

Sysmac Library

ユーザーズマニュアル 高速アナログ検査ライブラリ編 形 SYSMAC-XR016

おことわり

- (1) 本マニュアルの内容の一部または全部を無断で複写、複製、転載することを禁じます。
- (2) 本マニュアルの内容に関しては、改良のため予告なしに仕様などを変更することがあります。あらかじめご了承ください。
- (3) 本マニュアルの内容に関しては、万全を期しておりますが、万一ご不審な点や誤りなど、お気づきのことがありましたら、お手数ですが、弊社支店または営業所までご連絡ください。その際、巻末記載の Man.No. (マニュアル No.) も併せてお知らせください。

商標

- Sysmac は、オムロン株式会社 FA 機器製品の日本およびその他の国における商標または登録商標です。
- Microsoft、Windows、Windows Vista、Excel、Visual Basic は、米国 Microsoft Corporation の米国およびその他の国における登録商標です。
- EtherCAT[®]は、ドイツ Beckhoff Automation GmbH によりライセンスされた特許取得済み技術であり登録商標です。
- ODVA、CIP、CompoNet、DeviceNet、EtherNet/IP は ODVA の商標です。
- SD、SDHC ロゴは、SD-3C,LLC の商標です。 

その他、本文中に掲載しているシステム名および製品名は、それぞれ各社の商標または、登録商標です。

著作権

- スクリーンショットはマイクロソフトの許可を得て使用しています。

はじめに

このたびは、NJ/NX シリーズ CPU ユニット、NY シリーズ産業用 PC をお買い上げいただきまして、ありがとうございます。

本マニュアルは、高速アナログ検査ライブラリのファンクションブロック（以下、FB と省略する場合があります）を使用する上で、必要な情報を記載しています。お使いになる前に本マニュアルをよく読んで、機能・性能などを十分ご理解いただき、システムの構築にご活用ください。

本マニュアルは、FB の仕様について記載しています。コントローラや各ユニット、コンポの使用上の制約事項、組み合わせによる制約事項などについては記載されていません。

ご使用にあたりましては、必ず各製品のユーザーズマニュアルをお読みいただきますようお願い申し上げます。

また、お読みになった後も本マニュアルは大切に保管して、いつも手元においてお使いください。

本ライブラリの特長

高速アナログ検査ライブラリは、NX シリーズ 高速アナログ入力ユニット 形 NX-HAD□□□のアナログ入力値を時系列的に記録します。

データの最大値、最小値、平均値など特徴量の算出、マスタデータとの比較判定やデータのファイル保存などの生産工程における製品検査に必要な機能を提供します。

対象となる読者の方々

本マニュアルは、次の方を対象に記述しています。

電気の知識（電気工事士あるいは同等の知識）を有する方で

- FA 機器の導入を担当される方
- FA システムを設計される方
- FA 機器を設置、接続される方
- FA 現場を管理される方

また、プログラミング言語に関して、国際標準規格 IEC 61131-3 もしくは国内標準規格 JIS B 3503 に規定される内容をご理解いただいている方を対象に記述しています。

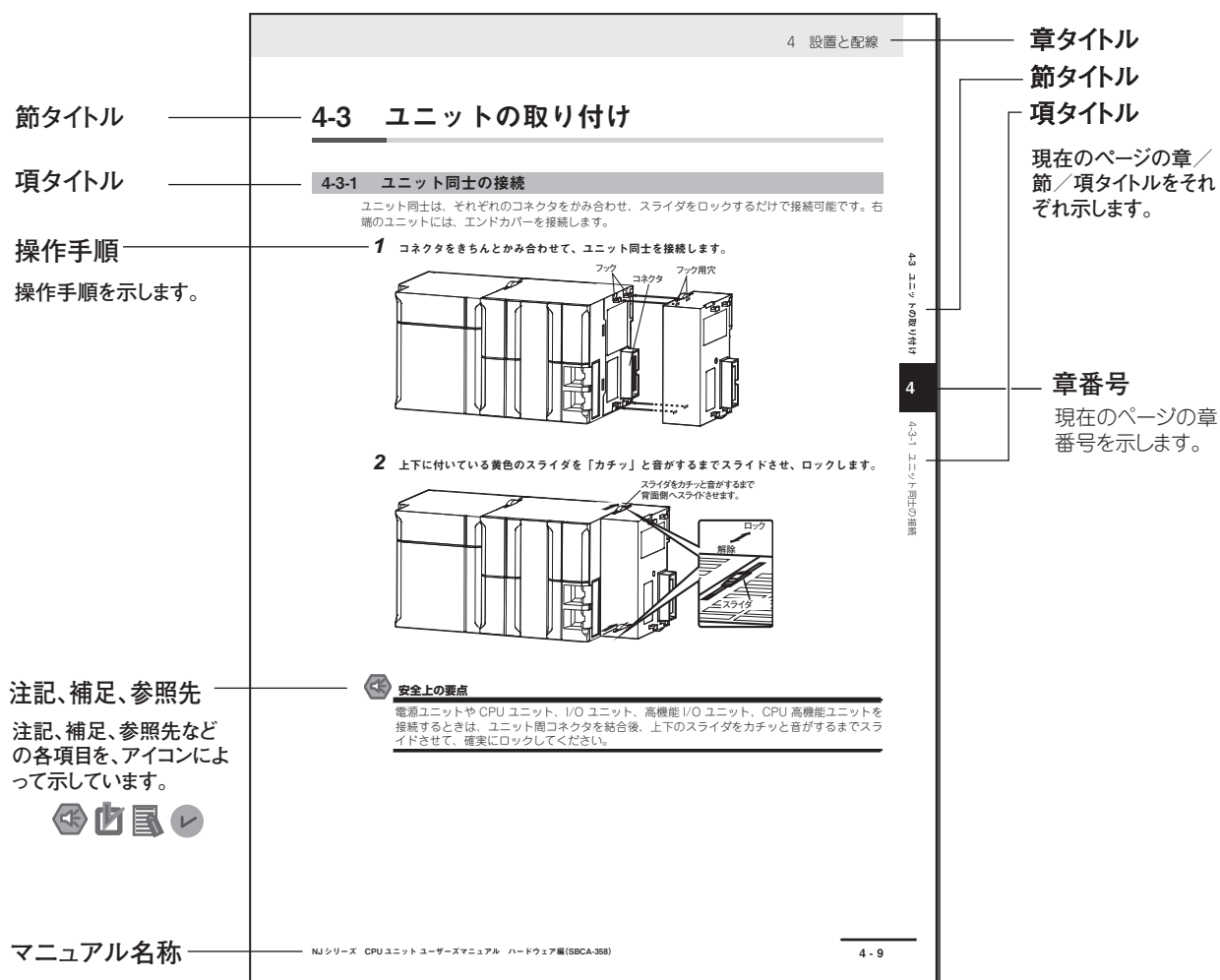
対象となる商品

本ライブラリが対応する、NJ/NX シリーズ CPU ユニットや NY シリーズ産業用 PC、および Sysmac Studio の形式およびバージョンについては、『SYSMAC-XR□□□ Sysmac Library カタログ (SBCZ-049)』の「Sysmac Library バージョン情報」を参照してください。カタログは、当社 Web サイト (<https://www.fa.omron.co.jp/products/family/3459/download/catalog.html>) からダウンロードできます。

マニュアルの読み方

ページ構成

このマニュアルの各ページの構成は次のとおりです。



(注) このページは、説明用の見本です。実際の内容とは異なります。

アイコン

このユーザーズマニュアルで使われているアイコンには、次のような意味があります。



安全上の要点

製品を安全に使用するために実施または、回避すべきことを示します。



使用上の注意

製品が動作不能、誤動作、または性能や機能への悪影響を予防するために実施または回避すべきことを示します。



参考

必要に応じて読んでいただきたい項目です。
知っておくと便利な情報や、使用する上で参考となる内容について説明しています。



バージョン関連情報

CPU ユニットや産業用 PC、Sysmac Studio のバージョンによる性能や機能の違いについて説明しています。

目次

はじめに.....	1
本ライブラリの特長	1
対象となる読者の方々	1
対象となる商品	1
マニュアルの読み方	2
ページ構成	2
アイコン	3
ご承諾事項	8
安全上の注意	10
安全に使用していただくための表示と意味	10
図記号の説明	10
注意	10
使用上の注意	12
ライブラリの使用時	12
関連マニュアル	13
マニュアル改訂履歴	16
目次構成	17

第 1 章 Sysmac Library の使用手順

1-1 インストーラからインストールした Sysmac Library の使用手順	1 - 2
1-1-1 Sysmac Library を新規で使用する場合	1 - 2
1-1-2 Sysmac Library をバージョンアップして使用する場合	1 - 4
1-2 CPU ユニットまたは産業用 PC 内の Sysmac Library の使用手順	1 - 6

第 2 章 高速アナログ検査ライブラリ

2-1 機能概要	2 - 2
2-1-1 システム構成例	2 - 2
2-1-2 ライブラリ構成	2 - 2
2-1-3 データフロー、FB/FUN 構成	2 - 3

第 3 章 FB の共通仕様

3-1 共通変数	3 - 2
3-1-1 入力変数と出力変数の定義	3 - 2
3-1-2 Execute 型 FB	3 - 3
3-1-3 Enable 型 FB	3 - 5
3-2 注意事項	3 - 7
3-2-1 ネスティング	3 - 7
3-2-2 動作オプション	3 - 7
3-2-3 FB の再起動	3 - 7

第 4 章 FB/FUN の個別仕様(NX_HAD)

DeviceVariableToArray_***	4 - 2
ライブラリ情報	4 - 2
入力変数	4 - 2
出力変数	4 - 3
入出力変数	4 - 3
機能	4 - 3
使用上の注意	4 - 4
サンプルプログラム 1	4 - 4
サンプルプログラム 2	4 - 7
ScaleTrans_HAD	4 - 11
ライブラリ情報	4 - 11
入力変数	4 - 11
出力変数	4 - 12
入出力変数	4 - 12
機能	4 - 12
サンプルプログラム	4 - 13
LimitAlarm_HAD	4 - 14
ライブラリ情報	4 - 14
入力変数	4 - 14
出力変数	4 - 15
入出力変数	4 - 15
機能	4 - 16
タイミングチャート	4 - 17
トラブルシューティング	4 - 18
サンプルプログラム	4 - 19

第 5 章 FB/FUN の個別仕様(DataRecorder)

TrigControl	5 - 2
ライブラリ情報	5 - 2
入力変数	5 - 2
出力変数	5 - 3
入出力変数	5 - 3
構造体	5 - 3
機能	5 - 4
タイミングチャート	5 - 6
トラブルシューティング	5 - 7
サンプルプログラム	5 - 7
DataRecorder	5 - 12
ライブラリ情報	5 - 12
入力変数	5 - 12
出力変数	5 - 13
入出力変数	5 - 13
構造体	5 - 14
機能	5 - 14
タイミングチャート	5 - 16
トラブルシューティング	5 - 17
サンプルプログラム	5 - 17
LimitTest	5 - 18
ライブラリ情報	5 - 18
入力変数	5 - 18
出力変数	5 - 19
入出力変数	5 - 19
構造体	5 - 19
機能	5 - 19
タイミングチャート	5 - 21
使用上の注意	5 - 21
トラブルシューティング	5 - 22

サンプルプログラム	5 - 22
CalcFeatureValues	5 - 23
ライブラリ情報	5 - 23
入力変数	5 - 23
出力変数	5 - 24
入出力変数	5 - 24
構造体	5 - 24
機能	5 - 24
タイミングチャート	5 - 25
使用上の注意	5 - 26
トラブルシューティング	5 - 26
サンプルプログラム	5 - 26
LogDataToCSV	5 - 31
ライブラリ情報	5 - 31
入力変数	5 - 31
出力変数	5 - 32
入出力変数	5 - 32
構造体	5 - 32
機能	5 - 32
参考	5 - 34
タイミングチャート	5 - 34
使用上の注意	5 - 35
トラブルシューティング	5 - 35
サンプルプログラム	5 - 36
CSVToLogData	5 - 37
ライブラリ情報	5 - 37
入力変数	5 - 37
出力変数	5 - 37
入出力変数	5 - 38
構造体	5 - 38
機能	5 - 38
タイミングチャート	5 - 39
使用上の注意	5 - 40
トラブルシューティング	5 - 40
サンプルプログラム	5 - 41

付録

A-1 ライブラリ情報の参照	A - 2
A-1-1 ライブラリの属性、FB や FUN の属性	A - 2
A-1-2 ライブラリの属性、FB や FUN の属性を参照する方法	A - 2
A-2 FB や FUN のソースコードの参照	A - 5

索引

ご承諾事項

「当社商品」について特別の合意がない場合には、お客様のご購入先にかかわらず、本ご承諾事項記載の条件を適用いたします。

定義

本ご承諾事項中の用語の定義は次のとおりです。

- ・「当社商品」：「当社」の FA システム機器、汎用制御機器、センシング機器、電子・機構部品
- ・「カタログ等」：「当社商品」に関する、ベスト制御機器オムロン、電子・機構部品総合カタログ、その他のカタログ、仕様書、取扱説明書、マニュアル等であって電磁的方法で提供されるものも含みます。
- ・「利用条件等」：「カタログ等」に記載の、「当社商品」の利用条件、定格、性能、動作環境、取り扱い方法、利用上の注意、禁止事項その他
- ・「お客様用途」：「当社商品」のお客様におけるご利用方法であって、お客様が製造する部品、電子基板、機器、設備またはシステム等への「当社商品」の組み込み、または利用を含みます。
- ・「適合性等」：「お客様用途」での「当社商品」の(a)適合性、(b)動作、(c)第三者の知的財産の非侵害、(d)法令の遵守および(e)各種規格の遵守

記載事項のご注意

「カタログ等」の記載内容については次の点をご理解ください。

- ・定格値および性能値は、単独試験における各条件のもとで得られた値であり、各定格値および性能値の複合条件のもとで得られる値を保証するものではありません。
- ・参考データはご参考として提供するもので、その範囲で常に正常に動作することを保証するものではありません。
- ・利用事例はご参考ですので、「当社」は「適合性等」について保証いたしかねます。
- ・「当社」は、改善や当社都合等により、「当社商品」の生産を中止し、または「当社商品」の仕様を変更することがあります。

ご利用にあたってのご注意

ご採用およびご利用に際しては次の点をご理解ください。

- ・定格・性能ほか「利用条件等」を遵守しご利用ください。
- ・お客様ご自身にて「適合性等」をご確認いただき、「当社商品」のご利用の可否をご判断ください。「当社」は「適合性等」を一切保証いたしかねます。
- ・「当社商品」がお客様のシステム全体の中で意図した用途に対して、適切に配電・設置されていることをお客様ご自身で、必ず事前に確認してください。
- ・「当社商品」をご使用の際には、(i)定格および性能に対し余裕のある「当社商品」のご利用、冗長設計などの安全設計、(ii)「当社商品」が故障しても、「お客様用途」の危険を最小にする安全設計、(iii)利用者に危険を知らせるための、安全対策のシステム全体としての構築、(iv)「当社商品」および「お客様用途」の定期的な保守、の各事項を実施してください。
- ・「当社」は DDoS 攻撃（分散型 DoS 攻撃）、コンピュータウイルスその他の技術的な有害プログラム、不正アクセスにより、「当社商品」、インストールされたソフトウェア、またはすべてのコンピュータ機器、コンピュータプログラム、ネットワーク、データベースが感染したとしても、そのことにより直接または間接的に生じた損失、損害その他の費用について一切責任を負わないものとします。

お客様ご自身にて、①アンチウイルス保護、②データ入出力、③紛失データの復元、④「当社商品」またはインストールされたソフトウェアに対するコンピュータウイルス感染防止、⑤「当社商品」に対する不正アクセス防止についての十分な措置を講じてください。

- 「当社商品」は、一般工業製品向けの汎用品として設計製造されています。従いまして、次に掲げる用途での使用は意図しておらず、お客様が「当社商品」をこれらの用途に使用される際には、「当社」は「当社商品」に対して一切保証をいたしません。ただし、次に掲げる用途であっても「当社」の意図した特別な商品用途の場合や特別の合意がある場合は除きます。
 - a) 高い安全性が必要とされる用途（例：原子力制御設備、燃焼設備、航空・宇宙設備、鉄道設備、昇降設備、娯楽設備、医用機器、安全装置、その他生命・身体に危険が及ぶ用途）
 - b) 高い信頼性が必要な用途（例：ガス・水道・電気等の供給システム、24 時間連続運転システム、決済システムほか権利・財産を取扱う用途など）
 - c) 厳しい条件または環境での用途（例：屋外に設置する設備、化学的汚染を被る設備、電磁的妨害を被る設備、振動・衝撃を受ける設備など）
 - d) 「カタログ等」に記載のない条件や環境での用途
- 上記の(a)から(d)に記載されている他、「本カタログ等記載の商品」は自動車（二輪車含む。以下同じ）向けではありません。自動車に搭載する用途には利用しないで下さい。自動車搭載用商品については当社営業担当者にご相談ください。

保証条件

「当社商品」の保証条件は次のとおりです。

- 保証期間 ご購入後 1 年間といたします。
（ただし「カタログ等」に別途記載がある場合を除きます。）
- 保証内容 故障した「当社商品」について、以下のいずれかを「当社」の任意の判断で実施します。
 - a) 当社保守サービス拠点における故障した「当社商品」の無償修理
（ただし、電子・機構部品については、修理対応は行いません。）
 - b) 故障した「当社商品」と同数の代替品の無償提供
- 保証対象外 故障の原因が次のいずれかに該当する場合は、保証いたしません。
 - a) 「当社商品」本来の使い方以外のご利用
 - b) 「利用条件等」から外れたご利用
 - c) 本ご承諾事項「ご利用にあたってのご注意」に反するご利用
 - d) 「当社」以外による改造、修理による場合
 - e) 「当社」以外の者によるソフトウェアプログラムによる場合
 - f) 「当社」からの出荷時の科学・技術の水準では予見できなかった原因
 - g) 上記のほか「当社」または「当社商品」以外の原因（天災等の不可抗力を含む）

責任の制限

本ご承諾事項に記載の保証が、「当社商品」に関する保証のすべてです。

「当社商品」に関連して生じた損害について、「当社」および「当社商品」の販売店は責任を負いません。

輸出管理

「当社商品」または技術資料を、輸出または非居住者に提供する場合は、安全保障貿易管理に関する日本および関係各国の法令・規制を遵守ください。お客様が法令・規則に違反する場合には、「当社商品」または技術資料をご提供できない場合があります。

安全上の注意

安全に使用していただくための表示と意味

本マニュアルでは、NJ/NX シリーズ CPU ユニット、NY シリーズ産業用 PC 上で本ライブラリを安全に使用していただくために、注意事項を次のような表示と図記号で示しています。
ここで示した注意事項は、安全に関する重大な内容を記載しています。必ず守ってください。
表示と意味は次のとおりです。

 警告	正しい取り扱いをしなければ、この危険のために、軽傷・中程度の傷害を負ったり万一の場合には重傷や死亡に至ったりするおそれがあります。また、同様に重大な物的損害をもたらすおそれがあります。
 注意	正しい取り扱いをしなければ、この危険のために、時に軽傷・中程度の傷害を負ったり、あるいは物的損害を受けたりするおそれがあります。

図記号の説明

	○に斜線の記号は、禁止を意味しています。 具体的な内容は、○の中と文章で示します。 左図の場合は、「分解禁止」を表します。
	△記号は、注意（警告を含む）を意味しています。 具体的な内容は、△の中と文章で示します。 左図の場合は、「感電注意」を表します。
	△記号は、注意（警告を含む）を意味しています。 具体的な内容は、△の中と文章で示します。 左図の場合は、「一般的な注意」を表します。
	●記号は、強制を意味しています。 具体的な内容は、●の中と文章で示します。 左図の場合は、「一般的な強制事項」を表します。

注意

注意

ライブラリを使用する前に、関係する各マニュアルの記載事項をよく読んでください。

非常停止回路、インタロック回路、リミット回路など、安全保護に関する回路は、必ず外部の制御回路で構成してください。

作成したユーザプログラムおよび各種データ、設定値は十分な動作確認を行ったあと、本運転に移行してください。



Sysmac ライブラリ、およびそのマニュアルは、本マニュアルの「対象となる読者の方々 (P.1)」に該当する方、もしくはその方の指導を受けた方以外はご使用にならないでください。



ユーザプログラムやパラメータの値が、装置のスペックや動作方法において適切であるかは、ユーザ自身で確認してください。



-
- サンプルプログラムは、本ライブラリの機能を使用する部分の記述例です。
 - 実際の装置で使用する場合は、装置の動作に関する安全回路やインタロックプログラム、ほかの機器からの入出力、制御手順等に関するユーザプログラムを併用してください。
 - サンプルプログラムを使用してプログラムを作成する際は、サンプルプログラムの内容を十分に理解してください。
 - 実際の装置が意図した動作をするように、ユーザプログラムを作成してください。
 - 作成したユーザプログラムは、十分な動作確認を行ったあと、本運転に移行してください。
-



使用上の注意

ライブラリの使用時

- 入力パラメータの値は有効範囲内で指定してください。
- 出力変数「Enabled」や「ENO」を持つFBやFUNでは、「Enabled」や「ENO」の値がFALSEの場合に、FBやFUNの演算結果を制御対象への指令値として用いないでください。
- 「Execute」を持つFBで、同一インスタンスによる再起動は行わないでください。FBの出力値が初期値に戻ります。

関連マニュアル

本マニュアルに関連するマニュアルは、下表のとおりです。あわせてご覧ください。

マニュアル名称	Man. No.	形式	用途	内容
NJ/NX シリーズ CPU ユニット ユーザーズマニュアル ソフトウェア編	SBCA-467	形 NX701-□□□□ 形 NX102-□□□□ 形 NX1P2-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	NJ/NX シリーズ CPU ユニットのプログラミング/システムの立ち上げについて知りたいとき。 おもにソフトウェアに関する情報。	NJ/NX シリーズ CPU ユニットに関して、以下の内容を説明します。 • CPU ユニットの動作 • CPU ユニットの機能 • 初期設定 • IEC 61131-3 ベースの言語仕様とプログラミング
NJ/NX シリーズ コマンドリファレンス マニュアル 基本編	SBCA-468	形 NX701-□□□□ 形 NX102-□□□□ 形 NX1P2-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	NJ/NX シリーズの基本命令仕様の詳細について知りたいとき。	各命令（IEC 61131-3 仕様）の詳細を説明します。
NJ/NX シリーズ トラブルシューティング マニュアル	SBCA-469	形 NX701-□□□□ 形 NX102-□□□□ 形 NX1P2-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	NJ/NX シリーズで検出する異常の詳細について知りたいとき。	NJ/NX シリーズ システムにて検出する異常管理の考え方と各異常項目について説明します。
NJ/NY シリーズ NC 統合コントローラ ユーザーズマニュアル	SBCE-428	形 NJ501-5300 形 NY532-5400	NJ/NY シリーズで数値制御を行うとき。	数値制御機能について説明します。
NJ/NY シリーズ コマンドリファレンスマニュアル G コード編	SBCE-429	形 NJ501-5300 形 NY532-5400	G コード/M コード命令仕様の詳細について知りたいとき。	G コード/M コード命令の詳細を説明します。
Sysmac Studio Version 1 オペレーションマニュアル	SBCA-470	形 SYSMAC-SE2□□□	Sysmac Studio の操作方法、機能について知りたいとき。	Sysmac Studio の操作方法について説明します。
CNC Operator オペレーションマニュアル	SBCE-430	形 SYSMAC-RTNC0□□□□	CNC Operator の概要、使用方法について知りたいとき。	CNC Operator の概要、インストール方法、基本操作、接続操作、主な機能の操作方法などについて説明します。
NJ/NX シリーズ CPU ユニット ユーザーズマニュアル モーション制御編	SBCE-433	形 NX701-□□□□ 形 NX102-□□□□ 形 NX1P2-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	モーション制御の設定やプログラミングの考え方について知りたいとき。	モーション制御のための CPU ユニットの設定や動作、プログラミングの考え方について説明します。
NJ/NX シリーズ コマンドリファレンス マニュアル モーション編	SBCE-434	形 NX701-□□□□ 形 NX102-□□□□ 形 NX1P2-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	モーション命令仕様の詳細について知りたいとき。	各モーション命令の詳細を説明します。

マニュアル名称	Man. No.	形式	用途	内容
NJ シリーズ CPU ユニッ ト ユーザーズマニュアル ハードウェア編	SBCA-466	形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	NJ シリーズ CPU ユニットの 概要／設計／取 付／保守などの 基本的な仕様 について知りたい とき。 おもにハードウ ェアに関する情 報。	NJ シリーズのシステム全 体概要、および CPU ユニ ットに関して、以下の内容 を説明します。 ・ 特長やシステム構成 ・ 概要 ・ 各部の名称と機能 ・ 一般仕様 ・ 設置と配線 ・ 保守点検
NY シリーズ IPC マシンコントローラ 産業用パネル型 PC / 産 業用 ボックス型 PC ユーザーズマニュアル ソフトウェア編	SBCA-436	形 NY532-□□□□ 形 NY512-□□□□	NY シリーズ産 業用 PC のコン トローラ機能の プログラミング ／システムの立 ち上げについて 知りたいとき。	NY シリーズのコントローラ機能に関して、以下の内 容を説明します。 ・ コントローラの動作 ・ コントローラの機能 ・ コントローラの設定 ・ IEC 61131-3 ベースの 言語仕様とプログラミ ング
NY シリーズ コマンドリファレンス マニュアル 基本編	SBCA-437	形 NY532-□□□□ 形 NY512-□□□□	NY シリーズ産 業用 PC の基本 命令仕様の詳細 について知りたい とき。	各命令 (IEC 61131-3 仕 様) の詳細を説明します。
NY シリーズ トラブルシューティング マニュアル	SBCA-438	形 NY532-□□□□ 形 NY512-□□□□	NY シリーズ産 業用 PC で検出 する異常の詳細 について知りたい とき。	NY シリーズ システムに て検出する異常管理の考 え方と各異常項目につい て説明します。
NY シリーズ IPC マシンコントローラ 産業用パネル型 PC / 産 業用 ボックス型 PC ユーザーズマニュアル モーション制御編	SBCE-379	形 NY532-□□□□ 形 NY512-□□□□	NY シリーズ産 業用 PC のモー ション制御の設 定やプログラミ ングの考え方 について知りたい とき。	モーション制御のための コントローラの設定や動 作、プログラミングの考 え方について説明します。
NY シリーズ コマンドリファレンス マニュアル モーション編	SBCE-380	形 NY532-□□□□ 形 NY512-□□□□	NY シリーズ産 業用 PC のモー ション命令仕様 の詳細について 知りたいとき。	各モーション制御命令の 詳細を説明します。
NY シリーズ IPC マシンコントローラ 産業用パネル型 PC / 産 業用 ボックス型 PC ユーザーズマニュアル 内蔵 EtherNet/IP™ポート 編	SBCD-369	形 NY532-□□□□ 形 NY512-□□□□	NY シリーズ産 業用 PC の内蔵 EtherNet/IP ポ ートを使用する とき。	内蔵 EtherNet/IP ポート に関して説明します。 基本設定、タグデータリン ク、その他の機能について 記述しています。

マニュアル名称	Man. No.	形式	用途	内容
NX シリーズ EtherCAT [®] カプラユニッ ト ユーザーズマニュアル	SBCD-361	形 NX-ECC20□	NX シリーズ EtherCAT カプ ラユニット、およ び EtherCAT ス レーブターミナ ルの使用方法に ついて知りたい とき。	NX シリーズ EtherCAT カプラユニットと NX ユ ニットで構成される EtherCAT スレーブター ミナルのシステム概要や 構成方法、および EtherCAT を介して NX ユニットを設定、制御、モ ニタするための、 EtherCAT カプラユニッ トのハードウェアや設定 方法、機能について説明し ます。
NX シリーズ アナログ I/O ユニット ユーザーズマニュアル 高速アナログ入力ユニッ ト編	SBCA-461	形 NX-HAD□□□	NX シリーズ 高 速アナログ入力 ユニットの使用 方法について知 りたいとき。	NX シリーズ 高速アナロ グ入力ユニットのハード ウェアや設定方法、機能に ついて説明します。

マニュアル改訂履歴

マニュアル改訂記号は、表表紙・裏表紙に記載されている Man. No.の後尾に付記されます。

Man. No.	SBCA-471B
----------	------------------

↑
改訂記号

改訂記号	改訂年月	改訂理由・改訂ページ
A	2018 年 5 月	初版
B	2019 年 1 月	対象形式追加

目次構成

		1
1	Sysmac Libraryの使用手順	2
2	高速アナログ検査ライブラリ	3
3	FBの共通仕様	4
4	FB/FUNの個別仕様(NX_HAD)	A
5	FB/FUNの個別仕様(DataRecorder)	A
A	付録	I
I	索引	

Sysmac Library の使用手順

インストーラからインストールした Sysmac Library、および CPU ユニットまたは産業用 PC 内の Sysmac Library の使用手順について説明します。

1-1	インストーラからインストールした Sysmac Library の使用手順	1 - 2
1-1-1	Sysmac Library を新規で使用する場合	1 - 2
1-1-2	Sysmac Library をバージョンアップして使用する場合	1 - 4
1-2	CPU ユニットまたは産業用 PC 内の Sysmac Library の使用手順.....	1 - 6

1-1 インストーラからインストールした Sysmac Library の使用手順

インストーラからインストールした Sysmac Library の使用手順を記載します。
次の 2 種類があります。

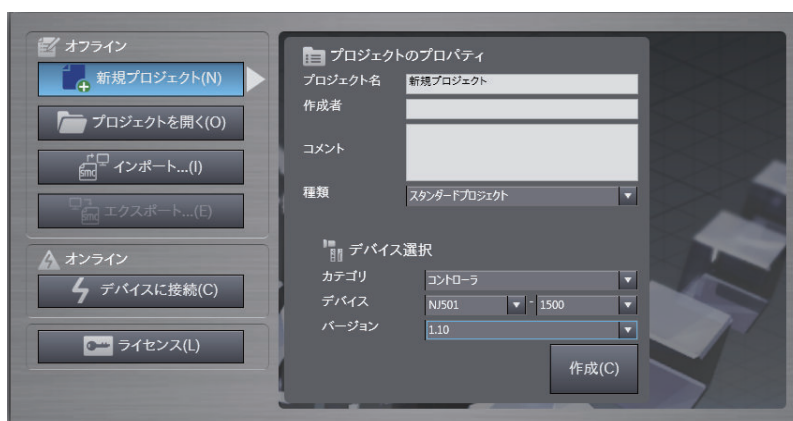
- Sysmac Library を新規で使用する場合
- Sysmac Library をバージョンアップして使用する場合

バージョン関連情報

Sysmac Library を使用できるのは、Sysmac Studio Ver.1.14 以降です。

1-1-1 Sysmac Library を新規で使用する場合

- 1 Sysmac Studio を起動し、Sysmac Library を使用するプロジェクトを開くか、もしくは新規で作成します。

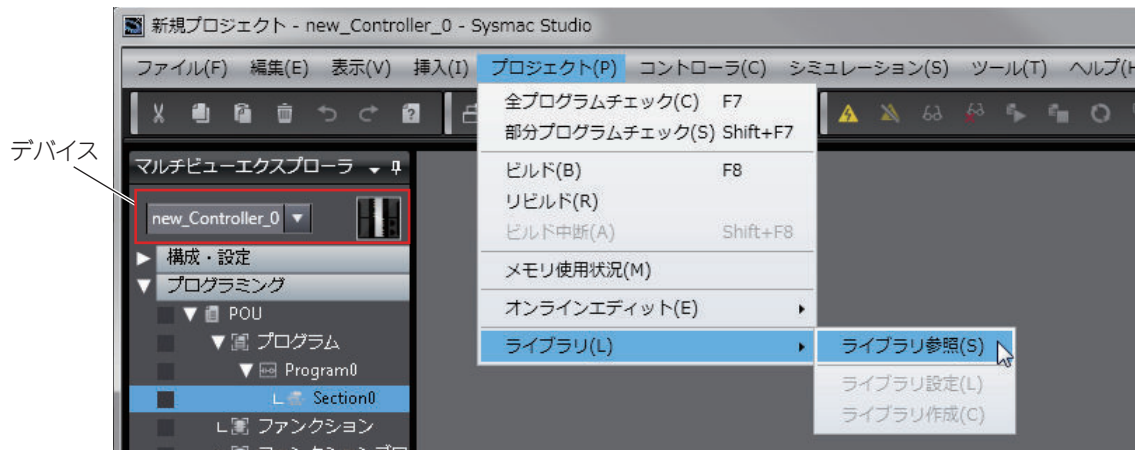


使用上の注意

新規でプロジェクトを作成する場合、Sysmac Library を利用するためには、以下の設定にしてください。以下の設定をしないと手順 2 以降の操作ができません。

- プロジェクトのプロパティの種類はスタンダードプロジェクトまたはライブラリプロジェクトを選択
- デバイス選択のカテゴリはコントローラを選択
- デバイス選択のコントローラおよびバージョンはを参照してください。

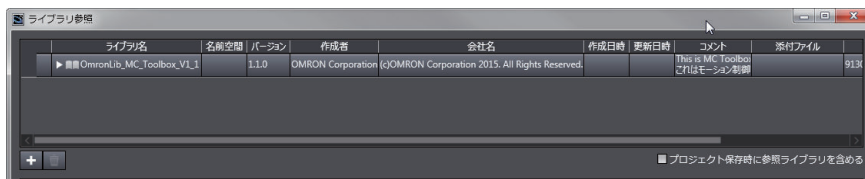
- 2 [プロジェクト] – [ライブラリ] – [ライブラリ参照] を選択します。



使用上の注意

プロジェクトに複数のデバイスを登録している場合は、現在選択しているデバイスが NJ/NX シリーズ CPU ユニットまたは NY シリーズ産業用 PC であることを確認してください。デバイスに NJ/NX シリーズ CPU ユニットまたは NY シリーズ産業用 PC を選択しないとライブラリを参照するメニューが表示されません。選択しているデバイスが NJ/NX シリーズ CPU ユニットまたは NY シリーズ産業用 PC の場合、マルチビューエクスプローラに表示されるデバイスのアイコンが、 になります。

3 リストに Sysmac Library を追加し、[OK] をクリックします。



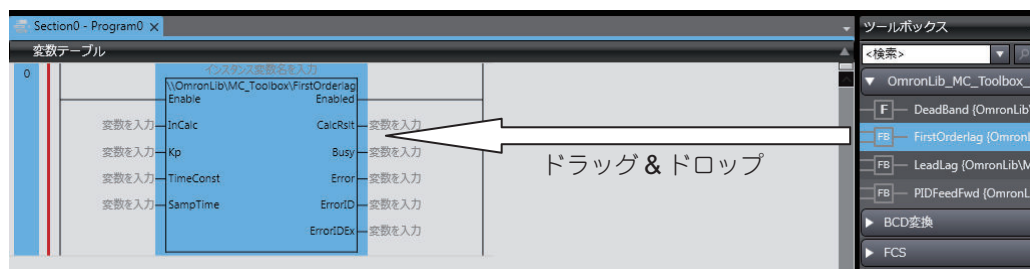
Sysmac Library がプロジェクトに読み出されます。

ラダーエディタや ST エディタを選択すると、Sysmac Library の FB や FUN がツールボックスに表示されます。

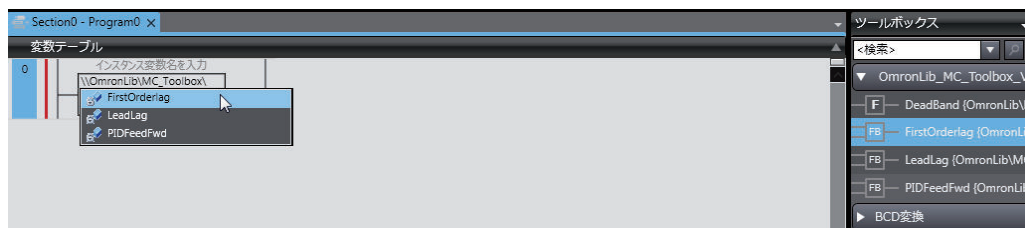
上記画面でのライブラリの追加方法および設定内容は、『Sysmac Studio Version 1 オペレーションマニュアル (SBCA-470)』を参照してください。

4 次の二つの方法のいずれかで Sysmac Library の FB や FUN を回路に挿入します。

- ツールボックスの FB や FUN を選択し、ラダーエディタ上にドラッグ&ドロップ。

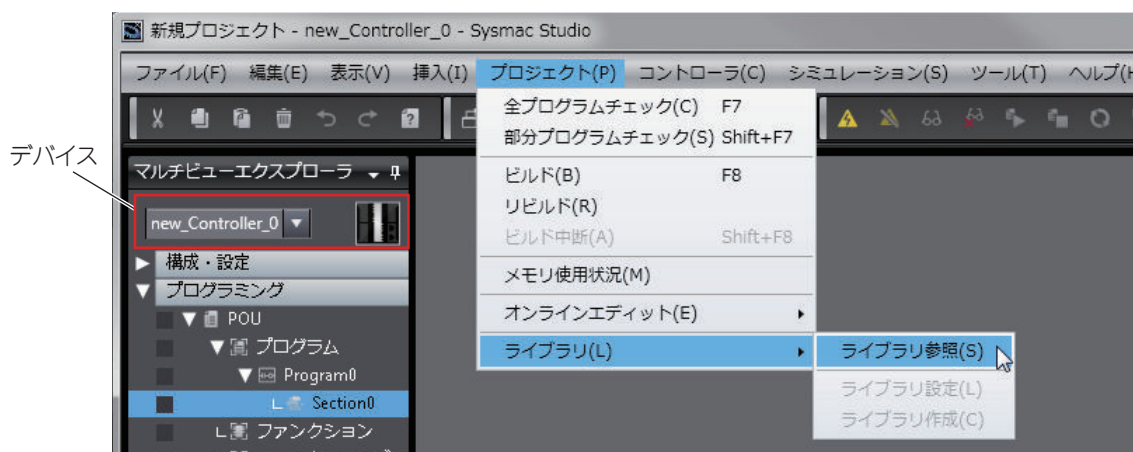


- ラダーエディタ上で右クリックし、メニューの「ファンクションブロックを挿入」を選択し、完全限定名(¥¥名前前空間名¥FB 名)を入力。



1-1-2 Sysmac Library をバージョンアップして使用する場合

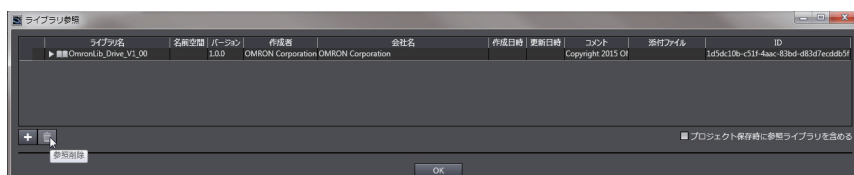
- 1 Sysmac Studio を起動し、旧バージョンの Sysmac Library を使用しているプロジェクトを開きます。
- 2 [プロジェクト] - [ライブラリ] - [ライブラリ参照] を選択します。



使用上の注意

プロジェクトに複数のデバイスを登録している場合は、現在選択しているデバイスが NJ/NX シリーズ CPU ユニットまたは NY シリーズ産業用 PC であることを確認してください。デバイスに NJ/NX シリーズ CPU ユニットまたは NY シリーズ産業用 PC を選択しないとライブラリを参照するメニューが表示されません。選択しているデバイスが NJ/NX シリーズ CPU ユニットまたは NY シリーズ産業用 PC の場合、マルチビューエクスプローラに表示されるデバイスのアイコンが、 になります。

- 3 旧バージョンの Sysmac Library を選択し、[参照削除] ボタンをクリックします。



- 4 リストに Sysmac Library を追加し、[OK] をクリックします。



使用上の注意

Sysmac Library をバージョンアップ後、全プログラムチェックを実行し、ビルドウィンドウのプログラムチェックの結果にエラーがないことを確認してください。
メインメニューから [プロジェクト] - [全プログラムチェック] を選択します。

1-2 CPU ユニットまたは産業用 PC 内の Sysmac Library の使用手順

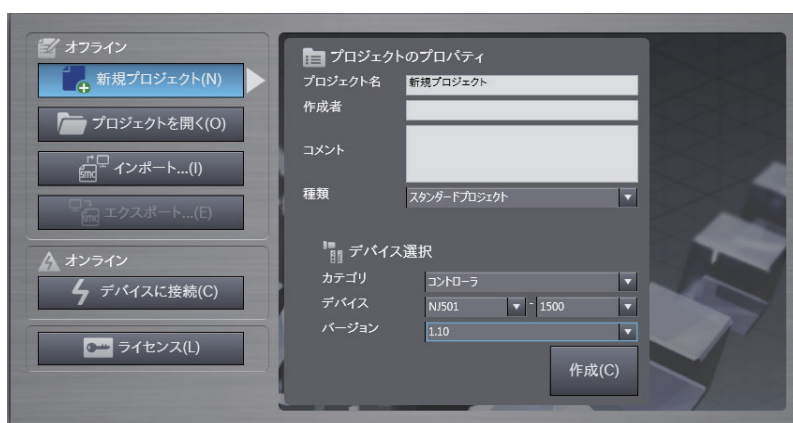
Sysmac Library をパソコンにインストールしていない場合でも、CPU ユニットまたは産業用 PC 内の Sysmac Library をパソコンにアップロードして使用できます。

CPU ユニットまたは産業用 PC 内の Sysmac Library の使用手順を、次に示します。

バージョン関連情報

Sysmac Library を使用できるのは、Sysmac StudioVer.1.14 以降です。

1 Sysmac Studio を起動し、Sysmac Library を使用するプロジェクトを新規で作成します。



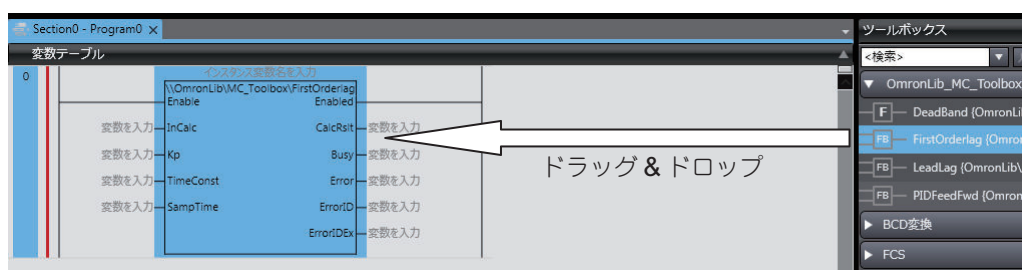
2 CPU ユニットまたは産業用 PC にオンライン接続します。

3 Sysmac Library を使用している POU をアップロードします。

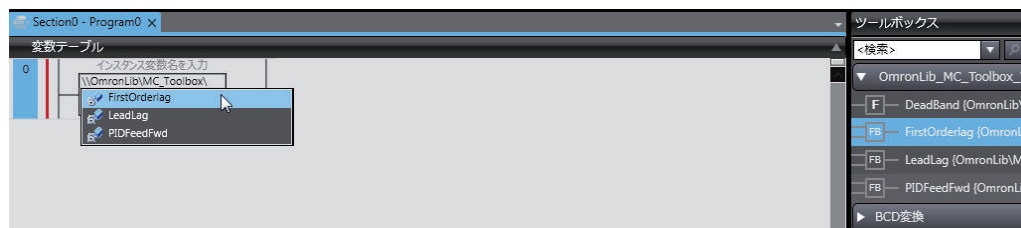
ラダーエディタや ST エディタを選択すると、アップロードした POU が使用している Sysmac Library の FB や FUN がツールボックスに表示されます。

4 次の二つの方法のいずれかで Sysmac Library の FB や FUN を回路に挿入します。

- ツールボックスの FB や FUN を選択し、ラダーエディタ上にドラッグ&ドロップ。



- ラダーエディタ上で右クリックし、メニューの「ファンクションブロックを挿入」を選択し、完全限定名（¥¥名前空間名¥FB 名）を入力。



使用上の注意

- アップロードした Sysmac Library のライブラリファイルが、パソコンの所定のフォルダにない場合、所定のフォルダにライブラリファイルがインストールされます。ライブラリファイルが、パソコンの所定のフォルダにある場合は何もインストールされません。所定のフォルダとは、インストーラでインストールした場合に、ライブラリファイルが保存されるフォルダです。
- CPU ユニットまたは産業用 PC 内の Sysmac Library をアップロードした場合は、インストーラでインストールした場合とは異なり、Sysmac Library のマニュアルとヘルプファイルはインストールされません。マニュアルとヘルプファイルが必要な場合は、インストーラでインストールしてください。

高速アナログ検査ライブラリ

高速アナログ検査ライブラリの各 FB に共通する仕様について、説明します。

2-1	機能概要	2 - 2
2-1-1	システム構成例	2 - 2
2-1-2	ライブラリ構成	2 - 2
2-1-3	データフロー、FB/FUN 構成	2 - 3

2-1 機能概要

高速アナログ検査ライブラリは、NX シリーズ 高速アナログ入力ユニット 形 NX-HAD□□□のアナログ入力値を時系列で記録します。

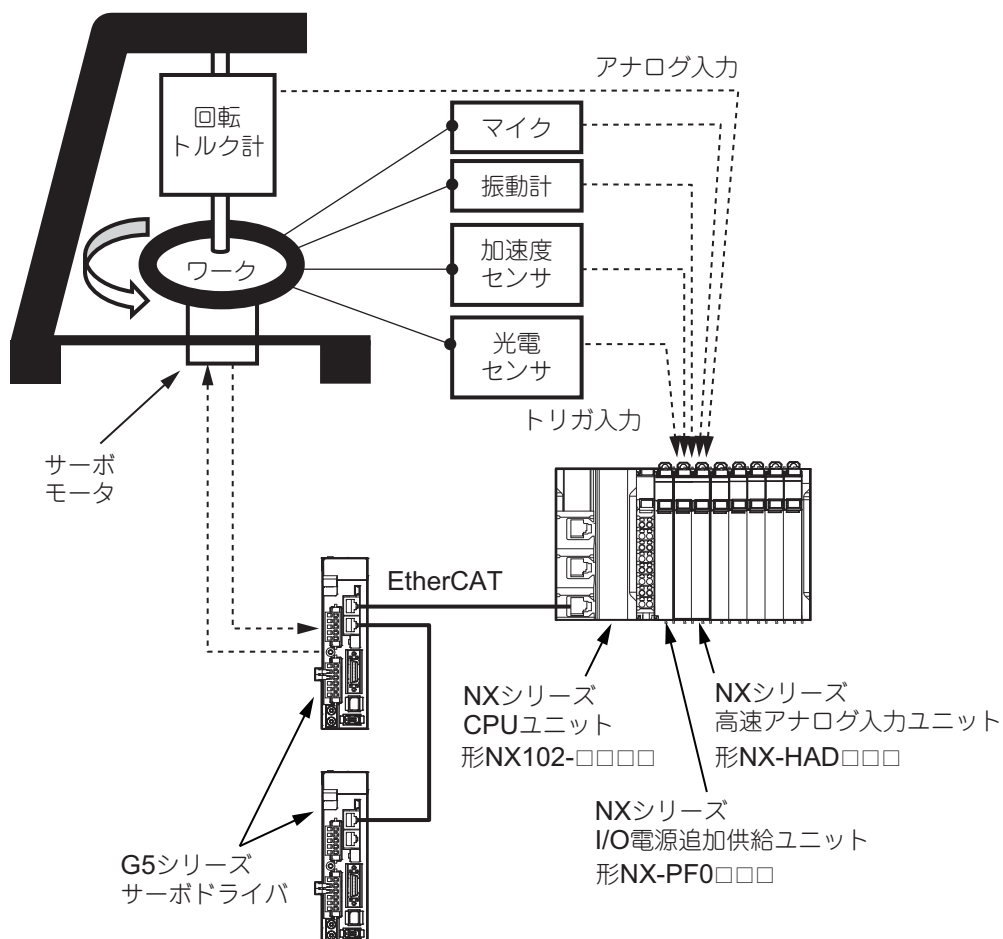
さらに、記録したデータの特徴量の算出や合否判定など、生産工程での製品検査に必要な各種機能を提供します。

2-1-1 システム構成例

本ライブラリを使用した特性検査機器のアプリケーションのシステム構成例を示します。

検査時に発生する音、振動、トルク値のアナログ入力信号を NX シリーズ 高速アナログ入力ユニットに取り込み、本ライブラリの FB/FUN でワークの良否判定を行います。

光電センサなどの入力をトリガ入力として使用することで、検査に必要なアナログ入力データが無駄なく取得できます。



2-1-2 ライブラリ構成

NX_HAD と DataRecorder の二つのライブラリファイルで構成されます。

同時、またはそれぞれ単体での利用が可能です。

各ライブラリファイルに含まれる FB/FUN は以下のとおりです。

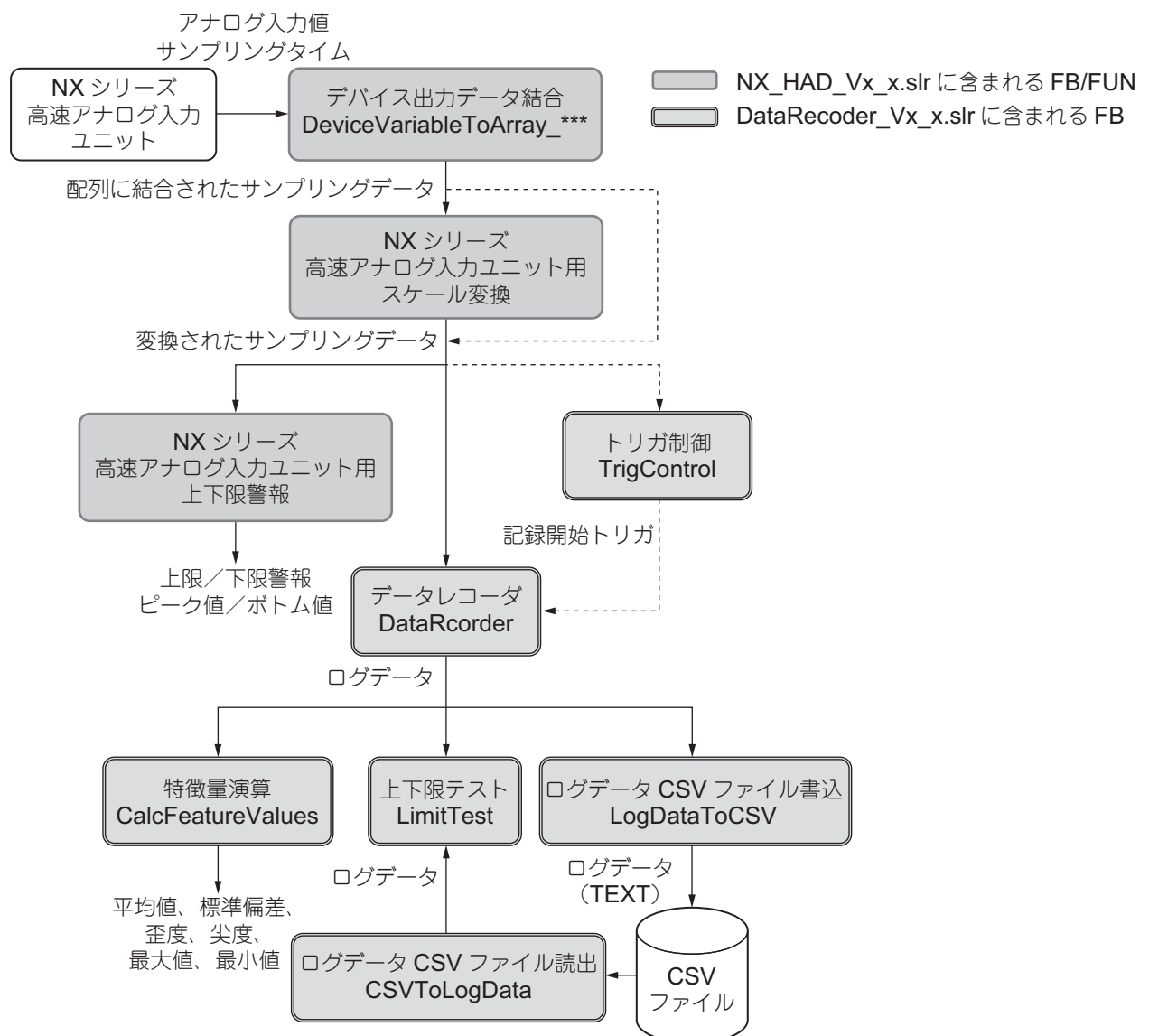
ライブラリファイル名	FB/FUN	FB/FUN 名
OmronLib_NX_HAD_Vx_x.slr*1	FUN	DeviceVariableToArray_***
	FUN	ScaleTrans_HAD
	FB	LimitAlarm_HAD
OmronLib_DataRecorder_Vx_x.slr*1	FB	TrigControl
	FB	DataRecorder
	FB	LimitTest
	FB	CalcFeatureValues
	FB	LogDataToCSV
	FB	CSVToLogData

*1. Vx_x はライブラリファイルのバージョンを示します。

2-1-3 データフロー、FB/FUN 構成

両ライブラリファイルを使用する際のデータの流れと構成を以下に示します。

- アナログ入力値を時系列的にロギングし、そのロギングデータに対する演算、検査、ファイル記録などを行いたいときは NX_HAD.slr および DataRecorder.slr ファイルの二つを使用します。
- 1 タスク周期分のアナログ入力値の警報判定や、ピーク・ボトム値の検出だけを行いたいときは、NX_HAD.slr ファイルだけ使用します。



FB の共通仕様

Sysmac Library の各 FB に共通する仕様について、説明します。

3-1	共通変数	3 - 2
3-1-1	入力変数と出力変数の定義	3 - 2
3-1-2	Execute 型 FB	3 - 3
3-1-3	Enable 型 FB	3 - 5
3-2	注意事項	3 - 7
3-2-1	ネスティング	3 - 7
3-2-2	動作オプション	3 - 7
3-2-3	FB の再起動	3 - 7

3-1 共通変数

複数の FB や FUN に共通して使われる変数(EN, Execute, Enable, Abort, ENO, Done, CalcRslt, Enabled, Busy, CommandAborted, Error, ErrorID, ErrorIDEx) の仕様を FUN、Execute 型 FB と Enable 型 FB に分けて説明します。

3-1-1 入力変数と出力変数の定義

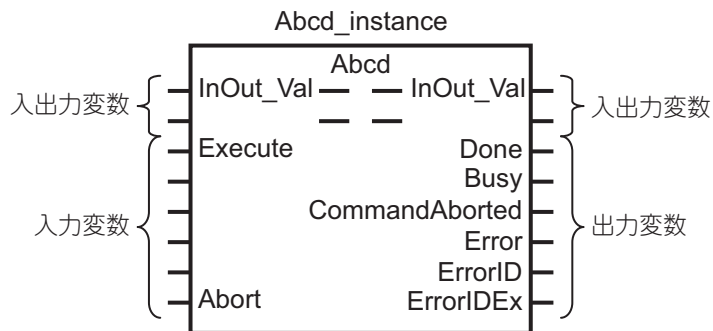
FB や FUN で使用される共通の入力変数と出力変数は以下のとおりです。

変数	入出力	データ型	使用する FB/FUN タイプ			名称	定義
			FB		FUN		
			Execute 型	Enable 型			
EN	入力	BOOL			○	起動	TRUE にすると処理を実行します。
Execute		BOOL	○			起動	立ち上がり時に処理を実行します。
Enable		BOOL		○		実行	TRUE にすると処理を実行します。
Abort		BOOL	○			中断	処理を中断します。 中断方法を選択することができます。
ENO	出力	BOOL			○	完了	処理が正常終了したときに TRUE になります。 処理が異常終了、実行中、または実行条件が非成立のときに FALSE になります。
Done		BOOL	○			完了	処理が正常終了したときに TRUE になります。 処理が異常終了、実行中、または実行条件が非成立のときに FALSE になります。
Busy		BOOL	○	○		実行中	処理が実行中の場合は TRUE になります。 処理が実行中ではない場合は FALSE になります。
CalcRslt		LREAL		○		演算結果	演算結果が出力されます。
Enabled		BOOL		○		有効	出力が有効な場合に TRUE になります。 モーション制御や温度制御のように制御量を演算する場合に用います。
Command Aborted		BOOL	○			実行中断	処理が中断されたときに TRUE になります。 次に処理を再実行したときに FALSE になります。
Error		BOOL	○	○		エラー	エラーが発生したときに TRUE になります。 処理が正常終了、実行中、または実行条件が非成立のときに FALSE になります。
ErrorID		WORD	○	○		エラーコード	エラーコードが出力されます。

変数	入出力	データ型	使用する FB/FUN タイプ			名称	定義
			FB		FUN		
			Execute 型	Enable 型			
ErrorIDEx		DWORD	○	○		拡張エラーコード	拡張エラーコードが出力されます。

3-1-2 Execute 型 FB

- Execute（起動）の立ち上がりエッジを検出して処理を開始します。
- 起動を受け付けたときに、Busy（実行中）は TRUE となります。処理が正常に完了した場合、Busy は FALSE になり、Done（完了）は TRUE となります。
- 同一インスタンスの FB を連続して起動する場合は、前回実行時の Done（完了）が FALSE になった後、1 タスク周期以上あけてから次の Execute（起動）を TRUE にしてください。
- 出力変数に CommandAborted（実行中断）を持つ FB の場合、処理が中断された場合に CommandAborted が TRUE になり、Busy は FALSE になります。
- FB で異常が発生した場合、Error（エラー）が TRUE になり、Busy は FALSE になります。
- モーション制御や温度制御の演算結果を出力する FB の場合、FB の処理を中断するための BOOL 型の入力変数 Abort（中断）があります。Abort を TRUE にすると、CommandAborted（実行中断）が TRUE になり FB の処理を中断します。

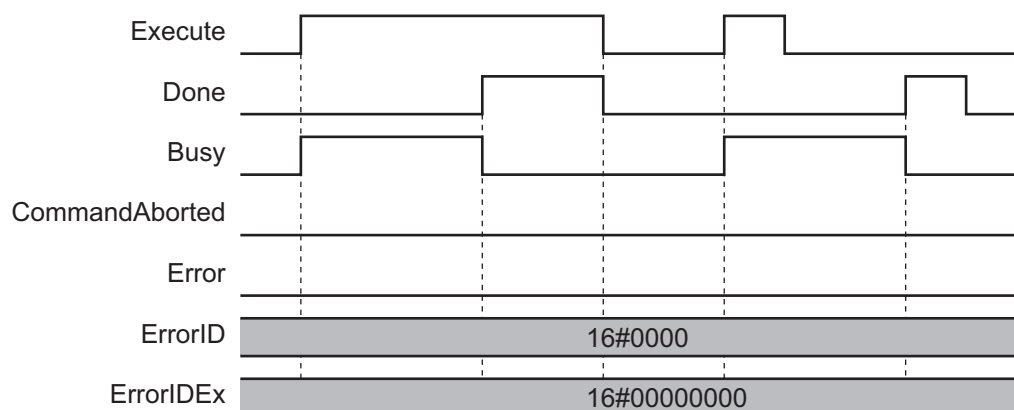


- Execute が TRUE の状態で Done, CommandAborted, Error が TRUE になった場合に Execute を FALSE にすると、Done, CommandAborted, Error も FALSE になります。
- Execute が FALSE の状態で Done, CommandAborted, Error が TRUE になった場合、Done, CommandAborted, Error は、1 タスク周期だけ TRUE になります。
- FB で異常が発生した場合、ErrorID（エラーコード）、ErrorIDEx（拡張エラーコード）に該当するエラーコード、拡張エラーコードが設定されます。これらは、Error が FALSE になったあとも保持され、Execute が FALSE → TRUE になったときに ErrorID は 16#0000、ErrorIDEx は 16#0000 0000 が設定されます。

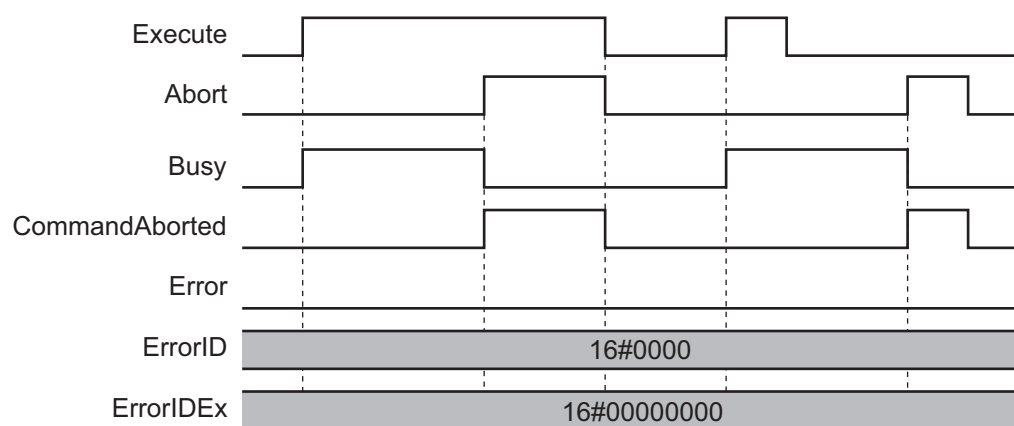
タイミングチャート

正常に完了した場合、実行を中止した場合、実行を中断した場合、異常が発生した場合のタイミングチャートを示します。

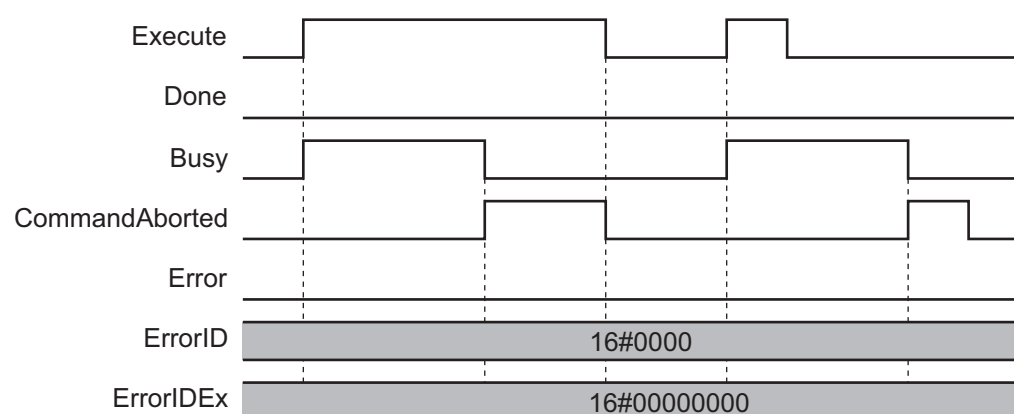
● 正常に完了した場合



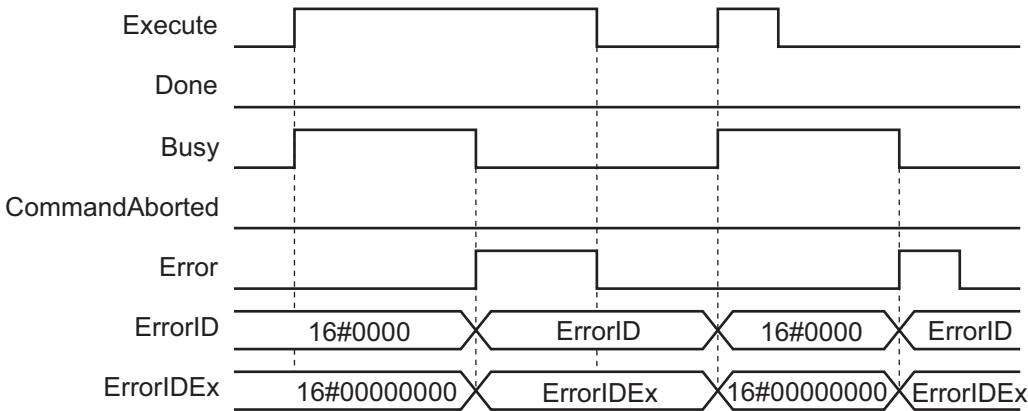
● 実行を中止した場合



● 実行を中断した場合

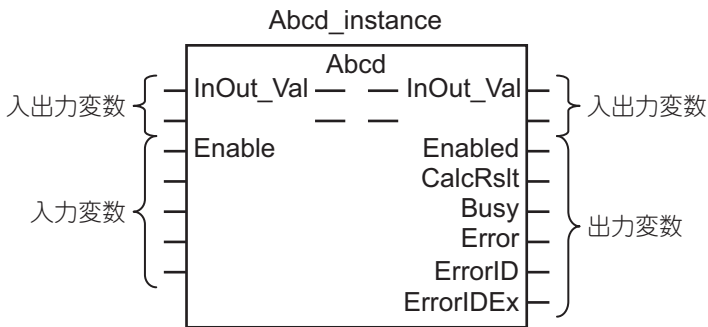


● 異常が発生した場合



3-1-3 Enable 型 FB

- Enable（実行）が TRUE の間、処理が実行されます。
- 起動を受け付けた場合に、Busy（実行中）は TRUE となります。出力値を演算している間、Enabled（有効）が TRUE になります。
- FB で異常が発生した場合、Error（エラー）が TRUE になり、Busy, Enabled は FALSE になります。Enable を TRUE → FALSE にすると、Enabled, Busy, Error は FALSE になります。

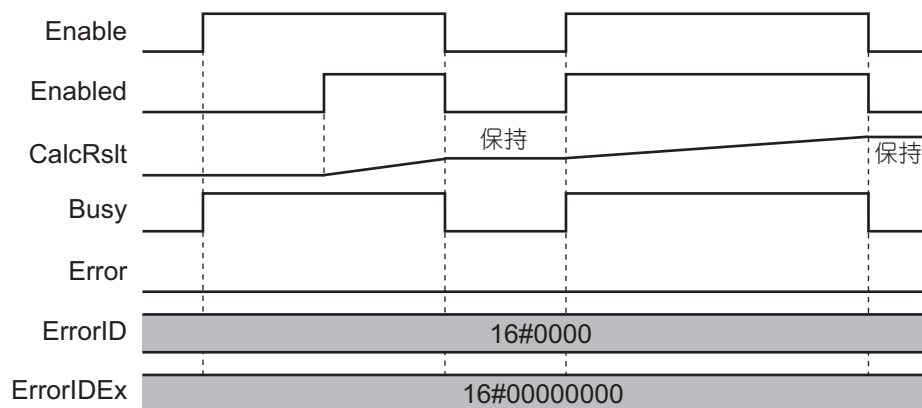


- FB で異常が発生した場合、ErrorID（エラーコード）、ErrorIDEx（拡張エラーコード）に該当するエラーコード、拡張エラーコードが設定されます。これらは、Error が FALSE になったあとも保持され、Enable が FALSE → TRUE になったときに ErrorID は 16#0000、ErrorIDEx は 16#0000 0000 が設定されます。
- モーション制御や温度制御など制御量を演算する FB では、CalcRslt（演算結果）が不正な値の場合、Enabled が FALSE になります。この時の CalcRslt は使用しないでください。また、FB が正常完了、または異常発生後、Enable が FALSE から TRUE になるまで CalcRslt の値は保持されます。Enable を TRUE にしたのが同一インスタンスの FB の場合は、保持された CalcRslt の値をもとに演算します。異なるインスタンスの FB の場合は初期値をもとに演算します。

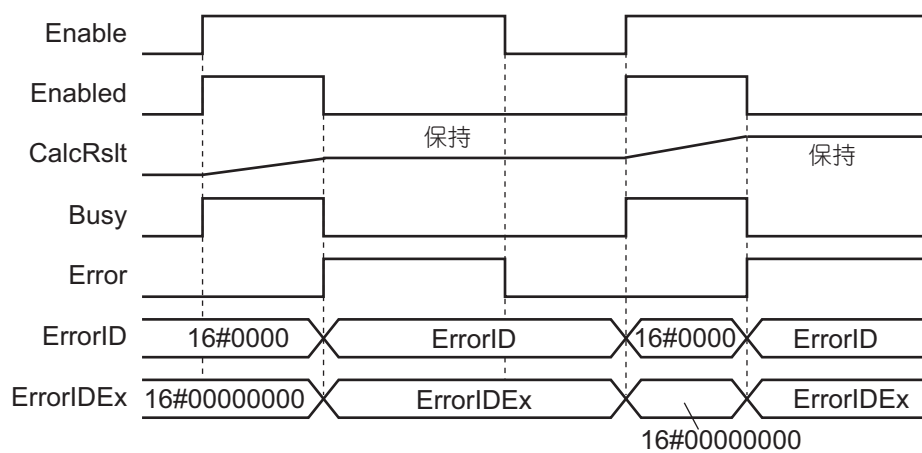
タイミングチャート

正常に完了した場合、異常が発生した場合のタイミングチャートを示します。

● 正常に完了した場合



● 異常が発生した場合



3-2 注意事項

ここでは、本 FB を使用する場合の注意について説明します。

3-2-1 ネスティング

本 FB を呼び出す場合のネスティング階層は、4 階層までです。
ネスティングの詳細は、『NJ/NX シリーズ CPU ユニット ユーザーズマニュアル ソフトウェア編 (SBCA-467)』または『NY シリーズ IPC マシンコントローラ 産業用パネル PC/産業用ボックス PC ユーザーズマニュアル ソフトウェア編 (SBCA-436)』を参照してください。

3-2-2 動作オプション

本 FB では、動作オプションの(@) (入力立ち上がり微分) は使用できません。

3-2-3 FB の再起動

Execute 型の FB で、同一インスタンスによる再起動はできません。
再起動すると、出力値が初期値になります。
再起動の詳細は、『NJ/NX シリーズ CPU ユニット ユーザーズマニュアル モーション制御編 (SBCE-433)』または『NY シリーズ IPC マシンコントローラ 産業用パネル PC/産業用ボックス PC ユーザーズマニュアル モーション制御編 (SBCE-379)』を参照してください。

FB/FUN の個別仕様(NX_HAD)

NX_HAD.slr に含まれる FB/FUN の個別仕様について説明します。

DeviceVariableToArray_***	4 - 2
ScaleTrans_HAD	4 - 11
LimitAlarm_HAD	4 - 14

DeviceVariableToArray_***

NX シリーズ 高速アナログ入力ユニットから取り込んだ、1 タスク周期分のアナログ入力値をひとつの配列変数に結合します。

FB/FUN 名末尾の「_***」は、本 FUN の入力サンプリング数の最大値を示し、020 と 100 の二つがあります。必要とするサンプリング数に適した FUN を選択してください。

FB/FUN 名	名称	FB/FUN	グラフィック表現	ST 表現
DeviceVariableToArray_100	デバイス出力データ結合	FUN		DeviceVariableToArray_100(Input01, Input02, Input03, Input04, Input05, Input06, Input07, Input08, Input09, Input10, StartPos, InputSize, NextPos, dataArray);
DeviceVariableToArray_020				DeviceVariableToArray_020(Input01, Input02, StartPos, InputSize, NextPos, dataArray);

ライブラリ情報

項目	内容
ライブラリファイル名	OmronLib_NX_HAD_Vx_x.slr (x はバージョンを示す)
名前空間	OmronLib\NX_HAD
FB/FUN 番号	DeviceVariableToArray_100 : 00192 DeviceVariableToArray_020 : 00191
ソースコード	非公開

入力変数

	名称	データ型	内容	有効範囲	単位	初期値
EN	実行	BOOL	TRUE : 実行する FALSE : 実行しない	TRUE, FALSE	—	FALSE

	名称	データ型	内容	有効範囲	単位	初期値
Input01[] - Input10[]	デバイス出力 データ 01 ～10	ARRAY[0..9] OF INT	I/O データから取得する アナログ入力値を入力	—	—	—
StartPos	開始位置	UINT	デバイス出力データを 「DataArray[]」のどの要素 番号から結合するのかを 指定。 詳細は「機能 (P.4 - 3)」を参照	データ型に 従う	—	0
InputSize	データ数	UINT	I/O データから取得する 「サンプリング数」を入力	データ型に 従う	—	1

出力変数

	名称	データ型	内容	有効範囲	単位	初期値
ENO	結合結果	BOOL	TRUE : 正常終了 FALSE : 異常終了、または 実行条件非成立	TRUE, FALSE	—	—
NextPos	次位置	UINT	次の結合開始要素番号を 出力。 詳細は「機能 (P.4 - 3)」を参照	データ型に 従う	—	—

入出力変数

	名称	データ型	内容	有効範囲	単位	初期値
DataArray[] ^{*1}	結合データ	ARRAY[*] OF REAL	結合された入力データを 格納	データ型に 従う	—	—

*1. 配列の要素数は任意です。ただし、配列要素数は「InputSize」以上でなければなりません。また、配列要素の開始番号は 0 でも 0 以外でもかまいません。

機能

NX シリーズ 高速アナログ入力ユニットから取得するアナログ入力値は、ひとつまたは複数の ARRAY[0..9] OF INT 型配列であるため、これらをひとつの REAL 型配列に結合します。

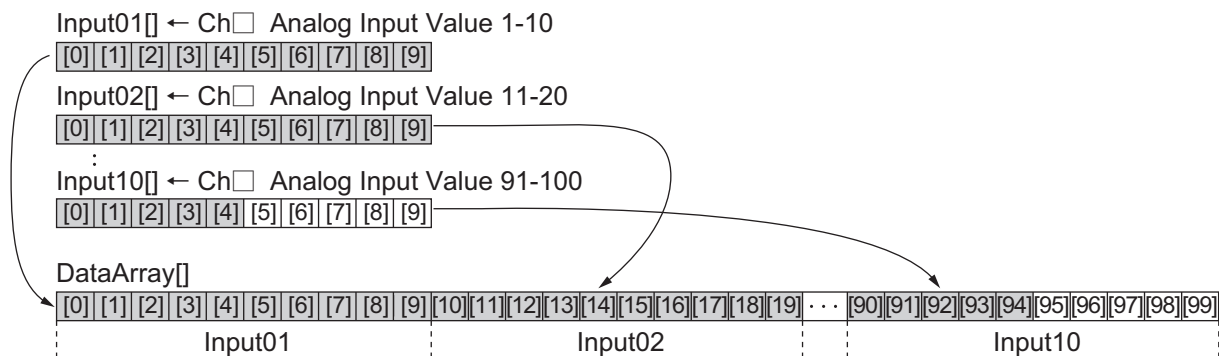
「Input01[0]」から順に、「InputSize」で指定されたデータ数分結合します。

結合成功時は「ENO」に TRUE を出力します。

以下の場合、配列の結合は行われず「ENO」に FALSE を出力します。

- ・「InputSize」が 0 である場合
- ・「DataArray[]」の配列要素数が「InputSize」未満である場合
- ・「StartPos」が「DataArray[]」の範囲外である場合

例：「DataArray[]」の配列要素数が 100、「InputSize」が 95 の場合、「Input10[4]」までを結合します。

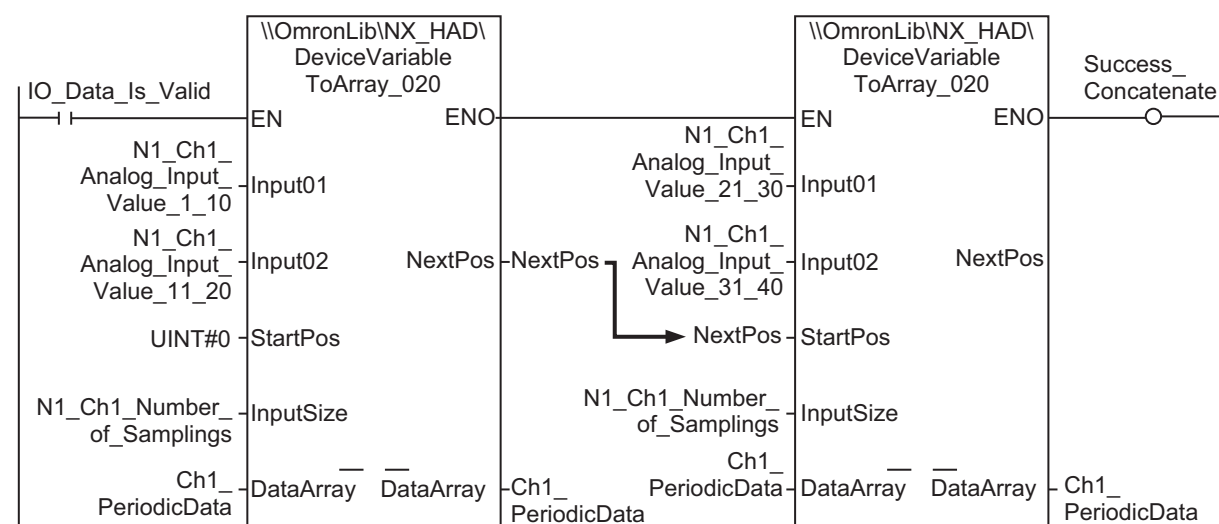


・ 直列接続

以下のように本 FUN を直列接続すると、ひとつの FUN の入力サンプリング数の最大値を超えるデータを結合することができます。

また、DeviceVariableToArray_020 と DeviceVariableToArray_100 を混在して接続することもできます。

例：サンプリング数が 40 個の場合



使用上の注意

直列接続する場合、以下のとおりとしてください。

- すべての「InputSize」には、I/O データから結合する入力 Ch のサンプリング数を入力してください。例の場合は「N1_Ch1_Number_of_Sampling」を入力しています。
- 直列接続する各 FUN の「DataArray[]」には同じ変数を指定してください。
- 「DataArray[]」の配列要素数は、「InputSize」以上の値を指定してください。
- 1 番目の FUN の「StartPos」には 0 を入力してください。
- 2 番目以降の FUN の「StartPos」には、ひとつ前の FUN の「NextPos」の値を入力してください。

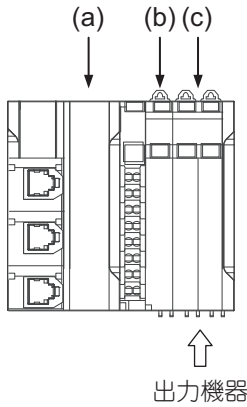
サンプルプログラム 1

NX シリーズ 高速アナログ入力ユニットが 1 タスク周期あたり 100 回サンプリングしたアナログ入力値をひとつの配列変数に結合します。さらに、結合したデータをスケール変換し、そのデータの上下限を監視するプログラムです。

NX シリーズ 高速アナログ入力ユニットを、NX シリーズ CPU ユニットに接続したときの例です。

システム構成

システム構成を以下に示します。



記号	説明	形式	内容
(a)	NX シリーズ CPU ユニット	形 NX102-□□□□	—
(b)	I/O 電源追加供給ユニット	形 NX-PF0730	• NX ユニット番号：1
(c)	NX シリーズ 高速アナログ入力ユニット	形 NX-HAD401	• NX ユニット番号：2 • 使用するチャンネル：Ch1

ユニット動作設定

NX シリーズ 高速アナログ入力ユニットを下表のように設定します。設定方法は、『NX シリーズ アナログ I/O ユニット ユーザーズマニュアル 高速アナログ入力ユニット編 (SBCA-461)』を参照してください。

設定項目	設定値	設定の意味
Ch1 有効／無効	TRUE	Ch1 有効
Ch1 レンジ設定	0	-10～+10V
Ch1 サンプリング数設定	100	100 回

プログラム

• 外部変数

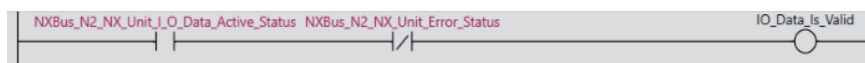
名称	データ型	コ ン ス タ ン ト	コ メ ン ト
NXBus_N2_NX_Unit_I_O_Data_Active_Stat us	BOOL		
NXBus_N2_NX_Unit_Error_Status	BOOL		
N2_Ch1_Number_of_Samplings	UINT		
N2_Ch1_Analog_Input_Value_1_10	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_11_20	ARRAY[0..9] OF INT		

名称	データ型	コン ス タ ン ト	コメント
N2_Ch1_Analog_Input_Value_21_30	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_31_40	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_41_50	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_51_60	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_61_70	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_71_80	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_81_90	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_91_100	ARRAY[0..9] OF INT		
Ch1_Input_Array100	ARRAY[0..99] OF REAL		1 タスク周期分のアナ ログ入力値

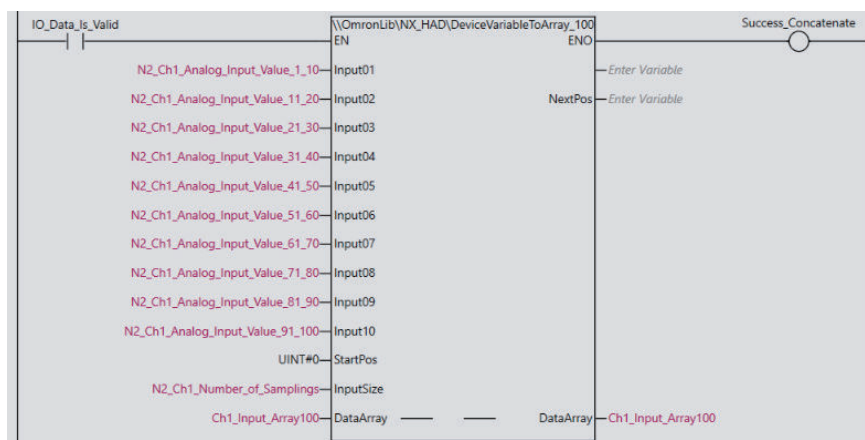
- 内部変数

名称	データ型	初期値	割 付 先	保 持	コン ス タ ン ト	コメント
LimitAlarm_instance	OmronLib\NX_HAD \LimitAlarm_HAD					
IO_Data_Is_Valid	BOOL					NX シリーズ 高速 アナログ入力ユニ ットからの入力値 が正常なときに TRUE になる
Success_Concatenate	BOOL					
Alarm	BOOL					「QHH」、「QH」、 「QL」、「QLL」のい ずれかが TRUE の ときに TRUE にな る
QHH	BOOL					
QH	BOOL					
QL	BOOL					
QLL	BOOL					
PeakVal	REAL					
BottomBal	REAL					
Clear_PkBtm	BOOL	FALSE				
EN_P1	BOOL	FALSE				本変数の値が TRUE の間、スケー ル変換、警報判定を 実行する

- タスク設定
プライマリ定周期タスクに配置してください。
- LD プログラム
(1) NX シリーズ 高速アナログ入力ユニットからの正常なデータ受信を確認します。

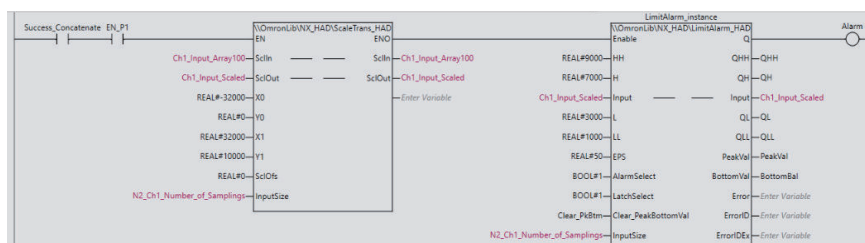


(2) 1 タスク周期分のアナログ入力値をひとつの変数「Ch1_Input_Array100[]」に結合します。



(3) 1 タスク周期分のアナログ入力値をスケール変換後、入力値が「HH」、「H」、「L」、「LL」で設定した範囲外にあるときは「Alarm」を TRUE にします。

スケール変換設定	-32,000~+32,000 → 0~10,000
警報設定	
上限「HH」警報	9,000 を超えたとき
上限「H」警報	7,000 を超えたとき
下限「L」警報	3,000 未満のとき
下限「LL」警報	1,000 未満のとき



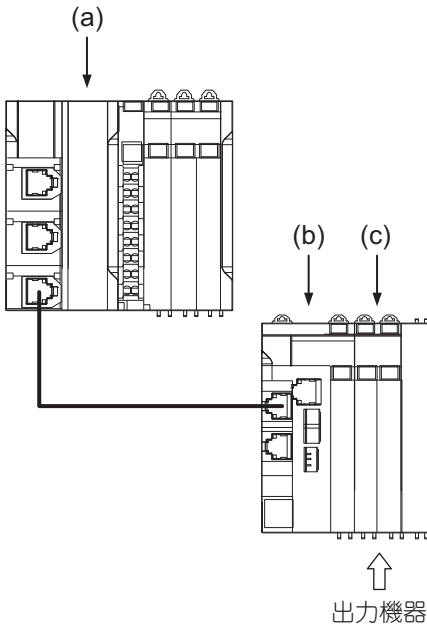
サンプルプログラム 2

NX シリーズ 高速アナログ入力ユニットが 1 タスク周期あたり 100 回サンプリングしたアナログ入力値をひとつの配列変数に結合します。さらに、結合したデータをスケール変換し、そのデータの上下限を監視するプログラムです。

NX シリーズ 高速アナログ入力ユニットを、EtherCAT スレーブターミナルに接続したときの例です。

システム構成

システム構成を以下に示します。



記号	説明	形式	内容
(a)	NX シリーズ CPU ユニット	形 NX102-□□□□	—
(b)	EtherCAT カプラユニット	形 NX-ECC203	• ノードアドレス : 1
(c)	NX シリーズ 高速アナログ入力ユニット	形 NX-HAD401	• NX ユニット番号 : 1 • 使用するチャンネル : Ch1

ユニット動作設定

NX シリーズ 高速アナログ入力ユニットを下表のように設定します。設定方法は、『NX シリーズ アナログ I/O ユニット ユーザーズマニュアル 高速アナログ入力ユニット編 (SBCA-461)』を参照してください。

設定項目	設定値	設定の意味
Ch1 有効／無効	TRUE	Ch1 有効
Ch1 レンジ設定	0	-10～+10V
Ch1 サンプリング数設定	100	100 回

プログラム

- 外部変数

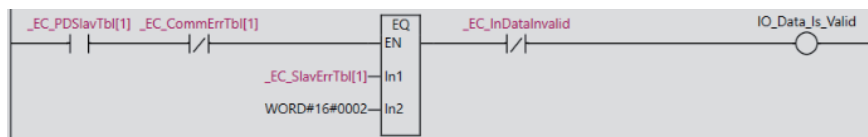
名称	データ型	コ ン ス タ ン ト	コ メ ン ト
_EC_PD_SlavTbl	ARRAY[1..192] OF BOOL	レ	
_EC_CommErrTbl	ARRAY[1..192] OF BOOL	レ	
_EC_SlavErrTbl	ARRAY[1..192] OF WORD	レ	
_EC_InDataInvalid	BOOL	レ	
N1_Ch1_Number_of_Samplings	UINT		
N1_Ch1_Analog_Input_Value_1_10	ARRAY[0..9] OF INT		

名称	データ型	コン ス タ ン ト	コメント
N1_Ch1_Analog_Input_Value_11_20	ARRAY[0..9] OF INT		
N1_Ch1_Analog_Input_Value_21_30	ARRAY[0..9] OF INT		
N1_Ch1_Analog_Input_Value_31_40	ARRAY[0..9] OF INT		
N1_Ch1_Analog_Input_Value_41_50	ARRAY[0..9] OF INT		
N1_Ch1_Analog_Input_Value_51_60	ARRAY[0..9] OF INT		
N1_Ch1_Analog_Input_Value_61_70	ARRAY[0..9] OF INT		
N1_Ch1_Analog_Input_Value_71_80	ARRAY[0..9] OF INT		
N1_Ch1_Analog_Input_Value_81_90	ARRAY[0..9] OF INT		
N1_Ch1_Analog_Input_Value_91_100	ARRAY[0..9] OF INT		
Ch1_Input_Array100	ARRAY[0..99] OF REAL		1 タスク周期分のアナ ログ入力値

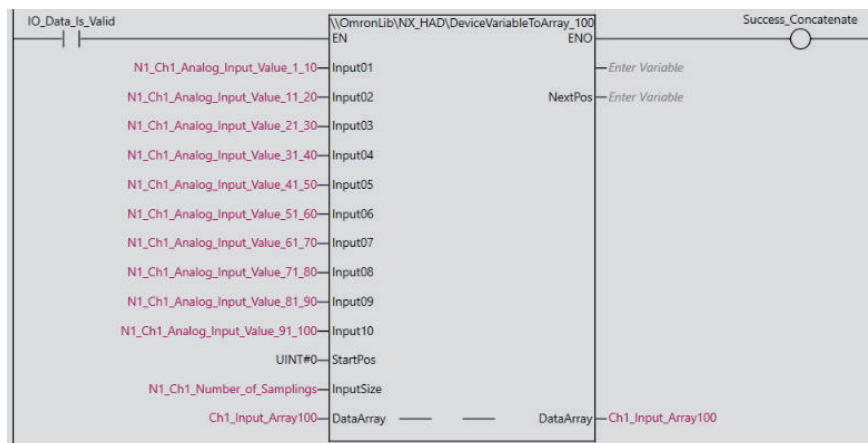
- 内部変数

名称	データ型	初期値	割 付 先	保 持	コン ス タ ン ト	コメント
LimitAlarm_instance	OmronLib\NX_HAD \LimitAlarm_HAD					
IO_Data_Is_Valid	BOOL					NX シリーズ 高速 アナログ入力ユニ ットからの入力値 が正常なときに TRUE になる
Success_Concateenate Alarm	BOOL					
QHH	BOOL					「QHH」、 「QH」、 「QL」、 「QLL」 のい ずれかが TRUE の ときに TRUE にな る
QH	BOOL					
QL	BOOL					
QLL	BOOL					
PeakVal	REAL					
BottomBal	REAL					
Clear_PkBtm	BOOL	FALSE				
EN_P1	BOOL	FALSE				本変数の値が TRUE の間、スケー ル変換、警報判定を 実行する

- タスク設定
プライマリ定周期タスクに配置してください。
- LD プログラム
(1) NX シリーズ 高速アナログ入力ユニットからの正常なデータ受信を確認します。

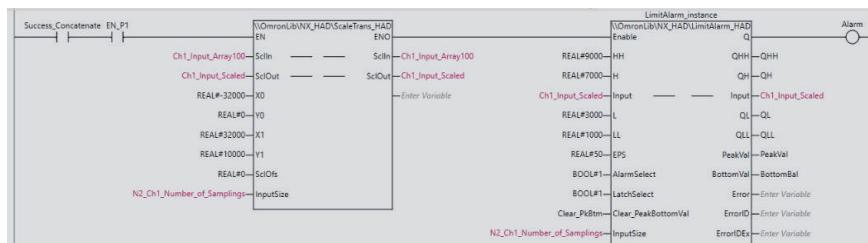


(2) 1 タスク周期分のアナログ入力値をひとつの変数「Ch1_Input_Array100[]」に結合します。



(3) 1 タスク周期分のアナログ入力値をスケール変換後、入力値が「HH」、「H」、「L」、「LL」で設定した範囲外にあるときは「Alarm」を TRUE にします。

スケール変換設定	-32,000～+32,000 → 0～10,000
警報設定	上上限「HH」警報 9,000 を超えたとき
	上限「H」警報 7,000 を超えたとき
	下限「L」警報 3,000 未満のとき
	下下限「LL」警報 1,000 未満のとき



使用上の注意

- サンプルプログラムは、本ライブラリの機能を使用する部分の記述例です。実際の装置のプログラムを作成する場合は、装置の動作に関する安全回路やインタロックプログラム、ほかの機器からの入出力、制御手順等に関するプログラムを作成してください。
- 実際の装置が意図した動作となるように、ユーザプログラムを作成してください。
- 作成したユーザプログラムは、十分な動作確認を行ったのち、本運転に移行してください。

ScaleTrans_HAD

NX シリーズ 高速アナログ入力ユニットから取り込んだデータのスケール変換を行います。

FB/FUN 名	名称	FB/FUN	グラフィック表現	ST 表現
ScaleTrans_HAD	NX シリーズ 高速アナログ入力ユニット用スケール変換	FUN	\\OmronLib\NX_HAD\ScaleTrans_HAD EN ScIIn ScIOut X0 Y0 X1 Y1 ScIOfs InputSize ENO ScIIn ScIOut	ScaleTrans_HAD(ScIIn, ScIOut, X0, Y0, X1, Y1, ScaleOfs, InputSize);

ライブラリ情報

項目	内容
ライブラリファイル名	OmronLib_NX_HAD_Vx_x.slr (x はバージョンを示す)
名前空間	OmronLib\NX_HAD
FB/FUN 番号	00193
ソースコード	非公開

入力変数

	名称	データ型	内容	有効範囲	単位	初期値
EN	実行	BOOL	TRUE : 実行する FALSE : 実行しない	TRUE, FALSE	—	FALSE
X0	入力レンジ 下限値	REAL	入力レンジの下限値	データ型に従う	—	-32000
Y0	出力レンジ 下限値	REAL	出力レンジの下限値	データ型に従う	—	-32000
X1	入力レンジ 上限値	REAL	入力レンジの上限値	データ型に従う	—	32000
Y1	出力レンジ 上限値	REAL	出力レンジの上限値	データ型に従う	—	32000
ScIOfs	オフセット 値	REAL	出力値に加えるオフセット値	データ型に従う	—	0.0
InputSize	データ数	UINT	I/O データから取得する「サンプリング数」を入力	1~1000	—	1

出力変数

	名称	データ型	内容	有効範囲	単位	初期値
ENO	変換結果	BOOL	TRUE : 正常終了 FALSE : 異常終了、または実行条件非成立	TRUE, FALSE	—	—

入出力変数

	名称	データ型	内容	有効範囲	単位	初期値
ScIIn[*] ^{*1}	入力値	ARRAY[*] OF REAL	変換の対象となるデータ配列	データ型に従う	—	—
ScLOut[*] ^{*2}	出力値	ARRAY[*] OF REAL	スケール変換後のデータ配列	データ型に従う	—	—

- *1. 配列の要素数は任意です。ただし、配列要素数は「InputSize」以上でなければなりません。また、配列要素の開始番号は 0 でも 0 以外でもかまいません。
- *2. 配列の要素数は任意です。ただし、配列要素数は「ScIIn[]」と同じでなければなりません。また、配列要素の開始番号は 0 でも 0 以外でもかまいません。また、配列要素の開始番号が「ScIIn[]」と違ってても実行可能です。

機能

「ScIIn[]」の各要素を、「X0」と「X1」で指定された入力レンジから、「Y0」から「Y1」で指定した出力レンジに変換します。

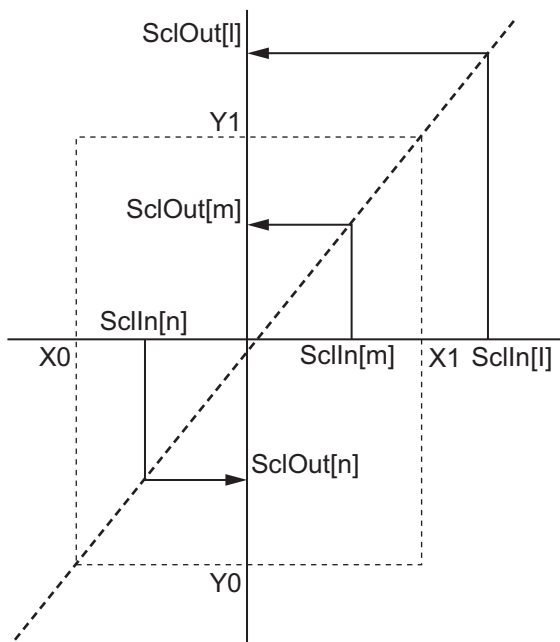
出力レンジに変換した値に「ScLOfs」を加算した値を、「ScLOut[]」に格納します。

入力値が入力レンジ外であっても実行可能です。

変換成功時は「ENO」に TRUE を出力します。

以下の場合、スケール変換は行われず、「ENO」に FALSE を出力します。

- ・「ScIIn[]」、「ScLOut[]」の配列要素数が「InputSize」未満である場合
- ・「ScIIn[]」と「ScLOut[]」の配列要素数が不一致の場合
ただし開始番号の違いは可
- ・「X0」と「X1」が等しい、または「Y0」と「Y1」が等しい場合
- ・「InputSize」が 0 または 1001 以上の場合



$$\text{ScLOut}[n] = \frac{Y1 - Y0}{X1 - X0} \times (\text{ScIIn}[n] - X0) + Y0 + \text{ScLOfs}$$

nの範囲は、「ScIIn[]」の先頭要素番号から先頭要素番号に「InputSize」-1を加えたものまでです。

サンプルプログラム

「サンプルプログラム 1 (P.4 - 4)」、「サンプルプログラム 2 (P.4 - 7)」を参照してください。

LimitAlarm_HAD

NX シリーズ 高速アナログ入力ユニットから取り込んだデータに対して上上限、上限、下限、下下限の警報を行います。

FB/FUN 名	名称	FB/FUN	グラフィック表現	ST 表現
LimitAlarm_HAD	NX シリーズ 高速アナログ入力ユニット用 上下限警報	FB	LimitAlarm_HAD_instance \\OmronLib\NX_HAD\LimitAlarm_HAD Enable Q HH QHH H QH Input Input L QL LL QLL EPS PeakVal AlarmSelect BottomVal LatchSelect Error Clear_PeakBottomVal ErrorID InputSize ErrorIDEx	LimitAlarm_HAD_instance(Enable, HH, H, Input, L, LL, EPS, AlarmSelect, LatchSelect, Clear_PeakBottomVal, InputSize, Q, QHH, QH, QL, QLL, PeakVal, BottomVal, Error, ErrorID, ErrorIDEx);

ライブラリ情報

項目	内容
ライブラリファイル名	OmronLib_NX_HAD_Vx_x.slr (x はバージョンを示す)
名前空間	OmronLib\NX_HAD
FB/FUN 番号	00194
ソースコード	非公開

入力変数

	名称	データ型	内容	有効範囲	単位	初期値
Enable	実行	BOOL	TRUE : 実行する FALSE : 実行しない	TRUE, FALSE	—	FALSE
HH	上上限設定値	REAL	入力値の上上限設定値	データ型に従う	—	0.0
H	上限設定値	REAL	入力値の上限設定値	データ型に従う	—	0.0
L	下限設定値	REAL	入力値の下限設定値	データ型に従う	—	0.0
LL	下下限設定値	REAL	入力値の下下限設定値	データ型に従う	—	0.0

	名称	データ型	内容	有効範囲	単位	初期値
EPS	ヒステリシス	REAL	警報のヒステリシス	0 以上	—	0.0
AlarmSelect	警報選択	BOOL	TRUE : HH、H、L、LL 警報 FALSE : H、L 警報	TRUE, FALSE	—	FALSE
LatchSelect	ラッチ選択	BOOL	TRUE : 警報をラッチする FALSE : 警報をラッチしない	TRUE, FALSE	—	FALSE
Clear_PeakBottomVal	ピーク値、ボトム値クリア	BOOL	TRUE : ピーク値、ボトム値をクリアする	TRUE, FALSE	—	FALSE
InputSize	データ数	UINT	I/O データから取得する「サンプリング数」を入力	1~1000	—	1

出力変数

	名称	データ型	内容	有効範囲	単位	初期値
Q	警報出力	BOOL	TRUE : 「QHH」、「QH」、「QL」、「QLL」のいずれかが TRUE FALSE : 「QHH」、「QH」、「QL」、「QLL」がすべて FALSE	TRUE, FALSE	—	—
QHH	上上限警報	BOOL	TRUE : 上上限警報 ON FALSE : 上上限警報 OFF	TRUE, FALSE	—	—
QH	上限警報	BOOL	TRUE : 上限警報 ON FALSE : 上限警報 OFF	TRUE, FALSE	—	—
QL	下限警報	BOOL	TRUE : 下限警報 ON FALSE : 下限警報 OFF	TRUE, FALSE	—	—
QLL	下下限警報	BOOL	TRUE : 下下限警報 ON FALSE : 下下限警報 OFF	TRUE, FALSE	—	—
PeakVal	ピーク値	REAL	最大入力値	データ型に従う	—	—
BottomVal	ボトム値	REAL	最小入力値	データ型に従う	—	—
Error	エラー	BOOL	TRUE : 異常終了 FALSE : 正常終了、実行中、または実行条件非成立	TRUE, FALSE	—	—
ErrorID	エラーコード	WORD	異常終了時の異常 ID 正常終了時は 16#0	*1	—	—
ErrorIDEx	拡張エラーコード	DWORD	異常終了時の拡張異常 ID 正常終了時は 16#0	*1	—	—

*1. 詳細は「トラブルシューティング (P.4 - 18)」を参照してください。

入出力変数

	名称	データ型	内容	有効範囲	単位	初期値
Input[]*1	入力値	ARRAY[*] OF REAL	監視対象となるデータ配列	データ型に従う	—	—

*1. 配列の要素数は任意です。ただし、配列要素数は「InputSize」以上でなければなりません。また、配列要素の開始番号は 0 でも 0 以外でもかまいません。

機能

「Enable」が TRUE の間、「Input[]」の各要素の値が、「HH」、「H」設定値を超えたとき、または「L」、「LL」設定値未満となったときに警報を出力します。

入力、各設定値と警報出力の関係は以下のとおりです。

- 「Input[]」が「HH」より大きいとき、「QHH」を TRUE にします。
「QHH」が TRUE のときに入力値が「HH」-「EPS」より小さくなると「QHH」を FALSE にします。
- 「Input[]」が「H」より大きいとき、「QH」を TRUE にします。
「QH」が TRUE のときに入力値が「H」-「EPS」より小さくなると「QH」を FALSE にします。
- 「Input[]」が「L」より小さいとき、「QL」を TRUE にします。
「QL」が TRUE のときに入力値が「L」+「EPS」より大きくなると「QL」を FALSE にします。
- 「Input[]」が「LL」より小さいとき、「QLL」を TRUE にします。
「QLL」が TRUE のときに入力値が「LL」+「EPS」より大きくなると「QLL」を FALSE にします。
- 「QHH」、「QH」、「QL」、「QLL」のいずれかが TRUE のとき、「Q」を TRUE にします。

「AlarmSelect」が TRUE の場合、「HH」、「H」、「L」、「LL」の四つを警報判定します。

「AlarmSelect」が FALSE の場合、「H」、「L」の二つを警報判定します。その場合、「QHH」、「QLL」は常に FALSE です。

「InputSize」で指定した要素数の入力値に対して警報判定を行います。

そのため、「QHH」、「QH」、「QL」、「QLL」が複数同時に TRUE になることがあります。

詳細は下記を参照してください。

以下の場合、「Error」に TRUE を出力し、監視を終了します。

- 「Input[]」の配列要素数が「InputSize」未満である場合
- 各警報設定値が下記の関係になっていない場合

$$「LL」 \leq 「L」 \leq 「H」 \leq 「HH」$$
- 「EPS」が非数または $\pm\infty$ の場合
- 「InputSize」が 0 または 1001 以上の場合

実行中、実行開始から現在までの最大入力値を「PeakVal」に、最小入力値を「BottomVal」に出力します。

「Clear_PeakBottomVal」が TRUE のとき、「PeakVal」、「BottomVal」に 0 を出力します。

「Clear_PeakBottomVal」が継続して TRUE のとき、「PeakVal」、「BottomVal」は 0 を出力し続けます。

「Clear_PeakBottomVal」を FALSE にすると、それ以降の入力値で最大値、最小値を出力します。

「Enable」が FALSE になると、本 FUN の実行を終了します。

- 「Q」、「QHH」、「QH」、「QL」、「QLL」はすべて FALSE になります。
- 「PeakVal」、「BottomVal」は、「Enable」が再度 TRUE になるまで維持します。

- 警報の出力／解除について

警報の出力／解除の動作は「LatchSelect」によって決まります。

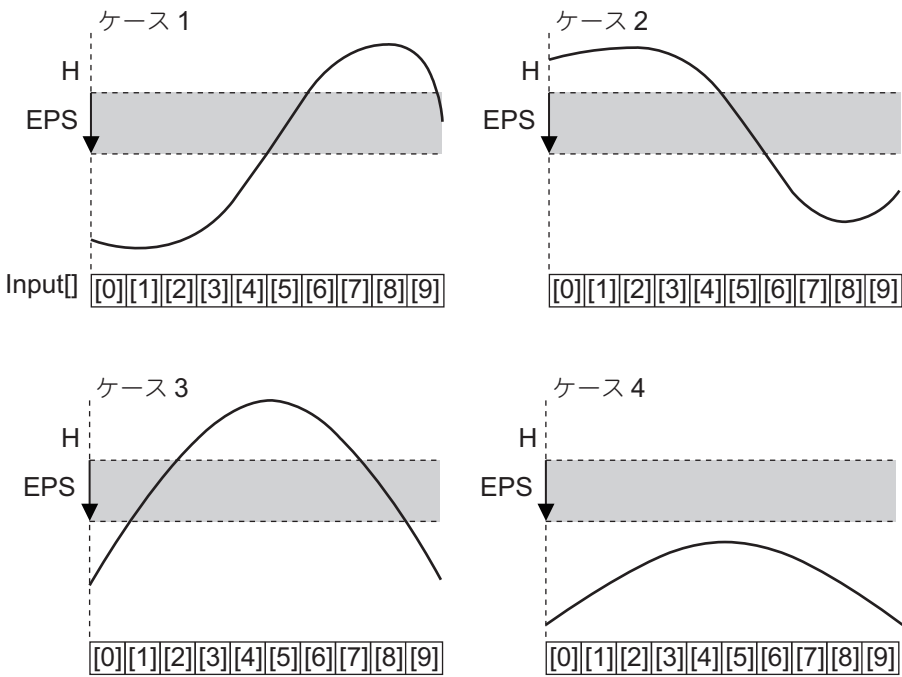
a) 「LatchSelect」が FALSE の場合

「Input[]」のうち、「InputSize」で指定した要素数の最終要素に対して警報判定します。

b) 「LatchSelect」が TRUE の場合

「Input[]」のうち、「InputSize」で指定したすべての要素に対して警報判定します。

例として、NX シリーズ 高速アナログ入力ユニットの 1 タスク周期分の入力変化が「Input[]」に格納されたときの「QH」の動作を示します。

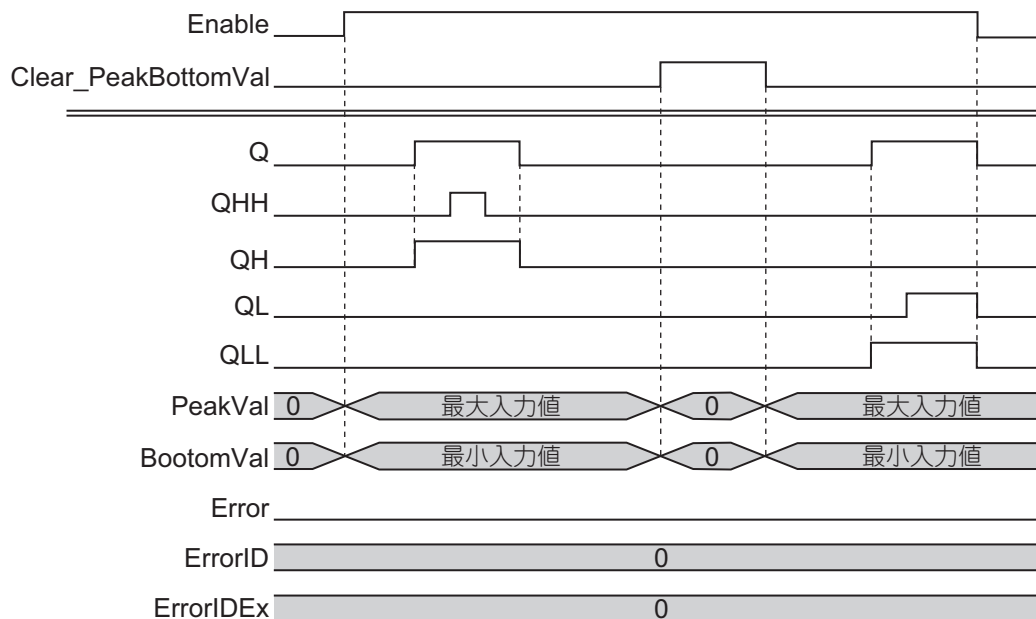


LatchSelect	QH			
	ケース 1	ケース 2	ケース 3	ケース 4
FALSE	TRUE	FALSE	FALSE	FALSE
TRUE	TRUE	TRUE	TRUE	FALSE

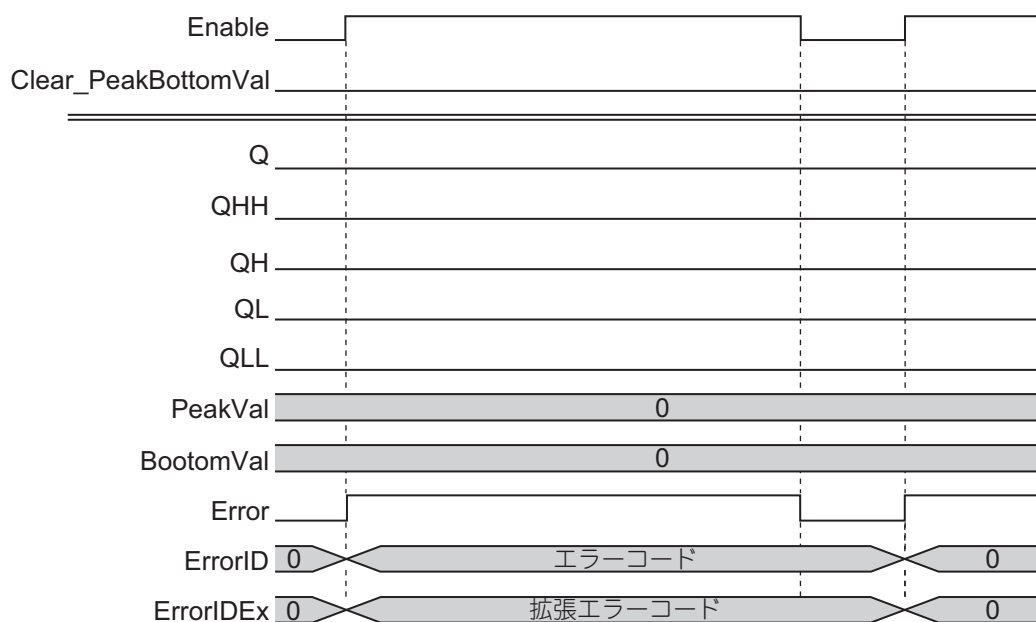
- 実行中の設定変更について
「HH」、「H」、「L」、「LL」、「EPS」は、実行中は常時反映します。
「AlarmSelect」、「LatchSelect」は実行中に変更しても反映しません。
実行中に変更された場合、警報出力は FALSE となり、変更後の設定値で警報判定します。

タイミングチャート

- タイミングチャートを、以下に示します。
- 「Enable」を TRUE にすると警報監視を開始します。
実行開始から現在までの最大入力値を「PeakVal」に、最小入力値を「BottomVal」に出力します。
 - 「Enable」を FALSE にすると警報監視を終了します。
 - 「Clear_PeakBottomVal」が TRUE である間は、「PeakVal」、「BottomVal」に 0 を出力します。
 - 異常が発生した場合、「Error」を TRUE にします。同時に、「ErrorID」、「ErrorIDEx」を出力します。
 - 異常発生時、「Enable」が TRUE である間、「Error」は保持されます。
「ErrorID」、「ErrorIDEx」は、「Enable」を FALSE にした後、再度「Enable」を TRUE にするまで保持します。
 - 正常終了時のタイミングチャート



- 異常終了時のタイミングチャート



トラブルシューティング

エラーコード	拡張エラーコード	状態	内容	対処法
16#0000	16#00000000	正常終了	—	—
16#3D00	16#00000001	データ数不正	「Input[]」配列の要素数が「InputSize」未満	「Input[]」配列の要素数を見直してください。
	16#00000002	警報設定大小関係不正	「HH」、「H」、「L」、「LL」の大小関係が間違っている	「LL」 ≤ 「L」 ≤ 「H」 ≤ 「HH」となるように設定してください。
	16#00000003	EPS 不正	「EPS」の値が有効範囲外	「EPS」の値を有効範囲内に修正してください。
	16#00000004	InputSize 不正	「InputSize」の値が有効範囲外	「InputSize」の値を有効範囲内に修正してください。

サンプルプログラム

「サンプルプログラム 1 (P.4 - 4)」、「サンプルプログラム 2 (P.4 - 7)」を参照してください。

FB/FUN の個別仕様 (DataRecorder)

DataRecorder.slr に含まれる FB/FUN の個別仕様について説明します。

TrigControl.....	5 - 2
DataRecorder.....	5 - 12
LimitTest.....	5 - 18
CalcFeatureValues.....	5 - 23
LogDataToCSV.....	5 - 31
CSVToLogData.....	5 - 37

TrigControl

DataRecorder FB がログデータの記録を開始するトリガ情報を生成します。

FB/FUN 名	名称	FB/ FUN	グラフィック表現	ST 表現
TrigCon trol	トリガ制 御	FB	TrigControl_instance \\OmronLib\DataRecorder\TrigControl Enable Enabled TrigSource TrigSource TrigInput TrigOutput TrigConfig TrigMark TrigInputTimeStamp Error InputDataTimeStamp ErrorID InputSize ErrorIDEx TrigControl_insta nce(Enable, TrigSource, TrigInput, TrigConfig, TrigInputTimeSta mp, InputDataTimeSta mp, InputSize, Enabled, TrigOutput, TrigMark, Error, ErrorID, ErrorIDEx);	

ライブラリ情報

項目	内容
ライブラリファイル名	OmronLib_DataRecorder_Vx_x.slr (x はバージョンを示す)
名前空間	OmronLib\DataRecorder
FB/FUN 番号	00195
ソースコード	非公開

入力変数

	名称	データ型	内容	有効範囲	単位	初期値
Enable	実行	BOOL	TRUE : 実行する FALSE : 実行しない	TRUE, FALSE	—	—
TrigInput	トリガ入力	BOOL	トリガを検出するデジタル信号を入力する。 トリガモードが「デジタル入力トリガモード」または「タイムスタンプトリガモード」の場合に有効	TRUE, FALSE	—	—
TrigConfig	トリガ設定	OmronLib \DataRecorder \sTrigConfig	トリガ条件を設定	—	—	—

	名称	データ型	内容	有効範囲	単位	初期値
TrigInputTimestamp	トリガ入力タイムスタンプ	ULINT	I/O データから取得する「トリガ入力タイムスタンプ」を入力する。 トリガモードが「タイムスタンプトリガモード」の場合に有効	データ型に従う	nsec	0
InputDataTimestamp	アナログ入力タイムスタンプ	ULINT	I/O データから取得する「アナログ入力タイムスタンプ」を入力	データ型に従う	nsec	0
InputSize	データ数	UINT	I/O データから取得する「サンプリング数」を入力	1～1000	—	1

出力変数

	名称	データ型	内容	有効範囲	単位	初期値
Enabled	実行中	BOOL	実行中に TRUE になる	TRUE, FALSE	—	—
TrigOutput	トリガ出力	BOOL	トリガ条件が成立したとき、TRUE を出力	TRUE, FALSE	—	—
TrigMark	トリガ情報	OmronLib DataRecorder TrigMark	トリガ条件が成立したときの「TrigSource[]」の要素番号と配列要素の値を出力	—	—	—
Error	エラー	BOOL	TRUE：異常終了 FALSE：正常終了、実行中、または実行条件非成立	TRUE, FALSE	—	—
ErrorID	エラーコード	WORD	異常終了時の異常 ID 正常終了時は 16#0	*1	—	—
ErrorIDEx	拡張エラーコード	DWORD	異常終了時の拡張異常 ID 正常終了時は 16#0	*1	—	—

*1. 詳細は「トラブルシューティング (P.5 - 7)」を参照してください。

入出力変数

	名称	データ型	内容	有効範囲	単位	初期値
TrigSource[]*1	トリガ判定用入力値	ARRAY[*] OF REAL	トリガ条件の判定対象となるデータ配列を入力	—	—	—

*1. 配列の要素数は任意です。ただし、配列要素数は「InputSize」以上でなければなりません。また、配列要素の開始番号は 0 でも 0 以外でもかまいません。

構造体

- OmronLib\DataRecorder\sTrigConfig

メンバの名称	データ型	内容
TrigMode	USINT	トリガモードを設定します。詳細は「機能 (P.5 - 4)」を参照してください。 - アナログレベルトリガモード 「TrigSource[]」の値と変化方向でトリガを判定します。 - デジタル入力トリガモード トリガ入力の立ち上がりでトリガを検出します。 - タイムスタンプトリガモード トリガ入力の立ち上がりでトリガを検出します。 トリガ条件が成立した時刻をデジタル入力トリガモードよりも正確に取得できます。 トリガ入力には、NX シリーズ 高速アナログ入力ユニットのトリガ入力機能や、変化時刻入力方式の NX シリーズ デジタル入力ユニットを使用します。
TrigSlope	BOOL	トリガを検出する「TrigSource[]」の変化方向を指定します。 「アナログレベルトリガモード」の場合に有効です。 FALSE : 立ち上がり TRUE : 立ち下がり
AnalogTrigLevel	REAL	トリガを検出する入力しきい値を指定します。 「アナログレベルトリガモード」の場合に有効です。

- OmronLib\DataRecorder\sTrigMark

メンバの名称	データ型	内容
Pos	UINT	トリガ条件が成立したときの、「TrigSource[]」の要素番号を出力します。
Data	REAL	トリガ条件が成立したときの、「TrigSource[]」に格納されているデータを出力します。

機能

「Enable」が TRUE の間、「TrigConfig」で指定したトリガ条件を満たしているかを判定します。
トリガ条件が成立したとき、「TrigOutput」を TRUE にし、「TrigMark」にトリガ条件が成立した「TrigSource[]」の要素番号とその配列要素に格納されているデータを出力します。

一度トリガ条件が成立すると、「TrigOutput」、「TrigMark」は「Enable」が TRUE の間、保持します。

再起動する場合は、「Enable」を 1 タスク周期以上 FALSE にしてください。

「Enable」が FALSE になったとき、トリガ判定を終了し、「Enabled」、「TrigOutput」を FALSE にします。

また、「TrigMark.Pos」、「TrigMark.Data」には 0 を出力します。

以下の場合、「Error」を TRUE にし、トリガ判定を終了します。

- 「TrigSource[]」の要素数が「InputSize」未満の場合
- 「TrigConfig.TrigMode」が 0、1、2 以外の場合
- 「InputSize」が 0、または 1001 以上の場合
- 本 FB をイベントタスクで実行した場合

トリガモードの設定と動作は以下のとおりです。

- アナログレベルトリガモード：「TrigConfig.TrigMode」が 0
「TrigConfig.TrigMode」が 0 のとき、トリガ条件に「TrigConfig.TrigSlope」と「TrigConfig.AnalogTrigLevel」を使用します。
「InputSize」で指定した「TrigSource[]」の各要素が、「TrigConfig. AnalogTrigLevel」をまたいだときにトリガ条件の成立と判定します。このとき、「TrigOutput」を TRUE にし、トリガ条件が成立した「TrigSource[]」の要素番号とその配列要素の値を「TrigMark」に出力します。

例：「TrigConfig.TrigMode」が 0、「AnalogInputLevel」が 250、「TrigConfig.Slope」が FALSE（立ち上がり）、「TrigSource[0..9]」、「InputSize」が 10 の場合

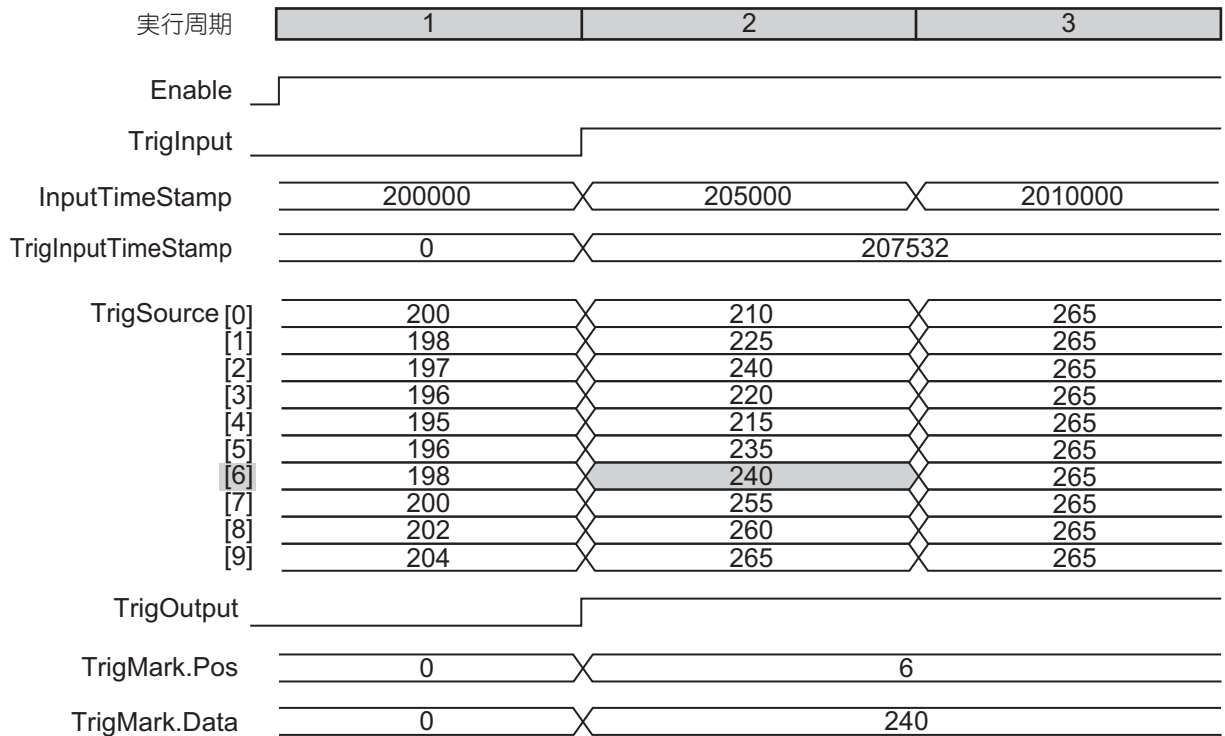
実行周期	1	2	3
Enable			
TrigSource [0]	200	210	265
[1]	198	225	265
[2]	197	240	265
[3]	196	220	265
[4]	195	215	265
[5]	196	235	265
[6]	198	240	265
[7]	200	255	265
[8]	202	260	265
[9]	204	265	265
TrigOutput			
TrigMark.Pos	0	7	
TrigMark.Data	0	255	

- デジタル入力トリガモード：「TrigConfig.TrigMode」が 1
「TrigInput」が TRUE になったとき、「TrigOutput」を TRUE にし、「TrigSource[]」の先頭要素番号とその配列要素の値を「TrigMark」に出力します。
判定条件には「TrigConfig.TrigSlope」と「TrigConfig.AnalogTrigLevel」を使用しません。

例：「TrigConfig.TrigMode」が 1、「TrigSource[0..9]」、「InputSize」が 10 のとき

実行周期	1	2	3
Enable			
TrigInput			
TrigSource [0]	200	210	265
[1]	198	225	265
[2]	197	240	265
[3]	196	220	265
[4]	195	215	265
[5]	196	235	265
[6]	198	240	265
[7]	200	255	265
[8]	202	260	265
[9]	204	265	265
TrigOutput			
TrigMark.Pos	0	0	
TrigMark.Data	0	210	

- タイムスタンプトリガモード：「TrigConfig.TrigMode」が 2
「TrigInput」が TRUE になったとき、トリガ出力を TRUE にします。このとき、「InputDataTimeStamp」、「TrigInputTimeStamp」から、トリガ条件が成立した時刻直後の「TrigSource[]」の配列要素番号を特定し、その要素番号と配列要素の値を「TrigMark」に出力します。
判定条件には「TrigConfig.TrigSlope」と「TrigConfig.AnalogTrigLevel」を使用しません。

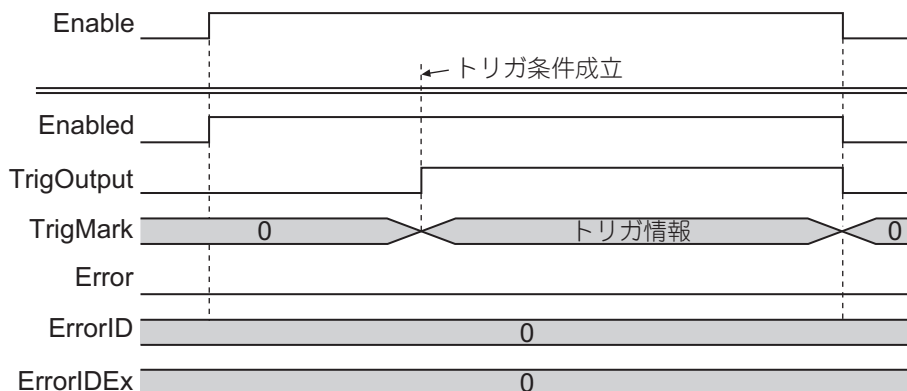


- 実行中の設定変更について
「TrigConfig」は、実行中に変更されると、変更された設定によってトリガ判定を行います。
ただし、トリガ条件成立（「TrigOutput」が TRUE になった）後の変更は反映されません。

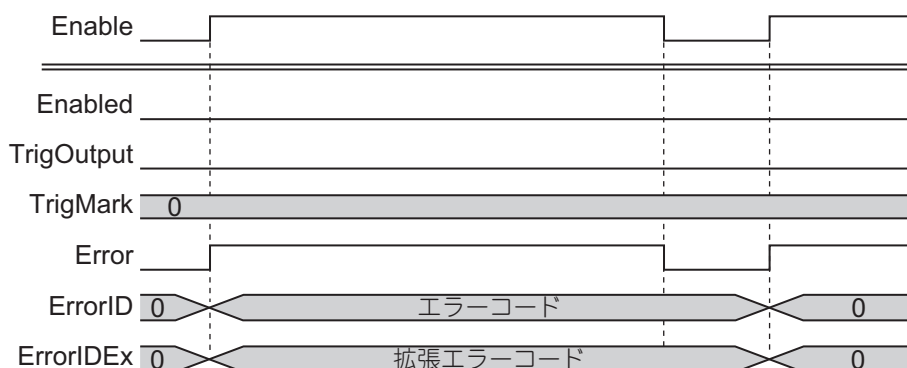
タイミングチャート

タイミングチャートを、以下に示します。

- 「Enable」が TRUE になるとトリガ判定を開始します。
- 「Enable」が FALSE になるとトリガ判定を終了します。
- トリガ条件が成立すると、「TrigOutput」を TRUE にし、「TrigMark」にトリガ条件が成立した「TrigSource[]」の要素番号とその配列要素の値をトリガ情報として出力します。
- 異常が発生した場合、「Error」を TRUE にします。同時に、「ErrorID」、「ErrorIDEx」を出力します。
- 異常発生時、「Enable」が TRUE である間、「Error」は保持します。
「ErrorID」、「ErrorIDEx」は「Enable」を FALSE にした後、再度「Enable」を TRUE にするまで保持します。
- 正常終了時のタイミングチャート



- ・ 異常終了時のタイミングチャート



トラブルシューティング

エラーコード	拡張エラーコード	状態	内容	対処法
16#0000	16#00000000	正常終了	—	—
16#3D01	16#00000001	要素数不正	「TrigSource[]」配列の要素数が「InputSize」未満	「TrigSource[]」配列の要素数を見直してください。
	16#00000002	トリガモード不正	トリガモード「TrigConfig.TrigMode」が有効範囲外	トリガモードを見直してください。
	16#00000003	InputSize 不正	「InputSize」の値が有効範囲外	「InputSize」の値を有効範囲内に修正してください。
	16#00000004	イベントタスク実行	本 FB がイベントタスクで実行された	本 FB は定周期タスクで実行してください。

サンプルプログラム

NX シリーズ 高速アナログ入力ユニットが 1 タスク周期あたり 100 回サンプリングしたアナログ入力値を DeviceVariableToArray_100 でひとつの配列変数に結合します。さらに、結合したデータを DataRecorder で記録し、記録したデータと検査基準データを LimitTest で比較し検査するプログラムです。

NX シリーズ 高速アナログ入力ユニットを、NX シリーズ CPU ユニットに接続したときの例です。本サンプルプログラムは、プライマリ定周期タスクで実行するプログラム 0 と定周期タスク 17 で動作するプログラム 1 の二つの POU で構成されます。

プログラム 0 (プライマリ定周期タスク)

- 外部変数

名称	データ型	コン ス タ ン ト	コメント
NXBus_N2_NX_Unit_I_O_Data_Active_Stat us	BOOL		
NXBus_N2_NX_Unit_Error_Status	BOOL		
N2_Ch1_Analog_Input_Value_1_10	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_11_20	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_21_30	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_31_40	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_41_50	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_51_60	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_61_70	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_71_80	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_81_90	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_91_100	ARRAY[0..9] OF INT		
N2_Ch1_Number_of_Samplings	UINT		
N2_Analog_Input_Time_Stamp	ULINT		
Ch1_Input_Array100	ARRAY[0..99] OF REAL		1 タスク周期分のアナ ログ入力値
LogData	ARRAY[0..4999] OF OmronLib\DataRecorder \sLogData		記録先データ配列
RecordEnd	BOOL		
MasterReadEnd	BOOL		

- 内部変数

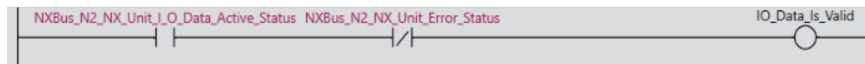
名称	データ型	初 期 値	割 付 先	保 持	コン ス タ ン ト	コメント
TrigControl_instance	OmronLib\DataRecorder \TrigControl					
DataRecorder_instanc e	OmronLib\DataRecorder \DataRecorder					
IO_Data_Is_Valid	BOOL					
TrigConfig	OmronLib\DataRecorder \sTrigConfig					
TrigMark	OmronLib\DataRecorder \sTrigMark					
TrigOut	BOOL					
DataBuf	ARRAY[0..4999] OF OmronLib\DataRecorder \sLogData					

名称	データ型	初期値	割付先	保持	コンスタント	コメント
RecordInfo	OmronLib\DataRecorder \sRecordInfo					
LogEnable	BOOL					
Success_Concatenate	BOOL					

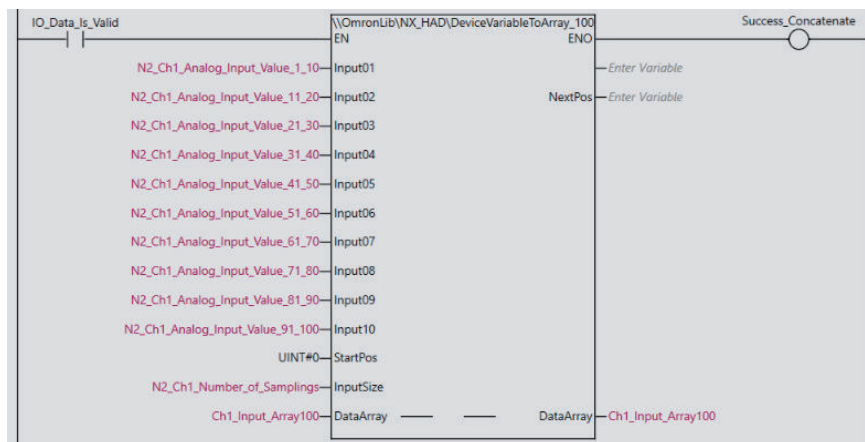
- タスク設定
プライマリ定周期タスクに配置してください。

- LD プログラム

- (1) NX シリーズ 高速アナログ入力ユニットからの正常なデータ受信を確認します。NX シリーズ 高速アナログ入力ユニットを EtherCAT スレーブターミナルに接続する場合は「サンプルプログラム 2 (P.4 - 7)」を参照してください。

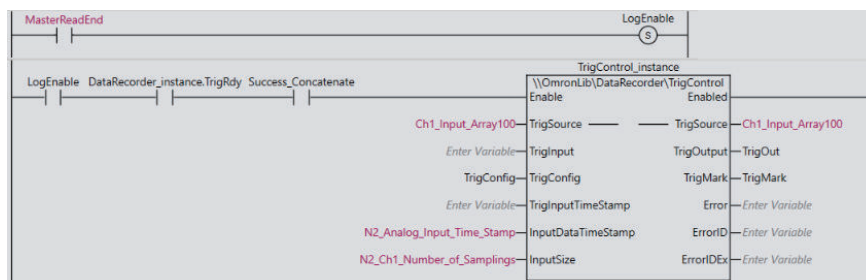


- (2) 1 タスク周期分のアナログ入力値をひとつの変数「Ch1_Input_Array100[]」に結合します。

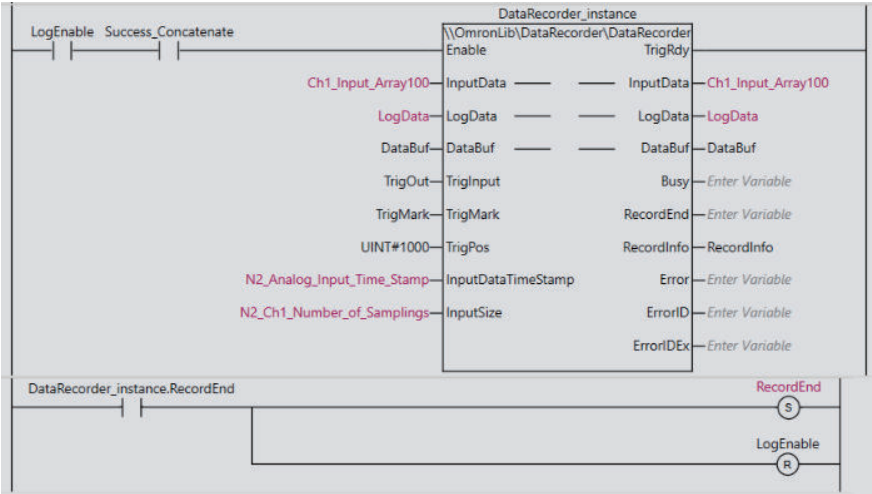


- (3) 後述の「プログラム 1 (定周期タスク 17) (P.5 - 10)」で実行する、マスタデータ CSV ファイル読出しが完了すると、NX シリーズ 高速アナログ入力ユニットの入力値記録のトリガ待ち状態になります。トリガ条件は「TrigConfig」で設定します。

- トリガ条件
「TrigConfig.TrigMode」=0 (アナログレベルトリガモード)
「TrigMode.TrigSlope」=FALSE (立ち上がり)
「TrigMode.AnalogTrigLevel」=REAL#100



(4) トリガ条件が成立すると、トリガ条件成立前の入力値 1,000 サンプル分と、トリガ条件成立後の 4,000 サンプル分のデータを「LogData[]」に記録します。



プログラム 1 (定周期タスク 17)

- 外部変数

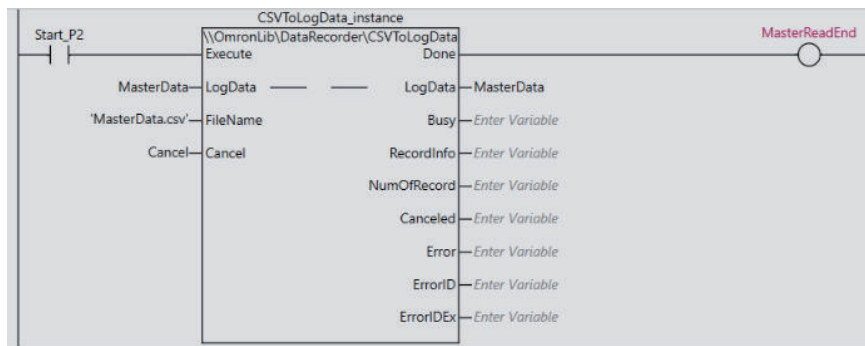
名称	データ型	コン ス タ ン ト	コメント
RecordEnd	BOOL		
MasterReadEnd	BOOL		
LogData	ARRAY[0..4999] OF OmronLib\DataRecorder \sLogData		

- 内部変数

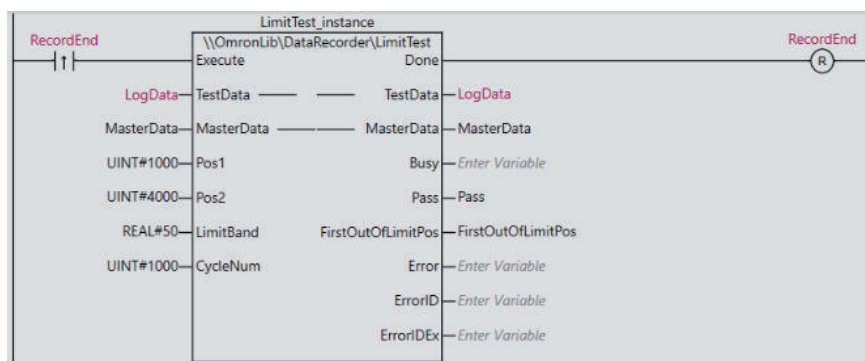
名称	データ型	初 期 値	割 付 先	保 持	コン ス タ ン ト	コメント
CSVToLogData_instance	OmronLib\DataRecorder \CSVToLogData					
LimitTest_instance	OmronLib\DataRecorder \LimitTest					
MasterData	ARRAY[0..4999] OF OmronLib\DataRecorder \sLogData					
FirstOutOfLimitPos	UINT					
Start_P2	BOOL	FA LS E				本変数の値を TRUE にすると、プログラムを実行します。
Pass	BOOL					

名称	データ型	初期値	割付先	保持	コンスタント	コメント
Cancel	BOOL					

- ・ タスク設定
定周期タスク 17 に配置してください。
- ・ LD プログラム
(1) 「Start_P2」を TRUE にすると、SD メモリカード内の「MasterData.csv」を読み出し、検査基準データとします。
検査基準データはあらかじめ準備しておく必要があります。



- (2) 「プログラム 0 (プライマリ定周期タスク) (P.5 - 8)」で記録した「LogData[0..4999]」と csv ファイルから読み出した検査基準データ「MasterData[0..4999]」を検査します。
- ・ 検査範囲：要素番号 1,000 から 4,000 まで
 - ・ 合格基準：同一要素番号の検査基準データに対して -50～+50
- 検査は 1 定周期タスクあたり、1,000 要素ずつ行います。検査完了時、合格であれば「Pass」を TRUE にします。



使用上の注意

- ・ サンプルプログラムは、本ライブラリの機能を使用する部分の記述例です。実際の装置のプログラムを作成する場合は、装置の動作に関する安全回路やインタロックプログラム、ほかの機器からの入出力、制御手順等に関するプログラムを作成してください。
- ・ 実際の装置が意図した動作となるように、ユーザプログラムを作成してください。
- ・ 作成したユーザプログラムは、十分な動作確認を行ったのち、本運転に移行してください。

DataRecorder

指定した要素数のデータ配列を毎タスク周期、ひとつの配列変数に連結し、時系列ログデータを作成します。

FB/FUN 名	名称	FB/FUN	グラフィック表現	ST 表現
DataRecorder	データレコーダ	FB	DataRecorder_instance	<pre>DataRecorder_instance(Enable, InputData, LogData, DataBuf, TrigInput, TrigMark, TrigPos, InputDataTimeStamp, InputSize, TrigRdy, Busy, RecordEnd, RecordInfo, Error, ErrorID, ErrorIDEx);</pre>

ライブラリ情報

項目	内容
ライブラリファイル名	OmronLib_DataRecorder_Vx_x.slr (x はバージョンを示す)
名前空間	OmronLib\DataRecorder
FB/FUN 番号	00196
ソースコード	非公開

入力変数

	名称	データ型	内容	有効範囲	単位	初期値
Enable	実行	BOOL	TRUE : 実行する FALSE : 実行しない	TRUE, FALSE	—	FALSE
TrigInput	トリガ入力	BOOL	記録開始トリガ。 BOOL 型変数、I/O データや、TrigControl FB の「トリガ出力」を入力	TRUE, FALSE	—	FALSE
TrigMark	トリガ情報	OmronLib\DataRecorder\sTrigMark	TrigControl FB が出力する「トリガ情報」を入力	—	—	Pos=0 Data=0

	名称	データ型	内容	有効範囲	単位	初期値
TrigPos	トリガ位置	UINT	データ記録時、トリガ入力時のデータを、「LogData[]」上のどの要素番号に記録するかを指定	データ型に従う	—	0
InputDataTimeStamp	アナログ入力タイムスタンプ	ULINT	I/O データから取得する「アナログ入力タイムスタンプ」を入力	データ型に従う	nsec	0
InputSize	データ数	UINT	I/O データから取得する「サンプリング数」を入力	1～1000	—	1

出力変数

	名称	データ型	内容	有効範囲	単位	初期値
TrigRdy	トリガ入力可	BOOL	トリガ入力を受付可能ときに TRUE になる	TRUE, FALSE	—	—
Busy	記録中	BOOL	トリガ入力が入ったときから記録を完了までの間、TRUE になる	TRUE, FALSE	—	—
RecordEnd	記録完了	BOOL	記録時、TRUE になる	TRUE, FALSE	—	—
RecordInfo	記録情報	OmronLib \DataRecorder \sRecordInfo	記録開始時刻、トリガ入力時刻などの情報を出力	—	—	—
Error	エラー	BOOL	TRUE：異常終了 FALSE：正常終了、実行中、または実行条件非成立	TRUE, FALSE	—	—
ErrorID	エラーコード	WORD	異常終了時の異常 ID 正常終了時は 16#0	*1	—	—
ErrorIDEx	拡張エラーコード	DWORD	異常終了時の拡張異常 ID 正常終了時は 16#0	*1	—	—

*1. 詳細は「トラブルシューティング (P.5 - 17)」を参照してください。

入出力変数

	名称	データ型	内容	有効範囲	単位	初期値
InputData[*] ^{*1}	入力データ	ARRAY[*] OF REAL	記録元のデータ配列を入力	—	—	—
LogData[*] ^{*2}	ログデータ	ARRAY[*] OF OmronLib \DataRecorder \sLogData	記録先のデータ配列を入力。 配列要素数が、記録可能なレコード数になる	—	—	—
DataBuf[*] ^{*2}	入力バッファ	ARRAY[*] OF OmronLib \DataRecorder \sLogData	記録するデータを一時的に格納するバッファです。 変数を割り当てる必要がありますが、本 FB 実行中の中間データですので、書き込み、参照はしないでください。	—	—	—

*1. 配列の要素数は任意です。ただし、配列要素数は「InputSize」以上でなければなりません。また、配列要素の開始番号は 0 でも 0 以外でもかまいません。

*2. 配列の要素数は任意です。ただし、配列要素数は「TrigPos」以上でなければなりません。また、配列要素の開始番号は 0 でも 0 以外でもかまいません。

構造体

- OmronLib\DataRecorder\sTrigMark
sTrigMark の構造は「TrigControl (P.5 - 2)」を参照してください。
- OmronLib\DataRecorder\sRecordInfo

名称	データ型	内容
StartDateTime	DT	先頭データの日付時刻を記録します。
EndDateTime	DT	末尾データの日付時刻を記録します。 記録途中で本 FB の実行を終了した場合は記録されません。
TriggerDateTime	DT	「TrigInput」が入力された日付時刻を記録します。 使用方法により、記録内容が異なります。詳細は「機能 (P.5 - 14)」を参照してください。
RecordNum	UINT	「LogData[]」に記録したデータの要素数を記録します。
TrigPos	UINT	「TrigInput」が TRUE になったときの「LogData[]」の要素番号を記録します。

- OmronLib\DataRecorder\sLogData

名称	データ型	内容
xTime	ULINT	「xData」がサンプリングされたタイムスタンプを記録します。
xData	REAL	入力データを記録します。

機能

「Enable」が TRUE の間、毎タスク周期「InputData[]」から、「InputSize」で指定された要素数のデータを「DataBuf[]」に記録します。

「Enable」が TRUE になってから、「DataBuf[]」に「TrigPos」で指定したデータ数の記録が終わると「TrigRdy」を TRUE にします。

「TrigRdy」が TRUE のときに「TrigInput」が TRUE になると、「TrigRdy」は FALSE、「LogData[]」に「TrigPos」以降のデータを記録します。この記録中、「Busy」を TRUE にします。

記録が完了すると、「TrigPos」までの「DataBuf[]」のデータを「LogData[]」に連結して出力し、「RecordEnd」を TRUE にします。

同時に「RecordInfo」を出力します。

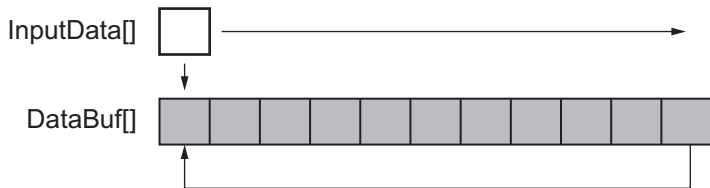
「TrigRdy」が FALSE のときに「TrigInput」を TRUE にしても無視します。

記録完了後に「Enable」を FALSE にすると、本 FB の実行を終了します。

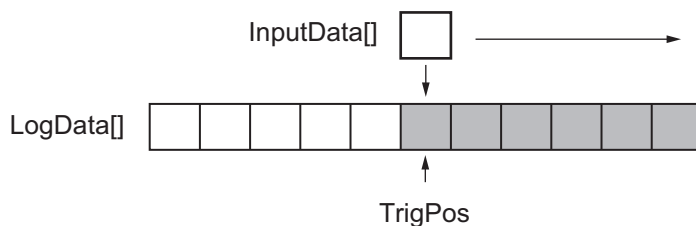
「RecordEnd」、「RecordInfo」は、「Enable」を FALSE にした後は、再度「Enable」を TRUE にするまで保持します。

- データの格納方法

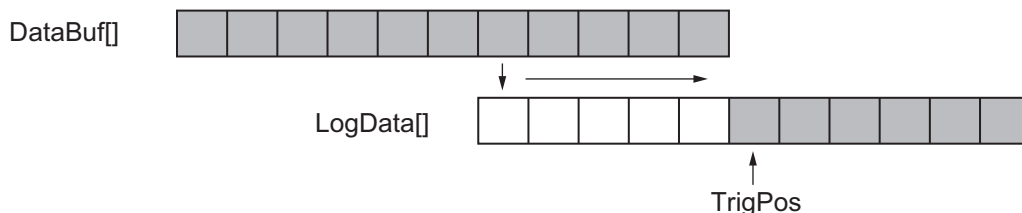
- (1) 「Enable」が TRUE になってから「TrigInput」が TRUE になるまでは、「InputData[]」を「DataBuf[]」に記録し続けます。



- (2) 「TrigInput」が TRUE になると、「TrigPos」から順に「LogData[]」の最終要素まで記録します。



- (3) 最終要素まで記録が終わると、「LogData[]」の先頭から「TrigPos」-1 までの要素に「DataBuf[]」のデータを、新しい順に連結し、「LogData[]」の先頭から末尾要素までのログデータを完成させます。



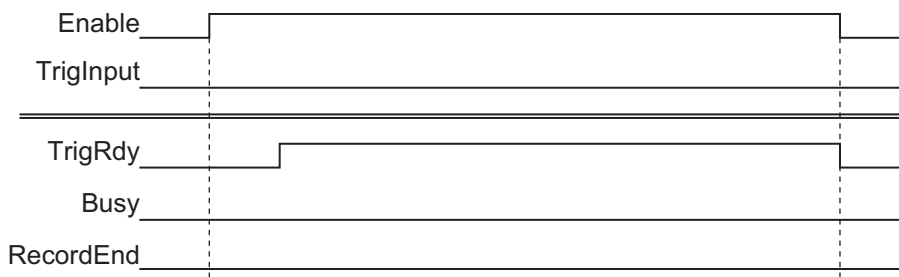
以下の場合、「Error」に TRUE を出力し、記録を終了します。

- a) 「InputData[]」の配列要素数が「InputSize」未満の場合
- b) 「LogData[]」または「DataBuf[]」の配列要素数が「InputData[]」の配列要素数未満の場合
- c) 「TrigMark.Pos」が「InputData[]」の配列範囲外要素番号の場合
- d) 「TrigPos」が「LogData[]」または「DataBuf[]」が範囲外の場合
- e) 「InputSize」が 0、または 1001 以上の場合
- f) 本 FB をイベントタスクで実行した場合

「TrigInput」、「TrigMark」の入力は以下のとおりとしてください。

TrigInput	TrigMark
任意の変数を入力する場合	変数を割り当てない
TrigControl FB の TrigOutput を入力する場合	TrigControl FB の TrigMark

- 実行中の設定変更について
「TrigPos」は、実行中の変更は受け付けません。「Enable」が FALSE の状態で変更してください。
- 本 FB 実行中に「Enable」を FALSE にしたときの動作
a) 「TrigInput」を TRUE にする前に「Enable」を FALSE にしたときの動作：
直ちに「InputData[]」の記録を終了し、「TrigRdy」を FALSE にします。
「LogData[]」、「RecordInfo」は不定です。



- b) 「Busy」が TRUE になった後、「RecordEnd」が TRUE になる前に「Enable」を FALSE にしたときの動作：
「Enable」が FALSE になる 1 タスク周期前の「InputData[]」まで記録されます。「Enable」が FALSE になった以降の「InputData[]」は記録されません。

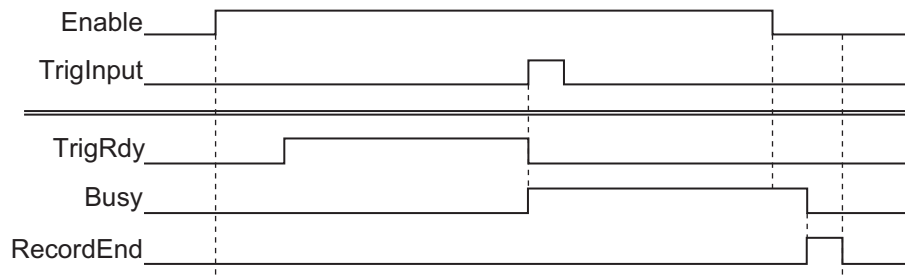
「Busy」は直ちに FALSE とはならず、「LogData[]」の作成完了（「RecordEnd」が TRUE となる）まで TRUE になります。

「RecordEnd」は 1 タスク周期だけ TRUE を出力します。

「RecordInfo.RecordNum」は、「LogData[]」配列先頭から「Enable」が FALSE になる 1 タスク周期前までのデータ数です。

「RecordInfo.RecordNum」が示す配列要素以降の「LogData[]」の配列要素のデータは不定です。

「RecordInfo.EndDateTime」は「Enable」が FALSE になった日付時刻を出力します。



「Busy」が TRUE の間は、本 FB を再起動することはできません。

「RecordEnd」が TRUE になった後に、「Enable」を 1 タスク周期以上 FALSE にしてください。

タイミングチャート

タイミングチャートを、以下に示します。

- 「Enable」が TRUE になると「InputData[]」の「DataBuf[]」への記録を開始します。
- 「Enable」が FALSE になると記録を終了します。
- 「TrigInput」が TRUE になると「LogData[]」への記録を開始します。

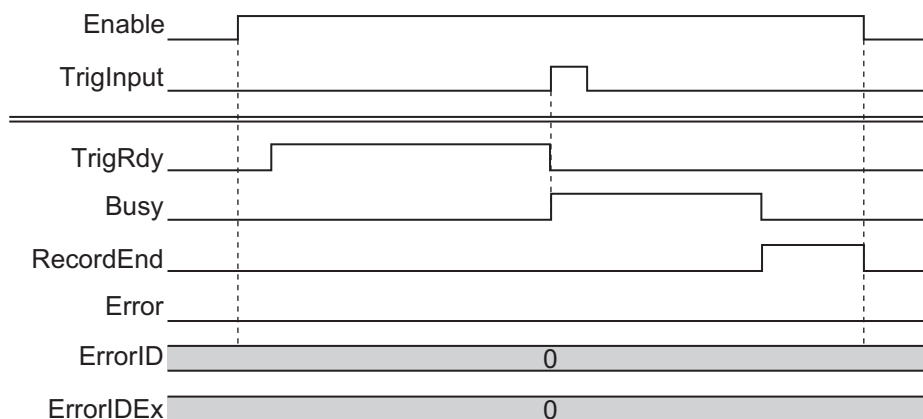
記録中は「Busy」が TRUE になります。記録完了で「RecordEnd」が TRUE になります。

異常が発生した場合、「Error」を TRUE にします。同時に、「ErrorID」、「ErrorIDEx」を出力します。

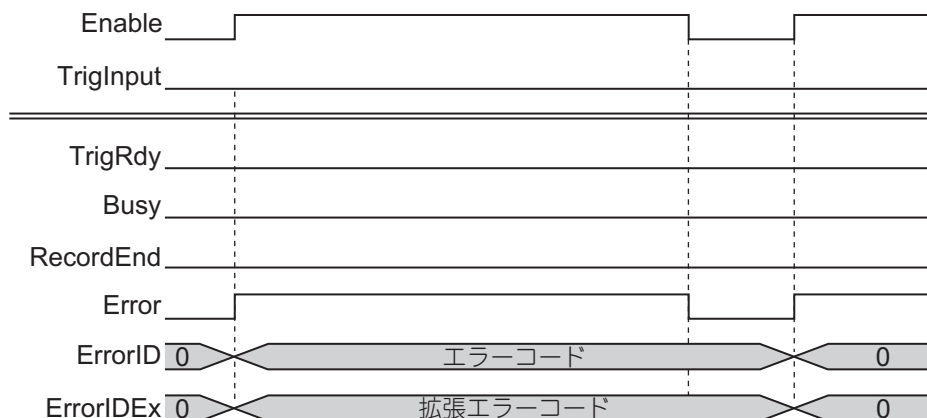
異常発生時、「Enable」が TRUE である間、「Error」は保持します。

「ErrorID」、「ErrorIDEx」は「Enable」を FALSE にした後、再度「Enable」を TRUE にするまで保持します。

- 正常終了時のタイミングチャート



- 異常終了時のタイミングチャート



トラブルシューティング

エラーコード	拡張エラーコード	状態	内容	対処法
16#0000	16#00000000	正常終了	—	—
16#3D02	16#00000001	InputData[]要素数不正	「InputData[]」配列の要素数が InputSize 未満	「InputData[]」配列の要素数を見直してください。
	16#00000002	LogData[], DataBuf[]要素数不正	「LogData[]」配列、または「DataBuf[]」配列の要素数が「InputData[]」配列の要素数未満	「LogData[]」配列、または「DataBuf[]」配列の要素数を見直してください。
	16#00000003	InputSize 不正	「InputSize」の値が有効範囲外	「InputSize」の値を有効範囲内に修正してください。
	16#00000004	TrigMark.Pos 不正	「TrigMark.Pos」が「InputSize」を超過	本 FB の「TrigMark」には、TrigControl FB の「TrigMark」出力を接続してください。
	16#00000005	TrigPos 不正	「TrigPos」が「LogData[]」配列、または「DataBuf[]」配列の範囲外	「TrigPos」の設定値を見直してください。 「LogData[]」配列、または「DataBuf[]」配列の要素数を見直してください。
	16#00000006	イベントタスク実行	本 FB がイベントタスクで実行された	本 FB は定周期タスクで実行してください。

サンプルプログラム

「サンプルプログラム (P.5 - 7)」を参照してください。

LimitTest

テスト対象のデータ配列の各要素値が、検査基準データの許容範囲内に収まっているかどうかを検査します。

FB/FUN 名	名称	FB/FUN	グラフィック表現	ST 表現
LimitTest	上下限テスト	FB	LimitTest_instance \\OmronLib\DataRecorder\LimitTest Execute Done TestData MasterData Pos1 Pos2 LimitBand CycleNum TestData MasterData Busy Pass FirstOutOfLimitPos Error ErrorID ErrorIDEx	LimitTest_instance(Execute, TestData, MasterData, Pos1, Pos2, LimitBand, CycleNum, Done, Busy, Pass, FirstOutOfLimitPos, Error, ErrorID, ErrorIDEx);

ライブラリ情報

項目	内容
ライブラリファイル名	OmronLib_DataRecorder_Vx_x.slr (x はバージョンを示す)
名前空間	OmronLib\DataRecorder
FB/FUN 番号	00197
ソースコード	非公開

入力変数

	名称	データ型	内容	有効範囲	単位	初期値
Execute	起動	BOOL	TRUE : 実行する FALSE : 実行しない	TRUE, FALSE	—	FALSE
Pos1	検査範囲要素 1	UINT	検査対象範囲の開始要素番号を指定	データ型に従う	—	0
Pos2	検査範囲要素 2	UINT	検査対象範囲の最終要素番号を指定	データ型に従う	—	0
LimitBand	許容値	REAL	許容範囲を指定	0 以上	—	0.0
CycleNum	1 周期検査要素数	UINT	1 タスク周期で検査する要素数を指定。 0 を指定した場合、検査対象範囲要素を 1 タスク周期で処理	データ型に従う	—	0

出力変数

	名称	データ型	内容	有効範囲	単位	初期値
Done	完了	BOOL	TRUE: 正常終了 FALSE: 異常終了、実行中、または実行条件非成立	TRUE, FALSE	—	—
Busy	実行中	BOOL	TRUE: 実行中 FALSE: 実行中ではない	TRUE, FALSE	—	—
Pass	検査合格	BOOL	TRUE: 合格 FALSE: 不合格要素がある	TRUE, FALSE	—	—
FirstOutOfLimitPos	先頭不合格要素番号	UINT	最初に見つかった許容範囲外要素の要素番号	データ型に従う	—	—
Error	エラー	BOOL	TRUE: 異常終了 FALSE: 正常終了、実行中、または実行条件非成立	TRUE, FALSE	—	—
ErrorID	エラーコード	WORD	異常終了時の異常 ID 正常終了時は 16#0	*1	—	—
ErrorIDEx	拡張エラーコード	DWORD	異常終了時の拡張異常 ID 正常終了時は 16#0	*1	—	—

*1. 詳細は「トラブルシューティング (P.5 - 22)」を参照してください。

入出力変数

	名称	データ型	内容	有効範囲	単位	初期値
TestData[*] ^{*1}	検査対象データ	ARRAY[*] OF OmronLib \DataRecorder \sLogData	検査対象となるデータ配列を指定	—	—	—
MasterData[*] ^{*2}	検査基準データ	ARRAY[*] OF OmronLib \DataRecorder \sLogData	検査基準となるデータ配列を指定	—	—	—

*1. 配列の要素数は任意です。また、配列要素の開始番号は 0 でも 0 以外でもかまいません。

*2. 配列の要素数は任意です。ただし、「TestData[]」と同じ要素数でなければなりません。また、配列要素の開始番号は 0 でも 0 以外でもかまいません。

構造体

- OmronLib\DataRecorder\sLogData
「TrigControl (P.5 - 2)」を参照してください。

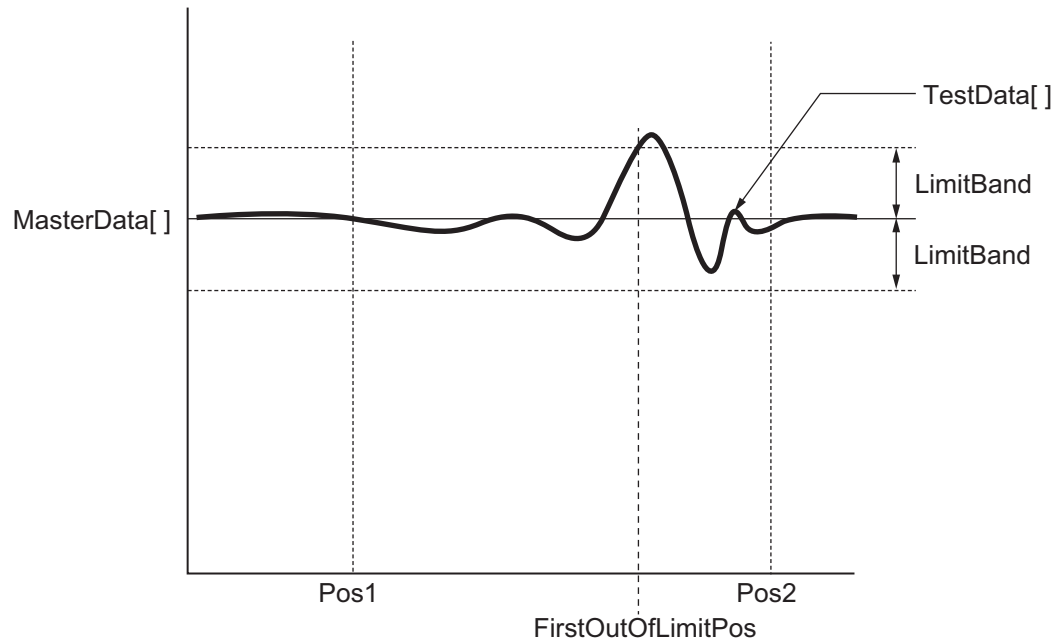
機能

「TestData[]」の各要素が、「MasterData[]」±「LimitBand」の範囲内にあるかどうかを検査します。

「TestData[]」の検査範囲は、「Pos1」と「Pos2」で指定した間の要素です。範囲外の要素は検査しません。

「Execute」の立ち上がりで、「TestData[]」の各要素を「Pos1」から順に検査します。

「Pos1」から「Pos2」までの「TestData[]」の全要素が、「MasterData[]」±「LimitBand」の範囲内であるとき、「Pass」を TRUE にし、「FirstOutOfLimitPos」には 65535 を出力します。検査中に許容範囲外の要素を検出した場合、「Pos2」まで検査を行わないでください。範囲外要素を検出した時点で「Busy」を FALSE、「Done」を TRUE にします。「FirstOutOfLimitPos」には、許容範囲外の値が最初に検出された要素番号を、配列先頭からの相対値で出力します。



合格判定基準は以下のとおりです。

$$\text{MasterData}[x] - \text{LimitBand} \leq \text{TestData}[x] \leq \text{MasterData}[x] + \text{LimitBand}$$

以下の場合、「Error」に TRUE を出力し、検査を終了します。

- 「TestData[]」と「MasterData[]」の配列要素数が不一致の場合
- 「LimitBand」の値が負数または非数、∞の場合
- 「Pos1」、「Pos2」が「TestData[]」、または「MasterData[]」配列範囲外の場合

「Pos1」、「Pos2」の指定方法

「Pos1」、「Pos2」の値は、先頭要素からの相対値で指定してください。

例：TestData [10..100]のとき、「Pos1」が0の場合は「TestData[10]」を指します。

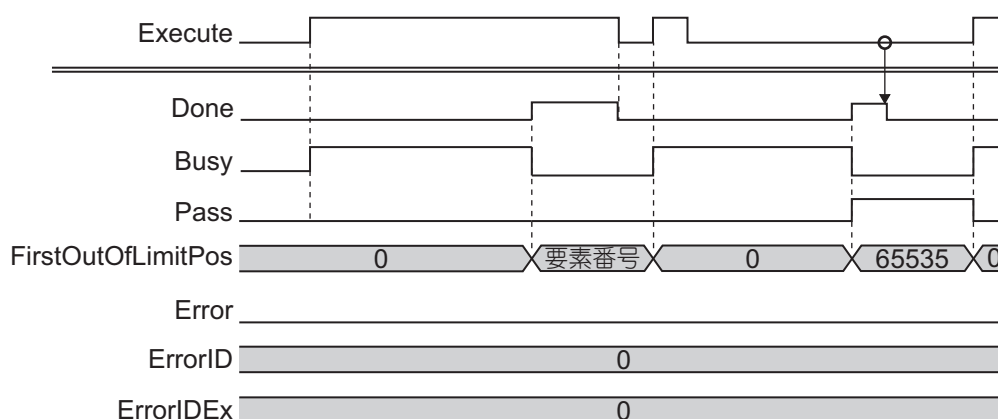
- 実行中の設定変更について
本 FB の多重起動および、実行中の「Pos1」、「Pos2」、「LimitBand」、「CycleNum」の変更は受け付けません。
- 「CycleNum」の指定について
「CycleNum」に 1 タスク周期あたり検査要素数を指定することにより、本 FB の 1 タスク周期の実行時間、および検査完了までの時間を調節することができます。
小さな値を指定した場合、1 タスク周期毎の実行時間が短くなりますが、検査完了までの実行周期が長くなります。
大きな値を指定した場合、検査完了までの時間が短くなりますが、1 タスク周期毎の本 FB 実行時間が長くなります。
「TestData[]」の配列要素数、本 FB を配置するタスクの周期、本 FB 以外に動作するプログラムの負荷状況を考慮して調節してください。

タイミングチャート

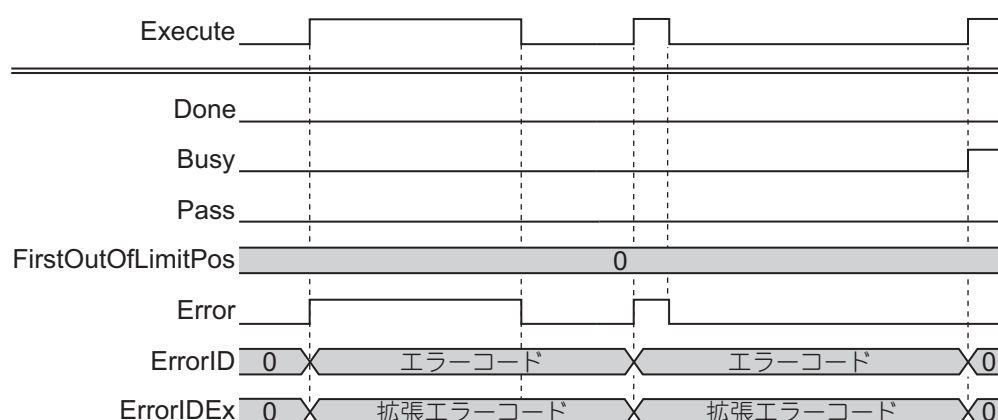
タイミングチャートを、以下に示します。

- 「Execute」の立ち上がりで「Busy」を TRUE にします。
- 全要素の判定が完了すると「Done」を TRUE にし、検査結果を「Pass」、「FirstOutOfLimitPos」に出力します。
「Done」は「Execute」を FALSE にするまで保持します。
「Pass」、「FirstOutOfLimitPos」は「Execute」を FALSE にした後、再度「Execute」を TRUE にするまで保持します。
- 異常が発生した場合、「Error」を TRUE にします。同時に、「ErrorID」、「ErrorIDEx」を出力します。
「Error」は、「Execute」を FALSE にするまで保持します。
「ErrorID」、「ErrorIDEx」は「Execute」を FALSE にした後、再度「Execute」を TRUE にするまで保持します。
- 「Busy」が TRUE の間に「Execute」が FALSE となった場合は、「Done」、「Error」は検査終了後の 1 サイクルのみ TRUE になります。

- 正常終了時のタイミングチャート



- 異常終了時のタイミングチャート



使用上の注意

本 FB は、「Execute」の値が FALSE になっても、また実行時間がタスク周期を超えても、処理を最後まで続けます。

処理が正常終了したかどうかは、「Done」の値が TRUE になることで確認してください。

トラブルシューティング

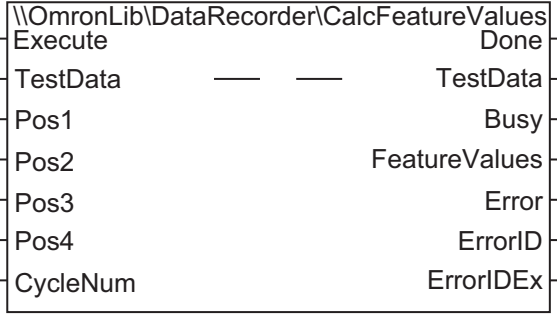
エラーコード	拡張エラーコード	状態	内容	対処法
16#0000	16#00000000	正常終了	—	—
16#3D03	16#00000001	配列要素数不一致	「TestData[]」と「MasterData[]」の配列要素数が不一致	「TestData[]」と「MasterData[]」の配列要素数を見直してください。
	16#00000002	LimitBand 不正	「LimitBand」の値が有効範囲外	「LimitBand」の値を有効範囲内に修正してください。
	16#00000003	検査範囲不正	「Pos1」、「Pos2」が「TestData[]」または「MasterData[]」の範囲外	「Pos1」、「Pos2」の値を見直してください。

サンプルプログラム

「サンプルプログラム (P.5 - 7)」を参照してください。

CalcFeatureValues

テスト対象のデータ配列に対し、平均、標準偏差、歪度、尖度、最大値、最小値を演算します。

FB/FUN 名	名称	FB/FUN	グラフィック表現	ST 表現
CalcFeatureValues	特徴量演算	FB		CalcFeatureValues_instance(Execute, TestData, Pos1, Pos2, Pos3, Pos4, CycleNum, Done, Busy, FeatureValues, Error, ErrorID, ErrorIDEx);

ライブラリ情報

項目	内容
ライブラリファイル名	OmronLib_DataRecorder_Vx_x.slr (x はバージョンを示す)
名前空間	OmronLib\DataRecorder
FB/FUN 番号	00198
ソースコード	非公開

入力変数

	名称	データ型	内容	有効範囲	単位	初期値
Execute	起動	BOOL	TRUE : 実行する FALSE : 実行しない	TRUE, FALSE	—	FALSE
Pos1	演算範囲指定 1	UINT	FeatureValues[0]を演算する範囲の最終要素番号を指定	データ型に従う	—	65535
Pos2	演算範囲指定 2	UINT	FeatureValues[1]を演算する範囲の最終要素番号を指定	データ型に従う	—	65535
Pos3	演算範囲指定 3	UINT	FeatureValues[2]を演算する範囲の最終要素番号を指定	データ型に従う	—	65535
Pos4	演算範囲指定 4	UINT	FeatureValues[3]を演算する範囲の最終要素番号を指定	データ型に従う	—	65535
CycleNum	1 周期演算要素数	UINT	1 タスク周期で演算する要素数を指定する。 0 を指定した場合、演算対象範囲要素を 1 タスク周期で処理	データ型に従う	—	0

出力変数

	名称	データ型	内容	有効範囲	単位	初期値
Done	完了	BOOL	TRUE : 正常終了 FALSE : 異常終了、実行中、または実行条件非成立	TRUE, FALSE	—	—
Busy	実行中	BOOL	TRUE : 実行中 FALSE : 実行中ではない	TRUE, FALSE	—	—
FeatureValues[]	特徴量	ARRAY[0..3] OF OmronLib\DataRecorder\sFeatureValue	特徴量演算結果	—	—	—
Error	エラー	BOOL	TRUE : 異常終了 FALSE : 正常終了、実行中、または実行条件非成立	TRUE, FALSE	—	—
ErrorID	エラーコード	WORD	異常終了時の異常 ID 正常終了時は 16#0	*1	—	—
ErrorIDEx	拡張エラーコード	DWORD	異常終了時の拡張異常 ID 正常終了時は 16#0	*1	—	—

*1. 詳細は「トラブルシューティング (P.5 - 26)」を参照してください。

入出力変数

	名称	データ型	内容	有効範囲	単位	初期値
TestData[]*1	演算対象データ	ARRAY[*] OF OmronLib\DataRecorder\sLogData	特徴量を演算する対象となる配列を指定	—	—	—

*1. 配列の要素数は任意です。また、配列要素の開始番号は 0 でも 0 以外でもかまいません。

構造体

- OmronLib\DataRecorder\sFeatureValue

名称	データ型	内容
xMean	REAL	平均値
xSD	REAL	標準偏差
xSkewness	REAL	歪度
xKurtosis	REAL	尖度
xMax	REAL	最大値
xMin	REAL	最小値

- OmronLib\DataRecorder\sLogData
「DataRecorder (P.5 - 12)」を参照してください。

機能

「Execute」の立ち上がりで、「Pos1」、「Pos2」、「Pos3」、「Pos4」で指定した範囲の「TestData[]」の要素の平均値、標準偏差、歪度、尖度、最大値、最小値を演算し、演算結果を「FeatureValues[]」に格納します。

- 演算範囲指定について

演算範囲と計算結果格納場所の関連は以下のとおりです。

4つの演算範囲の要素が重複しても演算できます。また、開始要素と最終要素の大小が逆転しても演算できます。

演算範囲		演算結果格納場所
開始要素	最終要素	
TestData[配列先頭]	TestData[Pos1]*1	FeatureValues[0]
TestData[Pos1+1]*2	TestData[Pos2]*1	FeatureValues[1]
TestData[Pos2+1]*2	TestData[Pos3]*1	FeatureValues[2]
TestData[Pos3+1]*2	TestData[Pos4]*1	FeatureValues[3]

*1. 「Pos1」、「Pos2」、「Pos3」、「Pos4」に、65535 または「TestData[]」の要素数を超える値を設定すると、その演算範囲、およびそれ以降の演算範囲が無効となり、演算結果はすべて 0 が格納されます。

*2. 「Pos1」+1、「Pos2」+1、「Pos3」+1 が「TestData[]」の範囲外となる場合は、その演算範囲、およびそれ以降の演算範囲が無効となり、演算結果はすべて 0 が格納されます。

Pos1	Pos2	Pos3	Pos4	FeatureValues [0]	FeatureValues [1]	FeatureValues [2]	FeatureValues [3]
65535	無効	無効	無効	0	0	0	0
有効値	65535	無効	無効	演算結果	0	0	0
有効値	有効値	65535	無効	演算結果	演算結果	0	0
有効値	有効値	有効値	65535	演算結果	演算結果	演算結果	0
有効値	有効値	有効値	有効値	演算結果	演算結果	演算結果	演算結果

- 「Pos1」、「Pos2」、「Pos3」、「Pos4」の指定方法

「Pos1」、「Pos2」、「Pos3」、「Pos4」は、「TestData[]」配列先頭からの相対値で指定してください。

例: 「TestData[10..100]」のとき、「Pos1」が 0 の場合は「TestData[10]」を指します。

- 実行中の設定変更について

本 FB の多重起動および、実行中の「Pos1」、「Pos2」、「Pos3」、「Pos4」、「CycleNum」の変更は受け付けません。

- 「CycleNum」の指定について

「CycleNum」に 1 タスク周期あたり演算要素数を指定することにより、本 FB の 1 タスク周期の実行時間、および演算完了までの時間を調節することができます。

小さな値を指定した場合、1 タスク周期毎の実行時間が短くなりますが、演算完了までの実行周期が長くなります。

大きな値を指定した場合、演算完了までの時間が短くなりますが、1 タスク周期毎の本 FB 実行時間が長くなります。

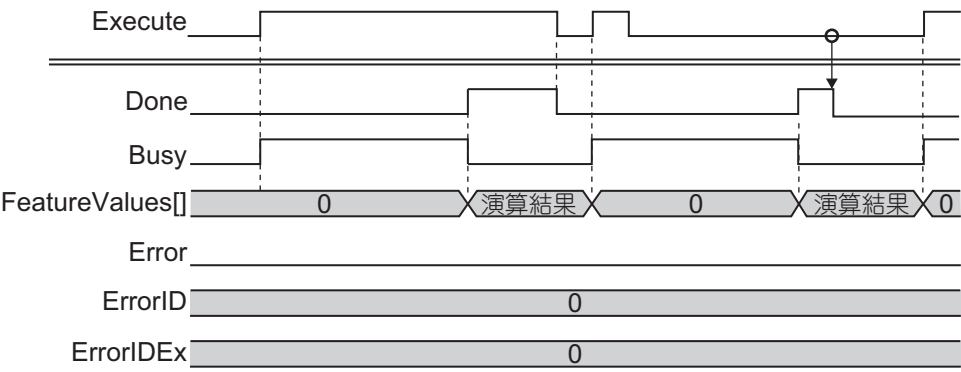
「TestData[]」の配列要素数、本 FB を配置するタスクの周期、本 FB 以外に動作するプログラムの負荷状況を考慮して調節してください。

タイミングチャート

タイミングチャートを、以下に示します。

- 「Execute」の立ち上がりで「Busy」を TRUE にします。
- 全特微量の演算が完了すると「Done」を TRUE にし、演算結果を「FeatureValues[]」に出力します。
- 「Done」は「Execute」を FALSE にするまで保持します。
- 「FeatureValues[]」は「Execute」を FALSE にした後、再度「Execute」を TRUE にするまで保持します。

- 「Busy」が TRUE の間に「Busy」が FALSE となった場合は、「Done」は演算終了後の 1 サイクルのみ TRUE になります。



使用上の注意

本 FB は、「Execute」の値が FALSE になっても、また実行時間がタスク周期を超えても、処理を最後まで続けます。
処理が正常終了したかどうかは、「Done」の値が TRUE になることで確認してください。

トラブルシューティング

エラーコード	拡張エラーコード	状態	内容	対処法
16#0000	16#00000000	正常終了	—	—

サンプルプログラム

NX シリーズ 高速アナログ入力ユニットが、1 タスク周期あたり 100 回サンプリングしたアナログ入力値を DeviceVariableToArray_100 でひとつの配列変数に結合します。さらに、結合したデータを DataRecorder で記録し、記録したデータと検査基準データを「LimitTest」で比較し検査するプログラムです。
NX シリーズ 高速アナログ入力ユニットを、NX シリーズ CPU ユニットに接続したときの例です。
本サンプルプログラムは、プライマリ定周期タスクで実行するプログラム 0 と定周期タスク 17 で動作するプログラム 1 の二つの POU で構成されます。

プログラム 0 (プライマリ定周期タスク)

- 外部変数

名称	データ型	コ ン ス タ ン ト	コ メ ン ト
NXBus_N2_NX_Unit_I_O_Data_Active_Stat us	BOOL		
NXBus_N2_NX_Unit_Error_Status	BOOL		
N2_Ch1_Analog_Input_Value_1_10	ARRAY[0..9] OF INT		

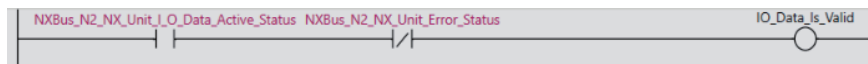
名称	データ型	コ ン ス タ ン ト	コメント
N2_Ch1_Analog_Input_Value_11_20	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_21_30	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_31_40	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_41_50	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_51_60	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_61_70	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_71_80	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_81_90	ARRAY[0..9] OF INT		
N2_Ch1_Analog_Input_Value_91_100	ARRAY[0..9] OF INT		
N2_Ch1_Number_of_Samplings	UINT		
N2_Analog_Input_Time_Stamp	ULINT		
N2_Ch1_Trigger_Input_Time_Stamp	ULINT		
N2_Ch1_Trigger_Input	BOOL		
Ch1_Input_Array100	ARRAY[0..99] OF REAL		1 タスク周期分のアナログ入力値
LogData	ARRAY[0..4999] OF OmronLib\DataRecorder \sLogData		記録先データ配列
RecordInfo	OmronLib\DataRecorder \sRecordInfo		
RecordEnd	BOOL		

- 内部変数

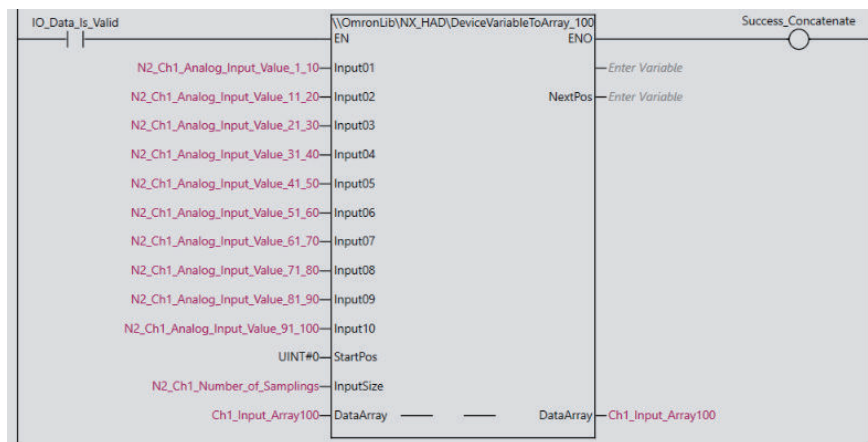
名称	データ型	初期値	割 付 先	保 持	コ ン ス タ ン ト	コメント
TrigControl_instance	OmronLib \DataRecorder \TrigControl					
DataRecorder_instance	OmronLib \DataRecorder \DataRecorder					
IO_Data_Is_Valid	BOOL					
TrigConfig	OmronLib \DataRecorder \sTrigConfig					
TrigMark	OmronLib \DataRecorder \sTrigMark					
TrigOut	BOOL					
DataBuf	ARRAY[0..4999] OF OmronLib \DataRecorder \sLogData					

名称	データ型	初期値	割付先	保持	コンスタント	コメント
Start_P3	BOOL	FALSE				本変数の値を TRUE にすると、プログラムを実行します。
LogEnable	BOOL					
Success_Concatenate	BOOL					

- タスク設定
プライマリ定周期タスクに配置してください。
- LD プログラム
 - (1) NX シリーズ 高速アナログ入力ユニットからの正常なデータ受信を確認します。NX シリーズ 高速アナログ入力ユニットを EtherCAT スレーブターミナルに接続する場合は、「サンプルプログラム 2 (P.4 - 7)」を参照してください。

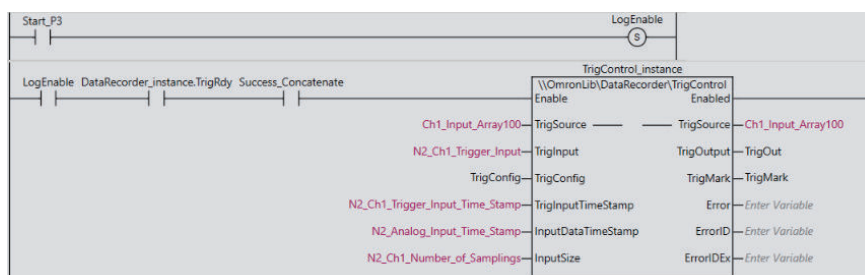


- (2) 1 タスク周期分のアナログ入力値をひとつの変数「Ch1_Input_Array100[]」に結合します。

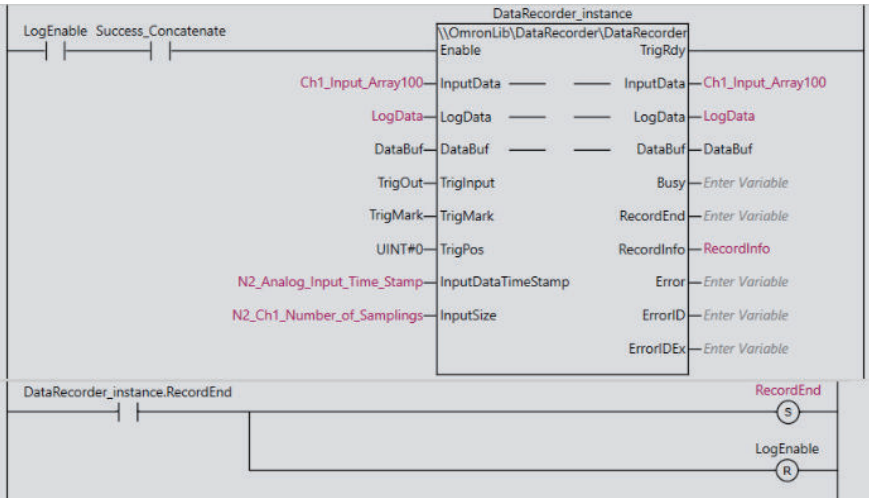


- (3) 「Start_P3」を TRUE にすると、NX シリーズ 高速アナログ入力ユニットの入力値記録のトリガ待ち状態になります。トリガ条件は「TrigConfig」で設定します。

- トリガ条件
「TrigConfig.TrigMode」=2 (タイムスタンプトリガモード)
「TrigInput」= 「N2_Ch1_Trigger_Input」(NX シリーズ 高速アナログ入力ユニットの Ch1 トリガ入力)



(4) トリガ条件が成立すると、トリガ条件成立から 5,000 サンプリングのデータを「LogData[]」に記録します。



プログラム 1 (定周期タスク 17)

- 外部変数

名称	データ型	コン ス タ ン ト	コメント
RecordEnd	BOOL		
RecordInfo	OmronLib\DataRecorder \sRecordInfo		
LogData	ARRAY[0..4999] OF OmronLib\DataRecorder \sLogData		

- 内部変数

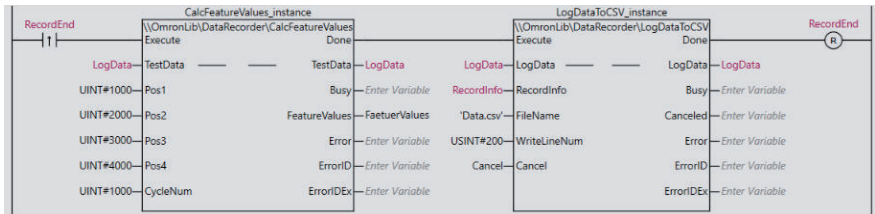
名称	データ型	初 期 値	割 付 先	保 持	コン ス タ ン ト	コメント
CalcFeatureValues_inst tance	OmronLib\DataRecorder \CalcFeatureValues					
LogDataToCSV_instan ce	OmronLib\DataRecorder \LogDataToCSV					
FaetuerValues	ARRAY[0..3] OF OmronLib \DataRecorder \sFeatureValue					
Cancel	BOOL					

- タスク設定
定周期タスク 17 に配置してください。

・ LD プログラム

- (1) 「プログラム 0 (プライマリ定周期タスク) (P.5 - 26)」で実行するデータの記録が完了し、「RecordEnd」が TRUE になると、記録データ「LogData[]」の特徴量を、「Pos1」～「Pos4」で指定するデータの区間ごとに算出します。

特徴量の算出が完了すると、「LogData[]」のデータを「Data.csv」というファイル名で SD メモリカードに書き込みます。



「FeatureValues[]」には以下のように特徴量を格納します。

	.xMean	.xSD	.xSkewness	.xKurtosis	.xMax	.xMin	備考
FeatureValues[0]	平均値	標準偏差	歪度	尖度	最大値	最小値	LogData[0..1000]の特徴量
FeatureValues[1]	平均値	標準偏差	歪度	尖度	最大値	最小値	LogData[1001..2000]の特徴量
FeatureValues[2]	平均値	標準偏差	歪度	尖度	最大値	最小値	LogData[2001..3000]の特徴量
FeatureValues[3]	平均値	標準偏差	歪度	尖度	最大値	最小値	LogData[3001..4000]の特徴量



使用上の注意

- ・ サンプルプログラムは、本ライブラリの機能を使用する部分の記述例です。実際の装置のプログラムを作成する場合は、装置の動作に関する安全回路やインタロックプログラム、ほかの機器からの入出力、制御手順等に関するプログラムを作成してください。
- ・ 実際の装置が意図した動作となるように、ユーザプログラムを作成してください。
- ・ 作成したユーザプログラムは、十分な動作確認を行ったのち、本運転に移行してください。

LogDataToCSV

DataRecorder FB で作成したログデータを SD メモリカードに CSV 形式ファイル(*.csv)で出力します。

FB/FUN 名	名称	FB/FUN	グラフィック表現	ST 表現
LogDataToCSV	ログデータ CSV ファイル書込	FB	LogDataToCSV_instance \\OmronLib\DataRecorder\LogDataToCSV Execute Done LogData LogData RecordInfo Busy FileName Canceled WriteLineNum Error Cancel ErrorID ErrorIDEx	LogDataToCSV_instance(Execute, LogData, RecordInfo, FileName, WriteLineNum, Cancel, Done, Busy, Canceled, Error, ErrorID, ErrorIDEx);

ライブラリ情報

項目	内容
ライブラリファイル名	OmronLib_DataRecorder_Vx_x.slr (x はバージョンを示す)
名前空間	OmronLib\DataRecorder
FB/FUN 番号	00199
ソースコード	非公開

入力変数

	名称	データ型	内容	有効範囲	単位	初期値
Execute	起動	BOOL	TRUE : 実行する FALSE : 実行しない	TRUE, FALSE	—	FALSE
RecordInfo	記録情報	OmronLib\DataRecorder\RecordInfo	DataRecorder FB が出力するレコード情報を入力	—	—	—
FileName	ファイル名	STRING[66]	書き込む CSV ファイルのファイル名。 ディレクトリを含めて指定可能	最大 66 バイト (半角英数字 65 文字 + 終端 NULL 文字)	—	''
WriteLineNum	1 周期出力行数	USINT	本パラメータでは、実行周期ごとに書き込む行数を指定	1～255	—	1
Cancel	中断	BOOL	TRUE : ファイル書き込みを中断	TRUE, FALSE	—	FALSE

出力変数

	名称	データ型	内容	有効範囲	単位	初期値
Done	完了	BOOL	TRUE : 正常終了 FALSE : 異常終了、実行中、または実行条件非成立	TRUE, FALSE	—	—
Busy	実行中	BOOL	TRUE : 実行中 FALSE : 実行中ではない	TRUE, FALSE	—	—
Canceled	中断終了	BOOL	TRUE : 中断 FALSE : 中断されていない	TRUE, FALSE	—	—
Error	エラー	BOOL	TRUE : 異常終了 FALSE : 正常終了、実行中、または実行条件非成立	TRUE, FALSE	—	—
ErrorID	エラーコード	WORD	異常終了時の異常 ID 正常終了時は 16#0	*1	—	—
ErrorIDEx	拡張エラーコード	DWORD	異常終了時の拡張異常 ID 正常終了時は 16#0	*1	—	—

*1. 詳細は「トラブルシューティング (P.5 - 35)」を参照してください。

入出力変数

	名称	データ型	内容	有効範囲	単位	初期値
LogData[] *1	ログデータ	ARRAY[*] OF OmronLib \DataRecorder \sLogData	CSV ファイルに読み出す ログデータを指定	—	—	—

*1. 配列の要素数は任意です。また、配列要素の開始番号は 0 でも 0 以外でもかまいません。

構造体

- OmronLib\DataRecorder\sLogData
「DataRecorder (P.5 - 12)」を参照してください。
- OmronLib\DataRecorder\sRecordInfo
「DataRecorder (P.5 - 12)」を参照してください。

機能

「Execute」の立ち上がりで、「LogData[]」に格納されているログデータと、「RecordInfo」を、SD メモリカードにカンマ区切りの CSV ファイル形式で出力します。出力ファイル名は、「FileName」で指定します。

「FileName」は、ディレクトリを含めて指定できます。指定したディレクトリが存在しない場合は、異常になります。

ディレクトリを指定しない場合は、出力ファイルを SD メモリカードのルートディレクトリに作成します。

書き込み実行中に「Cancel」を TRUE にした場合、ファイルの書き込みを中断します。中断の処理終了で「Canceled」を TRUE にします。中断されても、すでに出力済のファイルは SD メモリカードに残ります。

- ファイルフォーマット
ファイルフォーマットは以下のとおりです。

```

"StartDateTime"      , "(RecordInfo.StartDateTime)"
"EndDateTime"        , "(RecordInfo.EndDateTime)"
"TriggerDateTime"     , "(RecordInfo.TriggerDateTime)"
"RecordNum"           , "(RecordInfo.RecordNum)"
"TriggerPos"          , "(RecordInfo.TrigPos)"
"TimeStamp"           , "Data"
"(LogData[m].xTime)"  , "(LogData[m].xData)"
"(LogData[m+1].xTime)" , "(LogData[m+1].xData)"
:
:
:
"(LogData[n-1].xTime)" , "(LogData[n-1].xData)"
"(LogData[n].xTime)"   , "(LogData[n].xData)"

```

左のデータと右のデータはカンマ(,)で区切ります。
 ()は、カッコ内に記載された変数の値を出力します。
 m は「LogData[]」の先頭要素番号、n は最終要素番号を意味します。

出力例は以下のとおりです。

```

"StartDateTime" , "2017-11-22-10:20:30.123456789"
"EndDateTime"   , "2017-11-22-10:20:305.123456789"
"TriggerDateTime" , "2017-11-22-10:20:303.123456789"
"RecordNum"     , "10000"
"TriggerPos"    , "5000"
"TimeStamp"     , "Data"
"123456789"     , "0.98765"
"123456790"     , "0.88765"
"123456791"     , "0.98765"
:
:
:
"123457000"     , "0.88765"
"123457001"     , "0.89765"

```

- 実行中の設定変更について
 本 FB の多重起動および、実行中の「FileName」、「WriteLineNum」の変更は受け付けません。
- 「WriteLineNum」の指定について
 「WriteLineNum」により 1 タスク周期あたりの出力行数を指定することにより、本 FB の 1 タスク周期の実行時間、および出力完了までの時間を調節することができます。
 小さな値を指定した場合、1 タスク周期毎の実行時間が短くなりますが、演算完了までの実行周期が長くなります。
 大きな値を指定した場合、演算完了までの時間が短くなりますが、1 タスク周期毎の本 FB 実行時間が長くなります。
 「LogData[]」の配列要素数、FB を配置するタスクの周期、本 FB 以外に動作するプログラムの負荷状況を考慮して、調節してください。
 ただし、本 FB 内で 1 タスク周期あたり 1KB を上限として制限します。

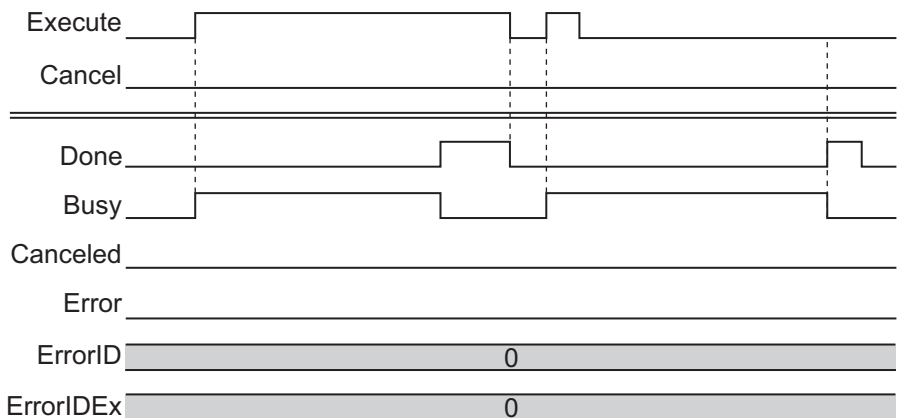
参考

- 「RecordInfo.StartDateTime」、「RecordInfo.EndDateTime」、「RecordInfo.TriggerDateTime」は、DtToString 命令によって、文字列に変換して書き込みます。DtToString 命令の詳細は、『NJ/NX シリーズ コマンド リファレンスマニュアル基本編 (SBCA-468)』を、参照してください。
- 「LogData[].xData」は、RealToFormatString 命令を使用し文字列に変換します。
けた数は、全体が 8、小数部が 6 です。
RealToFormatString 命令の詳細は、『NJ/NX シリーズ コマンド リファレンスマニュアル基本編 (SBCA-468)』を、参照してください。

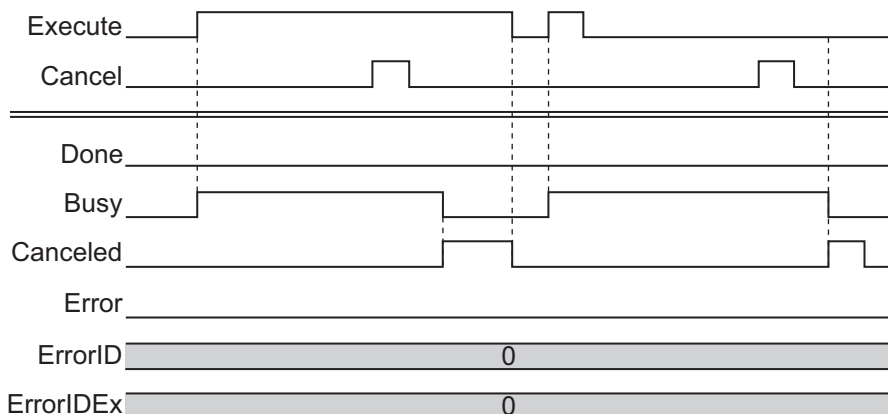
タイミングチャート

タイミングチャートを、以下に示します。

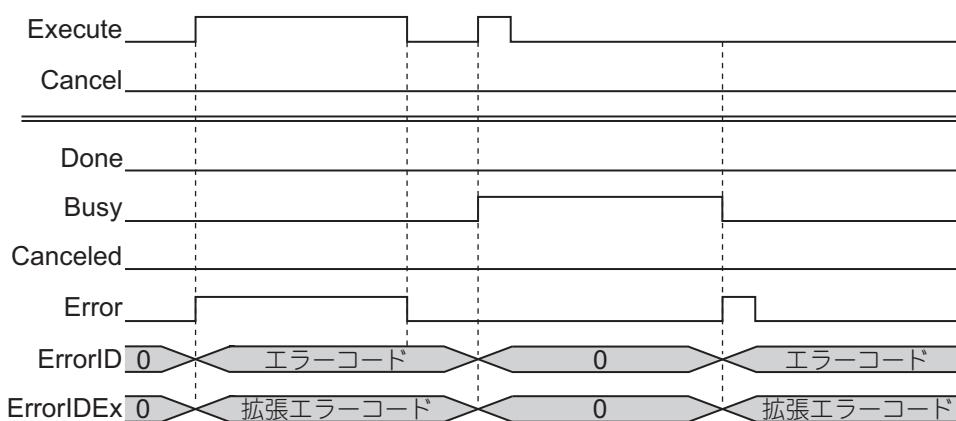
- 「Execute」の立ち上がりで「Busy」を TRUE にします。
 - 全ログデータの書き込みが完了すると、「Done」を TRUE にします。
「Done」は「Execute」を FALSE にするまで保持します。
 - 書き込み実行中に「Cancel」が TRUE になると、ファイル書き込みを中断し、「Canceled」を TRUE にします。
 - 異常が発生した場合、「Error」を TRUE にします。同時に、「ErrorID」、「ErrorIDEx」を出力します。
「Error」、「Canceled」は、「Execute」を FALSE にするまで保持します。
「ErrorID」、「ErrorIDEx」は「Execute」を FALSE にした後、再度「Execute」を TRUE にするまで保持します。
 - 「Busy」が TRUE の間に「Execute」が FALSE となった場合は、「Done」、「Canceled」、「Error」は書き込み/中断処理終了後の 1 サイクルのみ TRUE になります。
- ・ 正常終了時のタイミングチャート



- ・ 中断終了時のタイミングチャート



- ・ 異常終了時のタイミングチャート



使用上の注意

- ・ 本 FB は、「Execute」の値が FALSE になっても、また実行時間がタスク周期を超えても、処理を最後まで続けます。処理が正常終了したかどうかは、「Done」の値が TRUE になることで確認してください。
- ・ 本 FB の実行中に、他の FB から書き込み対象ファイルへアクセスを行わないでください。

トラブルシューティング

エラーコード	拡張エラーコード	状態	内容	対処法
16#0000	16#00000000	正常終了	—	—
16#0400	16#00000000	入力値範囲外	「FileName」で指定したファイル名に使用できない文字が含まれている。 「FileName」で指定したディレクトリ名が長すぎる。	「FileName」を正しく設定してください。 「FileName」で指定した文字列を確認して範囲内に収まるように設定してください。
16#1400	16#00000000	SD メモリカード使用不可	本 FB を NJ/NX シリーズ CPU ユニットで使用する場合に発生します。 『NJ/NX シリーズ トラブルシューティングマニュアル (SBCA-469)』を参照してください。	

エラーコード	拡張 エラーコード	状態	内容	対処法
16#1401	16#00000000	SD メモリカードライト プロテクト		
16#1402	16#00000000	SD メモリカード容量不 足		
16#1404	16#00000000	ファイル・ディレクトリ最 大数超過		
16#1405	16#00000000	ファイルアクセス中		
16#140A	16#00000000	指定ファイル書込禁止		
16#140B	16#00000000	ファイルオープン最大数 超過		
16#140D	16#00000000	ファイル・ディレクトリ名 最大長超過		
16#140E	16#00000000	SD メモリカードアクセ ス失敗		
16#4400	16#00000000	共有フォルダ使用不可	本 FB を NY シリーズ産業用 PC で使用 する場合に発生します。 『NY シリーズ トラブルシューティング マニュアル (SBCA-469)』を参照して ください。	
16#4402	16#00000000	共有フォルダ容量不足		
16#4404	16#00000000	ファイル・ディレクトリ最 大数超過		
16#440D	16#00000000	ファイル・ディレクトリ名 最大長超過		
16#440E	16#00000000	共有フォルダアクセス失 敗		
16#3D05	16#00000001	「WriteLineNum」不正	「WriteLineNum」 が範囲外。	有効範囲内の値を 設定してください。

サンプルプログラム

「サンプルプログラム (P.5 - 7)」を参照してください。

CSVToLogData

LimitTest FB の検査基準データとして、SD メモリカードに記録されたログデータを CSV 形式のファイルから「LogData[]」配列変数に読み出します。

FB/FUN 名	名称	FB/FUN	グラフィック表現	ST 表現
CSVToLogData	ログデータ CSV ファイル 読出	FB	CSVToLogData_instance \\OmronLib\DataRecorder\CSVToLogData Execute Done LogData LogData FileName Busy Cancel RecordInfo NumOfRecord Canceled Error ErrorID ErrorIDEx	CSVToLogData_instance(Execute, LogData, FileName, Cancel, Done, Busy, RecordInfo, NumOfRecord, Canceled, Error, ErrorID, ErrorIDEx);

ライブラリ情報

項目	内容
ライブラリファイル名	OmronLib_DataRecorder_Vx_x.slr (x はバージョンを示す)
名前空間	OmronLib\DataRecorder
FB/FUN 番号	00200
ソースコード	非公開

入力変数

	名称	データ型	内容	有効範囲	単位	初期値
Execute	起動	BOOL	TRUE : 実行する FALSE : 実行しない	TRUE, FALSE	—	FALSE
FileName	ファイル名	STRING[66]	読み出す CSV ファイルのファイル名。 ディレクトリを含めて指定可能	最大 66 バイト (半角英数字 65 文字 + 終端 NULL 文字)	—	—
Cancel	中断	BOOL	TRUE : ファイル読み出しを中断	TRUE, FALSE	—	FALSE

出力変数

	名称	データ型	内容	有効範囲	単位	初期値
Done	完了	BOOL	TRUE : 正常終了 FALSE : 異常終了、実行中、または実行条件非成立	TRUE, FALSE	—	—

	名称	データ型	内容	有効範囲	単位	初期値
Busy	実行中	BOOL	TRUE : 実行中 FALSE : 実行中ではない	TRUE, FALSE	—	—
RecordInfo	記録情報	OmronLib \DataRecorder \sRecordInfo	読み出したログデータの 記録情報。	—	—	—
NumOfRecord	読み出しレコード数	UINT	読み出したレコード数	データ型に従う	—	—
Canceled	中断終了	BOOL	TRUE : 中断 FALSE : 中断されていない	TRUE, FALSE	—	—
Error	エラー	BOOL	TRUE : 異常終了 FALSE : 正常終了、実行中、または実行条件非成立	TRUE, FALSE	—	—
ErrorID	エラーコード	WORD	異常終了時の異常 ID 正常終了時は 16#0	*1	—	—
ErrorIDEx	拡張エラーコード	DWORD	異常終了時の拡張異常 ID 正常終了時は 16#0	*1	—	—

*1. 詳細は「トラブルシューティング (P.5 - 40)」を参照してください。

入出力変数

	名称	データ型	内容	有効範囲	単位	初期値
LogData[] *1	ログデータ	ARRAY[*] OF OmronLib \DataRecorder \sLogData	CSV ファイルに読み出す ログデータを指定	—	—	—

*1. 配列の要素数は任意です。ただし、読み出すファイルに記録されたデータ数以上でなければなりません。また、配列要素の開始番号は 0 でも 0 以外でもかまいません。

構造体

- OmronLib\DataRecorder\sLogData
「DataRecorder (P.5 - 12)」を参照してください。
- OmronLib\DataRecorder\sRecordInfo
「DataRecorder (P.5 - 12)」を参照してください。

機能

「Execute」の立ち上がりで、「FileName」で指定されたファイルからログデータを読み出し、「LogData[]」に格納します。

また、記録情報を「RecordInfo」に格納します。

ファイルから読み出したレコード数を「NumOfRecord」に出力します。

「RecordInfo」、「NumOfRecord」は「Execute」を FALSE にした後、再度「Execute」を TRUE にするまで保持します。

「FileName」は、ディレクトリを含めて指定できます。ディレクトリを指定しない場合は、SD メモリカードのルートディレクトリを参照します。指定されたファイルが存在しない場合は、異常になります。

また、本 FB で扱うことができるのは、LogDataToCSV FB で作成された CSV 形式のファイルです。ファイルフォーマットが異なる場合は異常になります。

読み出し実行中に「Cancel」を TRUE にした場合、ファイルの読み出しを中断します。中断処理終了で「Canceled」を TRUE にします。読み出しを中断した場合、「RecordInfo」、「NumOfRecord」を出力しません。「LogData[]」の内容は不定です。

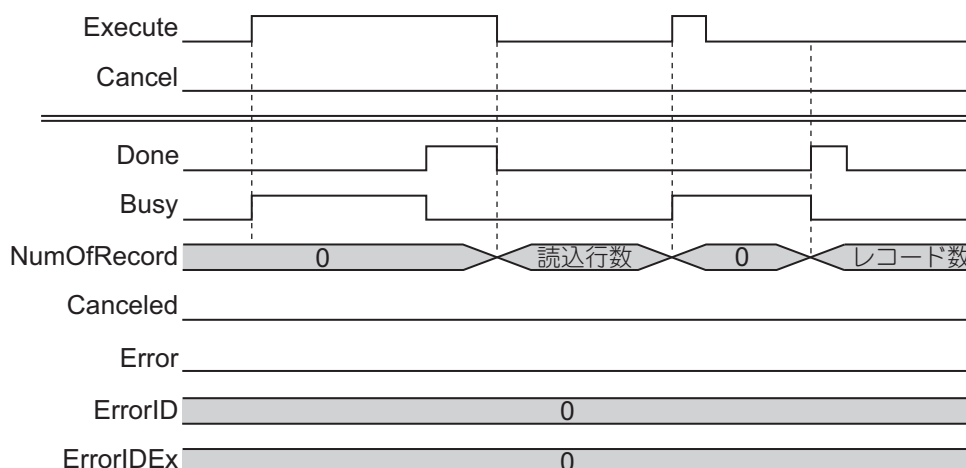
- ファイルフォーマット
LogDataToCSV FB を参照してください。
- 実行中の設定変更について
本 FB の多重起動および、実行中の「FileName」の変更は受け付けません。

タイミングチャート

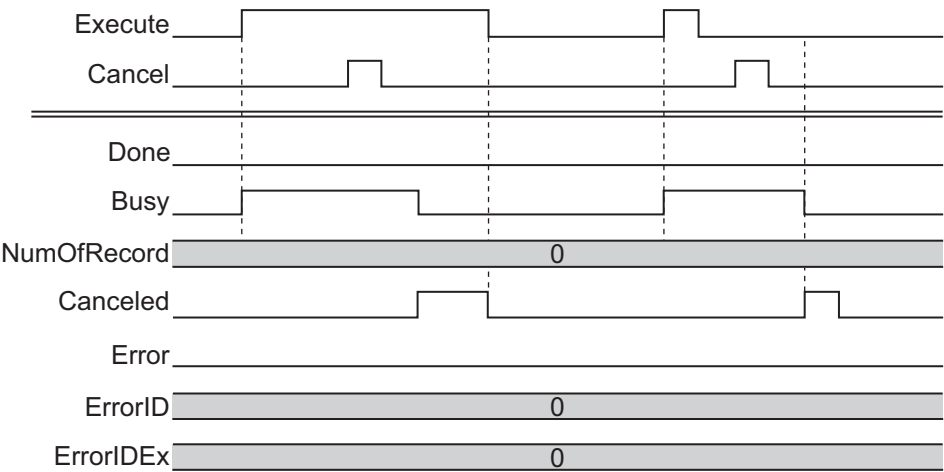
タイミングチャートを、以下に示します。

- 「Execute」の立ち上がりで「Busy」を TRUE にします。
- 全ログデータの読み出しが完了すると、「Done」を TRUE にします。
「Done」は「Execute」を FALSE にするまで保持します。
「RecordInfo」、「NumOfRecord」は「Execute」を FALSE にした後、再度「Execute」を TRUE にするまで保持します。
- 読み出し実行中に「Cancel」が TRUE になると、ファイル読み出しを中断し、「Canceled」を TRUE にします。
- 異常が発生した場合、「Error」を TRUE にします。同時に、「ErrorID」、「ErrorIDEx」を出力します。
「Error」、「Canceled」は、「Execute」を FALSE にするまで保持します。
「ErrorID」、「ErrorIDEx」は「Execute」を FALSE にした後、再度「Execute」を TRUE にするまで保持します。
- 「Busy」が TRUE の間に「Execute」が FALSE となった場合は、「Done」、「Canceled」、「Error」は読み出し/中断処理終了後の 1 サイクルのみ TRUE になります。

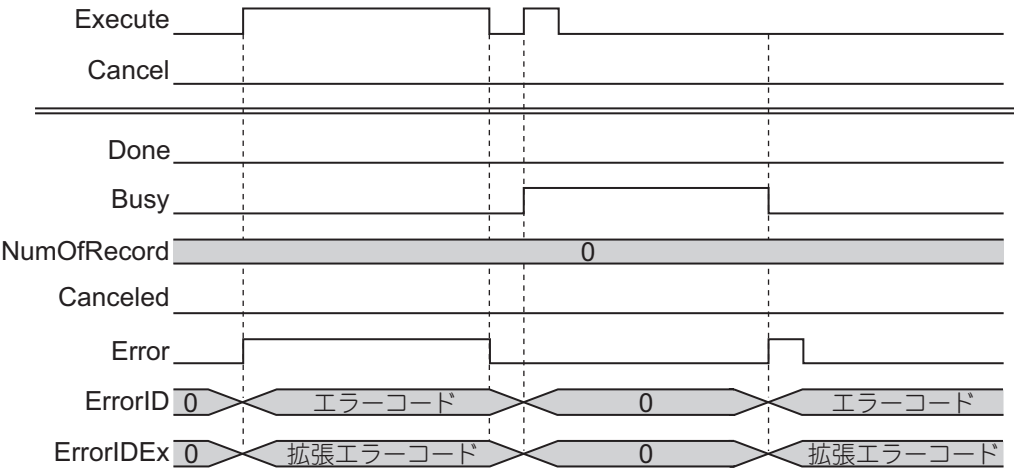
- 正常終了時のタイミングチャート



- 中断終了時のタイミングチャート



・ 異常終了時のタイミングチャート



使用上の注意

- ・ 本 FB は、「Execute」の値が FALSE になっても、また実行時間がタスク周期を超えても、処理を最後まで続けます。処理が正常終了したかどうかは、「Done」の値が TRUE になることで確認してください。
- ・ 本 FB の実行中に、他の FB から読み出し対象ファイルへアクセスを行わないでください。

トラブルシューティング

エラーコード	拡張 エラーコード	状態	内容	対処法
16#0000	16#00000000	正常終了	—	—
16#0400	16#00000000	入力値範囲外	「FileName」で指定したファイル名に使用できない文字が含まれている。 「FileName」で指定したディレクトリ名が長すぎる。	「FileName」を正しく設定してください。 「FileName」で指定した文字列を確認して範囲内に収まるように設定してください。

エラーコード	拡張 エラーコード	状態	内容	対処法
16#1400	16#00000000	SD メモリカード使用不可	本 FB を NJ/NX シリーズ CPU ユニットで使用する場合に発生します。 『NJ/NX シリーズ トラブルシューティングマニュアル (SBCA-469)』を参照してください。	
16#1401	16#00000000	SD メモリカードライトプロテクト		
16#1402	16#00000000	SD メモリカード容量不足		
16#1404	16#00000000	ファイル・ディレクトリ最大数超過		
16#1405	16#00000000	ファイルアクセス中		
16#140A	16#00000000	指定ファイル書込禁止		
16#140B	16#00000000	ファイルオープン最大数超過		
16#140D	16#00000000	ファイル・ディレクトリ名最大長超過		
16#140E	16#00000000	SD メモリカードアクセス失敗	本 FB を NY シリーズ産業用 PC で使用する場合に発生します。 『NY シリーズ トラブルシューティングマニュアル (SBCA-438)』を参照してください。	
16#4400	16#00000000	共有フォルダ使用不可		
16#4402	16#00000000	共有フォルダ容量不足		
16#4404	16#00000000	ファイル・ディレクトリ最大数超過		
16#440D	16#00000000	ファイル・ディレクトリ名最大長超過		
16#440E	16#00000000	共有フォルダアクセス失敗	ファイル内レコード数に対して「LogData[]」の要素数が不足。	ファイル内レコード数を確認し、レコード数以上の要素数に「LogData[]」を拡張してください。
16#3D06	16#00000001	「LogData[]」要素数不足		
	16#00000002	フォーマット不正	ファイル内レコード内容がフォーマットと不一致。	読み出し対象のファイルが LogDataToCSV で出力されたものであるかを確認してください。

サンプルプログラム

「サンプルプログラム (P.5 - 7)」を参照してください。



付録

ライブラリ情報の参照方法や、FB や FUN のソースコードの参照方法など、知っておくと便利な情報について説明します。

A-1	ライブラリ情報の参照.....	A - 2
A-1-1	ライブラリの属性、FB や FUN の属性	A - 2
A-1-2	ライブラリの属性、FB や FUN の属性を参照する方法	A - 2
A-2	FB や FUN のソースコードの参照.....	A - 5

A

A-1 ライブラリ情報の参照

弊社へライブラリに関して問い合わせをする場合、当該ライブラリを特定するために、ライブラリ情報を参照することができます。

ライブラリ情報は、当該ライブラリが、弊社の提供するライブラリやユーザが作成したライブラリの中の、どのライブラリなのかを特定するための有効な情報です。

ライブラリ情報には、ライブラリ自身の属性とライブラリ内の FB や FUN の属性があります。

- ライブラリの属性
ライブラリ自身を特定する情報
 - FB や FUN の属性
ライブラリ内のファンクションブロックやファンクションを特定する情報
- ライブラリ情報は Sysmac Studio で参照することができます。

A-1-1 ライブラリの属性、FB や FUN の属性

ライブラリ情報のライブラリの属性、FB や FUN の属性は次のとおりです。

ライブラリの属性

No.*1	属性	内容
(1)	ライブラリファイル名	ライブラリファイルの名称
(2)	ライブラリバージョン	ライブラリのバージョン
(3)	作者名	ライブラリの作成者名
(4)	コメント	ライブラリの内容説明*2

*1. 「A-1-2 ライブラリの属性、FB や FUN の属性を参照する方法 (P.A - 2)」の画面イメージに記載している番号です。

*2. 英語と日本語で表示されます。

FB や FUN の属性

No.*1	属性	内容
(5)	FB/FUN 名	ファンクションブロックやファンクションの名称
(6)	名前空間	ファンクションブロックやファンクションの名前空間名
(7)	FB/FUN バージョン	ファンクションブロックやファンクションのバージョン
(8)	作成者	ファンクションブロックやファンクションの作成者名
(9)	FB/FUN 番号	ファンクションブロックやファンクションの番号
(10)	コメント	ファンクションブロックやファンクションの機能説明*2

*1. 「A-1-2 ライブラリの属性、FB や FUN の属性を参照する方法 (P.A - 2)」の画面イメージに記載している番号です。

*2. 英語と日本語で表示されます。

A-1-2 ライブラリの属性、FB や FUN の属性を参照する方法

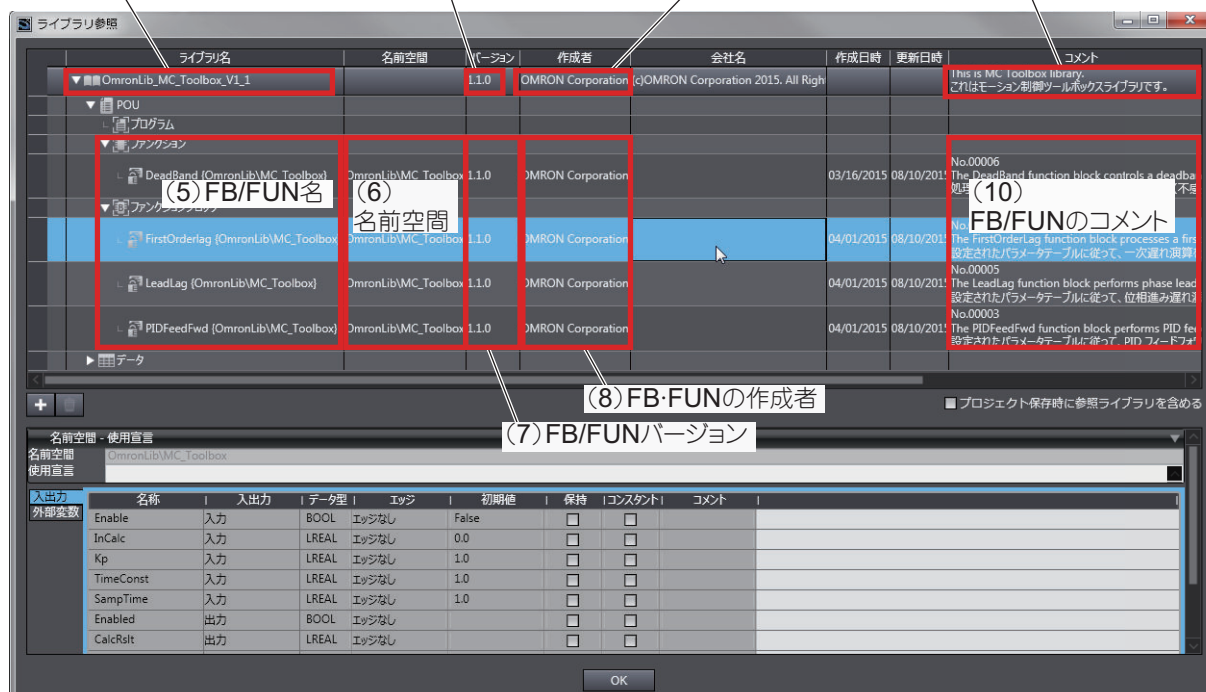
ライブラリ情報のライブラリの属性、FB や FUN の属性は、Sysmac Studio の次の箇所で参照できます。

- ライブラリ参照画面
- ツールボックス
- プログラミング画面

ライブラリ参照画面

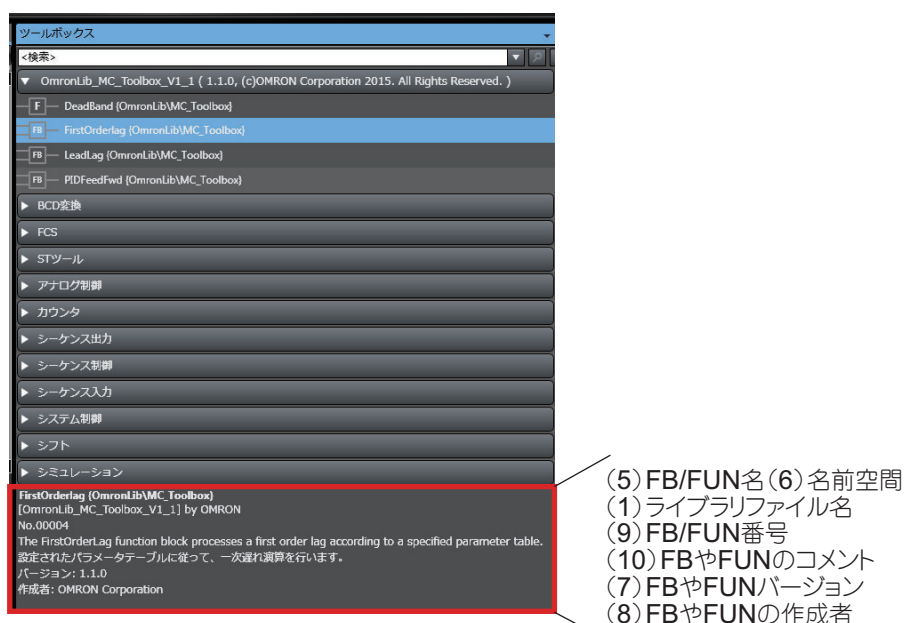
ライブラリを参照すると、下図の位置にライブラリ情報が表示されます。

- (1)ライブラリファイル名 (2)ライブラリバージョン (3)ライブラリの作成者 (4)ライブラリのコメント



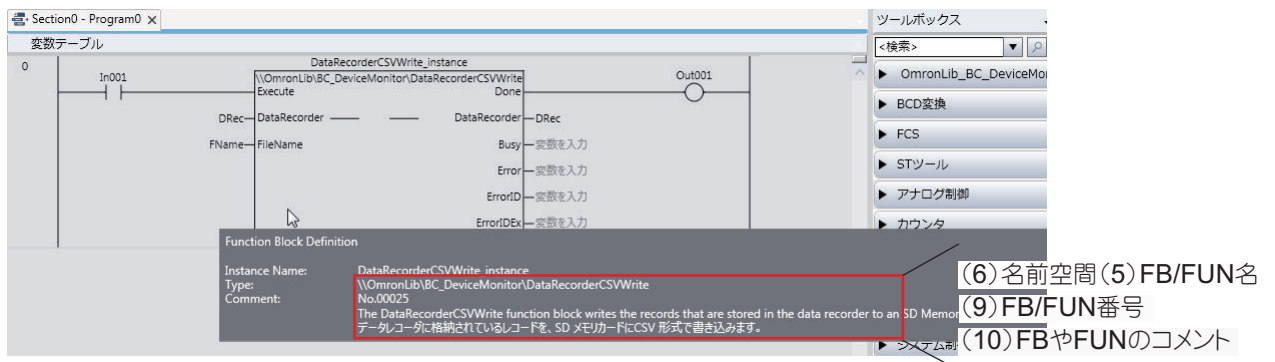
ツールボックス

FB や FUN を選択すると、ツールボックスの下部にライブラリ情報が表示されます。(1)のライブラリ名の右横の [by OMRON] は、弊社が提供するライブラリであることを示します。



プログラミング画面

FB や FUN にフォーカスをあてると、ツールチップが表示され、ライブラリ情報が表示されます。



A-2 FB や FUN のソースコードの参照

弊社が提供する FB や FUN をユーザの環境に合わせてカスタマイズするために、FB や FUN のソースコードを参照することができます。

ソースコードをコピーすることで、ユーザの FB や FUN を作成することができます。

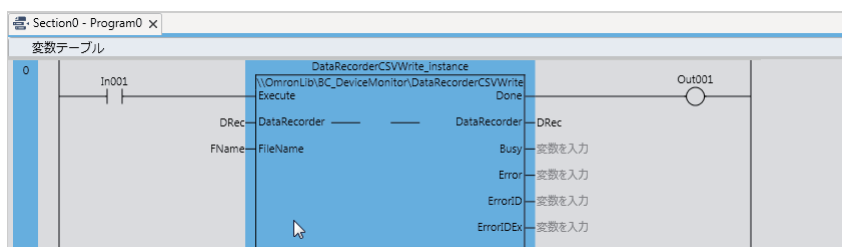
下記はカスタマイズする項目の例です。

- 使用しているコントローラのメモリ容量に合わせた“配列サイズ”のカスタマイズ
- ユーザで定義されているデータ型に合わせた“データ型”のカスタマイズ

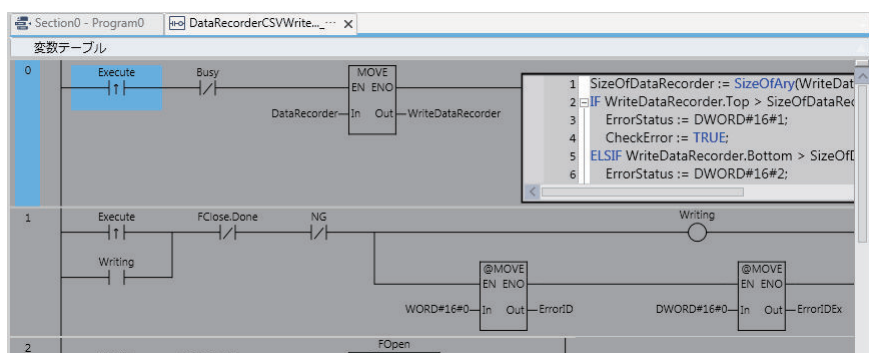
ただし、ソースコードが参照できるのは、FB や FUN の個別仕様に記載されているライブラリ情報のソースコード公開／非公開が“公開”になっているものだけです。

FB や FUN のソースコードを参照する方法は、以下のとおりです。

1 プログラム上で FB/FUN を選択します。

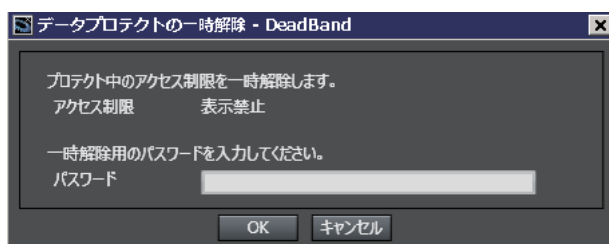


2 ダブルクリックまたは右クリックしメニューの「下の階層へ」を選択します。ソースコードが表示されます。



使用上の注意

- ソースコードが非公開の FB や FUN の場合、上記の手順 2 で、下記のダイアログボックスが表示されます。[キャンセル] ボタンをクリックしてください。





索引

索引

C

CalcFeatureValues.....	5 - 23
CSVToLogData.....	5 - 37

D

DataRecorder.....	5 - 12
DeviceVariableToArray_***.....	4 - 2

L

LimitAlarm_HAD.....	4 - 14
LimitTest.....	5 - 18
LogDataToCSV.....	5 - 31

S

ScaleTrans_HAD.....	4 - 11
---------------------	--------

T

TrigControl.....	5 - 2
------------------	-------

- 本誌に記載のない条件や環境での使用、および原子力制御・鉄道・航空・車両・燃焼装置・医療機器・娯楽機械・安全機器、その他人命や財産に大きな影響が予測されるなど、特に安全性が要求される用途に使用される際には、当社の意図した特別な商品用途の場合や特別の合意がある場合を除き、当社は当社商品に対して一切保証をいたしません。
- 本製品の内、外国為替及び外国貿易法に定める輸出許可、承認対象貨物(又は技術)に該当するものを輸出(又は非居住者に提供)する場合は同法に基づく輸出許可、承認(又は役務取引許可)が必要です。

オムロン株式会社 インダストリアルオートメーションビジネスカンパニー

- 製品に関するお問い合わせ先
お客様相談室

フリー
通話 **0120-919-066**

携帯電話・PHS・IP電話などではご利用いただけませんので、下記の電話番号へおかけください。

電話 **055-982-5015** (通話料がかかります)

■営業時間：8:00～21:00 ■営業日：365日

- FAXやWebページでもお問い合わせいただけます。

FAX **055-982-5051** / **www.fa.omron.co.jp**

- その他のお問い合わせ

納期・価格・サンプル・仕様書は貴社のお取引先、または貴社担当オムロン販売員にご相談ください。
オムロン制御機器販売店やオムロン販売拠点は、Webページでご案内しています。

オムロン制御機器の最新情報をご覧ください。

www.fa.omron.co.jp

緊急時のご購入にもご利用ください。

オムロン商品のご用命は