



スマートリス CANopen 通信



《ご注意》

- 本製品の取扱いは、作業に熟練した方が行ってください。
また、ご使用に先立ち取扱説明書をよくお読みください。
- この取扱説明書は、実際にご使用になるお客様までお届けください。
- この取扱説明書は、必ず保管いただくようお願いいたします。

【はじめに】安全に関するご注意


- ・ご使用(据付、運転、保守・点検など)の前に、必ずこの取扱説明書とその他の付属書類をすべて熟読し、正しくご使用ください。機器の知識、安全の情報そして注意事項のすべてについて習熟してからご使用ください。
- ・お読みになった後は、お使いになる方がいつでも見られるところに必ず保管してください。
- ・この取扱説明書では、安全注意事項のランクを「危険」「注意」として区分してあります。



取扱いを誤った場合に、危険な状況が起こりえて、死亡または重傷を受ける可能性が想定される場合。



取扱いを誤った場合に、危険な状況が起こりえて、中程度の傷害や軽傷を受ける可能性が想定される場合および物的損害だけの発生が想定される場合。

なお、に記載した事項でも、状況によっては重大な結果に結びつく可能性があります。いずれも重要な内容を掲載していますので必ず守ってください。

危険

- 運搬、設置、配線、運転・操作、保守・点検の作業は、専門知識と技能を持った人が実施してください。感電、けが、火災、装置破損のおそれがあります。
- 人員輸送用装置に使用される場合には、装置側に安全のための保護装置を設けてください。暴走、落下による人身事故や、装置破損のおそれがあります。

【はじめに】取扱説明書の見方・目次

本取扱説明書は、smartris ドライバの CANopen 用の説明書です。

ドライバの取扱いについては、smartris ドライバ取扱説明書 (No. DM1802)、また、モータ部・ギヤ部の取扱いについては、smartris ギヤモータ取扱説明書 (No. DM1801) を参照してください。

目次

【はじめに】安全に関するご注意	1
【はじめに】取扱説明書の見方・目次	2
【1】概要	1-1
1-1 本書の目的	1-1
1-2 安全に関する注意事項	1-1
1-3 認証	1-2
1-4 スタートアップ	1-3
1-5 用語と略称	1-4
【2】CANopen 運転	2-1
2-1 CANopen ネットワークのトポロジー	2-1
2-2 クライアントサーバ	2-2
2-3 EDS ファイル (Electronic Data Sheet)	2-2
2-4 オブジェクトディクショナリ	2-2
2-5 システムの特長	2-3
2-6 CANopen 通信オブジェクト (COB)	2-4
2-7 SDO プロトコル	2-7
2-8 PDO プロトコル	2-10
2-9 エマージェンシメッセージ (EMCY)	2-29
2-10 SYNC プロトコル	2-36
2-11 エラーコントロールプロトコル	2-36
2-12 CAN 通信エラー	2-39
2-13 ネットワークマネジメント (NMT)	2-41
2-14 ストアとリストア	2-45
2-15 識別子	2-48
2-16 プロファイル DSP402	2-49
2-17 運転モード	2-51
【3】測定単位の変換	3-1
3-1 単位変換パラメータ	3-2

【はじめに】取扱説明書の見方・目次

【4】セーフティ	4-1
4-1 セーフティオブジェクト	4-1
4-2 セーフティステートの状態機械 DSP402	4-2
【5】CANopen オブジェクトディクショナリ	5-1
5-1 ゼネラルオブジェクト (DS301)	5-1
5-2 メーカー固有オブジェクト(設定パラメータ)	5-15
5-3 メーカー固有オブジェクト(運転時間モニタリングデータ)	5-32
5-4 DSP402 プロファイルオブジェクト	5-53
【6】CANopen 運転モード	6-1
6-1 運転モード	6-1
6-2 位置プロファイルモード	6-1
6-3 速度プロファイルモード	6-2
6-4 速度プロファイルのオブジェクト	6-9
6-5 トルクプロファイルモード	6-19
6-6 トルクプロファイルのオブジェクト	6-24
【7】CANopen オブジェクトリスト	7-1
【8】機能	8-1
8-1 過速度検出	8-1
8-2 デジタル I/O	8-1
8-3 他の機能	8-3
【9】診断	9-1
【10】サンプルプログラム	10-1
10-1 速度プロファイル手順の設定	10-1
10-2 バージョンの読み込み	10-3
10-3 PDO を介しての制御	10-3
10-4 TPDO (RPDO) のリマッピング手順	10-5
【11】各種の設定	11-1
11-1 初期設定	11-1
11-2 ハートビート・メカニズム	11-12
11-3 位置のモニタリング	11-14
【12】保証	12-1

【1】 概要

1-1 本書の目的

本書には、本製品(スマートリス)を安全に設置して運転することに関し記載されています。
使用に先立ち、本書をよく読んでください。
本製品を安全に使用するため、記載してある安全な取扱いと警告に従ってください。
本書を、常に使用できる状態で保管してください。

⚠ 注意

- 本書は、指定された作業者が、本製品を使用するための取扱説明書です。
- 本書、CANopen に関する取扱説明書です。

本書は、以下の条件と制限で記載されています。

- ・ 本書は、弊社に属している情報の記載があります。
- ・ CANopen ネットワーキングの実施のため、サーボドライブのユーザのための情報が記載されています。
- ・ 本書に含められている文書と図は、説明と参照用です。
- ・ 企業および個々の名前と例において使われたデータは、言及されない限り架空です。
- ・ 本書の記載内容に関して、将来予告なしに変更することがありますので承願います。

1-2 安全に関する注意事項

本製品を安全に動作させるためには、本書に記載されている安全な手順を実施することです。
本製品と周辺装置が動作している場合、作業者和その周辺場所を保護することになります。

⚠ 注意

- システム全体で各種のスイッチが正常に動作し、警告表示がないか確認する必要があります。
- スタートアップ前に、配線のチェック、ドライバの損傷がないか確認してください。
- 仕様外の電圧の印加やケーブルの逆接続は、ドライバが故障する可能性があります。
- 電源オンの状態や運転中にケーブルの着脱をしないでください。
- 作業者は、法令と基準に適合する安全な据付けを行う責任があります。

注) 設置する前に、本章をよく読んでください。

本製品は、静電気に敏感な部品が含まれていますので、不適切に扱うと故障します。
静電気故障を防ぐために、高絶縁のプラスチックフィルムや合成繊維に触れないようにしてください。

導電性の製品の上に設置し、静電気を放電するように接地してください。

作業者の負傷や製品故障の原因となる危険を避けるため、カバーや制御盤の扉を閉めてください。

⚠ 危険

- 電源オンの状態で、ドライバの配線を外すことや接続をしないでください。アーク放電が原因となる故障や感電事故を避けるためです。
- モータ停止時にもケーブルには、高電圧が印加されています。ケーブルを外さないでください。
- 電源オフ後に電源接続を外す場合、装置に触れるのは、最低 1 分間待ってください。コンデンサや接点が帯電しているためです。
装置に触れる前に測定器で電圧測定することを推奨します。

⚠ 注意

- 人体への危害、製品や装置の故障を避けるための注意です。
- DC 電源は、本書に記載してある電圧の仕様範囲内で印加してください。
 - 本製品を電源ケーブルに接続する場合、危険な高圧線と分離し安全基準に適合した絶縁を行ってください。
 - 本製品を運転する前に、本書に従った手順で設置されているか安全を確認してください。
 - セーフトルクオフ機能を使用する場合は、事前に動作を確認してください。
 - もし装置の火災が発生した場合、消火するため装置への直接の放水をしないでください。

⚠ 注意

- 製品が正常に運転できること、理解することの確認情報です。

安全対策は、基準に基づき作業者と機械のために行う必要があります。

1-3 認証

(1) CE 適合

本製品は、以下の EC 指令に適合していることが認証機関で証明されています。

- ・ 機械指令 (2006/42/EC) ・ RoHS 指令 (2011/65/EU)
- ・ EMC 指令 (2014/30/EU) ・ WEEE 指令 (2012/19/UE)

(2) セーフティ

安全に関する以下の EN 規格に適合しています。

- ・ **EN 61800-5-1** : Adjustable speed electrical power drive systems - Part 5-1: Safety requirements
– Electrical, thermal and energy.

(3) UL 認証

UL61800-5-1 Standard for Adjustable Speed Electrical Drive Systems – Part 5-1: Safety Requirements - Electrical, Thermal and Energy

USL: Approval according standards of the United States in accordance with UL 61800-5-1

CSL: Approval according to national Canadian standards in accordance with C22.2 No.274

【1】概要

(4) EMC 要求

本製品は、エミッションとイミュニティの条件で " 第 2 環境 " (産業環境) のカテゴリーの要求を満たしています。

- EN 61800-3 : Adjustable speed electrical power drive systems - Part 3: EMC requirements and specific test methods.

(5) 機能安全適合 (STO)

本製品は、機能安全に適した 2 チャネルの STO (セーフトルクオフ) 入力が備わっています。

その機能は、PWM 出力を停止し、ドライバを安全にトルクオフさせることです。

設計回路は、TÜV Süd でテストし認証されています。

本製品の設計回路で、" セーフトルクオフ " の機能安全は、以下の EN 規格に適合しています。

- EN61508 : Functional safety of electrical/electronic/programmable electronic safety-related systems
- EN61800-5-2 and category : Adjustable speed electrical power drive systems - Part 5-2: Safety requirements - Functional
- EN ISO 13849-1:2015 : Safety of machinery - Safety-related parts of control systems - Part 1 : General principles for design.

サブシステムには、次の特性による安全条件が記載されています。

EN 13849-1	EN 61508	PFHD [1/h]
PLe	SIL3	-

1-4 スタートアップ

本製品が EC 指令の規格に対応して設置されているか確認されるまで、EMC 指令としてスタートアップすることが、禁止されています。

(1) 正しい使い方

本製品は、永久磁石を用いた同期形のサーボモータの駆動に適用できます。

(機械やシステムのフィードバックシステムに対応したサーボモータ)

本製品は、産業用途に使用されることで認証されています。なお、住宅地域で使用する場合は、追加 EMC 対策が必要になります。

お客様は、最終製品の危険分析を準備しておく必要があります。

注意

- 産業用以外の用途で使用するお客様は、最初に弊社の承認が必要です。

(2) 不適切な使い方

本製品は、同期形サーボモータ以外のモータに対応していません。また、フィードバックシステムが適合していないモータにも対応できません。

なお、燃えやすい物、可燃しやすい気体、粉塵などの危険地域への設置は、火災、爆発の誘因となります。これらの環境に、本製品のドライバ、モータを設置し使用しないでください。

1-5 用語と略称

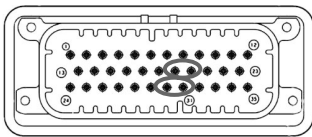
略 称	名 称	説 明
CAN	Controller Area Network	コントロールエリアネットワーク
CiA	CAN in Automation	CAN のユーザおよびメーカの国際的組織 (ドイツ)
COB	Communication Object	通信オブジェクト、CAN ネットワークのトランスポートユニット
COB-ID	Communication Object Identifier	通信オブジェクトの識別子
DS301	Profile 301 standardizes	CANopen 通信プロファイルの標準規格書
DSP402	Profile 402 standardizes	CANopen デバイスプロファイルの標準規格書
EDS	Electronic Data Sheet	電子データシート
EMCY	Emergency Object	エマージェンシオブジェクト
EMC	Electromagnetic Compatibility	電磁適合性
HMI	Human Machine Interface	ヒューマン・マシン・インタフェース
I/O	Input/Output	入力 / 出力
LSB	Least Significant Bit/Byte	最下位ビット / バイト
MASTER	Master Controller	ドライバの制御と通信を行う装置
MSB	Most Significant Bit/Byte	最上位ビット / バイト
MSM	Macro State Machine	マクロ状態機械
NMT	Network Management	ネットワーク管理
Node ID	Node Identifier	ネットワークのデバイスに割付けられるノードアドレス
OD	Object Dictionary	オブジェクトディクショナリ
PDO	Process Data Object	プロセスデータオブジェクト
PDS	Power Drive System	パワードライバシステム
REG	Register	レジスタ
RO	Read-Only Access.	読出し専用のアクセス
RPDO	Receive (Incoming) PDO	PDO 受信
RW	Read/Write Access.	読出し / 書込みのアクセス
RX	Receiver	マスタコントローラから送信し、スレーブが受信するメッセージ
SD	Servo Drives	smartris ドライバ
SDO	Service Data Object	サービスデータオブジェクト
STO	Safe Torque Off	セーフトルクオフ
TX	Transmitter	スレーブから送信し、マスタコントローラが受信するメッセージ
TPDO	Transmit (Outgoing) PDO	PDO 送信

【2】 CANopen 運転

CANopen は、通信プロトコルで、オートメーションに使用され、システムに組み込み用のデバイスプロフィールです。

CANopen 規格は、アドレス方式、通信プロトコル、およびデバイスプロフィールで定義されたアプリケーション層を利用します。

2-1 CANopen ネットワークのトポロジー



X1 コネクタ

表 2-1 CANopen 信号

ピン No.	信 号	内 容
19	GND_CAN/ GND_RS485	GND (CAN または RS485Modbus 通信)
20	CAN_T/RS485_T	120 Ω 終端抵抗 (CAN_H/RS485+ へ接続)
30	CAN_L/RS485-	CAN_L または RS485- の接続
31	CAN_H/RS485+	CAN_H または RS485 + の接続

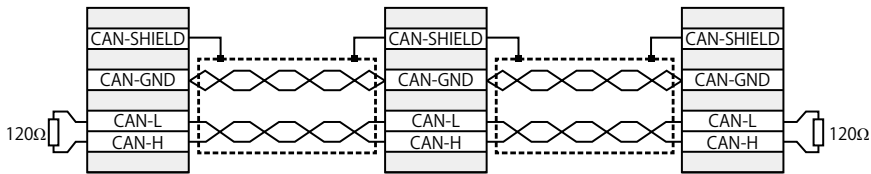


図 2-1 CANopen ネットワーク

注) ネットワークの両端は 120 Ω の抵抗で終端してください。
smartris ドライバには 120 Ω の終端用抵抗が内蔵されていて、X1 コネクタの 20 ピン (CAN_T/RS485_T) と 31 ピン (CAN_H/RS485+) を外部で接続することにより、終端抵抗が接続されます。

■ CANopen ボーレートとノード ID

CiA DS301 ver.4.02 と DSP402 ver.2.0 の指令に準拠しています。

- CANopen 通信のボーレート：50K、125K、500K、800K、1000K (初期設定：1000Kb)
- ノード ID (初期設定：1)

⚠ 注意

CAN バスに複数のドライバが接続されている場合、異なるノード ID に設定する必要があります。

注) 詳細は、11 章 各種の設定の「ノード ID の設定方法、ボーレートの変更方法」を参照してください。

2-2 クライアントサーバ

CAN のマスタ(クライアント)は、各ノードに対してリクエストを行うコントローラです。
CAN のスレーブ(サーバ)は、CAN のマスタから送信されたリクエストに応答します。
CAN プロトコルは、シングルマスタとマルチマスタのネットワークをどちらも許容しています。
(マスタは、コントローラまたは PLC、スレーブは、smartris ドライバ)

全てのスレーブ(smartris ドライバ含む)は、競合しないように 1 ～ 127 のノード ID を必ず割り当てる必要がありますが、ネットワーク上の CAN のマスタは、ノード ID を割り当てる必要がありません。

スレーブは、緊急時以外、要求されていないメッセージを送信することがありません。
各スレーブに設定されているノード ID 宛てにマスタからリクエストされた場合、またはすべてのスレーブへリクエストされたメッセージにのみ応答して送信します。

⚠ 注意

2 台のドライバに同じノード ID を設定した場合、CANopen ネットワークが正常に動作しません。

2-3 EDS ファイル (Electronic Data Sheet)

EDS ファイルは、デバイス(smartris ドライバ)に関する内容の標準フォーマットです。
次の情報が含まれています。

- ・ファイルのプロパティ (名称、バージョン、リリース日など)
- ・一般的なデバイス情報 (製造メーカー名、コード)
- ・デバイス名、タイプ、バージョン
- ・サポートされるボーレートとブートアップオプション
- ・サポートされるオブジェクトと属性の記述

2-4 オブジェクトディクショナリ

デバイスプロファイルの最も重要な部分は、オブジェクトディクショナリに記述されています。
オブジェクトディクショナリとは、CAN のネットワーク経由でアクセス可能なオブジェクトをグループ化したもので、インデックスとサブインデックスで構成されます。
ディクショナリ内の各々のオブジェクトは、16 ビットのインデックスで区別されています。
一般的なオブジェクトディクショナリの構造を次に示します。

Index、Sub-index (HEX)	オブジェクト (シンボルネーム)	名 称	タイプ	属 性	M/O
--------------------------	---------------------	-----	-----	-----	-----

- ・ **Index、Sub-index** : インデックスは、オブジェクトディクショナリ内の各オブジェクトを区別するため、1 つ 1 つ異なる値が割り当てられています。
サブインデックスは、インデックスを更に細分化して区別するものです。
- ・ **オブジェクト** : オブジェクト名を含みます。オブジェクトの種類を示しています。
- ・ **名称** : オブジェクトの内容を示しています。
- ・ **タイプ** : オブジェクトのデータタイプを示しています。
例 : ブーリアン、浮動小数点、符号なし整数、符号付き整数など
- ・ **属性** : オブジェクトのアクセス権を示しています。
例 : rw (リード、ライトのアクセス)、wo (ライト)、ro (リード)、const (リード、一定値)
- ・ **M/O** : オブジェクトが強制、またはオプションを示します。

【2】 CANopen 運転

次の表に標準のオブジェクトディクショナリを示します。

表 2-2 オブジェクトディクショナリ (標準)

インデックス (HEX)	オブジェクト
0000	未使用
0001 ~ 001F	静的データタイプ
0020 ~ 003F	複合データタイプ
0040 ~ 005F	メーカ固有の複合データタイプ
0060 ~ 007F	デバイスプロファイル固有の静的データタイプ
0080 ~ 009F	デバイスプロファイル固有の複合データタイプ
00A0 ~ 0FFF	予約領域
1000 ~ 1FFF	通信プロファイルエリア
2000 ~ 5FFF	メーカ固有プロファイルエリア
6000 ~ 9FFF	標準デバイスプロファイルエリア
A000 ~ FFFF	予約領域

2-5 システムの特長

- ・ **サービスデータオブジェクト (SDO)** : SDO メッセージは、全てのオブジェクトディクショナリに対してリード・ライトアクセスで使用可能です。通常、SDO はデバイスの初期設定時に使用されます。
- ・ **プロセスデータオブジェクト (PDO)** : 目標位置、目標速度等のリアルタイムデータ送信や入出力の定義は、PDO メッセージにより実行されます。データは、最大 8 バイトのデータブロックで、最大 4 つの TPDO (Transmit PDO) および RPDO (Receive PDO) がドライバで送受信されます。
- ・ **ネットワークマネジメント (NMT)** : NMT 状態機械は、CANopen デバイスの通信状態を表します。
- ・ **エマージェンシオブジェクト** : エマージェンシメッセージは、デバイス内部で致命的なエラーが発生したことがトリガとなり、関連するアプリケーションデバイスから優先順位の高いメッセージとしてデバイスから送信されます。
- ・ **SYNC メッセージ** : SYNC メッセージは、各ノードで同期を取るためのメッセージです。PDO 通信で使うことが可能です。
- ・ **ノードガードプロトコル** : NMT マスタコントローラは、周期的なクエリをスレープへ送信し、スレープはそれに応答します。マスタおよびスレープの通信状態を監視するための機能です。NMT マスタコントローラは、定義された時間内にスレープへクエリを送信します。
- ・ **ハートビート・ファンクションプロトコル** : ネットワークのノードからハートビートメッセージが ms 間隔で CAN バスへ自動送信されます。
ハートビートは、ネットワーク上のノードの生存有無を検出するのに有効です。
- ・ **イベントタイマ** : イベントタイマは、PDO を周期的に送信するために使用されます。
注意点として、各ノード自身に設定するため、他の CANopen デバイスとは同期されません。
- ・ **パラメータのストアとリストア** : スタアおよびリストア処理を実行するとパラメータは不揮発性メモリの EEPROM へセーブされます。(通信、メーカ固有、デバイスプロファイル関連のパラメータ)
- ・ **状態機械** : ドライバは、DSP402 による状態機械に沿って制御を実行します。
- ・ **運転モード** : DSP402 プロファイルに従って、異なる運転モードを使用できます。
ドライバは、ハードウェアの入出力端子で制御できるメーカ固有の運転モードをサポートしています。

注) smartris ドライバは、CiA DS301 ver.4.02 と DSP402 ver.2.0 指令に準拠しています。

2-6 CANopen 通信オブジェクト (COB)

通信オブジェクトは、DS301 CANopen 通信プロファイルで標準化されています。
全ての通信オブジェクトは、機能によって 4 グループに分類することができます。

特殊オブジェクト

エマージェンシ(EMCY)
SYNC

PDO

TPDO-1 RPDO-1
TPDO-2 RPDO-2
TPDO-3 RPDO-3
TPDO-4 RPDO-4

通信オブジェクト

ネットワークマネジメント

NMT (サービス)
NMT (ノードガーディング)
NMT (ハートビート)

SDO

Tx SDO
Rx SDO

図 2-2 CANopen 通信オブジェクト (COB)

- 特殊オブジェクト (CAN メッセージ制御用)
エマージェンシオブジェクト (EMCY) : デバイスまたは周辺機器のエラー信号
SYNC オブジェクト (同期オブジェクト) : ネットワークデバイスとタイミングを同期
- ネットワークマネジメント (NMT)
NMT (サービス) : ネットワーク制御およびネットワーク状態の初期化
NMT (ノードガーディング) : マスタとスレーブの互いの通信状態をモニタリング
NMT (ハートビート) : スレーブの通信状態をモニタリング
- PDO (プロセスデータオブジェクト) プロセスデータのリアルタイム送受信
TPDO-1 ~ 4 : スレーブからマスタへ送信される PDO
RPDO-1 ~ 4 : マスタからスレーブへ送信される PDO
- SDO (サービスデータオブジェクト) オブジェクトへのリード・ライトアクセス
Rx SDO : マスタからスレーブへ送信する SDO
Tx SDO : スレーブからマスタへ送信する SDO

【2】 CANopen 運転

マスタコントローラとドライバ間の通信は、以下の通信オブジェクト (COB) が使用できます。

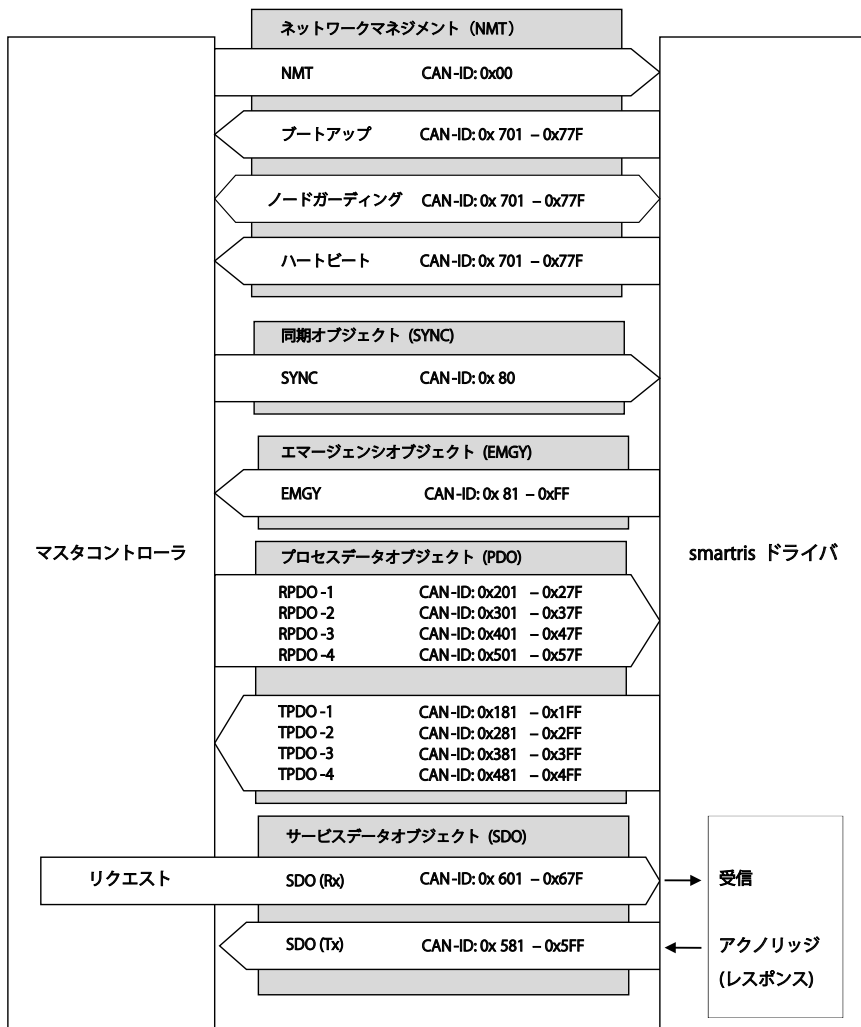


図 2-3 マスタコントローラと smartris ドライバ間の通信

注) 詳細な情報は、「CiA DS301 Standard」を参照してください。

【2】 CANopen 運転

CANopen では、一部標準化されたドライバ内のパラメータへアクセスすることができます。

(例：目標速度プロファイルや加速プロファイル等)

固有の番号(インデックスとサブインデックス)は、各パラメータ(CANopen オブジェクト)に割り付けられています。調整可能な全パラメータは、オブジェクトディクショナリ(OD)に含まれます。

CAN バスを介して、CANopen のオブジェクトにアクセスする方法は以下の 2 つです。

- ・ サービスデータオブジェクト (SDO) を使用してアクセス
全パラメータへアクセスが可能な SDO を使用します。
ドライバは、SDO を受信すると、応答(アクノリッジ)として SDO を送信するアクセスタイプです。

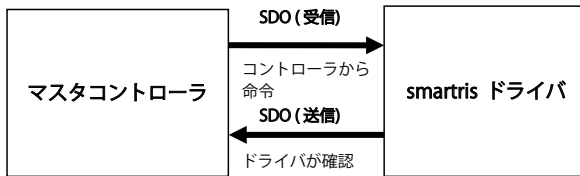


図 2-4 SDO 通信

- ・ プロセスデータオブジェクト (PDO) を使用してアクセス
一部のパラメータへアクセスが可能な PDO を使用します。
ドライバは、RPDO を受信しても、応答(アクノリッジ)しません。また、TPDO を送信してもマスタからの応答を必要としません。

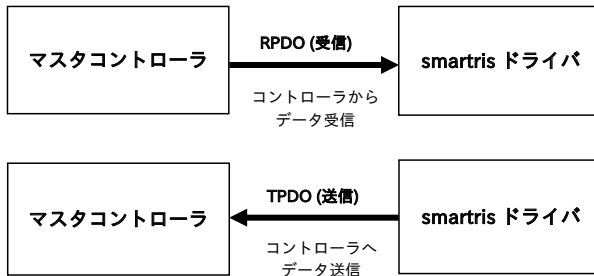


図 2-5 PDO 通信

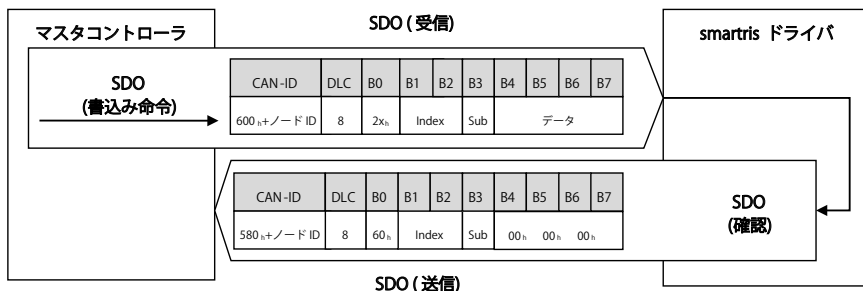
【2】 CANopen 運転

2-7 SDO プロトコル

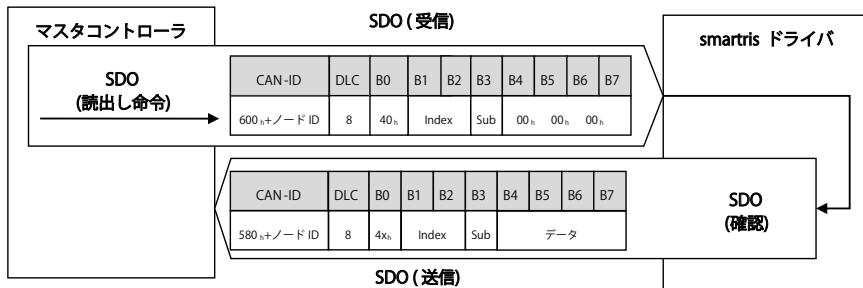
SDO プロトコルは、デバイスのパラメータ設定(オブジェクトディクショナリへアクセス)時の書き込みと読み出しに使用されます。通信は、常に SDO クライアントから開始されます。

SDO クライアント(マスタコントローラ、PC アプリケーション、PLC)からの要求により、ドライバは、データを作成して送信します。次に示す SDO の通信プロトコルがサポートされています。

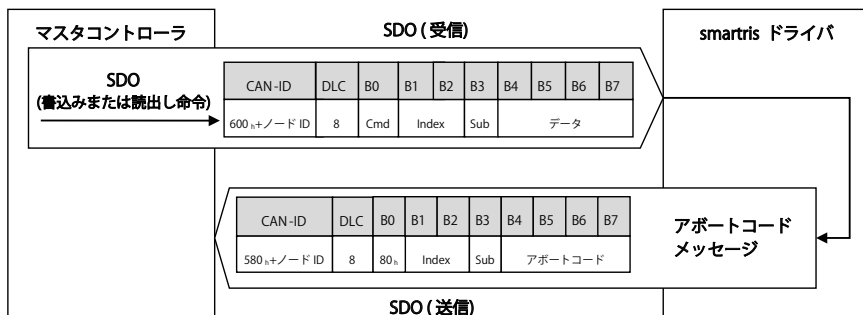
・ SDO ダウンロードプロトコル(書き込み)



・ SDO アップロードプロトコル(読み出し)



・ SDO アボートプロトコル



【2】CANopen 運転

■ SDO ダウンロードプロトコル(書込み)

SDO ダウンロードサービスは、パラメータ書込み時に使用され、以下のように定義されています。

SDO ダウンロードメッセージ構造

表 2-3 SDO ダウンロードメッセージ構造

COB-ID	応答 リクエスト	DLC	データ							
			D0	D1	D2	D3	D4	D5	D6	D7
0x600+ ノード ID	Rx	8	0x2x	Index		Sub- Index	データ LSB	データ	データ	データ MSB
0x580+ ノード ID	Tx	8	0x60	Index		Sub- Index	0x00	0x00	0x00	0x00

SDO ダウンロードメッセージ - データフィールド (D0)

表 2-4 SDO ダウンロードメッセージ・データフィールド

D0	内 容	データバイト数
0x22	ライトリクエスト (ドメインダウンロード開始)	-
0x23	ライトリクエスト (ドメインダウンロード開始)	4 バイト
0x27	ライトリクエスト (ドメインダウンロード開始)	3 バイト
0x2B	ライトリクエスト (ドメインダウンロード開始)	2 バイト
0x2F	ライトリクエスト (ドメインダウンロード開始)	1 バイト
0x60	ライトレスポンス (ドメインダウンロード開始)	-

■ SDO アップロードプロトコル(読出し)

SDO アップロードサービスは、パラメータ読出し時に使用され、以下のように定義されています。

SDO アップロードメッセージ構造

表 2-5 SDO アップロードメッセージ構造

COB-ID	応答 リクエスト	DLC	データ							
			D0	D1	D2	D3	D4	D5	D6	D7
0x600+ ノード ID	Rx	8	0x40	Index		Sub- Index	0x00	0x00	0x00	0x00
0x580+ ノード ID	Tx	8	0x4x	Index		Sub- Index	データ LSB	データ	データ	データ MSB

SDO アップロードメッセージ - データフィールド (D0)

表 2-6 SDO アップロードメッセージ・データフィールド

D0	内 容	データバイト数
0x40	リードリクエスト (ドメインアップロード開始)	-
0x43	リードレスポンス (ドメインアップロード開始)	4 バイト
0x47	リードレスポンス (ドメインアップロード開始)	3 バイト
0x4B	リードレスポンス (ドメインアップロード開始)	2 バイト
0x4F	リードレスポンス (ドメインアップロード開始)	1 バイト

【2】 CANopen 運転

■ SDO アボートコード (Abort Code)

SDO 通信で書き込みまたは読み出しに失敗した場合、ドライバは、通常の SDO メッセージでなく SDO アボートコードを自動的に送信します。

アボートメッセージは、以下のように定義されており、異常内容を示すアボートコードが含まれています。

SDO アボートコード構造：

表 2-7 SDO アボートメッセージ構造

COB-ID	リクエスト 応答	DLC	データ							
			D0	D1	D2	D3	D4	D5	D6	D7
0x580+ ノード ID	Tx	8	0x80	Index	Sub Index	アボートコード				

アボートコードは、次の表で定義されます。符号なし 32 ビットで符号化されます。

表 2-8 SDO アボートコードと定義

アボートコード	定 義
0504 0000h	SDO プロトコルタイムアウト
0504 0001h	無効、不明のクライアント / サーバ指令
0504 0002h	禁止のブロックサイズ (ブロックモードのみ)
0504 0003h	禁止のシーケンス番号 (ブロックモードのみ)
0504 0004h	CRC エラー (ブロックモードのみ)
0504 0005h	メモリ不足
0601 0000h	サポートされていないオブジェクトへのアクセス
0601 0001h	書き込み専用オブジェクトへの読み込み
0601 0002h	読み出し専用オブジェクトへの書き込み
0602 0000h	オブジェクトディクショナリに存在していないオブジェクト
0604 0030h	設定可能な範囲外の数値を書込み
0604 0041h	指定したオブジェクトは PDO ヘマッピングできない。
0604 0042h	マッピングするオブジェクトの数およびデータ長が PDO のデータ長を超過
0604 0043h	パラメータの不一致理由
0604 0047h	デバイスの内部不一致
0606 0000h	ハードウェアエラーによるアクセス失敗
0607 0010h	データタイプ不適合 (サービスパラメータのデータ長が不整合)
0607 0012h	データタイプ不適合 (サービスパラメータのデータ長が長い)
0607 0013h	データタイプ不適合 (サービスパラメータのデータ長が短い)
0609 0011h	サブインデックスが存在しない。
0609 0030h	パラメータ無効の値 (ダウンロードのみ)
0609 0031h	パラメータの値が大きい。(ダウンロードのみ)
0609 0032h	パラメータの値が小さい。(ダウンロードのみ)
0609 0036h	最大値が最小値より少ない。
060A 0000h	ステートで許可されていない動作
060A 0023h	リソースが使用不可 (SDO 接続)
0800 0000h	ゼネラルエラー
0800 0020h	データがアプリケーションに送信または、ストアできない。
0800 0021h	データが送信されないかアプリケーションに送信または、ストアできない。 (ローカル制御による)
0800 0022h	データが送信されないかアプリケーションに送信または、ストアできない。 (現在のデバイス状態による)
0800 0023h	オブジェクトディクショナリのダイナミックジェネレーションが失敗したかオブジェクトディクショナリが存在しない。 例：オブジェクトディクショナリをファイルから生成時に、ファイルエラーのため生成に失敗
0800 0024h	データ使用不可
0800 0030h	データの書き込み不可 (電源リセットが必要)

2-8 PDO プロトコル

PDO プロトコルは、複数のノード間でリアルタイムにデータを送受信する場合に使用されます。PDO には SDO のようなハンドシェイクがありません。データは 1 ～ 8 バイトに制限されて送信します。

- ・ **PDO 通信パラメータの設定**：PDO で設定した COB-ID（送信タイプ）を含みます。
PDO 通信に関するパラメータは、RPDO が 1400h, 1401h, 1402h, 1403h、TPDO が 1800h, 1801h, 1802h, 1803h で設定可能です。

次の表に通信パラメータ設定を示します。

表 2-9 通信パラメータ

Sub-Index	名 称	データタイプ	内 容
00h	エントリ数	UNSIGNED8	最上位数のサブパラメータを示しています。
01h	COB-ID	UNSIGNED32	使用する CAN-ID を指定します。 PDO の有効/無効選択の機能を有したビットを含みます。
02h	送信タイプ	UNSIGNED8	トリガ（TPDO の場合）または受信（RPDO の場合）条件を決定します。
03h	禁止時間	UNSIGNED16	TPDO が再び送信可能な時間を [ms] で表します。
04h	エントリ適合性	UNSIGNED8	予約領域（使用不可）
05h	イベントタイマ	UNSIGNED16	TPDO では送信間隔 [ms]、RPDO ではタイムアウトをそれぞれ決定します。

- ・ **PDO マッピングパラメータの設定**：オブジェクトディクショナリに記載されたマッピング可能なオブジェクトを RPDO が 1600h, 1601h, 1602h, 1603h、TPDO が 1A00h, 1A01h, 1A02h, 1A03h で設定可能です。

PDO マッピングには、以下の制限があります。

- ・ 各々の PDO は、4 つ以内のオブジェクトでマッピングをすること
- ・ 各々の PDO のデータ長は 64 ビット（8 バイト）を超えないこと

ドライバには出荷時設定として、複数のオブジェクトが既にマッピングされています。

ネットワークが初期化された後、これらのマッピングを変更することができます。

ドライバが “Pre-operational” ステート（NMT 状態機械）のときにマッピングを変更してください。

注 意

“Operational” ステートでもマッピングの変更が可能です。が推奨していません。

“Operational” ステートでマッピングを変更する場合は、データの一貫性が必要です。

注）PDO プロトコルは、“Operational” ステート（NMT 状態機械）で使用できます。

【2】 CANopen 運転

■ COB-ID

PDO の最も重要な通信パラメータは、COB-ID (CAN 識別子) で、通信オブジェクト識別子とも呼ばれます。データを識別し、バス上でアクセスの優先順位を決定します。

CAN メッセージはマスタコントローラ (送信ノード) から送信されますが、すべての CAN メッセージは CAN プロトコルに従って各ノード (デバイス) へ送信されます。

COB-ID は、通信パラメータの Sub-index 1 に定められています。32 ビット値でコード化され、コントロールビットを含んでいます。

ビット 29 は、標準メッセージまたは拡張メッセージのいずれかを定義しています。ビット 29 が 0 の場合、次の 19 ビット (ビット 10 からビット 28) が無視され、残りの 10 ビット (ビット 0 からビット 9) が、PDO として使用される CAN-ID とみなされます。つまり、CAN 標準ベースのフォーマットを使用して通信フレームが送信されます。ビット 29 が 1 の場合、29 ビット (ビット 0 からビット 28) の CAN-ID としてみなされます。これは、PDO が CAN 拡張フォーマットで、通信フレームが送信されることを意味します。

ビット 30 は、CAN のリモートフレーム (RTR) の許可を示していますが、全ての CAN 実装でサポートされている機能ではありません。一部のリモートフレームの送信は、無効にすることができません。

ビット 31 は、PDO の送信および受信機能の有効/無効を切替えることができます。

無効に設定した場合、ノードは PDO を送信しません。同時に、他のノードから受信した PDO を処理することはなく、無視されます。

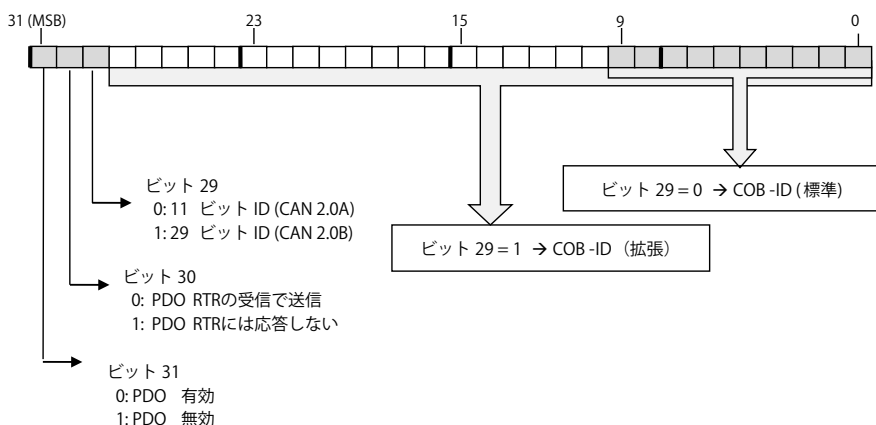


図 2-6 COB-ID ビット

■ PDO 通信

PDO 送信には、以下の方式があります。

- ・同期方式：SYNC メッセージをトリガとして、周期的または非周期的に PDO を送信します。
- ・非同期方式：デバイス内部のイベント (ステータスワードの値が変更) やタイマ、リモートリクエスト (RTR) をトリガとして、PDO を送信します。

以下に PDO の送信タイプについて示します。

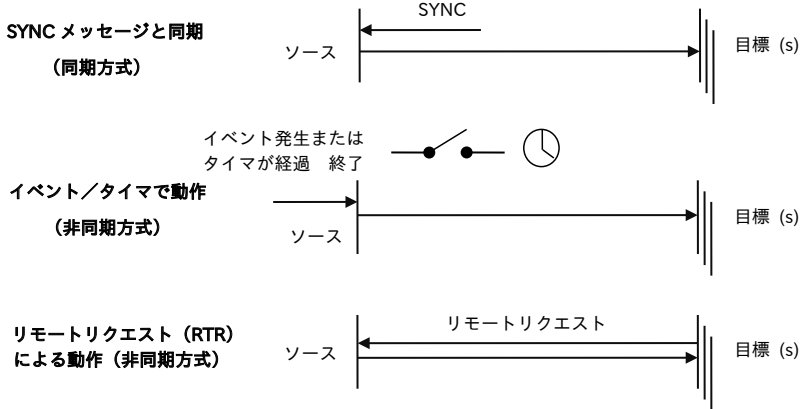


図 2-7 PDO 送信タイプ

(1) SYNC メッセージと同期

複数の CANopen デバイスを同期させるために SYNC (同期) オブジェクトは、周期的に送信されます。PDO メッセージの送信は、SYNC オブジェクトの受信でトリガされます。受信した SYNC メッセージの回数によって PDO を送信するトリガの条件を変更することが可能です。

SYNC オブジェクトの詳細は、「2-10 SYNC プロトコル」を参照してください。

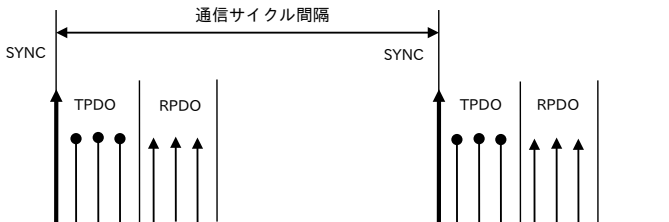


図 2-8 SYNC 送信タイプ

(2) イベント/タイマで動作

一般的に、この送信タイプはデバイスプロファイル、アプリケーションプロファイル、メーカー固有の特定アプリケーションのイベント発生によりトリガされ、メッセージを送信します。

smartris では、ステータスワードの値が変更されたタイミングをイベント発生とみなします。

イベントが発生しなかった場合でもイベントタイマに設定された時間が経過するとトリガされます。

【2】 CANopen 運転

(3) リモートリクエストによるポーリング

リモートリクエスト (RTR) によるポーリングをトリガとします。

PDO の送信は、RTR を受信することで開始されます。

PDO の送信タイプのパラメータは、どのようなメッセージをどのように取り扱うか等のトリガを設定することができます。各通信パラメータの Sub-index 2 で送信タイプの設定が可能です。

表 2-10 設定できる送信タイプ一覧

送信タイプ	周期	非周期	同期	非同期	RTR
0	-	○	○	-	-
1 ~ 240	○	-	○	-	-
241 ~ 251	予約領域				
252	-	-	○	-	○
253	-	-	-	○	○
254、255	-	-	-	○	○

■非周期的同期 (0)

送信タイプ 0 は、SYNC メッセージと同期しますが、周期的ではありません。

- ・RPDO は、SYNC メッセージを受信した後、処理されます。
- ・TPDO は、周期的な SYNC メッセージを受信する度にデータを読みみますが、メッセージを送信するためにはイベント発生が必要です。SYNC メッセージを受信した時にトリガされ、マッピングされたオブジェクトのデータは、メッセージ送信後に更新されます。

■周期的同期 (1 ~ 240)

送信タイプ 1 ~ 240 では、SYNC メッセージを n 回 ($n = 1 \sim 240$) 受信した後に PDO を送受信します。

- ・RPDO は、常に SYNC メッセージによってトリガされます。
- ・TPDO は、SYNC メッセージの直後に送信され、1 ~ 240 の間で設定された回数と同期して周期的に送信されます。1 ~ 240 の設定値は、PDO 送信のトリガに必要な SYNC メッセージの受信回数を意味します。

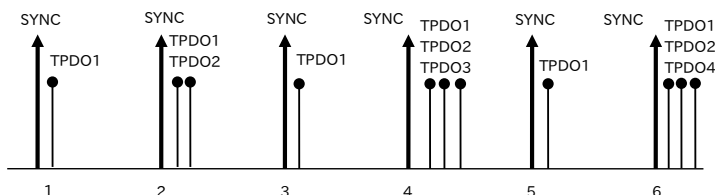


図 2-9 TPDO 送信タイプ例：1 ~ 240 (TPDO1=1, TPDO2=2, TPDO3=4, TPDO4=6)

■ RTR (252、253)

送信タイプ 252 と 253 は、リモートフレームのリクエスト (RTR) によって PDO が送信されます。これらの設定は、TPDO のみに適用可能です。

送信タイプ 252 は、他ノードと同期された SYNC メッセージを受信後、さらにリクエスト (RTR) を受信することで TPDO が送信されます。

【2】CANopen 運転

- ・ SYNC メッセージを受信することにより TPDO にマッピングされたデータの値が保持・トリガされます。(この時点では、TPDO が送信されません)
- ・ SYNC メッセージ受信後、リクエスト (RTR) を受信することで TPDO がマスタへ送信されます。

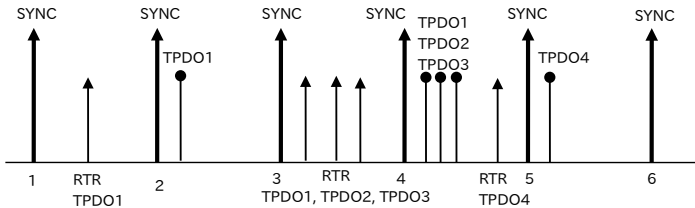


図 2-10 TPDO 送信タイプ 252

送信タイプ 253 は、他ノードと非同期でデータを連続して取得できます。
リクエスト (RTR) を受信すると、直ちに TPDO を送信します。

■非同期 (254、255)

非同期は、SYNC オブジェクトとは無関係であることを意味しています。

送信タイプ 254 と 255 は、ドライバの内部イベントによってトリガされ、PDO を送信するイベントドリブン方式です。smartris ドライバでの内部イベントは、以下の 2 つで定義されています。

- ・ ステータスワードが変化するとき
- ・ 任意に設定したイベントタイマが経過 (初期値は、イベントタイマ 0：無効)

■禁止タイマ (Inhibit Timer)

送信タイプが 254 または 255 に設定されている場合、PDO を送信する最小インターバル時間を禁止タイマ (Inhibit Timer) として設定することができます。

禁止タイマで設定された時間が経過するまで、同一の PDO を送信することはありません。

禁止タイマは、通信パラメータの Sub-Index 3 で設定可能です (単位: 100 μ s)。

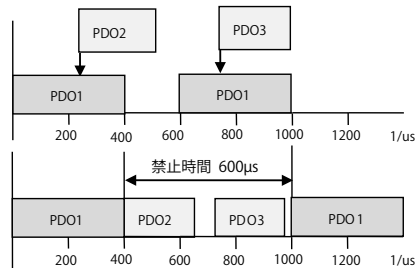


図 2-11 非同期タイプ (254 または 255) 設定時の禁止タイマ

■イベントタイマ (Event Timer)

イベントタイマは、非同期 (送信タイプ 254 または 255) に対応した機能です。

PDO が送信されるタイミングは、以下の通りです。

- ・ イベントタイマで設定した時間が経過
- ・ イベントタイマが動作中に内部イベントが発生した場合 (タイマはリセットされます)。

【2】 CANopen 運転

イベントタイマは、通信パラメータの Sub-Index 5 で設定可能です (単位：1ms)。
0 を設定した場合は、イベントタイマ機能は無効になります。

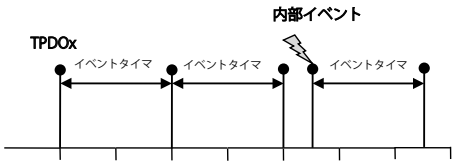


図 2-12 非同期 PDO のイベントタイマ

注) イベントタイマで周期的な TPDO が送信されている場合、内部イベントが無効です。

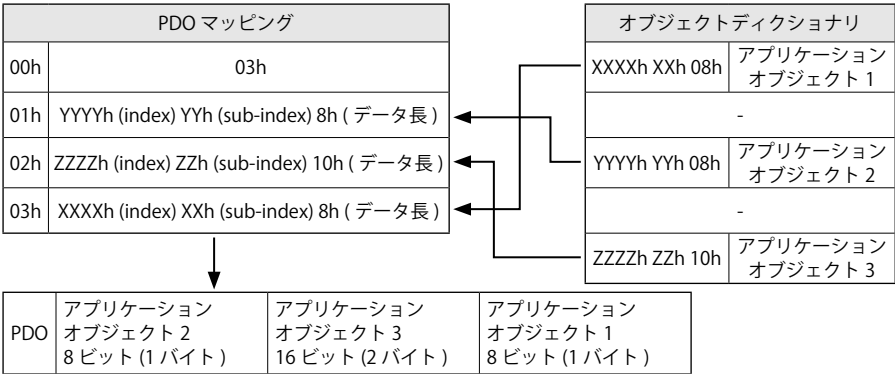
⚠ 注意

PDO は最小で 2.5ms 間隔で送信してください。
PDO にコントロールワードがマッピングされている場合は、40ms 待機してから次の PDO を送信してください。(ドライバが、コントロールワードの指令を受け取って処理するための時間を確保する必要があるためです。)
また、ドライバが “Pre-operational” ステートの場合、初期マッピングを変更することができます。

■ PDO マッピング

PDO マッピングは、ドライバが使用するオブジェクト(リアルタイムデータ)のマッピングを参照します。
出荷時の状態で既にマッピングされていますが、任意のオブジェクトへ変更することができます。
マッピングされたオブジェクトを変更する場合は、PDO が無効となる NMT 状態機械の “Pre-operational” ステートへ移行する必要があります。

例：PDO マッピング



【2】CANopen 運転

■ PDO 受信 (RPDO)

• 1400h ~ 1403h : RPDO 通信オブジェクト

smartris ドライバが PDO を受信する際の通信パラメータに関するオブジェクトです。

- Sub-index 00h は、最大のサブインデックス数を表します。
- Sub-index 01h は、PDO の COB-ID です。

31 (MSB)	30	29	28	11	10	0 (LSB)
有効	予約領域	フレーム	0000h	11 ビット CAN-ID		

名 称	ビット	数 値	内 容
有効	31	0b 1b	PDO 有効 PDO 無効
予約領域	11 ~ 30	-	-
11 ビット CAN-ID	0 ~ 10	x	COB-ID を表す 11 ビット

注) PDO 有効または NMT 状態機械が “Operational” ステートの場合、これらの設定変更ができません。

- Sub-index 02h は、送信タイプです。PDO の受信特性を定義します。
次の表は、使用可能な送信タイプを示しています。

表 2-11 送信タイプ

送信タイプ		内 容
0 ~ 240	00h ~ F0h	同期
241 ~ 253	F1h ~ FDh	予約領域
254, 255	FEh, FFh	イベントドリブン

0 ~ 240 「同期」

- 受信した PDO データは、次の SYNC メッセージの受信後に反映されます。
0 ~ 240 のいずれの設定も全て同じ挙動を示します。

254, 255 「イベントドリブン」

- PDO が受信できることを意味します。受信した PDO データをすぐに反映させます。

- Sub-index 03h は、禁止時間 (Inhibit time) です。(単位: 100μs)
0 が設定された場合は、本機能が無効となります。
PDO 有効 (Sub-index 01h のビット 31 が 0b) に設定されている場合は、設定の変更ができません。
設定変更する場合は、PDO 無効 (Sub-index 01h のビット 31 が 1b) にしてください。
- Sub-index 05h は、イベントタイマです。(単位: 1 ms)
0 が設定された場合は、本機能が無効となります。
また、送信タイプ (Sub-index 02h) が 254 または 255 の場合でのみ設定可能です。

これらのパラメータは、ドライバが NMT 状態機械の “Pre-operational” ステートのとき変更できません。

パラメータ変更の手順

- PDO 無効: COB-ID (Sub-index 01h) のビット 31 を “1” に設定
- 各 Sub-index へ任意の設定値を書込み
- PDO 有効: COB-ID (Sub-index 01h) のビット 31 を “0” に設定

例: RPDO1 (ノード ID1) の送信タイプを SYNC メッセージと同期へ変更

(1) RPDO1 無効

Rx: COB-ID 0x601 - 23 00 14 01 00 00 00 80

[cmd: 0x23] [index: 0x1400] [sub-index: 0x01] [データ: 0x80000000] - PDO 無効 (ビット 31= 1)

Tx: COB-ID 0x581 - 60 00 14 01 00 00 00 00

[resp: 0x60] [index: 0x1400] [sub-index: 0x01] [データ: 0x00000000] - 書き込み成功

【2】 CANopen 運転

(2) 送信タイプの変更

Rx: COB-ID 0x601 - 2F 00 14 02 01 00 00 00
[cmd: 0x2F] [index: 1400] [sub-index: 0x02] [データ : 0x00000001] - Tx タイプ = 1
Tx: COB-ID 0x581 - 60 00 14 02 00 00 00 00
[resp: 0x60] [index: 1400] [sub-index: 0x02] [データ : 0x00000000] - 書き込み成功

(3) PDO 有効と COB-ID (0x201) の設定

Rx: COB-ID 0x601 - 23 00 14 01 01 02 00 00
[cmd: 0x23] [index: 0x1400] [sub-index: 0x01] [データ : 0x00000201]
- PDO 有効 (ビット 31=0)、COB-ID=0x201
Tx: COB-ID 0x581 - 60 00 14 01 00 00 00 00
[resp: 0x60] [index: 0x1400] [sub-index: 0x01] [データ : 0x00000000] - 書き込み成功

メッセージ内容

バイト 0	バイト 1	バイト 2	バイト 3	バイト 4	バイト 5	バイト 6	バイト 7
Cmd または Resp	Index		Sub-Index	データ			

・ 1600h ~ 1603h : RPDO マッピングパラメータ

smartris ドライバが PDO を受信する際のマッピングに関するオブジェクトです。
・ Sub-index 00h は、受信する PDO に含まれたオブジェクト数(有効なオブジェクトエントリ数)です。
設定値は、以下のように定義されています。

表 2-12 設定値の定義

設定値	内 容
00h	マッピング無効
01h	Sub-index 01h 有効
02h	Sub-index 01h、02h 有効
...	...
08h	Sub-index 01h ~ 08h 有効

・ Sub-index の 01h ~ 08h は、マッピングされたオブジェクトの情報を設定します。
オブジェクトの情報は、Index および Sub-Index、オブジェクトのデータ長です。

バイト : MSB	LSB		
Index (16 ビット)		Sub-Index (8 ビット)	データ長 (8 ビット)

SDO で PDO のマッピング変更が正常に設定できない場合、ドライバはアボートコードを送信します。

smartris ドライバは、 1 つの PDO につき 8 バイト分を使用できます。
従って、PDO でマッピング可能な範囲は、最大 8 つのオブジェクト (全てのオブジェクトが 1 バイト長)を使用できます。

【2】 CANopen 運転

■ RPDO 初期マッピング

RPDO の CAN フレームは、11 ビットのヘッダで識別されます。

- RPDO1 : 0x200 + ノード ID
- RPDO2 : 0x300 + ノード ID
- RPDO3 : 0x400 + ノード ID
- RPDO4 : 0x500 + ノード ID

RPDO の初期マッピングを以下の表に示します。

表 2-13 RPDO 定義

Index	Sub-Index	内 容	タイプ	属性	初期値	備 考
RPDO1						
1400h	0	受信 PDO1 (RPDO1)	UNSIGNED8	ro	5	エントリ数
	1	COB ID (PDO1)	UNSIGNED32	rw	0x201	PDO 有効 (ビット 31=0)
	2	送信タイプ	UNSIGNED8	rw	0xFE	イベントドリブン
	3	禁止 (Inhibit) 時間	UNSIGNED16	rw	0x00	禁止時間無効
	4	エントリ互換	UNSIGNED8	rw	0	使用不可
	5	イベントタイマ	UNSIGNED16	rw	0	0 = 無効 [ms]
1600h	0	RPDO マッピング N	UNSIGNED8	rw	3	エントリ数
	1	アプリケーションオブジェクト 1	UNSIGNED32	rw	0x6040 0010	コントロールワード (2 バイト)
	2	アプリケーションオブジェクト 2	UNSIGNED32	rw	0x6060 0008	運転モード (1 バイト)
	3	アプリケーションオブジェクト 3	UNSIGNED32	rw	0x60FE 0120	デジタル出力 (4 バイト)
	4	アプリケーションオブジェクト 4	UNSIGNED32	rw	0	-
	5	アプリケーションオブジェクト 5	UNSIGNED32	rw	0	
	6	アプリケーションオブジェクト 6	UNSIGNED32	rw	0	
	7	アプリケーションオブジェクト 7	UNSIGNED32	rw	0	
	8	アプリケーションオブジェクト 8	UNSIGNED32	rw	0	
RPDO2						
1401h	0	受信 PDO2	UNSIGNED8	ro	5	エントリ数
	1	COB ID (PDO2 を使用)	UNSIGNED32	rw	0x301	PDO 有効 (ビット 31=0)
	2	送信タイプ	UNSIGNED8	rw	0xFE	イベントドリブン
	3	禁止 (Inhibit) 時間	UNSIGNED16	rw	0x00	禁止時間無効
	4	エントリ互換	UNSIGNED8	rw	0	使用不可
	5	イベントタイマ	UNSIGNED16	rw	0	0 = 無効 [ms]
1601h	0	RPDO マッピング N	UNSIGNED8	rw	2	エントリ数
	1	アプリケーションオブジェクト 1	UNSIGNED32	rw	0x6040 0010	コントロールワード (2 バイト)
	2	アプリケーションオブジェクト 2	UNSIGNED32	rw	0x607A 0020	目標位置 (4 バイト)
	3	アプリケーションオブジェクト 3	UNSIGNED32	rw	0	-
	4	アプリケーションオブジェクト 4	UNSIGNED32	rw	0	
	5	アプリケーションオブジェクト 5	UNSIGNED32	rw	0	
	6	アプリケーションオブジェクト 6	UNSIGNED32	rw	0	
	7	アプリケーションオブジェクト 7	UNSIGNED32	rw	0	
	8	アプリケーションオブジェクト 8	UNSIGNED32	rw	0	
RPDO3						
1402h	0	受信 PDO3	UNSIGNED8	ro	5	エントリ数
	1	COB ID (PDO3)	UNSIGNED32	rw	0x401	PDO 有効 (ビット 31=0)
	2	送信タイプ	UNSIGNED8	rw	0xFE	イベントドリブン
	3	禁止 (Inhibit) 時間	UNSIGNED16	rw	0x00	禁止時間無効
	4	エントリ互換	UNSIGNED8	rw	0	使用不可
	5	イベントタイマ	UNSIGNED16	rw	0	0 = 無効 [ms]
1602h	0	RPDO マッピング N	UNSIGNED8	rw	2	エントリ数
	1	アプリケーションオブジェクト 1	UNSIGNED32	rw	0x6040 0010	コントロールワード (2 バイト)
	2	アプリケーションオブジェクト 2	UNSIGNED32	rw	0x60FF 0020	目標速度 (4 バイト)
	3	アプリケーションオブジェクト 3	UNSIGNED32	rw	0	-
	4	アプリケーションオブジェクト 4	UNSIGNED32	rw	0	
	5	アプリケーションオブジェクト 5	UNSIGNED32	rw	0	
	6	アプリケーションオブジェクト 6	UNSIGNED32	rw	0	
	7	アプリケーションオブジェクト 7	UNSIGNED32	rw	0	
	8	アプリケーションオブジェクト 8	UNSIGNED32	rw	0	

【2】 CANopen 運転

表 2-14 RPDO 定義 (続き)

Index	Sub-Index	内 容	タイプ	属性	初期値	備 考
受信 PDO (RPDO4)						
1403h	0	受信 PDO4	UNSIGNED8	ro	5	エントリ数
	1	COB ID (PDO4)	UNSIGNED32	rw	0x501	PDO 有効 (ビット 31=0)
	2	送信タイプ	UNSIGNED8	rw	0xFE	イベントドリブン
	3	禁止 (Inhibit) 時間	UNSIGNED16	rw	0x00	禁止時間無効
	4	エントリ互換	UNSIGNED8	rw	0	使用不可
1603h	5	イベントタイム	UNSIGNED16	rw	0	0 = 無効 [ms]
	0	RPDO マッピング N	UNSIGNED8	rw	2	エントリ数
	1	アプリケーションオブジェクト 1	UNSIGNED32	rw	0x6040 0010	コントロールワード (2 バイト)
	2	アプリケーションオブジェクト 2	UNSIGNED32	rw	0x6071 0010	目標トルク (2 バイト)
	3	アプリケーションオブジェクト 3	UNSIGNED32	rw	0	
	4	アプリケーションオブジェクト 4	UNSIGNED32	rw	0	
	5	アプリケーションオブジェクト 5	UNSIGNED32	rw	0	
	6	アプリケーションオブジェクト 6	UNSIGNED32	rw	0	
	7	アプリケーションオブジェクト 7	UNSIGNED32	rw	0	
	8	アプリケーションオブジェクト 8	UNSIGNED32	rw	0	

RPDO1 の初期マッピング：状態機械の制御

表 2-15 RPDO1 マッピング

Index+Sub-index	名 称	初期値	バイト数
1600h: 0	マップされたオブジェクト数	3	-
1600h: 1	コントロールワード	6040 0010h	2
1600h: 2	運転モード	6060 0008h	1
1600h: 3	デジタル出力	60FE 0120h	4

RPDO2 の初期マッピング：状態機械と目標位置の制御

表 2-16 RPDO2 マッピング

Index+Sub-index	名 称	初期値	バイト数
1601h: 0	マップされたオブジェクト数	2	-
1601h: 1	コントロールワード	6040 0010h	2
1601h: 2	目標位置	607A 0020h	4

RPDO3 の初期マッピング：状態機械と目標速度の制御

表 2-17 RPDO3 マッピング

Index+Sub-index	名 称	初期値	バイト数
1602h: 0	マップされたオブジェクト数	2	-
1602h: 1	コントロールワード	6040 0010h	2
1602h: 2	目標速度	60FF 0020h	4

RPDO4 の初期マッピング：状態機械と目標トルクの制御

表 2-18 RPDO4 マッピング

Index+Sub-index	名 称	初期値	バイト数
1603h: 0	マップされたオブジェクト数	2	-
1603h: 1	コントロールワード	6040 0010h	2
1603h: 2	目標トルク	6071 0010h	2

マスタが RPDO を送信する際、メッセージ長の選択に注意してください。

例：RPDO1 の初期マッピング

以下の 3 つのオブジェクトで構成されています。

- ・コントロールワード：0x60400010 = 6040h (index)、00h (sub-index)、10h (16 ビット = 2 バイト長)
- ・運転モード：0x60600008 = 6040h (index)、00h (sub-index)、08h (8 ビット = 1 バイト長)
- ・デジタル出力：0x60FE0120 = 60FEh (index)、01h (sub-index)、20h (32 ビット = 4 バイト長)

3 つのオブジェクトのデータ長は合計 7 バイトなので、RPDO1 のメッセージ長(DLC)は 7 バイトです。

RPDO1 のメッセージ長を 7 バイト以上で送信すると、ドライバは、エラーコード 0x8220 のエマージェンシメッセージを送信します。

注) 詳細は、10 章 サンプルプログラムの「10-3 PDO を介して制御」を参照してください。

■ RPDO リマッピングの手順

マッピングを変更(リマッピング)する手順を以下に示します。

このとき NMT 状態機械は、“Pre-operational” ステートで実施してください。

(1) PDO のマッピングを変更するために、まずは PDO を無効にしてください。

RPDO 通信設定(1400h ~ 1403h)の Sub-index 01h の 31 ビットを 1 (無効)に設定します。

(2) マッピングパラメータ(1600h ~ 1603h)の Sub-index 00h を 0 に設定して、PDO のマッピングを一時的に無効にします。

(3) 任意の Sub-index の設定値を変更して、リマッピングしてください。

オブジェクト(Index、sub-Index、データ長)の内容を任意の Sub-index へ書込みます。注)1

(4) マッピングパラメータ(1600h ~ 1603h)の Sub-index 00h を 1 に設定して、PDO のマッピングを有効にします。注)2

(5) RPDO 通信設定(1400h ~ 1403h)の Sub-index 01h の 31 ビットを 0 (有効)に設定します。

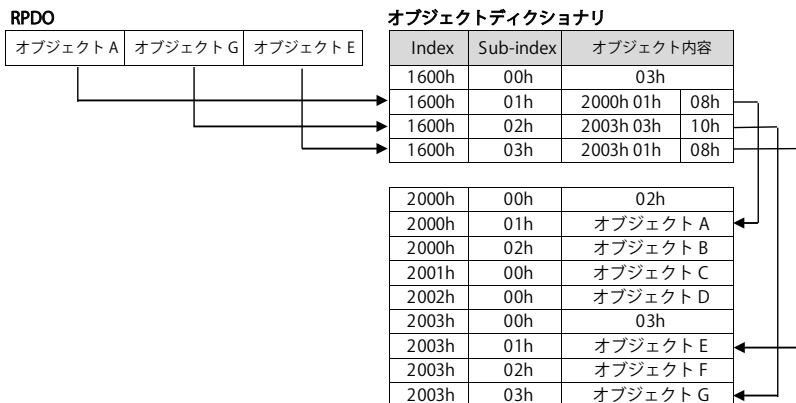
(6) RPDO のリマッピングが完了

注) 1. 存在しないオブジェクトをマッピングした場合、SDO アボートコード(0602 0000h)が送信されます。マッピング不可のオブジェクトをマッピングした場合、SDO アボートコード(0604 0041h)が送信されます。

2. マッピングするオブジェクトの数またはデータ長が PDO のデータ長を超えている場合、SDO アボートコード(0604 0042h)が送信されます。

【2】 CANopen 運転

例：RPDO リマッピング



注) 詳細は、10 章 サンプルプログラムの「10-4 RPDO(TPDO) リマッピング手順」を参照してください。

デバイスは、マッピングされたデータ長より長い PDO メッセージを受信した場合、受信したメッセージの先頭から設定されたデータ長分だけ処理します。

例：コントロールワードと目標速度(ノード ID 1)の RPDO1 のリマッピング

(1) RPDO1 無効

Rx: COB-ID 0x601 - 23 00 14 01 00 00 00 80

[cmd: 0x23] [index: 0x1400] [sub-index: 0x01] [データ : 0x80000000] - PDO 無効 (ビット 31= 1)

Tx: COB-ID 0x581 - 60 00 14 01 00 00 00 00

[resp: 0x60] [index: 0x1400] [sub-index: 0x01] [データ : 0x00000000] - 書き込み成功

(2) RPDO1 のマッピング無効

Rx: COB-ID 0x601 - 2F 00 16 00 00 00 00 00

[cmd: 0x2F] [index: 0x1600] [sub-index: 0x00] [データ : 0x00000000] - マッピング無効

Tx: COB-ID 0x581 - 60 00 16 00 00 00 00 00

[resp: 0x60] [index: 0x1600] [sub-index: 0x00] [データ : 0x00000000] - 書き込み成功

(3) コントロールワード(6040h)のマッピング

Rx: COB-ID 0x601 - 23 00 16 01 10 00 40 60

[cmd: 0x23] [index: 1600] [sub-index: 0x01] [data: 0x60400010] - 6040h: コントロールワードのマッピング

Tx: COB-ID 0x581 - 60 00 16 01 00 00 00 00

[resp: 0x60] [index: 1600] [sub-index: 0x01] [data: 0x00000000] - 書き込み成功

(4) 目標速度(60FFh)のマッピング

Rx: COB-ID 0x601 - 23 00 16 02 20 00 FF 60

[cmd: 0x23] [index: 1600] [sub-index: 0x02] [data: 0x60FF0020] - 60FFh: 目標速度のマッピング

Tx: COB-ID 0x581 - 60 00 16 02 00 00 00 00

[resp: 0x60] [index: 1600] [sub-index: 0x02] [data: 0x00000000] - 書き込み成功

【2】CANopen 運転

(5) RPDO1 のマッピング有効

Rx: COB-ID 0x601 - 2F 00 16 00 02 00 00 00

[cmd: 0x2F] [index: 1600] [sub-index: 0x00] [データ: 0x00000002] - マッピング有効

Tx: COB-ID 0x581 - 60 00 16 00 00 00 00 00

[resp: 0x60] [index: 1600] [sub-index: 0x00] [データ: 0x00000000] - 書き込み成功

(6) PDO 有効と COB-ID (0x201) の設定

Rx: COB-ID 0x601 - 23 00 14 01 01 02 00 00

[cmd: 0x23] [index: 0x1400] [sub-index: 0x01] [データ: 0x00000201]

— PDO 有効 (ビット 31=0)、COBID=0x201

Tx: COB-ID 0x581 - 60 00 14 01 00 00 00 00

[resp: 0x60] [index: 0x1400] [sub-index: 0x01] [データ: 0x00000000] - 書き込み成功

メッセージ内容

バイト 0	バイト 1	バイト 2	バイト 3	バイト 4	バイト 5	バイト 6	バイト 7
Cmd または Resp	Index		Sub-Index	データ			

■ PDO 送信 (TPDO)

• 1800h ~ 1803h : TPDO 通信オブジェクト

smartris ドライバが PDO を送信する際の通信パラメータに関するオブジェクトです。

- Sub-index 00h は、最大のサブインデックス数を表します。
- Sub-index 01h は、PDO の COB-ID です。

31 (MSB)	30	29	28	11	10	0 (LSB)
有効	RTR	フレーム	0000h	11 ビット CAN-ID		

名 称	ビット	数 値	内 容
有効	31	0b 1b	PDO 有効 PDO 無効
RTR	30	0b 1b	RTR の受信で送信 RTR には応答しない
予約領域	11 ~ 29	-	-
11 ビット CAN-ID	0 ~ 10	x	COB-ID を表す 11 ビット

- Sub-index 02h は、送信タイプです。PDO の送信特性を定義します。

次の表は、使用可能な送信タイプを示しています。

表 2-19 送信タイプ

送信タイプ		内 容
0	00h	同期 (非周期)
1 ~ 240	01h ~ F0h	同期 (N 回の SYNC メッセージ毎)
241 ~ 251	F0h ~ FBh	予約領域
252	FCh	RTR-only (同期)
253	FDh	RTR-only (イベントドリブン)
254, 255	FEh, FFh	内部イベント (ステータスワードの値が変化、またはイベントタイマが経過、または RTR 受信) によって送信されます。

[2] CANopen 運転

0 ～ 240 「同期」

- SYNC メッセージ受信後に smartris ドライバは、PDO にマッピングされたデータを取得し、PDO として送信します。

SYNC メッセージが周期的でない場合は、内部イベントの発生(ステータスワードの値が変化)後に、次の SYNC メッセージでデータを取得して PDO を送信します。

周期的な場合は、設定した回数(N 回) SYNC メッセージを受信した後に、PDO が送信されます。

252,253 「RTR-only」

- SYNC メッセージと同期している場合、smartris ドライバは、RTR を受信するたびにデータを取得して PDO に保持します。PDO は、SYNC メッセージを受け取った後に送信されます。イベントドリブンの場合、RTR の受信でデータを取得し、直ぐに PDO を送信します。

254,255 「イベントドリブン」

- 内部イベント(ステータスワードの値が変化またはイベントタイマが経過)によって送信されます。

- Sub-index 03h は、禁止時間(Inhibit time)です。(単位：100μs)

0 が設定された場合は、本機能が無効となります。送信タイプに 254 (FEh)または 255 (FFh) が設定されているときの PDO を送信する最小インターバルです。

PDO 有効(Sub-index 01h のビット 31 が 0b)に設定されている場合は、設定の変更ができません。設定変更する場合は、PDO 無効(Sub-index 01h のビット 31 が 1b)にしてください。

- Sub-index 05h は、イベントタイマです。(単位：1 ms)

0 が設定された場合は、本機能が無効となります。送信タイプに 254 (FEh)または 255 (FFh) が設定されているときの PDO を送信する最大インターバルです。

これらのパラメータは、ドライバが NMT 状態機械の "Pre-operational" ステートのとき変更できます。

パラメータ変更の手順

- PDO 無効：COB-ID (Sub-index 01h)のビット 31 を "1" に設定
- 各 Sub-index へ任意の設定値を書込み
- PDO 有効：COB-ID (Sub-index 01h)のビット 31 を "0" に設定

例：TPDO1 (ノード ID 1) のイベントタイマを 100ms に修正

(1) TPDO1 無効

Rx: COB-ID 0x601 - 23 00 18 01 00 00 00 80

[cmd: 0x23] [index: 0x1800] [sub-index: 0x01] [データ: 0x80000000] - PDO 無効 (ビット 31= 1)

Tx: COB-ID 0x581 - 60 00 18 01 00 00 00 00

[resp: 0x60] [index: 0x1800] [sub-index: 0x01] [データ: 0x00000000] - 書込み成功

(2) イベントタイマ変更

Rx: COB-ID 0x601 - 2B 00 18 05 64 00 00 00

[cmd: 0x2B] [index: 0x1800] [sub-index: 0x05] [データ: 0x00000064] - イベントタイマ :100ms =0x64

Tx: COB-ID 0x581 - 60 00 18 05 00 00 00 00

[resp: 0x60] [index: 0x1800] [sub-index: 0x05] [データ: 0x00000000] - 書込み成功

(3) PDO 有効と COB-ID(0x181) の設定

Rx: COB-ID 0x601 - 23 00 18 01 81 01 00 00

[cmd: 0x23] [index: 0x1800] [sub-index: 0x01] [データ : 0x000000181]

– PDO 有効 (ビット 31=0)、COB-ID=0x181

Tx: COB-ID 0x581 - 60 00 18 01 00 00 00 00

[resp: 0x60] [index: 0x1800] [sub-index: 0x01] [データ : 0x000000000] – 書き込み成功

メッセージ内容

バイト 0	バイト 1	バイト 2	バイト 3	バイト 4	バイト 5	バイト 6	バイト 7
Cmd または Resp	Index		Sub-Index	データ			

・ 1A00h ~ 1A03h : TPDO マッピングパラメータ

smartris ドライバが PDO を送信する際のマッピングに関するオブジェクトです。

- Sub-index 00h は、受信する PDO に含まれたオブジェクト数 (有効なオブジェクトエントリ数) です。

設定値は、次の表のように定義されています。

表 2-20 設定値の定義

設定値	内 容
00h	マッピング無効
01h	Sub-index 01h 有効
02h	Sub-index 01h と 02h 有効
...	...
08h	Sub-index 01h ~ 08h 有効

- Sub-index 01h ~ 08h は、マッピングされたオブジェクトの情報を設定します。
オブジェクトの情報は、Index および Sub-Index、オブジェクトのデータ長です。

バイト : MSB

LSB

Index (16 ビット)	Sub-Index (8 ビット)	データ長 (8 ビット)
----------------	-------------------	--------------

smartris ドライバは 1 つの PDO につき 8 バイト分を使用できます。

従って、PDO でマッピング可能な範囲は、最大 8 つのオブジェクト (全てのオブジェクトが 1 バイト長) を使用できます。

■ TPDO 初期マッピング

TPDO の CAN フレームは、11 ビットのヘッダで識別されます。

- TPDO1: 0x180 + ノード ID
- TPDO2: 0x280 + ノード ID
- TPDO3: 0x380 + ノード ID
- TPDO4: 0x480 + ノード ID

TPDO の初期マッピングを次表に示します。

【2】 CANopen 運転

表 2-21 TPDO 定義

Index	Sub-Index	内 容	タイプ	属 性	初期値	備 考
送信プロセスデータオブジェクト (TPDO1)						
1800h	0	送信 PDO1 (TPDO1)	UNSIGNED8	ro	5	エントリ数
	1	COB ID (TPDO1)	UNSIGNED32	rw	0x181	PDO 有効 (ビット 31=0)
	2	送信タイプ	UNSIGNED8	rw	0xFF	イベントドリブン
	3	禁止時間 (Inhibit Time)	UNSIGNED16	rw	0x5	500 us [n × 100us]
	4	エントリ互換	UNSIGNED8	rw	0	使用不可
	5	イベントタイマ	UNSIGNED16	rw	0	0 = 無効 [ms]
1A00h	0	TPDO1 マッピング	UNSIGNED8	rw	3	エントリ数
	1	アプリケーションオブジェクト 1	UNSIGNED32	rw	0x60410010	ステータスワード (2 バイト)
	2	アプリケーションオブジェクト 2	UNSIGNED32	rw	0x60610008	運転モード表示 (1 バイト)
	3	アプリケーションオブジェクト 3	UNSIGNED32	rw	0x60FD0020	デジタル入力 (4 バイト)
	4	アプリケーションオブジェクト 4	UNSIGNED32	rw	0	-
	5	アプリケーションオブジェクト 5	UNSIGNED32	rw	0	
	6	アプリケーションオブジェクト 6	UNSIGNED32	rw	0	
	7	アプリケーションオブジェクト 7	UNSIGNED32	rw	0	
	8	アプリケーションオブジェクト 8	UNSIGNED32	rw	0	
送信プロセスデータオブジェクト (TPDO2)						
1801h	0	送信 PDO2 (TPDO2)	UNSIGNED8	ro	5	エントリ数
	1	COB ID (TPDO2)	UNSIGNED32	rw	0x281	PDO 有効 (ビット 31=0)
	2	送信タイプ	UNSIGNED8	rw	0xFD	RTR-only
	3	禁止時間 (Inhibit Time)	UNSIGNED16	rw	0x5	500 us [n × 100us]
	4	エントリ互換	UNSIGNED8	rw	0	使用不可
	5	イベントタイマ	UNSIGNED16	rw	0	0 = 無効 [ms]
1A01h	0	TPDO2 マッピング	UNSIGNED8	rw	2	エントリ数
	1	アプリケーションオブジェクト 1	UNSIGNED32	rw	0x60410010	ステータスワード (2 バイト)
	2	アプリケーションオブジェクト 2	UNSIGNED32	rw	0x60640020	位置検出値 (4 バイト)
	3	アプリケーションオブジェクト 3	UNSIGNED32	rw	0	-
	4	アプリケーションオブジェクト 4	UNSIGNED32	rw	0	
	5	アプリケーションオブジェクト 5	UNSIGNED32	rw	0	
	6	アプリケーションオブジェクト 6	UNSIGNED32	rw	0	
	7	アプリケーションオブジェクト 7	UNSIGNED32	rw	0	
	8	アプリケーションオブジェクト 8	UNSIGNED32	rw	0	
送信プロセスデータオブジェクト (TPDO3)						
1802h	0	送信 PDO3 (TPDO3)	UNSIGNED8	ro	5	エントリ数
	1	COB ID (TPDO3)	UNSIGNED32	rw	0x381	PDO 有効 (ビット 31=0)
	2	送信タイプ	UNSIGNED8	rw	0xFD	RTR-only
	3	禁止時間 (Inhibit Time)	UNSIGNED16	rw	0x5	500 us [n × 100us]
	4	エントリ互換	UNSIGNED8	rw	0	使用不可
	5	イベントタイマ	UNSIGNED16	rw	0	0 = 無効 [ms]
1A02h	0	TPDO3 マッピング	UNSIGNED8	rw	2	エントリ数
	1	アプリケーションオブジェクト 1	UNSIGNED32	rw	0x60410010	ステータスワード (2 バイト)
	2	アプリケーションオブジェクト 2	UNSIGNED32	rw	0x606C0020	速度検出値 (4 バイト)
	3	アプリケーションオブジェクト 3	UNSIGNED32	rw	0	-
	4	アプリケーションオブジェクト 4	UNSIGNED32	rw	0	
	5	アプリケーションオブジェクト 5	UNSIGNED32	rw	0	
	6	アプリケーションオブジェクト 6	UNSIGNED32	rw	0	
	7	アプリケーションオブジェクト 7	UNSIGNED32	rw	0	
	8	アプリケーションオブジェクト 8	UNSIGNED32	rw	0	

【2】CANopen 運転

表 2-22 TPDO 定義(続き)

Index	Sub-Index	内 容	タイプ	属 性	初期値	備 考
送信プロセスデータオブジェクト (TPDO4)						
1803h	0	送信 PDO4 (TPDO4)	UNSIGNED8	ro	3	エントリ数
	1	COB ID (TPDO4)	UNSIGNED32	rw	0x481	PDO 有効 (ビット 31=0)
	2	送信タイプ	UNSIGNED8	rw	0xFD	RTR-only
	3	禁止時間 (Inhibit Time)	UNSIGNED16	rw	0x5	500 us [n × 100us]
	4	エントリ互換	UNSIGNED8	rw	0	使用不可
	5	イベントタイマ	UNSIGNED16	rw	0	0 = 無効 [ms]
1A03h	0	TPDO4 マッピング	UNSIGNED8	rw	2	エントリ数
	1	アプリケーションオブジェクト 1	UNSIGNED32	rw	0x60410010	ステータスワード (2 バイト)
	2	アプリケーションオブジェクト 2	UNSIGNED32	rw	0x60770010	トルク検出値 (2 バイト)
	3	アプリケーションオブジェクト 3	UNSIGNED32	rw	0	
	4	アプリケーションオブジェクト 4	UNSIGNED32	rw	0	
	5	アプリケーションオブジェクト 5	UNSIGNED32	rw	0	
	6	アプリケーションオブジェクト 6	UNSIGNED32	rw	0	
	7	アプリケーションオブジェクト 7	UNSIGNED32	rw	0	
	8	アプリケーションオブジェクト 8	UNSIGNED32	rw	0	

TPDO1 の初期マッピング：状態機械のステータス

表 2-23 TPDO1 マッピング

Index + Sub-index	名 称 (PDO1 送信)	初期値	バイトサイズ
1A00h: 0	マップされたオブジェクト数	3	-
1A00h: 1	ステータスワード	6041 0010h	2
1A00h: 2	運転モードディスプレイ	6061 0008h	1
1A00h: 3	デジタル入力	60FD 0020h	4

TPDO2 の初期マッピング：状態機械のステータスと位置

表 2-24 TPDO2 マッピング

Index + Sub-index	名 称 (PDO1 送信)	初期値	バイトサイズ
1A01h: 0	マップされたオブジェクト数	2	-
1A01h: 1	ステータスワード	6041 0010h	2
1A01h: 2	位置検出値	6064 0020h	4

TPDO3 の初期マッピング：状態機械のステータスと電流

表 2-25 TPDO3 マッピング

Index + Sub-index	名 称 (PDO1 送信)	初期値	バイトサイズ
1A02h: 0	マップされたオブジェクト数	2	-
1A02h: 1	ステータスワード	6041 0010h	2
1A02h: 2	速度検出値	606C 0020h	4

TPDO4 の初期マッピング：状態機械のステータスとトルク電流

表 2-26 TPDO4 マッピング

Index + Sub-index	名 称 (PDO1 送信)	初期値	バイトサイズ
1A03h: 0	マップされたオブジェクト数	2	-
1A03h: 1	ステータスワード	6041 0010h	2
1A03h: 2	トルク検出値	6077 0010h	2

注) 詳細は、10 章 サンプルプログラムの「10-3 PDO を介しての制御」を参照してください。

【2】 CANopen 運転

■ TPDO リマッピングの手順

マッピングを変更(リマッピング)する手順を以下に示します。

このとき NMT 状態機械は、“Pre-operational” ステートで実施してください。

(1) PDO のマッピングを変更するために、まず、PDO を無効にしてください。

TPDO 通信設定(1800h ~ 1803h)の Sub-index 01h の 31 ビットを 1 (無効)に設定します。

(2) マッピングパラメータ(1A00h ~ 1A03h)の Sub-index 00h を 0 に設定して、PDO のマッピングを一時的に無効にします。

(3) 任意の Sub-index の設定値を変更して、リマッピングしてください。

オブジェクト(Index、sub-Index、データ長)の内容を任意の Sub-index へ書込みます。注)1

(4) マッピングパラメータ(1A00h ~ 1A03h)の Sub-index 00h を 1 に設定して、PDO のマッピングを有効にします。注)2

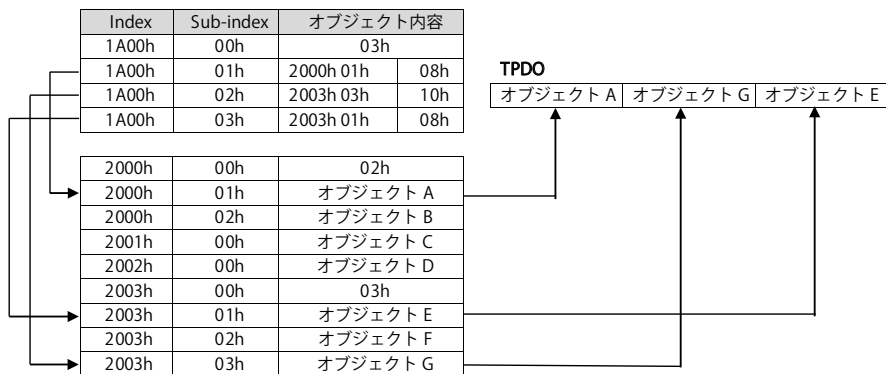
(5) TPDO 通信設定(1800h ~ 1803h)の Sub-index 01h の 31 ビットを 0 (有効)に設定します。

(6) TPDO のマッピングが完了

注) 1. 存在しないオブジェクトをマッピングした場合、SDO アボートコード(0602 0000h)が送信されます。マッピング不可のオブジェクトをマッピングした場合、SDO アボートコード(0604 0041h)が送信されます。

2. マッピングするオブジェクトの数またはデータ長が PDO のデータ長を超えている場合、SDO アボートコード(0604 0042h)が送信されます。

例：TPDO リマッピング



注) 詳細は、10 章 サンプルプログラムの「10-4 TPDO(RPDO) リマッピング手順」を参照してください。

例：ステータスワードと速度検出値(ノード ID：1)の TPDO1 のリマッピング

(1) TPDO1 無効

Rx: COB-ID 0x601 - 23 00 18 01 00 00 00 80

[cmd: 0x23] [index: 0x1800] [sub-index: 0x01] [データ: 0x80000000] - PDO 無効(ビット 31= 1)

Tx: COB-ID 0x581 - 60 00 18 01 00 00 00 00

[resp: 0x60] [index: 0x1800] [sub-index: 0x01] [データ: 0x00000000] - 書き込み成功

【2】 CANopen 運転

(2) TPDO1 のマッピング無効

Rx: COB-ID 0x601 - 2F 00 1A 00 00 00 00 00

[cmd: 0x2F] [index: 0x1A00] [sub-index: 0x00] [データ : 0x00000000] - マッピング無効

Tx: COB-ID 0x581 - 60 00 1A 00 00 00 00 00

[resp: 0x60] [index: 0x1A00] [sub-index: 0x00] [データ : 0x00000000] - 書き込み成功

(3) ステータスワード (6041h) のマッピング

Rx: COB-ID 0x601 - 23 00 1A 01 10 00 41 60

[cmd: 0x23] [index: 1A00] [sub-index: 0x01] [データ : 0x60410010] - 6041h: ステータスワードのマッピング

Tx: COB-ID 0x581 - 60 00 1A 01 00 00 00 00

[resp: 0x60] [index: 1A00] [sub-index: 0x01] [データ : 0x00000000] - 書き込み成功

(4) 速度検出値 (606Ch) のマッピング

Rx: COB-ID 0x601 - 23 00 1A 02 20 00 6C 60

[cmd: 0x23] [index: 1A00] [sub-index: 0x02] [データ : 0x606C0020] - 606Ch: 速度検出値のマッピング

Tx: COB-ID 0x581 - 60 00 1A 02 00 00 00 00

[resp: 0x60] [index: 1A00] [sub-index: 0x02] [データ : 0x00000000] - 書き込み成功

(5) TPDO1 のマッピング有効

Rx: COB-ID 0x601 - 2F 00 1A 00 02 00 00 00

[cmd: 0x2F] [index: 1A00] [sub-index: 0x00] [データ : 0x00000002] - マッピング有効

Tx: COB-ID 0x581 - 60 00 1A 00 00 00 00 00

[resp: 0x60] [index: 1A00] [sub-index: 0x00] [データ : 0x00000000] - 書き込み成功

(6) PDO 有効と COB-ID (0x181) 設定

Rx: COB-ID 0x601 - 23 00 18 01 81 01 00 00

[cmd: 0x23] [index: 0x1800] [sub-index: 0x01] [データ : 0x000000181]

— PDO 有効 (ビット 31=0)、COBID=0x181

Tx: COB-ID 0x581 - 60 00 18 01 00 00 00 00

[resp: 0x60] [index: 0x1800] [sub-index: 0x01] [データ : 0x00000000] - 書き込み成功

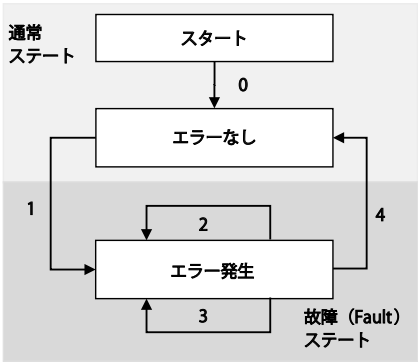
メッセージ内容

バイト 0	バイト 1	バイト 2	バイト 3	バイト 4	バイト 5	バイト 6	バイト 7
Cmd または Resp	Index		Sub-Index	データ			

【2】 CANopen 運転

2-9 エマージェンシメッセージ (EMCY)

ドライバでエラーが発生した場合、EMCY メッセージが送信されます。
直近で発生したエラーメッセージは、オブジェクト (603Fh: 0) で確認できます。
過去のエラー履歴は、(1003h: 1) に最大 15 個のエラーメッセージを記憶します。



次の表のステータス遷移が可能です。
表 2-27 ステータス遷移

遷移	要 因	内 容
0	初期化完了	エラーなし ドライバは、エラーコード 0000h (エラーリセット / エラーなし) を送信します。
1	エラー発生	エラーなしの状態からエラーが発生。ドライバは、故障ステートへ移行します。 ドライバの LED と EMCY メッセージを確認してください。
2	エラーのアクノリッジ失敗	全てのエラーが解消されず、エラーのアクノリッジが実行されます。
3	新たなエラーが発生	新たなエラーが発生 EMCY メッセージとして最新のエラーコードが (1003h: 1) に書込まれます。
4	エラーのアクノリッジ成功	全てのエラーが解消され、エラーのアクノリッジが実行されます。 EMCY メッセージでエラーコード 0000h (エラーリセット / エラーなし) が送信されます。

エマージェンシ (EMCY) オブジェクトは、CANopen デバイスでエラーが発生することによってトリガされ、ネットワークへ送信されます。
エマージェンシオブジェクトのメッセージ構造は、CANopen DSP402 で以下の通りに規定されています。

表 2-28 エマージェンシメッセージ構造

COB-ID	Rx/Tx	DLC	バイト							
			0	1	2	3	4	5	6	7
0x80+ ノード ID	Tx	8	エラーコード		レジスタ	メーカー固有エラーフィールド				
			E0	E1	R0	M0	M1	M2	M3	M4

【2】CANopen 運転

■標準エラーコードフィールド

表 2-29 エマージェンシエラーコード

エラーコード	名 称	内 容
0x0000	NO ERROR	エラーリセットまたはエラーなし
0x1000	GENERIC ERROR	一般のエラー
0x2000	CURRENT	電流
0x2100	CURRENT INPUT	デバイスの入力電流
0x2200	CURRENT INSIDE	デバイスの内部電流
0x2300	CURRENT OUTPUT	デバイスの出力電流
0x3000	VOLTAGE	電圧
0x3100	VOLTAGE MAINS	主電源の電圧
0x3200	VOLTAGE INSIDE	デバイスの内部電圧
0x3300	VOLTAGE OUTPUT	出力電圧
0x4000	TEMPERATURE	温度
0x4100	TEMP AMBIENT	周囲温度
0x4200	TEMP DEVICE	デバイス温度
0x5000	HARDWARE	デバイスハードウェア
0x6000	SOFTWARE DEVICE	デバイスソフトウェア
0x6100	SOFTWARE INTERNAL	内部ソフトウェア
0x6200	SOFTWARE USER	ユーザソフトウェア
0x6300	DATA SET	データ設定
0x7000	ADDITIONAL MODULE	追加モジュール
0x8000	MONITORING	モニタリング
0x8100	COMMUNICATION	通信
0x8200	PROTOCOL ERROR	プロトコルエラー
0x9000	EXTERNAL ERROR	外部エラー
0xF000	ADDITIONAL FUNC	追加機能
0xFF00	DEVICE SPECIFIC	デバイス仕様

■標準レジスタフィールド

CANopen デバイスは、内部エラーをエマージェンシメッセージのバイト 2 の領域に表示します。ビット 1 は、エラーの内容にかかわらずエラーが発生すると強制的に ON されます。

表 2-30 エマージェンシ・レジスタフィールド

レジスタ	ビット	名 称	内 容
0x00	-	NO ERROR	エラーなし
0x01	1	REGISTER GENERIC ERROR	一般のエラー
0x02	2	REGISTER CURRENT	電流
0x04	3	REGISTER VOLTAGE	電圧
0x08	4	REGISTER TEMPERATURE	温度
0x10	5	REGISTER COMMUNICATION ERROR	通信エラー
0x20	6	REGISTER DEVICE PROFILE	デバイスプロファイル
0x40	7	REGISTER RESERVED	予約領域 (常に 0)
0x80	8	REGISTER MANUFACTURER	メーカー固有

次の表は、smartris ドライバに実装されているアラーム (Fault (F) 故障 または Warning (W) 警告) を定義しています。

「LED コード」欄は、LED の点滅回数を示しています ([x] が緑色の LED、[y] が黄色の LED)。

例えば [x, y] = 6, 2 は、緑色の LED が 6 回点滅し、その後、黄色の LED が 2 回点滅します。

「エラーコード」欄は、アラームの番号を示しています。直近で発生したアラームは、オブジェクト (603Fh) 「エラーコード」で読出すことができます。

アラームが故障 (Fault) の場合、ドライバは、出力を遮断して停止します。

【2】 CANopen 運転

表 2-31 エマージェンシ内容

エラー	エラーコード	内 容	説 明	F/W	LEDコード
No Error	0x0000	エラーなし	リセット命令の実行または電源リセット	-	-
電流アラーム					
Short Circuit Motor	0x2340	短絡(モータ側)	過電流が発生	F	3,1
Load Level Fault	0x2350	過負荷異常(I2t サーマル)	I2t の積算による過電流アラーム(過負荷)	F	5,2
	0x2351	ワーニング(I2t サーマル)	I2t のリミットによる過電流アラーム(過負荷)	W	-
	0x2352	負荷レベル(I2t)解除待機中	過負荷連続発生時の過負荷タイマ解除待機中(1 分間)	F	5,2
	0x2353	負荷レベル(I2t)解除待機中警告	過負荷連続発生(最大回数到達)の警告	W	-
電圧アラーム					
Over Voltage	0x3210	DC リンク過電圧	過電圧アラームの発生	F	4,2
	0x3211	過電圧 1	過電圧アラームの発生(早期検出)	F	4,2
DC Link Under Voltage	0x3220	DC リンク不足電圧	不足電圧アラームの発生	F	4,1
温度アラーム					
Temperature Drive	0x4300	ドライバ温度	ヒートシンク過温度	F	1,1
	0x4301	ドライバ温度ワーニング	ドライバ温度のワーニング	W	-
	0x4310	ドライバ温度超過	ヒートシンク温度が最大範囲より高すぎる。	F	1,3
	0x4320	ドライバ温度低下	ヒートシンク温度が最低範囲より低すぎる。	F	1,3
Temperature Internal 1 Board	0x4500	プリント基板温度	プリント基板温度(> 68° C)	F	1,4
	0x4501	プリント基板温度ワーニング	プリント基板温度ワーニング(> 63° C)	W	-
	0x4510	プリント基板温度超過	プリント基板温度が最大範囲より高すぎる。	F	1,5
	0x4520	プリント基板温度低下	プリント基板温度が最低範囲より低すぎる。	F	1,5
Temperature Internal 1 Motor	0x4A00	モータ温度	モータ温度(> 140° C)	F	1,10
	0x4A01	モータ温度ワーニング	モータ温度ワーニング(> 130° C)	W	-
	0x4A10	モータ温度超過	モータ温度が最大範囲より高すぎる。	F	1,6
	0x4A20	モータ温度低下	モータ温度が最低範囲より低すぎる。	F	1,6
ハードウェアアラーム					
Input Stages	0x5430	入力ステージ	一般的な入力ステージ	-	-
	0x5431	センサオフセット	センサのオフセット	F	3,10
Hardware Memory	0x5500	ハードウェアメモリ	ハードウェアメモリの一般的なエラー	-	-
	0x5501	EEPROM 書込みのハードウェアエラー : バス(Vbus)電圧の低下	バス電圧があまりに低いため完全な書込みを保証することができません。	F	5,3
Hardware Memory EEPROM – User	0x5530	EEPROM	EEPROM 一般的なエラー	-	-
	0x5531	EEPROM ゼネラルエラー	EEPROM 書込みの一般的なエラー	F	6,1
	0x5532	EEPROM パラメータエラー 1	EEPROM パラメータ書込みエラー	F	6,1
	0x5533	EEPROM パラメータエラー 2			
	0x5534	EEPROM パラメータエラー 3			
	0x5535	EEPROM パラメータエラー 4			
	0x5536	EEPROM パラメータエラー 5			
	0x5537	EEPROM パラメータエラー 6			
	0x5538	EEPROM パラメータエラー 7			
	0x5539	EEPROM パラメータエラー 8			
	0x553A	EEPROM パラメータエラー 9			
	0x553B	EEPROM パラメータエラー 10			
	0x553C	EEPROM パラメータエラー 11			
	0x553D	EEPROM パラメータエラー 12			
	0x553E	EEPROM パラメータエラー 13			
	0x553F	EEPROM パラメータエラー 14			
	0x5540	EEPROM パラメータエラー 15			
	0x5541	EEPROM パラメータエラー 16			
	0x5542	EEPROM パラメータエラー 17			
	0x5543	EEPROM パラメータエラー 18			
	0x5544	EEPROM パラメータエラー 19			
	0x5545	EEPROM パラメータエラー 20			
	0x5546	EEPROM パラメータエラー 21			
	0x5547	EEPROM パラメータエラー 22			
	0x5548	EEPROM パラメータエラー 23			
	0x5549	EEPROM パラメータエラー 24			

【2】CANopen 運転

表 2-32 エマージェンシ内容(続き)

エラー	エラーコード	内 容	説 明	F/W	LEDコード
Hardware Memory EEPROM – User	0x554A	EEPROM パラメータエラー 25	EEPROM パラメータ書き込みエラー	F	6,1
	0x554B	EEPROM パラメータエラー 26			
	0x554C	EEPROM パラメータエラー 27			
	0x554D	EEPROM パラメータエラー 28			
	0x554E	EEPROM パラメータエラー 29			
	0x554F	EEPROM パラメータエラー 30			
	0x5550	EEPROM パラメータエラー 31			
	0x5551	EEPROM パラメータエラー 32			
	0x5552	EEPROM パラメータエラー 33			
	0x5553	EEPROM パラメータエラー 34			
	0x5554	EEPROM パラメータエラー 35			
	0x5555	EEPROM パラメータエラー 36			
	0x5556	EEPROM パラメータエラー 37			
	0x5557	EEPROM パラメータエラー 38			
	0x5558	EEPROM パラメータエラー 39			
	0x5559	EEPROM パラメータエラー 40			
	0x555A	EEPROM パラメータエラー 41			
	0x555B	EEPROM パラメータエラー 42			
	0x555C	EEPROM パラメータエラー 43			
	0x555D	EEPROM パラメータエラー 44			
	0x555E	EEPROM パラメータエラー 45			
	0x555F	EEPROM パラメータエラー 46			
	0x5560	EEPROM パラメータエラー 47			
	0x5561	EEPROM パラメータエラー 48			
	0x5562	EEPROM パラメータエラー 49			
	0x5563	EEPROM パラメータエラー 50			
	0x5564	EEPROM パラメータエラー 51			
	0x5565	EEPROM パラメータエラー 52			
	0x5566	EEPROM パラメータエラー 53			
	0x5567	EEPROM パラメータエラー 54			
	0x5568	EEPROM パラメータエラー 55			
	0x5569	EEPROM パラメータエラー 56			
	0x556A	EEPROM パラメータエラー 57			
	0x556B	EEPROM パラメータエラー 58			
	0x556C	EEPROM パラメータエラー 59			
	0x556D	EEPROM パラメータエラー 60			
	0x556E	EEPROM パラメータエラー 61			
	0x556F	EEPROM パラメータエラー 62			
Hardware Memory EEPROM – Factory Parameters	0x5A00	ゴールデンイメージ EEPROM データエリア	ゴールデンイメージの EEPROM データエリアの一般的なエラー	-	-
	0x5A01	ゴールデンイメージワーニングデータ	ゴールデンイメージのワーニングデータがフリー。	W	-
	0x5A02	ゴールデンイメージエラーデータ	ゴールデンイメージのデータの書き込みなし	F	8,1
ソフトウェアアラーム					
Software Device	0x6000	ソフトウェアデバイス	ソフトウェアデバイスの一般的なエラー	-	-
	0x6001	パラメータアップデート	CANopen によるアップデート要求が不可のワーニング (RS232 のみ許可)	W	-
パラメータアラーム					
Data Set	0x6300	データ設定パラメータ表	データ設定のプログラミングエラー	-	-
	0x6301	データレコード No. 1	データ設定のプログラミングエラー	F	7,1
	0x6302	データレコード No. 2			
	0x6303	データレコード No. 3			
	0x6304	データレコード No. 4			
	0x6305	データレコード No. 5			
	0x6306	データレコード No. 6			
	0x6307	データレコード No. 7			
	0x6308	データレコード No. 8			
	0x6309	データレコード No. 9			
	0x630A	データレコード No. 10			
	0x630B	データレコード No. 11			

【2】 CANopen 運転

表 2-33 エマージェンシ内容(続き)

エラー	エラーコード	内 容	説 明	F/W	LEDコード			
Data Set	0x630C	データレコード No. 12	データ設定プログラミングエラー	F	7,1			
	0x630D	データレコード No. 13						
	0x630E	データレコード No. 14						
	0x630F	データレコード No. 15						
	0x6401	データレコード No. 16 注)1						
	0x6402	データレコード No. 17						
	0x6403	データレコード No. 18						
	0x6404	データレコード No. 19 注)2						
	0x6405	データレコード No. 20						
	0x6406	データレコード No. 21						
	0x6407	データレコード No. 22						
	0x6408	データレコード No. 23						
	0x6409	データレコード No. 24						
	0x640A	データレコード No. 25						
	0x640B	データレコード No. 26						
	0x640C	データレコード No. 27						
	0x640D	データレコード No. 28						
	0x640E	データレコード No. 29						
	0x640F	データレコード No. 30						
Parameter Error	0x6410	データレコード No. 31	一般的なパラメータエラー	-	-			
	0x6411	データレコード No. 32						
	0x6320	パラメータエラー						
	0x6321	データコンフィグレーション 1 不一致				コンフィグレーションエラー	F	6,4
	0x6322	データコンフィグレーション 2 不一致						
アディショナルモジュールアラーム								
Encoder SinCos	0x7350	SinCos エンコーダ	SinCos エンコーダの一般的なエラー	F	2,6			
	0x7351	Rx エラー	受信メッセージエラー					
	0x7352	Tx エラー	送信メッセージエラー					
	0x7353	位置読み込み命令エラー	位置読み込みエラー					
	0x7354	ステータス命令エラー	SinCos エンコーダステータスエラー					
	0x7355	命令タイプエラー	SinCos エンコーダタイプエラー					
	0x7356	初期化タイムアウト命令	SinCos 初期化中のタイムアウトエラー					
Resolver	0x7370	レゾルバ	一般的なエラー	-	-			
	0x7373	レゾルバ位相の調整不可	レゾルバ読み込み中の初期調整エラー	F	2,4			
	0x7374	レゾルバ初期化	レゾルバ初期化アラームの発生	F	2,4			
	0x7375	レゾルバハードウェアエラー	レゾルバの故障・接続エラー	F	2,10			
	0x7376							
	0x7377							
	0x7378	レゾルバハードウェアエラー LOS, DOS, LOT : 位相初期化中						

注) 1. メーカーコード : No.1 ~ 19

2. メーカーコード : No.1 ~ 20

【2】CANopen 運転

表 2-34 エマージェンシ内容(続き)

エラー	エラーコード	内 容	説 明	F/W	LEDコード
Communication	0x7500	通信	-	-	-
	0x7530	フィールドバス No.1 - CANopen	CANopen 一般的なエラー	-	-
	0x7531	CANopen プロトコル初期化エラー	初期化エラー	W	-
	0x7532	CANopen プロトコルハードウェアエラー	ハードウェアエラー	F	5,4
アラームモニタリング					
Communication CANopen	0x8100	CANopen 通信	通信エラー	F	6,2
	0x8110	CAN オーバーラン	CAN コントローラ RX バッファハードウェアオーバーラン(オーバーフロー)		
	0x8111	Tx バッファオーバーフロー	TX ソフトウェアバッファオーバーフロー		
	0x8112	Rx バッファオーバーフロー	RX ソフトウェアバッファオーバーフロー		
	0x8120	CAN バッシブ	CAN バッシブエラー		
	0x8121	CAN Bus Off	CAN バスオフエラー		
	0x8130	ハートビート/ノードガーディング	ハートビートかライフノードガーディング	W	-
	0x8131	ノードガーディングエラー(スレーブがメッセージ未受信)	スレーブがガーディングメッセージを未受信		
	0x8132	ノードガーディングエラー(未接続)	ノードのライフタイムが経過し未接続		
	0x8133	ノードガーディングエラー(最低 1 つのメッセージ未受信)	スレーブが最低 1 つのメッセージを未受信		
	0x8140	Bus Off 復帰	CAN Bus Off から復帰		
	0x8150	CAN ID 衝突	CAN-ID 衝突		
	0x8160	CAN 初期ステート	通信ステートメッセージ：CANopen INIT ステート		
	0x8170	CAN 動作ステート	通信ステートメッセージ：CANopen ACTIVE ステート		
0x8180	CAN Bus Off ステート	通信ステートメッセージ：CANopen Bus Off ステート			
0x8190	CAN バッシブステートエラー	通信ステートメッセージ：CANopen バッシブステート			
プロトコルアラーム					
Protocol Error	0x8200	プロトコルエラー	一般のプロトコルエラー	W	-
	0x8210	PDO プロトコル-長さのエラー	PDO が長すぎるため未処理		
	0x8220	PDO プロトコル- PDO 長エラー	PDO が長すぎるためエラー		
	0x8230	PDO プロトコル- MPDO エラー	MPDO DAM が処理されず、オブジェクト先が使用不可		
	0x8240	PDO プロトコル- SYNC データ長エラー	SYNC データ長が範囲外		
	0x8250	PDO プロトコル- RPDO タイムアウトエラー	RPDO タイムアウト		
Torque Profile Control	0x8300	トルクプロファイル	トルクプロファイルのプロファイルの一般的なエラー	F	6,6
	0x8341	トルクタイプ	選択されたタイプが扱われていないためエラー		
	0x8351	トルクダイナミックブレーキ	ダイナミックブレーキ未実装のエラー		
Velocity Speed Control	0x8400	速度プロファイル	速度プロファイルの一般的なエラー	F	6,7
	0x8410	電流モードの追従エラー	トルクプロファイルが選択されている場合、電流指令と検出電流の差が、設定されている最大電流エラー値より大きい。	F	5,10
	0x8411	速度モードの追従エラー	速度プロファイルが選択されている場合、速度指令と検出速度の差が、設定されている最大速度エラー値より大きい。		
	0x8412	オーバースピード	検出速度がオーバースピード値を超えた。	F	5,8

【2】 CANopen 運転

表 2-35 エマージェンシ内容(続き)

エラー	エラーコード	内 容	説 明	F/W	LEDコード
Position Controller	0x8500	ポジションコントローラ	ポジションコントローラのプロファイルの一般的なエラー	F	-
	0x8501	警告：リミット無効	警告：位置のリミット無効	W	-
	0x8502	警告：リミット	警告：目標位置がリミット内		
	0x8551	コンフィグレーションエラー	ポジションコントローラのコンフィグレーション設定エラー。製造コードに注意：番号 1,2,3,4	F	10,10
	0x8552	位置の有限状態機械	ポジションナの位置発生器の有限状態機械エラー		
Positioning Controller	0x8553	設定計算	設定計算中のエラー（加減速と速度プロファイル）	F	10,10
	0x8600	ポジショニングコントローラ	ポジショニングコントローラプロファイルの一般的なエラー		
Homing Controller	0x8651	設定エラー	ポジションコントローラのコンフィグレーション設定エラー。製造コードに注意：番号 1,2,3,4	F	6,8
	0x8652	ホーミング有限状態機械エラー	ホーミング発生器の有限状態機械エラー		
	0x8653	設定計算	設定計算中のエラー（加減速と速度プロファイル）		
CANopen EEPROM	0x8B00	ストアとリストア処理	ストアとリストア処理のゼネラルエラー	F	8,2
	0x8B01	パラメータのストア / リストア / ロードの警告	警告：ストア / リストア / ロード命令の無効ドライバは、「Ready to Switch ON」または「Disabled」のステータスでないため	V	-
	0x8B02	パラメータのストアエラー	オブジェクト(1010h)「パラメータストア」のマネジメントエラー	F	8,2
	0x8B03	メモリストア EEPROM フル	オブジェクトパラメータのメモリストアフルのエラー		
	0x8B04	EEPROM リストア	オブジェクト(1011h)「パラメータリストア」のマネジメントエラー		
	0x8B05	メモリリストア EEPROM フル	オブジェクトパラメータのメモリリストアフルのエラー		
	0x8B10	EEPROM から CANopen オブジェクト初期化		F	8,2
	0x8B11		0x6081		
	0x8B12		0x6082		
	0x8B13		0x6083		
	0x8B14		0x6084		
	0x8B15		0x60C5		
	0x8B16		0x60C6		
	0x8B17		0x607F		
	0x8B18		0x6088		
	0x8B19		0x6096		
	0x8B1A		0x6097		
	0x8B1B		0x606D		
	0x8B1C		0x606E		
	0x8B1D		0x606F		
	0x8B1E		0x6070		
	0x8B1F		0x6075		
	0x8B20	CANopen オブジェクト初期化	0x6076		
	0x8B21		0x6072		
	0x8B22		0x6073		
	0x8B23		0x60E0		
	0x8B24		0x60E1		
	0x8B25		0x6087		
	0x8B26		0x6086		
	0x8B27		0x607B		
	0x8B28		0x607D		
	0x8B29		0x6099		
	0x8B2A		0x609A		
	0x8B2B		0x607C		
	0x8B2C		0x6065		
	0x8B2D		0x6066		
	0x8B2E		0x6067		
	0x8B2F		0x6068		
	0x8B30		0x60F2		

表 2-36 エマージェンシ内容(続き)

エラー	エラーコード	内 容	説 明	F/W	LEDコード
DSP402 FSM	0x8C00	プロファイル 402 状態機械	プロファイル 402 状態機械の一般的なエラー	F	6,5
	0x8C01	運転モードエラー	運転モード(6060h)が、「Operation Enabled」ステート時に書込まれた。		
	0x8C02	プロファイルタイプ	プロファイルタイプが未定義		
	0x8C03	プロファイルエラー	選択されたプロファイルが、取扱われていない。		
	0x8C04	プロファイルなし	RUN ステートでプロファイルが未選択		
	0x8D00	アナログプロファイル	アナログプロファイルの状態エラー		

2-10 SYNC プロトコル

CAN ネットワーク上のデバイス(一般的にマスタコントローラ)は、周期的な SYNC メッセージを送信することにより、互いの同期をとることができます。SYNC オブジェクトは、CAN ネットワーク上で同期をとるためのメッセージです。

SYNC の通信の優先順位は高く、ジッタを最小とするためデータ情報を含んでいません。

SYNC オブジェクトは、PDO (送信タイプ:同期)のトリガとして使用されます。接続されたすべてのデバイスは、SYNC メッセージを受信することで PDO の処理が開始されます。

SYNC オブジェクトの識別子は、Index 1005h (初期設定値: 0x80)に設定されています。

ドライバは、SYNC メッセージを送信することはありません(受信のみ)。

マスタコントローラ(プロデューサ)から SYNC メッセージが送信されます。

表 2-37 SYNC メッセージ構造

COB-ID	Rx/Tx	DLC	バイト							
			0	1	2	3	4	5	6	7
0x80	Tx	0	データなし							

PDO (送信タイプ:同期)のトリガ

・TPDO: PDO のデータは、SYNC オブジェクトが受信されるまで送信されません。

2-11 エラーコントロールプロトコル

CANopen ネットワークのモニタ用途で、以下の2つのエラーコントロールプロトコルが実装されています。

- (1) ハートビート
- (2) ノード/ライフガーディング

ハートビートプロトコルは、マスタコントローラが全ての CAN ネットワークに接続されたデバイスを監視し、通信状態を確認するため使用されます。

古い CANopen システムでは、ハートビートプロトコルの代わりに CAN リモートフレームベースのノード/ライフガーディングプロトコルが使用されることもあります。

全てのエラーコントロールプロトコルは、「COB-ID 0x700 + ノード ID」で定義されています。

【2】 CANopen 運転

⚠ 注意

CANopen 通信のモニタとして、少なくとも 1 つはエラーコントロールプロトコルを使用することを推奨します。

CiA (CAN in Automation) は、RTR フレームを利用したガーディングを新しく実装するためにハートビートを使用することを推奨しています。

CiA は CAN リモートフレームをベースとしたサービスの使用を推奨していません。

注) 出荷時は、エラーコントロールプロトコルとしてハートビートが有効 (200ms 毎) に設定されています。

■ ノードガーディングプロトコル

マスタとスレーブが互いを監視して、通信異常を検出する機能です。マスタが COB-ID (700h + ノード ID) の RTR メッセージをスレーブへ送信し、メッセージを受信したスレーブが応答メッセージとして同一の COB-ID (700h + ノード ID) のメッセージをマスタへ送信します。

ただし、ノードガーディングは、CAN バスに高負荷がかかるためハートビートの使用を推奨します。

応答メッセージの構造を次の表に示します。

表 2-38 ノードガーディングメッセージ構造

COB-ID	Rx/Tx	DLC	バイト							
			0	1	2	3	4	5	6	7
0x700 + ノード ID	Tx	1	7 ビットトグル + NMT ステート	-						

ビット 7 は、メッセージ転送毎に 1 と 0 が交互に変わります。メッセージが消失していないかを判断するために使用されます。

ビット 6 ~ 0 は、スレーブの現在の NMT ステートを定義します。

ノードガーディング機能は、3 つの時間を設定する必要があります。

- ・ ガードタイム (Guard time) : 連続する RTR メッセージの受信インターバル時間
スレーブ毎に設定が可能で、オブジェクト (100Ch) 「ガードタイム」 (単位: ms) で設定します。
- ・ ライフタイムファクタ : ガードタイムで設定した時間に乗算するための係数
スレーブ毎に設定が可能で、オブジェクト (100Dh) 「ライフタイムファクタ」で設定します。
- ・ 可能なライブタイム (Possible live time) : ガードタイムとライフタイムファクタの乗算結果です。

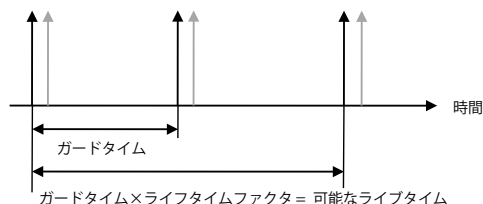


図 2-13 ノードガーディングのタイムメッセージ

【2】CANopen 運転

ノードガーディング機能は、以下の条件を満たす必要があります。

- ・マスタは「可能なライブタイム」時間内に、次の RTR メッセージを送信してください。
- ・スレーブは「可能なライブタイム」時間内に、応答(メッセージを返信)する必要があります。
- ・スレーブは、トグル(ビット 7)と NMT ステートと(ビット 6～0)を正しく返信する必要があります。

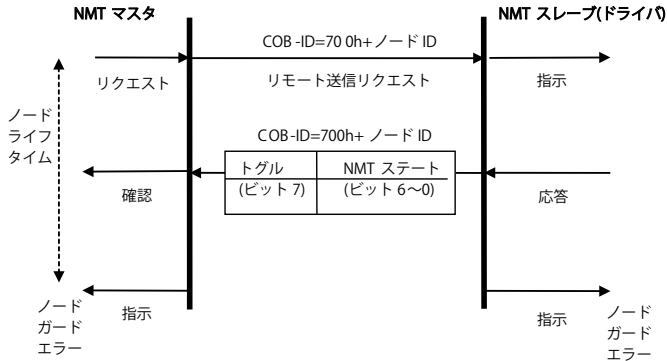


図 2-14 ノードガーディング・タイムフレーム

■ハートビートプロトコル

マスタが各スレーブとの通信状態を監視するため機能です。

各スレーブは、周期的にマスタコントローラへハートビートメッセージを送信します。

マスタは、各スレーブからのハートビートメッセージが送られてくることを周期的に確認しています。

マスタは、ハートビートタイムの設定時間内にメッセージを受信できなかった場合、異常を検出します。

ハートビートメッセージは、COB-ID (700h + ノード ID) で送信され、NMT ステートを含んでいます。

表 2-39 ハートビートメッセージ構造

COB-ID	Rx/Tx	DLC	バイト							
			0	1	2	3	4	5	6	7
0x700 + ノード ID	Tx	1	NMT ステート	-						

【2】 CANopen 運転

以下は、ハートビートの動作を示しています。

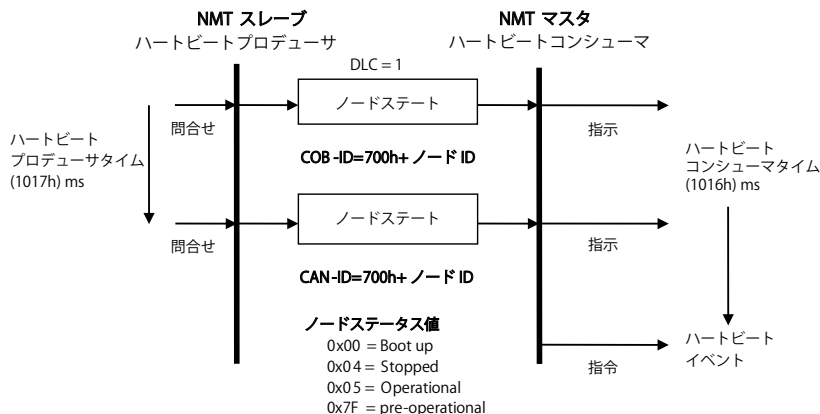


図 2-15 ハートビート・タイムフレーム

「プロデューサハートビートタイム」(1017h)の設定値が0以外のとき、ハートビート機能が有効になります。マスタとスレーブ間の設定は、オブジェクトで変更することができます。マスタ側で設定するハートビートタイム(1016h)の時間内に、信号を受信できなければ、エラーメッセージ(ハートビートイベント)が発生します。スレーブ側の「ハートビートタイム」(1016h)が0の場合、コンシューマがモニタできます。

注)詳細は、11 章 各種の設定の「11-2 ハートビート・メカニズム」を参照してください。

2-12 CAN 通信エラー

CAN 通信でエラーが発生した場合、EMCY メッセージを送信します。
ドライバは、CAN バスとの通信が遮断されると FAULT ステートへ移行します。

CAN 通信の状態をモニタするためのオブジェクトを以下に示します。

- 4100h : CAN 通信ステート
- 4101h : CAN 通信カウンタ
- 4102h : CAN 通信設定
- 4103h : CAN 通信エラー

「CAN 通信ステート」(4100h: 5)で現在の「CAN ドライバステート」を読み込むことができます。

- INIT ステート : 0x001
- ACTIVE ステート : 0x002
- BUSOFF ステート : 0x004
- PASSIVE ステート : 0x008

送受信メッセージのエラー回数は、「CAN 通信ステート」(4100h)の Sub-index 1 と 2 に保存されます。

- **CAN オーバーラン(オブジェクト喪失)**

メッセージが喪失された場合、マスタがエマージェンシメッセージ(0x8110)を送信します。
「CAN 通信ステート」(4100h: 3)の「現状フラグエラー」には、「オーバーランエラー」(ビット 4)をビットセットします。

- **CAN エラー Passive モード**

ドライバが Passive エラーステートの場合、エマージェンシメッセージ(0x8120)を送信します。
「CAN 通信ステート」(4100h: 3)の「現状フラグエラー」には、「Passive エラー」(ビット 7)をビットセットします。

- **CAN Bus Off**

ドライバが Bus Off ステートの場合、エマージェンシメッセージ(0x8180)を送信します。
「CAN 通信ステート」(4100h: 3)の「現状フラグエラー」には、「Bus Off」(ビット 8)をビットセットします。
「CAN 通信カウンタ」(4101h: 1)の「Bus Off カウンタ」には、Bus Off が発生した回数が保存されます。

- **Bus Off からリカバリ**

ドライバが Bus Off ステートになっている間に、正常にメッセージを受信していれば、エマージェンシメッセージ(0x8140)を送信します。
「CAN 通信ステート」(4100h: 3)の「現状フラグエラー」には、「Bus Off リカバリ」(ビット 9)をビットセットします。
「CAN 通信カウンタ」(4101h: 2)の「Bus Off リカバリカウンタ」には、Bus Off からリカバリした回数が保存されます。

【2】 CANopen 運転

2-13 ネットワークマネジメント (NMT)

ネットワークマネジメント (NMT) は、アプリケーション階層のサービスの一つです。

NMT は、ネットワーク内でコンフィグレーション、初期化、エラー処理を行います。

NMT 命令は、ドライバの通信ステートを制御するために使用され、全てのノードへメッセージを返信します。NMT スレーブは、ネットワーク内の個別のノード ID (1 ~ 127 の範囲) で識別されます。

ドライバは、電源投入後またはリセット後すぐに NMT ステート “Pre-operational” へ移行します。

NMT ステート “Pre-operational” では、SDO による設定値の変更が可能です。

ネットワークマネジメント (NMT) は、ノード重視のマスタ - スレーブ構造です。

NMT オブジェクトは、NMT サービスの実行に使用されます。NMT サービスを通じて、ノードの初期化、スタート、モニタ、リセット、停止が行われます。全てのノードは、NMT スレーブとして見なされます。

NMT では、ネットワーク内で一つのデバイスが NMT マスタの役割を満たすことが要求されます。

■ NMT サービス

• モジュールコントロールサービス (Module Control Services)

モジュールコントロールサービスを通じて、NMT マスタは、NMT スレーブのステートを制御します。

NMT ステートは、“Stopped”、“Pre-operational”、“Operational”、“Initializing” の 4 つです。

• エラーコントロールサービス (Error Control Service)

エラーコントロールサービスを通じて NMT は、CAN ネットワークの異常を検出します。

エラーコントロールを実行する機能は、ノードガーディングまたはハートビートの 2 種類あります。

• ブートアップサービス (Boot-up Service)

ブートアップサービスを通じて、NMT スレーブはステートが初期化 (Initializing) ステートから運転準備 (Pre-operational) ステートへ遷移したことを示します。

■ NMT 状態機械

ドライバの電源投入後またはリセット後、初期化 (Initializing) ステートで内部の処理を実行し、運転準備 (Pre-operational) へ自動的に遷移します。

【2】 CANopen 運転

次の図は、smartris ドライバの NMT ステートを示しています。

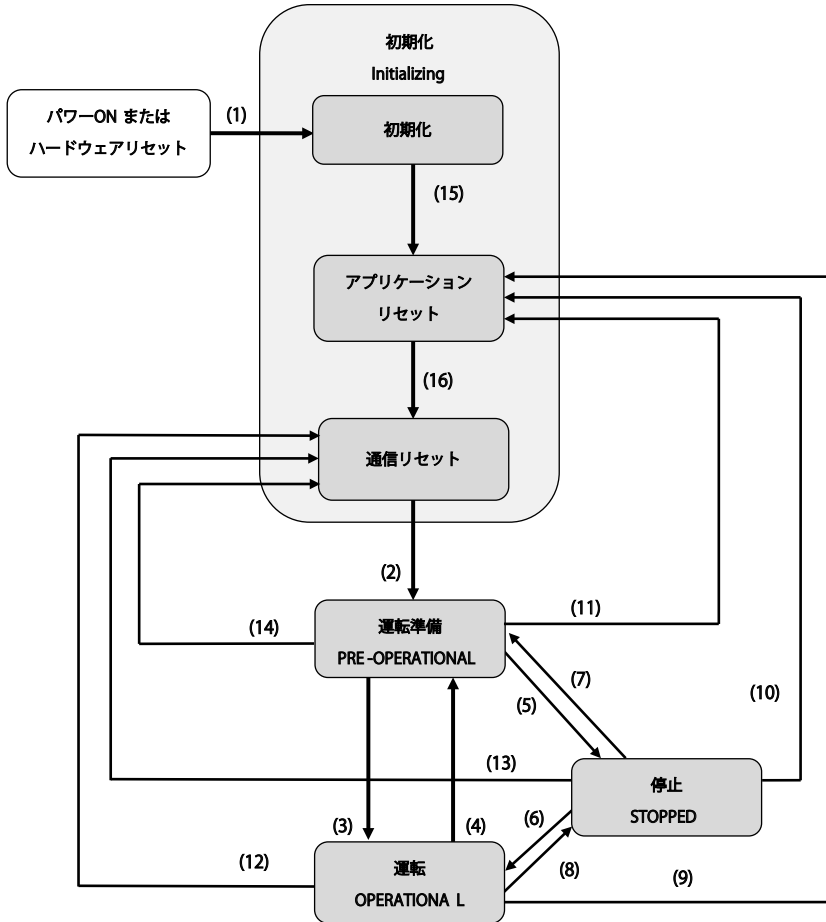


図 2-16 NMT 状態機械

次の表は、遷移の内容を示しています。

表 2-40 NMT ネットワークマネジメント

遷 移	内 容
(1), (15), (16)	パワー ON で NMT ステートの初期化が自動的に開始・遷移
(2)	NMT ステートの初期化完了：“Pre-operational”へ自動的に遷移
(3)	NMT サービス：リモートノード開始指示かローカル制御
(4), (7)	NMT サービス：“Pre-operational”に遷移
(5), (8)	NMT サービス：リモートノード停止指示
(6)	NMT サービス：リモートノード開始指示
(9), (10), (11)	NMT サービス：アプリケーションリセット指示
(12), (13), (14)	NMT サービス：通信リセット指示

[2] CANopen 運転

次の表に示す通信タイプで、ネットワーク通信ステートがサポートされています。

表 2-41 NMT ネットワークマネジメント

ステート	内 容	SDO	PDO	NMT	SYNC
初期化 (Initialization)	ドライバは、ブート中のため準備未完了状態です。 通信は応答せずに何も送信しません。	-	-	-	-
運転準備中 (Pre-operational)	ドライバは、ブートシーケンスが完了し、準備状態です。 SDO と NMT には応答しますが、PDO に応答しません。	○	-	○	○
運転 (Operational)	ドライバは、PDO、SDO、NMT メッセージに応答します。	○	○	○	○
停止 (Stopped)	ドライバは、NMT オブジェクトのみに応答します。 (ハートビートを含む)	-	-	○	-

■ネットワーク初期化

電源をオンすると、ネットワークマネジメント (NMT) の状態機械が開始します。

内部リセットか電源投入後の最初のステートは、NMT 初期化ステートです。

このステートでは、ドライバ内部の EEPROM から RAM へ全てのパラメータデータをロードします。NMT 初期化ステート終了後、ドライバは、“Pre-operational” ステートに遷移してブートアップメッセージを送信します。

NMT 初期化ステートでは、CANopen デバイスの完全または部分的なリセットを有効にするため、以下の 3 つの NMT サブステートに分かれます。

• 初期化 “Initialization” ステート

電源投入またはリセット後に、CANopen デバイスが最初に遷移する NMT ステートです。

CANopen デバイスの基本的な初期化が終了した後、CANopen デバイスは自動的に NMT サブステートのアプリケーションリセットに入ります。

• アプリケーションリセット

この NMT サブステートで、メーカー固有のプロファイルと標準デバイスプロファイルのパラメータを設定します。設定完了後に NMT サブステートの通信リセットへ自動的に遷移します。

• 通信リセット

NMT サブステートで通信プロファイルエリアのパラメータが設定されます。

電源投入後のパラメータ設定は、直近でストアされた値です。ストアが実行されなかった場合や、リストア命令の失敗が発生することによってリセット動作が先行した場合、初期値が設定されます。

その他の NMT ステートは、以下の通りです。

• ネットワーク “Pre-Operational” ステート

“Pre-operational” ステートでは、SDO での通信は可能ですが、PDO での通信は許可されていません。

エマージェンシオブジェクトやハートビートメッセージなどのエラーコントロールサービスが、このステートで開始されます。

ノードは、NMT「リモートノードスタート」を送信することで “Operational” ステートへ遷移します。

• ネットワーク “Operational” ステート

“Operational” ステートは、全ての通信オブジェクトが有効になります。(PDO 処理を含む)

SDO を介してオブジェクトディクショナリにアクセスすることもできます。

【2】CANopen 運転

・ネットワーク “Stopped” ステート

“Stopped” ステートにデバイスを切替えることにより、ノードガーディングとハートビート(有効な場合)以外の通信を強制的に停止します。

■ NMT メッセージ

NMT メッセージは、以下の構造で 2 バイトのデータを含んでいます。

表 2-42 NMT メッセージ構造

COB-ID	Rx/Tx	DLC	バイト							
			0	1	2	3	4	5	6	7
0x00	Rx	2	コマンド	ノード ID				-		

バイト 0 の「コマンド」には、以下のコマンドで NMT ステートを遷移させることができます。

表 2-43 バイト 0 の「コマンド」の定義

コマンド	意 味	内 容	遷 移	目標ステート
0x01	リモートノード 開始	NMT マスタは、このサービスを通して選択された NMT スレーブのステートを “Operational” に遷移させます。	(3) (6)	OPERATIONAL
0x02	リモートノード 停止	NMT マスタは、このサービスを通して選択された NMT スレーブのステートを “Stopped” に遷移させます。	(5) (8)	STOPPED
0x80	「Pre-Operational」 ステートへ移行	NMT マスタは、このサービスを通して選択された NMT スレーブのステートを “Pre-operational” に遷移させます。	(4) (7)	PRE-OPERATIONAL
0x81	アプリケーション リセット	NMT マスタは、このサービスを通して選択された NMT スレーブのステートをあらゆるステートからサブステート「アプリケーションリセット」に遷移させます。	(9) (10) (11)	RESET APPLICATION
0x82	通信リセット	NMT マスタは、このサービスを通して選択された NMT スレーブのステートをあらゆるステートからサブステート「通信リセット」に遷移させます。 設定後、選択されたリモートノードは、通信リセットされます。	(12) (13) (14)	RESET COMMUNICATION

バイト 1 の数値定義

表 2-44 NMT の内容

ノード ID	内 容
ノード ID	全てのデバイスに対して送信する場合：0x00h を設定（グローバルモード） 特定のデバイスに対して送信する場合：各ノード ID (0x01 ～ 0x7F) を設定

■ ブートアップメッセージ (Bootup Message)

電源投入後またはリセット後、初期化完了のブートアップメッセージをマスタコントローラへ送信します。

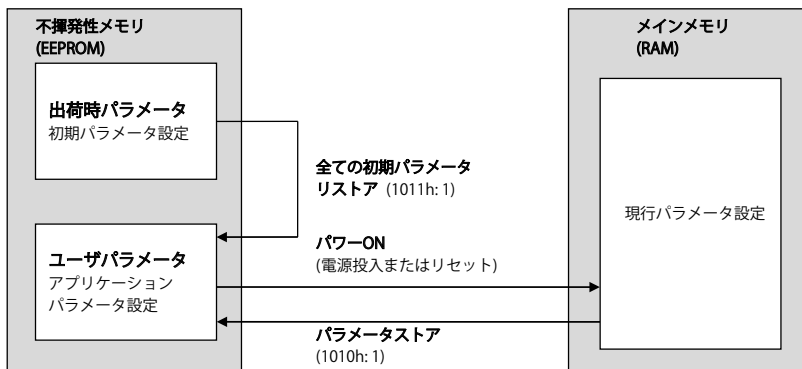
このメッセージが送信された後、NMT ステートは “Pre-operational” へ遷移します。

表 2-45 初期化完了のブートアップメッセージ構造

COB-ID	Rx/Tx	DLC	バイト							
			0	1	2	3	4	5	6	7
0x700 + ノード ID	Tx	1	0x00				-			

【2】 CANopen 運転

2-14 ストアとリストア



CANopen では、ストアとリストアの 2 つのオブジェクトを定義しています。

- ・「パラメータストア」(1010h: 1)
- ・「パラメータリストア」(1011h: 1)

「パラメータストア」(1010h: 1)へ“save”を ASCII コードで書込むことで、全てのパラメータ設定値が EEPROM へストアされます。

以下は、「パラメータストア」(1010h)の Sub-Index:2 (通信パラメータ)に“save”を書込むことで EEPROM にストアされるオブジェクトです。

- ・100Ch: ガードタイム 注)
- ・100Dh: ライフタイムファクタ 注)
- ・1017h: ハートビートタイムプロデューサ 注)

注) ドライバを運転中でも変更可能なオブジェクトです。

以下は、「パラメータストア」(1010h)の Sub-Index: 3 (アプリケーションパラメータ)に“save”を書込むことで EEPROM にストアされるオブジェクトです。

- ・6065h: 追従エラーウィンドウ・・・(未実装)
- ・6066h: 追従エラータイムアウト・・・(未実装)
- ・6067h: 位置ウィンドウ・・・(未実装)
- ・606Dh: 速度ウィンドウ
- ・606Eh: 速度ウィンドウ時間
- ・606Fh: 速度スレッシュホールド
- ・6070h: 速度スレッシュホールド時間
- ・6068h: 位置ウィンドウタイム・・・(未実装)
- ・6072h: 最大トルク・・・(未実装)
- ・6073h: 最大トルク電流
- ・6075h: モータ定格電流
- ・6076h: モータ定格トルク・・・(未実装)
- ・607Bh: 位置範囲リミット・・・(未実装)
- ・607Ch: ホームオフセット・・・(未実装)
- ・607Dh: ソフトウェア位置リミット・・・(未実装)
- ・607Eh: 極性注)
- ・607Fh: 最大速度プロファイル
- ・6080h: 最高モータ速度
- ・6081h: 速度プロファイル・・・(未実装)
- ・6082h: 速度エンド・・・(未実装)

- ・6083h: 加速プロファイル 注)
 - ・6084h: 減速プロファイル 注)
 - ・6086h: 運転プロファイルタイプ
 - ・6087h: トルクスロープ 注)
 - ・6088h: トルクプロファイルタイプ
 - ・6096h: 速度ファクタ
 - ・6097h: 加速ファクタ
 - ・6099h: ホーミング速度・・・(未実装)
 - ・609Ah: ホーミング加速・・・(未実装)
 - ・60C5h: 最大加速
 - ・60C6h: 最大減速
 - ・60F2h: 位置決めオプションコード・・・(未実装)
- 注) ドライバを運転中でも変更可能なオブジェクトです。

以下は、「パラメータストア」(1010h)の Sub-Index:4 (出荷時パラメータ)に“save”を書込むことで EEPROM にストアされるオブジェクトです。

- ・2000h: ノード ID
- ・2001h: ボーレート
- ・3002h: ブレーキパラメータ 注)
- ・3005h: フィルタパラメータ 注) (Sub. 5,6h のみ)
- ・3007h: ダイナミックブレーキパラメータ 注)
- ・3008h: エマージェンシ入力有効パラメータ 注)
- ・3050h: アナログ出力 1 注)
- ・3100h: コンフィグレーション(ステータスワード) 注)
- ・3200h: Pid 電流 注)
- ・3201h: Pid 速度 注)
- ・3202h: Pid ポジショナ 注)・・・(未実装)
- ・3300h: 速度フルスケール
- ・4102h: CAN 通信設定

注) ドライバを運転中でも変更可能なオブジェクトです。

全てのパラメータは、EEPROM ヘストアできますが、ドライバの再起動または NMT リセットメッセージが送信されるまで反映されません。

誤ったパラメータを EEPROM ヘストアした場合、出荷時パラメータをロードすることで修復できます。マスタから「パラメータリストア」(1011h)に“load”を書込み、送信してください。

■出荷時パラメータのリストア

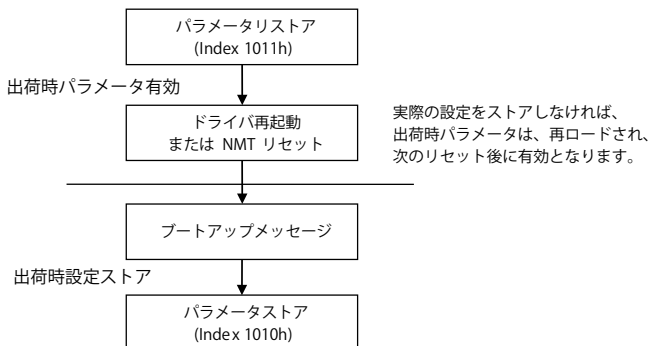


図 2-17 パラメータリストアのフローチャート

【2】 CANopen 運転

次の表に初期設定の通信パラメータを示します。

表 2-46 通信パラメータ

名 称	Index	Sub-Index	数値フィールド	初期パラメータ
P301 DEV TYPE	0x1000	0	デバイスタイプ	0xFF7A0192
P301 ERR REG	0x1001	0	エラーレジスタ	0
P301 MANUF STATUS REG	0x1002	0	メーカー固有ステータスレジスタ	0
P301 PREDEF ERR FIELD	0x1003	0	エントリ数	15
		1 ~ 15	履歴 [1] ~ [15]	0
P301 COBID SYNC	0x1005	0	COB-ID SYNC	COB-ID = 80000080h +ID
P301 GUARD TIME	0x100C	0	ガードタイム	0 = 無効
P301 LIFETIME FACTOR	0x100D	0	ライフタイムファクタ	0 = 無効
P301 COBID EMERGENCY	0x1014	0	COB-ID EMCY	COB-ID = 80h +ID
P301 PRODUCER HB TIME	0x1017	0	プロデューサハートビートタイム	200 ms
P301 IDENTITY OBJECT	0x1018	0	エントリ数	4
		1	ベンダ ID	0x01FB
		2	製品コード	0
		3	改訂番号	0
		4	シリアル番号	0
P301 ERR BEHAVIOR	0x1029	0	エントリ数	1
		1	通信エラー	0
P301 RXPDO 1 PARAM	0x1400	0	エントリ数	5
		1	COB-ID	201h、PDO 有効
		2	送信タイプ	0xFE = イベントドリブン
		3	禁止時間	0 = 無効 (us)
		4	適合性エントリ	0 = 使用不可
		5	イベントタイム	0 = 無効 (ms)
P301 RXPDO 2 PARAM	0x1401	0	エントリ数	5
		1	COB-ID	301h、PDO 有効
		2	送信タイプ	0xFE = イベントドリブン
		3	禁止時間	0 = 無効 (us)
		4	適合性エントリ	0 = 使用不可
		5	イベントタイム	0 = 無効 (ms)
P301 RXPDO 3 PARAM	0x1402	0	エントリ数	5
		1	COB-ID	401h、PDO 有効
		2	送信タイプ	0xFE = イベントドリブン
		3	禁止時間	0 = 無効 (us)
		4	適合性エントリ	0 = 使用不可
		5	イベントタイム	0 = 無効 (ms)
P301 RXPDO 4 PARAM	0x1403	0	エントリ数	5
		1	COB-ID	501h、PDO 有効
		2	送信タイプ	0xFE = イベントドリブン
		3	禁止時間	0 = 無効 (us)
		4	適合性エントリ	0 = 使用不可
		5	イベントタイム	0 = 無効 (ms)
P301 RXPDO 1 MAPPING	0x1600	0	エントリ数	3
		1	マッピングエントリ 1	0x60400010 = コントロールワード
		2	マッピングエントリ 2	0x60600008 = 運転モード
		3	マッピングエントリ 3	0x60FE0120 = デジタル出力
		4 to 8	マッピングエントリ 4 ~ 8	0
P301 RXPDO 2 MAPPING	0x1601	0	エントリ数	2
		1	マッピングエントリ 1	0x60400010 = コントロールワード
		2	マッピングエントリ 2	0x607A0020 = 目標位置
		3 to 8	マッピングエントリ 3 ~ 8	0
P301 RXPDO 3 MAPPING	0x1602	0	エントリ数	2
		1	マッピングエントリ 1	0x60400010 = コントロールワード
		2	マッピングエントリ 2	0x60FF0020 = 目標速度
		3 to 8	マッピングエントリ 3 ~ 8	0
P301 RXPDO 4 MAPPING	0x1603	0	エントリ数	2
		1	マッピングエントリ 1	0x60400010 = コントロールワード
		2	マッピングエントリ 2	0x60710010 = 目標トルク
		3 to 8	マッピングエントリ 3 ~ 8	0

【2】CANopen 運転

表 2-47 通信パラメータ (続き)

名 称	Index	Sub-Index	数値フィールド	初期パラメータ
P301 TXPDO 1 PARAM	0x1800	0	エントリ数	5
		1	COB-ID	181h、PDO 有効
		2	送信タイプ	0xFF = イベントドリブン
		3	禁止時間	0x05 = 500us
		4	適合性エントリ	0= 使用不可
		5	イベントタイマ	0= 無効 (ms)
P301 TXPDO 2 PARAM	0x1801	0	エントリ数	5
		1	COB-ID	281h、PDO 有効
		2	送信タイプ	0xFD = RTR-only
		3	禁止時間	0x05 = 500us
		4	適合性エントリ	0= 使用不可
		5	イベントタイマ	0= 無効 (ms)
P301 TXPDO 3 PARAM	0x1802	0	エントリ数	5
		1	COB-ID	381h、PDO 有効
		2	送信タイプ	0xFD = RTR-only
		3	禁止時間	0x05 = 500us
		4	適合性エントリ	0= 使用不可
		5	イベントタイマ	0= 無効 (ms)
P301 TXPDO 4 PARAM	0x1803	0	エントリ数	5
		1	COB-ID	481h、PDO 有効
		2	送信タイプ	0xFD = RTR-only
		3	禁止時間	0x05 = 500us
		4	適合性エントリ	0= 使用不可
		5	イベントタイマ	0= 無効 (ms)
P301 TXPDO 1 MAPPING	0x1A00	0	エントリ数	3
		1	マッピングエントリ 1	0x60410010 = ステータスワード
		2	マッピングエントリ 2	0x60610008 = 運転モード表示
		3	マッピングエントリ 3	0x60FD0020 = デジタル入力
		4	マッピングエントリ 4 ~ 8	0
P301 TXPDO 2 MAPPING	0x1A01	0	エントリ数	2
		1	マッピングエントリ 1	0x60410010 = ステータスワード
		2	マッピングエントリ 2	0x60640020 = 位置検出値
		3	マッピングエントリ 3 ~ 8	0
P301 TXPDO 3 MAPPING	0x1A02	0	エントリ数	2
		1	マッピングエントリ 1	0x60410010 = ステータスワード
		2	マッピングエントリ 2	0x606C0020 = 速度検出値
		3	マッピングエントリ 3 ~ 8	0
P301 TXPDO 4 MAPPING	0x1A03	0	エントリ数	2
		1	マッピングエントリ 1	0x60410010 = ステータスワード
		2	マッピングエントリ 2	0x60770010 = トルク検出値
		3	マッピングエントリ 3 ~ 8	0

2-15 識別子

次の表に、使用されている識別子 (COB-ID) の一覧を示します。

表 2-48 識別子

オブジェクトタイプ	識別子 (16 進数)	オブジェクトタイプ	識別子 (16 進数)
SDO (マスタからドライバ)	0x600 + ノード ID	RPDO3	0x400 + ノード ID
SDO (ドライバからマスタ)	0x580 + ノード ID	RPDO4	0x500 + ノード ID
TPDO1	0x180 + ノード ID	SYNC	0x80
TPDO2	0x280 + ノード ID	EMCY	0x80 + ノード ID
TPDO3	0x380 + ノード ID	HEARTBEAT	0x700 + ノード ID
TPDO4	0x480 + ノード ID	BOOTUP	0x700 + ノード ID
RPDO1	0x200 + ノード ID	NMT	0x00
RPDO2	0x300 + ノード ID	-	-

【2】 CANopen 運転

2-16 プロファイル DSP402

■プロファイル DSP402 状態機械

ドライバは、DSP402 による状態機械に基づいて制御・遷移しています。

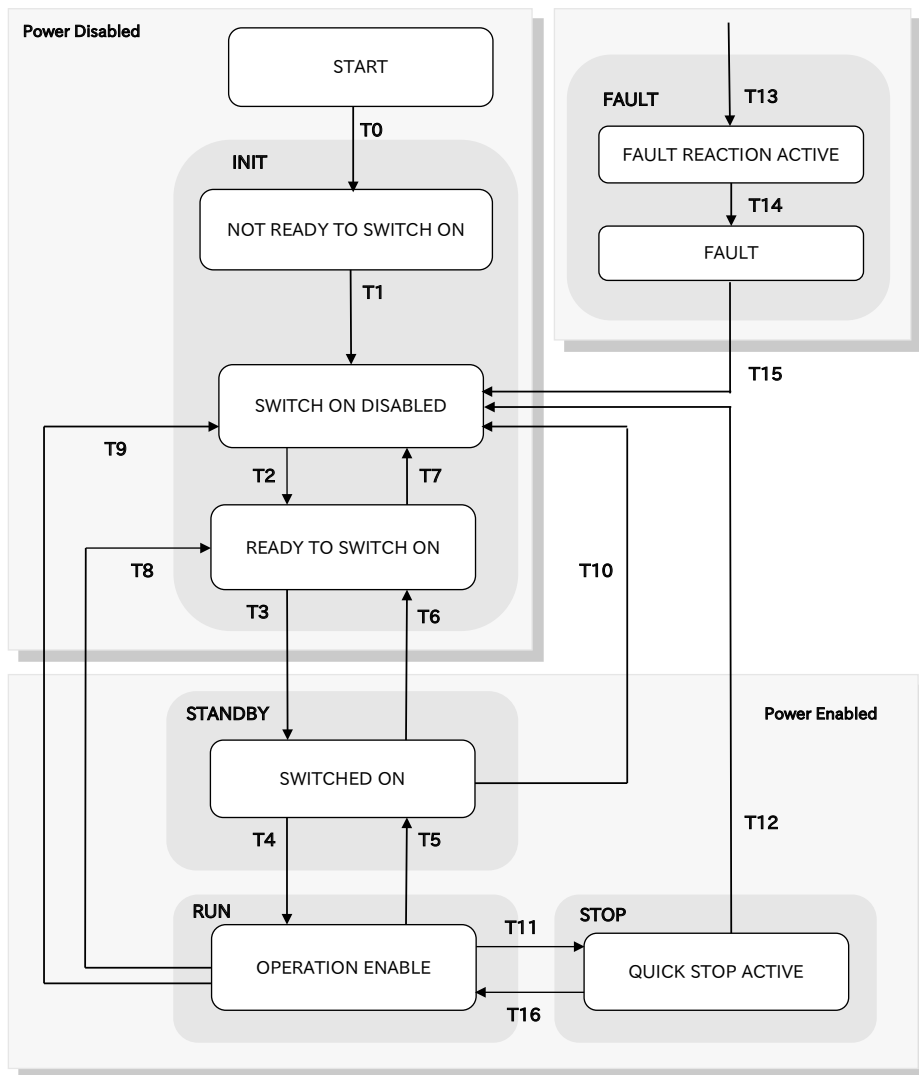


図 2-18 状態機械 DSP402

注) 詳細な情報は、「CiA DSP402 standard」を参照してください。

表 2-49 ステータスワード

ステータス	内 容
NOT READY TO SWITCH ON	INIT ステート： <ul style="list-style-type: none"> ・ドライバが初期化中で、セルフテスト中 ・ドライバの機能無効 ・このステートは、最後でのみ通信が有効となる内部ステートです。 ・ユーザは、このステートを取得することもモニタすることも不可
SWITCH ON DISABLED	INIT ステート： <ul style="list-style-type: none"> ・出力電圧オフ ・ドライバの初期化完了 ・ドライバのパラメータのセットアップ ・ドライバのパラメータの変更可能 ・ドライバの機能無効（サーボ OFF 状態） ・パラメータを EEPROM にセーブ可能 ・“SWITCH ON DISABLED” は、ユーザが切替える最低限のステートです。
READY TO SWITCH ON	INIT ステート： <ul style="list-style-type: none"> ・出力電圧オフ ・ドライバの初期化完了 ・ドライバのパラメータのセットアップ ・ドライバのパラメータの変更可能 ・ドライバの機能無効（サーボ OFF 状態） ・パラメータを EEPROM にセーブ可能
SWITCHED ON	STANDBY ステート： <ul style="list-style-type: none"> ・出力電圧オフ ・RUN ステート（Operation Enable）への準備完了 ・ドライバのパラメータの変更可能 ・ドライバの機能無効（サーボ OFF 状態） ・パラメータを EEPROM にセーブ可能
OPERATION ENABLE	RUN ステート： （ドライバの正常運転に対応） <ul style="list-style-type: none"> ・故障（Fault）検出なし ・モータに動力を供給 ・ドライバ機能が有効（サーボ ON 状態） ・ドライバのパラメータの変更可能 ・自動ブレーキが有効な場合、ブレーキパラメータのタイミングに従ってブレーキを解放 ・EEPROM へのパラメータのセーブとリストア不可
QUICK STOP ACTIVE	STOP ステート： <ul style="list-style-type: none"> ・故障（Fault）検出なし ・モータに動力を供給 ・ドライバ機能が有効（サーボ ON 状態） ・ドライバのパラメータの変更可能 ・ドライバが動作を停止し、急停止となり保持トルクを発生 ・EEPROM へのパラメータのセーブとリストア不可
FAULT REACTION ACTIVE	FAULT ステート： <ul style="list-style-type: none"> ・ドライバのパラメータの変更可能 ・ドライバで故障（Fault）が発生 ・故障処理の実行 ・ドライバ機能が無効（サーボ OFF 状態） ・ユーザによるパラメータ取得不可 ・パラメータを EEPROM にセーブ可能
FAULT	FAULT ステート： <ul style="list-style-type: none"> ・ドライバのパラメータの変更可能 ・ドライバで故障（Fault）が発生 ・ドライバ機能が無効（サーボ OFF 状態） ・パラメータを EEPROM にセーブ可能 ・ドライバは、Fault アラームのリセットの受入れ可

【2】 CANopen 運転

ステートの変更は、エラー発生やコントロールワード(6040h)の変更など内部イベントによってトリガされます。オブジェクト(6041h)「ステータスワード」は、現在のステートのフィードバックとして使用可能です。ドライバの電源投入後に初期化処理が自動的に行われ“Switch ON Disabled”ステートへ切替わります。“Switch ON Disabled”ステートでは、ドライバはコントロールワードの命令を待機している状態です。コントロールワードが“Operation Enabled”ステートまで遷移すると、ドライバは、運転開始します。

セーフティステートは、DSP402 には実装されていません。セーフティは、緊急事態が発生した場合、ドライバおよび装置全体を保護するために定義されました。(4章 セーフティを参照)

ステータスワードでセーフティステートを読み込むことができます。

次の表は、ドライバ表面の LED と状態機械のステートを示しています。

表 2-50 ドライバステータス

ドライバステート	CANopen ステート	ステータス 1 LED 緑色	ステータス 2 LED 黄色	LED 表示
初期状態 (INIT)	スイッチ ON 準備中 (Not Ready to Switch ON)	“点滅”	“点滅”	1 同時に点滅 2 同時に点滅
	スイッチ ON 無効 (Switch ON Disabled)	“点滅” 交互	“点滅” 交互	1 交互に点滅 2 交互に点滅
	スイッチ ON 準備完了 (Ready to Switch ON)			
運転準備 (STANDBY)	スイッチ ON (Switched ON)	“点滅”	OFF	1 点滅 50% 2 消灯
故障 (FAULT)	故障 (Fault) 故障処理中 (Fault Reaction Fault)	“点滅” コード [x]	“点滅” コード [y]	1 9 章を参照 2
運転 (RUN)	運転有効 (Operation Enabled)	ON	OFF	1 点灯 2 消灯
停止 (STOP)	急停止動作 (Quick Stop Active)	ON	ON	1 点灯 2 点灯
セーフティ (SAFETY)	-	OFF	“点滅”	1 消灯 2 点滅
CAN 通信エラー (COMMUNICATION CAN ERROR)	-	OFF	ON	1 消灯 2 点灯

2-17 運転モード

以下の運転モードが smartris ドライバで使用できます。

・位置プロファイルモード(未実装)

エンコーダやレゾルバのフィードバック値(位置検出値)と位置指令が一致するように制御します。smartris ドライバの位置ループは、モータトルクをより多く発生させることで位置誤差を補正します。

・速度プロファイルモード

コントローラからの速度指令値を受けて、ドライバの出力電圧および出力周波数を計算します。速度指令値と加減速度を設定してください。

・トルクプロファイルモード

コントローラからのトルク(電流)指令を受けて、ドライバの出力電圧および出力周波数を計算します。

トルクスロープとトルクプロファイルパラメータを設定してください。

【2】 CANopen 運転

■速度プロファイルの運転シーケンス

運転シーケンスを次のフローチャートに示します。

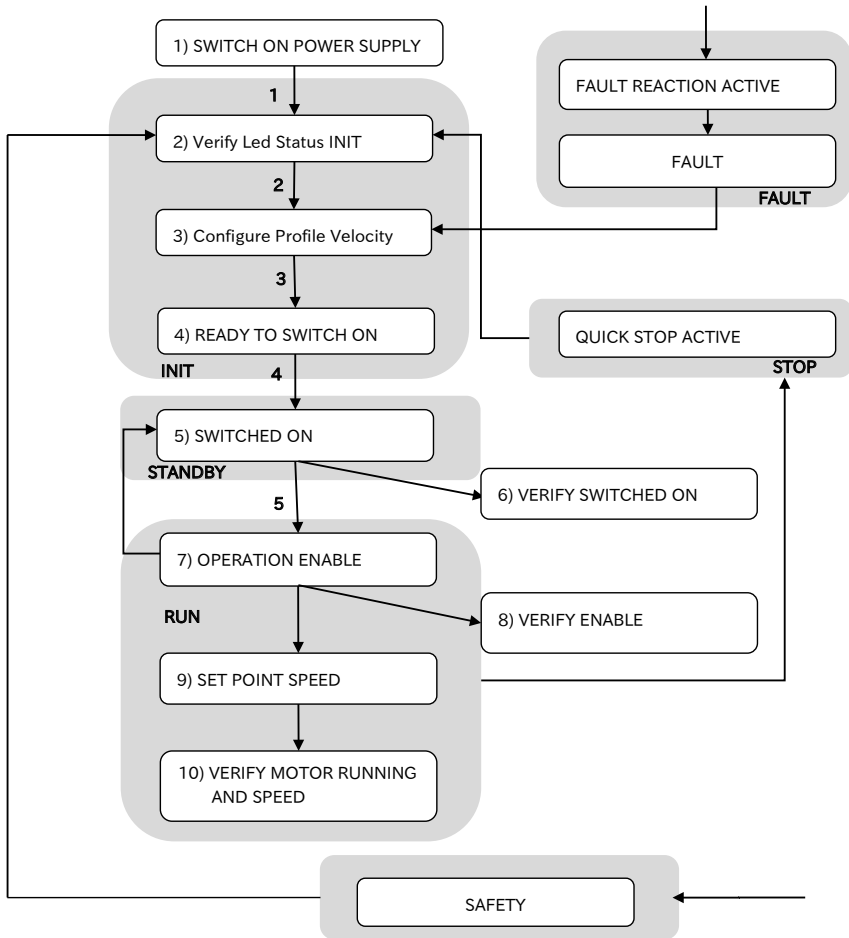


図 2-19 CANopen 速度プロファイルの運転シーケンス

- **STO (SAFETY)** 命令は、運転指令を直ぐに遮断し、フリーラン停止します。
- **STOP** 命令は、運転指令を直ぐに遮断し、サーボ ON 状態で停止します。
- **FAULT** (9 章 診断を参照) では、運転指令を直ぐに遮断し、メカブレーキが動作することがあります。

【2】 CANopen 運転

■モータを運転させるまでの CAN コマンドのシーケンス

- ドライバの電源 ON
- LED ステータスで初期状態 (INIT) を確認
- 速度プロファイル (6060h) の設定 → 0x03
CAN コマンド 2F 60 60 00 03 00 00 00
- “READY TO SWITCH ON” ステートを設定
コントロールワード (6040h) に書込み → 0x06
CAN コマンド 2B 40 60 00 06 00 00 00
- “SWITCHED ON” ステートを設定
コントロールワード (6040h) に書込み → 0x07
CAN コマンド 2B 40 60 00 07 00 00 00
- ドライバの “SWITCHED ON” を確認
ステータスワード (6041h) の読み込み
CAN コマンド 40 41 60 00 00 00 00 00
- “OPERATION ENABLED” ステートの設定
コントロールワード (6040h) の書込み → 0x0F
CAN コマンド 2B 40 60 00 0F 00 00 00
- ブレーキの解放音を確認
- LED ステータスで有効 (ENABLE) を確認
- ドライバの “ENABLED” を確認
ステータスワード (6041h) の読み込み
CAN コマンド 40 41 60 00 00 00 00 00
- 速度指令値の設定
目標速度 (60FFh) の書込み → 0x960000 (例: 600r/min を設定)
初期設定の単位は inc/s です。モータ軸で 1 回転あたり 16384inc/s となります。
16384 [inc/s] -> 983040 [inc/min]=[rpm] -> 0x960000 [inc/min]=[rpm]
CAN コマンド: 23 FF 60 00 00 00 96 00
- モータの回転を確認
- 加速完了後のモータ速度を確認
速度検出値 (606Ch) の読み込み
CAN コマンド 40 6C 60 00 00 00 00 00
(モータ速度は、2053h [単位: rpm/4] でもモニタ可能です。)

■ “Operation Enabled” ステートからの遷移

“Operation Enabled” ステートから他のステートへ遷移したことを確認できるオブジェクトが実装されています。

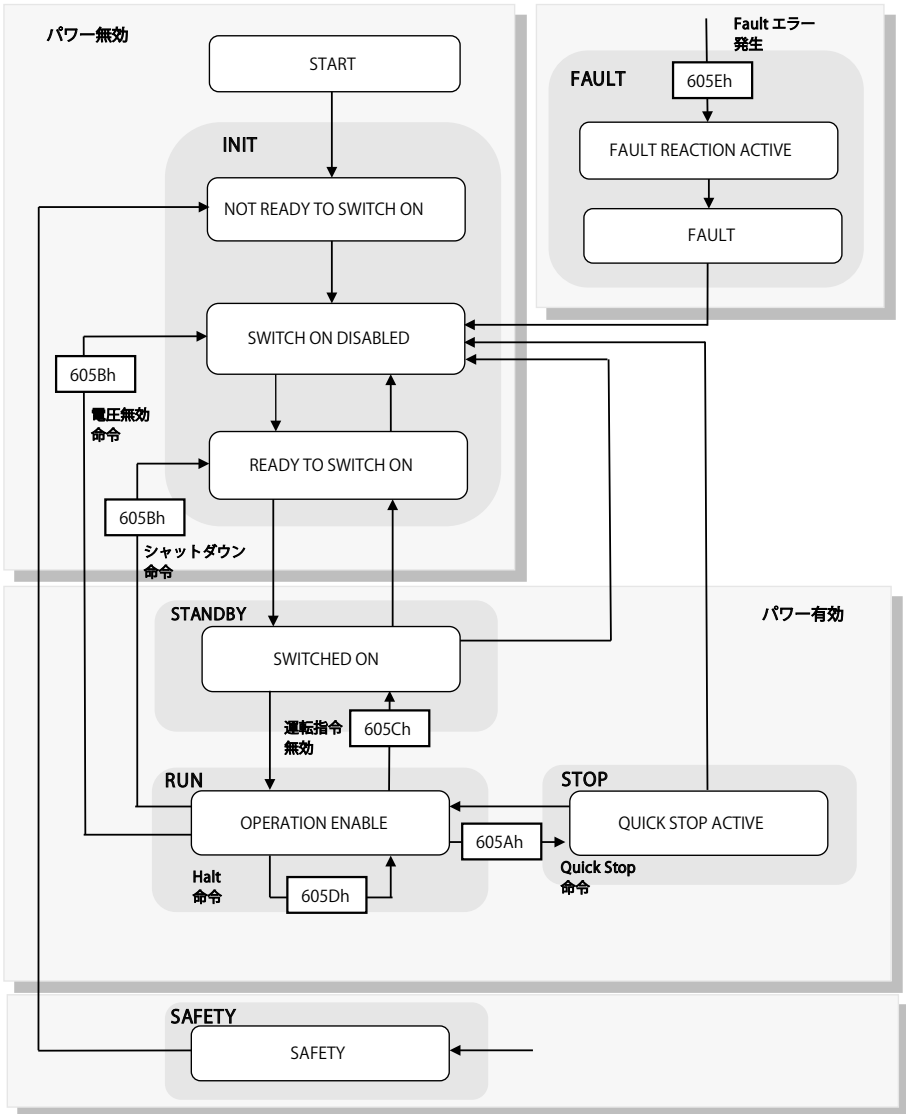


図 2-20 “Operation Enabled” ステートからの遷移

[2] CANopen 運転

以下にオブジェクトを示します。

- 605Ah : Quick Stop オプションコード
- 605Bh : Shutdown オプションコード
- 605Ch : Disable Operation オプションコード
- 605Dh : Halt オプションコード
- 605Eh : Fault Reaction オプションコード

• “Quick Stop Active” (605Ah) への遷移

“Operation Enabled” (RUN) から “Quick Stop Active” (STOP) へ遷移するためには、コントロールワード (6040h) のビット 2「急停止」を ON にすることが必要です。

ドライバは、最大トルクで即座に停止 (ランプ減速なし) します。

この遷移は、オブジェクト (605Ah) 「急停止オプションコード」に示されています。(リード専用)

• “Ready to Switch ON” または “Switch ON Disabled” (605Bh) への遷移

“Operation Enabled” (RUN) から “Read to Switch ON” (INIT) へ移行するためには、コントロールワード (6040h) のビット 0「スイッチ ON」を OFF にすることが必要です。

ドライバは、ダイナミックブレーキ設定によって減速します。(3007h, Sub-Index 1)

“Operation Enabled” (RUN) から “Switch ON Disabled” (INIT) へ移行するためには、コントロールワード命令 (6040h) のビット 1「電圧有効」を OFF にすることが必要です。

ドライバは、ダイナミックブレーキ設定によって減速します。(3007h, Sub-Index 1)

この遷移は、オブジェクト (605Bh) 「シャットダウンオプションコード」に示されています。(リード専用)

• “Switched ON” (605Ch) への遷移

“Operation Enabled” (RUN) から “Switched ON” (STANDBY) へ遷移する際、ドライバは、ダイナミックブレーキによりモータをランプ減速させます。

ダイナミックブレーキ使用時の減速時間は、「ダイナミックブレーキパラメータ」(3007h: 4) で設定することができます。

ダイナミックブレーキでのランプ減速を使用する場合は、あらかじめ「ダイナミックブレーキオプション」(3007h: 1) の設定を有効 (= 1) としている必要があります。初期設定が有効です。

この遷移は、オブジェクト (605Ch) 「運転無効オプションコード」に示されています。(リード専用)

ダイナミックブレーキが有効で “Operation Enabled” から他のステートへ遷移する方法は以下 2 つです。

(1) 「緊急停止」の使用 (DIG-IN3 を ON)

24V のデジタルインプット端子で動作させます。24V から 0V (または 0V から 24V) で、ドライバは、ダイナミックブレーキのランプ減速が実行され、停止後は、“Switched ON” (STANDBY) になります。

(2) コントロールワードを使用

“Operation Enabled” (RUN) から “Switched ON” (STANDBY) へ移行するためには、コントロールワード (6040h) のビット 3「運転無効」を OFF にすることが必要です。

ドライバは、ダイナミックブレーキでモータをランプ減速させ、停止後に “Switched ON” (STANDBY) へ遷移します。

「ダイナミックブレーキオプション」(3007h: 1) の設定を無効 (= 0) としていた場合、モータは、フリーラン停止し、即座に “Operation Enabled” (RUN) から “Switched ON” (STANDBY) へ遷移します。

• “Operation Enabled” (605Dh) を維持

“Operation Enabled” (RUN) に留まる場合、モータ運転中にコントロールワード (6040h) のビット 8「Halt」を ON することで、“Operation Enabled” を維持したまま、モータは即座に停止 (ランプ減速なし) します。

モータは、速度 0 を維持するためにトルクを発生します（サーボ ON 状態）。

この状態は、次の運転モードで使用されます。

- ・速度プロファイル
- ・トルクプロファイル

マスタコントローラがコントロールワードのビット 8 (Halt ビット命令) を変更した場合

- ・0 → 1 速度(トルク)発生器は、出力を 0 までランプ状に制御
- ・1 → 0 速度(トルク)発生器は、出力を目標値までランプ状に制御

速度プロファイルの場合、速度発生器は、速度プロファイルと加速プロファイルを参照します。
トルクプロファイルの場合、トルク発生器は、トルクスロープとトルクプロファイルを参照します。

詳細は、速度プロファイルとトルクプロファイルの Halt ビット命令(コントロールワードのビット 8)の内容を参照してください。

この遷移は、「Halt オプションコード」(605Dh)に示されています。(リード専用)

モータを停止させる場合でも“Operation Enabled” (RUN) を保持するためには、目標速度を 0rpm に設定する必要があります。

この場合、ドライバは、「減速プロファイル」(6084h)に設定された減速ランプの値で減速を実行します。

モータが 0 rpm で停止した後も“Operation Enabled” (RUN) ステートは維持されます。

・“Fault Reaction Active” (605Eh) への遷移

ダイナミックブレーキが有効に設定されている状態で異常を検出した場合、ドライバは、適切なランプ減速(減速ランプが許容されている場合)でモータを停止させ、“Operation Enabled” (RUN) を含む全てのステートから“Fault Reaction Active”へ遷移します。

ダイナミックブレーキが無効に設定されている場合、即座に停止(ランプ制御なし)して、“Fault Reaction Active”へ移行します。

次の異常を検出した場合、ダイナミックブレーキが有効の設定でも、即座にモータの回転を停止させます。

- ・過電流
- ・過電圧
- ・過速度
- ・回転検出器のフィードバックエラー

この遷移は、オブジェクト(605Eh)「故障処理オプションコード」に示されています。(読み出し専用)

・「緊急停止入力」(DIG-IN3)が入力された場合

モータがトルクを発生しているドライバのステートは、“Operation Enabled” (RUN) または“Quick Stop Active” (STOP) です。

デジタル入力端子(DIG-IN3)に「緊急停止入力：有効」が設定されていて、ドライバがその入力を受け付けると「ダイナミックブレーキパラメータ」(3007h)の設定に従って、モータを停止させます。

停止後、“Switched ON” (STANDBY)へ移行します。

初期設定では、DIG-IN3：緊急停止機能が有効に設定されています。

・STO 設定：セーフティステートへの遷移

“Operation Enabled” (RUN) を含む全ステートから、“Safety” への遷移が可能です。

STO がトリガ入力された場合、ドライバは、スイッチオフ状態となり、モータをフリーラン停止させます。

詳細は、「4 章 セーフティ」を参照してください。

[3] 測定単位の変換

ドライバの内部単位は、初期設定でインクリメント [inc] に設定されています。
この内部単位を [rpm] に変更する場合は、ドライバのパラメータ設定で変更が可能です。

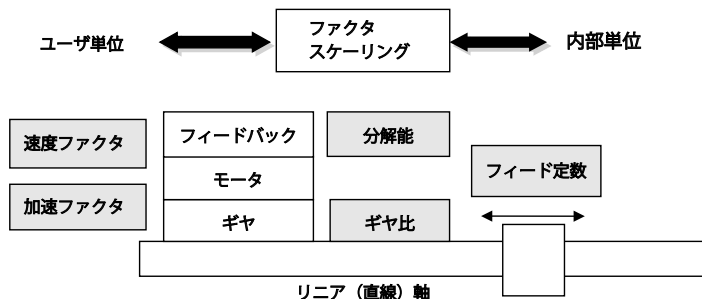


図 3-1 ファクタグループ

ドライバに設定された内部単位を参照するオブジェクト（ファクタグループ）は、以下の通りです。

- ・ 60FFh：目標速度
- ・ 606Dh：速度ウィンドウ
- ・ 606Fh：速度しきい値
- ・ 6081h：速度プロファイル(位置プロファイルで仕様-> 未実装)
- ・ 6082h：速度エンド(位置プロファイルで使用-> 未実装)

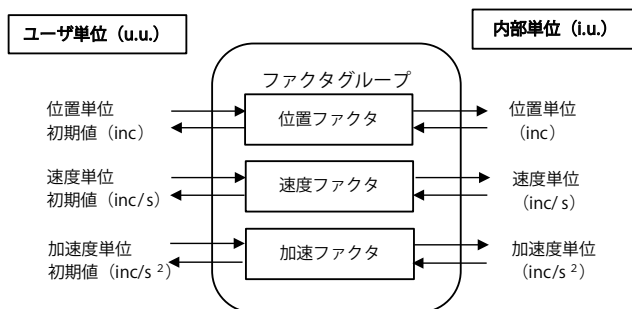


図 3-2 ファクタグループ単位

全てのパラメータ設定値は、内部単位で保存されていて、ファクタグループに設定された値を使用し、ユーザー単位（例：rpm）に変換することができます。

初期設定値を次の表に示します。

表 3-1 ユーザ単位の初期値

オブジェクト	名 称	内部単位	ユーザー単位の初期値
距 離	位置単位	Inc	Inc
速 度	速度単位	Inc/s	Inc/s
加速度	加速度単位	Inc/s ²	Inc/s ²

ファクタグループとして定義されたオブジェクトは、デバイスの内部単位（インクリメント）と物理単位の関係を設定します。ユーザー単位は [u.u.]、内部単位は [i.u.] として表記・定義しています。
注）詳細は、11 章 各種の設定「ユーザー単位の変換方法」を参照してください。

【3】測定単位の変換

⚠ 注意

初期設定時の単位は、インクリメント [inc] です。

単位を変更した場合、いくつかのオブジェクトも強制的に変更されます。

(例：加速 / 減速プロファイル、最大加速 / 最大減速プロファイルなど)

3-1 単位変換パラメータ

次の表の Index 設定値を参照して、ドライバが処理を実行します。

表 3-2 Index 設定値

Index	名 称	オブジェクトコード	データタイプ	属 性	備 考
608Fh	エンコーダ分解能 (位置)	ARRAY	UNSIGNED32	rw	使用不可
6090h	エンコーダ分解能 (速度)	ARRAY	UNSIGNED32	rw	
6091h	ギヤ比	ARRAY	UNSIGNED32	rw	
6092h	フィード定数	ARRAY	UNSIGNED32	rw	
6096h	速度ファクタ	ARRAY	UNSIGNED32	rw	使用可
6097h	加速ファクタ	ARRAY	UNSIGNED32	rw	

■オブジェクト 6096h：速度ファクタ (Velocity factor)

このオブジェクトは、速度の単位をユーザによって定義した単位に合わせて設定できます。

オブジェクト内容：

Index	EDS 名称	オブジェクトタイプ	データタイプ	カテゴリ
6096h	Velocity Factor	ARRAY	UNSIGNED32	強制

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	数値範囲	初期値
00h	サポートされた最上位のサブインデックス	ro	なし	2	2
01h	分子	rw	なし	1 ~ 2147483647	1
02h	分母	rw	なし	1 ~ 2147483647	1

速度ファクタの分子と分母をそれぞれ設定してください。

$$\text{速度ファクタ} = \frac{\text{分子 (6096h Sub-Index 01)}}{\text{分母 (6096h Sub-Index 02)}}$$

内部単位は、次式によりユーザで定義した単位 [u.u.] で計算されます。

$$\text{速度 [i.u.]} = \text{速度 [u.u.]} \times \left(\frac{\text{分子 (6096h Sub-Index 01)}}{\text{分母 (6096h Sub-Index 02)}} \right)$$

初期設定は、分子と分母に 1 が設定されています。

異なる単位を定義する場合は、以下の例を参考に分子と分母の値を変更してください。

[3] 測定単位の変換

例 1：単位 rpm を使用する場合

- ・ドライバ内部の単位 (固定) : [inc/s]
- ・ユーザが使用する単位 : [rpm] = [rotation/min]

$$\text{速度 [inc/s]} = \text{速度 [rotation/min]} \times \left(\frac{16384 \text{ [inc/rotation]}}{60 \text{ [s]}} \right)$$

分子 = 回転検出器 (レゾルバ、アブソリュートエンコーダ) の分解能 : 16384 [inc/rotation]、
分母 = 60 [s]

例 2：単位 drpm を使用する場合

- ・ドライバ内部の単位 (固定) : [inc/s]
- ・ユーザが使用する単位 : [drpm/s] = [inc/rotation] の 10 倍

$$\text{速度 [inc/s]} = \text{速度 [drpm]} \times \left(\frac{10 \times 16384 \text{ [inc/rotation]}}{60 \text{ [s]}} \right)$$

分子 = 10 × 回転検出器 (レゾルバ、アブソリュートエンコーダ) の分解能 : 16384 [inc/rotation]、
分母 = 60 [s]

例 3：単位 rph を使用する場合

- ・ドライバ内部の単位 (固定) : [inc/s]
- ・ユーザが使用する単位 : [rph] = [rotation/hour]

$$\text{速度 [inc/s]} = \text{速度 [rph]} \times \left(\frac{16384 \text{ [inc/rotation]}}{3600 \text{ [s]}} \right)$$

分子 = 回転検出器 (レゾルバ、アブソリュートエンコーダ) の分解能 : 16384 [inc/rotation]、
分母 = 3600 [s]

例 4：単位 m/s を使用する場合

- ・ドライバ内部の単位 (固定) : [inc/s]
 - ・ユーザが使用する単位 : [m/s]
- モータ 1 回転が 0.3m に対応している場合

$$\text{速度 [inc/s]} = \text{速度 [m/s]} \times \left(\frac{16384 \text{ [inc/rotation]}}{0.3 \text{ [m/rotation]}} \right)$$

分子 = 回転検出器 (レゾルバ、アブソリュートエンコーダ) の分解能 : 16384 [inc/rotation]、
分母 = 0.3 [m/rotation]

注) 分子と分母のデータタイプの範囲は整数で [-214783648 ~ +2147483647] のため、小数点以下を表すことができません。

ファクタグループは、次のオブジェクトが使用されます。

- ・60FFh : 目標速度 (Target Velocity)
- ・606Dh : 速度ウィンドウ (Velocity Window)
- ・606Fh : 速度スレッシュホールド (Velocity Threshold)
- ・6081h : 速度プロファイル (Profile Velocity) → 未実装
- ・6082h : 速度エンド (End Velocity) → 未実装

【3】測定単位の変換

⚠ 注意

設定値に誤りがある場合、ドライバはアボートコードを送信します。
初期化処理中に正常な値を検出すると、ドライバはエマージェンシプロトコルでエラーメッセージを送信します。
エラーコード(0x8B19)を参照してください。

ドライバは、次のアボートコードを送信することがあります。

- 0x06040030 = 範囲外の数値

■オブジェクト 6097h: 加速ファクタ (Accelerator factor)

このオブジェクトは、加速度の単位をユーザによって定義した単位に合わせて設定できます。
オブジェクト内容：

Index	EDS 名称	オブジェクトタイプ	データタイプ	カテゴリ
6097h	Acceleration Factor	ARRAY	UNSIGNED32	強制

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	数値範囲	初期値
00h	サポートされた最上位のサブインデックス	ro	なし	2	2
01h	分子	rw	なし	1 ~ 2147483647	1
02h	分母	rw	なし	1 ~ 2147483647	1

加速ファクタの分子と分母は、それぞれ設定してください。

$$\text{加速ファクタ} = \frac{\text{分子 (6097h Sub-Index 01)}}{\text{分母 (6097h Sub-Index 02)}}$$

内部単位は、次式によりユーザで定義した単位 [u.u.] で計算されます。

$$\text{加速度 [i. u.]} = \text{加速度 [u. u.]} \times \left(\frac{\text{分子 (6097h Sub-Index 01)}}{\text{分母 (6097h Sub-Index 02)}} \right)$$

初期設定は、分子と分母に 1 が設定されています。

異なる単位を定義する場合は、以下の例を参考に分子と分母の値を変更してください。

例 1：単位 rpm/s を使用する場合

- ドライバ内部の単位(固定)：[inc/s²]
- ユーザが使用する単位：[rpm/s]

$$\text{加速度 [inc/s}^2\text{]} = \text{加速度 [rpm/s]} \times \left(\frac{16384 [\text{inc/rotation}]}{60 [\text{s}]} \right)$$

分子 = 回転検出器(レゾルバ、アブソリュートエンコーダ)の分解能:16384 [inc/rotation]、
分母 = 60 [s]

【3】測定単位の変換

例 2：単位 rps/s^2 を使用する場合

- ドライバ内部の単位 (固定) : $[\text{inc/s}^2]$
- ユーザが使用する単位 : $[\text{rps/s}^2]$

$$\text{加速度 } [\text{inc/s}^2] = \text{加速度 } [\text{rps/s}] \times \left(\frac{16384 [\text{inc/rotation}]}{1 [\text{s}]} \right)$$

分子 = 回転検出器 (レゾルバ、アブソリュートエンコーダ) の分解能 : $16384 [\text{inc/rotation}]$ 、
分母 = $1 [\text{s}]$

ファクタグループは、次のオブジェクトが使用されます。

- 6083h : 加速プロファイル
- 6084h : 減速プロファイル
- 60C5h : 最大加速
- 60C6h : 最大減速 :

⚠ 注意

設定値に誤りがある場合、ドライバは、アボートコードを送信します。
初期化中に正しい値でなければ、ドライバは、エマージェンシプロトコルでエラーメッセージを送信します。
エラーコード (0x8B1A) を参照してください。

ドライバは、次のアボートコードを送信することがあります。

- 0x06040030 = 範囲外の数値

【4】セーフティ

全てのステートからセーフティ (SAFETY)ステートに移行することができます。
セーフティステートから抜けるためには、「コントロールワード」(6040h)へ「Disable Voltage」のビットを送信する必要があります。

4-1 セーフティオブジェクト

「セーフティステート」(4000h)は、ドライバがセーフティ状態なのか、またはセーフティステートの状態を通知します。

■オブジェクト 4000h：セーフティステート

このオブジェクトは、ドライバのセーフティステートを通知します。読み出し専用です。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
4000h	セーフティステート	ARRAY	UNSIGNED16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	値の範囲	初期値
0	エントリ数	ro	なし	-	2
1	セーフティステート	ro	なし	0、1	-
2	STO 機能	ro	なし	0、1	-

数値の定義：

Sub-Index	フィールド	設 定	定 義
1	セーフティステート	0b	ドライバがセーフティステートでない。
		1b	ドライバがセーフティステートを検出
2	STO 機能	0b	STO セーフティステートが発生していない。
		1b	STO セーフティステートが発生

注)詳細は、8 章 機能の「8-3 他の機能 セーフティ状態の確認」を参照してください。

【4】セーフティ

4-2 セーフティステートの状態機械 DSP402

下図は、セーフティステートを示しています。このステートは、状態機械 (DSP402) に追加されています。

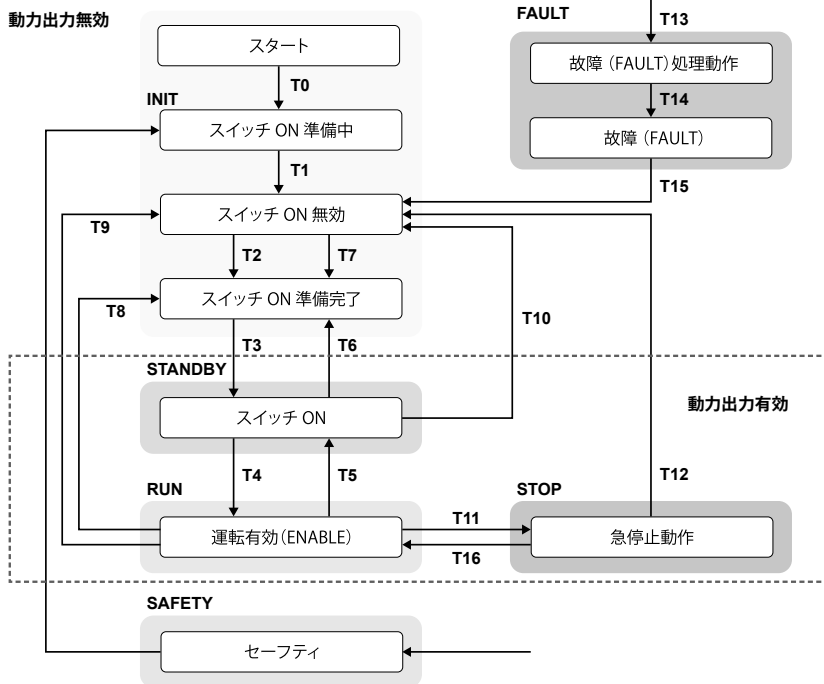


図 4-1 セーフティステートの状態機械 DSP402

■ STO の特長

STO は、2 チャンネルの構造です。この構造を以下のブロック図に示します。

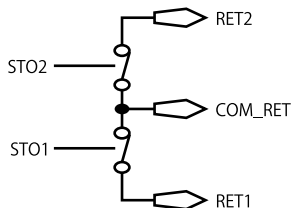


図 4-2 STO 出力フィードバックリレー

STO 機能付きドライバの場合、モータを運転させるためには、2 つの絶縁された STO デジタル入力端子に DC24V を入力する必要があります。
STO デジタル入力端子の状態は、「デジタル入力」(60FDh) に書込まれます。

表 4-1 入出力リレーの動作

入力 1	入力 2	出力 1	出力 2	出力状態
STO1	STO2	RET1	RET2	ステータス
0	0	閉	閉	セーフティ
24V	0	開	閉	
0	24V	閉	開	
24V	24V	開	開	ノーマルモード

セーフティステートでは、モータにトルクが発生しません。STO 機能が動作することで、ドライバはセーフティステートへ移行し、モータのトルクが発生せずに安全な状態を維持します。

モータが駆動中に STO 機能を動作させた場合、モータはフリーランになります。

セーフティステートでは、モータのトルクが発生しないため、システムの不慮の再起動や誤差動に対して、安全性を高めることが可能です。

注意

ドライバは、STO 機能が動作するとトルクが発生しないため、負荷を保持することができません。

- モータ運転中に STO 機能が動作した場合、ドライバはモータを制御せず、フリーランで停止します。
- ドライバにセーフティトルク OFF (STO) 機能がある場合、その他の機能を動作させる前に、この回路が正常に動作することを確認してください。

【5】 CANopen オブジェクトディクショナリ

5-1 ゼネラルオブジェクト (DS301)

■オブジェクト 1000h：デバイスタイプ (Device Type)

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1000h	Device Type	VAR	U32	0

MSB ビット 31	LSB ビット 0
追加情報	デバイスプロファイル番号

初期値: 0xFF7A0192 (0192 は、デバイスがプロファイル 402 を使用していることを意味します。)

■オブジェクト 1001h：エラーレジスタ (Error Register)

エラーレジスタは、それぞれ特定のエラータイプに対応する 8 ビットのフィールドです。エラーが発生すると、以下に示すビットがセットされます。

ビット	内 容
0	一般のエラー (Generic Error)
1	電流
2	電圧
3	温度
4	通信エラー (オーバーラン、エラーステート)
5	デバイスプロファイル仕様
6	予約領域
7	メーカー固有

■オブジェクト 1002h：メーカー固有ステータスレジスタ (Manufacturer Status Register)

このオブジェクトは、メーカー固有の共通のステータスレジスタを示しています。オブジェクトのサイズと位置だけが定義されます。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1002h	Manufacturer Status Register	VAR	U32	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	初期値
00h	Manufacturer Status Register	ro	なし	-

■オブジェクト 1003h：予め定義されたエラーフィールド (Pre-defined Error Field)

このオブジェクトは、エマージェンシオブジェクトを介して通知されたエラーがドライバに保持され、異常履歴を確認することができます。最大 8 回分の異常を保持します。Sub-index: 0 に 0 を書き込むことで、異常履歴が全て削除されます。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1003h	Pre-defined Error Field	VAR	U32	強制

【5】CANopen オブジェクトディクショナリ

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	初期値
00h	エラーを検出した数	rw	なし	-
01h	最新のエラーコード	ro	なし	
02h	最新の1つ前のエラーコード	ro	なし	
03h ~ FFh	過去のエラーコード	ro	なし	

新しいエラーが発生した場合、Sub-index 1 に保存されます。Sub-index へ既に入力されている 1 ~ 15 のエラー履歴は、Sub-index 2 以降へ移行します。その際に、Sub-index 15 に保存されていたエラーが削除されます。また、過去に発生したエラーの数は、Sub-index 0 を読出すことで確認できます。

異常履歴が無いまたは削除した場合、Sub-index 1 ~ 15 は読出すことができず、アボートメッセージ(アボートコード：0800 0024h)を送信します。

エラーフィールドは、以下の構造で定義されています。

MSB ビット				LSB ビット	
31	24	23	16	15	0
メーカ固有エラーコード		エラーレジスタ		エラーコード	

Sub-index: 0 に 00h を書込んだ場合、Sub-index 1 ~ 15 の全ての異常履歴が消去されます。00h 以外の値が許可されず、アボートメッセージ(エラーコード：0609 0030h)が送信されます。

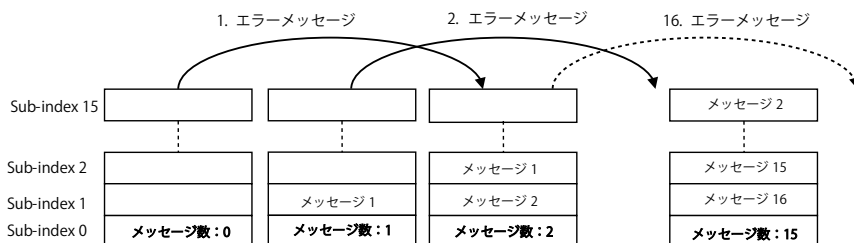


図 5-1 メッセージ履歴リスト

注) 詳細は、8 章 機能の「8-3 他の機能 異常履歴の確認」を参照してください。

■オブジェクト 1005h：COB-ID SYNC

COB-ID は、SYNC オブジェクトです。ビット 30 が設定されていれば SYNC メッセージを発生します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1005h	COB-ID SYNC	VAR	UNSIGNED32	強制

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	初期値
00h	COB-ID SYNC	rw	なし	80h

【5】 CANopen オブジェクトディクショナリ

■オブジェクト 1008h：メーカー固有デバイス名 (Manufacturer Device Name)

このオブジェクトは、デバイス名を示しています。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1008h	Manufacturer Device Name	VAR	文字列(4 文字)	強制

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	初期値
00h	Manufacturer Device Name	cost	なし	-

例：smartris のデバイス名は、「SMA」の文字列です。

■オブジェクト 1009h：メーカー固有ハードウェアバージョン (Manufacturer Hardware Version)

このオブジェクトは、デバイスのハードウェアのバージョンを示しています。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1009h	Manufacturer Hardware Name	VAR	文字列(4 文字)	強制

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	初期値
00h	Manufacturer Hardware Name	cost	なし	-

例：ハードウェアバージョン 1.1 は、「11」の文字列です。

■オブジェクト 100Ah：ソフトウェアバージョン (Manufacturer Software Version)

このオブジェクトは、デバイスのソフトウェア(ファームウェア)バージョンを示しています。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
100Ah	Manufacturer Software Name	VAR	文字列(4 文字)	強制

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	初期値
00h	Manufacturer Software Name	cost	なし	-

例：ファームウェアバージョン 1.0.8 は、「108」の文字列です。

■オブジェクト 100Ch：ガードタイム (Guard Time)

ガードタイムは、ms 単位で入力します。0 を設定するとノードガーディング機能をオフにします。本オブジェクトに、「ライフタイムファクタ」(100Dh) の設定値を乗算することで、ライフガーディングプロトコルのライフタイムが計算できます。

【5】CANopen オブジェクトディクショナリ

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
100Ch	Guard Time	VAR	U16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	Guard Time	rw	なし	0 ～ 65535	0

⚠ 注意

- ノードゲーディングよりハートビートプロトコルの方が高い優先度で処理されます。
- ノードゲーディングとハートビートプロトコルが同時に動作すると、ノードゲーディングタイマが停止します。
この場合、EMCY メッセージは送信されません。

■オブジェクト 100Dh：ライフタイムファクタ (Life Time Factor)

「ガードタイムファクタ」(100Ch)にライフタイムファクタを乗算すると、ノードゲーディングのライフタイムが計算できます。0 は、ノードゲーディングをオフに切替えます。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
100Ah	Life Time Factor	VAR	U8	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	Life Time Factor	rw	なし	0 ～ 255	0

■オブジェクト 1010h：パラメータフィールドストア (Store Parameters Field)

このオブジェクトは、EEPROM へパラメータを保存するときの設定用オブジェクトです。サブインデックスは、次のように分類されています。

- Sub-index 0：サポートされている全てのサブインデックスを表します。
- Sub-index 1：「デバイスにストアできる全てのパラメータ」を参照
- Sub-index 2：「通信関係のパラメータ」を参照
(Index 1000h ～ 1FFFh メーカー固有の通信パラメータ)
- Sub-index 3：「アプリケーション関係のパラメータ」を参照
(Index 6000h ～ 9FFFh メーカー固有のアプリケーションパラメータ)
- Sub-index 4：「メーカー固有のパラメータ」を参照
- Sub-Index 5：予約領域

デバイスの状態機械が “Operation Enabled” または “Quick Stop” の場合、この指令を実行できません。

もし、これらの指令が実行できなかった場合、ワーニングメッセージを送信し、「ワーニングオブジェクト」(2003h)の3番目のビットを1に設定します。

【5】 CANopen オブジェクトディクショナリ

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1010h	Store Parameter Field	ARRAY	U32	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲
00h	エントリ数	c	なし	5
01h	全パラメータのセーブ	rw	なし	0
02h	通信パラメータのセーブ	rw	なし	0
03h	アプリケーションパラメータのセーブ	rw	なし	0
04h	メーカ固有パラメータのセーブ	rw	なし	0
05h	予約領域	rw	なし	0

誤ってパラメータを保存することを避けるため “save” (0x65766173) が、適切な Sub-index に書込まれた場合のみ、保存が実行されます。

“save” : 0x65766173

Signature ISO 8859 ("ASCII")	e	v	a	s
Hex	65h	76h	61h	73h

CANopen デバイスは、適切な Sub-index に正しい数値を受信すると、パラメータをストアします。その後、正常にストアが完了したことを知らせる SDO が送信されます。

もし、ストアに失敗すると、CANopen デバイスは、SDO でアボートコード：0606 0000h を送信します。

誤った数値を受信した場合、CANopen デバイスは、ストアせずに SDO でアボートコード：0800 002xh を送信します。

各 Sub-index の読出し時は、以下の情報を取得できます。

MSB ビット		LSB ビット	
31	2	1	0
予約領域		Auto	CMD

リードアクセスの構造：

ビット	フィールド	コンフィグレーション	内 容
0	Cmd	0b 1b	CANopen デバイスは、上位コントローラからの指令によってパラメータをセーブしません。 CANopen デバイスは、上位コントローラからの指令によってパラメータをセーブします。
1	Auto	0b 1b	CANopen デバイスは、自動でパラメータをセーブしません。 CANopen デバイスは、自動でパラメータをセーブします。

【5】CANopen オブジェクトディクショナリ

■オブジェクト 1011h：初期パラメータのリストア (Restore Default Parameters)

このオブジェクトは、初期パラメータのリストアを実行します。

各サブインデックスを读出することで、デバイスがリストアに関する情報を送信します。

パラメータグループは、次のように分類されます。

- Sub-index 0：最大のサブインデックスを表します。
- Sub-index 1：全てのオブジェクトを出荷時設定にリストア
- Sub-index 2：全ての通信関係のオブジェクトを出荷時設定にリストア (0x0000 ～ 0x1FFF)
- Sub-index 3：全てのアプリケーション関係のオブジェクトを出荷時設定にリストア (0x2000 ～)
- Sub-index 4：メーカーが固有に定義したオブジェクトを出荷時設定にリストア
- Sub-index 5：予約領域

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1011h	Store Parameter Field	ARRAY	U32	オプション

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲
00h	エン트리数	c	なし	5
01h	全パラメータのリストア	rw	なし	0
02h	通信パラメータのリストア	rw	なし	0
03h	アプリケーションパラメータのリストア	rw	なし	0
04h	メーカー固有パラメータのリストア	rw	なし	0
05h	予約領域	rw	なし	0

オブジェクト「初期パラメータのリストア」は、出荷時設定のパラメータをロードします。

“load” (0x64616F6C) が書込まれた場合のみリストア処理を実行します。

“load”：0x64616F6C

Signature ISO 8859 ("ASCII")	d	a	o	l
Hex	64h	61h	6Fh	6Ch

CANopen デバイスは、適切な Sub-index に正しい数値を受信すると、初期パラメータにリストアします。

その後、正常にリストアが完了したことを知らせる SDO が送信されます。

もしリストアに失敗すると、CANopen デバイスは、SDO でアボートコード：0606 0000h を送信します。

誤った数値を受信した場合、CANopen デバイスは、初期パラメータをリストアせずに SDO でアボートコード：0800 002xh を送信します。

初期パラメータは、CANopen デバイスがリセット (NMT サービスノードリセット：Sub-index 01h ～ 7Fh, NMT サービス通信リセット：Sub-index 02h) されるか、電源の再起動後に設定されます。

【5】 CANopen オブジェクトディクショナリ

各 Sub-index の読み出し時は、以下の情報を取得できます。

MSB ビット		LSB ビット	
31	1		0
予約領域		CMD	

リードアクセス構造：

ビット	フィールド	コンフィグレーション	定 義
0	Cmd	0b 1b	CANopen デバイスは、初期パラメータをリストアしません。 CANopen デバイスは、初期パラメータをリストアします。

■オブジェクト 1014h：COB-ID エマージェンシメッセージ(Emergency Message)

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1010h	COB-ID EMCY	VAR	U32	オプション

エントリ内容：

Sub-Index	アクセス	PDO マッピング	初期値
00h	ro	no	0x80+ ノード ID

■オブジェクト 1017h：プロデューサハートビートタイム(Producer Heartbeat Time)

このオブジェクトは、ハートビートメッセージを送信する周期を設定します。単位は、ms です。
0 に設定した場合、ハートビート機能を無効にします。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1017h	Producer Heartbeat Time	VAR	U16	強制

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	Producer Heartbeat Time	rw	なし	0 ～ 65535	200

ハートビートプロデューサの NMT ステートのデータバイト

- 0 (00h) : “Boot-Up”
- 4 (04h) : “Stopped”
- 5 (05h) : “Operational”
- 127 (7Fh) : “Pre-operational”

【5】 CANopen オブジェクトディクショナリ

■オブジェクト 1018h：識別オブジェクト (Identity Object)

このオブジェクトは、CANopen デバイスの一般的な識別情報を表します。

- **Sub-index 01h**：CANopen デバイスの個々のベンダに割り当てられた番号を含みます。
0000 0000h は、無効のベンダ ID を示しています。
- **Sub-index 02h**：CANopen デバイスの特殊タイプを識別した番号を含みます。
0000 0000h は、予約領域です。
- **Sub-index 03h**：CANopen デバイスのモデルチェンジとマイナチェンジの改訂番号を含みます。
改訂番号は、CANopen デバイスの履歴を識別できます。
CANopen の機能が異なる場合、改訂番号が増加します。
増加した改訂番号は、同じような履歴の CANopen のデバイスとのバージョンの違いを識別します。
0000 0000h は、予約領域です。
- **Sub-index 04h**：CANopen デバイスでの生産ロットと仕様改訂で識別されるシリアル番号を含みます。0000 0000h は、予約領域です。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1018h	Identity Object	RECORD	Identity	強制

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	初期値
00h	サポートされる最大の Sub-index	ro	なし	4
01h	ベンダ (Vendor) ID	ro	なし	000001FBh
02h	製品コード (Product Code)	ro	なし	予約領域
03h	改訂番号 (Revision Number)	ro	なし	
04h	シリアル番号 (Serial Number)	ro	なし	

■オブジェクト 1400h：PDO1 通信パラメータ受信 (Receive PDO1 Communication Parameter)

デバイスが受信可能な、PDO1 の通信パラメータを表します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1400h	Receive PDO Communication Parameter	PDO_COMM_PAR	RECORD	強制

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データタイプ	初期値
00h	エン트리数	ro	なし	UNSIGNED8	5
01h	COB-ID	ro	なし	UNSIGNED32	0x201
02h	送信タイプ	ro	なし	UNSIGNED8	0xFE
03h	禁止時間 (Inhibit Time)	ro	なし	UNSIGNED16	0x00
04h	互換性 (Compatibility) エン트리	ro	なし	UNSIGNED8	(予約領域)
05h	イベントタイマ	ro	なし	UNSIGNED16	0x00

【5】 CANopen オブジェクトディクショナリ

■オブジェクト 1401h : PDO2 通信パラメータ受信 (Receive PDO2 Communication Parameter)

デバイスが受信可能な、PDO2 の通信パラメータを表します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1401h	Receive PDO Communication Parameter	PDO_COMM_PAR	RECORD	強制

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データタイプ	初期値
00h	エントリ数	ro	なし	UNSIGNED8	5
01h	COB-ID	ro	なし	UNSIGNED32	0x301
02h	送信タイプ	ro	なし	UNSIGNED8	0xFE
03h	禁止時間 (Inhibit Time)	ro	なし	UNSIGNED16	0x00
04h	互換性 (Compatibility) エントリ	ro	なし	UNSIGNED8	(予約領域)
05h	イベントタイマ	ro	なし	UNSIGNED16	0x00

■オブジェクト 1402h : PDO3 通信パラメータ受信 (Receive PDO3 Communication Parameter)

デバイスが受信可能な、PDO3 の通信パラメータを表します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1402h	Receive PDO Communication Parameter	PDO_COMM_PAR	RECORD	強制

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データタイプ	初期値
00h	エントリ数	ro	なし	UNSIGNED8	5
01h	COB-ID	ro	なし	UNSIGNED32	0x401
02h	送信タイプ	ro	なし	UNSIGNED8	0xFE
03h	禁止時間 (Inhibit Time)	ro	なし	UNSIGNED16	0x00
04h	互換性 (Compatibility) エントリ	ro	なし	UNSIGNED8	(予約領域)
05h	イベントタイマ	ro	なし	UNSIGNED16	0x00

■Object 1403h : PDO4 通信パラメータ受信 (Receive PDO4 Communication Parameter)

デバイスが受信可能な、PDO4 の通信パラメータを表します。

Object Description:

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1403h	Receive PDO Communication Parameter	PDO_COMM_PAR	RECORD	強制

Entry Description:

Sub-Index	内 容	アクセス	PDO マッピング	データタイプ	初期値
00h	エントリ数	ro	なし	UNSIGNED8	5
01h	COB-ID	ro	なし	UNSIGNED32	0x501
02h	送信タイプ	ro	なし	UNSIGNED8	0xFE
03h	禁止時間 (Inhibit Time)	ro	なし	UNSIGNED16	0x00
04h	互換性 (Compatibility) エントリ	ro	なし	UNSIGNED8	(予約領域)
05h	イベントタイマ	ro	なし	UNSIGNED16	0x00

【5】 CANopen オブジェクトディクショナリ

■オブジェクト 1600h : PDO1 マッピングパラメータ受信 (Receive PDO1 Mapping Parameter)

Sub-index 1 以降には、マッピングする Index、Sub-index および Sub-index の長さを表します。

- ・設定値 : Index (16 ビット) + Sub-index (8 ビット) + 長さ (8 ビット)

オブジェクト内容 :

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1600h	Receive PDO1 Mapping Parameter	PDO_MAPPING	RECORD	強制

エン트리内容 :

Sub-Index	内 容	アクセス	PDO マッピング	データタイプ	初期値
00h	エントリ数	rw	なし	UNSIGNED8	3
01h	マッピングエントリ 1	rw	なし	UNSIGNED32	0x6040 0010
02h	マッピングエントリ 2	rw	なし	UNSIGNED32	0x6060 0008
03h	マッピングエントリ 3	rw	なし	UNSIGNED32	0x60FE 0120
04h	マッピングエントリ 4	rw	なし	UNSIGNED32	0
05h	マッピングエントリ 5	rw	なし	UNSIGNED32	0
06h	マッピングエントリ 6	rw	なし	UNSIGNED32	0
07h	マッピングエントリ 7	rw	なし	UNSIGNED32	0
08h	マッピングエントリ 8	rw	なし	UNSIGNED32	0

■Object 1601h : PDO2 マッピングパラメータ受信 (Receive PDO2 Mapping Parameter)

Sub-index 1 以降には、マッピングする Index、Sub-index および Sub-index の長さを表します。

- ・設定値 : Index (16 ビット) + Sub-index (8 ビット) + 長さ (8 ビット)

オブジェクト内容 :

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1601h	Receive PDO2 Mapping Parameter	PDO_MAPPING	RECORD	強制

エン트리内容 :

Sub-Index	内 容	アクセス	PDO マッピング	データタイプ	初期値
00h	エントリ数	rw	なし	UNSIGNED8	2
01h	マッピングエントリ 1	rw	なし	UNSIGNED32	0x6040 0010
02h	マッピングエントリ 2	rw	なし	UNSIGNED32	0x607A 0020
03h	マッピングエントリ 3	rw	なし	UNSIGNED32	0
04h	マッピングエントリ 4	rw	なし	UNSIGNED32	0
05h	マッピングエントリ 5	rw	なし	UNSIGNED32	0
06h	マッピングエントリ 6	rw	なし	UNSIGNED32	0
07h	マッピングエントリ 7	rw	なし	UNSIGNED32	0
08h	マッピングエントリ 8	rw	なし	UNSIGNED32	0

■オブジェクト 1602h : PDO3 マッピングパラメータ受信 (Receive PDO3 Mapping Parameter)

Sub-index 1 以降には、マッピングする Index、Sub-index および Sub-index の長さを表します。

- ・設定値 : Index (16 ビット) + Sub-index (8 ビット) + 長さ (8 ビット)

オブジェクト内容 :

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1602h	Receive PDO3 Mapping Parameter	PDO_MAPPING	RECORD	強制

【5】 CANopen オブジェクトディクショナリ

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データタイプ	初期値
00h	エン트리数	rw	なし	UNSIGNED8	2
01h	マッピングエン트리 1	rw	なし	UNSIGNED32	0x6040 0010
02h	マッピングエン트리 2	rw	なし	UNSIGNED32	0x60FF 0020
03h	マッピングエン트리 3	rw	なし	UNSIGNED32	0
04h	マッピングエン트리 4	rw	なし	UNSIGNED32	0
05h	マッピングエン트리 5	rw	なし	UNSIGNED32	0
06h	マッピングエン트리 6	rw	なし	UNSIGNED32	0
07h	マッピングエン트리 7	rw	なし	UNSIGNED32	0
08h	マッピングエン트리 8	rw	なし	UNSIGNED32	0

■オブジェクト 1603h：PDO4 マッピングパラメータ受信 (Receive PDO4 Mapping Parameter)

Sub-index 1 以降には、マッピングする Index、Sub-index および Sub-index の長さを表します。

・設定値：Index (16 ビット) + Sub-index (8 ビット) + 長さ (8 ビット)

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1603h	Receive PDO4 Mapping Parameter	PDO_MAPPING	RECORD	強制

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データタイプ	初期値
00h	エン트리数	rw	なし	UNSIGNED8	2
01h	マッピングエン트리 1	rw	なし	UNSIGNED32	0x6040 0010
02h	マッピングエン트리 2	rw	なし	UNSIGNED32	0x6071 0010
03h	マッピングエン트리 3	rw	なし	UNSIGNED32	0
04h	マッピングエン트리 4	rw	なし	UNSIGNED32	0
05h	マッピングエン트리 5	rw	なし	UNSIGNED32	0
06h	マッピングエン트리 6	rw	なし	UNSIGNED32	0
07h	マッピングエン트리 7	rw	なし	UNSIGNED32	0
08h	マッピングエン트리 8	rw	なし	UNSIGNED32	0

■Object 1800h：PDO1 通信パラメータ送信 (Transmit PDO1 Communication Parameter)

デバイスが受信可能な、PDO1 の通信パラメータを表します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1800h	Transmit PDO Communication Parameter	PDO_COMM_PAR	RECORD	強制

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データタイプ	初期値
00h	エン트리数	ro	なし	UNSIGNED8	5
01h	COB-ID	ro	なし	UNSIGNED32	0x181
02h	送信タイプ	ro	なし	UNSIGNED8	0xFF
03h	禁止時間 (Inhibit Time)	ro	なし	UNSIGNED16	0x05
04h	エン트리互換	ro	なし	UNSIGNED8	(予約領域)
05h	イベントタイマ	ro	なし	UNSIGNED16	0x00

【5】 CANopen オブジェクトディクショナリ

■オブジェクト 1801h：PDO2 通信パラメータ送信 (Transmit PDO2 Communication Parameter)

デバイスが受信可能な、PDO2 の通信パラメータを表します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1801h	Transmit PDO Communication Parameter	PDO_COMM_PAR	RECORD	Mandatory

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データタイプ	初期値
00h	エントリ数	ro	なし	UNSIGNED8	5
01h	COB-ID	ro	なし	UNSIGNED32	0x281
02h	送信タイプ	ro	なし	UNSIGNED8	0xFD
03h	禁止時間 (Inhibit Time)	ro	なし	UNSIGNED16	0x05
04h	エントリ互換	ro	なし	UNSIGNED8	(予約領域)
05h	イベントタイマ	ro	なし	UNSIGNED16	0x00

■オブジェクト 1802h：PDO3 通信パラメータ送信 (Transmit PDO3 Communication Parameter)

デバイスが受信可能な、PDO3 の通信パラメータを表します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1802h	Transmit PDO Communication Parameter	PDO_COMM_PAR	RECORD	強制

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データタイプ	初期値
00h	エントリ数	ro	なし	UNSIGNED8	5
01h	COB-ID	ro	なし	UNSIGNED32	0x381
02h	送信タイプ	ro	なし	UNSIGNED8	0xFD
03h	禁止時間 (Inhibit Time)	ro	なし	UNSIGNED16	0x05
04h	エントリ互換	ro	なし	UNSIGNED8	(予約領域)
05h	イベントタイマ	ro	なし	UNSIGNED16	0x00

■オブジェクト 1803h：PDO4 通信パラメータ送信 (Transmit PDO4 Communication Parameter)

デバイスが受信可能な、PDO4 の通信パラメータを表します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1803h	Transmit PDO Communication Parameter	PDO_COMM_PAR	RECORD	強制

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データタイプ	初期値
00h	エントリ数	ro	なし	UNSIGNED8	5
01h	COB-ID	ro	なし	UNSIGNED32	0x481
02h	送信タイプ	ro	なし	UNSIGNED8	0xFD
03h	禁止時間 (Inhibit Time)	ro	なし	UNSIGNED16	0x05
04h	エントリ互換	ro	なし	UNSIGNED8	(予約領域)
05h	イベントタイマ	ro	なし	UNSIGNED16	0x00

【5】 CANopen オブジェクトディクショナリ

■オブジェクト 1A00h : PDO1 マッピングパラメータ送信 (Transmit PDO1 Mapping Parameter)

Sub-index 1 以降には、マッピングする Index、Sub-index および Sub-index の長さを表します。

・設定値 : Index (16 ビット) + Sub-index (8 ビット) + 長さ (8 ビット)

オブジェクト内容 :

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1A00h	Transmit PDO1 Mapping Parameter	PDO_MAPPING	RECORD	強制

エン트리内容 :

Sub-Index	内 容	アクセス	PDO マッピング	データタイプ	初期値
00h	エントリ数	rw	なし	UNSIGNED8	3
01h	マッピングエントリ 1	rw	なし	UNSIGNED32	0x6041 0010
02h	マッピングエントリ 2	rw	なし	UNSIGNED32	0x6061 0008
03h	マッピングエントリ 3	rw	なし	UNSIGNED32	0x60FD 0020
04h	マッピングエントリ 4	rw	なし	UNSIGNED32	0
05h	マッピングエントリ 5	rw	なし	UNSIGNED32	0
06h	マッピングエントリ 6	rw	なし	UNSIGNED32	0
07h	マッピングエントリ 7	rw	なし	UNSIGNED32	0
08h	マッピングエントリ 8	rw	なし	UNSIGNED32	0

■オブジェクト 1A01h : PDO2 マッピングパラメータ送信 (Transmit PDO2 Mapping Parameter)

Sub-index 1 以降には、マッピングする Index、Sub-index および Sub-index の長さを表します。

・設定値 : Index (16 ビット) + Sub-index (8 ビット) + 長さ (8 ビット)

オブジェクト内容 :

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1A01h	Transmit PDO2 Mapping Parameter	PDO_MAPPING	RECORD	強制

エン트리内容 :

Sub-Index	内 容	アクセス	PDO マッピング	データタイプ	初期値
00h	エントリ数	rw	なし	UNSIGNED8	2
01h	マッピングエントリ 1	rw	なし	UNSIGNED32	0x6041 0010
02h	マッピングエントリ 2	rw	なし	UNSIGNED32	0x6064 0020
03h	マッピングエントリ 3	rw	なし	UNSIGNED32	0
04h	マッピングエントリ 4	rw	なし	UNSIGNED32	0
05h	マッピングエントリ 5	rw	なし	UNSIGNED32	0
06h	マッピングエントリ 6	rw	なし	UNSIGNED32	0
07h	マッピングエントリ 7	rw	なし	UNSIGNED32	0
08h	マッピングエントリ 8	rw	なし	UNSIGNED32	0

【5】 CANopen オブジェクトディクショナリ

■オブジェクト 1A02h : PDO3 マッピングパラメータ送信 (Transmit PDO3 Mapping Parameter)

Sub-index 1 以降には、マッピングする Index、Sub-index および Sub-index の長さを表します。

・設定値 : Index (16 ビット) + Sub-index (8 ビット) + 長さ (8 ビット)

オブジェクト内容 :

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1A02h	Transmit PDO3 Mapping Parameter	PDO_MAPPING	RECORD	強制

エン트리内容 :

Sub-Index	内 容	アクセス	PDO マッピング	データタイプ	初期値
00h	エン트리数	rw	なし	UNSIGNED8	2
01h	マッピングエン트리 1	rw	なし	UNSIGNED32	0x6041 0010
02h	マッピングエン트리 2	rw	なし	UNSIGNED32	0x606C 0020
03h	マッピングエン트리 3	rw	なし	UNSIGNED32	0
04h	マッピングエン트리 4	rw	なし	UNSIGNED32	0
05h	マッピングエン트리 5	rw	なし	UNSIGNED32	0
06h	マッピングエン트리 6	rw	なし	UNSIGNED32	0
07h	マッピングエン트리 7	rw	なし	UNSIGNED32	0
08h	マッピングエン트리 8	rw	なし	UNSIGNED32	0

■オブジェクト 1A03h : PDO4 マッピングパラメータ送信 (Transmit PDO4 Mapping Parameter)

Sub-index 1 以降には、マッピングする Index、Sub-index および Sub-index の長さを表します。

・設定値 : Index (16 ビット) + Sub-index (8 ビット) + 長さ (8 ビット)

オブジェクト内容 :

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
1A03h	Transmit PDO4 Mapping Parameter	PDO_MAPPING	RECORD	強制

エン트리内容 :

Sub-Index	内 容	アクセス	PDO マッピング	データタイプ	初期値
00h	エン트리数	rw	なし	UNSIGNED8	2
01h	マッピングエン트리 1	rw	なし	UNSIGNED32	0x6041 0010
02h	マッピングエン트리 2	rw	なし	UNSIGNED32	0x6077 0010
03h	マッピングエン트리 3	rw	なし	UNSIGNED32	0
04h	マッピングエン트리 4	rw	なし	UNSIGNED32	0
05h	マッピングエン트리 5	rw	なし	UNSIGNED32	0
06h	マッピングエン트리 6	rw	なし	UNSIGNED32	0
07h	マッピングエン트리 7	rw	なし	UNSIGNED32	0
08h	マッピングエン트리 8	rw	なし	UNSIGNED32	0

【5】 CANopen オブジェクトディクショナリ

5-2 メーカ固有オブジェクト（設定パラメータ）

■オブジェクト 2000h：ノード ID (Node ID)

このオブジェクトは、CAN デバイスのノード ID を設定します。
変更後の値は、電源の再起動で有効となります。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
2000h	Node ID	VAR	UNSIGNED8	強制

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	CAN Node ID	rw	なし	1 ～ 127	1

ドライバは、次のアボートコードを送信することがあります。

- 0x06040030 = 範囲外の数値

このオブジェクトの設定値は、EEPROM へ保存することができます。

! EEPROM への保存

- ドライバが“Operational Enabled”または“Quick Stop Active”のときは EEPROM へ保存できません。
- SDO で本オブジェクト(2000h)に新しいノード ID を書込む。
- 「パラメータストア」(1010h)に“save”を書込む。(処理時間に注意)
- NMT ノードリセット

注) 詳細は、11 章 各種の設定の「ノード ID の設定方法」を参照してください。

■オブジェクト 2001h：CAN ボーレート (CAN Baudrate)

このオブジェクトでノードの CAN のボーレートを設定します。
変更後の値は、電源の再起動で有効となります。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
2001h	CAN Baudrate	VAR	UNSIGNED16	強制

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	CAN Baudrate	rw	なし	表を参照	0x03E8

有効エン트리：

ボーレート	エントリ	ボーレート	エントリ	ボーレート	エントリ
10 kBit/s	0x000A	100 kbit/s	0x0064	500 kbit/s	0x01F4
20 kbit/s	0x0014	125 kbit/s	0x007D	800 kbit/s	0x0320
50 kbit/s	0x0032	250 kbit/s	0x00FA	1000 kbit/s	0x03E8

ドライバは、次のアボートコードを送信することがあります。

- 0x06040030 = 範囲外の数値

【5】CANopen オブジェクトディクショナリ

このオブジェクトは、EEPROM を変更してセーブすることができます。

EEPROM への保存

- ドライバが “Operational Enabled” または “Quick Stop Active” のときは EEPROM へ保存できません。
- SDO で本オブジェクト (2000h) に新しいノード ID を書込む。
- 「パラメータストア」 (1010h) に “save” を書込む。(処理時間に注意)
- NMT ノードリセット

注)詳細は、11 章 各種の設定の「ボーレートの変更方法」を参照してください。

■オブジェクト 3001h：絶対値リミットパラメータ (Absolute Limits Parameters)

このオブジェクトは、絶対値リミットを示しています。読み出し専用のパラメータです。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3001h	Absolute Limits Parameters	ARRAY	UNSIGNED32	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	-	5
01h	速度絶対値 (ABS)	ro	なし	0 ~ 2147483647 rpm	アプリケーション により定義
02h	加速絶対値 (ABS)	ro	なし	0 ~ 2147483647 rpm/s	
03h	速度プロファイルリミット	ro	なし	0 ~ 65535 rpm	
04h	予約領域	ro	なし	-	-
05h					

数値定義：

Sub-Index	フィールド	コンフィグレーション	定 義
01h	速度絶対値 (ABS)	rpm	速度プロファイルの最大絶対値 607Fh がリミットです。(最大速度プロファイル)
02h	加速絶対値 (ABS)	rpm/s	加速プロファイルの最大絶対値 60C5h (最大加速)、60C6h (最大減速) がリミットです。
03h	速度プロファイル リミット (最小値)	rpm	プロファイルモードの速度リミットです。最大速度プロファイル (607Fh) と速度フルスケール (3300h) との最小値

これらのパラメータは、最大定格で、読み出し専用です。

・速度パラメータは、速度絶対値のオブジェクト (3001h: 1) より小さく設定する必要があります。

例：最高速度が4500 rpm の場合、最高速度プロファイル(607Fh: 0)と速度フルスケール(3300h: 0)は、4500 rpm 以下に設定します。

これらのオブジェクトは、ユーザ単位に変換されてセーブされていることに注意してください。

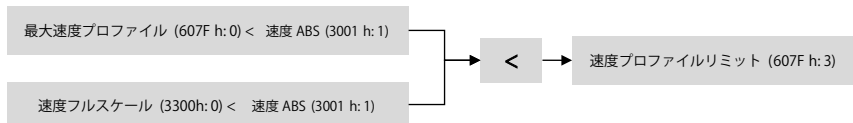
・加速パラメータは、加速絶対値のオブジェクト (3001h: 2) より小さくする必要があります。

例：加速絶対値が 2228 rpm/s の場合、最大加速(60C5h: 0)と最大減速(60C6h: 0)は、2228 rpm/s 以下に設定します。

これらのオブジェクトは、ユーザ単位に変換されてセーブされていることに注意してください。

【5】 CANopen オブジェクトディクショナリ

- ・速度プロファイルリミットのオブジェクト(607Fh: 3)は、速度プロファイルのリミット値です。このオブジェクトは、最大速度プロファイル(607Fh)と速度フルスケール(3300h)との最小値です。そのため、目標速度(60FFh: 0)は、速度プロファイルリミット(607Fh: 3)により制限されます。



ドライバは、次のアボートコードを送信することがあります。

- ・0x06090011 = Sub-index なし

■オブジェクト 3002h：モータブレーキパラメータ (Motor Brake Parameters)

このオブジェクトは、ブレーキパラメータの設定を示しています。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3002h	Brake Parameters	ARRAY	INTEGER16	強制 IF

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	-	7
01h	モータブレーキオプション	rw	なし	0、1	1
02h	モータブレーキ遅れ時間	rw	なし	1 ~ 32767	20
03h	モータブレーキ解放時間	rw	なし	1 ~ 32767	10
04h	ブレーキタイムアウト	rw	なし	1 ~ 32767	200
05h	自動 / 手動モード設定	rw	なし	0、1	0
06h	モータブレーキステータス	ro	なし	0、1	-
07h	ブレーキタイプ	ro	なし	1、2	1

数値定義：

Sub-Index	フィールド	コンフィグレーション	定 義
01h	モータブレーキオプション	0b 1b	モータブレーキ無効 モータブレーキ有効
02h	モータブレーキ遅れ時間	ms × 10	ブレーキ解放指令の遅れ時間 STBY ステータスとブレーキ解放の間の遅れ時間
03h	ブレーキ解放時間	ms × 10	ブレーキ解放前の STOP と RUN モード間の遅れ時間 遅れ時間は、モータの種類により異なります。
04h	ブレーキタイムアウト	ms × 10	フリーラン減速による最大設定時間 このタイムアウトの終わりにブレーキが動作し、ドライバは、STBY ステートになります。
05h	自動 / 手動モード設定	0b 1b	自動モードでブレーキ動作 手動モードでブレーキ動作
06h	モータブレーキステータス	0b 1b	ブレーキ状態：動作 → モータ軸ロック ブレーキ状態：解放 → モータ軸フリー
07h	ブレーキタイプ	1 2	電磁ブレーキ スプリングブレーキ

【5】CANopen オブジェクトディクショナリ

■自動 / 手動ブレーキ

自動ブレーキモードが初期設定されています。

オブジェクト (3002h: 5) で設定を変更し、設定値を EEPROM へセーブできます。

■手動ブレーキモード

ドライバに手動モードが設定されている場合、マスタコントローラがブレーキを制御します。

「デジタル出力」(60FEh のビット 1) を使用してモータブレーキを制御できます。

- ビット 1 = 0 : ブレーキ動作 → モータ軸ロック
- ビット 1 = 1 : ブレーキ解放 → モータ軸フリー

■自動ブレーキモード

ドライバに自動モードが設定されている場合、ブレーキは、次の図の通りに動作します。

- “Switched ON” (STANDBY) → “Operation Enabled” (RUN)

“Switched ON” (STANDBY ステート) から “Operation Enabled” (RUN ステート) へ移行する場合、次の図の通り、ブレーキは動作します。

ドライバが “Operation Enabled” へ移行すると、「モータブレーキ遅れ時間」(3002h: 2) で設定したタイマ経過後にブレーキを解放します。

そして「ブレーキ解放時間」(3002h: 3) 経過後、モータは加速を開始します。

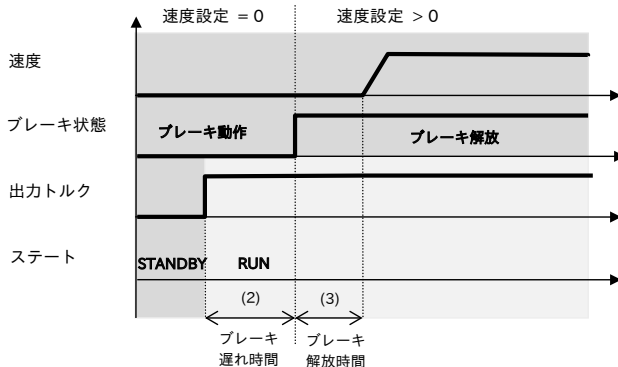


図 5-2 ブレーキタイミング (RUN ステートへ)

【5】 CANopen オブジェクトディクショナリ

- “Operation Enabled” (RUN) → “Switched ON” (STANDBY)

次の図は、“Operation Enabled” (RUN ステート) から “Switched ON” (STANDBY ステート) へ移行する場合のブレーキタイミングを示しています。

“Switched ON” (STANDBY ステート) へ移行し、モータがランプ停止した後にブレーキが動作します。ダイナミックブレーキを有効に設定していなければ、モータがフリーラン停止します。

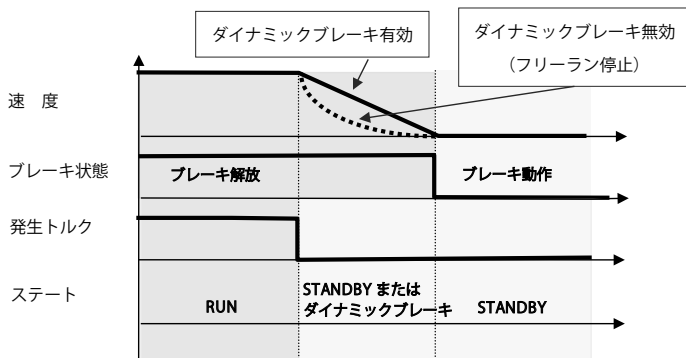


図 5-3 ブレーキタイミング (STANDBY ステートへ)

- “Operation Enabled” (RUN) → “Switched ON” (STANDBY) – 特殊ケース

次の図は、“Operation Enabled” (RUN ステート) から “Switched ON” (STANDBY ステート) へ移行する時に、ブレーキの開放時間がブレーキタイムアウト (オブジェクト 3002h: 4) の設定値よりも大きい場合のブレーキタイミングを示しています。

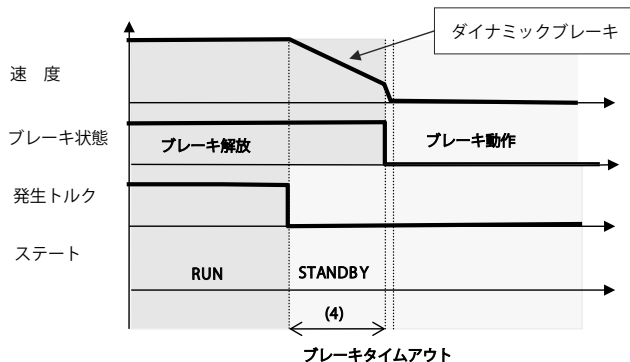


図 5-4 ブレーキタイムフレーム (STANDBY ステートへ)

【5】CANopen オブジェクトディクショナリ

・“Operation Enabled”(RUN) → 故障状態 (FAULT)

(1) ダイナミックブレーキ無効に設定：ブレーキが直ぐ動作

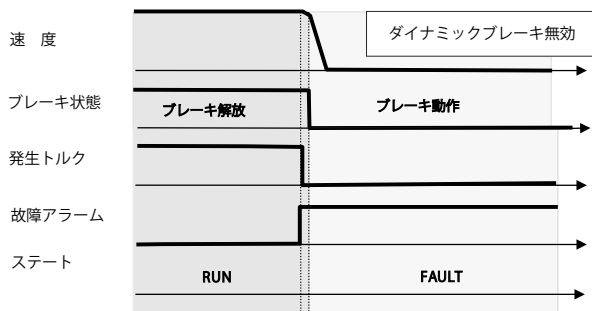


図 5-5 ブレーキタイムフレーム(ダイナミックブレーキ無効の故障状態へ)

(2) ダイナミックブレーキ有効に設定：速度が 0 になるとブレーキ動作

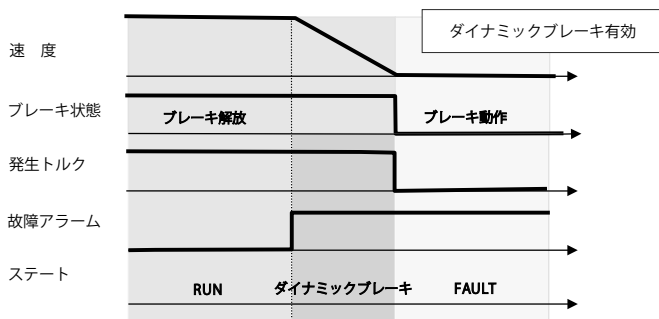


図 5-6 ブレーキタイムフレーム(ダイナミックブレーキ有効の故障状態へ)

ドライバは、次のアボートコードを送信することがあります。

- ・ 0x08000002：書き込み無効
- ・ 0x06040030：範囲外の数値
- ・ 0x06010002：読み出し専用のため書き込み不可(オブジェクト 0x3002: 6 と 0x3002: 7)
- ・ 0x06090011：Sub-index なし

ブレーキパラメータは、運転中に変更可能です。

本オブジェクトの設定値を変更後、EEPROM へ保存できます。

⚠ EEPROM への保存

- ドライバが“Operational Enabled”または“Quick Stop Active”のときはEEPROMへ保存できません。
- SDOで本オブジェクト(3002h)に新しいノードIDを書込む。
- 「パラメータストア」(1010h)に“save”を書込む。(処理時間に注意)
- NMTノードリセット

注)詳細は、8章 機能の「8-3 他の機能 モータブレーキマネジメント」を参照してください。

【5】 CANopen オブジェクトディクショナリ

■オブジェクト 3007h：ダイナミックブレーキパラメータ (Dynamic Brake Parameters)

このオブジェクトは、ダイナミックブレーキのパラメータを示します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3007h	Dynamic Brake Parameters	ARRAY	INTEGER16	強制 IF

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	-	7
01h	ダイナミックブレーキオプション	rw	なし	0、1	1
02h	保持トルク時間	rw	なし	1 ～ 32767	30
03h	ダイナミックブレーキステータス	ro	なし	0、1	-
04h	ステップランブ減算	rw	なし	1 ～ 8191	45
05h	ダイナミックブレーキ 最大タイムアウト	rw	なし	1 ～ 32767	300

数値定義：

Sub-Index	フィールド	コンフィグレーション	Definition
01h	ダイナミックブレーキオプション	0b 1b	ダイナミックブレーキを無効に設定 ダイナミックブレーキを有効に設定
02h	保持トルク時間	ms × 10	ランブ減速の終わりで、STOP ステータスとブレーキ解放間の遅れ時間 (STANDBY ステートに留まる前)
03h	ダイナミックブレーキステータス	0b 1b	ダイナミックブレーキの動作なし ダイナミックブレーキの動作中
04h	ステップランブ減速	rpm × 100/ sec	RUN ステートから STANDBY ステートへ遷移中の間で、ダイナミックブレーキによるランブ減速度の設定です。
05h	ダイナミックブレーキ最大タイムアウト	ms × 10	ダイナミックブレーキ最大タイムアウトは、ダイナミックブレーキ状態から抜けるための最大時間です。 「ステップランブ減速」より大きい必要があります。

マスタコントローラは、ダイナミックブレーキの有効 / 無効 (3007h: 1) を設定変更することができます。これは EEPROM にセーブすることができます。

0：ダイナミックブレーキモード無効

1：ダイナミックブレーキモード有効 (初期値)

【5】 CANopen オブジェクトディクショナリ

■ダイナミックブレーキ有効

次の図は、ダイナミックブレーキが動作したときの停止タイミングを示しています。
この状態は、ドライバが RUN ステート “Operation Enabled” から STANDBY ステート “Switched ON” へ移行する場合に発生します。

・標準ケース “Operation Enabled” (RUN) → “Switched ON” (STANDBY):

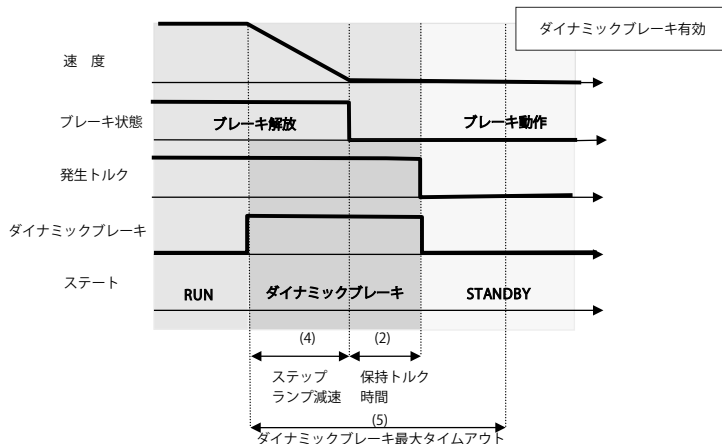


図 5-7 ダイナミックブレーキでの停止タイミング

・特殊ケース “Operation Enabled” (RUN) → “Switched ON” (STANDBY):

ダイナミックブレーキによるランプ減速が完了する前に、強制的にブレーキを動作させたときのタイミングを示しています。

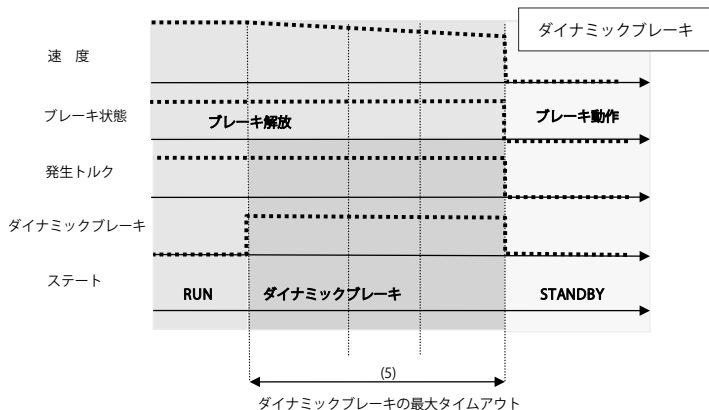


図 5-8 ダイナミックブレーキでの停止タイミング

【5】 CANopen オブジェクトディクショナリ

■ダイナミックブレーキ無効

次の図は、ダイナミックブレーキ無効時の停止タイミングを示しています。
モータはフリーラン停止し、 $\pm 5 \text{ rpm}$ 以下になるとブレーキが動作します。
モータがフリーラン減速中に「ブレーキタイムアウト」(3002h: 4) 設定値以上の時間が経過するとブレーキが強制的に動作します。(点線を参照)

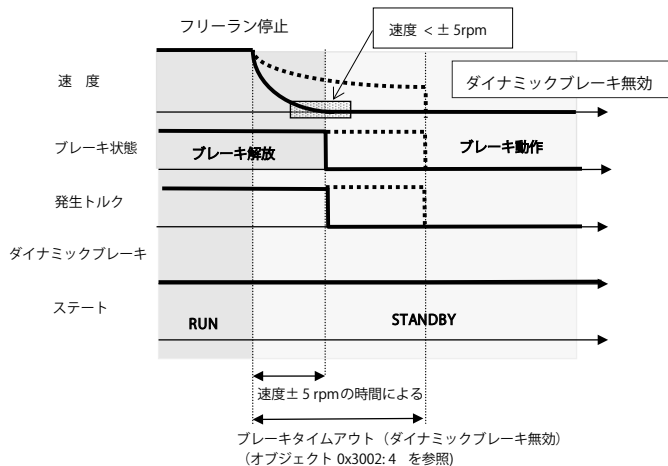


図 5-9 タイムフレーム(ダイナミックブレーキ無効)

ダイナミックブレーキ有効時に、RUN ステートから別のステートへ移行できる項目を次に示します。

- ・「コントロールワード」(6040h)に運転無効「Disable Operation」を設定し、STANDBY ステートへ移行
- ・デジタル入力3が ON されたとき、Emergency Input Enable ステートへ移行
- ・異常が発生すると FAULT ステートへ移行

ドライバは、次のアボートコードを送信することがあります。

- ・0x08000002：書き込み無効
- ・0x06040030：範囲外の数値
- ・0x06090031：パラメータ設定値が大きい。
(オブジェクト 0x3007: 4 を 0x3007: 5 より小さくする必要あり)
- ・0x06090032：パラメータ設定値が小さい。
(オブジェクト 0x3007: 5 を 0x3007: 4 より大きくする必要あり)
- ・0x06010002：数値が読出し専用のため書き込み不可(オブジェクト 0x3007: 3)
- ・0x06090011：Sub-index なし

注) ダイナミックブレーキのパラメータは、運転中に変更可能です。

【5】CANopen オブジェクトディクショナリ

このオブジェクトの設定は、EEPROM へ保存することができます。

⚠ EEPROM への保存

- ドライバが“Operational Enabled” または “Quick Stop Active” のときは EEPROM へ保存できません。
- SDO で本オブジェクト (3007h) に新しいノード ID を書込む。
- 「パラメータストア」 (1010h) に “save” を書込む。(処理時間に注意)
- NMT ノードリセット

注)詳細は、8 章 機能の「8-3 他の機能 ダイナミックブレーキ」を参照してください。

■オブジェクト 3008h：エマージェンシ入力有効パラメータ

(Emergency Input Enable Parameters)

このオブジェクトは、緊急停止入力「Emergency Input Enable」の機能を有効にするパラメータです。詳細は、8 章 機能の「8-2 デジタル I/O」を参照してください。

緊急停止入力は、“Operation Enable” および “Quick Stop Active” ステートから抜けるためのデジタル入力端子です。(本機能は、セーフティとして認証されていません。)

“Operation Enable” ステート「RUN」から “Switched ON” ステート「STANDBY」

または

“Quick Stop Active” ステート「STOP」から “Switched ON” ステート「STANDBY」

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3008h	Emergency Enabling Input Parameters	ARRAY	INTEGER16	強制 IF

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	-	7
01h	エマージェンシ有効オプション	rw	なし	0、1	1
02h	エマージェンシ入力反転	rw	なし	0、1	0
03h	エマージェンシ入力ステータス	ro	なし	0、1	0

数値定義：

Sub-Index	フィールド	コンフィグレーション	定 義
01h	エマージェンシ有効オプション “Emergency Enable Option”	0b	デジタル入力 3 を汎用に設定
		1b	デジタル入力 3 をエマージェンシ有効として設定
02h	エマージェンシ入力反転 “Emergency Input Neg”	0b	エマージェンシ入力レベルの反転なし
		1b	エマージェンシ入力レベルの反転
03h	エマージェンシ入力ステータス “Emergency Input Status”	0b	ステータス L レベル：エマージェンシ動作なし
		1b	ステータス H レベル：エマージェンシ動作

⚠ 注 意

緊急停止信号として扱われますが、セーフティとして認証されていません。

セーフティで認証されたパワー無効については、smartris ドライバの取扱説明書 (No.1802) の「5 章 安全機能 STO」を参照してください。

【5】 CANopen オブジェクトディクショナリ

デジタル入力 3 に緊急停止入力「Emergency Input Enable」 3008h, Sub-index 01h を「0：有効」に設定されている場合の状態機械 DSP402 の遷移を以下に示します。

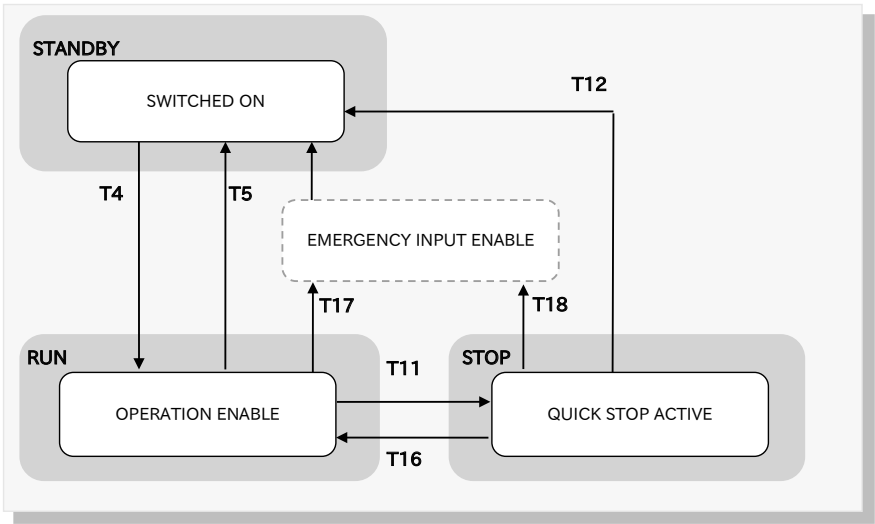


図 5-10 緊急停止入力有効時の状態機械

表 5-1 緊急停止入力有効の遷移

遷 移	命 令	内 容
T4	コントロールワード命令 = "Enable Operation" エマージェンシ入力ステータス L レベル	"Operation Enable" ステートへ移行 モータの運転が可能です。
T5	コントロールワード命令 = "Disable Operation" エマージェンシ入力ステータス L レベル	"Switched ON" ステートへ移行 ダイナミックブレーキを有効に設定している場合、 モータは、ランプ状に減速します。
T11	コントロールワード命令 = "Quick Stop"	"Quick Stop Active" ステートへ移行 サーボ ON 状態で停止します。
T16	コントロールワード命令 = "Enable Operation"	"Operation Enable" ステートへ移行 モータの運転が可能です。
T12	コントロールワード命令 = "Disable Operation"	"Switched ON" ステートへ移行 STANDBY 状態となります。
T17	エマージェンシ入力ステータス H レベル	エマージェンシ入力が ON "Switched ON" ステートへ移行が可能です。 ダイナミックブレーキを有効に設定している場合、 モータは、ランプ状に減速します。
T18	エマージェンシ入力ステータス H レベル	エマージェンシ入力が ON "Switched ON" ステートへ移行が可能です。

【5】 CANopen オブジェクトディクショナリ

■ CANopen ステートの遷移

- ・ **T4** : SWITCHED ON → OPERATION ENABLE

デジタル入力 3 「Emergency Input Enabled」が L レベルになり、コントロールワード (6040h) に “Enable Operation”= 0x0F が設定されると遷移



図 5-11 エマージェンシイネーブルステータス L レベル “Operation Enable”

- ・ **T5**、**T12** : OPERATION ENABLE → SWITCHED ON または QUICK STOP ACTIVE → SWITCHED ON
デジタル入力 3 「Emergency Input Enabled」が L レベルになり、コントロールワード (6040h) に “Disable Operation”= 0x07 が設定されると遷移

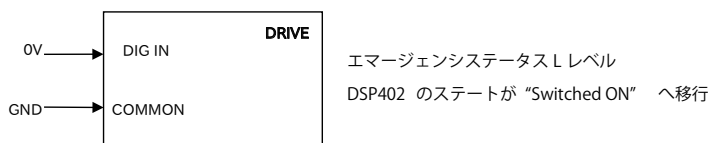


図 5-12 エマージェンシイネーブルステータス L レベル “Switched ON”

- ・ **T17**、**T18** : OPERATION ENABLE → SWITCHED ON または QUICK STOP ACTIVE → SWITCHED ON
デジタル入力 3 「Emergency Input Enabled」が H レベルになると遷移

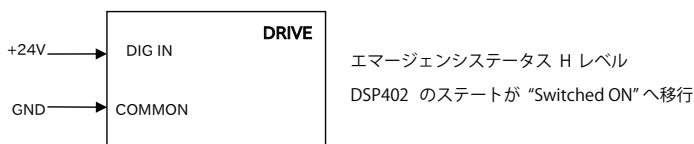


図 5-13 エマージェンシイネーブルステータス H レベル

ドライバは、次のアボートコードを送信することがあります。

- ・ 0x08000002 = 書き込み無効 (アプリケーションにより定義されるオプション)
- ・ 0x06010002 = 読み出し専用のため書き込み不可 (例: オブジェクト 3007h, Sub.3)
- ・ 0x06090011 = 該当する Sub-index なし

このオブジェクトの設定は、EEPROM へ保存することができます。

! EEPROM への保存

- ドライバが “Operational Enabled” または “Quick Stop Active” のときは EEPROM へ保存できません。
- SDO で本オブジェクト (3008h) に新しいノード ID を書込む。
- 「パラメータストア」 (1010h) に “save” を書込む。(処理時間に注意)
- NMT ノードリセット

【5】 CANopen オブジェクトディクショナリ

■オブジェクト 3050h：アナログ出力 1（Analog Output 1）

このオブジェクトは、アナログ出力パラメータを示します。

アナログ出力は、0 ～ 10 V です。

アナログ出力のコンフィグレーションを以下のように設定することができます。

- 0 = アナログ出力無効

- 1 = アナログ出力を汎用として設定します。アナログ出力は、0 ～ 4095 ビットが 0 ～ 10 V になります。

デジタル入力値は、Sub-index 2 で書込むことができます。

- 2 = アナログ出力を速度モニタとして設定します。

アナログ出力は、モータ速度に比例した電圧を出力します。

出力信号は 0 ～ 10 V で、－速度フルスケール～＋速度フルスケール [rpm]（オブジェクト 3300h）の範囲です。

- 3 = アナログ出力を電流モニタとして設定します。

アナログ出力は、電流に比例した電圧を出力します。

出力信号は 0 ～ 10 V で、0 ～ ピーク電流（オブジェクト 3303h: 2）の範囲です。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3050h	Analog Output 1	ARRAY	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	3	3
01h	コンフィグレーション	rw	なし	0 ～ 2	-
02h	デジタル値	rw	なし	0 ～ 4095	-
03h	出力	ro	なし	0 ～ 4095	-

数値定義：

Sub-Index	フィールド	数 値	定 義
01h	コンフィグレーション	0	無効（アナログ出力機能が無効）
		1	汎用（アナログ出力がデジタル値に比例）
		2	速度モニタリング（アナログ出力がモータ速度に比例）
		3	電流モニタ（アナログ出力が電流に比例）
02h	デジタル値	0 ～ 4095	アナログ出力の設定値 [0 ～ 4095] → [0 ～ 10 V]
03h	DAC 出力	0 ～ 4095	出力モニタのビット [0 ～ 4095] → [0 ～ 10 V]

ドライバは、次のアボートコードを送信することがあります。

- 0x08000002 = 書き込み無効（アプリケーションにより定義されるオプション）
- 0x06040030 = 範囲外の数値
- 0x05040001 = 汎用タイプのコンフィグレーションでないため命令が無効
- 0x06010002 = 読み出し専用のため書き込み禁止
- 0x06090011 = 該当する Sub-index なし

【5】 CANopen オブジェクトディクショナリ

このオブジェクトは、EEPROM を変更してセーブすることができます。

EEPROM への保存

- ドライバが “Operational Enabled” または “Quick Stop Active” のときは EEPROM へ保存できません。
- SDO で本オブジェクト (3050h) に新しいノード ID を書込む。
- 「パラメータストア」 (1010h) に “save” を書込む。(処理時間に注意)
- NMT ノードリセット

注) 詳細は、8 章 機能の「8-3 他の機能 DAC モニタリング」を参照してください。

■オブジェクト 3100h：コンフィグレーション 1 (Statusword)

このオブジェクトは、statuword (6041h) のコンフィギュレーションを変更します。

このオブジェクトの設定値を変更することで、EEPROM に保存して、ステータスワードの BIT 割り当てを変更することができます。初期値は、0 です。

以下に示す 2 つの設定変更が可能です。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3100h	Configuration 1	VAR	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	Statusword のコンフィグレーション	rw	なし	0 ~ 32767	0

数値定義：

値	Statusword のビット		
	ビット 8	ビット 14	ビット 15
0	緊急停止入力有効	セーフティ状態	異常状態
1	緊急停止入力有効	セーフティ状態	運転準備中

ビットの詳細：

名 称	意 味	内 容
緊急停止入力有効	緊急停止入力有効 機能の状態を示します。	0: アクティブでない。 1: アクティブ
セーフティ状態	ドライバのセーフティ状態を示します。	0: セーフティ状態でない。 1: セーフティ状態
異常状態	ドライバの異常状態を示します。	0: 異常なし 1: 異常検出中
運転準備中	ドライバの RUN (Operational enable) 状態を示します。	0: 運転準備が完了してない。 1: 運転準備完了 (電圧有効、STO がアクティブでない、異常なし 緊急停止入力が OFF)

注) モータ運転中でもこれらの設定値は、変更可能です。また、EEPROM へも保存することが可能です。

【5】 CANopen オブジェクトディクショナリ

■オブジェクト 3200h：電流 PID (Current PID)

このオブジェクトは、電流 PID パラメータを設定します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3200h	Current Pid	ARRAY	INTEGER16	強制

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	6	6
01h	PidCur Kp	rw	なし	0 ～ 32767	アプリケーションにより定義
02h	PidCur Ki	rw	なし	0 ～ 32767	
03h	PidCur Kv	rw	なし	0 ～ 32767	
04h	PidCur Kd (予約領域)	ro	なし	0 ～ 32767	
05h	PidCur N (予約領域)	ro	なし	0 ～ 32767	予約領域
06h	PidCur FF (予約領域)	ro	なし	0 ～ 32767	

ドライバは、次のアバートコードを送信することがあります。

- ・ 0x08000002 = 書き込み無効 (アプリケーションにより定義されるオプション)
- ・ 0x06040030 = 範囲外の数値
- ・ 0x06090011 = 該当する Sub-index なし

注) 運転中でもこれらの設定を変更することができます。

このオブジェクトは、EEPROM を変更してセーブすることができます。

! EEPROM への保存	
<ul style="list-style-type: none">● ドライバが “Operational Enabled” または “Quick Stop Active” のときは EEPROM へ保存できません。● SDO で本オブジェクト (3200h) に新しいノード ID を書込む。● 「パラメータストア」 (1010h) に “save” を書込む。(処理時間に注意)● NMT ノードリセット	

■オブジェクト 3201h：速度 PID (Speed PID)

このオブジェクトは、速度 PID のパラメータを設定します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3201h	Speed PID	ARRAY	INTEGER16	強制

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	Number of Entries	ro	なし	6	6
01h	PidVel Kp	rw	なし	0 ～ 32767	アプリケーションにより定義
02h	PidVel Ki	rw	なし	0 ～ 32767	
03h	PidVel Kv	rw	なし	0 ～ 32767	
04h	PidVel Kd	ro	なし	0 ～ 32767	
05h	PidVel N	ro	なし	0 ～ 32767	予約領域
06h	PidVel FF	ro	なし	0 ～ 32767	

【5】CANopen オブジェクトディクショナリ

ドライバは、次のアボートコードを送信することがあります。

- 0x08000002 = 書き込み無効 (アプリケーションにより定義されるオプション)
- 0x06040030 = 範囲外の数値 (エントリ内容を参照)
- 0x06090011 = 該当する Sub-index なし

注) 運転中でもこれらの設定を変更することができます。

このオブジェクトは、EEPROM を変更してセーブすることができます。

! EEPROM への保存

- ドライバが “Operational Enabled” または “Quick Stop Active” のときは EEPROM へ保存できません。
- SDO で本オブジェクト (3201h) に新しいノード ID を書き込む。
- 「パラメータストア」 (1010h) に “save” を書き込む。 (処理時間に注意)
- NMT ノードリセット

■ オブジェクト 3300h : 速度フルスケール (Velocity Full Scale)

このオブジェクトは、速度のフルスケールを示しています。

オブジェクト内容：

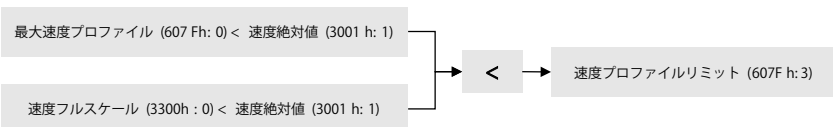
Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3300h	Velocity Full Scale	VAR	UNSIGNED16	強制

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	rw	なし	0 ~ 32767	アプリケーションにより定義

速度フルスケールは、速度絶対値 (3001h: 1) より小さく設定する必要があります。

本オブジェクトは、「最大速度プロファイル」 (607Fh) と同じように速度プロファイルモードで速度リミットを定義します。以下にリミット設定の図式を示します。



ドライバは、次のアボートコードを送信することがあります。

- 0x06040030 = 範囲外の数値
- 0x06090031 = パラメータ設定値が大きい。 (0x3001: 1 より小さくする必要があります。)

【5】 CANopen オブジェクトディクショナリ

このオブジェクトは、EEPROM を変更してセーブすることができます。



EEPROM への保存

- ドライバが “Operational Enabled” または “Quick Stop Active” のときは EEPROM へ保存できません。
- SDO で本オブジェクト (3300h) に新しいノード ID を書込む。
- 「パラメータストア」 (1010h) に “save” を書込む。(処理時間に注意)
- NMT ノードリセット

【5】 CANopen オブジェクトディクショナリ

5-3 メーカ固有オブジェクト（運転時間モニタリングデータ）

■オブジェクト 2002h：ドライバ制御ステート (Drive Control State)

このオブジェクトは、ドライバの状態を取得します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
2002h	Drive Control State	VAR	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲
00h	Drive Control State	ro	あり	下表を参照

有効入力：

数 値	ビット	名 称	内 容
0x0001	0	速度運転 (Run Velocity)	速度制御モードでモータを運転中
0x0002	1	スタンバイ (Standby)	ドライバはスタンバイ状態で PWM 出力オフです。
0x0004	2	停止 (Stop)	ドライバは停止し、保持トルクが発生しています。
0x0008	3	オフ (Off)	未使用
0x0010	4	アラーム (Alarm)	ドライバがアラームを検出中
0x0020	5	電流運転 (Run Current)	トルク制御モードでモータを運転中
0x0040	6	初期化 (Init)	ドライバが初期化中で、PWM 出力オフです。
0x0080	7	セーフティ (Safety)	STO が動作しておりドライバはセーフティ状態で、PWM 出力オフです。
0x0100	8	ポジションナ運転 (Run Positioner)	ポジションナ制御モード

■オブジェクト 2003h：ワーニング (Warning)

このオブジェクトは、ドライバにログとして記録されたワーニング(警告)を取得することができます。

ワーニングをクリアするには、コントロールワード(6040h)のワーニングビット(ビット 11)をセットします。

定義されているワーニングは、以下の通りです。

- ・CAN 通信ワーニング
- ・I²t リミットワーニング
- ・EEPROM ワーニング
- ・パラメータ更新ワーニング
- ・出荷時パラメータワーニング
- ・CANopen パラメータワーニング
- ・DAC コンフィグレーションワーニング
- ・温度ワーニング
- ・トルク機能制限ワーニング
- ・アナログモードワーニング

【5】 CANopen オブジェクトディクショナリ

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
2003h	Warning	VAR	UNSIGNED32	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲
00h	Drive Warning	ro	あり	下表を参照

ワーニングリスト：

ビット	名 称	内 容
0	ノードガーディング	マスタがノードガーディング・メッセージを失った。
1	I ² T リミット	ドライバが、I ² T リミットとなり過負荷
2	ストア / リストア / ロード EEPROM 命令	ドライバが“Ready to Switch ON”または“Disabled State” 状態ではないときに、ストア / リストア / ロード命令を 受け付けた。
3	出荷時パラメータの更新	予約領域
4	出荷時パラメータ書込み	
5	CANopen 無効アラーム	
6	CANopen オブジェクト初期化	
7	DAC コンフィグレーション	DAC コンバータの設定が無効
8	ヒートシンク温度	ヒートシンク温度が、閾値よりも 5℃低いためのワーニング (閾値はアプリケーションによる)
9	プリント基板温度	プリント基板温度が、63℃より高いためのワーニング
10	モータ温度	モータ温度が、130℃より高いためのワーニング
11	CAN バス通信エラー	CAN バスエラーのワーニング (passive または Bus Off)
12	トルクリミット機能	トルクリミットが動作したことによるワーニング
13	トルクリミット中	トルクリミット中のワーニング
14	アナログモードパラメータ	アナログモードの初期化エラーのワーニング
15	CAN テストモード	CAN 自動テストのワーニング
16 ~ 19	予約領域	-
20	I ² T 動作不可のワーニング	モータを保護するためのワーニング。ドライバが、パワー ON 中に I ² T アラームを 2 回連続して検出した場合に発生し ます。このワーニングが動作中に運転した場合、アラーム (0x2352) が発生します。
21 ~ 31	フリー	フリー

■オブジェクト 2004h：サーボドライバステート (State Servo Drive Machine)

このオブジェクトは、DSP402 プロファイルに準拠した smartris 独自の状態機械に従ったドライバ状態を示しています。

状態の定義：

- ・0 = サーボドライバステート：INIT - Not ready to switch on
- ・1 = サーボドライバステート：SAFETY
- ・2 = サーボドライバステート：STOP - Quick stop active
- ・3 = サーボドライバステート：RUN - Operation enabled
- ・4 = サーボドライバステート：STANDBY – Switched on
- ・5 = サーボドライバステート：DYNAMIC BRAKE
- ・6 = サーボドライバステート：予約領域

【5】 CANopen オブジェクトディクショナリ

- ・7 = サーボドライバステート：予約領域
- ・10 = サーボドライバステート：INIT - Switch on disabled
- ・11 = サーボドライバステート：INIT - Ready to switch on
- ・16 = サーボドライバステート：FAULT - Fault reaction active
- ・17 = サーボドライバステート：FAULT - Fault
- ・18 = サーボドライバステート：FAULT - Error

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
2004h	Drive Status	VAR	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	初期値
00h	Drive State	ro	あり	下表を参照

有効エントリ：

数 値	ドライバステート	DSP402 ステート	数 値	ドライバステート	DSP402 ステート
0	INIT	Not Ready to Switch ON	8, 9	-	-
1	SAFETY	-	10	INIT	Switch ON Disabled
2	STOP	Quick Stop Active	11	INIT	Ready to Switch ON
3	RUN	Operation Enabled	12 to 15	-	-
4	STANDBY	Switched ON	16	FAULT	Fault Reaction Active
5	DYNAMIC BRAKE	-	17	FAULT	Fault
6, 7	予約領域	-	18	FAULT	Error

■オブジェクト 2030h：ドライバ温度(Temperature Drive)

このオブジェクトは、ドライバ温度を示します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
2030h	Temperature Drive	VAR	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	単 位
00h	Drive temperature	ro	あり	-150 ～ 1250	° C /10

■オブジェクト 2031h：モータ温度(Temperature Motor)

このオブジェクトは、モータ温度を示します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
2031h	Temperature Motor	VAR	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	単 位
00h	Motor temperature	ro	あり	-400 ～ 1300	° C /10

【5】 CANopen オブジェクトディクショナリ

■オブジェクト 2032h：ヒートシンク温度(Temperature Heat Sink)

このオブジェクトは、ヒートシンク温度を示します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
2032h	Temperature Heat Sink	VAR	INTEGER16	オプション

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	単 位
00h	Heat Sink temperature	ro	あり	-400 ~ 1300	°C/10

■オブジェクト 2041h：バス電圧(Voltage Bus)

このオブジェクトは、バス電圧値を示します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
2041h	Voltage Bus	VAR	INTEGER16	オプション

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	単 位
00h	Voltage Bus	ro	あり	0 ~ 11000	V/100

■オブジェクト 2050h：トルク電流(Torque Current)

このオブジェクトは、トルク電流値を示します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
2050h	Torque Current	VAR	INTEGER16	オプション

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	単 位
00h	Torque Current	ro	あり	-32767 ~ 32767	A/100

■オブジェクト 2053h：速度フィルタ(Velocity Filtered)

このオブジェクトは、速度フィルタ値を示します。

速度フィルタは、オブジェクト(3005h:2)のフィルタパラメータを使用して変更できます。

速度に対して1次遅れのフィルタで、単位は、[rpm/4]です。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
2053h	Velocity Filtered	VAR	INTEGER16	オプション

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	単 位
00h	Velocity Filtered	ro	あり	-32767 ~ 32767	rpm/4

【5】 CANopen オブジェクトディクショナリ

■オブジェクト 2054h：I²t エネルギ(Energy I²t)

このオブジェクトは、I²t エネルギの%を示します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
2054h	Energy I ² t	VAR	UNSIGNED16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	単 位
00h	Energy I ² t	ro	あり	0 ～ 65536	[%]

次の図は、トルク電流 (2050h) と I²t エネルギ (2054h) を示しています。

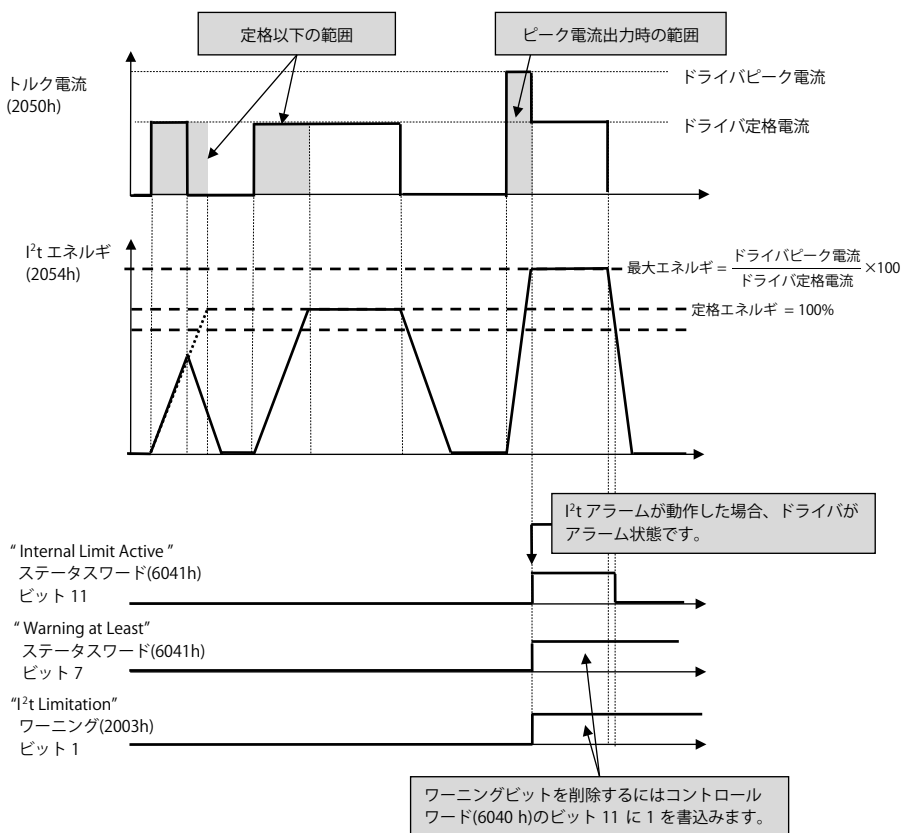


図 5-14 I²t エネルギモニタ

【5】 CANopen オブジェクトディクショナリ

■オブジェクト 2060h：インパルス (Impulse)

このオブジェクトは、インクリメント単位での電気角（モータの極対数による）を示しています。
最大値は、回転検出器の分解能です。
分解能は、フィードバックパラメータのオブジェクト (3004h: 2) で読出すことができます。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
2060h	Impulse	VAR	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	単 位
00h	Impulse	ro	なし	0 ～ 16384	Inc

■オブジェクト 2070h：トルク電流ウィンドウ (Torque Current Windowed)

このオブジェクトは、トルク電流値を移動平均でモニタします。
読出し専用のオブジェクトです。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
2070h	Torque Current Windowed	VAR	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	単 位
00h	Torque Current Windowed	ro	なし	-32767 ～ 32767	A/100

■ Object 3003h：ドライバサイズパラメータ (Drive Size Parameters)

このオブジェクトは、ドライバの最新パラメータを定義しています。
読出し専用のオブジェクトです。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3003h	Drive Size Parameters	VAR	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	7	2
01h	電流検出のフルスケール	ro	なし	0 ～ 32767	アプリケーション により定義
02h	ピーク電流	ro	なし	0 ～ 32767	
03h	定格電流	ro	なし	0 ～ 32767	
04h	I_t^2	ro	なし	0 ～ 32767	
05h ～ 07h	予約領域	ro	なし	0 ～ 32767	

【5】 CANopen オブジェクトディクショナリ

数値定義：

Sub-Index	フィールド	コンフィグレーション	定 義
01h	電流検出のフルスケール	Arms	電流検出の最大スケール
02h	ピーク電流	Arms/100	ドライバのピーク電流
03h	定格電流	Arms/100	ドライバの定格電流
04h	I _t タイムアウト	sec	過負荷アラームが発生する時間
05h ~ 07h	予約領域	-	-

■オブジェクト Object 3004h：フィードバックパラメータ (Feedback Parameters)

このオブジェクトは、回転検出器の仕様を定義しています。読み出し専用のオブジェクトです。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3004h	Feedback Parameters	ARRAY	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	15	15
01h	フィードバックタイプ	ro	なし	0 ~ 32767	アプリケーションによる
02h	分解能	ro	なし	0 ~ 32767	0x4000
03h	絶対位置 (初期値)16 ビット LSB 注)	ro	なし	0 ~ 32767	-
04h	絶対位置 (初期値)16 ビット MSB 注)	ro	なし	0 ~ 32767	-
05h	絶対位置 (検出値)16 ビット LSB 注)	ro	なし	0 ~ 32767	-
06h	絶対位置 (検出値)16 ビット MSB 注)	ro	なし	0 ~ 32767	-
0Ah	エンコーダコマンド 注)	rw	なし	0 ~ 4	0

数値定義：

Sub-Index	フィールド	コンフィグレーション	定 義
01h	フィードバックタイプ	0 ~ 2	0：レゾルバ/2：アブソリュートエンコーダ
02h	分解能	0 ~ 32767	回転検出器の分解能
03h	絶対位置 (初期値)16 ビット LSB 注)	0 ~ 32767	例) 位置 = 0x0029 7421 の場合、値 = 0x7421
04h	絶対位置 (初期値)16 ビット MSB 注)	0 ~ 32767	例) 位置 = 0x0029 7421 の場合、値 = 0x0029
05h	絶対位置 (検出値)16 ビット LSB 注)	0 ~ 32767	絶対位置
06h	絶対位置 (検出値)16 ビット MSB 注)	0 ~ 32767	
0Ah	エンコーダコマンド 注)	0 ~ 4	0：なし / 1 ~ 3：予約領域 / 4：位置読み出し

注) Sub-Index 03h ~ 0Ah はアブソリュートエンコーダ仕様の smartris でのみ使用できます。

0Ah を「4: 位置読み出し」に設定することで Sub-Index 05h,06h の値がリセットされます。

ドライバは、次のアボートコードを送信することがあります。

- 0x06090011 = Sub-index なし

■オブジェクト 3005h：フィルタパラメータ (Filter Parameters)

このオブジェクトは、ドライバのフィルタパラメータを示します。

オブジェクト (2053h) 速度フィルタのフィルタ極 (1 次遅れ) は、Sub-index 05h のみで変更できます。

【5】 CANopen オブジェクトディクショナリ

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3005h	Filter Parameters	VAR	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	10	10
01h	フィルタ極 1 セットポイント	ro	なし	0 ～ 10000	アプリケーション により定義
02h	フリー	ro	なし	0 ～ 10000	
03h	速度フィルタ極 1	ro	なし	0 ～ 10000	
04h	速度フィルタ極 2	ro	なし	0 ～ 10000	
05h	速度フィルタ (2053h) フィルタ極 1	nw	なし	0 ～ 10000	50
06h	実速度 (606Ch) フィルタ極 1	ro	なし	0 ～ 10000	
07h	Pid フィルタ極 1	ro	なし	0 ～ 10000	アプリケーション により定義
08h	Pid フィルタ極 2	ro	なし	0 ～ 10000	
09h	フリー	ro	なし	0 ～ 10000	
10h	フリー	ro	なし	0 ～ 10000	

数値定義：

Sub-Index	フィールド	コンフィグレーション	定 義
01h	フィルタ極 1 設定ポイント	0 ～ 10000 Hz	予約領域(値の変更不可)
02h	フリー	0 ～ 10000 Hz	-
03h	速度フィルタ極 1	0 ～ 10000 Hz	予約領域(値の変更不可)
04h	速度フィルタ極 2	0 ～ 10000 Hz	
05h	速度フィルタ (2053h) フィルタ極 1	0 ～ 10000 Hz	オブジェクト (2053h) の速度フィルタを変更することができます。0 は設定不可です。 単位：Hz (初期値 50Hz)
06h	実速度 (606Ch) フィルタ極 1	0 ～ 10000 Hz	オブジェクト (606Ch) の実速度フィルタを変更することができます。0 は設定不可です。 単位：Hz (初期値 50Hz)
07h	Pid フィルタ極 1	0 ～ 10000 Hz	予約領域(値の変更不可)
08h	Pid フィルタ極 2	0 ～ 10000 Hz	
09h	フリー	0 ～ 10000 Hz	-
10h		0 ～ 10000 Hz	

ドライバは、次のアボートコードを送信することがあります。

- 0x06090011 = Sub-index なし
- 0x08000002 = オブジェクトの書き込み無効のため、データ書き込み不可

Sub-index 05h のみ設定値を変更できます。EEPROM へ保存することも可能です。

! EEPROM への保存
<ul style="list-style-type: none"> ● ドライバが “Operational Enabled” または “Quick Stop Active” のときは EEPROM へ保存できません。 ● SDO で本オブジェクト (3005h) に新しいノード ID を書き込む。 ● 「パラメタストア」 (1010h) に “save” を書き込む。(処理時間に注意) ● NMT ノードリセット

【5】 CANopen オブジェクトディクショナリ

■オブジェクト 3006h：モータ仕様 (Motor Specifics)

このオブジェクトは、サーボモータの仕様を示しています。読み出し専用のオブジェクトです。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3006h	Motor Specific Settings	ARRAY	INTEGER16	オプション

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	3	3
01h	モータ製品コード	ro	なし	0 ~ 32767	アプリケーション により定義
02h	モータ最高速度	ro	なし	0 ~ 32767	
03h	極数 N	ro	なし	0 ~ 32767	
04h	モータ Kt	ro	なし	0 ~ 32767	準備中
05h	モータ検出タイプ	ro	なし	0 ~ 32767	準備中

数値定義：

Sub-Index	フィールド	コンフィグレーション	備 考
01h	モータ製品コード	-	1: B40, 2: B63, 3: B71
02h	モータ最高速度	rpm	モータ回転数の最高速度
03h	極数 N	pole	モータ極数
04h	モータ Kt	-	準備中
05h	モータ検出タイプ	-	準備中

■オブジェクト 3021h：デジタル入力 1 (Digital Input 1)

このオブジェクトは、デジタル入力 1 の設定を示しています。読み出し専用のオブジェクトです。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3021h	Digital Input 1	ARRAY	INTEGER16	オプション

エン트리内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	5	5
01h	コンフィグレーション	ro	なし	0 ~ 14	アプリケーション により定義
02h	ステート	ro	なし	0, 1	
03h	レベル	ro	なし	0, 1	
04h	フリー	-	-	-	-
05h					

【5】 CANopen オブジェクトディクショナリ

数値定義：

Sub-Index	フィールド	数 値	定 義
01h	コンフィグレーション	0 ~ 14	3, 4 の機能のみ使用できます。(他は未実装) 0：定義なし 1：“RUN” 機能としてデジタル入力設定 2：“STOP” 機能としてデジタル入力設定 3：“EMERGENCY” 機能としてデジタル入力設定 4：“RESET” 機能としてデジタル入力設定 5：“DCW” 機能としてデジタル入力設定 6：“DCCW” 機能としてデジタル入力設定 7：“SETVEL1” 機能としてデジタル入力設定 8：“SETVEL2” 機能としてデジタル入力設定 9：“SETVEL3” 機能としてデジタル入力設定 10：“SETVEL4” 機能としてデジタル入力設定 11：“VEL/CUR” 機能としてデジタル入力設定 12：“DIR” 機能としてデジタル入力設定 13：“POS” 機能としてデジタル入力設定 14：“HOME” 機能としてデジタル入力設定
02h	ステート	0、1	0：L レベル デジタル入力動作なし 1：H レベル デジタル入力動作
03h	レベル	0、1	0：ポジティブエッジ（標準設定） 1：ネガティブエッジ

ドライバは、次のアボートコードを送信することがあります。

- 0x06090011 = Sub-index なし

■オブジェクト 3022h：デジタル入力 2 (Digital Input 2)

このオブジェクトは、デジタル入力 2 の設定を示します。読み出し専用のオブジェクトです。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3022h	Digital Input 2	ARRAY	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	5	5
01h	コンフィグレーション	ro	なし	0 ~ 14	アプリケーション により定義
02h	ステート	ro	なし	0、1	
03h	レベル	ro	なし	0、1	
04h	フリー	-	-	-	-
05h					

【5】 CANopen オブジェクトディクショナリ

数値定義：

Sub-Index	フィールド	数 値	定 義
01h	コンフィグレーション	0 ~ 14	3, 4 の機能のみ使用できます。(他は未実装) 0：定義なし 1：“RUN” 機能としてデジタル入力設定 2：“STOP” 機能としてデジタル入力設定 3：“EMERGENCY” 機能としてデジタル入力設定 4：“RESET” 機能としてデジタル入力設定 5：“DCW” 機能としてデジタル入力設定 6：“DCCW” 機能としてデジタル入力設定 7：“SETVEL1” 機能としてデジタル入力設定 8：“SETVEL2” 機能としてデジタル入力設定 9：“SETVEL3” 機能としてデジタル入力設定 10：“SETVEL4” 機能としてデジタル入力設定 11：“VEL/CUR” 機能としてデジタル入力設定 12：“DIR” 機能としてデジタル入力設定 13：“POS” 機能としてデジタル入力設定 14：“HOME” 機能としてデジタル入力設定
02h	ステート	0、1	0 = L レベル：デジタル入力動作なし 1 = H レベル：デジタル入力動作
03h	レベル	0、1	0：ポジティブエッジ（標準設定） 1：ネガティブエッジ

ドライバは、次のアボートコードを送信することがあります。

- 0x06090011 = Sub-index なし

■オブジェクト 3023h：デジタル入力 3（Digital Input 3）

このオブジェクトは、デジタル入力 3 の設定を示します。読み出し専用のオブジェクトです。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3023h	Digital Input 3	ARRAY	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	5	5
01h	コンフィグレーション	ro	なし	0 ~ 14	アプリケーション により定義
02h	ステート	ro	なし	0、1	
03h	レベル	ro	なし	0、1	
04h	フリー	-	-	-	-
05h					

【5】 CANopen オブジェクトディクショナリ

数値定義：

Sub-Index	フィールド	数 値	定 義
01h	コンフィグレーション	0 ~ 14	3, 4 の機能のみ使用できます。(他は未実装) 0 = 定義なし 1 = "RUN" 機能としてデジタル入力設定 2 = "STOP" 機能としてデジタル入力設定 3 = "EMERGENCY" 機能としてデジタル入力設定 4 = "RESET" 機能としてデジタル入力設定 5 = "DCW" 機能としてデジタル入力設定 6 = "DCCW" 機能としてデジタル入力設定 7 = "SETVEL1" 機能としてデジタル入力設定 8 = "SETVEL2" 機能としてデジタル入力設定 9 = "SETVEL3" 機能としてデジタル入力設定 10 = "SETVEL4" 機能としてデジタル入力設定 11 = "VEL/CUR" 機能としてデジタル入力設定 12 = "DIR" 機能としてデジタル入力設定 13 = "POS" 機能としてデジタル入力設定 14 = "HOME" 機能としてデジタル入力設定
02h	ステート	0, 1	0 = L レベル：デジタル入力動作なし 1 = H レベル：デジタル入力動作
03h	レベル	0, 1	0 : ポジティブエッジ (標準設定) 1 : ネガティブエッジ

ドライバは、次のアボートコードを送信することがあります。

- 0x06090011 = Sub-index なし

■オブジェクト 3024h：デジタル入力 4 (Digital Input 4)

このオブジェクトは、デジタル入力 4 の設定を示します。読み出し専用のオブジェクトです。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3024h	Digital Input 4	ARRAY	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	5	5
01h	コンフィグレーション	ro	なし	0 ~ 14	アプリケーション により定義
02h	ステート	ro	なし	0 ~ 1	
03h	レベル	ro	なし	0 ~ 1	
04h, 05h	フリー	-	-	-	-

【5】 CANopen オブジェクトディクショナリ

数値定義：

Sub-Index	フィールド	数 値	定 義
01h	コンフィグレーション	0 ~ 14	3, 4 の機能のみ使用できます。(他は未実装) 0：定義なし 1：“RUN” 機能としてデジタル入力設定 2：“STOP” 機能としてデジタル入力設定 3：“EMERGENCY” 機能としてデジタル入力設定 4：“RESET” 機能としてデジタル入力設定 5：“DCW” 機能としてデジタル入力設定 6：“DCCW” 機能としてデジタル入力設定 7：“SETVEL1” 機能としてデジタル入力設定 8：“SETVEL2” 機能としてデジタル入力設定 9：“SETVEL3” 機能としてデジタル入力設定 10：“SETVEL4” 機能としてデジタル入力設定 11：“VEL/CUR” 機能としてデジタル入力設定 12：“DIR” 機能としてデジタル入力設定 13：“POS” 機能としてデジタル入力設定 14：“HOME” 機能としてデジタル入力設定
02h	ステート	0, 1	0=L レベル：デジタル入力動作なし 1=H レベル：デジタル入力動作
03h	レベル	0, 1	0：ポジティブエッジ（標準設定） 1：ネガティブエッジ

ドライバは、次のアボートコードを送信することがあります。

- 0x06090011 = Sub-index なし

■オブジェクト 3030h：デジタル出力機能(Digital Output Function)

オブジェクトは、デジタル出力の機能について示します。読み出し専用のオブジェクトです。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3030h	Digital Output Function	ARRAY	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	6	6
01h	コンフィグレーション	ro	なし	0 ~ 32767	アプリケーション により定義 -
02h	ステート機能	ro	なし	0 ~ 32767	
03h	レベル機能	ro	なし	0 ~ 32767	
04h	デジタル出力モニタ	ro	なし	0 ~ 32767	
05h, 06h	予約領域	ro	なし	0 ~ 32767	

各ビットのファンクション (Func.) No. および機能は、以下の通りです。

ビット	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Func. No.	F16	F15	F14	F13	F12	F11	F10	F9	F8	F7	F6	F5	F4	F3	F2	F1

- ビット 0 - F1：異常の有無：ドライバがアラーム状態であるかどうかを示します。
「1」の場合は、ドライバにアラームがなく正常な状態です。
- ビット 1 - F2：警告の有無：ドライバが警告を発生したかどうかを示します。
警告の内容を知るために 2003h を確認してください。

【5】 CANopen オブジェクトディクショナリ

- ビット 2 - F3：目標到達：予約領域
- ビット 3 - F4：ブレーキ状態：ブレーキの状態を示します。
- ビット 4 - F5：準備中：予約領域
- ビット 5 - F6：モータ駆動中：モータが回転していることを示します。
閾値は、オブジェクト 606Fh:0「速度閾値」および 6070h:0「速度スレッシュホールド時間」で設定します。速度がこの閾値を超えると「1」になります。
全てのプロファイルで使用可能です。
- ビット 6 - F7：360°検出：予約領域

Sub-Index	フィールド	数値	定義
1	コンフィグレーション	すべてのビットは、0 または 1 で有効な機能が定義されています。	各ビットの機能設定です。 0：無効 / 1：有効
2	ステート機能		各ビットの状態を示します。 0：アクティブ / 1：アクティブでない
3	レベル機能		各ビットの出力レベルを示します。 0：ポジティブ (標準) / 1：ネガティブ
4	デジタル出力モニタ	デジタル出力 1：ビット 0 = 0 or 1 デジタル出力 2：ビット 1 = 0 or 1 デジタル出力 3：ビット 2 = 0 or 1 デジタル出力 4：ビット 3 = 0 or 1	各 BIT の状態をモニタします。 0：H レベル / 1：L レベル

以下は、デジタル出力の初期設定です。

- デジタル出力 1：異常の有無 F1
- デジタル出力 2：警告の有無 F2
- デジタル出力 3：モータ駆動中 F6
- デジタル出力 4：ブレーキ状態 F4

3030h: 1 の値は、0x002B (0000 0000 0010 1011b) になります。

例) 3030h: 2 の値が、0x0002 (0000 0000 0010b) の場合、出力 2 (Warning Function 2) が H レベルのため、ドライバに異常が発生していて、3030h:4 は、0x0002 (0000 0000 0010b) となります。

3030h: 3 の値が 0x0000 の場合、全てのデジタル出力がポジティブとなります。

■オブジェクト 3031h：デジタル出力 1 (Digital Output 1)

このオブジェクトは、デジタル出力 1 の機能について示します。読み出し専用のオブジェクトです。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3031h	Digital Output 1	ARRAY	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	5	5
01h	コンフィグレーション	ro	なし	0 ~ 7	アプリケーションにより定義
02h	ステート	ro	なし	0, 1	
03h	レベル	ro	なし	0, 1	
04h, 05h	予約領域	-	-	-	

【5】 CANopen オブジェクトディクショナリ

数値定義：

Sub-Index	フィールド	数値	定義
1	コンフィグレーション	0 ～ 7	0：機能の割り当て無し 1：異常有無 / 2：警告有無 3：目標到達（未実装） / 4：ブレーキ状態 5：準備中（未実装） / 6：モータ駆動中 7：360°検出（未実装）
2	ステート機能	0, 1	0：アクティブでない。 1：アクティブ
3	レベル機能	0, 1	0：ポジティブ 1：ネガティブ

■オブジェクト 3032h：デジタル出力 2 (Digital Output 2)

このオブジェクトは、デジタル出力 2 の機能について示します。読み出し専用のオブジェクトです。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3032h	Digital Output 2	ARRAY	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	5	5
01h	コンフィグレーション	ro	なし	0 ～ 7	アプリケーション により定義
02h	ステート	ro	なし	0, 1	
03h	レベル	ro	なし	0, 1	
04h, 05h	予約領域	-	-	-	

数値定義：

Sub-Index	フィールド	数 値	定 義
1	コンフィグレーション	0 ～ 7	0：機能の割り当て無し 1：異常有無 / 2：警告有無 3：目標到達（未実装） / 4：ブレーキ状態 5：準備中（未実装） / 6：モータ駆動中 7：360°検出（未実装）
2	ステート機能	0, 1	0：アクティブでない。 1：アクティブ
3	レベル機能	0, 1	0：ポジティブ 1：ネガティブ

■オブジェクト 3033h：デジタル出力 3 (Digital Output 3)

このオブジェクトは、デジタル出力 3 の機能について示します。読み出し専用のオブジェクトです。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3033h	Digital Output 3	ARRAY	INTEGER16	オプション

【5】 CANopen オブジェクトディクショナリ

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	5	5
01h	コンフィグレーション	ro	なし	0～7	アプリケーション により定義
02h	ステート	ro	なし	0, 1	
03h	レベル	ro	なし	0, 1	
04h, 05h	予約領域	-	-	-	

数値定義：

Sub-Index	フィールド	数 値	定 義
1	コンフィグレーション	0～7	0：機能の割り当て無し 1：異常有無 / 2：警告有無 3：目標到達（未実装） / 4：ブレーキ状態 5：準備中（未実装） / 6：モータ駆動中 7：360°検出（未実装）
2	ステート機能	0, 1	0：アクティブでない。 1：アクティブ
3	レベル機能	0, 1	0：ポジティブ 1：ネガティブ

■オブジェクト 3034h：デジタル出力 4 (Digital Output 4)

このオブジェクトは、デジタル出力 4 の機能について示します。読み出し専用のオブジェクトです。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
3034h	Digital Output 4	ARRAY	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	5	5
01h	コンフィグレーション	ro	なし	0～7	アプリケーション により定義
02h	ステート	ro	なし	0, 1	
03h	レベル	ro	なし	0, 1	
04h, 05h	予約領域	-	-	-	

数値定義：

Sub-Index	フィールド	数 値	定 義
1	コンフィグレーション	0～7	0：機能の割り当て無し 1：異常有無 / 2：警告有無 3：目標到達（未実装） / 4：ブレーキ状態 5：準備中（未実装） / 6：モータ駆動中 7：360°検出（未実装）
2	ステート機能	0, 1	0：アクティブでない。 1：アクティブ
3	レベル機能	0, 1	0：ポジティブ 1：ネガティブ

【5】 CANopen オブジェクトディクショナリ

■オブジェクト 4100h：CAN 通信ステータス (Status Communication CAN)

このオブジェクトは、通信ネットワークのステータスを示しています。

CAN コントローラのレジスタの状態も示しています。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
4100h	Status Communication CAN	ARRAY	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	5	5
01h	TEC レジスタ	ro	なし	0 ～ 255	-
02h	REC レジスタ	ro	なし	0 ～ 255	アプリケーション により定義
03h	実際のエラーフラグ	ro	なし	0 ～ 32767	-
04h	実際の状態機械	ro	なし	0 ～ 32767	
05h	実際のドライバステート	ro	なし	0 ～ 32767	

数値定義：

Sub-Index	フィールド	数 値	定 義
01h	TEC レジスタ	0 ～ 255	エラーカウンタを CAN へ送信
02h	REC レジスタ	0 ～ 255	エラーカウンタを CAN が受信
03h	実際のエラーフラグ	0 ～ 32767	CAN ドライバがエラー発生を表示
04h	CAN 通信の状態機械	0 ～ 7	以下は、CAN 通信の状態機械のステートを示しています。 0：IDLE ステート (最初のステート) 1：INIT ステート (CAN コントローラの初期化) 2：TELL ステート (CAN プロトコルでモータ運転が選択されず、CAN 通信がオブジェクトのリード専用で使用) 3：INACTIVE ステート (CAN ドライバがスリープ状態) (未実装) 4：STATE OK (通信の送受信が正常) 5：WARNING ステート (ネットワークが受動エラーの場合) 6：FAULT ステート (ネットワークが Bus off で、通信エラーにより FAULT ステートへ移行) 7：RESET ステート (未実装)
05h	Actual State Drive	0 ～ 8	全てのビットが CAN コントローラのステートを示します。 ビット 0: INIT ステート - CAN コントローラが初期化 ビット 1: ACTIVE ステート - CAN コントローラが動作 ビット 2: BUSOFF ステート - CAN コントローラが Bus Off ビット 3: PASSIVE ステート - CAN コントローラが受動

Sub-index 3 実際のフラグエラーのビット定義：

ビット	意 味	内 容	備 考
0 ～ 4	予約領域	不使用	-
5	オーバーフローエラー	CAN コントローラのオーバーランエラー	1：データオーバーラン検出 0：データオーバーランなし
6	Tx バッファオーバーフローエラー	バッファのオーバーフロー送信	1：Tx バッファのオーバーフロー 0：Tx バッファのオーバーフローなし
7	Rx バッファオーバーフローエラー	バッファのオーバーフロー受信	1：Rx バッファのオーバーフロー 0：Rx バッファのオーバーフローなし
8	受動エラー	CAN コントローラの受動エラー	1：受動エラー検出 0：受動エラーなし
9	Bus Off エラー	CAN コントローラの Bus Off	1：Bus Off エラー検出 0：Bus Off なし
10 ～ 15	予約領域	不使用	-

【5】 CANopen オブジェクトディクショナリ

■オブジェクト 4101h：CAN 通信カウンタ (Counter Communication CAN)

このオブジェクトは、CAN 通信のカウントを示しています。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
4101h	Counter Communication CAN	ARRAY	INTEGER16	オプション

オブジェクト内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	5	5
01h	Bus Off カウンタ	ro	なし	0 ～ 32767	0
02h	Bus Off カウンタリカバリ	ro	なし	0 ～ 32767	
03h	CAN 通信カウンタリセット	ro	なし	0 ～ 32767	
04h, 05h	フリー	ro	なし	0 ～ 32767	-

エントリ内容：

Sub-Index	フィールド	数 値	定 義
01h	Bus Off カウンタ	0 ～ 32767	Bus Off ステートになった回数
02h	Bus Off カウンタリカバリ	0 ～ 32767	Bus Off ステートを抜けることができた回数
03h	CAN 通信リセットカウンタ	0 ～ 32767	CAN 通信で発生した故障後にリセットした回数

■オブジェクト 4102h：CAN 通信設定 (Settings Communication CAN)

このオブジェクトは、CAN 通信の設定を示しています。

- ・ドライバは、Bus Off ステートの場合、故障 (Fault) のアラームになります。
- ・ドライバは、Bus Off ステートから回復することができます。
- ・ユーザは、通信エラーを管理するため異なる感度を設定することができます。

ドライバは、感度を高く (H) 設定すると、受動エラーでもアラームに移行します。

故障 (Fault) ステートに入るタイムアウトは、Sub-index で設定できます。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
4102h	Settings Communication CAN	ARRAY	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	5	5
01h	CAN エラー感度	rw	なし	0 ～ 32767	1
02h	受動エラータイムアウト	rw	なし	0 ～ 32767	100
03h ～ 05h	フリー	ro	なし	0 ～ 32767	-

数値定義：

Sub-Index	フィールド	数 値	定 義
01h	CAN エラー感度	0、1	0 = H 高感度の故障モード：ドライバは、タイムアウト (4102h：2) 後に受動エラーのアラーム状態に移行します。 1 = L 低感度の故障モード：ドライバは、Bus Off エラーのアラーム状態に移行します。(受動エラーは、ワーニングフラグで読出し可能)
02h	受動エラータイムアウト	0 ～ 32767	受動エラーで故障ステートに入るためのタイムアウト タイムアウトは CAN エラー感度が 0=H に設定されたとき使用されます。 単位：[10ms]

【5】CANopen オブジェクトディクショナリ

■オブジェクト 4103h：CAN 通信エラー（Error Communication CAN）

このオブジェクトは、CAN 通信で発生したエラーを示しています。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
4103h	Error Communication CAN	ARRAY	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	5	5
01h	最新エラーフラグ	rw	なし	0 ～ 32767	0
02h	全エラーフラグ	rw	なし	0 ～ 32767	
03h ～ 05h	フリー	ro	なし	0 ～ 32767	-

数値定義：

Sub-Index	フィールド	数 値	定 義
01h	最新エラーフラグ	0 ～ 32767	発生した最新のエラーフラグを示します。 0x0010：オーバーフローエラー 0x0020：Tx バッファのオーバーフローエラー 0x0040：Rx バッファのオーバーフローエラー 0x0080：受動エラー 0x0100：Bus Off エラー
02h	全エラーフラグ	0 ～ 32767	選択されたビットで発生したエラーを示します。 ビット 4：オーバーフローエラー ビット 5：Tx バッファのオーバーフローエラー ビット 6：Rx バッファのオーバーフローエラー ビット 7：受動エラー ビット 8：Bus Off エラー

■オブジェクト 4200h：アラームモニタリング(Alarm Monitoring)

このオブジェクトは、アラームのモニタリングを示しています。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
4200h	Alarm Monitoring	ARRAY	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	エントリ数	ro	なし	5	5
01h	エラーコード	rw	なし	0 ～ 32767	0
02h	メーカーコード LSB	rw	なし	0 ～ 32767	
03h	メーカーコード MSB	ro	なし	0 ～ 32767	-
04h, 05h	フリー	ro	なし	0 ～ 32767	

【5】 CANopen オブジェクトディクショナリ

■オブジェクト 6402h：モータタイプ(Motor Type)

このオブジェクトは、ドライバにより駆動されるモータタイプを示しています。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
6402h	Motor Type	VAR	UNSIGNED16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	Motor Type	rw	なし	0～0xFFFF	0x000A

数値定義：

Sub-Index	フィールド	定 義
0000h	非標準モータ	-
0001h	DC モータ（位相変調）	
0002h	DC モータ（周波数制御）	
0003h	同期モータ（永久磁石界磁）	
0004h	同期モータ（界磁巻線）	-
0005h	スイッチドリラクタンスモータ	AC 同期スイッチドリラクタンスモータ
0006h	巻線型モータ	AC 巻線モータ
0007h	かご型誘導モータ	AC 誘導モータ
0008h	ステッピングモータ	-
0009h	マイクロステッピングモータ	-
000Ah	ブラシレスモータ（正弦波駆動）	AC 同期ブラシレスモータ
000Bh	ブラシレスモータ（矩形波駆動）	AC 同期ブラシレスモータ
000Ch	AC 同期リラクタンスモータ c	-
000Dh	DC モータ（永久磁石界磁）	
000Eh	DC モータ（直巻）	
000Fh	DC モータ（界磁巻線）	
0010h	DC モータ（複巻）	
0011h～7FFh	予約領域	
7FFh	モータタイプの割当てなし	
8000h～FFFFh	メーカ固有	

■オブジェクト 6403h：モータカタログ番号(Motor Catalogue Number)

このオブジェクトは、モータのサイズを ASCII コードで示しています。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
6403h	Motor Catalogue Number	VAR	文字列	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	Motor Catalogue Number	rw	なし	4	形式に依存

【5】 CANopen オブジェクトディクショナリ

有効エントリ：

数 値	タイプ	数 値	タイプ
'I'	-	'C1'	カスタム 1
'S'	ECO-S	'C2'	カスタム 2
'M'	ECO-M, PRO-M	'C3'	カスタム 3
'L'	PRO-L	'C4'	カスタム 4

■オブジェクト 6404h：モータ製品コード (Motor Manufacturer)

このオブジェクトは、モータの名称を示しています。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
6404h	Motor Manufacturer	VAR	文字列	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	数値範囲	初期値
00h	Motor Manufacturer	rw	なし	4	形式に依存

有効エントリ：

数値・文字列	タイプ	モータ製品コード
'NaN'	なし	未定義
'B40'	ECO-S	B40E4J – C1078
'B63'	ECO-M, PRO-M	B6304K – H32mm – 48Vdc
'B71'	PRO-L	B7108Q – H40mm – 48Vdc

■オブジェクト 6502h：サポートされるドライバモード (Supported Drive Modes)

このオブジェクトは、サポートされるドライバモードを示しています。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
6502h	Supported Drive Modes	VAR	U32	強制

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	Supported Drive Modes	ro	なし	0 ～ 32767	-

有効エントリ：

ビット	内 容	備 考
0	位置プロファイルモード	サポート
1	速度モード	
2	速度プロファイルモード	
3	位置プロファイルモード	
4	予約領域	-
5	ホーミングモード	
6	補間された位置モード	
7	サイクル同期位置モード	
8	サイクル同期速度モード	
9	サイクル同期トルクモード	サポート
10 ～ 15	予約領域	
16	メーカー固有	
17	メーカー固有 – テストモード (予約領域)	
18 ～ 31	メーカー固有	-

【5】 CANopen オブジェクトディクショナリ

5-4 DSP402 プロファイルオブジェクト

■オブジェクト 603Fh：エラーコード(Error Code)

このオブジェクトは、ドライバで発生した最新のエラーのコードを示しています。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
603Fh	Error code	VAR	U16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00Fh	Error code	ro	なし	エマージェンシ表を参照	-

オブジェクト(603Fh)は、発生した最新アラームのエラーコードです。

注)詳細は、2章の「表 2-31 ～ 36 エマージェンシ内容」を参照してください。

■オブジェクト 6040h：コントロールワード(Controlword)

このオブジェクトは、CiA-402 FSA、CiA-402 モードとメーカー固有のエンティティを制御するために使用します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
6040h	Controlword	VAR	UNSIGNED16	強制

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	Controlword	rw	あり (初期値)	下表を参照	-


このオブジェクトは、ビット単位で以下の内容です。

ビット	名 称	内 容
0	スイッチ ON (Switch ON)	各ビット指令が状態機械 DSP402 へ移行
1	電圧有効 (Enable voltage)	
2	急停止 (Quick stop)	
3	運転有効 (Enable operation)	
4～6	運転モード固有 (Operation mode specific)	これらのビットは、プロファイルモードが選択されると異なる意味になります。
7	故障リセット (Fault reset)	故障をリセットし、エラーがなければ再スタートします。(使用不可)
8	中止 (Halt)	0 = 命令中の動作を継続します。 1 = 命令中の動作を中断します。
9	運転モード固有 (Operation mode specific)	このビットは、プロファイルモードが選択されると異なる意味になります。
10	予約領域	-
11	メーカー固有 - ワーニングアック (Manufacturer specific - Warning Ack)	1 に設定するとステータスワードのワーニングビットが削除されます。
12～15	メーカー固有 (Manufacturer specific)	フリー

【5】 CANopen オブジェクトディクショナリ

マスタコントローラは、状態機械 DSP402 のステートへ移行するためにコントロールワードのビット 0～3 と 7 を命令として送信します。命令の定義を次の表に示します。

表 5-2 命令の定義

命 令	コントロールワードのビット					遷 移
	ビット 7	ビット 3	ビット 2	ビット 1	ビット 0	
	Fault Reset	Enable Operation	Quick Stop	Enable Voltage	Switch ON	
Shutdown	0	X	1	1	0	T2, T6, T8
Switch ON	0	0	1	1	1	T3
Switch ON	0	1	1	1	1	T3 注)2
Disable Voltage	0	X	X	0	X	T7, T9, T10, T12
Quick Stop	0	X	0	1	X	T7, T10, T11
Disable Operation	0	0	1	1	1	T5
Enable Operation	0	1	1	1	1	T4, T16
Fault Reset 注)1		X	X	X	X	T15

注) 1. FAULT ステートから抜けるために FAULT リセットが発生 – 使用不可

2. “Switched ON” ステートを実行後、“Enable Operation” へ自動的に遷移 – 使用不可

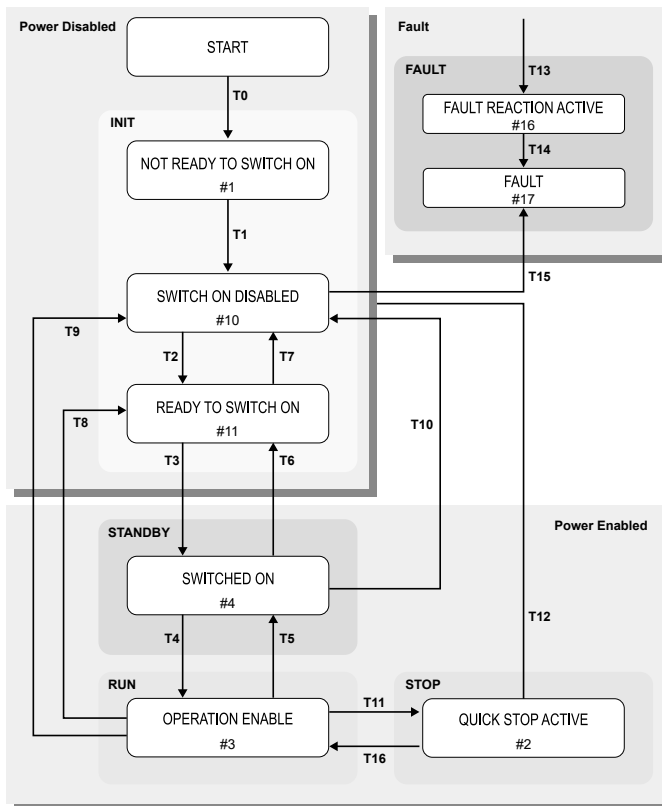


図 5-15 状態機械 DSP402・コントロールワードの遷移

【5】 CANopen オブジェクトディクショナリ

⚠ 注意

遷移命令の送信間隔は、最低 40ms 待つ必要があります。
 状態機械は、“Switched ON” から “Operation Enable” ステートへ 40ms で移行します
 コントロールワードを正確に処理するためには、PDO か SDO を介して、コントロールワードの命令または変更する数値を 40ms 後に送信してください

コントロールワードは、状態機械 DSP402 の遷移に使用されます。

次の表は、マスタコントローラが命令を実行するために、書込む必要があるビットを示しています。

表 5-3 コントロールワードのビット

ステート	遷 移	コントロールワード															
		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Shutdown	T2, T6, T8	×	×	×	×	×	×	×	×	0	×	×	×	×	1	1	0
Switch ON	T3	×	×	×	×	×	×	×	×	0	×	×	×	0	1	1	1
Disable Voltage	T7, T9, T10, T12	×	×	×	×	×	×	×	×	0	×	×	×	×	×	0	×
Quick Stop	T7, T10, T11	×	×	×	×	×	×	×	×	0	×	×	×	×	0	1	×
Disable Operation	T5	×	×	×	×	×	×	×	×	0	×	×	×	0	1	1	1
Enable Operation	T4, T16	×	×	×	×	×	×	×	×	0	×	×	×	1	1	1	1
Fault Reset	T15	×	×	×	×	×	×	×	×	↑	×	×	×	×	×	×	×

表 5-4 状態機械 DSP402 の遷移

スタートステート		遷 移	命 令	ゴールステート	
INT	Switch ON Disable (#10)	T2	SHUTDOWN	Ready to Switch ON (#11)	INIT
	Ready to Switch ON (#11)	T3	SWITCH ON	Switch ON (#4)	STANDBY
		T7	QUICK STOP	Switch ON Disable (#10)	INIT
STANDBY	Switch ON (#4)	T4	ENABLE OPERATION	Operation Enabled (#3)	RUN
		T6	SHUTDOWN	Ready to Switch ON (#11)	INIT
		T10	DISABLE VOLTAGE または QUICK STOP	Switch ON Disable (#10)	INIT
RUN	Operation Enabled (#3)	T5	DISABLE OPERATION	Switch ON (#4)	STANDBY
		T8	SHUTDOWN	Ready to Switch ON (#11)	INIT
		T9	DISABLE VOLTAGE	Switch ON Disable (#10)	INIT
		T11	QUICK STOP	Quick Stop Active (#2)	STOP
STOP	Quick Stop Active (#2)	T16	ENABLE OPERATION	Operation Enabled (#3)	RUN
		T12	DISABLE VOLTAGE	Switch ON Disable (#10)	INIT

【5】CANopen オブジェクトディクショナリ

■オブジェクト 6041h：ステータスワード (Statusword)

このオブジェクトは、現在のドライバ状態、運転モード等を示すのに使用します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
6041h	Statusword	VAR	UNSIGNED16	強制

エントリ内容：

Sub-Index	Description	内 容	アクセス	PDO マッピング	データ範囲
00h	Statusword	ro	あり (初期値)	下表を参照	-

このオブジェクトは、ビット単位で示されて各ビットは、以下の内容です。

ビット	名 称	内 容
0	スイッチ ON 準備完了 (Ready to Switch ON)	状態機械 DSP402 のステータスワードのビット
1	スイッチ ON (Switched ON)	
2	運転有効 (Operation Enabled)	
3	故障 (Fault)	
4	電圧有効 (Voltage Enabled)	0：バス (Vbus) 電圧が不足電圧の値より小さい。 1：バス (Vbus) 電圧が不足電圧の値より大きい。
5	急停止 (Quick Stop)	状態機械 DSP402 のステータスワードのビット 急停止機能の状態に対応しています。
6	スイッチ ON 無効 (Switch ON Disabled)	状態機械 DSP402 のステータスワードのビット
7	ワーニング (Warning)	0：ワーニングなし (エラーまたは故障なし) 1：ワーニング発生 (ワーニングを確認するには、 オブジェクト 2003h のワーニングリスト参照)
8	エマージェンシ入力有効 (Emergency Input Enable - メーカ固有)	0：エマージェンシ機能が無効の入力 1：エマージェンシ機能が有効の入力
9	リモート	1：コントロールワードを処理
10	目標到達	0：設定値に未到達 1：設定値に到達
11	内部リミット動作	0：I ² T リミットが動作していないことを表示 1：I ² T リミット動作を表示
12	運転モード固有 1	・位置プロファイルモード：予約領域 ・速度プロファイルモード：速度 ・トルクプロファイルモード：予約領域 ・ホーミングモード：予約領域
13	運転モード固有 2	・位置プロファイルモード：予約領域 ・速度プロファイルモード：予約領域 ・トルクプロファイルモード：予約領域 ・ホーミングモード：予約領域
14	ドライバセーフティ	0：セーフティ状態でない。 1：セーフティ状態
15	ドライバ故障 (Fault) / 運転準備完了	Obj.3100h, Sub.0 = 0 の場合 (初期値) 0：ドライバの異常なし 1：ドライバの異常あり (異常検出中) Obj.3100h, Sub.0 = 1 の場合 0：運転準備が未完了 1：運転準備完了

【5】 CANopen オブジェクトディクショナリ

マスタコントローラは、状態機械 DSP402 のどのステートに遷移したか把握するために、ステータスワード(ビット 0、1、2、3、5、6)を読み込むことができます。命令の意味を次の表に示します。

表 5-3 命令の意味

ステート		ステート 数	ステータスワード															
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INIT	Not ready to Switch ON	#1	×	×	×	×	×	×	×	×	×	0	×	×	0	0	0	0
INIT	Switch ON disabled	#10	×	×	×	×	×	×	×	×	×	1	×	×	0	0	0	1
INIT	Ready to switch ON	#11	×	×	×	×	×	×	×	×	×	0	1	×	0	0	0	1
STANDBY	Switched ON	#4	×	×	×	×	×	×	×	×	×	0	1	×	0	0	1	1
RUN	Operation enabled	#3	×	×	×	×	×	×	×	×	×	0	1	×	0	1	1	1
STOP	Quick stop active	#2	×	×	×	×	×	×	×	×	×	0	0	×	0	1	1	1
FAULT	Fault reaction active	#16	×	×	×	×	×	×	×	×	×	0	×	×	1	1	1	1
FAULT	Fault	#17	×	×	×	×	×	×	×	×	×	0	×	×	1	0	0	0

■オブジェクト 6007h：アバート接続オプションコード (Abort Connection Option Code)

このオブジェクトは、以下のイベントが発生した場合に、どのような動作を行うかを示しています。
Bus-off、ハートビート喪失、ライフガーディング喪失、NMT 停止ステートの入力、通信リセット、ハードウェアリセット
読み出し専用のオブジェクトです。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
6007h	Abort Connection Option Code	VAR	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	Abort Connection Option Code	ro	なし	下表を参照	-

数値定義：

数 値	内 容	備 考
-1	メーカー固有： ・Bus Off：故障 (Fault) 信号、故障ステートへ移行 ・ハートビート喪失：動作なし。マスタコントローラは、ハートビートメッセージが失われた場合、再度設定する必要があります。 ・ライフガーディング喪失：故障信号、オブジェクト (100Dh) のライフタイムファクタにより記載されたメッセージの損失数の後で、ドライバは、故障ステートへ移行。 ・NMT 停止ステート入力：動作なし ・ハードウェアリセット：動作なし ・通信のリセット：動作なし	使用可
0	動作なし	-
1	故障 (Fault) 信号	
2	電圧無効指令	
3	急停止指令	

【5】CANopen オブジェクトディクショナリ

■オブジェクト 605Ah：急停止オプションコード(Quick Stop Option Code)

このオブジェクトは、急停止機能が実行された場合に、どのような動作を行うかを示しています。
“Operation Enabled” から “Quick Stop Active” への遷移の動作を示しています。
読み出し専用のオブジェクトです。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
605Ah	Quick Stop Option Code	VAR	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	Quick Stop Option Code	ro	なし	下表を参照	-

数値定義：

数 値	内 容	備 考
-1	メーカー固有：瞬時停止	-
0	ドライバ機能無効	使用可
1	ランプ状の減速、“Switch ON Disabled”へ遷移	
2	ランプ状の急停止での減速、“Switch ON Disabled”へ遷移	
3	電流リミットで減速、“Switch ON Disabled”へ遷移	
4	電圧リミットで減速、“Switch ON Disabled”へ遷移	
5	ランプ状の減速、“Quick Stop Active”のまま	
6	ランプ状の急停止での減速、“Quick Stop Active”のまま	
7	電流リミットで減速、“Quick Stop Active”のまま	
8	電圧リミットで減速、“Quick Stop Active”のまま	-

■オブジェクト 605Bh：シャットダウンオプションコード(Shutdown Option Code)

このオブジェクトは、“Operation Enabled” ステートから “Ready to Switch ON” ステートへ遷移する場合、どのような動作を行うかを示しています。読み出し専用のオブジェクトです。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
605Bh	Shutdown Option Code	VAR	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	Shutdown Option Code	ro	なし	下表を参照	-

数値定義：

数 値	内 容	備 考
-32768 ～ -1	メーカー固有	-
1	ドライバ機能無効、即座に停止	使用可
2	ランプ状の減速：ドライバ機能無効	-

【5】 CANopen オブジェクトディクショナリ

■オブジェクト 605Ch：運転無効オプションコード(Disable Operation Option Code)

このオブジェクトは、“Operation Enabled” ステートから “Switched ON” ステートへ移行する場合にどのような動作を行うのかを示しています。読み出し専用のオブジェクトです。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
605Ch	Disable Operation Option Code	VAR	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	Disable Operation Option Code	ro	なし	下表を参照	-

数値定義：

数 値	内 容	備 考
-32768 ～ -1	メーカー固有	-
1	ドライバ機能無効、即座に停止	使用可
2	ランプ状の減速：ドライバ機能無効	

■オブジェクト 605Dh：中止オプションコード(Halt Option Code)

このオブジェクトは、中止(Halt)機能が実行された場合に、どのような動作が行われるかを示しています。読み出し専用のオブジェクトです。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
605Dh	Halt Option Code	VAR	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	Halt Option Code	ro	なし	下表を参照	-

数値定義：

数 値	内 容	備 考
-1	メーカー固有：即座に停止	使用可
1	ランプ状の減速、運転有効を保持	
2	ランプ状の急停止での減速、運転有効を保持	
3	電流リミットで減速、運転有効を保持	
4	電圧リミットで減速、運転有効を保持	

■オブジェクト 605Eh：異常処理オプションコード(Fault Reaction Option Code)

このオブジェクトは、ドライバが異常を検出した後の動作を示しています。読み出し専用です。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
605Eh	Fault Reaction Option Code	VAR	INTEGER16	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	Fault Reaction Option Code	ro	なし	次表を参照	-

【5】 CANopen オブジェクトディクショナリ

数値定義：

数 値	内 容	備 考
-32768 ～ -1	メーカー固有	-
0	ドライバ機能無効、モータはフリーラン停止	使用可
1	ランプ状の減速	
2	ランプ状に急停止する減速	-
3	電流リミットの減速	
4	電圧リミットの減速	

■オブジェクト 6060h：運転モード (Modes of Operation)

このオブジェクトにより運転モードを選択できます。

設定された運転モードは、オブジェクト (6061h) 「運転モードディスプレイ」で確認できます。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
6060h	Modes of Operation	VAR	INTEGER8	強制

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	Modes of Operation	rw	あり (初期値)	下表を参照	0

数値定義：

数 値	内 容	備 考
0	モード変更なし / モード割付けなし	-
1	位置プロファイルモード	
2	速度モード	使用可
3	速度プロファイルモード	
4	トルクプロファイルモード	-
5	予約領域	
6	ホーミングモード	
7	補間された位置モード	
8	サイクル同期位置モード	
9	サイクル同期速度モード	
10	サイクル同期トルクモード	
-1	メーカー固有 1 (アナログまたはハードウェア制御モード)	
-2	メーカー固有 2 (予約領域：テスト用)	
-3	メーカー固有 3 (Modbus 制御モード)	

- ・メーカー固有 1：アナログかハードウェア制御に予約されています。
- ・メーカー固有 2：メーカー用 (CANopen 自動速度テスト) に予約されています。
- ・メーカー固有 3：Modbus 制御モード用に予約されています。

ドライバは、次のアボートコードを送信することがあります。

- ・0x060B0002 = ドライバがトルクを発生しているため書込み不可
("Operation Enabled" または "Quick Stop Active" ステートのため)
- ・0x05040001 = サポートされていないモードのため設定不可

⚠ 注 意

メーカー固有プロファイル (アナログモードとテストモード) を設定することはできません。

【5】 CANopen オブジェクトディクショナリ

■オブジェクト 6061h：運転モード表示 (Modes of Operation Display)

このオブジェクトは、設定された運転モードを示しています。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
6061h	Modes of Operation Display	VAR	INTEGER8	強制

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	Modes of Operation Display	ro	あり(初期値)	下表を参照	0

数値定義：

数 値	内 容
0	モード変更なし / モード割付けなし
1	位置プロファイルモード
2	速度モード
3	速度プロファイルモード
4	トルクプロファイルモード
5	予約領域
6	ホーミングモード
7	補間された位置モード
8	サイクル同期位置モード
9	サイクル同期速度モード
10	サイクル同期トルクモード
-1	メーカー固有 (アナログまたはハードウェア制御モード)
-2	メーカー固有 (予約領域：テスト用)
-3	メーカー固有 (Modbus 制御モード)

■オブジェクト 607Eh：極性 (Polarity)

このオブジェクトは、指令値の符号を設定します。

速度／トルクプロファイルの場合、ビット 6 が 0 の場合はそのまま、1 の場合は指令値× -1 の値を使用します。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
607Eh	Polarity	VAR	UNSIGNED8	強制

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	Polarity	rw	なし	0 ～ 192	0

数値定義：

数 値	内 容
0 ～ 5	予約領域
6	速度 / トルク極性
7	位置極性

【5】 CANopen オブジェクトディクショナリ

ドライバは、次のアボートコードを送信することがあります。

・ 0x08000002 = 書き込み無効(アプリケーションにより定義されるオプション)

このオブジェクトは、EEPROM を変更してセーブすることができます。

! EEPROM への保存

- ドライバが“Operational Enabled”または“Quick Stop Active”のときはEEPROMへ保存できません。
- SDO で本オブジェクト(607Eh)に新しいノード ID を書き込む。
- 「パラメータストア」(1010h)に“save”を書き込む。(処理時間に注意)
- NMT ノードリセット

■オブジェクト 60FDh: デジタル入力 (Digital Inputs)

このオブジェクトは、デジタル入力を示しています。

ローワードは、CANopen 402 プロファイルで定義されたデジタル入力のステートを含みます。

ハイワードは、全てのデジタル入力のステートを表しています。

デジタル入力のステートは、オブジェクト(60FDh)によって出力されます。

- ・ ホーミングプロファイルのリミットまたは指令スイッチ (未実装)
- ・ デジタル入力 1 ～ 4: プログラマブルでアプリケーションにより定義
- ・ セーフトルクオフ (STO)

オブジェクト内容:

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
60FDh	Digital Inputs	VAR	UNSIGNED32	オプション

エントリ内容:

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲
00h	Digital Inputs	ro	なし	0 ～ 0xFFFFFFF

ビット構造:

MSB ビット						LSB ビット	
31	16	15	4	3	2	1	0
デジタル入力ステータス (メカ固有)		予約領域		インター ロック	ホーム スイッチ	＋リミット スイッチ	－リミット スイッチ

データ内容:

ビット	コンフィグ レーション	数 値	定 義	備 考
0	－リミットスイッチ	0b 1b	－リミットスイッチに未到達 －リミットスイッチに到達	“DCW” 機能が設定されていれば、 デジタル入力のステートのビット
1	＋リミットスイッチ	0b 1b	＋リミットスイッチに未到達 ＋リミットスイッチに到達	“DCCW” 機能が設定されていれば、 デジタル入力のステートのビット
2	ホームスイッチ	0b 1b	ホームスイッチに未到達 ホームスイッチに到達	“HOME” 機能が設定されていれば、 デジタル入力のステートのビット

【5】 CANopen オブジェクトディクショナリ

ビット	コンフィグレーション	数 値	定 義	備 考
3	インターロック	0b 1b	インターロック動作なし インターロック動作あり	未使用
4～15	予約領域	-	-	-
16	デジタル入力 DigIn1	0b 1b	リードステータス：L レベル リードステータス：H レベル	機能の設定による
17	デジタル入力 DigIn2	0b 1b	リードステータス：L レベル リードステータス：H レベル	
18	デジタル入力 DigIn3	0b 1b	リードステータス：L レベル リードステータス：H レベル	
19	デジタル入力 DigIn4	0b 1b	リードステータス：L レベル リードステータス：H レベル	
20	デジタル入力 STO1	0b 1b	リードステータス：L レベル リードステータス：H レベル	デジタル入力に STO 回路に接続済
21	デジタル入力 STO2	0b 1b	リードステータス：L レベル リードステータス：H レベル	デジタル入力の接続なし 常に H レベル
20～31	デジタル入力	-	-	使用不可

■オブジェクト 60FEh：デジタル出力 (Digital Outputs)

このオブジェクトは、デジタル出力指令を示しています。

オブジェクト内容：

Index	EDS 名称	オブジェクトコード	データタイプ	カテゴリ
60FEh	Digital Outputs	ARRAY	U32	オプション

エントリ内容：

Sub-Index	内 容	アクセス	PDO マッピング	データ範囲	初期値
00h	サポートされた最大の Sub-index	c	なし	1、2	2
01h	物理的输出	ro	可能	0	00000000h
02h	ビットマスク	ro	なし	0	00000000h

Sub-index 01h のビット構造

MSB ビット		LSB ビット
31	16 15	1 0
デジタル出力指令 (メーカ固有)	予約領域	モータ ブレーキ指令

Sub-index 01h の数値定義

ビット	コンフィグレーション	数 値	定 義	備 考
0	モータブレーキ指令	0b 1b	ブレーキ動作 → モータ軸ロック ブレーキ解放 → モータ軸フリー	ブレーキの手動モードで 使用可
1～15	予約領域 (各ビット)	-	予約領域	-
16	デジタル出力 1 ドライバ状態	0b 1b	スイッチ OFF：ドライバ故障 スイッチ ON：ドライバ OK	ドライバの状態
17	デジタル出力 2	0b 1b	スイッチ OFF：警告なし スイッチ ON：警告あり	警告の有無
18	デジタル出力 3	0b 1b	スイッチ OFF：モータ停止 スイッチ ON：モータ駆動	モータの状態
19	デジタル出力 4	0b 1b	スイッチ OFF：ブレーキ動作 スイッチ ON：ブレーキ解放	ブレーキの状態

【5】 CANopen オブジェクトディクショナリ

Sub-index 02h のビット構造

MSB ビット		LSB ビット	
31	16 15	1	0
デジタル出力 有効 / 無効 (メーカー固有)		予約領域	モータブレーキ マネジメント

Sub-index 02h の数値定義

ビット	コンフィグレーション	数 値	定 義	備 考
0	モータブレーキマネジメント	0b 1b	出力無効 出力有効	常に有効
1 ~ 15	予約領域	-	予約領域	-
16	デジタル出力 1 有効	0b 1b	出力無効 出力有効	常に有効
17	デジタル出力 2 有効	0b 1b	出力無効 出力有効	
18	デジタル出力 3 有効	0b 1b	出力無効 出力有効	
19	デジタル出力 4 有効	0b 1b	出力無効 出力有効	

注) Sub-index 2 は読出し専用で、出力は常に有効です。

【6】 CANopen 運転モード

6-1 運転モード

⚠ 注意

運転モードは、オブジェクト (6060h) で選択され、ドライバが “Operation Enabled” ステータスでない場合に変更が実行されます。

運転モードを以下に示します。

・位置プロファイル (未実装)

このモードでは、定義された目標位置へ移動することが可能です。
オブジェクト (6060h) 「運転モード」に 1 を設定してください。

・速度プロファイル

このモードでは、目標速度の定義なしで、速度の設定ポイントに追従することができます。
オブジェクト (6060h) 「運転モード」に 3 を設定してください。

・トルクプロファイル

このモードでは、目標トルクの定義なしで電流の設定ポイントに追従することができます。
オブジェクト (6060h) 「運転モード」に 4 を設定してください。

6-2 位置プロファイルモード

■オブジェクト 6064h：位置検出値 (Position Actual Value)

回転検出器の現在値を示し、電源 ON で値は 0 にリセットされます。
このオブジェクトは、速度プロファイル、トルクプロファイルにおいても使用可能です。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
6064h	VAR	INTEGER32	強制

エントリ内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	ro	あり	-	なし	inc

注) 位置プロファイルモードは、未実装です。

【6】 CANopen 運転モード

6-3 速度プロファイルモード

速度プロファイルモード(PV)では、マスタコントローラから送信される目標速度指令に従って、PIDコントローラにより速度を制御します。

過負荷時を除きドライバは、目標速度値に従って電圧および周波数を出力します。

速度プロファイルモードを動作させるために、事前に必要な設定を以下に示します。

- ・速度プロファイルモードは、「運転モード」(6060h)に3を設定します。
- ・状態機械 DSP402 の“Operation Enabled”ステートであることを、「ステータスワード」(6041h)で確認します。

状態機械の遷移は、コントロールワード(6040h)を使用します。

目標速度は、オブジェクトディクショナリの「目標速度」(60FFh)で設定します。

ドライバは、加減速ランプおよび最大速度を考慮して、設定された目標速度まで加速(減速)します。

■速度プロファイルモードの制御構成

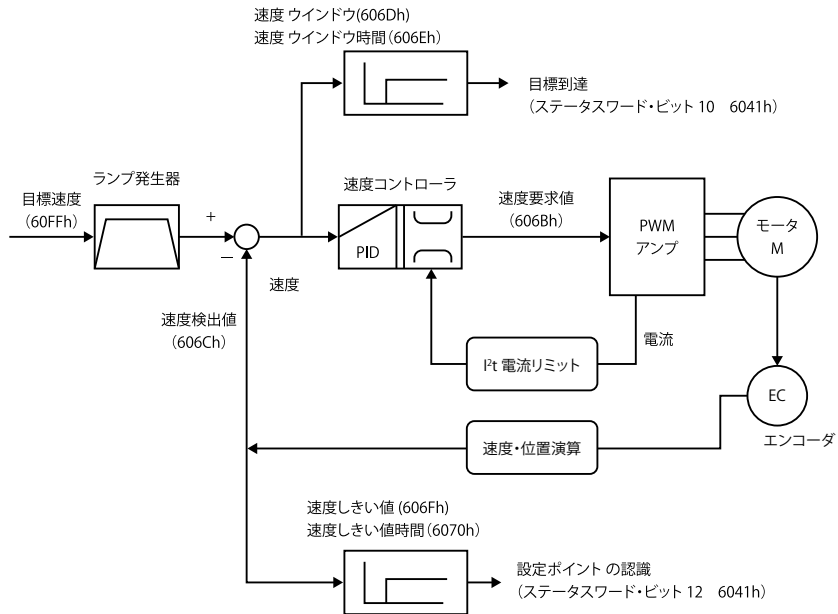


図 6-1 速度プロファイルモードの制御構成

【6】 CANopen 運転モード

■オブジェクトリスト

速度プロファイルモード使用時に関連するオブジェクトを以下に示します。

表 6-1 速度プロファイルのオブジェクト

Index	Sub-Index	名 称	リード/ライト	M/O	データタイプ	PDO	使用可
0x603F	0	エラーコード	RO	O	U16	-	○
0x6040	0	コントロールワード	R/W	M	U16	RPDO	○
0x6041	0	ステータスワード	RO	M	U16	TPDO	○
0x6060	0	運転モード	R/W	M	I8	RPDO	○
0x6061	0	運転表示モード	RO	M	I8	TPDO	○
0x60FF	0	目標速度	R/W	M	I32	RPDO	○
0x607F	0	最大速度プロファイル	R/W	O	U32	-	○
0x6080	0	最高モータ速度	R/W	O	U32	-	-
0x6083	0	加速プロファイル	R/W	O	U32	-	○
0x6084	0	減速プロファイル	R/W	O	U32	-	○
0x60C5	0	最大加速	R/W	O	U32	-	○
0x60C6	0	最大減速	R/W	O	U32	-	○
0x607E	0	極性	R/W	O	U8	-	○
0x606B	0	速度要求値	RO	O	I16	-	○
0x606C	0	速度検出値	RO	M	I32	TPDO	○
0x606D	0	速度ウインドウ	R/W	O	U16	-	○
0x606E	0	速度ウインドウ時間	R/W	O	U16	-	○
0x606F	0	速度スレッシュホールド	R/W	O	U16	-	○
0x6070	0	速度スレッシュホールド時間	R/W	O	U16	-	○
0x3300	0	速度フルスケール	R/W	O	U16	-	○
0x6086	0	運転プロファイルタイプ	R/W	O	I16	-	-
0x60E0	0	ポジティブトルクリミット値	R/W	O	U16	-	○
0x60E1	0	ネガティブトルクリミット値	R/W	O	U16	-	○
0x6085	0	急停止減速	R/W	O	I16	-	-
0x2053	0	速度フィルタ	RO	O	I16	TPDO	○

速度プロファイルモードを開始させるために、必要な設定を以下に示します。

- ・オブジェクト (6060h)「運転モード」に 3 を設定することで、運転モードは速度プロファイルモードへ切替わります。
- ・状態機械 DSP402 の “Operation Enabled” ステートへ移行するために、コントロールワード (6040h) を使用します。
- ・オブジェクト (6083h)「加速プロファイル」に加速度、オブジェクト (6084h)「減速プロファイル」に減速度をそれぞれ設定します。
- ・オブジェクト (60FFh)「目標速度」に目標速度を設定することにより、運転を開始します。

必要に応じて、オブジェクト (6040h)「コントロールワード」のビット 8 を 0 にクリアすることで運転を開始することができます。

ドライバを運転中に「目標速度」(60FFh)の値を変更することができ、変更後は、オブジェクト (6083h)と(6084h)で定義された加減速度に従って即座に追従します。

【6】 CANopen 運転モード

次の図は、速度プロファイルのブロック図を示します。

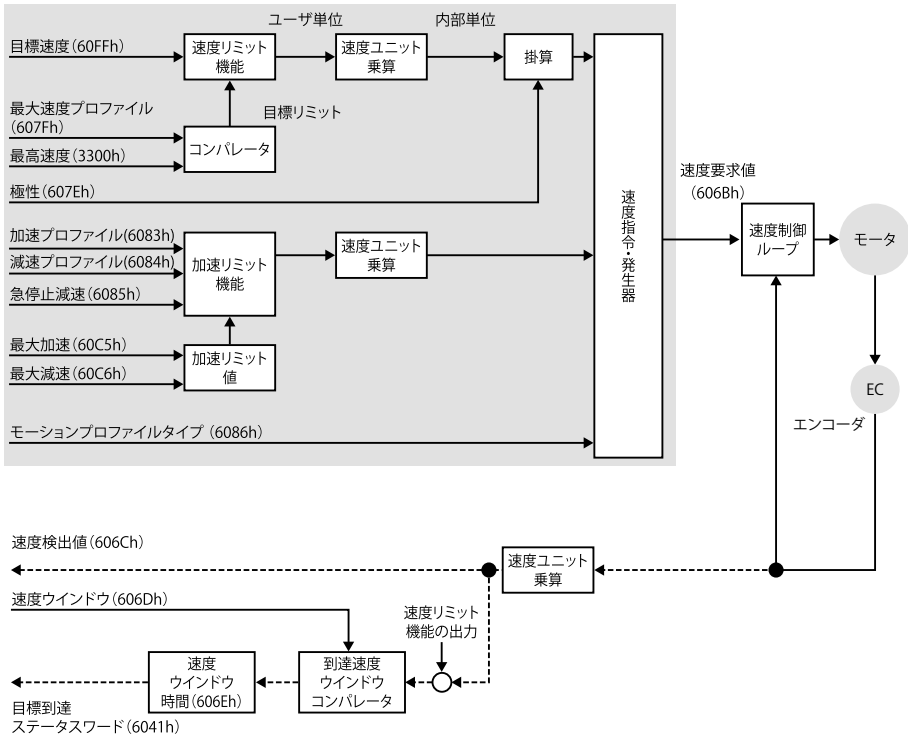


図 6-2 速度制御の構成

次のいずれかの条件を満たした場合、運転を終了します。

- ・「目標速度」(60FFh) に 0 を設定 (モータに常時トルクが発生しています。)
- ・コントロールワード (6040h) のビット 8 (Halt : 中止) を 1 に設定して停止
- ・エラー発生による停止 (ドライバは Fault ステートへ移行)
- ・コントロールワード (6040h) で “Disable Operation”、 “Disable Voltage”、 “Quick Stop” のいずれかへ遷移する命令によって “Operation Enabled” ステートを抜けたために停止
- ・セーフティ状態 (STO 入力) による停止 (モータがフリーラン停止します。)

以下のオブジェクトまたはビットでモニタすることも可能です。

- ・オブジェクト「速度検出値」(606Ch)
- ・オブジェクト「速度ウインドウ」(606Dh) → 目標到達：「ステータスワード」(6041h) のビット 10
- ・オブジェクト「速度スレッシュホールド」(606Fh) → 速度：「ステータスワード」(6041h) のビット 12

【6】 CANopen 運転モード

■コントロールワードビット

オブジェクト (6040h) 「コントロールワード」のビット 8 は、マスタコントローラによって設定可能です。

表 6-2 速度プロファイルのコントロールワードビット

ビット	数 値	定 義
ビット 8 = 中止 (Halt)	0b	運転の実行または継続
	1b	中止命令により運転は停止します。オプションコード (605Dh) 注)

注) オプションコード (605Dh) 未実装

■ステータスワードビット

オブジェクト (6041h) 「ステータスワード」の以下のビットは、マスタコントローラによって設定可能です。

表 6-3 速度プロファイルのステータスワードビット

ビット	数 値	定 義
ビット 10 = 目標到達	0b	Halt (コントロールワードのビット 8) = 0 : 目標速度に未到達 Halt (コントロールワードのビット 8) = 1 : 減速を開始
	1b	Halt (コントロールワードのビット 8) = 0 : 目標速度に到達 Halt (コントロールワードのビット 8) = 1 : 速度 0
ビット 12 = 速度	0b	速度が速度スレッシュホールド (606Fh) より小さい。
	1b	速度が速度スレッシュホールド (606Fh) より大きい。

⚠ 注 意

モータ出力トルクは、オブジェクト (60E0h) 「ポジティブトルクリミット値」とオブジェクト (60E1h) 「ネガティブトルクリミット値」により制限することも可能です。

【6】CANopen 運転モード

■運転モードの内容

速度プロファイルモードの運転では、設定した目標速度に従って動作します。

手順：

- ・速度プロファイルモードは、「運転モード」(6060h)に3を設定します。
- ・「加速プロファイル」(6083h)と「減速プロファイル」(6084h)へ加減速ランプ(ユーザ単位)の値を設定します。
- ・目標速度(ユーザ単位)を「目標速度」(60FFh)に0を設定します。
- ・運転モードを開始するために「コントロールワード」(6040h)を設定します。
- ・「目標速度」(60FFh)で目標速度を設定します。

新しく設定した目標速度は、ドライバが運転中であれば即座に反映されて動作します。

または、「コントロールワード」(6040h)のビット8 (Halt) が1の運転停止状態でも、目標速度の変更が反映され、0を設定することにより新しく設定した速度で動作します。

■プロファイルタイプ

プロファイルタイプは、「運転プロファイルタイプ」(6086h)により設定することができます。

- ・直線ランプ：設定値“0”(初期値)

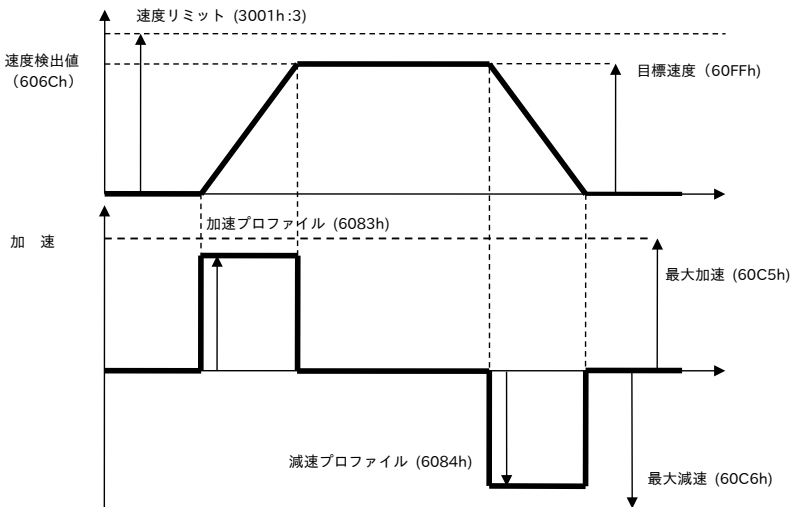


図 6-3 プロファイルタイプ：0 直線ランプ(台形)

- ・ Sin^2 ランプ：設定値“1”(使用不可)
- ・ジャークフリー (Jerk-free) ランプ：設定値“2”(使用不可)
- ・ジャークリミテッド (Jerk limited) ランプ：設定値“3”(使用不可)

【6】 CANopen 運転モード

■出力クエリ

- ・ドライバのステータスが返されるオブジェクト：「ステータスワード」(6041h)
ドライバがリセットされると値は、0 となります。
- ・速度指令(ユーザ単位)が返されるオブジェクト：「速度要求値」(606Bh)
- ・実際の速度(ユーザ単位)が返されるオブジェクト：「速度検出値」(606Ch)

■ウィンドウ停止

- ・速度ウィンドウ(ユーザ単位)の値は、「速度ウィンドウ」(606Dh)へ返されます。
目標速度に対して速度検出値の許容範囲を設定可能です。
「速度ウィンドウ」(606Dh)は、「目標速度」(60FFh)と「速度検出値」(606Ch)を比較して、判定が行われます。
「速度ウィンドウ時間」(606Eh)で設定した時間を経過後も「速度ウィンドウ」(606Dh)の範囲内の場合、「ステータスワード」(6041h)のビット 10「目標到達」が 1 がセットされます。

(1) 速度プロファイルモード (Halt Bit = 0)

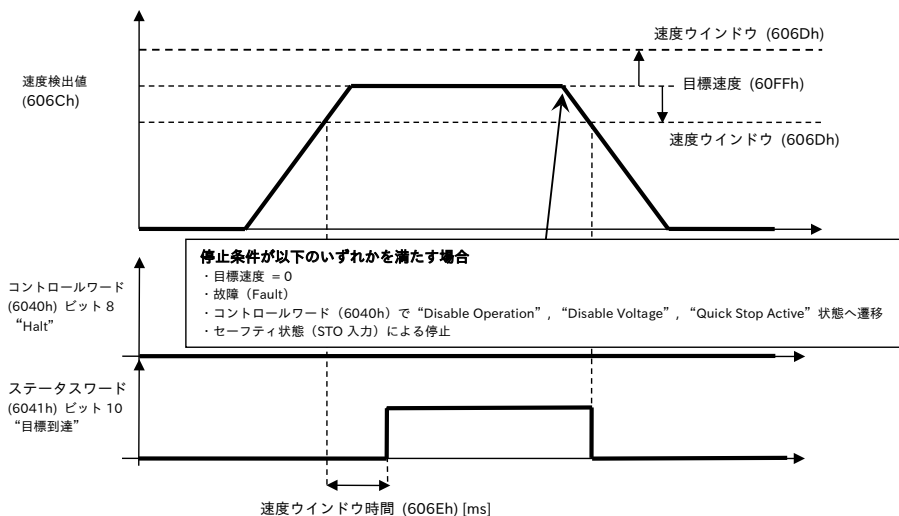


図 6-4 速度ウィンドウ (Halt ビット = 0)

【6】CANopen 運転モード

(2) 速度プロファイルモード (Halt ビット = 1)

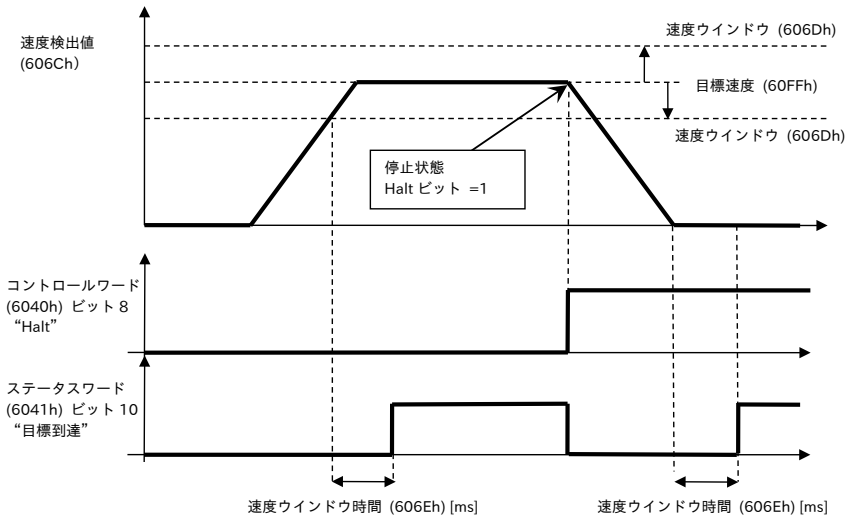


図 6-5 速度ウィンドウ (Halt ビット = 1)

- ・「速度スレッシュホールド」(606Fh)は、ウィンドウ停止を設定します。

「速度検出値」(606Ch)が「速度スレッシュホールド」(606Fh)の値を超えた状態で、「速度スレッシュホールド時間」(6070h)を経過した場合、「ステータスワード」(6041h)のビット 12「速度」が 0 にクリアされます。

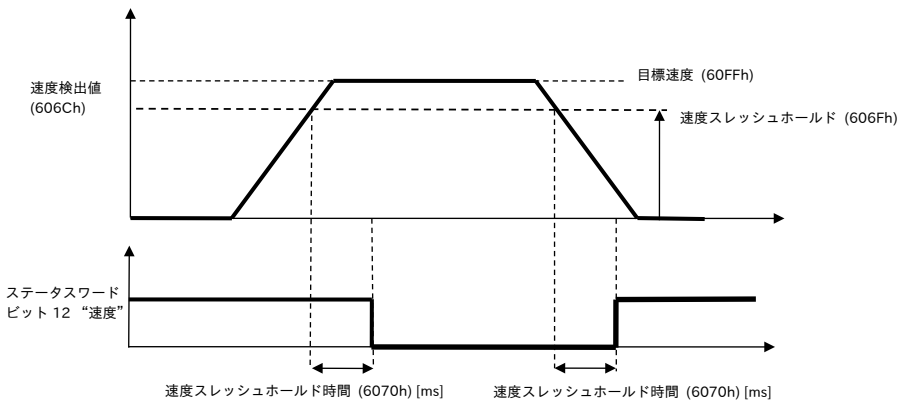


図 6-6 速度スレッシュホールド

【6】 CANopen 運転モード

6-4 速度プロファイルのオブジェクト

■オブジェクト 60FFh：目標速度(Target Velocity)

このオブジェクトは、速度プロファイルモードで速度指令発生器の入力として使用され、目標速度(速度指令)を設定して使用します。

「加速プロファイル」(6083h)と「減速プロファイル」(6084h)で設定した加減速度に従って、速度を制御します。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
60FFh	VAR	INTEGER32	強制

エントリ内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単位
00h	rw	あり	-2147483647 ～ 2147483647	メーカー固有	u.u.

ドライバは、次のアボートコードを送信することがあります。

- 0x06090031：(3001h: 3)の設定値を超えたパラメータ値の書込み
「速度リミット」(3001h: 3)より小さく設定する必要があります。
- 0x06040030：設定可能な範囲を超えたパラメータ値の書込み

注意

モータの最低回転数(速度)は 0.25 rpm です。

■オブジェクト 607Fh：最大速度プロファイル(Max Profile Velocity)

このオブジェクトは、最大速度を示しています。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
607Fh	VAR	UNSIGNED32	オプション

エントリ内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単位
00h	rw	なし	1 ～ 2147483647	メーカー固有	rpm

速度の単位は、rpm です。

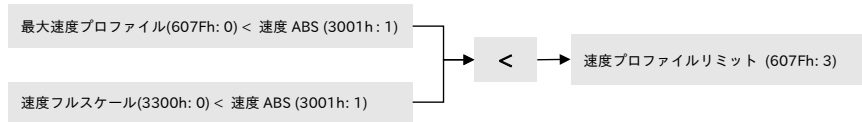
最大速度プロファイルは、「速度最大絶対値」(3001h: 1)より小さく設定する必要があります。

注意

このオブジェクトは、その他のオブジェクトによって設定値が制限されることがあります。設定値が正しくなければアボートコードが送信され、初期化中に数値が正しくなければエマージェンシプロトコルのエラーメッセージ(0x8B17h)が送信されるので、内容を確認してください。

【6】CANopen 運転モード

「最大速度プロファイル」(607Fh: 0) と「速度フルスケール」(3300h: 0) は、共に速度リミットを定義します。リミット設定の図式次に示します。



ドライバは、次のアボートコードを送信することがあります。

- 0x06090031：(3001h: 1)の設定値を超えたパラメータ値の書き込み
「速度 ABS」(3001h: 1)より小さく設定する必要があります。
- 0x06040030：設定可能な範囲を超えたパラメータ値の書き込み

このオブジェクトは、EEPROM を変更してセーブすることができます。

! EEPROM への保存	
●	ドライバが “Operational Enabled” または “Quick Stop Active” でないこと
●	「最大速度プロファイル」(607Fh)に新しい値を書込む
●	「パラメータストア」(1010h)に “save” を書き込み、完了するまで待機(処理時間に注意)
●	NMT ノードリセット

注)このオブジェクトは、位置プロファイルでも使用されます。

■オブジェクト 6086h：運転プロファイルタイプ(Motion Profile Type)

このオブジェクトは、運転を実行するのに設定された運転プロファイルを示しています。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
6086h	VAR	INTEGER16	オプション

エントリ内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	rw	あり	-32768 ～ 32767	0	[-]

数値定義：

数 値	定 義	備 考
-32768 ～ -1	メーカ固有	未実装
0	リニアランプ (台形プロファイル)	初期値
1	Sin^2 ランプ	未実装
2	ジャークフリー (Jerk-free) ランプ	
3	ジャークリミテッド (Jerk-limited) ランプ	
4 ～ 32767	予約領域	-

ドライバは、次のアボートコードを送信することがあります。

- 0x08000024：データ使用不可(数値が 0 でなければ、ドライバがアボートコードを送信します)

【6】 CANopen 運転モード

⚠ 注意

このオブジェクトは、その他のオブジェクトによって設定値が制限されることがあります。設定値が正しくなければアボートコードが送信され、初期化中に数値が正しくなければエマージェンシプロトコルのエラーメッセージ(0x8B26h)が送信されるので、内容を確認してください。

このオブジェクトは、EEPROM を変更してセーブすることができます。

! EEPROM への保存

- ドライバが“Operational Enabled”または“Quick Stop Active”でないこと
- 「運転プロファイルタイプ」(6086h)に新しい値を書込む。
- 「パラメータストア」(1010h)に“save”を書込み、完了するまで待機(処理時間に注意)
- NMT ノードリセット

注)このオブジェクトは、位置プロファイルでも使用されます。

■オブジェクト 6083h：加速プロファイル(Profile Acceleration)

このオブジェクトは、加速指令を示しています。

- ・設定可能範囲は、[10 ～ 319000] rpm/s です。
- ・加速プロファイルは、最大加速(60C5h：0)より小さくする必要があります。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
6083h	VAR	UNSIGNED32	オプション

エントリ内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	rw	なし	1 ～ 2147483647	メーカー固有	u.u.

加速ファクタグループ(6097h)を使用して変換することができます。

$$\text{加速プロファイル } [\text{inc/s}^2] = \left(\frac{\text{加速ファクタ 分子 (6097h:1)}}{\text{加速ファクタ 分母 (6097h:2)}} \right) \times \text{加速プロファイル (6083h) [u.u.]}$$

加速プロファイル(6083h)の単位は、[inc/s²]で、分子、分母ともに1です。

⚠ 注意

このオブジェクトは、その他のオブジェクトによって設定値が制限されることがあります。設定値が正しくなければアボートコードが送信され、初期化中に数値が正しくなければエマージェンシプロトコルのエラーメッセージ(0x8B13h)が送信されるので、内容を確認してください。

【6】CANopen 運転モード

ドライバは、次のアボートコードを送信することがあります。

- 0x06090031：(60C5h: 0)の設定値を超えたパラメータ値の書き込み
「最大加速」(60C5h: 0)より小さく設定する必要があります。
- 0x06040030：設定可能範囲外のパラメータ値の書き込み

加速プロファイルは、運転中に変更することができます。

このオブジェクトは、EEPROM を変更してセーブすることができます。

! EEPROM への保存

- ドライバが“Operational Enabled”または“Quick Stop Active”でないこと
- 「加速プロファイル」(6083h)に新しい値を書込む。
- 「パラメータストア」(1010h)に“save”を書込み、完了するまで待機(処理時間に注意)
- NMT ノードリセット

注)このオブジェクトは、位置プロファイルでも使用されます。

■オブジェクト 6084h：減速プロファイル(Profile Deceleration)

このオブジェクトは、減速指令を示しています。

- 設定可能範囲は、[10 ～ 319000] rpm/s です。
- 減速プロファイルは、最大減速(60C6h: 0)より小さくする必要があります。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
6084h	VAR	UNSIGNED32	オプション

エントリ内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	rw	なし	1 ～ 2147483647	メーカー固有	u.u.

加速ファクタグループ(6097h)を使用して変換することができます。

$$\text{減速プロファイル } [\text{inc/s}^2] = \left(\frac{\text{加速ファクタ 分子 (6097h: 1)}}{\text{加速ファクタ 分母 (6097h: 2)}} \right) \times \text{減速プロファイル (6084h) [u.u.]}$$

減速プロファイル(6084h)の単位は、[inc/s²]で、分子、分母ともに1です。

⚠ 注 意

このオブジェクトは、その他のオブジェクトによって設定値が制限されることがあります。設定値が正しくなければアボートコードが送信され、初期化中に数値が正しくなければエマージェンシプロトコルのエラーメッセージ(0x8B14h)が送信されるので、内容を確認してください。

ドライバは、次のアボートコードを送信することがあります。

- 0x06090031：(60C6h: 0)の設定値を超えたパラメータ値の書き込み
「最大減速」(60C6h: 0)より小さく設定する必要があります。
- 0x06040030：設定可能範囲外のパラメータ値の書き込み

【6】 CANopen 運転モード

減速プロファイルは、運転中に変更することができます。

このオブジェクトは、EEPROM を変更してセーブすることができます。

⚠ EEPROM への保存

- ドライバが“Operational Enabled”または“Quick Stop Active”でないこと
- 「減速プロファイル」(6084h)に新しい値を書込む。
- 「パラメータストア」(1010h)に“save”を書込み、完了するまで待機(処理時間に注意)
- NMT ノードリセット

注)このオブジェクトは、位置プロファイルでも使用されます。

■オブジェクト 60C5h：最大加速(Max Acceleration)

このオブジェクトは、最大加速を示しています。

許容できる加速値で、モータを保護し、過電流を防ぐため加速制限の設定に使用します。

- ・設定可能範囲は、[10 ～ 319000] rpm/s です。
- ・最大加速は、「加速 ABS」(3001h: 2)より小さく設定する必要があります。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
60C5h	VAR	UNSIGNED32	オプション

エントリ内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	rw	なし	1 ～ 2147483647	メーカー固有	u.u.

加速ファクタグループ(6097h)により変換することができます。

$$\text{最大加速 [inc/s}^2\text{]} = \left(\frac{\text{加速ファクタ 分子 (6097h: 1)}}{\text{加速ファクタ 分母 (6097h: 2)}} \right) \times \text{最大加速 (60C5h) [u.u.]}$$

最大加速(60C5h)の単位は、[inc/s²]で、分母、分子ともに1です。

⚠ 注意

このオブジェクトは、その他のオブジェクトによって設定値が制限されることがあります。設定値が正しくなければアボートコードが送信され、初期化中に数値が正しくなければエマージェンシプロトコルのエラーメッセージ(0x8B15h)が送信されるので、内容を確認してください。

ドライバは、次のアボートコードを送信することがあります。

- ・0x06090031：(3001h: 2)の設定値を超えたパラメータ値の書込み
「加速 ABS」(3001h: 2)より小さく設定する必要があります。
- ・0x06040030：設定可能範囲外のパラメータ値の書込み

【6】CANopen 運転モード

このオブジェクトは、EEPROM を変更してセーブすることができます。

⚠ EEPROM への保存

- ドライバが “Operational Enabled” または “Quick Stop Active” でないこと
- 「最大加速」(60C5h) に新しい値を書込む。
- 「パラメータストア」(1010h) に “save” を書き込み、完了するまで待機(処理時間に注意)
- NMT ノードリセット

注) このオブジェクトは、位置プロファイルでも使用されます。

■オブジェクト 60C6h：最大減速(Max Deceleration)

このオブジェクトは、最大減速を示しています。

許容できる減速値で、モータを保護し、過電圧を防ぐために減速制限の設定に使用します。

- ・設定可能範囲は、[10 ~ 319000] rpm/s です。
- ・最大減速は、加速 ABS (3001h: 2) より小さくする必要があります。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
60C6h	VAR	UNSIGNED32	オプション

エントリ内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	rw	なし	1 ~ 2147483647	メーカー固有	u.u.

加速ファクタグループ(6097h)により変換することができます。

$$\text{最大減速 [inc/s}^2\text{]} = \left(\frac{\text{加速ファクタ 分子 (6097h: 1)}}{\text{加速ファクタ 分母 (6097h: 2)}} \right) \times \text{最大減速 (60C6h) [u.u.]}$$

最大減速(60C6h)の単位は、[inc/s²] で、分子、分母ともに 1 です。

⚠ 注意

このオブジェクトは、その他のオブジェクトによって設定値が制限されることがあります。設定値が正しくなければアボートコードが送信され、初期化中に数値が正しくなければエマージェンシプロトコルのエラーメッセージ(0x8B16h)が送信されるので、内容を確認してください。

ドライバは、次のアボートコードを送信することがあります。

- ・ 0x06090031：(3001h: 2) の設定値を超えたパラメータ値の書き込み
「加速 ABS」(3001h: 2) より小さく設定する必要があります。
- ・ 0x06040030：設定可能範囲外のパラメータ値の書き込み

【6】 CANopen 運転モード

このオブジェクトは、EEPROM を変更してセーブすることができます。

! EEPROM への保存

- ドライバが “Operational Enabled” または “Quick Stop Active” でないこと
- 「最大減速」(60C6h) に新しい値を書込む。
- 「パラメータストア」(1010h) に “save” を書き込み、完了するまで待機(処理時間に注意)
- NMT ノードリセット

注) このオブジェクトは、位置プロファイルでも使用されます。

■オブジェクト 606Bh：速度要求値(Velocity Demand Value)

このオブジェクトは、速度指令発生器の出力値を示しています。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
606Bh	VAR	INTEGER32	オプション

エントリ内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	ro	なし	-2147483648 ～ 2147483648	メーカー固有	u.u.

■オブジェクト 606Ch：速度検出値(Velocity Actual Value)

このオブジェクトは、回転検出器で得られた速度の検出値を示しています。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
606Ch	VAR	INTEGER32	条件付き強制(pv または csv がサポートされている)

エントリ内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	ro	あり(初期値)	-2147483647 ～ 2147483647	メーカー固有	u.u.

■オブジェクト 606Dh：速度ウィンドウ(Velocity Window)

このオブジェクトは、速度ウィンドウを示しています。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
606Dh	VAR	UNSIGNED16	オプション

エントリ内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	rw	あり(初期値)	1 ～ 65535	メーカー固有	u.u.

加速ファクタグループ(6096h)を使用して変換することができます。

$$\text{速度ウィンドウ [inc/s]} = \left(\frac{\text{速度ファクタ 分子 (6096h: 1)}}{\text{速度ファクタ 分母 (6096h: 2)}} \right) \times \text{速度ウィンドウ (606Dh) [u.u.]}$$

【6】CANopen 運転モード

⚠ 注意

このオブジェクトは、その他のオブジェクトによって設定値が制限されることがあります。設定値が正しくなければアボートコードが送信され、初期化中に数値が正しくなければエマーゲンシプロトコルのエラーメッセージ(0x8B1Bh)が送信されるので、内容を確認してください。

ドライバは、次のアボートコードを送信することがあります。

- 0x05040001：書き込み数値が0のため無効

このオブジェクトは、EEPROM を変更してセーブすることができます。

! EEPROM への保存

- ドライバが“Operational Enabled”または“Quick Stop Active”でないこと
- 「速度ウインドウ」(606Dh)に新しい値を書込む。
- 「パラメータストア」(1010h)に“save”を書込み、完了するまで待機(処理時間に注意)
- NMT ノードリセット

注)このオブジェクトは、位置プロファイルでも使用されます。

■オブジェクト 606Eh：速度ウインドウ時間(Velocity Window Time)

このオブジェクトは、速度ウインドウ時間を示しています。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
606Eh	VAR	UNSIGNED16	オプション

エントリ内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	rw	なし	1 ～ 65535	メーカー固有	ms

⚠ 注意

このオブジェクトは、その他のオブジェクトによって設定値が制限されることがあります。設定値が正しくなければアボートコードが送信され、初期化中に数値が正しくなければエマーゲンシプロトコルのエラーメッセージ(0x8B1Ch)が送信されるので、内容を確認してください。

ドライバは、次のアボートコードを送信することがあります。

- 0x06090032：書き込みする値が小さすぎる場合

【6】 CANopen 運転モード

このオブジェクトは、EEPROM を変更してセーブすることができます。

⚠ EEPROM への保存

- ドライバが “Operational Enabled” または “Quick Stop Active” でないこと
- 「速度ウィンドウ時間」(606Eh)に新しい値を書込む。
- 「パラメータストア」(1010h)に “save” を書き込み、完了するまで待機(処理時間に注意)
- NMT ノードリセット

■オブジェクト 606Fh：速度スレッシュホールド (Velocity Threshold)

このオブジェクトは、速度スレッシュホールドを示しています。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
606Fh	VAR	UNSIGNED16	オプション

エントリ内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	rw	なし	0 ～ 65535	メーカー固有	u.u.

加速ファクタグループ(6096h)を使用することにより変更することができます。

$$\text{速度スレッシュホールド [inc/s]} = \left(\frac{\text{速度ファクタ分子 (6096h:1)}}{\text{速度ファクタ分母 (6096h:2)}} \right) \times \text{速度スレッシュホールド (606Fh) [u. u.]}$$

⚠ 注意

このオブジェクトは、その他のオブジェクトによって設定値が制限されることがあります。設定値が正しくなければアボートコードが送信され、初期化中に数値が正しくなければエマージェンシプロトコルのエラーメッセージ(0x8B1Dh)が送信されるので、内容を確認してください。

このオブジェクトは、EEPROM を変更してセーブすることができます。

⚠ EEPROM への保存

- ドライバが “Operational Enabled” または “Quick Stop Active” でないこと
- 「速度スレッシュホールド」(606Fh)に新しい値を書込む。
- 「パラメータストア」(1010h)に “save” を書き込み、完了するまで待機(処理時間に注意)
- NMT ノードリセット

【6】 CANopen 運転モード

■オブジェクト 6070h：速度スレッシュホールド時間(Velocity Threshold Time)

このオブジェクトは、速度スレッシュホールド時間を示しています。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
6070h	VAR	UNSIGNED16	オプション

エントリ内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	rw	なし	1 ～ 65535	メーカー固有	ms

注 意

このオブジェクトは、その他のオブジェクトによって設定値が制限されることがあります。設定値が正しくなければアボートコードが送信され、初期化中に数値が正しくなければエマージェンシプロトコルのエラーメッセージ(0x8B1Eh)が送信されるので、内容を確認してください。

ドライバは、次のアボートコードを送信することがあります。

- 0x06090032：書込みする値が小さすぎる場合

このオブジェクトは、EEPROM を変更してセーブすることができます。

EEPROM への保存

- ドライバが“Operational Enabled”または“Quick Stop Active”でないこと
- 「速度スレッシュホールド時間」(6070h)に新しい値を書込む。
- 「パラメータストア」(1010h)に“save”を書込み、完了するまで待機(処理時間に注意)
- NMT ノードリセット

【6】 CANopen 運転モード

6-5 トルクプロファイルモード

トルクプロファイルモード(PT)では、マスタコントローラから送信される目標トルク(電流)指令に従って、モータの出力トルクを制御します。

電圧レギュレータ(トルク制御器)は、目標トルク指令に比例した出力電流を出力します。

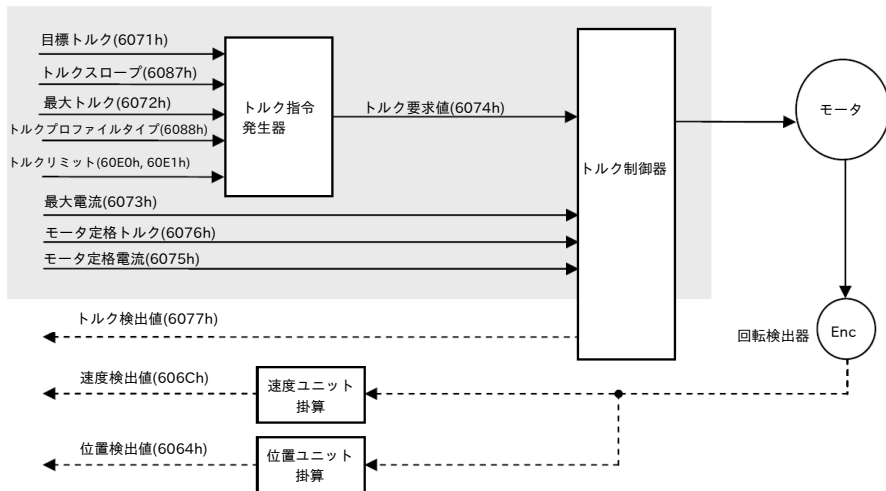


図 6-7 トルク制御の構成

トルクプロファイルモードを動作させるために、事前に必要な設定を以下に示します。

- ・トルクプロファイルモードは、「運転モード」(6060h)に4を設定します。
- ・状態機械 DSP402 の“Operation Enabled”ステートであることを、「ステータスワード」(6041h)で確認します。状態機械の遷移は、コントロールワード(6040h)を使用します。

目標トルクは、オブジェクトディクショナリの「目標トルク」(6071h)で設定されます。

【6】CANopen 運転モード

■トルクプロファイルのオブジェクト

トルクプロファイルモード使用時に、関連するオブジェクトを以下に示します。

表 6-4 トルクプロファイルのオブジェクト

Index	Sub-index	名 称	リード/ライト	M/O	データタイプ	PDO	使用可
0x603F	0	エラーコード	RO	O	U16	-	○
0x6040	0	コントロールワード	R/W	M	U16	RPDO	○
0x6041	0	ステータスワード	RO	M	U16	TPDO	○
0x6060	0	運転モード	R/W	M	I8	RPDO	○
0x6061	0	運転表示モード	RO	M	I8	TPDO	○
0x6071	0	目標トルク	R/W	M	I16	RPDO	○
0x6087	0	トルクスロープ	R/W	O	U32	-	○
0x6072	0	最大トルク	R/W	O	U16	-	-
0x6073	0	最大電流	R/W	O	U16	-	○
0x6076	0	モータ定格トルク	R/W	O	U32	-	-
0x6075	0	モータ定格電流	R/W	O	U32	-	○
0x6077	0	トルク検出値	RO	O	I16	TPDO	○
0x6078	0	電流検出値	RO	O	I16	-	○
0x6079	0	DC リンク電圧	RO	O	U32	-	○
0x60E0	0	ポジティブトルクリミット値	R/W	O	U16	-	○
0x60E1	0	ネガティブトルクリミット値	R/W	O	U16	-	○
0x6074	0	トルク要求値	RO	O	I16	-	-
0x6088	0	トルクプロファイルタイプ	R/W	O	I16	-	○

次のいずれかの条件を満たした場合、運転を終了します。

- ・「目標トルク」(6071h) を 0 に設定 (モータに常時トルクが発生しています。)
- ・コントロールワード (6040h) のビット 8 (Halt : 中止) を 1 に設定して停止
- ・エラー発生による停止 (ドライバは、Fault ステートへ移行)
- ・コントロールワード (6040h) で “Disable Operation”、 “Disable Voltage”、 “Quick Stop” のいずれかへ遷移する命令によって “Operation Enabled” ステートを抜けたため停止
- ・セーフティ状態 (STO 入力) による停止 (モータがフリーラン停止します。)

以下のオブジェクトまたはビットでモニタすることも可能です。

- ・「トルク検出値」(6077h)
- ・「電流検出値」(6078h)
- ・目標到達 : 「ステータスワード」(6041h) のビット 10

■コントロールワードビット

「コントロールワード」(6040h) のビット 8 は、マスタコントローラによって設定可能です。

表 6-5 トルクプロファイルのコントロールワードビット

ビット	数 値	定 義
ビット 8 = 中止 (Halt)	0b	運転が実行されて継続されます。 中止命令により軸が停止します。オプションコード (605Dh) 注)
	1b	

注) オプションコード (605Dh) 未実装。

[6] CANopen 運転モード

■ステータスワードビット

「ステータスワード」(6041h)の以下のビットは、マスタコントローラによって設定可能です。

表 6-6 トルクプロファイルのステータスワードビット

ビット	数 値	定 義
ビット 10 = 目標到達	0b	Halt (コントロールワードのビット 8) = 0: 目標トルクに未到達 Halt (コントロールワードのビット 8) = 1: 減速を開始
	1b	Halt (コントロールワードのビット 8) = 0: 目標トルクに到達 Halt (コントロールワードのビット 8) = 1: 速度 0

⚠ 注 意

トルクプロファイルでは、ダイナミックブレーキ機能が使用できません。
ダイナミックブレーキ機能なしでトルク(電流)が制御されます。
詳細は、「ダイナミックブレーキパラメータ」(3007h: 1)を参照してください。(5-21)

■運転モードの内容

トルクプロファイルモードの運転では、設定した目標トルクに従って動作します。

手順

- ・トルクプロファイルモードは、「運転モード」(6060h)に 4 を設定します。
- ・モータ定格電流 (mA rms) の値を「モータ定格電流」(6075h) に設定します。
(EEPROM に新しい数値をセーブする手順に従います。)
- ・トルクプロファイルのタイプを選択するために、「トルクプロファイルタイプ」(6078h)を設定します。
- ・「トルクプロファイルタイプ」がリニアランプ(台形)の場合、トルク変化の割合を「トルクスロープ」(6087h) で設定します。
- ・「目標トルク」(6071h)を 0 に設定します。
- ・運転モードを開始するために「コントロールワード」(6040h)を設定します。
- ・「目標トルク」(6071h)でトルク設定ポイントを設定します。

新しく設定した目標トルクは、ドライバ運転中であれば即座に反映されて動作します。

または、「コントロールワード」(6040h)のビット 8 (Halt) が 1 の運転停止状態でも、目標速度の変更が反映され、0 を設定することにより新しく設定した速度で動作します。

注)出力トルクは「ポジティブトルクリミット」(60E0h)と「ネガティブトルクリミット」(60E1)で制限されます。

■プロファイルタイプ

プロファイルのタイプは、「トルクプロファイルタイプ」(6088h)により設定することができます。

- ・**ステップ: 数値 “1”** 目標トルクに直ぐに到達します。新しい目標トルク値は、即座に反映されます。
- ・**リニアランプ: 数値 “0”** 「モータ定格電流」(6075h) = 12500 mA rms で「トルクスロープ」(6087h) = 1000 の場合、出力電流が 1000ms で 12500 mA rms (12.5A rms)、100 ms で 1250mA rms (1.25A rms) となります。

【6】CANopen 運転モード

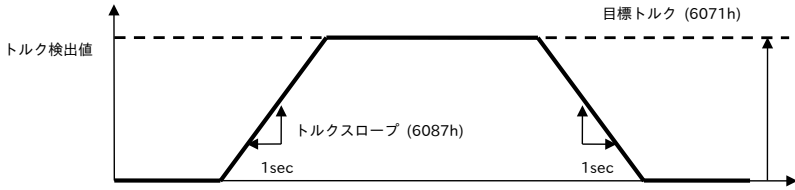


図 6-8 プロファイルタイプ：0 直線ランプ(台形)

- Sin^2 ランプ：数値 “1”(使用不可)

■出力クエリ

- ドライバのステータスが返されるオブジェクト：「ステータスワード」(6041h)
ドライバがリセットされると値は、0 となります。
- モータ出力トルクが返されるオブジェクト：「トルク検出値」(6077h)
- モータ出力電流にフィルタを介して返されるオブジェクト：「電流検出値」(6078h)

(1) トルクプロファイルモード(Halt ビット = 0)

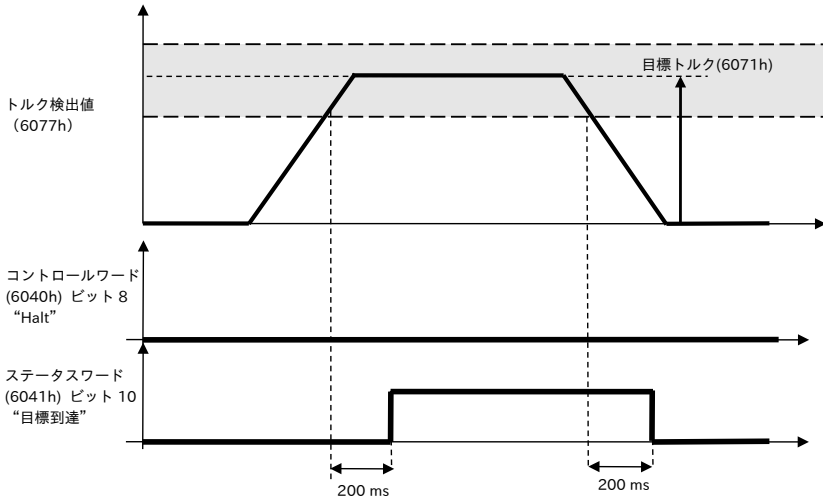


図 6-9 トルク到達(Halt ビット = 0)

【6】 CANopen 運転モード

(2) トルクプロファイルモード (Halt ビット = 1)

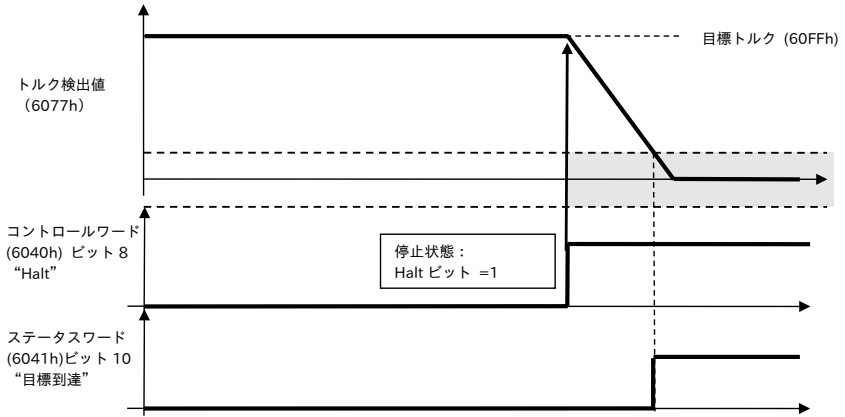


図 6-10 速度ウインドウ (Halt ビット = 1)

【6】 CANopen 運転モード

6-6 トルクプロファイルのオブジェクト

■オブジェクト 6071h：目標トルク (Target Torque)

このオブジェクトは、トルクプロファイルモードでトルク制御器の入力として使用され、目標トルク (トルク指令) を設定して使用します。

目標トルクは、トルク分電流：I_q で与えられるトルク設定ポイントです。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
6071h	VAR	INTEGER16	強制

エントリ内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	rw	あり	1 ～ 65535	メーカー固有	%

例：S サイズモータを使用時、トルク分電流 2000 mA rms 相当のトルクが必要な場合、「モータ定格電流」(6075h) が 12500 mA rms なので、以下の目標トルク (6071h) になります。

$$\text{目標トルク [6071h]} = 2000 \text{ mA rms} \times \frac{1000}{12500 \text{ mA rms}} = 160$$

目標トルク (6071h) には、160 を設定ください。これは、モータ定格電流の 16.0% を示しています。

ドライバは、次のアバートコードを送信することがあります。

- ・ 0x06090031：書き込みする値が大きすぎる場合

■オブジェクト 6075h：モータ定格電流 (Motor Rated Current)

このオブジェクトは、設定したモータ定格電流を示しています。

電流に関連しているすべてのパラメータは、この値を参照します。単位は mA rms です。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
6075h	VAR	UNSIGNED32	オプション

エントリ内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	rw	なし	1 ～ 2147483647	メーカー固有	mA rms

「モータ定格電流」(6075h: 0) の設定値は、「ドライバ定格電流」(3003h: 3) の設定値以下で使用してください。

モータ定格電流 (6075h: 0) [mA rms]

≤

ドライバ定格電流 (3003h: 3) [Arms/100]

例：ドライバ定格電流 (3003h: 3) が 125 [A rms/10] (=12.5 A rms) に設定されている場合、モータ定格電流 (6075h: 0) ≤ 12500 [mA rms] が設定可能範囲となります。

【6】 CANopen 運転モード

ドライバは、次のアボートコードを送信することがあります。

- 0x06090031：（3003h: 3）の設定値を超えたパラメータ値の書き込み
「ドライバ定格電流」（3003h: 3）より小さく設定する必要があります。
- 0x06040030：設定可能範囲外のパラメータ値の書き込み

⚠ 注意

このオブジェクトは、その他のオブジェクトによって設定値が制限されることがあります。設定値が正しくなければアボートコードが送信され、初期化中に数値が正しくなければエマージェンシプロトコルのエラーメッセージ(0x8B1Fh)が送信されるので、内容を確認してください。

このオブジェクトは、EEPROM を変更してセーブすることができます。

❗ EEPROM への保存

- ドライバが“Operational Enabled”または“Quick Stop Active”でないこと
- 「モータ定格電流」（6075h）に新しい値を書込む。
- 「パラメータストア」（1010h）に“save”を書込み、完了するまで待機(処理時間に注意)
- NMT ノードリセット

■オブジェクト 6073h：最大電流(Max Current)

このオブジェクトは、モータの最大許容トルクの設定を示しています。
単位は [1000/ モータ定格電流] で与えられます。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
6073h	VAR	UNSIGNED16	オプション

エントリ内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	rw	なし	1 ～ 32767	メーカー固有	1000/ モータ定格電流

「最大電流」(6073h: 0) の値は、[mA rms] に変換されて、「ドライバピーク電流」(3003h: 2) [mA rms] 以下に設定する必要があります。

最大電流 (6073h: 0) → [mA rms] ≤ ドライバピーク電流 (3003h: 2)

例：ドライバピーク電流 (3003h: 2) が 410 [A rms/100] に設定されていて、モータ定格電流が 12500 mA rms の場合、最大電流 (6073h: 0) を次の式で表わすことができます。

$$\text{最大電流 [6073h]} = 410 \times 100 \text{ (A rms)} \times \frac{1000}{12500 \text{ (mA rms)}} = 3280$$

【6】CANopen 運転モード

ドライバは、次のアボートコードを送信することがあります。

- 0x06040030：設定可能範囲外のパラメータ値の書き込み
- 0x06090031：(3003h: 2)の設定値を超えたパラメータ値の書き込み
「ドライバピーク電流」(3003h: 2)より小さく設定する必要があります。

⚠ 注意

このオブジェクトは、その他のオブジェクトによって設定値が制限されることがあります。設定値が正しくなければアボートコードが送信され、初期化中に数値が正しくなければエマージェンシプロトコルのエラーメッセージ(0x8B23h)が送信されるので、内容を確認してください。

このオブジェクトは、EEPROM を変更してセーブすることができます。

⚠ EEPROM への保存

- ドライバが“Operational Enabled”または“Quick Stop Active”でないこと
- 「最大電流」(6073h)に新しい値を書込む。
- 「パラメータストア」(1010h)に“save”を書込み、完了するまで待機(処理時間に注意)
- NMT ノードリセット

■オブジェクト 6087h：トルクスロープ(Torque Slope)

このオブジェクトは、出力トルクの変化率の設定を示しています。

単位は、[(モータ定格電流 /1000) /s] で与えられます。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
6087h	VAR	UNSIGNED32	強制 IF

エン트리内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	rw	なし	1 ~ 2147483647	1000	(モータ定格電流 /1000)/s

ドライバは、次のアボートコードを送信することがあります。

- 0x06040030：設定可能範囲外のパラメータ値の書き込み

⚠ 注意

このオブジェクトは、その他のオブジェクトによって設定値が制限されることがあります。設定値が正しくなければアボートコードが送信され、初期化中に数値が正しくなければエマージェンシプロトコルのエラーメッセージ(0x8B25h)が送信されるので、内容を確認してください。

【6】 CANopen 運転モード

このオブジェクトは、EEPROM を変更してセーブすることができます。

⚠ EEPROM への保存

- ドライバが “Operational Enabled” または “Quick Stop Active” でないこと
- 「トルクスロープ」(6087h) に新しい値を書込む。
- 「パラメータストア」(1010h) に “save” を書き込み、完了するまで待機(処理時間に注意)
- NMT ノードリセット

■オブジェクト 6088h：トルクプロファイルタイプ(Torque Profile Type)

このオブジェクトは、設定されたトルク変化が発生するプロファイルタイプを示しています。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
6088h	VAR	INTEGER16	オプション

エントリ内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	rw	なし	-32767 ～ 32767	メーカ固有	-

数値定義：

数 値	定 義	適 用
-1	ステップ (Immediately)	あり
0	リニアランプ (台形プロファイル) (初期値)	あり (初期値)
1	\sin^2 ランプ	使用不可

ドライバは、次のアボートコードを送信することがあります。

- ・ 0x08000024：データ設定不可(設定不可の値の場合)

⚠ 注 意

このオブジェクトは、その他のオブジェクトによって設定値が制限されることがあります。設定値が正しくなければアボートコードが送信され、初期化中に数値が正しくなければエマージェンシプロトコルのエラーメッセージ(0x8B18h)が送信されるので、内容を確認してください。

このオブジェクトは、EEPROM を変更してセーブすることができます。

⚠ EEPROM への保存

- ドライバが “Operational Enabled” または “Quick Stop Active” でないこと
- 「トルクプロファイルタイプ」(6088h) に新しい値を書込む。
- 「パラメータストア」(1010h) に “save” を書き込み、完了するまで待機(処理時間に注意)
- NMT ノードリセット

【6】CANopen 運転モード

■オブジェクト 6074h：トルク要求値(Torque Demand)

このオブジェクトは、トルク指令発生器の出力を示しています。

単位は、[1000/ モータ定格電流] です。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
6074h	VAR	INTEGER16	オプション

エン트리内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	Ro	なし	-32767 ~ 32767	メーカー固有	1000/ モータ定格電流

■オブジェクト 6077h：トルク検出値(Torque Actual Value)

このオブジェクトは、実際のモータの瞬時出力トルク値を示しています。

単位は、[1000/ モータ定格電流] です。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
6077h	VAR	INTEGER16	強制

エン트리内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	ro	あり	-32767 ~ 32767	メーカー固有	1000/ モータ定格電流

■オブジェクト 6078h：トルク電流(Torque Actual Current)

このオブジェクトは、実際のモータに流れる電流値を示しています。

単位は、[1000/ モータ定格電流] です。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
6078h	VAR	INTEGER16	オプション

エン트리内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	ro	なし	-32767 ~ 32767	メーカー固有	1000/ モータ定格電流

■オブジェクト 6079h：DC リンク電圧(DC Link circuit Voltage)

このオブジェクトは、ドライバ内部の瞬時の DC リンク電圧を示しています。単位は、[mv] です。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
6079h	VAR	UNSIGNED32	オプション

エン트리内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	ro	なし	0 ~ 4294967296	メーカー固有	mV

注) このオブジェクトは、全プロファイルで読出し可能です。

[6] CANopen 運転モード

■オブジェクト 60E0h：ポジティブトルクリミット値(Positive Torque Limit Value)

このオブジェクトは、設定された最大のポジティブトルクを示していて、単位がドライバピーク電流に対する%で与えられます。

正転方向で力行動作の場合、または逆転方向で回生動作の場合に機能します。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
60E0h	VAR	UNSIGNED16	オプション

エン트리内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	ro	なし	0 ～ 100	100	%

注) 1. このオブジェクトは、速度プロファイルでも使用されます。

2. 各ドライバのピーク電流値は、以下の通りです。

ECO-S 用：41.7A ECO-M、PRO-M 用：96.3A PRO-L 用：136.2A

ドライバは、次のアボートコードを送信することがあります。

- 0x06040030：設定可能範囲外のパラメータ値の書込み

⚠ 注意

このオブジェクトは、その他のオブジェクトによって設定値が制限されることがあります。設定値が正しくなければアボートコードが送信され、初期化中に数値が正しくなければエマーゲンシプロトコルのエラーメッセージ(0x8B23h)が送信されるので、内容を確認してください。

注)このオブジェクトは、EEPROM にセーブすることができません。RAM は、変更することができます。

■オブジェクト 60E1h：ネガティブトルクリミット値(Negative Torque Limit Value)

このオブジェクトは、設定された最大のネガティブトルクを示していて、単位がドライバピーク電流に対する%で与えられます。

逆転方向で力行動作の場合、または正転方向で回生動作の場合に機能します。

オブジェクト内容：

Index	オブジェクトコード	データタイプ	カテゴリ
60E1h	VAR	UNSIGNED16	オプション

エン트리内容：

Sub-Index	アクセス	PDO マッピング	データ範囲	初期値	単 位
00h	ro	なし	0 ～ 100	100	%

【6】 CANopen 運転モード

- 注) 1. このオブジェクトは、速度プロファイルでも使用されます。
2. 各ドライバのピーク電流値は以下の通りです。
ECO-S 用：41.7A、ECO-M・PRO-M 用：96.3A、PRO-L 用：136.2A

ドライバは、次のアボートコードを送信することがあります。

- ・ 0x06040030：設定可能範囲外のパラメータ値の書込み

⚠ 注意
このオブジェクトは、その他のオブジェクトによって設定値が制限されることがあります。設定値が正しくなければアボートコードが送信され、初期化中に数値が正しくなければエマージェンシプロトコルのエラーメッセージ(0x8B24h)が送信されるので、内容を確認してください。

注) このオブジェクトは、EEPROM にセーブすることができません。RAM は変更することができません。

【7】 CANopen オブジェクトリスト

Index	Sub.	名 称	タイプ	O/M	属 性	備 考
DS301 標準オブジェクト						
1000h	0	Device Type	UINT32	M	RO	設 定
1001h	0	Error Register	UINT32	O	RO	
1002h	0	Manufacturer Status Register	UINT32	O	RO	
1003h	0	Pre-Defined Error Field	UINT32	M	RO	アラーム
	1		UINT32	M	RO	
	2		UINT32	O	RO	
	3		UINT32	O	RO	
	4		UINT32	O	RO	
	5		UINT32	O	RO	
	6		UINT32	O	RO	
	7		UINT32	O	RO	
	8	History Error Field	UINT32	O	RO	
	9		UINT32	O	RO	
	10		UINT32	O	RO	
	11		UINT32	O	RO	
	12		UINT32	O	RO	
	13		UINT32	O	RO	
	14		UINT32	O	RO	
	15		UINT32	O	RO	
1005h	0	Cob-ID Sync	UINT32	-	R/W	設 定
1008h	0	Manufacturer Device Name	STRING	M	RO	通 信
1009h	0	Manufacturer Hardware Version	STRING	M	RO	
100Ah	0	Manufacturer Software Version	STRING	M	RO	
100Ch	0	Guard Time	UINT16	O	R/W	設 定
100Dh	0	Lifetime Factor	UINT8	O	R/W	メモリパラメータ
1010h	0	Store Parameter Fields	UINT32	O	R/W	
	1	Save all Parameters	UINT32	M	R/W	
	2	Save Communication Parameters	UINT32	O	R/W	
	3	Save Application Parameters	UINT32	O	R/W	
	4	Save Manufacturer Parameters	UINT32	O	R/W	
1011h	5	Save Data Factory Parameters	UINT32	O	R/W	予約領域
	0	Restore Default Parameter	UINT32	O	R/W	メモリパラメータ
	1	Restore all Default Parameters	UINT32	O	R/W	
	2	Restore Communication Default Parameters	UINT32	O	R/W	使用不可
	3	Restore Application Default Parameters	UINT32	O	R/W	
	4	Restore Manufacturer Default Parameters	UINT32	O	R/W	
	5	Restore Data Factory Parameters	UINT32	O	R/W	予約領域
1014h	0	Cob-ID Emergency Message	UINT32	O	RO	使用不可
1017h	0	Producer Heartbeat Time	UINT16	M	R/W	設 定
1018h	0	Identity Object	UINT32	M	RO	-
	1	Vendor Id	UINT32	M	RO	設 定
	2	Product Code	UINT32	O	RO	使用不可
	3	Revision number	UINT32	O	RO	
	4	Serial number	UINT32	O	RO	
1029h	0	Error Behavior	UINT8	O	RO	
	1	Communication Error	-	O	R/W	
1400h	0	Receive PDO Communication Parameter 1	UINT8	M	R/W	設 定
	1	COB-ID	UINT32	M	R/W	
	2	Transmission Type	UINT8	M	R/W	
	3	Inhibit Time	UINT16	O	R/W	予約領域
	4	Compatibility Entry	UINT8	O	R/W	
1401h	5	Event Timer	UINT16	O	R/W	設 定
	0	Receive PDO Communication Parameter 2	UINT8	M	R/W	
	1	COB-ID	UINT32	M	R/W	
	2	Transmission Type	UINT8	M	R/W	予約領域
	3	Inhibit Time	UINT16	O	R/W	
	4	Compatibility Entry	UINT8	O	R/W	
	5	Event Timer	UINT16	O	R/W	設 定

【7】 CANopen オブジェクトリスト

Index	Sub.	名 称	タイプ	O/M	属 性	備 考
1402h	0	Receive PDO Communication Parameter 3	UINT8	M	R/W	設 定
	1	COB-ID	UINT32	M	R/W	
	2	Transmission Type	UINT8	M	R/W	
	3	Inhibit Time	UINT16	O	R/W	予約領域
	4	Compatibility Entry	UINT8	O	R/W	
1403h	5	Event Timer	UINT16	O	R/W	設 定
	0	Receive PDO Communication Parameter 4	UINT8	M	R/W	
	1	COB-ID	UINT32	M	R/W	
	2	Transmission Type	UINT8	M	R/W	予約領域
	3	Inhibit Time	UINT16	O	R/W	
1600 h	4	Compatibility Entry	UINT8	O	R/W	設 定
	5	Event Timer	UINT16	O	R/W	
	0	Receive PDO Mapping Parameter 1	UINT8	M	R/W	
	1	Mapping Entry 1	UINT32	M	R/W	
	2	Mapping Entry 2	UINT32	M	R/W	
	3	Mapping Entry 3	UINT32	M	R/W	
	4	Mapping Entry 4	UINT32	M	R/W	
	5	Mapping Entry 5	UINT32	M	R/W	
	6	Mapping Entry 6	UINT32	M	R/W	
	7	Mapping Entry 7	UINT32	M	R/W	
	8	Mapping Entry 8	UINT32	M	R/W	
	0	Receive PDO Mapping Parameter 2	UINT32	M	R/W	
	1	Mapping Entry 1	UINT32	M	R/W	
	2	Mapping Entry 2	UINT32	M	R/W	
	3	Mapping Entry 3	UINT32	M	R/W	
1601h	4	Mapping Entry 4	UINT32	M	R/W	
	5	Mapping Entry 5	UINT32	M	R/W	
	6	Mapping Entry 6	UINT32	M	R/W	
	7	Mapping Entry 7	UINT32	M	R/W	
	8	Mapping Entry 8	UINT32	M	R/W	
	0	Receive PDO Mapping Parameter 3	UINT32	M	R/W	
	1	Mapping Entry 1	UINT32	M	R/W	
	2	Mapping Entry 2	UINT32	M	R/W	
1602h	3	Mapping Entry 3	UINT32	M	R/W	
	4	Mapping Entry 4	UINT32	M	R/W	
	5	Mapping Entry 5	UINT32	M	R/W	
	6	Mapping Entry 6	UINT32	M	R/W	
	7	Mapping Entry 7	UINT32	M	R/W	
	8	Mapping Entry 8	UINT32	M	R/W	
	0	Receive PDO Mapping Parameter 4	UINT8	M	R/W	
	1	Mapping Entry 1	UINT32	M	R/W	
1603h	2	Mapping Entry 2	UINT32	M	R/W	
	3	Mapping Entry 3	UINT32	M	R/W	
	4	Mapping Entry 4	UINT32	M	R/W	
	5	Mapping Entry 5	UINT32	M	R/W	
	6	Mapping Entry 6	UINT32	M	R/W	
	7	Mapping Entry 7	UINT32	M	R/W	
	8	Mapping Entry 8	UINT32	M	R/W	
	0	Transmit PDO Communication Parameter 1	UINT8	M	R/W	
1800h	1	COB-ID	UINT32	M	R/W	予約領域
	2	Transmission Type	UINT8	M	R/W	
	3	Inhibit Time	UINT16	O	R/W	
	4	Compatibility Entry	UINT8	O	R/W	設 定
	5	Event Timer	UINT16	O	R/W	
1801h	0	Transmit PDO Communication Parameter 2	UINT8	M	R/W	予約領域
	1	COB-ID	UINT32	M	R/W	
	2	Transmission Type	UINT8	M	R/W	
	3	Inhibit Time	UINT16	O	R/W	設 定
	4	Compatibility Entry	UINT8	O	R/W	
	5	Event Timer	UINT16	O	R/W	設 定

【7】 CANopen オブジェクトリスト

Index	Sub.	名 称	タイプ	O/M	属 性	備 考
1802h	0	Transmit PDO Communication Parameter 3	UINT8	M	R/W	設 定
	1	COB-ID	UINT32	M	R/W	
	2	Transmission Type	UINT8	M	R/W	
	3	Inhibit Time	UINT16	O	R/W	予約領域
	4	Compatibility Entry	UINT8	O	R/W	
1803h	5	Event Timer	UINT16	O	R/W	設 定
	0	Transmit PDO Communication Parameter 4	UINT8	M	R/W	
	1	COB-I	UINT32	M	R/W	
	2	Transmission Type	UINT8	M	R/W	予約領域
	3	Inhibit Time	UINT16	O	R/W	
1A00h	4	Compatibility Entry	UINT8	O	R/W	設 定
	5	Event Timer	UINT16	O	R/W	
	0	Transmit PDO Mapping Parameter 1	UINT8	M	R/W	
	1	Mapping Entry 1	UINT32	M	R/W	
	2	Mapping Entry 2	UINT32	M	R/W	
	3	Mapping Entry 3	UINT32	M	R/W	
	4	Mapping Entry 4	UINT32	M	R/W	
	5	Mapping Entry 5	UINT32	M	R/W	
1A01h	6	Mapping Entry 6	UINT32	M	R/W	設 定
	7	Mapping Entry 7	UINT32	M	R/W	
	8	Mapping Entry 8	UINT32	M	R/W	
	0	Transmit PDO Mapping Parameter 2	UINT8	M	R/W	
	1	Mapping Entry 1	UINT32	M	R/W	
	2	Mapping Entry 2	UINT32	M	R/W	
	3	Mapping Entry 3	UINT32	M	R/W	
	4	Mapping Entry 4	UINT32	M	R/W	
1A02h	5	Mapping Entry 5	UINT32	M	R/W	設 定
	6	Mapping Entry 6	UINT32	M	R/W	
	7	Mapping Entry 7	UINT32	M	R/W	
	8	Mapping Entry 8	UINT32	M	R/W	
	0	Transmit PDO Mapping Parameter 3	UINT8	M	R/W	
	1	Mapping Entry 1	UINT32	M	R/W	
	2	Mapping Entry 2	UINT32	M	R/W	
	3	Mapping Entry 3	UINT32	M	R/W	
1A03h	4	Mapping Entry 4	UINT32	M	R/W	設 定
	5	Mapping Entry 5	UINT32	M	R/W	
	6	Mapping Entry 6	UINT32	M	R/W	
	7	Mapping Entry 7	UINT32	M	R/W	
	8	Mapping Entry 8	UINT32	M	R/W	
	0	Transmit PDO Mapping Parameter 4	UINT32	M	R/W	
	1	Mapping Entry 1	UINT32	M	R/W	
	2	Mapping Entry 2	UINT32	M	R/W	
メーカー固有オブジェクト						
2000h	0	Node ID	UINT8	M	R/W	設 定
2001h	0	CAN Baudrate	UINT16	M	R/W	
2002h	0	Drive Status	INT16	O	RO	TELL
2003h	0	Warning	UINT32	O	RO	
2004h	0	State Servo Drive Machine	INT16	O	RO	
2030h	0	Drive Temperature	INT16	O	RO	
2031h	0	Motor Temperature	INT16	O	RO	
2032h	0	Heat Sink Temperature	INT16	O	RO	
2041h	0	Voltage Bus	INT16	O	RO	
2050h	0	Torque Current	INT16	O	RO	

【7】 CANopen オブジェクトリスト

Index	Sub.	名 称	タイプ	O/M	属 性	備 考
2053h	0	Velocity Filtered	INT16	O	RO	TELL
2054h	0	Energy I2t	UINT16	O	RO	
2060h	0	Impulse	INT16	O	RO	
2070h	0	Torque Current Window	INT16	O	RO	設 定
3001h	0	Limits Parameter	UINT32	O	RO	-
	1	Velocity ABS	UINT32	O	RO	TELL
	2	Acceleration ABS	UINT32	O	RO	
	3	Limit Velocity Profile	UINT32	O	RO	
	4	-	UINT32	O	RO	予約領域
	5	-	UINT32	O	RO	
3002h	0	Brake Parameters	INT16	M IF	RO	設 定
	1	Motor Brake Option	INT16	M IF	R/W	
	2	Motor Brake Delay	INT16	M IF	R/W	
	3	Brake Unlock time	INT16	M IF	R/W	
	4	Brake Timeout	INT16	M IF	R/W	
	5	Automatic/Manual Mode Configuration	INT16	M IF	R/W	
	6	Motor Brake Status	INT16	M IF	RO	TELL
3003h	7	Brake Type	INT16	M IF	RO	設 定
	0	Drive Size Parameters	INT16	O	RO	-
	1	Maximum Current	INT16	O	RO	TELL
	2	Peak Current	INT16	O	RO	
	3	Rated Current	INT16	O	RO	
	4	I ² T	INT16	O	RO	予約領域
	5	予約領域	INT16	O	RO	
	6	予約領域	INT16	O	RO	
3004h	7	予約領域	INT16	O	RO	
	0	Feedback Parameters	INT16	O	RO	TELL
	1	Feedback Type	INT16	O	RO	
	2	Resolution	INT16	O	RO	
	3	Absolute Position (Initial Value16bit LSB)	INT16	O	RO	
	4	Absolute Position (Initial Value16bit MSB)	INT16	O	RO	
	5	Absolute Position (Actual Value16bit LSB)	INT16	O	RO	
	6	Absolute Position (Actual Value16bit MSB)	INT16	O	RO	
3005h	A	Encoder Command	INT16	O	R/W	設 定
	0	Filter Parameters	INT16	O	RO	
	1	Filter Polo 1 Set Point (予約領域)	INT16	O	RO	TELL
	2	-	INT16	O	RO	-
	3	Filter Polo 1 Velocity (予約領域)	INT16	O	RO	TELL
	4	Filter Polo 2 Velocity (予約領域)	INT16	O	RO	
	5	Filter Polo 1 Velocity Filtered	INT16	O	R/W	設 定
	6	Filter Polo 1 Velocity Actual	INT16	O	R/W	
	7	Filter Polo 1 PID (予約領域)	INT16	O	RO	TELL
	8	Filter Polo 2 PID (予約領域)	INT16	O	RO	
	9	-	INT16	O	RO	-
	10	-	INT16	O	RO	
3006h	0	Motor Specific Settings	INT16	O	RO	TELL
	1	Motor Part Number	INT16	O	RO	
	2	Max Motor Speed	INT16	O	RO	
	3	N Poli	INT16	O	RO	
	4	-	INT16	O	RO	-
	5	-	INT16	O	RO	
3007h	0	Dynamic Brake Parameter	INT16	M IF	RO	設 定
	1	Dynamic Brake Option	INT16	M IF	R/W	
	2	Holding Torque Time	INT16	M IF	R/W	
	3	Dynamic Brake Status	INT16	M IF	RO	TELL
	4	Decrement Step Ramp	INT16	M IF	R/W	設 定
	5	Max Timeout Dynamic Stop	INT16	M IF	R/W	-

注) 2070h、3004h Sub.3 ~ 6、A : FW Ver. 2.3.1 から追加

【7】 CANopen オブジェクトリスト

Index	Sub.	名 称	タイプ	O/M	属 性	備 考
3008h	0	Emergency Enable Parameter	INT16	M IF	RO	設 定
	1	Emergency Enable Option	INT16	M IF	R/W	
	2	Emergency Input Neg	INT16	M IF	R/W	
	3	Emergency Status	INT16	M IF	RO	
3021h	0	Digital Input 1	INT16	O	RO	TELL
	1	Configuration	INT16	O	RO	
	2	State	INT16	O	RO	
	3	Level	INT16	O	RO	
	4	-	INT16	O	RO	-
	5	-	INT16	O	RO	
3022 h	0	Digital Input 2	INT16	O	RO	TELL
	1	Configuration	INT16	O	RO	
	2	State	INT16	O	RO	
	3	Level	INT16	O	RO	
	4	-	INT16	O	RO	-
	5	-	INT16	O	RO	
3023h	0	Digital Input 3	INT16	O	RO	TELL
	1	Configuration	INT16	O	RO	
	2	State	INT16	O	RO	
	3	Level	INT16	O	RO	
	4	-	INT16	O	RO	-
	5	-	INT16	O	RO	
3024h	0	Digital Input 4	INT16	O	RO	TELL
	1	Configuration	INT16	O	RO	
	2	State	INT16	O	RO	
	3	Level	INT16	O	RO	
	4	-	INT16	O	RO	-
	5	-	INT16	O	RO	
3030h	0	Function Digital Output	INT16	O	RO	設 定
	1	Enable Function	INT16	O	RO	
	2	State Function	INT16	O	RO	
	3	Level Function	INT16	O	RO	
	4	Digital Low Level	INT16	O	RO	
	5	-	INT16	O	RO	
	6	-	INT16	O	RO	予約領域
3031h	0	Digital Output 1	INT16	O	RO	設 定
	1	Configuration	INT16	O	RO	
	2	State	INT16	O	RO	
	3	Level	INT16	O	RO	予約領域
	4	-	INT16	O	RO	
	5	-	INT16	O	RO	
3032h	0	Digital Output 2	INT16	O	RO	設 定
	1	Configuration	INT16	O	RO	
	2	State	INT16	O	RO	
	3	Level	INT16	O	RO	予約領域
	4	-	INT16	O	RO	
	5	-	INT16	O	RO	
3033h	0	Digital Output 3	INT16	O	RO	設 定
	1	Configuration	INT16	O	RO	
	2	State	INT16	O	RO	
	3	Level	INT16	O	RO	予約領域
	4	-	INT16	O	RO	
	5	-	INT16	O	RO	
3034h	0	Digital Output 4	INT16	O	RO	設 定
	1	Configuration	INT16	O	RO	
	2	State	INT16	O	RO	
	3	Level	INT16	O	RO	-
	4	-	INT16	O	RO	
	5	-	INT16	O	RO	

注) 3030h ~ 3034h : FW Ver.2.3.1 から追加

【7】 CANopen オブジェクトリスト

Index	Sub.	名 称	タイプ	O/M	属 性	備 考
3050h	0	Analog Output 1	INT16	O	RO	TELL
	1	Configuration	INT16	O	R/W	
	2	Value	INT16	O	R/W	
	3	Output	INT16	O	RO	予約領域
	4	-	INT16	O	RO	
5	-	INT16	O	RO		
3100h	0	Configuration 1	INT16	O	R/W	
3200h	0	Current PID	INT16	M	R/W	設 定
	1	PidCur Kp	INT16	M	R/W	
	2	PidCur Ki	INT16	M	R/W	
	3	PidCur Kv	INT16	M	R/W	予約領域
	4	PidCur Kd	INT16	M	R/W	
	5	PidCur N	INT16	M	R/W	
3201h	6	PidCur FF	INT16	M	R/W	
	0	Speed PID	INT16	M	R/W	設 定
	1	PidVel Kp	INT16	M	R/W	
	2	PidVel Ki	INT16	M	R/W	
	3	PidVel Kv	INT16	M	R/W	予約領域
	4	PidVel Kd	INT16	M	R/W	
	5	PidVel N	INT16	M	R/W	
6	PidVel FF	INT16	M	R/W		
3300h	0	Velocity Full Scale	UINT16	O	R/W	設 定
4000h	0	Number of Entries	UINT16	O	RO	
	1	Safety State	UINT16	O	RO	
	2	STO Function	UINT16	O	RO	
4100h	0	Status Communication CAN	INT16	O	RO	TELL
	1	TEC Register	INT16	O	RO	
	2	REC Register	INT16	O	RO	
	3	Actual Flags Error	INT16	O	RO	
	4	Actual State Machine Communication	INT16	O	RO	
4101h	5	Actual State CAN Controller	INT16	O	RO	
	0	Counter Communication CAN	INT16	O	RO	
	1	Counter Bus Off	INT16	O	RO	
4102h	2	Counter Recovery Bus Off	INT16	O	RO	
	0	Settings Communication CAN	INT16	O	RO	設 定
	1	Sensibility CAN Error	INT16	O	R/W	
2	Timeout Error Passive	INT16	O	R/W		
4103h	0	Error Communication CAN	INT16	O	RO	TELL
	1	Last Error Flag	INT16	O	RO	
	2	All Error Flags	INT16	O	RO	
4200h	0	Alarm Monitoring	INT16	O	RO	TELL
	1	Last Error Code	INT16	O	RO	
	2	Manufacturer Code	INT16	O	RO	
	3	Manufacturer Code	INT16	O	RO	
DSP402 標準オブジェクト						
6007h	0	Abort Connection Option Code	UINT16	O	RO	DS402 状態機械 アラーム
603Fh	0	Error Code	UINT16	O	RO	
6040h	0	Control Word	UINT16	M	R/W	DS402 状態機械
6041h	0	Status Word	UINT16	M	RO	
605Ah	0	Quick Stop Option Code	INT16	O	RO	
605Bh	0	Shutdown Option Code	INT16	O	RO	
605Ch	0	Disable Option Code	INT16	O	RO	
605Dh	0	Halt Option Code	INT16	O	RO	
605Eh	0	Fault Reaction Code	INT16	O	RO	
6060h	0	Modes of Operation	INT8	M	R/W	
6061h	0	Modes of Operation Display	INT8	M	RO	
6062h	0	Position Demand Value	INT32	O	RO	使用不可
6063h	0	Position Actual internal Value	INT32	O	RO	
6064h	0	Position Actual Value	INT32	M	RO	設 定

注) 3100h、4101h～4103h、4200h : FW Ver.2.3.1 から追加

【7】 CANopen オブジェクトリスト

Index	Sub.	名 称	タイプ	O/M	属 性	備 考
6065h	0	Following Error Windows	UINT32	O	R/W	使用不可
6066h	0	Following Error Timeout	UINT16	O	R/W	
6067h	0	Position Windows	UINT32	O	R/W	
6068h	0	Position Window Time	UINT16	O	R/W	速度プロファイル
6068h	0	Velocity Demand Value	INT32	O	RO	
606Ch	0	Velocity Actual Value	INT32	M	RO	
606Dh	0	Velocity Window	UINT16	O	R/W	
606Eh	0	Velocity Window Time	UINT16	O	R/W	
606Fh	0	Velocity Threshold	UINT16	O	R/W	
6070h	0	Velocity Threshold Time	UINT16	O	R/W	トルクプロファイル
6071h	0	Target Torque	INT16	M	R/W	
6072h	0	Max Torque	UINT16	O	R/W	使用不可
6073h	0	Max Current	UINT16	O	R/W	トルクプロファイル
6074h	0	Torque Demand	INT16	O	RO	
6075h	0	Motor Rated Current	UINT32	O	R/W	使用不可
6076h	0	Motor Rated Torque	UINT32	O	R/W	トルクプロファイル
6077h	0	Torque Actual Value	INT16	O	RO	
6078h	0	Current Actual Value	INT16	O	RO	TELL
6079h	0	DC Link Circuit Voltage	UINT32	O	RO	
607Ah	0	Target Position	INT32	M	R/W	使用不可
607Bh	0	Position Range Limit	INT32	O	R/W	
607Ch	0	Home Offset	INT32	O	R/W	
607Dh	0	Software Position Limit	INT32	O	R/W	
607Eh	0	Polarity	UINT8	O	R/W	速度プロファイル
607Fh	0	Max Profile Velocity	UINT32	O	R/W	
6080h	0	Max Motor Speed	UINT32	O	R/W	使用不可
6081h	0	Profile Velocity	UINT32	M	R/W	
6082h	0	End Velocity	UINT32	O	R/W	速度プロファイル
6083h	0	Profile Acceleration	UINT32	O	R/W	
6084h	0	Profile Deceleration	UINT32	O	R/W	DS402 状態機械
6085h	0	Quick Stop Deceleration	UINT32	O	R/W	
6086h	0	Motion Profile Type	INT16	O	R/W	速度プロファイル
6087h	0	Torque Slope	UINT32	M	R/W	トルクプロファイル
6088h	0	Torque Profile Type	INT16	O	R/W	
6096h	0	Velocity Factor Group	UINT32	O	R/W	設 定
	1	Num Velocity Factor	UINT32	O	R/W	
	2	Den Velocity Factor	UINT32	O	R/W	
6097h	0	Acceleration Factor Group	UINT32	O	R/W	
	1	Num Acceleration Factor	UINT32	O	R/W	
	2	Den Acceleration Factor	UINT32	O	R/W	
60C5h	0	Max Acceleration	UINT32	O	R/W	速度プロファイル
60C6h	0	Max Deceleration	UINT32	O	R/W	
60E0h	0	Positive Torque Limit Value	UINT16	O	R/W	トルクプロファイル
60E1h	0	Negative Torque Limit Value	UINT16	O	R/W	速度プロファイル
60FDh	0	Digital Inputs	UINT32	O	RO	TELL
	0	Digital Outputs	UINT32	O	RO	設 定
	1	Physical Outputs	UINT32	O	R/W	
60FEh	2	Bit Mask	UINT32	O	R/W	
	0	Target Velocity	INT32	M	R/W	速度プロファイル
6402h	0	Motor Type	UINT16	O	R/W	TELL
6403h	0	Motor Catalogue Number	STRING	O	R/W	
6404h	0	Motor Manufacturer	STRING	O	R/W	
6502h	0	Supported Drive Modes	UINT32	M	RO	

注) 6502h : FW Ver.2.3.1 から追加

8-1 過速度検出

最高速度制限の 10%以上で、過速度のアラームが発生します。

8-2 デジタル I/O

以下のデジタル I/O が使用可能です。

- デジタル入力: DIG-INx
- デジタル出力: DIG-OUTx
- セーフティデジタル入力: DIG-STO

初期設定を次に示します。

- DIG-IN1：予約領域
- DIG-IN2：予約領域
- DIG-IN3：緊急停止
ダイナミックブレーキでモータを緊急停止させます。
ドライバは、RUN から STANDBY ステートへ移行します。
- DIG-IN4：リセット
ドライバのハードウェアリセット

■デジタル出力

- DIG-OUT1：ドライバの状態
 - 1：ドライバ正常 (Drive OK)
 - 0：ドライバ異常 (Fault)
- DIG-OUT2：警告の有無
 - 1：警告あり
 - 0：警告なし
- DIG-OUT3：ドライバ運転状態
 - 1：運転可能
 - 0：運転不可
- DIG-OUT4：ブレーキ状態
 - 1：ブレーキ解放 (モータ軸フリー)
 - 0：ブレーキ動作 (モータ軸ロック)

【8】機能

■デジタル入力：セーフティ STO

STO セーフティ機能を利用するための端子です。

STO が動作した(電圧供給が遮断された)場合、ドライバは SAFETY ステートへ移行します。
異常を検出した場合、FAULT ステートへ移行します。

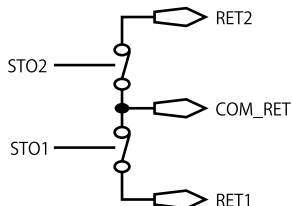


図 8-1 STO 出力フィードバックリレー

以下の図は、STO の状態機械を示しています。

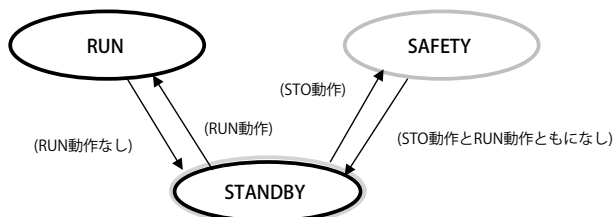


図 8-2 STO 状態機械の遷移

⚠ 注意

SAFETY ステートから STANDBY ステートへ復帰する際には、STO と RUN 信号がともになし(OFF)とする必要があります。以下は、SAFETY ステートから復帰するまでに辿るドライバの状態機械の遷移です。

- CANopen : SAFETY → “SWITCH ON DISABLED” → “READY TO SWITCH ON” → “SWITCH ON” → “OPERATION ENABLED”

⚠ 注意

STO 機能が動作すると、トルクが発生しないフリーラン状態となるため、負荷を保持することができません。

- 運転中に STO 機能を動作させると、モータがフリーランとなるため危険です。
- STO 機能が搭載されたドライバを使用する場合、運転前に STO 回路が正しく動作することを確認してください。

8-3 他の機能

次に CANopen の追加機能の命令を示しています。

- ・緊急停止入力の有効化 (Emergency Digital Input Enable)
- ・セーフティ状態の確認 (Safety)
- ・異常履歴の確認 (Fault History)
- ・ダイナミックブレーキ (Dynamic Brake)
- ・モータブレーキマネジメント (Motor Brake Management)
- ・DAC モニタリング (DAC Monitoring)

■緊急停止入力の有効化 (Emergency Digital Input Enable)

緊急停止入力は、デジタル入力 3 (DIG-IN3 端子)を使用します。

以下のように「緊急停止有効パラメータ」(3008h: 1)へ 1 を設定すると緊急停止入力が有効となります。

Tx/Rx	ID	数 値	内 容
Rx	0x601	2b 08 30 01 01 00 00 00	“Emergency Enable Input” の書込み：有効
Tx	0x581	60 08 30 01 00 00 00 00	-

注)「緊急停止パラメータ」(3008h: 1)の初期値は、1 で有効設定となっています。

上位コントローラは、ドライバのデジタル入力 3 を使用することでモータの緊急停止動作が可能です。

ドライバのデジタル入力 3 を ON した場合、RUN ステート “Operation Enabled” から STANDBY ステート “Switched ON” へ移行します。

ダイナミックブレーキの設定(3007h: 1)を有効にしている場合、ランプで減速停止します。デジタル入力 3 が ON 状態で保持されていると RUN ステートへ移行できません。

⚠ 注 意

この緊急停止動作は、安全機能ではありません。安全に RUN ステートへ復帰するためには、追加の保護回路が必要です。

緊急停止入力が ON の場合、「ステータスワード」(6041h)のビット 8 が 1 となり、状態機械が “Switched ON” (STANDBY) になります。

上位コントローラは、ステータスワード(6041h)が “xxxx xxx1 x01x 0011b” になることを確認

Tx/Rx	ID	数 値	内 容
Rx	0x601	40 41 60 00 00 00 00 00	ステータスワードを讀出し
Tx	0x581	4b 41 60 00 33 15 00 00	“Switched ON” 状態と緊急停止入力 ON 状態 1533h = xxxx xxx1 x01x 0011b

「緊急停止入力状態モニタ」(3008h: 3)で緊急停止入力 “Emergency Input Enable” の状態を讀出し

Tx/Rx	ID	数 値	内 容
Rx	0x601	40 08 30 03 00 00 00 00	“Emergency Input Enable” の状態を讀出し
Tx	0x581	4b 08 30 03 01 00 00 00	01h：緊急停止入力動作中

【8】機能

「緊急停止入力」(3008h: 2)で入力を受付ける論理の変更

Tx/Rx	ID	数 値	内 容
Rx	0x601	2b 08 30 02 01 00 00 00	01h : b 接点へ設定を変更 (初期値は 00h : a 接点)
Tx	0x581	60 08 30 02 00 00 00 00	正常に書き込み完了

■セーフティ状態の確認

以下は、セーフティ状態の確認手順です。

ドライバの STO1 端子または STO2 端子へ DC24V の供給を遮断すると STO 入力が動作し、セーフティステート (SAFETY) へ移行します。

上位コントローラでステータスワード (6041h) が “x1xx xxxx xxxx xxxxb” になることを確認

Tx/Rx	ID	数 値	内 容
Rx	0x601	40 41 60 00 00 00 00 00	ステータスワードを読み出し
Tx	0x581	4b 41 60 00 23 40 00 00	4023h = x1xx xxxx xxxx xxxxb ビット 14 0b : セーフティ OFF、1b : セーフティ ON

または、「セーフティステート」(4000h: 1)で確認

Tx/Rx	ID	数 値	内 容
Rx	0x601	40 00 40 01 00 00 00 00	セーフティステートの読み出し
Tx	0x581	4b 00 40 01 01 00 00 00	0b : セーフティ OFF、1b : セーフティ ON

または、「ドライバステータス」(2002h: 0)を確認

Tx/Rx	ID	数 値	内 容
Rx	0x601	40 02 20 00 00 00 00 00	“Drive Control State” の読み出し
Tx	0x581	4b 02 20 00 80 00 00 00	Value = 0x80 は、STO 機能が ON 状態で SAFETY ステートであることを示します。

■異常履歴の確認 (Fault History)

以下は、異常履歴の確認手順です。

ドライバに記憶された発生した異常の数を「異常履歴」(1003h: 0)で読み出し

Tx/Rx	ID	数 値	内 容
Rx	0x601	40 03 10 00 00 00 00 00	異常の数の読み出し
Tx	0x581	4F 03 10 00 02 00 00 00	Value = 0x02 は、2 つの異常が記録されていることを示します。

「異常履歴」(1003h: 0)に 0 を書き込むことで異常履歴の消去

Tx/Rx	ID	数 値	内 容
Rx	0x601	22 03 10 00 00 00 00 00	異常履歴を消去
Tx	0x581	60 03 10 00 00 00 00 00	正常に書き込み完了

「異常履歴」(1003h: 1 ~ 15)でエラーメッセージコードを確認

Tx/Rx	ID	数 値	内 容
Rx	0x601	40 03 10 01 00 00 00 00	エラーメッセージの読み出し
Tx	0x581	43 03 10 01 50 23 00 00	CANopen からの応答 (0x2350 : 過負荷異常)

注) エラーメッセージコードは、2 章の「表 2-31 ~ 36 エマージェンシ内容」に記載されています。

異常履歴が記録されていない Sub-index 値を指定してエラーメッセージコードを確認した場合、アボートコード 0x08000024 を受信します。

Tx/Rx	ID	数 値	内 容
Rx	0x601	40 03 10 08 00 00 00 00	エラーメッセージの読出し (Sub-index: 8)
Tx	0x581	80 03 10 08 24 00 00 08	CANopen からの応答

■ダイナミックブレーキ

ダイナミックブレーキ機能を有効に設定している場合、ランプで減速していき“Operation Enabled” (RUN ステート) から抜けます。「ダイナミックブレーキパラメータ」(3007h: 4) で設定した時間に従って減速します。減速が終了して停止すると、ブレーキが動作してモータがロックされ、“Switched ON” (STANDBY ステート) に変わります。

ダイナミックブレーキ機能を無効に設定している場合、フリーラン減速します。

モータ速度(回転数)が0になると、ブレーキが動作してロックされますが、「ブレーキタイムアウト時間」(3002h: 4) で設定した時間を超えると、強制的にブレーキが動作します。

アプリケーションによりパラメータの設定値が変更可能です。

パラメータを変更するには「ダイナミックブレーキパラメータ」(3007h) の各 Sub-index へ SDO で新たな値を書込んだ後、「パラメータストア」(1010h) に “save” を書込むことで、EEPROM へ保存できます。

ダイナミックブレーキを動作させるため、「ダイナミックブレーキオプション」(3007h: 1) へ 1 を書き込み

Tx/Rx	ID	数 値	内 容
Rx	0x601	2B 07 30 01 01 00 00 00	ダイナミックブレーキオプション有効
Tx	0x581	60 07 30 01 00 00 00 00	-

例：ステップランプ減速(Decrement Step Ramp)に 100 「[rpm × 100]/sec」を設定

Tx/Rx	ID	数 値	内 容
Rx	0x601	2B 07 30 04 64 00 00 00	ステップランプ減速へ書き込み
Tx	0x581	60 07 30 04 00 00 00 00	-

EEPROM へパラメータをストア。

Tx/Rx	ID	数 値	内 容
Rx	0x601	23 10 10 01 73 61 76 65	全パラメータのストア
Tx	0x581	60 10 10 01 00 00 00 00	-

全ノードをリセット

Tx/Rx	ID	数 値	内 容
Rx	0x00	81 00	全ノードリセット命令
Tx	0x701	00	ブートアップ
Tx	0x81	00 00 00 00 00 00 00 00	エマージェンシブプロトコル=エラーなし

【8】 機能

■モータブレーキマネジメント (Motor Brake Management)

モータブレーキへ外部機器から電源を供給して動作させる場合に変更してください。

この機能を有効にするため、「モータブレーキオプション」(3002h: 1)へ1を書込み

Tx/Rx	ID	数 値	内 容
Rx	0x601	2B 02 30 01 01 00 00 00	ブレーキ有効
Tx	0x581	60 02 30 01 00 00 00 00	-

EEPROM へのパラメータのストア

Tx/Rx	ID	数 値	内 容
Rx	0x601	23 10 10 01 73 61 76 65	全パラメータのストア
Tx	0x581	60 10 10 01 00 00 00 00	-

全ノードをリセット

Tx/Rx	ID	数 値	内 容
Rx	0x00	81 00	全ノードリセット命令
Tx	0x701	00	ブートアップ
Tx	0x81	00 00 00 00 00 00 00 00	エマージェンシプロトコル=エラーなし

使用するブレーキの仕様を「モータブレーキタイプ」(3002h: 7)で確認できます。

Tx/Rx	ID	数 値	内 容
Rx	0x601	40 02 30 07 00 00 00 00	ブレーキタイプの読出し
Tx	0x581	4B 02 30 07 01 00 00 00	1 : PM ブレーキ

モータブレーキは、自動モードと手動モードに設定を切替えることができます。

- ・自動モード：ブレーキは、ドライバが“Operation Enabled”(RUN ステート)に設定・遷移した場合、自動的に解放(モータブレーキへ DC24V が供給)されます。

- ・手動モード：「デジタル出力」(60FEh: 1)からブレーキを動作させることができます。

手動モードに設定するには、「自動/手動モード設定」(3002h: 5)に1を書込みます。

Tx/Rx	ID	数 値	内 容
Rx	0x601	2B 02 30 05 01 00 00 00	3002h: 5 へ1を書込み (1 : 手動モード)
Tx	0x581	60 02 30 05 00 00 00 00	-

ブレーキを動作させるため、「デジタル出力」(60FEh: 1)に0を書込みます。

Tx/Rx	ID	数 値	内 容
Rx	0x601	23 FE 60 01 00 00 00 00	0 : ブレーキ動作命令
Tx	0x581	60 FE 60 01 00 00 00 00	ブレーキ解放 → モータ軸ロック

ブレーキを開放させるため、「デジタル出力」(60FEh: 1)に1を書込みます。

Tx/Rx	ID	数 値	内 容
Rx	0x601	23 FE 60 01 01 00 00 00	1 : ブレーキ解放命令
Tx	0x581	60 FE 60 01 00 00 00 00	モータ軸ロック → ブレーキ解放

■ DAC モニタリング

アナログ出力をモニタとして設定できます。オブジェクトは 3050h です。

アナログ出力 (3050h: 1) = 0 で無効 (出荷時設定)

Tx/Rx	ID	数 値	内 容
Rx	0x601	2B 50 30 01 00 00 00 00	0 : アナログ出力 無効の設定
Tx	0x581	60 50 30 01 00 00 00 00	

アナログ出力を汎用 (General Purpose) として設定

Tx/Rx	ID	数 値	内 容
Rx	0x601	2B 50 30 01 01 00 00 00	1 : 汎用 (General Purpose) の設定
Tx	0x581	60 50 30 01 00 00 00 00	

アナログ出力 (3050h: 2) に数値を設定し、出力を読むことができます。

例 : アナログ出力 (3050h: 2) に 0x0800 (=2048) を書込み

Tx/Rx	ID	数 値	内 容
Rx	0x601	2B 50 30 02 00 08 00 00	数値 = 0x0800
Tx	0x581	60 50 30 02 00 00 00 00	

アナログ出力の読出し「アナログ出力 (3050h: 3)」

Tx/Rx	ID	数 値	内 容
Rx	0x601	40 50 30 03 00 00 00 00	アナログ出力の読出し 0x0800 = 2048 ビット : 5V 出力
Tx	0x581	4B 50 30 03 00 08 00 00	

アナログ出力を速度モニタとして設定

Tx/Rx	ID	数 値	内 容
Rx	0x601	2B 50 30 01 02 00 00 00	2 : 速度モニタの設定
Tx	0x581	60 50 30 01 00 00 00 00	

アナログ出力の読出し「アナログ出力 (3050h: 3)」

Tx/Rx	ID	数 値	内 容
Rx	0x601	40 50 30 03 00 00 00 00	アナログ出力の読出し 0x0FFF = 4095 ビット : 10V 出力 (10V 出力 : 最高回転数)
Tx	0x581	4B 50 30 03 FF 0F 00 00	

【9】 診断

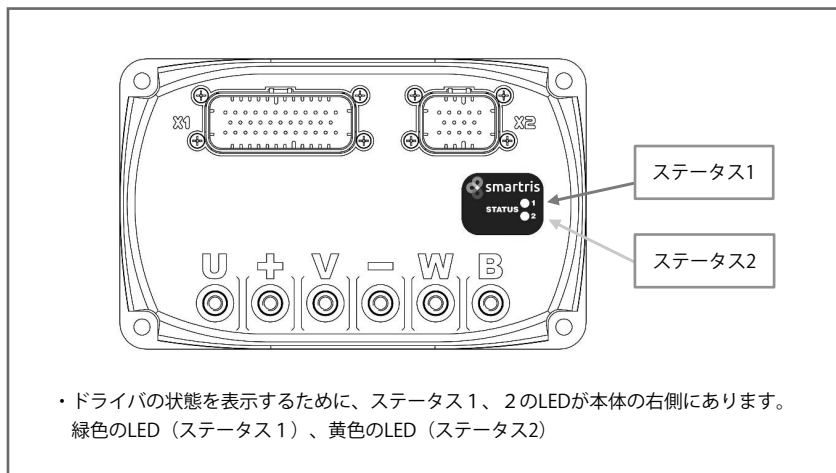


表 9-1 ステータス LED

ドライバ状態	CANopen 状態	ステータス 1 LED（緑色）	ステータス 2 LED（黄色）	LED 表示
初期状態 (INIT)	スイッチオン 準備中 (Not Ready to Switch ON)	“点滅”	“点滅”	<div> <div></div> 1 同時に点滅 <div></div> 2 同時に点滅 </div>
	スイッチオン 無効 (Switch ON Disabled) スイッチオン 準備完了 (Ready to Switch ON)	交互に “点滅”	交互に “点滅”	<div> <div></div> 1 交互に点滅 <div></div> 2 交互に点滅 </div>
運転準備 (STANDBY)	スイッチオン (Switched ON)	“点滅”	OFF	<div> <div></div> 1 点滅 50% <div></div> 2 消灯 </div>
故 障 (FAULT)	故障 (Fault) 故障処理中 (Fault Reaction Fault)	“点滅” コード [x]	“点滅” コード [y]	<div> <div></div> 1 表 9-2 参照 <div></div> 2 </div>
運転 (RUN)	運転有効 (Operation Enabled)	ON	OFF	<div> <div></div> 1 点灯 <div></div> 2 消灯 </div>
停 止 (STOP)	急停止動作 (Quick Stop Active)	ON	ON	<div> <div></div> 1 点灯 <div></div> 2 点灯 </div>
セーフティ (SAFETY)	-	OFF	“点滅”	<div> <div></div> 1 消灯 <div></div> 2 点滅 </div>
通信エラー (COMMUNICATION ERROR)	-	OFF	ON	<div> <div></div> 1 消灯 <div></div> 2 点灯 </div>

表 9-2 アラーム一覧

分 類	アラーム	ステータス 1 LED (緑色)	ステータス 2 LED (黄色)	アラームの内容
		⚙️ コード [x]	⚙️ コード [y]	
A 温 度	モータ過熱	1	10	モータ温度が設定温度を超過 モータ温度が高いために運転不可
	ヒートシンク過熱		1	ヒートシンク温度が設定値を超過 ヒートシンク温度が高いために運転不可
	ヒートシンク温度 範囲外		3	ヒートシンクの温度センサが測定範囲外 温度センサの動作不良
	プリント板過熱		4	プリント板の温度が設定値を超過 プリント板の温度が高いために運転不可
	プリント板温度 範囲外		5	プリント板の温度センサが測定範囲外 温度センサの動作不良
	モータ過熱範囲外		6	モータの温度センサが測定範囲外 温度センサの動作不良
B フィード バック	レゾルバ	2	10	レゾルバのコネクタ、配線の確認
	レゾルバ初期化		4	レゾルバの初期化エラー
	アブソリュート エンコーダ		6	アブソリュートエンコーダの故障
C 電 流	電流センサの オフセット	3	10	電流センサのオフセットが範囲外
	過電流		1	モータ過電流 モータの配線、短絡の確認
D 電 圧	不足電圧	4	1	DC バスの電圧が設定値以下 ＋の電源端子電圧を確認
	過電圧		2	DC バスの電圧が設定値以上 ＋の電源端子電圧を確認
E 機能性	速度エラー	5	10	速度指令と実際の速度との誤差大
	過負荷保護 (I ² T)		2	モータ過負荷保護 (I ² T)
	ハードウェア		3	ハードウェアエラー
	外部ハードウェア		4	CANopen インターフェースエラー
	オーバースピード		8	CANopen 通信での過速度エラー
F 通 信	EEPROM	6	1	EEPROM に記憶されているパラメータ異常
	CANopen		2	CANopen の通信エラー
	アブソリュート エラー		3	内部の通信エラー
	パラメータ初期化		4	パラメータ初期化異常
	プロファイル		5	プロファイル設定エラー
	トルクプロファイル		6	トルクプロファイルエラー
	速度プロファイル		7	速度プロファイルエラー
	ホーミング プロファイル		8	ホーミングプロファイルエラー
G プログラミング	プログラムエラー	7	×	プログラムコードのエラー
	出荷パラメータ	8	1	出荷設定パラメータのプログラミングエラー
	CANopen パラ メータ		2	CANopen パラメータのプログラミング エラー
L ポジショナ	ポジショナ	10	1	ポジショナプロファイルエラー

【10】 サンプルプログラム

本章は、CANopen 通信で運転を開始するための簡易的なプログラムおよびシーケンスを示しています。

以下のサンプルプログラムは、全てドライバのノード ID を 1 に設定しています。
CAN メッセージの数値は、以下の表に対応しています。

	Cmd		Index	Sub-index	数 値			
バイト	0	1	2	3	4 (LSB)	5	6	7 (MSB)
数 値	2F	60	60	00	03	00	00	00

例：2F 60 60 00 03 00 00 00 は、SDO ダウンロード(書込み)です。

バイト 0：“2Fh”は、1 バイトの「書込み要求」命令です。

バイト 1、2：“6060h”は、運転モード Index 0x6060 を表しています。

バイト 3：“00h”は、Sub-index 0x00 を表しています。

バイト 4、5、6、7：0x00000003 は、速度プロファイルモード= 3 を表しています。

10-1 速度プロファイル手順の設定

⚠ 注意

ドライバが“Operation Enabled”ステート以外の状態で、速度プロファイルモードを設定してください。

・運転モードの設定

Tx/Rx	ID	数 値	内 容
Rx	0x601	2F 60 60 00 03 00 00 00	リクエスト：速度プロファイルの運転モード番号の設定
Tx	0x581	60 60 60 00 00 00 00 00	アンサ：成功

・“Switched ON”ステートへ移行

Tx/Rx	ID	数 値	内 容
Rx	0x601	2B 40 60 00 06 00 00 00	リクエスト：“Ready to Switch ON”ステートに変更
Tx	0x581	60 40 60 00 00 00 00 00	アンサ：リクエスト成功
Rx	0x601	2B 40 60 00 07 00 00 00	リクエスト：“Switched ON”ステートに変更
Tx	0x581	60 40 60 00 00 00 00 00	アンサ：成功
Rx	0x601	40 41 60 00 00 00 00 00	リクエスト：SDO ステータスワードの読出し
Tx	0x581	4B 41 60 00 33 10 00 00	アンサ：ステータスワード 0x1033 “Switched ON” = xxxxxxxx01x0011b

・加減速の設定

単位変換 (p11-6 を参照) を実施した後に、以下のコマンドが有効となります。

Tx/Rx	ID	数 値	内 容
Rx	0x601	23 83 60 00 E8 03 00 00	リクエスト：加速度設定 1000 rpm/s
Tx	0x581	60 83 60 00 00 00 00 00	アンサ：成功
Rx	0x601	23 84 60 00 E8 03 00 00	リクエスト：減速度設定 1000 rpm/s
Tx	0x581	60 84 60 00 00 00 00 00	アンサ：成功

【10】 サンプルプログラム

・“Operation Enabled” ステートへの移行

Tx/Rx	ID	数 値	内 容
Rx	0x601	2B 40 60 00 0F 00 00 00	リクエスト：“Operation Enabled” ステートに変更
Tx	0x581	60 40 60 00 00 00 00 00	アンサ：成功
Rx	0x601	40 41 60 00 00 00 00 00	リクエスト：SDO ステータスワードの読出し
Tx	0x581	4B 41 60 00 37 12 00 00	アンサ：ステータスワード 0x1237 “Operation Enabled” = xxxxxxxx01x0111b

・目標速度の設定

単位変換 (p.11-6 を参照) を実施した後に、以下のコマンドが有効となります。

Tx/Rx	ID	数 値	内 容
Rx	0x601	23 FF 60 00 E8 03 00 00	リクエスト：速度設定 1000 rpm
Tx	0x581	60 FF 60 00 00 00 00 00	アンサ：成功
Rx	0x601	40 6C 60 00 00 00 00 00	リクエスト：速度検出値を読出し
Tx	0x581	43 6C 60 00 E8 03 00 00	アンサ：速度 0x03E8 = 1000 rpm.
Rx	0x601	23 FF 60 00 00 00 00 00	リクエスト：目標速度を 0 (停止) に設定 (ランプで減速)
Tx	0x581	60 FF 60 00 00 00 00 00	アンサ：成功

・その他の停止方法

コントロールワードによる停止 (Operation Enable -> Switched ON)

Tx/Rx	ID	数 値	内 容
Rx	0x601	2B 40 60 00 07 00 00 00	リクエスト：“Switched ON” を設定。トルク発生なし。
Tx	0x581	60 40 60 00 00 00 00 00	アンサ：成功
Rx	0x601	40 41 60 00 00 00 00 00	リクエスト：SDO ステータスワードを読出し
Tx	0x581	4B 41 60 00 33 16 00 00	アンサ：ステータスワード 0x1633 “Switched ON” = xxxxxxxx01x0011b.

コントロールワードによる停止 (Operation Enable -> Switched ON Disabled)

Tx/Rx	ID	数 値	内 容
Rx	0x601	2B 40 60 00 00 00 00 00	リクエスト：“Switched ON Disabled” を設定。トルク発生なし。
Tx	0x581	60 40 60 00 00 00 00 00	アンサ：成功
Rx	0x601	40 41 60 00 00 00 00 00	リクエスト：ステータスワードの読込み
Tx	0x581	4B 41 60 00 50 16 00 00	アンサ：ステータスワード 0x1650 “Switched ON Disabled” = xxxxxxxx1x0000b.

Quick Stop Active での停止

Tx/Rx	ID	数 値	内 容
Rx	0x601	2B 40 60 00 02 00 00 00	リクエスト：“Quick Stop Active” を設定
Tx	0x581	60 40 60 00 00 00 00 00	アンサ：成功
Rx	0x601	40 41 60 00 00 00 00 00	リクエスト：SDO ステータスワードを読出し
Tx	0x581	4B 41 60 00 17 16 00 00	アンサ：ステータスワード 0x1716 “Quick Stop Active” = xxxxxxxx00x0110b

【10】 サンプルプログラム

■ SDO プロトコルでのトレースログ(目標速度 1000 rpm)

ID	名称	ノード	送受信データ	説明	データ
701	HBGuard_001	Node1	Boot-up	-	00
81	EMCY_001	Node1	00 00 00 00 00 00 00 00	Error reset or no error	00 00 00 00 00 00 00 00
81	EMCY_001	Node1	70 81 11 00 00 00 00 00	Communication-generic	70 81 11 00 00 00 00 00
601	CSDO_001	Node1	03	[6060, 00] Initiate download Rq.	2F 60 60 00 03 00 00 00
581	SSDO_001	Node1	-	[6060, 00] Initiate download Rsp.	60 60 60 00 00 00 00 00
601	CSDO_001	Node1	06 00	[6040, 00] Initiate download Rq.	2B 40 60 00 06 00 00 00
581	SSDO_001	Node1	-	[6040, 00] Initiate download Rsp.	60 40 60 00 00 00 00 00
601	CSDO_001	Node1	07 00	[6040, 00] Initiate download Rq.	2B 40 60 00 07 00 00 00
581	SSDO_001	Node1	-	[6040, 00] Initiate download Rsp.	60 40 60 00 00 00 00 00
601	CSDO_001	Node1	-	[6041, 00] Initiate upload Rq.	40 41 60 00 00 00 00 00
581	SSDO_001	Node1	33 10	[6041, 00] Initiate upload Rsp.	4B 41 60 00 33 10 00 00
601	CSDO_001	Node1	0F 00	[6040, 00] Initiate download Rq.	2B 40 60 00 0F 00 00 00
581	SSDO_001	Node1	-	[6040, 00] Initiate download Rsp.	60 40 60 00 00 00 00 00
601	CSDO_001	Node1	E8 03 00 00	[60FF, 00] Initiate download Rq.	23 FF 60 00 E8 03 00 00
581	SSDO_001	Node1	-	[60FF, 00] Initiate download Rsp.	60 FF 60 00 00 00 00 00
601	CSDO_001	Node1	-	[606C, 00] Initiate upload Rq.	40 6C 60 00 00 00 00 00
581	SSDO_001	Node1	E7 03 00 00	[606C, 00] Initiate upload Rsp.	43 6C 60 00 E7 03 00 00

10-2 バージョンの読出し

ドライバのファームウェアバージョンを讀出す命令(オブジェクト 100Ah)

Tx/Rx	ID	数 値	内 容
Rx	0x601	40 0A 10 00 00 00 00 00	リクエスト：リリースされたファームウェアの讀出し アンサ：ASCII コード 0x31, 0x30, 0x38, 0x00="108" (ドライバのファームウェアバージョン：1.08)
Tx	0x581	43 0A 10 00 31 30 38 00	

10-3 PDO を介しての制御

■ RPDO

例：マスタコントローラから RPDO の初期値を設定

Tx/Rx	ID	数 値	内 容
Rx	0x201	06 00 03 00 00 00 00 00	RPDO1：全サイズ 7 バイト コントロールワード：0x0006 = "Ready to Switch ON" 運転モード：0x03、デジタル出力：0x00000000 を設定
Rx	0x301	07 00 A0 86 01 00	RPDO2：全サイズ 6 バイト コントロールワード：0x0007 = "Switched ON" 目標位置：0x186A0 = 100000 (インクリメント単位) を設定
Rx	0x401	0F 00 E8 03 00 00	RPDO3：全サイズ 6 バイト コントロールワード：0x000F = "Operation Enabled" 目標速度：0x000003E8 = 1000 rpm を設定
Rx	0x501	00 00 E8 03	RPDO4：全サイズ 4 バイト コントロールワード：0x0000 = "Disable Voltage" 目標トルク：0x003E8 = 1000 (モータ定格電流) を設定

【10】 サンプルプログラム

データ長が異なる場合、ドライバは、次のようなエマージェンシメッセージを送信します。

例：RPDO1 を DLC（データ長）= 8 で送信した場合

Time	Chn	Dr	ID	Name	Node	Transfer Data	Error	Interpretation	Data
542.029968	1	Tx	201	ID1_RPDO1	Master	07 00 03 86 01 00 00 00	-	-	07 00 03 86 01 00 00 00
Digital output_1						1 [1]			
Mode_of_operation_1						3 [3]			
Controlword_1						7 [7]			
542.030115	1	Rx	81	EMCY_001	Node1	20 82 11 00 00 00 00 00	E	PDO length exceeded	20 82 11 00 00 00 00 00

■ RPDO と TPDO（SYNC）

RPDO と TPDO に SYNC タイプの通信が定義されている場合、マスタコントローラは、SYNC メッセージの後に RPDO を送信する必要があります。

ドライバは、SYNC メッセージ後に TPDOs を送信します。

Tx/Rx	ID	数 値	内 容
Rx	0x201	0F 00 03 00 00 00 00 00	RPDO1：コントロールワード：0x000F = "Operation Enabled" 運転モード：0x03、デジタル出力：0x00000000 を設定
Rx	0x401	0F 00 E8 03 00 00 00 00	RPDO3：コントロールワード：0x000F = "Operation Enabled" 目標速度：0x000003E8 = 1000 rpm を設定
Rx	0x80	-	SYNC メッセージ
Tx	0x181	C0 02 03 00 00 00 00 00	TPDO1：ステータスワード：0x02C0、運転モード：0x03 デジタル入力：0x00000000 を設定
Tx	0x381	C0 02 00 00	TPDO3：ステータスワード：0x02C0 速度検出値：0x000003E8 = 1000 rpm を設定

■ RPDO、RTR タイプ

RPDO が RTR を設定している場合（マスタコントローラ）

Tx/Rx	ID	数 値	内 容
Rx	0x201	0F 00 03 00 00 00 00 00	RPDO1：コントロールワード：0x000F = "Operation Enabled", 運転モード：0x03、デジタル出力：0x00000000 を設定
Rx	0x401	0F 00 E8 03 00 00 00 00	RPDO3：コントロールワード：0x000F = "Operation Enabled" 目標速度：0x000003E8 = 1000 rpm を設定
Rx	0x181	RTR	TPDO1：ステータスワード：0x02C0、運転モード：0x03
Tx	0x181	C0 02 03 00 00 00 00 00	デジタル入力：0x00000000 を設定
Rx	0x381	RTR	TPDO3：ステータスワード：0x02C0
Tx	0x381	C0 02 E8 03 00 00 00 00	速度検出値：0x000003E8 = 1000 rpm を設定

【10】 サンプルプログラム

10-4 TPDO (RPDO) のリマッピング手順

次のマップで TPDO4 (1803h) をリマッピングします。

- ・ ポジション 1：ステータスワード (6041h)
- ・ ポジション 2：速度検出値 (606Ch)

Tx/Rx	ID	数 値	内 容
Rx	0x601	23 03 18 01 00 00 00 80	リクエスト：TPDO4 通信の無効
Tx	0x581	60 03 18 01 00 00 00 00	アンサ：成功
Rx	0x601	2F 03 1A 00 00 00 00 00	リクエスト：TPDO4 マッピングパラメータ無効
Tx	0x581	60 03 1A 00 00 00 00 00	アンサ：成功
Rx	0x601	2F 03 18 02 01 00 00 00	リクエスト：送信タイプ SYNC の設定
Tx	0x581	60 03 18 02 00 00 00 00	アンサ：成功
Rx	0x601	23 03 1A 01 10 00 41 60	リクエスト：ポジション 1 のマップ（ステータスワード）
Tx	0x581	60 03 1A 01 00 00 00 00	アンサ：成功
Rx	0x601	23 03 1A 02 20 00 6C 60	リクエスト：ポジション 2 のマップ（速度検出値）
Tx	0x581	60 03 1A 02 00 00 00 00	アンサ：成功
Rx	0x601	2F 03 1A 00 02 00 00 00	リクエスト：マップされたオブジェクト数（2）を書込み
Tx	0x581	60 03 1A 00 00 00 00 00	アンサ：成功
Rx	0x601	23 03 18 01 81 04 00 00	リクエスト：TPDO4 有効で COB-ID の設定
Tx	0x581	60 03 18 01 00 00 00 00	アンサ：成功

通信オブジェク (1803h) と TPDO マッピングパラメータ (1A03h) の読出し

Tx/Rx	ID	数 値	内 容
Rx	0x601	40 03 18 00 00 00 00 00	リクエスト：1803:0 を読出し
Tx	0x581	4F 03 18 00 05 00 00 00	アンサ：Sub-index 数、数値＝ 5
Rx	0x601	40 03 18 01 00 00 00 00	リクエスト：1803:1 を読出し
Tx	0x581	43 03 18 01 81 04 00 00	アンサ：COB-ID = 0x00000481
Rx	0x601	40 03 18 02 00 00 00 00	リクエスト：1803:2 を読出し
Tx	0x581	4F 03 18 02 01 00 00 00	アンサ：Tx タイプ：Sync 1
Rx	0x601	40 03 18 03 00 00 00 00	リクエスト：1803:3 を読出し
Tx	0x581	4B 03 18 03 05 00 00 00	アンサ：禁止時間（Inhibit time）：数値＝ 5
Rx	0x601	40 03 18 04 00 00 00 00	リクエスト：1803:4 を読出し
Tx	0x581	80 03 18 04 11 00 09 06	アンサ：アバートコード Sub-index なし（未実装のため）
Rx	0x601	40 03 18 05 00 00 00 00	リクエスト：1803:5 を読出し
Tx	0x581	4B 03 18 05 00 00 00 00	アンサ：イベント時間＝ 0
Rx	0x601	40 03 1A 00 00 00 00 00	リクエスト：1A03:0 を読出し
Tx	0x581	4F 03 1A 00 02 00 00 00	アンサ：マップされたオブジェクト数：数値＝ 2
Rx	0x601	40 03 1A 01 00 00 00 00	リクエスト：1A03:1 を読出し
Tx	0x581	43 03 1A 01 10 00 41 60	アンサ：マップされたオブジェクト 1：ステータスワード 0x6041
Rx	0x601	40 03 1A 02 00 00 00 00	リクエスト：1A03:2 を読出し
Tx	0x581	43 03 1A 02 20 00 6C 60	アンサ：マップされたオブジェクト 2：速度検出値 0x606C
Rx	0x601	40 03 1A 03 00 00 00 00	リクエスト：1A03:3 を読出し
Tx	0x581	43 03 1A 03 00 00 00 00	アンサ：マップされたオブジェクト 3：なし
Rx	0x601	40 03 1A 04 00 00 00 00	リクエスト：1A03:4 を読出し
Tx	0x581	43 03 1A 04 00 00 00 00	アンサ：マップされたオブジェクト 4：なし
Rx	0x601	40 03 1A 05 00 00 00 00	リクエスト：1A03:5 を読出し
Tx	0x581	43 03 1A 05 00 00 00 00	アンサ：マップされたオブジェクト 5：なし
Rx	0x601	40 03 1A 06 00 00 00 00	リクエスト：1A03:6 を読出し
Tx	0x581	43 03 1A 06 00 00 00 00	アンサ：マップされたオブジェクト 6：なし
Rx	0x601	40 03 1A 07 00 00 00 00	リクエスト：1A03:7 を読出し
Tx	0x581	43 03 1A 07 00 00 00 00	アンサ：マップされたオブジェクト 7：なし
Rx	0x601	40 03 1A 08 00 00 00 00	リクエスト：1A03:8 を読出し
Tx	0x581	43 03 1A 08 00 00 00 00	アンサ：マップされたオブジェクト 8：なし

11-1 初期設定

■パワー ON

パワー ON で CANOpen 通信に問題がなければ、ドライバは、以下のメッセージを送信します。

Tx/Rx	ID	数 値	内 容
Tx	0x701	00	ブートアップ
Tx	0x81	00 00 00 00 00 00 00 00	エマージェンシプロトコルのエラーなし

SDO プロトコルを介してオブジェクトの Index と Sub-index に設定されている初期値を変更して EEPROM にストアすることができます。

これによりドライバを再起動した場合でも、EEPROM にストアされた値を読み出します。

⚠ 注 意

ドライバが “Operation Enabled” または “Quick Stop Active” 以外の状態で、ストア処理を実行してください。

EEPROM にパラメータをストアするには、オブジェクト 1010h を送信しリセットする必要があります。

Tx/Rx	ID	数 値	内 容
Rx	0x601	23 10 10 01 73 61 76 65	全パラメータをストア
Tx	0x581	60 10 10 01 00 00 00 00	-

全ノードをリセット (またはパワー OFF/ パワー ON でドライバを再起動)

Tx/Rx	ID	数 値	内 容
Rx	0x00	81 00	全ノードのリセット命令
Tx	0x701	00	ブートアップ
Tx	0x81	00 00 00 00 00 00 00 00	エマージェンシプロトコル = エラーなし

■ノード ID の設定方法

ノード ID の初期値は 1 です。以下は、ノード ID の変更方法を示します。

⚠ 注 意

ノード ID を変更するには、マスタコントローラとドライバを必ず接続する必要があります。

ノード ID の設定手順 (SDO で書込み)

マスタコントローラは、SDO メッセージ ID = 0x601 (定義 0x600 + ノード ID) を送信

- データ値 “Command” = 0x2F
 “Index” = 0x2000
 “Sub-Index” = 0x00
 “Data” = ノード ID の値 (例: 3)

【11】 各種の設定

ドライバは、SDO メッセージ ID = 0x581 (定義 0x580 + ノード ID) を応答

- データ値 "Command" = 0x60
"Index" = 0x2000
"Sub-Index" = 0x00
"Data" = 0

これらの SDO メッセージを示します。

ID	名 称	ノード	送信データ	説 明	データ
601	CSDO_001	Node1	03	[2000, 00] Initiate download Rq.	2F 00 20 00 03 00 00 00
581	SSDO_001	Node1	-	[2000, 00] Initiate download Rsp.	60 00 20 00 00 00 00 00

■ EEPROM へ数値のセーブ手順 (SDO へ書き込み)

⚠ 注 意

ドライバが "Operation Enabled" または "Quick Stop Active" 以外の状態で、ストア処理を実行してください。

マスタコントローラは、SDO メッセージ ID = 0x601 (定義 0x600 + ノード ID) を送信

- データ値 "Command" = 0x23
"Index" = 0x1010 (ストア)
"Sub-Index" = 0x01
"Data" = 0x73617665 (ASCII コードで "save")

ドライバは、SDO メッセージ ID = 0x581 (定義 0x580 + ノード ID) を応答

- データ値 "Command" = 0x60
"Index" = 0x1010
"Sub-Index" = 0x01
"Data" = 0

以下に SDO メッセージを示します。

ID	名 称	ノード	送信データ	説 明	Data
601	CSDO_001	Node1	73 61 76 65	[1010, 01] Initiate download Rq.	23 10 10 01 73 61 76 65
581	SSDO_001	Node1	-	[1010, 01] Initiate download Rsp.	60 10 10 01 00 00 00 00



次の手順

パラメータのストア処理の完了後にドライバをリセットします。

■ 全ノードをリセット (NMT プロトコル)

メッセージ ID = 0x00 (NMT プロトコル) を送信

- データ値 "Command" = 0x81
"Index" = 0x00

以下に SDO メッセージを示します。

ID	名 称	ノード	送信データ	説 明	エラー	データ
0	NMTZeroMag	-	-	Reset all Nodes	-	81 00

■リセット後 (NMT プロトコル)

ドライバは、ブートアップメッセージ ID = 0x703 (定義 0x700 + ノード ID) を応答

- データ値 "Index" = 0x00

ID	名 称	ノード	送信データ	説 明	エラー	データ
703	-	Node3	Boot-up	-	-	00



次の手順

ドライバは、エマージェンシメッセージを送信(エマージェンシプロトコル)

ドライバは、メッセージ ID = 0x83 (定義 0x80 + ノード ID) を送信

- データ値 "Error code" = 0x0

"Reg" = 0x0

"Data" = 0

以下に "Error Reset or No Error" を示します。

ID	名 称	ノード	送信データ	説 明	エラー	データ
83	EMCY_003	Node3	00 00 00 00 00 00 00 00	Error Reset or No Error	E	00 00 00 00 00 00 00 00



次の手順

ノード ID が正しく変更されたことを確認します。

■ノード ID の確認手順 (SDO を読み出し)

マスタコントローラは、SDO メッセージ ID = 0x603 (定義 0x600 + 新ノード ID) を送信

- データ値 "Command" = 0x40

"Index" = 0x2000

"Sub-Index" = 0x00

"Data" = 0x00

ドライバは、SDO メッセージ ID = 0x583 (定義 0x580 + 新ノード ID) を応答

- データ値 "Command" = 0x4F

"Index" = 0x2000

"Sub-Index" = 0x00

"Data" = 0x3

以下に SDO メッセージを示します。

ID	名 称	ノード	送信データ	説 明	データ
603	CSDO_003	Node3	-	[2000, 00] Initiate upload Rq.	40 00 20 00 00 00 00 00
583	SSDO_003	Node3	03	[2000, 00] Initiate upload Rsp.	4F 00 20 00 03 00 00 00

【11】 各種の設定

■ボーレートの変更方法

ボーレートの初期値は 1000Kb です。以下に、ボーレートの変更方法を示します。

ボーレート値の設定方法 (SDO へ書き込み)

マスタコントローラは、SDO メッセージ ID = 0x603 (定義 0x600 + ノード ID) を送信

- データ値 "Command" = 0x2B
"Index" = 0x2001
"Sub-Index" = 0x00
"Data" = ボーレート (例 : 500Kb = 0x01F4)

ドライバは、SDO メッセージ ID = 0x583 (定義 0x580 + ノード ID) を応答

- データ値 "Command" = 0x60
"Index" = 0x2001
"Sub-Index" = 0x00
"Data" = 0

次に SDO メッセージを示します。

ID	名 称	ノード	送信データ	説 明	データ
603	CSDO_003	Node3	F4 01	[2001, 00] Initiate download Rq.	2B 01 20 00 F4 01 00 00
583	SSDO_003	Node3	-	[2001, 00] Initiate download Rsp.	60 01 20 00 00 00 00 00

■ EEPROM へのセーブ手順 (SDO へ書き込み)

⚠ 注 意

ドライバが "Operation Enabled" または "Quick Stop Active" 以外の状態で、ストア処理を実行してください。

マスタコントローラは、SDO メッセージ ID = 0x603 (定義 0x600 + ノード ID) を送信

- データ値 "Command" = 0x23
"Index" = 0x1010 (ストア)
"Sub-Index" = 0x01
"Data" = 0x73617665 (ASCII コードで "save" の意味)

ドライバは、SDO メッセージ ID = 0x583 (定義 0x580 + ノード ID) を応答

- データ値 "Command" = 0x60
"Index" = 0x1010
"Sub-Index" = 0x01
"Data" = 0

以下に SDO メッセージを示します。

ID	名 称	ノード	送信データ	説 明	データ
603	CSDO_003	Node3	73 61 76 65	[1010, 01] Initiate download Rq.	23 10 10 01 73 61 76 65
583	SSDO_003	Node3	-	[1010, 01] Initiate download Rsp.	60 10 10 01 00 00 00 00



次の手順

パラメータのストア処理の完了後にドライバをリセットします。

■全ノードをリセット (NMT プロトコル)

マスタコントローラは、メッセージ ID = 0x00 (NMT プロトコル) を送信

- データ値 "Command" = 0x81
"Index" = 0x00

以下に SDO メッセージを示します。

ID	名 称	ノード	送信データ	説 明	エラー	データ
0	NMTZeroMag	-	-	Reset all Nodes	-	81 00

■リセット後 (NMT プロトコル)

ドライバは、ブートアップメッセージ ID = 0x703 (定義 0x700 + ノード ID) を応答

- データ値 "Index" = 0x00

ID	名 称	ノード	送信データ	説 明	エラー	データ
703	-	Node3	Boot-up	-	-	00



次の手順

ドライバは、エマージェンシメッセージを送信(エマージェンシプロトコル)

ドライバは、メッセージ ID = 0x83 (定義 0x80 + ノード ID) を送信

- データ値 "Error code" = 0x0
"Reg" = 0x0
"Data" = 0

以下に "Error Reset or No Error" を示します。

ID	名 称	ノード	送信データ	説 明	エラー	データ
83	EMCY_003	Node3	00 00 00 00 00 00 00 00	Error Reset or No Error	E	00 00 00 00 00 00 00 00



次の手順

ボーレートが正しく変更されたことを確認します。

■ボーレートの確認手順 (SDO を読み込み)

マスタコントローラは、SDO メッセージ ID = 0x603 (定義 0x600 + ノード ID) を送信

- データ値 "Command" = 0x40
"Index" = 0x2001
"Sub-Index" = 0x00
"Data" = 0x00

ドライバは、SDO メッセージ ID = 0x583 (定義 0x580 + ノード ID) を応答

- データ値 "Command" = 0x4B
"Index" = 0x2001
"Sub-Index" = 0x00
"Data" = 0x01F4

以下に SDO メッセージを示します。

ID	名 称	ノード	送信データ	説 明	データ
603	CSDO_003	Node3	-	[2001, 00] Initiate upload Rq.	40 01 20 00 00 00 00 00
583	SSDO_003	Node3	F4 01	[2001, 00] Initiate upload Rsp.	4B 01 20 00 F4 01 00 00

【11】 各種の設定

■単位の変換方法

出荷時のドライバ内部の単位は、速度オブジェクトが [inc/s]、加速オブジェクトが [inc/s²] です。

単位変換が必要な場合は、ファクタグループオブジェクトを変更する必要があります。

(例：速度オブジェクトを [rpm]、加速オブジェクトを [rpm/s] へ変換)

速度ファクタグループは、以下の式で表されます。

$$\text{速度ファクタ} = \frac{\text{分子}}{\text{分母}}$$

初期単位の [inc/s] では、分子と分母に 1 が設定されています。

初期単位を変換するには、オブジェクト (0x6096) に分子と分母の値を書込み、EEPROM へ新たな数値をセーブします

$$\text{速度 [内部単位]} = \text{速度 [ユーザ単位]} \times \left(\frac{\text{分子}}{\text{分母}} \right)$$

例：速度設定単位を rpm に変換

$$\text{速度 [rpm]} \times \left(\frac{\text{分子}}{\text{分母}} \right) = \text{速度 [inc/s]}$$

smartris 用のレゾルバおよびアブソリュートエンコーダの分解能は、モータ 1 回転で 16384 [inc] なので、分子に 16384、分母に 60 を設定してください。

加速ファクタグループも同様に設定できます。(オブジェクト 0x6097)

■ファクタグループ値の設定手順 (SDO へ書き込み)

速度ファクタグループの分子に書き込み (数値 = 16384)

マスタコントローラは、SDO メッセージ ID = 0x603 (定義 0x600 + ノード ID) を送信

- データ値 "Command" = 0x23
"Index" = 0x6096
"Sub-Index" = 0x01
"Data" = 16384 = 0x4000

ドライバは、SDO メッセージ ID = 0x583 (定義 0x580 + ノード ID) を応答

- データ値 "Command" = 0x6096
"Index" = 0x01
"Sub-Index" = 0x00
"Data" = 0

速度ファクタグループの分母に書き込み (数値 = 60)

マスタコントローラは、SDO メッセージ ID = 0x603 (定義 0x600 + ノード ID) を送信

- データ値 "Command" = 0x23
"Index" = 0x6096
"Sub-Index" = 0x02
"Data" = 60 = 0x3C

ドライバは、SDO メッセージ ID = 0x583 (定義 0x580 + ノード ID) を送信

- データ値 "Command" = 0x6096
"Index" = 0x02
"Sub-Index" = 0x00
"Data" = 0

【11】 各種の設定

加速ファクタグループの分子に書き込み (数値 = 16384)

マスタコントローラは、SDO メッセージ ID= 0x603 (定義 0x600 + ノード ID) を送信

- データ値 "Command"= 0x23
"Index"= 0x6097
"Sub-Index"= 0x01
"Data"= 16384 = 0x4000

ドライバは、SDO メッセージ ID = 0x583 (定義 0x580 + ノード ID) を応答

- データ値 "Command"= 0x6097
"Index"= 0x01
"Sub-Index"= 0x00
"Data"= 0

加速ファクタグループの分母に書き込み (数値 = 60)

マスタコントローラは、SDO メッセージ ID = 0x603 (定義 0x600 + ノード ID) を送信

- データ値 "Command"= 0x23
"Index"= 0x6097
"Sub-Index"= 0x02
"Data"= 60 = 0x3C

ドライバは、SDO メッセージ ID = 0x583(定義 0x580 + ノード ID) を応答

- データ値 "Command"= 0x6097
"Index"= 0x02
"Sub-Index"= 0x00
"Data"= 0

以下に SDO メッセージを示します。

ID	名 称	ノード	送信データ	説 明	エラー	データ
603	CSDO_003	Node3	00 40 00 00	[6096, 01] Initiate download Rq.	-	23 96 60 01 00 40 00 00
583	SSDO_003	Node3	-	[6096, 01] Initiate download Rsp.	-	60 96 60 01 00 00 00 00
603	CSDO_003	Node3	3C 00 00 00	[6096, 02] Initiate download Rq.	-	23 96 60 02 3C 00 00 00
583	SSDO_003	Node3	-	[6096, 02] Initiate download Rsp.	-	60 96 60 02 00 00 00 00
603	CSDO_003	Node3	00 40 00 00	[6097, 01] Initiate download Rq.	-	23 97 60 01 00 40 00 00
583	SSDO_003	Node3	-	[6097, 01] Initiate download Rsp.	-	60 97 60 01 00 00 00 00
603	CSDO_003	Node3	3C 00 00 00	[6097, 02] Initiate download Rq.	-	23 97 60 02 3C 00 00 00
583	SSDO_003	Node3	-	[6097, 02] Initiate download Rsp.	-	60 97 60 02 00 00 00 00



次の手順

変換した単位を EEPROM にセーブする前にストアとリセット処理が必要です。

【11】 各種の設定

■ EEPROM へのセーブ手順 (SDO へ書き込み)

⚠ 注意

ドライバが "Operation Enabled" または "Quick Stop Active" 以外のステートで、ストア処理を実行してください。

マスタコントローラは、SDO メッセージ ID = 0x603 (定義 0x600 + ノード ID) を送信

- データ値 "Command" = 0x23
"Index" = 0x1010 (ストア)
"Sub-Index" = 0x01
"Data" = 0x73617665 (ASCII コードで "save")

ドライバは、SDO メッセージ ID = 0x583 (定義 0x580 + ノード ID) を応答

- データ値 "Command" = 0x60
"Index" = 0x1010
"Sub-Index" = 0x01
"Data" = 0

以下に SDO メッセージを示します。

ID	名 称	ノード	送信データ	説 明	エラー	データ
603	CSDO_003	Node3	73 61 76 65	[1010, 01] Initiate download Rq.	-	23 10 10 01 73 61 76 65
583	SSDO_003	Node3	-	[1010, 01] Initiate download Rsp.	-	60 10 10 01 00 00 00 00



次の手順

パラメータのストア処理の完了後にドライバをリセットします。

■全ノードをリセット (NMT プロトコル)

マスタコントローラは、メッセージ ID = 0x00 (NMT プロトコル) を送信

- データ値 "Command" = 0x81
"Index" = 0x00

以下に SDO メッセージを示します。

ID	名 称	ノード	送信データ	説 明	エラー	データ
0	NMTZeroMag	-	-	Reset all Nodes	-	81 00

■リセット後 (NMT プロトコル)

ドライバは、ブートアップメッセージ ID = 0x703 (定義 0x700 + ノード ID) を応答

- データ値 "Index" = 0x00

ID	名 称	ノード	送信データ	説 明	Error	データ
703	-	Node3	Boot-up	-	-	00



次の手順

ドライバは、エマージェンシメッセージを送信 (エマージェンシプロトコル)

【11】 各種の設定

ドライバは、メッセージ ID = 0x83 (定義 0x80 + ノード ID) を送信

- データ値 "Error code" = 0x0

"Reg" = 0x0

"Data" = 0

以下に "Error Reset or No Error" を示します。

ID	名 称	ノード	送信データ	説 明	エラー	データ
83	EMCY_003	Node3	00 00 00 00 00 00 00 00	Error Reset or No Error	E	00 00 00 00 00 00 00 00

■関連するオブジェクトの再設定

ファクタグループが変更された場合、速度プロファイルモードの速度と加減速に関連するオブジェクトを、変換後の単位に合わせて再設定し EEPROM にセーブする必要があります。

⚠ 注 意

ファクタグループを変更後、関連するオブジェクトの初期値を設定可能な値(最大値以下)に修正しなければなりません。

設定値が異常な場合、ドライバはリセット(または再起動)後にエマージェンシメッセージを送信します。

エラーコード	初期のオブジェクトエラー	エラーコード	初期のオブジェクトエラー
0x8B11	CANopen 初期オブジェクト 0x6081	0X8B21	CANopen 初期オブジェクト 0x6072
0x8B12	CANopen 初期オブジェクト 0x6082	0X8B22	CANopen 初期オブジェクト 0x6073
0x8B13	CANopen 初期オブジェクト 0x6083	0X8B23	CANopen 初期オブジェクト 0x60E0
0x8B14	CANopen 初期オブジェクト 0x6084	0X8B24	CANopen 初期オブジェクト 0x60E1
0x8B15	CANopen 初期オブジェクト 0x60C5	0X8B25	CANopen 初期オブジェクト 0x6087
0x8B16	CANopen 初期オブジェクト 0x60C6	0X8B26	CANopen 初期オブジェクト 0x6086
0x8B17	CANopen 初期オブジェクト 0x607F	0X8B27	CANopen 初期オブジェクト 0x607B
0x8B18	CANopen 初期オブジェクト 0x6088	0X8B28	CANopen 初期オブジェクト 0x607D
0x8B19	CANopen 初期オブジェクト 0x6096	0X8B29	CANopen 初期オブジェクト 0x6099
0x8B1A	CANopen 初期オブジェクト 0x6097	0X8B2A	CANopen 初期オブジェクト 0x609A
0x8B1B	CANopen 初期オブジェクト 0x606D	0X8B2B	CANopen 初期オブジェクト 0x607C
0x8B1C	CANopen 初期オブジェクト 0x606E	0X8B2C	CANopen 初期オブジェクト 0x6065
0x8B1D	CANopen 初期オブジェクト 0x606F	0X8B2D	CANopen 初期オブジェクト 0x6066
0x8B1E	CANopen 初期オブジェクト 0x6070	0X8B2E	CANopen 初期オブジェクト 0x6067
0x8B1F	CANopen 初期オブジェクト 0x6075	0X8B2F	CANopen 初期オブジェクト 0x6068
0X8B20	CANopen 初期オブジェクト 0x6076	0X8B30	CANopen 初期オブジェクト 0x60F2

【11】 各種の設定

関連する以下のオブジェクトを変更する必要があります。表の上から順に設定してください。

Index	Sub-Index	オブジェクト名称	初期値 [内部単位]	設定値 [rpm 単位]
0x606D	0	速度ウィンドウ	13653 inc/sec	50 rpm
0x606F	0	速度スレッシュホールド	1365 inc/s	5 rpm
0x60C5	0	最大加速	608393 inc/s ²	2228 rpm/s 注)
0x60C6	0	最大減速	608393 inc/s ²	2228 rpm/s 注)
0x6083	0	加速プロファイル	273066 inc/s ²	1000 rpm/s
0x6084	0	減速プロファイル	273066 inc/s ²	1000 rpm/s

注) ドライバサイズにより設定する値が異なります。

S サイズ: 2228 rpm/s、M サイズ: 2387 rpm/s、L サイズ: 1910rpm/s

⚠ 注意

書込むオブジェクトの順番に注意してください。設定途中で 0x06090031 や 0x06040030 のアボートコードを検出した場合は、EEPROM ヘストアおよびリセット処理を実行することで解消されます。

■設定手順の一例 (60C5h の設定例)

最大加速のセーブ: 2228 [rpm/s] (S サイズの場合)

マスタコントローラは、SDO メッセージ ID = 0x603 (定義 0x600 + ノード ID) を送信

- データ値 "Command" = 0x23
"Index" = 0x60C5
"Sub-Index" = 0x00
"Data" = 2228 = 0x08B4

ドライバは、SDO メッセージ ID = 0x583 (定義 0x580 + ノード ID) を応答

- データ値 "Command" = 0x60C5
"Index" = 0x00
"Sub-Index" = 0x00
"Data" = 0

ID	名 称	ノード	送信データ	説 明	エラー	データ
603	CSDO_003	Node3	B4 08 00 00	[60c5, 00] Initiate download Rq.	-	23 C5 60 00 B4 08 00 00
583	SSDO_003	Node3	-	[60c5, 00] Initiate download Rsp.	-	60 C5 60 00 00 00 00 00



次の手順

他の関連する全てのオブジェクトに対して、rpm 単位に合わせた数値を書込む必要があります。

■ EEPROM へ新たな数値のセーブ手順 (SDO へ書込み)

⚠ 注意

ドライバが "Operation Enabled" または "Quick Stop Active" 以外の状態で、ストア処理を実行してください。

マスタコントローラは、SDO メッセージ ID = 0x603 (定義 0x600 + ノード ID) を送信

- データ値 "Command"=0x23
"Index"=0x1010 (ストア)
"Sub-Index"=0x01
"Data"=0x73617665 (ASCII コードで "save")

ドライバは、SDO メッセージ ID = 0x583 (定義 0x580 + ノード ID) を応答

- データ値 "Command"= 0x60
"Index"=0x1010
"Sub-Index"= 0x01
"Data"= 0

次に SDO メッセージを示します。

ID	名 称	ノード	送信データ	説 明	データ
603	CSDO_003	Node3	73 61 76 65	[1010, 01] Initiate download Rq.	23 10 10 01 73 61 76 65
583	SSDO_003	Node3	-	[1010, 01] Initiate download Rsp.	60 10 10 01 00 00 00 00



次の手順

パラメータのストア処理の完了後にドライバをリセットします。

■全ノードをリセット (NMT プロトコル)

マスタコントローラは、メッセージ ID = 0x00 (NMT プロトコル) を送信

- データ値 "Command"= 0x81
"Index"= 0x00

次に SDO メッセージを示します。

ID	名 称	ノード	送信データ	説 明	エラー	データ
0	NMTZeroMag	-	-	Reset all Nodes	-	81 00

■リセット後 (NMT プロトコル)

ドライバは、ブートアップメッセージ ID = 0x703 (定義 0x700 + ノード ID) を応答

- データ値 "Index"= 0x00

ID	名 称	ノード	送信データ	説 明	エラー	データ
703	-	Node3	Boot-up	-	-	00



次の手順

ドライバは、エマージェンシメッセージを送信(エマージェンシプロトコル)

ドライバは、メッセージ ID = 0x83 (定義 0x80 + ノード ID) を送信

- データ値 "Error code"= 0x0
"Reg"= 0x0
"Data"= 0

以下に "Error Reset or No Error" が示されています。

ID	名 称	ノード	送信データ	説 明	エラー	データ
83	EMCY_003	Node3	00 00 00 00 00 00 00 00	Error Reset or No Error	E	00 00 00 00 00 00 00 00

【11】 各種の設定

11-2 ハートビート・メカニズム

ハートビート・メカニズムはネットワークのマススタと各スレーブ間の通信ロスを検出することができます。

smartris ドライバは、ハートビート・メカニズムが定義された CANopen プロトコルの DS301 と DSP402 に適合しています。

ハートビートが正常に動作していて、スレーブ(ドライバ)がマススタとの通信ロスを検出したとき、自動的にドライバは、“Fault” ステートへ移行してアラームが送信されます。

■ハートビート・メカニズムのメッセージ構造

CANopen プロトコルの DS301 には、CANopen ノードがハートビートメッセージを送受信するための設定について記載されています。

ハートビートを生成・送信するノードは「プロデューサ」、ハートビートをモニタ・受信するノードは「コンシューマ」と呼ばれます。

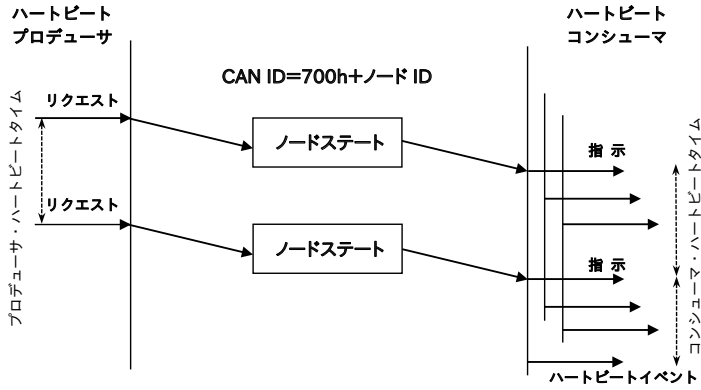


図 11-1 ハートビート・メカニズム (DS301)

■マススタハートビート

マススタハートビートは、次の特徴があります。

- ・CANopen マススタによって生成・送信(プロデュース)
- ・CANopen スレーブノードによってモニタ・受信(コンシューム)
- ・COB ID = 0x700
- ・データフレームなし

マススタのハートビートメッセージ：

COB-ID	Rx/Tx	DLC	バイト							
			0	1	2	3	4	5	6	7
0x700	Rx	0	-	-	-	-	-	-	-	-

■スレーブハートビート

スレーブハートビートは、次の特徴があります。

- ・ネットワークのスレーブノードによって生成・送信(プロデュース)
- ・CANopen マスタによってモニタ・受信(コンシューム)
- ・COB ID の範囲：0x701 ～ 0x77F

データフレームは、1 バイト長で下表に対応するスレーブノードの NMT 通信ステートの内容を含みます。

ハートビート値	内 容
0x0	Boot-up
0x1	Off bus
0x4	Stopped
0x5	Operational
0x7F	Pre-operational

スレーブのハートビートメッセージ：

COB-ID	Rx/Tx	DLC	バイト							
			0	1	2	3	4	5	6	7
0x700 + ノード ID	Tx	1	NMT ステート	-	-	-	-	-	-	-

ネットワークが PLC 1 台とドライバ 2 台で構成されている場合、2 台のドライバは、プロデュースで PLC がコンシューマとして設定されます。

■ドライバ・コンフィグレーション

「プロデュース ハートビートタイム」(1017h)を設定することで、ドライバがハートビートプロデュースになります。「ハートビート・プロデュースタイム」(1017h: 0)を設定した直後に、プロデュースのハートビートプロトコルが開始されて、周期的にハートビートメッセージがドライバから送信されます。

ハートビートのモニタも、プロデュースの送信周期が設定されている場合、直ぐに開始されます。NMT ステートが “Pre-operational” へ遷移している間にハートビートプロトコルが動作した場合、ブートアップメッセージの送信を開始します。

ブートアップメッセージは、1 データバイト (0x00) のハートビートメッセージです。

「プロデュース ハートビートタイム」(1017h)の設定は、「5 章 CANopen オブジェクトディクショナリ」を参照してください。

送信周期は、1ms 毎に設定します。コンシューマの設定値は、プロデュースの数値より短く設定してください。ハートビートメッセージが受信されるたびに、プロデュースの送信周期が再スタートします。

例：ドライバ・コンフィグレーション

Tx Rx	ID	数 値	内 容
		定義：Cmd + Index + Sub-index + 数値 バイト：(1) + (2) + (1) + (4)	
Rx	0x601	2B 17 10 00 64 00 00 00	リクエスト：ハートビートタイム 100ms (0x64) アンサ：ドライバは、周期的に 0x05 (Operational ステート) でハートビートメッセージを送信
Tx	0x581	60 17 10 00 00 00 00 00	
Tx	0x701	05	
Tx	0x701	05	
Tx	0x701	05	
...	

【11】 各種の設定

11-3 位置のモニタリング

モータ位置を把握する際に、ギヤ(減速機)を考慮する必要があります。

仮にギヤ比が 21 の場合、モータ軸が 21 回転することでギヤモータ出力軸(ホイール)は 1 回転します。

また、「位置検出値」(6064h)の値は、モータ軸の回転方向によって増加または減少します。

⚠ 注意

「位置検出値」(6064h)は、リセットか再起動で 0 に初期化されます。

■ホイールの累積回転数

モータ軸から見て時計方向に回転する場合、「位置検出値」(6064h)は 0 から単調増加していきます。

「位置検出値」(6064h)の値が 386007 の場合のモータ軸の累積回転数を求めます。

$$\text{モータ累積回転数} = \frac{\text{位置検出値 (6064h: 0)}}{\text{フィードバック分解能 (3004h: 2)}} = \frac{386007}{16384} = 23.56 \text{ 回転}$$

次に、モータ累積回転数とギヤ比からホイールの回転数を求めます(ギヤ比 21 の場合)。

$$\text{ホイール回転} = \frac{\text{モータ回転}}{\text{ギヤ比}} = \frac{23.56}{21} = 1.122 \text{ 回転}$$

計算結果から、ホイールは 1.122 回転したことになります。

■ホイール位置(角度)計算

ホイール位置は 360°で設定されており、電源投入時の位置を 0°とします。

「位置検出値」(6064h)の値が 386007、ギヤ比 21 のとき、ホイール位置の計算方法を以下に示します。

$$\text{ホイール位置} = \left(\left[\frac{\text{位置検出値 (6064h: 0)}}{\text{フィードバック分解能 (3004h: 2)}} \right] \div (\text{ギヤ比}) \right) \times 360 = 1.122 \times 360 = 403.89^\circ$$

ホイール位置は、403.89°から 360°を引いて 43.89°になります。

当社納入製品の保証範囲は、当社製作範囲に限定致します。

保証(期間および内容)

保証期間	新品に限り、工場出荷後 18 ヶ月または稼働後 12 ヶ月のうちいずれか短い方をもって保証期間と致します。
保証内容	<p>保証期間内において、取扱説明書に準拠する適切な据付、連結ならびに保守管理が行われ、かつ、カタログに記載された仕様もしくは別途合意された条件下で正しい運転が行われたにも拘わらず、本製品が故障した場合は、下記保証適用除外の場合を除き無償で当社の判断において修理または代品を提供致します。</p> <p>ただし、本製品がお客様の他の装置等と連結している場合において、当該装置等からの取り外し、当該装置等への取り付け、その他これらに付帯する工事費用、輸送等に要する費用ならびにお客様に生じた機会損失、操業損失その他の間接的な損害については当社の補償外とさせていただきます。</p>
保証適用除外	<p>下記項目については、保証適用除外とさせていただきます。</p> <ol style="list-style-type: none"> 1. 本製品の据付、他の装置等との連結の不具合に起因する故障 2. 本製品の保管が当社の定める保管要領書に定める要領によって実施されていないなど、保守管理が不十分であり、正しい取扱いが行われていないことが原因による故障 3. 仕様を外れる運転その他当社の知り得ない運転条件、使用状態に起因する故障または当社推奨以外の潤滑油を使用したことによる故障 4. お客様の連結された装置等の不具合または特殊仕様に起因する故障 5. 本製品をお客様にて分解、部品交換、および改造を施した場合 6. お客様範囲であるシーケンス回路等の不具合により弊社製品に二次的故障が発生した場合 7. お客様の支給受け部品もしくはご指定部品の不具合により生じた故障 8. 地震、火災、水害、塩害、ガス害、落雷、その他の不可抗力が原因による故障 9. 正常なご使用方法でも、電解コンデンサ等の消耗部品が自然消耗、摩耗、劣化した場合の当該消耗部品に関する保証 10. 前各号の他当社の責めに帰すことのできない事由による故障

営業所(住友重機械精機販売株式会社)		https://sjs.sumitomodrive.com	TEL	FAX
北海道	〒007-0847	札幌市東区北 47 条東 16-1-38	011-781-9802	011-781-9807
仙台	〒980-0811	仙台市青葉区一番町 3-3-16(オー・エックス芭蕉の辻ビル)	022-264-1242	022-224-7651
北関東	〒330-0854	さいたま市大宮区桜木町 4-242(鐘塚ビル)	048-650-4700	048-650-4615
千葉	〒260-0045	千葉市中央区弁天 1-15-1(細川ビル)	043-206-7730	043-206-7731
東京	〒141-6025	東京都品川区大崎 2-1-1(ThinkPark Tower)	03-6737-2520	03-6866-5171
横浜	〒220-0005	横浜市西区南幸 2-19-4(南幸折目ビル)	045-290-6893	045-290-6885
長野	〒380-0936	長野市岡田町 166(森ビル)	026-226-9050	026-226-9045
富山	〒939-8071	富山市上袋 327-1	076-491-5660	076-491-5604
金沢	〒920-0919	金沢市南町 4-55(WAKITA 金沢ビル)	076-261-3551	076-261-3561
静岡	〒422-8063	静岡市駿河区馬淵 3-2-25(T.K BLD)	054-654-3123	054-654-3124
中部	〒460-0003	名古屋市中区錦 1-5-11(名古屋伊藤忠ビル)	052-218-2980	052-218-2981
四日市	〒510-0064	三重県四日市市新正 4-17-20	059-353-7467	059-354-1320
滋賀	〒529-1601	滋賀県蒲生郡日野町大字松尾 334	0748-53-8900	0748-53-3510
京都	〒604-8187	京都市中京区御池通東洞院西入ル笹屋町 435(京都御池第一生命ビル)	075-231-2515	075-231-2615
大阪	〒530-0005	大阪市北区中之島 2-3-33(大阪三井物産ビル)	06-7635-3663	06-7711-5119
神戸	〒650-0044	神戸市中央区東川崎町 1-3-3(神戸ハーバーランドセンタービル)	078-366-6610	078-366-6625
岡山	〒701-0113	岡山県倉敷市栗坂 854-10	086-463-5678	086-463-5608
広島	〒732-0827	広島市南区福荷町 4-1(広島福荷町 NK ビル)	082-568-2521	082-262-5544
四国	〒792-0003	愛媛県新居浜市新田町 3-4-23(SIS ビル)	0897-32-7137	0897-34-1303
北九州	〒802-0001	北九州市小倉北区浅野 2-14-1(KMM ビル)	093-531-7760	093-531-7778
福岡	〒812-0025	福岡市博多区店屋町 8-30(博多フコク生命ビル)	092-283-3277	092-283-3177

修理・メンテナンスのお問い合わせ			TEL	FAX
サービスセンター(住友重機械精機販売株式会社)				
北海道	〒007-0847	札幌市東区北 47 条東 16-1-38	011-781-9803	011-781-9807
東京	〒335-0031	埼玉県戸田市美女木 5-9-13	048-449-4755	048-449-4785
名古屋	〒474-0023	愛知県大府市大東町 2-97-1(サービステクニカルセンター)	0562-45-6402	0562-44-1998
大阪	〒567-0865	大阪府茨木市横江 2-1-20	072-637-3901	072-637-5774
岡山	〒701-0113	岡山県倉敷市栗坂 854-10	086-464-3681	086-464-3682
福岡	〒812-0025	福岡市博多区店屋町 8-30(博多フコク生命ビル)	092-431-2678	092-431-2694

技術的なお問い合わせ				
お客様相談センター(住友重機械工業株式会社 PTC 事業部)		https://www.shi.co.jp/ptc/		
営業時間		フリーダイヤル	0120-42-3196	
月曜日～金曜日 9:00～11:45 13:00～16:45		携帯電話から	0570-03-3196	
(祝日・弊社休業日を除く)		FAX	0562-48-5183	

記載内容は、製品改良などの理由により予告なく変更することがあります。