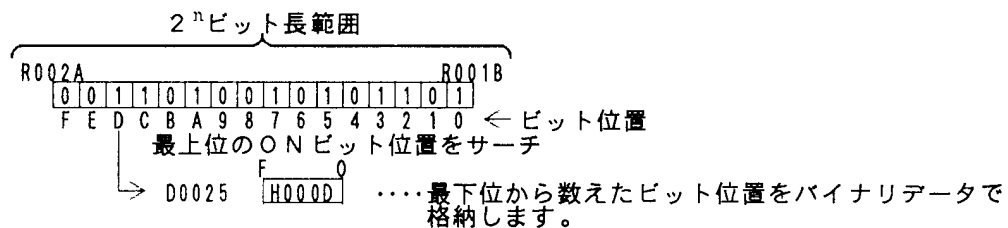
・ データのうち、最上位のビットは、フラグをセット
 ・ 対し、エシマシは、1レタス。最
 ・ 才オオは、8タス。最下
 ・ 位は、1レタス。

入力	処 理		出力	ERF
OFF	不実行		OFF	
ON	ONビット有り	実行	ON	
	ONビット無し	不実行	OFF	○

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A	先頭データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○							○
n	テーブルサイズ																											○	
B	エンコードエリア												○	○	○	○	○	○	○	○	○	○		○	○	○	○		○

X001A • H000D
| | | [R001B ENC (4) D0025]

・ a 接点X001AがONの時、デバイスR001Bを先頭とするサイズ2⁴(16)ビット範囲のデータの最上位のONビット位置“D”をサーチし、レジスタD0025に格納して出力をONにします



FUN1 2 1 デコード (DEC) *

連続したビットデータのなかの特定の1ビットをONにします。

関連命令
エンコード

実行出力

[A DEC (n) B] _____

機能

- オペランド B を先頭とする 2^n ビット長エリアデータの中から、オペランド A のレジスタの下位 n ビットの数値に対応するビットだけを ON にし、他は全て OFF (0) にします。
- オペランド n (テータサイズ) のデータ範囲は 1 ~ 8 です。
- オペランド B をレジスタで指定した場合は、その最下位ビット (0) がビット位置の先頭になります。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デバイス										レジスタ										INDEX			定数	修飾			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A	先頭データ											○	○	○	○	○	○	○	○	○	○	○	○		○	○	○		○
n	データサイズ																												○
B	エンコードエリア	○	○	○	○	○	○				○		○		○	○	○	○	○	○	○	○		○					○

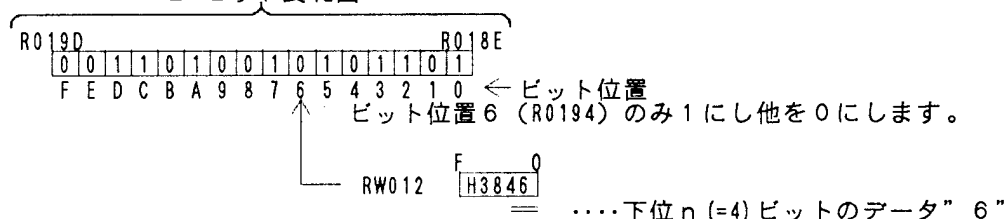
プログラム例

X002B H3846
RW012 DEC (4) R018E

動作

- ・ a 接点X002BがONの時、レジスタRW012のデータ(H3846)の下位n=4で示される“6”をビット位置の番号とし、デバイスR018Eを先頭とする2⁴(=16)ビットデータの6ビット目に該当するデバイスを1 (ON)とし、他のデバイスには0 (OFF)を格納します。

2ⁿビット長範囲



データビットカウンタ

FUN 1 2 2 データビットカウンタ (BC) *

連続したビットデータ (最大 16) の中の ON ビット数をカウントし
ます。

関連命令

条件入力

実行出力

— [A BC B] —

機能

- オペランド A の最大 16 ビットデータのうち、ON (1) のビット数をカウントして、オペランド B (16 ビットデータ) にその数を格納します。
- オペランド A のデータ範囲は、H0000~HFFFF です。
- オペランド A で桁指定を行った場合は、対象桁分のビット数をカウントします。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	カウンテータ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
B	格納エリア		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○	○	○	○

プログラム例

R001F HF73C H000B
— [RW022 BC D0102] —

動作

- a 接点 R001F が ON の時、レジスタ RW022 のデータ (HF73C) のうち、ON (1) になっているビット数をカウントし、結果 11 (H000B) をレジスタ D0102 に格納します。さらに出力を ON にします。

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
RW022	1	1	1	1	0	1	1	1	0	0	1	1	1	1	0	0

1 の数をカウント (11個)

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
D0102	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	

→ 11個 (オペランド B にはバイナリデータとして格納)

© 2004 Blackwell Publishing Ltd, *Journal of Internal Medicine* 255: 103–110

FUN 1 2 3 倍長ビットカウンタ (DBC) ※

関連命令

条件入力	実行出力

—[A+1 • A DEC B]—————

機能

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

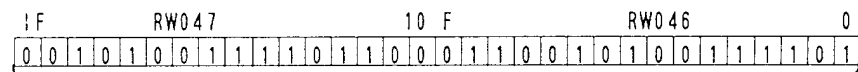
		デバイス										レジスタ										INDEX			定数	修飾			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A+1・A	カウンテータ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
B	格納エリア		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○	○		○

プログラム例

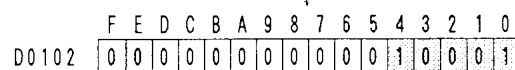
R002F H29EC653D H0011
| | RW047·RW046 DBC D0103

動作 ・ a接点R002FがONの時、レジスタRW047・RE0046のデータ(H29EC653D)のうち、ON(1)にな

- ・ a 接点R002FがONの時、レジスタRW047・RE0046のデータ(H29EC653D)のうち、ON(1)になっているビット数をカウントし、結果17(H0011)をレジスタD0103に格納します。そして出力をONにします。



1 の数をカウント (17個)



→17個 (オペランドBはバイナリデータとして格納)

FUN 1 2 4 データサーチ (SCH) *

レジスタより任意のデータをサーチします。

関連命令

判定出力

—[A SCH (n) B → C]—————

機能

- レジスタ A の 16 ビットデータと同一のデータをレジスタ B を先頭とする n ワードのテーブル内データよりサーチします。最初にサーチしたデータのレジスタ番号を C + 1 に、サーチしたデータの個数を C にそれぞれ格納します。
- データがまったくサーチされなかった場合は、オペランド C に 0、オペランド C + 1 には実行前の値がそのまま残ります。また出力は OF F にします。
- オペランド n (テーブルサイズ) のデータ範囲は 1 ~ 64 です。
- オペランド A、B のデータ範囲は、-32768 ~ 32767 です。

入力	処 理		出力
OFF	不実行		OFF
ON	実行	一致データ有	ON
		一致データ無	OFF

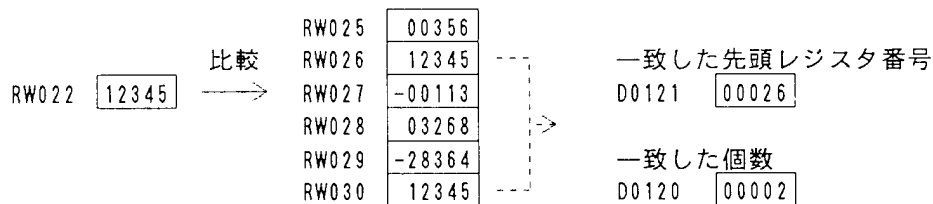
オペランド		デ バ イ ス										レ ジ ス タ										INDEX				定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	基準レジスタ											○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○	
n	テーブルサイズ																										○		
B	サーチ用レジスタ											○	○	○	○	○	○	○	○	○	○	○						○	
C	格納レジスタ												○	○	○	○	○	○	○	○	○	○	○	○	○	○	○		○

		デバイス										レジスタ										INDEX			定数	修飾			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	基準レジスタ											○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
n	テーブルサイズ																											○	
B	サーチ用レジスタ											○	○	○	○	○	○	○	○	○	○	○							○
C	格納レジスタ												○	○	○	○	○	○	○	○	○	○		○	○	○	○		○

プログラム例

動作

- ・ a 接点X0101がONの時、レジスタRW022のデータ(12345)を基準値とします。レジスタRW025を先頭とする6ワード分のテーブル内データから基準値(12345)をサーチします。結果最初に一致したレジスタ番号(26)をD0121に、一致したデータの個数(2)をD0120に格納します。また実行出力をONにします。



補足

- ・オペランドCにインデックスKを指定すると、最初に一致したレジスタ番号(C+1)は切り捨てられますのでご注意ください。

FUN 125 プッシュ (PUSH) *

レジスタデータをスタックエリアに格納します。

条件入力 スタックフル出力

条件入力	スタックフル出力
-[A PUSH (n) B + C]	

機能

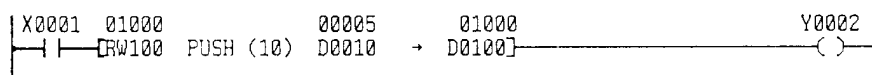
- オペランドAのレジスタデータをオペランドCを先頭とするサイズn分のテーブル(スタックエリア)に1プッシュ(格納)し、オペランドBのレジスタ(ポインタ)を1インクリメントします。
- データ総数がオペランドn(スタックサイズ)の範囲外の場合は、プッシュせず出力をOFFにし、エラーフラグをセットします。(テーブルオペレーションエラー)
- オペランドnのデータ範囲は、1~64です。
- オペランドAのデータ範囲は、-32768~32767です。

入力	処 理				出力	ERF
OFF	不実行				OFF	
O N	スタックに	$0 \leq B < n-1$	実行	$B \leftarrow B+1$	OFF	
	空き有	$B = n-1$			O N	
	スタックフル	$B = n$			O N	
	範囲外	$B > n, B < 0$	不実行	OFF	○	

オペランド

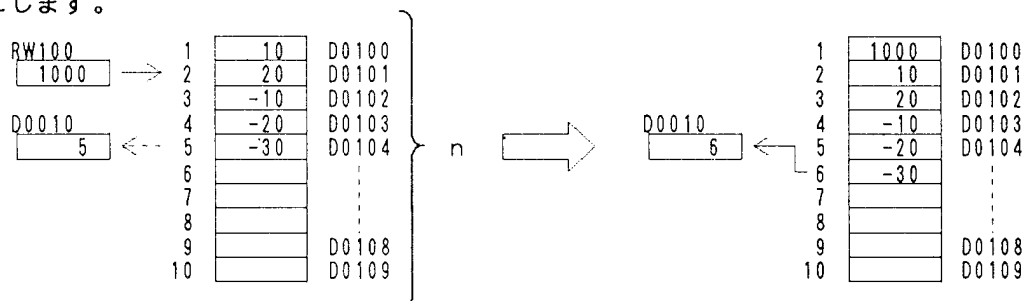
		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A	基準レジスタ	○	○	○	○	○	○				○	○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
n	スタックサイズ																												○
B	総数レジスタ												○	○	○	○	○	○	○	○	○	○		○	○	○	○		○
C	先頭レジスタ											○	○	○	○	○	○	○	○	○	○	○							○

プログラム例 1

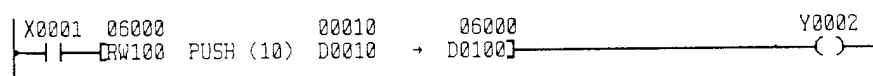


動作

- 動作
- ・ a 接点X0001がONの時、レジスタRW100のデータ(1000)をレジスタD0100を先頭とするサイズ10ワードのスタックの先頭にプッシュします。PUSH実行以前のデータは1つづつ下にシフトされ、データ総数(D0010)は1インクリメント(+1)されます。また出力をOFFにします。

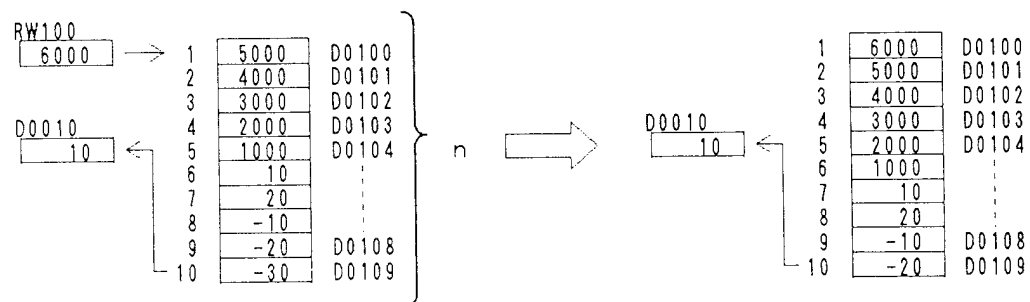


プログラム例 2



動作

- スタックフルの状態でプッシュすると、スタックの底 (D0109) のデータ (-30) を捨ててシフトダウンした後、新データをプッシュします。データ総数は、インクリメントせず出力を ON にします。



ポップファースト

FUN127 ポップファースト (POPF) *

スタックエリアの最終データを引き出します。

関連命令
プッシュ
ポップラスト

条件入力

スタック空出力

[A POPF (n) B → C]

機能

- オペランドAを先頭とするサイズnのテーブルの最初にプッシュされたデータ(末尾データ)をポップし、オペランドC(16ビット長エリア)に格納します。さらにレジスタBのデータを1デクリメントします。
- スタック内にデータが無い場合は、ポップを行わず出力をOFFにします。
- レジスタBがスタックサイズ範囲外の時は、ポップを行わず出力もOFFにします。
- オペランドnのデータ範囲は1~64です。

入力	処 理		出力	ERF
OFF	不実行		OFF	
ON	スタックにデータ有	$1 < B \leq n$	実行	OFF
		$B = 1$		ON
	データ無範囲外	$B = 0$	不実行	ON
		$B > n, B < 0$		OFF ○

オペランド

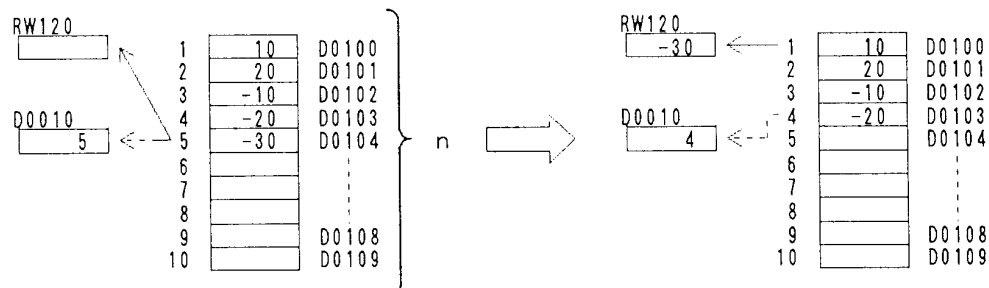
		デバイス										レジスタ										INDEX			定数	修飾		
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	先頭レジスタ												○	○	○	○	○	○	○	○	○							○
n	スタックサイズ																											○
B	総数レジスタ												○	○	○	○	○	○	○	○	○		○	○	○	○		○
C	データ格納先	○	○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○	○	○	○		○

プログラム例 1

X0003 00010 00005 -00030 Y0002
[D0100 POPF (10) D0010 → RW120] ()

動作

- a 接点X0003がONの時、レジスタD0100を先頭とするスタックから、スタックの末尾データ(D0104:-30)をレジスタRW120に格納します。またデータ総数レジスタD0010を1デクリメントし、出力をOFFにします。



サブルーチンコール

FUN 1 2 8 サブルーチンコール (CALL) *

サブルーチンプログラムを呼び出します。

関連命令

サブルーチンリターン
サブルーチンエントリー

条件入力

実行出力

-[CALL N. XX]

機能

- ・サブルーチンエントリーNO. XXで管理されているサブルーチンプログラムをコールします。
- ・サブルーチンプログラム中では、同じサブルーチンエントリーNO. を複数回コールすることはできません。
- ・オペランドXXの範囲は、0~255です。256以上になるとエラーフラグをセットします（アドレスバウンダリエラー）。ただしT1では0~15の範囲です。
- ・コールのネスティングは、6段階までできます。7段階以上を指定するとエラーフラグをセットします。（ネスティングオーバーエラー）ただしT1ではコールのネスティングは行えません。
- ・サブルーチンリターン命令によりメインプログラムに復帰します。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定	修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
XX	エントリーNO.																											○	○

プログラム例

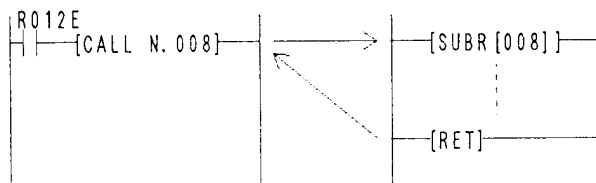
動作



- ・a接点R012E ONの時、サブルーチンエントリーNO. 8のサブルーチンをコールし、実行した後、出力をONにします。

メインプログラム

サブルーチンプログラム



サブルーチンリターン

FUN 1 2 9 サブルーチンリターン (RET) *

サブルーチンプログラムを終了します。

関連命令

サブルーチンエントリー



機能

- ・ サブルーチンプログラムを終了し、元のプログラムへ復帰します。

プログラム例

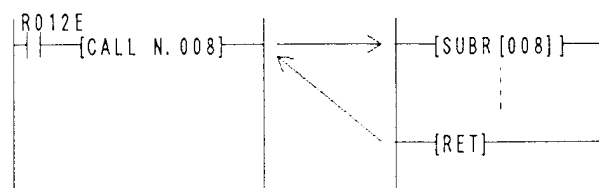


動作

- ・ サブルーチンプログラムの実行を終了して、元のプログラム（サブルーチンコール命令の次の回路）に戻ります。

メインプログラム

サブルーチンプログラム



補足

- ・ サブルーチンエントリー (SUBR) 命令とペアで使用して下さい。

条件ジャンプ／ジャンプラベル

FUN 1 3 0 / 1 3 6 条件ジャンプ (JUMP) * ジャンプラベル (LBL)

指定先回路へジャンプします。

関連命令

条件入力

常時OFF

—[JUMP N. XX]—

—[LBL (XX)]—

機能

- 指定されたラベル番号の回路にジャンプします。
- ジャンプ先のラベル番号（オペランドXX）の範囲は0～255です。この範囲外のラベル番号を指定したときはエラーフラグをセットします（アドレスバウンダリエラー）。
- ジャンプ先のラベル番号に対応するジャンプラベル命令が登録されていないときはエラーフラグをセットします（アドレス登録エラー）。
- 逆戻りのジャンプはできません。
- メインプログラム、サブルーチンプログラムおよび割り込みプログラム間でジャンプすることはできません。

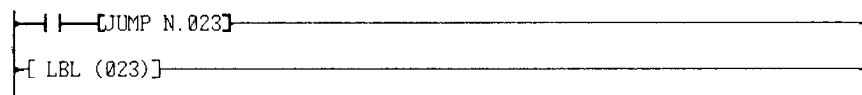
入力	処 理	出力
OFF	不実行	OFF
ON	実行	OFF

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
XX (1)	飛先 LBL NO																										○	○
XX (2)	LBL NO																										○	

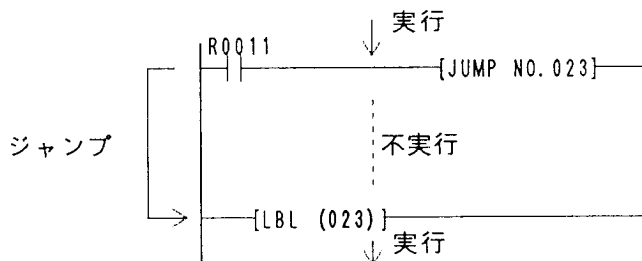
注) XX (1) : JUMP XX (2) : LBL

プログラム例



動作

- a 接点R0011がONの時、ラベル番号23の回路にジャンプし、LBLに続くプログラムを実行します。JUMP、LBL間のプログラムは実行されません。



補足

- プログラムツールT-PDSから、プログラム画面のモニタ表示を行う場合、ジャンプ中はオンライントレース表示が正しく行なわれなくなります。この場合、飛先のLBL命令を先頭に画面を再表示して下さい。

繰り返し

FUN 1 3 2 / 1 3 3 繰り返し (FOR/NEXT)

指定回路間を繰り返し実行します。

関連命令

条件入力

—[FOR n]—

—[NEXT]—

機能

- FOR～NEXT間をn回繰り返し実行します。nは1～32767の範囲です。
 - 指定回数繰り返し実行後、NEXT命令の次の回路を実行します。
 - FORとNEXTは必ず1:1で使います。ペアとなるNEXT命令の前にFOR命令を置きます。
 - FOR～NEXT命令間にI RET命令、RET命令、END命令は使用できません。また、ジャンプ命令で抜け出すこともできません。
 - FOR～NEXTのネスティングは最大6重です。この範囲を超えるとネスティングエラーが発生します。
- ただしT1ではネスティングは行えません。

FOR

入力	処 理	出力
OFF	FOR～NEXT間 繰り返し無し	OFF
ON	FOR～NEXT間 指定回数繰り返し	ON

NEXT

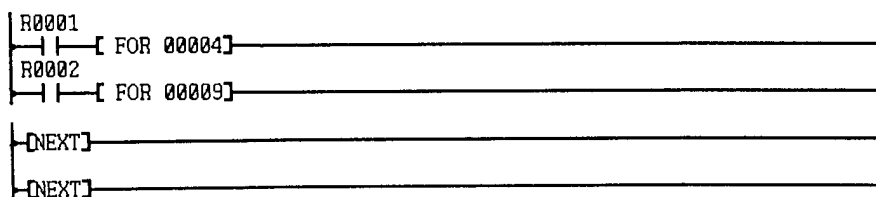
入力	処 理	出力
OFF	繰り返し強制終了	OFF
ON	繰り返し継続	ON

オペランド

(FOR命令)

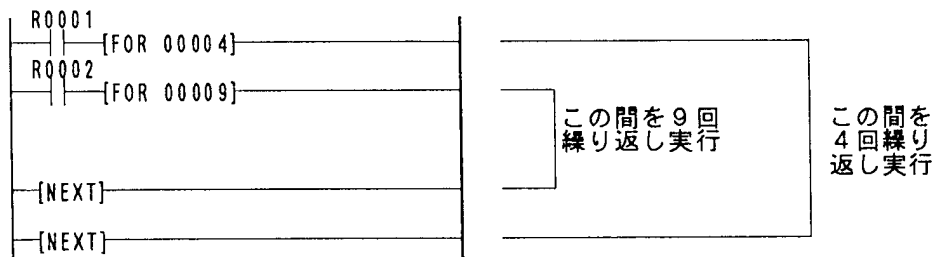
(FOR命令)		デ バ イ ス										レ ジ ス タ										INDEX			定	修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
n	繰返回数											○	○	○	○	○	○	○	○	○	○				○	○	○	○	

プログラム例



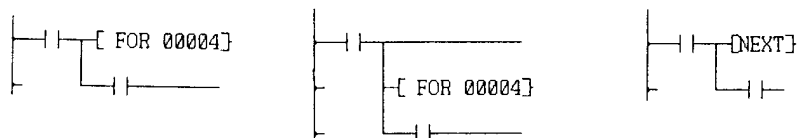
動作

- a 接点R0001、R0002がONの時、指定回路間を繰り返し実行します。



補足

- FOR命令とNEXT命令の前には以下のような分流接続は行えません。



スターコントロールセット n / リセット n

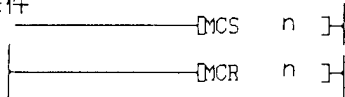
FUN1 3 4 / 1 3 5 マスターコントロールセット n / リセット n (MCS / MCR)

指定回路間の母線入力を ON / OFF します。

関連命令

マスターコントロールセット
マスターコントロールリセット

入力条件



機能

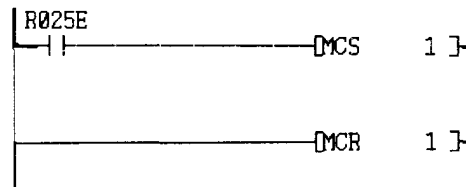
- MCS の入力が OFF のとき MCR までの母線入力を OFF にします。
- MCS の入力 ON のときは通常動作を行います。
- MCS、MCR は必ず 1 : 1 で使用します。ひとつの MCS、MCR 間にさらに MCS、MCR を組み込むことができます。
- MCS、MCR のネスティング (n) の範囲は 1 ~ 7 です。この範囲を超えるとネスティングエラーが発生します。

入力	処 理	出力
OFF	母線入力 OFF	
ON	通常動作	

オペランド

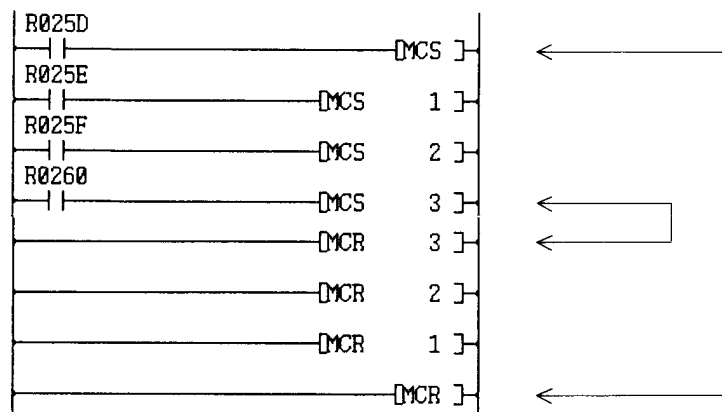
		デバイス										レジスタ										INDEX			定数	修飾			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
n	ネスティング NO.																											○	

プログラム例



動作

- a 接点 R025E が ON の時は通常動作を行います。
- a 接点 R025E が OFF になると MCS、MCR 間の母線入力が OFF になります。
- ネスティング指定するとき、MCS 命令はネスティング番号の小さい番号から、MCR 命令はネスティング番号の大きい番号からプログラミングします。
- n のない MCS / MCR 命令は番号を 0 とみなします。



サブルーチンエントリー

FUN 1 3 7 サブルーチンエントリー (SUBR)

サブルーチンプログラムの開始位置を示します。

関連命令

サブルーチンコール
サブルーチンリターン

[SUBR(XX)]

機能

- ・サブルーチンプログラムの入口を示します。
- ・オペランドXXの定数範囲は0～255です。ただし、T2は0～31の範囲です。
- ・サブルーチンコールにより、オペランドXXで示される番号のサブルーチンプログラムを実行します。
- ・サブルーチンリターン命令とペアで使用します。

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
XX	ラベルNO.																											○

プログラム例

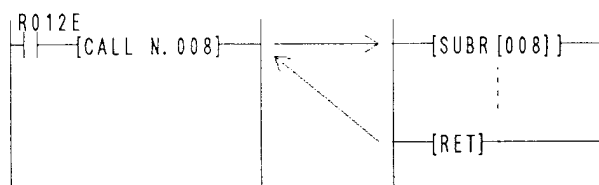
[SUBR(008)]

動作

- ・サブルーチンプログラムの番号を定義し、入口を示します。
- ・本命令は、各回路番号の先頭にしかプログラムできません。

メインプログラム

サブルーチンプログラム



割り込みプログラム許可

FUN1 4 0 割り込みプログラム許可 (EI) *

ユーザ割り込みプログラムの実行を許可します。

関連命令

割り込みプログラム禁止

条件入力

実行出力

——— [EI] ———

機能

- 割り込み禁止状態にあるとき (DI 命令実行中のとき)、この入力が ON になると、ユーザ割り込みプログラムの実行を許可します。
- この入力が OFF のときは、割り込み禁止状態は変化しません。

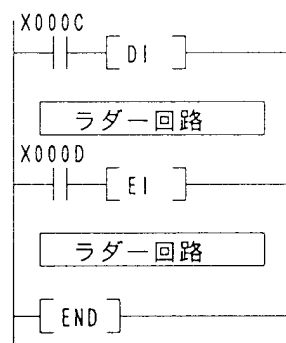
入力	処 理
OFF	不実行
ON	実行

プログラム例



動作

- ユーザ割り込み禁止状態にあるとき (DI 命令実行中のとき)、a 接点 X000D が ON すると、DI 命令実行による割り込み禁止状態を解除して割り込みプログラムの実行を可能にし、出力を ON に設定します。
- ただし、a 接点 X000D が OFF の状態にあるときは、割り込み禁止状態をそのまま持続し、出力を OFF に設定します。
- DI 命令と次の EI 命令の間で割り込み要因が発生しても、DI 命令と次の EI 命令の間に設定されている処理が終了するまで、割り込みプログラムの実行は待機状態となります。



X000C ON の時、この区間で割り込みプログラムの実行を禁止します。この区間で割り込み要因が発生した場合、次の EI 命令に出会うまで割り込みプログラムの実行を待機させます。

補足

- 本命令を割り込みプログラム内で使用することは許されません。
- DI の実行から EI の実行がスキャンをまたぐような指定はできません。

割り込みプログラム禁止

FUN1 4 1 割り込みプログラム禁止 (DI)※

ユーザ割り込みプログラムの実行を禁止します。

関連命令

割り込みプログラム許可

条件入力

実行出力

[DI]

機能

- この入力がONになると、ユーザ割り込みプログラムの実行が禁止され出力がONとなります。この割り込み禁止状態は、次にEI命令を実行するまで続きます。
- この入力がOFFの時は、割り込みプログラムは実行許可状態は変化しません。

入力	処 理
OFF	不実行
ON	実行

プログラム例



動作

- a接点X000CがONすると、割り込みプログラムの実行が禁止されます。
- a接点X000CがOFFすると、割り込み許可状態は変化しません。
- DI命令と次のEI命令の期間で割り込み原因が発生しても、DI命令と次のEI命令の間に設定されている処理が終了するまで、割り込みプログラムの実行は待機状態となります。

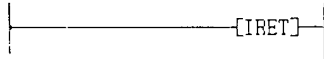
割り込みプログラム終了

FUN 1 4 2 割り込みプログラム終了 (IRET).

ユーザ割り込みプログラムの実行を終了します。

関連命令

割り込みプログラム許可
割り込みプログラム禁止



機能

- 本命令は割り込みプログラム内のみで使用します。
- この命令を実行すると、割り込みプログラムは処理を終了し、メインプログラムまたはサブルーチンプログラムへ復帰します。

プログラム例



動作

- 割り込みプログラムは処理を終了し、メインプログラムまたはサブルーチンプログラムへ復帰します。

補足

- 割り込みプログラム内では E I 命令は使用できません。

ウォッチドッグタイマリセット

FUN1 4 3 ウォッチドッグタイマリセット (WDT)

スキャン周期を延長します。

関連命令

条件入力

実行出力

——[WDT n]——

機能

- ・入力がONの時、ウォッチドッグタイマのスキャン周期（初期設定値は200ms）をタイマ値n（単位10ms）に指定する時間だけ延長します。
- ・タイマ値（n）に指定可能なデータの範囲は1～100（10ms～1sec）。
- ・T1ではタイマ値（1～100）に関らず、命令実行後200ms（固定）スキャン周期を延長します。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 修				
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数 飾		
n	タイマ値																											○	

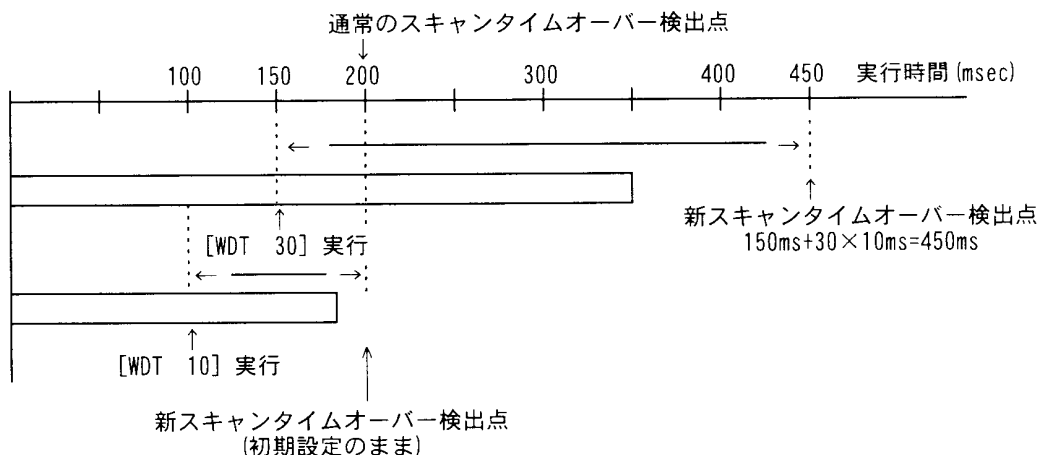
プログラム例

R0010
——[WDT 030]——

動作

以下T3の例で説明します。

- ・通常のスキャンタイムオーバー検出値（ウォッチドッグタイマがスキャン周期の終わりを検出する時点）は200msに設定されています。
- ・a接点R0010がONすると、プログラム開始から接点R0010がONするまでの時間（150ms）にタイマ値30×10msを加算した値（450ms）がスキャンタイムオーバー検出値として再設定されます。
- ・本命令により再設定されたスキャンタイムオーバー検出値が200ms未満の場合、スキャンタイムオーバー値は200msのままです。
- ・a接点R0010がOFFの時はスキャン周期は元のままの200msです。



補足

- ・WDT命令を使用する場合には、必ずメインプログラムおよびサブプログラムで使用して下さい。メインプログラムおよびサブプログラム以外でWDT命令を使用するとスキャン動作が正しく行なわれなくなる場合があります。

FUN1 44 ステップシーケンスイニシャライズ (ST I Z)

指定した範囲内のステップシーケンスを初期設定します。

関連命令

ステップシーケンス入力
ステップシーケンス出力

条件入力	実

行出力

— [STIZ (n) A] —

機能

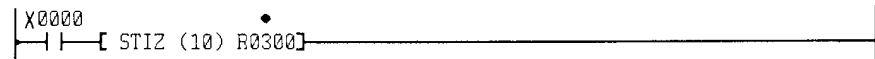
- [illegible]

入力	処 理	出力
OFF	不実行	OFF
ON	OFFからONに立上り時実行	ON

オペランド

[illegible]

プログラム例



- ・ a 接点X0000がONすると、その立上がりでステップシーケンスの先頭デバイスR0300だけをONに設定し、それ以降のR0309までのデバイスを全てOFFに設定します。
- ・ a 接点X0000がOFFならば初期設定を行いません。

デバイスシーケンスの初期設定例（デバイスサイズ $n = 10$ の場合） ON : 1

ON : 1
OFF : 0

	R0009	R0008	R0007	R0006	R0005	R0004	R0003	R0002	R0001	R0000
命令実行前	ON	ON	ON	OFF	ON	OFF	ON	ON	OFF	OFF

↓

[illegible]

先頭デバイスのみON
 その他は全てOFF

補足

- ・本命令はステップシーケンス入力命令 (STIN) およびステップシーケンス出力命令 (STOT) と併用します。
- ・本命令は入力が OFF から ON に変化したときのみ動作します。

ステップシーケンス入力

FUN 1 4 5 ステップシーケンス入力 (STIN)*

ステップシーケンスの条件入力接点です。

関連命令

ステップシーケンス出力

ステップシーケンスイニシャライズ

条件入力

実行出力

—[STIN A]—

機能

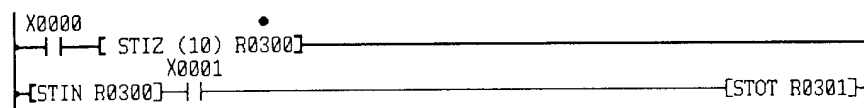
- STINへの入力がONになるとオペランドAのデバイスがONのとき、出力をONにします。
- シーケンスが次のステップに移行するとオペランドAのデバイスはOFFになります。

入力	処 理	出力
OFF	不実行	OFF
ON	A = ON	ON
	A = OFF	OFF

オペンド

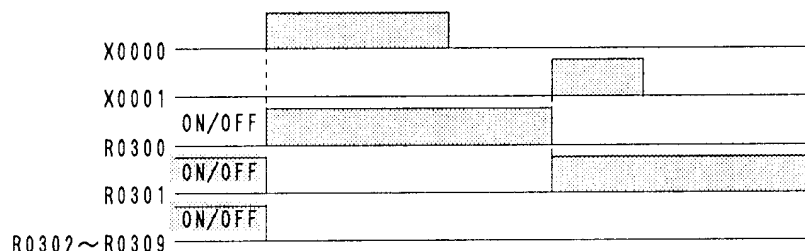
		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	ステップデバイス					○																						○

プログラム例



動作

- a接点X0000の立上がりで、FUN 1 4 4ステップシーケンスイニシャライズを実行します。(R0300をONにし、それ以降のR0309までのデバイスを全てOFFにします。)
- a接点X0000がOFFならば初期設定を行いません。なお、a接点X0001がOFFの間はR0300がONでR0301がOFFの状態が続きますが、X0001がONになるとR0300をOFFにしてR0301がONになります。
- 1回路のなかで直列、並列表示が可能です。(1回路内に最大11個使用可能です。)



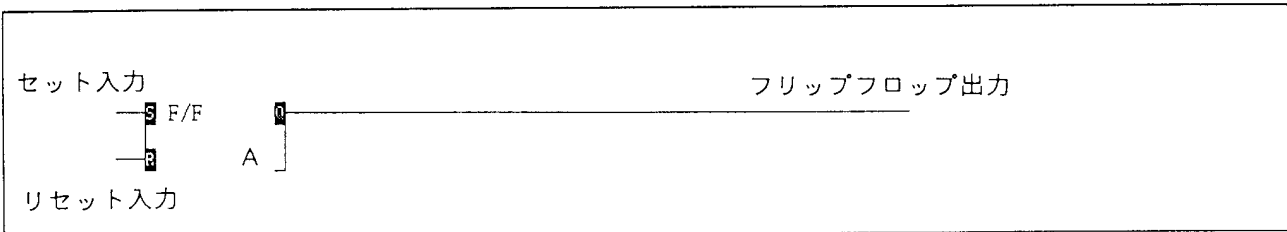
補足

- 本命令はFUN 1 4 4 (ステップシーケンスイニシャライズ) ・ FUN 1 4 6 (ステップシーケンス出力) と併用して使用します。

フリップフロップ

FUN 1 4 7 フリップフロップ (F/F)

リセット入力優先のフリップフロップです。	関連命令



機能

- セット入力がON、リセット入力がOFFの時だけ、オペランドAで指定するデバイスとフリップフロップ出力をONにします。
- リセット入力がONの時は、セット入力の状態に関係なく、オペランドAで指定するデバイスとフリップフロップ出力をOFFにします。
- SとRの両方がOFFの時は、オペランドAで指定するデバイスは変化せずに前の状態を維持します。

入力	処 理	出力
S=1	R = 1	OFF
	R = 0	ON
S=0	R = 1	OFF
	R = 0	前の状態

オペランド

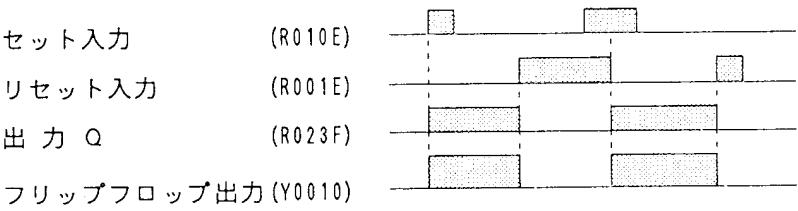
		デ バ イ ス										レ ジ ス タ										INDEX			定	修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
A	F/Fデバイス		○	○	○	○	○																						○

プログラム例



動作

- セット入力R010EがON、リセット入力R011EがOFFの時だけ、出力Q (R023F) がONとなります。
- リセット入力がONの時は、セット入力R010Eの状態に関係なく、出力Q (R023F) をOFFに設定します。
- SとRの両方がOFFの時は、出力Q (R023F) は変化せずに前の状態を維持します。



FUN 1 4 8 タイマトリガ (TRG)

タイマのカウンタアップ動作を起動します。

関連命令

条件入力

判定出力

[TRG A]

機能

- ・ 入力 OFF から ON になると、オペランド A で指定するタイマレジスタによるカウンタアップ動作を 0 から開始し、最大値 65535 に達するとタイムオーバー出力を ON にします。
- ・ カウンタアップ中に入力が OFF になっても、カウンタアップ動作を持続します。
- ・ オペランド A に指定可能なタイマレジスタは T000 ~ T063 です。
- ・ 各タイマレジスタの 1 カウントは 0.01 秒です。最大 655.35 秒までカウンタアップします。

入力	処 理		出力
OFF	前の状態		OFF
ON	実 行	最大値に達してない	OFF
		最大値に達した	ON

オペランド

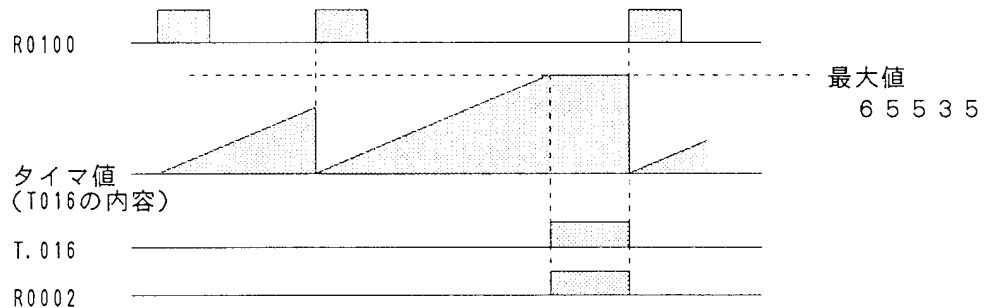
		デ バ イ ス										レ ジ ス タ										INDEX			定	修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
A	タイマレジスタ																	○											

プログラム例



動作

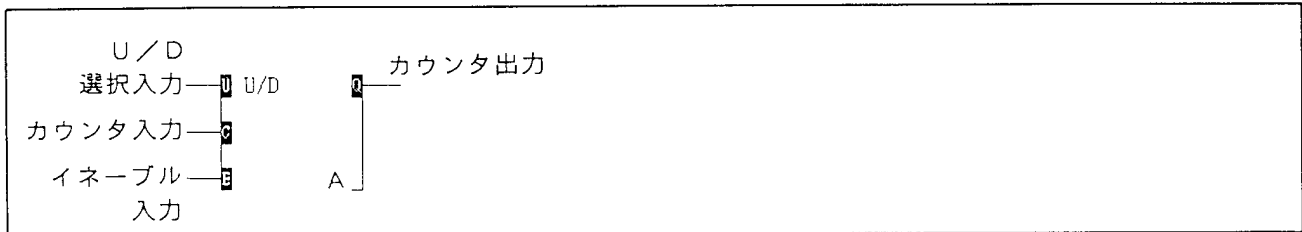
- ・ a 接点 R0100 が OFF から ON になると、タイマレジスタ T016 が 0 からカウンタアップを開始します。a 接点 R0100 が途中で OFF に変化しても、カウンタアップ動作を持続し、T016 が最大値 (65535) までカウントすると、タイマデバイス T.016 とコイル R0002 が ON します。
- ・ a 接点 R0100 の次の OFF から ON への変化で、タイマレジスタはリセットされ、再び 0 からカウンタアップ動作を開始します。
- ・ カウンタアップ中に入力が再び OFF から ON になると、タイマをリセットして、0 からカウンタアップを再開します。



アップダウンカウンタ

FUN1 4 9 アップダウンカウンタ (U/D)

リミット値までカウントすると出力をONにする可逆カウンタです。	関連命令



機能

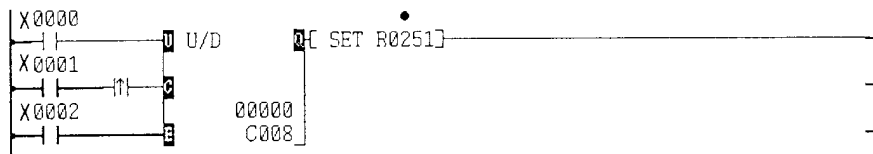
- オペランドAが指定するカウンタレジスタは、イネーブル入力EがONの時、カウンタ入力CのOFFからONへの立ち上がりの回数をカウントし、カウント値がリミット値に達するとカウンタ出力をONにします。リミット値は、アップカウントの場合は65535、ダウンカウントの場合は0です。
- UP/DOWN選択入力UがONの時はアップカウント、OFFの時はダウンカウントです。
- イネーブル入力EがOFFの時はカウント値をクリアし、カウンタ出力QをOFFにします。

入力	処 理	出力
E=OFF	カウント値←0	OFF
E=ON	カウント値<リミット値	OFF
	カウント値=リミット値 かつ カウンタ入力C=OFF	OFF
	カウント値=リミット値 かつ カウンタ入力C=ON	ON

オペランド

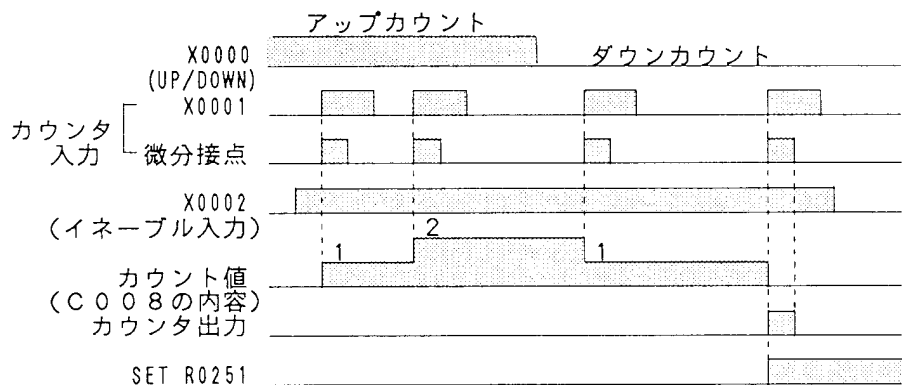
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	定 数	修 飾
A	カウンタレジスタ																			○								○

プログラム例



動作

- a接点X0000が、ONの時アップカウント、OFFの時ダウンカウントとなります。
- イネーブル入力(X0002)がONの時、カウンタ動作が許可されます。
- a接点X0002(イネーブル入力)がON状態にある場合、カウンタ入力Cの立ち上がりでアップ方向またはダウン方向へのカウントを行います。
- カウント値がリミット値(この場合は0)に達した時、カウンタ出力QがONします。



補足

- カウンタ入力Cに微分接点|↑|を入れないとスキャン毎にカウントしてしまいます。

FUN150 診断表示 (DIAG)

エラーメッセージの表示を可能にします。

関連命令

診断表示リセット
ASCII 変換

条件入力

登録個数オーバー出力
および実行出力

-[DIAG A B]-

機能

- 条件入力が OFF から ON に変化したときに、オペランド A が指定するエラーコードを内部エラーコードリストとアナンシエータテーブルに登録します。このとき、エラーが発生しなければ、1 スキャンのみ出力は OFF となります。以後入力が ON のままであれば、出力も ON のままです。
- エラーコードが前回登録したものと同一場合でも、内部エラーコードリストに登録します。
- 内部エラーコードリストとアナンシエータテーブルは特殊リレー/レジスタの固定アドレスに割り当てられています。
- エラーコードには ASCII コード 12 文字分のエラーメッセージを付けることができます。このエラーメッセージはオペランド B を先頭とする 6 個のレジスタが使用されます。このメッセージは TPD の「システム情報」の「9. 診断表示」またはコンピュータリンク機能を使用して、最初に登録したエラーを確認できます。
- オペランド A (エラーコード) の指定範囲は 1 ~ 64 です。この範囲を越えるとエラーとなります。
- 登録可能なエラーコード数は最大 16 個です。16 個以上登録しようとするエラーとなります。
- 本命令はユーザ側で診断プログラム (工程渋滞やイリーガルステートの診断など) を作成するときに役に立ちます。

入力	処 理	出力
OFF	不実行	OFF
OFF から ON	1 ≤ エラーコード ≤ 64 かつ エラーコード登録個数 < 16 の時、実行。 エラーコード > 64、または エラーコード登録個数 ≥ 16 の時、不実行。	OFF ON
ON	不実行	ON
ON から OFF	不実行	OFF

オペランド

		デバイス										レジスタ										INDEX			定修				
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
A	エラーコード											○	○	○	○	○	○	○	○	○	○				○	○	○	○	○
B	メッセージ先頭											○	○	○	○	○	○	○	○	○	○								○

プログラム例



動作

- a 接点 R0001 が OFF から ON に変化したとき、オペランド A で指定したエラーコード 02 が内部エラーコードリストの最後 (SW021) に登録されます (このリストにはすでに 3 個のエラーコードが登録されています)。この時登録個数オーバー出力を OFF にします。
- 同時に、このエラーコードに対応するアナンシエータテーブル (SW034 ~ SW037) 内の対応するビット (この場合はビット 2) を ON します。
- エラーコード 02 と、オペランド B で指定した D1020 を先頭とする 6 個のレジスタ (D1020 ~ D1025) の内容 (エラーメッセージ) との対応づけを行います。
- R0001 が OFF の時は、本命令を実行せず、登録個数オーバー出力を OFF にします。
- エラーコードリストへの登録時に、エラーコード登録個数が 16 以上になると登録個数オーバー出力 Y035A が 1 スキャンだけ ON になります。
- 登録個数オーバー出力を使用しないときは、ON 時微分接点は不要です。

内部エラーコードリスト

SW016	15
SW017	4
SW018	15
SW019	6
SW020	8
SW021	2
...	...
SW032	15
SW033	16

カレントエラーコード

登録個数 (3→4)

エラーコードに対応させるエラーメッセージ (ここでは、LIMIT OVER)

←最新のエラーコード

エラーコード
リスト

エラーメッセージレジスタ

上位	下位	
I (49H)	L (4CH)	D1020
I (49H)	M (4DH)	D1021
(20H)	T (54H)	D1022
V (56H)	O (4FH)	D1023
R (52H)	E (45H)	D1024
(20H)	(20H)	D1025

エラーコードに
対応するビット

アナンシェータテーブル

SW034	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
SW035	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17
SW036	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33
SW037	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49

※網掛け部 () のビットオン

補足

- 内部エラーコードリストとアナンシェータテーブルの内容の削除は、診断リセット命令 (D I A R) を使用してください。
- ただし、同一のエラーコードが内部エラーコードリストに登録されている時は、先に登録されたものが削除されますが、まだエラーコードが残っていますので、アナンシェータテーブルの内容はONのままです。

診断表示リセット

FUN151 診断表示リセット (DIAR)

DIAG命令で設定したエラーメッセージの表示を取り消します。

関連命令
DIAG

条件入力

エラー出力または実行出力

-[DIAR A]-

機能

- 条件入力がOFFからONに変化したときに、オペランドAが指定するエラーコードを内部エラーコードリストとアナンシエータテーブルから削除します。このとき、エラーが発生しなければ、1スキャンのみ出力はOFFとなります。以後入力がONのままであれば、出力もONのままです。
- 同じエラーコードが複数存在する場合は、先頭に近いエラーコードだけを内部エラーコードリストから削除します。この時、アナンシエータテーブルの対応ビットはONのままです。
- 内部エラーコードリストとアナンシエータテーブルは特殊リレー/レジスタの固定アドレスに割り当てられています。
- オペランドA（エラーコード）の指定範囲は1～64です。この範囲を越えるとエラーが発生します。
- 以下のエラーが発生した時には、エラー出力をONにします。その後入力がONのままであれば、出力もONのままです。内部エラーコードリストに対する登録個数が0。エラーコードが内部エラーコードリストに登録されていない。エラーコードが1～64の範囲内に無い。

入力	処 理	出力
OFF	不実行	OFF
OFFからON	1 ≤ エラーコード ≤ 64 かつ エラーコード登録がある時、 実行。 エラーコード > 64、または エラーコード登録個数 = 0、 エラーコードが登録されてい ない時、不実行。	OFF ON
ON	不実行	ON
ONからOFF	不実行	OFF

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A	エラーコード											○	○	○	○	○	○	○	○	○	○				○	○	○	○	○

プログラム例



動作

- a接点R0002がOFFからONに変化すると、オペランドAで指定したエラーコード06を内部エラーコードリストから削除し、削除した位置より下にあるエラーコードを順に繰上げ、エラーコード登録個数を3に変更します。アナンシエータテーブル内の対応ビット6をリセット（OFFに）します。
- a接点R0002がOFFの時は、本命命令を実行せず、出力をOFFにします。
- オペランドAで指定したエラーコードがカレントエラー（SW016）の内容と同じであった場合、SW018の内容を削除すると共に、新しくSW018の内容となるSW019の内容がカレントエラー（SW016）としてセットされます。
- エラーが発生した場合には1スキャンだけ出力コイルのY035AがONになります。
- エラー出力を使用しないときはON時微分接点は不要です。

命令実行前の
内部エラーコードリスト

SW016	15	カレントエラーコード
SW017	04	登録個数
SW018	15	1
SW019	06	2
SW020	08	3
SW021	02	4
⋮	⋮	⋮
⋮	⋮	⋮
⋮	⋮	⋮
SW033		16

エラーコード
リスト

命令実行後の
内部エラーコードリスト

15	←登録個数が変化(04→03)
03	
15	
08	←コード06が削除され、
02	次以降のコードが繰り
⋮	上がります(同時に、
⋮	アナンシェータテーブ
⋮	ルのビット6がリセッ
⋮	トされます)

補足

- 内部エラーコードリストとアナンシェータテーブルに登録されたエラーコードのリセットは、プログラマからのリセット操作によっても可能です。ただし、この場合は、エラーコードリストの先頭レジスタ(SW018)の内容だけがリセットされます。先頭レジスタの内容がカレントエラー(SW016)に表示されるので、プログラマ画面に表示されるエラーメッセージも変わります(プログラマによるリセット操作は、上記の例の場合には結果的には[D1AR15]の実行と等しくなります)。

ステータスラッチセット

FUN 152 ステータスラッチセット (STLS)

デバイス／レジスタデータをラッチします。

関連命令

ステータスラッチリセット

条件入力

—[STLS]—

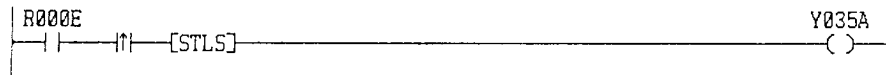
実行出力

機能

- 条件入力がOFFからONに変化した時、ラッチ状態がREADYであるとき、レジスタ／デバイスのデータをラッチします。このとき、出力を1スキャンだけONにします。ラッチ状態がLATCHであるとき、すでにデータはラッチされているので、ラッチ動作は行わず出力を1スキャンだけOFFにします。
- 条件入力が2スキャン以上ONであるときは、上記の出力を1スキャン出力した後にONの状態を続けます。
- ラッチ対象となるデバイス／レジスタはプログラマで設定します。(最大32)。

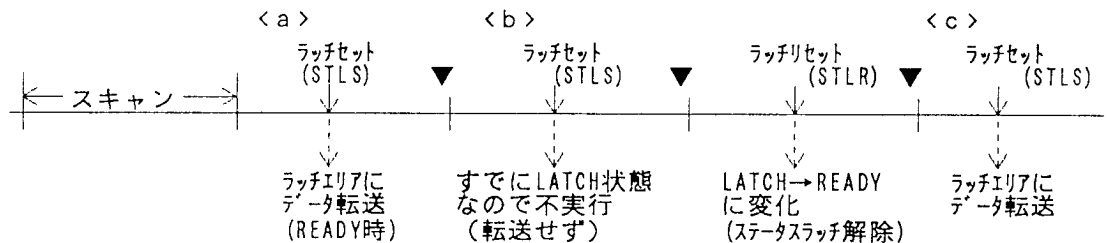
入力	処 理	出力
OFF	不実行	OFF
OFFからON	ラッチ状態がREADYである時、実行	ON
	ラッチ状態がLATCHである時、不実行	OFF
ON	不実行	ON
ONからOFF	不実行	OFF

プログラム例



動作

- a 接点R000EがONすると、ラッチステータスが“READY”であれば、プログラマ側で設定済みの最大32個のデバイス／レジスタの内容をラッチし、出力Y035AをONにします。この時ラッチステータスは“LATCH”に変わります（下図の<a>または<c>の位置です）。
ラッチステータスがすでに“LATCH”状態である場合は、STLS命令ではラッチを実行せず、以前にラッチしたデータとなります。このとき、出力Y035AはOFFになります（下図のの位置です）
- 実行出力を使用しない時は、ON時微分接点は不要です。



補足

- 上記プログラム例では、STLS命令の前にON時微分接点命令がありますので、実行時の出力は1スキャン出力されます。このON時微分接点命令が無い場合には、1スキャンのみ実行結果が出力され、その後は条件入力と同じ状態になります。
- ラッチしたデータは、ステータスラッチセット(STLR:FUN 153)命令を実行するかプログラムでラッチリセットを行うまで、保持されます。また、HALT→RUNの時にも、ラッチデータはリセットされます。
- プログラムでセットするラッチ条件とステータスラッチセット命令の併用は可能です。この場合、先に実行されたときのデータがラッチされます。なお、プログラムでセットしたラッチ条件で、ラッチが行われた時のデータは、ラッチ条件が成立したスキャンの終わりのデバイス/レジスタの内容です(上図の▼の位置です)。また、ステータスラッチセット命令を使用した場合は、その命令が実行された時点でのデータがラッチされます。プログラムによってはラッチされるデータが異なることがありますので、ご注意下さい。

ステータスラッチリセット

FUN 153 ステータスラッチリセット (STLR)

ラッチ状態の解除を行います。

関連命令

ステータスラッチセット

条件入力

—[STLR]—

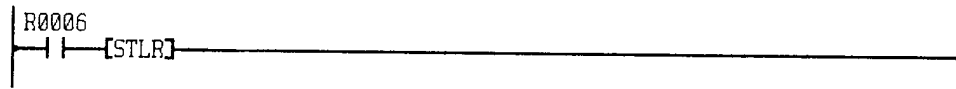
実行出力

機能

- 入力がOFFからONに変化すると、ラッチステータスの内容に関係なく、STLS命令あるいはプログラマによるラッチ状態を解除し、実行出力をONにします。入力がOFFのときは出力をOFFにします。
- 条件入力がONの時、常に出力はONです。

入力	処 理	出力
OFF	不 実 行	OFF
ON	OFFからONに立ち上がり時、実行	ON

プログラム例



動作

- a接点R0006がOFFからONに変化した時、ラッチステータスが“LATCH”または“READY”のどちらの場合でも、ラッチ状態を解除し、ラッチステータスを“READY”に設定します。設定後、実行出力をONにします。

補足

- ラッチエリアでのステータスの保持は、ステータスラッチリセット（プログラマ側のラッチリセットまたは、Fun 153のSTLR命令）が実行されるまで継続します。
- ステータスをリセットするためには、プログラマのキー操作による方法とラッチリセットコマンドの使用による方法の2種類があります。

FUN 154 カレンダー設定 (CLND) *

カレンダーを設定します。

関連命令

条件入力	エラー出力
-[A CLND]	

条件入力	エラー出力
-[A CLND]-----	

条件入力	エラー出力
-[A CLND]	

機能

- ・オペアンプが正常に動作しない場合、エラーメッセージが表示されます。

入力	処 理	出力
OFF	不実行	OFF
ON	実行：正常	OFF
	不実行：異常	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	入力パラメータ先頭アドレス											○	○	○	○	○	○	○	○	○	○							○

プログラム例

R0137 H0091
[RW306 CLND]

動作

- ・ a 接点 R 0 1 3 7 が ON すると、レジスタ RW 3 0 6 ~ RW 3 1 1 の内容に基づいてカレンダーを設定し、出力を OFF します。ただし、設定データが指定範囲外の数値であった場合は演算を実行しないで、エラー出力を ON にします。
- ・ a 接点 R 0 1 3 7 が OFF の時は、演算を実行せずに、出力を OFF にします。

カレンダー設定データ格納レジスタ

RW306	00H	年
RW307	00H	月
RW308	00H	日
RW309	00H	時
RW310	00H	分
RW311	00H	秒

データの範囲 (BCDコード2桁)
00～99 (1990～2089)

00~99 (1990~2089)

補足

- ・曜日は次のようになります。

データ	00	01	02	03	04	05	06
曜日	日	月	火	水	木	金	土

なお曜日は自動設定となっていますので設定の必要はありません。

- ・ T 2 の C P U モジュールのうち、カレンダー機能の内蔵されていない P U 2 1 4 ではカレンダー設定はできません。

カレンダー演算

FUN155 カレンダー演算 (CLDS)*

現在の時刻とカレンダー設定データとの差を求めます。

関連命令

条件入力

エラー出力

[A CLDS B]

機能

- 現在の時刻から、レジスタAを先頭とする6個のレジスタに格納されているカレンダー設定データを減じて、その結果をレジスタBを先頭とする6個のレジスタに格納します。
- カレンダー設定データが指定範囲外の数値の場合は演算を実行しないでエラー出力をONします。
- 演算結果が指定範囲内の数値であればエラー出力をOFFに、指定範囲外の数値であればエラー出力をONに設定します。

入力	処理	出力
OFF	不実行	OFF
ON	実行：正常	OFF
	不実行：異常	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	設定データ											○	○	○	○	○	○	○	○	○	○							○
B	演算結果												○	○	○	○	○	○	○	○	○							○

プログラム例

```
R0137 H0091 H0000
├──┤ [RW023 CLDS D0156]
```

動作

- a 接点X0003がONすると、PC本体で管理している現在の時刻からレジスタRW023～RW028に格納されているカレンダーデータの差を算出し、その結果をレジスタD0156～D0161に格納します。設定データまたは演算結果が指定範囲外の数値の場合、エラー出力をONに設定します。
- a 接点X0003がOFFの時は、演算を実行せず、出力をOFFにします。

		カレンダー設定データ						演算結果					
		15	8	7	0			15	8	7	0		
現在時刻	—	RW023	00H		年			D0156	00H		年		
		RW024	00H		月			D0157	00H		月		
		RW025	00H		日			D0158	00H		日		
		RW026	00H		時			D0159	00H		時		
		RW027	00H		分			D0160	00H		分		
		RW028	00H		秒			D0161	00H		秒		

使用可能なデータの範囲：BCD2桁の数値

年：00～99 月：01～12 日：01～31 時：00～59 分：00～59 秒：00～59
(1990～2089)

補足

- T2のCPUモジュールのうち、カレンダー機能の内蔵されていないPU214では、カレンダー演算機能を使用することはできません。

本質継承型 P I D

FUN 1 5 6 本質継承型 P I D (P I D 3)

本質継承型 P I D 演算を実行します。

関連命令

条件入力

[A PID3 B → C]

判定出力

機能

- オペランド B で指定するレジスタを先頭とする 1 3 個のレジスタに格納されているパラメータに基づいて、オペランド A で指定するレジスタを先頭とする 6 個のレジスタに格納されている入力データ及び設定値について、本質継承型（不完全微分先行型） P I D 演算を以下のように行い、その結果（操作出力・内部操作量・前回微分項、現在値、設定値等）をオペランド C で指定するレジスタを先頭とする 1 0 個のレジスタに格納します。

入力	処 理		出力	ERF
OFF	初期化		OFF	—
ON	演算実行	内部カウンタ C = n	ON	
	演算不実行	内部カウンタ C ≠ n	OFF	

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	入力データ先頭											○	○	○	○	○	○	○	○	○	○							○
B	パラメータ先頭											○	○	○	○	○	○	○	○	○	○							○
C	出力データ先頭												○	○	○	○	○	○	○	○	○							○

桁指定はサポートしません。

データの構成

A レジスタ

A	プロセス測定値入力 PVC
A+1	A モード設定値 ASV
A+2	C モード設定値 CSV
A+3	M モード操作入力 MMV
A+4	MV トラッキング入力 TMV
A+5	モード指定 MODE
A モード:オートモード	
C モード:カスケードモード	
M モード:マニュアルモード	

B レジスタ

B	比例ゲイン K p
B+1	積分時間 T I
B+2	微分時間 T D
B+3	制御不感帯 G P
B+4	A モード初期値 ISV
B+5	入力フィルタ係数 FT
B+6	SV 変化率リミット DSV
B+7	MMV 変化率リミット DMMV
B+8	初期ステータス STS
B+9	MV 上限リミット MH
B+10	MV 下限リミット ML
B+11	MV 変化率リミット DMV
B+12	制御周期設定値 n

C レジスタ

C	操作出力 MV
C+1	前回偏差 e _{n-1}
C+2	前回微分項 D _{n-1}
C+3	前回現在値 P V _{n-1}
C+4	前回設定値 S V _{n-1}
C+5	積分項剰余 i _r
C+6	微分項剰余 D _r
C+7	今回内部操作 M V _n
C+8	内部カウンタ C
C+9	制御周期 Δ t

補足

本命令は、必ずスキャンモードを定刻スキャンに設定して、メインプログラムでのみ使用して下さい。

[計算式]

(1) デジタルフィルタ部

$$PV_n = (1 - FT) \cdot PVC + FT \cdot PV_{n-1} \quad (\text{初期値は、} PV_0 = PV_{0-1} = PVC) \\ (0.000 \leq FT \leq 0.999)$$

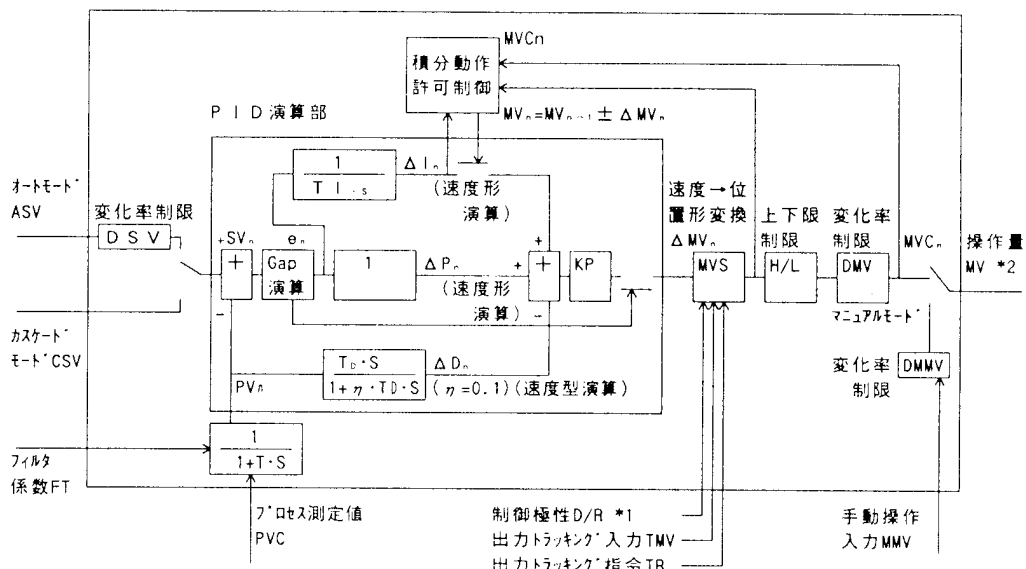
(2) PID部

$$\Delta MV_n = K_p (\Delta P_n + \Delta I_n + \Delta D_n) \quad (L_{n-1} \times \Delta I_n > 0 \cdots \text{積分停止}) \\ (|SV_n - PV_n| < |GP| \text{の時、} \Delta MV_n = 0) \\ (ダイレクト指定時：減算、リバース指定時：加算) \\ (初期値は、} MV_0 = MV) \\ MV_n = MV_{n-1} \pm \Delta MV_n$$

ここで

$$\Delta P_n = e_n - e_{n-1} \quad (\text{初期値は、} e_{0-1} = e_0) \\ e_n = SV_n - PV_n \quad (|SV_n - PV_n| < |GP| \text{の時、} e_n = 0) \\ \Delta I_n = \frac{e_n \cdot \Delta t + I_r}{T_I} \quad (T_I = 0 \text{の時は} \Delta I = 0 \text{とします。} (\text{積分停止})) \\ \Delta t = (\text{定刻スキャン時間設定値}) \times n \\ \Delta D_n = \frac{T_D (PV_{n-1} - PV_n) - \Delta t \cdot D_{n-1} + D_r}{\Delta t + \eta \cdot T_D} \\ D_n = D_{n-1} + \Delta D_n \quad (\text{初期値は、} D_0 = D_{0-1} = 0) \\ \eta = 0.1 \text{ (固定値)} \\ L_{n-1} = MV_{n-1} - MV \quad (\text{初期値は、} MV_{0-1} = MV)$$

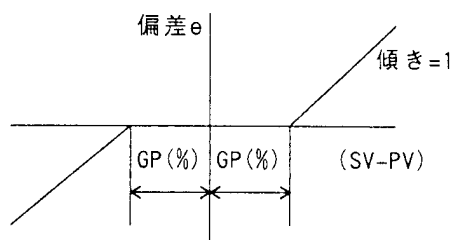
[機能ブロック]



*1 速度→位置形変換における制御極性D/R指定はダイレクト指定時は減算処理、リバース指定時は加算処理を行います。

*2 操作量出力MVの出力極性切替は、外部ロジック(ラダー回路)にて行って下さい。

(Gap演算)



(微分動作制御)
 $|SV_n - PV_n| < |GP|$
 の時 $\Delta MV_n = 0$

(積分動作制御)
 $L_n = MV_n - MV$
 $L_{n-1} \times \Delta I_n > 0 \cdots \text{積分停止}$
 $L_{n-1} \times \Delta I_n \leq 0 \cdots \text{積分許可}$

データ範囲

A レジスタ

A	: プロセス測定値入力 PVC [%]	(データ範囲: 0.00~100.00)
A+3	: オートモード設定値 ASV [%]	(データ範囲: 0.00~100.00)
A+2	: カスケードモード設定値 CSV [%]	(データ範囲: 0.00~100.00)
A+3	: マニュアルモード操作量入力 MMV [%]	(データ範囲: -25.00~125.00)
A+4	: 操作量トラッキング入力 TMV [%]	(データ範囲: -25.00~125.00)
A+5	: モード指定 MODE	

	15		12	11		8	7		4	3		0
A + 5											TR	MODE

MODE: 動作モード指定
<0><1>
0 0 : マニュアルモード
0 1 : オートモード
1 0 : カスケードモード
1 1 : (オートモード)

TR : トラッキング指令
0 : なし
1 : あり

B レジスタ

B	: 比例ゲイン Kp	(データ範囲: 0.00~327.67)
B+1	: 積分時間 TI [min]	(データ範囲: 0.000~32.767)
B+2	: 微分時間 TD [min]	(データ範囲: 0.000~32.767)
B+3	: 制御不感帯 GP [%]	(データ範囲: 0.00~10.00)
B+4	: オートモード設定値初期値 ISV [%]	(データ範囲: 0.00~100.00)
B+5	: プロセス入力フィルタ係数 FT	(データ範囲: 0.000~0.999)
B+6	: 設定値ASV変化率リミット DSV [%/Δt]	(データ範囲: 0.00~327.67)
B+7	: マニュアルモード操作量MMV変化率リミット DMMV [%/Δt]	(データ範囲: 0.00~327.67)
B+8	: ステータス S T S	

15	12	11	8	7	4	3	0	
B + 8							D/R	IMODE

IMODE: 初期モード指定
<0><1>
0 0 : マニュアルモード
0 1 : オートモード
1 0 : カスケードモード
1 1 : (オートモード)

D/R : 極性D/R指令
0 : タイレクト
1 : リバース

B+9	: 操作量上限リミット MH [%]	(データ範囲: -25.00~125.00)
B+10	: 操作量下限リミット ML [%]	(データ範囲: -25.00~125.00)

MH<MLの場合、MHの値はMLの値になります。

B+11	: 操作量変化率リミット DMV [%/Δt]	(データ範囲: 0.00~327.67)
B+12	: 制御周期設定値 n	制御周期 Δt=n×(スキャン時間)

n スキャンごとにPID演算を実行する。
n ≤ 0 のときは、n = 1 として実行する。

Cレジスタ

C	: 操作量 MV [%]	(データ範囲:-25.00～ 125.00)
C+1	: 前回偏差 e_{n-1}	(データ範囲:-327.68～327.67)
C+2	: 前回微分項 D_{n-1}	(データ範囲:-327.68～327.68)
C+3	: 前回現在値 PV_{n-1}	(データ範囲: 0.00～ 100.00)
C+4	: 前回設定値 SV_{n-1}	(データ範囲: 0.00～ 100.00)
C+5	: 積分項剰余 I_r	
C+6	: 微分項剰余 D_r	
C+7	: 内部操作 MV_n	(データ範囲:-25.00～ 125.00)
C+8	: 内部カウンタ C	

このカウンタはスキャン回数をカウントするもので、 $C=n$ から毎スキャン、デクリメントします。
 $C \leq 0$ になるとP I D演算を実行し、再び $C=n$ とします。

- C+9: 制御周期 $\Delta t \leq [S]$ (データ範囲: 000.00～327.67)
 制御周期 Δt は、P Cに設定した定刻スキャン時間の n 倍になります。(nはB+12のレジスタで設定)
 (P Cの定刻スキャン時間は、10～200msの範囲で10ms単位で設定できます。)

[動作説明補足]

- (1) 操作量初期値は、前回までの最終値 MV を出力します。
- (2) MV トラッキング
 オートモードとカスケードモードでのみ有効な機能です。(マニュアルモードではこの指定は無効です。)
 トラッキング指令 TR がONの時、操作量 MV の上下限リミットは有効ですが、 MV の変化率制限は無効です。この時、内部 MV_n は $MV_n = MV$ とします。
 そして、トラッキング指令がOFFへ切替わる時、その時点の MV_n の値から ΔMV の加算/減算を開始します。
- (3) オートモード
 ASV の変更は、オートモードとマニュアルモードで有効です。
 オートモードになった時、 SV_n を変化率制限をしながら ASV に漸近させます。

$$SV_n = SV_{n-1} \pm |DSV|$$
 また、オートモードからマニュアルモードへの切替えがバンプレスとなるように $MMV = MV$ となります。
 よって、オートモード中の MMV の変更は行えません。
 操作量 MV の上下限リミット、変化率制限はともに有効です。
- (4) カスケードモード
 CSV の変更はすべてのモードで有効です。
 カスケードモードになったとき、 $SV_n = CSV$ となります。この時、操作量 MV の上下限リミット、変化率制限はともに有効です。
 また、カスケードモードからオートモード及びマニュアルモードへの切替えがバンプレスとなるように $ASV = SV_n$ 、 $MMV = MV$ となります。
 よって、カスケードモード中の ASV 、 MMV の変更は行えません。
- (5) マニュアルモード
 MMV の変更はマニュアルモードでのみ有効です。
 マニュアルモードになったとき、 MV を変化率制限しながら MMV に漸近させます。

$$MV = MMV \pm |DMMV|$$
 操作量 MV の上下限リミット、変化率制限はともに無効です。
 また、マニュアルモードからオートモードまたはカスケードモードへの切替えがバンプレスとなるように $MV_n = MV$ となります。
 そしてオートモードまたはカスケードモードへの切替え時には、その時点の MV_n の値から ΔMV の加算/減算を開始します。
- (6) 制御極性D/R指定
 ダイレクト指定では、 PV が増加した場合に MV を増加させます。(ΔMV の減算)

$$MV_n = MV_{n-1} - \Delta MV$$
 リバース指定では、 PV が増加した場合に MV を減少させます。(ΔMV の加算)

$$MV_n = MV_{n-1} + \Delta MV$$

(7) マニュアルモード操作量設定値MMV入力の変化率制限 DMMV

MVをMMVに変化率DMMVで漸近させます。

1. $+|DMMV| \geq (MMV-MV) \geq -|DMMV|$ の場合 : $MV=MMV$
2. $(MMV-MV) > +|DMMV|$ の場合 : $MV=MV+|DMMV|$
3. $(MMV-MV) < -|DMMV|$ の場合 : $MV=MV-|DMMV|$

(8) オートモード設定値ASV入力の変化率制限 DSV

SV_nをASVに変化率DSVで漸近させます。

1. $+|DSV| \geq (ASV-SV_{n-1}) \geq -|DSV|$ の場合 : $SV_n=ASV$
2. $(ASV-SV_{n-1}) > +|DSV|$ の場合 : $SV_n=SV_{n-1}+|DSV|$
3. $(ASV-SV_{n-1}) < -|DSV|$ の場合 : $SV_n=SV_{n-1}-|DSV|$

(9) 操作出力の上下制限制限 MH, ML

1. $MH < ML$ の場合 : $MH=ML$

上下制限出力をMVX_nとすると

2. $MV_n > MH$ の場合 : $MVX_n=MH$
3. $MV_n < ML$ の場合 : $MVX_n=ML$

(10) 操作出力の変化率制限 DMV

1. $+|DMV| \geq (MVX_n-MV) \geq -|DMV|$ の場合 : $MV=MVX_n$
2. $(MVX_n-MV) > +|DMV|$ の場合 : $MV=MV+|DMV|$
3. $(MVX_n-MV) < -|DMV|$ の場合 : $MV=MV-|DMV|$

(11) 命令入力OFF時

PIDの初期化動作を行います。内容は次の通りです。

1. 動作モードMODE(A+5のレジスタ)を初期モードIMODE(B+8のレジスタ)にて指定されたモードにします。
2. 操作量初期値は前回までの最終値MVを出力します。
3. オートモード設定値をオートモード初期値(B+4のレジスタ)に設定された値にする。

ASV=ISV

4. 内部データを以下のように初期化します。

MV₀=MV

MMV=MV

PV₀=PVC

SV₀=ASVまたはCSV(初期モードに依存。ただしマニュアルモード時はASVとします。)

D₀=0

I_r=0

D_r=0

C=n

e₀₋₁=e₀

(12) 命令入力ON時

PID動作を行います。

設定した制御周期n(B+12のレジスタ)スキャンごとにPID演算を行います。

(内部カウンタCが0以下なら実行し、新たに内部カウンタCをnに設定し、命令出力をONにします。内部カウンタが0より大きい場合、Cをデクリメントし、命令出力をOFFにします。)

[モードデータ関連表]

		設定値			操作量						
		オートモード設定値 ASV	カスケードモード設定値 CSV	設定値 SV ₀ (前回保持率)	マニュアルモード	トラッキング	内部操作量 MV ₀	操作量出力制限		制限出力 MVC ₀	出力操作量 MV
					設定操作量 MMV	設定操作量 TMV		上下限制限	変化率制限		
マニュアルモード		— (変更可能)	— (変更可能)	SV ₀ ... (前回保持率)	設定MMV	— (変更可能)	MV	×	×	MV	MMV←MV ₀
オートモード	MVトラッキング ="OFF"	設定ASV	— (変更可能)	ASV←SV ₀	MV (変更不可)	— (変更可能)	PID演算 結果	○	○	内部操作量 MV ₀ 出力制限	MVC ₀
	MVトラッキング ="ON"					設定TMV	MV	○	×		
カスケードモード	MVトラッキング ="OFF"	SV ₀ ... (変更不可)	設定CSV	CSV	MV (変更不可)	— (変更可能)	PID演算 結果	○	○		
	MVトラッキング ="ON"					設定TMV	MV	○	×		

プログラム例

アナログ入力モジュールから取り込んだデータ（0～2000を100%とする）を測定値入力PVC値としてオートモードによるPID演算を行い、演算結果の操作結果の操作出力MV値をアナログ出力モジュールから出力するプログラム例を示します。

（割り付け）

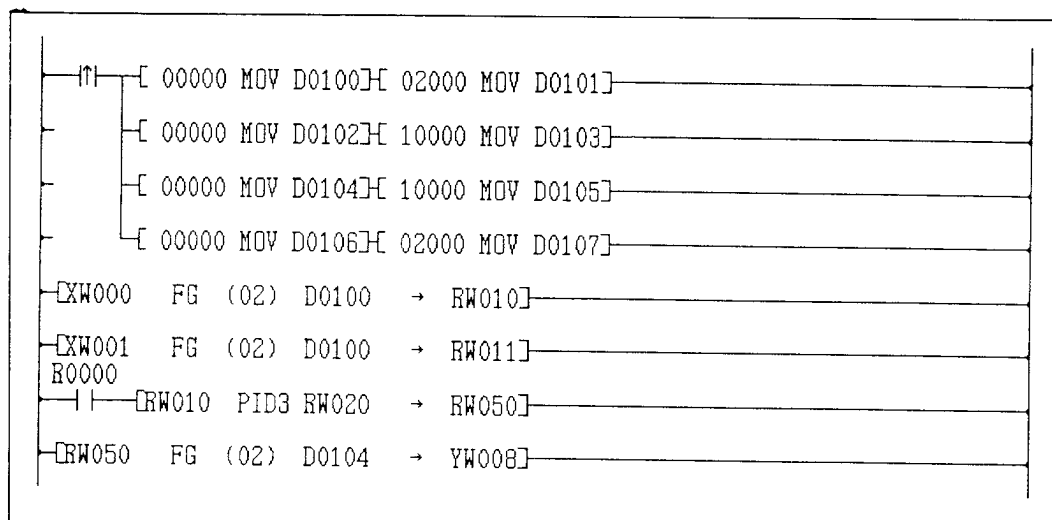
XW000 : アナログ入力データ（測定値入力）
 XW001 : アナログ入力データ（オートモード設定値）
 YW008 : アナログ出力データ（操作量）
 D0100～D0103 : アナログデータ→PID命令データ変換パラメータ（FG命令用）
 D0104～D0107 : PID演算結果操作量→アナログデータ変換パラメータ（FG命令用）
 RW010～RW015 : PID命令入力データ
 RW020～RW031 : PID命令パラメータデータ
 RW050～RW059 : PID命令出力データ
 R0000 : PID演算実行

（PID演算条件）

- ・制御周期：1秒（定刻スキャン時間は100msとします）
- ・オートモード指定、トラッキングなしとします。（RW015=1）

パラメータは以下の通りです。

レジスタ	データ	備 考
RW020	100	比例ゲイン=1
RW021	1000	積分時間=1分
RW022	0	微分時間=0
RW023	100	制御不感帯=1%
RW024	5000	オートモード設定値初期値
RW025	0	入力フィルタ係数=0
RW026	10000	設定値ASV変化率リミット=100%
RW028	5	初期モード=オートモード、制御極性=リバース
RW029	10000	操作量上限リミット=100%
RW030	0	操作量下限リミット=0%
RW031	10000	操作量変化率リミット=100%
RW032	10	制御周期=スキャン時間100ms×10=1秒



ドラムシーケンサ

FUN1 5 8 ドラムシーケンサ (DRUM)

カウント値に従ってステップを移行し、各データテーブルのデータを出力するドラムシーケンサです。

関連命令
カムシーケンサ

条件入力

最終ステップ実行出力

—[A DRUM (n) B → C (m)]—

機能

- オペランドB（カウント値）とカウント値設定テーブル（A + 2 n 以降）とを比較し、カウント値がカウント値設定テーブルの値未満の時はレジスタB + 1 で示されるステップに対応したデータ出力パターンテーブルAの内容を、オペランドCからmビットのテーブルに出力します。
- オペランドBがカウント値設定テーブルの値以上になると、ステップN 0 . を+ 1 し、次ステップに移行します。
- 最終ステップ（n - 1）でカウント値がカウント値設定テーブルの値以上になると、ステップ0に移行します。
- テーブルサイズ（ステップ数）の範囲は $1 \leq n \leq 256$ です。

出力ビット

入力	処 理	出力
OFF	イニシャライズ	OFF
0 N	B + 1 ≠ (n - 1)	OFF
	B + 1 = (n - 1) (最終ステップ時)	0 N

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	先頭レジスタ											○	○	○	○	○	○	○	○	○	○							○
n	テーブルサイズ																											○
B	カウントレジスタ												○	○	○	○	○	○	○	○	○				○	○		○
C	出力デバイス		○	○	○	○	○																					
m	ビット数																											○

データ出力パターン・カウント値設定テーブル

A+1	A	ステップ0の出力パターン
A+3	A+2	ステップ1の出力パターン
⋮	⋮	
A+2n-1	A+2n-2	ステップn-1の出力パターン
	A+2n	ステップ0の設定値
	A+2n+1	ステップ1の設定値
	⋮	
	A+3n-1	ステップn-1の設定値

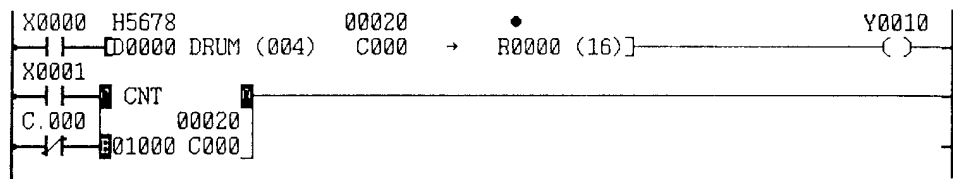
カウンタレジスタ・ステップN0.

B	カウンタレジスタ
B+1	ステップN0.

データ出力デバイス（最大32ビット）

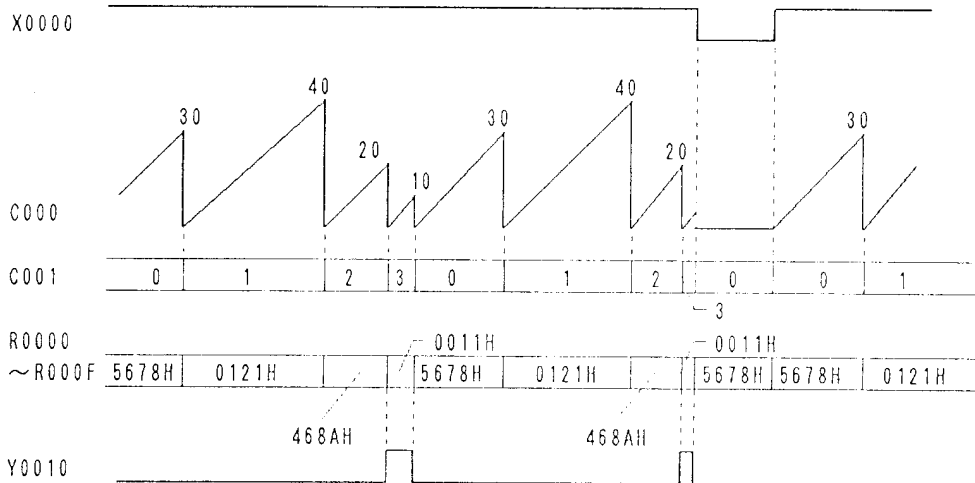
C+m-1	C

プログラム例



動作

- ・ a 接点 X 0 0 0 0 が O N になると、C 0 0 1 の値（初期値 = 0）によりステップ N O . を 0 にします。
- ・ カウンタレジスタ C 0 0 0 と、ステップ 0 の設定カウント値 D 0 0 0 8 (A + 2 n) とを比較し、C 0 0 0 が 3 0 未満の時は出力パターンテーブル D 0 0 0 1 ・ D 0 0 0 0 の下位 1 6 ビット (m = 1 6) を R 0 0 0 0 から始まる 1 6 ビットに格納します。
- ・ C 0 0 0 が 3 0 以上になると、C 0 0 1 を + 1 し、ステップ 1 に移行します。
- ・ ステップ 3 で C 0 0 0 が 1 0 (設定値) 以上になると、ステップ 0 に移行します。



ステップNO.	設定カウント値	出力パターン
0	D0008=30	D0001・D0000=12345678H
1	D0009=40	D0003・D0002=23400121H
2	D0010=20	D0005・D0004=A523468AH
3	D0011=10	D0007・D0006=FB120011H

- ・ 命令出力 (Y 0 0 1 0) は最終ステップ (ステップ 3) の時のみ O N になります。
- ・ a 接点 X 0 0 0 0 が O F F になると、カウンタレジスタ (C 0 0 0)、ステップ N O . (C 0 0 1) はクリアされ、出力 Y 0 0 1 0 は O F F になります。

補足

- ・ カウンタレジスタ B にはカウンタレジスタ C のほか、加算命令などが使用できます。
- ・ ステップの設定値が 0 の時は、次のステップに進みます。

カムシーケンサ

FUN 1 5 9 カムシーケンサ (CAM)

カウンタ値の増減に従ってデバイスをON/OFFするカムシーケンサです。

関連命令
ドラムシーケンサ

条件入力

実行出力

—[A CAM (n) B → C]——

機能

- オペランドBのカウンタ値に従い、オペランドCからC+n-1のデバイスをON/OFFします。
- デバイスのON/OFF制御は、オペランドAから始まるデータテーブルに設定される立ち上がり設定値、立ち下がり設定値に従って行われます。
- 設定できる対象デバイス数は64までです。(1 ≤ n ≤ 64)

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	先頭レジスタ											○	○	○	○	○	○	○	○	○	○								○
n	テーブルサイズ																											○	
B	カウンタレジスタ												○	○	○	○	○	○	○	○	○				○	○	○		○
C	先頭デバイス		○	○	○	○	○																						

立ち上がり／立ち下がりカウンタ設定 (1 ≤ n ≤ 64)

立ち下がり設定 立ち上がり設定値

A+1	A
A+3	A+2
⋮	⋮
⋮	⋮
A+2n-1	A+2n-2

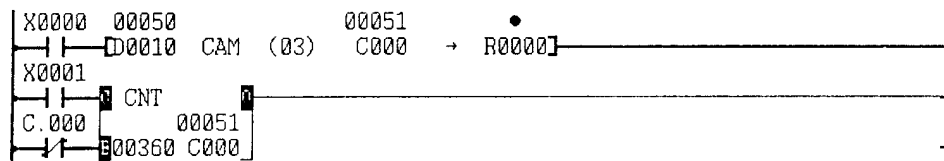
カウンタレジスタ

B

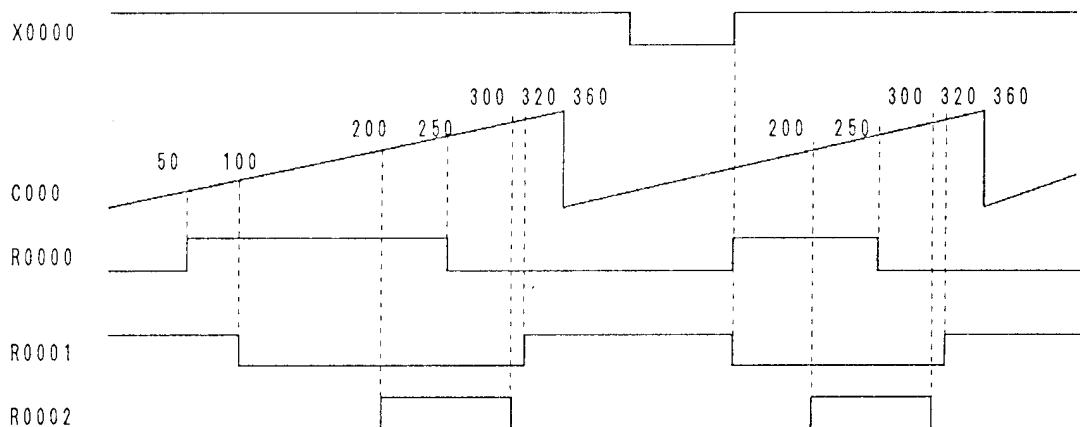
対象デバイス (1 ≤ n ≤ 64)

C
C+1
⋮
⋮
C+n-1

プログラム例



- 動作
- ・ a 接点 X 0 0 0 0 が O N の時、カウンタレジスタ C 0 0 0 と D 0 0 1 0 を先頭とする 6 ワードのテーブルの内容を比較し、デバイス R 0 0 0 0、R 0 0 0 1、R 0 0 0 2 を O N / O F F 制御します。



立上がり設定	立下がり設定	対象デバイス
D0010=50	D0011=250	R0000
D0012=320	D0013=100	R0001
D0014=200	D0015=300	R0002

- ・ $D0010 \leq C000 < D0011$ の時、R 0 0 0 0 を O N にする。
- ・ $D0013 \leq C000 < D0012$ の時、R 0 0 0 1 を O F F にする。
- ・ $D0014 \leq C000 < D0015$ の時、R 0 0 0 2 を O N にする。
- ・ a 接点 X 0 0 0 0 が O F F の時は比較出力は行わず命令出力も O F F になります。

データの上限リミットを設定します。

関連命令

リミットオーバー判定出力

$$-[A \quad UL \quad B \rightarrow C] \text{-----}$$
[illegible]

入力	処 理	出力
OFF	不実行	OFF
O N	実行 : $A \leq B$	OFF
	実行 : $A > B$	O N

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾				
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K		
A	被演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○	
B	上限リミット値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
C	リミット結果		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○		○	○	○	○	○

R0104 18435 15000 15000 Y035A
UL D1234 → RW021

- ・ a 接点 R0104 が ON すると、レジスタ RW018 の内容 (18435) と上限リミット値 D1234 の内容 (15000) を比較し、上限リミット値 15000 を RW021 に格納すると共に、リミットオーバー判定出力を ON に設定します。
- ・ a 接点 R0104 が OFF の時は、演算を実行せず、出力を OFF に設定します。

[illegible]

FUN 1 6 1 下限リミット (LL) *

データの下限リミットを設定します。

関連命令

条件入力	リミットオーバー判定出力
- [A LL B → C] _____	

$$\neg [A \text{ LL } B \rightarrow C] \text{-----}$$

機能

- ・オペランドAで指定する被演算データとオペランドBで指定する下限リミット値を比較し、以下の演算を行います。
 - 被演算データ $\geq$ 下限リミット値の時：
 - 被演算データをオペランドCで指定する場所に格納し、リミットオーバー判定出力をOFFに設定します。
 - 被演算データ $<$ 下限リミット値の時：
 - 下限リミット値をオペランドCで指定する場所に格納し、リミットオーバー判定出力をONに設定します。
- ・被演算データと下限リミット値の範囲は-32768～32767です。

オペランド

- オペランド

入力	処 理	出力
OFF	不実行	OFF
O N	実行 : $A \geq B$	OFF
	実行 : $A < B$	O N

		デバイス										レジスタ										INDEX			定数	修飾			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A	被演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
B	下限リミット値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
C	リミット結果		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○

プログラム例

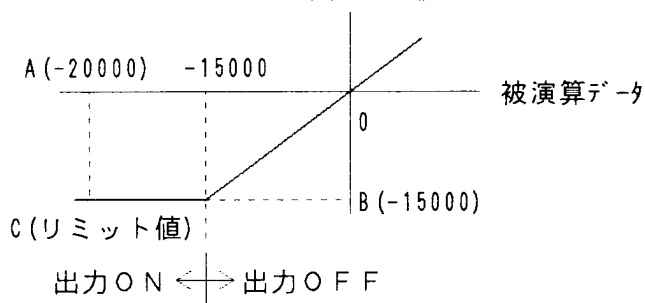
R0105	-20000	-15000	-15000	Y035A	
└─┴─┘	RW019	LL	RW020	→ RW022	└─┴─┘

動作

- ・ a 接点 R0105 が ON すると、レジスタ RW019 の内容 (- 20000) と下限リミット値 RW020 の内容 (- 15000) を比較し、下限リミット値 - 15000 を RW022 に格納すると共に、リミットオーバー判定出力を ON に設定します。
- ・ a 接点 R0105 が OFF の時は、演算を実行せず、出力を OFF に設定します。

- ・ a 接点 R0105 が ON すると、レジスタ RW019 の内容 (−20000) と下限リミット値 RW020 の内容 (−15000) を比較し、下限リミット値 −15000 を RW022 に格納すると共に、リミットオーバー判定出力を ON に設定します。
- ・ a 接点 R0105 が OFF の時は、演算を実行せず、出力を OFF に設定します。

下限リミット値



FUN 1 6 2 最大值 (MAX) *

最大値とその格納位置を算出します。

関連命令

条件入力

実行出力

— [A MAX (n) B] —————

機能

- ・カン格の次す
 ・タラの値のま
 ・ジョ大のす
 ・レのし、最
 ・の探内レに
 ・個値ブルN
 ・n値1～64
 ・す大一定は
 ・と最て指力
 ・頭らたで出
 ・先かまB行
 ・タの中、ド
 ・タのし、ド
 ・スル格納後
 ・ジブル格バ
 ・レーに才実
 ・すのタを。イ
 ・定]ジンま
 ・指nレいし
 ・で[るポ納一
 ・Aズす格テ
 ・ドイ定示に
 ・サ指をタ可
 ・ラ成B指置
 ・べ成B指置
 ・オらド納レ
 ・オ復若

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デバイス										レジスタ										INDEX			定数	修飾		
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	テーブル先頭											○	○	○	○	○	○	○	○	○	○	○						○
n	テーブルサイズ																											○
B	最大値格納先											○	○	○	○	○	○	○	○	○	○		○	○	○	○		○

プログラム例 | X0001 00100 20000

X0001 00100 20000
 | | | [RW100 MAX (10) RW200]

動作

- ・ a 接点 X 0 0 0 1 が ON すると、レジスタ RW 1 0 0 を先頭とする 1 0 個のレジスタから

- ・ a 接点X0001がONすると、レジスタRW100を先頭とする10個のレジスタから成るテーブルの中から最大値を探し、最大値の20000をRW200に、ポインタ7をRW201にそれぞれ格納し、実行出力をONに設定します。
- ・ a 接点X0001がOFFの時は、演算を実行せず、出力をOFFにします。

テーブル (n=10)

		ホ*イン	
RW100	100	0	
RW101	10000	1	
RW102	-10000	2	
.	10	3	
.	0	4	
.	200	5	
.	-300	6	
RW107	20000	7	→ 最大値
RW108	-30	8	
RW109	20	9	

	ホ*イン
RW200	20000
RW201	7

補足

- ・オペランドBにインデックスKを指定すると、ポインタデータ (B+1) は切り捨てられますのでご注意ください。

© 2006 The Authors
Journal compilation © 2006 Blackwell Publishing Ltd

関連命令

実行出力

[illegible]

タ平格
のデシ
スに
スタ
レサ
グ
は、
64で
す。
1～
3276
8まで

頭納得
先格B。ン田
をにドすヲ
スプラしオ
ジーベ定ク
レテオ設ズ
るのをにイ
す」値N可
定nの0ル
指しそをプ
でズ、力一
アイし出テ
ドサ出行ナ
算能ン
ラ成を、可
べら値し定
オカ均納指

オペランド

プログラム例

X0002 00100 00200 Y035A
| | | [RW100 AVE (10) RW220] ————— ()

- ・ a 接点X0002がONすると、レジスタRW100を先頭とする10個のレジスタの内容の平均値を算出し、平均値200をRW220に格納し、実行出力をONに設定します。
- ・ a 接点X0002がOFFの時は、演算を実行せず、実行出力をOFFにします。

RW100	100
RW101	10000
RW102	-10000
.	10
.	0
.	200
.	-300
RW107	2000
RW108	-30
RW109	20

→ 平均值

RW220	200

- このテーブルを利用するためには、オペランドBで指定するレジスタを先頭とするn個の各レジスタには X_0 から X_{n-1} のn個の X_i 値 ($i=0 \sim n-1$) を、次のn個の各レジスタには X_0 から X_{n-1} に対応する関数値 Y_0 から Y_{n-1} のn個の Y_i 値 ($i=0 \sim n-1$) をそれぞれ前もって格納しておく必要があります。
- 任意のX値（入力データの値）に対応する関数F(X)値（オペランドCで指定するレジスタの内容）を以下のように計算します。

(1) $X_0 < \text{入力データの値} < X_{n-1}$ の場合

$$F(X) = (\text{入力データ} - X_{m-1}) \times \frac{Y_m - Y_{m-1}}{X_m - X_{m-1}}$$

ここで、 X_m 、 X_{m-1} は入力データの値を挟む X_i 値、 Y_m 、 Y_{m-1} は X_m 、 X_{m-1} に対応する Y_i 値です。

(2) 入力データの値 $\leq X_0$ の場合

$$F(X) = Y_0$$

(3) 入力データの値 $\geq X_{n-1}$ の場合

$$F(X) = Y_{n-1}$$

条件入力	オーバーフロー／アンダーフロー出力
-[A+1・A RT B]_____	

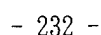
- ・オペランドA + 1・Aで指定する演算データ（倍長データ）の絶対値の平方根を求め、演算結果に演算データの符号を付加し、オペランドBで指定する場所に格納します。演算結果が32767を超えるときオーバーフロー、-32768より小さくなるとアンダーフローの状態となります。
- ・オーバーフローまたはアンダーフロー発生時には、リミット値の32767または-32768をオペランドBで指定する場所に格納し、オーバーフロー／アンダーフロー出力をONします。
- ・オペランドAで指定可能な演算データの範囲は-2147483648～2147483647です。

オペランド

プログラム例



- ・ a 接点 X 0 0 2 1 が O N すると、倍長レジスタ R W 0 1 1・R W 0 1 0 の内容 (- 1 0 3 2 5 6 8 4 3 2) の絶対値の平方根 3 2 1 3 3 を求め、一符号を付けてレジスタ D 1 0 1 2 に格納します。
- ・ a 接点 X 0 0 2 1 が O F F の時は、演算を実行せず、オーバーフロー／アンダーフロー出力を O F F にします。



関連命令

判定出力

機能

- [illegible]

オペランド

入力	処 理	出力
OFF	不実行	OFF
ON	M ≠ 0 : 正常	ON
	M = 0 : 不実行	OFF

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	演算データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
B	積分定数格納	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○						○
C	積分値格納先		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○	○						○

R0010 00005 00003 00066 Y035A
| | | D0777 INTG RW010 → RW020] ————— ()

- ・ a 接点 R0010 が ON すると、レジスタ RW010 の内容 (M=3) と RW011 の内容 (N=10) の積分定数を用いて、レジスタ D0777 の内容 (5) を積分します。積分結果から、積分値 Y=66 を RW020 へ、剰余 YR=2 を RW021 にそれぞれ格納し、出力を ON します。
- ・ a 接点 R0010 が OFF の時は、演算を実行せず、出力を OFF にします。

〔使用する積分式〕

積分定数 M
N

$$\text{積分値 } Y = Y_0 + \text{INT} \left(\frac{|N| * X + Y_{RQ}}{|M|} \right)$$

積分値 Y
剩餘 Y_F

$$\text{剩餘 } Y_R = (|N| * X + Y_{RQ}) - \text{INT} \left(\frac{|N| * X + Y_{RQ}}{|M|} \right) * |M|$$

ここで、

Y_0 、 Y_{E0} は前回演算時の Y 、 Y_E の値

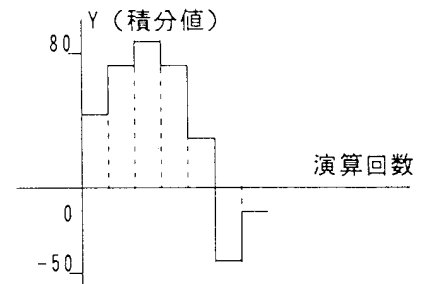
INT(a)は除算後の商を取り出す関数

例. $\text{INT}\left(\frac{50}{3}\right) = 16$ 、 $\text{INT}\left(\frac{18}{5}\right) = 3$

- 算出した積分値 $Y > 32767$ の時は、 Y に $7FFFH$ を、 Y_R に $FFFFH$ を格納し、実行出力を ON にします。積分値 Y が $Y < -32768$ の時は、 Y に $8000H$ を、 Y_R に $0000H$ を格納し、実行出力を ON にします。

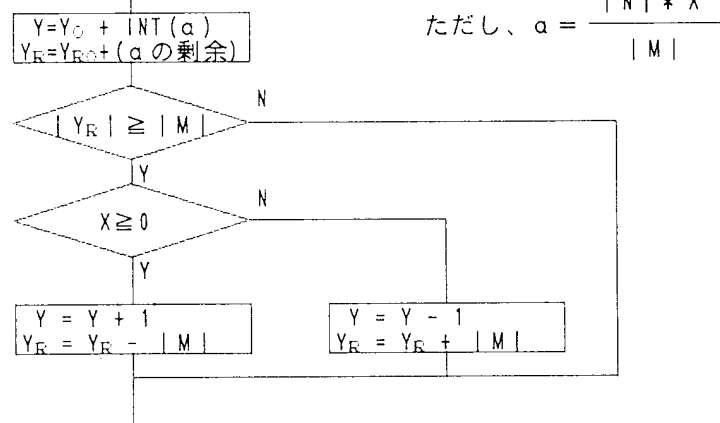
積分演算の結果

	X	M	N	Y	Y _R
	D0777	RW010	RW011	RW020	RW021
実行前の初期	5	3	10	50	0
1回目の演算	5	3	10	66	2
2回目の演算	5	3	10	83	1
3回目の演算	-5	3	10	67	-1
4回目の演算	-10	3	10	34	-2
5回目の演算	-23	3	10	-43	-1
6回目の演算	10	3	10	-10	0



注記

- NまたはXが0の場合、積分値Yと剰余Y_Rの値は変化しませんが、実行出力はONとなります。
- 本命令を実行するとき、Y_Rの値（実行前の初期値）は $-M < Y_R < M$ を満たすようにしてください。それは、Y、Y_Rの値が次の手順によって算出されることによります。



この手順から分かるように、剰余の積算値 $|Y_R|$ が除数 $|M|$ より大きくなると、積分値Yに+1（または-1）を加算し、 $|Y_R| < |M|$ になるように処理します。したがって、期待される積分値Yを得るためには、 $-M < Y_R < M$ （剰余の積算値 $|Y_R|$ が除数 $|M|$ より小）となる必要があります。

関連命令

完了出力

Copyright © 2006 John Wiley & Sons, Ltd.

- ・オペランドBで指定する位置を先頭とする連続した2つの格納場所に格納されているモード／レートMとレートNに基づいてラング関数演算を下記の要領で繰り返し実行し、1回の演算が終了したときにオペランドCで指定する場所を先頭とする連続した5つの格納場所の内容（現在値P、現在値の剰余Pr、増分K、増分の剰余Kr、目標値Sx）を更新し、現在値PがオペランドAで指定する目標値Siに到達すれば演算を完了し、完了出力をONします。
- ・指定可能な目標値の範囲、オペランドBで指定する場所に格納可能なデータの範囲は-32768～32767です。
- ・ラング関数は2つのモードがあります。1つは増分（ $|N|/M$ ）を一定とするレート指定、もう一つは目標値を達するまでのスキャン実行回数（N）を一定とするスキャン指定です。

入力	処 理	出力
OFF	不実行	OFF
O N	実行：未完了	OFF
	実行：完了	O N

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	目標値	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
B	モードデータ格納	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○						
C	現在値格納先		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○	○					

X0012 10000 00030 02000 Y035A
| | | D0123 RAMP RW018 → RW057 |—————()

- ・ a 接点X0012がONすると、レジスタRW057に事前に格納していた現在値の初期値P₀に増分K（レジスタRW059の内容）を加算し、これを現在値PとしてレジスタRW057に格納します。同様にして、現在値の剰余P_rに増分の剰余K_rを加え、結果をRW058に格納し、完了出力をOFFにします。この演算を繰り返し、現在値Pが目標値S（レジスタD0123の内容：10000）に到達したら完了出力をONします。
- ・ a 接点X0012がOFFの時は、演算を実行せず、完了出力をOFFにします。

```

graph TD
    Start([処理開始]) --> D1{S ≠ Sx}
    D1 -- N --> U1[スキャン指定モードでの  
パラメータ更新]
    D1 -- Y --> D2{モード・M ≥ 0}
    D2 -- N --> U1
    D2 -- Y --> U2[レート指定モードでの  
パラメータ更新]
    U1 --> U3[現在値Pの更新]
    U2 --> U3
    U3 --> End([処理終了])
  
```

[ランプ関数演算式]

- (1) 各指定モードの更新
初回の処理の時、または目標値Sが前回の処理時と違う
($S \neq S_x$) 場合に実行します。

- a) レート指定モードでのパラメータ更新 ($M > 0$ の時)
 $S_x = S$

$$K = \text{SIGN}(S_x - P_0) * \text{INT}\left(\frac{|N|}{M}\right)$$

$$K_r = |N| - \text{INT}\left(\frac{|N|}{M}\right) * M \quad \text{これは、}\left(\frac{|N|}{M}\right)\text{の剰余}$$

ここで、 $N=0$ の時は、 $K=0$ 、 K_r =前回処理時の K_r の値となります。

- b) スキャン指定モードでのパラメータ変更 ($M \leq 0$ の時)

$$S_x = S$$

$$K = \text{INT}\left(\frac{S_x - P_0}{|N|}\right)$$

$$K_r = (S_x - P_0) - \text{INT}\left(\frac{S_x - P_0}{|N|}\right) * |N| \quad \text{これは、}\left(\frac{S_x - P_0}{|N|}\right)\text{の剰余}$$

ここで、 $N=0$ の時は、 $K=S - P_0$ (P_0 : 前回処理時の K_r の値)、 $K_r=0$ となります。
また、このモードで ($S_x - P$) がオーバーフローかアンダーフローした場合は、 $|S_x - P| = H7FF$
Fのリミット値を用いて演算します。

上記の a)、b) の両モードの各式において、

$\text{SIGN}(a)$ は a の符号を取り出す関数です。
例. $\text{SIGN}(-10) = -1$ 、 $\text{SIGN}(0) = +1$ 、 $\text{SIGN}(53) = +1$
 $\text{INT}(b)$ は b で行われた演算の商を取り出す関数です。

$$\text{例.} \quad \text{INT}\left(\frac{17}{3}\right) = 5, \text{INT}\left(\frac{0}{3}\right) = 0, \text{INT}\left(\frac{21}{3}\right) = 7$$

また、上記の両モードで、 $N = -32768$ の場合は N の絶対値の $|N| = 32768$ を用いて演算します。
目標値 S_x と現在値 P_0 との差は $|S_x - P_0| \leq 32767$ の範囲で使用して下さい。

- (2) 現在値の更新

- a) レート指定モードで演算実行の時

$$\begin{aligned} &|Pr_0 + K_r| < M \text{ の場合 :} \\ &P = P_0 + K \\ &Pr = Pr_0 + K_r \end{aligned}$$

$$|Pr_0 + K_r| \geq M \text{ の場合 :}$$

$$\begin{aligned} &Pr_0 + K_r > 0 \text{ ならば、} \\ &P = P_0 + K + 1 \\ &Pr = M - (Pr_0 + K_r) \end{aligned}$$

$$\begin{aligned} &Pr_0 + K_r < 0 \text{ ならば、} \\ &P = P_0 + K - 1 \\ &Pr = M + (Pr_0 + K_r) \end{aligned}$$

b) スキャン指定モードで演算実行の時

$|Pr0 + Kr| < |N|$ の場合 (ただし、 $N \neq 0$) :

$$P = P0 + K$$

$$Pr = Pr0 + Kr$$

$|Pr0 + Kr| \geq |N|$ の場合 (ただし、 $N \neq 0$) :

$Pr0 + Kr > 0$ ならば、

$$P = P0 + K + 1$$

$$Pr = N - (Pr0 + Kr)$$

$Pr0 + Kr < 0$ ならば、

$$P = P0 + K - 1$$

$$Pr = N + (Pr0 + Kr)$$

ただし、 $N=0$ のときは以下ようになります。

$Pr0 \leq N$ の場合

$$P = P0 + K$$

$$Pr = 0$$

$Pr0 > N$ の場合

$$P = P0 + K + 1$$

$$Pr = 0$$

なお、上記の両モードの各式で次回の処理結果が以下になることが予測できる場合は、次回の処理を演算後、完了出力をOFFにします。

$P < Sx$ ($K \geq 0$ の時)

$P > Sx$ ($K < 0$ の時)

これは、剰余のパラメータを含む現在値P0と次回処理の増分Kの和が、次回処理の目標値Sxに達しない場合です。

また、次回の処理結果が以下になることが予測できる場合は、次回の処理結果が $P=Sx$ 、 $Pr=0$ となり、完了出力をONにします。

$P \geq Sx$ ($K \geq 0$ の時)

$P \leq Sx$ ($K < 0$ の時)

ランプ演算関数の実行例

(1) レート指定モードの場合

	S	M	N	P	Pr	K	Kr	Sx	出力
初期状態	10000	3	10	0	0	0	0	0	OFF
1回目の演算	10000	3	10	3	1	3	1	10000	OFF
2回目の演算	10000	3	10	6	2	3	1	10000	OFF
3回目の演算	10000	3	10	10	0	3	1	10000	OFF
4回目に演算	10000	3	10	13	1	3	1	10000	OFF
5回目の演算	0	3	10	10	0	-3	-1	0	OFF

(注) ここで
Sを0とする

(2) スキャン指定モードの場合

	S	M	N	P	Pr	K	Kr	Sx	出力
初期状態	-10000	-15	3	8000	0	0	0	0	OFF
1回目の演算	-10000	-15	3	2000	0	-6000	0	-10000	OFF
2回目の演算	-10000	-15	3	-4000	0	-6000	0	-10000	OFF
3回目の演算	-10000	-15	3	-10000	0	-6000	0	-10000	ON

	S	M	N	P	Pr	K	Kr	Sx	出力
初期状態	10000	-15	3	8001	0	0	0	0	OFF
1回目の演算	10000	-15	3	8667	1	666	1	10000	OFF
2回目の演算	10000	-15	3	9333	2	666	1	10000	OFF
3回目の演算	10000	-15	3	10000	0	666	1	10000	ON

	S	M	N	P	Pr	K	Kr	Sx	出力
初期状態	-10000	-3	0	-32	-1	0	0	0	OFF
1回目の演算	-10000	-3	0	-10000	0	-9968	0	-10000	ON

微分先行速度型 P I D 演算を実行します。	関連命令

判定出力

条件入力

判定出力

[A PID B → C]

・オペランドBで指定するレジスタを先頭とする7個のレジスタに格納されている比例係数 K_P 、積分係数 K_{IH} 、 K_{IL} 、微分係数 K_{DH} 、 K_{DL} 、ギャップ定数 G 、リミット定数 L とオペランドCで指定するレジスタに続く4個のレジスタに格納されている前回入力偏差 e_{-1} 、前回入力値 P_{-1} 、前々回入力値 P_{-2} 、剰余データ r （初期値=0）を用いて、オペランドAで指定するレジスタを先頭とする2個のレジスタに格納されている入力値 P 、設定値 S について、微分先行速度型PID演算を以下のように行い、その結果（出力データ ΔM 、前回入力偏差 e_{-1} 、前回入力値 P_{-1} 、前々回入力値 P_{-2} 、剰余データ r ）をオペランドCで指定するレジスタを先頭とする5個のレジスタに格納します。

入力	処 理	出力
OFF	不実行	OFF
0 N	実行 : KIHとKDL $\neq 0$	0 N
	不実行: KIHまたはKDL = 0	OFF

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A	入力データ先頭											○	○	○	○	○	○	○	○	○	○	○							○
B	ハラメータ先頭											○	○	○	○	○	○	○	○	○	○	○							○
C	出力データ先頭												○	○	○	○	○	○	○	○	○	○	○						○

R0030 00025 00001 00188 Y035A
 [R0030] [00025] [00001] + [00188] [Y035A]

- ・ a 接点 R0030 が ON すると、オペランド B で指定するレジスタを先頭とする 7 個のレジスタ RW200 の内容 ($K_P=1$)、RW201 の内容 ($K_{IH}=4$)、RW202 の内容 ($K_{IL}=10$)、RW203 の内容 ($K_{PH}=20$)、RW204 の内容 ($K_{DL}=5$)、RW205 の内容 ($G=0$)、RW206 の内容 ($L=100$) とオペランド C で指定するレジスタ (RW300) に続く 4 個のレジスタ (RW301~RW304) に格納されている前回入力偏差 e_{-1} (78)、前回入力値 P_{-1} (22)、前々回入力値 P_{-2} (20)、剰余データ r (0) を用いて、オペランド A で指定するレジスタ RW100 内容 ($P=25$) とし、RW101 の内容 ($S=100$) の入力データに対する微分先行速度型 PID 演算を実行し、その結果 ($\Delta M=180$ 、 $e_{-1}=75$ 、 $P_{-1}=25$ 、 $P_{-2}=22$ 、 $r=2$) をオペランド C で指定するレジスタを先頭とする 5 個のレジスタ (RW300~RW304) にそれぞれ格納します。演算後に実行出力を ON にします。
- ・ a 接点 R0030 が OFF の時は、演算を実行せず、出力を OFF としますが、 ΔM と r に 0 を、 e_{-1} に e ($=S-P$) の値を、 P_{-1} と P_{-2} に P の値を設定します。

B	比例係数	K_p
B+1	積分係数	K_{IH}
B+2	積分係数	K_{IL}
B+3	微分係数	K_{DH}
B+4	微分係数	K_{DL}
B+5	ギャップ定数	G
B+6	リミット定数	L

C	出力データ	ΔM
C+1	前回入力偏差	e_{-1}
C+2	前回入力値	P_{-1}
C+3	前々回入力値	P_{-2}
C+4	剰余データ	r

[微分先行速度型 P I D 演算]

$$\text{出力データ } \Delta M = K_p \cdot \{(e - e_{-1}) + \text{INT}\left(\frac{|K_{IL}| \cdot e + I_r}{|K_{IH}|}\right) + \text{INT}\left[\frac{|K_{DH}|}{|K_{DL}|} \cdot (2P_{-1} - P - P_{-2})\right]\}$$

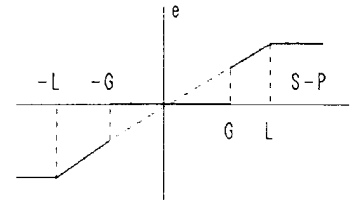
ここで、 e は入力偏差であり、 $(S-P)$ の値に対するリミット・ギャップ演算から求めます（下図参照）。また I_r は、

$$\text{INT}\left(\frac{|K_{IL}| \cdot e + I_r}{|K_{IH}|}\right) \quad (I_r \text{の初期値は} 0)$$

の剰余を表わしています。

また、 $\text{INT}(a)$ は除算 a から商を取り出す関数です。

例、 $\text{INT}\left(\frac{50}{3}\right) = 16$ 、 $\text{INT}\left(\frac{18}{5}\right) = 3$



- オペランド A で指定するレジスタに格納可能なデータの範囲は-32768～32767です。
- 算出した出力データ $\Delta M > 32767$ 、または $\Delta M < -32768$ の時は、オペランド C で指定するレジスタにはリミット定数 L を格納し、実行出力を ON にします。
- $K_{IH}=0$ 、または $K_{DL}=0$ の場合は、演算を実行せずに実行出力を OFF にし、エラーフラグ（除算エラー）をセットします。

偏差 2 乗速度型 P I D 演算を実行します。

[A PID2 B → C] _____

・ オペランドBで指定するレジスタを先頭とする7個のレジスタに格納されている比例係数 K_p 、積分係数 K_{IH} 、 K_{IL} 、微分係数 K_{DH} 、 K_{DL} 、ギャップ定数 G 、リミット定数 L とオペランドCで指定するレジスタに続く7個のレジスタに格納されている前回入力偏差の2乗 $e^{-1} \cdot |e^{-1}|$ 、前回入力値の2乗 $P_{-1} \cdot |P_{-1}|$ 、前々回入力値の2乗 $P_{-2} \cdot |P_{-2}|$ 、剰余データ r （初期値=0）に基づいて、オペランドAで指定するレジスタを先頭とする2個のレジスタに格納されている入力値 P 、設定値 S について、偏差2乗速度型PID演算を以下のように行い、その結果（出力データ ΔM 、前回入力偏差の2乗 $e^{-1} \cdot |e^{-1}|$ 、前回入力値の2乗 $P_{-1} \cdot |P_{-1}|$ 、前々回入力値の2乗 $P_{-2} \cdot |P_{-2}|$ 、剰余データ r ）をオペランドCで指定するレジスタを先頭とする5個のレジスタ（3個の倍長レジスタを含む）に格納します。

入力	処 理	出力
OFF	不実行	OFF
ON	実行 : KIHとKDL≠0	ON
	不実行 : KIHまたはKDL=0	OFF

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A	入力データ先頭											○	○	○	○	○	○	○	○	○	○	○							
B	ハ・ラメータ先頭											○	○	○	○	○	○	○	○	○	○	○							
C	出力データ先頭												○	○	○	○	○	○	○	○	○	○		○					

|R0040 00010 00001 10914 Y035A
| | [RW100 PID2 BW200] + RW400] ----- ()

- ・ a 接点 R0040 が ON すると、オペランド B で指定するレジスタを先頭とする 7 個のレジスタ RW200 の内容 ($K_P=1$)、RW201 の内容 ($K_{IH}=7$)、RW202 の内容 ($K_{IL}=10$)、RW203 の内容 ($K_{DH}=20$)、RW204 の内容 ($K_{DL}=5$)、RW205 の内容 ($G=0$)、RW206 の内容 ($L=100$) とオペランド C で指定するレジスタ (RW400) に続く 4 個のレジスタ (倍長レジスタ RW402・RW401・RW404・RW403・RW406・RW405 および単調レジスタ RW407) に格納されている前回入力偏差 $e_{-1} \cdot |e_{-1}| (8649)$ 、前回入力値 $P_{-1} \cdot |P_{-1}| (49)$ 、前々回入力値 $P_{-2} \cdot |P_{-2}| (25)$ 、剰余データー $r (0)$ を用いて、オペランド A で指定するレジスタ RW100 内容 ($P=25$) と RW101 の内容 ($S=100$) の入力データについての偏差 2 乗速度型 PID 演算を実行し、演算結果 ($\Delta M=10914$ 、 $e_{-1} \cdot |e_{-1}|=8100$ 、 $P_{-1} \cdot |P_{-1}|=100$ 、 $P_{-2} \cdot |P_{-2}|=49$ 、 $r=3$) をオペランド C で指定するレジスタを先頭とする 5 個のレジスタ (RW300~RW304) にそれぞれ格納します。演算後に実行出力を ON します。
- ・ a 接点 R0040 が OFF の時は、本命令を実行せず、実行出力を OFF にしますが、 ΔM と r に 0 を、 $e_{-1} \cdot |e_{-1}|$ に $e \cdot |e| (= (S-P) \cdot |S-P|)$ の値を、 $P_{-1} \cdot |P_{-1}|$ と $P_{-2} \cdot |P_{-2}|$ に $P \cdot |P|$ の値を設定します。

B	比例係数	Kp
B+1	積分係数	K _{IH}
B+2	積分係数	K _{IL}
B+3	微分係数	K _{DH}
B+4	微分係数	K _{DL}
B+5	ギャップ定数	G
B+6	リミット定数	

C	出力データ	ΔM		
C+2・C+1	前回入力偏差の2乗	$e_{-1} \cdot e_{-1}$		32ビット長
C+4・C+3	前回入力値の2乗	$P_{-1} \cdot P_{-1}$		32ビット長
C+6・C+5	前々回入力値の2乗	$P_{-2} \cdot P_{-2}$		32ビット長
C+7	剰余データ	r		

[偏差 2 乗速度型 P I D 演算]

$$\text{出力データ } \Delta M = K_p \cdot \{ (e \cdot |e| - e_{-1} \cdot |e_{-1}|) + \text{INT} \left(\frac{|K_{IL}| \cdot e \cdot |e| + |r|}{|K_{IH}|} \right) + \text{INT} \left[\frac{|K_{DH}|}{|K_{DL}|} \cdot (2P_{-1} \cdot |P_{-1}| - P \cdot |P| - P_{-2} \cdot |P_{-2}|) \right] \}$$

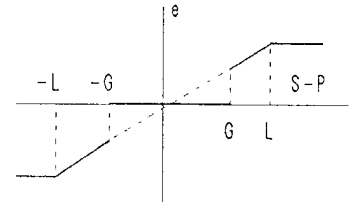
ここで、 e は入力偏差であり、 $(S-P)$ の値に対するリミット・ギャップ演算から求めます（図参照）。また r は

$$\text{INT} \left(\frac{|K_{IL}| \cdot e \cdot |e| + |r|}{|K_{IH}|} \right) \quad (|r| \text{の初期値は} 0)$$

の剰余を表わしています。

また、 $\text{INT}(a)$ は除算 a から商を取り出す関数です。

例. $\text{INT} \left(\frac{50}{3} \right) = 16$ 、 $\text{INT} \left(\frac{18}{5} \right) = 3$



- レジスタに設定可能なデータの範囲は、単調データは-32768~32767、倍長データは-2147483648~2147483647です。
- 算出した出力データ $\Delta M > 32767$ 、または $\Delta M < -32768$ の時は、オペランドCで指定するレジスタにはリミット定数 L を格納し、実行出力をONします。
- $K_{IH}=0$ 、または $K_{DL}=0$ の場合は、演算を実行せずに実行出力をOFFにし、しかもエラーフラグ（除算エラー）をセットします。

正弦関数演算を実行します。

[A SIN B] _____

- ・オペランドAで指定するデータ（入力データ：単位はdeg）の1/100の値の正弦値を算出し、算出値を10000倍してオペランドBで指定する場所に格納します。格納後、実行出力をONにします。
- ・指定可能な入力データの範囲は-32768～32767です。

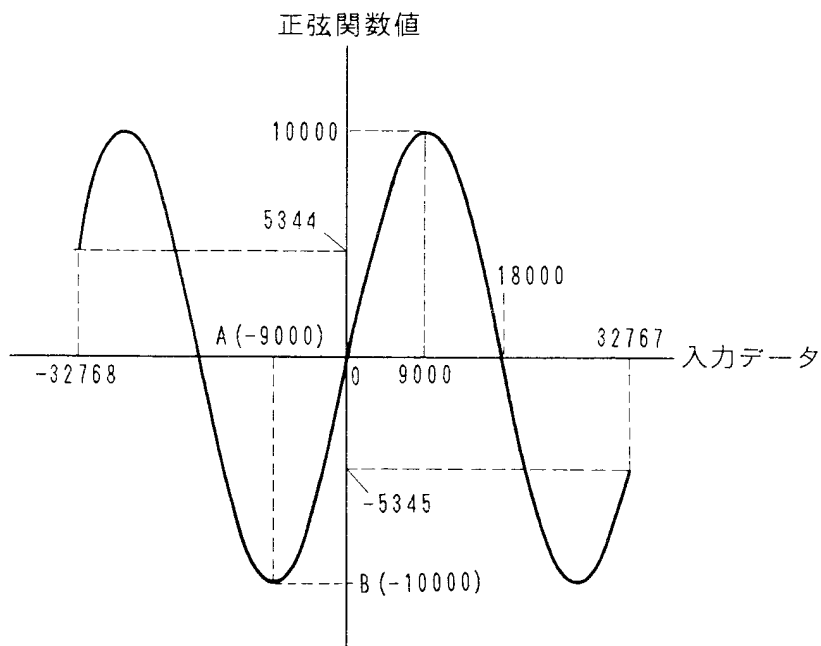
入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

opr	名 称
A	入力データ
B	関数値格納先

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A	入力データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
B	関数値格納先		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○	○		○

[RW022 SIN D0214]

- a接点R0030がOFFの時は、演算を実行せず、出力をOFFにします。



正接関数

FUN1 7 4 正接関数 (TAN) *

正接関数演算を実行します。

関連命令

条件入力

実行出力

[A TAN B]

機能

- オペランドAで指定するデータ（入力データ：単位はdeg）の1/100の値の正接値を算出し、算出値を10000倍してオペランドBで指定する場所に格納します。格納後、実行出力をONします。
- 指定可能な入力データの範囲は-32768～32767です。
- 入力データが下記の範囲にある場合は、リミット値32767をオペランドBで指定するレジスタに格納します。
-28697 ≤ 入力データ ≤ -27000、 -10697 ≤ 入力データ ≤ -9000、
7303 ≤ 入力データ ≤ 9000、 25303 ≤ 入力データ ≤ 27000

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K
A	入力データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○
B	関数値格納先		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○	○	○	○

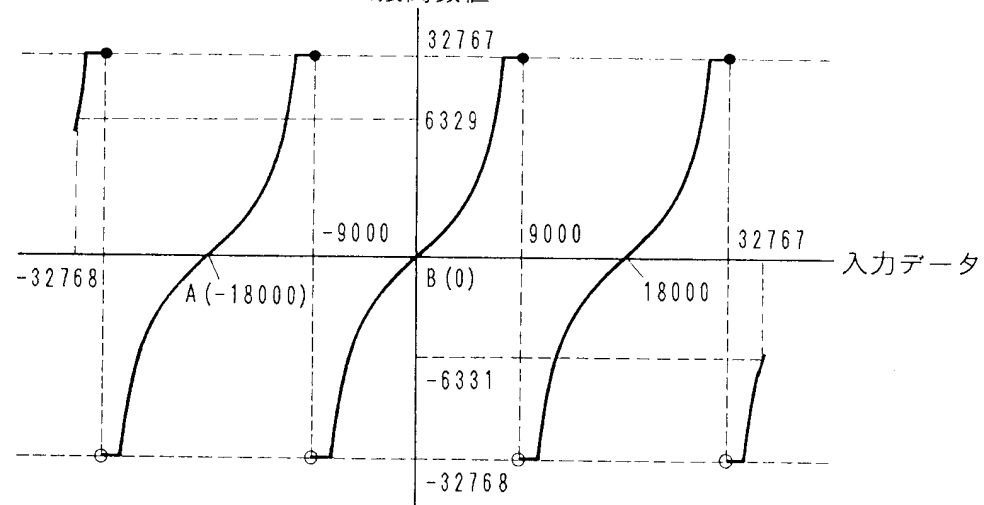
プログラム例

R0012 -18000 00000
[RW031 TAN D1025]

動作

- a接点R0012がONすると、レジスタRW031の内容（-18000）の1/100の値の正接値を計算し、計算結果を10000倍した値（0）をレジスタD1025に格納し、実行出力をONします。
- a接点R0012がOFFの時は、演算を実行せず、出力をOFFにします。

正接関数値



逆余弦関数

FUN 1 7 6 逆余弦関数 (ACOS) *

逆余弦関数演算を実行します。

関連命令

条件入力

実行出力

[A ACOS B]

機能

- オペランドAで指定するデータの1/10000の値の逆余弦値を算出し、算出値(単位はdeg)を100倍してオペランドBで指定する場所に格納します。格納後、実行出力をONします。
- 指定可能な入力データの範囲は-32768~32767です。
- 入力データ>10000の場合は、オペランドBで指定する場所には0を格納します。入力データ<-10000の場合は、オペランドBで指定する場所には18000を格納します。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

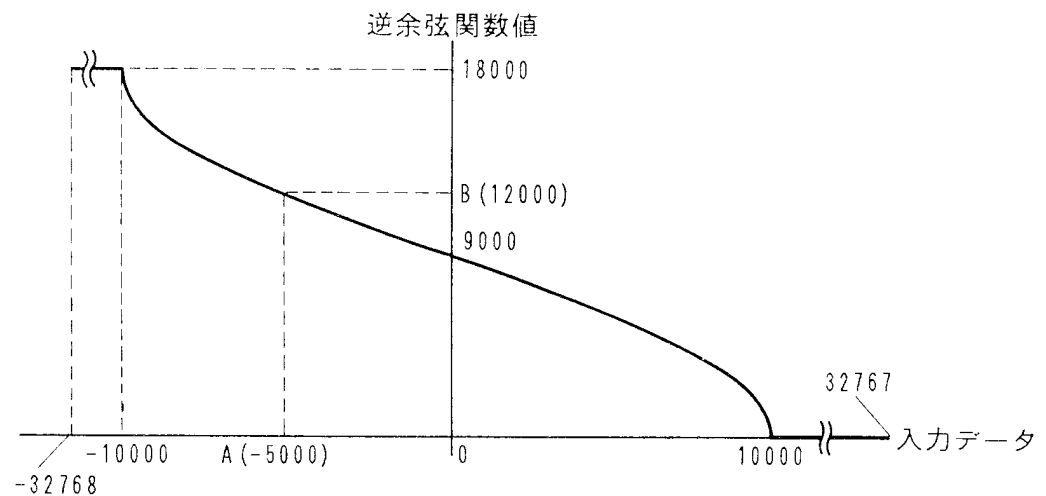
		デ バ イ ス										レ ジ ス タ												INDEX			定 数	修 飾	
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	入力データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	
B	関数値格納先		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○	○	○	

プログラム例

```
R0012 -05000      12000
├──┤ [RW012 ACOSD0118]
```

動作

- a接点R0012がONすると、レジスタRW012の内容(-05000)の1/10000の値の逆余弦($\cos^{-1}$)値を計算し計算結果を100倍した値(12000)をレジスタD0118に格納し、実行出力をONします。
- a接点R0012がOFFの時は、演算を実行せず、出力をOFFにします。



指数関数

FUN1 7 8 指数関数 (EXP) *

指数関数演算を実行します。

関連命令

条件入力

オーバーフロー出力

[A EXP B+1・B]

機能

- オペランドAで指定するデータ（入力データ）の絶対値の $1/1000$ の値の指数を算出し、算出値（倍長データ）をオペランドB+1・Bで指定する格納先に格納します。
- 指定可能な入力データの範囲は-32768～32767です。
- |入力データ| ≥ 21488 の時は、演算結果がオーバーフローするので代わりにリミット値2147483647を格納し、オーバーフロー出力をONにします。

入力	処 理	出力
OFF	不実行	OFF
ON	実行：正常	OFF
	実行：オーバーフロー発生	ON

オペランド

		デバイス										レジスタ										INDEX			定数	修飾			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A	入力データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
B+1・B	指数結果データ			○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○		○	○	○		○	

プログラム例

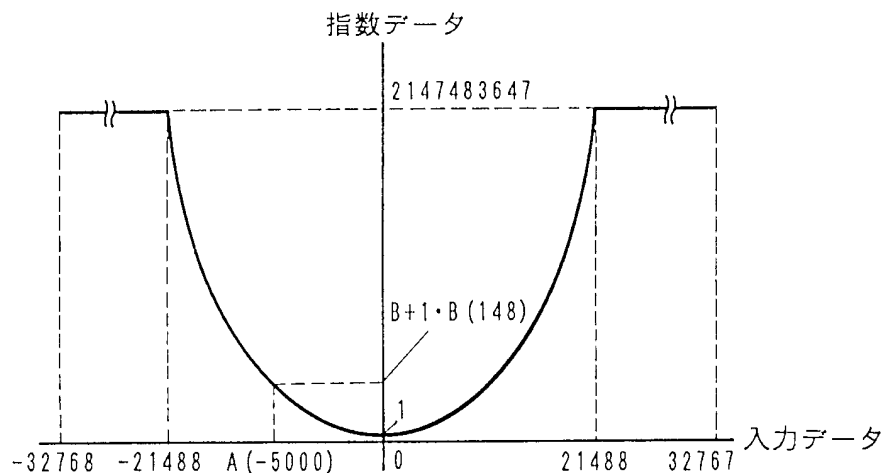
```

X0002 -05000      0000000148
┌──┴──┐ D0158 EXP RW020・RW019 ───────────────────┐ Y035A

```

動作

- a 接点X0002がONすると、レジスタD0158の内容（-5000）の絶対値の $1/1000$ の値の指数を演算し、演算結果を（148）を倍長レジスタRW019およびRW020に格納し、実行出力をONします。
- a 接点X0002がOFFの時は、演算を実行せず、出力をOFFにします。



常用対数関数

FUN 1 7 9 常用対数関数 (LOG) *

常用対数関数演算を実行します。

関連命令

条件入力

実行出力

[A LOG B]

機能

- オペランドAで指定するデータ（入力データ）の絶対値の常用対数を算出し、算出値を1000倍した後、オペランドBで指定する場所に格納し、実行出力をONします。
- 指定可能な入力データの範囲は-32768~32767です。
- 入力データ=0の時は、演算結果は0となります。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

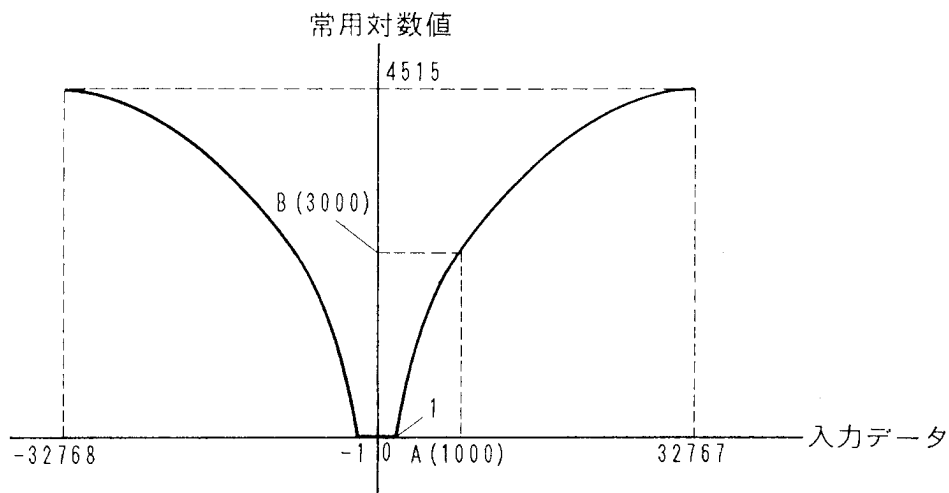
		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K		
A	入力データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
B	対数値格納先		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○	○	○

プログラム例

```
X0001 01000 03000
├──┤ D1234 LOG RW012 ───┤
```

動作

- a接点X0001がONすると、レジスタD1234の内容(01000)の絶対値の常用対数を演算し、演算結果(3000)をRW012に格納し、実行出力をONします。
- a接点X0001がOFFの時は、演算を実行せず、出力をOFFにします。



絶対値

FUN1 8 0 絶対値 (ABS) *	
絶対値を求めます。	関連命令

条件入力	実行出力
[A ABS B]	

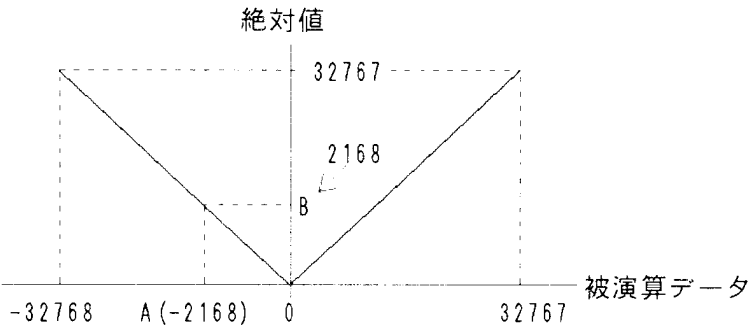
機能	
・オペランドAで指定するデータ（被演算データ：符号付き16ビット）の絶対値を求め、オペランドBで指定する場所に格納し、実行出力をONにします。	
・被演算データの範囲は-32768~32767です。	
・被演算データの値が-32768の時は、32767がオペランドBで指定する場所に格納されます。	

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド		デバイス										レジスタ										INDEX			定数	修飾			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	入力データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
B	絶対値格納先		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○		○	○	○		○

プログラム例	X0006 -02168 02168 [RW038 ABS D0121]

動作	・a接点X0006がONすると、レジスタRW038の内容（-02168）の絶対値（2168）をD0121に格納し、実行出力をONします。
	・a接点X0006がOFFの時は、演算を実行せず、出力をOFFにします。



倍長データの絶対値を求めます。

關連命令

条件入力

実行出力

$$\neg [A+1 \cdot A \quad \text{DABS} \quad B+1 \cdot B]$$

データ (倍長: 符号付指定)
 演算 B + 1・B で
 演算し、オペラ
 指定し、ON します
 出力は 2147483648 ~ 2147483647 です。
 絶対範囲は 2147483648 の時は
 絶対範囲の値が指定する場所に格納されます。
 格納の範囲は 2147483647 の場合は
 絶対範囲の値が指定する場所に格納されます。
 場所データ B + 1・B
 ライ場所データ B + 1・B
 プレ演算 B + 1・B
 オキ 32 被演算 B + 1・B
 被演算 B + 1・B
 被演算 B + 1・B

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

		デ バ イ ス										レ ジ ス タ										INDEX				定 数	修 飾	
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K		
A+1・A	入力データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
B+1・B	絶対値 格納先			○	○	○	○	○			○		○	○	○	○	○	○	○	○	○			○	○	○		

X0007 -1234567890 1234567890
 RW043·RW042 DABS RW022·RW021

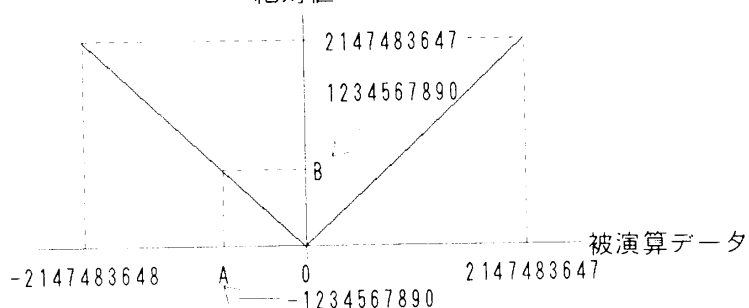
動作

- ・ a 接点 X 0 0 0 7 が ON すると、レジスタ RW 0 4 3 ・ RW 0 4 2 の内容 (- 1 2 3 4 5 6 7 8 9 0) の絶対値 (1 2 3 4 5 6 7 8 9 0) を、RW 0 2 2 ・ RW 0 2 1 に格納し実行出力を ON します。
- ・ a 接点 X 0 0 0 7 が OFF の時は、演算を実行せず、実行出力を OFF にします。

→
絶対値

演算結果	
上位	下位
RW022	RW021
{1234567890}	

絶対値



条件入力

実行出力

[A NEG B]

- ・ オペランドAで指定するデータ（被演算データ：16ビット）の2の補数を求め、オペランドBで指定する場所に格納し、実行
- ・ 出力をONします。
- ・ 被演算データの範囲は-32768～32767です。
- ・ 演算結果の有効範囲は-32768～32767です。被演算データが-32768の時は、演算結果も-32768となります。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

[illegible]

R0012 04660 -04660
| | [RW042 NEG D0326]

- ・ a 接点 R 0 0 1 2 が O N すると、レジスタ R W 0 4 2 の内容 (0 4 6 6 0) の 2 の補数 (- 0 4 6 6 0) を演算し、D 0 3 2 6 に格納後、実行出力を O N します。
- ・ a 接点 R 0 0 1 2 が O F F の時は、演算を実行せず、実行出力を O F F にします。

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	
RW042	0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	[4660]

↓ 全ビットを反転

1	1	1	0	1	1	0	1	1	1	0	0	1	0	1	1

↓ 1 を加算

00326 1 1 1 0 1 1 0 1 1 1 0 0 1 1 0 0 [-4660]

FUN 1 8 3 倍長2の補数 (DNEG) *

倍長データの2の補数を求めます。

関連命令

条件入力 実行出力

$$- [A+1 \cdot A \quad \text{DNEG} \quad B+1 \cdot B]$$

機能

- ・オペランドA + 1・Aで指定する倍長データ（被演算データ：32ビット）の2の補数を求め、オペランドB + 1・Bで指定する場所に格納し、実行出力をONします。
- ・被演算データの範囲は-2147483648～2147483647です。
- ・演算結果の有効範囲は-2147483648～2147483647です。被演算データが-2147483648の時は、演算結果も-2147483648となります。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A+1・A	入力データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○		○	○
B+1・B	補数格納先		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○			

プログラム例

R0013	-1234567890	1234567890

| | [RW043·RW042 DNEG D0324·D0323]

動作

- 動作
- ・ a 接点 R 0 0 1 3 が O N すると、レジスタ R W 0 4 3 ・ R W 0 4 2 の内容 (- 1 2 3 4 5 6 7 8 9 0) の 2 の補数 (1 2 3 4 5 6 7 8 9 0) を演算し、D 0 3 2 4 ・ D 0 3 2 3 に格納後、実行出力を O N します。
 - ・ a 接点 R 0 0 1 3 が O F F の時は、演算を実行せず、実行出力を O F F にします。

	上位	下位	
被演算データ	RW403	RW402	[88669FD2EH]

↓ 全ビットを反転

↓ 1 を加算

	上位	下位	
補数格納先	D0324	D0323	[499602D2H]

FUN 1 8 4 ダブルワード変換 (DW) *

1 6ビットデータ形式を32ビットデータ形式に変換します。

関連命令

実行出力

$$- [A \quad DW \quad B+1 \cdot B]$$

機能

- ・オペランドAで指定する16ビットデータ（被演算データ）を、オペランドB + 1・Bで指定する場所の下位16ビットの桁に格納し、その上位16ビットの桁に、被演算データの最上位ビットが1（データが負）の時はFFFFHを、最上位ビットが0（データが正）の時は0000Hを格納し、実行出力をONします。
- ・被演算データの範囲は-32768～32767です。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

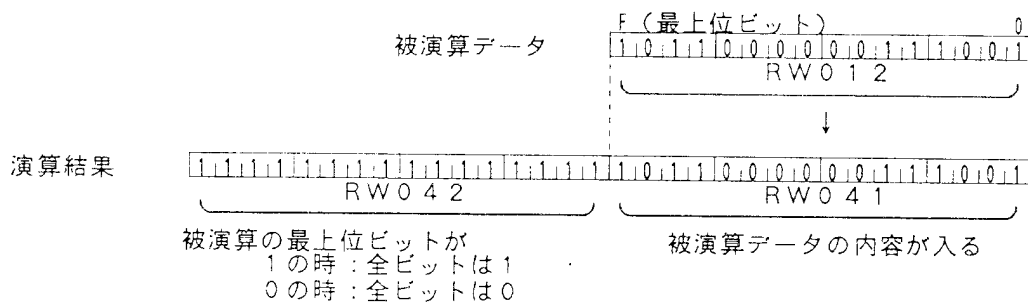
		デバイス										レジスタ										INDEX			定数	修飾		
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	被変換データ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○
B+1・B	データ格納先		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○		○	○	

プログラム例



動作

- ・ a 接点 R 0 0 3 0 が O N すると、レジスタ R W 0 1 2 の内容 (- 1 2 3 4 5) を R W 0 4 1 に、 F F F F H (被演算データの最上位ビットが 1 なので) を R W 0 4 2 にそれぞれ格納後、実行出力を O N します。
- ・ a 接点 R 0 0 3 0 が O F F の時は、演算を実行せず、実行出力を O F F にします。



[7セグメント表示用データ]

表 示 デ ー タ		7セグメント の構成		7セグメントデータ							
16進数	下位4ビット			B7	B6	B5	B4	B3	B2	B1	B0
0	0 0 0 0	B0 B5 B1 B6 B4 B2 B3		0	0	1	1	1	1	1	1
1	0 0 0 1			0	0	0	0	0	1	1	0
2	0 0 1 0			0	1	0	1	1	0	1	1
3	0 0 1 1			0	1	0	0	1	1	1	1
4	0 1 0 0			0	1	1	0	0	1	1	0
5	0 1 0 1			0	1	1	0	1	1	0	1
6	0 1 1 0			0	1	1	1	1	1	0	1
7	0 1 1 1			0	0	1	0	0	1	1	1
8	1 0 0 0			0	1	1	1	1	1	1	1
9	1 0 0 1			0	1	1	0	1	1	1	1
A	1 0 1 0			0	1	1	1	0	1	1	1
B	1 0 1 1			0	1	1	1	1	1	0	0
C	1 1 0 0			0	0	1	1	1	0	0	1
D	1 1 0 1			0	1	0	1	1	1	1	0
E	1 1 1 0			0	1	1	1	1	0	0	1
F	1 1 1 1			0	1	1	1	0	0	0	1

ASCII変換

FUN186 ASCII変換(ASC)*

半角カナ英数字をアスキーコードに変換します。

関連命令

条件入力

実行出力

[A ASC B]

機能

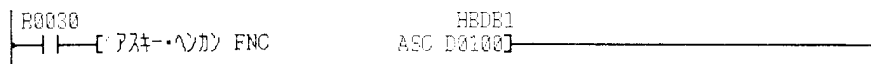
- オペランドAで指定する半角のカナおよび英数字（スペースを含む最大16文字）をASCIIコードに変換し、オペランドBで指定するレジスタを先頭とする8個のレジスタに格納し、実行出力をONします。
- 変換されたASCIIコードは、オペランドBで指定する先頭レジスタの下位バイト、上位バイト、それに続くレジスタの下位バイト、上位バイトの順に格納されます。残りのレジスタには命令実行前の内容がそのまま残ります。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A	アスキー文字																											○	
B	コード格納先												○	○	○	○	○	○	○	○	○	○		○					○

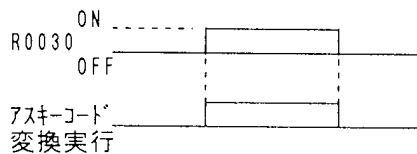
プログラム例



動作

- a接点R0030がONすると、半角文字（アスキー・ヘンカン FNC）をASCIIコードに変換し、レジスタD0100の下位バイト、上位バイト、D0101の下位バイト、上位バイトの順に、D0106の下位バイトまで格納した後、実行出力をONします。
- レジスタD0106の上位バイト、D0107の下位バイトと上位バイトには命令実行前の内容がそのまま残ります。
- a接点R0030がOFFの時は、本命令を実行せず、実行出力をOFFにします。

命令実行のタイミング



変換後のコードの格納

D0100	L	ア (81H)	D0104	L	ン (D0H)
	H	ズ (8DH)		H	(20H)
D0101	L	キ (87H)	D0105	L	エ (46H)
	H	ー (B0H)		H	ン (4EH)
D0102	L	・ (A5H)	D0106	L	シ (43H)
	H	ハ (CDH)		H	
D0103	L	ン (D0H)	D0107	L	-----
	H	カ (B6H)		H	

実行前の内容が残っています

バイナリ変換

FUN1 8 8 バイナリ変換 (BIN) *	
4桁のBCD (2進化10進) データを16ビットのバイナリデータに変換します。	関連命令

条件入力	実行出力
[A BIN B]	

機能

- オペランドAで指定する4桁のBCDデータを16ビットのバイナリデータに変換し、オペランドBで指定する場所に格納し、実行出力をONします。
- 指定可能な4桁のBCDデータの範囲は0000H~9999Hです。0~9以外の値 (例えば、13A6やABC5) を指定した場合は、変換を行わずに実行出力をOFFに設定し、エラーフラグ (BCDデータエラー) をセットします。

入力	処 理	出力
OFF	不実行	OFF
ON	実行: BCDデータが正常	ON
	不実行: BCDデータが範囲外	OFF

オペランド

		デ バ イ ス										レ ジ ス タ												INDEX			定 数	修 飾
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K		
A	BCDデータ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
B	バイナリ格納先		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○	○	○

プログラム例

R0017 H4907 04907	Y035C
[RW028 BIN D0127]	()

動作

- a接点R0017がONすると、レジスタRW028に格納されているBCDデータ4907Hをバイナリコード (132BH) に変換し、レジスタD0127に格納した後、実行出力をONします。
- a接点R0017がOFFの時は、演算を実行せず、実行出力をOFFにします。

FE DC BA 9 8 7 6 5 4 3 2 1 0
 RW028 0 1 1 0 0 1 1 0 0 1 0 0 0 1 1 1 [4907H] BCD形式

↓ バイナリ変換

D0127 0 0 0 1 0 1 1 1 0 1 1 0 1 0 1 1 [4907] バイナリ形式

FUN 1 8 9 倍長バイナリ変換 (DBIN) *

8 桁の B C D (2 進化 1 0 進) データを 3 2 ビットのバイナリデータに変換します。	関連命令

条件入力	実行出力
$-[A+1 \cdot A \quad \text{DBIN} \quad B+1 \cdot B]$	

$$-[A+1 \cdot A \quad \text{DBIN} \quad B+1 \cdot B] \quad \text{-----}$$

機能

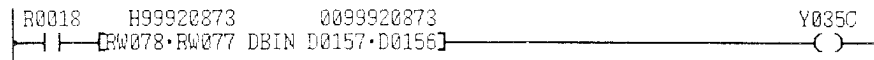
- ・オートのバイナリデータに格納し、実行出力をONにします。
- ・指定可能な8桁のBCDデータを00000000H～99999999H（0～9以外の値例えば、123ABC78やDEF12345）を指定した場合、交換を行わずに実行出力をOFFに設定し、エラーフラグ（BCDデータエラー）をセットします。

入力	処 理	出力
OFF	不実行	OFF
ON	実行 : BCDデータが正常	ON
	不実行 : BCDデータが範囲外	OFF

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾				
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K		
A+1・A	BCDデータ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○		○	○	
B+1・B	演算結果		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○		○	○			○

プログラム例



動作

- ・ a 接点 R 0 0 1 8 が ON すると、連続した 2 個のレジスタ R W 0 7 8 ・ R W 0 7 7 に格納されている B C D データ 9 9 9 2 0 8 7 3 H をバイナリコード (0 5 F 4 A B E 9 H) に変換し、連続した 2 個のレジスタ D 0 1 5 7 ・ D 0 1 5 6 に格納した後、実行出力を ON します。
- ・ a 接点 R 0 0 1 8 が O F F の時は、演算を実行せず、実行出力を O F F にします。

1F RW078 RW077 0

9	9	9	2	:	0	8	7	3

[99920873H] 倍長BCD形式

↓ バイナリ変換

 D0157 D0156

0	5	F	4	:	A	B	E	9

[99920873] 倍長バイナリ形式

条件入力

実行出力

—[A BCD B]—————

機能

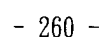
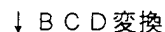
- オペランドAで指定する16ビットのバイナリデータを4桁のBCDデータに変換し、オペランドBで指定する場所に格納後、実行出力をONします。
- 指定可能なバイナリデータの範囲は0000～9999 (0000H～270FH) です。270FHを越える値を指定した場合は、変換を行わずに実行出力をOFFに設定し、エラーフラグ(BCDデータエラー)をセットします。

オペランド

プログラム例



- ・ a 接点 R 0 0 1 9 が ON すると、レジスタ D 0 2 1 1 に格納されているバイナリデータ 2 6 9 4 H を B C D コード (9 8 7 6 H) に変換し、レジスタ R W 0 2 2 に格納した後、実行出力を ON します。
- ・ a 接点 R 0 0 1 9 が O F F の時は、演算を実行せず、実行出力を O F F にします。



倍長 B C D 変換

FUN1 9 1 倍長 B C D 変換 (D B C D) *

32ビットのバイナリデータを8桁のBCD(2進化10進)データに変換します。

関連命令

条件入力

実行出力

[A+1・A DBCD B+1・B]

機能

- オペランドA+1・Aで指定する32ビットのバイナリデータを8桁のBCDデータに変換し、オペランドB+1・Bで指定する場所に格納後、実行出力をONします。
- 指定可能なバイナリデータの範囲は00000000~99999999(00000000H~05F5E0FFH)です。05F5E0FFHを超える値を指定した場合は、変換を行わずに実行出力をOFFに設定し、エラーフラグ(BCDデータエラー)をセットします。

入力	処 理	出力
OFF	不実行	OFF
ON	実行：被変換データが正常	ON
	不実行：被変換データが範囲外	OFF

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾				
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K		
A+1・A	バ イナリデータ	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○		○	○	
B+1・B	演算結果		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○			○	○

プログラム例

[R011A 0098765432 H98765432 Y035C
[D0321・D0320 DBCD RW040・RW039]]

動作

- a接点R011AがONすると、連続した2個のレジスタD0321・D0320に格納されているバイナリデータ05E30A78HをBCDコード(98765432H)に変換し、連続した2個のレジスタRW040・RW039に格納した後、実行出力をONします。
- a接点R011AがOFFの時は、演算を実行せず、実行出力をOFFにします。

1F D0321 D0320 0
0 5 E 3 0 A 7 8 [098765432] 倍長バイナリ形式

↓ B C D 変換

RW040 RW039
9 8 7 6 5 4 3 2 [98765432H] 倍長BCD形式

条件入力

オーバーフロー出力

-[A B+ B → C]-----

オランダAで指する4桁BCDデータ(被加算数)とオペランドBで指定する4桁BCDデータ(加算数)を加算し、オペランドCで指定する場所に加算結果を格納します。各桁指定でA~Fを指定するとは加算できる。いずれかの桁に0~9以外の値を使用すると、演算を実行せず、実行出力をOFFに設定し、エラーフラグ(BCDエラー)をセットします。加算した結果がオーバーフローする場合(9999Hを超える場合)、リミット値9999HをオペランドCで指定するレジスタに格納し、オーバーフロー出力をONします。

オペランド

プログラム例

動作

- ・ a 接点 R 0 0 1 B が O N すると、レジスタ R W 0 1 8 に格納されている B C D データ 3 7 6 5 H とレジスタ R W 0 2 2 に格納されている B C D データ 4 8 4 9 H を加算し、演算結果 (B C D データ 8 6 1 4 H) をレジスタ D 0 2 2 5 に格納した後、オーバーフロー出力を O F F します。
- ・ a 接点 R 0 0 1 B が O F F の時は、演算を実行せず、オーバーフロー出力を O F F にします。



FUN 1 9 3 B C D減算 (B -) *

4桁のBCDデータ間でキャリーなし10進減算を実行します。

関連命令

条件入力

アンダーフロー出力

$$- [A \quad B - B \rightarrow C] \text{-----}$$

機盤

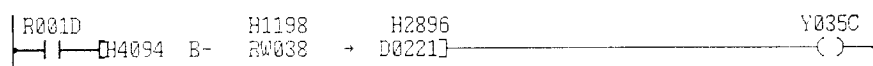
- ・オペランドAで指定する4桁BCDデータ(被減数)からオペランドBで指定する4桁BCDデータ(減数)を引き、オペランドCで指定する場所に減算の結果を格納します。
- ・減算の結果がアンダーフローになる場合(即ち、0000Hより小さくなる場合)、リミット値0000HをオペランドCで指定する場所に格納し、アンダーフロー出力をONします。
- ・指定可能な被減数と減数の範囲は0000H~9999Hです。各桁でA~Fを指定することはできません。いずれかの桁に0~9以外の値を使用すると、演算を実行せず、アンダーフロー出力をOFに設定し、エラーフラグ(BCDエラー)をセットします。

入力	処 理		出力
OFF	不実行		OFF
ON	実行	正常	OFF
		演算結果がアンダーフロー	ON
	不実行	被減数または減数が指定範囲外	OFF

オペランド

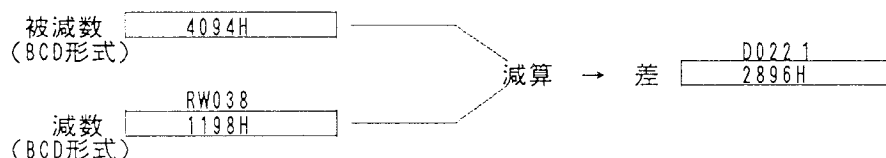
		デ バ イ ス										レ ジ ス タ										INDEX			定 修				
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
A	被減数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
B	減 数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
C	差		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○	○		○

プログラム例



動作

- ・a接点R001DがONすると、オペランドAで指定するBCDデータ4094HからレジスタRW038に格納されているBCDデータ1198Hを引き、演算結果（BCDデータ2896H）をレジスタD0221に格納した後、アンダーフロー出力をOFFに設定します。
- ・a接点R001DがOFFの時は、演算を実行せず、アンダーフロー出力をOFFにします。



FUN 1 9 4 B C D 乘算 (B*) *

関連命令

実行出力

$$- [A \quad B^* \quad B \rightarrow C]$$

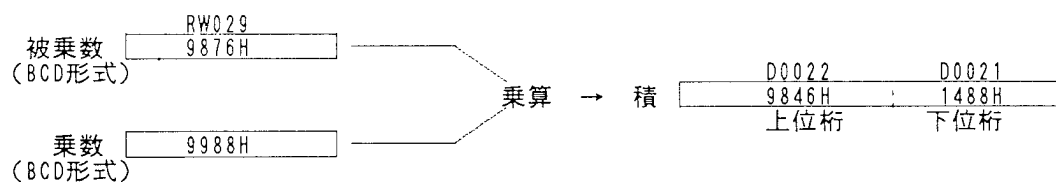
- ・オペランドAで指定する4桁BCDデータ(被乗数)とオペランドBで指定する4桁BCDデータ(乗数)の積を計算し、オペランドCで指定する場所に乗算の結果を格納した後、実行出力をONします。
- ・指定可能な被乗数と乗数の範囲は0000H~9999Hです。各桁でA~Fを指定することはできません。いずれかの桁に0~9以外の値を使用すると、演算を実行せず、実行出力をOFFに設定し、エラーフラグ(BCDエラー)をセットします。

入力	処 理		出力
OFF	不実行		OFF
ON	実行	正常	ON
	不実行	被乗数または乗数が範囲外	OFF

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K		
A	被乗数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
B	乗 数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
C	積		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○	○	○			○

R001F H9876 H98641488 Y035C
| | | RW029 B* H9988 → D0022·D0021] ————— ()

- ・ a 接点 R001F が ON すると、レジスタ RW029 に格納されている BCD データ 9876H とレジスタ RW038 に格納されている BCD データ 9988H の積を求め、演算結果 (BCD データ 98641488H) を 2 個の連続したレジスタ D0022・D0021 に格納した後、実行出力を ON します。
- ・ a 接点 R001F が OFF の時は、演算を実行せず、実行出力を OFF にします。



倍長 B C D 加算

FUN196 倍長BCD加算(DB+)※

8桁のBCDデータ間でキャリーなし10進加算を実行します。

関連命令

条件入力

オーバーフロー出力

—[A+1・A DB+ B+1・B → C+1・C]——

機能

- オペランドA+1・Aで指定する8桁BCDデータ(被加算数)とオペランドB+1・Bで指定する8桁BCDデータ(加算数)を加算し、加算結果をオペランドC+1・Cで指定する場所に格納します。
- 指定可能な被加算数と加算数の範囲は00000000H~99999999Hです。各桁でA~Fを指定することはできません。いずれかの桁に0~9以外の値を使用すると、演算を実行せず、オーバーフロー出力をOFFに設定し、エラーフラグ(BCDエラー)をセットします。
- 加算結果がオーバーフローする場合(99999999Hを超える場合)リミット値99999999HをオペランドC+1・Cで指定する場所に格納し、オーバーフロー出力をONします。

入力	処 理		出力
OFF	不実行		OFF
ON	実行	正常	OFF
		演算結果がオーバーフロー	ON
	不実行	被加算数または加数が指定範囲外	OFF

オペランド

オペランド

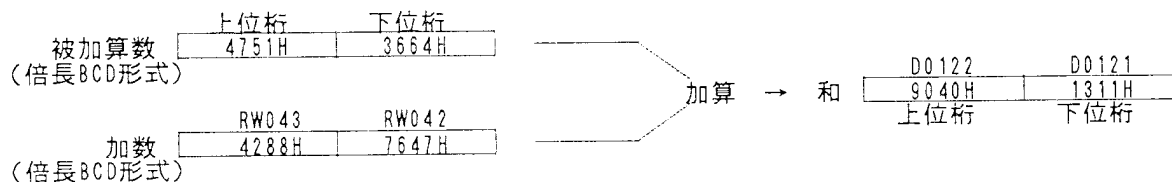
		デ バ イ ス										レ ジ ス タ												INDEX			定 数	修 飾
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K		
A+1・A	被加算数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
B+1・B	加 数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
C+1・C	和		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○		○

プログラム例

R001C H42887647 H90401311 Y035C
 —[H47513664 DB+ RW043・RW042 → D0122・D0121]——()

動作

- a接点R001CがONすると、オペランドA+1・Aで指定するBCDデータ47513664HとレジスタRW043・RW042に格納されているBCDデータ42887647Hを加算し、演算結果(BCDデータ90401311H)の上位4桁をレジスタD0122に、下位4桁をレジスタD0121に格納した後、オーバーフロー出力をOFFに設定します。
- a接点R001CがOFFの時は、演算を実行せず、オーバーフロー出力をOFFにします。



關連命令

実行出力

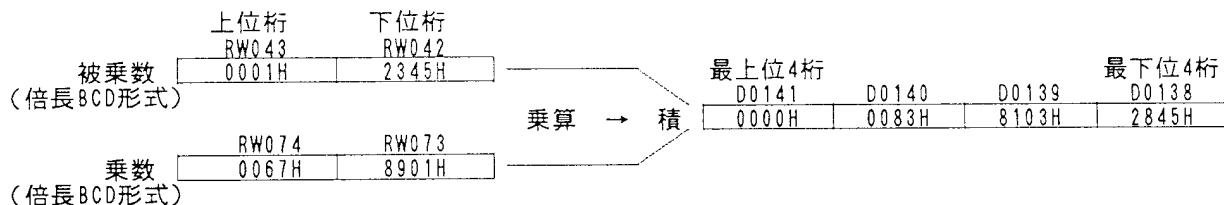
$$- [A+1 \cdot A \quad DB* \quad B+1 \cdot B \quad \rightarrow \quad C+1 \cdot C]$$
[illegible]

入力	処 理		出力
OFF	不実行		OFF
ON	実行	正常	ON
	不実行	被乗数または乗数が範囲外	OFF

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A+1	A 被乗数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
B+1	B 乗 数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
C+1	C 積		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○	○		○	○	○		

R0020 H00012345 H00678901 H81032845 Y035A
 | | | [RW043·RW042 DB* RW074·RW073 → D0139·D0138] ————— ()

- ・ a 接点R0020がONすると、倍長レジスタRW043・RW042に格納されているBCDデータ00012345Hと倍長レジスタRW074・RW073に格納されているBCDデータ00678901Hの積を求め演算結果（BCDデータ8381032845H）を4倍長レジスタD0141・D0140・D0139・D0138に格納した後、実行出力をONします。
- ・ a 接点R0020がOFFの時は、演算を実行せず、実行出力をOFFにします。



- ・オペランド (C+1・C) にインデックスレジスタを指定すると、積データ (C+3・C+2) は切り捨てられますのでご注意ください。

FUN1 9 9 倍長BCD除算 (DB /) *

8桁のBCDデータ間で除算を実行します。

関連命令

条件入力 実行出力

$$-[A+1 \cdot A \quad DB/ \quad B+1 \cdot B \quad \rightarrow \quad C+1 \cdot C] \text{-----}$$

機能

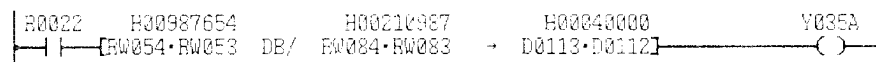
- ・オペランド A + 1 ・ A で指定する 8 桁 B C D データ (被除数)
- ・オペランド B + 1 ・ B で指定する 8 桁 B C D データ (除数)
- ・の間で除算を実行し、演算結果の商と剰余をオペランド C + 1
- ・ C とオペランド C + 3 ・ C + 2 で指定する場所にそれぞれ格納した後に、実行出力を ON します。
- ・指定した可能な被除数を除数の範囲は 00000000H ~ 99999999H です。各桁で A ~ F を指定することはできません。いずれかの桁に 0 ~ 9 以外の値を使用すると、演算を実行せず、実行出力を OFF に設定し、エラーフラグ (BCD エラー) をセットします。
- ・除数に 0 を指定した場合は、演算を行わず、実行出力を OFF に設定し、エラーフラグ (除算エラー) をセットします。

入力	処 理		出力
OFF	不実行		OFF
ON	実行	正常	ON
	不実行	除数 = 0	OFF
		被除数または除数が範囲外	OFF

オペランド

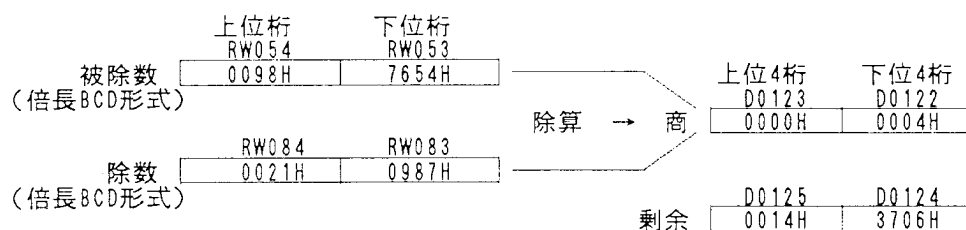
		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A+1	被除数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
B+1	除 数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
C+1	商		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○	○		○	○	○		○

プログラム例



動作

- ・a接点R0022がONすると、倍長レジスタRW054・RW053に格納されているBCDデータ00987654Hを倍長レジスタRW084・RW083に格納されているBCDデータ00210987Hで割り、演算結果の商00000004HをレジスタD0123・D0122に、剰余00143706HをレジスタD0125・D0124に格納した後、実行出力をONします。
- ・a接点R0022がOFFの時は、演算を実行せず、実行出力をOFFにします。



補足

- オペランド (C+1・C) にインデックスレジスタを指定すると、剰余データ (C+3・C+2) は切り捨てられますのでご注意ください。

4桁のBCDデータ間でキャリー付き10進加算を実行します。	関連命令

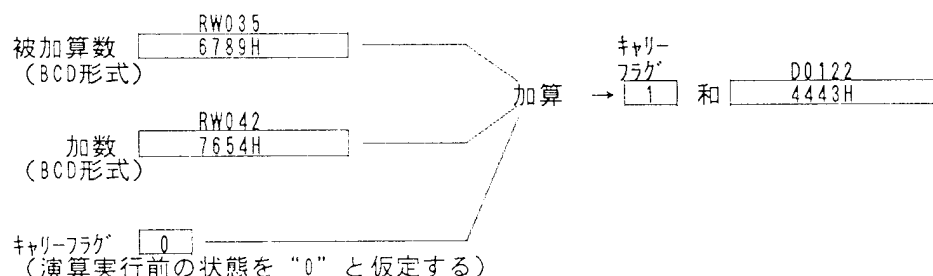
オーバーフロー出力

$$-[A \quad B+C \quad B \quad \rightarrow \quad C] \text{-----}$$

- ・データ（被加数）とオペランド（加数）を加算し、オペランドの値を格納します。
- ・指定のF値に設定された値（0～9）を出力します。
- ・F値は、加算の結果（0～9）を出力します。

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	被加算数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
B	加 数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○			○	○	○	○	○
C	和		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○	○		○	○	○	○		○

- ・ a 接点 R 0 0 2 3 が O N すると、レジスタ R W 0 3 5 に格納されている B C D データ 6 7 8 9 H とレジスタ R W 0 4 2 に格納されている B C D データ 7 6 5 4 H を加算し、演算結果 (B C D データ 4 4 4 3 H) をレジスタ D 0 1 2 2 に格納した後、キャリーフラグをセットし、オーバーフロー出力を O N します。
- ・ a 接点 R 0 0 2 3 が O F F の時は、演算を実行せず、オーバーフロー出力を O F F にします。



FUN 2 0 1 キャリー付きBCD減算 (B - C) *

4桁のBCDデータ間でキャリー付き10進減算を実行します。

関連命令

条件入力

アンダーフロー出力

$$-[A \quad B-C \quad B \rightarrow C] -$$

機能

- ・オペランドAで指定する4桁BCDデータ（被減数）からオペランドBで指定する4桁BCDデータ（減数）を引き、オペランドCで指定する場所に減算の結果を格納します。
- ・指定可能な被減数と減数の範囲は0000H～9999Hです。各桁でA～Fを指定することはできません。いずれかの桁に0～9以外の値を使用すると、演算を実行せず、アンダーフロー出力をOFFに設定し、エラーフラグ（BCDエラー）をセットします。
- ・減算の結果がアンダーフローする場合（即ち、0000Hより小さくなる場合）、キャリーフラグをセットし、アンダーフロー出力をONします。減算の結果がアンダーフローしない場合は、キャリーフラグをリセットし、アンダーフロー出力をOFFにします。

オペランド

入力	処 理		出力
OFF	不実行		OFF
ON	実行	演算結果が正常	OFF
		演算結果がアンダーフロー	ON
	不実行	被減数または減数が指定範囲外	

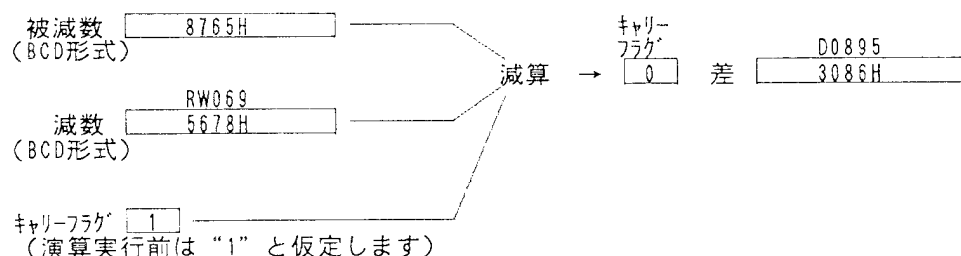
オペランド		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K		
A	被減数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
B	減 数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○	○	○	○
C	差		○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○		○	○	○	○		

プログラム例

R0025 H5678 H3086 Y035A
H8765 B-C RW069 → D0895

動作

- ・a 接点 R0025 が ON すると、オペランド A で指定する BCD データ 8765 H からレジスタ RW069 に格納されている BCD データ 5678 H を引き、演算結果 (BCD データ 3086 H) をレジスタ D0895 に格納した後、キャリーフラグをリセットし、アンダーフロー出力を OFF にします。
- ・a 接点 R0025 が OFF の時は、演算を実行せず、アンダーフロー出力を OFF にします。



キャリー付き倍長BCD加算

FUN2 0 2 キャリー付き倍長BCD加算 (DB+C)*

8桁のBCDデータ間でキャリー付10進加算を実行します。

関連命令

条件入力

オーバーフロー出力

—[A+1・A DB+C B+1・B → C+1・C]——

機能

- オペランドA+1・Aで指定する8桁BCDデータ(被加算数)とオペランドB+1・Bで指定する8桁BCDデータ(加算数)を加算し、加算結果をオペランドC+1・Cで指定する場所に格納します。
- 指定可能な被加算数と加算数の範囲は00000000H~99999999Hです。いずれかの桁で0~9以外の値(A~F)を使用すると、演算を実行せず、エラーフラグ(BCDエラー)をセットし、オーバーフロー出力をOFFに設定します。
- 演算結果がオーバーフローする場合(99999999Hを超える場合)、キャリーフラグをセットし、オーバーフロー出力をONします。加算結果がオーバーフローしない場合、キャリーフラグをクリアし、オーバーフロー出力をOFFにします。

入力	処 理		出力
OFF	不実行		OFF
ON	実行	演算結果が範囲内	OFF
		演算結果がオーバーフロー	ON
	不実行	被加算数または加数が指定範囲外	OFF

オペランド

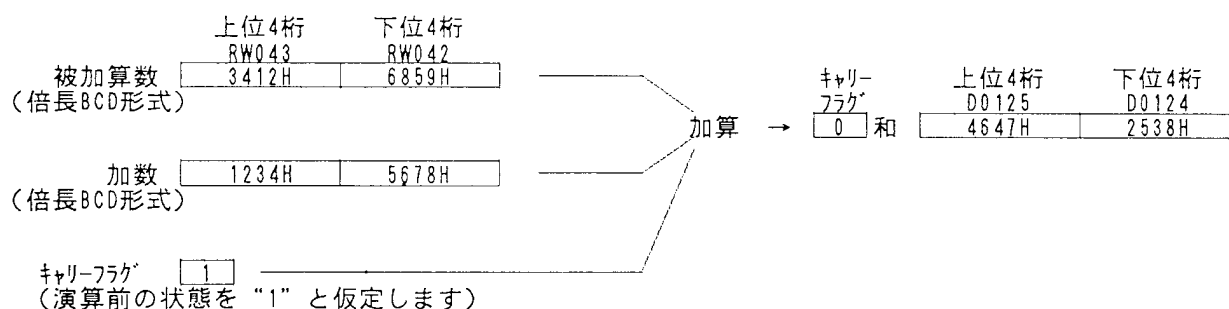
		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K		
A+1・A	被加算数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
B+1・B	加 数	○	○	○	○	○	○			○		○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
C+1・C	和		○	○	○	○	○				○		○	○	○	○	○	○	○	○	○		○	○			○	○

プログラム例

R0024 H34126859 H46472538 Y035A
└─┬─┐ └─┬─┐ └─┬─┐ └─┬─┐
RW043・RW042 DB+C H12345678 → D0125・D0124 ─┬─┐ └─┬─┐

動作

- a接点R0024がONすると、倍長レジスタRW043・RW042に格納されている倍長BCDデータ34126859HとオペランドB+1・Bで指定するBCDデータ12345678Hを加算し、演算結果(BCDデータ46472538H)の上位4桁をレジスタD0125に、下位4桁をレジスタD0124に格納した後、キャリーフラグをリセットし、オーバーフロー出力をOFFにします。
- a接点R0024がOFFの時は、演算を実行せず、オーバーフロー出力をOFFにします。



[T 2 E / T 2 N / T 3 / T 3 H 使用可能]

関連命令

実行出力

100%

タデタ
点指定
数Bの最
小・タ3
固定+シ
固B残リ
ビット長
2ラ(倍部
3べし指
す、格納
定しにしま
指換タビ
で変ス8ON
Aにジ位力
・タレ出
1一の個行
+デの次
A点2号、
ド少数符
ラン動連
オをるッ
・仮部)
オタ数)ノ
ハ、実A
範囲は-2
-1474836
-68~2147
483647です。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

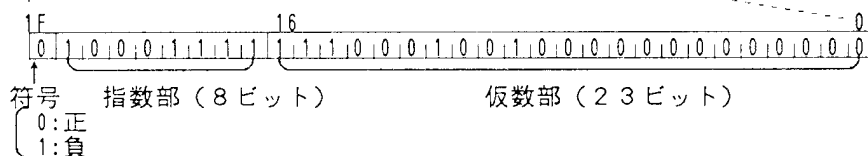
		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A+1・A	入力データ											○	○	○	○	○	○	○	○	○	○	○		○	○		○	○
B+1・B	演算結果												○	○	○	○	○	○	○	○	○	○	○	○	○	○		○

R003E 0000123456 123456.0
| | RW047·RW046 FLT D0128·D0127

- ・a 接点 R003E が ON すると、倍長レジスタ RW047・RW046 に格納されている固定小数点データ 123456 (0001E240H) を浮動小数点データ (47F12000H) に変換し、倍長レジスタ D0128・D0127 に格納 (D0128 の最上位ビットに 0 (符号)、D0128 の次の 8 ビットに指数部を、D0128 の次の 4 ビットと D0127 の全 16 ビットに仮数部を格納) した後、実行出力を ON します。
- ・a 接点 R003E が OFF の時は、演算を実行せず、実行出力を OFF にします。

上位 RW047	下位 RW046	固定小数点データ [123456]
0001H	E240H	

47F1H	2000H	浮動小数点データ「123456.0」



[T 2 E / T 2 N / T 3 / T 3 H 使用可能]

浮動小数点データを32ビット固定小数点データに変換します。

関連命令

条件入力

実行出力

—[A+1 • A FIX B+1 • B]—————

機能

- ・オペランドA+1・Aで指定する倍長レジスタに格納されている入力データ（浮動小数点データ）を32ビット固定小数点データに変換し、オペランドB+1・Bで指定する倍長レジスタに格納し、実行出力をONにします。
- ・オペランドA+1・Aで指定可能な浮動小数点データの範囲は-2147483648～2147483647です。指定した値が-2147483648より小さい場合は-2147483648（CF000000H）を、2147483647より大きい場合は2147483647（7FFFFFFFH）をリミット値として指定します。
- ・入力データの小数点以下の桁は切り捨てます。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾	
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J
A+1・A	入力データ											○	○	○	○	○	○	○	○	○	○	○			○	○	
B+1・B	演算結果												○	○	○	○	○	○	○	○	○	○		○	○	○	

プログラム例

```
|R003F      123456.0      0000123456
```



 RW092·RW091 FIX D0176·D0175

動作

- ・ a 接点 R 0 0 3 F が ON すると、倍長レジスタ R W 0 9 2・R W 0 9 1 に格納されている浮動小数点データ 1 2 3 4 5 6、0 (4 7 F 1 2 0 0 0 H) を固定小数点データ (1 2 3 4 5 6) に変換し、倍長レジスタ D 0 1 7 6・D 0 1 7 5 に格納した後、実行出力を ON します。
- ・ a 接点 R 0 0 3 F が O F F の時は、演算を実行せず、実行出力を O F F にします。

上位	下位	
RWD92	RWD91	
47F1H	2000H	浮動小数点データ [123456.0]

[符号部 : 0
指数部 : 143
仮数部 : 0.883789062

↓32ビット固定小数点データに変換

0001H	E240H	固定小数点データ「123456」

浮動小数点絶対値

FUN2 0 6 浮動小数点絶対値 (FABS) *

[T 3 / T 3 Hのみ使用可能]

浮動小数点データの絶対値を求めます。

関連命令

条件入力

実行出力

-[A+1・A FABS B+1・B]

機能

- オペランド A + 1・A で指定する倍長レジスタに格納されている浮動小数点データ (被演算データ) の絶対値をとり、オペランド B + 1・B で指定する倍長レジスタに格納し、実行出力を ON します。
- 指定可能な被演算データの範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
- 被演算データが 00000000H の時は絶対値を 0 とします。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	定 数	修 飾
A+1・A	被演算データ											○	○	○	○	○	○	○	○	○	○	○	○	○	○	○		○
B+1・B	結果格納先												○	○	○	○	○	○	○	○	○	○	○	○	○	○		○

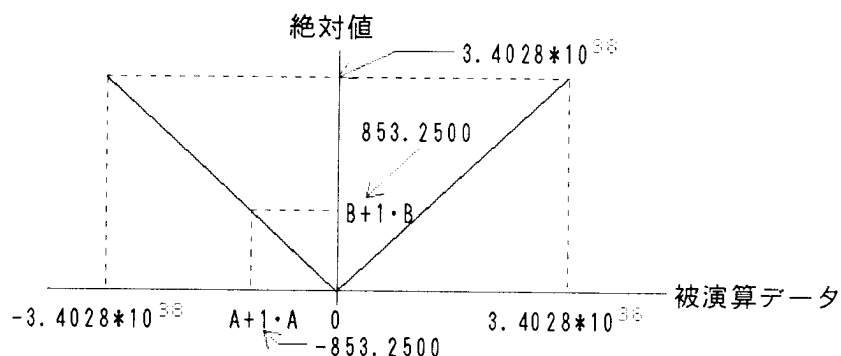
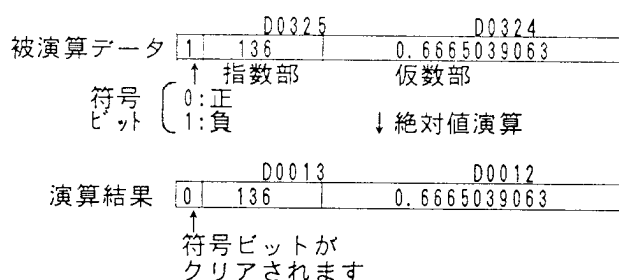
プログラム例

```

R0040      -853.2500      853.2500
┌──┴──┐ ┌──┴──┐
└─┬─┬─┘ └─┬─┬─┘
  D0325・D0324 FABS D0013・D0012
  
```

動作

- a 接点 R 0 0 4 0 が ON すると、倍長レジスタ D 0 3 2 5・D 0 3 2 4 に格納されている浮動小数点データ 8 5 3 . 2 5 0 0 (C 4 5 5 5 0 0 0 H) の絶対値をとり、倍長レジスタ D 0 0 1 3・D 0 0 1 2 に格納した後、実行出力を ON します。
- a 接点 R 0 0 4 0 が OFF の時は、演算を実行せず、実行出力を OFF にします。



© 2006 The Authors
Journal compilation © 2006 Blackwell Publishing Ltd

[T 3 / T 3 Hのみ使用可能]

関連命令

実行出力

- ・オペランドA + 1・Aで指定する倍長レジスタに格納されている浮動小数点データ（被演算データ）の符号を反転し、オペランドB + 1・Bで指定する倍長レジスタに格納し、実行出力をONします。
- ・指定可能な被演算データの範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。

- ・ a 接点 R 0 0 4 1 が O N すると、倍長レジスタ D 0 0 1 3・D 0 0 1 2 に格納されている浮動小数点データ 8 5 3 . 2 5 0 0 (4 4 5 5 5 0 0 0 H) の符号を反転し、演算結果 - 8 5 3 . 2 5 0 0 (C 4 5 5 5 0 0 0 H) を倍長レジスタ R W 0 8 9・R W 0 8 8 に格納した後、実行出力を O N にします。
- ・ a 接点 R 0 0 4 1 が O F F の時は、演算を実行せず、実行出力を O F F にします。

[illegible]

[T 2 E / T 2 N / T 3 / T 3 H 使用可能]

関連命令

オーバーフロー出力

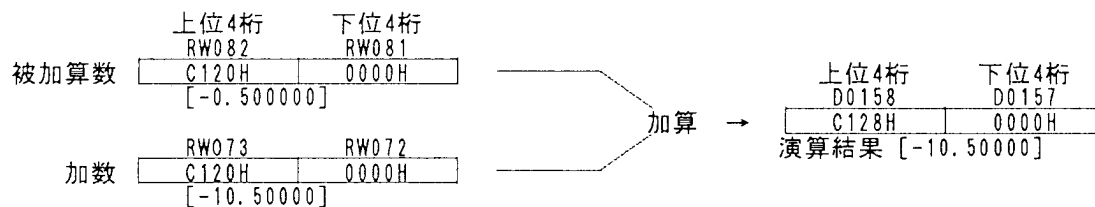
- ・オペランドA + 1・Aで指定する倍長レジスタに格納されている浮動小数点データ（被加数）とオペランドB + 1・Bで指定する倍長レジスタに格納されている浮動小数点データ（加数）を加算し、格納します。
- ・指定可能な被加数と加数の範囲は、 $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。なお、加算結果がオーバーフローする場合（加算結果が 3.40282×10^{38} を超える場合、または -3.40282×10^{38} より小さくなる場合）は、オーバーフロー出力をONし、演算結果としてリミット値（ 3.40282×10^{38} または -3.40282×10^{38} ）を格納します。
- ・浮動小数点データ00000000Hは0として扱います。

入力	処 理		出力
OFF	不実行		OFF
ON	実行	演算結果が範囲内	OFF
		演算結果がオーバーフロー	ON

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾				
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K		
A+1・A	被加算数											○	○	○	○	○	○	○	○	○	○	○			○	○			○	
B+1・B	加 数											○	○	○	○	○	○	○	○	○	○	○	○			○	○			○
C+1・C	和												○	○	○	○	○	○	○	○	○	○		○	○	○			○	

R004B -0.500000 -10.000000 -10.500000 Y035A
| | RW082·RW081 F+ RW073·RW072 → D0158·D0157] ————— ()

- ・ a 接点 R 0 0 4 B が O N すると、倍長レジスタ R W 0 8 2 ・ R W 0 8 1 に格納されている浮動小数点データ 0 . 5 0 0 0 0 0 0 と倍長レジスタ R W 0 7 3 ・ R W 0 7 2 に格納されている浮動小数点データ 1 . 0 0 0 0 0 0 0 を加算し、演算結果（浮動小数点データ 1 0 . 5 0 0 0 0 0 ）を倍長レジスタ D 0 1 5 8 ・ D 0 1 5 7 に格納した後、オーバーフロー出力を O F F にします。
- ・ a 接点 R 0 0 4 B が O F F の時は、演算を実行せず、オーバーフロー出力を O F F にします。



[illegible]

[T 2 E / T 2 N / T 3 / T 3 H 使用可能]

関連命令

オーバーフロー出力

条件入力	オーバーフロー出力
$- [A+1 \cdot A \quad E- \quad B+1 \cdot B \quad \rightarrow \quad C+1 \cdot C] \text{-----}$	

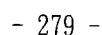
- ・オペランドA + 1・Aで指定するレジスタに格納されている浮動小数点データ（被減数）からオペランドB + 1・Bで指定するレジスタに格納されている浮動小数点データ（減数）を引き、オペランドC + 1・Cで指定するレジスタに減算の結果を格納します。
- ・指定可能な被減数と減数の範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
- ・減算の結果がオーバーフローする場合（減算結果が 3.40282×10^{38} を超える場合、または -3.40282×10^{38} より小さくなる場合）は、オーバーフロー出力をONし、演算結果としてリミット値（ 3.40282 ）です。
- ・浮動小数点データ00000000Hは0として扱います。

2×10^{38} または -3.40282×10^{38}) を格納しま

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾	
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J
A+1・A	被減数											○	○	○	○	○	○	○	○	○	○	○		○	○		
B+1・B	減 数											○	○	○	○	○	○	○	○	○	○	○		○	○		
C+1・C	差												○	○	○	○	○	○	○	○	○		○	○	○		

R004C 10.00000 -0.500000 10.500000 Y035A
 [RW078·RW077 F- RW034·RW033 + D0123·D0122] ————— ()

- ・a 接点 R004C が ON すると、倍長レジスタ RW078・RW077 に格納されている浮動小数点データ 10.00000 から倍長レジスタ RW034・RW033 に格納されている浮動小数点データ 0.50000 を引き、演算結果（浮動小数点データ 10.50000）を倍長レジスタ D0123・D0122 にそれぞれ格納し、オーバーフロー出力を OFF に設定します。
- ・a 接点 R004C が OFF の時は、演算を実行せず、オーバーフロー出力を OFF にします。



[T 2 E / T 2 N / T 3 / T 3 H 使用可能]

関連命令

オーバーフロー出力

$$- [A+1 \cdot A \quad F* \quad B+1 \cdot B \quad \rightarrow \quad C+1 \cdot C] \text{ } \underline{\hspace{10em}}$$

機能

入力	処 理	
OFF	不実行	
ON	実行	演算結果が範囲内
		演算結果がオーバーフロー

入力	処 理		出力
OFF	不実行		OFF
O N	実行	演算結果が範囲内	OFF
		演算結果がオーバーフロー	O N

オペランド

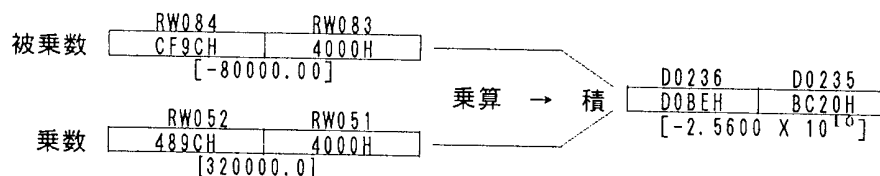
		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A+1・A	被乗数											○	○	○	○	○	○	○	○	○	○	○			○	○		○	
B+1・B	乗 数											○	○	○	○	○	○	○	○	○	○	○	○			○	○		○
C+1・C	積												○	○	○	○	○	○	○	○	○		○			○	○		○

プログラム例

R004D -80000.00 320000.0 -2.5600E 10 Y035A
 | I | RW084·RW083 F* RW052·RW051 → D0236·D0235 ()

動作

- ・ a 接点 R004D が ON すると、倍長レジスタ RW084・RW083 に格納されている浮動小数点データ 8000.00 と倍長レジスタ RW052・RW051 に格納されている浮動小数点データ 320000.0 の積を求め、演算結果（浮動小数点データ 5600E10）を倍長レジスタ D0236・D0235 に格納した後、オーバーフロー出力を OFF にします。
- ・ a 接点 R004D が OFF の時は、演算を実行せず、オーバーフロー出力を OFF にします。



FUN 2 1 1 浮動小数点除算 (F /) *

関連命令

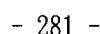
$$\vdash [A+1 \cdot A \quad E/ \quad B+1 \cdot B \quad \rightarrow \quad C+1 \cdot C] \text{-----}$$

入力	処 理		出力
OFF	不実行		OFF
ON	実行	演算結果が範囲内	OFF
		除数 = 0	ON
		演算結果がオーバーフロー	ON

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A+1・A	被除数											○	○	○	○	○	○	○	○	○	○	○		○	○			○
B+1・B	除 数											○	○	○	○	○	○	○	○	○	○	○		○	○			○
C+1・C	商												○	○	○	○	○	○	○	○	○	○	○	○	○	○		○

R004E 42.00000 4.000000 10.50000 Y035A
| | | RW038·RW037 F/ RW077·RW076 → D0222·D0221 | ()

- ・ a 接点 R004E が ON すると、倍長レジスタ RW038・RW037 に格納されている 42.00000 を倍長レジスタ RW077・RW076 に格納されている浮動小数点データ 4.000000 で割り、演算結果の商 10.500000 をレジスタ D0222・D0221 に格納した後、オーバーフロー出力を OFF にします。
- ・ a 接点 R004E が OFF の時は、演算を実行せず、実行出力を OFF にします。



© 2006 The Authors
Journal compilation © 2006 Blackwell Publishing Ltd

[T2E / T2N / T3 / T3H 使用可能]

関連命令

判定出力

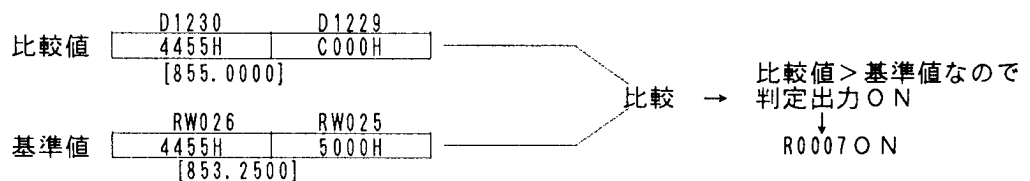
- ・オランダデータ（比較値）をオランダデータ（基準値）に比較し、基準値の時、基準出力をOFFに設定し、比較値 ≤ 基準値の時、基準出力をONに設定します。
- ・指定可能な比較値と基準値の範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
- ・格納されている比較値と基準値は比較によって変わることはありません。

入力	処 理		出力
OFF	不実行		OFF
ON	実行	$[A+1 \cdot A] > [B+1 \cdot B]$	ON
		$[A+1 \cdot A] \leq [B+1 \cdot B]$	OFF

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A+1・A	比較値											○	○	○	○	○	○	○	○	○	○	○		○	○			○
B+1・B	基準値											○	○	○	○	○	○	○	○	○	○	○		○	○			○

R0041 855.0000 853.2500 R0007
| | | [D1230·D1229 F> RW026·RW025] ————— ()

- ・ a 接点 R 0 0 4 1 が ON すると、倍長レジスタ D 1 2 3 0 ・ D 1 2 2 9 に格納されている 8 5 5 ・ R 0 0 0 0 (比較値)を倍長レジスタ R W 0 2 6 ・ R W 0 2 5 に格納されている
- ・ 動小数点データ 8 5 3 ・ 2 5 0 0 (基準値)と比較します。比較値 > 基準値となるので判定出力を ON し、この結果、R 0 0 0 7 が ON します。
- ・ a 接点 R 0 0 4 1 が OFF の時は、比較を行わず、判定出力を OFF に設定します。



FUN 2 1 3 浮動小数点比較・より大きいまたは等しい (F >=)

浮動小数点データを基準値に対して比較し、基準値より大あるいは等しいかどうかを判定します。

判定出力

$$- [A+1 \cdot A \quad F] = B+1 \cdot B$$

機能

- ・オペランドA + 1・Aで指定するレジスタに格納されている浮動小数点データ（比較値）をオペランドB + 1・Bで指定するレジスタに格納されている浮動小数点データ（基準値）に比較し、比較値 $\geq$ 基準値の時は判定出力をONに設定し、比較値 $<$ 基準値の時は判定出力をOFFに設定します。
- ・指定可能な比較値と基準値の範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
- ・格納されている比較値と基準値は比較によって変わることはありません。

入力	処 理		出力
OFF	不実行		OFF
ON	実行	$[A+1 \cdot A] \geq [B+1 \cdot B]$	ON
		$[A+1 \cdot A] < [B+1 \cdot B]$	OFF

オペランド

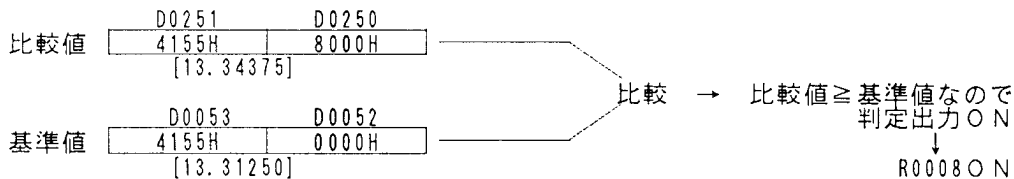
		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A+1・A	比較値											○	○	○	○	○	○	○	○	○	○	○			○	○			○
B+1・B	基準値											○	○	○	○	○	○	○	○	○	○	○			○	○			○

プログラム例

R0042 13.34375 13.31250 R0008
| | | D0251·D0250 F>= D0053·D0052]—————()

動作

- ・ a 接点 R0042 が ON すると、倍長レジスタ D0251・D0250 に格納されている
13.34375 (比較値) を倍長レジスタ D0053・D0052 に格納されている浮
動小数点データ 13.31250 (基準値) に比較します。比較値 ≥ 基準値となるので
判定出力を ON し、R0008 が ON になります。
・ a 接点 R0042 が OFF の時は、比較を行わず、判定出力を OFF に設定します。



© 2006 The Authors
Journal compilation © 2006 Blackwell Publishing Ltd

[T 2 E / T 2 N / T 3 / T 3 H 使用可能]

関連命令

判定出力

条件入力	判定出力
$\neg [A+1 \cdot A \quad F= \quad B+1 \cdot B \quad]$	_____

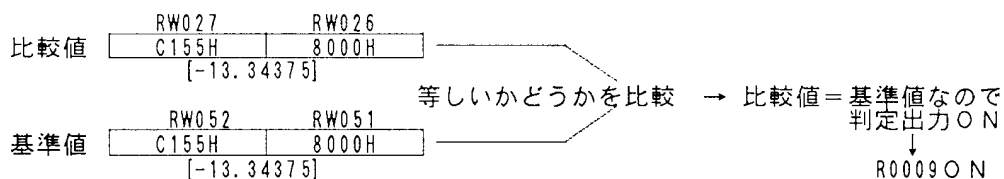
- ・オランダA + 1・Aで指定するレジスタに格納されている浮動小数点データ（比較値）をオランダB + 1・Bで指定するレジスタに格納されている浮動小数点データ（基準値）に比較し、比較値は基準値の時とは判定出力をOFFに設定し、比較値≠基準値の時とは判定出力をONに設定します。
- ・指定可能な比較値と基準値の範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
- ・格納されている比較値と基準値は比較によって変わることはありません。

入力	処 理		出力
OFF	不実行		OFF
0 N	実行	$[A+1 \cdot A] = [B+1 \cdot B]$	0 N
		$[A+1 \cdot A] \neq [B+1 \cdot B]$	OFF

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A+1・A	比較値											○	○	○	○	○	○	○	○	○	○	○		○	○			○
B+1・B	基準値											○	○	○	○	○	○	○	○	○	○	○		○	○			○

R0043 -13.34375 -13.34375 R0009
 | | [RW027·RW026 F= RW052·RW051] ————— ()

- ・a 接点 R0043 が ON すると、倍長レジスタ RW027・RW026 に格納されている -13.34375 (比較値) を倍長レジスタ D0052・D0051 に格納されている浮動小数点データ -13.34375 (基準値) に比較します。比較値 = 基準値となるので、判定出力を ON に設定します。
- ・a 接点 R0043 が OFF の時は、比較を行わず、判定出力を OFF に設定します。



浮動小数点比較・等しくない

FUN2 1 5 浮動小数点比較・等しくない (F<>) [T2E/T2N/T3/T3H使用可能]

浮動小数点データを基準値に対して比較し、等しくないことを判定します。 関連命令

条件入力

判定出力

-[A+1・A F<> B+1・B]

機能

- オペランドA+1・Aで指定するレジスタに格納されている浮動小数点データ（比較値）をオペランドB+1・Bで指定するレジスタに格納されている浮動小数点データ（基準値）に比較し、比較値≠基準値の時は判定出力をONに設定し、比較値=基準値の時は判定出力をOFFに設定します。
- 指定可能な比較値と基準値の範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
- 格納されている比較値と基準値は比較によって変わることはありません。

オペランド

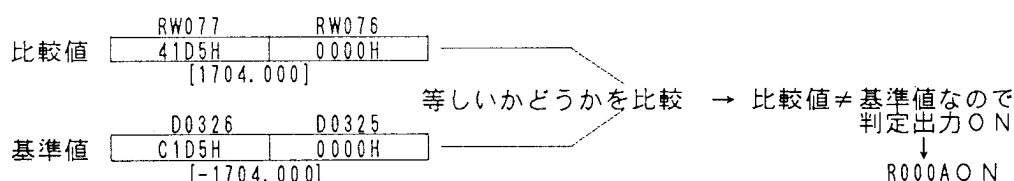
入力	処 理	出力
OFF	不実行	OFF
ON	実行	[A+1・A] ≠ [B+1・B] ON
		[A+1・A] = [B+1・B] OFF

		デバイス										レジスタ										INDEX			定数	修飾		
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A+1・A	比較値											○	○	○	○	○	○	○	○	○	○	○			○	○		○
B+1・B	基準値											○	○	○	○	○	○	○	○	○	○	○			○	○		○

プログラム例 R0044 1704.000 -1704.000 R000A
[RW077・RW076 F<> D0326・D0325]

動作

- a接点R0044がONすると、倍長レジスタRW077・RW076に格納されている1704.000（比較値）を倍長レジスタD0326・D0325に格納されている浮動小数点データ-1704.000（基準値）に比較します。比較値≠基準値となるので判定出力をONに設定します。
- a接点R0044がOFFの時は、比較を行わず、判定出力をOFFに設定します。



[T 2 E / T 2 N / T 3 / T 3 H 使用可能]

関連命令

判定出力

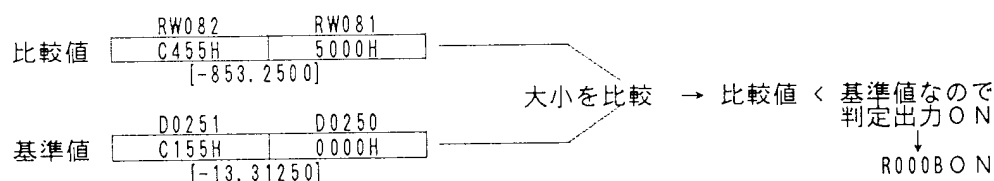
- ・オペランドA + 1・Aで指定するレジスタに格納されている浮動小数点データ（比較値）をオペランドB + 1・Bで指定するレジスタに格納されている浮動小数点データ（基準値）に比較し、基準値の時、判定出力をOFFに設定し、比較値の時、判定出力をONに設定します。
- ・指定可能な比較値と基準値の範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
- ・格納されている比較値と基準値は比較によって変わることはありません。

入力	処 理		出力
OFF	不実行		OFF
ON	実行	$[A+1 \cdot A] < [B+1 \cdot B]$	ON
		$[A+1 \cdot A] \geq [B+1 \cdot B]$	OFF

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A+1・A	比較値											○	○	○	○	○	○	○	○	○	○	○		○	○			○
B+1・B	基準値											○	○	○	○	○	○	○	○	○	○	○		○	○			○

R0045 -853.2500 -13.31250 R000B
 | | [RW082·RW081 F< D0251·D0250] ————— ()

- ・ a 接点 R0045 が ON すると、倍長レジスタ RW082・RW081 に格納されている 1853.2500 (比較値) を倍長レジスタ D0251・D0250 に格納されている浮動小数点データー 13.31250 (基準値) に比較します。比較値 < 基準値となるので、判定出力を ON に設定します。
- ・ a 接点 R0045 が OFF の時は、比較を行わず、判定出力を OFF に設定します。



浮動小数点比較・より小さいまたは等しい

FUN 2 1 7 浮動小数点比較・より小さいまたは等しい ($F \leq$) [T 2 E / T 2 N / T 3 / T 3 H 使用可能]

浮動小数点データを基準値に対して比較し、小または等しいことを判定します。	関連命令

条件入力

$$- [A+1 \cdot A \quad F \Leftarrow B+1 \cdot B]$$

機能

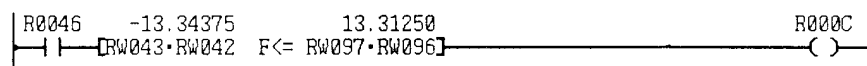
- ・オペランドA + 1・Aで指定するレジスタに格納されている浮動小数点データ（比較値）をオペランドB + 1・Bで指定するレジスタに格納されている浮動小数点データ（基準値）に比較し、比較値 ≤ 基準値の時は判定出力をOFFに設定し、比較値 > 基準値の時は判定出力をOFFに設定します。
- ・指定可能な比較値と基準値の範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
- ・格納されている比較値と基準値は比較によって変わることはありません。

入力	処	理	出力
OFF	不実行		OFF
ON	実行	$[A+1 \cdot A] \leq [B+1 \cdot B]$	ON
		$[A+1 \cdot A] > [B+1 \cdot B]$	OFF

オペランド

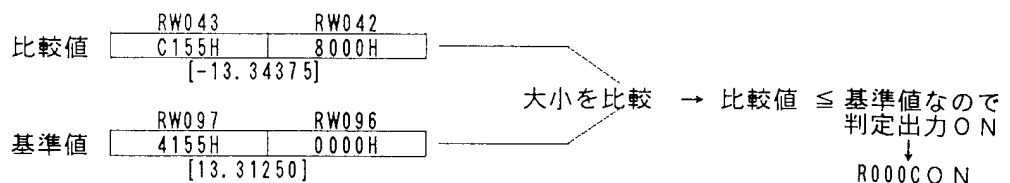
		デバイス										レジスタ										INDEX			定数	修飾			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A+1・A	比較値											○	○	○	○	○	○	○	○	○	○	○			○	○			○
B+1・B	基準値											○	○	○	○	○	○	○	○	○	○	○			○	○			○

プログラム例



動作

- ・ a 接点 R 0 0 4 6 が O N すると、倍長レジスタ R W 0 4 3 ・ R W 0 4 2 に格納されている - 1 3 , 3 4 3 7 5 (比較値) を倍長レジスタ R W 0 9 7 ・ R W 0 9 6 に格納されている浮動小数点データ 1 3 , 3 1 2 5 0 (基準値) に比較します。比較値 ≤ 基準値となるので判定出力を O N に設定します。
- ・ a 接点 R 0 0 4 6 が O F F の時は、比較を行わず、判定出力を O F F に設定します。



FUN 2 1 8 浮動小数点・上限リミット (FUL) *

[T 3 / T 3 Hのみ使用可能]

浮動小数点データに上限リミットを設定します。

關連命令

条件入力

リミットオーバー判定出力

$$\neg [A+1 \cdot A \quad \text{FUL} \quad B+1 \cdot B \quad \rightarrow \quad C+1 \cdot C] \text{---}$$

機能

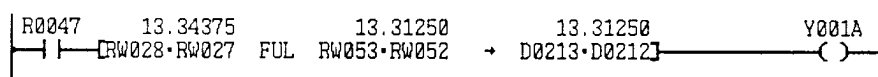
- 浮指ミ演格算す
るでリ被に演定一
てB限はタ被指オ
れ1(上時スCオ
さ+タ値レま・ッ
納B一ト長し・リ
格D点ミ倍定+リ
ナンドミする設C
タ数リミにドし
スベラ限定Fン格
ジオ動上指Fラ納
レが浮±でOペを
すタいるタCを値
定一デー・力ハ
指デれ算+定のミ
で算さ演C判値リ
A演納被D一限定
・(被格シランバ
1、タ比較ベオリ
+タ比オト限SON
Aース比オト限SON
ドデジにをッ上ジ
ラ点レ)タミ>レカ
ラ数(値)タミ>レカ
ベ小すトデし・倍
・オ動定ッ算納デ
の判定指定
~ 3.40282×10^{-38}

入力	処 理		出力
OFF	不実行		OFF
ON	実行	$[A+1 \cdot A] \leq [B+1 \cdot B]$	OFF
		$[A+1 \cdot A] > [B+1 \cdot B]$	ON

オペランド

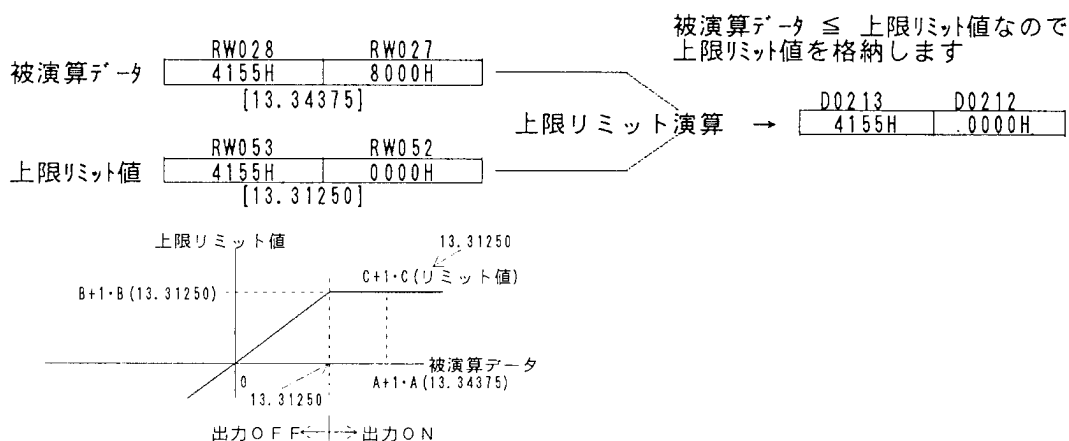
		デバイス										レジスタ												INDEX			定数	修飾		
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K				
A+1・A	被演算データ											○	○	○	○	○	○	○	○	○	○	○			○	○			○	
B+1・B	上限リミット値											○	○	○	○	○	○	○	○	○	○	○				○	○			○
C+1・C	リミット結果												○	○	○	○	○	○	○	○	○	○	○	○	○	○			○	

プログラム例



動作

- ・ a 接点 R 0 0 4 7 が O N すると、倍長レジスタ R W 0 2 8 ・ R W 0 2 7 に格納されている 1 3 ・ 3 4 3 7 5 (被演算データ) を倍長レジスタ R W 0 5 3 ・ R W 0 5 2 に格納されてい
る。浮動小数点データ 1 3 ・ 3 1 2 5 0 (上限リミット値) に比較します。被演算データ
≦ 上限リミット値となるので、上限リミット値 1 3 ・ 3 1 2 5 0 を倍長レジスタ D 0 2 1
3 ・ D 0 2 1 2 に格納し、リミットオーバー判定出力を O N に設定します。
- ・ a 接点 R 0 0 4 7 が O F F の時は、演算を行わず、リミットオーバー判定出力を O F F に
設定します。



[T 3 / T 3 Hのみ使用可能]

関連命令

リミットオーバー判定出力

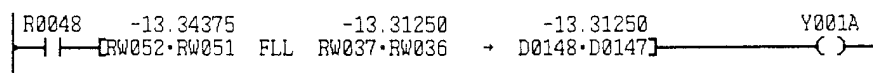
浮指ミ演格算す
 るでリ被に演定バ
 ー
 1. B 限は下時ス。でオ
 され1. (下)のジ。C
 納B+値レましC
 格ドデット倍定+リ
 タン点ミすに設C
 スペ小限定F格納
 ジオ動下指Fラ格
 レが浮≧でOペを
 する。いCをオ値
 定一テ1.出時ツ。下
 指デれ算+定のミま
 で算演C判値りしタ
 A 演納被ドット限定一
 1. (被格、ランバトに下設
 +タ比較ベオリタN演算
 Aース比オト限スO被
 ドデジにをッ下ジをな
 ラ数(値)タミ>レ長力可
 ペ小すドレシタ倍定
 ・オ動定ッ算納デる判指
 指
 3. 40282x10⁻³⁶

入力	処 理		出力
OFF	不実行		OFF
ON	実行	$[A+1 \cdot A] \geq [B+1 \cdot B]$	OFF
		$[A+1 \cdot A] < [B+1 \cdot B]$	ON

オペランド

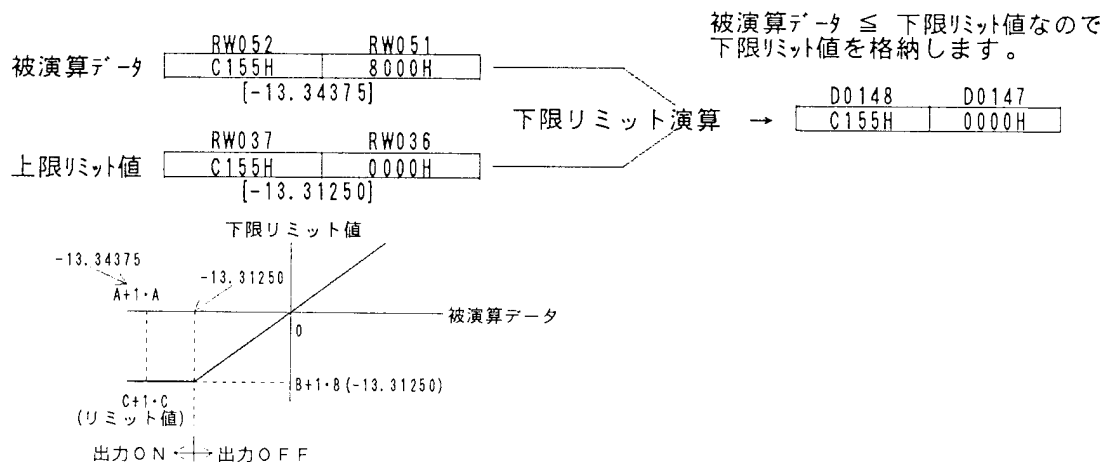
		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A+1-A	被演算データ											○	○	○	○	○	○	○	○	○	○	○			○	○		
B+1-B	下限リミット値											○	○	○	○	○	○	○	○	○	○	○	○			○	○	
C+1-C	リミット結果												○	○	○	○	○	○	○	○	○	○	○			○	○	

プログラム例



動作

- ・ a 接点 R0048 が ON すると、倍長レジスタ RW052・RW051 に格納されている -13.34375 (被演算データ) を倍長レジスタ RW037・RW036 に格納されている浮動小数点データ -13.31250 (下限リミット値) に比較します。被演算データ ≤ 下限リミット値となるので、下限リミット値 -13.31250 を倍長レジスタ D0148・D0147 に格納し、リミットオーバー判定出力を ON に設定します。
- ・ a 接点 R0048 が OFF の時は、演算を行わず、リミットオーバー判定出力を OFF に設定します。



条件入力

デッドバンド内判定出力

$-[A+1 \cdot A \quad FDB \quad B+1 \cdot B \quad \rightarrow \quad C+1 \cdot C \quad]$

- オペランドA + 1・Aで指定するレジスタの内容(被演算データ(浮動小数点データ))がオペランドB + 1・Bで指定するレジスタの内容(デッドバンド幅(浮動小数点データ))の範囲内にあるかどうかを判定し、その結果に基づいて、オペランドC + 1・Cで指定するレジスタに格納する演算結果を以下のよう
に決定します。被演算データとデッドバンド幅のデータの範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
 - (1) |被演算データ| ≤ |デッドバンド幅|の場合、演算結果0(0000000H)を格納し、デッドバンド内判定出力をONにします。
 - (2) |被演算データ| > |デッドバンド幅|で、しかも被演算データ < 0の場合、演算結果の被演算データ + |デッドバンド幅|を格納し、バンド内判定出力をOFFにします。
 - (3) |被演算データ| < |デッドバンド幅|で、しかも被演算データ > 0の場合、演算結果の被演算データ - |デッドバンド幅|を格納し、バンド内判定出力をOFFにします。

入力	処 理		出力
OFF	不実行		OFF
ON	実行	$ A+1 \cdot A \leq B+1 \cdot B $	ON
		$ A+1 \cdot A > B+1 \cdot B $	OFF

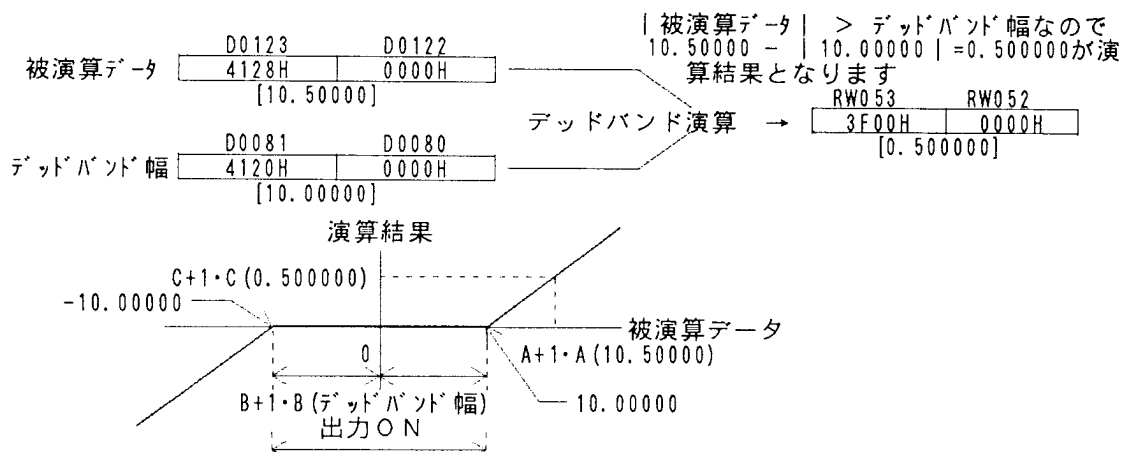
オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾				
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K		
A+1・A	被演算データ											○	○	○	○	○	○	○	○	○	○	○	○		○	○			○	
B+1・B	デッドバンド											○	○	○	○	○	○	○	○	○	○	○	○			○	○			○
C+1・C	結果格納先												○	○	○	○	○	○	○	○	○	○		○	○	○				○

R0049 10.50000 10.00000 0.50000 Y001A
 | | D0123·D0122 FDB D0081·D0080 → RW053·RW052] ()

・ a 接点 R 0 0 4 9 が ON すると、倍長レジスタ D 0 1 2 3 ・ D 0 1 2 2 の内容 (1 0 . 0 0 0 0) をデッドバンド幅の示す倍長レジスタ D 0 0 8 1 ・ D 0 0 8 0 の内容 (1 0 0 0 0 0) と比較し、その差判定出力を O F F にします。倍長レジスタ R W 0 5 3 ・ R W 0 2 に格納し、デッドバンドの出力を O F F にします。

・ a 接点 X 0 0 4 9 が O F F の時は、演算を実行せず、判定出力を O F F にします。



浮動小数点开平

FUN2 2 1 浮動小数点开平 (FRT) *

[T 3 / T 3 H のみ使用可能]

浮動小数点データの平方根を計算します。

関連命令

条件入力

実行出力

-[A+1・A FRT B+1・B]

機能

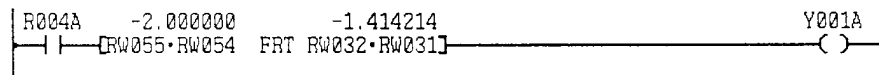
- オペランド A + 1 ・ A で指定する倍長レジスタに格納されている浮動小数点データ (演算データ) の絶対値の平方根を求め、演算データの符号を求めた平方根に付加し、オペランド B + 1 ・ B で指定する倍長レジスタに格納し、実行出力を ON します。
- 指定可能な演算データの範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
- 浮動小数点データ 00000000H は 0 として扱います。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

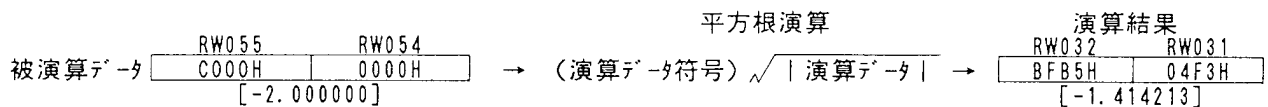
		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A+1・A	被演算データ											○	○	○	○	○	○	○	○	○	○	○			○	○			○
B+1・B	平方根												○	○	○	○	○	○	○	○	○	○		○	○	○			○

プログラム例



動作

- a 接点 R 0 0 4 A が ON すると、倍長レジスタ RW 0 5 5 ・ RW 0 5 4 に格納されている -2.000000 (演算データ) の絶対値の平方根を算出し、演算データの符号を算出した結果に付加し、倍長レジスタ RW 0 3 2 ・ RW 0 3 1 に格納し、実行出力を ON に設定します。
- a 接点 R 0 0 4 A が OFF の時は、演算を行わず、実行出力を OFF に設定します。



[T 3 / T 3 Hのみ使用可能]

関連命令

実行出力

$$- [A+1 \cdot A \quad \text{FPID} \quad B+1 \cdot B \quad \rightarrow \quad C+1 \cdot C]$$

・オペランドB+1・Bで指定する倍長レジスタを先頭とする7個の倍長レジスタに格納されている浮動小数点形式で表示したパラメータ（比例係数 K_p 、積分係数 K_{IH} 、 K_{IL} 、微分係数 K_{DH} 、 K_{DL} 、ギャップ定数 G 、リミット定数 L ）とオペランドC+1・Cで指定する倍長レジスタに続く4個の倍長レジスタに格納されているパラメータ（前回入力偏差 e_{-1} 、前回入力値 P_{-1} 、前々回入力値 P_{-2} ）に基づいて、オペランドA+1・Aで指定する倍長レジスタを先頭とする2個の倍長レジスタに格納されている2つの浮動小数点データ（入力値 P 、設定値 S ）について以下のよう¹に微分先行速度型PID演算を実行し、演算結果（出力データ ΔM 、前回入力偏差 e_{-1} 、前回入力値 P_{-1} 、前々回入力値 P_{-2} ）をオペランドC+1・Cで指定する倍長レジスタを先頭とする4個の倍長レジスタに格納し、実行出力をONします。

入力	処 理	出力
OFF	不実行	OFF
ON	実行 : KIHとKDL $\neq 0$	ON
	不実行 : KIHまたはKDL = 0	OFF

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A+1・A	入力先頭											○	○	○	○	○	○	○	○	○	○	○	○		○	○		
B+1・B	ハ'ラメータ先頭											○	○	○	○	○	○	○	○	○	○	○	○					
C+1・C	出力先頭												○	○	○	○	○	○	○	○	○	○	○	○				

R0001 20.00000 1.500000 741.3750
| | | [RW054·RW053] FPID RW079·RW078 → D0148·D0147]

- ・ a 接点 R0001 が ON すると、オペランド B+1・B で指定する倍長レジスタを先頭とする 7 個の倍長レジスタ RW079・RW078 の内容 ($K_p=1.500000$)、RW081・RW080 の内容 ($K_{IH}=5.000000$)、RW083・RW082 の内容 ($K_{IL}=25.00000$)、RW085・RW084 の内容 ($K_{DH}=18.00000$)、RW087・RW086 の内容 ($K_{LL}=6.400000$)、RW089・RW088 の内容 ($G=0$)、RW091・RW090 の内容 ($L=100.0000$) とオペランド C+1・C で指定する倍長レジスタ (D0148・D0147) に続く 3 個の倍長レジスタ (D0150・D0149 ~ D0154・D0153) に格納されている前回入力偏差 e_{-1} (42.00000)、前回入力値 P_{-1} (28.00000)、前々回入力値 P_{-2} (16.00000) に基づいて、オペランド A+1・A で指定する倍長レジスタ RW054・RW053 の内容 ($P=20.00000$) と RW056・RW055 の内容 ($S=100.0000$) の入力データに対する微分先行速度型 PID 演算を実行し、その結果 ($\Delta M=741.375$ 、 $e_{-1}=80.00000$ 、 $P_{-1}=20.00000$ 、 $P_{-2}=42.00000$) をオペランド C+1・C で指定する倍長レジスタを先頭とする 5 個の倍長レジスタ (D0148・D0147 ~ D0154・D0153) にそれぞれ格納し、実行出力を ON します。
- ・ a 接点 R0001 が OFF の時は、演算を実行せず、実行出力を OFF としますが、 ΔM に 0 を、 e_{-1} に 80.00000 を、 P_{-1} と P_{-2} に 20.00000 を設定します。

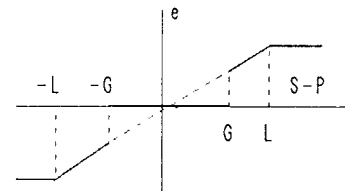
A+1・A	入力値 P	B+1・B	比例係数 K _P	C+1・C	出力データ ΔM
A+3・A+2	設定値 S	B+3・B+2	積分係数 K _{IH}	C+3・C+2	前回入力偏差 e ₋₁
		B+5・B+4	積分係数 K _{IL}	C+5・C+4	前回入力値 P ₋₁
		B+7・B+6	微分係数 K _{DH}	C+7・C+6	前々回入力値 P ₋₂
		B+9・B+8	微分係数 K _{DL}		
		B+11・B+10	キップ定数 G		
		B+13・B+12	リミット定数 L		

[浮動小数点速度型 P I D 演算]

$$\text{出力データ } \Delta M = K_P \cdot \left\{ (e - e_{-1}) + \left(\frac{|K_{IL}| \cdot e}{|K_{IH}|} \right) + \left(\frac{|K_{DH}|}{|K_{DL}|} \right) \cdot (2P_{-1} - P - P_{-2}) \right\}$$

ここで、 e は入力偏差であり、 $(S-P)$ の値に対するリミット・ギャップ演算から求めます（図参照）。

- $K_{IH} = 0$ 、または $K_{DL} = 0$ の場合は、演算を実行せずに実行出力を OFF し、しかもエラーフラグ（除算エラー）をセットします。
- 指定可能なデータの範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
- 浮動小数点データ 00000000H は 0 として扱います。
- 算出した出力データ ΔM が浮動小数点形式で表記できる値よりも 0 に近づく場合 (2^{-127} 以下の場合) は、出力データ ΔM を 0 に設定します。また、出力データ $\Delta M < -3.40282 \times 10^{38}$ の時は -3.40282×10^{38} を、出力データ $\Delta M > 3.40282 \times 10^{38}$ の時は 3.40282×10^{38} を ΔM のリミット値として格納します。
- 条件入力が OFF の時は、演算を実行せずに、 ΔM に 0、 e_{-1} に $e (=S-P)$ の内容、 P_{-1} と P_{-2} に P をそれぞれ格納し、実行出力を OFF に設定します。



浮動小数点偏差2乗型PID

FUN223 浮動小数点偏差2乗型PID (FPID2)

[T3/T3Hのみ使用可能]

浮動小数点偏差2乗速度型PID演算を実行します。

関連命令

条件入力

実行出力

$[A+1 \cdot A \quad \text{FPID2} \quad B+1 \cdot B \quad \rightarrow \quad C+1 \cdot C]$

機能

- オペランドB+1・Bで指定する倍長レジスタを先頭とする7個の倍長レジスタに格納されている比例係数 K_P 、積分係数 K_{IH} 、 K_{IL} 、微分係数 K_{DH} 、 K_{DL} 、ギャップ定数 G 、リミット定数 L とオペランドC+1・Cで指定する倍長レジスタに続く3個の倍長レジスタに格納されている前回入力偏差の2乗 $e_{-1} \cdot |e_{-1}|$ 、前回入力値の2乗 $P_{-1} \cdot |P_{-1}|$ 、前々回入力値の2乗 $P_{-2} \cdot |P_{-2}|$ を用いて、オペランドA+1・Aで指定する倍長レジスタを先頭とする2個の倍長レジスタに格納されている入力値 P 、設定値 S について、偏差2乗速度型PID演算を以下のように行い、その結果（出力データ ΔM 、前回入力偏差の2乗 $e_{-1} \cdot |e_{-1}|$ 、前回入力値の2乗 $P_{-1} \cdot |P_{-1}|$ 、前々回入力値の2乗 $P_{-2} \cdot |P_{-2}|$ ）をオペランドC+1・Cで指定する倍長レジスタを先頭とする4個の倍長レジスタに格納します。演算後に実行出力をONします。

入力	処 理	出力
OFF	不実行	OFF
ON	実行: K_{IH} と $K_{DL} \neq 0$	ON
	NOP: K_{IH} または $K_{DL} = 0$	OFF

オペランド

		デバイス										レジスタ										INDEX			定数	修飾		
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A+1・A	入力先頭											○	○	○	○	○	○	○	○	○	○	○			○	○		○
B+1・B	ハ'ラメータ先頭											○	○	○	○	○	○	○	○	○	○	○						○
C+1・C	出力先頭												○	○	○	○	○	○	○	○	○	○	○					○

プログラム例

R0002 10.00000 2.40000 5.9874E08
 $[R0012 \cdot RW011 \text{ FPID2 } D0123 \cdot D0122 \rightarrow RW053 \cdot RW052]$

動作

- a接点R0002がONすると、オペランドB+1・Bで指定する倍長レジスタを先頭とする7個の倍長レジスタD0123・D0122の内容($K_P=24.00000$)、D0125・D0124の内容($K_{IH}=10.00000$)、D0127・D0126の内容($K_{IL}=15.00000$)、D0129・D0128の内容($K_{DH}=45.00000$)、D0131・D0130の内容($K_{DL}=5.00000$)、D0133・D0132の内容($G=0$)、D0135・D0134の内容($L=100.0000$)とオペランドC+1・Cで指定する倍長レジスタRW053・RW052に続く3個の倍長レジスタRW055・RW054~RW059・RW058に格納されている前回入力偏差 $e_{-1} \cdot |e_{-1}|$ (24336.00)、前回入力値 $P_{-1} \cdot |P_{-1}|$ (64.00000)、前々回入力値 $P_{-2} \cdot |P_{-2}|$ (9.00000)を用いて、オペランドA+1・Aで指定するレジスタRW012・RW011の内容($P=10.00000$)とRW014・RW013の内容($S=10000.00$)の入力データについて、微分先行速度型PID演算を実行し、その結果($\Delta M=5.9874E08$ 、 $e_{-1} \cdot |e_{-1}|=10000.00$ 、 $P_{-1} \cdot |P_{-1}|=100.0000$ 、 $P_{-2} \cdot |P_{-2}|=64.0000$)をオペランドC+1・Cで指定する倍長レジスタを先頭とする4個の倍長レジスタ(RW053・RW052~RW059・RW058)にそれぞれ格納します。演算後に実行出力をONします。
- a接点R0002がOFFの時は、演算を実行せず、実行出力をOFFに設定しますが、 ΔM に0を、 $e_{-1} \cdot |e_{-1}|$ に $e \cdot |e|$ ($= (S - P) \cdot |S - P|$)の値を、 $P_{-1} \cdot |P_{-1}|$ と $P_{-2} \cdot |P_{-2}|$ に $P \cdot |P|$ の値を設定します。

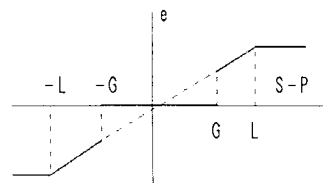
A+1・A	入力設定	P	B+1・B	比例係数	K_P
A+3・A+2	設定値	S	B+3・B+2	積分係数	K_{IH}
			B+5・B+4	積分係数	K_{IL}
C+1・C	出力データ	ΔM	B+7・B+6	微分係数	K_{DH}
C+3・C+2	前回入力偏差の2乗	$e_{-1} \cdot e_{-1} $	B+9・B+8	微分係数	K_{DL}
C+5・C+4	前回入力値の2乗	$P_{-1} \cdot P_{-1} $	B+11・B+10	ギャップ定数	G
C+7・C+6	前々回入力値の2乗	$P_{-2} \cdot P_{-2} $	B+13・B+12	リミット定数	L

[偏差2乗速度型P | D演算]

$$\text{出力データ } \Delta M = K_p \cdot \{ (e \cdot |e| - e_{-1} \cdot |e_{-1}|) + \left(\frac{|K_{IL}| \cdot |e| \cdot |e|}{|K_{IH}|} \right) + \left(\frac{|K_{DH}|}{|K_{DL}|} \right) \cdot (2P_{-1} \cdot |P_{-1}| - P \cdot |P| - P_{-2} \cdot |P_{-2}|) \}$$

ここで、 e は入力偏差であり、 $(S-P)$ の値に対するリミット・ギャップ演算から求めます（図参照）。

- 指定可能なデータの範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
- 浮動小数点データ00000000Hは0として扱います。
- 算出した出力データ ΔM が浮動小数点形式で表記できる値よりも0に近く場合（ 2^{-127} 以下の場合）は、出力データ ΔM を0に設定します。
- また、出力データ $\Delta M < -3.40282 \times 10^{38}$ の時は -3.40282×10^{38} を、出力データ $\Delta M > 3.40282 \times 10^{38}$ の時は 3.40282×10^{38} を ΔM のリミット値として格納します。
- 条件入力がOFFの時は、演算を実行せずに、 ΔM に0、 e_{-1} に e （ $=S-P$ ）の内容、 P_{-1} と P_{-2} に P をそれぞれ格納し、実行出力をOFFに設定します。
- $K_{IH}=0$ 、または $K_{DL}=0$ の場合は、演算を実行せずに実行出力をOFFし、しかもエラーフラグ（除算エラー）をセットします。



浮動小数点正弦関数

FUN2 2 4 浮動小数点正弦関数 (FSIN) *	[T 3 / T 3 H のみ使用可能]
浮動小数点データ形式で正弦関数演算を実行します。	関連命令

条件入力	実行出力
- [A+1・A FSIN B+1・B] -	

- 機能
- オペランド A + 1 ・ A で指定する倍長レジスタに格納されている浮動小数点データ（入力データ、単位：deg）について正弦値を算出し、オペランド B + 1 ・ B で指定する倍長レジスタに格納します。格納後、実行出力を ON します。
 - 指定可能な入力データの範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
 - 浮動小数点データ 00000000H を 0 として扱います。
 - 演算結果が 0 に近づき、浮動小数点データ形式で表記できない場合 (2^{-127} 以下の場合) は、演算結果を 0 (00000000H) とします。
 - A + 1 ・ A で指定されるデータが $-16777215 \sim 16777215$ の範囲を超えた場合には、B + 1 ・ B に 0 を格納します。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

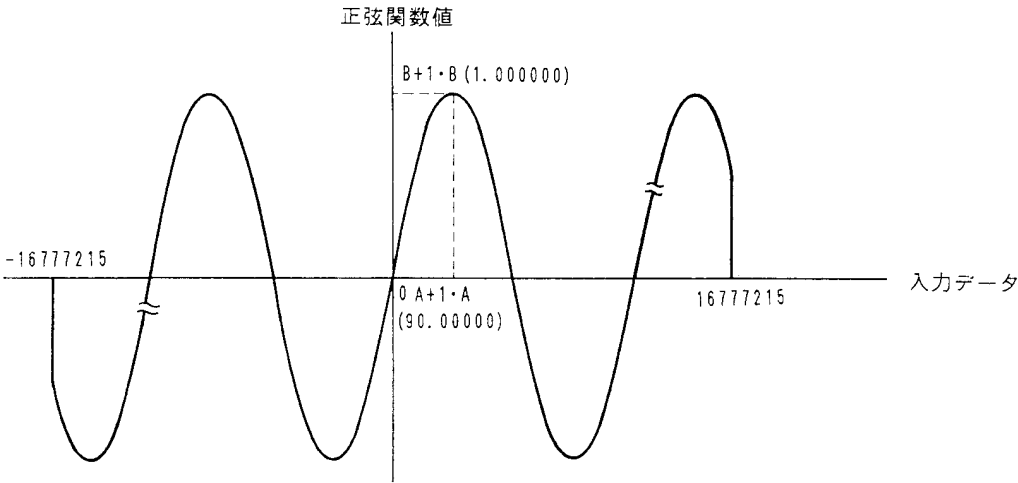
オペランド

		デバイス										レジスタ										INDEX			定数	修飾			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A+1・A	入力データ											○	○	○	○	○	○	○	○	○	○	○			○	○			○
B+1・B	結果格納先												○	○	○	○	○	○	○	○	○	○		○	○	○			○

プログラム例

```
R0050      90.00000      1.000000
┌──────────┴──────────┐
└─[RW029・RW028 FSIN D0227・D0226]─┘
```

- 動作
- a 接点 R 0 0 5 0 が ON すると、倍長レジスタ RW 0 2 9 ・ RW 0 2 8 の内容 (9 0 . 0 0 0 0) の正弦値を計算し、倍長レジスタ D 0 2 2 7 ・ D 0 2 2 6 に格納し、実行出力を ON します。
 - a 接点 R 0 0 5 0 が OFF の時は、演算を実行せず、出力を OFF にします。



[T 3 / T 3 Hのみ使用可能]

命令連関

実行出力

機能

- ・オペランドA + 1・Aで指定する倍長レジスタに格納されている浮動小数点データ（入力データ、単位：deg）について余弦値を算出し、オペランドB + 1・Bで指定する倍長レジスタに格納します。格納後、実行出力をONします。
- ・指定可能な入力データの範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
- ・浮動小数点データ00000000Hを0として扱います。
- ・A + 1・Aで指定されるデータが $-16777215 \sim 16777215$ の範囲を超えた場合には、B + 1・Bに0を格納します。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

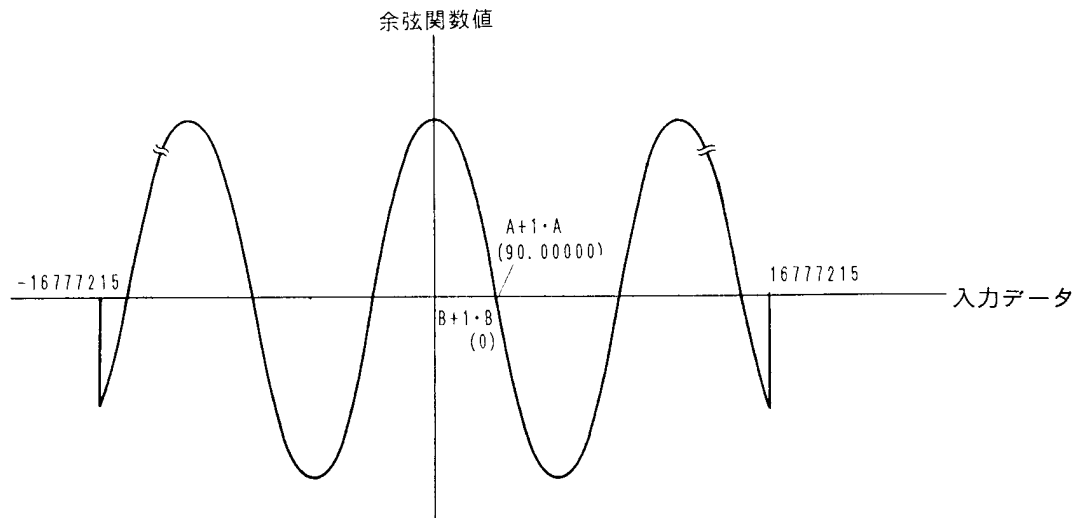
オペランド

		デバイス										レジスタ												INDEX			定数	修飾	
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A+1・A	入力データ											○	○	○	○	○	○	○	○	○	○	○			○	○			○
B+1・B	結果格納先												○	○	○	○	○	○	○	○	○	○		○	○	○			○

プログラム例	R0051	90.000000	0

動作

- ・ a 接点 R 0 0 5 1 が O N すると、倍長レジスタ R W 0 3 4 ・ R W 0 3 3 の内容 (9 0 . 0 0 0 0) の余弦値を計算し、倍長レジスタ D 0 1 2 4 ・ D 0 1 2 3 に格納し、実行出力を O N します。
- ・ a 接点 R 0 0 5 1 が O F F の時は、演算を実行せず、出力を O F F とします。



浮動小数点正接関数

FUN 2 2 6 浮動小数点正接関数 (FTAN) *

[T 3 / T 3 Hのみ使用可能]

浮動小数点データ形式で正接関数演算を実行します。

関連命令

条件入力

実行出力

-[A+1・A FTAN B+1・B]

機能

- ・オペランド A + 1・A で指定する倍長レジスタに格納されている浮動小数点データ (入力データ、単位: deg) について正接値を算出し、オペランド B + 1・B で指定する倍長レジスタに格納します。格納後、実行出力を ON します。指定可能な入力データの範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
- ・浮動小数点データ 00000000H を 0 として扱います。
- ・演算結果が浮動小数点データ形式で表記できない場合 (2^{-127} 以下) は、演算結果を 0 (00000000H) とします。
- ・入力データが $90 \pm 180n$ ($n=0, 1, 2, 3, \dots$) の場合は、リミット値を演算結果として格納し、実行出力を ON します。
- ・A + 1・A で指定されるデータが -16777215 から 16777215 の範囲を超えた場合には B + 1・B に 0 を格納します。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾	
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J
A+1・A	入力データ											○	○	○	○	○	○	○	○	○	○	○			○	○	
B+1・B	結果格納先												○	○	○	○	○	○	○	○	○	○	○	○	○	○	

プログラム例

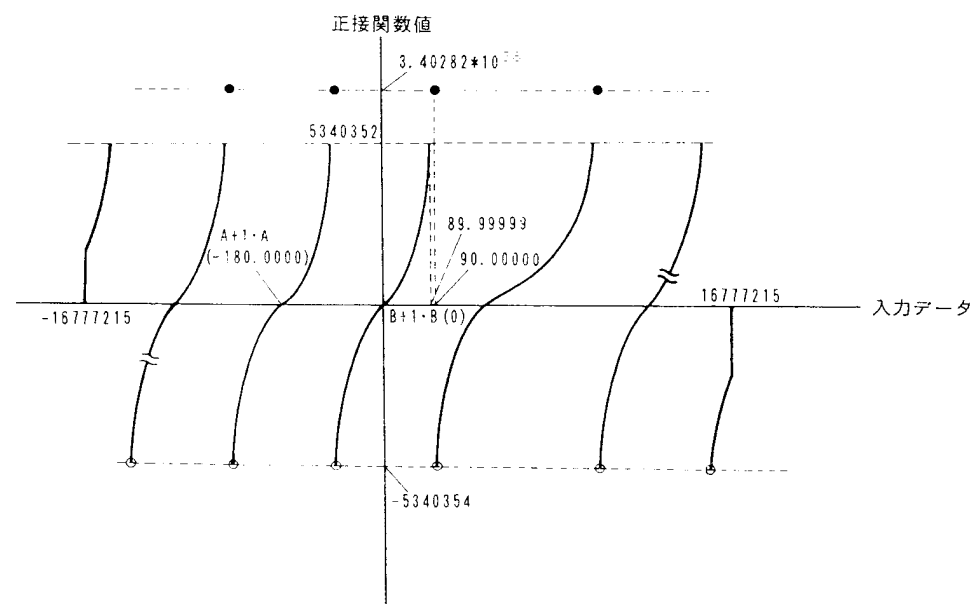
```

R0052  -180.0000      0
┌──┴──┐ [RW047・RW046 FTAN D0124・D0123] ───────────────────┐

```

動作

- ・a 接点 R 0 0 5 2 が ON すると、倍長レジスタ RW 0 4 7・RW 0 4 6 の内容 (-180.0000) の正接値を計算し、倍長レジスタ D 0 1 2 4・D 0 1 2 3 に格納し、実行出力を ON します。
- ・a 接点 R 0 0 5 2 が OFF の時は、演算を実行せず、出力を OFF とします。



[T 3 / T 3 Hのみ使用可能]

浮動小数点データ形式で逆正弦関数演算を実行します。

関連命令

条件入力

実行出力

$$- [A+1 \cdot A \quad \text{FASIN } B+1 \cdot B]$$

機能

- ・オペランドA+1・Aで指定する倍長レジスタに格納されている浮動小数点データについて逆正弦値（単位：deg）を算出し、オペランドB+1・Bで指定する倍長レジスタに格納します。格納後、実行出力をONします。
- ・指定可能な入力データの範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
- ・浮動小数点データ00000000Hを0として扱います。
- ・入力データ >1.000000 の場合は90.00000を、入力データ <-1.00000 の場合は-90.00000を演算結果として格納し、実行出力をONします。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

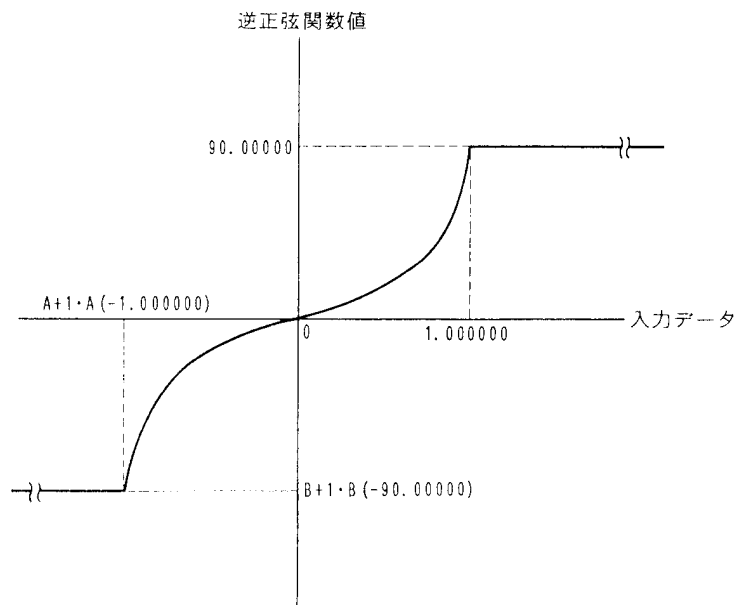
		デバイス										レジスタ										INDEX			定数	修飾				
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K		
A+1・A	入力データ											○	○	○	○	○	○	○	○	○	○	○			○	○			○	
B+1・B	結果格納先												○	○	○	○	○	○	○	○	○	○		○		○	○			○

プログラム例

R0053 -1.000000 -90.00000
| | | [RW064·RW063 FASIND0120·D0119]

動作

- ・a接点R0053がONすると、倍長レジスタRW064・RW063の内容（-1，000000）の逆正弦値を計算し、結果を倍長レジスタD0120・D0119に格納し実行出力をONにします。
- ・a接点R0053がOFFの時は、演算を実行せず、出力をOFFとします。



[T 3 / T 3 H のみ使用可能]

命令 連関

実行出力

機能

- ・オペランドA + 1・Aで指定する倍長レジスタに格納されている浮動小数点データについて逆余弦値（単位：deg）を算出し、オペランドB + 1・Bで指定する倍長レジスタに格納します。格納後、実行出力をONにします。
- ・指定可能な入力データの範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
- ・浮動小数点データ00000000Hを0として扱います。
- ・入力データ >1.000000 の時は0（00000000H）を、入力データ <-1.000000 の時は180（43340000H）を演算結果として格納し、実行出力をONにします。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

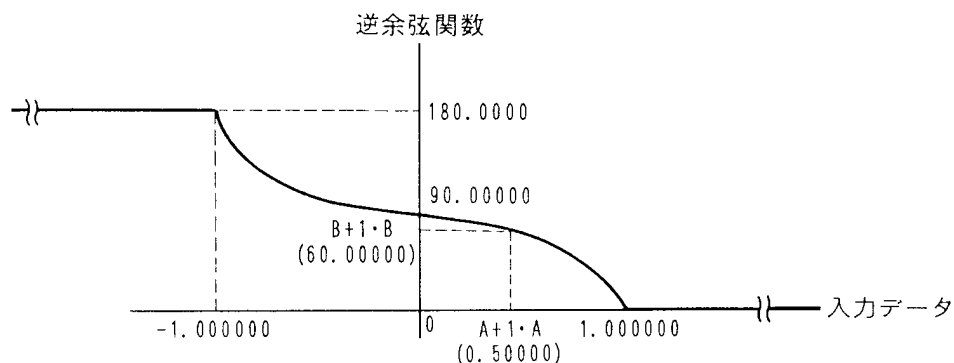
オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A+1・A	入力データ											○	○	○	○	○	○	○	○	○	○	○			○	○		○
B+1・B	結果格納先												○	○	○	○	○	○	○	○	○	○		○	○	○		○

R0054 0.500000 60.00000
| | | [RW018·RW017 FACOSD0111·D0110]

動作

- ・a接点R0054がONすると、倍長レジスタRW018・RW017の内容(0.500000)の逆余弦値を計算し、結果を倍長レジスタD0111・D0110に格納し実行出力をONにします。
- ・a接点R0054がOFFの時は、演算を実行せず、出力をOFFとします。



浮動小数点逆正接関数

FUN2 2 9 浮動小数点逆正接関数 (FATAN)*

[T3/T3Hのみ使用可能]

浮動小数点データ形式で逆正接関数演算を実行します。

関連命令

条件入力

実行出力

└C A+1・A FATAN B+1・B ┐

機能

- オペランドA+1・Aで指定する倍長レジスタに格納されている浮動小数点データについて逆正接値(単位:deg)を算出し、オペランドB+1・Bで指定する倍長レジスタに格納します。格納後、実行出力をONにします。
- 指定可能な入力データの範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
- 浮動小数点データ00000000Hは0として扱います。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

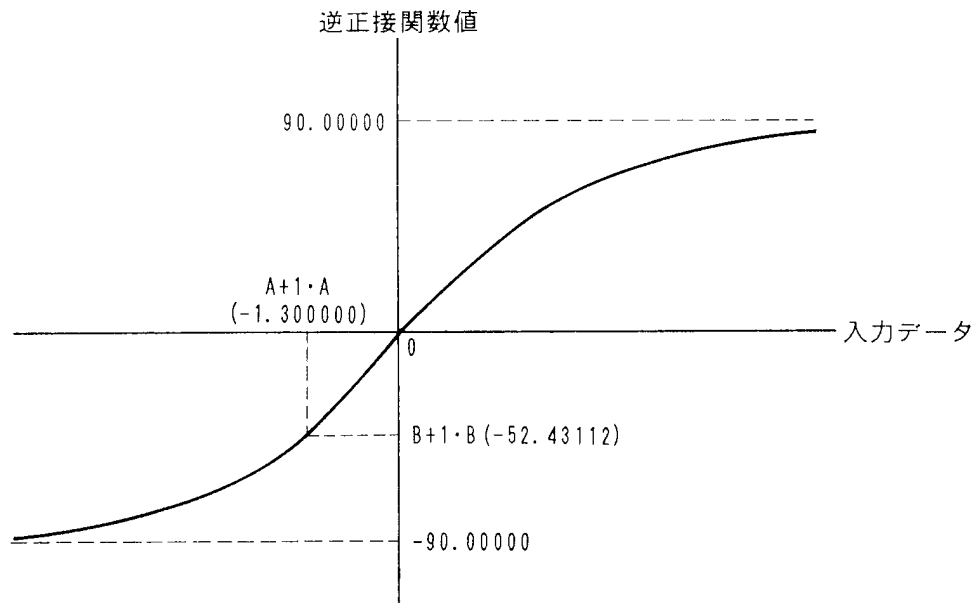
		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A+1・A	入力データ											○	○	○	○	○	○	○	○	○	○	○			○	○		○
B+1・B	結果格納先												○	○	○	○	○	○	○	○	○	○		○	○	○		○

プログラム例

R0055 -1.300000 -52.43112
┌┴┐ ┌RW048・RW047 FATAND0884・D0883┐

動作

- a接点R0055がONすると、倍長レジスタRW048・RW047の内容(-1.300000)の逆正接値を計算し、結果を倍長レジスタD0884・D0883に格納し実行出力をONにします。
- a接点R0055がOFFの時は、演算を実行せず、出力をOFFとします。



FUN230	浮動小数点指数関数 (FEXP) *

[T 3 / T 3 Hのみ使用可能]

浮動小数点データ形式で指数関数演算を実行します。

関連命令

条件入力

オーバーフロー出力

$$- [A+1 \cdot A \quad \text{FEXP} \quad B+1 \cdot B]$$

機能

- ・オペランドA+1・Aで指定する倍長レジスタに格納されている浮動小数点データ（入力データ）の指数を算出し、オペランドB+1・Bで指定する倍長レジスタに格納します。
- ・指定可能な入力データの範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
- ・浮動小数点データ00000000Hを0として扱います。
- ・入力データが88.02968596（42B00F34H）以上の時は、リミット値（ -3.40282×10^{38} または 3.40282×10^{38} ）を演算結果として格納し、オーバーフロー出力をONにします。
- ・演算結果が0に近づき、浮動小数点データ形式で表記できない場合（ 2^{-127} 以下の場合）は、演算結果を0（00000000H）とします。

入力	処 理		出力
OFF	不実行		OFF
ON	実行	正常	OFF
		演算結果がオーバーフロー	ON

オペランド

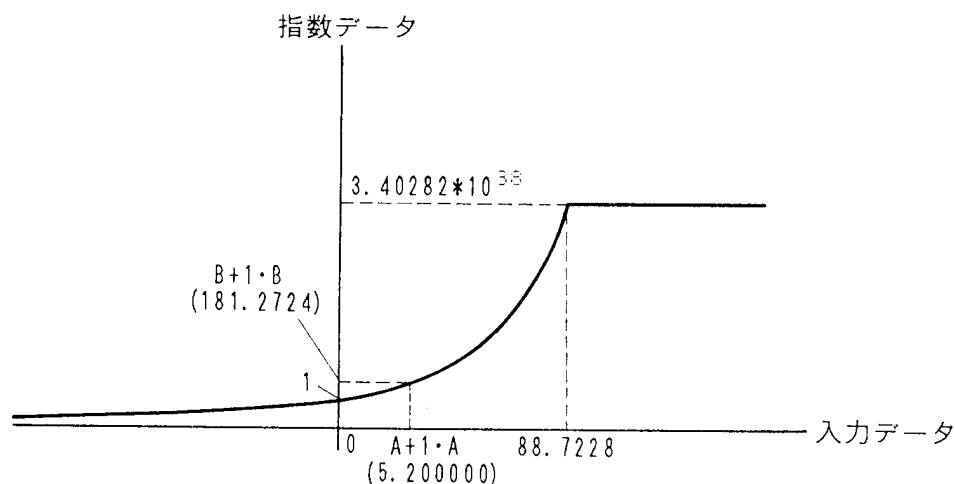
		デバイス										レジスタ										INDEX			定数	修飾			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K	
A+1・A	入力データ											○	○	○	○	○	○	○	○	○	○	○			○	○			○
B+1・B	結果格納先												○	○	○	○	○	○	○	○	○	○		○	○	○			○

プログラム例

R0056 5.200000 181.2724
[R0074-R0073 FEXP D0069-D0068]

動作

- ・ a 接点 R 0 0 5 6 が O N すると、倍長レジスタ R W 0 7 4 ・ R W 0 7 3 の内容 (5 , 2 0 0 0 0) の指数を計算し、結果を倍長レジスタ D 0 0 6 9 ・ D 0 0 6 8 に格納し、オーバーフロー出力を O F F にします。
- ・ a 接点 R 0 0 5 6 が O F F の時は、演算を実行せず、オーバーフロー出力を O F F にします。



小数点常用対数関数

FUN2 3 1 浮動小数点常用対数関数 (FLOG) *

[T 3 / T 3 H のみ使用可能]

浮動小数点データ形式で常用対数関数演算を実行します。

関連命令

条件入力

実行出力

-[A+1・A FLOG B+1・B]-

機能

- オペランド A + 1 ・ A で指定する倍長レジスタに格納されている浮動小数点データ (入力データ) の常用対数を算出し、オペランド B + 1 ・ B で指定する倍長レジスタに格納します。
- 指定可能な入力データの範囲は $-3.40282 \times 10^{38} \sim 3.40282 \times 10^{38}$ です。
- 入力データが 0 (00000000H) の時は、0 (00000000H) を演算結果として格納し、実行出力を ON に設定します。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾	
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J
A+1・A	入力データ											○	○	○	○	○	○	○	○	○	○	○			○	○	
B+1・B	結果格納先												○	○	○	○	○	○	○	○	○		○		○	○	

プログラム例

```

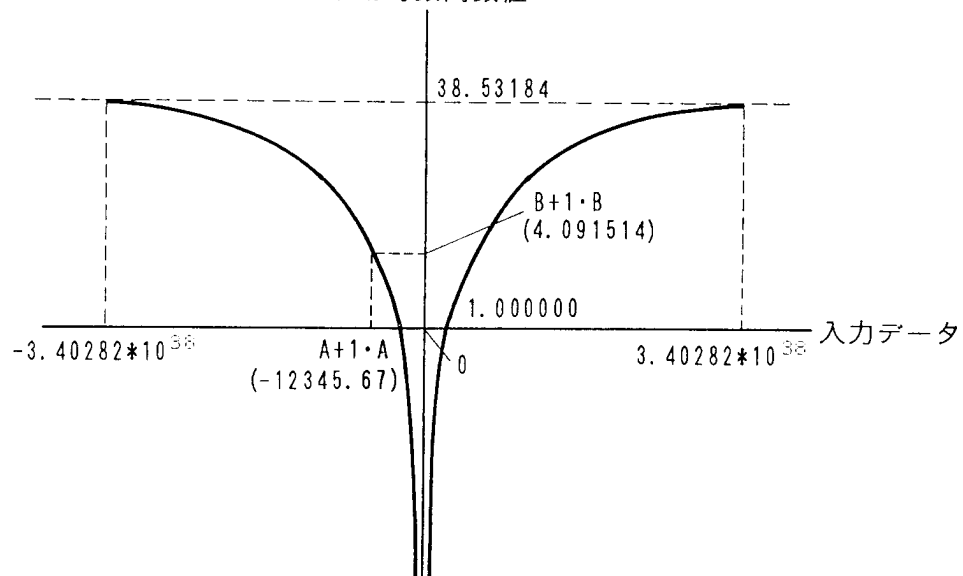
R0057      -12345.67      4.091514
┌──┴──┐ ┌RW048・RW047 FLOG D0129・D0128┐

```

動作

- a 接点 R 0 0 5 7 が ON すると、倍長レジスタ RW 0 4 8 ・ RW 0 4 7 の内容 (- 1 2 3 4 5 . 6 7) の常用対数を計算し、結果をレジスタ D 0 1 2 9 ・ D 0 1 2 8 に格納し、実行出力を ON にします。
- a 接点 R 0 0 5 7 が OFF の時は、演算を実行せず、実行出力を OFF にします。

常用対数関数値



浮動小数点本質継承

FUN 2 3 2 浮動小数点本質継承型 P I D (F P I D 3)

[T 3 H のみ使用可能]

浮動小数点本質継承型 P I D 演算を実行します。

関連命令

条件入力

判定出力

[A+1・A FPID3 B+1・B → C+1・C]

機能

- ・オペランド B+1・B で指定する倍長レジスタを先頭とする 13 個の倍長レジスタに格納されているパラメータに基づいて、オペランド A+1・A で指定する倍長レジスタを先頭とする 6 個の倍長レジスタに格納されている入力データ及び設定値について、本質継承型（不完全微分先行型）P I D 演算を以下のように行い、その結果（操作出力・内部操作量・前回微分項、現在値、設定値等）をオペランド C+1・C で指定する倍長レジスタを先頭とする 8 個のレジスタに格納します。

入力	処 理	出力	ERF
OFF	初期化	OFF	—
ON	演算実行	内部カウンタ C = n	
	演算不実行	内部カウンタ C ≠ n	

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定	修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
A	入力データ先頭											○	○	○	○	○	○	○	○	○	○								○
B	ハラメータ先頭											○	○	○	○	○	○	○	○	○	○								○
C	出力データ先頭												○	○	○	○	○	○	○	○	○								○

桁指定はサポートしません。

データの構成

A レジスタ	
A+1・A	プロセス測定値入力 PVC
	A モード設定値 ASV
	C モード設定値 CSV
	M モード操作入力 MMV
	MV トラッキング入力 TMV
	モード指定 MODE
	A モード:オートモード
	C モード:カスケードモード
	M モード:マニュアルモード

B レジスタ	
B+1・B	比例ゲイン K p
	積分時間 T I
	微分時間 T D
	制御不感帯 G P
	A モード初期値 ISV
	入力フィルタ係数 FT
	SV 変化率リミット DSV
	MMV 変化率リミット DMMV
	初期ステータス STS
	MV 上限リミット MH
	MV 下限リミット ML
	MV 変化率リミット DMV
	制御周期設定値 n

C レジスタ	
C+1・C	操作出力 MV
	前回偏差 e _{n-1}
	前回微分項 D _{n-1}
	前回現在値 PV _{n-1}
	前回設定値 SV _{n-1}
	今回内部操作 MV _n
	内部カウンタ C
	制御周期 Δ t

補足

本命令は、必ず、スキャンモードを定刻スキャンに設定して、メインプログラムでのみ使用して下さい。

[計算式]

(1) デジタルフィルタ部

$$PV_n = (1 - FT) PVC + FT \cdot PV_{n-1} \quad (\text{初期値は、} PV_0 = PV_{0-1} = PVC) \\ (0.000 \leq FT \leq 1.000)$$

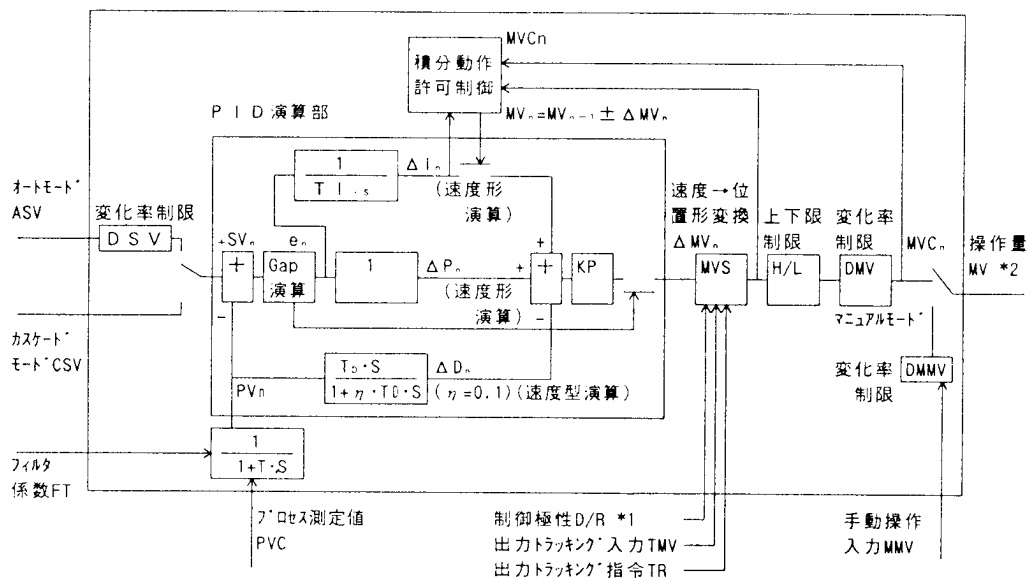
(2) P I D 部

$$\Delta MV_n = Kp (\Delta P_n + \Delta I_n + \Delta D_n) \quad (L_{n-1} \times \Delta I_n > 0 \cdots \text{積分停止}) \\ (|SV_n - PV_n| < |GP| \text{の時、} \Delta MV_n = 0) \\ MV_n = MV_{n-1} \pm \Delta MV_n \quad (\text{ダイレクト指定時：減算、リバース指定時：加算}) \\ (\text{初期値は、} MV_0 = MV) \quad (\text{初期値は、} MV_0 = MV)$$

ここで

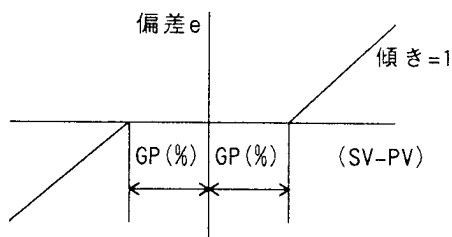
$$\Delta P_n = e_n - e_{n-1} \quad (\text{初期値は、} e_{0-1} = e_0) \\ e_n = SV_n - PV_n \quad (|SV_n - PV_n| < |GP| \text{の時、} e_n = 0) \\ \Delta I_n = \frac{e_n \cdot \Delta t}{T I} \quad (T I = 0 \text{の時は} \Delta I = 0 \text{とします。} (\text{積分停止})) \\ \Delta t = (\text{定刻スキャン時間設定値}) \times n \\ \Delta D_n = \frac{T D (PV_{n-1} - PV_n) - \Delta t \cdot D_{n-1}}{\Delta t + \eta \cdot T D} \\ D_n = D_{n-1} + \Delta D_n \quad (\text{初期値は、} D_0 = D_{0-1} = 0) \\ \eta = 0.1 \quad (\text{固定値}) \\ L_{n-1} = MV_{n-1} - MV \quad (\text{初期値は、} MV_{0-1} = MV)$$

[機能ブロック]



- *1 速度→位置形変換における制御極性D/R指定はダイレクト指定時は減算処理、リバース指定時は加算処理を行います。
- *2 操作量出力MVの出力極性切換は、外部ロジック（ラダー回路）にて行って下さい。

(Gap 演算)



(微分動作制御)
 $|SV_n - PV_n| < |GP|$
 の時 $\Delta MV_n = 0$

(積分動作制御)
 $L_n = MV_n - MV$
 $L_{n-1} \times \Delta I_n > 0 \cdots \text{積分停止}$
 $L_{n-1} \times \Delta I_n \leq 0 \cdots \text{積分許可}$

データ範囲

A レジスタ

A+ 1・A	: フォース測定値入力 PVC [%]	(データ範囲: 0.00~100.00)
A+ 3・A+ 2	: オートモード設定値 ASV [%]	(データ範囲: 0.00~100.00)
A+ 5・A+ 4	: カスケードモード設定値 CSV [%]	(データ範囲: 0.00~100.00)
A+ 7・A+ 6	: マニュアルモード操作量入力 MMV [%]	(データ範囲: -25.00~125.00)
A+ 9・A+ 8	: 操作量トラッキング入力 TMV [%]	(データ範囲: -25.00~125.00)
A+11・A+10	: モード指定 MODE	

31	24	23	16	15	8	7	0	
							TR	MODE

A+11・A+10	MODE: 動作モード指定	TR : トラッキング指令
	<0><1>	0 : なし
	0 0 : マニュアルモード	1 : あり
	0 1 : オートモード	
	1 0 : カスケードモード	
	1 1 : (オートモード)	

B レジスタ

B+ 1・B	: 比例ゲイン K p	(データ範囲: 0.00~327.67)
B+ 3・B+ 2	: 積分時間 T I [min]	(データ範囲: 0.000~32.767)
B+ 5・B+ 4	: 微分時間 T D [min]	(データ範囲: 0.000~32.767)
B+ 7・B+ 6	: 制御不感帯 G P [%]	(データ範囲: 0.00~ 10.00)
B+ 9・B+ 8	: オートモード設定値初期値 ISV [%]	(データ範囲: 0.00~100.00)
B+11・B+10	: フォース入力フィルタ係数 F T	(データ範囲: 0.000~ 1未満)
B+13・B+12	: 設定値ASV変化率リミット DSV [%/Δt]	(データ範囲: 0.00~327.67)
B+15・B+14	: マニュアルモード操作量MMV変化率リミット DMMV [%/Δt]	(データ範囲: 0.00~327.67)
B+17・B+16	: ステータス S T S	

31	24	23	16	15	8	7	0	
							DR	IMODE

B+17・B+16	IMODE: 初期モード指定	DR : 極性D/R指令
	<0><1>	0 : ダイレクト
	0 0 : マニュアルモード	1 : リバース
	0 1 : オートモード	
	1 0 : カスケードモード	
	1 1 : (オートモード)	
B+19・B+18	: 操作量上限リミット M H [%]	(データ範囲: -25.00~125.00)
B+21・B+20	: 操作量下限リミット M L [%]	(データ範囲: -25.00~125.00)
		MH<MLの場合、MHの値はMLの値になります。
B+23・B+22	: 操作量変化率リミット DMV [%/Δt]	(データ範囲: 0.00~327.67)
B+25・B+24	: 制御周期設定値 n	制御周期 Δt=n×(スキャン時間)
		n スキャンごとに P I D 演算を実行する。
		n ≤ 0 のときは、n = 1 として実行する。

1.4.2.2 制御パラメータ設定項目と設定値の範囲

Cレジスタ

C+ 1・C : 操作量 MV [%] (データ範囲:-25.00~125.00)

C+ 3・C+ 2: 前回偏差 e_{n-1}

C+ 5・C+ 4: 前回微分項 D_{n-1}

C+ 7・C+ 6: 前回現在値 PV_{n-1}

C+ 9・C+ 8: 前回設定値 SV_{n-1}

C+11・C+10: 内部操作 MV_n

C+13・C+12: 内部カウンタ C

このカウンタはスキャン回数をカウントするもので、 $C = n$ から毎スキャン、デクリメントします。 $C \leq 0$ になると PID 演算を実行し、再び $C = n$ とします。

C+15・C+14: 制御周期 Δt [S]

制御周期 Δt は、 PC に設定した定刻スキャン時間の n 倍になります。(n は $B+12$ のレジスタで設定)
(PC の定刻スキャン時間は、10~200ms の範囲で 10ms 単位で設定できます。)

[動作説明補足]

- (1) 操作量初期値は、前回までの最終値 MV を出力します。
- (2) MV トラッキング
オートモードとカスケードモードでのみ有効な機能です。(マニュアルモードではこの指定は無効です。)
トラッキング指令 TR が ON の時、操作量 MV の上下限リミットは有効ですが、 MV の変化率制限は無効です。この時、内部 MV_n は $MV_n = MV$ とします。
そして、トラッキング指令が OFF へ切換わる時、その時点の MV_n の値から ΔMV の加算/減算を開始します。
- (3) オートモード
 ASV の変更は、オートモードトマニュアルモードで有効です。
オートモードになった時、 SV_n を変化率制限をしながら ASV に漸近させます。
 $SV_n = SV_{n-1} \pm |DSV|$
また、オートモードからマニュアルモードへの切換えがバンプレスとなるように $MMV = MV$ となります。
よって、オートモード中の MMV の変更は行えません。
操作量 MV の上下限リミット、変化率制限はともに有効です。
- (4) カスケードモード
 CSV の変更はすべてのモードで有効です。
カスケードモードになったとき、 $SV_n = CSV$ となります。この時、操作量 MV の上下限リミット、変化率制限はともに有効です。
また、カスケードモードからオートモード及びマニュアルモードへの切換えがバンプレスとなるように $ASV = SV_n$ 、 $MMV = MV$ となります。
よって、カスケードモード中の ASV 、 MMV の変更は行えません。
- (5) マニュアルモード
 MMV の変更はマニュアルモードでのみ有効です。
マニュアルモードになったとき、 MV を変化率制限しながら MMV に漸近させます。
 $MV = MMV \pm |DMMV|$
操作量 MV の上下限リミット、変化率制限はともに無効です。
また、マニュアルモードからオートモードまたはカスケードモードへの切換えがバンプレスとなるように $MV_n = MV$ となります。
そしてオートモードまたはカスケードモードへの切換え時には、その時点の MV_n の値から ΔMV の加算/減算を開始します。
- (6) 制御極性 D/R 指定
ダイレクト指定では、 PV が増加した場合に MV を増加させます。(ΔMV の減算)
 $MV_n = MV_{n-1} - \Delta MV$
リバース指定では、 PV が増加した場合に MV を減少させます。(ΔMV の加算)
 $MV_n = MV_{n-1} + \Delta MV$

4.2.2.2 制御動作の初期化動作とモード切替動作

(7) マニュアルモード操作量設定値MMV入力の変化率制限 DMMV

MMVをMMVに変化率DMMVで漸近させます。

1. $+|DMMV| \geq (MMV-MV) \geq -|DMMV|$ の場合 : $MV=MMV$
2. $(MMV-MV) > +|DMMV|$ の場合 : $MV=MMV+|DMMV|$
3. $(MMV-MV) < -|DMMV|$ の場合 : $MV=MMV-|DMMV|$

(8) オートモード設定値ASV入力の変化率制限 DSV

SV_nをASVに変化率DSVで漸近させます。

1. $+|DSV| \geq (ASV-SV_{n-1}) \geq -|DSV|$ の場合 : $SV_n=ASV$
2. $(ASV-SV_{n-1}) > +|DSV|$ の場合 : $SV_n=SV_{n-1}+|DSV|$
3. $(ASV-SV_{n-1}) < -|DSV|$ の場合 : $SV_n=SV_{n-1}-|DSV|$

(9) 操作出力の上下制限 MH、ML

1. $MH < ML$ の場合 : $MH=ML$
- 上下制限出力をMVX_nとすると
2. $MV_n > MH$ の場合 : $MVX_n=MH$
3. $MV_n < ML$ の場合 : $MVX_n=ML$

(10) 操作出力の変化率制限 DMV

1. $+|DMV| \geq (MVX_n-MV) \geq -|DMV|$ の場合 : $MV=MVX_n$
2. $(MVX_n-MV) > +|DMV|$ の場合 : $MV=MV+|DMV|$
3. $(MVX_n-MV) < -|DMV|$ の場合 : $MV=MV-|DMV|$

(11) 命令入力OFF時

PIDの初期化動作を行います。内容は次の通りです。

1. 動作モードMODE(A+11・A+10のレジスタ)を初期モードIMODE(B+17・B+16のレジスタ)にて指定されたモードにします。
2. 操作量初期値は前回までの最終値MVを出力します。
3. オートモード設定値をオートモード初期値(B+4のレジスタ)に設定された値にする。
 $ASV=ISV$
4. 内部データを以下のように初期化します。
 $MV_0=MV$
 $MMV=MV$
 $PV_0=PV_C$
 $SV_0=ASV$ または CSV (初期モードに依存。ただしマニュアルモード時はASVとします。)
 $D_0=0$
 $C=n$
 $e_{0-1}=e_0$

(12) 命令入力ON時

PID動作を行います。

設定した制御周期n(B+25・B+24のレジスタ)スキャンごとにPID演算を行います。

(内部カウンタCが0以下なら実行し、新たに内部カウンタCをnに設定し、命令出力をONにします。内部カウンタが0より大きい場合、Cをデクリメントし、命令出力をOFFにします。)

[モードデータ関連表]

		設定値			操作量						
		オートモード設定値 ASV	カスケードモード設定値 CSV	設定値 SV。	マニュアルモード 設定操作量 MMV	トラッキング 設定操作量 TMV	内部操作量 MV。	操作量出力制限		制限出力 MVC。	出力操作量 MV
								上下制限	変化率制限		
マニュアルモード		— (変更可能)	— (変更可能)	SV。 ₋₁ (前回保持率)	設定MMV	— (変更可能)	MV	×	×	MV	MMV-MV。
オートモード	MVトラッキング ="OFF"	設定ASV	— (変更可能)	ASV-SV。	MV (変更不可)	— (変更可能)	PID演算 結果	○	○	内部操作量 MV。 出力制限	MVC。
	MVトラッキング ="ON"					設定TMV	MV	○	×		
カスケードモード	MVトラッキング ="OFF"	SV。 (変更不可)	設定CSV	CSV	MV (変更不可)	— (変更可能)	PID演算 結果	○	○	内部操作量 MV。 出力制限	MVC。
	MVトラッキング ="ON"	設定TMV				MV	○	×			

直接入出力

FUN235 直接入出力 (I/O)*

入出力モジュールと対応レジスタ間でのデータ転送を実行します。

関連命令

条件入力

実行出力

[I/O (n) A]

機能

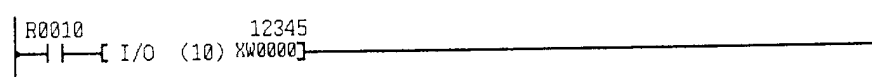
- オペランドAで指定するレジスタを先頭とするnワード分のレジスタと、対応するI/Oモジュールとの間で次のようにデータの入出力を行った後、実行出力をONにします。
 - 対応するI/Oモジュールが入力モジュールの場合：
入力モジュールの内容を対応するレジスタに格納します（データ入力）。
 - 対応するI/Oモジュールが出力モジュールの場合：
レジスタの内容を出力モジュールに転送します（データ出力）。
- 指定可能なテーブルサイズ（オペランドn）の範囲は1～64です。

入力	処 理	出力
OFF	不実行	OFF
ON	実行	ON

オペランド

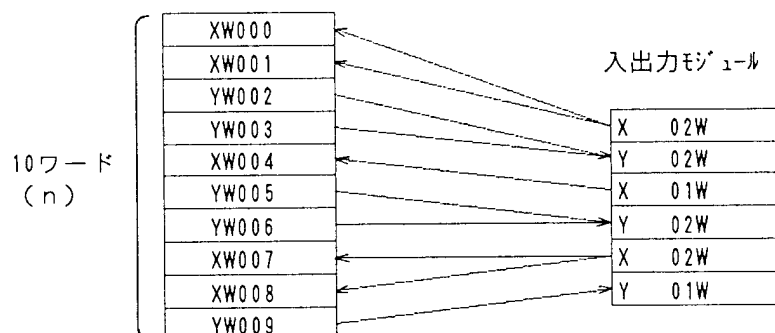
		デバイス										レジスタ										INDEX			定数	修飾			
opr	名称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
n	テーブルサイズ																											○	
A	入出力先頭											○	○															○	

プログラム例



動作

- a接点R0010がONすると、レジスタXW000を先頭とする10ワード分のレジスタと、対応する入出力モジュールとの間でデータ転送を実行し、入力モジュールの内容を対応するレジスタに格納し、また、レジスタの内容を対応する出力モジュールに転送した後、実行出力をONにします。
- a接点R0010がOFFの時は、処理を実行せず、実行出力をOFFにします。



補足

T1においてI/O命令が有効な入出力は次の通りです。

T1-16: X0000～X0007, Y0020～Y0027

T1-28: X0000～X000D, Y0020～Y002D

T1-40: X0000～X0007, Y0020～Y0027, T2 I/Oモジュール

拡張データ転送

FUN 236	拡張データ転送 (XFER)	
連続したレジスタのデータを、他の転送対象別にブロック転送します。		関連命令

入力 —[A XFER B → C]— 出力

機能

入力が ON のとき、A と A+1 で間接指定される転送元と、C と C+1 で間接指定される転送先との間で、ブロック転送が行われます。転送サイズ（ワード数）は B で指定されます。

転送サイズは、1 ~ 256 ワードです。（EEPROM への書き込み時を除く）

転送可能な組み合わせは、以下の通りです。

- 本体レジスタ ↔ 本体レジスタ
- 本体レジスタ ↔ 拡張 F レジスタ（IC メモリカード または T2E/T2N の内部メモリ）
- 本体レジスタ ↔ TOSLINE-S20/S20LP/2000E
- 本体レジスタ ↔ EEPROM（D レジスタ）
- 本体レジスタ ↔ T2E, T2N, T1S 通信ポート

実行条件

入力	動作	出力	ERF
OFF	不実行	OFF	-
ON	正常実行	ON	-
	エラー発生時	ON	ON

オペランド

		デバイス										レジスタ													定数	インデックス		
	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	転送元パラメータ											○	○	○	○	○	○	○	○	○	○							○
B	転送サイズ											○	○	○	○	○	○	○	○	○	○							○
C	転送先パラメータ												○	○	○	○	○	○	○	○	○							○

転送元パラメータ		
A	Bank / CH	種別
A+1	先頭アドレス	

転送サイズ及びステータス		
B	転送ワード数	
B+1	ステータスフラグ	
B+16	(最大 16 ワード)	

転送先パラメータ		
C	Bank / CH	種別
C+1	先頭アドレス	

- 転送対象（種別）については、以降の表を参照下さい。
- ステータスフラグは、TOSLINE-S20/2000E が転送元のとき、または T2E/T2N/T1S の通信ポートが 転送元／転送先のときに付加されます。

転送パラメータ

転送対象		Bank / CH	種別	先頭アドレス	転送ワード数	ステータスフラグ
本体レジスタ	XW/YW レジスタ	0	H00	0 - 511 (T3H) 0 - 255 (T3) 0 - 127 (T2N) 0 - 63 (T2/T2E)	1 - 256 (T3H/T3) 1 - 64 (T2/T2E/T2N)	なし
	W レジスタ	0	H01	0 - 1023 (T3/T2/T2E) 0 - 2047 (T3H/T2N)	1 - 256	なし
	LW レジスタ	0	H02	0 - 255 (T3H/T3/T2/T2E/T2N)	1 - 256	なし
	RW レジスタ	0	H03	0 - 999 (T3H) 0 - 511 (T3) 0 - 255 (T2N/T1S) 0 - 127 (T2/T2E) 0 - 63 (T1)	1 - 256 (T3H/T3) 1 - 128 (T2/T2E/ T2N) 1 - 64 (T1/T1S)	なし
	D レジスタ	0	H04	0 - 8191 (T3H/T3/T2N) 0 - 4095 (T2/T2E/T1S) 0 - 1023 (T1)	1 - 256	なし
	F レジスタ	0	H05	0 - 32767 (T3H) 0 - 8191 (T3) 0 - 1023 (T2/T2E/T2N)	1 - 256	なし
	拡張 F レジスタ (IC メモリカード)	1 - 15 (T3H/T3/T2)	H05	0 - 8191	1 - 256	なし
		1 - 2 (T3H)	H06	0 - 65535 (Bank 1) 0 - 57343 (Bank 2)	1 - 256	なし
	拡張 F レジスタ (内部メモリ)	1 - 3 (T2E/T2N)	H05	0 - 8191	1 - 256	なし
	TOSLINE-S20/S20LP スキャンメモリ	1 - 2 (T3H/T3) 1 (T2/T2E/ T2N)	H10	0 - 4095 (S20LP) 0 - 1023 (S20)	1 - 256	あり ^{*1}
	TOSLINE-2000E スキャンメモリ	1 - 2 (T3H/T3)	H12	0 - 2047	1 - 256	あり
	EEPROM (D レジスタ)	0	H20	0 - 8191 (T3H/T3) 0 - 4095 (T2) 0 - 2047 (T1S/T2E/T2N) 0 - 511 (T1)	読み出し時: 1 - 256 書き込み時: 1 - 128 (T3H) 1 - 64 (T3) 1 - 32 (T1S/T2/ T2E/T2N) 1 - 16 (T1)	なし
	通信ポート (T2E/T2N/T1S)	0	H30	0	1 - 256 ^{*2}	あり ^{*3}

*1) S20 が転送元るときスキャンヘルシーマップが作成される (最大 16 ワード)。S20LP はなし。

*2) 通信ポートが転送元 (受信) のときは 256 ワード固定。

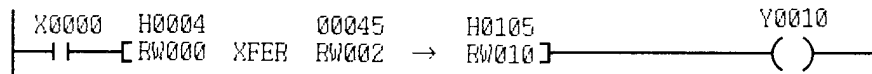
*3) 通信ポートが転送元 (受信) のときは 2 ワードのステータスフラグが付加され、転送先 (送信) のときは 1 ワードのステータスフラグが付加される。

＜通信ポート＞

T2E/T2N の通信ポートとのデータ転送については、

「T2E/T2N 通信機能説明書」 UM-TS02E**-J003 を参照して下さい。

プログラム例 1：本体 D レジスタ→拡張 F レジスタ（IC メモリカード）

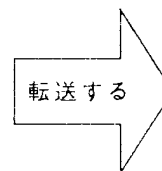


転送元指定ジスタ			転送ワード数	転送先指定レジスタ (ICメモリカード)			
RW000	H00	H04	45ワード転送	RW010	H01	H05	バンク1・
RW001	00000			RW011	00000		レジスタ指定
							レジスタNo.

動作

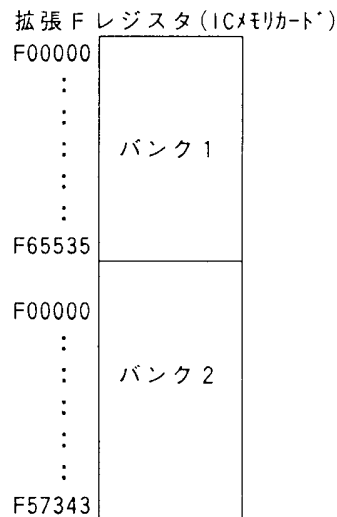
- ・ A 接点 X0000 が ON すると、レジスタ RW000、RW001 で指定された D0000 を先頭としたレジスタ RW002 のデータ 45 ワード分を (D0000～D0044)、レジスタ RW010、RW011 で指定された IC メモリカード内の拡張 F レジスタバンク 1 の F0000 を先頭とする領域へ転送し、出力を ON にします。

転送元 (Dレジスタ)	
D0000	00001
D0001	00002
D0002	00003
D0042	00043
D0043	00044
D0044	00045



ICメモリカード	
バンク1	転送先
F0000	00001
F0001	00002
F0002	00003
F0042	00043
F0043	00044
F0044	00045

《転送種別 06Hの時》



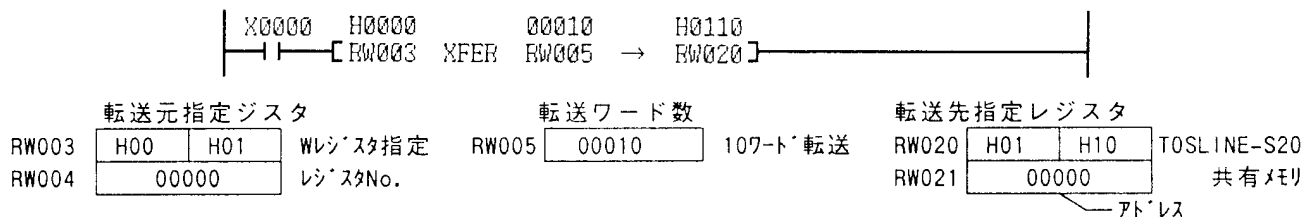
- ・転送種別 0 5 H 指定にてバンク 1 ～ 1 5、または転送種別 0 6 H 指定にて使用する場合は、I/O 割付により C P U の割付部に M M R (ICメモリカード装着) の指定を行って下さい。
- ・T 2 では F レジスタは 1 0 2 4 W (0 ～ 1024) までですが、メモリカード内は 8 1 9 2 W / 1 バンクとなります。
- ・T 3 H では転送種別を 0 6 H としますと ICメモリカードの拡張ファイルレジスタは F00000 ～ F65535 64KW (バンク1)、F00000 ～ F57343 56KW (バンク2) として取り扱えます。

T2E/T2Nにおける構成を下記に示します。

拡張 F レジスタ	
F 0000 . . .	バンク 1
F 8191 F 0000 . . .	バンク 2
F 8191 F 0000 . . .	バンク 3
F 8191	

- ・ T2E / T2NではFレジスタは1024W (0~1024) までですが、内部メモリでは8192W / 1バンクとなります。
- ・ T2E / T2Nでは、拡張Fレジスタ使用の場合でもI / O割付によるCPU割付のMMR指定は必要ありません。
- ・ 拡張Fレジスタは停電保持しません。電源立ち上がり時、データは“0”クリアされます。

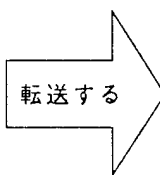
プログラム例 2：本体Wレジスタ→TOSLINE-S20 共有メモリ



動作

- ・ A 接点X0000がONすると、レジスタRW003、RW004で指定されたW0000を先頭としたレジスタRW005のデータ10ワード分を、レジスタRW020、RW021で指定されたTOSLINE-S20の共有メモリ、アドレス0を先頭とする領域へ転送し、出力をONにします。TOSLINE-S20の共有メモリとは、TOSLINE-S20のモジュール内のメモリです。Wレジスタとは別になります。

転送元 (Wレジスタ)		TOSLINE-S20 共有メモリ 転送先	
W0000	00001	0000	00001
W0001	00100	0001	00100
W0002	12345	0002	12345
W0007	03344	0007	03344
W0008	11565	0008	11565
W0009	00045	0009	00045



- 補足
- ・ TOSLINE-S20への転送の際は送信に設定した領域に対して行って下さい。
 - ・ TOSLINE-S20への転送で、転送元のレジスタをWレジスタ以外のレジスタに指定すると、一括入出力処理で、Wレジスタのデータが、TOSLINE-S20のメモリ領域に転送（上書き）されます。
この場合、プログラムの伝送入出力割付の設定部の「LINK」または「GLOBAL」の設定をblankにすると、一括入出力処理は行われません。
 - ・ 転送元がTOSLINE-S20に指定された場合、転送ワード数指定レジスタの後に転送ワード数分のステータスフラグ（ビットマップ）が作成されます。

例) 転送ワード数指定レジスタがRW030で転送ワード数が100の場合、次のようにステータスフラグが作成されます。

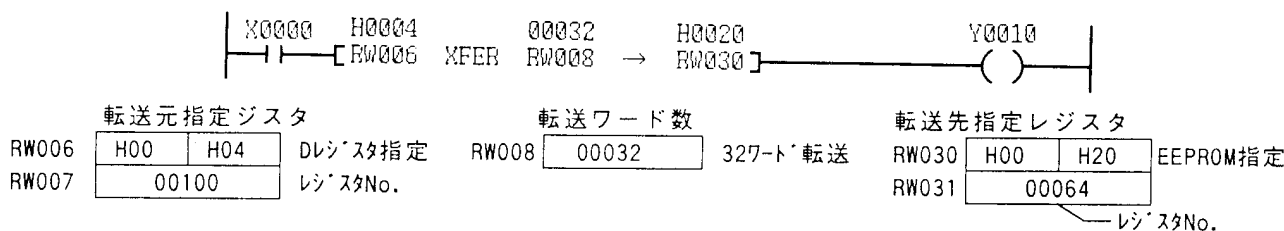
F															0ビット				
RW030	0															← 転送ワード数			
RW031	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	} ステータスフラグ		
RW032	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17			
RW036	96	95	94	93	92	91	90	89	88	87	86	85	84	83	82	81	} ← 転送数分		
RW037													100	99	98	97			

余りビットに0がセットされます。

1 ~ 100 : 対応ワードのヘルシーステータス

- 1 : スキャン伝送正常
- 0 : スキャン伝送異常

プログラム例 3：本体 D レジスタ → E E P R O M (D レジスタ)



動作

- ・ A 接点 X0000 が ON すると、レジスタ RW006、RW007 で指定された D0100 を先頭としたレジスタ RW008 のデータ 32 ワード分 (D0100 ~ D0131) を、レジスタ RW030、RW031 で指定された E E P R O M の D0064 を先頭とする領域へ転送し、出力を ON にします。

転送元 (Dレジスタ)		EEPROM 転送先 (Dレジスタ)	
D0100	00001	D0064	00001
D0101	00002	D0065	00002
D0102	00003	D0066	00003
D0129	00043	D0093	00043
D0130	00044	D0094	00044
D0131	00045	D0095	00045

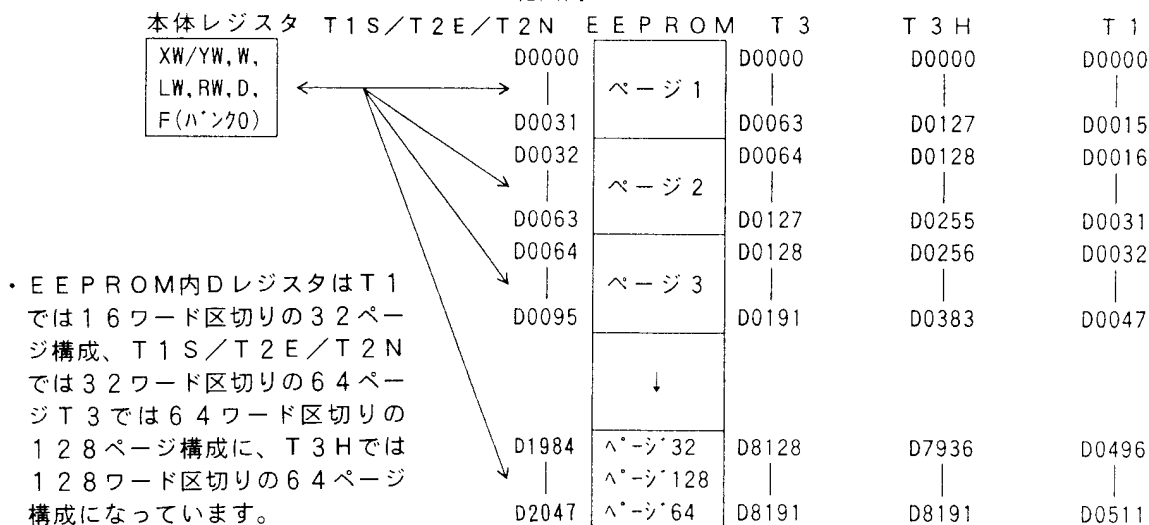
転送する

(補足説明)

E E P R O M (D レジスタ) 構成

読み出しは最大 256 ワード

書き込みは 1 ページの範囲内



E E P R O M へ書き込み動作で留意事項

- ・ E E P R O M への転送 (書き込み) の際の転送ワード数は T1 で 16 ワード、T1S/T2E/T2N で 32 ワード、T3 で 64 ワード、T3H で 128 ワードが最大です。
- ・ E E P R O M への転送 (書き込み) の際、2 ページにまたがる書き込み指定は実行できません。1 回の書き込みは 1 ページ以内の指定にしてください。(例えば、D0120 を先頭に 32 ワードの書き込み指定は実行できません)
- ・ 本命令で E E P R O M にデータを書き込む場合、E E P R O M の書き込み回数のアラーム (特殊リレー S0007) には対応しておりませんのでご留意願います。
- ・ E E P R O M への転送 (書き込み) を行ってから約 10 ms の間、E E P R O M は書き込みモードになり、読み書きができなくなりますので、XFER 命令を 2 回以上実行する場合は最低 10 ms の間隔をおいて下さい。

X F E R 命令では次のときエラーとなります。(S0051セット)
エラー時には転送は実行されませんが、実行出力はO Nになります。

転送対象	エラーの要因
本体レジスタ↔本体レジスタ	・転送ワード数が0もしくは256を越えた時。 ・転送元(先)先頭番号もしくは先頭番号+転送ワード数が転送可能範囲アドレスを越えた時。
本体レジスタ↔拡張Fレジスタ	・転送ワード数が0もしくは256を越えた時。 ・転送元(先)先頭番号もしくは先頭番号+転送ワード数が転送可能範囲アドレスを越えた時。 ・I Cメモ리카ードが装着されていない、もしくはMMRの割付がない時。 ・本体レジスタ→拡張Fレジスタ転送時、I Cメモ리카ードがライトプロテクト状態の時。 ・プログラムが格納されたI Cメモ리카ードを使用した場合。(T3Hでのみ検出)
本体レジスタ↔TOSLINE-S20 (TOSLINE-2000E)	・転送ワード数が0または256を越えた時。 ・転送元(先)先頭番号もしくは先頭番号+転送ワード数が転送可能範囲アドレスを越えた時。 ・C H. No. が1、2以外の時(T2/T2Eでは1以外の時)。 ・TOSLINEが未実装もしくはI/O割付がされていないとき。 ・ステータスエリア作成領域が足りない時。 ・TOSLINE-S20において、スキャンアクセスを2ワードに指定している場合に転送先頭番号が奇数の時。 ・TOSLINE-S20において、スキャンアクセスを2ワードに指定している場合に転送サイズ(ワード数)が奇数の時。 ・TOSLINEモジュールが異常の時。
本体レジスタ↔E E P R O M (Dレジスタ)	・E E P R O Mからの読み出しで転送ワード数が0もしくは256を越えた時。 ・転送元(先)先頭番号もしくは先頭番号+転送ワード数が転送可能範囲アドレス(T1:D0511 T2/T2E:D4091 T2N/T3/T3H:D8191)を越えた時。 ・E E P R O M書き込み時、転送ワード数が0または最大値(T1:16 T2/T2E/T2N:32 T3:64 T3H:128)を越えた時。また、書き込み範囲が2ページにまたがる時。 ・E E P R O Mが書き込みモード中に命令を実行した時。 ・C P UモジュールがE E P R O Mを実装したタイプでない時。
その他	・転送対象に該当のない時。 ・外部装置同士(拡張Fレジスタ・TOSLINE-S20・EEPROM)を転送しようとした時。 ・インデックス修飾による実行時にI、J、Kの値がレジスタ範囲を越えた時(この場合のみ実行出力はO F F)。 ・T1の場合、本体レジスタ同士を転送しようとした時。



特殊モジュールデータ入力

FUN 237 特殊モジュールデータ入力 (READ) *

特殊モジュールがデータを読み込みます。

関連命令

特殊モジュールデータ出力

(WRITE)

条件入力

実行出力

—— [A READ B → C] ——

機能

- 特殊のジュールの拡張メモリからユーザレジスタエリアにデータを転送します。

入力	処 理	出力	ERF
OFF	不実行	OFF	
ON	実行：正常時	ON	
	不実行：エラー時※	ON	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定	修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
A	スロット/レジスタ											○	○															○	○
B	転送A*ラメータ											○	○	○	○	○	○	○	○	○	○	○							
C	転送先 先頭レジスタ												○	○	○	○	○	○	○	○	○	○							○

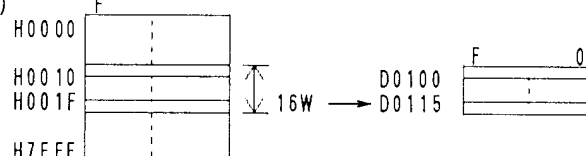
プログラム例

```
R0000
├── [XW000 READ RW010 → D0100] ───┘
```

動作

- a 接点R0000がONの時、XW000に対応する特殊モジュールの拡張メモリエリアのRW010で示されるアドレス(H0010)から、RW011で示されるワード分のデータを、D0100レジスタ以降のテーブルに転送します。(拡張メモリエリアのアドレスH0010から16ワード分)
- RW011に指定できる転送ワード数は最大256ワードです。
- RW011のビット(F)をセットすると、特殊モジュールに対して割り込みを発生します。

拡張メモリエリア(XW000の特殊モジュール)
(ワードアドレス)



RW010

F	0
H0010	----
H0010	----

---- 拡張メモリアドレス
---- 転送ワード数

RW011 (F): 特殊モジュールに対する割り込み要求

補足

- 詳細は使用する特殊モジュールの仕様を参照して下さい。
- T1では、T2特殊モジュールにのみ有効です。

- ※ READ 命令では次の時、エラーとなります。(S0051をセット)
エラー時には転送は実行されません。
1. スロットNo./入出力アドレス指定オペランドが定数またはレジスタ指定以外。
 2. 指定モジュールが活線着脱中(脱中)。
 3. 指定モジュールNOSYNC時。
 4. 指定モジュールが特殊 X' , Y' , $X'+Y'$ 以外。
 5. 転送ワード数が256ワードを越えている。
 6. 拡張メモリアドレスがH80000~HFFFF
 7. 転送元の指定が範囲外(アドレス+サイズがオーバーした。)
 8. 転送先の指定が範囲外(")

特殊モジュールデータ出力

FUN 238 特殊モジュールデータ出力 (WRITE) *

特殊モジュールにデータを出力します。

関連命令

特殊モジュールデータ入力

(READ)

条件入力

実行出力

——[A WRITE B → C]——

機能

- ユーザレジスタの内容を特殊モジュールの拡張メモリエリアにデータを転送します。

入力	処 理	出力	ERF
OFF	不実行	OFF	
ON	実行：正常時	ON	
	不実行：エラー時※	ON	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定	修			
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K	数	飾	
A	転送元 先頭レジスタ											○	○	○	○	○	○	○	○	○	○								○
B	転送パラメータ											○	○	○	○	○	○	○	○	○	○								
C	スロット/レジスタ											○	○															○	○

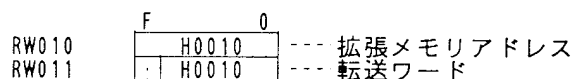
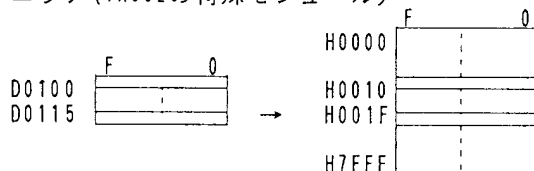
プログラム例

```
R0000
├──┤ D0100 WRITE RW010 → YW002 ───┤
```

動作

- a 接点R0000がONの時、D0100レジスタ以降のテーブルからRW011で示されるワード分のデータをYW002に対応する特殊モジュールのRW010で示される拡張メモリエリアに転送します。(拡張メモリエリアのアドレスH0010から16ワード分)
- RW011に指定できる転送ワード数は最大256ワードです。
- RW011のビット(F)をセットすると、特殊モジュールに対して割り込みを発生します。

拡張メモリエリア (YW002の特殊モジュール)



RW011 (F): 特殊モジュールに対する割り込み要求

補足

- 詳細は使用する特殊モジュールの仕様を参照して下さい。
- T 1 では、T 2 特殊モジュールにのみ有効です。

伝 送 送 信

FUN 2 3 9 伝 送 送 信 (SEND)	
ネットワーク経由で指定したPCのレジスタヘデータを書き込みます。 (TOSLINE-S20LP, イーサネット)	関連命令 R E C V

条件入力 -[A SEND B]- 完了出力

機能

- レジスタAでネットワークと転送先PCを指定し、A+2, A+3, A+4で指定した自PCの転送元レジスタのデータを、A+5, A+6で指定した相手側PCのレジスタに転送します。
- オペランドBに転送完了ステータスが格納されます。
- 転送レジスタサイズは、1～128ワード(S20LP)、1～485ワード(イーサネット)で、これを越えるワード数を指定すると送信パラメータエラーとなります。
(S20LP=TOSLINE-S20LPの略)

入力	処理	出力	ERF
OFF	不実行	OFF	OFF
ON	実行中	OFF	
	正常完了	ON	
	不実行：エラー時	ON	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾			
	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I	J	K			
A	転送パラメータ											○	○	○	○	○	○	○	○	○	○								○
B	完了ステータス												○	○	○	○	○	○	○	○	○								○

桁指定はサポートしてません。

データの構成

転送パラメータ

＜S20LPの場合＞

	F	C	B	8	7	0
A	MID	CH	ステーションNo.			
A+1	要求コード(通常は0)					
A+2	転送レジスタサイズ					
A+3	自局レジスタ種別					
A+4	自局レジスタ先頭 No.					
A+5	相手局レジスタ種別					
A+6	相手局レジスタ先頭 No.					
A+7	応答時間制限					

＜イーサネットの場合＞

	F	C	B	8	7	0
A	MID	CH		0 (未使用)		
A+1	要求コード					
A+2	転送レジスタサイズ					
A+3	自局レジスタ種別					
A+4	自局レジスタ先頭 No.					
A+5	相手局レジスタ種別					
A+6	相手局レジスタ先頭 No.					
A+7	応答時間制限					
A+8	相手局 I P アドレス					
A+9						
A+10	相手局 U D P ポート No.					

イーサネットの場合の転送パラメータはここではレジスタライト要求を例としてレジスタ構成を記載しています。イーサネットに使用する場合の本命令の詳細は別冊の「イーサネット取扱説明書」を参照して下さい。

完了ステータス

	F	E	D	C	B	8	7	0
B	Abn	Busy	Status		0		TermSTS	
B+1	詳細情報 (TermSTS=0BH 以外の場合、データ=0)							

転送パラメータの情報

転送パラメータ	S 2 0 L P	イーサネット
MID (ネットワークタイプ)	2	3
CH (自局チャンネルNo.)	T 2 N T 3 H	1 1 ~ 4 (CPU に近い方から 1, 2, 3, 4 の順)
ステーション No. (相手局 No.)	1 ~ 6 4	0
要求コード	0	2 1 H : レジスタ伝送要求 (他の要求コードについては「イーサネット取扱説明書」を参照)
転送レジスタサイズ(ワード)	1 ~ 1 2 8 ・ T, C レジスタは最大 8 4 ワードです。 ・ T511 と T512 をまたがっての指定は不可です。	1 ~ 4 8 5 ・ T, C レジスタは最大 323 ワードです。 ・ T511 と T512 をまたがっての指定は不可です。
レジスタ種別 (自局/相手局)	H0000: XW/YW レジスタ H0001: W レジスタ H0002: LW レジスタ H0003: RW レジスタ H0004: D レジスタ H0005: F レジスタ H**05: 拡張 F レジスタ (F0000~F8191) (**はバンクNo. T3H : 01 - 0F T2N/T2E : 01 - 03) H**06: 拡張 F レジスタ (T3H のみ) (** はバンクNo. 01 - 02、バンクNo. 01:F0000~F65535 02:F0000~F57343) H0007: T レジスタ H0008: C レジスタ H0009: SW レジスタ	
レジスタ先頭 No. (自局/相手局)	送信データを格納したレジスタの先頭レジスタ No. を指定します。	
応答時間制限	相手局からの応答時間の制限値を設定します。(0~32767, 0.1s 単位) Fビットが 0 の場合は設定した時間に、1 を立てた場合は次のデフォルト値となります。 S20LP 4.1 s Ethernet ... 30 s	
I P アドレス	相手局イーサネットの I P アドレスを指定。	
U D P アドレス	相手局イーサネットの U D P ポート No. を指定	

転送完了ステータス情報:

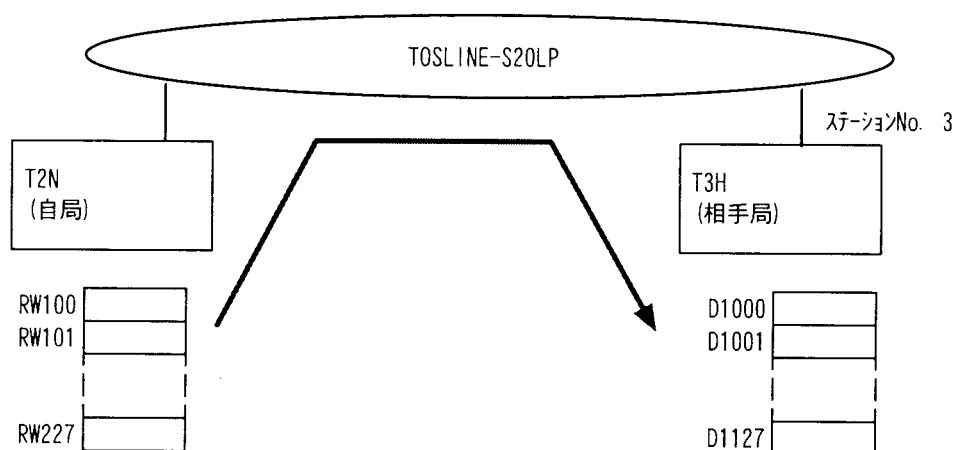
Abn	0: 正常完了 1: 異常完了	
Busy	0: 初期状態 1: 伝送ポートビジー	
Status	0: 初期状態 1: 送信要求中 2: 送信完了及び受信待ち状態 3: 受信完了	
TermSTS	17-時 7ビット目 0: 自局で発生 1: 相手局で発生	
	S 2 0 L P	イーサネット
	00H: 正常完了 01H: レジスタ指定エラー 02H: 応答タイムアウト 03H: パラメータエラー 04H: メモリライトプロテクト 05H: (予約) 06H: モジュールエラー イニシャライズ中 送信完了タイムアウト 07H: 送信CH. なし 08H: ステーションアドレス範囲外 09H: 伝送ワード数異常 0AH: バウンダリーエラー 0BH: 伝送異常 0CH: I/O無応答発生 0DH: ICカード指定異常(T 3 Hのみ) 0EH: 再送タイムアウト 0FH: (予約)	00H: 正常完了 01H: レジスタ指定エラー 02H: 応答タイムアウト 03H: パラメータエラー 04H: メモリライトプロテクト 05H: (予約) 06H: モジュールエラー イニシャライズ中 送信完了タイムアウト 07H: 送信CH. なし 08H: ステーションアドレス範囲外 09H: 伝送ワード数異常 0AH: バウンダリーエラー 0BH: 伝送異常 0CH: I/O無応答発生 0DH: ICカード指定異常(T 3 Hのみ) 0EH: 再送タイムアウト 送信データ容量オーバ 0FH: (予約)
詳細情報	ステータスデータ(TermSTS)が0BHの時、詳細情報が格納されます。 (0BH 以外は0が格納されます。) 詳細情報の内容はT 2 N及びT 3 Hの「TOSLINE-S20LP 取扱説明書」と「イーサネット取扱説明書」を参照して下さい。	

プログラム例



RW010	2	1	3	S20LP, ch. 1, 相手局 No. 3
RW011		0		…要求コード
RW012		128		…送信ワード数 128ワード
RW013		3		…自局 RWレジスタ
RW014		100		…RW100から
RW015		4		…相手局 Dレジスタ
RW016		1000		…D1000から
RW017		10		…応答制限時間 1秒

				送信要求中
RW050	0	0	1	0
RW051				0

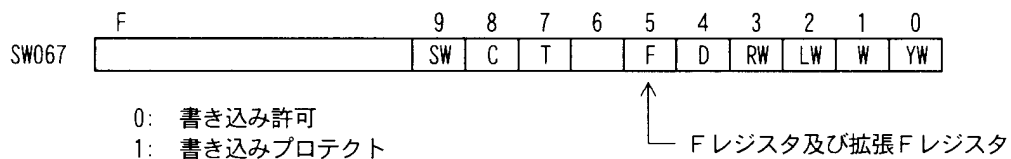


a 接点 R0020 が ON の時、RW100 から 128 ワード分のデータを S20LP 上のステーション NO. 3 の T3H の D1000 以降に転送します。

RW050 のビット D と C は、最初に命令を実行する前に 0 クリアして下さい。(RUN 起動初回のみ)

(補足)

- 命令語の入力は完了出力がONするまでONに保持して下さい。
- SEND 命令では次の場合レジスタ指定エラー (TermSTS=01H) となります。(S0051 を ON)
 - (1) 相手局ステーション No. が最大 No. を越えて設定された場合。(S2OLP時)
 - (2) 相手局ステーション No. に自ステーション No. が設定された場合。(S20LP)
相手先 I Pアドレスに自 I Pアドレスが設定された場合。(イサネット)
 - (3) オペランドが無効レジスタの場合。(T/Cレジスタは同種別間での転送のみが可能です)
 - (4) 送信レジスタサイズに範囲外の値を設定した場合。また、T511 と T512 にまたがって指定した場合。
 - (5) 自局/相手局の指定レジスタが範囲外を指定している場合。
 - (6) メモリプロテクトが指定されていた場合。
 - (7) タイムアウト時間内に応答が受信できなかった時。
 - (8) T3Hの拡張Fレジスタが対象の時、以下のチェックを行い異常であった時。
 - MMR指定されているか
 - ICカードは実装されているか
 - ICカードの種別はプログラムでないか
 - ICカードはライトプロテクトか
- ライトプロテクト
特殊レジスタ SW067 によって相手局からのSEND命令によるレジスタの書き込みに対するプロテクトを行うことができます。



- 転送完了ステータス (オペランドB) レジスタは最初のスキャンでリセット (0クリア) して下さい。

補足

- 本命例は条件入力が「OFF」から「ON」に切替わるタイミングで命令の実行を開始します。
- 本命令は複数スキャンにまたがる処理のため、完了出力が「ON」になるまで、条件入力を「ON」に保持して下さい。
- 命令実行中は伝送パラメータ、転送完了ステータス内のデータを変更しないで下さい。

[REDACTED]

伝 送 受 信

FUN 240 伝 送 受 信 (RECV)	
ネットワーク経由で指定したPCのレジスタヘデータを読み込みます。 (TOSLINE-S20LP, イーサネット)	関連命令 S E N D

条件入力 -[A RECV B]- 完了出力

機能

- レジスタAでネットワークと転送元PCを指定し、A+2, A+3, A+4で指定した自PCの転送元レジスタに、A+5, A+6で指定した相手側PCのレジスタデータを転送します。
- オペランドBに転送完了ステータスが格納されます。
- 転送レジスタサイズは、1～128ワード(S20LP)、1～485ワード(イーサネット)で、これを越えるワード数を指定すると送信パラメータエラーとなります。
(S20LP=TOSLINE-S20LPの略)

入力	処理	出力	ERF
OFF	不実行	OFF	OFF
ON	実行中	OFF	
	正常完了	ON	
	不実行：エラー時	ON	ON

オペランド

		デ バ イ ス										レ ジ ス タ										INDEX			定数	修飾		
	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
A	転送パラメータ											○	○	○	○	○	○	○	○	○	○							○
B	完了ステータス												○	○	○	○	○	○	○	○	○							○

桁指定はサポートしません。

データの構成

転送パラメータ

<S20LPの場合>

	F	C	B	8	7	0
A	MID	CH	ステーションNo.			
A+1	要求コード (通常は 0)					
A+2	転送レジスタサイズ					
A+3	自局レジスタ種別					
A+4	自局レジスタ先頭 No.					
A+5	相手局レジスタ種別					
A+6	相手局レジスタ先頭 No.					
A+7	応答時間制限					

<イーサネットの場合>

	F	C	B	8	7	0
A	MID	CH	0 (未使用)			
A+1	要求コード					
A+2	転送レジスタサイズ					
A+3	自局レジスタ種別					
A+4	自局レジスタ先頭 No.					
A+5	相手局レジスタ種別					
A+6	相手局レジスタ先頭 No.					
A+7	応答時間制限					
A+8	相手局 I P アドレス					
A+9						
A+10	相手局 U D P ポート No.					

イーサネットの場合の転送パラメータはここではレジスタリード要求を例としてレジスタ構成を記載しています。イーサネットに使用する場合の本命令の詳細は別冊の「イーサネット取扱説明書」を参照して下さい。

完了ステータス

	F	E	D	C	B	8	7	0
B	Abn	Busy	Status		0			TermSTS
B+1	詳細情報 (TermSTS=0BH 以外の場合、データ=0)							

転送パラメータの情報

転送パラメータ		S 2 0 L P	イーサネット
MID (ネットワークタイプ)		2	3
CH (自局チャンネルNo.)	T 2 N	1	1
	T 3 H	1、2 (CPU に近い方から 1, 2 の順)	1 ~ 4 (CPU に近い方から 1, 2, 3, 4 の順)
ステーション No. (相手局 No.)		1 ~ 6 4	0
要求コード		0	2 1 H : レジスタ伝送要求 (他の要求コードについては「イーサネット取扱説明書」を参照)
転送レジスタサイズ (ワード)		1 ~ 1 2 8 ・ T, C レジスタは最大 8 4 ワードです。 ・ T511 と T512 をまたがった指定は不可です。	1 ~ 4 8 5 ・ T, C レジスタは最大 323 ワードです。 ・ T511 と T512 をまたがった指定は不可です。
レジスタ種別 (自局/相手局)		H0000: XW/YW レジスタ H0001: W レジスタ H0002: LW レジスタ H0003: RW レジスタ H0004: D レジスタ H0005: F レジスタ H**05: 拡張 F レジスタ (F0000~F8191) (**はバンクNo. T3H: 01 - 0F T2N: 01 - 03) H**06: 拡張 F レジスタ (T3H のみ) (** はバンクNo. 01 - 02、バンクNo. 01: F0000~F65535 02: F0000~F57343) H0007: T レジスタ H0008: C レジスタ H0009: SW レジスタ	
レジスタ先頭 No. (自局/相手局)		送信データを格納したレジスタの先頭レジスタ No. を指定します。	
応答時間制限		相手局からの応答時間の制限値を設定します。(0~32767, 0.1s 単位) フラグが 0 の場合は設定した時間に、1 を立てた場合は次のデフォルト値となります。 S20LP 4.1 s Ethernet ... 30 s	
I P アドレス			相手局イーサネットの I P アドレスを指定。
U D P アドレス			相手局イーサネットの U D P ポート No. を指定

転送完了ステータス情報:

Abn	0: 正常完了 1: 異常完了	
Busy	0: 初期状態 1: 伝送ポートビジー	
Status	0: 初期状態 1: 送信要求中 2: 送信完了及び受信待ち状態 3: 受信完了	
TermSTS	Iラ-時 7ビット目 0: 自局で発生 1: 相手局で発生	
	S 2 0 L P	イーサネット
	00H: 正常完了 01H: レジスタ指定エラー 02H: 応答タイムアウト 03H: パラメータエラー 04H: メモリライトプロテクト 05H: (予約) 06H: モジュールエラー イニシャライズ中 送信完了タイムアウト 07H: 送信 C H. なし 08H: ステーションアドレス範囲外 09H: 伝送ワード数異常 0AH: バウンダリーエラー 0BH: 伝送異常 0CH: I / O 無応答発生 0DH: I C カード指定異常 (T 3 H のみ) 0EH: 再送タイムアウト 0FH: (予約)	00H: 正常完了 01H: レジスタ指定エラー 02H: 応答タイムアウト 03H: パラメータエラー 04H: メモリライトプロテクト 05H: (予約) 06H: モジュールエラー イニシャライズ中 送信完了タイムアウト 07H: 送信 C H. なし 08H: ステーションアドレス範囲外 09H: 伝送ワード数異常 0AH: バウンダリーエラー 0BH: 伝送異常 0CH: I / O 無応答発生 0DH: I C カード指定異常 (T 3 H のみ) 0EH: 再送タイムアウト 送信データ容量オーバ 0FH: (予約)
詳細情報	ステータスデータ (TermSTS) が 0BH の時、詳細情報が格納されます。 (0BH 以外は 0 が格納されます。) 詳細情報の内容は T 2 N 及び T 3 H の「TOSLINE-S20LP 取扱説明書」と「イーサネット取扱説明書」を参照して下さい。	

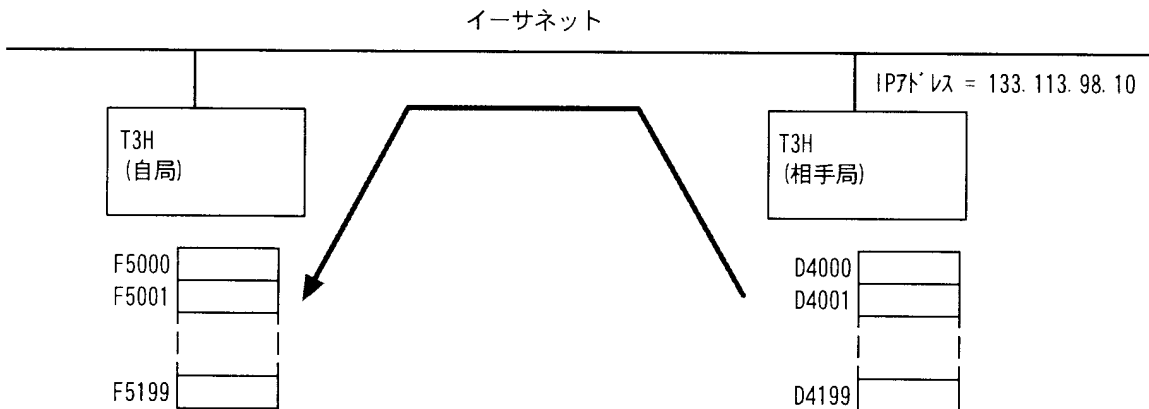
プログラム例



RW030	3	1	0	…イーサネット, c h. 1
RW031	33 (H21)			…要求コード H 2 1 (レジスタリード/ライト)
RW032	200			…受信ワード数 2 0 0ワード
RW033	5			…自局 Fレジスタ
RW034	5000			…F 5 0 0 0から
RW035	4			…相手局 Dレジスタ
RW036	4000			…D 4 0 0 0から
RW037	50			…応答制限時間 5秒
RW038	H71	H85		…相手局 I Pアドレス
RW039	H0A	H62		133. 113. 98. 10 = H85. H71. H62. H0A
RW040	1024			…相手局 U D Pポート No.

受信待ち状態

RW060	0	0	2	0	0
RW061	0				



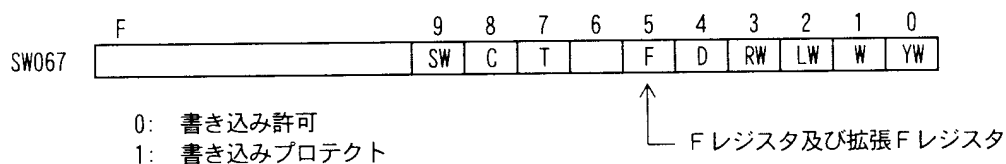
a 接点 R0030 が ON の時、イーサネット上の IPアドレス (133. 113. 98. 10) の T3H の D4000 から 200ワード分のデータを読み込み、T3H (自局側) の F5000 以降に転送します。

転送が完了すると、RW060 の status がセットされ、出力が ON します。

RW050 のビット D と C は、最初に命令を実行する前 0 クリアして下さい。(RUN 起動初回のみ)

(補足)

- 命令語の入力は完了出力がONするまでONに保持して下さい。
- RECV 命令では次の場合レジスタ指定エラー (TermSTS=01H) となります。(S0051 を ON)
 - (1) 相手局ステーション No. が最大 No. を越えて設定された場合。(S2OLP時)
 - (2) 相手局ステーション No. に自ステーション No. が設定された場合。(S2OLP)
相手先IPアドレスに自IPアドレスが設定された場合。(イーサネット)
 - (3) オペランドが無効レジスタの場合。(T/Cレジスタは同種別間での転送のみが可能です)
 - (4) 送信レジスタサイズに範囲外の値を設定した場合。また、T511 と T512 にまたがって指定した場合。
 - (5) 自局/相手局の指定レジスタが範囲外を指定している場合。
 - (6) メモリプロテクトが指定されていた場合。
 - (7) タイムアウト時間内に応答が受信できなかった時。
 - (8) T3Hの拡張Fレジスタが対象の時、以下のチェックを行い異常であった時。
 - MMR指定されているか
 - ICカードは実装されているか
 - ICカードの種別はプログラムでないか
 - ICカードはライトプロテクトか
- ライトプロテクト
特殊レジスタ SW067 によって自局のレジスタへの書き込みに対するプロテクトを行うことができます。



- 転送完了ステータス (オペランドB) レジスタは最初のスキャンでリセット(0クリア)して下さい。

補足

- 本命令は条件入力「OFF」から「ON」に切替わるタイミングで命令の実行を開始します。
- 本命令は複数スキャンにまたがる処理のため、完了出力「ON」になるまで、条件入力を「ON」に保持して下さい。
- 命令実行中は伝送パラメータ、転送完了ステータス内のデータを変更しないで下さい。

[REDACTED]

S F C イ ニ シ ャ ラ イ ズ

FNC 2 4 1 S F C イ ニ シ ャ ラ イ ズ (S F I Z)

S F C プ ロ グ ラ ム を 起 動 し ま す 。

関 連 命 令

条 件 入 力

実 行 出 力

[SFIZ(n) A]

機 能

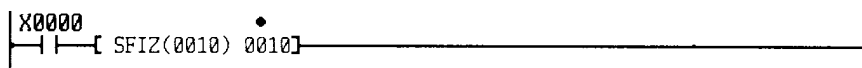
- S F C プ ロ グ ラ ム 開 始 前 に 本 命 令 を 使 用 し ま す 。
- 入 力 が O F F か ら O N に 変 化 し た 時 、 オ ペ ラ ン ド A で 設 定 す る 先 頭 ス テ ッ プ を O N に 設 定 し 、 オ ペ ラ ン ド n で 指 定 す る ス テ ッ プ サ イ ズ 内 の そ れ 以 降 の ス テ ッ プ を O F F に 設 定 し ま す 。
- オ ペ ラ ン ド n の デ ー タ 範 囲 は 1 ~ 4 0 9 6 で す 。
- オ ペ ラ ン ド A の デ ー タ 範 囲 は 0 ~ 4 0 9 5 で す 。

入 力	処 理	出 力
O F F	不 実 行	O F F
O N	O F F か ら O N に 立 ち 上 が り 時 実 行	O N

オ ペ ラ ン ド

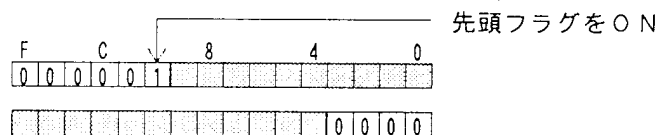
		デ バ イ ス										レ ジ ス タ										INDEX			定 数	修 飾		
opr	名 称	X	Y	S	L	R	Z	T	C	I	O	XW	YW	SW	LW	RW	W	T	C	D	F	IW	OW	I			J	K
n	ステップ*サイズ*																											○
A	先頭ステップ*																											○

プ ロ グ ラ ム 例



機 能

- a 接 点 X 0 0 0 0 が O N と な る と 、 そ の 立 ち 上 が り で ス テ ッ プ 先 頭 フ ラ グ 1 0 を O N に 設 定 し 、 そ れ 以 降 1 9 ま で 9 個 の フ ラ グ を す べ て O F F に し ま す 。
- a 接 点 X 0 0 0 0 が O F F な ら ば 初 期 設 定 は 行 い ま せ ン 。



部 は 不 変

S F C イニシャルステップ

S F Cプログラムの開始シンボルです。

関連命令

S F Cイニシャライズ (S F C)

S F Cイニシャライズ (L D)

A



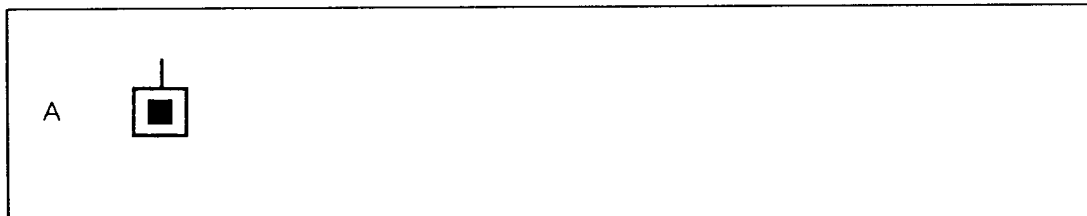
機能

- S F Cプログラムの開始を宣言するシンボルです。
- ラダー回路でS F Cイニシャライズ命令 (S F I Z) が実行されるか、またはS F CシンボルのS F Cイニシャライズ実行により起動します。
- S F Cプログラムの先頭（あるいはS F Cイニシャライズの次ステップ）に必ず配置してください。
- S F Cイニシャルステップにもラダープログラムの入力ができます。

オペランド

オペランドA: ステップNO.
(0～4095)

S F C エンドステップ	
S F C 終了のシンボルです。	関連命令

**機能**

- S F C プログラムの終了を示します。
- このステップが実行される（ひとつ上の遷移条件が成立する）と、実行ステップをインisialステップに移します。

オペランド

オペランド A ステップ No.
 (0 ~ 4095)
(インisialステップと同一ステップ No.)

補足

エンドステップのステップ No. は同一 S F C プログラムのインisialステップのステップ No. と一致させて下さい。

S F C ステップ

プログラムを実行するステップです。

A



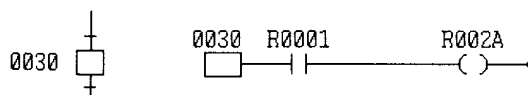
機能

- 1ブロックのプログラムを格納する機能ブロックです。
- このステップが実行される（ひとつ上の遷移条件が成立する）と、格納されているラダープログラムを実行します。
- ステップ内には11行×11列のラダー回路または1コイルの実行命令が格納できます。
- ステップとステップは連続して配置できません。ステップとステップの間には、必ず遷移条件（分岐条件）を配置してください。

オペランド

オペランドA ステップNO.
(0～4095)

プログラム例



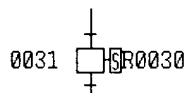
動作

ステップ30が実行状態になると、ラダープログラムを実行します。

R0001がONのときR002AがONになります。

ステップ30が不実行状態のときは、R0001の状態に関わらず、R002AはOFFになります。

プログラム例



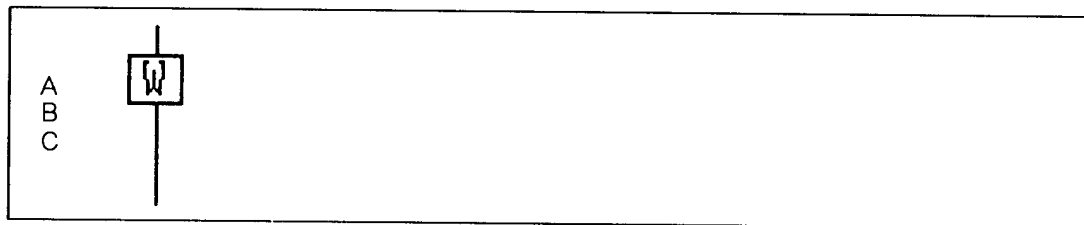
動作

ステップ31がONのとき、R0030をONにします。

S F C 待ち時間付きステップ

設定時間の間、実行を保持するステップです。

関連命令
ステップ



機能

- このステップの下の変移条件が成立しても、設定時間の間、ステップの実行状態を保ちます。
- 設定時間が経過すると（このステップの下の変移条件が成立しているとき）、次のステップに移ります。
- ステップには最大11行×11列のラダー回路、または1コイルの実行命令が格納できます。

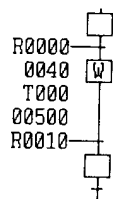
オペランド

オペランドA ステップNo.
(0~4095)

オペランドB タイマNo.
(0~511)

オペランドC 設定時間
(タイマT0~T63×10mS)
(T64~T511×100mS)

プログラム例



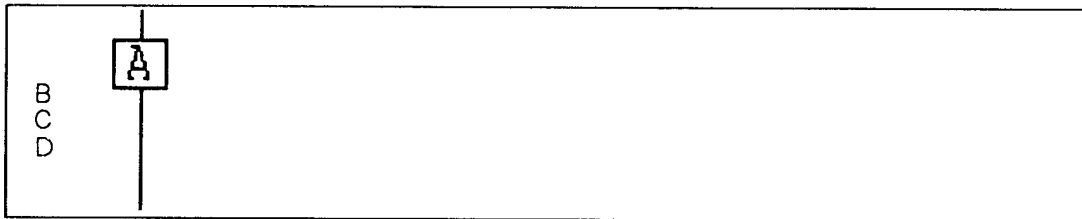
動作

R0010がONになっても、ステップ40は実行開始後5秒が経過するまで、実行状態を保持します。

S F C 渋滞検出付きステップ

シーケンスが設定時間以上渋滞するとアラームを発するステップです。

関連命令



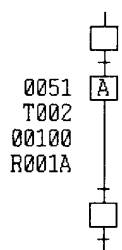
機能

- ステップが実行状態になってから、設定時間以上経過しても、次のステップに移移しないと、指定したデバイス（アラームデバイス）をONにします。
- ステップには最大11行×11列のラダー回路、または1コイルの実行命令を格納できます。

オペランド

- オペランドA ステップNo.
(0~4095)
- オペランドB タイマNo.
(0~511)
- オペランドC 監視時間
(タイマT0~T63×10mS)
(T64~T511×100mS)
- オペランドD アラームデバイス

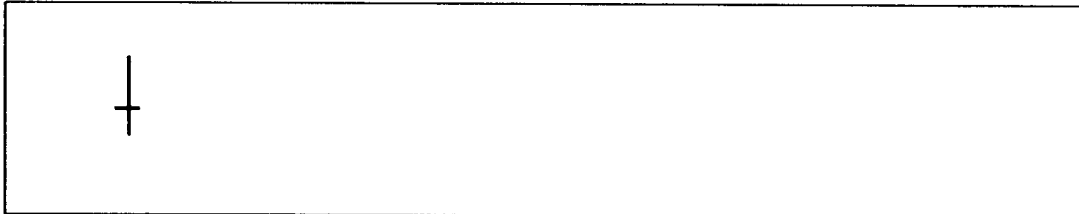
プログラム例



動作

ステップ51の実行時間が10秒以上になるとR001AをONにします。

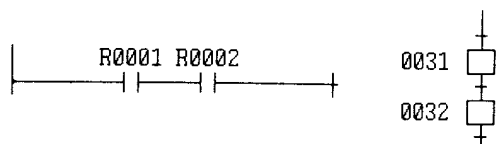
S F C 遷移	
ステップからステップへの移行を行うステップです。	関連命令



機能

- ステップからステップへの遷移条件を定義し条件が成立すれば、実行ステップを移行します。
- 遷移条件には、最大11行×10列のラダー回路、または1条件接点が記述できます。
- 遷移シンボルを連続して配置することはできません。1遷移シンボルに遷移条件が記述しきれないときは、遷移シンボル間にダミーステップを置いて、遷移条件を記述してください。

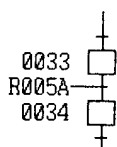
プログラム例



動作

ステップ31実行中に、R0001とR0002がONになると、ステップ31の実行を中止し、ステップ32を実行します。

プログラム例



動作

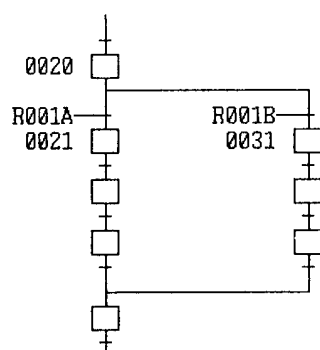
ステップ33実行中に、R005AがONになると、ステップ33の実行を中止し、ステップ34を実行します。

S F C 条件分岐	
条件にしたがってシーケンスを分岐、実行します。	関連コマンド

機能

- 各分岐回路の先頭におかれた遷移条件に従って S F C の流れを選択、実行します。
- 最大 5 回路の分岐回路を記述できます。
- 条件分岐で実行される分岐回路は 1 回路のみです。先に分岐条件の成立した回路が実行されます。同時に複数の分岐条件が成立したときは、左側に記述されている回路が選択、実行されます。
- 条件分岐線の上には必ずステップを配置してください。
- 分岐合流線の上には必ず遷移シンボルを配置してください。
- 条件分岐線により分岐した個数と、対応する合流線に合流する個数は等しくなくてはなりません。

プログラム例



動作

ステップ 2 0 を実行中に R001A が O N になるとステップ 2 1 を実行し、R001B が O N になるとステップ 3 1 を実行します。
R001A、R001B が同時に O N になったときは、ステップ 2 1 を実行します。

S F C 並列実行

複数の分岐した回路を同時に実行します。

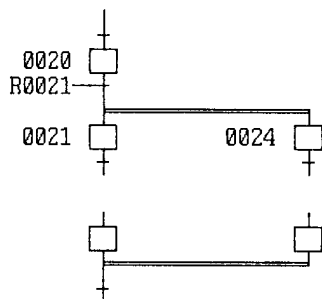
関連命令

=====

機能

- 並列分岐線に続く複数の分岐回路を同時に実行します。
- 最大5回路の分岐回路を記述できます。
- 並列実行は、すべての分岐回路の実行完了により終了します（最後の分岐回路の実行が完了するまで、並列実行移行のステップには移行しません）。
- 並列分岐線の上には必ず遷移シンボルを配置してください。
- 並列分岐線の下には必ずステップを配置してください。
- 並列合流線の上には必ず遷移シンボルを配置してください。
- 並列合流線の下には必ず遷移シンボルを配置してください。
- 並列分岐線により分岐した個数と、対応する合流線に合流する分岐の個数は等しくなくてはなりません。

プログラム例



動作

ステップ20を実行中、R0021がONになるとステップ21、ステップ24で始まる分岐回路を並列実行します。

S F C ジャンプ / ラベル

指定したステップにジャンプします。

関連命令

A



A



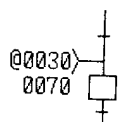
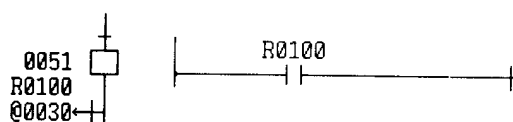
機能

- ジャンプ命令に記述されたジャンプ条件が成立すると、オペランドAで示すラベル命令にジャンプし、以降プログラムを実行します。

オペランド

オペランドA ラベルNO.
(0～1023)

プログラム例

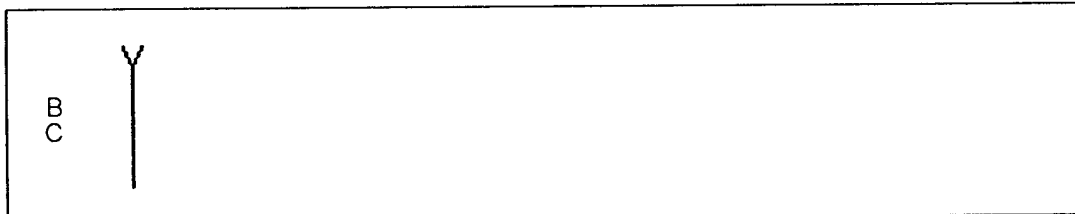


動作

ステップ51を実行中に、R0100がONになると、ラベル30にジャンプし、ステップ70を実行します。

S F C

S F C	S F Cイニシャライズ	
S F Cプログラムだけでプログラムを構成するときの、プログラム起動シンボルです。		関連命令



機能

- S F Cプログラムだけでプログラムを構成するときに使用します。
- オペランドBのデバイスがONになると、オペランドCのステップ個数のフラグを初期化し、S F Cイニシャライズに続くイニシャルステップを起動します。（ラダー回路のS F I Zと同じ機能をもつシンボルです）
- S F Cイニシャライズはプログラムの先頭に配置してください。

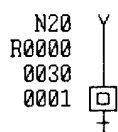
オペランド

オペランドA プログラムN0.
 オペランドB デバイスN0.
 オペランドC ステップ個数

補足

必ずこの命令の後はS F Cイニシャルステップとして下さい。他の命令はセットできません。

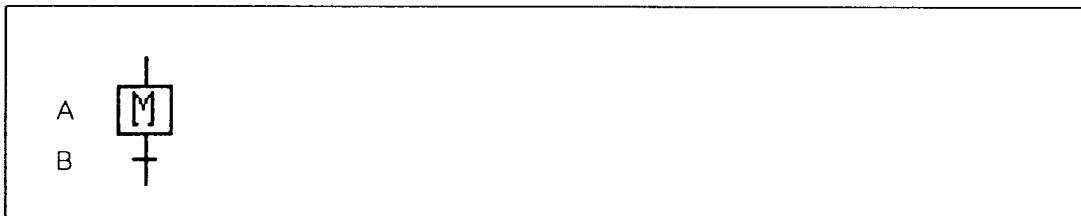
プログラム例



動作

R0000がONになると、イニシャルステップ(00001)を起動します。

S F C マクロステップ	
マクロプログラムにジャンプします。	関連命令 マクロラベル



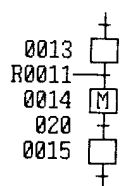
機能

- オペランド B で示すマクロプログラム（サブルーチンプログラム）に処理を移します。
- マクロプログラムのエンド命令（マクロエンド）実行により、マクロステップに復帰します。

オペランド

オペランド A	ステップ N0.
	(0 ~ 4095)
オペランド B	マクロ N0.

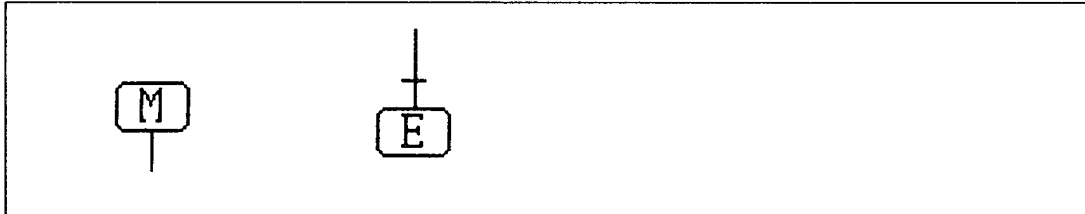
プログラム例



動作

R0011 が ON になるとマクロ N0. 20 のマクロプログラムに処理を移します。

S F C マクロラベル / マクロエンド	
マクロプログラムのスタート／エンドシンボルです。	関連命令 マクロステップ

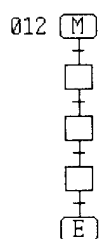


機能

- マクロプログラムの開始／終了位置を示します。
- マクロステップによりコールされ、マクロラベルから始まるプログラムを実行します。
- マクロエンド実行により、コールされたマクロステップに復帰します。

オペランド（マクロラベル）
オペランド A マクロ N0.
 (0～127)

プログラム例



動作

マクロ N0. 12 のオペランドを持つマクロステップが実行されると、マクロラベルに処理が移り、マクロプログラムを実行します。

S F C エンド

プログラムの終了を示します。

関連命令

A



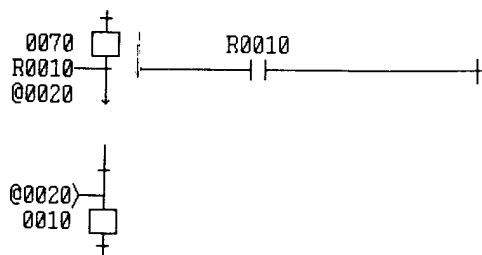
機能

- プログラムの終了を示します。
- 終了条件が成立すると、オペランド A で示すラベルに戻り、以降プログラムを実行します。

オペランド

オペランド A ラベル NO.
(0～1023)

プログラム例



動作

ステップ70を実行中に、R0010がONになるとラベル20に戻り、ステップ10を実行します。

さくいん（命令語）

数字／アルファベット	記号	FUN番号	ページ
1ビット左シフト	SHL 1	6 9	1 1 8
1ビット左ローテート	RTL 1	7 9	1 3 2
1ビット左ローテート（キャリー付き）	RLC 1	8 5	1 4 0
1ビット右シフト	SHR 1	6 8	1 1 7
1ビット右ローテート	RTR 1	7 8	1 3 1
1ビット右ローテート（キャリー付き）	RRC 1	8 4	1 3 9
2の補数	NEG	1 8 2	2 5 2
7セグメントデコード	7 SEG	1 8 5	2 5 5
ASCII変換	ASC	1 8 6	2 5 7
ASCII→HEX変換	ATOH	6 3	1 1 2
BCD加算	B+	1 9 2	2 6 2
BCD加算（キャリー付き）	B+C	2 0 0	2 7 0
BCD減算	B-	1 9 3	2 6 3
BCD減算（キャリー付き）	B-C	2 0 1	2 7 1
BCD乗算	B*	1 9 4	2 6 4
BCD除算	B/	1 9 5	2 6 5
BCD変換	BCD	1 9 0	2 6 0
HEX→ASCII変換	HTOA	6 2	1 1 1
OFF時微分接点			4 6
ON時微分接点			4 5
SFCイニシャライズ	SFIZ	2 4 1	3 3 4
SFCイニシャライズ（SFC）			3 4 4
a接点			4 3
b接点			4 4
mビットファイルnビット左シフト	TSHL	7 3	1 2 5
mビットファイルnビット右シフト	TSHR	7 2	1 2 3
mビットファイルnビット左ローテート	TRTL	8 3	1 3 7
mビットファイルnビット左ローテート（キャリー付き）	TRL C	8 9	1 4 5
mビットファイルnビット右ローテート	TRTR	8 2	1 3 5
mビットファイルnビット右ローテート（キャリー付き）	TRRC	8 8	1 4 3
nビット左シフト	SHL	7 1	1 2 1
nビット左ローテート	RTL	8 1	1 3 4
nビット左ローテート（キャリー付き）	RLC	8 7	1 4 2
nビット右シフト	SHR	7 0	1 1 9
nビット右ローテート	RTR	8 0	1 3 3
nビット右ローテート（キャリー付き）	RRC	8 6	1 4 1
PID（速度型）	PID	1 7 0	2 3 8
PID（偏差2乗型）	PID 2	1 7 1	2 4 0
PID（浮動小数点速度型）	FPID	2 2 2	2 9 2
PID（浮動小数点偏差2乗型）	FPID 2	2 2 3	2 9 4
PID（本質継承型）	PID 3	1 5 6	2 1 4
PID（浮動小数点本質継承型）	FPID 3	2 3 2	3 0 4
あ／ア			
アップダウンカウンタ	U/D	1 4 9	2 0 4
イニシャルステップ（SFC）			3 3 4
移動平均	MAVE	5 6	1 0 4

	記 号	FUN番号	ページ
インクリメント	+ 1	4 3	9 2
インクリメント (倍長)	D + 1	4 4	9 3
インバータ			4 9
インバートコイル			5 0
ウォッチドッグタイマリセット	W D T	1 4 3	1 9 8
エンコード	E N C	1 2 0	1 8 0
エンド (S F C)			3 4 7
エンドステップ (S F C)			3 3 6
オフディレイタイマ	T O F		5 8
オンディレイタイマ	T O N		5 7

か / カ

開平	R T	1 6 7	2 3 2
開平 (浮動小数点)	F R T	2 2 1	2 9 1
カウンタ	C N T		6 0
カウンタ (アップ/ダウン)	U / D	1 4 9	2 0 4
拡張転送命令	X F E R	2 3 6	3 1 0
下限リミット	L L	1 6 1	2 2 5
下限リミット (浮動小数点)	F L L	2 1 9	2 8 9
加算	+	2 7	7 3
加算 (B C D)	B +	1 9 2	2 6 2
加算 (キャリー付き)	+ C	3 5	8 1
加算 (倍長)	D +	3 1	7 7
加算 (倍長 B C D)	D B +	1 9 6	2 6 6
加算 (浮動小数点)	F +	2 0 8	2 7 8
カムシーケンサ	C A M	1 5 9	2 2 2
カレンダー演算	C L D S	1 5 5	2 1 3
カレンダー設定	C L D N	1 5 4	2 1 2
関数発生器	F G	1 6 5	2 2 9
逆正弦関数	A S I N	1 7 5	2 4 5
逆正弦関数 (浮動小数点)	F A S I N	2 2 7	2 9 9
逆正接関数	A T A N	1 7 7	2 4 7
逆正接関数 (浮動小数点)	F A T A N	2 2 9	3 0 1
逆余弦関数	A C O S	1 7 6	2 4 6
逆余弦関数 (浮動小数点)	F A C O S	2 2 8	3 0 0
キャリーセット	S E T C	1 1 8	1 7 8
キャリーリセット	R S T C	1 1 9	1 7 9
キャリー付き1ビット左ローテート	R L C 1	8 5	1 4 0
キャリー付き1ビット右ローテート	R R C 1	8 4	1 3 9
キャリー付きB C D加算	B + C	2 0 0	2 7 0
キャリー付きB C D減算	B - C	2 0 1	2 7 1
キャリー付きmビットファイルnビット左ローテート	T R L C	8 9	1 4 5
キャリー付きmビットファイルnビット右ローテート	T R R C	8 8	1 4 3
キャリー付きnビット左ローテート	R L C	8 7	1 4 2
キャリー付きnビット右ローテート	R R C	8 6	1 4 1
キャリー付き加算	+ C	3 5	8 1
キャリー付き減算	- C	3 6	8 3
キャリー付き倍長B C D加算	D B + C	2 0 2	2 7 2
キャリー付き倍長B C D減算	D B - C	2 0 3	2 7 3

	記 号	FUN番号	ページ
キャリー付き倍長加算	D + C	3 7	8 5
キャリー付き倍長減算	D - C	3 8	8 6
繰り返し F O R	F O R	1 3 2	1 9 2
繰り返し N E X T	N E X T	1 3 3	1 9 2
減算	-	2 8	7 4
減算 (B C D)	B -	1 9 3	2 6 3
減算 (キャリー付き)	- C	3 6	8 3
減算 (倍長)	D -	3 2	7 8
減算 (倍長 B C D)	D B -	1 9 7	2 6 7
減算 (浮動小数点)	F -	2 0 9	2 7 9
コイル			4 7
固定小数点変換	F I X	2 0 5	2 7 5

さ / サ

最小値	M I N	1 6 3	2 2 7
最大値	M A X	1 6 2	2 2 6
サブルーチンエントリー	S U B R	1 3 7	1 9 4
サブルーチンコール	C A L L	1 2 8	1 8 9
サブルーチンリターン	R E T	1 2 9	1 9 0
指数関数	E X P	1 7 8	2 4 8
指数関数 (浮動小数点)	F E X P	2 3 0	3 0 2
シフト (1 ビット左)	S H L 1	6 9	1 1 8
シフト (1 ビット右)	S H R 1	6 8	1 1 7
シフト (n ビット左)	S H L	7 1	1 2 1
シフト (n ビット右)	S H R	7 0	1 1 9
シフト (m ビットファイル n ビット左)	T S H L	7 3	1 2 5
シフト (m ビットファイル n ビット右)	T S H R	7 2	1 2 3
シフトレジスタ	S R	7 4	1 2 7
ジャンプ (S F C)			3 4 3
ジャンプコントロールセット	J C S		5 5
ジャンプコントロールリセット	J C R		5 5
ジャンプラベル	L B L	1 3 6	1 8 5
渋滞検出付きステップ (S F C)			3 3 9
終了	E N D		5 6
上限リミット	U L	1 6 0	2 2 4
上限リミット (浮動小数点)	F U L	2 1 8	2 8 8
乗算	*	2 9	7 5
乗算 (B C D)	B *	1 9 4	2 6 4
乗算 (倍長)	D *	3 3	7 9
乗算 (倍長 B C D)	D B *	1 9 8	2 6 8
乗算 (符号無し)	U *	3 9	8 7
乗算 (浮動小数点)	F *	2 1 0	2 8 0
条件ジャンプ	J U M P	1 3 0	1 9 1
条件分岐 (S F C)			3 4 1
常用対数関数	L O G	1 7 9	2 4 9
常用対数関数 (浮動小数点)	F L O G	2 3 1	3 0 3
除算	/	3 0	7 6
除算 (B C D)	B /	1 9 5	2 6 5
除算 (倍長)	D /	3 4	8 0

	記 号	FUN番号	ページ
除算（倍長 B C D）	D B /	1 9 9	2 6 9
除算（符号無し）	U /	4 0	8 8
除算（浮動小数点）	F /	2 1 1	2 8 1
シングルショット	S S		5 9
診断表示	D I A G	1 5 0	2 0 5
診断表示リセット	D I A R	1 5 1	2 0 7
ステータスラッチセット	S T L S	1 5 2	2 0 9
ステータスラッチリセット	S T L R	1 5 3	2 1 1
ステップ（S F C）			3 3 7
ステップシーケンスイニシャライズ	S T I Z	1 4 4	1 9 9
ステップシーケンス出力	S T O T	1 4 6	2 0 1
ステップシーケンス入力	S T I N	1 4 5	2 0 0
正弦関数	S I N	1 7 2	2 4 2
正弦関数（浮動小数点）	F S I N	2 2 4	2 9 6
正接関数	T A N	1 7 4	2 4 4
正接関数（浮動小数点）	F T A N	2 2 6	2 9 8
積分	I N T G	1 6 8	2 3 3
絶対値	A B S	1 8 0	2 5 0
遷移（S F C）			3 4 0
双方向シフトレジスタ	D S R	7 5	1 2 8
速度型 P I D	P I D	1 7 0	2 3 8
速度型 P I D（浮動小数点）	F P I D	2 2 2	2 9 2

た／タ

タイマ（オフディレイ）	T O F		5 8
タイマ（オンディレイ）	T O N		5 7
タイマトリガ	T R G	1 4 8	2 0 3
ダブルワード変換	D W	1 8 4	2 5 4
直接入出力	I / O	2 3 5	3 0 9
デジタルファイル	D F L	6 1	1 1 0
データサーチ	S C H	1 2 4	1 8 4
データビットカウンタ	B C	1 2 2	1 8 2
データ交換	X C H G	2 2	6 6
データ交換（倍長）	D X C H	2 3	6 7
データ転送	M O V	1 8	6 2
データ転送（倍長）	D M O V	1 9	6 3
データ否定転送	N O T	2 0	6 4
データ否定転送（倍長）	D N O T	2 1	6 5
テーブルビットセット	T S E T	1 1 6	1 7 4
テーブルビットリセット	T R S T	1 1 7	1 7 6
テーブル初期化	T I N Z	2 4	6 8
テーブル転送	T M O V	2 5	6 9
テーブル排他的論理和	T E O R	5 9	1 0 8
テーブル否定排他的論理和	T E N R	6 0	1 0 9
テーブル否定転送	T N O T	2 6	7 1
テーブル論理積	T A N D	5 7	1 0 6
テーブル論理和	T O R	5 8	1 0 7
テーブル→ビット転送	T B M	9 2	1 4 9
デクリメント	- 1	4 5	9 4

	記 号	FUN番号	ページ
デクリメント (倍長) —————	D - 1	4 6	9 5
デコード —————	D E C	1 2 1	1 8 1
テスト —————	T E S T	6 4	1 1 4
テスト (倍長) —————	D T S T	6 5	1 1 5
デッドバンド —————	D B	1 6 6	2 3 1
デッドバンド (浮動小数点) —————	F D B	2 2 0	2 9 0
デバイス / レジスタセット —————	S E T	1 1 4	1 7 0
デバイス / レジスタリセット —————	R S T	1 1 5	1 7 2
デバイスシフト —————	S F T	7 6	1 3 0
デマルチプレクサ —————	D P X	9 1	1 4 8
伝送 受信 —————	R E C V	2 4 0	3 2 8
伝送 送信 —————	S E N D	2 3 9	3 2 2
特殊モジュールデータ出力 —————	W R I T E	2 3 8	3 2 0
特殊モジュールデータ入力 —————	R E A D	2 3 7	3 1 8
ドラムシーケンサ —————	D R U M	1 5 8	2 2 0

な / ナ

ネガティブパルスコイル —————		5 4
ネガティブパルス接点 —————		5 2

は / ハ

排他的論理和 —————	E O R	5 2	1 0 0
排他的論理和 (テーブル) —————	T E O R	5 9	1 0 6
排他的論理和 (倍長) —————	D E O R	5 3	1 0 1
倍長 2 の補数 —————	D N E G	1 8 3	2 5 3
倍長 B C D 加算 —————	D B +	1 9 6	2 6 6
倍長 B C D 加算 (キャリー付き) —————	D B + C	2 0 2	2 7 2
倍長 B C D 減算 —————	D B -	1 9 7	2 6 7
倍長 B C D 減算 (キャリー付き) —————	D B - C	2 0 3	2 7 3
倍長 B C D 乗算 —————	D B *	1 9 8	2 6 8
倍長 B C D 除算 —————	D B /	1 9 9	2 6 9
倍長 B C D 変換 —————	D B C D	1 9 1	2 6 1
倍長乗除算 —————	D * /	4 2	9 0
倍長より大きい —————	D >	1 0 2	1 5 8
倍長より大きいまたは等しい —————	D > =	1 0 3	1 5 9
倍長より小さい —————	D <	1 0 6	1 6 2
倍長より小さいまたは等しい —————	D < =	1 0 7	1 6 3
倍長インクリメント —————	D + 1	4 4	9 3
倍長データ交換 —————	D X C H	2 3	6 7
倍長データ転送 —————	D M O V	1 9	6 3
倍長データ否定転送 —————	D N O T	2 1	6 5
倍長デクリメント —————	D - 1	4 6	9 5
倍長テスト —————	D T S T	6 5	1 1 5
倍長バイナリ変換 —————	D B I N	1 8 9	2 5 9
倍長ビットカウンタ —————	D B C	1 2 3	1 8 3
倍長加算 —————	D +	3 1	7 7
倍長減算 —————	D -	3 2	7 8
倍長乗算 —————	D *	3 3	7 9
倍長除算 —————	D /	3 4	8 0

	記 号	FUN番号	ページ
倍長乗除算	D * /	4 2	9 0
倍長絶対値	D A B S	1 8 1	2 5 1
倍長排他的論理和	D E O R	5 3	1 0 1
倍長否定排他的論理和	D E N R	5 5	1 0 3
倍長等しい	D =	1 0 4	1 6 0
倍長等しくない	D < >	1 0 5	1 6 1
倍長論理積	D A N D	4 9	9 7
倍長論理和	D O R	5 1	9 9
バイナリ変換	B I N	1 8 8	2 5 8
左シフト (1ビット)	S H L 1	6 9	1 1 8
左シフト (mビットファイルnビット)	T S H L	7 3	1 2 5
左シフト (nビット)	S H L	7 1	1 2 1
左ローテート (1ビット)	R T L 1	7 9	1 3 2
左ローテート (mビットファイルnビット)	T R T L	8 3	1 3 7
左ローテート (nビット)	R T L	8 1	1 3 4
否定排他的論理和	E N R	5 4	1 0 2
否定排他的論理和 (テーブル)	T E N R	6 0	1 0 7
否定排他的論理和 (倍長)	D E N R	5 5	1 0 3
ビットファイル比較	T C M P	9 5	1 5 1
ビットファイル・ビットテスト	T T S T	6 6	1 1 6
ビット→テーブル転送	B T M	9 3	1 5 0
等しい	=	9 8	1 5 4
等しい (倍長)	D =	1 0 4	1 6 0
等しい (符号無し)	U =	1 1 0	1 6 6
等しい (浮動小数点)	F =	2 1 4	2 8 4
等しくない	< >	9 9	1 5 5
等しくない (倍長)	D < >	1 0 5	1 6 1
等しくない (符号無し)	U < >	1 1 1	1 6 7
等しくない (浮動小数点)	F < >	2 1 5	2 8 5
フォーストコイル			4 8
プッシュ	P U S H	1 2 5	1 8 5
浮動小数点より大きい	F >	2 1 2	2 8 2
浮動小数点より大きいまたは等しい	F > =	2 1 3	2 8 3
浮動小数点より小さい	F <	2 1 6	2 8 6
浮動小数点より小さいまたは等しい	F < =	2 1 7	2 8 7
浮動小数点デッドバンド	F D B	2 2 0	2 9 0
浮動小数点・下限リミット	F L L	2 1 9	2 8 9
浮動小数点・上限リミット	F U L	2 1 8	2 8 8
浮動小数点加算	F +	2 0 8	2 7 8
浮動小数点开平	F R T	2 2 1	2 9 1
浮動小数点逆正弦関数	F A S I N	2 2 7	2 9 9
浮動小数点逆正接関数	F A T A N	2 2 9	3 0 1
浮動小数点逆余弦関数	F A C O S	2 2 8	3 0 0
浮動小数点減算	F -	2 0 9	2 7 9
浮動小数点指数関数	F E X P	2 3 0	3 0 2
浮動小数点常用対数関数	F L O G	2 3 1	3 0 3
浮動小数点乗算	F *	2 1 0	2 8 0
浮動小数点除算	F /	2 1 1	2 8 1
浮動小数点正弦関数	F S I N	2 2 4	2 9 6

	記 号	FUN番号	ページ
浮動小数点正接関数	F T A N	2 2 6	2 9 8
浮動小数点絶対値	F A B S	2 0 6	2 7 6
浮動小数点速度型 P I D	F P I D	2 2 2	2 9 2
浮動小数点等しい	F =	2 1 4	2 8 4
浮動小数点等しくない	F < >	2 1 5	2 8 5
浮動小数点符号反転	F N E G	2 0 7	2 7 7
浮動小数点偏差 2 乗型 P I D	F P I D 2	2 2 3	2 9 4
浮動小数点変換	F L T	2 0 4	2 7 4
浮動小数点本質継承型 P I D	F P I D 3	2 3 2	3 0 4
浮動小数点余弦関数	F C O S	2 2 5	2 9 7
符号無ししより大きい	U >	1 0 8	1 6 4
符号無ししより大きいまたは等しい	U > =	1 0 9	1 6 5
符号無ししより小さい	U <	1 1 2	1 6 8
符号無ししより小さいまたは等しい	U < =	1 1 3	1 6 9
符号無し乗算	U *	3 9	8 7
符号無し除算	U /	4 0	8 8
符号無し倍長単長除算	D I V	4 1	8 9
符号無し等しい	U =	1 1 0	1 6 6
符号無し等しくない	U < >	1 1 1	1 6 7
フリップフロップ	F / F	1 4 7	2 0 2
平均値	A V E	1 6 4	2 2 8
平方根 (開平)	R T	1 6 7	2 3 2
平方根 (浮動小数点开平)	F R T	2 2 1	2 9 1
並列実行 (S F C)			3 4 2
偏差 2 乗型 P I D	P I D 2	1 7 1	2 4 0
偏差 2 乗型 P I D (浮動小数点)	F P I D 2	2 2 3	2 9 4
ポジティブパルスコイル			5 3
ポジティブパルス接点			5 1
ポップファースト	P O P F	1 2 7	1 8 8
ポップラスト	P O P L	1 2 6	1 8 7
本質継承型 P I D	P I D 3	1 5 6	2 1 4

ま / マ

マクロエンド (S F C)			3 4 6
マクロステップ (S F C)			3 4 5
マクロラベル (S F C)			3 4 6
マスターコントロールセット	M C S		6 1
マスターコントロールセット n	M C S n	1 3 4	1 9 3
マスターコントロールリセット	M C R		6 1
マスターコントロールリセット n	M C R n	1 3 5	1 9 3
マルチプレクサ	M P X	9 0	1 4 7
右シフト (1 ビット)	S H R 1	6 8	1 1 7
右シフト (m ビットファイル n ビット)	T S H R	7 2	1 2 3
右シフト (n ビット)	S H R	7 0	1 1 9
右ローテート (1 ビット)	R T R 1	7 8	1 3 1
右ローテート (m ビットファイル n ビット)	T R T R	8 2	1 3 5
右ローテート (n ビット)	R T R	8 0	1 3 3

や／ヤ

	記 号	FUN番号	ページ
余弦関数	C O S	1 7 3	2 4 3
余弦関数 (浮動小数点)	F C O S	2 2 5	2 9 7
より大きい	>	9 6	1 5 6
より大きい (倍長)	D >	1 0 2	1 5 8
より大きい (符号無し)	U >	1 0 8	1 6 4
より大きい (浮動小数点)	F >	2 1 2	2 8 2
より大きいまたは等しい	> =	9 7	1 5 3
より大きいまたは等しい (倍長)	D > =	1 0 3	1 5 9
より大きいまたは等しい (符号無し)	U > =	1 0 9	1 6 5
より大きいまたは等しい (浮動小数点)	F > =	2 1 3	2 8 3
より小さい	<	1 0 0	1 5 6
より小さい (倍長)	D <	1 0 6	1 6 2
より小さい (符号無し)	U <	1 1 2	1 6 8
より小さい (浮動小数点)	F <	2 1 6	2 8 6
より小さいまたは等しい	< =	1 0 1	1 5 7
より小さいまたは等しい (倍長)	D < =	1 0 7	1 6 3
より小さいまたは等しい (符号無し)	U < =	1 1 3	1 6 9
より小さいまたは等しい (浮動小数点)	F < =	2 1 7	2 8 7

ら／ラ

ラベル (S F C)			3 4 3
ランプ関数	R A N P	1 6 9	2 3 5
ルート	R T	1 6 7	2 3 2
ルート (浮動小数点)	F R T	2 2 1	2 9 1
ローテート (1 ビット左)	R T L 1	7 9	1 3 2
ローテート (1 ビット右)	R T R 1	7 8	1 3 1
ローテート (n ビット左)	R T L	8 1	1 3 4
ローテート (n ビット右)	R T R	8 0	1 3 3
ローテート (m ビットファイル n ビット左)	T R T L	8 3	1 3 7
ローテート (m ビットファイル n ビット右)	T R T R	8 2	1 3 5
論理積	A N D	4 8	9 6
論理積 (テーブル)	T A N D	5 7	1 0 6
論理積 (倍長)	D A N D	4 9	9 7
論理否定 (データ否定転送)	N O T	2 0	6 4
論理否定 (倍長データ否定転送)	D N O T	2 1	6 5
論理否定 (テーブル否定転送)	T N O T	2 6	7 1
論理和	O R	5 0	9 8
論理和 (テーブル)	T O R	5 8	1 0 7
論理和 (倍長)	D O R	5 1	9 9

わ／ワ

割り込みプログラム許可	E I	1 4 0	1 9 5
割り込みプログラム禁止	D I	1 4 1	1 9 6
割り込みプログラム終了	I R E T	1 4 2	1 9 7

さく い ん (記号)

	命令語	FUN記号	ページ
*	乗算	2 9	7 5
+	加算	2 7	7 3
+ 1	インクリメント	4 3	9 2
+ C	キャリー付き加算	3 5	8 1
-	減算	2 8	7 4
- 1	デクリメント	4 5	9 4
- C	キャリー付き減算	3 6	8 3
/	除算	3 0	7 6
7 S E G	7セグメントデコード	1 8 5	2 5 5
<	より小さい	1 0 0	1 5 6
< =	より小さいまたは等しい	1 0 1	1 5 7
< >	等しくない	9 9	1 5 5
=	等しい	9 8	1 5 4
>	より大きい	9 6	1 5 2
> =	より大きいまたは等しい	9 7	1 5 3
A B S	絶対値	1 8 0	2 5 0
A C O S	逆余弦関数	1 7 6	2 4 6
A N D	論理積	4 8	9 6
A S C	A S C I I 変換	1 8 6	2 5 7
A S I N	逆正弦関数	1 7 5	2 4 5
A T A N	逆正接関数	1 7 7	2 4 7
A T O H	A S C I I → H E X 変換	6 3	1 1 2
A V E	平均値	1 6 4	2 2 8
B *	B C D 乗算	1 9 4	2 6 4
B +	B C D 加算	1 9 2	2 6 2
B + C	キャリー付き B C D 加算	2 0 0	2 7 0
B -	B C D 減算	1 9 3	2 6 3
B - C	キャリー付き B C D 減算	2 0 1	2 7 1
B /	B C D 除算	1 9 5	2 6 5
B C	データビットカウンタ	1 2 2	1 8 2
B C D	B C D 変換	1 9 0	2 6 0
B I N	バイナリ変換	1 8 8	2 5 8
B T M	ビット→テーブル転送	9 3	1 5 0
C A L L	サブルーチンコール	1 2 8	1 8 9
C A M	カムシーケンサ	1 5 9	2 2 2
C L D S	カレンダー演算	1 5 5	2 1 3
C L N D	カレンダー設定	1 5 4	2 1 2
C N T	カウンタ		6 0
C O S	余弦関数	1 7 3	2 4 3
D *	倍長乗算	3 3	7 9
D * /	倍長乗除算	4 2	9 0
D +	倍長加算	3 1	7 7
D + 1	倍長インクリメント	4 4	9 3
D + C	キャリー付き倍長加算	3 7	8 5
D -	倍長減算	3 2	7 8
D - 1	倍長デクリメント	4 6	9 5

	命令語	FUN記号	ページ
D - C	キャリー付き倍長減算	3 8	8 6
D /	倍長除算	3 4	8 0
D <	倍長より小さい	1 0 6	1 6 2
D < =	倍長より小さいまたは等しい	1 0 7	1 6 3
D < >	倍長等しくない	1 0 5	1 6 1
D =	倍長等しい	1 0 4	1 6 0
D >	倍長より大きい	1 0 2	1 5 8
D > =	倍長より大きいまたは等しい	1 0 3	1 5 9
D A B S	倍長絶対値	1 8 1	2 5 1
D A N D	倍長論理積	4 9	9 7
D B	デッドバンド	1 6 6	2 3 1
D B *	倍長 B C D 乗算	1 9 8	2 6 8
D B +	倍長 B C D 加算	1 9 6	2 6 6
D B + C	キャリー付き倍長 B C D 加算	2 0 2	2 7 2
D B -	倍長 B C D 減算	1 9 7	2 6 7
D B - C	キャリー付き倍長 B C D 減算	2 0 3	2 7 3
D B /	倍長 B C D 除算	1 9 9	2 6 9
D B C	倍長ビットカウンタ	1 2 3	1 8 3
D B C D	倍長 B C D 変換	1 9 1	2 6 1
D B I N	倍長バイナリ変換	1 8 9	2 5 9
D E C	デコード	1 2 1	1 8 1
D E N R	倍長否定排他的論理和	5 5	1 0 3
D E O R	倍長排他的論理和	5 3	1 0 1
D F L	ディジタルフィルタ	6 1	1 1 0
D I	割り込みプログラム禁止	1 4 1	1 9 6
D I A G	診断表示	1 5 0	2 0 5
D I A R	診断表示リセット	1 5 1	2 0 7
D I V	符号無し倍長単長除算	4 1	8 9
D M O V	倍長データ転送	1 9	6 3
D N E G	倍長 2 の補数	1 8 3	2 5 3
D N O T	倍長データ否定転送	2 1	6 5
D O R	倍長論理和	5 1	9 9
D P X	デマルチプレクサ	9 1	1 4 8
D R U M	ドラムシーケンサ	1 5 8	2 2 0
D S R	双方向シフトレジスタ	7 5	1 2 8
D T S T	倍長テスト	6 5	1 1 5
D W	ダブルワード変換	1 8 4	2 5 4
D X C H	倍長データ交換	2 3	6 7
E I	割り込みプログラム許可	1 4 0	1 9 5
E N C	エンコード	1 2 0	1 8 0
E N D	終了		5 6
E N R	否定排他的論理和	5 4	1 0 2
E O R	排他的論理和	5 2	1 0 0
E X P	指数関数	1 7 8	2 4 8
F *	浮動小数点乗算	2 1 0	2 8 0
F +	浮動小数点加算	2 0 8	2 7 8
F -	浮動小数点減算	2 0 9	2 7 9
F /	浮動小数点除算	2 1 1	2 8 1
F / F	フリップフロップ	1 4 7	2 0 2

	命令語	FUN記号	ページ
F < _____	浮動小数点より小さい	2 1 6	2 8 6
F < = _____	浮動小数点より小さいまたは等しい	2 1 7	2 8 7
F < > _____	浮動小数点等しくない	2 1 5	2 8 5
F = _____	浮動小数点等しい	2 1 4	2 8 4
F > _____	浮動小数点より大きい	2 1 2	2 8 2
F > = _____	浮動小数点より大きいまたは等しい	2 1 3	2 8 3
F A B S _____	浮動小数点絶対値	2 0 6	2 7 6
F A C O S _____	浮動小数点逆余弦関数	2 2 8	3 0 0
F A S I N _____	浮動小数点逆正弦関数	2 2 7	2 9 9
F A T A N _____	浮動小数点逆正接関数	2 2 9	3 0 1
F C O S _____	浮動小数点余弦関数	2 2 5	2 9 7
F D B _____	浮動小数点デッドバンド	2 2 0	2 9 0
F E X P _____	浮動小数点指数関数	2 3 0	3 0 2
F G _____	関数発生器	1 6 5	2 2 9
F I X _____	固定小数点変換	2 0 5	2 7 5
F L L _____	浮動小数点・下限リミット	2 1 9	2 8 9
F L O G _____	浮動小数点常用対数関数	2 3 1	3 0 3
F L T _____	浮動小数点変換	2 0 4	2 7 4
F N E G _____	浮動小数点符号反転	2 0 7	2 7 7
F O R _____	繰り返し F O R	1 3 2	1 9 2
F P I D _____	浮動小数点速度型 P I D	2 2 2	2 9 2
F P I D 2 _____	浮動小数点偏差 2 乗型 P I D	2 2 3	2 9 4
F P I D 3 _____	浮動小数点本質継承型 P I D	2 3 2	3 0 4
F R T _____	浮動小数点開平	2 2 1	2 9 1
F S I N _____	浮動小数点正弦関数	2 2 4	2 9 6
F T A N _____	浮動小数点正接関数	2 2 6	2 9 8
F U L _____	浮動小数点・上限リミット	2 1 8	2 8 8
H T O A _____	H E X → A S C I I 変換	6 2	1 1 1
I / O _____	直接入出力	2 3 5	3 0 9
I N T G _____	積分	1 6 8	2 3 3
I R E T _____	割り込みプログラム終了	1 4 2	1 9 7
J C R _____	ジャンプコントロールリセット		5 5
J C S _____	ジャンプコントロールセット		5 5
J U M P _____	条件ジャンプ	1 3 0	1 9 1
L B L _____	ジャンプラベル	1 3 6	1 9 1
L L _____	下限リミット	1 6 1	2 2 5
L O G _____	常用対数関数	1 7 9	2 4 9
M A V E _____	移動平均	5 6	1 0 4
M A X _____	最大値	1 6 2	2 2 6
M C R _____	マスターコントロールリセット		6 1
M C R n _____	マスターコントロールリセット n	1 3 5	1 9 3
M C S _____	マスターコントロールセット		6 1
M C S n _____	マスターコントロールセット	1 3 4	1 9 3
M I N _____	最小値	1 6 3	2 2 7
M O V _____	データ転送	1 8	6 2
M P X _____	マルチプレクサ	9 0	1 4 7
N E G _____	2 の補数	1 8 2	2 5 2
N E X T _____	繰り返し N E X T	1 3 3	1 9 2
N O T _____	データ否定転送	2 0	6 4

	命令語	FUN記号	ページ
OR	論理和	50	98
PID	速度型PID	170	238
PID2	偏差2乗型PID	171	240
PID3	本質継承型PID	156	214
POPF	ポップファースト	127	188
POPL	ポップラスト	126	187
PUSH	プッシュ	125	185
RAMP	ランプ関数	169	235
READ	特殊モジュールデータ入力	237	318
RECV	伝送 受信	240	328
RET	サブルーチンリターン	129	190
RLC	キャリー付きnビット左ローテート	87	142
RLC1	キャリー付き1ビット左ローテート	85	140
RRC	キャリー付きnビット右ローテート	86	141
RRC1	キャリー付き1ビット右ローテート	84	139
RST	デバイス/レジスタリセット	115	172
RSTC	キャリーリセット	119	179
RT	開平	167	232
RTL	nビット左ローテート	81	134
RTL1	1ビット左ローテート	79	132
RTR	nビット右ローテート	80	133
RTR1	1ビット右ローテート	78	131
SCH	データサーチ	124	184
SEND	伝送 送信	239	322
SET	デバイス/レジスタセット	114	170
SETC	キャリーセット	118	178
SFIZ	SFCイニシャライズ	241	334
SFT	デバイスシフト	76	130
SHL	nビット左シフト	71	121
SHL1	1ビット左シフト	69	118
SHR	nビット右シフト	70	119
SHR1	1ビット右シフト	68	117
SIN	正弦関数	172	242
SR	シフトレジスタ	74	127
SS	シングルショット		59
STIN	ステップシーケンス入力	145	200
STIZ	ステップシーケンスイニシャライズ	144	199
STLS	ステータスラッチセット	152	209
STLR	ステータスラッチリセット	153	211
STOT	ステップシーケンス出力	146	201
SUBR	サブルーチンエントリー	137	194
TAN	正接関数	174	244
TAND	テーブル論理和	57	106
TBM	テーブル→ビット転送	92	149
TCMP	ビットファイル比較	95	151
TENR	テーブル否定排他的論理和	60	109
TEOR	テーブル排他的論理和	59	108
TEST	テスト	64	114
TINZ	テーブル初期化	24	68

	命令語	FUN記号	ページ
TMOV	テーブル転送	25	69
TNOT	テーブル否定転送	26	71
TOF	オフディレイタイマ		58
TON	オンディレイタイマ		57
TOR	テーブル論理和	58	107
TRG	タイマトリガ	148	203
TRLC	キャリー付きmビットファイル nビット左ローテート	89	145
TRRC	キャリー付きmビットファイル nビット右ローテート	88	143
TRST	テーブルビットリセット	117	176
TRTL	mビットファイル nビット左ローテート	83	137
TRTR	mビットファイル nビット右ローテート	82	135
TSET	テーブルビットセット	116	174
TSHL	mビットファイルnビット左シフト	73	125
TSHR	mビットファイルnビット右シフト	72	123
TTST	ビットファイル・ビットテスト	66	116
U*	符号無し乗算	39	87
U/	符号無し除算	40	88
U/D	アップダウンカウンタ	149	204
U<	符号無しより小さい	112	168
U<=	符号無しより小さいまたは等しい	113	169
U<>	符号無し等しくない	111	167
U=	符号無し等しい	110	166
U>	符号無しより大きい	108	164
U>=	符号無しより大きいまたは等しい	109	165
UL	上限リミット	160	224
WDT	ウォッチドックタイマリセット	143	198
WRITE	特殊モジュールデータ出力	238	320
XCHG	データ交換	22	66
XFER	拡張転送命令	236	310

株式会社 **東芝**

＜技術問い合わせ先＞

社会インフラシステム社

制御・計装機器営業部

営業第四担当

(03)3457-4778(ダイヤルイン)

府 中 事 業 所

計測制御機器部

開発・設計第三担当

(042)333-2256(ダイヤルイン)

