

Sysmac Library

ユーザーズマニュアル
RFID通信ライブラリ編
形SYSMAC-XR019

おことわり

- 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。
- 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。
- 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または商標です。
- **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会社名および製品名は、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諾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、NX シリーズ RFID ユニット形 NX-V680C□ 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

RFID 通信ライブラリは、NJ/NX シリーズ CPU ユニット、NY シリーズ産業用 PC と NX シリーズ RFID ユニット形 NX-V680C□ を使って、個体識別による生産管理を行う際に、RF タグのメモリを読み書きする機能を提供します。RFID 通信ライブラリを使用することで、RFID ユニットの処理を実装する際のプログラミング工数を削減することができます。

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

本マニュアルは以下の製品を対象にしています。

項目	名称	形式	バージョン
Sysmac Library	NX-V680 ライブラリ	形 SYSMAC-XR019	Ver.1.0.0 以降
オートメーションソフトウェア	Sysmac Studio	形 SYSMAC-SE□□□□	Ver.1.23 以降
機器	CPU ユニット	形 NX701-□□□□	Ver.1.18 以降
		形 NJ101-□□□□	
		形 NJ501-□□□□	Ver.1.18 以降
		形 NJ301-□□□□	
		形 NX1P2-□□□□□□(1)	Ver.1.18 以降
		形 NX102-□□□□	Ver.1.30 以降
	産業用 PC	形 NY5□□-1□□□	Ver.1.18 以降
	NX シリーズ RFID ユニット	形 NX-V680C□	Ver.1.0 以降

マニュアルの読み方

アイコン

この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

製品の動作不能、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。



詳しい情報や、関連情報の記載されている場所を示します。

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
アイコン	2
目次	4
ご承諾事項	6
安全上の注意	8
安全に使用していただくための表示と意味	8
図記号の説明	8
注意	9
使用上の注意	10
関連マニュアル	11
マニュアル改訂履歴	14
Sysmac Library の使用手順	15
インストーラからインストールした Sysmac Library の使用手順	16
CPU ユニットまたは産業用 PC 内の Sysmac Library の使用手順	20
FB の共通仕様	23
共通変数	24
注意事項	29
FB の個別仕様	31
FB 共通で使われるデータ型	32
ReadData_V680	34
WriteData_V680	57
付録	79
ライブラリ情報の参照	80
FB や FUN のソースコードの参照	83

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まれます。
- ・「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

このユーザーズマニュアルでは、NJ/NX シリーズ CPU ユニット、NY シリーズ産業用 PC 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



⊘ 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

Sysmac Library、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」による使用を前提にしています。「対象となる読者の方々」に該当する方、もしくはその方の指導を受けた方以外は使用しないでください。



本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。



実際の装置が意図した動作をするように、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。



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。



サンプルプログラムを使用してユーザ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。



使用上の注意

ライブラリ使用時

- ライブラリを使用する場合、ライブラリのマニュアルに記載されていない FB や FUN が Sysmac Studio に表示される場合があります。マニュアルに記載されていない FB や FUN は使用しないでください。
- ライブラリで提供している FB や FUN のソースコードをユーザが変更することはできません。
- ライブラリで提供している FB の処理が正常終了または異常終了するまで、コントローラや通信カプラユニットの電源を OFF にしたり、通信を停止したりしないでください。

サンプルプログラムの使用時

- 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。

関連マニュアル

関連するマニュアルは下表のとおりです。併せてご覧ください。

マニュアル名称	Man.No.	形式	用途	内容
NX シリーズ RFID ユニット ユーザーズマニュアル	SGTG-717	形 NX-V680C□	NX シリーズ RFID ユニットの使用方法について知りたいとき	NX シリーズ RFID 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。
V680 シリーズ タグ・アンプ (FRAM タイプ) ユーザーズマニュアル	SCHI-707	形 V680-HA63B 形 V680-HS□□ 形 V680-H01-V2 形 V680-D2K□□□□□ 形 V680-D8K□□□□□ 形 V680S-D2K□□□□□ 形 V680S-D8K□□□□□	V680 シリーズの RF タグ・アンプ (FRAM タイプ) の仕様 / 性能 / 取付について知りたいとき。	V680 シリーズの RF タグ・アンプ (FRAM タイプ) に関して、一般仕様 / 交信仕様 / 取付方法について説明します。
V680 シリーズ タグ・アンプ (EEPROM タイプ) ユーザーズマニュアル	SCHI-709	形 V680-HA63A 形 V680-HS□□ 形 V680-H01-V2 形 V680-D1KP□□□□□	V680 シリーズの RF タグ・アンプ (EEPROM タイプ) の仕様 / 性能 / 取付について知りたいとき。	V680 シリーズの RF タグ・アンプ (EEPROM タイプ) に関して、一般仕様 / 交信仕様 / 取付方法について説明します。
NX シリーズ データリファレンス マニュアル	SBCA-410	形 NX-□□□□□□	NX シリーズの各ユニットのシステム構成に必要なデータを一覧で閲覧したいとき。	NX シリーズの各ユニットの「消費電力」、「質量」など、システム構築に必要なデータを集めて記載しています。
NX シリーズ システムユニット ユーザーズマニュアル	SBCA-409	形 NX-PD1□□□ 形 NX-PF0□□□ 形 NX-PC0□□□ 形 NX-TBX01	NX シリーズシステムユニットの使用方法について知りたいとき。	NX シリーズシステムユニットのハードウェアや機能について説明します。
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。
NX-IO Configurator オペレーション マニュアル	SBCD-373	形 CXONE-AL□□D-V4	NX-IO Configurator の操作方法、機能について知りたいとき。	NX-IO Configurator の操作方法について説明します。
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 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□ 形 NX1P2-□□□□	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。
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。
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。

マニュアル名称	Man.No.	形式	用途	内容
NX シリーズ EtherNet/IP™ カブラユニット ユーザーズマニュアル	SBCD-365	形 NX-EIC202	NX シリーズ EtherNet/IP カブラ ユニット、および EtherNet/IP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 EtherNet/IP カブラユニットと NX ユ ニットで構成され る EtherNet/IP スレーブ ターミナルのシステム概要 や構成方法、および NX ユ ニットを設定、制御、モニ タするための、EtherNet/IP カブ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 CPU ユニット ユーザーズ マニュアル ハードウェア編	SBCA-418	形 NX701-□□□□	NX シリーズ 形 NX701 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。	形 NX701 CPU ユニットの 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
NX シリーズ 形NX102CPU ユニット ユーザーズマニュアル ハードウェア編	SBCA-462	形 NX102-□□□□	NX シリーズ 形 NX102 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。	形 NX102 CPU ユニットの 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
NX シリーズ 形 NX1P2 CPU ユニット ユーザーズマニュアル ハードウェア編	SBCA-448	形 NX1P2-□□□□	NX シリーズ 形 NX1P2 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。	形 NX1P2 CPU ユニットの 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
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 ユーザーズマニュアル ハードウェア編	SBCA-435	形 NY532-□□□□	NY シリーズ産業用パ ネル型 PC 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。	NY シリーズのシステム全体 概要、および産業用パネル型 PC 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

マニュアル名称	Man.No.	形式	用途	内容
NY シリーズ IPC マシンコントローラ 産業用ボックス型 PC ユーザーズマニュアル ハードウェア編	SBCA-434	形 NY512-□□□□	NY シリーズ産業用 ボックス型 PC 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。	NY シリーズのシステム全体 概要、および産業用ボック ス型 PC 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
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 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□ 形 NX1P2-□□□□	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
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 シリーズマシンオート メーションコントロールソ フトウェアに関して、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
NJ/NX シリーズ CPU ユニット内蔵 EtherCAT® ポート ユーザーズマニュアル	SBCD-376	形 NX701-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□ 形 NX1P2-□□□□	NJ/NX シリー ズ CPU ユニットの内 蔵 EtherCAT ポート を使用するとき。	内蔵 EtherCAT ポートに関 して説明します。 概要、構成、機能、セット アップについて記述してい 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 EtherCAT® ポート編	SBCD-368	形 NY532-□□□□ 形 NY512-□□□□	NY シリーズ産業用 PC の内蔵 EtherCAT ポートを使用すると き。	内蔵 EtherCAT ポートに関 して説明します。 概要、構成、機能、セット アップについて記述してい ます。
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 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□ 形 NX1P2-□□□□	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。
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。

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-473B**

↑
Revision code

改訂記号	改訂年月	改訂理由・改訂ページ
A	2018 年 6 月	初版
B	2021 年 5 月	誤記修正

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

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。

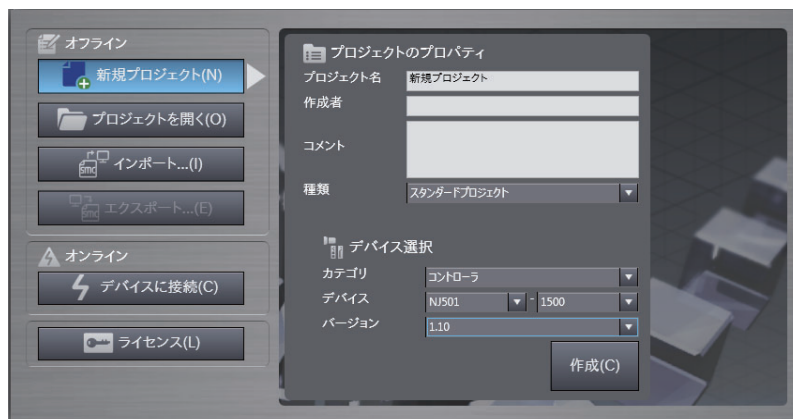
- Sysmac Library を新規で使用する場合
- Sysmac Library をバージョンアップして使用する場合

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。

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
- デバイス選択のバージョンは 1.01 以降を選択

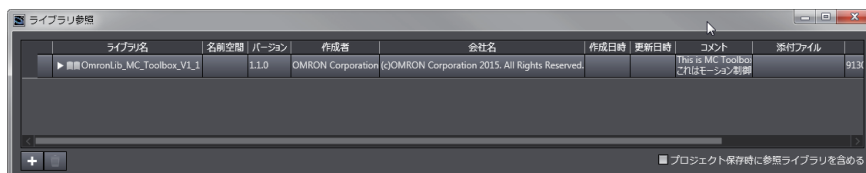
2 [プロジェクト] – [ライブラリ] – [ライブラリ参照] を選択します。



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。

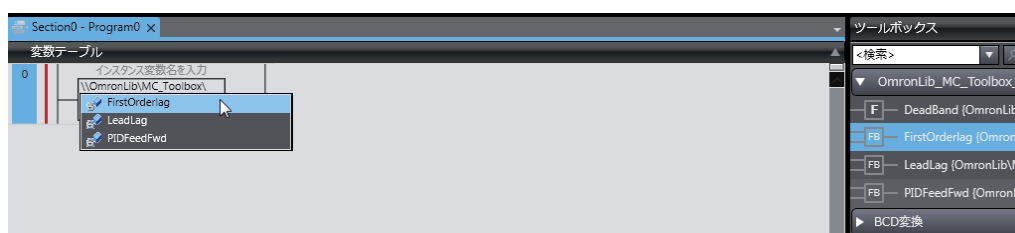
ラダーエディタや ST エディタを選択すると、Sysmac Library の FB や FUN がツールボックスに表示されます。

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2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。

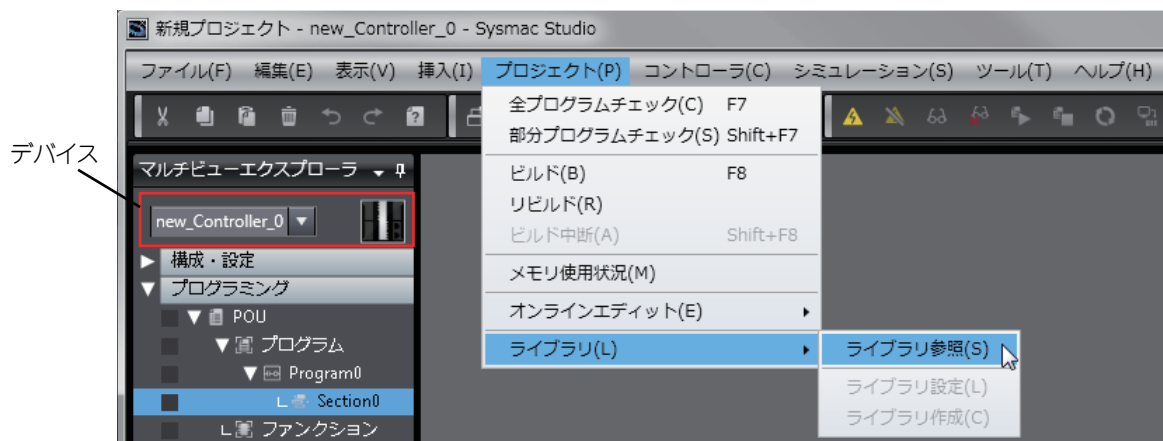


- ラダーエディタ上で右クリックし、メニューの [ファンクションブロックを挿入] を選択し、完全限定名 (¥¥ 名前空間名 ¥FB 名) を入力



Sysmac Library をバージョンアップして使用する場合

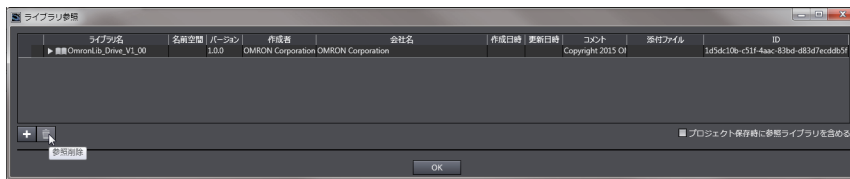
- 1** Sysmac Studio を起動し、旧バージョンの Sysmac Library を使用しているプロジェクトを開きます。
- 2** [プロジェクト] – [ライブラリ] – [ライブラリ参照] を選択します。



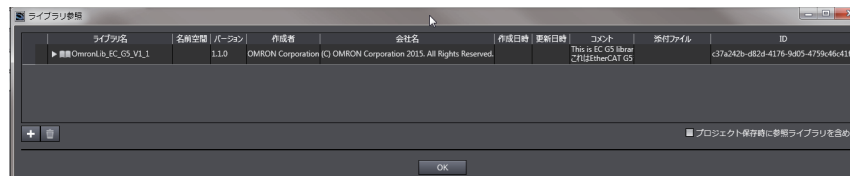
使用上の注意

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。

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

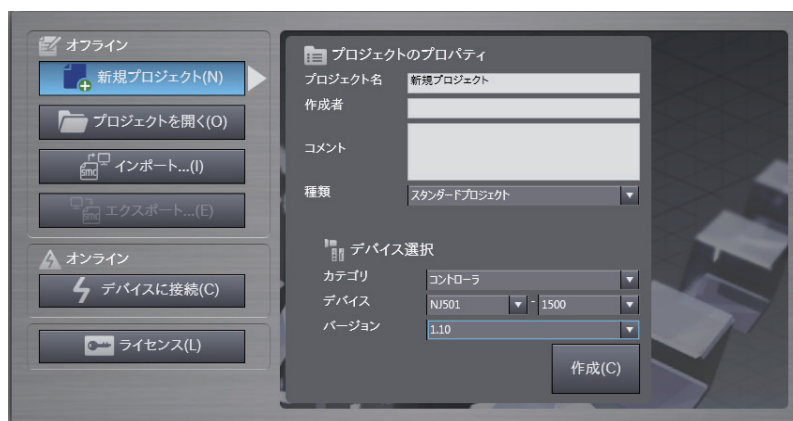
Sysmac Library をパソコンにインストールしていない場合でも、CPU ユニットまたは産業用 PC 内の Sysmac Library をパソコンにアップロードして使用できます。
CPU ユニットまたは産業用 PC 内の Sysmac Library の使用手順を、次に示します。



バージョン関連情報

Sysmac Library を使用できるのは、Sysmac Studio Ver.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。



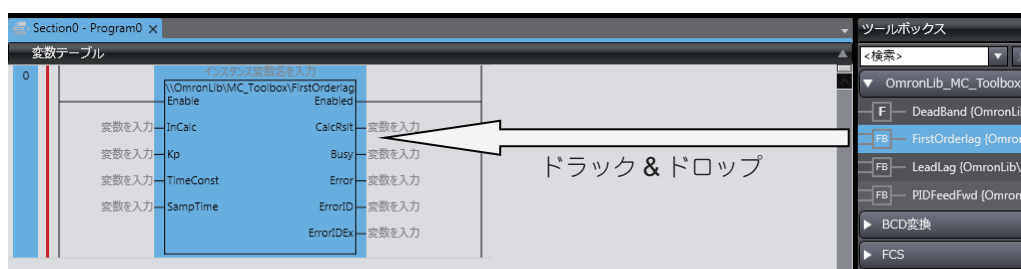
2 CPU ユニットまたは産業用 PC にオンライン接続します。

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

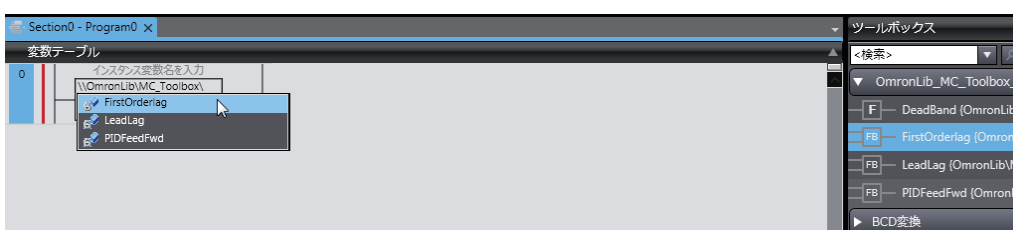
ラダーエディタや ST エディタを選択すると、アップロードした POU が使用している Sysmac Library の FB や FUN がツールボックスに表示されます。

4 次の 2 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。
-

FB の共通仕様

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。

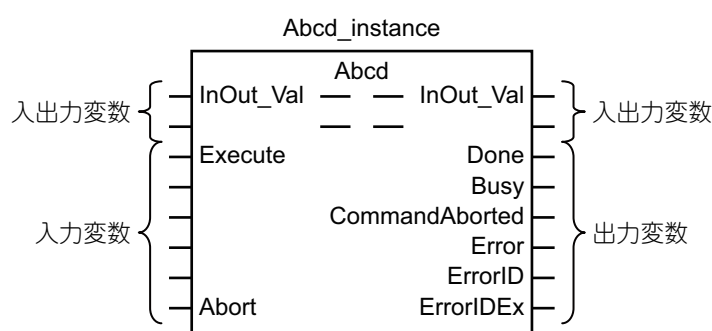
入力変数と出力変数の定義

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			○			起動	立ち上がり時に処理を実行します。
Enable				○		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。 処理を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。
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。

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。

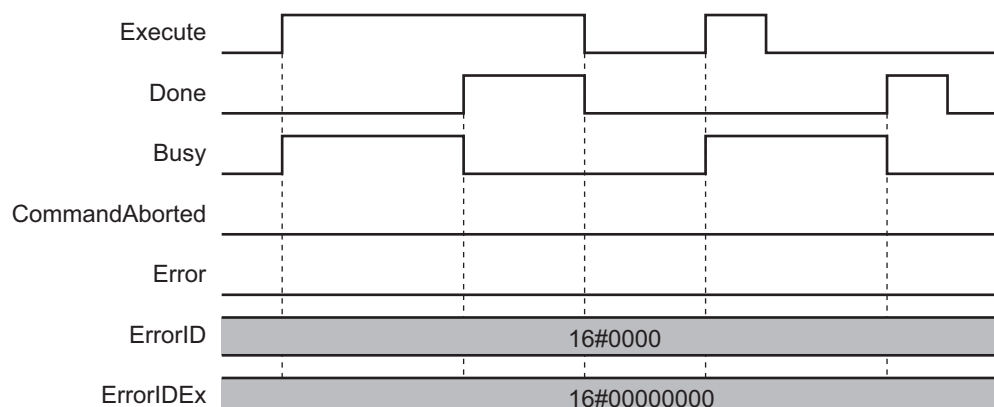


- Execute が TRUE の状態で Done, CommandAborted, Error が TRUE になった場合に Execute を FALSE にすると、Done, CommandAborted, Error も FALSE になります。
- Execute が FALSE の状態で Done, CommandAborted, Error が TRUE になった場合、Done, CommandAborted, Error は、1 タスク周期だけ TRUE になります。
- 異常が発生した場合、ErrorID（エラーコード）、ErrorIDEx（拡張エラーコード）に該当するエラーコード、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。

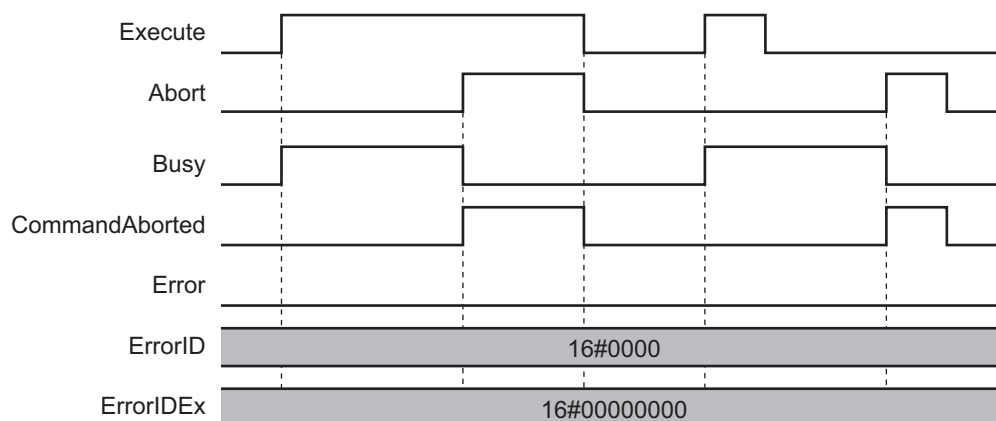
タイミングチャート

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。

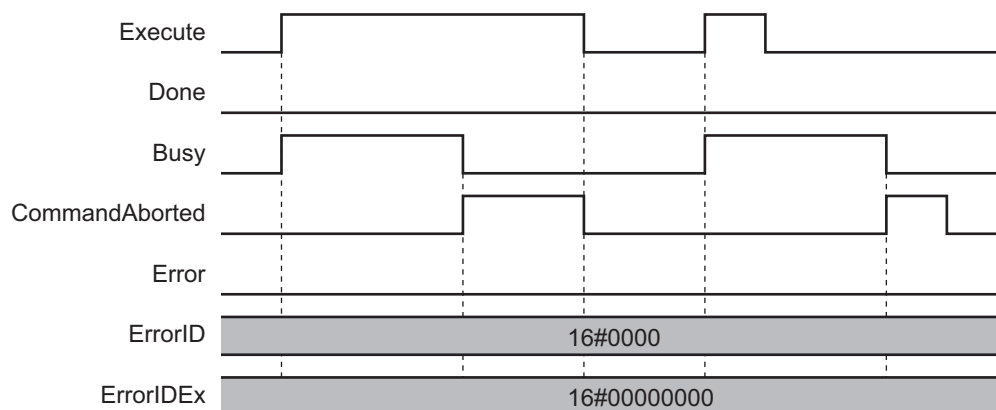
● 正常に完了した場合



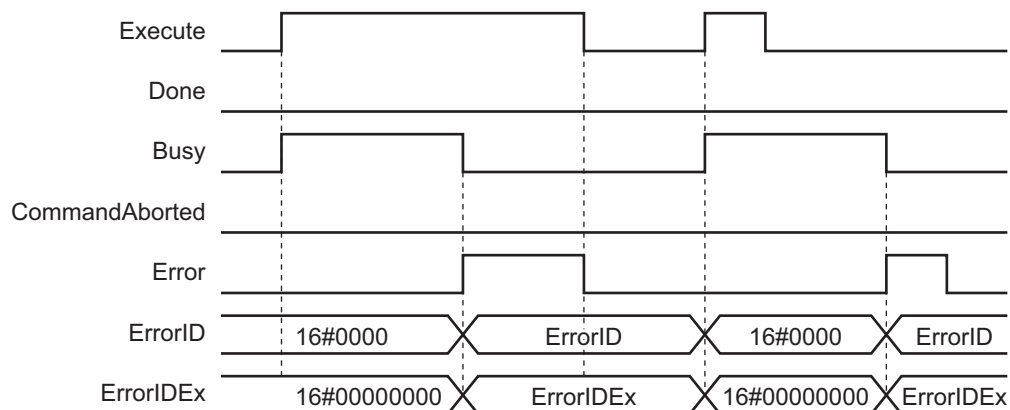
● 実行を中止した場合



● 実行を中断した場合

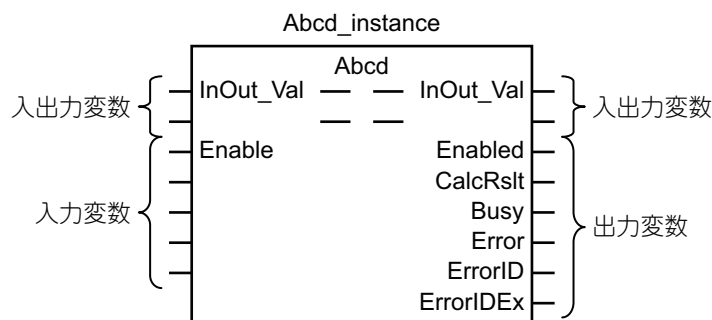


● 異常が発生した場合



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。

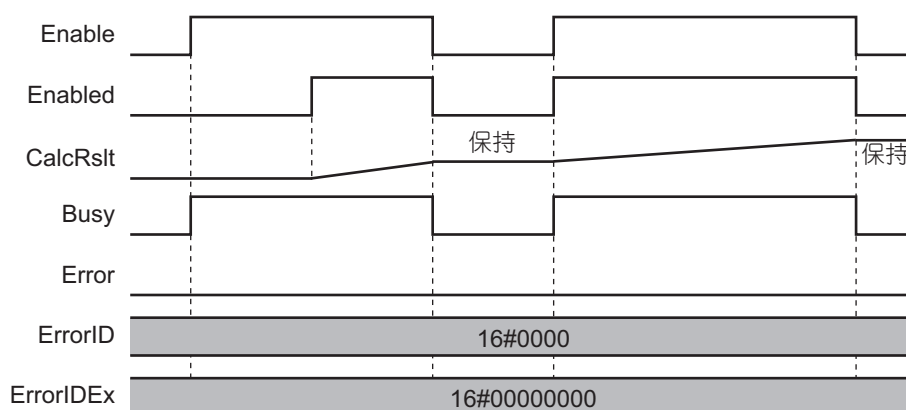


- 異常が発生した場合、ErrorID（エラーコード）、ErrorIDEx（拡張エラーコード）に該当するエラーコード、拡張エラーコードが設定されます。これらは、Error が FALSE になったあとも保持され、Enable が FALSE → TRUE になったときに ErrorID は 16#0000、ErrorIDEx は 16#0000 0000 が設定されます。
- モーション制御や温度制御など制御量を演算する FB では、CalcRslt（演算結果）が不正な値の場合、Enabled が FALSE になります。この時の CalcRslt は使用しないでください。また、FB が正常完了、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。

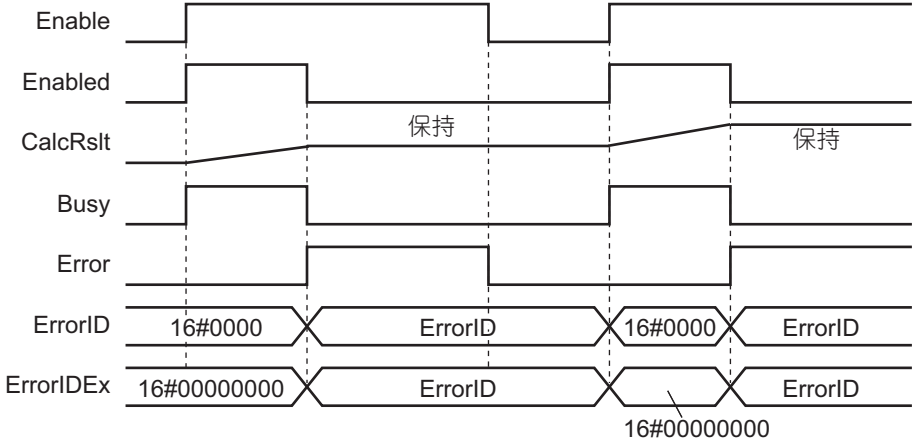
タイミングチャート

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



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。

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、ユーザーズマニュアル ソフトウェア編を参照してください。

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。

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、ユーザーズマニュアル モーション制御編を参照してください。

FB の個別仕様

FB 名	名称	ページ
ReadData_V680	V680 用 RF タグデータ読み出し	P.34
WriteData_V680	V680 用 RF タグデータ書き込み	P.57

FB 共通で使われるデータ型

ここでは、「NX_V680」ライブラリに含まれる FB で共通に使用するデータ型について説明します。

構造体「sV680DeviceVariables」

メンバ	名称	データ型	有効範囲	内容
Status	ステータス	WORD	データ型に従う	RFID ユニットの入力エリアにある I/O マップ「ステータス」から割り当てる。
RefreshCount	更新カウント	UINT	0 ～ 65535	RFID ユニットの入力エリアにある I/O マップ「更新カウント」から割り当てる。
ResponseCode	レスポンスコード	WORD	データ型に従う	RFID ユニットの入力エリアにある I/O マップ「レスポンスコード」から割り当てる。
InputSID	入力 SID	UINT	0 ～ 512	RFID ユニットの入力エリアにある I/O マップ「入力 SID」から割り当てる。
OutputSIDResponse	出力 SID 応答	UINT	0 ～ 512	RFID ユニットの入力エリアにある I/O マップ「出力 SID 応答」から割り当てる。
InputData1	入力データ 1	ARRAY[0..15] OF BYTE	データ型に従う	RFID ユニットの入力エリアにある I/O マップ「入力データ 1」から割り当てる。
UID	UID	ARRAY[0..7] OF BYTE	データ型に従う	RFID ユニットの入力エリアにある I/O マップ「UID」から割り当てる。
RFCommunicationsTime	交信時間	UINT	0 ～ 65535	RFID ユニットの入力エリアにある I/O マップ「交信時間」から割り当てる。
NoiseLevel	ノイズレベル	UINT	0 ～ 100	RFID ユニットの入力エリアにある I/O マップ「ノイズレベル」から割り当てる。
OperationCommand	動作指令	BYTE	データ型に従う	RFID ユニットの出力エリアにある I/O マップ「動作指令」に割り当てる。
RFCommunicationsOption	交信オプション	USINT	データ型に従う	RFID ユニットの出力エリアにある I/O マップ「交信オプション」に割り当てる。
CommandCode	コマンドコード	WORD	データ型に従う	RFID ユニットの出力エリアにある I/O マップ「コマンドコード」に割り当てる。
MemoryAddress	メモリアドレス	UINT	データ型に従う	RFID ユニットの出力エリアにある I/O マップ「メモリアドレス」に割り当てる。
DataSize	データサイズ	UINT	データ型に従う	RFID ユニットの出力エリアにある I/O マップ「データサイズ」に割り当てる。
RefreshCountResponse	更新カウント応答	UINT	0 ～ 65535	RFID ユニットの出力エリアにある I/O マップ「更新カウント応答」に割り当てる。
OutputSID	出力 SID	UINT	0 ～ 512	RFID ユニットの出力エリアにある I/O マップ「出力 SID」に割り当てる。

メンバ	名称	データ型	有効範囲	内容
InputSIDResponse	入力 SID 応答	UINT	0 ~ 512	RFID ユニットの出力エリアにある I/O マップ「入力 SID 応答」に割り当てる。
OutputData1	出力データ 1	ARRAY[0..15] OF BYTE	データ型に従う	RFID ユニットの出力エリアにある I/O マップ「出力データ 1」に割り当てる。
SelectUID	セレクト UID	ARRAY[0..7] OF BYTE	データ型に従う	RFID ユニットの出力エリアにある I/O マップ「セレクト UID」に割り当てる。



使用上の注意

- 構造体「sV680DeviceVariables」で使用する RFID ユニットの I/O 割付設定は以下となります。

RFID ユニットの I/O エントリマッピングの詳細については、『NX シリーズ RFID ユニット ユーザーズマニュアル (SGTG-717)』の 6 章 I/O データ仕様を参照ください。

エリア	データ名	サイズ (Byte)	データ型	割付
入力	Chn ステータス	2	WORD, BOOL	固定
	Chn 更新カウント	2	UINT	固定
	Chn レスポンスコード	2	WORD	固定
	Chn 測定結果	2	UINT	固定
	Chn 入力 SID	2	UINT	固定
	Chn 出力 SID 応答	2	UINT	固定
	Chn 入力データ 1	16	ARRAY[0..15] OF BYTE	固定
	Chn UID	8	ARRAY[0..7] OF BYTE	可変
	Chn 交信時間	2	UINT	可変
	Chn ノイズレベル	2	UINT	可変
出力	Chn 動作指令	1	BYTE, BOOL	固定
	Chn 交信オプション	1	USINT	固定
	Chn コマンドコード	2	WORD	固定
	Chn メモリアドレス	2	UINT	固定
	Chn データサイズ	2	UINT	固定
	Chn 更新カウント応答	2	UINT	固定
	Chn 出力 SID	2	UINT	固定
	Chn 入力 SID 応答	2	UINT	固定
	Chn 出力データ 1	16	ARRAY[0..15] OF BYTE	固定
	Chn セレクト UID	8	ARRAY[0..7] OF BYTE	可変

- 2CH ユニット NX-V680C2 を使用される場合、構造体「sV680DeviceVariables」に 1CH と 2CH の I/O ポートを混合して設定しないでください。FB が正しく動作しなくなります。必ず同じ CH の I/O ポートで揃えてください。

ReadData_V680

RFID ユニット（形 NX-V680）に装着されたアンテナの発信領域内にある RF タグのメモリからデータを読み出します。

プログラム 部品名	名称	FB/ FUN	グラフィック表現	ST 表現
ReadData_V680	V680 用 RF タグデータ読み出し	FB		<pre>ReadData_V680_instance(Execute := <パラメータ>, RFCommunicationsOption := < パラメータ>, ReadKind := <パラメータ>, MemoryAddr := <パラメータ >, DataSize := <パラメータ>, Abort := <パラメータ>, SelectUID := <パラメータ>, Done => <パラメータ>, Busy => <パラメータ>, Result => <パラメータ>, CommandAborted => <パラ メータ>, Error => <パラメータ>, ErrorID => <パラメータ>, ErrorIDEx => <パラメータ>, Warning => <パラメータ>, UID => <パラメータ>, RFCommunicationsTime => <パラメータ>, NoiseLevel => <パラメータ>, ReadData := <パラメータ>, DeviceVariables := <パラメ ータ>,);</pre>

FB/FUN 情報

項目	内容
ライブラリファイル名	OmronLib_NX_V680_Vx_x.slr（x はバージョンを示す）
名前空間	OmronLib\NX_V680
FB/FUN 番号	00213
ソースコード	非公開

入力変数

変数	名称	データ型	内容	有効範囲	単位	初期値
Execute	起動	BOOL	立ち上がり時に処理を開始します。 TRUE：実行する FALSE：実行しない	TRUE,FALSE	—	FALSE
RFCommunicationsOption	交信オプション	USINT	交信時の動作シーケンスを指定します。 0：トリガ 1：オート 2：リピート 3：FIFO トリガ 4：FIFO リピート 5：マルチトリガ 6：マルチリピート 7：セレクトティブ	0 ～ 7	—	0
ReadKind	リード種別	USINT	リードコマンドの種別を指定します。 ^{*1} 0：通常 1：誤り検出付 2：誤り訂正付	0 ～ 2	—	0
MemoryAddr	メモリアドレス	UINT	RF タグからデータを読み出すメモリの先頭アドレスを指定します。 ^{*2}	0 ～ 65535	Byte	0
DataSize	データサイズ	UINT	RF タグからデータを読み出すサイズを入力します。 ^{*3}	1 ～ 8192	Byte	0
Abort	中断	BOOL	立ち上がり時に処理を中断します。	TRUE,FALSE	—	FALSE
SelectUID	セレクト UID	ARRAY[0..7]OF BYTE	交信オプション「セレクトティブ」時に、交信対象の RF タグの UID を指定します。	データ型に合う	—	16#00

*1. リードコマンドの種類と動作の違いについては、『NX シリーズ RFID ユニット ユーザーズマニュアル (SGTG-717)』のコマンド一覧を参照ください。

*2. 実際に使用する RF タグのメモリマップを考慮して入力してください。

*3. 実際に使用する RF タグのメモリ容量を考慮して入力してください。0 のときは何もせずに終了します。

出力変数

変数	名称	データ型	内容	有効範囲	単位	初期値
Done	完了	BOOL	処理が完了したときに TRUE になります。	TRUE,FALSE	—	FALSE
Busy	実行中	BOOL	処理を受け付けたときに TRUE になります。	TRUE,FALSE	—	FALSE
Result	結果	BOOL	交信オプションが「リピート」、「FIFO リピート」、「マルチトリガ」、「マルチリピート」の場合、RF タグと交信した結果を出力するごとに、値を TRUE と FALSE と交互に変化します。	TRUE,FALSE	—	FALSE
Command Aborted	実行中断	BOOL	処理が中断されたときに TRUE になります。	TRUE,FALSE	—	FALSE
Error	エラー	BOOL	エラーが発生したときに TRUE になります。 TRUE: 異常終了 FALSE: 正常終了、または実行中	TRUE,FALSE	—	FALSE
ErrorID	エラーコード*	WORD	エラーコードが出力されます。 異常終了時の異常 ID 正常終了時は 16#0	*1	—	0
ErrorIDEx	拡張エラーコード*	DWORD	拡張エラーコードが出力されます。 異常終了時の異常 ID 正常終了時は 16#0	*1	—	0
Warning	ワーニング	BOOL	誤り訂正が発生したときに TRUE になります。	TRUE,FALSE	—	FALSE
UID	UID	ARRAY[0..7] OF BYTE	交信した RF タグの UID が出力されます	データ型に従う	—	16#00
RFCommunicationsTime	交信時間	UINT	計測した交信時間が出力されます。	0 ~ 65535	ms	0
NoiseLevel	ノイズレベル	UINT	測定したノイズレベルが出力されます。	0 ~ 99	—	0

*1. 詳細は、「トラブルシューティング (P.50)」を参照してください。

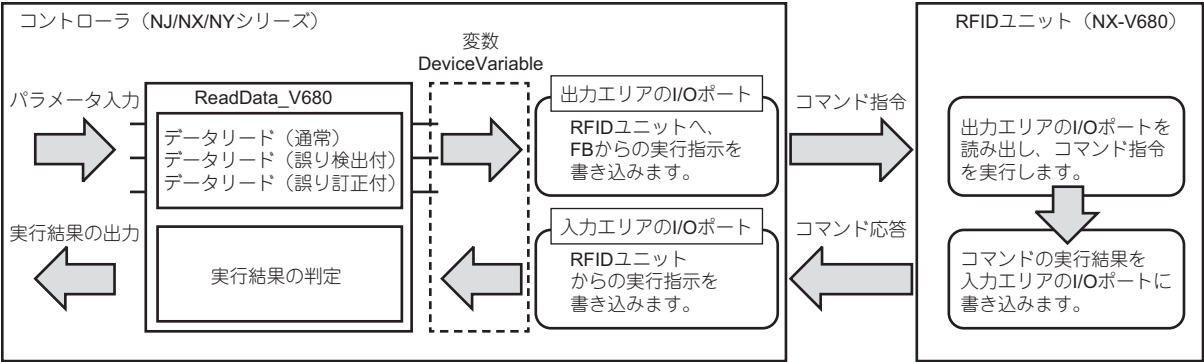
入出力変数

変数	名称	データ型	内容	有効範囲	単位	初期値
DeviceVariables	デバイス変数	OmronLib\RFID\sNX-V680DeviceVariables	RFID ユニットのデバイス変数（入力エリア / 出力エリアの I/O ポート）	データ型に従う	—	—
ReadData	読み出しデータ	ARRAY [可変長] OF BYTE	RF タグから読み出したデータ配列	データ型に従う	—	—

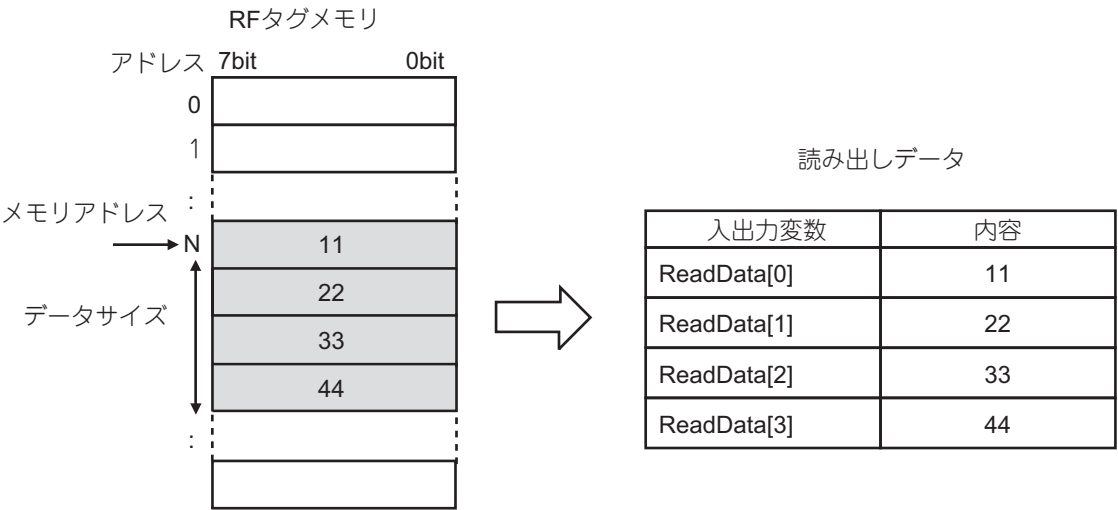
（注）任意長の BYTE 型配列が指定可能です。ただし、配列長さは DataSize 以上でなければなりません。配列先頭番号は 0 始まりでも n 始まりでも構いません。

機能

本 FB は、NJ/NX/NY シリーズコントローラとデータ交換している RFID ユニット（形 NX-V680）に対して、RF タグのメモリを読み出すコマンド制御を行います。そのため、制御したい RFID ユニットのデバイス変数（出力エリアの I/O ポートならびに入力エリアの I/O ポート）を FB の入出力変数「DeviceVariables」に設定する必要があります。



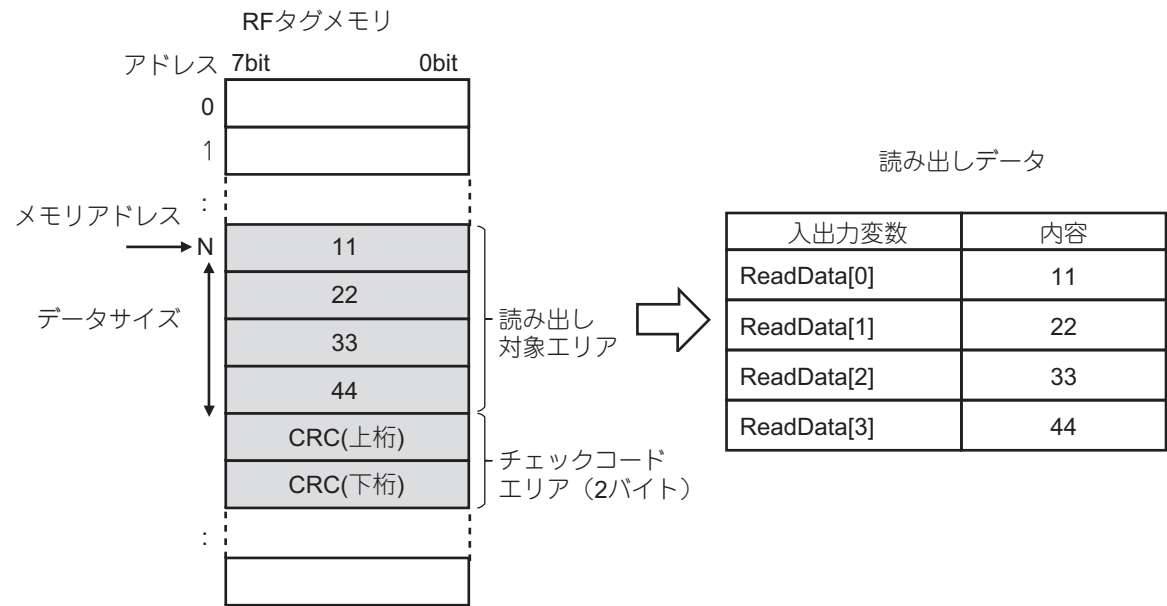
- Execute（起動）の立ち上がりで、RF タグのメモリから読み出したデータを、ReadData（読み出しデータ）で指定したバイト配列に格納します。



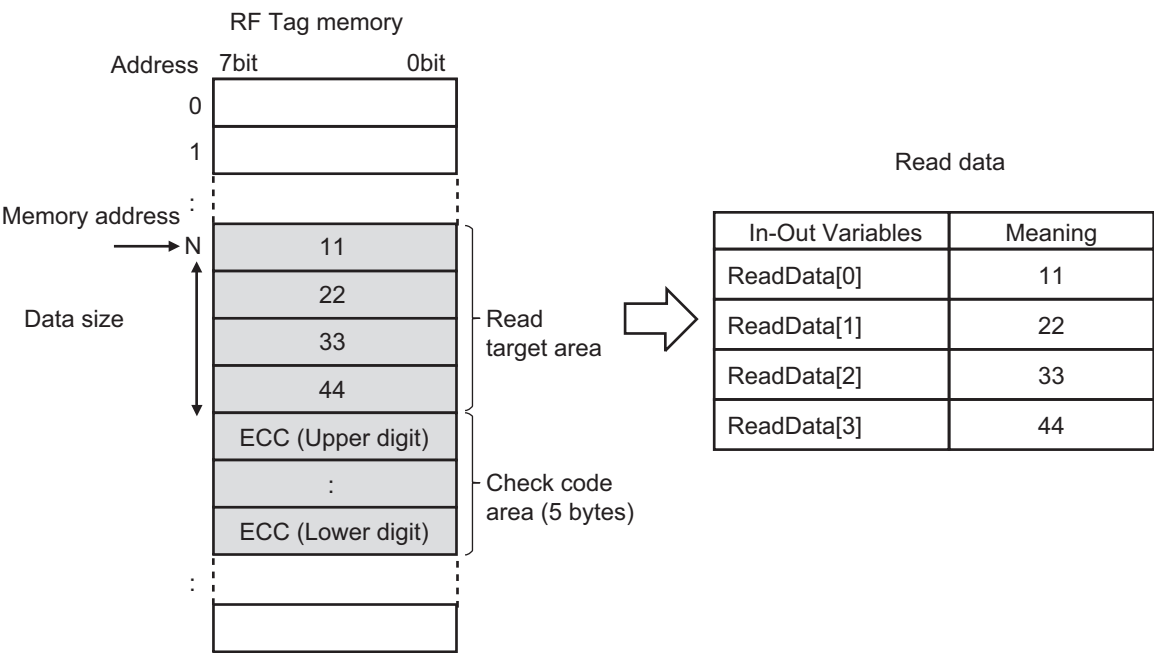
- 1 回の実行で RF タグから読み出せるメモリ容量は、指定する ReadKind（リード種別）によって異なります。MemoryAddr（メモリアドレス）と DataSize（データサイズ）で指定可能な範囲を以下に示します。

ReadKind	MemoryAddr	DataSize
0：通常	0 ～ 65535	1 ～ 8192
1：誤り検出付	0 ～ 65533	1 ～ 8190
2：誤り訂正付	0 ～ 65530	1 ～ 510

- ReadKind（リード種別）を " 誤り検出付 " で実行すると、RF タグからデータとチェックコードを読み出し、データの誤り検出を行います。本機能を使用する際は、ファンクションブロック「WriteData_V680」の " 誤り検出付 " で予め対象のエリアにデータとチェックコードを書き込んでおいてください。



- ReadKind（リード種別）を " 誤り訂正付 " で実行すると、RF タグからデータとチェックコードを読み出し、データの誤り検出と 1 ビットの誤り訂正を行います。本機能を使用する際は、ファンクションブロック「WriteData_V680」の " 誤り訂正付 " で予め対象のエリアにデータとチェックコードを書き込んでおいてください。



(注) ReadKind（リード種別）を " 誤り検出付 " で実行するとき、ファンクションブロック「WriteData_V680」の " 誤り検出付 "、" 誤り訂正付 " で予め対象のエリアにデータとチェックコードを書き込んでいない場合、交信エラー（RF タグデータ誤り検出）が発生します。

タイミングチャート

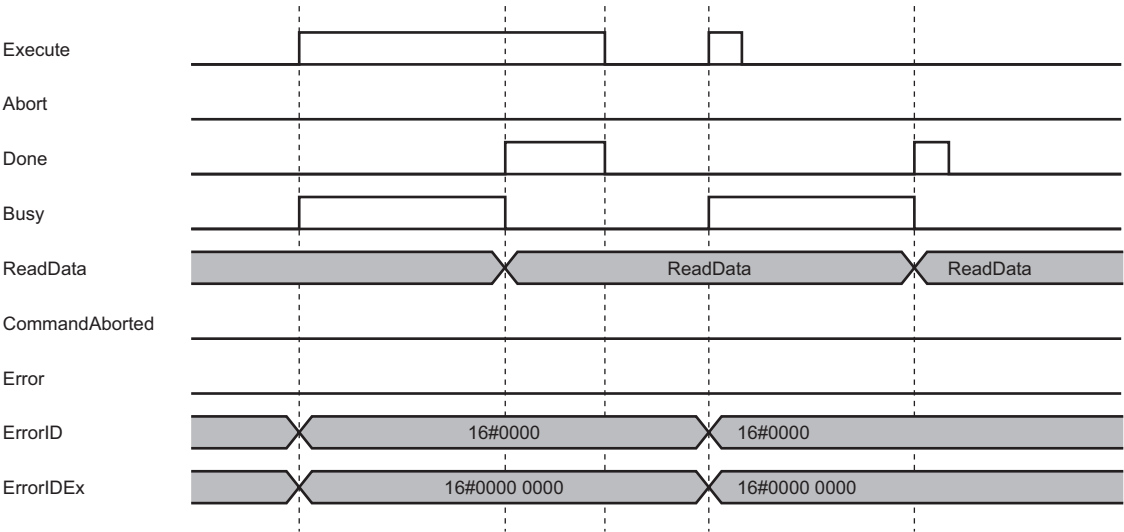
タイミングチャートを以下に示します。

1 個の RF タグと交信する（交信オプション：トリガ、オート、FIFO トリガ、セレクトティブ）

本 FB の RFCommunicationsOption（交信オプション）にトリガ / オート / FIFO トリガ / セレクトティブを指定して起動した場合、アンテナの交信領域にある 1 個の RF タグと交信した結果が出力されます。

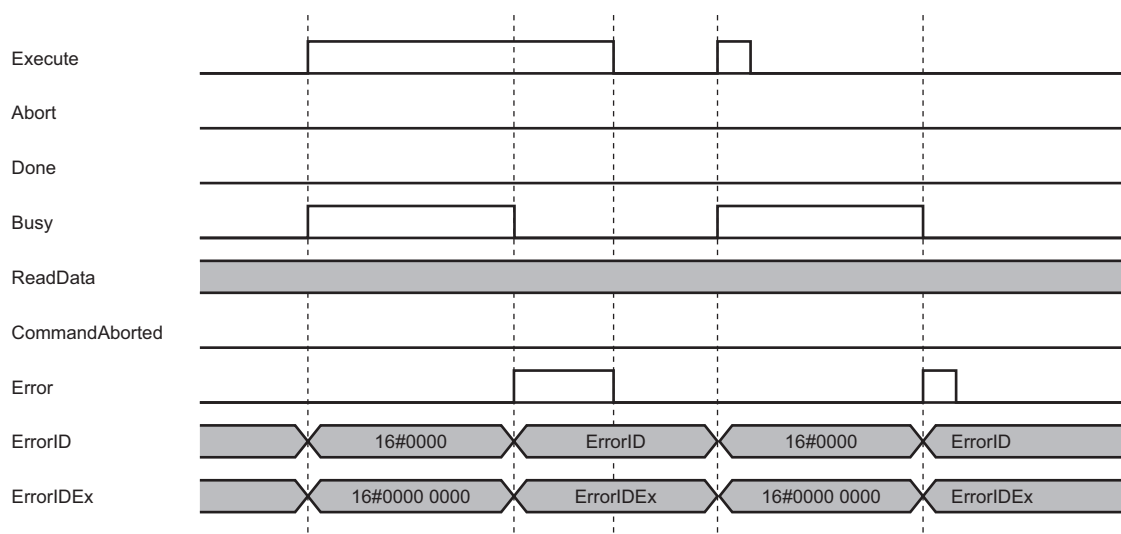
● 正常終了時のタイミングチャート

- Execute（起動）の立ち上がりと同時に Busy（実行中）が TRUE となります。
- RF タグのメモリからデータ読み出しが正常終了すると、Done（完了）が TRUE となり、Busy（実行中）が FALSE となります。同時に ReadData（読み出しデータ）や UID にデータ列、RFCommunicationsTime（交信時間）や NoiseLevel（ノイズレベル）に値が出力されます。
- 本 FB の実行完了後も Execute（起動）が TRUE である場合は、Done（完了）の出力は保持されます。
- 本 FB の実行完了前に Execute（起動）が FALSE となった場合は、Done（完了）は実行完了後の 1 タスク周期だけ TRUE となります。



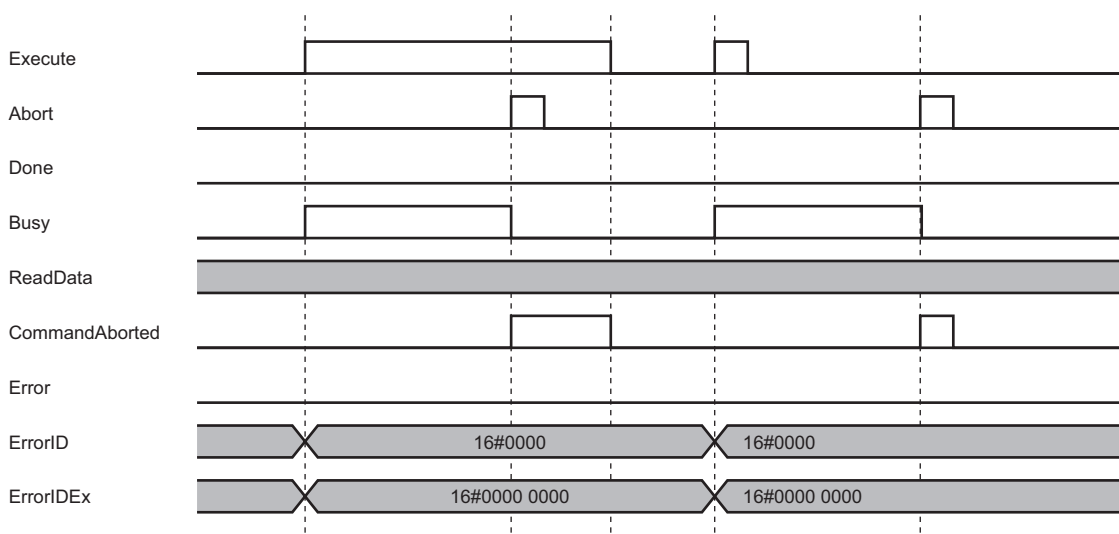
● 異常終了時のタイミングチャート

- 本 FB 実行中に異常が発生した場合、Error（エラー）が TRUE となり、Busy（実行中）が FALSE となります。
- ErrorID（エラーコード）、ErrorIDEx（拡張エラーコード）に出力された値を参照することによって、異常の原因を知ることができます。
- 本 FB の実行完了後も Execute（起動）が TRUE である場合は、Error（エラー）の出力は保持されます。
- ErrorID（エラーコード）や ErrorIDEx（拡張エラーコード）の出力は本 FB を再実行するまで保持されます。
- 本 FB の実行完了前に Execute（起動）が FALSE となった場合は、Error（エラー）は実行完了後の 1 タスク周期だけ TRUE となります。



● 中断時のタイミングチャート

本 FB 実行中に RF タグと通信を中断するには、Abort（中断）を TRUE にします。Abort（中断）が TRUE になると、Busy（実行中）が FALSE となり、CommandAborted（実行中断）が TRUE となります。

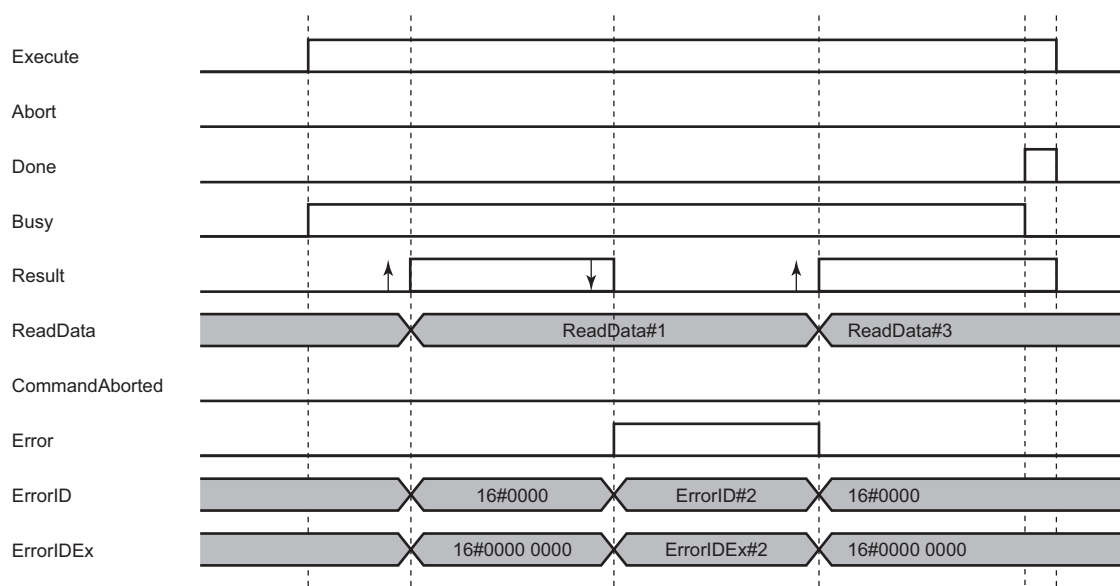


複数の RF タグと交信する（交信オプション：マルチトリガ）

本 FB の RFCommunicationsOption（交信オプション）にマルチトリガを指定して起動した場合、アンテナの交信領域内にある複数の RF タグと交信した結果が出力されます。

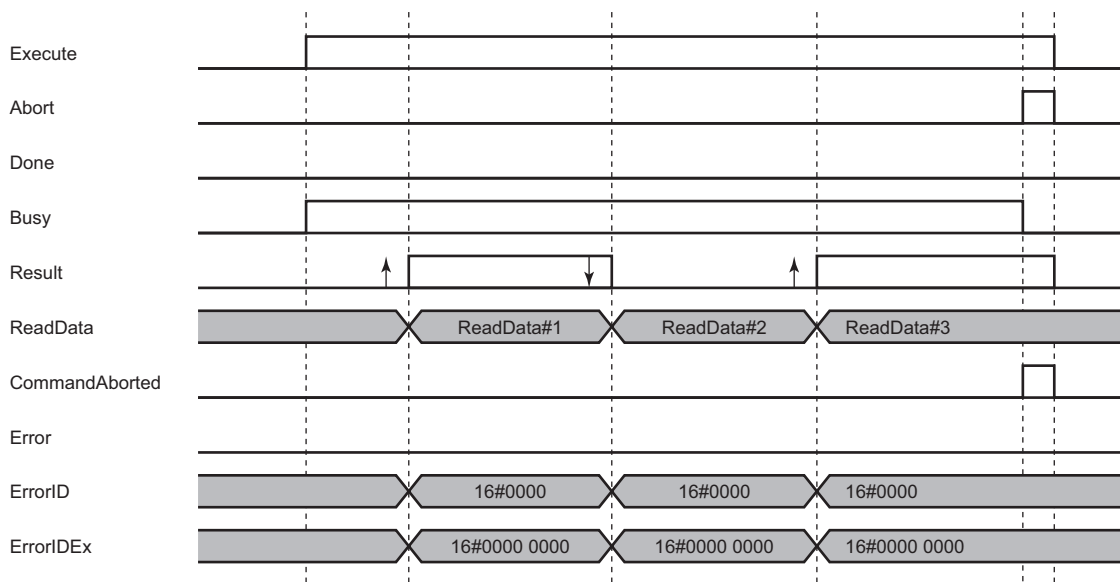
● 複数の RF タグと交信時のタイミングチャート

- Execute（起動）の立ち上がりと同時に Busy（実行中）が TRUE となります。
- 複数の RF タグとの交信毎に、Result（結果）が FALSE → TRUE → FALSE → TRUE とトグル動作します。
- 交信領域内にある全ての RF タグと交信が完了すると、Done（完了）が TRUE となり、Busy（実行中）が FALSE となります。
- RF タグのメモリからデータ読み出しが正常終了すると、Result（結果）の変化と同時に、ReadData（読み出しデータ）や UID にデータ列、RFCommunicationsTime（交信時間）や NoiseLevel（ノイズレベル）に値が出力されます。
- RF タグのメモリからデータ読み出しが異常終了すると、Result（結果）の変化と同時に、Error（エラー）が TRUE となります。ErrorID（エラーコード）、ErrorIDEx（拡張エラーコード）に出力された値を参照することによって、異常の原因を知ることができます。
- Done（完了）の立ち下がりと同時に Result（結果）が FALSE となります。



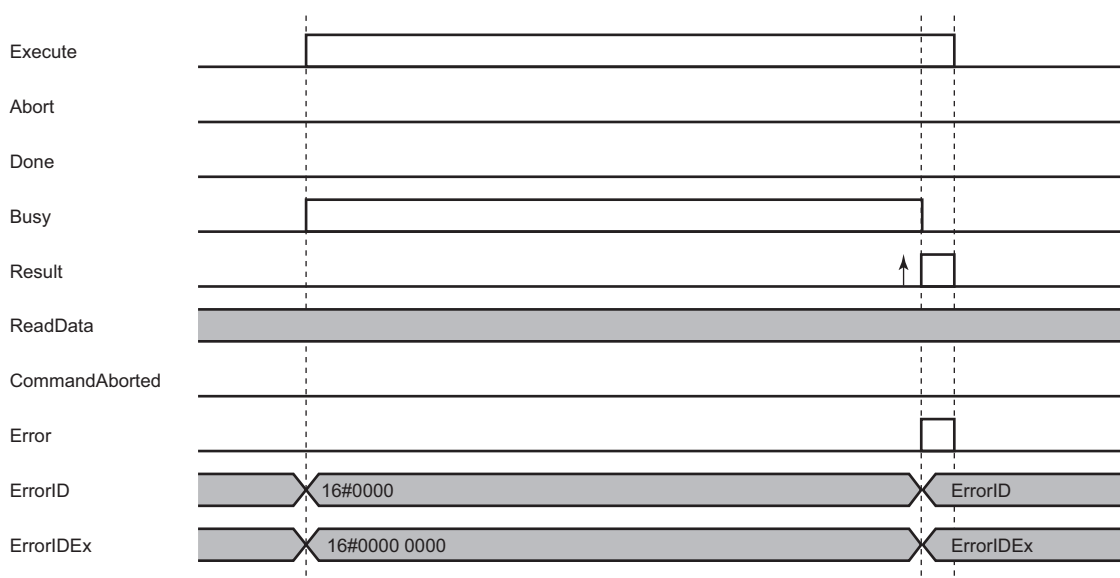
● 中断時のタイミングチャート

- 本 FB 実行中に RF タグと通信を中断するには、Abort（中断）を TRUE にします。Abort（中断）が TRUE になると、Busy（実行中）が FALSE となり、CommandAborted（実行中断）が TRUE となります。
- CommandAborted（実行中断）の立ち上がりと同時に Result（結果）が FALSE となります。



● タグ不在時のタイミングチャート

- 本 FB 実行中に一度も RF タグと通信が行われなかった場合、Result（結果）と Error（エラー）が TRUE となり、Busy（実行中）が FALSE となります。
- ErrorID（エラーコード）、ErrorIDEx（拡張エラーコード）に出力された値を参照することによって、異常の原因を知ることができます。
- 本 FB の実行完了後も Execute（起動）が TRUE である場合は、Result（結果）と Error（エラー）の出力は保持されます。
- ErrorID（エラーコード）や ErrorIDEx（拡張エラーコード）の出力は本 FB を再実行するまで保持されます。
- 本 FB の実行完了前に Execute（起動）が FALSE となった場合は、Result（結果）と Error（エラー）は実行完了後の 1 タスク周期だけ TRUE となります。

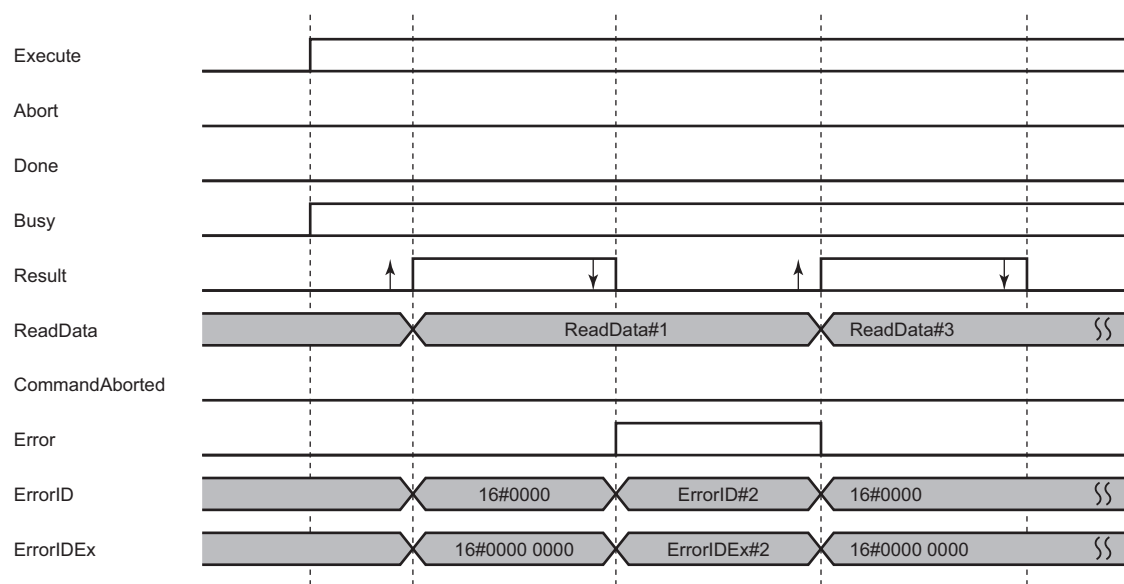


RF タグと繰り返し通信する（通信オプション：リピート、FIFO リピート、マルチリピート）

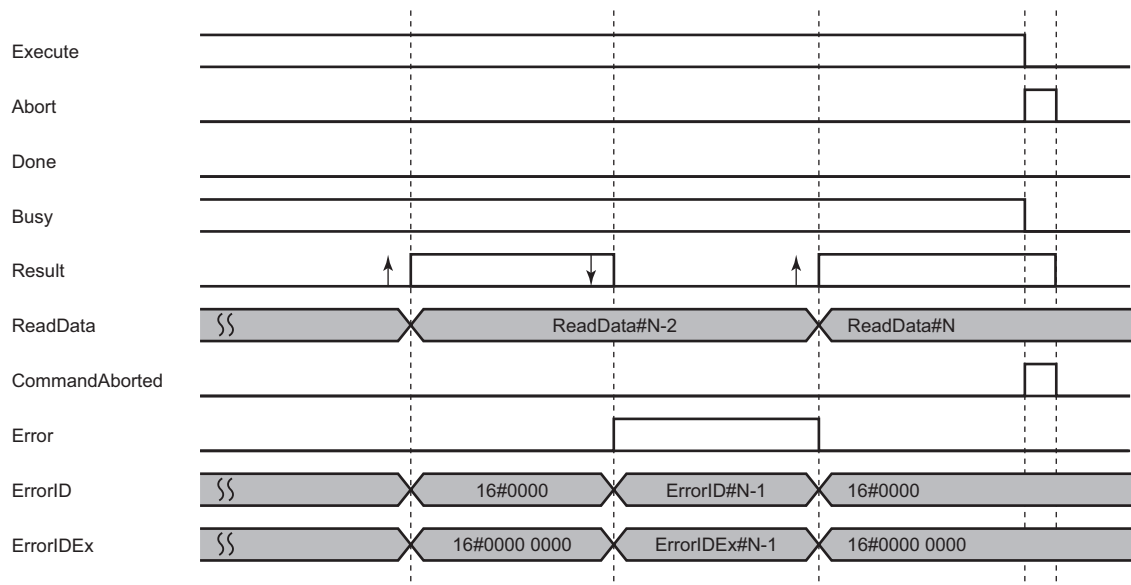
本 FB の RFCommunicationsOption（通信オプション）をリピート / FIFO リピート / マルチリピートを指定して起動した場合、アンテナの通信領域内に移動してくる RF タグと通信を繰り返した結果が出力されます。

● 繰り返し通信時のタイミングチャート

- Execute（起動）の立ち上がりと同時に Busy（実行中）が TRUE となります。
- 複数の RF タグとの通信毎に、Result（結果）が FALSE → TRUE → FALSE → TRUE とトグル動作します。
- RF タグのメモリからデータ読み出しが正常終了すると、Result（結果）の変化と同時に、ReadData（読み出しデータ）や UID にデータ列、RFCommunicationsTime（通信時間）や NoiseLevel（ノイズレベル）に値が出力されます。
- RF タグのメモリからデータ読み出しが異常終了すると、Result（結果）の変化と同時に、Error（エラー）が TRUE となります。ErrorID（エラーコード）、ErrorIDEx（拡張エラーコード）に出力された値を参照することによって、異常の原因を知ることができます。

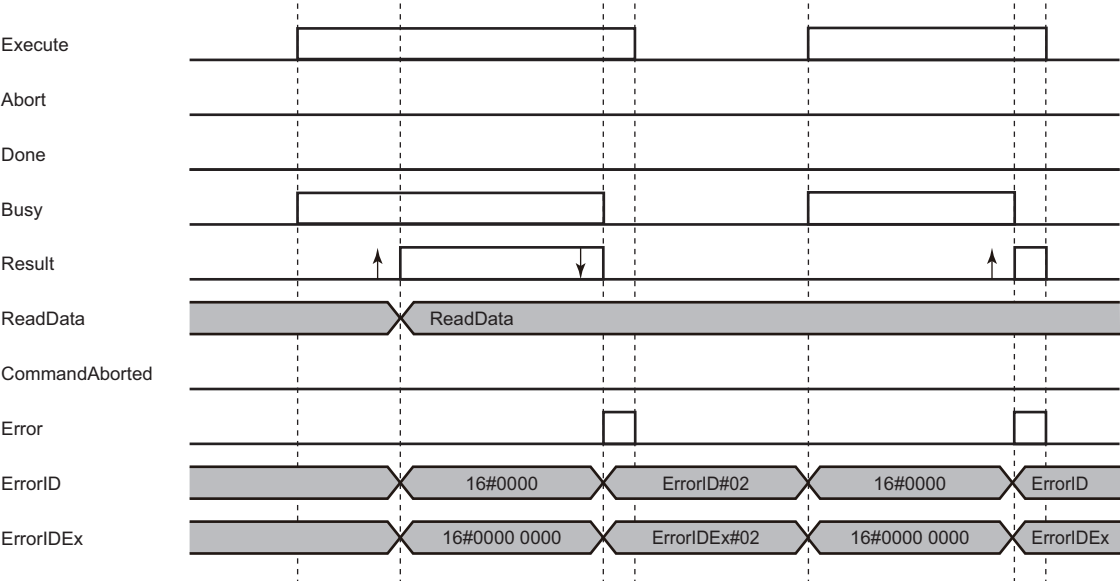


- RF タグと繰り返し通信を終了するには、Abort（中断）を TRUE にする必要があります。
Abort（中断）が TRUE になると、Busy（実行中）が FALSE となり、CommandAborted（実行中断）が TRUE となります。
- CommandAborted（実行中断）の立ち上がりと同時に Result（結果）が FALSE となります。



● 異常終了時のタイミングチャート

- 本 FB 実行中に特定の異常が発生した場合、Error（エラー）が TRUE、Busy（実行中）が FALSE となり、繰り返しを中断します。
- Error（エラー）の立ち上がりと同時に Result（結果）が FALSE となります。
- ErrorID（エラーコード）、ErrorIDEx（拡張エラーコード）に出力された値を参照することによって、異常の原因を知ることができます。
- 本 FB の実行完了後も Execute（起動）が TRUE である場合は、Error（エラー）の出力は保持されます。
- ErrorID（エラーコード）や ErrorIDEx（拡張エラーコード）の出力は本 FB を再実行するまで保持されます。
- 本 FB の実行完了前に Execute（起動）が FALSE となった場合は、Error（エラー）は実行完了後の 1 タスク周期だけ TRUE となります。



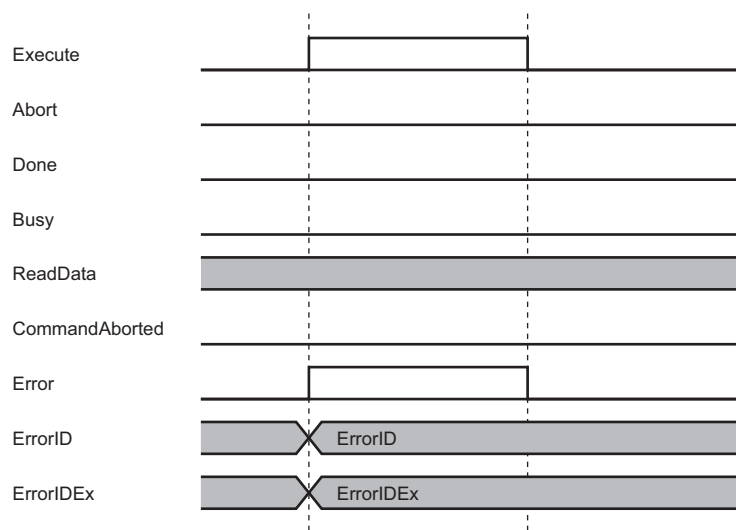
(注) 繰り返し通信を停止させる特定の異常は以下の通りとなります。

エラーコード	拡張エラーコード	状態
16#3D13	16#6AA00000	アンテナ構成異常
	16#6AA10000	アンプ電源異常
	16#6AA20000	アンプ断線検知
	16#6A720000	RF タグ不在エラー

共通の振る舞い

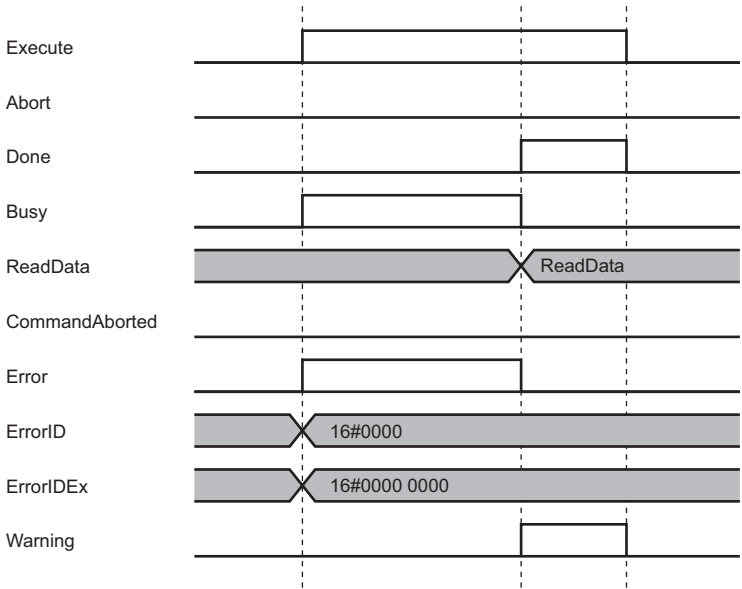
● 起動時に異常を検出した時のタイミングチャート

- 本 FB 起動時に異常が検知された場合、Error（エラー）が TRUE となります。Busy（実行中）は FALSE のままです。
- ErrorID（エラーコード）、ErrorIDEx（拡張エラーコード）に出力された値を参照することによって、異常の原因を知ることができます。
- 本 FB の実行完了後も Execute（起動）が TRUE である場合は、Error（エラー）の出力は保持されます。
- ErrorID（エラーコード）や ErrorIDEx（拡張エラーコード）の出力は本 FB を再実行するまで保持されます。

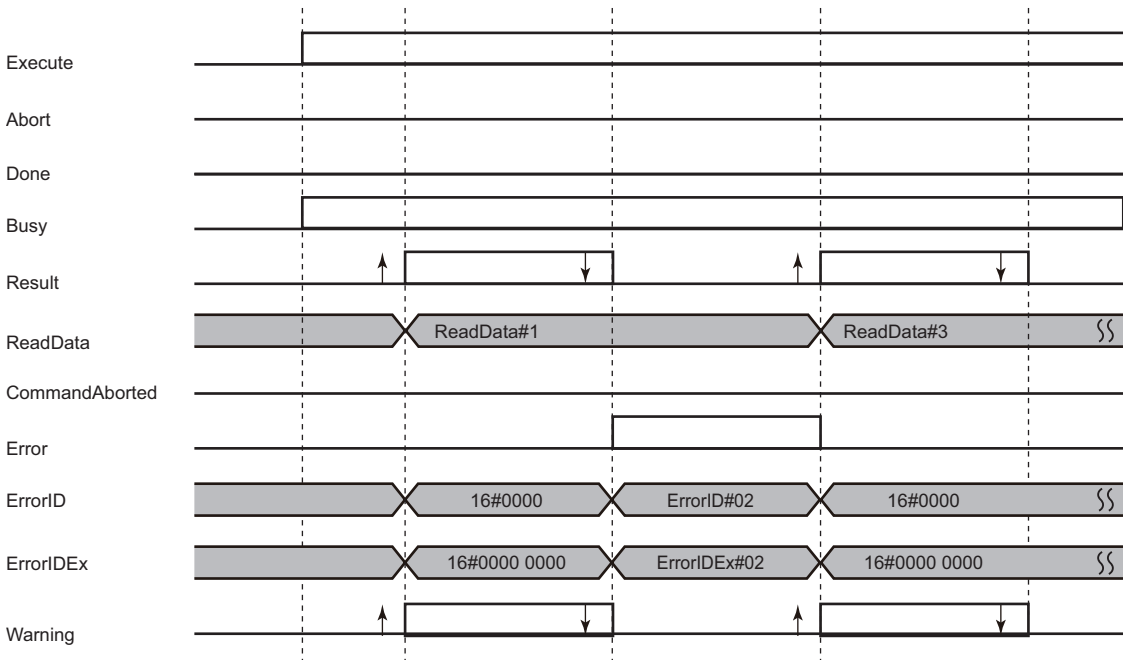


● 1 ビットの誤訂正を行って正常終了した時のタイミングチャート（リード種別：誤り訂正付）

- RF タグのメモリから読み出したデータで、1 ビットの誤りを検出し訂正を行ったとき、Done（完了）が TRUE となり、同時に Warning（ワーニング）が TRUE となります。
- 本 FB の実行完了後も Execute（起動）が TRUE である場合は、Warning（ワーニング）の出力は保持されます。



- 繰り返し通信にて、RF タグのメモリから読み出したデータで、1 ビットの誤りを検出し訂正を行ったとき、Result（結果）がトグル動作するタイミングで、同時に Warning（ワーニング）も変動します。
- 続けて誤り訂正が発生した場合、Warning（ワーニング）は TRUE のまま変わりません。



使用上の注意

- RFID ユニットが既にコマンド実行中でビジー状態の場合は、異常終了として Error（異常）を TRUE で出力し、本 FB の機能を実行しません。
- RFID ユニット（形 NX-V680C2）を使用される際、本 FB への入出力変数として使用するデバイス変数はチャンネル Ch1 か Ch2 のいずれかに統一してください。各チャンネルのデバイス変数が混在すると正しく動作できません。
- 本 FB を実行する前に、使用する形 NX-V680C□ のマニュアルをよく読み、安全を確認のうえで使用してください。
- RFID ユニットの動作パラメータ " データ格納順序 " が " 降順 " の時、奇数バイトのデータサイズを指定しないでください。正しいデータが読み出せません。
- RFID ユニットの使用されるチャンネルに、I/O エントリ " 入力データ 2 ～ 8 " は追加しないでください。入力データの合計サイズが増えると正しく動作できません。

トラブルシューティング

本 FB が異常終了した時に出力するエラーコードの一覧を以下に示します。

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#3D13	16#00000001	交信オプション不正	RFCommunicationsOption の値が有効範囲外です。	RFCommunicationsOption に正しい値を設定してください。
	16#00000002	リード種別不正	ReadKind の値が有効範囲外です。	ReadKind に正しい値を設定してください。
	16#00000003	メモリアドレス不正	MemoryAddr の値が有効範囲外です。	MemoryAddr に正しい値を設定してください。
	16#00000004	データサイズ不正	DataSize の値が有効範囲外です。	DataSize に正しい値を設定してください。
	16#00000005	データ配列不正	ReadData の配列長さが DataSize 未満です。	ReadData の配列長さを DataSize 以上にしてください。
	16#00000006	起動不可	RFID ユニットがコマンド実行可能状態でないため本 FB を起動できません。	RFID ユニットの状態が、コマンドが実行可能であることを確認のうえ、再度 FB を実行してください。
	16#6AA00000	アンテナ構成異常	サポート外のアンテナが接続されています。 ※ 形 NX-V680C2 のユニットと形 V680-H01-V2 の組み合わせ	形 NX-V680C2 には形 V680-H01-V2 以外のアンテナを接続してください。または、形 NX-V680C1 を使用してください。
	16#6AA10000	アンプ電源異常	アンプを駆動する電源が供給されていません。	I/O 電源の入力を確認してください。
	16#6AA20000	アンプ断線検知	アンプの断線を検出しました。認識できませんでした。	アンプを接続してください。または、アンプを交換してください。
	16#6A720000	RF タグ不在エラー	RF タグが交信領域内に存在しません。	RF タグが交信領域内に入るように設備を調整してください。
	16#6A700000	RF タグ交信失敗	RF タグと交信中にエラーが発生し正常終了できませんでした。	RF タグの移動速度、RF タグ間の距離を仕様範囲内になるように変更してください。 また、ノイズが多い環境ではノイズ対策を実施してください。
	16#6A760000	RF タグデータ誤り	RF タグのデータ誤りを検出しました。	データチェック対象範囲内のデータを確認し、想定外の書き込みを実施しなかったかを確認してください。
	16#6A7A0000	RF タグアドレスエラー	RF タグのアドレスが間違っています。	使用している RF タグのメモリ容量を確認し、指定するアドレスをメモリ容量の範囲内になるように修正してください。
	16#6A790000	RF タグ応答エラー	RF タグがエラー応答し正常終了できませんでした。	ノイズの多い環境ではノイズ対策を実施してください。 ^{*1}
	16#6A7F0000	RF タグカスタムコードエラー	使用できない RF タグと交信しました。	RF タグを変更してください。
	16#6AC00000	未定義コマンド	未定義のコマンドのため実行できません。	DeviceVariables に正しい変数を設定してください。
	16#6AC10000	コマンドパラメータ不正	コマンドのパラメータが間違っているため実行できません。	DeviceVariables に正しい変数を設定してください。
	16#6AC20000	コマンド実行不可	コマンドの実行条件が成立していないためコマンドが実行できません。	コマンドに対して、実行可能な交信オプションに修正してください。

*1. ノイズ対策で解決しない場合は RF タグを変更してください。

サンプルプログラム

プログラム説明

RF タグのメモリアドレス 8 から 16 バイトのデータを読み出します。

前提条件

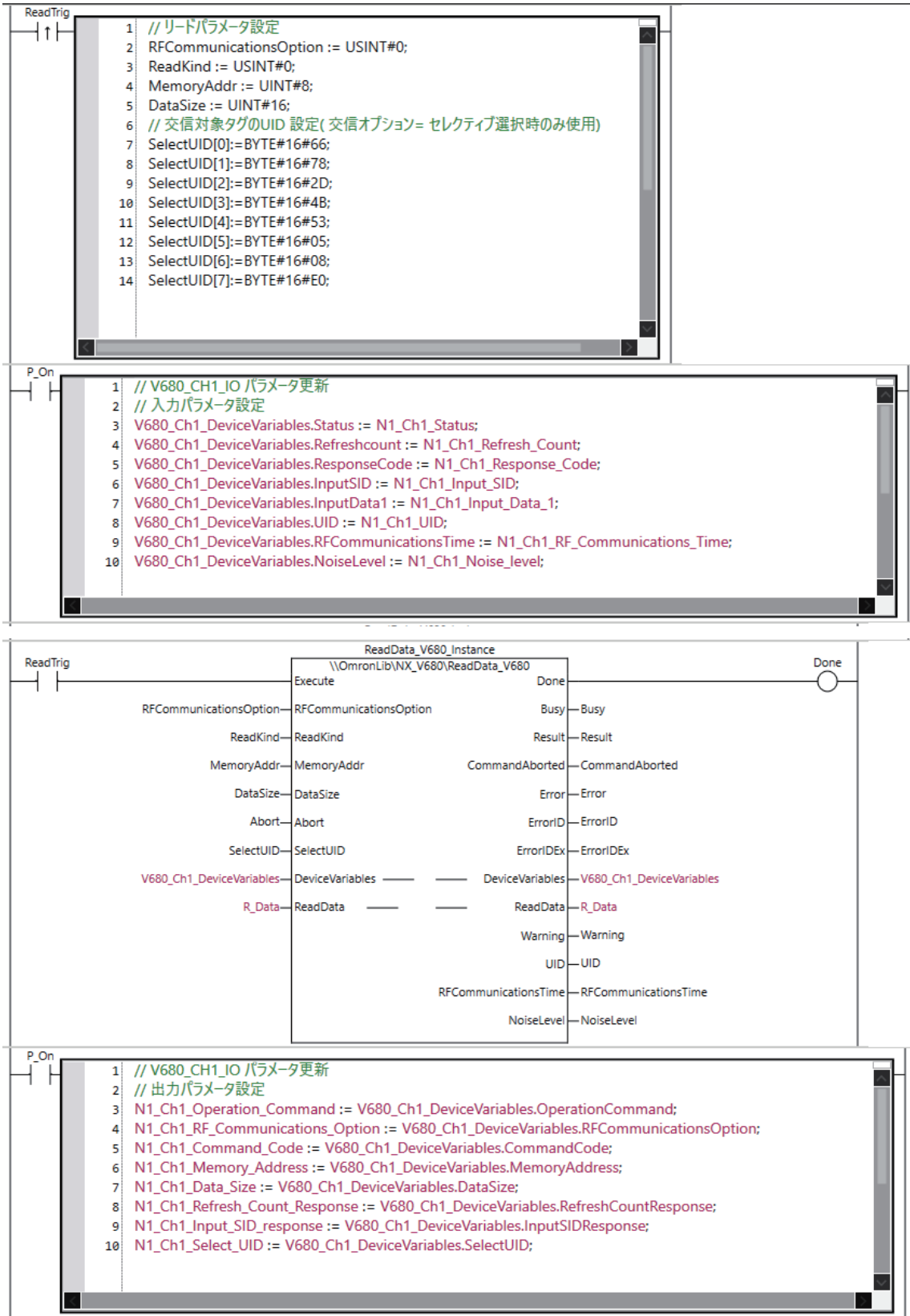
動作させたい RFID ユニットのデバイス変数を生成し、ユーザプログラム内で外部参照してください。デバイス変数生成方法の詳細は、□『Sysmac Studio Version1 オペレーションマニュアル (SBCA-470)』を参照してください。

主な変数

変数	名称	データ型	初期値	内容
ReadData_V680_Instance	リードデータ V680 FB	OmronLib\NX_V680\ReadData_V680	-	NX-V680 のリードを実施する為の FB
V680_Ch1_Device Variables	Ch1 デバイス変数	OmronLib\NX_V680\sV680DeviceVariables	-	Ch1 デバイス変数の構造体
R_Data	リードデータ用バッファ	ARRAY[0..8191] OF BYTE	16#00	"ReadData_V680_Instance" の出力用変数
ReadTrig	リード実行フラグ	BOOL	FALSE	リード実行時に TRUE
RFCommunicationsOption	交信オプション	USINT	0	交信オプションを設定
ReadKind	リード種別	USINT	0	リード種別を設定
MemoryAddr	メモリアドレス	UINT	0	交信開始アドレスを設定
DataSize	データサイズ	UINT	0	交信データサイズを設定
Abort	実行停止フラグ	BOOL	FALSE	リード実行停止時に TRUE
SelectUID	セレクト UID	ARRAY[0..7] OF BYTE	16#00	交信対象タグの UID を設定
CommandStage	コマンド状態	UINT	0	ST プログラムの実行状態
Execute	実行フラグ	BOOL	FALSE	ST プログラム内の実行状態フラグ
Done	実行完了フラグ	BOOL	-	"ReadData_V680_Instance" の出力用変数
Busy	実行中フラグ	BOOL	-	"ReadData_V680_Instance" の出力用変数
Result	結果フラグ	BOOL	-	"ReadData_V680_Instance" の出力用変数
CommandAborted	実行中断フラグ	BOOL	-	"ReadData_V680_Instance" の出力用変数

変数	名称	データ型	初期値	内容
Error	エラーフラグ	BOOL	-	"ReadData_V680_Instance" の出力用変数
Warning	ワーニングフラグ	BOOL	-	"ReadData_V680_Instance" の出力用変数
ErrorID	エラーコード	WORD	-	"ReadData_V680_Instance" の出力用変数
ErrorIDEx	拡張エラーコード	DWORD	-	"ReadData_V680_Instance" の出力用変数
RFCommunicationsTime	交信時間	UINT	-	"ReadData_V680_Instance" の出力用変数
NoiseLevel	ノイズレベル	UINT	-	"ReadData_V680_Instance" の出力用変数
UID	UID	ARRAY[0..7] OF BYTE	-	"ReadData_V680_Instance" の出力用変数

ラダー図



● インライン ST のコード（ラダー図 0 行め）：

```
// リードパラメータ設定
RFCommunicationsOption := USINT#0;
ReadKind := USINT#0;
MemoryAddr := UINT#8;
DataSize := UINT#16;
// 通信対象タグの UID 設定（通信オプション＝セレクトティブ選択時のみ使用）
SelectUID[0] := BYTE#16#66;
SelectUID[1] := BYTE#16#78;
SelectUID[2] := BYTE#16#2D;
SelectUID[3] := BYTE#16#4B;
SelectUID[4] := BYTE#16#53;
SelectUID[5] := BYTE#16#05;
SelectUID[6] := BYTE#16#08;
SelectUID[7] := BYTE#16#E0;
```

● インライン ST のコード（ラダー図 1 行め）：

```
// V680_CH1_I0 パラメータ更新
// 入力パラメータ設定
V680_Ch1_DeviceVariables.Status := N1_Ch1_Status;
V680_Ch1_DeviceVariables.Refreshcount := N1_Ch1_Refresh_Count;
V680_Ch1_DeviceVariables.ResponseCode := N1_Ch1_Response_Code;
V680_Ch1_DeviceVariables.InputSID := N1_Ch1_Input_SID;
V680_Ch1_DeviceVariables.InputData1 := N1_Ch1_Input_Data_1;
V680_Ch1_DeviceVariables.UID := N1_Ch1_UID;
V680_Ch1_DeviceVariables.RFCommunicationsTime := N1_Ch1_RF_Communications_Time;
V680_Ch1_DeviceVariables.NoiseLevel := N1_Ch1_Noise_level;
```

● インライン ST のコード（ラダー図 3 行め）：

```
// V680_CH1_I0 パラメータ更新
// 出力パラメータ設定
N1_Ch1_Operation_Command := V680_Ch1_DeviceVariables.OperationCommand;
N1_Ch1_RF_Communications_Option := V680_Ch1_DeviceVariables.RFCommunicationsOption;
N1_Ch1_Command_Code := V680_Ch1_DeviceVariables.CommandCode;
N1_Ch1_Memory_Address := V680_Ch1_DeviceVariables.MemoryAddress;
N1_Ch1_Data_Size := V680_Ch1_DeviceVariables.DataSize;
N1_Ch1_Refresh_Count_Response := V680_Ch1_DeviceVariables.RefreshCountResponse;
N1_Ch1_Input_SID_response := V680_Ch1_DeviceVariables.InputSIDResponse;
N1_Ch1_Select_UID := V680_Ch1_DeviceVariables.SelectUID;
```

ST

```
//V680_CH1_I0 パラメータ更新
// 入力パラメータ設定
V680_Ch1_DeviceVariables.Status := N1_Ch1_Status;
V680_Ch1_DeviceVariables.Refreshcount := N1_Ch1_Refresh_Count;
V680_Ch1_DeviceVariables.ResponseCode := N1_Ch1_Response_Code;
V680_Ch1_DeviceVariables.InputSID := N1_Ch1_Input_SID;
V680_Ch1_DeviceVariables.InputData1 := N1_Ch1_Input_Data_1;
V680_Ch1_DeviceVariables.UID := N1_Ch1_UID;
V680_Ch1_DeviceVariables.RFCommunicationsTime := N1_Ch1_RF_Communications_Time;
V680_Ch1_DeviceVariables.NoiseLevel := N1_Ch1_Noise_level;

CASE CommandStage OF
  (* アイドル *)
  0 :
    // ReadTrig が TRUE になると、コマンド実行を行う
    IF( ReadTrig = TRUE ) THEN
      // リードパラメータ設定
      RFCommunicationsOption := USINT#0;
      ReadKind := USINT#0;
      MemoryAddr := UINT#8;
      DataSize := UINT#16;
      // 通信対象タグの UID 設定 ( 通信オプション = セレクティブ選択時のみ使用 )
      SelectUID[0] := BYTE#16#66;
      SelectUID[1] := BYTE#16#78;
      SelectUID[2] := BYTE#16#2D;
      SelectUID[3] := BYTE#16#4B;
      SelectUID[4] := BYTE#16#53;
      SelectUID[5] := BYTE#16#05;
      SelectUID[6] := BYTE#16#08;
      SelectUID[7] := BYTE#16#E0;
      // リード実行フラグの設定
      Execute := TRUE;
      // リード実行状態に遷移
      CommandStage := UINT#1;
    END_IF;

    (* リード実行結果取得 *)
    1 :
      IF(Busy = FALSE) THEN
        IF (Error = FALSE) THEN// 正常終了
          (* ----- ↓ 正常処理を記載する ↓ ----- *)

          (* ↑ ----- ↑ *)
          // ユニット動作停止待ち状態に遷移
          CommandStage := UINT#2;

        ELSE
          (* ----- ↓ 異常処理を記載する ↓ ----- *)

          (* ↑ ----- ↑ *)
          // ユニット動作停止待ち状態に遷移
          CommandStage := UINT#2;
        END_IF;
      END_IF;
    END_IF;
  END_CASE;
```

```

(* ユニット動作停止待ち *)
2 :
// ReadTrig が FALSE になると、アイドル状態に移る
IF( ReadTrig = FALSE ) THEN
    CommandStage := UINT#0;
    // リード実行終了フラグの設定
    Execute := FALSE;
END_IF;

END_CASE;

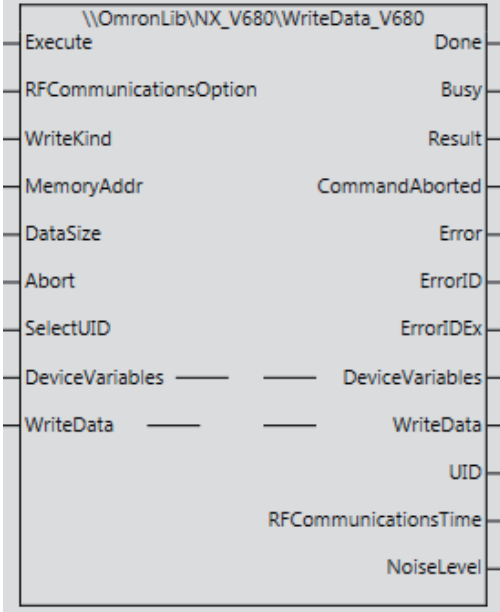
// リードデータ FB
ReadData_V680_instance
(
    Execute := Execute,
    RFCommunicationsOption := RFCommunicationsOption,
    ReadKind := ReadKind,
    MemoryAddr := MemoryAddr,
    DataSize := DataSize,
    Abort := Abort,
    SelectUID := SelectUID,
    Done => Done,
    Busy => Busy,
    Result => Result,
    CommandAborted => CommandAborted,
    Error => Error,
    ErrorID => ErrorID,
    ErrorIDEx => ErrorIDEx,
    Warning => Warning,
    UID => UID,
    RFCommunicationsTime => RFCommunicationsTime,
    NoiseLevel => NoiseLevel,
    DeviceVariables := V680_Ch1_DeviceVariables,
    ReadData := R_Data
);

//V680_CH1_I0 パラメータ更新
// 出力パラメータ設定
N1_Ch1_Operation_Command := V680_Ch1_DeviceVariables.OperationCommand;
N1_Ch1_RF_Communications_Option := V680_Ch1_DeviceVariables.RFCommunicationsOption;
N1_Ch1_Command_Code := V680_Ch1_DeviceVariables.CommandCode;
N1_Ch1_Memory_Address := V680_Ch1_DeviceVariables.MemoryAddress;
N1_Ch1_Data_Size := V680_Ch1_DeviceVariables.DataSize;
N1_Ch1_Refresh_Count_Response := V680_Ch1_DeviceVariables.RefreshCountResponse;
N1_Ch1_Input_SID_response := V680_Ch1_DeviceVariables.InputSIDResponse;
N1_Ch1_Select_UID := V680_Ch1_DeviceVariables.SelectUID;

```

WriteData_V680

RFID ユニット（形 NX-V680）に装着されたアンテナの発信領域内にある RF タグのメモリにデータを書き込みます。

プログラム 部品名	名称	FB/ FUN	グラフィック表現	ST 表現
WriteData_V680	V680 用 RF タグデータ書き込み	FB		WriteData_V680_instance(Execute := <パラメータ>, RFCommunicationsOption := <パラメータ>, WriteKind := <パラメータ>, MemoryAddr := <パラメータ >, DataSize := <パラメータ>, Abort := <パラメータ>, SelectUID := <パラメータ>, Done => <パラメータ>, Busy => <パラメータ>, Result => <パラメータ>, CommandAborted => <パラ メータ>, Error => <パラメータ>, ErrorID => <パラメータ>, ErrorIDEx => <パラメータ>, UID => <パラメータ>, RFCommunicationsTime => < パラメータ>, NoiseLevel => <パラメータ>, WriteData := <パラメータ>, DeviceVariables := <パラメ ータ>, >);

FB/FUN 情報

項目	内容
ライブラリファイル名	OmronLib_NX_V680_Vx_x.slr (x はバージョンを示す)
名前空間	OmronLib\NX_V680
FB/FUN 番号	00214
ソースコード	非公開

入力変数

変数	名称	データ型	内容	有効範囲	単位	初期値
Execute	起動	BOOL	立ち上がり時に処理を開始します。 TRUE：実行する FALSE：実行しない	TRUE,FALSE	—	FALSE
RFCommunicationsOption	交信オプション	USINT	交信時の動作シーケンスを指定します。 0：トリガ 1：オート 2：リピート 3：FIFO トリガ 4：FIFO リピート 5：マルチトリガ 6：マルチリピート 7：セレクトティブ *1	0 ～ 7	—	0
WriteKind	ライト種別	USINT	ライトコマンドの種別を指定します。*2 0：通常 1：誤り検出付 2：誤り訂正付	0 ～ 2	—	0
MemoryAddr	メモリアドレス	UINT	RF タグからデータを書き込むメモリの先頭アドレスを指定します。*3	0 ～ 65535	Byte	0
DataSize	データサイズ	UINT	RF タグからデータを書き込むサイズを入力します。*4	1 ～ 8192	Byte	0
Abort	中断	BOOL	立ち上がり時に処理を中断します。	TRUE,FALSE	—	FALSE
SelectUID	セレクト UID	ARRAY[0..7]OF BYTE	交信オプション「セレクトティブ」時に、交信対象の RF タグの UID を指定します。	データ型に従う	—	16#00

*1. "セレクトティブ"を使用する場合、RFID ユニットの使用されるチャンネルに、I/O エントリ "セレクト UID"を追加してください。

*2. ライトコマンドの種類と動作の違いについては、□『NX シリーズ RFID ユニット ユーザーズマニュアル (SDGR-717)』のコマンド一覧を参照ください。

*3. 実際に使用する RF タグのメモリマップを考慮して入力してください。

*4. 実際に使用する RF タグのメモリ容量を考慮して入力してください。0 のときは何もせずに終了します。

出力変数

変数	名称	データ型	内容	有効範囲	単位	初期値
Done	完了	BOOL	処理が完了したときに TRUE になります。	TRUE,FALSE	—	FALSE
Busy	実行中	BOOL	処理を受け付けたときに TRUE になります。	TRUE,FALSE	—	FALSE
Result	結果	BOOL	RF タグと交信した結果を出力するごとに、値を TRUE と FALSE と交互に変化します。	TRUE,FALSE	—	FALSE
Command Aborted	実行中断	BOOL	処理が中断されたときに TRUE になります。	TRUE,FALSE	—	FALSE
Error	エラー	BOOL	エラーが発生したときに TRUE になります。 TRUE : 異常終了 FALSE: 正常終了、または実行中	TRUE,FALSE	—	FALSE
ErrorID	エラーコード	WORD	エラーコードが出力されます。 異常終了時の異常 ID 正常終了時は 16#0	*1	—	0
ErrorIDEx	拡張エラーコード	DWORD	拡張エラーコードが出力されます。 異常終了時の異常 ID 正常終了時は 16#0	*1	—	0
UID ^{*2}	UID	ARRAY[0..7] OF BYTE	交信した RF タグの UID が出力されます。	データ型に従う	—	16#00
RFCommunicationsTime ^{*3}	交信時間	UINT	計測した交信時間が出力されます。	0 ~ 65535	ms	0
NoiseLevel ^{*4}	ノイズレベル	UINT	測定したノイズレベルが出力されます。	0 ~ 99	—	0

*1. 詳細は、「トラブルシューティング (P.70)」を参照してください。

*2. 使用する場合、RFID ユニットの使用されるチャンネルに、I/O エントリ "UID" を追加してください。

*3. 使用する場合、RFID ユニットの使用されるチャンネルに、I/O エントリ "交信時間" を追加してください。

*4. 使用する場合、RFID ユニットの使用されるチャンネルに、I/O エントリ "ノイズレベル" を追加してください。

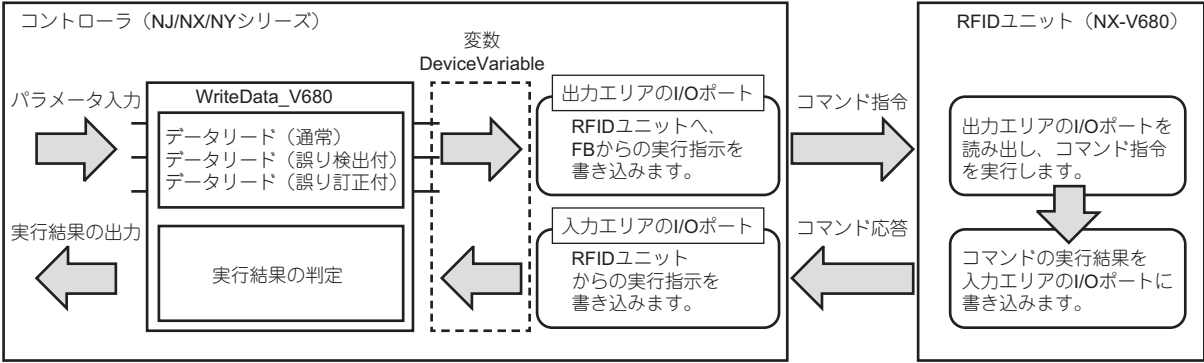
入出力変数

変数	名称	データ型	内容	有効範囲	単位	初期値
DeviceVariables	デバイス変数	OmronLib\RFID\sNX-V680DeviceVariables	RFID ユニットのデバイス変数 (入力エリア / 出力エリアの I/O ポート)	データ型に従う	—	—
WriteData	書き込みデータ	ARRAY[可変長] OF BYTE	RF タグへ書き込むデータ配列	データ型に従う	—	—

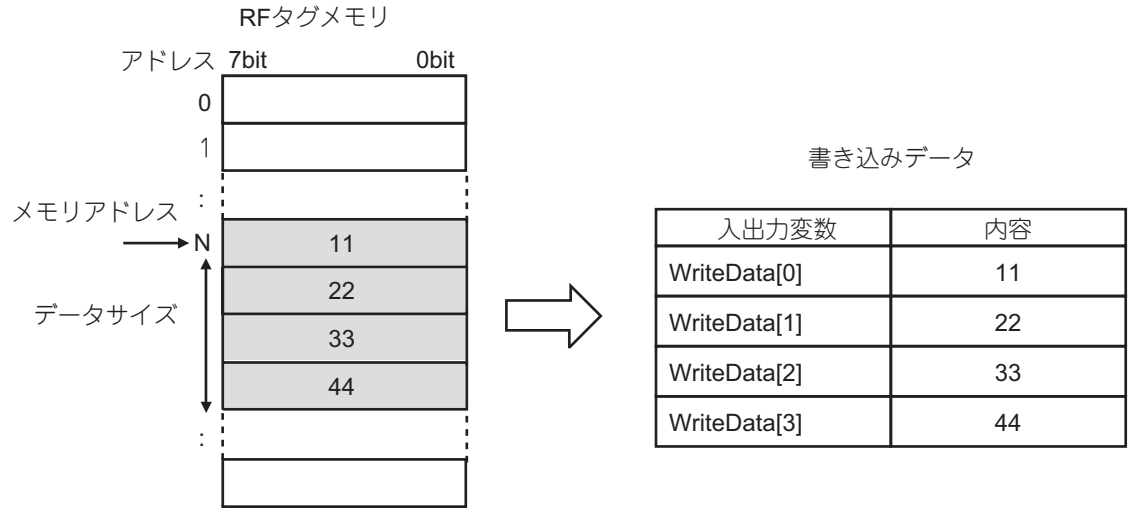
(注) 任意長の BYTE 型配列が指定可能です。ただし、配列長さは DataSize 以上でなければなりません。配列先頭番号は 0 始まりでも n 始まりでも構いません。

機能

本FBは、NJ/NX/NYシリーズコントローラとデータ交換しているRFIDユニット（形NX-V680）に対して、RFタグのメモリを書き込むコマンド制御を行います。そのため、制御したいRFIDユニットのデバイス変数（出力エリアのI/Oポートならびに入力エリアのI/Oポート）をFBの入出力変数「DeviceVariables」に設定する必要があります。



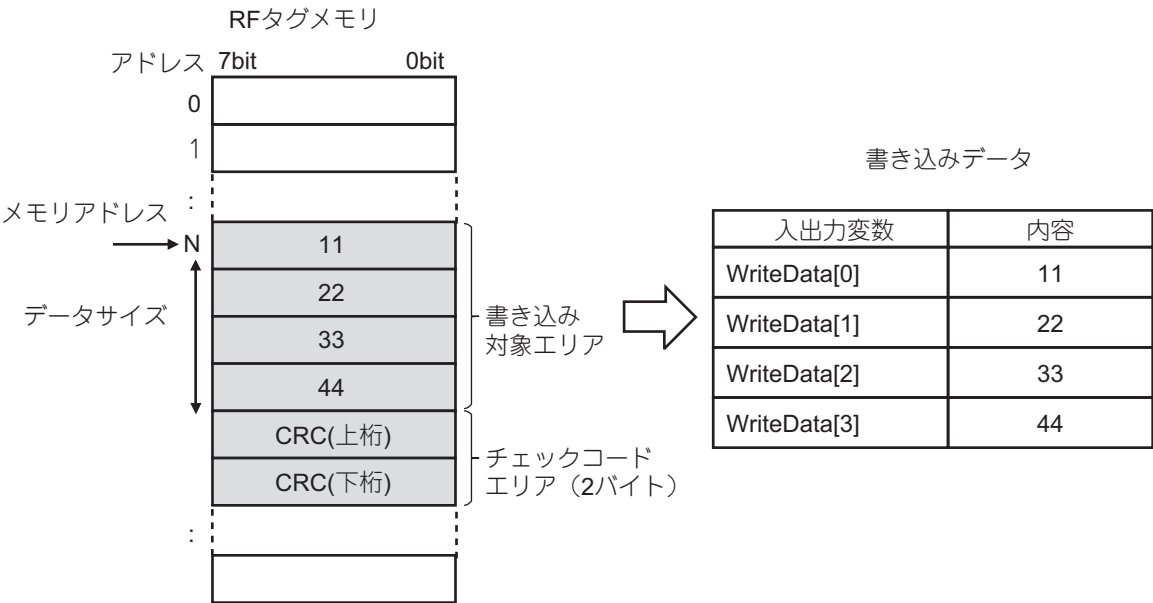
- Execute（起動）の立ち上がりで、WriteData（書き込みデータ）で指定したバイト配列のデータを、RFタグのメモリへ書き込みます。



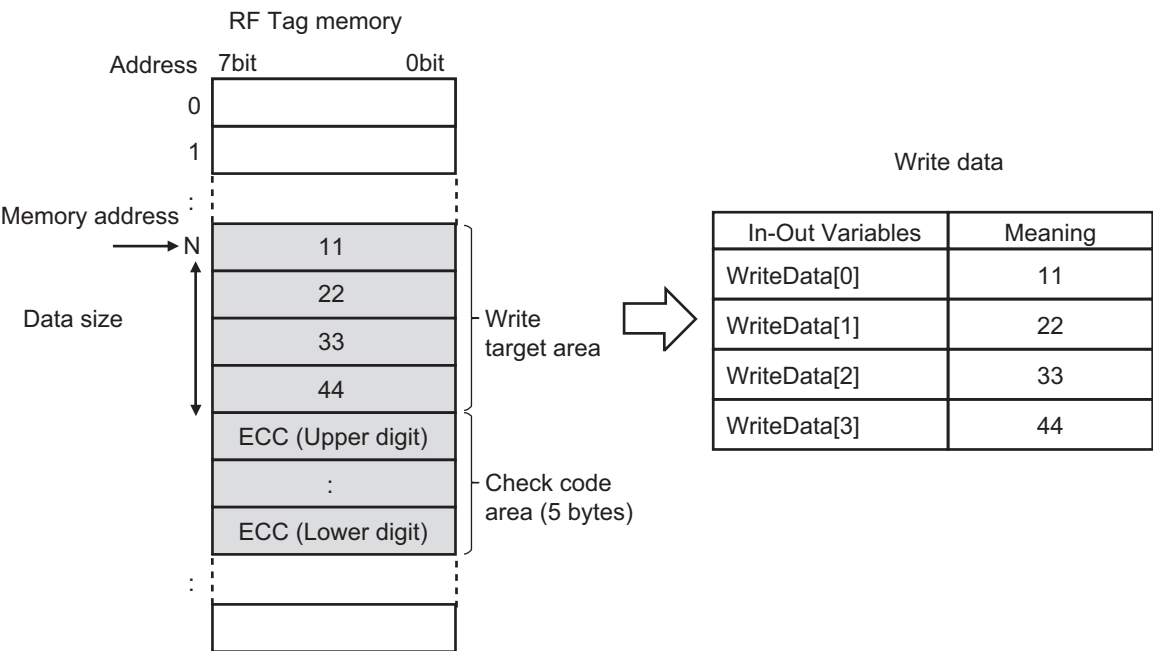
- 1回の実行でRFタグに書き込めるメモリ容量は、指定するWriteKind（ライト種別）によって異なります。MemoryAddr（メモリアドレス）とDataSize（データサイズ）で指定可能な範囲を以下に示します。

WriteKind	MemoryAddr	DataSize
0：通常	0 ～ 65535	1 ～ 8192
1：誤り検出付	0 ～ 65533	1 ～ 8190
2：誤り訂正付	0 ～ 65530	1 ～ 510

- WriteKind（ライト種別）を " 誤り検出付 " で実行すると、RF タグの対象エリアにデータを書き込み、後ろの 2 バイトのエリアに誤り検出用のチェックコードを書き込みます。



- WriteKind（ライト種別）を " 誤り訂正付 " で実行すると、RF タグの対象エリアにデータを書き込み、後ろの 5 バイトのエリアに誤り訂正用のチェックコードを書き込みます。



タイミングチャート

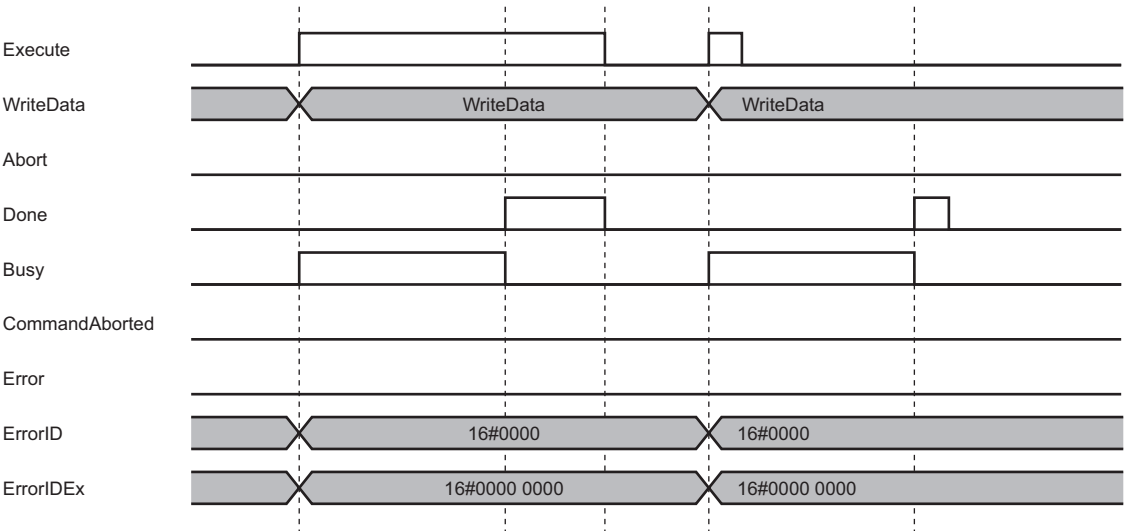
タイミングチャートを以下に示します。

1 個の RF タグと交信する（交信オプション：トリガ、オート、FIFO トリガ、セレクトティブ）

本 FB の RFCommunicationsOption（交信オプション）にトリガ / オート / FIFO トリガ / セレクトティブを指定して起動した場合、アンテナの交信領域にある 1 個の RF タグと交信した結果が出力されます。

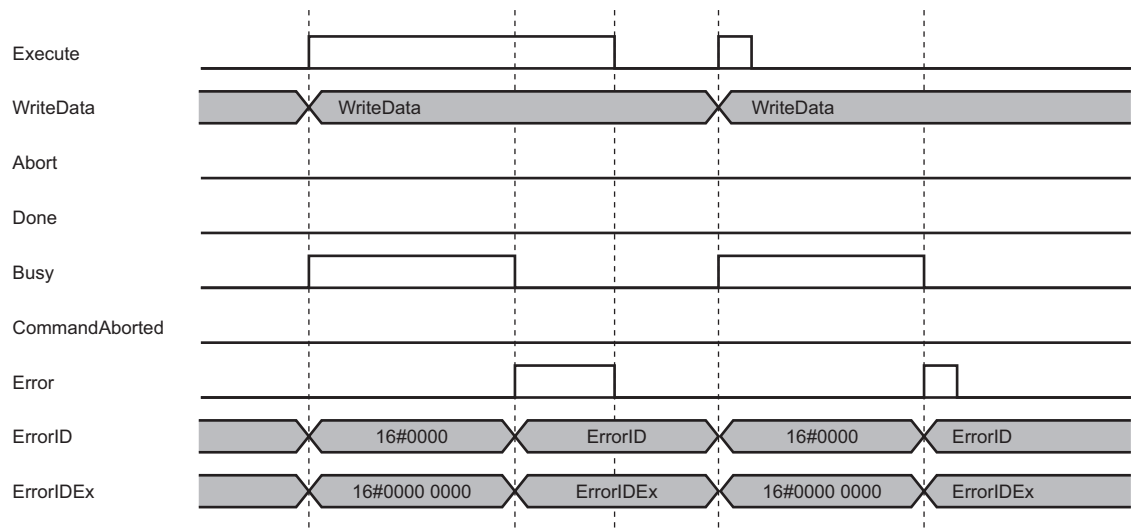
● 正常終了時のタイミングチャート

- Execute（起動）の立ち上がりと同時に Busy（実行中）が TRUE となります。
- RF タグのメモリにデータ書き込みが正常終了すると、Done（完了）が TRUE となり、Busy（実行中）が FALSE となります。同時に UID にデータ列、RFCommunicationsTime（交信時間）や NoiseLevel（ノイズレベル）に値が出力されます。
- 本 FB の実行完了後も Execute（起動）が TRUE である場合は、Done（完了）の出力は保持されます。
- 本 FB の実行完了前に Execute（起動）が FALSE となった場合は、Done（完了）は実行完了後の 1 タスク周期だけ TRUE となります。



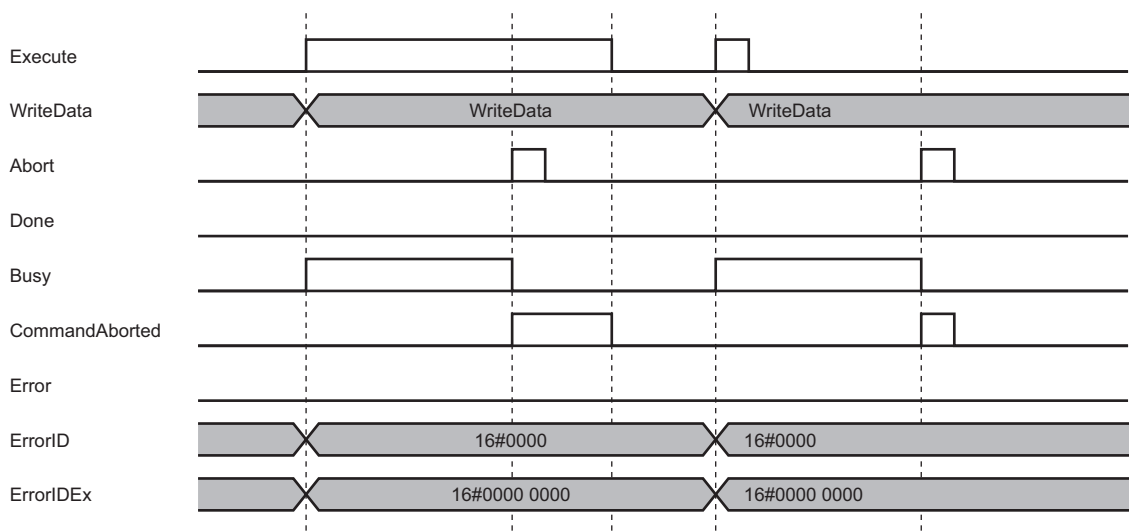
● 異常終了時のタイミングチャート

- 本 FB 実行中に異常が発生した場合、Error（エラー）が TRUE となり、Busy（実行中）が FALSE となります。
- ErrorID（エラーコード）、ErrorIDEx（拡張エラーコード）に出力された値を参照することによって、異常の原因を知ることができます。
- 本 FB の実行完了後も Execute（起動）が TRUE である場合は、Error（エラー）の出力は保持されます。
- ErrorID（エラーコード）や ErrorIDEx（拡張エラーコード）の出力は本 FB を再実行するまで保持されます。
- 本 FB の実行完了前に Execute（起動）が FALSE となった場合は、Error（エラー）は実行完了後の 1 タスク周期だけ TRUE となります。



● 中断時のタイミングチャート

本 FB 実行中に RF タグと通信を中断するには、Abort（中断）を TRUE にします。Abort（中断）が TRUE になると、Busy（実行中）が FALSE となり、CommandAborted（実行中断）が TRUE となります。

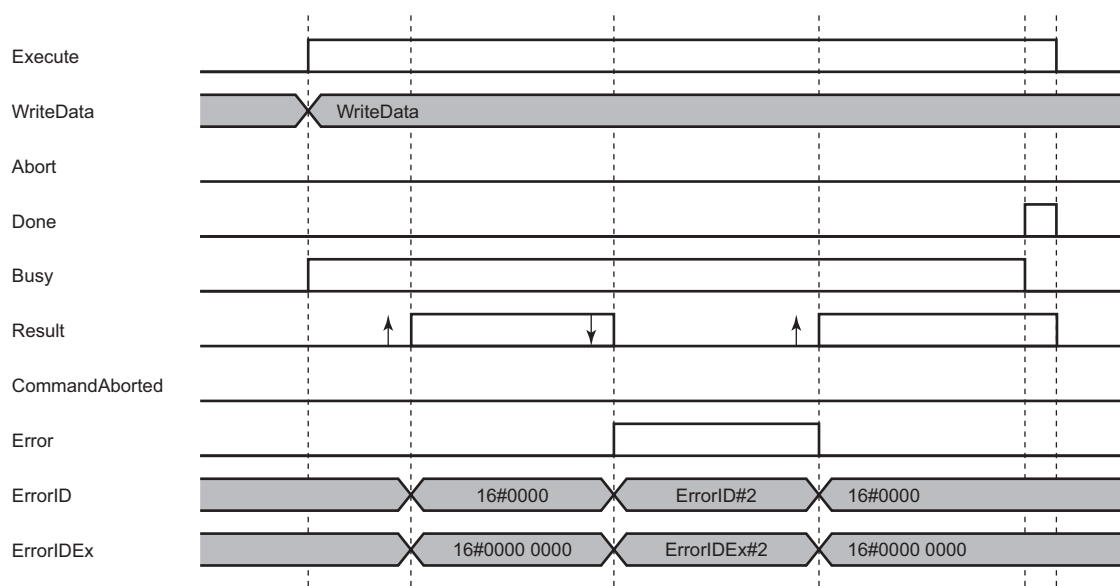


複数の RF タグと交信する（交信オプション：マルチトリガ）

本 FB の RFCommunicationsOption（交信オプション）にマルチトリガを指定して起動した場合、アンテナの交信領域内にある複数の RF タグと交信した結果が出力されます。

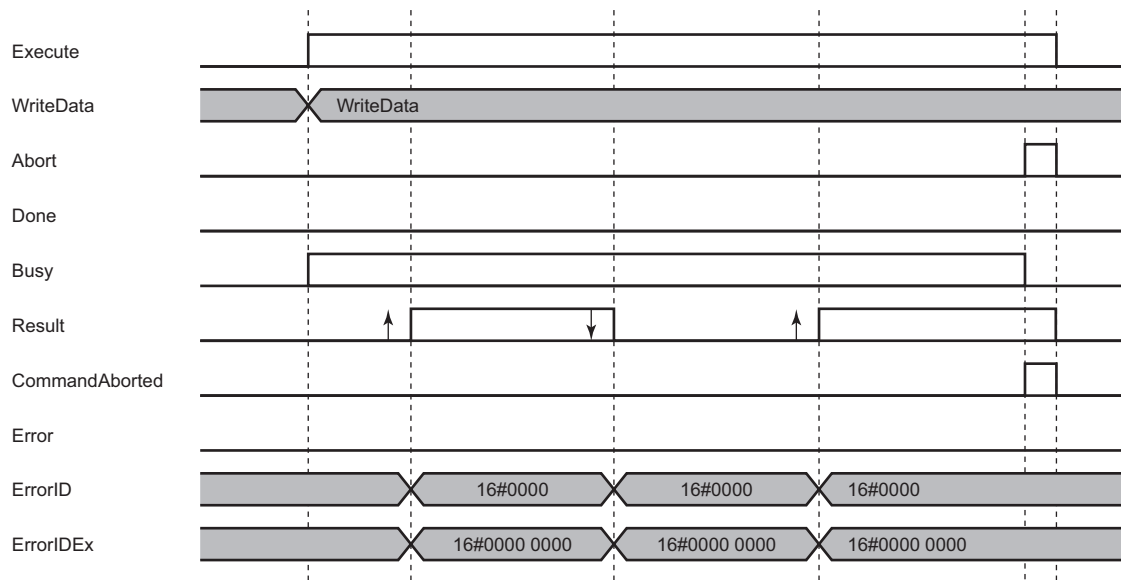
● 複数の RF タグと交信時のタイミングチャート

- Execute（起動）の立ち上がりと同時に Busy（実行中）が TRUE となります。
- 複数の RF タグとの交信毎に、Result（結果）が FALSE → TRUE → FALSE → TRUE とトグル動作します。
- 交信領域内にある全ての RF タグと交信が完了すると、Done（完了）が TRUE となり、Busy（実行中）が FALSE となります。
- RF タグのメモリにデータ書き込みが正常終了すると、Result（結果）の変化と同時に、UID にデータ列、RFCommunicationsOption（交信時間）や NoiseLevel（ノイズレベル）に値が出力されます。
- RF タグのメモリにデータ書き込みが異常終了すると、Result（結果）の変化と同時に、Error（エラー）が TRUE となります。ErrorID（エラーコード）、ErrorIDEx（拡張エラーコード）に出力された値を参照することによって、異常の原因を知ることができます。
- Done（完了）の立ち下がりと同時に Result（結果）が FALSE となります。



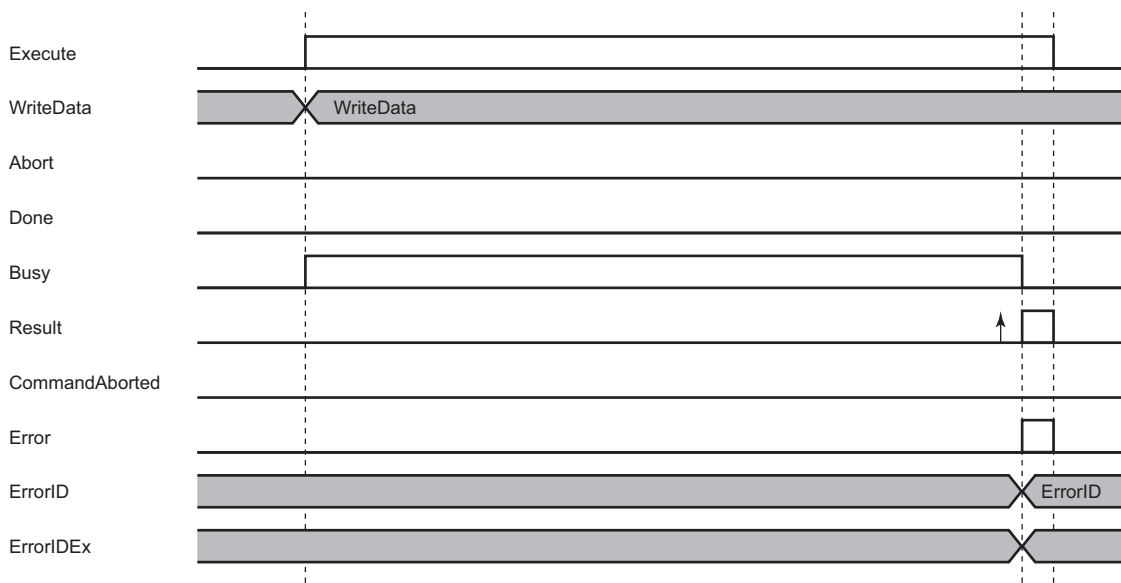
● 中断時のタイミングチャート

- 本 FB 実行中に RF タグと通信を中断するには、Abort（中断）を TRUE にします。Abort（中断）が TRUE になると、Busy（実行中）が FALSE となり、CommandAborted（実行中断）が TRUE となります。
- CommandAborted（実行中断）の立ち上がりと同時に Result（結果）が FALSE となります。



● タグ不在時のタイミングチャート

- 本 FB 実行中に一度も RF タグと通信が行われなかった場合、Result（結果）と Error（エラー）が TRUE となり、Busy（実行中）が FALSE となります。
- ErrorID（エラーコード）、ErrorIDEx（拡張エラーコード）に出力された値を参照することによって、異常の原因を知ることができます。
- 本 FB の実行完了後も Execute（起動）が TRUE である場合は、Result（結果）と Error（エラー）の出力は保持されます。
- ErrorID（エラーコード）や ErrorIDEx（拡張エラーコード）の出力は本 FB を再実行するまで保持されます。
- 本 FB の実行完了前に Execute（起動）が FALSE となった場合は、Result（結果）と Error（エラー）は実行完了後の 1 タスク周期だけ TRUE となります。

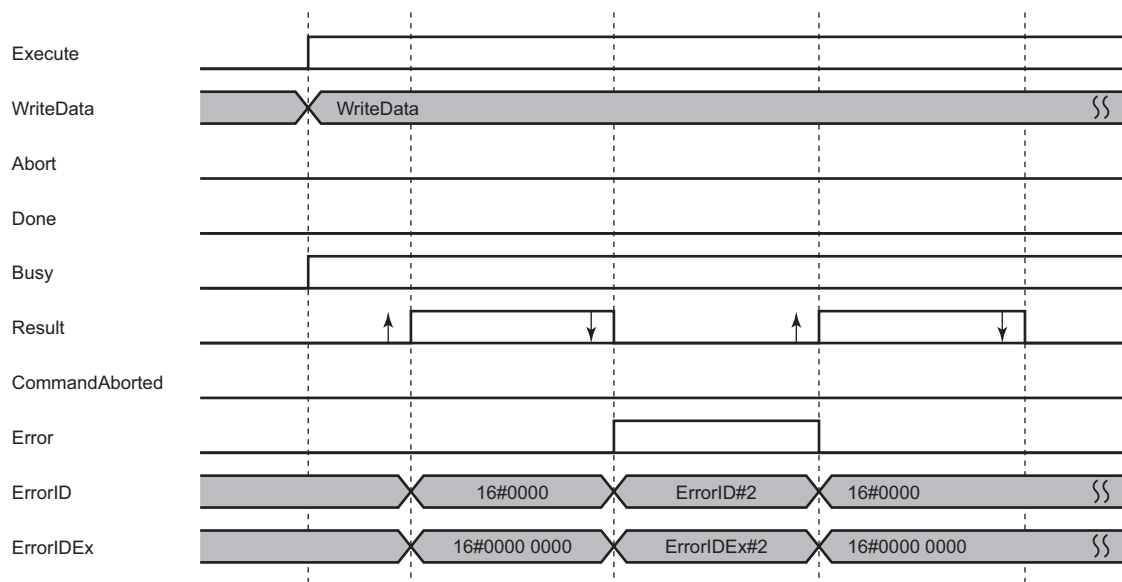


RF タグと繰り返し通信する（通信オプション：リピート、FIFO リピート、マルチリピート）

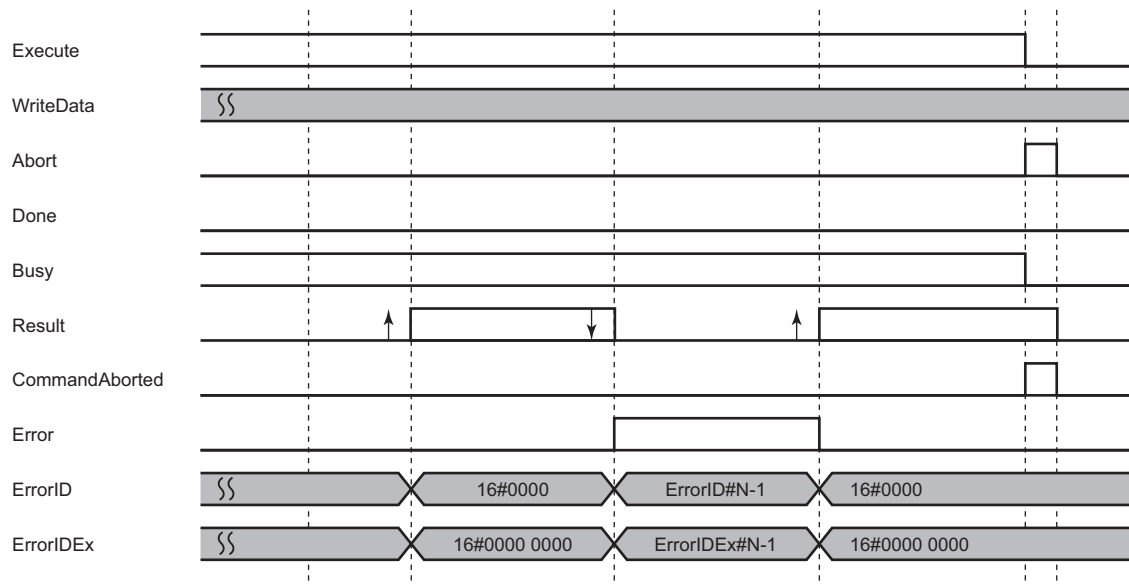
本 FB の RFCommunicationsOption（通信オプション）をリピート / FIFO リピート / マルチリピートを指定して起動した場合、アンテナの通信領域内に移動してくる RF タグと通信を繰り返した結果が出力されます。

● 繰り返し通信時のタイミングチャート

- Execute（起動）の立ち上がりと同時に Busy（実行中）が TRUE となります。
- 複数の RF タグとの通信毎に、Result（結果）が FALSE → TRUE → FALSE → TRUE とトグル動作します。
- RF タグのメモリにデータ書き込みが正常終了すると、Result（結果）の変化と同時に、ReadData（読み出しデータ）や UID にデータ列、RFCommunicationsTime（通信時間）や NoiseLevel（ノイズレベル）に値が出力されます。
- RF タグのメモリにデータ書き込みが異常終了すると、Result（結果）の変化と同時に、Error（エラー）が TRUE となります。ErrorID（エラーコード）、ErrorIDEx（拡張エラーコード）に出力された値を参照することによって、異常の原因を知ることができます。

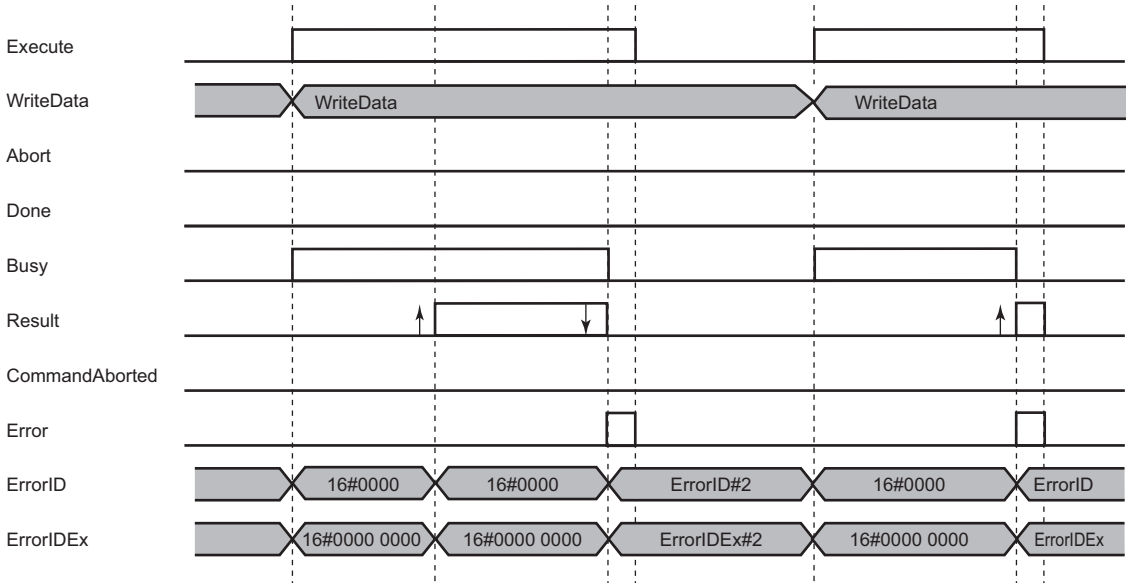


- RF タグと繰り返し通信を終了するには、Abort（中断）を TRUE にする必要があります。
Abort（中断）が TRUE になると、Busy（実行中）が FALSE となり、CommandAborted（実行中断）が TRUE となります。
- CommandAborted（実行中断）の立ち下がりと同時に Result（結果）が FALSE となります。



● 異常終了時のタイミングチャート

- 本 FB 実行中に特定の異常が発生した場合、Error（エラー）が TRUE、Busy（実行中）が FALSE となり、繰り返しを中断します。
- Error（エラー）の立ち下がりと同時に Result（結果）が FALSE となります。
- ErrorID（エラーコード）、ErrorIDEx（拡張エラーコード）に出力された値を参照することによって、異常の原因を知ることができます。
- 本 FB の実行完了後も Execute（起動）が TRUE である場合は、Error（エラー）の出力は保持されます。
- ErrorID（エラーコード）や ErrorIDEx（拡張エラーコード）の出力は本 FB を再実行するまで保持されます。
- 本 FB の実行完了前に Execute（起動）が FALSE となった場合は、Error（エラー）は実行完了後の 1 タスク周期だけ TRUE となります。



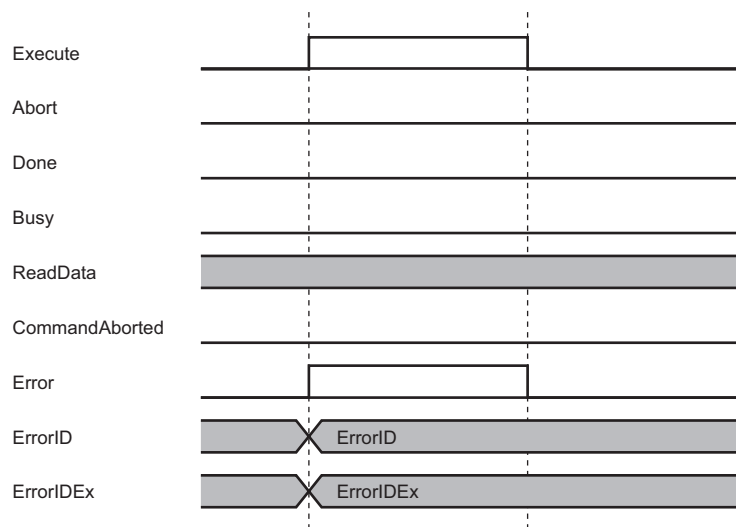
（注） 繰り返し交信を停止させる特定の異常は以下の通りとなります。

エラーコード	拡張エラーコード	状態
16#3D14	16#6AA00000	アンテナ構成異常
	16#6AA10000	アンプ電源異常
	16#6AA20000	アンプ断線検知
	16#6A720000	RF タグ不在エラー

共通の振る舞い

● 起動時に異常を検出した時のタイミングチャート

- 本 FB 起動時に異常が検知された場合、Error（エラー）が TRUE となります。Busy（実行中）は FALSE のままです。
- ErrorID（エラーコード）、ErrorIDEx（拡張エラーコード）に出力された値を参照することによって、異常の原因を知ることができます。
- 本 FB の実行完了後も Execute（起動）が TRUE である場合は、Error（エラー）の出力は保持されます。
- ErrorID（エラーコード）や ErrorIDEx（拡張エラーコード）の出力は本 FB を再実行するまで保持されます。



使用上の注意

- RFID ユニットが既にコマンド実行中でビジー状態の場合は、異常終了として Error（異常）を TRUE で出力し、本 FB の機能を実行しません。
- RFID ユニット（形 NX-V680C2）を使用される際、本 FB への入出力変数として使用するデバイス変数はチャンネル Ch1 か Ch2 のいずれかに統一してください。各チャンネルのデバイス変数が混在すると正しく動作できません。
- 本 FB を実行する前に、使用する形 NX-V680C□ のマニュアルをよく読み、安全を確認のうえで使用してください。
- RFID ユニットの動作パラメータ "データ格納順序" が "降順" の時、奇数バイトのデータサイズを指定しないでください。正しいデータが書き込めません。
- RFID ユニットの使用されるチャンネルに、I/O エントリ "出力データ 2 ～ 8" は追加しないでください。出力データの合計サイズが増えると正しく動作できません。

トラブルシューティング

本 FB が異常終了した時に出力するエラーコードの一覧を以下に示します。

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#3D14	16#00000001	交信オプション不正	RFCommunicationsOption の値が有効範囲外です。	RFCommunicationsOption に正しい値を設定してください。
	16#00000002	ライト種別不正	WriteKind の値が有効範囲外です。	WriteKind に正しい値を設定してください。
	16#00000003	メモリアドレス不正	MemoryAddr の値が有効範囲外です。	MemoryAddr に正しい値を設定してください。
	16#00000004	データサイズ不正	DataSize の値が有効範囲外です。	DataSize に正しい値を設定してください。
	16#00000005	データ配列不正	WriteData の配列長さが DataSize 未満です。	WriteData の配列長さを DataSize 以上してください。
	16#00000006	起動不可	RFID ユニットのコマンド実行可能状態でないため本 FB を起動できません。	RFID ユニットの状態が、コマンドが実行可能であることを確認のうえ、再度 FB を実行してください。
	16#6AA00000	アンテナ構成異常	サポート外のアンテナが接続されています。 ※ 形 NX-V680C2 のユニットと形 V680-H01-V2 の組み合わせ	形 NX-V680C2 には形 V680-H01-V2 以外のアンテナを接続してください。 または、形 NX-V680C1 を使用してください。
	16#6AA10000	アンプ電源異常	アンプを駆動する電源が供給されていません。	I/O 電源の入力を確認してください。
	16#6AA20000	アンプ断線検知	アンプの断線を検出しました。認識できませんでした。	アンプを接続してください。 または、アンプを交換してください。
	16#6A720000	RF タグ不在エラー	RF タグが交信領域内に存在しません。	RF タグが交信領域内に入るように設備を調整してください。
	16#6A700000	RF タグ交信失敗	RF タグと交信中にエラーが発生し正常終了できませんでした。	RF タグの移動速度、RF タグ間の距離を仕様範囲内になるように変更してください。 また、ノイズが多い環境ではノイズ対策を実施してください。
	16#6A710000	RF タグベリファイエラー	RF タグに正しくデータが書き込めませんでした。	RF タグの移動速度、RF タグ間の距離を仕様範囲内になるように変更してください。 また、ノイズが多い環境ではノイズ対策を実施してください。
	16#6A730000	RF タグデータ消失	RF タグに正しくデータが書き込めず、データが消失した可能性があります。	RF タグがリーダライタの交信領域に存在する状態で再処理してください。 RF タグの移動速度、RF タグ間の距離を仕様範囲内になるように変更してください。 また、ノイズが多い環境ではノイズ対策を実施してください。
	16#6A7A0000	RF タグアドレスエラー	RF タグのアドレスが間違っています。	使用している RF タグのメモリ容量を確認し、指定するアドレスをメモリ容量の範囲内になるように修正してください。

エラーコード	拡張エラーコード	状態	内容	対処法
16#3D14	16#6A7D0000	RF タグライトプロテクトエラー	RF タグのライトプロテクトされているエリアに書き込みを行おうとしました。	指定するアドレス、バイト数を正しい値に修正してください。または、ライトプロテクトを解除してください。
	16#6A790000	RF タグ応答エラー	RF タグがエラー応答し正常終了できませんでした。	ノイズの多い環境ではノイズ対策を実施してください。* ¹
	16#6A7E0000	RF タグロックエラー	RF タグのロックされているエリアに書き込みを行おうとしました。	RF タグを変更してください。
	16#6A7F0000	RF タグカスタムコードエラー	使用できない RF タグと交信しました。	RF タグを変更してください。
	16#6AC00000	未定義コマンド	未定義のコマンドのため実行できません。	DeviceVariables に正しい変数を設定してください。
	16#6AC10000	コマンドパラメータ不正	コマンドのパラメータが間違っているため実行できません。	DeviceVariables に正しい変数を設定してください。
	16#6AC20000	コマンド実行不可	コマンドの実行条件が成立していないためコマンドが実行できません。	コマンドに対して、実行可能な交信オプションに修正してください。

*1. ノイズ対策で解決しない場合は RF タグを変更してください。

サンプルプログラム

プログラム説明

RF タグのメモリアドレス 8 に、16 バイトのデータを書き込みます。

前提条件

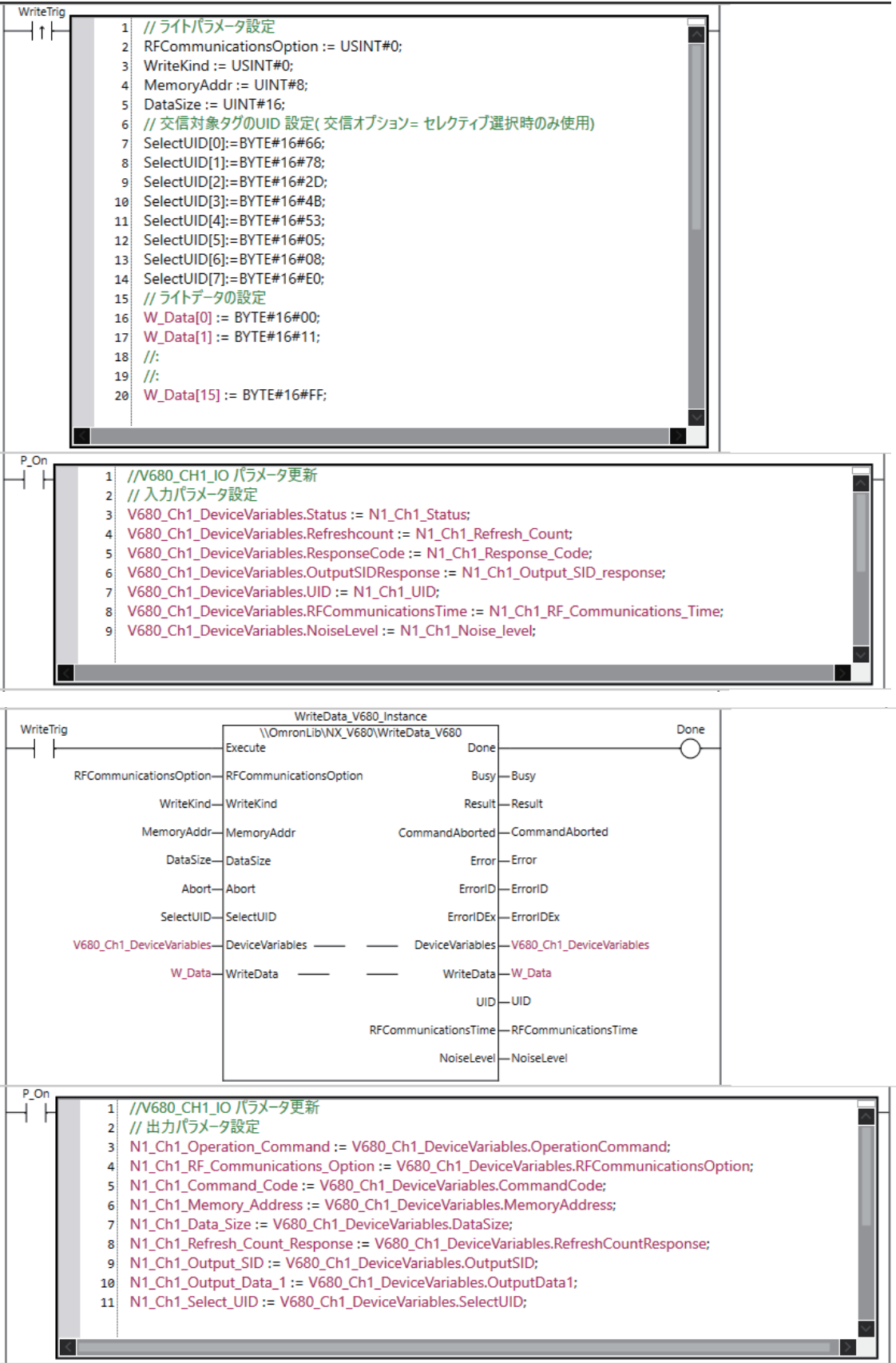
動作させたい RFID ユニットのデバイス変数を生成し、ユーザプログラム内で外部参照してください。デバイス変数生成方法の詳細は、□『Sysmac Studio Version1 オペレーションマニュアル (SBCA-470)』を参照してください。

主な変数

変数	名称	データ型	初期値	内容
WriteData_V680_Instance	ライトデータ V680 FB	OmronLib\NX_V680\WriteData_V680	-	NX-V680 のライトを実施する為の FB
V680_Ch1_DeviceVariables	Ch1 デバイス変数	OmronLib\NX_V680\sV680DeviceVariables	-	Ch1 デバイス変数の構造体
W_Data	ライトデータ用バッファ	ARRAY[0..8191] OF BYTE	16#00	ライトデータを設定
WriteTrig	ライト実行フラグ	BOOL	FALSE	ライト実行時に TRUE
RFCommunicationOption	交信オプション	USINT	0	交信オプションを設定
WriteKind	ライト種別	USINT	0	ライト種別を設定
MemoryAddr	メモリアドレス	UINT	0	交信開始アドレスを設定
DataSize	データサイズ	UINT	0	交信データサイズを設定
Abort	実行停止フラグ	BOOL	FALSE	ライト実行停止時に TRUE
SelectUID	セレクト UID	ARRAY[0..7] OF BYTE	16#00	交信対象タグの UID を設定
CommandStage	コマンド状態	UINT	0	ST プログラムの実行状態
Execute	実行フラグ	BOOL	FALSE	ST プログラム内の実行状態フラグ
Done	実行完了フラグ	BOOL	-	"WriteData_V680_Instance" の出力用変数
Busy	実行中フラグ	BOOL	-	"WriteData_V680_Instance" の出力用変数
Result	結果フラグ	BOOL	-	"WriteData_V680_Instance" の出力用変数
CommandAborted	実行中断フラグ	BOOL	-	"WriteData_V680_Instance" の出力用変数
Error	エラーフラグ	BOOL	-	"WriteData_V680_Instance" の出力用変数

変数	名称	データ型	初期値	内容
ErrorID	エラーコード	WORD	-	"WriteData_V680_Instance" の出力用変数
ErrorIDEx	拡張エラーコード	DWORD	-	"WriteData_V680_Instance" の出力用変数
UID	UID	ARRAY[0..7] OF BYTE	-	"WriteData_V680_Instance" の出力用変数
RFCommunicationsTime	交信時間	UINT	-	"WriteData_V680_Instance" の出力用変数
NoiseLevel	ノイズレベル	UINT	-	"WriteData_V680_Instance" の出力用変数

ラダー図



● インライン ST のコード（ラダー図 0 行め）：

```
// ライトパラメータ設定
RFCommunicationsOption := USINT#0;
WriteKind := USINT#0;
MemoryAddr := UINT#8;
DataSize := UINT#16;
// 交信対象タグの UID 設定（交信オプション＝セレクトティブ選択時のみ使用）
SelectUID[0] := BYTE#16#66;
SelectUID[1] := BYTE#16#78;
SelectUID[2] := BYTE#16#2D;
SelectUID[3] := BYTE#16#4B;
SelectUID[4] := BYTE#16#53;
SelectUID[5] := BYTE#16#05;
SelectUID[6] := BYTE#16#08;
SelectUID[7] := BYTE#16#E0;
// ライトデータの設定
W_Data[0] := BYTE#16#00;
W_Data[1] := BYTE#16#11;
//:
//:
W_Data[15] := BYTE#16#FF;
```

● インライン ST のコード（ラダー図 1 行め）：

```
//V680_Ch1_I0 パラメータ更新
// 入力パラメータ設定
V680_Ch1_DeviceVariables.Status := N1_Ch1_Status;
V680_Ch1_DeviceVariables.Refreshcount := N1_Ch1_Refresh_Count;
V680_Ch1_DeviceVariables.ResponseCode := N1_Ch1_Response_Code;
V680_Ch1_DeviceVariables.OutputSIDResponse := N1_Ch1_Output_SID_response;
V680_Ch1_DeviceVariables.UID := N1_Ch1_UID;
V680_Ch1_DeviceVariables.RFCommunicationsTime := N1_Ch1_RF_Communications_Time;
V680_Ch1_DeviceVariables.NoiseLevel := N1_Ch1_Noise_Level;
```

● インライン ST のコード（ラダー図 3 行め）：

```
//V680_Ch1_I0 パラメータ更新
// 出力パラメータ設定
N1_Ch1_Operation_Command := V680_Ch1_DeviceVariables.OperationCommand;
N1_Ch1_RF_Communications_Option := V680_Ch1_DeviceVariables.RFCommunicationsOption;
N1_Ch1_Command_Code := V680_Ch1_DeviceVariables.CommandCode;
N1_Ch1_Memory_Address := V680_Ch1_DeviceVariables.MemoryAddress;
N1_Ch1_Data_Size := V680_Ch1_DeviceVariables.DataSize;
N1_Ch1_Refresh_Count_Response := V680_Ch1_DeviceVariables.RefreshCountResponse;
N1_Ch1_Output_SID := V680_Ch1_DeviceVariables.OutputSID;
N1_Ch1_Output_Data_1 := V680_Ch1_DeviceVariables.OutputData1;
N1_Ch1_Select_UID := V680_Ch1_DeviceVariables.SelectUID;
```

ST

```
//V680_CH1_I0 パラメータ更新
// 入力パラメータ設定
V680_Ch1_DeviceVariables.Status := N1_Ch1_Status;
V680_Ch1_DeviceVariables.Refreshcount := N1_Ch1_Refresh_Count;
V680_Ch1_DeviceVariables.ResponseCode := N1_Ch1_Response_Code;
V680_Ch1_DeviceVariables.OutputSIDResponse := N1_Ch1_Output_SID_response;
V680_Ch1_DeviceVariables.UID := N1_Ch1_UID;
V680_Ch1_DeviceVariables.RFCommunicationsTime := N1_Ch1_RF_Communications_Time;
V680_Ch1_DeviceVariables.NoiseLevel := N1_Ch1_Noise_level;
```

CASE CommandStage OF

(* アイドル *)

0 :

// WriteTrig が TRUE になると、コマンド実行を行う

IF(WriteTrig = TRUE) THEN

// ライトパラメータ設定

RFCommunicationsOption := USINT#0;

WriteKind := USINT#0;

MemoryAddr := UINT#8;

DataSize := UINT#16;

// 通信対象タグの UID 設定 (通信オプション=セレクトティブ選択時のみ使用)

SelectUID[0]:=BYTE#16#66;

SelectUID[1]:=BYTE#16#78;

SelectUID[2]:=BYTE#16#2D;

SelectUID[3]:=BYTE#16#4B;

SelectUID[4]:=BYTE#16#53;

SelectUID[5]:=BYTE#16#05;

SelectUID[6]:=BYTE#16#08;

SelectUID[7]:=BYTE#16#E0;

// ライトデータの設定

W_Data[0] := BYTE#16#00;

W_Data[1] := BYTE#16#11;

// :

// :

W_Data[15] := BYTE#16#FF;

// ライト実行フラグの設定

Execute := TRUE;

// ライト実行状態に遷移

CommandStage := UINT#1;

END_IF;

(* ライト実行結果取得 *)

1 :

IF(Busy = FALSE) THEN

IF (Error = FALSE) THEN// 正常終了

(* ----- ↓ 正常処理を記載する ↓ ----- *)

(* ↑ ----- ↑ *)

// ユニット動作停止待ち状態に遷移

CommandStage := UINT#2;

ELSE

(* ----- ↓ 異常処理を記載する ↓ ----- *)

(* ↑ ----- ↑ *)

```

        // ユニット動作停止待ち状態に遷移
        CommandStage := UINT#2;
    END_IF;
END_IF;

(* ユニット動作停止待ち *)
2 :
// WriteTrig が FALSE になると、アイドル状態に遷移する
IF( WriteTrig = FALSE ) THEN
    CommandStage := UINT#0;
    // ライト実行終了フラグの設定
    Execute := FALSE;
END_IF;

END_CASE;

// ライトデータ FB
WriteData_V680_instance
(
    Execute := Execute,
    RFCommunicationsOption := RFCommunicationsOption,
    WriteKind := WriteKind,
    MemoryAddr := MemoryAddr,
    DataSize := DataSize,
    Abort := Abort,
    SelectUID := SelectUID,
    Done => Done,
    Busy => Busy,
    Result => Result,
    CommandAborted => CommandAborted,
    Error => Error,
    ErrorID => ErrorID,
    ErrorIDEx => ErrorIDEx,
    UID => UID,
    RFCommunicationsTime => RFCommunicationsTime,
    NoiseLevel => NoiseLevel,
    DeviceVariables := V680_Ch1_DeviceVariables,
    WriteData := W_Data
);

//V680_CH1_I0 パラメータ更新
// 出力パラメータ設定
N1_Ch1_Operation_Command := V680_Ch1_DeviceVariables.OperationCommand;
N1_Ch1_RF_Communications_Option := V680_Ch1_DeviceVariables.RFCommunicationsOption;
N1_Ch1_Command_Code := V680_Ch1_DeviceVariables.CommandCode;
N1_Ch1_Memory_Address := V680_Ch1_DeviceVariables.MemoryAddress;
N1_Ch1_Data_Size := V680_Ch1_DeviceVariables.DataSize;
N1_Ch1_Refresh_Count_Response := V680_Ch1_DeviceVariables.RefreshCountResponse;
N1_Ch1_Output_Data_1 := V680_Ch1_DeviceVariables.OutputData1;
N1_Ch1_Output_SID := V680_Ch1_DeviceVariables.OutputSID;
N1_Ch1_Select_UID := V680_Ch1_DeviceVariables.SelectUID;

```


付録

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。

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. 次の□「ライブラリの属性、FB や FUN の属性を参照する方法 (P.81)」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. □「ライブラリの属性、FB や FUN の属性を参照する方法 (P.81)」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。

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

(a) 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

The screenshot shows the 'Library Reference' window in Sysmac Studio. It contains a table with the following columns: Library Name, Name Space, Version, Creator, Company Name, Creation Date, Update Date, and Comment. The table lists several libraries, including 'OmronLib_MC_Toolbox_V1.1' and 'DeadBand (OmronLib_MC_Toolbox)'. Red boxes and labels point to specific fields: (1) Library Name, (2) Version, (3) Creator, (4) Comment, (5) FB/FUN Name, (6) Name Space, (7) FB/FUN Version, (8) FB/FUN Creator, and (10) FB/FUN Comment.

ライブラリ名	名前空間	バージョン	作成者	会社名	作成日時	更新日時	コメント
OmronLib_MC_Toolbox_V1.1		1.1.0	OMRON Corporation	(c)OMRON Corporation 2015. All Right			This is MC Toolbox library. これはモーション制御ツールボックスライブラリです。
DeadBand (OmronLib_MC_Toolbox)	OmronLib_MC_Toolbox	1.1.0	OMRON Corporation		03/16/2015	08/10/2015	No.00006 The DeadBand function block controls a deadba...
FirstOrderlag (OmronLib_MC_Toolbox)	OmronLib_MC_Toolbox	1.1.0	OMRON Corporation		04/01/2015	08/10/2015	No.00005 The FirstOrderLag function block processes a fir...
LeadLag (OmronLib_MC_Toolbox)	OmronLib_MC_Toolbox	1.1.0	OMRON Corporation		04/01/2015	08/10/2015	No.00005 The LeadLag function block performs phase lead...
PIDFeedFwd (OmronLib_MC_Toolbox)	OmronLib_MC_Toolbox	1.1.0	OMRON Corporation		04/01/2015	08/10/2015	No.00003 The PIDFeedFwd function block performs PID fe...

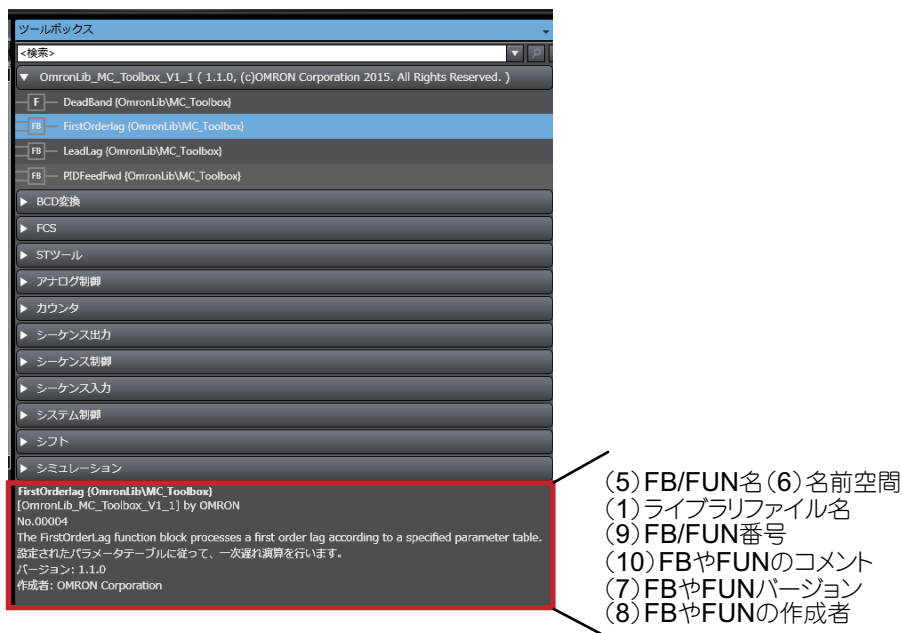
Below the table, there is a section for 'Name Space - Usage Declaration' and a table for 'Input/Output' details.

入出力	名称	入出力	データ型	エッジ	初期値	保持	コンスタント	コメント
外部変数	Enable	入力	BOOL	エッジなし	False	☐	☐	
	InCalc	入力	LREAL	エッジなし	0.0	☐	☐	
	Kp	入力	LREAL	エッジなし	1.0	☐	☐	
	TimeConst	入力	LREAL	エッジなし	1.0	☐	☐	
	SampTime	入力	LREAL	エッジなし	1.0	☐	☐	
	Enabled	出力	BOOL	エッジなし		☐	☐	
	CalcRslt	出力	LREAL	エッジなし		☐	☐	

(b) ツールボックス

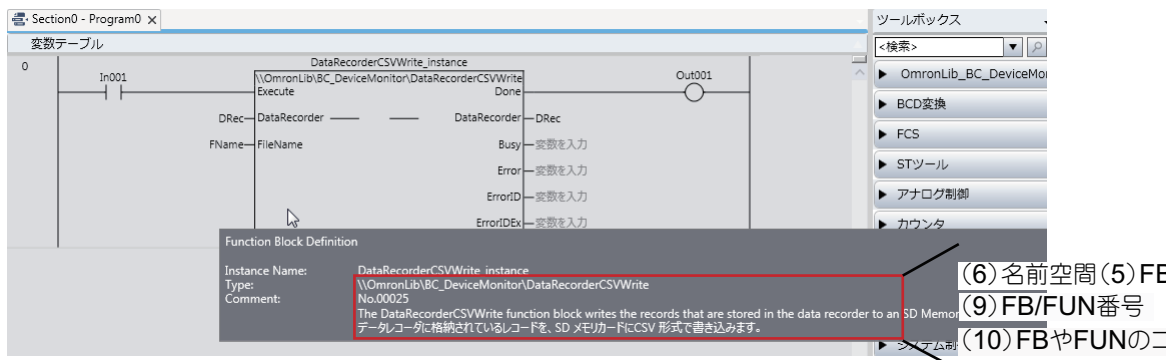
FB や FUN を選択すると、ツールボックスの下部にライブラリ情報が表示されます。

(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。



(c) 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。



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。

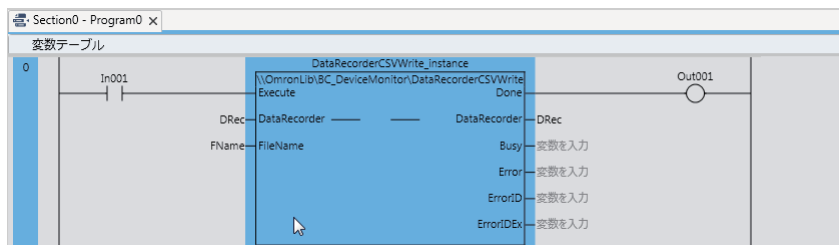
下記はカスタマイズする項目の例です。

- 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

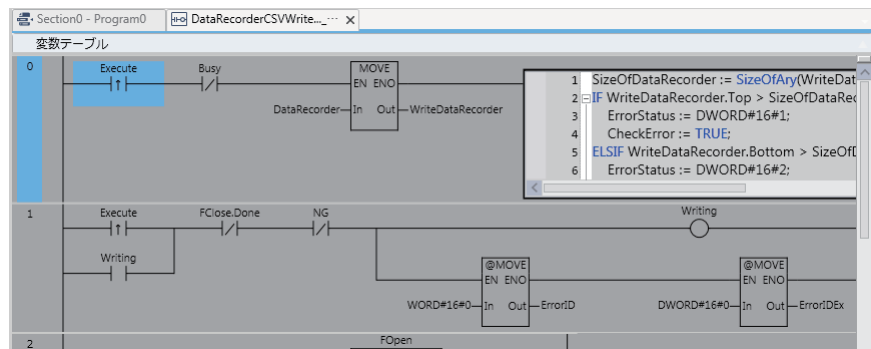
ただし、ソースコードが参照できるのは、FB や FUN の個別仕様に記載されているライブラリ情報のソースコード公開／非公開が「公開」になっているものだけです。

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。

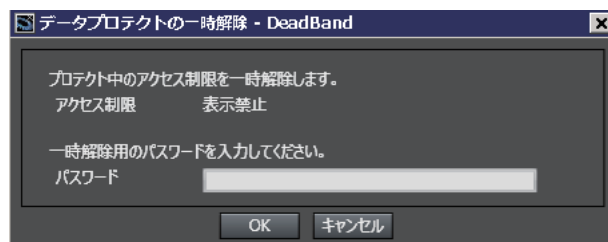


2 ダブルクリックまたは右クリックしメニューの[下の階層へ]を選択します。ソースコードが表示されます。



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。



- 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