

ソフトウェアマニュアル
プログラミング

HIDIC
S10 α シリーズ

ラダー図
For Windows[®]

2 α
4 α
S10mini
シリーズ

対象機種

HIDIC-S10/2 α	NESP-S25E
HIDIC-S10/2 α E	NESP-2 α E
HIDIC-S10/2 α H	NESP-2 α H
HIDIC-S10/2 α Hf	NESP-2 α Hf
HIDIC-S10/4 α	NESP-S25M
HIDIC-S10/4 α H	NESP-4 α H
S10mini モデルS	
S10mini モデルH	
S10mini モデルF	
S10mini モデルD	
S10mini モデルC	

本製品を輸出される場合には、外国為替及び外国貿易法の規制並びに米国輸出管理規則など外国の輸出関連法規をご確認の上、必要な手続きをお取りください。
なお、不明な場合は、弊社担当営業にお問合わせください。

1997年 9月 (第1版) S A J - 3 - 1 2 1 (A) (廃版)
1999年12月 (第2版) S A J - 3 - 1 2 1 (B) (廃版)
2000年 9月 (第3版) S A J - 3 - 1 2 1 (C) (廃版)
2003年 9月 (第4版) S A J - 3 - 1 2 1 (D) (廃版)
2007年 9月 (第5版) S A J - 3 - 1 2 1 (E)

- このマニュアルの一部、または全部を無断で転写したり複製することは、固くお断りいたします。
- このマニュアルの内容を、改良のため予告なしに変更することがあります。



安全上のご注意

- システムの構築やプログラムの作成などは、このマニュアルの記載内容をよく読み、書かれている指示や注意を十分理解してから行ってください。誤操作により、システムが故障することがあります。
- このマニュアルは、必要なときすぐに参照できるように、手近なところに保管してください。
- このマニュアルの記載内容について疑問点または不明点がございましたら、最寄りの弊社営業またはSEまでお知らせください。
- お客様の誤操作に起因する事故発生や損害については、弊社は責任を負いかねますのでご了承ください。
- 弊社提供ソフトウェアを改変して使用した場合に発生した事故や損害については、弊社は責任を負いかねますのでご了承ください。
- 弊社提供以外のソフトウェアを使用した場合の信頼性については、弊社は責任を負いかねますのでご了承ください。
- ファイルのバックアップ作業を日常業務に組み入れてください。ファイル装置の障害、ファイルアクセス中の停電、誤操作、その他何らかの原因によりファイルの内容を消失することがあります。このような事態に備え、計画的にファイルのバックアップを取っておいてください。
- 弊社製品が故障や誤動作したりプログラムに欠陥があった場合でも、使用されるシステムの安全が十分に確保されるよう、保護・安全回路は外部に設け、人身事故や重大な災害に対する安全対策が十分確保できるようなシステム設計としてください。
- 非常停止回路、インタロック回路などはPLCの外部で構成してください。PLCの故障により、機械の破損や事故の恐れがあります。
- 運転中のプログラム変更、強制出力、RUN、STOPなどは十分安全を確認してから行ってください。誤操作により、機械の破損や事故の恐れがあります。

このマニュアルは、以下のプログラムプロダクトの説明をしたものです。

<プログラムプロダクト>

S-7890-02 「ラダー図システム」 07-07

S-7890-17 「4 α ラダー図システム」 07-06

S-7890-18 「4 α Hラダー図システム」 07-06

S-7890-49 「4 α リプレース用ラダー図システム」 01-00

変更内容 (SAJ-3-121) 「4 α リプレース用ラダー図システム」 (01-00)

追加・変更内容	ページ
4 α リプレース用ラダー図システム新規サポートによる修正	iii, 11, 13, 18, 21, 26

上記追加変更の他に、記述不明瞭な部分、単なる誤字・脱字などについては、お断りなく訂正しました。

はじめに

このマニュアルはラダープログラムを作成するときの命令語について説明しています。
命令語には大きく分けてラダー命令と演算ファンクション命令があります。
ラダー命令はリレー回路の動作を行うものです。
演算ファンクションは加減乗除などの演算処理を行うものです。

このマニュアルでは特に指定がない場合、
2 α は2 α , 2 α E, 2 α H, 2 α Hf
4 α は4 α , 4 α H
S10miniはS10miniモデルS, H, F, D, C
を示します。

NESP (Nissan Electronic Sequence Processor) シリーズは、
下記の対応を参照のうえ使用してください。

【HIDIC-S10 α シリーズ】		【NESPシリーズ】
HIDIC-S10/2 α	…………	NESP-S25E
HIDIC-S10/2 α E	…………	NESP-2 α E
HIDIC-S10/2 α H	…………	NESP-2 α H
HIDIC-S10/2 α Hf	…………	NESP-2 α Hf
HIDIC-S10/4 α	…………	NESP-S25M
HIDIC-S10/4 α H	…………	NESP-4 α H

<関連マニュアル>

- ・ソフトウェアマニュアル オペレーション ラダー図 For Windows® V7 (マニュアル番号 SAJ-3-131)
- ・ソフトウェアマニュアル オペレーション 4 α リプレース用ラダー図 For Windows® (マニュアル番号 SAJ-3-156)

なお、HI-FLOWプログラムについては、下記マニュアルを参照してください。

- ・ソフトウェアマニュアル プログラミング HI-FLOW For Windows® (マニュアル番号 SAJ-3-122)
- ・ソフトウェアマニュアル オプション CPU間リンク For Windows® (マニュアル番号 SAJ-3-141)

<商標について>

- ・Microsoft® Windows® operating system, Microsoft® Windows® 95 operating system, Microsoft® Windows® 98 operating system, Microsoft® Windows® 2000 operating system, Microsoft® Windows® XP operating systemは、米国Microsoft Corporationの米国およびその他の国における登録商標です。
 - ・Ethernetは米国Xerox Corp.の登録商標です。
- その他、記載されている会社名、製品名は、各社の商標または登録商標です。

目 次

1	ラダー命令	1
1.1	ラダー回路の大きさ	2
1.2	右下がり回路と動作順序	3
1.3	ラダー回路とステップ	5
1.4	停復電時とSTOPからRUN時の状態	5
1.5	ラダー命令一覧	6
2	演算ファンクション	29
2.1	機能概要	30
2.2	機能仕様	32
2.3	演算ファンクション入力	34
2.4	演算ファンクション一覧表	35
3	処理時間	155
3.1	2 α の処理時間	156
3.2	4 α の処理時間	161
3.2.1	スキャンタイム	161
3.2.2	スキャンタイム計算例	162
3.3	平均スキャンタイムの表示	164
索 引	165

図 目 次

図 3-1	OSによる演算処理の概要	156
図 3-2	OS処理タイミング	159
図 3-3	リモートI/OとOS処理	160
図 3-4	PCs処理フロー	161
図 3-5	スキャンタイムの計算例	162
図 3-6	スキャンタイム表示プログラム	164

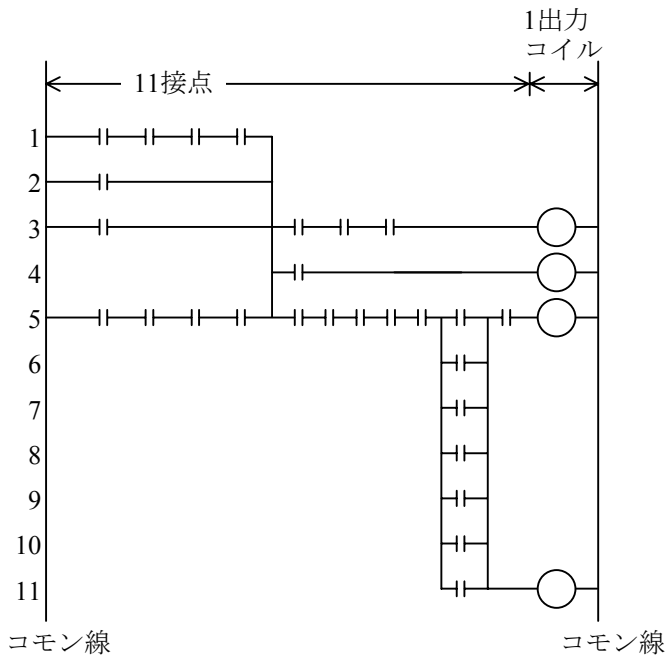
表 目 次

表 3-1	処理時間一覧	163
-------	--------------	-----

1 ラダー命令

1 ラダー命令

1. 1 ラダー回路の大きさ



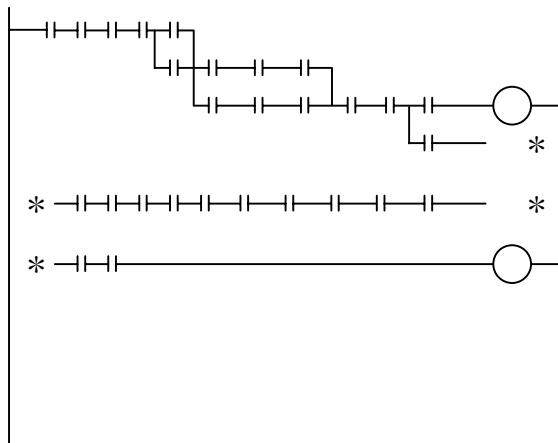
・ラダー回路は左側のコモン線から右側のコモン線に接続するコイルで終了します。この1つにまとまったラダー回路を1ブロックと呼びます。

・最大回路は次となります。

横：11接点 + 1出力コイル

縦：11行

[11接点以上をAND接続するとき]



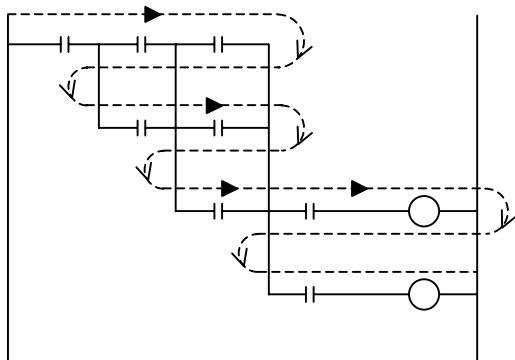
・AND接続が多い場合には図のように“折返し回路”となりAND接続11接点以上の回路が作成できます。

[制限]

- ・*印の前に分岐を残さないでください。
- ・*印以後に分岐回路を作成できません。

1.2 右下がり回路と動作順序

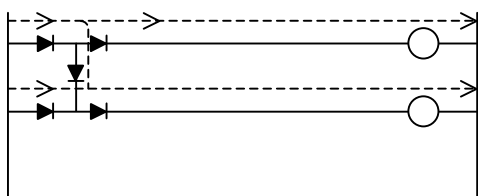
[右下がり回路の例]



- ・ラダー回路は左から右、上から下への回路構成となります。

[動作順序の考え方]

ラダー回路においてダイオード
を入れたイメージ



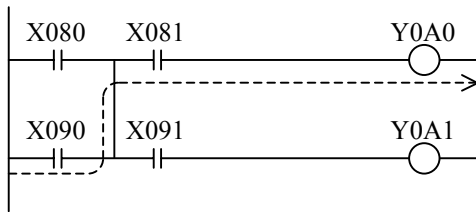
- ・ラダー回路の動作は、左から右、上から下への順に動作します。考え方としては、接点と下へ下がる分岐に、ダイオードを入れたイメージです。

1 ラダー命令

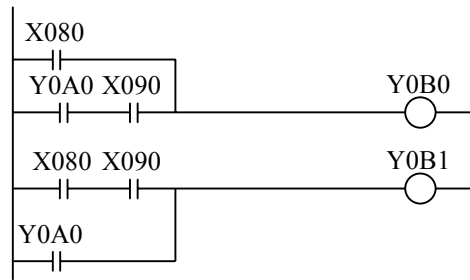
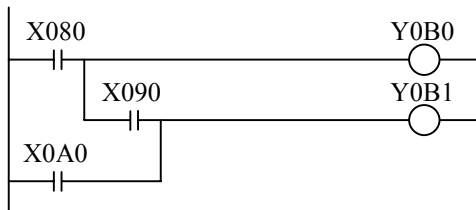
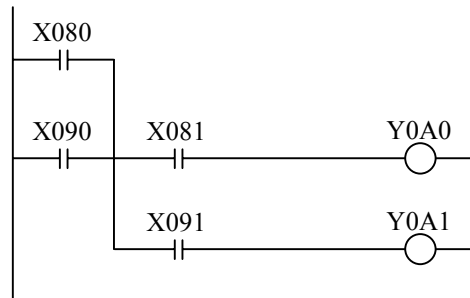
【右下がりラダー回路の動作】

下図の左欄のリレー回路例の動作を行いたいときは、右欄の右下がりラダー回路を作成してください。

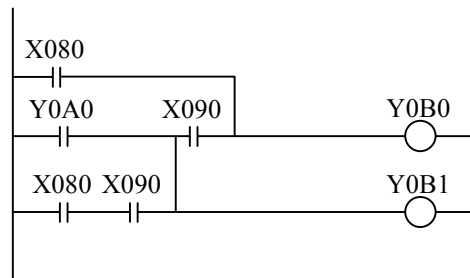
[リレー回路例]



[右下がりラダー回路例]



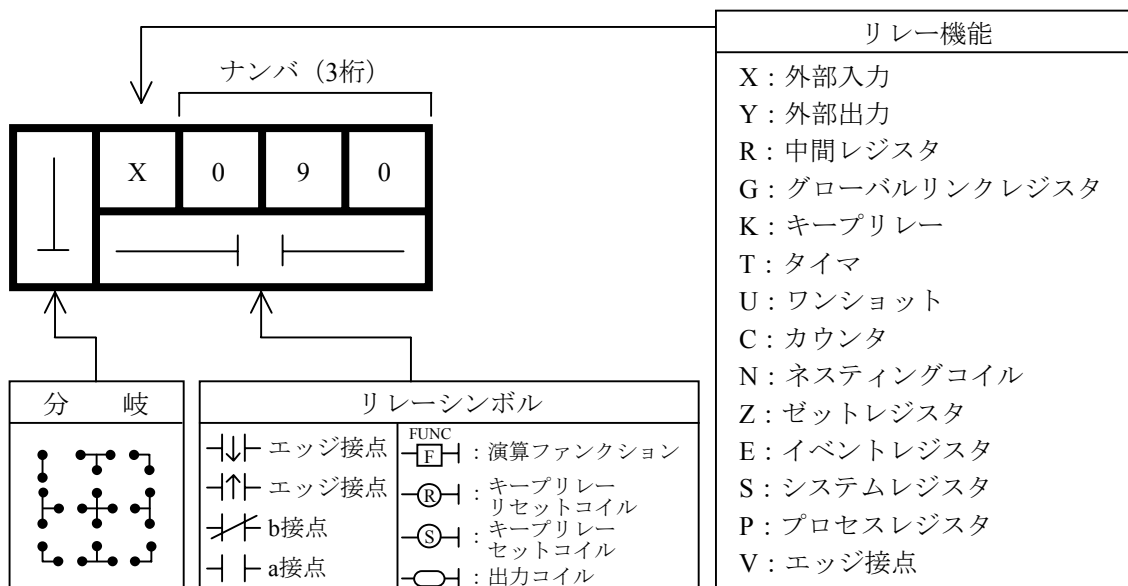
または



1. 3 ラダー回路とステップ

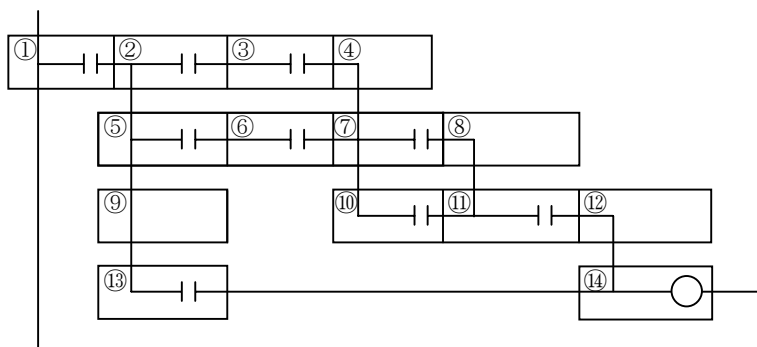
ラダー回路1ステップは次の構成になります。

[1ステップの構成]



● 分岐の , , は1ステップとなります。

[ラダー回路と命令語]



: 1ステップ

左図では14ステップとなります。







また、プログラムは○で示した順番で格納、実行されます。

1. 4 停復電時とSTOPからRUN時の状態

リレー機能	ON/OFF状態		計数値/数値	
	停復電時	STOP→RUN時	停復電時	STOP→RUN時
T, U	OFF	ホールド	0クリア	ホールド
C	ホールド	ホールド	ホールド	ホールド
K	ホールド	ホールド		
X, Y, R, M, G, N, Z, E, P, V	OFF	ホールド		
FW, DW			ホールド	ホールド

1 ラダー命令

1.5 ラダー命令一覧

	名称	シンボル							ナンバ		備考
									2α, S10mini	4α	
入出力	外部入力	X	●	-	-	-	-	-	000~FFF	000~FFF	リモートI/Oの入出力範囲は以下となります。 2α : 000~7FF 4α : 000~1FF 4αH : 000~3FF
	外部出力	Y	●	-	-	●	-	-	000~FFF	000~FFF	
内部補助機能	内部レジスタ	R	●	-	-	●	-	-	000~FFF	000~FFF	
	キーブリー	K	●	-	-	-	●	●	000~FFF	000~FFF	
	拡張内部レジスタ	M	●	-	-	●	-	-	000~FFF	000~FFF	
	オンディレイタイマ	T	●	-	-	●	-	-	000~1FF	000~1FF	000~00F (10msタイマに変更可。)
	ワンショットタイマ	U	●	-	-	●	-	-	000~0FF	000~0FF	
	アップダウンカウンタ	CU	-	-	-	●	-	-	00~FF	00~FF	CU : アップカウンタ CD : ダウンカウンタ CR : リセットコイル C0 : 接点
		CD	-	-	-	●	-	-			
		CR	-	-	-	●	-	-			
		C0	●	-	-	-	-	-			
	グローバルリンクレジスタ	G	●	-	-	●	-	-	000~FFF	000~FFF	
	ネスティングコイル	NM	-	-	-	●	-	-	01~FF	—	NM : マスタリセット) ナンバ 入力 NZ : ゼロコントロール) ごとに選択 N0 : 接点 (起動状態)
		NZ	-	-	-	●	-	-			
		N0	●	-	-	-	-	-			
	プロセスレジスタ	P	●	-	-	●	-	-	001~080	—	Cプログラム起動用
イベントレジスタ	E	●	-	-	●	-	-	000~FFF	000~FFF	CPU LED表示は2αのみで000~1FF。400~FFFは4チャネルアナログバルスカウントで使用。	
エッジ接点	V	-	●	●	-	-	-	000~FFF	000~FFF		
ゼットレジスタ	Z	●	-	-	●	-	-	000~3FF	000~3FF	0FE : トレーススタート信号 0FF : トレースストップ 信号 200 : H-7338方式割り込み起動信号 (2α, 4αのみ)	
システムレジスタ	S	●	-	-	-	-	-	000~005	演算ファンクションフラグ		
								010~01F	ラダープログラム制御レジスタ		
								100~15F	ラダープログラム制御カウンタ		
								300~47F	リモートI/O状態表示レジスタ		
								500~6FF	580~5BF 6CO~6FF	オプションモジュール状態表示レジスタ	
								BA0~BBF	I/O固定パターン		
								BF0~BFF	CPUステータスレジスタ		
ファンクションデータレジスタ	DW	-	-	-	-	-	000~FFF	000~7FF	定数データエリア		
ファンクションワークレジスタ	FW	-	-	-	-	-	000~BFF	000~BFF	ワークエリア		

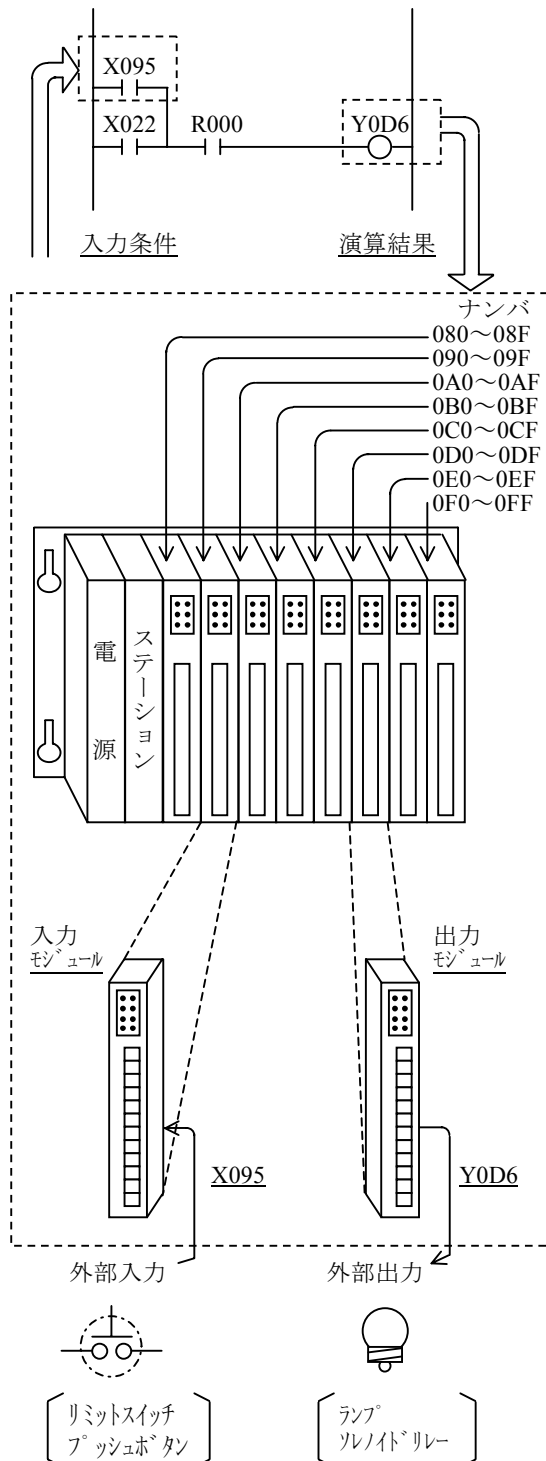
● : 使用できるシンボル

	2α, S10mini	4α
ナンバの範囲	000~FFF	000~FFF
リモートI/Oの入出力範囲	000~7FF	000~1FF(4α) 000~3FF(4αH)

PCsに接続された外部入出力モジュール経由で信号の入出力を行います。

X：入力モジュール経由で外部の入力信号を取り込みます。

Y：ラダープログラムの演算結果を出力モジュールから外部へ出力します。



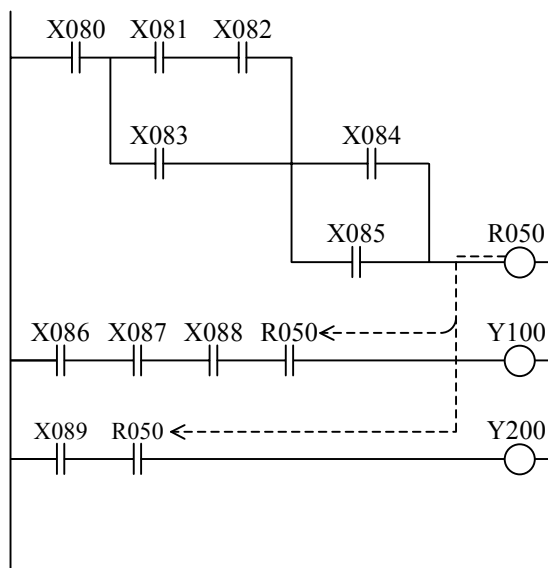
R, M 内部レジスタ

	2α, S10mini, 4α
ナンバの 範囲	000~FFF

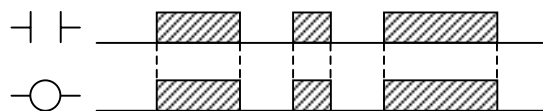
ラダー命令の演算結果を中継するための内部レジスタです。

コイル (—○—) がONと同時に接点 (—|—) がONし、コイル (—○—) がOFFすると同時に接点 (—|—) もOFFします。

<回路例>



<タイムチャート>



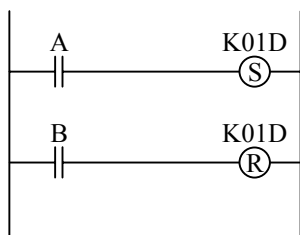
	2α, S10mini, 4α
ナンバの 範囲	000~FFF
セットリング パルス幅	最小1スキャンタイム
セット、リセット 同時入力	プログラムの下方にある 方が優先されます。

キープリレーは、セットコイル (—S—) がONすると、リセットコイル (—R—) がONするまで接点 (—|—) がONし、接点の状態は停電保持されます。

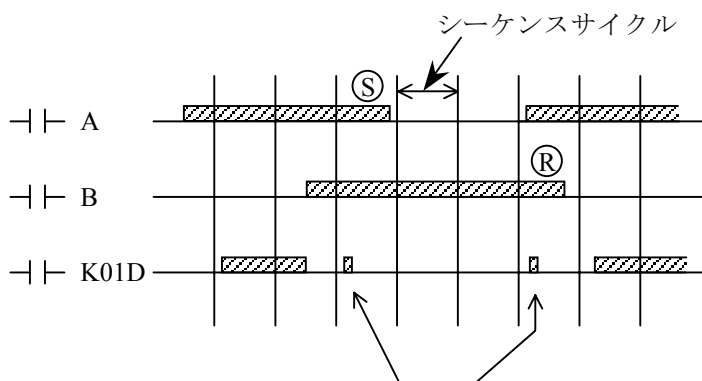
また、セットコイルとリセットコイルが同時にONした場合、プログラムの下方にある方が優先されます。

[リセット優先回路]

<回路例>



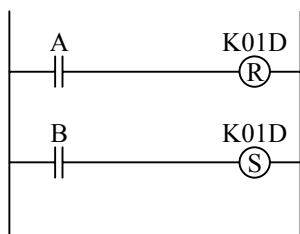
<タイムチャート>



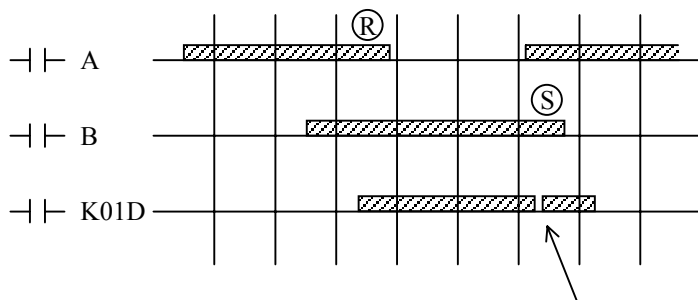
—S— が実行され —R— までの間、ONします。

[セット優先回路]

<回路例>



<タイムチャート>



—R— が実行され —S— が実行されるまでの間、OFFとなります。

T オンディレイタイマ

[100msタイマ]

	2α, S10mini, 4α
ナンバの 範囲	000~1FF
設定値	0.0~999.9秒
誤差	最小100ms+1スキャンタイム
セットリング [*] (*) パルス幅	最小100ms

オンディレイタイマのコイル (—○—) がONしてから、設定値の時間だけ遅れて、接点 (—|—) がONします。

設定値の範囲は0.0~999.9秒で0.1秒単位に設定できます。

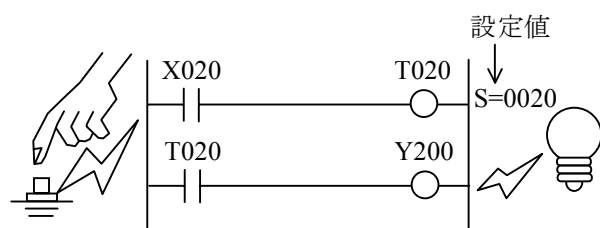
なお、先頭から16点 (T000~T00F) はエディション設定により10msタイマとして使用できます。

[10msタイマ (エディション設定による。)]

	2α, S10mini, 4α
ナンバの 範囲	000~00F
設定値	0.01~99.99秒
誤差	最小10ms+1スキャンタイム
セットリング [*] (*) パルス幅	最小10ms

(*) セットリングパルス幅とは、オンディレイタイマのコイルをONする接点がONしていなければならない最小時間のことを意味します。

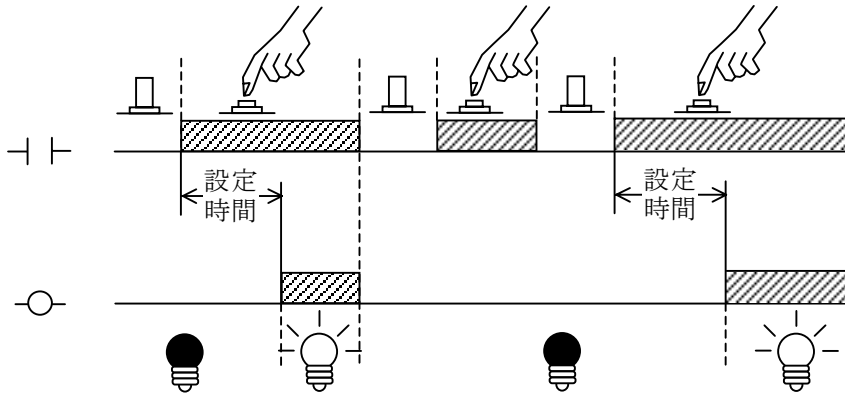
<回路例>



図に示した回路の場合、押しボタン

(X020) を押してから、設定時間後 (この場合2秒後) にランプ (Y200) が点灯し、手を離すと同時にランプが消灯します。

<タイムチャート>



コイル (—○—) がONしている時間が設定時間より短い場合、接点 (—|—) はONしません。
 2α のカウンタ値は、65535までカウントします。オーバーフロー時は、0からカウントを始めます。

<注意事項>

(1) S10/2 α シリーズ、S10miniシリーズ (モデルS, H, F, D)

100msタイマで使用する場合、オンディレイタイマのコイルのON/OFF検出は、ラダー回路の実行周期 (シーケンスサイクル) とは非同期に、100msごとに行われます (10msタイマの場合は、10msごとに行われます)。コイルがONしている時間が100msより短い場合、コイルのONが検出されず、オンディレイタイマが動作しない場合があります。オンディレイタイマを確実に動作させるためには、コイルのON時間が100ms以上になるようにラダー回路を作成してください。

(2) S10/4 α シリーズ、S10mini (モデルC)

100msタイマで使用する場合、オンディレイタイマのコイルのON/OFF検出は、ラダー回路のコイル実行時に行われます (10msタイマの場合は、ラダー回路の実行周期 (シーケンスサイクル) とは非同期に、10msごとに行われます)。

U ワンショット

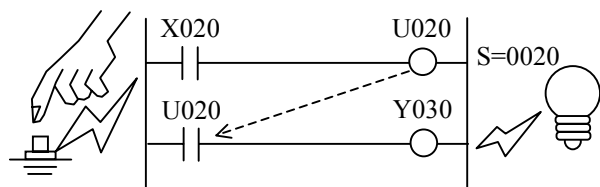
	2α, S10mini, 4α
ナンバの 範囲	000～0FF
設定値	0.0～999.9秒
誤差	最小100ms+1スキャンタイム
セットリング [*] (*) パルス幅	最小100ms

ワンショットのコイル (—○—) がONしてから設定値の時間だけ接点 (—|—) がONします。

設定値の範囲は0.0～999.9秒で0.1秒単位に設定できます。

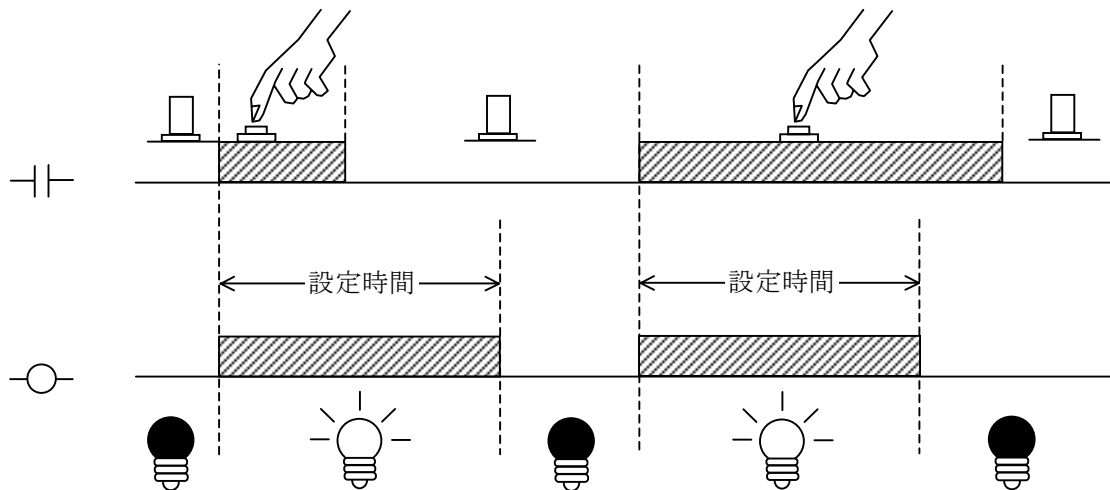
(* セットリングパルス幅とは、ワンショットのコイルをONする接点がONしていなければならない最小時間のことを意味します。

<回路例>



図に示した回路の場合、押しボタンを押すと、ランプ (Y030) が設定時間 (この場合2秒) だけ点灯します。

<タイムチャート>



2αのカウンタ値は、65535までカウントします。オーバーフロー時は、0からカウントを始めます。

<注意事項>

(1) S10/2αシリーズ、S10miniシリーズ (モデルS, H, F, D)

ワンショットのコイルのON/OFF検出は、ラダー回路の実行周期 (シーケンスサイクル) とは非同期に、100msごとに行われます。コイルがONしている時間が100msより短い場合、コイルのONが検出されず、ワンショットが動作しない場合があります。ワンショットを確実に動作させるためには、コイルのON時間が100ms以上になるようにラダー回路を作成してください。

(2) S10/4αシリーズ、S10mini (モデルC)

ワンショットのコイルのON/OFF検出は、ラダー回路のコイル実行時に行われます。

C アップダウンカウンタ

			2α, S10mini, 4α
ナンバの 範囲		CU	00~FF
		CD	
	CR		
	CO		
設定値	0~9999カウント		
セットリング ^(*) パルス幅	最小1スキャンタイム		
セット、リセット 同時入力	リセット優先		
停電時	不揮発		

アップカウンタ (CUΔΔ) とダウンカウンタ (CDΔΔ) から構成され、カウント値はアップカウンタのコイル (—○—) 立ち上がり時カウントアップ (+1)、ダウンカウンタのコイル立ち下がり時にカウントダウン (-1) します。

カウンタ接点 (⊥|⊥) はカウント値が設定値を超えたときにONします。

リセットコイルはカウント値を0クリアし、接点がOFFします。

CU：アップカウンタ

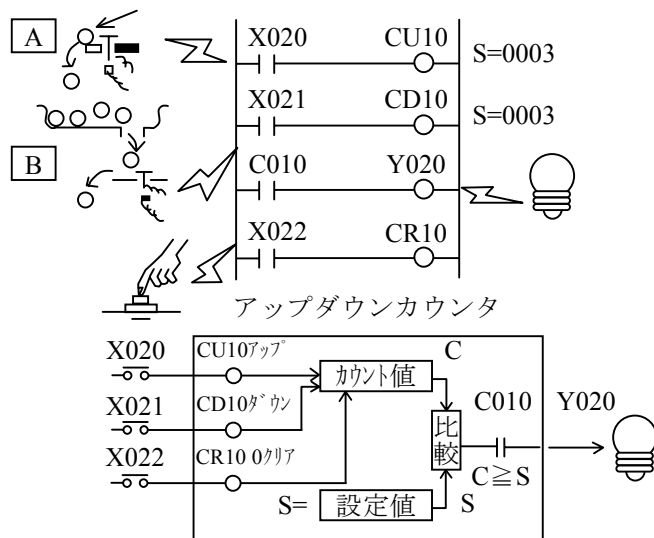
CD：ダウンカウンタ

CR：リセット

CO：カウンタ接点

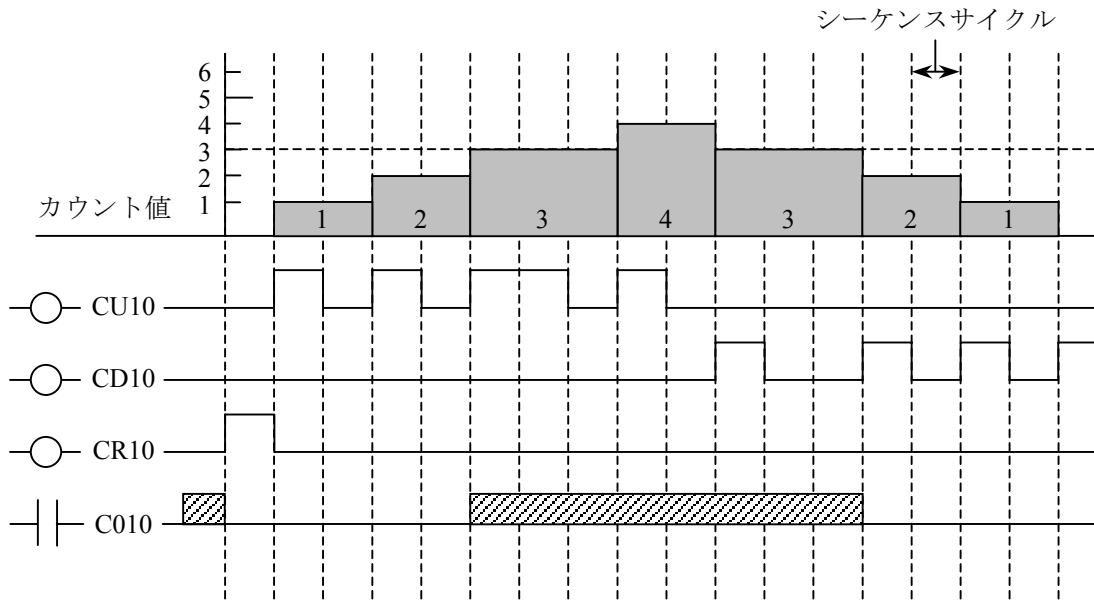
(*) セットリングパルス幅とは、アップカウンタ、ダウンカウンタ、リセットのコイルをONする接点がONしていなければいけない最小時間のことを意味します。

<回路例>



- 図の回路は、上方からバスケットに入ったボールの数をスイッチ **A** (X020) でカウントし、下方へ落ちたボールの数をスイッチ **B** (X021) でカウントし、現在バスケットに入っているボールの数をカウントします。
- バスケットに入っているボールの数が3個以上になるとランプ (Y020) が点灯します。また、押しボタン (X022) を押すとカウント値が0クリアされ、ランプが消灯します。

<タイムチャート>



(注)

- ・アップカウンタの場合、カウント値は設定値を超えてもカウントされ続けます。カウント値がオーバーフロー (0xFFFFを超えた場合) すると、0から再度カウントします。そのとき、ON状態のカウンタ接点はOFF状態に変わります。
- ・ダウンカウンタの場合、カウント値が0になるとカウントダウンは停止します。

G グローバル リンクレジスタ

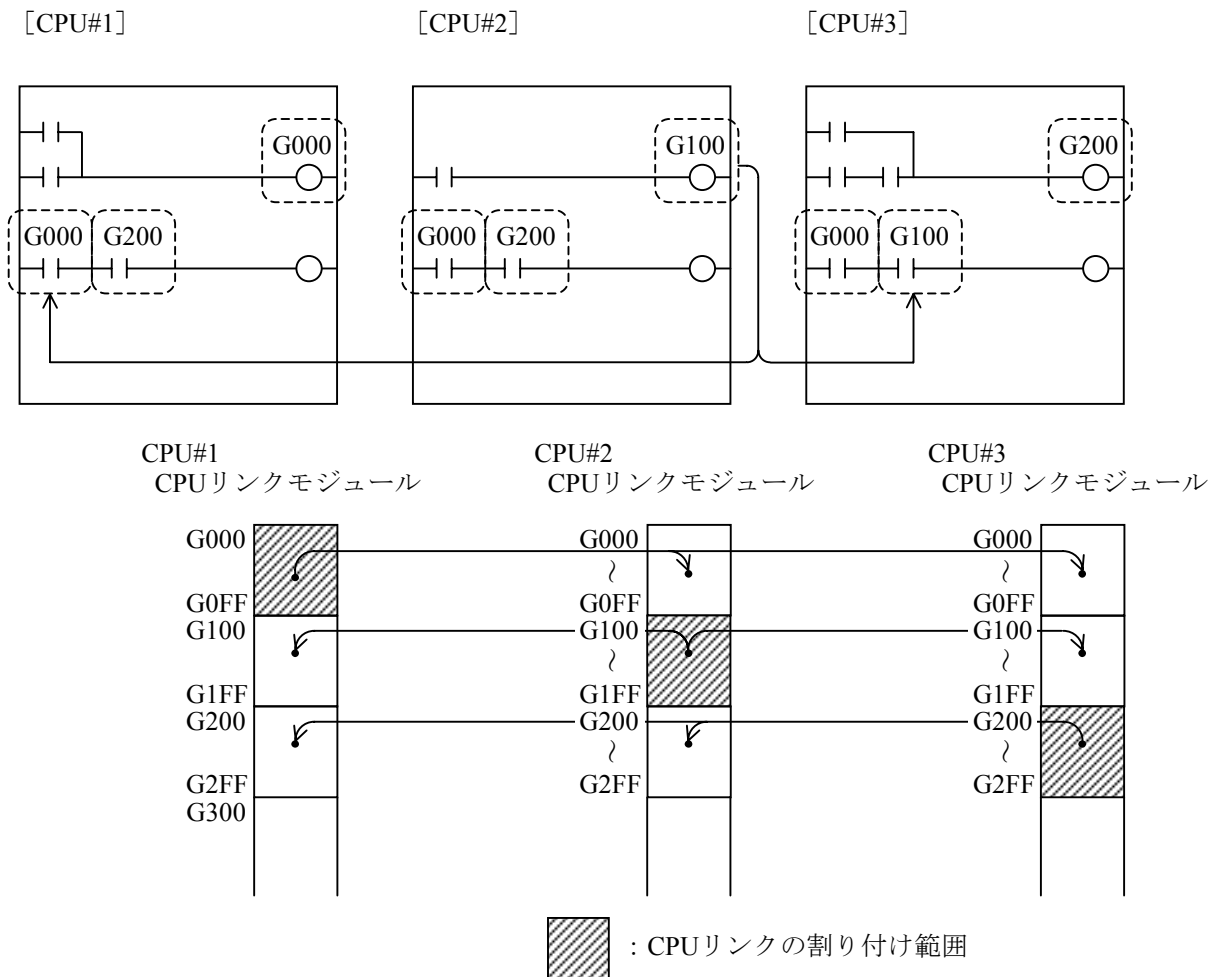
	2α, S10mini, 4α
ナンバの 範囲	000~FFF

CPU間リンクモジュール（オプション）実装時にCPU間のインタロック情報などを交換するためのレジスタです。

動作はコイル（ $\text{---}\bigcirc\text{---}$ ）のON（またはOFF）で、他CPUの同一ナンバの接点（ $\text{---}| \text{---}$ ）がON（またはOFF）します。

CPU間リンクの割り付けは、「ソフトウェアマニュアル オプション CPU間リンク For Windows®（マニュアル番号 SAJ-3-141）」を参照してください。

<回路例（CPU間リンクカード使用時）>



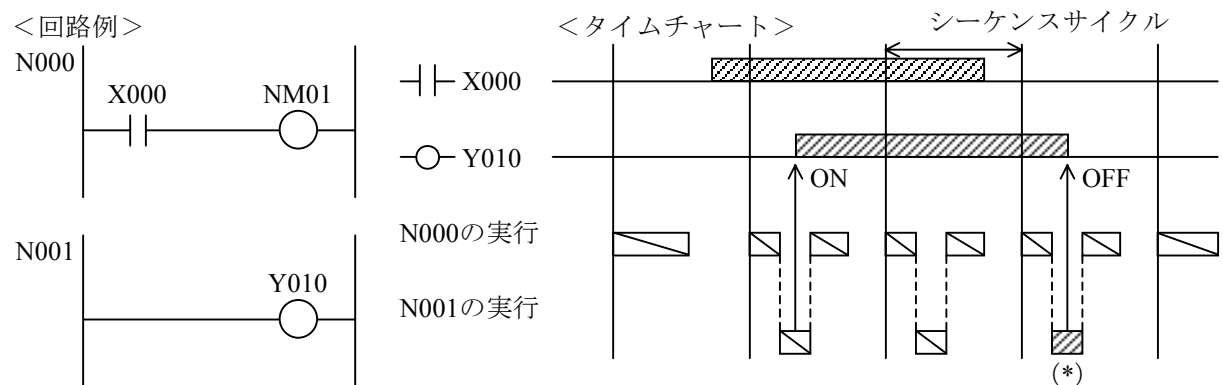
	2α, S10mini	4α
ナンバの範囲	001~OFF	—
ネスティング最大レベル	4レベル	—

シーケンスプログラムを制御対象プラントごとに分割し、モジュール化するための機能を持ちます。

ネスティングコイルにはNコイルの立ち下がり（ON→OFF）時に使用されているコイルをOFFする“マスタコントロール”と、前回の状態を保持する“ゾーンコントロール”の2種類から選択します。

また、Nは最大4レベルまでのネスティングが可能です。

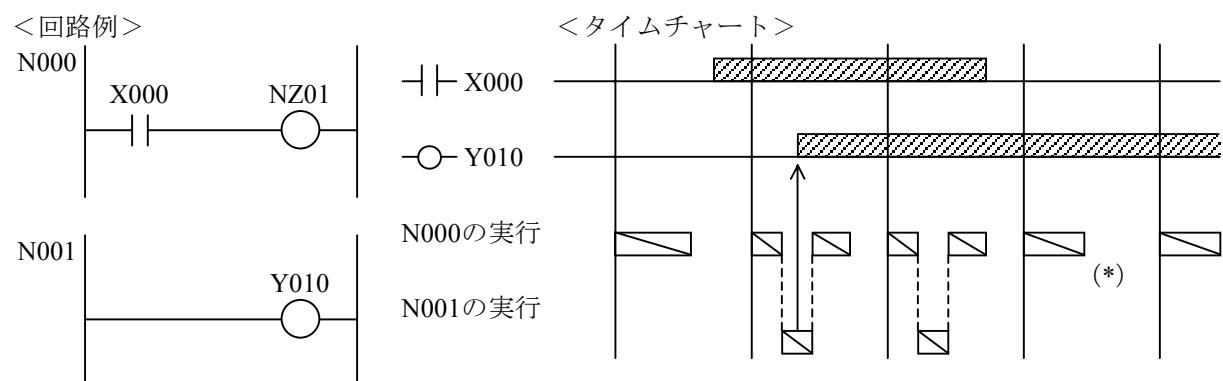
[マスタコントロール (NM)]



(*) NM01の立ち下がり（ON→OFF）のとき、NM01で使用されているコイル（—○—）をすべてOFFします。


K（キープリレー）および \uparrow 、 \downarrow エッジ接点を使用した回路の場合、OFFされない場合があります。

[ゾーンコントロール (NZ)]



(*) NZ01の立ち下がってもマスタリセット付きと異なり、N001で使用されているコイルは前回の状態を保持しています。

P プロセスレジスタ

	2α, S10mini	4α
ナンバの範囲	001~080	—
起動方法	レベル起動 	—

ユーザが、C言語やアセンブラ言語で作成したコンピュータ言語によるプログラム（以下タスクと呼びます）をラダープログラムにより起動するためのレジスタです。

このコイルをONすることによりナンバに対応したタスク起動（Queue）が行われます。

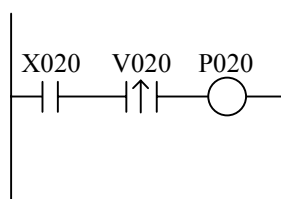
（注）S10miniモデルCではタスクは非サポートのため、このコイルがONしてもタスク起動は行われません。

[プロセスレジスタの割り付け]

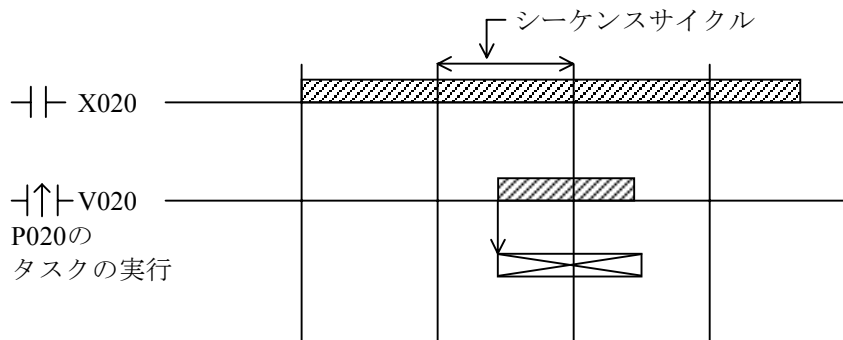
区分	ナンバ	名称	内容
ユーザ作成	P001	イニシャルタスク	CPUをGR（リセット）したときに必ず起動されるタスクです。システムのイニシャライズを行うプログラムを割り付けてください。
	P002~P07F	ユーザタスク	ユーザが作成したプログラムを割り付けてください。
システム専用	P080	システムタスク	CPMSなどでデバッガサポート用のタスクなどを割り付けます。ユーザは使用しないでください。

<回路例>

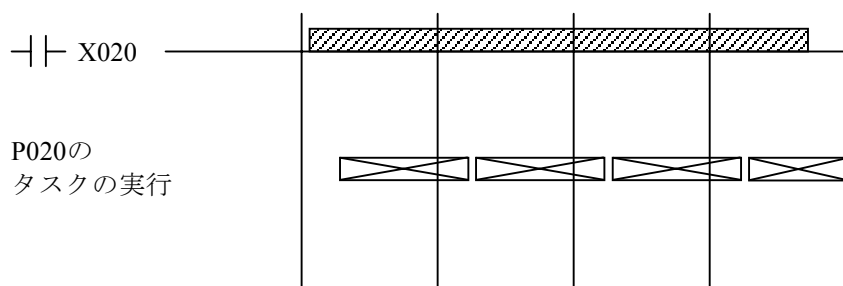
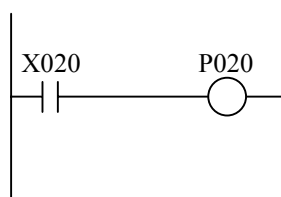
・ON時、1回実行



<タイムチャート>



・ON時、毎回実行



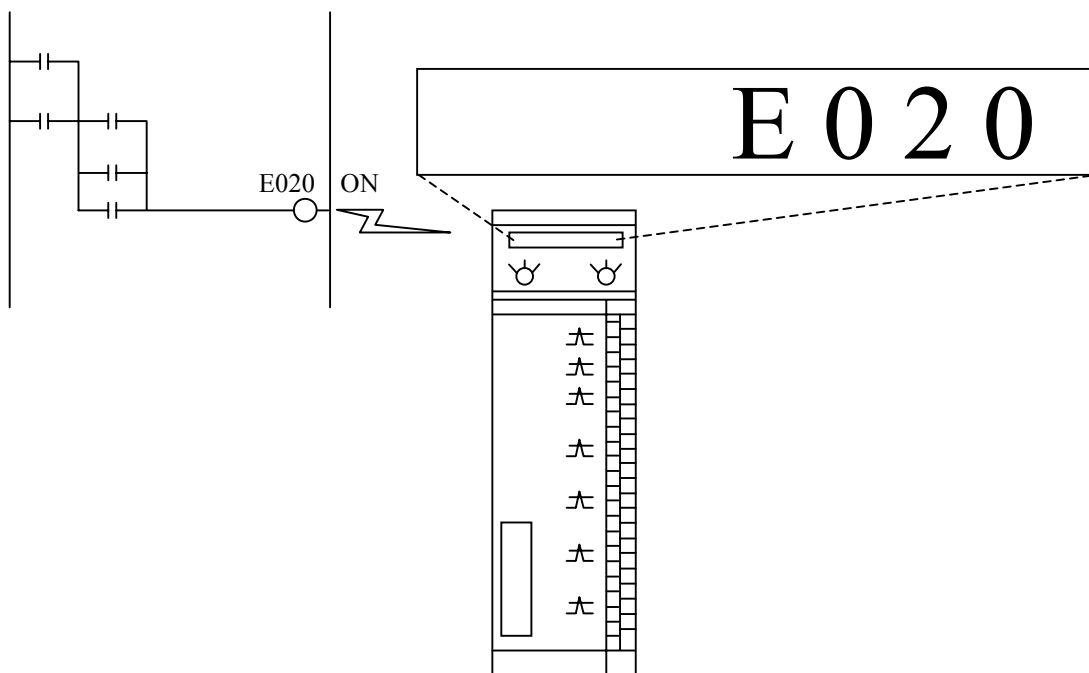
	2α, S10mini	4α
ナンバの 範囲	000~FFF	000~FFF
CPU LED 表示用の範囲	000~1FF	—

CPUの8桁のLEDにユーザのエラーなどのイベント情報を表示するためのレジスタです。

このレジスタのコイルがONしている場合、そのナンバがLEDに表示されます。

E400~EFFFは4チャンネルアナログパルスカウンタで使
用します。

<回路例>



V エッジ接点

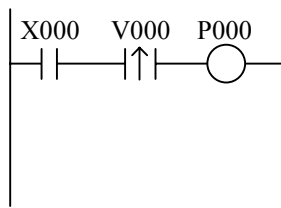
	2α, S10mini, 4α
ナンバの 範囲	000~FFF

エッジ接点には立ち上がりエッジ検出接点 (↑) と立ち下がりエッジ検出接点 (↓) があり、それぞれのエッジを検出した1スキャンだけONします。

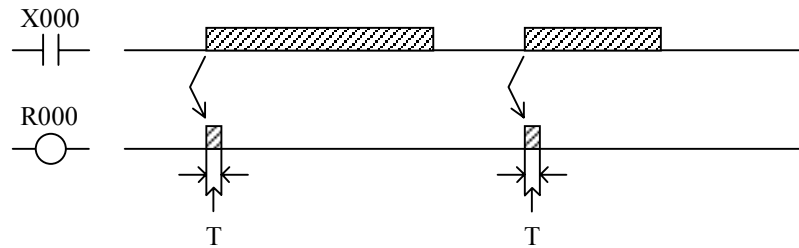
なお、同一ナンバを立ち上がりエッジ接点および立ち下がりエッジ接点として使用することはできません。

[立ち上がりエッジ接点]

<回路例>



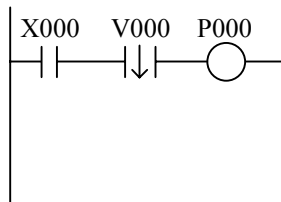
<タイムチャート>



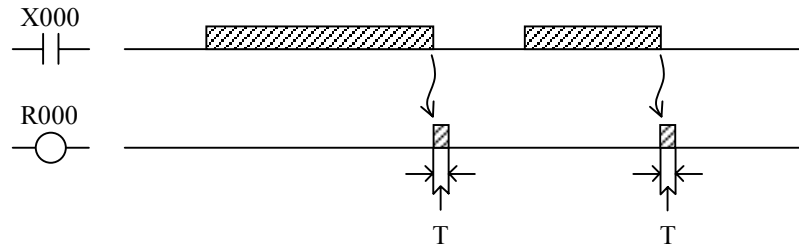
T: 1スキャンタイム

[立ち下がりエッジ接点]

<回路例>



<タイムチャート>



T: 1スキャンタイム

ナンバ の範囲	000~01F	シーケンスサイクル単位 でトレースするI/O情報を 設定します。
	0FE	ロジックトレーサ機能の スタート信号です。
	OFF	ロジックトレーサ機能の ストップ信号です。

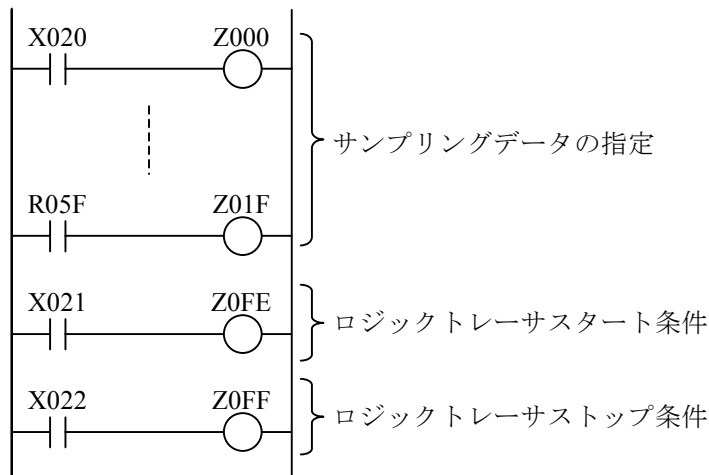
ゼットレジスタには、ロジックトレーサ機能（S10miniでは非サポート）およびH-7338方式による上位割り込み用のレジスタが配置され、それぞれ左の表のようになります。

Z200	上位計算機へ、H-7338方式の割り込みを発生します。
------	-----------------------------

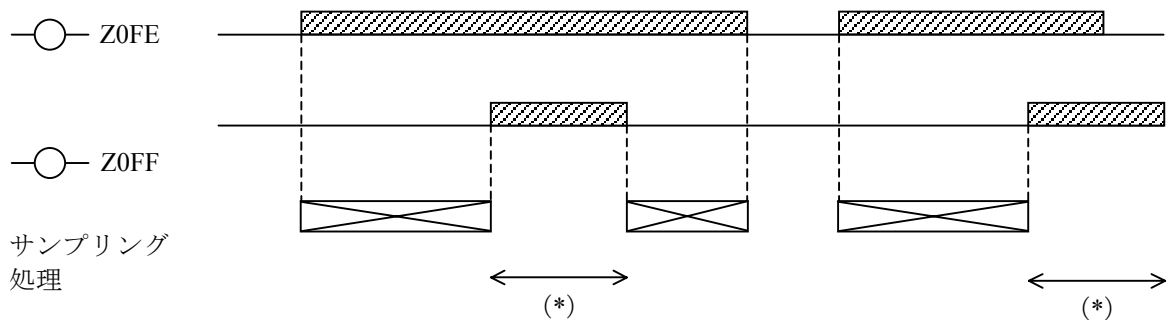
（2α, 4αのみ。S10miniでは、割り込みは発生しません。）

● ロジックトレーサとZコイル

<回路例>



<タイムチャート>

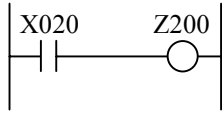


Z ゼットレジスタ

● 上位割り込みレジスタ (2 α , 4 α のみ)

[2 α の場合]

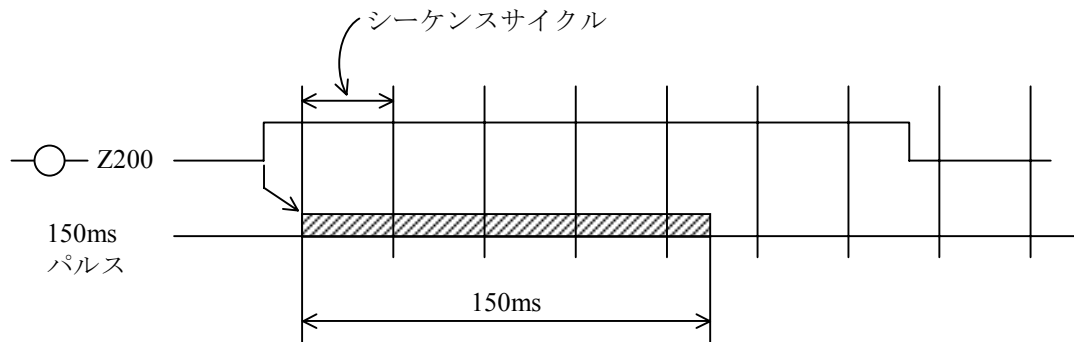
<回路例>



上位への割り込みZ200の立ち上がり検出時に150msの割り込みを発生します。なお、この処理はシーケンスサイクルと同期して行われます。150msの割り込みを送信中に、再度Z200の立ち上がりを検出した場合は再度150ms分のパルスを出力します。

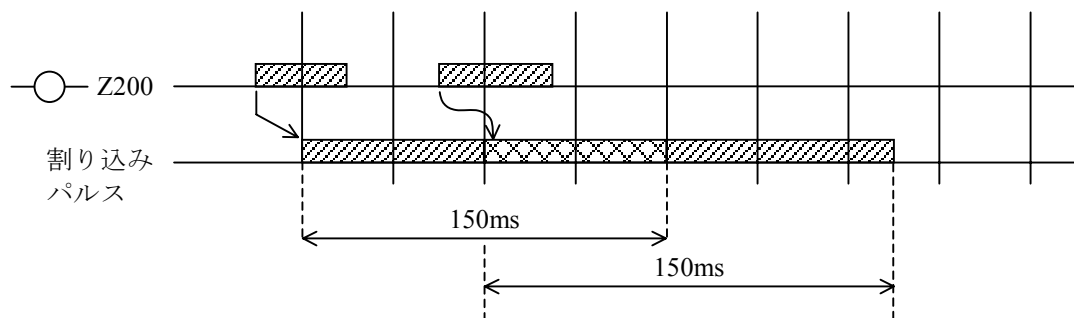
セットリングパルス幅：最小1シーケンスサイクル

<タイムチャート>



[割り込みパルスがのびる場合]

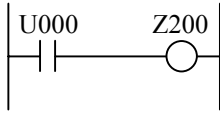
上記回路で、Z200のON \leftrightarrow OFFが150ms以内で2回以上ON \rightarrow OFFした場合、二重割り込みが発生しパルス幅が長くなります。



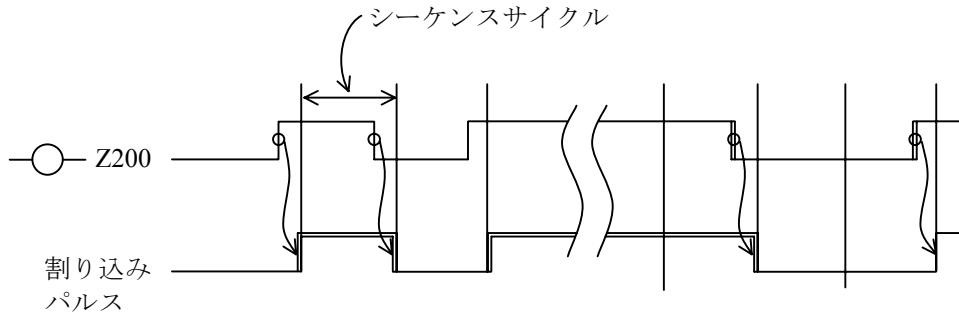
このようなことを防ぐには、割り込みを受け付ける上位計算機との間でインタロックを取ってください。

[4αの場合]

<回路例>



上位への割り込みZ200の立ち上がり検出時より、立ち下がり検出時までの時間、割り込み信号を出力します。なお、この処理はシーケンスサイクルと同期して行われます（通常ワンショットの出力を使用します）。



システムの動作状態などを反映した読み込み専用のレジスタです。

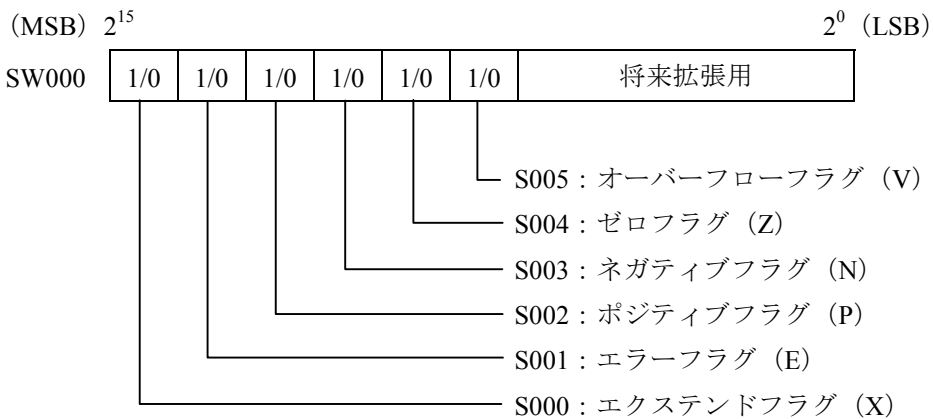
[システムレジスタ一覧表]

No.	レジスタNo.	概要
1	S000~S00F	演算ファンクションフラグレジスタ
2	S010~S01F	ラダープログラム制御レジスタ
3	S100~S15F	ラダープログラム制御カウンタ
4	S300~S47F	リモートI/O状態レジスタ
5	S500~S6FF	オプションモジュール状態レジスタ
6	SBA0~SBBF	I/O固定パターン
7	SBF0~SBFF	CPUステータスレジスタ

上記のレジスタ以外は将来拡張用となっています。

(1) 演算ファンクションフラグレジスタ

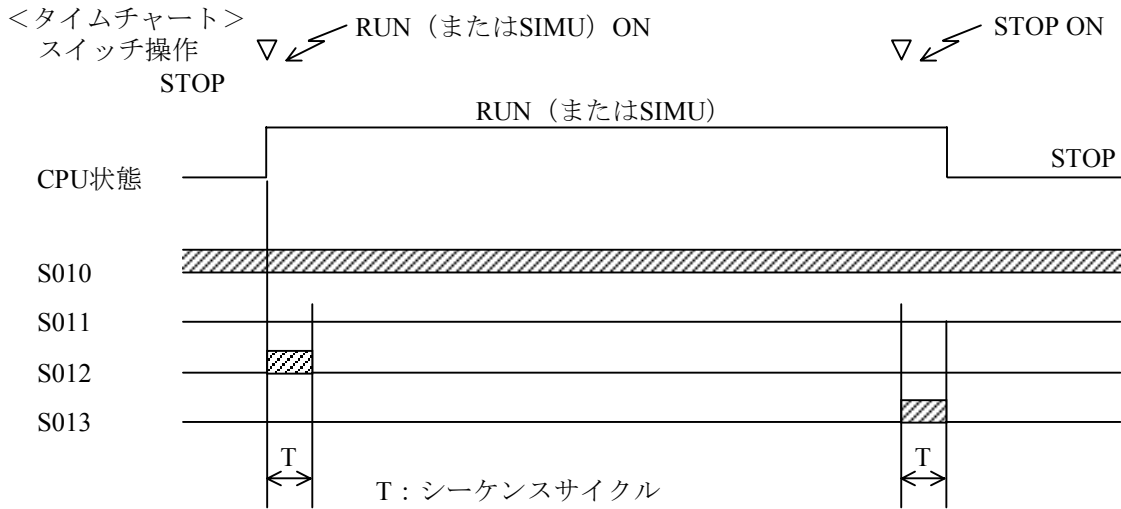
システム演算命令実行後のフラグの状態を示したレジスタです。各番号は次のように対応します。



S システムレジスタ

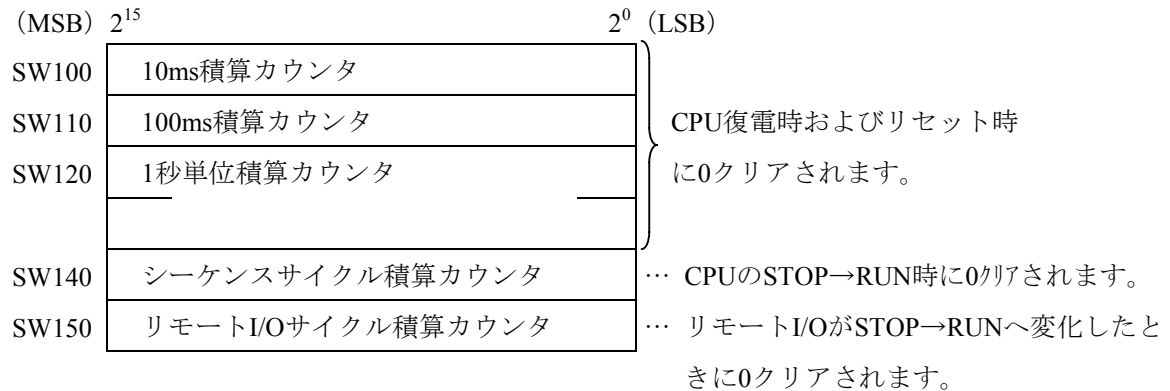
(2) ラダープログラム制御レジスタ

ラダープログラムの作成を、より容易にするための情報が格納されるレジスタです。



(3) ラダープログラム制御カウンタ

シーケンス制御で使用することのできる積算カウンタです。



- ・全カウンタとも、オーバーフロー時は“0”から再カウントします。
- ・カウンタの精度はOSの割り込みによって処理しているため、±10%程度の誤差が生じます。

(4) リモートI/O状態レジスタ

リモートI/Oの登録状態、タイムアウト、およびFUSE断ステーション情報を示すレジスタです。

[レジスタの割り付け]

S300	登録 ステーション	現在登録されているステーションに対応したレジスタに “1”が設定されます。
S380		
S400	タイムアウト ステーション	登録ステーション中タイムアウトエラーを発生している ステーションに対応したレジスタに“1”が設定されます。
S47F		
	FUSE断 ステーション	登録ステーション中FUSE断を発生しているステーション に対応したレジスタに“1”が設定されます。

[各ステーションとビットの対応]

No.	XまたはYのナンバ	登録ステーション	タイムアウトステーション	FUSE断ステーション
0	000~00F	S300	S380	S400
1	010~01F	S301	S381	S401
2	020~02F	S302	S382	S402
3	030~03F	S303	S383	S403
4	040~04F	S304	S384	S404
	}	}	}	}
124				
125	7D0~7DF	S37D	S3FD	S47D
126	7E0~7EF	S37E	S3FE	S47E
127	7F0~7FF	S37F	S3FF	S47F

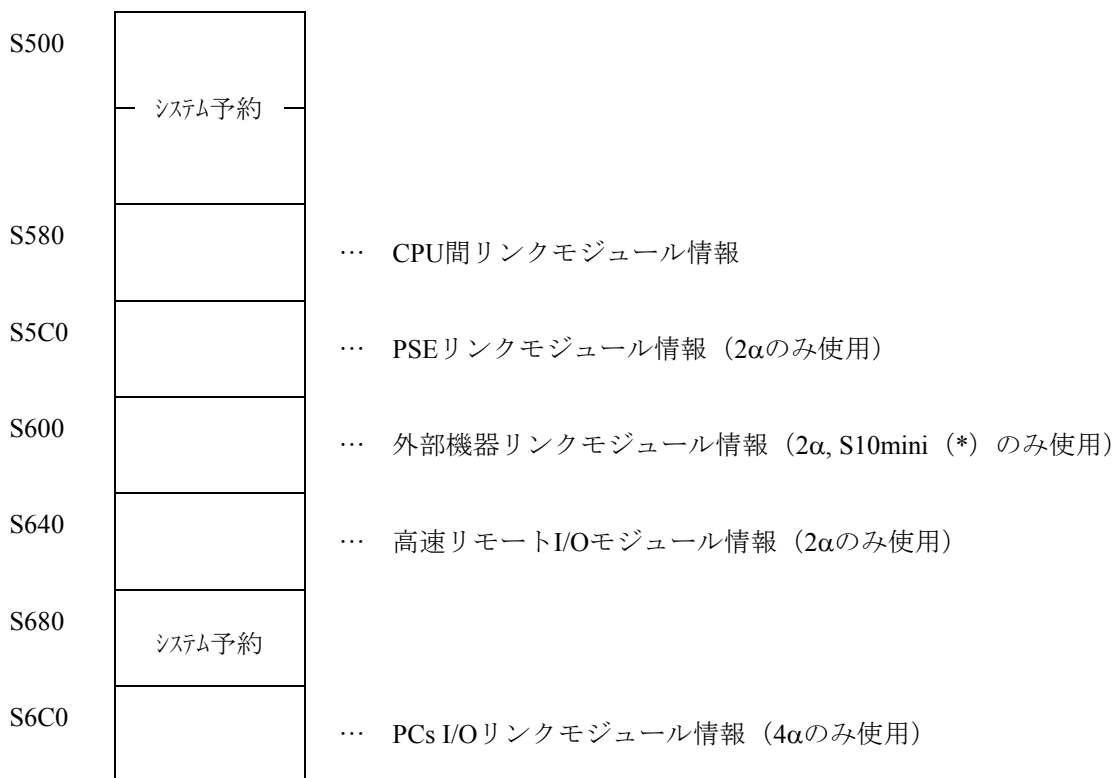
- ・4αのCPUベース内の出力モジュールがFUSE断を起こしたとき、S400に“1”が設定されます。
- ・2α, S10miniの場合、上記フラグはLED表示タイミング（2秒に1回）で設定されます。

S システムレジスタ

(5) オプションモジュール状態レジスタ

CPU間リンク、外部機器リンクなどのエラー情報が格納されるレジスタです。各モジュールごとの割り付けは以下ようになります。なお、このレジスタへのデータの設定は各モジュールのサポート用サブOSにて行うため、詳細なビット構成は、拡張モジュール購入時に付属されているマニュアルを参照してください。

[拡張モジュールレジスタの割り付け]



上記レジスタはCPU GR (復電時) で0クリアされます (2 α , S10miniはリセットでも同様となります)。

(*) S10miniモデルCは外部機器リンクモジュール非サポートとなります。

(6) I/O固定パターン (2α, S10miniのみ)

OSがI/Oメモリのイニシャルチェックに使用する固定パターンを格納したエリアです。このエリアのパターンが破壊された場合、CPUリセット時に全I/Oメモリ（演算命令ワークエリアも含む）をすべて0クリアします。

[I/O固定パターン]

(MSB) 2 ¹⁵															2 ⁰ (LSB)		
SWBA0	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	(=H55AA)
SWBB0	1	1	1	0	1	1	1	0	1	0	0	1	1	0	0	1	(=HEE99)

(7) CPUステータスレジスタ

現在のCPU状態を示すレジスタです。レジスタの各ビットは下記構成となっていて、このエリアをPSEなどで読み出すことにより、CPUの状態を知ることができます。

[CPUステータスのビット構成]

(MSB) 2 ¹⁵	2 ¹⁴	2 ¹³	2 ¹²	2 ¹¹	2 ¹⁰	2 ⁹	2 ⁸	2 ⁷	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰ (LSB)	
SWBF0	1/0	1/0	1/0	1/0	1/0	*	*	*	1/0	1/0	1/0	1/0	*	1/0	1/0	1/0
ビットNo.	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

*は将来拡張用

ビットNo.	リレー	各ビット内容	
		ION	0OFF
0	SBF0	STOP中	RUN中
1	SBF1	シミュレーション中	通常RUN中
2	SBF2	FORCE モード実行中	通常モード実行中
3	SBF3	プロテクト SW ON状態	プロテクト SW OFF状態
4	SBF4	リモートI/O 動作中	リモートI/O 停止中
5	SBF5	未使用	
6	SBF6		
7	SBF7		
8	SBF8	基本メモリ電池エラー警告	正常
9	SBF9	タイムアウトエラーステーションあり	なし
10	SBFA	FUSE断ステーションあり	なし
11	SBFB	オプションモジュールエラーあり	なし
12	SBFC	将来拡張用	
13	SBFD	GR時のみ0クリア	
14	SBFE	CPUダウン中表示	CPU正常運転中
15	SBFF	IPL用プログラム実行中	CPU OS実行中

** : 2α, S10miniのみ

2 演算ファンクション

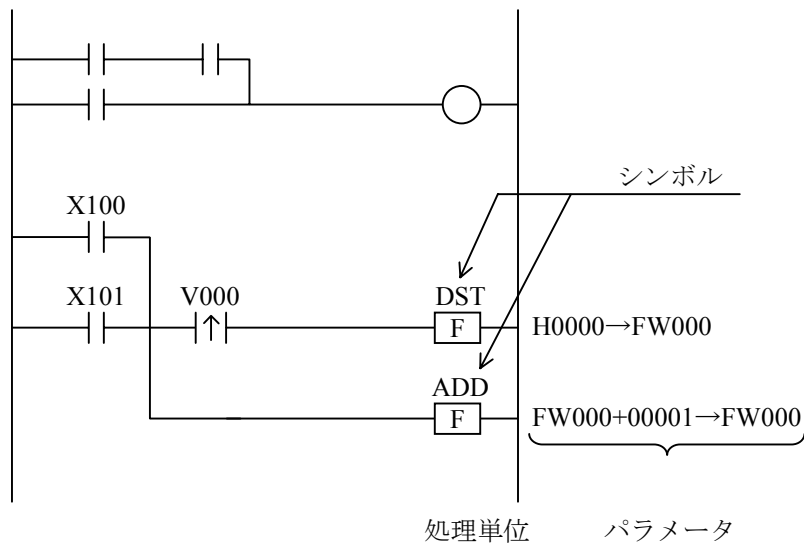
2 演算ファンクション

2.1 機能概要

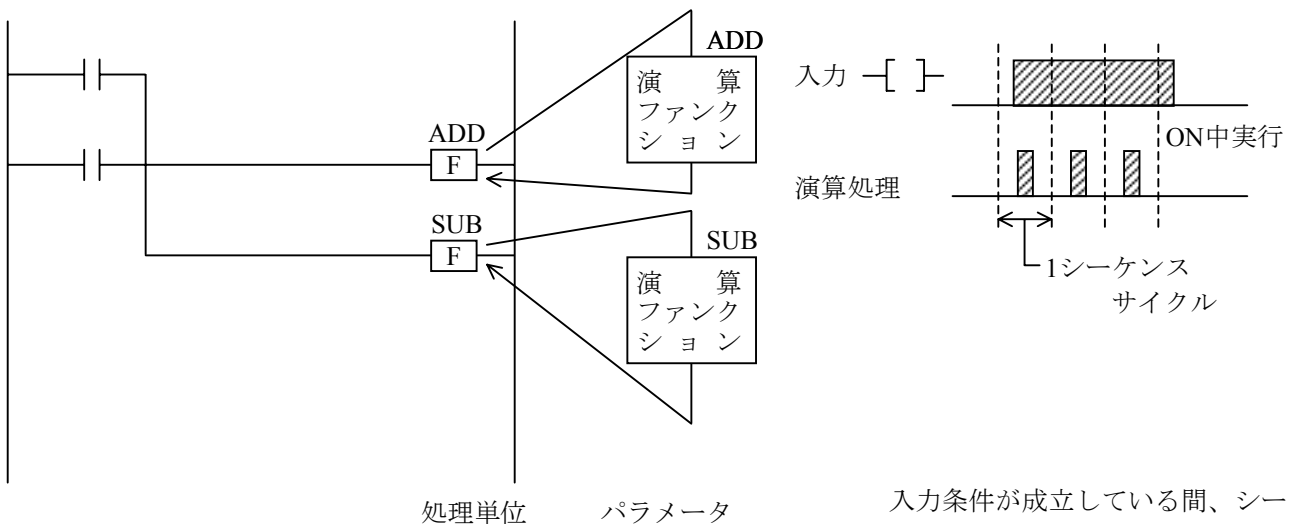
簡単な算術演算を行いたい場合は、演算ファンクションを使用すれば簡単にプログラムすることができます。

[演算ファンクションの動作]

<回路例>



<動作>



入力条件が成立している間、シーケンスサイクルごとに毎回演算を行います。

(1) パラメータ

演算ファンクションは処理内容に対応してシンボルを割り付けてあります。それぞれのシンボルに対してパラメータがあります。

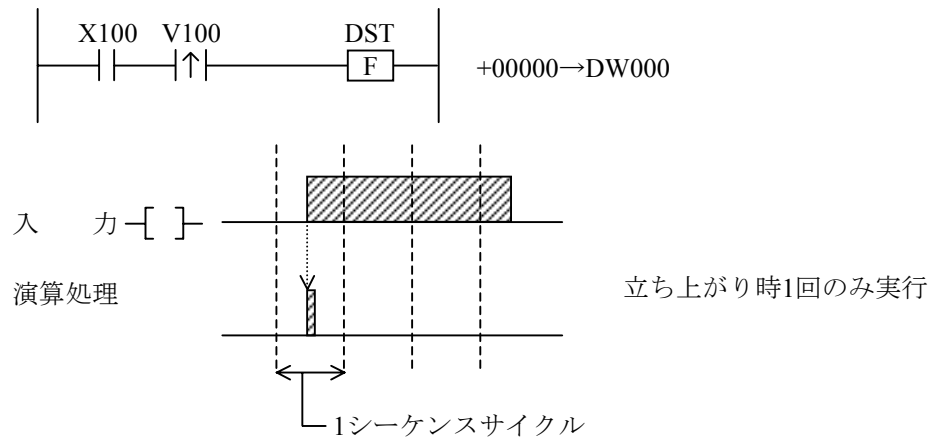
演算ファンクションで使用する定数データレジスタとワークレジスタは次のようになります。

名称	シンボル	ナンバ		使用目的
		2α, S10mini	4α	
ファンクション データレジスタ	DW	000~FFF	000~7FF	定数データエリア
ファンクション ワークレジスタ	FW	000~BFF		ワークエリア

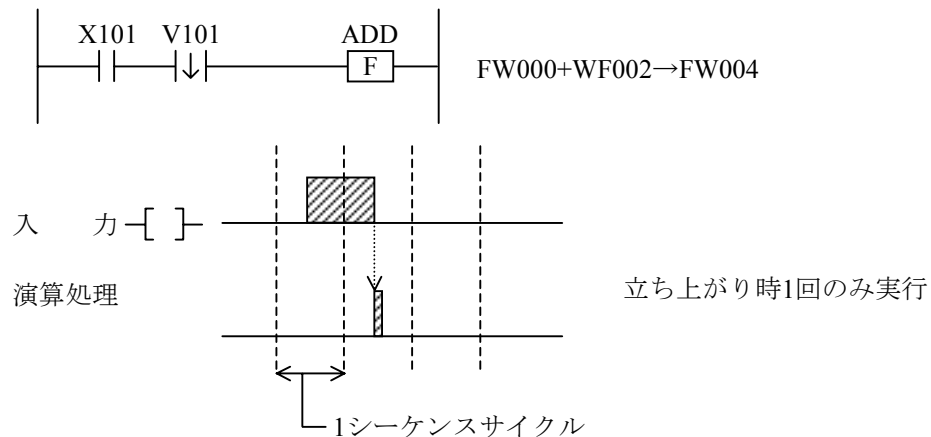
(2) 動作

演算ファンクションはコイルの励磁信号のON中、毎回起動されます。励磁信号の立ち上がり時または立ち下がり時に1回のみ起動させたい場合は、立ち上がり接点または立ち下がり接点と組み合わせます。

(例1) 立ち上がり接点との組み合わせ



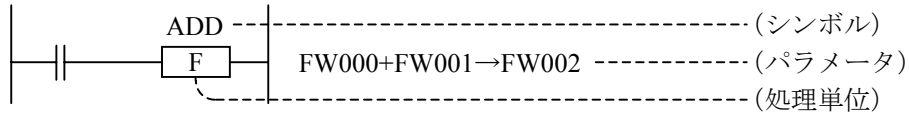
(例2) 立ち下がり接点との組み合わせ



2 演算ファンクション

2.2 機能仕様

(1) シンボル



シンボル：演算ファンクションの機能名称を示します。

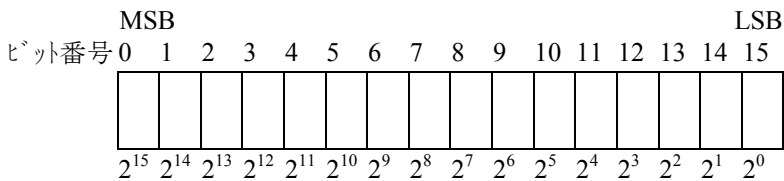
パラメータ：演算の対称となるアドレスや定数データを示します。

処理単位：演算の処理単位を示します。

(W : ワード
L : ロングワード)

(2) データフォーマット

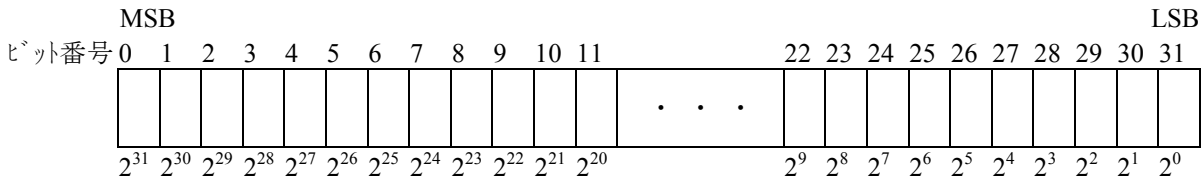
[ワード (XW000, YW000, RW000, … FW000, DW000, X000, Y000, R000, TC000, UC000, CC000, TS000)]



(注1) X000, R123などはワードとして取り扱います。

(注2) TC000, UC000, CC000, TS000, US000, CS000はワードサイズとして取り扱います。

[ロングワード (XL000, YL000, RL000, … FL000, DL000)]



演算ファンクションで使用されるデータは、符号付き16ビット単精度整数または符号付き32ビット倍精度整数です。16ビットで表されるデータをワードデータと呼び、32ビットで表されるデータをロングワードデータと呼びます。各ビットは上記のようにビット番号が付けられています。

ビット番号“0”のビットは符号ビットと呼ばれ、このビットが0のとき＝正、1のとき＝負の数を表します。したがって、演算ファンクションで取り扱うデータの範囲は次のようになります。

ワード：-32768 ≤ ワードデータ ≤ +32767 (10進数)
(H8000) (H7FFF) (16進数：Hで表します。)

ロングワード：-2147483648 ≤ ロングワードデータ ≤ +2147483647 (10進数)
(H80000000) (H7FFFFFFF) (16進数：Hで表します。)

(3) フラグの設定

演算ファンクションは、演算結果に従い各種フラグを設定します。以下にフラグの種類とフラグが設定されるエリアおよびフラグが設定される条件を示します。

X : イクステンド	S000	X
E : エラー	S001	E
P : ポジティブ	S002	P
N : ネガティブ	S003	N
Z : 0	S004	Z
V : オーバーフロー	S005	V
F.U. : 将来拡張用	S006	F.U.
	S007	F.U.

No.	種別	フラグ						特殊な条件	
		X	E	P	N	Z	V	ワード時	ロングワード時
1	ADD SUB INC DEC MUL	-	-	-	-	-	↕	V : 結果<-32768または32767<結果 のとき1、それ以外0	V : 結果<-2147483648または2147483647 <結果のとき1、それ以外0
2	DIV MOD	-	↕	-	-	-	↕	E : 除数=0のとき1、それ以外0 V : 商=32768のとき1、それ以外0	E : 除数=0のとき1、それ以外0 V : 商=2147483648のとき1、それ以外0
3	SCL	-	↕	-	-	-	↕	E : 除数=0のとき1、それ以外0 V : 結果<-32768または32767<結果の とき1、それ以外0	
4	TST	-	-	↕	↕	↕	-	P : データ>0のとき1、それ以外0 N : データ<0のとき1、それ以外0 Z : データ=0のとき1、それ以外0	
5	BTD	-	↕	-	-	-	↕	E : データ<0のとき1、それ以外0 V : データ>9999のとき1、それ以外0	E : データ<0のとき1、それ以外0 V : データ>99999999のとき、それ以外0
6	DTB	-	↕	-	-	-	-	E : あるディジット (4ビット) でHA~HFを検出時1、それ以外0	
7	APB AUB	-	↕	-	-	-	-	E : H30~H39, H41~H46以外のデータ 検出時1、それ以外0	
8	DTS	-	-	-	-	-	↕		V : データ<-32768または32767< データのとき1、それ以外0
9	ABS NEG	-	-	-	-	-	↕	V : データ=-32768のとき1、それ以外0	V : データ=-2147483648のとき1、それ以外0
10	ECD	-	↕	-	-	-	-	E : データ=0のとき1、それ以外0	
11	ASL	-	-	-	-	-	↕	V : シフト操作中に符号ビットが1回でも変化すれば1、それ以外0	
12	LIM	-	↕	-	-	-	-	E : 上限値<下限値のとき1、それ以外0	
13	BND ZON	-	↕	-	-	-	↕	E : 上限値<下限値のとき1、それ以外0 V : 結果<-32768または32767<結果 のとき1、それ以外0	E : 上限値<下限値のとき1、それ以外0 V : 結果<-2147483648または2147483647 <結果のとき1、それ以外0
14	上記以外	-	-	-	-	-	-	すべて保持	

- : 演算実行直前の値を保持 ↕ : 特殊な条件を参照

2 演算ファンクション

2.3 演算ファンクション入力

演算ファンクションの入力は、演算ファンクション入力ダイアグラムで行います。
 入力方法は、シンボルとパラメータはスペース（）で区切り、パラメータ間はカンマ（`,`）で区切り
 ます。

シンボル パラメータ, パラメータ, パラメータ

Enter

(1) ニーモニック入力の場合

設定可能エリア	入力例	備考
I/Oエリア（ビット）	X000	ビット指定
I/Oエリア（ワード）	YW000	Wはワードを示します。
ファンクションワークレジスタエリア	FW025	ワークエリア
ファンクションデータレジスタエリア	DW050	定数データエリア
T, U, C設定値エリア	TS003	Sは設定値を示します。
T, U, C計数值エリア	UC007	Cは計数值を示します。
高速I/O（ワード）エリア	IW000	

- I/O エリア：X, Y, R, K, T, U, C, G, N, P, E, V, Z, S
- ナンバは3桁入力で行います。
- I/O エリアで複数のエリアを持つもの（T, U, Cのセットコイルと接点のエリア、Nのセットコイルと接点エリアなど）を設定した場合は、接点エリアをアクセスします。

例：TW010→T010～T01Fの接点エリアを示します。

(2) データ入力の場合

(a) 10進数入力

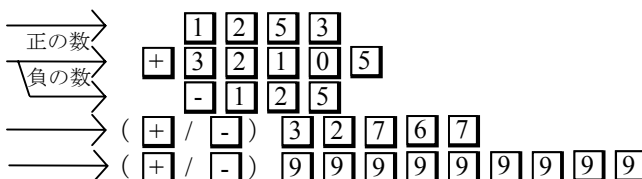
(i) 直接数値を入力（正の10進数）

(ii) +/-を入力後数値入力

(iii) 入力桁数は

ワードの場合：最大5桁

ロングワードの場合：最大9桁



(b) 16進数入力

(i) 'H'を入力後数値入力

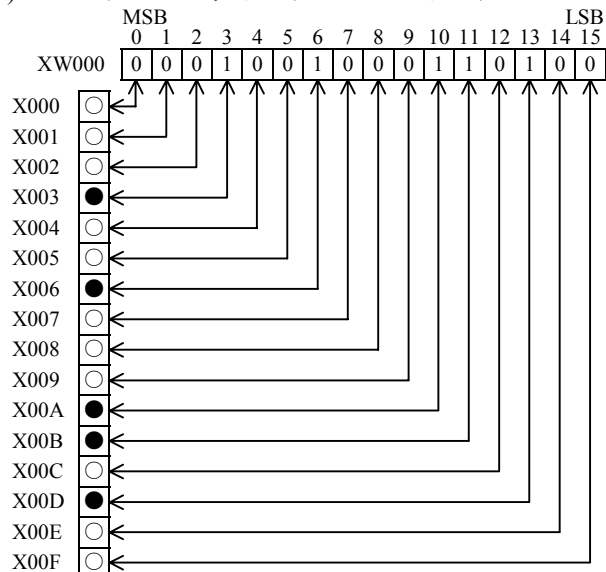
(ii) 入力桁数は

ワードの場合：最大4桁

ロングワードの場合：最大8桁



(3) I/Oエリアのビットエリアとワードエリアについて



XW000⇔X000～X00F

XW010⇔X010～X01F

⋮

⋮

⋮

XW100⇔X100～X10F

⋮

⋮

⋮

上記のように対応しています。

(● : ON)
 (○ : OFF)

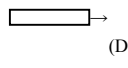
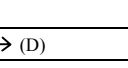
2.4 演算ファンクション一覧表

(1/7)

大分類	分類	パラメータの型	処理内容	処理単位	フラグ						ステップ数	処理時間 (ms)			ページ
					X	E	P	N	Z	V		2α	2αE, 2αH(f)	4α	
算術演算命令	加算	ワード	$(S) + (D) \rightarrow (R)$	ワード							4	0.06	0.036	0.6	42
		ワード定数	$(S) + n \rightarrow (R)$	ワード	-	-	-	-	-	◇	4	0.06	0.036	0.5	
		ロング	$(S)L + (D)L \rightarrow (R)L$	ロングワード							4	0.06	0.036	0.8	
	減算	ワード	$(S) - (D) \rightarrow (R)$	ワード							4	0.06	0.036	0.6	44
		ワード定数	$(S) - n \rightarrow (R)$	ワード	-	-	-	-	-	◇	4	0.06	0.036	0.5	
		ロング	$(S)L - (D)L \rightarrow (R)L$	ロングワード							4	0.06	0.036	0.8	
	+1	ワード	$(S) + 1 \rightarrow (S)$	ワード							2	0.04	0.024	0.5	46
		ロング	$(S)L + 1 \rightarrow (S)L$	ロングワード							2	0.04	0.024	0.6	
	-1	ワード	$(S) - 1 \rightarrow (S)$	ワード							2	0.04	0.024	0.5	48
		ロング	$(S)L - 1 \rightarrow (S)L$	ロングワード							2	0.04	0.024	0.6	
	乗算	ワード	$(S) \times (D) \rightarrow (R)$	ワード							4	0.08	0.048	0.9	50
		ワード定数	$(S) \times n \rightarrow (R)$	ワード	-	-	-	-	-	◇	4	0.08	0.048	0.8	
		ロング	$(S)L \times (D)L \rightarrow (R)L$	ロングワード							4	0.13	0.078	2.3	
	除算	ワード	$(S) \div (D) \rightarrow (R)$ (商)	ワード							4	0.09	0.054	1.0	52
		ワード定数	$(S) \div n \rightarrow (R)$ (商)	ワード	-	◇	-	-	-	◇	4	0.09	0.054	0.9	
		ロング	$(S)L \div (D)L \rightarrow (R)L$ (商)	ロングワード							4	0.38	0.228	3.0	
	剰余算	ワード	$(S) \% (D) \rightarrow (R)$ (剰余)	ワード							4	0.11	0.066	1.0	54
		ワード定数	$(S) \% n \rightarrow (R)$ (剰余)	ワード	-	◇	-	-	-	◇	4	0.11	0.066	0.9	
ロング		$(S)L \% (D)L \rightarrow (R)L$ (剰余)	ロングワード							4	0.46	0.276	3.0		
スケール変換	ワード	$(S) \times (D1) \div (D2) \rightarrow (R)$	ワード							5	0.12	0.072	3.1	56	
	ワード定数	$(S) \times n1 \div n2 \rightarrow (R)$	ワード	-	◇	-	-	-	◇	5	0.11	0.066	2.9		
論理演算命令	論理積	ワード	$(S) \wedge (D) \rightarrow (R)$	ワード							4	0.05	0.03	0.6	58
		ワード定数	$(S) \wedge n \rightarrow (R)$	ワード	-	-	-	-	-	-	4	0.05	0.03	0.5	
		ロング	$(S)L \wedge (D)L \rightarrow (R)L$	ロングワード							4	0.05	0.03	0.8	
	論理和	ワード	$(S) \vee (D) \rightarrow (R)$	ワード							4	0.05	0.03	0.6	60
		ワード定数	$(S) \vee n \rightarrow (R)$	ワード	-	-	-	-	-	-	4	0.05	0.03	0.5	
		ロング	$(S)L \vee (D)L \rightarrow (R)L$	ロングワード							4	0.05	0.03	0.8	
排他的論理和	ワード	$(S) \oplus (D) \rightarrow (R)$	ワード							4	0.05	0.03	0.6	62	
	ワード定数	$(S) \oplus n \rightarrow (R)$	ワード	-	-	-	-	-	-	4	0.05	0.03	0.5		
	ロング	$(S)L \oplus (D)L \rightarrow (R)L$	ロングワード							4	0.05	0.03	0.8		
否定	ワード	$(S) \rightarrow (R)$	ワード							3	0.04	0.024	0.4	64	
	ロング	$(S)L \rightarrow (R)L$	ロングワード							3	0.04	0.024	0.6		
比較演算命令	=	ワード	$(S) = (D)$ のとき $1 \rightarrow (R)$ $(S) \neq (D)$ のとき $0 \rightarrow (R)$	ワード							4	0.05	0.03	0.6	66
		ワード定数	$(S) = n$ のとき $1 \rightarrow (R)$ $(S) \neq n$ のとき $0 \rightarrow (R)$	ワード	-	-	-	-	-	-	4	0.05	0.03	0.5	
		ロング	$(S)L = (D)L$ のとき $1 \rightarrow (R)$ $(S)L \neq (D)L$ のとき $0 \rightarrow (R)$	ロングワード							4	0.06	0.036	0.7	
	≠	ワード	$(S) = (D)$ のとき $0 \rightarrow (R)$ $(S) \neq (D)$ のとき $1 \rightarrow (R)$	ワード							4	0.05	0.03	0.6	68
		ワード定数	$(S) = n$ のとき $0 \rightarrow (R)$ $(S) \neq n$ のとき $1 \rightarrow (R)$	ワード	-	-	-	-	-	-	4	0.05	0.03	0.5	
		ロング	$(S)L = (D)L$ のとき $0 \rightarrow (R)$ $(S)L \neq (D)L$ のとき $1 \rightarrow (R)$	ロングワード							4	0.06	0.036	0.7	
>	ワード	$(S) > (D)$ のとき $1 \rightarrow (R)$ $(S) \leq (D)$ のとき $0 \rightarrow (R)$	ワード							4	0.05	0.03	0.06	70	
	ワード定数	$(S) > n$ のとき $1 \rightarrow (R)$ $(S) \leq n$ のとき $0 \rightarrow (R)$	ワード	-	-	-	-	-	-	4	0.05	0.03	0.5		
	ロングワード	$(S)L > (D)L$ のとき $1 \rightarrow (R)$ $(S)L \leq (D)L$ のとき $0 \rightarrow (R)$	ロングワード							4	0.06	0.036	0.7		

2 演算ファンクション

(2/7)

大分類	分類	パラメータの型	処理内容	処理単位	フラグ						ステップ数	処理時間 (ms)			ページ
					X	E	P	N	Z	V		2α	2αE, 2αH(f)	4α	
比較演算命令	≥	ワード	(S) ≥ (D) のとき 1 → (R) (S) < (D) のとき 0 → (R)	ワード							4	0.05	0.03	0.6	72
		ワード定数	(S) ≥ n のとき 1 → (R) (S) < n のとき 0 → (R)	ワード	-	-	-	-	-	-	4	0.05	0.03	0.5	
		ロングワード	(S)L ≥ (D)L のとき 1 → (R) (S)L < (D)L のとき 0 → (R)	ロングワード							4	0.06	0.036	0.7	
	<	ワード	(S) < (D) のとき 1 → (R) (S) ≥ (D) のとき 0 → (R)	ワード							4	0.05	0.03	0.6	74
		ワード定数	(S) < n のとき 1 → (R) (S) ≥ n のとき 0 → (R)	ワード	-	-	-	-	-	-	4	0.05	0.03	0.5	
		ロングワード	(S)L < (D)L のとき 1 → (R) (S)L ≥ (D)L のとき 0 → (R)	ロングワード							4	0.06	0.036	0.7	
	≤	ワード	(S) ≤ (D) のとき 1 → (R) (S) > (D) のとき 0 → (R)	ワード							4	0.05	0.03	0.6	76
		ワード定数	(S) ≤ n のとき 1 → (R) (S) > n のとき 0 → (R)	ワード	-	-	-	-	-	-	4	0.05	0.03	0.5	
		ロングワード	(S)L ≤ (D)L のとき 1 → (R) (S)L > (D)L のとき 0 → (R)	ロングワード							4	0.06	0.036	0.7	
	テスト	ワード	(S) をテストし、P, N, Zフラグをセットします。	ワード			↕	↕	↕		2	0.04	0.024	0.4	78
		ロングワード	(S)L をテストし、P, N, Zフラグをセットします。	ロングワード			↕	↕	↕		2	0.04	0.024	0.5	
	データ転送命令	転送	ワード	(S) → (D)	ワード							3	0.04	0.024	0.4
ロングワード			(S)L → (D)L	ワード							3	0.04	0.024	0.6	
一括転送		ワード	(S)  (D)	ワード							4	0.06+ 0.007n	0.036 0.005n	0.3+ 0.4n	82
		ロングワード	または  (D)L	ロングワード							4	0.06+ 0.008n	0.036 0.005n	0.3+ 0.4n	
インダイレクト転送		ワード	i(S)  (D)	ワード							4	0.05+ 0.007n	0.03 0.005n	0.6+ 0.3n	84
		ワード	(S)  i(D)	ワード							4	0.05+ 0.007n	0.03 0.005n	0.6+ 0.3n	
同一データ一括転送		ワード	(S)  (D)	ワード							4	0.05+ 0.004n	0.03 0.003n	0.3+ 0.2n	88
		ワード定数	または  (D)	ワード							4	0.05+ 0.004n	0.03 0.003n	0.3+ 0.2n	
		ロングワード	または (S)L 	ロングワード							4	0.06+ 0.004n	0.036 0.003n	0.4+ 0.2n	
交換		ワード	(S) ↔ (D)	ワード							3	0.04	0.024	0.7	90
		ロングワード	(S)L ↔ (D)L	ロングワード							3	0.04	0.024	1.0	
FIFO書き込み		ワード	FIFO (S) → テーブル 	ワード							3	0.06	0.036	1.2	92
	ワード	FIFO テーブル  → (D)	ワード							3	0.07+ 0.006n	0.042 0.004n	1.3+ 0.2n		
データセット	ワード定数	n → (D)	ワード							3	0.04	0.024	0.3	96	
	ロング	nL → (D)L	ロングワード							4	0.05	0.03	0.4		
	ロング	S(アドレスデータ) → (D)L	ロングワード							3	0.04	0.024	0.4		
	ワード	S  D	ワード							5	0.07+ 0.004m	0.042 0.003m	0.5+ 0.1m		
サーチ	ワード	一致ナンバ n → (R)	ワード							5	0.08+ 0.004m	0.048 0.003m	0.5+ 0.2m	100	
	ロング		ロングワード							5	0.08+ 0.004m	0.048 0.003m	0.5+ 0.2m		

2 演算ファンクション

(3/7)

大分類	分類	パラメータの型	処理内容	処理単位	フラグ						ステップ数	処理時間 (ms)			ページ				
					X	E	P	N	Z	V		2α	2αE, 2αH(f)	4α					
データ変換命令	BIN ↓ BCD	ワード ロング	BIN→BCD (S) → (R)	ワード ロングワード	-	◇	-	-	-	◇	3	0.26	0.156	1.6	102				
	BCD ↓ BIN	ワード ロング	BCD→BIN (S) → (R)	ワード ロングワード	-	◇	-	-	-	-	3	0.21	0.126	1.3		104			
	BIN ↓ 7SEG	ワード ワード ロング	BIN→7seg (S) → (R, R+1)	ワード ロングワード ワード	-	-	-	-	-	-	3	0.11	0.066	0.6	106				
			BIN→7seg n → (R, R+1)								3	0.21	0.126	0.5					
			BIN→7seg (S)L → {R(L), R+2(L)}								3	0.11	0.066	0.9					
	BIN ↓ ASCII	ワード ワード	BIN→ASCII (パックモード) (S) → (R, R+1)	ワード ワード	-	-	-	-	-	-	3	0.13	0.078	0.6	108				
			BIN→ASCII (アンパックモード) (S) → (R, R+1, R+2, R+3)	ワード	-	-	-	-	-	-	3	0.14	0.084	0.8	110				
	ASCII ↓ BIN	ワード ワード	ASCII→BIN (パックモード) (S, S+1) → (R)	ワード ワード	-	◇	-	-	-	-	3	0.13	0.078	0.6	112				
			ASCII→BIN (アンパックモード) (S, S+1, S+2, S+3) → (R)	ワード	-	◇	-	-	-	-	3	0.13	0.078	0.8	114				
	SINGLE ↓ DOUBLE	ワード	(S) → (R)L	ワード	-	-	-	-	-	-	3	0.04	0.024	0.5	116				
	DOUBLE ↓ SINGLE	ロング	(S)L → (R)	ロングワード	-	-	-	-	-	◇	3	0.05	0.03	0.6	118				
	絶対値	ワード	(S) → (R)	ワード	-	-	-	-	-	◇	3	0.05	0.03	0.5	120				
		ロング	(S)L → (R)L	ロングワード	-	-	-	-	-	◇	3	0.05	0.03	0.6					
	+/-	ワード	-(S) → (R)	ワード	-	-	-	-	-	◇	3	0.05	0.03	0.5	122				
		ロング	-(S)L → (R)L	ロングワード	-	-	-	-	-	◇	3	0.05	0.03	0.6					
	デコード	ワード	(S) <table style="display: inline-table; border: 1px solid black; padding: 2px;"><tr><td style="padding: 0 5px;">n</td></tr></table> → (R) <table style="display: inline-table; border: 1px solid black; padding: 2px;"><tr><td style="padding: 0 5px;">0~0</td><td style="padding: 0 5px;"> </td><td style="padding: 0 5px;">0~0</td></tr></table> (15LSB)	n	0~0		0~0	ワード	-	-	-	-	-	-	3	0.05	0.03	0.4	124
		n																	
	0~0		0~0																
	ロング	(S) <table style="display: inline-table; border: 1px solid black; padding: 2px;"><tr><td style="padding: 0 5px;">n</td></tr></table> → (R) <table style="display: inline-table; border: 1px solid black; padding: 2px;"><tr><td style="padding: 0 5px;">L0</td><td style="padding: 0 5px;"> </td><td style="padding: 0 5px;">n</td><td style="padding: 0 5px;"> </td><td style="padding: 0 5px;">0~0</td></tr></table> (31LSB)	n	L0		n		0~0	ロングワード	-	-	-	-	-	3	0.05	0.03	0.5	
	n																		
L0		n		0~0															
エンコード	ワード	(S) <table style="display: inline-table; border: 1px solid black; padding: 2px;"><tr><td style="padding: 0 5px;">0~0</td><td style="padding: 0 5px;"> </td><td style="padding: 0 5px;">X~X</td></tr></table> → (R) <table style="display: inline-table; border: 1px solid black; padding: 2px;"><tr><td style="padding: 0 5px;">n</td></tr></table> (x~x:任意)	0~0		X~X	n	ワード	-	◇	-	-	-	-	-	3	0.11	0.066	0.6	126
	0~0		X~X																
n																			
ロング	(S) <table style="display: inline-table; border: 1px solid black; padding: 2px;"><tr><td style="padding: 0 5px;">L0</td><td style="padding: 0 5px;"> </td><td style="padding: 0 5px;">n</td><td style="padding: 0 5px;"> </td><td style="padding: 0 5px;">X~X</td></tr></table> → (R) <table style="display: inline-table; border: 1px solid black; padding: 2px;"><tr><td style="padding: 0 5px;">n</td></tr></table> (x~x:任意)	L0		n		X~X	n	ロングワード	-	◇	-	-	-	-	-	3	0.11	0.066	0.7
L0		n		X~X															
n																			

2 演算ファンクション

(4/7)

大分類	分類	パラメータの型	処理内容	処理単位	フラグ						ステップ数	処理時間 (ms)			ページ
					X	E	P	N	Z	V		2α	2αE, 2αH(f)	4α	
シフト命令	論理右シフト	ワード		ワード							4	0.06	0.036	0.6	128
		ワード定数		ワード							4	0.06	0.036	0.5	
		ロング		ロングワード							4	0.06	0.036	1.0	
		ロング定数		ロングワード							4	0.06	0.036	0.9	
	論理左シフト	ワード		ワード							4	0.06	0.036	0.6	130
		ワード定数		ワード							4	0.06	0.036	0.5	
		ロング		ロングワード							4	0.06	0.036	1.0	
		ロング定数		ロングワード							4	0.06	0.036	0.9	
算術右シフト	ワード		ワード							4	0.06	0.036	0.6	132	
	ワード定数		ワード							4	0.06	0.036	0.5		

大分類	分類	パラメータの型	処理内容	処理単位	フラグ						ステップ数	処理時間 (ms)			ページ
					X	E	P	N	Z	V		2α	2αE, 2αH(f)	4α	
シフト命令	算術右シフト	ロング		ロングワード							4	0.06	0.036	1.0	132
		ロング定数		ロングワード							4	0.06	0.036	0.9	
	算術左シフト	ワード		ワード							4	0.07	0.042	0.8	134
		ワード定数		ワード							4	0.07	0.042	0.7	
ロング		ロングワード						◇	4	0.07	0.042	1.3			
ロング定数		ロングワード						◇	4	0.07	0.042	1.2			
ローテート命令	左回転	ワード		ワード							4	0.06	0.036	0.6	136
		ワード定数		ワード							4	0.06	0.036	0.5	
	ロング		ロングワード							4	0.06	0.036	1.1		
	ロング		ロングワード							4	0.06	0.036	1.0		

2 演算ファンクション

(6/7)

大分類	分類	パラメータの型	処理内容	処理単位	フラグ						ステップ数	処理時間(ms)			ページ
					X	E	P	N	Z	V		2α	2αE, 2αH(f)	4α	
ローテイト命令	右回転	ワード	右回転 $(S) \xrightarrow{\quad} (R)$ 	ワード	-	-	-	-	-	-	4	0.06	0.036	0.6	138
		ワード定数	右回転 $(S) \xrightarrow{\quad} (R)$ 	ワード	-	-	-	-	-	-	4	0.06	0.036	0.5	
		ロング	右回転 $(S)L \xrightarrow{\quad} (R)L$ 	ロングワード	-	-	-	-	-	-	4	0.06	0.036	1.1	
		ロング	右回転 $(S)L \xrightarrow{\quad} (R)L$ 	ロングワード	-	-	-	-	-	-	4	0.06	0.036	1.0	
関連処理命令	LIMITE R	ワード	$(D1) < (S)$ のとき $(D1) \rightarrow (R)$ $(D2) \leq (S) \leq (D1)$ のとき $(S) \rightarrow (R)$ $(S) < (D2)$ のとき $(D2) \rightarrow (R)$	ワード	-	◇	-	-	-	-	5	0.07	0.042	0.7	140
		ワード定数	$n1 < (S)$ のとき $n1 \rightarrow (R)$ $n2 \leq (S) \leq n1$ のとき $(S) \rightarrow (R)$ $(S) < n2$ のとき $n2 \rightarrow (R)$	ワード	-	◇	-	-	-	-	5	0.07	0.042	0.5	
		ロング	$(D1)L < (S)L$ のとき $(D1)L \rightarrow (R)L$ $(D2)L \leq (S)L \leq (D1)L$ のとき $(S)L \rightarrow (R)L$ $(S)L < (D2)L$ のとき $(D2)L \rightarrow (R)L$	ロングワード	-	◇	-	-	-	-	5	0.08	0.048	1.0	
	DEAND	ワード	$(D1) < (S)$ のとき $(S) - (D1) \rightarrow (R)$ $(D2) \leq (S) \leq (D1)$ のとき $0 \rightarrow (R)$ $(S) < (D2)$ のとき $(S) - (D2) \rightarrow (R)$	ワード	-	◇	-	-	-	◇	5	0.08	0.048	0.7	142
		ワード定数	$n1 < (S)$ のとき $(S) - n1 \rightarrow (R)$ $n2 \leq (S) \leq n1$ のとき $0 \rightarrow (R)$ $(S) < n2$ のとき $(S) - n2 \rightarrow (R)$	ワード	-	◇	-	-	-	◇	5	0.08	0.048	0.5	
		ロング	$(D1)L < (S)L$ のとき $(S)L - (D1)L \rightarrow (R)L$ $(D2)L \leq (S)L \leq (D1)L$ のとき $0 \rightarrow (R)L$ $(S)L < (D2)L$ のとき $(S)L - (D2)L \rightarrow (R)L$	ロングワード	-	◇	-	-	-	◇	5	0.08	0.048	1.0	
	DEANDZONE	ワード	$(S) > 0$ のとき $(S) + (D1) \rightarrow (R)$ $(S) = 0$ のとき $0 \rightarrow (R)$ $(S) < 0$ のとき $(S) + (D2) \rightarrow (R)$	ワード	-	◇	-	-	-	◇	5	0.08	0.048	0.7	144
		ワード定数	$(S) > 0$ のとき $(S) + n1 \rightarrow (R)$ $(S) = 0$ のとき $0 \rightarrow (R)$ $(S) < 0$ のとき $(S) + n2 \rightarrow (R)$	ワード	-	◇	-	-	-	◇	5	0.08	0.048	0.5	
		ロング	$(S) > 0$ のとき $(S)L + (D1)L \rightarrow (R)L$ $(S) = 0$ のとき $0 \rightarrow (R)L$ $(S) < 0$ のとき $(S)L + (D2)L \rightarrow (R)L$	ロングワード	-	◇	-	-	-	◇	5	0.08	0.048	1.0	
ルート	ワード	$(S) \geq 0$ のとき $\sqrt{(S)} \rightarrow (R)$ $(S) < 0$ のとき $0 \rightarrow (R)$	ワード	-	-	-	-	-	-	3	0.17	0.102	0.8	146	
	ロング	$(S)L \geq 0$ のとき $\sqrt{(S)L} \rightarrow (R)L$ $(S)L < 0$ のとき $0 \rightarrow (R)L$	ロングワード	-	-	-	-	-	-	3	0.20	0.12	2.2		

2 演算ファンクション

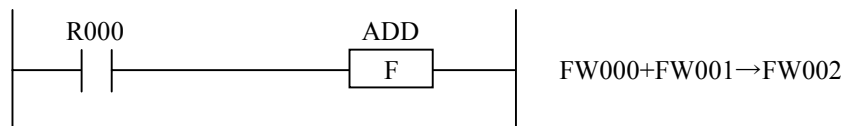
(77)

大分類	分類	パラメータの型	処理内容	処理単位	フラグ						ステップ数	処理時間 (ms)			ページ
					X	E	P	N	Z	V		2α	2αE, 2αH(f)	4α	
関数処理命令	M	ワード	(S)≥(D)のとき (S)→(R) (S)<(D)のとき (D)→(R)	ワード							4	0.05	0.03	0.6	148
	A	ワード定数	(S)≥nのとき (S)→(R) (S)<nのとき n→(R)	ワード	-	-	-	-	-	-	4	0.05	0.03	0.5	
	X	ロング	(S)L≥(D)Lのとき(S)L→(R)L (S)L<(D)Lのとき(D)L→(R)L	ロングワード							4	0.05	0.03	0.8	
	M	ワード	(S)≤(D)のとき (S)→(R) (S)>(D)のとき (D)→(R)	ワード							4	0.05	0.03	0.6	150
	I	ワード定数	(S)≤nのとき (S)→(R) (S)>nのとき n→(R)	ワード	-	-	-	-	-	-	4	0.05	0.03	0.5	
	N	ロング	(S)L≤(D)Lのとき(S)L→(R)L (S)L>(D)Lのとき(D)L→(R)L	ロングワード							4	0.05	0.03	0.8	
特殊命令	ク	-	Xのエリアをクリアします。	-	-	-	-	-	-	-	1	0.77	0.462	2.3	152
		-	Yのエリアをクリアします。	-	-	-	-	-	-	-	1	0.77	0.462	2.3	
		-	Gのエリアをクリアします。	-	-	-	-	-	-	-	1	0.77	0.462	2.3	
		-	Rのエリアをクリアします。	-	-	-	-	-	-	-	1	0.77	0.462	2.3	
		-	Kのエリアをクリアします。	-	-	-	-	-	-	-	1	0.77	0.462	2.3	
	リア	-	Tのエリアおよび計数値エリアをクリアします。	-	-	-	-	-	-	-	1	2.21	1.326	7.0	
		-	Uのエリアおよび計数値エリアをクリアします。	-	-	-	-	-	-	-	1	1.50	0.9	4.0	
		-	Cのエリアおよび計数値エリアをクリアします。	-	-	-	-	-	-	-	1	1.50	0.9	4.0	
		-	Vのエリアをクリアします。	-	-	-	-	-	-	-	1	0.77	0.462	2.3	
		-	Eのエリアをクリアします。	-	-	-	-	-	-	-	1	0.77	0.462	2.3	
-	ファンクションフラグをクリアします。	-	0	0	0	0	0	0	1	0.007	0.005	0.4			

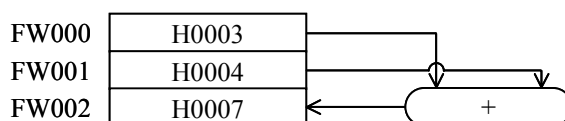
ADD 加算 : ADD

機能	ソースの内容とデスティネーションの内容またはイミディエートデータの加算処理を行い、結果をリザルトへ格納します。									
パラメータ の型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	○	○	×	-	-	-	-	-	↑ ↓
パラメータ	ワード	定数なし	ADD F	S.W+D.W→R.W						
		定数あり	ADD F	S.W+n.W→R.W						
	ロング	定数なし	ADD F	S.L+D.L→R.L						
		定数あり								
S : ソース格納アドレス D : デスティネーション格納アドレス n : イミディエートデータ R : リザルト (演算結果) を格納するアドレス W : ワード変数 L : ロング変数										
処理内容	ワード	(S) W+ (D) W→ (R) W								
	ワード定数	(S) W+ n W→ (R) W								
	ロング	(S) L+ (D) L→ (R) L								
入力手順	(Shift + F) A D D □ パラメータ, パラメータ, パラメータ Enter									
フラグの 設定	V : (ワードのとき) (R) < -32768または32767 < (R) のとき1、それ以外0 (ロングワードのとき) (R) < -2147483648または2147483647 < (R) のとき1、 それ以外0 他 : 保持									
注意事項	オーバーフロー発生時、リザルトには下記フルスケール値が格納されます。									
		ワード	ロングワード							
	正のオーバーフロー時	H7FFF	H7FFFFFFF							
	負のオーバーフロー時	H8000	H80000000							

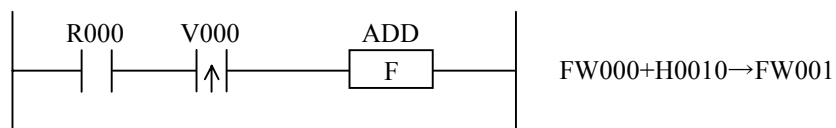
[ワード]



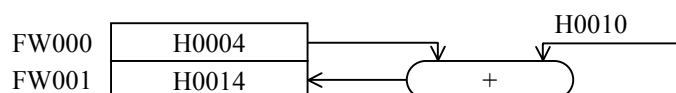
入力条件R000がONのとき、FW000の内容とFW001の内容を加算し、その結果をFW002へ格納します。



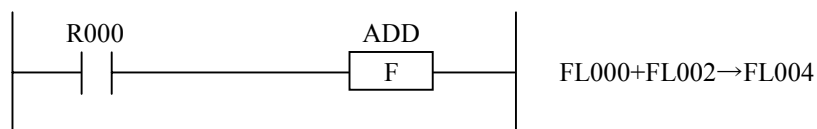
[ワード定数]



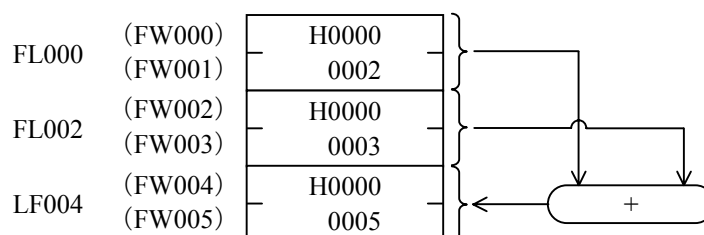
入力条件R000がOFF→ONの変化時、1回のみFW000の内容とイミディエートデータH0010を加算し、その結果をFW001へ格納します。












[ロング]



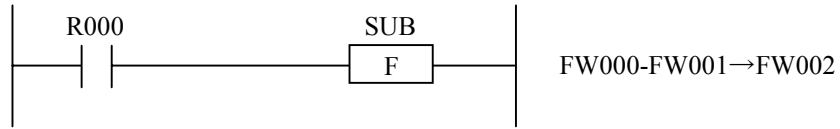
入力条件R000がONのとき、FL000の内容とFL002の内容を加算し、その結果をFL004へ格納します。



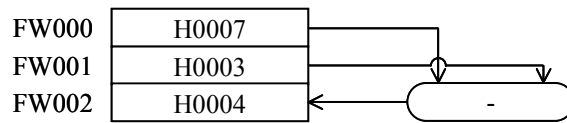
SUB 減算 : SUBTRACT

機能	ソースの内容とデスティネーションの内容またはイミディエートデータの減算処理を行い、結果をリザルトへ格納します。																		
パラメータの型	ワード		ロング		フラグ														
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V									
	○	○	○	×	-	-	-	-	-	↑↓									
パラメータ	ワード	定数なし		SW-DS→RW															
		定数あり		SW-nW→RW															
	ロング	定数なし		SL-nL→RL															
		定数あり																	
S : ソース格納アドレス					W : ワード														
D : デスティネーション格納アドレス					L : ロング														
n : イミディエートデータ																			
R : リザルト (演算結果) を格納するアドレス																			
処理内容	ワード	(S) W- (D) W→ (R) W																	
	ワード定数	(S) W- n W→ (R) W																	
	ロング	(S) L- (D) L→ (R) L																	
入力手順	 +     <input type="text"/> パラメータ, パラメータ, パラメータ 																		
フラグの設定	V : (ワードのとき) (R) < -32768または32767 < (R) のとき1、それ以外0 (ロングワードのとき) (R) < -2147483648または2147483647 < (R) のとき1、それ以外0 他 : 保持																		
注意事項	オーバーフロー発生時、リザルトには下記フルスケール値が格納されます。 <table border="1" data-bbox="402 1796 1161 1944"> <tr> <td></td> <td>ワード</td> <td>ロングワード</td> </tr> <tr> <td>正のオーバーフロー時</td> <td>H7FFF</td> <td>H7FFFFFFF</td> </tr> <tr> <td>負のオーバーフロー時</td> <td>H8000</td> <td>H80000000</td> </tr> </table>											ワード	ロングワード	正のオーバーフロー時	H7FFF	H7FFFFFFF	負のオーバーフロー時	H8000	H80000000
	ワード	ロングワード																	
正のオーバーフロー時	H7FFF	H7FFFFFFF																	
負のオーバーフロー時	H8000	H80000000																	

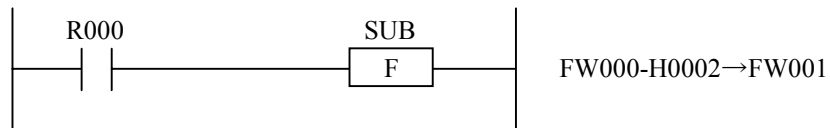
[ワード]



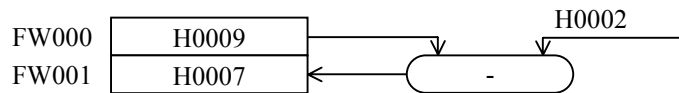
入力条件R000がONのとき、FW000の内容からFW001の内容を減算し、その結果をFW002へ格納します。



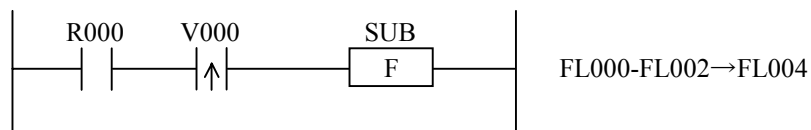
[ワード定数]



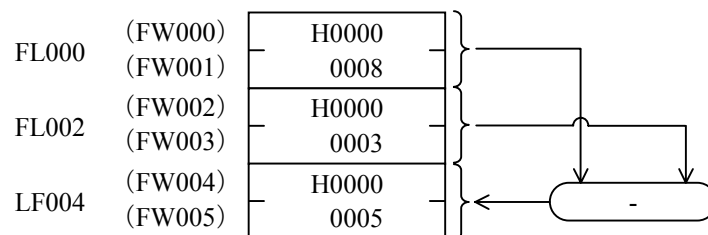
入力条件R000がONのとき、FW000の内容からイミディエートデータH0002を減算し、その結果をFW001へ格納します。



[ロング]



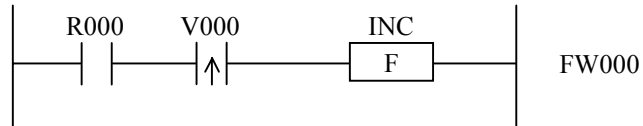
入力条件R000がOFF→ONの変化時1回のみ、FL000の内容からFL002の内容を減算し、その結果をFL004へ格納します。



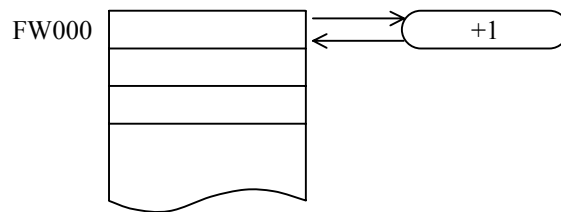
INC +1 : INCREMENT

機能	ソースの内容に1を加算します。									
パラメータ の型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	×	○	×	-	-	-	-	-	↑ ↓
パラメータ	ワード	定数なし	INC - F	SW						
		定数あり								
	ロング	定数なし	INC - F	SL						
		定数あり								
S : ソース格納アドレス					W : ワード L : ロング					
処理内容	ワード	(S) W+1 → (S) W								
	ロング	(S) L+1 → (S) L								
入力手順	(Shift + F) I N C ▣ パラメータ Enter									
フラグの 設定	V : (ワードのとき) 結果=-32768のとき1、それ以外0 (ロングワードのとき) 結果=-2147483648のとき1、それ以外0 他 : 保持									
注意事項	オーバーフロー発生時、リザルトには下記フルスケール値が格納されます。									
	ワード		ロングワード							
	H7FFF		H7FFFFFFF							

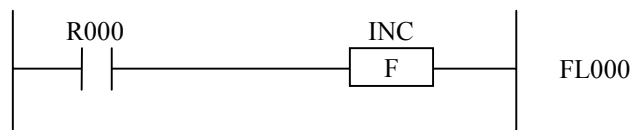
[ワード]



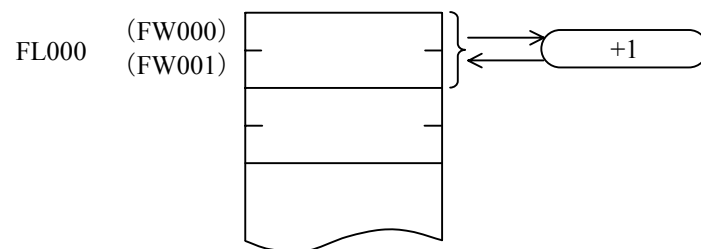
入力条件R000がOFF→ONの変化時、1回のみFW000の内容に1を加算します。



[ロング]



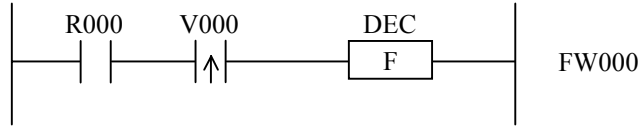
入力条件R000がONのとき、FL000の内容に1を加算します。



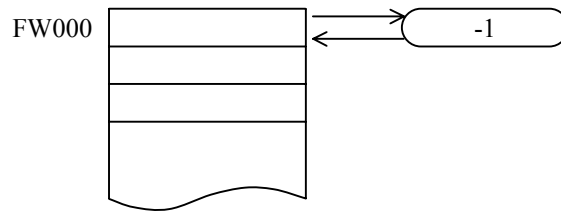
DEC -1 : DECREMENT

機能	ソースの内容に1を減算します。									
パラメータ の型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	×	○	×	-	-	-	-	-	↕
パラメータ	ワード	定数なし	DEC F	SW						
		定数あり								
	ロング	定数なし	DEC F	SL						
		定数あり								
S : ソース格納アドレス					W : ワード L : ロング					
処理内容	ワード	(S) W-1 → (S) W								
	ロング	(S) L-1 → (S) L								
入力手順	(Shift + F) D E C ┌ パラメータ Enter									
フラグの 設定	V : (ワードのとき) 結果=-32767のとき1、それ以外0 (ロングワードのとき) 結果=-2147483647のとき1、それ以外0 他 : 保持									
注意事項	オーバーフロー発生時、ソースには下記フルスケール値が格納されます。									
	ワード		ロングワード							
	H8FFF		H8000000							

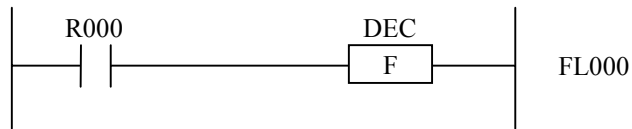
[ワード]



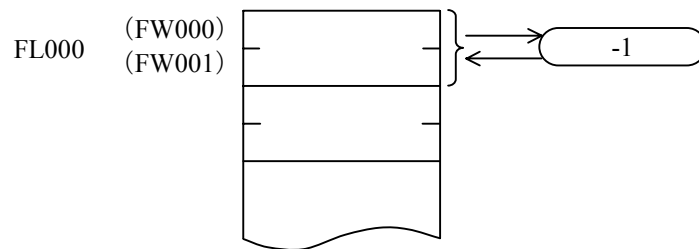
入力条件R000がOFF→ONの変化時、1回のみFW000の内容から1を減算します。



[ロング]



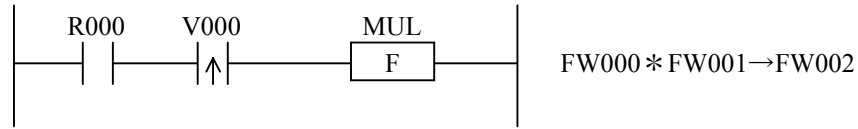
入力条件R000がONのとき、FL000の内容から1を減算します。



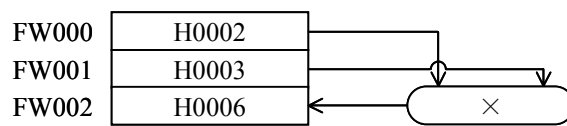
MUL 乗算 : MULTIPLY

機能	ソースの内容とデスティネーションの内容またはイミディエートデータの乗算処理を行い、結果をリザルトへ格納します。									
パラメータ の型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	○	○	×	-	-	-	-	-	↕
パラメータ	ワード	定数なし	MUL [F]	SW * DW → RW						
		定数あり	MUL [F]	SW * nW → RW						
	ロング	定数なし	MUL [F]	SL * DL → RL						
		定数あり								
	S : ソース格納アドレス				W : ワード					
	D : デスティネーション格納アドレス				L : ロング					
	n : イミディエートデータ									
	R : リザルト (演算結果) を格納するアドレス									
処理内容	ワード	(S) W × (D) W → (R) W								
	ワード定数	(S) W × n W → (R) W								
	ロング	(S) L × (D) L → (R) L								
入力手順	([Shift] + [F]) [M] [U] [L] [] パラメータ, パラメータ, パラメータ [Enter]									
フラグの 設定	V : (ワードのとき) (R) < -32768または32767 < (R) のとき1、それ以外0 (ロングワードのとき) (R) < -2147483648または2147483647 < (R) のとき1、 それ以外0 他 : 保持									
注意事項	オーバーフロー発生時、リザルトには下記フルスケール値が格納されます。									
		ワード	ロングワード							
	正のオーバーフロー時	H7FFF	H7FFFFFFF							
	負のオーバーフロー時	H8000	H80000000							

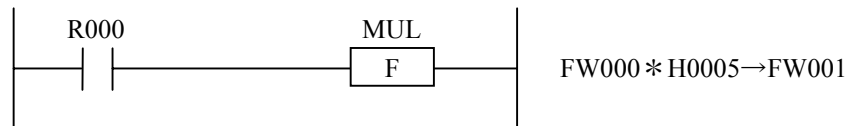
[ワード]



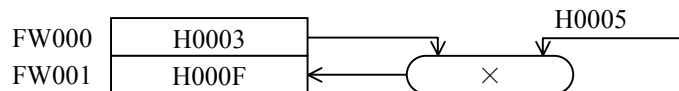
入力条件R000がOFF→ONの変化時1回のみ、FW000の内容とFW001の内容を乗算し、その結果をFW002へ格納します。



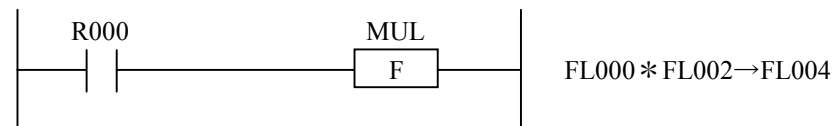
[ワード定数]



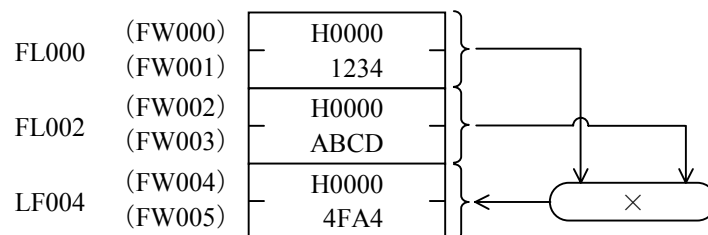
入力条件R000がONのとき、FW000の内容とイミディエートデータH0005を乗算し、その結果をFW001へ格納します。



[ロング]



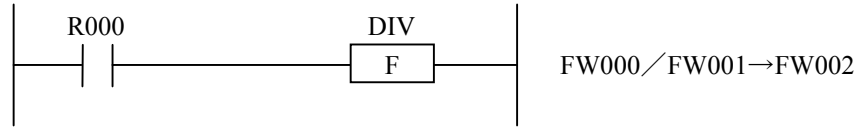
入力条件R000がONのとき、FL000の内容とFL002の内容を乗算し、その結果をFL004へ格納します。



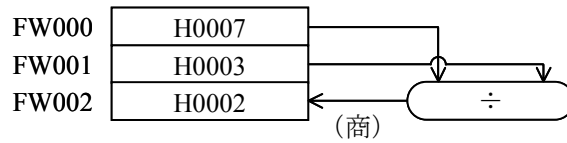
DIV 除算 : DIVIDE

機能	ソースの内容とデスティネーションの内容またはイミディエートデータの除算処理を行い、商（整数部のみ）をリザルトへ格納します。													
パラメータ の型	ワード		ロング		フラグ									
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V				
	○	○	○	×	-	↕	-	-	-	↕				
パラメータ	ワード	定数なし	$\frac{\text{DIV}}{\text{F}}$	SW/DW→RW										
		定数あり	$\frac{\text{DIV}}{\text{F}}$	SW/nW→RW										
	ロング	定数なし	$\frac{\text{DIV}}{\text{F}}$	SL/DL→RL										
		定数あり												
	S : ソース格納アドレス		W : ワード											
D : デスティネーション格納アドレス		L : ロング												
n : イミディエートデータ														
R : リザルト（演算結果）を格納するアドレス														
処理内容	ワード	(S) W ÷ (D) W → (R) W												
	ワード定数	(S) W ÷ n W → (R) W												
	ロング	(S) L ÷ (D) L → (R) L												
入力手順	$\left(\left[\text{Shift} \right] + \left[\text{F} \right] \right) \left[\text{D} \right] \left[\text{I} \right] \left[\text{V} \right] \square \text{パラメータ, パラメータ, パラメータ} \left[\text{Enter} \right]$													
フラグの 設定	E : (D) =0またはn=0のとき1、それ以外0 V : (ワードのとき) (R) =32768のとき1、それ以外0 (ロングワードのとき) (R) =2147483648のとき1、それ以外0 他 : 保持													
注意事項	<ul style="list-style-type: none"> 0除算時、エラーフラグ (E) がONし（オーバーフローフラグ (V) OFF）、リザルトは変化しません。 オーバーフロー発生時、リザルトには下記フルスケール値が格納されます。 <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>ワード</td> <td>ロングワード</td> </tr> <tr> <td>H7FFF</td> <td>H7FFFFFFF</td> </tr> </table>										ワード	ロングワード	H7FFF	H7FFFFFFF
ワード	ロングワード													
H7FFF	H7FFFFFFF													

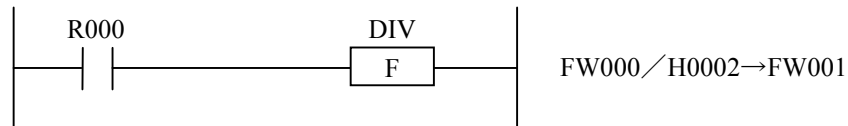
[ワード]



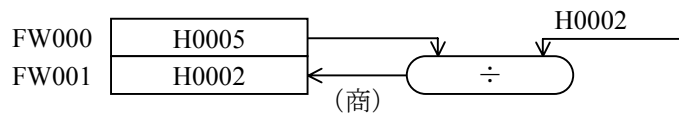
入力条件R000がONのとき、FW000の内容をFW001の内容で除算し、その結果（商）をFW002へ格納します。



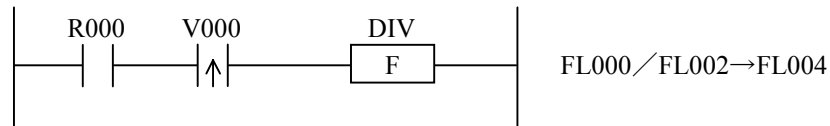
[ワード定数]



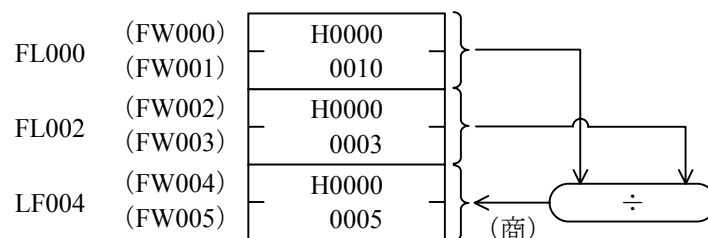
入力条件R000がONのとき、FW000の内容とイミディエートデータH0002を除算し、その結果（商）をFW001へ格納します。



[ロング]



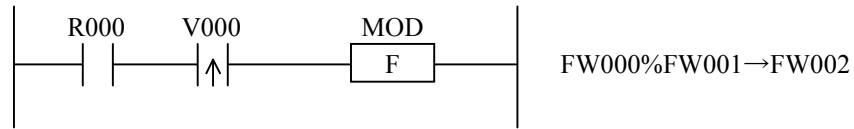
入力条件R000がOFF→ONの変化時1回のみ、FL000の内容をFL002の内容で除算し、その結果（商）をFL004へ格納します。



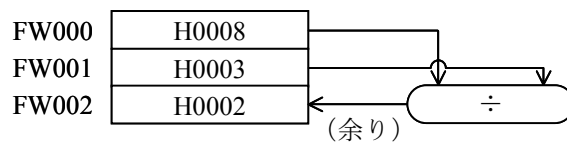
MOD 剰余：MOD

機能	ソースの内容とデスティネーションの内容またはイミディエートデータの除算処理を行い、剰余をリザルトへ格納します。									
パラメータ の型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	○	○	×	-	↕	-	-	-	↕
パラメータ	ワード	定数なし	MOD - F	SW%DW→RW						
		定数あり	MOD - F	SW%nW→RW						
	ロング	定数なし	MOD - F	SL%DL→RL						
		定数あり								
S：ソース格納アドレス		W：ワード								
D：デスティネーション格納アドレス		L：ロング								
n：イミディエートデータ										
R：リザルト（演算結果）を格納するアドレス										
処理内容	ワード	(S) W% (D) W→ (R) W								
	ワード定数	(S) W% n W→ (R) W								
	ロング	(S) L% (D) L→ (R) L								
入力手順	$(\text{Shift} + \text{F}) \text{MOD} \square$ パラメータ, パラメータ, パラメータ Enter									
フラグの 設定	E：(D)=0またはn=0のとき1、それ以外0 V：(ワードのとき) 商=32768のとき1、それ以外0 (ロングワードのとき) 商=2147483648のとき1、それ以外0 他：保持									
注意事項	<ul style="list-style-type: none"> ・0除算時、エラーフラグ (E) がONし (オーバーフローフラグ (V) OFF)、リザルトは変化しません。 ・オーバーフロー発生時、リザルトには0が格納されます。 									

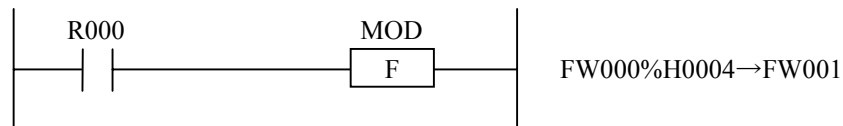
[ワード]



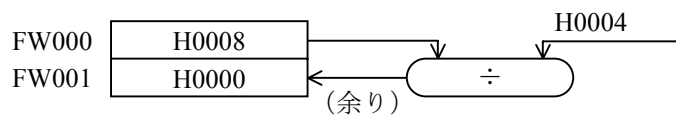
入力条件R000がOFF→ONの変化時1回のみ、FW000の内容をFW001の内容で除算し、その剰余をFW002へ格納します。



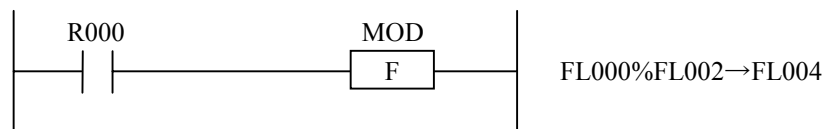
[ワード定数]



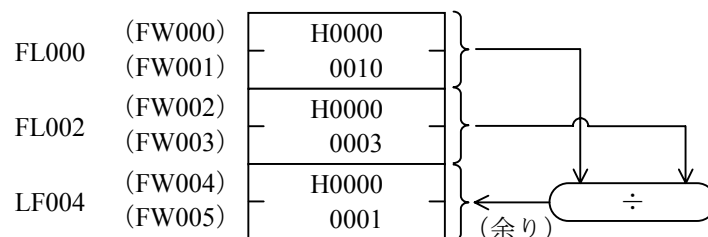
入力条件R000がONのとき、FW000の内容をイミディエートデータH0004で除算し、その剰余をFW001へ格納します。



[ロング]



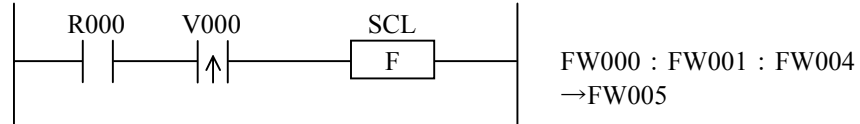
入力条件R000がONのとき、FW000の内容をFW002の内容で除算し、剰余をFW004へ格納します。



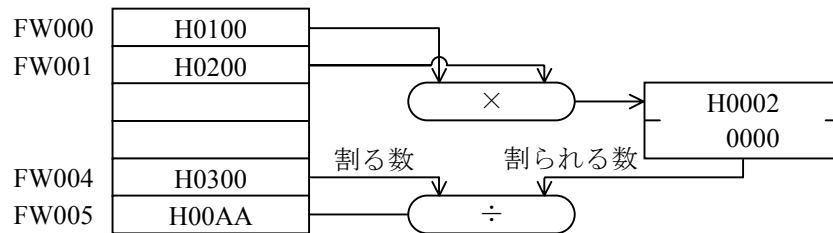
SCL スケール変換 : SCALE CHANGE

機能	ソースの内容のスケール変換（デスティネーションD1/D2またはイミディエートデータn1/n2との乗算）を行い、結果をリザルトへ格納します。											
パラメータ の型	ワード		ロング		フラグ							
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V		
	○	○	×	×	-	↕	-	-	-	↕		
パラメータ	ワード	定数なし	$\frac{\text{SCL}}{\text{F}}$	SW : D1W : D2W → RW								
		定数あり	$\frac{\text{SCL}}{\text{F}}$	SW : n1W : n2W → RW								
	ロング	定数なし	/									
		定数あり										
S : ソース格納アドレス D1, D2 : デスティネーション格納アドレス n1, n2 : イミディエートデータ R : リザルト（演算結果）を格納するアドレス W : ワード L : ロング												
処理内容	ワード	(S) W × (D1) W ÷ (D2) W → (R) W										
	ワード定数	(S) W × n1W ÷ n2W → (R) W										
入力手順	$\left(\left[\text{Shift} \right] + \left[\text{F} \right] \right) \left[\text{S} \right] \left[\text{C} \right] \left[\text{L} \right] \left[\text{ } \right] \text{パラメータ, パラメータ, パラメータ, パラメータ} \left[\text{Enter} \right]$											
フラグの 設定	E : (D2) =0またはn2=0のとき1、それ以外0 V : (R) < -32768または32767 < (R) のとき1、それ以外0 他 : 保持											
注意事項	<ul style="list-style-type: none"> 0除算時、エラーフラグ (E) がONし（オーバーフローフラグ (V) OFF）、リザルトは変化しません。 オーバーフロー発生時、リザルトには下記フルスケール値が格納されます。 											
					正のオーバーフロー時		H7FFF					
					負のオーバーフロー時		H8000					

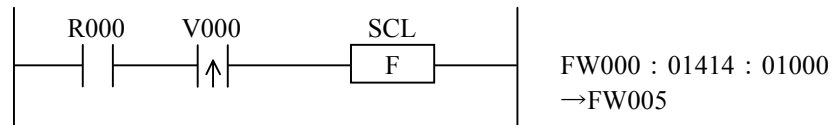
[ワード]



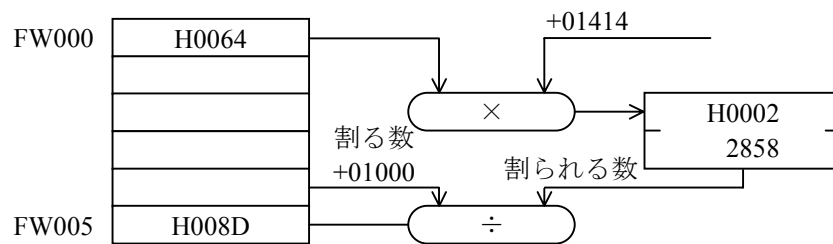
入力条件R000がON時FW000の内容のスケール変換を行い、その結果をFW005へ格納します。



[ワード定数]



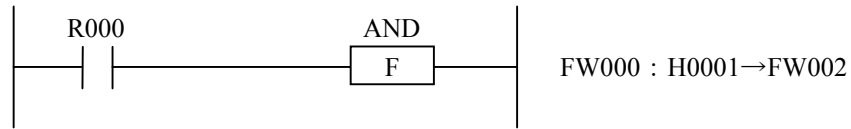
入力条件R000がOFF→ONの変化時1回のみFW000の内容のスケール変換を行い、その結果をFW005へ格納します。



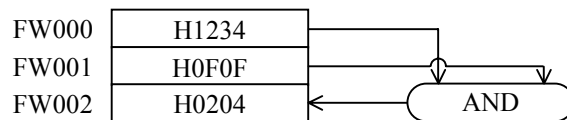
AND 論理積 : AND

機能	ソースの内容とデスティネーションの内容またはイミディエートデータの論理積処理を行い、結果をリザルトへ格納します。									
パラメータ の型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	○	○	×	-	-	-	-	-	-
パラメータ	ワード	定数なし	AND F	SW : DW→RW						
		定数あり	AND F	SW : nW→RW						
	ロング	定数なし	AND F	SL : DL→RL						
		定数あり								
S : ソース格納アドレス W : ワード D : デスティネーション格納アドレス L : ロング n : イミディエートデータ R : リザルト (演算結果) を格納するアドレス										
処理内容	ワード	(S) W ∧ (D) W → (R) W								
	ワード定数	(S) W ∧ n W → (R) W								
	ロング	(S) L ∧ (D) L → (R) L								
入力手順	(Shift + F) A N D ┘ パラメータ, パラメータ, パラメータ Enter									
注意事項	フラグはすべて保持。									

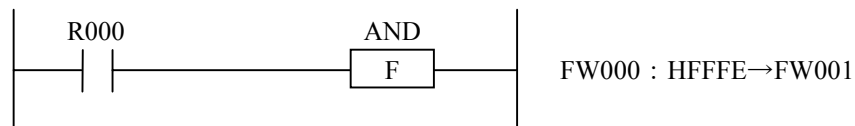
[ワード]



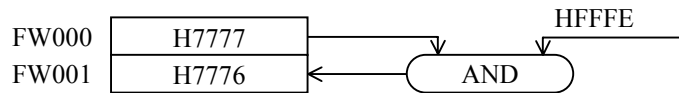
入力条件R000がONのとき、FW000の内容をFW001の内容の論理積を取り、その結果をFW002へ格納します。



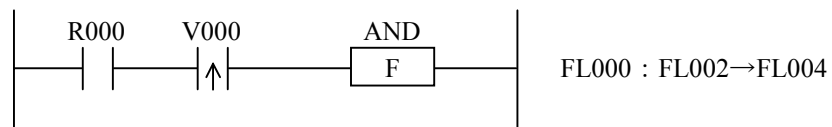
[ワード定数]



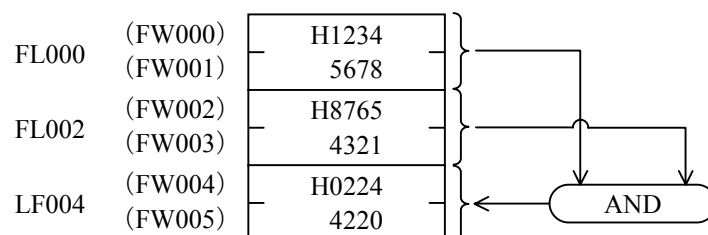
入力条件R000がONのとき、FW000の内容とイミディエートデータHFFFEの論理積を取り、その結果をFW001へ格納します。



[ロング]



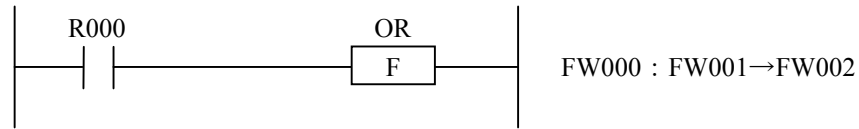
入力条件R000がOFF→ONの変化時1回のみ、FL000の内容とFL002の内容の論理積を取り、その結果をFW004へ格納します。



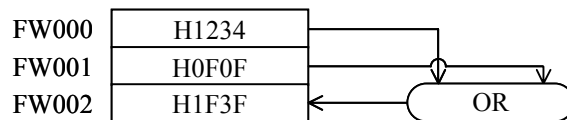
OR 論理和 : OR

機能	ソースの内容とデスティネーションの内容またはイミディエートデータの論理和处理を行い、結果をリザルトへ格納します。									
パラメータの型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	○	○	×	-	-	-	-	-	-
パラメータ	ワード	定数なし	OR - [F]	SW : DW→RW						
		定数あり	OR - [F]	SW : nW→RW						
	ロング	定数なし	OR - [F]	SL : DL→RL						
		定数あり								
S : ソース格納アドレス D : デスティネーション格納アドレス n : イミディエートデータ R : リザルト (演算結果) を格納するアドレス W : ワード L : ロング										
処理内容	ワード	(S) W∨ (D) W→ (R) W								
	ワード定数	(S) W∨ n W→ (R) W								
	ロング	(S) L∨ (D) L→ (R) L								
入力手順	([Shift] + [F]) [O] [R] [] パラメータ, パラメータ, パラメータ [Enter]									
注意事項	フラグはすべて保持。									

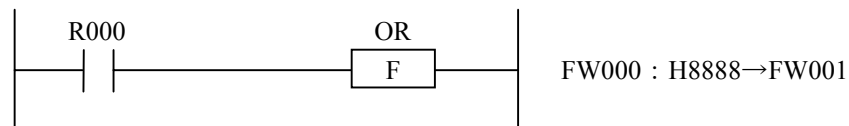
[ワード]



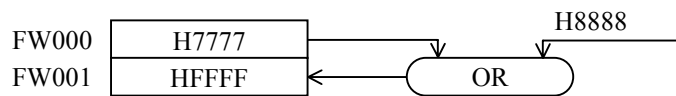
入力条件R000がONのとき、FW000の内容とFW001の内容の論理和を取り、その結果をFW002へ格納します。



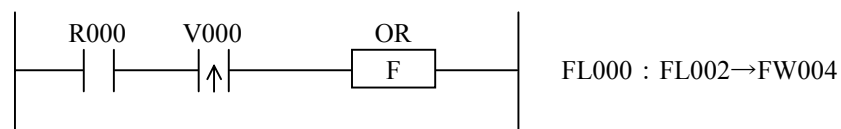
[ワード定数]



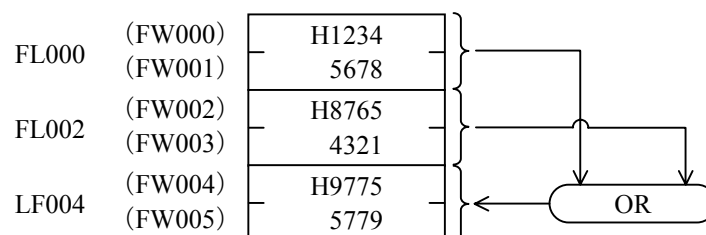
入力条件R000がONのとき、FW000の内容とイミディエートデータH8888の論理和を取り、その結果をFW001へ格納します。



[ロング]



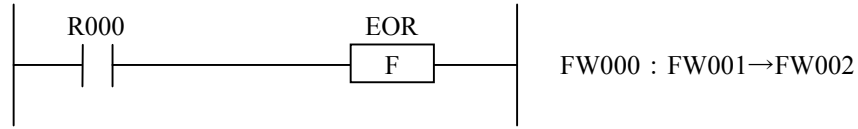
入力条件R000がOFF→ONの変化時1回のみ、FL000の内容とFL002の内容の論理和を取り、その結果をFW004へ格納します。



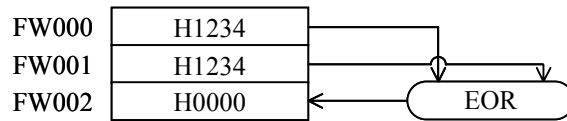
EOR 排他的論理和 : EXCLUSIVE OR

機能	ソースの内容とデスティネーションの内容またはイミディエートデータの排他的論理和処理を行い、結果をリザルトへ格納します。									
パラメータ の型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	○	○	×	-	-	-	-	-	-
パラメータ	ワード	定数なし		SW : DW → RW						
		定数あり		SW : nW → RW						
	ロング	定数なし		SL : DL → RL						
		定数あり								
S : ソース格納アドレス D : デスティネーション格納アドレス n : イミディエートデータ R : リザルト (演算結果) を格納するアドレス					W : ワード L : ロング					
処理内容	ワード	(S) W ⊕ (D) W → (R) W								
	ワード定数	(S) W ⊕ n W → (R) W								
	ロング	(S) L ⊕ (D) L → (R) L								
入力手順	+) ␣ パラメータ, パラメータ, パラメータ									
注意事項	フラグはすべて保持。									

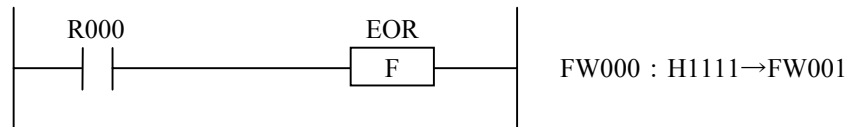
[ワード]



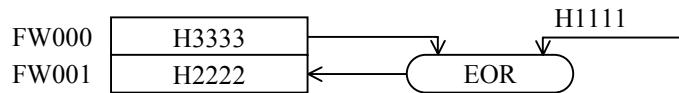
入力条件R000がONのとき、FW000の内容とFW001の内容の排他的論理和を取り、その結果をFW002へ格納します。



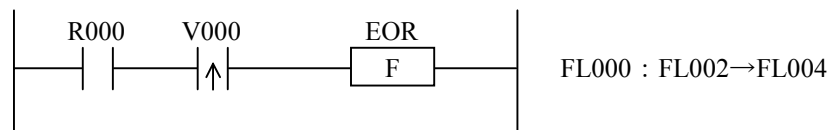
[ワード定数]



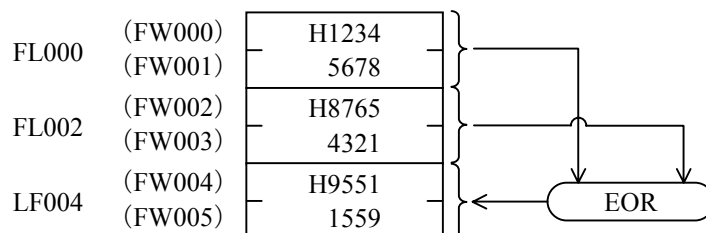
入力条件R000がONのとき、FW000の内容とイミディエートデータH1111の排他的論理和を取り、その結果をFW001へ格納します。



[ロング]



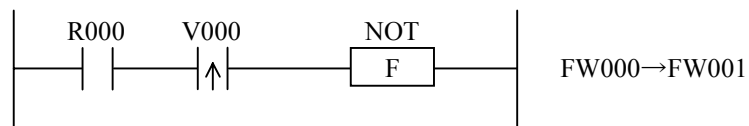
入力条件R000がOFF→ONの変化時1回のみ、FL000の内容とFL002の内容の排他的論理和を取り、その結果をFL004へ格納します。



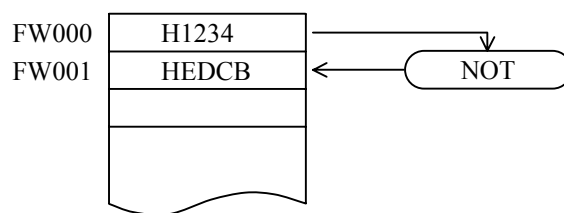
NOT 否定：NOT

機能	ソースの内容の否定（ビット反転）を、リザルトへ格納します。									
パラメータの型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	×	○	×	-	-	-	-	-	-
パラメータ	ワード	定数なし	NOT F	SW→RW						
		定数あり								
	ロング	定数なし	NOT F	SL→RL						
		定数あり								
	S：ソース格納アドレス R：リザルト（演算結果）を格納するアドレス W：ワード L：ロング									
	処理内容	ワード	(S) W → (R) W							
ロング		(S) L → (R) L								
入力手順	(Shift + F) N O T [] パラメータ, パラメータ [Enter]									
注意事項	フラグはすべて保持。									

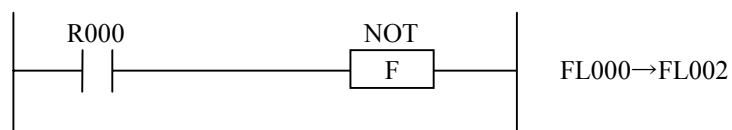
[ワード]



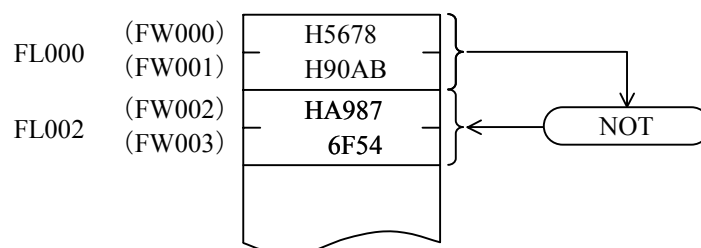
入力条件R000がOFF→ONの変化時、1回のみFW000の内容の否定を取り、FW001へ格納します。



[ロング]



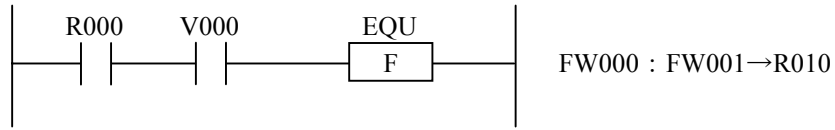
入力条件R000がONのとき、FL000の内容の否定を取り、FL002へ格納します。



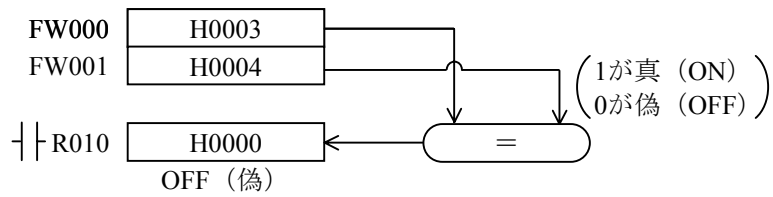
EQU = : EQUAL

機能	ソースの内容とデスティネーションの内容またはイミディエートデータの真偽判定を行い、等しければ1、等しくなければ0をリザルトへ設定します。									
パラメータの型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	○	○	×	-	-	-	-	-	-
パラメータ	ワード	定数なし		SW : DW→RW						
		定数あり		SW : nW→RW						
	ロング	定数なし		SL : DL→RW						
		定数あり								
		S : ソース格納アドレス				W : ワード				
		D : デスティネーション格納アドレス				L : ロング				
		n : イミディエートデータ								
		R : リザルト (演算結果) を格納するアドレス								
処理内容	ワード	(S) W = (D) W のとき 1 → (R) W (S) W ≠ (D) W のとき 0 → (R) W								
	ワード定数	(S) W = n W のとき 1 → (R) W (S) W ≠ n W のとき 0 → (R) W								
	ロング	(S) L = (D) L のとき 1 → (R) W (S) L ≠ (D) L のとき 0 → (R) W								
入力手順	+ □ パラメータ, パラメータ, パラメータ									
注意事項	フラグはすべて保持。									

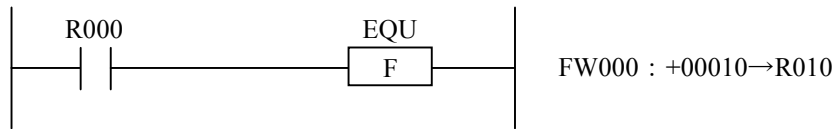
[ワード]



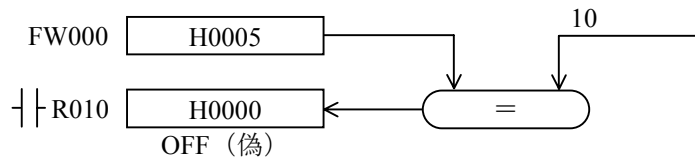
入力条件R000がOFF→ONの変化時、1回のみFW000の内容とFW001の内容の真偽判定を行い、その結果をR010へ設定します。



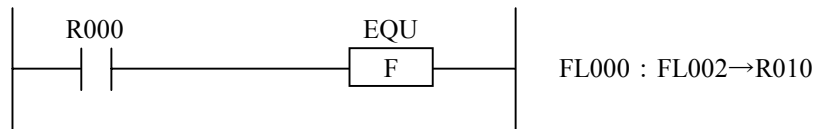
[ワード定数]



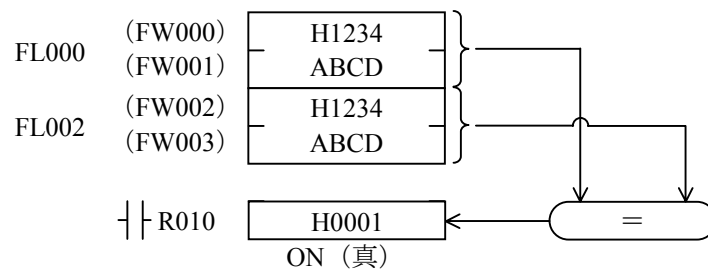
入力条件R000がONのとき、FW000の内容とイミディエートデータ10の真偽判定を行い、その結果をR010へ格納します。



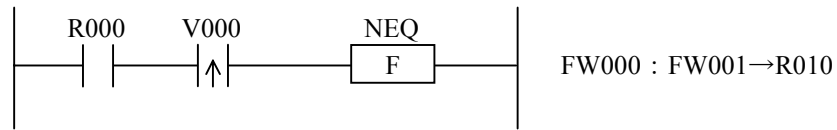
[ロング]



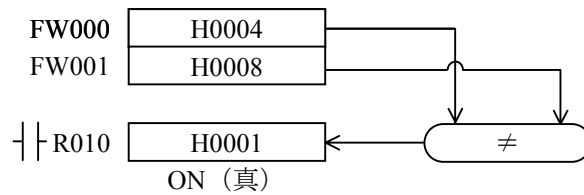
入力条件R000がONのとき、FL000の内容とFL002の内容の真偽判定を行い、その結果をR010へ格納します。



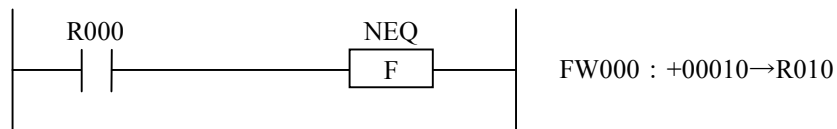
[ワード]



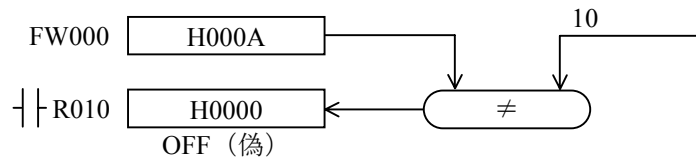
入力条件R000がOFF→ONの変化時、1回のみFW000の内容とFW001の内容の真偽判定を行い、その結果をR010へ設定します。



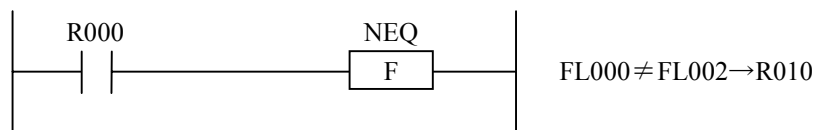
[ワード定数]



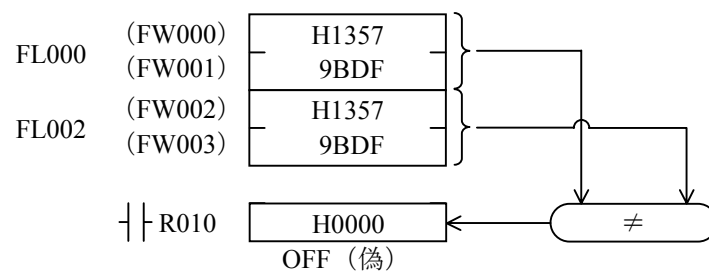
入力条件R000がONのとき、FW000の内容とイミディエートデータ10の真偽判定を行い、その結果をR010へ設定します。



[ロング]



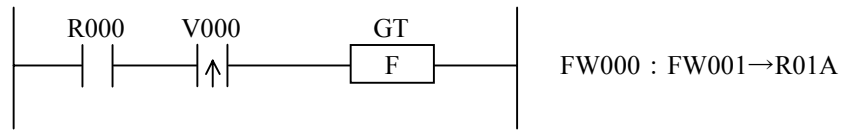
入力条件R000がONのとき、FL000の内容とFL002の内容の真偽判定を行い、その結果をR010へ設定します。



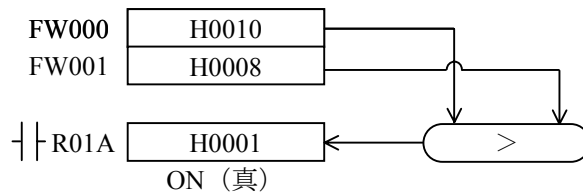
GT > : GREATER THAN

機能	ソースの内容とデスティネーションの内容またはイミディエートデータの大小比較を行い、結果をリザルトへ格納します。									
パラメータ の型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	○	○	×	-	-	-	-	-	-
パラメータ	ワード	定数なし	$\overline{\text{GT}}$ $\boxed{\text{F}}$	SW : DW→RW						
		定数あり	$\overline{\text{GT}}$ $\boxed{\text{F}}$	SW : nW→RW						
	ロング	定数なし	$\overline{\text{GT}}$ $\boxed{\text{F}}$	SL : DL→RW						
		定数あり								
S : ソース格納アドレス D : デスティネーション格納アドレス n : イミディエートデータ R : リザルト（演算結果）を格納するアドレス W : ワード L : ロング										
処理内容	ワード	(S) W > (D) W のとき 1 → (R) W (S) W ≤ (D) W のとき 0 → (R) W								
	ワード定数	(S) W > n W のとき 1 → (R) W (S) W ≤ n W のとき 0 → (R) W								
	ロング	(S) L > (D) L のとき 1 → (R) W (S) L ≤ (D) L のとき 0 → (R) W								
入力手順	$\left(\boxed{\text{Shift}} + \boxed{\text{F}} \right) \boxed{\text{G}} \boxed{\text{T}} \square$ パラメータ, パラメータ, パラメータ $\boxed{\text{Enter}}$									
注意事項	フラグはすべて保持。									

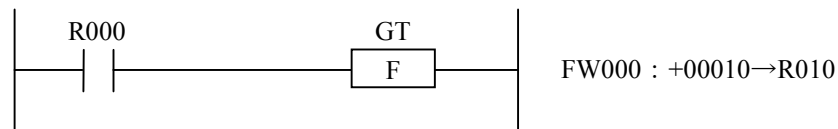
[ワード]



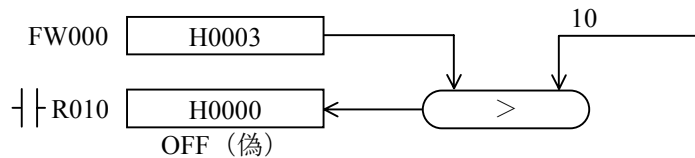
入力条件R000がOFF→ONの変化時、1回のみFW000の内容とFW001の内容の大小比較を行い、その結果をR01Aへ設定します。



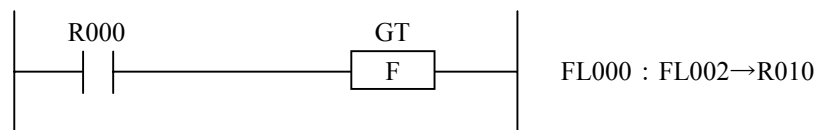
[ワード定数]



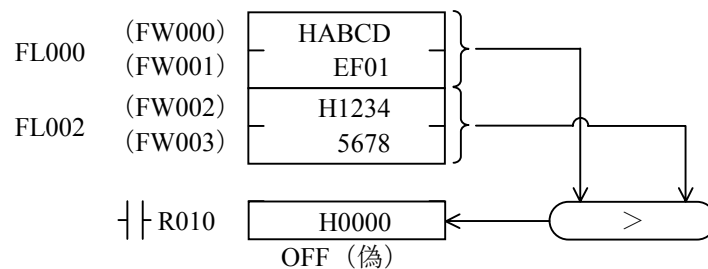
入力条件R000がONのとき、FW000の内容とイミディエートデータ10の大小比較を行い、その結果をR010へ設定します。



[ロング]



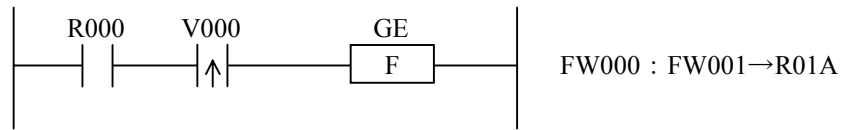
入力条件R000がONのとき、FL000の内容とFL002の内容の大小比較を行い、その結果をR010へ設定します。



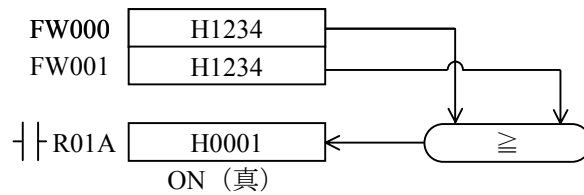
GE \geq : GREATER or EQUAL

機能	ソースの内容とデスティネーションの内容またはイミディエートデータの大小比較を行い、効果をリザルトへ格納します。									
パラメータの型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	○	○	×	-	-	-	-	-	-
パラメータ	ワード	定数なし	$\overline{\text{GE}}$ $\overline{\text{F}}$	SW : DW→RW						
		定数あり	$\overline{\text{GE}}$ $\overline{\text{F}}$	SW : nW→RW						
	ロング	定数なし	$\overline{\text{GE}}$ $\overline{\text{F}}$	SL : DL→RW						
		定数あり								
S : ソース格納アドレス D : デスティネーション格納アドレス n : イミディエートデータ R : リザルト (演算結果) を格納するアドレス W : ワード L : ロング										
処理内容	ワード	(S) $W \geq (D) W$ のとき 1→(R) W (S) $W < (D) W$ のとき 0→(R) W								
	ワード定数	(S) $W \geq n W$ のとき 1→(R) W (S) $W < n W$ のとき 0→(R) W								
	ロング	(S) $L \geq (D) L$ のとき 1→(R) W (S) $L < (D) L$ のとき 0→(R) W								
入力手順	$\left(\left[\text{Shift} \right] + \left[\text{F} \right] \right) \left[\text{G} \right] \left[\text{E} \right] \square$ パラメータ, パラメータ, パラメータ $\left[\text{Enter} \right]$									
注意事項	フラグはすべて保持。									

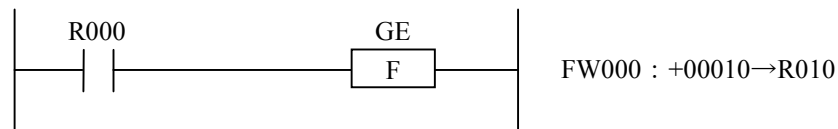
[ワード]



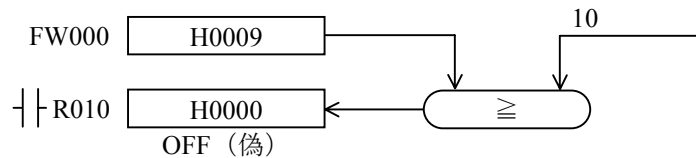
入力条件R000がOFF→ONの変化時、1回のみFW000の内容とFW001の内容の大小比較を行い、その結果をR01Aへ設定します。



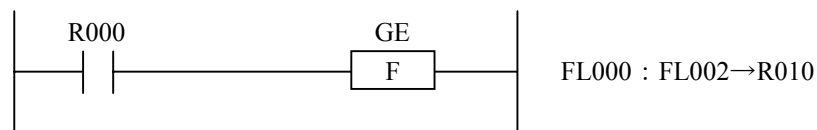
[ワード定数]



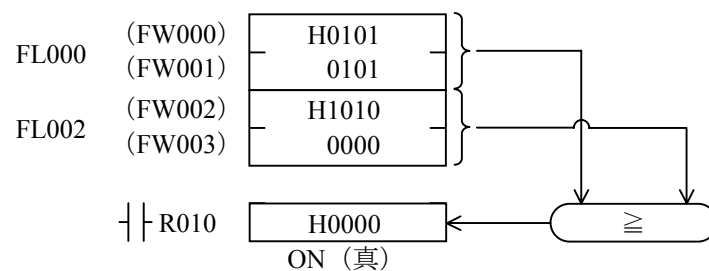
入力条件R000がONのとき、FW000の内容とイミディエートデータ10の大小比較を行い、その結果をR010へ設定します。



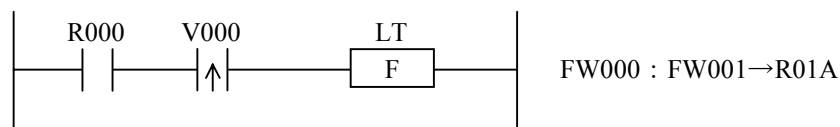
[ロング]



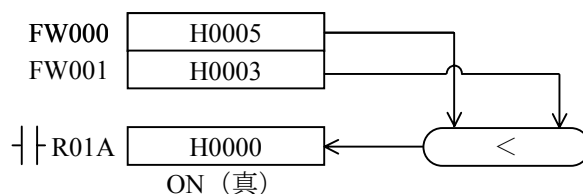
入力条件R000がONのとき、FL000の内容とFL002の内容の大小比較を行い、その結果をR010へ設定します。



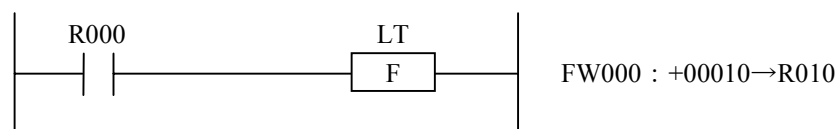
[ワード]



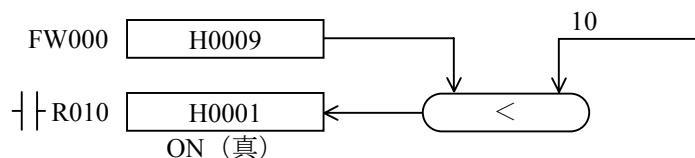
入力条件R000がOFF→ONの変化時、1回のみFW000の内容とFW001の内容の大小比較を行い、その結果をR01Aへ設定します。



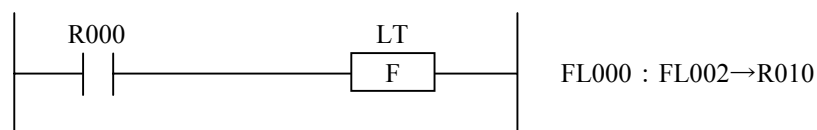
[ワード定数]



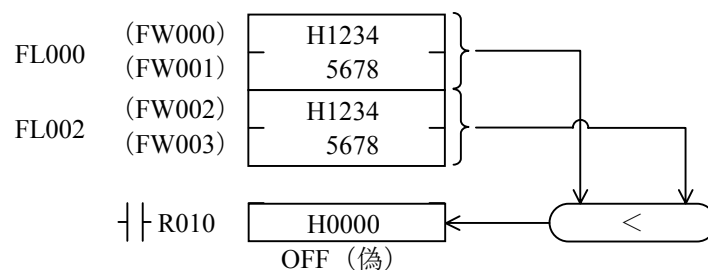
入力条件R000がONのとき、FW000の内容とイミディエートデータ10の大小比較を行い、その結果をR010へ設定します。



[ロング]



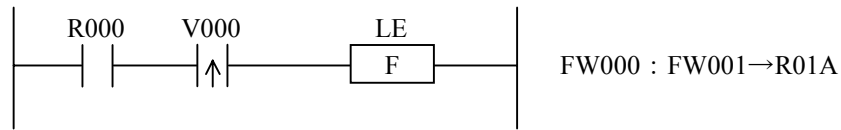
入力条件R000がONのとき、FL000の内容とFL002の内容の大小比較を行い、その結果をR010へ設定します。



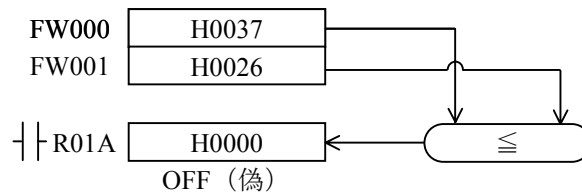
LE \leq : LESS or EQUAL

機能	ソースの内容とデスティネーションの内容またはイミディエートデータの大小比較を行い、結果をリザルトへ格納します。									
パラメータの型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	○	○	×	-	-	-	-	-	-
パラメータ	ワード	定数なし	LE - F	SW : DW → RW						
		定数あり	LE - F	SW : nW → RW						
	ロング	定数なし	LE - F	SL : DL → RW						
		定数あり								
S : ソース格納アドレス D : デスティネーション格納アドレス n : イミディエートデータ R : リザルト (演算結果) を格納するアドレス W : ワード L : ロング										
処理内容	ワード	(S) $W \leq (D) W$ のとき 1 → (R) W (S) $W > (D) W$ のとき 0 → (R) W								
	ワード定数	(S) $W \leq n W$ のとき 1 → (R) W (S) $W > n W$ のとき 0 → (R) W								
	ロング	(S) $L \leq (D) L$ のとき 1 → (R) W (S) $L > (D) L$ のとき 0 → (R) W								
入力手順	$\left(\left[\text{Shift} \right] + \left[\text{F} \right] \right) \left[\text{L} \right] \left[\text{E} \right] \square$ パラメータ, パラメータ, パラメータ $\left[\text{Enter} \right]$									
注意事項	フラグはすべて保持。									

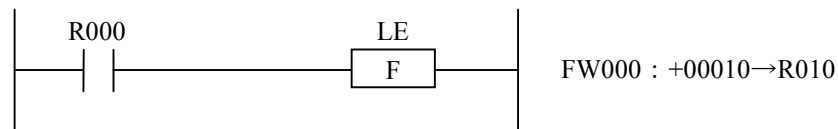
[ワード]



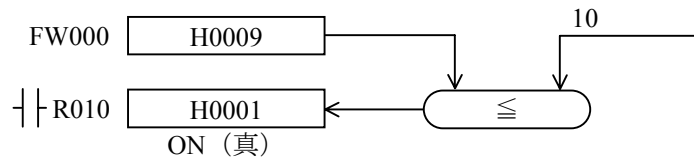
入力条件R000がOFF→ONの変化時、1回のみFW000の内容とFW001の内容の大小比較を行い、その結果をR01Aへ設定します。



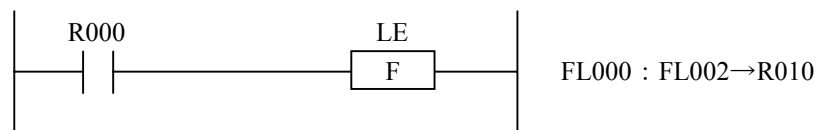
[ワード定数]



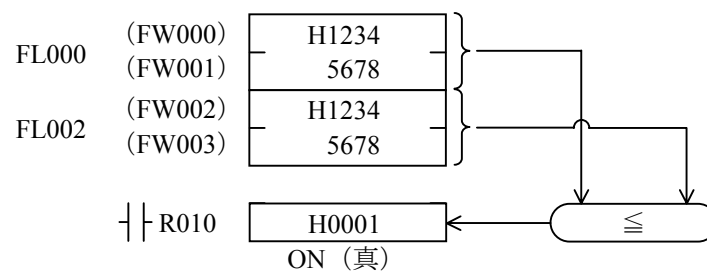
入力条件R000がONのとき、FW000の内容とイミディエートデータ10の大小比較を行い、その結果をR010へ設定します。





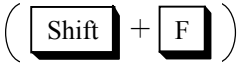




[ロング]



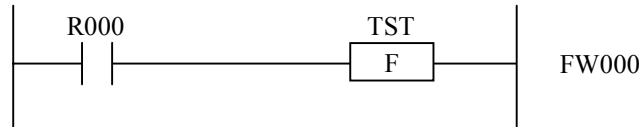
入力条件R000がONのとき、FL000の内容とFL002の内容の大小比較を行い、その結果をR010へ設定します。



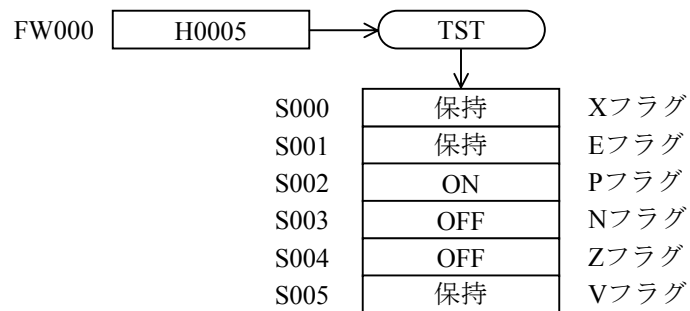
TST テスト : TEST

機能	ソースの内容の極性をテストし、ポジティブ (P)、ネガティブ (N)、ゼロ (Z) の各フラグを設定します。									
パラメータ の型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	×	○	×	-	-	↕	↕	↕	-
パラメータ	ワード	定数なし								
		定数あり								
	ロング	定数なし								
		定数あり								
	S : ソース格納アドレス					W : ワード L : ロング				
処理内容	ワード	(S) W>0 : P ON (N, Z OFF) (S) W=0 : Z ON (P, N OFF) (S) W<0 : N ON (P, Z OFF)								
	ロング	(S) L>0 : P ON (N, Z OFF) (S) L=0 : Z ON (P, N OFF) (S) L<0 : N ON (P, Z OFF)								
入力手順	    <input type="checkbox"/> パラメータ 									
注意事項	P, N, Z以外のフラグはすべて保持。									

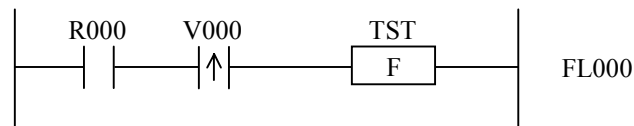
[ワード]



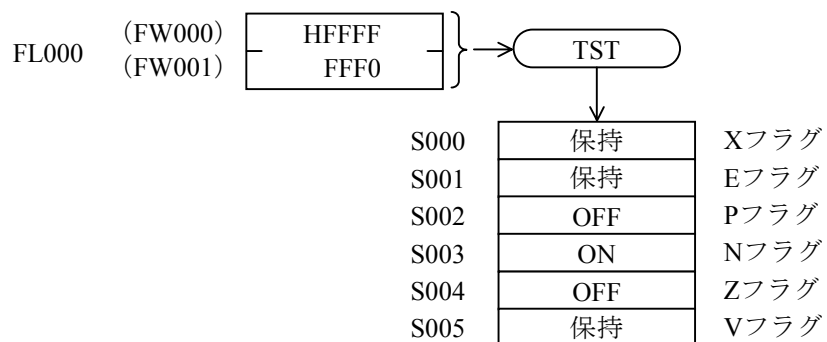
入力条件R000がONのとき、FW000の内容の極性をテストし、フラグを設定します。



[ロング]



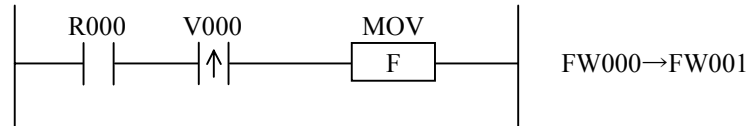
入力条件R000がOFF→ONの変化時、FL000の内容の極性をテストし、フラグを設定します。



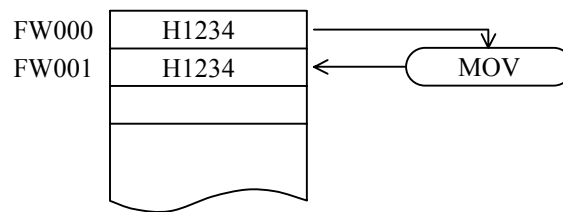
MOV 転送 : MOVE

機能	ソースの内容をデスティネーションへ転送します。									
パラメータ の型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	×	○	×	-	-	-	-	-	-
パラメータ	ワード	定数なし	MOV F	SW→DW						
		定数あり								
	ロング	定数なし	MOV F	SL→DL						
		定数あり								
S : ソース格納アドレス					W : ワード					
D : デスティネーション格納アドレス					L : ロング					
処理内容	ワード	(S) W → (D) W								
	ロング	(S) L → (D) L								
入力手順	(Shift + F) M O V ┐ パラメータ, パラメータ Enter									
注意事項	フラグはすべて保持。									

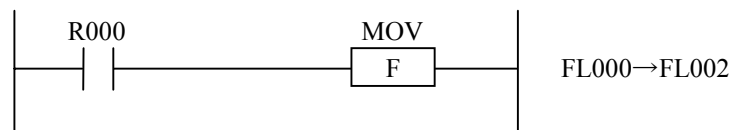
[ワード]



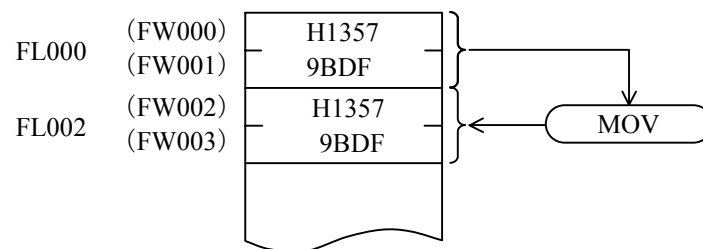
入力条件R000がOFF→ONの変化時、1回のみFW000の内容をFW001へ転送します。



[ロング]



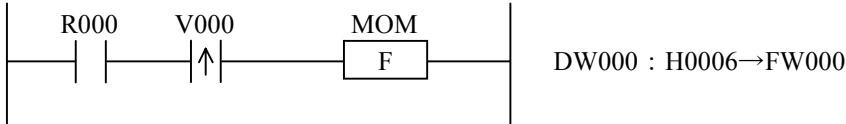
入力条件R000がONのとき、FL000の内容をFL002へ転送します。



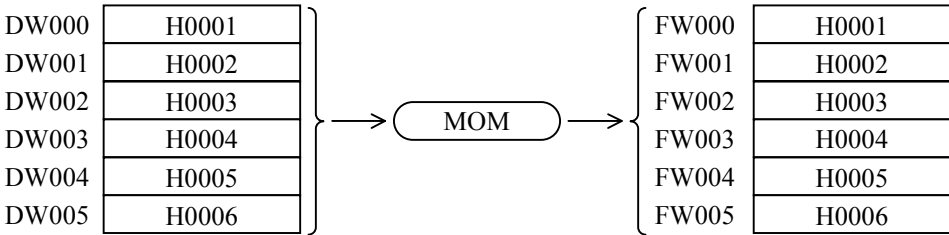
MOM 一括転送 : MOVE MULTI

機能	ソースからnステップをデスティネーションからnステップに転送します。									
パラメータ の型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	×	○	×	-	-	-	-	-	-
パラメータ	ワード	定数なし	MOM F	SW : nW→DW						
		定数あり								
	ロング	定数なし	MOM F	SL : nW→DL						
		定数あり								
S : ソース格納アドレス n : イミディエートデータ D : デスティネーション格納アドレス					W : ワード L : ロング					
処理内容	ワード				ロング					
入力手順	(Shift + F) M O M □ パラメータ, パラメータ, パラメータ Enter									
注意事項	<ul style="list-style-type: none"> ・フラグはすべて保持。 ・ステップ数$n \leq 0$または$256 < n$のとき、未処理。 									

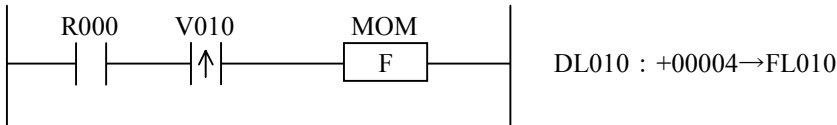
[ワード]



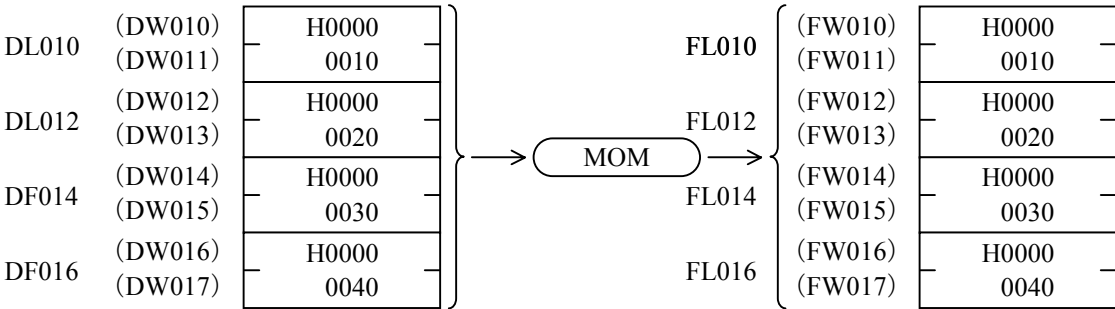
入力条件R000がOFF→ONの変化時、1回のみDW000から6ステップをFW000から6ステップに転送します。



[ロング]



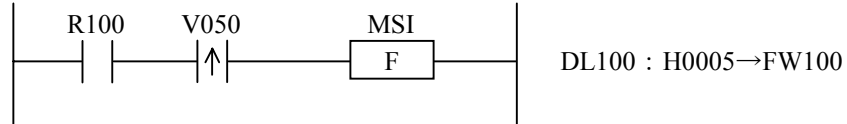
入力条件R001がOFF→ONの変化時、1回のみDL010から4ステップをFL010から4ステップに転送します。



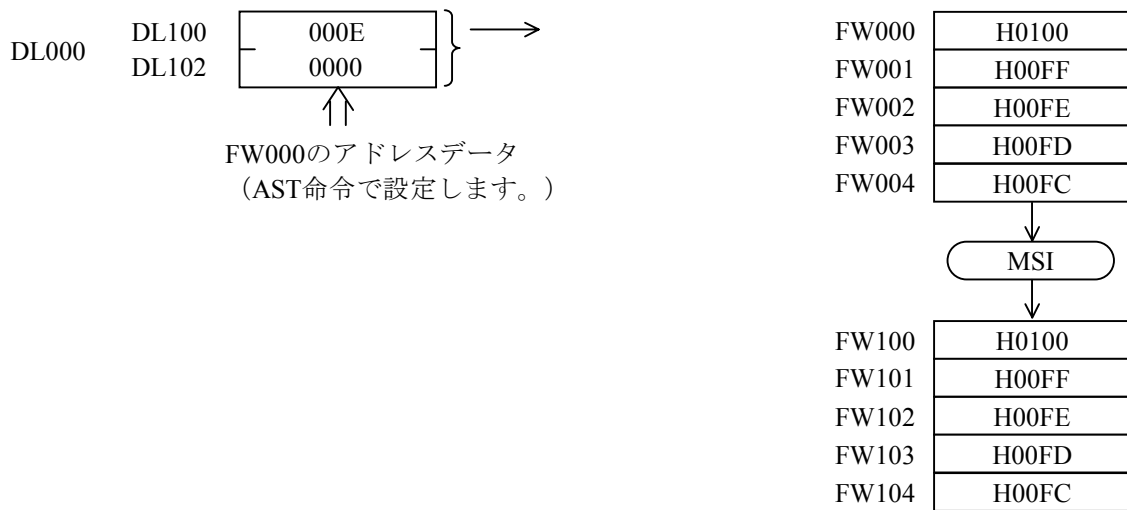
MSI ソースインダイレクト転送 : MOVE MULTI SOURCE INDIRECT

機能	ソース（インダイレクトモード）からnステップをデスティネーションからnステップへ転送します。									
パラメータの型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	×	×	×	-	-	-	-	-	-
パラメータ	ワード	定数なし	MSI F	SL : nW→DW						
		定数あり								
	ロング	定数なし								
		定数あり								
S : ソース格納アドレス n : イミディエートデータ D : デスティネーション格納アドレス					W : ワード L : ロング					
処理内容	ワード									
入力手順	(Shift + F) M S I □ パラメータ, パラメータ, パラメータ Enter									
注意事項	<ul style="list-style-type: none"> ・フラグはすべて保持。 ・ステップ数$n \leq 0$または$256 < n$のとき、未処理。 									

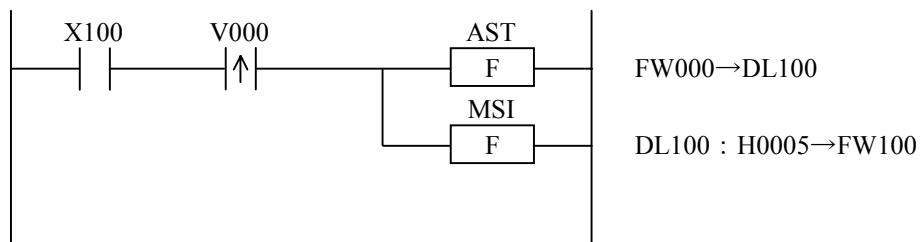
[ワード]



入力条件R100がOFF→ONの変化時、1回のみDL100に格納されたアドレスデータから5ステップをFW100から5ステップに転送します。



<ASTとの組み合わせ回路>

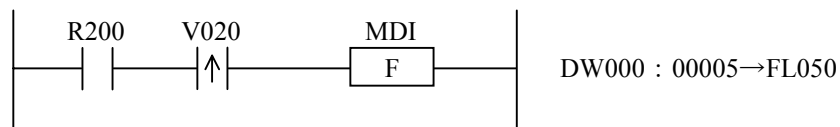


MDI デスティネーションインダイレクト転送 : MOVE MULTI DESTINATION INDIRECT

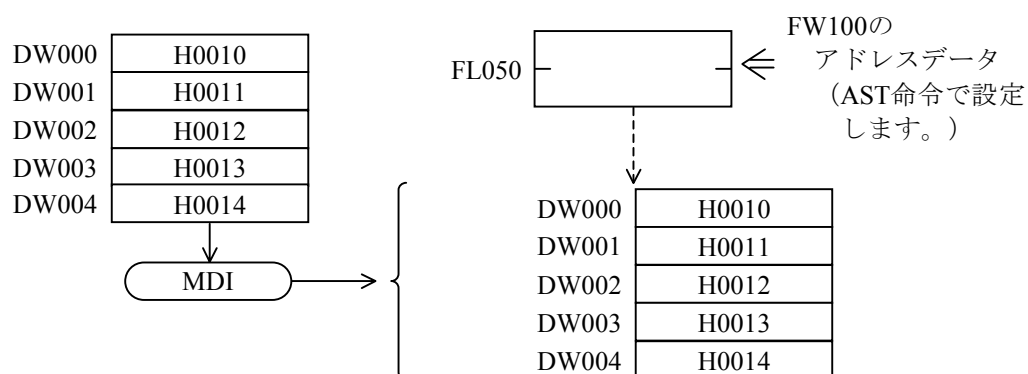
機能	ソースからnステップをデスティネーション（インダイレクトモード）からnステップへ転送します。									
パラメータの型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	×	×	×	-	-	-	-	-	-
パラメータ	ワード	定数なし	MDI F	SW : nW→DL						
		定数あり								
	ロング	定数なし								
		定数あり								
S : ソース格納アドレス					W : ワード					
n : イミディエートデータ					L : ロング					
D : デスティネーション格納アドレス										
処理内容	ワード									
入力手順	(Shift + F) M D I □ パラメータ, パラメータ, パラメータ Enter									
注意事項	<ul style="list-style-type: none"> ・フラグはすべて保持。 ・ステップ数$n \leq 0$または$256 < n$のとき、未処理。 									

MDI デスティネーションインダイレクト転送 : MOVE MULTI DESTINATION INDIRECT

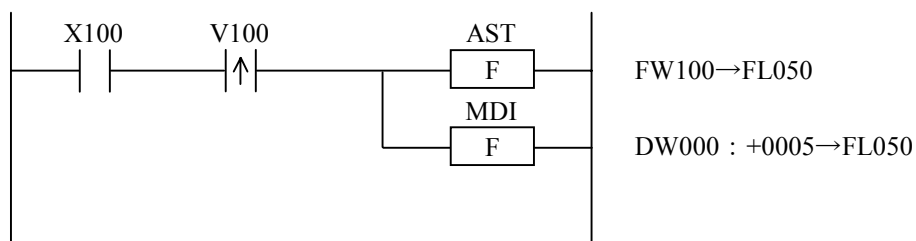
[ワード]



入力条件R200がOFF→ONの変化時、1回のみDW000から5ステップをFW050に格納されたアドレスデータから5ステップへ転送します。



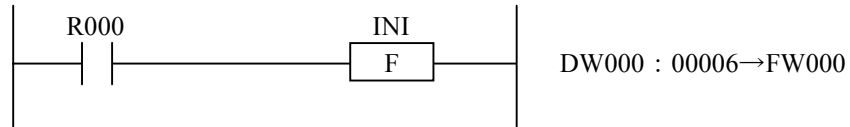
<ASTとの組み合わせ回路>



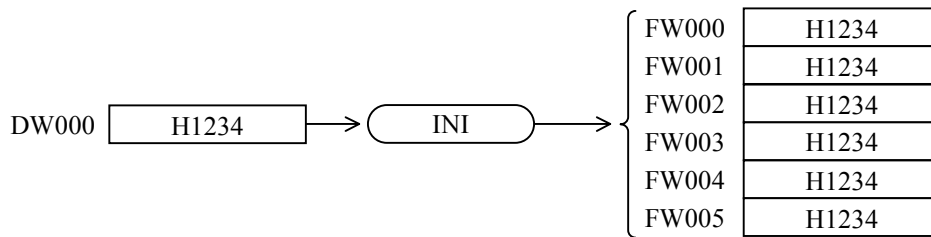
INI 同一データ一括転送：INITIAL

機能	ソースの内容またはイミディエートデータ n_0 をデスティネーションから n ステップに転送します。									
パラメータの型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	○	○	×	-	-	-	-	-	-
パラメータ	ワード	定数なし	INI F	SW : $nW \rightarrow DW$						
		定数あり	INI F	$n1W : n2W \rightarrow DW$						
	ロング	定数なし	INI F	SL : $nW \rightarrow DL$						
		定数あり								
S : ソース格納アドレス n_0, n : イミディエートデータ D : デスティネーション格納アドレス					W : ワード L : ロング					
処理内容	ワード		ワード定数			ロング				
入力手順	(Shift + F) I N I □ パラメータ, パラメータ, パラメータ Enter									
注意事項	<ul style="list-style-type: none"> ・フラグはすべて保持。 ・ステップ数$n \leq 0$または$256 < n$のとき、未処理。 									

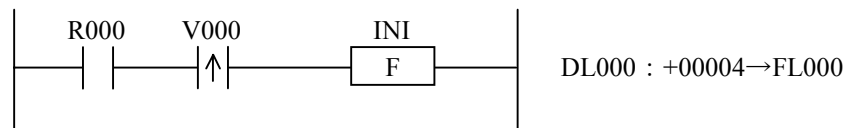
[ワード]



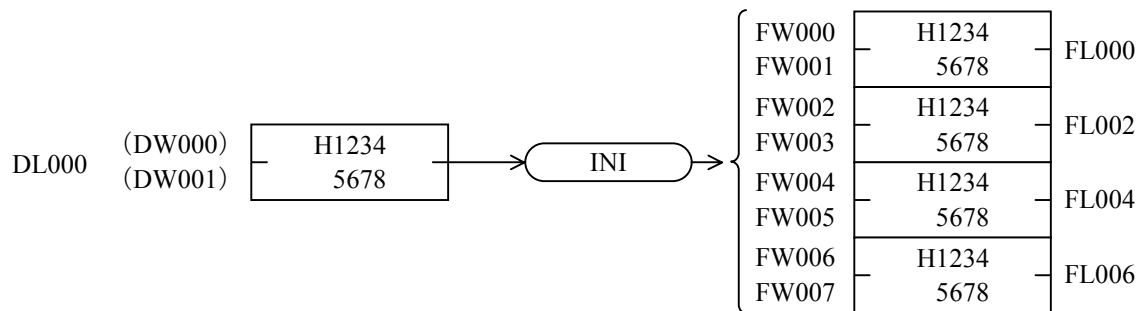
入力条件R000がONのとき、DW000の内容をFW000から6ステップに転送します。



[ロング]



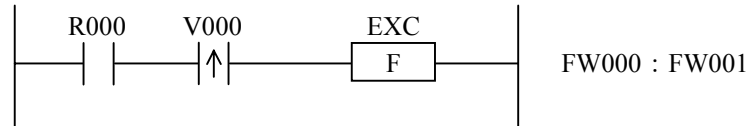
入力条件R000がOFF→ONの変化時、DL000の内容をFL000から4ステップに転送します。



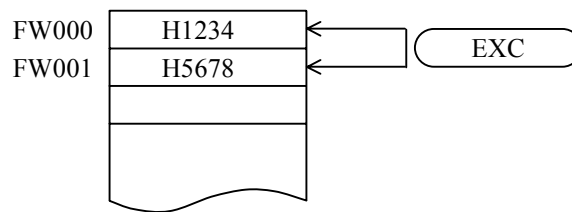
EXC 交換 : EXCHANGE

機能	ソースの内容とデスティネーションの内容の交換を行います。									
パラメータ の型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	×	○	×	-	-	-	-	-	-
パラメータ	ワード	定数なし	EXC F	SW : DW						
		定数あり								
	ロング	定数なし	EXC F	SL : DL						
		定数あり								
S : ソース格納アドレス		W : ワード								
D : デスティネーション格納アドレス		L : ロング								
処理内容	ワード	(S) W ↔ (D) W								
	ロング	(S) L ↔ (D) L								
入力手順	(Shift + F) E X C □ パラメータ, パラメータ Enter									
注意事項	フラグはすべて保持。									

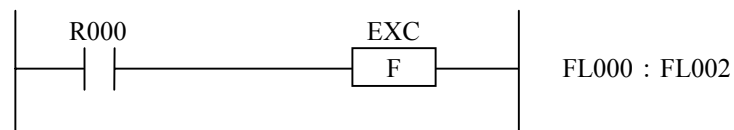
[ワード]



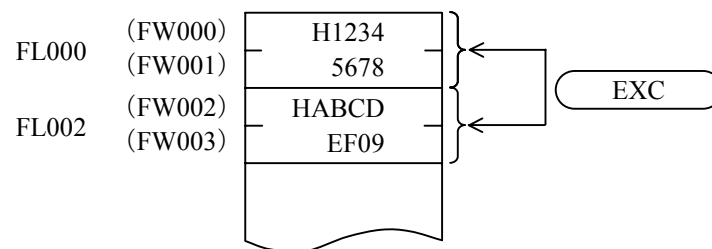
入力条件R000がOFF→ONの変化時、1回のみFW000の内容とFW001の内容の交換を行います。



[ロング]



入力条件R000がONのとき、FL000の内容とFL002の内容の交換を行います。

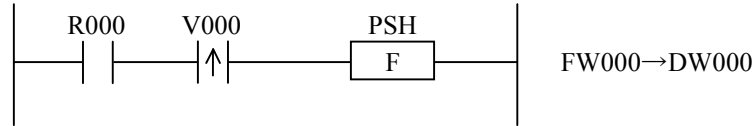


(注) ロング形式を指定する場合、アドレスが重複しないようにしてください。

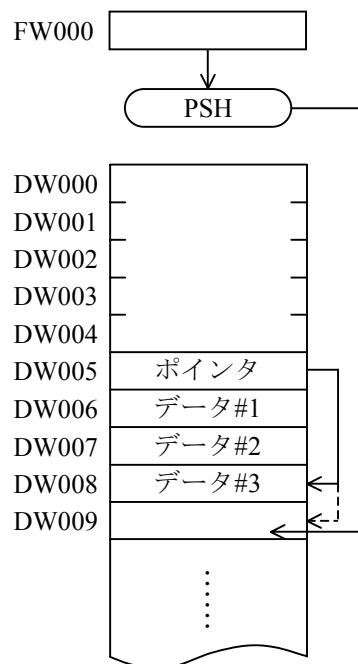
PSH FIFO書き込み : FIFO PUSH

機能	ソースの内容をFIFOテーブルへプッシュします。																																				
パラメータ の型	ワード		ロング		フラグ																																
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V																											
	○	×	×	×	-	-	-	-	-	-																											
パラメータ	ワード	定数なし	PSH F	SW→TBW																																	
		定数あり																																			
	ロング	定数なし																																			
		定数あり																																			
		S : ソース格納アドレス				W : ワード																															
		TB : FIFOテーブル先頭アドレス																																			
処理内容	<p>データの動作</p>				<p>FIFOデータテーブルの構成</p> <table border="1"> <tr> <td>FIFOトップ</td> <td>A+0</td> <td>n (データサイズ)</td> </tr> <tr> <td></td> <td>B+2</td> <td>ZEROフラグ</td> </tr> <tr> <td></td> <td>C+4</td> <td>アドレス</td> </tr> <tr> <td></td> <td>D+6</td> <td>FULLフラグ</td> </tr> <tr> <td></td> <td>FE+8</td> <td>アドレス</td> </tr> <tr> <td></td> <td>FF+10</td> <td>ポインタ</td> </tr> <tr> <td></td> <td>FW000+12</td> <td>データ#1</td> </tr> <tr> <td></td> <td></td> <td>⋮</td> </tr> <tr> <td></td> <td>n×2+10</td> <td>データ#n</td> </tr> </table> <p>データサイズで指定したデータ格納エリア</p>						FIFOトップ	A+0	n (データサイズ)		B+2	ZEROフラグ		C+4	アドレス		D+6	FULLフラグ		FE+8	アドレス		FF+10	ポインタ		FW000+12	データ#1			⋮		n×2+10	データ#n
FIFOトップ	A+0	n (データサイズ)																																			
	B+2	ZEROフラグ																																			
	C+4	アドレス																																			
	D+6	FULLフラグ																																			
	FE+8	アドレス																																			
	FF+10	ポインタ																																			
	FW000+12	データ#1																																			
		⋮																																			
	n×2+10	データ#n																																			
入力手順	<p>(Shift + F) P S H <input type="checkbox"/>パラメータ, <input type="checkbox"/>パラメータ Enter</p>																																				
注意事項	<ul style="list-style-type: none"> • ZEROフラグはOFFされます。 • プッシュ処理前にポインタがnであれば、FULLフラグをONし、プッシュ未処理とします。また、プッシュ処理後ポインタをインクリメントしてnになった場合もFULLフラグON。それ以外の場合、FULLフラグOFF。 • データサイズn ≤ 0または256 < nの場合、未処理。 • ポインタ < 0またはn < ポインタの場合、未処理。 • フラグ (X, E, P, N, Z, V) はすべて保持。 																																				

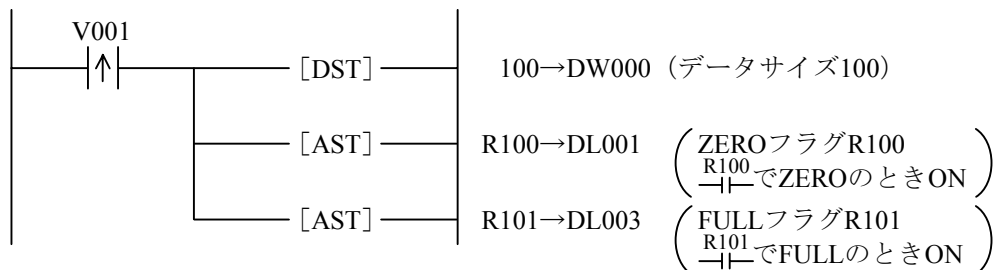
[ワード]



入力条件R000がOFF→ONの変化時、1回のみFW000の内容をDW000からのFIFOデータテーブルへ設定します。



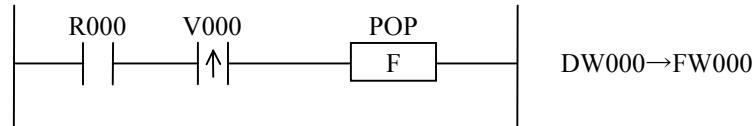
- データサイズnはデータセットDSTなどで行います。
- ZEROフラグ、FULLフラグのアドレスはアドレスセットASTで行います。



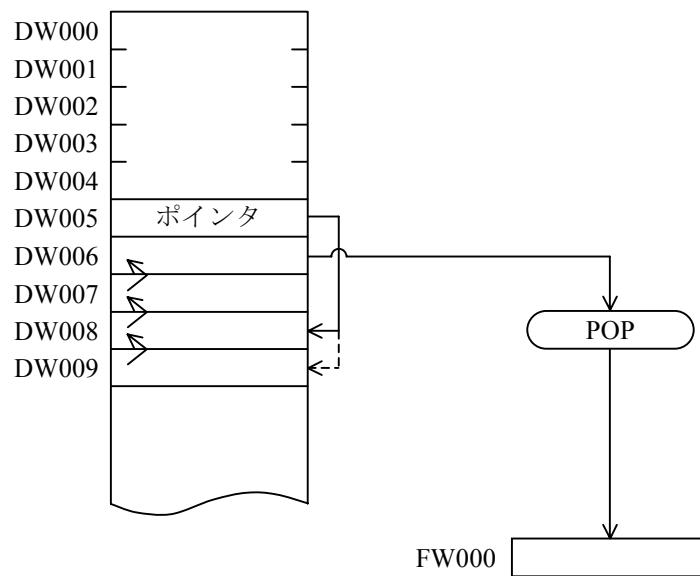
POP FIFO読み出し : FIFO POP

機能	FIFOテーブルのポップ処理を行い、ポップデータをデスティネーションへ格納します。																														
パラメータの型	ワード		ロング		フラグ																										
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V																					
	○	×	×	×	-	-	-	-	-	-																					
パラメータ	ワード	定数なし	POP F	TBW→DW																											
		定数あり																													
	ロング	定数なし																													
		定数あり																													
TB : FIFOテーブル先頭アドレス					W : ワード																										
D : デスティネーション格納アドレス																															
処理内容	<p>データの動作</p>				<p>FIFOテーブルの構成</p> <table border="1"> <tr> <td>FIFOトップ+0</td> <td>n (データサイズ)</td> <td rowspan="12">} データサイズで指定したデータ格納エリア</td> </tr> <tr> <td>+2</td> <td>ZEROフラグ</td> </tr> <tr> <td>+4</td> <td>アドレス</td> </tr> <tr> <td>+6</td> <td>FULLフラグ</td> </tr> <tr> <td>+8</td> <td>アドレス</td> </tr> <tr> <td>+10</td> <td>ポインタ</td> </tr> <tr> <td>+12</td> <td>データ#1</td> </tr> <tr> <td></td> <td>⋮</td> </tr> <tr> <td></td> <td>データ#n</td> </tr> <tr> <td>n×2+10</td> <td></td> </tr> </table>						FIFOトップ+0	n (データサイズ)	} データサイズで指定したデータ格納エリア	+2	ZEROフラグ	+4	アドレス	+6	FULLフラグ	+8	アドレス	+10	ポインタ	+12	データ#1		⋮		データ#n	n×2+10	
FIFOトップ+0	n (データサイズ)	} データサイズで指定したデータ格納エリア																													
+2	ZEROフラグ																														
+4	アドレス																														
+6	FULLフラグ																														
+8	アドレス																														
+10	ポインタ																														
+12	データ#1																														
	⋮																														
	データ#n																														
n×2+10																															
入力手順																															
注意事項	<ul style="list-style-type: none"> • FULLフラグはOFFされます。 • ポップ処理前にポインタが0であれば、ZEROフラグをONし、ポップ未処理とします。また、ポップ処理後ポインタをデクリメントして0になった場合もZEROフラグON。それ以外の場合、ZEROフラグOFF。 • データサイズ$n \leq 0$または$256 < n$の場合、未処理。 • ポインタ< 0または$n < \text{ポインタ}$の場合、未処理。 • フラグ (X, E, P, N, Z, V) はすべて保持。 																														

[ワード]



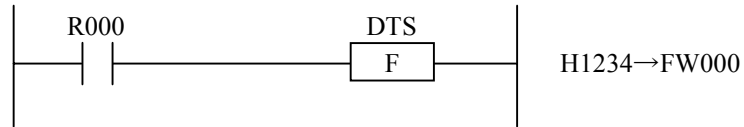
入力条件R000がOFF→ONの変化時、1回のみDW000よりのFIFOデータテーブルからデータを取り込み、FW000へ格納します。



DST データセット : DATA SET

機能	イミディエートデータをデスティネーションへ格納します。									
パラメータ の型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	×	○	×	○	-	-	-	-	-	-
パラメータ	ワード	定数なし								
		定数あり								
	ロング	定数なし								
		定数あり								
	n : イミディエートデータ		W : ワード							
D : デスティネーション格納アドレス		L : ロング								
処理内容	ワード定数	nW→ (D) W								
	ロング定数	nL→ (D) L								
入力手順	([Shift] + [F]) [D] [S] [T] □ パラメータ, パラメータ [Enter]									
注意事項	フラグはすべて保持。									

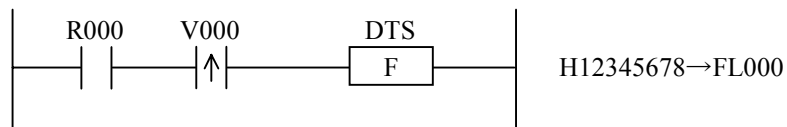
[ワード定数]



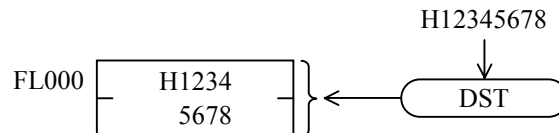
入力条件R000がONのとき、イミディエートデータH1234をFW000へ格納します。



[ロング定数]



入力条件R000がOFF→ONの変化時、1回のみイミディエートデータH12345678をFL000へ格納します。

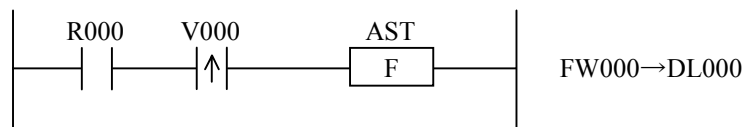


- イミディエートデータの10進入力の場合、データ範囲は
 $-999999999 \leq n \leq +999999999$
 となります。

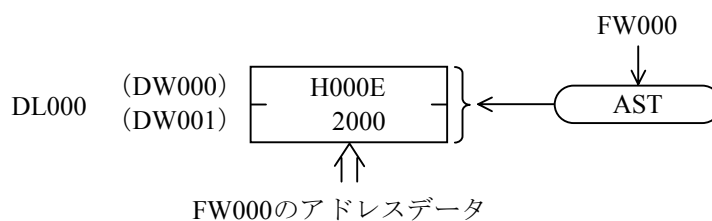
AST アドレスセット : ADDRESS SET

機能	ソースのアドレスデータをデスティネーションへ格納します。									
パラメータ の型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	×	×	○	×	-	-	-	-	-	-
パラメータ	ワード	定数なし								
		定数あり								
	ロング	定数なし	AST F	SW→DL						
		定数あり								
S : ソース格納アドレス		W : ワード								
D : デスティネーション格納アドレス		L : ロング								
処理内容	AST SW (アドレスデータ) → (D) L									
入力手順	(Shift + F) A S T □ パラメータ, パラメータ Enter									
注意事項	フラグはすべて保持。									

[ロング]



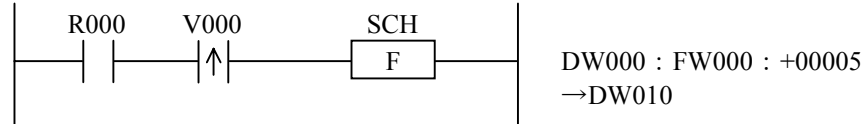
入力条件R000がOFF→ONの変化時、1回のみFW000のアドレスデータをDL000へ格納します。



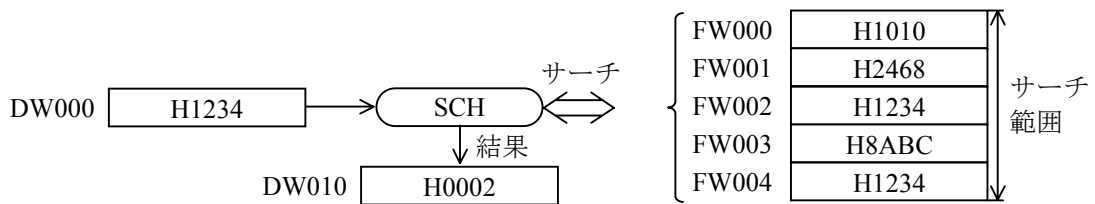
SCH サーチ : SEARCH

機能	ソースの内容がデスティネーションから指定エリア（サーチステップ数m）内に存在するかどうか探し、デスティネーションから探し出すまでのステップ数をリザルトに格納します。									
パラメータの型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	×	○	×	-	-	-	-	-	-
パラメータ	ワード	定数なし								
		定数あり	SCH F	SW : DW : nW → RW						
	ロング	定数なし								
		定数あり	SCH F	SL : DL : nW → RW						
S : ソース格納アドレス					W : ワード					
D : デスティネーション格納アドレス					L : ロング					
m : イミディエートデータ										
R : リザルト（演算結果）を格納するアドレス										
処理内容	ワード				ロング					
入力手順	(Shift + F) S C H □ パラメータ, パラメータ, パラメータ, パラメータ Enter									
注意事項	<ul style="list-style-type: none"> ・フラグはすべて保持。 ・一致データは、サーチ範囲の先頭から最初に見つけたもののみ有効。 ・サーチ範囲に一致データがないとき、リザルトには-1 (HFFF) が格納されます。 ・サーチステップ数$m \leq 0$または$256 < m$のとき、未処理。 									

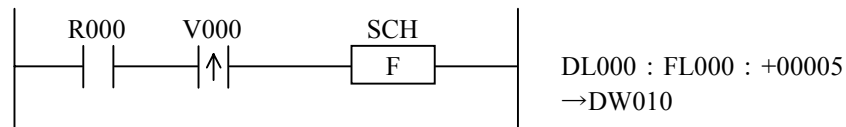
[ワード]



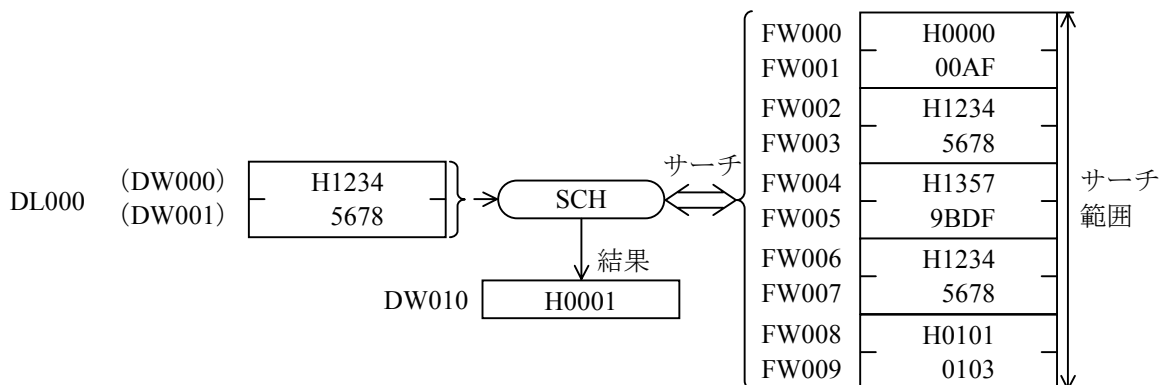
入力条件R000がOFF→ONの変化時、1回のみDW000の内容と同一データがFW000から5ステップ内に存在するかどうか探し、その結果をDW010へ格納します。











[ロング]



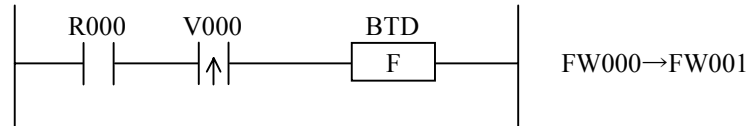
入力条件R000がOFF→ONの変化時、1回のみDL000の内容と同一データがFL000から5ステップ内に存在するかどうか探し、その結果をDW010へ格納します。



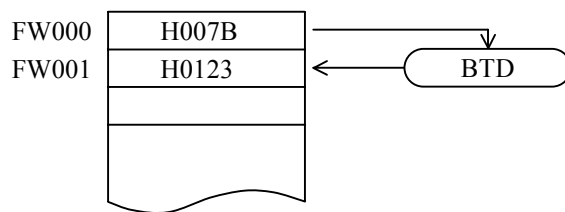
BTD BINARY→BCD変換

機能	ソースの内容をBIN（バイナリ）からBCDへ変換し、結果をリザルトへ格納します。 BCD：2進化10進数（Binary Coded Decimal）									
パラメータ の型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	×	○	×	-	↕	-	-	-	↕
パラメータ	ワード	定数なし		SW→RW						
		定数あり								
	ロング	定数なし		SL→DL						
		定数あり								
		S：ソース格納アドレス				W：ワード				
		R：リザルト（演算結果）を格納するアドレス				L：ロング				
処理内容	ワード	(S) W $\xrightarrow{\text{BIN} \rightarrow \text{BCD}}$ (R) W								
	ロング	(S) L $\xrightarrow{\text{BIN} \rightarrow \text{BCD}}$ (R) L								
入力手順	 + )    									
フラグの 設定	E：(S) <0のとき1、それ以外0 V：(ワードのとき) (S) >9999のとき1、それ以外0 (ロングワードのとき) (S) >99999999のとき1、それ以外0 他：保持									
注意事項	<ul style="list-style-type: none"> ・(S) <0のときエラーフラグ (E) がON（オーバーフローフラグ (V) OFF）し、未処理（リザルトは変化しません）。 ・オーバーフロー発生時、リザルトには下記フルスケール値が格納されます。 									
			ワード		ロングワード					
			H9999		H99999999					

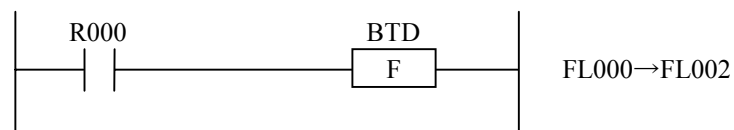
[ワード]



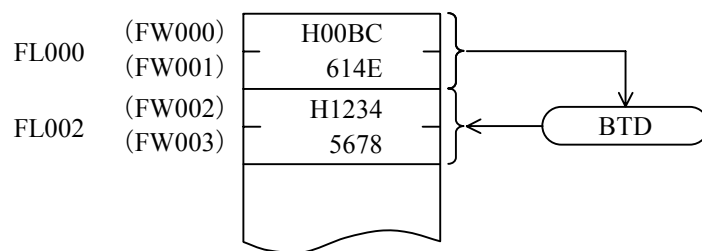
入力条件R000がOFF→ONの変化時、1回のみFW000の内容をBINからBCDへ変換し、その結果をFW001へ格納します。



[ロング]



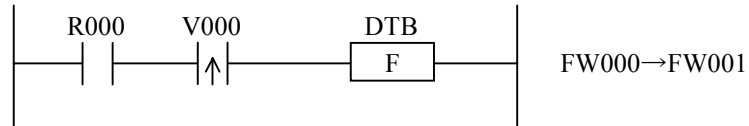
入力条件R000がONのとき、FL000の内容をBINからBCDへ変換し、その結果をFL002へ格納します。



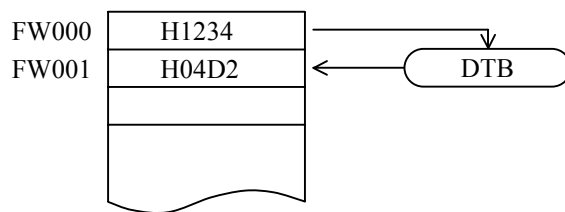
DTB BCD→BINARY変換

機能	ソースの内容をBCDからBIN（バイナリ）へ変換し、結果をリザルトへ格納します。																																																																								
パラメータ の型	ワード		ロング		フラグ																																																																				
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V																																																															
	○	×	○	×	-	↑ ↓	-	-	-	-																																																															
パラメータ	ワード	定数なし	DTB F	SW→RW																																																																					
		定数あり																																																																							
	ロング	定数なし	DTB F	SL→DL																																																																					
		定数あり																																																																							
S：ソース格納アドレス		W：ワード																																																																							
R：リザルト（演算結果）を格納するアドレス		L：ロング																																																																							
処理内容	ワード	(S) W $\xrightarrow{\text{BCD}\rightarrow\text{BIN}}$ (R) W																																																																							
	ロング	(S) L $\xrightarrow{\text{BCD}\rightarrow\text{BIN}}$ (R) L																																																																							
入力手順	(<input type="button" value="Shift"/> + <input type="button" value="F"/>) <input type="button" value="D"/> <input type="button" value="T"/> <input type="button" value="B"/> <input >="" <input="" type="button" value="Enter" パラメータ="" パラメータ,=""/>																																																																								
フラグの 設定	E：ソースデータ内のあるディジット（4ビット）がHA～HFの場合1、それ以外0 他：保持																																																																								
注意事項	上記エラーフラグ（E）がONした場合、未処理（リザルトは変化しません）。																																																																								
	<ディジットについて>																																																																								
	<table border="1" style="width:100%; text-align:center; border-collapse: collapse;"> <tr> <td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td> </tr> <tr> <td style="border: none;"> </td><td style="border: none;"> </td><td style="border: none;"> </td><td style="border: none;"> </td><td style="border: none;"> </td><td style="border: none;"> </td><td style="border: none;"> </td><td style="border: none;"> </td><td style="border: none;"> </td><td style="border: none;"> </td><td style="border: none;"> </td><td style="border: none;"> </td><td style="border: none;"> </td><td style="border: none;"> </td><td style="border: none;"> </td><td style="border: none;"> </td> </tr> <tr> <td colspan="4">└──────────┘</td> <td colspan="4">└──────────┘</td> <td colspan="4">└──────────┘</td> <td colspan="4">└──────────┘</td> </tr> <tr> <td colspan="4">ディジット</td> <td colspan="4">ディジット</td> <td colspan="4">ディジット</td> <td colspan="4">ディジット</td> </tr> </table>										0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15																	└──────────┘				└──────────┘				└──────────┘				└──────────┘				ディジット				ディジット				ディジット				ディジット		
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15																																																										
└──────────┘				└──────────┘				└──────────┘				└──────────┘																																																													
ディジット				ディジット				ディジット				ディジット																																																													

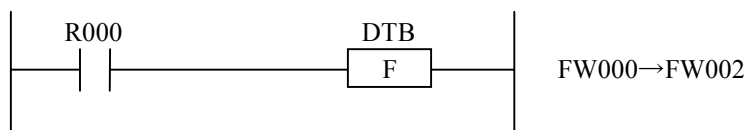
[ワード]



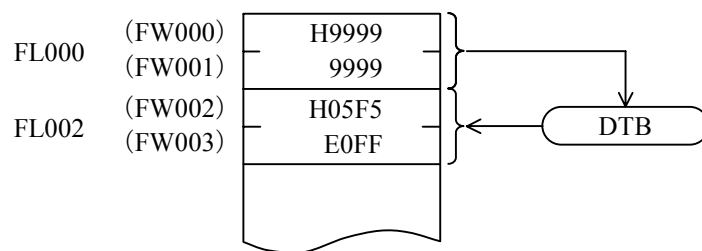
入力条件R000がOFF→ONの変化時、1回のみFW000の内容をBCDからBINへ変換し、その結果をFW001へ格納します。



[ロング]



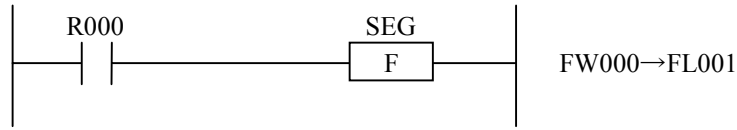
入力条件R000がONのとき、FL000の内容をBCDからBINへ変換し、その結果をFL002へ格納します。



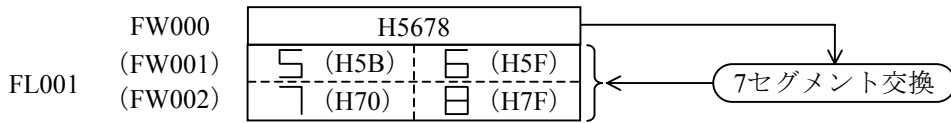
SEG BINARY→セグメント変換

機能	ソースの内容またはイミディエートデータをBIN（バイナリ）から7セグメントデータへ変換し、リザルトへ格納します。																																																												
パラメータの型	ワード		ロング		フラグ																																																								
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V																																																			
	○	○	○	×	-	-	-	-	-	-																																																			
パラメータ	ワード	定数なし	SEG F	SW→RW																																																									
		定数あり	SEG F	nW→RW																																																									
	ロング	定数なし	SEG F	SL→RL																																																									
		定数あり																																																											
S：ソース格納アドレス					W：ワード																																																								
n：イミディエートデータ					L：ロング																																																								
R：リザルト（演算結果）を格納するアドレス																																																													
処理内容	ワード	(S) W $\xrightarrow{\text{BIN} \rightarrow 7\text{セグメント}}$ (R, R+1) W																																																											
	ワード定数	n W $\xrightarrow{\text{BIN} \rightarrow 7\text{セグメント}}$ (R, R+1) W																																																											
	ロング	(S) L $\xrightarrow{\text{BIN} \rightarrow 7\text{セグメント}}$ { (R) L, (R+2) L }																																																											
入力手順																																																													
セグメントデータ構成	<p>(MSB) 0 1 2 3 4 5 6 7 \Rightarrow </p> <ul style="list-style-type: none"> ・該当ビットがONのとき表示あり ・7グメント対応表 <table border="1"> <tr> <td>No.</td> <td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td> </tr> <tr> <td>表示</td> <td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> <tr> <td>データ</td> <td>H7E</td><td>H30</td><td>H6D</td><td>H79</td><td>H33</td><td>H5B</td><td>H5F</td><td>H70</td><td>H7F</td><td>H7B</td><td>H77</td><td>H1F</td><td>H4E</td><td>H3D</td><td>H4F</td><td>H47</td> </tr> </table>										No.	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	表示																	データ	H7E	H30	H6D	H79	H33	H5B	H5F	H70	H7F	H7B	H77	H1F	H4E	H3D	H4F	H47
No.	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F																																													
表示																																																													
データ	H7E	H30	H6D	H79	H33	H5B	H5F	H70	H7F	H7B	H77	H1F	H4E	H3D	H4F	H47																																													
注意事項	フラグはすべて保持。																																																												

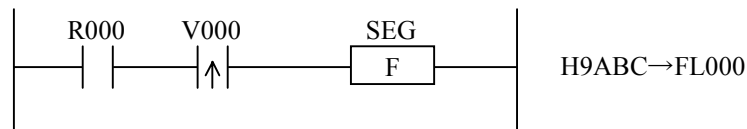
[ワード]



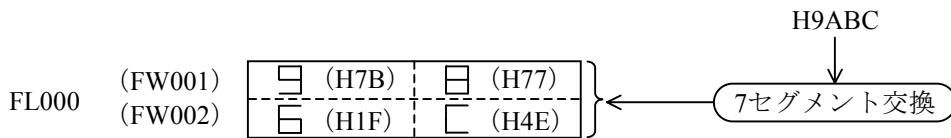
入力条件R000がONのとき、FW000の内容をバイナリデータから4文字の7セグメントデータに変換し、FL001へ格納します。



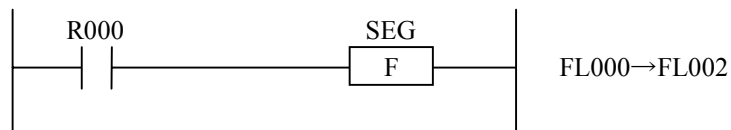
[ワード定数]



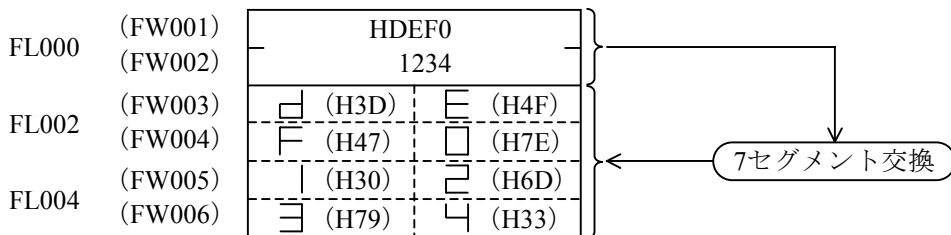
入力条件R000がOFF→ONの変化時、1回のみバイナリデータH9ABCを4文字の7セグメントデータに変換し、FL000へ格納します。



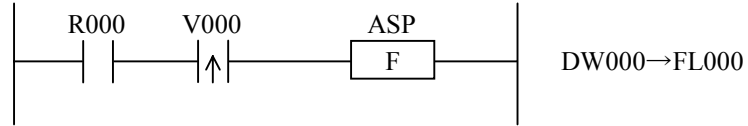
[ロング]



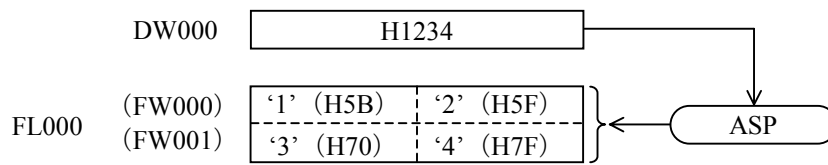
入力条件R000がONのとき、FL000の内容をバイナリデータから8文字の7セグメントデータに変換し、FL002へ格納します。



[ワード]



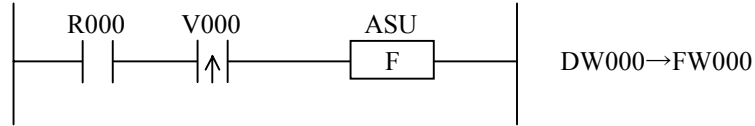
入力条件R000がOFF→ONの変化時、1回のみDW000の内容をバイナリデータからASCIIデータへ変換して、FL000へ格納します。



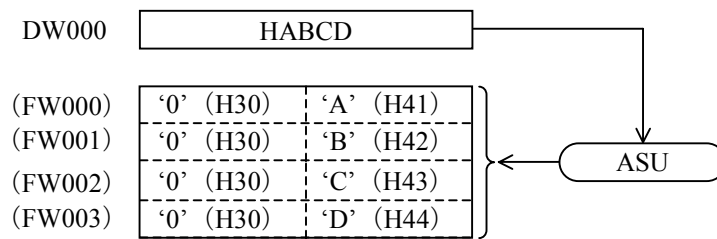
ASU BINARY→ASCII変換アンパックモード

機能	ソースの内容をバイナリデータからASCIIデータへ変換し、アンパックモードで、結果をリザルトへ格納します。																																											
パラメータ の型	ワード		ロング		フラグ																																							
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V																																		
	○	×	×	×	-	-	-	-	-	-																																		
パラメータ	ワード	定数なし	ASU F	SW→RW																																								
		定数あり																																										
	ロング	定数なし																																										
		定数あり																																										
S: ソース格納アドレス		W: ワード																																										
R: リザルト (演算結果) を格納するアドレス		L: ロング																																										
処理内容	ワード	(S) W $\xrightarrow[\text{アンパックモード}]{\text{BIN}\rightarrow\text{ASCII}}$ (R, R+1, R+2, R+3) W																																										
入力手順	(<input type="button" value="Shift"/> + <input type="button" value="F"/>) <input type="button" value="A"/> <input type="button" value="S"/> <input type="button" value="U"/> <input >="" <input="" type="button" value="Enter" パラメータ="" パラメータ,=""/>																																											
ASCII 対応表	<table border="1"> <tr> <td>バイナリ</td> <td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td> </tr> <tr> <td>ASCII</td> <td>H30</td><td>H31</td><td>H32</td><td>H33</td><td>H34</td><td>H35</td><td>H36</td><td>H37</td><td>H38</td><td>H39</td> </tr> </table> <table border="1"> <tr> <td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td> </tr> <tr> <td>H41</td><td>H42</td><td>H43</td><td>H44</td><td>H45</td><td>H46</td> </tr> </table>										バイナリ	0	1	2	3	4	5	6	7	8	9	ASCII	H30	H31	H32	H33	H34	H35	H36	H37	H38	H39	A	B	C	D	E	F	H41	H42	H43	H44	H45	H46
バイナリ	0	1	2	3	4	5	6	7	8	9																																		
ASCII	H30	H31	H32	H33	H34	H35	H36	H37	H38	H39																																		
A	B	C	D	E	F																																							
H41	H42	H43	H44	H45	H46																																							
注意事項	<ul style="list-style-type: none"> ・フラグはすべて保持。 ・結果は1バイトずつ上位データからR, R+1, R+2, R+3の下位バイトへ格納します。R~R+3の上位バイトはASCIIの '0' (H30) を格納します。 																																											

[ワード]



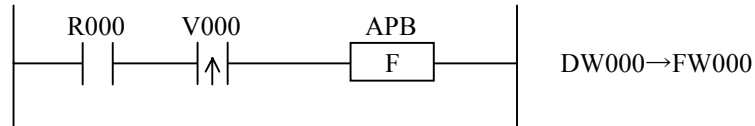
入力条件R000がOFF→ONの変化時、1回のみDW000の内容をバイナリデータからASCIIデータへ変換して、アンパックモードでFW000へ格納します。



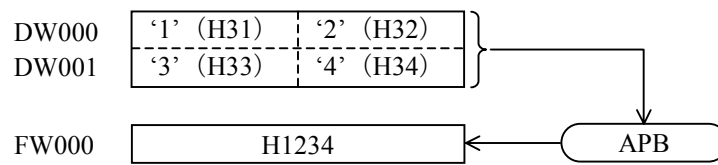
APB ASCII→BINARY変換パックモード

機能	ソースの内容をASCIIデータ（パックモード）からバイナリデータへ変換し、結果をリザルトへ格納します。																																											
パラメータの型	ワード		ロング		フラグ																																							
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V																																		
	○	×	×	×	-	↑ ↓	-	-	-	-																																		
パラメータ	ワード	定数なし	APB F	SW→RW																																								
		定数あり																																										
	ロング	定数なし																																										
		定数あり																																										
		S：ソース格納アドレス				W：ワード																																						
		R：リザルト（演算結果）を格納するアドレス				L：ロング																																						
処理内容	ワード	$(S, S+1) W \xrightarrow[\text{パックモード}]{\text{ASCII} \rightarrow \text{BIN}} (R) W$																																										
入力手順	$\left(\left[\text{Shift} \right] + \left[\text{F} \right] \right) \left[\text{A} \right] \left[\text{P} \right] \left[\text{B} \right] \square \text{ パラメータ, パラメータ } \left[\text{Enter} \right]$																																											
ASCII 対応表	<table border="1" style="margin-bottom: 10px;"> <tr> <td>バイナリ</td> <td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td> </tr> <tr> <td>ASCII</td> <td>H30</td><td>H31</td><td>H32</td><td>H33</td><td>H34</td><td>H35</td><td>H36</td><td>H37</td><td>H38</td><td>H39</td> </tr> </table> <table border="1" style="margin-bottom: 10px;"> <tr> <td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td> </tr> <tr> <td>H41</td><td>H42</td><td>H43</td><td>H44</td><td>H45</td><td>H46</td> </tr> </table>										バイナリ	0	1	2	3	4	5	6	7	8	9	ASCII	H30	H31	H32	H33	H34	H35	H36	H37	H38	H39	A	B	C	D	E	F	H41	H42	H43	H44	H45	H46
バイナリ	0	1	2	3	4	5	6	7	8	9																																		
ASCII	H30	H31	H32	H33	H34	H35	H36	H37	H38	H39																																		
A	B	C	D	E	F																																							
H41	H42	H43	H44	H45	H46																																							
フラグの 設定	E：H30～H39, H41～H46以外のデータ検出時1、それ以外0 他：保持																																											
注意事項	エラーフラグ（E）がONしたとき、未処理（リザルトは変化しません）。																																											

[ワード]



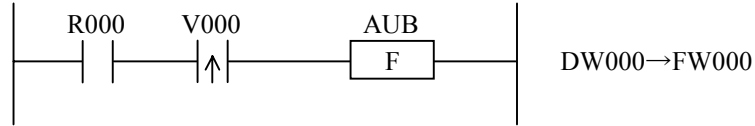
入力条件R000がOFF→ONの変化時、1回のみDW000の内容をASCIIデータ（パックモード）からバイナリデータへ変換し、その結果をリザルトへ格納します。



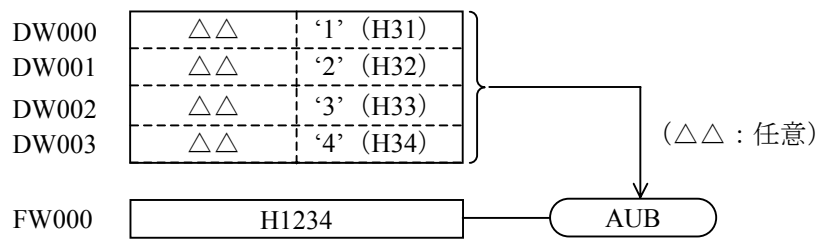
AUB ASCII→BINARY変換アンパックモード

機能	ソースの内容をASCIIデータ（アンパックモード）からバイナリデータへ変換し、結果をリザルトへ格納します。																																											
パラメータの型	ワード		ロング		フラグ																																							
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V																																		
	○	×	×	×	-	↑ ↓	-	-	-	-																																		
パラメータ	ワード	定数なし	AUB F	SW→RW																																								
		定数あり																																										
	ロング	定数なし																																										
		定数あり																																										
	S : ソース格納アドレス				W : ワード																																							
	R : リザルト（演算結果）を格納するアドレス				L : ロング																																							
処理内容	ワード	$(S, S+1, S+2, S+3) \text{ W} \xrightarrow[\text{アンパックモード}]{\text{ASCII} \rightarrow \text{BIN}} (R) \text{ W}$																																										
入力手順	(<input type="button" value="Shift"/> + <input type="button" value="F"/>) <input type="button" value="A"/> <input type="button" value="U"/> <input type="button" value="B"/> <input type="button" value="□"/> パラメータ, パラメータ <input type="button" value="Enter"/>																																											
ASCII 対応表	<table border="1" style="margin-bottom: 10px;"> <tr> <td>バイナリ</td> <td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td> </tr> <tr> <td>ASCII</td> <td>H30</td><td>H31</td><td>H32</td><td>H33</td><td>H34</td><td>H35</td><td>H36</td><td>H37</td><td>H38</td><td>H39</td> </tr> </table> <table border="1" style="margin-bottom: 10px;"> <tr> <td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td> </tr> <tr> <td>H41</td><td>H42</td><td>H43</td><td>H44</td><td>H45</td><td>H46</td> </tr> </table>										バイナリ	0	1	2	3	4	5	6	7	8	9	ASCII	H30	H31	H32	H33	H34	H35	H36	H37	H38	H39	A	B	C	D	E	F	H41	H42	H43	H44	H45	H46
バイナリ	0	1	2	3	4	5	6	7	8	9																																		
ASCII	H30	H31	H32	H33	H34	H35	H36	H37	H38	H39																																		
A	B	C	D	E	F																																							
H41	H42	H43	H44	H45	H46																																							
フラグの設定	E : H30~H39, H41~H46以外のデータ検出時1、それ以外0 他 : 保持																																											
注意事項	<ul style="list-style-type: none"> ・エラーフラグ (E) がONしたとき、未処理（リザルトは変化しません）。 ・S~S+3の上位バイトは任意。 																																											

[ワード]



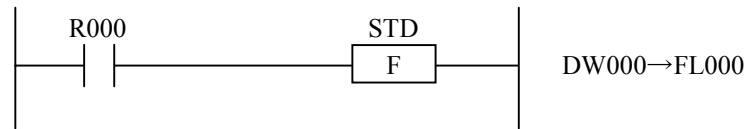
入力条件R000がOFF→ONの変化時、1回のみDW000の内容をASCIIデータ（アンパックモード）からバイナリデータへ変換して、その結果をFW000へ格納します。



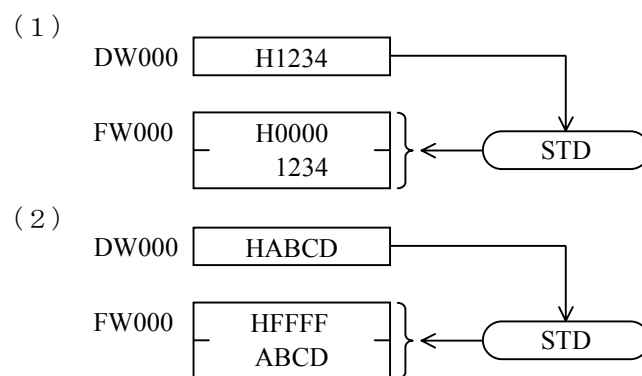
STD SINGLE→DOUBLE変換

機能	ソースの内容をワードデータからロングワードデータに符号拡張してリザルトへ格納します。									
パラメータ の型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	×	×	×	-	-	-	-	-	-
パラメータ	ワード	定数なし	STD F	SW→RL						
		定数あり								
	ロング	定数なし								
		定数あり								
	S: ソース格納アドレス		W: ワード							
R: リザルト (演算結果) を格納するアドレス		L: ロング								
処理内容	ワード	(S) W→ (R) L								
入力手順	(Shift + F) S T D □ パラメータ, パラメータ Enter									
注意事項	フラグはすべて保持。									

[ワード]



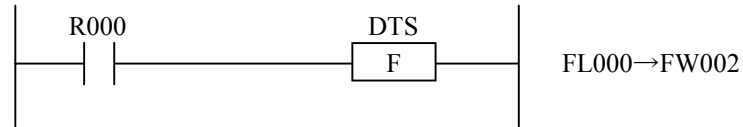
入力条件R000がONのとき、DW000の内容をワードデータからロングワードデータに符号拡張して、その結果をFL000へ格納します。



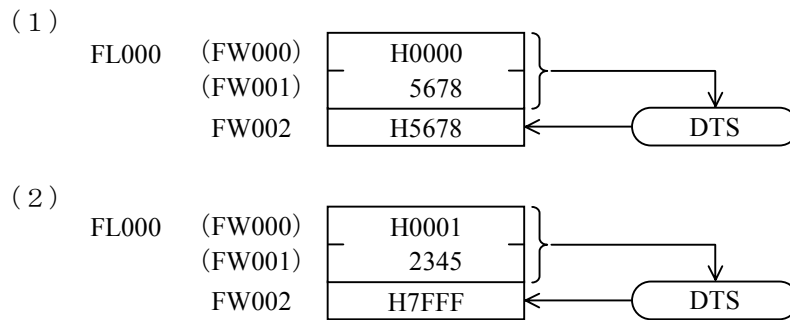
DTS DOUBLE→SINGLE変換

機能	ソースの内容のロングワードデータをワードデータとしてリザルトへ格納します。													
パラメータ の型	ワード		ロング		フラグ									
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V				
	×	×	○	×	-	-	-	-	-	↑ ↓				
パラメータ	ワード	定数なし												
		定数あり												
	ロング	定数なし	DTS - F	SL→RW										
		定数あり												
S : ソース格納アドレス		W : ワード												
R : リザルト (演算結果) を格納するアドレス		L : ロング												
処理内容	ワード	(S) L → (R) W												
入力手順	(Shift + F) D T S □ パラメータ, パラメータ Enter													
フラグの 設定	V : (S) < -32768 または 32767 < (S) のとき 1、それ以外 0 他 : 保持													
注意事項	オーバーフロー発生時、リザルトには下記フルスケール値が格納されます。 <table border="1" style="margin: 10px auto;"> <tr> <td>(S) L > 32767 のとき</td> <td>H7FFF</td> </tr> <tr> <td>(S) L < -32768 のとき</td> <td>H8000</td> </tr> </table>										(S) L > 32767 のとき	H7FFF	(S) L < -32768 のとき	H8000
(S) L > 32767 のとき	H7FFF													
(S) L < -32768 のとき	H8000													

[ワード]



入力条件R000がONのとき、FL000の内容のロングワードデータをワードデータとしてFW002へ格納します。



このときVフラグがONします。

Vフラグ

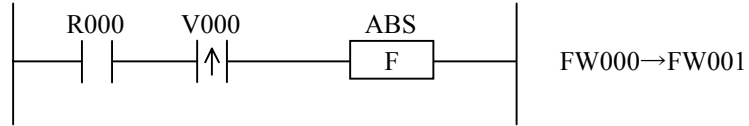
ON

 S005

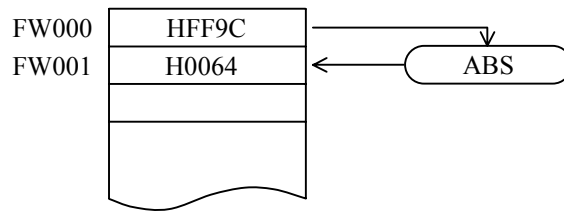
ABS 絶対値 : ABSOLUTE VALUE

機能	ソースの内容の絶対値をリザルトへ格納します。									
パラメータ の型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	×	○	×	-	-	-	-	-	↕
パラメータ	ワード	定数なし								
		定数あり								
	ロング	定数なし								
		定数あり								
S : ソース格納アドレス		W : ワード		R : リザルト (演算結果) を格納するアドレス		L : ロング				
処理内容	ワード	(S) W → (R) W								
	ロング	(S) L → (R) L								
入力手順	(Shift + F) A B S □ パラメータ, パラメータ Enter									
フラグの 設定	V : (ワードのとき) (S) = -32768のとき1、それ以外0 (ロングワードのとき) (S) = -2147483648のとき1、それ以外0 他 : 保持									
注意事項	オーバーフロー発生時、リザルトには下記フルスケール値が格納されます。									
	ワード		ロングワード							
	H7FFF		H7FFFFFFF							

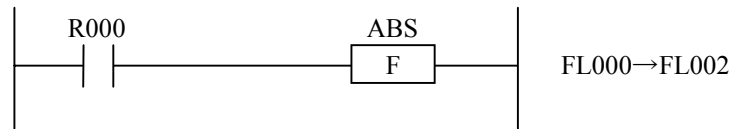
[ワード]



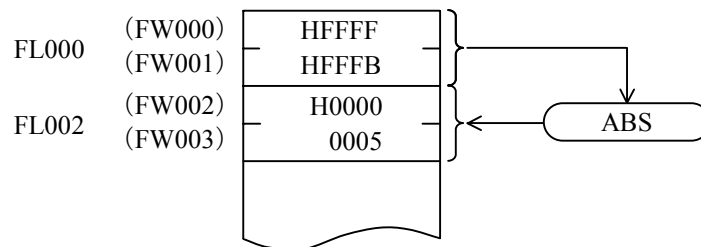
入力条件R000がOFF→ONの変化時、1回のみFW000の内容の絶対値を取り、FW001へ格納します。



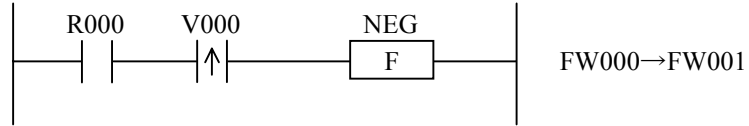
[ロング]



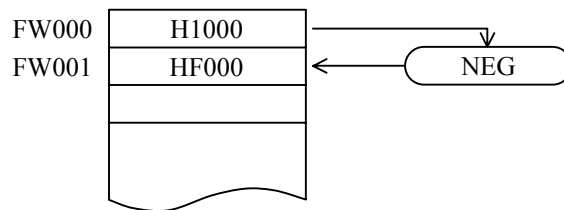
入力条件R000がONのとき、FL000の内容の絶対値を取り、FL002へ格納します。



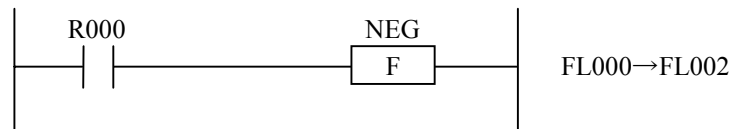
[ワード]



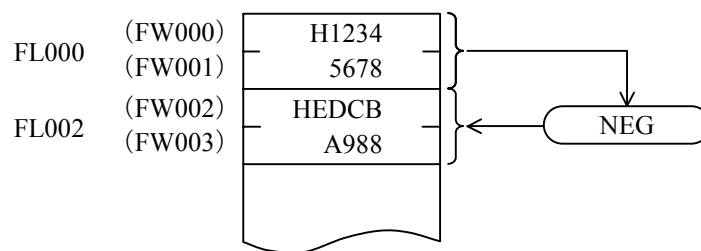
入力条件R000がOFF→ONの変化時、1回のみFW000の内容の符号変換を行い、その結果をFW001へ格納します。



[ロング]



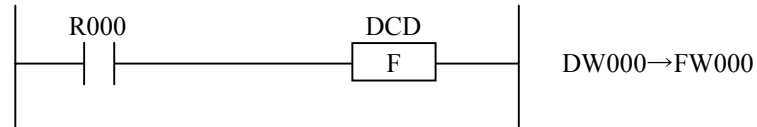
入力条件R000がONのとき、FL000の内容の符号変換を行い、その結果をFL002へ格納します。



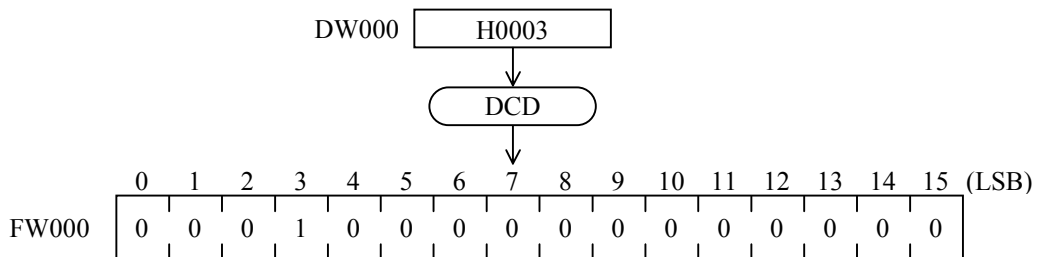
DCD デコード : DECODE

機能	ソースの内容をデコードして、結果をリザルトへ格納します。									
パラメータ の型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	×	○	×	-	-	-	-	-	-
パラメータ	ワード	定数なし								
		定数あり								
	ロング	定数なし								
		定数あり								
S : ソース格納アドレス					W : ワード					
R : リザルト (演算結果) を格納するアドレス					L : ロング					
処理内容	ワード	$\text{SW} \begin{array}{ c c } \hline 12 & 15 \\ \hline & n \\ \hline \end{array} \rightarrow \text{RW} \begin{array}{ c c c } \hline 0 & n & 15(\text{LSB}) \\ \hline 0 \sim 0 & 1 & 0 \sim 0 \\ \hline \end{array}$								
	ロング	$\text{SW} \begin{array}{ c c } \hline 11 & 15 \\ \hline & n \\ \hline \end{array} \rightarrow \text{RL} \begin{array}{ c c c } \hline 0 & n & 31(\text{LSB}) \\ \hline 0 \sim 0 & 1 & 0 \sim 0 \\ \hline \end{array}$								
入力手順	$\left(\text{Shift} + \text{F} \right) \text{D} \text{C} \text{D} \text{ } \square \text{ パラメータ, パラメータ } \text{Enter}$									
注意事項	<ul style="list-style-type: none"> フラグはすべて保持。 ソースデータはワード時下位4ビット、ロングワード時下位5ビットのみ有効。 									

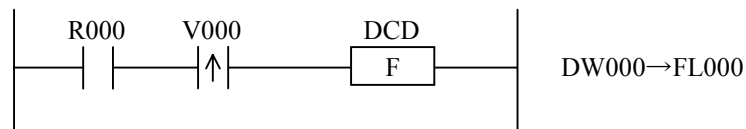
[ワード]



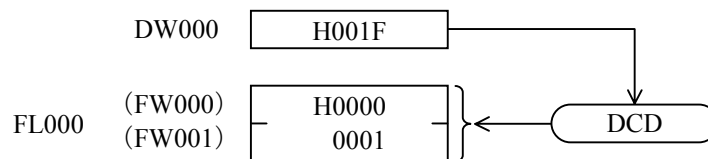
入力条件R000がONのとき、DW000の内容をデコードして、その結果をFW000へ格納します。



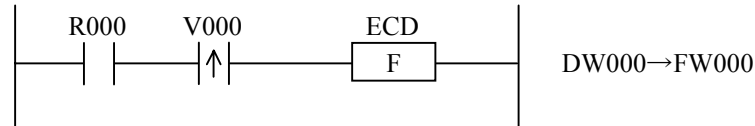
[ロング]



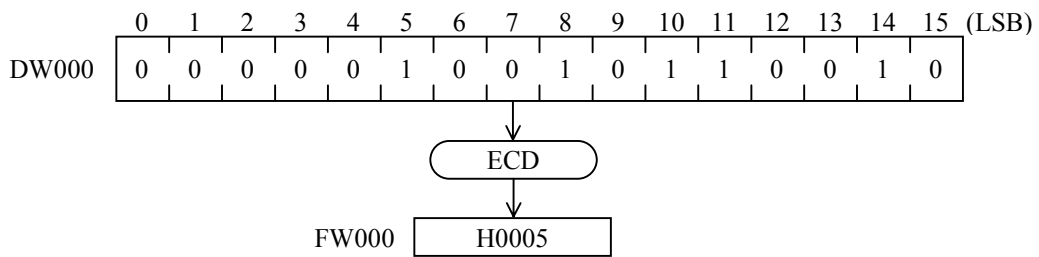
入力条件R000がOFF→ONの変化時、1回のみDW000の内容をデコードしてその結果をFL000へ格納します。



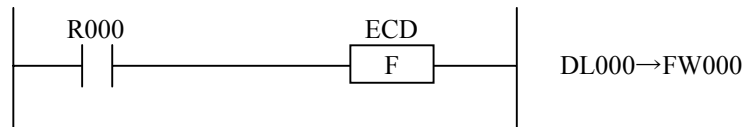
[ワード]



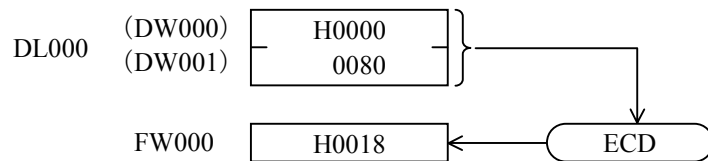
入力条件R000がOFF→ONの変化時、1回のみDW000の内容をエンコードして、その結果をFW000へ格納します。



[ロング]



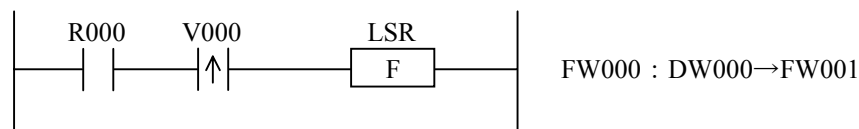
入力条件R000がONのとき、DL000の内容をエンコードして、その結果をFW000へ格納します。



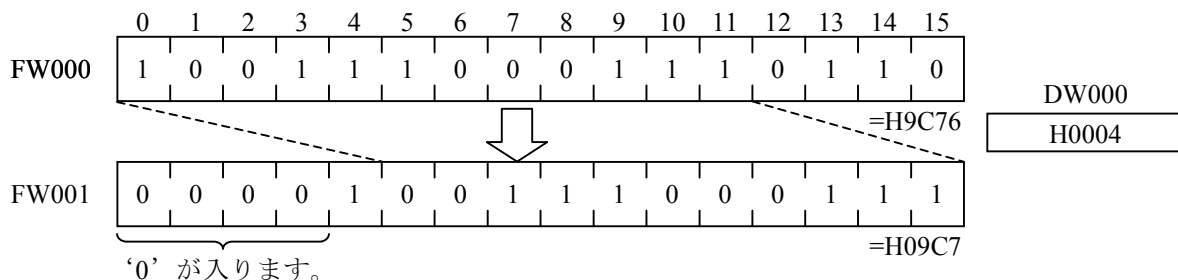
LSR 論理右シフト : LOGICAL SHIFT RIGHT

機能	ソースの内容をデスティネーションの内容またはイミディエートデータの指定ビット数、右へシフトし、結果をリザルトへ格納します。									
パラメータの型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	○	○	○	-	-	-	-	-	-
パラメータ	ワード	定数なし	LSR F	SW : DW→RW						
		定数あり	LSR F	SW : nW→RW						
	ロング	定数なし	LSR F	SL : DW→RL						
		定数あり	LSR F	SL : nW→RL						
S : ソース格納アドレス					W : ワード					
D : デスティネーション格納アドレス					L : ロング					
n : イミディエートデータ										
R : リザルト (演算結果) を格納するアドレス										
処理内容	ワード	(S) W								
	ワード定数	(S) W								
	ロング	(S) L								
	ロング定数	(S) L								
入力手順	(Shift + F) L S R □ パラメータ, パラメータ, パラメータ Enter									
注意事項	<ul style="list-style-type: none"> ・フラグはすべて保持。 ・シフトビット数はワード時下位4ビット、ロングワード時下位5ビットのみ有効。 									

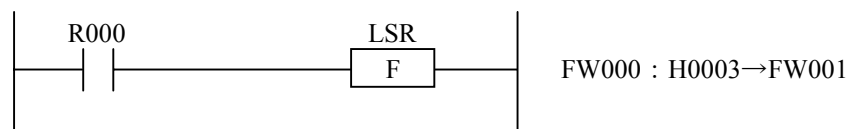
[ワード]



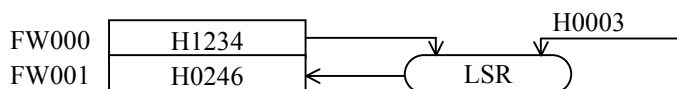
入力条件R000がOFF→ONの変化時、1回のみFW000の内容をDW000で指定された数だけ、右へシフトし、その結果をFW001へ格納します。



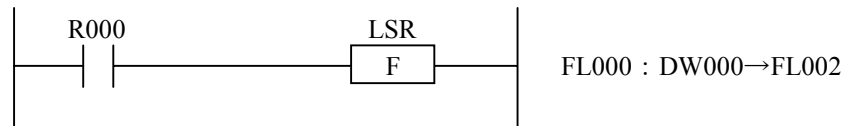
[ワード定数]



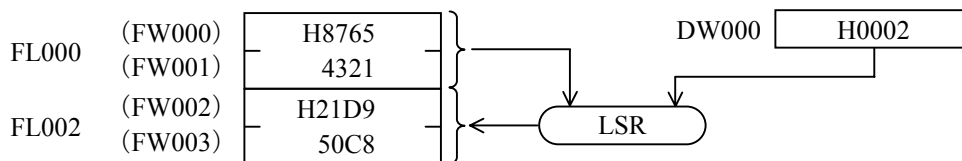
入力条件R000がONのとき、FW000の内容を3ビット右へシフトし、その結果をFW001へ格納します。



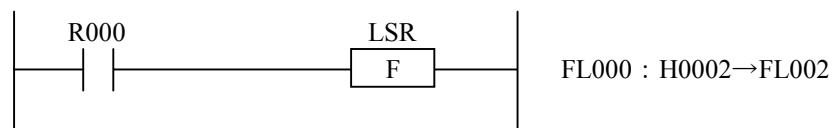
[ロング]



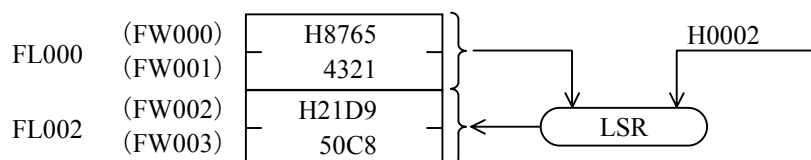
入力条件R000がONのとき、FL000の内容をDW000で指定された数だけ、右へシフトし、その結果をFL002へ格納します。



[ロング定数]



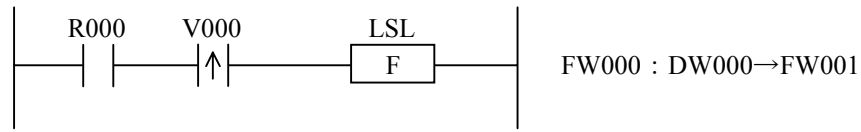
入力条件R000がONのとき、FL000の内容を2ビット右へシフトし、その結果をFL002へ格納します。



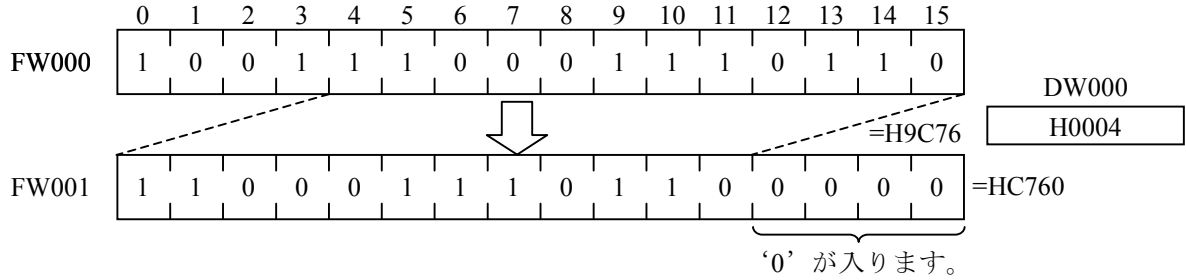
LSL 論理左シフト : LOGICAL SHIFT LEFT

機能	ソースの内容をデスティネーションの内容またはイミディエートデータの指定ビット数、左へシフトし、結果をリザルトへ格納します。									
パラメータの型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	○	○	○	-	-	-	-	-	-
パラメータ	ワード	定数なし	LSL F	SW : DW→RW						
		定数あり	LSL F	SW : nW→RW						
	ロング	定数なし	LSL F	SL : DW→RL						
		定数あり	LSL F	SL : nW→RL						
S : ソース格納アドレス					W : ワード					
D : デスティネーション格納アドレス					L : ロング					
n : イミディエートデータ										
R : リザルト (演算結果) を格納するアドレス										
処理内容	ワード	(S) W								
	ワード定数	(S) W								
	ロング	(S) L								
	ロング定数	(S) L								
入力手順	(Shift + F) L S L □ パラメータ, パラメータ, パラメータ Enter									
注意事項	<ul style="list-style-type: none"> ・フラグはすべて保持。 ・シフトビット数はワード時下位4ビット、ロングワード時下位5ビットのみ有効。 									

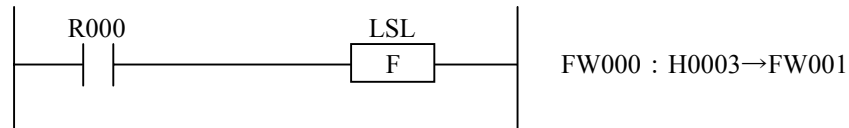
[ワード]



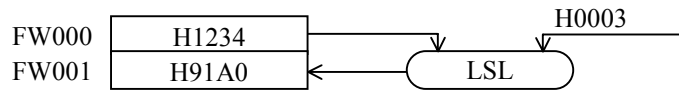
入力条件R000がOFF→ONの変化時、1回のみFW000の内容をDW000で指定された数だけ、左へシフトし、その結果をFW001へ格納します。



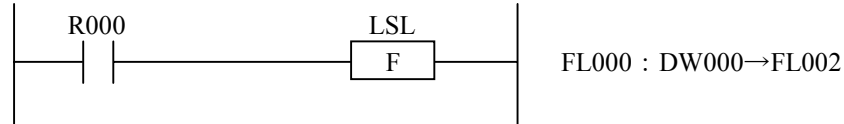
[ワード定数]



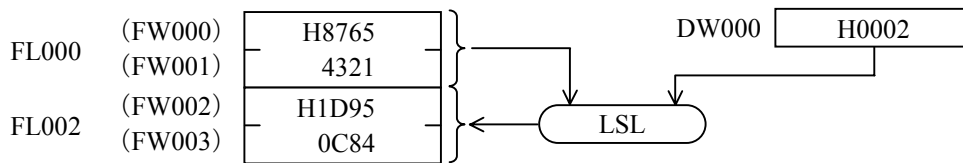
入力条件R000がONのとき、FW000の内容を3ビット左へシフトし、その結果をFW001へ格納します。



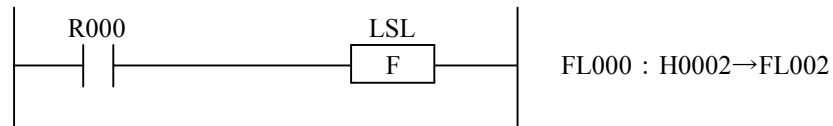
[ロング]



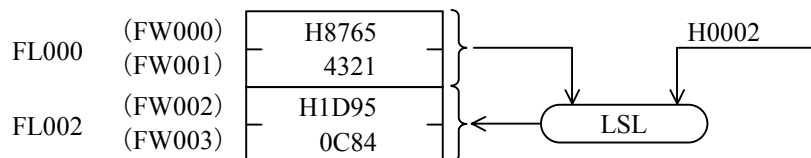
入力条件R000がONのとき、FL000の内容をDW000で指定された数だけ、左へシフトし、その結果をFL002へ格納します。



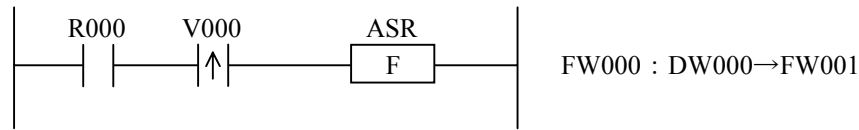
[ロング定数]



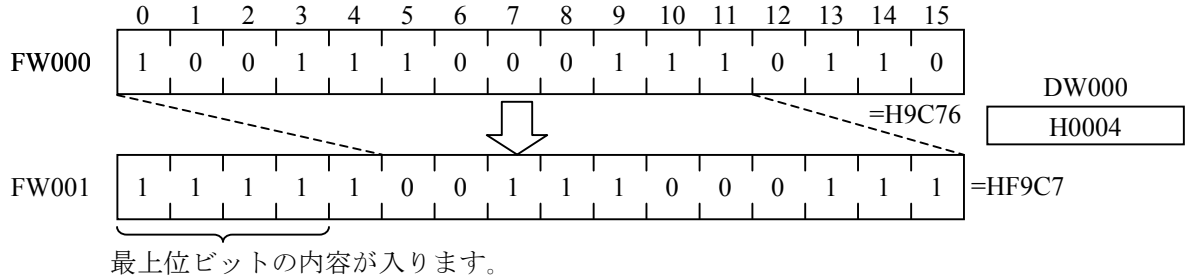
入力条件R000がONのとき、FL000の内容を2ビット左へシフトし、その結果をFL002へ格納します。



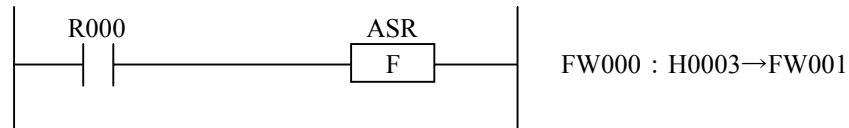
[ワード]



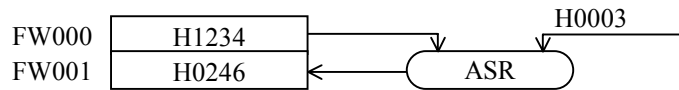
入力条件R000がOFF→ONの変化時、1回のみFW000の内容をDW000で指定された数だけ、右へシフトし、その結果をFW001へ格納します。



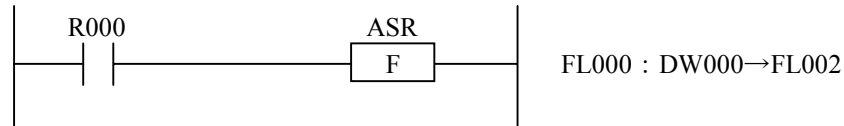
[ワード定数]



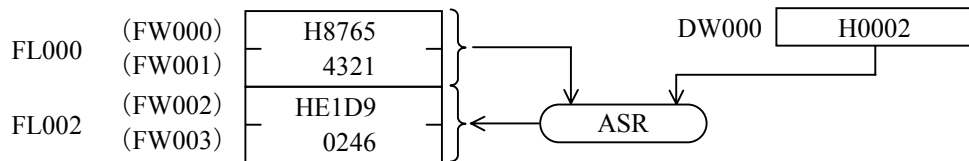
入力条件R000がONのとき、FW000の内容を3ビット右へシフトし、その結果をFW001へ格納します。



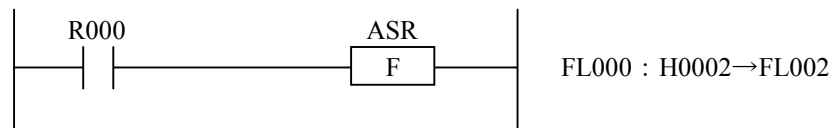
[ロング]



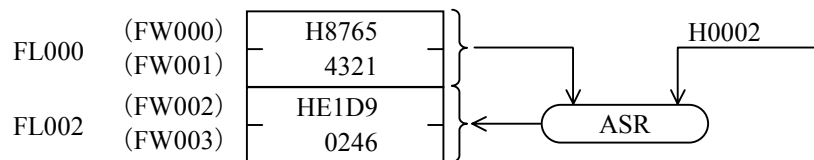
入力条件R000がONのとき、FL000の内容をDW000で指定された数だけ、右へシフトし、その結果をFL002へ格納します。



[ロング定数]



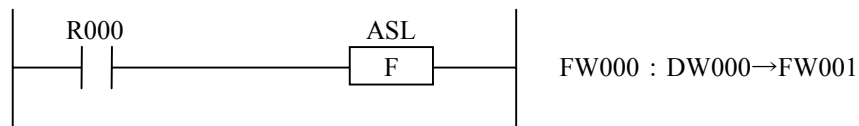
入力条件R000がONのとき、FL000の内容を2ビット右へシフトし、その結果をFL002へ格納します。



ASL 算術左シフト : ARITHMETIC SHIFT LEFT

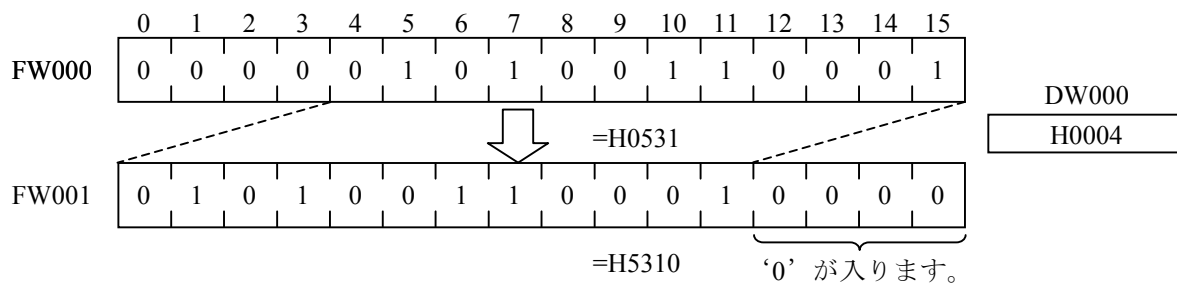
機能	ソースの内容をデスティネーションの内容またはイミディエートデータの指定ビット数、左へシフトし、結果をリザルトへ格納します。									
パラメータの型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	○	○	○	-	-	-	-	-	↕
パラメータ	ワード	定数なし	ASL F	SW : DW→RW						
		定数あり	ASL F	SW : nW→RW						
	ロング	定数なし	ASL F	SL : DW→RL						
		定数あり	ASL F	SL : nW→RL						
S : ソース格納アドレス					W : ワード					
D : デスティネーション格納アドレス					L : ロング					
n : イミディエートデータ										
R : リザルト (演算結果) を格納するアドレス										
処理内容	ワード									
	ワード定数									
	ロング									
	ロング定数									
入力手順	(Shift + F) A S L □ パラメータ, パラメータ, パラメータ Enter									
フラグの設定	V : シフト操作中に符号ビットが1回でも変化すれば1、それ以外0 他 : 保持									
注意事項	オーバーフロー発生時、リザルトには下記フルスケール値が格納されます。									
		ワード	ロングワード							
	(S) > 0 のとき	H7FFF	H7FFFFFFF							
(S) < 0 のとき	H8000	H80000000								

[ワード]

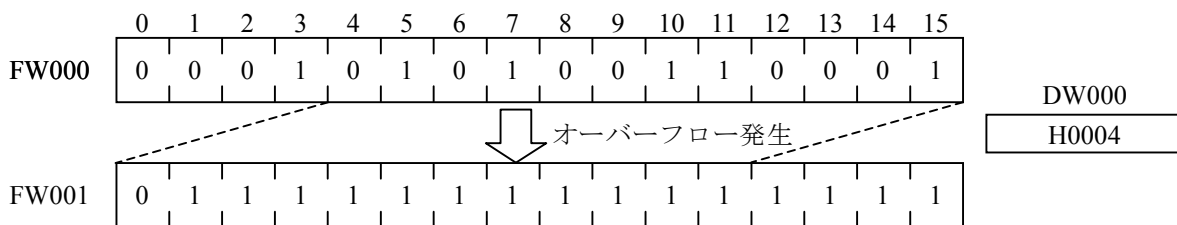


入力条件R000がONのとき、FW000の内容をDW000で指定された数だけ、左へシフトし、その結果をFW001へ格納します。

(1)



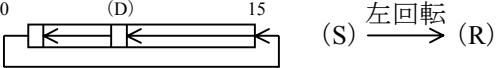
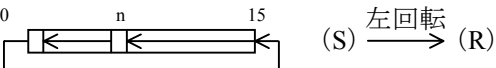
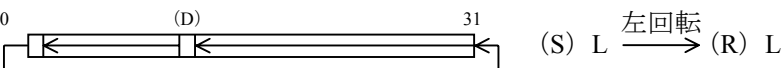
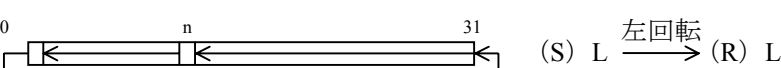
(2)



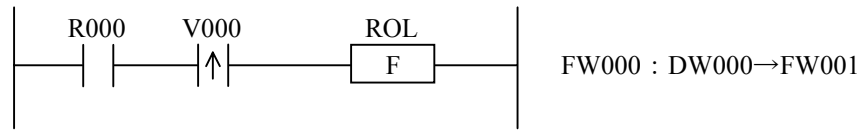
このとき、VフラグがONします。

Vフラグ ON S005

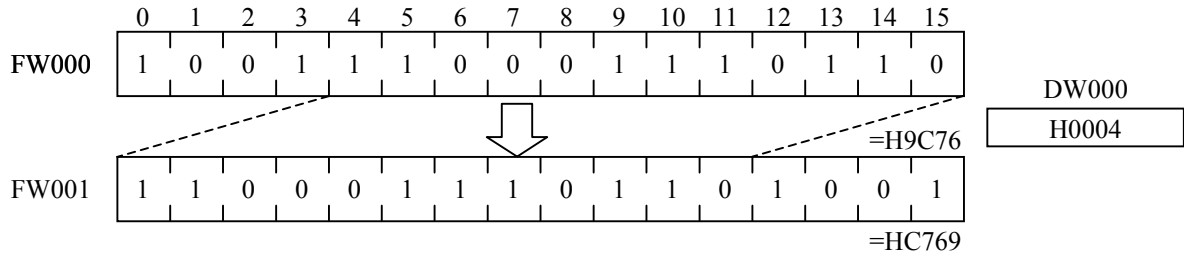
ROL 左回転 : ROTATE LEFT

機能	ソースの内容をデスティネーションの内容またはイミディエートデータの指定ビット数、左へ回転し、結果をリザルトへ格納します。									
パラメータの型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	○	○	○	-	-	-	-	-	-
パラメータ	ワード	定数なし	ROL F	SW : DW→RW						
		定数あり	ROL F	SW : nW→RW						
	ロング	定数なし	ROL F	SL : DW→RL						
		定数あり	ROL F	SL : nW→RL						
S : ソース格納アドレス					W : ワード					
D : デスティネーション格納アドレス					L : ロング					
n : イミディエートデータ										
R : リザルト (演算結果) を格納するアドレス										
処理内容	ワード									
	ワード定数									
	ロング									
	ロング定数									
入力手順	(Shift + F) R O L ┘ パラメータ, パラメータ, パラメータ Enter									
注意事項	<ul style="list-style-type: none"> ・フラグはすべて保持。 ・回転ビット数はワード時下位4ビット、ロングワード時下位5ビットのみ有効。 									

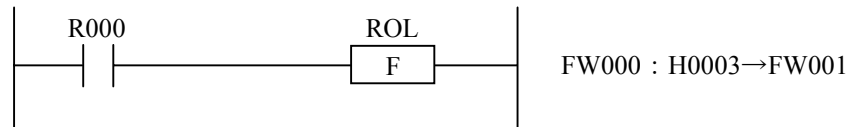
[ワード]



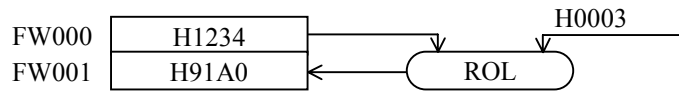
入力条件R000がOFF→ONの変化時、1回のみFW000の内容をDW000で指定された数だけ、左へ回転し、その結果をFW001へ格納します。



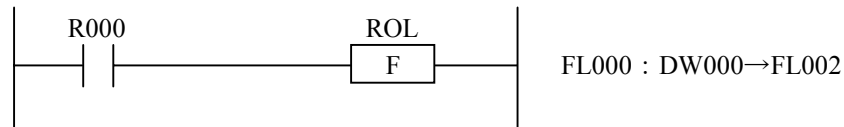
[ワード定数]



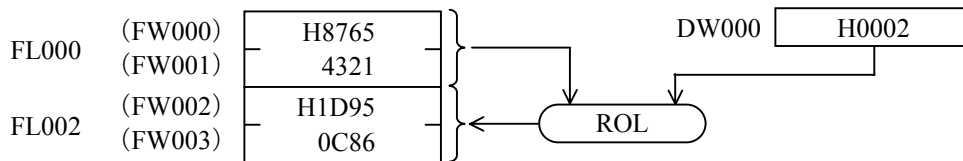
入力条件R000がONのとき、FW000の内容を3ビット左へ回転し、その結果をFW001へ格納します。



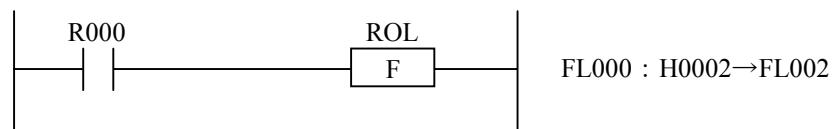
[ロング]



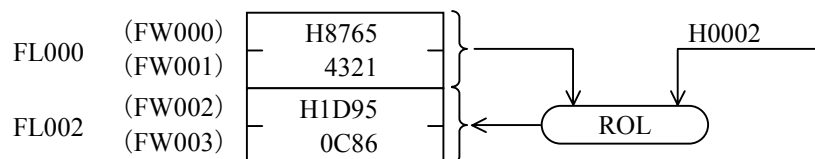
入力条件R000がONのとき、FL000の内容をDW000で指定された数だけ、左へ回転し、その結果をFL002へ格納します。



[ロング定数]



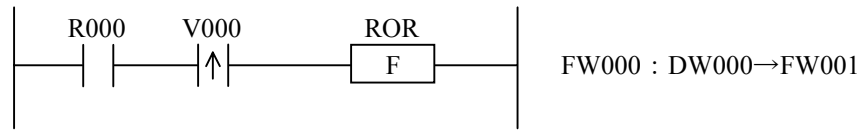
入力条件R000がONのとき、FL000の内容を2ビット左へ回転し、その結果をFL002へ格納します。



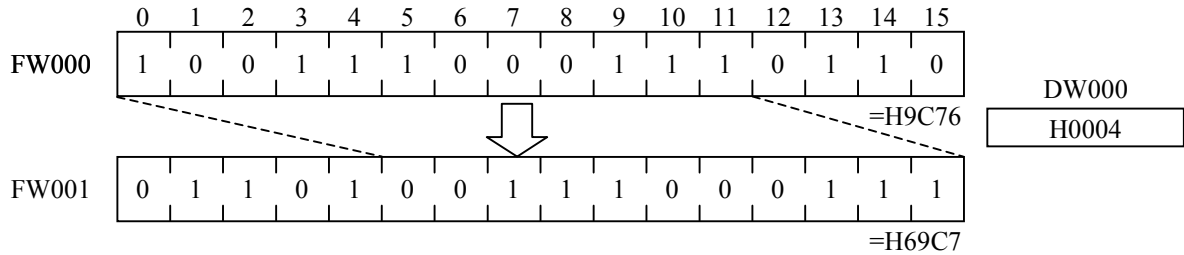
ROR 右回転 : ROTATE RIGHT

機能	ソースの内容をデスティネーションの内容またはイミディエートデータの指定ビット数、右へ回転し、結果をリザルトへ格納します。									
パラメータ の型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	○	○	○	-	-	-	-	-	-
パラメータ	ワード	定数なし	ROR F	SW : DW→RW						
		定数あり	ROR F	SW : nW→RW						
	ロング	定数なし	ROR F	SL : DW→RL						
		定数あり	ROR F	SL : nW→RL						
S : ソース格納アドレス					W : ワード					
D : デスティネーション格納アドレス					L : ロング					
n : イミディエートデータ										
R : リザルト (演算結果) を格納するアドレス										
処理内容	ワード									
	ワード定数									
	ロング									
	ロング定数									
入力手順	(SHIFT + F) R O R □ パラメータ, パラメータ, パラメータ Enter									
注意事項	<ul style="list-style-type: none"> ・フラグはすべて保持。 ・回転ビット数はワード時下位4ビット、ロングワード時下位5ビットのみ有効。 									

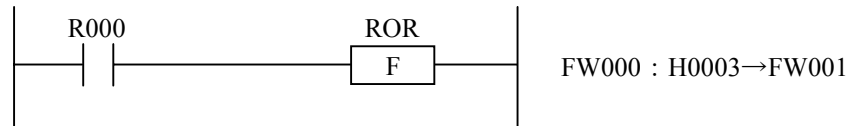
[ワード]



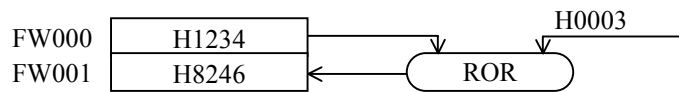
入力条件R000がOFF→ONの変化時、1回のみFW000の内容をDW000で指定された数だけ、右へ回転し、その結果をFW001へ格納します。



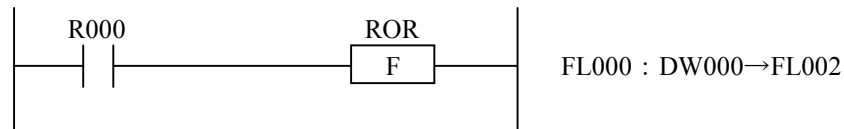
[ワード定数]



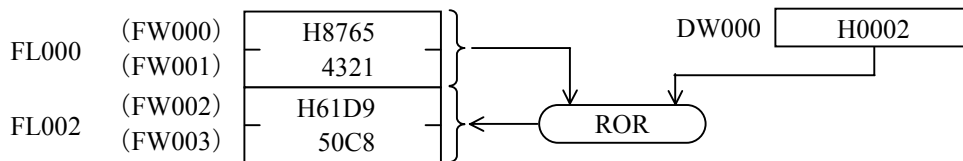
入力条件R000がONのとき、FW000の内容を3ビット右へ回転し、その結果をFW001へ格納します。



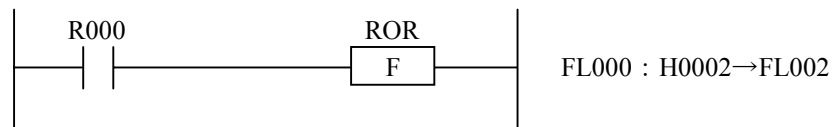
[ロング]



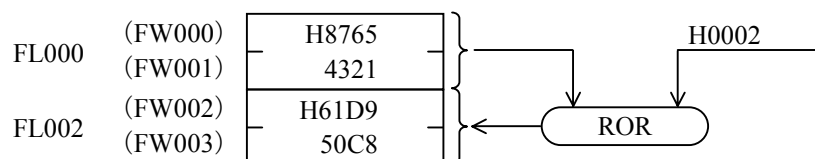
入力条件R000がONのとき、FL000の内容をDW000で指定された数だけ、右へ回転し、その結果をFL002へ格納します。



[ロング定数]



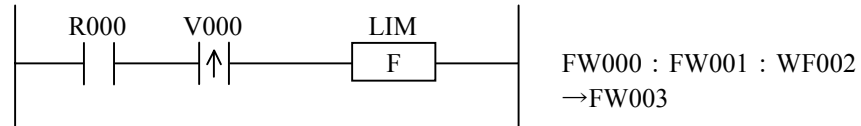
入力条件R000がONのとき、FL000の内容を2ビット右へ回転し、その結果をFL002へ格納します。



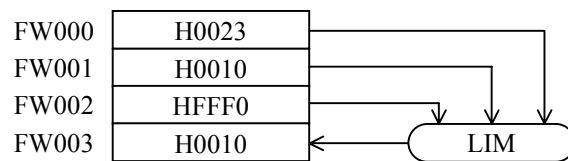
LIM リミッター : LIMITER

機能	ソースの内容を境界値（デスティネーションD1, D2の内容またはイミディエートデータn1, n2）と比較し、リザルトへ格納します。（右図参照）										
パラメータの型	ワード		ロング		フラグ						
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V	
	○	○	○	×	-	↕	-	-	-	-	
パラメータ	ワード	定数なし	$\boxed{\text{LIM}} \boxed{\text{F}}$	SW : D1W : D2W → RW							
		定数あり	$\boxed{\text{LIM}} \boxed{\text{F}}$	SW : n1W : n2W → RW							
	ロング	定数なし	$\boxed{\text{LIM}} \boxed{\text{F}}$	SL : D1L : D2L → RL							
		定数あり									
S : ソース格納アドレス		W : ワード		D1, D2 : デスティネーション格納アドレス		L : ロング					
n1, n2 : イミディエートデータ		R : リザルト（演算結果）を格納するアドレス									
処理内容	ワード	<ul style="list-style-type: none"> • (D1) < (S) のとき (D1) W → (R) W • (D2) ≤ (S) ≤ (D1) のとき (S) W → (R) W • (S) < (D2) のとき (D2) W → (R) W 									
	ワード定数	<ul style="list-style-type: none"> • n1 < (S) のとき n1W → (R) W • n2 ≤ (S) ≤ n1 のとき (S) W → (R) W • (S) < n2 のとき n2W → (R) W 									
	ロング	<ul style="list-style-type: none"> • (D1) L < (S) L のとき (D1) L → (R) L • (D2) L ≤ (S) L ≤ (D1) L のとき (S) L → (R) L • (S) L < (D2) L のとき (D2) L → (R) L 									
入力手順	$\left(\boxed{\text{Shift}} + \boxed{\text{F}} \right) \boxed{\text{L}} \boxed{\text{I}} \boxed{\text{M}} \square \text{パラメータ, パラメータ, パラメータ, パラメータ} \boxed{\text{Enter}}$										
フラグの設定	E : (D1) < (D2) または n1 < n2 のとき 1, それ以外 0 他 : 保持										
注意事項	エラーフラグON時、(D2) または n2 との比較は行いません。（右図参照）										

[ワード]



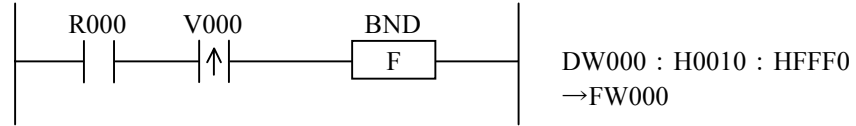
入力条件R000がOFF→ONの変化時、1回のみFW000の内容をFW001およびFW002の内容と比較して、結果をFW003へ格納します。



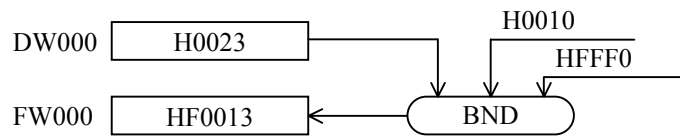
BND デッドバンド : DEAD BAND

機能	ソースの内容を境界値（デスティネーションD1, D2の内容またはイミディエートデータn1, n2）と比較し、境界範囲内をデータ0（不感帯）としてリザルトへ格納します。（右図参照）									
パラメータの型	ワード		ロング		フラグ					
	定数なし	定数あり	定数なし	定数あり	X	E	P	N	Z	V
	○	○	○	×	-	↕	-	-	-	↕
パラメータ	ワード	定数なし	BND F	SW : D1W : D2W → RW						
		定数あり	BND F	SW : n1W : n2W → RW						
	ロング	定数なし	BND F	SL : D1L : D2L → RL						
		定数あり								
S : ソース格納アドレス D1, D2 : デスティネーション格納アドレス n1, n2 : イミディエートデータ R : リザルト（演算結果）を格納するアドレス					W : ワード L : ロング					
処理内容	ワード	<ul style="list-style-type: none"> • (D1) < (S) のとき (S) W- (D1) W → (R) W • (D2) ≤ (S) ≤ (D1) のとき 0 → (R) W • (S) < (D2) のとき (S) W- (D2) W → (R) W 								
	ワード定数	<ul style="list-style-type: none"> • n1 < (S) のとき (S) - n1 W → (R) W • n2 ≤ (S) ≤ n1 のとき 0 → (R) W • (S) < n2 のとき (S) W- n2W → (R) W 								
	ロング	<ul style="list-style-type: none"> • (D1) L < (S) L のとき (S) L- (D1) L → (R) L • (D2) L ≤ (S) L ≤ (D1) L のとき 0 → (R) L • (S) L < (D2) L のとき (S) L- (D2) L → (R) L 								
入力手順	Shift + F B N D ┐ パラメータ, パラメータ, パラメータ, パラメータ Enter									
フラグの設定	E : (D1) < (D2) または n1 < n2 のとき1, それ以外0 V : (ワードのとき) (R) < -32768 または 32767 < (R) のとき1, それ以外0 (ロングワードのとき) (R) < -2147483648 または 2147483647 < (R) のとき1, それ以外0 他 : 保持									
注意事項	<ul style="list-style-type: none"> • (D1) < (D2) または n1 < n2 のとき、エラーフラグ (E) がON（オーバーフローフラグ (V) OFF）し、未処理（リザルトは変化しません）。 • オーバーフロー発生時、リザルトには下記フルスケール値が格納されます。 									
		ワード	ロングワード							
	正のオーバーフロー時	H7FFF	H7FFFFFFF							
負のオーバーフロー時	H8000	H80000000								

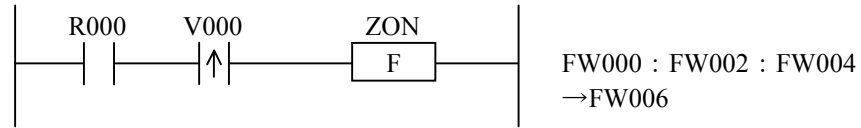
[ワード]



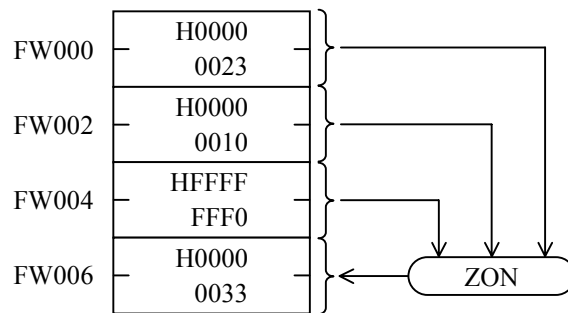
入力条件R000がOFF→ONの変化時、1回のみFW000の内容をイミディエートデータH0010およびHFFF0と比較して、結果をFW000へ格納します。



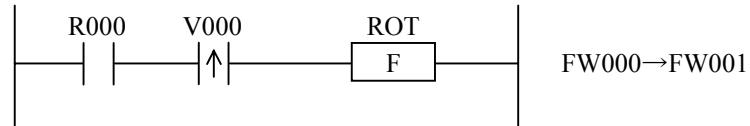
[ワード]



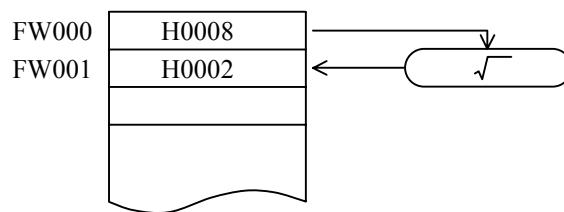
入力条件R000がOFF→ONの変化時、1回のみFW000の内容の正負によりFW002またはFW004の内容を加算し、その結果をFW006へ格納します。



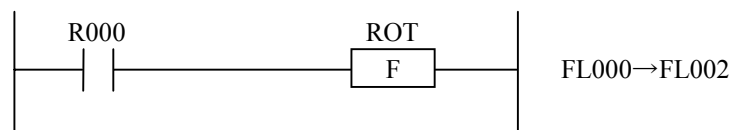
[ワード]



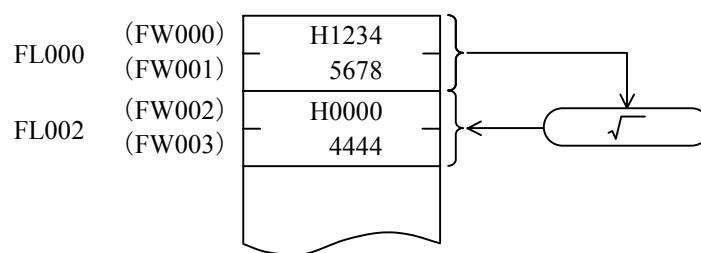
入力条件R000がOFF→ONの変化時、1回のみFW000の内容の平方根を取り、その結果をFW001へ格納します。



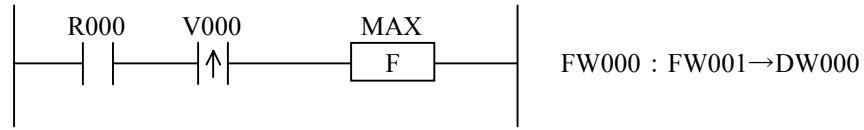
[ロング]



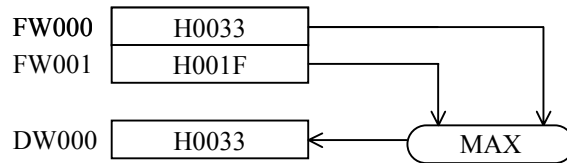
入力条件R000がONのとき、FL000の内容の平方根を取り、その結果をFL002へ格納します。



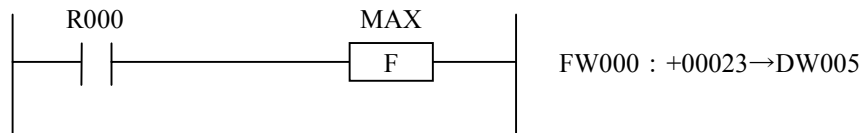
[ワード]



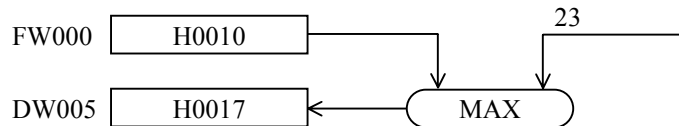
入力条件R000がOFF→ONの変化時、1回のみFW000の内容とFW001の内容の大小比較を行い、大きい値をDW000へ格納します。



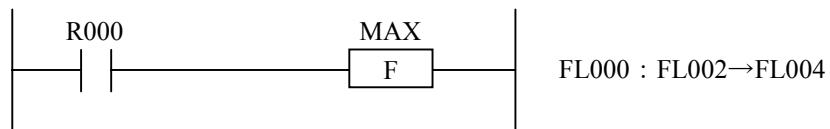
[ワード定数]



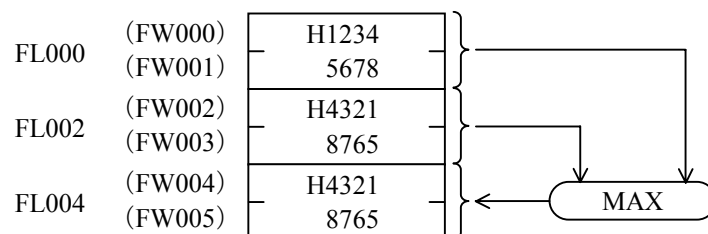
入力条件R000がONのとき、FW000の内容とイミディエートデータ23の大小比較を行い、大きい値をDW005へ格納します。



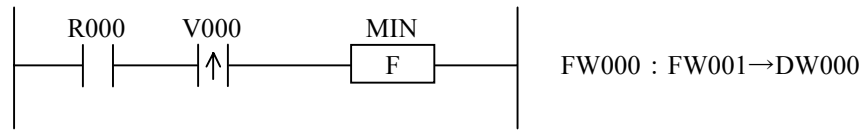
[ワード定数]



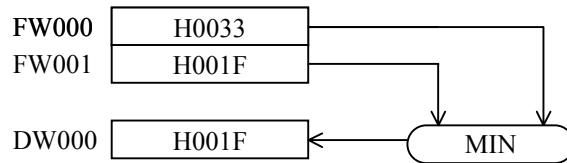
入力条件R000がONのとき、FL000の内容とFL002の内容の大小比較を行い、大きい値をFL004へ格納します。



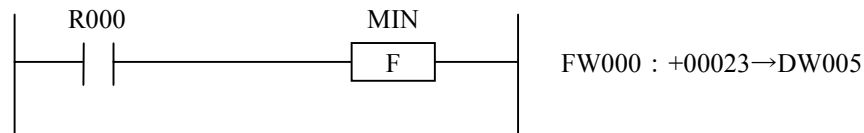
[ワード]



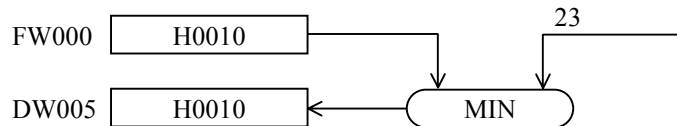
入力条件R000がOFF→ONの変化時、1回のみFW000の内容とFW001の内容の大小比較を行い、小さい値をDW000へ格納します。



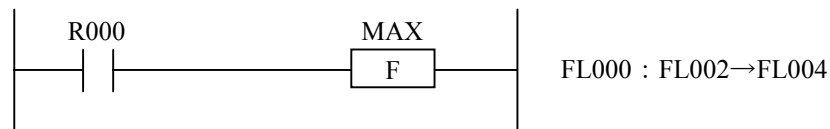
[ワード定数]



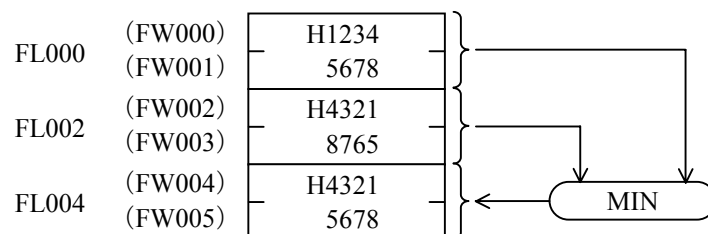
入力条件R000がONのとき、FW000の内容とイミディエートデータ23の大小比較を行い、小さい値をDW005へ格納します。



[ワード定数]



入力条件R000がONのとき、FL000の内容とFL002の内容の大小比較を行い、小さい値をFL004へ格納します。



CLR クリア : CLEAR

機能	指定のI/Oエリアをクリアします。また、TCLR, UCLR, CCLRについては、それぞれ計数値エリアもクリアします。							
パラメータの型			フラグ					
			X	E	P	N	Z	V
シンボル	XCLR		-	-	-	-	-	-
	YCLR		-	-	-	-	-	-
	GCLR		-	-	-	-	-	-
	RCLR		-	-	-	-	-	-
	KCLR		-	-	-	-	-	-
	TCLR		-	-	-	-	-	-
	UCLR		-	-	-	-	-	-
	CCLR		-	-	-	-	-	-
	VCLR		-	-	-	-	-	-
	ECLR		-	-	-	-	-	-
	FCLR		0	0	0	0	0	0
シンボル 入力手順	<p> + </p> <p> </p> <p> クリアしたいエリアのコード (X, Y, G, R, …) を入力します。 </p> <p> パラメータはありません。 </p>							

クリア エリア	Xエリア	← XCLRがクリアするエリア		
	Yエリア	← YCLRがクリアするエリア		
	Gエリア	← GCLRがクリアするエリア		
	Rエリア	← RCLRがクリアするエリア		
	Kエリア	← KCLRがクリアするエリア		
	Tエリア	← TCLRが クリアするエリア	→ T計数値エリア	
	Uエリア	← UCLRが クリアするエリア	→ U計数値エリア	
	Cエリア	← CCLRが クリアするエリア	→ C計数値エリア	
	Vエリア	← VCLRがクリアするエリア		
	Eエリア	← ECLRがクリアするエリア		
	SW000	ファンクションフラグエリア (X, E, P, N, Z, Vなど)	← FCLRがクリア するエリア	

3 処理時間

3 処理時間

3.1 2αの処理時間

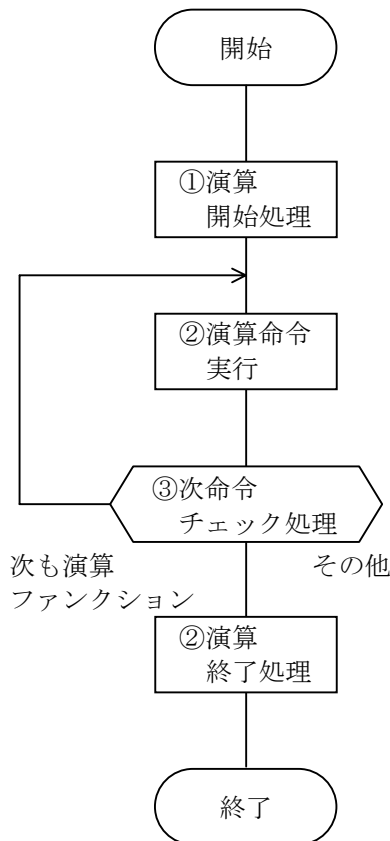
ラダー回路プログラムの処理時間はラダープログラムの実行時間と、演算命令、プロセスレジスタ、ネスティングコイルなどのOS処理時間の合計となります。

(1) ラダー回路命令の処理時間

シンボル	処理時間 μs/命令		
	2α	2αE	2αH(f)
┆┆, ┆┆, , ┆, ┆	0.33		0.075
┆┆, ┆┆, ○, ⊙, ⊖	0.66		0.15
END命令 (プログラミング時自動セット)	300	180	

(2) 演算ファンクションの処理時間

図3-1にOSによる演算処理の概要を示します。



②の処理時間を「2.4 演算ファンクション一覧表」に示します。

①③④のOS処理時間は下記のようにになります。

No.	処理時間 [ms]	
	2α	2αE, 2αH(f)
①	0.12	0.072
③	次が演算命令の場合 =0.04	=0.024
	その他 =0.01	=0.006
④	0.12	0.072

〔概略計算〕

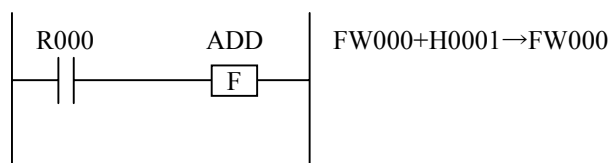
- ・演算ファンクション1命令単独の場合
0.3ms/命令 (2α)
0.18ms/命令 (2αE, 2αH(f))
- ・演算ファンクション1命令以上連続の場合
0.15ms/命令 (2α)
0.09ms/命令 (2αE, 2αH(f))

上記時間に演算ファンクションの使用個数をかけた時間が、演算ファンクションの概略処理時間になります。

図3-1 OSによる演算処理の概要

2αを例に次の回路の計算方法を示します。

(a)



処理時間=①+②+③+④

$$\textcircled{1} = 0.12$$

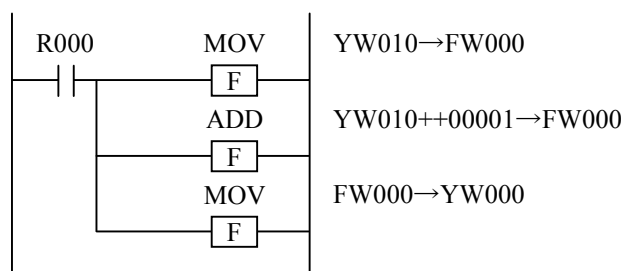
$$\textcircled{2} = 0.06$$

$$\textcircled{3} = 0.01$$

$$\textcircled{4} = 0.12$$

$$\text{計} \quad 0.31 \text{ [ms]}$$

(b)



処理時間=①+3×③+④

+2×②MOV+②ADD

$$\textcircled{1} = 0.12$$

$$3 \times \textcircled{3} = 2 \times 0.04 + 1 \times 0.01 = 0.09$$

$$\textcircled{4} = 0.12$$

$$2 \times \textcircled{2} \text{MOV} = 2 \times 0.04 = 0.08$$

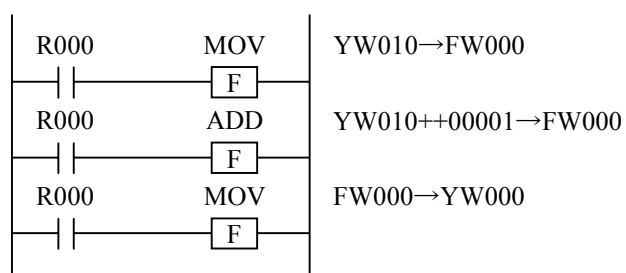
$$\textcircled{2} \text{ADD\#} = 0.06$$

$$\text{計} \quad = 0.47$$

[ms]

上記(b)の回路は次のようにも作成できます。しかし、この場合(b)よりも処理時間が長くなります。

(c)



処理時間= [①+②MOV×③+④]

+ [①+②ADD#+③+④]

+ [①+②MOV+③+④]

= 3×①+3×③+3×④

+ 2×②MOV+②ADD#

$$3 \times \textcircled{1} = 3 \times 0.12 = 0.36$$

$$3 \times \textcircled{3} = 3 \times 0.01 = 0.03$$

$$3 \times \textcircled{4} = 3 \times 0.12 = 0.36$$

$$2 \times \textcircled{2} \text{MOV} = 2 \times 0.04 = 0.08$$

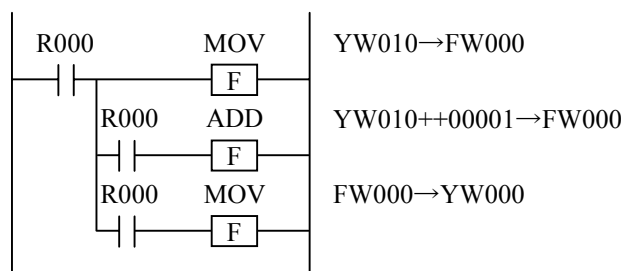
$$\textcircled{2} \text{ADD\#} = 0.06$$

$$\text{計} \quad = 0.89$$

[ms]

次の回路の場合は(c)と同じ計算となります。

演算ファンクションの次に1つでも接点などが入れば③はその他で計算してください。



- 演算ファンクションがONしていない場合は、ラダー回路命令の $\text{---}\bigcirc\text{---}$ と同じ $0.66\mu\text{s}/\text{命令とパラメータの実行時間}$ ($0.33\mu\text{s}/1\text{パラメータ}$) の合計となります。

3 処理時間

(3) 特殊コイルの処理時間

プロセスレジスタ (P) とネスティングコイル (N) は、演算ファンクションと同様にONしている場合、OSで処理されます。以下にそれぞれの処理時間を示します。

種別	処理時間 [ms]	
	2α	$2\alpha E, 2\alpha H(f)$
	0.5	0.3
	0.5	0.3

- CPU待ち行列が多くある場合、上記値より多少大きくなる場合があります。
- 各コイルがONしていない場合は $0.66\mu\text{s}$ /命令となります。

(4) アップダウンカウンタの処理時間

アップダウンカウンタは演算命令と同一処理となります。この場合は、演算命令の計算式の②に次の値をセットして計算してください。

$$\begin{aligned}\text{カウンタ処理時間} &\approx 70\mu\text{s}/\text{命令} (2\alpha) \\ &\approx 42\mu\text{s}/\text{命令} (2\alpha E, 2\alpha H(f))\end{aligned}$$

(5) タイマ (T) ワンショット (U) の処理時間

タイマおよびワンショットはプログラムの有無に関係なく、100msごとにエディション点数分だけ処理されます。

また、この100msのタイミングはOS内部の10msごとの割り込みによって図に示すように作成されます。

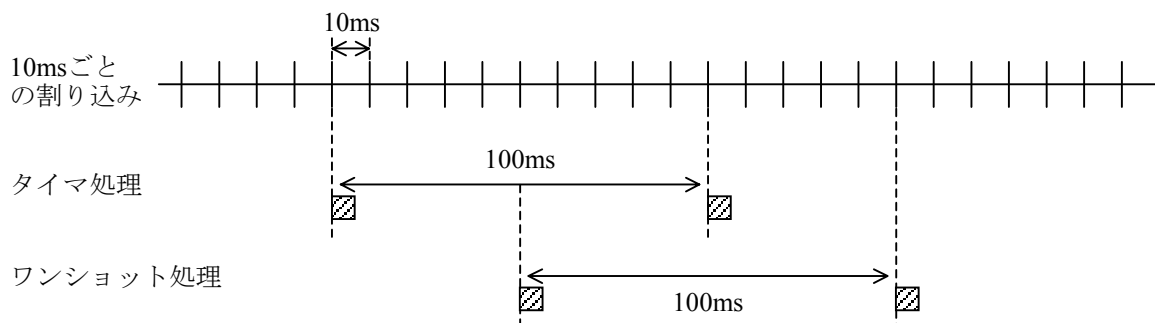


図 3-2 OS処理タイミング

タイマおよびワンショットは16点単位に処理されますが、それぞれの16点単位の処理時間は次のようになります。

タイマ16点の処理時間 … 250 μ s/16点

ワンショット16点の処理時間 … 300 μ s/16点

また、10msごとに発生するOSの基本処理時間は以下のとおりです。

10msごとの処理時間 … 約250 μ s

したがって、100msごとに発生するそれぞれの処理時間は以下のようになります。

$$\text{タイマ} = 250 + 250 \times \left(\frac{\text{タイマ点数}}{16} \right)$$

$$\text{ワンショット} = 250 + 300 \times \left(\frac{\text{ワンショット点数}}{16} \right)$$

- 10msタイマを使用した場合は、10msごとに次の処理時間が必要になります。

10msタイマ処理=250+250=500 μ s/16点

3 処理時間

(6) リモートI/Oサポート時間

リモートI/Oの1サイクル分の転送が終了すると、OSはリモートI/Oのエラー情報を退避した後に再起動します。このときのリモートI/Oの処理時間とOS処理の関係を図3-3に示します。

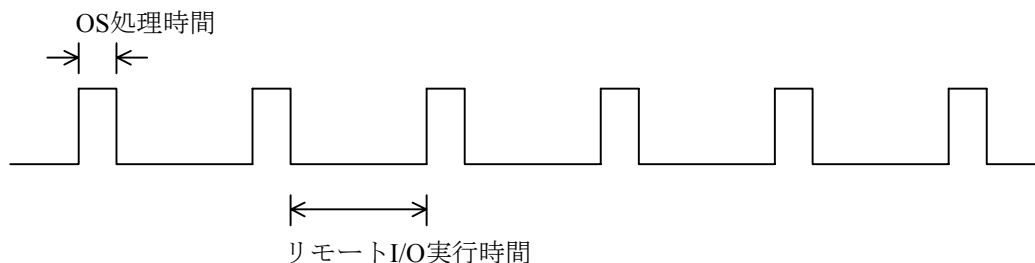


図3-3 リモートI/OとOS処理

リモートI/Oはエディション設定により点数を変更できますが、このときの点数とOSの処理時間は次のようになります。

リモートI/O点数	OS処理時間 (ms)	リモートI/O実行時間 (ms)
512	$0.2+0.2=0.4$	5.0
1024	$0.2+0.4=0.6$	10.0
1536	$0.2+0.6=0.8$	15.0
2048	$0.2+0.8=1.0$	20.0

- アナログ、パルスカウンタなどを使用する場合、各サポートOSの処理時間を上記データに加算してください。
なお、サポートOSの処理時間については、アナログ、パルスカウンタなどのマニュアルを参照してください。

3.2 4aの処理時間

3.2.1 スキャンタイム

PCsは、シーケンスRUN時、I/O転送（I/Oメモリと入出力モジュール間のデータ転送）やシーケンスプログラムの実行などを、高速に繰り返し実行しています。

図3-4に示すように、1回分の実行時間、1スキャンタイム（T）は、下式により求められます。

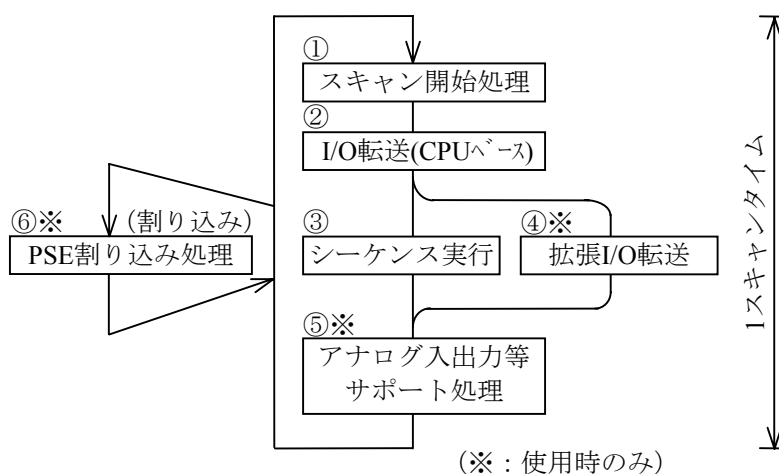


図3-4 PCs処理フロー

$$T = ① + ② + \max(③, ④) + ⑤ + ⑥$$

→ ③または④の大きい方の値を使用します。

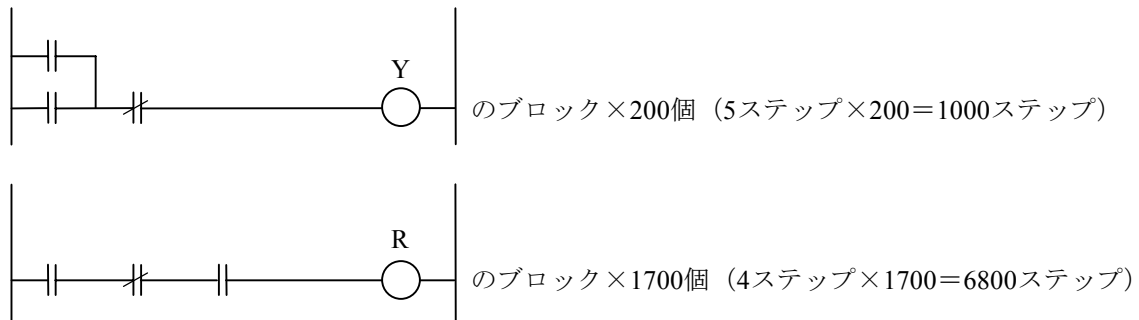
①～⑥のそれぞれの処理時間を表3-1に示します。

3 処理時間

3. 2. 2 スキャンタイム計算例

図3-5にスキャンタイムの計算例を示します。

(ただし、CPUベースは基本8スロット、アナログ入出力モジュールなどの特殊なモジュール、
10msタイム未使用、PSモニタなしとします。)



$$\begin{aligned} \text{スキャンタイム (T)} &= 2.0\text{ms} \text{ (①)} + 2.1\text{ms} \text{ (②)} \\ &\quad + \{2\mu\text{s} \times 5 \times 200 + 2\mu\text{s} \times 4 \times 1700\} \text{ (③)} \\ &= 19.7\text{ms} \end{aligned}$$

図3-5 スキャンタイムの計算例

表 3-1 処理時間一覧

No.	項目		処理時間 (最大)	備考	
①	スキャン開始距離		2.0ms		
②	I/O転送 (CPUベース)	4スロット	1.1ms	CPUベース内のDI/OとのI/O転送。	
		8スロット	2.1ms		
③	シ ー ケ ン ス 実 行	↑↑, ↑↑	2μs	それぞれの命令語の使用数により、各 処理時間が加算されます。	
		↑↑, ↑↑	2μs		
		⊖, ⊖	2μs		
		○	T		(ON時) 210μs (OFF時) 80μs
			U		(ON時) 300μs (OFF時) 180μs
			CU CD CR		(OFF→ON時) 240μs (他) 4μs
		他	2μs		
		⊖F⊖	(ON時)		
(OFF時)	8μs				
	↓, ↓, ↓	2μs			
④	拡張 I/O転送	CPUベース 4スロット時	11.6ms	拡張スロット付きCPUベースにて、拡 張I/Oモジュール (LWE800または LWE805) 実装の場合のみ。	
		CPUベース 8スロット時	8.0ms		
⑤	アナログ入出力等 サポート処理		1.2ms	アナログ入出力、プリセットカウンタ モジュール使用時のみ。	
⑥	PSE割り込み処理		PSEのリモート接続開始時、または読み出し処理、各種モニタ処理に より数ms~数100msスキャンタイムが伸びることがあります。		
⑦	その他		T=T×1.1 約5ms	(10msタイマ使用時のみ) (上位計算機接続時のみ)	
	・ 10msタイマ ・ 上位計算機				

3.3 平均スキャンタイムの表示

プログラム平均スキャンタイムは図3-6に示すプログラムを追加することにより、計算することができます。この値はユーザプログラムRUN時、8秒ごとの平均スキャンタイム (ms) を示し、FWBFFに格納します。

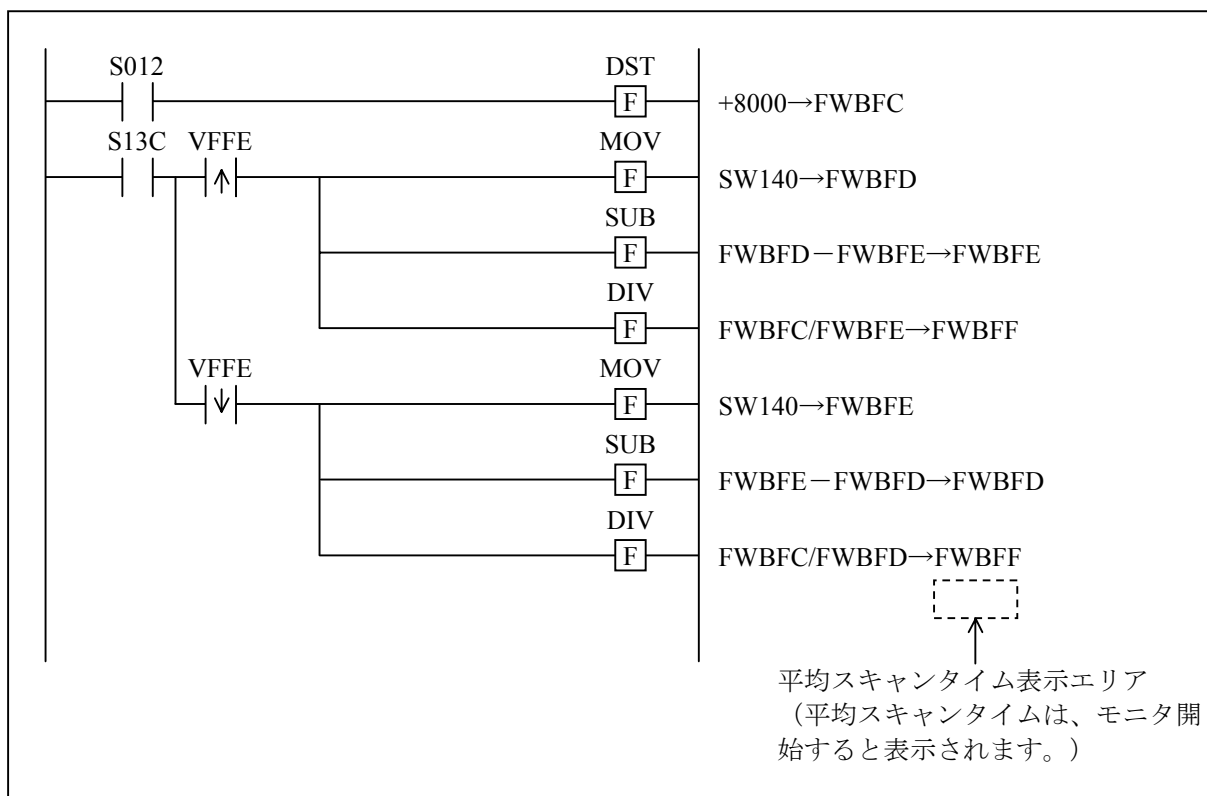


図3-6 スキャンタイム表示プログラム

索 引

〈英数字〉		命令	
命令		LSL	130
ABS	120	LSR	128
ADD	42	LT	74
AND	58	MAX	148
APB	112	MDI	86
ASL	134	MIN	150
ASP	108	MOD	54
ASR	132	MOM	82
AST	98	MOV	80
ASU	110	MSI	84
AUB	114	MUL	50
BND	142	NEG	122
BTD	102	NEQ	68
CCLR	152	NOT	64
DCD	124	OR	60
DEC	48	POP	94
DIV	52	PSH	92
DST	96	RCLR	152
DTB	104	ROL	136
DTS	118	ROR	138
ECD	126	ROT	146
ECLR	152	SCH	100
EOR	62	SCL	56
EQU	66	SEG	106
EXC	90	STD	116
FCLR	152	SUB	44
GCLR	152	TCLR	152
GE	72	TST	78
GT	70	UCLR	152
INC	46	VCLR	152
INI	88	XCLR	152
KCLR	152	YCLR	152
LE	76	ZON	144
LIM	140		

ご利用者各位

〒101-8010

東京都千代田区神田駿河台4丁目6番地
株式会社日立製作所

お 願 い

各位にはますますご清栄のことと存じます。

さて、この資料をより良くするために、お気付きの点はどんなことでも結構ですので、
下欄にご記入の上、弊社営業担当または弊社所員に、お渡しくださいますようお願い申
しあげます。なお、製品開発、サービス、その他についてもご意見を併記して頂ければ
幸甚に存じます。

ご住所 〒	_____
貴会社名 (団体名)	_____
芳 名	_____
製品名	
ご意見欄	_____ _____