KEYENCE

084823

1章 概要
2章 仕様
3章 ユニットの設置および保守
4章 ラダープログラミング
5章 CPU内蔵機能
6章 拡張I/Oユニット機能
・PLC本体エラー一覧
・CR/CM一覧
・索引

プログラマブルコントローラ

KV-8000シリーズ

ユーザーズマニュアル

お使いになる前に、このマニュアルをお読みください。
いつでも使用できるように大切に保管してください。

・CPUユニット
・電源ユニット
・拡張I/Oユニット
・バス接続ユニット

対応CPUユニット
・KV-8000
・KV-8000A

本社技術スタッフ直営！
FAQ よくあるご質問
www.keyence.co.jp/seigyofaq

はじめに

本書は、《KV-8000シリーズ》の接続･保守方法、CPU内蔵機能、ラダープログラムを作成する方法を説明しています。また、CPUユニット･拡張I/Oユニットの仕様や使用方法を掲載しています。上記以外の拡張特殊ユニットについては、各ユニットのユーザーズマニュアルをご覧ください。
実際にプログラムされる際には、『命令語リファレンスマニュアル』･『ST言語プログラミングマニュアル』･『スクリプトプログラミングマニュアル』･『KV STUDIOユーザーズマニュアル』を併せてお読みください。
本書はいつでもご利用いただけますよう、大切に保管してください。
本書が最終的にご利用いただく方のお手元に届くよう、ご配慮ください。

■ KV-8000シリーズ関連マニュアル

以下の全てのPDFマニュアルは、KV STUDIOのヘルプから開けます。また、最新のPDFマニュアルは弊社のホームページよりダウンロード可能です。
製本版マニュアルのご請求は、本書裏表紙のお問い合わせ電話番号にてご用命ください。

名称	内容	製本
KV-8000シリーズCPUユニット ユーザーズマニュアル	本書です。《KV-8000シリーズ》のシステム構成･仕様、ラダープログラムを作成する方法を説明しています。	なし（PDFのみ）
KV-8000/7000/5000/3000/1000シリーズ KV Nanoシリーズ 命令語リファレンスマニュアル	ラダープログラミングで使用可能な各命令語を説明しています。	なし（PDFのみ）
KV-8000シリーズ ST言語プログラミングマニュアル	ST言語プログラムを作成する方法と使用可能な演算子･制御文･関数を説明しています。	なし（PDFのみ）
KV-8000/7000/5000/3000/1000シリーズ KV Nanoシリーズ スクリプトプログラミングマニュアル	スクリプトプログラムを作成する方法と使用可能な演算子･制御文･関数を説明しています。	なし（PDFのみ）
KV STUDIO ユーザーズマニュアル	《KV STUDIO》の操作方法を説明しています。	なし（PDFのみ）

KV-8000(A)　CPU機能バージョン

KV-8000(A)は、出荷時期によりCPU機能バージョンが異なります。CPU機能バージョンは、KV STUDIO、アクセスウィンドウ、コントロールメモリCMから確認できます。
また、システムプログラムをKV STUDIOやメモリカードから更新することで、CPU機能バージョンをバージョンアップできます。

「システムプログラムアップデートについて」（3-24ページ）
「CPUモニタ」（5-136ページ）
「コントロールメモリ CM」（付-14ページ）

■ ソフトウェアのライセンスと著作権

本製品はオープンソースの対象ソフトウェアコンポーネントも含まれております。
オープンソースの対象ソフトウェアコンポーネントに関しては、KV STUDIO のインストールフォルダ直下にある"license"フォルダ以下の"license_KV-8000.txt"をご参照ください。

KV-8000(A)　CPU機能バージョンについて

KV-8000 シリーズは、CPU 機能バージョンにより使用できる機能が異なります。
CPU 機能バージョンは、KV STUDIO のメニューで【ツール】→【CPU 機能バージョン確認】→【PLC】から確認できます（※KV STUDIO Ver.11.1 以降が必要です）。
また、CPU 機能バージョンは、KV STUDIO のメニューで【ツール】→【CPU 機能バージョン確認】→【PLC】→〔システムプログラム更新〕から更新できます。（※ KV STUDIO Ver.11.1 以降が必要です。）

■ CPU機能バージョンによる機能差分

公開		2023/10	2022/09	2022/06	2021/11
CPU機能バージョン		2.7	2.6	2.5	2.4
KV STUDIOの対応バージョン		Ver.11.63以降	Ver.11.6以降	Ver.11.5以降	Ver.11.4以降
CPU機能	OPC UAサーバ機能	○	○	○	○
	ユーザ認証設定	○	―	―	―
	初期化機能	○	―	―	―
	プロジェクトロードの禁止	○	―	―	―
対応拡張ユニット	KV-XH64EC/XH32EC	○	○	―	―

公開		2021/09	2021/07	2021/04	2020/04	2019/04
CPU機能バージョン		2.3	2.2	2.1	2.0	1.0
KV STUDIOの対応バージョン		Ver.11.3以降	Ver.11.2以降	Ver.11.1以降	Ver.11.0以降	Ver.10.0以降
拡張ユニット	KV-XD02	○	○	―	―	―
CPU機能	デバイスコメント	最大128文字	最大128文字	最大128文字	最大32文字	最大32文字
	変数	○	○	○	○	―
	配列	ファンクションブロック型に対応	STRING型に対応	―	―	―
	構造体	○	○	○	○	―
	ファンクションブロック	データ型の自動変換に対応	○	○	○（EN/ENOに対応、インスタンスに対応）	○
	ST言語	○	○	○	―	―
命令語・スクリプト	ファンクションブロック命令	○	○	○	2命令追加（FBCALL, FBSTRT）	追加なし
	データ転送命令	○	○	○	1命令追加(BBMOV)	―
	データ処理命令	○	○	○	1命令追加(BBCMP)	―
	フロー命令	○	○	1命令追加(RETURN)	―	―
	浮動小数命令	○	○	2命令追加（ROUND, DROUND）	―	―
	サイズ取得命令	○	○	○	3命令追加(ACNT,WSIZE, BSIZE)	―
	サフィックス	.Aに対応	―	―	―	―

安全にご使用いただくために

■ 記号の見かた

本書では、人への危害や機器への損害を未然に防ぐため、守っていただきたい事項を下記のように表示区分しています。

危 険	ここに記載されている記載事項を遵守しない場合、結果的に死亡又は重傷を引き起こす危害が発生します。
警 告	ここに記載されている記載事項を遵守しない場合、結果的に死亡又は重傷を引き起こす危害が発生する可能性があります。
注 意	ここに記載されている記載事項を遵守しない場合、中程度の傷害又は軽傷を引き起こす危害が発生する可能性があります。
注 記	ここに記載されている記載事項を遵守しない場合、商品自体の損害(自損)のみならず、他の財物に対する損傷を引き起こす可能性があります。

▶ 重 要　かならずおこなう操作などについての注意を示しています。

ポイント　誤りやすい操作などについての注意を示しています。

参 考　本文の理解を深める事項や、知っておくと役に立つ情報を示しています。

参照すべき頁や別のマニュアルの参照ページを示しています。

■ 一般的な注意事項

危 険	• 人体および、人体の一部を保護する目的でこの商品を使用してはいけません。 • この商品は、防爆エリアで使用することを想定していないので、防爆エリアでは決して使用しないでください。 • この商品の動作状態によって人および財物に大きな危害が及ぶ可能性のある用途(発電所、航空、鉄道、船舶、車両、医療機器、娯楽遊具等)では使用しないでください。
警 告	• プログラマブルコントローラ本体に異常が発生した場合でも、フェールセーフの点から、システム全体が安全側に働くように、プログラマブルコント ローラを介さない方法で安全回路を設けてください。 • 出力回路や内部回路の故障により、正常に制御動作できなくなる場合が考えられます。火災など、重大な事故の要因となる制御については、安全回路を必ず設けてください。 • この商品を設置する際は、装置全体のリスクアセスメントを実施し、適切なリスク低減がなされていることを確認してください。異常発生時でも装置全体が安全に動作するよう、この商品を介さずに適切な保護手段を設けてください。
注 意	• 始業または操作時には、当商品の機能および性能が正常に動作していることを確認してからご使用ください。 • この取扱説明書に規定した方法以外で当社商品を使用した場合、商品が備えている保護が損なわれる可能性があります。
注 記	• 仕様に示された規格以外でのご使用、または改造された商品については、機能および性能の保証はできかねますのでご留意ください。 • 当社商品を他の機器と組み合わせてご使用になる場合、使用条件、環境などにより、機能および性能が満足できない場合がありますので、十分ご検討のうえご使用ください。

規格および規制に関する注意事項

■ CE およびUKCA マーキングについて

当社は、以下の要件に基づき、この商品がEU 指令および英国法規制の必須要求事項を満たしていることを確認しています。EU 諸国および英国にてこの商品を使用される場合には、以下の要件に留意して使用してください。

● EMC 指令（CE）およびElectromagnetic Compatibility Regulations（UKCA）

- 適用規格　(BS) EN61131-2, Class A
- KV-8000シリーズを設置される場合には、必ずIP54以上の能力をもつ導電性のエンクロージャ（制御盤など）内に設置してください。
- エンクロージャ（制御盤など）はFGに接地してください（プラス接地はできません）。
- エンクロージャ（制御盤など）外に出る信号線は必ずシールドケーブルを使用し、シールドケーブルはエンクロージャ（制御盤など）に接地してください。ただし、CPU ユニット、KV-XL202/XL402に接続されるシリアル通信ケーブル、USB ケーブルのシールドケーブルは接地しないでください。
- FG端子が設けられているユニットは、FG 端子を必ず接地してください。
- KV-XLE02、KV-XCM02にはカテゴリ5e以上の二重シールド付STPケーブルを使用してください。
- CPUユニットに接続されるUSBケーブルにはフェライトコア※1を2ターン巻きつけてください。
- パルス列位置決めユニット（KV-SH04PL）のDC24V/0V入力、パルス出力端子に接続されるケーブルにはそれぞれフェライトコア※2を1ターン巻きつけてください。
- 高速カウンタユニット（KV-SSC02）のシリアルエンコーダ通信端子に接続されるケーブルにはフェライトコア※2を1ターン巻きつけてください。

  ※1 当社評価時使用フェライトコア：TDK ZCAT-3035-1330

  ※2 弊社評価時使用フェライトコア：NEC/TOKIN ESD-SR-150

ただし、これらの要件は、この商品を組み込んだ機械装置全体がEMC指令およびElectromagnetic Compatibility Regulationsの必須要求事項を満たすことを保障するものではありません。機械装置全体に対する適合性の確認は、機械装置製造者の責務となります。

適合のためのポイント

＜制御盤について＞

- 天板や底板など制御盤内部に繋ぎ目が存在する場合には、塗装を剥ぐなどして導電性を保ってください。
- 制御盤自体の接地には、高周波でも低インピーダンスになるよう太い接地線を使用してください。
- メンテナンス時などに制御盤内の機器に触れる場合は、静電気の影響が無いように必ず除電してください。

＜シールドケーブルの接地について＞

- シールドケーブルの接地処理は拡張ユニットの近くでおこない、接地後のケーブルは接地前のケーブルから電磁誘導を受けないようにしてください。
- シールドケーブルの外被を一部取り除いて露出させたシールド部は、制御盤に対して広い面で接地してください。（クランプ金具などを使用する際は、接触部の塗装を剥がしてください。）
- シールド線に電線をハンダ付けして、その電線を接地処理する方法は高周波インピーダンスが高くなり、シールド効果がなくなりますのでおこなわないでください。

● 低電圧指令

低電圧指令の対象となる機種は、電源ユニットKV-PU1です。その他の機種は、低電圧指令対象外です。

- 適用規格　EN60950-1
- 過電圧カテゴリⅡ
- 汚染度2の環境下でご使用ください。
- 高度2000m以下に設置してください。
- 屋内に設置してください。
- KV-PU1を設置される場合には、必ずIP54以上の能力を持つ導電性のエンクロージャ（制御盤など）内に設置してください。

■ UL認証について

この商品は、以下のUL規格、及びCSA規格に適合し、UL認証を取得しています。

UL File No.	E207185
カテゴリ	NRAQ, NRAQ7

UL認定品としてこの商品を使用される場合には、以下の要件に留意して使用してください。

- IP54以上の能力を持つエンクロージャ（制御盤）内に必ず設置してください。
- KV-PU1の端子台の配線は、AWG#12～#22の銅線（より線）で、温度定格が105℃以上の電線を使用してください。締付けトルクは、1.4N•mです。
- CPUユニットのDC電源入力端子台の配線は、AWG#16～#22の銅線（より線）で、温度定格が105℃以上の電線を使用してください。締付けトルクは、0.5N•mです。
- KV-PU1を使用する場合は、CPUユニットのDC電源端子台から24Vを給電しないでください。
- 汚染度2の環境下で使用してください。
- 過電圧カテゴリⅠ（KV-PU1 使用時 Ⅱ）
- 高度2000m以下に設置してください。
- 屋内に設置してください。
- この商品の入出力端子（コネクタ等）に接続する回路がSELVに対応していることを確認してください。
- この商品を使用する際は、下記いずれかの電源を使用してください。
  - NFPA70（NEC: National Electrical Code）、及びCEC（Canadian Electrical Code）に規定されるClass 2出力を有するUL認証電源、あるいはCSA認証電源。
  - UL60950-1、及びCAN/CSA-C22.2 No.60950-1 に規定されるLimited Power Sourceとして評価されたUL認証電源、あるいはCSA認証電源。

■ 韓国電波法について

Class A 機器

この機器は、業務電磁波発生機器（Class A）であり、家庭以外の場所での使用を意図していますのでこの点にご注意ください。

A급 기기 (업무용 방송통신기자재)

이 기기는 업무용 (A 급) 전자파적합기기로서 판매자 또는 사용자는 이 점을
주의하시기 바라며 , 가정외의 지역에서 사용하는 것을 목적으로 합니다 .

規格および規制に関する注意事項（KV-5500/5000/3000シリーズの場合）

■ CEマーキングについて

当社は、以下の要件に基づき、この商品がEC指令の必須要求事項を満たしていることを確認しております。EU諸国にてこの商品を使用される場合には、以下の要件に留意頂きご使用ください。

● EMC指令

- 適用規格　EN61131-2, Class A
- KV-5500/5000/3000シリーズを設置される場合には、必ずIP54 以上の能力をもつ導電性のエンクロージャ（制御盤）内に設置してください。
- エンクロージャ（制御盤）外に出る信号線は必ずシールドケーブルを使用し、シールドケーブルは接地してください。
- FG端子が設けられているユニットは、FG端子を必ず接地してください。

これらの要件は、この商品を組み込んだ機械装置全体がEMC指令の必須要求事項を満たすことを保障するものではありません。機械装置全体に対する適合性の確認は、機械装置製造者の責務となります。

適合のためのポイント

＜制御盤について＞

- 天板や底板など制御盤内部に繋ぎ目が存在する場合には、塗装を剥ぐなどして導電性を保ってください。
- 制御盤自体の接地には、高周波でも低インピーダンスになるよう太い接地線を使用してください。

＜シールドケーブルの接地について＞

- シールドケーブルの接地処理は拡張ユニットの近くでおこない、接地後のケーブルは接地前のケーブルから電磁誘導を受けないようにしてください。
- シールドケーブルの外被を一部取り除いて露出させたシールド部は、制御盤に対して広い面で接地してください。（クランプ金具などを使用する際は、接触部の塗装を剥がしてください。）
- シールド線に電線をハンダ付けして、その電線を接地処理する方法は高周波インピーダンスが高くなり、シールド効果がなくなりますのでおこなわないでください。

● 低電圧指令

低電圧指令の対象となる機種は、KV-B8RCおよびKV-B16RCのみです。その他の機種は、低電圧指令対象外です。

- 適合規格　EN61131-2 (KV-B8RC/KV-B16RC)
- 過電圧カテゴリⅡ
- 汚染度2の環境下でご使用ください。
- KV-5500/5000/3000 シリーズを設置される場合には、必ず IP54 以上の能力を持つエンクロージャ（制御盤）内に設置してください。
- KV-B16RC のリレー負荷定格は、1コモンあたり8Aです。つまり、8チャンネル全てを使用した場合は各チャンネル1Aまで、4チャンネル使用した場合には、各チャンネルあたり2Aまでの負荷となります。
- KV-B8RCは独立接点であり、1チャンネルあたりのリレー負荷定格は2Aです。

■ UL認証について

KV-5500/5000/3000シリーズは、以下のUL/C-UL認証を取得しています。

UL File No.：　E207185
カテゴリ：　NRAQ, NRAQ7

UL認定品としてこの商品を使用される場合には、以下の要件に留意頂きご使用ください。

- KV-5500/5000/3000シリーズを設置される場合には、IP54以上の能力を持つエンクロージャ（制御盤）内に必ず設置してください。
- 特殊ユニット（KV-DN20を除く）、及びI/O ユニットの端子台への配線は、AWG#16-#22の銅線（より線）で、温度定格が60 ℃以上（KV-B16RCは105℃以上）の電線を使用してください。締付けトルクは、0.56N•mです。
- この商品を使用する際は、下記いずれかの電源を使用してください。
  - NFPA70（NEC: National Electrical Code）、及びCEC（Canadian Electrical Code）に規定されるClass 2出力を有するUL認証電源、あるいはCSA認証電源。
  - UL60950-1、及びCAN/CSA-C22.2 No.60950-1 に規定されるLimited Power Sourceとして評価されたUL認証電源、あるいはCSA認証電源。

■ 韓国電波法について

Class A 機器
この機器は、業務電磁波発生機器（Class A）であり、家庭以外の場所での使用を意図していますのでこの点にご注意ください。

A급 기기 (업무용 방송통신기자재)

이 기기는 업무용 (A 급) 전자파적합기기로서 판매자 또는 사용자는 이 점을
주의하시기 바라며 , 가정외의 지역에서 사용하는 것을 목적으로 합니다 .

マニュアルの構成

1章	概要	KV-8000シリーズのシステム構成･パソコンとの通信経路･ユニットの占有領域などを説明します。
2章	仕様	KV-8000シリーズのCPUユニット･AC電源ユニット･拡張I/Oユニット･バス接続ユニットの仕様および配線を説明します。
3章	ユニットの設置 および保守	KV-8000シリーズの設置方法、取り付け、配線時の注意事項や、保守時の手順を説明します。
4章	ラダープログラミング	ラダープログラムを作成する上で必要となる機能の仕様や設定方法を説明します。モジュール、ファンクションブロック、マクロ、ローカルデバイスを使用すればプログラミングを効率化できます。
5章	CPU内蔵機能	CPUユニット《KV-8000シリーズ》とアクセスウィンドウ、各通信ポート、ユーザメモリなどを使用して実現できる機能を説明します。
6章	拡張I/Oユニット機能	拡張I/Oユニット≪KV-SIR32XT≫を使用して実現できる機能を説明します。
	付　録	KV-8000シリーズのコントロールリレー(CR)･コントロールメモリ(CM)一覧やエラー一覧、本書で取り扱う命令語一覧を掲載しています。

目　次

はじめに
安全にご使用いただくために.....2
マニュアルの構成.....7
目　次.....8
マニュアルの読みかた.....17
ニモニックリストの使い方.....17
用語について.....18

1章　概要

1-1　システム構成.....1-2
システム構成の概要.....1-2
①電源について.....1-3
②CPUユニットについて.....1-3
③KV-8000/7000シリーズ用拡張ユニットについて.....1-3
④バス接続ユニットについて.....1-4
⑤KV-5000/3000シリーズ用拡張ユニットについて.....1-4
⑥エンドユニットについて.....1-4
通信機器との接続方法.....1-5
1-2　ユニットの予約について.....1-6

2章　仕様

2-1　シリーズ共通仕様.....2-2
一般仕様.....2-2
性能仕様.....2-3
電源断時の動作.....2-4
2-2　CPU ユニット.....2-5
パッケージ内容.....2-5
各部の名称と機能.....2-6
RUN-PRG切換スイッチ.....2-8
結線･内部回路図.....2-8
外形寸法図.....2-10
2-3　AC 電源ユニット.....2-11
各部の名称と機能.....2-11
仕様.....2-12
配線方法.....2-12
外形寸法図.....2-13
2-4　KV-8000/7000 シリーズ用拡張入出力ユニット(KV-SIR32XT).....2-14
各部の名称と機能.....2-14
仕様.....2-15
結線･内部回路図.....2-16
外形寸法図.....2-17

2-5　拡張入力ユニット 2-18
各部の名称と機能 2-18
仕様 2-19
結線･内部回路図 2-20
外形寸法図 2-24
2-6　拡張出力ユニット 2-25
各部の名称と機能 2-25
仕様 2-26
結線･内部回路図 2-28
外形寸法図 2-33
2-7　拡張入出力ユニット 2-34
各部の名称と機能 2-34
仕様 2-35
結線･内部回路図 2-36
外形寸法図 2-39
2-8　バス接続ユニット 2-40
各部の名称と機能 2-40
外形寸法図 2-41
2-9　エクステンションユニット 2-42
各部の名称と機能 2-42
仕　様 2-43
接続･設置について 2-43
供給電源切換について 2-44
外形寸法図 2-45

3章　ユニットの設置および保守

3-1　ユニットの接続 3-2
ユニット接続時の注意 3-2
ユニットの組み立て 3-3
DINレールへの取り付け 3-5
3-2　ユニットの設置 3-6
設置環境 3-6
取り付け位置 3-6
3-3　配線時の注意 3-8
各ユニットへの配線時の注意 3-8
端子台ユニットと接続ケーブル 3-11
コネクタの組み立てと結線 3-15
PCなどの外部機器とUSBで通信する場合の注意点 3-19
3-4　保守 3-20
点検とお手入れについて 3-20
保守について 3-20
復旧方法について 3-22
復旧方法 3-23
システムプログラムアップデートについて 3-24

4章　ラダープログラミング

4-1　シーケンス制御とは 4-2
シーケンス制御の種類 4-4
4-2　プログラミング言語 4-5
ラダープログラム 4-5
ST言語 4-6
KVスクリプト 4-7
ニモニック 4-8
サフィックス 4-9
4-3　プロジェクト 4-11
プロジェクトの構成 4-11
プログラム容量 4-14
プロジェクトのプロテクト設定 4-16
4-4　デバイス 4-20
デバイス一覧 4-20
デバイスの表記方法 4-24
停電保持設定（ラッチ） 4-26
ビットデバイス 4-27
ワードデバイス 4-37
インデックス修飾 4-44
間接指定 4-46
ビットデバイスのワード単位扱い 4-50
ワードデバイスのビットデバイス扱い 4-52
内部レジスタ 4-53
4-5　変数 4-55
変数の種類 4-55
変数とデータ型一覧 4-56
配列 4-57
構造体 4-59
変数の設定 4-63
構造体の定義 4-66
ACNT　配列要素数取得 4-67
WSIZE　ワードサイズ取得 4-69
BSIZE　ビットサイズ取得 4-71
4-6　定数 4-73
4-7　データの取り扱い方 4-75
バイナリデータ 4-75
数値の表記 4-76
数値表記の相関 4-78
文字コード 4-81
4-8　プログラムの構成と動作 4-82
CPUユニットの動作 4-82
スキャンタイム 4-90
プログラムの構成 4-93

4-9　モジュール 4-98
概要 4-98
モジュールの種類 4-100
モジュールの実行順序 4-109
モジュールの構成例 4-110
モジュールシステムデバイス 4-113
モジュールパスワード 4-114
MDTSRT　モジュール実行開始 4-115
MDSTOP　モジュール実行停止 4-115
4-10　ファンクションブロック 4-117
概要 4-117
使用手順 4-119
ファンクションブロックの種類 4-120
引数設定 4-131
ファンクションブロックの配置方法 4-135
ファンクションブロックの使用例 4-137
引数の省略設定 4-138
引数の非公開設定 4-139
ファンクションブロックのネスト 4-140
ファンクションブロックのパスワード設定 4-141
ファンクションブロック読み取り専用設定 4-142
ファンクションブロックのシステムデバイス 4-143
ファンクションブロックのプログラムプロパティ 4-144
システムファンクションブロック 4-146
ファンクションブロックのライブラリ登録 4-147
FB　ファンクションブロックの呼び出し 4-148
FEND　ファンクションブロックの停止 4-148
FUN　ファンクションの呼び出し 4-148
FBCALL　インスタンス型ファンクションブロックコール 4-151
FBSTRT　インスタンス型ファンクションブロックスタート 4-151
≪KV-8000≫用システムファンクションブロック 4-154
SR　双安定ファンクションブロック（セット優先） 4-155
RS　双安定ファンクションブロック（リセット優先） 4-157
R_TRIG　立ち上がりエッジ検出器 4-159
F_TRIG　立ち下がりエッジ検出器 4-160
CTU　アップカウンタ 4-161
CTD　ダウンカウンタ 4-163
CTUD　アップダウンカウンタ 4-165
TON_100MS　100ms 単位オンディレイ 4-167
TON_10MS　10ms 単位オンディレイ 4-169
TON_1MS　1ms 単位オンディレイ 4-171
TON_10US　10us 単位オンディレイ 4-173
ファンクションブロックとマクロの違い 4-175

4-11 マクロ 4-178
概要 4-178
使用手順 4-180
マクロの種類 4-181
引数設定 4-186
マクロシステムデバイス 4-188
マクロの実行中ONビット 4-189
マクロパスワード 4-191
MCALL サブルーチン型マクロの実行 4-193
MSTRT 自己保持型マクロの実行 4-193
MEND 自己保持型マクロの停止 4-193
マクロとサブルーチンプログラムの違い 4-196
4-12 ローカルデバイス 4-200
ローカルデバイスの記述方法 4-200
ローカルデバイス一覧 4-201
ローカルデバイスの割付 4-202
ローカルデバイス入力･変換時のエラー 4-204
4-13 割り込み 4-206
概要 4-206
割り込みプログラムの実行方法 4-208
割り込みの優先度 4-209
割り込みプログラムで使用できない命令語 4-210
割り込み時の入出力処理 4-212
サンプルプログラム 4-214
割り込み処理用命令 4-217
DI 割り込み禁止 4-218
EI 割り込み許可 4-218
DIC 割り込み禁止範囲の設定 4-220
INT 割り込み実行 4-222
RETI 割り込み終了 4-222
IEDGE 割り込み検出条件書き込み 4-224
4-14 ファイルレジスタ 4-226
概要 4-226
バンク切換 4-226
初期値設定 4-227
ファイルレジスター括設定 4-227
ファイルレジスタ命令 4-229
4-15 プログラミングのコツ 4-236
導入から運用までの流れ 4-236
プログラム作成を支援する 4-238
見やすいプログラムを作成する 4-252
プログラム作成のポイント 4-258
デバッグの進め方 4-265

5章　CPU内蔵機能

5-1　内蔵機能概要 5-2
内蔵機能でできること 5-2
5-2　メモリカード 5-3
メモリカードでできること 5-3
メモリカードについて 5-6
メモリカードで使用するデバイス 5-10
プロジェクトデータの保存･読み出し 5-13
オートローディング機能 5-24
ストレージ命令 5-26
5-3　CPU メモリ（ユーザメモリ） 5-27
CPUメモリ（ユーザメモリ）でできること 5-27
CPUメモリについて 5-29
CPUメモリで使用するデバイス 5-32
プロジェクトデータの保存･読み出し 5-35
オートローディング機能 5-45
RAMモード 5-47
ストレージ命令 5-49
5-4　レシピロード･セーブ 5-50
レシピロード機能/レシピセーブ機能の概要 5-50
レシピデータ生成･読み出しの手順 5-52
レシピロード･セーブの手順 5-54
レシピロード完了コード･レシピセーブ完了コード 5-56
5-5　ロギング･トレース 5-57
ロギング機能/トレース機能の概要 5-57
使用手順 5-59
設定 5-60
CPUユニットバッファ容量設定 5-70
CSVファイルについて 5-71
ロギング命令 5-73
LOGE　ロギング･トレース許可 5-74
LOGD　ロギング･トレース禁止 5-74
TRGD　ロギングデータ取得 5-76
ロギング･トレース用デバイス一覧 5-78
ロギング･トレース機能の注意事項 5-82
5-6　運転記録機能 5-86
運転記録の概要 5-86
使用手順 5-87
設定 5-88
デバイス記録バッファ容量設定 5-97
運転記録機能用デバイス一覧 5-98
TRGR　デバイス記録データ取得 5-106
運転記録機能の注意事項 5-108

5-7 USB 通信 5-111
パソコンとの接続方法 5-111
USBドライバのインストール 5-111
5-8 イーサネット /EtherNet/IP 通信 5-112
CPU内蔵EtherNet/IP通信機能でできること 5-112
OPC UAサーバ機能でできること 5-112
5-9 アクセスウィンドウ 5-113
アクセスウィンドウでできること 5-113
ユニットのモニタ 5-117
デバイスモード 5-120
エラー内容の確認 5-128
エラークリア 5-129
デジタルトリマ 5-130
ユニットテスト 5-132
時計 5-134
CPUモニタ 5-136
アラーム 5-138
ストレージ 5-141
アクセスウィンドウ設定 5-154
クリア 5-156
KV STUDIOを使用したアクセスウィンドウ設定 5-160
アクセスウィンドウ初期画面表示機能 5-161
ユーザメッセージ 5-165
操作禁止設定 5-166
キーロック 5-167
5-10 ユニット間同期機能 5-169
ユニット間同期とは 5-169
ユニット間同期モジュール 5-170
ユニット間同期モジュールを使用しない場合の動作について 5-171
リフレッシュタイミングの設定（拡張ユニット用リレー･DMのみ対象） 5-172
ユニット間同期リフレッシュデバイス 5-173
ユニット間同期リフレッシュデバイスの設定方法 5-173
CPUデバイス書き込み／読み出し 5-174
CPUデバイス書き込み／読み出しの設定方法 5-174
ユニット間同期のシステム構成 5-175
ユニット間同期の周期 5-175
ユニット間同期の性能仕様 5-176
ユニット間同期の設定方法 5-177
ユニット間同期の設定手順 5-178
ユニット間同期トレース 5-180
5-11 カムスイッチ 5-182
カムスイッチ機能の概要 5-182
カムスイッチ命令 5-182
MCMP 多段コンパレータ 5-183
ABSENC アブソリュートエンコーダ 5-187
5-12 時計機能 5-191

6章　拡張I/Oユニット機能

6-1　入出力機能 6-2
汎用入出力 6-2
6-2　ユニット割り込み機能 6-4
設定手順 6-4
6-3　ユニット間同期機能 6-6
ユニット間同期とは 6-6
ユニット間同期機能実行時の動作概要 6-6
ユニット間同期機能時のユニット仕様 6-8
6-4　インプットキャプチャ機能 6-9
6-5　バッファリング入力機能 6-10
6-6　PWM 出力機能 6-19
6-7　KV-SIR32XT 用ユニット専用命令 6-23
KV-SIR32XT用ユニット専用命令一覧 6-23
U_RDBF　バッファリングデータ読み出し 6-24
U_DISOVP　入力過電圧保護機能設定 6-26
U_DISOVP　入力過電圧保護機能設定 6-28
6-8　KV-SIR32XT 用ユニット専用関数 6-30
KV-SIR32XT用ユニット専用関数一覧 6-30
U_RDBF　バッファリングデータ読み出し 6-31
U_DISOVP　入力過電圧保護機能設定 6-32
U_DISOCP　出力過電流保護機能設定 6-33
U_RDBF　バッファリングデータ読み出し 6-34
U_DISOVP　入力過電圧保護機能設定 6-35
U_DISOCP　出力過電流保護機能設定 6-36
6-9　アクセスウィンドウ 6-37
アクセスウィンドウ 6-37
設定画面 6-39
モニタ画面 6-40
ユニットテスト 6-41
6-10　ユニットモニタ 6-42
ユニットモニタの概要 6-42
ユニットモニタの表示 6-42
6-11　I/O ユニット参考プログラム例 6-43
6-12　I/O ユニットのエラーコード 6-47

付　録

1　PLC 本体エラー一覧 ..... 付 -2

2　CR/CM 一覧 ..... 付 -10

コントロールリレー CR 付-10

コントロールメモリ CM 付-14

3　ASCII コード表 ..... 付 -18

ASCIIコード表 付-18

4　命令語一覧 ..... 付 -19

5　使用できない命令語一覧 ..... 付 -21

6　ニモニック短縮入力一覧 ..... 付 -23

7　ショートカットキー一覧 ..... 付 -25

ユニットエディタで有効なショートカットキー 付-25

KV STUDIOで有効なショートカットキー 付-27

GPPQ/GPPA形式のショートカット一覧 付-33

CX形式のショートカット一覧 付-34

Logix形式のショートカット一覧 付-35

キーの使い方 付-37

8　ユーザ認証機能 ..... 付 -39

9　使用できない文字一覧 ..... 付 -40

10　既存の KV シリーズ拡張ユニット使用時の注意事項について ..... 付 -43

対応機器一覧 付-43

使用時の制限事項 付-44

11　KV-8000 シリーズへの機種変更時の注意事項 ..... 付 -46

機種置き換え時の手順 付-46

CPUのハードウェア特殊機能などに関する差分 付-47

デバイス範囲の比較表 付-50

差分がある命令語 付-51

コントロールリレー、コントロールメモリの差分 付-51

変数とラベルの差分 付-52

12　索　引 ..... 付 -54

マニュアルの読みかた

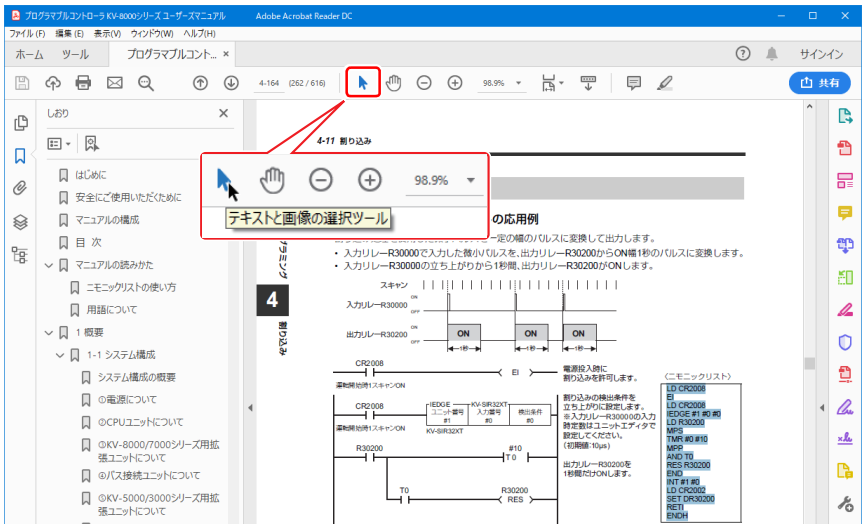
このマニュアルで使用されている用語およびニモニックリストの使い方について説明します。

ニモニックリストの使い方

参考ラダープログラムが紹介されている各ページには、ニモニックリストが記載されています。
このニモニックリストを利用すると、ラダープログラムが簡単に入力できます。

1 「Adobe Reader」でマニュアルを表示させます。

2 「選択ツール」を使用して、ニモニックリストをコピーします。

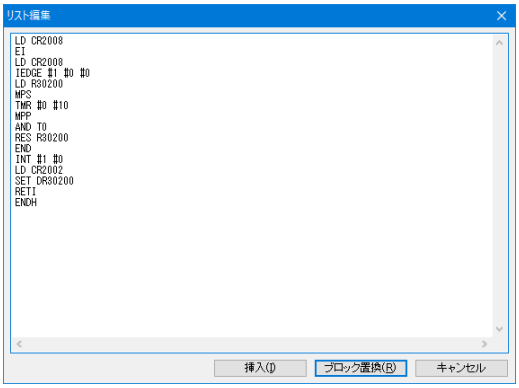


3 《KV STUDIO》のラダー編集エリアで、参考ラダープログラムを挿入したいセルをクリックし、メニューから【編集(E)】→【リスト編集(L)】を選択して、[リスト編集]ダイアログボックスを表示します。

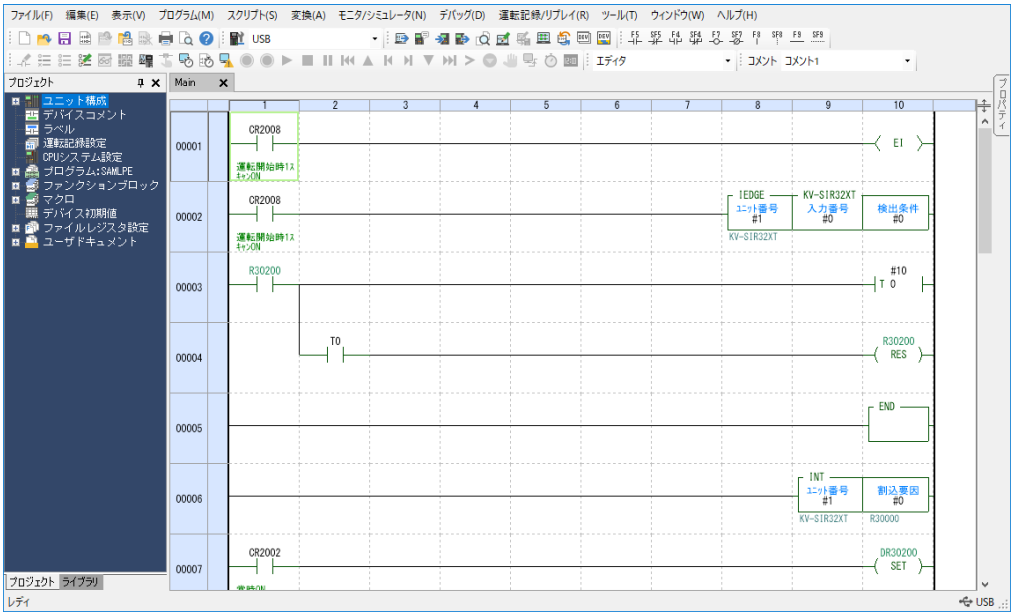
別手順

- Ctrl ＋ D

表示した[リスト編集]ダイアログボックスに、コピーしたニモニックリストを貼り付けます。
貼り付けは、[リスト編集]ダイアログボックスの右クリックメニューから[貼り付け(P)]を選択します。



4 〔挿入(I)〕ボタンをクリックすると、参考ラダープログラムが表示されます。



用語について

このマニュアルの説明では一部を除き、次のような用語を使用しています。

用語	説明
プログラマブルコントローラ	プログラムを変更することで自由に機器を制御できる電子装置で、PLC(Programmable Logic Controller)とも呼ばれています。
《KV-8000シリーズ》《KV-8000》	当社製プログラマブルコントローラKV-8000/8000Aです。
《KV-7000シリーズ》	当社製プログラマブルコントローラKV-7500/7300です。
《KV-5000/3000シリーズ》	当社製プログラマブルコントローラKV-5500/5000/3000です。
CPUユニット	《KV-8000》《KV-7500/7300》《KV-5500/5000/3000》です。
拡張ユニット	《KV-8000シリーズ》で使用できるCPUユニット以外の拡張I/Oユニット、拡張特殊ユニットです。
バス接続ユニット	《KV-8000シリーズ》で《KV-5000/3000シリーズ》用の拡張ユニットを使用する際に、必要になる接続ユニットです。バス接続ユニット使用時は右端にエンドユニットを取り付ける必要があります。
S-Unit	KV-8000シリーズのCPUインナーバスに対応した拡張ユニットです。
X-Unit	KV-8000シリーズのCPUインナーバスに対応した高機能拡張ユニットです。
CPUインナーバス	KV-8000シリーズCPUからバス接続ユニットまでの、ユニット割込に対応した高速インナーバスのことです。
エンドユニット	《KV-5000/3000シリーズ》用の拡張ユニットを使用する際に右端に取り付けるユニットです。
《KV STUDIO》	KV-8000シリーズのプログラム作成をサポートするソフトウェアです。
ラダープログラム	《KV STUDIO》で作成されたプログラムです。

1

概要

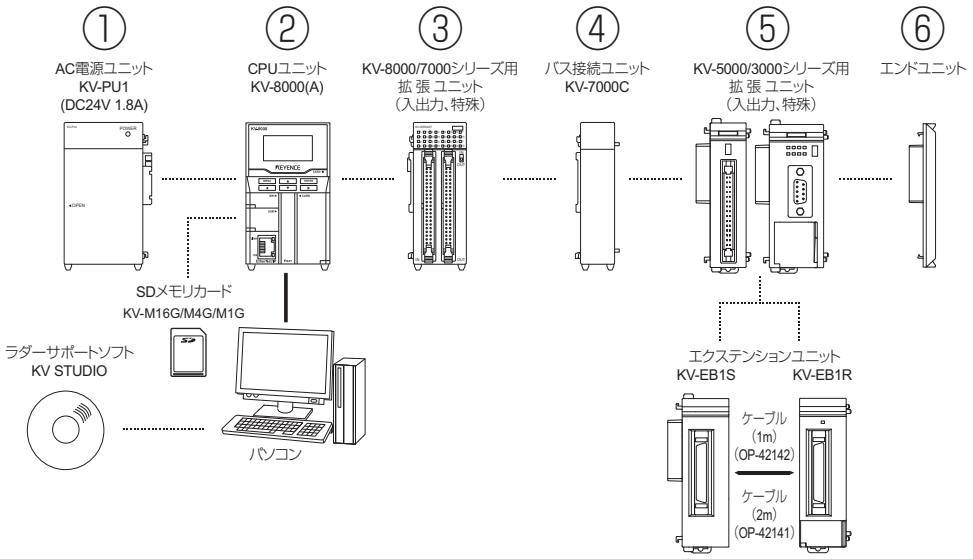
KV-8000シリーズのシステム構成･パソコンとの通信経路･ユニットの占有領域などを説明します。

1-1　システム構成..... 1-2
1-2　ユニットの予約について..... 1-6

1-1 システム構成

システム構成の概要

《KV-8000シリーズ》を使用する際のシステム構成は、以下のようになります。
各ユニットは、必ず電源がOFFの状態で接続してください。



ポイント　**バス接続ユニット、KV-5000/3000シリーズの拡張ユニットを使用する場合のみ、右端にエンドユニットが必要です。**

①電源について

■ AC100～240Vを使用する

AC100～240VでKV-8000シリーズを動作させる場合は、AC電源ユニット《KV-PU1》を一番左側に接続して使用します。CPUユニット1台に対して《KV-PU1》は1台のみ接続可能です。

「2-3 AC電源ユニット」

■ DC24Vを使用する

DC24VでKV-8000シリーズを動作させる場合は、AC電源ユニット《KV-PU1》は使用しません。DC24Vは《KV-8000》のDC電源入力端子台から供給します。

「2-2 CPUユニット」

②CPUユニットについて

CPUユニットの左側にはAC電源ユニット、右側には拡張入出力ユニットや拡張特殊ユニット、KV-7000Cが接続できます。右側にバス接続ユニットを使用してKV-5000/3000シリーズ用の拡張ユニットを接続する場合は、必ずエンドユニットを接続します。

参 考　バス接続ユニット接続時はKV-5000/3000シリーズ用の拡張ユニットを接続しない場合でも、エンドユニットは必要です。

「2-2 CPUユニット」

アクセスウィンドウによるデバイス値のモニタや、SDメモリカード、またはユーザメモリへのラダープログラムやデバイスデータの読み書きができます。

「5章 CPU内蔵機能」

KV-8000シリーズでは、モジュール・ファンクションブロック・マクロ・ローカルデバイス・KVスクリプトなどの機能を使用してプログラミングを効率化できます。

「4章 ラダープログラミング」
『スクリプトプログラミングマニュアル』
『KV STUDIOユーザーズマニュアル』

③KV-8000/7000シリーズ用拡張ユニットについて

KV-8000/7000シリーズ用拡張ユニットは必ずCPUユニットの右側、KV-7000Cの左側に接続します。
（KV-5000/3000シリーズの拡張ユニットを使用しない場合、KV-7000Cは接続不要です。）
1台のCPUユニットに16台までのKV-8000/7000シリーズ用拡張ユニットが接続できます。

「2-4 KV-8000/7000シリーズ用拡張入出力ユニット（KV-SIR32XT）」
『各拡張特殊ユニットのユーザーズマニュアル』

④バス接続ユニットについて

KV-8000シリーズにKV-5000/3000シリーズ用拡張入出力ユニットや拡張特殊ユニット使用時に取り付けます。KV-7000C使用時は、右側に必ずエンドユニットを取り付けてください。

「2-8 バス接続ユニット」

⑤KV-5000/3000シリーズ用拡張ユニットについて

KV-5000/3000シリーズ用拡張入出力ユニットや拡張特殊ユニットをKV-8000シリーズに接続して使用する場合は、必ずバス接続ユニットの右側に取り付けて使用してください。一番右端の拡張ユニットの右側に、必ずエンドユニットを接続します。
エクステンションユニットを2セット使用すると、KV-8000/7000シリーズ用の拡張ユニットと合わせて最大48台まで接続できます。（一部の拡張特殊ユニットを除く）

『KV-5000/3000シリーズ用の各拡張特殊ユニットのユーザーズマニュアル』

⑥エンドユニットについて

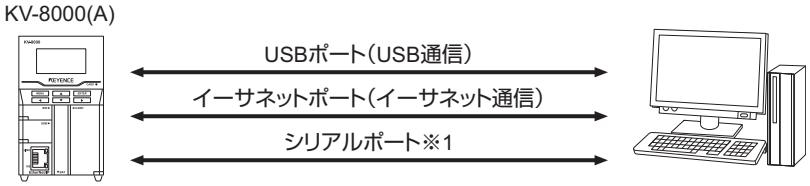
バス接続ユニット、KV-5000/3000シリーズ用拡張入出力ユニットや拡張特殊ユニットの右側に接続してください。接続されていない場合、エラーが発生し、動作できません。

ポイント **CPUユニットおよび、KV-8000/7000シリーズ用拡張ユニットのみ接続している場合、エンドユニットは必要ありません。**

通信機器との接続方法

CPUユニットとパソコンやタッチパネルと通信する場合の通信経路について説明します。

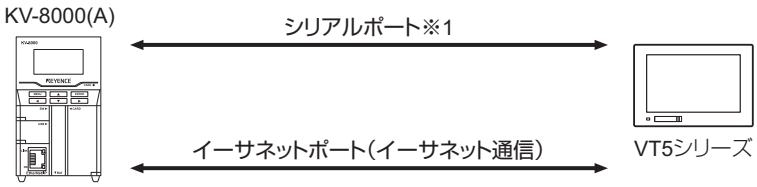
■ パソコンと通信する場合



※1 別途シリアルコミュニケーションユニットとパソコン側に変換器が必要

ポイント
VTや、EtherNet/IP™などのネットワークを経由して通信することもできます。
詳細については『KV STUDIOユーザーズマニュアル』を参照してください。

■ タッチパネルと通信する場合

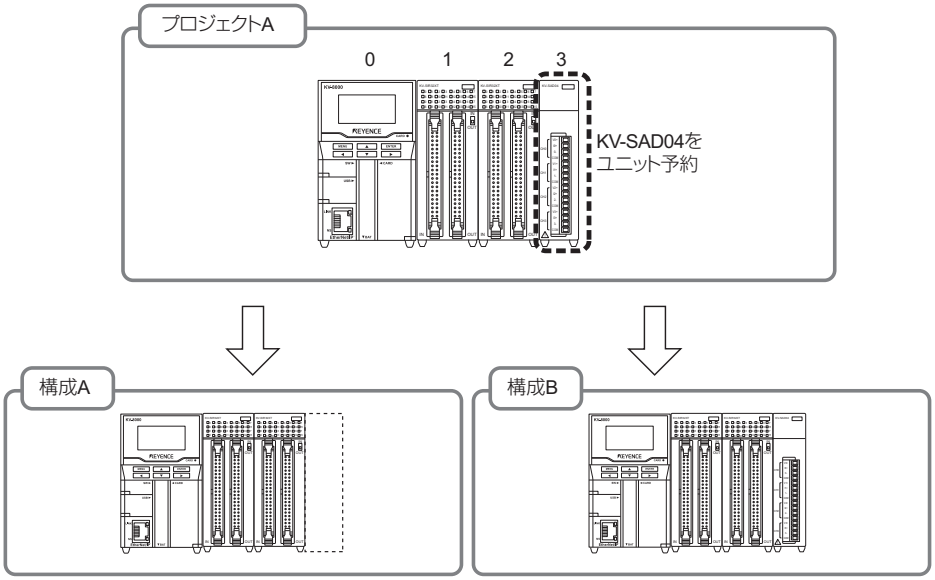


※1 別途シリアルコミュニケーションユニットとパソコン側に変換器が必要

ポイント
- **USBを使用して通信する場合、USBドライバのインストールが必要です。**
- **ご使用のOSがWindows 11/10/8/7の場合、KV STUDIOインストール時に自動的にインストールされます。**

1-2 ユニットの予約について

ユニットの予約機能を使用すると、ユニット構成が異なる場合でも、同じプロジェクトでユニットの管理ができます。たとえば、拡張アナログ入力ユニットをオプションで取り付けるような装置の場合でも、同じプロジェクトで動作するため、プロジェクトの管理が簡単になります。



動作可能　　動作可能

ポイント

- **ユニットを予約設定している際に、実際の構成にはユニットが存在する場合は、ユニット構成エラーは発生せず、正常に動作します。**
- **予約設定をしているユニット以外について、ユニットエディタで設定したユニット情報と、実際に接続されている機器情報と異なる場合、ユニット台数エラーまたはユニット種別エラーが発生し、動作できません。**
- **実際の構成に存在しないユニット(予約設定している)番号を命令語(ファンクションブロック命令など)のオペランドに使用すると、変換時はエラーが発生せず、実行時に演算エラー(軽度エラー)が発生します。なお、以下の方法で演算エラーを回避できます。**
  - **拡張ユニット有無リレーを実行条件にANDで加える**
  - **CPUシステム設定(エラー発生時の動作設定)で、CPU動作を続行(無視)にする**
- **KV-5000/3000シリーズの拡張ユニットを使用している場合、ユニット番号の変更を反映するためには電源入り切りが必要になる拡張ユニットがあります。**

参考

- 各ユニット番号の拡張ユニットの接続確認には、拡張ユニットn有無リレー(CR4001～)を使用できます。
- 予約ユニットの右側に拡張ユニットを追加することもできます。
- 複数台の同じユニットを連続して予約設定にし、実際の構成にそれよりも少ない台数のユニットを接続した場合、接続されたユニットは左から詰めて並べられたユニット番号として動作します。たとえば、ユニット番号、1、2、3にKV-SIR32XTを予約設定し、2台のKV-SIR32XTを使用すると、ユニット番号、1、2のユニットとして認識して動作します(1、3などとユニット番号の間隔を空けては使用できません)。

2

仕様

KV-8000シリーズのCPUユニット･AC電源ユニット･拡張I/Oユニット･バス接続ユニットの仕様および配線を説明します。

2-1 シリーズ共通仕様 2-2
2-2 CPUユニット 2-5
2-3 AC電源ユニット 2-11
2-4 KV-8000/7000シリーズ用拡張入出力ユニット(KV-SIR32XT) 2-14
2-5 拡張入力ユニット 2-18
2-6 拡張出力ユニット 2-25
2-7 拡張入出力ユニット..... 2-34
2-8 バス接続ユニット 2-40
2-9 エクステンションユニット 2-42

2-1 シリーズ共通仕様

KV-8000シリーズの仕様と電源断時の動作について説明します。

一般仕様

<table>
<tr><th>項目</th><th colspan="5">仕様</th></tr>
<tr><td>システム構成</td><td colspan="2">KV-5000/3000シリーズ拡張ユニットを使用したシステム構成の場合</td><td colspan="3">KV-8000/7000シリーズ拡張ユニットのみのシステム構成の場合</td></tr>
<tr><td>電源電圧</td><td colspan="2">DC 24V（±10%）</td><td colspan="3">DC 24V（-15%,+20%）</td></tr>
<tr><td>使用周囲温度※1</td><td colspan="2">0～+50℃（氷結しないこと）</td><td colspan="3">0～+55℃（氷結しないこと）</td></tr>
<tr><td>使用周囲湿度</td><td colspan="2">10～95%RH（結露しないこと）</td><td colspan="3">5～95%RH（結露しないこと）</td></tr>
<tr><td>保存周囲温度</td><td colspan="2">-20～70℃</td><td colspan="3">-25～75℃</td></tr>
<tr><td>保存周囲湿度</td><td colspan="2">10～95%RH（結露しないこと）</td><td colspan="3">5～95%RH（結露しないこと）</td></tr>
<tr><td>使用雰囲気</td><td colspan="5">塵埃、腐食性ガスがひどくないこと</td></tr>
<tr><td>使用標高</td><td colspan="5">2000m以下</td></tr>
<tr><td>汚染度</td><td colspan="5">2</td></tr>
<tr><td>過電圧カテゴリ</td><td colspan="5">Ⅰ（KV-PU1 使用時 Ⅱ）</td></tr>
<tr><td>耐ノイズ性</td><td colspan="5">1500Vp-p以上　パルス幅 1μs,50ns（ノイズシミュレータによる）
IEC規格準拠（IEC61000-4-2/3/4/6）</td></tr>
<tr><td>耐電圧</td><td colspan="5">AC 1500V 1分間 電源端子と入出力端子間、および外部端子一括とケース間</td></tr>
<tr><td>絶縁抵抗</td><td colspan="5">50MΩ以上
（DC 500Vメガにて電源端子と入出力端子間および外部端子一括とケース間）</td></tr>
<tr><td rowspan="9">耐振動※2</td><td rowspan="9">JIS B 3502
IEC61131-2に準拠</td><td colspan="3">断続的な振動がある場合</td><td>掃引回数</td></tr>
<tr><td>周波数</td><td>加速度</td><td>片振幅</td><td rowspan="8">X,Y,Z各方向
10回
（100分間）</td></tr>
<tr><td>5～9Hz</td><td>－</td><td>3.5mm</td></tr>
<tr><td>9～150Hz</td><td>9.8m/s^2</td><td>－</td></tr>
<tr><td colspan="3">連続的な振動がある場合</td></tr>
<tr><td>周波数</td><td>加速度</td><td>片振幅</td></tr>
<tr><td>5～9Hz</td><td>－</td><td>1.75mm</td></tr>
<tr><td>9～150Hz</td><td>4.9m/s^2</td><td>－</td></tr>
<tr><td>耐衝撃</td><td colspan="5">加速度150m/s^2、作用時間11ms、X, Y, Z 各方向3回</td></tr>
<tr><td>消費電流</td><td colspan="5">CPUユニット : 400mA以下
バス接続ユニット : 30mA以下</td></tr>
<tr><td>質量</td><td colspan="5">CPUユニット : 約340g
バス接続ユニット : 約130g
KV-B1（バッテリ）: 約10g</td></tr>
</table>

※1　制御盤内のユニット中央下部で規定します。
※2　DINレール取付時。

性能仕様

項目			仕様	
演算制御方式			ストアードプログラム方式	
入出力制御方式			リフレッシュ方式	
プログラム言語			拡張ラダー、KVスクリプト、ニモニック、ST言語	
命令語数			基本命令 : 80種181語 応用命令 : 55種73語 演算命令 : 128種327語 拡張命令 : 78種133語 合計 : 341種714語	
命令実行速度	基本命令		最小0.96ns	
	応用命令		最小5.75ns	
CPUメモリ容量 （プロジェクト容量＋ユーザメモリ容量）			64MB	
プログラム容量			約1500Kステップ	
I/Oユニット最大装着数※2			16台（KV-8000/7000シリーズ拡張ユニットのみ） 48台（KV-8000/7000シリーズ拡張ユニット、KV-5000/3000シリーズ拡張ユニット（エクステンションユニット（KV-EB1）使用時））	
最大I/O点数			拡張時最大3072点 （KV-EB1S/KV-EB1R 2台使用し、64点ユニット使用時）	
入力リレー		R	合計32000点 1ビット（R00000～R199915）	
出力リレー		R		
内部補助リレー		R		
リンクリレー		B	32768点 1ビット（B0000～B7FFF）	
内部補助リレー		MR	64000点 1ビット（MR000000～MR399915）	
ラッチリレー		LR	16000点 1ビット（LR00000～LR99915）	
コントロールリレー		CR	1280点 1ビット（CR0000～CR7915）	
タイマ		T	4000点 32ビット（T0000～T3999）	
カウンタ		C	4000点 32ビット（C0000～C3999）	
データメモリ		DM	65535点 16ビット（DM00000～DM65534）	
拡張データメモリ		EM	65535点 16ビット（EM00000～EM65534）	
ファイルレジスタ	カレントバンク	FM	524288点 16ビット	（FM00000～FM32767×16バンク）
	連番方式	ZF		（ZF000000～ZF524287）
リンクレジスタ		W	32768点 16ビット（W0000～W7FFF）	
テンポラリメモリ		TM	512点 16ビット（TM000～TM511）※3	
インデックスレジスタ※5		Z	23点 32ビット（Z1～Z23）	
コントロールメモリ		CM	7600点 16ビット（CM0000～CM7599）	
コメント・ラベル変数本体格納数	デバイスコメント		約224000個	
	ラベル※4		約285000個	
	変数※4		約100000個※6	
停電保持機能	プログラムメモリ		フラッシュROM　1万回書き換え可能	
	デバイス※1		不揮発RAM	
	カレンダタイマ		バックアップ用コンデンサ 約15日（@25度） （KV-B1（ バッテリ）使用時 約5年（@25度））	
自己診断機能			CPU異常、RAM異常、その他	

※1　対象デバイスは≪KV STUDIO≫の［CPUシステム設定］→［停電保持設定］で設定可能です。
※2　I/Oユニット最大装着数により制限される場合があります。
※3　TM000～TM003は演算命令の結果が格納される領域です。
※4　CPU機能バージョン1.0ではラベル、2.0以降では変数になります。
※5　Z11～Z23はシステム予約されているため使用できません。
※6　変数名の文字数によっては、100000個定義できない場合があります。

電源断時の動作

電源電圧の低下

- 電源電圧が低下すると、CPUユニットは停止し、出力をOFFします。

自動復帰

- 電源電圧が復旧すると、自動的に運転を再開します。

瞬時停電

- AC電源の場合は20ms未満、DC電源の場合は24V時:10ms、20.4V時:5ms未満の瞬停が発生しても、動作を継続します。

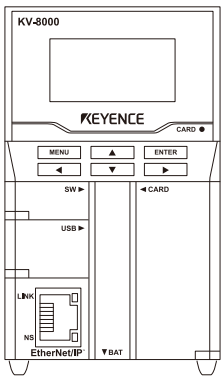
2-2 CPUユニット

パッケージ内容

梱包の中には次の機器や備品が入っています。使用する前にこれらが揃っているかお確かめください。

■ KV-8000(A) 同梱品

CPUユニット
KV-8000(A)



取扱説明書

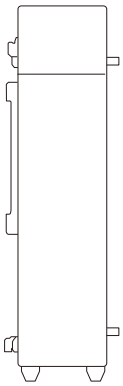
- 拡張ユニットコネクタカバー
- 電源ユニットコネクタカバー
- バッテリカバー

※ カバー3種類はCPUユニットに取り付けられた状態で同梱されています。

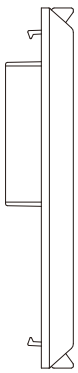
KV-5000/3000シリーズ拡張ユニットを使用する場合は、別途バス接続ユニットKV-7000Cが必要になります。
KV-7000Cの同梱品は以下のとおりです。

■ KV-7000C 同梱品

バス接続ユニット



エンドユニット



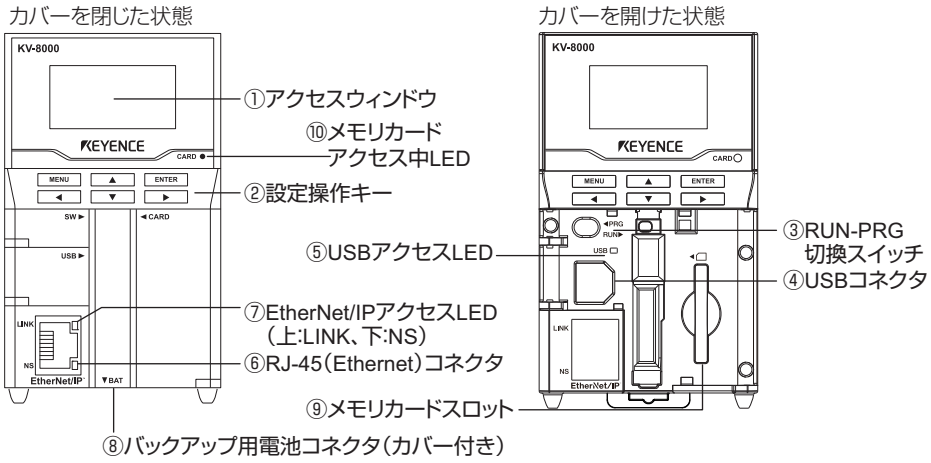
取扱説明書

梱包には万全を期しておりますが、万一不良、破損がございましたら、最寄りの営業所までご連絡ください。

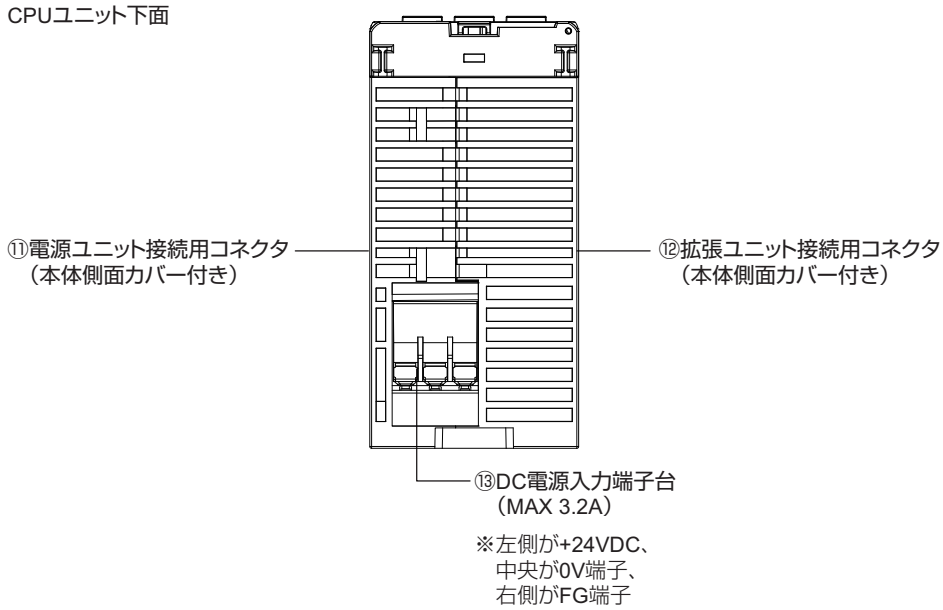
各部の名称と機能

● KV-8000(A)

CPUユニット正面



CPUユニット下面

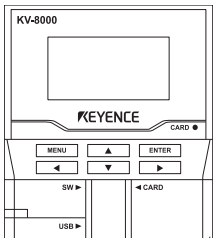


番号	名称	機能
①	アクセスウィンドウ	ユニットのモニタや設定をします。 バックライトで動作状態を示します。 RUN : 白 PRG : 赤
②	設定操作キー	アクセスウィンドウの操作に使用します。
③	RUN-PRG切換スイッチ	RUNモードとPRGモードを切り換えます。
④	USBコネクタ	USBとの接続用コネクタです。
⑤	USBアクセスLED	USBでのデータ送受信時に点灯します。
⑥	RJ-45（Ethernet）コネクタ	ネットワークケーブルのコネクタ（RJ-45モジュラージャック）を接続する差込口です（100BASE-TX用）。
⑦	EtherNet/IPアクセスLED （上 : LINK、下 : NS）	LINK…相手機器とのリンク状態を表示します。 緑点灯 : リンク成立 緑点滅 : データ送受信中 消灯 : リンク不成立 NS…データ送受信状況を表示します。 緑点灯 : コネクション確立 緑点滅 : コネクション未確立 赤点灯 : IPアドレスが重複している状態 赤点滅 : 1つ以上のコネクションでタイムアウトが発生 消灯 : IPアドレス未設定
⑧	バックアップ用電池コネクタ （カバー付き）	KV-B1を取り付ける際に使用します。
⑨	メモリカードスロット	メモリカード挿入用のスロットです。
⑩	メモリカードアクセス中LED	メモリカードへのアクセス中に点灯します。
⑪	電源ユニット接続用コネクタ （本体側面カバー付き）	KV-PU1（電源ユニット）接続用のコネクタです。
⑫	拡張ユニット接続用コネクタ （本体側面カバー付き）	拡張ユニット接続用のコネクタです。
⑬	DC電源入力端子台	DC24V電源入力用の端子台（M3ビス）です。（MAX3.2A）

注 記	**RJ-45（Ethernet）コネクタにUSBケーブルを接続しないでください。（破損の恐れがあります）**

RUN-PRG切換スイッチ

CPUユニットには、RUNモードとPRGモードの
2つのモードがあります。
各モードは本体右図のRUN－PRG切換スイッチで切換えます。



RUNモード ：プログラムを実行します。
PRGモード ：プログラムを停止し、すべての出力をOFFにします。

ポイント **重度エラーが発生した場合、切換スイッチが「RUN」の位置にあっても「PRGモード」に移行します。**

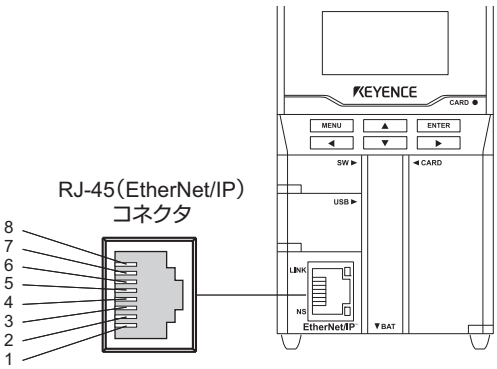
参考 切換スイッチが「RUN」の位置にある場合、《KV STUDIO》で「PRGモード」「RUNモード」の切換が可能です。

結線・内部回路図

■ KV-8000(A)

RJ-45コネクタでは、パソコンやタッチパネルなどの外部機器とイーサネット通信ができます。また、EtherNet/IP通信も可能です。

「5-8 イーサネット/EtherNet/IP通信」5-112ページ



ピン番号	MDI信号	信号機能
1	TD＋	送信データ（＋）
2	TD－	送信データ（－）
3	RD＋	受信データ（＋）
4	－	－
5	－	－
6	RD－	受信データ（－）
7	－	－
8	－	－

■ Ethernet通信仕様

項目	仕様
伝送速度	100Mbps(100BASE-TX)
伝送媒体	STPまたはカテゴリ5以上のUTP※1
最大ケーブル長※2	100m
最大ハブ接続段数※3	2
MDI/MDI-X 自動切換機能	あり
表示	LINK(緑) 相手機器とのリンク状態を表示します。 点灯:リンク成立。※4 消灯:リンク不成立。 NS(緑、赤)※5ネットワークの通信状態を表示します。 消灯:IPアドレスが未設定。 緑点灯:1つ以上のコネクションが確立。 緑点滅:コネクションが1つも確立できていない。 赤点灯:IPアドレスが重複している。 赤点滅:KV-8000がターゲット※6となるコネクション(タグ設定)が1つ以上タイムアウトしている。

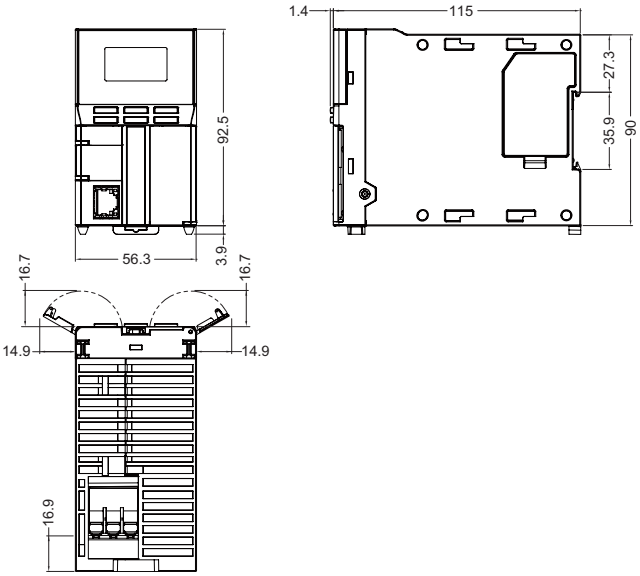
KV-8000(A)はIEEE802.3規格に準拠しています。

※1 STP＝シールドツイストペアケーブル
UTP＝非シールドツイストペアケーブル
※2 最大ケーブル長とは、CPUユニットとハブの間の距離のことです。
※3 イーサネットスイッチを使用する場合、接続段数の制限はありません。
※4 リンク成立とは、通信する互いの機器に電源が投入されていて、イーサネットに正しくケーブル接続されている状態をいいます。
※5 NS(緑、赤)表示灯は、電源投入時(Self test中)に、NSの緑NSの赤の順に点灯します。
※6 コネクションをオープンする側をオリジネータ、オープンされる側をターゲットといいます。

外形寸法図

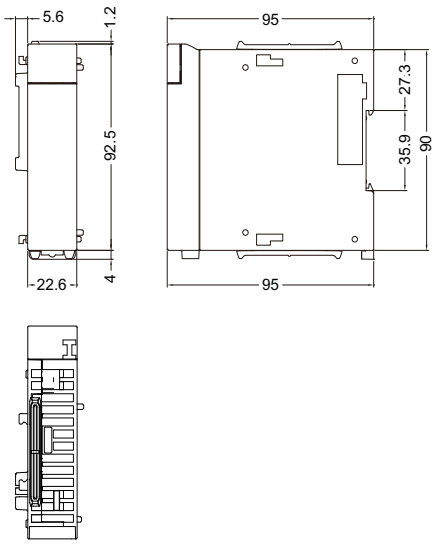
■ CPUユニット

単位 : mm



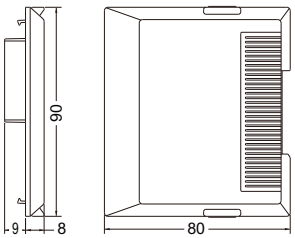
■ バス接続ユニット

単位 : mm



■ エンドユニット

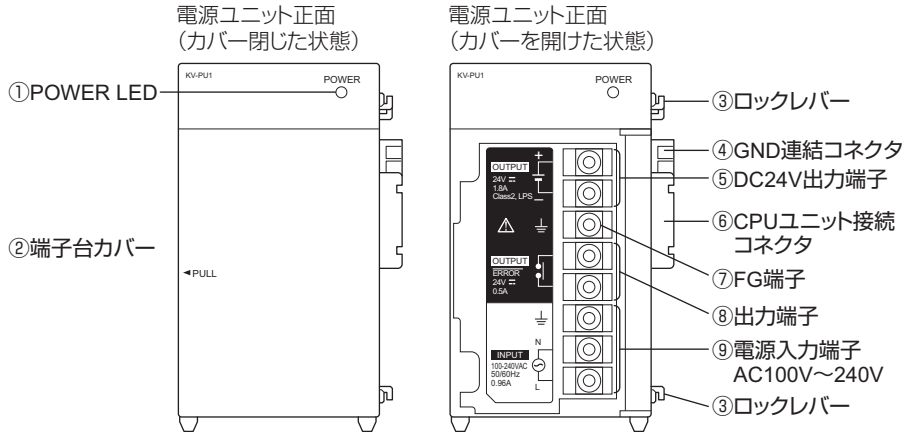
単位 : mm



2-3 AC電源ユニット

《KV-PU1 AC電源ユニット》の各部の名称と機能、および仕様について説明します。

各部の名称と機能



番号	名称	機能
①	POWER LED	DC24Vが供給できている場合ON（緑点灯）します。
②	端子台カバー	端子台の保護カバーです。
③	ロックレバー	ユニットの連結をロックします。
④	GND連結コネクタ	接続したユニットとGNDを連結します。
⑤	DC24V出力端子	サービス電源用DC24V出力端子です。（M3.5ビス）メイン電源用と合計で1.8Aです。
⑥	CPUユニット接続コネクタ	CPUユニット接続用コネクタです。
⑦	FG端子	FG端子です。（M3.5ビス）
⑧	出力端子	エラー出力用端子台です。（M3.5ビス） 正常時ON（PRG時含む）し、重度エラー発生時にOFFします。
⑨	電源入力端子 AC100V～240V	AC100～240V（50/60Hz）電源に接続します。（M3.5ビス）

注 記	
	• **過電流時（POWER LEDがOFF）は、一旦電源をOFFにして原因をとりのぞいてください。** • **過電圧がかかった場合には、出力が停止します。（POWER LEDが消灯） 電源をOFFし、約3分以上間隔をあけて、再度電源をONしてください。** • **KV-PU1使用時に、CPUユニットの電源端子より電源を供給しないでください。**

ポイント

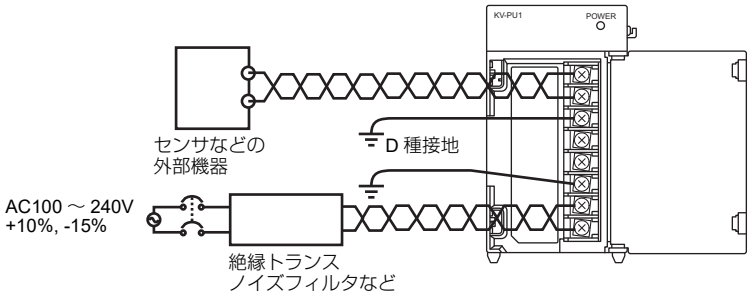
KV-PU1は、必ずCPUユニットのすぐ左側に接続してください。
また、KV-PU1使用時は、CPUユニットの電源端子を24V電源の供給源として使用できません。

仕様

項　目		仕　様
入力電源電圧		AC 100～240V（+10%,-15%）（50/60Hz）
出力電圧		DC 24V±10%
出力容量		1.8A（各種ユニットへの供給とサービス電源の合計）
内部消費電流		0.96A以下
瞬停時間		20ms以下（定格入出力条件にて）
立ち上がり時間		Max3秒以下
出力形態		リレー（B接点）
定格負荷		DC24V　0.5A
ON抵抗		50mΩ以下
応答時間	OFF→ON	10ms以下
	ON→OFF	5ms以下
リレー寿命		電気的 : 10万回以上（20回/分） 機械的 : 2000万回以上
リレー交換		不可
質量		約300g

配線方法

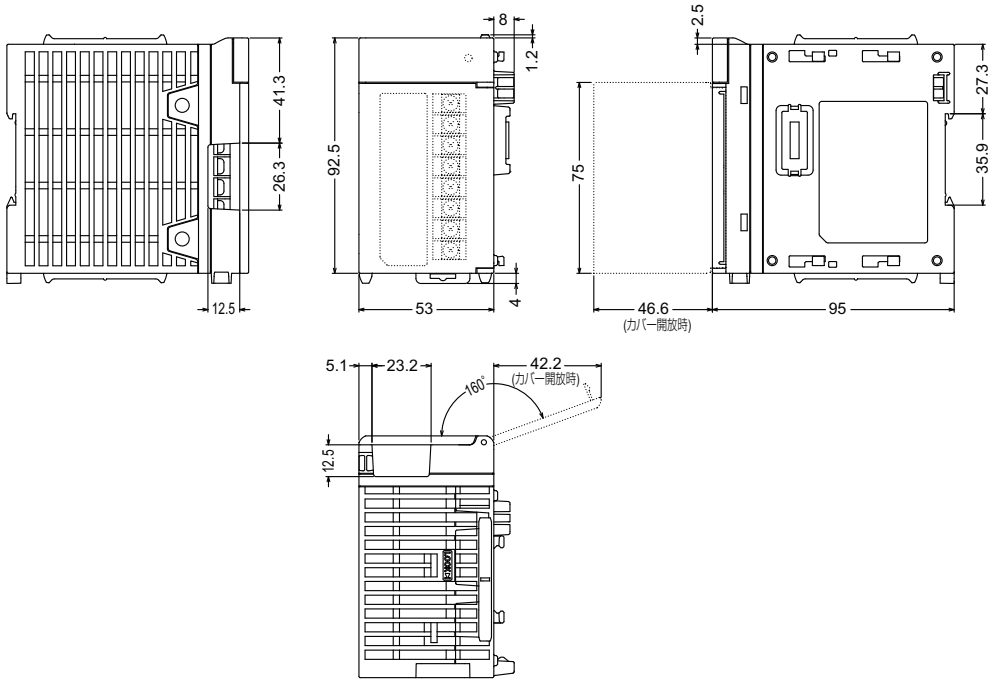
《KV-PU1 AC電源ユニット》は、図のように配線します。



ポイント

- **ラインノイズを低減するために、1:1の絶縁トランスやノイズフィルタなどを接続してください。**
- **誘電の影響を低減するために、ケーブルはツイスト状にしてください。**
- **KV-PU1は電源遮断装置としてヒューズを内蔵しています。**
- **1次側と2次側の回路は機器内で強化絶縁されています。**

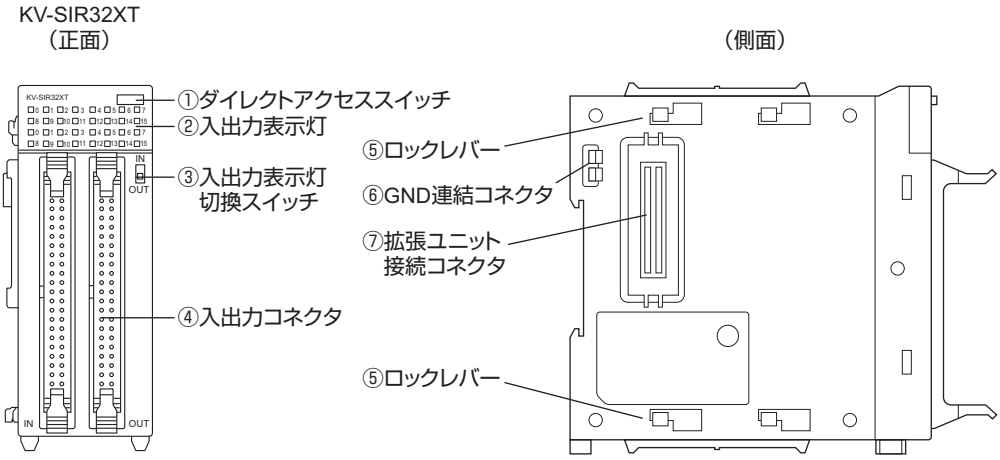
外形寸法図


41.3
26.3
12.5
8
1.2
92.5
53
4
2.5
75
46.6
(カバー開放時)
95
27.3
35.9
5.1
23.2
42.2
(カバー開放時)
160°
12.5
POWER

2-4 KV-8000/7000シリーズ用拡張入出力ユニット（KV-SIR32XT）

《KV-SIR32XT拡張入出力ユニット》の各部の名称と機能、および仕様について説明します。

各部の名称と機能



番号	名称	機能
①	ダイレクトアクセススイッチ	スイッチをONすることにより、CPUのアクセスウィンドウで拡張ユニットの状態をモニタできます。また、スイッチの色により、拡張ユニットの状態がわかります。 緑点灯：正常動作のとき 緑点滅：緑点灯状態でスイッチをONし、ユニットが選択されたとき 赤点灯：ユニットエラーが発生したとき（一部例外を除く※1） 赤点滅：赤点灯状態でスイッチをONし、ユニットが選択されたとき
②	入出力表示灯	ユニットの入出力状態を表示します。 赤点灯：導通状態 消灯：非導通状態
③	入出力表示灯切換スイッチ	入出力表示の切換スイッチです。 IN：入力の状態を表示 OUT：出力の状態を表示
④	入出力コネクタ	入出力用のコネクタです。
⑤	ロックレバー	ユニットの連結をロックします。
⑥	GND連結コネクタ	接続したユニットとGNDを連結します。
⑦	拡張ユニット接続コネクタ	CPUユニットまたはKV-8000/7000シリーズ用拡張ユニットと連結します。

※1　以下の場合、ユニットエラー状態ではなくても赤点灯します。
電源投入時、CPUユニットのユニット情報がCPU1台の構成になっている、ユニット台数エラー、ユニット種別エラーが発生しているなどCPUユニットがシステムエラーになっているとき
このとき、アクセスウィンドウでエラー表示されます。

※　KV-SIR32XT接続用40ピンコネクタはオプションで用意しています。

仕様

型式			KV-SIR32XT(32点+32点)	
外部接続方式			コネクタ(MIL規格)	
入力	入力点数		32点	
	入力モード		DC 24Vモード	DC 5Vモード※2 (過電圧保護機能あり※3)
	入力最大電圧		DC28.8V	DC6.0V
	入力定格電圧		DC 24V/5.1mA	DC 5V/8.8mA
	最小ON電圧		19V	3.5V
	最大OFF電流		1.5mA	—
	最大OFF電圧		—	1.5V
	コモン方式		16点/1コモン(2端子)※1	
	入力時定数			

入力時定数設定	TYP	MAX
1μs	1.2μs(0.7μs)	2.6μs
10μs	2.4μs(1.9μs)	3.9μs
20μs	10μs	12μs
100μs	73μs	82μs
500μs	420μs	450μs
1ms	840μs	900μs
5ms	4.5ms	4.8ms
10ms	9.0ms	9.6ms
50ms	45ms	48ms

※ 括弧内はON→OFF時

	入力インピーダンス		4.4kΩ	350Ω
出力	出力点数		32点	
	出力形態		MOSFET(N-ch)(過電流保護機能あり)※4	
	定格負荷		DC30V 0.2A(1.6A/1コモン)	
	OFF時漏れ電流		100μA以下	
	ON時残留電圧		DC0.5V以下	
	コモン方式		16点/1コモン(2端子)※1	
	動作時間	OFF→ON	1μs以下(負荷:5mA～200mA)	
		ON→OFF	5μs以下(負荷:5mA～200mA)	
内部消費電流			130mA以下	
質量			約190g	

※1 入力のCOM(COM0、COM1)、出力のCOM(COM2、COM3)の4つのCOMはそれぞれ独立です。各COMで二つある端子は内部で短絡しています。

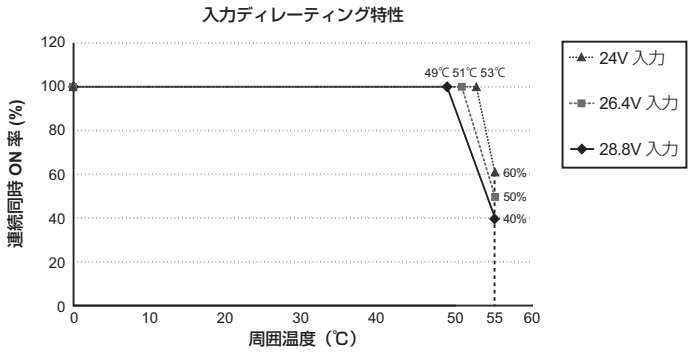
※2 DC5Vモード(ユニットエディタで設定)はIN100～IN115のみ使用できます。

※3 過電圧を検出した場合、電源の切り入りまたはPRG→RUNされるまで共通コモン内のすべての入力(IN100～IN115)がOFFします。

※4 1点でも過電流を検知すると、要因が取り除かれるまで共通コモン内(COM2:OUT200～215、COM3:OUT300～315)のすべての出力が保護動作(出力OFF)と自動復帰を繰り返します。

注 記

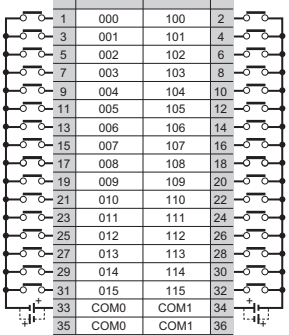
- **同時にONした入力の点数が仕様範囲を超えると、ユニットが故障することがあります。**
- **KV-SIR32XTの周囲温度と連続同時ON率との関係は以下のグラフのようになります。**



結線・内部回路図

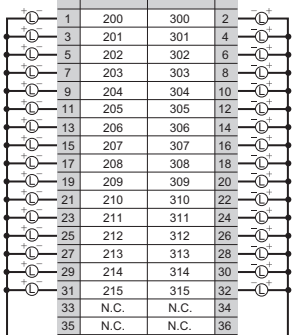
■ KV-SIR32XT入出力配線図

左側コネクタ



ピンNo.	信号名	信号名	ピンNo.
1	000	100	2
3	001	101	4
5	002	102	6
7	003	103	8
9	004	104	10
11	005	105	12
13	006	106	14
15	007	107	16
17	008	108	18
19	009	109	20
21	010	110	22
23	011	111	24
25	012	112	26
27	013	113	28
29	014	114	30
31	015	115	32
33	COM0	COM1	34
35	COM0	COM1	36
37	N.C.	N.C.	38
39	N.C.	N.C.	40

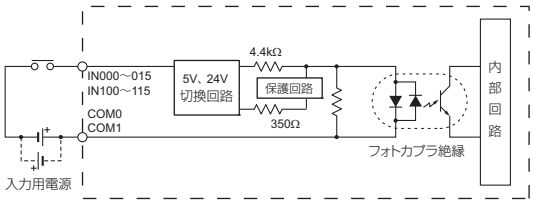
右側コネクタ



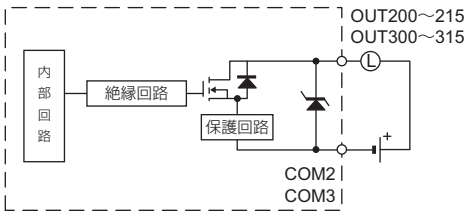
ピンNo.	信号名	信号名	ピンNo.
1	200	300	2
3	201	301	4
5	202	302	6
7	203	303	8
9	204	304	10
11	205	305	12
13	206	306	14
15	207	307	16
17	208	308	18
19	209	309	20
21	210	310	22
23	211	311	24
25	212	312	26
27	213	313	28
29	214	314	30
31	215	315	32
33	N.C.	N.C.	34
35	N.C.	N.C.	36
37	COM2	COM3	38
39	COM2	COM3	40

※ COM0～COM3はそれぞれ独立です。
※ 二つあるCOM端子は内部で短絡しています。

■ KV-SIR32XT内部回路図(入力回路)

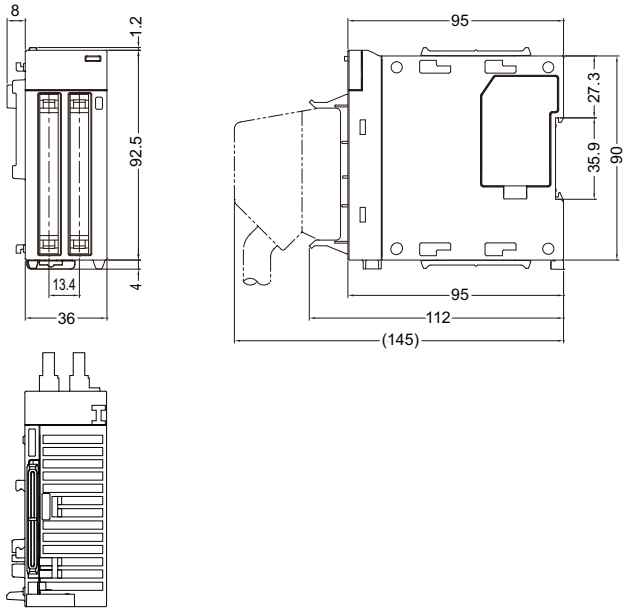


■ KV-SIR32XT内部回路図(出力回路)



外形寸法図

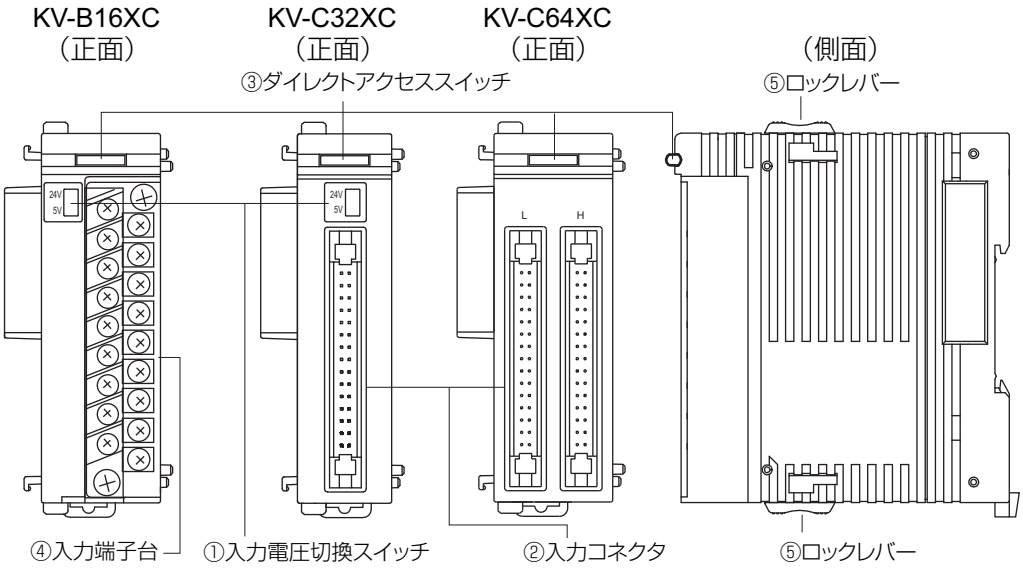
KV-SIR32XT



2-5 拡張入力ユニット

KV-5000/3000シリーズ拡張入力ユニット《KV-B16XC/C32XC/C64XC》各部の名称と機能、および仕様について説明します。

各部の名称と機能



番号	名　称	機　能
①	入力電圧切換スイッチ	入力電圧を切り換えます。5V⇔24V
②	入力コネクタ	入力用コネクタです。
③	ダイレクトアクセススイッチ	スイッチをONすることにより、CPUのアクセスウインドウで拡張ユニットの状態をモニタできます。また、スイッチの色により、拡張ユニットの状態がわかります。 緑点灯：正常動作のとき 緑点滅：緑点灯状態でスイッチをONし、ユニットが選択されたとき 赤点灯：ユニットエラーが発生したとき（一部例外を除く※1） 赤点滅：赤点灯状態でスイッチをONし、ユニットが選択されたとき
④	入力端子台	入力用端子台です。取りはずしが可能です。（M3ビス）
⑤	ロックレバー	ユニットの連結をロックします。

※1　以下の場合、ユニットエラー状態ではなくても赤点灯します。
　①電源投入時、CPUユニットのユニット情報がCPU1台の構成になっているとき
　このとき、アクセスウインドウでエラー表示されます。

※　KV-C32XC/C64XC接続用34ピンコネクタはオプションで用意しています。

参 考　ダイレクト処理が可能です。KV-B16XA/C32XA/C64XA/C64XBも使用可能ですが、ダイレクト処理はできません。

「ダイレクト処理について」4-83ページ

仕様

型　式	KV-B16XC（16点）		KV-C32XC（32点）		KV-C64XC（64点）
外部接続方式	脱着式端子台		コネクタ（MIL規格）※1		コネクタ（MIL規格）※1
入力端子	DC 24Vモード	DC 5Vモード	DC 24Vモード	DC 5Vモード	DC 24Vモード※4
入力最大電圧	DC26.4V				
入力定格電圧	DC 24V 5.3mA	DC 5V 1mA	DC 24V 5.3mA	DC 5V 1mA	DC 24V 4.1mA
最小ON電圧	19V	3.5V	19V	3.5V	19V
最大OFF電流	1.5mA	―	1.5mA	―	1.5mA
最大OFF電圧	―	1.5V	―	1.5V	―
コモン方式	16点/1コモン（2端子）※2		32点/1コモン（2端子）※2		32点/1コモン（2端子）×2※3
入力時定数（4段階切換）					

入力時定数設定	OFF→ON		ON→OFF	
	TYP	MAX	TYP	MAX
25µs	25µs	65µs	75µs	120µs
300µs	275µs	365µs	275µs	420µs
1ms	1ms	1.2ms	1ms	1.2ms
10ms	10ms	12ms	10ms	12ms

入力インピーダンス	4.3kΩ		5.6kΩ
内部消費電流	15mA以下		25mA以下
質量	約120g	約110g	約140g

※1　MIL-C-83503準拠

※2　KV-C32XC, KV-B16XCのCOMは2点ありますが、内部は共通になっています。

※3　KV-C64XCのCOMは4点ありますが、内部ではH側（2点）、L側（2点）で各部共通になっています。

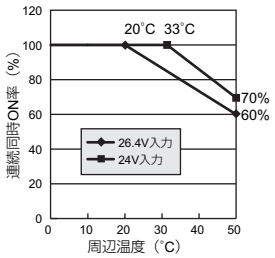
「KV-B16XC内部回路図」2-20ページ　「KV-C32XC内部回路図」2-21ページ
「KV-C64XC内部回路図」2-22ページ

※4　全端子2線式対応（ただし、2線式近接センサEVシリーズは非対応）

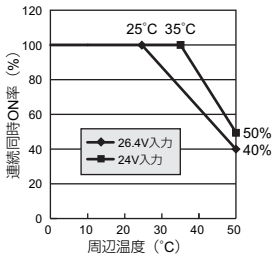
注 記

- **同時にONした入力の点数が仕様の範囲を超えると、ユニットが故障することがあります。**
- **KV-C32XCの2線式対応端子とKV-C64XCにおける周囲温度と連続同時ON率との関係は次のグラフのようになります。**

KV-C32XC　ディレーティング

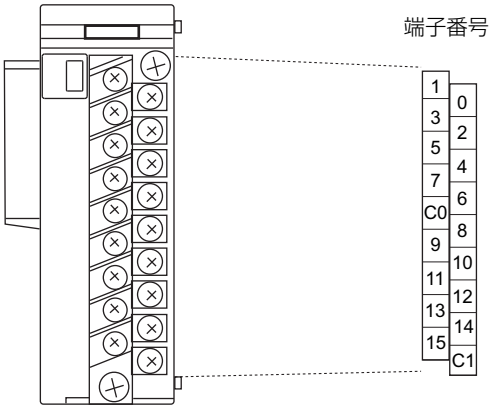


KV-C64XC　ディレーティング

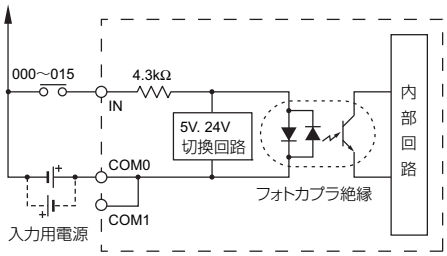


結線・内部回路図

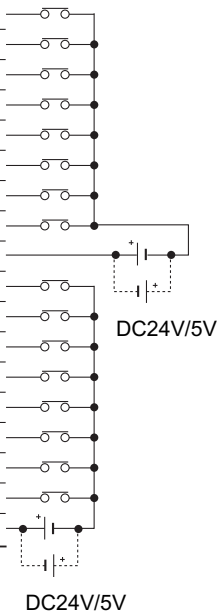
KV-B16XC結線図



KV-B16XC内部回路図

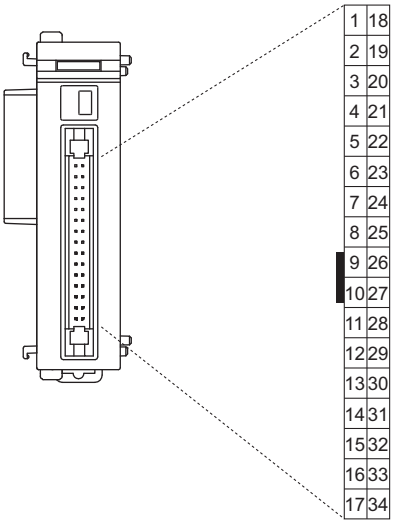


端子番号	方向	信号名
0	入力	000
1	入力	001
2	入力	002
3	入力	003
4	入力	004
5	入力	005
6	入力	006
7	入力	007
C0	入力	COM0
8	入力	008
9	入力	009
10	入力	010
11	入力	011
12	入力	012
13	入力	013
14	入力	014
15	入力	015
C1	入力	COM1

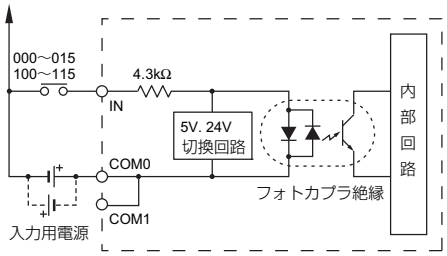


※COMは内部短絡です。

KV-C32XC結線図



KV-C32XC内部回路図



※ KV-5000/3000シリーズの拡張ユニットでは、
ピン番号の並びは、縦に1,2,3と並びます。

ピンNo.	方向	信号名
1	入力	000
2	入力	001
3	入力	002
4	入力	003
5	入力	004
6	入力	005
7	入力	006
8	入力	007
9	入力	008
10	入力	009
11	入力	010
12	入力	011
13	入力	012
14	入力	013
15	入力	014
16	入力	015
17	入力	COM0

DC24V/5V

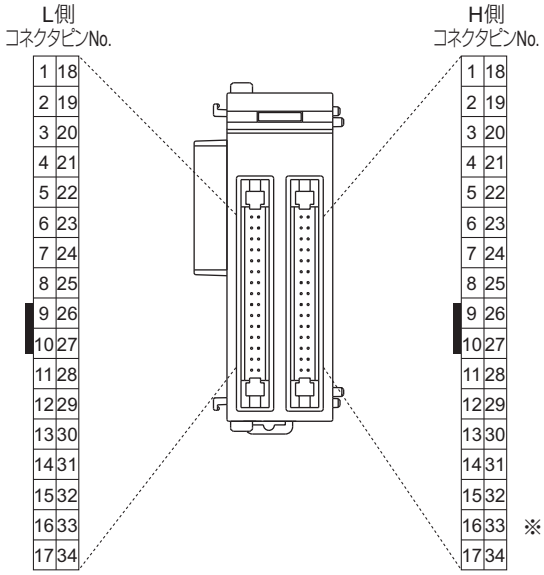
ピンNo.	方向	信号名
18	入力	100
19	入力	101
20	入力	102
21	入力	103
22	入力	104
23	入力	105
24	入力	106
25	入力	107
26	入力	108
27	入力	109
28	入力	110
29	入力	111
30	入力	112
31	入力	113
32	入力	114
33	入力	115
34	入力	COM1

DC24V/5V

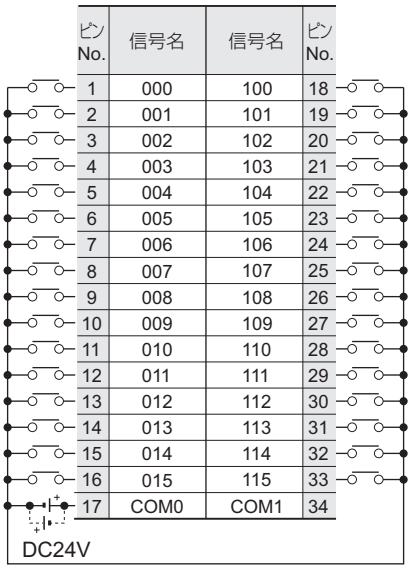
※COMは内部短絡です。

注 記	**入力同時連続ON率が仕様の範囲を超えないようにしてください。超えると破損することがあります。**

KV-C64XC結線図

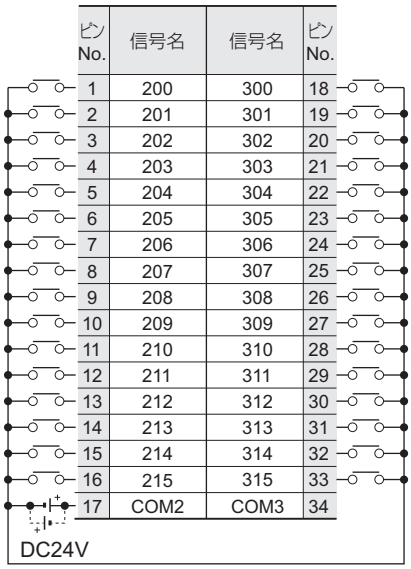


※ KV-5000/3000シリーズの拡張ユニットでは、
ピン番号の並びは、縦に1,2,3と並びます。



ピンNo.	信号名	信号名	ピンNo.
1	000	100	18
2	001	101	19
3	002	102	20
4	003	103	21
5	004	104	22
6	005	105	23
7	006	106	24
8	007	107	25
9	008	108	26
10	009	109	27
11	010	110	28
12	011	111	29
13	012	112	30
14	013	113	31
15	014	114	32
16	015	115	33
17	COM0	COM1	34

DC24V

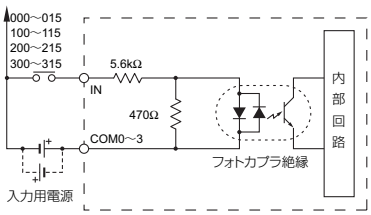


ピンNo.	信号名	信号名	ピンNo.
1	200	300	18
2	201	301	19
3	202	302	20
4	203	303	21
5	204	304	22
6	205	305	23
7	206	306	24
8	207	307	25
9	208	308	26
10	209	309	27
11	210	310	28
12	211	311	29
13	212	312	30
14	213	313	31
15	214	314	32
16	215	315	33
17	COM2	COM3	34

DC24V

※COM0,COM1は内部短絡です。
COM2,COM3は内部短絡です。
(COM0,1)–(COM2,3)は内部短絡ではありません。

KV-C64XC内部回路図



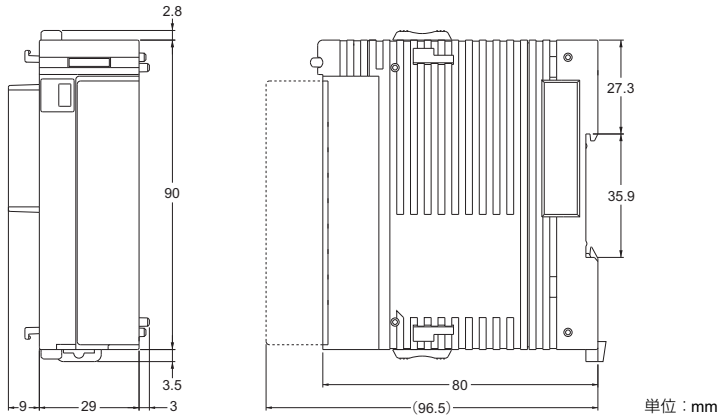
注 記	入力同時連続ON率が仕様の範囲を超えないようにしてください。超えると破損することがあります。

ポイント

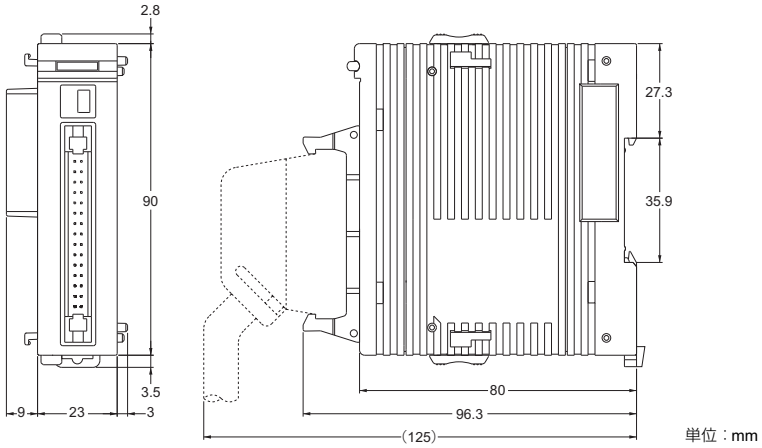
64点ユニットにはOP-42224コネクタ(34ピン)をご使用ください。
OP-23139は使用できません。

外形寸法図

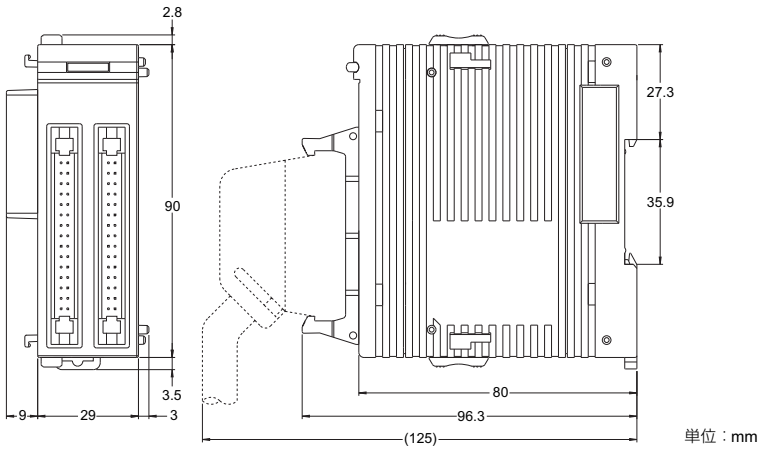
KV-B16XC



KV-C32XC



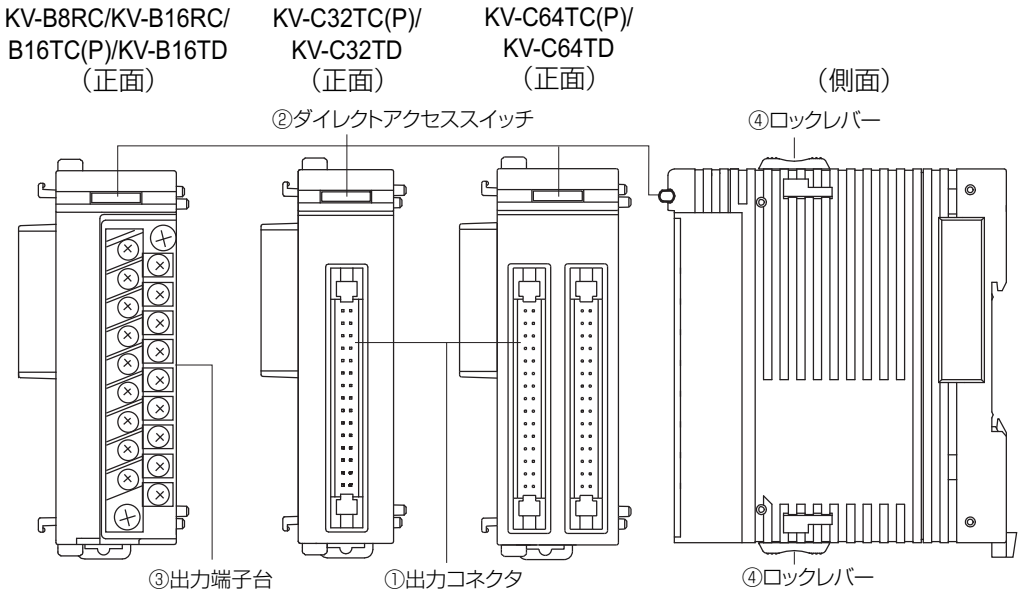
KV-C64XC



2-6 拡張出力ユニット

KV-5000/3000シリーズ拡張出力ユニット《KV-B8RC/KV-B16RC/KV-B16TC(P)/KV-C32TC(P)/KV-C64TC(P)/KV-B16TD/KV-C32TD/KV-C64TD》の各部の名称と機能、および仕様について説明します。

各部の名称と機能



番号	名　称	機　能
①	出力コネクタ	出力用コネクタです。
②	ダイレクトアクセス スイッチ	スイッチをONすることにより、CPUのアクセスウインドウで拡張ユニットの状態をモニタできます。また、スイッチの色により、拡張ユニットの状態がわかります。 緑点灯：正常動作のとき 緑点滅：緑点灯状態でスイッチをONし、ユニットが選択されたとき 赤点灯：ユニットエラーが発生したとき（一部例外を除く※1） 赤点滅：赤点灯状態でスイッチをONし、ユニットが選択されたとき
③	出力端子台	出力用端子台です。取りはずしが可能です。（M3ビス）
④	ロックレバー	ユニットの連結をロックします。

※1　以下の場合、ユニットエラー状態ではなくても赤点灯します。
　①電源投入時、CPUユニットのユニット情報がCPU1台の構成になっているとき
　このとき、アクセスウインドウでエラー表示されます。

※　KV-C32TC/C64TC/C32TCP/C64TCP/KV-C32TD/KV-C64TD接続用34ピンコネクタはオプションで用意しています。

参 考　ダイレクト処理が可能です。KV-B16RA/B16TA/C32TA(P)/C64TA(P)も使用可能ですが、ダイレクト処理はできません。
「ダイレクト処理について」4-83ページ

仕様

型　式		KV-B8RC（8点）	KV-B16RC（16点）	KV-B16TCP（16点）	KV-C32TCP（32点）	KV-C64TCP（64点）
出力形態		リレー		トランジスタ（PNP）		
外部接続方式		脱着式端子台		脱着式端子台	コネクタ※1	
定格負荷		AC 250V/DC 30V　2A（KV-B16RCは8A/1コモン）		DC 30V 0.2A		
OFF時漏れ電流		－		100μA以下		
ON時残留電圧		－		0.5V以下		
ON抵抗		50mΩ以下		－		
コモン方式		独立	8点/1コモン	16点/1コモン（2端子）※2	32点/1コモン（2端子）※2	64点/1コモン（4端子）※3
応答時間	OFF→ON	10ms以下		10μs以下		50μs以下
	ON→OFF			200μs以下		
内部消費電流		65mA以下	120mA以下	30mA以下	55mA以下	100mA以下
質量		約160g	約190g	約130g	約100g	約140g
リレー寿命		電気的10万回以上（20回/分）機械的2000万回以上		－		
リレー交換		不可		－		

※1　MIL-C-83503準拠

※2　KV-B16TCP, C32TCPの2点のCOMは、内部で短絡しています。

※3　KV-C64TCPの4点のCOMは、内部で短絡しています。

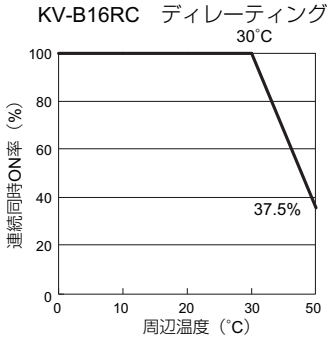
「KV-B8RC内部回路図」、「KV-B16RC内部回路図」、「KV-B16TCP内部回路図」2-29ページ
「KV-C32TCP内部回路図」2-30ページ
「KV-C64TCP内部回路図」2-32ページ

ポイント　**KV-C32TCP/KV-C64TCPは、1.6A/1コモンです。**

注　記

同時にONした出力の点数が仕様の範囲を超えると、ユニットが故障することがあります。
周囲温度と連続同時ON率は右のグラフのようになります。

KV-B16RC　ディレーティング



型　式		KV-B16TC KV-B16TD （16点）	KV-C32TC KV-C32TD （32点）	KV-C64TC KV-C64TD （64点）
出力形態		トランジスタ（NPN）（Cタイプ） MOSFET(N-ch)（過電流保護機能あり）（Dタイプ）		
外部接続方式		脱着式端子台	コネクタ※1	
定格負荷		DC 30V 0.2A（Cタイプ） DC 30V 0.3A（Dタイプ）	DC 30V 0.2A※4	
OFF時漏れ電流		100μA以下		
ON時残留電圧		0.5V以下		
コモン方式		16点/1コモン（2端子）※2	32点/1コモン（2端子）※2	64点/1コモン（4端子）※3
応答時間	OFF→ON	10μs以下（Cタイプ） 100μs以下（Dタイプ）		50μs以下（Cタイプ） 150μs以下（Dタイプ）
	ON→OFF	200μs以下（Cタイプ） 300μs以下（Dタイプ）		
内部消費電流		35mA以下（Cタイプ） 45mA以下（Dタイプ）	55mA以下（Cタイプ） 65mA以下（Dタイプ）	100mA以下（Cタイプ） 120mA以下（Dタイプ）
質量		約130g	約100g	約140g

※1　MIL-C-83503準拠

※2　KV-B16TC, B16TD, C32TC, C32TDの2点のCOMは、内部で短絡しています。

※3　KV-C64TC, C64TDの4点のCOMは、内部で短絡しています。

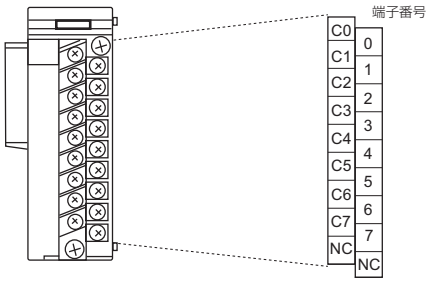
※4　コモン1端子あたりの定格負荷は1.6Aです。

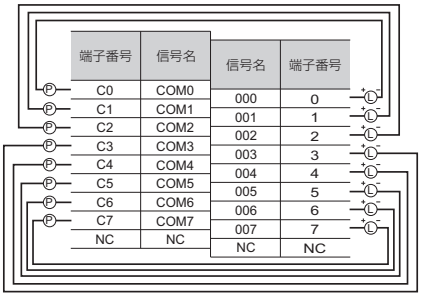
「KV-B16TC/B16TD内部回路図」2-29ページ
「KV-C32TC/C32TD内部回路図」2-30ページ
「KV-C64TC/C64TD内部回路図」2-31ページ

ポイント　**KV-C32T*/C64T*は、1.6A/1コモンです。**

結線・内部回路図

KV-B8RC結線図

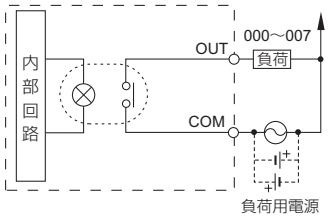




端子番号	信号名	信号名	端子番号
C0	COM0	000	0
C1	COM1	001	1
C2	COM2	002	2
C3	COM3	003	3
C4	COM4	004	4
C5	COM5	005	5
C6	COM6	006	6
C7	COM7	007	7
NC	NC	NC	NC

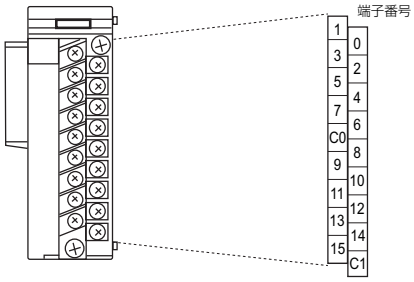
Ⓟ：

KV-B8RC内部回路図



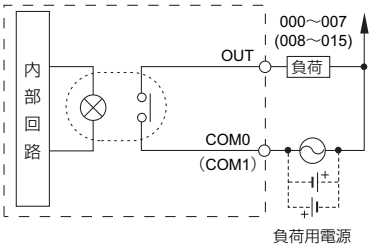
※COMは独立しています。

KV-B16RC/B16TC/B16TD/B16TCP結線図

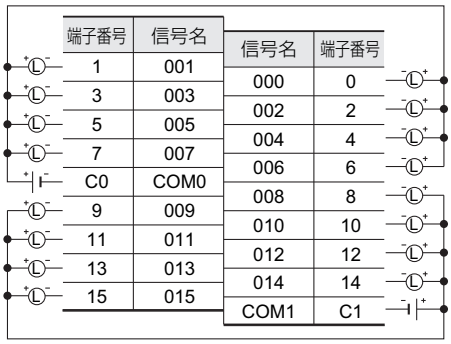


端子番号	信号名	信号名	端子番号
1	001	000	0
3	003	002	2
5	005	004	4
7	007	006	6
C0	COM	008	8
9	009	010	10
11	011	012	12
13	013	014	14
15	015	COM	C1

KV-B16RC内部回路図



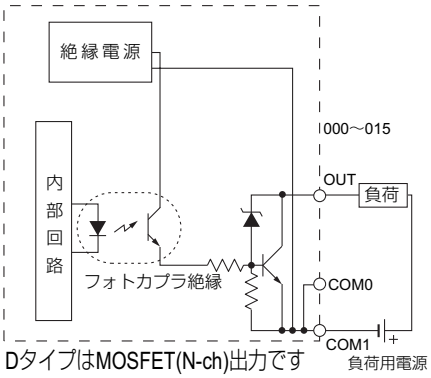
※COMは内部で短絡していません。



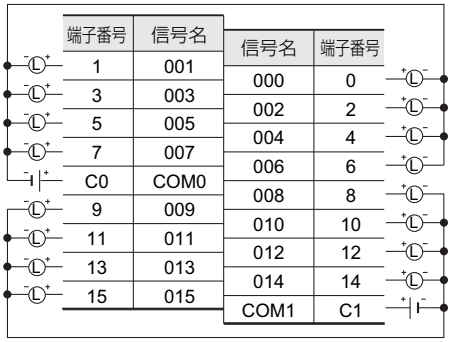
端子番号	信号名	信号名	端子番号
1	001	000	0
3	003	002	2
5	005	004	4
7	007	006	6
C0	COM0	008	8
9	009	010	10
11	011	012	12
13	013	014	14
15	015	COM1	C1

※COMは内部で短絡しています。

KV-B16TC/B16TD内部回路図



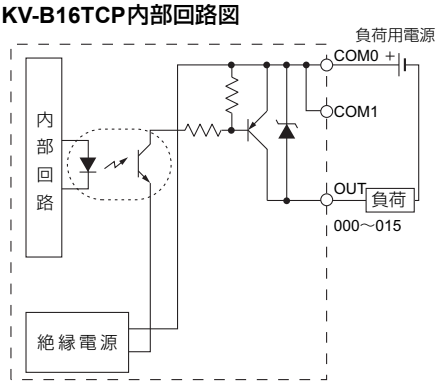
DタイプはMOSFET(N-ch)出力です



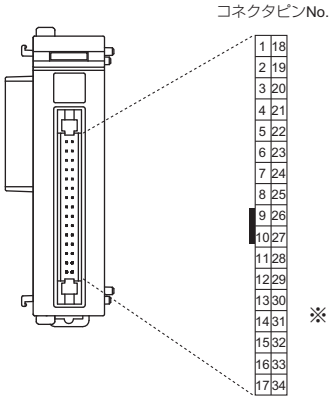
端子番号	信号名	信号名	端子番号
1	001	000	0
3	003	002	2
5	005	004	4
7	007	006	6
C0	COM0	008	8
9	009	010	10
11	011	012	12
13	013	014	14
15	015	COM1	C1

※COMは内部で短絡しています。

KV-B16TCP内部回路図



KV-C32TC/C32TD/C32TCP結線図

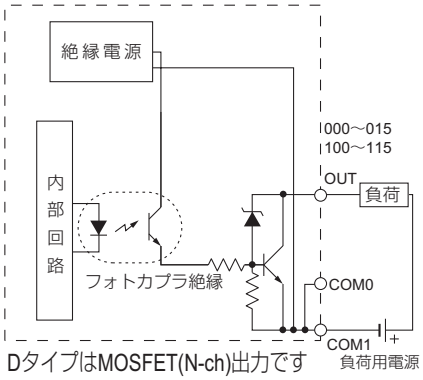


※ KV-5000/3000シリーズの拡張ユニットでは、
ピン番号の並びは、縦に1,2,3と並びます。

ピンNo.	信号名	信号名	ピンNo.
1	000	100	18
2	001	101	19
3	002	102	20
4	003	103	21
5	004	104	22
6	005	105	23
7	006	106	24
8	007	107	25
9	008	108	26
10	009	109	27
11	010	110	28
12	011	111	29
13	012	112	30
14	013	113	31
15	014	114	32
16	015	115	33
17	COM0	COM1	34

DC　※COMは内部で短絡しています。　DC

KV-C32TC/C32TD内部回路図

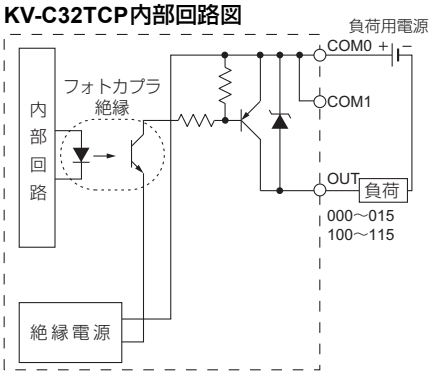


DタイプはMOSFET(N-ch)出力です

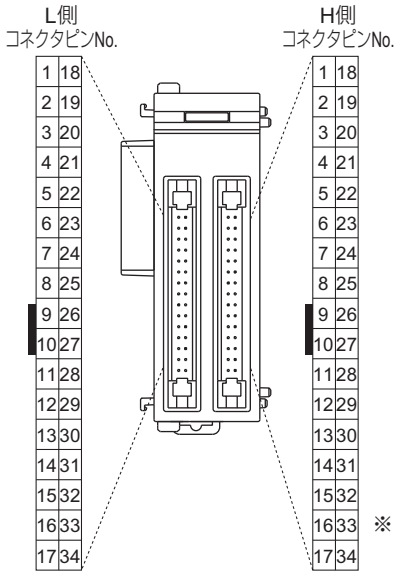
ピンNo.	信号名	信号名	ピンNo.
1	000	100	18
2	001	101	19
3	002	102	20
4	003	103	21
5	004	104	22
6	005	105	23
7	006	106	24
8	007	107	25
9	008	108	26
10	009	109	27
11	010	110	28
12	011	111	29
13	012	112	30
14	013	113	31
15	014	114	32
16	015	115	33
17	COM0	COM1	34

DC　※COMは内部で短絡しています。　DC

KV-C32TCP内部回路図

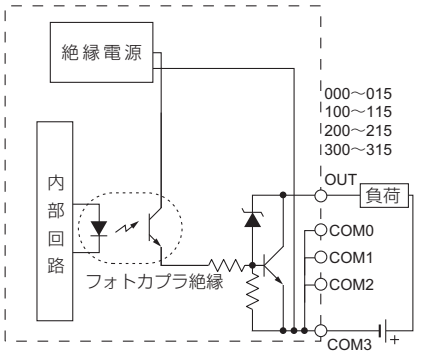


KV-C64TC/C64TD結線図

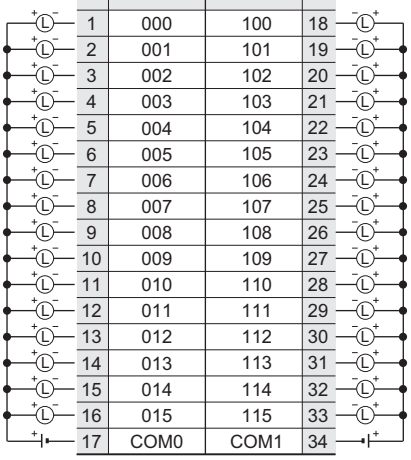


※ KV-5000/3000シリーズの拡張ユニットでは、
ピン番号の並びは、縦に1,2,3と並びます。

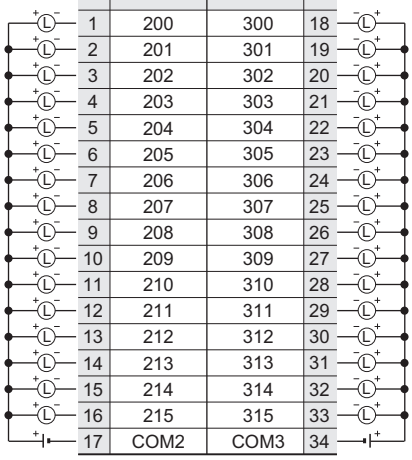
KV-C64TC/C64TD内部回路図



DタイプはMOSFET(N-ch)出力です



ピンNo.	信号名	信号名	ピンNo.
1	000	100	18
2	001	101	19
3	002	102	20
4	003	103	21
5	004	104	22
6	005	105	23
7	006	106	24
8	007	107	25
9	008	108	26
10	009	109	27
11	010	110	28
12	011	111	29
13	012	112	30
14	013	113	31
15	014	114	32
16	015	115	33
17	COM0	COM1	34



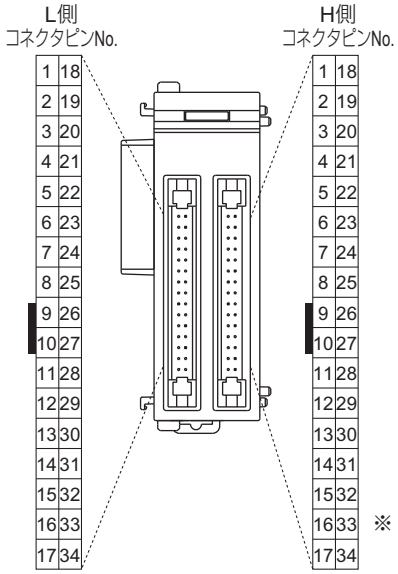
ピンNo.	信号名	信号名	ピンNo.
1	200	300	18
2	201	301	19
3	202	302	20
4	203	303	21
5	204	304	22
6	205	305	23
7	206	306	24
8	207	307	25
9	208	308	26
10	209	309	27
11	210	310	28
12	211	311	29
13	212	312	30
14	213	313	31
15	214	314	32
16	215	315	33
17	COM2	COM3	34

※COMは内部で短絡しています。

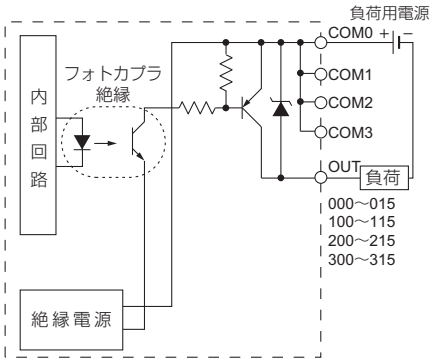
ポイント

64点ユニットにはOP-42224コネクタ（34ピン）をご使用ください。
OP-23139は使用できません。

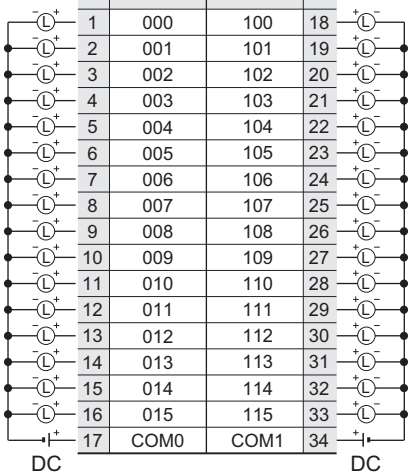
KV-C64TCP結線図



KV-C64TCP内部回路図

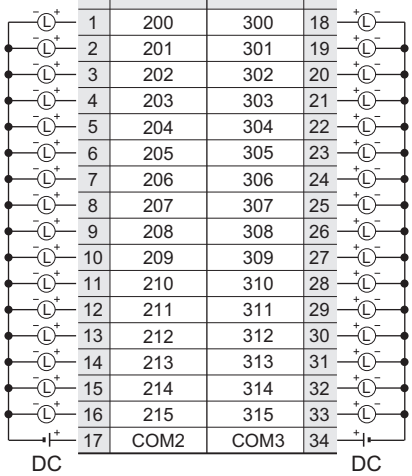


※ KV-5000/3000シリーズの拡張ユニットでは、
ピン番号の並びは、縦に1,2,3と並びます。



ピンNo.	信号名	信号名	ピンNo.
1	000	100	18
2	001	101	19
3	002	102	20
4	003	103	21
5	004	104	22
6	005	105	23
7	006	106	24
8	007	107	25
9	008	108	26
10	009	109	27
11	010	110	28
12	011	111	29
13	012	112	30
14	013	113	31
15	014	114	32
16	015	115	33
17	COM0	COM1	34

DC　　DC



ピンNo.	信号名	信号名	ピンNo.
1	200	300	18
2	201	301	19
3	202	302	20
4	203	303	21
5	204	304	22
6	205	305	23
7	206	306	24
8	207	307	25
9	208	308	26
10	209	309	27
11	210	310	28
12	211	311	29
13	212	312	30
14	213	313	31
15	214	314	32
16	215	315	33
17	COM2	COM3	34

DC　　DC

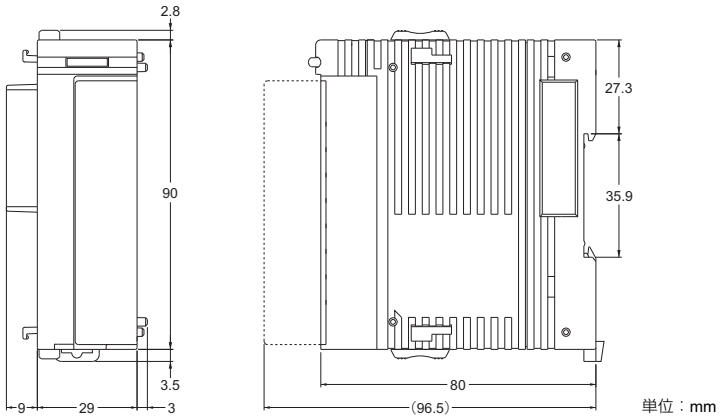
※COMは内部で短絡しています。

ポイント

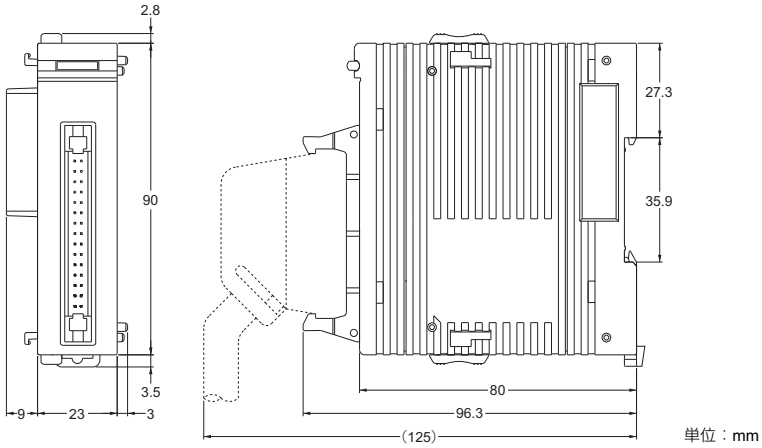
64点ユニットにはOP-42224コネクタ（34ピン）をご使用ください。
OP-23139は使用できません。

外形寸法図

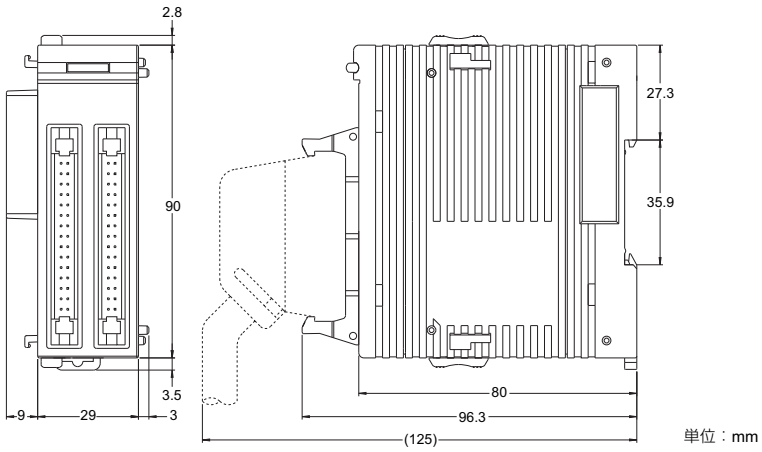
KV-B8RC/B16RC/B16TC/B16TD/B16TCP



KV-C32TC/C32TD/C32TCP



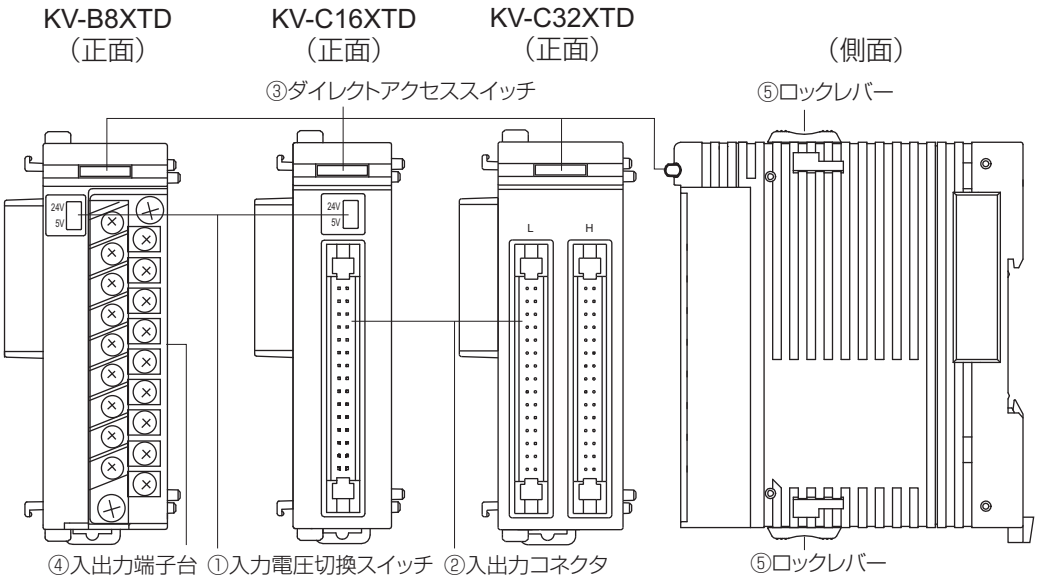
KV-C64TC/C64TD/C64TCP



2-7 拡張入出力ユニット

KV-5000/3000シリーズ拡張入力ユニット《KV-B8XTD/C16XTD/C32XTD》の各部の名称と機能、および仕様について説明します。

各部の名称と機能



番号	名　称	機　能
①	入力電圧切換スイッチ	入力電圧を切り換えます。5V⇔24V
②	入出力コネクタ	入出力用コネクタです。
③	ダイレクトアクセス スイッチ	スイッチをONすることにより、CPUのアクセスウインドウで拡張ユニットの状態をモニタできます。また、スイッチの色により、拡張ユニットの状態がわかります。 緑点灯：正常動作のとき 緑点滅：緑点灯状態でスイッチをONし、ユニットが選択されたとき 赤点灯：ユニットエラーが発生したとき（一部例外を除く※1） 赤点滅：赤点灯状態でスイッチをONし、ユニットが選択されたとき
④	入出力端子台	入出力用端子台です。取りはずしが可能です。（M3ビス）
⑤	ロックレバー	ユニットの連結をロックします。

※1 以下の場合、ユニットエラー状態ではなくても赤点灯します。
　①電源投入時、CPUユニットのユニット情報がCPU1台の構成になっているとき
　このとき、アクセスウインドウでエラー表示されます。

※ KV-C16XTD/C32XTD接続用34ピンコネクタはオプションで用意しています。

仕様

型　式			KV-B8XTD（8＋8点）		KV-C16XTD（16＋16点）		KV-C32XTD（32＋32点）
外部接続方式			脱着式端子台		コネクタ（MIL規格）※1		
入力	入力点数		8点		16点		32点
	入力モード		DC24Vモード	DC5Vモード	DC24Vモード	DC5Vモード	DC24Vモード※3
	入力最大電圧		DC26.4V				
	入力定格電圧		DC24V/5.3mA	DC5V/1mA	DC24V/5.3mA	DC5V/1mA	DC24V/4.1mA
	最小ON電圧		19V	3.5V	19V	3.5V	19V
	最大OFF電流		1.5mA	－	1.5mA	－	1.5mA
	最大OFF電圧		－	1.5V	－	1.5V	－
	コモン方式		8点/1コモン（1端子）※4		16点/1コモン（1端子）※4		32点/1コモン（2端子）※2
	入力時定数						

入力時定数設定	OFF→ON		ON→OFF	
	TYP	MAX	TYP	MAX
25μs	25μs	65μs	75μs	120μs
300μs	275μs	365μs	275μs	420μs
1ms	1ms	1.2ms	1ms	1.2ms
10ms	10ms	12ms	10ms	12ms

型　式			KV-B8XTD（8＋8点）		KV-C16XTD（16＋16点）		KV-C32XTD（32＋32点）
入力	入力インピーダンス		4.3kΩ				5.6kΩ
出力	出力点数		8点		16点		32点
	出力形態		MOSFET(N-ch)（過電流保護機能あり）				
	定格負荷		DC30V 0.3A		DC30V 0.2A		
	OFF時漏れ電流		100μA以下				
	ON時残留電圧		0.5V以下				
	コモン方式		8点/1コモン（1端子）※4		16点/1コモン（1端子）※4		32点/1コモン（2端子）※2
	動作時間	OFF→ON	100μs以下				150μs以下
		ON→OFF	300μs以下				
内部消費電流			30mA以下		40mA以下		65mA以下
質量			約130g		約110g		約130g

※1　MIL-C-83503準拠

※2　KV-C32XTDの入力用COM、出力用COMはそれぞれ2点ありますが、内部は共通になっています。入力と出力のCOMはそれぞれ独立です。

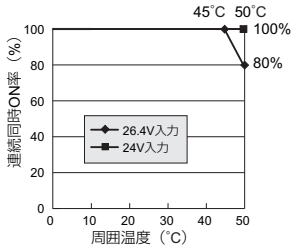
※3　全端子2線式対応（ただし、2線式近接センサEVシリーズは非対応）

※4　入力と出力のCOMはそれぞれ独立です。

「KV-B8XTD内部回路図」2-36ページ　「KV-C16XTD 内部回路図」2-37ページ
「KV-C32XTD 内部回路図」2-38ページ

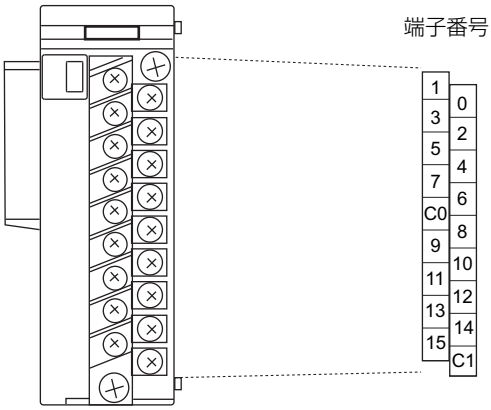
注 記	• 同時に ON した入力の点数が仕様範囲を超えると、ユニットが故障することがあります。 • KV-C32XTDの周囲温度と連続同時ON率との関係は以下のグラフのようになります。

KV-C32XTD　ディレーティング



結線・内部回路図

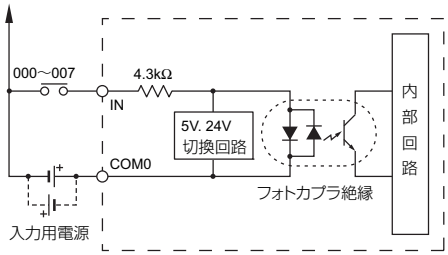
KV-B8XTD結線図



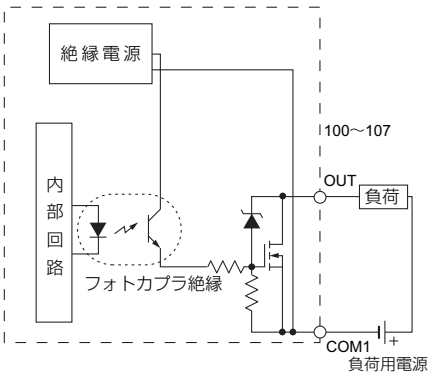
端子番号	方向	信号名	信号名	方向	端子番号
1	入力	001	000	入力	0
3	入力	003	002	入力	2
5	入力	005	004	入力	4
7	入力	007	006	入力	6
C0	入力	COM0	100	出力	8
9	出力	101	102	出力	10
11	出力	103	104	出力	12
13	出力	105	106	出力	14
15	出力	107	COM1	出力	C1

※COMは独立しています。

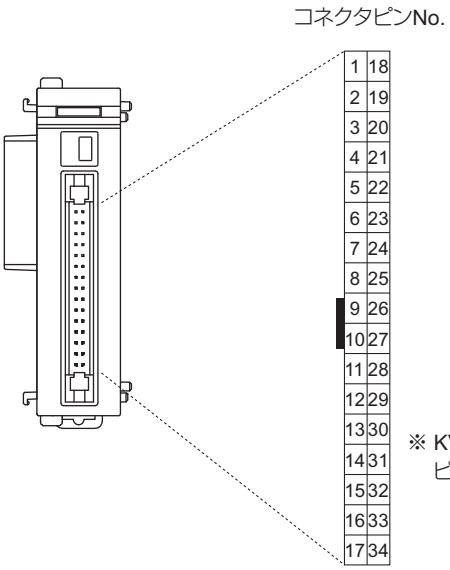
KV-B8XTD内部回路図（入力）



KV-B8XTD内部回路図（出力）

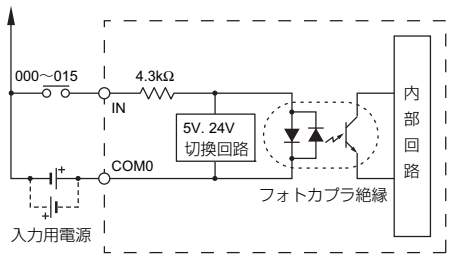


KV-C16XTD 結線図

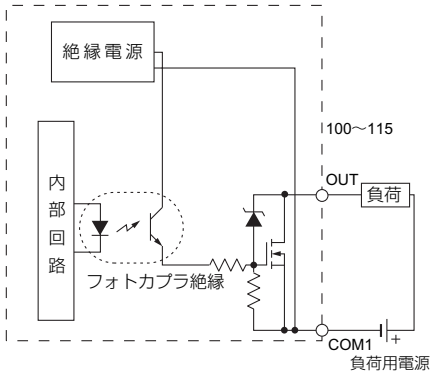


※ KV-5000/3000シリーズの拡張ユニットでは、
ピン番号の並びは、縦に1,2,3と並びます。

KV-C16XTD 内部回路図（入力）



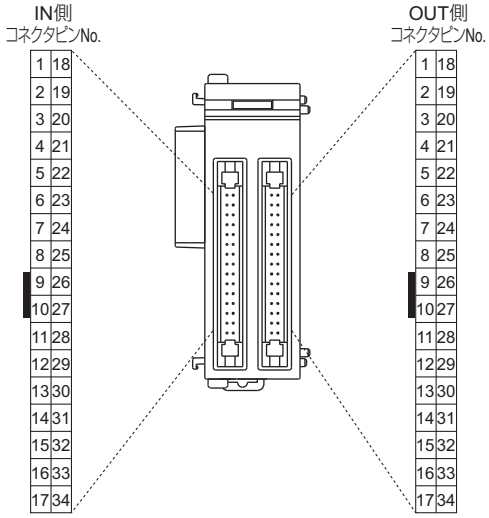
KV-C16XTD 内部回路図（出力）



ピンNo.	方向	信号名	信号名	方向	ピンNo.
1	入力	000	100	出力	18
2	入力	001	101	出力	19
3	入力	002	102	出力	20
4	入力	003	103	出力	21
5	入力	004	104	出力	22
6	入力	005	105	出力	23
7	入力	006	106	出力	24
8	入力	007	107	出力	25
9	入力	008	108	出力	26
10	入力	009	109	出力	27
11	入力	010	110	出力	28
12	入力	011	111	出力	29
13	入力	012	112	出力	30
14	入力	013	113	出力	31
15	入力	014	114	出力	32
16	入力	015	115	出力	33
17	入力	COM0	COM1	出力	34

※COMは独立しています。

KV-C32XTD 結線図



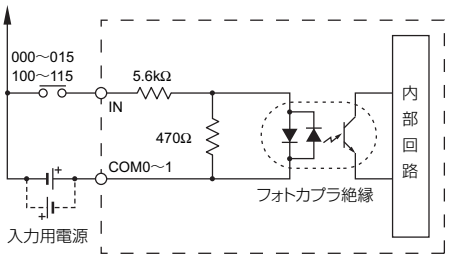
※ KV-5000/3000シリーズの拡張ユニットでは、
ピン番号の並びは、縦に1,2,3と並びます。

ピンNo.	方向	信号名	信号名	方向	ピンNo.
1	入力	000	100	入力	18
2	入力	001	101	入力	19
3	入力	002	102	入力	20
4	入力	003	103	入力	21
5	入力	004	104	入力	22
6	入力	005	105	入力	23
7	入力	006	106	入力	24
8	入力	007	107	入力	25
9	入力	008	108	入力	26
10	入力	009	109	入力	27
11	入力	010	110	入力	28
12	入力	011	111	入力	29
13	入力	012	112	入力	30
14	入力	013	113	入力	31
15	入力	014	114	入力	32
16	入力	015	115	入力	33
17	入力	COM0	COM1	入力	34

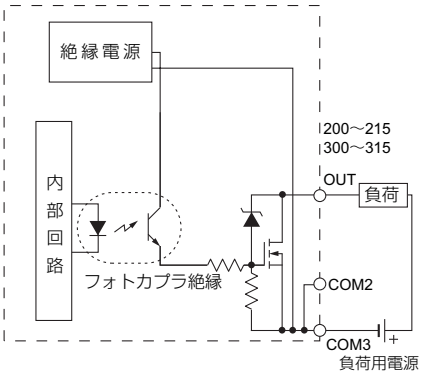
ピンNo.	方向	信号名	信号名	方向	ピンNo.
1	出力	200	300	出力	18
2	出力	201	301	出力	19
3	出力	202	302	出力	20
4	出力	203	303	出力	21
5	出力	204	304	出力	22
6	出力	205	305	出力	23
7	出力	206	306	出力	24
8	出力	207	307	出力	25
9	出力	208	308	出力	26
10	出力	209	309	出力	27
11	出力	210	310	出力	28
12	出力	211	311	出力	29
13	出力	212	312	出力	30
14	出力	213	313	出力	31
15	出力	214	314	出力	32
16	出力	215	315	出力	33
17	出力	COM2	COM3	出力	34

※COM0,COM1は内部短絡です。
COM2,COM3は内部短絡です。
(COM0,1)–(COM2,3)は内部短絡ではありません。

KV-C32XTD 内部回路図（入力）



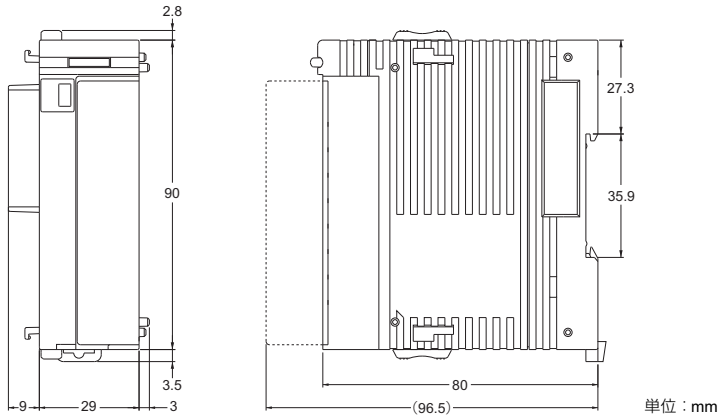
KV-C32XTD 内部回路図（出力）



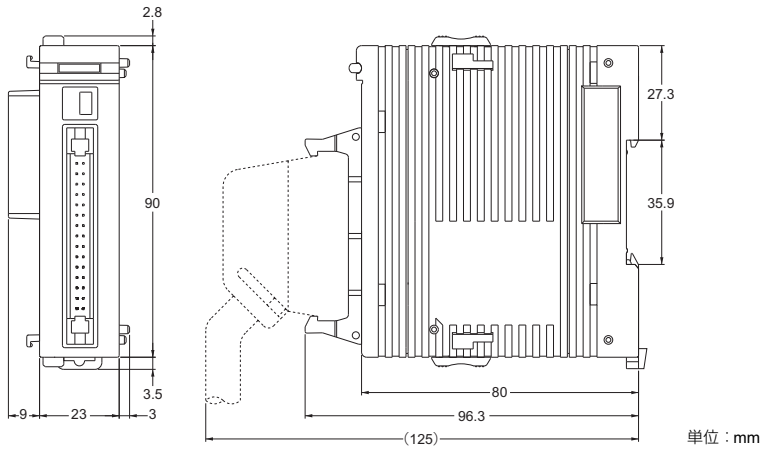
ポイント
32点入出力ユニットにはOP-42224コネクタ（34ピン）をご使用ください。
OP-23139は使用できません。

外形寸法図

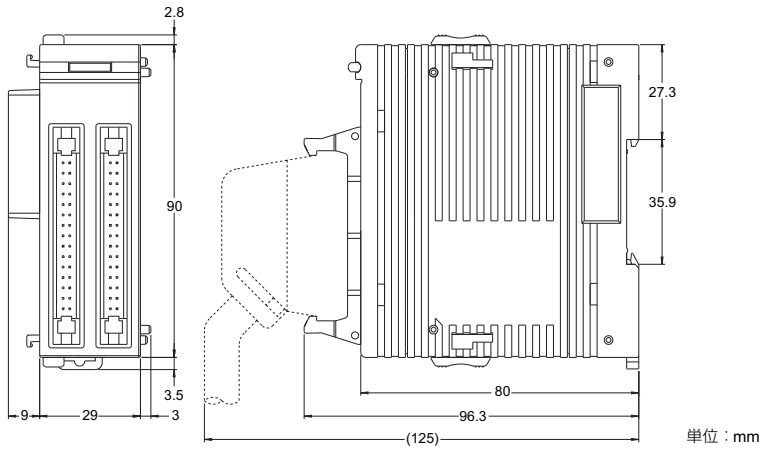
KV-B8XTD



KV-C16XTD



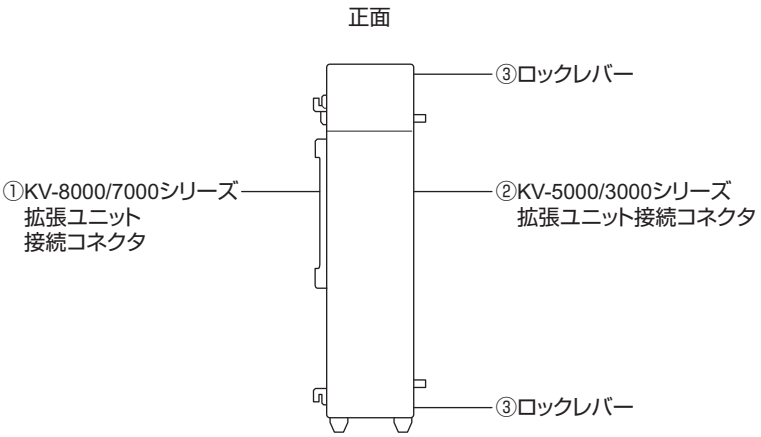
KV-C32XTD



2-8 バス接続ユニット

バス接続ユニット(KV-7000C)の各部の名称と機能について説明します。

各部の名称と機能

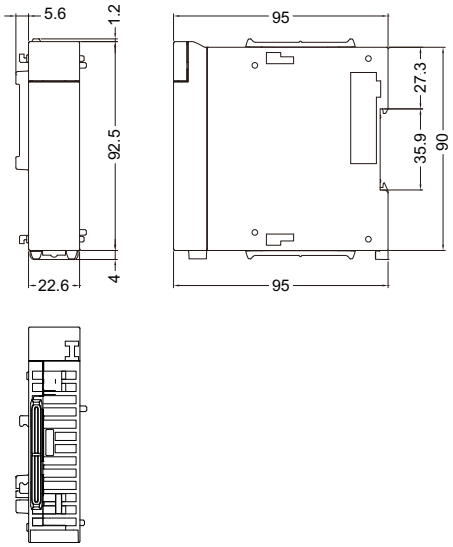


番号	名称	機能
①	KV-8000/7000シリーズ 拡張ユニット接続コネクタ	CPUユニットまたは拡張ユニットとの接続に使用します。
②	KV-5000/3000シリーズ 拡張ユニット接続コネクタ	KV-5000/3000シリーズ拡張ユニットとの接続に使用します。
③	ロックレバー	ユニットの連結をロックします。

ポイント

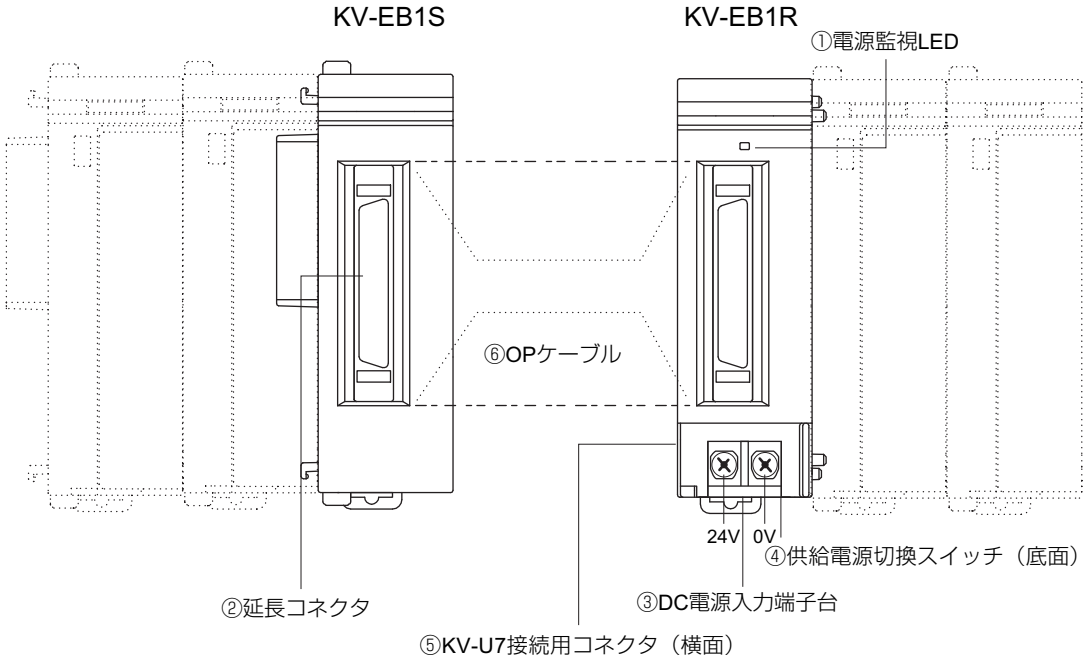
バス接続ユニットを使用する場合、システムの一番右端にエンドユニットを取り付ける必要があります。

外形寸法図



《KV-EB1S/KV-EB1R エクステンションユニット》の各部の名称と機能、および仕様について説明します。

各部の名称と機能



番号	名　称	機　能
①	電源監視LED	電源監視用のLEDです。※1
②	延長コネクタ	延長ケーブル接続用のコネクタです。
③	DC電源入力端子台	DC24V電源入力用の端子台です。(3.5Aまで)※2、※3
④	供給電源切換スイッチ	拡張ユニットへの供給電源の切り換え用スイッチです。 INT : 前段での電源を使用する場合。 EXT : KV-EB1Rに接続するDC電源またはKV-U7を使用する場合。
⑤	KV-U7接続用コネクタ	KV-U7電源ユニット接続用のコネクタです。※3、※4
⑥	延長ケーブル	KV-EB1SとKV-EB1Rとを接続します。 (OP-42142 1m) (OP-42141 2m)

※1　LEDが橙または赤点灯の場合、電源電圧が低下している可能性があります。電源を確認してください。

※2　電源の逆接続には、十分注意してください。逆接保護回路が入っていますが、電源の接続によっては、保護が働かない場合があります。
端子台から供給する電流は3.5A以下になるよう、拡張ユニットの消費電流に注意してください。

※3　KV-8000側の電源容量に制限がある場合、こちらから拡張側ユニットへ電源を供給します。

※4　KV-U7を接続する場合、拡張するユニットの消費電流に注意してください。

ポイント

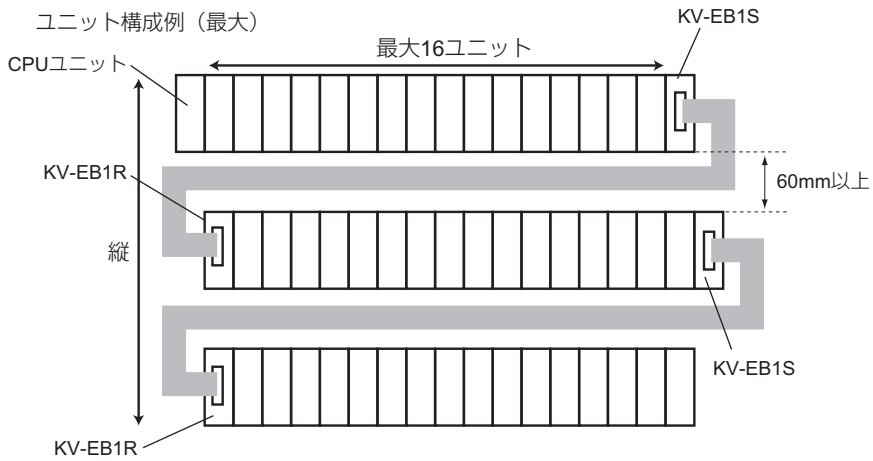
供給電源切換スイッチを切り換えるときは、必ず電源をOFFにしてください。

仕　様

型　式	KV-EB1S	KV-EB1R
電源電圧	DC24V±10%	
内部消費電流	15mA以下	25mA以下
質量	約90g	約115g

接続 ・ 設置について

KV-EB1Sは延長する前の段の一番右端に接続します。
KV-EB1Rは延長した後の段の一番左側に接続します。



■ 接続ケーブルと段数について

- 延長ケーブル1m（OP-42142）使用時
  縦段数3段まで（最大48ユニット接続可）
- 延長ケーブル2m（OP-42141）使用時
  縦段数2段まで（最大32ユニット接続可）

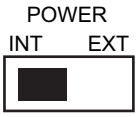
■ 設置について

エクステンションユニット使用時、上段と下段のユニットの間隔は、熱の影響を考慮して60mm以上離してください。
盤内の温度（ユニット下面での温度）が、ユニットの使用周囲温度を超える場合は、強制空冷や周囲との間隔を広げ通風をよくするなどの方法で温度を下げてください。

供給電源切換について

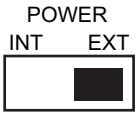
拡張ユニットへ供給する電源を切り換えることができます。

■ 前段の電源を使用する



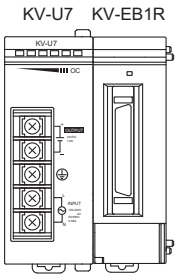
OPケーブルで送られる前段からの電源を使用する場合は、供給電源切換スイッチを「INT」に設定します。
KV-U7または別電源を使用せずに運転可能ですが、合計の消費電流が使用する電源の定格を越えないように注意してください。

■ 外部電源を使用する



KV-U7または別電源を使用する場合は、供給電源切換スイッチを「EXT」に設定します。

● KV-U7を使用する



KV-U7をKV-EB1Rの左側に接続します。

● 別電源を使用する

別電源をDC電源端子台に接続します。

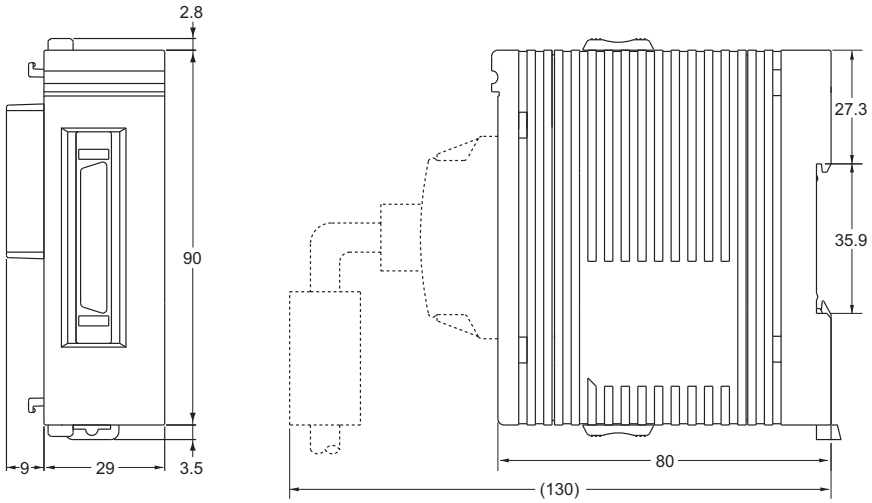
注 記	**KV-EB1Rから別電源を供給する場合は、以下のことに注意してください。** **• 電源投入時：KV-EB1R側から先に電源を供給してください。** **• 電源断　　：CPUユニット側から先に電源断してください。** **（エラー発生時はアクセスウィンドウ、または《KV STUDIO》でエラークリア後、PROG→RUNすれば動作します。）**

ポイント

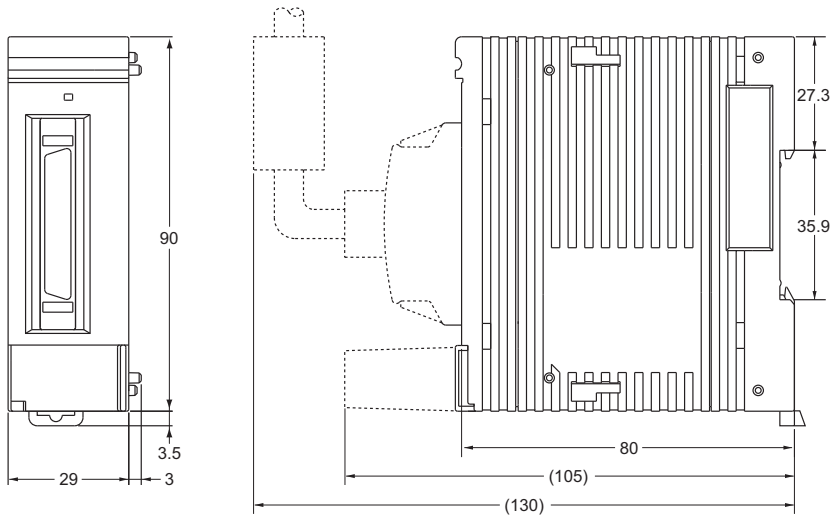
- **KV-EB1 を使用し 3 段増設して各電源に別電源使用の場合、電源ケーブルにフェライトコア（北川工業：RFC-10）を 2 ターン挿入する、KV-U7 使用の場合、FG を接地するなどのノイズ対策をおこなってください。**

外形寸法図

KV-EB1S



KV-EB1R



単位：mm

MEMO

3

ユニットの設置および保守

KV-8000シリーズの設置方法、取り付け、配線時の注意事項や、保守時の手順を説明します。

3-1 ユニットの接続 3-2
3-2 ユニットの設置 3-6
3-3 配線時の注意 3-8
3-4 保守 3-20

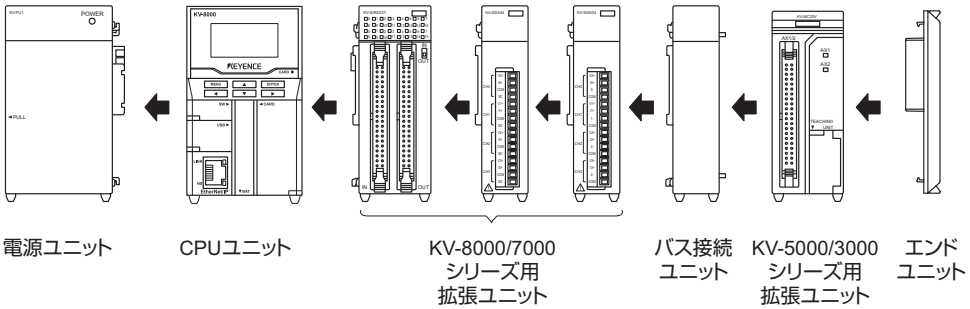
3-1 ユニットの接続

各ユニットの組み立て方法や、DINレールなどへの取り付け方法について説明しています。

ユニット接続時の注意

各ユニットを接続するときは、必ず次のことを守ってください。

- 各ユニットは、必ず電源がOFFの状態で接続してください。



ポイント

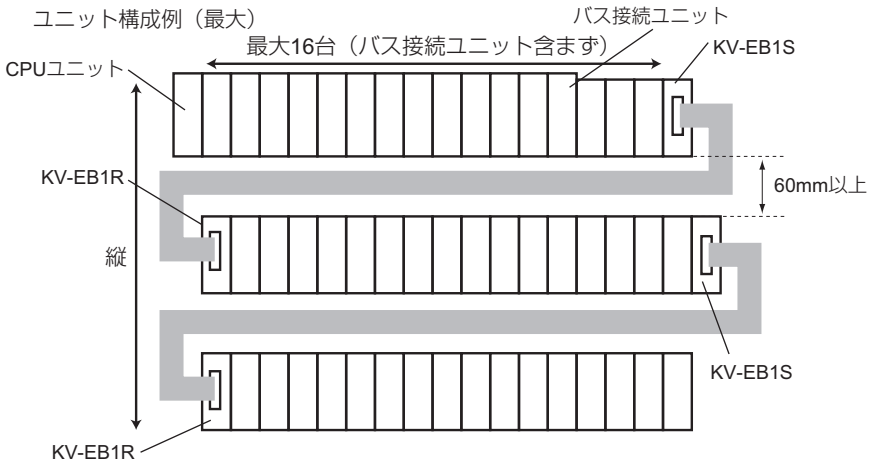
- **CPUユニットの時計データはバックアップ用コンデンサで保持しています。長期間通電しなかった場合、時計データが消失している可能性があります（消失した場合、CPUエラー85「時計データ消失」が発生します）。時計データの保持が必要な場合は、オプションのバッテリ（KV-B1）を取り付けて使用してください。**
- **バス接続ユニットを使用する場合、右端にエンドユニットが必要です。エンドユニットがない場合、エンドユニットなしエラーが発生します。**
- **バス接続ユニットを使用しない場合、右端のユニットには拡張ユニット連結コネクタカバーを取り付けて使用してください。**

■ 最大接続ユニット台数

1台のCPUユニットには、KV-8000/7000シリーズ用拡張ユニット、KV-5000/3000シリーズ用拡張ユニットの混在で最大48台のユニットが接続できます（電源ユニット、バス接続ユニットは除く）。

- CPUユニットには最大16台までのKV-8000/7000シリーズ用拡張ユニットが接続できます
- バス接続ユニットを使用することでKV-5000/3000シリーズ用拡張ユニットが接続できます
- CPUユニット横の接続台数が16台を超える場合は、エクステンションユニットを使用することで、各段16台、合計48台のユニットが接続できます（KV-8000/7000シリーズ用拡張ユニットは、1段目にのみ接続できます）

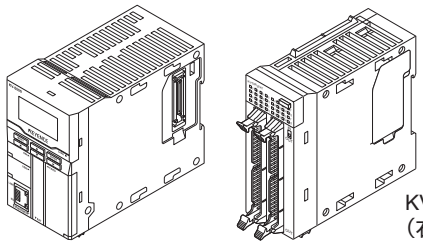
拡張ユニットには一部48台接続できない機種もあります。
各ユニットのマニュアルを参照してください。



ユニットの組み立て

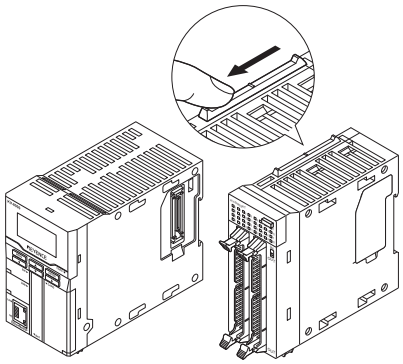
■ KV-8000/7000シリーズ拡張ユニットを接続する場合

ここでは、CPUユニットとKV-SIR32XTを接続する場合を例に説明します。



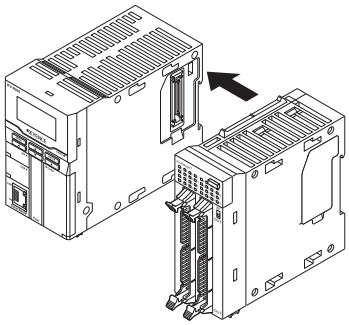
1 右側のユニットの上下に付いているロックレバーをはずします

- くぼみに指をかけ、ユニットの前面方向にスライドさせるとはずれます。



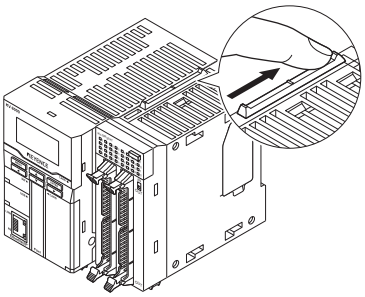
2 左側のユニットのロックレバースロットに、右側のユニットのロックレバーを挿し込んで接続します

- 2つのユニットが平行になるようにして持ち、コネクタの位置を合わせると簡単に接続できます。



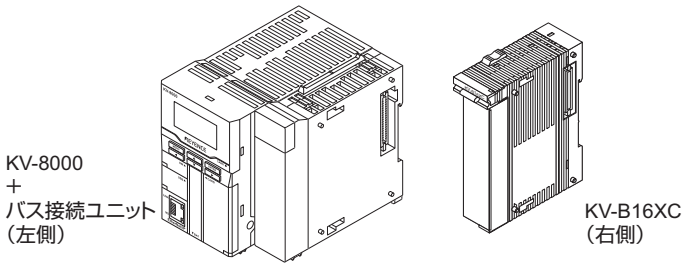
3 右側のユニットの上下に付いているロックレバーをロックします

- 手順1と同じ要領で、くぼみに指をかけ、ユニットの背面方向にスライドさせます。このときユニットとユニットの間にすき間がないかどうか確認します。すき間があれば、正しく接続されていません。



■ KV-5000/3000シリーズ拡張ユニットを接続する場合

ここでは、CPUユニットにバス接続ユニットとKV-B16XCを接続する場合を例に説明します。



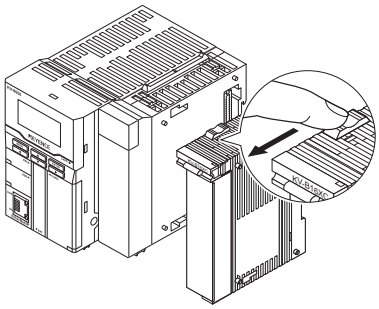
1 **右端に取り付けられたKV-8000/7000シリーズ拡張ユニットの拡張ユニット接続コネクタカバーをはずします**

2 **バス接続ユニットを取り付けます**

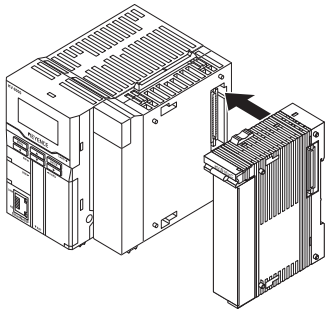
- KV-8000/7000シリーズ拡張ユニットを接続する場合の取付手順を参照してください。

3 **KV-5000/3000シリーズ拡張ユニットのロックレバーをはずします**

- ロックレバーに指をかけ、ユニットの前面方向にスライドさせるとはずれます。

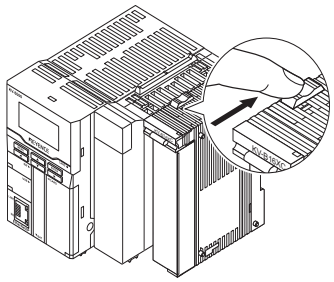


4 **バス接続ユニットのロックレバースロットに、KV-5000/3000シリーズ拡張ユニットのロックレバーを差し込んで接続します**



5 **右側のユニットの上下に付いているロックレバーをロックします**

- 手順3 と同じ要領で、ロックレバーに指をかけ、ユニットの背面方向にスライドさせます。このときユニットとユニットの間にすき間がないかどうか確認します。すき間があれば、正しく接続されていません。



6 **右端にエンドユニット取り付けます**

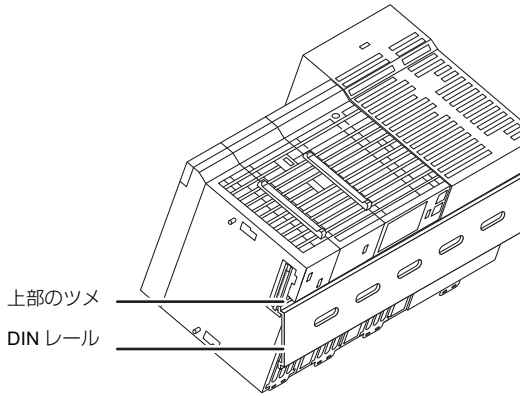
ポイント　**バス接続ユニットを使用した場合、右端にエンドユニットを取り付けないとエンドユニット未接続エラーが発生します。**

DINレールへの取り付け

接続されたユニットをDINレールに取り付ける方法と取りはずす方法を説明します。

■ 取りつけるときは

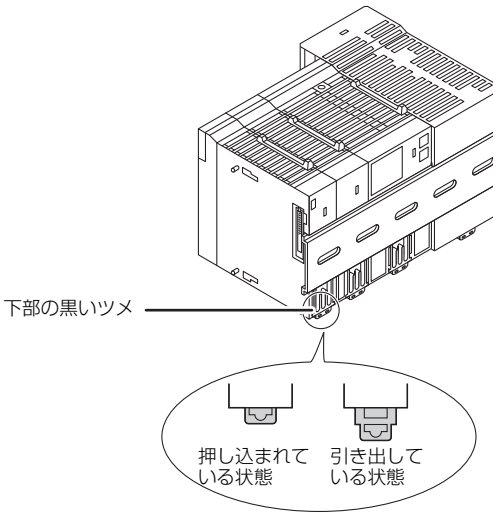
1 各ユニットの上側のDINレール取り付け用ツメに、DINレールの上側の溝を引っかけます



2 各ユニットの下側のDINレール取り付け用ツメに、DINレールの下側をはめます

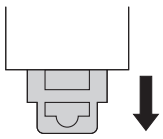
- 下側の黒いツメはバネで動くようになっています。「カチン」と音がするまで押しあてます。
- 取り付けにくい場所は全てのロックをはずしておき、DINレールに取り付けてから、ロックをしてください。

3 下側のDINレール取り付け用ツメが、押し込まれている状態(ロック状態)になっていることを確認します



■ 取りはずすときは

1 下側のDINレール取り付け用ツメを下側に引き出します



2 取り付けるときと逆の要領でDINレールをユニットから取りはずします

3-2 ユニットの設置

各ユニットを設置する環境についての注意や、盤内の取り付け位置について説明します。

設置環境

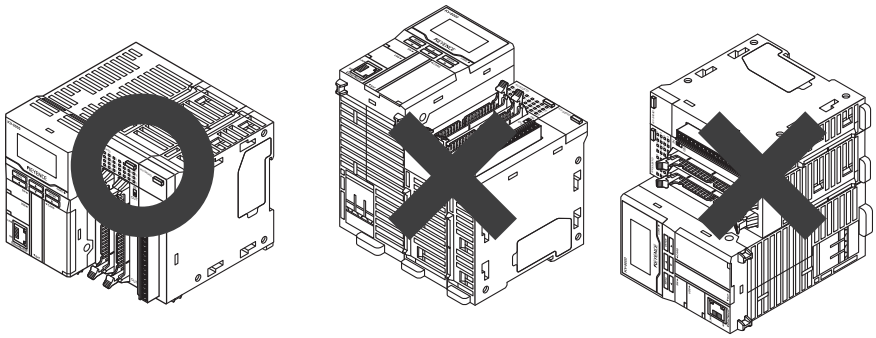
次のような場所には設置しないでください。

- 直射日光があたる場所
- 周囲温度が0～＋55℃の範囲を超える場所
- 周囲湿度が5～95%RHの範囲を超える場所
- 急激な温度変化で結露する場所
- 腐食性ガスや可燃性ガスのある場所
- 塵埃、塩分、鉄粉、油煙が多い場所
- 振動や衝撃が直接加わる場所
- 水、油、薬品などのしぶきがかかる場所
- 強磁界、強電界の発生する場所
- 標高2000mを超える場所

取り付け位置

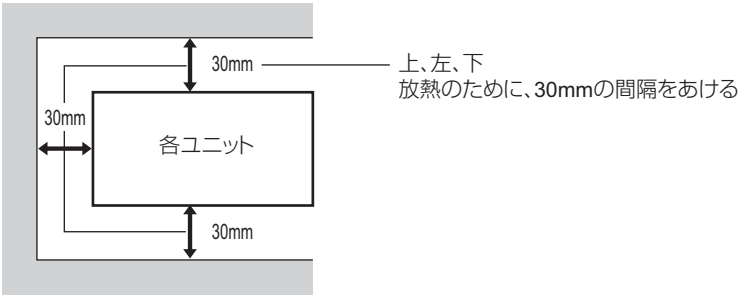
■ 設置の向き

盤内に取り付けるときは、必ずCPUユニットの前面が手前に向くようにします。



■ 周囲との間隔

周囲の壁や他の機器とは、必ず30mm以上離して設置してください。



ポイント

盤内の温度(ユニット中央下部の温度)が、使用周囲温度を超える場合は、強制空冷や周囲との間隔を広げ通風をよくするなどの方法で温度を下げてください。
エクステンションユニット使用時、上段と下段のユニット間隔は熱の影響を考慮して60mm以上離してください。

3-3 配線時の注意

コネクタの組み立て方法や各ユニットの結線と内部回路、および各ユニットの配線方法について説明します。

各ユニットへの配線時の注意

入出力機器や拡張入出力ユニットに配線するときの注意について説明します。必ず、配線前にお読みください。

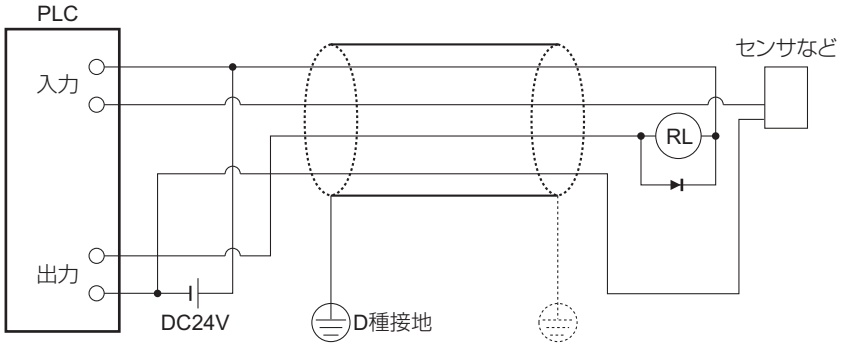
■ 入出力機器への配線時の注意

入出力機器に配線するときは、次のことに注意してください。

- 入力線と出力線は、それぞれ分離して配線してください。
- 入出力信号線に動力用配線を近づけると、高電圧･大電流の影響を受け動作異常が発生することがあります。入出力信号線と動力用配線は100mm以上離して配線してください。
- 入出力信号線は、AC100VやAC200Vの線とは分離して配線してください。
- 配管を使っての配線時には、配管が確実に接地されていることを確認してください。
- 配線時には、必ず全ての電源をOFFしてください。感電や製品損傷の恐れがあります。
- 配線作業などの後、通電、運転する場合は、必ず製品に付属の端子カバーを取り付けてください。感電の恐れがあります。

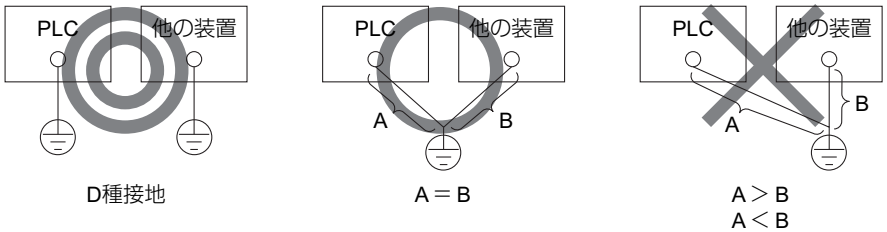
動力用配線と分離できないときは･･･

一括シールドのケーブルを使い、PLC側で接地してください。
（環境によっては、PLCと逆側や、両側で接地する必要が生じる場合があります。）



■ 接地時の注意

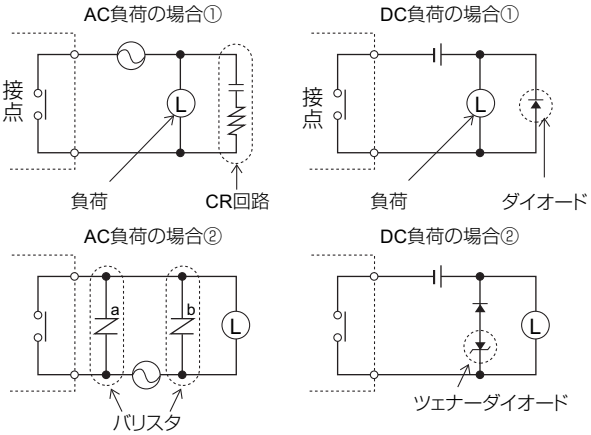
- 機器ごとに別々に接地（接地工事はD種接地）してください。また、この場合の接地抵抗は100Ω以下とします。
- 別々に接地できないときは、共用で接地します。ただし、この場合の電線はすべて同じ長さにします。
- 接地にはFG端子を使用してください。（電源端子は接地しないでください）



■ 接点保護について

クラッチやモータ、ソレノイドなどの誘導負荷を使う場合、負荷電源投入時に突入電流が流れたり、負荷電源遮断時に逆起電圧が発生したりします。突入電流や逆起電圧は、接点の寿命を短くする大きな要因となります。この現象の発生を抑えるために、接点保護回路を設けるようにしてください。

接点保護回路の例



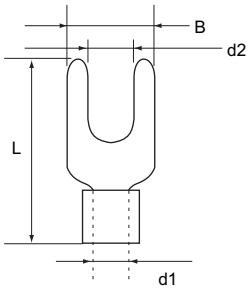
・電源電圧 24～ 48Vのとき　bの位置へ
100～200Vのとき　aの位置へ
取り付けてください。

- 負荷コイル定格は、接点容量以下のものを使用してください。
- ダイオードは、逆耐電圧が回路電圧の10倍以上のもので、順方向電流が負荷の電流以上のものを使用してください。
- ダイオード、バリスタ、およびCR回路は、リレーコイル端子に直接取り付けてください。

■ AC電源ユニット/電源端子台/入出力端子台への配線

KV-PU1の端子台の配線は、AWG#12 ～ #22 の銅線（より線）で、温度定格が105 ℃以上の電線を使用してください。締付けトルクは、1.4N•m です。
DC電源入力端子台およびネジ端子タイプの端子台の配線は、AWG#16 ～ #22 の銅線（より線）で、電源端子台については温度定格が105 ℃以上の電線を使用してください。締付けトルクは、0.5N•m です。
端子台ユニットの締付トルクは0.8N•mです。
Y端子は次のサイズの物を使用してください。

Y端子の各部分の寸法値抜粋
B ： Y字部分外側サイズ
d1 ： 電線差込部分の径
d2 ： Y字部分内側
（ネジに噛む部分）幅
L ： 全長



適合寸法
B ：6mm以下
L ：13mm以下
d2：3.2mm以上

推奨品

メーカ	型式	B	d1	d2	L	適合電線（より線）
日本端子株式会社	VD1.25－3	5.5	1.7	3.3	11.5	0.25～1.65mm^2（AWG22～16）
大同端子製造株式会社	F1.25－C3.5	6.0	1.7	3.7	12.8	0.25～1.65mm^2（AWG22～16）
株式会社ニチフ	0.3Y－3	5.2	1.0	3.2	12.0	0.3～0.5mm^2（AWG22～20）
日本圧着端子製造株式会社	0.5－3A	5.0	1.0	3.2	12.5	0.2～0.5mm^2（AWG26～22）
富士端子工業株式会社	V1.25－YASS3.5	5.7	3.3	3.8	20	0.25 ～1.65 mm^2 （AWG22～16）

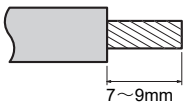
参 考　端子台へ配線時は、ドライバと端子台ネジが垂直になるようにしてください。ドライバと端子台ネジが垂直でない状態で締め付けすると、端子台を破損する恐れがあります。

■ ヨーロピアン端子への配線

ヨーロピアン端子台の配線は、AWG#12～#24の銅線（より線）を使用してください。締め付けトルクは、0.5N•mです。
ケーブルの端末処理については、下記を参照してください。

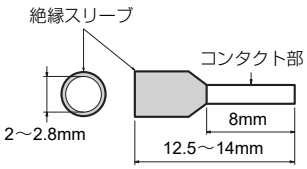
(1) より線、単線をそのまま処理する場合

(a) より線の末端は、“ヒゲ線”が出ないようによじってください。
(b) 電線の末端は、ハンダメッキしないでください。



(2) 絶縁スリーブ付棒端子を使用する場合

電線のシースの厚みによっては、絶縁スリーブに入れにくくなるため、外形図を参考に電線を選定してください。



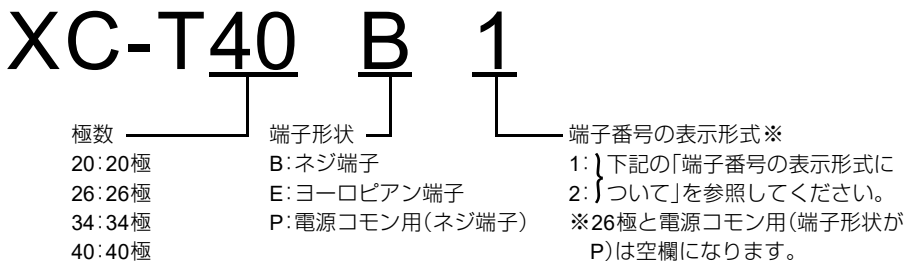
メーカー	形名
フェニックス･コンタクト株式会社	AI0.25-8BU(AWG24)
	AI0.34-8TQ(AWG22)
	AI0.5-8WH(AWG20)
	AI0.75-8GY(AWG18)

端子台ユニットと接続ケーブル

コネクタタイプのユニットの配線に使用できる端子台ユニットと接続ケーブルについて説明します。

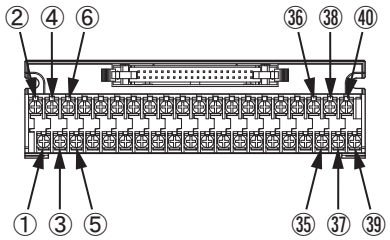
■ 端子台ユニットの型式とラインナップ

端子台ユニットは極数、端子形状、端子番号の表示形式により、型式が異なります。
端子台ユニットの型式の見方は、以下のとおりです。

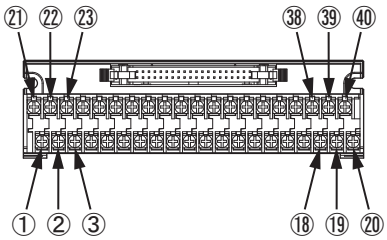


● 端子番号の表示形式について

「1」の表示例（40極の場合）



「2」の表示例（40極の場合）



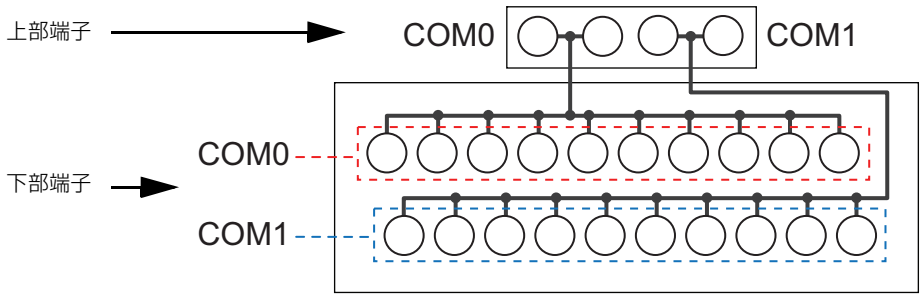
ポイント　**使用するユニットのコネクタピンNo.に対応した表示形式の端子台ユニットを使用してください。**

■ 電源コモン用端子台ユニットについて

電源コモン用端子台ユニットは、電源の配線を多点に分岐するためのネジ端子台ユニットです。
電源ラインに複数の機器を接続する場合に配線を簡略化できます。

● 内部結線図

XC-T20P

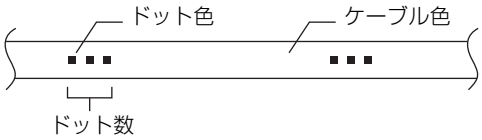


上部端子のCOM0（2箇所）と下部端子の上段側は全て内部で短絡しています。また、上部端子のCOM1（2箇所）と下部端子の下段側は全て内部で短絡しています。
XC-T40Pは下部端子の上段側、下段側の端子が各々20箇所になります。

■ 接続ケーブルの型式とラインナップ

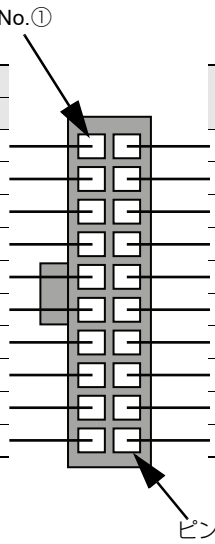
接続ケーブルは極数、形状、ケーブル長により、型式が異なります。
接続ケーブルの型式の見方は、以下のとおりです。

■ 接続ケーブル（片側コネクタ、片側バラ線）の線色



● XC-H20D-05

拡張ユニットのコネクタ側から見た図



ケーブル色	ドット		ケーブル色	ドット	
	色	数		色	数
橙	赤	1	橙	黒	1
灰	赤	1	灰	黒	1
白	赤	1	白	黒	1
黄	赤	1	黄	黒	1
桃	赤	1	桃	黒	1
橙	赤	2	橙	黒	2
灰	赤	2	灰	黒	2
白	赤	2	白	黒	2
黄	赤	2	黄	黒	2
桃	赤	2	桃	黒	2

● **XC-H34D-05**

拡張ユニットのコネクタ側から見た図

ピンNo.①

ケーブル色	ドット		ケーブル色	ドット	
	色	数		色	数
橙	赤	1	橙	黒	1
灰	赤	1	灰	黒	1
白	赤	1	白	黒	1
黄	赤	1	黄	黒	1
桃	赤	1	桃	黒	1
橙	赤	2	橙	黒	2
灰	赤	2	灰	黒	2
白	赤	2	白	黒	2
黄	赤	2	黄	黒	2
桃	赤	2	桃	黒	2
橙	赤	3	橙	黒	3
灰	赤	3	灰	黒	3
白	赤	3	白	黒	3
黄	赤	3	黄	黒	3
桃	赤	3	桃	黒	3
橙	赤	4	橙	黒	4
灰	赤	4	灰	黒	4

ピンNo.㉞

● **XC-H40D-05**

拡張ユニットのコネクタ側から見た図

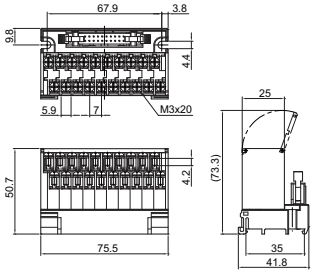
ピンNo.①

ケーブル色	ドット		ケーブル色	ドット	
	色	数		色	数
橙	赤	1	橙	黒	1
灰	赤	1	灰	黒	1
白	赤	1	白	黒	1
黄	赤	1	黄	黒	1
桃	赤	1	桃	黒	1
橙	赤	2	橙	黒	2
灰	赤	2	灰	黒	2
白	赤	2	白	黒	2
黄	赤	2	黄	黒	2
桃	赤	2	桃	黒	2
橙	赤	3	橙	黒	3
灰	赤	3	灰	黒	3
白	赤	3	白	黒	3
黄	赤	3	黄	黒	3
桃	赤	3	桃	黒	3
橙	赤	4	橙	黒	4
灰	赤	4	灰	黒	4
白	赤	4	白	黒	4
黄	赤	4	黄	黒	4
桃	赤	4	桃	黒	4

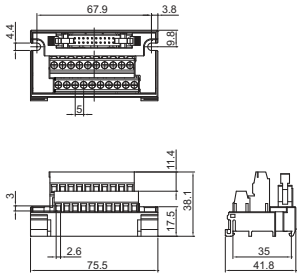
ピンNo.㊵

■ 外形寸法図

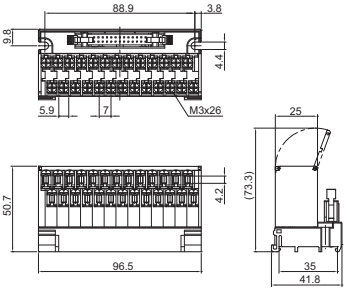
XC-T20B1/B2



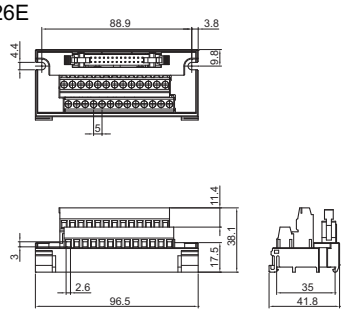
XC-T20E1



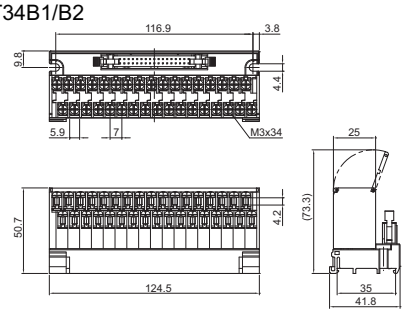
XC-T26B



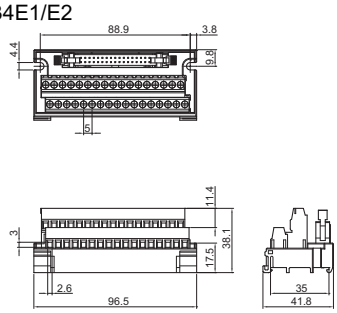
XC-T26E



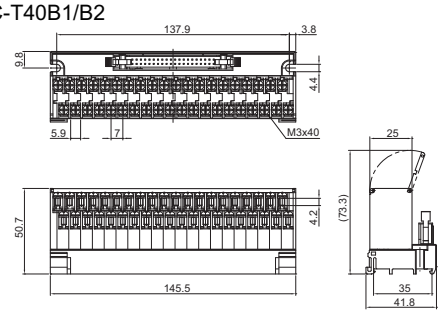
XC-T34B1/B2



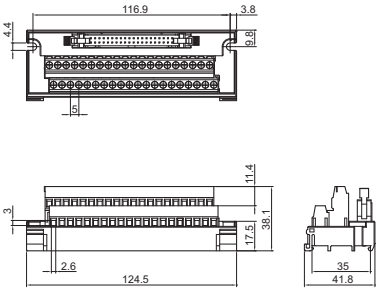
XC-T34E1/E2



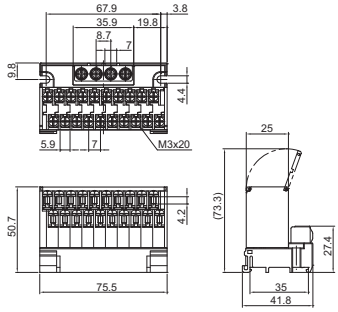
XC-T40B1/B2



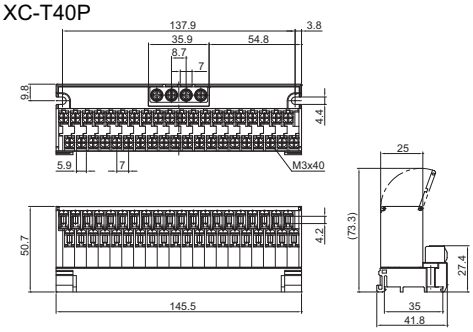
XC-T40E1/E2



XC-T20P



XC-T40P



コネクタの組み立てと結線

各ユニットに機器を接続するには、ユニット側に接続するコネクタ(メス)に、各機器からの電線を結線する必要があります。

■ 結線作業時の注意(OP-22184/51404/23139/42224)

結線する前は、次の点にご注意ください。

- 必ず、次の適合電線を使う
  電線サイズ： AWG24～22(より線)
  断面積： 0.2～0.3mm^2
  被覆外径： φ1.5～φ1.1
- 電線の先端は、フラットになるようにまっすぐ切る
- 結線作業後は、クランプ部が電線を完全に抱き込み、奥まで差し込まれていることを確認する
- 電線の切断部は、ひげが残らないようにカットする(被覆をむく必要はありません)。
- AWG28～26用細線コンタクトOP-30594もご用意しています。
  (OP-22184, OP-51404等と組み合わせでご使用ください。)
  電線サイズ： AWG28～26
  断面積： 0.08～0.14mm^2
  被覆外径： φ1.3～φ1.1

注 記	**ユニットは、合成樹脂製です。アセトン、トリクレン、ベンジンなどの溶解力の強い溶剤に触れると溶けるおそれがあります。また、圧接作業後に過大な力を加えると破損するおそれがあります。ご注意ください。**

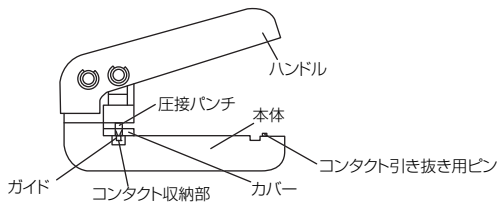
ポイント **64点ユニットにはOP-42224コネクタ(34ピン)をご使用ください。**
OP-23139は使用できません。

■ 専用圧接工具(OP-21734)について

電線を圧接するには、専用圧接工具を使うと簡単に作業することができます。
次の専用圧接工具を前もってご用意ください。

ポイント **専用圧接工具がないときは、ハンダ付けする必要があります。**

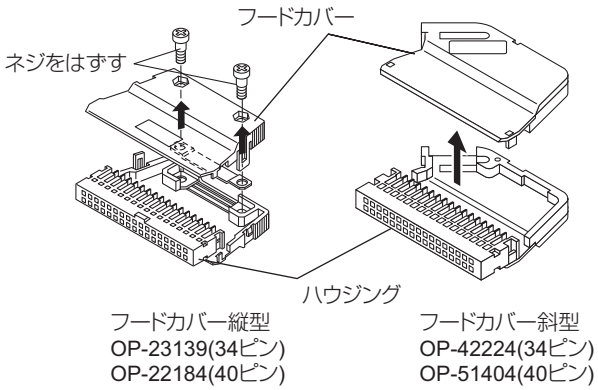
各部の名称



専用圧接工具：型式OP-21734

■ コネクタについて

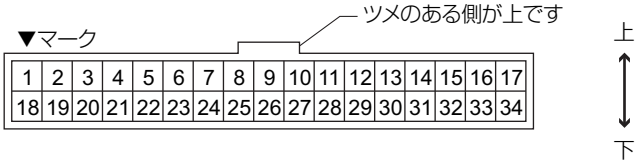
コネクタには上下のカバーが取り付けられています。結線するときは、上下のカバーをはずして結線します。



コネクタの端子番号

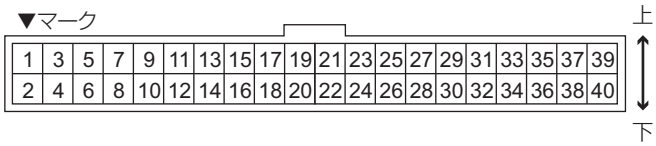
コネクタは、ユニットの種類により34ピンと40ピンの2種類あります。コネクタの端子番号は次のようになっています。結線作業時には、「各ユニットの結線・内部回路図」と併せて確認してください。

34ピンコネクタ：型式：OP-23139、OP-42224



1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34

40ピンコネクタ：型式：OP-22184、OP-51404



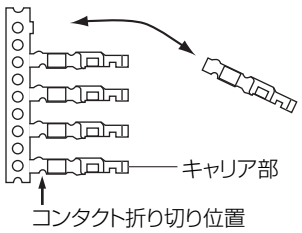
1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	31	33	35	37	39
2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40

※KV-5000/3000シリーズの拡張ユニット使用時はピン番号は横並びに、1, 2, 3…と並びます。

■ コネクタへの結線作業

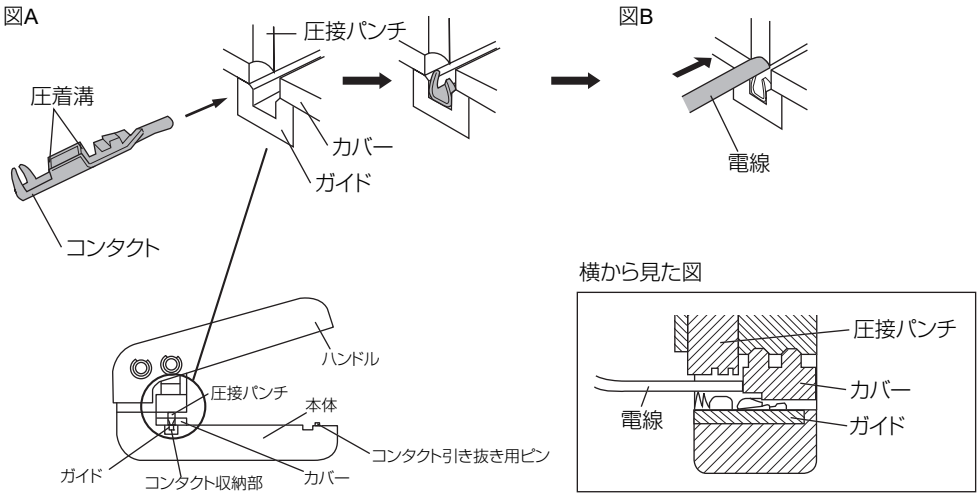
専用圧接工具を使って結線する方法について説明します。前もって、「コンタクト」「ハウジング」「電線」をご用意ください。

1 **コンタクトのキャリア部の折り切り位置から、矢印の方向に折るように切り離します**

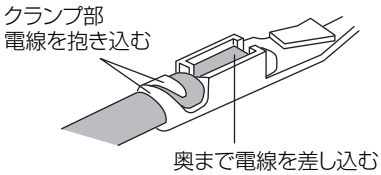


2 **コンタクトを、専用圧接工具のコンタクト収納部にまっすぐ突き当たるまで差し込みます（図A）**

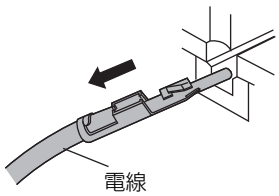
3 **電線を、コンタクトの圧接溝に沿うように、まっすぐ突き当たるまで差し込みます（図B）**



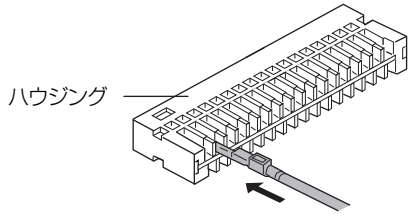
4 **クランプ部が、完全に電線を抱き込む形になるまで専用圧接工具のハンドルを握ります**



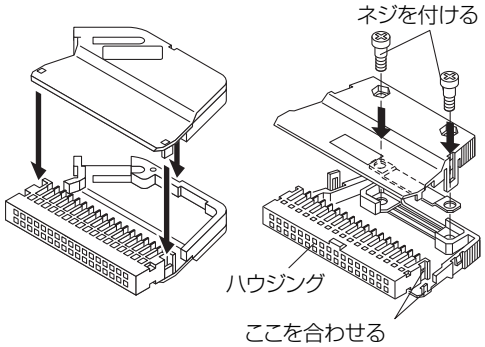
5 **電線を軽く引っ張り、専用圧接工具の収納部から電線が圧接されたコンタクトを取り出します**



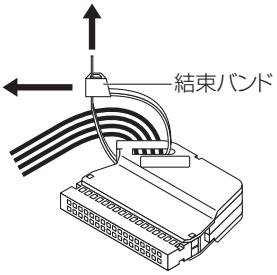
6 コンタクトが付いていない状態のハウジングの溝に沿うように、まっすぐ突き当たるまで差し込みます



7 コンタクトが差し込まれた状態のハウジングをコネクタに装着し、カバーを取り付けます



8 電線を図のように結束バンドで固定します

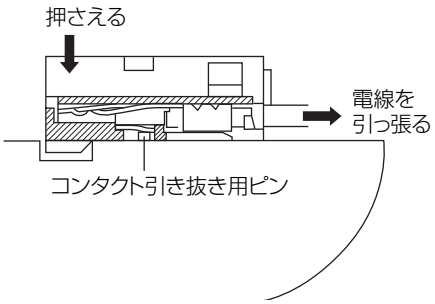


● コンタクトを取りはずすには

コネクタからコンタクトを取りはずすには、次のようにします。

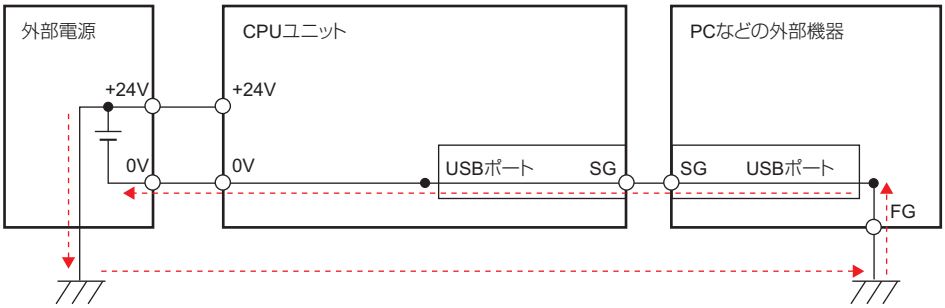
1 抜きたい端子を、専用圧接工具のフレームの矢印のところに合わせます

2 コンタクトを押さえながら、電線を引っ張り出します



PCなどの外部機器とUSBで通信する場合の注意点

CPUユニットの電源0VとUSBポートのSGは導通しています。接続先のSGと電位差が発生すると、CPUユニットおよび接続先の機器が故障する原因となります。たとえば、プラス接地（24V側接地）した場合、電位差が発生し下図の経路で電流が流れて、機器が破損することがあります。下図の様な配線はしないでください。



参 考

上記問題は以下の方法で防止できる場合があります。

- 24V側接地（プラス接地）をしない
- KV-PU1を使用する
- CPUユニットに接続されている電源に絶縁電源を使用する
- PCなど外部機器のFGを接地しない
- アイソレータを使用して、USBの信号線を電気的に絶縁する

3-4 保守

ユニットの点検やお手入れの方法、およびバッテリーやリレーの交換方法について説明しています。

点検とお手入れについて

■ 点検

CPUユニットや各ユニットを長期間使用していると、コネクタの接続部分がゆるむなど、そのまま使い続けると動作に支障を来たすおそれが生じます。
このため、CPUユニットや各ユニット本体、配線部分などは定期的に点検するようにしてください。
おもな点検項目は次のとおりです。

- 各ユニット接続部分のコネクタが、はずれたりゆるんだりしていませんか？
- コネクタ接続部分が、はずれたりゆるんだりしていませんか？
- 端子台のネジがゆるんでいませんか？
- ユニットと各機器間の配線ケーブルなどが破損していませんか？

■ お手入れ

CPUユニットや各ユニットを長期間使用していると、汚れが付着してきます。
付着した汚れは、清潔な乾いた布でふき取ってください。
また、コネクタ部分などの細かい部分に付着しているホコリや汚れは、一旦コネクタをはずし、綿棒などを使って取るようにしてください。

保守について

■ バッテリーについて

ラダープログラム･デバイスコメント･各種設定情報などはフラッシュROMに、デバイス値･エラー履歴などは不揮発RAMに、それぞれ記憶されています。
ただし、時計データはバックアップ用コンデンサで保持されているため、データがクリアされる場合があります。時計データのバックアップが必要な場合バッテリ（KV-B1）を使用してください。

「時計データの設定方法」（3-22ページ）

■ バッテリー低下時について

アクセスウィンドウに[CPUエラーE83]または[CPU Error E83]が表示されたら、すぐにリチウム電池を交換する必要があります。
バッテリーエラー発生時はCR2306がONします。
バッテリーエラーが発生してからデータを保持できる期間は最低1週間（常温）です。
エラーチェックは、電源立ち上げ後および、24時間毎におこないます。
バッテリーを使用していない場合、バックアップ用のコンデンサの電圧が低下すると、[CPUエラーE85]または[CPU Error E85]が表示されます。

■ 廃棄について

CPUユニットを廃棄する場合、各種規制に従い適切に廃棄を行ってください。
CPUユニットを廃棄する際は、データ漏洩を防ぐためにPLC内部の全てのデータを消去することを推奨します。
CPUユニット内のストレージに残された情報を復元できないように完全に消去するには、「初期化」を実施してください。「初期化」（5-158ページ）

■ バッテリー切れについて

バッテリーが切れると時計データの値は消えますが、データメモリやラダープログラム自体は消えません。時計データの情報が消えても動作に問題がないラダープログラムの場合は、アクセスウィンドウでエラークリアをすると一時的にプログラムを動作させることができます。
電池は次の製品をお買い求めください。
交換用リチウム電池　型番 : KV-B1

参 考　バッテリーを使用していない場合、バッテリー切れエラーは発生しません。

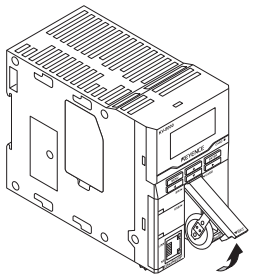
■ バッテリーレス運転について

CPUユニットはバッテリーを使用しない状態で動作させることも可能です。
設定は不要ですが、バックアップ用コンデンサの電圧が低下すると、時計データがクリアされ、[CPUエラーE85]または[CPU Error E85]が表示されます。

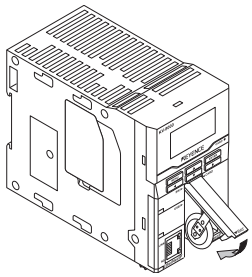
なお、以下の方法でエラーを回避できます。
- CPUシステム設定（エラー発生時の動作設定）で、時刻異常のCPU動作を続行（無視）にする。

■ バッテリー交換手順

1 **バックアップ用電池のカバーを外して、古い電池を取り外します**
- 電源OFF中に交換する場合は、交換前に10分間以上通電し、交換作業を10分以内に完了させてください。



2 **新しい電池を挿入します**



3 **電源をONして、アクセスウィンドウにエラーが表示していないことを確認します**

注 記	• バッテリの寿命は以下のとおりです。 25℃ : 5年 40℃ : 2年 75℃ : 1年 • 電源ON中もバッテリの交換可能です。 • 電池交換前に10 分以上通電しないと、データが消失することがあります。電池交換は10分以内でおこなってください。 • CPUユニットに使用されるバッテリーを火中に投じたり、分解しないでください。また、本バッテリーは充電式ではありませんので、充電しないでください。爆発の危険性があります。 • KV-B1以外の電池を使用されますと火災または爆発の恐れがありますので必ず指定の電池を使用してください。

復旧方法について

■ 時計データの設定方法

CPUユニットの時計データは内蔵のバックアップ用コンデンサで保持されており、通電しない状態で保持した場合、時計データはクリアされてしまいます。時計データを常に保持する場合は、バッテリー（KV-B1）を取り付けて使用してください。

※ バックアップ用コンデンサを満充電するためには30分以上通電してください。

バックアップ用コンデンサの保持時間（満充電時）

温度	TYP	MIN
25℃	60日	15日
40℃	40日	10日
75℃	12日	3日

必要に応じて、以下のいずれかの方法で設定してください。

- 《KV STUDIO》のモニタ実行中に、メニューから【デバッグ（D）】→【カレンダタイマの設定（C）】
- ラダープログラムでWTIME命令を使用する
- アクセスウィンドウから時計で設定する
- タッチパネル（VT5/VT3シリーズ）を使用している場合、PLC時刻同期機能を使用する
  （修復機能を使用している場合、自動で時刻を調整できます）

『VT STUDIOリファレンスマニュアル』

参 考

時計データがクリアされると、以下の様な機能が正常に動作しない場合があります。

- ロギング･トレース機能（ロギング/トレース実行時間が異なります）
- エラーモニタ機能（エラー発生時の時刻が異なります）
- ウィークリー接点命令

復旧方法

■ エラー発生時のクリア方法

CPUユニットでエラーが発生した場合、以下の方法でエラークリアを実行できます。

- アクセスウィンドウからエラークリアを実行します。
  「エラークリア」(5-129ページ)
- 《KV STUDIO》でモニタ中は、エラー発生時に起動するポップアップウィンドウからクリアできます。
- ラダープログラムでエラークリア要求リレー(CR3909)をSETします。(軽度エラーのみ)
- 電源の切り入りを実行します。

ポイント

- **CR に BOOL の配列型変数を割り当てた場合、配列の要素番号は数値で指定してください。**
  デバイスや変数で指定した場合、SET命令が正しく動作しません。
- **システムに関わる重大なエラーの一部では、電源の再起動が必要になる場合があります。**

■ PLCの状態をクリアする方法

CPUユニットの状態を出荷状態にしたり、プログラム内で使用されるデバイスの値をクリアしたりできます。

出荷状態にする場合

ラダープログラムやデバイスコメント、各種設定情報、デバイス値・エラー履歴・イベント履歴・CPUメモリなどをクリアします。
以下のいずれかの方法を使用してクリアしてください。

- 《KV STUDIO》のメニューから【ツール(T)】→【PLCクリア(1)】を選択し、[PLCクリア]ダイアログボックスで、「オールクリア」、または「初期化」にチェックして〔実行(E)〕ボタンをクリックします。
- アクセスウィンドウからオールクリア、または初期化を実行します。

注 記	
	• **初期化を実行すると、CPUユニット内のストレージを全て無効なデータで上書きし、中のデータは完全にクリアされます。** • **オールクリア/初期化中は電源を切らないでください。完了したかどうかはアクセスウィンドウの表示内容で確認できます。**

デバイスの値をクリアする場合

デバイス値・エラー履歴などをクリアします。
以下のいずれかの方法を使用してクリアしてください。

- 《KV STUDIO》のメニューから【ツール(T)】→【PLCクリア(1)】を選択し、[PLCクリア]ダイアログボックスで、「RAMクリア」にチェックして〔実行(E)〕ボタンをクリックします。
- アクセスウィンドウからRAMクリアを実行します。

CPUメモリ(ユーザメモリ)をクリアする場合

CPUメモリ内のユーザドキュメント・ファイルレジスタ設定・ロギングデータ・プロジェクトセーブデータなどをクリアします。
以下のいずれかの方法を使用してクリアしてください。

- 《KV STUDIO》のメニューから【ツール(T)】→【PLCクリア(1)】を選択し、[PLCクリア]ダイアログボックスで、「ユーザメモリクリア」にチェックして〔実行(E)〕ボタンをクリックします。
- アクセスウィンドウからユーザメモリクリアを実行します。

全履歴をクリアする場合

CPUユニットのエラー履歴、電源ON/OFF履歴、PLCイベント履歴をすべてクリアします。
以下の方法でクリアしてください。

- ラダープログラムでイベントログクリアリレー(CR3910)をSETします。

ポイント

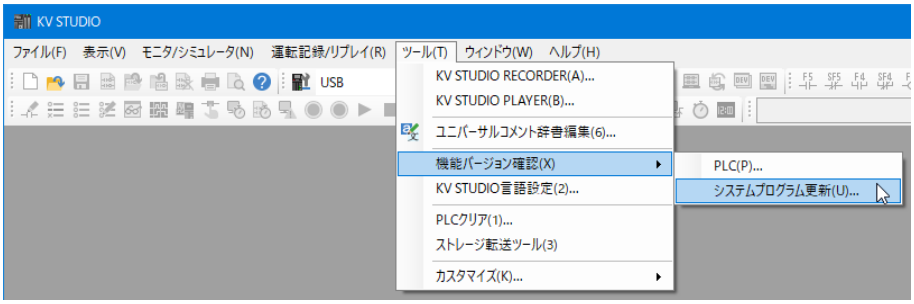
- **CR に BOOL の配列型変数を割り当てた場合、配列の要素番号は数値で指定してください。**
  デバイスや変数で指定した場合、SET命令が正しく動作しません。

システムプログラムアップデートについて

システムプログラムをKV STUDIOやメモリカードからアップデートすることで、CPU機能バージョンをバージョンアップできます。アップデート手順は以下のとおりです。

KV STUDIOからアップデートする場合

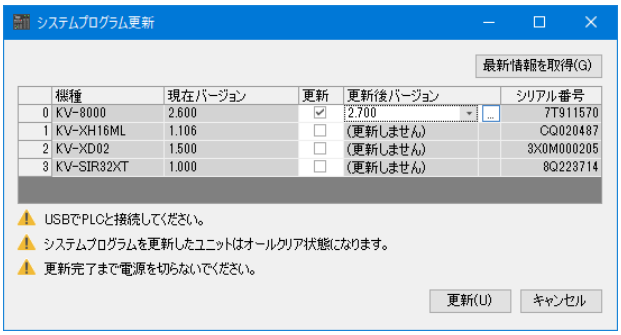
1 **KV STUDIOのメニューで【ツール(T)】→【CPU機能バージョン確認(X)】→【システムプログラム更新(U)】から確認できます。**



ポイント

- システムプログラムアップデートはUSBで接続した場合のみ実行できます(イーサネット接続など他の接続方法の場合実行できません)。

2 **接続しているKV-8000シリーズが最新バージョンかどうか確認し、最新バージョンでない場合は、〔更新〕ボタンをクリックしてシステムバージョンアップを開始します。**

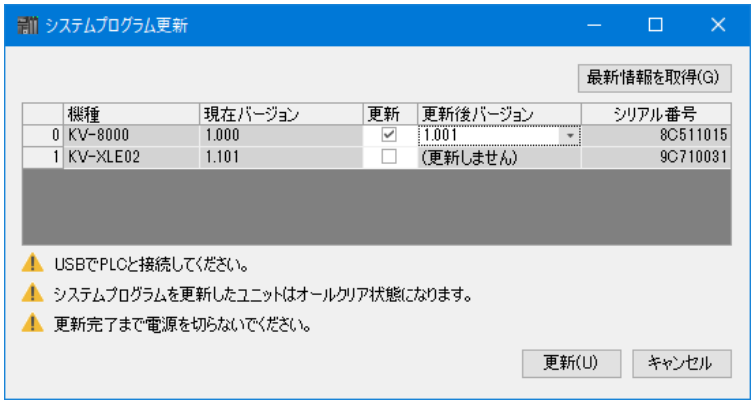


ポイント

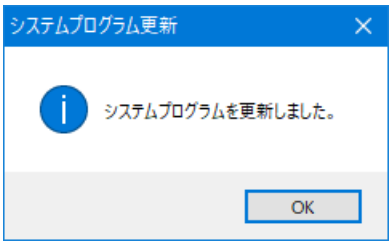
- システムアップデート実行中は電源を切らないでください。
- システムプログラムアップデートを実行すると、ユニット内部のプログラムは消去されます。システムプログラムアップデート前に、プログラムをバックアップしてください。
- システムプログラムを最新にアップデートするには、事前にKV STUDIOを最新にアップデートしてください。
- KV STUDIO Ver.11.63以降を使用している場合、システムプログラムバージョンが2.7以降だと更新後バージョンの右側の ... を押下することで、更新後バージョンのチェックコードを確認できます。ホームページのKV STUDIO変更履歴の中で公開しているチェックコードと比較して一致しているかを確認することで、当社製の正常なアップデートファイルかどうかを確認できます。

ポイント

- システムプログラムが最新の場合、以下のダイアログが表示されます。

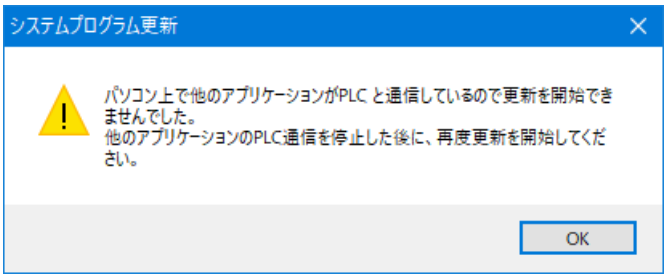


3 完了すると以下のダイアログが表示されます。



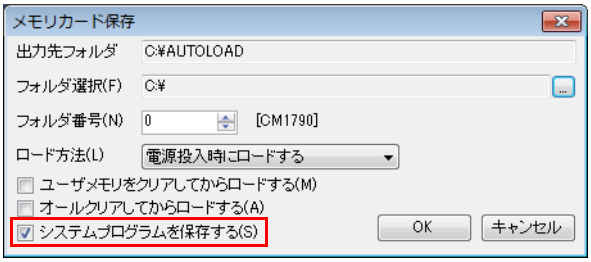
ポイント

- システムプログラムアップデート実行時にCPUユニットとパソコンの間で他の通信が実行される場合、以下のダイアログが表示され、システムアップデートは実行されません。



メモリカードからアップデートする場合

1 KV STUDIOのメニューで【ファイル(F)】→【メモリカード(M)】→【保存(W)】でメモリカードデータを作成します。



2 保存したメモリカードデータをメモリカードに格納しロードを実行します。

ポイント

- メモリカードからのロード(オートロード / ランロード含む)については 「ロード」(5-142ページ) 「オートローディング機能」(5-45ページ)を参照してください。
- メモリカードデータをCPUメモリに格納してロードを実行した場合、ロードは失敗します。

4

ラダープログラミング

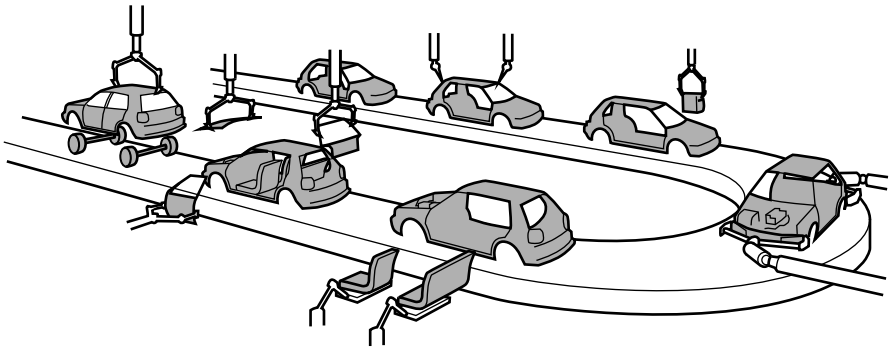
ラダープログラムを作成する上で必要となる機能の仕様や設定方法を説明します。モジュール、ファンクションブロック、マクロ、ローカルデバイスを使用すればプログラミングを効率化できます。

4-1 シーケンス制御とは 4-2
4-2 プログラミング言語 4-5
4-3 プロジェクト 4-11
4-4 デバイス 4-20
4-5 変数 4-55
4-6 定数 4-73
4-7 データの取り扱い方 4-75
4-8 プログラムの構成と動作 4-82
4-9 モジュール 4-98
4-10 ファンクションブロック 4-117
4-11 マクロ 4-178
4-12 ローカルデバイス 4-200
4-13 割り込み 4-206
4-14 ファイルレジスタ 4-226
4-15 プログラミングのコツ 4-236

4-1 シーケンス制御とは

下図のような自動車の組立工程では、機械や装置の各動作、順序および条件を制御装置にあらかじめ記憶させておきます。運転を開始すると、機械や装置の状態・条件に従って一連の目的動作が順次実行され、自動車が組み立てられていきます。このような制御をシーケンス制御と呼びます。
シーケンス制御のための装置に、《KV-8000シリーズ》のようなPLC（Programable Logic Controller）が使われます。

＜自動車工場の車体組立工程＞

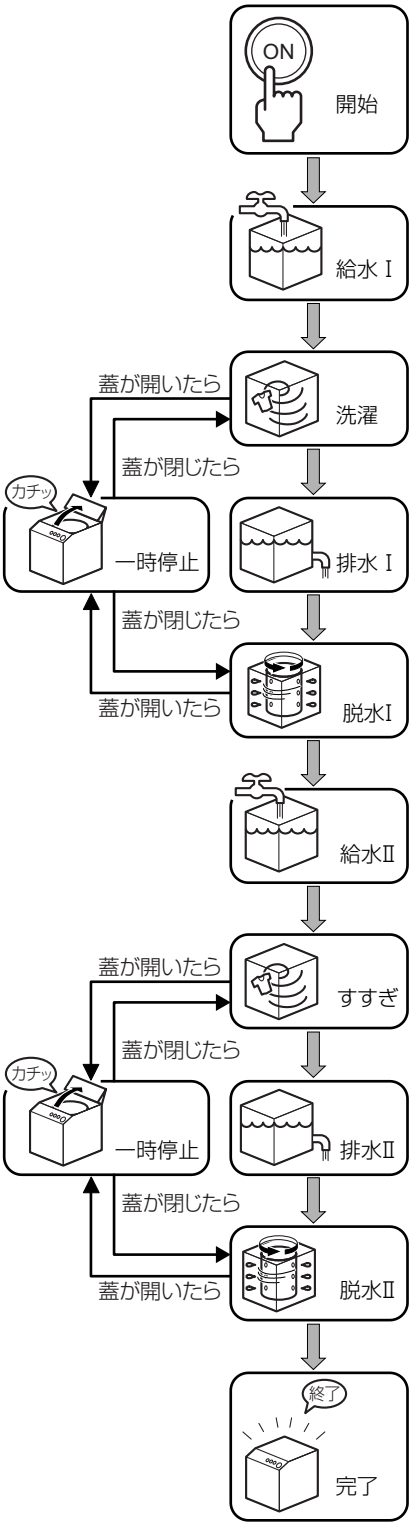


参 考

- JIS（日本工業規格）には、「シーケンス制御とは、あらかじめ定められた順序、または、一定の論理によって定められた順序に従って制御の各段階を逐次進めていく制御」と定義されています。
- PLCの制御には、シーケンス制御の他にフィードバック制御があります。
  フィードバック制御は、対象から検出した現在値と目標値を比較した結果が一致するまで制御を繰り返します。
  （例）温度制御、圧力制御、速度制御

■ シーケンス制御の例

シーケンス制御について、全自動洗濯機を例にして具体的に見てみましょう。
全自動洗濯機はスタートボタンを押すと、あらかじめ定められた順序に従って、洗濯工程が自動的に進行していきます。この洗濯工程の制御は、次のようになります。



①洗濯物を投入し、スタートボタンを押します。

②スタートボタンが押されたら、洗剤を投入して給水バルブを開けます。

③規定の給水量（水位）へ達したことを検出したら、給水バルブを閉じて洗濯用モータを起動します。

④規定の時間が経過したら、洗濯用モータを止めて排水バルブを開けます。

⑤排水完了（水位が0）を検出したら、排水バルブを閉じて脱水用モータを起動します。

⑥規定の時間が経過したことを検出したら、脱水用モータを止めて給水バルブを開けます。

⑦規定の給水量（水位）へ達したことを検出したら、給水バルブを閉じて洗濯用モータを起動します。

⑧規定の時間が経過したことを検出したら、洗濯用モータを止めて排水バルブを開けます。

⑨排水完了（水位が0）を検出したら、排水バルブを閉じて脱水用モータを起動します。

⑩規定の時間が経過したことを検出したら、脱水用モータを止めます。
洗濯完了を表示して、音で通知します。

シーケンス制御の種類

基本的なシーケンス制御は、主に「順序制御」「条件制御」「時限制御」の3種類に分けられます。これらの制御を組み合わせて、目的とする制御回路を設計します。

● 順序制御

順序制御は、検出器など、外部からの信号によって動作終了を確認し、あらかじめ定められた順序に従って、次の動作を開始する制御です。
〔全自動洗濯機のシーケンス制御における順序制御〕
「②給水Ⅰ」→給水が完了→「③洗濯（洗濯用モータの起動）」
「④排水Ⅰ」→排水が完了→「⑤脱水Ⅰ（脱水用モータの起動）」

参 考　順序制御で使用する代表的なラダー命令語･･･「STG」「JMP」など

● 条件制御

条件制御は、制御対象の状態信号を組み合わせて、あらかじめ定められた条件が満足された時に動作させる制御です。
〔全自動洗濯機のシーケンス制御における条件制御〕
「③洗濯」→洗濯槽の蓋が開けられたら→「一時停止」→蓋が閉じられたら→「③洗濯」
「⑤脱水Ⅰ」→洗濯槽の蓋が開けられたら→「一時停止」→蓋が閉じられたら→「⑤脱水Ⅰ」

参 考　条件制御で使用する代表的なラダー命令語･･･「STP」「STE」「CJ」「NCJ」「LABEL」など

● 時限制御

時限制御は、制御対象への動作指令がタイマなどを使った時間経過や時刻によって終了し、あらかじめ定められた順序に従って、次の動作を開始する制御です。
〔全自動洗濯機のシーケンス制御における時限制御〕
「③洗濯」→規定時間の経過を検出→「④排水Ⅰ」
「⑤脱水Ⅰ」→規定時間の経過を検出→「⑥給水Ⅱ」

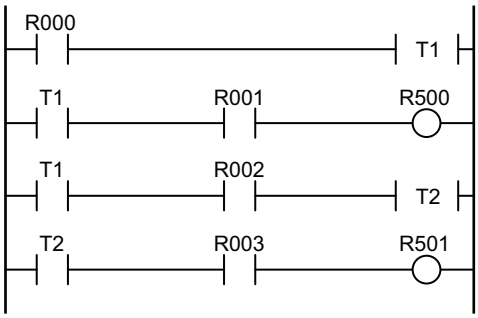
参 考　時限制御で使用する代表的なラダー命令語･･･「TMR」「LDWK」など

4-2 プログラミング言語

《KV STUDIO》では、「ラダープログラム」と「ST言語」、「KVスクリプト」を使用してプログラムを作成できます。
作成したプログラムをCPUユニットで動作させるための言語として、「ニモニック」が使用されます。また、16ビット/32ビットや符号付き/符号なしなどは命令語ではなく、サフィックスを使用して使い分けます。

ラダープログラム

ラダー回路とは、実体配線図を図式化したもので、《KV STUDIO》で書かれたラダー回路をラダープログラムといいます。
《KV STUDIO》では、汎用のスイッチやランプの他に、ON時間、OFF時間を制御する便利な命令語や、PID制御なども命令語として用意しており、組み合わせて記述することで、シーケンス制御を簡単に記述できます。

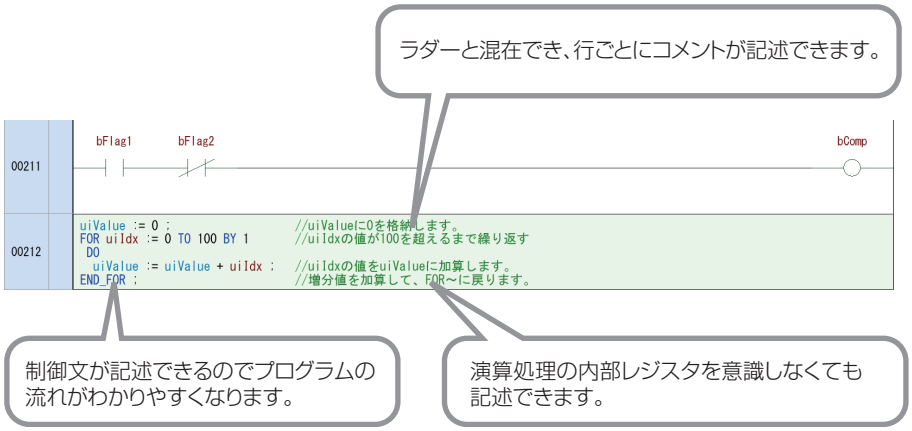


さらに、《KV STUDIO》では、拡張ラダーモードが使用でき、ラダーの行数を削減したり、処理順序を明確にしたりできるので、視認性が向上できます。

「拡張ラダーモード」(4-258ページ)

ST言語

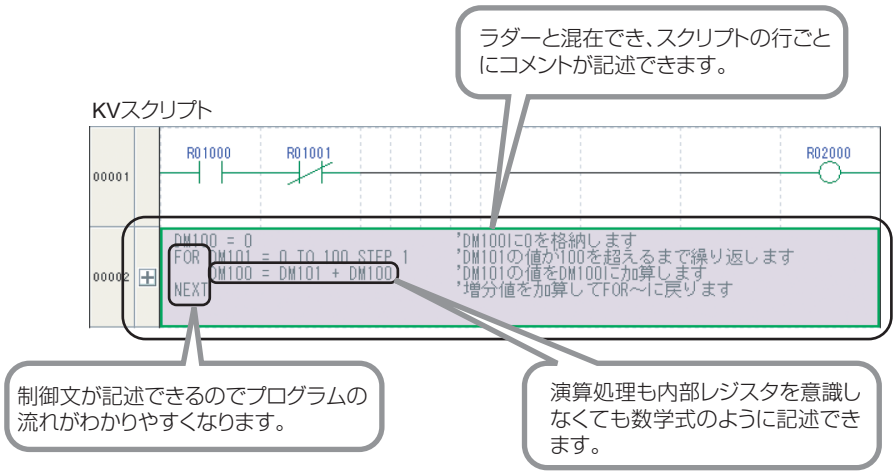
ラダープログラムは、シーケンス制御の記述に適している一方、数値を扱う複雑な演算処理や文字列を扱う処理については、記述が複雑になり、デバッグやメンテナンスも困難になりがちです。
ラダープログラムでは記述が複雑になる部分をST言語で代替記述して、効率よくプログラムを作成できます。



ポイント **ST言語を使用するには、CPU機能バージョン2.1以降である必要があります。**

KVスクリプト

ラダープログラムは、シーケンス制御の記述に適している一方、数値を扱う複雑な演算処理や文字列を扱う処理については、記述が複雑になり、デバッグやメンテナンスも困難になりがちです。
KVスクリプトは、これらラダープログラムの問題点を補うために開発されたプログラミング言語です。ラダープログラムでは記述が複雑になる部分をスクリプトで代替記述して、効率よくプログラムを作成できます。



参 考　スクリプトプログラムはラダープログラムに変換されてから実行されます。
スクリプトプログラムの動作確認などは、変換されたラダープログラムでモニタする事ができます。

『KV-8000/7000/5000/3000/1000シリーズ･KV Nanoシリーズスクリプトプログラミングマニュアル』

ニモニック

ニモニックは、作成したラダープログラムをCPUユニットで動作させるための言語です。
拡張ラダーは、ニモニックをシンボル化することによって、視覚的にわかりやすく表現したものです。
拡張ラダーで作成したプログラムは、CPUユニットへの転送時にニモニックリストへ変換されます。
ニモニックリストは《KV STUDIO》上で編集できますが、通常のラダープログラム作成時に意識する必要はありません。

LD	R000
OUT	R500
LD	R002
OUT	R501
LD	R003
OUT	R502

参 考

- ニモニックリストを編集するには、メニューから【編集(E)】→【リスト編集(L)】を選択します。
  別手順 Ctrl + D
- 本マニュアル内では、CPU 内蔵機能や命令語のサンプルプログラムにニモニックリストを記載しています。このニモニックリストを使用すると、サンプルプログラムを簡単に入力できます。
  「ニモニックリストの使い方」(17ページ)
- 拡張ラダーでプログラムを作成する時に、各命令をニモニックで入力することでプログラム作成工数を短縮できます。
  「ニモニック入力」4-238ページ

サフィックス

サフィックスとは命令語やデバイスの接尾語としてつくもので、使用すると、デバイスのデータ型の扱いを変更できます。以下の6種類があります。

□.U　□.S　□.D　□.L　□.F　□.DF　□.A

※ □にはサフィックス対応の命令語が入ります。

例

MOV.(U)
MOV.S
MOV.D
MOV.L
MOV.F
MOV.DF
MOV.A

□.U　16ビット符号無しBINデータとして扱います。(0～65535)
□.S　16ビット符号付きBINデータとして扱います。(－32768～＋32767)
□.D　32ビット符号無しBINデータとして扱います。(0～4294967295)
□.L　32ビット符号付きBINデータとして扱います。(－2147483648～＋2147483647)
□.F　単精度浮動小数点型実数データとして扱います。
(－3.4E＋38≦ N ≦－1.4E－45
N ＝ 0
＋1.4E－45≦ N ≦＋3.4E＋38
(有効桁数：約7桁))
□.DF 倍精度浮動小数点型実数データとして扱います。
(－1.79E＋308≦N≦－2.23E－308
N ＝ 0
＋2.23E－308≦ N≦＋1.79E＋308
(有効桁数：約16桁))
□.A　オペランドに使用されている変数のデータ型によって、命令語が扱うデータ型が変わります。また、.Aは一部の命令語のみ対応しています。詳細は、『命令語リファレンスマニュアル』を参照してください。

□.Uは、《KV STUDIO》上では表示が省略されます。また、サフィックスを持たない命令は、16ビット符号無しBINデータとして扱います。
サフィックス対応命令は、BINデータをサフィックスに応じた10進表現の値で実行します。
したがってBINデータが同じでもサフィックスが異なることにより、実行命令結果が異なることがあります。

ポイント　**.AはKV-8000シリーズCPUユニットのCPU機能バージョン2.3以降で使用できます。**

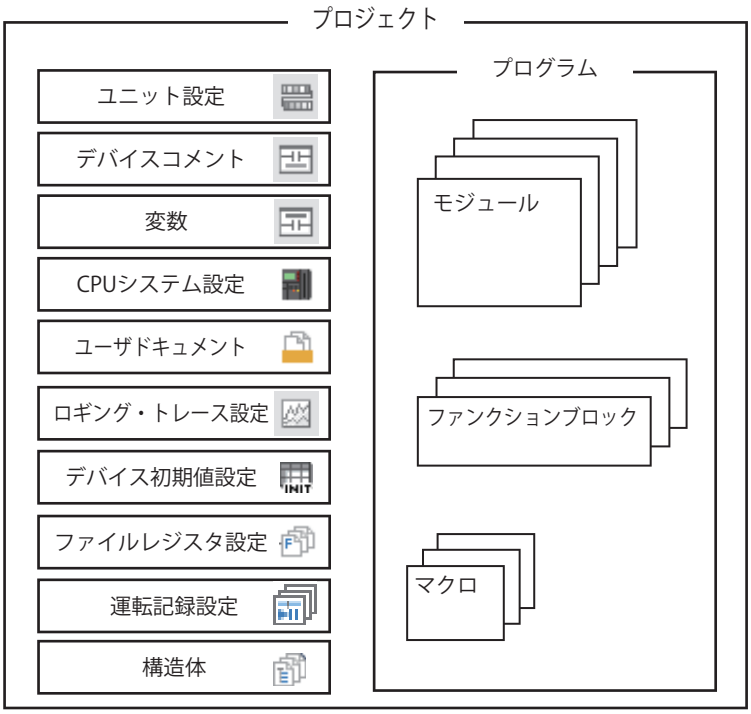
MEMO

4-3 プロジェクト

プロジェクトの構成

■ プロジェクトとは

プログラムや各ユニット設定など、《KV-8000シリーズ》を運転するために必要な全ての情報をプロジェクトという単位で管理します。
《KV STUDIO》でプロジェクトを新規作成すると、プロジェクト名と同じ名称のフォルダが自動的に作成されます。
プロジェクトに含まれる情報は次のとおりです。



ポイント **1つのプロジェクトで管理できるのは、1台の《KV-8000シリーズ》に関する情報のみに限られます。**

● プログラム

プログラムは、モジュール、ファンクションブロック、マクロで構成されます。
このプログラムを繰り返し実行することで、さまざまな制御が可能になります。

「4-8 プログラムの構成と動作」(4-82ページ)
「4-9 モジュール」(4-98ページ)
「4-10 ファンクションブロック」(4-117ページ)
「4-11 マクロ」(4-178ページ)

● ユニット設定

ユニット構成や各ユニットの詳細設定を管理します。《KV STUDIO》で簡単に設定できます。

『各拡張ユニットのユーザーズマニュアル』
『KV STUDIOユーザーズマニュアル』「3章 ユニットエディタ」

● デバイスコメント

各デバイスにコメントを付けて管理することで、各デバイスに割り当てた機能が把握しやすくなります。

「デバイスコメント」(4-252ページ)

● 変数

変数の名称とデータ型の設定を行います。変数を使用すると、定義した任意の名称を使用してプログラミングできるため、プログラムの可読性が向上します。

「4-5 変数」(4-55ページ)

● 構造体

構造体の設定を行います。構造体を使用すると、同じ構造を持つ変数を一括で登録・モニタできるようになります。

「 構造体」(4-59ページ)

● CPUシステム設定

以下の項目を設定します。

分類	項目	内容	参照ページ
プログラムの設定	モジュール実行順設定	各モジュールのスキャン内での実行順序を設定します。	「モジュールの実行順序」(4-109ページ)
	定周期実行とユーザ割り込み設定	定周期モジュールと、ユーザ割り込みの割り込み優先度を設定します。	「スキャンタイム」(4-90ページ)「割り込みの優先度」(4-209ページ)
	スキャンタイム関連の設定	一定スキャンタイム運転とEND処理時間を設定します。	「スキャンタイム」(4-90ページ)
	プロテクト設定	プロジェクトの読み書きに対するプロテクト設定をします。	「プロジェクトのプロテクト設定」(4-16ページ)
	その他の設定	プロジェクトで使用する言語を設定します。	『KV STUDIOユーザーズマニュアル』「プロジェクト言語設定」
システムの設定	ローカル全体割付の設定	ローカルデバイスの全体割り付けと各プログラム別の個別設定をします。	「ローカルデバイスの割付」(4-202ページ)
	CPUユニットバッファ容量設定	ロギング・トレース機能使用時のバッファ容量を設定します。	「ロギング/トレース設定(オプション)」(5-67ページ)
	デバイス記録バッファ容量設定	運転記録機能使用時のデバイス記録バッファ容量を設定します。	「デバイス記録バッファ容量設定」(5-97ページ)
	停電保持設定	電源断時にデバイスの状態や値を保持するデバイスを設定します。	「停電保持設定(ラッチ)」(4-26ページ)
	イベント/エラー設定	エラー発生時の運転/停止状態や、デバイス値変更イベントの記録に関する設定をします。	「イベント/エラー設定」(4-86ページ)
	デジタルトリマ上限値の設定	デジタルトリマの上限値と上限値設定の有効/無効を設定します。	「デジタルトリマ」(5-130ページ)
	アクセスウィンドウ設定	アクセスウィンドウで表示する言語や表示するコメント番号などを設定します。	「アクセスウィンドウ」(5-113ページ)
	その他の設定	VT5-WXのラダーモニタ機能で使用するデータの転送に関する設定およびEtherNet/IPのコネクションに変数を割り付けるためのダイアログを表示するかの設定を行います。	『KV STUDIOユーザーズマニュアル』「その他の設定」

● ロギング/トレース設定

デバイスデータのロギング･トレース機能を設定します。

「5-5 ロギング･トレース」(5-57ページ)

● デバイス初期値設定

運転開始時に、指定したデバイスに登録した値で自動的に初期化できるデバイス初期値を設定します。

「デバイス初期値」(4-257ページ)

プログラム容量

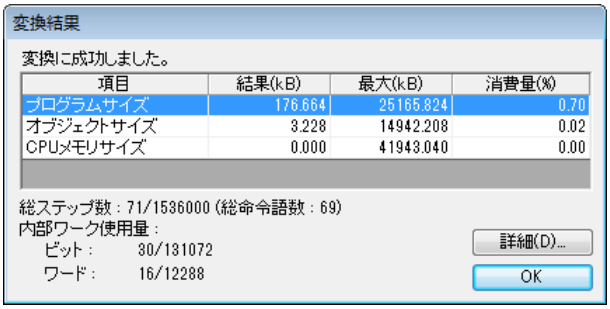
最大プログラム容量は下記のとおりです。

項目		KV-8000(A)
プログラムサイズ		64MB※1
オブジェクトサイズ		14.25MB
1モジュールのサイズ		16MB※2
モジュール個数		4000個
ファンクションブロック個数 + マクロ個数		4000個
モジュール実体個数		4000個
ステップ数		1500k step
内部ワーク使用量	ビット	131072ビット
	ワード	12288ワード
グローバルデバイスコメント･ローカルデバイスコメントの最大サイズ		制限なし
ユニット個別の設定データ･記録データの合計最大サイズ		3136KB
回路ブロック転送時の伝送データサイズ		384KB
構造体定義のサイズ		2MB

※1 CPUメモリ容量設定でサイズは変更できます。
※2 CPU機能バージョン2.0以降の場合です。CPU機能バージョン1.0の場合は4MB。
プログラム容量の使用割合は、プログラム変換時に《KV STUDIO》が自動的に計算して［変換結果］ダイアログに表示するため、計算や管理は必要ありません。

CPUメモリ容量設定によって、最大プログラム容量が設定できます。

「CPUメモリについて」(5-29ページ)



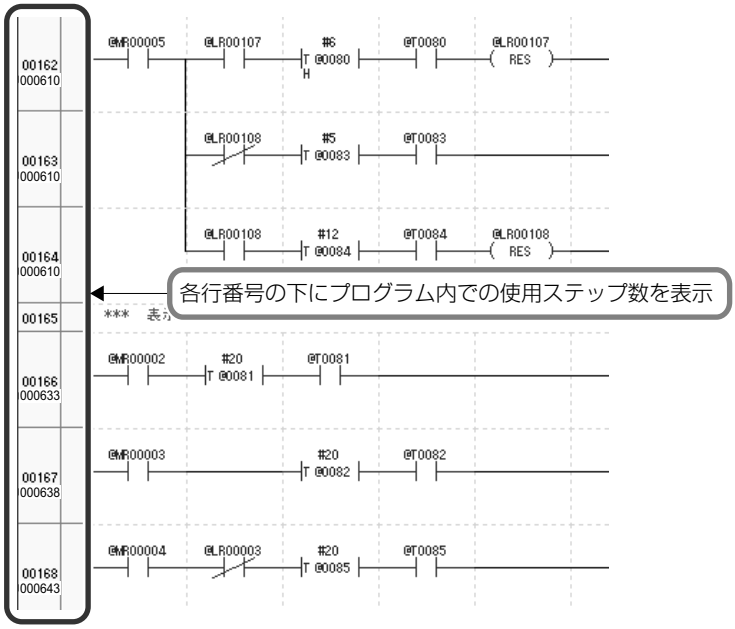
ポイント

ステップ数、プログラムサイズ、オブジェクトサイズ、内部ワーク使用量のいずれかが最大容量を超えると、プログラムの転送･運転ができません。

参考

<プログラムサイズ>
各命令語ごとにバイトサイズが決められており、オペランドの種別や命令語のサフィックス、インデックス修飾の有無で消費するサイズが変化します。
プログラムサイズの計算には、各命令で消費するバイトサイズの合計の他に、行コメント･デバイスコメント･変数･デバイス初期値･ユニット設定情報･運転記録設定･ロギング･トレース設定情報･CPUシステム設定で使用するバイト数が加算されます。デバイスコメントと行コメントと変数は文字数によりサイズが変化します。
上記の結果は初期値での場合であり、CPUメモリ容量設定を変更すると、管理できるデバイスコメント、変数の量も変化します。

＜総ステップ数＞
総ステップ数は、プロジェクト内で使用している命令語･行コメントの合計数です。
使用ステップ数は、《KV STUDIO》で変換後にラダー図左端に表示されます。



＜内部ワーク使用量＞
各命令ごとにオペランドに指定しているデバイスの前回状態など内部で使用するデータの合計です。

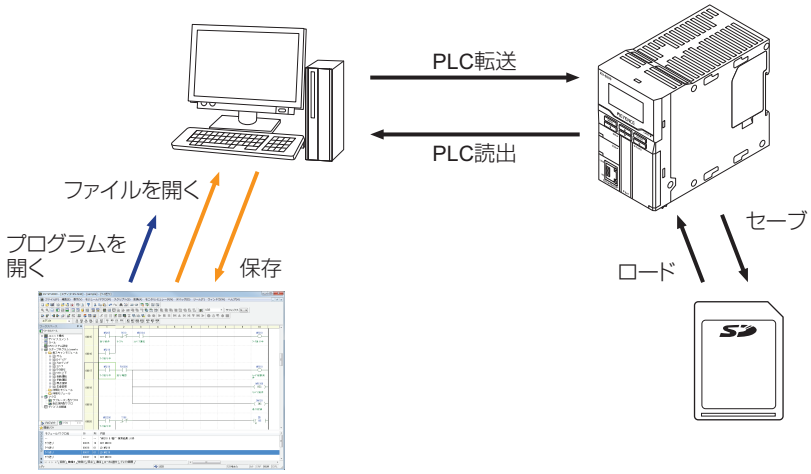
プロジェクトのプロテクト設定

■ プロテクト設定の概要

《KV STUDIO》で作成したプロジェクトにプロテクトを設定できます。
プロテクト設定は《KV STUDIO》の「CPUシステム設定」で設定してください。

項目	説明
ライトプロテクト	CPUユニットに格納されているラダープログラムがライトプロテクト(書き込み時に確認ダイアログを表示)されます。誤って上書きすることを防止できます。
リードプロテクト	CPUユニットに格納されているラダープログラムがリードプロテクト(読み出し不可)されます。プログラム資産の流出を防止できます。
プロジェクトパスワード	CPUユニットに格納されているラダープログラムを読み出す際にパスワード入力ダイアログが表示されます。 ※リードプロテクト設定時は、パスワード入力ダイアログも表示されず、読み出しできません。
プログラムパスワード	プロジェクト編集時に、プログラムを表示しようとすると、[パスワード解除]ダイアログが表示されます。
プロジェクトロード禁止	メモリカードまたはCPUメモリからのプロジェクトロードを禁止します。

ポイント **プロジェクトロード禁止機能はKV STUDIO Ver.11.63以降、システムプログラムバージョン、2.7以降のCPUユニットと組み合わせた場合のみ使用できます。**

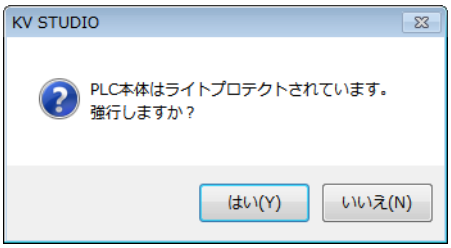


● プロジェクトをパソコン上で編集する場合

- ファイルを開く場合
  プロテクト設定はできません。
- ファイルを保存する場合
  プロテクト設定はできません。
- プログラムを開く場合
  プログラムパスワードを設定して、開く際に、[パスワード解除]ダイアログを表示できます。

● プロジェクトをパソコンからCPUユニットに読み書きする場合

- 読み出しを実行する場合1
  リードプロテクトを設定して読み出しを禁止できます。
  プロジェクトパスワード設定時も、確認ダイアログは表示されません。
- 読み出しを実行する場合2
  プロジェクトパスワードを設定して[パスワード解除]ダイアログを表示できます。
- 書き込みを実行する場合
  ライトプロテクトを設定して上書き時に[確認]ダイアログを表示できます。

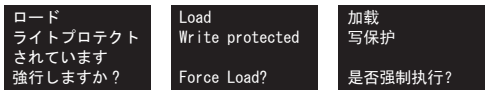


参 考

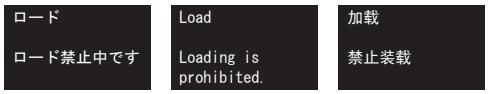
プロジェクトパスワード設定時は、パスワード認証に失敗すると、次回読み出しが可能になるまでに一定時間待つ必要があります。また待ち時間は認証失敗のたびに延長されます。パスワード認証に失敗したため、読み出し認証ができない場合、プロジェクトパスワード認証禁止状態リレー(CR2314)がONしています。
パスワード認証に連続で失敗した回数、および累積の回数は、プロジェクトパスワード連続認証失敗回数(CM1658)とプロジェクトパスワード累積認証失敗回数(CM1659)に格納されます。プロジェクトパスワード連続認証失敗回数は、認証成功時、プログラム転送正常終了時、オールクリア/初期化実行時にクリアされ、プロジェクトパスワード累積認証失敗回数は、オールクリア/初期化実行時のみクリアされます。

● メモリカードを使用してロード/セーブする場合

- ロードする場合
  ライトプロテクトを設定している場合は、ロード前に確認できます。



プロジェクトロードを禁止している場合はロードに失敗します。



- セーブする場合
  リードプロテクトおよびプロジェクトパスワードを設定すると、セーブは実行できますが、デバイスの値のみセーブし、ラダープログラムはセーブされません。

■ プロジェクトのパスワード

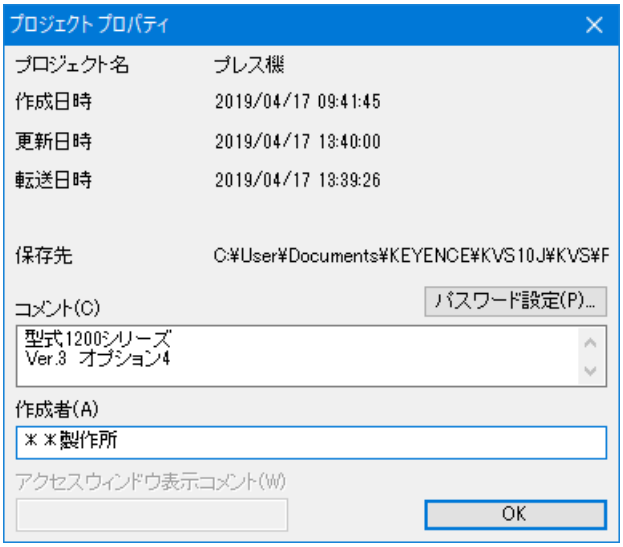
プロジェクトにパスワードを設定して、CPUユニットからの読み出しに制限をかけることができます。
次の手順でパスワードを設定します。

1 **[ワークスペース]を選択します。**

2 **右クリックして表示されるメニューから[プロジェクトプロパティ(F)]を選択します。**

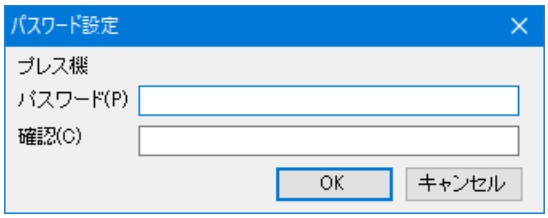
別手順 メニューから【ファイル(F)】→【プロジェクトプロパティ(F)】を選択します。

[プロジェクトプロパティ]ダイアログが表示されます。



3 **〔パスワード設定〕ボタンを押します。**

[パスワード設定]ダイアログが表示されます。



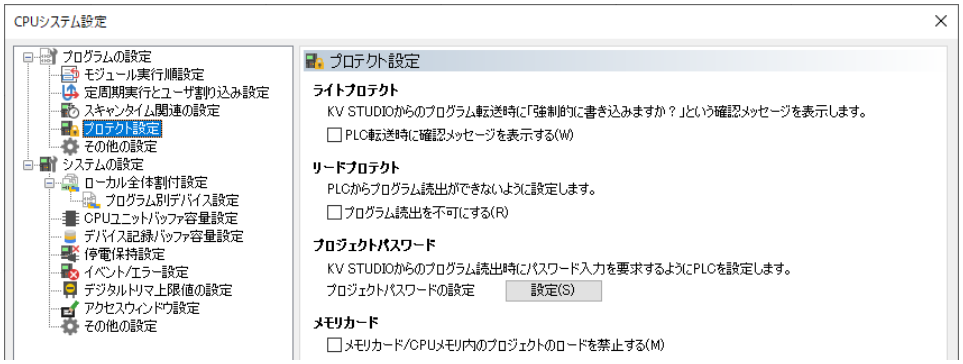
4 **パスワードとパスワードの確認を入力して、〔OK〕ボタンを押します。**

パスワード設定後、パスワード設定したプロジェクトをCPUユニットから読み出す時、[パスワード解除]ダイアログが表示されます。
パスワードを入力しないと、プロジェクトをCPUユニットから読み出せません。
また、パスワードが異なるプロジェクトをPLC転送するときも、PLC内のプロジェクトを保護するため、パスワードを入力するか、全項目を転送する必要があります。

■ プロジェクトのライト/リードプロテクト

プロジェクトにプロテクト設定をして、CPUユニット本体への書き込みや読み出しに制限をかけることができます。

プロテクト設定は、ワークスペースから【CPUシステム設定】→【プロテクト設定】を選択します。



それぞれチェックを入れてCPUユニットに転送することで、プロテクト設定が有効になります。

ポイント

リードプロテクトをかけた状態でCPUユニットに転送すると、その後ラダープログラムの実行状態はモニタできません。
デバッグ完了後にリードプロテクトをかけて、PLC転送を実行してください。
パスワード設定とリードプロテクトを両方使用した場合、パスワード確認ダイアログは表示されず、プロジェクトの読み出しはできません。

参考

- プログラムパスワードについては、「モジュールパスワード」(4-114ページ)、「ファンクションブロックのパスワード設定」(4-141ページ)、「マクロパスワード」(4-191ページ)を参照してください。
- メモリカードのプロテクト設定に関しては、メモリカード/CPUメモリからのプロジェクトロードを禁止できます。ユーザ認証設定と合わせて設定しておくと、KV STUDIOからユーザ認証に成功したユーザのみプロジェクト変更することができます。

4-4 デバイス

KV-8000(A)では、プログラムで用いられる要素として、ユーザが自由に設定できる変数と、リレー、タイマ・カウンタ、データメモリなど、専用の記号と番号で指定できるデバイスを用意しています。どちらも、命令語の後に記述して、命令で使用するデータを与えたり、命令の実行結果を格納するために使用します。この節ではデバイスについて説明します。

デバイス一覧

■ ビットデバイス

デバイス名	範囲※1	点数	主な機能	運転開始時		ローカルデバイス(@)※1	インデックス修飾(:Z/:#)	間接指定(*)※7	
				電源ON	PRG→RUN			参照	格納
入力・出力・内部補助リレー※9	R00000～R199915※9	32000点	CPU内部と拡張ユニットで使用できるデバイス	クリア/保持	クリア/保持	○	○	○	×
リンクリレー	B0000～B7FFF※8	32768点	CPU内部でのみ使用できるデバイス	クリア/保持	クリア/保持	×	○	○	×
内部補助リレー	MR000000～MR399915	64000点		クリア	クリア	○	○	○	×
ラッチリレー	LR00000～LR99915	16000点		保持	保持	○	○	○	×
タイマ(接点)	T0000～T3999	4000点	10μs 、1ms、10ms、100ms の減算タイマ・10ms のアップダウンタイマの接点デバイス	クリア	クリア	○	○	○	×
カウンタ(接点)	C0000～C3999	4000点	加算カウンタ・アップダウンカウンタの接点デバイス	クリア/保持※6	保持	○	○	○	×
コントロールリレー※3	CR0000～CR7915※3	1280点	PLCの機能を制御したり、状態を取得するためのデバイス	クリア	クリア	×	×	×	×

■ ワードデバイス

デバイス名	範囲	点数	主な機能	運転開始時		ローカルデバイス(@)	インデックス修飾(:Z/:#)	間接指定(*)	
				電源ON	PRG→RUN			参照	格納
データメモリ	DM00000～DM65534	65535点	数値データを格納できるデバイス(16ビット)	クリア/保持※2	保持	○	○	○	○
拡張データメモリ	EM00000～EM65534	65535点		クリア/保持※2	保持	○	○	○	○
ファイルレジスタ	連番方式：ZF000000～ZF524287	524288点		クリア/保持※2	保持	×	○	○	○
	カレントバンク：FM00000～FM32767	(32768×16バンク)点				○※10			
リンクレジスタ※8	W0000～W7FFF※8	32768点		クリア/保持※2	保持	×	○	○	○
テンポラリメモリ	TM000～TM511	512点	数値データを格納できるデバイス(16ビット)	クリア	クリア	○	○	○	○
タイマ(現在値/設定値)	T0～T3999	4000点	1ms、10ms、100ms、10μsの減算タイマ･10msのアップダウンタイマの現在値/設定値(32ビット)	現在値：クリア※4 設定値：保持	保持	○	○	○	×
カウンタ(現在値/設定値)	C0～C3999	4000点	加算カウンタ･アップダウンカウンタの現在値/設定値(32ビット)	クリア/保持※6	保持	○	○	○	×
インデックスレジスタ	Z1～Z23	23点	インデックス修飾で使用するデバイス(32ビット)	クリア	クリア	×	×	×	×
コントロールメモリ	CM0000～CM7599	7600点	PLCの機能を制御したり、状態を取得するためのデバイス(16 ビット)	保持	保持	×	×	×	×

※1 ローカルデバイスとして使用するために、ローカルデバイスに予約した個数分、グローバルデバイスの使用可能な範囲が制限されます。
「ローカルデバイス一覧」(4-201ページ)
※2 CPUシステム設定の[停電保持設定]でクリア/保持の設定が可能です。
※3 「CR/CM一覧」(付-10ページ)
※4 アップダウンタイマ(UDT)は、現在値を保持します。
※5 Z11～Z23は、システム予約されているため使用できません。
※6 CPUシステム設定の[停電保持設定]で保持/クリアの設定が可能です。(接点、現在値のみ)
※7 「格納」は、「＊(デバイス)」と記述できるデバイスです。「参照」は、間接指定できるデバイスです。
「間接指定」(4-46ページ)
※8 デバイス番号は、16進数で指定します。
※9 ダイレクト処理する場合は、「DR」で表記します。「DR」で表記する場合は、ローカルデバイスと間接指定が使用できません。
「ダイレクト処理について」(4-83ページ)
※10 ローカルデバイスは、バンク切り換えできません。

■ ファンクションブロック引数

ファンクションブロック内のみで使用される引数は以下のとおりです。

引数タイプ		引数名		説明	間接指定（＊）	ローカルデバイス（@）	インデックス修飾（：）
IN		任意の引数名		デバイスの値または定数を引数として受け取ります。	×	×	×
OUT		任意の引数名		演算結果を指定したデバイスに書き込みます。	×	×	×
IN-OUT		任意の引数名		デバイスを引数として受け取ります。	×	×	○※1
ユニット	先頭R	UR	UR0～UR9	ユニットに割り付けられた先頭リレー（R）を引数として受け取ります。	×	×	○
	先頭DM	UM	UM0～UM9	ユニットに割り付けられた先頭データメモリ（DM）を引数として受け取ります。	×	×	○
	番号	UV	UV0～UV9	ユニットの番号を引数として受け取ります。	×	×	×

※1 配列や構造体を指定した引数は、インデックス修飾を使用できません。
「引数設定」（4-131ページ）

■ マクロ引数デバイス

マクロ内のみで使用されるマクロ引数デバイスは以下のとおりです。

属性		記号	範囲	説明	間接指定（＊）	ローカルデバイス（@）	インデックス修飾（：）
デバイス		P	P0～P9	デバイスを引数として受け取ります。	×	×	○
値		V	V0～V9	デバイスの値または定数を引数として受け取ります。	×	×	×
ユニットデバイス	先頭R	UR	UR0～UR9	ユニットに割り付けられた先頭リレー（R）を引数として受け取ります。	×	×	○
	先頭DM	UM	UM0～UM9	ユニットに割り付けられた先頭データメモリ（DM）を引数として受け取ります。	×	×	○
	番号	UV	UV0～UV9	ユニットのユニット番号を引数として受け取ります。	×	×	×

「引数設定」（4-186ページ）

■ 定数

定数	範囲	主な機能
10進数(#)	#0～#65535(16ビット符号無し) －32768～＋32767(16ビット符号有り) #0～#4294967295(32ビット符号無し) －2147483648～＋2147483647(32ビット符号有り)	10進定数を指定する時に使用します。 参考 XYM表示モードにすると「K」で表示します。
単精度浮動小数点型実数	－3.4E＋38≦N≦－1.4E－45 N＝0 1.4E－45≦N≦＋3.4E＋38 (有効桁数：約7桁)	単精度浮動小数点型定数を指定する時に使用します。
倍精度浮動小数点型実数	－1.79E＋308≦N≦－2.23E－308 N＝0 2.23E－308≦N≦1.79E＋308 (有効桁数：約16桁)	倍精度浮動少数点型実数を指定する時に使用します。
16進数($)	$0000～$FFFF(16ビット) $00000000～$FFFFFFFF(32ビット)	16進定数を指定する時に使用します。 参考 XYM表示モードにすると「H」で表示します。
固定文字列(" ")	(例)　"型式ABC"、"月／08:30:15"	固定文字列を指定する時に使用します。

■ 内部レジスタ

デバイス名	範囲	点数	主な機能	運転開始時(電源ON、PRG→RUN)
内部レジスタ	－	1点	データを一時的に格納するデバイス(16ビット／32ビット／64ビット)	クリア

デバイスの表記方法

● インデックス修飾(インデックスレジスタによる修飾、インデックス定数による修飾)の記述

・インデックスレジスタによる修飾

デバイス(R(DR) / MR / LR / B / T / C / DM / EM / FM / ZF / W / TM / ＊DM / ＊EM / ＊FM / ＊ZF / ＊ W / ＊TM)の後に : Z□(□ : 1～10の数値) を続けて入力します。

例

DM0:Z01	**(Z1の値が100なら、DM100を意味します)**
R1000:Z01	**(Z1の値が100なら、R1604を意味します)**

・インデックス定数による修飾

デバイス(R(DR) / MR / LR / B / T / C / DM / EM / FM / ZF / W / TM / ＊DM / ＊EM / ＊FM / ＊ZF / ＊ W / ＊TM)の後に :#□(□ : －524287～524287) を続けて入力します。

例

DM0:#1	**(DM1を意味します)**
R1000:#1	**(R1001を意味します)**

・演算式によるインデックス修飾

デバイス(R(DR) / MR / LR / B / T / C / DM / EM / FM / ZF / W / TM / ＊DM / ＊EM / ＊FM / ＊ZF / ＊ W / ＊TM)の後に演算式を続けて入力します。

例

DM0:(DM100＋20)	**(DM100の値が50なら、DM70を意味します)**
R1000:(DM0＊4)	**(DM0の値が2なら、R1008を意味します)**

ポイント

演算式によるインデックス修飾はKVスクリプト内でのみ使用できます。
『KV-8000/7000/5000/3000/1000・KV Nanoシリーズスクリプトプログラミングユーザーズマニュアル』

● 間接指定(＊)の記述

デバイス(DM/EM/FM/ZF/W/TM)の前に ＊ を入力します。間接指定命令とあわせて使用します。

例

＊DM0　＊TM100

「間接指定」(4-46ページ)

● ローカルデバイスの記述

デバイス(R/MR/LR/T/C/DM/EM/FM/TM)の前に @ を入力します。
プログラム内のみで有効なデバイスです。間接指定やインデックス修飾も可能です。

例

＠DM0　＠R10000

「4-12 ローカルデバイス」(4-200ページ)

● XYM表記について

デバイス(R(DR)/MR/LR/DM/EM/FM)については、下表のXYM表記によるデバイスの指定が可能です。

デバイス名	キーエンス表記	XYM表記	XYM表記時のデバイス番号
リレー	R(DR)	X(DX)	0000～1999F※1
		Y(DY)	0000～1999F※1
内部補助リレー	MR	M	00000～63999
ラッチリレー	LR	L	00000～15999
データメモリ	DM	D	00000～65534
拡張データメモリ	EM	E	00000～65534
ファイルレジスタ	FM	F	00000～32767

※1　入出力が割り当たらないRについてはXYM表記時も0000～199915のデバイス番号で表示されます。

XYM表記にするには、《KV STUDIO》のメニューから【ツール】→【オプション】を選択して、[表示モードの設定]で「XYMで表示」にチェックを付けます。

● ワードデバイスのビット指定の記述

デバイス(DM / EM / FM / ZF / W / TM / CM)の後に.□(□:0~15の数値)を続けて入力します。

例

DM0.0　　(DM0の1ビット目を意味します)
EM10.4　　(EM10の5ビット目を意味します)

「ワードデバイスのビットデバイス扱い」(4-52ページ)

● サフィックスの記述

デバイス(R(DR)/ MR / LR / B / T / C / DM / EM / FM / ZF / W / TM / CM / ＊DM / ＊EM / ＊FM / ＊ZF / ＊ W / ＊TM)の後に.□(□:U,S,D,L,F,DF,B,T)を続けて入力します。

例

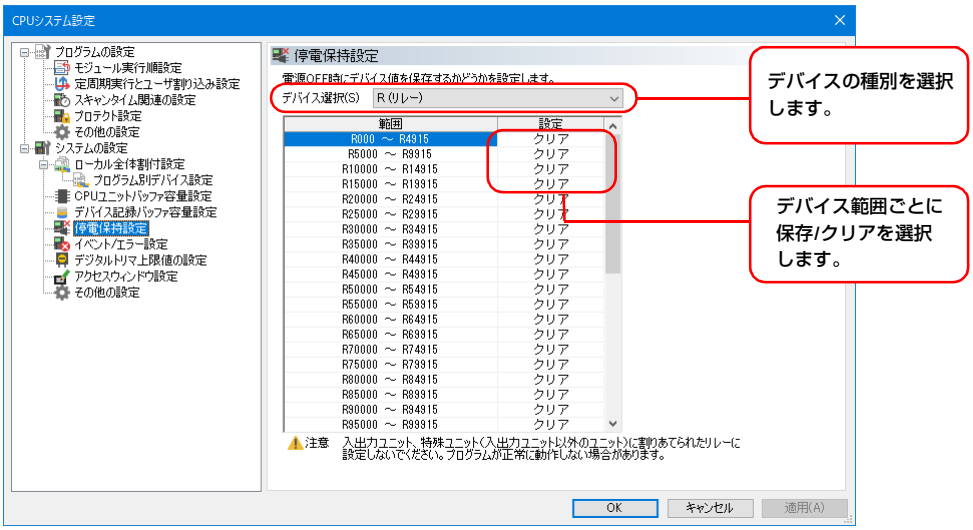
DM0.F　　(DM0～DM1の値を浮動小数点型データとして演算します)
R1000.U　　(R1000～R1015までの16ビットデータとして演算します)

ポイント

サフィックスの記述はST/スクリプト内でのみ使用できます。
『KV-8000シリーズST言語プログラミングマニュアル』
『KV-8000/7000/5000/3000/1000・KV Nanoシリーズスクリプトプログラミングユーザーズマニュアル』

停電保持設定（ラッチ）

停電保持機能は、電源が切断された場合もデバイスの内容を、電源切断前の状態で保持するか、クリアするかを設定する機能です。
設定は、《KV STUDIO》のワークスペースからCPUシステム設定を開いて、【システムの設定】→【停電保持設定】で設定します。



ビットデバイス

入力/出力リレー　R

概要 入力リレーは、外部機器からPLCにON/OFF情報を取り込むためのデバイスです。
出力リレーは、PLCからPLCの外部機器へON/OFF指令や演算結果を送り出すためのデバイスです。

表記 デバイス番号は、次のような構成になっています。
（例）

チャンネル番号（000～1999）接点番号（00～15）

R01012

デバイス名　デバイス番号（00000～199915）

参考
- R を省略して記述できます。（ただし、定数が入力可能なオペランドでは、R を省略した記述はできません。）
- ゼロサプレスして記述できます
  （例）R12
- ワードデバイスとして扱えます。
  「ビットデバイスのワード単位扱い」（4-50ページ）
- デバイス名に「D」を付けるとダイレクト入出力リレーとして動作します。
  （ダイレクト入出力に対応していないデバイスの場合、「D」をつけても通常のリレーとして動作します。）
  （例）DR5000･･･R5000はダイレクト出力リレーとして動作します
  「ダイレクト入出力リレーDR」（4-83ページ）
- XYM表示モードにすると、入力リレーを「X」、出力リレーを「Y」で表示します。
  「XYM表記について」（4-25ページ）

範囲 R00000～R199915（拡張入出力ユニットの番号として割り付け可能）

用途 入力リレー
- 接続する外部機器として、ボタンやスイッチ（切り換えスイッチ、リミットスイッチ、光電スイッチ、デジタルスイッチ）などがあります。
- 拡張ユニットに割り付けることで、割り込み･高速カウンタ･位置決め機能の各高速入力として使えます。

出力リレー
- 接続する外部機器として、ソレノイドや電磁開閉器、表示器（信号灯、デジタル表示器）などがあります。
- 拡張ユニットに割り付けることで、高速カウンタ位置決め機能の各高速出力として使えます。

詳細
- 入力端子1点につき入力リレーが1ビット割り当てられます。
- 出力端子1点につき出力リレーが1ビット割り当てられます。
- a接点（ノーマルオープン）、b接点（ノーマルクローズ）のどちらも指定できます。

初期状態 出力リレーは電源ON時、PRG→RUN時に状態がクリアされてOFFになります。

参 考

- リレー(R)のユニット用デバイス表記について
《KV STUDIO》では、通常の表記以外に、ユニット番号を基準としたリレー表記ができます。ユニット用デバイス表記を使用するには、メニューから【表示(V)】→【ユニット用デバイス表示(Y)】を選択するか、ラダー編集ウィンドウ上でShift + Spaceキーを押します。
ユニット用デバイス表記を使用すると、先頭デバイス番号が00000になるため、マニュアルを参照する際にユニットの先頭デバイス番号を考慮する必要がなくなります。

ユニット番号　ユニットリレー番号

R05_00010

チャンネル番号　接点番号

ゼロサプレスして入力できます。
（例）R5_10

ユニット番号 : CPUユニットを先頭に00、01、02、03、04…と、ユニットの接続順に割り付けられます。

ユニットリレー番号 : 00000から占有するリレー点数分がリレー番号として割り付けられます。

ユニット番号	00	01		02	03	04	05
KV-PU1	KV-8000 CPU	KV-SIR32XT 入力32点 出力32点	バス接続	KV-C16XTD 入力16点 出力16点	KV-C32XC 入力32点	KV-C64TD 出力64点	KV-B8XTD 入力8点 出力8点
ユニット リレー番号		00000～00115		00000～00015	00000～00115		00000～00015※1
		00200～00315		00100～00115		00000～00315	00100～00115※1

（例）KV-SIR32XTの入力5点目 ....01_0004
KV-C32XCの入力8点目 .....03_0007
KV-C64TDの出力18点目 .....04_0101
KV-B8XTDの入力4点目 .....05_0003
KV-B8XTDの出力4点目 .....05_0103

(※1)KV-B8XTDの場合、入力リレー、出力リレーを各1CH占有し、前半8ビット分のみが有効となります。

リンクリレー　B

概要 簡易PLCリンクなどでデータリンクをおこなうためのリレーとして使用します。他のリレーと異なり、16進表記でデバイス番号を指定します。

表記 デバイス番号は、次のような構成になっています。
（例）

チャンネル番号（000～7FF）接点番号（0～F）

B1A6F

デバイス名　デバイス番号（0000～7FFF）

参考
- ゼロサプレスして記述できます。
  （例）B7A
- ワードデバイスとして扱えます。
  「ビットデバイスのワード単位扱い」（4-50ページ）

ポイント
- **16進数（0～F）で指定してください。**
- **ローカルデバイス（@）は、使用できません。**

範囲 B0000～B7FFF

用途 EtherNet/IP通信などでデータリンクをおこなうためのリレーとして使用できます。

詳細 拡張ユニットのユニットデバイスに割り付けできません。

初期状態 リンクリレーは、電源ON時、PRG→RUN時に状態がクリアされてOFFになります。

内部補助リレー　R･B･MR･LR

■ R

概要 拡張ユニットリレーに割り当てていないRを内部補助リレーとして使用できます。

表記 デバイス番号は、次のような構成になっています。
（例）

チャンネル番号（000～1999）接点番号（00～15）

R01012

デバイス名　デバイス番号（00000～199915）

参考
- Rを省略して記述できます。
- ゼロサプレスして記述できます。
  （例）R12
- ワードデバイスとして扱えます。
  「ビットデバイスのワード単位扱い」（4-50ページ）

範囲 00000～199915（拡張入力ユニットの番号として割り付けた番号を除く）

初期状態 電源ON時、PRG↔RUN時に状態がクリアされてOFFになります。

■ B

概要 EtherNet/IP通信などでリンクリレーに割り当てていないBを内部補助リレーとして使用できます。

表記 デバイス番号は、次のような構成になっています。
（例）

チャンネル番号（000～7FF）接点番号（0～F）

B1EE7

デバイス名　デバイス番号（0000～7FFF）

参考
- ゼロサプレスして記述できます。
  （例）B7A
- ワードデバイスとして扱えます。
  「ビットデバイスのワード単位扱い」（4-50ページ）

ポイント
- **16進数（0～F）で指定してください。**
- **ローカルデバイス（@）は、使用できません。**

範囲 B0000～B7FFF

詳細 拡張ユニットのユニットデバイスに割り付けできません。

初期状態 電源ON時、PRG↔RUN時に状態がクリアされてOFFになります。

■ MR

概要 CPU内部のみで使用できるデバイスです。

表記 デバイス番号は、次のような構成になっています。
（例）

チャンネル番号（0000～3999）接点番号（00～15）

MR002809

デバイス名　デバイス番号（000000～399915）

詳細 拡張ユニットのユニットデバイスに割り付けできません。

初期状態 電源ON時、PRG↔RUN時に状態がクリアされてOFFになります。

参 考
- MRのRを省略して入力できます。
- ゼロサプレスして記述できます。
  （例）M2809
- XYM表記モードにすると、「M」で表示します。
  「XYM表記について」（4-25ページ）

■ LR（ラッチリレー）

概要 CPU内部のみで使用でき、状態を保持するデバイスです。

表記 デバイス番号は、次のような構成になっています。
（例）

チャンネル番号（000～999）接点番号（00～15）

LR02308

デバイス名　デバイス番号（00000～99915）

範囲 LR000～LR99915

詳細 拡張ユニットのユニットデバイスに割り付けできません。

初期状態 状態が保持されます。

参 考
- LRのRを省略して入力できます。
- ゼロサプレスして記述できます。
  （例）L2308
- XYM表記モードにすると、「L」で表示します。
  「XYM表記について」（4-25ページ）

タイマ（接点） T

概要 タイマは、接点（1ビット）、設定値（32ビット）、現在値（32ビット）を持ちます。減算式タイマの現在値が0と一致（タイムアップ）、もしくは加算・減算式タイマの現在値が設定値または0と一致すると、接点がONするデバイスです。

表記 デバイス番号は、次のような構成になっています。
（例）

デバイス番号（0000～3999）

T0501

デバイス名

接点、現在値、設定値は使用する命令語のオペランドによって決まります。

範囲 T0000～T3999

詳細 タイマには1msタイマ・10msタイマ・100msタイマ・10μsタイマ・アップダウンタイマの5種類があります。

初期状態 接点は電源ON時、PRG→RUN時に状態がクリアされてOFFになります。
現在値は電源ON時、PRG→RUN時に設定値と同じ値になります。
設定値は保持されます。

参考
- ゼロサプレスして記述できます。
（例）T501

■ タイマを動作させる時の記述方法

タイマ命令のオペランドに、使用するタイマのタイマ番号と設定値（タイムアップするまでの時間）を指定することで、タイマの動作を記述します。
（例）

デバイス番号（0000～3999）　設定値（0～4294967295）

TMR 0001 #1000
UDT 0002 #1000

タイマ命令

参 考
- 設定値にはデバイスを指定することもできます。
  『KV-8000/7000/5000/3000/1000シリーズ･KV Nanoシリーズ 命令語リファレンスマニュアル』「2-6 タイマ･カウンタ命令」
- タイマとカウンタで同じ番号のデバイスを使用できます。

【タイマの種類による動作の違い】

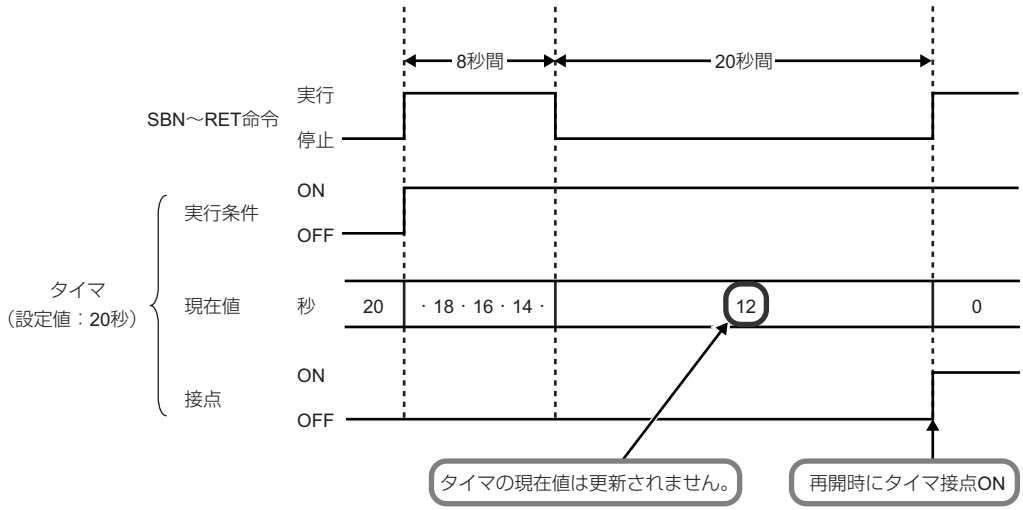
タイマの種類（命令）	デバイス番号	設定値	ONするまでの時間
100ms減算タイマ（TMR）	0000～3999	0～4294967295	n×100ms
10ms減算タイマ（TMH）			n×10ms
1ms減算タイマ（TMS）			n× 1ms
10μs減算タイマ（TMU）			n×10μs
10msアップダウンタイマ（UDT）			n×10ms

■ タイマの誤差

タイマの誤差は、次のとおりです。
- TMR（100msタイマ）：＋（±100ms＋1スキャンタイム）以内
- TMH（10msタイマ）：＋（±10ms＋1スキャンタイム）以内
- TMS（1msタイマ）：＋（±1ms＋1スキャンタイム）以内
- TMU（10μsタイマ）：＋（±10μs＋1スキャンタイム）以内
- UDT（アップダウンタイマ）：＋（±10ms＋1スキャンタイム）以内

■ タイマ命令使用時の注意

サブルーチン命令間（SBN～RET）、ステップ命令間（STP～STE）、条件分岐命令間（CJ/NCJ～LABEL）でタイマ命令を使用する場合、タイマ命令実行中に各命令間の実行条件をOFFすると、タイマの現在値は更新されません。
時間計測は引き続きされており、各命令間の実行が再開した時に経過した時間がタイマの現在値に反映されます。設定時間が経過していた場合には、通常は実行を再開した直後にタイマ接点がONします。ただし、設定時間を経過していても、実行再開までの時間が長い場合は、タイマ接点がONしないことがありますので、タイマ命令実行中に各命令間の実行が止まらない条件で使用してください。
（例）　SBN～RET命令間に設定値が20秒のタイマ命令が記述されている場合



参 考
- ONDL/OFDL/SHOT/FLIK命令も上記と同様に動作するため、注意が必要です。
- プロパティの「実行停止時にOFF処理する」にチェックしているプログラムでタイマ命令を使用した場合、各プログラムの実行停止時にOFF処理によってタイマ命令がリセットされるため、上記の現象は発生しません。

「待機モジュールの動作」（4-101ページ）
「ファンクションブロック」（4-117ページ）
「マクロの種類」（4-181ページ）

■ タイマの設定値・現在値を扱う方法

タイマの設定値・現在値を読み出したり、変更する時に使用する命令語を以下の表に示します。

	読み出し	変更
現在値	MOV、LDA命令	MOV、DW命令
設定値	（読み出しできません）	STA命令※

※　STA命令で変更した設定値は電源ON時、RUN→PRG時にクリアされ、元の値に戻ります。
タイマ命令で設定値をデバイスで指定した場合は、指定したデバイスの値でタイマの設定値が毎スキャン更新されます。

ポイント　**変更時に現在値＞設定値となる場合、現在値は設定値と同じ値に変更されます。**

カウンタ(接点)　C

概要 カウンタは、接点(1ビット)、設定値(32ビット)、現在値(32ビット)を持ちます。
カウンタの現在値が設定値と同じになる(カウントアップする)と、接点がONするデバイスです。

表記 デバイス番号は、次のような構成になっています。
(例)

デバイス番号(0000～3999)

C0001

デバイス名

接点、現在値、設定値は使用する命令語のオペランドによって決まります。

範囲 C0000～C3999

詳細 カウンタにはカウンタ、アウトカウンタ、アップダウンカウンタの3種類があります。

初期状態 デフォルトでは、接点の状態、現在値、設定値ともに保持されます。
CPUシステム設定の[停電保持設定]により電源ON時、PRG→RUN時に接点の状態、現在値、設定値をクリアする設定が可能です。

参考
- ゼロサプレスして記述できます。
  (例) C15
- カウンタとタイマで同じ番号のデバイスを使用可能です。

■ カウンタを動作させる時の記述方法

カウンタ命令のオペランドに、使用するカウンタの番号、設定値、カウント入力リレーを指定することでカウンタの動作を記述します。

デバイス番号(0000～3999)　設定値(0～4294967295)

C 0011 #01000 R03000

カウント入力リレー

OUTC 0012 #01000

UDC 0013 #01000

カウンタ命令

参考 設定値にはデバイスを指定することもできます。

【カウンタの種類による動作の違い】

カウンタの種類(命令)	カウンタ番号の割付	設定値	リセット方法
カウンタ(C)	0000～3999	0～4294967295	実行条件OFF
アウトカウンタ(OUTC)			RES命令
アップダウンカウンタ(UDC)			リセット入力ON

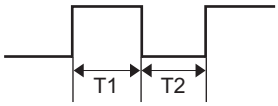
『KV-8000/7000/5000/3000/1000シリーズ ·KV Nanoシリーズ命令語リファレンスマニュアル』
「2-6 タイマ·カウンタ命令」

■ カウンタの最大計数速度

カウント入力のON時間、OFF時間がそれぞれスキャンタイムより長い時のみカウントします。
算出式は以下のとおりです。下記以上の計数速度が必要な時は、高速カウンタや割り込み機能を使ってください。

$$\text{最大計数速度} = \frac{\text{デューティー比（\%）}}{100} \times \frac{1}{\text{スキャンタイム（sec）}} \quad \text{（単位：回／sec）}$$

※信号のONしている時間とOFFしている時間の比率をデューティー比といいます。



$$\text{T1≦T2の場合}\quad \text{デューティー比} = \frac{T1}{T1+T2} \times 100 \quad \text{（単位：\%）}$$

$$\text{T1＞T2の場合}\quad \text{デューティー比} = \frac{T2}{T1+T2} \times 100 \quad \text{（単位：\%）}$$

■ カウンタの設定値・現在値を扱う方法

カウンタの設定値・現在値を読み出したり、変更する時に使用する命令語を以下の表に示します。

	読み出し	変更
現在値	MOV、LDA命令	MOV、DW命令
設定値	（読み出しできません）	STA命令※

※ カウンタ命令で設定値をデバイスで指定した場合、カウンタの設定値は指定したデバイスの値で毎スキャン更新されます。

ポイント **変更時に現在値＞設定値となる場合、現在値は設定値と同じ値に変更されます。**

コントロールリレー（接点） CR

概要 PLCの機能を制御したり、PLCの状態を知るために使用する特殊な内部補助リレーです。

範囲 CR0000～CR7915

詳細 「付-2 CR/CM一覧」（付-10ページ）

ワードデバイス

データメモリ／拡張データメモリ　DM/EM

概要 データメモリと拡張データメモリは、関数が演算する際に参照するデータを保存したり、演算の結果を格納するための1点当たり16ビットのデバイスです。

表記 デバイス番号は、次のような構成になっています。
（例）

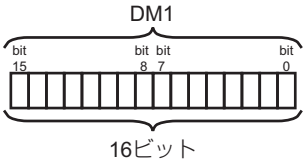
デバイス名　デバイス番号

DM06143

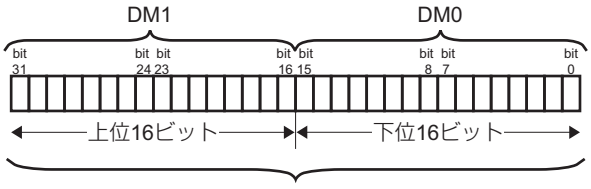
参考
- DM/EMのMを省略して記述できます。
- ゼロサプレスして記述できます。
  （例）D6143
- ビットデバイスとして扱えます。
  （例）DM1100..... DM1100の最下位ビットを扱います
- ビット位置を指定するとビットデバイスとして扱えます。
  （例）DM1200.12 ..... DM1200の13ビット目を扱います
  「ワードデバイスのビットデバイス扱い」（4-52ページ）
- XYM表記モードにすると、「D」で表示します。
  「XYM表記について」（4-25ページ）

範囲 DM00000～DM65534
EM00000～EM65534

詳細
- データメモリ（DM）のみ拡張ユニットのユニットデバイスに割り付けできます。
- デバイス1点のビット幅は16ビットで、符号無し10進数として扱う時は0～65535、符号付き10進数として扱う時は－32768～＋32767、16進数として扱う時は$0～$FFFFの値が扱えます。



- デバイス2点を使用して32ビットのデータを扱うこともできます。32ビットの場合、指定したデバイス番号のデータメモリに下位16ビット、その次のデバイス番号のデータメモリに上位16ビットを格納します。符号無し10進数として扱う時は0～4294967295、符号付き10進数として扱う時は－2147483648～＋2147483647、16進数として扱う時は$0～$FFFFFFFFの値が扱えます。



- 32ビットデータを扱う命令語のオペランドを、データメモリで指定する時は、下位16ビットを格納しているデバイス番号だけを指定します。

初期状態 CPUシステム設定の［停電保持設定］により保持する範囲を設定できます。

参 考

データメモリ(DM)のユニット用デバイス表記について
《KV STUDIO》では、通常の表記以外に、ユニット番号を基準とした表記ができます。
ユニット用デバイス表記を使用するには、メニューから【表示(V)】→【ユニット用デバイス表示(Y)】を選択します。

ユニット番号　デバイス番号
DM05_0010

ゼロサプレスして入力できます。
（例）DM5_10

ユニット番号：　CPUを先頭に00、01、02、03、04･･･と、ユニットの接続順に割り付けられます。

ユニットDM番号：　0000から占有するデータメモリ分が割り付けられます。

ユニット番号	00	01	02		03	04
KV-PU1	KV-8000 CPU	KV-SAD04 60ワード	KV-SDA04 40ワード	バス接続	KV-TP40 90ワード	KV-ML16V 222ワード
ユニット DM番号		0000～0059	0000～0039		0000～0089	0000～0221

（例）KV-SAD04の6番目 ..... 01_0005
KV-TP40の15番目 ..... 03_0014

※ CPU内蔵EtherNet/IP/イーサネット機能使用時のデバイスは、「00_」で始まるユニット用デバイスで表記されます。

ファイルレジスタ　ZF/FM

概要 ファイルレジスタは、すべての領域を連番方式で扱うZFと、領域を16のバンクに分割してバンク切り換え方式で扱うFMがあります。

表記 デバイス番号は、次のような構成になっています。
（例）

デバイス名　デバイス番号

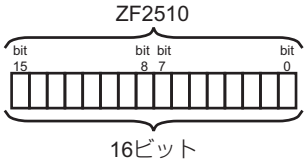
ZF000372

参考
- FMのMを省略して記述できます。
- ゼロサプレスして記述できます。
  （例）ZF372
- ビットデバイスとして扱えます。
  （例）FM1100 ..... FM1100の最下位ビットを扱います
- ビット位置を指定するとビットデバイスとして扱えます。
  （例）ZF1200.12..... ZF1200の13ビット目を扱います
  「ワードデバイスのビットデバイス扱い」（4-52ページ）
- XYM表記モードにすると、「FM」が「F」で表示します。
  「XYM表記について」（4-25ページ）

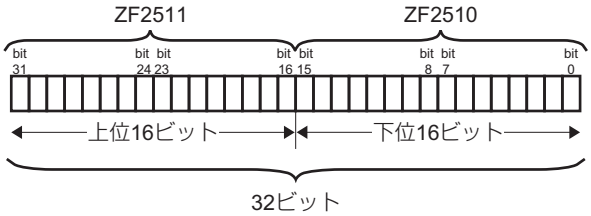
ポイント
- **ZFのローカルデバイス（@ZFxxx）は、使用できません。**
- **FMのローカルデバイス（@FM0～@FM32767）は、バンク切り換えできません。**
- **FMのローカルデバイス（@FM0～@FM32767）は、ワーク領域を使用します。**

範囲 ZF000000～ZF524287
FM00000～FM32767

詳細
- 拡張ユニットのユニットデバイスに割り付けできません。
- 連番方式とバンク切り換え方式では、デバイスの種別と範囲が異なります。
- デバイス1点のビット幅は16ビットで、符号無し10進数として扱う時は0～65535、符号付き10進数として扱う時は－32768～＋32767、16進数として扱う時は$0～$FFFFの値が扱えます。



- デバイス2点を使用して32ビットのデータを扱うこともできます。32ビットの場合、指定したデバイス番号のデータメモリに下位16ビット、その次のデバイス番号のデータメモリに上位16ビットを格納します。符号無し10進数として扱う時は0～4294967295、符号付き10進数として扱う時は－2147483648～＋2147483647、16進数として扱う時は$0～$FFFFFFFFの値が扱えます。

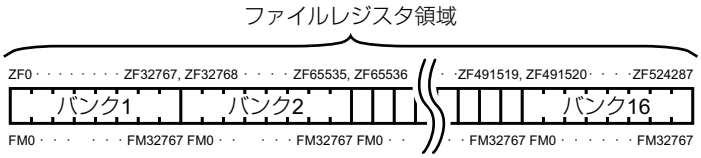


● ZF：連番方式でファイルレジスタを扱うとき

ファイルレジスタの全領域を、1つのデバイス範囲として扱います。

● FM：バンク切り換え方式でファイルレジスタを扱うとき

ファイルレジスタの領域を16分割し、バンクを切り換えて扱います。



バンクを切り換えることで、1つのデバイスを16個の異なるデバイス範囲として使用できます。
ただし、ZFデバイスと重複して使用できるため、データの取り扱いには注意が必要です。
バンクの切り換え方法などファイルレジスタの詳細については、「4-14 ファイルレジスタ」（4-226ページ）を参照してください。

初期状態 デフォルトでは、値が保持されます。
CPUシステム設定の［停電保持設定］により値をクリアする設定が可能です。

リンクレジスタ　W

概要 データリンクをおこなうためのレジスタとして使用します。他のワードデバイスと異なり、16進表記でデバイス番号を指定します。

表記 デバイス番号は、次のような構成になっています。
（例）

デバイス名　デバイス番号（0～7FFF）

W3A6F

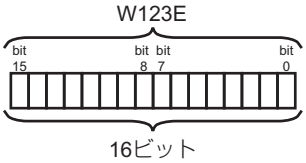
参考
- ゼロサプレスして記述できます。
  （例）W7A
- ビットデバイスとして扱えます。ビット位置の指定も可能です。
  （例）W1200.12..... W1200の13ビット目を扱います
  「ワードデバイスのビットデバイス扱い」（4-52ページ）

ポイント
- **16進数（0～F）で指定してください。**
- **ローカルデバイス（@）は、使用できません。**

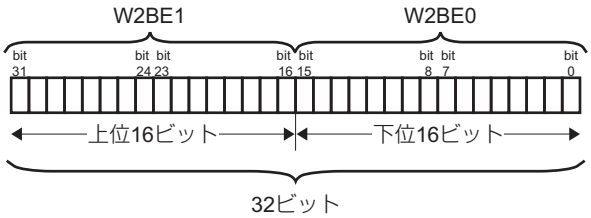
範囲 W0000～W7FFF

用途 データリンクをするためのレジスタとして使用できます。

詳細
- 拡張ユニットのユニットデバイスに割り付けできません。
- デバイス1点のビット幅は16ビットで、符号無し10進数として扱う時は0～65535、符号付き10進数として扱う時は－32768～＋32767、16進数として扱う時は$0～$FFFFの値が扱えます。



- デバイス2点を使用して32ビットのデータを扱うこともできます。32ビットの場合、指定したデバイス番号のデータメモリに下位16ビット、その次のデバイス番号のデータメモリに上位16ビットを格納します。符号無し10進数として扱う時は0～4294967295、符号付き10進数として扱う時は－2147483648～＋2147483647、16進数として扱う時は$0～$FFFFFFFFの値が扱えます。



初期状態 CPUシステム設定の［停電保持設定］により値を保持する設定が可能です。

テンポラリデータメモリ　TM

概要 テンポラリデータメモリは、データメモリと同じように扱うことができる1点当たり16ビットのワードデバイスです。一部のデバイス領域は、演算命令を使う場合に、演算のためのデータや演算の結果データを一時格納するためにも使用します。

表記 デバイス番号は、次のような構成になっています。
（例）

デバイス名　デバイス番号
TM010

範囲 TM000～TM003（演算命令で使用）
TM004～TM511（簡易間接指定などで使用）

TM番号	用途
TM000	演算用（DIV、MUL,ROOT命令）
TM001	
TM002	演算用（DIV命令）
TM003	
TM004～TM511	簡易間接指定・汎用

DIV、MULなどの命令に関する詳細は、『KV-8000/7000/5000/3000/1000シリーズ・KV Nanoシリーズ命令語リファレンスマニュアル』を参照してください。
簡易間接指定の詳細は、「間接指定」（4-46ページ）を参照してください。

初期状態 電源ON時、PRG→RUN時に値をクリアして0にします。

参考
- TM000～TM003はDIV、MULなどの演算命令で使用するため、それ以外の用途で使用しないことをおすすめします。
- ゼロサプレスして入力できます。
  （例）TM10
- TMは、内部キャッシュに格納されるので、高速アクセスできます。

タイマ（現在値・設定値）　T

タイマの設定値（32ビット）の設定、現在値（32ビット）の読み出し時に、ワードデバイスとして使用します。

「タイマ（接点）　T」（4-32ページ）

カウンタ（現在値・設定値）　C

カウンタの設定値（32ビット）の設定、現在値（32ビット）の読み出し時に、ワードデバイスとして使用します。

「カウンタ（接点）　C」（4-35ページ）

インデックスレジスタ　Z

概要 インデックス修飾する際に、対象デバイス番号に加える数を指定するための1点当たり32ビットのワードデバイスです。データメモリと同じように、データを保存したり、演算の結果を格納したりできます。

範囲 Z01～Z23

参考 ゼロサプレスして記述できます。
（例）Z3

ポイント
- **Z11～Z23は、システムで予約されているため使用できません。**
- **ワードデバイスのビット指定はできません。**
- **ローカルデバイスはありません。ZPUSH、ZPOP命令を使用してください。**
  『KV-8000/7000/5000/3000/1000シリーズ・KV Nanoシリーズ命令語リファレンスマニュアル』「ZPUSH命令」
  『KV-8000/7000/5000/3000/1000シリーズ・KV Nanoシリーズ命令語リファレンスマニュアル』「ZPOP命令」

初期状態 電源ON時、PRG→RUN時に値がクリアされて0になります。

「インデックス修飾」（4-44ページ）

コントロールメモリ　CM

概要 PLCの機能の制御、PLCの状態の確認のために使用する特殊なワードデバイスです。

範囲 CM0000～CM7599

詳細 「付-2 CR/CM一覧」（付-10ページ）

インデックス修飾

インデックス修飾とは、各デバイスに、インデックスレジスタ(Z)の値や定数を加算した番号のデバイスを指定する方法です。
加算できる値の範囲は、符号付き32ビットです。
インデックス修飾できるデバイスの種類は、R(DR)、B、MR、LR、T、C、DM、EM、FM、ZF、W、TMで、全てのデバイス範囲をインデックス修飾できます。
間接指定(＊)、ローカルデバイス(@)と併用できます。

ポイント

- **以下の場合はインデックス修飾できません。**
  - **CR/CM/Z**
  - **配列型変数/ラベル**
  - **グローバルデバイスに割り付けていない変数**
  - **ワードデバイスのビット位置指定時**

  「4-5 変数」(4-55ページ)
  「ワードデバイスのビットデバイス扱い」(4-52ページ)
- **インデックス修飾して使用可能範囲外のデバイスを引数に指定した場合は、コントロールリレーCR2012がONし、その命令語は実行されません。運転は継続します。**

■ インデックスレジスタによる修飾

命令語のオペランドが参照するデバイスを運転中に変更する場合に使用します。
インデックスレジスタによるデバイス修飾の記述方法は

(デバイス番号) : Z(1～10)

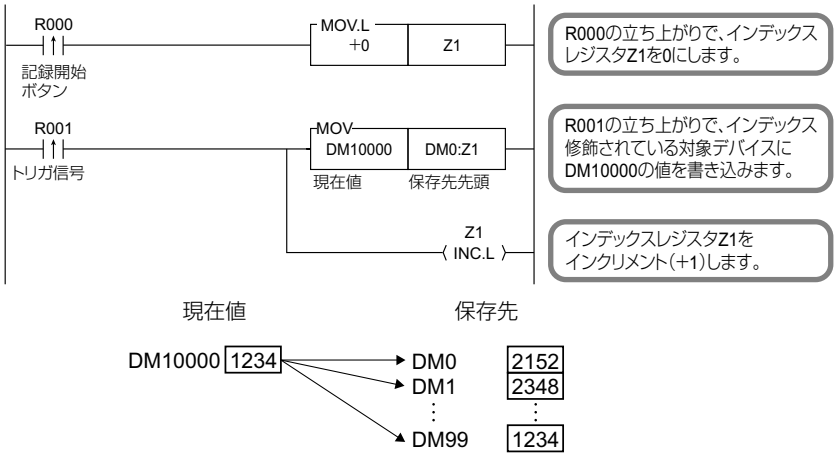
となります。

(例) DM1000:Z1.....Z1の値が－10の場合、DM990を参照します。
R000:Z5 .....Z5の値が＋16の場合、R100を参照します。

ポイント **Z11～Z23は、システム予約されているため使用できません。**

例

DM10000に格納される現在値を、R001の立ち上がりをトリガ信号として保存します。
R000の立ち上がりで記録を開始し、DM0から順番にデータを保存していきます。

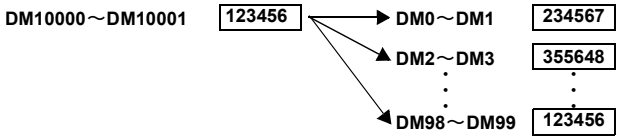


ポイント

インデックス修飾する値は、符号付き32ビットとして扱うため、インデックスレジスタを操作する命令語のサフィックスは「.L」で記述します。

参考

- インデックスレジスタ(Z)をプロジェクト内で 10 個以上使用する必要がある場合は、インデックスレジスタ命令(ZPUSH/ZPOP)を使用して、プログラムごとにインデックスレジスタを一括で退避・復帰させます。
- 『KV-8000/7000/5000/3000/1000シリーズ・KV Nanoシリーズ 命令語リファレンスマニュアル』「3-8 インデックス命令」
- 上記の例で2行目のMOV命令のサフィックスが「.L」の場合、DM10000～DM10001の値をDM0～DM1、DM2～DM3、…、DM98～DM99にデータを保存していきます。



■ インデックス定数による修飾

定数を使用してインデックス修飾する場合の記述方法は

(デバイス番号) : #(10進数表記の定数)

となります。

(例) DM1000:#-10 ...DM1000-10ワードでDM990になります。
R000:#16 .....R000＋16ビット(1チャンネル分)でR100になります。

参考

ユニットデバイスを使用したマクロ内でインデックス定数による修飾を使用すると、割り付けられた先頭デバイスから何番目かがわかりやすくなります。また、マクロ作成時に、引数デバイスとしてUR/UMを使用する場合にも有効です。

(例)

UR0:#2

UM1:#32

「使用例」(4-187ページ)

間接指定

■ 間接指定(＊)

間接指定(＊)は、命令語のオペランドに指定するデバイスをプログラム実行中に変更する場合に有効です。インデックス修飾も参照するデバイスを変更できますが、間接指定(＊)を使用すればデバイスの種類が変更可能です。

● 間接指定とは

各デバイスはCPUユニット内のメモリ上で管理され、それぞれに個別のアドレスを持っています。アドレスは32ビットで管理されるため、アドレス格納用にデバイスが2ワード必要になります。
間接指定(＊)は、そのアドレスを専用命令で操作して参照するデバイスを変更します。
インデックス修飾(: Z/:#)、ローカルデバイス(@)と併用できます。

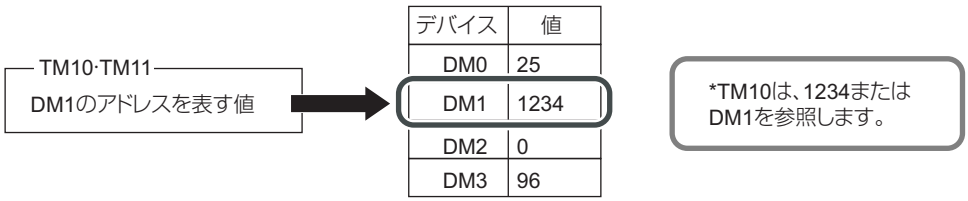
ポイント
- **間接指定(＊)は、簡易間接指定(#TM)と併用できません。**
- **間接指定して使用可能範囲外のデバイスをオペランドに指定した場合は、コントロールリレーCR2012がONし、その命令は実行されません。運転は継続します。**
- **配列の要素および構造体のメンバへの間接指定はできません。**

● 間接指定の記述方法

「＊(デバイス)」と命令語のオペランドに記述すると、そのデバイスに格納されているアドレスに該当するデバイスまたは、デバイス値を参照します。どちらを参照するかは使用する命令語のオペランドによります。

(例)

＜＊TM10と記述した場合＞



ポイント
- **「＊(デバイス)」と記述できる(アドレスを格納できる)デバイスはDM/EM/FM/ZF/TM/Wです。**
- **間接指定できるデバイスは、R/MR/LR/B/T/C/DM/EM/FM/ZF/TM/Wです。**
- **「*(変数)」と記述できる(アドレスを格納できる)変数はUDINT型に設定した変数のみです。**
- **配列の要素、配列の添え字、構造体のメンバは「*(変数)」での間接指定はできません。**

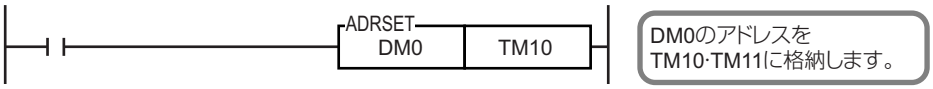
参考
- 間接指定するデバイスがタイマ(T)、カウンタ(C)の場合に、接点、現在値、設定値は使用する命令語のオペランドによって決まります。
- ローカルデバイス使用時の記述は、「＊@(デバイス)」になります。
- 「*(変数)」と記述できる(アドレスを格納できる)変数はUDINT型に設定した変数のみです。
- 配列の要素、配列の添え字、構造体のメンバは「*(変数)」での間接指定はできません。

● 間接指定の使用方法

次の手順で間接指定します。

1 **アドレスセット命令(ADRSET)を使用して、参照するデバイスのアドレスを書き込みます。**

(例)

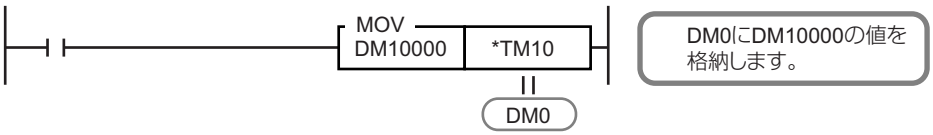


ポイント
- **アドレスを格納できるデバイスはTM、DM、EM、FM、ZF、Wです。**
- **アドレスを格納できる変数はUDINT型の変数のみです。ただし、UDINT型の構造体メンバや、UDINT型の配列の要素には格納できません。**

『KV-8000/7000/5000/3000/1000シリーズ ·KV Nanoシリーズ命令語リファレンスマニュアル』「ADRSET命令」

2 **命令語の各オペランドでデバイスを間接指定するには、アドレスを格納しているデバイスの先頭に「＊」を付けて使用します。**

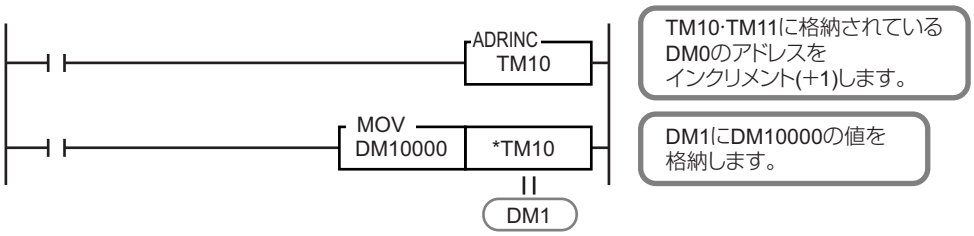
(例)



ポイント
アドレスを格納するデバイスに「＊」を付けて使用する前に、必ずADRSET命令で間接指定するデバイスのアドレスを格納してください。

3 **参照するデバイスを変更する場合は、間接指定専用命令(ADRINC/ADRDEC/ADRADD/ADRSUB)を使用して、格納されているアドレスを変更します。**

(例)

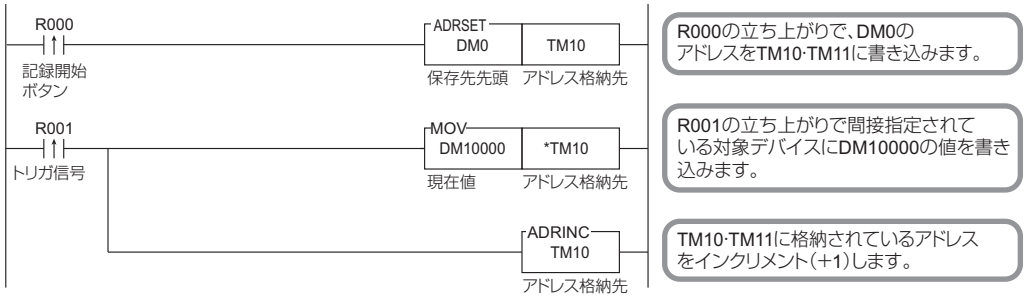


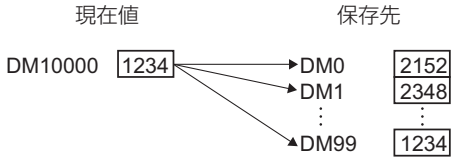
ポイント
参照するアドレスを変更する場合に、通常の算術命令(ADD/INC/SUB/DECなど)を使用しないでください。ラダープログラムが正しく動作しません。

『KV-8000/7000/5000/3000/1000シリーズ ·KV Nanoシリーズ命令語リファレンスマニュアル』「3-9 間接指定命令」

● 間接指定の使用例

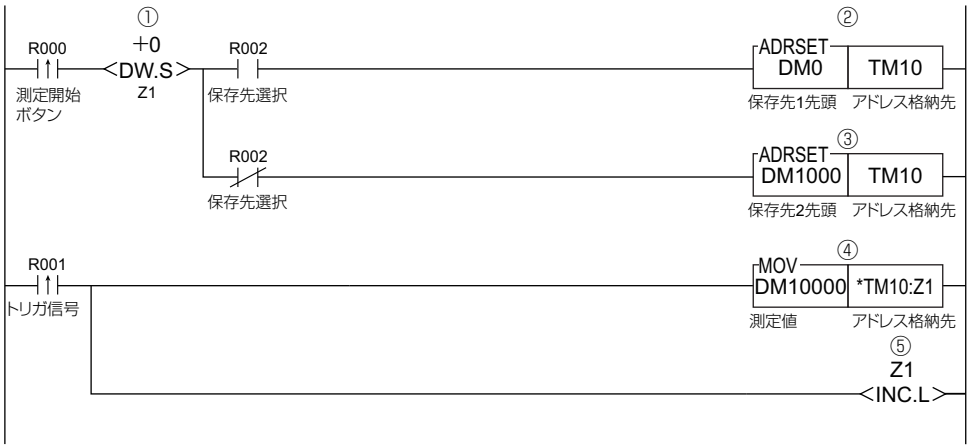
R001の立ち上がりをトリガ信号として、DM10000に格納される現在値を保存します。
R000の立ち上がりで記録を開始し、DM0から順番にデータを保存していきます。





● インデックス修飾との併用例

R001の立ち上がりをトリガ信号として、DM10000に格納される現在値を保存します。
R000の立ち上がりで測定を開始し、測定開始時にR002がONの場合はDM0から、OFFの場合は
DM1000から順番にデータを保存していきます。



[測定開始時]
①インデックスレジスタZ1に0を書き込みます。
②R002がONの場合は、TM10·TM11にDM0のアドレスを書き込みます。
③R002がOFFの場合は、TM10·TM11にDM1000のアドレスを書き込みます。
[トリガ発生時]
④TM10·TM11に格納されているアドレスのデバイスを、Z1でインデックス修飾(Z1の値を加算)した
デバイスにDM10000の値を書き込みます。

(例)
＊TM10→DM1000
Z1＝ 23
→
＊TM10:Z1→DM1023

⑤Z1をインクリメント(＋1)します。

■ 簡易間接指定　#TM

テンポラリデータメモリ（TM）を使用した間接指定は、「簡易間接指定」として使用できます。

ポイント
- **簡易間接指定とインデックス修飾、間接指定（＊）、ローカルデバイス（@）はいずれも併用できません。**
- **簡易間接指定は、過去のプログラム資産を継承するためにサポートしています。新規でプログラムを作成する場合は、インデックス修飾か間接指定を使用されることを推奨します。**

● 簡易間接指定の指定方法

命令語のオペランドに#TMnnnと記述し、TMnnnの中に格納する値を変更することにより、簡易間接指定できます。
TMnnnに格納されている値と簡易間接指定の参照先の関係は、以下の表のとおりになります。

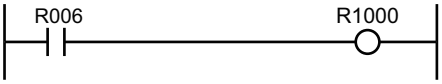
TMnnnに格納されている値	簡易間接指定の参照先	
0	DM0	DMを間接指定
1	DM1	
2	DM2	
⋮	⋮	
9997	DM9997	
9998	DM9998	
9999	DM9999	
10000	R000	リレー(R)を間接指定
10001	R001	
10002	R002	
⋮	⋮	
27913	R17913	
27914	R17914	
27915	R17915	

使用できるテンポラリデータメモリの範囲はTM0～TM511です。ただし、TM0～TM3は演算用として使用するため、重複しないように注意してください。

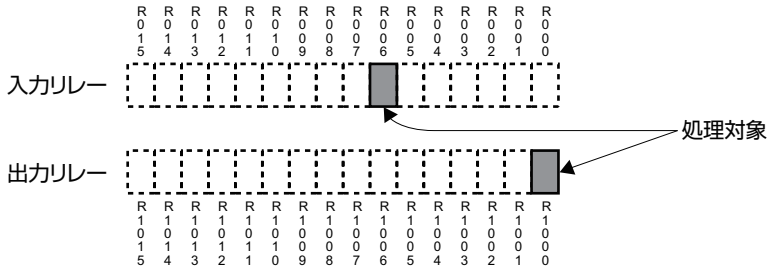
ポイント
- **簡易間接指定では、DM/R以外のデバイスおよびDM10000～DM65534、R18000～R199915を指定できません。**
- **テンポラリデータメモリ（TM）の値が27915より大きいときに簡易間接指定すると、命令実行時に演算フラグであるコントロールリレーCR2012がONし、その命令は実行されません。運転は継続されます。**

ビットデバイスのワード単位扱い

通常(シーケンス命令では)ビットデバイスを1ビットのデバイスとして処理します。

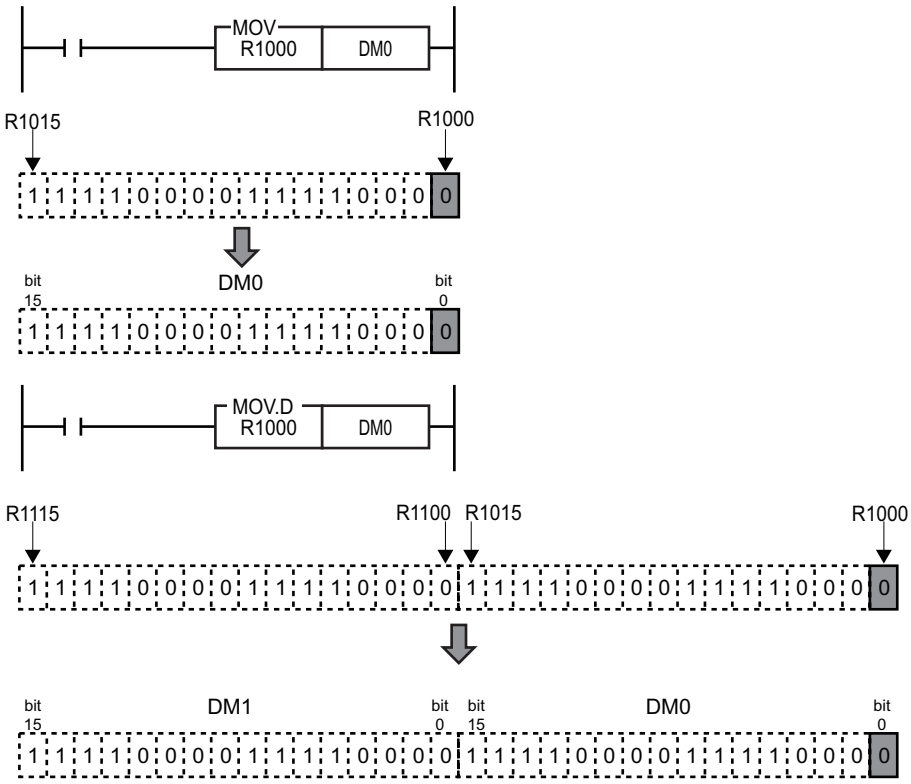


このような命令の場合、処理対象は、
R006、R1000どちらも1ビットです。

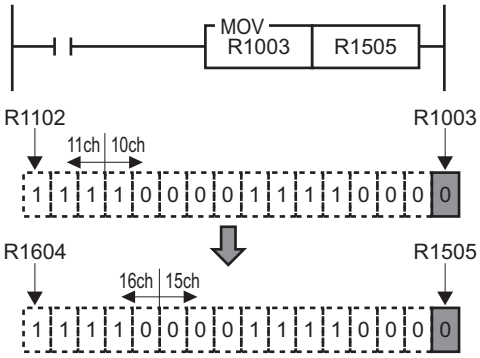


演算命令で、ビットデバイスを16ビット、32ビットとまとめて処理できます。
ワード単位で処理する命令語のオペランドに、ビットデバイスを指定すると、指定したビットデバイスを先頭としたワード単位に扱われます。

例



オペランドに指定したビットデバイスがチャンネルの先頭以外の時は、次のチャンネルにまたがったワード単位として扱われます。



ポイント　**チャンネルにまたがったワード単位で扱う場合、処理速度が遅くなります。**

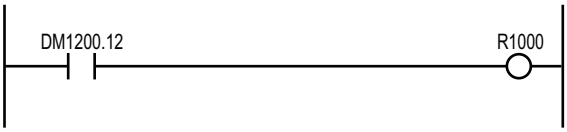
ワードデバイスのビットデバイス扱い

通常は、ワードデバイスを16ビット、32ビットのデバイスとして処理します。
ワードデバイスの任意のビットを扱うには、「ビット接点命令」「ビット出力命令」を使用します。
以下の命令語のオペランドに、ワードデバイスのビット位置を指定できます。

対象命令語：LD、LDB、AND、ANB、OR、ORB、LDP、ANP、ORP、LDF、ANF、ORF、LDPB、ANPB、ORPB、LDFB、ANFB、ORFB、OUT、OUB、SET、RST、KEEP、DIFU、PIFD

例

DM1200.12･･･DM1200の12ビットを扱います



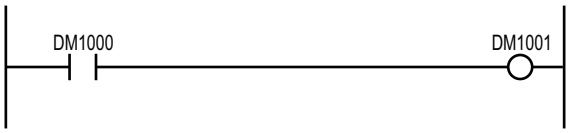
DM1200

0	1	1	1	0	1	1	0	0	1	1	0	1	1	0	0
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

上記以外のその他の命令語でワードデバイスを指定すると、指定したワードデバイスの最下位ビットのみをビットデバイスとして扱います。

例

DM1000の値が#46563（$B5E3）の場合



DM1000=#46563($B5E3)

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

最下位ビット
1:接点ON
0:接点OFF

上記の例の場合、接点はONするので、DM1001の最下位ビットが1になります。

DM1001=#14($E)

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

DM1001=#15($F)

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

参考　最下位ビット以外のビットは変化しません。

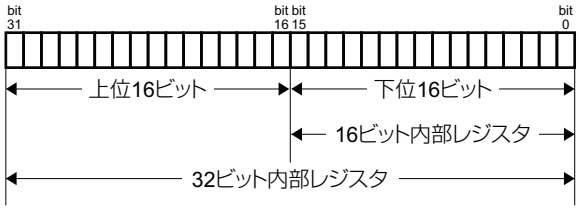
内部レジスタ

内部レジスタは、応用命令・演算命令・拡張命令の演算結果を格納したり、演算時に一時的にデータを格納したりするためのデバイスです。
内部レジスタに格納されたデータは、次に新しいデータが格納されるまで変化しません。
電源ON時、PRG→RUN時に値が0にクリアされます。

内部レジスタは、使用する命令語、サフィックスにより「16ビット内部レジスタ」、「32ビット内部レジスタ」、「64ビット内部レジスタ」に自動的に使い分けられます。
「サフィックス」(4-9ページ)

● 16ビット内部レジスタ・32ビット内部レジスタ

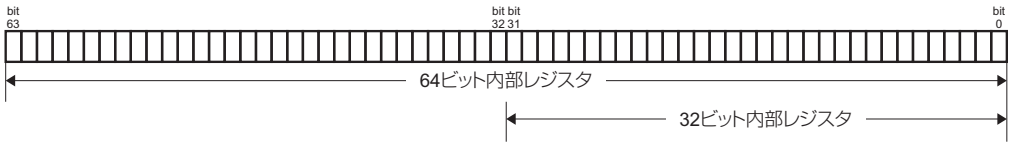
16ビット内部レジスタ・32ビット内部レジスタは次のように、上位16ビット・下位16ビットの32ビットで構成されています。



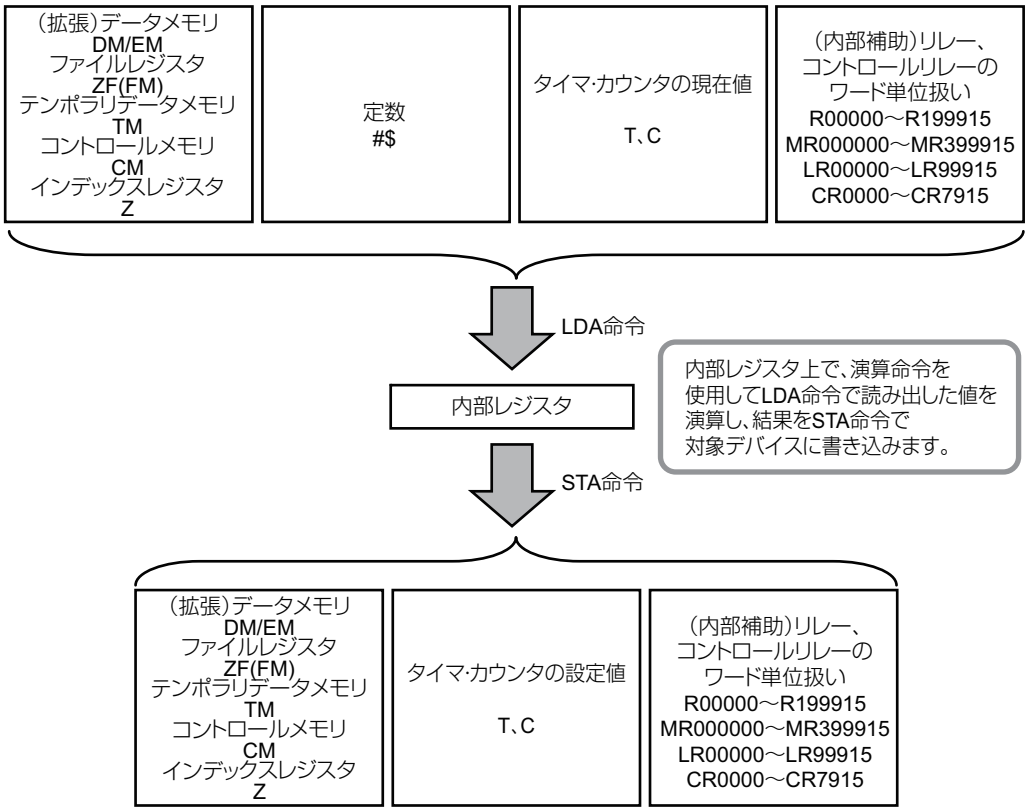
下位16ビットの内部レジスタを16ビット内部レジスタと呼びます。上位・下位合わせた32ビットの内部レジスタを32ビット内部レジスタと呼びます。

● 64ビット内部レジスタ

64ビット内部レジスタは次のように64ビットで構成され、倍精度浮動小数点型の命令やサフィックス(.DF)を扱う場合のみ使用します。64ビット内部レジスタの下位32ビットは、32ビット内部レジスタと共通です。

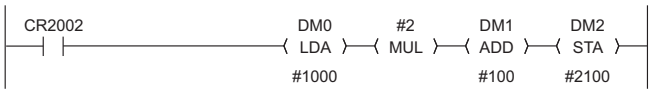


● 内部レジスタと演算命令



例

DM0を内部レジスタに読み出して、2をかけた後DM1の値を加算してDM2に格納します。

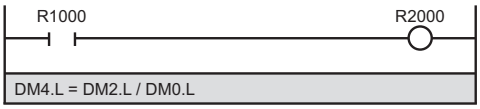


4-5 変数

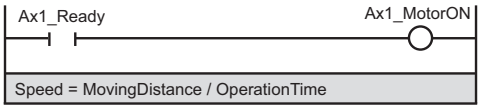
KV-8000(A)では、プログラムで用いられる要素として、ユーザが自由に設定できる変数と、リレー、タイマ･カウンタ、データメモリなど、専用の記号と番号で指定できるデバイスを用意しています。どちらも、命令語の後に記述して、命令で使用するデータを与えたり、命令の実行結果を格納するために使用します。この節では変数について説明します。

- あらかじめ定義した名前を用いてプログラミングできるため、プログラムの可読性が向上します。

デバイスでプログラミングした場合



変数でプログラミングした場合



- 変数の設定に合わせて必要なサイズが CPU ユニットのワーク領域に自動的に割り付けられます。そのため、割り付けを意識する必要がありません。
- 異なるプロジェクト間でプログラムをコピーした場合も、変数の割り付けは自動で行われます。デバイスのように、デバイスマップを変更してコピーする必要がなく、プログラムの流用性が向上します。
- タッチパネルディスプレイ VT5 シリーズも、KV-8000 シリーズの変数に対応しているため、デバイスのようにPLCのデバイス番号を変更した際にタッチパネル側のデバイス番号もあわせて変更する、といった作業が不要になります。
- プログラムで使用している未登録の変数は後から一括取り込みできるため、あらかじめすべての変数を登録してからプログラムを作成する必要はありません。
- デバイスと変数を混在して使用できます。

ポイント

- **変数を使用できるのは、CPU機能バージョン2.0以降のKV-8000のみです。変数を使用する場合は、Ver.11.0以降のKV STUDIOからCPU機能バージョンをアップデートしてください。**
- **タッチパネルで変数を使用できるのは、VT STUDIO Ver.8.0以降で、対応機種がVT5シリーズの場合のみです。**
- **VT STUDIOで変数を使用する場合、KV STUDIOの変数情報を取得しておく必要があります。詳細は、『VT STUDIO リファレンスマニュアル』を参照してください。**
- **KV-XD02で設定できるユニット変数はラダープログラムでは使用できません。**

変数の種類

変数には、以下の2種類があります。

■ グローバル変数

プロジェクト全体で共通して扱う変数です。グローバル変数として定義した変数は、複数のプログラムや、タッチパネルディスプレイVT5シリーズから参照できます。例えば、「Var_X」というグローバル変数を定義した場合、モジュールAで「Var_X」を使用した場合と、モジュールBで「Var_X」を使用した場合とで、同じ実体を参照します。
また、グローバル変数をCPUユニットのデバイスに手動で割り付けることも可能です。

■ ローカル変数

プログラム単位で扱う変数です。ローカル変数として定義した変数は、各プログラム内でのみ参照が可能です。例えば、モジュールAで「Var_Y」というローカル変数を定義し、モジュールBでも「Var_Y」という変数を定義した場合、モジュールAで使用する「Var_Y」とモジュールBで使用する「Var_Y」は異なる実体を参照します。ローカル変数はVT5シリーズからは参照できないため、VT5シリーズから参照する変数はグローバル変数として登録してください。
また、ローカル変数はCPUユニットのデバイスには手動で割り付けられません。

変数とデータ型一覧

変数を使用する場合、明示的にデータ型を定義する必要があります。CPUユニットは定義されたデータ型に応じて必要なデータサイズを自動的にワーク領域に割り付けます。
KV-8000(A)で使用できるワーク領域のサイズと、変数の仕様は以下のとおりです。

ワーク領域のサイズ

項目	内容
ビット	64000点
ワード	589824ワード
タイマ	4000点
カウンタ	4000点

変数の仕様

項目	内容
最大定義数	100000個※1
変数名の最大長	128文字（全角/半角問わずに128文字）
最大保持設定数	30000設定※2※3

※1 グローバル変数とローカル変数の合計です。変数名の文字数によっては 100000 個設定できない場合があります。
※2 ビット、ワード、カウンタの合計です。それぞれの個数は CPU システム設定で設定します。変数名の文字数によっては30000個設定できない場合があります。
※3 変数編集ウィンドウで登録した変数名1つにつき1設定とカウントします。例えば10000個の要素を持つ配列型変数を1つ設定した場合、設定数は1になります。

参 考
配列や構造体など1つの変数名で複数のビットやワードを占有する変数を設定することで、変数領域すべてを保持することも可能です。

KV-8000(A)で使用できる変数のデータ型と、データサイズは以下のとおりです。CPUユニットは変数のデータ型に応じて、以下のデータサイズを自動的にワーク領域に割り付けます※1※2。

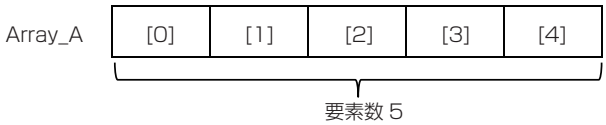
データ型		データサイズ（ビット単位）	アライメントサイズ※3（ビット単位）
型名	説明		
BOOL	ビット	1	1
UINT	1ワード符号なし整数	16	16
INT	1ワード符号あり整数	16	16
UDINT	2ワード符号なし整数	32	32
DINT	2ワード符号あり整数	32	32
REAL	単精度浮動小数点型	32	32
LREAL	倍精度浮動小数点型	64	32
STRING※4	文字列	（文字数※5+1）×8を16の整数倍に切り上げ	16
TIMER	タイマ	—	—
COUNTER	カウンタ	—	—
ARRAY	配列	データ型 × 配列サイズ	データ型に従う
構造体	—	定義による※6	メンバのアライメントサイズの最大
ファンクションブロック	ファンクションブロック	定義による※6	32

※1 CPUユニットのデバイスに手動で割り付けることも可能です。その場合、ワーク領域は消費しません。
※2 変数を定数設定した場合、ワーク領域は消費しません。
※3 メモリを確保するときの単位です。
※4 STRING 型で扱う文字コードはプロジェクト言語設定で設定した言語に従います。プロジェクト言語設定の詳細は『KV STUDIO ユーザーズマニュアル』を参照してください。
※5 文字数とはNULLまでのバイト数のことです。Shift-JISなどの2バイト文字の場合は1文字で文字数を2として計算します。
※6 構造体およびファンクションブロックは複数のデータ型を設定して使用するデータ型です。そのため、設定内容によりデータサイズが決まります。

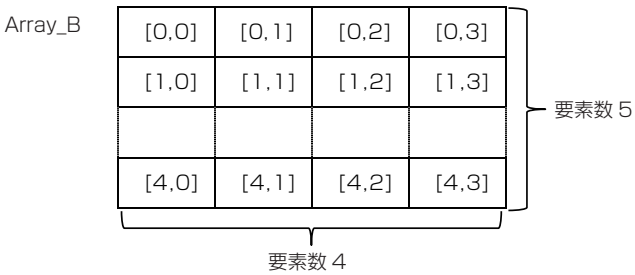
配列

配列とは、同じデータ型のデータの集合体を1つの名前で定義するデータ型です。配列を構成する1つ1つのデータを要素といいます。配列では、変数名の他に、データ型と要素数と次元数を設定します。次元数がnの配列を、n次元配列と呼びます。以下に、1次元配列～3次元配列の例とイメージ図を示します。

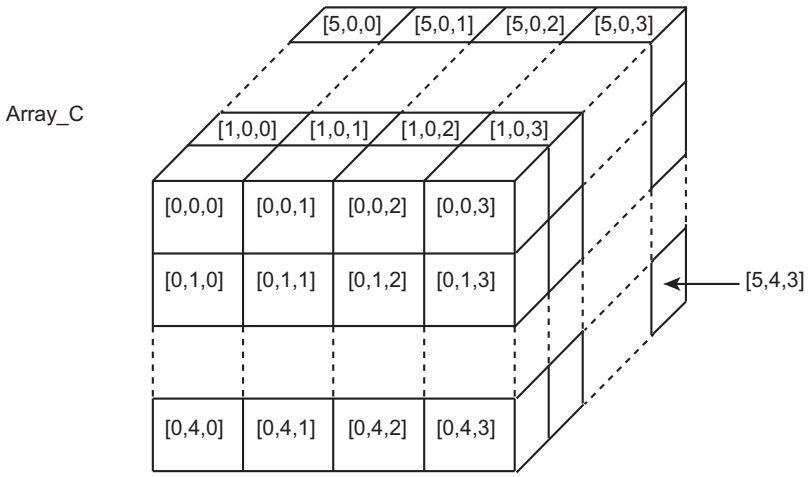
- 1次元配列　変数名：Array_A　要素数：5の場合



- 2次元配列　変数名：Array_B　要素数：5×4の場合



- 3次元配列　変数名：Array_C　要素数：6×5×4の場合



ポイント　**配列型変数ではすべての要素のデータ型が共通になります。**

■ 配列型変数の記載

● 配列型変数の要素の指定

プログラム上で配列の要素を指定して使用する場合は、配列名と[　]の添え字を組み合わせて使用します。2次元以上の配列の場合は、要素番号を「カンマ(,)」で区切って指定します。例えば、3次元配列Array_Cの[5,0,3]の要素を使用する場合、プログラム上では以下のとおり「Array_C[#5,#0,#3]」と記載します。(#は省略して入力できます)

● 配列型変数の全体の指定

配列を構成する全要素のことを「配列全体」といいます。プログラムで配列全体を指定する場合は、変数名だけを記載します。例えば、Array_Aという配列全体を指定する場合は、「Array_A」と記載します。

参 考　KV STUDIO Ver.11.0では配列をラダープログラムで扱えます。

ポイント
変数名だけを記載して配列全体として扱う命令語はACNT/WSIZE/BSIZE命令のみです。
その他の命令語で変数名だけを記載した場合は先頭要素から命令語が扱う範囲のみを扱います。
例えば、UINTの配列型変数Array_Aに対して「MOV.D 0 Array_A」と記述した場合はArray_A[0]とArray_A[1]の2ワードのみに0が格納されます。

配列の仕様

項目	内容
1次元あたりの最大要素数	589823
最大次元数	8
1配列あたりで占有可能な最大データサイズ	589823ワード
指定可能なデータ型	BOOL / UINT / INT / UDINT / DINT / REAL / LREAL / STRING / 構造体 / ファンクションブロック
指定可能な添え字※1	デバイス/定数/変数/演算式※2
添え字に使用できるデータ型	UINT / INT / UDINT / DINT

※1　添え字には0または正の整数を指定してください。その他の数を指定した場合、変換エラーまたは演算エラーが発生します。
※2　添え字に演算式を使用できるのは KV スクリプトのみです。ラダープログラム上では添え字に演算式を使用できません。

ポイント
- **要素数「10」で設定したArray_Aという配列に対してArray_A[10](11番目の要素)と指定するなど、指定可能な範囲を超えて要素を指定した場合、意図しない領域の値を参照する可能性があります。特に、添え字に変数や演算式を使用する場合は、配列を参照する前にACNT命令やWSIZE命令を実行して配列の要素数や配列全体のデータサイズを取得し、指定可能な範囲を超えた要素を参照しないようにしてください。配列の範囲外を指定した場合は、変換エラーまたは演算エラーが発生します。**
  「ACNT」(4-67ページ)、「WSIZE」(4-69ページ)、「BSIZE」(4-71ページ)
- **配列の添え字にデバイスを指定した場合は16ビット符号無し整数として扱われます。**
- **配列の添え字にビットデバイス(R、MRなど)、BOOL型の変数、ワードデータのビット指定などのビットデータは指定できません。**
- **KV STUDIOのバージョンによって、配列のデータ型の表記が以下のとおり異なります。**
  例: 要素数が3のUINT型配列の場合
  Ver.11.20以降　ARRAY[0..2] OF UINT
  Ver.11.10以前　UINT[3]

構造体

構造体とは、異なるデータ型の、複数の変数の集合体を1つの名前で定義するデータ型です。構造体を構成する1つ1つのデータをメンバといいます。構造体では、変数名の他に、構造体を構成するメンバ名とメンバのデータ型を設定します。構造体には以下のような特長があります。

- 事前に構造体のメンバとデータ型を登録しておくことで、同じ構造を持った変数を作りやすくなる
- 登録モニタウィンドウなどでモニタする際に構造体変数を登録すると、すべてのメンバを一括でモニタできる
- メンバ名で内容を確認できるので、視認性が高くなる

モータ軸の以下の情報をプログラムで使用する場合を例に説明します。

現在座標(CurCoordinate)
現在速度(CurSpeed)
軸制御中(Control)
軸エラー(Error)
軸エラーコード(ErrorCode)

構造体を使用しない場合、軸を複数使用しようとすると以下のように軸番号が異なるだけの同一内容の変数を登録する必要があります。
構造体を使用する場合は、軸情報のデータ構造を構造体として事前に定義しておくことで、複数の軸を使用する場合でもデータ型に同じ構造体を設定するだけで済みます。

<構造体を使用しない場合>

変数名	データ型
Ax1_CurCoordinate	DINT
Ax1_CurSpeed	DINT
Ax1_Control	BOOL
Ax1_Error	BOOL
Ax1_ErrorCode	UINT
Ax2_CurCoordinate	DINT
Ax2_CurSpeed	DINT
Ax2_Control	BOOL
Ax2_Error	BOOL
Ax2_ErrorCode	UINT

<構造体を使用する場合>

変数名	データ型
Ax1	AxisInfo
Ax2	AxisInfo

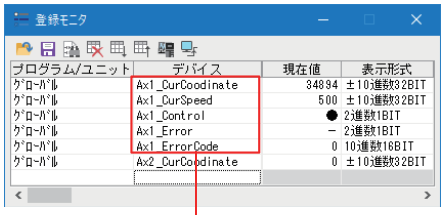
「AxisInfo」型データの定義

メンバ名	データ型
CurCoordinate	DINT
CurSpeed	DINT
Control	BOOL
Error	BOOL
ErrorCode	UINT

事前に設定した構造体を流用

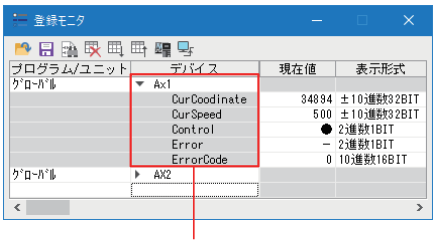
また、KV STUDIO上でモニタする際も、構造体を使用しない場合はすべての変数を手動で入力する必要がありますが、構造体を使用する場合は変数名を入力するだけで構造体内のメンバも一括でモニタできます。

＜構造体を使用しない場合＞



モニタしたい変数はすべて入力する必要があります。

＜構造体を使用する場合＞



構造体の場合、変数名を入力するとメンバを一括でモニタできます。

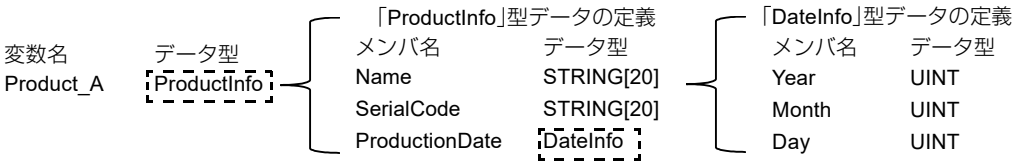
■ 構造体型変数の記載

プログラム上で構造体型変数を使用する際の記載方法について説明します。

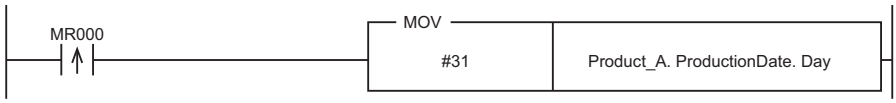
● 構造体型変数のメンバの指定

プログラムで構造体型変数のメンバを使用する場合は、「構造体型変数名+ピリオド(.)+メンバ名」と記載します。例えば、Ax1というAxisInfo型変数のCurSpeedというメンバを参照する場合は、「Ax1.CurSpeed」と記載します。

また、構造体型変数のメンバのデータ型に構造体を指定できます。以下の例では、「ProductInfo」という構造体型の変数Product_Aに「ProductionDate」という構造体型変数がネストされています。



Product_AのDayをプログラムで指定する場合は、「Product_A.ProductionDate.Day」と記載します。



ポイント **構造体のネストは最大8段までです。**

● 構造体型変数の全体の指定

構造体型変数を構成する全メンバを「構造体全体」といいます。プログラムで構造体全体を指定する場合は、変数名だけを記載します。例えば、Product_Aという構造体型変数の構造体全体を指定する場合は「Product_A」と記載します。

ポイント

- **FMOV命令などを使用して構造体全体に値を書き込む場合、転送データ数に構造体全体のデータ数より大きな値を指定すると、意図しない領域の値を変更してしまう可能性があります。FMOV命令などで構造体全体を参照する場合は、命令実行前にWSIZE命令を実行して構造体全体のデータサイズを取得してください。**
  「WSIZE」(4-69ページ)
- **変数名だけを記載して構造体全体として扱うのはWSIZE/BSIZE命令のみです。その他の命令語で変数名だけを記載した場合、構造体の先頭から命令語が扱う範囲のみを扱います。例えば、構造体型変数Ax1に対して「MOV.D 0 Ax1」と記述した場合、Ax1の先頭から2ワードのみに0が格納されます。**
  「BSIZE」(4-71ページ)

● 構造体メンバのオフセット指定

構造体のメンバをオフセット指定することで、各メンバのワーク領域内のメモリ配置を設定できます。メモリ配置を設定することで、外部機器とデータリンクする際に両者の構造体型メンバのメモリ配置を一致させることができます。構造体メンバのオフセットは、「自動」と「手動」から設定できます。

● オフセットが自動の場合

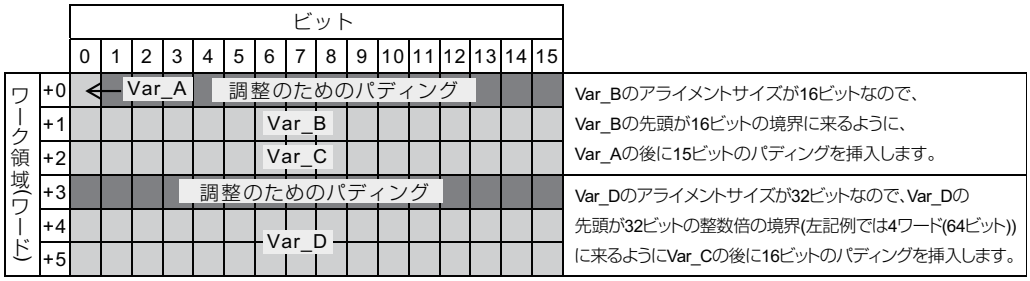
オフセットが自動の場合、メモリ配置は以下のルールに従って自動で行われます。境界の調整のためのパディングは自動的に挿入されます。

項目	内容
構造体の先頭	構造体のメンバで、最も大きいアライメントサイズの境界に構造体の先頭が割り付けられます。ただし、CPUユニットのデバイスに割り付ける場合は、アライメントサイズに関わらずワード単位での割付が可能です。
各メンバの先頭	各メンバのアライメントサイズの境界に割り付けられます。ただし、BOOL型の配列は16ビットの境界に割り付けられます。

例)以下のメンバを持つ構造体の場合

メンバ名	データ型
Var_A	BOOL
Var_B	INT
Var_C	INT
Var_D	DINT

構造体のメンバのうち、最もアライメントサイズが大きいのはVar_Dの32ビットなので、ワーク領域の先頭から数えて偶数番目の領域に構造体の先頭を配置します。各メンバの配置は以下のとおりです。



※次のアライメントサイズの境界位置までオフセットすることをパディングといいます。

● オフセットが手動の場合

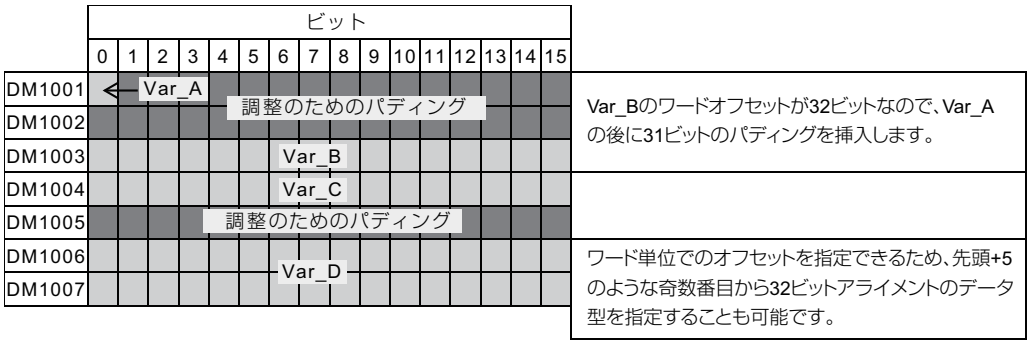
オフセットが手動の場合、以下のルールに従ってメモリ配置を任意に設定できます。

項目	内容
構造体の先頭	構造体のメンバで、最も大きいアライメントサイズの境界に構造体の先頭が割り付けられます。ただし、CPUユニットのデバイスに割り付ける場合は、ワード単位での割付が可能です。
各メンバの先頭	BOOL型のメンバはビット単位で、それ以外のデータ型のメンバはワード単位で、構造体の先頭からのオフセットを指定できます。 ただし、オフセット数の上限はワードオフセットの場合はワーク領域の終端まで、ビットオフセットの場合は15ビットです。

例)以下のメンバ、オフセット指定を持つ構造体をDM1001に割り付けた場合の場合

メンバ名	データ型	ワードオフセット	ビットオフセット
Var_A	BOOL	0	0
Var_B	INT	2	0
Var_C	INT	3	0
Var_D	DINT	5	0

CPUユニットのデバイスに割り付ける場合、ワード単位で指定できるため、DM1001のように奇数番目のデバイスに構造体の先頭を配置できます。



構造体の仕様

項目	内容
構造体の最大定義数	800
構造体1つあたりの最大メンバ数	500
構造体のネスト最大段数	8※1
指定可能なデータ型	BOOL / UINT / INT / UDINT / DINT / REAL / LREAL / STRING/ 配列 / 構造体

※1　stAAA.memberと記載した場合、ネスト段数は1段です。stAAA.stBBB.memberと記載した場合、ネスト段数は2段です。

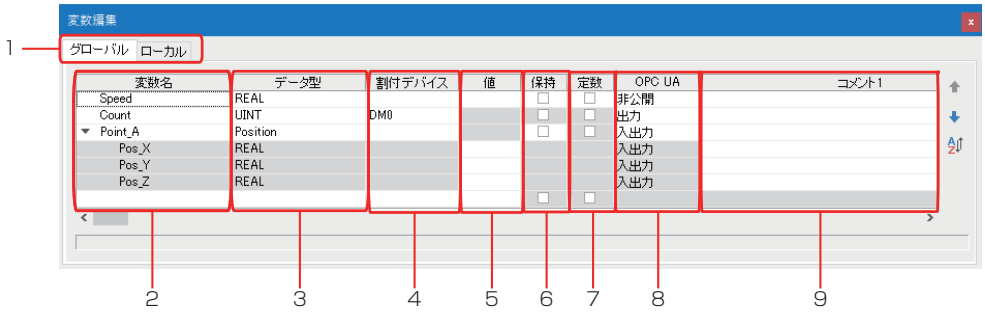
変数の設定

変数を定義してラダープログラムで使用するまでの手順を説明します。

■ 設定方法

1 **【表示(V)】▶【変数編集ウィンドウ(L)】を選択し、[変数編集]ダイアログを開きます。**

別手順 ワークスペースの「変数」をダブルクリック

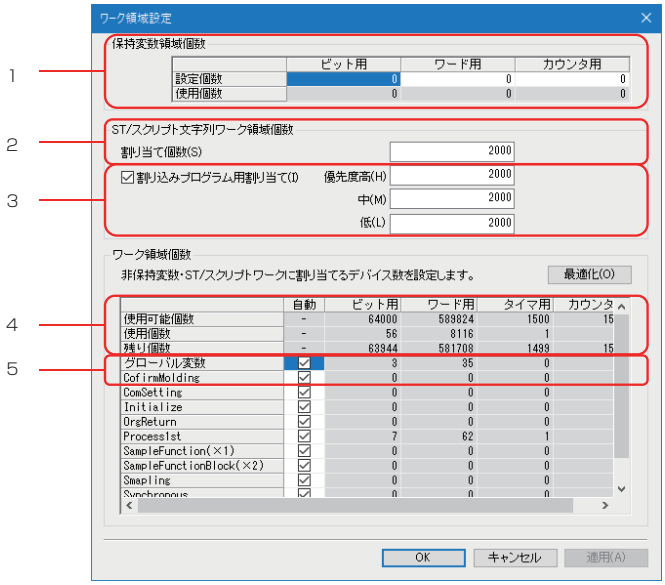


番号	項目	内容
1	グローバル/ローカルタブ	グローバル変数編集ウィンドウとローカル変数編集ウィンドウを切り換えます。 「変数の種類」(4-55ページ)
2	変数名	変数名を入力します(半角128文字/全角128文字)。
3	データ型	変数のデータ型を入力します。 BOOL/UINT/INT/UDINT/DINT/REAL/LREAL/STRING/ TIMER/COUNTER/配列/構造体/ファンクションブロック
4	割付デバイス	変数をCPUユニットのデバイスに手動で割り付ける場合に設定します。割付は必須ではありません。ローカル変数編集ウィンドウでは存在しない項目です。
5	値	項目7で定数にチェックを入れた場合: 変数の初期値を指定します。定数のチェックが入っている場合は値が固定され、プログラムから値を変更できなくなります。 項目7で定数にチェックを入れない場合: 変数の初期値を設定します。プログラム内で初期値以外の値に変更できます。 割付デバイスを設定している場合は、変数編集ウィンドウでは初期値を設定できません。デバイス初期値設定で、割り付けたデバイスの初期値を設定してください。 「デバイス初期値設定」(4-13ページ)
6	保持	チェックを入れた変数は、電源断時も値が保持されます。保持できる変数はビット、ワード、カウンタの合計で最大30000個※1です。変数の値を保持する場合は、[ワーク領域設定]ダイアログで保持変数領域個数を設定してください。保持にチェックを入れた変数の合計サイズが、保持変数領域個数より大きい場合は変換エラーが発生します。※3 「変数の割付」(4-64ページ) 割付デバイスを設定している場合は、上記の30000個に含まれないため設定したデバイスに対して停電保持設定を行ってください。 「停電保持設定(ラッチ)」(4-26ページ) 保持にチェックを入れていない変数は、PROGRAMモード→RUNモード切り換え時に「値」で設定した値に初期化されます。設定していない場合は0に初期化されます。
7	定数	チェックを入れた変数は、定数として扱われます。定数に設定した変数は、最大定義数100000個の中に含まれますが、ワーク領域を消費しません。
8	OPC UA	OPC UAサーバ機能で読み書き可能な変数を設定します。非公開に設定した場合、変数は読み書きできません。入出力に設定した場合、読み書き可能な変数として登録されます。出力に設定した場合、読み出しのみ可能な変数として登録されます。ローカル変数編集ウィンドウでは存在しない項目です。
9	コメント	変数のコメントを設定します(半角128文字/全角128文字)※2。設定したコメントはKV STUDIOのラダー編集ウィンドウに表示されます。1つの変数に対してコメント1～コメント8までを設定できます。

※1　変数名の文字数によっては30000個設定できない場合があります。
※2　バージョン2.1未満の場合32文字までです。
※3　FB命令を使用した場合は値が保持されません。FBCALL命令またはFBSTRT命令を使用してください。

■ 変数の割付

変数は定義したタイミングで、ワーク領域に割り付きます。ワーク領域の使用個数の確認や割り当て個数の変更は[ワーク領域設定]で行います。
KV STUDIOメニューの【表示(V)】▶【CPUシステム設定(P)】を選択し、「ローカル全体割付設定」をクリックします。〔ワーク領域設定〕をクリックすると、[ワーク領域設定]ダイアログが表示されます。



番号	項目	説明
1	設定個数	変数を保持する領域のサイズを設定します。「設定個数」が「使用個数」より大きくなるように設定してください。「設定個数」が「使用個数」より小さい場合は変換エラーが発生します。ビット用の「設定個数」は16の倍数を設定してください。「設定個数」を変更した場合はRUN中書き込みできません。
	使用個数	変数編集ウィンドウで「保持」にチェックを入れた変数の合計のサイズが表示されます。
2	ST/スクリプト文字列ワーク領域個数	ST/スクリプトで文字列関数を扱うときのワーク領域個数を設定します。ワード用のワーク領域を消費します。
3	割り込みプログラム用割り当て	割り込みプログラムや定周期モジュール、ユニット間同期モジュール内のST/スクリプトで文字列関数を扱うときのワーク領域個数を割り込みの優先度別に個別に設定します。ワード用のワーク領域を消費します。
4	使用可能個数	使用可能なワーク領域やファンクションブロック・マクロ引数、保持変数領域個数、@FMの使用個数の合計を表示します。
	使用個数	プロジェクト内で使用しているワーク領域の個数を表示します。
	残り個数	使用できるワーク領域の残り個数を表示します。 [残り個数] = [使用可能個数] – [使用個数]
5	プログラム別割付状況表示	各プログラムのワーク領域割付個数を表示します。割付個数は、設定した変数のサイズに応じて自動で計算されます。 「自動」のチェックボックスをOFFにすると、使用個数を手動で設定できます。

ポイント

- 割り込みプログラム（または定周期モジュール、ユニット間同期モジュール）内と割り込み以外の通常のモジュール内の両方でST/スクリプトの文字列関数を使用する場合、ST/スクリプト文字列用ワーク領域個数は、割り込み用と通常のモジュール用でそれぞれ割り当ててください。または、DI命令/EI命令を使用して通常のモジュール内の文字列処理中に割り込みが入らないようにプログラムしてください。
  割り当てを行わない場合、通常のモジュールで文字列関数処理中に割り込みプログラムの文字列関数が実行されると、同じワーク領域が使用されるため、正常に文字列処理が行われません。

構造体の定義

■ 設定方法

1 **【表示(V)】▶【構造体(J)】▶【新規作成(N)】を選択し、構造体を新規作成します。**

別手順 ワークスペースの「構造体」をダブルクリック

項目	内容
構造体名	構造体の名前を指定します。全角半角問わず最大128文字です。
オフセット	メンバの各メンバのオフセットを「自動/手動」から選択します。 「構造体メンバのオフセット指定」(4-61ページ)
コメント	入力欄に入力した文字列がKV STUDIOワークスペースの構造体名の右横に表示されます。コメント1～コメント8の8種類を設定できます。

2 **各メンバのメンバ名やデータ型を設定します。**

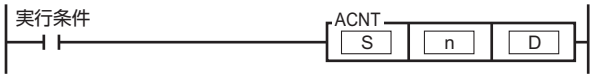
番号	項目	説明
1	メンバ名	構造体のメンバ名を指定します。全角半角問わず最大128文字です。
2	データ型	メンバのデータ型を選択します。
3	詳細表示	チェックを入れるとワードオフセットとビットオフセットが表示されます。
	オフセット	構造体の各メンバのオフセット情報が表示されます。オフセットを「自動」に設定している場合は、自動で設定されたオフセットが表示され、背景はグレーアウトし設定はできません。オフセットを「手動」に設定している場合は各メンバのワードオフセットとビットオフセットを入力します。
4	コメント	各メンバのコメントを設定します。コメント1～コメント8の8種類を設定できます。

ACNT

ACNT [ ] [ ] 配列要素数取得

指定した次元の配列要素数を取得します。

ラダープログラム



入力方法

オペランド	説明	占有領域
S	配列全体を指定します。※1※2	―
n	要素数を取得する次元数(1～8)を指定します。	16ビット
D	結果の格納先を指定します。	32ビット

※1 S に指定できるのは配列型の変数名だけです。配列以外の変数や変数以外の要素は指定できません。

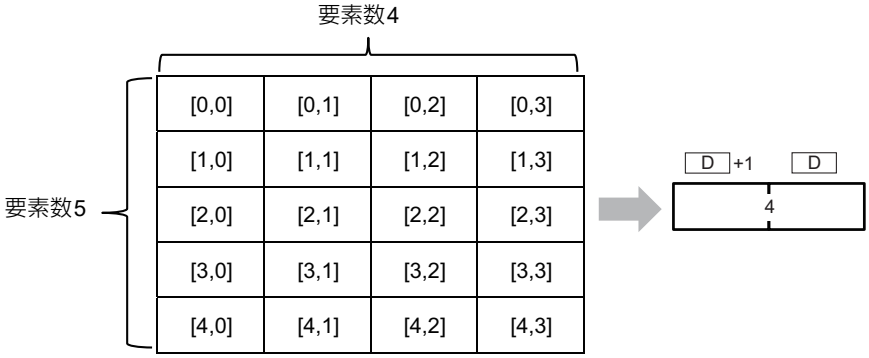
※2 FB引数の可変長配列は指定できません。

動作説明

ACNT 実行条件がONのとき、S で指定した配列の n 次元目の要素数を[D・D+1]に格納します。

例 S =Array_A(5×4の2次元配列)、n =2の場合

Array_A



ポイント

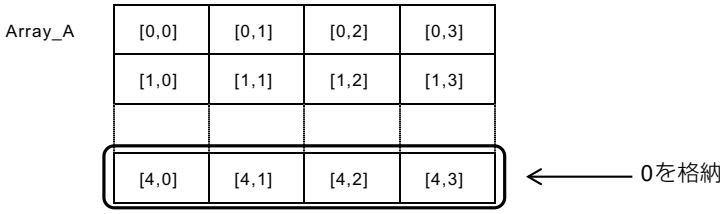
- **ACNT命令で取得するのはワードサイズではなく要素数です。例えば、UDINT(2ワード符号なし整数)が4個の次元を指定した場合、[D・D+1]には4が格納されます。ワードサイズの8が格納されるわけではありません。**

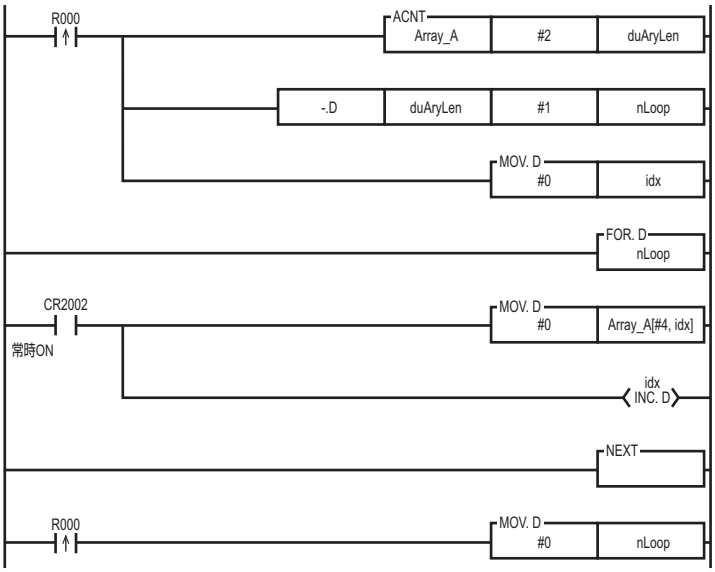
演算フラグ

CR2009	変化なし
CR2010	変化なし
CR2011	変化なし
CR2012	• 以下のいずれかの条件が成立した場合はON、それ以外はOFF • n に定数以外を指定し、n が0もしくは存在しない次元数の場合 • 間接参照、インデックス修飾が不適切な場合 • 以下の条件をすべて満たす場合は、変化しません。 • n に定数を指定している • D に間接指定、インデックス修飾を使用していない、およびオペランドの範囲外アクセスのチェックが行われない場合

サンプルプログラム

入力リレーR000の立ち上がりで、5×4の2次元配列Array_Aの2次元目の要素数をDM0に格納します。その後、Array_A[4,0]、Array_A[4,1]…Array_A[4,3]に0を格納します。





〈ニモニックリスト〉

```
LDP R000
ACNT Array_A #2 duAryLen
CAL-.D duAryLen #1 nLoop
MOV.D #0 idx
FOR.D nLoop
LD CR2002
MOV #0 Array_A[#4,idx]
INC.D idx
NEXT
LDP R000
MOV.D #0 nLoop
```

WSIZE

WSIZE [ ] ワードサイズ取得　指定した変数が占有するワードサイズを取得します。

ラダープログラム

入力方法

オペランド	説明	占有領域
S	変数を指定します。※1※2	―
D	結果の格納先を指定します。	32ビット

※1　S に指定できるのは変数のみです。デバイスは指定できません。
※2　FB引数の可変長配列、可変長文字列は指定できません。

動作説明

WSIZE　実行条件がONのとき、S が占有するワードサイズを[D ・D +1]に格納します。

例　S =以下の構造体型変数を指定した場合

メンバ名	データ型
member1	UINT
member2	STRING[20]
member3	REAL

→ STRING[20]のワード数は(20+1)/2を切り上げて11ワードなので、1 + 11 + 2 = 14となります。

ポイント

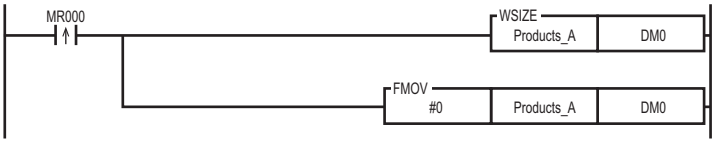
- **WSIZE命令で取得するのはワードサイズです。例えば、UDINT型のデータ(2ワード符号なし整数)を4つ持つ配列型変数や構造体型変数を指定した場合、2ワード × 4 = 8ワードなので[D ・D +1]には8が格納されます。要素数やメンバの個数が格納されるわけではありません。**
- **[D ・D +1]にはオフセットも含めたワードサイズが格納されます。オフセットの詳細は 「構造体メンバのオフセット指定」(4-61ページ)を参照してください。**
- **[D ・D +1]にはワードサイズに切り上げた結果が格納されます。例えば、BOOL[33]の場合は3が格納されます。**

演算フラグ

CR2009	変化なし
CR2010	変化なし
CR2011	変化なし
CR2012	• 以下のいずれかの条件が成立した場合はON、それ以外はOFF 　• 間接参照、インデックス修飾が不適切な場合 • 以下の条件をすべて満たす場合は、変化しません。 　• オペランドに間接指定、インデックス修飾を使用していない 　• 配列の添え字にデバイス、変数を使用していない

サンプルプログラム

入力リレーMR000の立ち上がりで変数「Products_A」が占有するワードサイズを求め、DM0に格納します。その後、「Products_A」が占有する領域に0を格納します。



〈ニモニックリスト〉

```
LDP MR000
WSIZE Products_A DM0
FMOV #0 Products_A DM0
```

ポイント

上記参考プログラムは変数のワードサイズが65535以下の場合のみ使用できます。
変数のワードサイズが65535より大きい場合は、レシピロード機能を使用して0を格納してください。
「レシピロード・セーブ」(5-50ページ)

BSIZE

BSIZE [ ] [ ]　ビットサイズ取得

指定した変数が占有するビットサイズを取得します。

ラダープログラム

入力方法

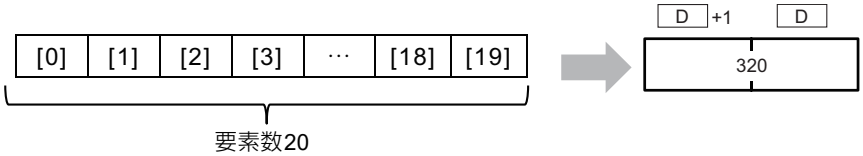
オペランド	説明	占有領域
S	変数を指定します。※1※2	―
D	結果の格納先を指定します。	32ビット

※1　S に指定できるのは変数のみです。デバイスは指定できません。
※2　FB引数の可変長配列、可変長文字列は指定できません。

動作説明

BSIZE　実行条件がONのとき、S が占有するビットサイズを[D・D +1]に格納します。

例　S にINT型の配列変数を指定した場合



→ INT型（16ビット）で要素数が20なので、16ビット×20=320ビット

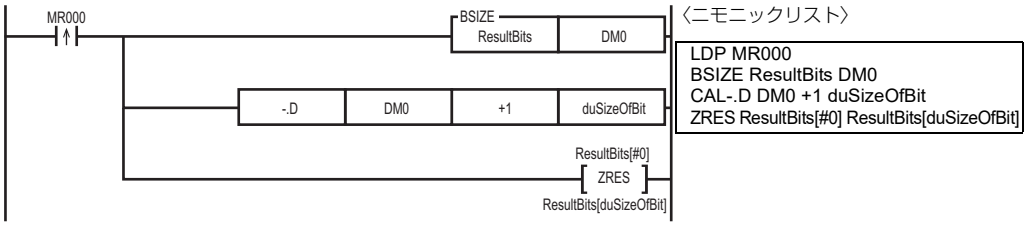
ポイント　**[D・D +1]にはオフセットも含めたビットサイズが格納されます。オフセットの詳細は 「構造体メンバのオフセット指定」（4-61ページ）を参照してください。**

演算フラグ

CR2009	変化なし
CR2010	変化なし
CR2011	変化なし
CR2012	• 以下のいずれかの条件が成立した場合はON、それ以外はOFF 　• 間接参照、インデックス修飾が不適切な場合 • 以下の条件をすべて満たす場合は、変化しません。 　• オペランドに間接指定、インデックス修飾を使用していない 　• 配列の添え字にデバイス、変数を使用していない

サンプルプログラム

入力リレーMR000の立ち上がりでBOOLの配列型変数「ResultBits」が占有するビットサイズを求め、DM0に格納します。配列の終端番号をUDINT型の変数duSizeOfBitに格納し、「ResultBits」のビットをすべてリセットします。



ポイント

- **配列の添え字にDMなどのデバイスを指定した場合、1ワード符号なし整数として扱われるため、65535ビットまでの配列しか扱えません。配列の要素数が65535より大きい場合はサンプルプログラムのようにUDINT型変数を添え字に使用します。**
- **構造体同士を比較する場合はBSIZE命令とBBCMP命令組み合わせて使用します。BBCMP命令の詳細は『KV-8000/7000/5000/3000/1000 シリーズ・KV Nano シリーズ　命令語リファレンスマニュアル』を参照してください。**
- **構造体全体を別の構造体に転送する場合は、BSIZE命令とBBMOV命令を組み合わせて使用します。BBMOV命令の詳細は『KV-8000/7000/5000/3000/1000 シリーズ・KV Nano シリーズ　命令語リファレンスマニュアル』を参照してください。**

4-6 定数

使用する命令語やそのサフィックスによって、表記可能な定数やその範囲が変わります。

#10進定数

概要 16ビットと32ビットのビット幅で表される数値を10進表記で示します。

表記
- 符号無し（サフィックスが「.U」または「.D」の場合）
  先頭に「#」、続いて10進表記の数値を表記します。
  （例）

10進数を表す記号　数値

#06143

- 符号付き（サフィックスが「.S」または「.L」の場合）
  符号を付けて10進表記の数値をそのまま表記します。

範囲
- 16ビット符号無しの場合：#0～#65535
- 32ビット符号無しの場合：#0～#4294967295
- 16ビット符号付きの場合：－32768～＋32767
- 32ビット符号付きの場合：－2147483648～＋2147483647

参考
- ゼロサプレスして入力できます。
- 符号付きの場合にも「#」を付けて入力できます。「#」を付けた場合「＋」のみ省略可能です。
- XYM表示モードでは、「K」を付けて記述できます。
  「XYM表記について」（5-120ページ）

$16進定数

概要 16ビットと32ビットの2種類のビット幅で表される数値を16進表記で示します。

表記 先頭に「$」、続いて16進表記の数値を記述します。
（例）

16進数を表す記号 数値

$1AF7

範囲
- 16ビットの場合：$0000～$FFFF
- 32ビットの場合：$00000000～$FFFFFFFF

ポイント
「$」をつけずにA～Fを含む16進数を入力した場合はエラーになり、A～Fを含まない場合はリレー（R）と認識されます。

参考
- ゼロサプレスして入力できます。
- XYM表示モードでは、「H」を付けて記述できます。
  「XYM表記について」（5-120ページ）

単精度浮動小数点型定数

概要 浮動小数命令のオペランドとして記述できます。

表記
- 符号を付けて10進表記の数値をそのまま表記します。
- 8桁以上で指数表記になります。

範囲

－3.4E＋38≦ N≦－1.4E－45
N＝0
＋1.4E－45≦ N≦＋3.4E＋38（有効桁数：約7桁）

参考 「#」を付けて入力できます。「＋」のみ省略可能です。

倍精度浮動小数点型定数

概要 倍精度浮動小数命令のオペランドとして記述できます。

表記
- 符号を付けて10進表記の数値をそのまま表記します。
- 15桁以上で指数表記になります。

範囲

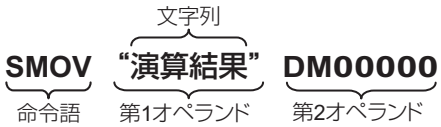
－1.79E＋308≦ N≦－2.23E－308
N＝0
＋2.23E－308≦ N≦＋1.79E＋308（有効桁数：約16桁）

参考
- 「#」を付けて入力できます。「＋」のみ省略可能です。
- 正規化数の範囲外である、非正規化数の倍精度浮動小数点（-2.23E-308＜N≦-4.9E-324、+4.9E-324≦N＜+2.23E-308）を演算すると、演算エラーが発生します。演算結果が非正規化数になる場合は値は0になります。

固定文字列

概要 文字列処理命令とカレンダ接点命令のオペランドとして記述できます。

表記 「""」の間に文字列を表記します。
（例）



ポイント **「""」を付けないと変数扱いになります。**

参考 「"」を文字として扱うには、「""」と入力します。

範囲 499文字以内（半角）

ポイント **KV-8000のCPU機能バージョンが2.0未満の場合は、31文字以内です。**

ポイント **ST言語内では、「"」の間に文字列を表記します。**

4-7 データの取り扱い方

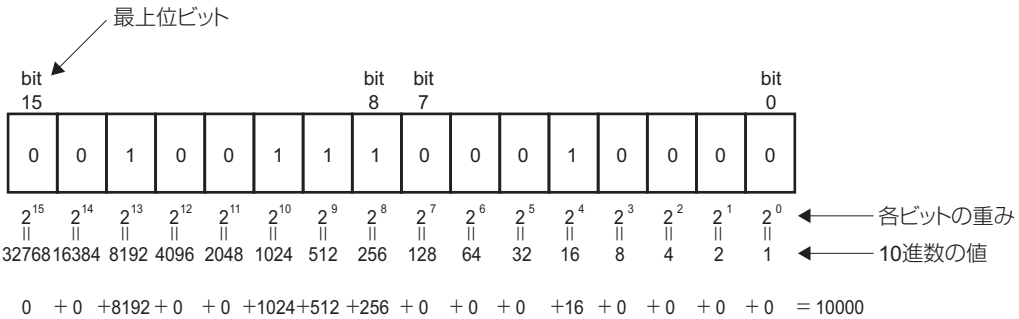
ラダープログラムで扱える数値は、2進数、10進数（符号無し／符号付き）、16進数、2進化10進数（BCD）、単精度浮動小数点型実数、倍精度浮動小数点型実数です。
扱える文字コードは、ASCIIコード･シフトJISコードです。

バイナリデータ

《KV-8000シリーズ》では、デバイスや内部レジスタでの処理を全てバイナリ（2進数）でおこないます。2進数1桁を「ビット」という単位で扱います。「0（OFF）」と「1（ON）」の2状態を表せます。

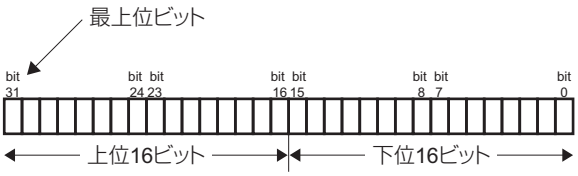
■ 16ビットバイナリデータ

16ビット（2進数で16桁）分のデータを通常「ワード」と言う単位で扱います。10進数で符号なしの場合、0～65535の値を表せます。
（例）



■ 32ビットバイナリデータ

32ビット（2ワード）分のデータは、10進数で符号なしの場合、0～4294967295の値を表せます。



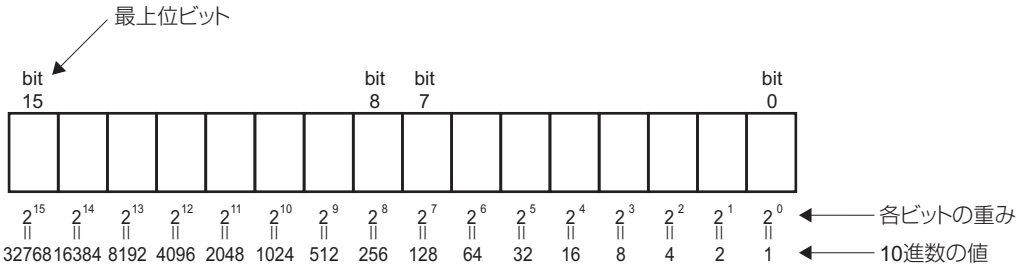
DM/EM/ZF（FM）/TMなど、16ビットのワードデバイスで32ビットデバイスを扱う場合は、連続した2ワードを使用します。下位16ビット分のデータを番号の小さいワードデバイスで、上位16ビット分のデータを番号の大きいワードデバイスで扱います。

参 考　デバイスを16ビット/32ビットのどちらのデータとして扱うかはサフィックスで決定します。
「サフィックス」（4-9ページ）

数値の表記

■ 2進数　BIN（Binary）

CPUユニット内部のバイナリデータをそのまま「0」と「1」だけで表記します。それぞれの桁は、1より大きくなると桁上がりします。



（例）0110110100110001

参 考　命令語のオペランドとして入力できません。

■ 10進数　DEC（Decimal）

CPUユニット内部のバイナリデータを10進数で表記します。

● 符号無し

16ビットデータの場合に扱える値の範囲は0～65535、32ビットデータの場合は0～4294967295となります。

● 符号付き

正負は最上位ビットで判断します。（0:正；1:負）
16ビットデータの場合に扱える値の範囲は－32768～＋32767、32ビットデータの場合は－2147483648～＋2147483647となります。

参 考　命令語のオペランドとして入力できます。数値の先頭に「#」または符号を付けます。
（例）#1000、＋12、－25489

■ 16進数　HEX（Hexadecimal）

CPUユニット内部のバイナリデータを16進数で表記します。
16進数は、4ビット分を1桁として扱います。1桁で表現できる値は10進数で0～15のため、9より大きな値はA、B、C、D、E、Fを使用します。Fより大きくなると桁上がりします。
16ビットデータの場合に扱える値の範囲は、$0000～$FFFF、32ビットデータの場合は$00000000～$FFFFFFFFとなります。

参 考　命令語のオペランドとして入力できます。数値の先頭に「$」を付けます。
（例）$00FF

■ 2進化10進数　BCD（Binary Coded Decimal）

2進化10進数は、16進数と同じように4ビット分を1桁として扱います。10進数と同じように9より大きくなると桁上がりするため、A～Fの記号は使用しません。
16ビットデータの場合に扱える値の範囲は0000～9999、32ビットデータの場合は00000000～99999999となります。

例

bit 15							bit 8	bit 7							bit 0
0	0	0	0	1	0	0	1	1	0	0	0	0	1	1	1

上記のように16ビットデータが格納されていると、本来は$0987（10進表記で#2439）という値ですが、2進化10進数では、これを10進表記の「0987」として扱います。

0	0	0	0	1	0	0	1	1	0	0	0	0	1	1	1
千の位				百の位				十の位				一の位			
0				9				8				7			

2進化10進数は、外部（デジタルスイッチなど）からのデータを扱う時に便利な表現方法です。
ただし、CPUユニット内で演算するにはバイナリデータとして扱う必要があります。外部から読み込んだ時はTBIN命令でバイナリデータに変換し、外部（デジタル表示器など）に出力する時はTBCD命令で2進化10進数に変換してください。

参 考　命令語のオペランドに2進化10進数（BCD）の値を入力するには、数値の先頭に「$」を付けて16進数として入力します。

■ 単精度浮動小数点型実数　FLOAT

浮動小数点型実数は、32ビットデータを小数点付きの数値として扱います。
次の範囲の値（N）が使用できます。

$$-3.4\times10^{+38} \leqq N \leqq -1.4\times10^{-45}$$

$$N=0$$

$$+1.4\times10^{-45} \leqq N \leqq +3.4\times10^{+38}$$

参 考

- 浮動小数命令のオペランドとして入力できます。
  通常の小数点を使用した表記（例：－1.2345）に加えて、「E」を使用した指数表記が可能です。
  （例）　$-2.743\times10^{12} \rightarrow -2.743E+12$
  　　$+8.9025\times10^{-4} \rightarrow +8.9025E-4$
- 浮動小数点型実数の精度は32ビット単精度で、有効桁数は約7桁です。
- 浮動小数点型実数を格納する32ビットデータの内部表現は、IEEEの標準形式です。

■ 倍精度浮動小数点型実数　DOUBLE-FLOAT

倍精度浮動小数点型実数は、64ビットデータを小数点付きの数値として扱います。
次の範囲の値（N）が使用できます。

$$-1.79\times10^{+308} \leqq N \leqq -2.23\times10^{-308}$$
$$N = 0$$
$$+2.33\times10^{-308} \leqq N \leqq +1.79\times10^{+308}$$

参 考

- 倍精度浮動小数命令のオペランドとして入力できます。
  通常の小数点を使用した表記（例：－1.2345）に加えて、「E」を使用した指数表記が可能です。
  （例）　$-2.1437956 \times 10^{12} \rightarrow -2.1437956E+12$
  $+7.63259856 \times 10^{-4} \rightarrow +7.63259856E-4$
- 倍精度浮動小数点型実数の精度は64ビット倍精度で、有効桁数は約16桁です。
- 倍精度浮動小数点型実数を格納する64ビットデータの内部表現は、IEEEの標準形式です。
- デバイスを短精度浮動小数点や倍精度浮動小数点としてのデータとして扱うかはサフィックスで決定します。
  「サフィックス」（4-9ページ）

数値表記の相関

2進数（BIN）、10進数（符号無し/符号付き）、16進数（HEX）、2進化10進数（BCD）の相関を次に示します。

■ 16ビットデータの場合

10進数		16進数	2進数（BIN）
符号無し	符号付き	（HEX）	
0	＋0	0	0000 0000 0000 0000
1	＋1	1	0000 0000 0000 0001
2	＋2	2	0000 0000 0000 0010
3	＋3	3	0000 0000 0000 0011
4	＋4	4	0000 0000 0000 0100
5	＋5	5	0000 0000 0000 0101
6	＋6	6	0000 0000 0000 0110
7	＋7	7	0000 0000 0000 0111
8	＋8	8	0000 0000 0000 1000
9	＋9	9	0000 0000 0000 1001
10	＋10	A	0000 0000 0000 1010
11	＋11	B	0000 0000 0000 1011
12	＋12	C	0000 0000 0000 1100
13	＋13	D	0000 0000 0000 1101
14	＋14	E	0000 0000 0000 1110
15	＋15	F	0000 0000 0000 1111
16	＋16	10	0000 0000 0001 0000
⋮	⋮	⋮	⋮
99	＋99	63	0000 0000 0110 0011
100	＋100	64	0000 0000 0110 0100
⋮	⋮	⋮	⋮
32767	＋32767	7FFF	0111 1111 1111 1111
32768	－32768	8000	1000 0000 0000 0000
⋮	⋮	⋮	⋮
65534	－2	FFFE	1111 1111 1111 1110
65535	－1	FFFF	1111 1111 1111 1111

2進化10進数（BCD）	2進数（BIN）
0000	0000 0000 0000 0000
0001	0000 0000 0000 0001
0002	0000 0000 0000 0010
0003	0000 0000 0000 0011
0004	0000 0000 0000 0100
0005	0000 0000 0000 0101
0006	0000 0000 0000 0110
0007	0000 0000 0000 0111
0008	0000 0000 0000 1000
0009	0000 0000 0000 1001
0010	0000 0000 0001 0000
0011	0000 0000 0001 0001
0012	0000 0000 0001 0010
0013	0000 0000 0001 0011
0014	0000 0000 0001 0100
0015	0000 0000 0001 0101
0016	0000 0000 0001 0110
⋮	⋮
0099	0000 0000 1001 1001
0100	0000 0001 0000 0000
⋮	⋮
0999	0000 1001 1001 1001
1000	0001 0000 0000 0000
⋮	⋮
9998	1001 1001 1001 1000
9999	1001 1001 1001 1001

■ 32ビットデータの場合

10進数		16進数	2進数(BIN)							
符号無し	符号付き	(HEX)								
0	＋0	0	0000	0000	0000	0000	0000	0000	0000	0000
1	＋1	1	0000	0000	0000	0000	0000	0000	0000	0001
2	＋2	2	0000	0000	0000	0000	0000	0000	0000	0010
3	＋3	3	0000	0000	0000	0000	0000	0000	0000	0011
4	＋4	4	0000	0000	0000	0000	0000	0000	0000	0100
5	＋5	5	0000	0000	0000	0000	0000	0000	0000	0101
6	＋6	6	0000	0000	0000	0000	0000	0000	0000	0110
7	＋7	7	0000	0000	0000	0000	0000	0000	0000	0111
8	＋8	8	0000	0000	0000	0000	0000	0000	0000	1000
9	＋9	9	0000	0000	0000	0000	0000	0000	0000	1001
10	＋10	A	0000	0000	0000	0000	0000	0000	0000	1010
11	＋11	B	0000	0000	0000	0000	0000	0000	0000	1011
12	＋12	C	0000	0000	0000	0000	0000	0000	0000	1100
13	＋13	D	0000	0000	0000	0000	0000	0000	0000	1101
14	＋14	E	0000	0000	0000	0000	0000	0000	0000	1110
15	＋15	F	0000	0000	0000	0000	0000	0000	0000	1111
16	＋16	10	0000	0000	0000	0000	0000	0000	0001	0000
⋮	⋮	⋮	⋮							
99	＋99	63	0000	0000	0000	0000	0000	0000	0110	0011
100	＋100	64	0000	0000	0000	0000	0000	0000	0110	0100
⋮	⋮	⋮	⋮							
2147483647	＋2147483647	7FFFFFFF	0111	1111	1111	1111	1111	1111	1111	1111
2147483648	－2147483648	8000000	1000	0000	0000	0000	0000	0000	0000	0000
⋮	⋮	⋮	⋮							
4294967294	－2	FFFFFFFE	1111	1111	1111	1111	1111	1111	1111	1110
4294967295	－1	FFFFFFFF	1111	1111	1111	1111	1111	1111	1111	1111

2進化10進数(BCD)	2進数(BIN)							
00000000	0000	0000	0000	0000	0000	0000	0000	0000
00000001	0000	0000	0000	0000	0000	0000	0000	0001
00000002	0000	0000	0000	0000	0000	0000	0000	0010
00000003	0000	0000	0000	0000	0000	0000	0000	0011
00000004	0000	0000	0000	0000	0000	0000	0000	0100
00000005	0000	0000	0000	0000	0000	0000	0000	0101
00000006	0000	0000	0000	0000	0000	0000	0000	0110
00000007	0000	0000	0000	0000	0000	0000	0000	0111
00000008	0000	0000	0000	0000	0000	0000	0000	1000
00000009	0000	0000	0000	0000	0000	0000	0000	1001
00000010	0000	0000	0000	0000	0000	0000	0001	0000
00000011	0000	0000	0000	0000	0000	0000	0001	0001
00000012	0000	0000	0000	0000	0000	0000	0001	0010
00000013	0000	0000	0000	0000	0000	0000	0001	0011
00000014	0000	0000	0000	0000	0000	0000	0001	0100
00000015	0000	0000	0000	0000	0000	0000	0001	0101
00000016	0000	0000	0000	0000	0000	0000	0001	0110
⋮	⋮							
00000099	0000	0000	0000	0000	0000	0000	1001	1001
00000100	0000	0000	0000	0000	0000	0001	0000	0000
⋮	⋮							
00000999	0000	0000	0000	0000	0000	1001	1001	1001
00001000	1000	0000	0000	0000	0001	0000	0000	0000
⋮	⋮							
99999998	1001	1001	1001	1001	1001	1001	1001	1000
99999999	1001	1001	1001	1001	1001	1001	1001	1001

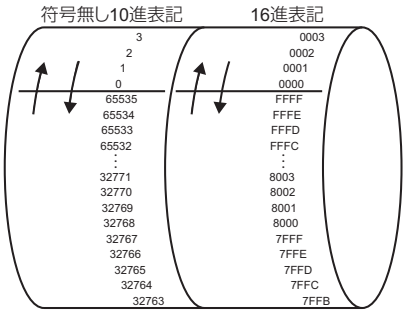
■ 最大値・最小値を超える時の数値の変化

算術演算結果が最大値または最小値を超えた場合、データ型式によりそれぞれ次のように数値が変化します。符号無し/符号付きのどちらのデータとして扱うかはサフィックスで決定します。

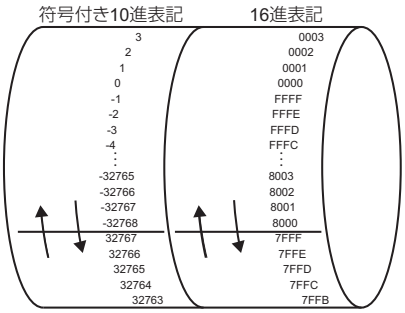
「サフィックス」(4-9ページ)

■ 16ビットデータの場合

65535を超える時の値の変化



最大値、最小値を超える時の値の変化



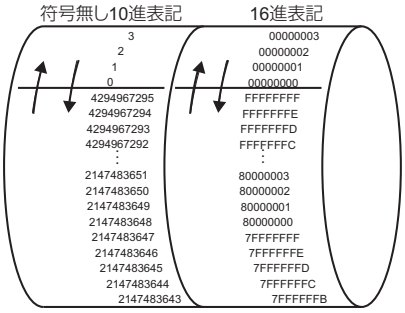
参 考

デバイスを16ビット/32ビットのどちらのデータとして扱うかはサフィックスで決定します。

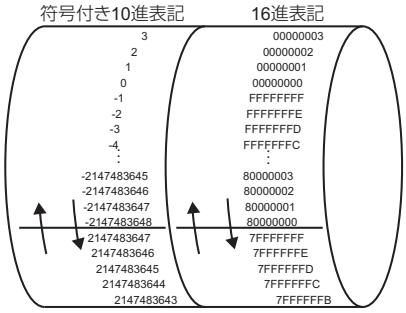
「サフィックス」(4-9ページ)

■ 32ビットデータの場合

4294967295を超える時の値の変化



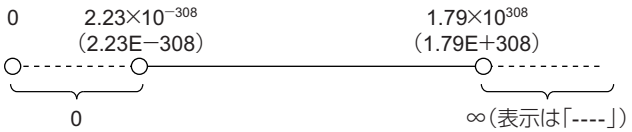
最大値、最小値を超える時の値の変化



参 考

- 単精度浮動小数点型データの場合
  オーバーフロー：絶対値が3.4×10^{38}より大きくなる場合、「∞(無限大)」になります。
  アンダーフロー：絶対値が1.4×10^{-45}より小さくなる場合、「0」になります。

- 倍精度浮動小数点型データの場合
  オーバーフロー：絶対値が1.79×10^{308}より大きくなる場合、「∞(無限大)」になります。
  アンダーフロー：絶対値が2.23×10^{-308}より小さくなる場合、「0」になります。



文字コード

ラダープログラムで文字列を扱うために、それぞれの文字に付ける番号を文字コードといいます。
KV-8000シリーズで使用できる文字コードが扱えます。

■ ASCIIコード

8ビットで1つの半角文字を表す文字コードです。
（例）A･･･$41　　1･･･$31

「ASCIIコード表」（付-18ページ）

■ UNICODE

16ビットで1つの全角文字を表す文字コードです。

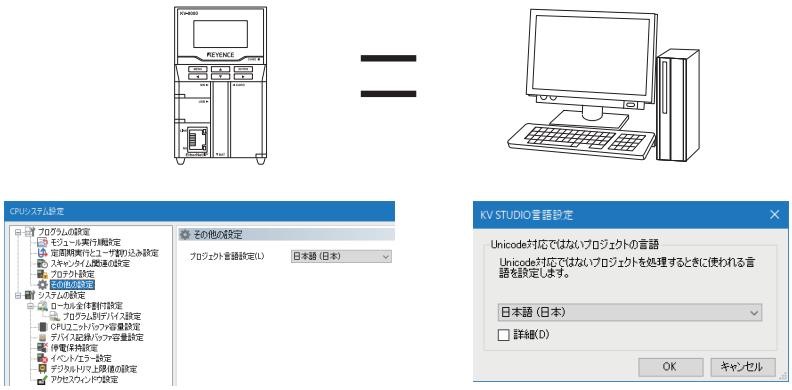
参 考　文字処理命令、カレンダ接点命令のオペランドとして入力できます。文字列を「"」で囲みます。
（例）"型式ABC"　　"月/08:30:15"

UNICODEでは、アルファベット･記号･日本語･中国語･韓国語などが使用できます。使用できない文字コードもありますのでご注意ください。

ポイント

- **UNICODEでは、日本、中国、台湾、韓国の漢字が統合されています。統合とは形が同じとみなせる漢字に同じコードを割り当てることです。このため、日本のフォントと他の国のフォントでは、形が異なる場合があります。**
  例えば、「骨」という字は中国で使われている字形では囲みの中のカギ型の部分の向きが左右逆になります。
- **言語により異なる文字が表示されることを防止するには、PLC内部の言語コードと、KV STUDIOで使用する言語コードを合わせておく必要があります。**
  KV STUDIOの言語設定は、プロジェクトを開いていない状態でのみ変更できます。

PLCの設定　　KV STUDIOの設定

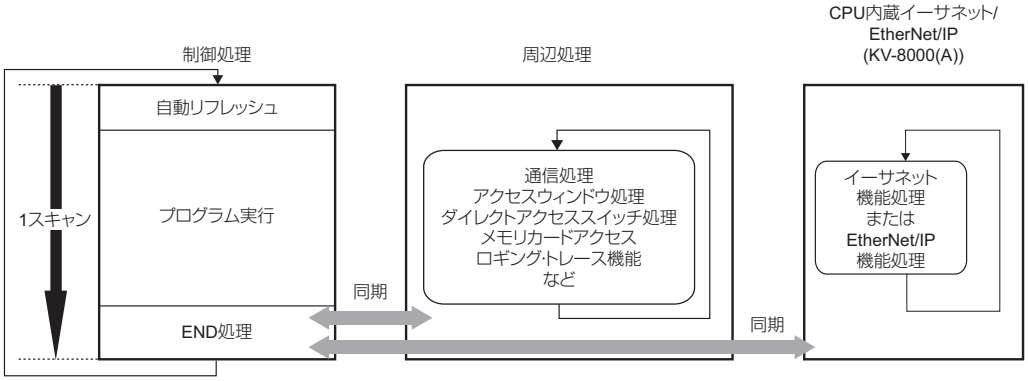


CPUシステム設定→その他の設定　　【ツール(T)】→【KV STUDIO言語設定(2)】

4-8 プログラムの構成と動作

CPUユニットの動作

CPUユニットは次のような手順でプログラムを繰り返し実行します。



制御処理、周辺処理、CPU内蔵イーサネット/EtherNet/IP処理を並列に実行することで高速化しています。

参 考　上記3 つの処理以外に、ユニット割り込み･定周期モジュール/ユニット間同期モジュール実行などによる割り込み処理があります。

「4-13 割り込み」(4-206ページ)

■ 制御処理

1回の制御処理を1スキャンと言い、1スキャンの実行時間がスキャンタイムになります。

● 自動リフレッシュ

CPU内蔵機能や拡張ユニットに割り付けられているデバイスの状態や値を更新します。まず出力するデバイスの状態や値を各ユニットに転送し、次に各ユニットの状態や値を入力デバイスに転送します。

● プログラム実行

ストアードプログラム方式の演算処理を行います。ラダープログラムの先頭ステップから順番に1命令ずつ実行します。

● END処理

同時に並列で行っている周辺処理やCPU内蔵イーサネット/EtherNet/IP機能処理で扱うデバイス値などの各データを制御処理に反映します。また、システム全体のエラーチェックをおこないます。

参 考　通常は、プログラム実行時間に応じてEND処理時間は自動的に決まっていますが、外部機器とのデータ通信量が多い場合などにEND処理時間を設定できます。

「END処理時間の設定」(4-92ページ)

■ 周辺処理

プログラム処理以外の通信やデータ管理などの情報を処理します。また、メモリカードやカレンダタイマなどに関する命令を実行します。制御処理と並列で動作することで、処理内容によるスキャンタイムへの影響が最小限になるように設計されています。

ダイレクト処理について

前述のように、通常は1スキャンに1回のみ自動リフレッシュ処理をおこない、CPU内蔵機能や拡張ユニットに割り付けられたデバイスの状態や値を更新します。プログラム実行中に情報を更新（ダイレクト処理）するには以下のような方法があります。

ポイント　**リアルタイム対応拡張ユニット接続時のみ、以下のダイレクト処理が実行できます。**

■ 拡張入出力ユニット

● ダイレクト入出力リレーDR

命令語のオペランドに入出力リレーを指定する際に、「R」ではなく「DR」で記述すると命令実行時に入出力を更新します。「DR」を使用できる命令語は『KV-8000/7000/5000/3000/1000シリーズ･KV Nanoシリーズ命令語リファレンスマニュアル』を参照してください。

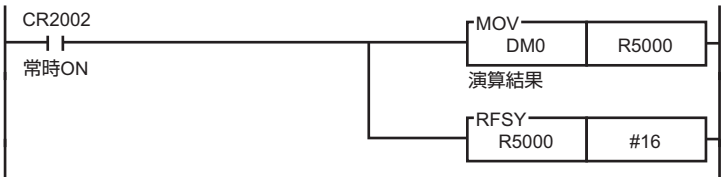
例）

参考　XYM表記の場合、「DX」「DY」で記述します。
「XYM表記について」（4-25ページ）

● RFSX/RFSY命令

命令実行時にオペランドで指定した入出力リレーをまとめて更新します。

例）



『KV-8000/7000/5000/3000/1000シリーズ･KV Nanoシリーズ命令語リファレンスマニュアル』「RFSX/RFSY命令」

■ 拡張特殊ユニット

各ユニットにバッファメモリが設けられていますので、プログラム実行中の任意のタイミングでユニットの情報を取得したり、指令を書き込んだりできます。《KV-8000》のCPU内蔵EtherNet/IP/イーサネットも同様です。

● ユニット専用命令語

各ユニットの用途に応じた専用の命令語により、バッファメモリのアドレスなどを意識せずにユニットとデータ交換をおこなえます。

『各拡張ユニットのユーザーズマニュアル「ユニット専用命令」』

● バッファメモリアクセス（UREAD/UWRIT/UFILL）命令

バッファメモリ読出し命令（UREAD）、バッファメモリ書込み命令（UWRIT）でバッファメモリ内のデータを直接読み書きします。

『KV-8000/7000/5000/3000/1000シリーズ命令語リファレンスマニュアル「UREAD/UWRIT/UFILL命令」』
『各拡張ユニットのユーザーズマニュアル「バッファメモリ一覧」』

■ CPU内蔵機能

CPUユニットは拡張特殊ユニットと同様にダイレクト処理できます。
その他に、フリーランカウンタ現在値を最新の値に更新できる専用命令語があります。
フリーランカウンタ現在値リフレッシュ･･･RFSFRC命令

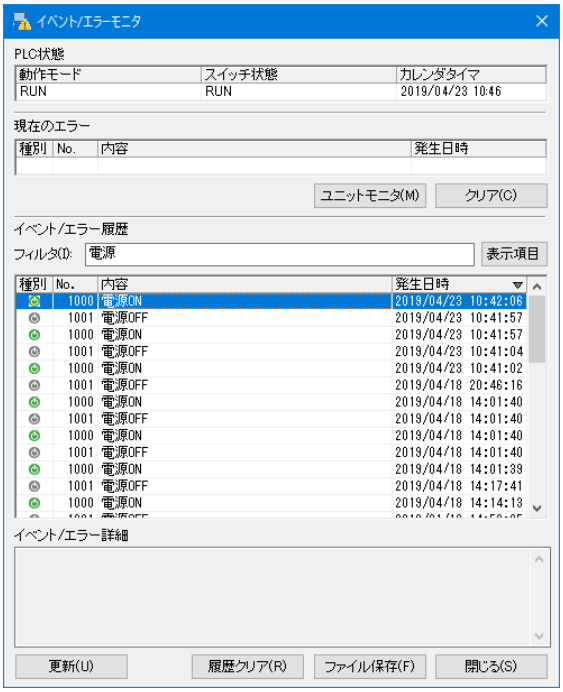
電源ON/OFF時の動作

電源ON時には、以下の項目をチェックして、異常がなければ運転を開始します。

- ハードウェアチェックおよび初期化
- デバイスチェックおよび初期化 「停電保持設定(ラッチ)」(4-26ページ)
- 拡張バスチェックおよび初期化
- ラダープログラムチェックおよび実行準備

■ 電源ON/OFF履歴

電源ON/OFFの履歴をそれぞれ最大20件残すことができます。履歴は不揮発RAMに記録されます。20件を超えると古いものから削除されます。また、《KV STUDIO》でモニタ中にイベント/エラーモニタで確認できます。

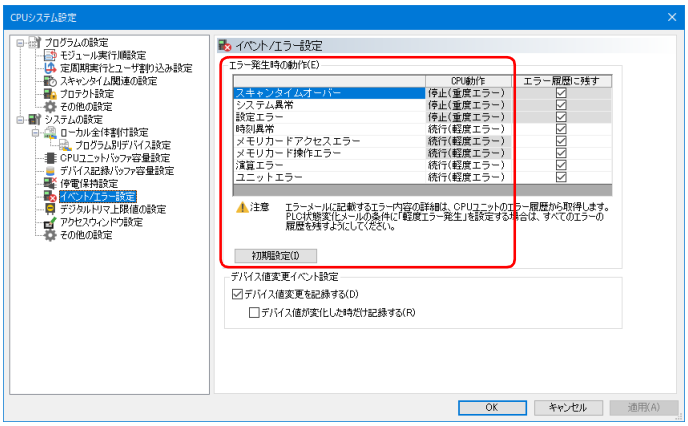


イベント/エラー設定

エラーには運転を停止する重度エラーと運転を継続する軽度エラーがあります。
CPUユニットのエラー詳細は「付-1 PLC本体エラー一覧」(付-2ページ)を参照してください。
拡張ユニットのエラー詳細は『各拡張ユニットのユーザーマニュアル』を参照してください。

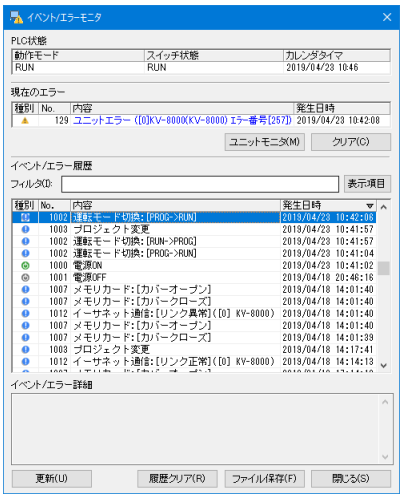
■ イベント/エラー設定

CPUユニットの各エラーの重度/軽度設定は、《KV STUDIO》のワークスペース上から〔CPUシステム設定〕を開いて、「イベント/エラー設定」でおこないます。



■ エラー履歴

重度エラーの履歴は32件、軽度エラーの場合は200件を最大残すことができます。最大件数を超えると古いものから削除されます。また、《KV STUDIO》でモニタ中に〔イベント/エラーモニタ〕で確認できます。



● 最新エラー詳細（CM）

最新エラーの詳細情報がコントロールメモリに記録されます。

コントロールメモリ	説　明	
CM5100	最新重度エラー 詳細情報 1件	年月（10進数で下二桁が月、百の桁から上が年。 例：1210→2012年10月）
CM5101		日
CM5102		時
CM5103		分
CM5104		秒
CM5105		エラー番号
CM5106		詳細情報有効データ数※
CM5107		詳細情報1※
CM5108		詳細情報2※
: :		: :
CM5126		詳細情報20※
CM5150	最新軽度エラー 詳細情報 1件	年月（10進数で下二桁が月、百の桁から上が年。 例：1210→2012年10月）
CM5151		日
CM5152		時
CM5153		分
CM5154		秒
CM5155		エラー番号
CM5156		詳細情報有効データ数※
CM5157		詳細情報1※
CM5158		詳細情報2※
: :		: :
CM5176		詳細情報20※

※ 詳細情報は発生するエラー番号によって次ページのようになります。詳細情報有効データ数に格納された番号までの詳細情報を参照してください。

• No.128（演算エラー）の場合

詳細情報n	説　明
詳細情報1 （CM5157）	エラーの詳細内容 0　（未使用） 1　間接指定が不適切 2　間接指定先がT/C 3　簡易間接指定が不適切 4　オペランド範囲外 5　浮動小数点型実数オーバーフロー 6　デバイス指定が不適切 7　ROOT命令の引数が負 8　ENCO命令でONしているビットがない 9　浮動小数点型実数フォーマット異常 10　三角関数命令の引数が不適切 11　変換できないデータ 12　テーブル指定が不適切 13　時刻比較命令の引数が不適切 14　カムスイッチオペランドが不適切 15　周波数カウンタ命令の引数が不適切 16　0除算 17　ロギングID不正 18　ダイレクト処理エラー 19　ダイレクト処理できないユニット 20　ジャンプ先サブルーチンが存在しない 21　RUN中書き込み中なので命令が実行不可 22　拡張ユニット専用命令対象エラー（ユニット番号） 23　システム予約 24　PIDAT 動作設定範囲外 25　PIDAT サンプリング周期範囲外 26　PIDAT 比例定数範囲外 27　PIDAT 積分定数範囲外 28　PIDAT 微分定数範囲外 29　PIDAT 比例定数2自由度化係数範囲外 30　PIDAT 積分定数2自由度化係数範囲外 31　PIDAT 微分定数2自由度化係数範囲外 32　PIDAT 操作量上限値･下限値範囲外 33　PIDAT 操作量変化幅制限値範囲外 34　PIDAT 測定値変化幅範囲外 35　PIDAT AT状態遷移異常 36　PIDAT 動作制御フラグ範囲外 37　演算周期オーバー 38　PIDAT AT調整パラメータ範囲外 39　PIDAT ATヒステリシス範囲外 40　PIDAT ATタイムアウト時間範囲外 41　PIDAT AT算出制御方式設定範囲外 42　PIDAT ATタイムアウト 43　PIDAT AT比例定数オーバー 44　PIDAT AT積分定数オーバー 45　PIDAT AT微分定数オーバー 46　負の数の累乗根は演算できない 47　ABS命令の引数が不適切 48　演算オーバーフロー 49　UR/UM/UV ユニット未接続 50　ユニットデバイス（R）未割付 51　ユニットデバイス（DM）未割付 52　ユニット種別不一致 53　運転記録ID不正 54　スタックオーバーフロー
詳細情報2 （CM5158）	00H固定
詳細情報3 （CM5159）	エラーが発生しているステップ番号上位
詳細情報4 （CM5160）	エラーが発生しているステップ番号下位

詳細情報n	説　明
詳細情報5～20 （CM5161～CM5176）	モジュール名をプロジェクト言語設定に従って格納します。 （モジュール名以外の部分は00H）

- No.50,53,55,56,58,59,63（ユニットIDに関するエラー）の場合

詳細情報n	説　明
詳細情報1 （CM5107）	ユニット番号（エラーの発生状況によってユニット番号が特定できない場合には、格納されません。）
詳細情報2～20 （CM5108～CM5126）	00H固定

- No.54（ユニットの台数に関するエラー）の場合

詳細情報n	説　明
詳細情報1 （CM5107）	ユニット設定情報でのユニットの台数
詳細情報2 （CM5108）	実際に接続されているユニットの台数
詳細情報3～20 （CM5109～CM5126）	00H固定

- No.127（オートローディング失敗エラー）の場合

詳細情報n	説　明
詳細情報1 （CM5157）	エラーの詳細内容 詳細については、「ロード/セーブ実行時のエラーコード」（5-144ページ）を参照してください。
詳細情報2～20 （CM5158～CM5176）	00H固定

- No.129（ユニットエラー）の場合

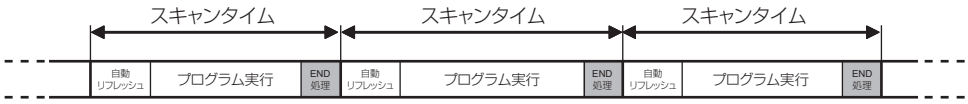
詳細情報n	説　明
詳細情報1 （CM5157）	ユニット番号
詳細情報2 （CM5158）	ユニット毎のエラー番号
詳細情報3～20 （CM5159～CM5176）	00H固定

- 上記以外のエラー番号の場合

詳細情報n	説　明
詳細情報1～20	00H固定

スキャンタイム

制御処理が一巡（自動リフレッシュ→プログラム実行→END処理）するのに必要な時間を、スキャンタイムといいます。スキャンタイムは、プログラムの量や使用している命令語の種類によって変化します。

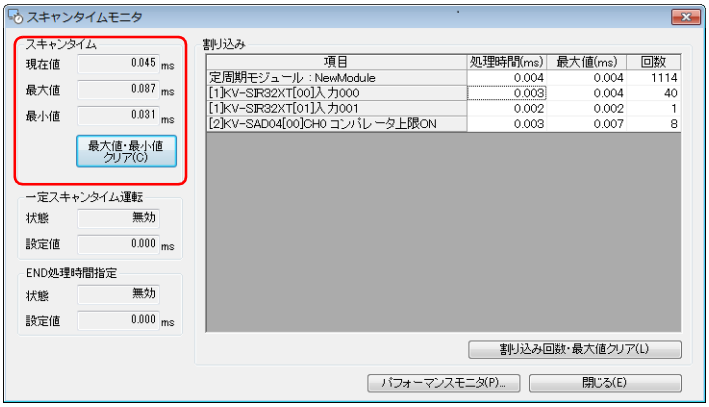


ポイント

スキャンタイムが300msを超えると「スキャンタイムオーバー（CPU本体エラー No.30）」が発生し、運転を継続できません。（PRGモードになります。）

参考

- スキャンタイムの実測値は、《KV STUDIO》でモニタ中にメニューの【デバッグ】→【スキャンタイムモニタ】で確認できます。



- スキャンタイムの実測値は、CM1000～1001（1μs単位）で確認できます。
  「付-2 CR/CM一覧」（付-10ページ）

一定スキャンタイム運転

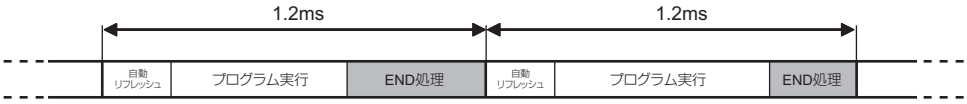
通常のプログラム実行時には、各プログラムの実行状態や処理内容、割り込み処理などによってスキャンタイムが変化します。
一定スキャンタイム運転機能を使用すると、スキャンタイムを一定に保つことができます。

● 一定スキャンタイム運転機能を使用しない場合（通常）

各プログラムが各命令の実行状態によりスキャンタイムが変化します。

● 一定スキャンタイム運転機能を使用する場合

例）設定値を「1.2ms」にした場合



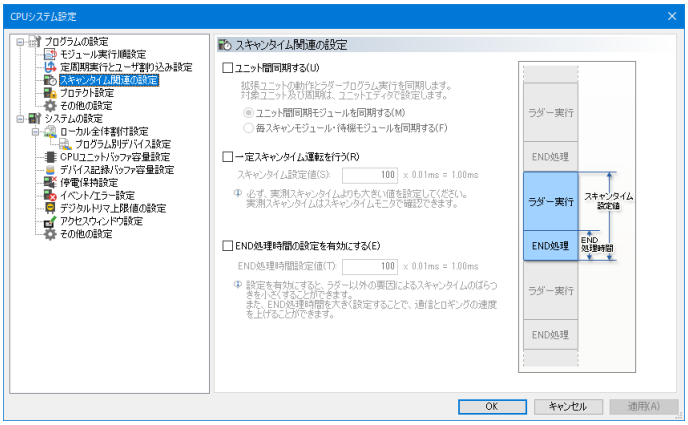
1スキャンの中ですべての処理が完了した後、設定した時間まで次のスキャンに進まずに待ちますので、スキャンタイムを一定にすることができます。

ポイント

- **設定した時間よりスキャンタイムの方が長い場合は、スキャンタイムが延びます。必ず、対象となるプログラムのスキャンタイムの実測値を確認した上で時間設定をおこなってください。**
- **次のスキャンが始まるまでの待ち時間に割り込み要因が発生した場合にも、割り込みプログラムを実行します。（割り込み許可時のみ）割り込みプログラム実行中に一定スキャンタイム運転の設定値に達しても、次のスキャンに進まず、スキャンタイムが割り込みプログラム実行完了まで延びます。**

■ 一定スキャンタイム運転の設定

一定スキャンタイム運転の設定値は、ワークスペースからCPUシステム設定を開いて、【プログラムの設定】→【スキャンタイム関連の設定】で設定します。



設定可能範囲は、「0.10ms～200.00ms」です。

ポイント

実際のスキャンタイムよりも大きい値を設定してください。

参考

- スキャンタイムの実測値は、CM1000～1001（1μs単位）で確認できます。
  「付-2 CR/CM一覧」（付-10ページ）
- スキャンタイム設定値を超えた場合に、CR2304が1スキャンのみONします。

END処理時間の設定

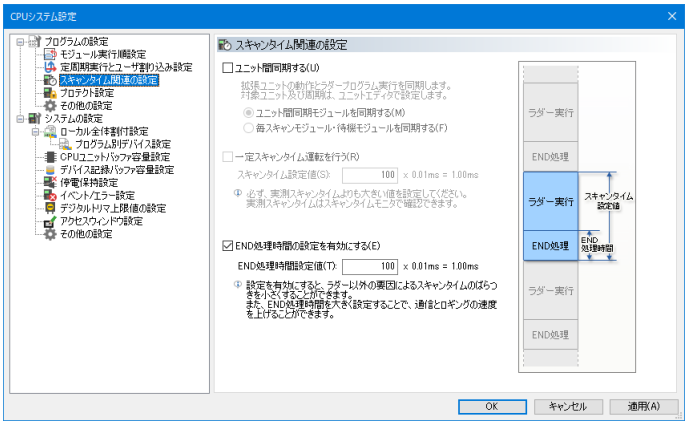
通常のプログラム実行時には、各プログラムの実行状態や処理内容、割り込み処理などによってスキャンタイムが変化します。
END処理時間を設定すると、毎スキャン一定時間のEND処理が実行されます。外部機器との通信データ量が多いにもかかわらず、スキャンタイム最大値が大きいため一定スキャンタイム運転を設定できない場合などに使用してください。

● END処理時間を設定する場合

例）設定値を「2ms」にした場合

■ END処理時間の設定

END処理時間の設定値は、ワークスペースからCPUシステム設定を開いて、【プログラムの設定】→【スキャンタイム関連の設定】で設定します。



設定可能範囲は、「0.02ms～200.00ms」です。

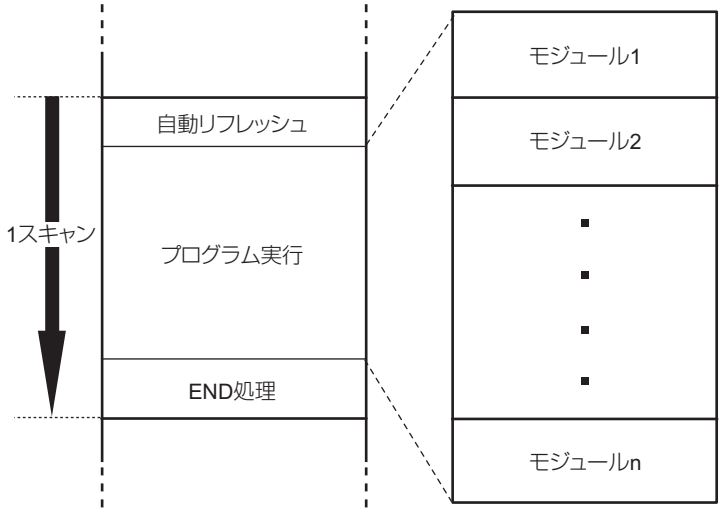
参 考

END処理時間の実測値は、CM1002～1003（1μs単位）で確認できます。

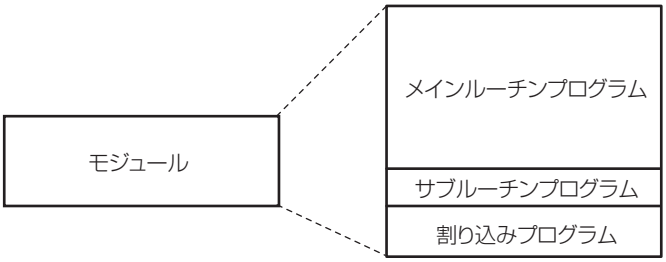
「付-2 CR/CM一覧」（付-10ページ）

プログラムの構成

1スキャンの中でのプログラム実行は、あらかじめ設定した実行順で各モジュール内のメインルーチンプログラムを先頭ステップから順番に実行します。



モジュール内には、それぞれメインルーチンプログラム、サブルーチンプログラム、割り込みプログラムを作成できます。



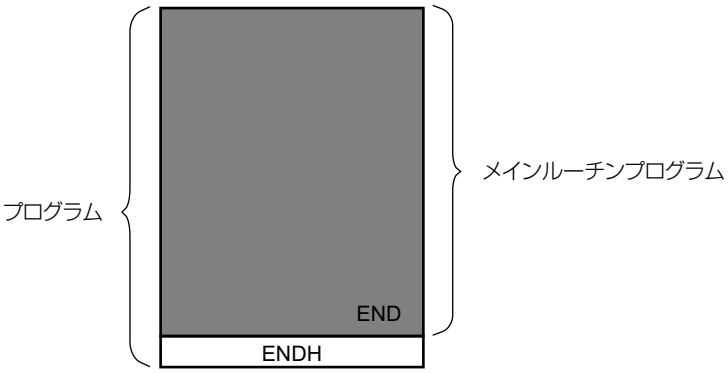
参 考

モジュール内でファンクションブロック、マクロを呼び出しできますが、ファンクションブロック、マクロ自体はメインルーチンプログラム、サブルーチンプログラムで構成します。割り込みプログラムは作成できません。

「4-10 ファンクションブロック」(4-117ページ)
「4-11 マクロ」(4-178ページ)

■ メインルーチンプログラム

プログラム内の先頭からEND(エンド)命令までのプログラムを、メインルーチンプログラムと呼びます。
サブルーチンプログラム、割り込みプログラムを記述しない場合は、プログラムの終了を示すENDH(エンドハイ)命令を、END命令の直後に記述します。



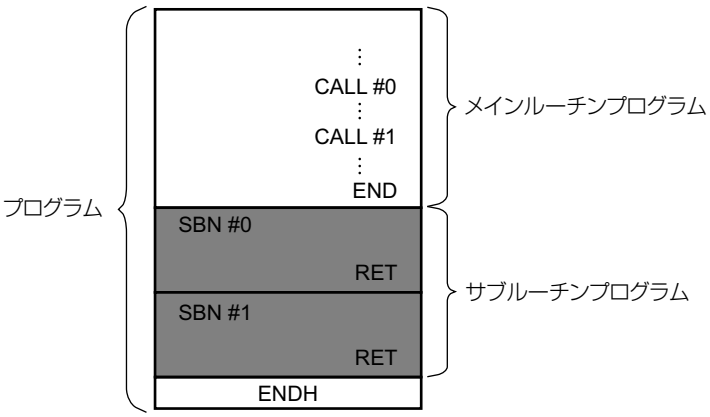
ポイント **END/ENDH命令がない場合は変換エラーになります。**

参考 END/ENDH命令は、プログラムの新規作成時に自動で挿入されるため、削除しないでください。

■ サブルーチンプログラム

サブルーチンプログラムは、メインルーチンプログラム内のCALL(サブルーチンコール) / ECALL(モジュール間サブルーチンコール)命令により実行するプログラムです。CALL命令の実行条件がONの場合、サブルーチンプログラムを実行します。
メインルーチン内の複数の箇所で同じ処理を実行したい場合や、設定した条件を満たした時にだけ処理したいプログラムは、サブルーチンプログラムに記述し、必要に応じて実行させられます。
サブルーチンプログラムは、END～ENDH命令間に最大100個(0～99)作成できます。
各サブルーチンプログラムの先頭はSBN(サブルーチンエントリ)命令で、終了はRET(サブルーチンリターン)命令で示します。
1つのプログラム内に、同じ番号のサブルーチンプログラムは複数記述できません。同じ番号のサブルーチンプログラムに対するCALL命令は、メインルーチンプログラム内に複数記述できます。

『KV-8000/7000/5000/3000/1000シリーズ･KV Nanoシリーズ 命令語リファレンスマニュアル』「CALL/SBN/RET命令」



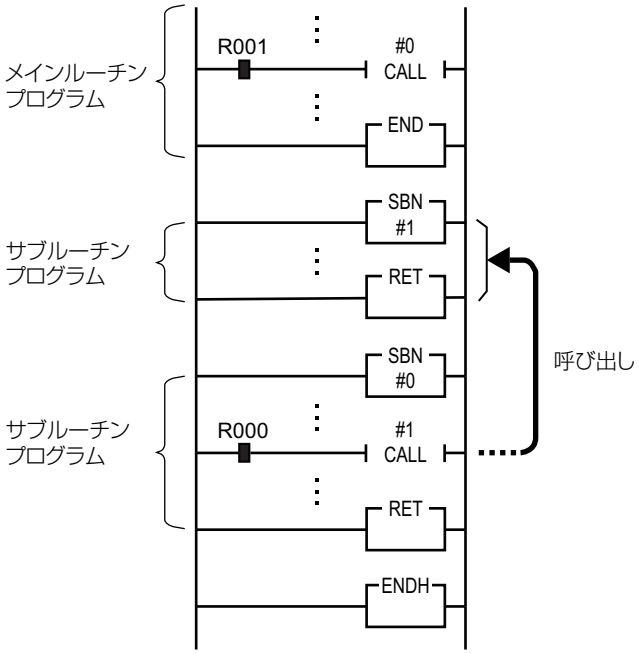
ポイント

- **CALL命令に対応するSBN命令が存在しない場合は変換エラーになります。**
- **ECALL命令に対応するSBN命令が存在しない場合は変換エラーまたは演算エラーになります。**
- **サブルーチンプログラム内で使用できない命令語は、INT/RETI/STG/JMP/ENDSです。**
- **サブルーチンプログラム内で微分実行型命令、タイマ命令、ファンクションブロック命令、マクロ命令を使用する時は注意が必要です。**

『KV-8000/7000/5000/3000/1000シリーズ・KV Nanoシリーズ命令語リファレンスマニュアル』
「微分実行型命令を使用する時の注意」
「タイマ命令使用時の注意」(4-34ページ)
「ファンクションブロック命令」(4-148ページ)
「マクロ命令」(4-193ページ)

参 考

- サブルーチンプログラム内で、別のサブルーチンプログラムを呼び出すこと(ネスティング)はCALL/ECALL命令合計で16重まで可能です。
- 自分自身をCALLすること(再帰呼び出し)は可能ですが、運転時にネスティングが16重を超えると「CALLネスト(CPU本体エラー No.10)」になります。



- 他のプログラム内のサブルーチンを実行する場合は、ECALL命令を使用します。

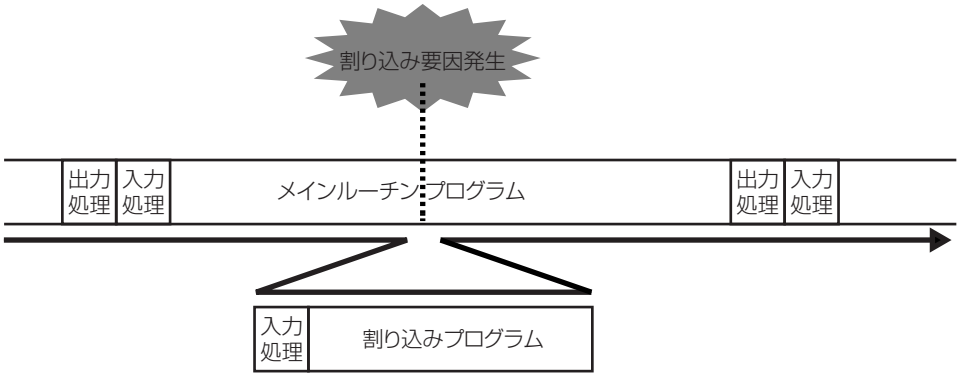
『KV-8000/7000/5000/3000/1000シリーズ・KV Nanoシリーズ命令語リファレンスマニュアル』
「ECALL命令」

- 処理する対象デバイスや値を指定して呼び出す場合、ファンクションブロック、マクロを使用してください。

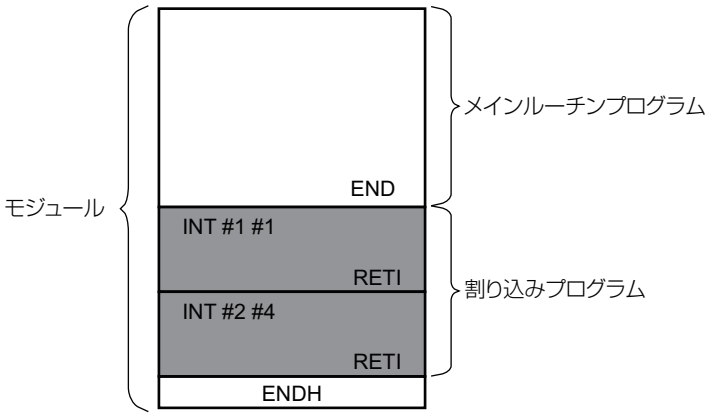
「マクロとサブルーチンプログラムの違い」(4-196ページ)

■ 割り込みプログラム

割り込み機能は短い信号を取り込む時や、入力応答時、高精度の時間測定時に使用します。割り込み要因が発生すると、メインルーチンプログラムの処理を中断して、直ちに割り込みプログラムを実行します。実行が終了すると、中断したところからメインルーチンプログラムの処理を再開します。
割り込み要因にはユニット割り込みのみがあります。



割り込みプログラムは、END～ENDH命令間に記述します。
1つのプロジェクト内には、割り込み要因1個につき1個の割り込みプログラムを作成できます。
割り込みプログラムはどのモジュールに記述してもかまいません。
各割り込みプログラムの先頭はINT（イント）命令、終了は RETI（リターンイント） 命令で示します。

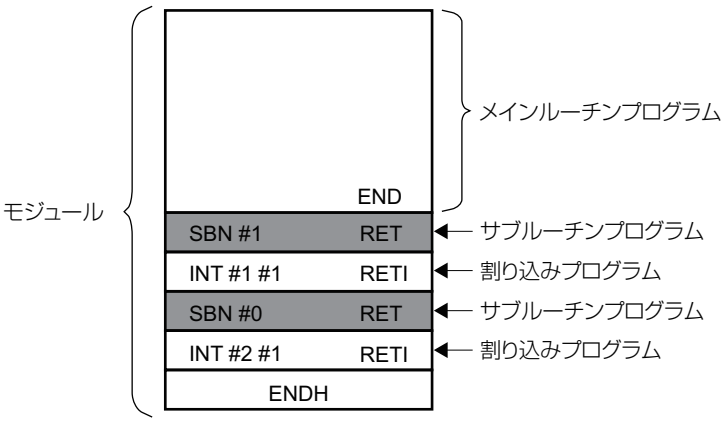


ポイント

- **割り込みプログラムはファンクションブロック、マクロ内に記述できません。**
- **割り込みプログラムを実行するためには、EI命令により割り込みプログラムの実行を「許可」にしてください。（運転開始時は割り込み「禁止」の状態になっています。）**
- **内部レジスタ、演算フラグ（CR2009～CR2012）、TM0～TM3、Z11～Z23は、割り込みプログラム開始時に自動的に退避され、終了時に復帰します。**
- **割り込みプログラム内では使用できない命令語があります。**
  「割り込みプログラムで使用できない命令語」（4-210ページ）

参 考

サブルーチンプログラムや割り込みプログラムを記述する順序に制限はありません。



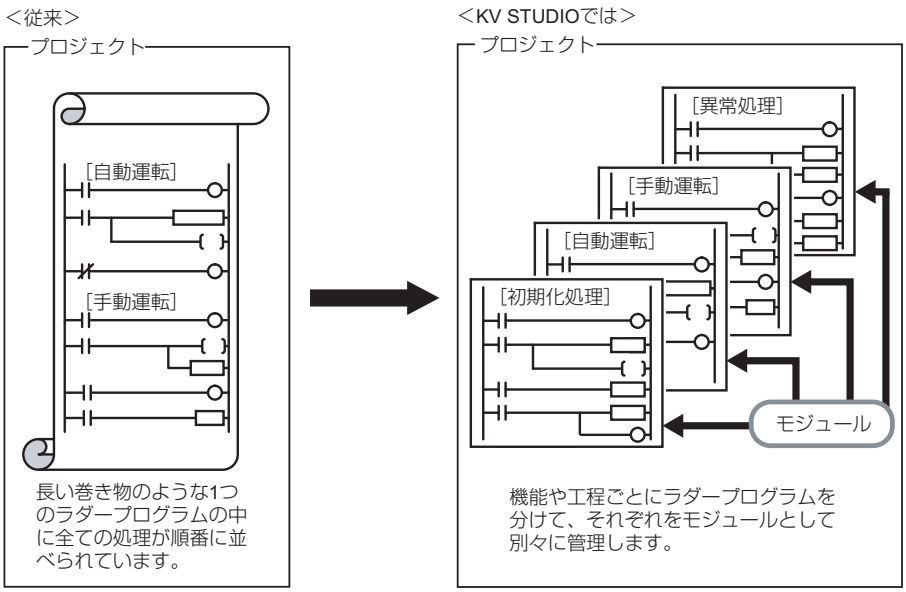
- モジュールが実行中でない場合も、割り込みプログラムは実行されます。
- モジュールの実行状態にかかわらず、割り込みプログラムは実行されますので、1 つの待機モジュールを割り込みプログラム専用として管理すると、プロジェクト全体の構成をより把握しやすくなります。

「4-13 割り込み」(4-206ページ)

4-9 モジュール

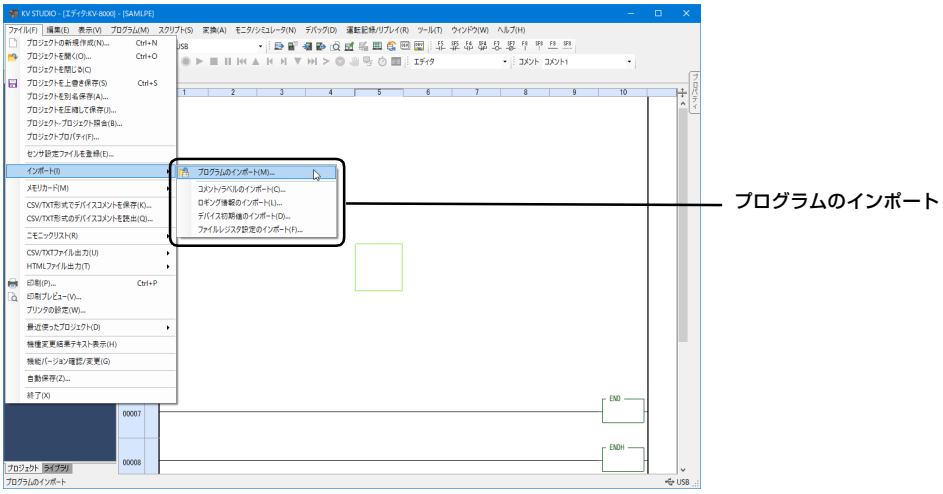
概要

装置を制御する一般的なラダープログラムには、例えば初期化処理･自動運転･手動運転･異常処理など複数の処理内容が含まれています。従来は、それら複数の処理を1つの連続したラダープログラムとして管理していました。《KV STUDIO》では、1つのプロジェクト内に処理･工程･機能などの単位で別々にプログラムを作成し、それらを独立した部品(モジュール)として管理できます。



■ モジュールの管理

《KV STUDIO》では、モジュールを各プロジェクトごとに[ワークスペース]を使用して管理します。プログラムのインポート機能を使用してプログラムの再利用が簡単にできます。



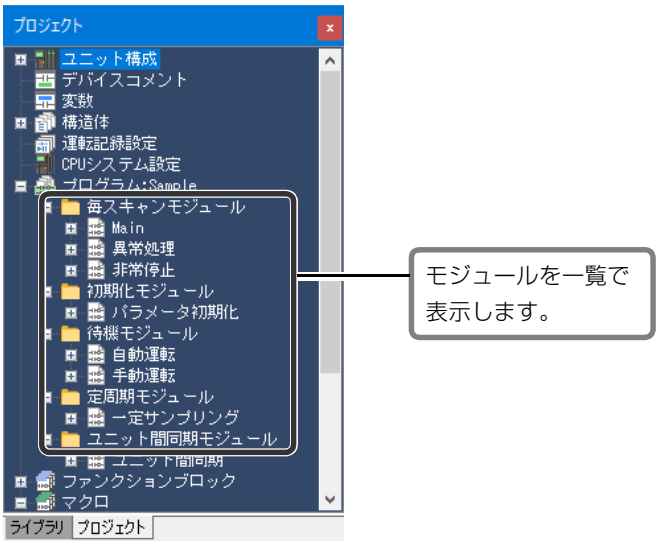
[ワークスペース]を表示するには、メニューの【表示(V)】から各項目を選択してください。また、ツールバーのアイコンからも表示の切り換えができます。
表示位置は変更可能です。

参 考 《KV STUIDO》を複数起動して、アプリケーション間でモジュールをコピーすることでも再利用できます。

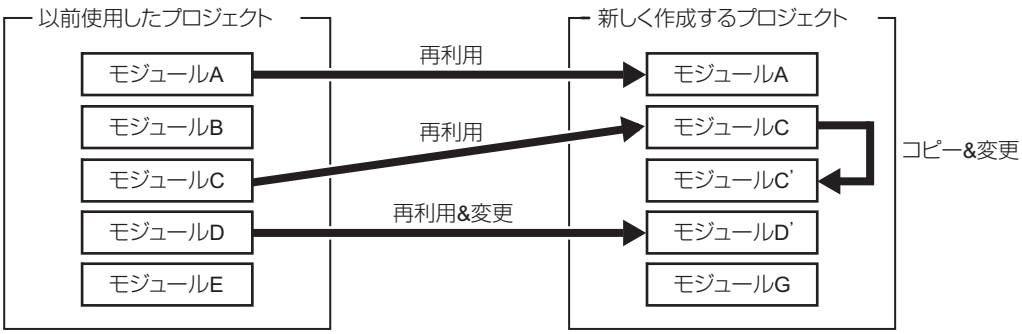
■ モジュールの特長

● プログラム全体の構成が把握しやすい

《KV STUDIO》の[ワークスペース]を使用すると、各モジュールがツリー表示されるため、プログラム作成者以外の人でもプログラムの全体構成が一目で把握できます。



● プログラムの必要な部分のみを再利用しやすい



● プログラム作成・変更やデバッグが効率良くできる

機能や工程別にモジュールを分けることによって、プログラムを作成・デバッグする際に考慮すべき範囲を少なくできます。また、ローカルデバイスを使用することで、モジュールごとにデバイスの割り付けを意識せずにすむため、複数人でのプログラム作成がしやすくなります。

「4-12 ローカルデバイス」(4-200ページ)

● パスワードが設定できる

モジュールの内容を容易に変更できなくするため、パスワードを設定できます。
パスワードを設定した場合、パスワードを入力しないとモジュールのプログラムを表示、編集できません。
[プログラムプロパティ]ダイアログでパスワードを設定します。

「モジュールパスワード」(4-114ページ)
「ファンクションブロックのパスワード設定」(4-141ページ)
「マクロパスワード」(4-191ページ)

モジュールの種類

モジュールには、「毎スキャンモジュール」「初期化モジュール」「待機モジュール」「定周期モジュール」「ユニット間同期モジュール」があります。
《KV STUDIO》のワークスペース上で、種類別に作成したモジュールを管理します。モジュール作成後に種類を変更することも可能です。

毎スキャンモジュール

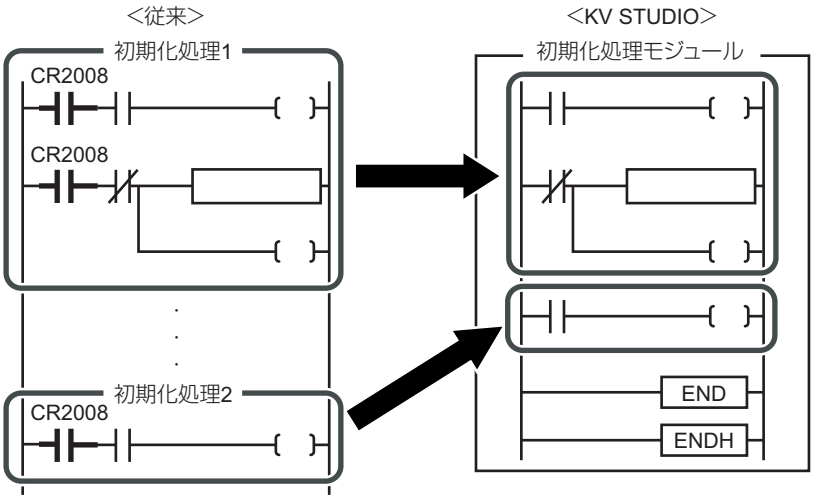
毎スキャンモジュールは、運転開始時（電源ON、PRG→RUN）の1スキャン目から実行します。
モジュール命令（MDSTOP（モジュールストップ）/MDSTRT（モジュールスタート））を使用して、運転中にモジュールの実行状態を制御できます。
モジュール命令を使用した際の動作は、待機モジュールと同様です。

「待機モジュールの動作」（4-101ページ）

参考
- モジュール命令を使用して実行状態を制御しない場合は、毎スキャン実行されます。
- 《KV STUDIO》でプロジェクトを新規作成すると、プロジェクト名と同じ名前の毎スキャンモジュールが自動的に1個作成されます。

初期化モジュール

初期化モジュールは、運転開始時の1スキャン目にのみ実行します。装置の初期化処理や《KV-8000シリーズ》の各機能の初期設定などに使用します。プロジェクト内に初期化モジュールがなくても《KV-8000シリーズ》は運転可能です。必要に応じて作成してください。



CR2008:運転開始時1スキャンのみON

初期化モジュールとして登録することで、2スキャン目からは初期化実行のプログラム部分を処理する時間が必要ないため、その分だけスキャンタイムを短くできます。

ポイント
- **初期化モジュール内で使用できない命令語があります。**
  「使用できない命令語一覧」（付-21ページ）
- **初期化モジュールは、モジュール命令（MDSTRT/MDSTOP）で実行状態を制御できません。**
- **初期化モジュールだけでは《KV-8000シリーズ》を運転できません。必ず1つ以上の毎スキャンモジュールまたは待機モジュールと合わせて使用してください。**

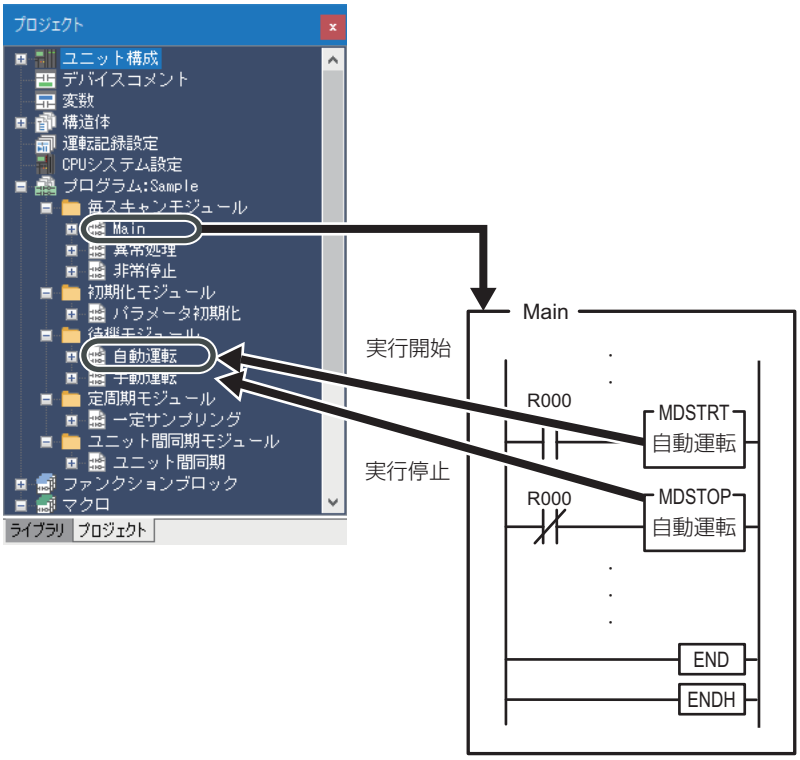
待機モジュール

待機モジュールは、運転開始時に停止しています。モジュール実行命令（MDSTRT/MDSTOP）を使用して、運転中に実行状態を制御できます。

「モジュール命令」（4-115ページ）

＜待機モジュール呼び出し例＞

毎スキャンモジュール「Main」から待機モジュール「自動運転」の実行状態を制御しています。



ポイント **待機モジュールだけでは《KV-8000シリーズ》を運転できません。必ず1つ以上の毎スキャンモジュールまたは初期化モジュールと合わせて使用してください。**

参考 毎スキャンモジュールも同様に実行状態を制御できます。

■ 待機モジュールの動作

● 待機モジュールの実行開始

モジュール実行開始命令が呼び出されたスキャンでは、待機モジュールは実行されません。次のスキャンから実行されます。

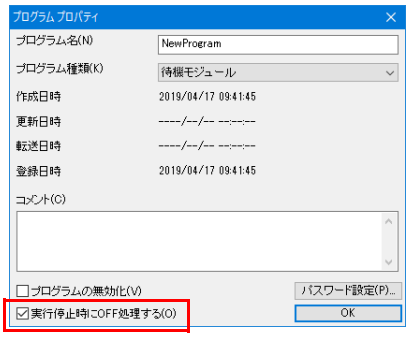
参考 待機モジュール実行開始時における微分実行型命令の初期状態は、立ち上がり検出の場合がON、立ち下がり検出の場合がOFFになります。例えば実行開始時の1スキャン目からONしている場合には、立ち上がり微分を検出しません。

● 待機モジュールの実行停止

待機モジュールの実行停止時(MDSTP命令実行時)の動作は、各プログラムの[プログラムプロパティ]→「実行停止時にOFF処理する」の設定により異なります。

ポイント

KV-8000Aの場合は設定は表示されず、常に「実行停止時にOFF処理する」にチェックしている場合と同じ動作になります。



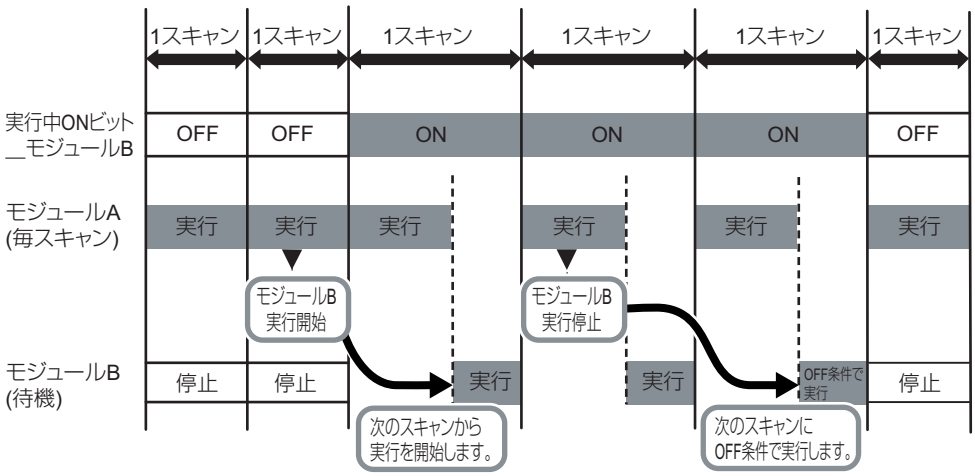
「実行停止時にOFF処理する」にチェックしていない場合

待機モジュールは、モジュール実行停止命令(MDSTP)が実行された次のスキャンで停止します。

「実行停止時にOFF処理する」にチェックしている場合

プログラム実行中に停止命令が呼び出された場合、呼び出したスキャンでは実行します。
呼び出した次のスキャンでは、実行条件をすべてOFFとして実行されます。※
さらに、次(呼び出し後の2つ目)のスキャンで停止します。

待機モジュールの動作イメージを以下の図に示します。



※ ただし、以下の命令は例外となります。

命令語	状態
OUB命令	OFFします。
OFDL命令	リセットします。
SHOT命令	
C命令	状態保持します。
CJ/NCJ/GOTO命令	分岐しません。

参 考

モジュール内のラダープログラムは、停止前の1スキャンのみMC命令の実行条件がOFF時と同じ動作をします。

『KV-8000/7000/5000/3000/1000シリーズ·KV Nanoシリーズ命令語リファレンスマニュアル』「MC/MCR」

定周期モジュール

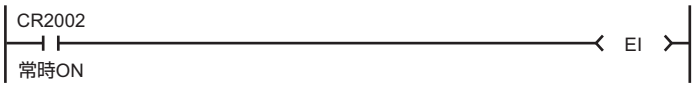
定周期モジュール内のプログラムは、通常のスキャンとは別にあらかじめ設定した周期で実行されます。定周期モジュールは、1プロジェクトにつき4個使用できます。

■ 定周期モジュールの実行方法

定周期モジュールを実行するために、定周期モジュール作成以外に必要な項目は以下のとおりです。

● 割り込み許可(EI)命令

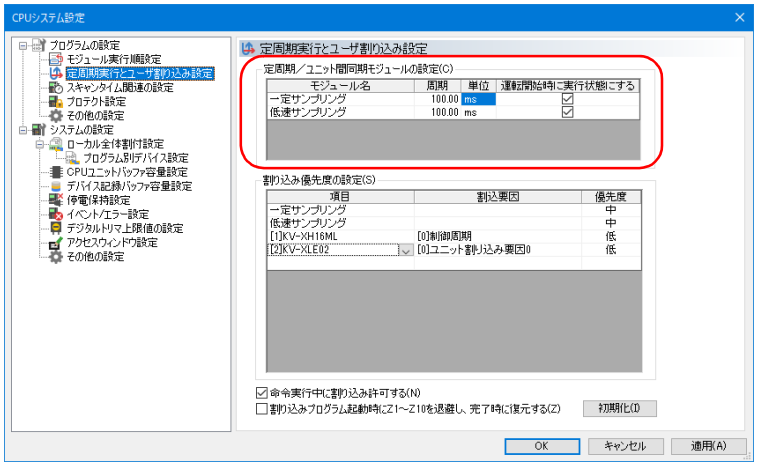
定周期実行の許可/禁止は、割り込みプログラムと同様にEI/DI命令でおこないます。
毎スキャンモジュールや待機モジュール内に割り込み許可(EI)命令を記述して実行します。
例)



「DI/EI命令」(4-218ページ)

● CPUシステム設定

《KV STUDIO》のワークスペースから[CPUシステム設定]を開いて、[プログラムの設定]の〔定周期実行とユーザ割り込み設定〕で実行周期と優先度を設定します。



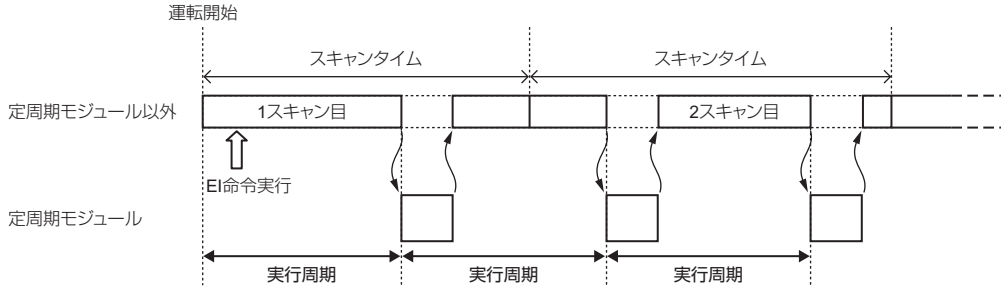
[定周期モジュールの設定]
設定できる周期は、0.01s～60.00s(0.01s刻み)または0.05ms～6000.00ms(0.01ms刻み)です。
定周期モジュールを運転開始時に実行状態にする場合は、「運転開始時に実行状態にする」にチェックを付けます。運転中に実行を開始/停止するには、モジュール命令(MDSTRT/MDSTOP)を使用します。
[割り込み優先順位の設定]
優先度を「中」にした定周期モジュールには、優先度を「高」に設定した割り込みプログラム、定周期モジュールのみ、多重に割り込み可能です。「中」「低」に設定した割り込みプログラム、定周期モジュールは割り込むことはできません。
また、優先度「低」に設定した割り込みプログラム実行中に優先度「中」の定周期モジュールは割り込んで実行できますが、「高」「中」に設定した割り込みプログラム、定周期モジュール実行中は、実行を待たされます。

ポイント **定周期モジュールの実行周期はKV STUDIOからのみ設定できます。ラダープログラムからは変更できません。**

■ 定周期モジュールの動作

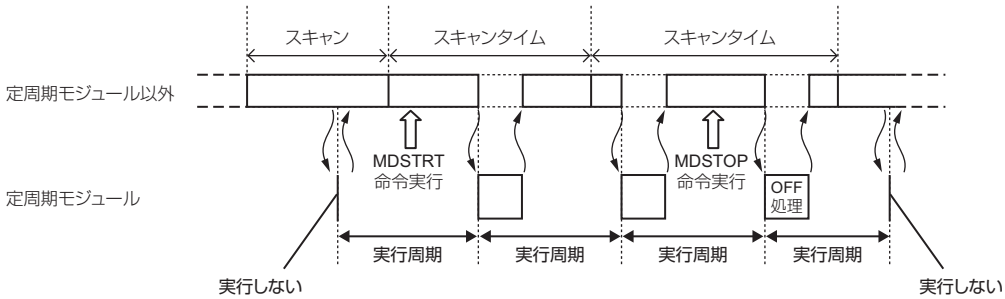
● 「運転開始時に実行状態にする」にチェックを付けている場合（デフォルト）

EI命令を実行すれば、定期周期モジュールの実行を開始します。



● 「運転開始時に実行状態にする」にチェックを付けていない場合

EI命令が実行されている状態で、モジュール命令を使用して定周期モジュールの実行状態を制御します。



OFF処理については「待機モジュールの実行停止」（4-102ページ）を参照してください。

ポイント

実行周期は、定周期モジュールの実行状態にかかわらず運転開始時からカウントされます。定周期モジュールを実行状態にした状態で運転中にEI命令を実行する場合、EI命令実行から定周期実行周期を待たずに実行されることがあります。

参考

- 以下の条件をすべて満たす場合、周期のばらつきは15μs以内になります。
  - 〔CPU　システム設定〕の〔定周期実行とユーザ割り込み設定〕で、「命令実行中に割り込み許可する」をチェックしている場合
  - 定周期モジュールより優先度の高いユーザ割り込みを使用していない
  - 優先度が高い割り込み要因が実行されていない
  - 優先度が低い割り込み要因が直前に実行されていない（実行時は最大4μs加算）
  - KV-5000/3000シリーズ用拡張ユニットを使用していない（使用時は最大14μs加算）
  - DI命令/DIC命令による割り込み禁止をしていない
  - ロギング/トレース設定で、1つのIDで513ワード以上のデバイスを登録していない
- 割り込み禁止状態（DI　命令実行時）に定周期実行のタイミングが発生した場合、同じ実行周期内は割り込み許可状態（EI命令実行）になり次第、定周期モジュールを実行します。

定周期モジュールは、通常のスキャンとは別に割り込み処理として実行します。割り込み処理の詳細は

「4-13 割り込み」（4-206ページ）を参照してください。

■ 開始/終了時のデバイスの処理について

通常の割り込み処理と同様に定周期モジュール実行開始/終了時に、以下の項目の退避/復帰を自動的におこないます。

- 内部レジスタ
- 演算フラグ（CR2009～CR2012）
- インデックスレジスタ（Z11～Z23）
- TM0～TM3
- 直前に実行したSTG命令のオペランドに記述しているリレー

参 考　インデックスレジスタZ01～Z10は、[CPUシステム設定]→[定周期実行とユーザ割り込み設定]で退避/復帰の処理をするように設定可能です。

ユニット間同期モジュール

ユニット間同期モジュール内のプログラムは、通常のスキャンとは別にあらかじめ設定した周期で実行されます。ユニット間同期モジュールは、1つのプロジェクトにつき1個使用できます。

■ ユニット間同期モジュールの実行方法

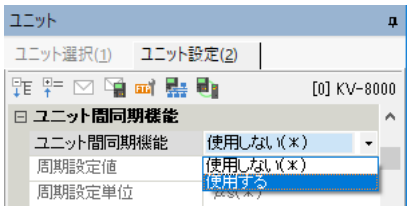
ユニット間同期モジュールを実行するために、ユニット間同期モジュール作成以外に必要な項目は以下のとおりです。

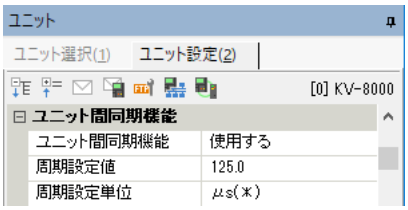
● ユニット間同期の設定

ユニットエディタで、以下の内容を設定します。

CPUユニットの設定

CPUユニットのユニット設定のユニット間同期機能を「使用する」に変更します。
その後、周期の設定値を変更します。



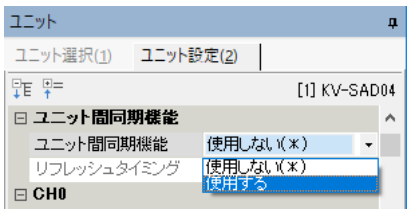


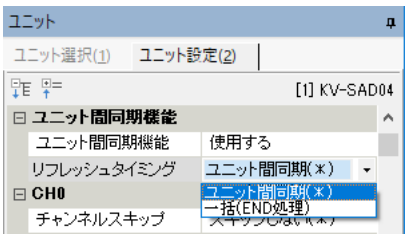
ポイント　**周期に設定できる値はユニット間同期をおこなう拡張ユニットの設定により異なります。各拡張ユニットのマニュアルを確認してください。**

拡張ユニットの設定

実行する拡張ユニットのユニット間同期機能を「使用する」に設定します。
必要に応じてリフレッシュタイミングの設定を変更します。

「ユニット間同期リフレッシュデバイス」(5-173ページ)

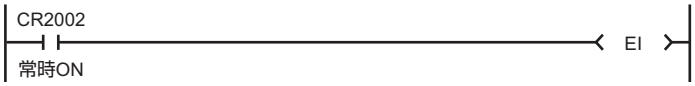




● ユニット間同期の許可設定

ユニット間同期の許可/禁止は割り込みプログラムと同様にEI/DI命令でおこないます。毎スキャンモジュールや待機モジュール内に割り込み許可(EI)命令を記述して実行します。

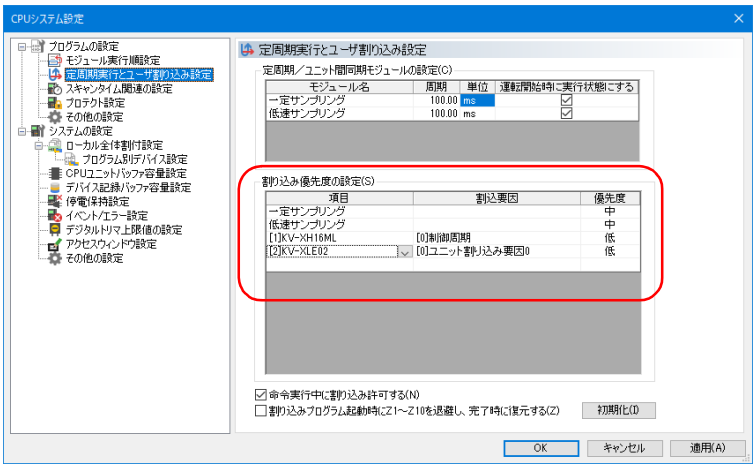
例)



「DI/EI命令」(4-218ページ)

● CPUシステム設定

《KV STUDIO》のワークスペースから[CPUシステム設定]を開いて、[プログラムの設定]の〔定周期実行とユーザ割り込み設定〕で優先度を設定します。



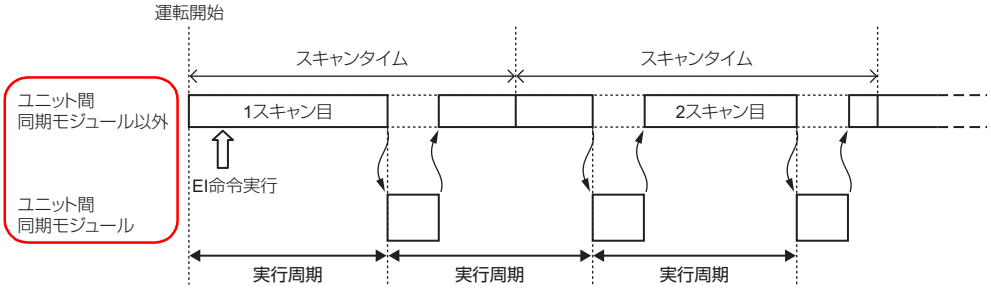
[割り込み優先順位の設定]

優先度を「高」にしたユニット間同期モジュールには、他の割り込みプログラム、定周期モジュールは割り込むことはできません。ユニット間同期モジュール実行中は、実行を待たされます。

■ ユニット間同期モジュールの動作

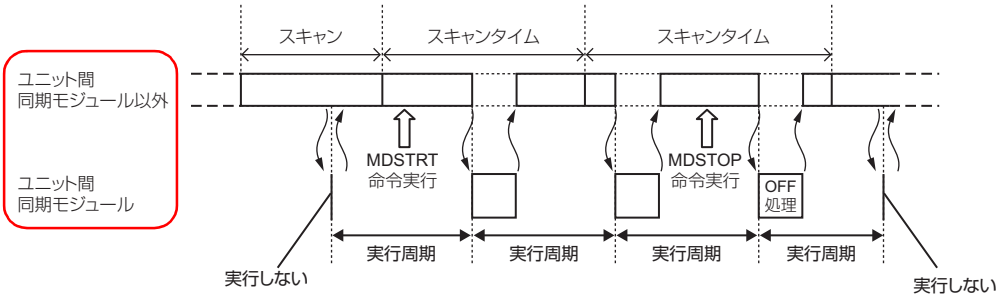
● 「運転開始時に実行状態にする」にチェックを付けている場合（デフォルト）

EI命令を実行すれば、ユニット間同期モジュールの実行を開始します。



● 「運転開始時に実行状態にする」にチェックを付けていない場合

EI命令が実行されている状態で、モジュール命令を使用してユニット間同期モジュールの実行状態を制御します。



OFF処理については「待機モジュールの実行停止」（4-102ページ）を参照してください。

ポイント

実行周期は、ユニット間同期モジュールの実行状態にかかわらず運転開始時からカウントされます。ユニット間同期モジュールを実行状態にした状態で運転中にEI命令を実行する場合、EI命令実行からユニット間同期実行周期を待たずに実行されることがあります。

参考

- 以下の条件をすべて満たす場合、周期のばらつきは15μs以内になります。
  - 〔CPUシステム設定〕の〔定周期実行とユーザ割り込み設定〕で、「命令実行中に割り込み許可する」をチェックしている場合
  - ユニット間同期モジュールより優先度の高いユーザ割り込みを使用していない
  - 優先度が高い割り込み要因が実行されていない
  - 優先度が低い割り込み要因が直前に実行されていない（実行時は最大4μs加算）
  - KV-5000/3000シリーズ用拡張ユニットを使用していない（使用時は最大14μs加算）
  - DI命令/DIC命令による割り込み禁止をしていない
- 割り込み禁止状態（DI 命令実行時）にユニット間同期実行のタイミングが発生した場合、同じ実行周期内は割り込み許可状態（EI命令実行）になり次第、ユニット間同期モジュールを実行します。

ユニット間同期モジュールは、通常のスキャンとは別に割り込み処理として実行します。割り込み処理の詳細は「4-13 割り込み」を参照してください。

■ 開始/終了時のデバイスの処理について

通常の割り込み処理と同様にユニット間同期モジュール実行開始/終了時に、以下の項目の退避/復帰を自動的におこないます。

- 内部レジスタ
- 演算フラグ（CR2009～CR2012）
- インデックスレジスタ（Z11～Z23）
- TM0～TM3
- 直前に実行したSTG命令のオペランドに記述しているリレー

参 考　インデックスレジスタZ01～Z10は、[CPUシステム設定]→[定周期実行とユーザ割り込み設定]で退避/復帰の処理をするように設定可能です。

モジュールの実行順序

プロジェクトに登録したモジュールの1スキャン内での実行順序を設定できます。
通常は、モジュールを作成した順になっています。1スキャン内でのラダープログラムの実行順序が動作に影響する場合にのみ変更してください。
ワークスペースから[CPUシステム設定]を開き、[プログラムの設定]→[モジュール実行順の設定]を選択します。

順番	モジュール名	モジュールの種類
1	位置決め初期化	初期化
2	高速カウンタ初期化	初期化
3	位置決め	毎スキャン
4	ログ情報更新	待機
5	高速カウンタ	毎スキャン
6	通信処理	待機
7	メイン運転処理	毎スキャン

順序の入れ替え可能な範囲

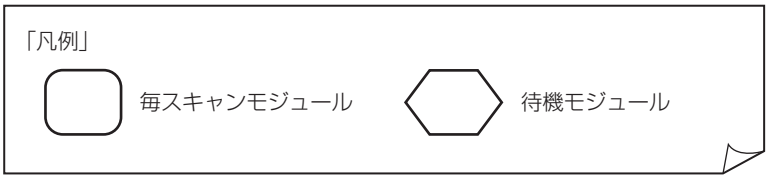
順序の入れ替え可能な範囲

1スキャン内でのモジュールの実行順序設定には、次のようなルールがあります。

- 初期化モジュール内で順序の入れ替えは可能ですが、その他の種類のモジュールとの順序の入れ替えはできません。
- 毎スキャンモジュールと待機モジュールの実行順序の入れ替えは可能ですが、初期化モジュールより前に移動できません。

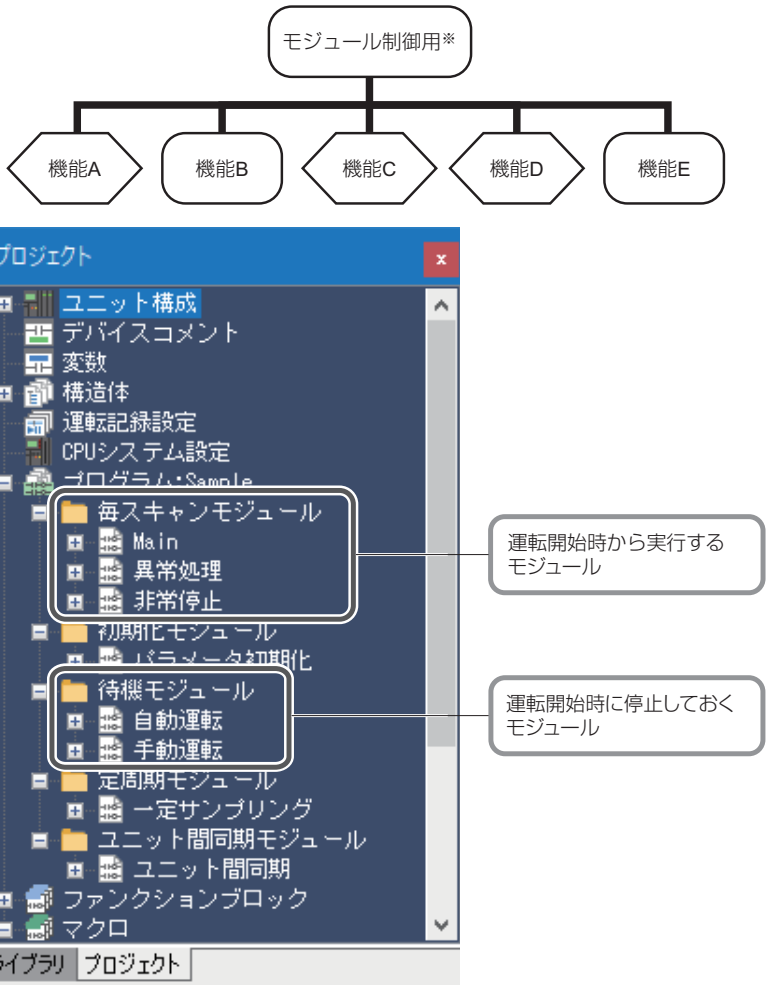
モジュールの構成例

モジュール構成の一例を示します。
説明を簡略化するために、初期化モジュールは省略しています。



● 機能別

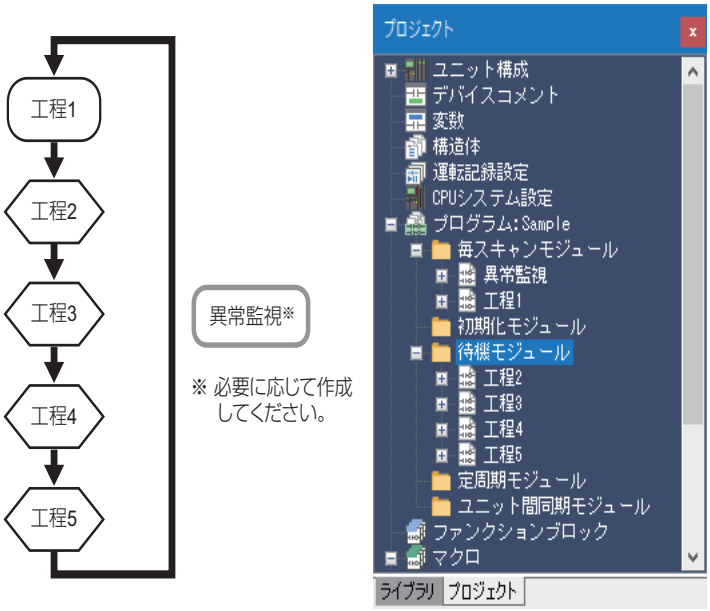
処理の内容別、接続する機器別、拡張ユニット別にモジュールを作成します。
毎スキャンモジュールと待機モジュールは、運転開始時の実行状態で使いわけてください。
必要のないモジュールを停止することで、デバッグがしやすくなり、スキャンタイムも短縮できます。



※ 「モジュール制御用」モジュールで、各機能の実行状態を制御します。必要に応じて作成してください。

● 工程別

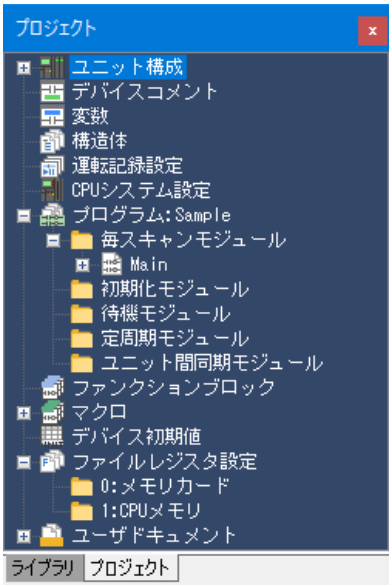
工程別、作業別にモジュールを作成します。
ここでは、運転開始時に実行するモジュールを毎スキャンモジュールで、それ以外を待機モジュールで作成しています。



参 考　モジュールの実行順は1スキャン内での実行順序のため、工程別にプログラムを作成する場合もモジュールの実行順を意識する必要はありません。

● モジュールを意識せずにプログラムする場合

プロジェクトの新規作成時に、「Main」という名称の毎スキャンモジュールが自動的に1つ作成されるため、モジュールを意識せずにプログラムを作成できます。



モジュールシステムデバイス

モジュールシステムデバイスとは、モジュールの動作状況を取得するためのデバイスです。
実行中ONビットのみ、他のプログラムから参照可能です。

ポイント　**モジュールシステムデバイスには、デバイスコメントを付けられません。**

参考　モジュールシステムデバイスは、《KV STUDIO》によってCPUユニットのワーク領域へ自動的に割り付けられます。

● モジュール実行開始時1スキャンのみONビット「@CR2008」

モジュールが実行を開始した1スキャン目のみONになります。
2スキャン目以降はOFFになります。
自モジュール内のみ参照可能です。自モジュール内で実行する各マクロ内では参照できません。
《KV STUDIO》上での表記は、「@CR2008」となります。

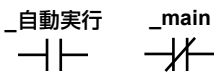
● モジュール実行開始時1スキャンのみOFFビット「@CR2007」

モジュールが実行を開始した1スキャン目のみOFFになります。
2スキャン目以降はONになります。
自モジュール内のみ参照可能です。自モジュール内で実行する各マクロ内では参照できません。
《KV STUDIO》上での表記は、「@CR2007」となります。

● 実行中ONビット「_（モジュール名）」

実行中ONビットは、そのモジュールが実行中にONします。
自モジュール内だけでなく、他のプログラムからも参照できるため、待機モジュールの実行を制御する場合などに有効です。
《KV STUDIO》上での表記は、「_（モジュール名）」（アンダースコア＋モジュール名）となります。

例



モジュールパスワード

■ モジュールのパスワード

モジュールにパスワードを設定して、《KV STUDIO》でモジュールの表示に制限をかけることができます。
次の手順でパスワードを設定します。

1 **[ワークスペース]上の設定したいモジュールを選択します。**

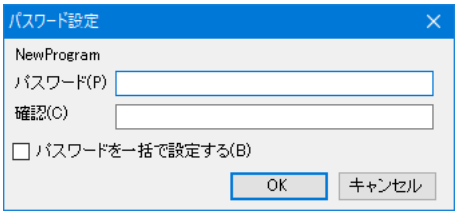
2 **右クリックして表示されるメニューから【プロパティ(A)】を選択します。**

[プログラムプロパティ]ダイアログが表示されます。

別手順
・ワークスペースでモジュールを選択し、メニューから【プログラム(M)】→【プロパティ(H)】を選択
・ワークスペースでモジュールを選択し、右クリックメニューから【パスワード設定(B)】を選択

3 **〔パスワード設定〕ボタンを押します。**

[パスワード設定]ダイアログが表示されます。

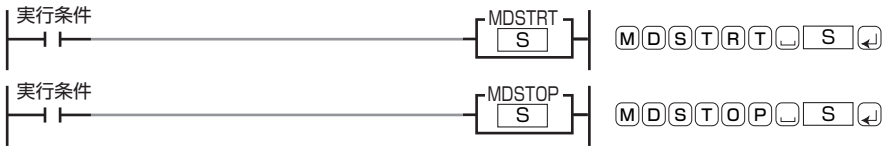


4 **パスワードとパスワードの確認を入力して、〔OK〕ボタンを押します。**

パスワード設定後、パスワード設定したモジュールを《KV STUDIO》で表示する時、[パスワード解除]ダイアログが表示されます。
パスワードを入力しないと、モジュールを《KV STUDIO》上で表示できません。

MDSTRT	MDSTRT	モジュール実行開始	指定したモジュールの実行を開始します。
MDSTOP	MDSTOP	モジュール実行停止	指定したモジュールの実行を停止します。

ラダープログラム　　入力方法



オペランド	説明	占有サイズ
S	モジュール名を指定します。※1	―

※1　"", # , $ は使用できません。《KV STUDIO》で設定したモジュール名を直接入力してください。
入力できる文字数は以下のとおりです。

- KV-8000(A)、KV-7000(CPU機能バージョン2.0以降)
  半角･全角問わず最大64文字
- KV-7000(CPU機能バージョン2.0未満)
  半角･全角問わず最大24文字
- KV-5500/5000/3000/1000、KV Nanoシリーズ
  半角英数字の場合は最大24文字、全角の場合は最大12文字

動作説明

MDSTRT　実行条件ONのとき S で指定したモジュールを起動します。
指定したモジュールは次のスキャンから動作を開始します。
すでに起動しているモジュールに対しては、何もしません。

MDSTOP　実行条件ONのとき S で指定したモジュールを停止します。
すでに停止しているモジュールに対しては、何もしません。
自分自身のモジュールを停止することも可能です。

CPUユニットにより、停止時のOFF処理の動作が異なります。詳細は「待機モジュールの動作」(4-101ページ)を参照してください。

ポイント

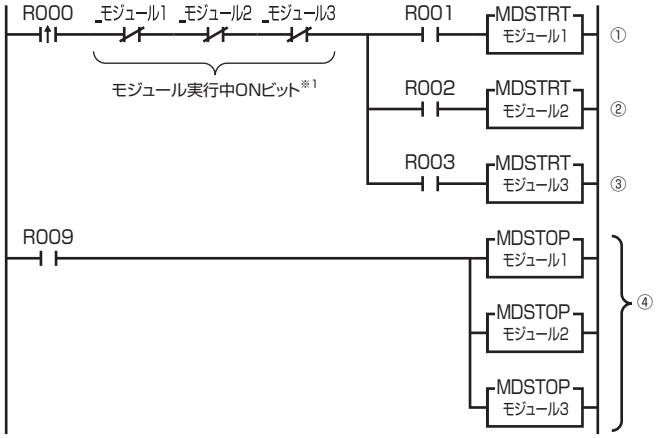
- **モジュール実行開始時における微分実行型命令の初期状態は、立ち上がり検出の場合がON、立ち下がり検出の場合がOFFになります。実行開始時に微分を直ちに検出することはありません。**
- **同じモジュールに対して同じスキャン内にMDSTRT命令、MDSTOP命令を複数回実行した場合は最後に実行したモジュール命令に従います。**
- **全てのモジュールを停止しないでください。全モジュールを停止した場合、全てのプログラムが実行されませんので、以下のいずれかの方法で復帰してください。(一旦プログラムを停止しなければ復帰できません。)**
  - **CPUユニットの電源を一度切り、再度、電源を入れてください。**
  - **CPUユニットの「RUN-PROG切換スイッチ」を一度、"PROGモード"に切り換えて再度、RUNモードにしてください。**
  - **《KV STUDIO》でプログラムのRUNを一度OFFし、再度ONしてください。**
- **初期化モジュールは実行開始･停止できません。**

演算フラグ

CR2009	変化なし
CR2010	変化なし
CR2011	変化なし
CR2012	変化なし

サンプルプログラム

入力リレーR000の立ち上がりを検出時に、モード切換（R001～R003）で選択されているモジュールを実行し、入力リレーR009がONすると、全てのモジュールの実行を停止します。いずれか1つのモジュールが実行中は、その他のモジュールを実行開始できません。



〈ニモニックリスト〉

```
LDP R000
ANB _ モジュール 1
ANB _ モジュール 2
ANB _ モジュール 3
MPS
AND R001
MDSTRT モジュール 1
MRD
AND R002
MDSTRT モジュール 2
MPP
AND R003
MDSTRT モジュール 3
LD R009
MDSTOP モジュール 1
MDSTOP モジュール 2
MDSTOP モジュール 3
```

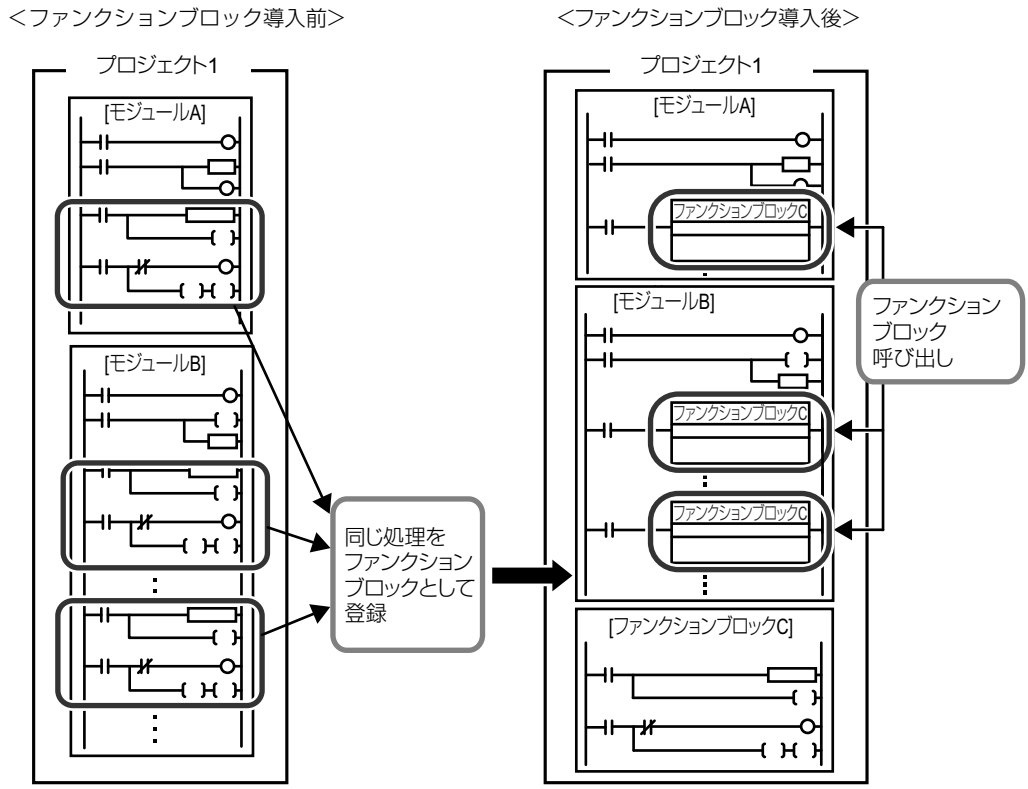
① R001がONのとき「モジュール1」を実行します。
② R002がONのとき「モジュール2」を実行します。
③ R003がONのとき「モジュール3」を実行します。
④ R009がONのとき、モジュール1～3は停止します。

※1　モジュール実行中ONビットの詳細は、「モジュールシステムデバイス」（4-113ページ）をを参照してください。

4-10 ファンクションブロック

概要

ファンクションブロックとは、プロジェクト内で繰り返し使用する回路ブロックを、一つの部品として任意の名称で登録したものです。ファンクションブロック化することにより、ラダー命令語と同じように使用できるため、プログラム入力工数が削減されます。回路変更時には、ファンクションブロックとして登録した1箇所のデータを修正するだけで、プロジェクト内で呼び出す全てのファンクションブロックの処理内容が変更されます。このことにより、プログラムの変更工数が削減され、修正ミスも防げます。



■ ファンクションブロックの管理

《KV STUDIO》では、[ワークスペース]を使用してプロジェクトごとに管理します。

■ ファンクションブロックの特長

● オリジナル命令語として使用できる

プログラム内に何度も出てくる同じラダー回路をファンクションブロックとして登録すれば、以後の入力工数や入力ミスを削減できます。
結果として、見た目のラダー行数が減り、プログラム構造を把握しやすくなります。
登録するファンクションブロック名は自由に付けることができます。

● 引数が使用できる

引数を使用すれば、ファンクションブロック内にある命令語のオペランド(デバイスや値)をファンクションブロック呼び出し時に指定できるため、汎用性の高いファンクションブロックを作成できます。
また、引数は任意の名称を設定できるため、引数の内容がわかりやすくなり、プログラムの可読性を向上することができます。
「引数設定」(4-131ページ)

● パスワードが設定できる

ファンクションブロックの内容を容易に変更できなくするため、パスワードを設定できます。
パスワードを設定した場合、パスワードを知らない人はファンクションブロックのプログラムを表示、編集できません。
[プログラムプロパティ]ダイアログでパスワードを設定します。

● 2種類のファンクションブロックが作成できる

ファンクションブロックには、『ファンクションブロック』と『ファンクション』の2種類があります。
『ファンクションブロック』と『ファンクション』の違いは以下のとおりです。

	『ファンクションブロック』	『ファンクション』
実行条件 「ファンクションブロックの種類」(4-120ページ)	実行条件がONすると、動作を実行します。動作が停止する条件は、ファンクションブロックの起動に使用した命令語によって異なります。 「ファンクションブロック」(4-120ページ)	実行条件がONしている間だけ実行します。実行条件がOFFすると動作を終了します。
プログラムの実体	インスタンスの個数分だけラダープログラムの実体を作成し、プログラム容量が増加します。 「ファンクションブロックインスタンス」(4-121ページ)	同一の領域を使用するため、プログラム容量は増えません。
ローカル変数/デバイスの使用時 「4-12 ローカルデバイス」(4-200ページ)	インスタンスの個数分だけ別々の領域が占有されます。そのため、デバイスの使用状況を意識せずに使用できます。 「ファンクションブロック」(4-120ページ)	共通の領域が使用されます。同一スキャンで複数回実行する場合は、ファンクション内で毎回初期化を実行してください。
ネスト構造 「ファンクションブロックのネスト」(4-140ページ)	最大8重まで対応できます。	最大8重まで対応できます。 (ただし、ファンクションからはファンクションブロックを呼び出しできません)

参考　ファンクションブロックとマクロの違いについては、「ファンクションブロックとマクロの違い」(4-175ページ)を参照してください。

使用手順

ファンクションブロックの使用手順は次のとおりです。

① ファンクションブロックの新規作成

【プログラム(M)】→【新規作成(M)】

プログラムの種類で『ファンクションブロック』/『ファンクション』を選択します。

「ファンクションブロックの種類」(4-120ページ)

▼

② 引数の設定

作成したファンクションブロックに引数を設定します。

「引数設定」(4-131ページ)

▼

③ ラダープログラムの作成

設定した引数を使用してファンクションブロックにラダープログラムを記述します。

▼

④ ファンクションブロックの配置

作成したファンクションブロックを配置します。

「ファンクションブロック命令」(4-148ページ)

ファンクションブロックの種類

ファンクションブロックには、『ファンクションブロック』と『ファンクション』があります。

ファンクションブロック

● 概要

ファンクションブロックは、実行条件がONの場合、ファンクションブロック内のプログラムを実行します。ファンクションブロックを実行するための命令語には3種類あり、それぞれ以下の特徴があります。

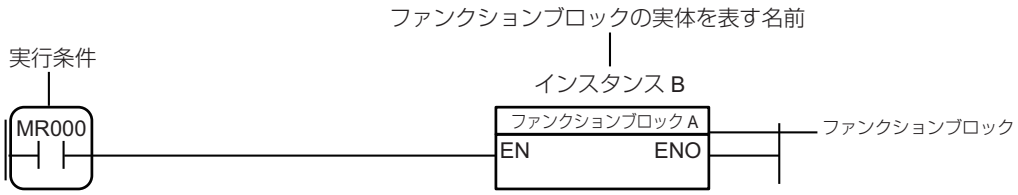
項目	FBCALL ファンクションブロック コール	FBSTRT ファンクションブロック スタート	FB ファンクションブロック 呼び出し
起動方法	ファンクションブロックの実行条件のON		
停止方法	• ファンクションブロックを実行したプログラム(モジュールやファンクションブロックなど)を停止する • MC～MCR命令間で実行した場合、MC命令の実行条件をOFFする		
	• FBCALL 命令の実行条件をOFFする	• ファンクションブロック内でFEND命令を実行する	
インスタンス名で指定	○	○	−

ポイント

- **モジュールのEND～ENDH間に記載したプログラムからファンクションブロックを起動した場合、モジュールを停止してもファンクションブロックは停止しません。**
- **MC～MCR命令間で実行した場合、MC命令の実行条件がOFFされると、引数タイプOUTの引数には実行結果が反映されません。**
- **FBCALL/FBSTRT命令はCPU機能バージョン2.0以降のKV-8000でのみ使用できる命令です。CPU機能バージョン2.0以降のKV-8000でプログラムを作成する場合は、FBCALL/FBSTRT命令を使用してファンクションブロックを実行することを推奨します。**
- **ST言語でインスタンス名を指定して実行した場合、実行条件をOFFすることで停止できます。**

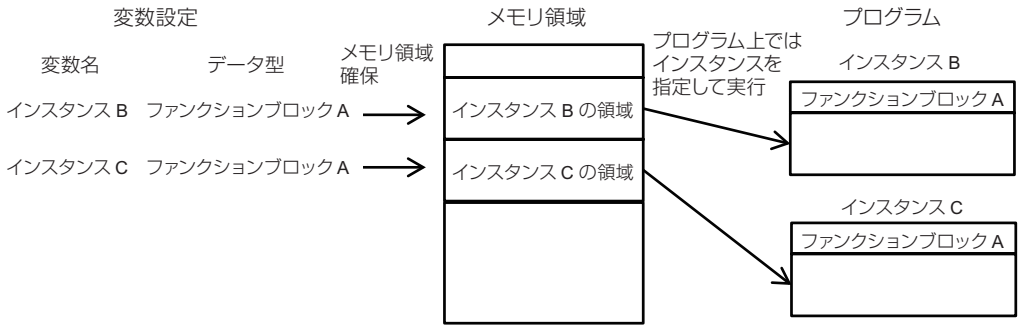
● 使用方法

ラダープログラムに配置して使用します。



● ファンクションブロックインスタンス

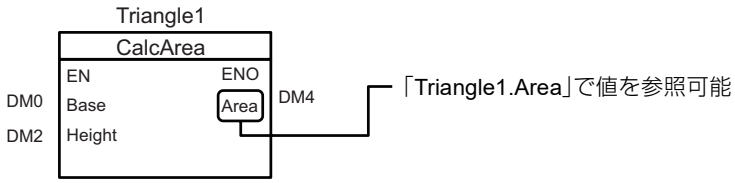
ファンクションブロックを配置すると、ファンクションブロック型の変数が登録されます。ファンクションブロック型の変数を登録すると、実行するための実体となるインスタンスが生成されます。同じファンクションブロックに対して異なる名前で変数を登録した場合は、それぞれが異なるインスタンスを生成し、異なるメモリ領域を占有します。インスタンスを指定してファンクションブロックを実行する場合は、FBCALL/FBSTRT命令を使用します。



ポイント

- **インスタンスを指定できるのは、FBCALL/FBSTRT命令でファンクションブロックを実行する場合のみです。FB命令の場合は、インスタンスは指定できず、必ず呼び出した回数だけの実体が生成されます。**
- **同一のインスタンスを同じプログラムで複数回指定することも可能です。その場合、生成する実体は1つです。**
- **FBCALL/FBSTRT命令を使用する場合は、Ver.11.0以降のKV STUDIOを使用して、KV-8000のCPU機能バージョンを2.0以降にアップデートしてください。**

ファンクションブロックのインスタンスを生成すると、指定したインスタンスのファンクションブロック引数をファンクションブロック内部のプログラム以外からも参照できるようになります。参照する場合は「インスタンス名.引数名」と記載します。
例えば以下のようにインスタンスを指定してファンクションブロックを呼び出した場合、「Triangle1.Area」と記載することで、ファンクションブロックの外部からインスタンスTriangle1のOUT引数「Area」の値を参照できます。



ただし、引数タイプの設定により、以下の制約があります。

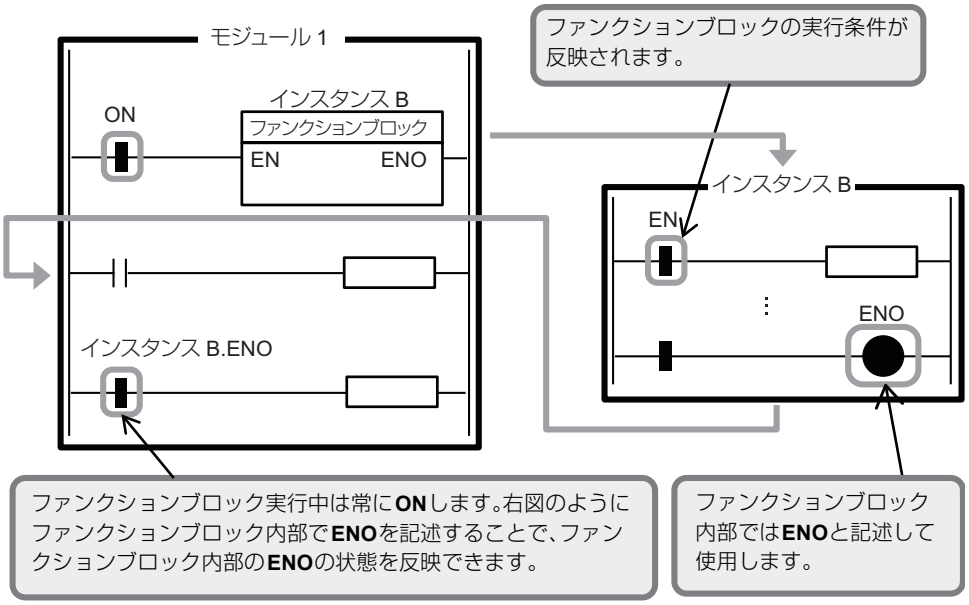
引数タイプ	内容
IN	「インスタンス名.引数名」で引数の値を参照、変更できます。
OUT	「インスタンス名.引数名」で引数の値を参照できます。値の変更はできません。
IN-OUT	「インスタンス名.引数名」での引数の値の参照および変更はできません。
ユニット	「インスタンス名.引数名」での引数の値の参照および変更はできません。

● ファンクションブロックのイネーブル入出力

イネーブル入力、イネーブル出力とはファンクションブロックシンボルの「EN」と「ENO」のことをいいます。

引数タイプ	内容
EN	ファンクションブロックの実行条件のON/OFF状態をファンクションブロック内部へ伝達します。ファンクションブロック内部のプログラムでは「EN」と記述することで使用できます。ただし、ファンクションブロック内部のプログラム以外からは「インスタンス名.EN」と記述して値を参照および変更することはできません。
ENO	ファンクションブロック実行中は常にONし、ファンクションブロックの実行を停止するとOFFします。ただし、ファンクションブロック内部のプログラムで値を書き込んだ場合は、書き込んだ値が反映されます。ファンクションブロック内部のプログラム以外からも「インスタンス名.ENO」と記述することで、値を参照できます。

ファンクションブロックを実行した際の、ENとENOの動作イメージを示します。



ポイント

- **「インスタンス名.ENO」でENOの値を参照する場合、FBCALL/FBSTRT命令でファンクションブロックを実行する必要があります。FB命令は、EN/ENOに対応していません。**
- **FBCALL/FBSTRT命令を使用する場合は、Ver.11.0以降のKV STUDIOを使用して、KV-8000のCPU機能バージョンを2.0以上にアップデートしてください。**
- **EN/ENOは引数として設定する必要はなく、引数の個数としてもカウントされません。**

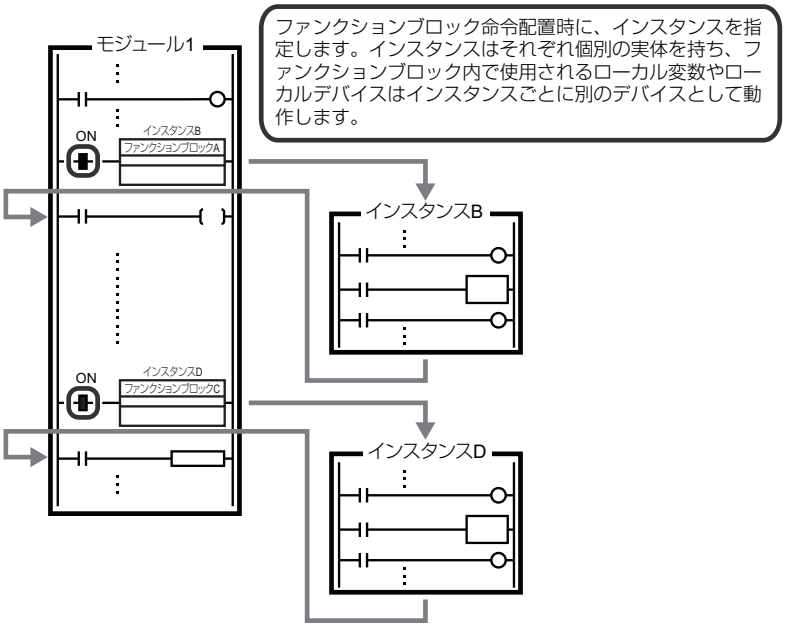
● 動作説明

ファンクションブロックの動作説明を下図に示します。

ここでは、ファンクションブロックAのインスタンスBと、ファンクションブロックCのインスタンスDを指定して実行する例を示します。

インスタンスの詳細は「ファンクションブロックインスタンス」(4-121ページ)を参照してください。

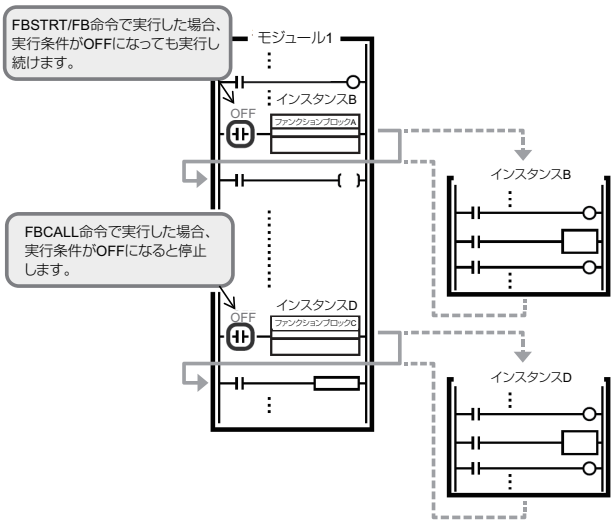
1 **実行条件がONの場合、ファンクションブロック A を起動します。**



ポイント

- **FBCALL/FBSTRT命令で起動したファンクションブロックは、指定したインスタンス名で区別され別々の実体として扱われます。**
- **FB命令で起動したファンクションブロックは、ファンクションブロックA#2_1のようにファンクションブロック名の後ろに管理番号を付けて区別します。この管理番号はKV STUDIOがモニタ/シミュレータ時のみ表示されます。**

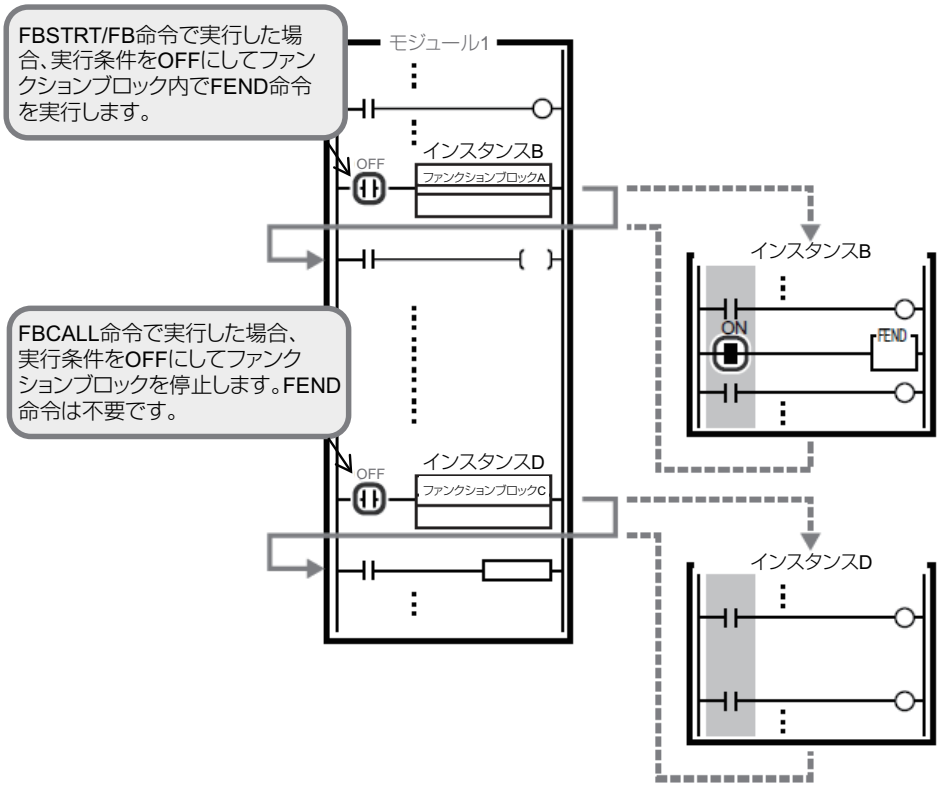
2 **ファンクションブロックを起動した命令によって、実行条件と動作状態は以下のとおりになります。**



● 停止条件

ファンクションブロックは以下の場合に停止します。

- 読み出し元のプログラムが停止した場合
- ファンクションブロック内でFEND命令を実行した場合（FBSTRT/FB命令の場合）
- ファンクションブロックの実行条件をOFFした場合（FBCALL命令やST言語からインスタンス名で実行した場合）



ポイント

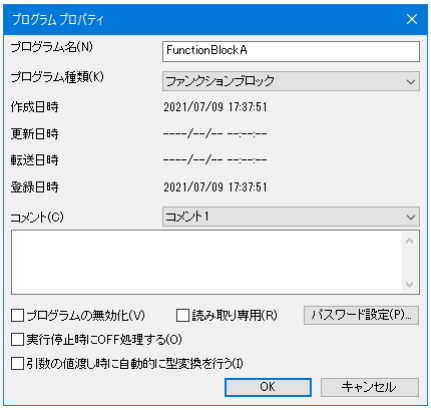
・FB/FBSTRT命令でファンクションブロックを実行した場合は、ファンクションブロックの実行条件をOFFにした状態でFEND命令を実行してください。ファンクションブロックの実行条件がONの状態でFEND命令を実行しても、ファンクションブロックは停止しません。
・FBCALL命令やST言語からインスタンス名でファンクションブロックを実行した場合は、ファンクションブロックの実行条件をOFFにしてファンクションブロックを停止してください。
・FBインスタンスを配列型で使用する場合、実行条件のON/OFFで実行/停止できるのは添え字で指定したインスタンスのみです。
添え字に定数以外を指定している場合、実行/停止前に添え字を変更すると、変更前の添え字で指定したインスタンスは実行/停止しないため、注意してください。

● 停止時の動作

ファンクションブロックの実行停止時の動作は、各プログラムの[プログラムプロパティ]→「実行停止時にOFF処理する」の設定により異なります。

ポイント

KV-8000Aの場合はファンクションブロックのプログラムプロパティに設定は表示されず、常に「実行停止時にOFF処理する」にチェックしていない場合と同じ動作になります。



「実行停止時にOFF処理する」にチェックしていない場合

ファンクションブロックは、以下のタイミングで停止します。

- ファンクションブロック内部でFEND命令が実行された次のスキャン(FBSTRT/FB命令の場合)
- ファンクションブロックの実行条件がOFFになったとき(FBCALL命令の場合)

ポイント

ファンクションブロックをEND～ENDH内に記述している場合は対象のモジュールを停止してもファンクションブロックは実行停止されません。
「ファンクションブロック命令」(4-148ページ)

「実行停止時にOFF処理する」にチェックしている場合

ファンクションブロックは、以下のタイミングで1スキャンのみ、ファンクションブロック内の実行条件をすべてOFFとして実行されます。※

- ファンクションブロック内部でFEND命令が実行された次のスキャン(FBSTRT/FBの場合)
- ファンクションブロックの実行条件がOFFになったとき(FBCALLの場合)

その次のスキャンから停止となります。

※ ただし、以下の命令は例外となります。

命令語	状態
OUB命令	OFFします。
OFDL命令	リセットします。
SHOT命令	
C命令	状態保持します。
CJ/NCJ/GOTO命令	分岐しません。

【FBSTRT/FB命令で実行したファンクションブロックの動作イメージ】

	1スキャン	1スキャン	1スキャン	1スキャン	1スキャン	1スキャン
ファンクションブロックAの実行条件	OFF	ON	※1	※1	OFF※2	OFF※3
ファンクションブロックA内にあるFEND命令の実行条件	-	OFF	OFF	ON	※1	-
ファンクションブロックAの状態	停止	実行	実行	実行	OFF条件で実行	停止

※1: ON/OFFどちらの状態でも動作に影響ありません。
※2: 必ずOFFしてください。OFFするまでFEND命令を無視して実行し続けます。
※3: ON状態であれば、ファンクションブロックの実行を開始します。

【FBCALL命令で実行したファンクションブロックの動作イメージ】

	1スキャン	1スキャン	1スキャン	1スキャン	1スキャン
ファンクションブロックAの実行条件	OFF	ON	ON	OFF	OFF※1
ファンクションブロックAの状態	停止	実行	実行	OFF条件で実行	停止

※1: ON状態であれば、ファンクションブロックの実行を開始します。

参 考

ファンクションブロックの実行停止時のOFF処理は、MC命令の実行条件がOFF時と同じ処理を1スキャンのみおこないます。

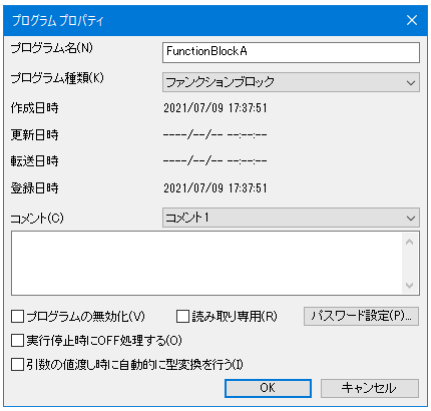
『KV-8000/7000/5000/3000/1000シリーズ・KV Nanoシリーズ 命令語リファレンスマニュアル』「MC/MCR」

● データ型の型変換

ファンクションブロック引数に対する値の格納については、各プログラムの[プログラムプロパティ]→「引数の値渡し時に自動的に型変換を行う」の設定により異なります。

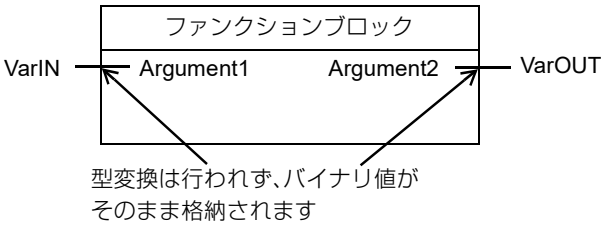
ポイント

- **Logix形式使用時は、「引数の値渡し時に自動的に型変換を行う」のチェックが初期値で有効になります。**
  それ以外の場合は、初期値で無効になります。
- **「引数の値渡し時に自動的に型変換を行う」の設定を行えるのはCPU機能バージョン2.3以降のKV-8000シリーズのみです。**



「引数の値渡し時に自動的に型変換を行う」のチェックが無効の場合

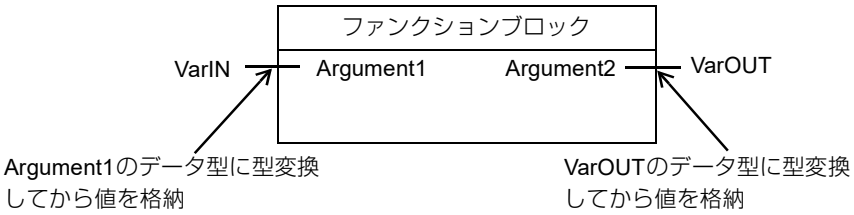
オペランドからIN引数に、またはOUT引数からオペランドに値を格納する際に、型変換を行わずにバイナリ値をそのまま格納します。



「引数の値渡し時に自動的に型変換を行う」のチェックが有効の場合

オペランドからIN引数に値を格納する際にはIN引数のデータ型に合わせて型変換を行ってから値を格納します。

OUT引数からオペランドに値を格納する際には、オペランドのデータ型に合わせて型変換を行ってから値を格納します。



ポイント

- **「引数の値渡し時に自動的に型変換を行う」のチェックが有効な場合は、オペランドには変数、もしくはFB引数(IN/OUT/IN-OUT)のみ指定できます。デバイス、およびFB引数(ユニット)は指定できません。**
- **「引数の値渡し時に自動的に型変換を行う」のチェックが有効な場合は、オペランドには間接参照、およびインデックス修飾している変数は指定できません。**

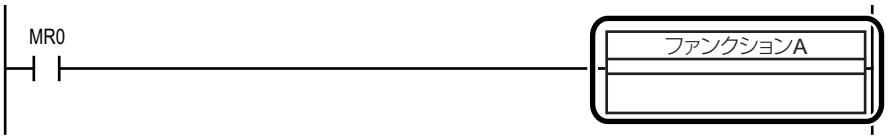
ファンクション

● 概要

ファンクションは、サブルーチンプログラムと同様に、実行条件がONの場合のみ、ファンクション内のプログラムを実行します。実行完了後、ファンクションの次のステップに進みます。実行条件がOFFの場合は、ファンクション内のプログラムを実行せずに次のステップに進みます。

● 使用方法

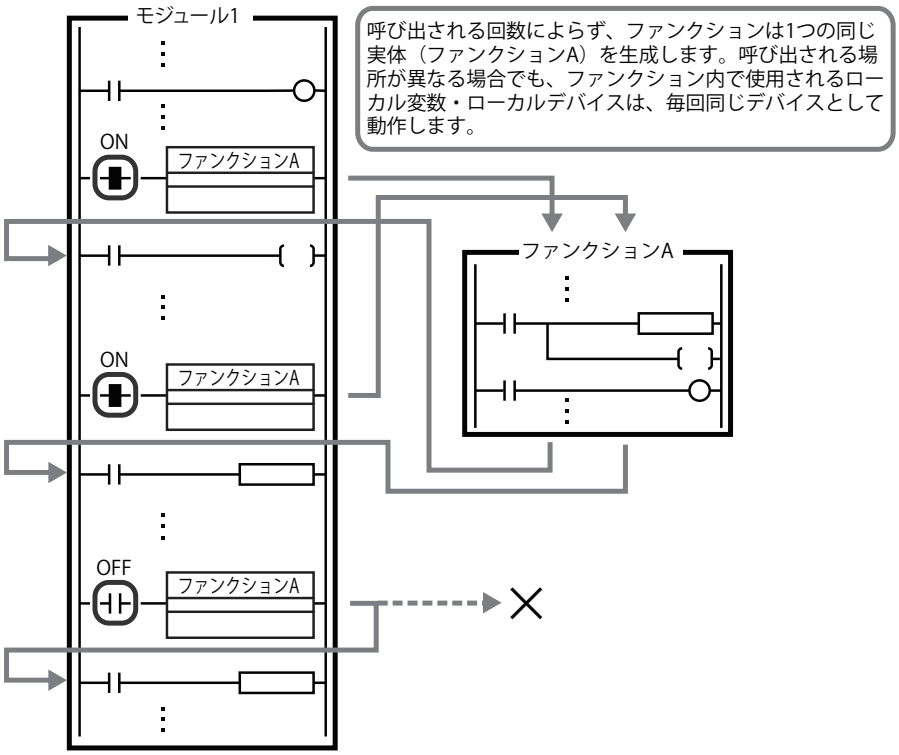
ラダープログラムに配置して使用します。



● 動作説明

ファンクションの動作説明を下図に示します。
モジュール1内でファンクションAを2回呼び出す例を示します。

実行条件がONの場合、モジュール1を抜けファンクションA内のラダープログラムを実行します。
実行完了後、モジュール1に戻り、ファンクションAの次のステップに進みます。
実行条件がOFFの場合、ファンクションAは実行せず、次のステップに進みます。



ポイント

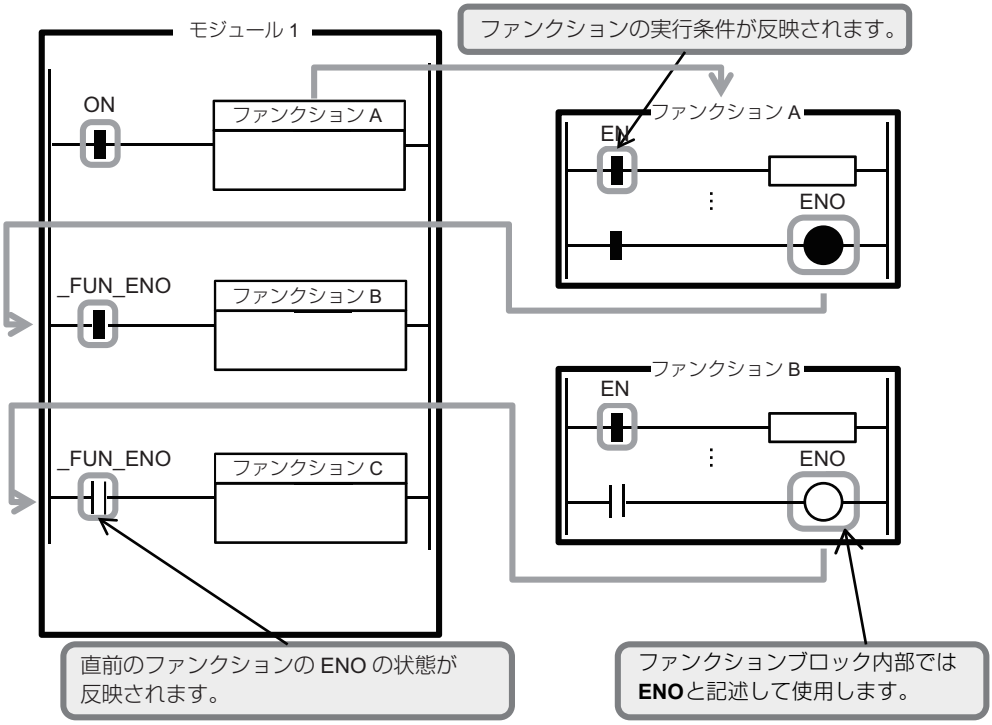
- ファンクション内で使用するローカル変数･ローカルデバイスは共通です。同一スキャン内に複数回実行する場合や、割り込みプログラムでファンクションを実行する場合は、注意してください。
- ファンクション内で使用されるローカルデバイスの値は、モニタ時には同一スキャン内で最終に実行された結果を表示します。
- ファンクションはインスタンスを生成しません。

● ファンクションのイネーブル入出力

ファンクション内部のプログラムでは、設定引数とは別に、EN、ENOという引数を使用できます。

引数タイプ	内容
EN	ファンクションの実行条件のON/OFF状態をファンクション内部へ伝達します。ファンクション内部のプログラムでは「EN」と記述することで使用できます。「EN」の状態は、ファンクション内部のプログラム以外からは直接参照できません。
ENO	ファンクション実行中は常にONし、ファンクションの実行を停止するとOFFします。ただし、ファンクション内部のプログラムで値を書き込んだ場合は、書き込んだ値が反映されます。 ファンクション内部のプログラム以外からは「_FUN_ENO」と記載することで、直前のファンクション内部で格納された「ENO」の値を参照できます。

ファンクションを実行したときのENとENO、_FUN_ENOの動作イメージを示します。下記例の場合、ファンクションAとファンクションBは実行されますが、ファンクションB内部のENOがOFFしているので、ファンクションCの実行条件である_FUN_ENOもOFFしたままとなり、ファンクションCは実行されません。このように、直前のファンクション内部の状態によって処理を分岐する場合に使用できます。



ポイント

- **「_FUN_ENO」で直前のファンクション内部のENOの値を参照する場合、Ver.11.0以降のKV STUDIOを使用してKV-8000のCPU機能バージョン2.0以降にアップデートしてください。**
- **EN/ENOは引数として設定する必要はなく、引数の個数としてもカウントされません。**

● ファンクションの戻り値

ファンクションの戻り値とは、ファンクション内部のプログラムで使用できる、ファンクション名と同じ名前を持つ引数です。戻り値の値はファンクション外部のプログラムから参照できます。スクリプトではFUN関数を使用することで、ファンクションの戻り値を直接扱えるため、ファンクション内部の演算結果を格納するための引数を設定する必要がありません。

以下は、Function1の戻り値をDM100に格納するためのST、KVスクリプトです。
ST記述　DM100 := Function1(DM0、DM1);
スクリプト記述　DM100 = FUN("Function1", DM0, DM1)

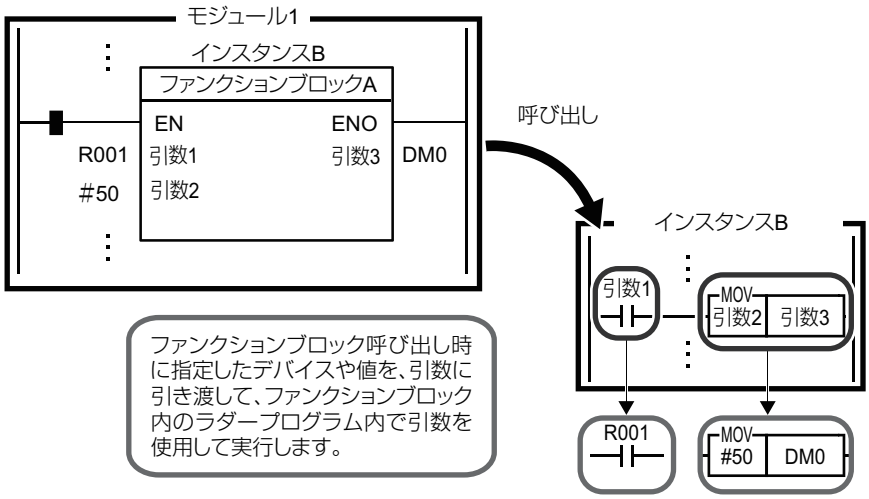
※　DM0、DM1はFunction1の引数です。

例えばFunction1の内部のプログラムが第1引数を底辺、第2引数を高さとする三角形の面積を求めるプログラムであれば、上記のように記述することでDM100に三角形の面積の演算結果が格納されます。

引数設定

■ 引数とは

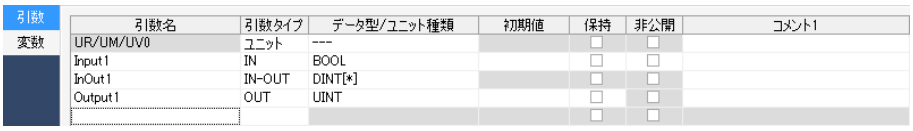
引数とは、ファンクションブロック実行時にファンクションブロックの実体に対して引き渡すデバイスや値です。ファンクションブロックの入出力、もしくはファンクションブロック呼び出し命令（FBCALL、FBSTRT、FB, FUN）のオペランドとして指定します。ファンクションブロックでは引数に対してデバイスや値が引き渡され、ファンクションブロック内のラダープログラムでは、引数名を使用してプログラムを作成します（下記の例では、引数1、引数2、引数3が引数名）。引数はファンクションブロックが実行中であれば毎スキャン更新されます。
引数を引き渡す際に使用するラベルは、そのファンクションブロック内でのみ使用できます。



■ 設定方法

引数は、ファンクションブロックのラダー編集ウィンドウ上部に表示される引数設定領域で設定します。
もしくは、ワークスペースの設定したいファンクションブロック上で、右クリックメニューから【引数設定(Z)】を選択して表示される[ファンクションブロック引数設定]ダイアログで設定します。

別手順 ワークスペース上でファンクションブロックを選択し、メニューから【プログラム(M)】→【引数設定(Z)】を選択



引数名	引数タイプ	データ型/ユニット種類	初期値	保持	非公開	コメント1
UR/UM/UV0	ユニット	---		☐	☐	
Input1	IN	BOOL		☐	☐	
InOut1	IN-OUT	DINT[*]		☐	☐	
Output1	OUT	UINT		☐	☐	
				☐	☐	

● 引数名

- 引数タイプ：**IN**、**OUT**、**IN-OUT**の場合
  引数の名称を最大128文字（半角全角区別なし）で設定します。
  設定した引数名は、ファンクションブロック内でローカル変数と同様に使用できます。
  「付-9 使用できない文字一覧」（付-40ページ）
- 引数タイプ：ユニットの場合
  UR*、UM*、UV*の引数が固定で割り当てられます。*は、引数設定の上から順に0からの数字です。

● 引数タイプ

引数には、IN、OUT、IN-OUT、ユニットの4つのタイプがあります。各タイプの仕様を下表に示します。引数は、合計で最大64個まで設定できます。また、各タイプには、IN、OUT、IN-OUTはそれぞれ32個まで、ユニットは8個まで設定できます。

引数名		引数タイプ		説明	表示位置 ※1	間接指定（＊）	インデックス修飾（：）
任意の引数名		IN ※1		ファンクションブロック実行開始時、指定したデバイスまたは定数の値を引数に格納します。	左	×	×
任意の引数名		OUT ※1		ファンクションブロック実行完了時、引数の値を指定したデバイスに格納します。	右	×	×
任意の引数名		IN-OUT		ファンクションブロック実行開始時、指定したデバイスへの参照を引数に格納します。 ファンクションブロック実行中に引数に対して行う処理は、都度指定したデバイスへ反映されます。	左右※1	×	○※3
UR	UR0～UR7	ユニット ※2	先頭R	ファンクションブロック実行開始時、指定したユニットに割り付けられた先頭リレー（R）への参照を引数に格納します。 ファンクションブロック実行中に引数に対して行う処理は、都度指定したデバイスへ反映されます。	左	×	○
UM	UM0～UM7		先頭DM	ファンクションブロック実行開始時、指定したユニットに割り付けられた先頭データメモリ（DM）への参照を引数に格納します。 ファンクションブロック実行中に引数に対して行う処理は、都度指定したデバイスへ反映されます。		×	○
UV	UV0～UV7		番号	ファンクションブロック実行開始時、指定したユニットの番号を引数に格納します。		×	×

※1　引数タイプにより、ファンクションブロックのシンボル上で表示される位置が異なります。
　　引数タイプIN-OUTの場合、左右の2箇所に同じ入出力デバイスが表示され、どちらからでも編集できます。
※2　ファンクションブロックの入出力には、対象となるユニット番号のみ指定します。
　　1つのファンクションブロックで引数として指定できるユニットは、最大8台です。
※3　データ型が配列・構造体の場合はインデックス修飾できません。

参考　可変長の文字列および配列を引数として使用する場合は、引数タイプをIN-OUTに設定してください。

● データ型/ユニット種類

引数のデータ型を設定します。設定できるデータ型は引数タイプにより異なります。

- 引数タイプ：**IN**、**OUT**、**IN-OUT**の場合
  引数のデータ型を設定します。設定できるデータ型は引数タイプにより異なります。

引数タイプ	BOOL (ビット)	UINT (1ワード符号なし整数)	INT (1ワード符号あり整数)	UDINT (2ワード符号なし整数)	DINT (2ワード符号あり整数)	REAL (単精度浮動小数点型)	LREAL (倍精度浮動小数点型)	STRING (文字列)	配列	構造体
IN	○	○	○	○	○	○	○	○※1	○※1	○
OUT	○	○	○	○	○	○	○	○※1	○※1	○
IN-OUT	○	○	○	○	○	○	○	○※1	○※1	○

※1　IN/OUT に設定した場合はあらかじめ最大文字数、要素数を設定します。IN-OUT に設定した場合は最大文字数、要素数を可変に設定できます。ただし、IN-OUTに設定したSTRING型配列は、要素数は可変に設定できますが、最大文字数は可変に設定できません。

文字数や配列の要素数が大きい引数については、IN/OUTに設定するとコピー処理に時間がかかるため、IN-OUTに設定して使用することを推奨します。

- 引数タイプ：ユニットの場合
  ユニットの種類を設定します。ユニットエディタで配置されているユニットから設定できます。

● 初期値

引数のタイプがIN、OUTの場合、初期値を設定できます。引数タイプがIN-OUTの場合、初期値の設定はできません。
初期値を設定した場合、引数の値を省略した場合に、初期値を使用してファンクションブロックを実行します。

ポイント

- **CPU機能バージョン2.0以降のKV-8000では以下の場合にも引数が初期化されます。**
  - **保持設定を行っている引数は、ファンクションブロック定義を変更してプログラムを転送したとき**
  - **保持設定を行っていない引数は、PROGRAMモード→RUNモードに切り換えたとき**
    ただし、FB命令で実行しているファンクションブロックの引数は、保持設定を行っている場合でもPROGRAMモード→RUNモード切り換え時に初期化されます。

  ファンクションブロックの起動、および停止時には初期化されません。
- **CPU機能バージョン1.0のKV-8000ではファンクションブロックの起動、および停止時にも引数が初期化されます。**

● 保持

引数タイプがIN、またはOUTの場合は保持設定を行えます。
「保持」にチェックを入れた場合、引数の値が電源断時も保持されるようになります。

ポイント

- **インスタンス名を指定しないFB命令を使用した場合は値が保持されません。保持が必要な場合は、FBCALL命令またはFBSTRT命令を使用してインスタンス名を指定して実行するようにしてください。**
- **引数の保持設定を行う場合はVer.11.0以降のKV STUDIOを使用してKV-8000のCPU機能バージョンを2.0以降にアップデートしてください。**

● 非公開

以下の条件を満たす引数は、非公開に設定できます。

- 引数タイプがIN
- 初期値が設定されている
- 非公開に設定しない引数は、すべて非公開に設定した引数よりも小さいNo.に並べられている
  （非公開に設定した引数は、No.がもっとも大きい引数に自動的に変更されます。）

非公開に設定した引数は、ファンクションブロックのシンボルに表示されません。代わりに[オペランド変更]ダイアログの詳細表示で表示･編集します。また、パスワードを設定して表示･編集を制限することもできます。

● コメント

引数の名称の下に表示する説明用のコメントを必要に応じて設定できます。

ポイント

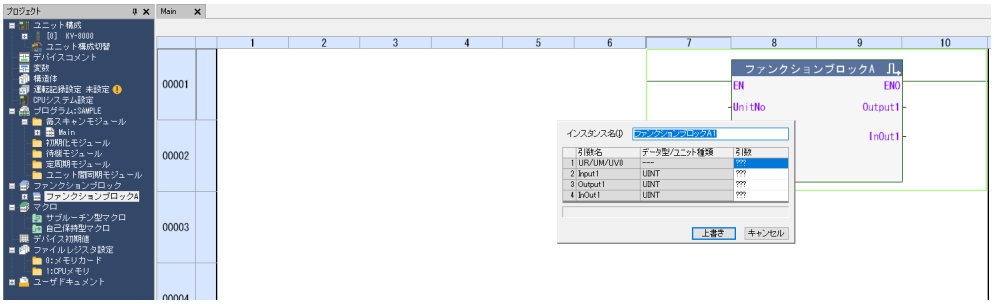
- **ファンクションブロック名称の下のコメント欄はプログラムプロパティで設定してください。**
- **CPU機能バージョン2.0未満の場合、引数一つにつき以下のサイズのローカルワークを使用します。**
  - **引数タイプがIN、OUTの場合**
    表示形式がLREAL（倍精度浮動小数点型）の場合は4ワード、それ以外は2ワード
  - **引数タイプがIN-OUTの場合は2ワード**
  - **引数タイプがユニットの場合は6ワード**
- **CPU機能バージョン2.0以降の場合、引数一つにつき以下のサイズのローカルワークを使用します。**
  - **引数タイプがIN、OUTの場合**
    データ型が構造体、配列、文字列の場合はデータサイズを偶数ワードに切り上げたサイズ、LREAL型は4ワード、それ以外のデータ型は2ワード
  - **引数タイプがIN-OUTの場合は4ワード**
  - **引数タイプがユニットの場合は6ワード**
- **同じ『ファンクション』をメインルーチンプログラムと割り込みプログラムで同時に実行した場合、割り込みプログラム実行時に引数の値は退避･復帰されます。**
- **退避･復帰する引数が使用するローカルワークのワード数の合計が6912ワードを超えると演算エラー[ラダースタックオーバーフロー]が発生します。**
- **ファンクションブロックの引数設定と、引数に指定した変数のデータ型が異なる場合、引数設定により以下の挙動を示します。**
  - **IN引数、OUT引数**
    構造体、配列（要素数指定）　：データ型･要素数が異なる場合は変換エラー
    文字列（最大文字数指定）　：データ型が異なる場合は変換エラー
    その他のデータ型　：データ型が異なる場合は変換時に警告
  - **IN-OUT引数**
    構造体、配列（要素数指定）　：データ型もしくは要素数が異なる場合は変換エラー
    配列（要素数可変）　：データ型が異なる場合は変換エラー
    文字列（最大文字数指定）　：データ型もしくは最大文字数が異なる場合は変換エラー
    その他のデータ型　：データ型が異なる場合は変換時に警告
  - **IN、OUTに設定した引数は、ワーク領域にコピーされます。そのため、引数のサイズが大きい場合は、コピーの時間が発生せずワーク領域の消費が少ないIN-OUT引数に設定することを推奨します。**

ファンクションブロックの配置方法

1 ワークスペースのファンクションブロックをドラッグして、ラダー編集エリアにドロップします。

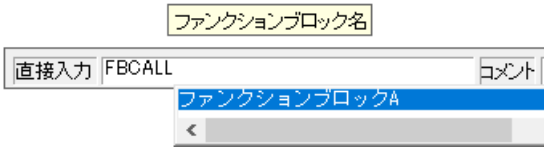
変数名を入力するダイアログが自動で表示されるので、変数名(インスタンス名)を入力して〔上書き〕をクリックします。

※ドラッグ&ドロップでファンクションブロックを配置した場合、FBCALL命令(ファンクションブロックの実行条件のOFFでファンクションブロックが停止する)が使用されます。

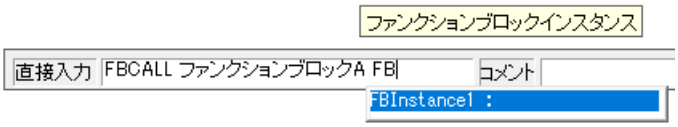


別手順　キーボードで「FBSTRT」もしくは「FBCALL」と入力します。

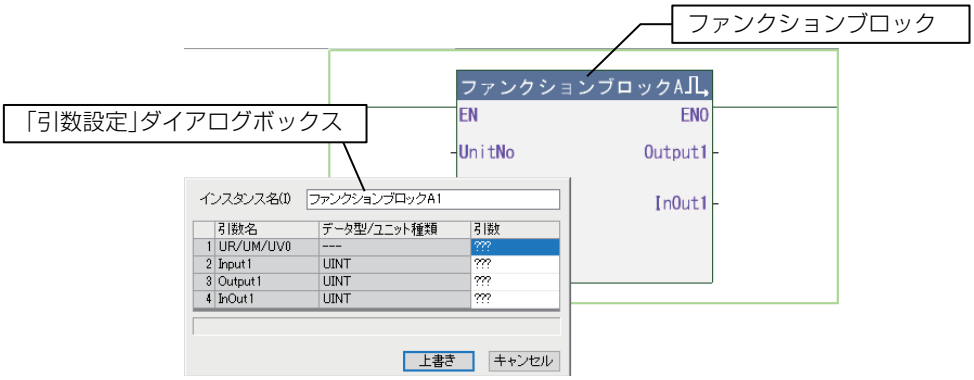
スペースキーを押してファンクションブロック名を候補から選択するか、直接入力します。



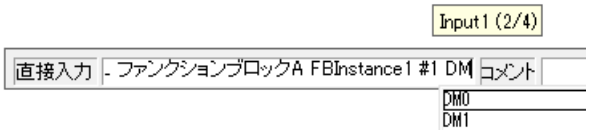
もう一度スペースキーを押して、インスタンス名を入力します。インスタンスがあらかじめ生成されている場合は候補として表示されます。



2 「オペランド変更」ダイアログボックスで割り当てるデバイスを指定し、〔上書き〕ボタンをクリックします。



別手順 直接入力ダイアログボックスで引数を直接入力ができます。



ポイント

- マクロ内には、ファンクションブロックを配置できません。
- ファンクションブロック内には、ファンクションブロックを配置できます。
  ただし、プログラムの種類(『ファンクションブロック』/『ファンクション』)の組み合わせによっては配置できない場合があります。
  「ファンクションブロックのネスト」(4-140ページ)

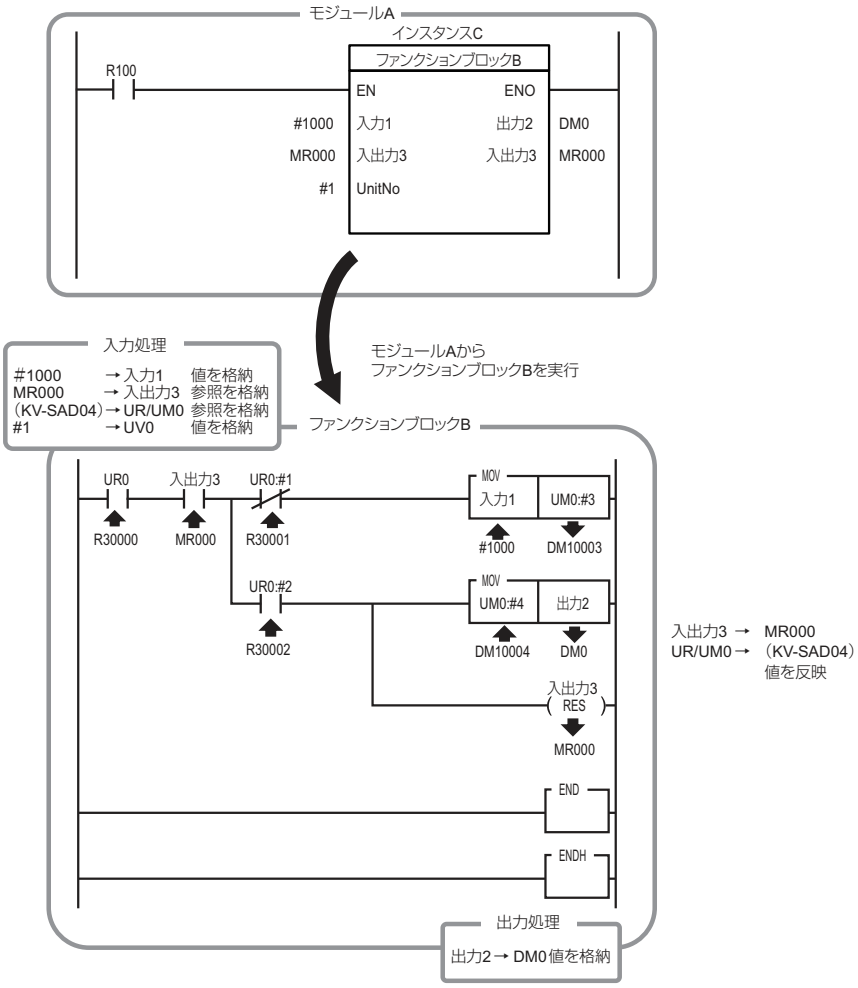
ファンクションブロックの使用例

ファンクションブロックの引数が、ラダープログラム内でどのように使用するかの使用例を下図に示します。
モジュールAからファンクションブロックBを実行した場合の例です。

【引数使用例】

No.	引数タイプ	引数名/デバイス
1	IN	入力1
2	OUT	出力2
3	IN-OUT	入出力3
4	ユニットデバイス	UR/UM/UV0

（ユニット番号1）…KV-SAD04　（R30000～R30115　DM10000～DM10059）



参考　ユニットデバイスUR0/UM0を使用するときは、間接参照を使用してデバイス番号を指定します。
例）ユニットの先頭リレー番号がR30000のときにR36604を指定する場合

R36604 = R30000 + 66ch (=66 × 16 = 1056) + 4
= R30000 + 1060
= UR0:#1060

引数の省略設定

ファンクションブロックでは、使用するパラメータの値や出力するデバイスを設定せずに、実行することができます。詳細な設定が用意できるようにファンクションブロックを作成し、簡易的に使用する場合は代表的な値を使用するなどの使い方ができます。

引数タイプがINもしくはOUTの引数は、入出力デバイスの指定を省略してファンクションブロックを使用できます。

ポイント
- **CPU機能バージョン1.0のKV-8000では、引数タイプがINの場合は初期値を設定している場合のみ省略できます。**
- **CPU機能バージョン1.0のKV-8000では、省略しないすべての引数が省略する引数よりも小さいNo.に並べられている必要があります(途中のNo.の引数は省略できません)。**

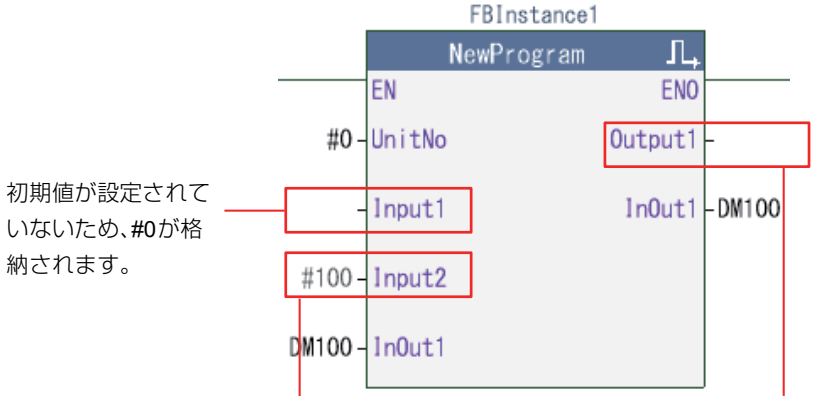
引数タイプがINの引数を省略した場合、ファンクションブロック実行開始時に初期値が設定されている場合は設定した初期値が、初期値が設定されていない場合は0が自動的に格納されます。
「初期値」(4-133ページ)
引数タイプがOUTの引数を省略した場合、ファンクションブロック実行完了時、その引数の値は出力されません。

以下のようなファンクションブロックの場合、Input1、Input2、Output1の引数は省略可能です。

IN引数

OUT引数

引数名	引数タイプ	データ型/ユニット種類	初期値	保持	非公開
UR/UM/UV0	ユニット	---		☐	☐
Input1	IN	UINT		☐	☐
Input2	IN	UINT	#100	☐	☐
Output1	OUT	UINT		☐	☐
InOut1	IN-OUT	UINT		☐	☐
				☐	☐



直接入力で引数を省略する場合は、「???」と記載します。以下はNo.3とNo.4の引数を省略する場合の例です。

直接入力 Program FBInstance1 #0 DM0 ??? ??? DM100 コメント

ポイント
- **「???」と入力して途中の引数を省略できるのは、CPU機能バージョン2.0以降のKV-8000のみです。**
- **「???」で引数を省略する場合、?はすべて半角文字で間にスペースなどを入れずに入力してください。**

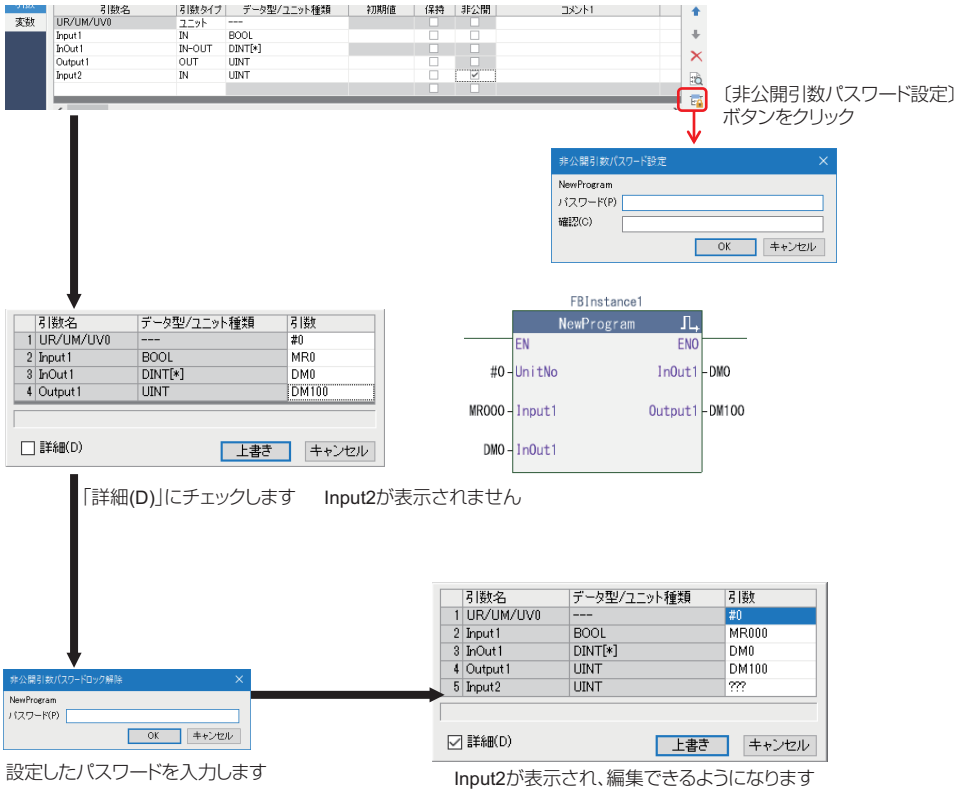
引数の非公開設定

ファンクションブロックでは、一部の引数を非公開設定できます。複雑な制御にも対応できるように複数の調整用のパラメータをあらかじめ用意したファンクションブロックを作成して、簡易的に使用したい担当者用に設定パラメータを隠した状態で公開できます。

「保持」(4-133ページ)

● 使用例

No.5のInput2を非公開に設定します。



参 考

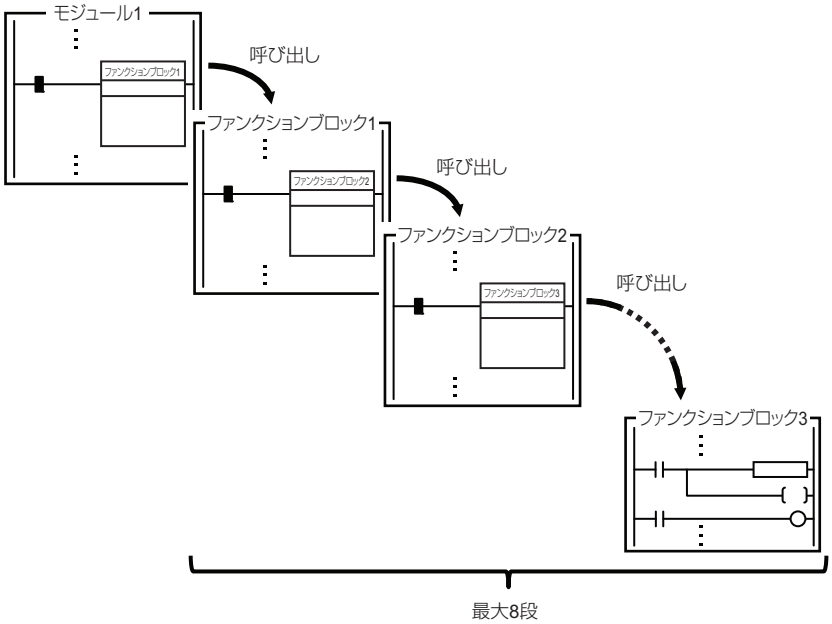
- 「詳細(D)」チェックボックスは、非公開に設定している引数がある場合のみ表示されます。
- 非公開に設定している引数の入力デバイスを編集しない(省略する)場合、初期値が使用されます。

ファンクションブロックのネスト

ファンクションブロックから、別のファンクションブロックを呼び出すことができます。(ファンクションブロックのネスト)
以下のように、ネストできるファンクションブロックは、プログラムの種類(『ファンクションブロック』/『ファンクション』)により異なります。

呼び出し元 プログラム種類	呼び出し可能プログラム種類	
	ファンクションブロック	ファンクション
ファンクションブロック	○	○
ファンクション	×	○

最大8段までネストできます。



ポイント

- 呼び出し元のファンクションブロックと同じファンクションブロックをネストすることはできません。
- プログラムの種類が『ファンクションブロック』の場合、インスタンスごとに別々の実体を生成します。
  ネストを使用する場合はプログラム・ローカルデバイスの容量に注意してください。
  プログラムの種類が『ファンクション』の場合、呼び出される回数によらず、生成する実体は1つになるため、ネストを使用してもプログラム・ローカルデバイスの容量は増加しません。
  「ファンクションブロックの種類」(4-120ページ)
- 8段以上ネストしたプログラムは《KV STUDIO》の変換時に変換エラーが発生します。
- 照合不一致状態でプログラムを転送し、実行時にネストが8段以上になった場合、CPUエラーE33「プログラムネスト制限オーバー」が発生し、運転を停止します。

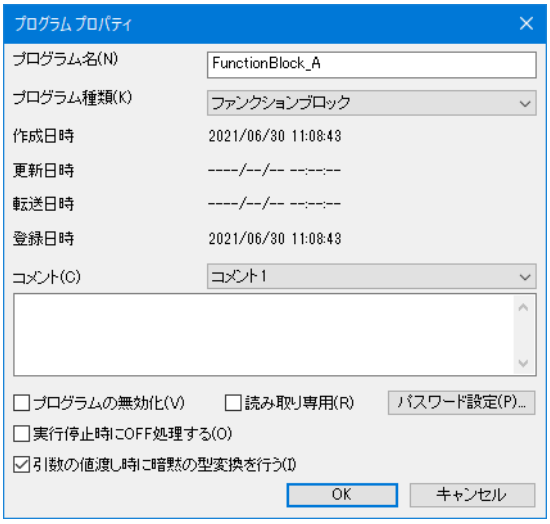
ファンクションブロックのパスワード設定

ファンクションブロックにパスワードを設定して、《KV STUDIO》でプログラムの表示に制限をかけることができます。
次の手順でパスワードを設定します。

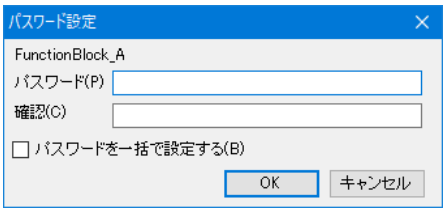
1 [ワークスペース]上の設定したいファンクションブロックを選択します。

2 右クリックして表示されるメニューから[プロパティ(**A**)]を選択します。
[プログラムプロパティ]ダイアログが表示されます。

別手順
- ワークスペースでファンクションブロックを選択し、メニューから【プログラム(M)】→【プロパティ(H)】を選択
- ワークスペースでファンクションブロックを選択し、右クリックメニューから【パスワード設定(B)】を選択



3 〔パスワード設定〕ボタンを押します。
[パスワード設定]ダイアログが表示されます。



4 パスワードとパスワードの確認を入力して、〔**OK**〕ボタンを押します。
パスワード設定後、パスワード設定したファンクションブロックを《KV STUDIO》で表示する時、[パスワード解除]ダイアログが表示されます。
パスワードを入力しないと、ファンクションブロックを《KV STUDIO》上で表示できません。

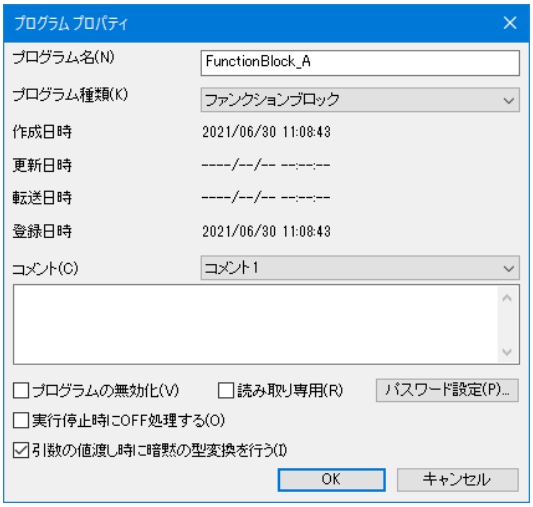
ファンクションブロック読み取り専用設定

ファンクションブロックを読み取り専用に設定して、《KV STUDIO》でプログラムの編集に制限をかけることができます。
次の手順で読み取り専用に設定します。

1 ［ワークスペース］上の設定したいファンクションブロックを選択します。

2 右クリックして表示されるメニューから［プロパティ（**A**）］を選択します。
［プログラムプロパティ］ダイアログが表示されます。

別手順
- ワークスペースでファンクションブロックを選択し、メニューから【プログラム（M）】→【プロパティ（H）】を選択
- ワークスペースでファンクションブロックを選択し、右クリックメニューから【パスワード設定（B）】を選択



3 「読み取り専用**(R)**」にチェックします。
クリックするごとに以下の状態に切り換わります。

状態	内容
☐	プログラムを編集できます。
☑	プログラムを編集できません。
■	プログラムを編集できます。プロジェクトを開き直すと 編集できない状態に戻ります。 一時的に編集する場合に使用します。

ファンクションブロックのシステムデバイス

ファンクションブロックシステムデバイスとは、ファンクションブロックの動作状況を取得するためのデバイスです。

ポイント **ファンクションブロックシステムデバイスには、デバイスコメントを付けられません。**

参考 ファンクションブロックシステムデバイスは、《KV STUDIO》によってCPUユニットの内部ワーク領域へ自動的に割り付けられます。

● ファンクションブロック実行開始時1スキャンのみONビット「@CR2008」

ファンクションブロックが実行を開始した1スキャン目のみONになります。
2スキャン目以降はOFFになります。
自ファンクションブロック内のみ参照可能です。ファンクションブロックを呼び出しているプログラム内では参照できません。
《KV STUDIO》上での表記は、「@CR2008」となります。

● ファンクションブロック実行開始時1スキャンのみOFFビット「@CR2007」

ファンクションブロックが実行を開始した1スキャン目のみOFFになります。
2スキャン目以降はONになります。
自ファンクションブロック内のみ参照可能です。ファンクションブロックを呼び出しているプログラム内では参照できません。
《KV STUDIO》上での表記は、「@CR2007」となります。

ポイント **ファンクションブロック内では、@CR2008などのシステムデバイスが使用できますが、ファンクション内では使用できません。**

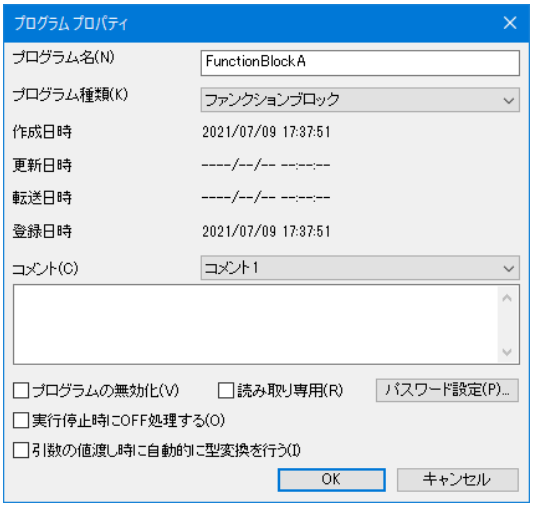
ファンクションブロックのプログラムプロパティ

ファンクションブロックのプログラムプロパティを使用してファンクションブロックを無効化したり、ファンクションブロックのコメント表示をしたりできます。
プログラムプロパティは次の手順で開きます。

1 [ワークスペース]上の設定したいファンクションブロックを選択します。

2 右クリックして表示されるメニューから[プロパティ(**A**)]を選択します。
[プログラムプロパティ]ダイアログが表示されます。

別手順
- ワークスペースでファンクションブロックを選択し、メニューから【プログラム(M)】→【プロパティ(H)】を選択
- ワークスペースでファンクションブロックを選択し、右クリックメニューから【パスワード設定(B)】を選択



● プログラムの無効化

チェックするとファンクションブロックを無効化できます。ワークスペースで右クリックメニューから「無効化　設定/解除」で無効化することもできます。

● 読み取り専用

チェックするとファンクションブロックを読み取り専用に設定できます。
「ファンクションブロック読み取り専用設定」(4-142ページ)

● 実行停止時にOFF処理する

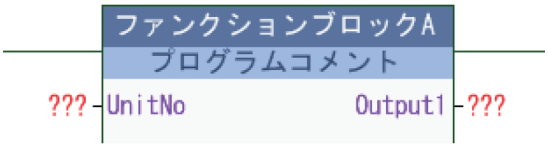
チェックすると実行停止時にOFF処理を実行します。
OFF処理については、「停止時の動作」(4-125ページ)を参照してください。

● 引数の値渡し時に自動的に型変換を行う

チェックするとオペランドから引数に値を渡すときに、オペランドと引数のデータ型が異なる場合にデータ型を変換します。
「データ型の型変換」(4-127ページ)を参照してください。

● コメント

ファンクションブロックのコメントを設定します。コメントを設定するとファンクションブロック名の下にコメントが表示されます。コメントには最大で256文字指定できます。



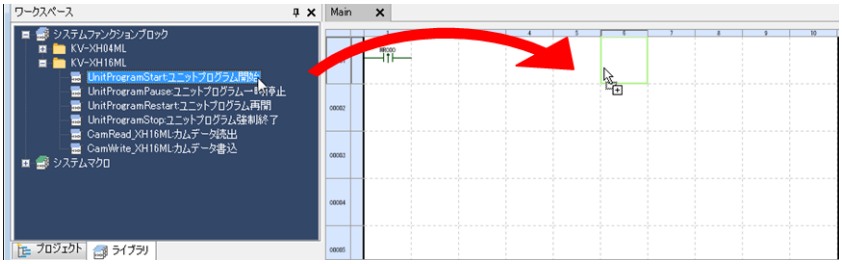
システムファンクションブロック

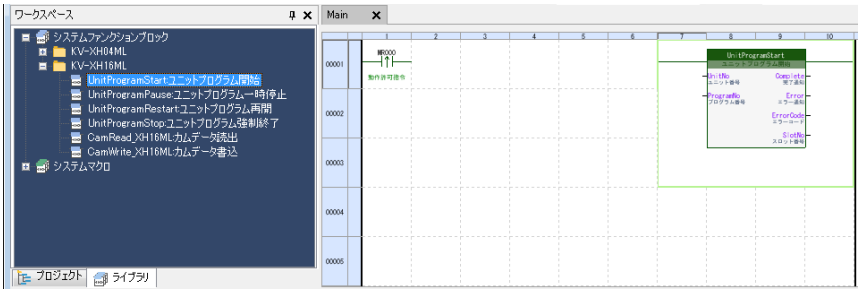
一部のユニットには、各ユニットの用途に応じた専用のシステムファンクションブロックが用意されています。

1 ［ワークスペース］のタブを**［ライブラリ］**に切り換えます。

2 システムファンクションブロックをドラッグして、ラダー編集エリアにドロップします。

選択したシステムファンクションブロックを呼び出すファンクション呼び出し命令が配置されます。同時に、選択したシステムファンクションブロックがプロジェクトに追加されます。





参考 システムファンクションブロックの機能の詳細は、各ユニットのユーザーズマニュアル参照してください。

ポイント

- **システムファンクションブロックは、通常のファンクションブロックと異なる色で表示されます。**
- **システムファンクションブロックは、初期状態では読み取り専用に設定されており、そのままでは編集できません。プログラムのプロパティから読み取り専用を解除すると、通常のファンクションブロックへと変化します。通常のファンクションブロックからシステムファンクションブロックへ戻すことはできません。**
  「ファンクションブロック読み取り専用設定」(4-142ページ)

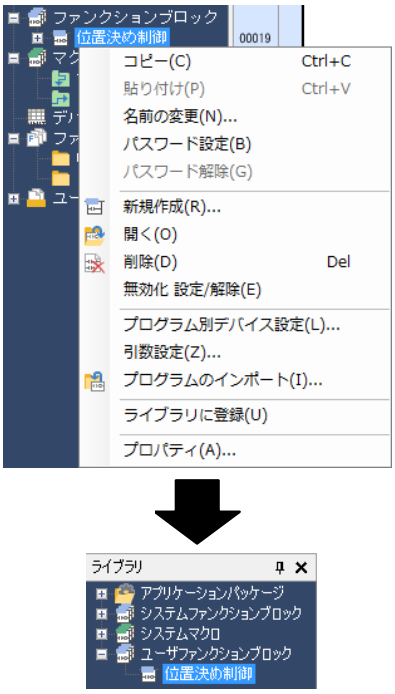
KV-8000(A)用システムファンクションブロックについては、「≪KV-8000≫用システムファンクションブロック」(4-154ページ)

ファンクションブロックのライブラリ登録

作成したファンクションブロックを資産として流用するためにライブラリに登録できます。ライブラリに登録したファンクションブロックは新規プロジェクトを作成した際にユーザファンクションブロックとしてあらかじめ登録されているため、プロジェクトを作成するたびにファンクションブロックをインポートする手間が省けます。

1 **[**ワークスペース**]**上でライブラリに登録したいファンクションブロックを選択します。

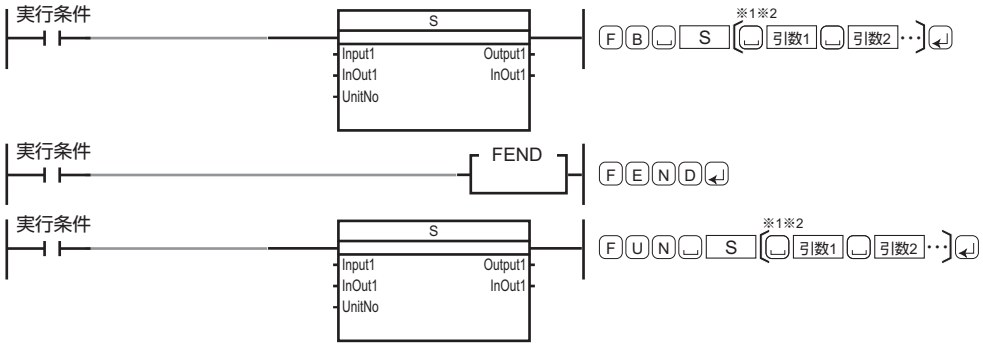
2 右クリックして表示されるメニューから【ライブラリに登録**(U)**】を選択します。



ポイント　ファンクションブロックをライブラリに登録できるのは、**KV STUDIO Ver.9.10**以降です。

FB		ファンクションブロック呼び出し	指定したファンクションブロックを呼び出します。
FEND	FEND	ファンクションブロックエンド	ファンクションブロックの実行を停止します。(ファンクションブロック内に記述)
FUN		ファンクション呼び出し	指定したファンクションを呼び出します。

ラダープログラム　　入力方法



オペランド	説明	占有サイズ
S	ファンクションブロック･ファンクション名を指定します。※3	―

※1　ファンクションブロック･ファンクション名称以外のオペランドの個数や、指定できるデータは、実行するファンクションブロック･ファンクションに設定されている引数設定によって以下のとおり異なります。

- IN　　:デバイス、変数、ラベル、定数を指定できます。
- OUT　　:デバイス、変数、ラベルを指定できます。定数は指定できません。
- IN-OUT　:デバイス、変数、ラベルを指定できます。定数は指定できません。
- ユニット　:デバイス、変数、ラベル、定数を指定できます。

※2　IN-OUT引数にT、C、TIMER型変数、COUNTER型変数を指定した場合、ファンクションブロック･ファンクション内で使用する命令語によって扱いが異なります。

※3　""、#、$は使用できません。KV STUDIOで設定してファンクションブロック･ファンクション名称を直接入力してください。半角英数字の場合は12文字、全角の場合は6文字まで指定できます。ただし、KV-8000/7000シリーズの場合は全角でも12文字まで指定できます。

ポイント　**FB命令、FEND命令、FUN命令は、KV-8000シリーズCPUユニットおよびCPU機能バージョン2.0以降のKV-7000シリーズCPUユニットで使用できます。**

「KV-8000(A)　CPU機能バージョンについて」(1ページ)

動作説明

FB 実行条件がONのとき、[S]で指定したファンクションブロックを起動します。
指定したファンクションブロックは、実行条件がOFFからONになったスキャンから実行を開始します。
一旦実行を開始すると、実行条件がOFFでもファンクションブロックを実行し続けます。

FEND 実行条件がONのとき、FEND命令が記述されているファンクションブロックを停止します。

FUN 実行条件がONのとき、[S]で指定したファンクションを実行します。

プログラムの設定により、停止時のOFF処理の動作が異なります。詳細は「停止時の動作」(4-125ページ)を参照ください。

ポイント

- **FB命令の呼び出し個数や順序を変更すると、RUN中書き込みできません。**
- **ファンクションブロックは以下の場合に停止します。**
  - **読み出し元のプログラムが停止した場合**
  - **ファンクションブロック内でFEND命令を実行した場合**
- **FB命令の実行条件がONのままの場合には、FEND命令は無視されます。**
- **FEND命令は、モジュール内やファンクション内に記述できません。**
- **FEND命令で他のファンクションブロックを停止することはできません**
- **ファンクションブロック内の微分実行型命令の前回状態は、ファンクションブロックの実行開始時に毎回初期化されます。(立ち上がり微分:ON、立ち下がり微分:OFF)**
- **ファンクション内では、微分実行型命令・タイマ命令は使用できません。**
- **以下の場合、各モジュール・ファンクションブロックが停止もしくは各命令間の実行条件がOFFしたとき、実行中のファンクションブロック・ファンクションは停止します。**
  - **FB/FUN命令をモジュール・ファンクションブロックに記述した場合**
  - **FB/FUN命令をマスターコントロール(MC～MCR命令間)に記述した場合**
  - **FUN命令をステージ(STG～JMP命令間)に記述した場合**
- **以下の場合、各命令間の実行条件がOFFしたとき、実行中のファンクションブロック・ファンクションは中断します。次に各命令間が実行されたときにファンクションブロックの実行を再開します。このときファンクションブロック内の微分実行型命令の前回状態は、初期化されません。(初期化モジュール内に記述した場合は、運転開始時1スキャンのみ実行します。)**
  - **FB/FUN命令をサブルーチン(SBN～RET命令間)に記述したとき**
  - **FB/FUN命令を条件分岐(CJ/NCJ～LABEL命令間)に記述したとき**
  - **FB/FUN命令をステップ(STP～STE命令間)に記述したとき**
  - **FB/FUN命令を割り込み(INT～RETI命令間)に記述したとき**
  - **FB命令をステージ(STG～JMP命令間)に記述したとき**
- **ファンクション内で、FB命令は使用できません。**
- **マクロ内でFB/FUN命令は使用できません。**
- **ファンクションブロック・ファンクション内に、そのファンクションブロック・ファンクション自身を呼び出すFB/FUN命令は使用できません。**
- **ファンクションブロック・ファンクション内でサブルーチンプログラムを使用できますが、割り込みプログラムは使用できません。**
- **プロジェクト内で同じファンクションブロック・ファンクションを複数回実行する場合、ファンクションブロック・ファンクション内でOUT/OUB命令にグローバルデバイス/グローバル変数を使用すると、2重コイルになります。**

演算フラグ

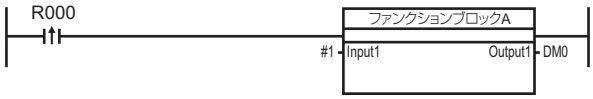
CR2009	実行するファンクションブロック･ファンクション内の処理結果に依存します。
CR2010	実行するファンクションブロック･ファンクション内の処理結果に依存します。
CR2011	実行するファンクションブロック･ファンクション内の処理結果に依存します。
CR2012	以下の場合はON、それ以外はOFF。 • 間接指定、インデックス修飾の範囲が不適切な場合 • 引数タイプ「ユニット」の引数に指定したユニットが接続されていない場合 • 引数タイプ「ユニット」の引数に指定したユニットと引数設定で指定したユニット種類が異なる場合 • オペランドの範囲外アクセスが発生した場合

※ 引数が不適切な場合、ファンクションブロックは実行されません。
CR2012がONした場合、CM5150～CM5176にエラーの詳細情報が格納されます。
「エラー履歴」(4-86ページ)

ポイント **FEND命令による演算フラグの変化はありません。**

サンプルプログラム

入力リレーR000の立ち上がりで、引数1(ユニット番号)を#1、引数2をDM0としてファンクションブロック「ファンクションブロックA」の実行を開始します。



〈ニモニックリスト〉

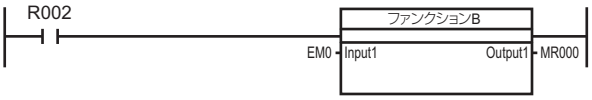
```
LDP R000
FB ファンクションブロック A #1 DM0
```

入力リレーR001がONのとき、ファンクションブロック「ファンクションブロックA」を停止します。

〈ニモニックリスト〉

```
LD R001
FEND
```

入力リレーR002がONのとき、引数1をEM0、引数2をMR000としてファンクション「ファンクションB」を実行します。

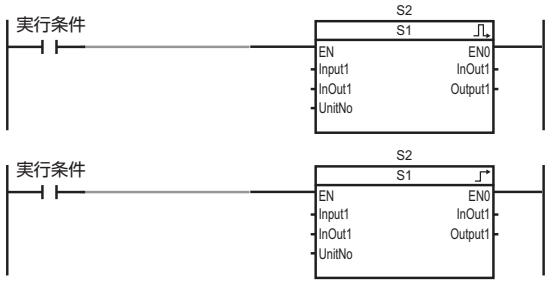


〈ニモニックリスト〉

```
LD R002
FUN ファンクション B EM0 MR000
```

FBCALL		ファンクションブロックコール	指定したファンクションブロックを呼び出します。
FBSTRT		ファンクションブロックスタート	指定したファンクションブロックを呼び出します。

ラダープログラム



入力方法

オペランド	説明	占有サイズ
S1	ファンクションブロック名を指定します。※2	文字数
S2	インスタンス名を指定しまます。※2	文字数

※1 第3オペランド以降の引数の個数(0～64個)や属性は、実行するファンクションブロックに設定されている引数設定によって異なります。

※2 定数のみ指定できます。ファンクションブロック名(半角全角問わず最大64文字)、インスタンス名(半角全角問わず最大128文字)を直接入力してください。

ポイント

FBCALL命令、FBSTRT命令はCPU機能バージョン2.0以降のKV-8000シリーズCPUユニットで使用できます。

「KV-8000(A)　CPU機能バージョンについて」(1ページ)

動作説明

FBCALL 実行条件が**ON**のとき、S1、S2で指定したファンクションブロックのインスタンスを第3オペランド以降で指定した引数で実行します。
指定したファンクションブロックは、OFF→ONになったスキャンから実行を開始します。
一旦実行を開始すると、実行条件がON中は常にファンクションブロックを実行し続けます。
実行条件がOFFすると、ファンクションブロックの実行を停止します。

FBSTRT 実行条件が**ON**のとき、S1、S2で指定したファンクションブロックのインスタンスを第**3**オペランド以降で指定した引数で実行します。
指定したファンクションブロックは、OFF→ONになったスキャンから実行を開始します。
一旦実行を開始すると、実行条件がOFFでもファンクションブロックを実行し続けます。
ファンクションブロックを停止する場合はファンクションブロック内でFEND命令を実行します。

プログラムの設定により、停止時のOFF処理の動作が異なります。
詳細は「停止時の動作」(4-125ページ)を参照してください。

ポイント

- FBCALL/FBSTRT命令ではファンクションを指定できません。ファンクションを実行する場合はFUN命令を使用してください。
- インスタンスを追加、削除、リネームした場合や、変数編集ウィンドウ上で順序を変更した場合は、RUN中書き込みできません。FBCALL/FBSTRT命令の削除や順序変更をした場合は、RUN中書き込みできます。
- ファンクションブロックは以下の場合に停止します。
  - 呼び出し元のプログラムが停止した場合
  - ファンクションブロック内でFEND命令を実行した場合（FBSTRT命令で呼び出した場合）
  - FBCALL命令の実行条件をOFFした場合（FBCALL命令で呼び出した場合）
  - FBCALL/FBSTRT命令の実行条件がONのままの場合には、FEND命令は無視されます。
- ファンクションブロック内の微分実行型命令の前回状態は、ファンクションブロックの実行開始時に毎回初期化されます。（立ち上がり微分：ON、立ち下がり微分：OFF）
- 以下の場合、各モジュール・ファンクションブロックが停止もしくは各命令間の実行条件がOFFしたとき、実行中のファンクションブロックは停止します。
  - FBCALL/FBSTRT命令をモジュール・ファンクションブロックに記述した場合
  - FBCALL/FBSTRT命令をマスターコントロール（MC～MCR命令間）に記述した場合
- 以下の場合、各命令間の実行条件がOFF したとき、実行中のファンクションブロックは中断します。次に各命令間が実行されたときにファンクションブロックの実行を再開します。このときファンクションブロック内の微分実行型命令の前回状態は、初期化されません。（初期化モジュール内に記述した場合は、 運転開始時1スキャンのみ実行します。）
  - FBCALL/FBSTRT命令をサブルーチン（SBN～RET命令間）に記述したとき
  - FBCALL/FBSTRT命令を条件分岐（CJ/NCJ～LABEL命令間）に記述したとき
  - FBCALL/FBSTRT命令をステップ（STP～STE命令間）に記述したとき
  - FBCALL/FBSTRT命令を割り込み（INT～RETI命令間）に記述したとき
  - FBCALL/FBSTRT命令をステージ（STG～JMP命令間）に記述したとき
- ファンクション内で、FBCALL/FBSTRT命令は使用できません。
- マクロ内でFBCALL/FBSTRT命令は使用できません。
- ファンクションブロック内に、そのファンクションブロック自身を呼び出すFBCALL/FBSTRT命令は使用できません。
- ファンクションブロック内でサブルーチンプログラムを使用できますが、割り込みプログラムは使用できません。
- プロジェクト内で同じファンクションブロックを複数回実行する場合、ファンクションブロック内でOUT/OUB命令にグローバルデバイス/グローバル変数を使用すると、2重コイルになります。

演算フラグ

CR2009	実行するファンクションブロック内の処理結果に依存します。
CR2010	実行するファンクションブロック内の処理結果に依存します。
CR2011	実行するファンクションブロック内の処理結果に依存します。
CR2012	以下の場合はON、それ以外はOFF。 • 間接指定、インデックス修飾の範囲が不適切な場合 • 引数タイプ「ユニット」の引数に指定したユニットが接続されていない場合 • 引数タイプ「ユニット」の引数に指定したユニットと引数設定で指定したユニット種類が異なる場合 • オペランドの範囲外アクセスが発生した場合

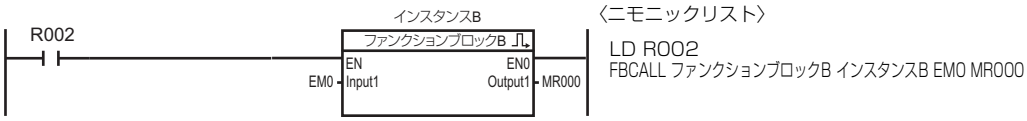
※ 引数が不適切な場合、ファンクションブロックは実行されません。
CR2012がONした場合、CM5150～CM5176にエラーの詳細情報が格納されます。
「エラー履歴」（4-86ページ）

ポイント

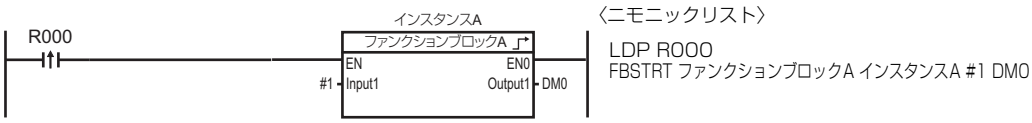
FEND命令による演算フラグの変化はありません。

サンプルプログラム

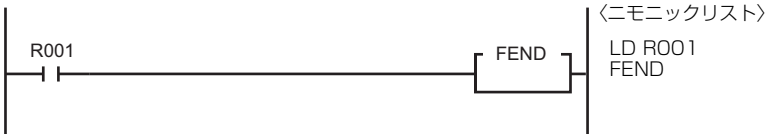
入力リレーR002がONのとき、引数1をEM0、引数2をMR000としてファンクションブロック「ファンクションブロックB」の「インスタンスB」という名前のインスタンスを実行します。入力リレーR002がOFFすると、実行を停止します。



入力リレーR000の立ち上がりで、引数1(ユニット番号)を#1、引数2をDM0としてファンクションブロック「ファンクションブロックA」の「インスタンスA」という名前のインスタンスを実行します。



入力リレーR001がONのとき、「ファンクションブロックA」の「インスタンスA」の実行を停止します。



≪KV-8000≫用システムファンクションブロック

ラダープログラム内で使用する≪KV-8000≫用システムファンクションブロックについて説明します。

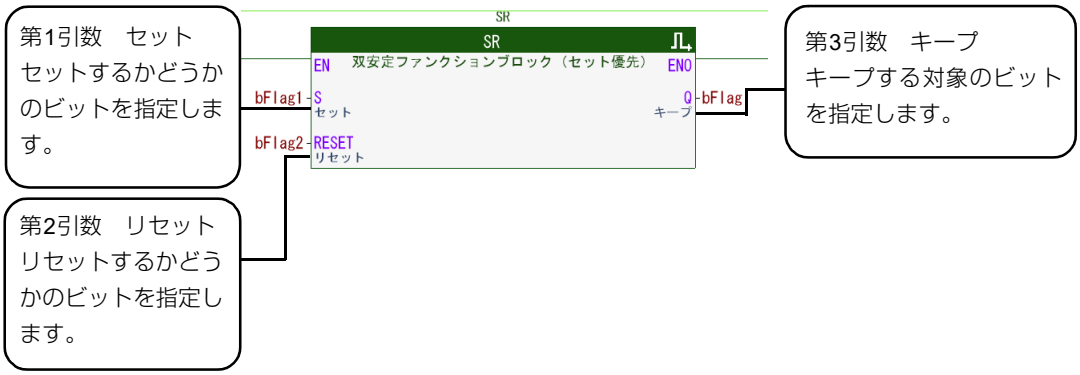
機能	ファンクションブロックプログラム名	動作概要	ページ
双安定ファンクションブロック(セット優先)	SR	セット/リセットの信号で出力信号を制御します。(セット優先)	4-155
双安定ファンクションブロック(リセット優先)	RS	セット/リセットの信号で出力信号を制御します。(リセット優先)	4-157
立ち上がりエッジ検出器	R_TRIG	立ち上がりエッジを検出します。	4-159
立ち下がりエッジ検出器	F_TRIG	立ち下がりエッジを検出します。	4-160
アップカウンタ	CTU	16ビット加算カウンタです。	4-161
ダウンカウンタ	CTD	16ビット減算カウンタです。	4-163
アップダウンカウンタ	CTUD	16ビットアップダウンカウンタです。	4-165
100ms単位オン･ディレイ	TON_100MS	100ms単位で指定するオンディレイタイマです。	4-167
10ms単位オン･ディレイ	TON_10MS	10ms単位で指定するオンディレイタイマです。	4-169
1ms単位オン･ディレイ	TON_1MS	1ms単位で指定するオンディレイタイマです。	4-171
10μs単位オン･ディレイ	TON_10US	10μs単位で指定するオンディレイタイマです。	4-173

SR

双安定ファンクションブロック（セット優先）

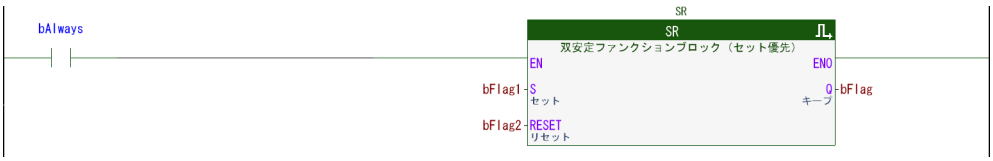
セット/リセットの信号の組み合わせで状態をビットの状態を制御します。（セット優先）

引数



サンプルプログラム

bFlag1とbFlag2の状態の組み合わせにより、bFlagの状態を制御します。

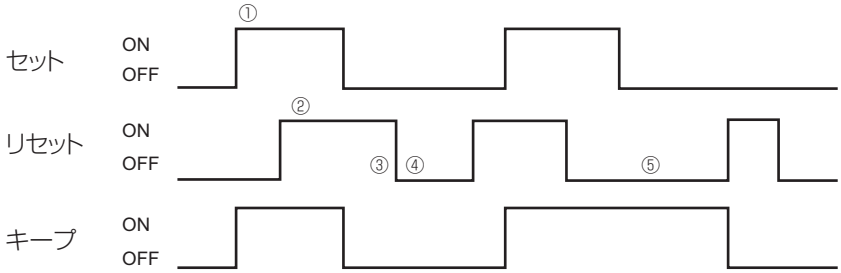


動作説明

「セット」で指定したビットと「リセット」で指定したビットの状態の組み合わせで「キープ」の状態を制御します。

組み合わせ表

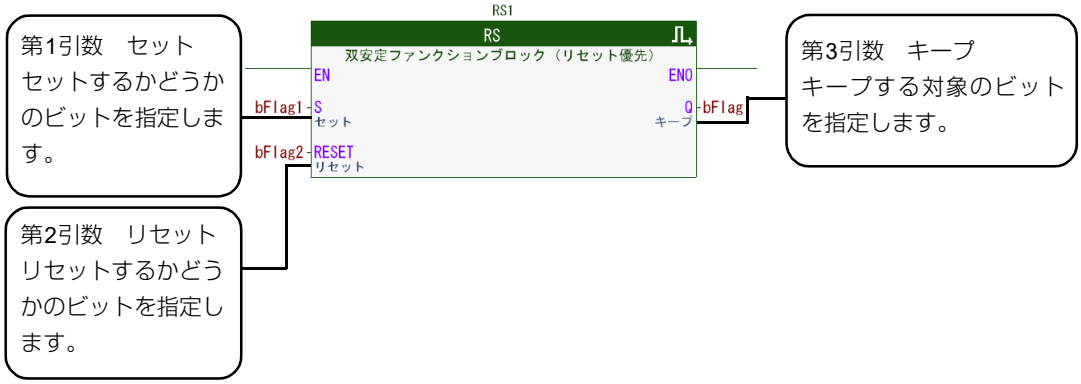
		RESET	
		ON	OFF
S	ON	ON	ON
	OFF	OFF	（直前の状態を保持）



① セットがON時、リセットはOFFなので、キープがONします。
② リセットがONしても、セットがONなので、キープはONします。
③ セットがOFFし、リセットがONなので、キープはOFFします。
④ キープがOFFの状態で、セット･リセットともにOFFしたので、キープはOFFのままです。
⑤ キープがONの状態で、セット･リセットともにOFFしたので、キープはONのままです。

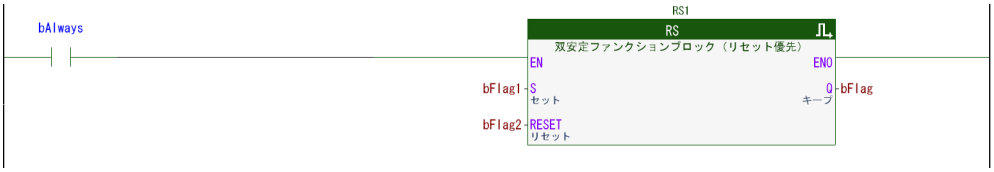
RS	双安定ファンクションブロック（リセット優先）	セット/リセットの信号の組み合わせで状態をビットの状態を制御します。(リセット優先)

引数



サンプルプログラム

bFlag1とbFlag2の状態の組み合わせにより、bFlagの状態を制御します。

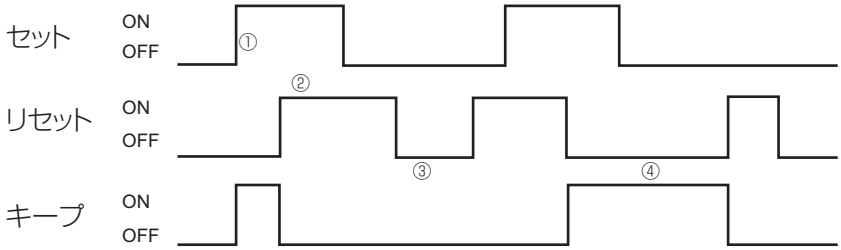


動作説明

「セット」で指定したビットと「リセット」で指定したビットの状態の組み合わせて「キープ」の状態を制御します。

組み合わせ表

		RESET	
		ON	OFF
S	ON	OFF	ON
	OFF	OFF	（直前の状態を保持）



① セットがON時、リセットはOFFなので、キープがONします。

② セットがON時に、リセットもONしたので、キープがOFFします。

③ キープがOFFの状態で、セット･リセットともにOFFしたので、キープはOFFのままです。

④ キープがONの状態で、セット･リセットともにOFFしたので、キープはONのままです。

R_TRIG　立ち上がりエッジ検出器

立ち上がり時に1スキャンだけONにします。

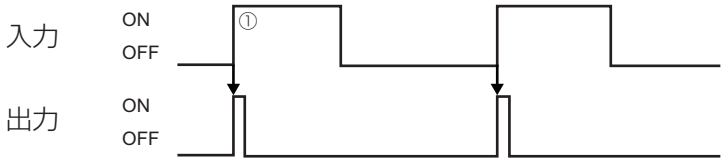
引数

サンプルプログラム

bRdyの立ち上がりでbStartを1スキャンだけONします。

動作説明

「入力」で指定したビットの立ち上がりで1スキャンだけ「出力」がONします。



①　入力の立ち上がり時に出力が1スキャンだけONします。

F_TRIG　立ち下がりエッジ検出器

立ち下がり時に1スキャンだけONにします。

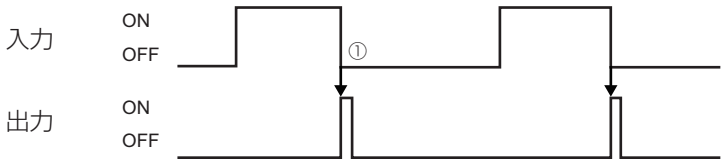
引数

サンプルプログラム

bLimitの立ち上がりでbStopを1スキャンだけONします。

動作説明

「入力」で指定したビットの立ち下がりで1スキャンだけ「出力」がONします。

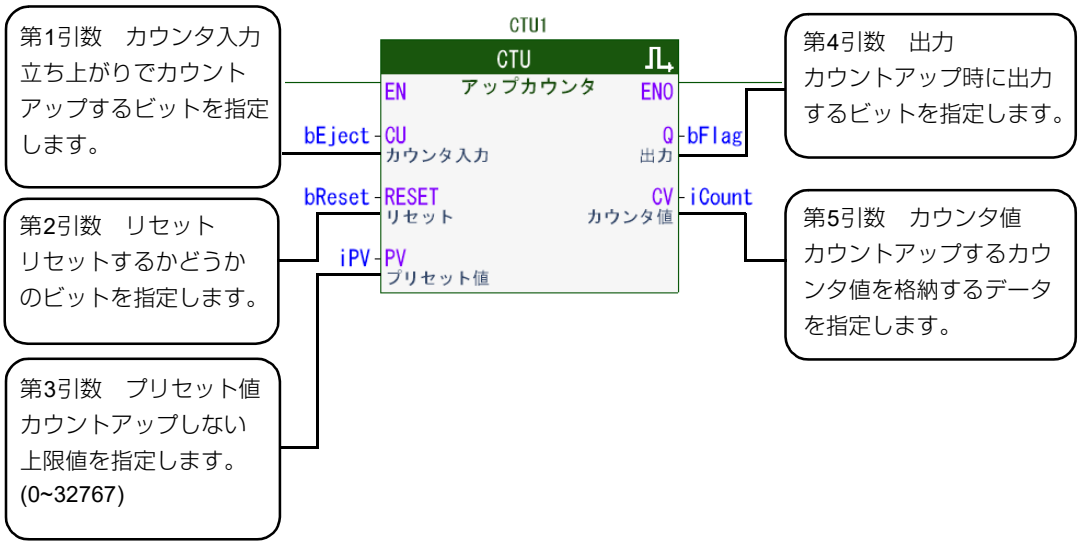


①　入力の立ち下がり時に出力が1スキャンだけONします。

CTU　アップカウンタ

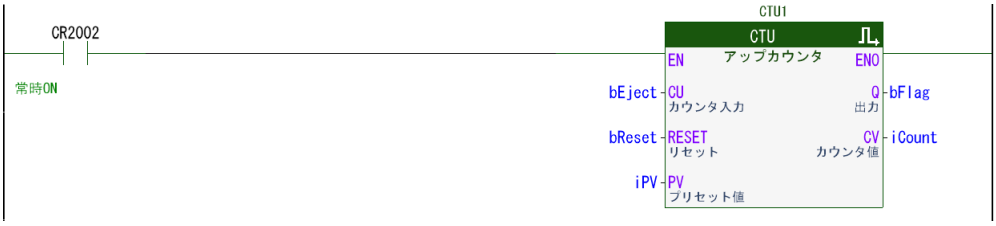
16ビット加算カウンタです。

引数



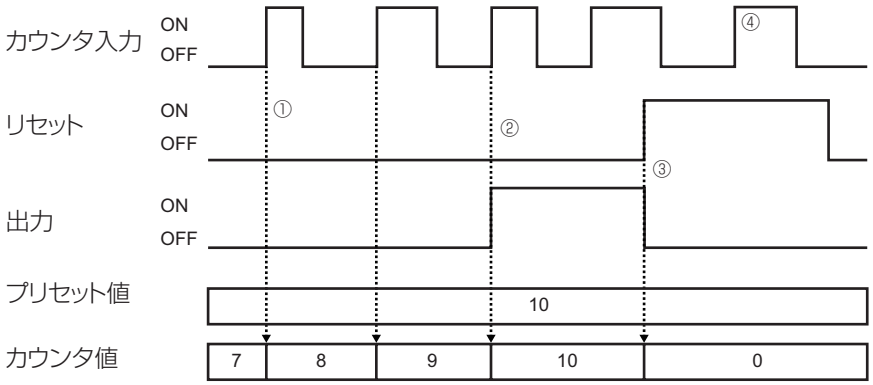
サンプルプログラム

bEjectの立ち上がり回数でiCountをアップカウントします。



動作説明

「カウンタ入力」で指定したビットの立ち上がりで「カウンタ値」をカウントアップします。「カウンタ値」が「プリセット値」以上になると、「出力」がONし、「カウンタ値」は加算されなくなります。「リセット」をONしている間、「カウンタ値」は0になります。

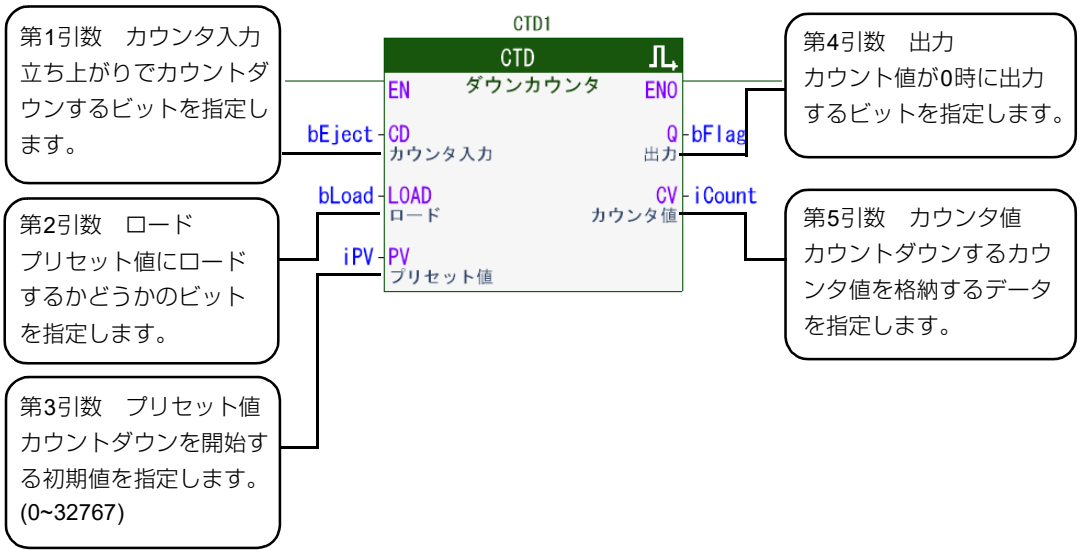


① カウンタ入力の立ち上がりでカウンタ値を加算します。
② カウンタ値がプリセット値以上になると、出力がONします。
③ リセットがONすると、カウンタ値は0になり出力もOFFします。
④ リセットがONしている間は、カウンタ入力がONしてもカウンタ値は加算しません。
（カウンタ入力がON中にリセットがOFFした場合、カウンタ入力の立ち上がりを検出し、カウントアップします。）

CTD　ダウンカウンタ

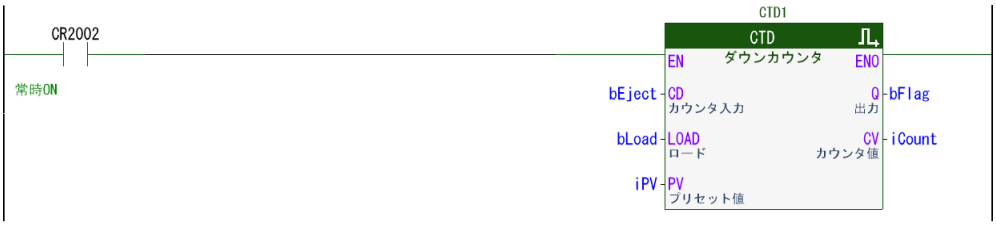
16ビット減算カウンタです。

引数



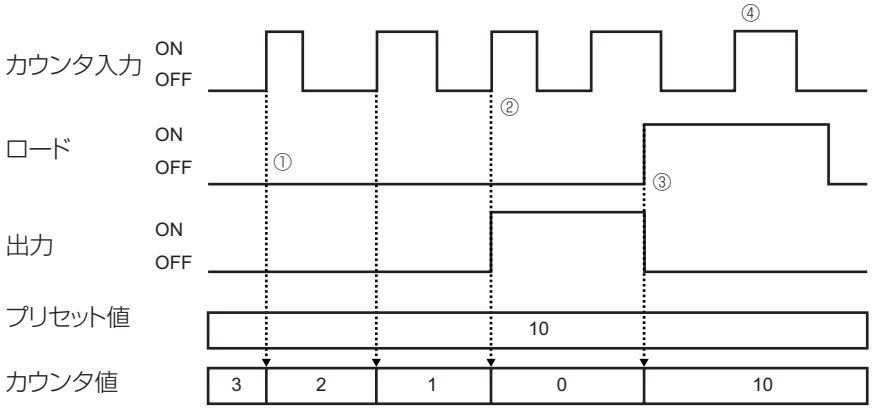
サンプルプログラム

bEjectの立ち上がり回数でiCountをダウンカウントします。



動作説明

「カウンタ入力」で指定したビットの立ち上がりで「カウンタ値」をカウントダウンします。「カウンタ値」は「プリセット値」から開始し、0になると「出力」がONし、「カウンタ値」は減算されなくなります。「ロード」をONしている間、「カウンタ値」は「プリセット値」になります。



① カウンタ入力の立ち上がりでカウンタ値を減算します。

② カウンタ値が0になると、出力がONします。

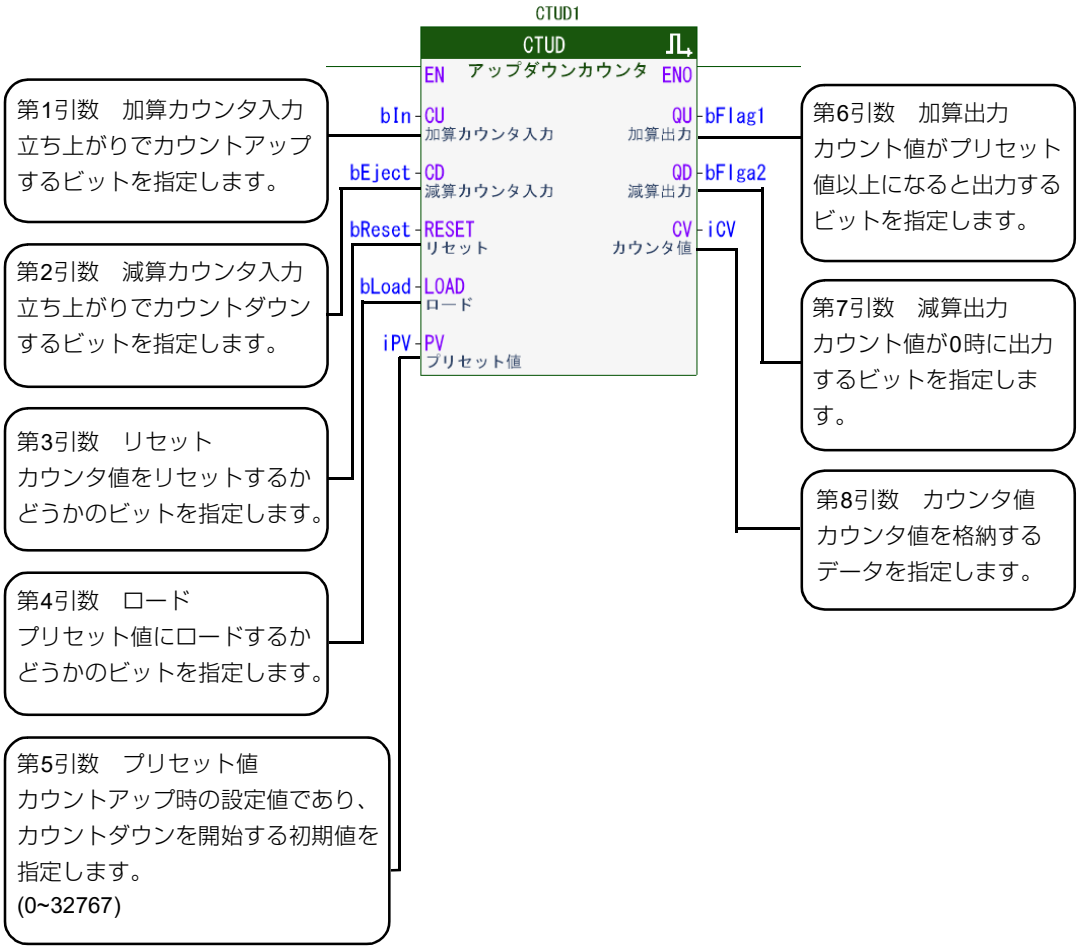
③ ロードがONすると、カウンタ値はプリセット値になり出力もOFFします。

④ ロードがONしている間は、カウンタ入力がONしてもカウンタ値は減算しません。
（カウンタ入力がON中にリセットがOFFした場合、カウンタ入力の立ち上がりを検出し、カウントアップします。）

CTUD　アップダウンカウンタ

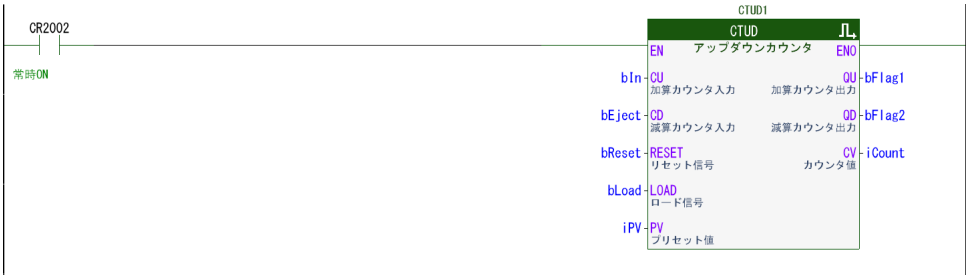
アップダウンカウンタです。

引数



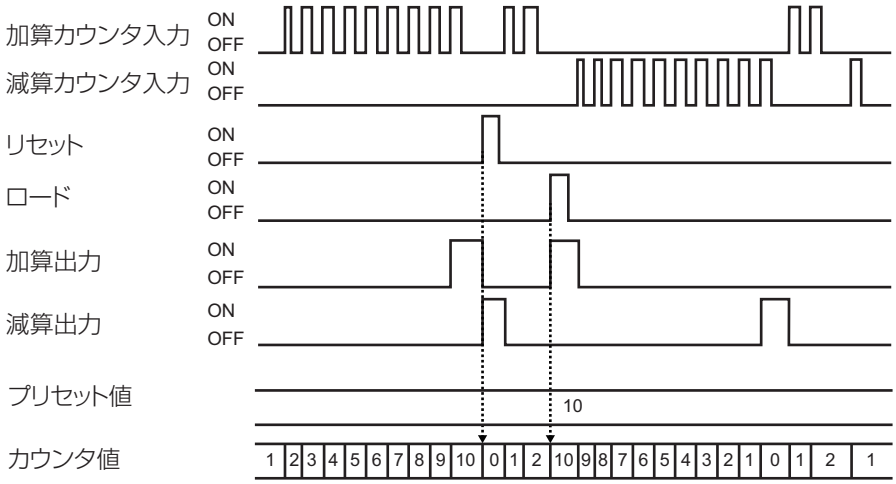
サンプルプログラム

bInでiCountを加算し、bEjectでiCountを減算します。



動作説明

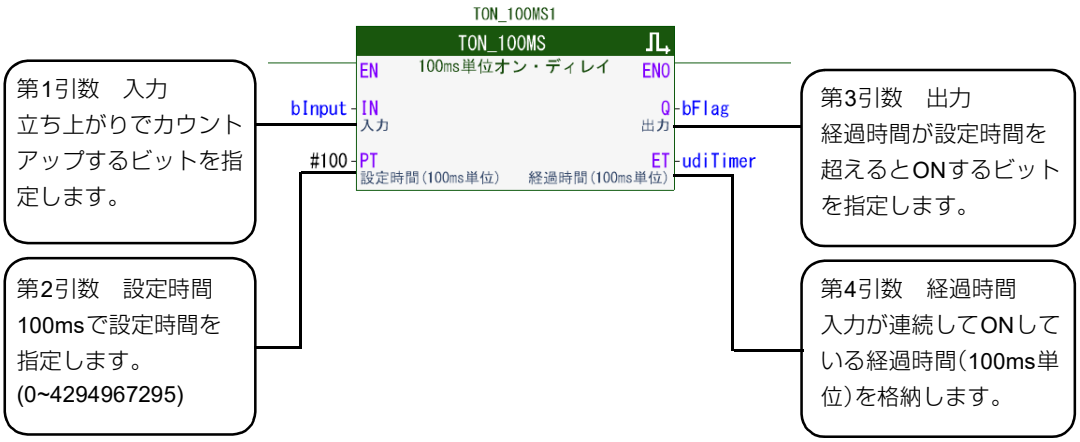
「加算カウンタ入力」で指定したビットの立ち上がりで「カウンタ値」をカウントアップし、「減算カウンタ入力」で指定したビットの立ち上がりで「カウンタ値」をカウントダウンします。「カウンタ値」は0から開始し、「プリセット値」と一致すると「加算出力」がONするか、「プリセット値」から開始し、0と一致すると「減算出力」がONします。「加算出力」「減算出力」がONしている間は、「カウンタ値」は加算減算されなくなります。「リセット」をONしている間、「カウンタ値」は「0」に、「ロード」をONしている間、「カウンタ値」は「プリセット値」になります。「リセット」と「ロード」が同時にONしている場合、「カウンタ値」は「0」になります。



TON_100MS　100ms単位オンディレイ

100ms単位オンディレイです。

引数

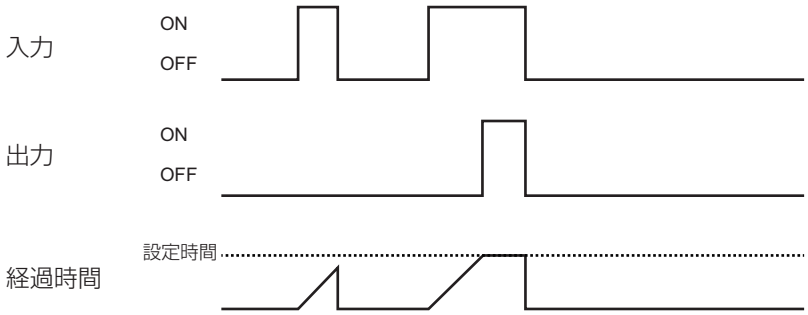


サンプルプログラム

bInputがON中の時間をカウントします。

動作説明

「入力」で指定したビットがONしている間「経過時間」が増加します。「設定時間」を経過すると「出力」がONし、「経過時間」は増えなくなります。「入力」がOFFすると、「経過時間」は0になり、「出力」もOFFします。



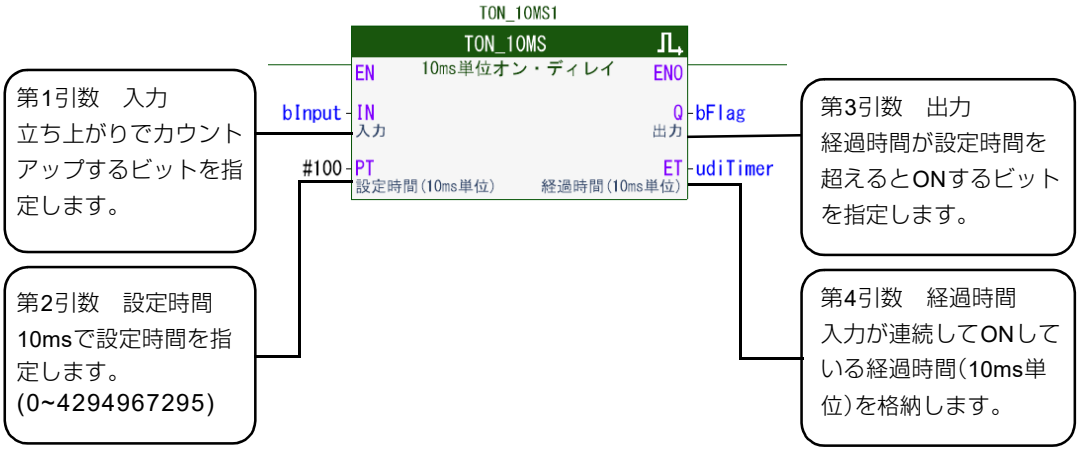
ポイント

- **CJ命令/NCJ命令などによりJMPして実行条件が飛ばされた場合、「経過時間」の値は更新されませんが、その間も計時は継続しています。次に実行されたときには、「経過時間」は0ではなく、計時していた時間に更新されます。ST/スクリプトのIF文中などで使用される場合は、内部的にCJやNCJで実行条件がJMPされていることがありますので注意してください。**
- **ファンクションブロック実行中に「設定時間」を変更しないでください。「経過時間」もオフセットされてしまいます。**

TON_10MS　10ms単位オンディレイ

10ms単位オンディレイです。

引数

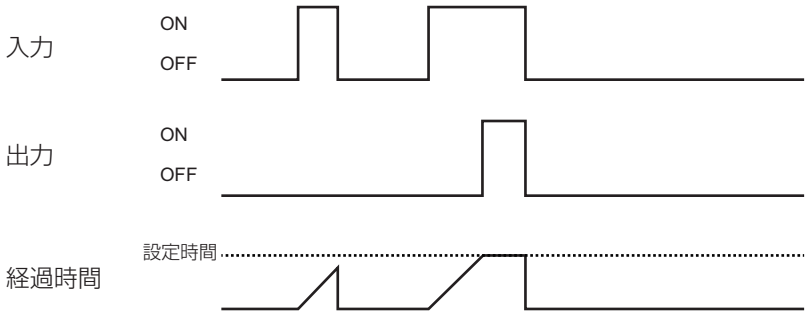


サンプルプログラム

bInputがON中の時間をカウントします。

動作説明

「入力」で指定したビットがONしている間「経過時間」が増加します。「設定時間」を経過すると「出力」がONし、「経過時間」は増えなくなります。「入力」がOFFすると、「経過時間」は0になり、「出力」もOFFします。



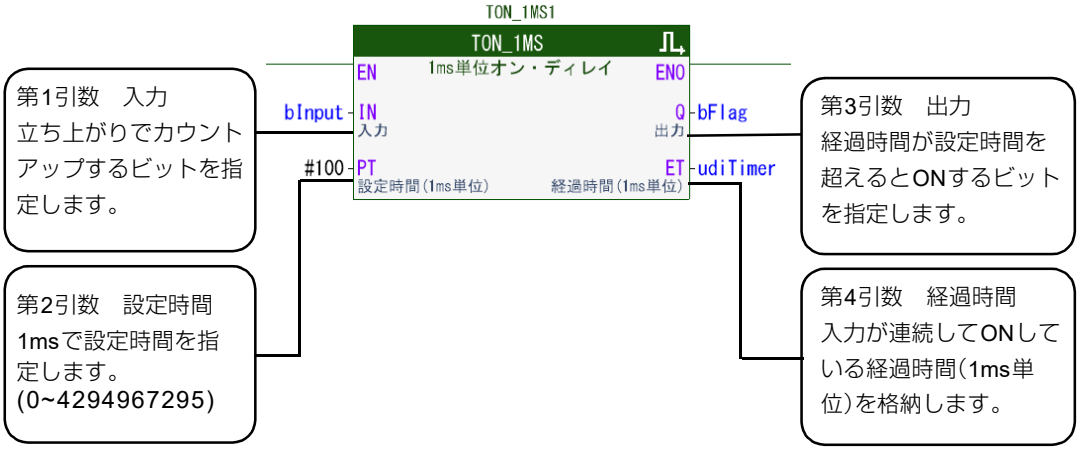
ポイント

- **CJ命令/NCJ命令などによりJMPして実行条件が飛ばされた場合、「経過時間」の値は更新されませんが、その間も計時は継続しています。次に実行されたときには、「経過時間」は0ではなく、計時していた時間に更新されます。ST/スクリプトのIF文中などで使用される場合は、内部的にCJやNCJで実行条件がJMPされていることがありますので注意してください。**
- **ファンクションブロック実行中に「設定時間」を変更しないでください。「経過時間」もオフセットされてしまいます。**

TON_1MS　1ms単位オンディレイ

1ms単位オンディレイです。

引数

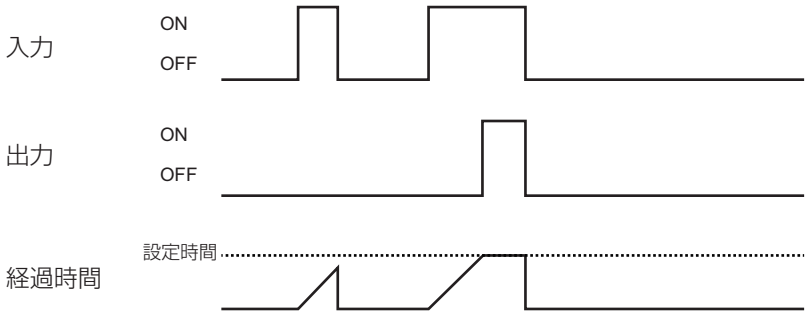


サンプルプログラム

bInputがON中の時間をカウントします。

動作説明

「入力」で指定したビットがONしている間「経過時間」が増加します。「設定時間」を経過すると「出力」がONし、「経過時間」は増えなくなります。「入力」がOFFすると、「経過時間」は0になり、「出力」もOFFします。



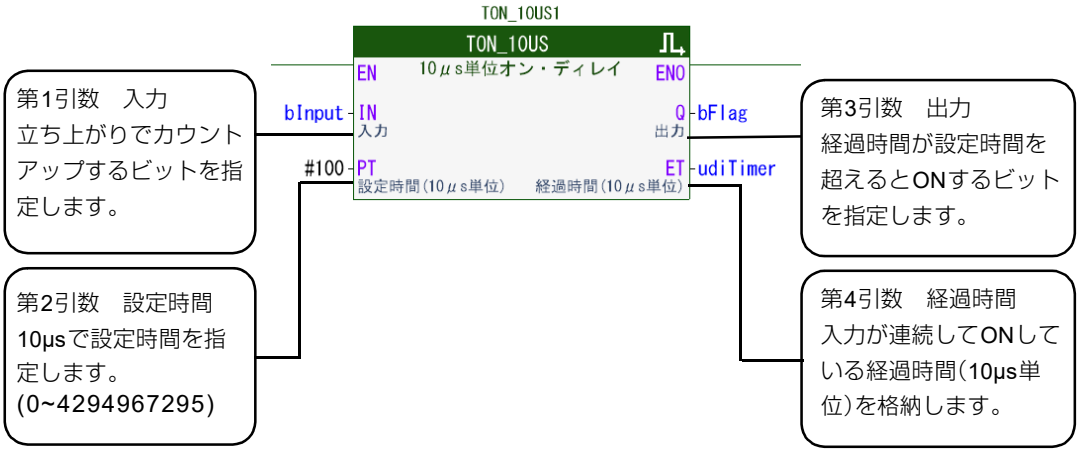
ポイント

- **CJ命令/NCJ命令などによりJMPして実行条件が飛ばされた場合、「経過時間」の値は更新されませんが、その間も計時は継続しています。次に実行されたときには、「経過時間」は0ではなく、計時していた時間に更新されます。ST/スクリプトのIF文中などで使用される場合は、内部的にCJやNCJで実行条件がJMPされていることがありますので注意してください。**
- **ファンクションブロック実行中に「設定時間」を変更しないでください。「経過時間」もオフセットされてしまいます。**

TON_10US　10μs単位オンディレイ

10μs単位オンディレイです。

引数

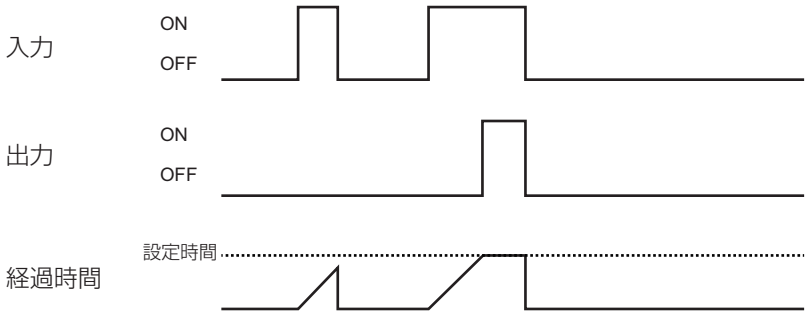


サンプルプログラム

bInputがON中の時間をカウントします。

動作説明

「入力」で指定したビットがONしている間「経過時間」が増加します。「設定時間」を経過すると「出力」がONし、「経過時間」は増えなくなります。「入力」がOFFすると、「経過時間」は0になり、「出力」もOFFします。



ポイント

- **CJ命令/NCJ命令などによりJMPして実行条件が飛ばされた場合、「経過時間」の値は更新されませんが、その間も計時は継続しています。次に実行されたときには、「経過時間」は0ではなく、計時していた時間に更新されます。ST/スクリプトのIF文中などで使用される場合は、内部的にCJやNCJで実行条件がJMPされていることがありますので注意してください。**
- **ファンクションブロック実行中に「設定時間」を変更しないでください。「経過時間」もオフセットされてしまいます。**

ファンクションブロックとマクロの違い

ファンクションブロックはマクロと同様に、ラダー回路を1つのラダー命令語のようにまとめる機能です。ファンクションブロックの動作をマクロの動作と比較して説明します。
マクロについての詳細は、「マクロ」(4-178ページ)を参照してください

● 実体の生成

【ファンクションブロック】
プログラムの種類:ファンクションブロックの場合
インスタンスの個数だけ実体が生成されます。
プログラムの種類:ファンクションの場合
プログラム内で呼び出す数によらず、1つの同じ実体が生成されます。

【マクロ】
プログラム内で呼び出すごとに、同じ数の実体が生成されます。

● 呼び出せる箇所

【ファンクションブロック】
プログラムの種類:ファンクションブロックの場合
モジュール、ファンクションブロックから呼び出せます。
プログラムの種類:ファンクションの場合
モジュール、ファンクションブロック、ファンクションから呼び出せます。
最大8段までネストできます。
「ファンクションブロックのネスト」(4-140ページ)

【マクロ】
モジュールから呼び出せます。マクロからマクロを呼び出すことはできません。

● 引数タイプ

【ファンクションブロック】
IN　:実行開始時、デバイスの値を引数に格納します。
OUT　:実行完了時、引数の値をデバイスに格納します。
IN-OUT　:実行開始時、デバイスへの参照を格納します。

【マクロ】
P　:実行開始時、デバイスへの参照を格納します。
V　:実行開始時、デバイスの値を引数に格納します。

● 引数に指定できる変数の種類

以下の変数はファンクションブロックのみで指定でき、マクロでは指定できません。

- IN、OUT引数　:固定長配列、構造体、固定長文字列
- IN-OUT引数　:固定長配列、可変長配列、構造体、固定長文字列、可変長文字列

参考　KV-8000シリーズで新規のプログラムを作成する際は、マクロではなくファンクションブロックでの作成を推奨します。

● その他の機能差分一覧

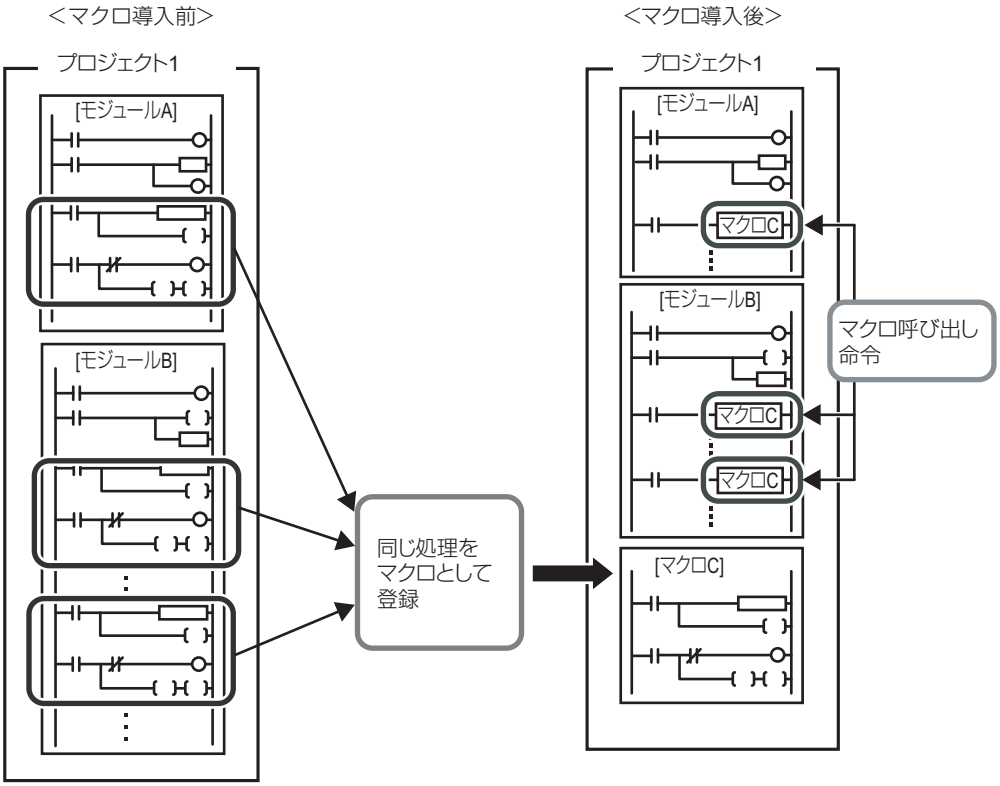
項目		ファンクションブロック	マクロ
プログラム	プログラム名	最大64文字	最大12文字
	読み取り専用設定	○	―
引数	個数	IN、OUT、IN-OUT：それぞれ最大32個 ユニット：最大8個 合計：最大64個	最大10個
	引数名	IN、OUT：IN-OUT： 最大128文字 ユニット： UR0～7、UM0～7、UV0～7 のローカルデバイス	P0～9、V0～9、 UR0～9、UM0～9、UV0～9 のローカルデバイス
	初期値設定	○	―
	非公開設定	○	―
	引数の省略	○	―
システムデバイス	@CR2008	○	○
	@CR2007	○	○
	実行中ONビット	―	○
その他※1	インスタンス	○	―
	EN/ENO	○	―
	戻り値	○	―

※1　CPU機能バージョン2.0以降でのみ対応しています。

4-11 マクロ

概要

プロジェクト内で何度も繰り返して使用するラダー回路に、任意の名前を付けて登録したものをマクロと呼びます。マクロ化することにより、ラダー命令語と同じように使用できるため、プログラム入力工数が削減されます。回路変更時には、マクロとして登録した1箇所のデータを修正するだけで、プロジェクト内で呼び出す全てのマクロ命令の処理内容が変更されます。このことにより、プログラムの変更工数が削減され、修正ミスも防げます。



■ マクロの管理

《KV STUDIO》では、[ワークスペース]を使用してプロジェクトごとに管理します。

■ マクロの特長

● オリジナル命令語として使用できる

プログラム内に何度も出てくる同じラダー回路をマクロとして登録すれば、以後の入力工数や入力ミスを削減できます。
結果として、見た目のラダー行数が減り、プログラム構造を把握しやすくなります。
登録するマクロ名は自由に付けることができます。

● 引数が使用できる

引数を使用すれば、マクロ内にある命令語のオペランド（デバイスや値）をマクロ呼び出し時に指定できるため、汎用性の高いマクロを作成できます。

「引数設定」（4-186ページ）

● 2種類のマクロが作成できる

マクロには、『サブルーチン型マクロ』と『自己保持型マクロ』があります。
『サブルーチン型マクロ』は、サブルーチンプログラムと同様に実行条件がONの場合のみ実行します。
『自己保持型マクロ』は、一旦実行を開始するとマクロ実行停止命令（MEND（マクロエンド））が実行されるまで、実行条件の状態にかかわらず動作を続けます。
拡張ユニットや外部機器に処理を要求し、完了までに時間がかかる（スキャンをまたぐ）処理の場合などに便利です。

「マクロの種類」（4-181ページ）

● ローカルデバイスが使用できる

別のマクロで同じ番号のローカルデバイスを使用しても、別デバイスとして扱われるため、マクロ内のみで使用するデバイスの割り付けが簡単にできます。

「4-12 ローカルデバイス」（4-200ページ）

● パスワードが設定できる

マクロの内容を容易に変更できなくするため、パスワードを設定できます。
パスワードを設定した場合、パスワードを知らない人はマクロのプログラムを表示、編集できません。
［プログラム プロパティ］ダイアログでパスワードを設定します。

● 呼び出すごとに実体を作成する

マクロは、モジュール内で呼び出されるごとに、ラダープログラムの実体を作成します。同じスキャン内で何度も同じマクロを実行する場合や、スキャンをまたいで実行するマクロが複数ある場合、ローカルデバイスを併用することで、使用するデバイスの衝突やそれぞれのマクロの実行状態を意識せずに使用できます。

「マクロとサブルーチンプログラムの違い」（4-196ページ）
「ファンクションブロックとマクロの違い」（4-175ページ）

使用手順

マクロの使用手順は次のとおりです。

① マクロの新規作成

【プログラム(M)】→【新規作成(M)】

プログラムの種類でマクロを選択します。 「マクロの種類」(4-181ページ)

▼

② ラダープログラムの作成

作成したマクロにラダープログラムを記述します。

▼

③ 引数の設定

ワークスペース上で、右クリック→【引数設定(Z)】

引数を登録します。 「引数設定」(4-186ページ)

▼

④ マクロ呼び出し命令の記述

作成したマクロを専用命令で呼び出します。 「マクロ命令」(4-193ページ)

必要に応じてシステムデバイスを使用します。「マクロシステムデバイス」(4-188ページ)

マクロの種類

マクロには、サブルーチン型マクロと自己保持型マクロがあります。

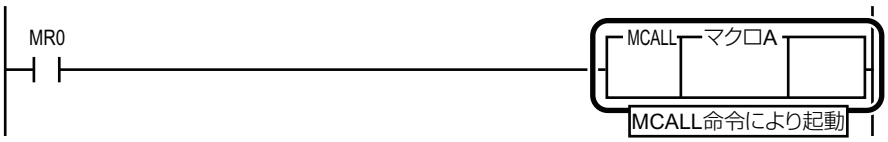
サブルーチン型マクロ

● 概要

サブルーチン型マクロは、サブルーチンと同様に実行条件がONの場合のみ、マクロ呼び出し命令（MCALL）ごとにマクロ内のプログラムを実行します。実行完了後、マクロ呼び出し命令の次のステップに進みます。実行条件がOFFの場合は、マクロ内のプログラムを実行せずに次のステップに進みます。

● 使用方法

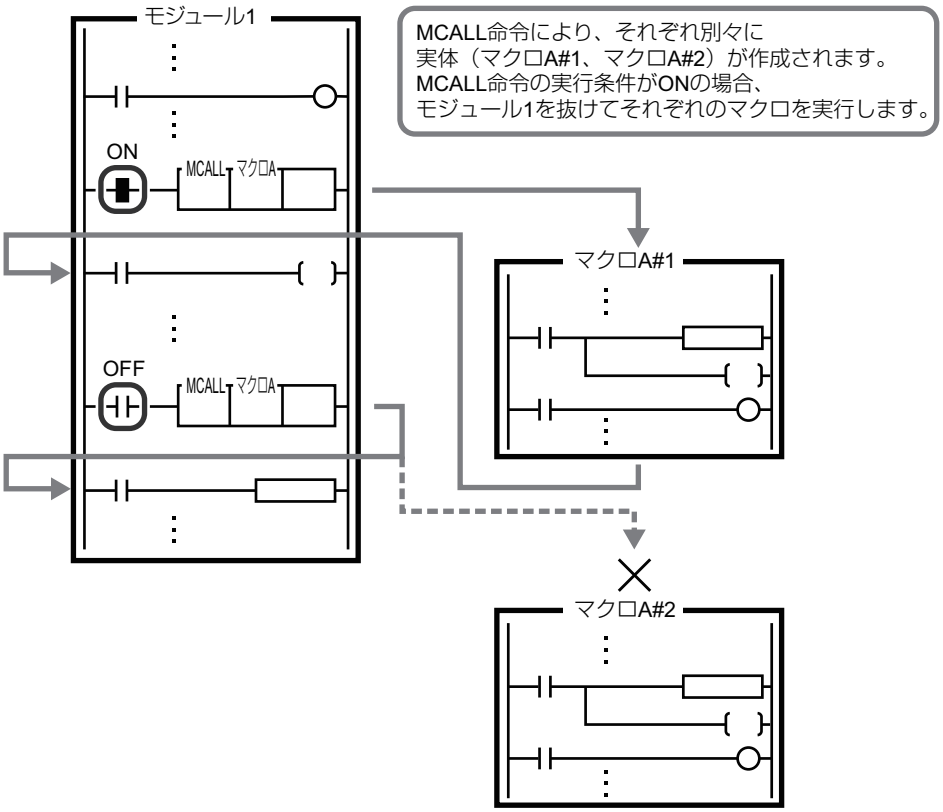
サブルーチン型マクロは、モジュールからのMCALL命令により起動します。



● 動作説明

サブルーチン型マクロの動作説明を下図に示します。
モジュール1内でマクロAを2回呼び出す例を示します。

参 考　作成したマクロの実体を区別するために、マクロ名の後ろに「#（管理番号）」を付けて管理します。《KV STUDIO》では、モニタ･シミュレータ時のみ表示されます。



● 実行停止時の動作

サブルーチン型マクロの実行停止時(実行条件ON→OFF時)の動作は、各プログラムの[プログラムプロパティ]→「実行停止時にOFF処理する」の設定により異なります。

ポイント

KV-8000Aの場合は設定は表示されず、常に「実行停止時にOFF処理する」にチェックしている場合と同じ動作になります。

「実行停止時にOFF処理する」にチェックしていない場合

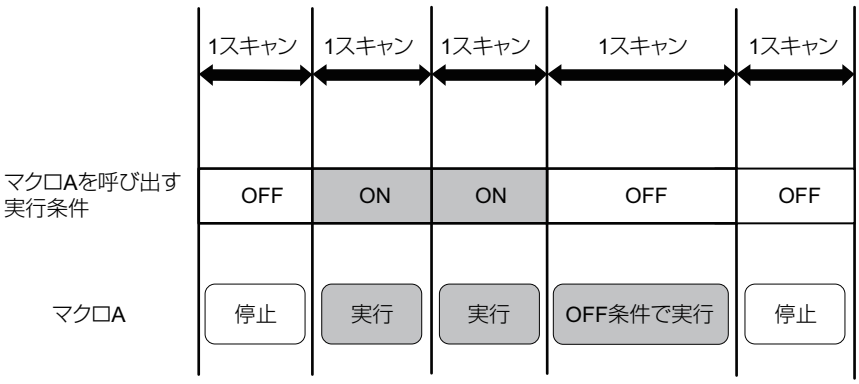
サブルーチン型マクロは、実行条件がOFFになったスキャンで停止します。

「実行停止時にOFF処理する」にチェックしている場合

サブルーチン型マクロは、実行条件がOFFになった1スキャンのみ、実行条件を全てOFFとして実行されます。※

その次のスキャンで停止します。

【サブルーチン型マクロの動作イメージ】



※ ただし、以下の命令は例外となります。

命令語	状態
OUB命令	OFFします。
OFDL命令	リセットします。
SHOT命令	
C命令	状態保持します。
CJ/NCJ/GOTO命令	分岐しません。

参 考

マクロの実行停止時のOFF処理は、MC命令の実行条件がOFF時と同じ処理を1スキャンのみおこないます。

『KV-8000/7000/5000/3000/1000シリーズ ·KV Nanoシリーズ命令語リファレンスマニュアル』「MC/MCR」

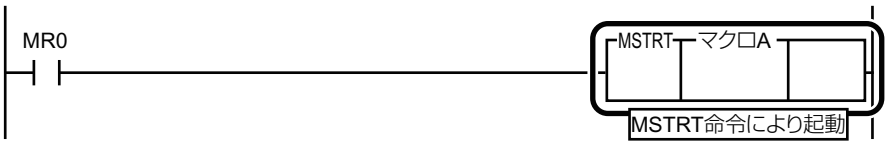
自己保持型マクロ

● 概要

自己保持型マクロは、サブルーチン型マクロと同様に実行条件がONの場合、マクロ呼び出し命令（MSTRT）ごとにマクロ内のプログラムを実行します。一旦マクロが起動すると、次スキャン以降もマクロ内のマクロ実行停止命令（MEND（マクロエンド））が実行されるまで、マクロ呼び出し命令の実行条件にかかわらず動作し続けます。

● 使用方法

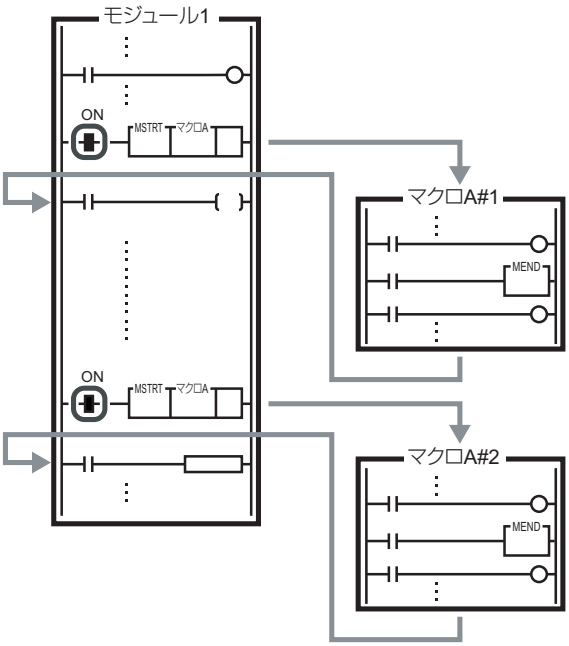
自己保持型マクロは、モジュールからのMSTRT命令により起動します。



● 動作説明

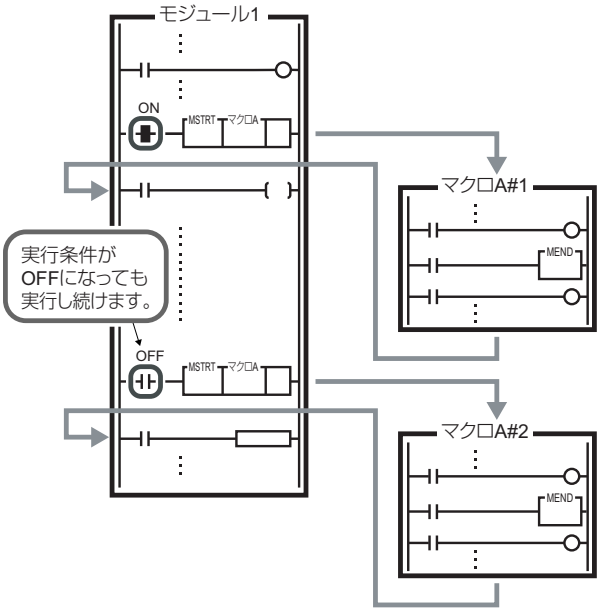
自己保持型マクロの動作説明を下図に示します。
ここでは、モジュール1内でマクロAを2回呼び出す例を示します。

1 実行条件がONの場合、マクロＡを起動します。

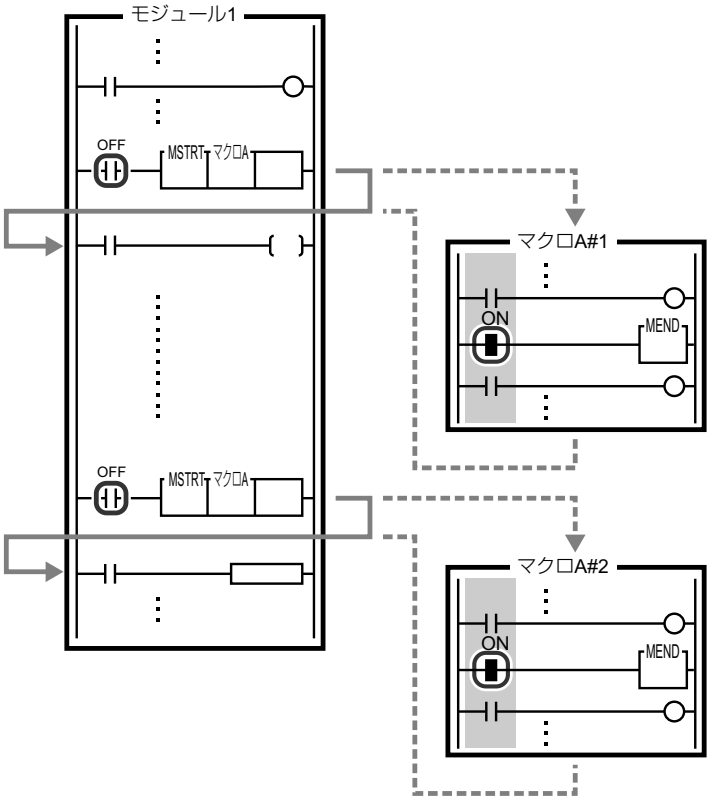


参 考　作成したマクロの実体を区別するために、マクロ名の後ろに「#（管理番号）」を付けて管理します。《KV STUDIO》では、モニタ･シミュレータ時のみ表示されます。

2 一旦起動すると、MSTRT命令の実行条件の状態にかかわらず実行し続けます。



3 マクロ内にあるMEND命令の実行条件がONすると、停止します。



ポイント

- 自己保持型マクロにはMEND命令が必ず必要です。
- マクロの実行を停止するには、マクロの実行条件を必ずOFFにしてください。マクロの実行条件がONの場合は、MEND命令の実行条件がONしてもマクロは停止しません。

● 実行停止時の動作

自己保持型マクロの実行停止時(MEND命令実行時)の動作は、各プログラムの[プログラムプロパティ]→「実行停止時にOFF処理する」の設定により異なります。

ポイント **KV-8000Aの場合は設定は表示されず、常に「実行停止時にOFF処理する」にチェックしている場合と同じ動作になります。**

「実行停止時にOFF処理する」にチェックしていない場合

自己保持型マクロは、マクロ停止命令(MEND)が実行された次のスキャンで停止します。

「実行停止時にOFF処理する」にチェックしている場合

自己保持型マクロは、マクロ停止命令(MEND)が実行された次の1スキャンのみ、実行条件を全てOFFとして実行されます。※
その次のスキャンから停止となります。

【自己保持型マクロの動作イメージ】

	1スキャン	1スキャン	1スキャン	1スキャン	1スキャン	1スキャン
マクロAを呼び出す実行条件	OFF	ON	※1	※1	OFF※2	OFF※3
マクロA内にあるMEND命令の実行条件	-	OFF	OFF	ON	※1	-
マクロA	停止	実行	実行	実行	OFF条件で実行	停止

※1: ON/OFFどちらの状態でも動作に影響ありません。
※2: 必ずOFFしてください。OFFするまでMEND命令を無視して実行し続けます。
※3: ON状態であれば、マクロの実行を開始します。

※ ただし、以下の命令は例外となります。

命令語	状態
OUB命令	OFFします。
OFDL命令	リセットします。
SHOT命令	
C命令	状態保持します。
CJ/NCJ/GOTO命令	分岐しません。

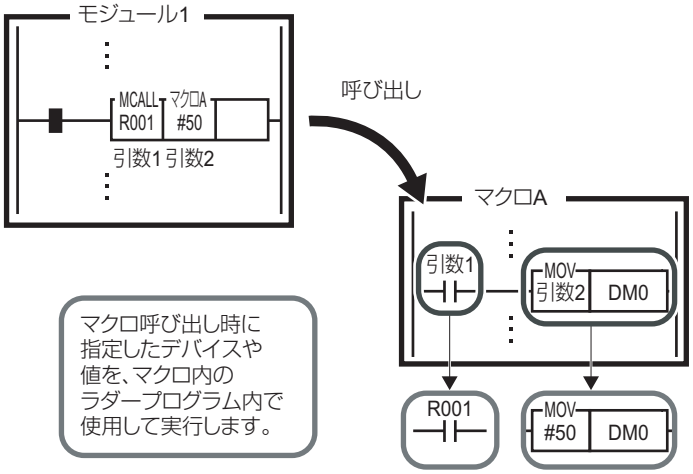
参考 マクロの実行停止時のOFF処理は、MC命令の実行条件がOFF時と同じ処理を1スキャンのみおこないます。

『KV-8000/7000/5000/3000/1000シリーズ ·KV Nanoシリーズ 命令語リファレンスマニュアル』「MC/MCR」

引数設定

■ 引数とは

引数とは、マクロ呼び出し時にマクロの実体に対して引き渡すデバイスや値です。マクロ呼び出し命令のオペランドとして指定します。マクロ内では、引き渡されたデバイスや値を使用してラダープログラムを実行します。引数は、マクロが実行中であれば毎スキャン更新されます。



■ 引数デバイス一覧

引数デバイスとは、その引き渡されたデバイスや値が、マクロ内のどの命令語のオペランドとして使用されるかを指定するためのデバイスです。引数デバイスは、マクロ内でしか使用できません。
ワークスペースの設定したいマクロ上で、右クリックメニューから【引数設定(Z)】を選択して表示される[マクロの引数設定]ダイアログで設定します。引数デバイスにはP、V、UR、UM、UVの5種類があります。各デバイスの仕様を下図に示します。マクロの引数は最大10個まで指定できます。

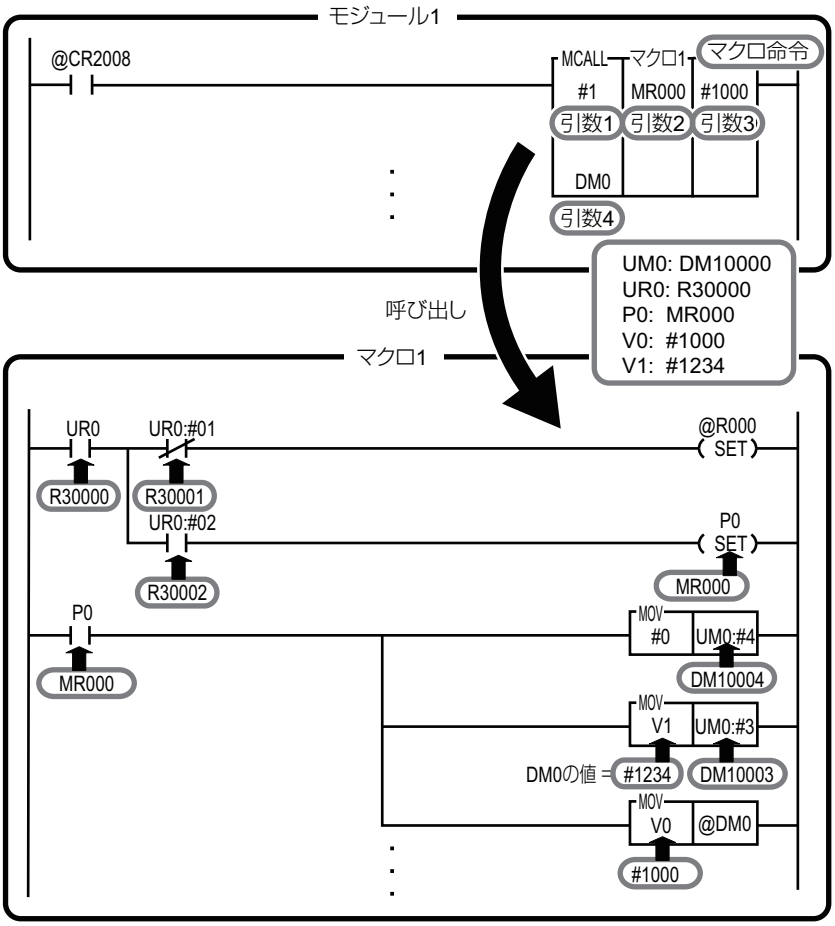
種類	記号	範囲	説明	間接指定(＊)	ローカルデバイス(@)	インデックス修飾(：)
デバイス	P	P0～P9	デバイスを引数として受け取ります。	×	×	○
値※2	V	V0～V9	デバイスの値または定数を引数として受け取ります。	×	×	×
ユニットデバイス(R)※1	UR	UR0～UR9	拡張ユニットに割り付けられた先頭リレー(R)を引数として受け取ります。	×	×	○
ユニットデバイス(DM)※1	UM	UM0～UM9	拡張ユニットに割り付けられた先頭データメモリ(DM)を引数として受け取ります。	×	×	○
ユニット番号※1	UV	UV0～UV9	拡張ユニットのユニット番号を受け取ります。	×	×	×

※1　マクロ呼び出し命令では、対象となるユニット番号のみ指定します。
　　1つのマクロ内で引数として指定できるユニットは、最大5台です。
※2　DRを指定すると、マクロ呼び出し直前にダイレクト処理をおこない、その結果を引数として受け取ります。

● 使用例

引数デバイスの使用例を下図に示します。
モジュール1からマクロ1のマクロ呼び出し命令を実行した場合の例です。

【引数デバイス使用例】
引数1 : UR0/UM0←#1
引数2 : P0←MR000
引数3 : V0←#1000
引数4 : V1←DM0
（ユニット番号1）…KV-SAD04　（R30000～R30115 / DM10000～DM10059）



ポイント　文字列を引数として使用する場合は、引数デバイス「P」を使用して文字列を格納した先頭デバイスを指定してください。引数デバイス「V」を使用して直接文字列を指定することはできません。

マクロシステムデバイス

マクロシステムデバイスとは、マクロの動作状況を取得するためのデバイスです。
実行中ONビットのみ、他のプログラムから参照可能です。

ポイント **マクロシステムデバイスには、デバイスコメントを付けられません。**

参考 マクロシステムデバイスは、《KV STUDIO》によってCPUユニットの内部ワーク領域へ自動的に割り付けられます。

● マクロ実行開始時1スキャンのみONビット「@CR2008」

マクロが実行を開始した1スキャン目のみONになります。
2スキャン目以降はOFFになります。
自マクロ内のみ参照可能です。マクロを呼び出しているモジュール内では参照できません。
《KV STUDIO》上での表記は、「@CR2008」となります。

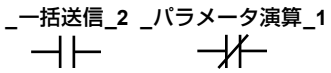
● マクロ実行開始時1スキャンのみOFFビット「@CR2007」

マクロが実行を開始した1スキャン目のみOFFになります。
2スキャン目以降はONになります。
自マクロ内のみ参照可能です。マクロを呼び出しているモジュール内では参照できません。
《KV STUDIO》上での表記は、「@CR2007」となります。

● 実行中ONビット「_(マクロ名)_(識別番号)」

実行中ONビットは、そのマクロが実行中にONします。
各モジュールから参照できるため、マクロ呼び出しの前に参照するなどして、マクロの実行を制御する場合に有効です。
《KV STUDIO》上での表記は、「_(マクロ名)_(識別番号)」(アンダースコア+マクロ名+アンダースコア+識別番号)となります。

例



参考 マクロの実行中ONビットを使用するには、呼び出すマクロごとに識別番号が必要です。
「マクロの実行中ONビット」(4-189ページ)

マクロの実行中ONビット

■ 概要

実行中ONビットは、そのマクロが実行中にONします。各モジュールから参照できるため、マクロ呼び出しの前に参照するなどして、マクロの実行を制御する場合に有効です。
マクロの実行中ONビットを使用するためには、呼び出すマクロごとに識別番号が必要です。
《KV STUDIO》上での表記は、「_(マクロ名)_(識別番号)」となります。

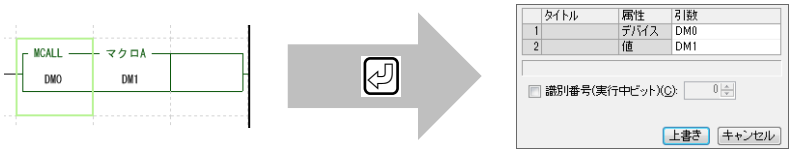
■ 実行中ONビット設定方法

実行中ONビットは、[オペランド編集]ダイアログの「識別番号(実行中ビット)」にチェックを入れると使用できます。

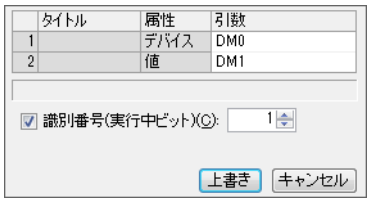
1 MCALL/MSTRT命令にカーソルを合わせて ⏎ キーを入力します。

[オペランド編集]ダイアログが表示されます。

別手順　MCALL/MSTRT命令にカーソルを合わせて、メニュー【編集(E)】→【オペランド変更(O)】を選択



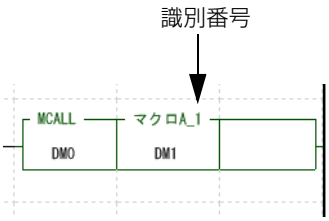
2 「識別番号(実行中ビット)」にチェックを入れて、識別番号を入力します。



ポイント　**マクロの実行中ONビットは、[識別番号(実行中ビット)]にチェックが入っていないと使用できません。**

3 〔上書き〕ボタンをクリックします。

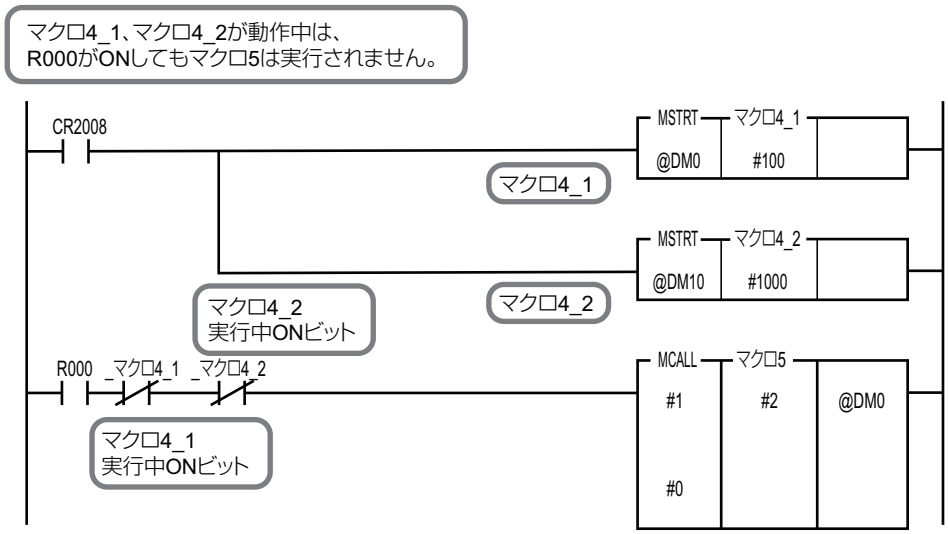
識別番号(実行中ビット)が設定されます。



同じプロジェクト内で同じマクロを何度も使用する場合は、番号を変えて使用します。

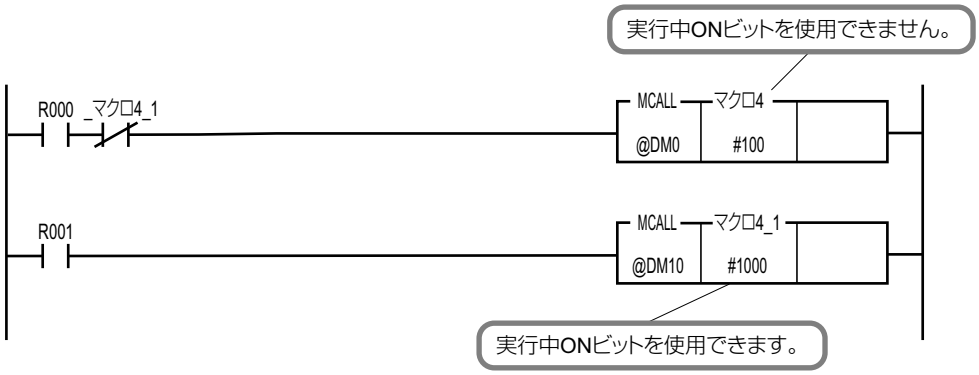
ポイント　**同じマクロに同じ番号を指定すると、変換エラーになります。**

● 使用例



ポイント **未使用マクロ(プロジェクト内のマクロ呼び出し命令で指定されていないマクロ)の実行中ONビットを記述すると、変換エラーになります。**

参 考 同じマクロで識別番号ありと識別番号なしのマクロ呼び出し命令を併用できます。



マクロパスワード

■ マクロのパスワード

マクロにパスワードを設定して、《KV STUDIO》でモジュールの表示に制限をかけることができます。
次の手順でパスワードを設定します。

1 **[ワークスペース]上の設定したいマクロを選択します。**

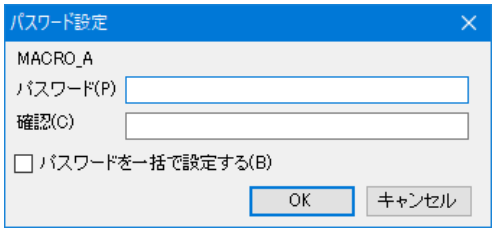
2 **右クリックして表示されるメニューから[プロパティ(A)]を選択します。**

[プログラムプロパティ]ダイアログが表示されます。

別手順
- ワークスペースでマクロを選択し、メニューから【プログラム(M)】→【プロパティ(H)】を選択
- ワークスペースでマクロを選択し、右クリックメニューから【パスワード設定(B)】を選択

3 **〔パスワード設定〕ボタンを押します。**

[パスワード設定]ダイアログが表示されます。



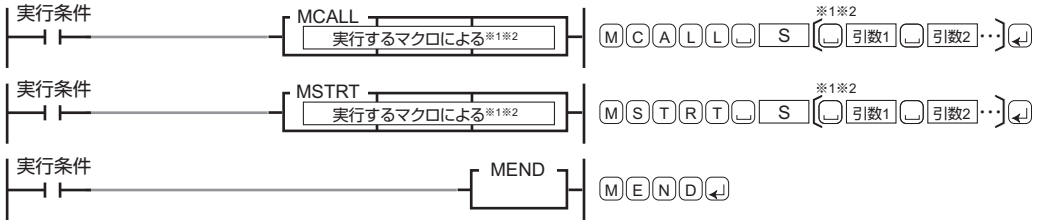
4 **パスワードとパスワードの確認を入力して、〔OK〕ボタンを押します。**

パスワード設定後、パスワード設定したマクロを《KV STUDIO》で表示する時、[パスワード解除]ダイアログが表示されます。
パスワードを入力しないと、マクロを《KV STUDIO》上で表示できません。

MEMO

MCALL	MCALL	マクロコール	指定したサブルーチン型マクロを実行します。(モジュール内に記述)
MSTRT	MSTRT	マクロスタート	指定した自己保持型マクロを実行します。(モジュール内に記述)
MEND	MEND	マクロエンド	自己保持型マクロの実行を停止します。(自己保持型マクロ内に記述)

ラダープログラム　　入力方法



オペランド	説明	占有サイズ
S	マクロ名称を指定します。※3※4	―

※1　マクロ名称以外のオペランドの個数や、指定できるデータは、実行するマクロに設定されている引数設定によって以下のとおり異なります。
- P　　:デバイス、変数、ラベルを指定できます。定数は指定できません。
- V　　:デバイス、変数、ラベル、定数を指定できます。
- ユニット:定数のみ指定できます。

※2　マクロ内で使用する命令語によって、T、C、TIMER 型変数、COUNTER 型変数の扱いが異なります。
例) MOV命令の場合は、現在値を扱います。詳細は各命令語の動作説明を参照してください。

※3　""、#、$は使用できません。KV STUDIOで設定してマクロ名称を直接入力してください。
半角英数字の場合は12文字、全角の場合は6文字まで指定できます。
ただし、KV-8000/7000シリーズの場合は全角でも12文字まで指定できます。

※4　識別番号を使用する場合は、「(マクロ名)_(識別番号)」になります。　例)一括送信_2
識別番号の詳細は、使用しているCPUユニットのユーザーズマニュアルを参照してください。

動作説明

MCALL 実行条件がONの間、[S]で指定したサブルーチン型マクロを毎スキャン実行します。
指定したマクロは、OFF→ONになったスキャンから実行を開始します。

MSTRT 実行条件がONのとき、[S]で指定した自己保持型マクロを起動します。
指定したマクロは、OFF→ONになったスキャンから実行を開始します。
一旦実行を開始すると、実行条件がOFFでもマクロを実行し続けます。

MEND 実行条件がONのとき、MEND命令が記述されている自己保持型マクロを停止します。

プログラムの設定により、停止時のOFF処理の動作が異なります。
サブルーチン型マクロ：「実行停止時の動作」(4-182ページ)
自己保持型マクロ：「実行停止時の動作」(4-185ページ)

ポイント

- **MCALL/MSTRT命令の呼び出し個数や順序を変更すると、RUN中書き込みできません。**
- **自己保持型マクロの実行を停止するには、マクロ内のMEND命令を使用します。**
  MSTRT命令の実行条件がONのままの場合には、MEND命令は無視されます。
- **MEND命令は、モジュール内やサブルーチン型マクロ内に記述できません。**
- **MEND命令で他のマクロを停止することはできません**
- **マクロ内の微分実行型命令の前回状態は、マクロの実行開始時に毎回初期化されます。**
  (立ち上がり微分：ON、立ち下がり微分：OFF)
- **MCALL/MSTRT命令の属性が値のオペランドにT/C/CTH/CTCを指定した場合は、以下の値がマクロに引き渡されます。**
  T/C ：現在値(32ビットデータ)
  CTH ：現在値(32ビットデータ)
  CTC ：設定値(32ビットデータ)
- **以下の場合、各モジュールが停止もしくは各命令間の実行条件がOFFしたとき、実行中のマクロは停止します。**
  - **MCALL/MSTRT命令をモジュールに記述した場合**
  - **MCALL/MSTRT命令をマスターコントロール(MC～MCR命令間)に記述した場合**
  - **MCALL命令をステージ(STG～JMP命令間)に記述した場合**
- **以下の場合、各命令間の実行条件がOFFしたとき、実行中のマクロは中断します。**
  次に各命令間が実行されたときにマクロの実行を再開します。
  このときマクロ内の微分実行型命令の前回状態は、初期化されません。
  (初期化モジュール内に記述した場合は、 運転開始時1スキャンのみ実行します。)
  - **MCALL/MSTRT命令をサブルーチン(SBN～RET命令間)に記述したとき**
  - **MCALL/MSTRT命令を条件分岐(CJ/NCJ～LABEL命令間)に記述したとき**
  - **MCALL/MSTRT命令をステップ(STP～STE命令間)に記述したとき**
  - **MCALL/MSTRT命令を割り込み(INT～RETI命令間)に記述したとき**
  - **MSTRT命令をステージ(STG～JMP命令間)に記述したとき**
- **マクロ内で、MCALL命令、MSTRT命令は使用できません。**
- **ファンクションブロック・ファンクション内でMCALL/MSTRT命令は使用できません。**
- **マクロ内でサブルーチンプログラムを使用できますが、割り込みプログラムは使用できません。**
- **プロジェクト内で同じマクロを複数回実行する場合、マクロ内でOUT/OUB命令にグローバルデバイス/グローバル変数を使用すると、2重コイルになります。**

演算フラグ

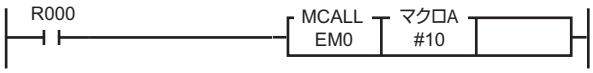
CR2009	実行するマクロ内の処理結果に依存します。
CR2010	実行するマクロ内の処理結果に依存します。
CR2011	実行するマクロ内の処理結果に依存します。
CR2012	間接指定またはインデックス修飾の範囲が不適切な場合、またはKV-8000/7000シリーズ使用時にUR/UV/UMで指定したユニットが接続されていない場合、オペランドの範囲外アクセスが発生した場合ON。それ以外OFF。 オペランドに間接指定、インデックス修飾を指定していない場合、およびオペランドの範囲外アクセスのチェックが行われない場合は変化しません。

※ 引数が不適切な場合、マクロは実行されません。
CR2012がONした場合、CM5150～CM5176にエラーの詳細情報が格納されます。
「エラー履歴」(4-86ページ)

ポイント **MEND命令による演算フラグの変化はありません。**

サンプルプログラム

入力リレーR000がONの間、引数1をEM0、引数2を#10としてサブルーチン型マクロ「マクロA」を実行します。



〈ニモニックリスト〉

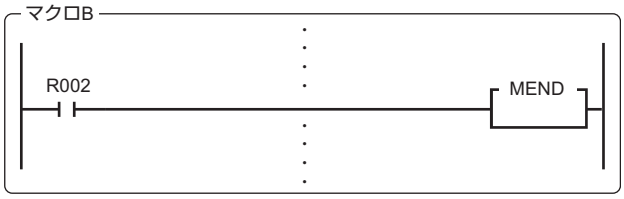
```
LD R000
MCALL マクロ A EM0 #10
```

入力リレーR001の立ち上がりで、引数1(ユニット番号)を#1、引数2を#10として自己保持型マクロ「マクロB」の実行を開始します。

〈ニモニックリスト〉

```
LDP R001
MSTRT マクロ B #1 #10
```

入力リレーR002がONの時、自己保持型マクロ「マクロB」を停止します。



〈ニモニックリスト〉

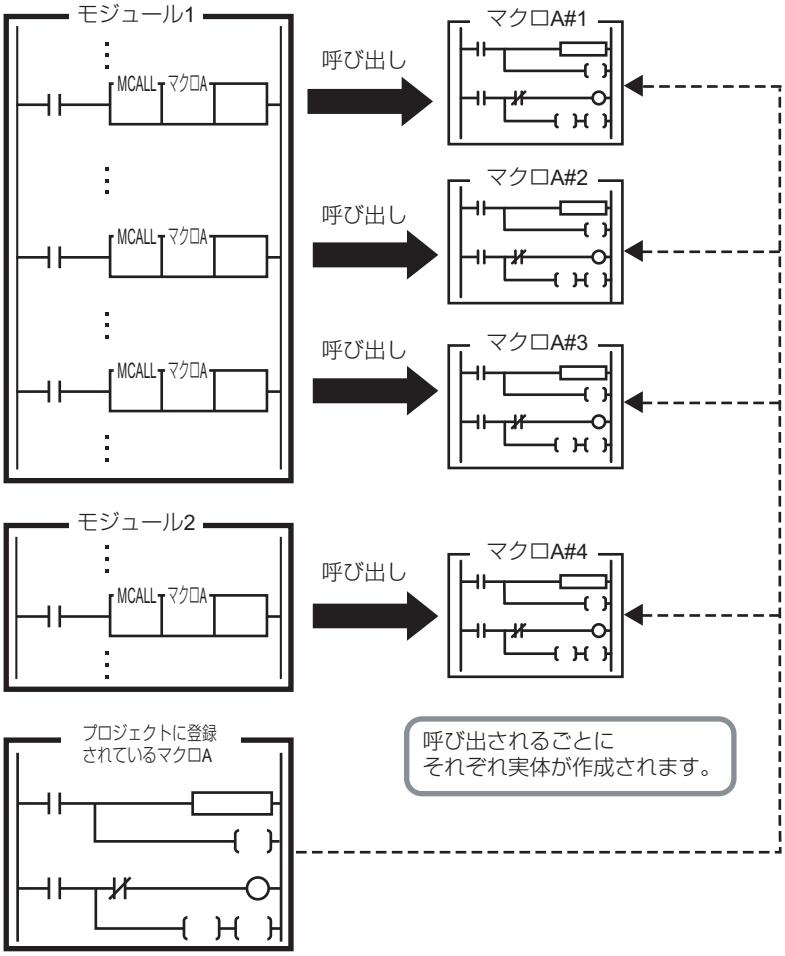
```
LD R002
MEND
```

マクロとサブルーチンプログラムの違い

マクロはサブルーチンプログラムと同様に、呼び出し命令によって動作します。マクロの動作をサブルーチンの動作と比較して説明します。
サブルーチンプログラムについての詳細は、「サブルーチンプログラム」(4-94ページ)を参照してください。

● 実体の作成

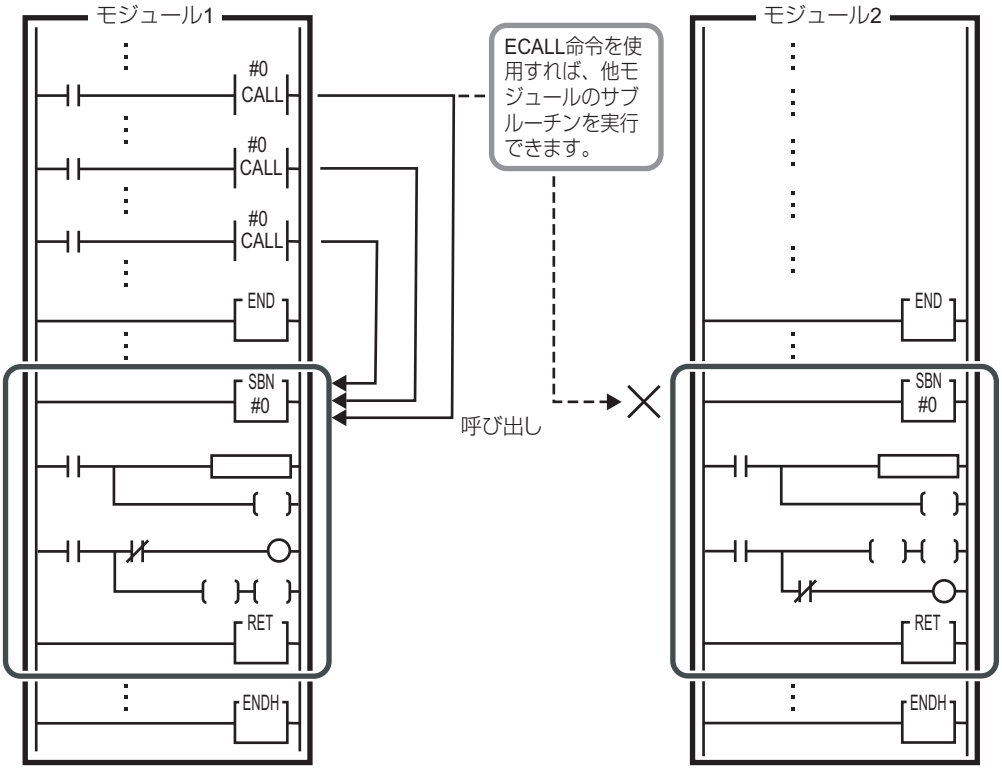
【マクロ】
モジュール内でマクロ呼び出し命令(MCALL、MSTRT)を記述するごとに、同じ数の実体が作成されます。



参 考

- プロジェクトに登録されていて一度も呼び出されないマクロ(未使用マクロ)は、CPU ユニットへ転送されません。未使用マクロは、プロジェクト照合時やPLCとの照合時に照合する対象に含まれません。
- 作成したマクロの実体を区別するために、マクロ名の後ろに「#(管理番号)」を付けて管理します。《KV STUDIO》では、モニタ・シミュレータ時のみ表示されます。

【サブルーチン】
プログラム内のEND命令とENDH命令間に作成し、そのプログラムのメインルーチンプログラム内に記述されたサブルーチンコール命令（CALL）によって実行されます。呼び出し命令ごとに実体が作成されることはありません。



● 呼び出せる箇所

【マクロ】
マクロ呼び出し命令は、モジュール内であれば、メインルーチン／サブルーチン／割り込みルーチンのどこにでも記述できます。マクロ内にはマクロ呼び出し命令を記述できません。

【サブルーチン】
サブルーチンコール命令は、サブルーチンを作成したプログラム内のみ有効です。他のモジュール内のサブルーチンを実行する場合は、モジュール間サブルーチンコール命令を使用します。

● 名称

【マクロ】
自由に名前を決められます。
「使用できない文字一覧」（付-40ページ）

【サブルーチン】
番号で管理します。

● 引数

【マクロ】
引数を持てます。

【サブルーチン】
引数を持てません。

● 実行条件OFF時の動作

【マクロ】
「実行停止時にOFF処理する」にチェックしていない場合、マクロ内のプログラムは実行されません。記述された各デバイスは、最後に実行された時の状態を保持します。
「実行停止時にOFF処理する」にチェックしている場合、停止時に1スキャンのみマクロ内のプログラムの全ての実行条件をOFFとして実行します。例えば、OUT命令で記述されたデバイスは自動的にOFFになります。

「マクロの種類」(4-181ページ)

ポイント **KV-8000Aの場合は設定は表示されず、常に「実行停止時にOFF処理する」にチェックしている場合と同じ動作になります。**

【サブルーチン】
サブルーチン内のプログラムは実行されません。記述された各デバイスは、最後に実行された時の状態を保持します。

● 微分実行型命令の初期状態

【マクロ】
微分実行型命令などで微分を検出するための前回状態は、実行開始時立ち上がり微分の場合がONに、立ち下がり微分の場合がOFFになります。実行開始時に直ちに微分を検出することはありません。

【サブルーチン】
初回の実行開始時のみマクロと同様です。2回目からの実行開始時は、前回実行時の状態になります。

● MC(マスターコントロール) ～ MCR(マスターコントロールリセット) 命令間で呼び出す場合の動作

・MC命令の実行条件がONの時
マクロ、サブルーチンとも通常通りの動作です。

・MC命令の実行条件がOFFの時
【マクロ】
上記の「実行条件OFF時の動作」と同様に停止します。(自己保持型マクロも停止します。)

【サブルーチン】
実行されません。

● 使用する場面

呼び出しているそれぞれの箇所で使用しているデバイスの状態や値を保持したい場合

【マクロ】･･･○

呼び出し命令ごとに別々に実体を作成するため、ローカルデバイスを併用すれば、デバイスの衝突などを意識せずに前回実行時の状態を保持できます。

【サブルーチン】･･･△

同じプログラムを実行することになるため、間接指定などを使用してデバイスの状態や値を退避･復帰させるプログラムを組む必要があります。

一通りの動作が完了するまで実行させたい場合

【マクロ】･･･○

『自己保持型マクロ』を使用すれば、実行開始後はマクロ実行停止命令（MEND）が実行されるまで、実行条件の状態にかかわらず動作し続けます。

また、このような処理を同時に複数実行したい場合も、デバイスの衝突やそれぞれの動作状態を意識せずに使用できます。

【サブルーチン】･･･△

動作完了時まで実行条件がOFFしないようにプログラムする必要があります。

また、このような処理を同時に複数実行したい場合は、デバイスの衝突やそれぞれの動作状態を意識してプログラムする必要があります。

ステップ数を削減したい場合

【マクロ】･･･△

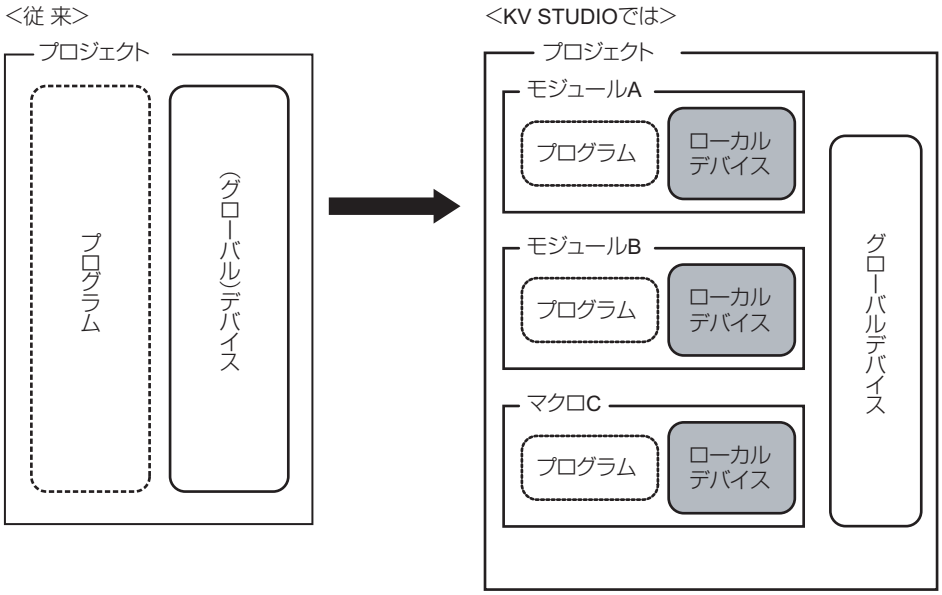
呼び出し命令ごとに実体を作成するため、ステップ数を削減できません。

【サブルーチン】･･･○

何度呼び出しても実体を作成しないため、呼び出し回数が多いほどステップ数の削減につながります。

4-12 ローカルデバイス

従来、PLCで扱うデバイスはプロジェクト全体で共通の存在でした。そのため、プログラム中に使用する全てのデバイスをユーザ側で管理する必要がありました。
《KV STUDIO》では、プログラムごとに独立したデバイスとして使用可能なローカルデバイスを扱えます。別のプログラムで同じ番号のローカルデバイスを使用しても、別のデバイスとして扱われるため、モジュール内やマクロ内のみで使用するデバイスの割り付けが簡単に管理できます。
各プログラム内のみで有効なローカルデバイスに対して、プロジェクト全体で共通して扱うデバイスをグローバルデバイスと呼びます。



ローカルデバイスの記述方法

ローカルデバイスの指定は、デバイス名の先頭に「@」を付けて記述します。

例

<グローバルデバイス>	<ローカルデバイス>
EM0	**@EM0**
MR0	**@MR0**

ポイント

タイマ・カウンタ命令のオペランドとしてデバイス番号を指定する場合のみ、「@+(デバイス番号)」と記述します。
(ニモニック入力例：　TMR @0 #100)

ローカルデバイス一覧

プロジェクト作成時に確保されるローカルデバイス/グローバルデバイスの個数･範囲の初期値を下記に示します。ローカルデバイスの予約個数は、[ワークスペース]上の[CPUシステム設定]→[ローカル全体割付設定]で変更できます。

デバイス名※1	ローカルデバイス	グローバルデバイス	合計個数
	プロジェクト全体での予約個数※2	個数･範囲	
リレー(R)※3	なし	32000 点(2000ch) R000 ～ R199915	32000 点 (2000ch)
内部補助リレー(MR)※3	32000 点(2000ch)	32000 点(2000ch) MR000 ～ MR199915	64000 点 (4000ch)
ラッチリレー(LR)※3	8000 点(500ch)	8000 点(500ch) LR000 ～ LR49915	16000 点 (1000ch)
タイマ(T)※4	1500 点	2500 点 T0 ～ T2499	4000 点
カウンタ(C)※4	1500 点	2500 点 C0 ～ C2499	4000 点
データメモリ(DM)	25000 点	40535 点 DM0 ～ DM40534	65535 点
拡張データメモリ(EM)	45000 点	20535 点 EM0 ～ EM20534	65535 点
テンポラリデータメモリ(TM)	なし	512 点 TM0 ～ TM511	512 点

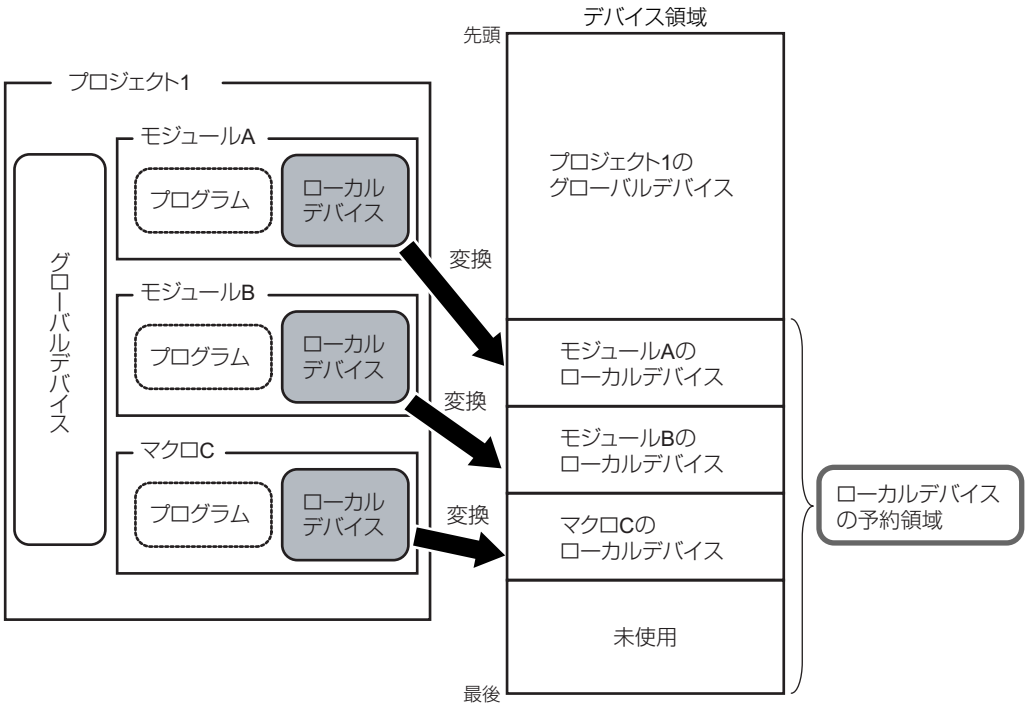
※1 リンクリレー(B)、リンクレジスタ(W)、ファイルレジスタ(ZF)、コントロールリレー(CR)、コントロールメモリ(CM)、インデックスレジスタ(Z)は、ローカルデバイスとして使用できません。
@FMxxxを使用した場合、ワーク領域をローカルデバイスとして消費します。
※2 プロジェクト新規作成時に、ローカルデバイスとして使用可能なデバイスの領域を一括で確保します。
(モジュールを10個使用して、マクロを50回呼び出すと想定して計算しています。)各デバイスの予約個数は任意に変更できます。
※3 チャンネル(ch)単位で使用個数を設定します。
※4 タイマ･カウンタ命令のオペランドとしてデバイス番号を指定する場合のみ、「@＋(デバイス番号)」と記述します。(ニモニック入力例： TMR @0 #100)。

ポイント

- **リレー(R)とテンポラリデータメモリ(TM)をローカルデバイスとして使用する場合は、使用個数を設定する必要があります。**
- **以下のデバイスは、グローバルデバイスとして扱う必要があります。**
  - **CPUユニットや拡張ユニットに割り当てられているデバイス**
  - **タッチパネルなどの外部接続機器で参照するデバイス**
  - **プロジェクト全体で共通して使用するデバイス**
  - **モジュール間でデータの受け渡しが必要なデバイス**
- **ローカルデバイスは、0番を先頭にできるだけ詰めて使用してください。**
- **RやDMはユニットに割り当てられるため、グローバルデバイスとして使用する領域が必要です。ローカルデバイスは、MRやEMを優先して使用してください。**
- **使用可能範囲外でローカルデバイス/グローバルデバイスを使用すると、変換エラーになります。**
- **オペランドに先頭デバイスを指定する各命令語、インデックス修飾、間接指定を使用する場合は、ローカルデバイス/グローバルデバイスの使用可能範囲を超えないようにしてください。**

ローカルデバイスの割付

ローカルデバイスは変換時、デバイスの種類ごとにあらかじめ用意されている予約領域に自動的に割り付けられます。



ポイント

ファンクションブロック、マクロの場合、プロジェクト内で呼び出す回数分のローカルデバイスを割り付けます。
ファンクションブロックの種類がファンクションの場合、実体は常に1つになるため、複数回呼び出しても、1つ分のみのローカルデバイスを割り付けます。

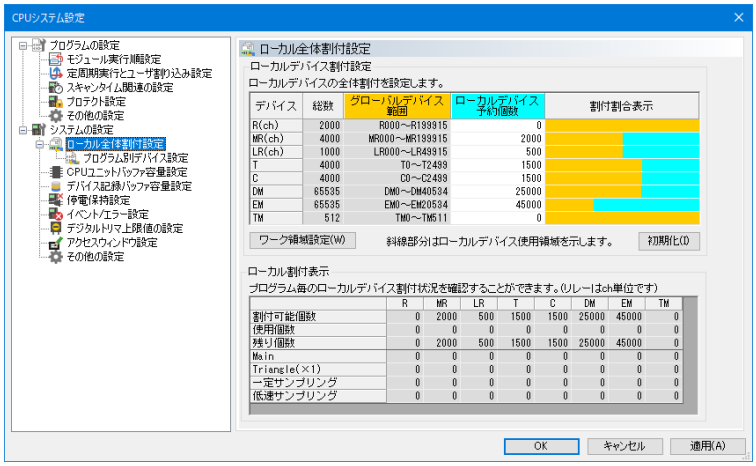
参 考

ローカルデバイスには、グローバルデバイスと同様にデバイスコメントを使用できます。

■ 全体割付設定

プロジェクト内で使用するローカルデバイスの予約個数はあらかじめ設定されていますが、デバイスの種別ごとに変更可能です。また、変更した設定は《KV STUDIO》に保存されるため、プロジェクトを新規作成するたびに設定する必要はありません。

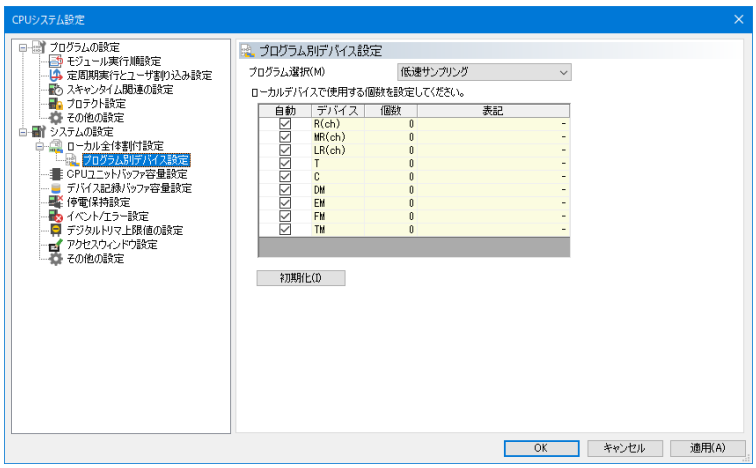
[ワークスペース]上の[CPUシステム設定]を開いて、[ローカル全体割付設定]を選択します。



■ プログラム別デバイス設定

プログラム別に使用するローカルデバイスの予約個数を、デバイスの種別ごとに設定できます。デフォルトは[自動]に設定されており、プログラム内で使用したローカルデバイスの個数が自動的に設定されますので、通常は意識する必要はありません。

[ワークスペース]上の[CPUシステム設定]を開いて、[プログラム別デバイス設定]を選択します。



ローカルデバイス入力・変換時のエラー

● ラダー入力時

ラダー入力時にエラーになる代表例を以下に示します。

ローカルデバイスとして使用できない種類のデバイスを使用している場合

B、W、ZF、CR、CM、Zはローカルデバイスとして使用できません。

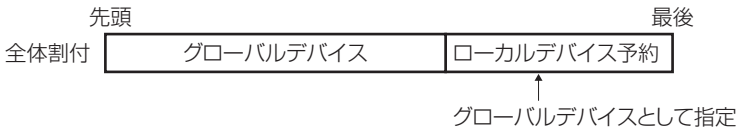
「ローカルデバイス一覧」(4-201ページ)

● ラダー変換時

ラダー変換時にエラーになる代表例を以下に示します。

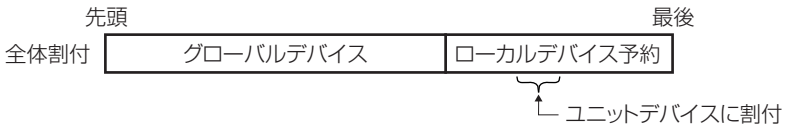
ローカルデバイスに予約されているデバイス領域をグローバルデバイスとして使用している場合

- [ローカルデバイスの全体割付設定]ダイアログの[グローバルデバイス範囲]を確認してください。



「全体割付設定」(4-203ページ)

- 《ユニットエディタ》で設定しているユニットデバイスがローカルデバイスの領域を使用している可能性があるため、確認してください。



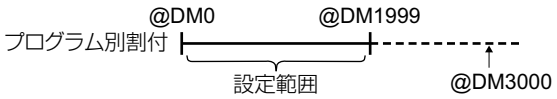
『KV STUDIO ユーザーズマニュアル』「ユニットエディタ」

ローカルデバイスに割り当てられた範囲外のローカルデバイスを使用している場合

自動設定にせず、手動で設定している場合のみエラーになる可能性があります。

[プログラム別ローカルデバイス設定]ダイアログの表記を確認してください。

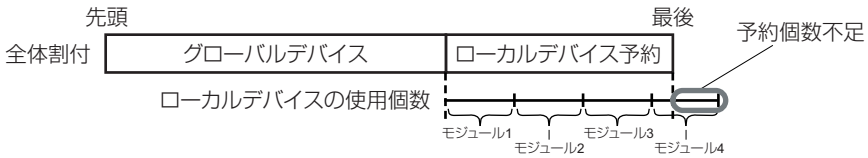
例



「プログラム別デバイス設定」(4-203ページ)

ローカルデバイスの予約個数が足りていない場合

[ローカルデバイスの全体割付設定]ダイアログの[ローカルデバイス割付状況表示]で、使用個数などが赤色表記されているデバイスを確認してください。該当するデバイスの予約個数を増やすか、各プログラム別に設定されているローカルデバイスの個数を減らして、使用可能個数(ローカル予約個数)が使用個数より多くなるように設定してください。

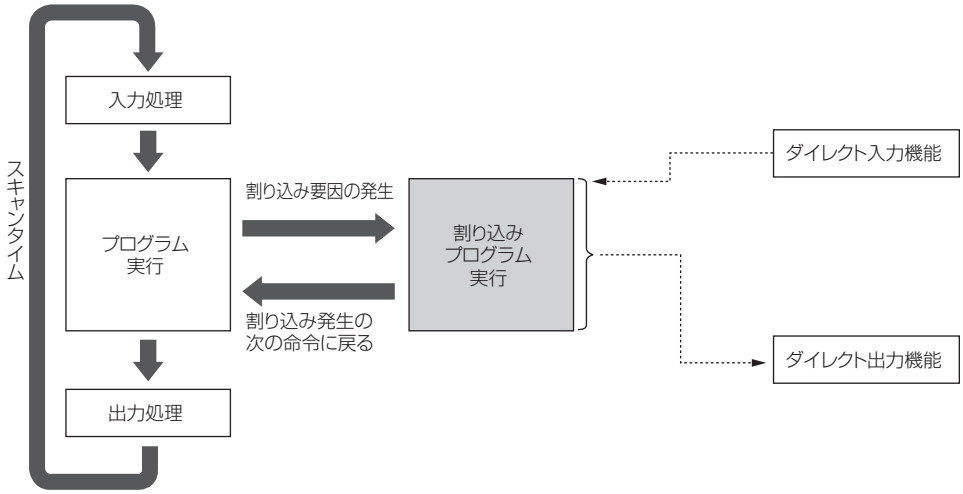


「全体割付設定」(4-203ページ)

4-13 割り込み

概要

通常PLCは、入力処理→プログラムの実行→出力処理を繰り返すことで、ラダーシーケンスが進んでいきます。したがって、スキャンタイムより短い信号は入力できません。
割り込み処理機能を使えば、スキャンタイムに依存せずに、割り込みが発生した時点で処理ができます。
割り込みが発生すると、プログラム実行中でもその場で一旦処理を中断し、割り込み要因に応じた割り込みプログラムを実行します。割り込みプログラムが終了すると、中断していたプログラムの処理を再開します。



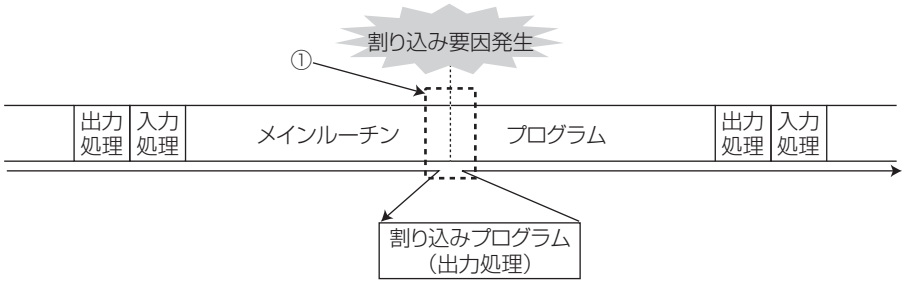
■ 割り込み要因の種類

プログラムにより設定できる割り込み要因には、拡張ユニットからの割り込み（64個まで）と定周期モジュール（4個まで）、ユニット間同期モジュールがあります。

• 「定周期モジュール」（4-103ページ）

■ デバイスの処理について

割り込み時のデバイスの処理は以下のとおりです。
入出力処理の詳細については、「割り込み時の入出力処理」(4-212ページ)を参照してください。



①割り込みプログラムの開始時/終了時の処理

割り込みプログラムの開始時に、内部レジスタ、演算フラグ、TM0～TM3、Z11～Z23の値をシステム内に記憶し、割り込みプログラム終了時に記憶した値に戻します。
割り込み中にこれらの値を変更してもメインルーチンプログラムには影響しません。変更した値は、割り込みプログラム中のみ有効になります。
その他のデバイスは、割り込みプログラム中に変更した値が、割り込みプログラム終了後、メインルーチンプログラムに反映されます。

ポイント **インデックスレジスタZ1～Z10は、CPUシステム設定の「ユーザ割り込み設定」で退避/復帰処理をするように設定できます。**

参考 自動的にリフレッシュされないデバイス値を最新の状態に更新するには、以下のように処理してください。
- 「DR」で入出力リレーを記述します。
- 「RFSX/RFSY」命令を記述します。

「ダイレクト処理について」(4-83ページ)

■ 処理時間

以下の条件をすべて満たす場合、割り込みプログラム開始までの時間は15μs以内(TYP.4μs)になります。
- 〔CPUシステム設定〕の〔ユーザ割り込み設定〕で、「命令実行中に割り込み許可する」をチェックしている場合
- その割り込みプログラムより優先度の高い割り込みプログラムを使用していない
- KV-5000/3000シリーズ用拡張ユニットを使用していない(使用した場合は最大+14μs)
- ユニット間同期の入力リフレッシュ、もしくは出力リフレッシュにて大量のデバイスをリフレッシュ対象にしていない
- ロギング/トレース設定で、1つのIDで513ワード以上のデバイスを登録していない

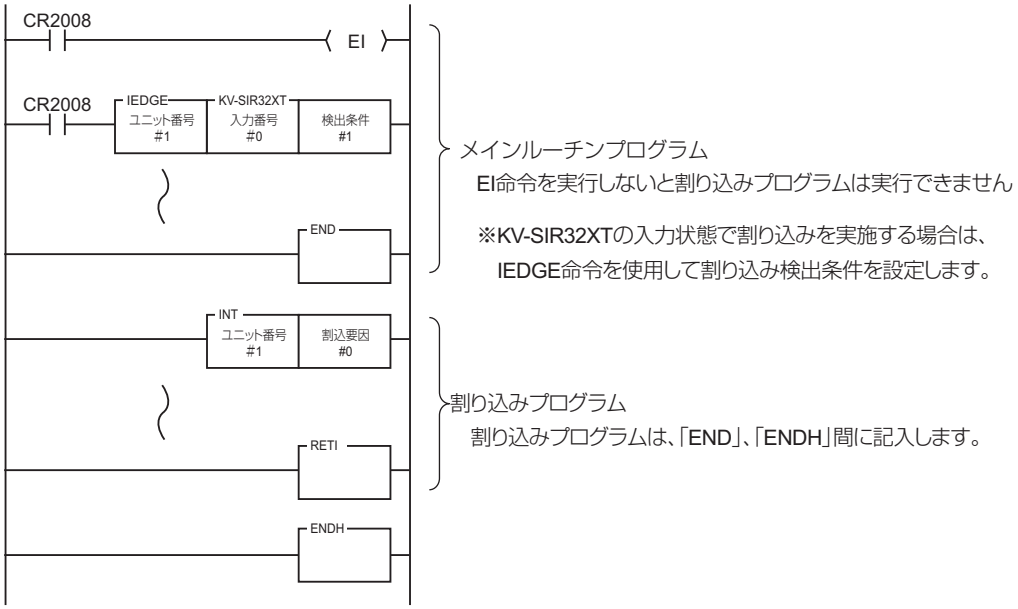
なお、割り込みプログラムの起動/終了処理(各デバイスのリフレッシュ/退避/復帰)時間は合計2.5μsです。

参考 割込み開始までの処理時間は上記のとおりですが、割り込みプログラムの先頭でKV-5000/3000シリーズ用拡張ユニットに対してダイレクト処理命令を実行した場合、処理完了までの時間がかかることがあります。
- KL-N20V, KV-CL20, KV-DN20, KV-H20, KV-H20S, KV-H40S使用時:115μs
- 上記以外のKV-5000/3000シリーズ用の拡張ユニット使用時:55μs

割り込みプログラムの実行方法

以下のようにプログラムを記述することで、割り込み処理が可能になります。

割り込みプログラムの例



■ 割り込み許可(EI)命令

運転開始時は、割り込みが禁止状態になっています。
割り込みプログラムを実行するには、割り込み許可(EI)命令を実行する必要があります。
メインルーチンプログラム内に、EI命令を記述してください。

参 考

- 割り込みを禁止する場合は、割り込み禁止(DI/DIC)命令を使用します。
- 割り込み禁止時に割り込みが発生しても、次に割り込みを許可されるまで処理を保留します。
- 割り込みプログラムの開始時は、割り込み許可状態です。
- 割り込み許可(EI)命令を実行していない場合でも周辺処理実行によるシステム割り込みは実行されます。

「EI命令」4-218ページ
「DI命令」4-218ページ、「DIC命令」4-220ページ

■ INT/RETI命令

END/ENDH命令の間に、割り込み要因をオペランドにしてINT命令を記述します。
INT命令の後に割り込み処理内容をプログラムして、最後にRETI命令を記述します。

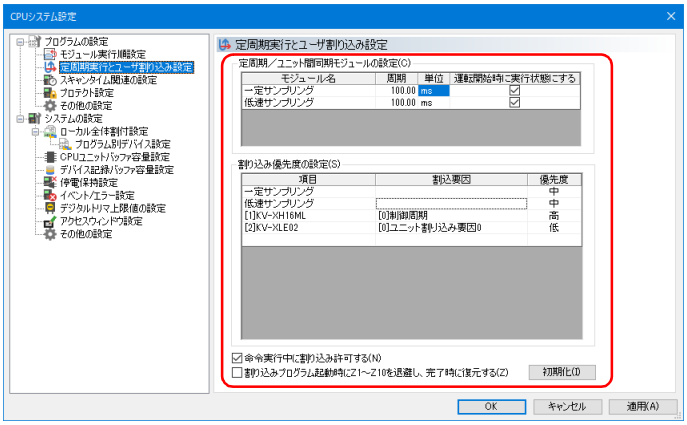
ポイント

- **1つの割り込み要因に対してINT命令は1回のみ記述できます。**
- **割り込みプログラム内では、使用できない命令語があります。**
  「割り込みプログラムで使用できない命令語」(4-210ページ)を参照してください。
- **KV-SIR32XTを使用して割り込み実行時に、割り込み応答を高速にする場合は、ユニットエディタの設定で入力時定数を小さく設定してください(初期値:10μs)。**

「INT/RETI命令」(4-222ページ)

割り込みの優先度

複数の割り込み要因が同時に発生した場合の優先度を設定できます。
《KV STUDIO》のワークスペースから[CPUシステム設定]を開いて、[プログラムの設定]の[定周期実行とユーザ割り込み設定]で設定します。



優先度は「高/中/低」で設定できます。割り込みプログラム実行中は、自身より高い優先度の割り込みは受け付けますが、同じか低い優先度の割り込みは実行中の割り込みプログラムが終了するまで保留されます。
また、「命令実行中に割り込み許可する」にチェックを付けると、処理時間の長い命令語の実行途中に割り込めるため、割り込み応答時間のばらつきを抑制できます。
割り込みプログラム実行中に別の割り込みが発生した場合、優先度の設定により動作が異なります。以下の表を参考にしてください。

実行中割り込みプログラム(A)の優先度	発生した割り込みプログラム(B)の優先度		
	高	中	低
高	保留	保留	保留
中	即実行	保留	保留
低	即実行	即実行	保留

保留　：現在実行中の割り込みプログラム(A)終了後に、発生した割り込みプログラム(B)を実行します。
即実行：現在実行中の割り込みプログラム(A)を中断し、発生した割り込みプログラム(B)を実行します。
　　割り込みプログラム(B)の実行が終了した後、割り込みプログラム(A)が再開します。

割り込みプログラムで使用できない命令語

命令の種類		命令語	
共通		@XXXX命令	微分実行型
基本命令	接点命令	LDP/LDF	ロードパルス／ロードパルフ
		ANP/ANF	アンドパルス／アンドパルフ
		ORP/ORF	オアパルス／オアパルフ
		LDPB/LDFB	ロードパルスバー／ロードパルフバー
		ANPB/ANFB	アンドパルスバー／アンドパルフバー
		ORPB/ORFB	オアパルスバー／オアパルフバー
	出力命令	DIFU/DIFD	ディフアップ／ディフダウン
		ONDL/OFDL	オンディレイ／オフディレイ
		SHOT	ワンショット
		FLIK	フリッカ
		ALT	オルタネート
	タイマ･カウンタ命令	TMR/TMH/TMS/TMU	タイマ／高速タイマ／高速1msタイマ／高速10μsタイマ
		C	カウンタ
		OUTC	アウトカウンタ
		ITVL	インターバルタイマ
		UDC	アップダウンカウンタ
		UDT	アップダウンタイマ
	接続/終了命令	MEP/MEF	メップ／メフ
応用命令	シフト命令	SFT	シフト
	ステージ処理命令	STG/JMP/ENDS	ステージ／ジャンプ／エンドステージ
		W-ON/W-OFF	ウェイトオン／ウェイトオフ
		W-UE/W-DE	ウェイトアップエッジ／ウェイトダウンエッジ
	フロー命令	SBN/RET/SCJ	サブルーチン開始/サブルーチン終了/LABEL命令までジャンプ
	ファイルレジスタ命令	FRSTM/FRLDM	ファイルレジスター括保存／ファイルレジスター括読出
演算命令	データ制御命令	RAMP/TPOUT/LLFLT	傾斜信号／時分割比例出力／進み遅れフィルタ
	文字処理命令	RCOM	文字列読み出し

命令の種類		命令語	
拡張命令	データ処理命令	HKEY/SORTN	16キー入力 ／バイナリデータ分割ソート
	カムスイッチ命令	ABSENC	アブソリュートエンコーダ
	PID命令	PID/PIDAT	PID／オートチューニング付きPID
	ストレージ命令	MWRIT/MREAD/MFREE/MMKDIR/MPRINT/MREADL/MCOPY/MMOV/MREN/MFREEK/MSTAT	ストレージライト／ストレージリード／ストレージ容量取得／ストレージディレクトリ作成／ストレージ文字列書込み／ストレージ1行読み出し／ストレージファイルコピー／ストレージファイル移動／ストレージファイル名変更／ストレージ空き容量取得／ストレージファイル状態取得
		MRMDIR/MDEL	ストレージフォルダ削除／ストレージファイル削除
	アクセスウィンドウ命令	AWSHOW/AWHIDE	ユーザメッセージ表示/ユーザメッセージ解除
	センサ設定命令	SPWR/SPRD/SSVC	センサパラメータ書込／センサパラメータ読出／センササービス実行

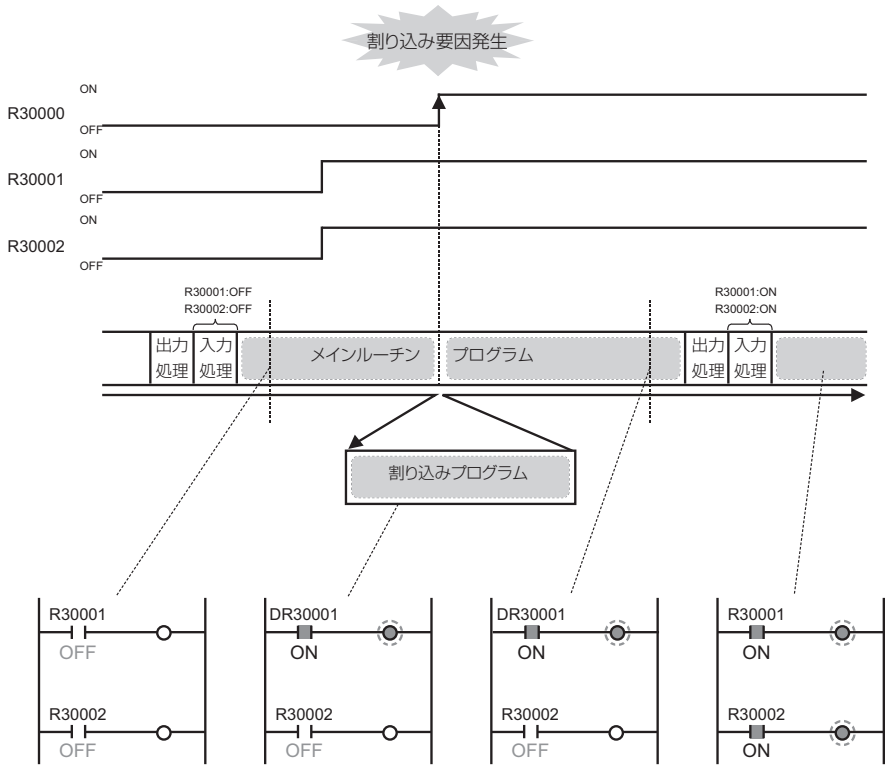
※ 定周期モジュール/ユニット間同期モジュール内では使用可能です。

割り込み時の入出力処理

■ 入力処理

割り込みプログラム開始時、そのスキャンの入力処理(I/Oリフレッシュ)で取得した状態です。
割り込みプログラム内で、その時点の入力リレーの状態を取得するため、DRでリレー番号を指定する、またはダイレクトリフレッシュ命令(RFSX命令)を使用してリフレッシュを行う必要があります。

例

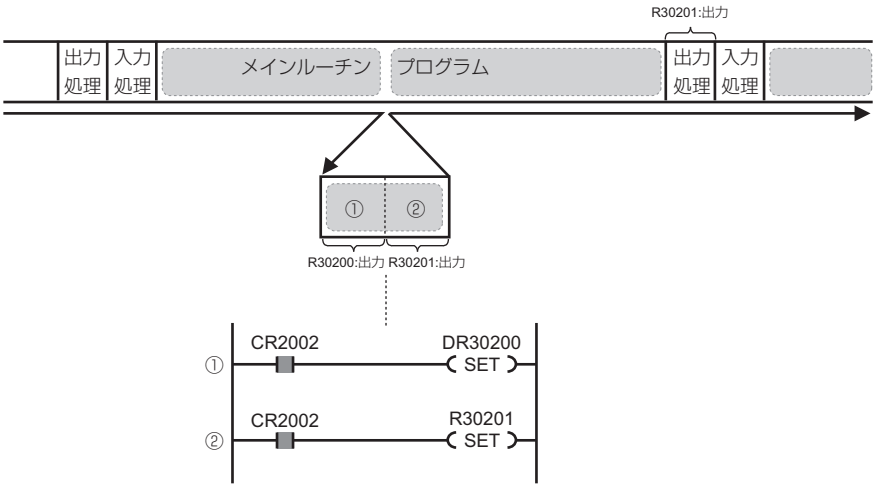


上記のような割り込みプログラムが実行されると、
①メインルーチンプログラムの入力処理 → R30001 : OFF　R30002 : OFF
②割り込みプログラムの入力処理 → R30001 : ON　R30002 : OFF
になります。
入力リレーDR30001はダイレクト入力なので、割り込みプログラム実行時の状態が取り込めます。
入力リレーR30002は割り込みプログラム中でも、メインルーチンプログラムの入力処理の状態で扱われます。
メインルーチンプログラムでは割り込みプログラム内でダイレクトリフレッシュをしているので、割り込みプログラム内の状態(ON)になります。

■ 出力処理

割り込みプログラムには出力処理がありません。割り込みプログラム終了後、メインルーチンプログラムの出力処理（自動リフレッシュ）で外部に出力します。
ただし、出力リレーをDRと記述すると（DR30200など）、プログラムが実行されると、直ちに外部に出力されます。（ダイレクト出力）

例



上記のような割り込みプログラムが実行されると、

- ①行目で、出力リレーDR30200はダイレクト出力リレーなので、直ちに外部に出力します。
- ②行目で、出力リレーR30201は内部処理ではONしますが、ダイレクト出力リレーではないのでメインルーチンプログラムの出力処理で外部に出力されます。

例

割り込みプログラム内でダイレクト出力したい場合は、出力リレーにDRを使用してください。

参考

自動的にリフレッシュされないデバイス値を最新の状態に更新するには、以下のように処理してください。

- 「DR」で入出力リレーを記述します。
- 「RFSX/RFSY」命令を記述します。
- 「UREAD/UWRIT」命令などダイレクトリフレッシュ命令を記述します。
- ユニット専用命令を記述します。

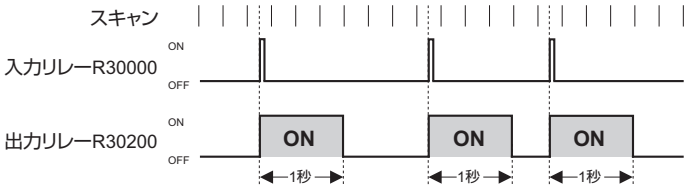
「ダイレクト処理について」（4-83ページ）

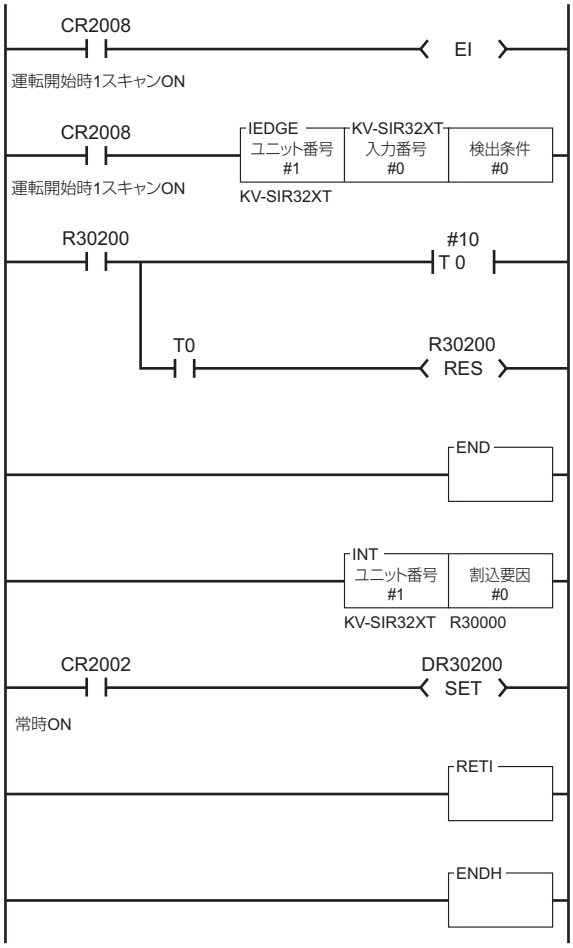
サンプルプログラム

■ 信号変換器を使用した割り込みの応用例

割り込み処理を使用した微小パルスを一定の幅のパルスに変換して出力します。

- 入力リレーR30000で入力した微小パルスを、出力リレーR30200からON幅1秒のパルスに変換します。
- 入力リレーR30000の立ち上がりから1秒間、出力リレーR30200がONします。





電源投入時に
割り込みを許可します。

割り込みの検出条件を
立ち上がりに設定します。
※入力リレーR30000の入力
時定数はユニットエディタで
設定してください。
（初期値:10μs）

出力リレーR30200を
1秒間だけONします。

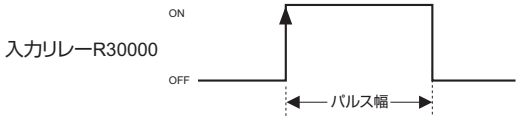
入力リレーR30000が
ONすると出力R30200を
ONします。

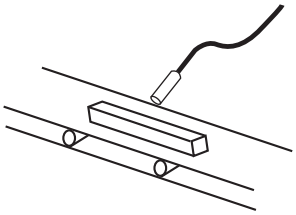
〈ニモニックリスト〉

```
LD CR2008
EI
LD CR2008
IEDGE #1 #0 #0
LD R30200
MPS
TMR #0 #10
MPP
AND T0
RES R30200
END
INT #1 #0
LD CR2002
SET DR30200
RETI
ENDH
```

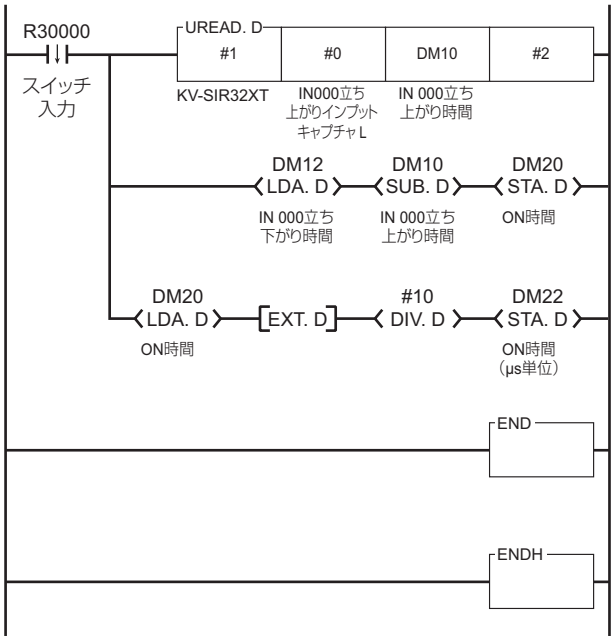
■ 高速パルスON幅の測定の応用例

入力リレーに入る入力パルスのパルス幅を測定します。





- インプットキャプチャ機能を使って、0.1μs 単位でパルス幅を測定します。
- 入力リレーR30000の立ち下がり時にUREAD命令で、R30000（IN000）の立ち上がりインプットキャプチャ値（バッファメモリ#0･#1）をデータメモリ[DM10･DM11]、立ち下がりインプットキャプチャ値（バッファメモリ#2･#3）をデータメモリ[DM12･DM13]に格納します。
- [DM12･DM13]から[DM10･DM11]の値を減算します。
- 測定された値をデータメモリ[DM20･DM21]に格納します。
- [DM20･DM21]の値を1μs単位に換算して[DM22･DM23]に格納します。



R30000立ち下がり時にインプットキャプチャの結果を読み出します。

立ち上がりから立ち下がりまでの時間を求めます。

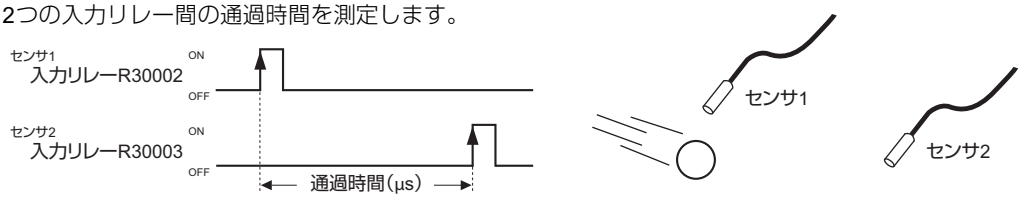
測定した時間をμs単位に換算します。

〈ニモニックリスト〉

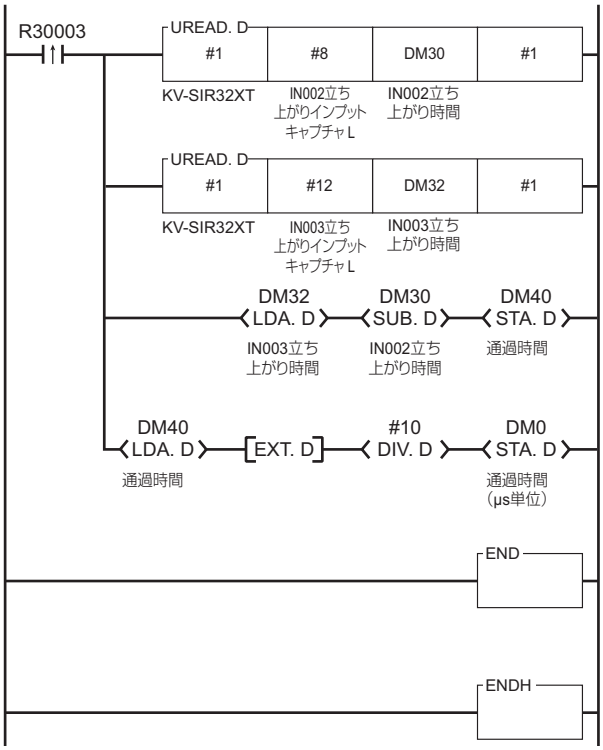
```
LDF R30000
UREAD.D #1 #0 DM10 #2
MPS
LDA.D DM12
CON
SUB.D DM10
CON
STA.D DM20
MPP
LDA.D DM20
CON
EXT.D
CON
DIV.D #10
CON
STA.D DM22
END
ENDH
```

■ 2点間通過時間の測定の応用例

2つの入力リレー間の通過時間を測定します。



- 入力リレーR30002から入力リレーR30003までの通過時間を測定します。
- 測定された値は、0.1μs単位でデータメモリ[DM0･DM1]に格納します。



R30003立ち上がり時にインプットキャプチャの結果を読み出します。

IN002の立ち上がりからIN003の立ち上がりまでの時間を求めます。

測定した時間をμs単位に換算します。

〈ニモニックリスト〉

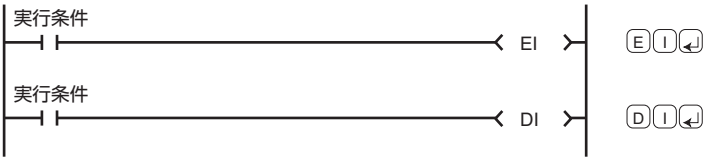
```
LDP R30003
UREAD.D #1 #8 DM30 #1
UREAD.D #1 #12 DM32 #1
MPS
LDA.D DM32
CON
SUB.D DM30
CON
STA.D DM40
MPP
LDA.D DM40
CON
EXT.D
CON
DIV.D #10
CON
STA.D DM0
END
ENDH
```

割り込み処理用命令

命　令	ニモニック	動作概要	ページ
割り込み禁止	DI	割り込みプログラムの実行を禁止します。	4-218
割り込み許可	EI	割り込みプログラムの実行を許可します。	4-218
割り込み禁止範囲	DIC	割り込み禁止範囲を設定します。	4-220
イント	INT	RETI命令までの割り込みプログラムを実行します。	4-222
リターンイント	RETI	割り込みプログラムの最後を示します。	4-222

DI	DI	割り込み禁止	割り込みプログラムの実行を禁止します。
EI	EI	割り込み許可	割り込みプログラムの実行を許可します。

ラダープログラム　　入力方法



動作説明

DI　実行条件がONすると、割り込みプログラムと定周期モジュールの実行を禁止します。
一時的に割り込みを禁止したいときに使用します。

EI　実行条件がONすると、割り込みプログラムと定周期モジュールの実行を許可します。
DI命令で禁止した割り込みプログラムを許可（解除）します。

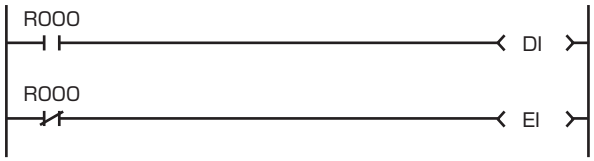
- 定周期モジュールは、KV-8000/8000A/7500/7300/5500//5000/3000のみ使用できます。
- 一度EI命令を実行すると、DI命令を実行するまでは割り込み許可となります。
- 運転開始時は常に割り込み禁止状態になっています。
- DI命令を実行中（割り込み禁止中）に割り込みが発生したときは、16段まで記憶されます。
  EI命令が実行されると記憶した割り込みが実行されます。
- 割り込みプログラムの開始時は、割り込み許可（EI）状態です。割り込みプログラムを実行中にさらに割り込みプログラムを実行するには、実行中の割り込みプログラムよりも優先度を高く設定する必要があります。優先度は、【CPU システム設定】→【定周期実行とユーザ割り込み設定】で設定します。
  最大3段まで受付可能です。

演算フラグ

演算フラグは変化しません。

サンプルプログラム

入力リレーR000がONのとき、割り込みプログラムの実行を禁止します。
R000がOFFのとき、割り込みプログラムの実行を許可します。



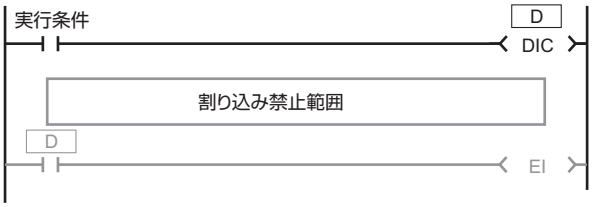
〈ニモニックリスト〉

```
LD R000
DI
LDB R000
EI
```

DIC

DIC　割り込み禁止範囲　割り込み禁止範囲を設定します。

ラダープログラム

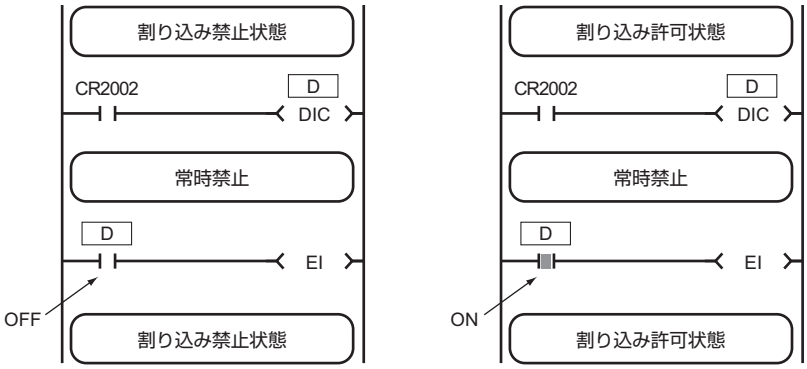


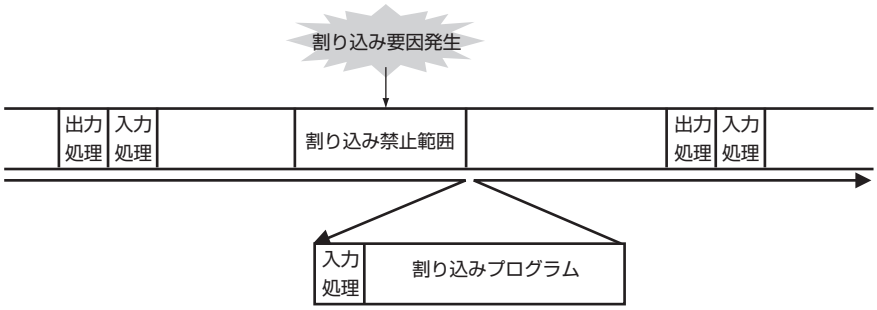
入力方法

オペランド	説明	占有サイズ
D	割り込み許可/禁止状態を出力するビットデータを指定します。	1ビット

動作説明

DIC　EI命令と組み合わせて使用します。
実行条件がONのとき、DIC命令からEI命令までの間、割り込み処理の実行を禁止します。
DIC命令実行前の割り込み許可/禁止状態を D に出力します。
DIC命令実行前の割り込み許可/禁止状態が許可であれば D をセットし、禁止であればリセットします。





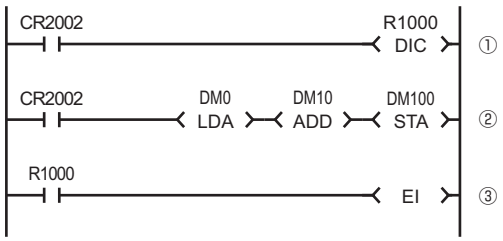
演算フラグ

CR2009	変化なし
CR2010	変化なし
CR2011	変化なし
CR2012	間接指定またはインデックス修飾の範囲が不適切な場合、オペランドの範囲外アクセスが発生した場合はON、それ以外はOFF。 オペランドに間接指定、インデックス修飾を指定していない場合、およびオペランドの範囲外アクセスのチェックが行われない場合は変化しません。

※ CR2012がONした場合、命令は実行されません。
CR2012がONした場合、CM5150～CM5176にエラーの詳細情報が格納されます。
「エラー履歴」(4-86ページ)

サンプルプログラム

演算処理中は、割り込みを禁止にします。DIC命令実行後、割込み処理の状態をDIC命令実行前の状態に戻します。



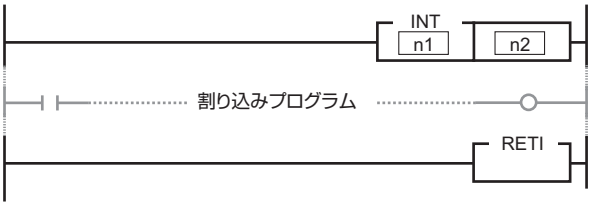
〈ニモニックリスト〉

```
LD CR2002
DIC R1000
LD CR2002
LDA DM0
CON
ADD DM10
CON
STA DM100
LD R1000
EI
```

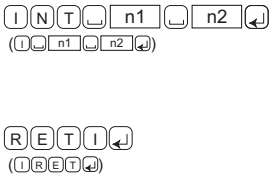
① 割り込み処理禁止
② DM0+DM10→DM100
③ 割り込み処理許可

INT	INT	割り込み実行	オペランドの立ち上がり、または立ち下がりでRETIまでの割り込みプログラムを実行します。
RETI	RETI	割り込み終了	割り込みプログラムの最後を示します。

ラダープログラム



入力方法



オペランド	説明	占有サイズ
n1 ※1	ユニット番号(0～48)を指定します。	16ビット
n2 ※1	ユニットの割り込み要因番号(0～31)を指定します。	16ビット
S ※2	割り込みに使用するビットを指定します。	1ビット

※1 定数及び、定数に設定した変数のみ指定可能です。
※2 定数のみ指定できます。

動作説明

INT 割り込みプログラムの開始を指定します。
RETI 割り込みプログラムの終了を指定します。

ポイント

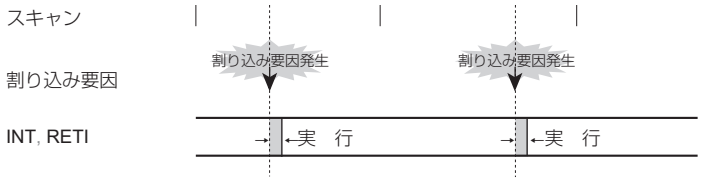
- **内部レジスタ、インデックスレジスタ、演算フラグ、R000～R015、TM0～TM3は、割り込みプログラム開始時に自動的に退避されます。退避したそれぞれの値は割り込みプログラム終了時に復帰します。**
- **KV-8000/7000シリーズの場合、R000～R015は退避しません。**
- **インデックスレジスタはデフォルトでZ11～Z23のみ退避/復帰処理されます。【CPUシステム設定】→【定周期実行とユーザ割り込み設定】でZ1～Z10も退避/復帰処理されるように設定できます。**
- **以下の条件の場合、INT命令で指定した割り込み条件が存在しないため、エラーが発生せず実行されないことがあります。ご注意ください。**
  - **使用できない高速カウンタのCH数を使用している**
  - **使用できない位置決め出力軸を指定している**
  - **使用できない基本入力点数を使用している**
    (上記の場合、KV Nanoシリーズでは、KV STUDIOは変換エラーになりません)
- **以下の条件の場合、INT命令で指定した割り込み条件が存在しないため、警告のみ発生しエラーは発生せず実行されないことがあります。ご注意ください。**
  - **指定したユニットが割り込み機能に対応していない**
  - **存在しない割り込み要因を指定している**

演算フラグ

CR2009	変化なし
CR2010	変化なし
CR2011	変化なし
CR2012	変化なし

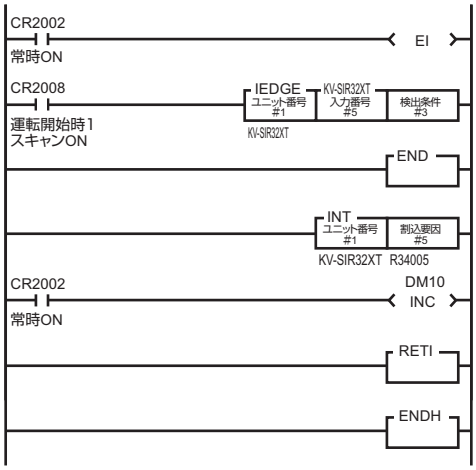
実行するタイミング

INT, RETI　　オペランドの立ち上がりまたは立ち下がりで命令を実行します。



サンプルプログラム

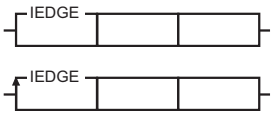
ユニット番号1の入力（IN005）の立ち上がりと立ち下がりの両方で割り込みを実行し、DM10の値をカウントアップします。



〈ニモニックリスト〉

```
LD CR2002
EI
LD CR2008
IEDGE #1 #5 #3
END
INT #1 #5
LD CR2002
INC DM10
RETI
ENDH
```

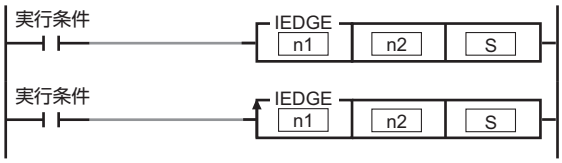
IEDGE
@IEDGE



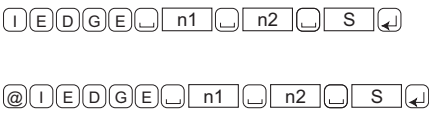
入力割り込み検出条件書き込み

割り込みに使用する検出条件を設定します。

ラダープログラム



入力方法



オペランド	説明	占有サイズ
n1	ユニット番号(0～48)を指定します。	16ビット
n2	入力番号またはそれを格納する先頭を指定します。(0～31)※2	16ビット
S	検出条件またはそれを格納する先頭を指定します。(0～3)※2	16ビット

※1　定数のみ指定可能です。
※2　ビットデータを指定する場合は、チャンネルの先頭を指定してください。

動作説明

IEDGE 実行条件がONの時、[S]で指定した1ワードのデータを[n1]番目のユニットの[n2]番目の要因の割り込みの検出条件に設定します。
[S]の値により、検出条件は以下のようになります。

S	検出条件
0～1	立ち上がり
2	立ち下がり
3	立ち上がり・下がり両方

ポイント
- [S]に4～65535を指定した場合は演算エラーにはなりませんが、検出条件は変更されません。
- 本命令が実行完了したタイミングで、上記検出条件の設定が変更されます。

@IEDGE 実行条件の立ち上がりで、1スキャンだけ実行します。

演算フラグ

CR2009	変化なし
CR2010	変化なし
CR2011	変化なし
CR2012	以下の場合はON、それ以外はOFF。 • 間接指定、インデックス修飾の範囲が不適切な場合。 • [n1]で指定したユニット番号が49以上の場合 • [n2]で指定した入力番号が32以上の場合 • [S]から連続した16ビットの領域を確保できない場合 • [n1]で指定したユニット番号のユニットがKV-SIR32XTではない場合 • オペランドの範囲外アクセスが発生した場合

※ CR2012がONした場合、命令は実行されません。
CR2012がONした場合、CM5150～CM5176にエラーの詳細情報が格納されます。

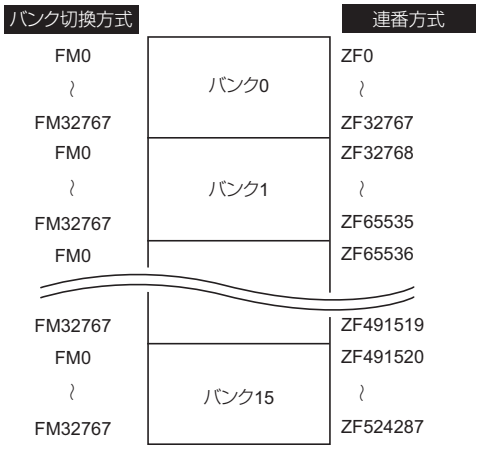
サンプルプログラム

INT命令（4-222ページ）を参照

4-14 ファイルレジスタ

概要

ファイルレジスタは、すべての領域を連番方式で扱う「ZF」と、領域を16個のバンク（0～15）に分割するバンク切換方式で扱う「FM」があります。



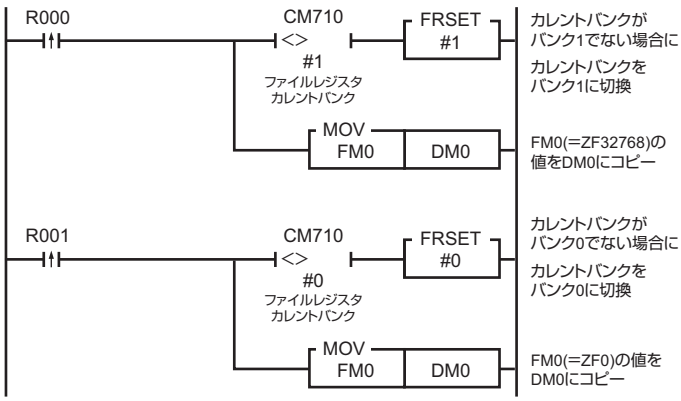
例えば、バンク1のFM0とZF32768は同じデータ格納場所を指します。

デバイス番号の詳細やデータの取り扱い方などは「4-4 デバイス」ファイルレジスタ　ZF/FM（4-39ページ）を参照してください。

バンク切換

バンクを切り換えるには、「FRSET命令」を使用します。
現在指定されているバンクをカレントバンクと呼びます。

例



〈ニモニックリスト〉

```
LDP R000
MPS
AND<> CM710 #1
FRSET #1
MPP
MOV FM0 DM0
LDP R001
MPS
AND<> CM710 #0
FRSET #0
MPP
MOV FM0 DM0
```

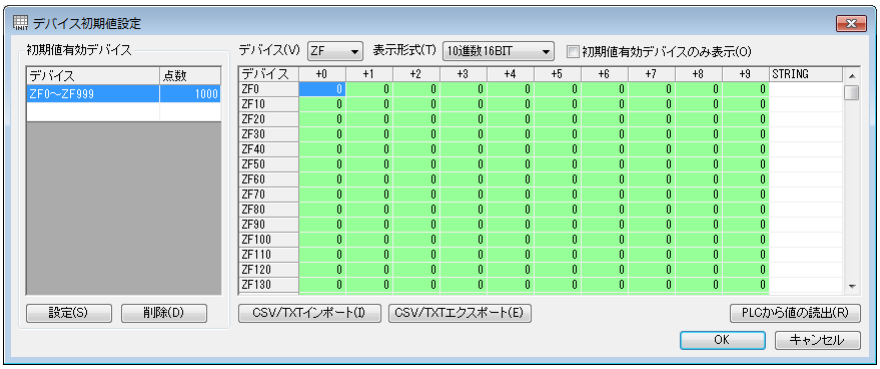
「FRSET命令」4-230ページ

参考

- カレントバンクの番号はCM710で確認できます。
- KVモード（上位リンク）の「BE」コマンドを使用して、バンクを切り換えることができます。
  『KV-XL202/XL402ユーザーズマニュアル』

初期値設定

ファイルレジスタ(連番方式)の値をあらかじめ設定する場合は、《KV STUDIO》のワークスペース上から開く「デバイス初期値設定」ダイアログで設定します。



このダイアログで設定したプロジェクトをCPUユニットに転送すると、PRG→RUNモード切り換え時に設定した初期値を自動的に読み出します。

『KV STUDIOユーザーズマニュアル「デバイス初期値設定」』

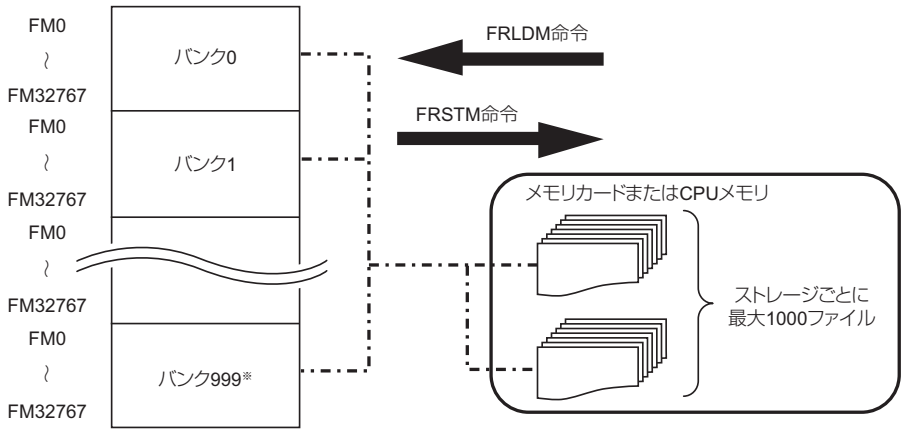
ファイルレジスター括設定

専用命令語を使用して、CPUユニットに挿入したメモリカードや、内蔵のCPUメモリを対象に、ファイルレジスタのデータ(ファイルレジスタ設定値)を保存・読出しができます。

FRSTM命令： 指定したバンクのファイルレジスタをメモリカードまたはCPUメモリに保存

FRLDM命令： メモリカードまたは CPU メモリに保存したファイルから指定したバンクのファイルレジスタに読出

「FRSTM命令」4-232ページ、「FRLDM命令」4-234ページ



※メモリカード/CPUメモリを使用した場合。内蔵のデバイスのみを使用した場合は15までです。

あらかじめ最大1000ファイルまでのファイルレジスタ設定値をメモリカードまたはCPUメモリに用意して、必要な時に読み出せば、デバイス数を気にせずに段取り替え用途で使用できます。

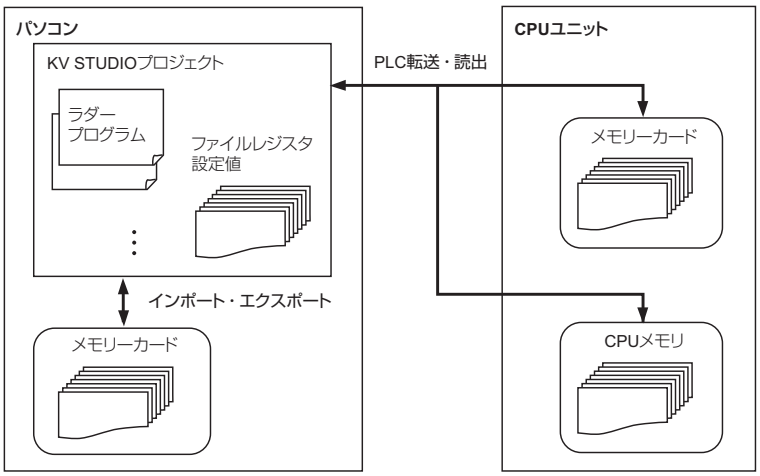
■ ファイルレジスタ設定値の管理

メモリカードまたはCPUメモリ内に保存されているファイル(ファイルレジスタ設定値)は、《KV STUDIO》で編集できます。ワークスペース上から開く「ファイルレジスタ設定」ダイアログで編集、新規作成、CSVファイルのインポートなども可能です。フィルレジスタ設定値は、プロジェクトごとに管理されます。

『KV STUDIOユーザーズマニュアル「ファイルレジスタ設定」』

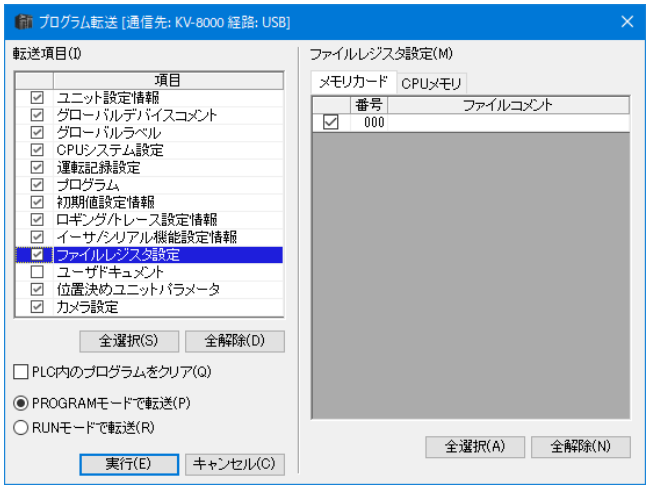
■ ファイルレジスタ設定値の転送

《KV STUDIO》で新規作成･編集したファイルレジスタ設定値をメモリカードまたはCPUメモリに書き込んだり、メモリカードまたはCPUメモリ内のファイルレジスタ設定値ファイルを読み出す方法は以下のとおりです。



● PLC転送･読出

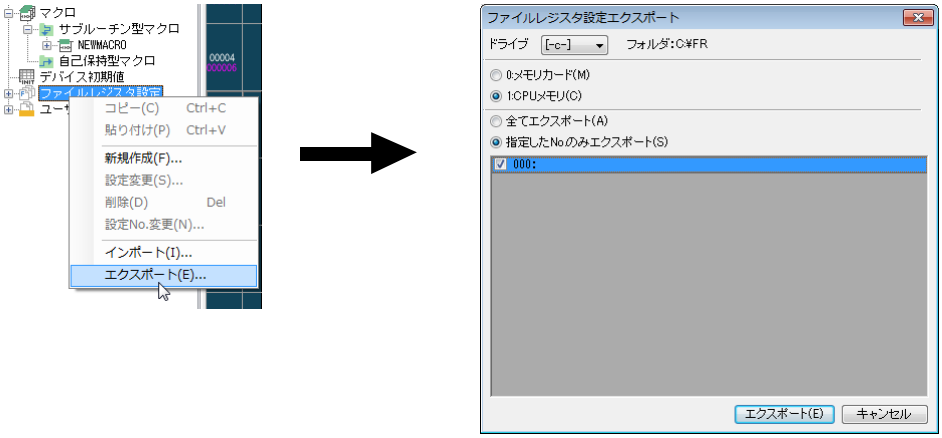
PLC転送･読出時に他のプロジェクト情報と同様に通信します。転送時、他のプロジェクト情報はCPUユニット本体に、ファイルレジスタ設定のみメモリカードまたはCPUメモリに保存されます。



『KV STUDIOユーザーズマニュアル「モニタ」』

● インポート･エクスポート

ストレージ転送ツールを使用して、インポート･エクスポートをおこないます。

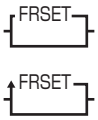


『KV STUDIOユーザーズマニュアル「ファイルレジスタ設定」』

ファイルレジスタ命令

命　令	ニモニック	動作概要	ページ
ファイルレジスタバンク切換	FRSET	ファイルレジスタのカレントバンク番号を確定した番号に切り換えます。	4-230
ファイルレジスター括保存	FRSTM	指定したバンク内のファイルレジスタをバイナリ形式でメモリカードまたはCPUメモリへ一括保存します。	4-232
ファイルレジスター括読出	FRLDM	メモリカードまたはCPUメモリに保存されているデータを指定したバンクのファイルレジスタに一括で読み出します。	4-234

FRSET @FRSET



ファイルレジスタバンク切り換え

ファイルレジスタのカレントバンク番号を指定した番号に切り換えます。

ラダープログラム

入力方法

オペランド	説明	占有サイズ
n	切り換え後のカレントバンク番号、またはその格納先を指定します。 （KV-8000/7000シリーズの場合、0～15、KV-5500/5000/3000の場合、0～3）	16ビット

動作説明

FRSET　実行条件がONのとき、ファイルレジスタのカレントバンク番号をnで指定した番号に切り換えます。

@FRSET　実行条件の立ち上がりで、1スキャンだけ実行します。

演算フラグ

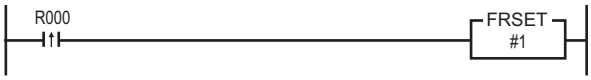
CR2009	変化なし
CR2010	変化なし
CR2011	変化なし
CR2012	• n ＞15の場合はON。（KV-5500/5000/3000の場合、n >3の場合はON) • 間接指定、インデックス修飾の範囲が不適切な場合はON。 • オペランドの範囲外アクセスが発生した場合ON。 上記以外はOFF

※　CR2012がONした場合、命令は実行されません。
CR2012がONした場合、CM5150～CM5176にエラーの詳細情報が格納されます。

「最新エラー詳細（CM）」（4-87ページ）

サンプルプログラム

入力リレーR000の立ち上がりで、ファイルレジスタのカレントバンクをバンク1に切り換えます。

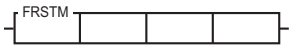


〈ニモニックリスト〉

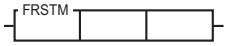
```
LDP R000
FRSET #1
```

FRSTM

KV-8000/7000シリーズの場合



KV-5000/3000シリーズの場合

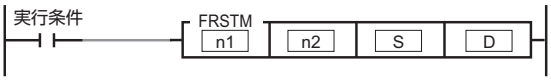


ファイルレジスタ一括保存

指定したバンク内のファイルレジスタをバイナリ形式で指定したストレージへ一括保存します。

KV-8000/7000シリーズ

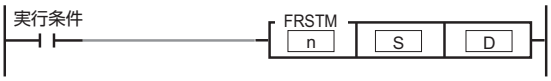
ラダープログラム



入力方法

KV-5000/3000シリーズ

ラダープログラム



入力方法

オペランド	説明	占有サイズ
n1	ドライブ番号、またはその格納先を指定します(0:メモリカード、1:CPUメモリ)。※1	16ビット
n2	バンク番号、またはその格納先を指定します(0～15)。※2	16ビット
S	データを保存するファイル番号、またはその格納先を指定します(0～999)。	16ビット
D	通知ビットデータの先頭アドレスを指定します。	1ビットx2

※1　KV-5000/3000シリーズの場合、ドライブ番号の選択は不要です。
※2　KV-5000/3000シリーズの場合、バンク番号の範囲は0～3です。

動作説明

FRSTM　実行条件の立ち上がりで、バンク番号 n2 のファイルレジスタ全てを、バイナリ形式で n1 で指定したストレージ(KV-5000/3000シリーズの場合、メモリカード)へ一括保存します(ファイルは常に上書きされます)。

ファイル番号 S は数値で指定します。

作成されるファイル名は、¥FR¥FR(ファイル番号3桁).FRU※ となります。

※KV-5000/3000シリーズの場合は、ファイルの拡張子はFRBとなります。
「FR」フォルダがない場合は、新規作成されます。

例　S に10が格納されている場合、¥FR¥FR010.FRU(またはFR010.FRB)が作成されます。

D には完了通知が、D +1には異常終了のエラー通知が書き込まれます。

- D は命令実行開始時にOFFし、ストレージへの書き込み処理終了時にONします。
  命令が正常終了、異常終了のどちらでもONします。
- D +1は命令が正常終了でOFFし、異常終了でONします。
  CM2390にエラーコードが格納されます。

ポイント　**ストレージ命令実行中リレー(CR3214)がONしているときに、命令は実行されません。CR3214を使用して排他制御をおこなってください。**

演算フラグ

CR2009	変化なし
CR2010	変化なし
CR2011	変化なし
CR2012	• n に格納されている値が「15」より大きい場合ON（KV-5500/5000/3000の場合「3」） • S に格納されている値が「999」より大きい場合ON • 間接指定･インデックス修飾の範囲が不適切な場合ON • オペランドの範囲外アクセスが発生した場合ON 上記以外はOFF。

※ CR2012がONした場合、命令は実行されません。
CR2012がONした場合、CM5150～CM5176にエラーの詳細情報が格納されます。
「最新エラー詳細（CM）」（4-87ページ）

サンプルプログラム

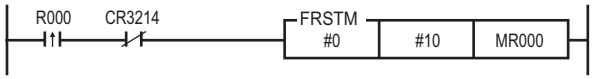
入力リレーR000の立ち上がりで、バンク0のファイルレジスタをファイル番号10でメモリカードに保存します。

KV-8000/7000シリーズ

〈ニモニックリスト〉

```
LDP R000
ANB CR3214
FRSTM #0 #0 #10 MR000
```

KV-5000/3000シリーズ

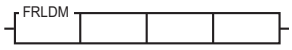


〈ニモニックリスト〉

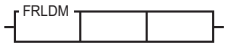
```
LDP R000
ANB CR3214
FRSTM #0 #10 MR000
```

FRLDM

KV-8000/7000シリーズの場合



KV-5000/3000シリーズの場合

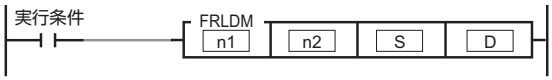


ファイルレジスタ一括読出

メモリカード(CPUメモリ)に保存されているデータを指定したバンク内のファイルレジスタに一括で読み出します

KV-8000/7000シリーズ

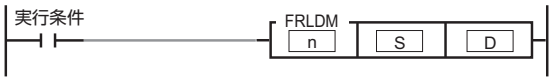
ラダープログラム



入力方法

KV-5000/3000シリーズ

ラダープログラム



入力方法

オペランド	説明	占有サイズ
n1	ドライブ番号、またはその格納先を指定します(0:メモリカード、1:CPUメモリ)。※1	16ビット
n2	バンク番号、またはその格納先を指定します(0～15)。※2	16ビット
S	データを保存するファイル番号、またはその格納先を指定します(0～999)。	16ビット
D	通知ビットデータの先頭アドレスを指定します。	1ビットx2

※1　KV-5000/3000シリーズの場合、ドライブ番号の選択は不要です。
※2　KV-5000/3000シリーズの場合、バンク番号の範囲は0～3です。

動作説明

FRLDM　実行条件の立ち上がりで、n1で指定したストレージ(KV-5000/3000シリーズの場合、メモリカード)に(FRSTM命令・または《KV STUDIO》の〔ファイルレジスタ設定〕により)保存されているデータを、バンク番号n2のファイルレジスタに一括で読み出します。(ファイルは常に上書きされます)。

ファイル番号Sは数値で指定します。

読み出されるファイル名は、¥FR¥FR(ファイル番号3桁).FRU※ となります。

※KV-5000/3000シリーズの場合は、ファイルの拡張子はFRBとなります。

例　Sに10が格納されている場合、¥FR¥FR010.FRU(またはFR010.FRB)から読み出します。

Dには完了通知が、D+1には異常終了のエラー通知が書き込まれます。

- Dは命令実行開始時にOFFし、ストレージへの書き込み処理終了時にONします。命令が正常終了、異常終了のどちらでもONします。
- D+1は命令が正常終了でOFFし、異常終了でONします。CM2390にエラーコードが格納されます。

ポイント　**ストレージ命令実行中リレー(CR3214)がONしているときに、命令は実行されません。CR3214を使用して排他制御をおこなってください。**

演算フラグ

CR2009	変化なし
CR2010	変化なし
CR2011	変化なし
CR2012	• n に格納されている値が「15」より大きい場合ON（KV-5500/5000/3000の場合「3」） • S に格納されている値が「999」より大きい場合ON • 間接指定･インデックス修飾の範囲が不適切な場合ON • オペランドの範囲外アクセスが発生した場合ON 上記以外はOFF。

※ CR2012がONした場合、命令は実行されません。
CR2012がONした場合、CM5150～CM5176にエラーの詳細情報が格納されます。
「最新エラー詳細（CM）」（4-87ページ）

サンプルプログラム

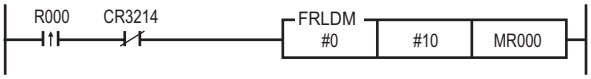
入力リレーR000の立ち上がりで、CPUメモリ内のファイル番号10のデータを、バンク0のファイルレジスタに読み出します。

KV-8000/7000シリーズ

〈ニモニックリスト〉

```
LDP R000
ANB CR3214
FRLDM #0 #0 #10 MR000
```

KV-5000/3000シリーズ



〈ニモニックリスト〉

```
LDP R000
ANB CR3214
FRLDM #0 #10 MR000
```

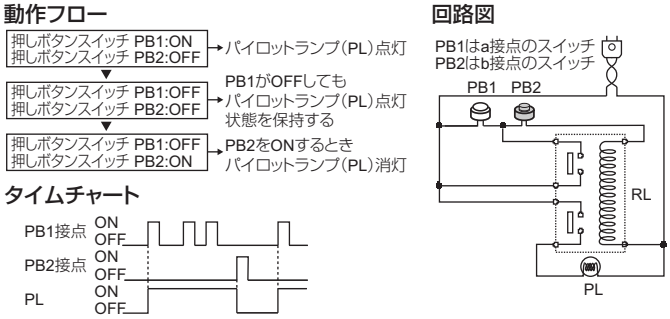
4-15 プログラミングのコツ

導入から運用までの流れ

《KV-8000シリーズ》の導入から運用までの流れを以下の図に示します。

動作内容の検討

動作フロー、仕様書、タイムチャート、回路図などを記述することによって動作内容を検討します。



プログラムの検討

◇ユニット構成

制御に使用する入出力機器が多い時や拡張機能を使用する時は、増設する拡張ユニットを検討します。

『KV-8000シリーズおよび拡張ユニットのユーザーズマニュアル』

◇モジュール構成

処理、機能、工程ごとなどに作成するモジュール構成を検討します。

「モジュールの構成例」(4-110ページ)

◇シーケンスプログラムの構成

モジュールごとに作成するプログラムの制御内容を、使用する命令語を想定しながら概略を検討します。

『KV-8000/7000/5000/3000/1000シリーズ･KV Nanoシリーズ命令語リファレンスマニュアル』

プログラムの作成

《KV STUDIO》でプログラムを作成します。

◇プロジェクトの新規作成

『KV STUDIO ユーザーズマニュアル』「プロジェクトの新規作成」

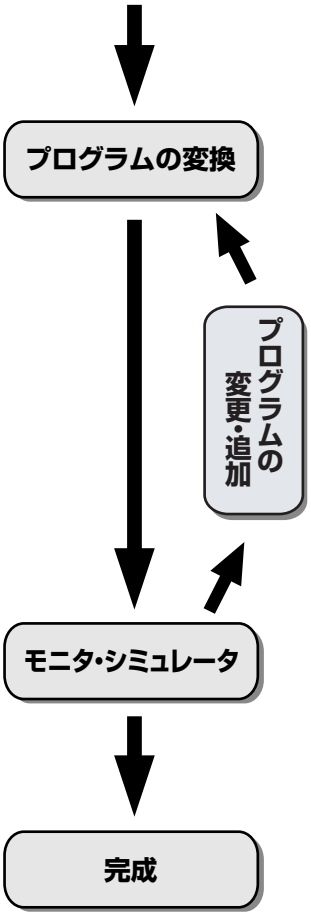
◇ユニットの設定

各ユニットに割り当てるユニットデバイスを、ユニットエディタ上で設定します。

『KV STUDIO ユーザーズマニュアル』「3章 ユニットエディタ」

◇プログラムの作成

「4-9 モジュール」(4-98ページ)
「プログラム作成を支援する」(4-238ページ)
「プログラム作成のポイント」(4-258ページ)
「4-10 ファンクションブロック」(4-117ページ)
『KV-8000/7000/5000/3000/1000シリーズ･KV Nanoシリーズ命令語リファレンスマニュアル』



PLCへ転送してモニタ時、シミュレータへの移行時にプログラムを変換します。変換エラーが出た場合は、[アウトプットウィンドウ]を参照しながらエラー原因を取り除いてください。

「変換」(4-266ページ)
「ローカルデバイス入力・変換時のエラー」(4-204ページ)
『KV STUDIO ユーザーズマニュアル』「付録-1 エラーメッセージ一覧」
「プログラム作成のポイント」(4-258ページ)

プログラムの変更や追加をします。

「見やすいプログラムを作成する」(4-252ページ)
「オンラインエディット」(4-269ページ)
「RUN中書き込みについて」(4-267ページ)
「シミュレータエディット」(4-271ページ)

モニタ・シミュレータの便利機能を活用して、デバッグをおこないます。

モニタ 「モニタ」(4-268ページ)
シミュレータ 「シミュレータ」(4-270ページ)

再利用性の高いプログラム→プログラムのインポート

「モジュールの管理」(4-98ページ)
「ファンクションブロックの管理」(4-117ページ)

プログラム作成を支援する

ラダープログラムを作成する上で、入力･修正工数を削減できる便利な機能を紹介します。

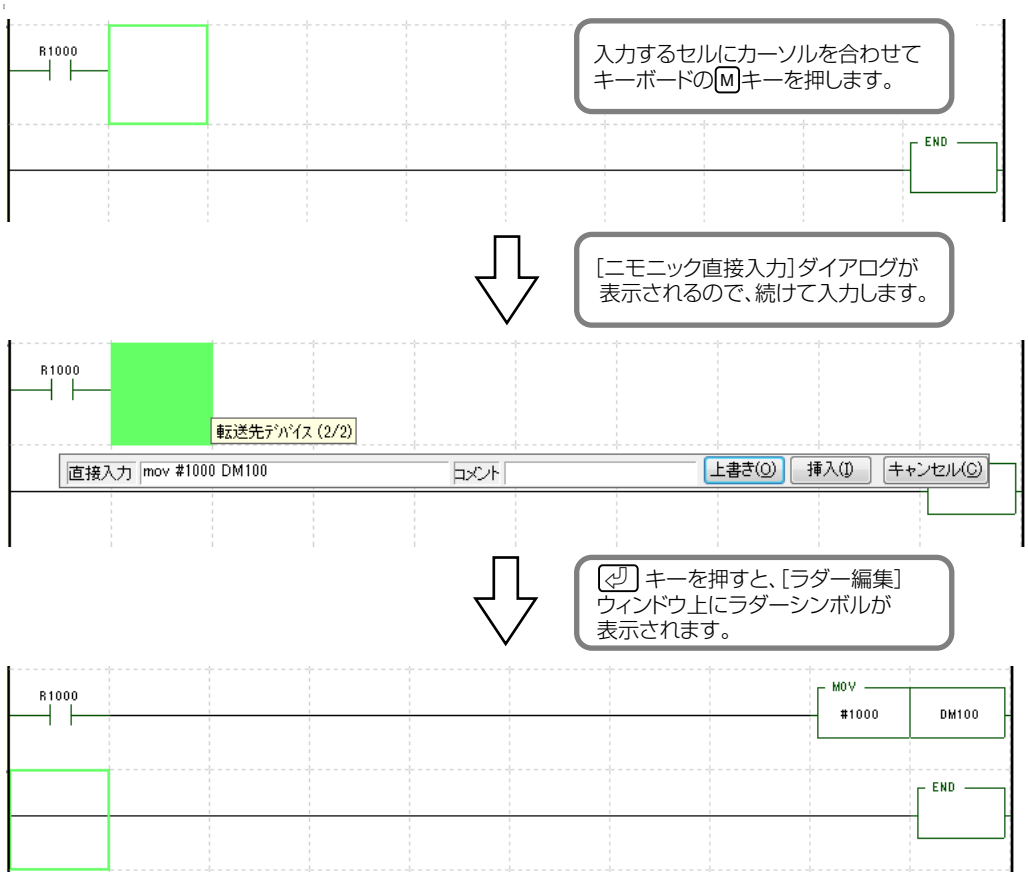
ニモニック入力

ニモニック入力はマウス操作なしで命令語を入力できるため、[命令語パレット]を利用するよりも短い時間でプログラムを作成できます。

■ 入力方法

命令語の先頭文字をキーボードで入力します。

（例） MOV命令を入力する場合



参 考 [ニモニック直接入力]ダイアログは、【ツール】→【オプション】の[編集･キーの設定]で「直接入力ダイアログの縮小表示」をチェックすると縮小表示されます。

各命令語の詳細については、『KV-8000/7000/5000/3000/1000シリーズ･KV Nanoシリーズ命令語リファレンスマニュアル』を参照してください。

参 考

- 入力する命令語やオペランドに指定する対象デバイスや値を、短縮して入力できます。
  (例) A 0 → LD R000
  「付-6 ニモニック短縮入力一覧」(付-23ページ)
- 既に入力済みのセルを[ニモニック直接入力]ダイアログで編集する時は、Escキーもしくは F2 キーを押します。(【ツール】→【オプション】の[編集·キーの設定]で「直接入力ダイアログをEnterキーで起動」をチェックすると、⏎ キーを[ニモニック直接入力]ダイアログに割り当てられます。)

■ 入力アシスト

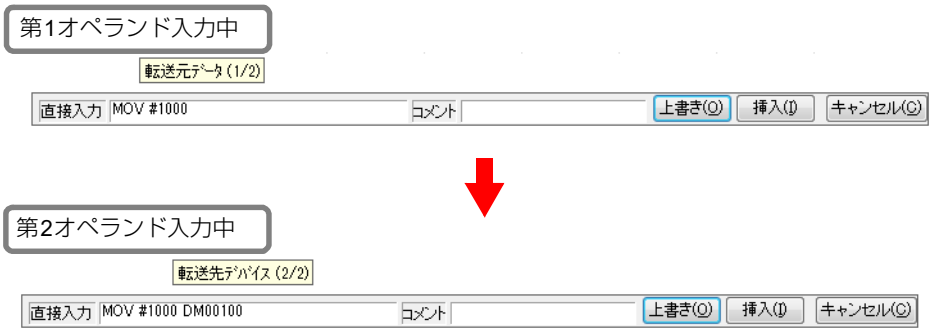
ニモニック入力には、命令語の入力をアシストする機能がいくつかあります。

● 命令語の一覧表示

命令語の頭文字を入力すると、入力した頭文字を先頭にした[命令語候補]ダイアログが表示され、命令語を選択できます。

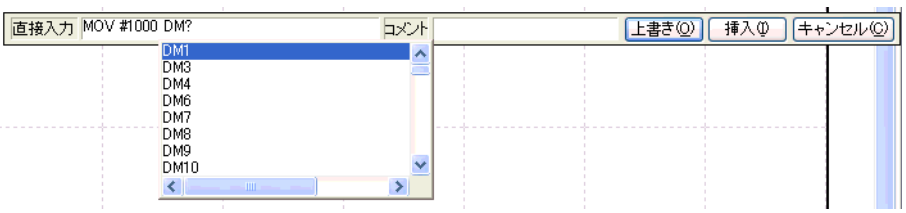
● 命令語のオペランド簡易説明表示

現在入力中のオペランドの簡単な説明と、何番目のオペランドかが表示されます。



● 未使用デバイスの簡易検索

デバイス種別または番号の途中まで入力した後に「?」と入力すると、未使用デバイスが一覧表示されます。



ショートカットキー

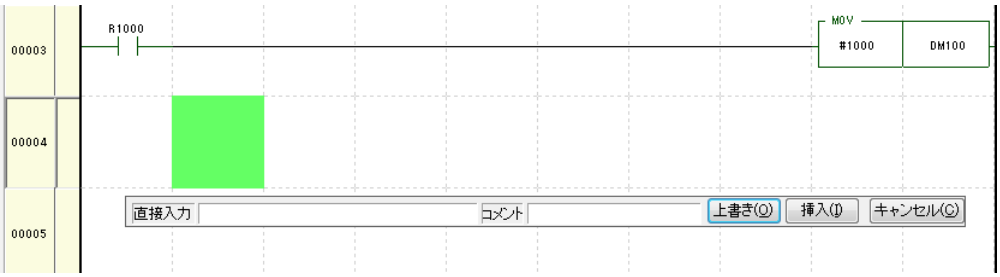
ショートカットキーを使用すると、マウスを使わずにキーボードだけで操作できます。また、キーボードの打鍵数が減るため、入力工数の削減につながります。
代表的なショートカットキーを説明します。

「付-7 ショートカットキー一覧」(付-25ページ)

■ 入力・編集

● Esc F2

[ニモニック直接入力]ダイアログが表示され、ニモニックでの直接入力・編集ができます。

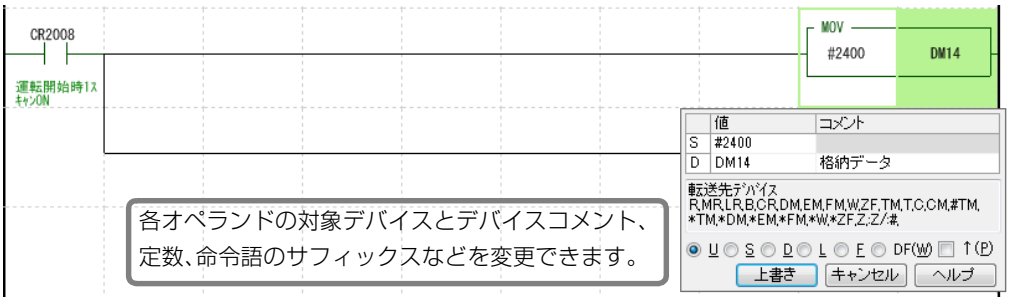


「ニモニック入力」(4-238ページ)

● ⏎

<エディタモードで空白のセルを選択している場合>
[ニモニック直接入力]ダイアログが表示され、ニモニックでの直接入力・編集ができます。

<エディタモードで入力済みのセルを選択している場合>
[オペランド編集]ダイアログが表示されます。



<モニタモード、シミュレータモードの場合>
[検索/置換]ダイアログボックスが表示されます。

● Ctrl + M

入力済みのセルを選択している場合、上記の[オペランド編集]ダイアログが表示され、コメントを編集できます。

● Ctrl ＋ X ／ C ／ V

Windowsアプリケーション標準の編集用ショートカットキーです。

Ctrl ＋ X ･･･ 切り取り

Ctrl ＋ C ･･･ コピー

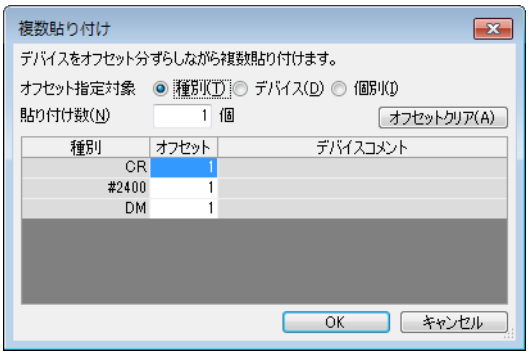
Ctrl ＋ V ･･･ 貼り付け

● Ctrl ＋ Shift ＋ V

貼り付ける個数を設定して、複数貼り付けます。

また、含まれているデバイスをオフセットすることも可能です。

『KV STUDIO ユーザーズマニュアル』「5-1 編集の各機能について―複数貼り付け」



● Ctrl ＋ Z ／ Y

Windowsアプリケーション標準の編集用ショートカットキーです。

Ctrl ＋ Z ･･･ 元に戻す

Ctrl ＋ Y ･･･ やり直し

● Tab ／ Shift ＋ Tab

ラダーシンボル単位でカーソルを移動します。

● Ctrl ＋ ↑ ／ ↓

ラダー回路ブロックの単位でカーソルを移動します。

● Ctrl ＋ →

選択しているセルがある行全体を選択します。

● Ctrl ＋ Shift ＋ ↑ ／ ↓

選択しているセルのオペランド番号を変更できます。

■ 挿入・削除

● Insert

選択しているセルに横接続線を挿入します。

● Delete

選択しているセルの命令語/接続線を削除します。

● Shift + ⏎ または Shift + Insert

行を挿入します。

● Shift + Delete

行を削除します。

● Ctrl + ⏎

行コメントを挿入します。

「行コメント」(4-254ページ)

■ 接続線編集

● Alt + ⏎ または Ctrl + Tab

選択しているセルから行末まで接続線を引きます。

● Alt + ← / ↑ / → / ↓

接続線がない場合、矢印の方向に接続線を引きます。
接続線がある場合、矢印の方向に接続線を削除します。

■ 検索・置換

● Ctrl + F

[検索／置換]ダイアログの[検索]タブが表示されます。

「検索・置換」(4-244ページ)

● Ctrl + H

[検索／置換]ダイアログの[置換]タブが表示されます。

● Space

クロスリファレンス表示

「クロスリファレンス」(4-244ページ)

■ 表示

● Ctrl ＋ F7

［デバイスコメント編集]ウィンドウが表示されます。

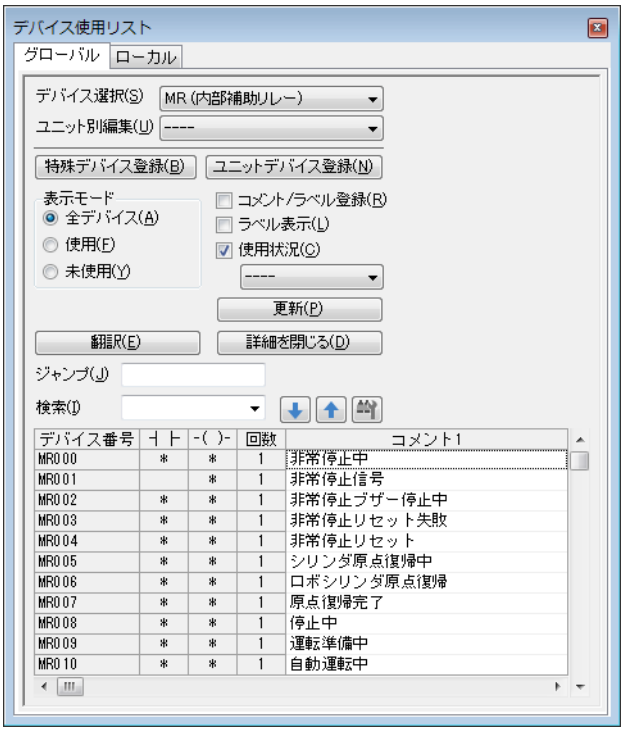
「デバイスコメント」(4-252ページ)

● Ctrl ＋ E

［デバイス使用リスト］ウィンドウが表示されます。

未使用デバイスを簡単に確認できます。

『KV STUDIO ユーザーズマニュアル 』「6-2 表示の各機能について―デバイス使用リスト」



● Ctrl ＋ Page Up ／ Ctrl ＋ Page Down

[ラダー編集]ウィンドウの表示を拡大／縮小します。

※ 【表示】→【自動倍率】にチェックがついている場合は動作しません。

● Ctrl ＋ Space

ラダーエディタ上でのコメント表示／非表示を切り換えます。

検索・置換

指定した条件に合う命令語やオペランドを検索して表示します。また、合致した項目をアウトプットウィンドウに一覧表示することもできます。
デバイスを検索する際は、スペースキーを押すだけで一覧表示されるクロスリファレンス機能を使用してください。

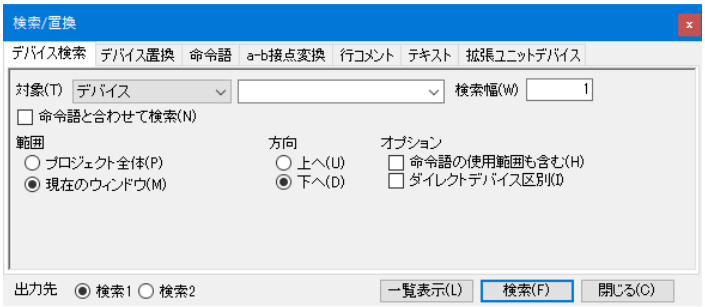
「クロスリファレンス」(4-244ページ)

1 [ラダー編集]ウィンドウを選択します。

2 右クリックして表示されるメニューから[検索(S)]を選択します。

別手順
- メニューから【編集(E)】→【デバイス検索(S)】を選択します。
- Ctrl + F

[検索/置換]ダイアログが表示されます。



対象として、「デバイス」「変数」「定数」「プログラム」「デバイスコメント」「変数コメント」「保留/無効」「無効化行」から選択できます。

クロスリファレンス

編集(E)→クロスリファレンスの作成(X)→オペランド一覧作成(O) Space

指定デバイスの検索結果一覧をアウトプットウインドウ[検索1]/[検索2]に表示します。

1 クロスリファレンスを作成したいデバイス(ラダーシンボル)を選択します。

2 メニューから【編集(E)】→【クロスリファレンスの作成(X)】→【オペランド一覧作成(O)】を選択します。

アウトプットウインドウ[検索1]/[検索2]に検索結果一覧が表示されます。

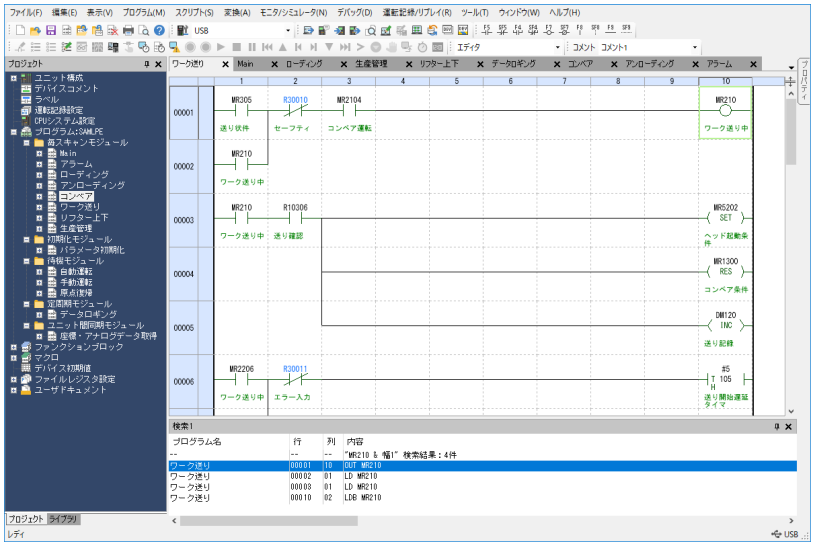
別手順
- (スペース) キー
- 右クリックして表示されるメニューから【クロスリファレンス作成(L)】→【オペランド一覧作成(O)】

3 **検索結果一覧から確認したい内容をダブルクリックします。**

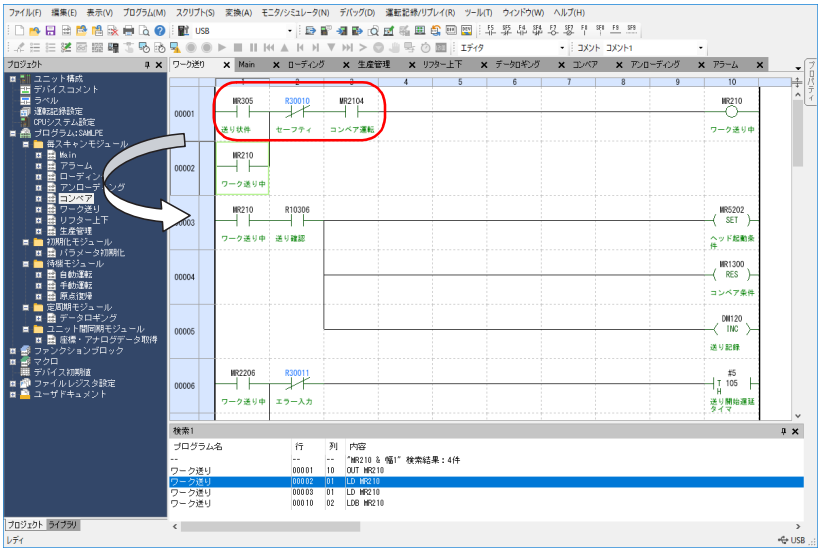
一覧に表示されているデバイスをダブルクリックすると、ラダー上の該当するデバイス(シンボル)にカーソルが移動します。

別手順 ・F3

・Shift + F3



4 **次の内容にフォーカスを移したいときは、F3 キーを押します。**



5 **一つ前の内容にフォーカスを移したいときは、Shift キー+ F3 キーを押します。**

リレーションマッピング

編集(E)→【リレーションマッピング(9)】 Ctrl + Shift + R

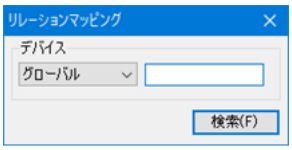
選択したデバイスが変化する要因となる回路ブロックや《KV STUDIO》の設定、タッチパネルからのデバイス書き換えイベントを検索･表示する機能です。

ポイント **対応機種がKV-8000(A)の場合にのみ使用できます。**

1 メニューの【編集(E)】→【リレーションマッピング(9)】を選択します。

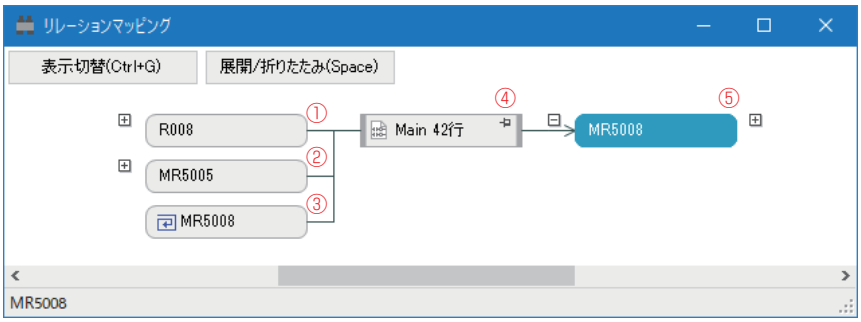
別手順
- ラダー編集ウィンドウで対象のデバイスを選択した状態で、右クリックメニューの【リレーションマッピング(9)】を選択
- Ctrl + Shift + R

2 対象のデバイスを入力し、〔検索(F)〕をクリックします。

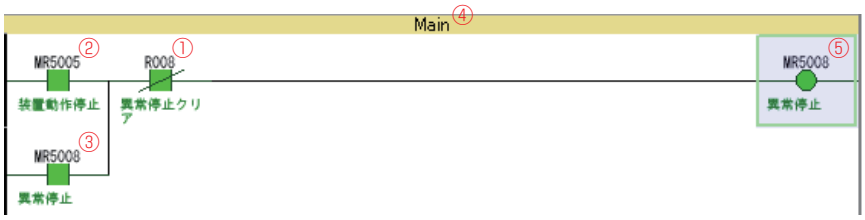


入力したデバイスを対象として[リレーションマッピング]ウィンドウが表示されます。[リレーションマッピング]ウィンドウでは、状態を変更しているラダープログラム/機能の情報を表示したり、状態を変更する要因を表示したりできます。

● リレーションマッピングウィンドウ



● ラダー図

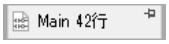


■ 各ブロックの説明

● デバイスブロック

ラダーブロック図で使用されているデバイスを表示します。[リレーションマッピング]ウィンドウ左上の〔表示切替〕ボタンをクリックするか、Ctrl + G を押すことでデバイスコメント表示とデバイス表示を切り換えます。

● ラダーブロック

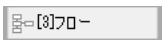


検索したデバイスが使用されているモジュール名と行番号を表示します。選択すると、選択ラダー表示ウィンドウに対象となるラダープログラムを表示します。

ポイント

選択ラダー編集ウィンドウ上ではラダープログラムの編集はできません。対象のラダープログラムを編集する場合は、選択ラダー編集ウィンドウ上で右クリックし、【ジャンプ】を選択し、元のラダープログラムを編集します。
スクリプトの展開はできません。

● フローブロック



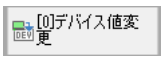
検索したデバイスがユニットプログラム（フロー）で使用されている場合に、対象ユニットのユニット番号を表示します。

● 設定ブロック

検索したデバイスが使用されているユニット設定の名称とユニット番号を表示します。設定ブロックに表示されるユニット設定は以下のとおりです。

- PLCリンク
- PROTOCOL STUDIO
- 産業用イーサネット（EtherNet/IP（アダプタ）、PROFINET、CC-Link IE Field、EtherCAT）
- 簡易PLCリンク
- FLリフレッシュ設定
- シリアルPLCリンク
- デバイス初期値設定
- ユニット間同期リフレッシュ

● デバイス値変更ブロック



検索したデバイスがタッチパネルやアクセスウィンドウからデバイス値を変更された場合に表示します。

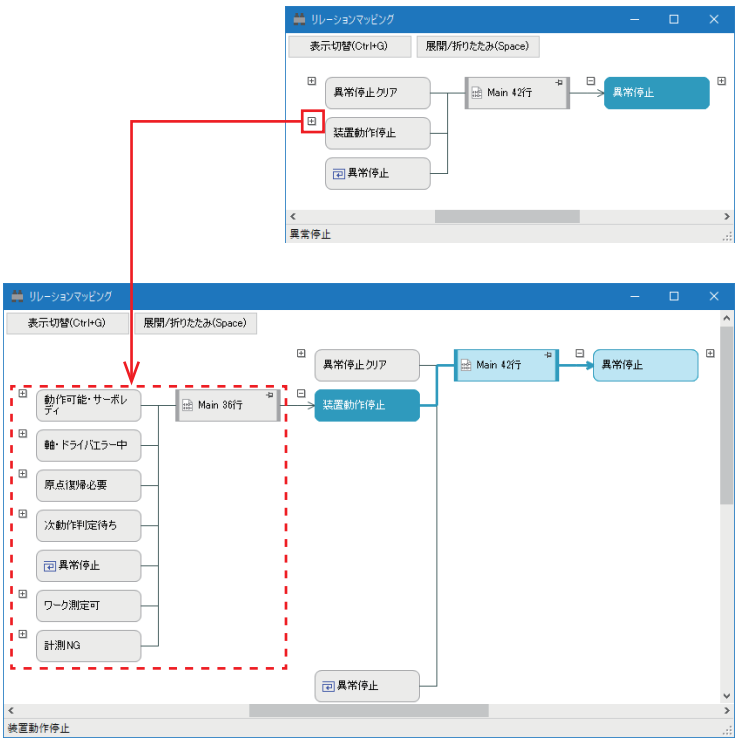
ポイント

デバイス値変更ブロックはモニタ/オンラインエディット/リプレイモードの場合のみ表示します。

■ リレーションマッピングの操作

● 追加検索

デバイスブロックに表示されている〔+〕ボタンをクリックすると、そのデバイスの状態を変更している要因を追加で検索、表示します。



● ピン止め

ラダーブロックに表示されている アイコンをクリックすると、ラダーブロックの表示を固定できます。

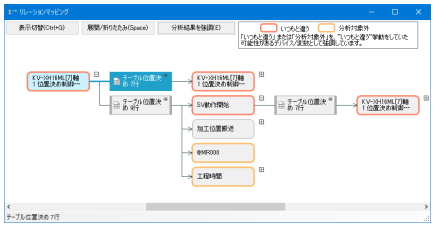
● ジャンプ

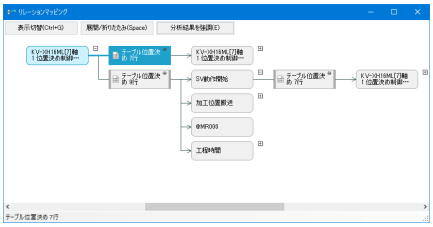
ラダーブロック、フローブロック、設定ブロックでのみ使用できます。ブロックを右クリックし、【ジャンプ(J)】をクリックすると以下の内容を表示します。

項目	内容
ラダーブロック	対象となるラダーの行へ移動します
フローブロック	対象となるフローを表示します。
設定ブロック	対象となる設定画面を表示します。

参 考

リプレイモードの分析レポートビューワからリレーションマッピングを起動した場合、〔分析結果を強調(E)〕ボタンが表示されます。運転記録のマスタデータと違う挙動をしていた可能性の高いデバイス/変数が強調されます。
詳細は『KV-XD02　ユーザーズマニュアル』を参照してください。



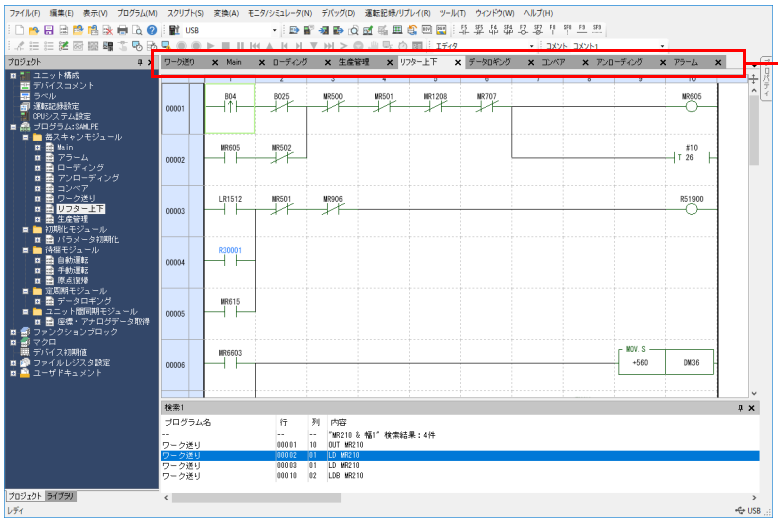


ウィンドウメニュー

ウィンドウメニューを活用することにより、プログラム作成の効率化を助けます。
ウィンドウメニューの中で、「タブ付きドキュメント」の使用方法について説明します。

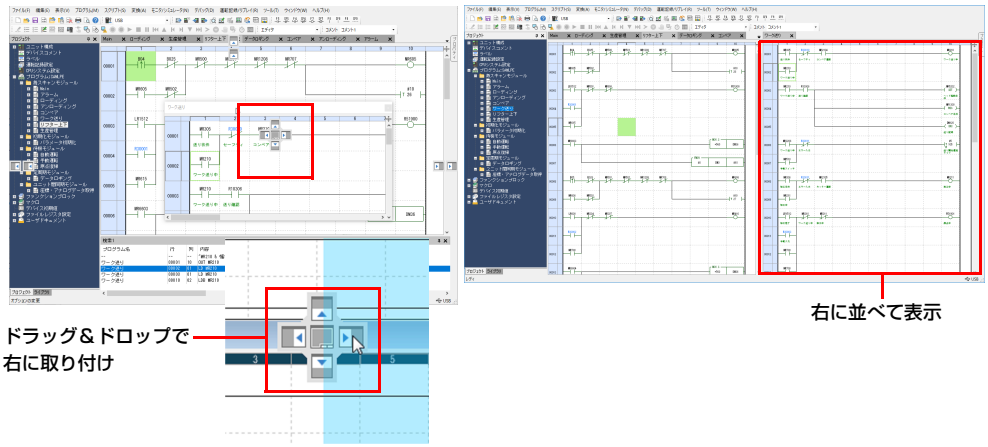
■ タブ付きドキュメント

KV STUDIO Ver.8以降では、メニューから【ウィンドウ(W)】→【タブ付きドキュメント】を選択し、プログラムをタブ形式で表示できます。



タブ
それぞれのタブを選択する、または Alt + F6 を押すことでタブを切り替えることができます。

タブをドラッグ＆ドロップして、ドッキング機能とあわせると、並べて表示なども簡単に実行できます。



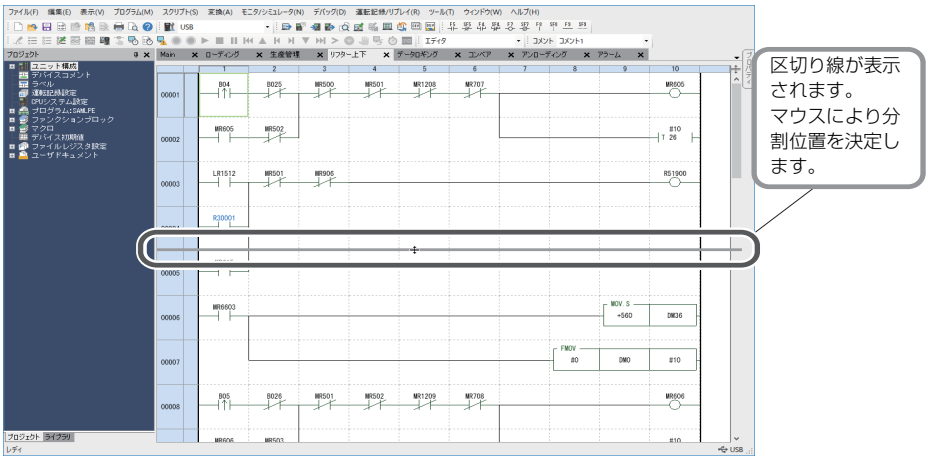
ドラッグ＆ドロップで
右に取り付け

右に並べて表示

■ 分割

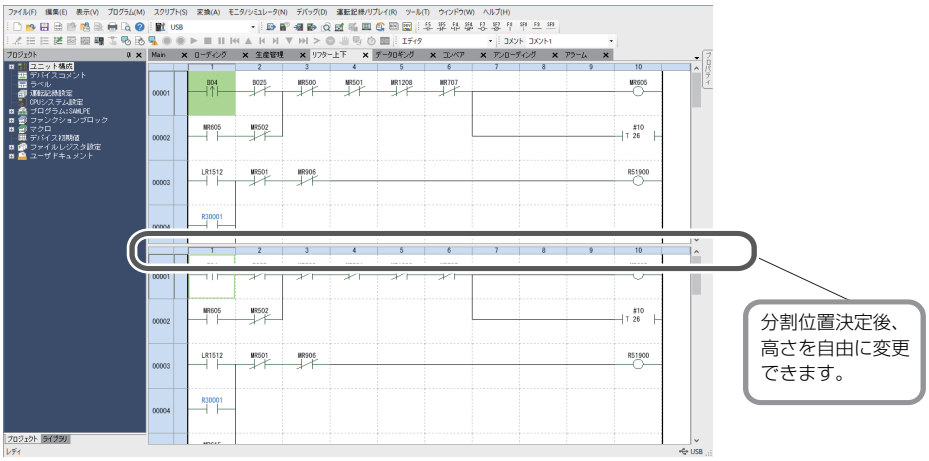
プログラムが長くなってくると、全てのプログラムを同時に表示できなくなります。
「分割」機能はプログラムを2つに分割表示し、見たい部分を同時に表示できます。

1 **メニューから【ウィンドウ(W)】→【分割(P)】を選択します。**



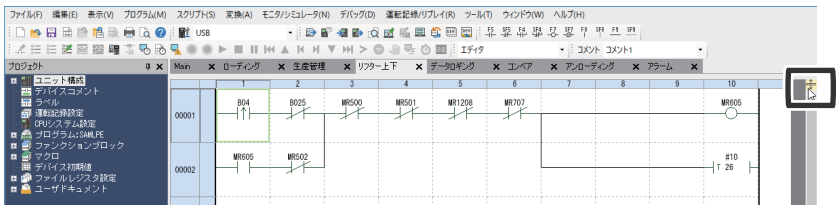
2 **マウスクリックで、分割位置を決定します。**

【分割画面】



分割位置決定時、デフォルトで上･下画面は先頭からのプログラムを表示します。
右スクロールバーにより、上･下画面それぞれ目的のプログラム位置を表示します。

別手順 マウスのカーソルを下図の位置に持っていくと「⇕」になり、ドラッグすると分割を開始します。



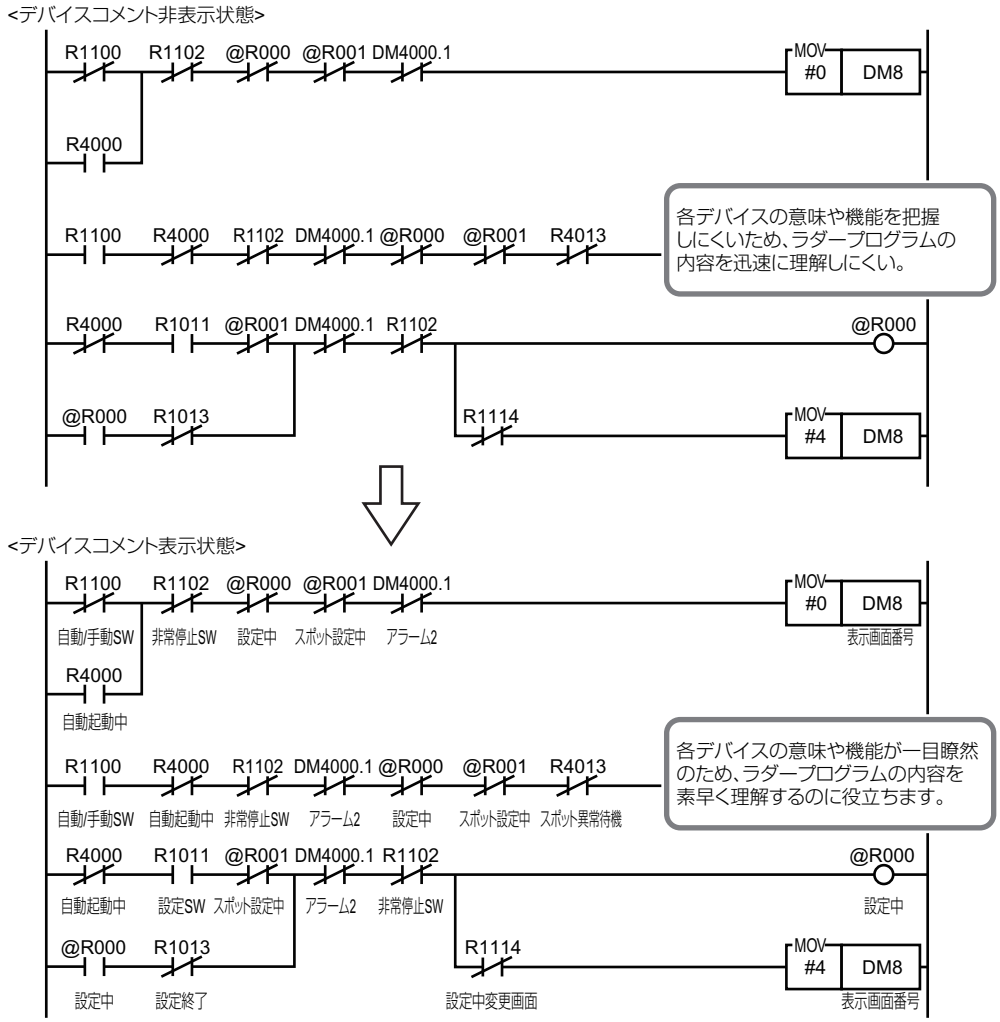
分割位置でドロップします。

参 考
- 上下の画面間でラダープログラムの一部をコピー&ペーストできます。
- 表示サイズもそれぞれ別々に設定できます。

見やすいプログラムを作成する

デバイスコメント

各デバイスごとにコメントの設定ができます。
設定したコメントは、《KV STUDIO》でラダーを表示する際に各デバイスの下側に表示させたり、アクセスウィンドウ上に表示したりできるため、各デバイスに割り当てた機能や接点の内容が把握しやすくなります。
また、作成者以外の人がプログラムを参照する場合に、デバイスコメントがあるとプログラムの理解に要する時間を短縮できます。グローバルデバイス、ローカルデバイスともにデバイスコメントを使用できます。ワードデバイスをビット扱いにした場合、ビット単位でデバイスコメントを使用できます。



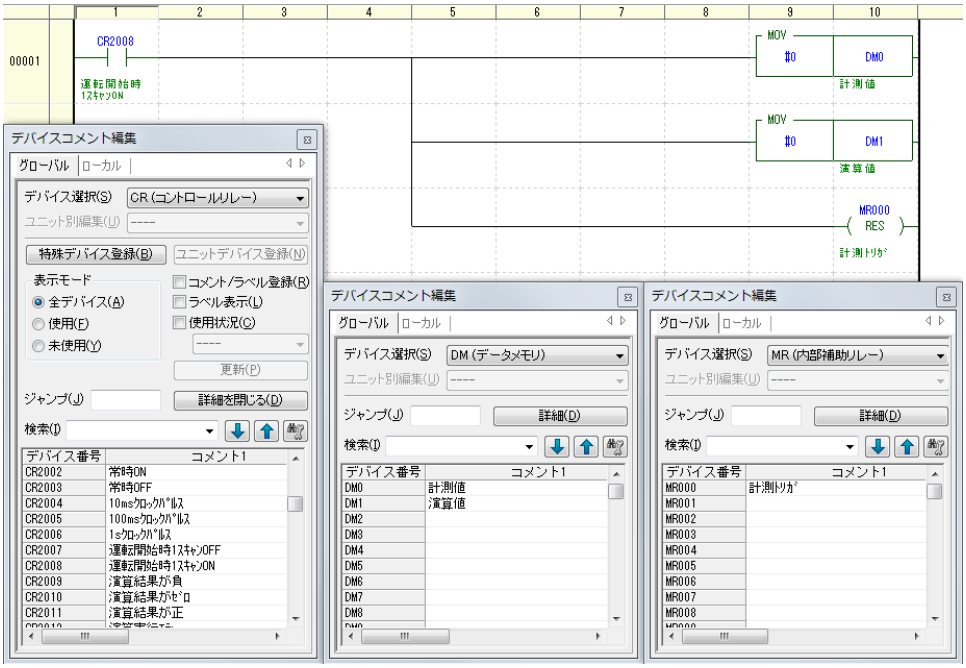
ポイント　インデックスレジスタZにはデバイスコメントを設定できません。

■ 入力方法

デバイスコメントの入力方法は、次の2種類があります。

- ［オペランド編集］ダイアログから入力する。（Ctrl ＋ M）
- ［デバイスコメント編集］ウィンドウから入力する。

参 考 ［デバイスコメント編集］ウィンドウを表示した状態でプログラムすると、デバイスコメントを確認しながら入力できるので便利です。同時に3画面まで表示できます。



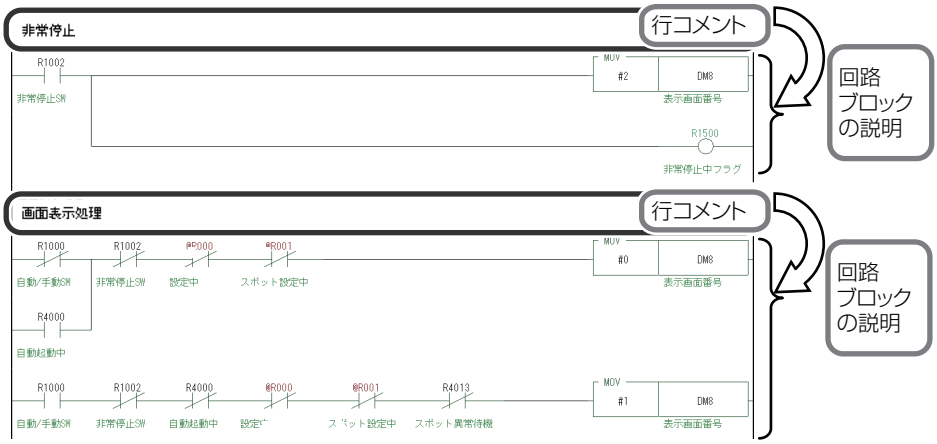
■ 表示方法

デバイスコメント設定後、メニューから【表示(V)】→【コメント表示(F)】により、ラダーエディタ上のデバイスコメントの表示／非表示を切り換えられます。

別手順 Ctrl ＋ Space

行コメント

[ラダー編集]ウィンドウ上の各行にコメントを記述できます。
行コメントにより、プログラムを編集する時に各回路ブロックの内容がひと目でわかります。
また、行コメントのみを一覧表示して、任意の行コメントの行へ移動できます。



■ 入力方法

次の手順で行コメントを入力します。

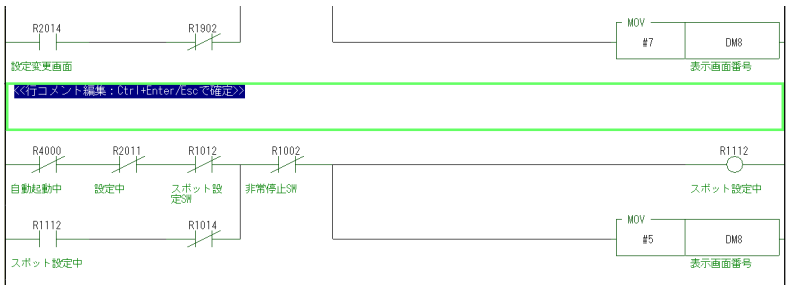
1 **[ラダー編集]ウィンドウ上で、行コメントを入力する行にカーソルを合わせます。**

2 **右クリックして表示されるメニューから[行コメントの挿入(H)]を選択します。**

別手順 ・メニューから【編集(E)】→【挿入(W)】→【行コメント(H)】を選択します。

・Ctrl ＋ ⏎

以下のように行コメントエディタが表示されます。行コメントを入力します。



3 **Esc 、または Ctrl ＋ ⏎ を押します。**

行コメント入力完了です。

■ 行コメントの検索方法

[ラダー編集]ウィンドウ内の行コメントを一覧表示し、一覧から選択した行に移動できます。
次の手順で行コメントを検索します。

別手順 Ctrl + L

1 [ラダー編集]ウィンドウを選択します。

2 右クリックして表示されるメニューから[検索(S)]を選択します。

別手順 メニューから【編集(E)】→【行コメント検索(M)】を選択します。

[検索／置換]ダイアログが表示されます。

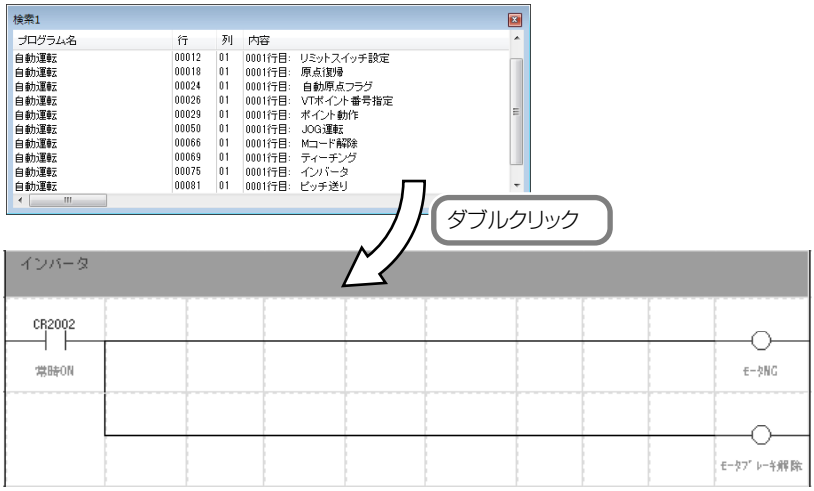
番号	項目	内容	設定範囲
1	行コメントの先頭行のみ表示	行コメントが複数行連続している場合に、全ての行を検索結果として表示するか、先頭行のみを表示するかを選択します。	チェックあり／なし (デフォルト：チェックあり)
2	検索範囲	行コメントを検索する対象範囲を選択します。	デフォルト：プログラム全体
3	一覧表示	[アウトプットウィンドウ]に対象範囲内にある行コメントを全て表示します。	–
4	出力先	一覧表示した際に表示するアウトプットウィンドウを[検索1]、[検索2]から選択します。	デフォルト：検索1

3 〔一覧表示〕ボタンを押します。

アウトプットウィンドウ[検索1]/[検索2]に行コメントの一覧が表示されます。

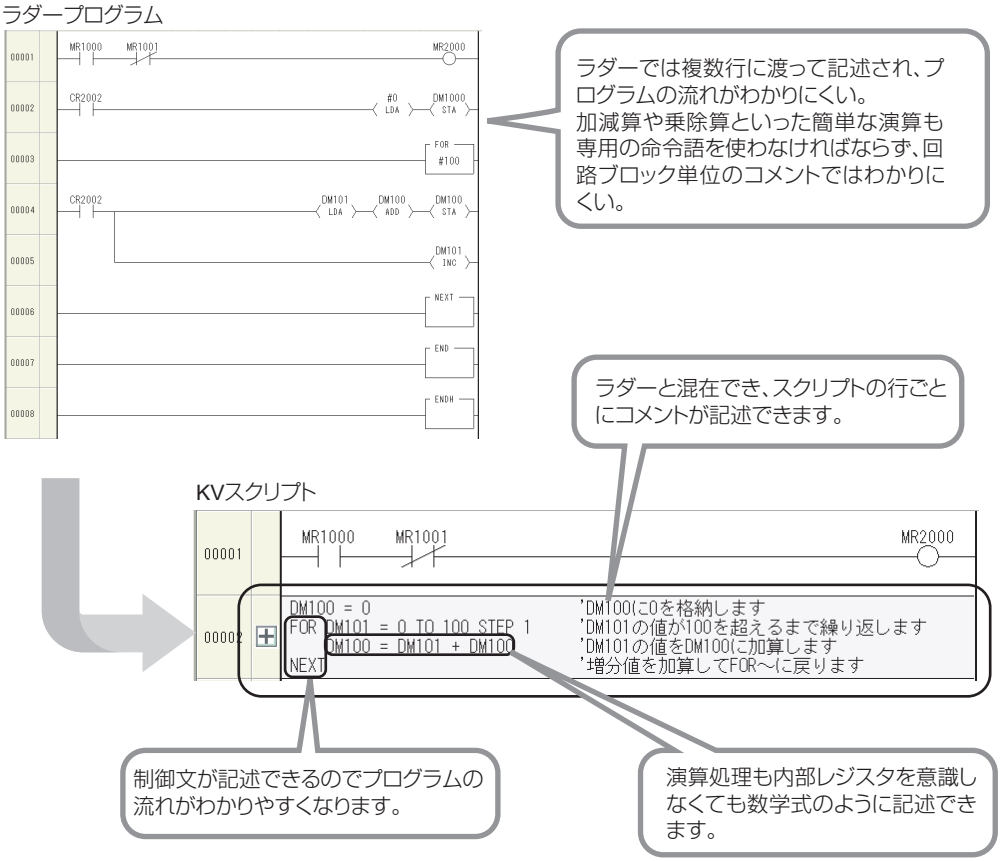
4 **ジャンプする行コメントをダブルクリックします。**

[ラダー編集]ウィンドウは選択した行コメントの行に移動します。



KVスクリプト

演算や文字列処理など、ラダープログラムで記述すると冗長になる部分をKVスクリプトで記述します。



『KV-8000/7000/5000/3000/1000シリーズ ·KV Nanoシリーズスクリプトプログラミングマニュアル』

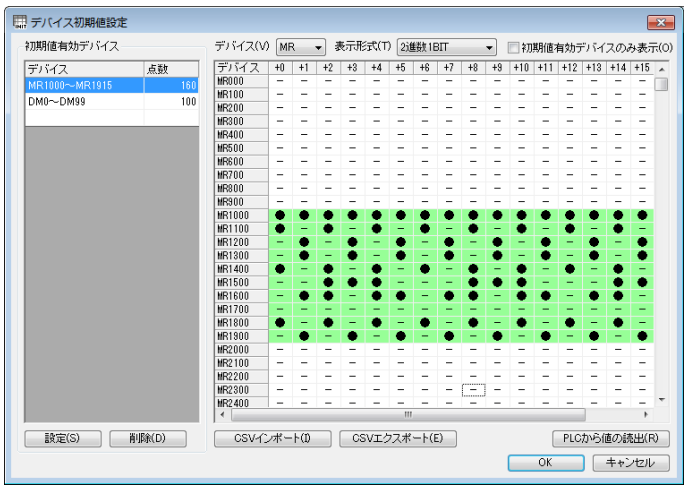
プログラムの構造化

デバッグしやすい、または再利用しやすいプログラムを作成するには、機能や工程ごとにモジュール（部品）化し、わかりやすい名前を付けて管理します。また、よく使うラダー回路ブロックをファンクションブロック、マクロ（オリジナル命令語）化し、命令語のように簡潔に入力できて覚えやすい名前を付けます。

「4-9 モジュール」（4-98ページ）
「4-10 ファンクションブロック」（4-117ページ）

デバイス初期値

ラダープログラムで全て記述すると冗長になるデバイス初期化のプログラムを、デバイス初期値設定で全て管理します。運転開始時に自動的に登録した値が書き込まれます。
ワークスペース上で〔デバイス初期値〕を選択します。



プログラム作成のポイント

ラダープログラムを作成する上で知っておくべきことや注意事項を説明します。

■ 拡張ラダーモード

通常のラダー表記の場合、OUT、TMR命令などの出力命令は、ラダー1行に1つしか配置できません。拡張ラダーを使用すると、1行に複数の出力命令を配置できますので、プログラムの行数削減、処理順序の明確化が可能です。

《KV STUDIO》では、拡張ラダーで入力する状態（拡張ラダーモード）と通常ラダーで入力する状態（通常ラダーモード）を切り換えられます。
次の手順で、モードを切り換えます。

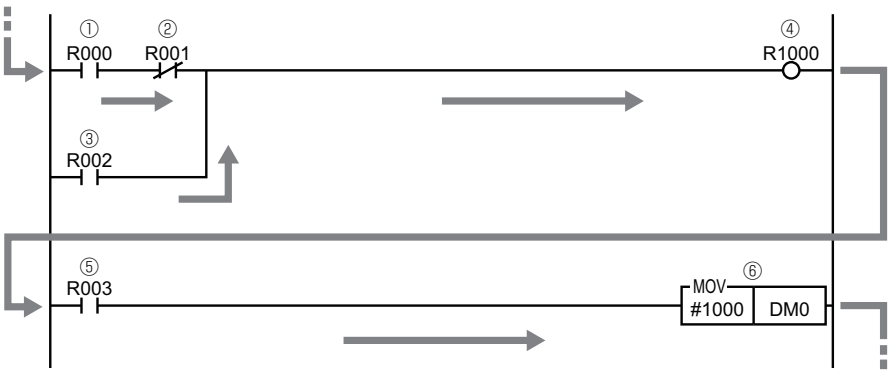
1 メニューから【編集(E)】→【拡張ラダーモード(G)】を選択します。

別手順 アイコンをクリック

参考
- 拡張ラダーモードが有効になっている時は、メニューの【拡張ラダーモード(G)】にチェックが表示され、枠付きのアイコンが表示されます。
- 拡張ラダーモードを選択すると、出力命令を入力する時に、カーソルが自動的に右端に移動せずに続けて命令語が入力できます。拡張ラダーモードでない場合は、出力命令を入力すると、カーソルが自動的に右端に移動します。
- デフォルトは拡張ラダーモードOFFです。

■ ラダープログラムの実行順序

ラダープログラムは、以下のように「左から右」、「上から下」の順番で実行されます。



参考
- プログラムはニモニックリストの順番に実行されます。

LD	R000	①
ANB	R001	②
OR	R002	③
OUT	R1000	④
LD	R003	⑤
MOV	#1000 DM0	⑥

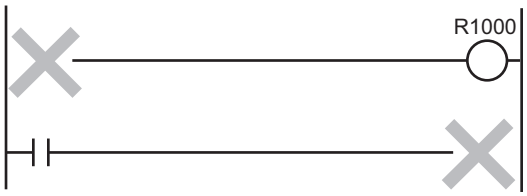
- モジュールの実行順序は《KV STUDIO》の［CPUシステム設定］の［モジュール実行順設定］で設定します。
「4-9 モジュールの実行順序」（4-109ページ）

■ 変換できないラダー

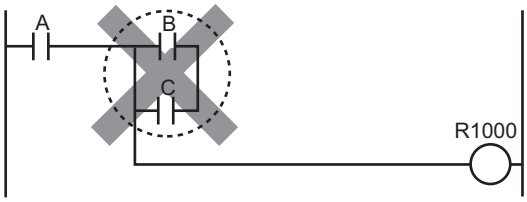
変換エラーになる代表的なパターンを以下に示します。

● 母線または出力線に接続されていない

すべての接点やコイルは、母線から出力線に至る経路上にあるようにし、未接続がないようにしてください。

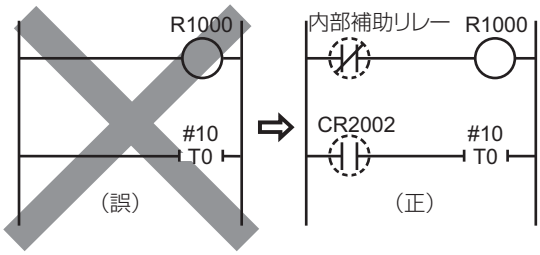


参 考　右図のラダーは未接続のため、変換エラーになります。

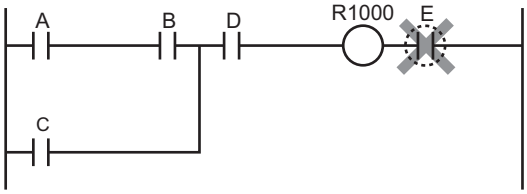


● 母線から直接記述できない命令を使用している

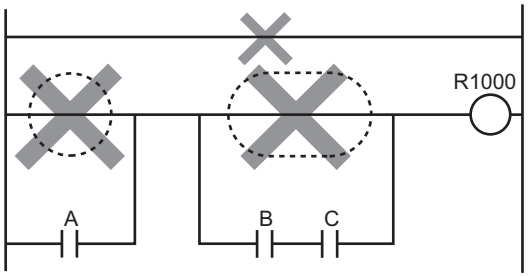
母線から直接、出力コイルやタイマ･カウンタ命令などを記述できません。
実行条件が必要ない場合は、未使用の内部補助リレーのb接点またはコントロールリレーCR2002(常時ON接点)をダミーとして挿入してください。



● 出力線に直接記述できない命令を使用している

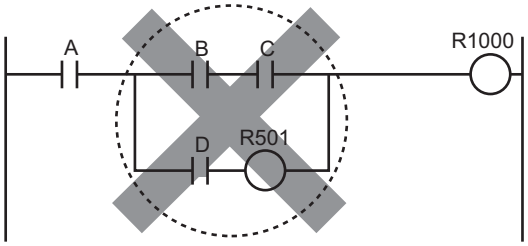


● 不正な短絡がある

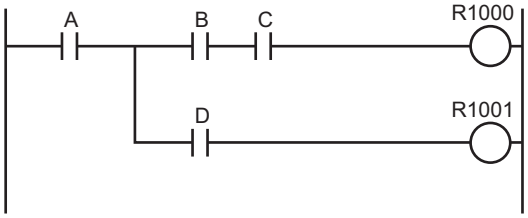


● 出力コイルが並列接続されている

● 出力コイルを含む回路群は、並列接続できません。



ただし、右図のように並列接続の回路群が、出力線にそれぞれ接続されていれば、並列接続が可能です。



● END/ENDH命令が存在しない

各プログラムごとに「END」「ENDH」命令が必要です。

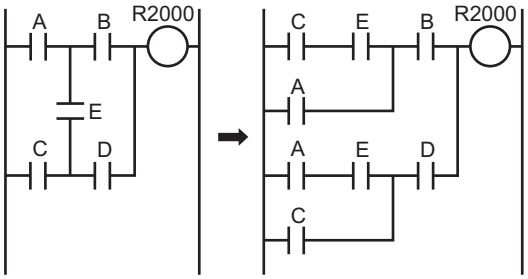
参 考

《KV STUDIO》でプログラムを新規作成すると、自動的に「END」「ENDH」命令が挿入されます。「END」「ENDH」命令はラダープログラムで記述してください。スクリプトで記述されると正常に動作しない場合があります。

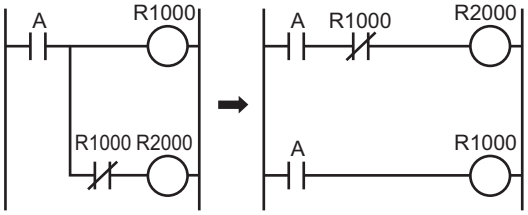
■ 書き換えの必要なラダー

NGラダー　　変換可能なラダー

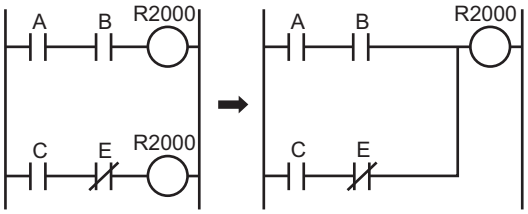
● 右図のような回路はプログラムできません。
書き換えてください。



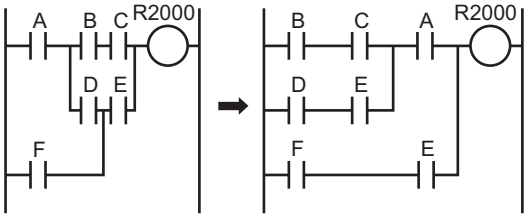
● 右図の場合、R2000は出力できません。
書き換えにより1スキャンのみ出力できるようになります。



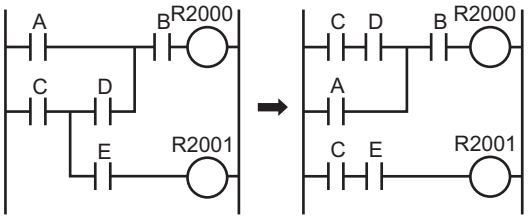
● 右図のように、同一コイルを2重に使用すると、後ろ側のプログラムが優先され、前側は無視されます。



● 右図のような回路はプログラムできません。
書き換えてください。



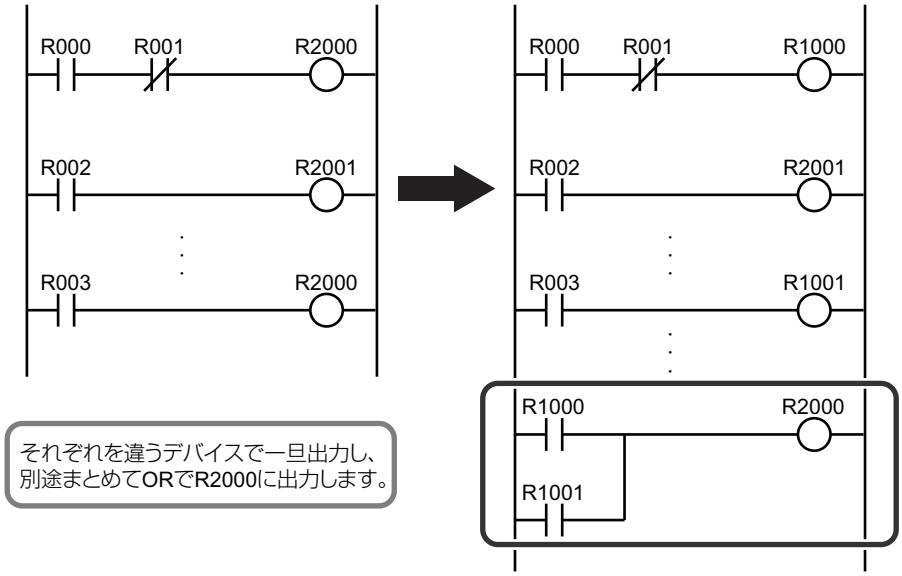
● 右図のような回路はプログラムできません。
書き換えてください。



■ 2重コイル

プロジェクト内で同じデバイスのOUT/OUB命令を2回以上使用すると、1スキャン内で1番最後に実行される命令が優先され、その他の命令は無視されます。
それぞれが離れた行、または違うプログラムに存在する場合は、以下のように対処します。

（例）下図のようにR2000の出力命令を2回使用している場合



2重コイルは変換エラーになりませんが、動作が煩雑になるため使用しないでください。
2重コイルの存在をチェックするには、《KV STUDIO》のメニューから【変換（A）】→【2重コイルチェック（D）】を選択してください。

ポイント

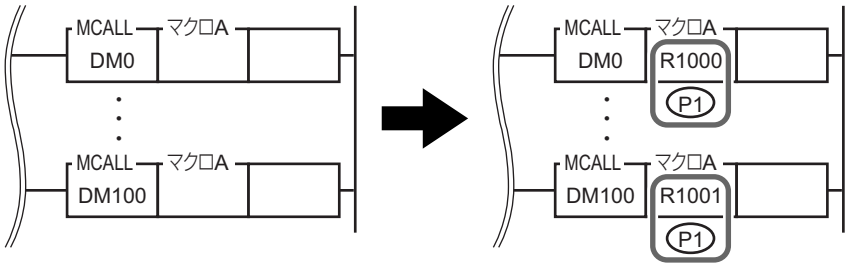
- **同じファンクションブロック、マクロを2回以上呼び出す場合、グローバルデバイスを使用したOUT/OUB命令があると2重コイルになります。**
- **2重コイルになっている場合でも、動作していないモジュールやファンクションブロック内に記載されている場合は影響ありません。**

● 同じファンクションブロック、マクロ間での2重コイルを避ける方法

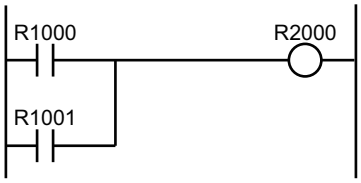
ファンクションブロック、マクロ内にグローバルデバイスを使用したOUT／OUB命令があり、そのプログラムを2回以上呼び出す場合に発生する2重コイルは、以下のように対処すると避けられます。

① ファンクションブロック、マクロ内にあるOUT/OUB命令のオペランドを引数タイプがIN-OUTの引数もしくは引数Pにします。

② ファンクションブロック、マクロ呼び出し命令で指定する①で追加した引数は、呼び出すごとに異なる内部補助リレーを指定します。



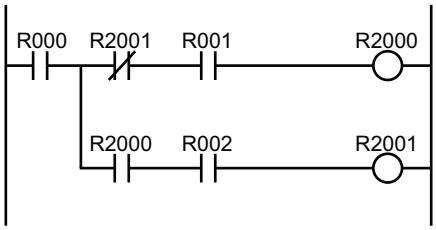
③ 別途まとめてORで出力するラダープログラムをファンクションブロック、マクロ外に追加します。



■ インターロック

プログラム内で一方が動作している場合に、他方の動作をできなくさせて、両者が同時に作動しないように排他処理された構成をインターロックといいます。
実際の装置には必要になる場合が多いのですが、『KV-8000/7000/5000/3000/1000シリーズ･KV Nanoシリーズ命令語リファレンスマニュアル』などで紹介しているサンプルプログラムでは、動作の説明が煩雑になるため省略しています。

（例） R2000とR2001が同時にONしない回路



R000とR001がON→R2000がONしている状態でR002がONしても、R2000のB接点が挿入されているため、R2001はONしません。

モジュールやマクロの実行中ONビットを使用すると、簡単にインターロック回路を作成できます。

「マクロの実行中ONビット」（4-189ページ）
「マクロシステムデバイス」（4-188ページ）

■ スキャンタイムの短縮

スキャンタイムが長くなると入出力や値の更新周期が大きくなるため、機械や装置の応答性が悪くなります。また、スキャンタイムが300msを超えると「スキャンタイムオーバー（CPU本体エラー No.30）」が発生し、運転を継続できません。
以下の項目を実施することによって、スキャンタイムを短縮できます。

● 必要のないモジュールを停止する

動作に必要のないモジュールを実行状態のままにしている場合は、モジュール命令（MDSTRT/MDSTOP）を使用して、必要な時だけモジュール実行させてください。

「モジュール命令」（4-115ページ）

● 必要のないラダー回路ブロックをジャンプする

動作に必要のないラダー回路ブロックを条件分岐（CJ/NCJ～LABEL命令）、ステップ（STP～STE命令）などで囲み、必要な時だけ実行させてください。

● 繰り返し（FOR～NEXT命令間）の内容を少なくする

FOR～NEXT命令間の繰り返し実行するラダー回路ブロック内に無駄な部分があると、繰り返し回数分の処理時間がスキャンタイムに加算されます。繰り返し実行する必要のない処理は、FOR～NEXT命令の外に記述してください。

参 考　CJ/GOTO命令を使用して繰り返し処理するプログラムを作成する場合も同様です。

● 32ビット/64ビットデータを扱うオペランドに偶数番号のデバイスを指定する

サフィックスが.D/.L/.F/.DFの場合など32ビット/64ビットデータを扱う命令語のオペランドに、DM/W/TMなどの16ビットのワードデバイスを指定する場合は、偶数番号のデバイスを使用してください。奇数番号からの2ワードを使用すると、処理速度が遅くなります。

参 考　DM/TMの場合は、ローカルデバイスも同様に偶数番号を使用してください。変換時、偶数番号のローカルデバイスは必ず偶数番号のグローバルデバイスに、奇数番号のローカルデバイスは奇数番号のグローバルデバイスに割り付けられます。そのため、各プログラムの割付個数よりも多くのデバイス領域を消費する場合があります。

■ モジュール、ファンクションブロック、マクロの有効活用

モジュール、ファンクションブロック、マクロを有効に活用してプログラムを構造化するためには、ローカルデバイスや変数を最大限使用してください。以下にローカルデバイスを使用する上での注意事項を示します。

- CPUユニットや使用するユニットに割り当てられているデバイス、タッチパネルなどの外部接続機器で参照するデバイス、プロジェクト全体で共通して使用したり、モジュール間でデータの受け渡しが必要なデバイス以外は、全てローカルデバイスとして使用してください。
- ローカルデバイスは、0番を先頭にできるだけ詰めて使用してください。
- RやDMはユニットに割り当てられるため、グローバルデバイスとして使用する領域が必要です。ローカルデバイスは、MR/LRやEMを優先して使用してください。

「4-12 ローカルデバイス」(4-200ページ)
「4-5 変数」(4-55ページ)

■ インデックス修飾と間接指定の使い分け

命令語のオペランドで指定するデバイスを運転中に変更する方法として、「インデックス修飾」と「間接指定」があります。

＜インデックス修飾＞
　○指定方法がシンプルである。
　△指定するデバイスの種類を変更できない。
　△インデックスレジスタZはローカルデバイスとして使用できない。
　　→「ZPOP」「ZPUSH」命令を使用すれば、インデックスレジスタの使用可能点数を意識せずに使用できます。

＜間接指定＞
　○指定するデバイスの種類を変更できる。
　△指定方法が複雑である。(アドレスを扱うには専用命令を使用する必要がある)

デバイスの種別を変更する場合以外は、簡単に扱える「インデックス修飾」を使用されることをおすすめします。

「インデックスレジスタによる修飾」(4-44ページ)

デバッグの進め方

プログラムの誤りを取り除くことをデバッグと言います。
《KV STUDIO》の「モニタ」機能を使用すれば、実際に接続したCPUユニットの状態を確認できます。
また、モニタ中にラダープログラムの編集が可能な「オンラインエディット」機能を併用すれば、より効率よくデバッグを進められます。
CPUユニットがなくても、パソコンだけでラダープログラムの動作が確認できる「シミュレータ」機能を使用すれば、CPUユニットの動作をパソコン上でシミュレートしてデバッグを進められます。
「リプレイ」機能を使用すると、装置でトラブルが発生した際のデータを保存して、あとから《KV STUDIO》で再生できるため、トラブル要因の解析に活用できます。

変換

モニタまたはシミュレータを使用する前に、作成したラダープログラムをCPUユニットで扱える形式に変換する必要があります。

メニューから【変換(A)】→【変換(C)】を選択します。

別手順 ・ アイコンをクリック
・ Ctrl ＋ F9

参 考
- メニューから【モニタ／シミュレータ(N)】→【PLC転送→モニタモード(C)】を選択すると、変換から転送までを自動的におこない、モニタを開始します。
- メニューから【モニタ／シミュレータ(N)】→【シミュレータ(L)】を選択すると、変換を自動的におこない、シミュレートを開始します。

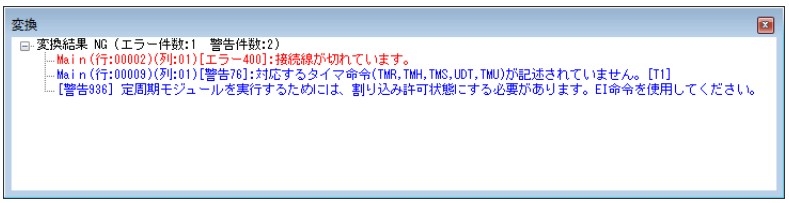
＜変換が正常に終了した場合＞

アウトプットウィンドウ[変換]に「変換結果　OK」と表示されます。
[変換結果]ダイアログにプログラム容量の使用割合が表示されます。

「プログラム容量」(4-14ページ)

＜変換が失敗した場合＞

アウトプットウィンドウ[変換]にエラー内容が表示されます。
各エラーをダブルクリックすると、該当箇所へ直接ジャンプします。1つずつ順番に修正してください。



「プログラム作成のポイント」(4-258ページ)
『KV STUDIO ユーザーズマニュアル』「付録-1 エラーメッセージ一覧」

転送

メニューから【モニタ／シミュレータ(N)】→【PLC転送(W)】を選択して、変換したラダープログラムをCPUユニットに転送します。

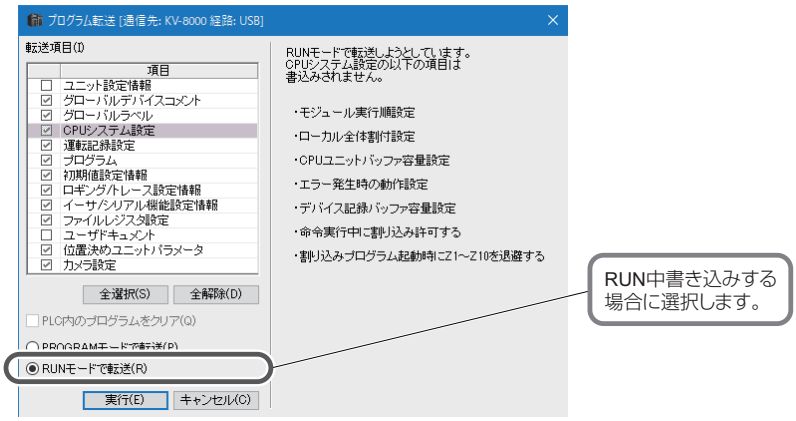
別手順 アイコンをクリック

ポイント
- **デフォルトの設定では、プログラム転送時にPLC本体に格納されている各プログラムを削除しません。使用していないプログラムを削除してプログラム転送をおこなう場合は、[プログラム転送]ダイアログの「PLC内のプログラムをクリア」にチェックを付けてください。**
- **PLC本体に格納されているプロジェクトの機能バージョンと、転送しようとしているプロジェクトの機能バージョンが異なる場合、転送前にオールクリア/初期化を実施する必要があります。**

参 考
メニューから【モニタ／シミュレータ(N)】→【PLC転送→モニタモード(C)】を選択すると、変換から転送までを自動的におこない、モニタを開始します。

■ RUN中書き込みについて

ラダープログラムを転送する際に、CPUユニットがRUN状態の場合はRUN中書き込みができます。
RUN中書き込みとは、CPUユニットがRUN状態のままラダープログラムを書き換える機能です。装置やラインを止めることなく、動作の微調整やパラメータの変更ができます。
RUN中書き込みのスキャンタイムの延びはMAX 2msです。



⚠ 注 意	**RUN中書き込みは、CPUユニットに接続されている制御機器の状態によっては危険を伴うおそれがあります。十分に注意して実行してください。**

● RUN中書き込みできないデータ

- CPUユニットに格納されているものとは異なるプロジェクト
- ユニット設定
- モジュール実行順設定
- ローカル全体割付設定
- プロテクト設定
- 命令実行中に割り込み許可する
- 割り込みプログラム起動時にZ1～Z10を退避する
- CPUユニットバッファ容量設定
- デバイス記録バッファ容量設定
- エラー発生時の動作設定

● RUN中書き込みできない条件

- ローカルデバイスの使用個数が変更された場合
- ファンクションブロックの呼び出しを追加/削除した場合※1
- ファンクションブロックの呼び出す順番を変更した場合※1
- 新たなファンクションの呼び出しを追加した場合※1
- インスタンスを追加、削除、リネームした場合
- 変数編集ウィンドウ上でインスタンスの順序を変更した場合※2
- ファンクションブロック･ファンクションの引数設定を変更した場合
- マクロ命令を追加/削除した場合
- マクロ命令の呼び出す順番を変更した場合
- マクロ引数設定を変更した場合
- 変数の追加/変更/削除やST/スクリプトの編集により自動的に確保しているローカルワーク領域を超えた場合
- プロジェクト言語設定を変更した場合

※1 CPU機能バージョン2.0以降でFBCALL/FBSTRTを使用して実行しているファンクションブロックの場合、これらの操作を行ってもRUN中書き込みできます。
※2 CPU機能バージョン2.0未満の場合は、インスタンスに対応していないため、これらの制約は発生しません。

ポイント
- **RUN中書き込み完了時に実行条件のON/OFF状態が変化しても、書き込んだ範囲内の微分命令は実行されません。**
- **タイマやカウンタの設定値を変更してRUN中書き込みしても、書き込み完了時に現在値を変更しません。**

モニタ

モニタとは、ラダーエディタで作成したラダープログラムをCPUユニットに転送し、実際に実行させて動作状況を確認するものです。

モニタを使うと、次のことができます。
- プログラムの動作状況が、ラダー図やリアルタイムチャートモニタでリアルタイムに確認できます。
- CPUユニットの運転モードが切り換えられます。
- 強制セット/リセットをすることにより、入力接点をモニタ画面上で強制ON/OFFできます。
- デバイスのデータやタイマ･カウンタの設定値や現在値を簡単に変更できます。
- デバイスの現在値などを、範囲を指定して保存／読み出しできます。
- デバイスの一部を初期値に戻せます。

モニタを開始するには、メニューから【モニタ／シミュレータ(N)】→【PLC転送→モニタモード(C)】を選択します。操作の前にCPUユニットとパソコンを接続し、通信できる状態にしておいてください。

別手順 ・ アイコンをクリック
・ Ctrl ＋ F8

参 考 《KV STUDIO》のタイトルバーに現在の状態(モニタ／オンラインエディット／エディタなど)が表示されます。

モニタの各操作は、ツールバーまたはメニューでおこないます。

各機能の詳細は『KV STUDIO ユーザーズマニュアル』「9章 モニタ／シミュレータ」を参照してください。

オンラインエディット

オンラインエディット機能を使用すると、《KV STUDIO》のモニタ画面上で直接ラダープログラムを変更できます。プログラム作成時の最終デバッグでの現場調整や簡単なプログラムを変更する時に便利な機能です。
モニタ実行中に、メニューから【モニタ／シミュレータ(N)】→【オンラインエディット開始(O)】を選択して、オンラインエディットを開始します。
編集箇所が水色に表示され、どのセルに修正を加えているかが一目で確認できます。
編集箇所の水色は転送が終了すると元の背景色に戻ります。

別手順
- モードバーで「オンラインエディット」を選択する
- アイコンをクリック
- F10

注 意	**PLCがRUN状態でオンラインエディット機能を使用する場合、使用方法を誤ると装置及び作業者に多大な被害を与える可能性があります。次に記載する注意事項を十分に確認の上、正しく使用してください。** **• 装置周辺をよく確認の上、不測の事態が発生しても人的・物的被害が起こらない状態で使用してください。** **• プログラム転送後直ちに動作する回路(実行条件がB接点の回路など)を追加しないでください。機器が暴走するおそれがあります。** **• ONしている出力命令(OUTなど)を削除して転送すると、出力がON状態で保持されます。**

参 考

オンラインエディット転送時は、変更した回路ブロックのみPLC本体に書き込みます。ただし、以下の場合は変更したプログラム全体を書き込みます。
- 複数のモジュールを変更した場合
- ローカルワーク領域を超えるようなST/スクリプト編集をおこなった場合
- ローカルワーク領域を超えるような変数の追加をおこなった場合
- ファンクションブロック内のプログラムを編集した場合(ファンクションブロック全体が転送されます。)
- マクロ内のプログラムを編集した場合(マクロ全体が転送されます。)
- ローカルデバイスの個別割当個数を変更した場合
- マクロの呼び出しを追加した場合
- ファンクションブロックの呼び出しを追加した場合
- ENDH命令を編集した場合
- 構造体を編集した場合(編集した構造体が使用されているモジュールやファンクションブロックが転送されます。)
- 複数のモジュールやファンクションブロックで使用されているグローバル変数、ファンクションブロックの引数を編集した場合(編集したグローバル変数、ファンクションブロックが使用されているモジュールおよびファンクションブロックが転送されます。)

RUN中書き込み時の制限事項や注意事項は、「RUN中書き込みについて」(4-267ページ)を参照してください。
オンラインエディット機能の詳細は、『KV STUDIO ユーザーズマニュアル』「9-9 オンラインエディット機能」を参照してください。

シミュレータ

シミュレータとは、ラダーエディタで作成したラダープログラムの動作を、PLCを接続せずにパソコン上でシミュレートするものです。
シミュレータを使うと、次のことができます。

- ラダーモニタ、登録モニタ(タイムチャート)、一括モニタを使用して、プログラムの動作状況をリアルタイムで確認できます。
- 連続スキャン／連続ステップ、1ステップ／1スキャンなど、さまざまな実行方法でシミュレートできます。
- 逆1ステップ、逆連続ステップなどの便利機能で、効率よくデバッグ作業ができます。
- タイマ、カウンタ、デバイスなどの設定値や現在値を簡単に変更できます。
- 接点の強制セットおよびリセットができます。

シミュレータを開始するには、メニューから【モニタ／シミュレータ(N)】→【シミュレータ(L)】を選択します。

別手順
- モードバーで「シミュレータ」を選択する
- アイコンをクリック
- Ctrl ＋ F2

参考 《KV STUDIO》のタイトルバーに現在の状態(シミュレータ／モニタ／エディタなど)が表示されます。

シミュレータの各操作は、ツールバーまたはメニューでおこないます。

各機能の詳細は、『KV STUDIO ユーザーズマニュアル』「9-11 シミュレータ機能－シミュレータの実行」を参照してください。

■ モニタとの違い

シミュレータでは、1スキャン実行／1ステップ実行／逆1ステップ実行など、モニタよりも細かい運転状態の制御が可能です。
CPUの動作のみシミュレートするため、拡張ユニットの動作を確認できません。また、一部のCPU内蔵機能やデバイスの動作にも制限があります。

『KV STUDIOユーザーズマニュアル』「9-11 シミュレータ機能－シミュレータでの制限」

参考 タイマの数値は、パソコンの性能によりばらつくことがあります。

シミュレータエディット

シミュレータエディット機能を使用すると、《KV STUDIO》のシミュレータ動作中に直接ラダープログラムを変更できます。プログラム作成時の最終デバッグでの現場調整や簡単なプログラムを変更する時に便利な機能です。
モニタ/オンラインエディットを実行していない状態で、メニューから【モニタ／シミュレータ(N)】→【シミュレータエディット】を選択して、シミュレータエディットを開始します。
編集箇所が水色に表示され、どのセルに修正を加えているかが一目で確認できます。
編集箇所の水色は転送が終了すると元の背景色に戻ります。

別手順 •モードバーで「シミュレータエディット」を選択する

ポイント
- **シミュレータエディットが使用できるのはKV-8000/7000シリーズCPUユニットのみです。**
- **シミュレータエディットからモニタモード、オンラインエディットモードへの移行はできません。一度エディタモードに移行したのちにモニタモード、オンラインエディットモードに移行してください。**

シミュレータエディット中に実行できる機能については「オンラインエディット」(4-269ページ)を参照してください。

リプレイ

リプレイとは、設定したトリガが成立した前後の運転記録を再生する機能です。運転記録を再生・停止・巻き戻し・コマ送りなどをしながら確認することで、トラブル発生時の状況を把握できます。CPUユニット内の運転記録を読み出すか、すでに取得した運転記録を開くかを選択してリプレイモードに移行します。

リプレイで再生できる運転記録の内容は以下のとおりです。

種類	詳細
デバイス	トリガ条件成立前後のPLCデバイス、拡張ユニットのバッファメモリの、スキャンタイムごとのデータ
カメラ	トリガ条件成立前後のフレームレートごとにKV-CA02で撮影した映像データ
イベント/エラー	トリガ条件成立前後のエラーやデバイス値変更などのイベント履歴
プロジェクトデータ	トリガ条件成立時のプロジェクトデータ

リプレイ中では以下のことができます。

- トラブル発生時のデバイスデータをラダープログラムで1スキャンごとに状態を確認したり、カメラで撮影したコマ送りの映像と照らし合わせて確認したりできます。
- デバイスの値が変化した、エラーが発生したなどのイベントを選択して運転記録を再生する位置を指定できるため、装置で異常が発生した際の状態を簡単に把握できます。
- トラブル発生時のデバイスデータをリアルタイムチャートモニタに登録し、時系列データとして表示できます。
- VT5シリーズと連携することで、トラブル発生時のデバイスデータをHMI上でモニタできます。

リプレイを開始するには、メニューから【運転記録/リプレイ(R)】→【リプレイモード(R)】を選択します。

別手順
- モードバーで「リプレイ」を選択
- メニューの【運転記録/リプレイ(R)】→【CPUユニット内の運転記録を読出→リプレイモード(O)】を選択
- メニューの【デバッグ(D)】→【イベント/エラーモニタ(E)】を選択し、〔リプレイモード〕をクリック

ポイント
- **リプレイを使用する場合、KV STUDIO Ver.10.0以降またはKV REPLAY VIEWERが必要です。**
- **リプレイを使用できるのは対応機種にKV-8000(A)を選択している場合のみです。**
- **リプレイモードからエディタモード以外には変更できません。**
- **リプレイモードの詳細は、『KV STUDIO ユーザーズマニュアル』を参照してください。**
- **リプレイモード移行時に起動しているプロジェクトと運転記録のプロジェクトが異なる場合、起動中のプロジェクトを閉じて運転記録のプロジェクトを起動するか、起動中のプロジェクトを使用して運転記録を再生するかを選択するダイアログが表示されます。**

各種モニタ機能

モニタやシミュレータを使用してデバッグを効率良く進めるために役立つ機能を説明します。
各機能の操作方法の詳細は、『KV STUDIO ユーザーズマニュアル』を参照してください。

■ デバイスの現在値や状態を変更する

● ビットデバイスのON/OFF

ビットデバイスの状態を変更するセルをダブルクリックします。ダブルクリックするごとに状態を反転します。

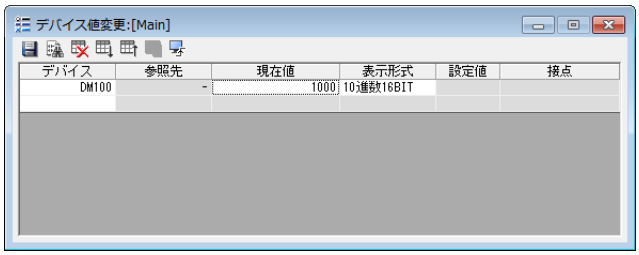
別手順 ビットデバイスの状態を変更するセルを選択した状態で、Space キーを押します。

ポイント **モニタ時は、タイマ・カウンタの接点および外部入出力リレーをON/OFFできません。**

参考 モニタ時に外部入出力リレーを強制的にON/OFFさせるには、メニューから【デバッグ(D)】の【強制セット／リセット登録(X)】を選択します。

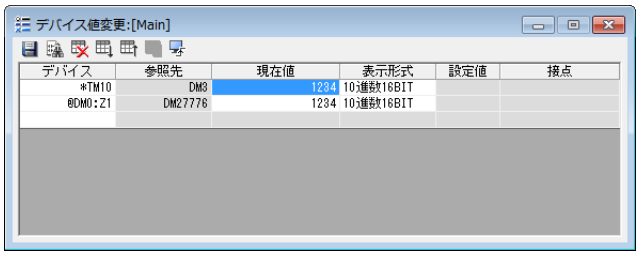
● デバイス値変更ウィンドウ

デバイスの現在値を変更するセルを選択して F2 キーを押します。



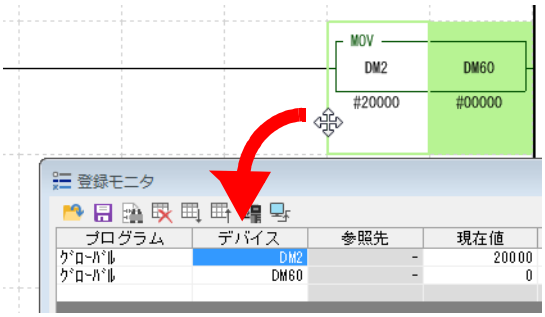
[デバイス値変更]ウィンドウが表示されます。複数のセルを選択して F2 キーを押した場合は、選択範囲内に含まれる全てのデバイスが表示されます。
現在値に値を入力すると変更できます。

参考 インデックス修飾または間接指定を使用している命令語を選択している場合は、参照しているデバイスを表示します。



● 登録モニタ／一括モニタ

登録モニタや一括モニタにデバイスを登録するには、モニタするデバイスが含まれた命令語を選択して、セルの端に合わせたカーソルが「✥」になった状態で、[登録モニタ]ダイアログ上にドラッグ&ドロップします。



参 考

- 複数のセルを選択してドラッグ & ドロップした場合は、選択範囲内に含まれる全てのデバイスが登録されます。
- [登録モニタ]ダイアログ上で直接デバイスを入力することも可能です。

● ウォッチウィンドウ

選択したセルがある回路ブロックに含まれるデバイスを全て自動的にモニタする機能です。
表示/非表示の切り換えは、メニューから【モニタ/シミュレータ(N)】→【ウォッチウィンドウの表示/非表示(Alt+3)】でおこないます。

● リアルタイムチャートモニタ

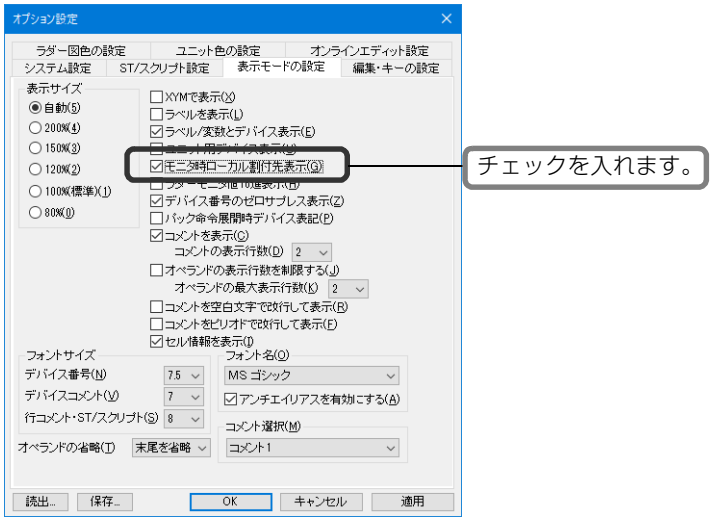
登録モニタに登録されている各デバイスの情報をタイムチャートで表示します。
登録モニタ上で右クリックして表示されるメニューから「リアルタイムチャートモニタ」を開きます。
取り込んだデータを過去にさかのぼって表示したり、選択したポイントの値を表示することもできます。

別手順 メニューから【モニタ／シミュレータ(N)】→【リアルタイムチャートモニタ(H)】を選択します。

『KV STUDIO ユーザーズマニュアル』「9-6 リアルタイムチャートモニタ」

■ ローカルデバイス／ローカルラベルの割付先で表示する

モニタやシミュレータ中に、作成したラダープログラム中のローカルデバイス／ローカルラベルを、変換時に割り付けられた割付先デバイスで表示できます。
メニューから【ツール(T)】→【オプション(O)】を選択して[オプション設定]ダイアログを開き、[表示モードの設定]タブで[モニタ時ローカル割付先表示]にチェックを入れてください。



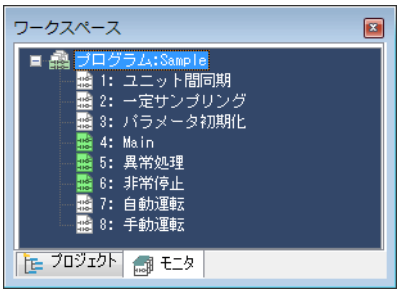
別手順 メニューから【表示】→【モニタ時ローカル割り付け先表示(W)】を選択

参 考 ローカルラベルはローカルワーク領域に割り当てられます。
「ローカルデバイス／ローカルラベルの割付先で表示する」(4-275ページ)

ポイント **変数の割付先は表示されません。**

■ プログラムの実行状態を確認する

モニタ/シミュレータ中にワークスペースの下側にある〔モニタ〕タブをクリックして表示される「プログラムモニタ」で確認できます。



■ 命令語をモニタする

ラダープログラム上に記述されたPIDAT命令に使用するデバイスをモニタします。PIDAT命令を選択した状態で右クリックして表示されるメニューから「命令モニタ(Y)」→「PIDAT(P)」を選択します。
測定値や設定値、その他PIDに使用するパラメータをモニタするのに便利です。

別手順 メニューから【モニタ/シミュレータ(N)】→【命令モニタ(Y)】→【PIDAT(P)】を選択します。
『KV STUDIOユーザーズマニュアル』「命令モニタ」

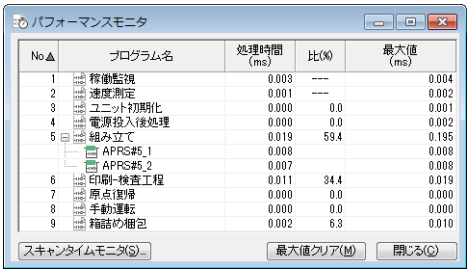
■ デバイスの立ち上がり立ち下がりをモニタする

モニタするデバイスが含まれた命令語を選択して、右クリックして表示されるメニューから「微分モニタ」を選択します。
別手順 【デバッグ(D)】→【微分モニタ(N)】を選択します。
『KV STUDIOユーザーズマニュアル』「微分モニタ」

■ プログラム単位、割り込みプログラム単位の実行時間をモニタする（パフォーマンスモニタ）

現在実行しているプログラム、割り込みプログラムの実行時間をモニタできます。

1 **モニタする場合はモニタモードに移行後、メニューから【デバッグ】→【パフォーマンスモニタ】を選択します。**

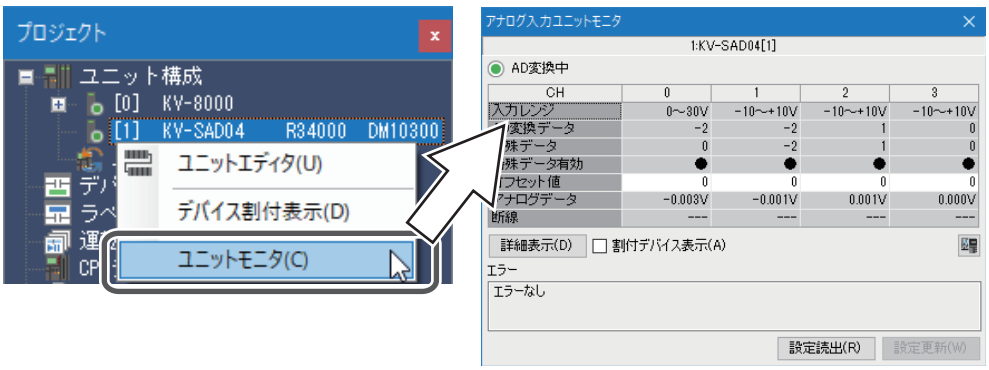


■ ユニットをモニタする

［ワークスペース］上で、「各ユニット」の右クリックメニューから【ユニットモニタ（C）】を選択すると、［ユニットモニタ］ダイアログが表示されます。

別手順
- メニューから【表示（V）】→【ユニットモニタ（J）】を選択します。
- ユニットエディタでモニタ時に、各ユニットを選択して右クリックメニューから表示します。

（例）KV-SAD04の場合



ポイント　シミュレータ時は表示できません。

MEMO

5

CPU内蔵機能

CPUユニット《KV-8000シリーズ》とアクセスウィンドウ、各通信ポート、ユーザメモリなどを使用して実現できる機能を説明します。

5-1 内蔵機能概要 5-2
5-2 メモリカード 5-3
5-3 CPUメモリ（ユーザメモリ） 5-27
5-4 レシピロード･セーブ 5-50
5-5 ロギング･トレース 5-57
5-6 運転記録機能 5-86
5-7 USB通信 5-111
5-8 イーサネット/EtherNet/IP通信 5-112
5-9 アクセスウィンドウ 5-113
5-10 ユニット間同期機能 5-169
5-11 カムスイッチ 5-182
5-12 時計機能 5-191

5-1 内蔵機能概要

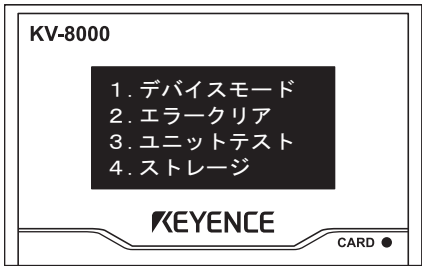
CPUユニットを使用してできる内蔵機能を説明します。

内蔵機能でできること

アクセスウィンドウ

デバイス値のモニタ
アラームの確認/復旧
メモリカード/CPUメモリの操作
ユニットモニタ

- 「アクセスウィンドウ」(5-113ページ)



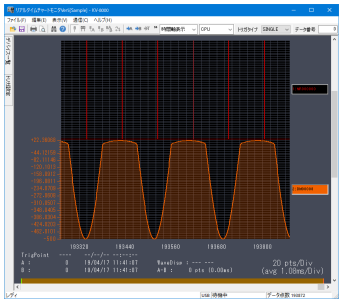
ロギング・トレース

データのロギング

- 「ロギング・トレース」(5-57ページ)

デバッグ時の効率アップ

- 「ロギング・トレース」(5-57ページ)
- リアルタイムチャートモニタ
  『KV STUDIOユーザーズマニュアル』



メモリカード/CPUメモリ

ロード/セーブ
稼働管理
ロギングデータの保存
ラダープログラムによるメモリカードおよびCPUメモリの制御
ユーザドキュメントの保存
ファイルレジスタ
運転記録機能

「メモリカード」(5-3ページ)
「CPUメモリ(ユーザメモリ)」(5-27ページ)
「5-6 運転記録機能」5-86ページ

内蔵通信ポート

パソコンとの通信

- 「USB通信」(5-111ページ)
- 「イーサネット/EtherNet/IP通信」(5-112ページ)

タッチパネルとの通信

- 「イーサネット/EtherNet/IP通信」(5-112ページ)

Ethernet・EtherNet/IP機器との通信

- 「イーサネット/EtherNet/IP通信」(5-112ページ)

5-2 メモリカード

メモリカードでできること

メモリカードを使って、ラダープログラムやコメント、および各種ユニットの設定情報をやりとりできます。また、《KV STUDIO》がなくてもメモリカードだけを使用して、CPUユニットのアクセスウィンドウやタッチパネルから、ラダープログラムの転送やデバイス値の保存、および読み出しができます。

- **メモリカードからCPUユニットへのデータ読み出し**
  アクセスウィンドウでメモリカードに保存された任意の(プロジェクト)フォルダを選択して、データを読み出せます。
  また、メモリカードに保存された「KVSAVE＊＊」(＊＊は番号:01～99)という名前の(プロジェクト)フォルダに格納しているデータを、CR5500 の OFF → ON 時に読み出せます。ドライブ番号はCM1880、フォルダ番号はCM1881で指定します。
  データはラダープログラム、デバイス、コメント、ユニット設定情報、ロギング設定情報などのプログラムデータと、DM、EM、W、FM、ZF、CM、R、MR、LR、CRなどのデバイスファイルデータがあります。
  「プロジェクトデータの保存・読み出し」(5-13ページ)

- **自動読み出し機能**
  電源投入時に、「AutoLoad」、または「AtLoad＊＊」(＊＊：01～99)という名前の(プロジェクト)フォルダに格納されているデータを自動的にCPUユニットに読み出します。また「RunLoad」、または「RnLoad＊＊」(＊＊：01～99)という名前のフォルダに保存しておくと、PRG→RUN切替時に自動的に読み出します。
  「オートローディング機能」(5-24ページ)

- **CPUユニットからメモリカードへのデータ保存**
  CPUユニットからメモリカードへ、プログラムファイル(ラダープログラム、デバイス、コメント、ユニット設定情報、ロギング設定情報)と全領域のデバイスファイル(DM、EM、W、FM、ZF、CM、R、MR、LR、B、CR)を保存できます。保存されたファイルは新規に作成された(プロジェクト)フォルダに格納されます。フォルダ名は自動的につけられ、動作正常終了時にアクセスウィンドウに表示されます。
  「プロジェクトデータの保存・読み出し」(5-13ページ)

- **レシピロード・セーブ機能**
  デバイス・変数の値をデータテーブルとして複数保存しておき、データテーブルを切り換えることでCPUユニット内のデバイス・変数の値を切り換えられます。段取り替えなどの用途に活用できます。
  「レシピロード・セーブ」(5-50ページ)

  ポイント **レシピロード・セーブ機能を使用する場合はCPU機能バージョンを2.0以降にバージョンアップしてください。**

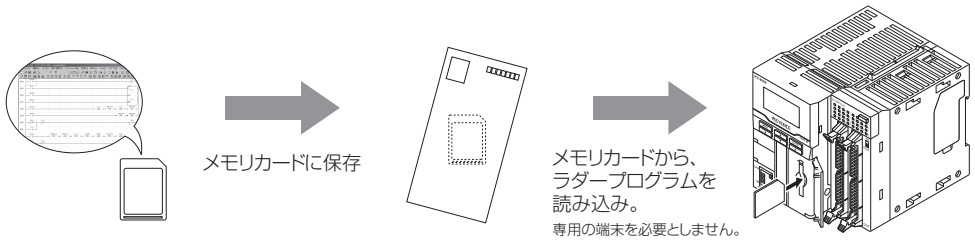
- **ロギング・トレース機能**
  記録するデバイスと取り込むタイミング(トリガ)を《KV STUDIO》で登録すれば、メモリカードにデータを自動的に格納します。CSV形式で保存されますので、Microsoft Excelなどの表計算ソフトに取り込んでそのまま使用することができます。
  「5-5 ロギング・トレース」(5-57ページ)

- **運転記録機能**
  保存するデータの種類と保存するタイミング(トリガ)を設定すると、トリガ発生時にCPUのデバイス値やカメラ入力ユニットの映像データを自動的にメモリカードに保存します。保存したデータは《KV STUDIO》のリプレイ機能で再生できます。
  「5-6 運転記録機能」(5-86ページ)

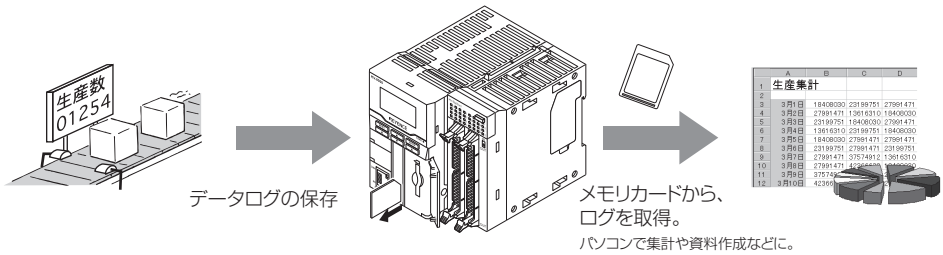
- **ラダープログラムによるメモリカード制御**
  ストレージ命令を使用すると、ラダープログラムからメモリカードにアクセスできます。
  ラダープログラムでCPUユニットのデバイスのデータ値をメモリカードに保存したり、メモリカードからデバイスのデータ値を別のCPUユニットで読み込んだりできます。メモリカードの空き容量も確認できます。
  「ストレージ命令」(5-26ページ)

● メモリカードの活用例

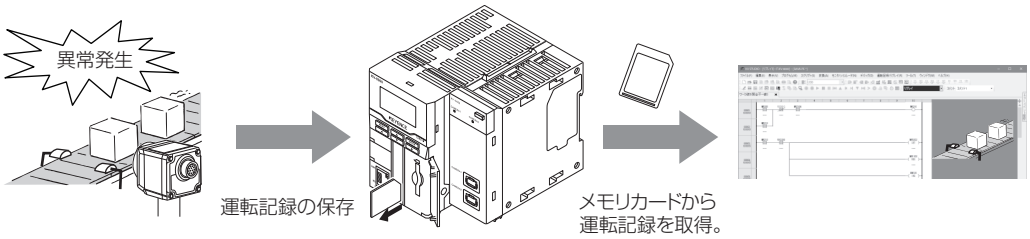
- ラダープログラムのメンテナンスが簡単。装置のメンテナンスや、仕様変更によるラダーデータの送付などにすばやく対応できます。



- デバイス値のロギングが容易。ログから集計し、資料作成が簡単になります。



- 異常要因の特定が容易。異常発生時のラダープログラムを1スキャンずつ確認したり、カメラで撮影したコマ送りの映像と照らし合わせて確認したりできるため、装置の異常要因の特定に活用できます。



■ メモリカードとCPUメモリの差分

機能	メモリカード	CPUメモリ
プログラムのロード/セーブ	○	○
レシピロード/セーブ	○	○
ファイルレジスタ	○	○
ロギング･トレース機能	○	○
運転記録機能	○	○
ラダープログラムによるファイル制御	○	○
ユーザドキュメントの格納	○	○
RAM上にファイル作成	−	○

「5-3 CPUメモリ（ユーザメモリ）」

メモリカードについて

■ 使用できるメモリカード

メモリカードと、メモリカードを読み出すアダプタやカードリーダが必要になります。メモリカードは、SDHCメモリカード、SDメモリカードに対応しています。
当社製以外のメモリカードでの動作は保証いたしません。

SD メモリカード

項目	KV-M16G	KV-M4G	KV-M1G
記憶素子	フラッシュROM		
規格	SDHC		SD
メモリ容量	16GB	4GB	1GB
書き換え回数（1ブロックあたり）	6万回		10万回
動作温度	－25～＋85℃		
保存温度	－40～＋85℃		

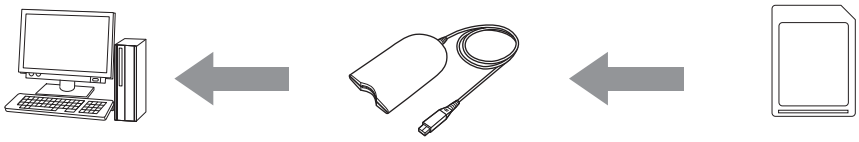
■ メモリカードのデータを読み書きする方法

メモリカードのデータを読み書きするには次の3つの方法があります。
①パソコンに直接接続して読み書きを実行する
②ストレージ転送ツールを使用してPLC経由で読み書きを実行する
③FTPサーバ機能を使用してPLC経由で読み書きを実行する

①パソコンに直接接続して読み書きを実行する
使用するパソコンにより、接続方法が異なります。

●USBポートを装備している場合

パソコンUSBポートに、USB用カードリーダライタを接続します。次に、メモリカードをカードリーダライタに接続します。



カードリーダライタ
USBポートを持つパソコンで、メモリカードのデータを読み書きするためのユニットです。
USBポートを持たないパソコンでは使用できません。
メモリカード用USBカードリーダの詳細については、カードリーダライタに添付の取扱説明書を参照してください。

●SDカードスロットを装備している場合

メモリカードをカードスロットに挿入します。

②ストレージ転送ツールを使用してPLC経由で読み書きを実行する

KV STUDIOのメニューの【ツール(T)】→【ストレージ転送ツール】を起動します。

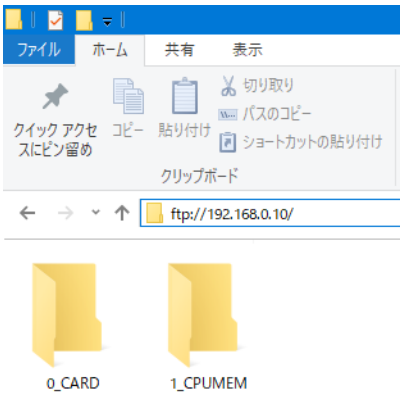
別手順　メニュー→KEYENCE KV STUDIO **→ストレージ転送ツール
(**は使用しているKV STUDIOのバージョンです)

メモリカード内のフォルダと、パソコン内のフォルダをバックアップまたはリストアします。全データの一括転送にのみ対応しており、ファイルやフォルダを指定した読み書きには対応していません。

詳細な内容については『KV STUDIOユーザーズマニュアル』を参照してください。

③FTPサーバ機能を使用してPLC経由で読み書きを実行する

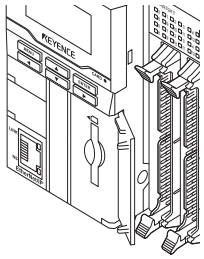
FTPサーバ機能に対応したCPUユニットやイーサネットユニットを使用すると、イーサネットケーブル経由でPLC内のメモリカードのデータを読み書きできます。
メモリカード内のデータは「0_CARD」フォルダに格納されています。



■ CPUユニットで使用する

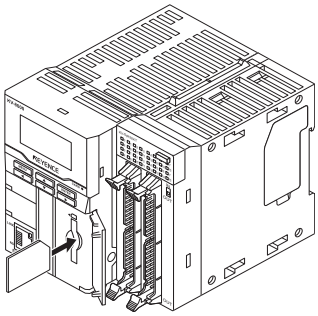
● メモリカードを装着する

1 メモリカードスロットカバーを開きます。



2 メモリカードをメモリカードスロットに差し込みます。

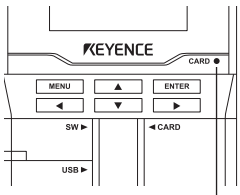
ラベル面が右側になるようにカチッという音がするまで差し込んでください。



3 メモリカードを装着した後、メモリカードスロットカバーをカチッとロックされるまで閉じます。

● メモリカードを取り出す

1 メモリカードアクセス中LEDが消えていることを確認し、メモリカードスロットカバーを開きます。



メモリカードアクセス中LED

2 メモリカードを差し込む方向(奥側)にメモリカードを押します。

カチッという音がして、メモリカードが掴める状態になったら、まっすぐにメモリカードを引き抜きます。

3 メモリカードを取り出した後、メモリカードスロットカバーをカチッとロックされるまで閉じます。

■ 使用上の注意事項

メモリカードを使用する上での一般的な注意事項を以下に記します。

注 記	• 次の場合、メモリカードに保存したデータが失われる恐れがあります。記録したデータが失われた場合は、弊社は一切その責任を負いかねます。あらかじめご了承ください。
	  • ユーザまたは第三者がメモリカードを誤った使用方法で使用したとき
	  • メモリカードが、静電気・電気的ノイズの影響を受けたとき
	  • メモリカードへの保存や消去などのアクセス中に、メモリカードを取り出したり、その他の機器の電源をOFFにしたとき。
	• 大切なデータは、CD/DVDなどの他のメディアに保存して、常にバックアップすることをおすすめします。
	• 本体のメモリカードアクセス中LEDが点灯中(カードへのデータの読み込み・書き込み)は、絶対にカードを抜いたり、本体の電源を切らないでください。カード内のデータが破損することがあります。

ポイント

- ノイズが厳しい環境では、メモリカードへのアクセスに時間がかかることがあります。(最悪の場合は、カードの破損・FATの破損などが発生することがあります。)
  その場合は、電源ケーブルにフェライトコアを巻いてご使用ください。
- また、強い電磁界を発生する機器から、離して設置してください。
- メモリカードスロットカバーが開いている状態では、メモリカードにアクセスを開始しません。必ず、メモリカードスロットカバーを閉じて使用してください。
- メモリカードアクセス中にメモリカードスロットカバーを開いた場合、その時点で行っていた処理は最後まで実行しますが、以降はメモリカードへの処理を受け付けなくなります。
- SDメモリカードは、出荷時にパフォーマンスが最大限に活用できるようにフォーマットされています。SDメモリカードをWindowsでフォーマットした場合、使用は可能ですがSDメモリカードへのデータ書き込み速度が低下するなどパフォーマンスを最大限に発揮できなくなることがあります。
  万が一、出荷時の状態にリフォーマットをする必要がある場合は、最寄りの営業所までお問い合わせください。

メモリカードで使用するデバイス

デバイス番号	属性 R:読込専用 空欄:読み書き可能	名称	内容
CR5400		メモリカードマウント要求	メモリカードアンマウントを実行したメモリカードを再度認識する際に使用します。メモリカードを挿入すると、自動的にマウントされるため、通常の挿入時は、メモリカードマウント要求を実行する必要はありません。 ※ 要求実行時はSET命令を使用してください。
CR5401		メモリカードアンマウント要求	挿入されているメモリカードをアンマウントします。アンマウントが完了すると、メモリカード認識完了リレーがOFFします。 ※ 要求実行時はSET命令を使用してください。
CR5402	R	メモリカード使用中	CPUユニットや拡張ユニットの機能を使用してメモリカードにアクセス中にONします。
CR5403	R	メモリカード認識完了	メモリカードがマウントされたらONします。アンマウントされるとOFFします。
CR5404	R	メモリカードあり	メモリカードが挿入されていることを確認したらONします。
CR5405	R	メモリカードライトプロテクト	メモリカードがライトプロテクト状態の場合ONします。
CR5500		プロジェクトロード実行要求	プロジェクトロード実行要求リレー。OFF→ONで、プロジェクトのロードを実行します(RUN中は実行できません)。
CR5502		プロジェクトセーブ実行要求	プロジェクトセーブ実行要求リレー。OFF→ONで、プロジェクトのセーブを実行します。
CR5504	R	プロジェクトロード実行完了	プロジェクトロード実行完了リレー。プロジェクトのロードが完了するとONします。プロジェクトロード実行要求リレーがOFF→ONすると、OFFします。
CR5505	R	プロジェクトロード実行失敗	プロジェクトロード実行失敗リレー。プロジェクトのロードが失敗するとONします。プロジェクトロード実行要求リレーがOFF→ONすると、OFFします。
CR5506	R	プロジェクトセーブ実行完了	プロジェクトセーブ実行完了リレー。プロジェクトのセーブが完了するとONします。プロジェクトセーブ実行要求リレーがOFF→ONすると、OFFします。
CR5507	R	プロジェクトセーブ実行失敗	プロジェクトセーブ実行失敗リレー。プロジェクトのセーブが失敗するとONします。プロジェクトセーブ実行要求リレーがOFF→ONすると、OFFします。

デバイス番号	属性 R:読込専用 空欄:読み書き可能	名称	内容
CR5508		レシピロード実行要求	レシピロード実行要求リレー。OFF→ONで、CM1870で指定したドライブの、CM1871で指定したフォルダ番号に格納しているレシピをロードして、デバイス・変数の値に上書きします。
CR5510	R	レシピロード実行完了	レシピロード実行完了リレー。レシピロードが完了するとONします。レシピロード実行要求リレーがOFF→ONすると、OFFします。
CR5511	R	レシピロード実行失敗	レシピロード実行失敗リレー。レシピロードが失敗するとONします。レシピロード実行要求リレーがOFF→ONすると、OFFします。
CR5512		レシピセーブ実行要求	レシピセーブ実行要求リレー。OFF→ONで、CM1875で指定したドライブの、CM1876で指定したフォルダ番号のレシピ内容を参照し、CPUユニットのデバイス・変数の値を使用してCM1877で指定したフォルダ番号のレシピを上書きします。
CR5514	R	レシピセーブ実行完了	レシピセーブ実行完了リレー。レシピセーブが完了するとONします。レシピセーブ実行要求リレーがOFF→ONすると、OFFします。
CR5515	R	レシピセーブ実行失敗	レシピセーブ実行失敗リレー。レシピセーブが失敗するとONします。レシピセーブ実行要求リレーがOFF→ONすると、OFFします。
CM1870		レシピロード時ドライブ番号	レシピロード時ドライブ番号。CR5508のOFF→ON時に、格納されたドライブ番号のレシピをロードします。
CM1871		レシピロード時フォルダ番号	レシピロード時ドライブ番号。CR5508のOFF→ON時に、格納されたフォルダ番号のレシピをロードします。
CM1873	R	レシピロード時完了コード	レシピロード実行完了リレーがONすると、完了コードが格納されます。
CM1875		レシピセーブ時ドライブ番号	レシピセーブ時ドライブ番号。CR5512のOFF→ON時に、格納されたドライブ番号にレシピをセーブします。
CM1876		レシピセーブ時参照フォルダ番号	レシピセーブ時参照フォルダ番号。CR5512のOFF→ON時に、格納されたフォルダ番号のレシピの内容を参照します
CM1877		レシピセーブ時出力フォルダ番号	レシピセーブ時出力フォルダ番号。CR5512のOFF→ON時に、格納されたフォルダ番号にレシピをセーブします。

デバイス番号	属性 R:読込専用 空欄:読み書き可能	名称	内容
CM1878	R	レシピセーブ時完了コード	レシピセーブ実行完了リレーがONすると、完了コードが格納されます。
CM1880		プロジェクトロード時 ドライブ番号(要求)	プロジェクトロード時ドライブ番号(要求)。CR5500のOFF→ON時に格納されたドライブ番号のプロジェクトをロードします。
CM1881		プロジェクトロード時 フォルダ番号(要求)	プロジェクトロード時フォルダ番号(要求)。CR5500のOFF→ON時に格納されたフォルダ番号のプロジェクトをロードします。
CM1882	R	プロジェクトロード時 ドライブ番号(完了)	ロードされたドライブ番号が格納されます。プロジェクトロードに失敗した場合は0が格納されます。
CM1883	R	プロジェクトロード時 フォルダ番号(完了)	ロードされたフォルダ番号が格納されます。プロジェクトロードに失敗した場合は0が格納されます。
CM1884	R	プロジェクトロード時完了コード	プロジェクトロード実行完了リレーがONすると、完了コードが格納されます。
CM1888		プロジェクトセーブ時 ドライブ番号(要求)	プロジェクトセーブ時ドライブ番号(要求)。プロジェクトセーブ実行要求リレーがOFF→ON時に指定したドライブにプロジェクトフォルダをセーブします。 0:メモリカード 1:CPUメモリ
CM1889		プロジェクトセーブ時 フォルダ番号(要求)	プロジェクトセーブ時フォルダ番号(要求)。プロジェクトセーブ実行要求リレーがOFF→ON時に指定したフォルダ番号のプロジェクトフォルダをセーブします。0,1～99で入力します。0を指定した場合、空き番号を検索してフォルダ番号とします。
CM1890	R	プロジェクトセーブ時 ドライブ番号(完了)	セーブされたドライブ番号が格納されます。プロジェクトセーブに失敗した場合は0が格納されます。
CM1891	R	プロジェクトセーブ時 フォルダ番号(完了)	セーブされたフォルダ番号が格納されます。プロジェクトセーブに失敗した場合は0が格納されます。
CM1892	R	プロジェクトセーブ時完了コード	プロジェクトセーブ実行完了リレーがONすると、完了コードが格納されます。

ポイント

- **CRにBOOLの配列型変数を割り当てた場合、配列の要素番号は数値で指定してください。デバイスや変数で指定した場合、SET命令が正しく動作しません。**

プロジェクトデータの保存 ・ 読み出し

■ ファイル構成

メモリカード内の（プロジェクト）フォルダで管理するデータは次のとおりです。

機　種	ファイルの種類		ファイル名	保存する手段
KV-8000(A)	CPUシステム設定 ラダープログラム グローバル変数 ローカル変数 ローカルデバイスコメント		□.YCD※1	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO （ファイル ＞メモリカード）
	変数情報※7		□.YVR※1	
	グローバルデバイスコメント		□.YGC※1	
	デバイス初期値設定情報		□.YDR※1	
	通信設定情報		□.YAS※1	
	ユニット設定情報		□.YLB※1	
	ロギング･トレース設定情報		□.YLG※1	
	CPUメモリ容量設定		□.YMC※1	
	運転記録設定		□.YTR※1	
	ユーザ認証設定		□.YUC※1	
	アクセスウィンドウ設定情報		□.YAW※1	
	デバイス ファイル	DM	DM.ZDV	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO （モニタ/シミュレータ ＞一括変更ウィンドウ）
		EM	EM.ZDV	
		FM	FM.ZDV※2 ※5	
		CM	CM.ZDV	
		R	R.ZDV	
		MR	MR.ZDV	
		LR	LR.ZDV	
		CR	CR.ZDV	
		B	B.ZDV	
		W	W.ZDV	
		ZF	ZF.YDV※5	
	メール設定情報		MAIL*.YSH（*=0～48）※4	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO （メール設定ツール）
	FTPクライアント設定情報		FTPC*.YSH（*=0～48）	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO （FTPクライアント設定ツール）
	OPC UAサーバ設定情報		UASVR*.YSH(*=0)	CR5502 OFF→ON時 アクセスウィンドウ KV STUDIO
	メモリカードデータ生成時 バージョン情報		Savelog.txt	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO
KV-XD02	データ活用ユニットパラメータファイル		*_XD02.YTC（*=01～16）	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO
			_XD02.YTV（=01～16）	
KV-CA02	カメラ入力ユニットパラメータファイル		*_CA02.YTC（*=01～16）	CR5502 OFF→ON時 アクセスウィンドウ KV STUDIO

機　種	ファイルの種類	ファイル名	保存する手段
KV-XLE02	イーサネットユニットパラメータファイル	*_XLE02.YTC（*=01～16）	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO
KV-XCM02	通信ユニットパラメータファイル	*_XCM02.YTC(*=01～16)	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO
KV-XL402	シリアルコミュニケーションユニットパラメータファイル	*_XL402.YTC（*=01～16）	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO
KV-XL202		*_XL202.YTC（*=01～16）	
KV-L2*V	PROTOCOL STUDIO 2 ファイル（PS1互換（標準/ラージ）モード）	PB*.ZSH（*=1A～48B）※3	KV-L2*V設定 （PROTOCOL STUDIO 2）
	PROTOCOL STUDIO 2 ファイル（プロトコルスタジオ2モード）	PB*.YSH（*=1A～48B）	CR5502　OFF→ON時 アクセスウィンドウ KV-L2*V設定 （PROTOCOL STUDIO 2）
KV-XH16ML	位置決め･モーションユニットパラメータファイル	*_XH16.YTC（*=01～16）	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO
KV-XH04ML		*_XH04.YTC（*=01～16）	
KV-XH64EC		*_XH64EC.YTC（*=01～16）	
KV-XH32EC		*_XH32EC.YTC（*=01～16）	
KV-XH16EC		*_XH16EC.YTC（*=01～16）	
KV-ML16V/MC40V/MC20V	位置決め･モーションユニットパラメータファイル	KMP*.YSH（*=1～48）	KV-ML/MC設定 （KV MOTION＋）
KV-LH20V	通信型位置決めユニットパラメータ設定（ポイントパラメータ、制御/システムパラメータ）	SPSP**.YSH（*=1～48）	通信型位置決め設定ツール ※6
KV-SH04PL	位置決めユニットパラメータファイル	*_SH.YTC（*=01～16）	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO
KV-H20G/H20S/H40S	位置決めユニット パラメータファイル	MB*.ZSH（*=1～48）	KV-H20S/40S/20G設定 （MOTION BUILDER）
	位置決めユニット コメントファイル	MC*.ZSH（*=1～48）	
MV-L40	MV-L40 パラメータファイル	MLS*.ZSH（*=1～18）	MV-L40設定 （MV LINK STUDIO）
	MV-L40 コメントファイル	MLC*.ZSH（*=1～18）	
KV-DN20	KV-DN20 パラメータファイル	DN*.ZSH（*=1～48）	KV STUDIO （KV-DN20設定ツール）
KV-CL20	KV-CL20 パラメータファイル	CL*.ZSH（*=1～48）	KV STUDIO （KV-CL20設定ツール）

※1　「□」はプロジェクトフォルダ名と同じになります。
※2　FMはカレントバンクのFMが保存され、カレントバンクのFMに読み出します。
※3　アクセスウィンドウからメモリカードやCPUメモリに保存できません。設定自体はDMやCMに保存されていますので、これらのファイルがないプロジェクトデータを読み出しても、正常に動作します。
※4　CPU内蔵イーサネット機能でメール設定を作成した場合に、「MAIL0.YSH」で保存されます。
※5　FMとZFの両方を読み出す場合、最初にZFのデータを、次にFMのデータを読み出します。
※6　アクセスウィンドウから保存した場合、生成されません。ただし、設定したポイントパラメータなどはDM.ZDVの中に格納されます。
※7　CPU機能バージョン2.0以降のKV-8000にのみ存在するデータです。

■ アクセスウィンドウを使用する

● 保存方法

CPUユニットのアクセスウィンドウ上からプロジェクトデータを保存すると、プロジェクトフォルダ「KVSAVE ＊＊」(＊＊は番号 : 01～99)を自動的に作成します。番号は、メモリカード内のフォルダ名として存在しない最小の二桁の数字になります。

「メモリカード」(5-3ページ)

● 読み出し方法

CPUユニットのアクセスウィンドウ上からプロジェクトフォルダ選択して読み出すことができます。

「メモリカード」(5-3ページ)

参 考

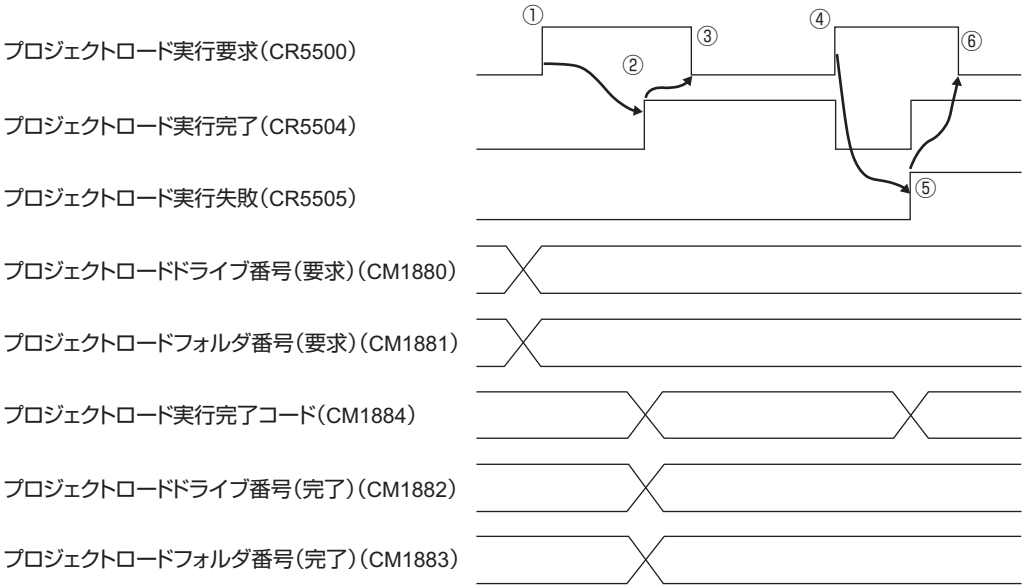
- 一部のデータのみ読み出す方法
  通常は、(プロジェクト)フォルダ内のすべてのデータを一括でメモリカードから読み出します。
  DMのデータのみ、あるいはプログラムデータのみといったように、特定のデータのみをメモリカード内のフォルダに保管しておくことで、メモリカードからCPUユニットに特定のデータのみを読み出しできます。
  メモリカードに保存されたフォルダから、パソコンで不必要なファイルを削除します。
- 電源投入時に「AutoLoad」(フォルダ指定の場合は「AtLoad＊＊」(＊＊ : 01～99))という名前の(プロジェクト)フォルダに格納されているプロジェクトデータを自動的に読み出します。
- PRG→RUN時に「RunLoad」(フォルダ指定の場合は「RnLoad＊＊」(＊＊ : 01～99))という名前の(プロジェクト)フォルダに格納されているプロジェクトデータを自動的に読み出します。

■ コントロールリレーを使用してロード/セーブする場合

● ロード

プログラムのロードはRUNモード中は実行できません。PRGモードに切り換えて、コントロールリレーを操作してください。

ロード時のタイムチャート



プロジェクトロード成功時

①プロジェクトロード時ドライブ番号(要求)とプロジェクトロード時フォルダ番号(要求)にロードしたいメモリカードデータの入ったドライブ番号、フォルダ番号をセットし、プロジェクトロード実行要求リレーをONします。(モニタや、タッチパネル、通信)

②プロジェクトロードが実行完了すると、プロジェクトロード実行完了リレーが ON します。プロジェクトロードドライブ番号(完了)、プロジェクトロードフォルダ番号(完了)にはロードしたメモリカードデータを読み出したドライブ番号とフォルダ番号が格納されます。(CPUユニット)

③プロジェクトロード実行完了リレーが ON したら、プロジェクトロード実行要求リレーを OFF します。(モニタや、タッチパネル、通信など)次回プロジェクト実行要求リレーがOFF→ONすると、プロジェクトロード実行完了リレーがOFFします。(CPUユニット)

プロジェクトロード失敗時

④プロジェクトロード時ドライブ番号(要求)とプロジェクトロード時フォルダ番号(要求)にロードしたいメモリカードデータの入ったドライブ番号、フォルダ番号をセットし、プロジェクトロード実行要求リレーをONします。(モニタや、タッチパネル、通信)

⑤プロジェクトロードが失敗すると、プロジェクトロード実行完了リレーとプロジェクトロード実行失敗リレーが ON します。プロジェクトロードドライブ番号(完了)とプロジェクトロードフォルダ番号(完了)は更新されません。(CPUユニット)完了時の結果については、プロジェクトロード時完了コード(CM1884)の内容を確認してください。完了コードの内容については、「ロード・セーブ実行時のエラーコード」(5-19ページ)を参照してください。

⑥プロジェクトロード実行失敗リレーが ON したら、プロジェクトロード実行要求リレーを OFF します。(モニタや、タッチパネル、通信など)次回プロジェクト実行要求リレーがOFF→ONすると、プロジェクトロード実行完了リレー、プロジェクトロード実行失敗リレーがOFFします。(CPUユニット)

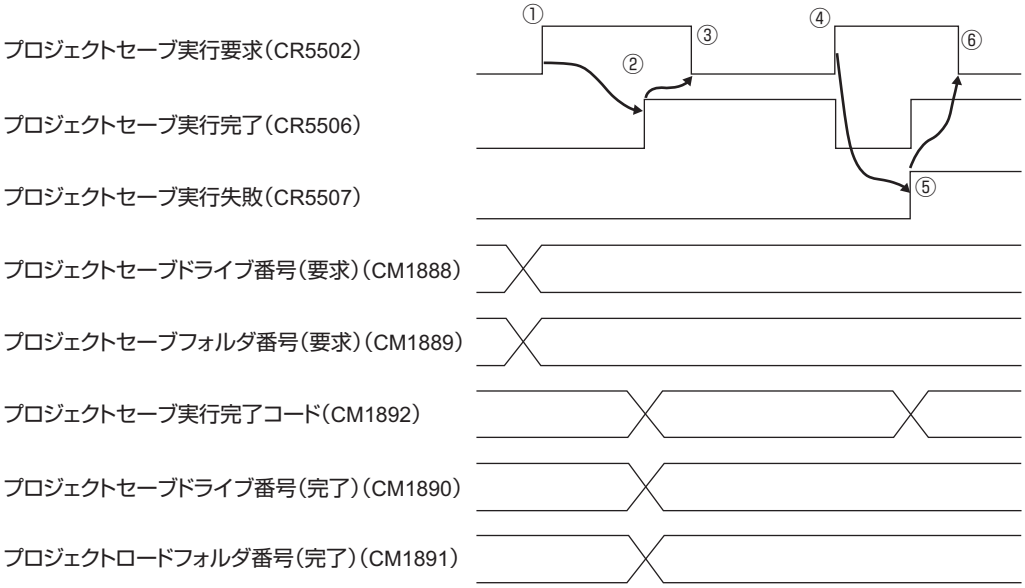
参考プログラム

プロジェクトロードは、RUNモード中は実行できないため、参考プログラムはありません。

● セーブ

プログラムのセーブを、ラダープログラムでコントロールリレーを制御しておこなう場合は、SET命令またはKEEP命令で実行してください。

セーブ時のタイムチャート



※ セーブ時にフォルダ番号を指定する場合は、プロジェクトセーブ時フォルダ番号（要求）（CM1888）を使用します。

プロジェクトセーブ成功時

①プロジェクトセーブ時ドライブ番号（要求）にセーブ先のドライブ番号を格納し、プロジェクトセーブ実行要求リレーをONします。指定したフォルダに格納したい場合にはプロジェクトセーブ時フォルダ番号（要求）にフォルダ番号を格納しておきます。（ラダーやモニタ、タッチパネル、通信など）

②プロジェクトセーブが実行完了すると、プロジェクトセーブ実行完了リレーが ON します。プロジェクトセーブ時ドライブ番号（完了）にはセーブ先のドライブ番号が、プロジェクトセーブ時フォルダ番号（完了）にはセーブ先のフォルダ番号が格納されます。（CPUユニット）

③プロジェクトセーブ実行完了リレーが ON したら、プロジェクトセーブ実行要求リレーを OFF します。（ラダーやモニタ、タッチパネル、通信など）次回プロジェクト実行要求リレーがOFF→ONすると、プロジェクトセーブ実行完了リレーがOFFします。（CPUユニット）

プロジェクトセーブ失敗時

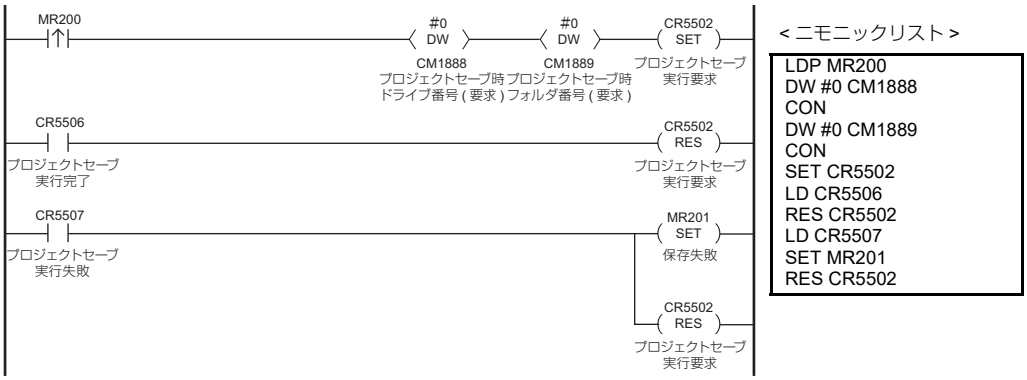
④プロジェクトセーブ時ドライブ番号(要求)にセーブ先のドライブ番号を格納し、プロジェクトセーブ実行要求リレーをONします。(ラダーやモニタ、タッチパネル、通信など)

⑤プロジェクトセーブが失敗すると、プロジェクトセーブ実行失敗リレーがONします。(CPUユニット)

⑥プロジェクトセーブ実行失敗リレーが ON したら、プロジェクトセーブ実行要求リレーを OFF します。(ラダーやモニタ、タッチパネル、通信など)次回プロジェクト実行要求リレーがOFF→ONすると、プロジェクトセーブ実行完了リレー、プロジェクトセーブ実行失敗リレーがOFFします。(CPUユニット)

参考プログラム

プロジェクトセーブは、SET命令を使用して実行してください。



ロード・セーブ実行時のエラーコード

オートロード完了コードはCM1791、ランロード完了コードはCM1795に、プロジェクトロード時完了コードはCM1884、プロジェクトセーブ時完了コードはCM1892に格納されます。

番号	メッセージ	原因	対処	発生時
2	空き番号がありません Unused number not available 没有空编号	既にファイルがあったため書き込めませんでした。	保存したいフォルダ番号に空き番号を設定してください。	セーブ
3	オートローディング失敗 Auto Loading Failed 自动加载失败	オートロードまたはランロード実行時に、CMで指定したフォルダ番号が範囲外でした。	0～99の番号を指定してください。	ロード/セーブ
4	ロードできません Cannot load data 无法加载	メモリカード/CPUメモリに指定したCMのフォルダがありません。	プロジェクトデータが入っていることを確認してください。	ロード
5	メモリカードがありません ---------- CPUメモリアンマウント Memory card not found ---------- CPU Memory unmounted 没有存储卡 ---------- CPU存储器卸载	メモリカード/CPUメモリを認識できない状態でロードまたはセーブを実行しました。	メモリカードからロード、またはメモリカードへセーブする場合は、メモリカードを挿入してください。CPUメモリをマウントしてください。	ロード/セーブ
6	アクセス競合 Access conflict 访问冲突	他の経路からプログラムの転送または読出しをしていたため、セーブ・ロードができませんでした。	プログラムの転送もしくは読み出しが終わってからセーブ・ロードを実行してください。	ロード/セーブ
7	オートローディング失敗 Auto Loading Failed 自动加载失败	ライトプロテクト/プロジェクトロード禁止が設定されているPLCに対して、オートロードまたはランロードを実行しました。	PLCのライトプロテクト/プロジェクトロード禁止を解除してから実行してください。	ロード
9	CPUユニット機種不一致 PLC model mismatch CPU单元机型不一致	別機種のメモリカードファイルを読み込もうとしました。	プロジェクトのCPUユニット機種を確認してください。	ロード
10	非対応のCPU機能バージョン Unsupported Function Version 不支持的功能版本	対応していない機能バージョンまたは別機種のプロジェクトを読み込もうとしました。	プロジェクトのCPUユニットの機能バージョンと機種を確認してください。	ロード
11	PRGモード時のみ可能です PRG mode only 仅PRG模式时可执行	RUNモードでロードしようとしました。	PRGモードに切り換えてロードしてください。	ロード
12	空き容量がありません No free space 没有可用空间	メモリカード/CPUメモリの空き容量がありません。	空き容量を増やしてください。	ロード/セーブ
15	プロジェクト領域サイズオーバー Project area size over 项目区域容量超出	CPUメモリ容量設定の書き込みに失敗しました(プロジェクト領域が不足しています)。	CPUメモリ容量設定の設定を見直すか、CPU内のプロジェクト領域のデータを削減してください。	ロード
16	ユーザメモリ領域サイズオーバー UserMem area size over 用户存储器区域容量超出	CPUメモリ容量設定の書き込みに失敗しました(ユーザメモリ領域が不足しています)。	CPUメモリ容量設定の設定を見直すか、CPU内のユーザメモリ領域のデータを削減してください。	ロード

番号	メッセージ	原因	対処	発生時
17	スイッチがPRG時のみ可能です Only when switch in PRG 仅开关处于PRG时可执行	スイッチ状態がRUNの時に、システムプログラムバージョンアップを実行しようとしました。	スイッチをPRGにしてから、システムプログラムバージョンアップを実行してください。	ロード
18	ユニット種別不一致 Unit Type　Mismatch 单元类型不一致	接続されているユニットと、一致していないユニットのYTCファイルをロードしようとしました。	YTCファイルのユニット種別を確認してください。	ロード
19	オールクリアが必要 All Clear Reqred Before loading 加载前需全部清空	事前にオールクリア/初期化が必要なプロジェクトをロードしようとしました。PLC内部のプロジェクトとロードするプロジェクトのバージョンが一致しない場合、ロード実行前にオールクリア/初期化を実行する必要があります。	オールクリア/初期化実行後にロードを実行してください。CPUメモリからロードを実行する場合は、ロードするデータもクリアされるため、事前にバックアップを実施してください。	ロード
20	ユニット非対応 機能バージョン Unsupported Unit Function Version 不支持的单元功能版本	拡張ユニットが対応していない機能バージョンのセーブデータを読み込もうとした。	プロジェクトの拡張ユニット機能バージョンを確認してください。拡張ユニットの機能バージョンを確認してください。	ロード
100	メモリカードカバーオープン Memory card cover open 存储卡外盖打开	メモリカードカバーが開いています。	メモリカードカバーを閉じてください。	ロード/セーブ
101	メモリカードがLOCKされています Memory card locked 存储卡被锁定	メモリカードがLOCKされています。	LOCKを解除するか、書込み可能なメモリカードを使用してください。	セーブ
102	ロードできません ---------- セーブできません Cannot load data ---------- Cannot save data 无法加载 ---------- 无法保存	メモリカードが破損しています。	メモリカードのファイルシステムをチェックしてください。	ロード ---------- セーブ
103	ファイル数上限を超過しました Max no. of files exceeded 超过了文件数上限	ユーザメモリ内のファイル数とフォルダ数の合計が上限を超過しています。	ユーザメモリ内のファイルまたはフォルダを削減してください。	ロード/セーブ
104	フォルダ階層が深すぎます Folder layer too deep 文件夹层数过多	ユーザメモリ内のフォルダ階層が深すぎます。	ユーザメモリ内のフォルダ階層を確認してください。	ロード/セーブ

番号	メッセージ	原因	対処	発生時
1000	ロードできません	• ファイルが破損しています。 • メモリカードが破損しています。 • ロード処理中にカードが抜かれました。 • 「ユーザメモリをクリアしてからロードする」の設定で、CPUメモリからロードしようとしました。 • 「オールクリアしてからロードする」の設定で、オートロード/ランロード以外の手段やCPUメモリからロードを実行しようとしました。	プロジェクトやメモリカードが適切かどうか確認してください。	ロード
	Cannot load data			
	无法加载			
1001	セーブできません	• メモリカードが破損しています。 • セーブ処理中にカードが抜かれました。	プロジェクトやメモリカードが適切かどうか確認してください。	セーブ
	Cannot save data			
	无法保存			

参 考　ロード･セーブ時の完了コードはCMの他にアクセスウィンドウでも確認できます。

■ KV STUDIOを使用する

KV STUDIOのメニューから、メモリカードからプロジェクトデータの読み出し･保存ができます。

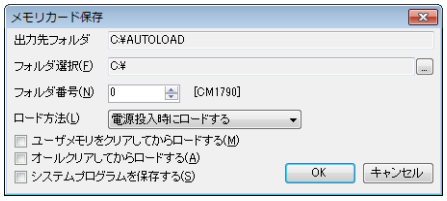
● ラダープログラムなど(デバイスファイル以外)

1 **メモリカードをパソコンに接続します。**

「メモリカードのデータを読み書きする方法」(5-6ページ)

2 **KV STUDIOのメニューから【ファイル(F)】→【メモリカード(M)】→【保存(W)】または【読出(R)】を選択します。**

保存を選択した場合

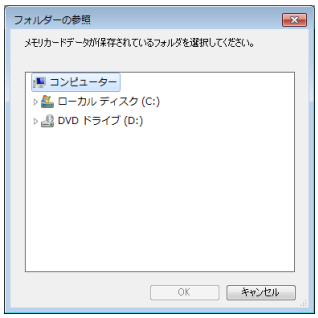


項目	説明
出力先フォルダ	メモリカードデータの出力先フォルダ名称を表示します。フォルダ選択、フォルダ番号、ロード方法などによりフォルダ名称は異なります。
フォルダ選択	メモリカードデータを出力するフォルダを選択します。右端の ... をクリックすると、[フォルダの参照]ダイアログボックスが表示されます。
フォルダ番号	フォルダ番号を指定します。 0入力時　：末尾に番号はつきません。 1～99入力時：フォルダの末尾に指定した番号を2桁で追加します。 右端のデバイス番号は、各ロード方法の場合にフォルダ番号を指定するデバイスを表示しています。 電源投入時にロードする　：　CM1790 PRG→RUN時にロードする：　CM1794
ロード方法	ロード方法を選択します。選択したロード方法により出力先のフォルダ名称は異なります。 電源投入時にロードする　：　「AUTOLOAD」,「AtLoad**」(**:01～99) PRG→RUN時にロードする：　「RUNLOAD」,「RnLoad**」(**:01～99)
ユーザメモリをクリアしてからロードする	チェックすると、ユーザメモリをクリアしてからロードします。
オールクリアしてからロードする	チェックすると、オールクリアしてからロードします。
システムプログラムを保存する	プロジェクトと一緒にユニットのシステムプログラムを保存します。ロード時にシステムプログラムの更新を実行します。「システムプログラムアップデートについて」(3-24ページ)

ポイント　**同一の名称のフォルダを選択した場合、データは上書きされて、以前のデータは失われます。**

読出を選択した場合

[フォルダーの参照]ダイアログボックスが表示されます。

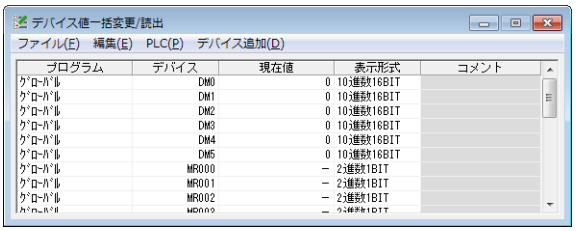


● デバイスファイル

1 **メモリカードをパソコンに接続します。**

「メモリカードのデータを読み書きする方法」(5-6ページ)

2 **KV STUDIOのメニューから【モニタ/シミュレータ(N)】→【デバイス値一括変更/読出(D)】を選択します。**



3 **デバイス値一括変更/読出ウィンドウのメニューから、【ファイル(F)】→【メモリカード読出(R)】または【メモリカード保存(Z)】を選択します。**

読出 : 読み出すドライブとフォルダ名を指定します。

保存 : 登録されているデバイスと現在値が保存されます。

参 考　KV STUDIOのデバイス初期値設定を使用すれば、同様にデバイス値をメモリカードで管理できます。

オートローディング機能

電源投入時やPRG→RUN時にメモリカードに保存されているプロジェクトデータを自動的に読み出します。
コントロールメモリ(CM)を使用して、複数のフォルダから読み出すフォルダを選択することができます。

■ 電源投入時に読み出す

1 《KV STUDIO》を使用して、「AutoLoad」、または「AtLoad＊＊」(＊＊：01～99)のフォルダ名称でメモリカードにプロジェクトデータを書き込みます。

「ラダープログラムなど(デバイスファイル以外)」(5-22ページ)

2 CPUユニットのメモリカードスロットにメモリカードを挿入し、メモリカードスロットカバーをカチッとロックされるまで閉じます。

3 CPUユニットの電源を投入します。

自動的にデータが読み出されます。

● 読み出すフォルダの指定方法

読み出すフォルダを指定する場合は、あらかじめCM1790に数値を格納します。

- CM1790の値が0の場合
  フォルダ名「AutoLoad」が読み出されます。
  「AutoLoad」フォルダがメモリカードに存在しない場合は、オートロードされません。電源投入前のプログラムで動作します。エラーは発生しません。

- CM1790の値が01～99の場合
  フォルダ名「AtLoad＊＊」(＊＊：CM1790の値(01～99))が読み出されます。

フォルダ番号の指定手順

1 電源投入している状態で、タッチパネルなどの外部機器からCM1790に数値を格納します。
2 電源を切ります。
3 電源を投入します。
4 CM1790で指定したフォルダが読み出されます。
5 読み出し完了後、CM1790に0が格納され、CM1792に読み出したフォルダ番号が格納されます。

参考

オートロード完了後、CM1790に0が格納されるので、次回電源投入時、同じフォルダ番号は読み出されません。ただし、CM1790に0が格納されている状態でメモリカードに「AutoLoad」フォルダがあると、次回電源投入時に読み出されるので注意してください。
オートロード実行時の状態はオートロード完了コード(CM1791)で確認できます。完了コードの内容については、「ロード･セーブ実行時のエラーコード」(5-19ページ)を参照してください。

ポイント

- **《KV STUDIO》のCPUシステム設定でライトプロテクト/プロジェクトロード禁止されている場合はエラー(オートローディング失敗)が発生し、PRGモードになります。エラークリア後、RUNモードで動作するとオートロード実行前のプログラムで動作 します。**
  以下の場合はエラー(オートローディング失敗)が発生し、PRGモードになります。
  - **CM1790の値が100以上の場合**
  - **CM1790で指定したフォルダがメモリカードにない場合**
- **PLC本体に格納されているプロジェクトの機能バージョンと転送するプロジェクトの機能バージョンが異なる場合、CPUのオールクリア/初期化が必要になります。**

■ PRG→RUN時に読み出す

1 **《KV STUDIO》を使用して、「RunLoad」、または「RnLoad＊＊」(＊＊：01～99)のフォルダ名称でメモリカードにプロジェクトデータを書き込みます。**

「ラダープログラムなど(デバイスファイル以外)」(5-22ページ)

2 **CPUユニットのメモリカードスロットにメモリカードを挿入し、メモリカードスロットカバーをカチッとロックされるまで閉じます。**

3 **CPUユニットをPRG→RUNに切り換えます。**

自動的にデータが読み出され、運転を開始します。

● 読み出すフォルダの指定方法

読み出すフォルダを指定する場合は、あらかじめCM1794に数値を格納します。

- CM1794の値が0の場合
  フォルダ名「RunLoad」が読み出されます。
  「RunLoad」フォルダがメモリカードに存在しない場合は、オートロードされません。PRG→RUN前のプログラムで動作します。エラーは発生しません。

- CM1794の値が01～99の場合
  フォルダ名「RnLoad＊＊」(＊＊はCM1794の値01～99)が読み出されます。

フォルダ番号の指定手順

1 タッチパネルなどの外部機器からCM1794に数値を格納します。
2 PRG→RUNに切り換えます。
3 CM1794で指定したフォルダが読み出されます。
4 読み出し完了後、CM1794に0が格納され、CM1796に読み出したフォルダ番号が格納されます。

参考

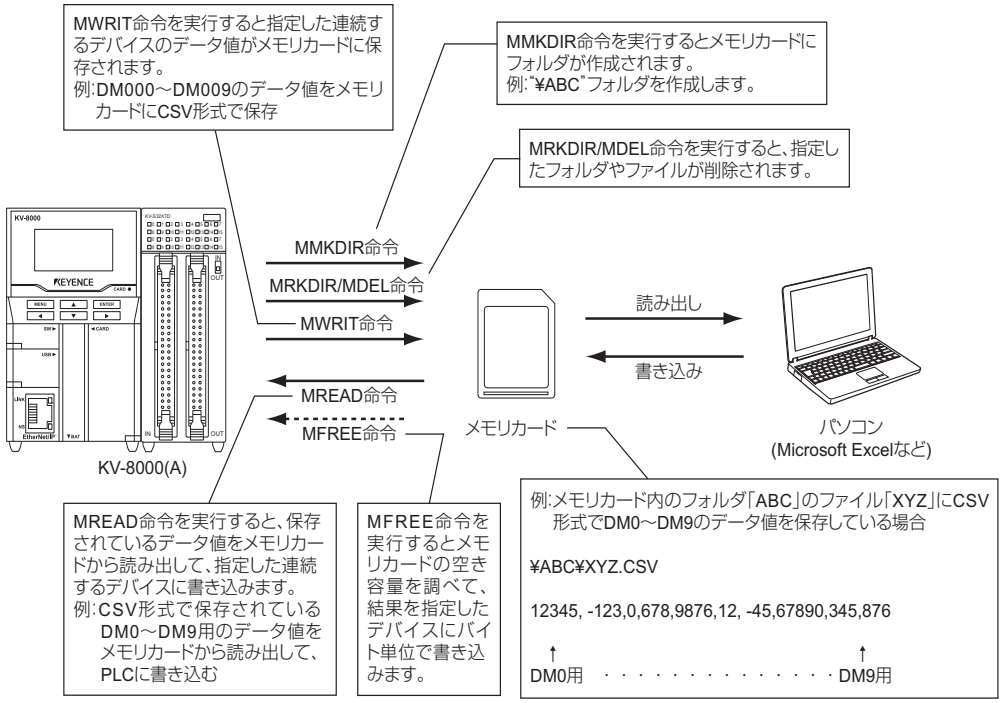
オートロード完了後、CM1794に0が格納されるので、次回電源投入時、同じフォルダ番号は読み出されません。ただし、CM1794に0が格納されている状態でメモリカードに「RunLoad」フォルダがあると次回PRG→RUNすると読み出されるので注意してください。
ランロード実行時の状態はランロード完了コード(CM1795)で確認できます。完了コードの内容については、「ロード･セーブ実行時のエラーコード」(5-19ページ)を参照してください。

ポイント

- **《KV STUDIO》のCPUシステム設定でライトプロテクト/プロジェクトロード禁止されている場合はエラー(オートローディング失敗)になりRUNモードにできません。エラークリア後、メモリカードを抜いてRUNモードで動作するとオートロード実行前のプログラムで動作します。**
  以下の場合はエラー(オートローディング失敗)になりRUNモードにできません。エラークリア後、メモリカードを抜いてRUNモードで動作するとオートロード実行前のプログラムで動作します。
  - **CM1794の値が100以上の場合**
  - **CM1794で指定したフォルダがメモリカードにない場合**
- **PLC本体に格納されているプロジェクトの機能バージョンと転送するプロジェクトの機能バージョンが異なる場合、CPUのオールクリア/初期化が必要になります。**

ストレージ命令

ストレージ命令を使用すると、ラダープログラムでデバイスのデータを読み書きできます。



参考

- ストレージ命令はメモリカードのほかにCPUメモリに対しても読み書きできます。
- ストレージ命令のエラーコードは CM2390 に格納されます。CM2390 の内容については、「KV-8000/7000/5000/3000/1000/KV Nanoシリーズ命令語リファレンスマニュアル」を参照してください。

5-3 CPUメモリ（ユーザメモリ）

CPUメモリ（ユーザメモリ）でできること

内蔵されたCPUメモリ（ユーザメモリ）領域を使用して、ラダープログラムのロード/セーブやデバイス値のロギング、ラダープログラムからファイルの作成などを実現できます。

ポイント　**メモリカードを使用しても同様のことが実現できます。**

- **CPUメモリからCPUユニットへのデータ読み出し**
  CPUメモリに保存された任意の（プロジェクト）フォルダをアクセスウィンドウで選択して、データを読み出すことができます。
  データはラダープログラム、デバイス、コメント、ユニット設定情報、ロギング設定情報などのプログラムデータと、DM、EM、W、FM、ZF、CM、R、MR、LR、CRなどのデバイスファイルデータがあります。
  「プロジェクトデータの保存・読み出し」（5-35ページ）

- **自動読み出し機能**
  電源投入時に、「AutoLoad」、または「AtLoad＊＊」（＊＊：01～99）という名前の（プロジェクト）フォルダに格納されているデータを自動的にCPUユニットに読み出します。また「RunLoad」、または「RnLoad＊＊」（＊＊：01～99）という名前のフォルダに保存しておくと、PRG→RUN切換時に自動的に読み出します。
  「オートローディング機能」（5-45ページ）

- **CPUユニットからCPUメモリへのデータ保存**
  CPUユニットからCPUメモリへ、プログラムファイル（ラダープログラム、デバイス、コメント、ユニット設定情報、ロギング設定情報）と全領域のデバイスファイル（DM、EM、W、FM、ZF、CM、R、MR、LR、B、CR）を保存できます。保存されたファイルは新規に作成された（プロジェクト）フォルダに格納されます。フォルダ名は自動的につけられ、動作正常終了時にアクセスウィンドウに表示されます。
  「プロジェクトデータの保存・読み出し」（5-35ページ）

- **レシピロード・セーブ機能**
  デバイス・変数の値をデータテーブルとして複数保存しておき、データテーブルを切り換えることでCPUユニット内のデバイス・変数の値を切り換えられます。段取り替えなどの用途に活用できます。
  「レシピロード・セーブ」（5-50ページ）

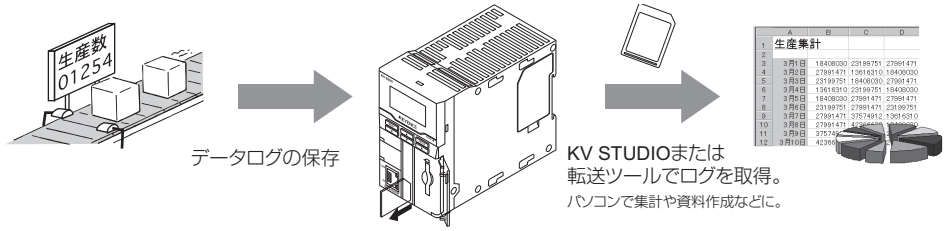
  ポイント　**レシピロード・セーブ機能を使用する場合はCPU機能バージョンを2.0以降にバージョンアップしてください。**

- **ロギング・トレース機能**
  記録するデバイスと取り込むタイミング（トリガ）を《KV STUDIO》で登録すれば、CPUメモリにデータを自動的に格納します。CSV形式で保存されますので、Microsoft Excelなどの表計算ソフトに取り込んでそのまま使用することができます。
  「5-5 ロギング・トレース」5-57ページ

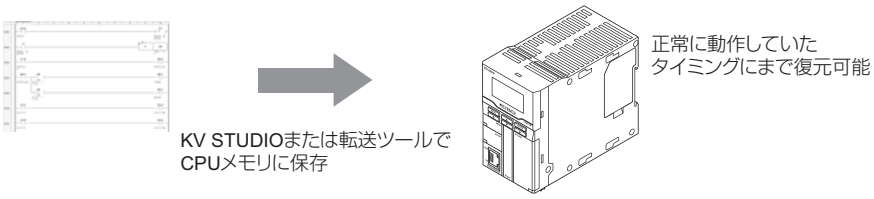
- **運転記録機能**
  保存するデータの種類と保存するタイミング（トリガ）を設定すると、トリガ発生時にCPUのデバイス値やカメラ入力ユニットの映像データを自動的にCPUメモリに保存します。保存したデータは《KV STUDIO》のリプレイ機能で再生できます。
  「5-6 運転記録機能」5-86ページ

● CPUメモリの活用例

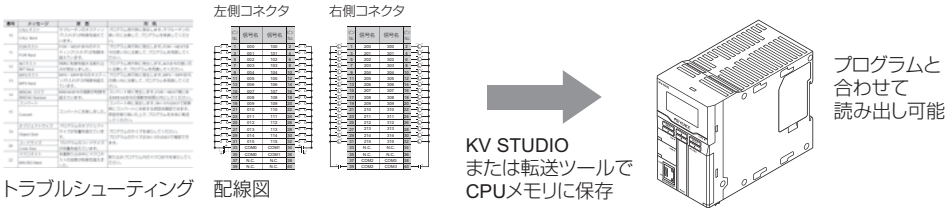
- デバイス値のロギングがメモリカードレスで実現可能。ログから集計し、資料作成が簡単になります。



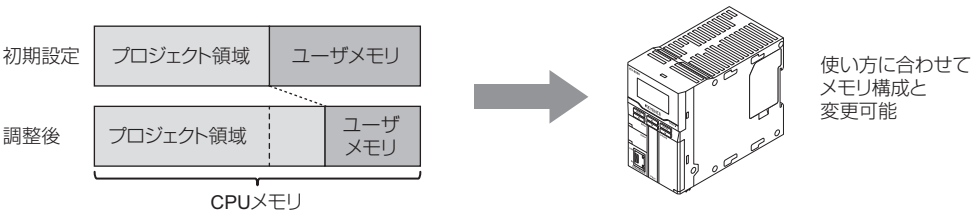
- 正常に動作していた時点のプログラムデータを保存しておくことで、変更後にトラブルが発生した場合でも簡単に現状復帰できます。



- ExcelやWord、PDFなどで作成した設計資料をプログラムと一緒に保存でき、保全作業時に設計と比較して確認できます。



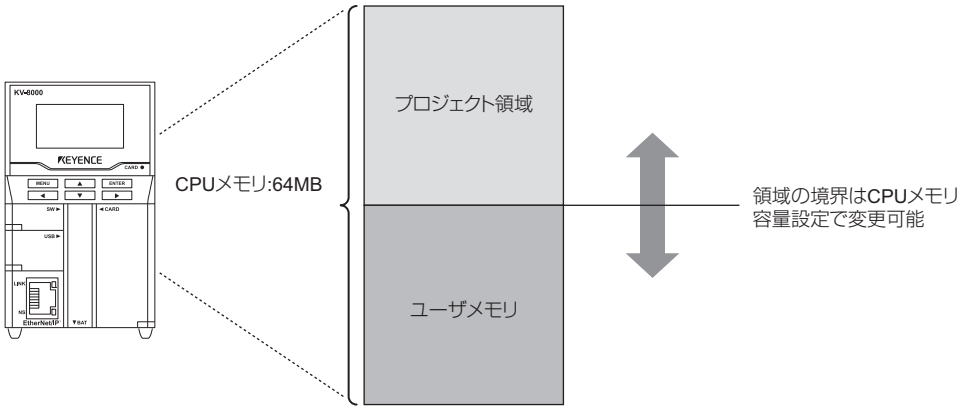
- デバイスコメントや変数を格納する領域として活用できます。



CPUメモリについて

■ 使用できるCPUメモリ領域

プログラム領域とユーザメモリ領域を足し合わせたものがCPUメモリ領域です。



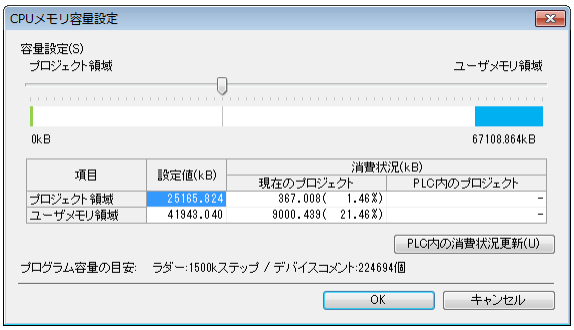
各領域に格納されるデータは以下のとおりです。

プロジェクト領域	ユーザメモリ領域
ユニット設定情報 デバイス初期値設定 CPUシステム設定 運転記録設定 ユーザ認証設定 ロギング･トレース設定 プログラム デバイスコメント 変数 ラダーモニタデータ	ファイルレジスタ ユーザドキュメント ロギングデータ プロジェクトデータ その他

ユニット設定/ユニットデータ/アクセスウィンドウ設定などは別領域です。

■ 設定方法

KV STUDIOの【ツール】→【CPUメモリ容量設定】から領域を変更できます。



■ CPUメモリ(ユーザメモリ)のデータを読み書きする方法

CPUメモリ(ユーザメモリ)のデータを読み書きするには次の3つの方法があります。

①ストレージ転送ツールを使用してPLC経由で読み書きを実行する
②サーバ機能を使用してPLC経由で読み書きを実行する
③KV STUDIOからPLC転送機能を使用して読み書きを実行する※
(※ユーザドキュメント、ファイルレジスタ設定のみ)

①ストレージ転送ツールを使用してPLC経由で読み書きを実行する

KV STUDIOのメニューの【ツール(T)】→【ストレージ転送ツール】を起動します。

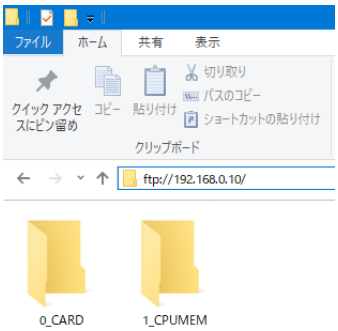
別手順　メニュー→すべてのプログラム→KEYENCE Applications→KV STUDIO Ver.8→ツール→ストレージ転送ツール

CPUメモリ(ユーザメモリ)内のフォルダと、パソコン内のフォルダをバックアップまたはリストアします。
全データの一括転送にのみ対応しており、ファイルやフォルダを指定した読み書きには対応していません。

詳細な内容については『KV STUDIOユーザーズマニュアル』を参照してください。

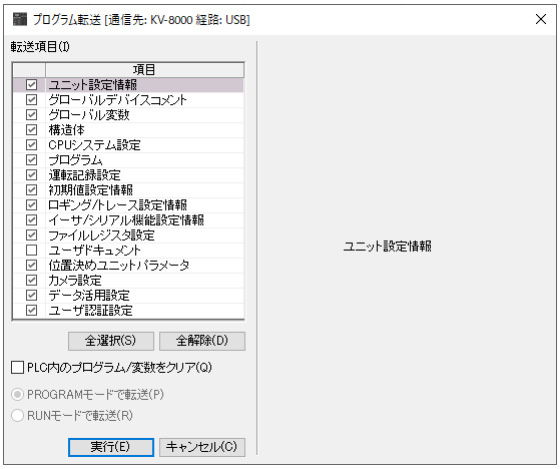
②FTPサーバ機能を使用してPLC経由で読み書きを実行する

FTPサーバ機能に対応したCPUユニットやイーサネットユニットを使用すると、イーサネットケーブル経由でPLC内のCPUメモリのデータを読み書きできます。
CPUメモリ内のデータは「1_CPUMEM」フォルダに格納されています。



③KV STUDIOからPLC転送機能を使用して読み書きを実行する

PLC転送時にユーザドキュメントとファイルレジスタの設定のみCPUメモリのデータを読み書きできます。



ポイント　**PLC転送時にロギングデータやロード・セーブデータなどのデータは転送できません。**

■ 使用上の注意事項

CPUメモリを使用するうえでの一般的な注意事項を以下に記します。

注 記	• **次の場合、CPUメモリに保存したデータが失われる恐れがあります。記録したデータが失われた場合は、弊社は一切その責任を負いかねます。あらかじめご了承ください。** 　• **ユーザまたは第三者がCPUメモリを誤った使用方法で使用したとき** 　• **CPUメモリが、静電気・電気的ノイズの影響を受けたとき** 　• **CPUメモリへの保存回数が1万回を超えたとき(PLC本体エラー143:CPU累積書込回数オーバー　が発生します)**

ポイント

- **メモリカードアクセス中にCPUメモリへの書き込みは実行できます。(メモリカードアクセス中のメモリカード書き込み、CPUメモリアクセス中のCPUメモリ書き込みは実行できません。)**
- **ユーザメモリに常時書き込みを実行する場合、書き込み速度に制限がかかります。高速での読み書きが必要になる場合、RAM上にファイルを作成し、電源断の直前に書き込みを行うことで高速に書き込みができます。「RAMモード」(5-47ページ)**
- **「隠しファイル属性」や「読み取り専用属性」がついたファイル/フォルダをユーザメモリに書き込んだ場合、これらの属性は解除されます。**

CPUメモリで使用するデバイス

デバイス番号	属性 R:読込専用 空欄:読み書き可能	名称	内容
CR3200		CPUメモリ保存要求	ONするとRAM上の差分データをCPUメモリに保存します。 ※ 要求実行時はSET命令を使用してください。
CR3201	R	CPUメモリ保存中	CPUメモリに保存中にONします。
CR3202	R	CPUメモリ保存完了	CPUメモリ保存が完了するとONします。CPUメモリ保存要求をONするとOFFします。
CR3203	R	CPUメモリ保存エラー	CPUメモリ保存時にエラーが発生するとONします。CPUメモリ保存要求をOFFするとOFFします。
CR3207		CPUメモリ命令語RAMモード	ON中、ファイルレジスタ命令またはストレージ命令がRAMディスク上で動作します。
CR5408		CPUメモリマウント要求	CPUメモリアンマウントを実行した後に再度認識する際に使用します。通常の電源投入時は、自動的にマウントされるため、CPUメモリマウント要求を実行する必要はありません。 ※ 要求実行時はSET命令を使用してください。
CR5409		CPUメモリアンマウント要求	CPUメモリをアンマウントします。アンマウントが完了すると、CPUメモリ認識完了リレーがOFFします。 ※ 要求実行時はSET命令を使用してください。
CR5410	R	CPUメモリ使用中	CPUメモリにアクセス中にONします。
CR5411	R	CPUメモリ認識完了	CPUメモリがマウントされたらONします。アンマウントされるとOFFします。
CR5500		プロジェクトロード実行要求	プロジェクトロード実行要求リレー。OFF→ONで、プロジェクトのロードを実行します(RUN中は実行できません)。
CR5502		プロジェクトセーブ実行要求	プロジェクトセーブ実行要求リレー。OFF→ONで、プロジェクトのセーブを実行します。
CR5504	R	プロジェクトロード実行完了	プロジェクトロード実行完了リレー。プロジェクトのロードが完了するとONします。プロジェクトロード実行要求リレーがOFF→ONすると、OFFします。
CR5505	R	プロジェクトロード実行失敗	プロジェクトロード実行失敗リレー。プロジェクトのロードが失敗するとONします。プロジェクトロード実行要求リレーがOFF→ONすると、OFFします。
CR5506	R	プロジェクトセーブ実行完了	プロジェクトセーブ実行完了リレー。プロジェクトのセーブが完了するとONします。プロジェクトセーブ実行要求リレーがOFF→ONすると、OFFします。

デバイス番号	属性 R:読込専用 空欄:読み書き可能	名称	内容
CR5507	R	プロジェクトセーブ実行失敗	プロジェクトセーブ実行失敗リレー。プロジェクトのセーブが失敗するとONします。プロジェクトセーブ実行要求リレーがOFF→ONすると、OFFします。
CR5508		レシピロード実行要求	レシピロード実行要求リレー。OFF→ONで、CM1870で指定したドライブの、CM1871で指定したフォルダ番号に格納しているレシピをロードして、デバイス・変数の値に上書きします。
CR5510	R	レシピロード実行完了	レシピロード実行完了リレー。レシピロードが完了するとONします。レシピロード実行要求リレーがOFF→ONすると、OFFします。
CR5511	R	レシピロード実行失敗	レシピロード実行失敗リレー。レシピロードが失敗するとONします。レシピロード実行要求リレーがOFF→ONすると、OFFします。
CR5512		レシピセーブ実行要求	レシピセーブ実行要求リレー。OFF→ONで、CM1875で指定したドライブの、CM1876で指定したフォルダ番号のレシピ内容を参照し、CPUユニットのデバイス・変数の値を使用してCM1877で指定したフォルダ番号のレシピを上書きします。
CR5514	R	レシピセーブ実行完了	レシピセーブ実行完了リレー。レシピセーブが完了するとONします。レシピセーブ実行要求リレーがOFF→ONすると、OFFします。
CR5515	R	レシピセーブ実行失敗	レシピセーブ実行失敗リレー。レシピセーブが失敗するとONします。レシピセーブ実行要求リレーがOFF→ONすると、OFFします。
CM1870		レシピロード時ドライブ番号	レシピロード時ドライブ番号。CR5508のOFF→ON時に、格納されたドライブ番号のレシピをロードします。
CM1871		レシピロード時フォルダ番号	レシピロード時ドライブ番号。CR5508のOFF→ON時に、格納されたフォルダ番号のレシピをロードします。
CM1873	R	レシピロード時完了コード	レシピロード実行完了リレーがONすると、完了コードが格納されます。
CM1875		レシピセーブ時ドライブ番号	レシピセーブ時ドライブ番号。CR5512のOFF→ON時に、格納されたドライブ番号にレシピをセーブします。
CM1876		レシピセーブ時参照フォルダ番号	レシピセーブ時参照フォルダ番号。CR5512のOFF→ON時に、格納されたフォルダ番号のレシピの内容を参照します
CM1877		レシピセーブ時出力フォルダ番号	レシピセーブ時出力フォルダ番号。CR5512のOFF→ON時に、格納されたフォルダ番号にレシピをセーブします。
CM1878	R	レシピセーブ時完了コード	レシピセーブ実行完了リレーがONすると、完了コードが格納されます。

デバイス番号	属性 R:読込専用 空欄:読み書き可能	名称	内容
CM1880		プロジェクトロード時 ドライブ番号(要求)	プロジェクトロード時ドライブ番号(要求)。CR5500のOFF→ON時に格納されたドライブ番号のプロジェクトをロードします。
CM1881		プロジェクトロード時 フォルダ番号(要求)	プロジェクトロード時フォルダ番号(要求)。CR5500のOFF→ON時に格納されたフォルダ番号のプロジェクトをロードします。
CM1882	R	プロジェクトロード時 ドライブ番号(完了)	ロードされたドライブ番号が格納されます。プロジェクトロードに失敗した場合は0が格納されます。
CM1883	R	プロジェクトロード時 フォルダ番号(完了)	ロードされたフォルダ番号が格納されます。プロジェクトロードに失敗した場合は0が格納されます。
CM1884	R	プロジェクトロード時完了コード	プロジェクトロード実行完了リレーがONすると、完了コードが格納されます。
CM1888		プロジェクトセーブ時 ドライブ番号(要求)	プロジェクトセーブ時ドライブ番号(要求)。プロジェクトセーブ実行要求リレーがOFF→ON時に指定したドライブにプロジェクトフォルダをセーブします。 0:メモリカード 1:CPUメモリ
CM1889		プロジェクトセーブ時 フォルダ番号(要求)	プロジェクトセーブ時フォルダ番号(要求)。プロジェクトセーブ実行要求リレーがOFF→ON時に指定したフォルダ番号のプロジェクトフォルダをセーブします。0,1～99で入力します。0を指定した場合、空き番号を検索してフォルダ番号とします。
CM1890	R	プロジェクトセーブ時 ドライブ番号(完了)	セーブされたドライブ番号が格納されます。プロジェクトセーブに失敗した場合は0が格納されます。
CM1891	R	プロジェクトセーブ時 フォルダ番号(完了)	セーブされたフォルダ番号が格納されます。プロジェクトセーブに失敗した場合は0が格納されます。
CM1892	R	プロジェクトセーブ時完了コード	プロジェクトセーブ実行完了リレーがONすると、完了コードが格納されます。

参考　ロード･セーブ時の完了コードはCM1884、CM1892の他にアクセスウィンドウでも確認できます。

ポイント
- **CRにBOOLの配列型変数を割り当てた場合、配列の要素番号は数値で指定してください。デバイスや変数で指定した場合、SET命令が正しく動作しません。**

プロジェクトデータの保存・読み出し

■ ファイル構成

CPUメモリ内の(プロジェクト)フォルダで管理するデータは次のとおりです。

機　種	ファイルの種類		ファイル名	保存する手段
KV-8000(A)	CPUシステム設定 ラダープログラム グローバル変数 ローカル変数 ローカルデバイスコメント		□.YCD※1	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO (ファイル ＞メモリカード)
	変数情報※7		□.YVR※1	
	グローバルデバイスコメント		□.YGC※1	
	デバイス初期値設定情報		□.YDR※1	
	通信設定情報		□.YAS※1	
	ユニット設定情報		□.YLB※1	
	ロギング･トレース設定情報		□.YLG※1	
	CPUメモリ容量設定		□.YMC※1	
	運転記録設定		□.YTR※1	
	ユーザ認証設定		□.YUC※1	
	アクセスウィンドウ設定情報		□.YAW※1	
	デバイス ファイル	DM	DM.ZDV	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO (モニタ/シミュレータ ＞一括変更ウィンドウ)
		EM	EM.ZDV	
		FM	FM.ZDV※2 ※5	
		CM	CM.ZDV	
		R	R.ZDV	
		MR	MR.ZDV	
		LR	LR.ZDV	
		CR	CR.ZDV	
		B	B.ZDV	
		W	W.ZDV	
		ZF	ZF.YDV※5	
	メール設定情報		MAIL*.YSH(*=0～48)※4	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO (メール設定ツール)
	FTPクライアント設定情報		FTPC*.YSH(*=0～48)	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO (FTPクライアント設定ツール)
	OPC UAサーバ設定情報		UASVR*.YSH(*=0)	CR5502 OFF→ON時 アクセスウィンドウ KV STUDIO
	メモリカードデータ生成時 バージョン情報		Savelog.txt	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO
KV-XD02	データ活用ユニットパラメータファイル		*_XD02.YTC(*=01～16)	CR5502　OFF→ON時 アクセスウィンドウ KV STDUIO
			_XD02.YTV(=01～16)	
KV-CA02	カメラ入力ユニットパラメータファイル		*_CA02.YTC(*=01～16)	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO

機　種	ファイルの種類	ファイル名	保存する手段
KV-XLE02	イーサネットユニットパラメータファイル	*_XLE02.YTC(*=01～16)	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO
KV-XCM02	通信ユニットパラメータファイル	*_XCM02.YTC(*=01～16)	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO
KV-XL402	シリアルコミュニケーションユニットパラメータファイル	*_XL402.YTC(*=01～16)	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO
KV-XL202		*_XL202.YTC(*=01～16)	
KV-L2*V	PROTOCOL STUDIO 2 ファイル(PS1互換(標準/ラージ)モード)	PB*.ZSH(*=1A～48B)※3	KV-L2*V設定 (PROTOCOL STUDIO 2)
	PROTOCOL STUDIO 2 ファイル(プロトコルスタジオ2モード)	PB*.YSH(*=1A～48B)	CR5502　OFF→ON時 アクセスウィンドウ KV-L2*V設定 (PROTOCOL STUDIO 2)
KV-XH16ML	位置決め･モーションユニットパラメータファイル	*_XH16.YTC(*=01～16)	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO
KV-XH04ML		*_XH04.YTC(*=01～16)	
KV-XH64EC		*_XH64EC.YTC(*=01～16)	
KV-XH32EC		*_XH32EC.YTC(*=01～16)	
KV-XH16EC		*_XH16EC.YTC(*=01～16)	
KV-ML16V/MC40V/MC20V	位置決め･モーションユニットパラメータファイル	KMP*.YSH(*=1～48)	KV-ML/MC設定 (KV MOTION＋)
KV-LH20V	通信型位置決めユニットパラメータ設定(ポイントパラメータ、制御/システムパラメータ)	SPSP**.YSH(*=1～48)	通信型位置決め設定ツール ※6
KV-SH04PL	位置決めユニットパラメータファイル	*_SH.YTC(*=01～16)	CR5502　OFF→ON時 アクセスウィンドウ KV STUDIO
KV-H20G/H20S/H40S	位置決めユニット パラメータファイル	MB*.ZSH(*=1～48)	KV-H20S/40S/20G設定 (MOTION BUILDER)
	位置決めユニット コメントファイル	MC*.ZSH(*=1～48)	
MV-L40	MV-L40 パラメータファイル	MLS*.ZSH(*=1～18)	MV-L40設定 (MV LINK STUDIO)
	MV-L40 コメントファイル	MLC*.ZSH(*=1～18)	
KV-DN20	KV-DN20 パラメータファイル	DN*.ZSH(*=1～48)	KV STUDIO (KV-DN20設定ツール)
KV-CL20	KV-CL20 パラメータファイル	CL*.ZSH(*=1～48)	KV STUDIO (KV-CL20設定ツール)

※1　「□」はプロジェクトフォルダ名と同じになります。
※2　FMはカレントバンクのFMが保存され、カレントバンクのFMに読み出します。
※3　アクセスウィンドウからメモリカードやCPUメモリに保存できません。設定自体はDMやCMに保存されていますので、これらのファイルがないプロジェクトデータを読み出しても、正常に動作します。
※4　CPU内蔵イーサネット機能でメール設定を作成した場合に、「MAIL0.YSH」で保存されます。
※5　FMとZFの両方を読み出す場合、最初にZFのデータを、次にFMのデータを読み出します。
※6　アクセスウィンドウから保存した場合、生成されません。ただし、設定したポイントパラメータなどはDM.ZDVの中に格納されます。
※7　CPU機能バージョン2.0以降のKV-8000にのみ存在するデータです。

■ アクセスウィンドウを使用する

● 保存方法

CPUユニットのアクセスウィンドウ上からプロジェクトデータを保存すると、プロジェクトフォルダ「KVSAVE ＊＊」(＊＊は番号 : 01～99)を自動的に作成します。番号は、CPUメモリ内のフォルダ名として存在しない最小の二桁の数字になります。

「ストレージ」(5-141ページ)

● 読み出し方法

CPUユニットのアクセスウィンドウ上からプロジェクトフォルダ選択して読み出すことができます。

「ストレージ」(5-141ページ)

参 考

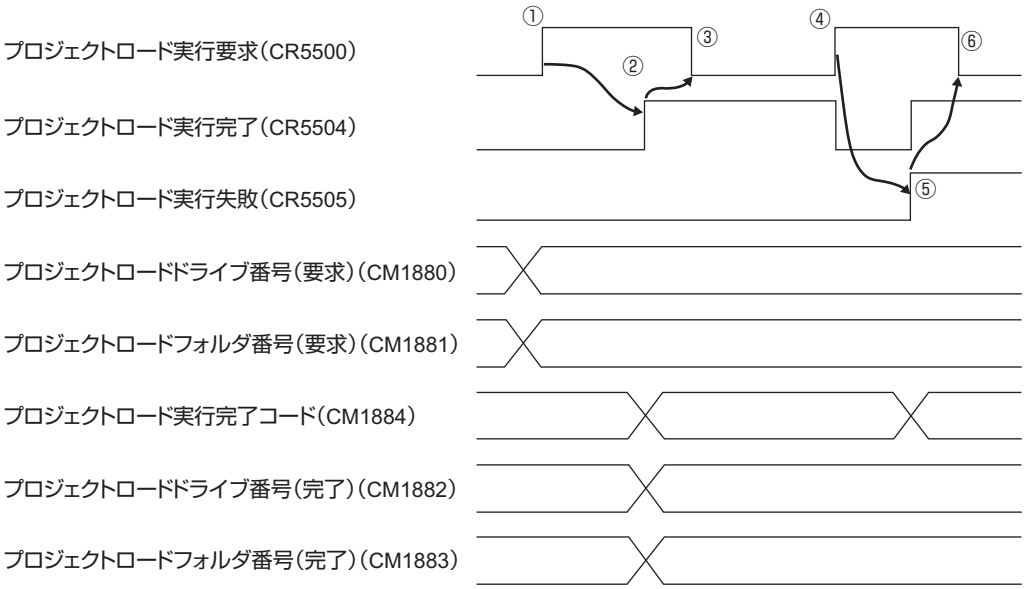
- 一部のデータのみ読み出す方法
  通常は、(プロジェクト)フォルダ内のすべてのデータを一括でCPUメモリから読み出します。
  読み出したいファイルのみを残し、そのほかのファイルを削除することで、一部のデータのみ読み出すことができます。
  ストレージ転送ツールを使用する場合は、パソコン上で不要なファイルを削除してから転送してください。
- 電源投入時に「AutoLoad」(フォルダ指定の場合は「AtLoad＊＊」(＊＊ : 01～99))という名前の(プロジェクト)フォルダに格納されているプロジェクトデータを自動的に読み出します。
- PRG➔RUN時に「RunLoad」(フォルダ指定の場合は「RnLoad＊＊」(＊＊ : 01～99))という名前の(プロジェクト)フォルダに格納されているプロジェクトデータを自動的に読み出します。

■ コントロールリレーを使用してロード/セーブする場合

● ロード

プログラムのロードはRUNモード中は実行できません。PRGモードに切り換えて、コントロールリレーを操作してください。

ロード時のタイムチャート



プロジェクトロード成功時

①プロジェクトロード時ドライブ番号(要求)とプロジェクトロード時フォルダ番号(要求)にロードしたいプロジェクトデータの入ったドライブ番号、フォルダ番号をセットし、プロジェクトロード実行要求リレーをONします。(モニタや、タッチパネル、通信)

②プロジェクトロードが実行完了すると、プロジェクトロード実行完了リレーが ON します。プロジェクトロードドライブ番号(完了)、プロジェクトロードフォルダ番号(完了)にはロードしたプロジェクトデータを読み出したドライブ番号とフォルダ番号が格納されます。(CPUユニット)

③プロジェクトロード実行完了リレーが ON したら、プロジェクトロード実行要求リレーを OFF します。(モニタや、タッチパネル、通信など)次回プロジェクト実行要求リレーがOFF→ONすると、プロジェクトロード実行完了リレーがOFFします。(CPUユニット)

プロジェクトロード失敗時

④プロジェクトロード時ドライブ番号(要求)とプロジェクトロード時フォルダ番号(要求)にロードしたいメモリカードの入ったドライブ番号、フォルダ番号をセットし、プロジェクトロード実行要求リレーをONします。(モニタや、タッチパネル、通信など)

⑤プロジェクトロードが失敗すると、プロジェクトロード実行完了リレーとプロジェクトロード実行失敗リレーが ON します。プロジェクトロードドライブ番号(完了)とプロジェクトロードフォルダ番号(完了)は更新されません。(CPUユニット)

⑥プロジェクトロード実行失敗リレーが ON したら、プロジェクトロード実行要求リレーを OFF します。(モニタや、タッチパネル、通信)次回プロジェクト実行要求リレーがOFF→ONすると、プロジェクトロード実行完了リレー、プロジェクトロード実行失敗リレーがOFFします。(CPUユニット)

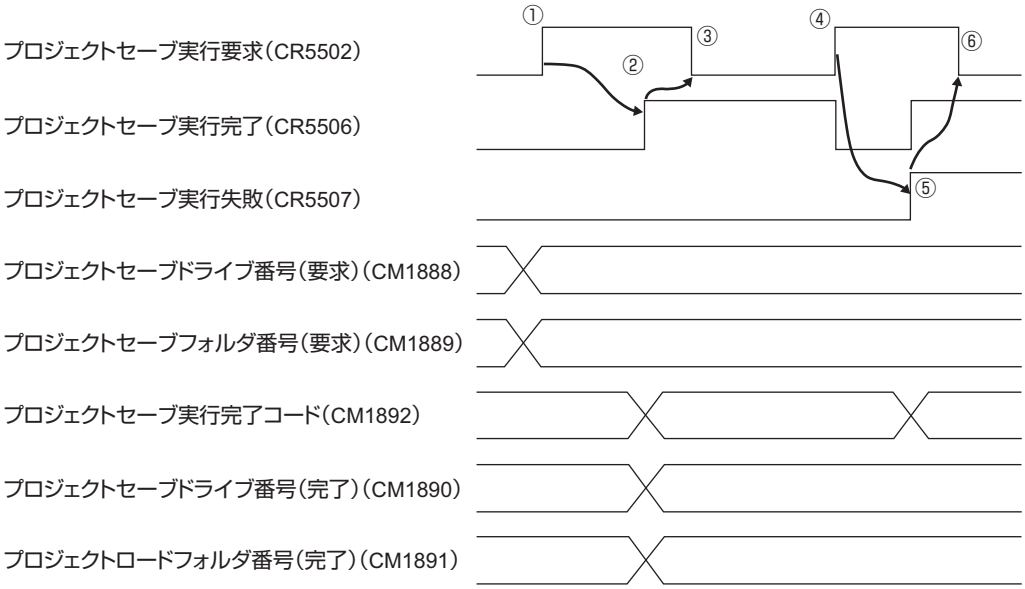
参考プログラム

プロジェクトロードは、RUNモード中は実行できないため、参考プログラムはありません。

● セーブ

プログラムのセーブを、ラダープログラムでコントロールリレーを制御しておこなう場合は、SET命令またはKEEP命令で実行してください。

セーブ時のタイムチャート



※ セーブ時にフォルダ番号を指定する場合は、プロジェクトセーブ時フォルダ番号(要求)(CM1888)を使用します。

プロジェクトセーブ成功時

①プロジェクトセーブ時ドライブ番号(要求)にセーブ先のドライブ番号を格納し、プロジェクトセーブ実行要求リレーをONします。指定したフォルダに格納したい場合にはプロジェクトセーブ時フォルダ番号(要求)にフォルダ番号を格納しておきます。(ラダーやモニタ、タッチパネル、通信など)

②プロジェクトセーブが実行完了すると、プロジェクトセーブ実行完了リレーが ON します。プロジェクトセーブ時ドライブ番号(完了)にはセーブ先のドライブ番号が、プロジェクトセーブ時フォルダ番号(完了)にはセーブ先のフォルダ番号が格納されます。(CPUユニット)

③プロジェクトセーブ実行完了リレーが ON したら、プロジェクトセーブ実行要求リレーを OFF します。(ラダーやモニタ、タッチパネル、通信など)次回プロジェクト実行要求リレーがOFF→ONすると、プロジェクトセーブ実行完了リレーがOFFします。(CPUユニット)

プロジェクトセーブ失敗時

④プロジェクトセーブ時ドライブ番号(要求)にセーブ先のドライブ番号を格納し、プロジェクトセーブ実行要求リレーをONします。指定したフォルダに格納したい場合にはプロジェクトセーブ時フォルダ番号(要求)にフォルダ番号を格納しておきます。(ラダーやモニタ、タッチパネル、通信など)

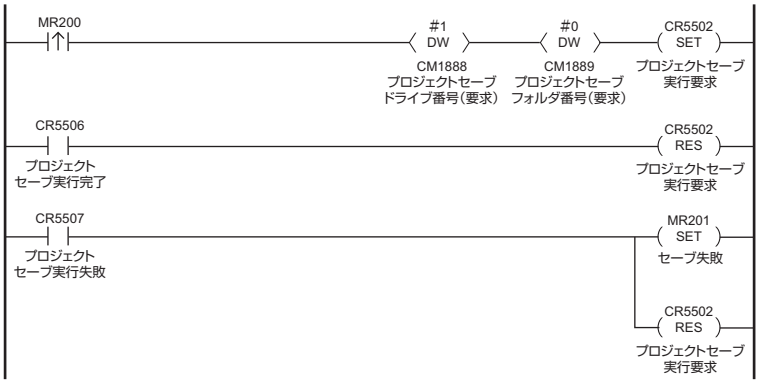
⑤プロジェクトセーブが失敗すると、プロジェクトセーブ実行失敗リレーがONします。(CPUユニット)

⑥プロジェクトセーブ実行失敗リレーが ON したら、プロジェクトセーブ実行要求リレーを OFF します。(ラダーやモニタ、タッチパネル、通信など)次回プロジェクト実行要求リレーがOFF→ONすると、プロジェクトセーブ実行完了リレー、プロジェクトセーブ実行失敗リレーがOFFします。(CPUユニット)

参考プログラム

MR200の立ち上がりでCPUメモリ(ドライブ番号:1)にプロジェクトをセーブするプログラムです。プロジェクトセーブフォルダ番号(要求)に0を格納すると、空き番号を検索してフォルダ番号とします。

プロジェクトセーブは、SET命令を使用して実行してください。



< ニモニックリスト >

```
LDP MR200
DW #1 CM1888
CON
DW #0 CM1889
CON
SET CR5502
LD CR5506
RES CR5502
LD CR5507
SET MR201
RES CR5502
```

ロード・セーブ実行時のエラーコード

オートロード完了コードはCM1791、ランロード完了コードはCM1795に、プロジェクトセーブ時完了コードはCM1892に格納されます。

番号	メッセージ	原因	対処	発生時
2	空き番号がありません Unused number not available 没有空编号	既にファイルがあったため書き込めませんでした。	保存したいフォルダ番号に空き番号を設定してください。	セーブ
3	オートローディング失敗 Auto Loading Failed 自动加载失败	オートロードまたはランロード実行時に、CMで指定したフォルダ番号が範囲外でした。	0～99の番号を指定してください。	ロード/セーブ
4	ロードできません Cannot load data 无法加载	メモリカード/CPUメモリに指定したCMのフォルダがありません。	プロジェクトデータが入っていることを確認してください。	ロード
5	メモリカードがありません ---------- CPUメモリアンマウント Memory card not found ---------- CPU Memory unmounted 没有存储卡 ---------- CPU存储器卸载	メモリカード/CPUメモリを認識できない状態でロードまたはセーブを実行しました。	メモリカードからロード、またはメモリカードへセーブする場合は、メモリカードを挿入してください。CPUメモリをマウントしてください。	ロード/セーブ
6	アクセス競合 Access conflict 访问冲突	他の経路からプログラムの転送または読出しをしていたため、セーブ・ロードができませんでした。	プログラムの転送もしくは読み出しが終わってからセーブ・ロードを実行してください。	ロード/セーブ
7	オートローディング失敗 Auto Loading Failed 自动加载失败	ライトプロテクト/プロジェクトロード禁止が設定されているPLCに対して、オートロードまたはランロードを実行しました。	PLCのライトプロテクト/プロジェクトロード禁止を解除してから実行してください。	ロード
9	CPUユニット機種不一致 PLC model mismatch CPU单元机型不一致	別機種のメモリカードファイルを読み込もうとしました。	プロジェクトのCPUユニット機種を確認してください。	ロード
10	非対応のCPU機能バージョン Unsupported Function Version 不支持的功能版本	対応していない機能バージョンまたは別機種のプロジェクトを読み込もうとしました。	プロジェクトのCPUユニットの機能バージョンと機種を確認してください。	ロード
11	PRGモード時のみ可能です PRG mode only 仅PRG模式时可执行	RUNモードでロードしようとしました。	PRGモードに切り換えてロードしてください。	ロード
12	空き容量がありません No free space 没有可用空间	メモリカード/CPUメモリの空き容量がありません。	空き容量を増やしてください。	ロード/セーブ
15	プロジェクト領域サイズオーバー Project area size over 项目区域容量超出	CPUメモリ容量設定の書き込みに失敗しました(プロジェクト領域が不足しています)。	CPUメモリ容量設定の設定を見直すか、CPU内のプロジェクト領域のデータを削減してください。	ロード
16	ユーザメモリ領域サイズオーバー UserMem area size over 用户存储器区域容量超出	CPUメモリ容量設定の書き込みに失敗しました(ユーザメモリ領域が不足しています)。	CPUメモリ容量設定の設定を見直すか、CPU内のユーザメモリ領域のデータを削減してください。	ロード

番号	メッセージ	原因	対処	発生時
17	スイッチがPRG時のみ可能です Only when switch in PRG 仅开关处于PRG时可执行	スイッチ状態がRUNの時に、システムプログラムバージョンアップを実行しようとしました。	スイッチをPRGにしてから、システムプログラムバージョンアップを実行してください。	ロード
18	ユニット種別不一致 Unit Type　Mismatch 单元类型不一致	接続されているユニットと、一致していないユニットのYTCファイルをロードしようとしました。	YTCファイルのユニット種別を確認してください。	ロード
19	オールクリアが必要 All Clear Reqred Before loading 加载前需全部清空	事前にオールクリア/初期化が必要なプロジェクトをロードしようとしました。PLC内部のプロジェクトとロードするプロジェクトのバージョンが一致しない場合、ロード実行前にオールクリア/初期化を実行する必要があります。	オールクリア/初期化実行後にロードを実行してください。CPUメモリからロードを実行する場合は、ロードするデータもクリアされるため、事前にバックアップを実施してください。	ロード
20	ユニット非対応 機能バージョン Unsupported Unit Function Version 不支持的单元功能版本	拡張ユニットが対応していない機能バージョンのセーブデータを読み込もうとした。	プロジェクトの拡張ユニット機能バージョンを確認してください。拡張ユニットの機能バージョンを確認してください。	ロード
21	※1 ※1 ※1	CR実行時に、CMで指定したドライブ番号が範囲外です。	ドライブ番号は0か1で指定してください。	ロード/セーブ
100	メモリカードカバーオープン Memory card cover open 存储卡外盖打开	メモリカードカバーが開いています。	メモリカードカバーを閉じてください。	ロード/セーブ
101	メモリカードがLOCKされています Memory card locked 存储卡被锁定	メモリカードがLOCKされています。	LOCKを解除するか、書込み可能なメモリカードを使用してください。	セーブ
102	ロードできません ---------- セーブできません Cannot load data ---------- Cannot save data 无法加载 ---------- 无法保存	メモリカードが破損しています。	メモリカードのファイルシステムをチェックしてください。	ロード ---------- セーブ
103	ファイル数上限を超過しました Max no. of files exceeded 超过了文件数上限	ユーザメモリ内のファイル数とフォルダ数の合計が上限を超過しています。	ユーザメモリ内のファイルまたはフォルダを削減してください。	ロード/セーブ
104	フォルダ階層が深すぎます Folder layer too deep 文件夹层数过多	ユーザメモリ内のフォルダ階層が深すぎます。	ユーザメモリ内のフォルダ階層を確認してください。	ロード/セーブ

番号	メッセージ	原因	対処	発生時
1000	ロードできません	• ファイルが破損しています。 • メモリカードが破損しています。 • ロード処理中にカードが抜かれました。 • 「ユーザメモリをクリアしてからロードする」の設定で、CPUメモリからロードしようとしました。 • 「オールクリアしてからロードする」の設定で、オートロード/ランロード以外の手段やCPUメモリからロードを実行しようとしました。	プロジェクトやメモリカードが適切かどうか確認してください。	ロード
	Cannot load data			
	无法加载			
1001	セーブできません	• メモリカードが破損しています。 • セーブ処理中にカードが抜かれました。	プロジェクトやメモリカードが適切かどうか確認してください。	セーブ
	Cannot save data			
	无法保存			

※1 CRを操作して実行した場合にのみ発生するエラーなので、アクセスウィンドウにメッセージは表示されません。

■ KV STUDIOを使用する

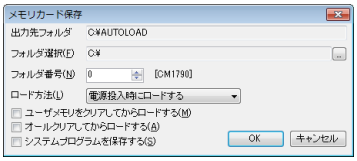
KV STUDIOのメニューから、ロード･セーブ用のプロジェクトデータの読み出し･保存ができます。

● ラダープログラムなど(デバイスファイル以外)

KV STUDIOのメニューから【ファイル(F)】→【メモリカード(M)】→【保存(W)】または【読出(R)】を選択します。

ポイント **ロードセーブ用のプロジェクトデータの作成手順はメモリカードへの保存の手順と同様です。作成したフォルダをCPUメモリに格納してください。**

保存を選択した場合

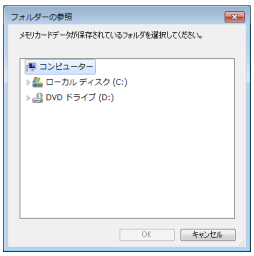


項目	説明
出力先フォルダ	ユーザメモリ用セーブデータの出力先フォルダ名称を表示します。フォルダ選択、フォルダ番号、ロード方法などによりフォルダ名称は異なります。
フォルダ選択	ユーザメモリ用セーブデータを出力するフォルダを選択します。右端の ... をクリックすると、[フォルダの参照]ダイアログボックスが表示されます。
フォルダ番号	フォルダ番号を指定します。 0入力時　：末尾に番号はつきません。 1～99入力時：フォルダの末尾に指定した番号を2桁で追加します。 右端のデバイス番号は、各ロード方法の場合にフォルダ番号を指定するデバイスを表示しています。 電源投入時にロードする　：　CM1790 PRG→RUN時にロードする：　CM1794
ロード方法	ロード方法を選択します。選択したロード方法により出力先のフォルダ名称は異なります。 電源投入時にロードする　：　「AUTOLOAD」,「AtLoad**」(**:01～99) PRG→RUN時にロードする：　「RUNLOAD」,「RnLoad**」(**:01～99)
ユーザメモリをクリアしてからロードする	チェックすると、ユーザメモリをクリアしてからロードします。
オールクリアしてからロードする	チェックすると、オールクリアしてからロードします。
システムプログラムを保存する	プロジェクトと一緒にユニットのシステムプログラムを保存します。ロード時にシステムプログラムの更新を実行します。「システムプログラムアップデートについて」(3-24ページ)

ポイント **同一の名称のフォルダを選択した場合、データは上書きされて、以前のデータは失われます。**

読出を選択した場合

[フォルダーの参照]ダイアログボックスが表示されます。



ポイント **CPUメモリからセーブデータを読み出したり、保存したセーブデータをPLCに転送したりする方法については「CPUメモリ(ユーザメモリ)のデータを読み書きする方法」(5-30ページ)を参照してください。**

オートローディング機能

電源投入時やPRG→RUN時にCPUメモリに保存されているプロジェクトデータを自動的に読み出します。
コントロールメモリ(CM)を使用して、複数のフォルダから読み出すフォルダを選択することができます。

■ 電源投入時に読み出す

1 《KV STUDIO》を使用して、「AutoLoad」、または「AtLoad＊＊」(＊＊：01～99)のフォルダ名称でプロジェクトデータを作成します。ストレージ転送ツールやFTPサーバ機能などを使用して、CPUメモリ(ユーザメモリ)にプロジェクトデータを書き込みます。

「ラダープログラムなど(デバイスファイル以外)」(5-44ページ)

2 CPUユニットの電源をOFFします。

3 CPUユニットの電源を投入します。

自動的にデータが読み出されます。

● 読み出すフォルダの指定方法

読み出すフォルダを指定する場合は、あらかじめCM1790に数値を格納します。

- CM1790の値が0の場合
  フォルダ名「AutoLoad」が読み出されます。
  「AutoLoad」フォルダがCPUメモリ(ユーザメモリ)に存在しない場合は、オートロードされません。
  電源投入前のプログラムで動作します。エラーは発生しません。

- CM1790の値が01～99の場合
  フォルダ名「AtLoad＊＊」(＊＊：CM1790の値(01～99))が読み出されます。

フォルダ番号の指定手順

1 電源投入している状態で、タッチパネルなどの外部機器からCM1790に数値を格納します。
2 電源を切ります。
3 電源を投入します。
4 CM1790で指定したフォルダが読み出されます。
5 読み出し完了後、CM1790に0が格納され、CM1792に読み出したフォルダ番号が格納されます。

参考　オートロード完了後、CM1790に0が格納されるので、次回電源投入時、同じフォルダ番号は読み出されません。ただし、CM1790に0が格納されている状態でメモリカードに「AutoLoad」フォルダがあると、次回電源投入時に読み出されるので注意してください。

ポイント
- **《KV STUDIO》のCPUシステム設定でライトプロテクト/プロジェクトロード禁止されている場合はエラー(オートローディング失敗)が発生し、PRGモードになります。エラークリア後、RUNモードで動作するとオートロード実行前のプログラムで動作します。**
  以下の場合はエラー(オートロードフォルダ)が発生し、PRGモードになります。
  - **CM1790の値が100以上の場合**
  - **CM1790で指定したフォルダがCPUメモリ(ユーザメモリ)にない場合**
- **PLC本体に格納されているプロジェクトの機能バージョンと転送するプロジェクトの機能バージョンが異なる場合、CPUのオールクリア/初期化が必要になります。**

■ PRG→RUN時に読み出す

1 **《KV STUDIO》を使用して、「RunLoad」、または「RnLoad＊＊」(＊＊：01～99)のフォルダ名称でCPUメモリ(ユーザメモリ)にプロジェクトデータを書き込みます。**

「ラダープログラムなど(デバイスファイル以外)」(5-44ページ)

2 **CPUユニットをPRG→RUNに切り換えます。**

自動的にデータが読み出され、運転を開始します。

● 読み出すフォルダの指定方法

読み出すフォルダを指定する場合は、あらかじめCM1794に数値を格納します。

- CM1794の値が0の場合
  フォルダ名「RunLoad」が読み出されます。
  「RunLoad」フォルダがCPUメモリに存在しない場合は、オートロードされません。PRG→RUN前のプログラムで動作します。エラーは発生しません。

- CM1794の値が01～99の場合
  フォルダ名「RnLoad＊＊」(＊＊はCM1794の値01～99)が読み出されます。

フォルダ番号の指定手順

1 タッチパネルなどの外部機器からCM1794に数値を格納します。
2 PRG→RUNに切り換えます。
3 CM1794で指定したフォルダが読み出されます。
4 読み出し完了後、CM1794に0が格納され、CM1796に読み出したフォルダ番号が格納されます。

参 考

オートロード完了後、CM1794に0が格納されるので、次回電源投入時、同じフォルダ番号は読み出されません。ただし、CM1790に0が格納されている状態でCPUメモリ(ユーザメモリ)に「RunLoad」フォルダがあると次回PRG→RUNすると読み出されるので注意してください。

ポイント

- **《KV STUDIO》のCPUシステム設定でライトプロテクト/プロジェクトロード禁止されている場合はエラー(オートローディング失敗)になりRUNモードにできません。エラークリア後、RUNモードで動作するとオートロード実行前のプログラムで動作します。**
  以下の場合はエラー(オートロードフォルダ)になりRUNモードにできません。エラークリア後、RUNモードで動作するとオートロード実行前のプログラムで動作します。
  - **CM1794の値が100以上の場合**
  - **CM1794で指定したフォルダがCPUメモリ(ユーザメモリ)にない場合**
- **PLC本体に格納されているプロジェクトの機能バージョンと転送するプロジェクトの機能バージョンが異なる場合、CPUのオールクリア/初期化が必要になります。**

RAMモード

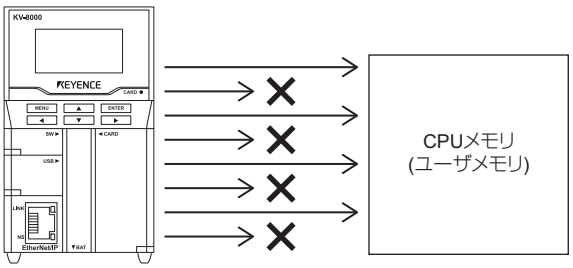
RAMモードとはCPUメモリにファイル保存する際に、一度RAMにファイルを作成し、完成したファイルをまとめて1回で保存することで、CPUメモリに対して、高速でファイル保存する機能です。

ポイント

- **RAMモードで使用するRAMは電源をOFFするとクリアされます。**
  電源をOFFする前に、CPUメモリ保存処理を実行してください。

RAMモード未使用時

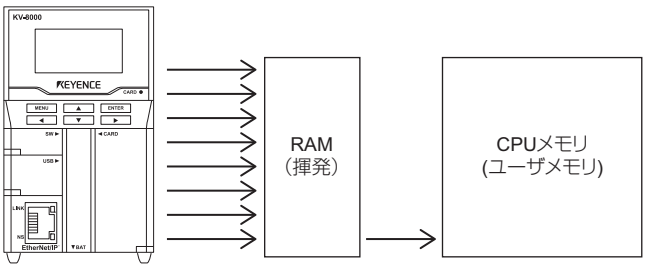
RAMモード未使用時



高速で書き込みを実行すると、
CPUメモリの書き込み回数制限に短期間
で到達するため、書き込み速度を制限

RAMモード使用時

RAMモード使用時



RAM上でファイル作成するため、
CPUメモリへの書き込み回数を
抑えることができる

ポイント

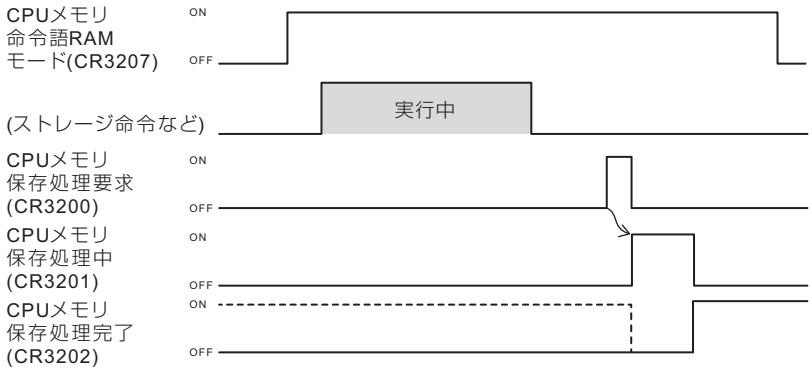
- **ファイルレジスタ命令またはストレージ命令でCPUメモリにアクセスする場合は、命令語RAMモード動作リレー(CR3207)をONしてRAMモードに切り換えます。**
- **ロギングを使用してCPUメモリにアクセスする場合はロギング/トレース設定のオプションで、「不揮発メモリへのファイル保存を行う」のチェックを外すとRAMモードで動作します。**
- **CPUメモリへの書き込み超過によるメモリの劣化を防止するため、CPUメモリへの書き込み速度を制限しています。**

■ 設定方法

ラダープログラム中でCPUメモリ命令語RAMモードリレー(CR3207)をONします。
CPUメモリに書き込む際には、CPUメモリ保存処理要求リレー(CR3200)をONします。

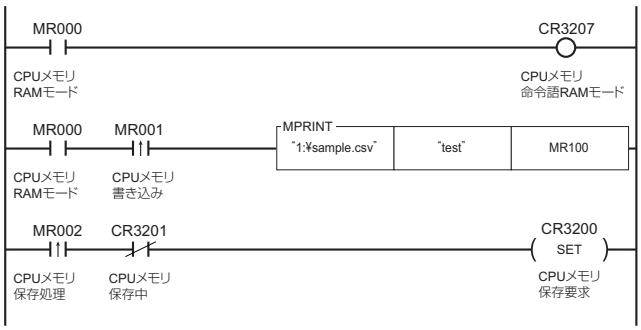
タイムチャート

タイムチャート



①CPUメモリ命令語RAMモードリレー(CR3207)をONし、ストレージ命令などを実行します。(ラダー)
②ストレージ命令の実行が完了し、保存が必要になったタイミングで、CPUメモリ保存処理要求リレー(CR3200)をONします。(ラダー)
③CPUメモリの保存を開始するとCPUメモリ保存処理中リレー(CR3201)がONします。CPUメモリ保存処理完了リレー(CR3202)がONしていた場合はOFFします。(CPU)
④CPUメモリ保存処理が完了するとCPUメモリ保存処理中リレーがOFFし、CPUメモリ保存処理完了がONします。(CPU)

ラダープログラム例



〈ニモニックリスト〉

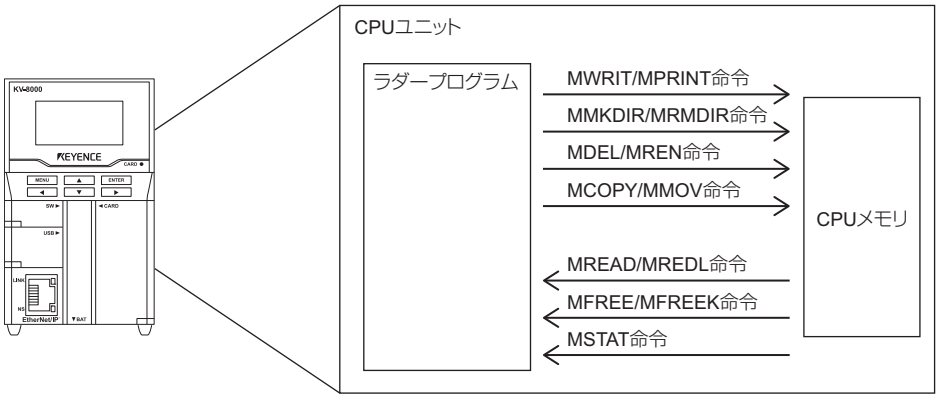
```
LD MR000
OUT CR3207
LD MR000
ANP MR001
MPRINT "1:¥sample.csv" "test" MR100
LDP MR002
ANB CR3201
SET CR3200
```

参 考

- CPUメモリ命令語RAMモードリレーがON中であってもCPUメモリ保存処理要求は実行できます。
- CPUメモリ保存要求はSET命令で実行してください。

ストレージ命令

ストレージ命令を使用すると、ラダープログラムでデバイスの値をCPUメモリに書き込んだり、CPUメモリに格納した値を読み出したりできます。

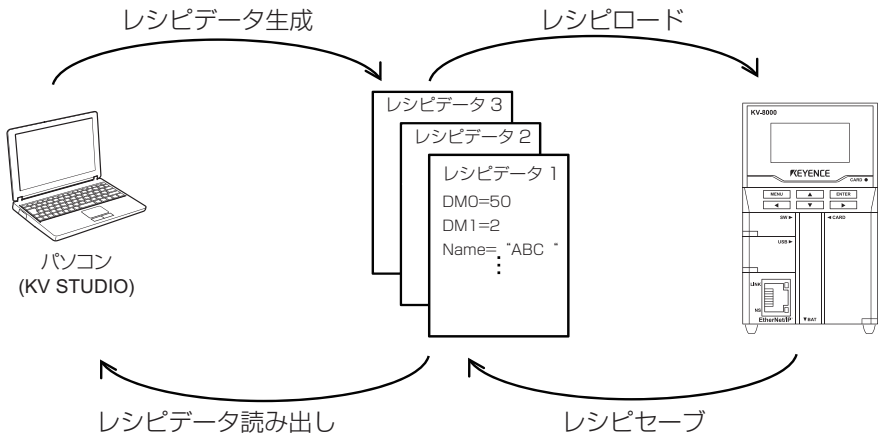


ストレージ命令については、『KV-8000/7000/5000/3000/1000･KV Nanoシリーズ命令語リファレンスマニュアル』を参照してください。

5-4 レシピロード・セーブ

レシピロード機能/レシピセーブ機能の概要

KV-8000(A)ではデバイス･変数の値をデータテーブルとして複数保存しておき、データテーブルを切り換えることでCPUユニット内のデバイス･変数や、KV-XD02のユニット変数の値を一斉に切り換えられます。このデータテーブルのことをレシピデータといいます。
レシピロード機能を使用するとあらかじめ保存されたレシピデータを読み出して、デバイス･変数の値に反映できます。
レシピセーブ機能を使用すると、現在のデバイス･変数の値を使用して新たなレシピデータを作成できます。
レシピデータはKV STUDIO上で生成し、CPUユニットのストレージ(メモリカード/CPUメモリ)に保存します。また、CPUユニットのストレージ内のレシピデータをKV STUDIOで読み出すことも可能です。



レシピロード･セーブ機能の仕様は以下のとおりです。

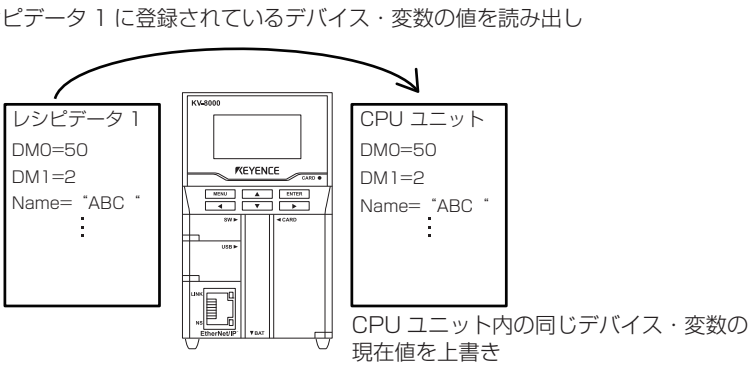
項目	内容
レシピデータの対象	デバイス/変数
レシピデータの最大個数	1000個(0～999)
レシピデータの格納先	メモリカード/CPUメモリ
RUNモードでのロード･セーブ	ロード･セーブともにRUNモード時でも可能
レシピロード･セーブの実行方法	ラダープログラムからCR/CMを操作して実行

ポイント
- **レシピロード･セーブ機能はCPU機能バージョン2.0以降のKV-8000でのみ使用できます。**
- **TIMER型、COUNTER型の変数はレシピロード･セーブ機能では読み出し･保存できません。**

● レシピロード機能

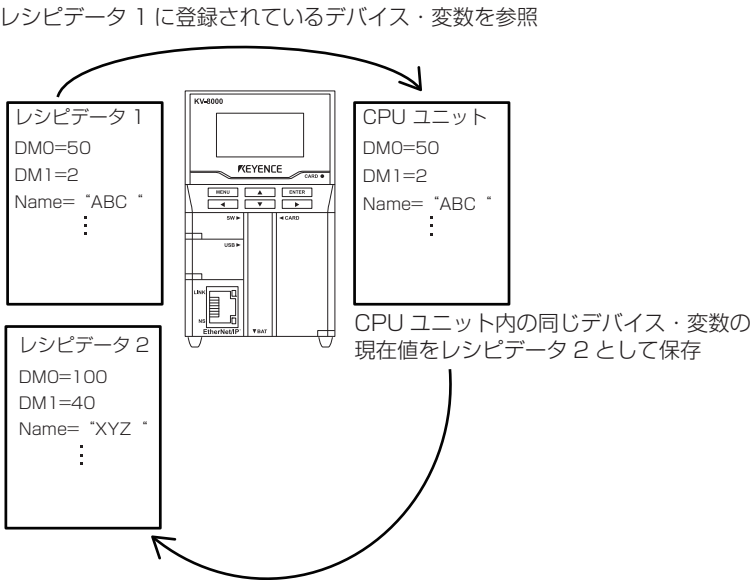
レシピロード機能とは、あらかじめ作成してメモリカード/CPUメモリに保存しておいたレシピデータの内容を読み出して、CPUユニットのデバイス･変数の値をレシピ中の値で上書きする機能です。読み出すレシピは、レシピのフォルダ番号を指定して選択します。

レシピデータの内容はKV STUDIOで設定してCPUユニットに転送します。



● レシピセーブ機能

レシピセーブ機能は、参照フォルダ番号内に保存されているレシピデータに登録されているデバイス・変数を対象にして、同じデバイス・変数の値を出力フォルダ番号で指定したレシピデータとして出力する機能です。
出力したレシピデータの内容はKV STUDIOで読み出せます。



参 考

設定した全ての変数のバックアップを保存する場合は、プロジェクトセーブ機能を使用します。
「プロジェクトデータの保存・読み出し」(5-35ページ)

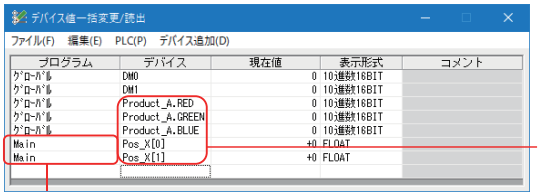
レシピデータ生成・読み出しの手順

ここではKV STUDIO上でレシピデータを作成してCPUユニットのストレージに転送する手順と、CPUユニットのストレージに存在するレシピデータをKV STUDIO上に読み出す手順について説明します。

■ レシピデータの生成(KV STUDIO→CPUユニット)

1 **KV STUDIOメニューの【モニタ/シミュレータ(N)】→【デバイス値一括変更/読出(D)】を選択します。**

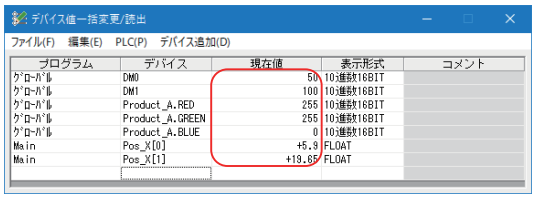
2 **デバイス値一括変更/読出ウィンドウ上でレシピデータに含めるデバイス・変数を登録します。**



構造体・配列型変数は
メンバや要素まで指定してください。

ローカル変数を入力する場合は
あらかじめプログラムを指定する必要があります。

3 **レシピデータとして登録する値を入力します。**

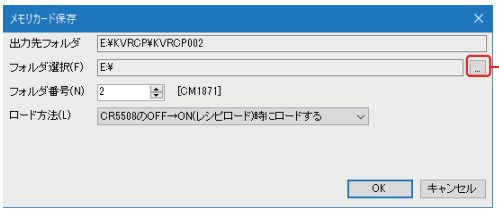


参 考

デバイス値一括変更/読出ウィンドウメニューの【PLC(P)】→【PLCから全項目読出(R)】を選択すると、接続しているCPUユニット内のデバイス・変数の値を一括で読み出して現在値に反映できます。

4 **デバイス値一括変更/読出ウィンドウメニューの【ファイル(F)】→【メモリカード保存(Z)】を選択します。**

ロード方法は「CR5508のOFF→ON(レシピロード)時にロードする」を選択します。レシピデータを作成するフォルダと、レシピデータの番号(フォルダ番号:0～999)を指定し、〔OK〕をクリックします。



フォルダを選択する
ダイアログが表示
されます。

ポイント

- **メモリカードに直接保存する場合は、メモリカード直下を指定してください。パソコンのフォルダなどメモリカード以外に保存する場合は、「KVRCP」フォルダごとメモリカード直下にコピーしてください。**
- **フォルダを選択する前に、ロード方法を選択してください。**
- **パソコンにデータ書き込み権限が付与されていない場合、メモリカードなどの外部メディアへの保存に失敗します。詳細は、パソコンの管理者にお問い合わせください。**

5 **レシピデータをストレージに格納します。**

メモリカードにレシピデータを保存した場合は、メモリカードをCPUユニットに挿入します。CPUメモリにレシピデータを保存する場合は、ストレージ転送ツールやFTPサーバ機能、FTPクライアント機能などを使用して「KVRCP」フォルダをCPUメモリ直下に転送します。

■ レシピデータの読み出し（CPUユニット→KV STUDIO）

1 **CPUユニットのストレージ内のレシピデータをパソコン上に転送します。**

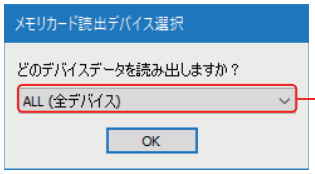
メモリカードにレシピデータが保存されている場合は、メモリカードを抜き取りパソコンに挿入します。CPUメモリにレシピデータが保存されている場合は、ストレージ転送ツールやFTPサーバ機能、FTPクライアント機能などを使用して「KVRCP」フォルダ内の「KVRCP***」フォルダ（***は読み出したいレシピデータの番号）をパソコン上に転送します。

2 **KV STUDIOメニューの【モニタ/シミュレータ(N)】→【デバイス値一括変更/読出(D)】を選択します。**

3 **デバイス値一括変更/読出ウィンドウメニューの【ファイル(F)】→【読出(R)】を選択します。**

フォルダを参照するダイアログが表示されるので、手順1でパソコンに転送した「KVRCP***」フォルダを選択します。

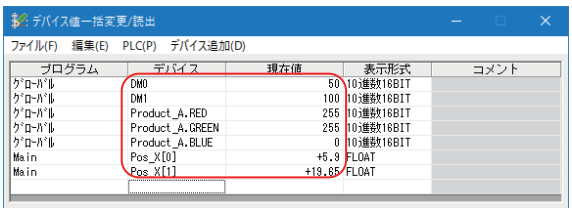
4 **レシピデータからKV STUDIOに読み出すデバイス・変数を選択します。**



「ALL（全デバイス）」を選択すると、レシピデータ内のすべてのデバイス・変数の値を読み出せます。

5 **読み出したレシピデータの内容がデバイス値一括変更/読出ウィンドウに反映されます。**

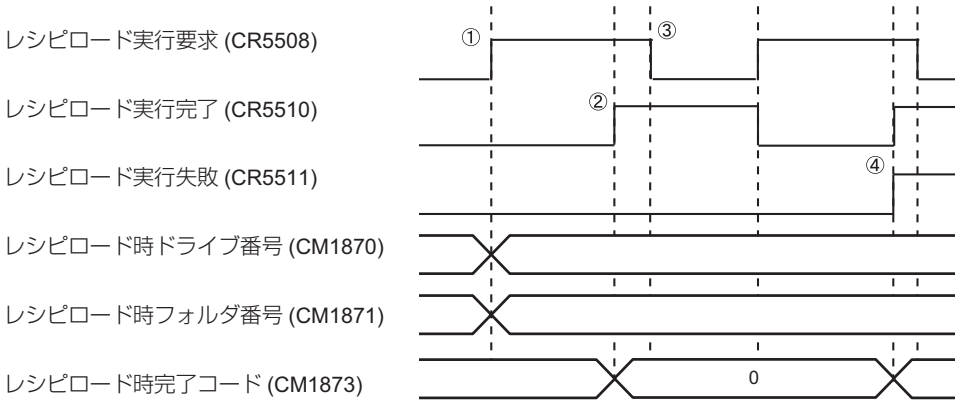
デバイス値一括変更/読出ウィンドウメニューの【ファイル(F)】→【CSV/TXTファイルに名前を付けて保存(A)】を選択すると、レシピデータの内容をCSV/TXT形式のファイルとして保存できます。



レシピロード・セーブの手順

ここでは、CPUユニットのストレージに保存されているレシピデータの内容をロードしてCPUユニットのデバイス・変数に反映させる手順と、CPUユニットのデバイス・変数の値を使用してレシピデータを上書き・新規作成する手順を説明します。

■ レシピロードの手順



① レシピロード時ドライブ番号（0：メモリカード/1：CPUメモリ）とレシピロード時フォルダ番号にロードしたいレシピデータが保存されているドライブ番号とフォルダ番号を指定し、レシピロード実行要求をONします。

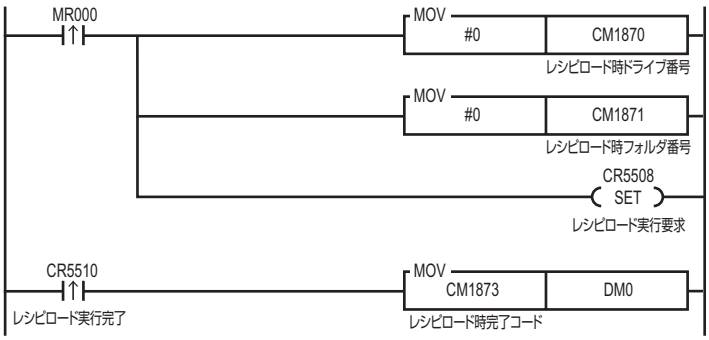
② レシピロードが正常に完了すると、レシピロード実行完了がONします。また、レシピロード時完了コードに「0」が格納されます。

③ レシピロード実行完了がONしたことを確認し、ラダープログラムからレシピロード実行要求をOFFします。レシピロード実行完了は次回のレシピロード実行要求のOFF→ON時に自動的にOFFします。

④ レシピロードに失敗すると、レシピロード実行完了とレシピデータ実行失敗がともにONします。レシピロード時完了コードには失敗した要因に応じた完了コードが格納されます。

参考プログラム

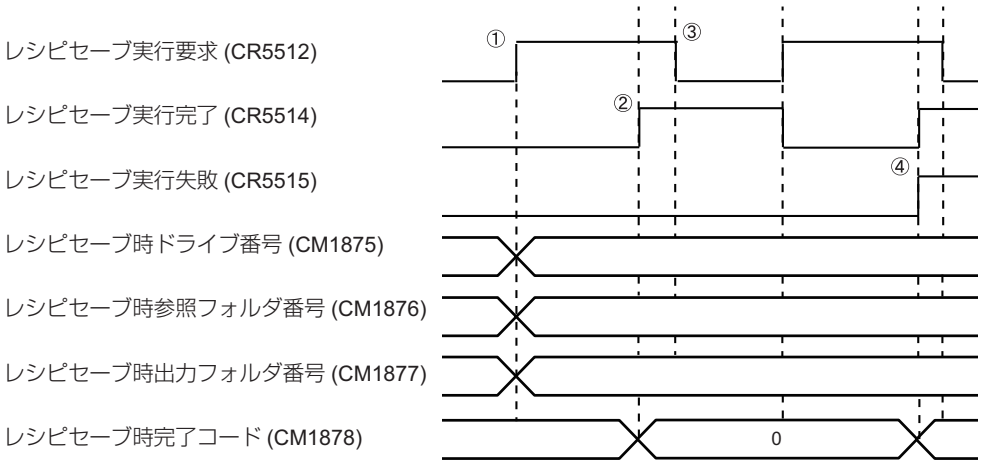
メモリカード内のフォルダ番号「0」のレシピデータをロードする参考プログラムです。



〈ニモニックリスト〉

```
LDP MR000
MOV #0 CM1870
MOV #0 CM1871
SET CR5508
LDP CR5510
MOV CM1873 DM0
```

■ レシピセーブの手順



① レシピセーブ時ドライブ番号(0:メモリカード/1:CPUメモリ)とレシピセーブ時出力フォルダ番号でレシピデータの出力先を指定し、レシピセーブ実行要求をONします。このときにレシピデータとして出力する対象のデバイス・変数は、レシピセーブ時参照フォルダ番号で指定した番号のレシピデータに記載されているデバイス・変数になります。

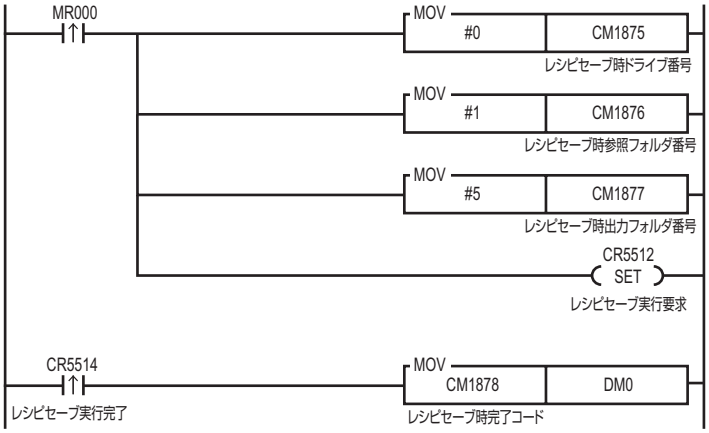
② レシピセーブが正常に完了すると、レシピセーブ実行完了がONします。また、レシピセーブ時完了コードに「0」が格納されます。

③ レシピセーブ実行完了がONしたことを確認し、ラダープログラムからレシピセーブ実行要求をOFFします。レシピセーブ実行完了は次回のレシピセーブ実行要求のOFF→ON時に自動的にOFFします。

④ レシピセーブに失敗すると、レシピセーブ実行完了とレシピデータ実行失敗がともにONします。レシピセーブ時完了コードには失敗した要因に応じた完了コードが格納されます。

参考プログラム

メモリカード内のフォルダ番号「1」のレシピデータに登録されているデバイス・変数を参照し、PLC内のデバイス・変数の値を使用してフォルダ番号「5」のレシピデータとしてセーブする参考プログラムです。



〈ニモニックリスト〉

```
LDP MR000
MOV #0 CM1875
MOV #1 CM1876
MOV #5 CM1877
SET CR5512
LDP CR5514
MOV CM1878 DM0
```

レシピロード完了コード・レシピセーブ完了コード

レシピロード完了コード（CM1873）、レシピセーブ完了コード（CM1878）に格納される完了コードの一覧です。

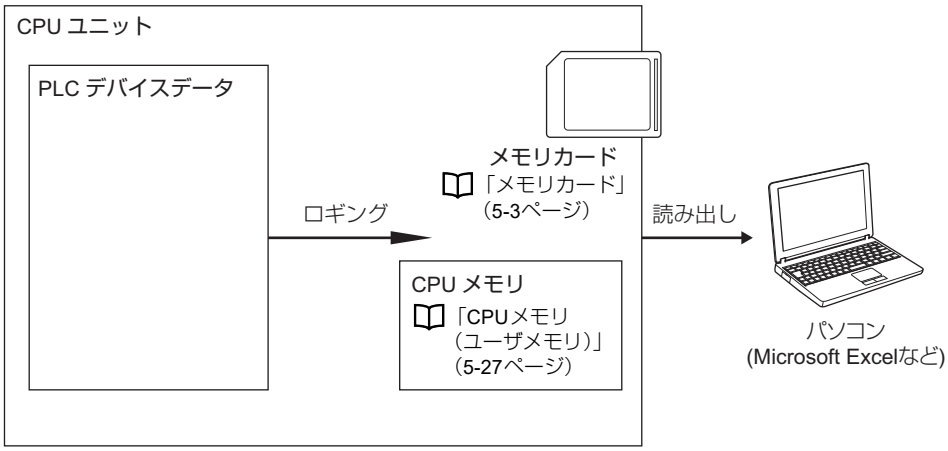
番号	内容	原因	対処	発生時
2	空き番号がありません	既にファイルがあったため書き込めませんでした。	保存したいフォルダ番号に空き番号を設定してください。	セーブ
3	指定フォルダ番号範囲外	CMで指定したフォルダ番号が範囲外でした。	0～999の番号を指定してください。	ロード/セーブ
4	データなし	メモリカード/CPUメモリに指定したCMのフォルダがありません。	レシピデータが入っていることを確認してください。	ロード/セーブ
5	メモリカードがありません ---------- CPUメモリアンマウント	メモリカード/CPUメモリを認識できない状態でロードまたはセーブを実行しました。	メモリカードからロード、またはメモリカードへセーブする場合は、メモリカードを挿入してください。 CPUメモリをマウントしてください。	ロード/セーブ
6	アクセス競合	他の経路からプログラムの転送または読出しをしていたため、セーブ･ロードができませんでした。	プログラムの転送もしくは読み出しが終わってからセーブ･ロードを実行してください。	ロード/セーブ
12	空き容量がありません	メモリカード/CPUメモリの空き容量がありません。	空き容量を増やしてください。	セーブ
21	指定ドライブ番号範囲外	CR実行時に、CMで指定したドライブ番号が範囲外です。	ドライブ番号は0か1で指定してください。	ロード/セーブ
22	未登録の変数が含まれます	レシピデータに含まれる変数が変数編集ウィンドウで定義されていません。	レシピデータに含まれる変数(構造体型、及びファンクションブロック型など含む)は変数編集ウィンドウで定義して下さい。	ロード/セーブ
23	変数データ型不一致	レシピデータに含まれる変数のデータ型がPLCに登録されている変数のデータ型と一致していません。	レシピデータに含まれる変数のデータ型は変数編集ウィンドウで定義しているデータ型と致させてください。	ロード/セーブ
24	非対応の変数データ型	レシピデータに非対応のデータ型の変数が含まれています。	レシピデータに含まれる変数のデータ型を確認してください。	ロード/セーブ
100	メモリカードカバーオープン	メモリカードカバーが開いています。	メモリカードカバーを閉じてください。	ロード/セーブ
101	メモリカードがLOCKされています	メモリカードがLOCKされています。	LOCKを解除するか、書込み可能なメモリカードを使用してください。	セーブ
102	メモリカード異常	メモリカードが破損しています。	メモリカードのファイルシステムをチェックしてください。	ロード ---------- セーブ
103	ファイル数上限を超過しました	ユーザメモリ内のファイル数とフォルダ数の合計が上限を超過しています。	ユーザメモリ内のファイルまたはフォルダを削減してください。	セーブ
1000	ロードできません	• ファイルが破損しています。 • メモリカードが破損しています。 • ロード処理中にカードが抜かれました。 • 「ユーザメモリをクリアしてからロードする」の設定で、CPUメモリからロードしようとしました。	プロジェクトやメモリカードが適切かどうか確認してください。	ロード
1001	セーブできません	• メモリカードが破損しています。 • セーブ処理中にカードが抜かれました。	プロジェクトやメモリカードが適切かどうか確認してください。	セーブ

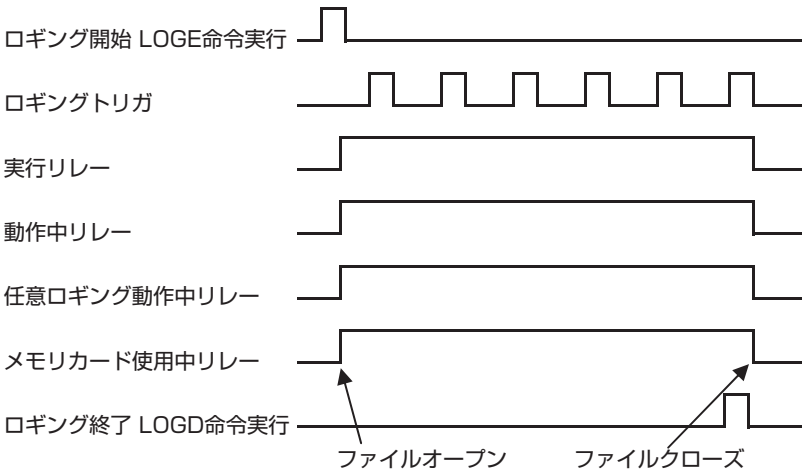
ロギング機能/トレース機能の概要

● ロギング機能

ロギング機能とは、指定したデバイスの状態や値を任意のタイミング(トリガ)で取り込む機能です。取り込んだデータは、CSVファイル形式でメモリカードまたはCPUメモリ(ユーザメモリ)に自動保存されます。取り込むデバイスやトリガなどは《KV STUDIO》で簡単に設定できます。
ロギング・トレースデータはCSV形式で保存されるため、Microsoft Excelなどの表計算ソフトでそのまま使用できます。CSV形式で保存されたデータは以下の方法で読み出せます。

- メモリカードを抜き出してパソコンで読み出す
- ストレージ転送ツールや、FTP機能を使用してPLC経由で読み出す
- リアルタイムチャートモニタを使用して読み出す





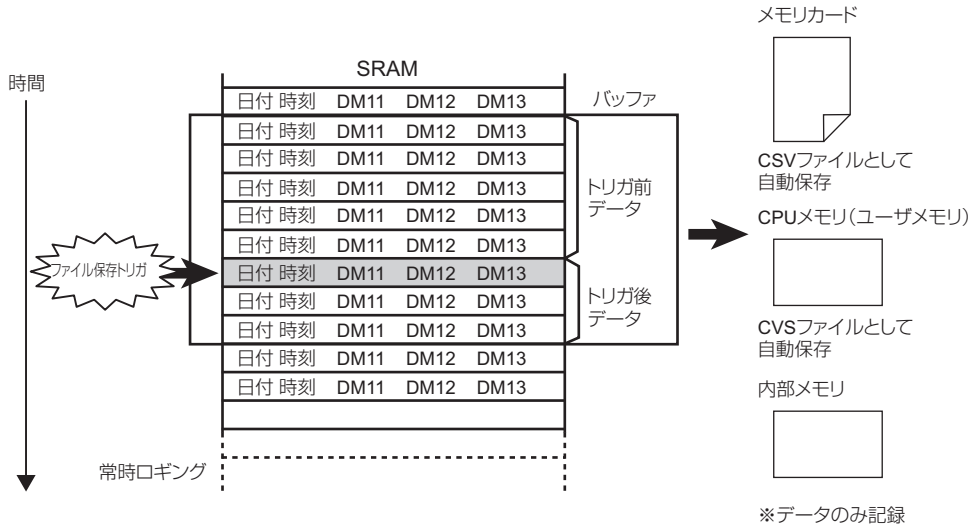
※オプション設定により各リレーの挙動は変わります。

参 考

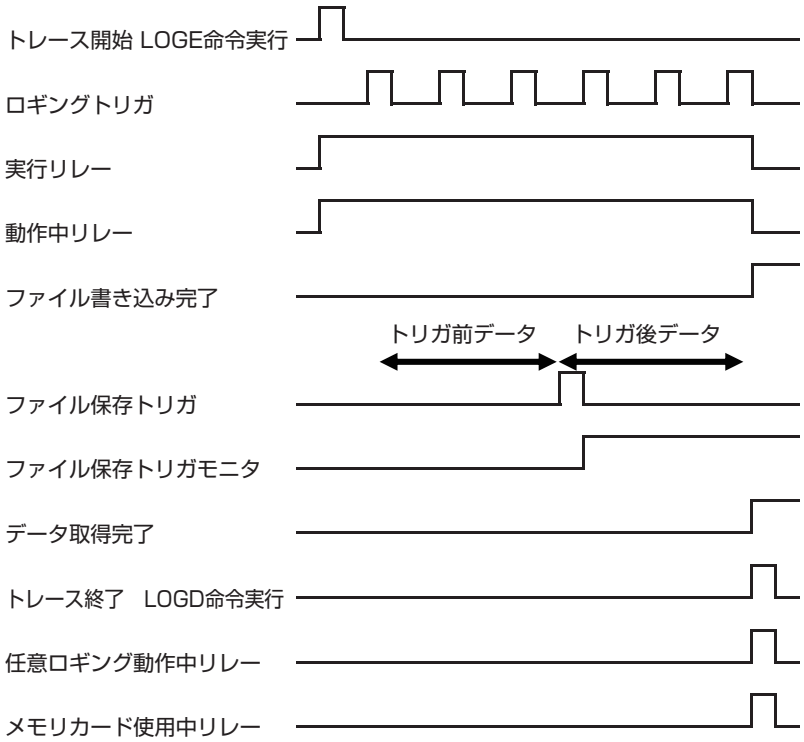
- ファイルが存在しない状態でロギング・トレースを開始すると自動でファイルを新規作成します。
- ロギング・トレース機能とストレージ命令は同時使用可能です。優先順位はストレージ命令 > ロギング・トレース機能となります。

● トレース機能

ロギング機能と同様にデバイスの状態や値をバッファメモリ上に記憶します。ファイル保存のタイミングで、その前後の指定したデータ数のみを抽出してメモリカードやCPUメモリ（ユーザメモリ）、または内部メモリに記録します。
例えば、装置の異常時にファイル保存するようにトリガ設定すると、異常発生前後のみのデータが自動保存されるので、異常解析に役立ちます。



参 考　ファイル保存先を「内部メモリ」にして保存したデータは、《KV STUDIO》の「リアルタイムチャートモニタ」上に読み出すことができます。



ポイント　**トレース機能で取得したデータをメモリカードに保存する場合はデータ取得完了をトリガにしてLOGD命令を実行してください。**
データ取得完了前にLOGD命令を実行してもファイルは生成されません。

使用手順

ロギング・トレース機能の使用手順は次のとおりです。

① デバイスやトリガを設定

【ツール(T)】－【ロギング/トレース設定(L)】

「設定」(5-60ページ)

▼

② ラダープログラム作成

「LOGE」「LOGD」「TRGD」命令

「ロギング命令」(5-73ページ)

▼

③《KV-8000》への転送

【モニタ/シミュレータ(N)】－【PLC転送→モニタ(C)】

▼

④ ロギング・トレースの実行

LOGE命令を実行するとロギング実行中になります。
ロギング実行中に、ロギングトリガ(トレース使用時はファイル保存トリガも必要)を発生するとデータをファイルに書き込んでいきます。
LOGD命令を実行してファイルをクローズします。

設定

ロギング・トレース機能を使用するための設定は《KV STUDIO》でおこないます。
ロギング・トレース機能を使用するには、以下の設定が必要です。

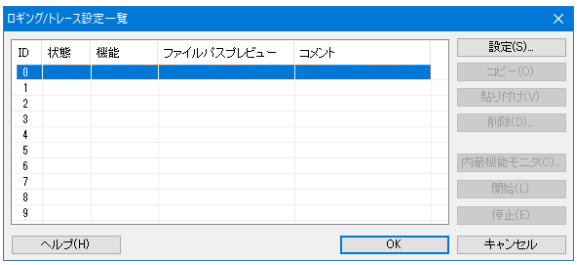
- データ保存先
- 保存するファイル名の設定
- 機能(ロギング/トレース)の選択
- ロギング・トレースするデバイスの設定
- トリガの設定
- オプション設定
- トレース詳細設定(トレース機能選択時のみ)

ロギング・トレース機能は最大10個まで作成でき、ロギング/トレースID(0～9)で区別します。
ロギング/トレースIDごとにファイル名、デバイス、トリガ設定します。

1 メニューから【ツール(T)】▶【ロギング/トレース設定】の順に選択します。

[ロギング/トレース設定一覧]ダイアログボックスが表示されます。

別手順 ・ボタンをクリック



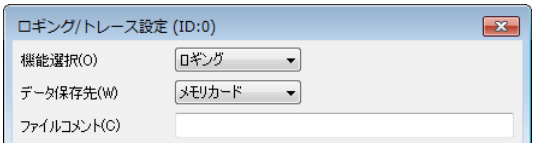
項　目	説　明
状態	モニタ中のみ現在のロギング・トレース機能の実行状態(停止中/動作中/エラー)を表示します。
機能	設定している機能(ロギング/トレース)を表示します。
ファイルパスプレビュー	ファイルが保存される場所(ファイルパス)の一例を表示します。
コメント	設定したファイルコメントを表示します。
設定	〔ロギング/トレース設定〕ダイアログボックスが表示されます。選択しているロギング/トレースIDの設定が行えます。
コピー	選択しているロギング/トレースIDの設定をコピーします。
貼り付け	選択しているロギング/トレースIDに、コピーした設定を貼り付けます。
削除	選択しているロギング/トレースIDの設定を削除します。
OK	設定した内容を確定してダイアログボックスを閉じます。
キャンセル	設定した内容をキャンセルしてダイアログボックスを閉じます。
内蔵機能モニタ	モニタ中のみ選択しているロギング/トレースIDの内蔵機能モニタが表示されます。
開始	モニタ中のみ選択しているロギング/トレースIDの実行を開始します。(ラダープログラム優先)
停止	モニタ中のみ選択しているロギング/トレースIDの実行を停止します。(ラダープログラム優先)

2 設定するロギング/トレースIDを選択して、〔設定〕ボタンを押します。

〔ロギング/トレース設定〕ダイアログボックスが表示されます。

項　目	説　明
機能選択	機能（ロギング/トレース）を選択します。
データ保存先	「メモリカード/CPUメモリ/内部メモリ」から選択します。 （「機能選択」で（トレース）を選択した場合のみ、「内部メモリ」が選択できます※1）
ファイルコメント	ロギング/トレース設定の【オプション】>【CSVファイルの先頭にファイルコメントを挿入】にチェックをした場合の、ファイルコメントを設定します。（半角64文字以内）CSVファイルの1行目に挿入できます。

※1　内部メモリを選択した場合は、〔リアルタイムチャートモニタ〕でデータを読み出します。
『KV STUDIOユーザーズマニュアル「リアルタイムチャートモニタ」』
内部メモリに保存した収集データは、電源を切ると消去されます。

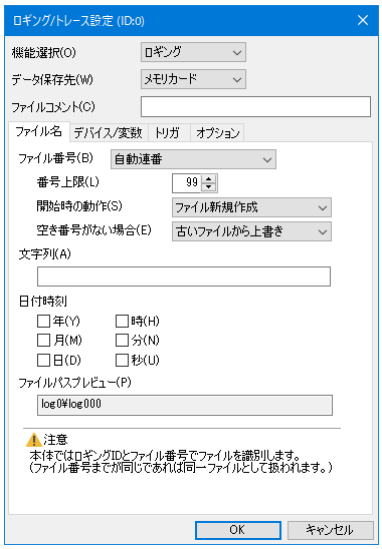


「ロギング機能/トレース機能の概要」（5-57ページ）

ファイル名

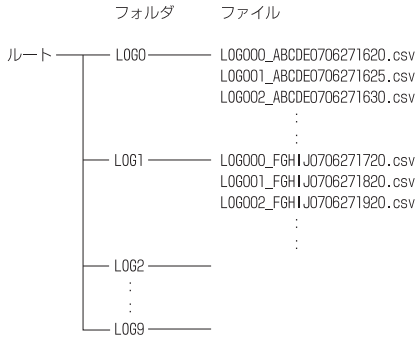
保存するファイルに関する設定をおこないます。
ロギング・トレースIDごとに保存するフォルダ名が決められています。(LOG0～LOG9)

参 考 データ保存先が「内部メモリ」の場合、「ファイル名」の各項目は設定できません。



ロギングデータのファイルは、メモリカードまたはCPUメモリ(ユーザメモリ)内に以下のように保存されます。

例



項　目			説　明
ファイル番号	ファイル番号		**「自動連番/固定/デバイス値(下三桁を使用)」**から選択します。
	自動連番※1		0～999の間で未使用の番号から最小のファイル番号を使用します。
		番号上限	ファイル番号の上限値(0～999)を設定します。
		開始時の動作	ロギング開始時の動作を**「ファイル新規作成/最新のファイルに追記」**から選択します。
		空き番号がない場合	空き番号がない場合の処理を**「古いファイルから上書き/ロギング停止」**から選択します。
	固定		0～999の固定値で指定します。
		番号	ファイル番号(0～999)を設定します。
		同一ファイル存在時	同一ファイルが存在する時の処理を**「古いファイルを上書き/追記」**から選択します。
	デバイス値(下三桁を使用)		実行許可した時点でのワードデバイスの値(0～999)で指定します。
		デバイス	ファイル番号を格納するデバイス、もしくはUINT/UDINT/TIMER/COUNTER型の変数を設定します。
		同一ファイル存在時	同一ファイルが存在する時の処理を**「古いファイルを上書き/追記」**から選択します。
文字列※2			ファイル名に付加する文字列を設定します。
日付時刻※2			チェックすると、ファイル名にLOGE命令実行時の作成日時情報が含まれます。日時情報はファイル名の新規作成時点での日時情報に固定されますので、追記時にはファイル名は変更されません。既存ファイルを上書きしたときには古いファイルが削除されるので、新しい日時情報のファイル名でファイルが作成されます。 年(00～99)　月(01～12)　日(01～31)　時(00～23) 分(00～59)　秒(00～59)
ファイルパスプレビュー			ファイルが保存される場所(ファイルパス)の一例が表示されます。

※1 空き番号検索が完了して、ファイルへの書込み準備ができると、ロギング動作中リレーがONします。それまでにトリガを発生させると、バッファオーバーフローが発生しやすくなりますので、ロギング動作中リレーがONしてからトリガを発生させるようにしてください。
空き番号検索にかかる時間は、番号上限の値が大きくなると長くなります。番号上限が999のときに必要な時間の目安は1秒です。

※2 ロギングID4、ファイル番号2、文字列「KEYENCE」、日時情報「年月日時分秒」を選択して、2019年4月17日15時16分17秒に新規作成した場合、メモリカードやCPUメモリのルートのフォルダ「log4」に「log002_KEYENCE190417.csv」ファイルができます。

ファイル名： log（ファイル番号）_（文字列）（日付時刻）

最大半角32文字

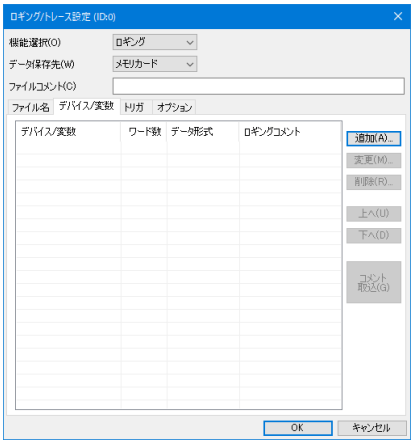
ファイル番号に空きが無い時などの動作の詳細については「ロギング・トレース開始時の処理について」（5-83ページ）を参照してください。

ポイント

ロギング・トレース機能で作成したファイルはフォルダ（ロギング・トレースID）とファイル番号で認識されます。
パソコンで、ストレージ内の同一フォルダ（ロギングID）内に同一ファイル番号で異なる文字列もしくは、日付時刻のファイルを複数作成しないでください。

デバイス/変数

保存するデバイス/変数の設定をおこないます。ビットデータとワードデータを混在して設定できます。
1つのロギングIDあたり、最大512デバイスまで設定できます。



例

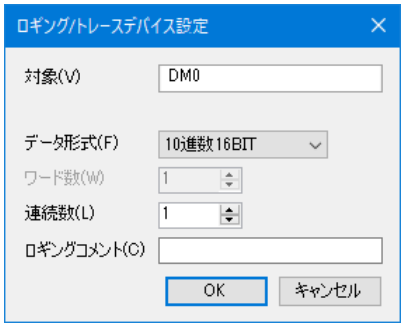
①512デバイスをすべてビットデータに設定した場合：
512個のビットデータのON/OFF状態（0または1）を記録します。

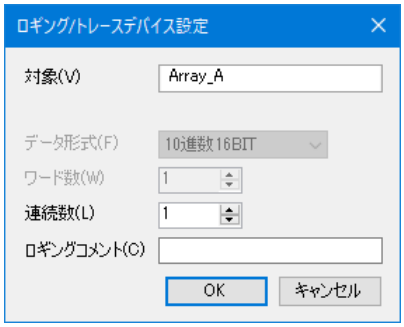
②512デバイスをすべて1ワードのワードデータに設定した場合：
512個の1ワードのワードデータの値を記録します。

③512デバイスをすべて2ワードのワードデータに設定した場合※1：
256個の2ワードのワードデータの値を記録します。

※1 TC/CC/TRM/Zなど1ワードで32ビットを占有するデバイスも含みます。

項　目	説　明
追加	〔追加〕ボタンを押すと、「ロギング/トレースデバイス設定」ダイアログボックスが表示されます。ロギング・トレースするデバイスを追加します。
変更	変更するデバイスを選択して〔変更〕ボタンを押すと、〔ロギング/トレースデバイス設定〕ダイアログボックスが表示されます。ロギング・トレースするデバイス設定を変更します。
削除	〔削除〕ボタンを押すと、選択しているデバイスを削除します。（複数選択可）
上へ	選択しているデバイス登録順を上方向に移動します。
下へ	選択しているデバイス登録順を下方向に移動します。
デバイスコメント取込	「デバイスコメント取込」ボタンを押すと、設定されたデバイスのデバイスコメントをロギングコメント欄に取り込みます。

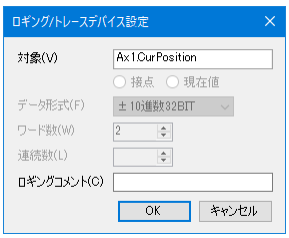




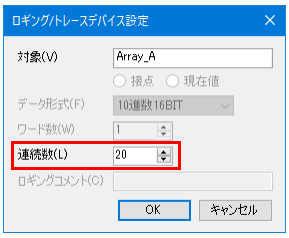
項　目	説　明
対象	ロギング・トレースする対象のデバイス、変数を設定します。
データ形式	デバイスの場合 「10進数16BIT/10進数32BIT/±10進数16BIT/±10進数32BIT/16進数16BIT/16進数32BIT/ASCII16BIT/FLOAT」から選択します。 SJISコードを取得したい場合は、ASCII16BITに設定してください。 「FLOAT」を選択した場合は指数形式で保存されます。(例:0.123E-03) 小数点付きのデータを取得したい場合、ラダープログラムで文字列に変換し、「ASCII16BIT」を選択して保存してください。 変数の場合 グローバル変数のみ指定可能で、変数編集ウィンドウで設定したデータ型が表示され、設定はできません。 LREAL以外のデータ型の変数を指定できます。
ワード数	データ形式で「ASCII16BIT」を選択するか、STRING型に設定した変数を入力した場合のみ設定できます。 設定範囲：1～512
連続数	対象デバイスで設定したデバイスを先頭にして、連続する何個のデバイスをロギング・トレースするか設定します。 変数を登録している場合は配列型変数でしか設定できません。

参 考

構造体型変数を設定する場合は、メンバ名まで指定してください。

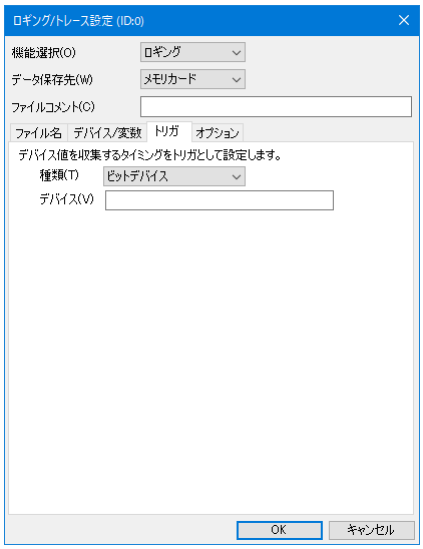


配列型変数を設定する場合は、配列名だけを入力すると
連続数を設定することで配列の要素を一括で設定できます。



トリガ

登録したデバイス/変数をロギング・トレースするタイミングを設定します。



項　目		説　明
種類※1		トリガの種類を以下から選択します。 「ビットデバイス」「周期」「トリガ命令(TRGD)」「データ取得トリガ(フロー)」
ビットデバイス		トリガとするビットデバイスもしくはBOOL/TIMER/COUNTER型の変数名を入力します。
周期	周期	周期(1～86400000)をms単位で設定します。
	ロギング開始時もトリガを発生	チェックすると、ロギング・トレース開始時もトリガを発生します。
トリガ命令		TRGD命令実行時にトリガを発生します。 「TRGD命令」5-76ページ

※1 **ビットデバイスの場合**
指定したビットデータがONのときにロギング・トレースを実行します。
ビットデータの状態はEND処理時点の状態を参照するため、1スキャン内でON/OFFされた場合はロギングは実行されません。
トリガを検出したEND処理時点のデバイス値を保存します。
ビットデータON中は、毎スキャン、END処理時点のデバイス/変数の値を保存します。

周期の場合
CPUユニット内蔵のタイマで、指定された時間間隔でロギング・トレースを実行します。
トリガを検出したEND処理時点のデバイス/変数の値を保存します。
スキャンタイムより短い時間が指定された場合は毎スキャン実行されます。
時間間隔の計測は、ロギング・トレース実行開始時点で開始されます。
PRGモードに移行すると、時間間隔の計測はリセットされます。

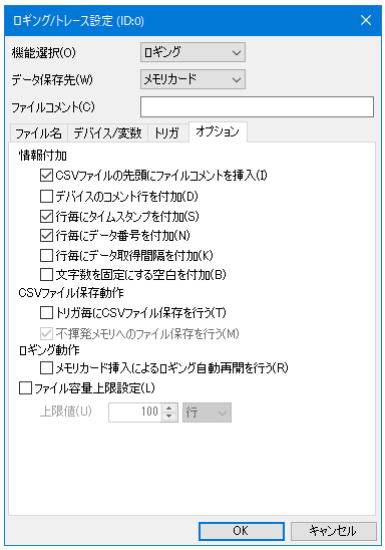
ポイント **トリガ発生直後のEND処理で指定したデバイス/変数の値が取り込まれます。このため、トリガ発生からデバイス/変数の値の保存まで最大1スキャンの遅れが生じます。**

TRGD命令の場合
END処理まで待たずに、TRGD命令が実行された時点でのデバイス/変数の値が取り込まれます。
1IDにつき8個までプログラムに記述できます。
割り込みプログラムや定周期モジュール/ユニット間同期モジュール内にも記述できます。

データ取得トリガ(フロー)の場合
X-Unit機能のユニットプログラム(フロー)において、ロギング/トレースブロックの「操作:データ取得トリガ」が実行開始した後のEND処理時点でのデバイス/変数の値が取り込まれます。値の取り込み後にブロックは終了します。
ユニットエディタで対応したユニットを配置して、ユニットプログラムを作成してください。

オプション

付加する情報や動作の詳細条件を設定します。トレース機能選択時に、ファイル保存先が「内部メモリ」の場合は設定できません。



項　目		説　明
情報付加※3	CSVファイルの先頭にファイルコメントを挿入	チェックすると、CSVファイルの先頭にファイルコメントを挿入します。
	デバイスのコメント行を付加※1	チェックすると、デバイスのコメントを挿入します。 コメントを設定していない場合、デバイス番号を付加します。
	行毎にタイムスタンプを付加※2	チェックすると、行毎にタイムスタンプを付加します。
	行毎にデータ番号を付加	チェックすると、行毎にデータ番号を付加します。ロギング・トレース開始時に0から始まります。ファイル番号の設定で「追記」を選択している場合は、ファイルの途中(追記が始まるところ)で0から始まります。 バッファオーバーフローで取りこぼしが発生したときは、このデータ番号を見れば、いくつ取りこぼしたかがわかります。 オーバーフローの対象になるバッファ容量は、ワークスペースの【CPUシステム設定】を開いて、【システムの設定】【CPUユニットバッファ容量設定】で設定します。
	行毎にデータ取得間隔を付加	直前のデータ取得からの経過時間を、1μs単位で記録します。
	文字数を固定にする空白を付加	チェックするとゼロサプレスが可能な数値の場合に空白を追加します。ファイルサイズは増加しますが、先頭文字位置を計算で求めることができます。
CSVファイル保存	トリガ毎にCSVファイル保存を行う	チェックすると、トリガ毎にメモリカード書込みを行います。 チェックしなかった場合は、トリガ毎の結果を内部のバッファ領域に格納し、LOGD命令実行時のみメモリカード書き込みを実行するため高速なロギングが可能です。
	不揮発メモリへのファイル保存を行う	チェックすると、トリガ毎に不揮発メモリに書込みを行います。
ロギング動作	メモリカード挿入によるロギング自動再開を行う	チェックすると、ロギング実行中にメモリカードを抜いた場合、次回メモリカードを挿入するまでCPUのメモリにロギングデータを蓄積し、メモリカードが挿入されると、自動的にメモリカードへの保存を再開します。自動再開時のメモリカードへの保存に関する動作は「開始時の動作」によります。 「ファイル名」(5-62ページ)

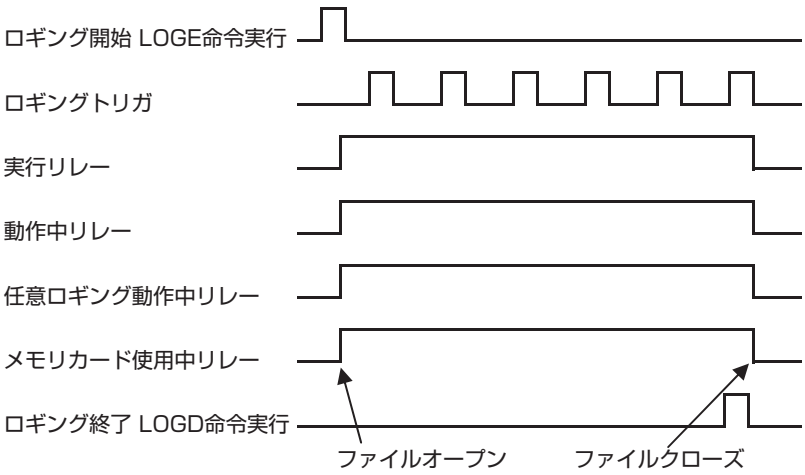
項　目	説　明
ファイル容量上限設定（ロギングのみ）	チェックすると、ファイル容量の上限値を設定できます。 「ファイル容量上限設定オーバー時の処理について」（5-84ページ）
上限値※6（ロギングのみ）	行単位またはKB単位で上限値を設定します。

※1　CPU本体にデバイスコメントが転送されていないデバイスには、デバイス番号（例えば、DM0）が入ります。
※2　タイムスタンプにはロギングした年月日時分秒のデータがCSV形式で記録されます。

例

2019/04/17,15:16:17

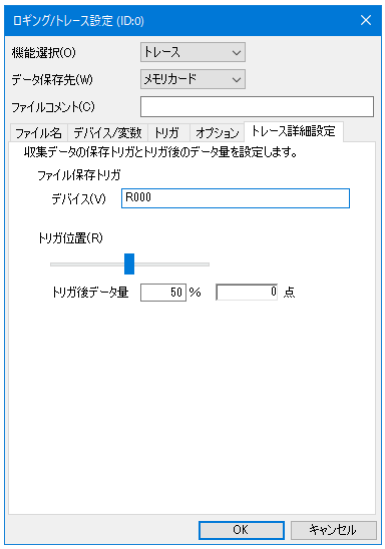
※3　「CSVファイル例」（5-72ページ）
※4　保存先がメモリカードの場合
ロギング実行中はファイルがオープンされたままになります。ファイルがオープンされた状態で電源がOFFすると、ロギングした結果がメモリカードに残りません。電源OFFする前に必ずLOGD命令を実行してください。
「トリガ毎にCSVファイル保存を行う」にチェックすると、ロギング実行中でもトリガ後の書込みさえ終了していれば、電源がOFFしてもロギングの結果が残ります。ただし、メモリカードに書込み中に電源がOFFするとメモリカードの内容が失われる危険があります。
また、トリガ毎にメモリカードへの書込みが発生するため、メモリカードにアクセスする時間が長くなります。そのため、バッファオーバーフローが発生しやすくなります。トリガの間隔が短い場合は、「トリガ毎にCSVファイル保存を行う」チェックを外してください。
保存先がCPUメモリの場合
常にトリガ毎にCSVファイル保存を行いますが、電源がOFFされてもCPUメモリに残るかどうかは、「不揮発メモリへのファイル保存を行う」のチェック状態により異なります。
チェックが入っている場合
トリガ毎にCPUメモリに保存されるため、電源をOFFしてもCPUメモリにファイルは残ります。ただし、アクセスする時間が長くなります。
チェックを外した場合
RAM上にデータを作成するため高速でロギングできます。ただし、電源OFF時にRAMモードで作成したデータはクリアされるため、電源をOFFする前にCR3200をONし、CPUメモリの保存処理要求を実行する必要があります。
「RAMモード」（5-47ページ）
※5　この機能は“自動再開”であって、“自動開始”ではありません。LOGE命令実行時にメモリカードが本体に挿入されていない場合は、ロギングエラーリレーがONして、ロギングは開始されません。
“自動再開”はメモリカードやCPUメモリが挿入されていて、ロギング実行中にメモリカードを抜き差ししたときに働く機能です。
※6　行数は、ファイルコメント、デバイスコメントも1行と数えます。
ファイルサイズは、ファイルコメント、デバイスコメントをすべて含めたサイズです。
「プログラム容量」（4-14ページ）



※オプション設定により各リレーの挙動は変わります。

トレース詳細設定

トレース機能を選択時のみ設定可能です。
ファイル保存時のトリガや保存方法を設定します。



項　目		説　明
ファイル保存トリガ	デバイス	ビットデバイスもしくはBOOL型の変数を指定します。※2
トリガ位置	トリガ後データ量(%)	トリガ後のデータ量を%で設定します。スライダーでも設定可能です。
	トリガ後データ量(点)※1	%で指定したトリガ後データ量を点数で表示します。

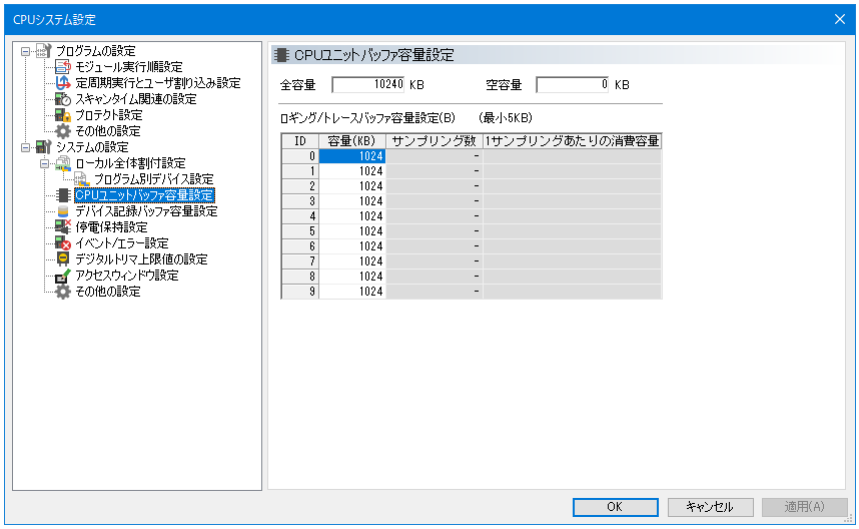
※1　保存するデータ量は、【CPUユニットバッファ容量設定】で設定します。詳細は「CPUユニットバッファ容量設定」(5-70ページ)」を参照してください。
※2　トリガの種類が「TRGD命令」の場合、TRGD命令・関数実行時にのみファイル保存トリガの状態を確認します。

参 考　トレース機能を使用して保存したトレースデータを、リアルタイムチャートモニタで読み出せます。
詳細は、「KV STUDIO ユーザーズマニュアル」を参照してください。

CPUユニットバッファ容量設定

ロギング機能/トレース機能でデータを保存するバッファの容量を設定します。

1 **ワークスペースから【CPUシステム設定】を開いて、【システムの設定】→【CPUユニットバッファ容量設定】の順にクリックします。**



2 **ID0～ID9で使用するバッファの容量を設定します。使用するIDの容量を大きくすると保存できるデータ量(サンプリング数)も大きくなります。バッファの全容量10240KBを超えない範囲で設定してください。**

CSVファイルについて

■ CSVファイルのフォーマット

ロギング・トレース機能で自動的に作成されるCSVファイルは、デバイスデータ、データ付加情報、ファイル付加情報からなります。文字数は半角で1文字、全角の場合は2文字としてカウントします。また、データ形式がASCII文字列の場合は左詰、それ以外の場合は右詰になります。

● ファイル付加情報

[ロギング/トレース設定]ダイアログボックスの〔オプション〕タブでの設定内容に応じて、以下の項目がファイルの先頭行から順に格納されます。

項目	1カラムの文字数※	例	備考
ファイルコメント	64	"タイトル"	1行目の長さは常に64文字です。文字数に「"」は含みません。
ファイル保存トリガ状態（トレースのみ）	32	"TRIGGER DATA NO" "COMM TRIGGER DATA NO"	文字数に「"」は含みません。発生位置と合わせて1行に記録されます。
ファイル保存トリガ発生位置（トレースのみ）	10	1234567890	ファイル保存トリガが発生したときのデータ番号が記録されます。
ロギングコメント	32	…,"起動リレー"　,…	文字数に「"」は含みません。

※　カンマを含みません。

● データ付加情報

〔オプション〕での設定内容に応じて、以下の項目がデバイスデータの左側から順に格納されます。

項目	1カラムの文字数※	例
データ番号	10	1234567890
タイムスタンプ（日付）	10	2019/04/17
タイムスタンプ（時刻）	8	11:12:13
データ取得間隔	10	1234567890

※　カンマを含みません。

● デバイスデータ

〔デバイス〕での設定に応じて、以下の文字数で格納されます。

項目	1カラムの文字数※	例	備考
ビットデバイス	1	…,1,…	ON:1　OFF : 0
±10進数16BIT	6	…,␣ -1234,…	正の場合、「＋」は付きません。
10進数16BIT	5	…,␣ 1234,…	―
±10進数32BIT	11	…,␣1234567890,…	正の場合、「＋」は付きません。
10進数32BIT	10	…,␣123456789,…	―
16進数16BIT	4	…,BA98,…	
16進数32BIT	8	…,BA987654,…	
FLOAT	13	…,␣␣␣2.4123E+12,…	指数形式固定です。小数点付きでロギングしたい場合は、ラダープログラムでFASC命令を使用して浮動小数点型のデータを文字列に変換し、ASCII16BITとしてロギングしてください。
ASCII16BIT	（ワード数）×2＋2	…,"a123"␣…␣,…	文字列の途中にNULLがあると、それ以降の文字列は記録されません。

■ CSVファイル例

データ番号　ファイルコメント　タイムスタンプ(ロギングした時刻)　データ取得間隔(μs)　ロギングコメント

製造ラインA寸法誤差

DATA No.	DATE	TIME	INTERVAL	製造個数	寸法誤差	累計個数	異常リレー
0	2019/04/17	11:20:09	0	851	-2	207326	0
1	2019/04/17	11:20:10	999987	852	-6	207327	0
2	2019/04/17	11:20:11	1000002	853	8	207328	0
3	2019/04/17	11:20:12	1000025	854	16	207329	0
4	2019/04/17	11:20:13	999980	855	31	207330	0
5	2019/04/17	11:20:14	1000003	856	22	207331	0
6	2019/04/17	11:20:15	999999	857	-38	207332	0
7	2019/04/17	11:20:16	999991	858	22	207333	0
8	2019/04/17	11:20:17	1000023	859	35	207334	0
9	2019/04/17	11:20:18	999998	860	-34	207335	0
10	2019/04/17	11:20:19	1000009	861	12	207336	0
11	2019/04/17	11:20:20	999984	862	21	207337	0
12	2019/04/17	11:20:21	1000001	863	-1	207338	0
13	2019/04/17	11:20:22	1000013	864	-37	207339	0
14	2019/04/17	11:20:23	999979	865	44	207340	0
15	2019/04/17	11:20:24	1000002	866	40	207341	0
16	2019/04/17	11:20:25	1000009	867	14	207342	0

← ファイルコメント行
← デバイスのコメント行
データ行

参考 デバイスのコメント行には、[ロギング/トレース設定]ダイアログで設定したロギングコメントが表示されます。ロギングコメントが空欄の場合は、デバイス名、もしくは変数名が表示されます。

ロギング命令

ロギングを実行するために、ロギング命令を使用したプログラムを作成します。

命令	ニモニック	動作概要	ページ
ロギング・トレース許可	LOGE	指定したIDのロギング・トレースを実行許可／禁止状態にします。	5-74
ロギング・トレース禁止	LOGD		5-74
ロギングデータ取得	TRGD	指定したIDのロギングデータを取得します。	5-76

■ ファイルサイズの計算

ファイルサイズの計算方法を以下に記載します。以下、単位はバイトです。

（ファイルコメント行のサイズ）
＋（デバイスコメント行のサイズ）
＋（データ行のサイズ）×（データ行の行数）
＝（ファイルのサイズ）

（ファイルコメント行のサイズ）＝（《KV STUDIO》で設定した文字数）＋4
半角文字は1文字、全角文字は2文字と数えます。

（デバイスコメント行のサイズ）　＝｛35×（デバイスコメントの個数）｝＋1
＝｛35×（（デバイスの個数）＋A＋B）｝＋1

「行毎にデータ番号を付加」　：　使用（A＝1）　不使用（A＝0）
「行毎にタイムスタンプを付加」：　使用（B＝2）　不使用（B＝0）

（データ行のサイズ）＝（デバイスのサイズの合計）＋1

デバイスのサイズ

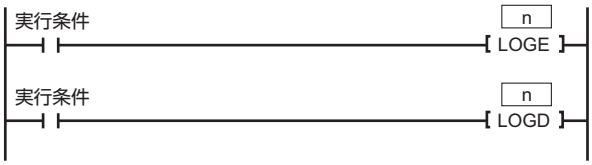
項　目			サイズ
ビットデバイス			2
ワードデバイス	1ワード	符号なし	6
		符号あり	7
	2ワード	符号なし	11
		符号あり	12
	浮動小数点型実数		14
	ASCII文字列（nワード）		（n×2）+3
データ番号			11
タイムスタンプ（日付＋時刻）			20
データ取得間隔			2～11

注　記	上記の計算から求めたファイルサイズと、メモリカード容量、CPUメモリ容量から記憶できるファイル数などの概略値を算出できますが、ディスクの使用状況などにより実際に保存できるファイル数などは変化します。十分に余裕をもってご使用ください。

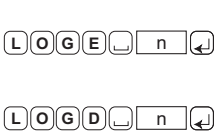
LOGE / LOGD

LOGE	-[LOGE]-	ロギング・トレース許可	指定したIDのロギング・トレースを実行許可／禁止状態にします。
LOGD	-[LOGD]-	ロギング・トレース禁止	

ラダープログラム



入力方法



オペランド	説明	占有サイズ
n	ロギング・トレースID(0～9)を指定します。※1 ※2	16ビット

※1 “$”での指定はできません。
※2 n にビットデバイスを指定した場合、連続した16ビットを占有します。
ワードデバイスを指定した場合は1ワードを占有します。

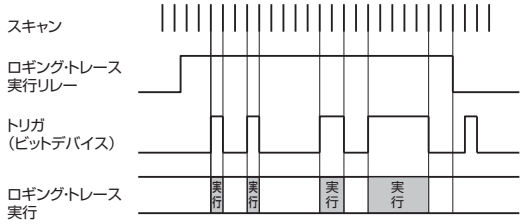
動作説明

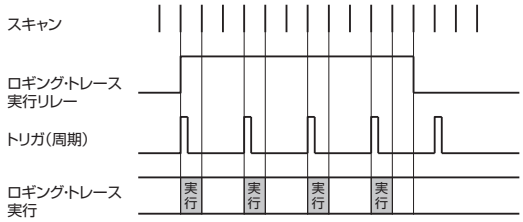
LOGE 実行条件がONすると、n で指定したIDのロギング・トレースを実行許可状態にします。命令実行時、指定したIDのCPUユニットバッファがクリアされ、ロギング・トレース実行リレーがセットされます。トリガが発生すると、ロギング・トレースが実行されます。
LOGE命令の実行条件の状態に関係なくLOGD命令が実行されるまで実行許可状態を継続します。
すべてのロギングIDのロギング・トレースを同じ実行条件で実行許可状態にできます。

LOGD 実行条件がONすると、n で指定したIDのロギング・トレースを実行禁止状態にします。命令実行後、IDごとのロギング・トレース実行リレーがリセットされ、ロギング・トレースが停止します。それまでのロギング内容がメモリカードまたはCPUメモリに出力されます。

ロギングID	ロギング実行リレー
0	CR000
1	CR100
2	CR200
3	CR300
4	CR400
5	CR500
6	CR600
7	CR700
8	CR800
9	CR900

ロギング・トレースのタイムチャート





演算フラグ

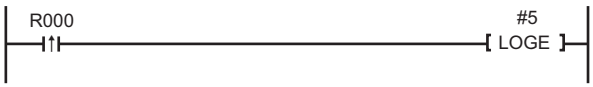
CR2009	変化なし
CR2010	変化なし
CR2011	変化なし
CR2012	n にデバイス指定時10以上を設定した場合、または間接指定・インデックス修飾の範囲が不適切な場合ON、それ以外OFF。オペランドに間接指定、インデックス修飾を指定していない場合は変化しません。

※ CR2012がONした場合、命令は実行されません。
CR2012がONした場合、CM5150～CM5176にエラーの詳細情報が格納されます。

「最新エラー詳細(CM)」(4-87ページ)

サンプルプログラム

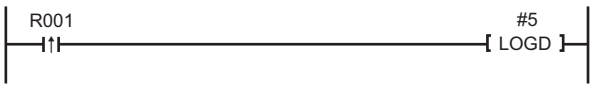
入力R000がONしたとき、ID5のロギング・トレースが実行許可状態になります。



〈ニモニックリスト〉

```
LDP R000
LOGE #5
```

入力R001がONしたとき、ID5のロギング・トレースが実行禁止状態になります。



〈ニモニックリスト〉

```
LDP R001
LOGD #5
```

TRGD

-[TRGD]- ロギングデータ取得

指定したIDのロギングデータを取得します。

ラダープログラム

入力方法

オペランド	説明	占有サイズ
n	取得するロギング・トレースID、またはそれを格納している先頭のデバイスを指定します。※1	

※1 ロギングIDは、0～10の範囲内で指定してください。リアルタイムチャートモニタや、KV COM+ for Excelのリアルタイムロギングで命令トリガを使用する場合は、「10」を指定します。

動作説明

TRGD 実行条件がONのとき、n で指定したロギング・トレースIDのデータを取得します。

•TRGD命令を実行した時点のデータが取得されます。

ビットデバイスや周期をトリガにしたロギング・トレースはEND処理のタイミングでデータを取得しますが、TRGD命令を使用することによってラダーの任意の時点でのデータを取得できます。

ポイント

- **TRGD命令でデータを取得するには、《KV STUDIO》のロギング・トレース設定で、トリガ種類を「トリガ命令（TRGD）」にしておく必要があります。**
- **リアルタイムチャートモニタで命令トリガを使用する場合は、「リアルタイムチャートモニタ」のトリガ設定でトリガの種類を「命令トリガ」に設定しておく必要があります。設定されていない場合に、TRGD命令を実行してもデータは取得されません。**
- **TRGD命令は、1つのプロジェクトにつき1IDごとに8個までしか使用できません。**

演算フラグ

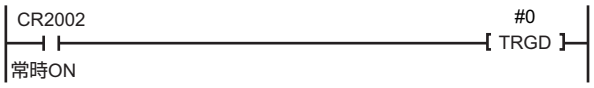
CR2009	変化なし
CR2010	変化なし
CR2011	変化なし
CR2012	• オペランドに指定した値が範囲外の場合はON。 • 間接指定、インデックス修飾の範囲が不適切な場合はON。 上記以外はOFF • n に定数を指定した場合は変化しません。

※ CR2012がONした場合、命令は実行されません。
CR2012がONした場合、CM5150～CM5176にエラーの詳細情報が格納されます。

「最新エラー詳細（CM）」（4-87ページ）

サンプルプログラム

定周期モジュール内に下記のラダープログラムを記述すれば、一定周期ごとにロギングを実行できます。



〈ニモニックリスト〉

```
LD CR2002
TRGD #0
```

ポイント　**定周期モジュールを実行する場合は、割り込み許可状態(EI命令を実行)にする必要があります。**

「定周期モジュール」(4-103ページ)
「4-13 割り込み」(4-206ページ)

ロギング・トレース用デバイス一覧

ロギング・トレース実行の制御情報やエラー情報がコントロールリレー、コントロールメモリに割り付けられています。
コントロールリレーにはロギング・トレースIDごとに1つずつ用意されているものと、システムに1つだけ用意されているものがあります。

ポイント **コントロールリレー・コントロールメモリの反映はEND処理で行われるため、ラダー中で操作をおこなってもEND処理まで結果は反映されません。**

■ ID個別のデバイス

ロギング・トレースID(0～9)ごとのデバイス番号					属性 R：読込み専用 空欄：読み書き可能	内　容
ID0	ID1	ID2	…	ID9		
CR000	CR100	CR200	…	CR900		ロギング・トレース実行
CR001	CR101	CR201	…	CR901	R	ロギング・トレース動作中
CR002	CR102	CR202	…	CR902	R	ロギング・トレースファイル書き込み終了 LOGE命令実行時に自動でOFFします。
CR003	CR103	CR203	…	CR903	R	ロギングバッファ50%未満警告
CR004	CR104	CR204	…	CR904	R	ロギングバッファオーバーフロー
CR005	CR105	CR205	…	CR905	R	ロギング実行中RUN中書き込みあり
CR008	CR108	CR208	…	CR908	R	ロギング・トレースエラー
CR009	CR109	CR209	…	CR909	R	メモリ容量なしエラー
CR010	CR110	CR210	…	CR910	R	デバイス・トリガ設定異常
CR011	CR111	CR211	…	CR911	R	トレースファイル保存トリガモニタ
CR012	CR112	CR212	…	CR912	R	トレースデータ取得完了
CM1630	CM1631	CM1632	…	CM1639	R	バッファ空き容量最小値(KB単位)※1
CM1680	CM1681	CM1682	…	CM1689	R/W	ファイル保存カウンタ (メモリカードのみ)
CM1500	CM1501	CM1502	…	CM1509	R	ファイル保存カウンタ(共通)
CM1660	CM1661	CM1662	…	CM1669	R	バッファ空き容量(KB単位)※2

※1 LOGE命令を実行してからのバッファの空き容量の最小値を表示します。次回のLOGE命令実行時にリセットされます。
※2 現在のバッファの空き容量を表示します。

■ 共通のデバイス

デバイス番号	属性 R：読込み専用 空欄：読み書き可能	内　容
CR3210	R	任意ロギング・トレース動作中
CR3513	R	内部バッファオーバーフローリレー
CR5400		メモリカードマウント要求
CR5401		メモリカードアンマウント要求
CR5402	R	メモリカード使用中
CR5403	R	メモリカード認識完了
CR5404	R	メモリカードあり
CR5405	R	メモリカードライトプロテクト
CR5408		CPUメモリマウント要求
CR5409		CPUメモリアンマウント要求
CR5410	R	CPUメモリ使用中
CR5411	R	CPUメモリ認識完了
CM2391	R	ストレージアクセス中電源OFF

■ 各デバイスの詳細

● ロギング・トレース実行リレー

ロギング・トレース実行の許可、禁止を行います。ロギング・トレース命令(LOGE)実行時、またはKV STUDIOでロギング・トレースを開始時にセットされますので、通常は直接操作する必要はありません。
このリレーがONすると、同じロギング/トレースIDのすべての状態リレーを一度OFFします。
エラーが発生した場合、リセットします。

● ロギング・トレース動作中リレー

実際にロギング・トレースが実行している間ONします。

● ロギング・トレースファイル書き込み完了リレー

ロギング使用時 : ファイル毎に指定したファイル容量上限値のデータを書き終えたときにONします。
トレース使用時 : ファイルまたは内部メモリにデータを書き終えたときにONします。
このリレーがONした場合ロギング・トレースは終了し、ロギング・トレース実行リレーがOFFします。
ロギング・トレースファイル書き込み完了リレーは、次回のLOGE命令実行時にOFFします。

「ファイル容量上限設定オーバー時の処理について」(5-84ページ)

● ロギングバッファ50%未満警告リレー

トリガ発生の頻度がファイル書き込み速度よりも速いために、バッファの空き容量が50%未満になったときにONします。ファイル毎に指定したファイル容量上限値のデータを書き終えたときにONします。

● バッファオーバーフローリレー

メモリカードまたはCPUメモリへのデータの書き込みが追いつかず、ロギング・トレース用のバッファがオーバーフローしたことを示します。
このリレーがONしたときは、データの取りこぼしが発生しています。データ番号を見れば、いつデータを取りこぼしたか確認できます。
オーバーフローが発生しても、メモリカードへの書き込みは継続されます。
バッファオーバーフローが発生するときは、

- メモリカードまたはCPUメモリへのアクセスを減らす
- データ量やトリガの頻度を下げる
- スキャンタイムを長くすることで軽減できます。

● ロギング・トレース実行中RUN中書込みありリレー

ロギング・トレース実行中にRUN中書き込みがあったときONします。
RUN中書き込み時、ロギング・トレースは動作を続けますが、メモリカードまたはCPUメモリへの書き込みが遅くなるため、バッファオーバーフローが発生しやすくなります。

● ロギング・トレースエラーリレー

ロギング・トレースが、メモリカードのエラーで終了したとき、ONしロギング・トレースは終了します。各種エラーリレーがONしていないか確認してください。このときバッファに残っているデータは無くなります。エラークリアは次回の「LOGE命令」実行時に行われます。

● メモリ容量なしエラーリレー

メモリカードまたはCPUメモリに十分な空き容量がないときにONします。
このリレーがONしたときは、ロギング・トレースは終了します。ロギング・トレースエラーリレーがONし、ロギング・トレース実行リレーがOFFします。このときバッファに残っているデータは無くなり、ロギングファイルは作成されません。エラークリアは次回の「LOGE命令」で行われます。

● デバイス・トリガ設定異常リレー

ロギング・トレースのデバイス設定、トリガ設定、その他設定が異常な場合にONします。
このリレーがONしたときは、ロギング・トレースが終了します。ロギング・トレースエラーリレーがONし、ロギング・トレース実行リレーがOFFします。

● トレースファイル保存トリガモニタリレー

トレース機能使用時に、ファイル保存トリガが発生するとONします。
次回のLOGE命令実行時にOFFします。

● トレースデータ取得完了リレー

トレース機能使用時に、ファイル保存トリガ後データ量で設定されたデータ量を取得し終わるとONします。
次回のLOGE命令実行時にOFFします。

● バッファ空き容量（KB単位）

現在のバッファの空き容量がKB単位で格納されています。KV-8000シリーズのバッファ容量は10240KBです。
ロギング使用時：バッファの余裕度合いが把握できます。値が小さくなると、バッファオーバーフローが発生する可能性が高くなります。
トレース使用時：ファイル保存トリガ発生前は、バッファ全体の容量を表します。ファイル保存トリガ発生後は、トレース完了までに取得する残りのデータ量を表します。

● ファイル保存カウンタ

ファイル保存が実行される度に1ずつインクリメントします。
0～65535の範囲でカウントされ、65535の次は0になります。電源断時も保存されます。
ファイル保存カウンタをリセットする場合は0を格納してください。LOGE命令実行時に自動的に0になりません。

● 任意ロギング・トレース動作中リレー

ロギング・トレースID0～9のいずれか1つ以上のロギング・トレースが、メモリカードに対して動作しているときONします。
他の機能を使用するときに、メモリカードへのロギング・トレースが動作しているかどうか確認するために用います。

● メモリカード使用中リレー

メモリカードを使用しているときにONします。ロギング・トレース機能に限らず何らかの機能がメモリカードを使用している間ONします。

● メモリカード認識完了リレー

メモリカードが挿入され、メモリカードスロットカバーが閉まっており、かつ初期化が完了し利用可能な状態になっているときにONします。

● メモリカードありリレー

メモリカードが挿入されており、かつメモリカードスロットカバーが閉まっている場合にONします。

● メモリカードライトプロテクトリレー

メモリカードが挿入されており、かつライトプロテクトがかかっている場合にONします。

● ストレージアクセス中電源OFF

ストレージアクセス中に電源がOFFするとエラーになり、0以外の値が格納されます。
値は自動的にクリアされません。

● 内部バッファオーバーフローリレー

内部で使用しているバッファ領域がオーバーフローした場合にONします。ロギング・トレース・リアルタイムチャートモニタ機能使用時に、大量のデバイスを高頻度で収集した場合などにONします。
収集しているデバイス数、またはサンプリング頻度を減らすことで軽減できます。

ロギング・トレース機能の注意事項

ロギング・トレース機能を使用する上での注意事項を以下に示します。

「CPUユニットで使用する」(5-8ページ)、「使用上の注意事項」(5-9ページ)もあわせてお読みください。

● 保存速度

- ロギング・トレース実行時のスキャンタイムは、最大で20μs×使用しているIDの数まで延びます。
- 複数のトリガが同時に発生する場合、メモリカードやCPUメモリに対するアクセス量が多くなるためスキャンタイムが延びたり、バッファオーバーフローが発生しやすくなります。

参考

データの取りこぼしが発生する場合に、一定スキャンタイム運転やEND処理時間の設定を使用して改善することができます。

「一定スキャンタイム運転」(4-90ページ)
「END処理時間の設定」(4-92ページ)

● ストレージへのアクセスの優先順位

- ロギング・トレース中、ストレージへのアクセスの優先順位は、以下の通りです。
  ストレージ命令 > ロギング・トレース命令 > FTPサーバ機能、FTPクライアント機能、アクセスウィンドウのファイル操作、KV-XD02からのアクセス
  このため、ロギング・トレースのトリガが大量に発生している場合、他の機能がストレージを使用できないことがあります。

● 電源断、カード抜き差し

メモリカードは、本体の電源をONしたまま、メモリカードアンマウント要求リレーをONし、メモリカード認識完了リレーがOFFになるのを確認して取り出してください。ロギング・トレース実行時は、この手順でメモリカードを取り出さないとデータが保存されません。

メモリカードスロットカバーの扱い

- メモリカードスロットカバーが閉まっていないときは、ロギング・トレースは開始されません。
- メモリカードスロットカバーを開くと、実行中の処理は継続されますが、それ以後の処理はメモリカードの処理を受け付けません。

電源断時、カードの抜き差しの注意

- メモリカードアクセス中LEDが点灯中(カードへのデータの読み込み・書き込み)は、絶対にカードを抜いたり、本体の電源を切らないでください。カード内のデータが破壊されることがあります。
- ロギング・トレース実行中に電源がOFFした場合、データはメモリカードに保存されません。
- CPUメモリアンマウント要求リレーONし、CPUメモリ認識完了リレーがOFFするのを待って電源をOFFしてください。アンマウントとは、メモリカードを安全に取り外し可能な状態にすることです。
- RAMモードで使用している場合、必要なデータは電源断を実行する前にCPUメモリ保存要求リレーをONし、CPUメモリ保存完了リレーがONになったのを確認して電源をOFFしてください。

カードの交換

- カード交換は、必ず手順にそって行ってください。
  「CPUユニットで使用する」(5-8ページ)
- 《KV STUDIO》のロギング・トレース設定で「メモリカード挿入によるロギング・トレース自動再開」を設定すると、ロギング・トレース実行中にメモリカードを抜いた場合、次回メモリカードを挿入するまで本体バッファにロギング・トレースデータを蓄積し、メモリカードが挿入されると、蓄積していたデータをメモリカードに書き込みます。このとき、ロギング・トレースエラーリレーはONしません。メモリカード挿入までの時間が長かったり、トリガ発生回数が多かったりすると、バッファオーバーフローが発生してデータが欠落することがあります。
  「メモリカード挿入によるロギング・トレース自動再開」を設定していない場合、ロギング・トレース実行中にメモリカードを絶対に抜かないでください。カード内のデータが破壊されることがあります。

● RUN中書き換え中の動作

- RUN中書換えを行っているとき、ロギング・トレースの実行が非常に遅くなります。このため、バッファオーバーフローが発生しやすくなります。
- ロギング・トレース設定をRUN中書き込みできますが、設定が有効になるのは次回ロギング・トレース動作開始時です。

● RUN/PRGモード切換時の動作

- ロギング・トレース機能はRUN状態のときのみ動作します。RUN→PRG切換を行うとすべてのロギング・トレースが停止します。
  LOGD命令が実行された場合と同様、それまでのロギング・トレース内容がメモリカード（またはCPUメモリ）に出力されます。
  また、PRG→RUN切換したときには、ロギング・トレースはすべて停止状態なので、開始するためにLOGE命令を実行する必要があります。

● ロギング・トレース開始時の処理について

電源を投入してロギング・トレース許可されたとき、指定されたファイル番号がすでに存在していると次のような動作になります。

ファイル番号の指定方法が「自動連番」のとき

- 「ファイル新規作成」で空き番号が無い場合の動作は「ファイル容量上限設定オーバー時の処理について」を参照してください
- 「最新のファイルに追記」で最新のファイルがファイル容量上限設定で設定した上限値を超えた場合：
  次の空き番号のファイルを使用します。空き番号が無い場合の動作は「ファイル容量上限設定オーバー時の処理について」を参照してください。
  ファイル容量上限設定をしていない場合は追記し続けます。
  「ファイル容量上限設定オーバー時の処理について」は次ページを参照してください。

ファイル番号の指定方法が「固定/デバイス値」のとき

- 「同一ファイル存在時」に「古いファイルを上書き」を選択している場合：
  古いデータを削除し、同一ファイル名で上書きします。
- 「同一ファイル存在時」に「追記」を選択している場合：
  指定されたファイルにデータを追記します。存在するファイルがすでにファイル容量上限設定で設定した上限値を超えているときにロギング・トレースを開始すると、ロギング・トレースエラーリレーがONし、ロギング・トレースを停止します。

「ロギング・トレース用デバイス一覧」（5-78ページ）

● ファイル容量上限設定オーバー時の処理について

ファイル容量の上限値を超えたときの動作は、ファイル番号の指定方法によって異なります。

ファイル容量上限値設定については、「オプション」(5-67ページ)参照

ファイル番号の指定方法が「自動連番」のとき

ファイル番号に空きがある場合は、次のファイル番号で新たなファイルを作成します。

ファイル番号がすべて使用済みの状態で、最後のファイルが容量オーバーしたとき

- 「空き番号がない場合」に「古いファイルから上書き」を選択している場合：
  最も古いデータを削除し、同一ファイル番号でファイルを新規作成します。
- 「空き番号がない場合」に「ロギング・トレース停止」を選択している場合：
  ロギング・トレース開始時にロギング・トレースエラーリレーがONし、ロギング・トレースを停止します。

ファイル番号の指定方法が「固定/デバイス」のとき

「同一ファイル存在時」に「古いファイルを上書き/追記」どちらを選択している場合も、ロギング・トレースを停止します。

参考　FTPクライアント機能で「ロギング/トレース転送設定」を設定している場合、LOGD命令を実行していなくても、ファイル容量上限設定オーバーで生成されたファイルは都度FTPアップロードされます。

● 使用できる文字の制限

①CSVファイル内の文字列　：ファイルコメント、デバイスコメント
②ファイル名の文字列　：ファイル名の文字列
③ASCII文字列　：デバイスのデータ形式で「ASCII文字列」を選択したとき
のそれぞれに使用できる文字には制限があります。

項　目	説　明
CSVファイル内の文字列	• 「,」は使用できません。
ファイル名の文字列	以下の文字は使用できません。 • 半角12文字　* / < > ? ¥ ｜ % , : ; " • 全角7文字　／ ： ？ ￥ ＊ ［ ］ • NEC特殊文字(8740H～879CH) ①、㌢、㍑、㍻、など • NEC選定IBM拡張文字(ED40H～EEFCH) • IBM選定IBM拡張文字(FA40H～FC4BH)
ASCII文字列	アスキーコードはすべてそのまま保存されます。制御文字や「"」などもそのまま保存されます。

● ファイル容量上限

1ファイルあたりのサイズ上限は256MBです。

● ストレージ容量上限

残り容量が10KB以下になると、メモリ容量なしエラーがONになることがあります。

MEMO

5-6 運転記録機能

運転記録の概要

運転記録機能を使用すると任意のタイミング(トリガ)前後のPLCの運転記録を保存できます。運転記録で保存できるデータは以下のとおりです。

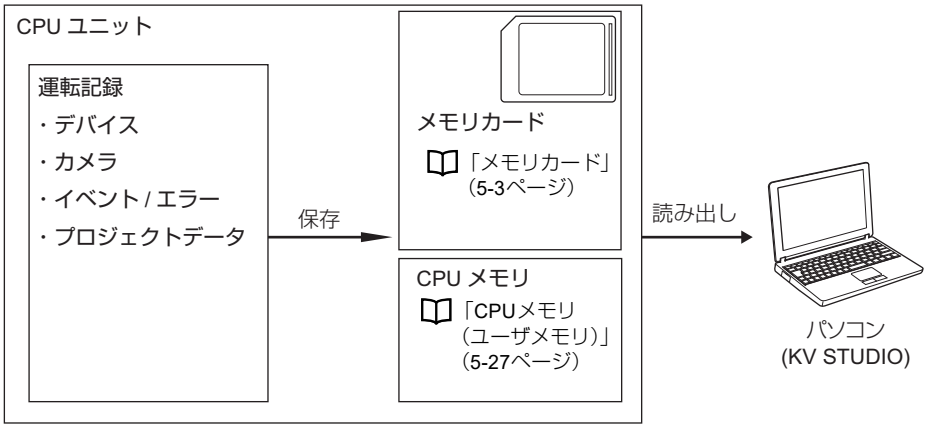
種類	詳細
デバイス	トリガ条件成立前後のPLCのデバイス、変数、拡張ユニットのバッファメモリの、スキャンタイムごとのデータ
カメラ	トリガ条件成立前後のフレームレートごとにKV-CA02で撮影した映像データ
イベント/エラー	トリガ条件成立前後のエラーやデバイス値変更などのイベント履歴
プロジェクトデータ	トリガ条件成立時のプロジェクトデータ

ポイント **運転記録機能は《KV-8000》でのみ使用できる機能です。**

保存した運転記録は、《KV STUDIO》または《KV REPLAY VIEWER》で再生できます。トラブル発生前後のビットデバイスのON/OFF状態やワードデバイスの値、カメラで撮影した映像※1をラダープログラム、リアルタイムチャートモニタ、リプレイVTビューワ※2で確認できます。時系列のデータとして再生・停止・巻き戻し・コマ送りなどを行いながら確認できるため、センサがチャタリングしていた、チャック前に完全に停止していなかったためにチャックミスが起こったなど、後からの確認が難しかったトラブル要因の解析に活用できます。

※1 小型標準カメラ《KV-CA1H》もしくは広視野・高解像度カメラ《KV-CA1W》およびカメラ入力ユニット《KV-CA02》が必要です。
※2 運転記録のデバイスデータをタッチパネルの画面データを使用してPC上でシミュレートする機能です。VT STUDIO Ver.7.10以降が必要です。

運転記録はCPUユニットに挿入したメモリカードまたはCPUメモリに保存でき、CPUユニット経由で読み出したり、パソコンなどにコピーしたファイルを開いたりすることで内容を確認できます。



ポイント
- **保存した運転記録は《KV STUDIO》のリプレイモード、または《KV REPLAY VIEWER》で内容を確認できます。リプレイモードの詳細は『KV STUDIO ユーザーズマニュアル』を参照してください。**
- **カメラの映像データはカメラ入力ユニット《KV-CA02》に記録され、トリガ条件成立時にCPUユニットに保存されます。詳細は『KV-CA02ユーザーズマニュアル』を参照してください。**
- **データ活用ユニット《KV-XD02》を使用すると、運転記録の内容を分析し、トラブル要因となった可能性が高いデバイス/変数を表示することができます。詳細は『KV-XD02ユーザーズマニュアル』を参照してください。**

使用手順

運転記録機能の使用手順は以下のとおりです。

① 保存するデータとトリガの設定

【運転記録/リプレイ(R)】－【運転記録設定(P)】

運転記録として保存するデータの種類と、保存する条件、期間を設定します。保存するデータはユニット設定やラダープログラムの内容から自動で設定され、手動で追加･削除することも可能です。

「設定」(5-88ページ)

▼

② カメラ入力ユニットの設定

カメラの設定

カメラの映像を記録する場合のみ必要です。

詳細は『KV-CA02ユーザーズマニュアル』を参照してください。

▼

③ 運転記録の収集、保存

設定をPLCに転送します。設定したファイル保存トリガの条件が成立すると、CPUユニットに運転記録が保存されます。

▼

④ 運転記録の再生

保存した運転記録を《KV STUDIO》のリプレイモードで、または《KV REPLAY VIEWER》で再生します。リプレイモードの使用方法は、『KV STUDIO ユーザーズマニュアル』を参照してください。

設定

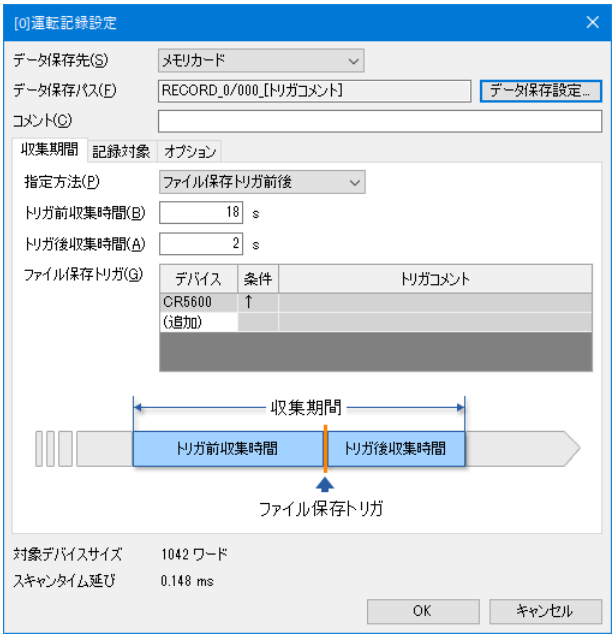
運転記録機能を使用するためには、《KV STUDIO》で以下の設定を行います。

- データ保存先
- データの取得方法
- ファイル保存トリガ
- データの収集時間
- データの収集開始タイミング
- 記録対象

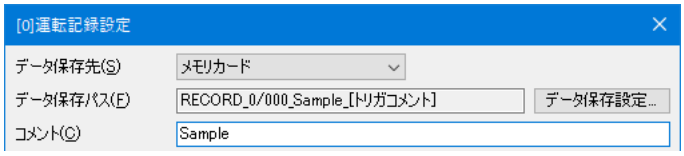
運転記録設定は最大4個（ID0～ID3）まで作成できます。IDごとに運転記録設定を行います。

1 メニューから【運転記録/リプレイ(R)】▶【運転記録設定(P)】▶【新規作成(N)】をクリックします。
[運転記録設定]ダイアログが表示されます。

別手順 ・ワークスペースの「運転記録設定」をダブルクリック



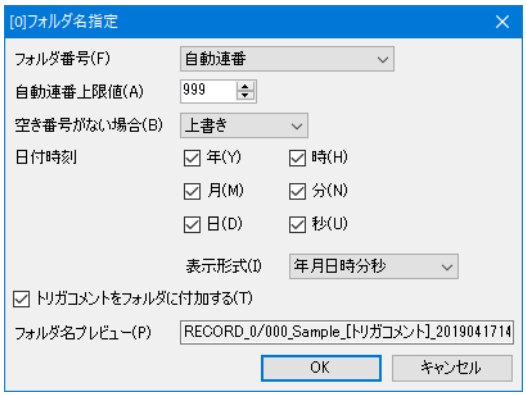
2 データ保存先やフォルダ名に関する設定を行います。



項目	説明
データ保存先	「メモリカード/CPUメモリ」から選択します。
データ保存パス	データ保存パスを設定します。 データ保存設定... をクリックすると[フォルダ名指定]ダイアログが表示されます。
コメント	保存ファイル名に挿入するコメントを設定します。

ポイント

コメントには一部使用できない文字があります。使用できない文字については、「データ保存先パスで使用できない文字」(5-109ページ)を参照してください。



項目			説明
フォルダ番号	自動連番	自動連番上限値	0～自動連番上限値で設定した値の間で未使用の番号から最小のフォルダ番号を使用します。
		空き番号が無い場合	空き番号がない場合の処理を「上書き/停止」から選択します。
	固定	番号	0～999の固定値で指定します。
	デバイス値	デバイス	デバイス、もしくはUINT/UDINT/TIMER/COUNTER型の変数の値(0～999)を使用します。
日付時刻			チェックすると、ファイル名に運転記録保存時の日時情報が含まれます。日時情報はファイル新規作成時点での日時情報に固定されるため、追記時にはファイル名は変更されません。既存ファイルを上書きしたときには古いファイルが削除されるため、新しい日時情報のファイル名でファイルが作成されます。 年(2000～2099)　月(01～12)　日(01～31) 時(00～23)　分(00～59)　秒(00～59)
トリガコメントをフォルダに付加する			チェックするとトリガコメントをフォルダ名に付加します。
フォルダ名プレビュー			フォルダ名の一例が表示されます。

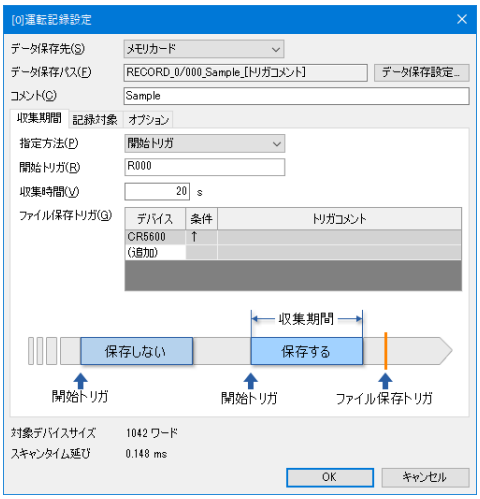
ポイント

自動連番上限値に到達する前にメモリカードの空き容量が少なくなると、追加で保存できなくなります。その場合、自動連番上限値を小さくし、空き番号がない場合の処理を「上書き」としておくと保存処理を継続できます(古いファイルから削除されていくのでご注意ください)。

収集期間

データを収集するタイミングや時間に関する設定を行います。

「指定方法」に「ファイル保存トリガ前後」を選択した場合と、「開始リレー」を選択した場合とで設定項目が一部異なります。



項目		説明
指定方法		「ファイル保存トリガ前後/開始リレー」から選択します。
「指定方法」が「ファイル保存トリガ前後」の場合	トリガ前収集時間	トリガ前のデータを保存する時間を設定します。
	トリガ後収集時間	トリガ後のデータを保存する時間を設定します。
「指定方法」が「開始リレー」の場合	開始リレー	データの収集を開始するリレーを設定します。
	収集時間	データを収集する時間を設定します。
ファイル保存トリガ	デバイス	収集したデータを保存するためのトリガとするビットデバイスもしくはBOOL/TIMER/COUNTER型の変数を指定します。
	条件	「↑」(立ち上がり)、「↓」(立ち下がり)から選択します。
	トリガコメント	ファイル保存トリガのコメントを設定します。設定したコメントは保存フォルダ名に付加できます。

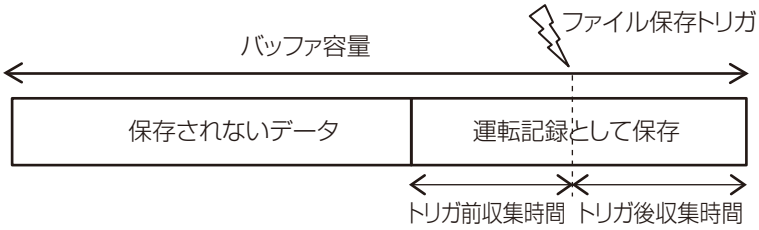
ポイント

トリガコメントには一部使用できない文字があります。使用できない文字については、「データ保存先パスで使用できない文字」(5-109ページ)を参照してください。

■「ファイル保存トリガ前後」と「開始リレー」の動作

●「ファイル保存トリガ前後」を選択した場合

運転記録設定の各IDに割り当てたバッファ容量分のデータを常に収集します。ファイル保存トリガの条件が成立しトリガ後収集期間が経過すると、データの収集を停止し、トリガ前収集時間とトリガ後収集時間分のデータを運転記録として保存します。トリガ前収集時間以前のデータは保存されません。
エラーリレーの立ち上がり前後の運転記録を確認する場合などに選択します。

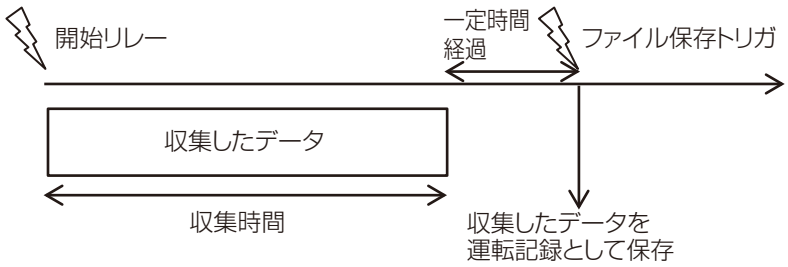


ポイント

- **トリガ前収集時間で指定した時間分のデータを収集していない状態でファイル保存トリガの条件が成立した場合、収集できたデータのみを運転記録として保存します。**
- **バッファ容量分のデータを収集すると、先頭に保存したデータから順番に上書きされていきます。**

●「開始リレー」を選択した場合

開始リレーで指定したデバイスがONすると、データの収集を開始し、設定した収集時間が経過するとデータの収集を停止します。完了信号、または次の開始リレーがONするまでの時間が一定時間を超えたことをファイル保存トリガの条件としておくと、ファイル保存トリガの条件が成立し、収集時間分のデータを運転記録として保存します。
設備が無応答で停止した場合など、エラーリレーとして検出できないような場合に選択します。



参 考

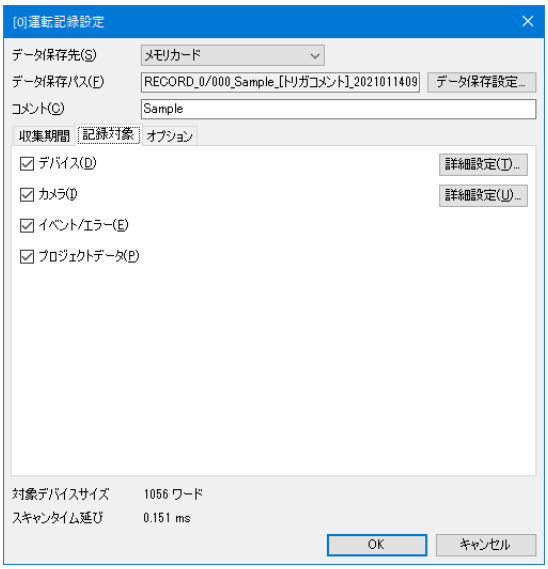
運転記録バッファ領域全保存モード（ID0：CR5614）がONしている状態でファイル保存トリガが成立するとバッファ容量分の全デバイスデータを保存できるため、収集期間以外のデバイスデータも確認できます。

ポイント

- **開始リレーがONした時点で収集したデータが存在する場合、収集したデータを破棄してからデータの収集を開始します**
- **収集したデータが存在しない状態でファイル保存トリガの条件が成立した場合、運転記録は保存されません。**
- **開始リレーのON後、収集時間が経過する前にファイル保存トリガが成立した場合、収集時間分のデータを収集してから運転記録を保存します。**

記録対象

運転記録として保存するデータの対象を設定します。



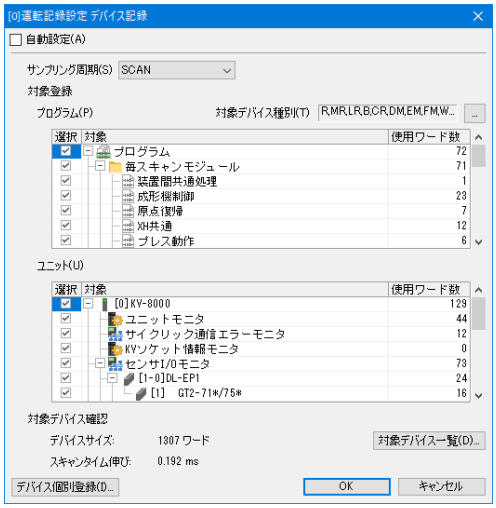
項目	説明
デバイス	PLCのデバイス、変数を保存します。初期値ではユニット構成やプロジェクトで使用されているデバイスを自動で一括登録しています。デバイスを手動で登録する場合は、〔詳細設定(T)〕をクリックします。
カメラ	カメラ入力ユニット《KV-CA02》で撮影した映像データを保存します。初期値では接続しているすべてのカメラの映像データを収集時間分だけ保存します。保存するカメラや収集時間を変更する場合は、〔詳細設定(U)〕をクリックします。
イベント/エラー	PLCで発生したイベント、エラーなどの履歴を保存します。保存できる履歴は以下のとおりです。 ・エラー履歴（重度エラー/軽度エラー） ・電源ON/OFF履歴 ・PLCイベント履歴 ・デバイス値変更履歴
プロジェクトデータ	ファイル保存トリガ発生時のプロジェクトデータを保存します。保存したプロジェクトデータは《KV STUDIO》のリプレイモード、または《KV REPLAY VIEWER》でラダーを表示する際のプロジェクトとして使用されます。

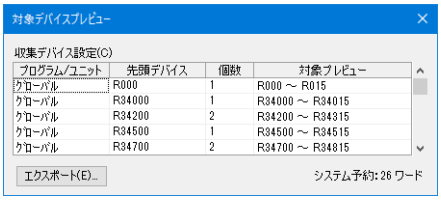
ポイント

- **記録対象にデバイスを含む場合、スキャンタイムが延びます。スキャンタイム延びの目安時間は[運転記録設定]ダイアログの左下に表示されます。スキャンタイム延びを抑えたい場合は、手動で設定し、登録数を少なくしてください。**
- **スキャンタイムが延びるのはデータ収集中のみです。「指定方法」が「ファイル保存トリガ前後」の場合はファイル保存中の期間以外、「指定方法」が「開始リレー」の場合は開始リレーがONしてから収集時間が経過するまでが対象です。それ以外のタイミングではスキャンタイムは延びません。**
- **登録するデバイス数を変更せずにスキャンタイムを一定にしたい場合は、一定スキャンタイム運転機能を使用してください。**
  「一定スキャンタイム運転」（4-90ページ）

■ デバイス詳細

〔詳細設定(T)〕をクリックすると、下記のダイアログが表示されます。〔詳細(D)〕をクリックすると、選択されているデバイス、変数の一覧を表示します。





項目	説明	
自動設定	チェックを外すと、収集対象のデバイス、変数をユニットの機能やモジュールごとに手動で設定できます。初期値ではユニット設定やラダープログラムで使用されているデバイス、変数が一括で登録されています※1。	
サンプリング周期	「SCAN/定期周期/トリガ命令(TRGR)」から選択します。	
	SCAN	END処理のタイミングで、スキャンごとにデータを収集します。
	定周期	選択すると右側に表示される、「周期」で設定した期間経過後の最初のEND処理でデータを収集します。
	トリガ命令	TRGR命令の実行タイミングでデータを収集します。

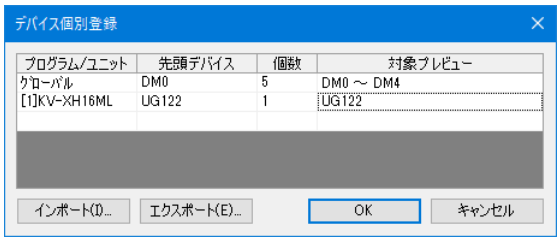
※1 対象範囲が不定な命令語で使用されているデバイス、変数やインデックス修飾、間接指定で指定されているデバイス、変数は自動で登録できません。これらのデバイス、変数のデータを記録する場合は手動で登録します。
例:「FMOV #0 DM0 EM0」の場合、EM0の値によって対象となるデバイスの範囲が変化するため、自動で登録できません。

参 考

保存するデータ数によって最大収集期間は変化します。最大収集期間はKV STUDIOがモニタモードのときに運転記録モニタから確認できます。最大収集期間を確認し、必要に応じてデバイス記録バッファ容量を大きくする、保存するデバイス、変数を減らす、などで調整してください。

「デバイス記録バッファ容量設定」(5-97ページ)

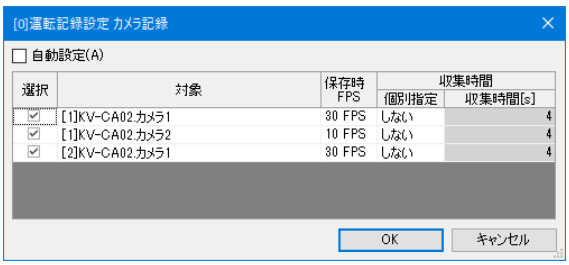
保存するデバイス、変数を手動で設定する場合は「自動設定(A)」のチェックを外した状態で〔デバイス個別登録(I)〕をクリックします。



項目	説明
プログラム/ユニット	保存するデバイス、変数の種別をプルダウンリストから選択します。
先頭デバイス	保存するデバイス、変数の先頭デバイスを指定します。
個数	複数のデバイス、変数を連続で保存する場合に連続個数を指定します。
対象プレビュー	「先頭デバイス」、「個数」で設定した結果を表示します。
インポート	保存するデバイス、変数の設定情報をインポートし、現在の設定を上書きします。
エクスポート	保存するデバイス、変数の設定情報をエクスポートします。

■ カメラ詳細

〔詳細設定(U)〕をクリックすると、下記のダイアログが表示されます。手動で設定する場合は、「自動設定(A)」チェックを外します。



項目		説明
選択		チェックすると、対象のカメラの撮影データを保存します。
保存時FPS		カメラの撮影データを保存するときのFPSを設定します。
収集時間	個別指定	「しない/する」から選択します。「する」を選択した場合、「収集時間」を手動で設定できるようになります。「しない」を選択した場合、運転記録の収集時間分の映像を収集します。
	収集時間	「個別指定」を「する」に設定した場合にのみ設定できます。運転記録設定の収集時間とは別に、カメラの撮影データの収集時間を設定します。

■ イベント/エラー詳細

運転記録機能で保存できるPLCのイベントやエラー履歴について説明します。

● エラー履歴

PLCで発生しているエラーの履歴を保存します。重度エラーの履歴は32件、軽度エラーの履歴は200件まで保存できます。

● 電源ON/OFF履歴

PLCの電源ON/OFF履歴を保存します。電源ON履歴、電源OFF履歴はそれぞれ20件まで保存できます。

● PLCイベント履歴

PLCで発生した以下のイベントの履歴を保存します。

イベント	対象ユニット	内容
運転モード切換	CPUユニット	運転モード(PROG⇔RUN)が切り換わりました。
プロジェクト変更	CPUユニット	KV STUDIOからのプロジェクト転送、もしくはメモリカード、CPUメモリからのプロジェクトロードによりプロジェクトが変更されました。
運転記録保存トリガ発生	CPUユニット	運転記録保存トリガが発生しました。
運転記録保存完了	CPUユニット	運転記録の保存が成功、または失敗しました。
メモリカード	CPUユニット	メモリカードスロットカバーがオープンまたはクローズしました。メモリカードがマウントまたはアンマウントされました。
カレンダタイマ設定変更	CPUユニット	カレンダタイマの設定が変更されました。
アラーム	CPUユニット	アラームリレーの状態が変化しました。
PLCクリア	CPUユニット	PLCの各種履歴がクリアされました。
イベントバッファオーバーフロー	CPUユニット	イベントが連続で発生したため、履歴の記録が追いつきませんでした。
イーサネット通信	CPUユニット KV-XD02 KV-XLE02 KV-XCM02	イーサネットのリンク状態(正常↔異常)が変化しました。
シリアル通信エラー	KV-XL202 KV-XL402	シリアル通信でパリティエラー、フレーミングエラー、オーバーランエラーのいずれかが発生しました。
シリアル通信エラー	KV-XL202 KV-XL402	シリアル通信でチェックサム異常が発生しました。
アプリイベント	KV-XD02	兆候監視アプリケーションで注意/警報が発報されました。
ユーザログイン	CPUユニット	ユーザがログインしました。
ユーザログアウト	CPUユニット	ユーザがユーザログアウトしました。
ユーザ認証機能有効化	CPUユニット	ユーザ認証機能を有効にしました。
ユーザ認証機能無効化	CPUユニット	ユーザ認証機能を無効にしました。
ユーザ追加	CPUユニット	ユーザを追加しました。
ユーザ削除	CPUユニット	ユーザを削除しました。
ユーザパスワード変更	CPUユニット	ユーザのパスワードを変更しました。
ユーザ権限設定変更	CPUユニット	ユーザのアクセスレベルを変更しました。
IPアドレス変更	CPUユニット	IPアドレスを変更しました。
システムプログラム更新	CPUユニット	システムプログラムを更新しました。
KVS接続(TLS)認証情報更新	CPUユニット	TLSの秘密鍵を更新しました。
OPC UA認証情報更新	CPUユニット	OPC UAの秘密鍵を更新しました。
OPC UAサーバ再起動	CPUユニット	OPC UAサーバが再起動しました。

● デバイス値変更

タッチパネルや他PLC、パソコンなどの外部機器からデバイス値が変更された履歴を保存します。

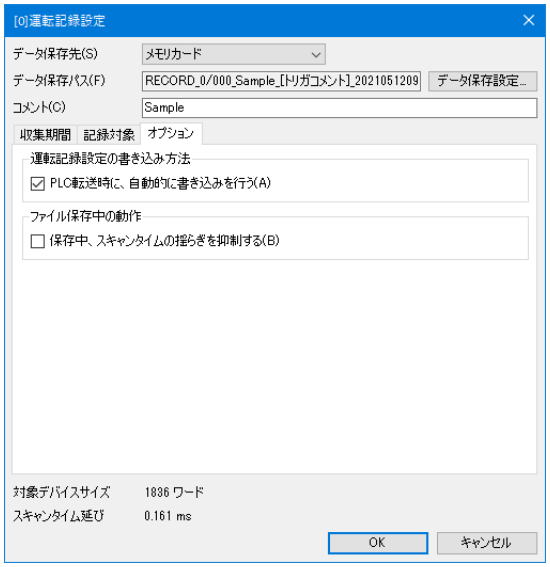
対象ユニット	内容
VT5/VT3	CPUユニットもしくは拡張ユニットに接続されたVT5/VT3からのデバイス値の変更
アクセスウィンドウ	CPUユニットのアクセスウィンドウの操作によるデバイス値の変更
Modbusサーバ	KV-XLE02/XCM02/XL202/XL402のModbusサーバ機能を使用して接続している外部機器からのデバイス値変更
KV上位リンク	KV-8000(A)もしくはKV-XLE02/XCM02/XL202/XL402のKV上位リンク機能を使用して接続している外部機器からのデバイス値変更
OPC UAサーバ	KV-8000(A)のOPC UAサーバ機能を使用して接続している外部機器からのデバイス値変更

■ プロジェクトデータ

PLCに転送されているプロジェクトデータを保存するかどうかを設定します。チェックをした場合はプロジェクトデータを運転記録として保存します。運転記録の容量を小さくする場合はチェックを外します。チェックを外した場合は、リプレイモードで再生する際にプロジェクトデータを用意する必要があります。

オプション

オプションを設定します。



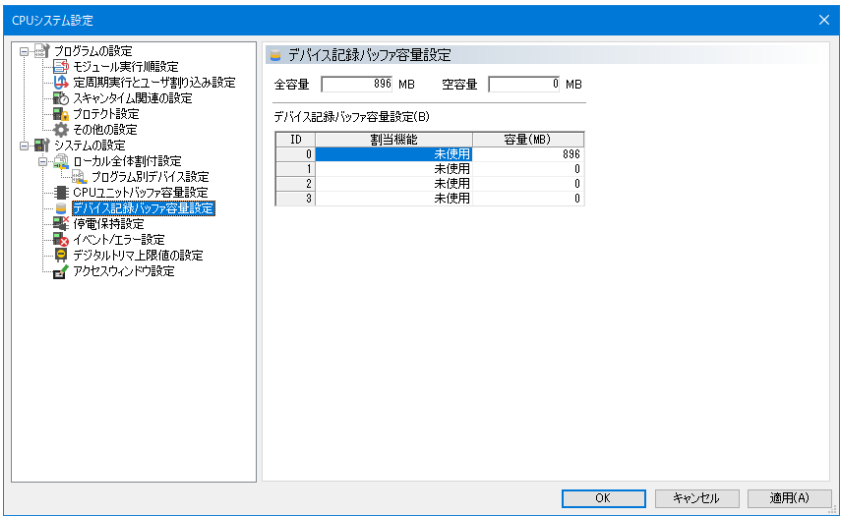
項目	説明
PLC転送時に、自動的に書き込みを行う	チェックすると、プロジェクト転送時に運転記録設定を自動的に書き込みます。 チェックを外すと、収集対象のデバイス･変数の情報を更新しなくなり、複数人でプロジェクトを設計しているときに運転記録設定の上書きを防げます。
保存中、スキャンタイムの揺らぎを抑制する	運転記録の保存中にスキャンタイムが速くなるのを抑制し、スキャンタイムをなるべく一定に保ちます。

デバイス記録バッファ容量設定

運転記録機能使用時に、デバイス値を記録するために使用するバッファ容量を設定します。

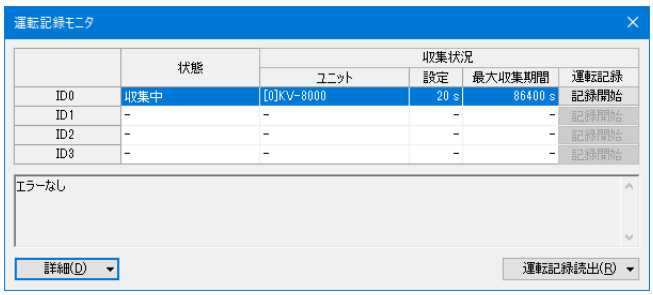
1 メニューから【表示(V)】▶【CPUシステム設定(P)】をクリックし、「デバイス記録バッファ容量設定」を選択します。

別手順 ・ワークスペースの「CPU　システム設定」をダブルクリックし、「デバイス記録バッファ容量設定」を選択



2 ID0～ID3の運転記録設定で使用するバッファの容量を設定します。使用するIDのバッファ容量を大きくすると、保存できるデータの量(データ収集時間)も大きくなります、バッファの全容量が896MBを超えないように設定してください。

参 考 KV STUDIOがモニタモードの場合、メニューの【運転記録/リプレイ(R)】▶【運転記録モニタ(M)】から表示できる運転記録モニタでデバイスの最大収集期間(単位:秒)を確認できます。最大収集期間が設定した値より短い場合は、バッファ容量を大きくする、収集するデバイスの量を減らす、など必要に応じて微調整してください。



運転記録機能用デバイス一覧

運転記録機能の実行情報やエラー情報がコントロールリレー(CR)、コントロールメモリ(CM)に割り付けられています。コントロールリレーにはIDごとに用意されているものと、全ID共通で1つだけ用意されているものがあります。

■ ID個別のデバイス

IDごとのデバイス番号				属性 R:読み込み専用 空欄:読み書き可能	デバイス名
ID0	ID1	ID2	ID3		
CR5600	CR5700	CR5800	CR5900		運転記録保存要求
CR5601	CR5701	CR5801	CR5901	R	運転記録動作中
CR5602	CR5702	CR5802	CR5902	R	運転記録保存トリガモニタ
CR5603	CR5703	CR5803	CR5903	R	運転記録保存中
CR5604	CR5704	CR5804	CR5904	R	運転記録収集中
CR5605	CR5705	CR5805	CR5905	R	運転記録設定異常
CR5607	CR5707	CR5807	CR5907	R	運転記録保存完了
CR5608	CR5708	CR5808	CR5908	R	運転記録保存エラー
CR5609	CR5709	CR5809	CR5909	R	デバイス記録バッファ不足
CR5614	CR5714	CR5814	CR5914		運転記録バッファ領域全保存モード
CM2000～ CM2001	CM2010～ CM2011	CM2020～ CM2021	CM2030～ CM2031	R	デバイス記録平均サンプリング周期
CM2002～ CM2003	CM2012～ CM2013	CM2022～ CM2023	CM2032～ CM2033	R	デバイス記録平均変化デバイス数
CM2004～ CM2005	CM2014～ CM2015	CM2024～ CM2025	CM2034～ CM2035	R	デバイス記録最大収集期間
CM6000	CM6200	CM6400	CM6600	R	運転記録保存完了コード
CM6002	CM6202	CM6402	CM6602		運転記録保存カウンタ
CM6004	CM6204	CM6404	CM6604	R	運転記録保存進捗詳細1
CM6005	CM6205	CM6405	CM6605	R	運転記録保存進捗詳細2
CM6006	CM6206	CM6406	CM6606	R	運転記録保存進捗
CM6010	CM6210	CM6410	CM6610	R	運転記録保存パス
:	:	:	:		
CM6171	CM6371	CM6571	CM6771		

■ ID共通のデバイス

デバイス番号	属性 R:読み込み専用 空欄:読み書き可能	デバイス名
CR5402	R	メモリカード使用中
CR5403	R	メモリカード認識完了
CR5404	R	メモリカードあり
CR5405	R	メモリカードライトプロテクト
CR5410	R	CPUメモリ使用中
CR5411	R	CPUメモリ認識完了

■ 各デバイスの詳細

● 運転記録保存要求リレー

運転記録保存要求リレーがOFF→ONするとCPUユニットは収集したデータを運転記録として保存します。運転記録機能でファイル保存トリガを設定していない場合でも、運転記録保存要求リレーをOFF→ONすると運転記録を保存できます。ラダープログラムから実行する場合はSET命令を使用してください。

ポイント
- **CRにBOOLの配列型変数を割り当てた場合、配列の要素番号は数値で指定してください。デバイスや変数で指定した場合、SET命令が正しく動作しません。**

● 運転記録動作中リレー

運転記録機能が動作している間ONします。

● 運転記録保存トリガモニタリレー

運転記録動作中にファイル保存トリガが発生すると、ONします。運転記録の保存が完了、または失敗すると、OFFします。

● 運転記録保存中リレー

運転記録の収集が完了し、保存を開始するとONします。運転記録の保存が完了、または失敗すると、OFFします。

● 運転記録収集中リレー

運転記録を収集している間、ONします。

● 運転記録設定異常リレー

運転記録機能の設定値が範囲外のときにONします。

● 運転記録保存完了リレー

運転記録の保存が完了するとONします。次のファイル保存トリガがONすると、OFFします。
KV-XD02のいつもと違う分析が動作している場合は、運転記録の保存完了後、いつもと違う分析が完了したタイミングでONします。

● 運転記録保存エラーリレー

運転記録の保存が失敗するとONします。次のファイル保存トリガがONすると、OFFします。

● デバイス記録バッファ不足リレー

バッファ容量が不足し、設定した収集時間分のデータを収集できないときにONします。

● 運転記録バッファ領域全保存モードリレー

このリレーをONした状態で運転記録の保存を開始すると、設定した収集時間分だけではなく、バッファに存在するすべてのデータを保存します。

● デバイス記録平均サンプリング周期

デバイス記録のサンプリング周期(単位:μs)の平均値が格納されます。

● デバイス記録平均変化デバイス数

デバイスの状態に変化があったデバイスの数(ワード数)の平均値が格納されます。

● デバイス記録最大収集期間

現在の設定で収集可能な期間(単位:秒)が格納されます。収集するデバイスの状態が変化すると収集可能な期間は短くなるため、同じ設定でもモニタ中に値が変化することがあります。

● 運転記録保存完了コード

運転記録の保存が完了すると、値が格納されます。保存完了時には0が格納され、保存失敗時には失敗要因に関する完了コードが格納されます。完了コードの内容は 「運転記録保存完了コード」(5-103ページ)を参照してください。

● 運転記録保存カウンタ

運転記録の保存が完了すると、1ずつ加算されます。0～65535の範囲で加算され、65535の次は0になります。電源断時にも値が保持されます。リセットする場合は、0を格納します。

● 運転記録保存進捗詳細1

保存実行中のファイル種別が格納されます。番号とファイル種別の関係は以下のとおりです。

番号	ファイル種別	内容
0	なし	保存中ではありません。
1	CPU履歴	CPUのイベント/エラー履歴です。
2	ラダープロジェクト	保存時にCPUユニットで実行していたプロジェクトです。
3	デバイス記録	運転記録設定で設定したデバイスデータです。
4	カメラ記録	KV-CA02で撮影した映像データです。
5	Savelog.txt	トリガ種別、時刻、フォルダ作成日時、ファルダ名、データサイズを記載したテキストファイルです。運転記録の保存に失敗した場合、Savelog.txtは保存されません。

● 運転記録保存進捗詳細2

運転記録保存の進捗がファイル種別ごとに0～100の範囲で格納されます。

● 運転記録保存進捗

運転記録保存の進捗が0～100の範囲で格納されます。

● 運転記録保存パス

運転記録の保存先のパスが格納されます。

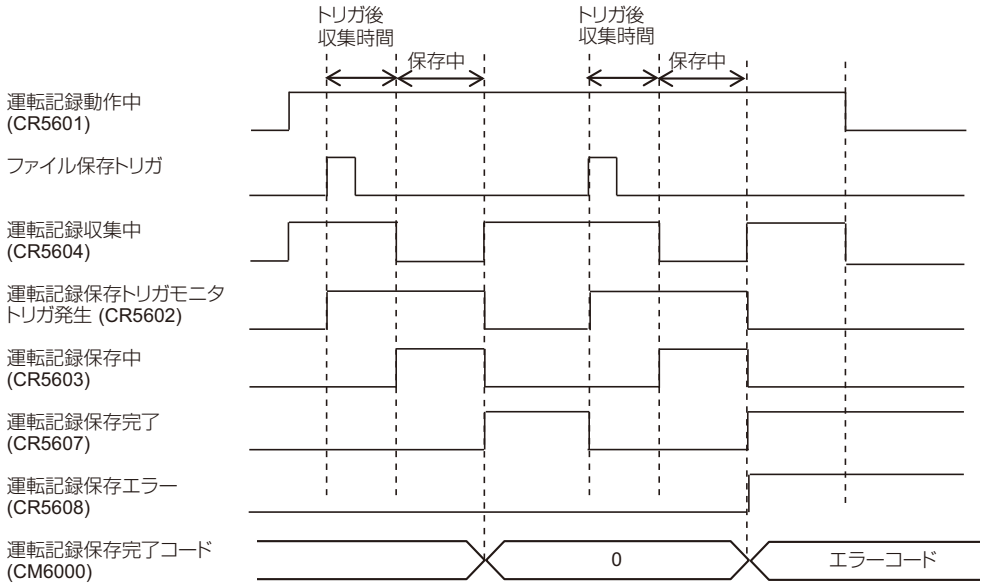
ポイント

メモリカード使用中などのID共通で使用するデバイスの詳細は 「ロギング・トレース用デバイス一覧」(5-78ページ)を参照してください。

運転記録保存時のデバイスの動作

収集期間の指定方法で「ファイル保存トリガ前後」を選択した場合と「開始リレー」を選択した場合とで、使用するデバイスや作成するラダープログラムが一部異なります。

● 「ファイル保存トリガ前後」を選択した場合



① 運転記録機能が動作中になると、運転記録動作中リレー（CR5601）と運転記録収集中（CR5604）がONし、CPUユニットがデータ収集を開始します。

② ファイル保存トリガの条件が成立すると、運転記録保存トリガモニタ（CR5602）がONし、トリガ後収集時間が経過するまでデータを収集します。

③ トリガ後収集時間が経過すると、CPUユニットが運転記録の保存を開始し、運転記録保存中リレー（CR5603）がONします。運転記録保存中リレーがONしている間は運転記録収集中リレーはOFFし、CPUユニットはデータの収集を停止します。

④ 運転記録の保存が完了すると、運転記録保存トリガモニタリレーと運転記録保存中リレーがOFFし、運転記録保存完了リレー（CR5607）がONします。運転記録保存完了コード（CM6000）に0が格納されます。

⑤ ファイルの保存に失敗した場合、運転記録保存トリガモニタ発生リレーと運転記録保存中リレーがOFFし、運転記録保存完了リレーと運転記録保存エラーリレー（CR5608）がONします。運転記録保存完了コードにはエラーコードが格納されます。

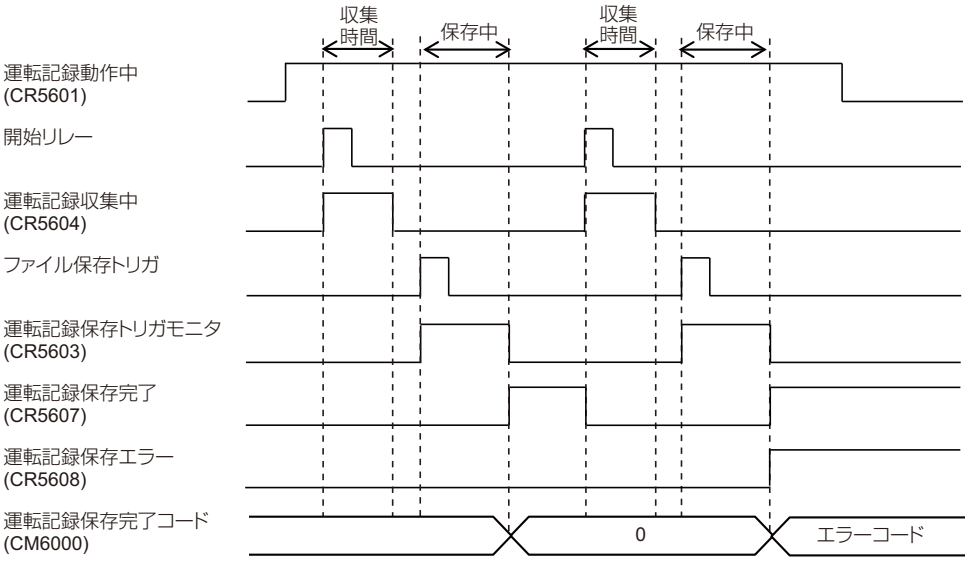
参 考

- 運転記録機能が動作中になるのはCPUユニットがPROGモードからRUNモードに切り換わったタイミングです。
- 運転記録機能に関連するCRおよびCMはCPUユニットが自動で制御するため、ラダープログラムから操作する必要はありません。

ポイント

運転記録保存完了リレーは、KV-XD02のいつもと違う分析が動作している場合、運転記録の保存完了後、いつもと違う分析が完了したタイミングでONします。

● 「開始リレー」を選択した場合



① 運転記録機能が動作中になると、運転記録動作中リレー（CR5601）がONします。

② 運転記録設定で開始リレーに指定したデバイスをONすると、運転記録収集中（CR5607）がONし、CPUユニットがデータの収集を開始します。収集時間が経過するとデータの収集を終了し、運転記録収集中はOFFします。

③ 収集時間の経過後にファイル保存トリガの条件が成立すると、CPUユニットが運転記録の保存を開始し、運転記録保存中リレー（CR5603）がONします（収集時間経過前にファイル保存トリガの条件が成立した場合は、収集時間経過後に運転記録の保存を開始します）。

④ 運転記録の保存が完了すると、運転記録保存トリガモニタリレーと運転記録保存中リレーがOFFし、運転記録保存完了リレー（CR5607）がONします。運転記録保存完了コード（CM6000）に0が格納されます。

⑤ 運転記録の保存に失敗した場合、運転記録保存トリガモニタリレーと運転記録保存中リレーがOFFし、運転記録保存完了リレーと運転記録保存エラーリレー（CR5608）がONします。
運転記録保存完了コードにはエラーコードが格納されます。

参 考

- 運転記録機能が動作中になるのはCPUユニットがPROGモードからRUNモードに切り換わったタイミングです。
- 運転記録機能に関連するCRおよびCMはCPUユニットが自動で制御するため、ラダープログラムから操作する必要はありません。

ポイント

運転記録保存完了リレーは、KV-XD02のいつもと違う分析が動作している場合、運転記録の保存完了後、いつもと違う分析が完了したタイミングでONします。

運転記録保存完了コード

運転記録保存完了コード(ID0:CM6000)に格納される完了コードの一覧です。

完了コード	タイトル	エラー要因	対処方法
0	保存成功	保存成功	保存成功
10	空き番号がありません	既にファイルがあったため書き込まなかった	保存したいフォルダ番号をずらすか、空き番号を作ってください
11	メモリカードがない、またはCPUメモリがアンマウントされている	メモリカード/CPUメモリを認識できない状態で運転記録を保存した	CPUメモリをマウントするか、メモリカードを挿入してください
12	空き容量無し	メモリカード/CPUメモリの空き容量がありません	メモリカード/CPUメモリの空き容量を増やしてください
100	カバーOPEN	メモリカードカバーが開いている	メモリカードカバーを閉じてください
101	カードLOCK	メモリカードがLOCKされている	LOCKを解除するか、書込み可能なメモリカードを使用してください
102	メモリカードアクセスエラー	メモリカードの劣化、接点部の異常、挿入状態の異常やノイズなどの外的要因の影響により、メモリカードへのアクセスに失敗した	メモリカードの挿入状態を確認してください。 メモリカードのファイルシステムをチェックしてください。 Windowsのコマンドプロンプトからchkdskコマンドを実行して、メモリカードのファイルシステムをチェックしてください。 例) メモリカードがEドライブのとき C:¥WINDOWS>chkdsk E: ノイズ源(例:モータなど大電流を流すケーブル)が近接する場合は、機器から遠ざけてください。
103	ファイル数上限超過	ユーザメモリ内のファイル数とフォルダ数の合計が上限を超過した	ユーザメモリ内のファイルまたはフォルダを削減してください。
104	プロジェクトデータ保存失敗	• プロジェクトデータを保存しようとしたが実行できなかった(その他プロジェクトデータ保存のエラー) • メモリカードが壊れている • プロジェクトデータ保存中にメモリカードが抜かれた	プロジェクトやメモリカードが適切かどうか確認してください
200	プロジェクトアクセス競合	他の経路からプログラムの転送または読出しをしていたため、プロジェクトセーブができなかった	プログラムの転送もしくは読出しが終わってから運転記録の保存を実行してください
201	カメラ入力ユニットアクセス失敗	カメラ入力ユニットが正常に動作していないか、ユニット構成情報と実際のユニット構成が異なっている	拡張バス通信エラーが発生していないか確認してください。 カメラ入力ユニットでエラーが発生していないか確認してください。 ユニットエディタのユニット構成と実機の構成が一致しているか確認してください。
202	変数不一致	運転記録設定の対象デバイスに含まれる変数が変数編集ウィンドウで定義されていません。	運転記録設定の対象デバイスに含まれる変数(構造体型、及びファンクションブロック型など含む)は変数編集ウィンドウで定義してください。

完了コード	タイトル	エラー要因	対処方法
1000	フォルダ作成失敗	フォルダ作成しようとしてできなかった (処理中にストレージの内容が変更された)	他のストレージアクセスと競合しました 再度、保存を実行してください
1001	ファイル作成失敗	ファイル作成しようとしてできなかった (処理中にストレージの内容が変更された)	他のストレージアクセスと競合しました 再度、保存を実行してください
1002	ストレージアクセス競合	ストレージLOCKできなかった (他のストレージアクセス処理と競合)	他のストレージアクセスと競合しました 再度、保存を実行してください

MEMO

TRGR

-[TRGR]- デバイス記録データ取得　指定したIDのデバイスデータ

ラダープログラム

入力方法

オペランド	説明	占有サイズ
n	取得する運転記録設定のID（0～3）、またはそれを格納しているデバイスを指定します。	16ビット

動作説明

TRGR　実行条件がONのとき、n で指定したIDのデバイスデータを取得します。
サンプリング周期に「SCAN」や「定周期」を設定すると、END処理のタイミングのデータを取得しますが、サンプリング周期に「トリガ命令」を設定し、TRGR命令を使用することでラダーの任意の時点でのデータを取得できます。

ポイント　**TRGR命令でデータを取得する場合、《KV STUDIO》の「運転記録設定」－「デバイス記録設定」でサンプリング周期を「トリガ命令（TRGR）」に設定しておく必要があります。**

演算フラグ

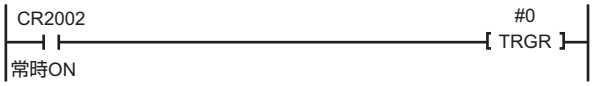
CR2009	変化なし
CR2010	変化なし
CR2011	変化なし
CR2012	• オペランドに指定した値が範囲外の場合はON。 • 間接指定、インデックス修飾の範囲が不適切な場合はON。 上記以外はOFF • n に定数を指定した場合は変化しません。

※ CR2012がONした場合、命令は実行されません。
CR2012がONした場合、CM5150～CM5176にエラーの詳細情報が格納されます。

「最新エラー詳細（CM）」（4-87ページ）

サンプルプログラム

定周期モジュール内に下記のラダープログラムを記述すれば、定周期モジュールが実行されるごとにデバイスデータ取得を実行できます。



〈ニモニックリスト〉
LD CR2002
TRGR #0

ポイント **定周期モジュールを実行する場合は、割り込み許可状態(EI命令を実行)にする必要があります。**

「定周期モジュール」4-103ページ
「4-13 割り込み」4-206ページ

運転記録機能の注意事項

運転記録機能を使用する上での注意事項を以下に示します。

● 保存速度

- 運転記録の収集時はスキャンタイムが延びます。実際のスキャンタイムはスキャンタイムモニタで確認できます。
- 運転記録保存中はデータ収集が停止するため、スキャンタイムは延びません。

参 考

スキャンタイムのばらつきが問題になる場合に、一定スキャンタイム運転やEND処理時間の設定を使用して改善することができます。

「一定スキャンタイム運転の設定」(4-91ページ)
「END処理時間の設定」(4-92ページ)

● ストレージへのアクセスの優先順位

ストレージへのアクセスの優先順位は以下のとおりです。
ストレージ命令 > ロギング・トレース命令 > 運転記録機能、FTPサーバ機能、FTPクライアント機能、アクセスウィンドウのファイル操作
このため、運転記録機能のファイル保存トリガが大量に発生している場合、他の機能がストレージを使用できないことがあります。

● 電源断、カード抜き差し

メモリカードは、本体の電源をONしたまま、メモリカードアンマウント要求リレーをONし、メモリカード認識完了リレーがOFFになることを確認してから取り出してください。運転記録の保存中はメモリカードを抜き出したり、メモリカードスロットカバーを開けたりしないでください。

メモリカードスロットカバーの扱い

- メモリカードスロットカバーが閉まっていないときは、運転記録は保存できません。
- メモリカードスロットカバーを開くと、実行中の処理は継続されますが、それ以後の処理は受け付けません。

電源断時、カードの抜き差しの注意

運転記録の保存中は保存が完了するまで保存が中断されるような操作をしないでください。保存が完了する前に保存を中断すると、運転記録は正しく保存されません。保存が中断される動作は以下のとおりです。

- CPUユニットの電源断
- メモリカードアンマウント要求リレー、CPUメモリアンマウント要求リレーをONしてストレージをアンマウントする
- メモリカードスロットカバーを開く(ファイル保存先がメモリカードの場合)

● RUN/PRGモード切換時の動作

運転記録機能はCPUユニットがRUNモードのときのみ動作します。RUN→PRG切換を行うとすべての運転記録機能が停止します。PRG→RUN切換を行うと、運転記録機能は自動で再開します。

● 運転記録保存時の処理について

運転記録の保存が実行されたとき、指定したフォルダ番号がすでに存在していると以下の動作になります。

フォルダ番号の指定方法が「自動連番」のとき

- 「空き番号がない場合」で「上書き」を選択している場合は、最も古いフォルダを削除し、同一フォルダ名で上書きします。
- 「空き番号がない場合」で「停止」を選択している場合は、保存に失敗し、運転記録の収集を再開します。

フォルダ番号の指定方法が「固定/デバイス値」のとき

すでに存在している番号のフォルダ削除し、同一フォルダ名で上書きします。

● データ保存先パスで使用できない文字

以下の文字はデータ保存先パスのコメントおよびトリガコメントに使用できません。

- 半角12文字* / < > ? ￥ ｜ % , : ; "
- 全角7文字／ ： ？ ￥ ＊ ［ ］
- NEC特殊文字（8740H～879CH）①、㌢、㍑、㍻、など
- NEC選定IBM拡張文字（ED40H～EEFCH）
- IBM選定IBM拡張文字（FA40H～FC4BH）

MEMO

5-7 USB通信

CPUユニットのUSBポートを使用して、《KV STUDIO》でラダープログラムのモニタ･デバッグができます。USBポートを使用してコマンド通信などはできません。

参 考　KV COM+ for Excel/KV COM+ Libraryを使用すれば、PLCデバイス値の読み書きが可能です。

パソコンとの接続方法

USBケーブル（OP-35331）をパソコンとCPUユニットに接続します。

ポイント

- **CPUユニットをパソコンのUSBコネクタに接続する前に、必ず《KV STUDIO》をインストールしてください。**
- **USBポートのSGは電源0Vと共通です。接続先のSGと電位差が発生すると、CPUユニットおよび接続先の機器が故障する原因となります。**

USBドライバのインストール

Windows 11/10/8/7のパソコン使用時は、自動的に認識されるためインストールする必要はありません。

CPU内蔵EtherNet/IP通信機能でできること

項目	参照マニュアル	参照ページ
ラダープログラムのリモート転送･リモートメンテナンス	『KV STUDIOユーザーズマニュアル』	「9-1　モニタ/シミュレータの各機能について」
EtherNet/IP経由のデバッグ作業		
VT5/VT3シリーズとの接続	『VT/DT PLC接続マニュアル』	「22章　イーサネット接続」
上位リンク通信	『EtherNet/IP機能ユーザーズマニュアル』	「8章　上位リンク通信機能」
MCプロトコル通信		「9章　MCプロトコル通信機能」
メール送受信		「10章　メール送受信機能」
FTPサーバ		「11章　FTPサーバ機能」
FTPクライアント		「12章　FTPクライアント機能」
KVソケット通信		「14章　KVソケット通信機能」
BOOTPクライアント		「付-2　ユニット設定レス通信(BOOTP)」
サイクリック通信		「4章　EtherNetI/P通信機能」
メッセージ通信（クライアント）		
メッセージ通信（サーバ）		
センサアプリケーション機能		「7章　センサアプリケーション機能」

使用できる機能の詳細は『EtherNet/IP機能ユーザーズマニュアル』を参照してください。

OPC UAサーバ機能でできること

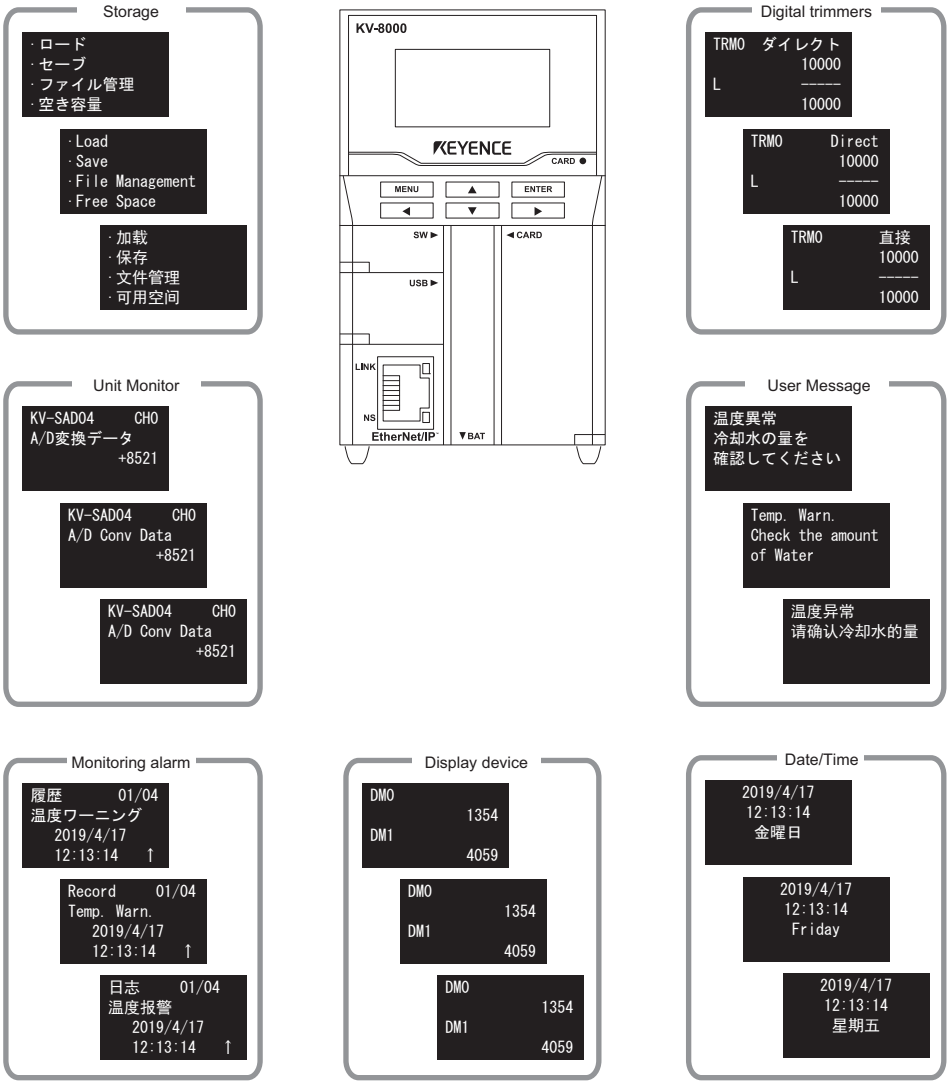
項目	参照マニュアル	参照ページ
OPC UAサーバ機器	『OPC UAサーバ機能ユーザーズマニュアル』	「1章 OPC UA機能の概要」

5-9 アクセスウィンドウ

CPUユニットや拡張ユニットの各データのモニタや変更が簡単にできるアクセスウィンドウの機能について説明します。

アクセスウィンドウでできること

KV-8000シリーズは、小型の2色バックライト付LCD表示器を搭載しています。
RUN中は白、PRGモード時(停止中)は赤になります。



■ 機能一覧

CPUユニットに搭載されているアクセスウィンドウでデバイスデータを変更･モニタできます。また、万一のエラー発生時にエラーコードが確認できます。
アクセスウィンドウには、以下の機能があります。

ユニットのモニタ

CPUに接続されたユニットのダイレクトアクセススイッチを押すと、アクセスウィンドウ上でそのユニットの状態をモニタできます。

「ユニットのモニタ」(5-117ページ)

デバイスモード

内部デバイスの表示、設定内容の変更が可能です。

「デバイスモード」(5-120ページ)

ストレージ

メモリカードやCPUメモリ内のプロジェクトセーブやロード、ファイル管理、空き容量の確認などの操作を行えます。

「ストレージ」(5-141ページ)

エラークリア

CPUユニットの重度エラーをクリアします。

「エラークリア」(5-129ページ)

デジタルトリマ

デジタル方式でトリマの設定を行えます。最大分解能は4294967296です。

「デジタルトリマ」(5-130ページ)

ユニットテスト

拡張ユニットの入出力などの状態を確認できます。

「ユニットテスト」(5-132ページ)

時刻表示／設定

CPUの内部時計の表示、時刻設定を行います。

「時計」(5-134ページ)

CPUモニタ

CPUの内蔵機能の状態をモニタできます。また、本体に入っているプログラムのプロジェクト名や、本体のシリアル番号、バージョンを確認できます。

「CPUモニタ」(5-136ページ)

アラームモニタ

任意に設定したアラームのモニタや履歴を確認できます。

「アラーム」(5-138ページ)

アクセスウィンドウ設定

アクセスウィンドウのコントラストなど、アクセスウィンドウの表示に関する設定を行えます。

「アクセスウィンドウ設定」(5-154ページ)

Language
表示する言語を、日本語、英語、中国語(簡体字)から選択できます。

RAMクリア
デバイスの値をクリアできます。
「RAMクリア」(5-156ページ)

オールクリア
設定した内容をすべてクリアできます。
「オールクリア」(5-157ページ)

初期化
CPUメモリの全エリアに無効なデータを書き込むことで設定した内容を復元できないようにクリアできます。
「初期化」(5-158ページ)

CPUメモリクリア
CPUメモリ内のデータをクリアできます。

■ メニュー構成

各モード間の移行は、メニュー画面で切り替えます。
「MENU」キーを押すと、メニューが表示されます。(他ユニットが選択されている場合は、そのユニットのダイレクトアクセスキーを押すか、「MENU」キーを2回押してください。)
画面に収まらないメニューは「▼」キーを押すと表示されます。「▲」「▼」キーで画面が前後にページ送りされます。

画面表示例では日本語表記を左側に、英語表記を中央、中国語表記を右側に表示しています。日本語／英語／中国語表記の切換は「11.LANGUAGE」で設定します。
「アクセスウィンドウ設定」(5-154ページ)

ポイント
一部のメニューでは中国語表記に対応していません。対応していない文言を中国語で表示すると英語表記になります。

CPUのメニュー内容は次のとおりです。

日本語	英　語	中国語
1.デバイスモード	1.Device Mode	1. 软元件模式
2.エラークリア	2.Error Clear	2. 错误清除
3.ユニットテスト	3.Unit Test	3. 单元测试
4.ストレージ	4.Storage	4. 存储
5.CPUモニタ	5.CPU Monitor	5. CPU监控
6.時計	6.Data/Time	6. 时钟
7.トリマ	7.Trimmer	7. 微调器
8.アラーム	8.Alarm	8. 报警
9.AW設定	9.AW Setting	9. 访问窗设定
10.クリア	10.Clear	10. 清除
11.LANGUAGE	11.Language	11.LANGUAGE

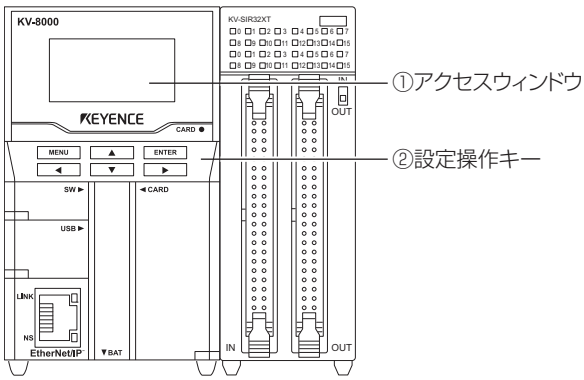
参 考
重度エラーが発生して履歴がクリアされていない場合、アクセスウィンドウの右上に「E」が表示されます。

エラークリアの方法は、「エラークリア」(5-129ページ)を参照してください。

アクセスウィンドウのメニュー構成は以下のようになっています。

- 1.デバイスモード(1.Device Mode/1.软元件模式)
- 2.エラークリア(2.Error Clear/2.错误清除)
- 3.ユニットテスト(3.Unit Test/3.单元测试)
- 4.ストレージ(4.Storage/4.存储)
  - ロード(Load/加载)
  - セーブ(Save/保存)
  - ファイル管理(File Management/文件管理)
  - 空き容量(Free Space/可用空间)
- 5.CPUモニタ(5.CPU Monitor/5.CPU监控)
  - プロジェクト名(Project Name/项目名)
  - バージョン(Version/版本)
  - イーサネット(Ethernet/以太网)
- 6.時計(6.Data/Time/6.时钟)
- 7.トリマ(7.Trimmer/7.微调器)
- 8.アラーム(8.Alarm/8.报警)
  - モニタ(Monior/报警监控器)
  - 履歴モニタ(Monitor Record/日志监控器)
  - 履歴クリア(Clear Record/日志清除)
- 9.AW設定(9.AW Setting/9.访问窗设定)
  - コントラスト(Contrast/对比)
  - コメント番号(Comment Number/注释编号)
- 10.クリア(10.Clear/10.清除)
  - RAMクリア(RAM Clear/RAM 清除)
  - ユーザメモリクリア(User Mem Clear/用户存储器清除)
  - オールクリア(All Clear/全部清除)
- 11.LANGUAGE(11.Language/11.LANGUAGE)
  - 初期化(Initialization/初始化)

■ 各部の名称と機能



①アクセスウィンドウ…各デバイスの現在値や設定値、およびエラーメッセージなどが表示されます。各ユニットのダイレクトアクセススイッチが押されると、各ユニットの情報が表示されます。

②設定操作キー…各モード間の移行、および各モードでの設定変更に使用します。それぞれのキーの機能は、各モードごとに異なります。詳細は、各モードの操作方法を参照してください。

ユニットのモニタ

CPUに接続されたI/Oユニットなどの拡張ユニットや特殊ユニットのダイレクトアクセススイッチを押すと、アクセスウィンドウ上で、そのユニットのモニタおよび設定ができます。
拡張ユニット、特殊ユニットの画面から、もとの画面に戻るには、現在アクセスウィンドウに表示されているユニットのダイレクトアクセススイッチをもう一度押すか、「MENU」キーを2回押してください。

参 考 ユニットエラーが発生した場合にはダイレクトアクセススイッチが赤色になり、アクセスウィンドウにエラー内容を表示します。

ユニットのモニタ内容や設定内容は、各ユニットのマニュアルを参照してください。

■ 設定操作キーの機能

ユニットのモニタおよび設定時のキー機能は次のとおりです。

設定操作キー	通　常	数値変更時
MENU	メニュー表示	通常表示に戻ります。
ENTER	1秒以上押しつづけると、設定変更を実行します。	1秒以上押しつづけると、設定変更を実行します。
▲	ページを切り換えます。設定変更時は選択肢を変更します。	設定値から変化量を加算します。
▼	ページを切り換えます。設定変更時は選択肢を変更します。	設定値から変化量を減算します。
◀	メニュー表示	変化量の桁を変更します。
▶	使用しません	

参 考 **設定操作キーのデバイス番号**
ユニットの設定操作キーを押すと、下記のコントロールリレー（CR）がONします。

設定操作キー	デバイス番号	コメント名称
MENU	CR4404	AW MENUキー押下中
ENTER	CR4405	AW ENTERキー押下中
▲	CR4400	AW 上キー押下中
▼	CR4401	AW 下キー押下中
◀	CR4402	AW 左キー押下中
▶	CR4403	AW 右キー押下中

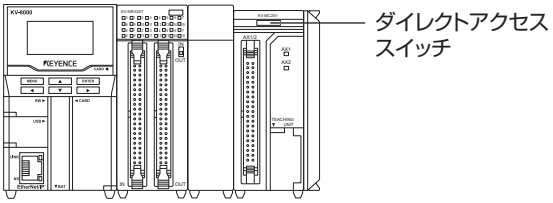
■ 操作方法(入出力ユニットの場合)

KV-SIR32XTを例に操作方法を説明します。

● ユニットモニタへの移行

1 ユニット部のダイレクトアクセススイッチを押します。

選択されたユニットのダイレクトアクセススイッチが点滅します。



2 リレーのON/OFF状態が表示されます。

アクセスウィンドウは選択されているユニットによって次のようになります。

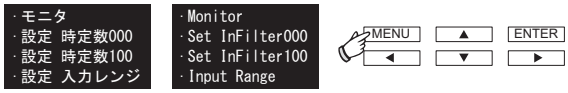
「▲」「▼」キーでページを切り換えます。

ポイント **ユニットモニタは中国語表記に対応していません。**

● メニュー表示

1 ユニットのダイレクトアクセススイッチが点滅(選択)している状態で、「MENU」キーを押す。

ユニットのメニューが表示されます。



● 設定変更

入力時定数を変更する例で説明します。

1 設定時定数000を選択し、続いて「ENTER」キーを押す。

現在の時定数の設定状態が確認できます。

2 この状態で「ENTER」キーを1秒以上押します。

設定値が点滅し変更モードに入ります。(RUN中は不可、PRGモード時のみ変更可)

3 上下キーで設定値を変更し、「ENTER」キーを1秒以上押します。

点滅が止まり設定されます。(MENUキーを押すとキャンセルになります。)

デバイスモード

PLCデバイスの値がモニタ、および変更できます。

■ 表示・設定変更できるデバイス

表示設定変更できるデバイスは次のとおりです。

表　示	名　称	表　示	名　称
R	リレー	ZF	ファイルレジスタ(連番方式)
MR	内部補助リレー	W	リンクレジスタ
LR	ラッチリレー	CM	コントロールメモリ
B	リンクリレー	TM	テンポラリメモリ
CR	コントロールリレー	T	タイマ
DM	データメモリ	C	カウンタ
EM	拡張データメモリ	TRM	デジタルトリマ
FM	ファイルレジスタ(バンク切換方式)	Z	インデックスレジスタ

ポイント　**タイマ・カウンタの接点の状態はモニタできません。**

● XYM表記について

リレー(R)、内部補助リレー(MR)、ラッチリレー(LR)、データメモリ(DM)、拡張データメモリ(EM)、ファイルレジスタ(FM)については、XYM表記で表示可能です。
プロジェクトの設定(KV STUDIOのツール>オプション>表示モードの設定>XYM表記)により、デフォルト表記／XYM表記が自動的に切り換わります。

デバイス名	デフォルト表記	XYM表記		XYM表記時のデバイス番号
リレー	R	入力	X	0000～1999F
		出力	Y	0000～1999F
		内部補助	R	0000～199915
内部補助リレー	MR	M		00000～63999
ラッチリレー	LR	L		00000～15999
データメモリ	DM	D		00000～65534
拡張データメモリ	EM	E		00000～65534
ファイルレジスタ	FM	F		00000～32767

■ 設定操作キーの機能

デバイスモードでの各キーの機能は次のとおりです。

設定操作キー	通　常	数値変更時
MENU	メニュー表示	通常表示に戻ります。
ENTER	数値変更状態への移行 1秒以上押し続けます。	変更値の登録決定 1秒以上押し続けると登録決定 1秒未満の場合は無視されます
▲	デバイス番号の選択	変更値の数値を増加させます。
▼	デバイス番号の選択	変更値の数値を減少させます。
◀	デバイス種別設定を表示	変更値の変更する桁を左に移動します。
▶	デバイス表記設定画面を表示	変更値の変更する桁を右に移動します。

■ 操作方法

● デバイスモードへの移行

1 「MENU」キーを押します。

メニューが表示されます。(他ユニットが選択されている場合は「MENU」キーを2回押してください。)

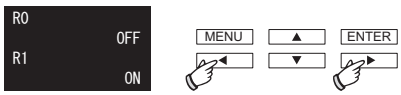
2 [1.デバイスモード]を選択して、「ENTER」キーを押します。

デバイスモードに切り換わり、モニタ表示の画面が表示されます。

「▲」「▼」キーでモニタするデバイス番号を変更できます。

● モニタするデバイス種別の変更

1 デバイスモードの状態で「◀ / ▶」キーを押します。



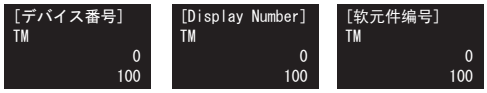
2 デバイスの種類が一覧で表示されます。

「▲」「▼」キーで表示したいデバイスを選択してください。
ここでは例として[TM]を選択します。

デバイスの種類はR→MR→LR→B→CR→DM→EM→FM→ZF→W→CM→TM→T→C→TRM→Zの順になっています。Zを選択している状態で「▼」キーを押すと、再びRが選択されます。
同様に、Rを選択している状態で「▲」キーを押すと、Zが選択されます。

3 **[・TM]が点滅表示します。**

ここで「ENTER」キーを押すと



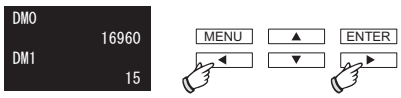
表示したいデバイス記号を決める画面になります。
現在のデバイス番号が上から3行目に点滅表示されます。

4行目に表示されている数値は、デバイス番号の変化量です。
デバイス番号は「▲」キー、「▼」キーで変化します。
図の場合、「▲」キーを押すと、100ずつデバイス番号が大きくなります。
デバイス番号の変化量は、「◀ / ▶」キーで10倍または1/10倍になります。
「▲」キー、「▼」キー、「◀ / ▶」キーで、表示したいデバイス番号を決めてください。

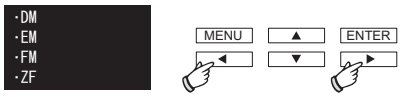
「ENTER」キーを押すと、指定した番号のTMの表示になります。

■ デバイス値の表示形式の変更

1 **デバイスモードの状態で「◀ / ▶」キーを押します。**



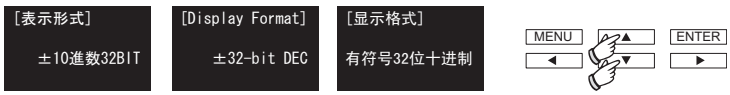
2 **デバイスの種類が一覧で表示されている状態で「◀ / ▶」キーを押します。**



3 **表示形式の選択画面になります。**

4 **表示形式を選択します。**

「▲」「▼」キーで表示形式を選択してください。



5 **「ENTER」キーを押すと、選択した表示形式で表示されます。**

ここで選択した表示形式は、すべてのデバイスに適用されます。

```
DM0
          +1000000
DM2
            +50000
```

選択できる表示形式は次のとおりです。

日本語	英語	中国語	表示内容
デフォルト	Default	默认格式	デバイス種別による(別表)
2進数1BIT	1-bit BIN	1位二进制	ON/OFF表示
10進数16BIT	16-bitDEC	无符号16位十进制	データ幅16ビット 符号無し10進数表示
±10進数16BIT	±16-bit DEC	有符号16位十进制	データ幅16ビット 符号付き10進数表示
16進数16BIT	16-bit HEX	16位十六进制	データ幅16 ビット16進数表示
2進数16BIT	16-bit BIN	16位二进制	データ幅16 ビット 2進数表示
10進数32BIT	32-bit DEC	无符号32位十进制	データ幅32ビット 符号無し10進数表示
±10進数32BIT	±32-bit DEC	有符号32位十进制	データ幅32ビット 符号付き10進数表示
16進数32BIT	32-bit HEX	32位十六进制	データ幅32 ビット16進数表示
ASCII 16BIT	ASCII 16BIT	16位ASCII码	アスキー文字列表記
FLOAT	FLOAT	単精度浮点型	データ幅32ビット 単精度浮動小数点表示
DOUBLE FLOAT	DOUBLE FLOAT	双精度浮点型	データ幅64ビット 倍精度浮動小数点表示

各デバイスの表示可能な形式　　**◎：表示可能(デフォルト)　○：表示可能　×：表示不可能**

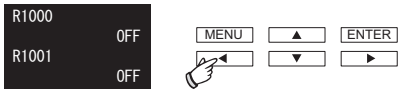
デバイス種別	ON/OFF	16Bit-BASE10 符号無し	16Bit-BASE10 符号付き	16Bit-BASE16	16Bit-BASE2	32Bit-BASE10 符号無し	32Bit-BASE10 符号付き	32Bit-BASE16	ASCII	FLOAT	DOUBLE FLOAT
R/MR/LR/B/CR	◎	○	○	○	○※1	×	×	×	×	×	×
DM/EM/FM/ZF/W/CM	×	◎	○	○	○※1	○	○	○	○※1	○※1	○※1
T/C/TRM	×	×	×	×	×	◎	○	○	×	×	×
Z	×	×	×	×	×	×	◎	○	×	×	×

デバイスによって選択できる表示形式が異なります。適応できない表示形式を選択した場合は、Default(デフォルト)の表示形式になります。

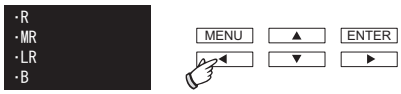
※1　表示は可能ですが、この形式でのアクセスウィンドウからのデバイス値書き換えには対応していません。

● デバイス名/コメント表示の切り換え

1 **デバイスモードの状態で「◀」キーを押します。**



2 **デバイスの種類が一覧で表示されている状態で「◀」キーを押します。**



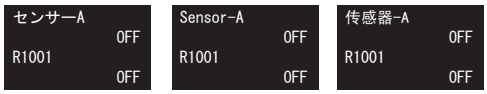
3 **表示形式の選択画面で「◀」キーを押します。**

4 **デバイス名/コメント表示の選択画面になります。**

「▲」「▼」キーでデバイス名を表示するか、コメントを表示するかを選択してください。
「▼」キーを押すと、[コメント]と表示されます。

5 **「ENTER」キーを押すと、コメント表示が選択され、デバイスモードのモニタに戻ります。**

R1000にコメント"センサーA"が登録されていて、R1001にはコメントが登録されていない場合、下図の表示になります。



アクセスウィンドウに表示できるコメントについて

- 半角16文字を超える場合には、自動的にスクロールされます。
- 表示できる文字はUnicodeです。
- Zにはコメントが設定できません。

● デバイス値の変更

- 表示形式がON/OFFの場合

1 変更したいデバイスが表示されている状態で「ENTER」キーを1秒以上押し続けます。

1画面に2つのデバイスが表示されているとき、値が変更できるデバイスは、上段に表示されているデバイスです。（このとき、下段のデバイスの表示はなくなります。）

2 デバイスの値が点滅表示になります。

この状態で「▼」「▲」キーを押すとON→OFF→ON→…と表示が切り換わります。
（この時点では、デバイス値の書き込みは実行していません。）

3 書き込みたいデバイス値に設定して「ENTER」キーを1秒以上押し続けます。

点滅がとまり書き込みが実行されます。「MENU」キーを押すと、書き込みを実行せずにモニタ表示に戻ります。

- 表示形式がON/OFF以外の場合（T、C、TRMを除く）
  ここではDM1000番の値100を201に変更する方法を例に説明します。

1 変更したいデバイスが表示されている状態で「ENTER」キーを1秒以上押しつづけます。

2 変更したいデバイスの値が点滅します。

一番下の段に[変化量]が表示されます。

3 「▲」キーを押します。

デバイスの値に「変化量」分だけ足されて、デバイス値の表示が[101]になります。
（このとき、デバイスへの書き込みは行われていません。）

4 「◀ / ▶」キーを1回押します。

「変化量」の桁が変わります。1回押すごとに1桁ずつ変化します。
100の桁を変更したい場合「◀」キーを2回押します。

5 「変化量」の値が[100]になります。

6 「▲」キーを1回押します。

「変化量」の値[100]が足されて、デバイスの値の表示が[201]になります。
（この時点では、デバイスへの書き込みはまだ行われていません。）

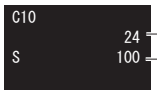
7 デバイス値を書き込みたい値にしてから「ENTER」キーを1秒以上押します。

値の書き込みが実行され、モニタの表示に戻ります。

「MENU」キーを押した場合は、書き込みを実行せずに、モニタの表示に戻ります。

- 表示形式がON/OFF以外の場合（T、C、TRM）
  T、C、TRMの場合、設定値と現在値の両方を変更できます。

タイマ現在値です。
タイマ設定値です。



カウンタ現在値です。
カウンタ設定値です。

トリマ値です。
トリマ上限値です。

デバイスが、ラダーで使われていない場合（未登録の場合）は左図の表示になります。

（上限値設定をしていない時は表示されません。
「デジタルトリマ」5-130ページ）

<設定値変更方法>

デバイスの値を変更するとき「ENTER」キーを1秒以上押しつづけると、次の項目を変更できます。

T　　：タイマ設定値
C　　：カウンタ設定値
TRM　：トリマ

<現在値変更方法>

「◀ / ▶」キーを押しながら、「ENTER」キーを1秒以上押しつづけると、次の項目を変更できます。

T　　：タイマ現在値
C　　：カウンタ現在値
TRM　：トリマ上限値（トリマ上限値はCMに割り当てられています。）

「デジタルトリマ」5-130ページ

エラー内容の確認

CPUユニットのエラーには軽度エラーと重度エラーの2種類があります。

	軽度エラー	重度エラー
アクセスウィンドウの色	白	赤
アクセスウィンドウの表示	エラー内容を表示しながら点滅 CPUエラー　W128 ラダー演算エラー	エラー内容を表示しながら点滅 CPUエラー　E52 ユニット未接続
CPUの状態	RUNモードのまま （ラダープログラムは動作します）	PROGRAMモードに切換 （ラダープログラムは停止します）
エラークリア方法	• アクセスウィンドウのいずれかのキーを押す • ラダープログラム上で CR3909 を SETする	CPU がPROGRAM モードになるため、ラダープログラムからはエラークリアできません。エラークリア方法は「エラークリア」(5-129ページ)を参照してください。

参 考　重度エラー発生中はアクセスウィンドウのいずれかのキーを押すとエラーの表示は消えますが、エラーはクリアされません。アクセスウィンドウを操作しない状態が30秒以上続くか、RUN⇔PRG切換スイッチをRUNに切り換えるとエラー内容を再表示します。

■ エラー発生時の対処

1 エラーコードを確認します。

アクセスウィンドウに表示されるエラーコードを確認します。軽度エラーと重度エラーでエラーコードの表示が下記のように異なります。

軽度エラー･･･「W○○○」
重度エラー･･･「E○○○」

CPUエラー　E52
ユニット未接続

右図の例では、エラーコード52の重度エラーが発生していることが分かります。

2 エラー内容と対処法を確認します。

「PLC本体エラー一覧」(付-2ページ)でエラー内容と対処法を確認し、エラー原因を取り除きます。

52	設定エラー	ユニット未接続 Unit Unconnected 単元未連接	ユニット設定情報では拡張ユニットありになっているのに、実際には接続されていません。	電源をOFFにしてから、ユニットエディタで設定した拡張ユニットを接続してください。

拡張ユニットで発生したエラーの場合は、各ユニットのユーザーズマニュアルを確認してください。

参 考　重度エラー発生中はCR3905がONし、CM5105にエラーコードが格納されます。軽度エラー発生中はCR3906がONし、CM5155にエラーコードが格納されます。これらのCRやCMを使用することで、タッチパネルからエラー内容を確認できます。「最新エラー詳細(CM)」(4-87ページ)

3 エラークリアを実行します。

軽度エラーが発生している場合は、アクセスウィンドウのいずれかのキーを押すか、ラダープログラム上でCR3909をSETしてください。
重度エラーが発生している場合は、アクセスウィンドウを操作してエラークリアする必要があります。
エラークリアの方法については、「エラークリア」(5-129ページ)を参照してください。

エラークリア

CPUユニットの重度エラーをクリアします。

■ 設定操作キーの機能

エラークリア時のキー機能は次のとおりです。

設定操作キー	機　能
MENU	メニュー表示
ENTER	1秒以上押し続けると表示中のエラーをクリアします。 1秒未満の場合は無効です。また、エラーが発生していない場合も無効です。
▲	無効
▼	無効
◀	MENUキーと相当
▶	ENTERキーと相当

■ 操作方法

1 「MENU」キーを押します。

メニューが表示されます。(他ユニットが選択されている場合は「MENU」キーを押してください。)

2 「2.エラークリア」を選択して、「ENTER」キーを押します。

以下のような画面が表示されます。

- エラーがある場合の画面例

- エラーがない場合の画面

3 エラーがある場合は、「ENTER」キーを1秒以上押し続けます。

表示中の重度エラーがクリアされます。
複数の重度エラーが発生している場合は別の重度エラーが表示されますので、クリアするには再度「ENTER」キーを1秒以上押します。 全ての重度エラーがクリアされると、「エラーなし」と表示します。

参 考

拡張ユニットでエラーが発生した場合はアクセスウィンドウに以下のような画面が表示されます。
下記例ではユニット番号4のKV-XH16MLでエラー番号1280(軸1ML-Ⅲスレーブ局通信エラー)が発生していることが分かります。

拡張ユニットでエラーが発生した場合は、エラー要因を取り除いたうえで拡張ユニット側でエラークリアをする必要があります。拡張ユニットのエラークリア方法は、各ユニットのユーザーズマニュアルを参照してください。

デジタルトリマ

デジタルトリマモードでは、トリマの値をデバイスモードより簡単に変更できます。
ラダープログラムでトリマの値を読み込むにはTMIN命令を使用します。

『KV-8000/7000/5000/3000/1000･KV Nanoシリーズ命令語リファレンスマニュアル「TMIN命令」』

■ 設定操作キーの機能

トリマ値変更時の各キーの機能は次のとおりです。

設定操作キー	数値変更時
MENU	メニュー表示
ENTER	表示するトリマの番号を変更します。 1回押すごとに、デバイス番号が1つ増えます。(TRM0～7)
▲	トリマ値から変化量を加算します。
▼	トリマ値から変化量を減算します。
◀	変化量の桁を変更します。1回押すごとに1桁ずつ増えます。(桁数の上限を超えると最下位に移動します。)
▶	変化量の桁を変更します。1回押すごとに1桁ずつ減ります。(桁数の下限を超えると最上位に移動します。)

デバイスモードでの値変更とは異なり、「ENTER」キーを押さずに値の変更がダイレクトにそのまま反映されます。

■ 操作方法

1 「MENU」キーを押します。

メニューが表示されます。(他ユニットが選択されている場合は「MENU」キーを押してください。)

2 [2.トリマ]を選択して、「ENTER」キーを押します。

3行目の値はトリマ上限値を表し、上限値設定を使用していない場合は表示されません。

3 「ENTER」キーで表示するトリマ番号を変更できます。

変更したいトリマ番号に切り換えます。

4 「◀ / ▶」キーで変化量を増減させます。

5 **「▲」キーで変化量分だけトリマ値が増加、「▼」キーで変化量分だけ減少します。**

デバイスモードとは異なり、トリマモードでは「ENTER」キーを押さずに値の変化がそのまま反映されます。

参 考

- トリマの現在値はTMIN命令で読み込みます。
- 上限値の設定を有効にするには、KV STUDIO上で設定が必要です。(【CPUシステム設定】▶【デジタルトリマ上限値の設定】)
- 上限値は、KV STUDIOやAWのデバイスモード(CM1700～CM1715)で設定できます。
- 上限値の設定を有効にしている状態で上限値の設定値を「0」にすると、上限値は4294967295になります。
- 上限値は、以下のCMに格納されています。

トリマ番号	デバイス
TRM0 上限値	CM1700、CM1701
TRM1 上限値	CM1702、CM1703
TRM2 上限値	CM1704、CM1705
TRM3 上限値	CM1706、CM1707
TRM4 上限値	CM1708、CM1709
TRM5 上限値	CM1710、CM1711
TRM6 上限値	CM1712、CM1713
TRM7 上限値	CM1714、CM1715

ユニットテスト

拡張ユニットの入出力状態等を確認できます。

ポイント **PRGモード時のみ実行できます。**

参考 KV STUDIOで設定するユニット構成の設定を含んだプロジェクトがCPUユニットに転送されていない場合でも確認できます。

■ 設定操作キーの機能

設定操作キー	数値変更時
MENU	ユニット選択画面に戻ります。
ENTER	入力ユニット：無効 出力ユニット：カーソル位置を1ビット分前に移動します。
▲	無効
▼	無効
◀	MENUキーと相当
▶	ENTERキーと相当

■ 操作方法

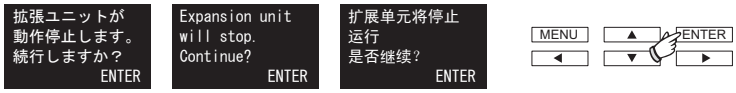
● ユニットテストへの移行

1 「MENU」キーを押します。

メニューが表示されます。（他ユニットが選択されている場合は「MENU」キーを押してください。）

2 「3.ユニットテスト」を選択して、「ENTER」キーを押します。

以下の画面が表示されます。

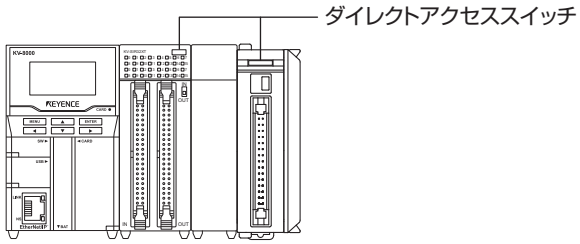


3 「ENTER」キーを1秒以上押し続けます。

以下の画面が表示されます。

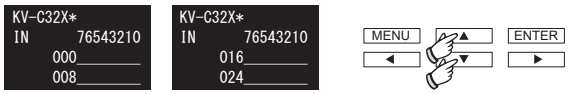
4 確認する拡張入出力ユニットのダイレクトアクセススイッチを選択します。

選択されたユニットのダイレクトアクセススイッチが点滅します。



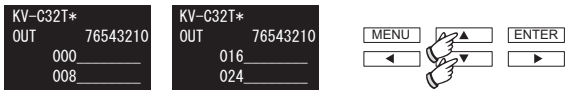
ユニットテストに対応しているユニットは、KV-8000/7000シリーズ用の拡張ユニットと、KV-5000/3000シリーズ用の拡張I/Oユニットです。
KV-8000/7000シリーズ用拡張ユニット使用時のユニットテストの内容については、各ユニットのユーザーズマニュアルを参照してください。
KV-5000/3000シリーズ用の拡張I/Oユニット使用時は以下のとおりです。

- 入力ユニットの場合
  例）KV-C32XC



「▲」「▼」キーで確認するリレー位置を16ビットずつ移動できます。

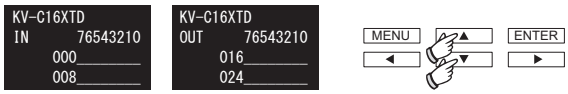
- 出力ユニットの場合
  例）KV-C32TD



「▲」「▼」キーで確認するリレー位置を8ビットずつ移動できます。
「◀」「▶」キーでカーソル位置を変更できます。
「ENTER」キーでカーソル位置の出力状態（ON/OFF）を反転できます。

ポイント **接続されている制御機器の状態を十分に確認してから操作してください。**

- 入出力ユニットの場合
  例）KV-C16XTD



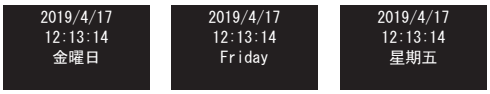
「▲」「▼」キーで確認するリレー位置を16ビット（出力リレーは8ビット）ずつ移動できます。
「◀」「▶」キーでカーソル位置を変更できます。
「ENTER」キーでカーソル位置の出力状態（ON/OFF）を反転できます。

ポイント **接続されている制御機器の状態を十分に確認してから操作してください。**

時計

時刻の表示や設定ができます。
時刻は24H表示です。
曜日は次のように表示されます。

日曜日：Sunday　：星期日
月曜日：Monday　：星期一
火曜日：Tuesday　：星期二
水曜日：Wednesday　：星期三
木曜日：Thursday　：星期四
金曜日：Friday　：星期五
土曜日：Saturday　：星期六



■ 設定操作キーの機能

時刻設定時のキー機能は次のとおりです。

設定操作キー	時計モニタ	日時設定	アジャスト画面
MENU	メニュー表示	時計モニタ表示	時計モニタ表示
ENTER	時刻表示中に、1秒以上押すと時刻設定をできます。	1秒以上長押しで設定を反映し、モニタ画面表示	アジャストを実行します。
▲	無効	設定変更	無効
▼	無効	設定変更	無効
◀	MENUキーと相当	設定項目変更	MENUキーと相当
▶	アジャスト画面へ移行します。	設定項目変更	無効

■ 操作方法

● 時計への移行

5 「MENU」キーを押します。

メニューが表示されます。（他ユニットが選択されている場合は「MENU」キーを2回押してください。）

6 [5.時計]を選択して、「ENTER」キーを押します。

以下のような画面が表示されます。

● 時刻設定

1 時刻表示の状態で「ENTER」キーを1秒以上押します。

「年」の項目が点滅します。

2 「◀」「▶」キーを押して設定する項目を選択します。

「◀」「▶」キーを押すごとに、設定する項目が「年」、「月」、「日」、「時間」、「分」、「秒」、「曜日」と切り換わります。

3 「▲」「▼」キーで項目を設定してください。

4 手順2、3を繰り返して、設定が終了したら、「ENTER」キーを1秒以上押しつづけてください。

画面最下段右に[実行]が表示され、時刻設定要求が受け入れられた状態になります。
（この時点では、まだ実行されていません。）
「ENTER」キーを離すと、時刻設定を実行して時刻表示に戻ります。

「MENU」キーを押すと、時刻設定を実行せずに時刻表示に戻ります。

ポイント
年、月、日、時、分、秒、曜日をすべて正確に設定してください。
年、月、日、時、分、秒に対する曜日のチェックは行われません。

■ 0秒アジャスト

0秒アジャストは、秒値を0にする機能です。

現在の秒の値	動　作
0～29	秒の値のみを0にします。
30～59	分の値を1つ増やして、秒の値を0にします。

1 時刻表示の状態で「▶」キーを押します。

3,4行目に[ENTERを押すとアジャスト実行]と表示され、時刻アジャストが可能になります。
（[アジャスト]をしない場合は、「◀」キーを押すと、[アジャスト]表示が消え、時刻表示に戻ります。）

2 「ENTER」キーを押します。

CPUモニタ

CPUユニットの内蔵機能をモニタできます。また、CPUユニットに格納されているプロジェクトの名称を表示できます。

■ モニタできる項目

アクセスウィンドウでモニタできる項目は次のとおりです。

日本語	英語	中国語	機能
プロジェクト名	Project Name	项目名	CPUユニットに格納されているプロジェクトの名称を表示します。(名称はKV STUDIOで設定)
バージョン	Version	版本	ユニットのシステムプログラムのバージョンと、シリアル番号を表示します。 ※対応しているのはKV-8000/7000シリーズ用拡張ユニットのみです。
イーサネット	Ethernet	以太网	CPU内蔵EtherNet/IP/イーサネット機能や、OPC UAサーバ機能のモニタができます。モニタできる内容はEtherNet/IP機能ユーザーズマニュアルを参照してください。

■ 操作方法

● CPUモニタへの移行

1 「MENU」キーを押します。

メニューが表示されます。(他ユニットが選択されている場合は「MENU」キーを2回押してください。)

2 「5.CPUモニタ」を選択して、「ENTER」キーを押します。

CPUモニタのメニューが表示されます。

● プロジェクト名

1 CPUモニタのメニューで「プロジェクト名」を選択して、「ENTER」キーを押します。

プロジェクト名が表示されます。

アクセスウィンドウに表示できるプロジェクト名について

全角8文字(半角16文字)まで1行に表示でき、1行に入りきらない場合はスクロールして表示できます。

● バージョン

1 CPUモニタのメニューで「バージョン」を選択して、「ENTER」キーを押します。

バージョンが表示されます。

参 考

「▲」「▼」キーを使用してCPUユニット以外の拡張ユニットのバージョン、シリアル番号も確認できます。
(KV-8000/7000シリーズ用以外の拡張ユニットの場合、バージョン画面に表示されません。)

アラーム

アラームリレー(CR3300～CR3415)にデバイスコメントとしてメッセージを登録しておくと、アラームリレーがONした場合にメッセージを自動的にアクセスウィンドウ上に表示します。また、発生したアラームの履歴を確認したりクリアすることもできます。

■ 使用方法

アラームで使用するデバイスは次のとおりです。

コントロールリレー	R/W	内容
CR3300～CR3415	R/W	アラームリレー(32個)です。あらかじめデバイスコメントを登録しておきます。
CR3500	R	アラームリレーが1つでもONしているとCR3500がONします。ONしているアラームがすべてOFFすると、自動的にCR3500もOFFします。
CR3501	R/W	ON→OFF履歴を保存します。
CR3502	R/W	OFF→ON時にアラームリレーの履歴をクリアします。

ポイント **アラームリレーは必ずSET、RES命令で操作してください。動作中のアラームリレーのリセットかつアラーム履歴の一括クリアはARES命令でも可能です。**

● アラームメッセージの登録

アラームリレー(CR3300～CR3415)にデバイスコメントを登録します。

ポイント **アクセスウィンドウに表示できるメッセージは全角8文字(半角16文字)までです。デバイスコメントを設定していない場合はアラームリレーの番号を表示します。**

● アラーム履歴について

アラーム履歴は、CPUユニットに最大16件まで保存されます。
16件を超える場合は、古い履歴から順に削除して新しい履歴を保存します。
1件につき、「アラームリレー番号」「時刻(年月日時分秒)」「ON/OFFの状態」を保存します。

- ON→OFF履歴リレー(CR3501)がOFFの場合
  アラームリレーがOFF→ONの場合のみ履歴を保存します。
- ON→OFF履歴リレー(CR3501)がONの場合
  アラームリレーがOFF→ONとON→OFFの両方とも履歴を保存します。

CR3502をONすると、アラームリレーの履歴をクリアします。クリアが完了すると、CR3502はOFFします。

■ 操作できる項目

アクセスウィンドウから操作できる機能は次のとおりです。

日本語	英語	中国語	設定内容
モニタ	Monitor	报警监控器	ONしているアラームリレーをリアルタイムでモニタできます。
履歴モニタ	Monitor Record	日志监控器	アラームリレーのON/OFF履歴をモニタできます。
履歴クリア	Clear Record	日志清除	アラームリレーのON/OFF履歴をクリアします。

■ 操作方法

● アラームへの移行

1 「MENU」キーを押します。

メニューが表示されます。(他ページが選択されている場合は「MENU」キーを押してください。)

2 「8.アラーム」を選択して、「ENTER」キーを押します。

アラームのメニューが表示されます。

● モニタ

1 アラームのメニューで「モニタ」を選択して、「ENTER」キーを押します。

アラームモニタが表示されます。

複数のアラームリレーがONしている場合、「▲」「▼」キーを押すと別のアラームを表示します。

①**[01/03]**

現在表示中のアラーム番号／発生しているアラーム総数です。

複数のアラームが発生している場合は、アラームリレー番号の小さいものを優先して表示します。

②**[速度上限異常]**

現在表示しているアラームの対象デバイスのデバイスコメントです。

参 考

- いずれかのキーを押すと、点滅が止まります。
- 他の画面を表示中でも、アラームが一つでも発生すると自動的にアラームモニタが表示されます。

● 履歴表示

1 アラームのメニューで「履歴モニタ」を選択して、「ENTER」キーを押します。

アラーム履歴が表示されます。
「▲」「▼」キーを押すと、1つ前/後の履歴を表示します。

①**[01/04]**
現在表示中のアラーム履歴番号／履歴総数です。
番号は最新の履歴が[01]になります。

②**[温度ワーニング]**
現在表示している履歴の対象デバイスのデバイスコメントです。

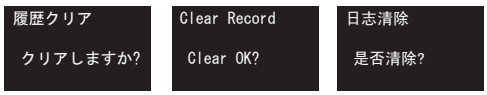
③**[↓]**
アラームリレーがONからOFFになったときの履歴であることを示しています。OFFからONになったときの履歴は「↑」で表示されます。

④**[2015/3/13 12:13:14]**
履歴が記録された時刻を示しています。

● 履歴クリア

1 アラームのメニューで「履歴クリア」を選択して、「ENTER」キーを押します。

履歴クリアの確認メッセージが表示されます。



2 「ENTER」キーを1秒以上押しつづけます。

履歴クリアが終了すると、次の表示になります。

「MENU」キーを押すとアラームメニュー表示に戻ります。

ストレージ

メモリカードやCPUメモリへのデータ保存やファイルの読み出しなどをアクセスウィンドウ上から操作できます。

メモリカードやCPUメモリ機能の詳細は、「5-2 メモリカード」(5-3ページ)、「5-3 CPUメモリ(ユーザメモリ)」(5-27ページ)を参照してください。

■ 操作できる項目

アクセスウィンドウから操作できるメモリカード操作は次のとおりです。

日本語	英語	中国語	設定内容
ロード	Load	加载	任意のプロジェクトフォルダ内のデータをCPUユニットに読み出します。
セーブ	Save	保存	CPUユニットのデータをメモリカードやCPUメモリに保存します。ラダープログラム、デバイス値、デバイスコメント、ユニット設定情報、ロギング設定情報などが保存されます。
ファイル管理	File Management	文件管理	メモリカードやCPUメモリ内のディレクトリ、ファイルを一覧表示します。また、一覧の中から任意のファイルをコピー、移動、削除できます。
空き容量	Free Space	可用空间	メモリカードやCPUメモリの空き容量を表示します。

ポイント **「ロード」はPRGモード時のみ可能です。**

■ 操作方法

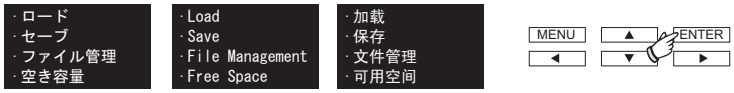
● ストレージへの移行

1 「MENU」キーを押します。

メニューが表示されます。(他ユニットが選択されている場合は「MENU」キーを2回押してください。)

2 「4.ストレージ」を選択して、「ENTER」キーを押します。

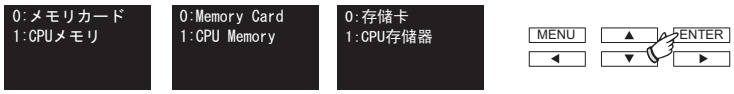
ストレージのメニューが表示されます。



● ロード

1 ストレージのメニューで「ロード」を選択して、「ENTER」キーを押します。

メモリカードとCPUメモリのどちらのストレージを選択するか表示されます。



2 「▲」「▼」でストレージを選択し、「ENTER」または「▶」キーを押します。

メモリカードまたはCPUメモリのルートフォルダにある(プロジェクト)フォルダが表示されます。(ファイルは表示されません。フォルダ名は"<D>KVSAVE01"のように、最初の3文字が<D>で表されます。)

参 考　フォルダ名が長い場合はスクロールして表示されます。

3 「▲」「▼」でフォルダを選択し、「ENTER」または「▶」キーを押します。

ロード実行の確認メッセージが表示されます。

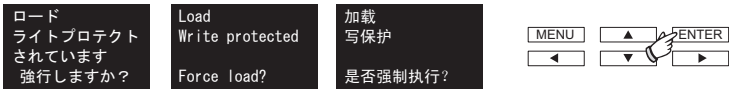
4 「ENTER」キーを1秒以上押し続けて、ロードを実行します。

ロード実行中は次の表示になります。

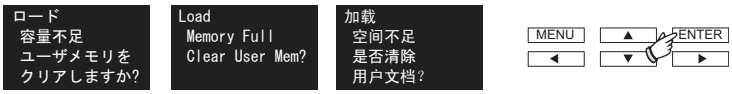
ロードを開始すると、途中でキャンセルはできません。

ポイント

- ライトプロテクトされている場合は、下図の表示が出ます。
  「ENTER」キーを1秒押しつづけると、強行してロードが実行されます。
  「MENU」キーを押すと、ロードがキャンセルされます。

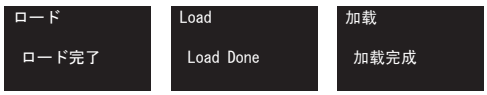


- ロード実行時にCPUメモリの空き容量が少なく、CPUメモリ容量設定を変更できない時には、下図の表示が出ます。
  「ENTER」キーを1秒押しつづけると、CPUメモリクリアが実行されます。
  「MENU」キーを押すと、ロードがキャンセルされます。



- ロード実行時にプロジェクトロードが禁止されている場合、下図の表示が出ます。
  「MENU」キーを押すとロードがキャンセルされます。

5 ロードが完了すると次の表示になります。

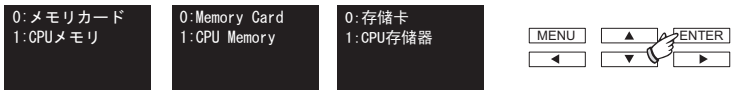


「MENU」「ENTER」「◀」「▶」キーを押すとストレージのメニューに戻ります。

● セーブ

1 ストレージのメニューで「セーブ」を選択して、「ENTER」キーを押します。

メモリカードとCPUメモリのどちらのストレージを選択するか表示されます。



2 「▲」「▼」でストレージを選択し、「ENTER」または「▶」キーを押します。

CPUのデータをセーブするか確認メッセージが表示されます。

3 「ENTER」キーを1秒以上押し続けて、セーブを実行します。

KVSAVE??(?? は二桁の半角数字)という名前のディレクトリを検索し、使われていない最小の二桁の数字のディレクトリを作成してセーブします。
セーブ実行中は現在セーブ中のファイル名と進捗が表示されます。

4 セーブが完了すると、セーブされたプロジェクトフォルダ名が表示されます。

「MENU」「ENTER」「◀」「▶」キーを押すとストレージのメニューに戻ります。

ロード・セーブ実行時のエラーコード

番号	メッセージ	原因	対処	発生時
2	空き番号がありません Unused number not available 没有空编号	既にファイルがあったため書き込めませんでした。	保存したいフォルダ番号に空き番号を設定してください。	セーブ
3	オートローディング失敗 Auto Loading Failed 自动加载失败	オートロードまたはランロード実行時に、CMで指定したフォルダ番号が範囲外でした。	0～99の番号を指定してください。	ロード/セーブ
4	ロードできません Cannot load data 无法加载	メモリカード/CPUメモリに指定したCMのフォルダがありません。	プロジェクトデータが入っていることを確認してください。	ロード
5	メモリカードがありません ---------- CPUメモリアンマウント Memory card not found ---------- CPU Memory unmounted 没有存储卡 ---------- CPU存储器卸载	メモリカード/CPUメモリを認識できない状態でロードまたはセーブを実行しました。	メモリカードからロード、またはメモリカードへセーブする場合は、メモリカードを挿入してください。CPUメモリをマウントしてください。	ロード/セーブ
6	アクセス競合 Access conflict 访问冲突	他の経路からプログラムの転送または読出しをしていたため、セーブ・ロードができませんでした。	プログラムの転送もしくは読み出しが終わってからセーブ・ロードを実行してください。	ロード/セーブ
7	オートローディング失敗 Auto Loading Failed 自动加载失败	ライトプロテクト/プロジェクトロード禁止が設定されているPLCに対して、オートロードまたはランロードを実行しました。	PLCのライトプロテクト/プロジェクトロード禁止を解除してから実行してください。	ロード
9	CPUユニット機種不一致 PLC model mismatch CPU单元机型不一致	別機種のメモリカードファイルを読み込もうとしました。	プロジェクトのCPUユニット機種を確認してください。	ロード
10	非対応のCPU機能バージョン Unsupported Function Version 不支持的功能版本	対応していない機能バージョンまたは別機種のプロジェクトを読み込もうとしました。	プロジェクトのCPUユニットの機能バージョンと機種を確認してください。	ロード
11	PRGモード時のみ可能です PRG mode only 仅PRG模式时可执行	RUNモードでロードしようとしました。	PRGモードに切り換えてロードしてください。	ロード
12	空き容量がありません No free space 没有可用空间	メモリカード/CPUメモリの空き容量がありません。	空き容量を増やしてください。	ロード/セーブ
15	プロジェクト領域サイズオーバー Project area size over 项目区域容量超出	CPUメモリ容量設定の書き込みに失敗しました(プロジェクト領域が不足しています)。	CPUメモリ容量設定の設定を見直すか、CPU内のプロジェクト領域のデータを削減してください。	ロード
16	ユーザメモリ領域サイズオーバー UserMem area size over 用户存储器区域容量超出	CPUメモリ容量設定の書き込みに失敗しました(ユーザメモリ領域が不足しています)。	CPUメモリ容量設定の設定を見直すか、CPU内のユーザメモリ領域のデータを削減してください。	ロード
17	スイッチがPRG時のみ可能です Only when switch in PRG 仅开关处于PRG时可执行	スイッチ状態がRUNの時に、システムプログラムバージョンアップを実行しようとしました。	スイッチをPRGにしてから、システムプログラムバージョンアップを実行してください。	ロード

番号	メッセージ	原因	対処	発生時
18	ユニット種別不一致 Unit Type　Mismatch 单元类型不一致	接続されているユニットと、一致していないユニットのYTCファイルをロードしようとしました。	YTCファイルのユニット種別を確認してください。	ロード
19	オールクリアが必要 All Clear Reqred Before loading 加载前需全部清空	事前にオールクリア/初期化が必要なプロジェクトをロードしようとしました。PLC内部のプロジェクトとロードするプロジェクトのバージョンが一致しない場合、ロード実行前にオールクリア/初期化を実行する必要があります。	オールクリア/初期化実行後にロードを実行してください。CPUメモリからロードを実行する場合は、ロードするデータもクリアされるため、事前にバックアップを実施してください。	ロード
20	ユニット非対応 機能バージョン Unsupported Unit Function Version 不支持的单元功能版本	拡張ユニットが対応していない機能バージョンのセーブデータを読み込もうとした。	プロジェクトの拡張ユニット機能バージョンを確認してください。拡張ユニットの機能バージョンを確認してください。	ロード
100	メモリカードカバーオープン Memory card cover open 存储卡外盖打开	メモリカードカバーが開いています。	メモリカードカバーを閉じてください。	ロード/セーブ
101	メモリカードがLOCKされています Memory card locked 存储卡被锁定	メモリカードがLOCKされています。	LOCKを解除するか、書込み可能なメモリカードを使用してください。	セーブ
102	ロードできません ---------- セーブできません Cannot load data ---------- Cannot save data 无法加载 ---------- 无法保存	メモリカードが破損しています。	メモリカードのファイルシステムをチェックしてください。	ロード ---------- セーブ
103	ファイル数上限を超過しました Max no. of files exceeded 超过了文件数上限	ユーザメモリ内のファイル数とフォルダ数の合計が上限を超過しています。	ユーザメモリ内のファイルまたはフォルダを削減してください。	ロード/セーブ
104	フォルダ階層が深すぎます Folder layer too deep 文件夹层数过多	ユーザメモリ内のフォルダ階層が深すぎます。	ユーザメモリ内のフォルダ階層を確認してください。	ロード/セーブ
1000	ロードできません Cannot load data 无法加载	• ファイルが破損しています。 • メモリカードが破損しています。 • ロード処理中にカードが抜かれました。 • 「ユーザメモリをクリアしてからロードする」の設定で、CPUメモリからロードしようとしました。 • 「オールクリアしてからロードする」の設定で、オートロード/ランロード以外の手段やCPUメモリからロードを実行しようとしました。	プロジェクトやメモリカードが適切かどうか確認してください。	ロード

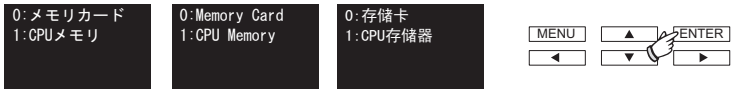
番号	メッセージ	原因	対処	発生時
1001	セーブできません	• メモリカードが破損しています。 • セーブ処理中にカードが抜かれました。	プロジェクトやメモリカードが適切かどうか確認してください。	セーブ
	Cannot save data			
	无法保存			

参 考　ロード･セーブの結果はCMでも確認できます。

● ファイル管理

1 ストレージのメニューで「ファイル管理」を選択して、「ENTER」キーを押します。

メモリカードとCPUメモリのどちらのストレージを選択するか表示されます。



2 「▲」「▼」でストレージを選択し、「ENTER」または「▶」キーを押します。

ルートフォルダにある(プロジェクト)フォルダおよびファイルが表示されます。

参 考

- 長いファイル名やフォルダ名はスクロールして表示されます。
- フォルダ名は、"<D>KVSAVE01"のように、最初の3文字が<D>で表されます。
- フォルダを移動する際には、「▶」キーまたは「◀」キーを使用してください。
- 「<D>.」は現在のフォルダ、「<D>..」は一つ上の階層のフォルダを表します。

3 「▲」「▼」でフォルダやファイルを選択し、「ENTER」キーを押します。

操作一覧が表示されます。

4 「▲」「▼」で操作を選択し、「ENTER」または「▶」キーを押します。

各操作の画面が表示されます。

「ファイル情報」(5-148ページ)
「コピー」(5-149ページ)
「移動」(5-151ページ)
「削除」(5-153ページ)

操作完了後、「MENU」「ENTER」「◀」「▶」キーを押すとファイル一覧画面に戻ります。

- **ファイル情報**

1 ファイルの情報を確認します。

参 考 長いファイル名はスクロールして表示されます。

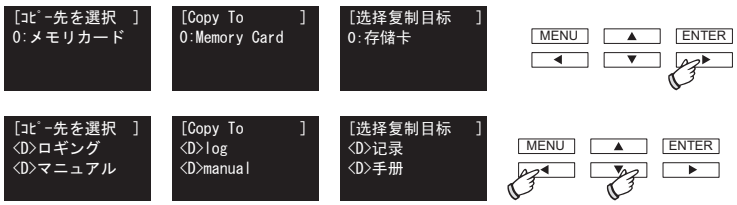
• コピー

1 「ENTER」または「▶」キーを押します。

操作ガイド画面が表示されます。

2 操作を確認し、「▶」キーを押します。

コピー先のフォルダ選択画面が表示されます。

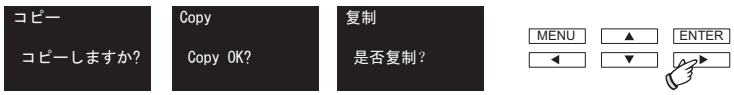


• フォルダを移動する際には、「▶」キーまたは「◀」キーを使用してください。
• 「<D>.」は現在のフォルダ、「<D>..」は一つ上の階層のフォルダを表します。

参 考 コピーはメモリカード→CPUメモリ、または、CPUメモリ→メモリカード、など異なるストレージに対してのみ実行できます(同じストレージのフォルダにはコピーできません)。

3 「ENTER」キーを押します。

コピー確認画面が表示されます。



• フォルダを移動する際には、「ENTER」/「▶」キーまたは「MENU」/「◀」キーを使用してください。
• 「<D>.」は現在のフォルダ、「<D>..」は一つ上の階層のフォルダを表します。

4 「ENTER」キーを1秒以上押し続けて、ファイルをコピーします。

ポイント

- コピー元と同名のファイル/フォルダが下図の表示が出ます。
  「ENTER」キーを1秒押しつづけると、上書きされます。
  「MENU」または「◀」キーを押すと、コピーがキャンセルされます。

- コピー中に「MENU」キーを押すとコピーを中断できます。

下図の画面で、「ENTER」キーを1秒以上押し続けてコピーをキャンセルします。

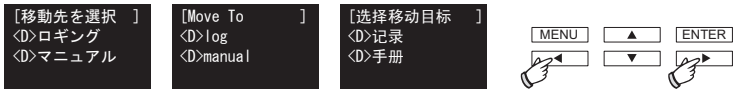
- **移動**

1 「ENTER」または「▶」キーを押します。

操作ガイド画面が表示されます。

2 操作を確認し、「▶」キーを押します。

移動先のフォルダ選択画面が表示されます。

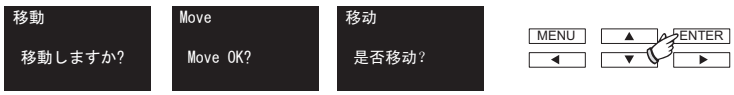


- フォルダを移動する際には、「▶」キーまたは「◀」キーを使用してください。
- 「<D>.」は現在のフォルダ、「<D>..」は一つ上の階層のフォルダを表します。

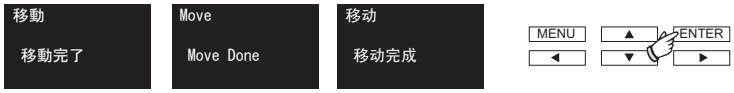
参 考 コピーはメモリカード→CPUメモリ、または、CPUメモリ→メモリカード、など異なるストレージに対してのみ実行できます(同じストレージのフォルダにはコピーできません)。

3 「ENTER」または「▶」キーを1秒以上押し続けます。

移動確認画面が表示されます。



4 「ENTER」または「▶」キーを1秒以上押し続けて、ファイルを移動します。



ポイント

- 移動元と同名のファイル/フォルダが下図の表示が出ます。
  「ENTER」または「▶」キーを1秒押しつづけると、上書きされます。
  「MENU」または「◀」キーを押すと、移動がキャンセルされます。

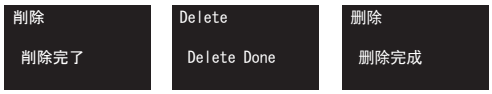
- 移動中に「MENU」キーを押すと移動を中断できます。

下図の画面で、「ENTER」キーを1秒以上押し続けて、移動をキャンセルします。

• 削除

1 「ENTER」キーを1秒以上押し続けます。

2 削除が完了します。



参 考　フォルダを選択した場合、そのフォルダ以下のファイル/フォルダもすべて削除されます。

● 空き容量

1 ストレージのメニューで「空き容量」を選択して、「ENTER」キーを押します。

メモリカードとCPUメモリのどちらのストレージを選択するか表示されます。

2 「▲」「▼」でストレージを選択し、「ENTER」または「▶」キーを押します。

各ストレージの空き容量が表示されます。

アクセスウィンドウ設定

アクセスウィンドウ設定で変更できる設定は次の2つです。

日本語	英語	中国語	設定内容
コントラスト	Contrast	对比	液晶画面(LCD)のコントラストを調整します。(1～11)
コメント番号	Comment Number	注释编号	表示するコメント番号を設定します。 (コメント1～コメント8)

■ コントラスト

1 「MENU」キーを押します。

メニューが表示されます。(他ユニットが選択されている場合は「MENU」キーを2回押してください。)

2 「▲」「▼」キーで「9.AW設定」を選択し、「ENTER」キーを押します。

AW設定メニューが表示されます。

3 「▲」「▼」キーで[コントラスト]を選択し、「ENTER」キーを押します。

4 「▲」「▼」キーでコントラストの数値が増減しますので、見やすいコントラストに調整します。

コントラストは1～11まで設定できます。

■ コメント番号

1 「MENU」キーを押します。

メニューが表示されます。(他ユニットが選択されている場合は「MENU」キーを2回押してください。)

2 「▲」「▼」キーで「9.AW設定」を選択し、「ENTER」または「▶」キーを押します。

AW設定メニューが表示されます。

3 「▲」「▼」キーで[コメント番号]を選択し、「ENTER」キーを押します。

4 「ENTER」または「▶」キーを1秒以上押し続けます。

コメント番号の切り換えができるようになります。

5 「▲」「▼」キーで選択し、「ENTER」キーを1秒以上押し続けます。

コメント番号が切り換わります。

クリア

クリアには、RAMクリア、オールクリア、初期化、ユーザメモリクリアがあります。

■ RAMクリア

RAMクリアでは、デバイス値、アラーム履歴・エラー履歴、PLCイベント履歴、デバイス値変更履歴、ファイルレジスタのカレントバンク番号を一括でクリアすることができます。また、AW設定での設定内容がデフォルトに戻ります。RAMクリアでは、R、MR、LR、B、CR、T/C（現在値）、CM、DM、EM、FM、ZF、W、Z が初期化され、ラダープログラム、ユニット設定情報、コメント、T/C（設定値）、ロギング設定情報は消えません。

ポイント **RAMクリアはPRGモード時のみ実行できます。**

1 「MENU」キーを押します。

メニューが表示されます。（他ユニットが選択されている場合は「MENU」キーを2回押してください。）

2 「▲」「▼」キーで[10.クリア]を選択して、「ENTER」または「▶」キーを押します。

クリアメニュー選択画面が表示されます。

3 「▲」「▼」キーで「RAMクリア」を選択します。

RAMクリア確認メッセージが表示されます。

4 「◀」キーを押しながら「ENTER」キーを1秒以上押し続けます。

再度確認画面が表示されます。

5 2度目の確認画面が表示されている状態で、「ENTER」キーを1秒以上押し続けるとRAMクリアが実行されます。

RAMクリア実行中は次の画面表示になります。RAMクリアが実行され終了すると、自動的に再起動します。

ポイント **「ENTER」キーを1秒以上押し続けなければ[お待ちください･･･]のメッセージは表示されず、無視されます。**

■ オールクリア

CPUユニットを出荷状態にします。(ラダークリア、デバイスオールクリア)

ポイント **オールクリアはPRGモード時のみ実行できます。**

1 「MENU」キーを押します。

メニューが表示されます。(他ユニットが選択されている場合は「MENU」キーを2回押してください。)

2 「▲」「▼」キーで[11.クリア]を選択して、「ENTER」または「▶」キーを押します。

クリアメニュー選択画面が表示されます。

3 「▲」「▼」キーで「オールクリア」を選択します。

オールクリア確認メッセージが表示されます。

4 「◀」キーを押しながら「ENTER」キーを1秒以上押し続けます。

再度確認画面が表示されます。

「MENU」キーを押すとオールクリアがキャンセルされ、メニュー表示に戻ります。

5 2度目の確認画面が表示されている状態で、「ENTER」キーを1秒以上押し続けるとオールクリアが実行されます。

オールクリア実行中は次の画面表示になります。オールクリアが実行され終了すると、自動的に再起動します。

ポイント **「ENTER」キーを1秒以上押し続けなければ[お待ちください･･･]のメッセージは表示されず、無視されます。**

6 再起動後、CPUエラー40のNo Ladder Programのメッセージが表示されます。

■ 初期化

CPUユニット内のすべての領域にNullを書き込むことで、復元できないようにクリアします。

ポイント **初期化はPRGモード時のみ実行できます。**

1 「MENU」キーを押します。

メニューが表示されます。(他ユニットが選択されている場合は「MENU」キーを2回押してください。)

2 「▲」「▼」キーで[11.クリア]を選択して、「ENTER」または「▶」キーを押します。

クリアメニュー選択画面が表示されます。

3 「▲」「▼」キーで[初期化]を選択します。

初期化確認メッセージが表示されます。

4 「◀」キーを押しながら「ENTER」キーを1秒以上押し続けます。

再度確認画面が表示されます。

「MENU」キーを押すとオールクリアがキャンセルされ、メニュー表示に戻ります。

5 2度目の確認画面が表示されている状態で、「ENTER」キーを1秒以上押し続けると初期化が実行されます。

初期化実行中は次の画面表示になります。初期化が実行され終了すると、自動的に再起動します。

ポイント **「ENTER」キーを1秒以上押し続けなければ[お待ちください...]のメッセージは表示されず、無視されます。**

6 再起動後、initializedのメッセージが表示されます。

■ ユーザメモリクリア

ユーザメモリクリアでは、ユーザメモリ内に格納されたロギングデータ、ユーザドキュメントなどのデータをクリアできます。また、ユーザメモリクリアでは、ラダープログラム、ユニット設定情報、コメント、T/C(設定値)、ロギング設定情報は消えません。

ポイント **ユーザメモリクリアはPRGモード時のみ実行できます。**

1 「MENU」キーを押します。

メニューが表示されます。(他ユニットが選択されている場合は「MENU」キーを2回押してください。)

2 「▲」「▼」キーで[10.クリア]を選択して、「ENTER」または「▶」キーを押します。

クリアメニュー選択画面が表示されます。

3 「▲」「▼」キーで「ユーザメモリクリア」を選択します。

ユーザメモリクリア確認メッセージが表示されます。

4 「◀」キーを押しながら「ENTER」キーを1秒以上押し続けます。

再度確認画面が表示されます。

5 2度目の確認画面が表示されている状態で、「ENTER」キーを1秒以上押し続けるとユーザメモリクリアが実行されます。

ユーザメモリクリア実行中は次の画面表示になります。ユーザメモリクリアが実行され終了すると、完了メッセージが表示されます。

ポイント **「ENTER」キーを1秒以上押し続けなければ[お待ちください･･･]のメッセージは表示されず、無視されます。**

KV STUDIOを使用したアクセスウィンドウ設定

KV STUDIOからアクセスウィンドウ設定を使用して、以下の設定ができます。

- 表示言語設定
- 表示コメント番号設定
- バックライトOFF時間設定
- ユーザメッセージ設定

■ 表示言語設定

アクセスウィンドウに表示させる言語を、日本語、英語、中国語（簡体字）から選択できます。
設定を有効にすると、電源投入時の表示言語はKV STUDIOの設定に従います。

ポイント

- **アクセスウィンドウで表示言語を変更できますが、電源を再投入すると、KV STUDIOで設定した値に戻ります。**
- **CMの値を変更して、電源切り入りを実行すると初期の表示言語を設定できます。（KV STUDIOとCMの両方で言語の設定がされている場合は、KV STUDIOの設定が優先されます。）**

■ 表示コメント番号設定

アクセスウィンドウのデバイスコメントをコメント1～コメント8のどれを表示するかを設定できます。
設定を有効にすると、電源投入時の表示言語はKV STUDIOの設定に従います。

表示コメント番号設定の設定値を変更すると、以下を表示する際のコメントが切り換わります。

- デバイスモニタのコメント表示
- アラームモニタのコメント表示
- ユーザメッセージ

ポイント

- **アクセスウィンドウで表示コメント番号設定を変更できますが、電源を再投入すると、KV STUDIOで設定した値に戻ります。**
- **CMの値を変更して、電源切り入りを実行すると初期の表示言語を設定できます。（KV STUDIOとCMの両方で表示コメント番号の設定がされている場合は、KV STUDIOの設定が優先されます。）**

■ バックライトOFF時間設定

アクセスウィンドウのキー操作がなかった場合、バックライトをOFFさせるまでの時間を設定できます。
設定できる時間は、1分/3分/10分/30分/60分/常時ON（初期値）です。

ポイント

- **バックライトOFF中に割り込み画面が表示された場合バックライトは一度ONします。その後、設定した時間経過すると再度OFFします。**
- **バックライトOFF中はキー操作を行うとONします。**
- **PRGモード中はバックライトはOFFしません。**
- **アクセスウィンドウからはバックライトOFF時間の変更はできません。**

■ ユーザメッセージ設定

KV STUDIOから、アクセスウィンドウに表示するメッセージを一覧で設定できます。
ユーザメッセージに対応するCRをONすると、アクセスウィンドウに設定したメッセージエディタにて設定した文言が表示されます。

「ユーザメッセージ」（5-165ページ）

アクセスウィンドウ初期画面表示機能

電源投入時に表示させるアクセスウィンドウの画面をラダープログラムで設定する機能です。
機能を使用しない場合は、前回の電源断時に表示していたページを表示します。

■ 表示方法

電源が入っている状態でラダープログラムなどからコントロールメモリ(CM)に数値を格納し、電源投入時のアクセスウィンドウの初期画面を指定します。
アクセスウィンドウ初期画面表示機能を使用する場合は、CM1780に1を格納し、CM1781～CM1788で表示する画面を設定します。

参 考　10進数は#、16進数は$で表現します。

● アクセスウィンドウ初期画面表示用コントロールメモリ一覧

デバイス番号	機能
CM1780	アクセスウィンドウ初期画面表示機能の有効、無効選択 (0:無効、1:有効)
CM1781	デバイス種別指定※1
CM1782 CM1783	表示画面指定
CM1784 CM1785	デバイス番号指定※2
CM1786	デバイスの表示形式指定※1
CM1787	デバイス名表記選択※1 (0:デバイス番号、1:デバイスコメント)
CM1788	キーロック状態選択 (0:キーロック無効、1:キーロック有効)

※1　表示画面の種類がデバイスモードの場合のみ有効です。
※2　表示画面の種類がデバイスモード、またはトリマモードの場合のみ有効です。

● アクセスウィンドウ初期画面表示機能の有効、無効を選択する

CM1780に格納する数値に応じて、機能を使用するか選択します。

CM1780 = 1 ： CM1781～1789で指定する画面を表示します。

CM1780 = 0 ： 前回の電源断時に表示していた画面を表示します。

● デバイス種別の指定範囲（表示画面の種類がデバイスモードの場合のみ有効）

CM1781に格納する数値に応じて、表示するデバイスの種別を指定します。

デバイス種別	CM1781（16進）	デバイス種別	CM1781（16進）
R	$0000	LR	$0012
CR	$0001	EM	$0017
T	$0002	FM	$0019
C	$0003	B	$001B
DM	$0006	W	$001C
CM	$0007	ZF	$002C
TM	$0008	Z	$0030
MR	$0011		

● 表示画面の種類と指定範囲

CM1782、CM1783に格納する数値に応じて、表示画面の種類を指定します。

画面	CM1782、CM1783の値（16進）
トップメニュー	$00000000
デバイスモード	$01000000
トリマ	$03000000
時計表示	$05000000
CPUモニタ メニュー	$06000000
CPUモニタ プロジェクト名	$06100000
CPUモニタ バージョン	$06200000
アラーム メニュー	$07000000
アラーム モニタ	$07100000
アラーム 履歴	$07200000
アラーム クリア	$07300000
ストレージ メニュー	$08000000
AW設定 メニュー	$0A000000
AW設定 コントラスト設定	$0A100000
クリアメニュー	$0C000000

● デバイス番号の指定方法

（表示画面の種類がデバイスモードまたはトリマモードの場合のみ有効）

CM1784、CM1785に格納する数値に応じて、表示するデバイス番号を指定します。

- R、MR、LR、CR、の場合、チャンネル番号をまたがって連番で指定します。

計算式は以下のようになります

aaa：チャンネル番号（000～999）

aaabb

bb：接点番号（00～15）

N : CM1784,CM1785に格納する値（10進数）

aaa : チャンネル番号

bb : 接点番号

N ＝ 16 × aaa ＋ bb

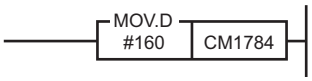
例

R1000を指定する場合

aaa ＝ 10、bb ＝ 0

N ＝ 16 × 10 ＋ 0 ＝ 160

CM1784,CM1785の値は160（$A0）となります。

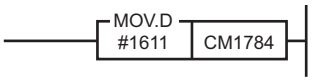


R10011を指定する場合

aaa ＝ 100、bb ＝ 11

N ＝ 16 × 100 ＋ 11 ＝ 1611

CM1784,CM1785の値は1611（$064B）となります。



- B、Wの場合表示するデバイス番号の数値（16進数）で指定します。

例

B1000を指定する場合

CM1784、CM1785の値は4096（$1000）となります。

W1FFFを指定する場合

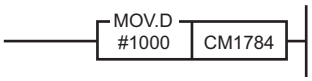
CM1784、CM1785の値は8191（$1FFF）となります。

- T、C、DM、CM、TM、EM、FM、ZFの場合、表示するデバイス番号の数値（10進数）で指定します。

例

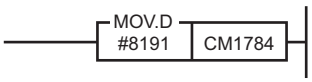
DM1000を指定する場合

CM1784、CM1785の値が1000（$3E8）となります。



DM8191を指定する場合

CM1784、CM1785の値が8191（$1FFF）となります。



● デバイスの表示形式と指定範囲

CM1786に格納する数値に応じて、デバイスの表示形式を指定します。

表示形式	CM1786の値（16進）	表示形式	CM1786の値（16進）
デフォルト※1	$0000	10進数32BIT	$0006
2進数1BIT	$0001	±10進数32BIT	$0007
10進数16BIT	$0002	16進数32BIT	$0008
±10進数16BIT	$0003	ASCII16BIT	$0009
16進数16BIT	$0004	FLOAT	$000A
2進数16BIT	$0005	DOUBLE FLOAT	$000B

※1 ビットデバイスはON/OFF、ワードデバイスは16Bit-BASE10　符号なしで表記します。

■ 初期表示言語設定

CM1736に格納する数値に合わせて、初期表示する言語を変更します。

1 : 英語（初期値）
2 : 日本語
3 : 中国語（簡体字）
上記以外の設定値は無効です。

ポイント **KV STUDIOのアクセスウィンドウ設定で、表示言語設定をしている場合は、KV STUDIOの設定を優先します。**

■ 初期表示コメント番号設定

CM1737に格納する数値に合わせて、初期表示するコメントを変更します。

1～8 : コメント1～コメント8
上記以外の設定値は無効です。

ポイント **KV STUDIOのアクセスウィンドウ設定で、初期表示コメント番号設定をしている場合は、KV STUDIOの設定を優先します。**

■ 表示例

1. 電源投入時のアクセスウィンドウがトップメニューを表示するように指定します。
   以下の設定をCMに格納します。

```
1.デバイスモード
2.エラークリア
3.ユニットテスト
4.ストレージ
```

```
1.Device Mode
2.Error Clear
3.Unit Test
4.Storage
```

```
1.软元件模式
2.错误清除
3.单元测试
4.存储
```

デバイス番号	CMの値（16進）	設定内容
CM1780	$0001	AW初期画面表示有効
CM1781	–	–
CM1782 CM1783	$00000000	トップメニュー
CM1784 CM1785	–	–
CM1786	–	–
CM1787	–	–
CM1788	$0000	キーロック無効

2. 電源投入時のアクセスウィンドウが、デバイスモード、R200のON/OFF表示するように指定します。
   以下の設定をCMに格納します。

```
R200
                OFF
R201
                 ON
```

デバイス番号	CMの値（16進）	設定内容
CM1780	$0001	AW初期画面表示有効
CM1781	$0000	R
CM1782 CM1783	$01000000	デバイスモード
CM1784 CM1785	$00000020	200
CM1786	$0000	デフォルト
CM1787	$0000	デバイス番号表記
CM1788	$0000	キーロック無効

ユーザメッセージ

ユーザメッセージ機能とは、メッセージエディタにて設定した半角64(全角32)文字までのメッセージをアクセスウィンドウに表示させる機能です。
ユーザメッセージに使用できるCRは63個用意されており、CRをONするだけで表示する文字列を変更できます。また表示コメント番号設定で、コメント1～8を切り換えると最大8言語のユーザメッセージを使用できます。

■ 表示方法

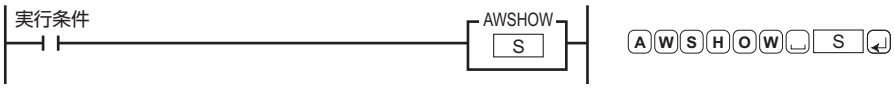
ユーザメッセージを表示させる方法には以下の2つがあります。

①AWSHOW命令を使用する。

専用の命令語を使用することでコントロールリレー(CR)を気にすることなく簡単にユーザメッセージを表示させることができます。
複数IDのユーザメッセージが同時にONした場合、数値の若いユーザメッセージが優先して表示されます。

各命令語についての詳細は、『KV-8000/7000/5000/3000/1000·KV Nanoシリーズ命令語リファレンスマニュアル「5-14 アクセスウィンドウ命令」』を参照してください。

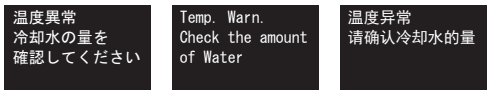
ラダープログラム　　入力方法



②コントロールリレー(CR)を使ってラダープログラムを作成する。

ID1のユーザメッセージを表示するには、CR4601をONします。
CR4601～CR4915をONすると、対応するIDのユーザメッセージが表示されます。

■ 表示例



表示中はCR4500からCR4503で指定した行が点滅します。
次の場合に表示は元に戻ります。

- アクセスウィンドウの「ENTER」キーを押したとき。このとき対応しているCRはOFFします。
- 対応しているCRをOFFしたとき。
- AWHIDE命令を実行したとき。

ポイント

- **電源投入時に表示するユーザメッセージIDを設定できます。**
- **ユーザメッセージ表示時に「ENTER」キーを押すとユーザメッセージを解除するかどうかを設定できます。**
- **現在表示中のユーザメッセージIDはCM1720で確認することができます。**
- **ユーザメッセージ表示中に「MENU」キーを押した場合、アクセスウィンドウの右上に「M」が表示されます。**

ユーザメッセージ中に他のメニューを表示することはできますが、キー操作が30秒以上なかった場合ユーザメッセージ表示画面に自動的に切り換わります。

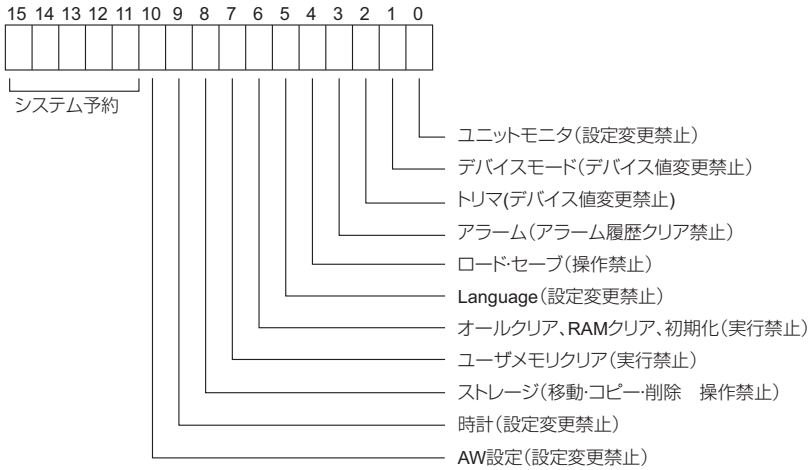
- **複数のユーザメッセージIDのCRがSETされた場合、右端にスクロールボタン(▲ / ▼)が表示されます。スクロールボタンが表示されている間に、「▲」または「▼」キーを押すと別のIDのユーザメッセージを表示できます。**

操作禁止設定

アクセスウィンドウ上での操作を機能別に禁止できます。
操作禁止の設定が可能な機能と項目は次のとおりです。

機能	項目
ユニットモニタ	設定変更
デバイスモード	デバイス値変更
トリマ	デバイス値変更
アラーム	アラーム履歴クリア
ロード･セーブ	ロード、セーブ
Language	設定変更
オールクリア/RAMクリア/初期化	実行
ユーザメモリクリア	実行
ストレージ	ファイル移動、コピー、削除
時計	設定変更
コメント切換	設定変更

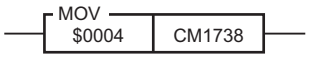
操作禁止設定はCM1738に機能ごとに1ビットずつ割り当てられていて、「ON」の場合に禁止します。



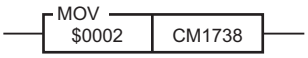
禁止されている操作をすると、次の画面が表示されます。

例

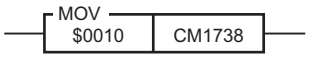
●デジタルトリマの書き込み禁止設定



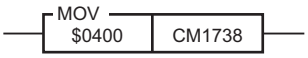
●デバイスモードでの書き込み禁止設定



●ロード･セーブ操作禁止



●AW設定の設定変更禁止



ポイント

デバイスモードでデバイス書き込み禁止を設定すると、以後、アクセスウィンドウで解除できなくなります。

キーロック

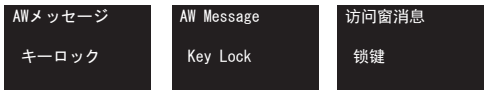
キーロックは、アクセスウィンドウのキー操作を禁止する機能です。設定変更できないようにする場合や表示する画面を固定する場合に使用します。

■ 操作方法

設定／解除いずれの場合も、「◀」キーを押しながら「MENU」キーを3秒以上押し続けてください。

● キーロックされていない場合

次の画面が表示され、キーロック状態になります。



キーロック状態でキーロック解除以外の操作をおこなうと、上の画面が表示されます。

● キーロックされている場合

次の画面が表示され、キーロックが解除されます。

ポイント

- キーロック中は、ダイレクトアクセススイッチを押しても選択されません。
- ユニットのモニタ、ユニットテスト中は、キーロックできません。

MEMO

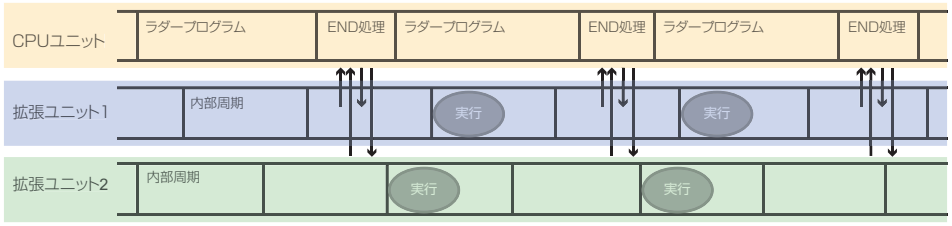
5-10 ユニット間同期機能

ユニット間同期とは

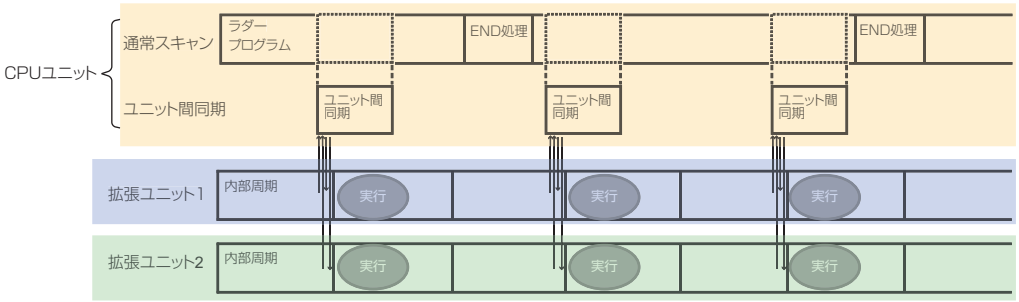
ユニット間同期機能は，複数ユニット間の制御タイミングを合わせる機能です。本機能を使用すると，同期対象のユニットがユニット間同期周期に入力または出力タイミングを合わせることができます。
ユニット間同期周期とは，ユニット間同期機能の定周期間隔です。
この機能を使用することで、複数のユニット間での同時サンプリング、同時位置決めなどが実現できます。

ユニット間同期の動作イメージ

ユニット間同期機能を使用しない場合の動作



ユニット間同期を使用する場合の動作



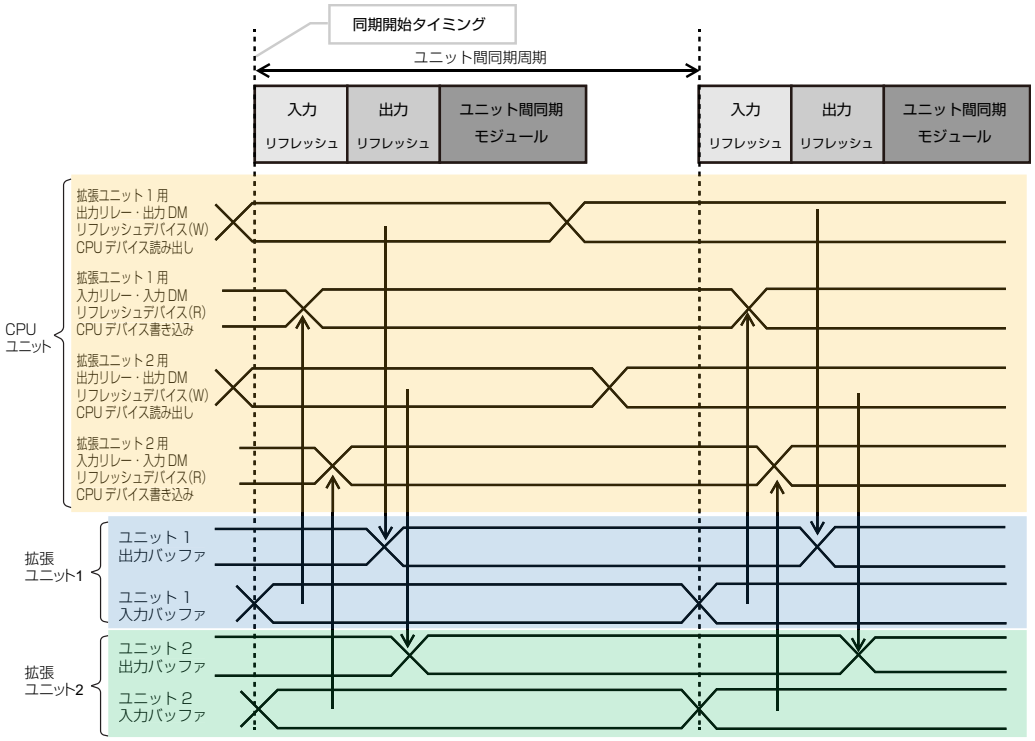
各拡張ユニットの内部周期を調整して内部周期の開始タイミングを調整するため、拡張ユニット間でばらつき1μs以下の同時制御、同時サンプリングができます。
開始タイミング、サンプリングタイミングの調整方法については各ユニットにより異なります。
詳細については各ユニットのユーザーズマニュアルを参照してください。

ポイント

- **ユニット間同期に対応しているユニットはKV-8000/7000シリーズ用拡張ユニット（S-Unit、X-Unit）のみです。KV-5000/3000シリーズ用の拡張ユニットは対応していません。**
- **ユニット間同期機能ではユニット間同期周期後に出力制御が実行されます。実行時高速で出力したい場合などはユニット間同期機能ではなく、ダイレクトリフレッシュ命令を使用してください。**
- **「CPUシステム設定」の「スキャンタイム関連の設定」でユニット間同期モジュールか、毎スキャンモジュール（待機モジュール）を同期するかを選択できます。**

ユニット間同期モジュール

ユニット間同期モジュールはユニット間同期実行開始時に各ユニットに対して出力リフレッシュと、入力リフレッシュを実行します(設定により実施しない場合もあります)。その後ユニット間同期モジュール内のラダープログラムを実行します。
ダイレクトリフレッシュ命令を使用しない場合、ユニット間同期モジュールで変更した内容は次回のユニット間同期実行時に反映されます。
動作指令などはユニット間同期の開始タイミングを基準にタイミングをそろえて実行されます。

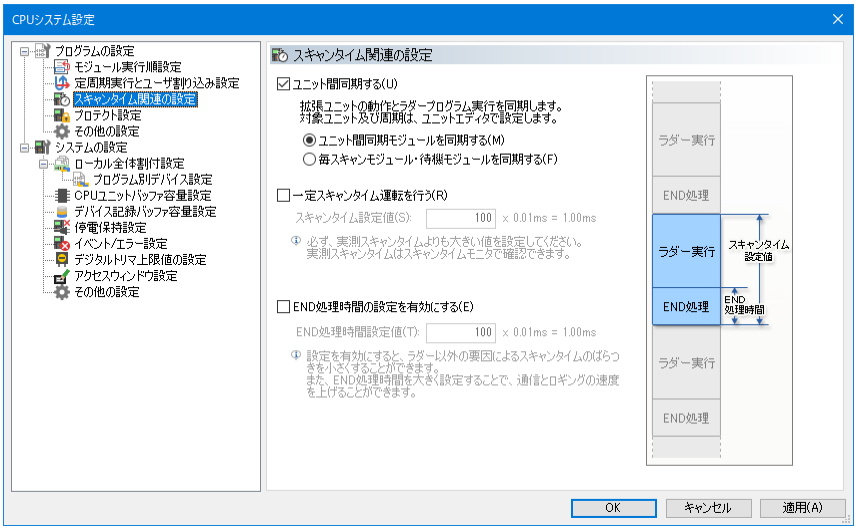


ポイント

- 入力リフレッシュのタイミングで読み出す値は、同期開始タイミングを基準にユニット間で同じタイミングのデータを読み出しできます。読み出すタイミングはユニットごとに異なるため、拡張ユニットのマニュアルを参照してください。
- 出力リフレッシュのタイミングで書き込んだ値(指令)は同期開始タイミングを基準に同じタイミングで実行されます。
- ユニット間同期モジュールの入力リフレッシュ、出力リフレッシュでリフレッシュされるデータは、ユニットごとに異なります。拡張ユニットによって、リフレッシュデバイス(ユニット間同期リフレッシュ)、CPUデバイス書き込み／読み出し機能がないユニットがあります。詳細については、拡張ユニットのマニュアルを参照してください。
- ユニット間同期機能のリフレッシュタイミングを「一括(END処理)」に設定した場合、ユニット間同期の入出力リフレッシュ時はリフレッシュされません。ただし、リフレッシュタイミングを「一括(END処理)」に設定した場合でも、リフレッシュデバイス(ユニット間同期リフレッシュ)やCPUデバイス書き込み/読み出し機能で設定したデバイスは常にユニット間同期の入出力リフレッシュ時にリフレッシュされます。

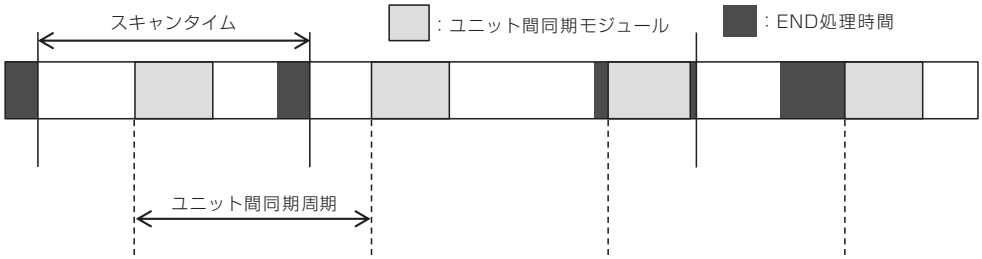
ユニット間同期モジュールを使用しない場合の動作について

ユニット間同期機能使用時、CPUシステム設定の「スキャンタイム関連の設定」で「毎スキャンモジュール･待機モジュールを同期する」にチェックすると、毎スキャンモジュール/待機モジュールをユニット間同期周期で実行することで、ユニット間同期モジュールを使用せずにユニット間同期が実行できます。



ユニット間同期モジュールを使用する場合の動作

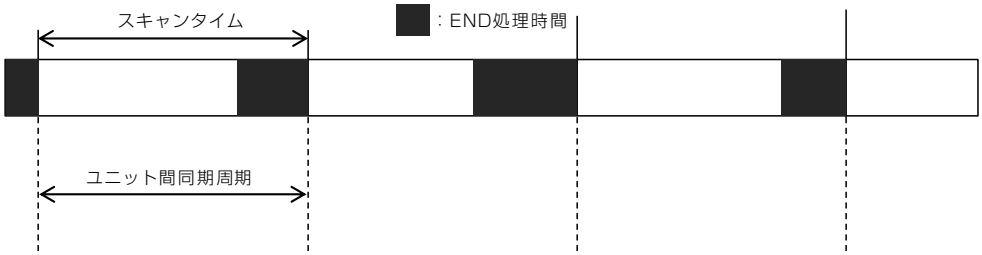
「ユニット間同期モジュールを同期する」を選択している場合



ユニット間同期周期ごとにユニット間同期モジュールを割り込み処理として実行します。

ユニット間同期モジュールを使用しない場合の動作

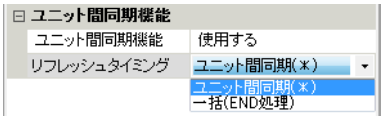
「毎スキャンモジュール･待機モジュールを同期する」を選択している場合



毎スキャンモジュール･待機モジュールがユニット間同期周期ごとに実行されます。つまり、スキャンタイム＝ユニット間同期周期としてCPUユニットのプログラム処理が実行されます。

リフレッシュタイミングの設定（拡張ユニット用リレー・DMのみ対象）

ユニット間同期機能使用時、拡張ユニットのユニット設定で「リフレッシュタイミング」を「一括（END処理）」に設定することで、各ユニットの入出力リレー、入出力DMに対する出力リフレッシュと入力リフレッシュを通常プログラム実行のEND処理で実行することができます。

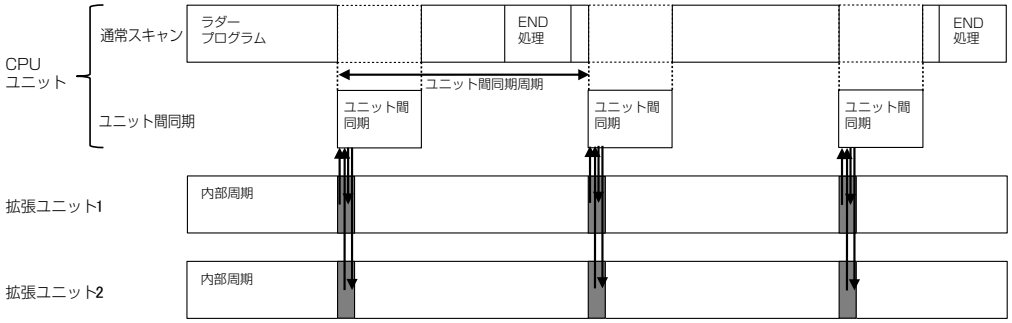


ポイント

リフレッシュタイミングの設定が有効なデータは拡張ユニット用リレー・DMのみです。リフレッシュデバイス（ユニット間同期リフレッシュ）、CPUデバイス書き込み／読み出し機能で設定したデバイスは、常にユニット間同期の入出力リフレッシュ時にリフレッシュされます。

リフレッシュタイミングを「ユニット間同期」に設定した場合

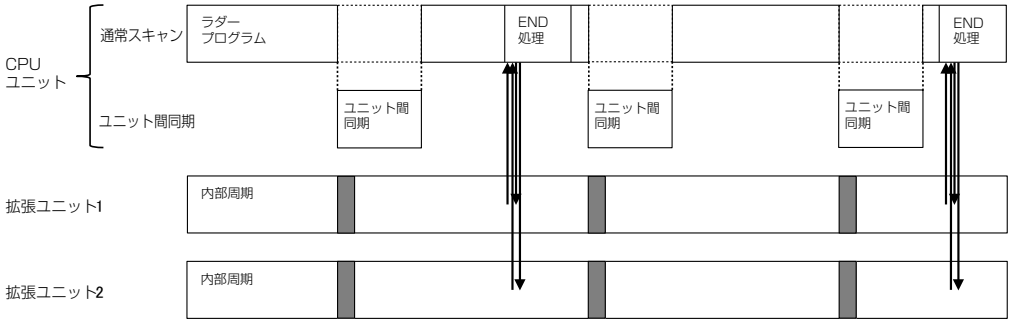
拡張ユニット用リレー・DMは、ユニット間同期実行開始時にリフレッシュ（書き込み／読み出し）されます。



ユニット間同期周期ごとにユニット間同期モジュールを割り込み処理として実行します。

リフレッシュタイミングを「一括（END処理）」に設定した場合

拡張ユニット用リレー・DMは、END処理でリフレッシュ（書き込み／読み出し）されます。



ポイント

ユニット間同期実行時にリレー・DMのリフレッシュが不要な拡張ユニットの設定を「一括（END処理）」に設定することで、ユニット間同期モジュールの実行時間を短くすることができます。反面、「一括（END処理）」に設定した拡張ユニットのリレー・DMのリフレッシュ状態は、ユニット間同期モジュールの実行タイミングの状態とは一致しなくなります。

ユニット間同期リフレッシュデバイス

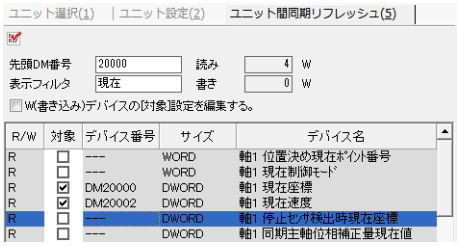
拡張ユニットによっては、拡張ユニットのバッファメモリから必要な機能を選択して、ユニット間同期の実行タイミングで設定したデータメモリ（DM）の現在値としてリフレッシュすることができます。

例：モーションユニットの座標や速度をユニット間同期のタイミングでCPUに読み出したり、反対にモーションユニットの同期制御の位相補正量にユニット間同期のタイミングでCPUユニットからデータを書き込んだりすることができます

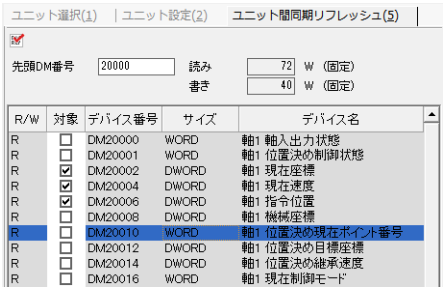
ユニット間同期リフレッシュデバイスの設定方法

ユニット間同期リフレッシュデバイスの設定は、KV STUDIOのユニットエディタで設定します。
ユニットエディタの各拡張ユニットのユニット設定タブで「ユニット間同期機能」を「使用する」に設定すると、「ユニット間同期リフレッシュ（5）」タブが表示されます。この「ユニット間同期リフレッシュ（5）」タブでCPUデバイスに書き込み、読み出ししたい内容を選択します。

モーションユニット≪KV-XH16ML/XH04ML≫の場合



パルス列位置決めユニット≪KV-SH04PL≫の場合



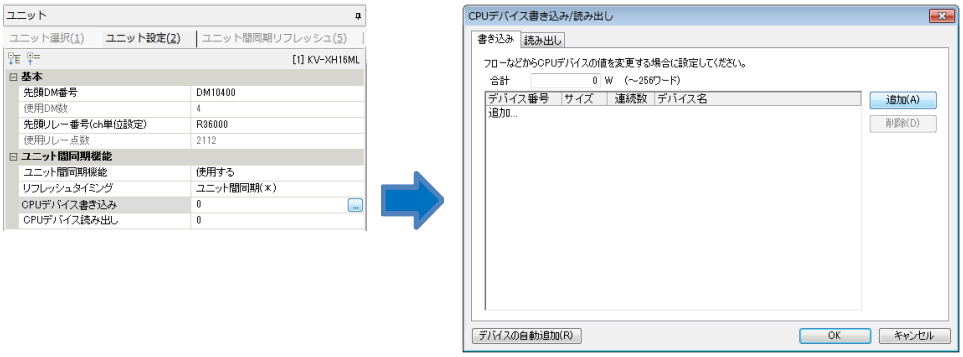
ユニット間同期リフレッシュデバイスの詳細については、各拡張ユニットのマニュアルを参照してください。

CPUデバイス書き込み／読み出し

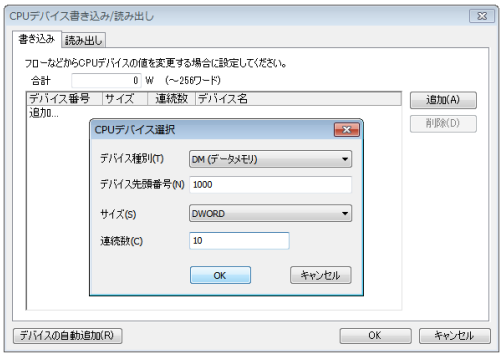
拡張ユニットによっては、CPUユニットの任意のデバイスにユニット間同期リフレッシュのタイミングでデータを書き込んだり、反対にCPUユニットからデータを読み出したりすることができます。ユニットプログラムにCPUのデータを読み出して演算を実行したり、反対に演算結果をCPUに書き込んだりする用途に使用できます。

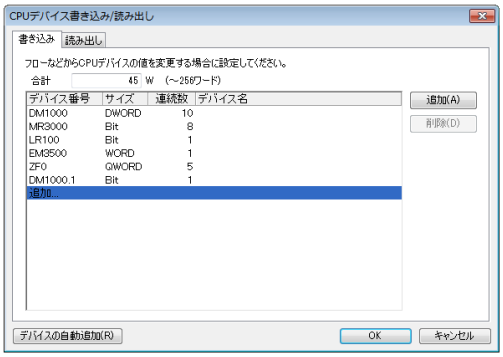
CPUデバイス書き込み／読み出しの設定方法

CPUデバイス書き込み／読み出しをおこなうデバイスは、KV STUDIOのユニットエディタで設定します。
ユニットエディタの各拡張ユニットのユニット設定タブで「ユニット間同期機能」を「使用する」に設定すると、「CPUデバイス書き込み」と「CPUデバイス読み出し」が選択できるようになります。



「CPUデバイス書き込み／読み出し」ダイアログボックス」の〔追加(A)〕ボタンをクリックすると、「CPUデバイス選択」ダイアログボックスを表示し、読み出し、書き込みするデバイスを選択することができます。





CPUデバイス書き込み／読み出しの詳細については、各拡張ユニットのマニュアルを参照してください。

ユニット間同期のシステム構成

対応機器一覧

品名		型式	機能バージョン
CPUユニット		KV-8000(A)	1.0以降
S-Unit	高速入出力ユニット	KV-SIR32XT	1.0以降
	高速アナログ入力/出力ユニット	KV-SAD04/SDA04	1.0以降
	高速カウンタユニット	KV-SSC02	1.0以降
	高速位置決めユニット	KV-SH04PL	1.0以降
X-Unit	位置決め･モーションユニット	KV-XH16ML/XH04ML/XH64EC/XH32EC/XH16EC	1.0以降
	イーサネットユニット	KV-XLE02	1.0以降
	シリアルコミュニケーションユニット	KV-XL402/XL202	1.0以降

ユニット間同期の周期

ユニット間同期で指定できる周期は40.0μs～10000.0μs、0.1ms～10.0msです。
ユニットエディタのCPUユニットの［ユニット設定］タブで設定します。
指定した周期でユニット間同期が実行されます。

ポイント

- **ユニット間同期リフレッシュのタイミングを「ユニット間同期」とした場合、ユニット間同期周期は125μs以上に設定してください。**
- **KV-SH04PLの標準モードやKV-XH16ML/XH04ML/XH64EC/XH32EC/XH16ECなど、一部のユニットではユニット間同期の周期を、ユニットの制御周期単位で設定する必要があります。その場合、設定できる最小のユニット間同期の周期は、最も大きい制御周期をもつユニットの制御周期に従います。**

ユニット間同期の性能仕様

ユニット間同期実行時に次のリフレッシュ時間が必要になります。
リフレッシュ時間に対して十分(5倍+75μs以上)に余裕がある周期をユニット間同期周期に設定してください。

N:ユニット番号
In_R[N]:ユニット番号Nの割り付け入力リレーのch数(占有ch数)
In_DM[N]:ユニット番号Nの割り付け入力DMのワード数(占有DM数)
Out_R[N]:ユニット番号Nの割り付け出力リレーのch数(占有ch数)
Out_DM[N]:ユニット番号Nの割り付け出力DMのワード数(占有ch数)
In_Buf[N]:ユニット番号Nから読み出すバッファメモリのワード数
Out_Buf[N]:ユニット番号Nに書き込むバッファメモリのワード数
In_CPU[N]:ユニット番号Nから読み出すCPUデバイスのワード数
Out_CPU[N]:ユニット番号Nに書き込むCPUデバイスのワード数
T_Ref[N]:ユニット番号Nの割り付けられたデバイスのリフレッシュ時間
T_Buf[N]:ユニット番号Nのバッファメモリ、CPUデバイスのリフレッシュ時間
T_Total:全ユニットの同期リフレッシュ時間

T_Ref[N]の計算式
(a)T_Ref[N] = 0
(b)T_Ref[N] = 2.5 + (0.04 × (In_R[N] + In_DM[N] + Out_R[N] + Out_DM[N]))

※(a) ユニット割り付けデバイスのリフレッシュタイミングをEND処理に設定している場合
(b) ユニット割り付けデバイスのリフレッシュタイミングをユニット間同期に設定している場合

T_Buf[N]の計算式
(a)T_Buf[N] = 0
(b)T_Buf[N] = 0.04 × (In_Buf[N] + Out_Buf[N]) + 0.1 × (In_CPU[N] + Out_CPU[N])

※(a) バッファメモリ、CPUデバイスをリフレッシュしない場合
(b) バッファメモリ、CPUデバイスをリフレッシュする場合

T_Total = T_Ref[1]+T_Buf[1]+ T_Ref[2]+T_Buf[2]+･･･

KV-SAD04を2台使用時の例

T_Ref[N]の計算 (リフレッシュタイミングをユニット間同期に設定)
(1台目)T_Ref[1]＝2.5＋(0.04 × (3＋21＋3＋32)) ＝4.9μs
(2台目)T_Ref[2]＝T_Ref[1]
T_Buf[N]の計算 (バッファメモリはリフレッシュしない)
(1台目)T_Buf[1]＝0μs
(2台目)T_Buf[2]＝T_Buf[1]

T_Total = 4.9＋4.9＋0＋0 = 9.8μs

ポイント

- 各機器の入出力に割り付けられたデバイスは次のとおりです。(初期値)

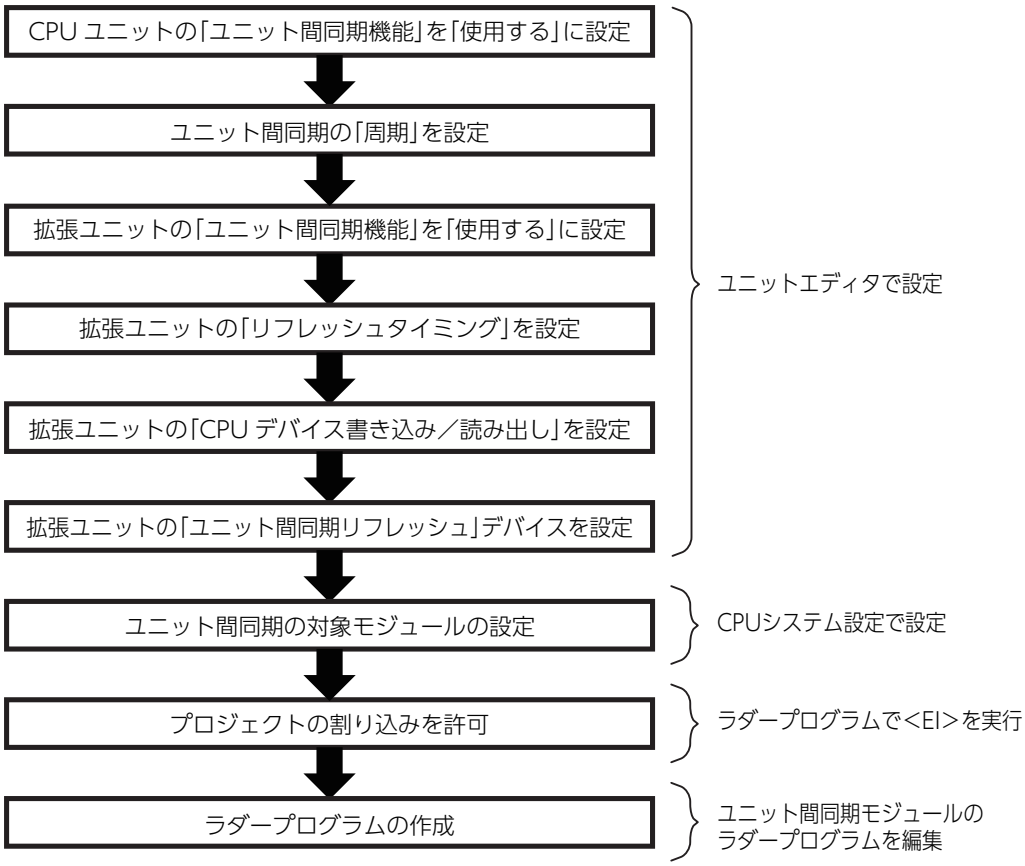
	In_R※1	In_DM※1	Out_R※1	Out_DM※1	In_Buf※2	Out_Buf※2	T_Ref
KV-SIR32XT	2ch(32点)	0	2ch(32点)	0	0	0	2.7μs
KV-SAD04	3ch(48点)	21	3ch(48点)	32	0	0	4.9μs
KV-SDA04	3ch(48点)	5	3ch(48点)	24	0	0	3.9μs
KV-SSC02	2ch(32点)	29	2ch(32点)	40	0	0	5.6μs
KV-SH04PL	10ch(160点)	60	10ch(160点)	0	0	0	5.7μs
KV-XH04ML	18ch(288点)	0	18ch(288点)	0	0	0	3.9μs
KV-XH16ML	66ch(1056点)	0	66ch(1056点)	0	0	0	7.8μs
KV-XH16EC	66ch(1056点)	0	66ch(1056点)	0	0	0	7.8μs
KV-XH32EC	130ch(2080点)	0	130ch(2080点)	0	0	0	12.9μs
KV-XH64EC	258ch(4128点)	0	258ch(4128点)	0	0	0	23.1μs
KV-XLE02※3	20～80ch(1280点)	0	20～60ch(960点)	0	0	0	4.1～8.1μs
KV-XL202※3	0～26ch(416点)	0	2～14ch(224点)	0	0	0	2.7～4.2μs
KV-XL402※3	0～26ch(416点)	0	2～14ch(224点)	0	0	0	2.7～4.2μs

※1　ユニット間同期時にリフレッシュする設定になっている場合の初期値
※2　初期値ではバッファメモリのリフレッシュは登録されていないですが、設定の変更は可能です。
※3　ユニットエディタの設定に応じてリレーの占有ch数は変化します。
T_Refは占有ch数に応じて変化します。

- **ユニット間同期設定時に以下の設定を使用するとリフレッシュ完了までにかかる時間が増加し、ユニット間同期周期の間に処理が完了できなくなる可能性があります。次の条件で使用時はユニット間同期の周期の設定に余裕をもって設定してください。**
  - **KV-5000/3000シリーズ用のユニットを使用している場合(最大14μs遅延)**
  - **DI/DIC命令を使用して割り込みを禁止している場合**

ユニット間同期の設定方法

ユニット間同期を実行するには、次の設定が必要になります。

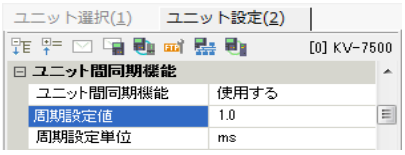


ユニット間同期の設定手順

CPUユニットにKV-XH16MLを2台接続しユニット間同期を実行する場合の手順を説明します。

1 ユニットエディタでKV-8000(A)の設定をします。

KV-8000(A)のユニット設定でユニット間同期機能を「使用する」に設定し、ユニット間同期の「周期」を設定します。
（次の例では1.0msに設定）



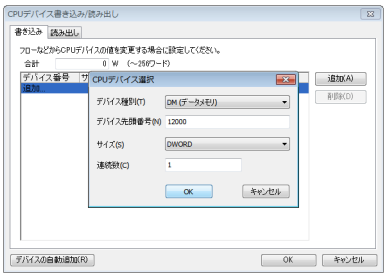
2 ユニットエディタでKV-XH16MLの設定をします。

KV-XH16MLのユニット設定(2)タブでユニット間同期機能を「使用する」に設定します。

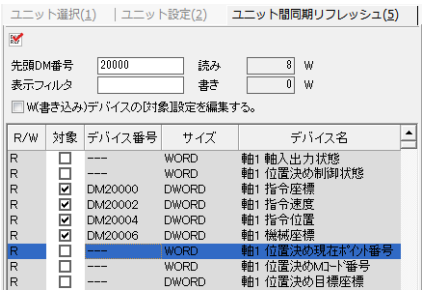
参 考 リフレッシュタイミングを「ユニット間同期」以外に「END処理」に設定することもできます。

必要に応じて、CPUユニットデバイス書き込み／読み出しや、ユニット間同期リフレッシュデバイスを設定します。

「CPUユニットデバイス書き込み／読み出し」の設定



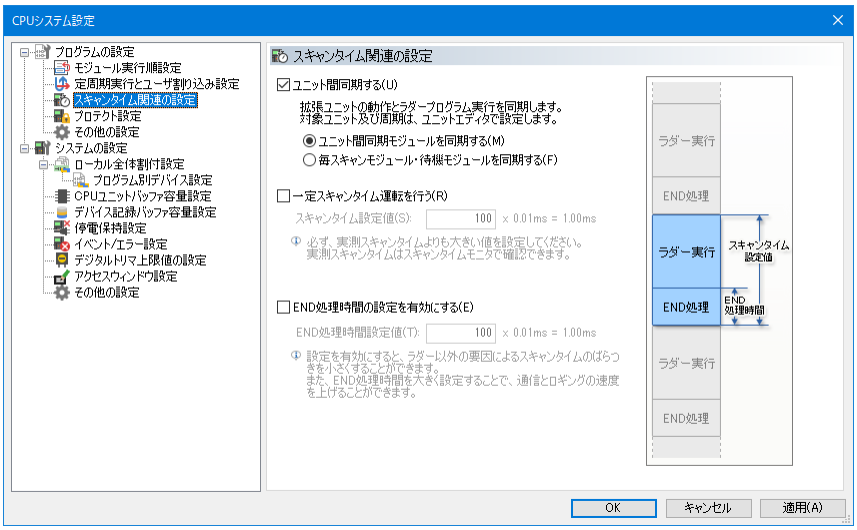
「ユニット間同期リフレッシュデバイス」の設定



2台目のKV-XH16MLにも同様の設定をします。

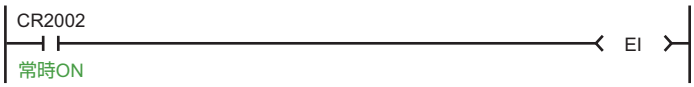
ポイント **KV-8000/7000シリーズ用拡張ユニット（S-Unit、X-Unit）のうち、S-Unitには「CPUユニットデバイス書き込み／読み出し」の設定（機能）はありません。また、ユニットによっては、「ユニット間同期リフレッシュ」の設定がないユニットがあります。**

3 CPUシステム設定でユニット間同期を実行する対象モジュールを設定します。



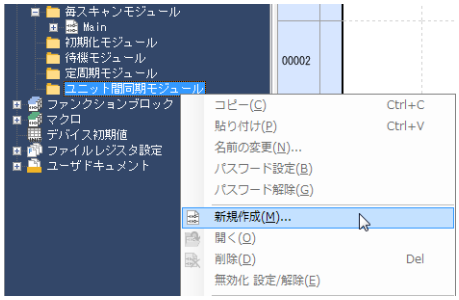
ポイント 「毎スキャンモジュール・待機モジュールを同期する」に設定した場合、手順4～5は不要です。

4 EI(割り込み許可)命令を毎スキャンモジュールなどのラダープログラムに記載します。



ポイント
- ユニット間同期リフレッシュおよびユニット間同期モジュールは割り込みプログラムとして実行されます。
- ユニット間同期割り込みよりも優先度が高い割り込みを実行している場合、ユニット間同期の周期が守れなくなる場合があります。

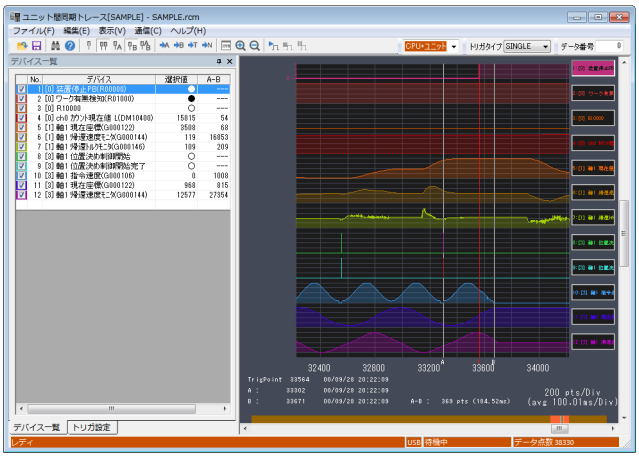
5 ユニット間同期モジュールを作成し、ラダープログラムに記載します。



6 作成したプログラムをPLCに転送します。

ユニット間同期トレース

ユニット間同期トレースとは、ユニット間同期を実行しているKV-8000/7000シリーズ用拡張ユニット(X-Unit)の内部データと、リレー(R)やデータメモリ(DM)などのCPUユニットのデバイス値をユニット間同期周期のタイミングで同時にサンプリングし、波形でモニタする機能です。



参 考　ユニット間同期トレースの基本的な操作方法は、リアルタイムチャートモニタと同じです。詳細は、『KV STUDIOユーザーズマニュアル』「9-6　リアルタイムチャートモニタ」を参照してください。

ポイント　**ユニット間同期トレースに対応しているユニットはKV-8000/7000シリーズ用拡張ユニット(X-Unit)のみです。KV-8000/7000シリーズ用拡張ユニット(S-Unit)、KV-5000/3000シリーズ用拡張ユニットは対応していません。**

■ 起動方法

1 **メニューから【モニタ/シミュレータ(N)】>【リアルタイムチャートモニタ(H)】を選択します。**

[リアルタイムチャートモニタ]が起動します。

2 **動作モードを「CPU+ユニット」に切り換えます。**

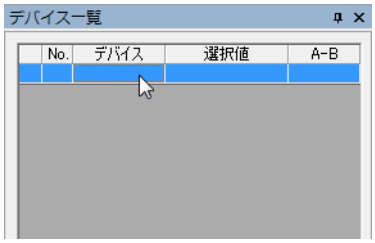
[ユニット間同期トレース]が起動します。

ポイント
- **ユニット間同期トレースは、KV STUDIOから起動したリアルタイムチャートからのみ起動できます。単独起動したリアルタイムチャートモニタからは起動できません。**
- **ユニット間同期トレースは、CPUユニットのユニット間同期機能の設定が「使用する」の場合のみ、使用可能です。「使用しない」に設定されている場合は、表示しません。**

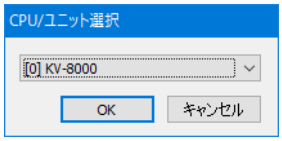
■ デバイスの登録方法

1 **[デバイス一覧]のセルをダブルクリックします。**

[CPU/ユニット選択]ダイアログが表示されます。



2 **ユニットを選択します。**

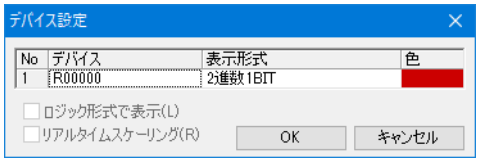


ポイント　**ユニットエディタでユニット間同期機能を「使用する」に設定しているCPUユニットおよびKV-8000/7000シリーズ用拡張ユニット(X-Unit)のみ表示します。**

3 **登録するデバイスを設定します。**

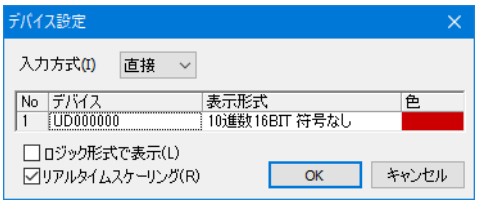
CPUユニットの場合

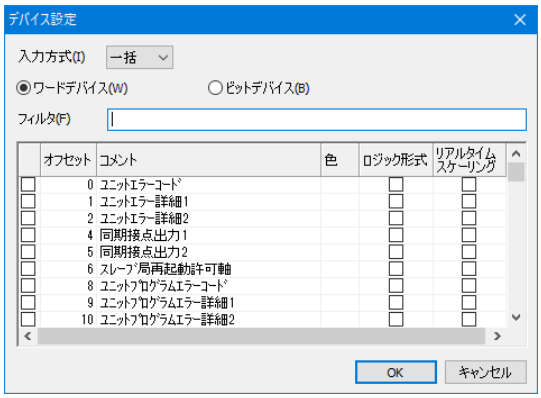
デバイスを直接入力して〔OK〕をクリックします。



KV-8000/7000シリーズ用拡張ユニット(X-Unit)の場合

入力方式を「直接/一括」から選択し、登録するデバイスを指定して〔OK〕をクリックします。





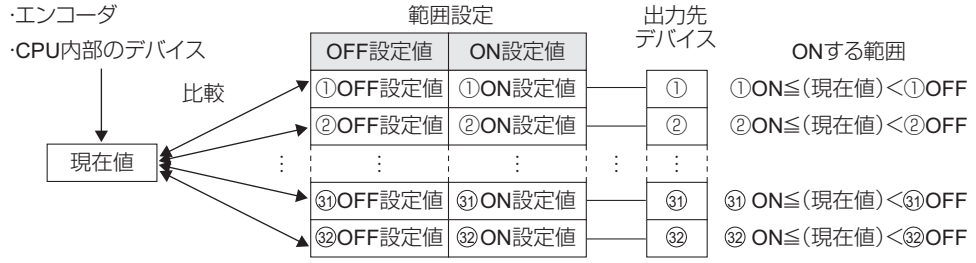
参考
- 「フィルタ」にコメントの一部を入力することで、入力した文字列がコメントに含まれるデバイスに絞り込めます。
- 登録できるデバイスや操作方法の詳細については各ユニットのマニュアルを参照してください。

5-11 カムスイッチ

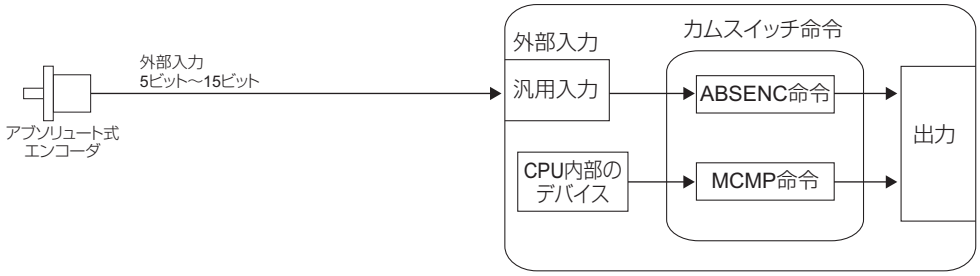
エンコーダからの値を角度変換してカム動作を擬似的に実行できます。

カムスイッチ機能の概要

あらかじめ設定している複数の範囲設定と、エンコーダなどから入力された現在値を比較します。
現在値がどの範囲設定にあるか検出し、該当する範囲設定に対応する出力先デバイスをONします。



入力方式によりそれぞれ専用の命令語を用意しています。



カムスイッチ命令

カムスイッチ命令は、次の2種類の命令があります。

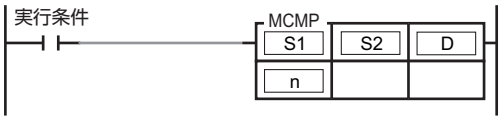
命　令	ニモニック	動作原理	ページ
多段コンパレータ	MCMP	多段コンパレータの動作を行います。	5-183
アブソリュート エンコーダ	ABSENC	アブソリュート式エンコーダを使用してカム動作を行います。	5-187

MCMP

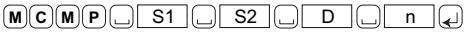
多段コンパレータ

多段コンパレータの動作を行います。

ラダープログラム



入力方法



オペランド	説明	占有サイズ
S1	比較するデータブロックの先頭デバイスを指定します。※1	32×n ビット
S2	比較するデバイスを指定します。※2	16ビット
D	比較結果を格納する先頭デバイスを指定します。※3	n ビット
n	比較するデータ数または、それを格納しているデバイスを指定します。(1～32)※4 ※5	16ビット

※1　ビットデバイスを指定した場合、チャンネルの先頭から連続したn×32ビットを扱います。
ワードデバイスを指定した場合、n×2ワードを扱います。

※2　ビットデバイスを指定した場合、連続した16ビットを扱います。
チャンネルの先頭以外(R002,R106,R1012など)を指定した場合、次のチャンネルにまたがって16ビット扱います。

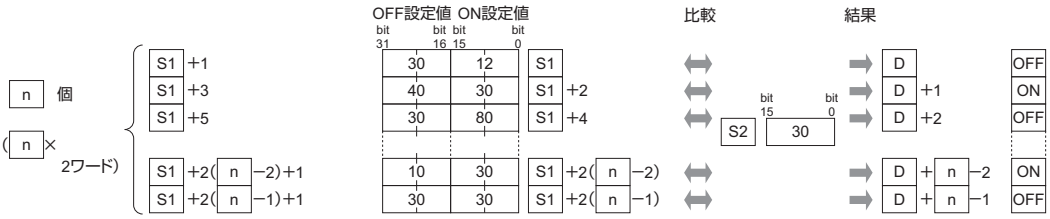
※3　ビットデバイスを指定した場合、nビットを占有します。
ワードデバイスを指定した場合、各ワードデバイスの最下位ビットが結果の格納先になります。
最下位ビット以外は変化しません。

※4　“$”は使用できません。

※5　ビットデバイスを指定した場合、連続する16ビットを占有します。
ワードデバイスを指定した場合、1ワードを占有します。

動作説明

実行条件がONのとき S1 から始まる n 個の32ビットのデータ(16ビット符号無しBINデータ2つ:OFF設定値、ON設定値)と S2 で指定したデバイスに格納されている16ビット符号無しBINデータを比較し、結果を D から始まるビットデバイスに格納します。



OFF設定値＞ON設定値の場合

OFF設定値＞ S2 ≧ON設定値の範囲のとき、D がONします。

OFF設定値＜ON設定値の場合

OFF設定値＞ S2 または S2 ≧ON設定値の範囲のとき、D がONします。

OFF設定値＝ON設定値の場合

常に、D がOFFします。

ポイント **実行条件がOFFしても、出力は変化しません。**

演算フラグ

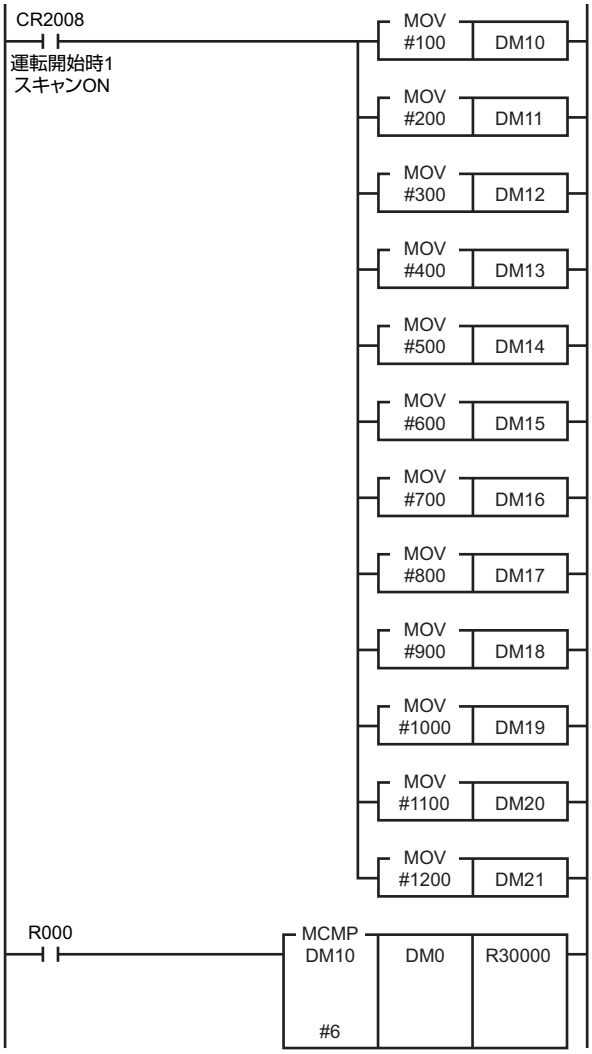
CR2009	変化なし
CR2010	変化なし
CR2011	変化なし
CR2012	以下の場合はON、それ以外はOFF。 ・n に範囲外の値を指定した場合 ・S1 でビットデバイスを指定したときチャンネルの先頭以外を設定した場合 ・間接指定、インデックス修飾の範囲が不適切な場合 ・D に間接指定でタイマ、カウンタを指定した場合

※CR2012がONした場合、命令は実行されません。
CR2012がONした場合、CM5150～CM5176にエラーの詳細情報が格納されます。

「最新エラー詳細(CM)」(4-87ページ)

サンプルプログラム

データメモリDM0の値を設定値と比較して出力リレーを以下の条件でON/OFFします。
DM0が#100以上#200未満の場合はR30000をON
DM0が#300以上#400未満の場合はR30001をON
DM0が#500以上#600未満の場合はR30002をON
DM0が#700以上#800未満の場合はR30003をON
DM0が#900以上#1000未満の場合はR30004をON
DM0が#1100以上#1200未満の場合はR30005をON

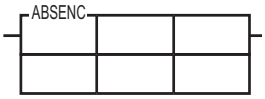


〈ニモニックリスト〉

```
LD CR2008
MOV #100 DM10
MOV #200 DM11
MOV #300 DM12
MOV #400 DM13
MOV #500 DM14
MOV #600 DM15
MOV #700 DM16
MOV #800 DM17
MOV #900 DM18
MOV #1000 DM19
MOV #1100 DM20
MOV #1200 DM21
LD R000
MCMP DM10 DM0 R30000 #6
```

MEMO

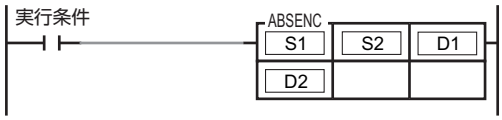
ABSENC



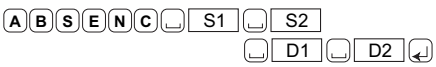
アブソリュート
エンコーダ

アブソリュートエンコーダを使用し、カムスイッチ動作を実行します。

ラダープログラム



入力方法



オペランド	説明	占有サイズ
S1	比較する角度設定値（0.1°単位：0～3599）が格納されているデータブロックの先頭デバイスを指定します。※1	32 × N※5ビット
S2	アブソリュートエンコーダの値を格納しているデバイスを指定します。※2	16ビット
D1	比較結果を格納する先頭デバイスを指定します。※3	N※5ビット
D2	パラメータを格納している先頭デバイスを指定します。※4	16×8ビット

※1 ビットデバイスを指定した場合、チャンネルの先頭から連続した[D2 ＋0]×32ビットを扱います。
ワードデバイスを指定した場合、[D2 ＋0]×2ワードを扱います。
※2 ビットデバイスを指定した場合、連続した16ビットを扱います。チャンネルの先頭以外（R002, R106,R1012など）を指定した場合、次のチャンネルにまたがって16ビットを扱います。
※3 ビットデバイスを指定した場合、[D2 ＋0]ビットを占有します。
ワードデバイスを指定した場合、各ワードデバイスの最下位ビットが結果の格納先になります。最下位ビット以外は変化しません。
※4 ビットデバイスを指定した場合、チャンネルの先頭から連続する128ビットを占有します。
ワードデバイスを指定した場合、8ワードを占有します。
※5 Nは D2 ＋0で指定した比較データ数です。

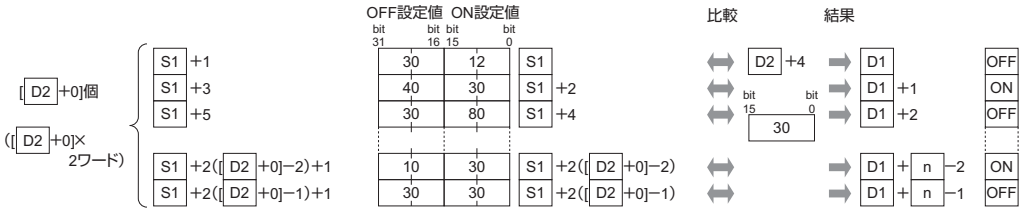
動作説明

パラメータ

デバイス		内　容
D2 ＋0	比較データ数	比較するデータ数を設定します。(1～32)
D2 ＋1	分解能	エンコーダの分解能を設定します。(32～32768)
D2 ＋2	余りグレイコード	余りグレイコードを設定します。(0～16384) (例) 分解能：32,64,128,256,512,1024,2048,4096,8192,16384,32768　36　360　720 余りグレイコード：0　14　76　152
D2 ＋3	ゼロ度設定値	ゼロ度に指定する値を設定します。(0～分解能－1)
D2 ＋4	現在角度 (0.1度単位)	補正後の現在角度が格納されます。(0～3599) ・S2 －[D2 ＋2]≧[D2 ＋3]の場合 (3600÷[D2 ＋1])×(S2 －[D2 ＋2]－[D2 ＋3]) ・S2 －[D2 ＋2]＜[D2 ＋3]の場合 (3600÷[D2 ＋1])×([D2 ＋1]＋S2 －[D2 ＋2]－[D2 ＋3])

デバイス		内　容		
D2 ＋5	動作エラー	D2 ＋5	D1	D2 ＋4
		bit0:出力応答エラー ＜条件＞ 1スキャン内で変化した、2つ以上の出力先のON設定値とOFF設定値が違う場合、またはON設定値どうし、OFF設定値どうしが違う場合エラーになります。 以下の場合はエラーになりません ・同時に変化した出力のON設定値とOFF設定値が同じ場合 ・同時に変化した出力のON設定値どうし、OFF設定値どうしが同じ場合	変化しません	変化します
		bit1:入力値範囲外エラー エンコーダの入力値が設定範囲外の場合、ONします。	変化しません	変化しません
		リセットはラダープログラムでおこなってください。		
D2 ＋6 D2 ＋7	ワークエリア	(ユーザ使用不可)		

実行条件がONのとき S1 から始まる[D2 ＋0]個の32ビットのデータ(16ビット符号無しBINデータ2つ:OFF設定値、ON設定値)と[D2 ＋4]で指定したデバイスに格納された現在角度の値を比較し、結果を D1 から始まるビットデバイスに格納します。



OFF設定値＞ON設定値の場合

OFF設定値＞[D2 ＋4]≧ON設定値の範囲のとき、D1 がONします。

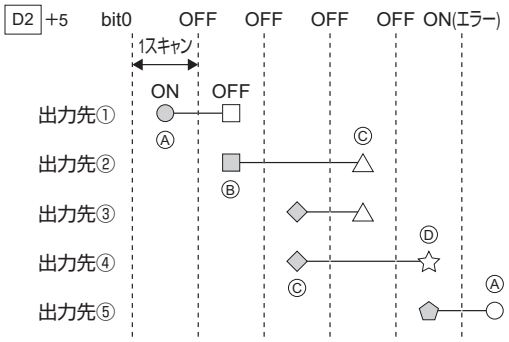
OFF設定値＜ON設定値の場合

OFF設定値＞[D2 ＋4]または[D2 ＋4]≧ON設定値の範囲のとき、D1 がONします。

OFF設定値＝ON設定値の場合

常に、D1 がOFFします。

■ 出力応答エラーについて（D2 ＋5 bit0）



〈エラーにならない場合〉

Ⓐ…1スキャン内で変化した出力先が1つの場合。

Ⓑ…1スキャン内で変化した2つ以上の出力先のON設定値と、OFF設定値が同じ。
・出力先①のOFF設定値＝出力先②のON設定値

Ⓒ…1スキャン内で変化した2つ以上の出力先のON設定値どうし、OFF設定値どうしが同じ。
・出力先②のOFF設定値＝出力先③のOFF設定値
・出力先③のON設定値＝出力先④のON設定値

〈エラーになる場合〉

1スキャン内で変化した2つ以上の出力先のON設定値とOFF設定値が違う場合。または、ON設定値どうし、OFF設定値どうしが違う場合

例）Ⓓ…1スキャン内で変化した2つ以上の出力先のON設定値とOFF設定値が違う
出力先④のOFF設定値≠出力先⑤のON設定値

ポイント

- **動作中に設定値を変更すると即座に出力が反映されます。出力に反映させたくない場合は実行条件をOFFしてください。**
- **実行条件にCR2002（常時ON）のコントロールリレーを使用しないでください。**
- **実行条件がOFFしても、出力は変化しません。**
- **割り込みプログラムおよび初期化モジュール内では使用できません。**

演算フラグ

CR2009	変化なし
CR2010	変化なし
CR2011	変化なし
CR2012	以下の場合はON、それ以外はOFF。 ・エンコーダ値、各パラメータ値、角度設定値に範囲外の値を指定した場合 ・S1 , D2 でビットデバイスを指定したときチャンネルの先頭以外を設定した場合 ・間接指定、インデックス修飾の範囲が不適切な場合 ・S1 , D1 , D2 に間接指定でタイマ、カウンタを設定した場合

※ CR2012がONした場合、命令は実行されません。
CR2012がONした場合、CM5150～CM5176にエラーの詳細情報が格納されます。

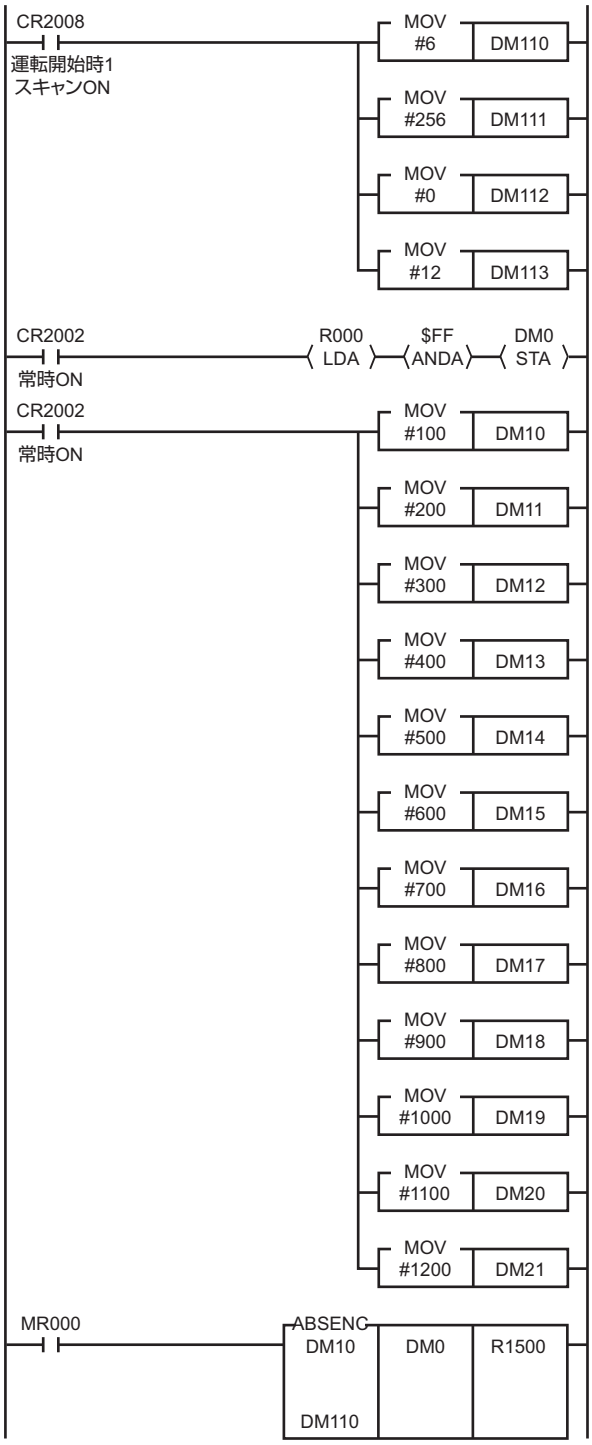
「エラー履歴」（4-86ページ）

サンプルプログラム

アブソリュートエンコーダからの入力値R000～R007をDM0に格納し、設定値と比較して以下の仕様で出力リレーをON/OFFします。

分解能：256/1回転　余りグレイコード：0　ゼロ度ティーチング値：12

DM0が10.0度以上20.0度未満の場合はR1500をON
DM0が30.0度以上40.0度未満の場合はR1501をON
DM0が50.0度以上60.0度未満の場合はR1502をON
DM0が70.0度以上80.0度未満の場合はR1503をON
DM0が90.0度以上100.0度未満の場合はR1504をON
DM0が110.0度以上120.0度未満の場合はR1505をON



〈ニモニックリスト〉

```
LD CR2008
MOV #6 DM110
MOV #256 DM111
MOV #0 DM112
MOV #12 DM113
LD CR2002
LDA R000
CON
ANDA $FF
CON
STA DM0
LD CR2002
MOV #100 DM10
MOV #200 DM11
MOV #300 DM12
MOV #400 DM13
MOV #500 DM14
MOV #600 DM15
MOV #700 DM16
MOV #800 DM17
MOV #900 DM18
MOV #1000 DM19
MOV #1100 DM20
MOV #1200 DM21
LD MR000
ABSENC DM10 DM0 R1500 DM110
```

5-12 時計機能

CPUユニットでは、内蔵時計ICによって、「年、月、日、時、分、秒、曜日」を計時しています。
エラー・電源ON/OFF履歴、トレース機能などの時刻記録に使用します。
時計の精度は常温時（＋25℃）において、±60s/月です。

参 考

- 内蔵時計は大容量キャパシタにより保持されているため、2週間以上（＋25℃）非通電状態が継続すると時計データはクリアされ、2000/1/1になります。
- 時計データがクリアされると、CR2311がONします。時計データを書き込むとOFFします。
- KV-B1を使用すると内蔵時計をバッテリでバックアップできます

■ 時計データの読み出し

時計データはコントロールメモリCM700～CM706に格納されています。
時計データのフォーマットは次のとおりです。

	bit 15 … bit 0		
CM700	年	0～99	年は、西暦2000年の下2ケタを示します。
CM701	月	1～12	データは、すべて16ビットBINデータ形式で記述します。
CM702	日	1～31	
CM703	時	0～23	
CM704	分	0～59	
CM705	秒	0～59	
CM706	曜日	0～6	曜日は、0：(日)、1：(月)、2：(火)、3：(水)、4：(木)、5：(金)、6：(土) になります。

- このデータは直接ラダーで読み出すことができます。（読み出し専用です）

■ 時計データの書き込み（時刻設定）

● アクセスウィンドウで設定

アクセスウィンドウに時計データを表示して時刻設定できます。

「時計」（5-134ページ）

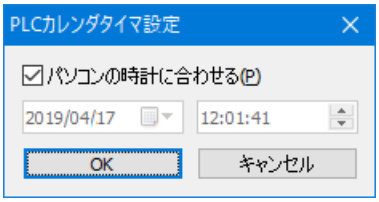
● ラダープログラムで設定

WTIME命令で時刻設定できます。また、AJST命令で0秒補正が可能です。

『KV-8000/7000/5000/3000/1000シリーズ・KV Nanoシリーズ 命令語リファレンスマニュアル』「時計処理命令」

● 《KV STUDIO》で設定

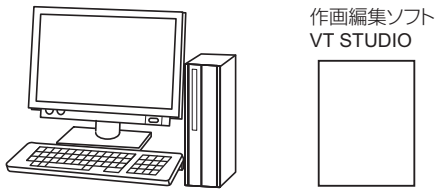
《KV STUDIO》でモニタ中に、メニューから【デバッグ】→【カレンダタイマ設定】を開きます。



『KV STUDIO ユーザーズマニュアル』「カレンダタイマ設定」

● タッチパネル(VT5/VT3シリーズ)で設定

当社製タッチパネル《VT5/VT3シリーズ》のPLC時刻同期機能を使用すると、CPUユニットの時刻を《VT5/VT3シリーズ》の時刻に合わせることができます。



詳細は『VT STUDIOリファレンスマニュアル』を参照してください。

参 考

Ver.4.6以降の《VT STUDIO》には、修復機能が用意されています。
接続先のCPUユニットの時計データがクリアされた場合に自動的に時刻をタッチパネルと合わせます。

6

拡張I/Oユニット機能

拡張I/Oユニット≪KV-SIR32XT≫を使用して実現できる機能を説明します。

6-1 入出力機能 6-2
6-2 ユニット割り込み機能 6-4
6-3 ユニット間同期機能 6-6
6-4 インプットキャプチャ機能 6-9
6-5 バッファリング入力機能 6-10
6-6 PWM出力機能 6-19
6-7 KV-SIR32XT用ユニット専用命令 6-23
6-8 KV-SIR32XT用ユニット専用関数 6-30
6-9 アクセスウィンドウ 6-37
6-10 ユニットモニタ 6-42
6-11 I/Oユニット参考プログラム例 6-43
6-12 I/Oユニットのエラーコード 6-47

6-1 入出力機能

≪KV-SIR32XT≫の入出力は汎用入出力のほかに以下の各機能を使用できます。

項目	参照ページ
割り込み入力	ユニット割り込み機能(6-4 ページ)
インプットキャプチャ	インプットキャプチャ機能(6-9 ページ)
バッファリング入力機能	バッファリング入力機能(6-10 ページ)
PWM出力機能	PWM出力機能(6-19 ページ)

汎用入出力

割り込み入力、インプットキャプチャ、バッファリング入力、汎用入力は同時に使用可能です。
PWM出力と汎用出力は同じ端子で同時に使用できません。
各入出力の端子番号は「 結線・内部回路図」(2-16ページ)を参照してください。
汎用入出力として使用する場合の便利な使用方法は以下のとおりです。

■ 入出力リフレッシュについて

通常は、プログラムの実行中に入出力信号の状態を更新(リフレッシュ)するのはスキャンタイムに1回(END処理時)のみです。プログラムの任意の位置で入出力信号を更新(ダイレクト処理)するには以下の方法があります。

ダイレクト入出力(DR)

命令語のオペランドに指定するリレー番号を「R」ではなく「DR」と指定します。

入出力リフレッシュ命令(RFSX、RFSY)

入出力の状態を複数ビット分まとめてリフレッシュします。

『KV-8000/7000/5000/3000/1000シリーズ ・KV Nanoシリーズ命令語リファレンスマニュアル』「RFSX/RFSY命令」

■ 入力レンジ設定

入力レンジ設定を使用して、IN100～IN115のレンジを5V/24Vか選択できます。
入力レンジ設定はユニットエディタから設定します。

「ユニットエディタ」(6-3ページ)

IN100～IN115の入力レンジが一括で切り換わります。

ポイント **IN000～IN015は、入力レンジを切り換えることはできません。**

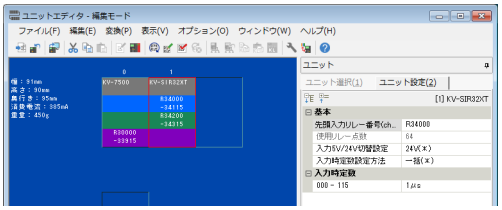
■ 入力時定数設定

入力時定数切換機能は、入力接点から信号として確実に取り込めるパルス幅を入力時定数として設定できる機能です。入力時定数を小さくすれば、より短い幅のパルスの入力が可能で、逆に大きくすれば、簡易ノイズフィルタとして利用できます。
チャタリングの発生しない無接点の信号を接続する場合や、エンコーダ等の高速信号を取り込む際には、入力時定数を小さくすることで高速の信号を扱うことができます。
入力時定数は、ユニットエディタから設定できます。1μs～50msの間で設定でき、初期値は10μsです。

『KV STUDIO　ユーザーズマニュアル』「3章　ユニットエディタ」

● ユニットエディタ

《KV-SIR32XT》の入力リレーの入力時定数、IN100～IN115のレンジ（5V/24V）は、ユニットエディタから変更できます。



■ 入力過電圧保護

5V入力モード使用時、定格（入力最大電圧6V）を超える電圧が印加された場合、過電圧保護が動作します。
過電圧保護が動作した場合、「入力過電圧エラー」（エラーコード101）が発生し、共通コモン内のすべての入力がOFF（入力表示灯もOFF）します。（共通コモン内とは、IN100～IN115のことを表します）

ポイント

過電圧保護を検出した場合、自動復帰はしません。要因を取り除き、PRG→RUNを実行すると復帰し、エラーもクリアされます。

■ 出力過電流保護

過電流を検知すると過電流保護が動作します。過電流保護が動作した場合、「出力過電流エラー」（202/203）が発生し、共通コモン内のすべての出力がOFFします。（共通コモンとは、OUT200～OUT215、OUT300～OUT315のことを表し、発生するエラー番号はコモンにより異なります。OUT200～OUT215の場合202、OUT300～OUT315の場合203）

ポイント

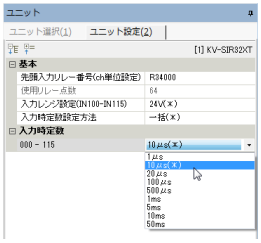
- **過電流保護を検出した場合、要因が取り除かれると自動復帰します。要因が取り除かれるまでの間ON/OFFを繰り返します。**
- **PWM出力機能使用時は、汎用出力の機能よりもPWM出力が優先されます。その機能が使用されているかは、バッファメモリの#400～#431で確認できます。**

6-2 ユニット割り込み機能 S-Unit機能

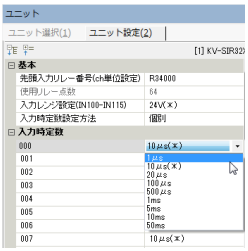
≪KV-SIR32XT≫の入力状態を確認してCPUに対してユニット割り込みを実行することができます。
ユニット割り込みの実行にはIEDGE(入力割り込み検出条件書き込み)命令を使用して、割り込み条件を設定し、その後INT(割り込み)命令を使用して割り込み機能を実行します。
ユニット割り込みには、立ち上がり、立ち下がり、立ち上がり/立ち下がり両方から選択することができます。

設定手順

1 ユニットエディタで割り込みに使用する入力端子の時定数を設定します。



参 考 個別に設定する場合



ポイント
- **入力時定数の初期値は10μsなので、入力がONしてから割り込みが動作するまでに10μs以上かかります。**
- **入力時定数は最短で1μsまで設定できます。**

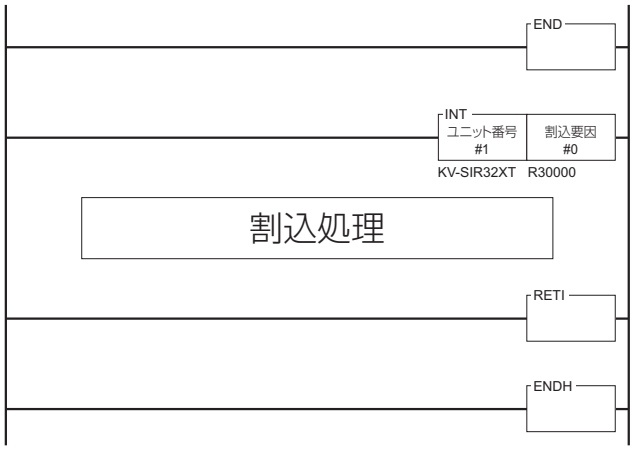
2 IEDGE(入力割り込み検出条件書き込み命令)を使用して検出条件を設定します。

割り込み検出条件には、立ち上がり、立ち下がり、立ち上がり/立ち下がり両方から選択できます。
下記はユニット番号1のIN000の立ち下がりを検出し割り込み機能を使用する場合の設定例です。

検出条件設定値	検出条件
0	立ち上がり
1	
2	立ち下がり
3	立ち上がり/立ち下がり両方

参 考 設定しない場合は立ち上がりで検出します。

3 END～ENDHの間に割り込み用のプロジェクトを作成します。



■ ユニット割込要因

INT命令で指定するユニット割込要因には以下の要因番号を使用します。

要因番号	割込要因
0	IN000
1	IN001
2	IN002
:	:
15	IN015
16	IN100
:	:
30	IN114
31	IN115

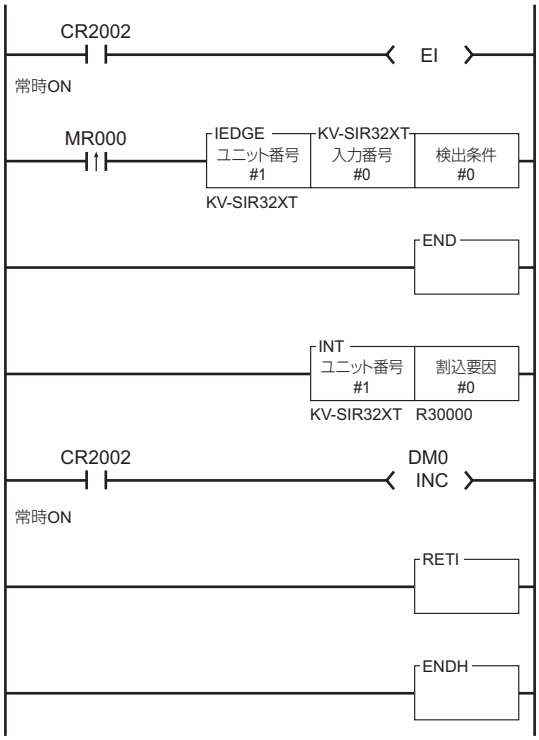
参 考

割込の検出条件は立ち上がり、立ち下がり、立ち上がり/立ち下がり両方から選択できます。設定するにはIEDGE命令を使用してください。

『KV-8000/7000/5000/3000/1000シリーズ･KV Nanoシリーズ命令語リファレンスマニュアル』「IEDGE命令」

使用参考例）

ユニット番号1台目のIN000がONした立ち上がりで割り込みを実行する場合の例を紹介します。



割込を許可します。

割込条件を設定します。

割込プログラムを作成し、プログラム内でDM0をカウントアップします。

< ニモニックリスト >

```
LD CR2002
EI
LDP MR000
IEDGE #1 #0 #0
END
INT #1 #0
LD CR2002
INC DM0
RETI
ENDH
```

6-3 ユニット間同期機能

S-Unit機能

CPUユニットに複数接続した≪KV-SIR32XT≫の動作開始タイミングを同時に実行することができます。
スキャンタイムやユニット間の制御周期のずれなどの影響を受けず制御できます。

ユニット間同期とは

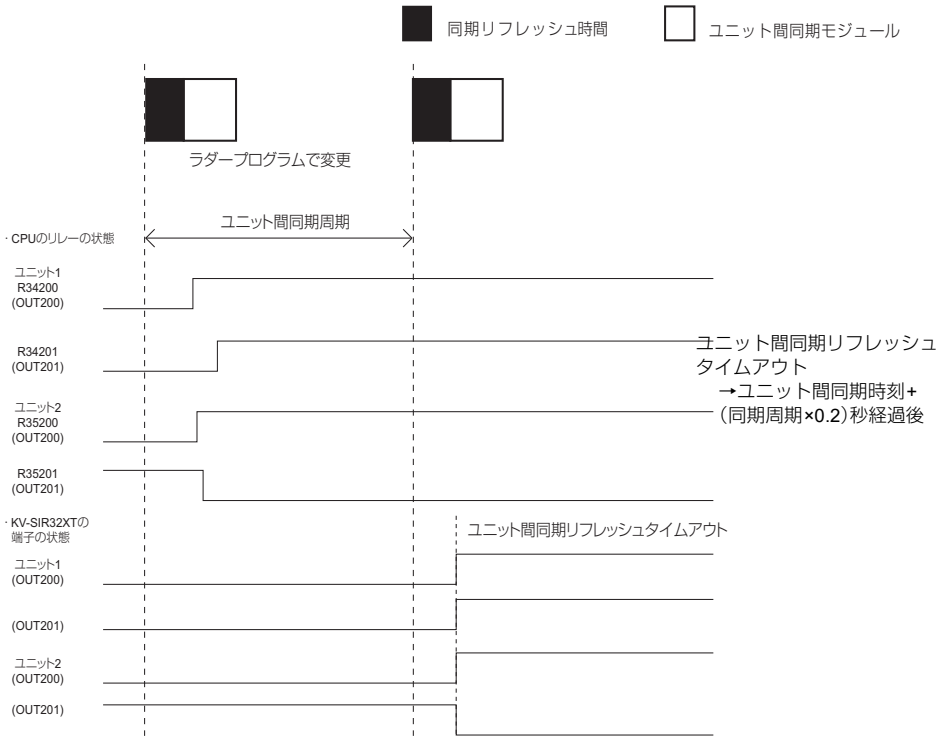
ユニット間同期機能は，複数のユニット間の信号での制御タイミングを合わせる機能です。
この機能を使用することで、複数のユニット間での同時サンプリング、同時位置決めなどが実現できます。

ユニット間同期の詳細については「5-10 ユニット間同期機能」（5-169ページ）を参照してください。

ユニット間同期機能実行時の動作概要

ユニット間同期機能実行時の動作を説明します。
リフレッシュタイミングをユニット間同期に設定した場合

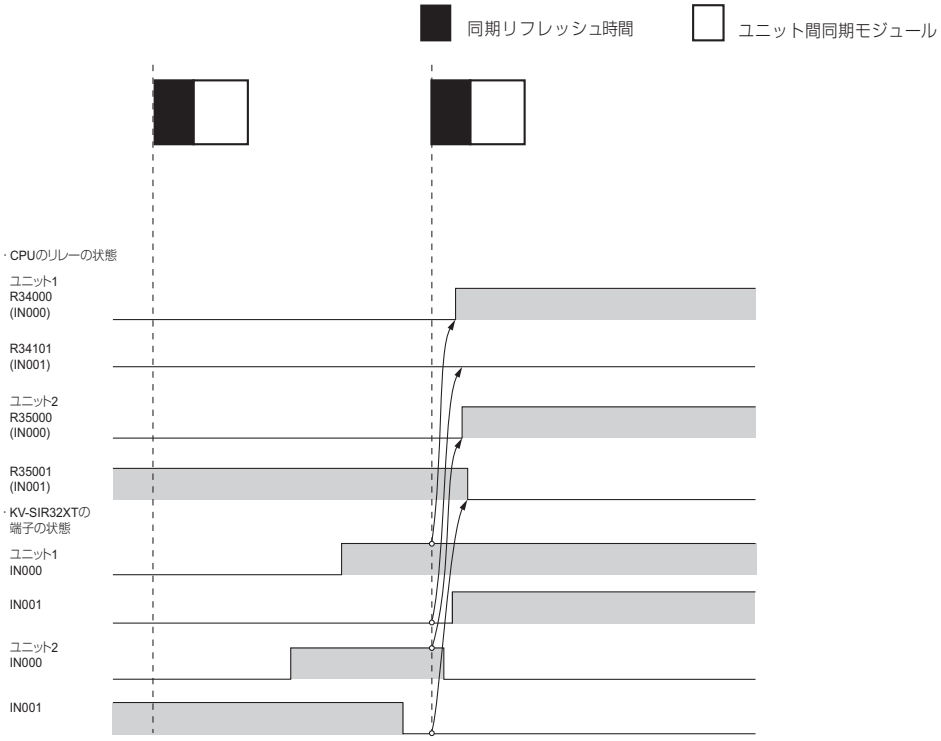
出力タイミング



ポイント

- ユニット間同期モジュール内で変更したリレーの状態は次回のユニット間同期の開始からユニット間同期リフレッシュタイムアウト後に出力端子に反映されます。
- ユニット間同期リフレッシュでは入力/出力の順にリフレッシュされるため、ユニット間同期リフレッシュタイミングで読み出したコンパレーター致や、同期接点出力などの信号をユニット間同期周期×0.2(ms)で複数のユニット、複数の端子に対して同時に出力できます。
- ユニット間同期モジュール内で、DRなどダイレクトリフレッシュを実行すると、即時出力に反映されます。
  ダイレクトリフレッシュを実行した場合、ユニット間で出力されるタイミングにはずれが発生することがあります。

入力タイミング



ポイント

- ユニット間同期リフレッシュ時に読み出されたデータは同期リフレッシュ実行直前の、同期開始時点での値になり、端子間、ユニット間で同じタイミングの値を読み出すことができます。
- ユニット間同期モジュール内でDRなどダイレクトリフレッシュ命令を使用すると、同期開始時のデータではなく、ダイレクトリフレッシュ命令実行時の値を読み出すことができます。

リフレッシュタイミングをEND処理に設定した場合

リフレッシュされるタイミングは、END処理、または、ダイレクトリフレッシュ命令実行時となり、端子間、ユニット間の同期がとれないことがあります。
ただし、入力に関しては、ユニット間同期周期ごとにバッファメモリの【#240　IN000~015　同期キャプチャ、#241　IN100～115　同期キャプチャ】に、ユニット間同期時の状態がラッチされるため、ユニット間同期モジュール内で読み出すことでユニット間同期時の状態を確認することができます。

ユニット間同期機能時のユニット仕様

項目		設定値
ユニット間同期周期		40.0μs～
リフレッシュデバイス	入力R	2ch（32点）
	入力DM	－
	出力R	2ch（32点）
	出力DM	－
	バッファメモリ	－

ポイント

ユニット間同期の周期は拡張ユニットによって異なります。設定できる最小の周期は、最も大きい制御周期をもつユニットの制御周期に従います。

6-4 インプットキャプチャ機能

≪KV-SIR32XT≫の入力ON/OFFのタイミングで≪KV-SIR32XT≫に内蔵されたフリーランカウンタ（32ビット0.1μs）の値をキャプチャします。インプットキャプチャの結果は常時更新され、バッファメモリに格納されるため、バッファメモリ命令と組み合わせて使用することで、入力のON時間を測定することもできます。インプットキャプチャ機能は32点全ての入力端子で使用できます。

バッファメモリアドレス	名称	属性 R:読取専用
0000～0001	IN000_立ち上がりインプットキャプチャ	R
0002～0003	IN000_立ち下がりインプットキャプチャ	R
0004～0005	IN001_立ち上がりインプットキャプチャ	R
:	:	:
N※1×4～N×4＋1	IN***_立ち上がりインプットキャプチャ	R
N※1×4＋2～N×4＋3	IN***_立ち下がりインプットキャプチャ	R
:	:	:
0124～0125	IN115_立ち上がりインプットキャプチャ	R
0126～0127	IN115_立ち下がりインプットキャプチャ	R
0128～0129	0.1μs　32ビット　フリーランカウンタ	R

※1　Nには0～31（0：IN000、1：IN001、～、15：IN015、16：IN100、17：IN101、～、31：IN115）が入ります。

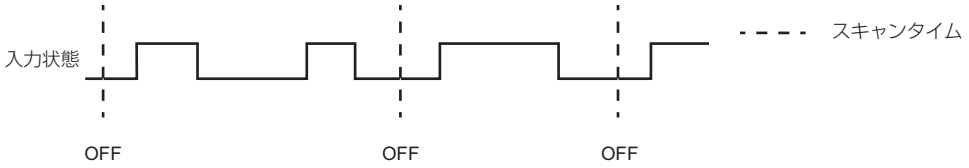
インプットキャプチャを使用したON時間測定の参考例については、「6-11 I/Oユニット参考プログラム例」を参照してください。

6-5 バッファリング入力機能

≪KV-SIR32XT≫の入力状態を最小1μsでサンプリングし、バッファリングする機能です。
バッファリングには2種類のモードがあり、連続バッファリング使用時は、バッファリングを行いながらデータを読み出すことにより、ユニット内のバッファリング用メモリのサイズを超える大量のデータ取得が可能です。トリガバッファリング使用時は、トリガが入力されるまでサンプリングを継続し、トリガが入力されたタイミングで前後合わせて設定数分のデータをサンプリングした後停止します。

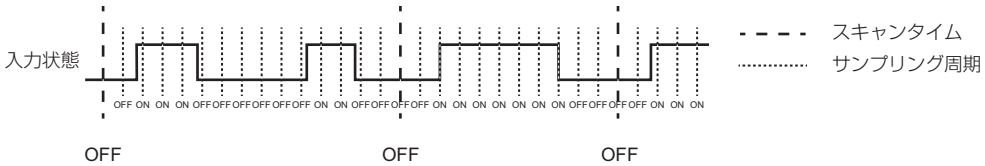
■ バッファリング動作概要

- スキャンごとに取得した場合



ON(OFF)時間がスキャンタイムより短い信号の場合、入力状態を正確にサンプリングできません。

- バッファリングを実行した場合

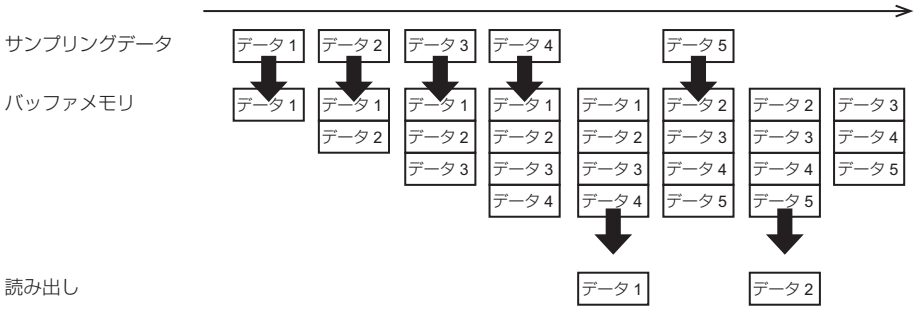


スキャンタイムより短い間隔でも入力状態を正確にサンプリングできます。

■ 動作モード

● 連続バッファリング

サンプリングされたデータが順番に格納されます。U_RDBF命令を使用して、バッファリングデータの読み出しを実行すると、データをCPUのデバイスに読み出し、バッファメモリのデータは削除されます。バッファメモリのデータが4096個に到達すると、バッファリングオーバーフローになり、バッファリングを停止します。

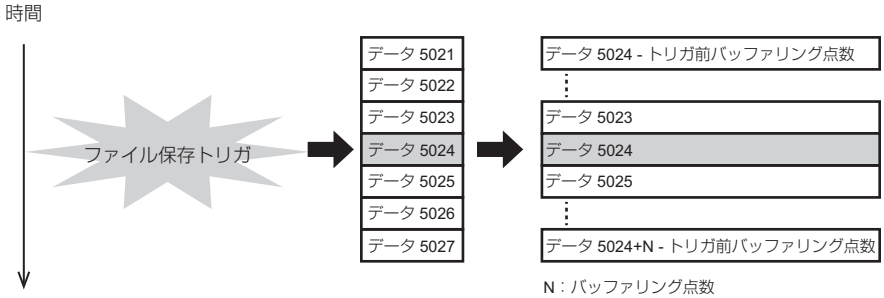


ポイント

- **一度読み出したデータは削除されます。**
- **読み出し実行時は古いデータから順番に読み出します。途中のデータを読み出すことはできません。**

● トリガバッファリング

連続バッファリングと同様にデバイスの状態や値を内部バッファ上に記憶します。ファイル保存のタイミングで、その前後の指定したデータ数のみを抽出して、バッファメモリに記録します。



ポイント　**トリガバッファリング実行中は、バッファオーバーフローは発生しません。**

● データの格納方法

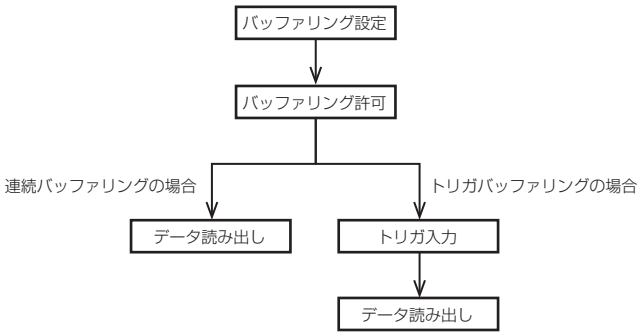
サンプリングされたデータは、16ビットごとにバッファメモリに格納されます。
ビット0～ビット15にIN000～IN015（またはIN100～IN115）の順にデータが格納されます。

	データ（10進で表記した場合）
1番古いデータ	0010 0101 1101 0001（9681）
2番目に古いデータ	0010 0101 1101 0000（9680）
：	：
N番目に古いデータ	0011 0001 1010 0101（12709）
：	：
4095番目に古いデータ	

ポイント　**サンプリング時の16点の入力情報を一つのバッファメモリアドレスに格納します。各点の時系列のデータを確認するには、読み出したデバイスの特定のビット位置の情報を時系列で確認する必要があります。**

● 使用方法

設定はすべてラダープログラムでおこないます。



● 使用するバッファメモリアドレス

Low : IN000～IN015の設定
Hi : IN100～IN115の設定

バッファメモリアドレス	名称	属性 R : 読取専用 R/W : 読書可能	概要
#512	Low_バッファリングモード設定	R/W	0 : 連続 1 : トリガ（初期値 : 0）
#513	Low_バッファリング点数設定	R/W	1～4095（初期値 : 100）
#514	Low_バッファリング周期設定	R/W	1～65535（μs）（初期値 : 10）
#515	（システム予約）	－	
#516	Low_トリガ条件設定	R/W	0 : トリガレジスタに1書き込み 1 : 指定入力ONエッジ 2 : 指定入力OFFエッジ 3 : ユニット間同期割り込み （初期値 : 0）
#517	（システム予約）	－	
#518	Low_トリガ端子設定	R/W	0～31（初期値 : 0）
#519	Low_トリガ前バッファリング点数設定	R/W	0～4095（初期値 : 0）
#520	Low_識別番号選択設定	R/W	0 : なし（初期値） 1 : 通し番号 2 : ユニット間同期周期別経過時間 3 : ユニット間同期周期別通し番号
#521～#527	（システム予約）	－	
#528	Low_バッファリングモード設定モニタ	R	0 : 連続 1 : トリガ（初期値 : 0）
#529	Low_バッファリング点数設定モニタ	R	1～4095（初期値 : 100）
#530	Low_バッファリング周期設定モニタ	R	1～65535（μs）（初期値 : 10）
#531	（システム予約）	－	
#532	Low_トリガ条件設定モニタ	R	0 : トリガレジスタに1書き込み 1 : 指定入力ONエッジ 2 : 指定入力OFFエッジ 3 : ユニット間同期割り込み （初期値 : 0）
#533	（システム予約）	－	
#534	Low_トリガ端子設定モニタ	R	0～31（初期値 : 0）
#535	Low_トリガ前バッファリング点数設定モニタ	R	0～4095（初期値 : 0）
#536	Low_識別番号選択設定モニタ	R	0 : なし 1 : 通し番号 2:ユニット間同期周期別経過時間 3:ユニット間同期周期別通し番号
#537	（システム予約）	－	
#538	Low_バッファリング状態モニタ	R	bit0 : バッファリング中 bit1 : トリガ待ち中 bit2 : バッファリング完了
#539	Low_現在バッファリング点数モニタ	R	0～4095
#540	Low_バッファリングオーバーフロー	R	0 : 正常 1 : オーバーフロー発生
#541	Low_トリガ前バッファリング点数モニタ	R	0～4095
#542～#543	（システム予約）	－	
#544～#575	（Hiバッファリング設定 #512～#543と相当）	R/W	
#576～#639	（システム予約）	－	
#640	Low_バッファリング許可	R/W	0 : 禁止 1 : 許可
#641	Hi_バッファリング許可	R/W	0 : 禁止 1 : 許可
#642～#655	（システム予約）	－	
#656	Low_トリガ	R/W	1 : トリガ入力 （入力後自動的に0に更新されます）
#657	Hi_トリガ	R/W	1 : トリガ入力 （入力後自動的に0に更新されます）

■ デバイス（Low の場合: IN000～IN015使用時、Hiの場合 : IN100～IN115使用時）

- **バッファリングモード設定（Lowの場合：#512、Hiの場合：#544）**

  バッファリングの実行モードを、連続バッファリング、トリガバッファリングから選択します。

  設定値　0： 連続バッファリング（初期値）

  　　1： トリガバッファリング

  バッファリング許可に1が書き込まれたタイミングの設定値が動作に反映されます。

- **バッファリング点数設定（Lowの場合：#513、Hiの場合：#545）**

  バッファリング実行時のサンプリング点数を設定します。連続バッファリングモードの場合、バッファメモリにサンプリングされたデータ数が設定数に到達すると、バッファリング完了ビットがONします。バッファリングは継続します。

  設定値　1～4095（初期値:100）

  トリガバッファリングモードの場合、トリガ前後のサンプリング点数がバッファリング点数設定分の点数に到達するとバッファリングを停止します。

  バッファリング許可に1が書き込まれたタイミングの設定値が動作に反映されます。

- **バッファリング周期設定（Lowの場合：#514、Hiの場合：#546）**

  バッファリングのサンプリング周期を設定します。

  設定値　1～ 65535（μs）（初期値:10）

  バッファリング許可に1が書き込まれたタイミングの設定値が動作に反映されます。

- **トリガ条件設定（Lowの場合：#516、Hiの場合：#548）**

  トリガバッファリングモードのトリガ条件を設定します。

  設定値　0： トリガ（Lowの場合：#656、Hiの場合：#657）に1が書き込まれた時（初期値）

  　　1： トリガ端子設定（Lowの場合：#518、Hiの場合：#550）で設定した入力のON時

  　　2： トリガ端子設定（Lowの場合：#518、Hiの場合：#550）で設定した入力がOFF時

  バッファリング許可に1が書き込まれたタイミングの設定値が動作に反映されます。

- **トリガ端子設定（Lowの場合：#518、Hiの場合：#550）**

  トリガ条件設定に1～2を設定した場合のトリガに使用する入力端子を設定します。

  設定値　0～31（IN000～IN115）（初期値:0（IN000））

  バッファリング許可に1が書き込まれたタイミングの設定値が動作に反映されます。

- **トリガ前バッファリング点数設定（Lowの場合：#519、Hiの場合：#551）**

  トリガバッファリングモード使用時にトリガ前のサンプリング点数を設定します。

  設定値　0～4095（初期値:0）

  バッファリング許可に1が書き込まれたタイミングの設定値が動作に反映されます。

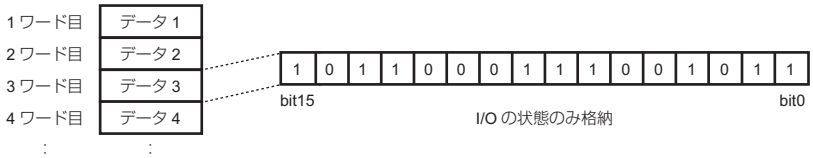
- **識別番号選択設定**

データに付与する識別番号を設定します。

設定値　0：なし（初期値）
　　1：通し番号
　　2:　ユニット間同期周期別経過時間
　　3:　ユニット間同期周期別通し番号

0（なし）に設定した場合

データに識別番号は付与されません。



1（通し番号）に設定した場合

16ビットのデータに通し番号と、トリガ認識の状態を付与し、32ビットデータとして格納します。

※　Trg ビット：トリガ認識後最初のデータの場合 1、その他の場合 0 が格納されます。
※　通し番号：　0 ～ 32767 の範囲でサンプリング順に番号が割り振られます。（32767 の次は 0 が格納されます）

2（ユニット間同期周期別経過時間）に設定した場合

16ビットのデータにユニット間同期周期の状況と、同期周期内の経過時間（μs）を付与し、32ビットデータとして格納します。

※　Trg ビット：トリガ認識後最初のデータの場合 1、その他の場合 0 が格納されます。
※　状態ビット：同期周期毎に 0,1 が切り換わります。
同じタイミングでは、拡張ユニット間で 0,1 の状態は共通になります。
※　経過時間：　ユニット間同期の実行タイミングからの経過時間 (μs) が、0 ～ 16383 の範囲で格納されます。

3(ユニット間同期周期別通し番号)に設定した場合

16ビットのデータにユニット間同期周期の状況と、同期周期内の通し番号を付与し、32ビットデータとして格納します。

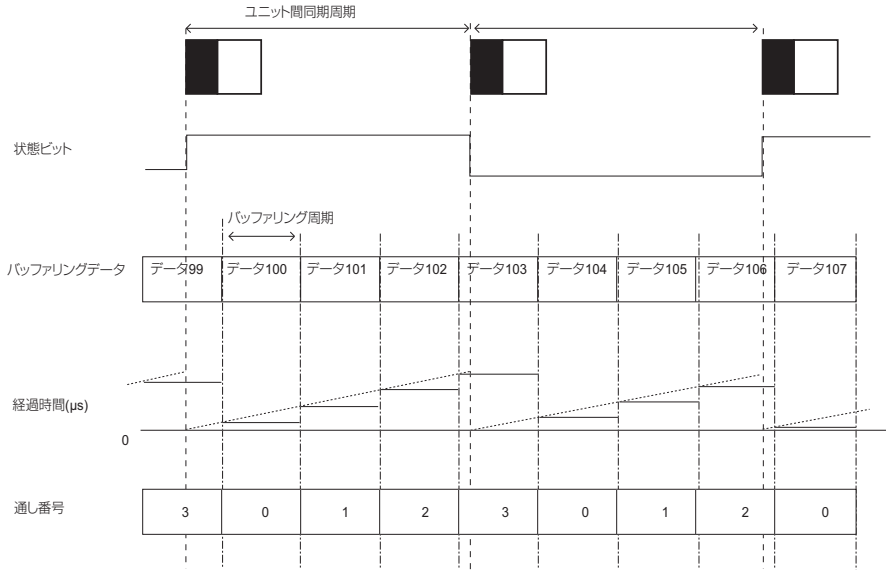
※ Trg ビット：トリガ認識後最初のデータの場合 1、その他の場合 0 が格納されます。

※ 状態ビット：同期周期毎に 0,1 が切り換わります。同じタイミングでは、拡張ユニット間で 0,1 の状態は共通になります。

※ 同期周期別通し番号：ユニット間同期の実行タイミングからの通し番号 (0~16383) が格納されます。

参 考

ユニット間同期周期別経過時間/通し番号



- **各種モニタ(バッファリングモード設定モニタ/バッファリング点数設定モニタ/バッファリング周期設定モニタ/トリガ条件設定モニタ/トリガ端子設定モニタ/トリガ前バッファリング点数設定モニタ/識別番号選択設定モニタ)**
  (Lowの場合:#528～#536、Hiの場合:#560～#568)
  現在実行中のバッファリングに使用されている設定値をモニタします。

参 考

バッファリング許可に1が書き込まれたタイミングの設定値が動作に反映されます。
1が書き込まれた後でパラメータが変更されている場合、設定値とモニタの値が異なる場合があります。
実際の動作に反映されている設定値は、設定モニタで確認してください。

- **バッファリング状態モニタ（Lowの場合：#538、Hiの場合：#570）**

  バッファリングの状態をモニタします。

  bit0 : バッファリング中

  bit1 : トリガ待ち中

  bit2 : バッファリング完了

  バッファリング許可に0が書き込まれている間は常に0が格納されます。トリガ前バッファリング点数をサンプリングするまで、またはトリガ入力後バッファリング点数に到達するまでの間1が格納されます。連続モードでバッファリング中、トリガモードでトリガ待ち状態の場合3が格納されます。トリガモードでバッファリング点数に到達した場合4が格納されます。連続モードでバッファリング点数設定での点数分以上、ユニット側にバッファリングデータがある場合7が格納されます。

  参 考　バッファリング許可に1が書き込まれたタイミングの設定値が動作に反映されます。
  1が書き込まれた後でパラメータが変更されている場合、設定値とモニタの値が異なる場合があります。

- **現在バッファリング点数モニタ（Lowの場合：#539、Hiの場合：#571）**

  バッファメモリ内のデータ点数をモニタします。

- **バッファリングオーバーフローモニタ（Lowの場合：#540、Hiの場合：#572）**

  連続バッファリングモードで、バッファリングの状態をモニタします。

  0 : 正常動作

  1 : オーバーフロー発生

  参 考　バッファリング許可に0が書き込まれたタイミングで0にクリアされます。

- **トリガ前バッファリング点数モニタ（Lowの場合：#541、Hiの場合：#573）**

  トリガバッファリングモードで、トリガ前バッファリング点数をモニタします。

  参 考　トリガバッファリングモードで、バッファリングを開始してから、トリガ前バッファリング点数設定分のバッファリングを行う前にトリガが入力された場合、トリガ前バッファリング点数は設定値よりも少ない状態でバッファリング完了します。

- **バッファリング許可（Lowの場合：#640、Hiの場合：#641）**

  バッファリングの実行を開始します。

  1を書き込む（UWRIT命令を実行する）とデータ領域のデータをクリアして、設定データを読み出してバッファリングを開始します。

  参 考　事前に一度0を格納してから1を書き込んで実行してください。

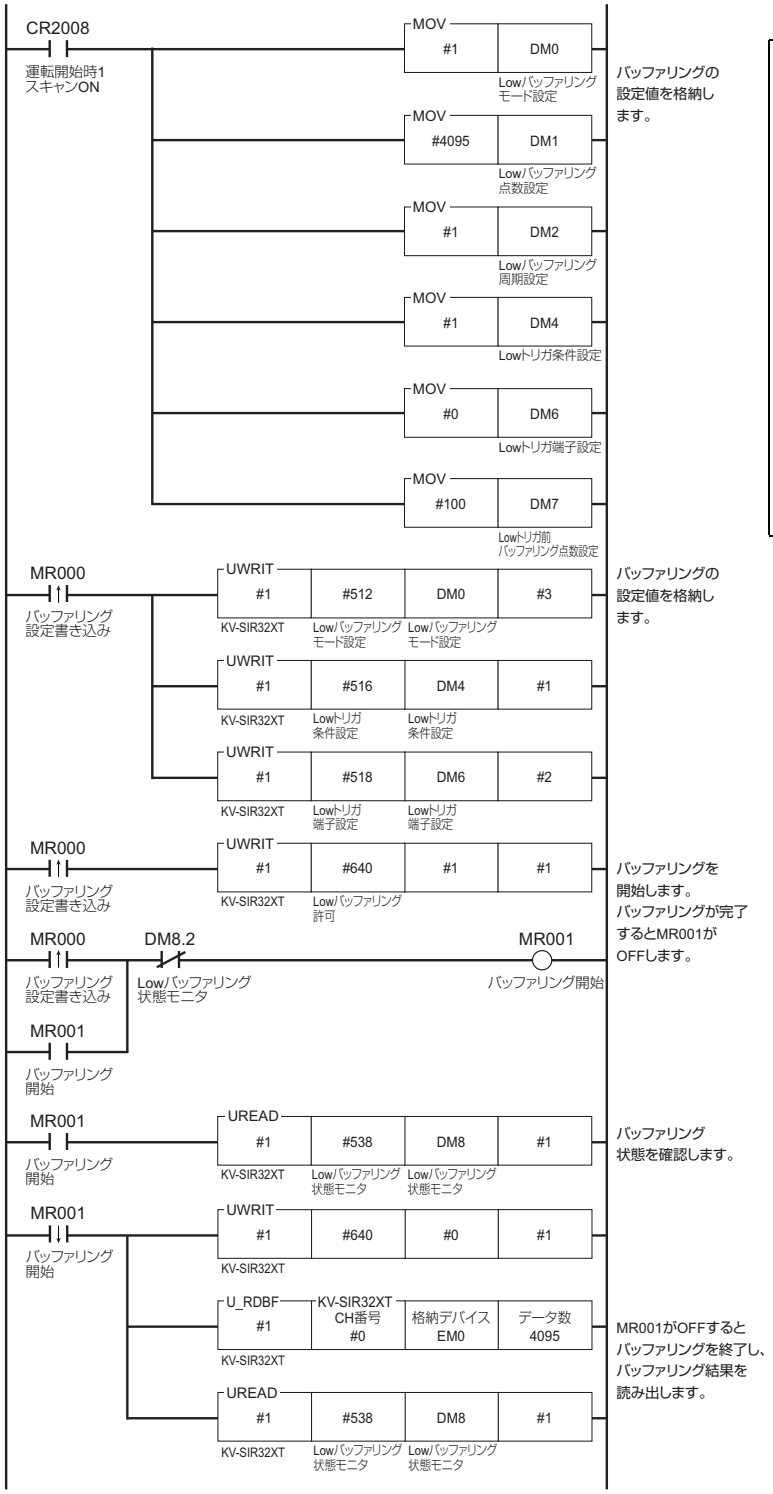
- **トリガ（Lowの場合：#656、Hiの場合：#657）**

  トリガバッファリングモードで、トリガ条件設定で0（トリガに1書き込み）を選択した場合、1を書き込む（UWRIT命令を実行する）と指定された回数分サンプリングして終了します。

  参 考　事前に0を書き込む必要はありません。

■ 参考プログラム

IN000～IN015の入力状態を1μsごとに4095点バッファリングします。
IN000のONをトリガとして、トリガ前100点、トリガ後3995点のバッファリングを実行します。

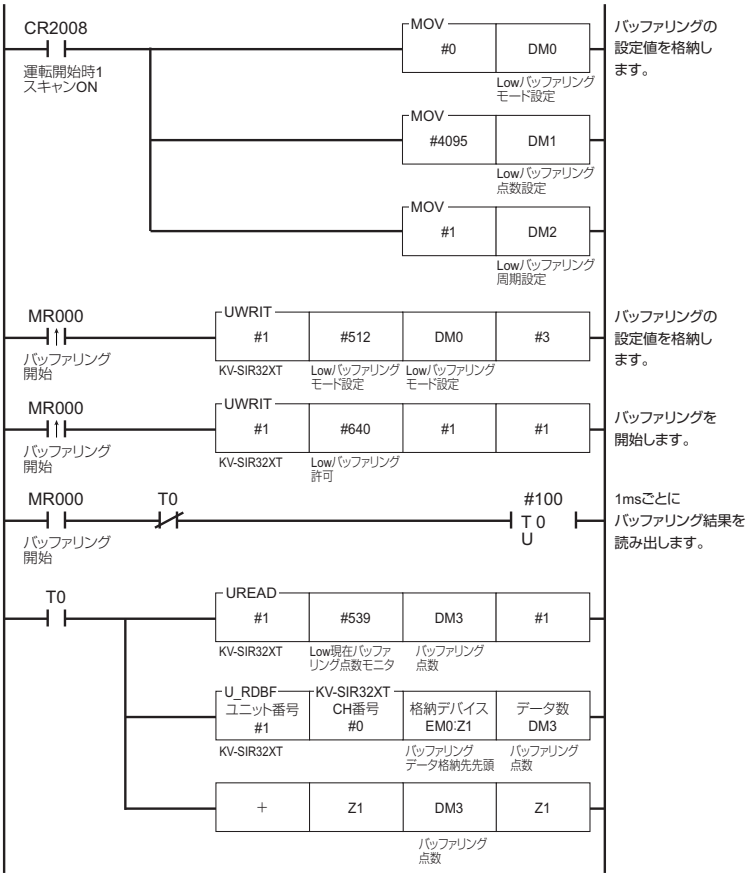


< ニモニックリスト >

```
LD CR2008
MOV #1 DM0
MOV #4095 DM1
MOV #1 DM2
MOV #1 DM4
MOV #0 DM6
MOV #100 DM7
LDP MR000
UWRIT #1 #512 DM0 #3
UWRIT #1 #516 DM4 #1
UWRIT #1 #518 DM6 #2
LDP MR000
UWRIT #1 #640 #1 #1
LDP MR000
OR MR001
ANB DM8.2
OUT MR001
LD MR001
UREAD #1 #538 DM8 #1
LDF MR001
UWRIT #1 #640 #0 #1
U_RDBF #1 #0 EM0 #4095
UREAD #1 #538 DM8 #1
```

■ 参考プログラム

IN000～IN015の入力状態を1μsごとに連続バッファリングします。バッファリングしたデータを、EM0を先頭にして1msごとに読み出します。

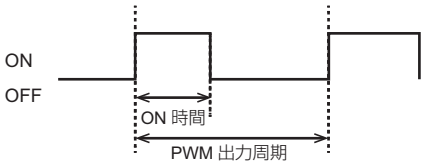


< ニモニックリスト >

```
LD CR2008
MOV #0 DM0
MOV #4095 DM1
MOV #1 DM2
LDP MR000
UWRIT #1 #512 DM0 #3
LDP MR000
UWRIT #1 #640 #1 #1
LD MR000
ANB T0
TMU #0 #100
LD T0
UREAD #1 #539 DM3 #1
U_RDBF #1 #0 EM0:Z1 DM3
CAL+ Z1 DM3 Z1
```

6-6 PWM出力機能

≪KV-SIR32XT≫の出力端子を使用して最大100kHz、0.1μs単位でパルス幅を指定してPWM出力する機能です。
PWM出力の周期と、ON時間を設定してパルス幅を調整して出力します。



ポイント

- **PWMは最大で8点まで同時に使用でき、個別に設定できます。各設定の中で出力する端子・周期・パルス数などを設定して使用します。**
- **1台のKV-8000シリーズに最大16台のKV-SIR32XTを接続可能なので、最大128点のPWM出力機能を使用できます。**

■ 動作モード

● パルス数指定モード

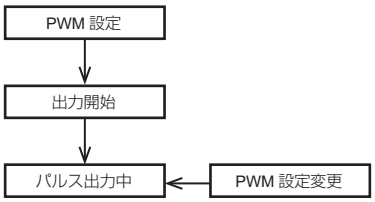
PWM制御して出力するパルス数を指定してパルス出力できます。

● 連続パルス出力モード

停止処理が実行されるまで連続してパルスを出力し続けます。

■ 使用方法

設定はすべてラダープログラムでおこないます。



参 考　パルス出力中に設定を変更すると1×PWM周期以内に出力する波形を変更できます。

● 使用するバッファメモリアドレス

バッファメモリアドレス	名称	属性 R：読出専用 R/W：読書可能	概要
#256	PWM0 出力許可	R/W	0：停止 1：PWM出力 （初期値：0）
#257	PWM0 出力端子	R/W	出力端子選択 （0～31（OUT 200～OUT315））
#258～#259	PWM0 周期	R/W	32ビット、0.1μs単位 0.1～429496729.5（μs）
#260～#261	PWM0 ON時間	R/W	32ビット、0.1μs単位 0.0～PWM0周期（μs）
#262～#263	PWM0 出力パルス数	R/W	32ビット、0～4294967295（PLS）
#264～#271	PWM1 設定	－	#256～#263相当
#272～#279	PWM2 設定	－	
#280～#287	PWM3 設定	－	
#288～#295	PWM4 設定	－	
#296～#303	PWM5 設定	－	
#304～#311	PWM6 設定	－	
#312～#319	PWM7 設定	－	
#320	PWM0 出力状態モニタ	R	bit0：出力許可状態モニタ bit1：出力完了 bit2：出力競合
#321	PWM0 出力端子モニタ	R	出力端子選択 （0～31（OUT200～OUT315））
#322～#323	PWM0 出力パルス数モニタ	R	32ビット、0～4294967295（PLS）
#328～#335	PWM1 出力状態モニタ	R	#320～#327相当
#336～#343	PWM2 出力状態モニタ	R	
#344～#351	PWM3 出力状態モニタ	R	
#352～#359	PWM4 出力状態モニタ	R	
#360～#367	PWM5 出力状態モニタ	R	
#368～#375	PWM6 出力状態モニタ	R	
#376～#383	PWM7 出力状態モニタ	R	
#400	OUT200機能モニタ	R	bit0：汎用出力中 bit1：PWM出力中
#401	OUT201機能モニタ	R	
:			
#415	OUT215機能モニタ	R	
#416	OUT300機能モニタ	R	
:			
#431	OUT315機能モニタ	R	
#432	OUT200～215モニタ	R	各端子の出力状態
#433	OUT300～315モニタ	R	

■ デバイス

- **PWMn(n:0～7)出力許可**
  PWM出力を実行するかどうかを選択します。
  0 : 出力しない
  1 : PWM出力実行
  0ビット目をONするとPWM出力を開始します。その他のビットは使用しません。

- **PWMn(n:0～7)出力端子**
  PWM出力を実行する出力端子を選択します。
  0～31:OUT200～OUT315

- **PWMn(n:0～7)周期**
  PWM出力の周期を0.1μs～429496729.5μsの範囲で設定します。
  実際の回路の遅延時間を考慮して10.0μs以上の値に設定することを推奨します。
  出力中に変更した場合、PWM出力の周期が完了したタイミングで反映されます。

- **PWMn(n:0～7)ON時間**
  PWM出力のON時間を0.0μs～429496729.5μsの範囲で設定します。
  ON時間に0.0μsを設定した場合常時OFFとなります。
  出力中に変更した場合、PWM出力の周期が完了したタイミングで反映されます。

- **PWMn(n:0～7)出力パルス数**
  PWM出力の出力パルス数を1～4294967295パルスの範囲で設定します。
  0に設定した場合、出力許可がOFFになるまで継続してパルスを出力します。
  出力許可に1が書き込まれたタイミングの設定値が動作に反映されます。出力許可中は書き換えは可能ですが、反映はされません。

- **PWMn(n:0～7)出力状態モニタ**
  PWM出力状態を表示します。PWM出力許可に0を格納すると、ビット0～2はOFFします。
  ビット0 : ON : 出力許可状態、OFF : 出力禁止状態
  ビット1 : ON : 出力完了状態、OFF : 出力未完了状態
  ビット2 : ON : ほかの出力機能が実行中のため、出力できませんでした。
  OFF: ほかの出力機能は実行されていません。

- **PWMn(n:0～7)出力端子モニタ**
  PWMn(n:0～7)で設定している出力端子0～31:OUT200～OUT315を表示します。

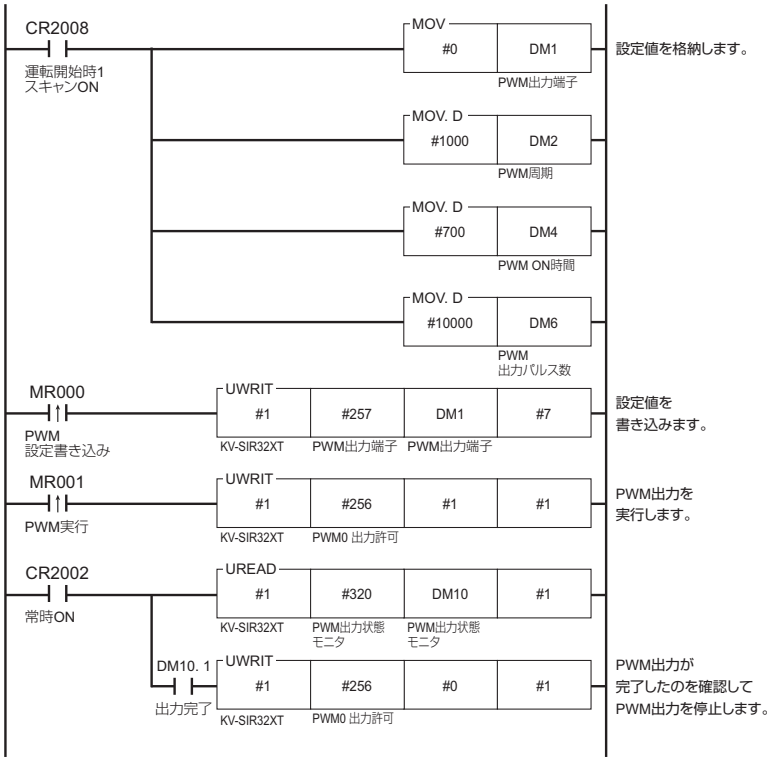
- **PWMn(n:0～7)出力パルス数モニタ**
  PWMn(n:0～7)で出力したパルス数を表示します。
  PWM出力許可に0を格納するとゼロクリアされます。

- **機能モニタn(n:200～315)**
  出力に使用している機能を表示します。
  ビット0 : ON : 汎用出力中(出力に割り付けられたリレーがON)
  ビット1 : ON : PWM出力中

- **出力モニタ**
  各端子の出力状態をモニタできます。

■ 参考プログラム

OUT200でPWM出力する場合のプログラム例です。
周期100.0μs（10kHz）、ON時間が70.0μsのパルスを10000パルス出力します。



< ニモニックリスト >

```
LD CR2008
MOV #0 DM1
MOV.D #1000 DM2
MOV.D #700 DM4
MOV.D #10000 DM6
LDP MR000
UWRIT #1 #257 DM1 #7
LDP MR001
UWRIT #1 #256 #1 #1
LD CR2002
MPS
UREAD #1 #320 DM10 #1
MPP
AND DM10.1
UWRIT #1 #256 #0 #1
```

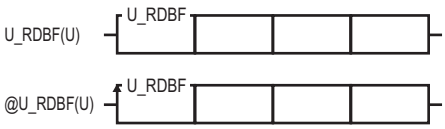
6-7 KV-SIR32XT用ユニット専用命令

ラダープログラム内で使用するKV-SIR32XT用ユニット専用命令について説明します。スクリプトプログラム内で使用する場合は、「6-8 KV-SIR32XT用ユニット専用関数」(6-30ページ)を参照してください。

KV-SIR32XT用ユニット専用命令一覧

機　能	命　令	動作概要	ページ
バッファリングデータ読み出し	U_RDBF	バッファリングされたI/Oの状態を一括でバッファメモリから読み出します。	6-24
入力過電圧保護機能無効化	U_DISOVP	入力の過電圧保護機能の有効/無効を設定します。	6-26
出力過電流保護機能無効化	U_DISOCP	出力の過電流保護機能の有効/無効を設定します。	6-28

U_RDBF
@U_RDBF

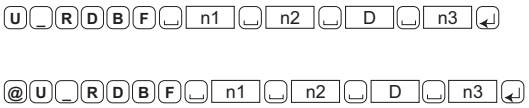


バッファリングデータ読み出し

バッファリングした入力データをバッファメモリから読み出します。

ラダープログラム

入力方法



オペランド	説明	占有サイズ
n1	ユニット番号を指定します。※1	―
n2	チャンネル番号、またはそれの格納先を指定します。※2	16ビット
D	読み出した値の格納先の先頭を指定します。	n3 ビット
n3	読み出すデータ数(0～8192)を指定します。※2	16ビット

※1　定数、および定数に設定した変数のみ指定可能です。

※2　ビットデータを指定した場合、連続する16ビットを占有します。ワードデータを指定した場合は1ワードを占有します。

動作説明

U_RDBF　実行条件がONの時、n1 で指定したユニット番号の n2 で指定したチャンネルのバッファリングデータを n3 ワード読み出して、D を先頭に格納します。

ポイント　**バッファリングの識別番号選択で0以外の値を設定した場合は、1つのデータサイズが2ワードになるため、読み出す点数は、2倍した値を入力してください。**

@U_RDBF　実行条件の立ち上がりで、1スキャンだけ実行します。

演算フラグ

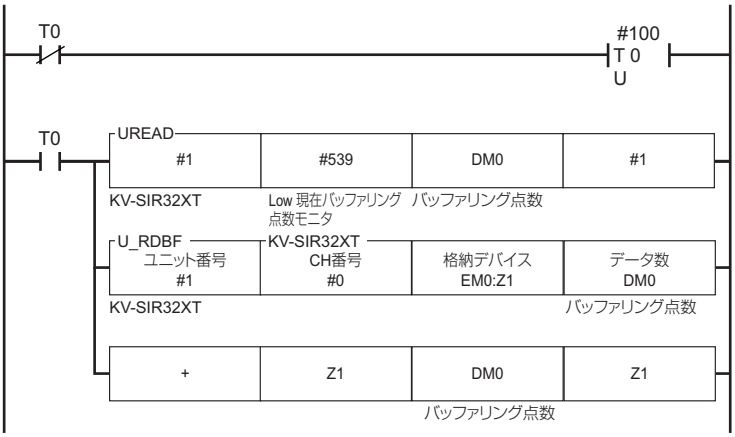
CR2009	変化なし
CR2010	変化なし
CR2011	変化なし
CR2012	以下の場合はON、それ以外はOFF。 • 間接指定、インデックス修飾の範囲が不適切な場合 • n1 で指定したユニット番号が64以上の場合 • n2 で指定したチャンネルが範囲外の場合 • D から、n3 ワードを確保できない場合 • n3 で指定したデータ数が8192よりも大きい場合 • n1 で指定したユニット番号のユニットがKV-SIR32XTではない場合

※ CR2012がONした場合、命令は実行されません。

KV-8000シリーズ CR2012がONした場合、CM5150～CM5176にエラーの詳細情報が格納されます。

サンプルプログラム

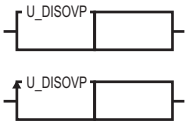
1msごとにバッファリングされた点数を読み出し、バッファリングされている点数分読み出します。



〈ニモニックリスト〉

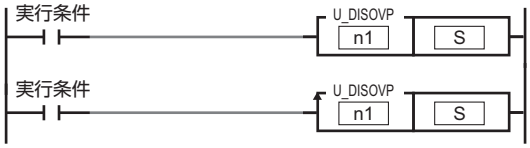
```
LDB T0
TMU #0 #100
LD T0
UREAD #1 #539 DM0 #1
U_RDBF #1 #0 EM0:Z1 DM0
CAL+ Z1 DM0 Z1
```

U_DISOVP
@U_DISOVP



入力過電圧保護機能設定　入力過電圧保護機能を有効にするか無効にするか設定します。

ラダープログラム



入力方法

オペランド	説明	占有サイズ
n1	ユニット番号を指定します。※1	―
S	書き込む値、または書き込む値の格納先を指定します。※2 （0：過電圧保護有効、1：過電圧保護無効）	16ビット

※1　定数、および定数に設定した変数のみ指定可能です。

※2　ビットデータを指定した場合、連続する16ビットを占有します。ワードデータを指定した場合は1ワードを占有します。

動作説明

U_DISOVP　実行条件がONの時、S に1を指定すると、n1 で指定したユニット番号のKV-SIR32XTの過電圧保護機能を無効化します。

ポイント
- **過電圧保護機能の有効、無効の設定は命令実行時に反映されます。**
- **電源投入時は0：過電圧保護有効　の状態で起動します。**

@U_DISOVP　実行条件の立ち上がりで、1スキャンだけ実行します。

演算フラグ

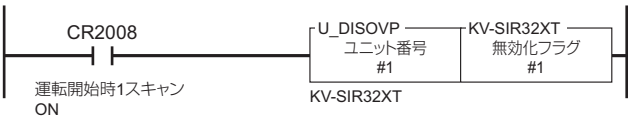
CR2009	変化なし
CR2010	変化なし
CR2011	変化なし
CR2012	以下の場合はON、それ以外はOFF。 • 間接指定、インデックス修飾の範囲が不適切な場合 • n1 で指定したユニット番号が64以上の場合 • n1 で指定したユニット番号のユニットがKV-SIR32XTではない場合

※ CR2012がONした場合、命令は実行されません。

KV-8000シリーズ CR2012がONした場合、CM5150～CM5176にエラーの詳細情報が格納されます。

サンプルプログラム

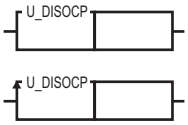
ユニット番号1のKV-SIR32XTの過電圧保護機能を無効化します。



〈ニモニックリスト〉

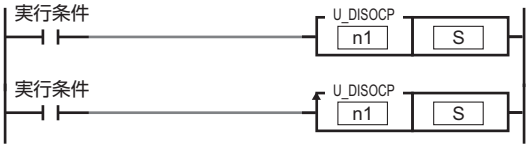
```
LD CR2008
U_DISOVP #1 #1
```

U_DISOCP
@U_DISOCP



出力過電流保護機能設定　出力過電流保護機能を有効にするか無効にするか設定します。

ラダープログラム



入力方法

オペランド	説明	占有サイズ
n1	ユニット番号を指定します。※1	―
S	書き込む値、または書き込む値の格納先を指定します。※2 （0：過電流保護有効、1：過電流保護無効）	16ビット

※1　定数、および定数に設定した変数のみ指定可能です。

※2　ビットデータを指定した場合、連続する16ビットを占有します。ワードデータを指定した場合は1ワードを占有します。

動作説明

U_DISOCP　実行条件がONの時、S に1を指定すると、n1 で指定したユニット番号のKV-SIR32XTの過電流保護機能を無効化します。

ポイント
- **過電流保護機能の有効、無効の設定は命令実行時に反映されます。**
- **電源投入時は0：過電流保護有効 の状態で起動します。**

@U_DISOCP　実行条件の立ち上がりで、1スキャンだけ実行します。

演算フラグ

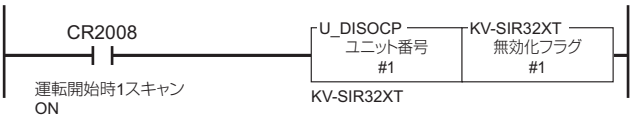
CR2009	変化なし
CR2010	変化なし
CR2011	変化なし
CR2012	以下の場合はON、それ以外はOFF。 • 間接指定、インデックス修飾の範囲が不適切な場合 • n1 で指定したユニット番号が64以上の場合 • n1 で指定したユニット番号のユニットがKV-SIR32XTではない場合

※ CR2012がONした場合、命令は実行されません。

KV-8000シリーズ CR2012がONした場合、CM5150～CM5176にエラーの詳細情報が格納されます。

サンプルプログラム

ユニット番号1のKV-SIR32XTの過電流保護機能を無効化します。



〈ニモニックリスト〉

```
LD CR2008
U_DISOCP #1 #1
```

6-8 KV-SIR32XT用ユニット専用関数

スクリプトプログラム内で使用するKV-SIR32XT用ユニット専用関数について説明します。ラダープログラム内で使用する場合は、「6-6 PWM出力機能」(6-19ページ)を参照してください。

KV-SIR32XT用ユニット専用関数一覧

■ ST言語関数

機　能	関　数	動作概要	ページ
バッファリングデータ読み出し	U_RDBF	バッファリングされたI/Oの状態を一括でバッファメモリから読み出します。	6-31
入力過電圧保護機能無効化	U_DISOVP	入力の過電圧保護機能の有効/無効を設定します。	6-32
出力過電流保護機能無効化	U_DISOCP	出力の過電流保護機能の有効/無効を設定します。	6-33

■ KVスクリプト関数

機　能	関　数	動作概要	ページ
バッファリングデータ読み出し	U_RDBF	バッファリングされたI/Oの状態を一括でバッファメモリから読み出します。	6-34
入力過電圧保護機能無効化	U_DISOVP	入力の過電圧保護機能の有効/無効を設定します。	6-35
出力過電流保護機能無効化	U_DISOCP	出力の過電流保護機能の有効/無効を設定します。	6-36

ST言語関数

U_RDBF　バッファリングデータ読み出し

バッファリングデータ読み出し

U_RDBF ([EN,] Unit, Ch, Dst, DataCount)

引数/戻り値	説明	認識する型								定数
		.U	.S	.D	.L	.F	.DF	.B	.T	
EN	実行条件。省略可能。省略時、常時(毎スキャン)実行。	—	—	—	—	—	—	.B	—	○
Unit	読み出すユニットのユニット番号を指定。※1 (0～48)	.U	.U	.U	.U	—	—	—	—	○
Ch	読み出すバッファリングデータのCH番号を指定。(0～1)※1	.U	.U	.U	.U	—	—	—	—	○
Dst	読み出したバッファリングデータを格納する対象データの先頭を指定。	.U	.U	.U	.U	—	—	—	—	—
DataCount	読み出すデータ点数を指定。(1～8192)	.U	.U	.U	.U	—	—	—	—	○
R	戻り値はなし。	—	—	—	—	—	—	—	—	—

動作説明

U_RDBF　実行条件がONの時、**Unit**で指定したユニット番号の**Ch**で指定したチャンネルのバッファリングデータを**DataCount**ワード読み出して、**Dst**を先頭とするデバイスに格納します。

詳細については、「U_RDBF命令」(6-24ページ)を参照してください。

● 書式例

U_RDBF(bEnable,#1,#0,auData,#4000);

ST言語関数

U_DISOVP 入力過電圧保護機能設定

入力過電圧保護機能設定

U_DISOVP ([EN,] Unit, Flag)

引数/戻り値	説明	認識する型								定数
		.U	.S	.D	.L	.F	.DF	.B	.T	
EN	実行条件。省略可能。省略時、常時(毎スキャン)実行。	—	—	—	—	—	—	.B	—	○
Unit	対象のユニット番号を指定します。※1	.U	.U	.U	.U	—	—	—	—	○
Flag	設定値を格納する対象データを指定。 (0:過電圧保護有効、1:過電圧保護無効)	.U	.U	.U	.U	—	—	—	—	○
R	戻り値はなし。	—	—	—	—	—	—	—	—	—

動作説明

U_DISOVP 実行条件がONの時、**Flag**に1を指定すると、**Unit**で指定したユニット番号のKV-SIR32XTの過電圧保護機能を無効化します。

詳細については、「U_DISOVP命令」(6-26ページ)を参照してください。

● 書式例

U_DISOVP(bEnable,#1,#1);

ST言語関数

U_DISOCP 出力過電流保護機能設定

出力過電流保護機能設定

U_DISOCP ([EN,] Unit, Flag)

引数/戻り値	説明	認識する型								定数
		.U	.S	.D	.L	.F	.DF	.B	.T	
EN	実行条件。省略可能。省略時、常時(毎スキャン)実行。	–	–	–	–	–	–	.B	–	○
Unit	対象のユニット番号を指定します。	.U	.U	.U	.U	–	–	–	–	○
Flag	設定値を格納する対象データを指定。 (0:過電流保護有効、1:過電流保護無効)	.U	.U	.U	.U	–	–	–	–	○
R	戻り値はなし。	–	–	–	–	–	–	–	–	–

動作説明

U_DISOCP 実行条件がONの時、**Flag**に1を指定すると、**Unit**で指定したユニット番号のKV-SIR32XTの過電流保護機能を無効化します。

詳細については、「U_DISOCP命令」(6-28ページ)を参照してください。

● 書式例

U_DISOCP(bEnable,#1,#1);

KVスクリプト関数

U_RDBF　バッファリングデータ読み出し

バッファリングデータ読出
U_RDBF （[実行条件]※1,ユニット番号,CH番号,格納デバイス,データ数）

引数/戻り値		説明	認識する型								定数	デバイス	式
			.U	.S	.D	.L	.F	.DF	.B	.T	#/$		
n1	ユニット番号※2	読み出すユニットのユニット番号を指定します。※2（0～48）	.U	.U	.U	U	–	–	–	–	○※2	–	–
n2	CH番号※2	読み出すバッファリングデータのCH番号を指定します。（0～1）	.U	.U	.U	U	–	–	–	–	○※2	○	○
D	読出データ格納デバイス	読み出したバッファリングデータを格納する先頭デバイスを指定します。	.U	.U	.U	U	–	–	–	–	–	○	○
n3	読み出すデータ数	読み出すデータ点数を指定します。（1～8192）	.U	.U	.U	U	–	–	–	–	○※2	○	○

※1　[　]は省略可能です（実行条件を省略した場合、常時（毎スキャン）実行します）。
※2　$（16進数指定）は使用できません。

動作説明

U_RDBF　実行条件がONの時、n1 で指定したユニット番号の n2 で指定したチャンネルのバッファリングデータを n3 ワード読み出して、D を先頭とするデバイスに格納します。

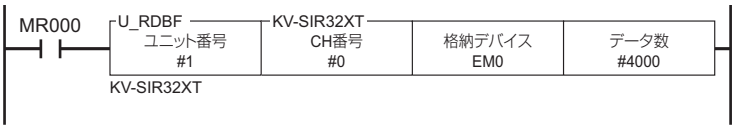
詳細については、「U_RDBF命令」（6-24ページ）を参照してください。

● 書式例

スクリプト記述　U_RDBF（MR0,#1,#0,EM0,#4000）

動作説明　MR0がONの時、1台目に接続されているKV-SIR32XTのバッファリングデータをEM0を先頭に4000ワード読み出します。

ラダー変換



KVスクリプト関数

U_DISOVP 入力過電圧保護機能設定

入力過電圧保護機能設定
U_DISOVP（[実行条件]※1,ユニット番号,設定値）

引数/戻り値	説明	認識する型								定数	デバイス	式
		.U	.S	.D	.L	.F	.DF	.B	.T	#/$		
n1 ユニット番号※2	ユニットのユニット番号を指定します。※2(0～48)	.U	.U	.U	U	–	–	–	–	○※2	–	–
S 設定値格納デバイス	書き込む値、または書き込む値の格納先を指定します。※2 (0:過電圧保護有効、1:過電圧保護無効)	.U	.U	.U	U	–	–	–	–	○※2	○	○

※1　[　]は省略可能です（実行条件を省略した場合、常時（毎スキャン）実行します）。
※2　$（16進数指定）は使用できません。

動作説明

U_DISOVP　実行条件がONの時、S に1を指定すると、n1 で指定したユニット番号のKV-SIR32XTの過電圧保護機能を無効化します。

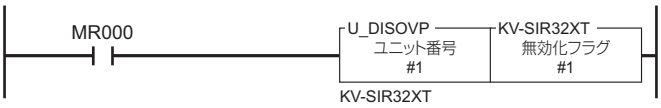
詳細については、「U_DISOVP命令」（6-26ページ）を参照してください。

● 書式例

スクリプト記述　U_DISOVP（MR0,#1,#1）

動作説明　MR0がONの時、1台目に接続されているKV-SIR32XTの入力過電圧保護機能を無効化します。

ラダー変換



KVスクリプト関数

U_DISOCP 出力過電流保護機能設定

出力過電流保護機能設定
U_DISOCP（[実行条件]※1,ユニット番号,設定値）

引数/戻り値	説明	認識する型								定数	デバイス	式
		.U	.S	.D	.L	.F	.DF	.B	.T	#/$		
n1 ユニット番号※2	ユニットのユニット番号を指定します。※2(0～48)	.U	.U	.U	U	–	–	–	–	○※2	–	–
S 設定値格納デバイス	書き込む値、または書き込む値の格納先を指定します。※2 (0:過電流保護有効、1:過電流保護無効)	.U	.U	.U	U	–	–	–	–	○※2	○	○

※1 [ ]は省略可能です（実行条件を省略した場合、常時（毎スキャン）実行します）。
※2 $（16進数指定）は使用できません。

動作説明

U_DISOCP 実行条件がONの時、S に1を指定すると、n1 で指定したユニット番号のKV-SIR32XTの過電流保護機能を無効化します。

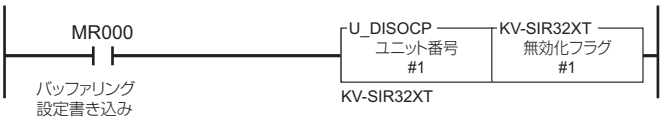
詳細については、「U_DISOCP命令」（6-28ページ）を参照してください。

● 書式例

スクリプト記述　U_DISOCP（MR0,#1,#1）

動作説明　MR0がONの時、1台目に接続されているKV-SIR32XTの出力過電流保護機能を無効化します。

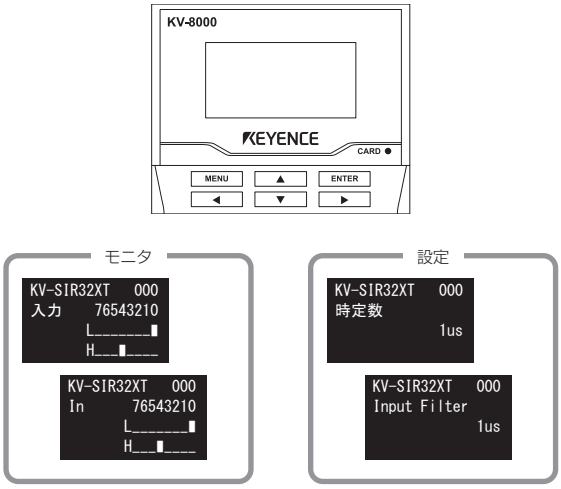
ラダー変換



6-9 アクセスウィンドウ

CPUユニットのアクセスウィンドウを使用して《KV-SIR32XT》の設定内容をモニタ、変更する手順について説明します。

アクセスウィンドウ



アクセスウィンドウでの設定変更について

一部の設定内容については、アクセスウィンドウで変更することができますが、基本的にはユニットエディタを使用するようにしてください。

参 考

アクセスウィンドウで設定変更する場合は、CPUユニットがPROGRAMモードの状態で実行します。
実際の設定変更（変更内容が有効になる）は、CPUユニットのモードをRUNモードに切り換えたときです。

メニュー構成

アクセスウィンドウのメニュー構成は以下のようになっています。

- モニタ
  - 入力IN000～IN015
  - 入力IN100～IN115
  - 出力OUT200～OUT215
  - 出力OUT300～OUT315
- 設定 時定数000
- 設定 時定数100
- 設定　入力レンジ

アクセスウィンドウを操作する

アクセスウィンドウは、CPUユニットにある設定操作キーで操作します。

設定操作キー	通　常	数値変更時
MENU	メニュー表示	メニュー表示
ENTER	1秒以上押しつづけると、設定変更を実行します。	1秒以上押しつづけると、設定変更を実行します。
▲	ページを切り換えます。設定変更時は選択肢を変更します。	設定値から変化量を加算します。
▼	ページを切り換えます。設定変更時は選択肢を変更します。	設定値から変化量を減算します。
◀	メニュー表示	変化量の桁を変更します。
▶	使用しません。	

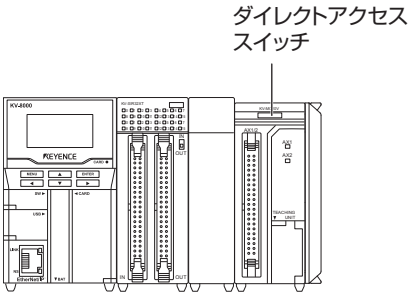
※設定変更はCPUユニットがPROGRAMモード時のみ可能です。

メニュー画面への移行

メニュー画面に移行する手順について説明します。

1 設定したいKV-SIR32XTのダイレクトアクセススイッチを押します。

ダイレクトアクセススイッチが点滅状態になり、選択したKV-SIR32XTのIN000～IN015のモニタ画面がアクセスウィンドウに表示されます。



2 CPUユニットの「MENU」キーを押します。

メニュー画面が表示されます。

メニュー画面は「▲」「▼」キーでスクロールします。

ポイント

CPUユニットのメニュー画面に戻りたい場合は、もう一度「MENU」キーを押すか、点滅しているダイレクトアクセススイッチを押してください。

「5-9 アクセスウィンドウ」

設定画面

CPUユニットのアクセスウィンドウを使用して《KV-SIR32XT》の設定内容を変更する手順について説明します。

ポイント **設定変更はCPUユニットがPRGモード時のみ可能です。**

設定変更方法

《KV-SIR32XT》の設定内容の一部をアクセスウィンドウで変更できます。
入力時定数を変更する手順について説明します。

1 メニュー画面から設定する項目を「▲」「▼」キーで選択し、「ENTER」キーを押します。

選択した設定画面が表示されます。

2 「ENTER」キーを1秒以上押し続けます。

設定値が点滅します。

3 「▲」「▼」キーで選択し、「ENTER」キーを1秒以上押し続けます。

参考 設定変更できる項目と設定範囲は次のようになります。

項　目	設定範囲
入力時定数000	1μs/10μs/20μs/100μs/500μs/1ms/5ms/10ms/50ms
入力時定数100	
入力レンジ（100～115）	5V、24V

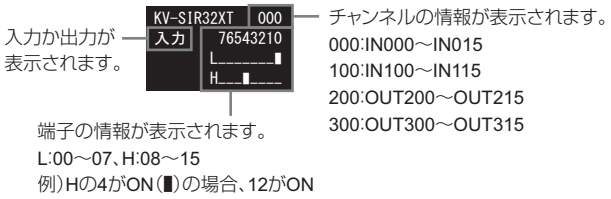
モニタ画面

CPUユニットのアクセスウィンドウを使用して《KV-SIR32XT》のI/Oの状態をモニタする手順について説明します。

モニタ方法

1 メニュー画面からモニタ画面中で「▲」「▼」キーを使用して設定/モニタするチャンネルを選択し、「ENTER」キーを押します。

選択したチャンネルのI/Oの状態がモニタ画面に表示されます。



ポイント

モニタ画面からは状態の変更はできません。
出力状態を変更する必要がある場合は、ユニットテスト機能を使用してください。

参 考 KV-SIR32XTのLED表示灯で、入出力の状態を確認することもできます。

ユニットテスト

CPUユニットのアクセスウィンドウを使用して《KV-SIR32XT》の出力の状態をテストする手順について説明します。

ポイント **ユニットテストはCPUユニットがPRGモード時のみ可能です。**

参考 KV STUDIOで設定するユニット構成の設定を含んだプロジェクトがCPUユニットに転送されていない場合でも確認できます。

テスト方法

1 メニュー画面から「・テスト」で「ENTER」キーを押した後、テストしたいチャンネルを「▲」「▼」キーで選択し、「ENTER」キーを押します。

選択したチャンネルの出力のテスト画面が表示されます。

2 テストしたい端子を「▲」「▼」「◀」「▶」キーで選択し、「ENTER」キーを押します。

選択した端子の出力のON/OFF状態が切り換わります。

6-10 ユニットモニタ

ラダーサポートソフト《KV STUDIO》のユニットモニタ機能は、モニタするユニットで使用しているリレーやDMの値を、ユニットにあわせて最適に表示する機能です。ここでは、ユニットモニタの見かたや操作方法について説明します。

ユニットモニタの概要

ユニットモニタは、モニタするユニットで使用しているリレーやDMの値を、ユニットにあわせて最適に表示するウィンドウです。

ユニットモニタには次のような特長があります。

- ユニット毎に割り付けられているDM、リレーを最初から登録した状態で表示できます。
- ユニット毎に表示内容やレイアウトが雛形として用意されているため、モニタが簡単です。
- リレーやDMの割付状況を気にしなくても、モニタしたいユニットを選択するだけで、選択したユニットのモニタができます。

ポイント　**ユニットモニタは、モニタ・オンラインエディット中にのみメニューから選択できます。**

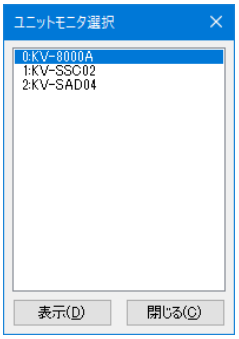
ユニットモニタの表示

次の手順でモニタしたいユニットを表示します。

1 **メニューから【モニタ/シミュレータ】【ユニットモニタ】を選択します。**

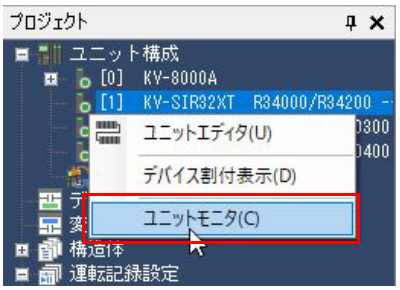
「ユニットモニタ選択」ダイアログボックスが表示されます。

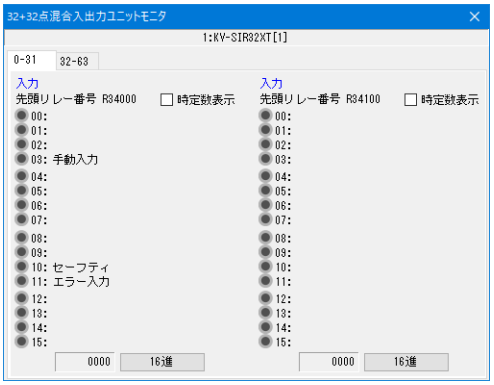
2 **モニタしたいユニットを選択して、「表示」ボタンをクリックします。**



別手順

- ワークスペースのユニット設定で、モニタしたいユニットを選択し、ダブルクリックします。または、右クリックメニューから「ユニットモニタ」を選択します。
- ユニットエディタ上で、モニタしたいユニットを選択し、右クリックメニューから「ユニットモニタ」を選択します。または、メニューから【表示】【ユニットモニタ】を選択します。





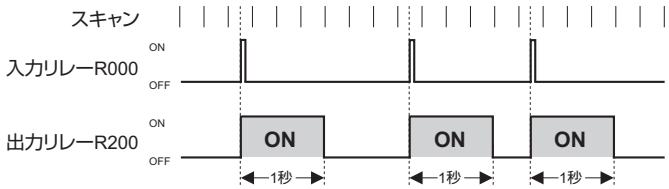
6-11 I/Oユニット参考プログラム例

≪KV-SIR32XT≫を使用した場合に使用できるサンプルプログラムをご紹介します。
以下のプログラムではKV-SIR32XTをR000から割り付けた場合の例で説明します。

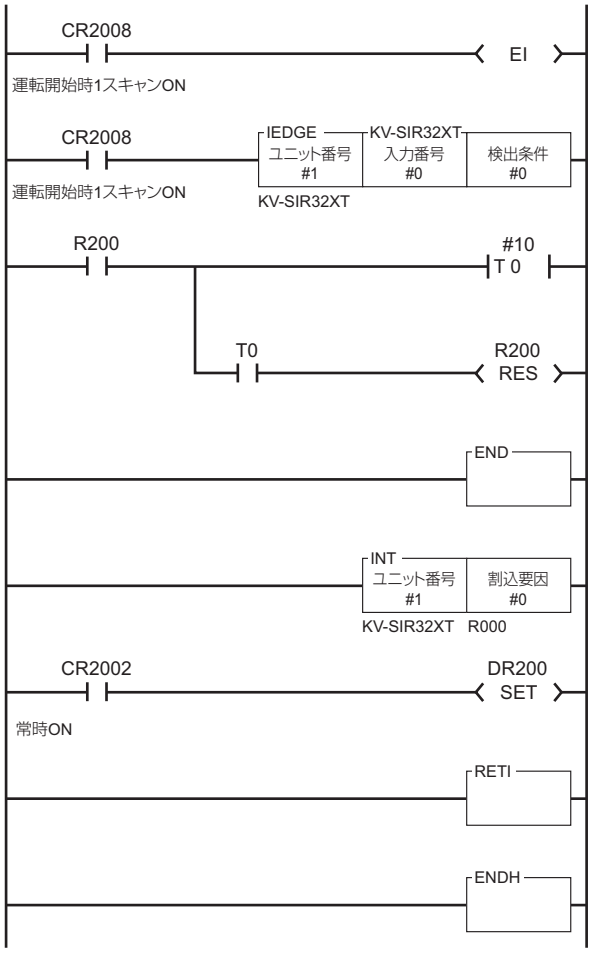
■ 信号変換器を使用した割り込みの応用例

割り込み処理を使用した微小パルスを一定の幅のパルスに変換して出力します。

- 入力リレーR000で入力した微小パルスを、出力リレーR200からON幅1秒のパルスに変換します。
- 入力リレーR000の立ち上がりから1秒間、出力リレーR200がONします。



信号変換機を使用した割り込み



電源投入時に割り込みを許可します。

入力リレーR000の割り込み検出条件を立ち上がりに設定します。

出力リレーR200を1秒間だけONします。

入力リレーR000がONすると出力R200がONします。

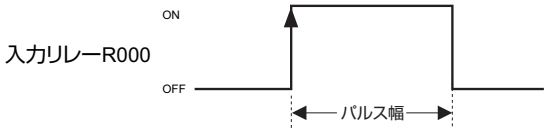
ユニットエディタの
KV-SIR32XT 設定
先頭 DM 番号：－
先頭リレー番号：000
IN000 入力時定数：1μs

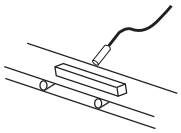
< ニモニックリスト >

```
LD CR2008
EI
LD CR2002
IEDGE #1 #0 #0
LD R200
MPS
TMR #0 #10
MPP
ANB T0
RES R200
END
INT #1 #0
LD CR2002
SET DR200
RETI
ENDH
```

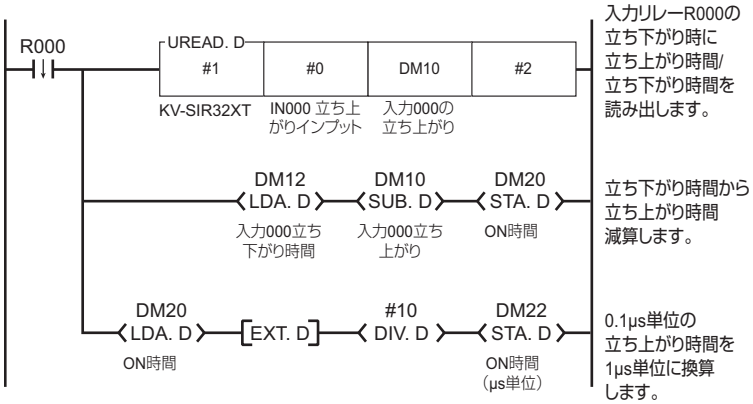
■ 高速パルスON幅の測定の応用例

入力リレーに入る入力パルスのパルス幅を測定します。





- インプットキャプチャ機能を使用して、0.1μs単位でパルス幅を測定します。
- 入力リレーR000の立ち下がり時に立ち上がり時間/立ち下がり時間を読み出します。
- 立ち下がり時間から立ち上がり時間を減算することで、入力リレーのON時間を測定します。
- ON時間をμs単位に換算します。

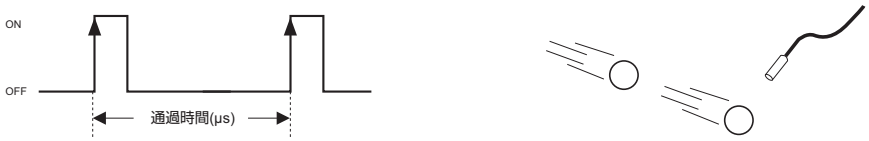


ユニットエディタの
KV-SIR32XT 設定
先頭 DM 番号：－
先頭リレー番号：000
IN000 入力時定数：1μs

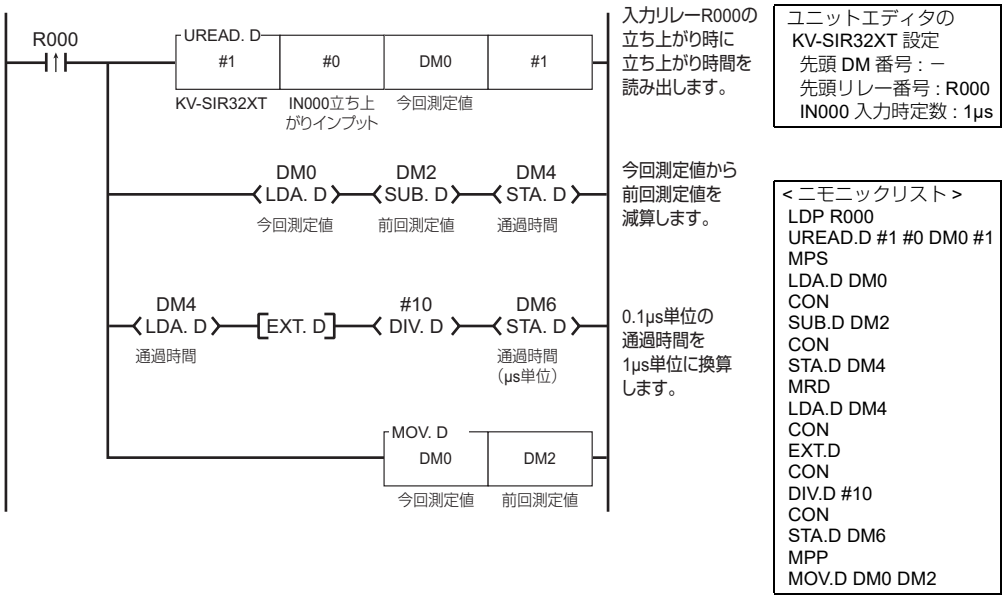
```
< ニモニックリスト >
LDF R000
UREAD.D #1 #0 DM10 #2
MPS
LDA.D DM12
CON
SUB.D DM10
CON
STA.D DM20
MPP
LDA.D DM20
CON
EXT.D
CON
DIV.D #10
CON
STA.D DM22
```

■ ワークの通過間隔の測定の応用例

ワークが検出される通過間隔を測定します。

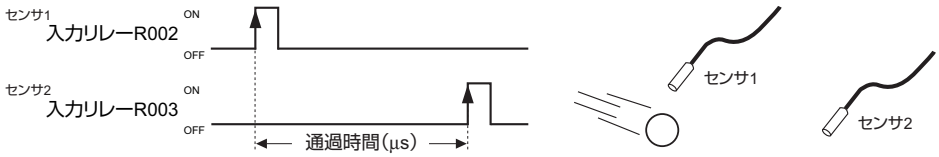


- 入力リレーR000がONしてから再度R000がONするまでの通過間隔を測定します。
- R000の立ち上がりで立ち上がり時間を読み出しします。
- 今回の測定値と前回の測定値との差分を1μs単位でデータメモリ[DM6/DM7]に格納します。
- 今回の測定値を前回の測定値に格納します。

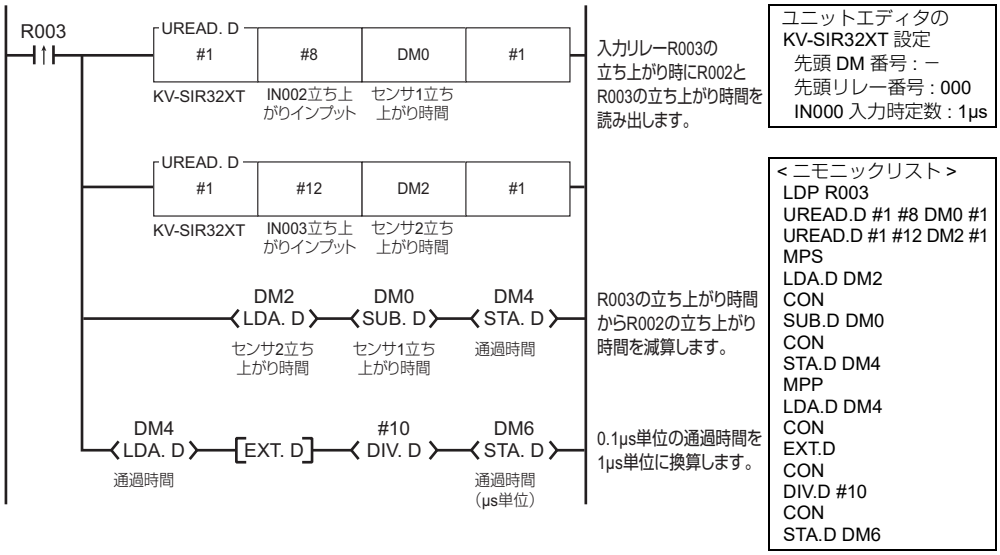


■ 2点間通過時間の測定の応用例

2つの入力リレー間の通過時間を測定します。



- 入力リレーR002から入力リレーR003までの通過時間を測定します。
- R003の立ち上がりでセンサ1とセンサ2の立ち上がり時間を読み出します。
- センサ1とセンサ2の立ち上がり時間の差分を1μs単位でデータメモリ[DM6/DM7]に格納します。



6-12 I/Oユニットのエラーコード

≪KV-SIR32XT≫使用時に発生するエラーコードは以下のとおりです。

番号	メッセージ	原因	対処
101	入力過電圧 IN100～IN115 Input over voltage IN100～IN115	IN100～IN115を5Vレンジで使用しており、定格電圧以上の電圧が入力されました。	電源をOFFし、要因を取り除き、再度電源を入力してください。
202	出力過電流 OUT200～OUT215 Output overcurrent OUT200～OUT215	OUT200～OUT215のいずれかで過電流を検出しています。	要因を取り除いてください。
203	出力過電流 OUT300～OUT315 Output overcurrent OUT300～OUT315	OUT300～OUT315のいずれかで過電流を検出しています。	要因を取り除いてください。

MEMO

付　録

KV-8000シリーズのコントロールリレー（CR）・コントロールメモリ（CM）一覧やエラー一覧、本書で取り扱う命令語一覧を掲載しています。

1　PLC本体エラー一覧 付-2
2　CR/CM一覧 付-10
3　ASCIIコード表 付-18
4　命令語一覧 付-19
5　使用できない命令語一覧 付-21
6　ニモニック短縮入力一覧 付-23
7　ショートカットキー一覧 付-25
8　ユーザ認証機能 付-39
9　使用できない文字一覧 付-40
10　既存のKVシリーズ拡張ユニット使用時の注意事項について 付-43
11　KV-8000シリーズへの機種変更時の注意事項 付-46
12　索　引 付-54

1 PLC本体エラー一覧

CPUユニット本体で発生するエラーについて説明します。
これらのエラーは、次の方法で確認できます。

- アクセスウィンドウ
- 《KV STUDIO》の〔エラーモニタ〕
- CR/CM

重度エラー/軽度エラーはグループによって分類されています。重度エラーが発生するとラダープログラムが停止するため、ラダープログラムからはエラークリアできません。

重度エラー･･･　スキャンタイムオーバーエラー/システム異常/設定エラー

軽度エラー･･･　時刻異常/演算エラー/ユニットエラー/メモリカード操作エラー/メモリカードアクセスエラー/その他のエラー

参 考　重度エラー発生中はCR3905がONし、CM5105にエラーコードが格納されます。軽度エラー発生中はCR3906がONし、CM5155にエラーコードが格納されます。

番号	グループ	メッセージ	原因	対処
10	設定エラー	サブルーチン ネスト制限オーバー SBN Exceeds Nesting Limit 子程序嵌套超限	サブルーチンのネスティング(入れ子)が制限を超えました。	サブルーチンのネスト制限は16段です。 ラダー実行時に発生するエラーは、《KV STUDIO》の変換ではエラーになりません。サブルーチンの使い方に注意して、ラダーを見直してください。
11	設定エラー	FOR-NEXT ネスト制限オーバー FOR-NEXT Exceeds Nesting Limit FOR-NEXT 嵌套超限	FOR～NEXT命令のネスティング(入れ子)が制限を超えました。	FOR～NEXT命令のネスト制限は16段です。 ラダー実行時に発生するエラーは、《KV STUDIO》の変換ではエラーになりません。FOR～NEXT命令の使い方に注意して、ラダーを見直してください。
15	設定エラー	コンバートエラー Failed To Convert 转换错误	コンバートに失敗しました。	《KV STUDIO》で本体からプロジェクトを読み出して、変換を実行します。変換時のエラー･警告の内容を確認し、原因を取り除いた状態でプロジェクトを本体に転送してください。 ※一部の警告については、原因が取り除かれていなくてもクリアされることがあります。
22	設定エラー	マクロ ネスト制限オーバー Macro Exceeds Nesting Limit 宏嵌套超限	多重割り込み中にマクロネストの段数が制限を超えました。	割り込みプログラム内のマクロ命令を減らしてください。
29	システム異常	ラダースタックエラー Ladder Stack Error 梯形图堆栈错误	システムエラーが発生しました。	電源を再投入してください。 頻発する場合は、最寄りの営業所までお問い合わせください。
30	スキャンタイム オーバー	スキャンタイム オーバー Scan Time Over 扫描超时	スキャンタイムが設定時間以上を超えました。	以下などによりスキャンタイムが300msを超えてしまっていないか、ラダープログラムを見直してください。 • FOR～NEXT命令など、繰り返し行う命令 • CJ/NCJ～LABEL命令など、処理をジャンプする命令 • CALL,MCALL命令などプログラムを呼び出す命令 • FMOV,WSR,MAX命令など、処理時間の長い命令 • 割り込みプログラムを高頻度で実行していないか

番号	グループ	メッセージ	原因	対処
31	システム異常	ラダースタック オーバーフロー Ladder Stack Overflow 梯形图堆栈溢出	システムエラーが発生しました。	電源を再投入してください。 頻発する場合は、最寄りの営業所までお問い合わせください。
33	設定エラー	プログラムネスト制限 オーバー Program Exceeds Nesting Limit 程序嵌套超限	プログラムのネストの段数が制限を超えました。	プログラムのネスト段数の制限は8段です。 ラダー実行時に発生するエラーは、《KV STUDIO》の変換ではエラーになりません。ファンクションブロック/ファンクションの使い方に注意して、ラダーを見直してください。
40	設定エラー	ラダープログラムなし No Ladder Program 无梯形图程序	プログラムがありません。	ラダープログラムを本体に転送してから、動作させてください。
50	設定エラー	ユニット設定 チェックサム異常 Unit Info Checksum Error 单元设定检验和异常	拡張ユニットの設定が破損しています。	ユニット設定情報を本体に転送してください。 強電界、強磁界が発生する場所に設置している場合は、機器を遠ざけてください。
51	設定エラー	ユニット設定情報 未設定 No Unit Info Set 单元设定信息未设定	ユニット設定情報が設定されていません。	ユニット設定情報を本体に転送してください。
52	設定エラー	ユニット未接続 Unit Unconnected 单元未连接	ユニット設定情報では拡張ユニットありになっているのに、実際には接続されていません。	電源をOFFにしてから、ユニットエディタで設定した拡張ユニットを接続してください。
53	システム異常	拡張バスユニット タイムアウト Expansion Bus Unit Timeout 扩展总线单元超时	拡張ユニットと一定期間バス通信できませんでした。	ノイズ源（例：モータなど大電流を流すケーブル）が近接する場合は、機器から遠ざけてください。 頻発する場合は、最寄りの営業所までお問い合わせください。
54	設定エラー	ユニット台数 不一致 Number of Units Mismatch 单元台数不一致	ユニットエディタの設定と実際の拡張ユニットの接続台数が異なります。	電源をOFFにしてから、ユニットエディタで設定した拡張ユニットを接続してください。
55	設定エラー	ユニット種別 不一致 Unit Type Mismatch 单元类型不一致	ユニットエディタの設定と実際の拡張ユニットの種類が異なります。	電源をOFFにしてから、ユニットエディタで設定した拡張ユニットを接続してください。
56	システム異常	拡張バス通信エラー Expansion Bus Comm Error 扩展总线通信错误	拡張ユニットと正常にバス通信できませんでした。	拡張ユニットが正常に接続されているか確認してください。 ノイズ源（例：モータなど大電流を流すケーブル）が近接する場合は、機器から遠ざけてください。 電源断の直前に起きた場合はエラー履歴に残りません。 頻発する場合は、最寄りの営業所までお問い合わせください。

番号	グループ	メッセージ	原因	対処
57	設定エラー	エンドユニット未接続 No End Unit 结束单元未连接	エンドユニットが接続されていません。	システムの電源をOFFにしてから、エンドユニットを接続してください。
59	設定エラー	ユニットバージョンエラー Unit Version Error 单元版本错误	接続された拡張ユニットのバージョンでは使用できない機能が設定されています。または、CPUユニットに接続できない拡張ユニットを接続しています。	電源をOFFにしてから、ユニットエディタで設定した拡張ユニットを接続してください。 またはシステムバージョンアップに対応しているユニットの場合は、システムバージョンを最新にしてください。
63	設定エラー	電源再投入が必要 Power Restoring Needed 需要重新接通电源	ユニット構成の変更を反映するには電源の再投入が必要です。	電源を再投入してください。
80	システム異常	FLASH ROM異常 Flash ROM Failure 闪存异常	フラッシュROMが異常です。	最寄りの営業所までお問い合わせください。
83	時刻異常	電池電圧の低下 Battery Voltage Degradation 电池电压降低	KV-B1の電圧が低下しています。	KV-B1を交換してください。
85	時刻異常	時計データ消失 Clock Data Lost 时钟数据缺失	内蔵のバックアップ用コンデンサの電圧が低下したため、時計データが消失しました。長期間通電されなかったか、ハードウェアの故障です。	必要に応じて時計を再設定してください。時計データを保持するためには、通電して内蔵のバックアップ用コンデンサを充電するか、オプションのバッテリ(KV-B1)を取り付けるかを行う必要があります。 頻発する場合は、最寄りの営業所までお問い合わせください。
86	システム異常	RTC 異常 RTC Failure RTC 异常	時計データが異常です。	頻発する場合は、最寄りの営業所までお問い合わせください。
87	メモリカードアクセスエラー	メモリカードアクセスエラー Memory Card Access Error 存储卡访问错误	メモリカードの劣化、接点部の異常、挿入状態の異常やノイズなどの外的要因の影響により、メモリカードへのアクセスに失敗しました。	メモリカードの挿入状態を確認してください。 メモリカードのファイルシステムをチェックしてください。 Windowsのコマンドプロンプトから chkdskコマンドを用いてチェックしてください。 (例)メモリカードがEドライブのとき C：￥WINDOWS＞CHKDSK E： ノイズ源(例：モータなど大電流を流すケーブル)が近接する場合は、機器から遠ざけてください。
88	メモリカード操作エラー	メモリカードが抜かれました Memory Card Removed 存储卡被取出	メモリカードアクセス中にカードが抜かれました。	メモリカードアクセス中はメモリカードを抜かないでください。

番号	グループ	メッセージ	原因	対処
89	メモリカード操作エラー	ストレージアクセス中電源OFF During Storage Access PowerOFF 存储体访问过程中断电	ストレージアクセス中に電源が切れました。	ストレージアクセス中は電源を切らないでください。
90	その他のエラー	ストレージアクセス完了待ち Waiting Storage Access 存储体访问完成等待	ストレージアクセス中のため、RUNモードに切り換えできませんでした。	ストレージアクセス完了後に操作してください。
92	システム異常	アクセスウィンドウ異常 Access Window Failure 访问窗异常	アクセスウィンドウと正常に通信できませんでした。	エラークリア後、電源を再投入してください。 頻発する場合は、最寄りの営業所までお問い合わせください。
95	その他のエラー	ファイルアクセス中 Accessing File 文件访问中	同時に処理できない機能を実行しました。	PLC転送･ロード/セーブ･オールクリア･初期化･ユーザメモリに対する操作の実行中は、PRG→RUN切換･ユニットテスト･オールクリアを実行できないため、実行タイミングをずらして実行してください。
102	システム異常	POWER OFF エラー POWER OFF Error 断电错误	ハードウェアの故障か、ノイズなどにより、電源断時のデータ退避処理が正常に終了できませんでした。	ノイズ源（例：モータなど大電流を流すケーブル）が近接する場合は、機器から遠ざけてください。 強電界、強磁界が発生する場所に設置している場合は、機器を遠ざけてください。 以下の機能を使用して、RAMクリアを実行してください。 • 《KV STUDIO》のPLCクリア • 増設アクセスウィンドウカセット 頻発する場合は、最寄りの営業所までお問い合わせください。
103	システム異常	電断処理超過 Shutdown Timeout 断电处理超时	システムエラーが発生しました。	強電界、強磁界が発生する場所に設置している場合は、機器を遠ざけてください。 それでも解決しない場合は最寄りの営業所までお問い合わせください。
106	システム異常	リセット異常 Abnormal Reset 复位异常	ハードウェアの故障か、ノイズなどにより、電源断時のデータ退避処理が正常に開始できませんでした。	ノイズ源（例：モータなど大電流を流すケーブル）が近接する場合は、機器から遠ざけてください。 強電界、強磁界が発生する場所に設置している場合は、機器を遠ざけてください。 以下の機能を使用して、RAMクリアを実行してください。 • 《KV STUDIO》のPLCクリア • アクセスウィンドウ 頻発する場合は、最寄りの営業所までお問い合わせください。
107	その他のエラー	システムエラー System Error 系统错误	システムエラーが発生しました。	システムは運転続行できますが、重大なエラーにつながる恐れがあります。 強電界、強磁界が発生する場所に設置している場合は、機器を遠ざけてください。 頻発する場合は、最寄りの営業所までお問い合わせください。

番号	グループ	メッセージ	原因	対処
108	その他のエラー	コンバートワーニング Convert Warning 转换警告	コンバートでワーニングが発生しました。	ロギング設定、運転記録設定作成時のプロジェクトの情報と本体のプロジェクトの情報が異なります。 《KV STUDIO》で本体からプロジェクトを読みだして変換を実行し、再度本体に転送してください。 上記でクリアされない場合は、変換時に表示される警告の内容を確認し、原因を取り除いた状態でプロジェクトを本体に転送してください。 ※一部の警告については、原因が取り除かれていなくてもクリアされる場合があります。
118	システム異常	不正な伝送データ Invalid Transfer Data 不正确的传输数据	システムエラーが発生しました。	電源を再投入してください。 頻発する場合は、最寄りの営業所までお問い合わせください。
120	システム異常	ウォッチドッグタイマ Watchdog Timer 看门狗定时器	システムエラーが発生しました。	電源を再投入してください。 強電界、強磁界が発生する場所に設置している場合は、機器を遠ざけてください。 頻発する場合は、最寄りの営業所までお問い合わせください。
122	システム異常	システムエラー System Error 系统错误	システムエラーが発生しました。	電源を再投入してください。 強電界、強磁界が発生する場所に設置している場合は、機器を遠ざけてください。 頻発する場合は、最寄りの営業所までお問い合わせください。
125	その他のエラー	エラーバッファオーバーフロー Error Buffer Overflow 错误缓存溢出	システム内で同時に発生しているエラー発生数が制限を超えました。	同時に発生するエラーが16個を超えないようにしてください。 エラークリアしてください。
127	設定エラー	オートローディング失敗 Auto Loading Failed 自动加载失败	オートローディングに失敗しました。	データが存在しない場合や、PLC機種が異なるデータの場合は、ロードできません。 PLCに格納されているプロジェクトにライトプロテクト/プロジェクトロード禁止がかかってる場合は、オートローディングできません。 上記に該当しない場合は、ロード･セーブ実行時のエラーコードをご確認ください。「ロード･セーブ実行時のエラーコード」(5-19ページ)
128	演算エラー	ラダー演算エラー Ladder Calc 梯形图运算错误	演算エラー （CR2012がONするエラー）	ラダーを確認して、演算エラーの原因を修正してください。
129	ユニットエラー	ユニットエラー Unit Error 单元错误	拡張ユニットでエラーが発生しました。	各ユニットのマニュアルを参照してください。 エラーが発生しているユニットを確認する方法については、「エラークリア」(5-129ページ)を参照してください。
131	システム異常	FLASH ROMフォーマット FLASH ROM Format 闪存格式	オールクリア/初期化中に電源が切れた、ハードウェアの故障、ノイズなどの影響により、フラッシュROMのフォーマットが異常です。	FLASH ROMフォーマットが発生した場合、20分程度FLASH ROMのフォーマットを継続する場合があります。20分程度経過後、エラーが解除されたことを確認し、オールクリア後、プロジェクトを再度転送してください。 頻発する場合は、最寄りの営業所までお問い合わせください。

番号	グループ	メッセージ	原因	対処
132	システム異常	ラダーファイル不正 Invalid Ladder File 梯形图文件不正确	オールクリア/初期化中に電源が切れた、ハードウェアの故障、ノイズなどにより、フラッシュROMに格納しているラダーが異常です。	オールクリア後、プロジェクトを再度転送してください。 頻発する場合は、最寄りの営業所までお問い合わせください。
135	メモリカード操作エラー	メモリカードカバーオープン Memory Card Cover Open 存储卡外盖打开	メモリカードアクセス時にメモリカードカバーが開いていました。	メモリカードカバーを閉じてください。
136	メモリカード操作エラー	メモリカードLOCKされています Memory Card is Locked 存储卡被锁定	メモリカード書き込み処理時に、メモリカードがLOCK(ライトプロテクト)されていました。	LOCKを解除するか、書込み可能なメモリカードを使用してください。
137	メモリカード操作エラー	非対応のメモリカード種類 Unsupported Memory Card Type 不支持的存储卡种类	非対応のメモリカードを装着しました。	SDカード、またはSDHCカードを使用してください。
141	その他のエラー	CPUメモリ書込頻度オーバー CPU Memory Write Frequency Over CPU存储器 写入频度超出	高頻度でCPUメモリへの書き込みが実行されました。	CPUメモリへの書込み頻度を減らしてください。
142	その他のエラー	CPUメモリ累積書込回数警告 CPU Memory Write Cycle Warning CPU存储器累积写入次数警告	CPUメモリへの累積書込み回数が耐用回数に近づいています。	CPUメモリへの書込み頻度を減らしてください。また、交換するCPUユニットを準備してください。
143	その他のエラー	CPUメモリ累積書込回数オーバー CPU Memory Write Cycle Over CPU存储器 累积写入次数超出	CPUメモリへの累積書込み回数が耐用回数を超えました。	CPUユニットのCPUメモリデータを読み出した後、CPUユニットを交換してください。
144	その他のエラー	初期化済み Initialized 初始化完成	初期化を実行しました。	ラダープログラムを本体に転送してから、動作させてください。
250	その他のエラー	Start Up Error Please Reset Start Up Error Please Reset Start Up Error Please Reset	起動中に異常が発生しました。	電源を再投入してください。(エラークリアはできません。また、エラー履歴に残りません。) 頻発する場合は、最寄りの営業所までお問い合わせください。
251	その他のエラー	AW Timeout Error AW Timeout Error AW Timeout Error	ハードウェアの故障か、ノイズなどにより、アクセスウィンドウと正常に通信できませんでした。	振動源やノイズ源(例：モータなど大電流を流すケーブル)が近接する場合は、機器から遠ざけて、電源を再投入してください。 頻発する場合は、最寄りの営業所までお問い合わせください。
—	その他のエラー	Safe Mode Safe Mode Safe Mode	異常なシステムプログラムのため、正常起動できませんでした。	KV STUDIOからのシステムプログラム更新時に通信エラーが発生がした場合は、他のパソコンや他のUSBケーブルを使用して再度システムプログラム更新を実施してください。 頻発する場合、KV STUDIOと通信できない場合は最寄りの営業所までお問い合わせください。

- No.128（演算エラー）の場合

詳細情報n	説　明
詳細情報1 （CM5157）	エラーの詳細内容 0　（未使用） 1　間接指定が不適切 2　間接指定先がT/C 3　簡易間接指定が不適切 4　オペランド範囲外 5　浮動小数点型実数オーバーフロー 6　デバイス指定が不適切 7　ROOT命令の引数が負 8　ENCO命令でONしているビットがない 9　浮動小数点型実数フォーマット異常 10　三角関数命令の引数が不適切 11　変換できないデータ 12　テーブル指定が不適切 13　時刻比較命令の引数が不適切 14　カムスイッチオペランドが不適切 15　周波数カウンタ命令の引数が不適切 16　0除算 17　ロギングID不正 18　ダイレクト処理エラー 19　ダイレクト処理できないユニット 20　ジャンプ先サブルーチンが存在しない 21　RUN中書き込み中なので命令が実行不可 22　拡張ユニット専用命令対象エラー（ユニット番号） 23　システム予約 24　PIDAT 動作設定範囲外 25　PIDAT サンプリング周期範囲外 26　PIDAT 比例定数範囲外 27　PIDAT 積分定数範囲外 28　PIDAT 微分定数範囲外 29　PIDAT 比例定数2自由度化係数範囲外 30　PIDAT 積分定数2自由度化係数範囲外 31　PIDAT 微分定数2自由度化係数範囲外 32　PIDAT 操作量上限値･下限値範囲外 33　PIDAT 操作量変化幅制限値範囲外 34　PIDAT 測定値変化幅範囲外 35　PIDAT AT状態遷移異常 36　PIDAT 動作制御フラグ範囲外 37　演算周期オーバー 38　PIDAT AT調整パラメータ範囲外 39　PIDAT ATヒステリシス範囲外 40　PIDAT ATタイムアウト時間範囲外 41　PIDAT AT算出制御方式設定範囲外 42　PIDAT ATタイムアウト 43　PIDAT AT比例定数オーバー 44　PIDAT AT積分定数オーバー 45　PIDAT AT微分定数オーバー 46　負の数の累乗根は演算できない 47　ABS命令の引数が不適切 48　演算オーバーフロー 49　UR/UM/UV ユニット未接続 50　ユニットデバイス（R）未割付 51　ユニットデバイス（DM）未割付 52　ユニット種別不一致 53　運転記録ID不正 54　スタックオーバーフロー
詳細情報2 （CM5158）	00H固定
詳細情報3 （CM5159）	エラーが発生しているステップ番号上位
詳細情報4 （CM5160）	エラーが発生しているステップ番号下位

詳細情報n	説　明
詳細情報5～20 （CM5161～CM5176）	モジュール名をプロジェクト言語設定に従って格納します。 （モジュール名以外の部分は00H）

- No.50,53,55,56,58,59,63（ユニット番号に関するエラー）の場合

詳細情報n	説　明
詳細情報1 （CM5107）	ユニット番号（エラーの発生状況によってユニット番号が特定できない場合には、格納されません。）
詳細情報2～20 （CM5108～CM5126）	00H固定

- No.54（ユニットの台数に関するエラー）の場合

詳細情報n	説　明
詳細情報1 （CM5107）	ユニット設定情報でのユニットの台数
詳細情報2 （CM5108）	実際に接続されているユニットの台数
詳細情報3～20 （CM5109～CM5126）	00H固定

- No.127（オートローディング失敗エラー）の場合

詳細情報n	説　明
詳細情報1 （CM5157）	エラーの詳細内容 詳細については、「ロード/セーブ実行時のエラーコード」（5-144ページ）を参照してください。
詳細情報2 （CM5158）	詳細情報1が20の場合、ユニット番号 上記以外の場合、00H固定
詳細情報2～20 （CM5159～CM5176）	00H固定

- No.129（ユニットエラー）の場合

拡張情報n	解　説
拡張情報1 （CM5157）	ユニット番号
拡張情報2 （CM5158）	ユニット毎のエラー番号
拡張情報3～20 （CM5159～CM5176）	00H固定

- 上記以外のエラー番号の場合

拡張情報n	解　説
拡張情報1～20	00H固定

コントロールリレー CR

コントロールリレー（CR0000～CR7915）は、CPUユニットの動作状態の確認、各内蔵機能の設定やモニタなどができます。
システム予約されているコントロールリレーは使用できません。

CR	属性 R：読出専用 空欄：読書可能	機能	
CR0000		ロギング･トレース機能（ID0）	ロギング･トレース 実行（ON：許可、OFF：禁止）
CR0001	R		ロギング･トレース 動作中
CR0002	R		ロギング･トレース ファイル書き込み終了
CR0003	R		ロギング バッファ50%未満警告
CR0004	R		ロギング バッファオーバーフロー
CR0005	R		ロギング 実行中RUN中書き込みあり
CR0006～ CR0007	－		システム予約
CR0008	R		ロギング･トレース エラー
CR0009	R		ロギング･トレース メモリ容量なしエラー
CR0010	R		ロギング･トレース デバイス･トリガ設定異常
CR0011	R		トレース ファイル保存トリガモニタ
CR0012	R		トレース データ取得完了
CR0013	R		ロギング･トレース 設定書き込みあり
CR0014～ CR0015	－		システム予約
CR0100～ CR0915		ロギング･トレース機能（ID1～ID9）	
CR1000～ CR1015		リアルタイムチャートモニタ（ID10）	
CR1100～ CR2001	－	システム予約	
CR2002	R	常時ON	
CR2003	R	常時OFF	
CR2004	R	10msクロックパルス（duty50%）	
CR2005	R	100msクロックパルス（duty50%）	
CR2006	R	1sクロックパルス（duty50%）	
CR2007	R	運転開始時１スキャンのみOFF	
CR2008	R	運転開始時１スキャンのみON	
CR2009		演算結果が負またはオーバーフローのときON	
CR2010	R	演算結果がゼロのときON	
CR2011	R	演算結果が正のときON	
CR2012	R	演算実行エラーのときON	
CR2013～ CR2302	－	システム予約	
CR2303		一定スキャンタイム運転（ON：有効、OFF：無効）	
CR2304	R	スキャンタイム設定（（一定スキャンタイム運転、END処理時間設定）オーバー時1スキャンのみON）	
CR2305	－	システム予約	
CR2306	R	バックアップ電池異常（ON:異常、OFF：正常）	
CR2307		スキャンタイム最小値最大値のクリア	
CR2308		END処理時間設定 （ON：有効、OFF:無効）（CR2303と同時にONした場合は、CR2303が優先）	
CR2309～ CR2310	－	システム予約	
CR2311	R	RTC電圧低下エラー（ON：エラー発生中、OFF：正常）	
CR2312	－	システム予約	
CR2313		VTセンサアプリケーション画面強制切換（切換する：ON、切換しない：OFF）	
CR2314	R	プロジェクトパスワード認証禁止状態（ON：認証禁止、OFF:認証可能）	
CR2315～ CR2813	－	システム予約	

CR	属性 R：読出専用 空欄：読書可能	機能
CR2814		DASC、FASC命令（全サフィックス）実行時にゼロサプレスを（ON:する、OFF：しない）
CR2815		DASC.S、DASC.L、FASC実行時に＋の符号を省略（ON:する、OFF：しない）
CR2900～ CR3005	－	システム予約
CR3100	R	ユニット間同期リフレッシュ時間超過
CR3101～ CR3115	－	システム予約
CR3200		CPUメモリ　保存処理要求
CR3201	R	CPUメモリ　保存処理中
CR3202	R	CPUメモリ　保存処理完了
CR3203	R	CPUメモリ　保存処理エラー
CR3204～ CR3206	－	システム予約
CR3207		命令語 RAMモード動作 （CPUメモリに対するファイルレジスタ命令･ストレージ命令のRAMモード有効化 （ON:有効、OFF:無効））
CR3208～ CR3209	－	システム予約
CR3210	R	任意ロギング･トレース動作中
CR3211～ CR3213	－	システム予約
CR3214	R	ストレージ命令実行中
CR3215	－	システム予約
CR3300～ CR3415		アラームリレー
CR3500	R	アラーム動作リレー（ON：CR3300～CR3415のいずれかがON）
CR3501		アラームリレーがON→OFFのときに履歴を記録（ON：残す、OFF：残さない）
CR3502		OFF→ON時にアラームリレー履歴クリア
CR3503～ CR3512	－	システム予約
CR3513	R	ロギング内部バッファオーバフロー
CR3514		HKEY命令　複数キー禁止
CR3515	R	HKEY命令　スキャン終了
CR3600～ CR3615	R	HKEYの情報格納エリア
CR3700～ CR3815	－	システム予約
CR3900		OFF→ON時に重度エラー履歴クリア
CR3901		OFF→ON時に軽度エラー履歴クリア
CR3902	R	重度エラー履歴が最大件数
CR3903	R	軽度エラー履歴が最大件数
CR3904	R	RUN/PRG切換SW状態（ON:RUN側/OFF:PRG側）
CR3905	R	重度エラー発生中
CR3906	R	軽度エラー発生中
CR3907～ CR3908		システム予約
CR3909		現在のエラークリア（ラダープログラム中でSETしてください）
CR3910		OFF→ON時にすべてのエラー、イベント履歴をクリア
CR3911～ CR4000	－	システム予約
CR4001～ CR4300	R	拡張ユニット有無（拡張ユニット1～48）
CR4301～ CR4315	－	システム予約
CR4400	R	AW　上キー押下中
CR4401	R	AW　下キー押下中
CR4402	R	AW　左キー押下中

CR	属性 R：読出専用 空欄：読書可能	機能
CR4403	R	AW　右キー押下中
CR4404	R	AW　MENUキー押下中
CR4405	R	AW　ENTキー押下中
CR4406～ CR4414	－	システム予約
CR4415	R	AW　ユーザメッセージ表示中
CR4500		AW　1行目ブリンク
CR4501		AW　2行目ブリンク
CR4502		AW　3行目ブリンク
CR4503		AW　4行目ブリンク
CR4504～ CR4600	－	システム予約
CR4601～ CR4915		AW ユーザメッセージ1～63
CR5000～ CR5315	－	システム予約
CR5400		メモリカードマウント要求
CR5401		メモリカードアンマウント要求
CR5402	R	メモリカード使用中
CR5403	R	メモリカード認識完了
CR5404	R	メモリカードあり
CR5405	R	メモリカードライトプロテクト
CR5406～ CR5407	－	システム予約
CR5408		CPUメモリマウント要求
CR5409		CPUメモリアンマウント要求
CR5410	R	CPUメモリ使用中
CR5411	R	CPUメモリ認識完了
CR5412～ CR5415	―	システム予約
CR5500		プロジェクトロード実行要求
CR5501	―	システム予約
CR5502		プロジェクトセーブ実行要求
CR5503	―	システム予約
CR5504	R	プロジェクトロード実行完了
CR5505	R	プロジェクトロード実行失敗
CR5506	R	プロジェクトセーブ実行完了
CR5507	R	プロジェクトセーブ実行失敗
CR5508		レシピロード実行要求
CR5509	―	システム予約
CR5510	R	レシピロード実行完了
CR5511	R	レシピロード実行失敗
CR5512		レシピセーブ実行要求
CR5513	―	システム予約
CR5514	R	レシピセーブ実行完了
CR5515	R	レシピセーブ実行失敗

CR	属性 R：読出専用 空欄：読書可能	機能	
CR5600		運転記録機能（ID0）	運転記録0 保存要求
CR5601	R		運転記録0 動作中
CR5602	R		運転記録0 保存トリガモニタ
CR5603	R		運転記録0 保存中
CR5604	R		運転記録0 収集中
CR5605	R		運転記録0 設定異常
CR5606	—		システム予約
DR5607	R		運転記録0 保存完了
CR5608	R		運転記録0 保存エラー
CR5609	R		デバイス記録0 バッファ不足
CR5610～ CR5613	—		システム予約
CR5614			運転記録0 バッファ領域全保存モード
CR5615	—		システム予約
CR5700～ CR5919		運転記録機能（ID1～ID3）	
CR6000～ CR7915		システム予約	

コントロールメモリ CM

コントロールメモリ（CM0000～CM7599）は、CPUユニットの動作状態の確認、各内蔵機能の設定やモニタなどができます。
システム予約されているコントロールメモリは使用できません。

CM	属性 R：読出専用 空欄：読書可能	機能	保持 ○：保持
CM0000～CM0699	–	システム予約	
CM0700	R	RTCからの読み出し（年）	
CM0701	R	RTCからの読み出し（月）	
CM0702	R	RTCからの読み出し（日）	
CM0703	R	RTCからの読み出し（時）	
CM0704	R	RTCからの読み出し（分）	
CM0705	R	RTCからの読み出し（秒）	
CM0706	R	RTCからの読み出し（曜日）	
CM0707～CM0709	–	システム予約	
CM0710	R	ファイルレジスタのカレントバンク番号	
CM0711	–	システム予約	
CM0712	R	演算エラー詳細情報	
CM0713～CM0899	–	システム予約	
CM900	R	CPU機能バージョン(16進数表記で$0202の場合、バージョン2.02)	
CM901～CM906	–	システム予約	
CM907	R	プロジェクトバージョン(16進数表記で$0202の場合、バージョン2.02)	
CM908～CM999	–	システム予約	
CM1000～CM1001	R	スキャンタイム測定値（1μs）（32ビット）	
CM1002～CM1003	R	END処理時間測定値（1μs単位）（32ビット）	
CM1004～CM1005	R	スキャンタイム最小値（1μs単位）（32ビット）	
CM1006～CM1007	R	スキャンタイム最大値（1μs単位）（32ビット）	
CM1008～CM1009		一定スキャンタイム運転時設定値（1μs単位）（32ビット）	
CM1010～CM1011		END処理時間設定値（1μs単位）（32ビット）	
CM1012～CM1013	R	設定超過時スキャンタイム（1μs単位）（32ビット）	
CM1014～CM1015	R	END処理時間超過値（1μs単位）（32ビット）	
CM1016～CM1029	–	システム予約	
CM1030～CM1031	R	32ビット1msフリーランカウンタ	
CM1032～CM1033	R	32ビット10μsフリーランカウンタ	
CM1034～CM1035	R	32ビット1μsフリーランカウンタ	
CM1036～CM1037	R	32ビット0.1μsフリーランカウンタ	
CM1038～CM1499	–	システム予約	
CM1500～CM1509	R	ロギング･トレース（ID0～ID9）ファイル保存カウンタ	
CM1510～CM1629	–	システム予約	

CM	属性 R：読出専用 空欄：読書可能	機能	保持 ○：保持
CM1630～ CM1639	R	ロギング･トレース リングバッファの空き容量最小 (ID0～ID9)(KB単位)	
CM1640～ CM1649	－	システム予約	
CM1650	R	センサ設定命令実行数	
CM1651～ CM1657	－	システム予約	
CM1658	R	プロジェクトパスワード連続認証失敗回数	○
CM1659	R	プロジェクトパスワード累積認証失敗回数	○
CM1660～ CM1669	R	ロギング･トレース リングバッファの空き容量(ID0~ID9)(KB単位)	
CM1670～ CM1679	－	システム予約	
CM1680～ CM1689	R	ロギング･トレース ファイル保存カウンタ (ID0～ID9)(メモリカード用)	○
CM1690～ CM1699	－	システム予約	
CM1700～ CM1701		デジタルトリマ0上限値(32ビット)	
CM1702～ CM1703		デジタルトリマ1上限値(32ビット)	
CM1704～ CM1705		デジタルトリマ2上限値(32ビット)	
CM1706～ CM1707		デジタルトリマ3上限値(32ビット)	
CM1708～ CM1709		デジタルトリマ4上限値(32ビット)	
CM1710～ CM1711		デジタルトリマ5上限値(32ビット)	
CM1712～ CM1713		デジタルトリマ6上限値(32ビット)	
CM1714～ CM1715		デジタルトリマ7上限値(32ビット)	
CM1716～ CM1719	－	システム予約	
CM1720	R	AW　ユーザメッセージ表示番号	
CM1721～ CM1735	－	システム予約	
CM1736		AW　初期表示言語設定	
CM1737		AW　初期表示コメント番号	
CM1738		AW　操作禁止設定	
CM1739	－	システム予約	
CM1740～ CM1772	R	ラダープロジェクト名(最大32文字＋終端コード)	
CM1773～ CM1779	－	システム予約	
CM1780		AW　初期画面設定 有効	
CM1781		AW　初期画面設定 デバイス種別	
CM1782		AW　初期画面設定 画面ID	
CM1783		AW　初期画面設定 画面ID	
CM1784		AW　初期画面設定 デバイス番号	
CM1785		AW　初期画面設定 デバイス番号	
CM1786		AW　初期画面設定 表示形式	
CM1787		AW　初期画面設定 デバイス名表記	
CM1788		AW　初期画面設定 キーロック状態	
CM1789	－	システム予約	
CM1790		オートロード　フォルダ番号(要求)	
CM1791	R	オートロード　完了コード	
CM1792	R	オートロード　フォルダ番号(完了)	
CM1793	－	システム予約	
CM1794		ランロード　フォルダ番号(要求)	
CM1795	R	ランロード　完了コード	

CM	属性 R：読出専用 空欄：読書可能	機能		保持 ○：保持
CM1796	R	ランロード フォルダ番号（完了）		
CM1797～ CM1868	–	システム予約		
CM1869	R	拡張ユニット接続台数		
CM1870		レシピロード　ドライブ番号		○
CM1871		レシピロード　フォルダ番号		○
CM1872	–	システム予約		
CM1873	R	レシピロード　完了コード		○
CM1874	–	システム予約		
CM1875		レシピロード　ドライブ番号		○
CM1876		レシピセーブ　参照フォルダ番号		○
CM1877		レシピセーブ　出力フォルダ番号		○
CM1878	R	レシピセーブ　完了コード		○
CM1879	–	システム予約		
CM1880		プロジェクトロード時ドライブ番号（要求）		○
CM1881		プロジェクトロード時フォルダ番号（要求）		○
CM1882	R	プロジェクトロード時ドライブ番号（完了）		○
CM1883	R	プロジェクトロード時フォルダ番号（完了）		○
CM1884	R	プロジェクトロード時完了コード		○
CM1875～ CM1877	–	システム予約		
CM1888		プロジェクトセーブ時ドライブ番号（要求）		○
CM1889		プロジェクトセーブ時フォルダ番号（要求）		○
CM1890	R	プロジェクトセーブ時ドライブ番号（完了）		○
CM1891	R	プロジェクトセーブ時フォルダ番号（完了）		○
CM1892	R	プロジェクトセーブ時完了コード		○
CM1893～ CM1999	–	システム予約		
CM2000～ CM2001	R	運転記録機能（ID0）	デバイス記録0 平均サンプリング周期	
CM2002～ CM2003	R		デバイス記録0 平均変化デバイス数	
CM2004～ CM2005	R		デバイス記録0 最大収集期間	
CM2006～ CM2009	–		システム予約	
CM2010 ～CM2039		運転記録機能（ID1～ID3）		
CM2040～ CM2389	–	システム予約		
CM2390	R	ストレージ命令用エラーコード		
CM2391		ストレージアクセス中電源OFF （発生時に0以外の値になる。プログラムから0にしてください。）		
CM2392		メモリカードリトライ発生回数（電源断時も保持）		
CM2393		メモリカードベリファイエラー発生回数（電源断時も保持）		
CM2394		メモリカードタイムアウト発生回数（電源断時も保持）		
CM2395		メモリカードエラー時リセット遅延時間		
CM2396～ CM2399	–	システム予約		
CM2400～ CM2401	R	ユニット間同期リフレッシュ開始時間（0.1μs）		
CM2402～ CM2403	R	ユニット間同期リフレッシュ終了時間（0.1μs）		
CM2404～ CM2405		ユニット間同期リフレッシュ開始時間最大（0.1μs） （リセットするにはプログラムから0にしてください。）		○
CM2406～ CM2407		ユニット間同期リフレッシュ終了時間最大（0.1μs） （リセットするにはプログラムから0にしてください。）		○
CM2408～ CM2409	R	ユニット間同期リフレッシュ実行時間（0.1μs）		
CM2410～ CM5099	–	システム予約		

CM	属性 R：読出専用 空欄：読書可能	機能		保持 ○：保持
CM5100	R	最新重度エラー詳細	年月（10進数で下二桁が月、百の桁から上が年。例：1210→2012年10月）	○
CM5101	R		日	○
CM5102	R		時	○
CM5103	R		分	○
CM5104	R		秒	○
CM5105	R		エラー番号	○
CM5106	R		有効データ数	○
CM5107	R		詳細情報01	○
CM5108	R		詳細情報02	○
:	R			○
CM5126	R		詳細情報20	○
CM5127～ CM5149	–	システム予約		
CM5150	R	最新軽度エラー詳細	年月（10進数で下二桁が月、百の桁から上が年。例：1210→2012年10月）	○
CM5151	R		日	○
CM5152	R		時	○
CM5153	R		分	○
CM5154	R		秒	○
CM5155	R		エラー番号	○
CM5156	R		有効データ数	○
CM5157	R		詳細情報01	○
CM5158	R		詳細情報02	○
:	R			○
CM5176	R		詳細情報20	○
CM5177～ CM5871	–	システム予約		
CM5872～ CM5873	R	CPUメモリ使用量（32ビット、Byte単位）		○
CM5874～ CM5999	–	システム予約		
CM6000		運転記録機能（ID0）	運転記録0 保存完了コード	
CM6001	–		システム予約	
CM6002			運転記録0 保存カウンタ	
CM6003	–		システム予約	
CM6004	R		運転記録0 保存進捗詳細1	
CM6005	R		運転記録0 保存進捗詳細2	
CM6006	R		運転記録0 保存進捗	
CM6007～ CM6009	–		システム予約	
CM6010～ CM6171	R		運転記録0 保存パス	
CM6172～ CM6199	–		システム予約	
CM6200～ CM6799		運転記録機能（ID1～ID3）		
CM6800～ CM7599	–	システム予約		

ASCIIコード表

ASCIIコード表

下位4ビット ＼ 上位4ビット	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	DLE	SP	0	@	P	`	p				ー	タ	ミ		
1	SOH	DC1	!	1	A	Q	a	q			。	ア	チ	ム		
2	STX	DC2	”	2	B	R	b	r			「	イ	ツ	メ		
3	ETX	DC3	#	3	C	S	c	s			」	ウ	テ	モ		
4	EOT	DC4	$	4	D	T	d	t			、	エ	ト	ヤ		
5	ENQ	NAK	%	5	E	U	e	u			・	オ	ナ	ユ		
6	ACK	SYN	&	6	F	V	f	v			ヲ	カ	ニ	ヨ		
7	BEL	ETB	’	7	G	W	g	w			ァ	キ	ヌ	ラ		
8	BS	CAN	(	8	H	X	h	x			ィ	ク	ネ	リ		
9	HT	EM	)	9	I	Y	i	y			ゥ	ケ	ノ	ル		
A	LF	SUB	*	:	J	Z	j	z			ェ	コ	ハ	レ		
B	VT	ESC	+	;	K	[	k	{			ォ	サ	ヒ	ロ		
C	FF	→	,	<	L	¥	l	|			ャ	シ	フ	ワ		
D	CR	←	―	=	M	]	m	}			ュ	ス	ヘ	ン		
E	SO	↑	.	>	N	^	n	~			ョ	セ	ホ	゛		
F	SI	↓	／	?	O	＿	o	DEL			ッ	ソ	マ	゜		

4 命令語一覧

命令語と、その処理概要を一覧しています。詳しい内容については、各命令語の説明ページを参照してください。
実行条件の記号は、⎍：毎スキャン実行、⎍：立ち上がりで1スキャンだけ実行、⎍：立ち下がりで1スキャンだけ実行をあらわしています。

分類	ニモニック	シンボル	処理概要	実行条件	説明ページ
	ロギング命令				
ロギング	LOGE	n LOGE	指定したロギングIDのロギングを実行許可状態にします。	⎍	5-74
	LOGD	n LOGD	指定したロギングIDのロギングを実行禁止状態にします。	⎍	5-74
	TRGD	n TRGD	指定したIDのロギングデータを取得します。	⎍	5-76
	運転記録命令				
運転記録	TRGR	n TRGR	指定した運転記録IDのデバイスデータを収集します。	⎍	5-106
	カムスイッチ命令				
カムスイッチ	MCMP	MCMP S1 S2 D n	多段コンパレータの動作を行います。	⎍	5-183
	ABSENC	ABSENC S1 S2 D1 D2	アブソリュート式エンコーダを使用してカム動作を行います。	⎍	5-187
	高速処理命令				
割り込み	DI	DI	割り込みプログラムの実行を禁止します。	⎍	4-218
	EI	EI	割り込みプログラムの実行を許可します。	⎍	4-218
	DIC	D DIC	割り込み禁止範囲を設定します。	⎍	4-220
	INT	INT	RETI命令までの割り込みプログラムを実行します。	—	4-222
	RETI	RETI	割り込みプログラムの最後を示します。	—	4-222
	モジュール命令				
モジュール	MDSTRT	MDSTRT S	指定したモジュールの実行を開始します。	⎍	4-115
	MDSTOP	MDSTOP S	指定したモジュールの実行を停止します。	⎍	

分類	ニモニック	シンボル	処理概要	実行条件	説明ページ
	ファンクションブロック命令				
ファンクションブロック	FBCALL	S2 S1 EN ENO 実行するファンクションブロックによる	指定したファンクションブロックを実行します。	⎍	4-151
	FBSTRT	S2 S1 EN ENO 実行するファンクションブロックによる	指定したファンクションブロックを実行します。	⎍	4-151
	FB	S 実行するファンクションブロックによる	指定したファンクションブロックを実行します。	⎍	4-148
	FEND	FEND	ファンクションブロックの実行を停止します。	⎍	4-148
	FUN	S 実行するファンクションブロックによる	指定したファンクションを実行します。	⎍	4-148
	マクロ命令				
マクロ	MCALL	MCALL 実行するマクロによる	指定したサブルーチン型マクロを実行します。	⎍	4-178
	MSTRT	MSTRT 実行するマクロによる	指定した自己保持型マクロを実行します。	⎍	
	MEND	MEND	自己保持型マクロの実行を停止します。	⎍	

5 使用できない命令語一覧

初期化モジュール、ファンクションブロック、マクロ、サブルーチンプログラム、割り込みプログラム内で使用できない命令語を説明します。

● 初期化モジュール、割り込みプログラム（定周期モジュール/ユニット間同期モジュールは除く）

命令の種類		命令語	
共通		@XXXX命令	微分実行型
基本命令	接点命令	LDP/LDF	ロードパルス／ロードパルフ
		ANP/ANF	アンドパルス／アンドパルフ
		ORP/ORF	オアパルス／オアパルフ
		LDPB/LDFB	ロードパルスバー／ロードパルフバー
		ANPB/ANFB	アンドパルスバー／アンドパルフバー
		ORPB/ORFB	オアパルスバー／オアパルフバー
	出力命令	DIFU/DIFD	ディフアップ／ディフダウン
		ONDL/OFDL	オンディレイ／オフディレイ
		SHOT	ワンショット
		FLIK	フリッカ
		ALT	オルタネート
	タイマ･カウンタ命令	TMR/TMH/TMS/TMU	タイマ／高速タイマ／高速1msタイマ／高速10μsタイマ
		C	カウンタ
		OUTC	アウトカウンタ
		ITVL	インターバルタイマ
		UDC	アップダウンカウンタ
		UDT	アップダウンタイマ
	接続/終了命令	MEP/MEF	メップ／メフ
応用命令	シフト命令	SFT	シフト
	ステージ処理命令	W-ON/W-OFF	ウェイトオン／ウェイトオフ
		W-UE/W-DE	ウェイトアップエッジ／ウェイトダウンエッジ
	フロー命令	SBN/RET/SCJ	サブルーチン開始/サブルーチン終了/LABEL命令まで
	ファイルレジスタ設定	FRSTM/FRLDM	ファイルレジスター括保存／ファイルレジスター括読出
演算命令	データ制御命令	RAMP/TPOUT/LLFLT	傾斜信号／時分割比例出力／進み遅れフィルタ
	文字処理命令	RCOM	文字列読み出し
拡張命令	データ処理命令	HKEY/SORTN	16キー入力／バイナリデータ分割ソート
	カムスイッチ命令	ABSENC	アブソリュートエンコーダ
	PID命令	PID/PIDAT	PID／オートチューニング付きPID
	ストレージ命令	MWRIT/MREAD/MFREE/MMKDIR/MPRINT/MREADL/MCOPY/MMOV/MREN/MFREEK/MSTAT	ストレージライト／ストレージリード／ストレージ容量取得／ストレージディレクトリ作成／ストレージ文字列書込み／ストレージ1行読み出し／ストレージファイルコピー／ストレージファイル移動／ストレージファイル名変更／ストレージ空き容量取得／ストレージファイル状態取得
		MRMDIR/MDEL	ストレージフォルダ削除／ストレージファイル削除
	センサ設定命令	SPWR/SPRD/SSVC	センサパラメータ書込／センサパラメータ読出／センササービス実行
	アクセスウィンドウ命令	AWSHOW/AWHIDE	ユーザメッセージ表示／ユーザメッセージ表示解除

● ファンクションブロック

命令の種類		命令語	
応用命令	マクロ命令	MCALL/MSTRT	マクロコール／マクロスタート

※ FEND（ファンクションブロックエンド）命令は、ファンクションブロック内に記述します。ファンクションやモジュールなどほかの種類のプログラムでは使用できません

● ファンクション

命令の種類		命令語	
応用命令	マクロ命令	MCALL/MSTRT	マクロコール／マクロスタート
	ファンクションブロック命令	FBCALL/FBSTRT/FB	ファンクションブロック

● マクロ

命令の種類		命令語	
応用命令	マクロ命令	MCALL/MSTRT	マクロコール／マクロスタート
	ファンクションブロック命令	FBCALL/ FBSTRT/FB/FUN	ファンクションブロック／ファンクション

※ MEND（マクロエンド）命令は、自己保持型マクロ内に記述します。サブルーチン型マクロやモジュールなどほかの種類のプログラムでは使用できません。

● サブルーチンプログラム

命令の種類		命令語	
応用命令	ステージ処理命令	STG/JMP/ENDS	ステージ／ジャンプ／エンドステージ
拡張命令	高速処理命令	INT/RETI	イント／リターンイント

6 ニモニック短縮入力一覧

《KV STUDIO》でニモニック入力する場合に、短縮または代替入力が可能な命令語やデバイスを説明します。

● 命令語

命令語		省略して入力する文字列
接点命令	LD,AND	A,X
	LDB,ANB	B,LDI,LDNOT,ANI,ANDNOT,/ ※3
	ORB	ORI,ORNOT
	LDP,ANP	P
	LDF,ANF	F
ビット接点命令	BLD,BAND	BA
	BLDB,BANB	BB
比較命令	LD=,AND=	=
	LD<,AND<	<
	LD>,AND>	>
	LD<=,AND<=	<=
	LD>=,AND>=	>=
	LD<>,AND<>	<>
出力命令	OUT	O,Y
	OUB	OB
	SET	S
	RES	R,RST
	DIFU	U,PLS
	DIFD	D,PLF
ビット出力命令	BOUT	BO
	BOUB	BOB
	BSET	BS
	BRES	BR
タイマ･カウンタ命令※1	TMR	T,TIM,OUTT
	TMH	TH,TIMH,OUTH
	TMS	TS,TIMHH,OUTS
	TMU	TU,TMUH,OUTH
	C	CNT
	UDC	CNTR
接続／終了命令	END	E,FEND
	ENDH	H
ステージ命令	W-ON	WON
	W-OFF	WOFF
	W-UE	WU, WUE
	W-DE	WD,WDE
フロー命令	MC	IL
	MCR	ILC
バッファメモリ命令	UREAD	FROM
	UWRIT	TO
算術／比較命令	INC	++
	DEC	--
	ROOT	SQRT
論理演算命令	EORA	XOR,XORA
	ENRA	XNORA
データ制御命令	APR	SCL

命令語		省略して入力する文字列
浮動小数点命令	DFLOAT	FLTD,DFLTD,DBL,DBLL
	DINTG	INTD,DINTD,FIXD,FIXLD
	FTODF	ECON
	DFTOF	EDCON
文字処理命令	DASC	DSTR
	RDASC	DVAL
	HASC	HSTR
	RHASC	HVAL
	FASC	FSTR
	RFASC	FVAL
シンプル演算命令	CAL+	+
	CAL-	-※2
	CAL*	*
	CAL/	/ ※3
	CAL&	&,WAND
	CAL｜	｜,WOR
	CAL~	~,WXOR
	CAL^	^,NOT,CML
	CAL>>	>>,SFR
	CAL<<	<<,SFL
高速処理命令	INT	I
	RETI	IRET

※1 タイマ･カウンタ命令は、命令語とデバイス番号の間のスペースなしでも入力できます。
（例）TMR 0002 DM00001 → T2 D1
※2 [ラダー編集]ウィンドウで直接「-」を入力すると、ヨコ接続線が入力されます。
「直接入力」ダイアログボックス（Esc）上で「-」を入力すると、「CAL-命令」が入力されます。
※3 ラダー編集]ウィンドウで直接「/」を入力すると、「LDB命令」が入力されます。
「直接入力」ダイアログボックス（Esc）上で「/」を入力すると、「CAL/命令」が入力されます。

● デバイス名

デバイス名	短縮入力
R	なし（番号のみ）
MR	M
LR	L
DM	D
EM	E
FM	F

● デバイス番号と定数

ゼロサプレスできます。

7 ショートカットキー一覧

各機能の機能を実行するショートカットキー一覧です。

ユニットエディタで有効なショートカットキー

● ファイル

インポート	Ctrl + I
閉じる	Alt + F4

● 編　集

切り取り	Ctrl + X
コピー	Ctrl + C
貼り付け	Ctrl + V
削除	Delete
ユニットコメント	Ctrl + Shift + U
ブロックコメント	Ctrl + Shift + B
システムコメント	Ctrl + Shift + S
ユニット予約	Ctrl + R
リレー番号の一括入力	Ctrl + A

● 変　換

エラー行ジャンプ	F4
リレー/DM自動割付	F5
デバイスチェック	F6

● 子機設定

子機設定	Ctrl + N
子機情報のインポート	Ctrl + K
KLユニット情報をコピー	Ctrl + Alt + C
KLユニット情報を貼り付け	Ctrl + Alt + V
KLアドレス設定	Ctrl + E

● 表　示

ユニット設定の表示	↵
内蔵機能モニタ	Ctrl + Alt + U
カスタムモニタ	Ctrl + Alt + M
バッファメモリモニタ	Ctrl + Alt + R

● ウインドウ

ユニット配置エリア	Alt + U
ユニット選択	Alt + 1
ユニット設定	Alt + 2
リフレッシュ設定	Alt + 3
メッセージエリア	Alt + 4

● ヘルプ

ユニットエディタの使い方	F1

● ユニット配置エリアでの操作

カーソル移動	←・→・↑・↓
行頭へのカーソル移動	Home
行末へのカーソル移動	End
配置したユニットの削除	Delete
複数選択	Shift + ←・→・↑・↓

● ユニット選択ウインドウでの操作

カーソル移動	←・→・↑・↓
選択したユニットの追加	↵

● ユニット設定ウインドウでの操作

カーソル移動	↑・↓
表示ユニット移動	Ctrl + ←・→・↑・↓
入力	Tab
入力/確定	↵
キャンセル	Esc

● メッセージエリアでの操作

カーソル移動	↑・↓

ユニットエディタ「子機設定画面」で有効なショートカットキー

● ファイル

子機情報のインポート	Ctrl + K
閉じる	Alt + F4

● 編　集

切り取り	Ctrl + X
コピー	Ctrl + C
貼り付け	Ctrl + V
削除	Delete
ユニット/子機コメント入力	Ctrl + Shift + U
ユニット子機予約	Ctrl + R

● 変　換

エラー行ジャンプ	F4
アドレス自動割付	F5
アドレスチェック	F6

● 表　示

ユニット/子機設定内容	↵

● ウィンドウ

子機配置エリア	Alt + C
ユニット選択ウィンドウ	Alt + 1
ユニット設定ウィンドウ	Alt + 2
アドレス設定ウィンドウ	Alt + 3
メッセージエリア	Alt + 4

● ヘルプ

ユニットエディタの使い方	F1

● 子機配置エリアでの操作

カーソル移動	↑・↓
選択したユニットのユニット設定ウィンドウを表示	↵
配置したユニットの削除	Delete

● ユニット選択ウィンドウでの操作

カーソル移動	←・→・↑・↓
選択したユニットの追加	↵

● ユニット設定ウィンドウでの操作

カーソル移動	↑・↓
入力	Tab
入力/確定	↵
キャンセル	Esc

● アドレス設定ウィンドウでの操作

カーソル移動	↑・↓

KV STUDIOで有効なショートカットキー

◆の付いた項目は、ショートカットキーの割り付け変更ができます。

初期画面で有効なショートカットキー

● ファイル

プロジェクトの新規作成	Ctrl + N
プロジェクトを開く	Ctrl + O

● 表示

ワークスペース	Alt + 1

● モニタ/シミュレータ

PLC本体モニタ	Ctrl + F1
PLCから読み出し	Ctrl + F5

エディタモードで有効なショートカットキー

● ファイル

プロジェクトの新規作成	Ctrl + N
プロジェクトを開く	Ctrl + O
プロジェクトを上書き保存	Ctrl + S
印　刷	Ctrl + P

● 編　集

元に戻す		Ctrl + Z
やり直し		Ctrl + Y
切り取り		Ctrl + X
コピー		Ctrl + C
貼り付け		Ctrl + V
複数貼り付け		Ctrl + Shift + V
検索		Ctrl + F
オペランド置換		Ctrl + H
クロスリファレンス作成		(スペース)
クロスリファレンス行コメント一覧作成		Ctrl + L
クロスリファレンス前へ◆		Shift + F3
クロスリファレンス次へ◆		F3
ジャンプ	行/行ステップ番号	Ctrl + Shift + G
	先頭	Ctrl + Home
	末尾	Ctrl + End
	前ブロック	Ctrl + ↑
	次ブロック	Ctrl + ↓
	前シンボル	Shift + Tab
	次シンボル	Tab
	行頭	Home
	行末	End
	変更 前ブロック	Ctrl + Alt + ←
	変更 次ブロック	Ctrl + Alt + →
挿入	セル	Ins
	空行	Shift + ↵ ／ Shift + Ins
	行コメント	Ctrl + ↵
	折り返しラダー	Ctrl + Shift + F8
行削除		Shift + Delete
行の無効化－設定		Ctrl + Q
行の無効化－解除		Ctrl + W
ブックマーク	挿入	Ctrl + Shift + Ins
	設定	Ctrl + Shift + ↵
	解除	Ctrl + Shift + Delete
行選択		Ctrl + → (Shift + ←)※1 (Shift + →)※2
ラダーの整列		Ctrl + Shift + F
リスト編集		Ctrl + D
オペランド変更		↵
デバイスコメント変更		Ctrl + M
接続線の編集	タテ接続線の入力◆	F8
	ヨコ接続線の入力◆	F9 (−)
	行末までのヨコ接続線	Alt + ↵ (Ctrl + Tab)

接続線の編集	タテ接続線の削除◆	Shift + F8
	ヨコ接続線の削除◆	Shift + F9
	タテ接続線の入力･削除	Alt + ↑ / Alt + ↓
	ヨコ接続線の入力･削除	Alt + ← / Alt + →
カーソル位置のデバイスのクロスリファレンスを表示		(スペース)
カーソル位置の左の命令を削除		Backspace
カーソル位置の命令を削除		Delete
カーソル位置に命令を直接入力		Esc
カーソル位置のシンボルを右にずらす		Ins
a接点入力◆		F5
b接点入力◆		Shift + F5
a接点OR入力◆		F4
b接点OR入力◆		Shift + F4
a接点コイル入力◆		F7
b接点コイル入力◆		Shift + F7
SET◆		Alt + F1
RES◆		Alt + F2
TMR◆		Alt + F3
TMS◆		Alt + F5
C◆		Alt + 0
DIFU◆		Alt + F8
DIFD◆		Alt + F9
MOV◆		Alt + F10
LDA◆		Alt + F11
STA◆		Alt + F12
LDP◆		Alt + F7
LDF◆		Alt + Shift + F7
ORP◆		Alt + Shift + F8
ORF◆		Alt + Shift + F9
MEP◆		Alt + Shift + F5
MEF		Alt + Ctrl + F5
INV		Alt + Ctrl + F10

a-b接点変更、サフィックス変更、B接点入力	/
立ち上がり/立ち下がり変更	Ctrl + /
直接入力/現在値変更◆	F2
次のペイン◆	Ctrl + Shift + Alt + F11 ※
前のペイン◆	Ctrl + Shift + Alt + F12 ※
デバイス番号のインクリメント	Ctrl + Shift + ↑
デバイス番号のデクリメント	Ctrl + Shift + ↓

※ ラダー編集ウインドウを分割表示している場合のみ有効です。

● 表　示

ワークスペース	Alt + 1
アウトプットウインドウ	Alt + 2
命令語パレット◆	Shift + Ctrl + P
オペランド編集ダイアログ	↵ ※
コメント編集ウインドウ	Ctrl + F7
デバイス使用リスト	Ctrl + E
表示拡大	Ctrl + PageUp
表示縮小	Ctrl + PageDown
変数表示	Ctrl + Backspace
コメント表示◆	Ctrl + (スペース)
ユニット用デバイス表示	Shift + (スペース)
編集ウインドウの切り替え	Alt + F6

※ 選択中のセルに命令語が入力されている場合のみ有効です。

● ST/スクリプト

エリアスクリプト挿入	Ctrl + R
ボックススクリプト挿入	Ctrl + B
エリア型ST挿入	Ctrl + Shift + A
ボックス型ST挿入	Ctrl + Shift + B
選択ST/スクリプトを変換	Ctrl + T
変換ラダーの表示/非表示	Ctrl + G
ST/スクリプト関数一覧	Ctrl + (スペース)
ST/スクリプト関数ヒント	Ctrl + Shift + (スペース)

● 変　換

変換◆	Ctrl + F9

※1 カーソルが左端(第1カラム)にある場合のみ有効です。
※2 カーソルが右端(第10カラム)にある場合のみ有効です。

● モニタ/シミュレータ

モニタモード◆	Ctrl + Shift + F3
PLC転送→モニタモード◆	Ctrl + F8
PLC読出→モニタモード	Ctrl + F1
PLC読出◆	Ctrl + F5
オンラインエディット開始◆	F10
シミュレータ	Ctrl + F2

● ツール

ユニットエディタ	Ctrl + U

● ヘルプ

命令語ヘルプ	F1
命令語リファレンスマニュアル	Ctrl + 1

モニタモードで有効なショートカットキー

● 編　集

検索		Ctrl + F ／ ↵
ジャンプ	行/行ステップ番号	Ctrl + Shift + G
	先頭	Ctrl + Home
	末尾	Ctrl + End
	前ブロック	Ctrl + ↑
	次ブロック	Ctrl + ↓
	前シンボル	Shift + Tab
	次シンボル	Tab
	行頭	Home
	行末	End
	変更 前ブロック	Ctrl + Alt + ←
	変更 次ブロック	Ctrl + Alt + →
接点/コイル		F12
クロスリファレンス作成		(スペース)
クロスリファレンス前へ		Shift + F3
クロスリファレンス次へ		F3
クロスリファレンス行コメント一覧作成		Ctrl + L
行選択		Ctrl + →
接点のON/OFF切替		/
セット登録		Ctrl + Shift + J
リセット登録		Ctrl + Shift + K
解除		Ctrl + Shift + L

● 表　示

ワークスペース	Alt + 1
アウトプットウインドウ	Alt + 2
表示拡大	Ctrl + PageUp
表示縮小	Ctrl + PageDown
変数表示	Ctrl + Backspace
コメント表示◆	Ctrl + (スペース)
ユニット用デバイス表示	Shift + (スペース)
デバイス値変更	F2
編集ウインドウの切り替え	Alt + F6

● モニタ/シミュレータ

エディタへ戻る◆		Ctrl + F1
モニタ開始		F4
モニタ停止		Shift + F4
オンラインエディット	開始	F10
	設定	F12
シミュレータ		Ctrl + F2
ウォッチウインドウの表示/非表示		Alt + 3

● デバッグ

1スキャン実行	F9
1ステップ実行	F8

● ヘルプ

命令語ヘルプ	F1
命令語リファレンスマニュアル	Ctrl + 1

シミュレータモードで有効なショートカットキー

● 編　集

検索		Ctrl + F (↵)
行選択		Ctrl + →
ジャンプ	先頭	Ctrl + Home
	末尾	Ctrl + End
	前ブロック	Ctrl + ↑
	次ブロック	Ctrl + ↓
	前シンボル	Shift + Tab
	次シンボル	Tab
	行頭	Home
	行末	End
	変更 前ブロック	Ctrl + Alt + ←
	変更 次ブロック	Ctrl + Alt + →
	接点/コイル	F12
クロスリファレンス作成		(スペース)
クロスリファレンス前へ		Shift + F3
クロスリファレンス次へ		F3
クロスリファレンス 行コメント一覧作成		Ctrl + L
接点のON/OFF切替		/

● 表　示

ワークスペース	Alt + 1
アウトプットウインドウ	Alt + 2
ラダー表示拡大	Ctrl + PageUp
ラダー表示縮小	Ctrl + PageDown
変数表示	Ctrl + Backspace
コメント表示◆	Ctrl + (スペース)
ユニット用デバイス表示	Shift + (スペース)
デバイス値変更	F2

● モニタ/シミュレータ

エディタへ戻る◆	Ctrl + F1
モニタモード◆	Ctrl + Shift + F3
オンラインエディット 開始	F10

● ヘルプ

命令語ヘルプ	F1
命令語リファレンス マニュアル	Ctrl + 1

オンラインエディット中に有効なショートカットキー

● 編　集

元に戻す	Ctrl + Z
やり直し	Ctrl + Y
切り取り	Ctrl + X
コピー	Ctrl + C
貼り付け	Ctrl + V
複数貼り付け	Ctrl + Shift + V
検索	Ctrl + F
オペランド置換	Ctrl + H
オンラインエディット 編集箇所検索	Ctrl + . , Ctrl + O

ジャンプ	行/行ステップ番号	Ctrl + Shift + G
	先頭	Ctrl + Home
	末尾	Ctrl + End
	前ブロック	Ctrl + ↑
	次ブロック	Ctrl + ↓
	前シンボル	Shift + Tab
	次シンボル	Tab
	行頭	Home
	行末	End
	変更 前ブロック	Ctrl + Alt + ←
	変更 次ブロック	Ctrl + Alt + →
	接点/コイル	F12
クロスリファレンス作成		(スペース)
クロスリファレンス前へ◆		Shift + F3
クロスリファレンス次へ◆		F3
クロスリファレンス 行コメント一覧作成		Ctrl + L
セル挿入		Ins

空行挿入		Shift + ↵ ／ Shift + Ins
行削除		Shift + Delete
行コメントの挿入		Ctrl + ↵
行の無効化－設定		Ctrl + Q
行の無効化－解除		Ctrl + W
ブックマーク	挿入	Ctrl + Shift + Ins
	設定	Ctrl + Shift + ↵
	解除	Ctrl + Shift + Delete
行選択		Ctrl + →
接続線の編集	タテ接続線の入力◆	F8
	ヨコ接続線の入力◆	F9 (−)
	行末までのヨコ接続線	Alt + ↵ (Ctrl + Tab)
	タテ接続線の削除◆	Shift + F8
接続線の編集	ヨコ接続線の削除◆	Shift + F9 (−)
	タテ接続線の入力･削除	Alt + ↑ / Alt + ↓
	ヨコ接続線の入力･削除	Alt + ← / Alt + →
折り返しラダー挿入		Ctrl + Shift + F8
ラダー整列		Ctrl + Shift + F
リスト編集		Alt + D
オペランド変更		↵
デバイスコメント変更		Alt + M
カーソル位置の左の命令を削除		Backspace
カーソル位置の命令を削除		Delete
カーソル位置に命令を直接入力		Esc
a接点入力◆		F5
b接点入力◆		Shift + F5
a接点OR入力◆		F4
b接点OR入力◆		Shift + F4
a接点コイル入力◆		F7
b接点コイル入力◆		Shift + F7
SET◆		Alt + F1
RES◆		Alt + F2
TMR◆		Alt + F3
TMS◆		Alt + F5
C◆		Alt + 0
LDP◆		Alt + F7

DIFU◆	Alt + F8
DIFD◆	Alt + F9
LDF◆	Alt + Shift + F7
ORP◆	Alt + Shift + F8
ORF◆	Alt + Shift + F9
MOV◆	Alt + F10
LDA◆	Alt + F11
STA◆	Alt + F12
MEP◆	Alt + Shift + F5
MEF	Alt + Ctrl + F5
INV	Alt + Ctrl + F10
a-b接点変更、サフィックス変更、B接点入力	/
立ち上がり/立ち下がり変更	Ctrl + /
直接入力/現在値変更	F2
セット登録	Ctrl + Shift + J
リセット登録	Ctrl + Shift + K
解除	Ctrl + Shift + L

● 表　示

ワークスペース	Alt + 1
アウトプットウインドウ	Alt + 2
命令語パレット◆	Shift + Ctrl + P
オペランド編集ダイアログ	↵ ※
コメント編集ウインドウ	Ctrl + F7
デバイス使用リスト	Ctrl + E
表示拡大	Ctrl + PageUp
表示縮小	Ctrl + PageDown
変数表示	Ctrl + Backspace
コメント表示◆	Ctrl + (スペース)
ユニット用デバイス表示	Shift + (スペース)

※選択中のセルに命令語が入力されている場合のみ有効です。

● ST/スクリプト

エリアスクリプト挿入	Ctrl + R
ボックススクリプト挿入	Ctrl + B
エリア型ST挿入	Ctrl + Shift + A
ボックス型ST挿入	Ctrl + Shift + B
選択ST/スクリプトを変換	Ctrl + T

● モニタ/シミュレータ

エディタへ戻る◆	Ctrl + F1
モニタモード◆	Ctrl + Shift + F3
オンラインエディット転送	F11
シミュレータ	Ctrl + F2
ウォッチウインドウの表示/非表示	Alt + 3

● ヘルプ

命令語ヘルプ	F1
命令語リファレンスマニュアル	Ctrl + 1

リプレイモードで有効なショートカットキー

● 編　集

検索		Ctrl + F ／ ↵
ジャンプ	行/行ステップ番号	Ctrl + Shift + G
	先頭	Ctrl + Home
	末尾	Ctrl + End
	前ブロック	Ctrl + ↑
	次ブロック	Ctrl + ↓
	前シンボル	Shift + Tab
	次シンボル	Tab
	行頭	Home
	行末	End
	変更 前ブロック	Ctrl + Alt + ←
	変更 次ブロック	Ctrl + Alt + →
	接点/コイル	F12
クロスリファレンス作成		(スペース)
クロスリファレンス前へ		Shift + F3
クロスリファレンス次へ		F3
クロスリファレンス行コメント一覧作成		Ctrl + L
行選択		Ctrl + →

● 表　示

ワークスペース	Alt + 1
アウトプットウィンドウ	Alt + 2
表示拡大	Ctrl + PageUp
表示縮小	Ctrl + PageDown
表示拡大･縮小	Alt +ホイール Ctrl +ホイール
変数表示	Ctrl + Backspace
コメント表示	Ctrl + (スペース)
ユニット用デバイス表示	Shift + (スペース)
編集ウインドウの切り替え	Alt + F6

● モニタ/シミュレータ

エディタへ戻る	Ctrl + F1
モニタ開始	F4
モニタ停止	Shift + F4
オンラインエディット開始	F10
シミュレータ	Ctrl + F2
ウォッチウィンドウの表示/非表示	Alt + 3

● ヘルプ

命令語ヘルプ	F1
命令語リファレンスマニュアル	F1 + 1

GPPQ/GPPA形式のショートカット一覧

各ショートカットは、キーの割り付けが変更できます。

a接点入力	F5	
b接点入力	GPPQ形式	F6
	GPPA形式	Shift + F5
a接点OR入力	GPPQ形式	Shift + F5
	GPPA形式	F6
b接点OR入力	Shift + F6	
コイル入力	F7	
b接点コイル入力	未割付	
タテ接続線の入力	GPPQ形式	Shift + F9
	GPPA形式	F10
タテ接続線の削除	Ctrl + F10	
ヨコ接続線の入力	F9	
ヨコ接続線の削除	Ctrl + F9	
SET	未割付	
RES	未割付	
TMR	未割付	
TMS	未割付	
C	未割付	
LDP	Shift + F7	
LDF	Shift + F8	
ORP	Alt + F7	
ORF	Alt + F8	
DIFU	未割付	
DIFD	未割付	
MOV	未割付	
LDA	未割付	
STA	未割付	
MEP	Alt + F5	
接続線を引く/消す(右)	Alt + →	
接続線を引く/消す(左)	Alt + ←	
接続線を引く/消す(下)	Alt + ↓	
接続線を引く/消す(上)	Alt + ↑	
変換	Ctrl + Alt + F4 / F4	
直接入力/現在値変更	未割付	

クロスリファレンス 前へ	未割付
クロスリファレンス 次へ	未割付
PLC転送→モニタモード	F3
PLC読出	未割付
オンラインエディット開始	Shift + F3
エディタへ戻る	F2
次のペイン	Ctrl + Shift + Alt + F11
前のペイン	Ctrl + Shift + Alt + F12
モニタモード	F3
コメント表示	Ctrl + F5
命令語パレット	Ctrl + Shift + P

CX形式のショートカット一覧

各ショートカットは、キーの割り付けが変更できます。

a接点入力	C
b接点入力	未割付
a接点OR入力	W
b接点OR入力	X
コイル入力	O
b接点コイル入力	Q
SET	Shift + F8
RES	Shift + F9
TMR	Ctrl + Alt + O
TMS	未割付
C	未割付
LDP	未割付
LDF	未割付
ORP	未割付
ORF	未割付
DIFU	未割付
DIFD	未割付
MOV	未割付
LDA	未割付
STA	未割付
MEP	Shift + Alt + F5
接続線を引く/消す(右)	H
接続線を引く/消す(左)	未割付
接続線を引く/消す(下)	V
接続線を引く/消す(上)	U
変換	F7
直接入力/現在値変更	I
クロスリファレンス 前へ	B
クロスリファレンス 次へ	N
PLC転送→モニタモード	F3
PLC読出	Ctrl + F5
オンラインエディット開始	F10

エディタへ戻る	F2
次のペイン	Ctrl + Shift + Alt + F11
前のペイン	Ctrl + Shift + Alt + F12
モニタモード	Ctrl + Shift + F3
コメント表示	Alt + Y
命令語パレット	Ctrl + Shift + P

Logix形式のショートカット一覧

各ショートカットは、キーの割り当てが変更できます。

a接点入力	F5
b接点入力	Shift + F5
a接点OR入力	F4
b接点OR入力	Shift + F4
コイル入力	F7
b接点コイル入力	Shift + F7
SET	Alt + F1
RES	Alt + F2
TMR	Alt + F3
TMS	Alt + F5
C	Alt + 4
LDP	Alt + F7
LDF	Alt + Shift + F7
ORP	Alt + Shift + F8
ORF	Alt + Shift + F9
DIFU	Alt + F8
DIFD	Alt + F9
MOV	Alt + F10
LDA	Alt + F11
STA	Alt + F12
MEP	Alt + Shift + F5
MEF	Ctrl + Alt + F5
a-b接点変更	Ctrl + T
リスト編集	Ctrl + Shift + D
接続線を引く/消す(右)	Alt + →
接続線を引く/消す(左)	Alt + ←
接続線を引く/消す(下)	Alt + ↓
接続線を引く/消す(上)	Alt + ↑
行末までの横接続線	Ctrl + Alt + ↵

変換	Ctrl + F9
直接入力/現在値変更	F2
クロスリファレンス	Ctrl + E
クロスリファレンス前へ	Shift + F3
クロスリファレンス次へ	F3
PLC転送→モニタモード	Ctrl + F8
PLC読出	Ctrl + F5
オンラインエディット開始	F10
エディタへ戻る	Ctrl + F1
命令語/マクロ/パックパレット	Alt + Ins
行コメントの挿入	Ctrl + D
ウィンドウの切換	Ctrl + Tab
ウィンドウの切換逆順	Ctrl + Shift + Tab
変数編集ウィンドウ	Ctrl + W
行･ステップへのジャンプ	Ctrl + G
空行の挿入	Ctrl + R
行の無効化設定	Ctrl + Q
行の無効化解除	Ctrl + Shift + W
ワークスペースの表示	Alt + 0
アウトプットウィンドウ表示	Alt + 1
検索結果ウィンドウ表示	Alt + 2
デバイス使用リスト表示	Ctrl + Shift + E
エリアスクリプト挿入	Ctrl + A
ボックススクリプト挿入	Ctrl + B
エリア型ST挿入	Ctrl + Shift + A
ボックス型ST挿入	Ctrl + Shift + B

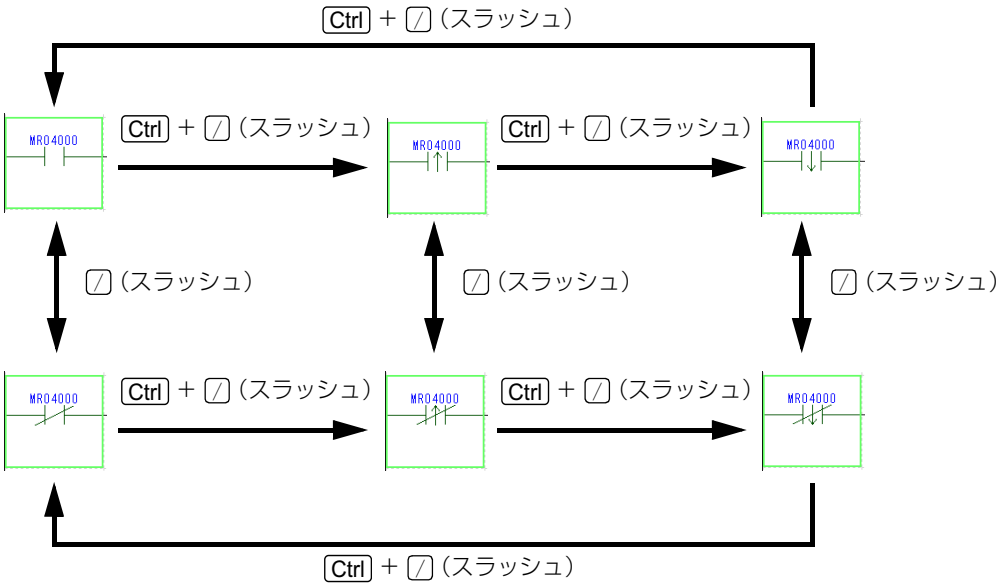
選択ST/スクリプトを変換	Ctrl + Shift + T
変換ラダーの表示/非表示	Ctrl + Shift + G
連続ステップ実行	Shift + F8
1ステップ実行	F8
逆連続ステップ実行	Shift + F7
逆1ステップ実行	F7
一時停止	Shift + F6

/キーの使い方

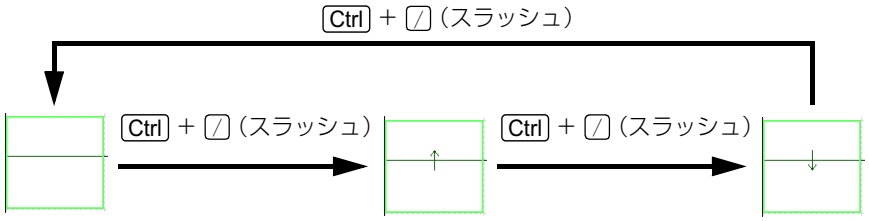
● /キーを押すと、カーソル位置の設定が入れ替わります。

変更できる設定は次のとおりです。

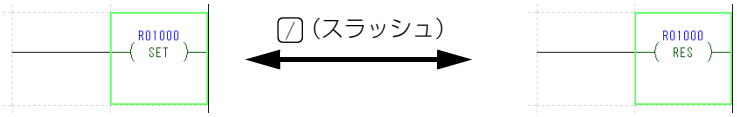
•a接点、b接点　　立ち上がり、立ち下がり



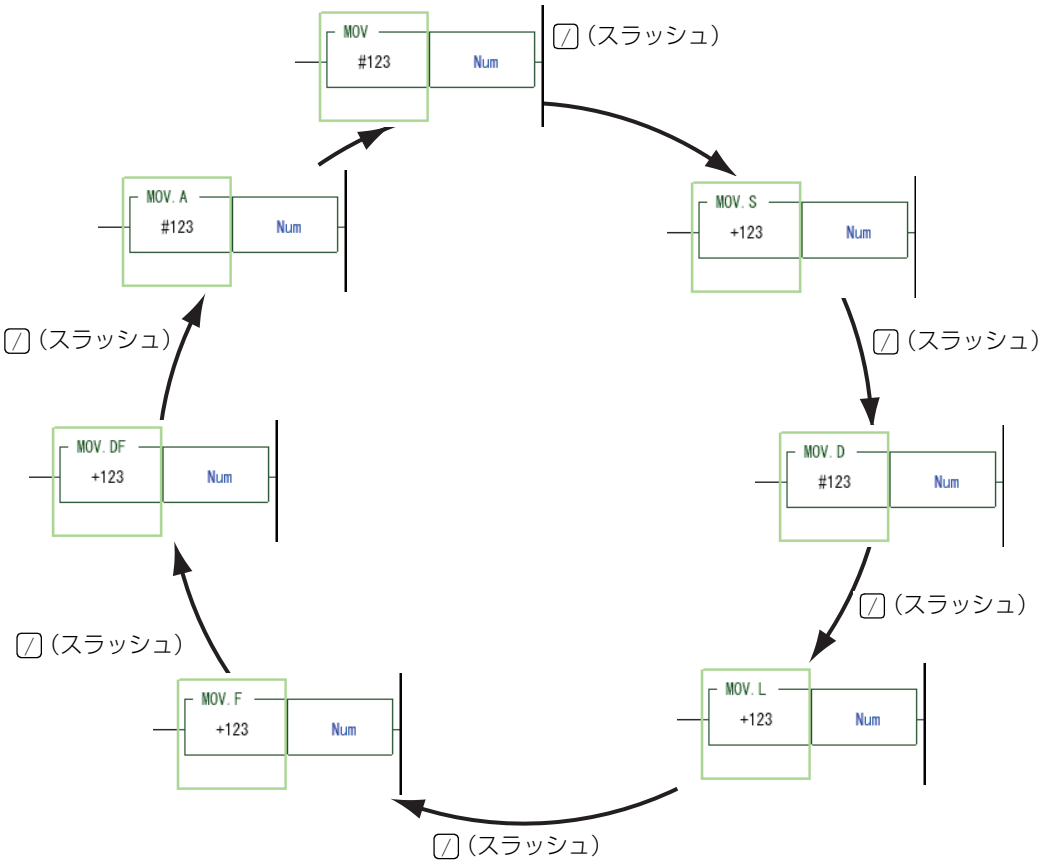
ポイント　Ctrl + / キーを押すと次のように変化します。



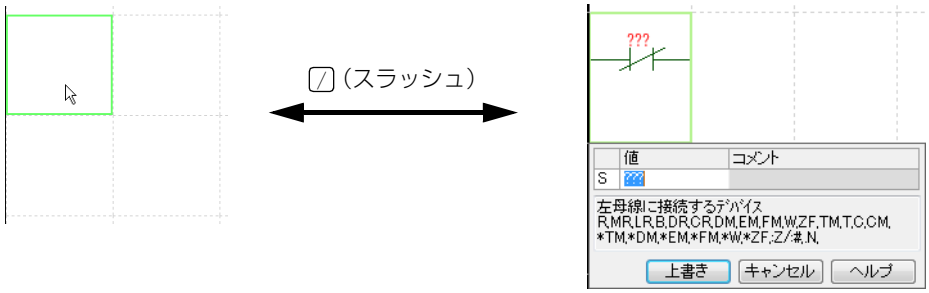
•SET、RES



•サフィックス



● 命令語のないセル上でキーを押すと、b接点が入力されます。



※ 直接入力ダイアログでのみ入力可能です。

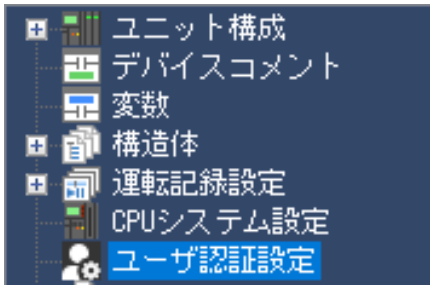
8 ユーザ認証機能

ユーザ認証設定を選択すると、システムプログラム更新やプログラム転送などの機能を登録されたユーザのみ実行できるように設定することができます。

ポイント **ユーザ認証機能は、KV STUDIO Ver.11.63以降を使用した場合に表示されます。**

1 ワークスペースでユーザ認証設定を選択します。

メニュー画面が表示されます。



2 管理者を登録します。

すでに登録済みの場合、次の手順に進んでください。

3 アクセスレベルごとの権限を設定し、各ユーザ名に対してアクセスレベルを設定します。

ポイント **ユーザ認証設定を有効にするには、設定後にKV-8000に転送を行う必要があります。**

ユーザ認証設定の仕様

項　目	設定範囲	備考
ユーザ数	1～128人	1人以上の管理者を設定してください
ユーザ名長	1～32文字	同じユーザ名は設定できません。
アクセスレベル※1	管理者/設計者/ユーザ	
パスワード	8～64文字	小文字･大文字･数字･記号で設定します。 パスワード更新時、同じパスワードは設定できません。

※1　アクセスレベルの詳細は以下の通りです。

アクセスレベル	操作権限
管理者	ユーザ認証設定更新を含む全権限※2
設計者	ユーザ認証設定更新以外の全権限※3
ユーザ	設定により権限を変更できます。※3
認証なし	設定により権限を変更できます。 ユーザ名、パスワード認証を入力しないで使用します。

※2　権限の詳細は以下の通りです。

アクセス権限	認証対象
システムプログラム更新	KV STUDIOからのシステムプログラム更新
ユーザ認証設定更新	KV STUDIOからの PLC転送およびユーザ認証設定更新
PLC転送	KV STUDIOからの PLC転送および照合/同期からの変更
オンラインエディット転送	･KV STUDIOからのオンラインエディット転送 ･KV STUDIOからのT/Cの設定値更新
PLCクリア	KV STUDIOからのPLCクリア
履歴クリア	KV STUDIOのイベント/エラー/モニタの履歴クリア KV STUDIOの登録モニタからCRを指定した履歴クリア
PROG/RUN切り替え	KV STUDIOからの変更
カレンダタイマ設定	KV STUDIOからのカレンダタイマ設定

※3　ユーザ認証設定は自身のパスワードのみ更新でき、PLC照合/同期からのみユーザ認証設定を転送できます。

ポイント **ユーザ認証機能には、オプションで、パスワードの有効期間を設定したり、パスワードに2種類以上の文字種を必要とする設定にしたり、ログインタイムアウト時間の設定を行ったりする機能もオプションで用意しています。**

9 使用できない文字一覧

プロジェクト名、モジュール名、ファンクションブロック名、マクロ名、変数、構造体、デバイスコメントで使用できない文字を説明します。

● プロジェクト名に使用できない文字

使用できない文字	! ″ # $ % & ‘ * , . / : ; < = > ?
	@ ¥ ^ ` | ~（スペース） _

● プログラム名に使用できない文字

使用できない文字	! ″ # $ % & ’ ( ) * , . / : ; < = > ?
	@ [¥] ^ ` { | } ~ (スペース) (__)※1 + −
先頭に使用できない文字	0 1 2 3 4 5 6 7 8 9 _
末尾に使用できない文字または文字列	_ _0 _1 _2 _3 _4 _5 _6 _7 _8 _9
使用できない文字列	ｸﾞﾛｰﾊﾞﾙ グローバル Global 全局 全域 글로벌 FUN_ENO

※1 CPU機能バージョン2.1以降のKV-8000シリーズでは使用できます。

● ファンクション/ファンクションブロック/構造体名に使用できない文字

使用できない文字	! ″ # $ % & ‘ * , . / : ; < = > ? @ ¥
	^ ‘ | ~ (スペース) (__) + −
先頭に使用できない文字	0 1 2 3 4 5 6 7 8 9 _ ※1
末尾に使用できない文字または文字列※2	_ _0 _1 _2 _3 _4 _5 _6 _7 _8 _9
使用できない文字列	ｸﾞﾛｰﾊﾞﾙ グローバル Global 全局 全域 글로벌 • TIMER 定时器 定時器 타이머 COUNTER 计数器 計數器 카운터 FUN_ENO • IEC61131-3で規定されたキーワード （キーワードの内容は 変数に使用できない文字 を参照してください）

※1 CPU機能バージョン2.1以降のKV-8000シリーズでは使用できます。
※2 CPU機能バージョン2.1以降のKV-8000シリーズでは末尾の_(数字)は使用できます。

● 変数に使用できない文字

使用できない文字	! ″ # $ % & ’ ( ) * + , − . / : ; < =
	> ? @ [¥] ^ ` { | } ~（スペース）
先頭に使用できない文字	0 1 2 3 4 5 6 7 8 9 _
使用できない文字列	ｸﾞﾛｰﾊﾞﾙ グローバル Global 全局
	（命令語と同じ名前）
	（デバイス名）
	• アルファベット1文字＋ R または M ＋数字 • アルファベット1文字＋数字 • X ＋（A ～ F） • Y ＋（A ～ F）

使用できない文字列	（ST言語関数と同じ名前）
	（KVスクリプト関数と同じ名前）
	• KVスクリプトで規定された以下のキーワード !!,#PRAGMA,',(,),*,*=,+,+=,-,-=,.B,.D,.DF,.F,.L,.S,.T,.U,/,/=,<,<=,<>,=,>,>=,AND,AS,B,BREAK,C,CASE,CM,CR,CTC,CTH,DIM,DM,DO,DOUBLE,DR,ELSE,ELSE IF,EM,END,END IF, END SELECT,END TYPE,END WHILE, FALSE,FM,FOR,IF,IS,LR,MC,MCR,MOD,MR,NEXT,NOT,OFF,ON,OR,P,R,REM,SELECT,SELECT CASE,STEP,STRWORK,T, THEN,TM,TO,TRUE,TYPE,UM,UNTIL,UR,UV,V,W,WHILE,XOR, Z,ZF,[,\\,\\A,\\B,\\F,\\N,\\R,\\T,\\V,]
	• IEC61131-3で規定された以下のキーワード TRUE, FALSE TIME, DATE,TIME_OF_DAY, TOD, DATE_AND_TIME, DT BOOL, SINT, INT, DINT, LINT, USINT, UINT, UDINT, ULINT, REAL, LREAL, STRING, WSTRING, BYTE, WORD, DWORD, LWORD ANY, ANY_DERIVED, ANY_ELEMENTARY, ANY_MAGNITUDE, ANY_NUM, ANY_REAL, ANY_INT, ANY_BIT, ANY_STRING, ANY_DATE ACTION, END_ACTION, ARRAY, OF, AT, CASE, END_CASE, CONFIGURATION, END_CONFIGURATION, CONSTANT, EN, ENO, EXIT, F_EDGE, FOR, TO, BY, DO, END_FOR, FUNCTION, END_FUNCTION, FUNCTION_BLOCK, END_FUNCTION_BLOCK, IF, THEN, ELSIF, ELSE, END_IF, INITIAL_STEP, END_STEP, NOT, MOD, AND, XOR, OR, PROGRAM, WITH, END_PROGRAM, R_EDGE, READ_ONLY, READ_WRITE, REPEAT, UNTIL, END_REPEAT, RESOURCE, END_RESOURCE, RETAIN, NON_RETAIN, RETURN, STEP, STRUCT, END_STRUCT, TASK, TRANSITION, FROM, END_TRANSITION, TYPE, END_TYPE, VAR,VAR_INPUT, VAR_OUTPUT, VAR_IN_OUT, VAR_TEMP, VAR_EXTERNAL, VAR_ACCESS, VAR_CONFIG,VAR_GLOBAL, END_VAR, WHILE, DO, END_WHILE • その他（IEC61131-3の仕様） • 変数名の中に、2つ以上連続した"_"が現れてはならない。 "a_b_c"はできますが、"a__bc"は使用できません。

● 構造体のメンバ名に使用できない文字

使用できない文字	( ) * / + − < > = & ! ¥ ″ # $ % % & | ' ~ ` @ [ ] { } : ; , . ? ^ (半角スペース) (全角スペース) (__)※1
先頭に使用できない文字	0 1 2 3 4 5 6 7 8 9 _ ※1
末尾に使用できない文字または文字列※2	_ _0 _1 _2 _3 _4 _5 _6 _7 _8 _9
使用できない文字列	ｸﾞﾛｰﾊﾞﾙ グローバル Global 全局 全域 글로벌 • 数値と区別できない文字列 　• K 　• K + 数値 　• K + 数値 + サフィックス(U/S/D/L/F/DF) 　• K + 数値 + E 　• K + 数値 + E + 数値 　• K + 数値 + E + 数値 + サフィックス(F/DF) 　• H + 16進数値(10進数値、もしくはABCDEFの文字列) • KVスクリプトで規定されたキーワード （キーワードの内容は変数名に使用できない文字を参照してください） • IEC61131-3で規定されたキーワード （キーワードの内容は変数名に使用できない文字を参照してください）

※1 CPU機能バージョン2.1以降のKV-8000シリーズでは使用できます。
※2 CPU機能バージョン2.1以降のKV-8000シリーズでは末尾の_(数字)は使用できます。

● デバイスコメントに使用できない文字

使用できない文字	,

10 既存のKVシリーズ拡張ユニット使用時の注意事項について

対応機器一覧

既存のKVシリーズ拡張ユニットに対するKV-8000シリーズの対応状況は以下のとおりです。

分類	機種	対応状況 ○：使用可能 －：使用不可
電源ユニット	KV-U7	－ （KV-EB1への供給用としては使用可能）
入力ユニット	KV-B16XA/KV-B16XC/ KV-C32XA/KV-C32XC/ KV-C64XA/KV-C64XB/ KV-C64XC	○
出力ユニット	KV-B8RC/KV-B16RA/KV-B16RC/ KV-B16TA/KV-B16TC/KV-B16TD/ KV-B16TAP/KV-B16TCP/ KV-C32TA/KV-C32TC/KV-C32TD/ KV-C32TAP/KV-C32TCP/ KV-C64TA/KV-C64TC/KV-C64TD/ KV-C64TAP/KV-C64TCP	○
入出力ユニット	KV-B8XTD/KV-C16XTD/KV-C32XTD	○
アナログユニット	KV-AD40/KV-AD40V/KV-AD40G/ KV-DA40/KV-DA40/ KV-AM40V	○
温度ユニット	KV-TF40/KV-TP40	○
位置決めユニット	KV-H20S/KV-H40S/KV-H20G	○
位置決め· モーションユニット	KV-ML16V/KV-MC20V/KV-MC40V/KV-MX1	○
通信型位置決めユニット	KV-LH20V	○
高速カウンタユニット	KV-SC20/KV-SC20V	○
コミュニケーション ユニット	KV-XLE02	○
	KV-XCM02	○
	KV-XL202/XL402	○
	KV-L20/KV-L20R	－
	KV-L20V/KV-L21V	○
	KV-EP21V	○
	KV-LE20/KV-LE20A	－
	KV-LE20V/KV-LE21V	○
	KV-FL20	－
	KV-FL20V	○
	KV-LM20	－
	KV-LM20V/KV-LM21V	○
	KL-N20V/MV-L40/KV-CL20/KV-DN20	○
システム分配 配置ユニット	KV-EB1	○
エラー出力ユニット	KV-DR1	－
Bluetoothユニット	KV-BT1	－

使用時の制限事項

KV-8000シリーズで既存のKVシリーズの拡張ユニットを使用する場合、以下の制限事項が発生します。

■ KV-XLE02

- 機能バージョン1.2以降のユニットのみ接続可能です。
- KV-8000Aの場合メールの送受信機能が使用できません。

■ KV-XCM02

- KV-8000Aの場合メールの送受信機能が使用できません。

■ KV-XL202/XL402

- 機能バージョン1.1以降のユニットのみ接続可能です。

■ KV-XH16ML/XH04ML/XH64EC/XH32EC/XH16EC

- KV-8000Aの場合、同期接点出力が使用できません。

■ KV-L21V/KV-L20V

- 上位リンクモードを使用した場合、KV-L21V/KV-L20V経由で読み書きできるRの範囲はR00000～R99915まで、MRの範囲はMR00000～MR99915までです。（XYMモード使用時はM00000～M64000まで使用できます）
- 上位リンクモードを使用した場合、KV-L21V/KV-L20V経由で読み書きできる拡張ユニットバッファメモリは0～32767です。
- プロトコルモードを使用した場合、KV-L21V/KV-L20V経由で読み書きできる拡張ユニットバッファメモリは0～32767です。
- リンクモードを使用した場合、KV-L21V/KV-L20V経由で読み書きできるRの範囲はR00000～R99915まで、MRの範囲はMR00000～MR99915までです。

■ KV-LE21V/KV-LE20V

- KV-LE21V/KV-LE20Vを経由した場合、VM65536以降にはアクセスできません。
- 上位リンクモードを使用した場合、KV-LE21V/KV-LE20V経由で読み書きできるRの範囲はR00000～R99915まで、MRの範囲はMR00000～MR99915まで（XYMモード使用時はM00000～M64000まで使用できます）、VMの範囲はVM00000～VM63999までです。
- MCプロトコルモードを使用した場合、KV-LE21V/KV-LE20V経由で読み書きできる拡張ユニットバッファメモリは0～32767です。
- FTPサーバ機能を使用した場合、R100000以上のデバイスファイルは表示されません。
- FTPサーバ機能を使用した場合、6ケタ以上のMR（たとえば、MR100000以上のデバイス）の末尾一桁は表示できません。
- FTPサーバ機能を使用した場合、CPUメモリへの読み書きはできません。
- FTPクライアント機能を使用した場合、CPUメモリへの読み書きはできません。
- KV-8000Aの場合メールの送受信機能が使用できません。

■ KV-EP21V

- KV-EP21Vを経由した場合、VM65536以降にはアクセスできません。
- 上位リンクモードを使用した場合、KV-LE21V/KV-LE20V経由で読み書きできるRの範囲はR00000～R99915まで、MRの範囲はMR00000～MR99915まで（XYMモード使用時はM00000～M64000まで使用できます）、VMの範囲はVM00000～VM63999までです。
- MCプロトコルモードを使用した場合、KV-LE21V/KV-LE20V経由で読み書きできる拡張ユニットバッファメモリは0～32767です。
- FTPサーバ機能を使用した場合、R100000以上のデバイスファイルは表示されません。
- FTPサーバ機能を使用した場合、6ケタ以上のMR（たとえば、MR100000以上のデバイス）の末尾一桁は表示できません。
- FTPサーバ機能を使用した場合、CPUメモリへの読み書きはできません。
- FTPクライアント機能を使用した場合、CPUメモリへの読み書きはできません。
- メッセージ通信機能を使用した場合、KV-EP21V経由で読み書きできる拡張ユニットバッファメモリは0～32767です。
- KV-8000Aの場合メールの送受信機能が使用できません。

■ KV-LM21V/KV-LM20V

- メガリンクモードを使用した場合、KV-LM21V/KV-LM20V経由で読み書きできる拡張ユニットバッファメモリは0～32767、VMは0～63999です。
- シリアルPLCリンクモードを使用した場合、KV-LM21V/KV-LM20V経由で読み書きできるリンクビットデバイスはB0000～B1FFF、リンクレジスタはW0000～W3FFFです。

■ KV-FL20V

- KV-FL20Vで読み書きできるリンクビットデバイスはB0000～B3FFF、リンクレジスタはW0000～W3FFF、R1000～R99915です。

■ KV-CL20

- トランジェント伝送機能を使用する場合、T/C/CRに対する書き込みはできません。

■ KL-N20V

- 通信状態確認モードで起動する場合、ダイレクトアクセススイッチが緑点灯するまでダイレクトアクセススイッチを押し続けてください。ダイレクトアクセススイッチを離すのが早いと、正常に起動できずにCPUエラーE63[電源再投入が必要]が発生する場合があります。

11 KV-8000シリーズへの機種変更時の注意事項

既存のKVシリーズからKV-8000シリーズへの機種変更対応状況は以下のとおりです。

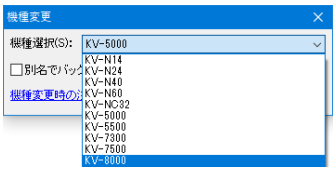
元機種	KV-8000シリーズへの機種変更
KV-7500/7300	○
KV-5500/5000/3000	○
KV-1000	○
KV-700	○
KV Nano	不可（一度KV-3000に機種変更してから実行してください。）
KV-10/16/24/40	○

KV-7000シリーズおよび、KV-5000/3000シリーズからKV-8000シリーズに機種変更する場合に注意する内容は以下のとおりです。

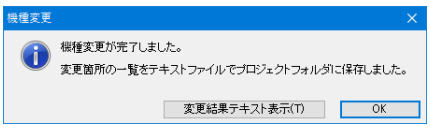
項目	内容
ハードウェアの機能	置き換えできません。 各機器で使用できない機能については、変更時に削除または初期化されます。 変更後に仕様を確認し、仕様範囲内で使用してください。 「CPUのハードウェア特殊機能などに関する差分」（付-47ページ）
デバイス	コントロールリレー/コントロールメモリ（CR/CM）以外のデバイスは自動置換できます。コントロールリレー/コントロールメモリは置換可能なデバイスのみ自動で置換できます。 「コントロールリレー、コントロールメモリの差分」（付-51ページ）
拡張ユニット	KV-7000シリーズ拡張ユニット そのまま流用できます。 KV-5000/3000シリーズ拡張ユニット 流用できますが、注意事項があります。注意事項については以下を確認してください。 「既存のKVシリーズ拡張ユニット使用時の注意事項について」（付-43ページ）
命令語	命令語により以下の挙動になります。 • 未登録変数に変換 • 自動で適切な命令語/オペランドに変更 「差分がある命令語」（付-51ページ）

機種置き換え時の手順

1 《KV STUDIO》のメニューから【ファイル（F）】→【機種変更（L）】を選択して、機種変更を実行します。



参考　機種変更の結果は、「変換結果テキスト」に保存されます。「変換結果テキスト」を表示して、未登録変数に変換されていないかなどを個別に確認してください。



CPUのハードウェア特殊機能などに関する差分

項目				KV-8000シリーズ	KV-7000シリーズ		KV-5000/3000シリーズ	
				KV-8000(A)	KV-7500	KV-7300	KV-5500 KV-5000	KV-3000
基本仕様	電源	DC		DC 24V (-15%, +20%)※5	DC 24V (-15%, +20%)※5		DC 24V(±10%)	
		AC(電源ユニット)		AC 100～240V (+10%,-15%)(50/60Hz)	AC 100～240V (+10%,-15%)(50/60Hz)		AC 100～240V±10% (50/60Hz)	
	バッテリ			別売(KV-B1)	別売(KV-B1)		同梱	
	通信ポート	USB	1.1(12Mbps)	○	○		○	
			2.0(480Mbps)	○	○		–	
		RS-232C		–	–	○	–	○
		イーサネット※6	Ethernet	○※7	○	–	○	–
			FL-net	–	–	–	○(KV-5000)	–
			EtherNet/IP	○	○	–	○(KV-5500)	–
			ソケット通信	○ (最大16本)	○ (Ver.1.1以降) (最大16本)	–	○(KV-5000) (最大8本)	–
	メモリ	CPUメモリ容量※1		合計64MB	合計64MB	合計21MB	–	
		内訳	プログラム容量※1	1500kstep	1500kstep	160kstep	260kstep	160kstep
			デバイスコメント容量※3	224,000個	224,000個	102,000個	96,000個	30,000個
			ユーザメモリ容量※1	40MB	40MB	10MB	–	
		デバイスコメント	コメント本数	8本	8本		1本※2	
			ユニバーサルコメント	○	○		–	
		メモリカードレス	ロギング	○	○		–	
			ファイルレジスタ	○	○		–	
			ロード/セーブ	○	○		–	
	ローカルワーク			589824個	51000個		51000個	
	最大I/O点数			3072点	3072点		3096点	
	接続可能ユニット台数	KV-8000シリーズ拡張ユニット		最大16台 (KV-7000シリーズ拡張ユニットとの合計)	–		–	
		KV-7000シリーズ拡張ユニット(S-Unit, X-Unit)		最大16台 (KV-8000シリーズ拡張ユニットとの合計)	最大16台		–	
		KV-5000/3000シリーズ拡張ユニット		最大48台 (KV-8000/7000シリーズ拡張ユニットとの合計)	最大48台 (KV-7000シリーズ拡張ユニットとの合計)		最大48台	
	命令語実行速度	基本命令実行速度		0.96ns	0.96ns		10.0ns	
	停電保持	デバイス		不揮発RAMでバックアップ	不揮発RAMでバックアップ		バッテリでSRAMバックアップ	
		変数/ラベル		○	–		–	
		カレンダタイマ(RTC)		バックアップ用コンデンサでバックアップ (約15日) バッテリ使用時 (約5年(@25℃))	バックアップ用コンデンサでバックアップ (約15日) バッテリ使用時 (約5年(@25℃))		バッテリバックアップ	
		プログラム		フラッシュROM (書き込み回数1万回)	フラッシュROM (書き込み回数1万回)		フラッシュROM (書き込み回数10万回)	

項目			KV-8000シリーズ	KV-7000シリーズ		KV-5000/3000シリーズ	
			KV-8000(A)	KV-7500	KV-7300	KV-5500 KV-5000	KV-3000
入出力	入力		−	−		16点 （ラインドライバ対応）	
	出力		−	−		8点	
	コネクタ		−	−		MILコネクタ	
	サービス電源出力		−	−		+5V	
アクセスウィンドウ	アクセスウィンドウ	全角･半角	全角･半角	全角･半角		半角	
		サイズ	縦 : 4文字 横 : 半角16文字	縦 : 4文字 横 : 半角16文字		縦 : 4文字 横 : 半角12文字	
		言語	日, 英, 中	日, 英, 中		カタカナ or 英字	
		ユーザメッセージ	○	○		−	
メモリカード	スロット		○（SDHC対応）	○（SDHC対応）		○（SDHC対応）	
	ラダープログラムからのアクセス		可（ストレージ命令）	可（ストレージ命令）		可（ストレージ命令）	
	ロギング/トレース機能		○	○		○	
	FTPサーバ機能使用時のフォルダ名称		0_CARD	0_CARD		MMC	
CPUメモリ	ラダープログラムからのアクセス		○	○		−	
	ロギング/トレース機能		○	○		−	
	ユーザードキュメント		○	○		−	
内蔵機能	高速カウンタ		−	−		○	
	内蔵位置決め		−	−		○	
	運転記録機能		○	−		−	
	ユーザ認証機能		○	−		−	
	周波数カウンタ		−	−		○	
	指定周波数パルス出力		−	−		○	
	カムスイッチ機能		−	−		○	
	ロギング/トレース機能	バッファ容量	10240KB	2048KB		1024KB	
		上限デバイス数	512個	128個		128個	
	シリアル通信	動作モード	−	KV（上位リンク）モード KV STUDIOモード		KV（上位リンク）モード KV（テキスト送受信）モード KV STUDIOモード	
	割り込み	割り込み要因 （入力リレー）	−	−		10点（R000～R009）	
		割り込み要因 （高速カウンタコンパレータ）	−	−		4点（CTC0～CTC3）	
		割り込み要因 （拡張ユニット割り込み）	○	○		−	
		割り込み要因最大数	64本	64本		14本	
		ダイレクト入力	−	−		16点（R000～R015）	
		ダイレクト出力	−	−		8点（R500～R507）	
レシピロード･セーブ			メモリカード CPUメモリ	−		−	
プロジェクトデータの保存･読み出し			メモリカード CPUメモリ	メモリカード CPUメモリ		メモリカード	
運転記録の保存･読み出し			メモリカード CPUメモリ	−		−	
プログラム	対応言語		ラダー/KVスクリプト/ニモニック	ラダー/KVスクリプト/ニモニック		ラダー/KVスクリプト/ニモニック	
	定周期実行	周期	50μs	50μs		200μs	
		本数	4本	4本		1本	
	パスワード	モジュールパスワード	最大32文字	最大32文字		−	
		ファンクションブロックパスワード	最大32文字	最大32文字		−	
		マクロパスワード	最大32文字	最大32文字		最大6文字	
		プロジェクトパスワード	最大32文字	最大32文字		最大6文字	
	プログラム数		4000個	4000個		950個	

項目			KV-8000シリーズ	KV-7000シリーズ		KV-5000/3000シリーズ	
			KV-8000(A)	KV-7500	KV-7300	KV-5500 KV-5000	KV-3000
その他の機能	ブレイク		×	×		モニタ中、 シミュレータ中	
	内蔵フリーランカウンタ		32bit　0.1μs 32bit　1μs 32bit　10μs 32bit　1ms	32bit　0.1μs 32bit　1μs 32bit　10μs 32bit　1ms		32bit　1μs	
	入力リフレッシュ禁止		−	−		○	
	バス通信	V-BUS RT	○	○		○	
		CPUインナーバス	○	○		−	
	システム異常検出		○（KV-PU1使用時）	○（KV-PU1使用時）		○（KV-DR1使用時）	
	初期化		○	−		−	
	プロジェクトロード禁止		○	−		−	

※1　プロジェクト領域（プログラム、デバイスコメントなど）、CPUメモリ領域の各容量をユーザが設定可能。内訳はデフォルト設定の場合。
※2　KV STUDIO上で設定可能な本数は2本だが、PLCで保存可能な本数は1本。
※3　プロジェクト領域を全てデバイスコメントで使った場合の目安。
※4　KV-7000シリーズはユニット割り込み、KV-5000/3000シリーズは内蔵入力割り込み。
※5　KV-5000/3000シリーズの拡張ユニットを使用する場合はDC24V±10%です。
※6　詳細は『EtherNet/IP機能ユーザーズマニュアル』を参照してください。
※7　KV-8000Aではメール通信機能は使用できません。

デバイス範囲の比較表

<table>
<tr><th colspan="2" rowspan="2">項目</th><th>KV-8000(A)</th><th colspan="2">KV-7500/7300</th><th>KV-5500/5000/3000</th></tr>
<tr><th></th><th>（Ver2.3以降）</th><th>（Ver2.2以前）</th><th></th></tr>
<tr><td rowspan="19">CPUユニットデバイス</td><td>CPU入力リレー</td><td colspan="3">－</td><td>R000～R015</td></tr>
<tr><td>CPU出力リレー</td><td colspan="3">－</td><td>R500～R507</td></tr>
<tr><td rowspan="2">内部補助リレー</td><td>R00000～R199915</td><td>R00000～R199915</td><td>R00000～R99915</td><td>R01000～R99915</td></tr>
<tr><td colspan="3">MR000000～MR399915</td><td>MR00000～MR99915</td></tr>
<tr><td>ラッチリレー</td><td colspan="4">LR00000～LR99915</td></tr>
<tr><td>コントロールリレー</td><td colspan="3">CR0000～CR7915</td><td>CR0000～CR3915</td></tr>
<tr><td>タイマ</td><td colspan="4">T0000～T3999</td></tr>
<tr><td>カウンタ</td><td colspan="4">C0000～C3999</td></tr>
<tr><td>データメモリ</td><td colspan="4">DM00000～DM65534</td></tr>
<tr><td>拡張データメモリ</td><td colspan="4">EM00000～EM65534</td></tr>
<tr><td>ファイルレジスタ</td><td colspan="3">FM00000～FM32767（×16）
（ZF00000～ ZF524287）</td><td>FM00000～FM32767（×4）
（ZF00000～ZF131171）</td></tr>
<tr><td>テンポラリデータメモリ</td><td colspan="4">TM000～TM511</td></tr>
<tr><td>コントロールメモリ</td><td>CM0000～CM7599</td><td colspan="3">CM0000～CM5999</td></tr>
<tr><td>高速カウンタ</td><td colspan="3">－</td><td>CTH0～CTH1</td></tr>
<tr><td>高速カウンタコンパレータ</td><td colspan="3">－</td><td>CTC0～CTC3</td></tr>
<tr><td>リンクリレー</td><td colspan="3">B0000～B7FFF</td><td>B0000～B3FFF</td></tr>
<tr><td>リンクデータメモリ</td><td colspan="3">W0000～W7FFF</td><td>W0000～W3FFF</td></tr>
<tr><td>インデックスレジスタ</td><td colspan="4">Z01～Z12</td></tr>
<tr><td>デジタルトリマ</td><td colspan="4">TRM0～TRM7</td></tr>
</table>

差分がある命令語

KV-5000/3000シリーズからKV-8000シリーズへ機種変更するときに差分がある命令語は以下のとおりです。

※ KV-7000シリーズからKV-8000シリーズへ機種変更する場合、命令語に差分はありません。

命令語		機種変更時	KV STUDIO変換時	実行時
ストレージ命令	MFREE	MFREE #0 D1 D2 に置換	–	正常動作
	MFREEK	MFREEK #0 D1 D2 に置換	–	正常動作
アクセスウィンドウ命令	AWNUM	未登録変数に変更	エラー発生	–
	AWMSG	未登録変数に変更	エラー発生	–
位置決め命令	PLSX	未登録変数に変更	エラー発生	–
	PLSY	未登録変数に変更	エラー発生	–
	JOGX	未登録変数に変更	エラー発生	–
	JOGY	未登録変数に変更	エラー発生	–
	ORGX	未登録変数に変更	エラー発生	–
	ORGY	未登録変数に変更	エラー発生	–
	HOMEX	未登録変数に変更	エラー発生	–
	HOMEY	未登録変数に変更	エラー発生	–
	CHGSPX	未登録変数に変更	エラー発生	–
	CHGSPY	未登録変数に変更	エラー発生	–
	RFSPSX	未登録変数に変更	エラー発生	–
	RFSPSY	未登録変数に変更	エラー発生	–
高速処理命令	HSP	未登録変数に変更	エラー発生	–
	CTH	未登録変数に変更	エラー発生	–
	CTC	未登録変数に変更	エラー発生	–
	RFSCTH	未登録変数に変更	エラー発生	–
割り込み命令	INT	未登録変数に変更	エラー発生	–
ファイルレジスタ命令	FRSTM	FRSTM #0 n S D に置換	–	正常動作
	FRLDM	FRLDM #0 n S D に置換	–	正常動作
カムスイッチ命令	INCENC	未登録変数に変更	エラー発生	–
周波数カウンタ命令	FCNT	未登録変数に変更	エラー発生	–
	RCNT	未登録変数に変更	エラー発生	–
	PLSOUT	未登録変数に変更	エラー発生	–

※ KV-8000Aに変換する場合、メール送受信機能に関連するユニット専用命令は未登録変数に変更されます。

コントロールリレー、コントロールメモリの差分

CR/CM（コントロールリレー/コントロールメモリ）の差分については、《KV STUDIO》のメニューから【ヘルプ(H)】→【お役立ちナビ（参考プログラム・技術資料など）】の中の「置き換え資料」-「CR/CM変換差分表（Excelファイル）」を参照してください。

変数とラベルの差分

項目		変数	ラベル
対応CPUユニット		• KV-8000(A)	• KV-7500/7300 • KV-5500/5000/3000
仕様	最大定義数※1	100000※2	約285000
	名前の最大長	128文字(全角/半角問わず)	32文字
	ワーク領域への自動割付	グローバル変数、ローカル変数共に対応	ローカルラベルのみ対応
	初期値	○	―
	定数	○	○
	保持	○(最大30000定義)※3	―
データ型	BOOL(ビット)	○	○
	INT(1ワード符号あり整数)	○	○
	UINT(1ワード符号なし整数)	○	○
	DINT(2ワード符号あり整数)	○	○
	UDINT(2ワード符号なし整数)	○	○
	REAL(単精度浮動小数点)	○	○
	LREAL(倍精度浮動小数点)	○	○
	STRING(文字列)	○	○
	TIMER(タイマ)	○	○
	COUNTER(カウンタ)	○	○
	配列	○	○
	構造体	○	―
	ファンクションブロック	○	―
その他	コメント	8種類	1種類
	VT STUDIOとの連動※4	○	―

※1 定義数とは変数、ラベルとして登録した名前の個数です。例えば10000個の要素を持つ配列型の変数、ラベルを登録した場合、定義数は1としてカウントされます。
※2 変数名の文字数によっては、100000個設定できない場合があります。
※3 変数名の文字数によっては、30000個設定できない場合があります。
※4 VT STUDIOで変数を使用する場合、VT STUDIOを8.0以降にバージョンアップし、VT機種をVT5シリーズ、もしくはSoft-VTに設定してください。

MEMO

12 索　引

本書で使用している用語の索引です。アルファベット、五十音順に並んでいます。

数字

100ms 単位オンディレイ 4-167
10ms 単位オンディレイ 4-169
10us 単位オンディレイ 4-173
10 進数 4-76
10 進定数 4-55, 4-73
16 進数 4-76
16 進定数 4-73
16 ビットバイナリデータ 4-75
1ms 単位オンディレイ 4-171
2 進化 10 進数 4-77
2 進数 4-76
32 ビットバイナリデータ 4-75
初期化 5-158

A

ABSENC 命令 5-187
ACNT 命令 4-67
ASCII コード 4-81

B

B 4-29
BCD 4-77
BIN 4-76
BSIZE 命令 4-71

C

C 4-35, 4-42
CM 4-43
CPU で使用するデバイス 5-32
CPU メモリ(ユーザメモリ)でできること 5-27
CPU メモリについて 5-29
CPU モニタ 5-136
CPU ユニットの動作 4-82
CR 4-36
CSV ファイルについて 5-71
CTD 4-163
CTU 4-161
CTUD 4-165

D

DEC (Decimal) 4-76
DI 命令 4-218
DIC 命令 4-220
DIN レールへの取り付け 3-5
DOUBLE-FLOAT 4-78

E

EI 命令 4-218
EN 4-122
ENO 4-122
EtherNet/IP 通信 5-112

F

FB 命令 4-148
FBCALL 命令 4-151
FBSTRT 命令 4-151
FEND 命令 4-148
FLOAT 4-77
FM 4-39
FRLDM 命令 4-234
FRSET 命令 4-230
FRSTM 命令 4-232
F_TRIG 4-160
FUN 命令 4-148

H

HEX 4-76

I

IEDGE 命令 4-224
INT 命令 4-222

K

KV-SIR32XT 用ユニット専用命令一覧 6-23
KV-SIR32XT 用ユニット専用関数一覧 6-30
KV スクリプト 4-7

L

LOGD 命令 5-74
LOGE 命令 5-74
LR 4-31

M

MC ～ MCR 命令間で呼び出す場合の動作 4-198
MCALL 命令 4-193
MCMP 命令 5-183
MDSTOP 命令 4-115
MDSTRT 命令 4-115
MEND 命令 4-193
MR 4-31
MSTRT 命令 4-193

O

OPC UA サーバ機能でできること 5-112

R

R 4-27, 4-30
RAM クリア 5-156
RAM モード 5-47
RETI 命令 4-222
RS 4-157
R_TRIG 4-159
RUN-PRG 切換スイッチ 2-8

S

SR 4-155
ST 言語 4-6

T

T 4-32, 4-42
TM 4-42
TON_100MS 4-167
TON_10MS 4-169
TON_10US 4-173
TON_1MS 4-171
TRGD 命令 5-76
TRGR 命令 5-106

U

U_DISOCP 命令 6-28
U_DISOVP 命令 6-26
U_RDBF 命令 6-24
USB 通信 5-111
USB ドライバのインストール 5-111

W

W 4-41
WSIZE 命令 4-69

Z

Z 4-43
ZF 4-39

あ

アクセスウィンドウ 5-113, 6-37
　アクセスウィンドウでできること 5-113
　アクセスウィンドウの各部の名称 5-116
　機能一覧 5-114
アクセスウィンドウ初期画面表示機能 5-161
アクセスウィンドウ設定 5-154
アスキーコード 4-81
アップカウンタ 4-161
アップダウンカウンタ 4-165

い

一括モニタ 4-274
一般仕様 2-2
インスタンス 4-120, 4-121
インデックス修飾 4-44, 4-265
インデックス修飾との併用例 4-48
インデックス定数 4-45
インデックスレジスタ 4-43, 4-44

え

エクステンションユニット 2-42
エディタモードで
　有効なショートカットキー .....付 -27
　スクリプト .....付 -28
　ツール .....付 -29
　表示 .....付 -28
　ファイル .....付 -27
　ヘルプ .....付 -29
　変換 .....付 -28
　編集 .....付 -27
　モニタ / シミュレータ .....付 -29
エラークリア 5-129
エラーメッセージ一覧 .....付 -2

お

オートローディング機能 5-45
オールクリア 5-157
オプション 5-67
オンラインエディット 4-269
オンラインエディット中に
　有効なショートカットキー .....付 -30
　スクリプト .....付 -31
　表示 .....付 -31
　ヘルプ .....付 -32
　編集 .....付 -30
　モニタ / シミュレータ .....付 -32

か

外形寸法図 2-10
　AC 電源ユニット 2-13
　CPU ユニット 2-10
　エクステンションユニット 2-45
　エンドユニット 2-10
　拡張出力ユニット 2-33
　拡張入出力ユニット 2-39
　拡張入出力ユニット(KV-SIR32XT) 2-17
　拡張入力ユニット 2-24
　バス接続ユニット 2-10, 2-41
カウンタ 4-35, 4-42
カウンタの最大計数速度 4-36
カウンタの設定値･現在値を扱う方法 4-36
カウンタを動作させる時の記述方法 4-35
拡張出力ユニット 2-25
拡張入力ユニット 2-18
各部の名称と機能 2-6
　AC 電源ユニット 2-11
　エクステンションユニット 2-42

拡張出力ユニット 2-25
拡張入出力ユニット（KV-SIR32XT） 2-14
拡張入出力ユニット 2-34
拡張入力ユニット 2-18
バス接続ユニット 2-40
各ユニットへの配線時の注意 3-8
接地時の注意 3-8
接点保護について 3-9
入出力機器への配線時の注意 3-8
カムスイッチ機能の概要 5-182
カムスイッチ命令 5-182
簡易間接指定 4-49
間接指定 4-46, 4-265

き

キーロック 5-167
行コメント 4-254
行コメントの検索方法 4-255

く

グローバルデバイス 4-200
グローバルデバイス表記 4-200

け

結線･内部回路図 2-8
Ethernet 通信仕様 2-9
拡張出力ユニット 2-28
拡張入出力ユニット（KV-SIR32XT） 2-16
拡張入力ユニット 2-20
結線時の注意
(OP-22184/OP-51404/OP-23139/
OP-42224) 3-15

こ

構造体 4-59
構造体の定義 4-66
工程別 4-111
固定文字列 4-74
コネクタの組み立てと結線 3-15
コントロールメモリ 4-43
コントロールリレー 4-36

さ

最大計数速度 4-36
最大値･最小値を超える時の数値の変化 4-80
変数の種類 4-55
サフィックス 4-9
サブルーチン型マクロ 4-181
サブルーチンプログラム .....4-94, 4-196, 付 -22

し

シーケンス制御の種類 4-4
時限制御 4-4
順序制御 4-4
条件制御 4-4
識別番号 4-189
自己保持型マクロ 4-183
システム構成の概要 1-2
①電源について 1-3
② CPU ユニットについて 1-3
③ KV-7000 シリーズ用
拡張ユニットについて 1-3
④バス接続ユニットについて 1-4
⑤ KV-5000/3000 シリーズ用
拡張ユニットについて 1-4
⑥エンドユニットについて 1-4
実行条件 OFF 時の動作 4-198
実行中 ON ビット 4-113, 4-189
実体の作成 4-196
シミュレータモードで
有効なショートカットキー .....付 -30
デバッグ .....付 -30
表示 .....付 -30
ヘルプ .....付 -30
編集 .....付 -30
モニタ / シミュレータ .....付 -30
出力リレー 4-27
ショートカットキー一覧 .....付 -25
/ キーの使い方 .....付 -37
CX 形式のショートカット一覧 .....付 -34
GPPQ/GPPA 形式の
ショートカット一覧 .....付 -33
KV STUDIO で
有効なショートカットキー .....付 -27
Logix 形式のショートカット一覧 .....付 -35
ユニットエディタで
有効なショートカットキー .....付 -25
仕様
AC 電源ユニット 2-12
エクステンションユニット 2-43
拡張出力ユニット 2-26
拡張入出力ユニット（KV-SIR32XT） 2-15
拡張入力ユニット 2-19
使用する場面 4-199
使用できない命令語一覧 .....付 -21
使用できない文字一覧 .....付 -40
構造体のメンバ名に使用できない文字 .....付 -42
デバイスコメントに使用できない文字 .....付 -42
ファンクション / ファンクションブロック /
構造体名に使用できない文字 .....付 -40
プログラム名に使用できない文字 .....付 -40
プロジェクト名に使用できない文字 .....付 -40
変数に使用できない文字 .....付 -40
使用できるメモリカード 5-6
初期画面で有効なショートカットキー .....付 -27
表示 .....付 -27
ファイル .....付 -27
モニタ / シミュレータ .....付 -27
初期化モジュール .....4-100, 付 -21
シリアル通信 5-86
シリアル通信でできること 5-87

す

数値の表記 4-76
数値表記の相関 4-78
スキャンタイム 4-90
ストレージ命令 5-26, 5-49

せ

性能仕様 2-3
接続線編集 4-242
設置環境 3-6

そ

双安定ファンクションブロック
（セット優先）....4-155
双安定ファンクションブロック
（リセット優先）....4-157
操作禁止設定 5-166
総ステップ数 4-15

た

待機モジュール 4-101
退避 4-96
タイマ 4-32, 4-42
タイマの誤差 4-33
タイマの設定値･現在値を扱う方法 4-34
タイマ命令使用時の注意 4-34
タイマを動作させる時の記述方法 4-33
ダウンカウンタ 4-163
立ち上がりエッジ検出器 4-159
立ち下がりエッジ検出器 4-160
端子台ユニットと接続ケーブル 3-11
短縮入力 ....付 -23
単精度浮動小数点型実数 4-77
単精度浮動小数点型定数 4-73

つ

通信機器との接続方法 1-5
　タッチパネルと通信する場合 1-5
　パソコンと通信する場合 1-5

て

ディレーティング
　拡張出力ユニット 2-26
　拡張入力ユニット 2-19
デジタルトリマモード 5-130
　設定操作キーの名称と機能 5-130
　デジタルトリマの変更操作 5-130
手順
　ファンクションブロックを
　　使用する手順 4-119
　マクロを使用する手順 4-180
デバイス番号一覧 ....付 -10
デバイス一覧 4-20
　定数 4-23
　内部レジスタ 4-23
　ビットデバイス 4-20
　ファンクションブロック引数 4-22
　マクロ引数デバイス 4-22
　ワードデバイス 4-21
デバイスコメント 4-252
デバイス値変更ウィンドウ 4-273
デバイスの現在値や状態を変更する 4-273
デバイスの立ち上がり立下りを
　モニタする 4-276
デバイスの表記方法 4-24
デバイスモード 5-120
　設定操作キーの名称と機能 5-120
　デバイス値の表示形式の変更 5-122
　デバイス値の変更
　　（表示形式が ON/OFF 以外の場合
　　（T、C、TRM））....5-127
　デバイス値の変更
　　（表示形式が ON/OFF 以外の場合
　　（T、C、TRM を除く））....5-126
　デバイス値の変更表示形式が
　　ON/OFF の場合 5-125
　デバイス名 / コメント表示の切り換え 5-124
　デバイスモードでの操作方法 5-121
　デバイスモードへの移行 5-121
　表示･設定変更できるデバイス 5-120
　モニタするデバイス種別の変更 5-121
電源断時の動作 2-4
点検とお手入れについて 3-20
テンポラリデータメモリ 4-42

と

登録モニタ 4-274
時計 5-134
取り付け位置 3-6

な

内蔵機能でできること 5-2
内部補助リレー 4-30
内部レジスタ 4-53
内部ワーク使用量 4-15

に

ニモニック 4-8
ニモニック短縮入力一覧 ....付 -23
入力アシスト 4-239
入力リレー 4-27

ね

ネスティング 4-95

は

倍精度浮動小数点型実数 4-78
倍精度浮動小数点型定数 4-74
配線方法
　AC 電源ユニット 2-12
バイトサイズ 4-14
バイナリデータ 4-75
配列 4-57
バス接続ユニット 2-40, 2-42
パスワード 4-18
パソコンとの接続方法 5-88, 5-111
パッケージ内容 2-5
バンク切換 4-226
汎用入出力 6-2

ひ

引数 4-131, 4-186
引数設定 4-131
引数デバイス 4-186
ビットデバイスの ON/OFF 4-273
ビットデバイスのワード単位扱い 4-50
微分実行型命令の初期状態 4-198
表示モードの設定 4-275

ふ

ファイルレジスタ 4-39
ファンクション ....4-128, 付 -21
ファンクションブロック ....4-120, 付 -21
ファンクションブロックとマクロの違い 4-175
ファンクションブロックの管理 4-117
ファンクションブロックの
　システムデバイス 4-143
　ファンクションブロック実行開始時
　　1 スキャンのみ OFF ビット 4-143
　ファンクションブロック実行開始時
　　1 スキャンのみ ON ビット 4-143
ファンクションブロックの種類 4-120
ファンクションブロックの
　プログラムプロパティ 4-144
　コメント 4-145
　実行停止時に OFF 処理する 4-144
　プログラムの無効化 4-144
　読み取り専用 4-144
符号 4-76
復帰 4-96
復旧方法について 3-22
浮動小数点型定数 4-73
プログラムサイズ 4-14
プログラム単位、割り込みプログラム単位の
　実行時間をモニタする
　（パフォーマンスモニタ）....4-277
プログラムの構成 4-93
プログラム容量 4-14
プロジェクトデータの保存･読み出し 5-35
プロジェクトの構成 4-11
プロジェクトのプロテクト設定 4-16
プロジェクトプロパティ 4-18
分割 4-251

へ

変換結果 4-14
変数 4-55
変数とデータ型一覧 4-56
変数の設定 4-63

ほ

保守について 3-20

ま

毎スキャンモジュール 4-100
マクロ ....付 -22
マクロシステムデバイス 4-188
マクロ実行開始時 1 スキャンのみ
　OFF ビット 4-188
マクロ実行開始時 1 スキャンのみ
　ON ビット 4-188
マクロ実行中 ON ビット 4-188, 4-189
マクロとサブルーチンプログラムの違い 4-196
マクロの管理 4-178
マクロの種類 4-181

む

無限大 4-80

め

命令語 ....付 -23
命令語候補 4-239
命令語の一覧表示 4-239
命令語のオペランド簡易説明表示 4-239
命令語をモニタする 4-276
メインルーチンプログラム 4-94
メモリカードでできること 5-3
メモリカードについて 5-6
メモリカードの使用上の注意事項 5-9
メモリカードを装着する 5-8
メモリカードを取り出す 5-8

も

文字コード 4-81
モジュールシステムデバイス 4-113
モジュール実行開始時 1 スキャンのみ
　OFF ビット 4-113
モジュール実行開始時 1 スキャンのみ
　ON ビット 4-113
モジュール実行中 ON ビット 4-113
モジュールの構成例 4-110
モジュールの実行順序 4-109
モジュールの種類 4-100
モジュールを意識せずにプログラムする 4-112

モニタモードで
有効なショートカットキー ..... 付 -29
デバッグ ..... 付 -29
表示 ..... 付 -29
ヘルプ ..... 付 -29
編集 ..... 付 -29
モニタ / シミュレータ ..... 付 -29

ゆ

ユーザメッセージ 5-165
表示方法 5-165
表示例 5-165
ユーザメモリクリア 5-159
ユニット間同期機能 5-169
CPU デバイス書き込み／読み出し 5-174
CPU デバイス書き込み／読み出しの
設定方法 5-174
ユニット間同期とは 5-169
ユニット間同期トレース 5-180
ユニット間同期の周期 5-175
ユニット間同期の性能仕様 5-176
ユニット間同期の設定手順 5-178
ユニット間同期の設定方法 5-177
ユニット間同期のシステム構成 5-175
ユニット間同期モジュール 5-170
ユニット間同期リフレッシュデバイス 5-173
ユニット間同期リフレッシュデバイスの
設定方法 5-173
リフレッシュタイミングの設定
（拡張ユニット用リレー･
DM のみ対象） 5-172
ユニット間同期機能 (S-Unit) 6-6
ユニット間同期機能実行時の動作概要 6-6
ユニット間同期とは 6-6
ユニットエディタで
有効なショートカットキー ..... 付 -25
ウインドウ ..... 付 -25
子機設定 ..... 付 -25
表示 ..... 付 -25
ファイル ..... 付 -25
ヘルプ ..... 付 -25
変換 ..... 付 -25
編集 ..... 付 -25
メッセージエリアでの操作 ..... 付 -25
ユニット設定ウインドウでの操作 ..... 付 -25
ユニット選択ウインドウでの操作 ..... 付 -25
ユニット配置エリアでの操作 ..... 付 -25
ユニット接続時の注意 3-2
ユニットエディタ「子機設定画面」で
有効なショートカットキー ..... 付 -26
ユニットテスト 5-132
ユニットの組み立て 3-3
ユニットのモニタ 5-117
設定操作キーの機能 5-117
操作方法（入出力ユニットの場合） 5-118
ユニットモニタの概要 6-42
ユニットモニタの表示 6-42
ユニット用デバイス表記 4-28
ユニットをモニタする 4-277

よ

呼び出せる箇所 4-197
読み出し禁止 4-18
予約領域 4-202

ら

ラダープログラム 4-5
ラッチリレー 4-31

り

リプレイモードで
有効なショートカットキー .....付 -32
リンクリレー 4-29
リンクレジスタ 4-41

れ

レシピデータ生成･読み出しの手順 5-52
レシピロード完了コード･
レシピセーブ完了コード 5-56
レシピロード機能 / レシピセーブ機能の
概要 5-50
レシピロード･セーブの手順 5-54

ろ

ロギング･トレース機能の注意事項 5-82
ロギング･トレース用デバイス一覧 5-78
ロギング機能 5-57
ロギング命令 5-73

わ

ワードデバイス 4-37
ワードデバイスのビットデバイス扱い 4-52
割り込み 4-206
割り込みプログラムで
使用できない命令語 4-210
割り込みプログラム 4-96

改訂履歴

印刷年月日	版　　数	改　訂　内　容
2019年 04月	初　版	
2019年 09月	2　版	
2020年 04月	改訂1版	
2021年 04月	2訂1版	
2021年 07月	3訂1版	
2021年 09月	4訂1版	
2021年 11月	5訂1版	
2022年 02月	6訂1版	
2022年 05月	7訂1版	
2022年 09月	8訂1版	
2023年 10月	9訂1版	

保証について

1　対象製品

以下に規定する保証は、本マニュアルに記載している製品（以下「対象製品」という）に適用します。
なお、対象製品に内蔵されているリレーや電池などの消耗品は対象外とさせていただきます。

2　保証期間

対象製品の保証期間は、貴社のご指定場所に納入後3年間とします。ただし、次に該当する場合の保証期間は1年間とさせていただきます。

① 対象製品を海外でご購入いただいた場合。
② 海外で修理をご依頼いただいた場合。

3　保証範囲

（1）当社の責任による故障が発生した場合は、代替品との交換または修理が可能です。上記保証期間内の場合は、無償での対応になります。但し、保証期間内であっても、次に該当する故障の場合は保証対象外とさせていただきます。なお、代替品との交換または修理を行なった場合でも保証期間の起算日は対象製品の当初ご納入日とさせていただきます。

① 取扱説明書、ユーザーズマニュアル、別途取り交わした仕様書などに記載された以外の不適当な条件・環境・取り扱い・使用方法に起因した故障。
② お客様の装置または、ソフトウエアの設計内容など、対象製品以外に起因した故障。
③ 当社以外による改造、修理に起因した故障。
④ 取扱説明書、ユーザーズマニュアルなどに記載している消耗部品が正しく保守、交換されていれば、防止できたと確認できる故障。
⑤ 当社出荷時の科学・技術水準では、予見が不可能だった事由による故障。
⑥ その他、火災、地震、水害などの災害及び電圧異常など当社の責任ではない外部要因による故障。

（2）保証範囲は上記（1）を限度とし、対象製品の故障に起因するお客様での二次損害（装置の損傷、機会損失、逸失利益等）及びいかなる損害も保証の対象外とさせていただきます。

4　適用用途

当社製品は、一般工業向けの汎用品として設計・製造されております。
従いまして、下記のような用途での使用は意図しておりませんので適用外とさせていただきます。ただし、事前に当社までご相談いただき、お客様の責任において製品の仕様をご確認のうえ、定格・性能に対してご了承いただき、必要な安全対策を講じていただく場合は適用可能とさせていただきます。なお、この場合においても保証範囲は上記と同様といたします。

① 原子力発電、航空、鉄道、船舶、車両、医療機器等の人命や財産に多大な影響が予想される設備
② 電気、ガス、水道等の公共設備
③ 屋外での使用および、それに準ずる取扱説明書などで規定していない条件・環境での使用
④ 上記①及び②に準じる安全に関して高度な配慮と注意が要求される用途

全商品、送料無料で
当日出荷

必要な時に、必要な量だけ
在庫不要でトータルコストを削減

調べたいことを、いつでも、すぐに調べられる!

▼ 本社技術スタッフ直営 ▼ FAQ(よくあるご質問)

www.keyence.co.jp/seigyofaq ← CLICK!

マニュアルと合わせてご活用ください。

■ 制御システムに関するお問合せ 06-6379-1164

月～金曜(祝日を除く)8:30～20:00

■ ユーザー登録 www.keyence.co.jp/kvuser

ユーザー登録で以下のサービスが受けられます。

KV STANDARD SERVICE

- **無償バージョンアップ**
  専用ホームページから最新版ソフトをダウンロードできます。
- **サイトライセンス**
  ソフトひとつ分の費用で、事務所内のPCなら何台でも使用可能。

■ 輸出書類サービス www.keyence.co.jp/yushutsu

輸出に必要な書類をその場でダウンロードできます。

株式会社 キーエンス
本社・研究所／制御システム事業部
〒533-8555 大阪市東淀川区東中島1-3-14

仕様は改良のため予告なく変更することがあります。
記載されている会社名、製品名等は、それぞれ各社の商標または登録商標です。

制御10-2041

Copyright© 2023 KEYENCE CORPORATION.
All rights reserved.

2103-1 084823