OMRON

CK3Aシリーズ

DirectPWMアンプ

ユーザーズマニュアル

形CK3A-G305L
形CK3A-G310L
形CK3A-G320L

DirectPWMアンプ

SBCE-518B

おことわり

(1) 本マニュアルの内容の一部または全部を無断で複写、複製、転載することを禁じます。

(2) 本マニュアルの内容に関しては、改良のため予告なしに仕様などを変更することがあります。
あらかじめご了承ください。

(3) 本マニュアルの内容に関しては、万全を期しておりますが、万一ご不審な点や誤りなど、お気づきのことがありましたら、お手数ですが、弊社支店または営業所までご連絡ください。
その際、巻末記載のMan.No.（マニュアルNo.）も併せてお知らせください。

商標

- Microsoft、Windows、Excel、Visual Basic、Microsoft Edgeは、米国Microsoft Corporationの米国およびその他の国における登録商標または商標です。

その他、本文中に掲載している会社名および製品名は、それぞれ各社の商標または登録商標です。

著作権

- スクリーンショットはマイクロソフトの許諾を得て使用しています。

はじめに

このたびは、CK3A DirectPWM アンプをお買い上げいただきまして、ありがとうございます。

本マニュアルは CK3A DirectPWM アンプを適切に取り付け、設置、配線、構成、トラブルシューティングする上で必要な情報を記載しています。

対象となる読者の方々

本マニュアルは、電気の知識（電気工事士あるいは同等の知識）を有する、次の方を対象に記述しています。

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

お知らせ

本ユーザーズマニュアルには CK3A アンプを正しくご使用いただくために欠かせない情報を記載しています。

本マニュアルをよく読んで、CK3A アンプの適切な使用方法をご理解ください。また、お読みになった後も本マニュアルを大切に保管して、いつも手元においてお使いいただくと共に、最終ユーザ様までお届けくださいますようお願いいたします。

対象となる製品

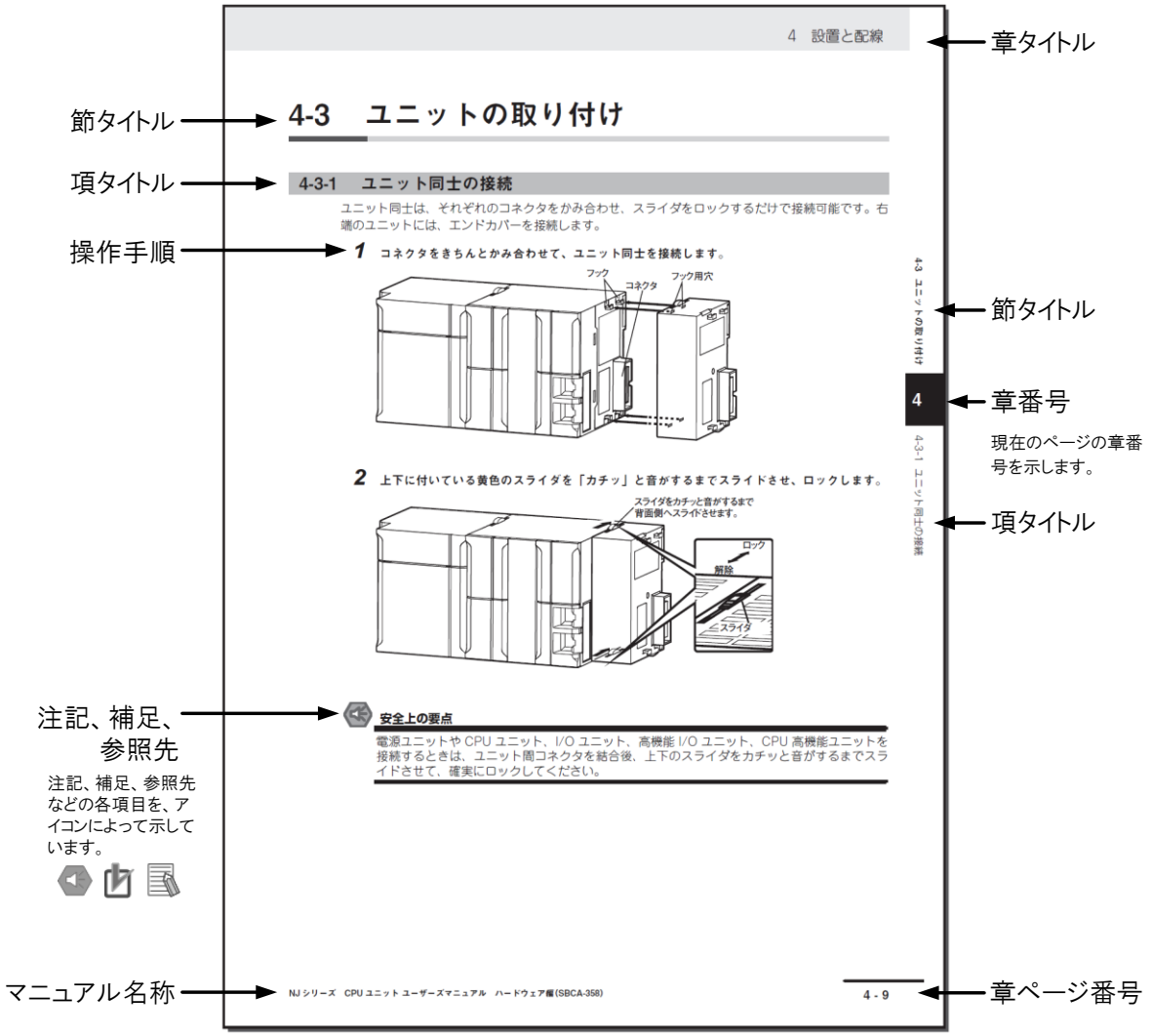
本マニュアルは以下の製品を対象にしています。

- 形 CK3A-G305L
- 形 CK3A-G310L
- 形 CK3A-G320L

マニュアルの読み方

ページ構成

このマニュアルの各ページの構成は次のとおりです。



（注） このページは、説明用の見本です。実際の内容とは異なります。

PMAC スクリプト

本マニュアルには、IDE ソフトウェアで使用できる PMAC（言語）スクリプトサンプルを記載します。スクリプトサンプルは、次の形式で囲んで表示します。

```
GLOBAL MyVar1        // グローバル変数 1
GLOBAL MyVar2 = 0    // グローバル変数 2
```

アイコン

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

安全上の要点

製品を安全に使用するために実施または、回避すべきことを示します。

使用上の注意

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

参考

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

目次構成

1 アンプの概要
2 仕様
3 取り付けと設置
4 配線
5 ソフトウェア構成
6 異常の確認と処置
A 付録

目次

はじめに 1
対象となる読者の方々 1
お知らせ 1
対象となる製品 1
マニュアルの読み方 2
ページ構成 2
PMAC スクリプト 2
アイコン 3
目次構成 5
ご承諾事項 11
安全上の注意 13
安全に使用していただくための表示と意味 13
図記号の説明 13
注意事項 14
安全上の要点 19
使用上の注意 22
開梱後の確認事項 27
銘板位置 27
付属品 28
法規と規格 29
EU 指令への適合 29
UL 指令への適合 29
UKCA 規格への適合 29
KC 規格への適合 29
バージョン 31
ハードウェアバージョン 31
ファームウェアバージョン 31
関連マニュアル 32
用語説明 33
マニュアル改訂履歴 34
第 1 章　アンプの概要 1-1
1-1　概要 1-2
1-1-1 アンプの機能 1-2
1-1-2 基本構成 1-3
1-1-3 各部の名称と配置 1-4
1-1-4 各部の機能 1-8
1-2　使用手順 1-10

1-2-1 準備 1-10
1-2-2 安全性 1-10
1-2-3 取り付けと配線 1-10
1-2-4 安全性試験と電源投入 1-10
1-2-5 アンプのコントローラ設定 1-11
1-2-6 試運転 1-11

第 2 章　仕様 2-1

2-1　アンプの仕様 2-2
2-1-1 一般/機械 2-2
2-1-2 環境 2-2
2-1-3 電気 2-3
2-1-4 性能 2-5
2-1-5 回生エネルギー吸収 2-6

2-2　機能とデータ報告 2-10
2-2-1 アンプ機能 2-10
2-2-2 データ報告 2-11
2-2-3 ステータスビット 2-11

2-3　部品番号表示 2-12

第 3 章　取り付けと設置 3-1

3-1　外形寸法と取り付け寸法 3-2
3-1-1 形 CK3A-G305L の寸法と取り付け 3-2
3-1-2 形 CK3A-G310L、形 CK3A-G320L の寸法と取り付け 3-3

3-2　設置 3-4
3-2-1 換気 3-4
3-2-2 盤内クリアランス 3-4
3-2-3 取り付け方向 3-6
3-2-4 設置条件 3-7
3-2-5 異物をユニットに入れない 3-7

第 4 章　配線配線 4-1

4-1　主回路電源とシャント（CN1） 4-3
4-1-1 コネクタピン配置 4-3
4-1-2 電線寸法 4-4
4-1-3 配線例 4-5

4-2　ロジック電源（CN2） 4-10
4-2-1 コネクタピン配置 4-10
4-2-2 電線寸法 4-10
4-2-3 配線例 4-10

4-3　モータ接続（CN3） 4-11
4-3-1 コネクタピン配置 4-11
4-3-2 電線寸法 4-12
4-3-3 配線例 4-13

4-4 安全トルク OFF STO（CN4） 4-14
4-4-1 コネクタピン配置 4-14
4-4-2 電線寸法 4-15
4-4-3 配線例 4-15
4-5 DirectPWM 接続（CN5） 4-17
4-6 全体配線図の例 4-18
4-6-1 形 CK3A-G305L 4-18
4-6-2 形 CK3A-G310L、形 CK3A-G320L 4-19
4-6-3 アンプ異常リレー 4-20
4-7 EMC 指令に準拠した配線 4-21
4-7-1 周辺機器の接続例 4-21
4-8 周辺機器の選択 4-23
4-8-1 主回路 AC 電源 4-23
4-8-2 主回路 DC 電源（低電圧モード） 4-23
4-8-3 ロジック電源 4-23
4-8-4 ノイズフィルタ 4-24
4-8-5 電圧サージアブソーバ 4-24
4-8-6 配線用遮断器（MCCB） 4-24
4-8-7 外部シャント抵抗 4-25
4-8-8 DirectPWM ケーブル 4-25
4-8-9 モータの適合性 4-26

第 5 章 ソフトウェア構成 5-1

5-1 ソフトウェア構成の概略 5-3
5-1-1 コントローラ-アンプインタフェース 5-3
5-1-2 要約 5-3
5-1-3 対象となるユーザの方々 5-4
5-1-4 対象となるハードウェア 5-4
5-1-5 Gate3 エイリアス名 5-4
5-2 基本構成 5-5
5-2-1 PWM 周波数 5-5
5-2-2 Gate3 設定 5-5
5-2-3 チャネル設定 5-5
5-2-4 モータ設定 5-6
5-3 ADC ストローブワードの詳細 5-7
5-3-1 概要 5-7
5-3-2 明示的アドレス 5-8
5-4 アンプ機能の詳細 5-9
5-4-1 特長 5-9
5-4-2 バス放電制御 5-9
5-4-3 異常解除 5-11
5-4-4 ファン制御 5-11
5-4-5 ダイナミックブレーキ制御 5-12
5-4-6 低電圧（LV）モード 5-13
5-5 ADC レジスタの詳細 5-14

5-5-1 ADC A の情報 5-14
5-5-2 ADC B の情報 5-14
5-5-3 ADC レジスタ明示的アドレス 5-15

5-6 ステータスビットの詳細 5-16

5-6-1 ステータスビットセット 1 5-16
5-6-2 ステータスビットセット 2 5-17
5-6-3 ステータスビットセット 3 5-18

5-7 アンプデータの詳細 5-19

5-8 CK3A サンプル PLC 5-21

5-8-1 サンプル PLC の実装 5-21
5-8-2 遅延タイマーサブルーチン 5-22
5-8-3 ユーザ共有メモリの説明 5-23
5-8-4 CK3A サンプル定義 4 モータ 5-24
5-8-5 CK3A サンプル PLC 4 モータ 5-26
5-8-6 CK3A サンプル定義 8 モータ 5-28
5-8-7 CK3A サンプル PLC 8 モータ 5-32
5-8-8 電源投入時に CK3A PLC を有効にする 5-34
5-8-9 CK3A PLC の使用 5-35

第 6 章 異常の確認と処置 6-1

6-1 LED 表示 6-2

6-2 表示コード 6-3

6-2-1 正常モード動作 6-3
6-2-2 エラーコード 6-3
6-2-3 エラーコードの確認と処置 6-6
6-2-4 異常しきい値 6-10
6-2-5 アンプ動作処置 6-11

付録 A-1

A-1 IDE を使用して自動モータの設定 A-3

A-1-1 IDE システム設定 A-3

A-2 Power PMAC3 手動モータの設定 A-12

A-2-1 IDE プロジェクトの作成 A-12
A-2-2 基本的な最適化設定、システム設定およびGate設定 A-16
A-2-3 エンコーダの進行中位置の設定と検証 A-17
A-2-4 モータの設定と試運転 A-22
A-2-5 モータ動作方向の反転 A-46

A-3 Gate3「スクリプト」モータ設定サンプル A-47

A-3-1 パルスエンコーダ付きロータリサーボモータ A-48
A-3-2 オムロン G5 サーボモータ A-50
A-3-3 オムロン 1S サーボモータ（ACC-84E） A-53
A-3-4 正弦波エンコーダ付きのリニアサーボモータ A-54
A-3-5 BiSS エンコーダ付きのリニアサーボモータ（ACC-84E） A-57

A-4 Gate1「スクリプト」モータ設定サンプル A-59

A-4-1 Gate1(ACC-24E2)ありの Power PMAC A-60

A-4-2 Gate1（ACC-24E2）ありの Turbo PMAC A-62

A-5 Gate1 用 CK3A サンプル PLC A-64

A-5-1 Gate1 ありの Power PMAC A-64
A-5-2 Gate1 ありの Turbo PMAC A-67

A-6 STO 安全上の注意 A-72

A-6-1 危険分析 A-72
A-6-2 STO 仕様 A-75
A-6-3 STO の動作原理 A-76

ご承諾事項

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

定義

本ご承諾事項中の用語の定義は次のとおりです。

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

記載事項のご注意

「カタログ等」の記載内容については次の点をご理解ください。

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

ご利用にあたってのご注意

ご採用およびご利用に際しては次の点をご理解ください。

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

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

- 「当社商品」は、一般工業製品向けの汎用品として設計製造されています。従いまして、次に掲げる用途での使用は意図しておらず、お客様が「当社商品」をこれらの用途に使用される際には、「当社」は「当社商品」に対して一切保証をいたしません。ただし、次に掲げる用途であっても「当社」の意図した特別な商品用途の場合や特別の合意がある場合は除きます。

a) 高い安全性が必要とされる用途（例：原子力制御設備、燃焼設備、航空・宇宙設備、鉄道設備、昇降設備、娯楽設備、医用機器、安全装置、その他生命・身体に危険が及びうる用途）
b) 高い信頼性が必要な用途（例：ガス・水道・電気等の供給システム、24 時間連続運転システム、決済システムほか権利・財産を取扱う用途など）
c) 厳しい条件または環境での用途（例：屋外に設置する設備、化学的汚染を被る設備、電磁的妨害を被る設備、振動・衝撃を受ける設備など）
d) 「カタログ等」に記載のない条件や環境での用途

- 上記の(a)から(d)に記載されている他、「本カタログ等記載の商品」は自動車（二輪車含む。以下同じ）向けではありません。自動車に搭載する用途には利用しないで下さい。自動車搭載用商品については当社営業担当者にご相談ください。

保証条件

「当社商品」の保証条件は次のとおりです。

- 保証期間　ご購入後 1 年間といたします。

（ただし「カタログ等」に別途記載がある場合を除きます。）

- 保証内容　故障した「当社商品」について、以下のいずれかを「当社」の任意の判断で実施します。

a) 当社保守サービス拠点における故障した「当社商品」の無償修理
（ただし、電子・機構部品については、修理対応は行いません。）
b) 故障した「当社商品」と同数の代替品の無償提供

- 保証対象外　故障の原因が次のいずれかに該当する場合は、保証いたしません。

a)「当社商品」本来の使い方以外のご利用
b)「利用条件等」から外れたご利用
c) 本ご承諾事項「ご利用にあたってのご注意」に反するご利用
d)「当社」以外による改造、修理による場合
e)「当社」以外の者によるソフトウェアプログラムによる場合
f) 「当社」からの出荷時の科学・技術の水準では予見できなかった原因
g) 上記のほか「当社」または「当社商品」以外の原因（天災等の不可抗力を含む）

責任の制限

本ご承諾事項に記載の保証が、「当社商品」に関する保証のすべてです。
「当社商品」に関連して生じた損害について、「当社」および「当社商品」の販売店は責任を負いません。

輸出管理

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

安全上の注意

CK3A DirectPWM アンプを安全かつ正しく使用していただくために、設置および配線の前に、安全上の注意を注意深く確実にお読みください。本ユーザーズマニュアルは最終ユーザ様までお届けいただき、参照できるようにすぐアクセスできる場所に保管ください。

安全に使用していただくための表示と意味

ここで示した注意事項は、安全に関する重大な内容を記載しています。必ず守ってください。

CK3A DirectPWM アンプを安全に使用していただくために、注意事項を次のような表示と図記号で示しています。

⚠ 警告	正しい取扱いをしなければ、この危険のために、軽傷・中程度の傷害を負ったり万一の場合には重傷や死亡に至る恐れがあります。また、同様に重大な物的損害をもたらす恐れがあります。

図記号の説明

本マニュアルには、次の図記号を記載しています。

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

注意事項

本マニュアルに記載されているイラストは、詳細を示す目的で、カバーや安全シールドがない状態を示している場合があります。本製品をお使いになるときは、必ず指定どおりにカバーとシールドを取り付け、本マニュアルに従って製品をお使いください。

輸送および開梱時

⚠ 警 告

製品を輸送するときは、ケーブルを持って運ばないでください。
けがやユニットの破損、誤動作につながるおそれがあります。

製品を踏んだり、上に重いものを載せないでください。
けがやユニットの破損、誤動作につながるおそれがあります。

設置と配線およびメンテナンス時

⚠ 警 告

本マニュアルの指示に従い、製品を電源、モータ、その他の機器に
正しく接続してください。重傷につながるおそれがあります。

適切な外部シャント抵抗を使用してください。外部シャント抵抗を使用する場合は、安全を確保するために、温度センサなどの外部保護装置を取り付けてください。
焼損のおそれがあります。

金属などの不燃性物質の上に製品および周辺機器を設置してください。
火災のおそれがあります。

端子台・コネクタの配線及び接続は本マニュアルに従って正しく行ってください。配線の順番、コネクタの向きを十分確認してから通電してください。感電、火災、機器の損傷、誤動作、けがにつながるおそれがあります。

ケーブルを傷つけたり、引っ張ったり、強く曲げたり、過度の圧力をかけたり、重いものを載せないでください。故障、焼損の可能性があります。

製品や周辺機器の近くに可燃物を置かないでください。 火災のおそれがあります。	
低電圧短絡線（CN1）が取り付けられている場合は、アンプに AC 電力を供給しないでください。破損、誤動作、けがのおそれがあります。	
アンプ（とモータ）を必ず D 種接地（第 3 種接地）してください。感電のおそれがあります。	
アンプのカバーは取り外さないでください。 感電のおそれがあります。	
適切な工具を使用して、端子とコネクタを配線してください。お使いになる前に短絡がないことを確認してください。感電のおそれがあります。	
モータケーブルの保安用接地線をアンプの⏚または FG にしっかりと接続してください。感電のおそれがあります。	
外部配線の短絡や製品の故障から保護するために、ヒューズ、配線用遮断器、または漏電サーキットブレーカなどの安全対策を施してください。火災のおそれがあります。	
CK3A シリーズはソリッド ステート短絡保護を提供するため、最終製品には次のマークを付ける必要があります: 一体型ソリッド ステート短絡保護は分岐回路保護を提供しません。分岐回路保護は、米国電気工事規定および追加の地域規定、または同等の規定に従って提供する必要があります。	

STO

⚠ 警 告

STO のフィードバック回路は、SIL 3 保護レベルを保証するために、最低限の診断範囲を備えた適切な定格の外部 SIL 3 保護デバイスに接続する必要があります。 そうしないと、破損、誤動作、けがのおそれがあります。	
ほとんどの場合、STO をアクティブにするには STO 入力の 1 つだけをコマンドするだけで十分ですが、安全操作のためには、STO IN1 と STO IN2 の両方の入力を個別にコマンドする必要があります。そうしないと、破損、誤動作、けがのおそれがあります。 STO IN1 と STO IN2 を短絡させないでください。	
STO 入力の電源電圧の 24V_Return は、CK3A 24V_Return ロジック電源に接続する必要があります。 接続しないと、損傷、誤動作、または怪我の原因となる可能性があります。	

ソフトウェア構成時

⚠ 警 告

モータ固有の保護パラメータ、特に Motor[x].I2tSet、Motor[x].I2tTrip、Motor[x].MaxDac などの I2T 熱保護を必ずコントローラに正しく設定してください。モータの焼損、火災、重傷につながるおそれがあります。

主電源投入中

⚠ 警 告

配線や点検を行う前に、主回路の電源を切り、5 分以上待ってください。感電や火傷のおそれがあります。

フェールセーフ対策

⚠ 警 告

製品の故障や外部要因によりアンプ異常やその他の異常が発生した場合も、システム全体が安全側に働くように主回路電源を OFF にするなど、外部で安全対策を施してください。重大な事故につながるおそれがあります。

非常停止、インターロック、リミット回路など、安全保護に関する回路は、必ず外部の制御回路で構成してください。重大な事故につながるおそれがあります。

信号線の断線、瞬時停電による異常信号などに備えて、ご使用者側でフェールセーフ対策を施してください。重大な事故につながるおそれがあります。

縦軸には、サーボ制御停止時の転倒防止のため、ブレーキまたはカウンターバランス機構を使用してください。重傷につながるおそれがあります。

試運転時

⚠ 警 告

本マニュアルおよび関連マニュアルの指示に従い、アンプを適切にプログラムしてください。
機器の損傷につながるおそれがあります。

Power PMAC コントローラを交換するとき、または IDE ソフトウェアから構成を変更するときは、必ず安全性を確認してください。製品の運転モードにかかわらず、機器や機械が想定外の動作をするおそれがあります。
機器の損傷、誤動作、または怪我につながるおそれがあります。

運転時

⚠ 警 告

運転中は機械の動作エリアに入らないでください。
けがのおそれがあります。

高温になるため、電源が供給されている間や電源が OFF になってから少なくとも 5 分間は、アンプのヒートシンク、シャント抵抗、またはモータに触れないでください。
火災ややけどのおそれがあります。

定格電圧で仕様の電力が供給されるよう適切な対策を施してください。電源事情が悪い場所では特に注意してください。
故障のおそれがあります。

瞬時停電後に電源が復旧すると、機械動作が突然再開する場合があります。電源復旧時は、機械に近づかないでください。けがのおそれがあります。機械動作が再開した場合でも、近くの人の安全を確保できる対策を施してください。

主回路の電源が ON の間は、アンプへの接続を抜き差ししないでください。
故障、機器の損傷、または重傷のおそれがあります。

モータ接続コネクタが接続されていないときにコントローラがモータコマンドを送信しても、アンプはエラーを出しません。コントローラがモータコマンドを送信していないか、主回路の電源が OFF になっていることを確認するまで、モータ接続コネクタを接続しないでください。
故障、機器の損傷、または重傷のおそれがあります。

ソフトスタート中はモータを有効にしないでください。AC 主電源を ON にした後、モータを有効にする前に、最低 5 秒お待ちください。
故障、機器の損傷、または重傷のおそれがあります。

安全上の要点

輸送時

- 製品を落下させたり、異常な振動・衝撃を与えないでください。

製品の故障、焼損、感電のおそれがあります。

装着時

- 取り付け方向と周囲との距離を必ず守ってください。
- 端子台、コネクタ、およびその他のロック機構が適切にロックしていることを確認してからご使用ください。

設置時

- 給気口や排気口をふさがないでください。故障、運転停止、機器の損傷のおそれがあります。
- 製品に異物が入らないようにしてください。
- 次のような環境に設置や保管をしないでください。

  a) 日光が直接当たる場所
  b) 周囲温度が仕様値の範囲を超える場所
  c) 相対湿度が仕様値の範囲を超える場所
  d) 急激な温度変化で結露するような場所
  e) 腐食性ガス、可燃性ガスのある場所
  f) ちり、ほこり、塩分、鉄粉が多い場所
  g) 水、油、薬品などの飛まつがかかる場所
  h) 本体に直接振動や衝撃が伝わる場所

- 感電、火災、機器の損傷、誤動作につながるおそれがあります。

配線時

- 端子台・コネクタの配線及び接続は本マニュアルに従って正しく行ってください。配線の順番、コネクタの向きを十分確認してから通電してください。
- ケーブルを無理に曲げたり、引っ張ったりしないでください。
- ケーブルのコード部に重いものを載せないでください。断線するおそれがあります。
- 製品内に配線クズや切粉などの異物が入らないようにしてください。製品の焼損、故障、誤動作の原因となります。特に接続時は、覆いを付けるなどの対策を行ってください。
- ケーブルカバーや適切なメカニズムを使用して、システムの可動部分と固定部分を分離してください。
- コンタクタやその他の電気リレー機器を使用せずに、製品をモータに接続してください。

電源設計時

- 電源は、本マニュアルに記載されている電源の定格範囲内としてください。
- 短絡および過電流に備えて、ブレーカなどの安全対策を施してください。
- 本マニュアルに記載する範囲を超える電圧を印加しないでください。

モータ選択時

- 仕様をよく確認してから、製品に接続するモータを選択してください。

ソフトウェア構成時

- 本マニュアルの「第 5 章 ソフトウェア構成」で説明している、アンプ固有のパラメータを必ずコントローラに正しく設定してください。
- モータ固有の保護パラメータを必ずコントローラに正しく設定してください。特に、I2T 熱保護設定に注意してください。

電源投入時

- 端子台・コネクタなどロック式やねじ式機構のあるものは、必ずロックしていることを確認してから電源投入してください。
- アンプの電源を ON にする正しい手順は以下のとおりです。

1 常に主回路電源の前に、DC24V ロジック電源をオンにします。

2 アンプが起動するまで最低 10 秒待ちます。

3 主回路電源（AC200V など）をオンにします。

4 5 秒以上待ちます。

- アンプの起動中は、主電源を入れたり、コントローラからアンプにコマンドを出さないでください。
- 主回路の電源を入れる前に、コントローラがアンプにモータコマンドを送信していないことを確認してください。
- 主回路の電源を入れた後、コントローラからアンプにモータコマンドを送信する前に、最低 5 秒待機してください。

上記のすべてを行わない場合、重傷、機器の損傷、または誤動作を引き起こす可能性があります。

通電時

- 高温になるため、電源が供給されている間や電源が OFF になってから少なくとも 5 分間は、アンプのヒートシンク、シャント抵抗、またはモータに触れないでください。火災ややけどのおそれがあります。
- 異常が発生した場合は、異常の原因を取り除いて安全を確保してから、システムのリセットと再起動を行ってください。破損、誤動作、けがのおそれがあります。

電源 OFF 時

- アンプの電源を OFF にする正しい手順は以下のとおりです。

1 常に DC24V ロジック電源より先に主回路電源（AC200V など）をオフにします。

2 コントローラのストローブワード設定によりアンプがコンデンサに蓄えたエネルギーを放電するようプログラミングされている場合は、最低 3 秒待機してください。

3 DC24V ロジック電源を OFF にしてください。

- 次のいずれかを実行する前に、必ず電源を OFF にしてください。

a) ユニットを着脱するとき。
b) ケーブルを接続、配線するとき。
c) 端子台やコネクタを取り付けたり、取り外したりするとき。

上記のすべてを行わない場合、重傷、機器の損傷、または誤動作を引き起こす可能性があります。

保守時

- 本製品の掃除に腐食性の薬品を使用しないでください。故障や誤動作の原因になります。
- 製品の廃棄については、地方自治体により規制を受ける場合があります。

使用上の注意

一般的な注意事項

- 次のような場所で使用する際は、遮蔽対策を十分に行ってください。

a) 強い高周波電磁ノイズを発生する機器の近く
b) 静電気などによるノイズが発生する場所
c) 強い電界や磁界が生じる場所
d) 放射線を被ばくするおそれのある場所
e) 電源線や動力線が近くを通る場所

設置と取り付け時

- 本マニュアルの指示に従い、アンプを正しく設置してください。
- 規定の締め付けトルクでアンプを取り付けてください。
- 製品および周辺機器の取り付けねじを規定トルクで締め付けてください。
- 次のような場所で使用する際は、遮蔽対策を十分に行ってください。

  a) 強い高周波電磁ノイズを発生する機器の近く
  b) 静電気などによるノイズが発生する場所
  c) 強い電界や磁界が生じる場所
  d) 放射線を被ばくするおそれのある場所
  e) 電源線や動力線が近くを通る場所
- 発熱体の近傍への設置を避け、通風の確保などをして正しく設置してください。
- 製品の取り付けまたは交換後は、実際の運転に移る前に、動作を十分に確認してください。

上記のすべてを行わない場合、怪我、機器の損傷、誤動作、または火災につながるおそれがあります。

配線時

- ケーブルを正しく安全に配線してください。
- 配線と組み立ては慎重に行ってください。
- 圧着端子を使用して、ねじ留め式端子台を配線します。むき出しのより線を端子台に直接接続しないでください。
- 必ずマニュアルに記載している電源電圧を使用してください。
- 安全機能を備えたシステムを構築する場合は、関連する安全基準と取扱説明書などのすべての関連情報を理解し、基準に準拠するようにシステムを設計してください。
- アンプのメガーテスト（絶縁抵抗測定）を試みる前に、アンプへのすべての接続を切断してください。アンプの絶縁耐力試験は行わないでください。
- ねじ、金属片、オイルなどの導電性または可燃性の異物をアンプやコネクタに入れないでください。アンプ上部のコネクタには特に注意してください。
- コネクタの取り付けまたは取り外しの際は保護具を着用し、特に主回路コネクタに注意してください。コネクタオープナーの突起が下死点に達した後は、力を加えないでください。（目安として、100N 以上の力を加えないでください）。
- ワイヤ端子やコネクタに無理な力を加えないでください。

上記のすべてを行わない場合、怪我、機器の損傷、誤動作、または火災につながるおそれがあります。

電源の ON/OFF 時

- ロジック電源が OFF の場合は、主回路電源の再投入を行わないでください。
- ロジック電源を ON にした後、10 秒待ってから主回路電源を ON にしてください。

調整時

- 試運転中にコントローラやコンピュータとの通信に問題が発生した場合、モータを停止する手段はありません。汎用入力の非常停止入力に外付けの非常停止スイッチ等を接続し、必ずモータを停止させてください。
- コントローラのパラメータを極値に調整しようとしないでください。動作が不安定になるおそれがあります。

上記のすべてを行わない場合、怪我、機器の損傷、誤動作、または火災につながるおそれがあります。

運転時

- コネクタがしっかりと差し込まれ、端子台がきっちりとねじ込まれていることを確認してください。
- 安全を確保するために、機械に適切な停止装置を取り付けてください。ダイナミックブレーキ機構は停止装置ではありません。
- 機械の外部に即時停止装置を設置して、運転を停止し、電源が直ちに遮断されるようにしてください。
- 地震、落雷、洪水、電力サージの後は、必ず安全チェックを行ってください。
- 製品の近くに可燃物を置かないでください。
- 油や水に浸かっているケーブルを使用しないでください。
- アンプが故障した場合は、電源でアンプへの電力供給を遮断してください。
- アンプを長期間使用しないときは、電源を OFF にしてください。
- 異臭、異音、発煙、異常発熱、振動が発生した場合は、ただちに運転を停止し、電源を OFF にしてください。
- 適切な外部シャント抵抗を使用してください。外部シャント抵抗を使用する場合は、安全を確保するために、温度センサなどの外部保護装置を取り付けてください。
- 上記のすべてを行わない場合、怪我、機器の損傷、誤動作、または火災につながるおそれがあります。

保守と異常の確認と処置

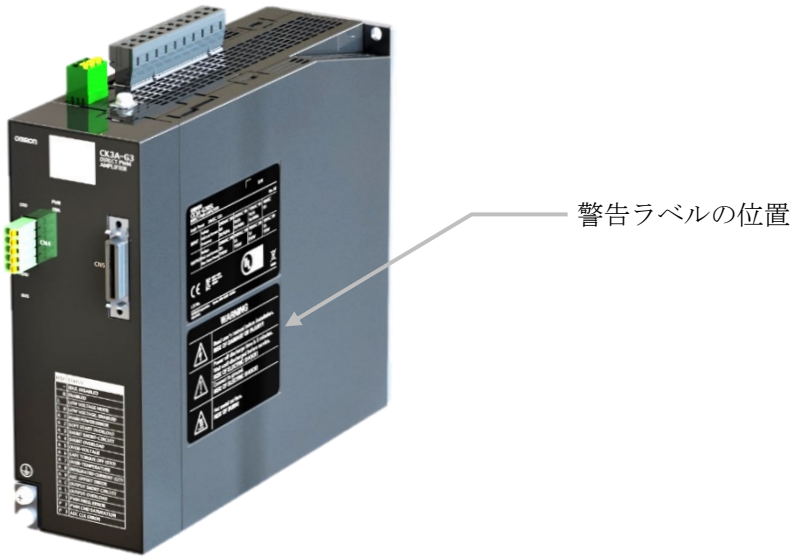
- 次のいずれかを実行する前に、必ず電源を OFF にしてください。

a) ユニットを着脱するとき。
b) ケーブルを接続、配線するとき。
c) 端子台やコネクタを取り付けたり、取り外したりするとき。

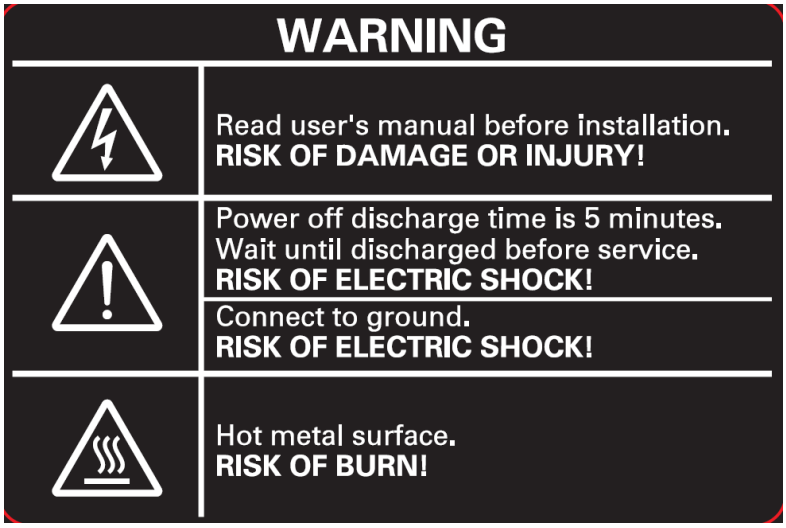
- 上記のすべてを行わない場合、重傷、機器の損傷、または誤動作を引き起こす可能性があります。
-
- 本製品を分解して修理や改造はしないでください。破損や誤動作、けがにつながるおそれがあります。

警告ラベル位置

アンプの以下の場所に警告ラベルが貼られています。本ラベルに記載する注意事項を必ず守ってください。



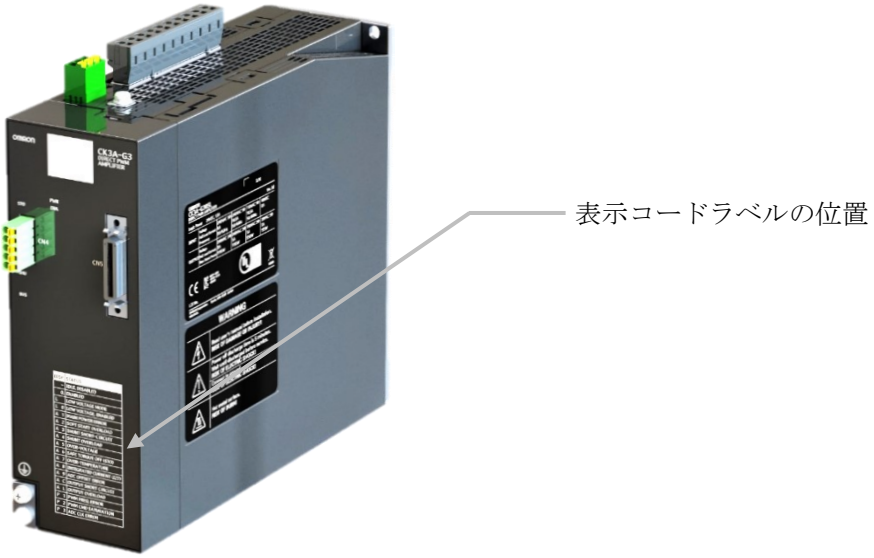
● **警告表示の注意事項**



（注）上記は警告表示の例です。

表示コード位置

アンプの以下の場所に表示コードラベルが貼られています。これは、手軽に参照していただくことを目的としています。表示コードの詳細については、本マニュアルの「第 6 章 異常の確認と処置」を参照してください。



● 表示コード情報

DISP.	STATUS
–	IDLE, DISABLED
0.	ENABLED
L	LOW VOLTAGE MODE
L 0	LOW VOLTAGE, ENABLED
A 1	MAIN POWER ERROR
A 2	SOFT START OVERLOAD
A 3	SHUNT SHORT-CIRCUIT
A 4	SHUNT OVERLOAD
A 5	OVER-VOLTAGE
A 6	SAFE TORQUE OFF (STO)
A 7	OVER-TEMPERATURE
A 8	INTEGRATED CURRENT (I2T)
A 9	ADC OFFSET ERROR
A C	OUTPUT SHORT CIRCUIT
A L	OUTPUT OVERLOAD
P 1	PWM FREQ. ERROR
P 2	PWM CMD SATURATION
P 3	ADC CLK ERROR

廃棄時

WEEE 指令に従って廃棄してください。

開梱後の確認事項

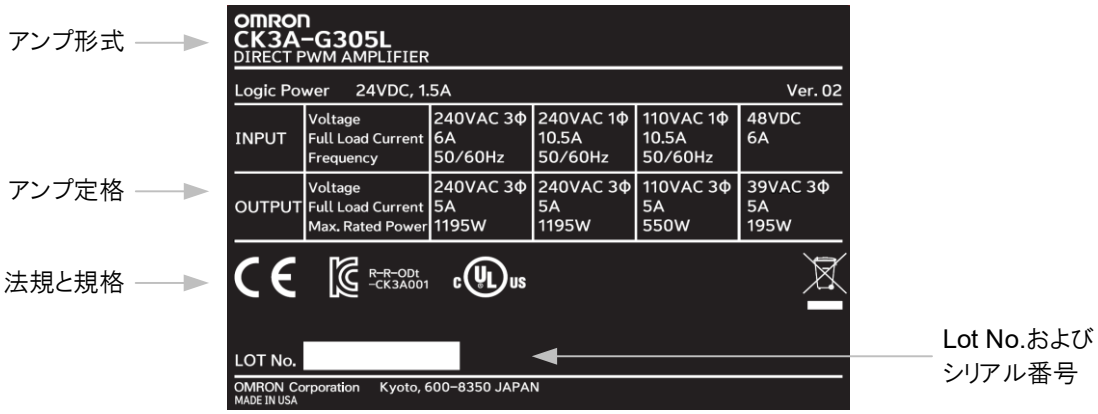
製品を開梱したら、次の項目を確認してください。

- これはあなたが注文した形式ですか？
- 輸送中に受けた損傷がありますか？

銘板位置

アンプの以下の場所に銘板ラベルが貼られています。

銘板情報



（注） 形 CK3A-G305L 銘板の例です。

付属品

アンプには以下のものが付属しています。

- 取扱説明書 × 1 部
- 警告ラベル × 1 枚
- General Compliance Information and instructions for EU × 1 部
- 付属コネクタ（下表参照）

UL/CSA 認証が必要な場合は、アンプ周辺に警告ラベルを貼ってください。

下表に記載以外のコネクタ、取り付けねじ、取付台などは付属しません。お客様が調達してください。

形式	主電源 接続コネクタ (CN1)	制御電源 接続コネクタ (CN2)	モータ 接続コネクタ (CN3)	STO 接続コネクタ (CN4)	PWM 接続コネクタ/ケーブル (CN5)
形 CK3A-G305L	搭載	搭載	搭載	搭載	非搭載
形 CK3A-G310L					
形 CK3A-G320L					

欠品やアンプ破損などの問題がある場合は、製品を購入したオムロン販売店または営業所にご連絡ください。

法規と規格

CK3A DirectPWM アンプが適合する規格について説明します。

EU 指令への適合

項目	規格
EMC 指令	EN61800-3 second environment
低電圧指令	EN61800-5-1 C2 category
機能安全	EN61800-5-2 SIL3 (STO)
RoHS	EN IEC 63000

UL 指令への適合

項目	規格
UL 規格	UL 61800-5-1
CSA 規格	CSA C22.2 No. 274

UKCA 規格への適合

項目	規格
UKCA	2016 No. 1091
UKCA	2016 No. 1101
EMC 指令	2016 No. 1091
低電圧指令	2016 No. 1101
機能安全	2008 No. 1597
RoHS	2012 No. 3032

KC 規格への適合

項目	規格
工業環境におけるイミュニティ規格	KS C 9610-6-2
工業環境におけるエミッション規格	KS C 9610-6-4

本製品を韓国で使用する場合には、以下の注意事項を守ってください。

사 용 자 안 내 문
이 기기는 업무용 환경에서 사용할 목적으로 적합성평가를 받은 기기로서 가정용 환경에서 사용하는 경우 전파간섭의 우려가 있습니다.

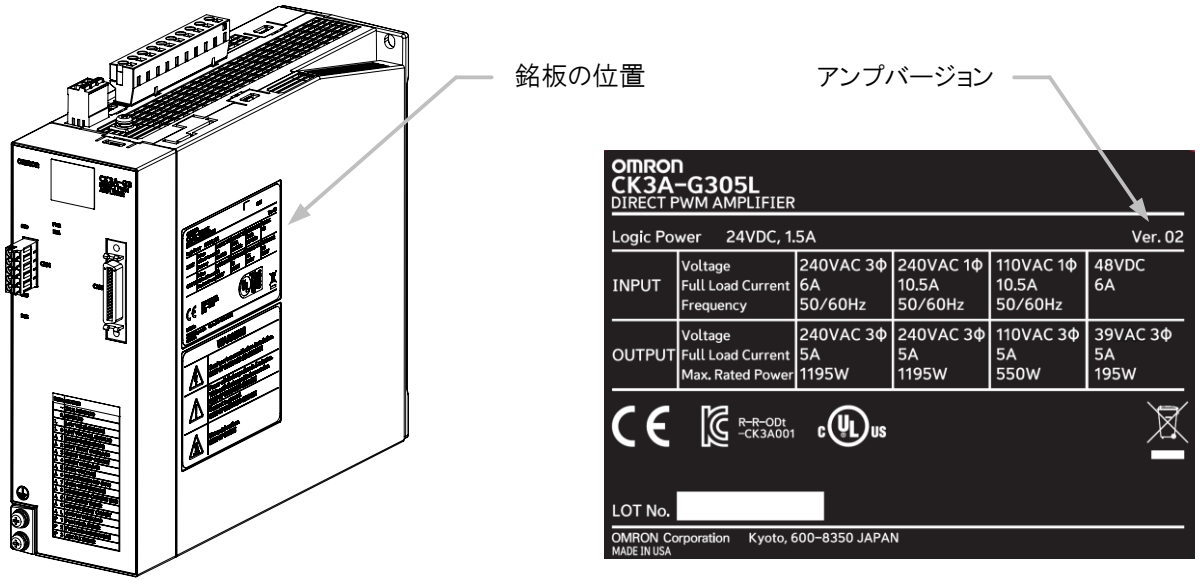
この機器は業務用環境で使用する目的で適合性評価を受けた機器のため、家庭用環境で使用する場合、電波干渉のおそれがあります。

バージョン

CK3A DirectPWM アンプはユニットバージョンを使用しています。ユニットバージョンは、製品の修正、アップグレードなどの違いを管理するのに役立ちます。

ハードウェアバージョン

CK3A DirectPWM アンプのハードウェアユニットバージョンは銘板に表示しています。



ファームウェアバージョン

アンプのファームウェアバージョンは、本マニュアルの「第 5 章 ソフトウェア構成」の手順に従って確認することができます。

関連マニュアル

併せてご覧いただく関連マニュアルは、下表のとおりです。入手方法は当社営業にお問い合わせください。

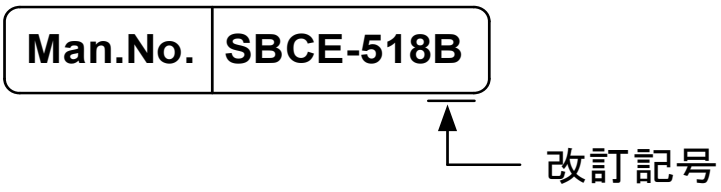
マニュアル名称	Man.No.	用途	内容
CK3A シリーズ DirectPWM アンプ ユーザーズマニュアル (本書)	SBCE-518	設置、配線、基本的なソフトウェア構成、メンテナンス、トラブルシューティングなど、仕様について学習します。	• アンプの概要 • 構成や機能と仕様 • 取り付けや設置と配線 • 基本的なソフトウェア構成 • 保守と異常の確認と処置
CK3M シリーズ プログラマブル多軸モーションコントローラ ユーザーズマニュアル ハードウェア編	SBCE-435	CK3M の概要／設計／取付／保守などの基本的な仕様について知りたいとき。おもにハードウェアに関する情報。	• 特長やシステム構成 • 概要 • 各部の名称と機能 • 一般仕様 • 設置と配線 • 保守点検
Power PMAC ソフトウエアリファレンスマニュアル	SBCE-405	PowerPMAC コントローラのコマンドセットと構造体要素について知りたいとき。	• Power PMAC データ構造 • すべてのコマンド一覧と説明 • CK3M および UMAC を含む、すべての ASIC、座標系、およびモータ構造体要素の一覧と説明
Power PMAC ユーザーズマニュアル	SBCE-404	Power PMAC コントローラの機能と使用例について知りたいとき。	• アンプに関連するパラメータ設定 • モータの基本機能 • エンコーダ構成例 • モータ設定例 • Power PMAC プログラミング例
Power PMAC IDE(V4) ユーザーズマニュアル	SBCE-439	Power PMAC コントローラの統合開発環境 IDE の使用方法を知りたいとき。	• Power PMAC IDE ソフトウェアの操作手順 • システム設定を使用した DirectPWM アンプの構成
ACC-24E3 ユーザーズマニュアル	SBCE-449	UMAC アクセサリ ACC-24E3 の概要／設計／取付／保守などの基本的な仕様について知りたいとき。	• 特長やシステム構成 • 概要 • 各部の名称と機能 • 一般仕様 • 設置と配線

用語説明

用語	説明
Power PMAC	• PMAC は Programmable Multi Axis Controller の略。 • Power は、本マニュアル作成時の CPU の世代を指す。
CK3M	CK3M シリーズ プログラマブル多軸モーションコントローラ（Power PMAC CPU 搭載）
UMAC	• Universal Machine and Automation Controller の略。 • 通常、Power PMAC CPU とその他のアクセサリを含むラック。 • Delta Tau Data Systems 社で設計された。
Gate3	DirectPWM オプションを持つことができる CK3M/UMAC 軸サーボカード内の Delta Tau ASIC。
コントローラ	本マニュアルでは、Gate3 ASIC を搭載した Power PMAC コントローラの略語として使用。
アンプ	本マニュアルでは、CK3A DirectPWM アンプの略語として使用。
IDE	• Integrated Development Environment の略。 • Power PMAC コントローラのプログラミングに使用する主要ソフトウェア。
F.L.A.	Full Load Amperage の略。
IPM	• Intelligent Power Module の略。 • 本マニュアルでは、アンプのパワーエレクトロニクス回路を指す。
FPGA	フィールドプログラマブルゲートアレイ
ADC	• アナログデジタルコンバータ • 本マニュアルでは、アンプの ADC 電流センサを指す。
ADC ストローブワード	Power PMAC 構造体要素 AdcAmpStrobe を指す。
PWM	Pulse Width Modulation の略。
DirectPWM	本マニュアルでは、コントローラ-CK3A インタフェースを指す。
主回路電源	アンプへの AC 主電源を指す。
ロジック電源	アンプへの DC24V 制御電源を指す。
バス電圧	アンプ内の DC バス電圧を指す。
PLC	• Programmable Logic Controller の略。 • 本マニュアルでは、Power PMAC PLC プログラムを指す。
I2T	熱保護モデル、アンプ内蔵 I2T または PMAC 構成可能 I2T を指す。
ECT	Encoder Conversion Table の略。

マニュアル改訂履歴

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



改訂記号	改訂年月	改訂理由･改訂ページ
A	2022 年 4 月	初版
B	2025 年 2 月	CK3A-G320L 形式追加 UKCA 規格追加、UL マークの修正、安全情報の追加

1

アンプの概要

本章では、CK3A DirectPWM アンプの機能と全体的な構成、各部品の名前、操作手順について説明します。

1-1 概要 1-2
1-1-1 アンプの機能 1-2
1-1-2 基本構成 1-3
1-1-3 各部の名称と配置 1-4
1-1-4 各部の機能 1-8
1-2 使用手順 1-10
1-2-1 準備 1-10
1-2-2 安全性 1-10
1-2-3 取り付けと配線 1-10
1-2-4 安全性試験と電源投入 1-10
1-2-5 アンプのコントローラ設定 1-11
1-2-6 試運転 1-11

1-1 概要

ここでは、CK3A DirectPWM アンプの機能と基本構成について説明します。

1-1-1 アンプの機能

製品について

CK3A DirectPWM（以下、アンプと呼ぶこともあります）は、デジタル DirectPWM サーボカードとインタフェースするように設計された一連のアンプ製品であり、通常、オムロン株式会社および Delta Tau Data Systems 社（以降は、「Delta Tau 社」と省略します）製の Turbo（従来品）および Power PMAC コントローラと統合します。

超低レイテンシサーボ制御

本製品は、デジタル DirectPWM を使用して、ASIC（DSPGATE）マシンインタフェースを介してコントローラから直接実際のパワートランジスタの ON/OFF 信号を受け取り、閉ループサーボ制御用のデジタル位相電流フィードバックとステータスを提供します。

本アーキテクチャは、産業用アプリケーション向けの最速のコントローラ-アンプインタフェースの 1 つを提供し、超高速のサーボ更新レートを可能にします。

柔軟性

CK3A アンプは、Power PMAC コントローラの柔軟性と、さまざまなエンコーダフィードバックデバイスとのインタフェース機能を組み合わせることで、以下の種類のモータに対応可能です。

- AC/DC 同期ブラシレス（ロータリまたはリニア）モータ – サーボモータなど。
- DC ブラシモータ – ボイスコイルアクチュエータなど。
- AC 非同期 – 誘導モータなど（この種類のモータのセットアップについてはサポートにお問い合わせください）。

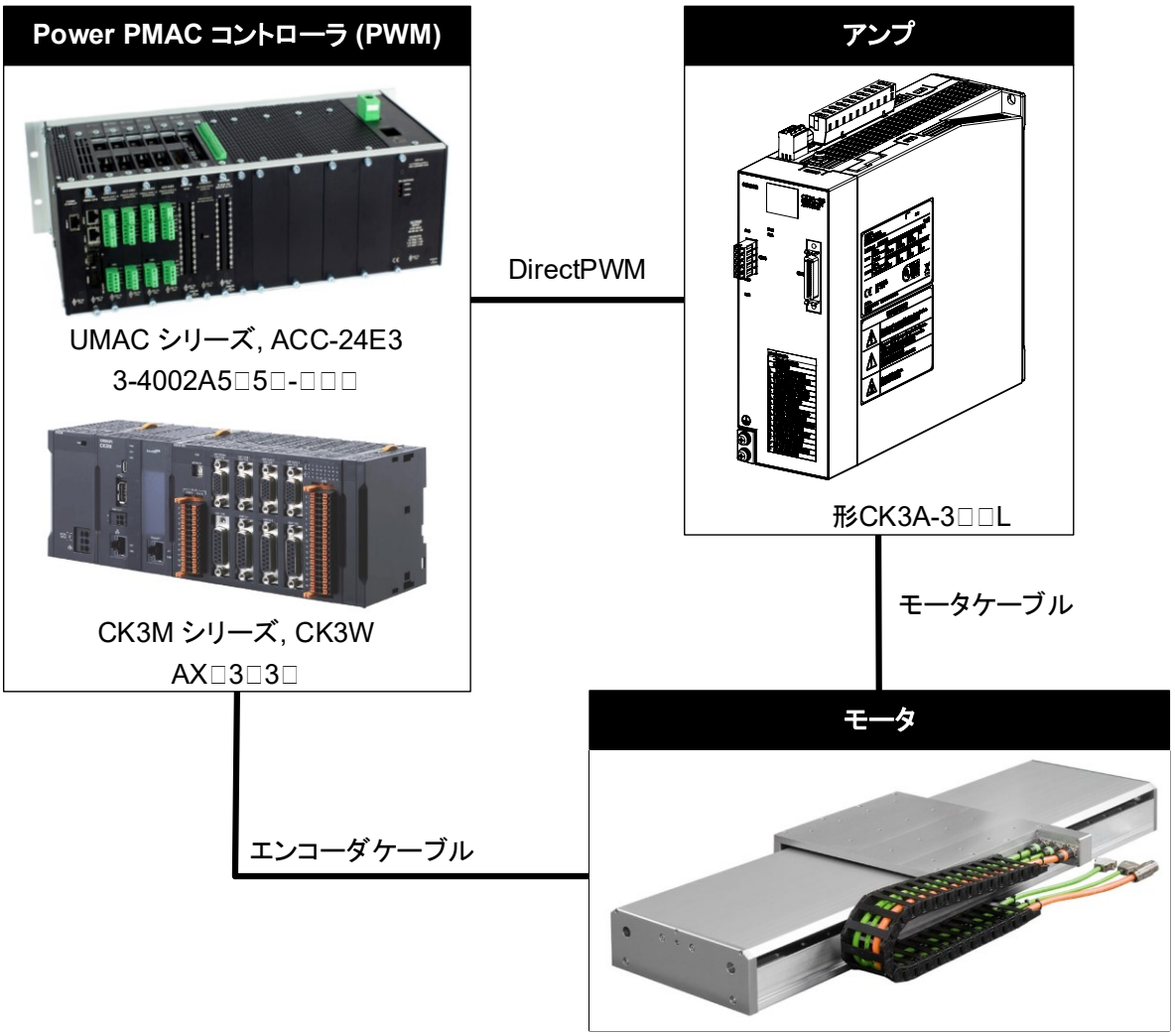
アンプの機能

- ナノスケールリニアサーボ位置決め精度
- 高 PWM 周波数 – 最大 20kHz
- 高分解能電流検出 – 16 ビット ADC
- 高速電流 ADC クロック – 最大 6.125MHz
- デュアル STO 入力
- STO 状態出力
- 基本機能 – エネルギー放電、ダイナミックブレーキ、ファン制御
- 基本データ報告 – DC バス電圧、パワーモジュール温度、ファームウェアバージョン
- デュアル 7 セグメント LED の状態表示
- 内部または外部シャント抵抗
- 低電圧主電源動作対応 (CK3A-G305L 及び CK3A-G320L)

1-1-2 基本構成

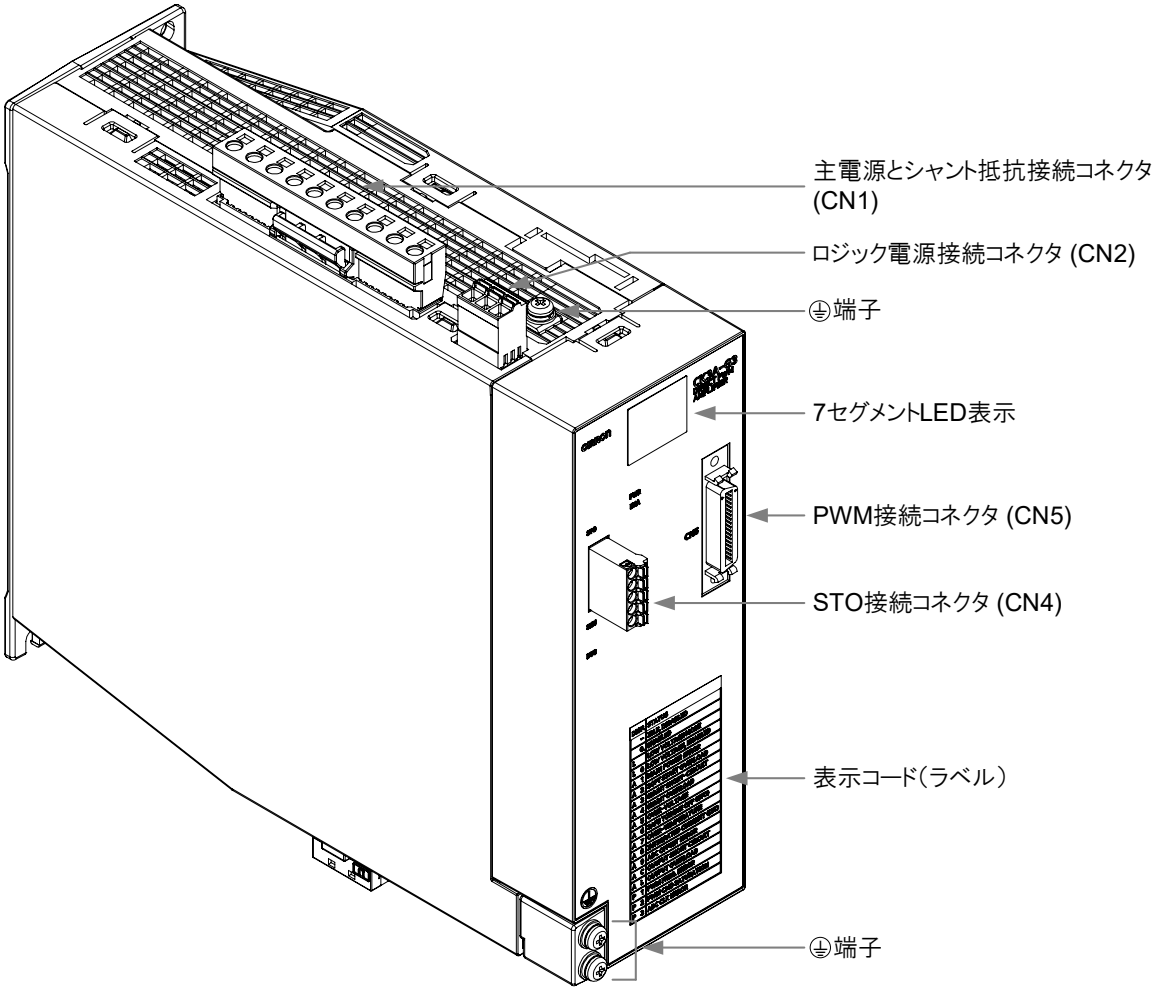
CK3A DirectPWM アンプは、PWM ケーブルを介して CK3M または UMAC コントローラに接続します。

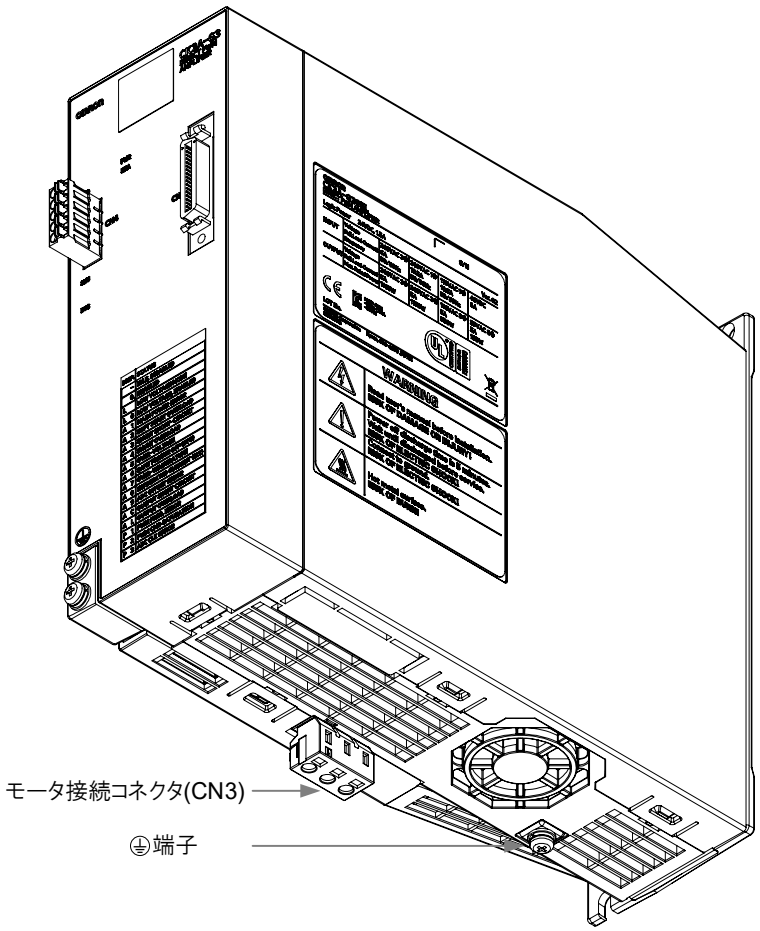
- モータは CK3A DirectPWM アンプに直接接続します。
- エンコーダはコントローラに直接接続します。

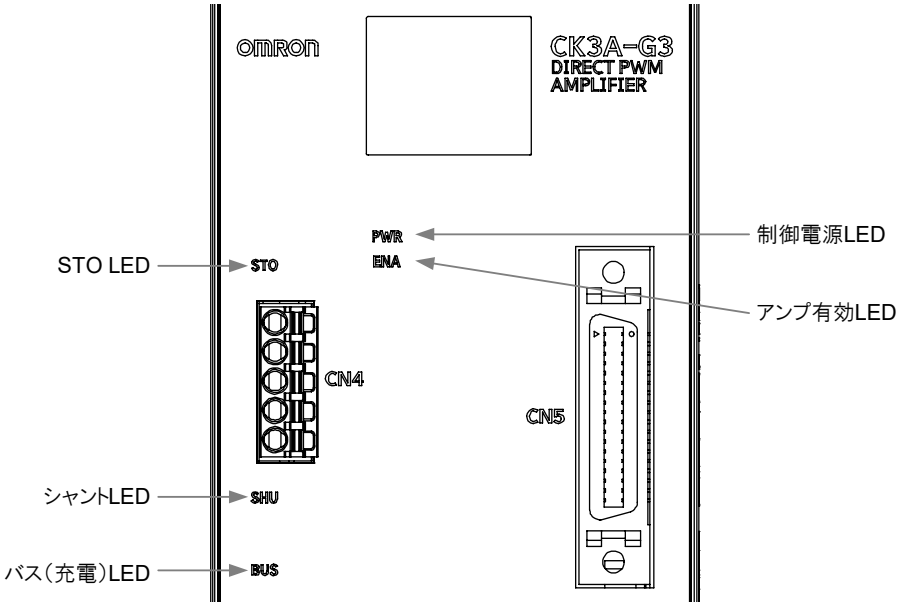


1-1-3 各部の名称と配置

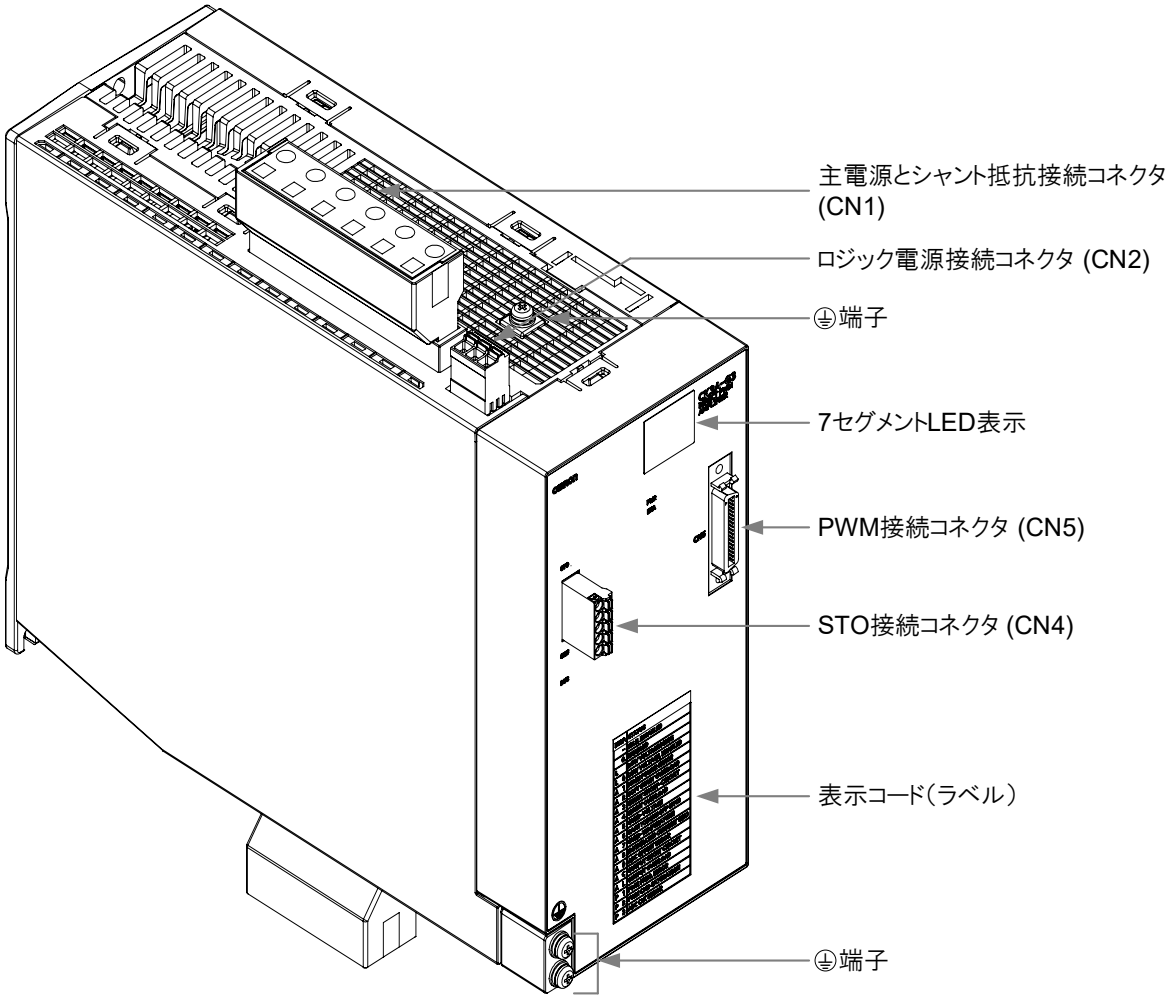
形 CK3A-G305L

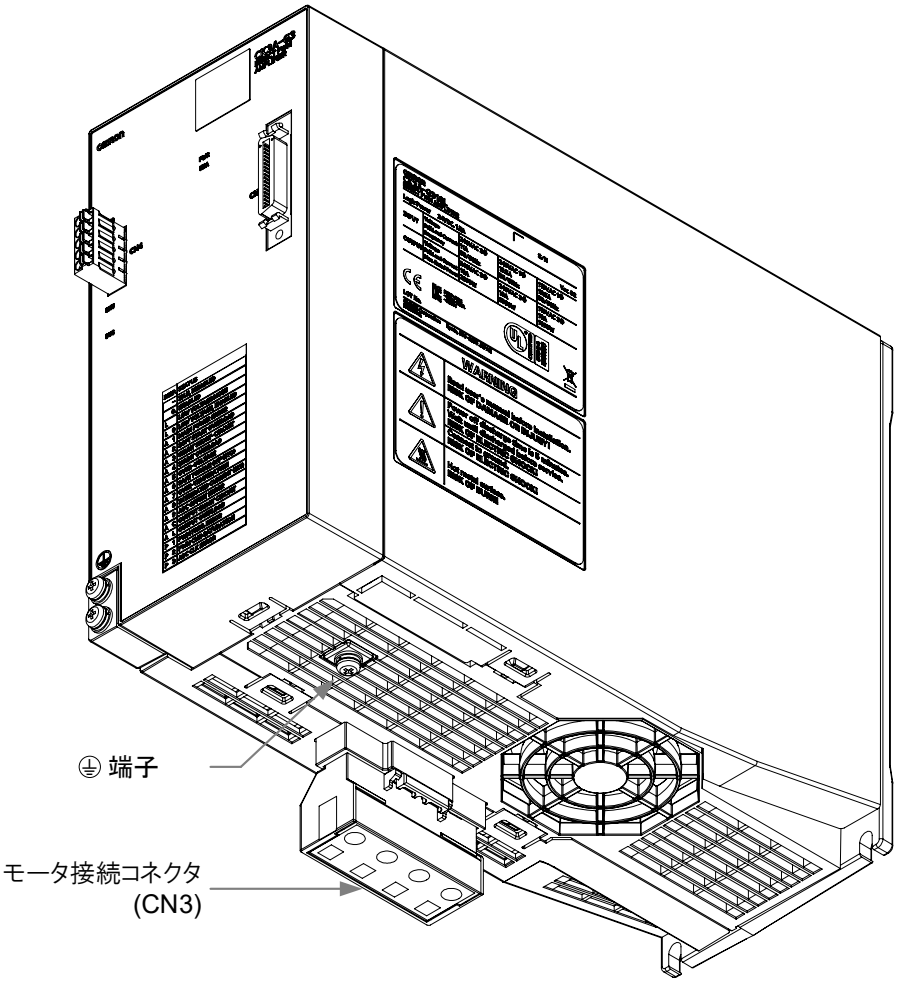


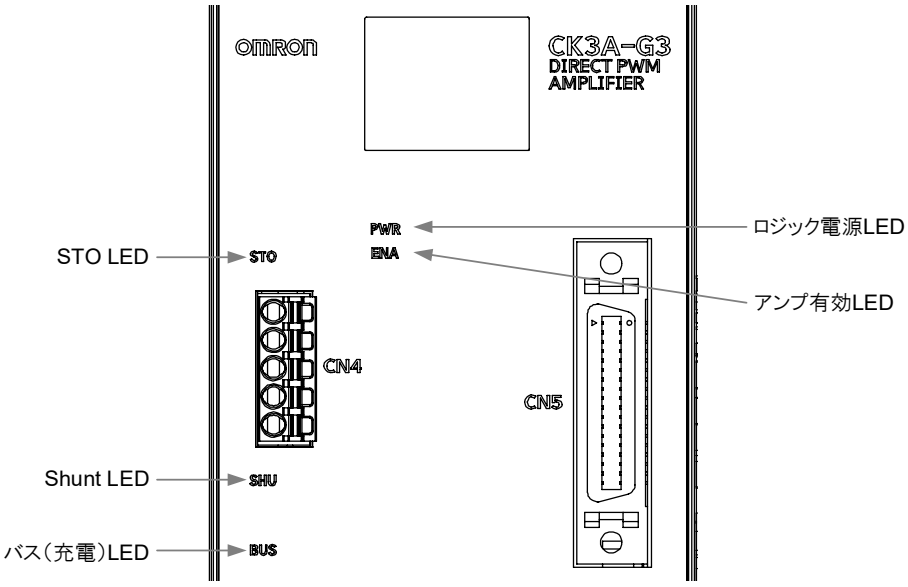

モータ接続コネクタ(CN3)
⏚端子


omron
CK3A-G3
DIRECT PWM
AMPLIFIER
PWR
ENA
STO
CN4
CN5
SHU
BUS
STO LED
制御電源LED
アンプ有効LED
シャントLED
バス(充電)LED

形 CK3A-G310L/CK3A-G320L




⏚ 端子
モータ接続コネクタ
(CN3)


omron
CK3A-G3
DIRECT PWM
AMPLIFIER
PWR
ENA
ロジック電源LED
アンプ有効LED
STO LED
STO
CN4
CN5
Shunt LED
SHU
バス(充電)LED
BUS

1-1-4 各部の機能

状態表示 LED

下記の LED は、アンプの前面にあります。

名称	表示色	内容
PWR	緑	ロジック電源の状態を示します
ENA	緑	アンプの有効状態を示します
SHU	黄	シャント動作がアクティブかどうかを示します
STO	赤/緑	STO 入力状態を示します
BUS	赤	バスの充電状態を示します

詳細は、「第 6 章 異常の確認と処置」を参照してください。

7 セグメント LED 表示

デュアル 7 セグメント表示はアンプの前面にあります。この表示は、アンプの動作と異常の状態を表示します。電源投入時に、7 セグメント表示はすべての表示を 6 回スクロールします。エラーコードの詳細については、「第 6 章 異常の確認と処置」をご覧ください。

参考

アンプが OFF のときは、LED が見づらい場合があります。表示が白い面として表示される場合があります。

主回路とシャント接続コネクタ（CN1）

CN1 コネクタは、次の機能に使用します。

- **形 CK3A-G305L（10 ピン）**
  - 主回路電源接続
  - 内部シャント抵抗接続（短絡線による）、または外部シャント抵抗接続
  - 低電圧モード（短絡線による）
- **形 CK3A-G310L、形 CK3A-G320L（6 ピン）**
  - 主回路電源接続
  - 内部シャント抵抗接続（短絡線による）、または外部シャント抵抗接続

ロジック電源接続コネクタ（CN2）

CN2 コネクタ（3 ピン）は、DC24V ロジック電源をアンプに供給するために使用します。

モータ接続コネクタ（CN3）

CN3 コネクタは、モータをアンプに接続するために使用します。形 CK3A-G305L では 3 ピンコネクタ、形 CK3A-G310L と形 CK3A-G320L では 4 ピンコネクタです。

安全トルク OFF コネクタ（CN4）

CN4 コネクタ（5 ピン）は、STO 入力を無効化または接続するため、および STO 状態出力に使用します。STO を無効にするための短絡線は、工場出荷時にコネクタに取り付け済みです

DirectPWM 接続コネクタ（CN5）

CN5 コネクタ（36 ピン）は、アンプをコントローラに接続するために使用します。これは、事前構成済みのケーブルコネクタです。

接地端子

アンプの接地⏚端子の位置と数は下表のとおりです。

位置	端子数	用途案
上面	1	主回路電源入力の保護接地端子（PE）
正面	2	制御盤内の保安用接地（FG）
底面	1	モータケーブルとシールドの保安用接地（FG）

1-2 使用手順

ここでは、CK3A DirectPWM アンプをシステムに統合するために必要な手順について説明します。

1-2-1 準備

項目	備考
Power PMAC コントローラ	システム構成に PWM オプションが表示されている必要があります
ロジック電源	本マニュアルに記載する仕様の範囲内である必要があります
主回路電源	本マニュアルに記載する仕様の範囲内である必要があります
モータ仕様	必要な動作条件とアンプ仕様の範囲内である必要があります
寸法、盤サイズ、場所	本マニュアルに記載する条件と仕様の範囲内である必要があります

1-2-2 安全性

項目	備考
全体的な安全性	リスク評価に基づいて安全対策を決定します
警告と注意の確認	本マニュアルの「安全上の注意」に記載する通知を確認します
設計	機械全体の安全性と警告に基づいて安全対策を設計します

1-2-3 取り付けと配線

項目	備考
取り付け	本マニュアルに記載する取り付けガイドラインに従います ● 人間工学的および環境的な配置 ● 取り付け方向 ● 単一ユニットのクリアランス ● 複数ユニットのクリアランス
配線	本マニュアルに記載されている配線図とガイドラインを使用します ● 安全対策 ● 正しいピン配置 ● 電線ゲージ ● 人間工学に基づくケーブル管理

1-2-4 安全性試験と電源投入

項目	備考
安全性試験	導入した ON/OFF 安全対策を試験します（STO、E-Stop など）
安全上の要点	本マニュアルの「安全上の要点」を確認します
電源投入	「安全上の要点」に記載する電源投入時の手順を参照します

1-2-5 アンプのコントローラ設定

安全上の要点

この手順は、主回路の電源をオフにして実行してください。

項目	備考
準備	● 使用中のコンピュータに Power PMAC IDE ソフトウェアをインストールします ● 本マニュアルに記載する関連マニュアルの一覧を確認します
Power PMAC コントローラ	● Power PMAC と IDE ソフトウェア間の通信を確立します ● 既存のプロジェクトをロードするか、工場出荷時のデフォルト設定から開始します ● コントローラの入力画面要素または IDE システム設定に関連するマニュアルを使用します
エンコーダ設定	エンコーダ仕様書と関連マニュアルを使用して、エンコーダ固有の構成と設定を行います
アンプ設定	本マニュアルに記載するアンプ固有の設定を行います
モータ設定	モータ仕様書と関連マニュアルを使用して、モータ固有の構成と設定を行います

1-2-6 試運転

項目	備考
エンコーダフィードバックの確認	関連するハードウェアまたはユーザーズマニュアルを使用します
ON/OFF 有効	● コントローラの電流ループゲインをゼロに設定して、ON と OFF（開ループ）を有効にします ● 本マニュアルで記載する状態表示ごとにアンプ異常がないことを確認してください
モータ設定	関連するハードウェア、ユーザーズマニュアル、または IDE システム設定を使用します
サーボ ON/OFF	サーボ ON/OFF モータ動作を確認します

参考

または、IDE の自動システム設定を使用してこの手順を実行することもできます。

2

仕様

本章では、CK3A DirectPWM アンプの一般的な仕様について説明します。

2-1 アンプの仕様 2-2
2-1-1 一般/機械 2-2
2-1-2 環境 2-2
2-1-3 電気 2-3
2-1-4 性能 2-5
2-1-5 回生エネルギー吸収 2-6
2-2 機能とデータ報告 2-10
2-2-1 アンプ機能 2-10
2-2-2 データ報告 2-11
2-2-3 ステータスビット 2-11
2-3 部品番号表示 2-12

2-1　アンプの仕様

ここでは、アンプの一般的、および機械的、環境的、機能的、電気的仕様について説明します。

2-1-1　一般/機械

項目		仕様
軸数		1
構造		盤内内蔵型
保護外郭		IP20（IP54 盤内設置）
接地		200V D 種接地（第 3 種接地）
耐振動		5.88m/s^2 以下の加速度で 10～60Hz （共振周波数で連続運転しないでください）
通風用クリアランス		「3-2　設置」を参照してください。
取り付けねじ締付トルク		1.2N·m
冷却		自然対流と内蔵ファン
質量	形 CK3A-G305L	1.81kg
	形 CK3A-G310L	2.67kg
	形 CK3A-G320L	2.77kg
寸法	形 CK3A-G305L	65.0W × 180.0H × 212.5D（単位：mm）
	形 CK3A-G310L	90.0W × 180.0H × 238.0D（単位：mm）
	形 CK3A-G320L	90.0W × 180.0H × 238.0D（単位：mm）

2-1-2　環境

項目	仕様
使用周囲温度	0～55℃
使用周囲湿度	10～90%RH（結露・氷結がないこと）
保存周囲温度	-25～70℃
保存周囲湿度	10～90%RH（結露・氷結がないこと）
使用兼保存周囲雰囲気	腐食性ガスがないこと
最大使用高度	2,000m
汚染度	2

2-1-3 電気

以下に形 CK3A-G3□□L それぞれの主要な電気仕様について詳しく説明します。

	項目		形 CK3A-G305L	形 CK3A-G310L	形 CK3A-G320L
ロジック電源	電圧		DC24V*1		
	消費電流		1.5A		
	突入電流		2.5A		
	突入時間		5ms		
主回路電源	三相 AC	電圧	AC240V± 5%*2		
		F.L.A.	$6A_{rms}$	$11A_{rms}$	$18A_{rms}$
		周波数	50/60Hz		
	単相 AC	電圧	AC240V± 5%*2 AC110V± 5%*3		AC240V± 5%*2 *7
		F.L.A.	$10.5A_{rms}$	$19.5A_{rms}$	$28A_{rms}$
		周波数	50/60Hz		
	単相 DC *4	(低)電圧	DC48V *5	-	DC100V±10%*6
		F.L.A	$6A_{rms}$	-	19A
出力	定格電流		$5A_{RMS}$	$10A_{RMS}$	$20A_{RMS}$
	最大(ピーク)電流		$10A_{RMS}$	$20A_{RMS}$	$60A_{RMS}$
	最大定格電力(三相 AC)		1195W	2390W	4400W
	最大定格電力(単相 AC240V)		1195W	2390W	4400W
	最大定格電力(単相 AC110V)		550W	1095W	-
	最大定格電力(単相 DC)		195W	-	1600W
	ピーク電流の時間		2s		
PWM インタフェース	電流フィードバック分解能		16 ビット		
	最大電流 ADC 読み取り値		15.735A	31.470A	93.844A
	最小 PWM 不感時間		2μs	3μs	
	PWM 周波数		8～20kHz		8～16kHz*8
シャント抵抗	内部シャント抵抗		25 Ω 30W	17 Ω 80W	
	外部シャント抵抗		20 Ω 60W	17 Ω 60W	

*1. ロジック電源の入力電圧の許容変動範囲は DC22.0 ～ 26.4 V です。

*2. この主回路入力電圧の許容変動範囲は AC 85 ～ 170V です。

*3. 低電圧の主電源入力(単相 DC)で動作させるためには、すべての形式で ADC ストローブワードを正しく設定する必要があります。

*4. この主回路入力電圧の許容変動範囲は DC44 ～ 60V です。

*5. この主回路入力電圧の許容変動範囲は DC90 ～ 110V です。

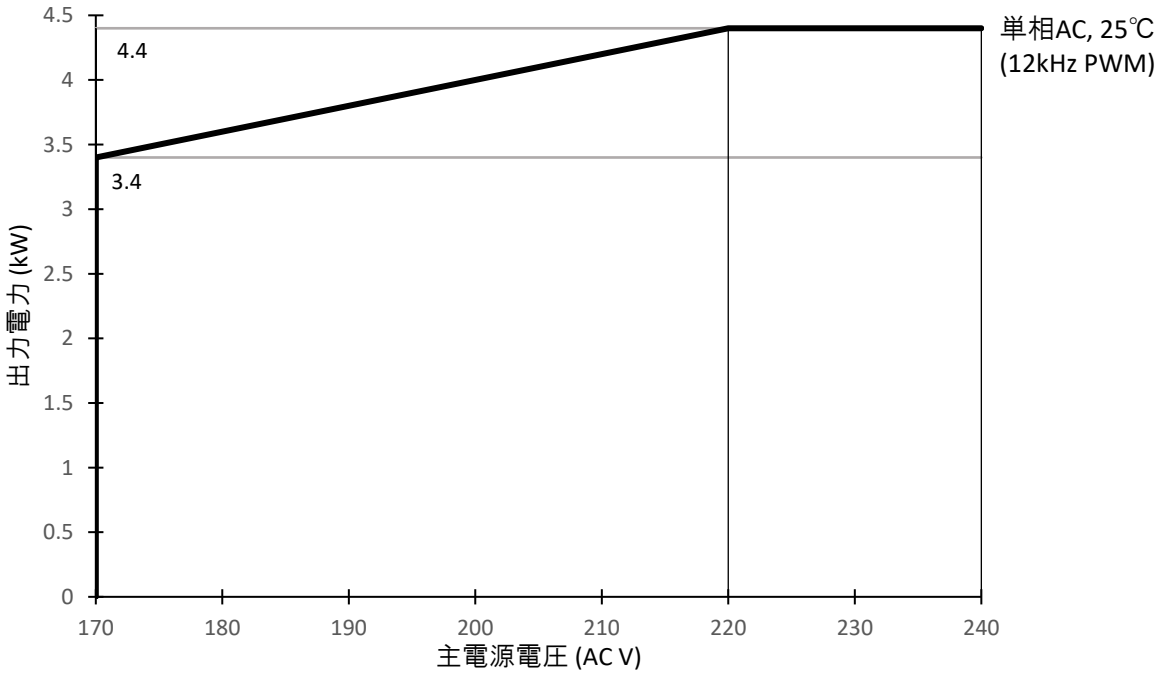
*6. CK3A-G320L は、単相 AC の主回路入力電圧で最大定格電力のディレーティングが必要になる場合があります。25℃でのディレーティング量を示す次のページの図を参照してください。

*7. 10kHz を超える PWM 周波数を使用する場合、形 CK3A-G320L では最大定格電力のディレーティングが必要になります。ディレーティング量を示す次のページの図を参照してください。

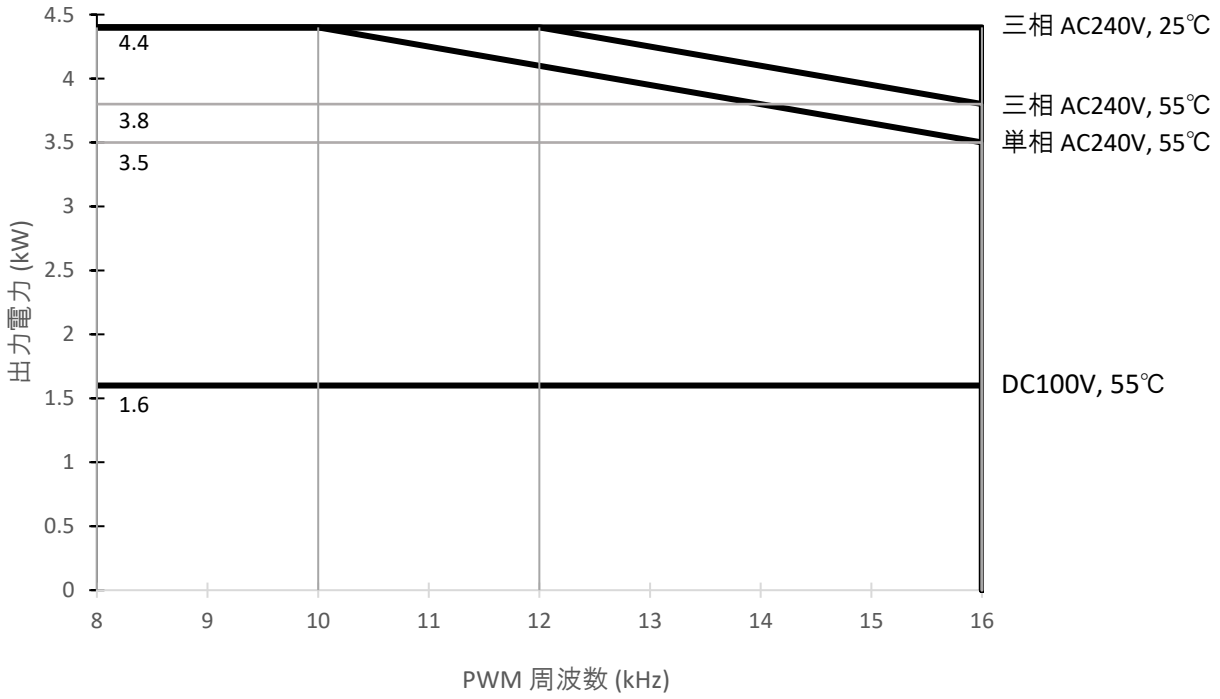
参考

ADC ストローブワードの設定に加えて、形 CK3A-G310L を低電圧(DC48V)主電源入力で動作させるには、特別な形式と、工場での製品調整が必要です。このオプションについては、オムロン営業担当者にお問い合わせください。

単相入力電圧時の電力出力ディレーティング（形 CK3A-G320L のみ）



PWM 周波数による電力出力ディレーティング（形 CK3A-G320L のみ）



2-1-4 性能

安全上の要点

安全を確保するために、機械に適切な停止装置を取り付けてください。ダイナミックブレーキ機構は停止装置ではありません。

使用上の注意

- ダイナミックブレーキ機能が ON の場合、アンプを有効にして、最低 50 ミリ秒待機してからモーションコマンドを使用してください。
- （動作中のモータ停止後）ダイナミックブレーキが実行された場合、最低 500 ミリ秒待機してからアンプを再度有効にしてください。

仕様	値	内容
パワードライバへの STO 入力 OFF	< 150ms	
過電流 I2T から IPM OFF	< 10ms	A8 異常
位相短絡から IPM OFF	< 3μs	AC 異常
電流ループ応答時間	< 1ms	1mH 三相ブラシレスモータ Y 巻線
ダイナミックブレーキリレー応答時間	< 20ms	機械的リレータイム定数
アンプ OFF までの I2T 時間	< 2.5s	形 CK3A-G305L: 200%出力 形 CK3A-G310L: 200%出力 形 CK3A-G320L: 300%出力
ソフトスタート時間	< 650ms	ソフトスタート中はアンプを有効にしないでください
瞬停時ホールド	10ms	三相 AC208V@定格負荷
DC バス電圧放電時間、放電 ON (DC36V 未満になるまでの放電時間)	< 2.5s	シャント抵抗による強制放電
DC バス電圧放電時間、放電 OFF (DC36V 未満になるまでの放電時間)	< 5min	自然放電 (形 CK3A-G305L)
	< 5min	自然放電 (形 CK3A-G310L)
	< 6min	自然放電 (形 CK3A-G320L)
電流 ADC クロック周波数範囲	2.450～6.250MHz	コントローラで設定してください
主回路電源周期時間、放電 ON	1min 以上	ソフトスタート回路または放電回路の過負荷を防ぐため
主回路電源周期時間、放電 OFF	10s 以上	ソフトスタート回路の過負荷を防ぐため

2-1-5 回生エネルギー吸収

アンプにはコンデンサが内蔵されており、モータ減速とダイナミックブレーキ中に逆起電力により生成した回生エネルギーを吸収します。内蔵コンデンサがすべての回生エネルギーを吸収できない場合、内部（または外部）シャント抵抗が残りのエネルギーを吸収します。

モータからの回生エネルギー量が多すぎる場合、回生動作を停止し、内部シャント抵抗の焼損を防ぐために過電圧異常（A5）でモータを無効にします。

回生の連続時間が 2 秒を超えると、回生動作を停止し、過熱や損傷を防ぐためにシャント過負荷異常（A3）でモータが無効になります。上記が発生した場合、クールダウンのため 5 分間待機してください。

このような場合、モータの回生エネルギーを削減する必要があります。ここでは、回生エネルギーの計算と推奨事項を説明します。

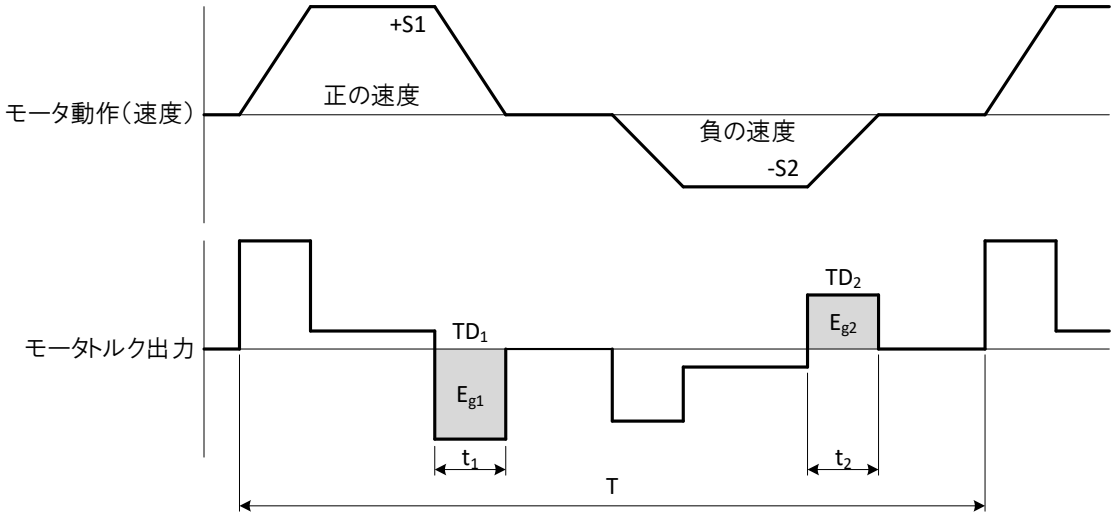
アンプ内回生吸収容量

次の表は、アンプの電力、内部シャント抵抗の仕様、回生電力の吸収容量、および最大持続時間を示しています。これらの値は、AC200V 主電源に基づいています。

形式	形 CK3A-G305L	形 CK3A-G310L	形 CK3A-G320L
定格 RMS 電力 [W]	1195W	2390W	4400W
内部シャント抵抗の仕様	25Ω 30W	17Ω 80W	
内蔵コンデンサの吸収エネルギー [J]	46J	62J	
内部シャント抵抗の平均回生エネルギー [W]	18W	32W	
最大連続回生時間 [s]	2s	2s	
最小許容シャント抵抗 [Ω]	20Ω	15Ω	

回生エネルギーの計算

● **横軸**



記号	内容
S1、S2	減速開始時のモータ速度、ロータリモータは[r/min]、リニアモータは[m/s]
TD_1、TD_2	減速トルク [N·m]
t_1、t_2	減速時間 [s]

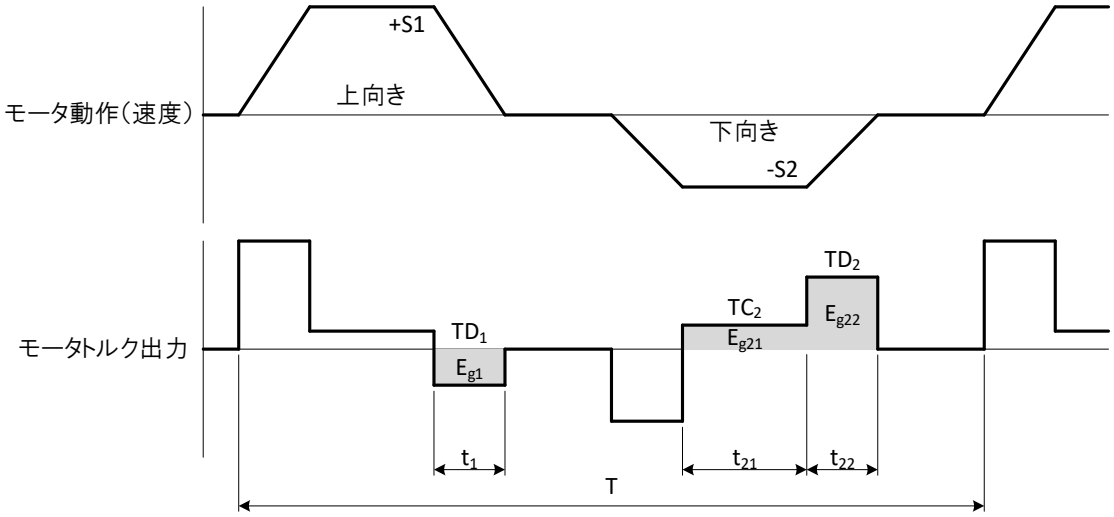
Eg_1、Eg_2 [J]の回生エネルギー吸収値は、次のように計算できます。

ロータリモータ回生エネルギー [J]	リニアモータ回生エネルギー [J]
$E_{g1} = \frac{1}{2} \times \frac{2\pi}{60} \times S1 \times TD_1 \times t_1$	$E_{g1} = \frac{1}{2} \times S1 \times T_1 \times t_1$
$E_{g2} = \frac{1}{2} \times \frac{2\pi}{60} \times S2 \times TD_2 \times t_2$	$E_{g2} = \frac{1}{2} \times S2 \times T_2 \times t_2$

参考

モータの巻線抵抗のため、実際の回生エネルギーは計算値の約 90%になります。

● 縦軸



記号	内容
S1、S2	減速開始時のモータ速度、ロータリモータは[r/min]、リニアモータは[m/s]
TD_1、TD_2	減速トルク [N·m]
TC_2	下向き定速トルク [N·m]
t_1、t_{22}	減速時間 [s]
t_{21}	下向き定速時間 [s]

Eg_1 および Eg_{21}、Eg_{22} [J]の各領域の回生エネルギー吸収値は、次のように計算できます。

ロータリモータ回生エネルギー [J]	リニアモータ回生エネルギー [J]
$E_{g1} = \frac{1}{2} \times \frac{2\pi}{60} \times S1 \times TD_1 \times t_1$	$E_{g1} = \frac{1}{2} \times S1 \times TD_1 \times t_1$
$E_{g21} = \frac{2\pi}{60} \times S2 \times TC_2 \times t_{21}$	$E_{g21} = S2 \times TC_2 \times t_{21}$
$E_{g22} = \frac{1}{2} \times \frac{2\pi}{60} \times S2 \times TD_2 \times t_{22}$	$E_{g22} = \frac{1}{2} \times S2 \times TD_2 \times t_{22}$
$E_{g2} = E_{g21} + E_{g22}$	

下向きの動きの総回生エネルギーは、$Eg_2=Eg_{21}+Eg_{22}$ であることに注意してください。

参考

モータの巻線抵抗のため、実際の回生エネルギーは計算値の約 90%になります。

必要回生エネルギーとアンプの吸収容量の関係

事前に計算した回生エネルギー E_{g1} と E_{g2} [J]の両方の値が、アンプのコンデンサの回生吸収容量 E_C [J]以下の場合、何もする必要はありません。

ただし、E_{g1} または E_{g2} [J]どちらかがアンプのコンデンサの回生吸収容量 E_C [J]よりも大きい場合は、次の式を使用して平均回生電力 P_r [W]を計算します。

$E_g = (E_{g1} – E_C) + (E_{g2} – E_C)$ [J]
$P_r = E_g / T$ [W]

記号	内容
P_r	1 周期動作における平均回生電力吸収値 [W]
E_g	1 周期動作における回生エネルギー吸収値 [J]
E_C	内蔵コンデンサの吸収エネルギー [J]
T	動作周期時間 [s]

（注）　$(E_{g1}–E_C)$、または $(E_{g2}–E_C)≤0$ の場合、0 を使用してください。

平均回生電力 Pr [W]がアンプの内蔵抵抗の平均回生電力以下の場合は、何もする必要はありません。

平均回生電力 Pr [W]がアンプの内蔵抵抗の平均回生電力より大きい場合は、以下の対策を 1 つ以上行うことができます。

- 外部シャント抵抗を接続する。
- モーションプロファイルの速度を下げる。回生エネルギーは、モータ速度の 2 乗に比例する。
- このため、モーションプロファイルの減速時間を短縮し、単位時間あたりの回生エネルギーを削減する。
- 動作周期時間を長くして、連続する移動の間隔を長くする。

2-2 機能とデータ報告

CK3A DirectPWM アンプは、PWM ケーブルを介してコントローラとインタフェースします。
アンプには、コントローラから設定できる機能があります。また、コントローラから読み取ることができるデータとステータスビットがあります。

使用上の注意

PWM ケーブルが抜かれているか、ロジック電源がオフの場合は、機能、データ、およびステータス情報にはアクセスできません。

参考

- アンプのデータ報告は、リアルタイムでの使用を目的としたものではありません。コントローラからの取得方法やチャネル数によっては、更新速度が数百ミリ秒になる場合があります。
- アンプのデータ報告は、プログラミング、トラブルシューティング、または機械オペレータのステータス表示中の汎用使用を目的としています。
- アンプの機能とデータ報告は、コントローラの ADC ストローブワードの設定によって制御されます。

2-2-1 アンプ機能

アンプの機能は下表のとおりです。

項目	機能オプション
冷却ファン制御	• 自動（アンプファームウェアによる）または • 常時 ON
ダイナミックブレーキ制御	• サーボ OFF（または異常）イベント後にモータの惰性運転（フリーラン）を許可するまたは • サーボ OFF（または異常）イベント後にモータを停止させる
異常解除機能	アンプ異常解除プロセス（ADC ストローブワードを使用）
低電圧モード制御	• 正常な AC 入力（仕様による）または • 低電圧 DC48V 主回路電源入力（形 CK3A-G305L のみ） • 低電圧 DC100V 主回路電源入力（形 CK3A-G320L のみ）
バス放電制御	• 主電源回路が OFF になっているとき、残留エネルギーを消費しないまたは • 主電源回路が OFF になっているとき、残留エネルギーを消費する

使用上の注意

ダイナミックブレーキは、故障状態の場合にだけモータを停止することを目的としています。ダイナミックブレーキ機能が ON の場合は、動作中のモータを頻繁に停止（サーボ OFF）しないでください。

2-2-2 データ報告

アンプは下表のデータを（コントローラに）報告することができます。

項目	仕様
DC バス電圧	DCV 単位の DC リンクバス電圧を報告します
パワーモジュール温度	パワーモジュール温度を報告します（℃単位）
ファームウェアバージョン	アンプのファームウェアバージョンを報告します
電流定格	アンプの電流定格を報告します（アンペア単位）

2-2-3 ステータスビット

アンプは下表のステータスビットを（コントローラに）報告することができます。

ステータスビットセット 1	ステータスビットセット 2	ステータスビットセット 3
主電源回路異常	主電源回路状態	I2T インテグレータ状態
ソフトスタート異常	過熱異常	ADC オフセット異常
STO 状態	シャント過負荷異常	アンプレディ状態
シャント短絡異常	PWM 周波数異常	-
過電圧異常	PWM コマンド異常	-
I2T 異常	アンプ有効状態	-
短絡異常	-	-
過負荷異常	-	-

2-3　部品番号表示

部品番号の表示は以下のとおりです。

アンプ部品番号

C	K	3	A	-	G	3			L

05:	5A 連続/10A ピーク
10:	10A 連続/20A ピーク
20:	20A 連続/60A ピーク

3

取り付けと設置

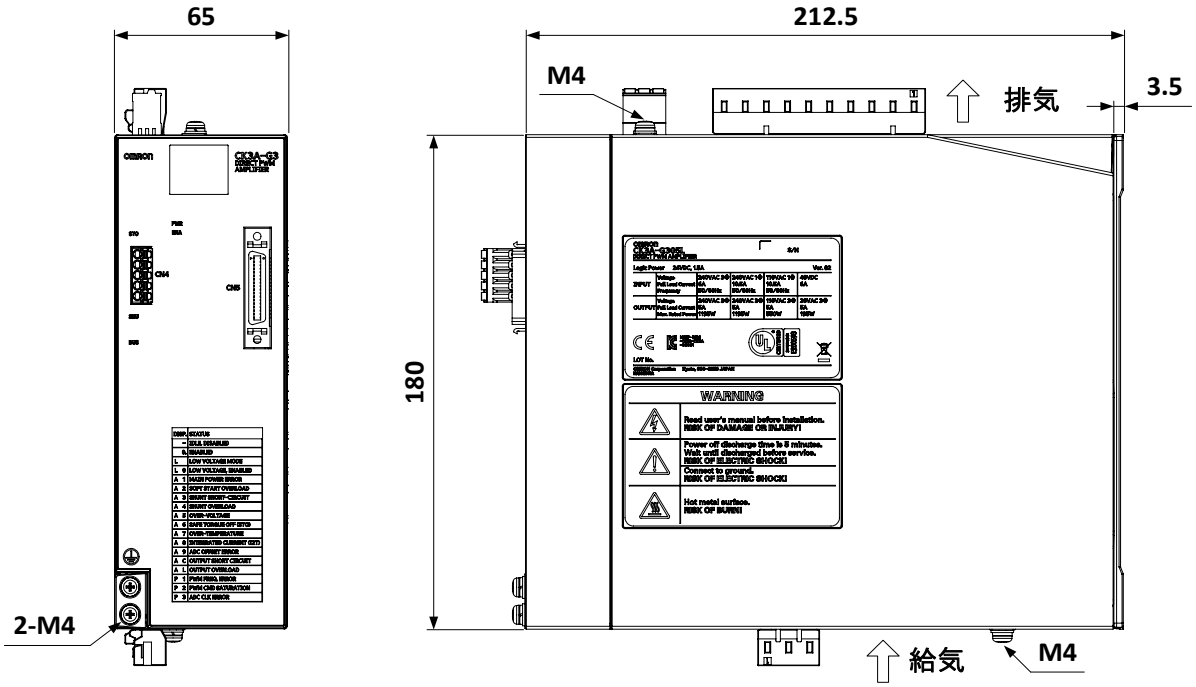
本章では、CK3A DirectPWM アンプの取り付け方法と設置方法について説明します。すべての寸法はミリメートルです。

3-1 外形寸法と取り付け寸法 3-2
3-1-1 形 CK3A-G305L の寸法と取り付け 3-2
3-1-2 形 CK3A-G310L の寸法と取り付け 3-3
3-2 設置 3-4
3-2-1 換気 3-4
3-2-2 盤内クリアランス 3-4
3-2-3 取り付け方向 3-6
3-2-4 設置条件 3-7
3-2-5 異物をユニットに入れない 3-7

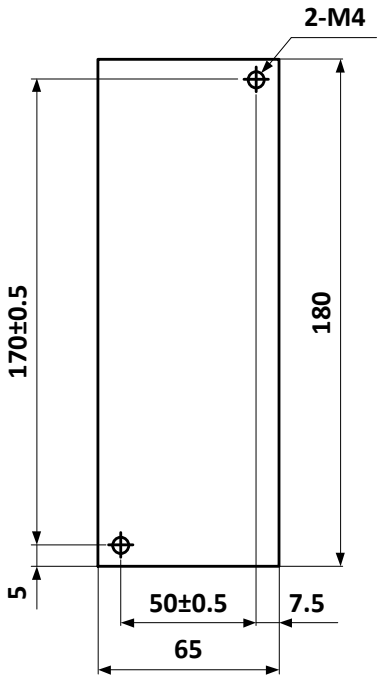
3-1　外形寸法と取り付け寸法

3-1-1　形 CK3A-G305L の寸法と取り付け

外形寸法

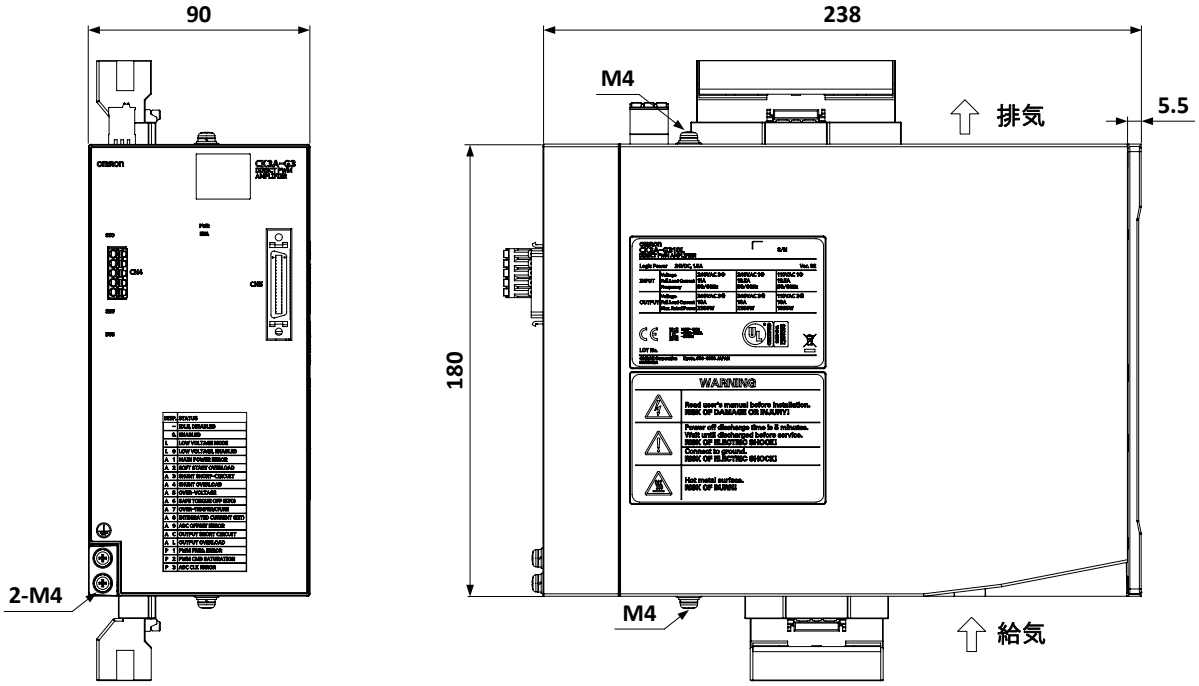


取り付け寸法

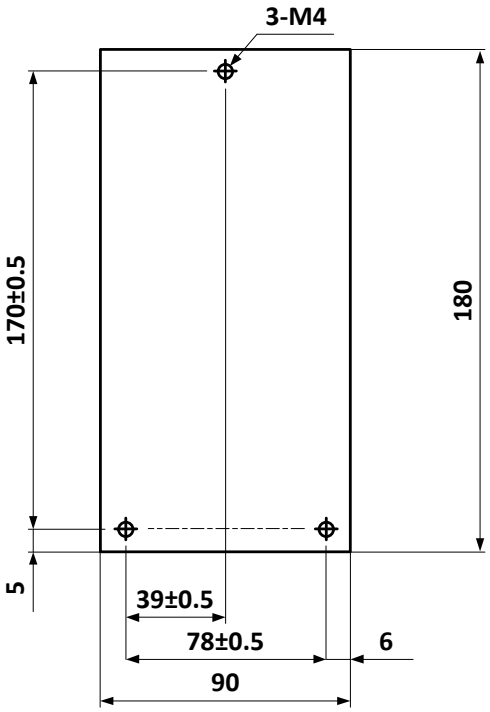


3-1-2　形 CK3A-G310L、形 CK3A-G320L の寸法と取り付け

外形寸法



取り付け寸法



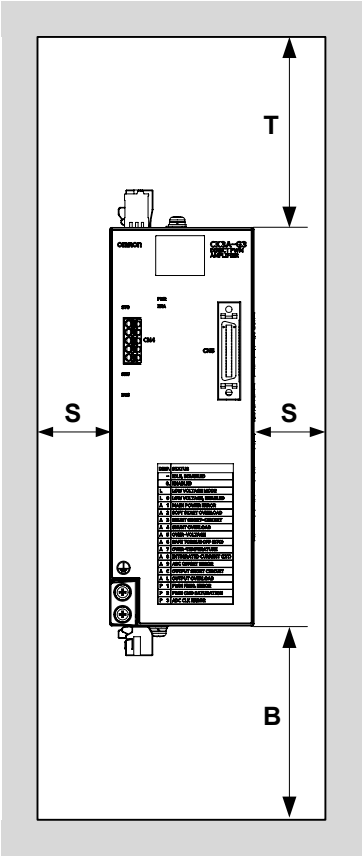
3-2　設置

3-2-1　換気

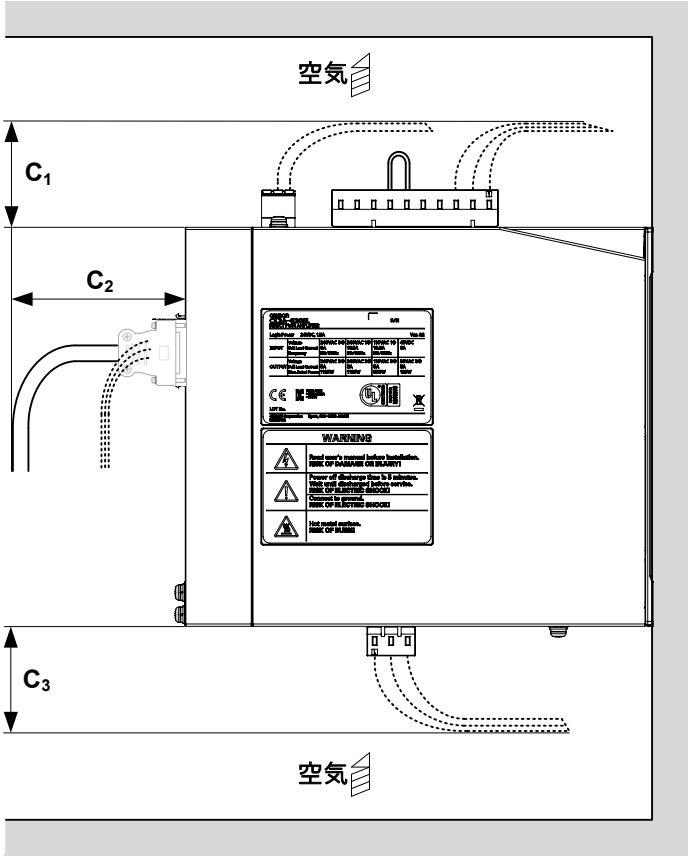
下記の寸法仕様に従いアンプを設置してください。アンプからの熱が適切に放散するよう盤内の対流を確保してください。アンプを並べて設置する場合は、盤内の温度むらを防ぐために、換気システムか空気循環用のファンを設置してください。

3-2-2　盤内クリアランス

単一ユニット設置の場合

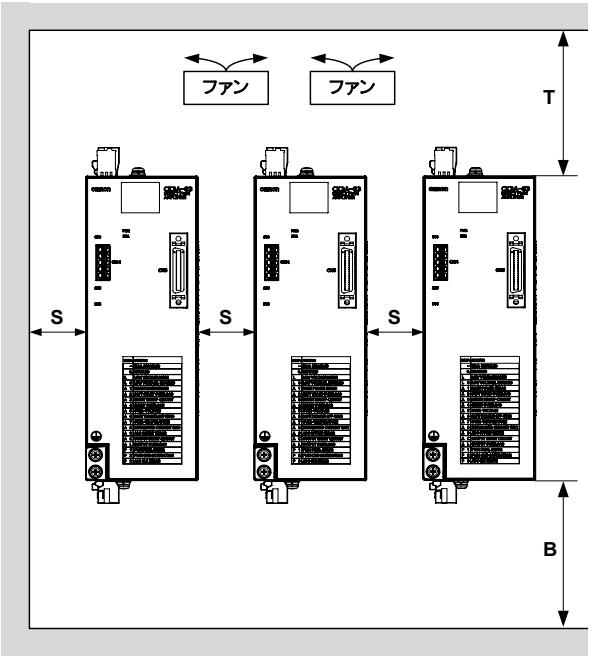


正面

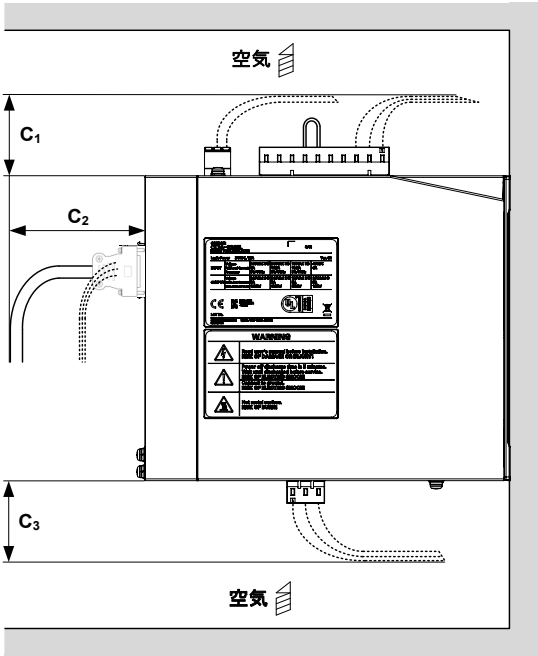


側面

複数ユニット横並びの場合



正面



側面

● 形 CK3A-G305L

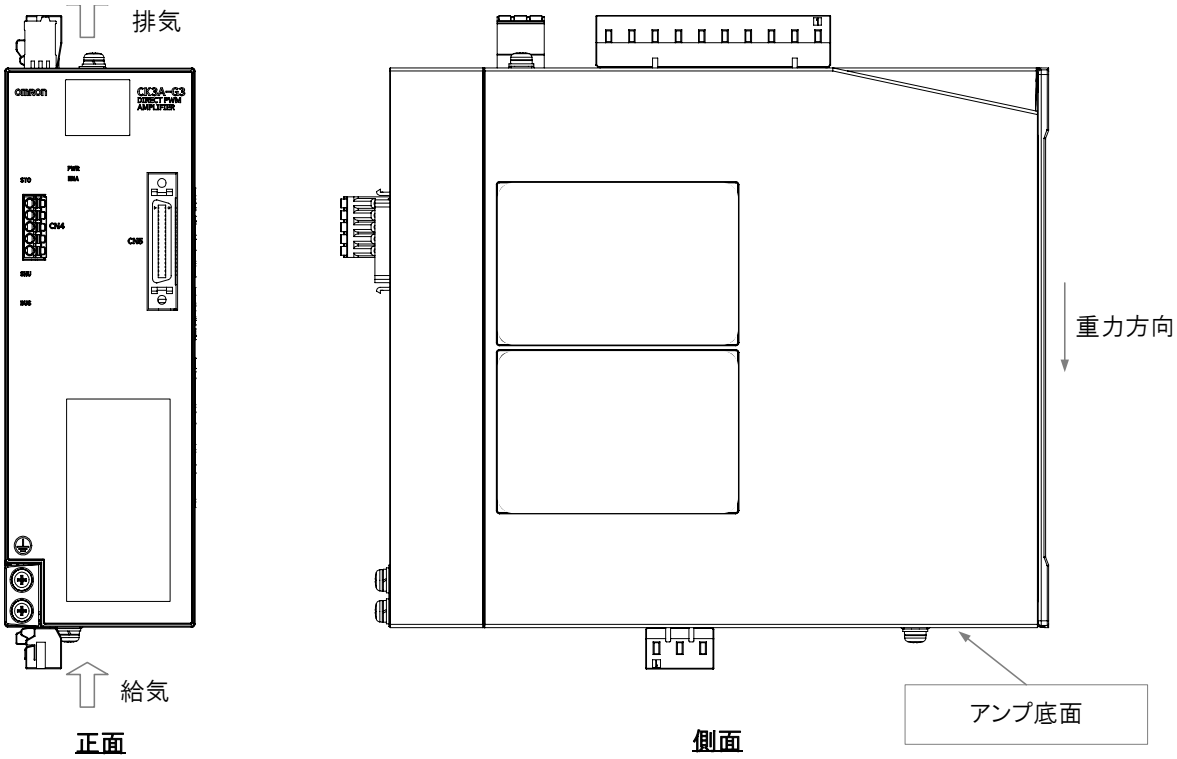
寸法	最小距離［mm］
T（上）	110
B（下）	110
S（横）	10
C_1	70
C_2	100
C_3	70

● 形 CK3A-G310L、形 CK3A-G320L

寸法	最小距離［mm］
T（上）	140
B（下）	140
S（横）	10
C_1	100
C_2	100
C_3	100

3-2-3　取り付け方向

アンプは、底面が床と平行になり、重力方向に垂直になるよう取り付ける必要があります。



3-2-4 設置条件

- 盤の環境（使用温度や湿度などの）条件は、本マニュアルに記載する環境仕様に準拠している必要があります。これらの条件以外でアンプを運転すると、誤動作のおそれあります。

寸法	距離
使用周囲温度	0～55℃
使用周囲湿度	90％以下（結露のないこと）
使用周囲雰囲気	腐食性ガスなし
使用高度	1,000m 以下

- 引き出しなどの密閉された空間にアンプを並べて設置すると、各ユニットからの放熱により周囲温度が上昇する場合があります。ファンまたはクーラを使用して、指定された使用周囲温度を維持してください。
- 高レベルの信頼性を維持するために、温度上昇が最小限の環境で使用することをお勧めします。
- 高温環境でアンプを継続して使用すると、アンプの寿命が短くなります。ファンまたはクーラを使用して、周囲温度を 40℃以下に維持してください。
- 垂直な金属面にアンプを取り付けます。
- 導電性を提供するために、表面の塗料をすべて除去してください。
- カスタム取り付けブラケットを使用する場合は、導電性メッキを適用してください。
- アンプを取り付けるための推奨締め付けねじは 1.2N·m です。
- 取り付けねじのねじ部分に、アンプの重量と推奨トルクに耐えるのに十分な深さがあることを確認してください。
-

3-2-5 異物をユニットに入れない

- 金属粒子、破片、油、機械加工油、ほこり、水などの異物がアンプに入らないように、設置および使用時に対策を講じてください。
-
- アンプにカバーをかぶせるか、他の予防策を講じて、設置中にドリルくずなどの異物がアンプに入らないようにしてください。設置が完了したら、必ずカバーを取り外してください。使用中にカバーを付けたままにすると、アンプからの放熱が妨げられ、誤動作の原因になります。

4

配線

本章では、CK3A DirectPWM アンプの配線方法について説明します。

4-1 主回路電源とシャント（CN1）.....4-3

4-1-1 コネクタピン配置.....4-3
4-1-2 電線寸法.....4-4
4-1-3 配線例.....4-5

4-2 ロジック電源（CN2）.....4-10

4-2-1 コネクタピン配置.....4-10
4-2-2 電線寸法.....4-10
4-2-3 配線例.....4-10

4-3 モータ接続（CN3）.....4-11

4-3-1 コネクタピン配置.....4-11
4-3-2 電線寸法.....4-12
4-3-3 配線例.....4-13

4-4 安全トルク OFF STO（CN4）.....4-14

4-4-1 コネクタピン配置.....4-14
4-4-2 電線寸法.....4-15
4-4-3 配線例.....4-15

4-5 DirectPWM 接続（CN5）.....4-17

4-6 全体配線図の例.....4-18

4-6-1 形 CK3A-G305L.....4-18
4-6-2 形 CK3A-G310L.....4-19
4-6-3 アンプ異常リレー.....4-20

4-7 EMC 指令に準拠した配線.....4-21

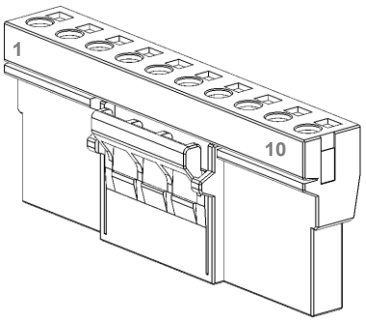
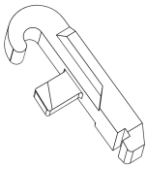
4-7-1 周辺機器の接続例.....4-21

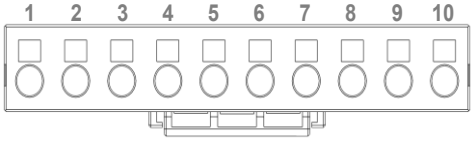
4-8　周辺機器の選択 4-23
4-8-1 主回路 AC 電源 4-23
4-8-2 主回路 DC 電源（低電圧モード） 4-23
4-8-3 ロジック電源 4-23
4-8-4 ノイズフィルタ 4-24
4-8-5 電圧サージアブソーバ 4-24
4-8-6 配線用遮断器（MCCB） 4-24
4-8-7 外部シャント抵抗 4-25
4-8-8 DirectPWM ケーブル 4-25
4-8-9 モータの適合性 4-26

4-1　主回路電源とシャント（CN1）

4-1-1　コネクタピン配置

形 CK3A-G305L

嵌合コネクタ	スプリングオープナー
10 ピンリセプタクル TE コネクティビティ社 品番 1-2134249-0 	TE コネクティビティ社 品番 1981045-1 



ピン	記号	内容と仕様
1	L1	**主回路電源入力です。** • L1-L2-L3 全体で三相 AC240V 50/60Hz • L1-L2 全体で単相 AC110V または AC240V 50/60Hz、L3 はフローティングのまま（オープン）にします • L1-L2 全体で単相 DC48V、L3 はフローティングのまま（オープン）にします
2	L2	
3	L3	
4	NC	接続しません（フローティングのまま）
5	NC	
6	B1	**シャント抵抗端子です。** • 内部シャント抵抗の場合、B2-B3 を短絡し、B1 はフローティングのままにします • 外部シャント抵抗の場合、B1-B2 の間に抵抗を接続し、B3 はフローティングのままにします
7	B2	
8	B3	
9	LV1	**低電圧動作です。** • DC48V 動作時の LV1-LV2 短絡*1 • 単相または三相 AC 動作の場合はフローティングのまま（オープン）にします
10	LV2	

*1.　コントローラで ADC ストローブワードを低電圧モードに設定する必要があります。

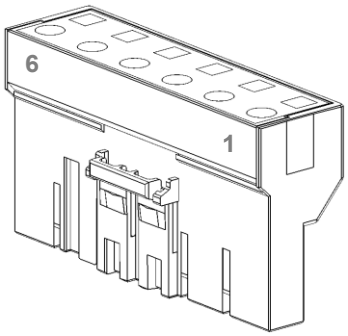
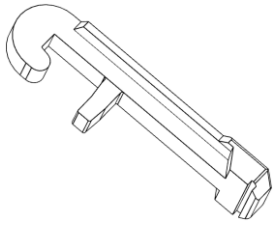
安全上の要点

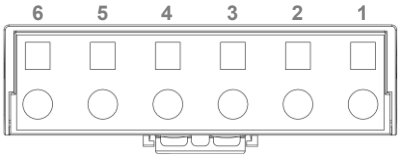
低電圧短絡線を取り付けている場合は、AC 電源を供給しないでください。

参考

コネクタは、内部シャント抵抗用の B2-B3 短絡線を取り付けた状態で出荷されます。

形 CK3A-G310L、形 CK3A-G320L

嵌合コネクタ	スプリングオープナー
6 ピンリセプタクル TE コネクティビティ社 品番 6-2260663-1 	TE コネクティビティ社 品番 2260828-1 



ピン	記号	内容と仕様
1	L1	**主回路電源入力です。** • L1-L2-L3 全体で三相 AC240V 50/60Hz • L1-L2 全体で単相 AC110V または AC240V 50/60Hz、L3 はフローティングのまま（オープン）にします
2	L2	
3	L3	
4	B3	**シャント抵抗端子です。** • 内部シャント抵抗の場合、B2-B3 を短絡し、B1 はフローティングのまま（オープン）にします • 外部シャント抵抗の場合、B1-B2 に接続し、B3 をフローティングのまま（オープン）にします
5	B2	
6	B1	

参考

コネクタは、内部シャント抵抗用の B2-B3 短絡線を取り付けた状態で出荷されます。

4-1-2 電線寸法

使用上の注意

単相または三相 AC 主回路電源には、定格 AC600V 以上の電線ケーブルを使用してください。

主回路電源、内部抵抗短絡線、低電圧モード短絡線に推奨する電線ゲージは下表のとおりです。

項目	形 CK3A-G305L	形 CK3A-G310L	形 CK3A-G320L
主回路電源と内部抵抗の短絡線	AWG18～14 0.75～2.5mm^2	AWG12～8 4～10mm^2	AWG10～8 6～10mm^2
保護アース ⏚	AWG12、2.5mm^2	AWG8、10mm^2	
ねじ寸法	M4		
締め付けトルク	1.2N·m		

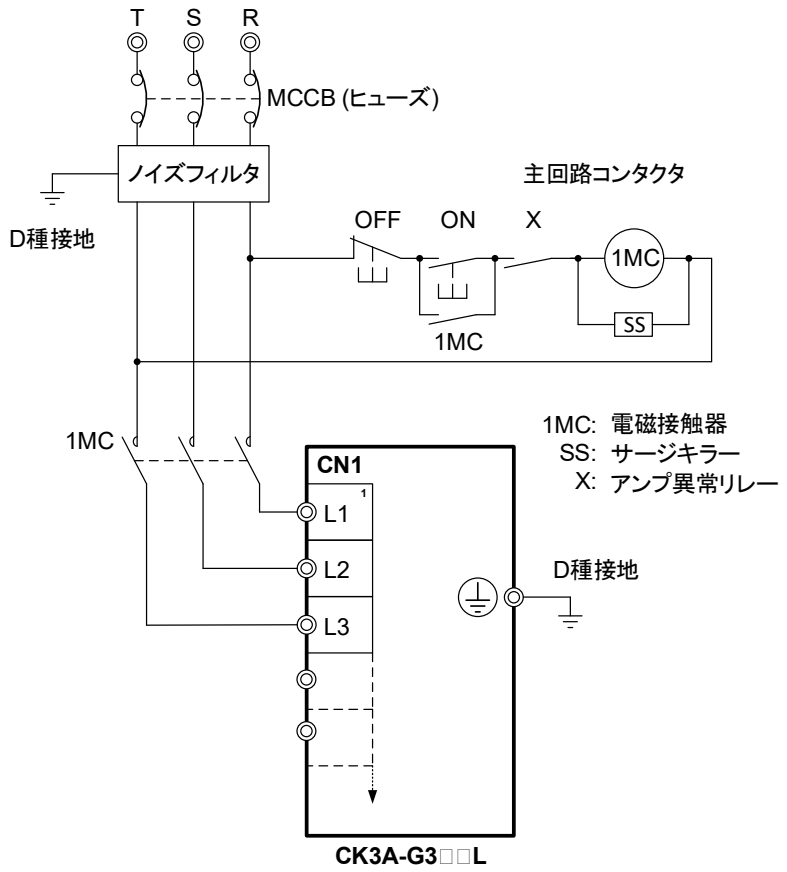
（注） 外部シャント抵抗には、抵抗に付属の内蔵線を使用してください。

4-1-3 配線例

三相 AC

アンプは、三相 AC 主回路電源入力で動作するように設計されています。下記の図は、この配線方式を示しています。

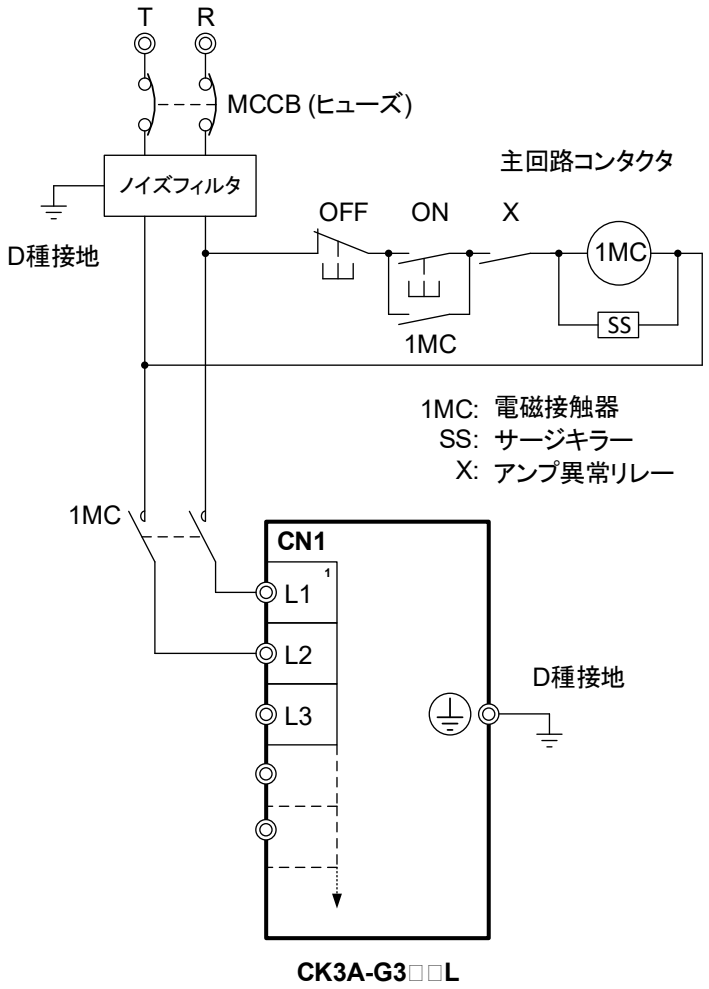
形 CK3A-G3□□L の三相主回路入力



単相 AC

アンプは、単相 AC 主回路電源入力で動作することができます。下記の図は、この配線方式を示しています。

形 CK3A-G3□□L の単相主回路入力



DC 入力（低電圧モード）CK3A-G305L

形 CK3A-G305L は、低電圧（DC48V）の主回路電源入力で動作することができます。下記の図は、この配線方式を示しています。

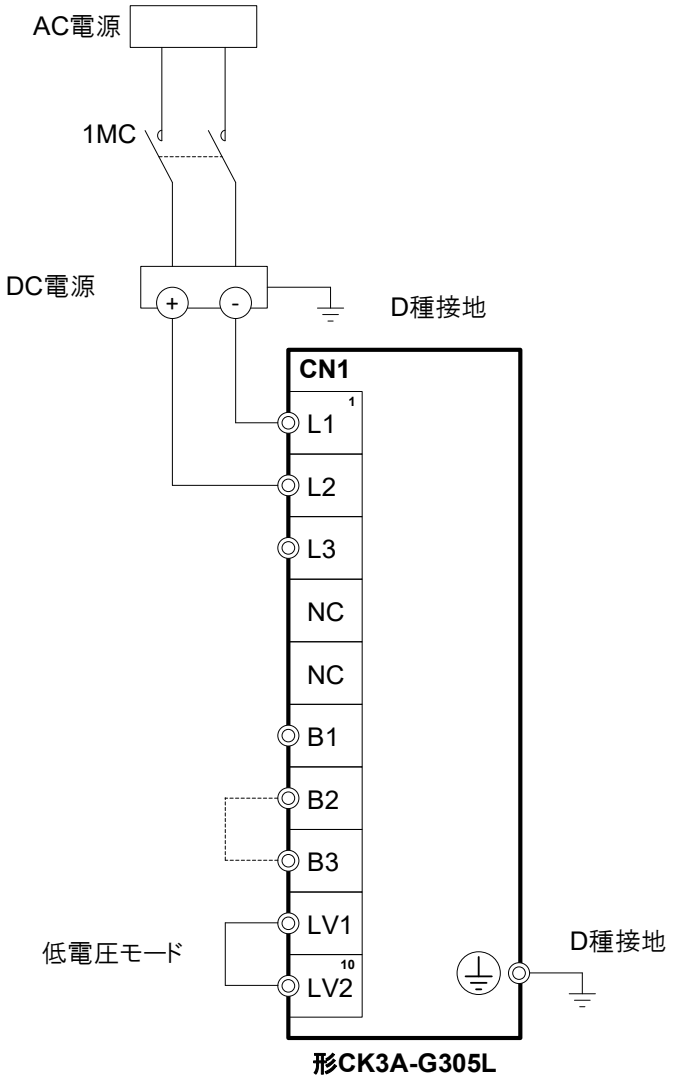
安全上の要点

低電圧短絡線を取り付けている場合は、AC 電源を供給しないでください。

使用上の注意

低電圧動作を成功させるためには、コントローラの ADC ストローブワード設定を適宜設定する必要があります。

形 CK3A-G305L の DC 主回路入力



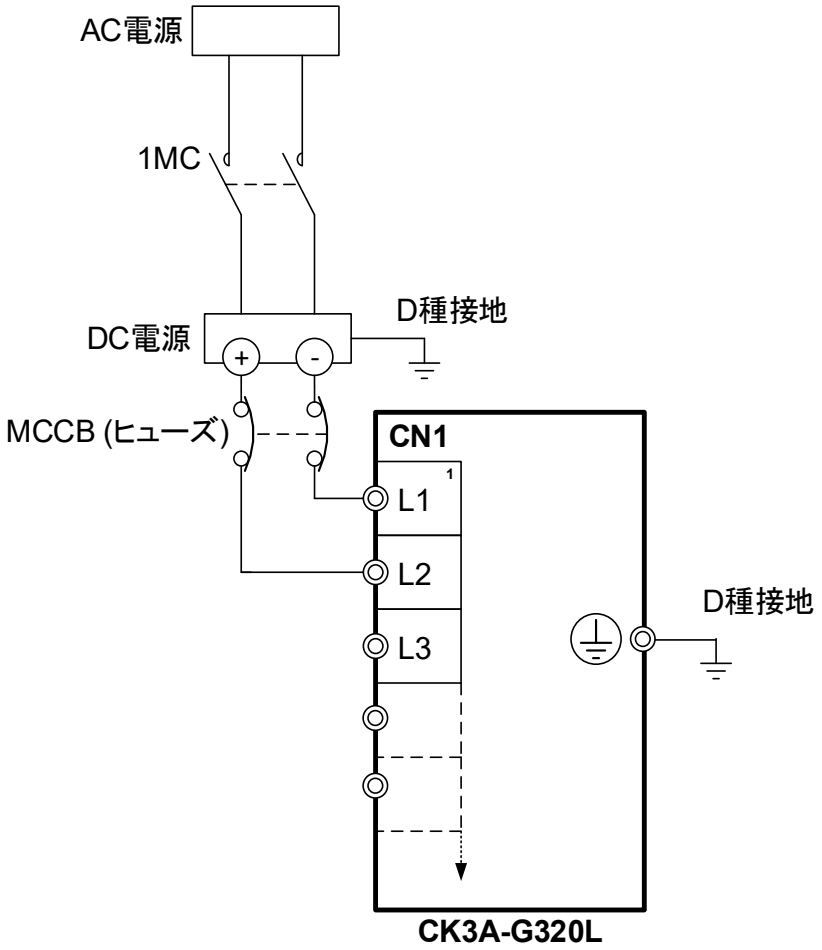
DC 入力（低電圧モード）CK3A-G320L

形 CK3A-G320L は、低電圧（DC100V）の主回路電源入力で動作することができます。下記の図は、この配線方式を示しています。

使用上の注意

低電圧動作を成功させるためには、コントローラの ADC ストローブワード設定を適宜設定する必要があります。

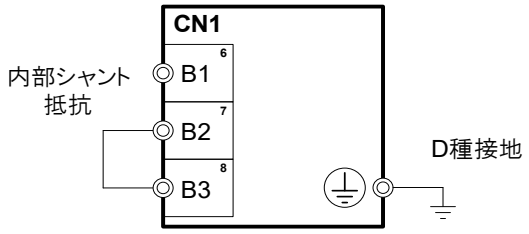
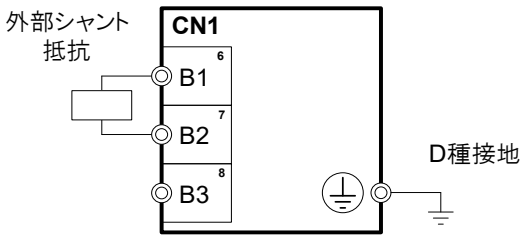
形 CK3A-G320L の DC 主回路入力



シャント抵抗の配線

下図は、内部および外部シャント抵抗の配線構成を示しています。

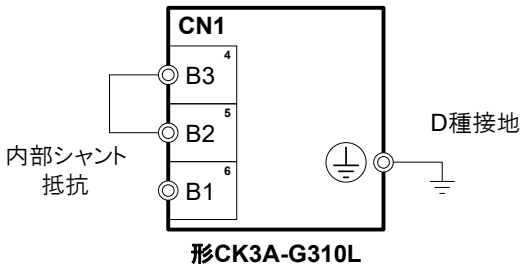
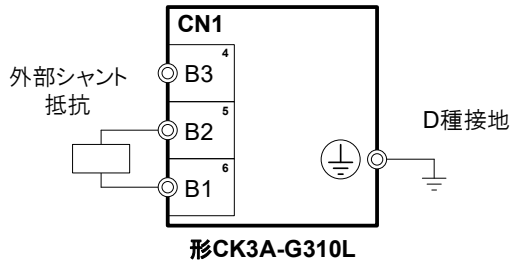
● 形 CK3A-G305L

内部シャント抵抗	外部シャント抵抗
	

使用上の注意

内部シャント抵抗を使用するための B2 と B3 間のワイヤは、付属の CN1 コネクタにあらかじめ取り付けられています。外部シャント抵抗を使用する場合は、このワイヤを取り外してください。

● 形 CK3A-G310L、形 CK3A-G320L

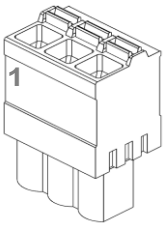
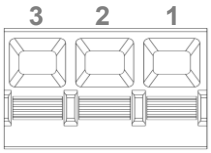
内部シャント抵抗	外部シャント抵抗
	

使用上の注意

内部シャント抵抗を使用するための B2 と B3 間のワイヤは、付属の CN1 コネクタにあらかじめ取り付けられています。外部シャント抵抗を使用する場合は、このワイヤを取り外してください。

4-2　ロジック電源（CN2）

4-2-1　コネクタピン配置

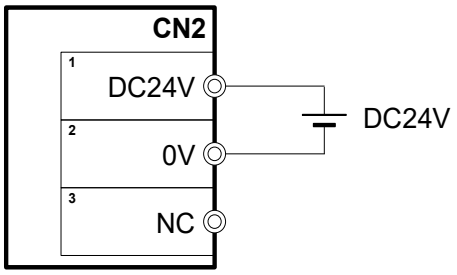
嵌合コネクタ		
3ピン端子台 フエニックス･コンタクト社品番 1754571		

ピン	記号	内容と仕様
1	DC24V	ロジック電源入力
2	0V	
3	NC	接続しません（フローティングのまま）

4-2-2　電線寸法

ロジック電源入力に推奨する電線ゲージは、AWG20～16、0.5～1.5mm^2 です。

4-2-3　配線例

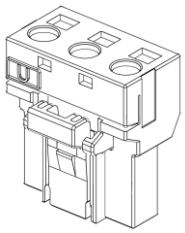
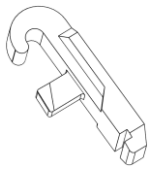


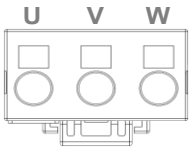
形CK3A-G3□□L

4-3　モータ接続（CN3）

4-3-1　コネクタピン配置

形 CK3A-G305L

嵌合コネクタ		スプリングオープナー
3 ピンリセプタクル TE コネクティビティ社 品番 3-2229794-1		TE コネクティビティ社品番 1981045-1 



ピン	記号	内容と仕様
1	U	**モータ相出力端子です。** ● ブラシレスモータの場合、U、V、W を接続します ● ブラシモータの場合、U と W を接続し、V をフローティングのまま（オープン）にします
2	V	
3	W	

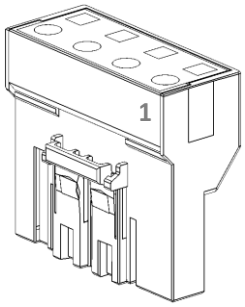
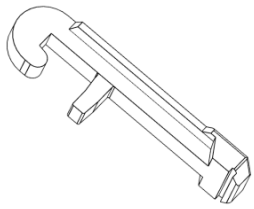
使用上の注意

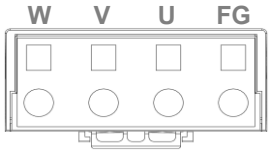
モータ保安用接地をアンプの保護アース⏚に接続します。

安全上の要点

モータ接続コネクタが接続されていないときにコントローラがモータコマンドを送信しても、アンプはエラーを出しません。コントローラがモータコマンドを送信していない、または主電源を接続していないことを確認するまではモータ接続コネクタを接続しないでください。

形 CK3A-G310L、形 CK3A-G320L

嵌合コネクタ	スプリングオープナー
4 ピンリセプタクル TE コネクティビティ社品番 4-2260663-1 	TE コネクティビティ社品番 2260828-1 



ピン	記号	内容と仕様
1	FG	**モータ相出力端子です。** • FG は保安用接地で、モータ保安用接地線をこの端子に接続します • 三相ブラシレスモータの場合、U、V、W を接続します • 二相ブラシモータの場合、U と W を接続し、V をフローティングのまま(オープン)にします
2	U	
3	V	
4	W	

4-3-2 電線寸法

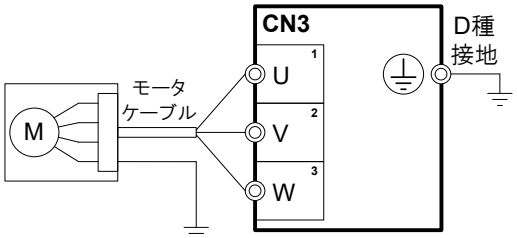
モータ相端子と保安用接地に推奨する電線ゲージは下表のとおりです。

項目	形 CK3A-G305L	形 CK3A-G310L	形 CK3A-G320L
主回路電源と短絡線	AWG18～14 0.75～2.5mm^2	AWG14～8 2.5～10mm^2	AWG12～8 4～10mm^2
保護アース ⏚	AWG12、4mm^2		
ねじ寸法	M4		
締め付けトルク	1.2N·m		

4-3-3 配線例

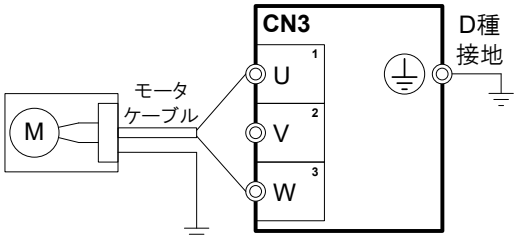
形 CK3A-G305L

● ブラシレスモータ



形CK3A-G305L

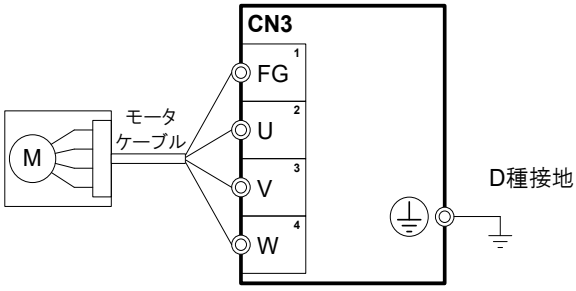
● ブラシモータ



形CK3A-G305L

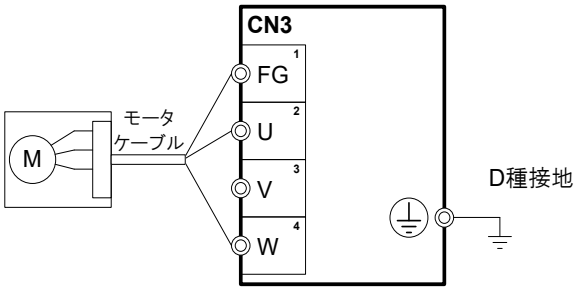
形 CK3A-G310L、形 CK3A-G320L

● ブラシレスモータ



形CK3A-G310L

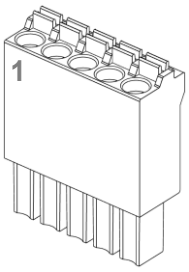
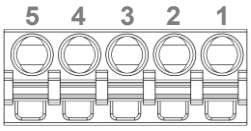
● ブラシモータ



形CK3A-G310L

4-4　安全トルク OFF STO（CN4）

4-4-1　コネクタピン配置

嵌合コネクタ		
5ピン端子台 フエニックス･コンタクト社品番 1745920		

ピン	記号	内容と仕様
1	STO OUT	**安全トルク OFF STO 制御端子です。** • STO 機能を無効にするには：STO OUT と STO DIS を短絡し、他のすべての端子をフローティングのまま（オープン）にします。このワイヤは工場出荷時に接続されています。 • STO 入力 1 だけを使用するには：STO IN1 と STO OUT 間のリレー入力を配線し、STO IN2 を STO UT に接続し、STO DIS をフローティングのまま（オープン）にします。 • STO 入力 2 だけを使用するには：STO IN2 と STO OUT 間のリレー入力を配線し、STO IN1 を STO OUT に接続し、STO DIS をフローティングのまま（オープン）にします。 • STO 入力 1 と 2 の両方を使用するには：STO IN1 と STO OUT 間のリレー入力 1、STO IN2 と STO OUT の間のリレー入力 2 を配線し、STO DIS をフローティングのまま（オープン）にします。
2	STO DIS	
3	STO IN1	
4	STO IN2	
5	STO FB	安全トルク OFF STO 状態端子です。 この端子は、STO 状態を反映する PNP タイプの出力です。

使用上の注意

STO を無効にするためのピン 1 と 2 間のワイヤは、付属の CN4 コネクタにあらかじめ取り付けられています。STO 機能を使用するには、このワイヤを取り外してください。

使用上の注意

UL SIL 3 の安全要件を満たすには、システムで両方の STO 入力を使用する必要があります。
STO を 1 つだけ使用するか、STO を完全に無効にするシステムは、UL SIL 3 の安全要件を満たしません。

使用上の注意

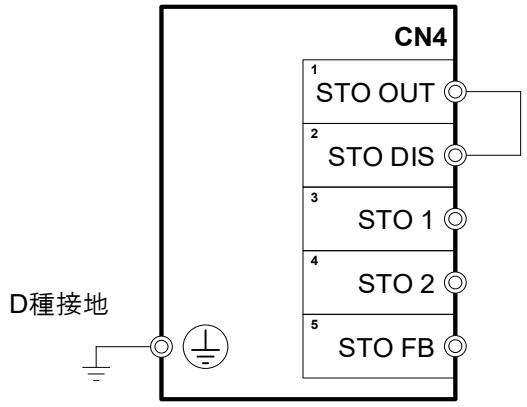
特定の安全コントローラまたは安全センサーは、自己診断のために L パルスを出力する場合があります。これらのパルスは、短い (通常 1 ミリ秒以下) 低パルスです。これらの信号を真の STO 信号と区別するために、アンプの入力回路は、非常に短い持続時間の信号をフィルタリングするように設計されています。OFF が認識されるようにするには、信号を少なくとも 5 ミリ秒間維持する必要があります。

4-4-2 電線寸法

STO 機能に推奨する電線ゲージは AWG20～16、0.5～1.5mm^2 です。

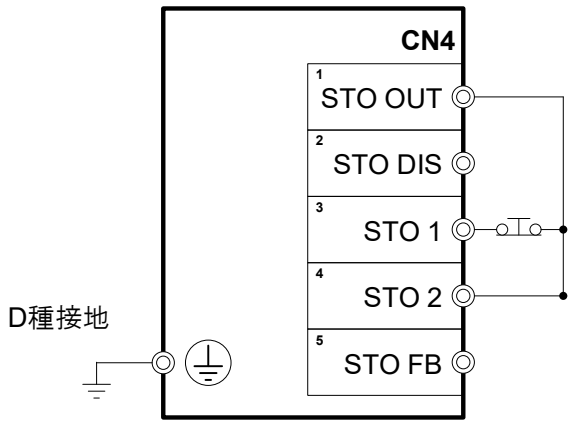
4-4-3 配線例

STO を無効化する場合



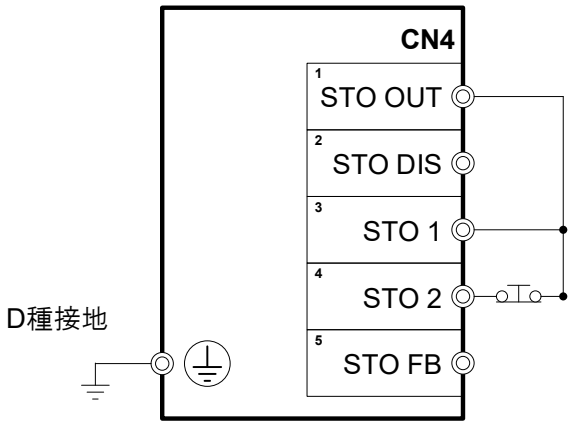
形CK3A-G3□□L

STO1 だけを使用する場合



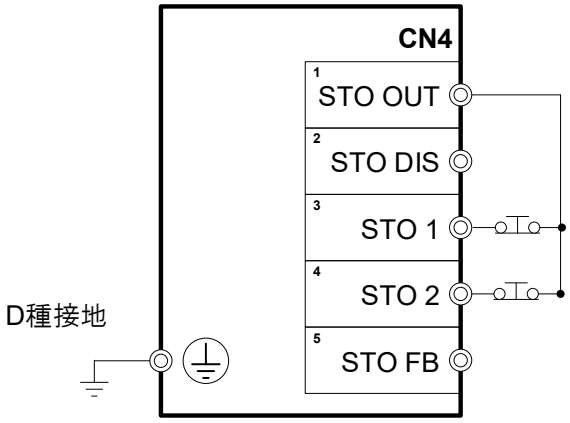
形CK3A-G3□□L

STO2 だけを使用する場合



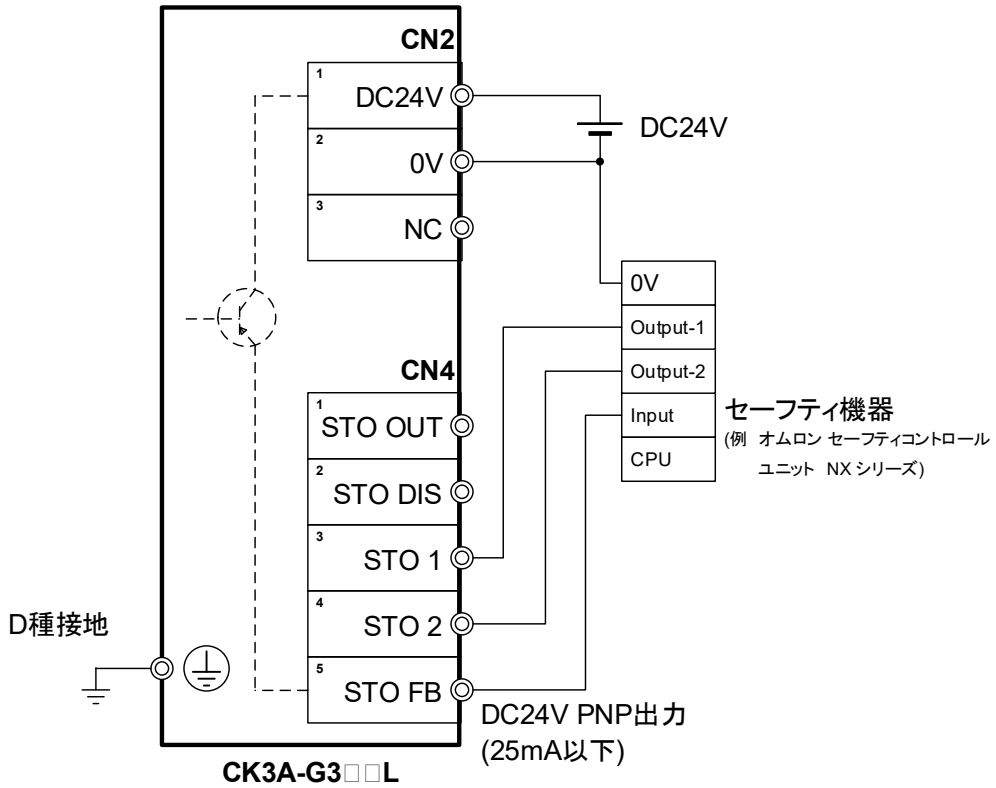
形CK3A-G3□□L

STO1 と STO2 の両方を使用する場合



形CK3A-G3□□L

STO フィードバックの場合



CK3A-G3□□L

4-5 DirectPWM 接続（CN5）

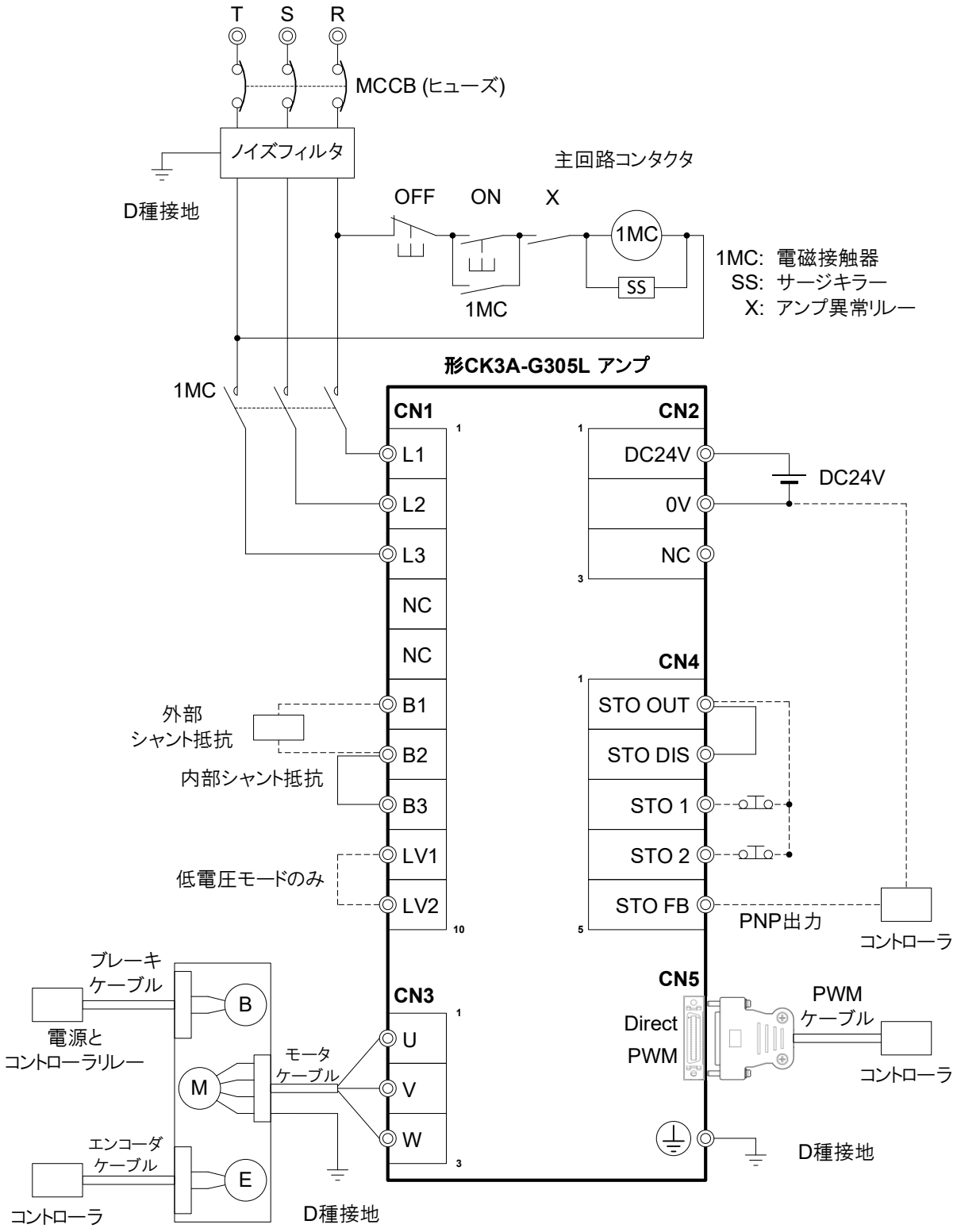
コネクタピン配置

ピン	記号	内容	機能*1
1	PHACLK1+	フェーズクロック	入力
2	N.C.	未接続	-
3	ADCCLK1+	A/D コンバータクロック	入力
4	ADCSTB1+	A/D コンバータストローブ	入力
5	ADCDAT1A+	A 相電流データ	出力
6	ADCDAT1B+	B 相電流データ	出力
7	AENA1+	アンプ有効	入力
8	FAULT1+	アンプ異常	出力
9	PWMATOP1+	A 相先頭コマンド	入力
10	PWMABOT1+	A 相最後尾コマンド	入力
11	PWMBTOP1+	B 相先頭コマンド	入力
12	PWMBBOT1+	B 相最後尾コマンド	入力
13	PWMCTOP1+	C 相先頭コマンド	入力
14	PWMCBOT1+	C 相最後尾コマンド	入力
15	0V	基準電圧	Common
16	+5V	デジタル電力	入力
17	N.C.	未接続	
18	N.C.	未接続	
19	PHACLK1-	フェーズクロック	入力
20	N.C.	未接続	-
21	ADCCLK1-	A/D コンバータクロック	入力
22	ADCSTB1-	A/D コンバータストローブ	入力
23	ADCDAT1A-	A 相電流データ	出力
24	ADCDAT1B-	B 相電流データ	出力
25	AENA1-	アンプ有効	入力
26	FAULT1-	アンプ異常	出力
27	PWMATOP1-	A 相先頭コマンド	入力
28	PWMABOT1-	A 相最後尾コマンド	入力
29	PWMBTOP1-	B 相先頭コマンド	入力
30	PWMBBOT1-	B 相最後尾コマンド	入力
31	PWMCTOP1-	C 相先頭コマンド	入力
32	PWMCBOT1-	C 相最後尾コマンド	入力
33	0V	基準電圧	Common
34	+5V	デジタル電力	入力
35	N.C.	未接続	
36	N.C.	未接続	

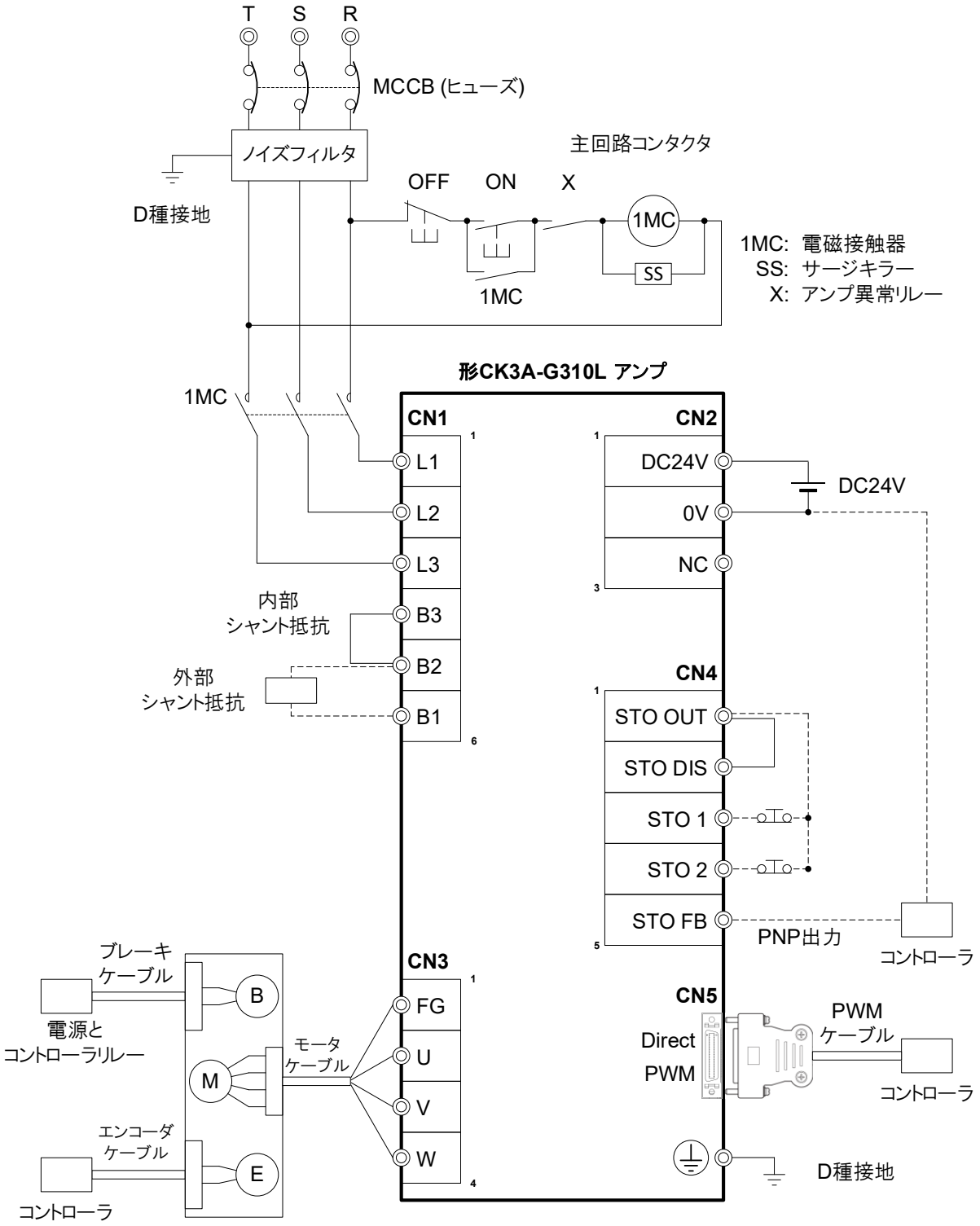
*1. アンプ側から見た機能。

4-6 全体配線図の例

4-6-1 形 CK3A-G305L



4-6-2 形 CK3A-G310L、形 CK3A-G320L



4-6-3 アンプ異常リレー

アンプに異常が発生した時に主電源を切断したい場合は、コントローラからリレーコイルを制御する出力信号を出す必要があります。これは、対応するモータアンプの異常ビットをミラーリングする PLC で実行することができます。

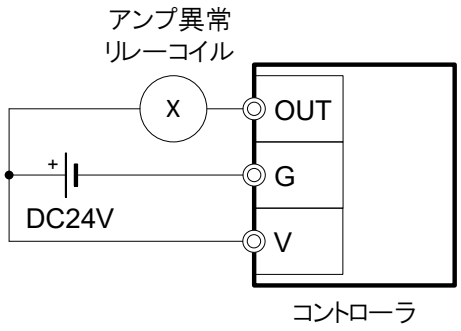
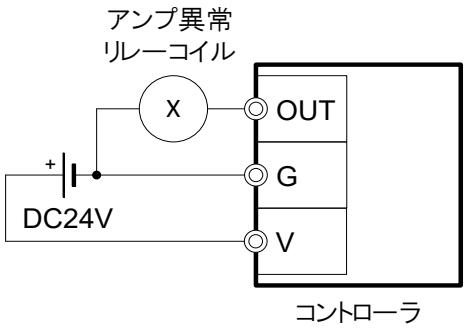
サンプル PLC

```
PTR DigitalOutput1->Gate3[0].GpioData[0].16

OPEN PLC SamplePLC
DigitalOutput1 = Motor[1].AmpFault
CLOSE
```

参考

上記 PLC のコード行は、既存の使用中の PLC にマージすることができます。独自の PLC である必要はありません。

配線例

NPN タイプ出力	PNP タイプ出力
	

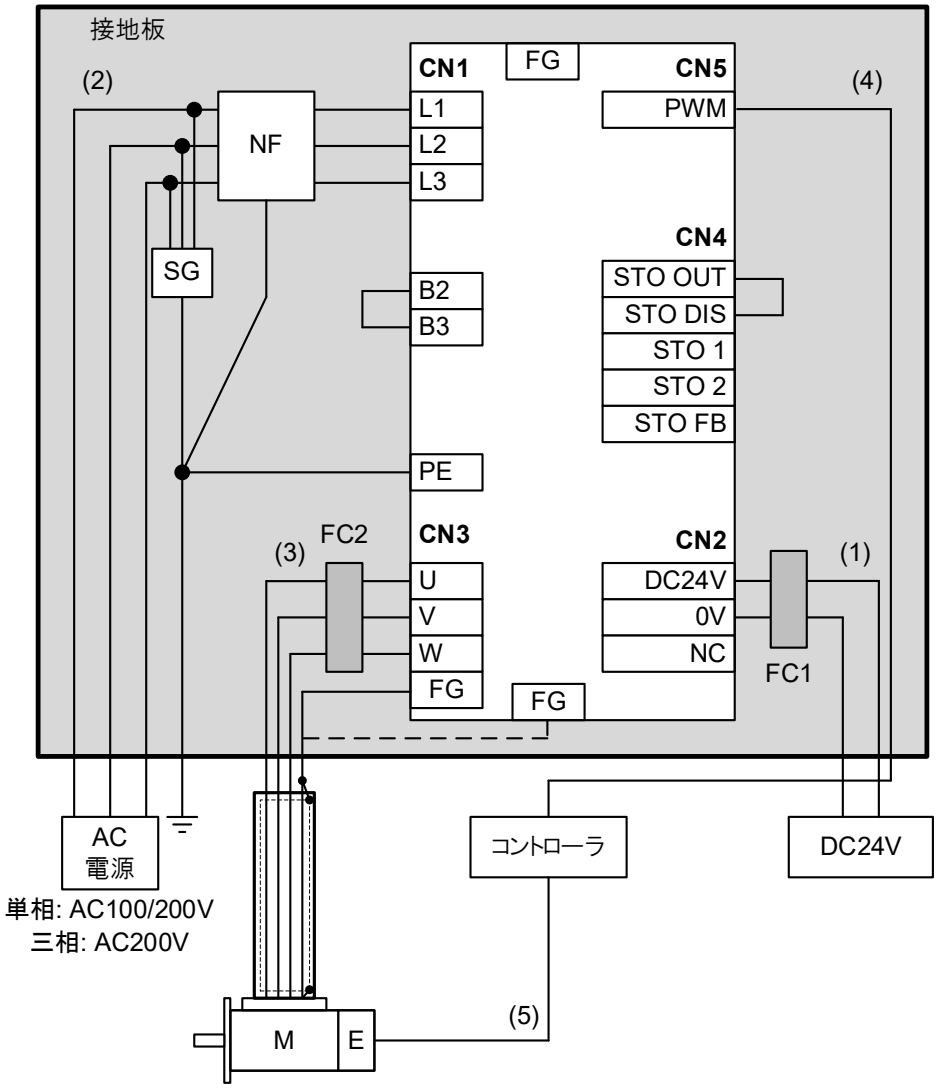
4-7 EMC 指令に準拠した配線

CK3A DirectPWM アンプは、ここで説明する配線条件の下で EMC 指令（EN 61800-3）に準拠します。
以下の条件は、CK3A シリーズが EMC 指令に準拠するために決定しています。
製品を装置に設置する場合、お客様は、マシン全体が EMC 指令に準拠していることを確認するためのチェックを行ってください。

EMC 指令への適合に必要な条件を以下に示します。

- アンプを接地板に取り付けてください。
- 電力線にノイズフィルタと雷サージ吸収素子（サージアブソーバ）を取り付けてください。
- I/O 信号とエンコーダには編組シールドケーブルを使用してください。
- シールドには、錫メッキされた軟鋼線を使用してください。
- 各ケーブルのシールドを接地してください。

4-7-1 周辺機器の接続例



（注）　単相入力の場合、L1 と L2 を接続します。
（注）　形 CK3A-G305L の場合、モータ接続コネクタに FG が存在しないため、代わりに FG ねじ留め式端子を使用してください。

- 上図に示すように、ユニット保安用接地のために接地板を一点接地してください。
- 最小太さ 2.5mm^2 の保護接地線を使用し、保護接地線ができるだけ短くなるように配線してください。
- アンプの主回路コネクタ CN1 の近くにサージアブソーバとノイズフィルタを取り付けてください。

部材詳細

記号	名称	メーカ	形式	アンプ形式	備考
SG	サージアブソーバ	双信電機株式会社	LT-C12G801WS	形 CK3A-G3□□L	単相 AC100/200V
			LT-C32G801WS		三相 AC200V
NF	ノイズフィルタ	オムロン株式会社	形 R88A-FI1S105	形 CK3A-G305L	単相 AC100/200V
			形 R88A-FI1S208		三相 AC200V
		シャフナー	FN シリーズ	形 CK3A-G3□0L	単相 AC100V
		オムロン株式会社	形 R88A-FI1S116		単相 AC200V
			形 R88A-FI1S216		三相 AC200V
FC1	フェライト磁心	NECトーキン株式会社	ESD-SR-250	形 CK3A-G3□□L	-
FC2	フェライト磁心	星和電機株式会社	E04SR301334		-
M	モータ	-	-	-	-
E	エンコーダ	-	-	-	-
-	コントローラ	オムロン株式会社	形 CK3M シリーズ	形 CK3A-G3□□L	DirectPWM 対応
			形 UMAC シリーズ		DirectPWM 対応

（注）　詳細な仕様については、各メーカまたは当社営業にご相談ください。

ケーブル情報

No.	インタフェース	フェライト磁心	最大長	シールド有無
(1)	ロジック電源ケーブル	2 巻き	3m	なし
(2)	主回路電源ケーブル	-	3m	なし
(3)	モータケーブル	2 巻き	20m	あり
(4)	DirectPWM ケーブル	-	3.6m	-
(5)	エンコーダケーブル	-	-	あり

4-8 周辺機器の選択

4-8-1 主回路 AC 電源

主回路 AC 電源は、本マニュアルの「第 2 章 仕様」に記載する仕様に従って選択する必要があります。

4-8-2 主回路 DC 電源（低電圧モード）

主回路 DC 電源は、本マニュアルの「第 2 章 仕様」に記載する仕様に従って選択する必要があります。推奨（DC48V）電源は下表のとおりです。

電源	メーカ
形 S8VK-G シリーズ	オムロン株式会社
形 S8FS-G シリーズ	オムロン株式会社
形 S8FS-C シリーズ	オムロン株式会社

使用上の注意

DC 電源定格の選択は、アプリケーションの要件によって異なります。

- 同じ電源を使用しているアンプの数（総実効負荷）
- 必要な過渡電流または加速電流（定格 DC 電圧）
- 定常状態、アイドルまたは定速で必要な電流（定格 DC 電圧）

4-8-3 ロジック電源

ロジック電源は、本マニュアルの「第 2 章 仕様」に記載する仕様に従って選択する必要があります。推奨（DC24V）電源は下表のとおりです。

電源	メーカ
形 S8VK-G シリーズ	オムロン株式会社
形 S8VK-S シリーズ	オムロン株式会社

4-8-4 ノイズフィルタ

AC 電圧環境では、ラインノイズフィルターが（システムからおよびシステムへの）双方向で電磁ノイズを除去します。T タイプフィルタはお勧めしません。PI タイプフィルタをお勧めします。

- ノイズフィルタは、ドライバおよび電源と同じ盤に取り付けてください。
- ノイズフィルタは、電源のできるだけ近くに取り付けてください。
- ノイズフィルタは、キャビネットの入力電源のできるだけ近くに取り付けてください。
- ノイズフィルタは、本マニュアルの「第 2 章 仕様」に記載する主電源入力仕様に従って選択する必要があります。

推奨ノイズフィルタは下表のとおりです。

アンプ形式	ノイズフィルタ	電圧入力	メーカ
形 CK3A-G305L	形 R88A-FI1S105	単相 AC100V	オムロン株式会社
	形 R88A-FI1S105	単相 AC200V	オムロン株式会社
	形 R88A-FI1S208	三相 AC200V	オムロン株式会社
形 CK3A-G310L	FN シリーズ	単相 AC100V	シャフナー
	形 R88A-FI1S118	単相 AC200V	オムロン株式会社
	形 R88A-FI1S218	三相 AC200V	オムロン株式会社
形 CK3A-G320L	形 R88A-FI1S330-SE	単相 AC400V	オムロン株式会社
	形 R88A-FI1S330-SE	三相 AC400V	オムロン株式会社

（注） 詳細な仕様については、お近くの当社営業にお問い合わせください。

4-8-5 電圧サージアブソーバ

サージアブソーバは、電源入力ラインの雷サージ電圧および異常電圧から保護します。推奨サージアブソーバは下表のとおりです。

電圧入力	サージアブソーバ	メーカ
単相 AC100V	LT-C12G801WS	双信電機株式会社
単相 AC200V	LT-C12G801WS	双信電機株式会社
三相 AC200V	LT-C32G801WS	双信電機株式会社

（注） 詳細な仕様については、メーカにお問い合わせください。

4-8-6 配線用遮断器（MCCB）

最大入力電流に基づいて、配線用遮断器を選択してください。

- アンプの瞬間最大出力は定格出力の約 2 倍、最大出力時間は 2 秒です。
- したがって、定格電流の 200%で 10 秒以上作動できる配線用遮断器を選択してください。
- 複数のアンプを使用する場合、定格電流が実効負荷電流の合計よりも大きい配線用遮断器を選択してください。
- 配線用遮断器を選択するときは、忘れずに他の機器の消費電流も追加してください。

推奨ヒューズ

UL 規格認証製品で LISTED と㊀マーク表示があるヒューズを使用してください。温度定格 75℃以上の銅線を使用してください。推奨ヒューズは下表のとおりです。

アンプ形式	主回路供給	推奨ヒューズ
形 CK3A-G305L	三相 AC240V	UL クラス RK5 10A
	単相 AC110V または AC240V	UL クラス RK5 15A
	単相 DC48V	UL クラス RK5 10A
形 CK3A-G310L	三相 AC240V	UL クラス RK5 15A
	単相 AC110V または AC240V	UL クラス RK5 25A
形 CK3A-G320L	三相 AC240V	UL クラス RK5 20A
	単相 AC240V	UL クラス RK5 30A
	単相 DC100V	UL クラス RK5 20A

4-8-7 外部シャント抵抗

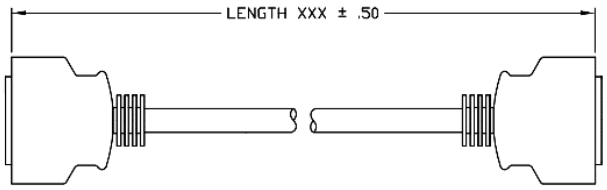
外部シャント抵抗は、本マニュアルの「第 2 章 仕様」に記載する仕様に従って選択する必要があります。推奨外部シャント抵抗は下表のとおりです。

アンプ形式	外部シャント抵抗	メーカ	仕様
形 CK3A-G305L	形 R88A-RR30020	オムロン株式会社	60W、20Ω
形 CK3A-G310L	形 R88A-RR30017	オムロン株式会社	60W、17Ω
形 CK3A-G320L			

（注） 詳細な仕様については、お近くの当社営業にお問い合わせください。

4-8-8 DirectPWM ケーブル

形式	名前	長さ(L)
形 200-602739-036x	CABPWM-2	0.9m（36 インチ）
形 200-602739-072x	CABPWM-4	1.8m（72 インチ）
形 200-602739-144x	CABPWM-6	3.6m（144 インチ）



4-8-9 モータの適合性

CK3A シリーズの製品ラインは、さまざまなモータとの接続が可能です。アンプは、AC/DC ブラシレス（同期）サーボロータリまたはリニアモータ、および AC 誘導（非同期）モータを含む、ほぼすべての種類の三相モータを制御することができます。さらに、アンプはボイスコイルアクチュエータなどの永久磁石 DC ブラシモータを制御できます（アンプの三相のうち 2 つを使用）。
モータ選定については、本マニュアルの対象範囲に含みません。モータ選定は、さまざまなアプリケーションのニーズに依存し、通常、適切な選定ツールを使用して実行します。

モータの仕様と候補を選択したうえで、モータの仕様シートとアンプの仕様シートを簡単に比較すると、互換性がわかります。基本的に、下表に示すように、考慮すべき主要な仕様パラメータが 4 つあります。

項目	形 CK3A-G305L	形 CK3A-G310L	形 CK3A-G320L
定格電圧（主回路入力）	AC110V または AC240V		AC240V
定格電流	最大 $5A_{rms}$	最大 $10A_{rms}$	最大 $20A_{rms}$
ピーク電流	最大 10 A_{rms}, 2 秒	最大 20 A_{rms}, 2 秒	最大 60 A_{rms}, 2 秒
モータインダクタンス	最小 1mH		

定格電圧

電圧は速度に正比例します。

● モータ定格電圧が供給電圧以下の場合

モータ定格電圧が主回路供給電圧以下の場合、モータ定格速度を達成することが可能です（ここでは問題ありません）。

使用上の注意

- この場合、コントローラの Motor[x].PwmSf 設定は次のとおりにしてください。
  16,384 * （モータ電圧/供給電圧）
- 損失を考慮して、必要に応じて 10%増やすことができます。

● モータ定格電圧が供給電圧よりも大きい場合

モータの定格電圧が（最大）主回路供給電圧よりも大きい場合、モータは機能しますが、定格速度に達しない場合があります。この場合、アプリケーションにモータ定格速度が必要かどうかを検討してください。定格速度が必要ない場合、モータは使用可能です。定格速度が必要な場合、モータは使用できません。

参考

- 基本的に、十分な供給電圧と適切な電流ループ調整により、モータ定格/最高速度を達成できるはずです。ただし、特定のアプリケーションでは、フィードバック機器や過度な負荷などによる制限により、これが不可能な場合があります。
- 一般に、最大モータ速度は、ライン間入力電圧をモータ逆起電力定数で割ることによって決定することができます。適切なサーボアプリケーションのためには、これを少しディレーティングするのが賢明です。

定格電流とピーク電流

電流はトルクに正比例します。

- 定格電流は通常、定常状態の位置保持または定速運動と相関関係があります。
- ピーク電流は通常、過渡（加速）運動と相関関係があります。

● モータ定格電流がアンプ定格電流以下の場合

モータの定格電流がアンプの定格電流以下の場合、モータは使用可能で（ここでは問題ありません）、最適な定常状態の動作性能を実現できます。

安全上の要点

この場合、モータの焼損を防ぐために、コントローラの I2T 保護を正しく設定する必要があります。

● モータ定格電流がアンプ定格電流よりも大きい場合

モータの定格電流がアンプの定格電流よりも大きい場合、モータは機能しますが、定常状態または一定の動作で最適なトルクを達成できない場合があります。

参考

モーションアプリケーションでは極めて一般的に定格に余裕のあるモータを使用します。多くの場合、位置保持または定速では、定格電流のごく一部しか使用されません。

● モータのピーク電流がアンプのピーク電流以下の場合

モータのピーク電流がアンプのピーク電流以下の場合、モータは使用可能で（ここでは問題ありません）、最適な過渡（加速）動作性能を実現できます。

安全上の要点

この場合、モータの焼損を防ぐために、コントローラの I2T 保護を正しく設定する必要があります。

● モータのピーク電流がアンプのピーク電流よりも大きい場合

モータのピーク電流がアンプのピーク電流よりも大きい場合、モータは機能しますが、過渡（加速）動作で最適なトルクを達成できない場合があります。
この場合、アプリケーションに最高加速度が必要かどうかを検討してください。最高加速度が必要ない場合は、モータは使用可能です。最高加速度が必要な場合、モータは使用できません。

参考

ピーク時の時間については、モーションアプリケーションの最大加速度（ピーク電流引き込み）プロファイルは通常 500 ミリ秒未満であることを忘れないでください。

モータインダクタンス

PWM 出力は、オンオフ電圧信号をリップルが少ない比較的滑らかな電流に変えるために、かなりのモータインダクタンスを必要とします。通常、サーボモータのモータインダクタンスは 1～15mH です。
CK3A シリーズはこの範囲を簡単に駆動できます。

低インダクタンスモータ（1mH 未満）の場合、大きなリップル電流がモータを流れる PWM スイッチングが原因で問題が発生し、過剰なエネルギー浪費と加熱が発生する可能性があります。アプリケーションで 1mH 未満のモータが必要な場合は、外部インダクタを追加してインダクタンスを増やすことができます。

インダクタンスが 15mH を超えるモータは駆動できますが、反応が遅く、通常は高性能サーボモータの範囲外となります。

5

ソフトウェア構成

本章では、CK3A DirectPWM アンプのソフトウェア構成について説明します。

5-1 ソフトウェア構成の概略 5-3

5-1-1 コントローラ-アンプインタフェース 5-3
5-1-2 要約 5-3
5-1-3 対象となるユーザの方々 5-4
5-1-4 対象となるハードウェア 5-4
5-1-5 Gate3 エイリアス名 5-4

5-2 基本構成 5-5

5-2-1 PWM 周波数 5-5
5-2-2 Gate3 設定 5-5
5-2-3 チャネル設定 5-5
5-2-4 モータ設定 5-6

5-3 ADC ストローブワードの詳細 5-7

5-3-1 概要 5-7
5-3-2 明示的アドレス 5-8

5-4 アンプ機能の詳細 5-9

5-4-1 特長 5-9
5-4-2 バス放電制御 5-9
5-4-3 異常解除 5-11
5-4-4 ファン制御 5-11
5-4-5 ダイナミックブレーキ制御 5-12
5-4-6 低電圧（LV）モード 5-13

5-5 ADC レジスタの詳細 5-14

5-5-1 ADC A の情報 5-14
5-5-2 ADC B の情報 5-14

5-5-3 ADC レジスタ明示的アドレス 5-15

5-6 ステータスビットの詳細 5-16

5-6-1 ステータスビットセット 1 5-16
5-6-2 ステータスビットセット 2 5-17
5-6-3 ステータスビットセット 3 5-18

5-7 アンプデータの詳細 5-19

5-8 CK3A サンプル PLC 5-21

5-8-1 サンプル PLC の実装 5-21
5-8-2 遅延タイマーサブルーチン 5-22
5-8-3 ユーザ共有メモリの説明 5-23
5-8-4 CK3A サンプル定義 4 モータ 5-24
5-8-5 CK3A サンプル PLC 4 モータ 5-26
5-8-6 CK3A サンプル定義 8 モータ 5-28
5-8-7 CK3A サンプル PLC 8 モータ 5-32
5-8-8 電源投入時に CK3A PLC を有効にする 5-34
5-8-9 CK3A PLC の使用 5-35

5-1 ソフトウェア構成の概略

ここでは、理解を深めるためにソフトウェア構成の概要を説明します。

5-1-1 コントローラ-アンプインタフェース

CK3A DirectPWM アンプは、PWM ケーブルを介してコントローラとインタフェースします。PC と直接通信することはなく、構成設定も含んでいません。

ADC ストローブワード（後述）と呼ばれるコントローラの構造体要素を使用して、アンプの機能を制御し、コントローラに報告するステータスビットとデータ情報を指定することができます。

5-1-2 要約

以下のトピックについて、「第 5 章 ソフトウェア構成」で説明します。

トピック	内容
基本構成	アンプの基本構成を可能にする ADC ストローブワードの「デフォルト」設定を含む要素の設定
ADC ストローブワードの詳細	以下を含む ADC ストローブワードの説明： • アンプ機能の制御フィールド • ステータスビットの制御フィールド • データ報告の制御フィールド
アンプ機能の詳細	以下を行う方法の説明と例： • 表示異常を解除する • ダイナミックブレーキを設定する • ファンの動作モードを制御する • 放電モードを設定する（主電源切断時残留エネルギー） • 低電圧動作を設定する
ADC レジスタの詳細	ステータスビットとデータ情報が存在するレジスタの一覧
ステータスビットの詳細	すべてのステータスビットの一覧とアドレス
データ情報の詳細	データ情報の一覧とアドレス
CK3A サンプル PLC	ユーザがアンプ機能を簡単に制御し、ステータスビットとデータ情報を読み取ることができる 4 チャネルと 8 チャネルのサンプルコード

参考

CK3A アンプの基本的な（デフォルトの）動作にはアンプの機能、ステータスビット、およびデータ情報は必要ありません。アプリケーションによってそれらの使用が必要な場合があります。

5-1-3 対象となるユーザの方々

「第 5 章 ソフトウェア構成」では、Power PMAC Gate3 とチャネルアーキテクチャを一般的に理解されている方々が対象です。

5-1-4 対象となるハードウェア

「第 5 章 ソフトウェア構成」は、以下の軸カードを使用する Power PMAC コントローラの CK3M および UMAC シリーズに関連しています。

- 形 CK3W-AX□3□3□
- 形 ACC-24E3 3-4002A5050-□□□□-□□□

5-1-5 Gate3 エイリアス名

IDE スクリプト環境では、CK3WAX[i]および ACC-24E3[i]に関連付けられている構造体要素はハードウェアエイリアス名です。それらは Gate3[i]と同じです。簡単にするために、「第 5 章 ソフトウェア構成」のすべての例で Gate3[i]の命名法を使用します。

5-2 基本構成

ここでは、CK3A DirectPWM アンプの基本構成を可能にするために必要な最小限の設定について説明します。

参考

- ここでは、モータとエンコーダの設定については説明しません。その目的には、「付録」の章を参照してください。
- ここで説明する設定は、電源を入れ直した後に CK3A DirectPWM アンプを正しく起動するために、Power PMAC メモリ(および IDE プロジェクト)に保存する必要があります。

5-2-1 PWM 周波数

CK3A DirectPWM アンプを適切に動作させるには、PWM 周波数が[8～20]kHz の範囲にある必要があります。PWM 周波数は、次の関係を持つ Power PMAC フェーズ周波数(Gate3[i].PhaseFreq の設定)と PWM Frequency (Gate3[i].Chan[j].PwmFreqMult の設定)によって決定されます。

$$\text{PWM 周波数} = \frac{\text{Gate3[i].Chan[j].PwmFreqMult} + 1}{2} \times \text{フェーズ周波数}$$

5-2-2 Gate3 設定

Gate3[i].AdcAmpStrobe = $901001
Gate3[i].AdcAmpHeaderBits = 4
Gate3[i].AdcAmpClockDiv = 5

使用上の注意

4 台の Gate3[i]アンプのうち 1 台を低電圧モードで使用する場合は、
Gate3[i].AdcAmpStrobe = $ B01001 を設定します。

5-2-3 チャネル設定

形 CK3A-G305L

Gate3[i].Chan[j].PwmDeadTime = 2 / 0.0533
Gate3[i].Chan[j].PackInData = 0
Gate3[i].Chan[j].PackOutData = 0

形 CK3A-G310L、形 CK3A-G320L

Gate3[i].Chan[j].PwmDeadTime = 3 / 0.0533
Gate3[i].Chan[j].PackInData = 0
Gate3[i].Chan[j].PackOutData = 0

5-2-4 モータ設定

Motor[x].AdcMask = $FFFF0000
Motor[x].CurrentNullPeriod = 0

PMAC I2T 設定の場合、各形式の最大 ADC 値は下表のとおりです。

形式	最大 ADC
形 CK3A-G305L	15.735
形 CK3A-G310L	31.470
形 CK3A-G320L	93.844

安全上の要点

Motor[x].I2tSet、Motor[x].I2tTrip、Motor[x].MaxDac などの PMAC I2T 設定を正しく設定してください。モータ定格電流がアンプよりも低い場合特に注意してください。モータを焼損させるおそれがあります。

5-3 ADC ストローブワードの詳細

ここでは、ADC ストローブワードの詳細について説明します。このソフトウェア設定では、ユーザは次のことができます。

- アンプ機能を制御する
- アンプステータスビットにアクセスする
- アンプのデータ情報にアクセスする

ADC ストローブワードの構造体要素は Gate3[i].AdcAmpStrobe です。

5-3-1 概要

IDE スクリプト環境では、ADC ストローブワードは、CK3A DirectPWM アンプに関連する次のフィールドで構成される 24 ビット要素です。

Hex	9				0				1				0				0				1			
ビット #	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
二進数	1	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1

固定

異常解除: =1 の後 =0

ファン制御: =0 自動, =1 ON

ダイナミックブレーキ制御: =1 OFF, =0 ON

低電圧モード: =0 OFF, =1 ON

バス放電制御: =0 OFF, =1 ON

固定

=0: ステータスビットセット2を報告
=1: ステータスビットセット3を報告
=2: 電圧(÷4)を報告 (DC V単位)
=3: 温度を報告 (℃単位)
=6: ファームウエアバージョンを報告
=7: 電流定格を報告
その他の設定は内部用途です

5-3-2 明示的アドレス

Gate3[i].AdcAmpStrobe 構造体要素は、ビット単位（個々のビット）のマッピングには使用できません。必要に応じて、代わりに明示的アドレス指定を使用する必要があります。

以下は、最初の 8 つの Gate3[i]インデックスのアドレスの一覧で、通常は最初の 32 のアンプチャネルに対応します。

インデックス i	構造体要素	チャネル	アドレス
0	Gate3[0].AdcAmpStrobe	1～4	**$90020C**
1	Gate3[1].AdcAmpStrobe	5～8	**$90420C**
2	Gate3[2].AdcAmpStrobe	9～12	**$90820C**
3	Gate3[3].AdcAmpStrobe	13～16	**$90C20C**
4	Gate3[4].AdcAmpStrobe	17～20	**$91020C**
5	Gate3[5].AdcAmpStrobe	21～24	**$91420C**
6	Gate3[6].AdcAmpStrobe	25～28	**$91820C**
7	Gate3[7].AdcAmpStrobe	29～32	**$91C20C**

参考

- ドル記号（$）は、PMAC スクリプト言語の 16 進数値を示します。
- 明示的アドレスは、Gate3[i].AdcAmpStrobe.a から Sys.piom を減算し、結果を 16 進数で読み取ることによって検出されます。

5-4 アンプ機能の詳細

ここでは、コントローラから設定できるアンプ機能について説明します。

5-4-1 特長

ADC ストローブワードによって制御されるアンプ機能により、ユーザは次のことが可能になります。

- バス放電モードを設定する
- 表示異常を解除（リセット）する
- ダイナミックブレーキモードを設定する
- ファンの動作モードを設定する
- 低電圧モードを設定する

参考

- アンプ機能は、同じ Gate3[i]上の 4 台のアンプチャネル全体に影響します。
- 希望する ADC ストローブワードの設定をここに記載する例を参考に見つけ、その後の電源投入までに IDE プロジェクトファイルまたは構成に保存してください。

5-4-2 バス放電制御

ADC ストローブワードのビット#5 は、主回路の電源が切断されたときのアンプの残留エネルギー放電モードを制御します。

= 0 放電 OFF（推奨）

ADC ストローブワードのビット#5 を 0 に設定した場合、主回路の電源が切断されると、アンプは残留エネルギーをゆっくりと消費します。主電源を入れ直すまでの時間が短いため、このモードを推奨します。

使用上の注意

このモードでは、主電源を入れ直すまでに最低 10 秒待機してください。

参考

このモードでの残留エネルギー（<DC36V）の放電には、形 CK3A-G305L と形 CK3A-G310L では 5 分以下、形 CK3A-G320L では 6 分以下程度かかります。

次の例は、通常アンプチャネル 1～4 に関連付けられている Gate3 インデックス 0 を使用して、IDE ターミナルウィンドウからバス放電用の ADC ストローブワードを OFF に設定する簡単な方法を示しています。

```
Gate3[0].AdcAmpStrobe = (Gate3[0].AdcAmpStrobe & $FFFFDF) | $0
```

= 1 放電 ON

ADC ストローブワードのビット#5 を 1 に設定した場合、主回路の電源が切断されると、アンプは残留エネルギーを内部または外部のシャント抵抗にすばやく放電します。

安全上の要点

シャント抵抗が接続されていない場合、アンプは帯電したままになります。

- ユーザは完全に放電を成功させるために、主回路の電源を入れ直すまでに毎回最低 1 分間待機する必要があります。
- この機能は、ユニットを開いて安全に作業する必要がある修理の場合に主に役立ちます。
- この機能は、緊急停止またはメンテナンス中にすべてのエネルギーを機械設備から除去することが安全上義務付けられている一部のシステムで役立つ場合があります。

使用上の注意

このモードでは：

- ユーザは、主回路の電源を入れ直すまでに最低 1 分間待機する必要があります。
- 正常に動作させるには、シャント抵抗（内部または外部）を接続する必要があります。

参考

このモードでの残留エネルギー（<DC36V）の放電には、最長 2.5 秒かかります。

次の例は、通常アンプチャネル 1～4 に関連付けられている Gate3 インデックス 0 を使用して、IDE ターミナルウィンドウからバス放電用の ADC ストローブワードを ON に設定する簡単な方法を示しています。

```
Gate3[0].AdcAmpStrobe = (Gate3[0].AdcAmpStrobe & $FFFFDF) | $20
```

5-4-3 異常解除

ADC ストローブワードのビット＃16 は、異常状態が除去された後、アンプの表示異常を解除します。

参考

- ソフトスタート（A2）、ADC オフセット（A9）、および短絡（AC）異常を解除するためには、アンプのロジック電源再投入が必要です。
- この機能は、アンプの表示異常だけを解除します。PMAC のアンプ異常（Motor[x].AmpFault）状態は解除されません。
- PMAC でアンプの異常状態を解除する一般的なコマンドはありません。異常状態が解消され、モータが有効になっている場合に、自動的に解除されます。

異常解除操作

4 台の Gate3[i]アンプのいずれかの表示異常を解除するには、ADC ストローブワードのビット＃16 を最低 10 ミリ秒の間 1 に設定してから、通常動作のために 0 に戻します。

次の例は、通常アンプチャネル 1～4 に関連付けられている Gate3 インデックス 0 を使用して、IDE ターミナルウィンドウから ADC ストローブワードビット＃16 を 1 に設定してから 0 に設定する簡単な方法を示しています。

```
Gate3[0].AdcAmpStrobe = (Gate3[0].AdcAmpStrobe & $FEFFFF) | $10000
```

```
Gate3[0].AdcAmpStrobe = (Gate3[0].AdcAmpStrobe & $FEFFFF) | $0
```

5-4-4 ファン制御

ADC ストローブワードのビット＃17 は、アンプのファン動作を制御します。

= 0 ファン自動

ADC ストローブワードのビット＃17 を 0 に設定した場合、4 台の Gate3[i]アンプすべてに関してファンが自動モードに設定されます。電源モジュールの温度が 35℃に達すると ON になり、32℃に戻ると OFF になります。

次の例は、通常アンプチャネル 1～4 に関連付けられている Gate3 インデックス 0 を使用して、IDE ターミナルウィンドウから ADC ストローブワードをファン自動モードに設定する簡単な方法を示しています。

```
Gate3[0].AdcAmpStrobe = (Gate3[0].AdcAmpStrobe & $FDFFFF) | $0
```

= 1 ファン常時 ON

ADC ストローブワードのビット＃17 を 1 に設定した場合、4 台の Gate3[i]アンプすべてに関してファンが常時 ON に設定されます。

次の例は、通常アンプチャネル 1～4 に関連付けられている Gate3 インデックス 0 を使用して、IDE ターミナルウィンドウから ADC ストローブワードをファン常時 ON に設定する簡単な方法を示しています。

```
Gate3[0].AdcAmpStrobe = (Gate3[0].AdcAmpStrobe & $FDFFFF) | $20000
```

5-4-5 ダイナミックブレーキ制御

ADC ストローブワードのビット＃20 は、ダイナミックブレーキ機能を制御します。

安全上の要点

- ダイナミックブレーキ機能は、異常時の停止を目的としているため、短時間設定です。通常の運転で動作を停止するために使用しないでください。
- 電源 OFF かつダイナミックブレーキ機能 ON のときは、外部駆動源を使用してモータを駆動しないでください。
- 一般的なガイドラインとして、ダイナミックブレーキの使用間隔は最低 1 分空けてください。
- 垂直モータを停止するために、カウンターバランスなしでダイナミックブレーキに依存しないでください。
- アンプが有効になっている場合、ダイナミックブレーキ設定は有効になりません。ダイナミックブレーキ機能の ON/OFF を設定する前に、アンプ有効信号を OFF にする必要があります。

ダイナミックブレーキ機能は、ON の場合、停止（サーボオフ）イベント後にモータが自由に惰性運転するのを防ぎます。この状況は、以下に当てはまります。

- ユーザコマンドまたは致命的フォローエラーなどの PMAC 機能によってモータが停止（サーボ OFF）した場合
- 主回路電源 OFF、ロジック電源 OFF などのアンプエラーの場合

= 0 ダイナミックブレーキ ON

ADC ストローブワードのビット＃20 を 0 に設定した場合、4 台の Gate3[i]アンプすべてに関してダイナミックブレーキが ON に設定されます。

使用上の注意

- ダイナミックブレーキ機能が ON の場合、アンプを有効にして、最低 50 ミリ秒待機してからモーションコマンドを使用してください。
- （動作中のモータ停止後）ダイナミックブレーキが実行された場合、最低 500 ミリ秒待機してからアンプを再度有効にしてください。
- ダイナミックブレーキ機能が ON かつモータが動作中の場合、ユーザは最初に JOG/を発行してモータを減速して停止させてから、停止（サーボ OFF）コマンドを発行する必要があります。
- アンプが有効になっているときは、ダイナミックブレーキ設定の ON/OFF を切り替えないでください。

次の例は、通常アンプチャネル 1～4 に関連付けられている Gate3 インデックス 0 を使用して、IDE ターミナルウィンドウから ADC ストローブワードをダイナミックブレーキ機能 ON に設定する簡単な方法を示しています。

```
Gate3[0].AdcAmpStrobe = (Gate3[0].AdcAmpStrobe & $EFFFFF) | $0
```

= 1 ダイナミックブレーキ OFF

ADC ストローブワードのビット＃20 を 1 に設定した場合、4 台の Gate3[i]アンプすべてに関してダイナミックブレーキが OFF に設定されます。

次の例は、通常アンプチャネル 1～4 に関連付けられている Gate3 インデックス 0 を使用して、IDE ターミナルウィンドウから ADC ストローブワードをダイナミックブレーキ機能 OFF に設定する簡単な方法を示しています。

```
Gate3[0].AdcAmpStrobe = (Gate3[0].AdcAmpStrobe & $EFFFFF) | $100000
```

5-4-6 低電圧（LV）モード

ADC ストローブワードのビット＃21 は、アンプを低電圧の主回路電源で使用するかどうかを指定します。

- 形 CK3A-G305L の場合、主回路電源電圧は DC48V となります。
- 形 CK3A-G310L の場合、この機能を使用するには工場での変更が必要です。
- 形 CK3A-G320L の場合、主回路電源電圧は DC100V となります。
- Gate3[i]インターフェースに接続された 4 台のアンプのうち少なくとも 1 台を低電圧範囲で使用する場合は、Gate3[i]インターフェースの ADC ストローブワードを低電圧モードに設定する必要があります。同じ Gate3[i]上の他のアンプは、引き続き AC 入力で使用できます。

参考

低電圧モードが設定され、主電源（DC48V、形 CK3A-G305L の場合）が印加されている場合、アンプは左側の 7 セグメント表示に L を表示します。

= 0 正常モード

ADC ストローブワードのビット＃21 を 0 に設定した場合、Gate3[i] インターフェイスに接続された 4 台のアンプのいずれも低電圧範囲で動作できません。

次の例は、通常アンプチャネル 1～4 に関連付けられている Gate3 インデックス 0 を使用して、IDE ターミナルウィンドウから ADC ストローブワードを正常モードに設定する簡単な方法を示しています。

```
Gate3[0].AdcAmpStrobe = (Gate3[0].AdcAmpStrobe & $DFFFFF) | $0
```

= 1 低電圧モード

ADC ストローブワードのビット＃21 を 1 に設定した場合、Gate3[i] インターフェイスに接続された 4 台のアンプのいずれも低電圧主電源入力範囲で動作できます。

次の例は、通常アンプチャネル 1～4 に関連付けられている Gate3 インデックス 0 を使用して、IDE ターミナルウィンドウから ADC ストローブワードを低電圧モードに設定する簡単な方法を示しています。

```
Gate3[0].AdcAmpStrobe = (Gate3[0].AdcAmpStrobe & $DFFFFF) | $200000
```

使用上の注意

形 CK3A-G305L の場合、低電圧範囲の主電源で適切に動作させるには、主回路電源接続コネクタ（CN1）の低電圧短絡線を取り付ける必要があります。

5-5 ADC レジスタの詳細

ここでは、ADC レジスタの詳細について説明します。

Power PMAC コントローラ（通常は 1 台のアンプに接続）の各チャネルには、次のように定義された 2 つの ADC レジスタがあります。

名称	構造体要素
ADC **A**	Gate3[i].Chan[j].AdcAmp[**0**]
ADC **B**	Gate3[i].Chan[j].AdcAmp[**1**]

5-5-1 ADC A の情報

ADC A は、次の情報を伝達する 32 ビットレジスタです。

ビット	ADC A の情報	ADC ストローブワード設定
7:0	内部使用	-
15:8	**ステータスビットセット 1**	**「正しい」ストローブワード設定**
31:16	モータ U 相電流フィードバック	「正しい」ストローブワード設定

5-5-2 ADC B の情報

ADC B は、次の情報を伝達する 32 ビットレジスタです。

ビット	ADC B の情報	ADC ストローブワード設定
7:0	内部使用	-
15:08	**ステータスビットセット 2**	**ビット[11:08] = 0**
	ステータスビットセット 3	**ビット[11:08] = 1**
	DC バス電圧	**ビット[11:08] = 2**
	パワーモジュール温度	**ビット[11:08] = 3**
	アンプファームウェア	**ビット[11:08] = 6**
	アンプの電流定格	**ビット[11:08] = 7**
31:16	モータ V 相電流フィードバック	「正しい」ストローブワード設定

参考

ビット[11:08]の他のすべての設定は、内部使用するデータを報告します。

5-5-3 ADC レジスタ明示的アドレス

Gate3[i].Chan[j].AdcAmp[k]構造体要素は、ビット単位（個々のビット）のマッピングには使用できません。代わりに明示的アドレス指定を使用する必要があります。以下は、最初の 8 つの Gate3[i]インデックスのアドレスの一覧で、通常は最初の 32 のアンプチャネルに対応します。

	CH	ADC A 構造体要素	アドレス	ADC B 構造体要素	アドレス
Gate3[0]	1	Gate3[0].Chan[0].AdcAmp[0]	**$900020**	Gate3[0].Chan[0].AdcAmp[1]	**$900024**
	2	Gate3[0].Chan[1].AdcAmp[0]	**$9000A0**	Gate3[0].Chan[1].AdcAmp[1]	**$9000A4**
	3	Gate3[0].Chan[2].AdcAmp[0]	**$900120**	Gate3[0].Chan[2].AdcAmp[1]	**$900124**
	4	Gate3[0].Chan[3].AdcAmp[0]	**$9001A0**	Gate3[0].Chan[3].AdcAmp[1]	**$9001A4**
Gate3[1]	5	Gate3[1].Chan[0].AdcAmp[0]	**$904020**	Gate3[1].Chan[0].AdcAmp[1]	**$904024**
	6	Gate3[1].Chan[1].AdcAmp[0]	**$9040A0**	Gate3[1].Chan[1].AdcAmp[1]	**$9040A4**
	7	Gate3[1].Chan[2].AdcAmp[0]	**$904120**	Gate3[1].Chan[2].AdcAmp[1]	**$904124**
	8	Gate3[1].Chan[3].AdcAmp[0]	**$9041A0**	Gate3[1].Chan[3].AdcAmp[1]	**$9041A4**
Gate3[2]	9	Gate3[2].Chan[0].AdcAmp[0]	**$908020**	Gate3[2].Chan[0].AdcAmp[1]	**$908024**
	10	Gate3[2].Chan[1].AdcAmp[0]	**$9080A0**	Gate3[2].Chan[1].AdcAmp[1]	**$9080A4**
	11	Gate3[2].Chan[2].AdcAmp[0]	**$908120**	Gate3[2].Chan[2].AdcAmp[1]	**$908124**
	12	Gate3[2].Chan[3].AdcAmp[0]	**$9081A0**	Gate3[2].Chan[3].AdcAmp[1]	**$9081A4**
Gate3[3]	13	Gate3[3].Chan[0].AdcAmp[0]	**$90C020**	Gate3[3].Chan[0].AdcAmp[1]	**$90C024**
	14	Gate3[3].Chan[1].AdcAmp[0]	**$90C0A0**	Gate3[3].Chan[1].AdcAmp[1]	**$90C0A4**
	15	Gate3[3].Chan[2].AdcAmp[0]	**$90C120**	Gate3[3].Chan[2].AdcAmp[1]	**$90C124**
	16	Gate3[3].Chan[3].AdcAmp[0]	**$90C1A0**	Gate3[3].Chan[3].AdcAmp[1]	**$90C1A4**
Gate3[4]	17	Gate3[4].Chan[0].AdcAmp[0]	**$910020**	Gate3[4].Chan[0].AdcAmp[1]	**$910024**
	18	Gate3[4].Chan[1].AdcAmp[0]	**$9100A0**	Gate3[4].Chan[1].AdcAmp[1]	**$9100A4**
	19	Gate3[4].Chan[2].AdcAmp[0]	**$910120**	Gate3[4].Chan[2].AdcAmp[1]	**$910124**
	20	Gate3[4].Chan[3].AdcAmp[0]	**$9101A0**	Gate3[4].Chan[3].AdcAmp[1]	**$9101A4**
Gate3[5]	21	Gate3[5].Chan[0].AdcAmp[0]	**$914020**	Gate3[5].Chan[0].AdcAmp[1]	**$914024**
	22	Gate3[5].Chan[1].AdcAmp[0]	**$9140A0**	Gate3[5].Chan[1].AdcAmp[1]	**$9140A4**
	23	Gate3[5].Chan[2].AdcAmp[0]	**$914120**	Gate3[5].Chan[2].AdcAmp[1]	**$914124**
	24	Gate3[5].Chan[3].AdcAmp[0]	**$9141A0**	Gate3[5].Chan[3].AdcAmp[1]	**$9141A4**
Gate3[6]	25	Gate3[6].Chan[0].AdcAmp[0]	**$918020**	Gate3[6].Chan[0].AdcAmp[1]	**$918024**
	26	Gate3[6].Chan[1].AdcAmp[0]	**$9180A0**	Gate3[6].Chan[1].AdcAmp[1]	**$9180A4**
	27	Gate3[6].Chan[2].AdcAmp[0]	**$918120**	Gate3[6].Chan[2].AdcAmp[1]	**$918124**
	28	Gate3[6].Chan[3].AdcAmp[0]	**$9181A0**	Gate3[6].Chan[3].AdcAmp[1]	**$9181A4**
Gate3[7]	29	Gate3[7].Chan[0].AdcAmp[0]	**$91C020**	Gate3[7].Chan[0].AdcAmp[1]	**$91C024**
	30	Gate3[7].Chan[1].AdcAmp[0]	**$91C0A0**	Gate3[7].Chan[1].AdcAmp[1]	**$91C0A4**
	31	Gate3[7].Chan[2].AdcAmp[0]	**$91C120**	Gate3[7].Chan[2].AdcAmp[1]	**$91C124**
	32	Gate3[7].Chan[3].AdcAmp[0]	**$91C1A0**	Gate3[7].Chan[3].AdcAmp[1]	**$91C1A4**

参考

- ドル記号（$）は、PMAC スクリプト言語の 16 進数値を示します。
- 明示的アドレスは、Gate3[i].Chan[j].AdcAmp[k].a から Sys.piom を減算し、結果を 16 進数で読み取ることによって検出されます。

5-6 ステータスビットの詳細

ここでは、CK3A アンプからコントローラに報告できる 3 セットのステータスビットの詳細について説明します。

5-6-1 ステータスビットセット 1

ADC ストローブワードの「正しい」設定では必ず、**ADC A**、**Gate3[i].Chan[j].AdcAmp[0]**に次のステータスビットがあります。

ビット#	項目	仕様	異常/表示コード
8	主電源	=0 異常なし、=1 異常	A1
9	ソフトスタート	=0 異常なし、=1 異常	A2
10	STO 状態	= 0トリガなし、= 1トリガあり	A6
11	シャント短絡	=0 異常なし、=1 異常	A3
12	過電圧	=0 異常なし、=1 異常	A5
13	I2T 異常	=0 異常なし、=1 異常	A8
14	短絡	=0 異常なし、=1 異常	AC
15	過負荷	=0 異常なし、=1 異常	AL

Gate3 インデックス 0 チャネル 0 を使用したダイレクトマッピングアクセスの例は、通常、アンプ/モータ#1 に関連付けられています。

```
// CH1、Gate3[0].Chan[0].AdcAmp[0]、ステータスセット 1
PTR Ck3a1PwrFlt           ->U.IO:$900020.8.1
PTR Ck3a1SoftStartFlt     ->U.IO:$900020.9.1
PTR Ck3a1StoStatus        ->U.IO:$900020.10.1
PTR Ck3a1ShuntShortFlt    ->U.IO:$900020.11.1
PTR Ck3a1OverVoltFlt      ->U.IO:$900020.12.1
PTR Ck3a1I2tFlt           ->U.IO:$900020.13.1
PTR Ck3a1ShortFlt         ->U.IO:$900020.14.1
PTR Ck3a1OverLoadFlt      ->U.IO:$900020.15.1
```

5-6-2 ステータスビットセット 2

ADC ストローブワードのビット[11:08]が 0 に設定されている場合、アンプは **ADC B**、**Gate3[i].Chan[j].AdcAmp[1]**でステータスビットセット 2 を報告します。

ビット#	項目	仕様	異常/表示コード
8	電源入力状態	=1 AC 電源投入、=0 切断	A1
10	過熱	=0 異常なし、=1 異常	A7
11	シャント過負荷	=0 異常なし、=1 異常	A4
12	PWM 周波数異常	=0 異常なし、=1 異常	P1
13	PWM コマンド異常	=0 異常なし、=1 異常	P2
14	アンプ有効状態	=0 無効、=1 有効	0（有効な場合）

参考

他のビット値は内部使用のために予約済です。

Gate3 インデックス 0 チャネル 0 を使用したダイレクトマッピングアクセスの例は、通常、アンプ/モータ#1 に関連付けられています。

```
Gate3[0].AdcAmpStrobe = (Gate3[0].AdcAmpStrobe & $FFF0FF) | $0
```

```
// CH1、Gate3[0].Chan[0].AdcAmp[1] ステータスセット 2
PTR Ck3a1PwrStatus        ->U.IO:$900024.8.1
PTR Ck3a1OverTempFlt      ->U.IO:$900024.10.1
PTR Ck3a1ShuntOverLFlt    ->U.IO:$900024.11.1
PTR Ck3a1PwmFreqFlt       ->U.IO:$900024.12.1
PTR Ck3a1PwmCmdFlt        ->U.IO:$900024.13.1
PTR Ck3a1AmpEna           ->U.IO:$900024.14.1
```

5-6-3 ステータスビットセット 3

ADC ストローブワードのビット[11:08]が 1 に等しい場合、アンプは **ADC B**、**Gate3[i].Chan[j].AdcAmp[1]**でステータスビットセット 3 を報告します。

ビット#	項目	仕様	異常/表示コード
10	I2T インテグレータ状態	=0 OFF、=1 ON	-
12	ADC オフセット	=0 異常なし、=1 異常	A9
14	アンプレディ	=0 レディ、=1 レディではない	-

参考

- 他のビット値は内部使用のために予約済です。
- アンプの初期化中（約 6 秒）、アンプレディビットは 1 に設定されます。起動が完了すると 0 に設定され、アンプの準備ができたことを示します。

Gate3 インデックス 0 チャネル 0 を使用したダイレクトマッピングアクセスの例は、通常、アンプ/モータ#1 に関連付けられています。

```
Gate3[0].AdcAmpStrobe = (Gate3[0].AdcAmpStrobe & $FFF0FF) | $100
```

```
// CH1、Gate3[0].Chan[0].AdcAmp[1] ステータスセット 3
PTR Ck3a1I2tint       ->U.IO:$900024.10.1
PTR Ck3a1AdcOffsetFlt ->U.IO:$900024.12.1
PTR Ck3a1Ready        ->U.IO:$900024.14.1
```

5-7 アンプデータの詳細

ここでは、CK3A アンプからコントローラに報告できるデータの詳細について説明します。

概要と例

ADC ストローブワードのビット[11:08]は、どのアンプデータをコントローラに報告するかも指定します。報告したデータは ADC B ビット[15:08]にあります。

ビット#[11:08]設定	報告するアンプデータ
= 2	バス電圧(4 で割った値:DCV 単位)
= 3	アンプパワーモジュール温度(℃単位)
= 6	アンプファームウェアバージョン
= 7	アンプ電流定格(アンペア単位) 形 CK3A-G305L: 5 形 CK3A-G310L: 10 形 CK3A-G320L: 20

以下は、IDE ターミナルウィンドウから簡単にアクセスするための例です。通常、アンプ/モータ#1 に関連付けられている Gate3 インデックス 0 チャネル 0 を使用しています。

グローバル定義

```
GLOBAL Ck3a1DcBus, Ck3a1Temp, Ck3a1FwVer, Ck3a1CurRating
```

DC バス電圧

```
Gate3[0].AdcAmpStrobe = (Gate3[0].AdcAmpStrobe & $FFF0FF) | $200
```

```
Ck3a1DcBus = ((Gate3[0].Chan[0].AdcAmp[1] & $FF00) >> 8) * 4
```

パワーモジュール温度

```
Gate3[0].AdcAmpStrobe = (Gate3[0].AdcAmpStrobe & $FFF0FF) | $300
```

```
Ck3a1Temp = (Gate3[0].Chan[0].AdcAmp[1] & $FF00) >> 8
```

アンプファームウェアバージョン

```
Gate3[0].AdcAmpStrobe = (Gate3[0].AdcAmpStrobe & $FFF0FF) | $600
```

```
Ck3a1FwVer = (Gate3[0].Chan[0].AdcAmp[1] & $FF00) >> 8
```

アンプの電流定格

```
Gate3[0].AdcAmpStrobe = (Gate3[0].AdcAmpStrobe & $FFF0FF) | $700
```

```
Ck3a1CurRating = (Gate3[0].Chan[0].AdcAmp[1] & $FF00) >> 8
```

5-8 CK3A サンプル PLC

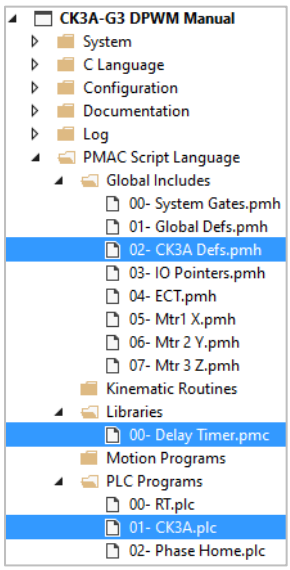
ここでは、推奨変数、4 モータおよび 8 モータ用のポインタ定義、サンプル PLC プログラムの Power PMAC スクリプトの一覧を提供し、ユーザが次のことを簡単に行えるようにします。

- アンプ機能を制御する
- アンプデータを読み取る
- アンプステータスビットを読み取る

5-8-1 サンプル PLC の実装

ここに示すサンプル PLC を正常に実装するには、次の手順が必要です。

1. IDE プロジェクトに遅延タイマーサブルーチンが存在することを確認してください。
2. 使用する Sys.Udata[i]構造が既存のアプリケーションと競合しないことを確認してください。
3. **CK3A 定義**を IDE プロジェクトの **Global Includes** フォルダ下のファイルにコピーします。
4. **CK3A PLC** を IDE プロジェクトの **PLC Programs** フォルダ下のファイルにコピーします。
5. 電源投入時に CK3A PLC を有効にします（オプション）。
6. ダウンロードが成功したことを確認してください（コンパイルエラーなし）。



参考

IDE プロジェクトツリー内のファイル名は、運転に重要ではありません。

5-8-2 遅延タイマーサブルーチン

遅延タイマーサブルーチンは、Power PMAC スクリプト PLC プログラムに時間遅延を導入するために使用します。この遅延タイマーサブルーチンは、ほとんどの IDE プロジェクトテンプレートに存在します。参考のためここに示します。

```
OPEN SUBPROG DelayTimer
SUB: sec (DelayTimeSec)
LOCAL EndTimeSec
EndTimeSec = Sys.Time + DelayTimeSec
WHILE (EndTimeSec > Sys.Time){}
RETURN

SUB: msec (DelayTimeMsec)
LOCAL EndTimeMsec
EndTimeMsec = Sys.Time + DelayTimeMsec * 0.001
WHILE (EndTimeMsec > Sys.Time){}
RETURN
CLOSE
```

5-8-3 ユーザ共有メモリの説明

特定のチャネルの ADC A レジスタにステータスビットセット 1 を報告します。これらのビットには、ADC ストローブワードを特別に設定しなくてもいつでもアクセスできます。

ただし、ステータスビットセット 2、および 3 は、ADC B レジスタの同じ位置を共有しています。ADC ストローブ設定のビット[11:08]が、現在報告しているセットを示します。

- ADC ストローブワードのビット[11:08]が 0 に等しい場合、ステータスセット 2 を報告します
- ADC ストローブワードのビット[11:08]が 1 に等しい場合、ステータスセット 3 を報告します

このため、PLC でビット[11:08]の設定が（0 から 1 の間で）変更されるたびに、保持レジスタとして「ミラーワード」を使用します。

「符号なし」ユーザ共有メモリレジスタ Sys.Udata[i]が、この種類の手順には最適です。ここに示すサンプルコードで使用するレジスタは、次のように定義されています。

ステータスセット 2	Sys.Udata[i]	アドレス	ステータスセット 3	Sys.Udata[i]	アドレス
チャネル 1	**Sys.Udata[256000]**	**$FA000**	チャネル 1	**Sys.Udata[256040]**	**$FA0A0**
チャネル 2	**Sys.Udata[256001]**	**$FA004**	チャネル 2	**Sys.Udata[256041]**	**$FA0A4**
チャネル 3	**Sys.Udata[256002]**	**$FA008**	チャネル 3	**Sys.Udata[256042]**	**$FA0A8**
チャネル 4	**Sys.Udata[256003]**	**$FA00C**	チャネル 4	**Sys.Udata[256043]**	**$FA0AC**
チャネル 5	**Sys.Udata[256004]**	**$FA010**	チャネル 5	**Sys.Udata[256044]**	**$FA0B0**
チャネル 6	**Sys.Udata[256005]**	**$FA014**	チャネル 6	**Sys.Udata[256045]**	**$FA0B4**
チャネル 7	**Sys.Udata[256006]**	**$FA018**	チャネル 7	**Sys.Udata[256046]**	**$FA0B8**
チャネル 8	**Sys.Udata[256007]**	**$FA01C**	チャネル 8	**Sys.Udata[256047]**	**$FA0BC**

PLC で提案する方法は次のとおりです。

- ADC ストローブワードのビット[11:08]を= 0 に設定する
- ステータスセット 2 を Sys.Udata[25600x]にコピーする
- ADC ストローブワードのビット[11:08]を= 1 に設定する
- ステータスセット 3 を Sys.Udata[25604x]にコピーする

個々のビットを Sys.Udata[i]レジスタにマッピングすると、各セットのステータスビットが反映されます。

参考

- これらの Sys.Udata[i]レジスタをアプリケーションで現在使用していないことを確認してください。
- レジスタインデックスを変更する場合は注意が必要です。
- 「符号なし」ユーザ共有メモリレジスタの明示的アドレスは、sys.Udata[i].a から sys.pushm を減算し、結果を 16 進数で読み取ることで見つけることができます。

5-8-4 CK3A サンプル定義 4 モータ

```
// GATE3[0]、CH1～4 アンプ機能 ====================== //
#define Gate30WpDis   Gate3[0].WpKey = $AAAAAAAA

GLOBAL Ck3a0Clrf, Ck3a0FanCtrl, Ck3a0BusDisCtrl, Ck3a0DbCtrl

PTR Gate30DataCtrl        ->U.IO:$90020C.16.4
PTR Ck3a0BusDisState      ->U.IO:$90020C.13.1
PTR Ck3a0ClrfBit          ->U.IO:$90020C.24.1
PTR Ck3a0FanState         ->U.IO:$90020C.25.1
PTR Ck3a0DbState          ->U.IO:$90020C.28.1

// CH1～4 ADC B レジスタ [15:08] ============================ //
PTR Ck3a1AdcBData         ->U.IO:$900024.8.8  // Gate3[0].Chan[0].AdcAmp[1]
PTR Ck3a2AdcBData         ->U.IO:$9000A4.8.8  // Gate3[0].Chan[1].AdcAmp[1]
PTR Ck3a3AdcBData         ->U.IO:$900124.8.8  // Gate3[0].Chan[2].AdcAmp[1]
PTR Ck3a4AdcBData         ->U.IO:$9001A4.8.8  // Gate3[0].Chan[3].AdcAmp[1]

// CH1～4 アンプデータ ============================== //
GLOBAL Ck3a1DcBus, Ck3a1Temp, Ck3a1FwVer, Ck3a1CurRating
GLOBAL Ck3a2DcBus, Ck3a2Temp, Ck3a2FwVer, Ck3a2CurRating
GLOBAL Ck3a3DcBus, Ck3a3Temp, Ck3a3FwVer, Ck3a3CurRating
GLOBAL Ck3a4DcBus, Ck3a4Temp, Ck3a4FwVer, Ck3a4CurRating

// CH1、Gate3[0].Chan[0].AdcAmp[0]、ステータスセット 1 ============== //
PTR Ck3a1PwrFlt           ->U.IO:$900020.8.1
PTR Ck3a1SoftStartFlt     ->U.IO:$900020.9.1
PTR Ck3a1StoStatus        ->U.IO:$900020.10.1
PTR Ck3a1ShuntShortFlt    ->U.IO:$900020.11.1
PTR Ck3a1OverVoltFlt      ->U.IO:$900020.12.1
PTR Ck3a1I2tFlt           ->U.IO:$900020.13.1
PTR Ck3a1ShortFlt         ->U.IO:$900020.14.1
PTR Ck3a1OverLoadFlt      ->U.IO:$900020.15.1

// CH2、Gate3[0].Chan[1].AdcAmp[0]、ステータスセット 1 ============== //
PTR Ck3a2PwrFlt           ->U.IO:$9000A0.8.1
PTR Ck3a2SoftStartFlt     ->U.IO:$9000A0.9.1
PTR Ck3a2StoStatus        ->U.IO:$9000A0.10.1
PTR Ck3a2ShuntShortFlt    ->U.IO:$9000A0.11.1
PTR Ck3a2OverVoltFlt      ->U.IO:$9000A0.12.1
PTR Ck3a2I2tFlt           ->U.IO:$9000A0.13.1
PTR Ck3a2ShortFlt         ->U.IO:$9000A0.14.1
PTR Ck3a2OverLoadFlt      ->U.IO:$9000A0.15.1

// CH3、Gate3[0].Chan[2].AdcAmp[0]、ステータスセット 1 ============== //
PTR Ck3a3PwrFlt           ->U.IO:$900120.8.1
PTR Ck3a3SoftStartFlt     ->U.IO:$900120.9.1
PTR Ck3a3StoStatus        ->U.IO:$900120.10.1
PTR Ck3a3ShuntShortFlt    ->U.IO:$900120.11.1
PTR Ck3a3OverVoltFlt      ->U.IO:$900120.12.1
PTR Ck3a3I2tFlt           ->U.IO:$900120.13.1
PTR Ck3a3ShortFlt         ->U.IO:$900120.14.1
PTR Ck3a3OverLoadFlt      ->U.IO:$900120.15.1
```

```
// CH4、Gate3[0].Chan[3].AdcAmp[0]、ステータスセット 1 ============== //
PTR Ck3a4PwrFlt           ->U.IO:$9001A0.8.1
PTR Ck3a4SoftStartFlt     ->U.IO:$9001A0.9.1
PTR Ck3a4StoStatus        ->U.IO:$9001A0.10.1
PTR Ck3a4ShuntShortFlt    ->U.IO:$9001A0.11.1
PTR Ck3a4OverVoltFlt      ->U.IO:$9001A0.12.1
PTR Ck3a4I2tFlt           ->U.IO:$9001A0.13.1
PTR Ck3a4ShortFlt         ->U.IO:$9001A0.14.1
PTR Ck3a4OverLoadFlt      ->U.IO:$9001A0.15.1

// CH1、SYS.UDATA[256000] ステータスセット 2 ======================== //
PTR Ck3a1PwrStatus        ->U.USER:$FA000.0.1
PTR Ck3a1OverTempFlt      ->U.USER:$FA000.2.1
PTR Ck3a1ShuntOverLFlt    ->U.USER:$FA000.3.1
PTR Ck3a1PwmFreqFlt       ->U.USER:$FA000.4.1
PTR Ck3a1PwmCmdFlt        ->U.USER:$FA000.5.1
PTR Ck3a1AmpEna           ->U.USER:$FA000.6.1

// CH2、SYS.UDATA[256001] ステータスセット 2 ======================== //
PTR Ck3a2PwrStatus        ->U.USER:$FA004.0.1
PTR Ck3a2OverTempFlt      ->U.USER:$FA004.2.1
PTR Ck3a2ShuntOverLFlt    ->U.USER:$FA004.3.1
PTR Ck3a2PwmFreqFlt       ->U.USER:$FA004.4.1
PTR Ck3a2PwmCmdFlt        ->U.USER:$FA004.5.1
PTR Ck3a2AmpEna           ->U.USER:$FA004.6.1

// CH3、SYS.UDATA[256002] ステータスセット 2 ======================== //
PTR Ck3a3PwrStatus        ->U.USER:$FA008.0.1
PTR Ck3a3OverTempFlt      ->U.USER:$FA008.2.1
PTR Ck3a3ShuntOverLFlt    ->U.USER:$FA008.3.1
PTR Ck3a3PwmFreqFlt       ->U.USER:$FA008.4.1
PTR Ck3a3PwmCmdFlt        ->U.USER:$FA008.5.1
PTR Ck3a3AmpEna           ->U.USER:$FA008.6.1

// CH4、SYS.UDATA[256003] ステータスセット 2 ======================== //
PTR Ck3a4PwrStatus        ->U.USER:$FA00C.0.1
PTR Ck3a4OverTempFlt      ->U.USER:$FA00C.2.1
PTR Ck3a4ShuntOverLFlt    ->U.USER:$FA00C.3.1
PTR Ck3a4PwmFreqFlt       ->U.USER:$FA00C.4.1
PTR Ck3a4PwmCmdFlt        ->U.USER:$FA00C.5.1
PTR Ck3a4AmpEna           ->U.USER:$FA00C.6.1

// CH1、SYS.UDATA[256040] ステータスセット 3 ======================== //
PTR Ck3a1I2tint           ->U.USER:$FA0A0.2.1
PTR Ck3a1AdcOffsetFlt     ->U.USER:$FA0A0.4.1
PTR Ck3a1Ready            ->U.USER:$FA0A0.6.1

// CH2、SYS.UDATA[256041] ステータスセット 3 ======================== //
PTR Ck3a2I2tint           ->U.USER:$FA0A4.2.1
PTR Ck3a2AdcOffsetFlt     ->U.USER:$FA0A4.4.1
PTR Ck3a2Ready            ->U.USER:$FA0A4.6.1

// CH3、SYS.UDATA[256042] ステータスセット 3 ======================== //
PTR Ck3a3I2tint           ->U.USER:$FA0A8.2.1
PTR Ck3a3AdcOffsetFlt     ->U.USER:$FA0A8.4.1
PTR Ck3a3Ready            ->U.USER:$FA0A8.6.1

// CH4、SYS.UDATA[256043] ステータスセット 3 ======================== //
PTR Ck3a4I2tint           ->U.USER:$FA0AC.2.1
PTR Ck3a4AdcOffsetFlt     ->U.USER:$FA0AC.4.1
PTR Ck3a4Ready            ->U.USER:$FA0AC.6.1
```

5-8-5 CK3A サンプル PLC 4 モータ

```
OPEN PLC Ck3aPLC
// CH1～4 ステータスビットセット 2
Gate30WpDis Gate30DataCtrl = 0
CALL DelayTimer.msec(1)
Sys.Udata[256000] = Ck3a1AdcBData
Sys.Udata[256001] = Ck3a2AdcBData
Sys.Udata[256002] = Ck3a3AdcBData
Sys.Udata[256003] = Ck3a4AdcBData
CALL DelayTimer.msec(1)

// CH1～4 ステータスビットセット 3
Gate30WpDis Gate30DataCtrl = 1
CALL DelayTimer.msec(1)
Sys.Udata[256040] = Ck3a1AdcBData
Sys.Udata[256041] = Ck3a2AdcBData
Sys.Udata[256042] = Ck3a3AdcBData
Sys.Udata[256043] = Ck3a4AdcBData
CALL DelayTimer.msec(1)

// CH1～4 DC バス電圧
Gate30WpDis Gate30DataCtrl = 2
CALL DelayTimer.msec(1)
IF(Ck3a1Ready == 0) Ck3a1DcBus = Ck3a1AdcBData * 4
IF(Ck3a2Ready == 0) Ck3a2DcBus = Ck3a2AdcBData * 4
IF(Ck3a3Ready == 0) Ck3a3DcBus = Ck3a3AdcBData * 4
IF(Ck3a4Ready == 0) Ck3a4DcBus = Ck3a4AdcBData * 4
CALL DelayTimer.msec(1)

// CH1～4 温度
Gate30WpDis Gate30DataCtrl = 3
CALL DelayTimer.msec(1)
IF(Ck3a1Ready == 0) Ck3a1Temp = Ck3a1AdcBData
IF(Ck3a2Ready == 0) Ck3a2Temp = Ck3a2AdcBData
IF(Ck3a3Ready == 0) Ck3a3Temp = Ck3a3AdcBData
IF(Ck3a4Ready == 0) Ck3a4Temp = Ck3a4AdcBData
CALL DelayTimer.msec(1)

// CH1～4 ファームウェアバージョン
Gate30WpDis Gate30DataCtrl = 6
CALL DelayTimer.msec(1)
IF(Ck3a1Ready == 0) Ck3a1FwVer = Ck3a1AdcBData
IF(Ck3a2Ready == 0) Ck3a2FwVer = Ck3a2AdcBData
IF(Ck3a3Ready == 0) Ck3a3FwVer = Ck3a3AdcBData
IF(Ck3a4Ready == 0) Ck3a4FwVer = Ck3a4AdcBData
CALL  DelayTimer.msec(1)

// CH1～4 電流定格
Gate30WpDis Gate30DataCtrl = 7
CALL DelayTimer.msec(1)
IF(Ck3a1Ready == 0) Ck3a1CurRating = Ck3a1AdcBData
IF(Ck3a2Ready == 0) Ck3a2CurRating = Ck3a2AdcBData
IF(Ck3a3Ready == 0) Ck3a3CurRating = Ck3a3AdcBData
IF(Ck3a4Ready == 0) Ck3a4CurRating = Ck3a4AdcBData
CALL DelayTimer.msec(1)

// GATE3[0]、CH1～4 バス放電制御
IF(Ck3a0BusDisCtrl == 1 && Ck3a0BusDisState == 0) Gate30WpDis Ck3a0BusDisState = 1
IF(Ck3a0BusDisCtrl == 0 && Ck3a0BusDisState == 1) Gate30WpDis Ck3a0BusDisState = 0
```

```
// GATE3[0]、CH1～4 ファン制御
IF(Ck3a0FanCtrl == 1 && Ck3a0FanState == 0) Gate30WpDis Ck3a0FanState = 1
IF(Ck3a0FanCtrl == 0 && Ck3a0FanState == 1) Gate30WpDis Ck3a0FanState = 0

// GATE3[0]、CH1～4 ダイナミックブレーキ制御
IF(Ck3a0DbCtrl == 1 && Ck3a0DbState == 1) Gate30WpDis Ck3a0DbState = 0
IF(Ck3a0DbCtrl == 0 && Ck3a0DbState == 0) Gate30WpDis Ck3a0DbState = 1

// GATE3[0]、CH1～4 異常解除
IF(Ck3a0Clrf == 1)
{
 Gate30WpDis Ck3a0ClrfBit = 1
 CALL DelayTimer.msec(10)
 Gate30WpDis Ck3a0ClrfBit = 0
 Ck3a0Clrf = 0
}

// CH1 ロジック電源 OFF?
IF(Ck3a1Ready == 1) Ck3a1DcBus,4 = 0
IF(Ck3a2Ready == 1) Ck3a2DcBus,4 = 0
IF(Ck3a3Ready == 1) Ck3a3DcBus,4 = 0
IF(Ck3a4Ready == 1) Ck3a4DcBus,4 = 0
CLOSE
```

5-8-6 CK3A サンプル定義 8 モータ

```
// GATE3[0]、CH1～8 アンプ機能 ====================== //
#define Gate30WpDis   Gate3[0].WpKey = $AAAAAAAA
#define Gate31WpDis   Gate3[1].WpKey = $AAAAAAAA

GLOBAL Ck3a0Clrf, Ck3a0FanCtrl, Ck3a0BusDisCtrl, Ck3a0DbCtrl
GLOBAL Ck3a1Clrf, Ck3a1FanCtrl, Ck3a1BusDisCtrl, Ck3a1DbCtrl

PTR Gate30DataCtrl    ->U.IO:$90020C.16.4
PTR Ck3a0BusDisState  ->U.IO:$90020C.13.1
PTR Ck3a0ClrfBit      ->U.IO:$90020C.24.1
PTR Ck3a0FanState     ->U.IO:$90020C.25.1
PTR Ck3a0DbState      ->U.IO:$90020C.28.1

PTR Gate31DataCtrl    ->U.IO:$90420C.16.4
PTR Ck3a1BusDisState  ->U.IO:$90420C.13.1
PTR Ck3a1ClrfBit      ->U.IO:$90420C.24.1
PTR Ck3a1FanState     ->U.IO:$90420C.25.1
PTR Ck3a1DbState      ->U.IO:$90420C.28.1

// CH1～8 ADC B レジスタ [15:08] ============================ //
PTR Ck3a1AdcBData     ->U.IO:$900024.8.8  // Gate3[0].Chan[0].AdcAmp[1]
PTR Ck3a2AdcBData     ->U.IO:$9000A4.8.8  // Gate3[0].Chan[1].AdcAmp[1]
PTR Ck3a3AdcBData     ->U.IO:$900124.8.8  // Gate3[0].Chan[2].AdcAmp[1]
PTR Ck3a4AdcBData     ->U.IO:$9001A4.8.8  // Gate3[0].Chan[3].AdcAmp[1]

PTR Ck3a5AdcBData     ->U.IO:$904024.8.8  // Gate3[1].Chan[0].AdcAmp[1]
PTR Ck3a6AdcBData     ->U.IO:$9040A4.8.8  // Gate3[1].Chan[1].AdcAmp[1]
PTR Ck3a7AdcBData     ->U.IO:$904124.8.8  // Gate3[1].Chan[2].AdcAmp[1]
PTR Ck3a8AdcBData     ->U.IO:$9041A4.8.8  // Gate3[1].Chan[3].AdcAmp[1]

// CH1～8 アンプデータ ============================== //
GLOBAL Ck3a1DcBus, Ck3a1Temp, Ck3a1FwVer, Ck3a1CurRating
GLOBAL Ck3a2DcBus, Ck3a2Temp, Ck3a2FwVer, Ck3a2CurRating
GLOBAL Ck3a3DcBus, Ck3a3Temp, Ck3a3FwVer, Ck3a3CurRating
GLOBAL Ck3a4DcBus, Ck3a4Temp, Ck3a4FwVer, Ck3a4CurRating

GLOBAL Ck3a5DcBus, Ck3a5Temp, Ck3a5FwVer, Ck3a5CurRating
GLOBAL Ck3a6DcBus, Ck3a6Temp, Ck3a6FwVer, Ck3a6CurRating
GLOBAL Ck3a7DcBus, Ck3a7Temp, Ck3a7FwVer, Ck3a7CurRating
GLOBAL Ck3a8DcBus, Ck3a8Temp, Ck3a8FwVer, Ck3a8CurRating

// CH1、Gate3[0].Chan[0].AdcAmp[0]、ステータスセット 1 ============== //
PTR Ck3a1PwrFlt           ->U.IO:$900020.8.1
PTR Ck3a1SoftStartFlt     ->U.IO:$900020.9.1
PTR Ck3a1StoStatus        ->U.IO:$900020.10.1
PTR Ck3a1ShuntShortFlt    ->U.IO:$900020.11.1
PTR Ck3a1OverVoltFlt      ->U.IO:$900020.12.1
PTR Ck3a1I2tFlt           ->U.IO:$900020.13.1
PTR Ck3a1ShortFlt         ->U.IO:$900020.14.1
PTR Ck3a1OverLoadFlt      ->U.IO:$900020.15.1

// CH2、Gate3[0].Chan[1].AdcAmp[0]、ステータスセット 1 ============== //
PTR Ck3a2PwrFlt           ->U.IO:$9000A0.8.1
PTR Ck3a2SoftStartFlt     ->U.IO:$9000A0.9.1
PTR Ck3a2StoStatus        ->U.IO:$9000A0.10.1
PTR Ck3a2ShuntShortFlt    ->U.IO:$9000A0.11.1
PTR Ck3a2OverVoltFlt      ->U.IO:$9000A0.12.1
PTR Ck3a2I2tFlt           ->U.IO:$9000A0.13.1
PTR Ck3a2ShortFlt         ->U.IO:$9000A0.14.1
PTR Ck3a2OverLoadFlt      ->U.IO:$9000A0.15.1
```

```
// CH3、Gate3[0].Chan[2].AdcAmp[0]、ステータスセット 1 ============== //
PTR Ck3a3PwrFlt           ->U.IO:$900120.8.1
PTR Ck3a3SoftStartFlt     ->U.IO:$900120.9.1
PTR Ck3a3StoStatus        ->U.IO:$900120.10.1
PTR Ck3a3ShuntShortFlt    ->U.IO:$900120.11.1
PTR Ck3a3OverVoltFlt      ->U.IO:$900120.12.1
PTR Ck3a3I2tFlt           ->U.IO:$900120.13.1
PTR Ck3a3ShortFlt         ->U.IO:$900120.14.1
PTR Ck3a3OverLoadFlt      ->U.IO:$900120.15.1

// CH4、Gate3[0].Chan[3].AdcAmp[0]、ステータスセット 1 ============== //
PTR Ck3a4PwrFlt           ->U.IO:$9001A0.8.1
PTR Ck3a4SoftStartFlt     ->U.IO:$9001A0.9.1
PTR Ck3a4StoStatus        ->U.IO:$9001A0.10.1
PTR Ck3a4ShuntShortFlt    ->U.IO:$9001A0.11.1
PTR Ck3a4OverVoltFlt      ->U.IO:$9001A0.12.1
PTR Ck3a4I2tFlt           ->U.IO:$9001A0.13.1
PTR Ck3a4ShortFlt         ->U.IO:$9001A0.14.1
PTR Ck3a4OverLoadFlt      ->U.IO:$9001A0.15.1

// CH5、Gate3[1].Chan[0].AdcAmp[0]、ステータスセット 1 ============== //
PTR Ck3a5PwrFlt           ->U.IO:$904020.8.1
PTR Ck3a5SoftStartFlt     ->U.IO:$904020.9.1
PTR Ck3a5StoStatus        ->U.IO:$904020.10.1
PTR Ck3a5ShuntShortFlt    ->U.IO:$904020.11.1
PTR Ck3a5OverVoltFlt      ->U.IO:$904020.12.1
PTR Ck3a5I2tFlt           ->U.IO:$904020.13.1
PTR Ck3a5ShortFlt         ->U.IO:$904020.14.1
PTR Ck3a5OverLoadFlt      ->U.IO:$904020.15.1

// CH6、Gate3[1].Chan[1].AdcAmp[0]、ステータスセット 1 ============== //
PTR Ck3a6PwrFlt           ->U.IO:$9040A0.8.1
PTR Ck3a6SoftStartFlt     ->U.IO:$9040A0.9.1
PTR Ck3a6StoStatus        ->U.IO:$9040A0.10.1
PTR Ck3a6ShuntShortFlt    ->U.IO:$9040A0.11.1
PTR Ck3a6OverVoltFlt      ->U.IO:$9040A0.12.1
PTR Ck3a6I2tFlt           ->U.IO:$9040A0.13.1
PTR Ck3a6ShortFlt         ->U.IO:$9040A0.14.1
PTR Ck3a6OverLoadFlt      ->U.IO:$9040A0.15.1

// CH7、Gate3[1].Chan[2].AdcAmp[0]、ステータスセット 1 ============== //
PTR Ck3a7PwrFlt           ->U.IO:$904120.8.1
PTR Ck3a7SoftStartFlt     ->U.IO:$904120.9.1
PTR Ck3a7StoStatus        ->U.IO:$904120.10.1
PTR Ck3a7ShuntShortFlt    ->U.IO:$904120.11.1
PTR Ck3a7OverVoltFlt      ->U.IO:$904120.12.1
PTR Ck3a7I2tFlt           ->U.IO:$904120.13.1
PTR Ck3a7ShortFlt         ->U.IO:$904120.14.1
PTR Ck3a7OverLoadFlt      ->U.IO:$904120.15.1

// CH8、Gate3[1].Chan[3].AdcAmp[0]、ステータスセット 1 ============== //
PTR Ck3a8PwrFlt           ->U.IO:$9041A0.8.1
PTR Ck3a8SoftStartFlt     ->U.IO:$9041A0.9.1
PTR Ck3a8StoStatus        ->U.IO:$9041A0.10.1
PTR Ck3a8ShuntShortFlt    ->U.IO:$9041A0.11.1
PTR Ck3a8OverVoltFlt      ->U.IO:$9041A0.12.1
PTR Ck3a8I2tFlt           ->U.IO:$9041A0.13.1
PTR Ck3a8ShortFlt         ->U.IO:$9041A0.14.1
PTR Ck3a8OverLoadFlt      ->U.IO:$9041A0.15.1
```

```
// CH1、SYS.UDATA[256000] ステータスセット 2 ======================== //
PTR Ck3a1PwrStatus        ->U.USER:$FA000.0.1
PTR Ck3a1OverTempFlt      ->U.USER:$FA000.2.1
PTR Ck3a1ShuntOverLFlt    ->U.USER:$FA000.3.1
PTR Ck3a1PwmFreqFlt       ->U.USER:$FA000.4.1
PTR Ck3a1PwmCmdFlt        ->U.USER:$FA000.5.1
PTR Ck3a1AmpEna           ->U.USER:$FA000.6.1

// CH2、SYS.UDATA[256001] ステータスセット 2 ======================== //
PTR Ck3a2PwrStatus        ->U.USER:$FA004.0.1
PTR Ck3a2OverTempFlt      ->U.USER:$FA004.2.1
PTR Ck3a2ShuntOverLFlt    ->U.USER:$FA004.3.1
PTR Ck3a2PwmFreqFlt       ->U.USER:$FA004.4.1
PTR Ck3a2PwmCmdFlt        ->U.USER:$FA004.5.1
PTR Ck3a2AmpEna           ->U.USER:$FA004.6.1

// CH3、SYS.UDATA[256002] ステータスセット 2 ======================== //
PTR Ck3a3PwrStatus        ->U.USER:$FA008.0.1
PTR Ck3a3OverTempFlt      ->U.USER:$FA008.2.1
PTR Ck3a3ShuntOverLFlt    ->U.USER:$FA008.3.1
PTR Ck3a3PwmFreqFlt       ->U.USER:$FA008.4.1
PTR Ck3a3PwmCmdFlt        ->U.USER:$FA008.5.1
PTR Ck3a3AmpEna           ->U.USER:$FA008.6.1

// CH4、SYS.UDATA[256003] ステータスセット 2 ======================== //
PTR Ck3a4PwrStatus        ->U.USER:$FA00C.0.1
PTR Ck3a4OverTempFlt      ->U.USER:$FA00C.2.1
PTR Ck3a4ShuntOverLFlt    ->U.USER:$FA00C.3.1
PTR Ck3a4PwmFreqFlt       ->U.USER:$FA00C.4.1
PTR Ck3a4PwmCmdFlt        ->U.USER:$FA00C.5.1
PTR Ck3a4AmpEna           ->U.USER:$FA00C.6.1

// CH5、SYS.UDATA[256004] ステータスセット 2 ======================== //
PTR Ck3a5PwrStatus        ->U.USER:$FA010.0.1
PTR Ck3a5OverTempFlt      ->U.USER:$FA010.2.1
PTR Ck3a5ShuntOverLFlt    ->U.USER:$FA010.3.1
PTR Ck3a5PwmFreqFlt       ->U.USER:$FA010.4.1
PTR Ck3a5PwmCmdFlt        ->U.USER:$FA010.5.1
PTR Ck3a5AmpEna           ->U.USER:$FA010.6.1

// CH6、SYS.UDATA[256005] ステータスセット 2 ======================== //
PTR Ck3a6PwrStatus        ->U.USER:$FA014.0.1
PTR Ck3a6OverTempFlt      ->U.USER:$FA014.2.1
PTR Ck3a6ShuntOverLFlt    ->U.USER:$FA014.3.1
PTR Ck3a6PwmFreqFlt       ->U.USER:$FA014.4.1
PTR Ck3a6PwmCmdFlt        ->U.USER:$FA014.5.1
PTR Ck3a6AmpEna           ->U.USER:$FA014.6.1

// CH7、SYS.UDATA[256006] ステータスセット 2 ======================== //
PTR Ck3a7PwrStatus        ->U.USER:$FA018.0.1
PTR Ck3a7OverTempFlt      ->U.USER:$FA018.2.1
PTR Ck3a7ShuntOverLFlt    ->U.USER:$FA018.3.1
PTR Ck3a7PwmFreqFlt       ->U.USER:$FA018.4.1
PTR Ck3a7PwmCmdFlt        ->U.USER:$FA018.5.1
PTR Ck3a7AmpEna           ->U.USER:$FA018.6.1

// CH8、SYS.UDATA[256007] ステータスセット 2 ======================== //
PTR Ck3a8PwrStatus        ->U.USER:$FA01C.0.1
PTR Ck3a8OverTempFlt      ->U.USER:$FA01C.2.1
PTR Ck3a8ShuntOverLFlt    ->U.USER:$FA01C.3.1
PTR Ck3a8PwmFreqFlt       ->U.USER:$FA01C.4.1
PTR Ck3a8PwmCmdFlt        ->U.USER:$FA01C.5.1
PTR Ck3a8AmpEna           ->U.USER:$FA01C.6.1
```

```
// CH1、SYS.UDATA[256040] ステータスセット 3 ======================== //
PTR Ck3a1I2tint           ->U.USER:$FA0A0.2.1
PTR Ck3a1AdcOffsetFlt     ->U.USER:$FA0A0.4.1
PTR Ck3a1Ready            ->U.USER:$FA0A0.6.1

// CH2、SYS.UDATA[256041] ステータスセット 3 ======================== //
PTR Ck3a2I2tint           ->U.USER:$FA0A4.2.1
PTR Ck3a2AdcOffsetFlt     ->U.USER:$FA0A4.4.1
PTR Ck3a2Ready            ->U.USER:$FA0A4.6.1

// CH3、SYS.UDATA[256042] ステータスセット 3 ======================== //
PTR Ck3a3I2tint           ->U.USER:$FA0A8.2.1
PTR Ck3a3AdcOffsetFlt     ->U.USER:$FA0A8.4.1
PTR Ck3a3Ready            ->U.USER:$FA0A8.6.1

// CH4、SYS.UDATA[256043] ステータスセット 3 ======================== //
PTR Ck3a4I2tint           ->U.USER:$FA0AC.2.1
PTR Ck3a4AdcOffsetFlt     ->U.USER:$FA0AC.4.1
PTR Ck3a4Ready            ->U.USER:$FA0AC.6.1

// CH5、SYS.UDATA[256044] ステータスセット 3 ======================== //
PTR Ck3a5I2tint           ->U.USER:$FA0B0.2.1
PTR Ck3a5AdcOffsetFlt     ->U.USER:$FA0B0.4.1
PTR Ck3a5Ready            ->U.USER:$FA0B0.6.1

// CH6、SYS.UDATA[256045] ステータスセット 3 ======================== //
PTR Ck3a6I2tint           ->U.USER:$FA0B4.2.1
PTR Ck3a6AdcOffsetFlt     ->U.USER:$FA0B4.4.1
PTR Ck3a6Ready            ->U.USER:$FA0B4.6.1

// CH7、SYS.UDATA[256046] ステータスセット 3 ======================== //
PTR Ck3a7I2tint           ->U.USER:$FA0B8.2.1
PTR Ck3a7AdcOffsetFlt     ->U.USER:$FA0B8.4.1
PTR Ck3a7Ready            ->U.USER:$FA0B8.6.1

// CH8、SYS.UDATA[256047] ステータスセット 3 ======================== //
PTR Ck3a8I2tint           ->U.USER:$FA0BC.2.1
PTR Ck3a8AdcOffsetFlt     ->U.USER:$FA0BC.4.1
PTR Ck3a8Ready            ->U.USER:$FA0BC.6.1
```

5-8-7 CK3A サンプル PLC 8 モータ

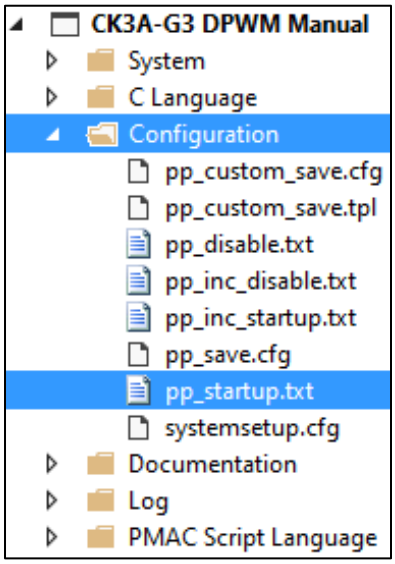
```
OPEN PLC Ck3aPLC
// CH1～8 ステータスビットセット 2
Gate30WpDis Gate30DataCtrl = 0
Gate31WpDis Gate31DataCtrl = 0
CALL DelayTimer.msec(1)
Sys.Udata[256000] = Ck3a1AdcBData
Sys.Udata[256001] = Ck3a2AdcBData
Sys.Udata[256002] = Ck3a3AdcBData
Sys.Udata[256003] = Ck3a4AdcBData
Sys.Udata[256004] = Ck3a5AdcBData
Sys.Udata[256005] = Ck3a6AdcBData
Sys.Udata[256006] = Ck3a7AdcBData
Sys.Udata[256007] = Ck3a8AdcBData
CALL DelayTimer.msec(1)

// CH1～8 ステータスビットセット 3
Gate30WpDis Gate30DataCtrl = 1
Gate31WpDis Gate31DataCtrl = 1
CALL DelayTimer.msec(1)
Sys.Udata[256040] = Ck3a1AdcBData
Sys.Udata[256041] = Ck3a2AdcBData
Sys.Udata[256042] = Ck3a3AdcBData
Sys.Udata[256043] = Ck3a4AdcBData
Sys.Udata[256044] = Ck3a5AdcBData
Sys.Udata[256045] = Ck3a6AdcBData
Sys.Udata[256046] = Ck3a7AdcBData
Sys.Udata[256047] = Ck3a8AdcBData
CALL DelayTimer.msec(1)

// CH1～8 DC バス電圧
Gate30WpDis Gate30DataCtrl = 2
Gate31WpDis Gate31DataCtrl = 2
CALL DelayTimer.msec(1)
IF(Ck3a1Ready == 0) Ck3a1DcBus = Ck3a1AdcBData * 4
IF(Ck3a2Ready == 0) Ck3a2DcBus = Ck3a2AdcBData * 4
IF(Ck3a3Ready == 0) Ck3a3DcBus = Ck3a3AdcBData * 4
IF(Ck3a4Ready == 0) Ck3a4DcBus = Ck3a4AdcBData * 4
IF(Ck3a5Ready == 0) Ck3a5DcBus = Ck3a5AdcBData * 4
IF(Ck3a6Ready == 0) Ck3a6DcBus = Ck3a6AdcBData * 4
IF(Ck3a7Ready == 0) Ck3a7DcBus = Ck3a7AdcBData * 4
IF(Ck3a8Ready == 0) Ck3a8DcBus = Ck3a8AdcBData * 4
CALL DelayTimer.msec(1)

// CH1～8 温度
Gate30WpDis Gate30DataCtrl = 3
Gate31WpDis Gate31DataCtrl = 3
CALL DelayTimer.msec(1)
IF(Ck3a1Ready == 0) Ck3a1Temp = Ck3a1AdcBData
IF(Ck3a2Ready == 0) Ck3a2Temp = Ck3a2AdcBData
IF(Ck3a3Ready == 0) Ck3a3Temp = Ck3a3AdcBData
IF(Ck3a4Ready == 0) Ck3a4Temp = Ck3a4AdcBData
IF(Ck3a5Ready == 0) Ck3a5Temp = Ck3a5AdcBData
IF(Ck3a6Ready == 0) Ck3a6Temp = Ck3a6AdcBData
IF(Ck3a7Ready == 0) Ck3a7Temp = Ck3a7AdcBData
IF(Ck3a8Ready == 0) Ck3a8Temp = Ck3a8AdcBData
CALL DelayTimer.msec(1)
```

```
// CH1～8 ファームウェアバージョン
Gate30WpDis Gate30DataCtrl = 6
Gate31WpDis Gate31DataCtrl = 6
CALL DelayTimer.msec(1)
IF(Ck3a1Ready == 0) Ck3a1FwVer = Ck3a1AdcBData
IF(Ck3a2Ready == 0) Ck3a2FwVer = Ck3a2AdcBData
IF(Ck3a3Ready == 0) Ck3a3FwVer = Ck3a3AdcBData
IF(Ck3a4Ready == 0) Ck3a4FwVer = Ck3a4AdcBData
IF(Ck3a5Ready == 0) Ck3a5FwVer = Ck3a5AdcBData
IF(Ck3a6Ready == 0) Ck3a6FwVer = Ck3a6AdcBData
IF(Ck3a7Ready == 0) Ck3a7FwVer = Ck3a7AdcBData
IF(Ck3a8Ready == 0) Ck3a8FwVer = Ck3a8AdcBData
CALL DelayTimer.msec(1)

// CH1～8 電流定格
Gate30WpDis Gate30DataCtrl = 7
Gate31WpDis Gate31DataCtrl = 7
CALL DelayTimer.msec(1)
IF(Ck3a1Ready == 0) Ck3a1CurRating = Ck3a1AdcBData
IF(Ck3a2Ready == 0) Ck3a2CurRating = Ck3a2AdcBData
IF(Ck3a3Ready == 0) Ck3a3CurRating = Ck3a3AdcBData
IF(Ck3a4Ready == 0) Ck3a4CurRating = Ck3a4AdcBData
IF(Ck3a5Ready == 0) Ck3a5CurRating = Ck3a5AdcBData
IF(Ck3a6Ready == 0) Ck3a6CurRating = Ck3a6AdcBData
IF(Ck3a7Ready == 0) Ck3a7CurRating = Ck3a7AdcBData
IF(Ck3a8Ready == 0) Ck3a8CurRating = Ck3a8AdcBData
CALL DelayTimer.msec(1)

// GATE3[0]、CH1～4 バス放電制御
IF(Ck3a0BusDisCtrl == 1 && Ck3a0BusDisState == 0) Gate30WpDis Ck3a0BusDisState = 1
IF(Ck3a0BusDisCtrl == 0 && Ck3a0BusDisState == 1) Gate30WpDis Ck3a0BusDisState = 0
// GATE3[0]、CH1～4 ファン制御
IF(Ck3a0FanCtrl == 1 && Ck3a0FanState == 0) Gate30WpDis Ck3a0FanState = 1
IF(Ck3a0FanCtrl == 0 && Ck3a0FanState == 1) Gate30WpDis Ck3a0FanState = 0
// GATE3[0]、CH1～4 ダイナミックブレーキ制御
IF(Ck3a0DbCtrl == 1 && Ck3a0DbState == 1) Gate30WpDis Ck3a0DbState = 0
IF(Ck3a0DbCtrl == 0 && Ck3a0DbState == 0) Gate30WpDis Ck3a0DbState = 1

// GATE3[1]、CH5～8 バス放電制御
IF(Ck3a1BusDisCtrl == 1 && Ck3a1BusDisState == 0) Gate31WpDis Ck3a1BusDisState = 1
IF(Ck3a1BusDisCtrl == 0 && Ck3a1BusDisState == 1) Gate31WpDis Ck3a1BusDisState = 0
// GATE3[1]、CH5～8 ファン制御
IF(Ck3a1FanCtrl == 1 && Ck3a1FanState == 0) Gate31WpDis Ck3a1FanState = 1
IF(Ck3a1FanCtrl == 0 && Ck3a1FanState == 1) Gate31WpDis Ck3a1FanState = 0
// GATE3[1]、CH5～8 ダイナミックブレーキ制御
IF(Ck3a1DbCtrl == 1 && Ck3a1DbState == 1) Gate31WpDis Ck3a1DbState = 0
IF(Ck3a1DbCtrl == 0 && Ck3a1DbState == 0) Gate31WpDis Ck3a1DbState = 1

// GATE3[0]、CH1～4 異常解除
IF(Ck3a0Clrf == 1)
{
 Gate30WpDis Ck3a0ClrfBit = 1
 CALL DelayTimer.msec(10)
 Gate30WpDis Ck3a0ClrfBit = 0
 Ck3a0Clrf = 0
}
```

```
// GATE3[1]、CH5～8 異常解除
IF(Ck3a1Clrf == 1)
{
 Gate31WpDis Ck3a1ClrfBit = 1
 CALL DelayTimer.msec(10)
 Gate31WpDis Ck3a1ClrfBit = 0
 Ck3a1Clrf = 0
}

// CH1～8 ロジック電源 OFF?
IF(Ck3a1Ready == 1) Ck3a1DcBus,4 = 0
IF(Ck3a2Ready == 1) Ck3a2DcBus,4 = 0
IF(Ck3a3Ready == 1) Ck3a3DcBus,4 = 0
IF(Ck3a4Ready == 1) Ck3a4DcBus,4 = 0
IF(Ck3a5Ready == 1) Ck3a5DcBus,4 = 0
IF(Ck3a6Ready == 1) Ck3a6DcBus,4 = 0
IF(Ck3a7Ready == 1) Ck3a7DcBus,4 = 0
IF(Ck3a8Ready == 1) Ck3a8DcBus,4 = 0
CLOSE
```

5-8-8 電源投入時に CK3A PLC を有効にする

電源投入時に CK3A PLC を有効にするには、IDE プロジェクトの Configuration フォルダの下にある startup.txt ファイルを使用します。



参考

ENABLE PLC コマンドは、ファイルではなく、PLC の名前を参照します。

5-8-9 CK3A PLC の使用

機能制御

推奨するアンプ機能の構造は、**Ck3a{i}{機能名}**のようになります。

- {i}は Gate3 インデックスです
- {機能名}は希望する機能です

下表は、可能なコマンドの要約です。

使用上の注意

機能の変更によって生じる希望する ADC ストローブワードを後続の電源サイクルで適用するためには、更新してメインプロジェクトファイルまたは構成に保存する必要があります。

CH	異常解除	ダイナミックブレーキ制御	ファン制御	バス放電制御
Gate3[**0**] CH1～4	Ck3a**0**Clrf=1	ON の場合は Ck3a**0**DbCtrl=1 OFF の場合は Ck3a**0**DbCtrl=0	ON Ck3a**0**FanCtrl=1 自動 Ck3a**0**FanCtrl=0	ON Ck3a**0**BusDisCtrl=1 OFF Ck3a**0**BusDisCtrl=0
Gate3[**1**] CH5～8	Ck3a**1**Clrf=1	ON の場合は Ck3a**1**DbCtrl=1 OFF の場合は Ck3a**1**DbCtrl=0	ON Ck3a**1**FanCtrl=1 自動 Ck3a**1**FanCtrl=0	ON Ck3a**1**BusDisCtrl=1 OFF Ck3a**1**BusDisCtrl=0

参考

Ck3a[i]Clrf パラメータは、PLC によって自動的に 0 に戻されます。

使用上の注意

アンプが有効になっているときは、ダイナミックブレーキ設定の ON/OFF を切り替えないでください。

データの読み取り

推奨するアンプデータ構造は、**Ck3a{x}{データ名}**のようになります。

- {x}はモータ/アンプ番号です
- {データ名}は読み取りたいデータです

これらの変数は、モニタリング用の IDE ウォッチウィンドウに配置したり、PLC プログラムやオペレータインタフェース表示で使用することができます。読み取り例は以下のとおりです。

チャネル 1

ウォッチ

コマンド/クエリ	応答
Ck3a1DcBus	294
Ck3a1Temp	25
Ck3a1FwVer	60
Ck3a1CurRating	5

ウォッチ

コマンド/クエリ	応答
Ck3a1DcBus,4	294 23 60 5

チャネル 1～4

ウォッチ

コマンド/クエリ	応答
Ck3a1DcBus,4,4	294 298 300 294
Ck3a1Temp,4,4	23 28 25 24
Ck3a1FwVer,4,4	60 60 60 60
Ck3a1CurRating,4,4	5 5 5 10

ウォッチ

コマンド/クエリ	応答
Ck3a1DcBus,4	294 23 60 5
Ck3a2DcBus,4	298 28 60 5
Ck3a3DcBus,4	300 25 60 5
Ck3a4DcBus,4	294 24 60 10

参考

- **{variable},{n}**構文は、次の n 個の連続する変数を報告します。
- **{variable},{n},{m}**構文は、m 個の変数で区切られた次の n 個の変数を報告します。

ステータスビットの読み取り

推奨するアンプステータス構造は、**Ck3a {x} {status name}**のようになります。

- {x}はモータ/アンプ番号です
- {ステータス名}は読み取りたいステータスです

これらの変数は、モニタリング用の IDE ウォッチウィンドウに配置したり、PLC プログラムやオペレータインタフェース表示で使用することができます。読み取り例は以下のとおりです。

チャネル 1

ウォッチ

コマンド/クエリ	応答	
Ck3a1PwrFlt	0	ステータスセット1
Ck3a1SoftStartFlt	0	
Ck3a1StoStatus	0	
Ck3a1ShuntShortFlt	0	
Ck3a1OverVoltFlt	0	
Ck3a1I2tFlt	0	
Ck3a1ShortFlt	0	
Ck3a1OverLoadFlt	0	
Ck3a1PwrStatus	1	ステータスセット2
Ck3a1OverTempFlt	0	
Ck3a1ShuntOverLFlt	0	
Ck3a1PwmFreqFlt	0	
Ck3a1PwmCmdFlt	0	
Ck3a1AmpEna	1	
Ck3a1I2tint	0	ステータスセット3
Ck3a1AdcOffsetFlt	0	
Ck3a1Ready	0	

ウォッチ

コマンド/クエリ	応答	
Ck3a1PwrFlt,8	0 0 0 0 0 0 0 0	セット 1
Ck3a1PwrStatus,6	1 0 0 0 0 1	セット 2
Ck3a1I2tint,3	0 0 0	セット 3

チャネル 1～4

ウォッチ

コマンド/クエリ	応答	
Ck3a1PwrFlt,8	0 0 0 0 0 0 0 0	CH1
Ck3a1PwrStatus,6	1 0 0 0 0 1	
Ck3a1I2tint,3	0 0 0	
Ck3a2PwrFlt,8	0 0 0 0 0 0 0 0	CH2
Ck3a2PwrStatus,6	1 0 0 0 0 1	
Ck3a2I2tint,3	0 0 0	
Ck3a3PwrFlt,8	0 0 0 0 0 0 0 0	CH3
Ck3a3PwrStatus,6	1 0 0 0 0 1	
Ck3a3I2tint,3	0 0 0	
Ck3a4PwrFlt,8	0 0 0 0 0 0 0 0	CH4
Ck3a4PwrStatus,6	1 0 0 0 0 1	
Ck3a4I2tint,3	0 0 0	

参考

{variable},{n}構文は、次の n 個の連続する変数を報告します。

6

異常の確認と処置

本章では、CK3A シリーズ DirectPWM アンプの LED 表示、エラー表示コード、および対処方法について説明します。

6-1 LED 表示 6-2

6-2 表示コード 6-3

6-2-1 正常モード動作 6-3
6-2-2 エラーコード 6-3
6-2-3 エラーコードの確認と処置 6-6
6-2-4 異常しきい値 6-10
6-2-5 アンプ動作処置 6-11

6-1 LED 表示

下記の LED は、アンプの前面にあります。以下は、LED の色指定一覧と説明です。

名称	LED 名称	状態	表示色	内容
ロジック電源	PWR	点灯	緑	24V ロジック電源がかかっている、かつ範囲内
	PWR	消灯	―	24V ロジック電源がかかっていない、または範囲外 （アンプ OFF）
アンプ有効	ENA	点灯	緑	アンプが有効 （例：PMAC コマンドアウトまたは PMAC コマンドジョグ、サーボ ON）
	ENA	消灯	―	アンプが無効 （PMAC コマンドの終了、サーボ OFF、またはアンプ OFF）
STO	STO	点灯	緑	パワードライバ通電、正常動作 （モータサーボ制御できる）
	STO	点灯	赤	パワードライバ通電なし、STO がトリガされ、または無効になっていない。 （モータサーボ制御できない）
シャント	SHU	点灯	緑	シャント動作アクティブ（エネルギー追跡またはブレーキ） （通常、高負荷減速中に一瞬点滅）
	SHU	消灯	―	シャント動作非アクティブ （正常動作、問題なし）
バス	BUS	点灯	赤	バスリンク充電 （主回路電源投入、コンデンサ充電）
	BUS	薄暗い	赤	バスリンクエネルギー消費中 （主回路電源の取り外し）
	BUS	非常に薄暗い	―	放電終了、ロジック電源 ON（常に約 20V 残留） またはアンプ OFF が約 5 分未満
	BUS	消灯	―	アンプ OFF が約 5 分以上

参考

- BUS LED の強度は、DC リンクの電圧レベルに正比例します。
- 形 CK3A-G305L または 形 CK3A-G310L でロジック電源入力が DC22.0V 未満または DC26.4V を超える場合、アンプのロジック電源はオフになります。

6-2　表示コード

ここでは、デュアル 7 セグメント表示のエラーコードについて説明します。

6-2-1　正常モード動作

正常モード動作では、アンプは以下のコードを表示します。

表示	異常	内容
	異常なし	アンプ OFF
-	異常なし	アンプが無効 （モータ停止、サーボ OFF）
0.	異常なし	アンプが有効 （PMAC コマンドアウトまたはジョグのように閉ループ、サーボ ON）
L-	異常なし	低電圧モード設定（形 CK3A-G305L*1 または形 CK3A-G320L のストローブワード設定） アンプが無効（モータ停止、サーボ OFF）
	警告	低電圧モードに設定されておらず、電圧が DC80～100V に低下した場合（アンプは無効ではない）
L0.	異常なし	低電圧モード設定（形 CK3A-G305L*1 または形 CK3A-G320L のストローブワード設定） アンプが有効（PMAC コマンドアウトまたはジョグのように閉ループ、サーボ ON）

*1. CK3A-G305L の CN1 にも LV 短絡線を取り付ける必要があります。

6-2-2　エラーコード

異常が発生した場合、アンプはエラーコードを表示し、電源ドライバ回路を OFF にします。

表示	異常	内容	リセット
P1	**PWM 周波数**	検出された PWM 周波数が許容範囲外	異常解除
P2	**PWM コマンド**	PWM コマンドが長時間 ON のままであったため、IPMトランジスタが継続的に導通	異常解除
P3	**ADC クロック**	検出された ADC クロック周波数が許容範囲外 [2.450～6.250]MHz	自動リセット

表示	異常	内容	リセット
A1	**主電源回路**	主回路の電源入力(AC または DC)が存在しない、または取り外されている	自動リセット
A2	**ソフトスタート過負荷**	十分な時間がなく、主回路の電源再投入数超過	ロジック電源再投入
A3	**シャント短絡**	シャント回路で短絡が検出	異常解除
A4	**シャント過負荷**	シャントまたはブレーキの最大連続時間(2 秒)超過	異常解除
A5	**過電圧**	内部バス電圧がしきい値(DC403V)を超えた	異常解除
A6	**STO 入力**	STO 無効短絡線が取り付けされていない、または、STO 入力が取り外されている	有効/異常解除
A7	**過熱**	パワーモジュールの温度がしきい値(80℃)を超えた	異常解除
A8	**I2T**	200%公称出力での許容可能時間がしきい値(2 秒)を超えた (形 CK3A-G305L または形 CK3A-G310L) 300%公称出力での許容可能時間がしきい値(2 秒)を超えた (形 CK3A-G320L)	異常解除
A9	**ADC オフセット**	現在の ADC オフセット値が許容値(アンプファームウェア)より大きい	ロジック電源再投入
AC	**短絡**	最大パワーモジュール出力での許容可能時間がしきい値(3 マイクロ秒)を超えた	異常解除
AL	**過負荷**	アンプの最大指定出力での許容可能時間がしきい値(6 ミリ秒)を超えた	異常解除

使用上の注意

- ソフトスタート A2 異常が発生した後、5 分間冷却します。
- ソフトスタート(A2)、シャント短絡(A3)ADC オフセット(A9)、および短絡(AC)異常では、表示異常を解除するためにロジック電源再投入が必要です。
- 表示異常の解除は、異常状態が取り除いた場合だけ行われます。

参考

- 一部の表示異常は、異常状態が取り除くとすぐに自動的に解除されます。
- 一部の異常解除は、アンプを有効にするか、ADC ストローブワードを使用して異常解除で行われます。

6-2-3 エラーコードの確認と処置

表示	異常および異常の確認と処置
	アンプ OFF • ロジック電源と配線をチェックし、DC24V 入力を測定して、DC22.0～26.4V であることを確認してください
L-	**低電圧警告（AC 主電源入力の場合）** • 低電圧モードの場合、正常表示である • 主電源が AC で、電圧が DC80～100V に低下した場合、警告（アンプは無効ではない） • 主回路（AC）の電源と配線を確認してください
A1	**主電源異常** • 主回路（AC または DC）の電源と配線を確認してください • 低電圧モードが設定されている場合は、CN1 に低電圧短絡線が取り付けられていることを確認してください • 低電圧モードが設定されている場合は、ADC ストローブワードビット＃21（LV モード）が 1 に設定されていることを確認してください
A2	**ソフトスタート過負荷** • 機械運転における主回路の電源投入数を減らしてください • 放電モードが OFF の場合は、主電源再投入時間が最低 10 秒にしてください • 放電モードが ON の場合は、主電源再投入時間が最低 1 分にしてください • 異常が発生した後、5 分間冷却
A3	**シャント短絡** • 内部シャントを使用する場合は、B2 - B3 短絡（CN1）が正しい電線ゲージであり、正しく取り付けられていることを確認してください • 外部シャントを使用する場合は、CN1 への配線接続を確認してください • 外部シャント抵抗を測定し、無限ではないことを確認してください • 外部シャント線がパネル内の他の電線と短絡していないことを確認してください
A4	**シャント過負荷** • 過電圧 A5 異常が発生の場合、A4 も表示される（正常である） • 外部抵抗を使用する場合は、仕様が推奨シャント抵抗に準拠していることを確認してください • 推奨シャント抵抗を使用してください • シャント抵抗を測定し、無限ではないことを確認してください • モータの減速プロファイルを減らしてみてください
A5	**過電圧** • 内部シャントを使用する場合は、B2 - B3 短絡（CN1）が正しい電線ゲージであり、正しく取り付けられていることを確認してください • 外部シャントを使用する場合は、推奨抵抗を使用し、CN1 への接続を確認してください • シャント抵抗を測定し、意図／仕様より低い値ではないことを確認してください • IDE で報告された電圧を読み出し、予想範囲内にあることを確認してください • モータの減速プロファイルを減らしてみてください

表示	異常および異常の確認と処置
A6	**STO 入力** • STO異常は、ADC ストローブワードにより解除、またはアンプが有効(サーボ ON)になるまで、STO 異常表示 A6 がラッチされる • STO 入力を使用しない場合は、CN4.1 と CN4.2 間の STO 無効短絡線が取り付けされていることを確認してください • STO 入力を使用する場合は、CN4.2 がオープンしている(フローティング)ことを確認してください • STO 入力を使用する場合は、配線を確認し、CN4.1 で DC24V の測定を行ってください • CN4.1 に DC24V がない場合は、ロジック電源 ON になっていることを確認してください • STO 入力を使用する場合は、STO1 および STO2 入力で DC24V の測定を行ってください • CN4 コネクタが接続／切断されたときの STO LED の変化を確認してください
A7	**過熱** • 本マニュアルに記載されている動作環境は、仕様範囲内であることを確認してください • 本マニュアルに記載されているキャビネットと取付スペースは、仕様範囲内にあることを確認してください • ファンにほこりやごみが詰まっていないことを確認してください • ファンが作動していて、視覚的で、空気が吹いているのを感じているかを確認してください • IDE からファン ON／OFF を切り替える
A8	**I2T 統合電流異常** • モータの配線に緩みや断続的な短絡がないか確認してください • モータが常に硬い表面を押していないことを確認してください • モータにブレーキが付いている場合は、動作中にブレーキが解除されていることを確認してください • 形 CK3A-G305L を使用している場合は、より多くの電力を得るために形 CK3A-G310L または形 CK3A-G320L の使用を検討してください • 形 CK3A-G310L を使用している場合は、より多くの電力を得るために形 CK3A-G320L の使用を検討してください • モータの加速／減速プロファイルを減らしてみてください
A9	**ADC オフセット** • ADC ストローブワードを使用して異常を解除する • ロジック電源再投入
AC	**短絡** • アンプが有効(サーボ ON)の場合は、ダイナミックブレーキの ON／OFF 設定を切り替えないでください • モータの配線が正しく、適切な電線ゲージを使用していることを確認してください • モータのリード線が短絡していないこと、またはアースに接続されていないことを確認してください • モータのフェーズ間抵抗を測定し、フェーズ抵抗が一致していることを確認してください • モータの加速／減速プロファイルを減らしてみてください
AL	**過負荷** • ダイナミックブレーキ機能が ON の場合は、JOG/を発行する前に、モータを停止(サーボ OFF)しないでください。モータが停止する。 • ダイナミックブレーキ機能が ON の場合は、アンプを有効にしてから 50 ミリ秒以上待ってからモーション指令をしてください • ダイナミックブレーキ機能が ON の場合は、モータが停止(サーボ OFF)後、500 ミリ秒以上待ってからモータを再度サーボ制御する • モータの配線が適切な電線ゲージを使用して、正しく配線されていることを確認してください • モータシャーシが接地されており、潜在的な電圧がないことを確認してください • モータのフェーズ間抵抗を測定し、フェーズ抵抗が一致していることを確認してください • 主回路の接続が適切な電線ゲージを使用して、正しく配線されていることを確認してください

表示	異常および異常の確認と処置
P1	**PWM 周波数範囲外** • コントローラの PWM 周波数設定が範囲内にあることを確認してください • IDE ソフトウェアでアンプによって報告された PWM 周波数を読み出す • PWM 接続を確認し、コネクタがしっかりと接続されていることを確認してください • PWM ケーブルを別のサーボチャネルに接続してみてください（コントローラカードが不良かどうかを確認してください） • PWM ケーブルを交換してみてください • モータの配線に緩みや断続的な短絡がないか確認してください
P2	**PWM コマンド（飽和）ON 長すぎる** • コントローラの PWM 周波数設定が[8～20]kHz の範囲内にあることを確認してください • Motor[x].PwmSf が正しく設定されていることを確認してください • モータの加速プロファイルを減らしてみてください
P3	**ADC クロック範囲外** • コントローラの ADC クロック周波数設定が[2.450～6.250]MHz の範囲内にあることを確認してください • PWM ケーブルと（緩い）接続を確認してください • PWM ケーブルを別のサーボチャネルに接続してみてください（コントローラカードが不良かどうかを確認してください） • PWM ケーブルを交換してみてください

参考

上記すべてのトラブルシューティング手順を行っても異常状態が発生する場合は、アンプを交換してください。

安全上の要点

CK3A DirectPWM アンプは、単相または三相 AC 主電源で交換可能に動作するように設計されているため、三相動作で位相が失われた場合でも異常が発生しません。

6-2-4 異常しきい値

項目		異常	しきい値
過電圧（A5）		A5	DC403V
シャント ON		―	DC390V
シャント OFF		―	DC370V
電圧不足警告		L-	AC 主電源と電圧が DC80～100V 間の場合
低電圧（A1）–正常 AC モード		A1	DC80V
低電圧（A1）–低電圧モード		A1	DC43V
過熱トリップ（A7）		A7	80 ℃
過熱放出		―	70 ℃
自動ファン ON		―	35 ℃
自動ファン OFF		―	32 ℃
過負荷（AL）	形 CK3A-G305L	AL	> 6msec@15.55A_{peak}
	形 CK3A-G310L		> 6msec@28.85A_{peak}
	形 CK3A-G320L		> 6msec@90.91A_{peak}
I2T 統合開始	形 CK3A-G305L	―	5.1A_{rms}
	形 CK3A-G310L		10.1A_{rms}
	形 CK3A-G320L		20.1A_{rms}
I2T トリップ時間	形 CK3A-G305L	A8	> 2sec@10A_{rms}
	形 CK3A-G310L		> 2sec@20A_{rms}
	形 CK3A-G320L		> 2sec@60A_{rms}

6-2-5 アンプ動作処置

問題	考えられる根本原因	確認と対策
7 セグメント表示異常	• 異常状態が発生 • アンプの損傷	• 「エラーコードの確認と処置」を参照し、異常状態を取り除き、表示異常を解除してください
STO LED が点灯しない	• アンプの損傷	
BUS LED が点灯しない	• 主回路電源 OFF • アンプの損傷／LED の焼け	• A1 エラーコードのトラブルシューティングを参照してください
アンプが ON しない（7 セグメント表示が空白、PWR LED が点灯しない）	• ロジック電源が供給されていない • ロジック電源範囲外 • 電線ゲージが細すぎる • 配線 • アンプの損傷	• 電源は仕様範囲内 • 電源は作動している • 配線は正しい電線ゲージを使用している • 配線が正しい
ENA LED が点灯しない	• 無効（サーボ OFF） • コントローラの異常 • PWM ケーブルが不良 • 異常状態が発生 • アンプの損傷／LED の焼け	• アンプを有効にする（サーボ ON） • Motor[x].AmpEna が PMAC で必ず1にしてください • PWM ケーブルを交換してください • 「エラーコードの確認と処置」を参照し、異常状態を取り除き、表示異常を解除してください
バスは放電モードで消費しない	• 内部シャント抵抗器の短絡線が取り付けられていない • 外部シャント抵抗が取り付けられていない • 使用している外部シャント抵抗が仕様の範囲外 • 放電 ON モードの ADC ストローブワード設定が行っていない • 主電源の取り外し後、ロジック電源の取り外しが速すぎる	• 内部シャント用の CN1 短絡線が適切なゲージを使用して、正しく配線されていることを確認してください • 外部シャント抵抗が取り付けられ、正しく配線されていることを確認してください • 推奨のオムロン製外部シャント、または正しい仕様のシャントを使用してください • ADC ストローブワードを正しく放電 ON に設定して、ADC レジスタのステータスビットを確認してください • 主電源を取り外した後、5秒以上待ってからロジック電源を取り外してください

参考

アンプ ON の場合、STO LED は常に赤点灯または緑点灯にする必要があります。

A

付録

本章には、CK3A DirectPWM アンプに関する付録が含まれています。

A

A-1 IDE を使用して自動モータの設定 A-3

A-1-1 IDE システム設定 A-3

A-2 Power PMAC3 手動モータの設定 A-12

A-2-1 IDE プロジェクトの作成 A-12
A-2-2 基本的な最適化設定、システム設定およびGate設定 A-16
A-2-3 エンコーダの進行中位置の設定と検証 A-17
A-2-4 モータの設定と試運転 A-22
A-2-5 モータ動作方向の反転 A-46

A-3 Gate3「スクリプト」モータ設定サンプル A-47

A-3-1 パルスエンコーダ付きロータリサーボモータ A-48
A-3-2 オムロン G5 サーボモータ A-50
A-3-3 オムロン 1S サーボモータ（ACC-84E） A-53
A-3-4 正弦波エンコーダ付きのリニアサーボモータ A-54
A-3-5 BiSS エンコーダ付きのリニアサーボモータ（ACC-84E） A-57

A-4 Gate1「スクリプト」モータ設定サンプル A-59

A-4-1 Gate1(ACC-24E2)ありの Power PMAC A-60
A-4-2 Gate1（ACC-24E2）ありの Turbo PMAC A-62

A-5 Gate1 用 CK3A サンプル PLC A-64

A-5-1 Gate1 ありの Power PMAC A-64
A-5-2 Gate1 ありの Turbo PMAC A-67

A-6　STO 安全上の注意 A-72
A-6-1 危険分析 A-72
A-6-2 STO 仕様 A-75
A-6-3 STO の動作原理 A-76

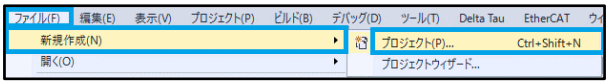
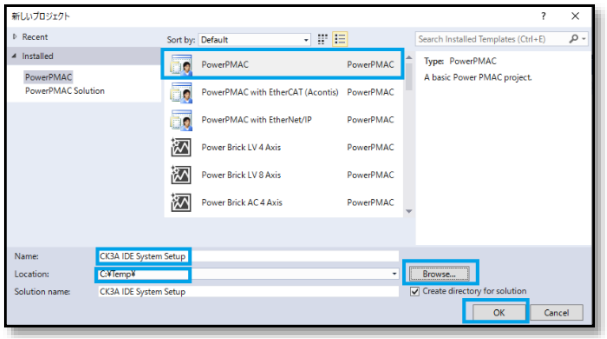
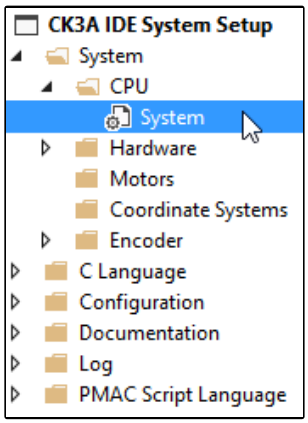
A-1　IDE を使用して自動モータの設定

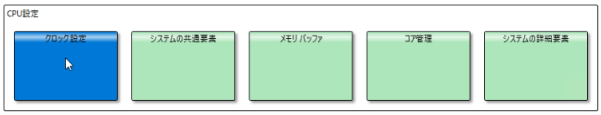
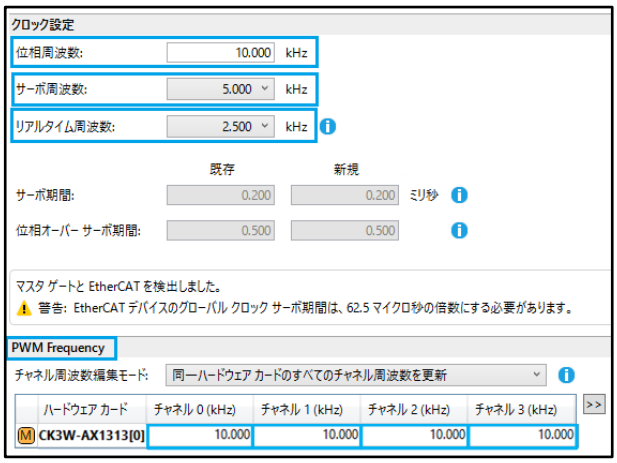
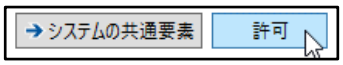
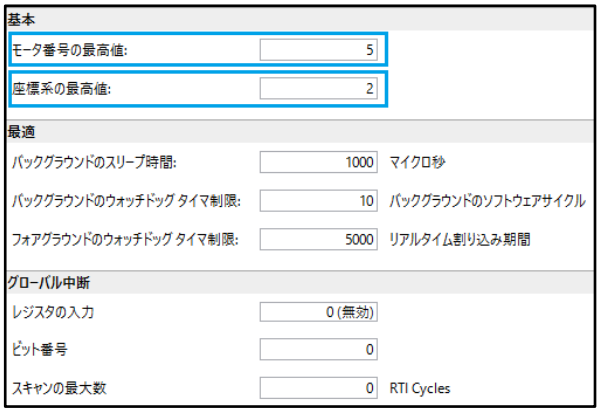
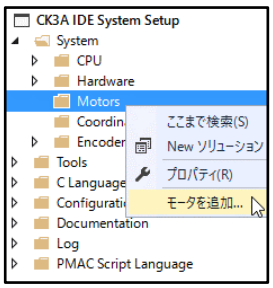
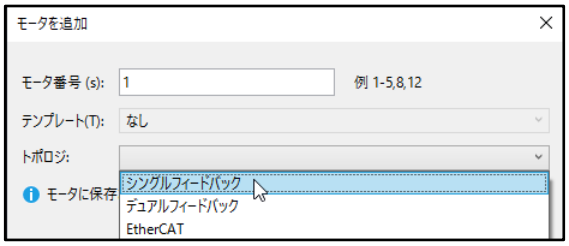
ここでは、IDE システム設定ツールを使用して、CK3A DirectPWM アンプでブラシレスサーボモータを設定する手順について説明します。

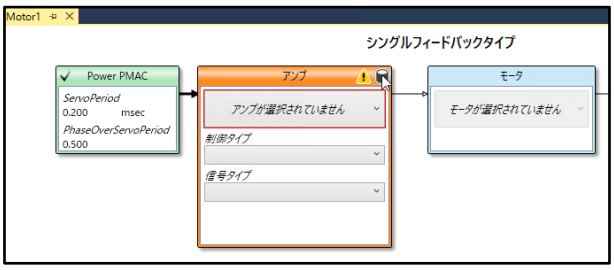
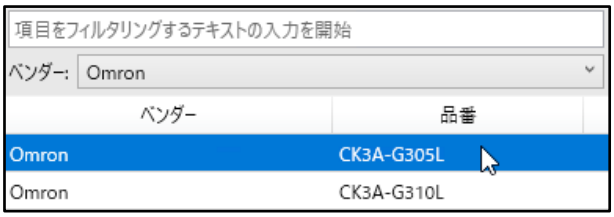
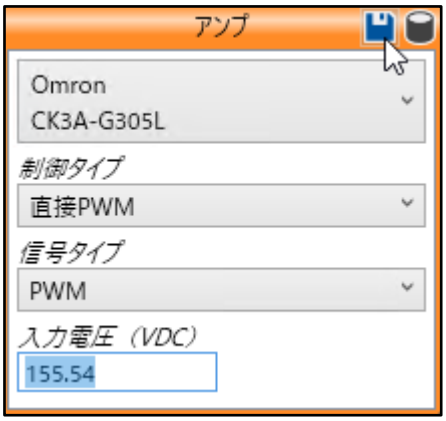
構成内容

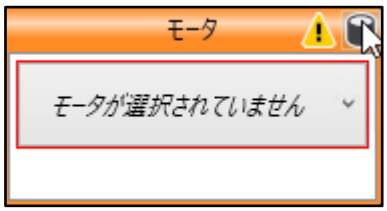
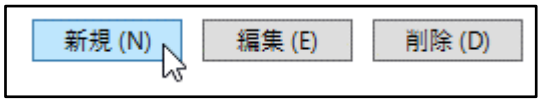
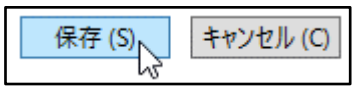
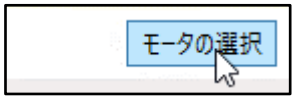
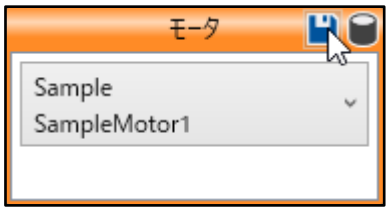
項目	内容
コントローラ	この例は、CK3M AX または UMAC ACC-24E3 に適用
アンプ	形 CK3A-G305L（AC110V 供給電圧）
モータ	ロータリサーボモータ、AC200V、2.4／10.2A、5 極ペア
エンコーダ	20 ビットシングルターンアブソリュート（Panasonic プロトコル）

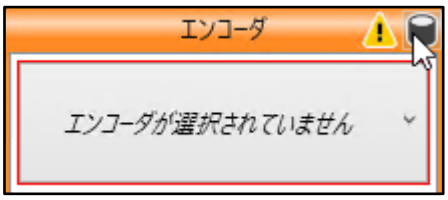
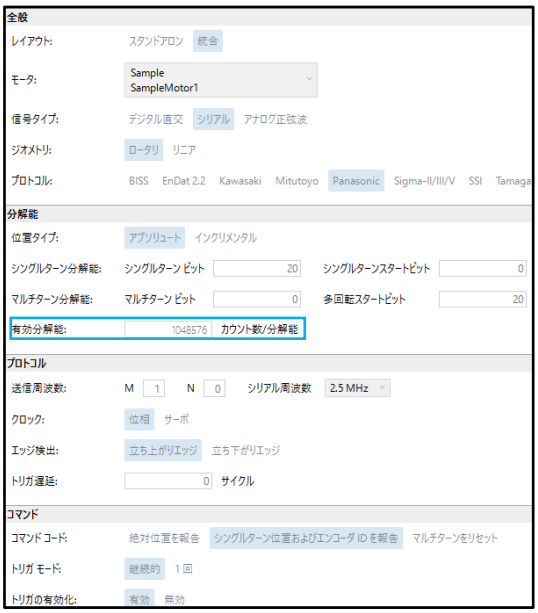
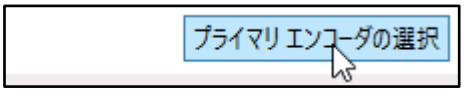
A-1-1　IDE システム設定

手順	説明図
新規 IDE プロジェクトを作成します。 • ［ファイル］－［新規作成］－［プロジェクト］を選択します。 • テンプレートを選択します。 • プロジェクト名の割り当てを行います。 • パソコンのフォルダ場所を選択します。	 
前述の手順では、IDE ソリューションエクスプローラウィンドウにプロジェクトツリーを作成します。 • ［システム］をダブルクリックします。	

手順	説明図
CPU(システムおよびグローバル)設定 • [クロック設定]をクリックします。	
• クロック設定 • フェーズ周波数を設定します。 • サーボ周波数を設定します。 • リアルタイム周波数を設定します。 • PWM 周波数を設定します。 CK3A チャネルの場合は、[8～20kHz]である必要があります。	
• 完了したら、[許可]をクリックします。 • [システムの共通要素]をクリックすると、次の手順に進みます。	
• システムの共通要素 • モータ番号の最高値+1 を設定します。 (モータ#0 を含む) • 座標系の最高値+1 を設定します。 (座標系 0 を含む) • 完了したら、[許可]をクリックします。	
• [システム]ウィンドウを閉じて、仮に完了します。	
モータを追加します。 • プロジェクトツリーで[モータ]フォルダを右クリックします。 • [モータを追加]をクリックします。	
• モータ番号を選択します。 • トポロジを選択します(この例では[シングルフィードバック])。	

手順	説明図
アンプを指定します。 • ［アンプ詳細］アイコンをクリックします。または、プルダウンメニューの DB から初期値または以前に保存したアンプを選択します。	
• ［メーカ］を選択します。 （Omron と Delta Tau が初期値です） • [Omron]で CK3A モデルを選択します。	
• ［アンプの選択］をクリックします。 ここで変更を行う必要はありません。	
• ［アンプの選択］をクリックします。	
• 入力（供給）電圧を入力します。 この例では、AC110V = 約 DC155.54V • ［保存］アイコンをクリックします。	

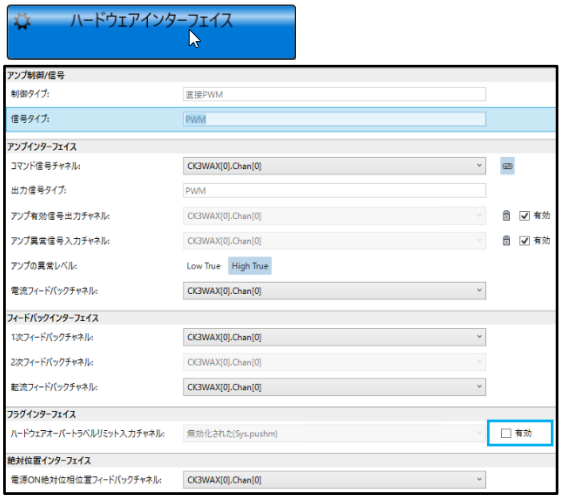
手順	説明図
モータを指定します。 • [モータ詳細]アイコンをクリックします。または、プルダウンメニューの DB から以前に保存したモータを選択します。	
• 新規モータに[新規]をクリックします。	
• 新規モータ情報を入力します。	
• [保存]をクリックします。	
• [モータの選択]をクリックします。	
• [保存]アイコンをクリックします。	

手順	説明図
エンコーダを指定します。 • [エンコーダ詳細]アイコンをクリックします。または、プルダウンメニューの DB スから以前に保存したエンコーダを選択します。	
• 新規エンコーダに[新規]をクリックします。	
• 新規エンコーダ情報を入力します。 有効分解能結果(次の手順でのユーザ単位演算用)に注意してください。	
• [保存]をクリックします。	
• [プライマリエンコーダの選択]をクリックします。	
• [位相方法]を選択します。 • 該当する場合、[位相方法]のフィードバックデバイスを選択します。 • 軸工学のユーザ単位[度 (deg)]を選択します。 • ユーザ単位毎のカウント数を入力します。 この例では、ユーザ単位が 1048576／360 となります。 • 完了したら、[保存]アイコンをクリックします。	

手順	説明図

ハードウェアインターフェイス

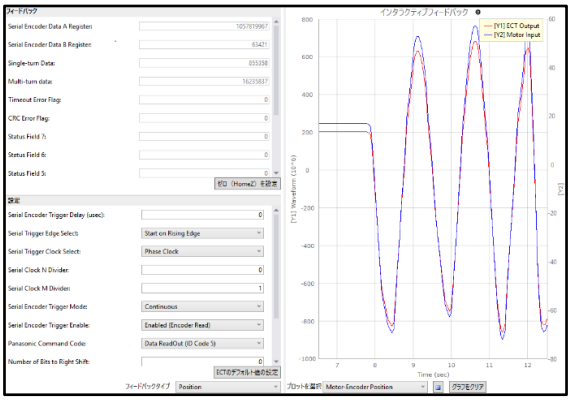
- [ハードウェアインターフェイス]をクリックします。
- 通常、本ページで考慮すべき 1 つの項目は、オーバートラベルリミットです。配線されていない場合は無効にする必要があります。



- [許可]をクリックします。
- [インタラクティブフードバック]をクリックすると、次の手順に進みます。

インタラクティブフードバック

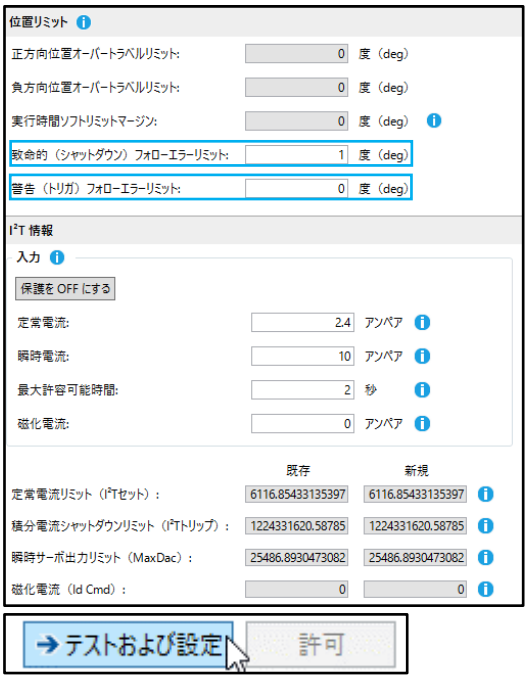
- 本ページは、エンコーダフィードバックの検証に役立ちます。通常、可能で安全な場合は、モータを手で動かし、位置の変化を監視します。
- 一般的に、本ページの変更は行いません。



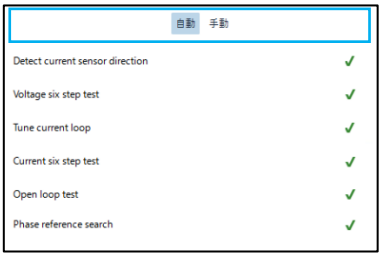
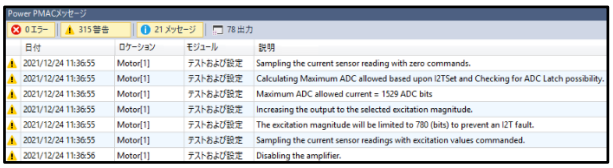
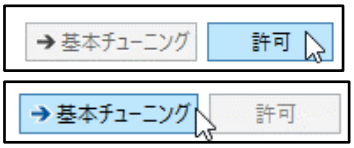
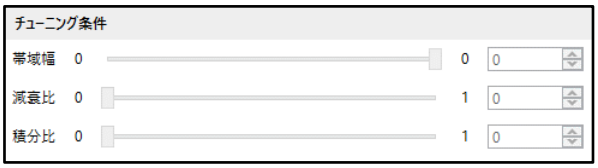
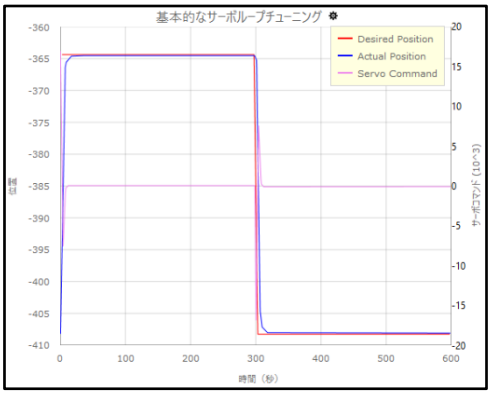
- [許可]をクリックします。
- [運転リミット]をクリックすると、次の手順に進みます。

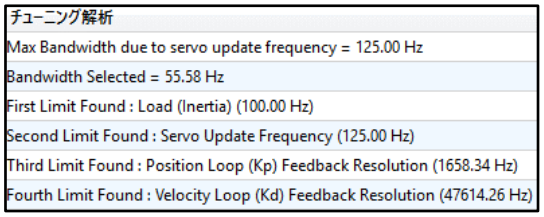
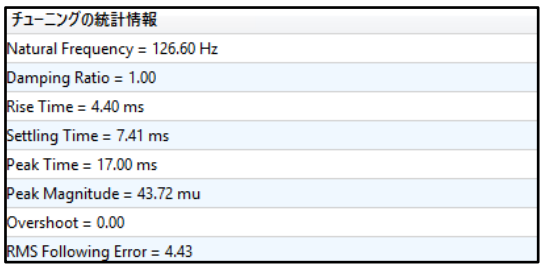
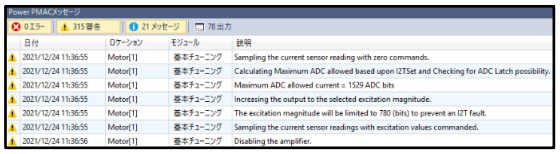
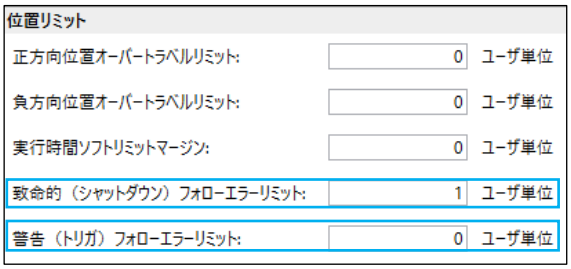
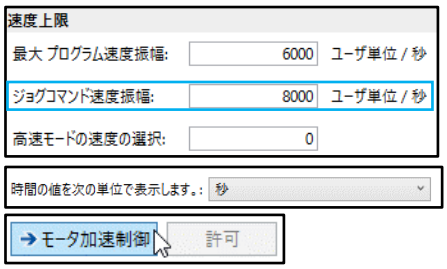
運転リミット

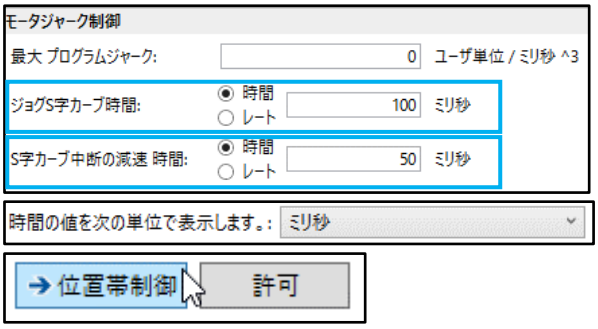
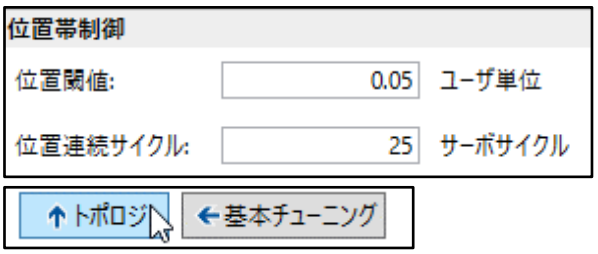
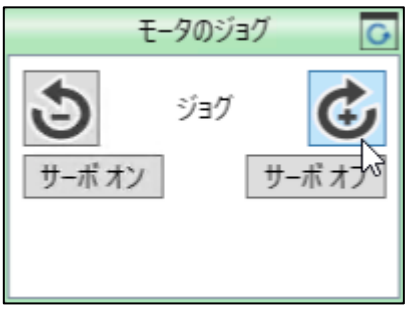
- 通常、本ページで考慮すべき項目は、[致命的(シャットダウン)フォローエラーリミット]と[警告(トリガ)フォローエラーリミット]です。
- I2T 設定は自動的に計算されます。
- 特にモータの定格が CK3A よりも低い場合は、I2T 保護を OFF にすることは決してお勧めしません。



- [許可]をクリックします。
- [テストおよび設定]をクリックすると、次の手順に進みます。

手順	説明図
テストおよび設定 • これらのテストは、多くの場合、高い加速率でモータを動かします。注意が必要です。モータはしっかり装着と固定する必要があります。 • ユーザは、[自動]モードまたは[手動]モードを選択できます。 • [自動]モードは、リストされたテストを順番に実行します。 • [自動]モードでテストが失敗すると、自動的に[手動]モードに切り替わります。 • [手動]モードでは、ユーザは実行するテストを選択できます。 • テスト中、構造体要素の設定が出力ウィンドウに表示されます。これは、トラブルシューティングの処置に役立つ可能性があります。 • さらに、Power PMAC メッセージウィンドウでより詳細なメッセージを監視できます。これは、トラブルシューティングの処置に役立つ可能性があります。 • 次の手順を続行する前に、すべてのテストに合格する必要があります。 • [許可]をクリックします。 • [基本チューニング]をクリックします。	 Step No. | PowerPMAC Command | Value 3 | Motor[1].PhaseOffset | 683 3 | Motor[1].PwmSf | 13337 3 | Motor[1].PhasePosSf | 2048/(4096 * 1 * 209715.200000)  
基本(オート)チューニング • これは、初期ユーザ入力なしのオートチューニングツールです。 • [チューニングを開始]をクリックして、処理を初期化します。 • この識別とチューニング処理中に、モータは、多くの場合、高い加速率で移動します。注意が必要です。モータはしっかり装着と固定する必要があります。	
• この結果は、ステップ移動応答のプロットを示しています。	

手順	説明図
• この結果は、オートチューニングアルゴリズムによって選ばれた条件を示しています。これらのパラメータ(例えば、0 積分比)を変更することが可能です。その後、[チューニングを開始]をクリックします。	
• この結果は、チューニング解析を示しています。これは、帯域幅の選択といくつのシステムリミットを説明しています。	
• この結果は、ステップ移動応答の統計を示しています。	
• (注)　Power PMAC メッセージ出力ウィンドウは、オートチューニング処理中の情報を提供します。	
• 完了したら、[許可]をクリックします。 • [試運転]をクリックすると、次の手順に進みます。	
試運転 • [モータ制限パラメータ制御]をクリックすると、試運転を開始します。	
位置リミット • 通常、致命的フォローエラーリミットおよび警告フォローエラーリミットの確認がここでの主な考慮項目です。 • 絶対値原点復帰パラメータを設定した後、絶対値エンコーダに正負ソフトリミットを設定できます。インクリメンタルエンコーダの場合、ソフトリミットは通常、原点復帰ルーチンの実行後に設定されます。 • [許可]をクリックした後、[モータ速度制御]をクリックして、次の項目に進みます。	
速度上限 • 一般的に、ここでの主な考慮項目はジョグ速度(振幅)です。最大プログラム速度は通常、座標系設定と組み合わせてモーションプログラムに構成されます。 • ユーザ入力の時間単位を選択できます。	

手順	説明図
加速／減速上限 • 一般的に、ここでの主な考慮項目は、ジョグ直線加速／減速および中断直線減速の時間または速度です。最大プログラム加減速度は、通常、座標系設定と組み合わせてモーションプログラムに構成されます。 • [許可]をクリックした後、[モータジャーク制御]をクリックして、次の項目に進みます。	
モータジャーク制御 • 一般的に、ここでの主な考慮項目は、ジョグ直線加速／減速および中断直線減速の時間または速度です。最大プログラム加減速度は、通常、座標系設定と組み合わせてモーションプログラムに構成されます。 • [許可]をクリックした後、[位置帯制御]をクリックして、次の項目に進みます。	
モータの位置帯制御 • [位置閾値]は、Motor[x].InPos ビットが TRUE に設定する前に、[位置連続サイクル]時間に保持するフォローエラーウィンドウを定義します。 • これで基本的な試運転は終了です。 • [トポロジ]をクリックして、終了します。	
モータのジョグ • このコントロールパネルは、オプションのジョグ試運転です。	

A-2　Power PMAC3 手動モータの設定

ここでは、Gate3、ACC-84E、および CK3A DirectPWM アンプありの Power PMAC コントローラを使用してブラシレスモータを手動で設定する手順について説明します。手順は、以下に示します。

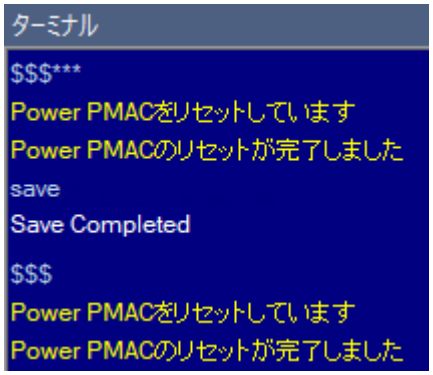
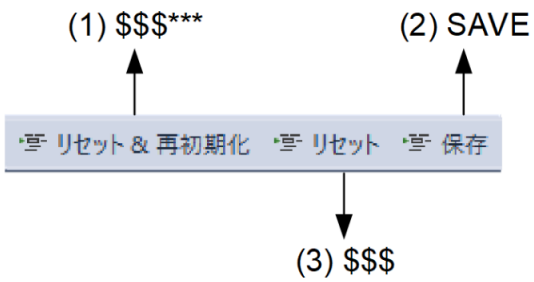
1. IDE プロジェクトを作成します。
2. 基本的な最適化設定、システム設定およびゲート設定を行います。
3. エンコーダの進行中位置の設定と検証を行います。
4. モータの設定と試運転を行います。

A-2-1　IDE プロジェクトの作成

工場出荷時設定のリセット

新規プロジェクトの場合、クリーンな開始点を確保するために、工場出荷時設定から開始することを強くお勧めします。工場出荷時設定のリセットは、グローバルリセット[$$$***]、[SAVE]、通常リセット[$$$]順番を発行することで実行できます。

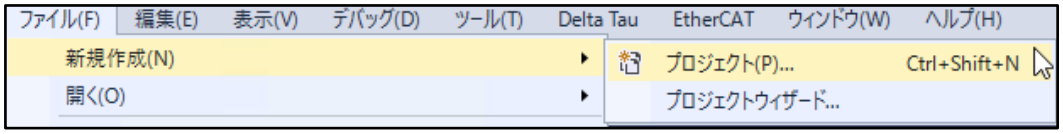
この操作は、ターミナルウィンドウから、または IDE ツールバー画面を使用して実行できます。

ターミナルウィンドウ	IDE ツールバー
	

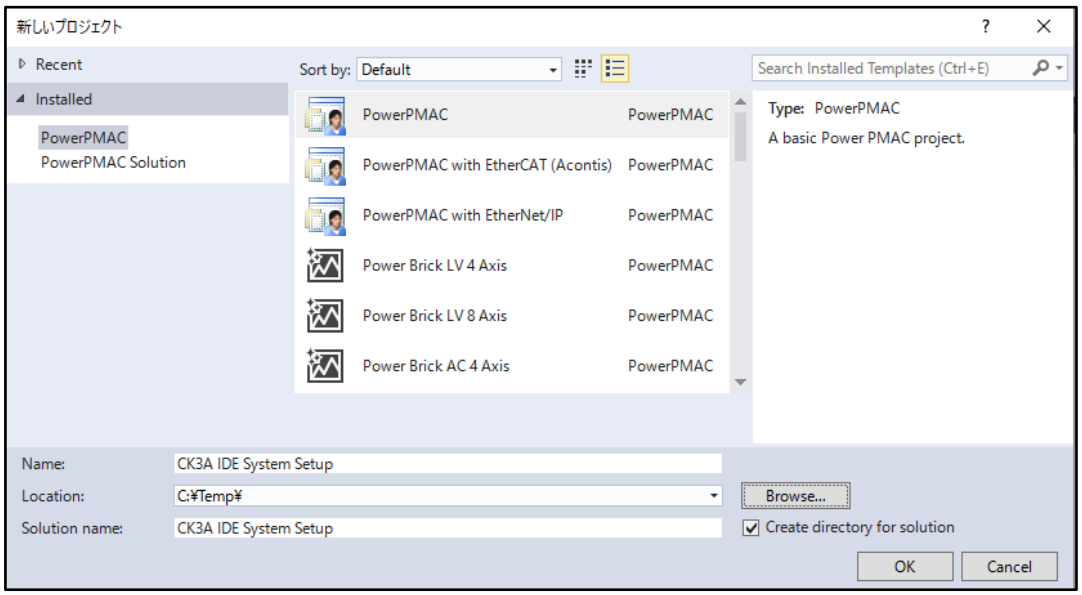
新規 IDE プロジェクト

新規プロジェクトの作成は、[ファイル]メニューから行います。

- [ファイル] - [新規作成] - [プロジェクト]を選択します。



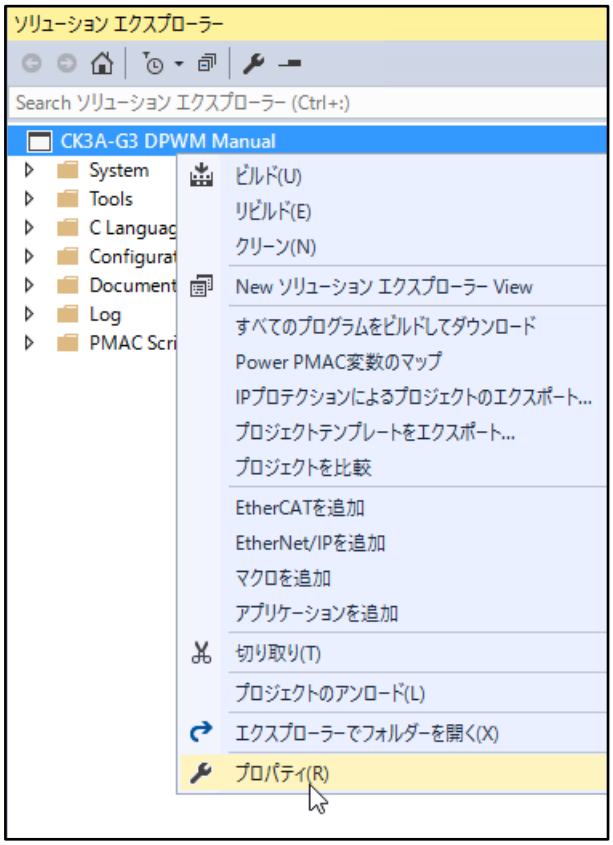
- テンプレートを選択します。
- プロジェクトの場所を選択します。
- プロジェクト名を入力します。

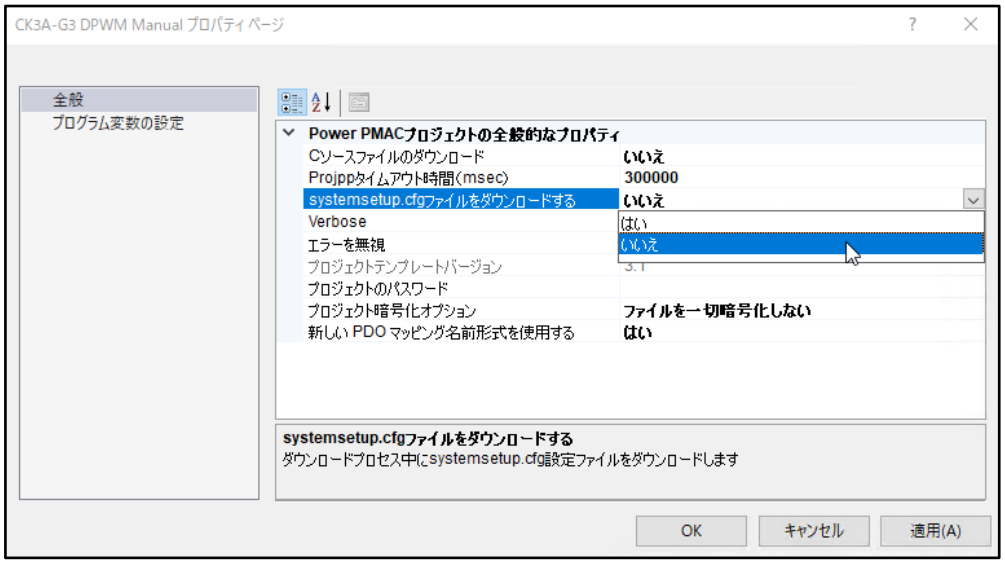


Systemsetup.cfg

IDE システム設定ツールを使用しない場合は、systemsetup.cfg ファイルを無効にする必要があります。

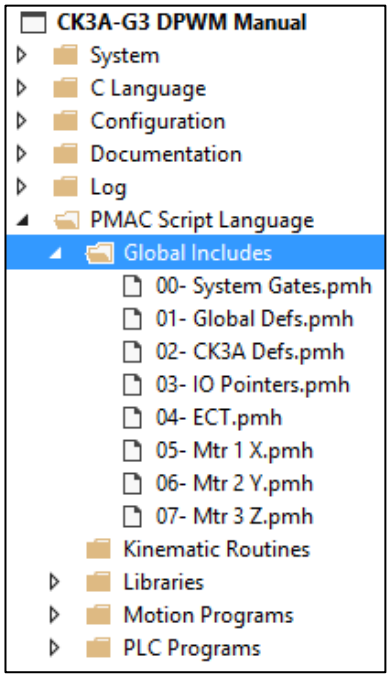
- プロジェクト名を右クリックします。
- ［プロパティ］を選択します。
- ［Systemsetup.cfg ファイルをダウンロードする］-［いいえ］-［OK］を選択します。

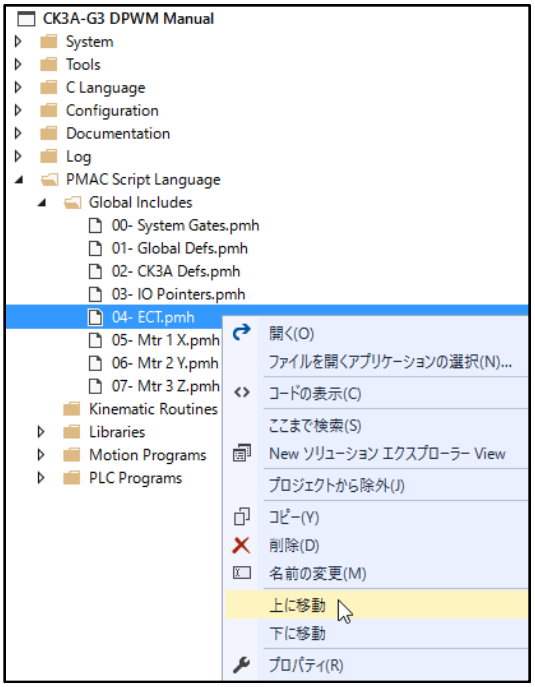




プロジェクト構成

下記で説明する大多数のパラメータは、通常、[Global Includes]の下に配置しています。このフォルダ内のファイルは、ユーザ設定に従って管理できます。ファイルの追加、削除、既存プロジェクトからの挿入、名前の変更、並べ替え(上下に移動)などができます。これらの操作については、IDE のマニュアルを参照してください。1つお勧めの例を下記に示します。



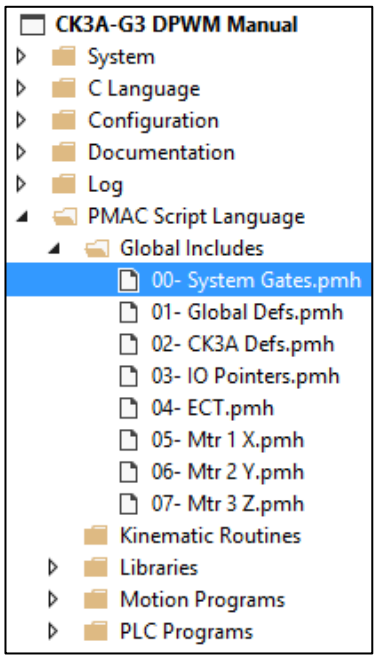


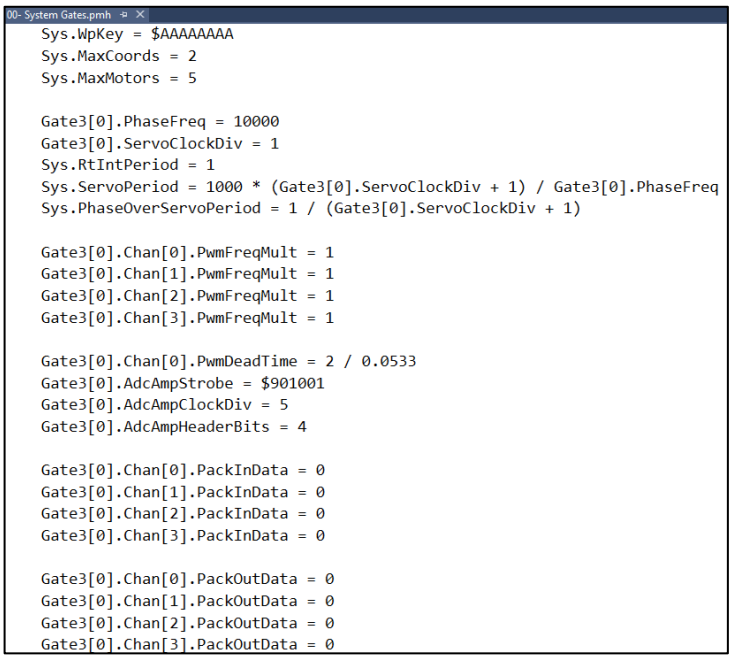
下記の表は、お勧めファイルの簡単説明です。

ファイル	典型的な内容	内容例
System Gates	システム構造体要素	Sys.MaxMotors
	Gate3 構造体要素	Gate3[0].PhaseFreq
	チャネル構造体要素	Gate3[0].Chan[0].PwmFreqMult
Global Defs	ユーザ定義変数	GLOBAL MyVariable
CK3A Defs	CK3A 固有変数	GLOBAL Ck3a1DcBus
IO Pointers	デジタル入出力ポインタ	PTR Input1->Gate3[0].GpioData[0].0.1
	アナログ入出力ポインタ	PTR AdcIn1->S:IO:$900040.16.16
ECT	エンコーダ変換テーブル	EncTable[1].type
Mtr1 X	モータ構造体要素	Motor[1].ServoCtrl

A-2-2　基本的な最適化設定、システム設定およびGate設定

ここでは、基本的な最適化設定、クロック周波数、および CK3A の固有構造体要素について説明します。





構造体要素	内容と実例
Sys.WpKey	**Gate3 の書き込み保護キー** 書き込み保護の無効、標準:= $AAAAAAAA
Sys.MaxCoords	**使用する座標系の最高値+1** 例:= 2(座標系 0..1 を使用、0 は通常未使用)
Sys.MaxMotors	**使用するモータ番号の最高値+1** 例:= 5(モータ#0..4 を使用、0 は通常未使用)
Gate3[0].PhaseFreq	**[Hz]単位のフェーズクロック周波数** 例:= 10000(10kHz)
Gate3[0].ServoClockDiv	**サーボクロック分周器(サーボ周波数を定義)** サーボ周波数 = フェーズ周波数/(ServoClockDiv + 1) 例:= 1(5kHz)
Sys.ServoPeriod	**補間用サーボ更新期間** 正しい周波数設定に必要です。固定式として設定できます。 = 1000 * (Gate3[0].ServoClockDiv + 1) / Gate3[0].PhaseFreq
Sys.PhaseOverServoPeriod	**位相とサーボの比率** 正しい周波数設定に必要です。固定式として設定できます。 = 1 / (Gate3[0].ServoClockDiv + 1)
Gate3[i].Chan[j].PwmFreqMult	**PWM 周波数乗算器(PWM 周波数を定義)** PWM 周波数 = (PwmFreqMult + 1)* PhaseFreq / 2 例:= 1(10kHz、CK3A の範囲は 8～20kHz)
Gate3[i].Chan[j].PwmDeadTime	**PWM 不感時間** 常に= 2 / 0.0533(形 CK3A-G305L の場合は 2μsec) 常に= 3 / 0.0533(形 CK3A-G310L または形 CK3A-G320L の場合は 3μsec)
Gate3[i].AdcAmpStrobe	**アンプ ADC ストローブワード(CK3A 固有)** 例:= $901001(本マニュアルで詳しく説明します)

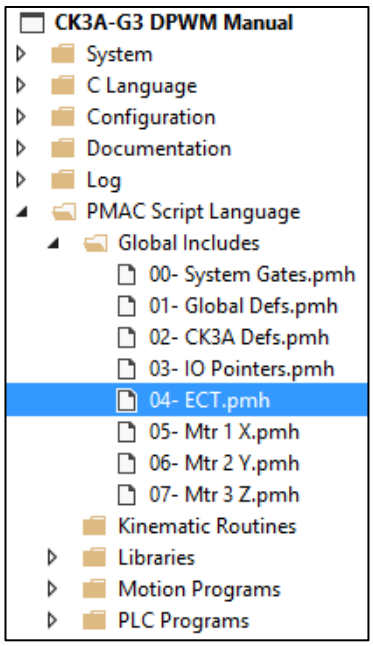
Gate3[i].AdcAmpClockDiv	**アンプ A/DC コンバータのクロック周波数（CK3A 固有）** CK3A の許容 Gate3 設定 = 5（3.125MHz）、= 4（6.250MHz）
Gate3[i].AdcAmpHeaderBits	**アンプ ADC ヘッダビット数（CK3A 固有）** CK3A の場合は、常に= 4
Gate3[i].Chan[j].PackInData	**ADC 入力パックデータ有効** CK3A の場合は、常に= 0
Gate3[i].Chan[j].PackOutData	**ADC 出力パックデータ有効** CK3A の場合は、常に= 0

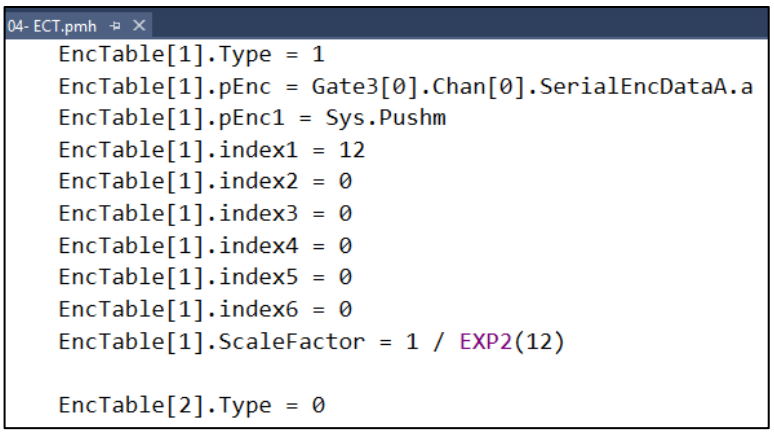
参考

説明の詳細は、「Power PMAC ソフトウエアリファレンスマニュアル (SBCE-405)」に記載されています。

A-2-3 エンコーダの進行中位置の設定と検証

エンコーダ変換テーブル ECT は、進行中の実際位置を生成する主要なエンコーダ入力処理構造です。ECT の結果は、通常、モータサーボループで使用されます。





参考

- 説明の詳細については、「Power PMAC ソフトウエアリファレンスマニュアル (SBCE-405)」を参照してください。
- EncTable[n].Type=0 は、ECT の終了を示します（効率性のために強くお勧めします）。

以下は、最も一般的なタイプのエンコーダフィードバックデバイスの ECT エントリ設定用一般的な目安です。

参考

省略の場合、EncTable[n].Index1 から EncTable[n].Index6 までは、通常= 0 です。

パルスエンコーダ用 ECT エントリ

構造体要素	典型的な設定
EncTable[n].Type	= 1
EncTable[n].pEnc	= Gate3[i].Chan[j].ServoCapt.a
EncTable[n].pEnc1	= Sys.pushm
EncTable[n].ScaleFactor	= 1 / 256

標準アナログ正弦波エンコーダ x16384（UMAC および CK3M）用 ECT エントリ

構造体要素	典型的な設定
EncTable[n].Type	= 1
EncTable[n].pEnc	= Gate3[i].Chan[j].ServoCapt.a
EncTable[n].pEnc1	= Sys.pushm
EncTable[n].ScaleFactor	= 1

● このタイプの処理に必要な他の UMAC（ACC-24E3）設定

構造体要素	設定
Motor[x].EncType	= 6
Gate3[i].Chan[j].AtanEna	= 1

● このタイプの処理に必要な他の CK3M（形 CK3W-AX2323□）設定

構造体要素	設定
Motor[x].EncType	= 6
Gate3[i].Chan[j].AtanEna	= 1
Gate3[i].EncClockDiv	= 3
Gate3[i].AdcEncClockDiv	= 3
Gate3[i].AdcEncCtrl	= $3FFFC000

アナログ正弦波エンコーダ ACI x65536（UMAC）用 ECT エントリ

構造体要素	典型的な設定
EncTable[n].Type	= 1
EncTable[n].pEnc	= Gate3[i].Chan[j].ServoCapt.a
EncTable[n].pEnc1	= Gate3[i].Chan[j].AtanSumOfSqr.a
EncTable[n].ScaleFactor	= 1

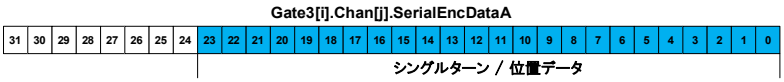
● このタイプの処理に必要な他の設定

構造体要素	設定
Motor[x].EncType	= 7
Gate3[i].Chan[j].AtanEna	= 1
Gate3[i].AdcEncHeaderBits	= 0
Gate3[i].AdcEncStrobe	= $800000

Gate3 のシリアルエンコーダ（CK3W-AX および ACC-24E3）用 ECT エントリ

構造体要素	典型的な設定
EncTable[n].Type	= 1
EncTable[n].pEnc	= Gate3[i].Chan[j].SerialEncDataA.a
EncTable[n].pEnc1	= Sys.pushm
EncTable[n].Index1	=ビット#31 + Index2 に左にシフトするデータビット数。これは 2 番目のシフト操作です。 下図の例では、= 8
EncTable[n].Index2	=ビット#0 に右にシフトするデータビット数（まれなケース）。これは最初のシフト操作です。 下図の例では、= 0
EncTable[n].ScaleFactor	= 1 / $2^{\text{EncTable[n].Index1}}$ 下図の例では、= 1 / 2^{8}

例：



● このタイプの処理に必要な他の設定

構造体要素	設定
Gate3[i].Chan[j].SerialEncEna	= 1
Gate3[i].SerialEncCtrl	「Power PMAC ソフトウエアリファレンスマニュアル (SBCE-405)」を参照ください。
Gate3[i].Chan[j].SerialEncCmd	「Power PMAC ソフトウエアリファレンスマニュアル (SBCE-405)」を参照ください。

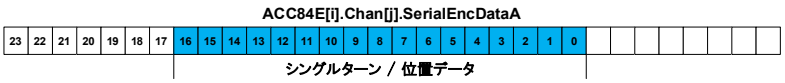
参考

Gate3[i].SerialEncCtrl および Gate3[i].Chan[j].SerialEncCmd を正常に設定すると、Gate3[i].Chan[j].SerialEncEna=1 の生シリアルエンコーダデータが Gate3[i].Chan[j].SerialEncDataA および Gate3[i].Chan[j].SerialEncDataB に表示されます。プロトコルによって、一部の情報は、Gate3[i].Chan[j].SerialEncDataC または Gate3[i].Chan[j].SerialEncDataD に存在する場合があります。

ACC-84E のシリアルエンコーダ（UMAC）用 ECT エントリ

構造体要素	典型的な設定
EncTable[n].Type	= 1
EncTable[n].pEnc	= ACC84E[i].Chan[j].SerialEncDataA.a
EncTable[n].pEnc1	= Sys.pushm
EncTable[n].Index1	=ビット#31 + Index2 に左にシフトするデータビット数。これは 2 番目のシフト操作です。 下図の例では、= 15
EncTable[n].Index2	= 8 + ビット#0 に右にシフトするデータビット数（まれなケース）。これは最初のシフト操作です。 下図の例では、= 8
EncTable[n].ScaleFactor	= 1 / $2^{\text{EncTable[n].Index1}}$ 下図の例では、= 1 / 2^{15}

例：



● **このタイプの処理に必要な他の設定**

構造体要素	設定
ACC84E[i].SerialEncCtrl	「Power PMAC ソフトウエアリファレンスマニュアル (SBCE-405)」を参照ください。
ACC84E[i].Chan[0].SerialEncCmd	「Power PMAC ソフトウエアリファレンスマニュアル (SBCE-405)」を参照ください。

参考

ACC84E[i].SerialEncCtr および ACC84E[i].Chan[j].SerialEncCmd を正常に設定すると、生シリアルエンコーダデータが ACC84E[i].Chan[j].SerialEncDataA および ACC84E[i].Chan[j].SerialEncDataB に表示されます。プロトコルによって、一部の情報は、ACC84E[i].Chan[j].SerialEncDataC または ACC84E[i].Chan[j].SerialEncDataD に存在する場合があります。

エンコーダフィードバックの終了と検証

安全上の要点

- エンコーダデータがないことは、閉ループ制御で非常に危険な状態になる可能性があります。これは、サーボループがモータの実物理位置を認識しなくなったためです。通常は「スタック」していると考えており、激しく反応し、暴走状態を引き起こすことがよくあります。
- モータ閉ループコマンドを発行する前に、エンコーダデータが更新され、正しくカウントされていることを確認してください。
- Motor[x].PosSf および Motor[x].Pos2Sf は、サーボループを含むモータ構造体要素の一部です。初期設定後に変更する場合は、加速度、速度設定、サーボループゲインなどの多くの要素を調整する必要があります。

ECT エントリの結果出力は、通常、以下の構造体要素を使用して位置および速度サーボループで使用されるモータに関連付けられます。

参考

- ECT の処理した出力は、EncTable[n].PrevEnc に表示されます。
- EncTable[n].PrevEnc データは、EncTable[n].ScaleFactor で乗算されません。

構造体要素	内容と実例
Motor[x].ServoCtrl	**モータチャネルをアクティブ** 標準：= 1
Motor[x].pEnc	**位置フィードバックポインタ** 標準：= EncTable[n].a (対応する ECT)
Motor[x].pEnc2	**速度フィードバックポインタ、シングルフィードバックの場合は常に=Motor[x].pEnc** 標準：= EncTable[n].a (対応する ECT)
Motor[x].PosSf	**位置スケール値（ユーザ単位への変換可能）** 度単位の 20 ビットロータリモータ、= 360 / 1048576 1nm スケールのリニアモータ、= 1 / 1000000
Motor[x].Pos2Sf	**速度スケール値、シングルフィードバックの場合は常に=Motor[x].PosSf** 度単位の 20 ビットロータリモータ、= 360 / 1048576 1nm スケールのリニアモータ、= 1 / 1000000
Motor[x].Posunit	**IDE 位置ウィンドウ単位の設定** 度単位の場合、= 11 mm の場合、= 3

● 両方向でのエンコーダカウントの検証

これは、IDE の位置ウィンドウを監視しながら、モータを手で動かすこと（例えば、時計回り、反時計回り、正負方向など）によって行うことができます。

また、ユーザは停止中に位置が安定しているか（固有のディザリング量範囲内）確認する必要があります。

トラブルシューティングの目的で、IDE 位置ウィンドウに位置の変化が見られない場合は、「生エンコーダカウント」を確認すると役立つ場合があります。下記の表は、エンコーダタイプごとのレジスタを示しています。

エンコーダ「処理」タイプ	生カウント構造体要素
パルスエンコーダ	Gate3[i].Chan[j].ServoCapt
アナログ正弦波エンコーダ	Gate3[i].Chan[j].ServoCapt
アナログ正弦波エンコーダ ACI	Gate3[i].Chan[j].ServoCapt
Gate3 のシリアルエンコーダ	Gate3[i].Chan[j].SerialEncDataA Gate3[i].Chan[j].SerialEncDataB（常にデータがあるとは限りません）
ACC-84E のシリアルエンコーダ	ACC84E[i].Chan[j].SerialEncDataA ACC84E[i].Chan[j].SerialEncDataB（常にデータがあるとは限りません）

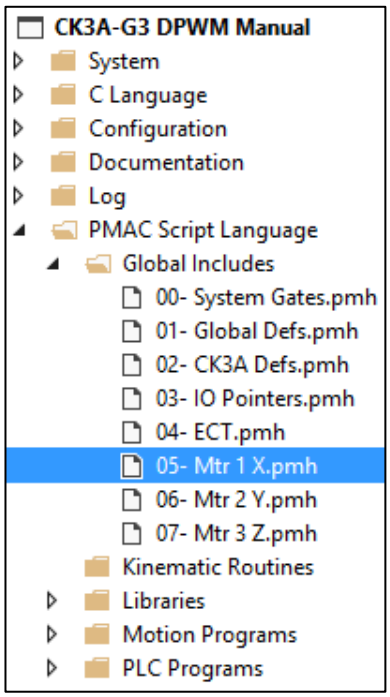
● 正しい距離の報告

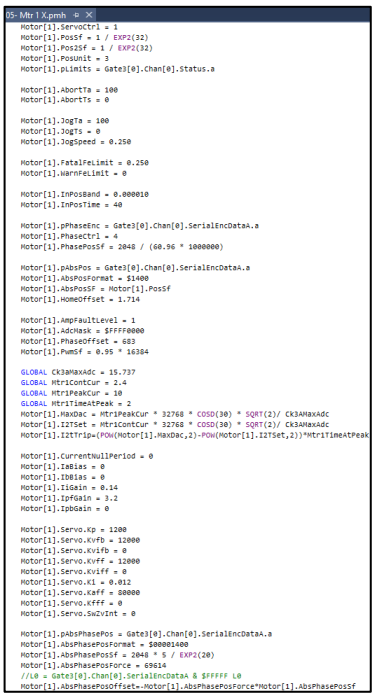
ユーザは、モータを既知の量だけ移動して、IDE 位置ウィンドウに表示されている経過距離を記録することにより、フィードバックデバイスが正しくカウントされているかどうかを確認できます。場合によっては、#nHMZ（n はモータ番号）コマンドを使用して、位置表示をゼロにすることができます。

カウントが正しくない場合は、EncTable[n].ScaleFactor、Motor[x].PosSf、Motor[x].Pos2Sf、および Motor[x].EncType が正しく設定されていることを確認してください。

A-2-4　モータの設定と試運転

ここでは、残りのモータ構造体要素用一般的な目安と、特定の試運転手順について説明します。





一般的なモーション要素

構造体要素	内容
Motor[x].AbortTa	**中断減速** >= 0 の場合の時間[msec] < 0 の場合の反転速度[msec^2／ユーザ単位]
Motor[x].AbortTs	**S 字カーブ中断の減速** >= 0 の場合の時間[msec] < 0 の場合の反転ジャーク速度[msec^3／ユーザ単位]（AbortTa < 0 の場合、必ず設定します）
Motor[x].JogTa	**ジョグ加速／減速** >= 0 の場合の時間[msec] < 0 の場合の反転速度[msec^2／ユーザ単位]
Motor[x].JogTs	**ジョグ S 字カーブ加速／減速** >= 0 の場合の時間[msec] < 0 の場合の反転ジャーク速度[msec^3／ユーザ単位]（AbortTa < 0 の場合、必ず設定します）
Motor[x].JogSpeed	**ジョグ速度** [ユーザ単位／msec]
Motor[x].FatalFeLimit	**フォローエラー異常前の最大位置エラー** [ユーザ単位]
Motor[x].WarnFeLimit	**フォローエラー警告前の最大位置エラー** [ユーザ単位]
Motor[x].InPosBand	**インポジションしきい値** [ユーザ単位]
Motor[x].InPosTime	**位置偏差が Motor[x].InPosBand 範囲内にあり、Motor[x].InPos が TRUE に設定されるまでの時間（連続サーボサイクル数）** = {msec 単位の時間} / Sys.ServoPeriod

参考

Motor[x].pLimits は、ハードウェア認識により、ファームウェアによって自動的に設定されます。
Motor[x].pLimits は、ハードウェアのオーバートラベルリミットの（フラグポインタ）ソースを指定します。
配線されていない場合、または無効にしたい場合は、Motor[x].pLimits を「=0」に設定できます。

アンプと DirectPWM の共通制御要素

構造体要素	内容
Motor[x].AmpFaultLevel	**アンプの異常ロジック状態** CK3A の場合は、常に= 1
Motor[x].AdcMask	**電流フィードバック分解能の指定（ビット単位）** CK3A の場合は、常に= $FFFF0000
Motor[x].PhaseOffset	**モータフェーズ間の角距離** 標準：= 683
Motor[x].PwmSf	**PWM 出力スケール値（分圧器）** モータ定格電圧が主供給電圧以上、= 16384 * 0.95 モータ定格電圧が主供給電圧未満、= 16384*モータ電圧／主電圧

進行中位相（転流）位置

以下の定義は、下の表で使用します。

名称	内容
ST	シングルターンシリアルロータリエンコーダの位置データ
NoOfPolePairs	ロータリブラシレスモータのペア極数
CtsPerRev	ロータリモータの 1 回転あたりのカウント数（通常 x4 ライン）
LinesPerRev	正弦波ロータリエンコーダの正弦波サイクル数
ECLmm	リニアモータの電気サイクル長さまたは磁気ピッチ（mm）（例：30.48mm）
RESmm	リニアエンコーダ分解能（mm）（例：1μm = 0.001mm）

構造体要素	内容と実例
Motor[x].PhaseCtrl	**モータ転流制御** 標準：= 4
Motor[x].pPhaseEnc	**転流進行中位置（ポインタ）ソース** パルスエンコーダ = Gate3[i].Chan[j].PhaseCapt.a アナログ正弦波エンコーダ = Gate3[i].Chan[j].PhaseCapt.a アナログ正弦波エンコーダ ACI = Gate3[i].Chan[j].PhaseCapt.a Gate3 のシリアルエンコーダ = Gate3[i].Chan[j].SerialEncDataA.a ACC-84E のシリアルエンコーダ = ACC84E[i].Chan[j].SerialEncDataA.a
Motor[x].PhaseEncRightShift	**位相位置ソースデータを（ビット#0 に）右にシフトするビット数** パルスエンコーダ = 0 アナログ正弦波エンコーダ = 0 アナログ正弦波エンコーダ ACI = 0 Gate3 のシリアルエンコーダ この例では= 0 （Gate3 のほとんどのシリアルプロトコル） ACC-84E のシリアルエンコーダ この例では= 8 （ACC-84E の標準）

構造体要素	内容と実例
Motor[x].PhaseEncLeftShift	**位相位置ソースデータを(ビット#31 に)左にシフトするビット数** パルスエンコーダ = 0 アナログ正弦波エンコーダ = 0 アナログ正弦波エンコーダ ACI = 0 Gate3 のシリアルエンコーダ この例では= 8 ACC-84E のシリアルエンコーダ この例では= 15 (7 データビット + PhaseEncRightShift)
Motor[x].PhasePosSf	**転流角スケール値**
パルスエンコーダ	ロータリモータ = 2048 * NoOfPolePairs / CtsPerRev リニアモータ = 2048 * RES_{mm} / (256 * ECL_{mm})
アナログ正弦波エンコーダ	ロータリモータ = 2048 * NoOfPolePairs / (LinesPerRev * 16384) リニアモータ = 2048 * RES_{mm} / (16384 * ECL_{mm})
アナログ正弦波エンコーダ ACI	ロータリモータ = 2048 * 4 * NoOfPolePairs / (LinesPerRev * 65536) リニアモータ = 2048 * 4 * RES_{mm} / (65536 * ECL_{mm})
Gate3 のシリアルエンコーダ	ロータリモータ = 2048 * NoOfPolePairs / ($2^{Motor[x].PhaseEncLeftShift + ST}$) リニアモータ = 2048 * RES_{mm} / ($2^{Motor[x].PhaseEncLeftShift}$ * ECL_{mm})
ACC-84E のシリアルエンコーダ	ロータリモータ = 2048 * NoOfPolePairs / ($2^{Motor[x].PhaseEncLeftShift + ST}$) リニアモータ = 2048 * RES_{mm} / ($2^{Motor[x].PhaseEncLeftShift}$ * ECL_{mm})

PMAC I2T 保護

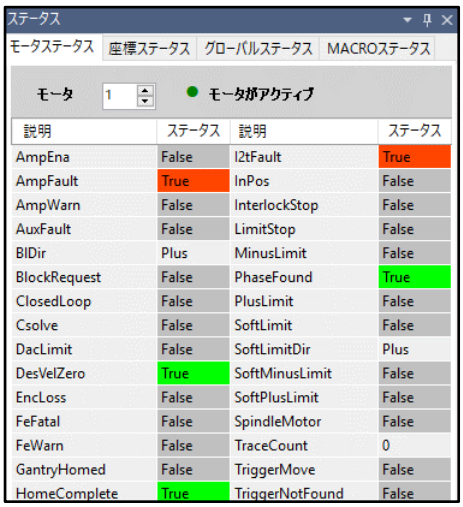
安全上の要点

- PMAC I2T を正しく設定する必要があります。特にモータ電流がアンプよりも低い場合。正しく設定しないと、モータに損傷を与えるリスクの可能性があります。
- 適切な保護のために、アンプとモータ間のより低い電流仕様を常に I2T 演算で使用する必要があります。

CK3A アンプには、過熱から保護するための独自の内蔵 I2T 保護機能がありますので、設定できません。ここでは、モータの過熱から保護する PMAC I2T 機能の設定について説明します。

Motor[x].I2tSum(積分値)が Motor[x].I2tTrip に到達すると、PMAC I2T 異常が発生します。これが発生する場合、アンプイネーブル信号がすぐに OFF に設定され(モータを停止)、アンプ異常と I2T 異常のステータスビットが設定されます(IDE モータステータスウィンドウに表示されます)。

これらのステータスビットには、モータ構造体要素 Motor[x].AmpFault および Motor[x].I2TFault を使用してアクセスできます。



説明	ステータス	説明	ステータス
AmpEna	False	I2tFault	True
AmpFault	True	InPos	False
AmpWarn	False	InterlockStop	False
AuxFault	False	LimitStop	False
BIDir	Plus	MinusLimit	False
BlockRequest	False	PhaseFound	True
ClosedLoop	False	PlusLimit	False
Csolve	False	SoftLimit	False
DacLimit	False	SoftLimitDir	Plus
DesVelZero	True	SoftMinusLimit	False
EncLoss	False	SoftPlusLimit	False
FeFatal	False	SpindleMotor	False
FeWarn	False	TraceCount	0
GantryHomed	False	TriggerMove	False
HomeComplete	True	TriggerNotFound	False

PMAC I2T の設定には、アンプの最大電流、定格電流、最大電流での許容可能時間、および電流スケーリングに関する知識が必要です。下記の表は、CK3A アンプの仕様を示しています。

仕様	形 CK3A-G305L	形 CK3A-G310L	形 CK3A-G320L
定格 RMS 電流	$5A_{rms}$	$10A_{rms}$	$20A_{rms}$
最大 RMS 電流	$10A_{rms}$	$20A_{rms}$	$60A_{rms}$
電流スケーリング（max. ADC）	$15.735A_{peak}$	$31.470A_{peak}$	$93.844A_{peak}$
最大電流での許容可能時間	2sec	2sec	2sec

適切な保護のために、アンプとモータ間のより低い電流仕様を常に I2T 演算で使用する必要があります。下記の表は、この選択処理をまとめたものです。

定格 RMS 電流条件	I2T 演算で使用する仕様
モータ≤アンプ	モータ定格 RMS 電流
モータ>アンプ	アンプ定格 RMS 電流
最大 RMS 電流条件	**I2T 演算で使用する仕様**
モータ≤アンプ	• モータの最大 RMS 電流 • 最大電流でのモータ許容可能時間
モータ>アンプ	• アンプの最大 RMS 電流 • 最大電流でのアンプ許容可能時間

参考

CK3A アンプの定格電流と最大電流の仕様は RMS に示されます。モータ電流の仕様が「ピーク」で示される場合、「√2」で分割する必要があります。

以下の定義を前提として、PMAC I2T を設定するための式を下の表に示します。

名称	内容
MaxRmsCur	モータとアンプ間最大 RMS 電流の最小値
RatedRmsCur	モータとアンプ間最低定格 RMS 電流
TimeAtMaxCur	最大電流での時間（秒単位）
Ck3aMaxAdc	アンプの電流スケーリング（アンペアピークで定義）

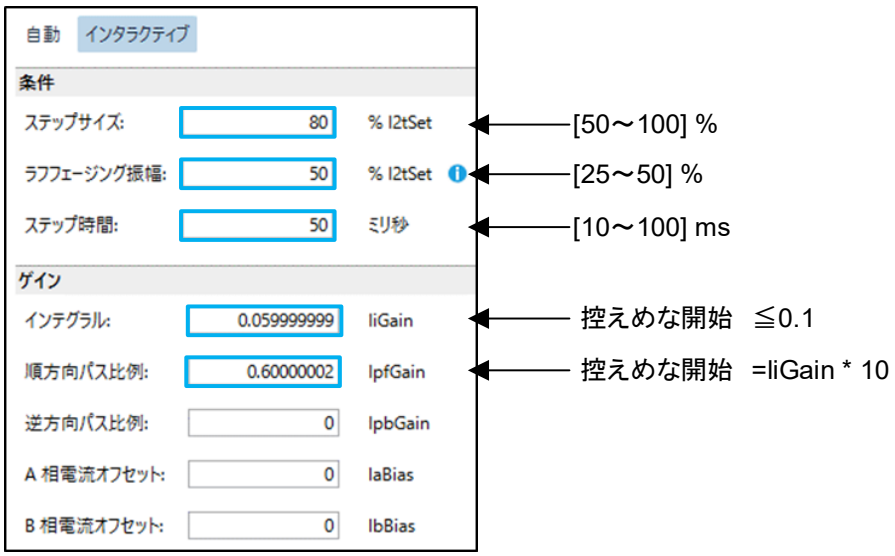
構造体要素	内容と実例
Motor[x].MaxDac	**最大サーボ出力** = MaxRmsCur * 32768 * SQRT(2) * COSD(30) / Ck3aMaxAdc
Motor[x].I2tSet	**定常電流制限** = RatedRmsCur * 32768 * SQRT(2) * COSD(30) / Ck3aMaxAdc
Motor[x].I2tTrip	**最大（シャットダウン）電流制限** = ((Motor[x].MaxDac * Motor[x].MaxDac) - (Motor[x].I2tSet * Motor[x].I2tSet)) * TimeAtMaxCur

参考

SQRT()と COSD()は、PMAC スクリプトの平方根とコサイン（度数）です。

電流ループチューニング

ここでは、IDE の手動電流ループチューニングツールの概要を示します。



フィールド	内容と範囲
ステップサイズ	**電流ステップ中の適用電流** 標準範囲：[Motor[x].I2tSet/2 - Motor[x].I2tSet]
ラフフェーシング振幅	**電流ステップ前にモータをゼロ電気サイクルに強制する適用電流** 標準範囲：[Motor[x].I2tSet/4 - Motor[x].I2tSet/2]
ステップ時間	**電流ステップ時間** 標準範囲：[10～100]msec

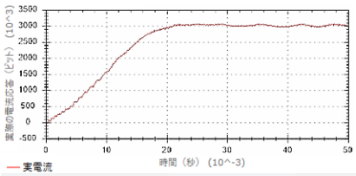
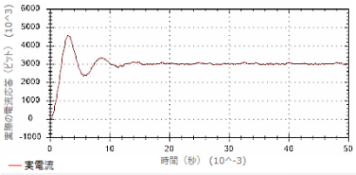
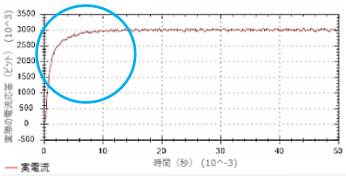
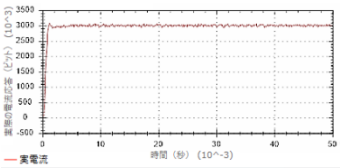
構造体要素	内容と実例
Motor[x].IiGain	**電流ループ積分ゲイン** 控えめな開始、= 0.1 以下
Motor[x].IpfGain	**電流ループ順方向パス比例ゲイン** 控えめな開始、= Motor[x].IiGain * 10
Motor[x].IpbGain	**電流ループバックパス比例ゲイン** この例では使用しません。

大抵の最新モータの電流ループは、Motor[x].IiGain および Motor[x].IpfGain だけを使用して調整できます。目標は、200～1200Hz 範囲のステップ応答固有周波数（通常は位置ループの 10 倍速度）、1～3msec 以下の整定時間、および 3～5%のオーバーシュート以下を達成します。これらの特性は、大抵のサーボモータアプリケーションに適しています。

参考

モータとアンプのパラメータに関する基本的な知識があれば、（開始）電流ループゲインを分析的に計算することができます。これについては、Power PMAC のユーザーズマニュアルで説明します。

以下の図は、電流ループ調整の一般的な目安です。

ステップ応答	監視	対策
 ゲイン 解析 統計情報 統計情報 値 Rise Time 14.599753 ms Peak Time 42.50 ms Peak Magnitude 3,130.00 counts Overshoot 0.68 % Damping Ratio 1.000 Natural Frequency 38.15 Hz Settling Time 18.83 ms	「鈍い」応答 固有周波数が非常に低く、整定時間が長い	Motor[x].IiGain と Motor[x].IpfGain の（比例的に）増加
 ゲイン 解析 統計情報 統計情報 値 Rise Time 1.199980 ms Peak Time 3.20 ms Peak Magnitude 4,470.00 counts Overshoot 51.42 % Damping Ratio 0.207 Natural Frequency 161.59 Hz Settling Time 11.76 ms	「不足減衰」応答 オーバーシュートが多すぎる	Motor[x].IpfGain の増加
 ゲイン 解析 統計情報 統計情報 値 Rise Time 3.599939 ms Peak Time 30.70 ms Peak Magnitude 3,050.00 counts Overshoot 0.73 % Damping Ratio 1.000 Natural Frequency 154.74 Hz Settling Time 6.07 ms	「過減衰」応答（曲線形状） 固有周波数が低くなり、整定時間が長くなる	Motor[x].IiGain の増加、または Motor[x].IpfGain の減少
 ゲイン 解析 統計情報 統計情報 値 Rise Time 0.699988 ms Peak Time 9.10 ms Peak Magnitude 3,052.00 counts Overshoot 0.93 % Damping Ratio 1.000 Natural Frequency 795.79 Hz Settling Time 1.12 ms	良い「臨界減衰 」応答 固有周波数が約 800Hz、整定時間が約 1msec	なし（終了）

参考

電流ループの積分ゲインをアンプ/モータの組み合わせの帯域幅を超えて増加させると、特に応答の遅いモータでは、応答のオーバーシュートと発振が発生する可能性があります。これは、位置ループの性能に悪影響を与える可能性があるため、望ましくありません。

位相基準の確立（モータ位相）

安全上の要点

ここでのモータ位相調整により、モータが動きます。注意が必要です。

ここでは、2 つ最も一般的な「初期」位相方法について説明します。この2つ方法の間に基本的な違いはありません。どちらの方法も、あらゆるタイプのブラシレスモータとエンコーダで使用できます。以下の表は、提案と詳細情報を提供します。

「初期」位相	提案した用途	最終アプリケーション位相方法
手動位相	モータアセンブリに絶対センサがある場合	絶対センサを使用して絶対位相を設定します（後述で説明します）
自動ステッパー	モータアセンブリに絶対センサがない場合	ステッパー位相方法パラメータを PMAC に保存し、電源投入後に位相コマンドを発行します。

参考

- 位相基準確立の詳細については、Power PMAC のユーザーズマニュアルを参照してください。
- モータアセンブリにホールや絶対値エンコーダなどの絶対センサがある場合は、絶対位相方法を実装してコントローラに保存できます。これについては、後述で説明します。
- モータの初期位相調整は、連結してないモータで行うのが最適です。

● 手動位相方法

手動位相方法は、Motor[x].IbBia オフセットに電流を強制することにより、モータを整流電気サイクルのゼロ位置（通常は PMAC の位相 B）にしっかりとロックすることで構成されます。モータがしっかりとロックされるほど、位相基準は良くなります。この手順は通常、以下に説明する手順を使用して、IDE のターミナルウィンドウから実行します。

手順	コマンド	説明
1	#xK	モータが停止され、静止していることを確認してください。
2	Motor[x].IbBias = Motor[x].I2tSet/4	電流量（トルク）[Motor[x].I2tSet/4 - Motor[x].I2tSet]
3	#xOUT0	アンプを有効にします（これによりモータが移動します） モータが落ち着くのを待ちます（移動の停止）
4	Motor[x].PhasePos = 0	これがゼロ位相位置の場所であることを PMAC に伝えます。
5	Motor[x].PhaseFound = 1	ステータスビットの設定
6	Motor[x].IbBias = 0 #xK	Motor[x].IbBias をリセットし、モータを停止します。

x はモータ番号を示します。

参考

- この手順を行う中、モータが振動する可能性があり、Motor[x].IbBias を減らすことができます。あるいは、安全で可能であれば、モータを手で安定させることもできます。
- モータアセンブリに絶対センサがない場合は、手動位相方法をオンデマンド位相用の PLC プログラムに実装（および保存）できます。
- 開ループテスト（後述でさらに説明）は、この方法の成功を検証します。

● 自動ステッパーモータ位相方法

自動ステッパーモータ位相方法は、Power PMAC ファームウェアに組み込まれている 2 つ位相基準ルーチンの 1 つです。以下の構造体要素を設定する必要があります。

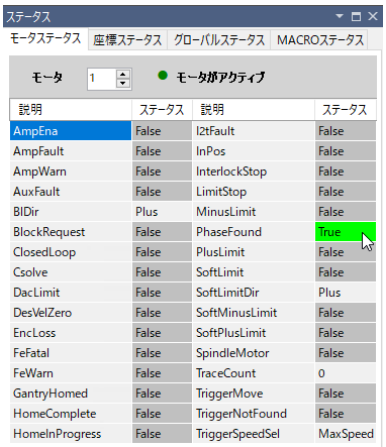
構造体要素	内容と実例
Motor[x].PhaseFindingDac	**位相検索出力(電流/トルク)の大きさ** 標準範囲:[Motor[x].I2tSet/4 – Motor[x].I2tSet/2]
Motor[x].PhaseFindingTime	**位相検索時間** = {msec 単位の時間} / (2 * Sys.ServoPeriod * (Sys.RtIntPeriod + 1))
Motor[x].AbsPhasePosOffset	**有効位相検索移動を適用する最小動作** 標準:= 2048 / 5

これらの設定が PMAC にダウンロードされると、位相検索移動の開始がターミナルまたは PLC プログラムのいずれかから発行される可能性があります。

- ターミナルウィンドウから:#x$(x はモータ番号)
- PLC またはサブルーチンから:Motor[x].PhaseFindingStep = 1

位相検索移動が成功すると、Motor[x].PhaseFound のステータスビットが TRUE に設定されます。

位相検索移動が成功しない場合は、下記アドバイスのいずれかを試してください。



- Motor[x].PhaseFindingDac の増加
- Motor[x].PhaseFindingTime の増加
- (インクリメンタルエンコーダ用)エンコーダデコードの反転
- Gate3[i].Chan[j].EncCtrl =7 <-> 3
- シリアルエンコーダの場合、カウント方向の反転は簡単ではありません。Motor[x].PwmSf と Motor[x].PhaseOffset を同時に反対の符号に設定して、転流方向の反転をしてみてください。
- モータの2つのリード線を交換
- モータを負荷から切り離して、もう一度試してください。

開ループテスト

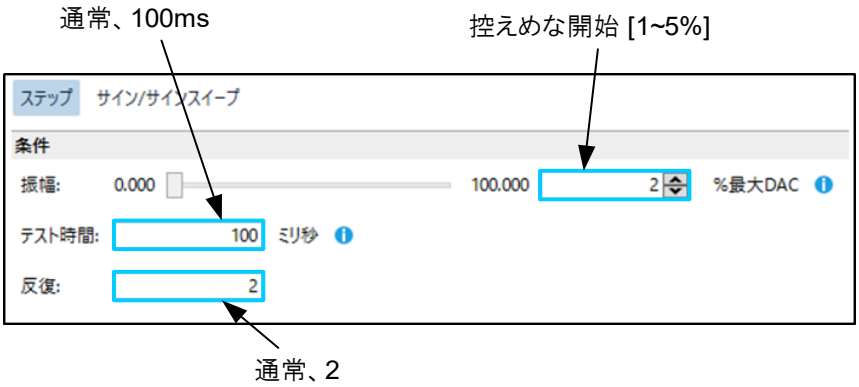
安全上の要点

- 開ループテストに合格しなかったモータの位置ループを閉じないでください。これは危険な暴走状態につながる可能性があります。
- 開ループテストはモータを移動します。注意が必要です。

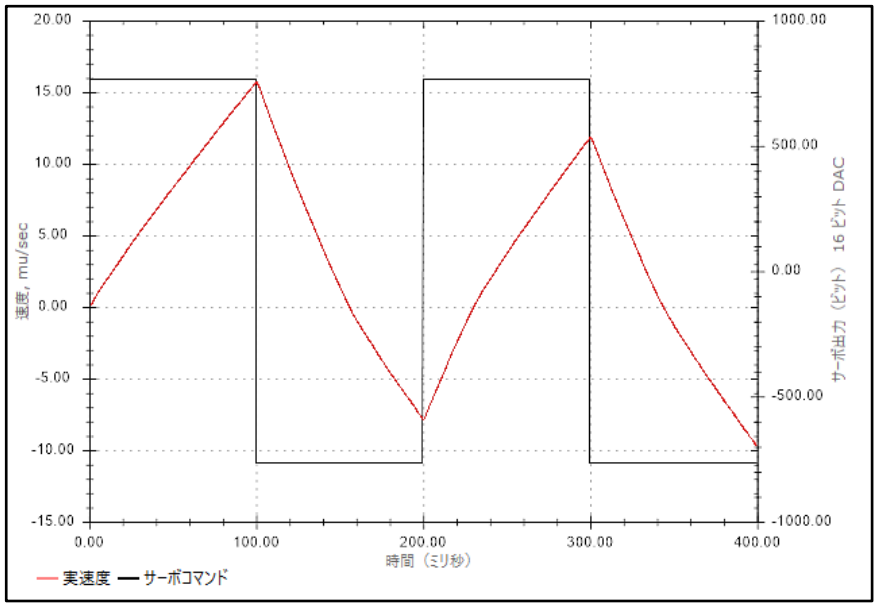
開ループテストは、整流された「ブラシレス」モータの設定における重要なステップです。主に、以下機能の検証を提供します。

- 初期位相の成功
- アンプの出力はエンコーダのカウントセンスと同じ方向

開ループテストは、通常、以下に示す IDE チューニングツールを使用して実行されます。



開ループテストの目標は、正（開ループ）コマンドが発行されたときに正（速度）移動を取得することであり、その逆も同様です。期待の「合格」結果は、以下の「鋸歯」状になります。



● 制限された移動量

一部のモータは、移動量が制限されているため、開ループテストの実行が困難な場合があります。この場合、IDE のターミナルウィンドウから手動の開ループテストを実行できます。

- #xK コマンドによって、モータ x を停止すること（サーボ OFF）を常に覚えておいてください。
- #xOUT1 の場合、モータは正方向に移動していますか?
- #xOUT-1 の場合、モータは負方向に移動していますか?

これは、IDE のモータ速度ウィンドウを確認することで検証できます。出力率は、必要に応じて増加できます。

● 開ループテストのトラブルシューティング

一般的に、開ループテストは以下の 2 つ結果のいずれかで失敗する可能性があります。

- モータが移動しません（ロックされている感じ）
- モータは開ループコマンドの反対方向に移動します（応答は反転した「鋸歯」状を示します）

これは、以下の 1 つまたは組み合わせを示します。いずれかの項目が変更された場合は、別の開ループテストを試行する前にモータ位相の再調整が必要です。

- 転流サイクルサイズ、Motor[x].PhasePosSf の設定が正しくない

- 初期位相の実行が不成功

- 反転したインクリメンタルエンコーダ方向、Gate3[i].Chan[j].EncCtrl = 7 <-> 3
  （シリアルエンコーダに適用しません）

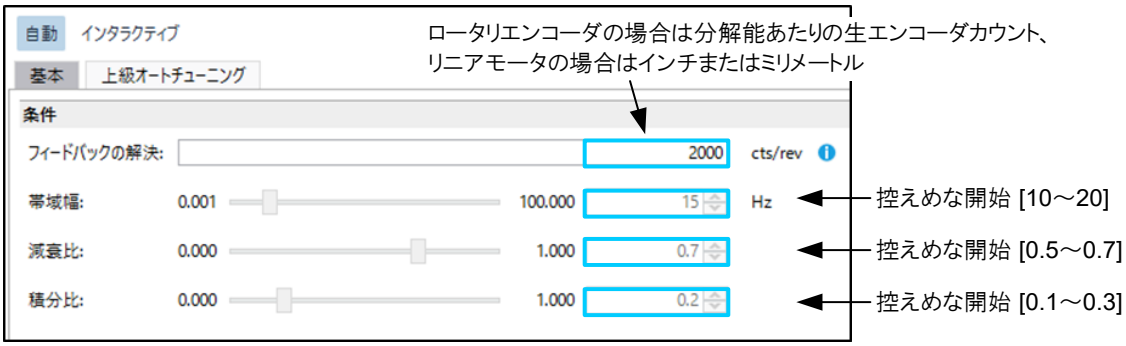
- シリアルエンコーダの方向がアンプの出力と一致していません。この場合、モータの2つのリード線を交換によって、転流カウント方向を反転することを提案します。または、Motor[x].PwmSf と Motor[x].PhaseOffset を同時に反対の符号に設定します。

位置ループ（基本）チューニング

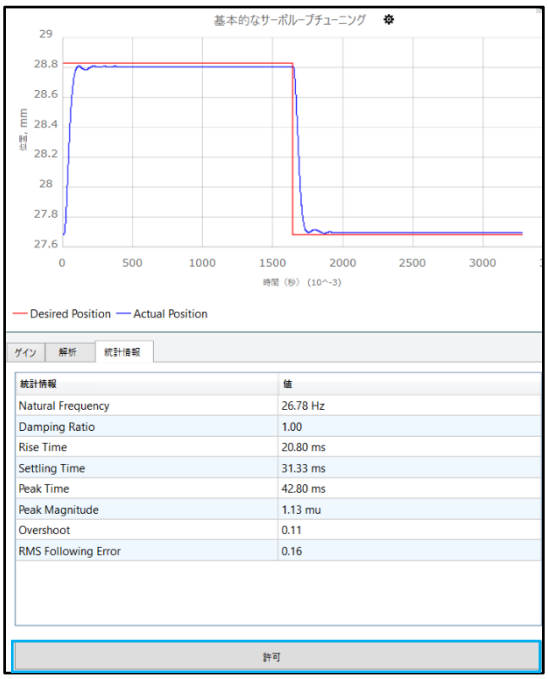
安全上の要点

開ループテストに合格する前に、位置ループチューニングを実行しないでください。これは危険な暴走状態につながる可能性があります。

基本オートチューニングツールでイニシャルゲインを取得できます。



オートチューニングが成功すると、ステップ応答と統計情報が表示されます。

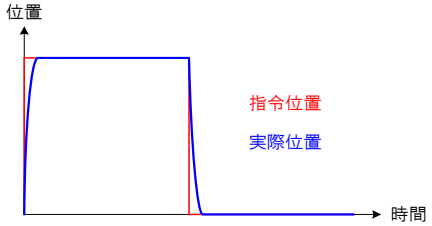


統計情報	値
Natural Frequency	26.78 Hz
Damping Ratio	1.00
Rise Time	20.80 ms
Settling Time	31.33 ms
Peak Time	42.80 ms
Peak Magnitude	1.13 mu
Overshoot	0.11
RMS Following Error	0.16

ユーザーは、オートチューニング ツールによって生成されたゲインを受け入れるか、インタラクティブな微調整を実行するかを選択できます。インタラクティブな微調整で使用する最も一般的な移動プロファイルは、「ステップ」および「放物線」移動プロファイルです。

● ステップ移動（基本）チューニング

ステップ移動チューニングの目標は、指令されたステップ位置の上に実際位置を重ね合わせることです。

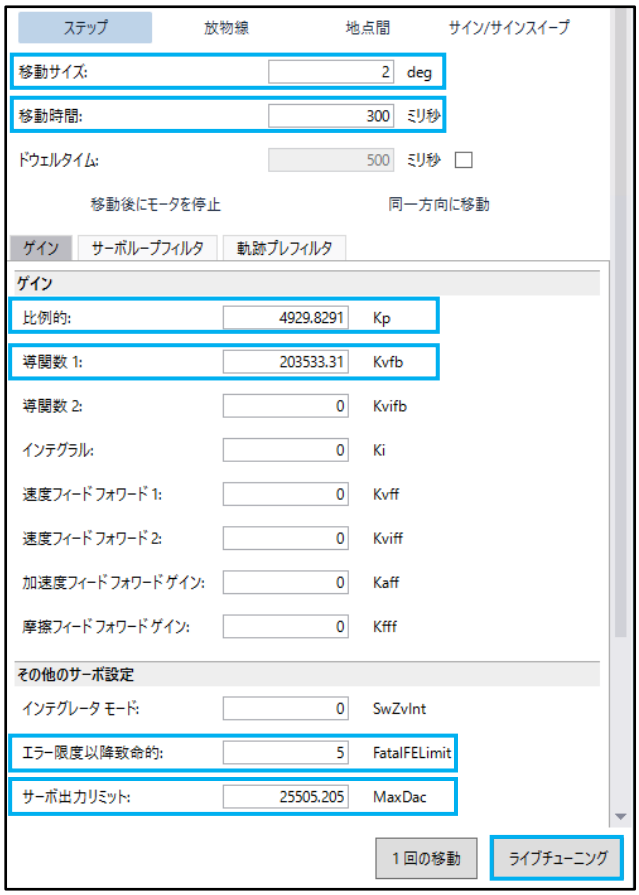


つまり、以下の構造体要素のゲインを最適化します。

- Motor[x].Servo.Kp
- Motor[x].Servo.Kvfb

ステップ移動チューニングの準備：

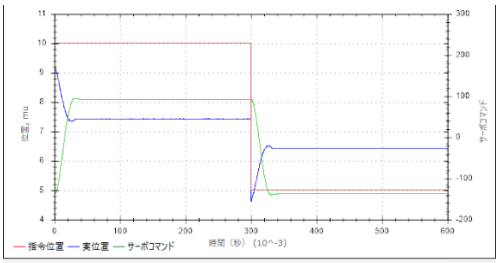
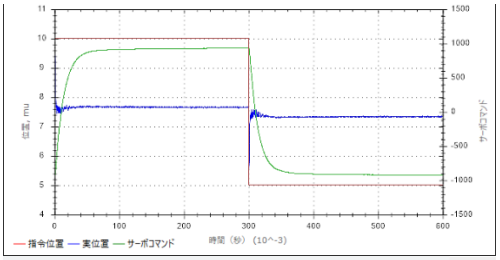
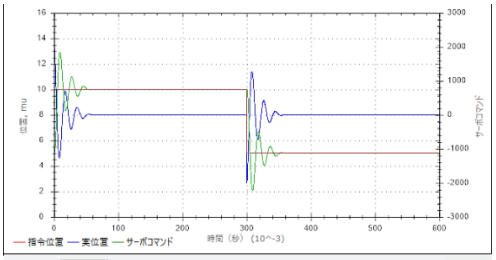
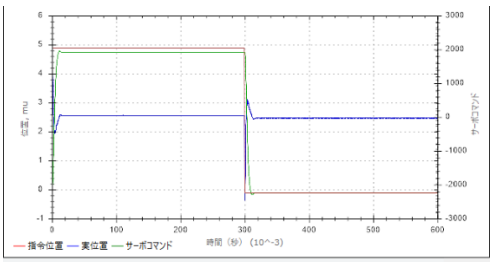
- 他のすべてのゲインがゼロに設定されていることを確認してください。
- 移動サイズ（負荷を移動）を選択します。
  ロータリモータの標準範囲：[1～5 度]
  リニアモータの標準範囲：[0.1～0.5mm]
- 標準範囲の移動時間：[300～2000msec]を選択します。
- ［エラー限度以降致命的］は、［ステップ移動］よりも大きくなければなりません。ステップ移動チューニング演習のために一時的に少し増加することができます。
- ステップ移動応答の過渡部分でのサーボエフォートコマンドと比較して、サーボ出力リミット（Motor[x].MaxDac）に注意してください。



参考

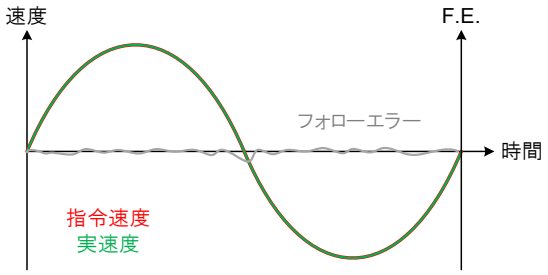
最初に（シングル）ステップ移動を試してください。安全で安定している場合は、ライブチューニング機能を使用して、プロット上の応答の変化を監視しながら、チューニングを止めずにゲインを変更します。

以下の図は、ステップ移動チューニングの一般的目安です。

ステップ応答	監視	対策
 ゲイン 解析 統計情報 統計情報 値 固有周波数 23.70 Hz 減衰比 0.82 立ち上がり時間 17.20 ミリ秒 整定時間 24.83 ミリ秒 ピーク時間 30.60 ミリ秒 ピーク振幅 3.24 mu オーバーシュート 1.20 % RMSフォローエラー 1.57	• 「鈍い」応答。モータが緩む場合がある。 • サーボの「エフォート」コマンドが少なすぎる。この例では、Motor[x].MaxDac の約 1%。 • 比較的長い整定時間	Motor[x].Servo.Kp と Motor[x].Servo.Kvfb の（比例的に）増加
 ゲイン 解析 統計情報 統計情報 値 固有周波数 18.69 Hz 減衰比 1.00 立ち上がり時間 29.80 ミリ秒 整定時間 43.14 ミリ秒 ピーク時間 278.60 ミリ秒 ピーク振幅 4.33 mu オーバーシュート 0.05 % RMSフォローエラー 0.88	• 「過減衰」応答	Motor[x].Servo.Kp の増加、および／または Motor[x].Servo.Kvfb の減少
 ゲイン 解析 統計情報 統計情報 値 固有周波数 59.56 Hz 減衰比 0.17 立ち上がり時間 3.20 ミリ秒 整定時間 39.92 ミリ秒 ピーク時間 9.20 ミリ秒 ピーク振幅 7.93 mu オーバーシュート 58.71 % RMSフォローエラー 0.65	• 「不足減衰」の応答 • オーバーシュートが多すぎる	Motor[x].Servo.Kvfb の増加
 ゲイン 解析 統計情報 統計情報 値 固有周波数 99.47 Hz 減衰比 1.00 立ち上がり時間 5.60 ミリ秒 整定時間 8.06 ミリ秒 ピーク時間 11.60 ミリ秒 ピーク振幅 4.89 mu オーバーシュート 0.89 % RMSフォローエラー 0.50	• 「臨界減衰 」応答 • 連結モータの場合、約 80% の Motor[x].MaxDac を達成するようにしてください。 • 速い整定時間(<10msec) • オーバーシュートがなし、またはそれに近い（<1%） • 高い固有周波数（約 100Hz）	良いステップ移動応答

● 放物線移動（基本）チューニング

放物線移動チューニングの目標は、最小および中央（ゼロ付近）のフォローエラーによって、指令された「放物線」位置の上に実速度を「重ね合わせ」ことです。

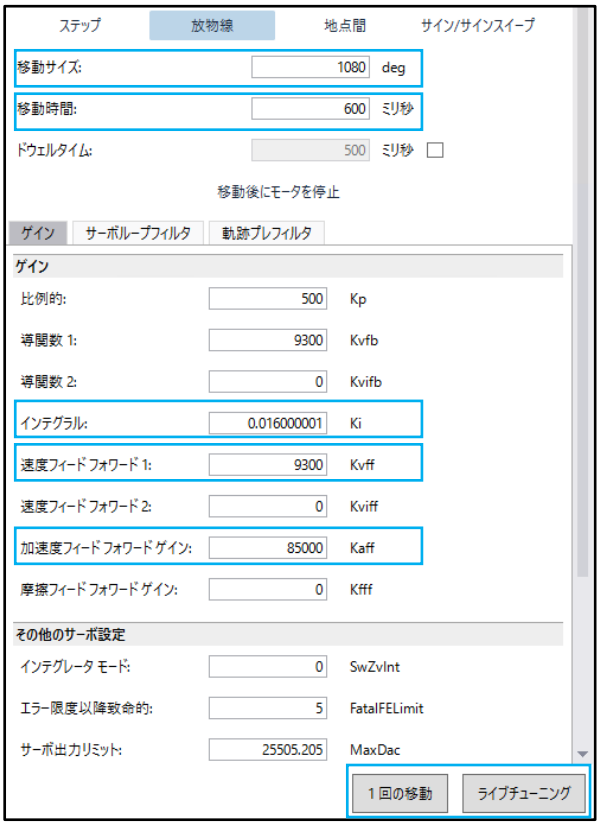


つまり、以下の構造体要素のゲインを最適化します。

- Motor[x].Servo.Kvff
- Motor[x].Servo.Kaff
- Motor[x].Servo.Ki

放物線移動チューニングの準備：

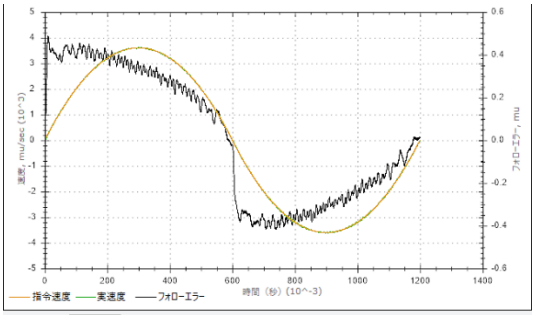
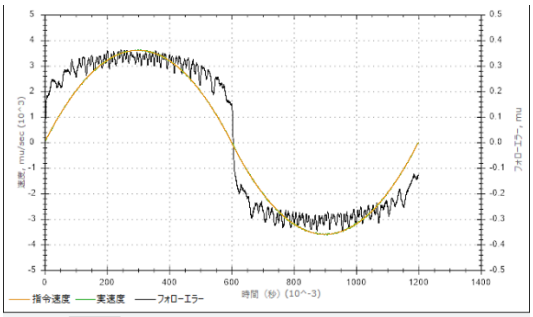
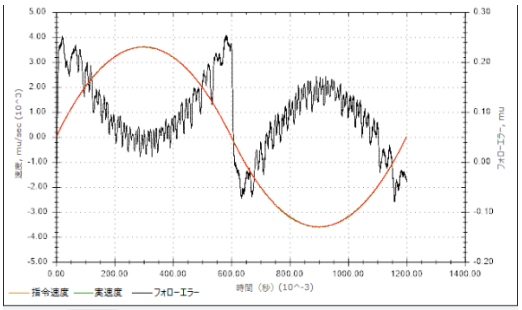
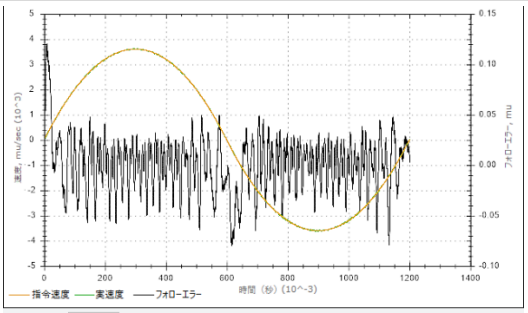
- Motor[x].Servo.Kvfb を Motor[x].Servo.Kvff にコピーします（この 2 つのゲインはほとんど常に等しい）。
- 移動サイズ（負荷を移動）を選択します。
  ロータリモータの標準範囲：[1～4 回転]
  リニアモータの標準範囲：[10～80mm]
- 標準範囲の移動時間：[600～2000msec]を選択します。
- 通常、モータの予想最大速度の約 20～40%を達成することが望ましい。



参考

最初に（シングル）放物線移動を試してください。安全で安定している場合は、ライブチューニング機能を使用して、プロット上の応答の変化を監視しながら、チューニングを止めずにゲインを変更します。

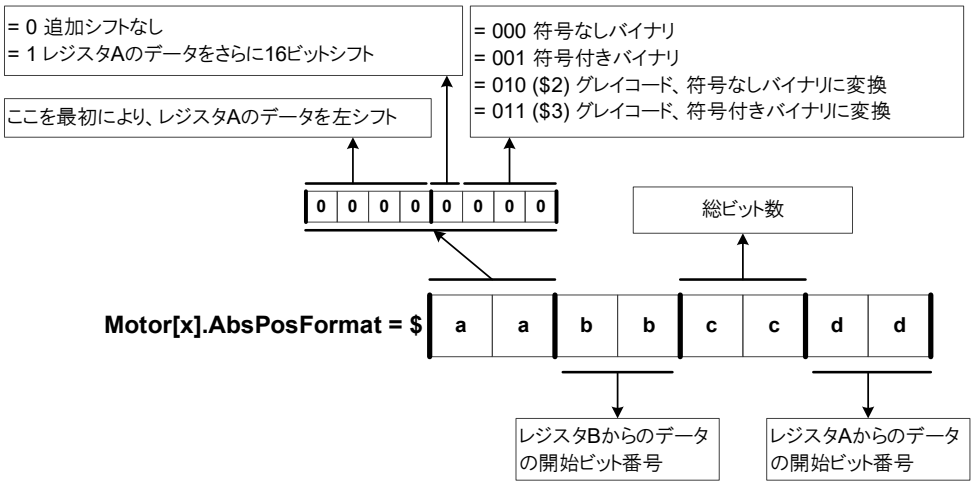
Motor[x].Servo.Ki の前に、最初に Motor[x].Servo.Kaff を最適化することをお勧めします。以下の図は、放物線移動チューニングの一般的目安です。

ステップ応答	監視	対策
 ゲイン 解析 統計情報 統計情報 値 速度の相関関係 0.9086 加速度の相関関係 0.3274 エラー発生後の最大値 0.4892 RMSフォローエラー 0.3076	• 高加速度相関(フォローエラーは V 字型に見える)	Motor[x].Servo.Kaff の増加
 ゲイン 解析 統計情報 統計情報 値 速度の相関関係 0.9651 加速度の相関関係 0.0075 エラー発生後の最大値 0.3628 RMSフォローエラー 0.2821	• 高速度相関(フォローエラーはステップのように見える) • 移動および整定(ポイントツーポイント)アプリケーションの場合、Motor[x].Servo.Kvff を単独で調整しないでください。常に Motor[x].Servo.Kvfb と同じにしてください。	Motor[x].Servo.Kvff の増加
 ゲイン 解析 統計情報 統計情報 値 速度の相関関係 0.0238 加速度の相関関係 0.0387 エラー発生後の最大値 0.2528 RMSフォローエラー 0.1184	• 加速度相関と高速度相関がなし、またはそれに近い • フォローエラースケールに注意 • ゼロを中心としたフォローエラー形状	Motor[x].Servo.Ki の増加
 ゲイン 解析 統計情報 統計情報 値 速度の相関関係 0.0024 加速度の相関関係 0.2278 エラー発生後の最大値 0.1204 RMSフォローエラー 0.0278	• ゼロ付近で最適化されたフォローエラー • プロットの中央にある位置偏差(方向の変化)の急上昇については、(輪郭を描くアプリケーションの場合)少量の Motor[x].Servo.Kfff を追加することができます。	良い放射線移動応答

絶対位置の設定（絶対原点復帰）

ここでは、絶対シリアルエンコーダの「絶対」位置読み出しを設定するための基本的な操作について説明します。この機能を設定するために必要な要素を以下に示します。

構造体要素	内容と実例
Motor[x].pAbsPos	**電源 ON 絶対位置ポインタ** 通常、Gate3 の場合、= Gate3[i].Chan[j].SerialEncDataA.a 通常、ACC-84E の場合、= ACC84E[i].Chan[j].SerialEncDataA.a
Motor[x].AbsPosSf	**絶対位置読み出しスケール値** = Motor[x].PosSf
Motor[x].AbsPosFormat	**電源 ON 絶対データフォーマット** 以下の内容と実例（詳細）を参照してください。
Motor[x].HomeOffset	**位置決め基準オフセット** 希望するゼロ位置を示すためにユーザが（モータ単位で）設定します。

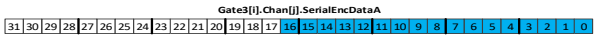
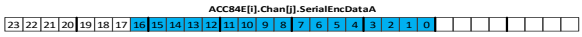
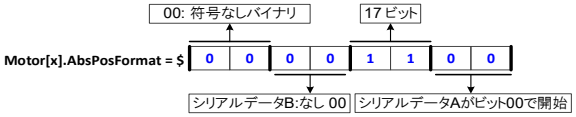
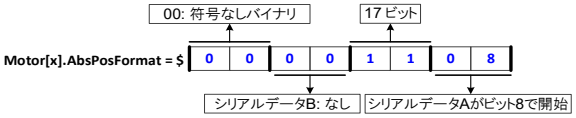
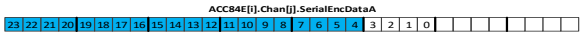
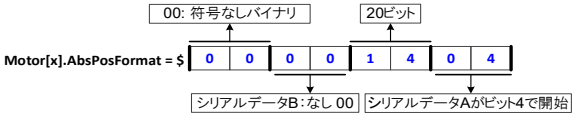
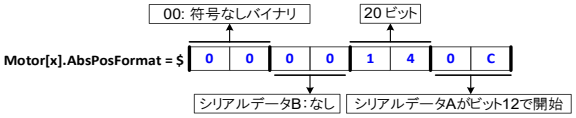
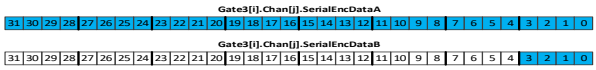
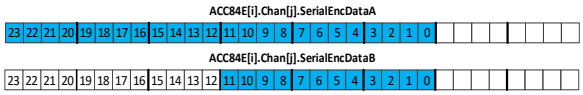
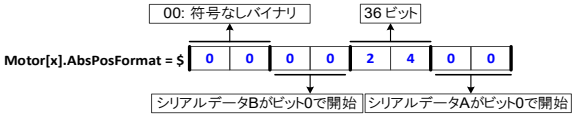
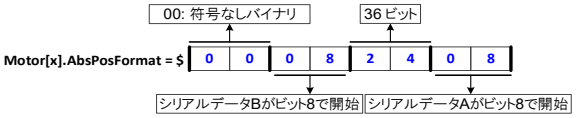
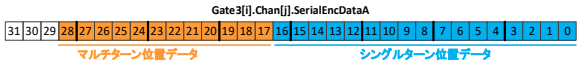
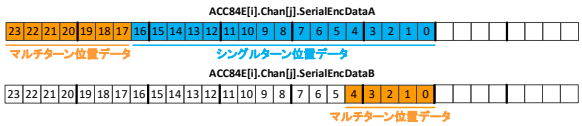
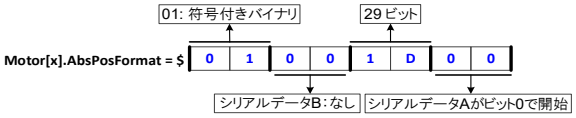
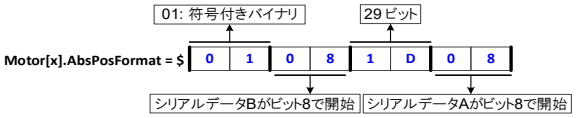


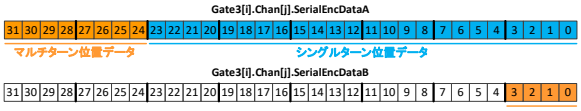
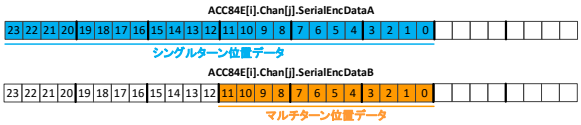
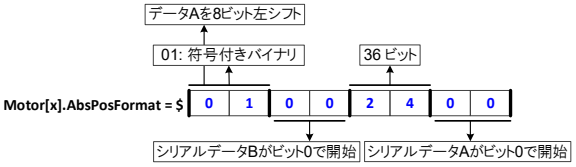
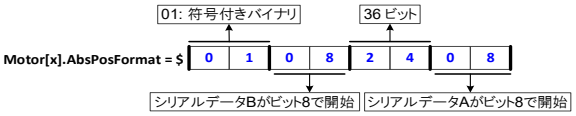
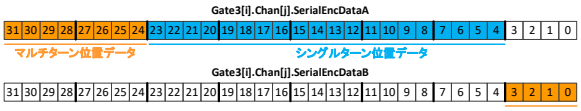
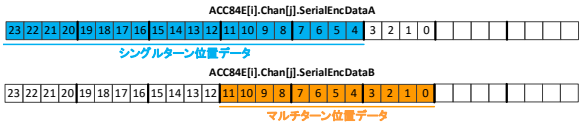
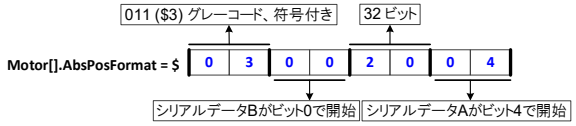
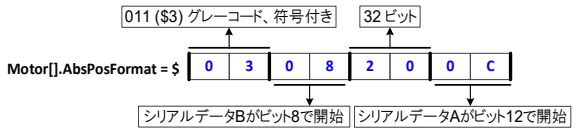
これらの設定が PMAC にダウンロードされたら、この機能は以下のように使用できます。

- ターミナルウィンドウから：#xHMZ はモータ x の絶対位置を読み出します
- プログラムから（PLC またはモーション）：HOMEZx はモータ x の絶対位置を読み出します
- または、Motor[x].PowerOnMode (= $2)のビット＃1 は、電源投入時に読み出した絶対位置を指定します。電源投入時にエンコーダ電源が利用可能であることを確認してください（電源が軸カードの外部にある場合）。

参考

グレイコード変換は、シリアルエンコーダコマンドワード Gate3[i].Chan[j].SerialEncCmd または AACC84E[i].Chan[j].SerialEncCmd にすでに実装されている場合は、ここでは省略してください。

例	Motor[x].AbsPosFormat の設定
17 ビットのバイナリシングルターン（またはリニアスケール）があり、シリアルデータ A レジスタの下位フィールドにマルチターン位置データがない絶対シリアルエンコーダ • Gate3  • ACC-84E  	• Gate3  • ACC-84E 
20 ビットのバイナリシングルターン（またはリニアスケール）があり、シリアルデータ A レジスタのビット#4 から開始するマルチターン位置データがない絶対シリアルエンコーダ • Gate3 • ACC-84E  	• Gate3  • ACC-84E 
シリアルデータ A と B レジスタに連続して配置された 36 ビットのシングルターン（またはリニアスケール）位置データがある絶対シリアルエンコーダ • Gate3  • ACC-84E  	• Gate3  • ACC-84E 
シリアルデータ A レジスタのビット#0 から始まり、ビット#28 まで継続的に拡張する、17 ビットのシングルターンおよび 12 ビットのマルチターン位置データがある 29 ビットのバイナリ絶対シリアルエンコーダ • Gate3  • ACC-84E  	• Gate3  • ACC-84E 

例	Motor[x].AbsPosFormat の設定
24 ビットのシングルターン位置データと 12 ビットのマルチターン位置データがある 36 ビットバイナリの絶対シリアルエンコーダ • Gate3  • ACC-84E  	• Gate3  • ACC-84E 
20 ビットのシングルターン位置データ(ビット#4 から開始)と 12 ビットのマルチターン位置データがある 32 ビットグレイコード絶対シリアルエンコーダ • Gate3  • ACC-84E  (注) グローバルコマンドワードでグレイコードがすでに設定されている場合は、ここで設定しないでください。	• Gate3  • ACC-84E 

参考

同じシリアルエンコーダデータレジスタで、シングルターンデータとマルチターンデータの間にギャップがある場合(連続していない場合)、ギャップをなくすため、位置データを手動で(PLC などで)アセンブルし、ユーザメモリレジスタ(Sys.Udata [i]など)に配置する必要があります。そのあと、絶対原点復帰の構造体要素をこのレジスタに適用できます。

絶対位相基準の設定

使用上の注意

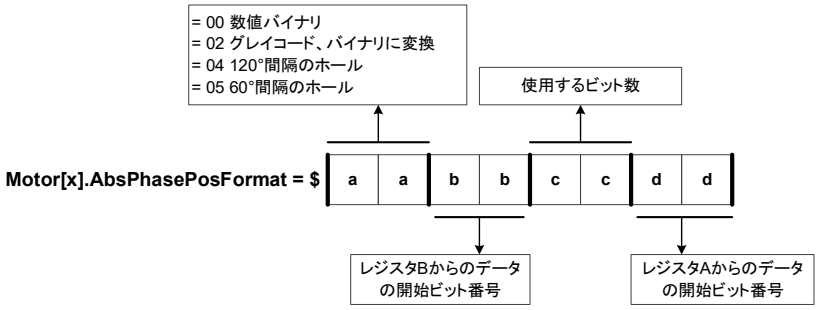
- 絶対位相基準を設定する前に、モータが正常に位相調整される必要があります。
- Motor[x].PhaseFindingDac、および Motor[x].PhaseFindingTime はゼロに等しくなければなりません。

ここでは、ホール効果センサと絶対シリアルエンコーダ付きのモータの絶対位相（転流）基準の設定について説明します。ここでは、以下の定義が使用されます。

名称	内容
ST	シングルターンシリアルロータリエンコーダの位置データ
NoOfPolePairs	ロータリブラシレスモータのペア極数
ECLmm	リニアモータの電気サイクル長さまたは磁気ピッチ（mm）（例：30.48mm）
RESmm	リニアエンコーダ分解能（mm）（例：1μm = 0.001mm）

絶対位相基準を設定するための主要な構造体要素を以下に示します。

構造体要素	内容と実例
Motor[x].pAbsPhasePos	**電源 ON 絶対位相位置ポインタ** ホールの場合、=Gate3[i].Chan[j].Status.a 絶対シリアルエンコーダの場合、=Gate3[i].Chan[j].SerialEncDataA.a
Motor[x].AbsPosSf	**絶対位置読み出しスケール値** ホールの場合、=±2048 / 12（UVW 遷移により） シリアル絶対ロータリエンコーダの場合、=2048 * NoOfPolePairs / 2ST シリアル絶対リニアエンコーダの場合、=2048 * RESmm / ECLmm
Motor[x].AbsPhasePosFormat	**電源 ON 絶対位相データフォーマット** 120°間隔のホールの場合、=$400030C 絶対シリアルエンコーダについては、以下の内容と実例（詳細）を参照してください。
Motor[x].AbsPhasePosOffset	**絶対位相位置基準** ホールの場合、= UVW の 1 <-> 3 遷移と交差する位相位置の値 絶対シリアルエンコーダの場合、=–Motor[x].AbsPhasePosForce * Motor[x].AbsPhasePosSf



これらの設定が PMAC にダウンロードされたら、この機能は以下のように使用できます。

- ターミナルウィンドウから：#x$はモータ x の絶対位置を読み出します
- PLC プログラムから：Motor[x].PhaseFindingStep = 1 は、モータ x の絶対位置を読み出します
- または、Motor[x].PowerOnMode (= $4)のビット#2 は、電源投入時の絶対位相基準を指定します。電源投入時にエンコーダ電源が利用可能であることを確認してください（電源が軸カードの外部にある場合）。

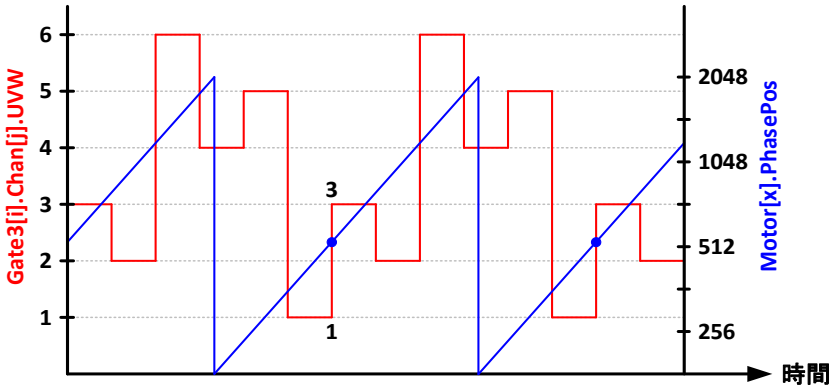
● ホール効果センサによる絶対位相基準

ここでは、従来の 120°ホール効果センサ間隔を使用した絶対位相基準の設定について説明します。

使用上の注意

- 絶対位相基準を設定する前に、モータが正常に位相調整される必要があります。
- Motor[x].PhaseFindingDac、および Motor[x].PhaseFindingTime はゼロに等しくなければなりません。

IDE プロットを使用して、モータを**正方向**（通常は低速）に移動しながら、構造体要素 Gate3[i].Chan[j].UVW および Motor[x].PhasePos を収集します。



このプロットは、2 つの重要なホール設定を示しています。

- UVW の遷移が 1 から 3 の場合、Motor[x].PhasePosSf= 2048 / 12
  UVW の遷移が 3 から 1 の場合、Motor[x].PhasePosSf= –2048 / 12

- 1<-> 3 遷移で、Motor[x].AbsPhasePosOffset = {Motor[x].PhasePos の値}

● 絶対値エンコーダによる絶対位相基準

使用上の注意

- 絶対位相基準を設定する前に、モータが正常に位相調整される必要があります。
- Motor[x].PhaseFindingDac、および Motor[x].PhaseFindingTime はゼロに等しくなければなりません。

絶対位相基準に使用できる最大ビット数は 32 です。フル位置データストリームを使用する必要はありません。簡単にするために、シングルターンデータだけを使用することをお勧めします。ただし、1つ重要なことは、使用する位置データの MSB（最上位ビット）に常に含めることです。

Motor[x].AbsPhasePosForce は単なる保持レジスタです。その値は、以下の絶対位相基準のテスト手順を実行することによって求められます。

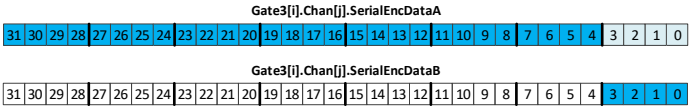
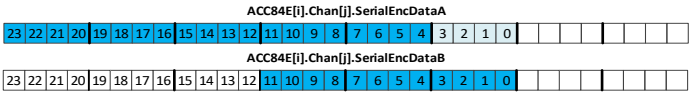
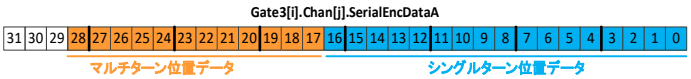
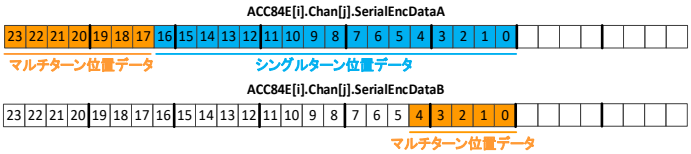
手順	コマンド	説明
1	#xK	モータが停止され、静止していることを確認してください。
2	Motor[x].IbBias = Motor[x].I2tSet/4	電流量（トルク）[Motor[x].I2tSet/4 - Motor[x].I2tSet]
3	#xOUT0	アンプを有効にします（これによりモータが移動します） モータが落ち着くのを待ちます（移動の停止）
4	Read Motor[x].AbsPhasePosForce	= {値}、以下の例を参照してください。
6	Motor[x].IbBias = 0 #xK	Motor[x].IbBias をリセットし、モータを停止します。

x はモータ番号を示します。

参考

この手順を行う中、モータが振動する可能性があり、Motor[x].IbBias を減らすことができます。あるいは、安全で可能であれば、モータを手で安定させることもできます。

例	絶対位相基準設定
1	17 ビットのバイナリシングルターン（またはリニアスケール）があり、シリアルデータ A レジスタの下位フィールドにマルチターン位置データがない絶対シリアルエンコーダ • Gate3 Gate3[i].Chan[j].SerialEncDataA 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 **16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0** ロータリ：Motor[x].AbsPhasePosSf = 2048 * NoOfPolePairs / 2^{17} リニア：Motor[x].AbsPhasePosSf = 2048 * RESmm / ECLmm Motor[x].pAbsPhasePos = Gate3[i].Chan[j].SerialEncDataA.a Motor[x].AbsPhasePosFormat = $00001100 Motor[x].AbsPhasePosrce = テスト中の{**Gate3[i].Chan[j].SerialEncDataA & $1FFFF**}の値 Motor[x].AbsPhasePosOffset = –Motor[x].AbsPhasePosForce * Motor[x].AbsPhasePosSf • ACC-84E ACC84E[i].Chan[j].SerialEncDataA 23 22 21 20 19 18 17 **16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0** ロータリ：Motor[x].AbsPhasePosSf = 2048 * NoOfPolePairs / 2^{17} リニア：Motor[x].AbsPhasePosSf = 2048 * RESmm / ECLmm Motor[x].pAbsPhasePos = ACC84E[i].Chan[j].SerialEncDataA.a Motor[x].AbsPhasePosFormat = $00001108 Motor[x].AbsPhasePosForce = テスト中の{**ACC84E[i].Chan[j].SerialEncDataA & $1FFFF**}の値 Motor[x].AbsPhasePosOffset = –Motor[x].AbsPhasePosForce * Motor[x].AbsPhasePosSf
2	20 ビットのバイナリシングルターン（またはリニアスケール）があり、シリアルデータ A レジスタのビット#4 から開始するマルチターン位置データがない絶対シリアルエンコーダ • Gate3 Gate3[i].Chan[j].SerialEncDataA 31 30 29 28 27 26 25 24 **23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4** 3 2 1 0 ロータリ：Motor[x].AbsPhasePosSf = 2048 * NoOfPolePairs / 2^{20} リニア：Motor[x].AbsPhasePosSf = 2048 * RESmm / ECLmm Motor[x].pAbsPhasePos = Gate3[i].Chan[j].SerialEncDataA.a Motor[x].AbsPhasePosFormat = $00001404 Motor [x] .AbsPhasePosForce = テスト中の{**(Gate3[i].Chan[j].SerialEncDataA & $FFFFF0) >> 4**}の値 Motor[x].AbsPhasePosOffset = –Motor[x].AbsPhasePosForce * Motor[x].AbsPhasePosSf • ACC-84E ACC84E[i].Chan[j].SerialEncDataA **23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4** 3 2 1 0 ロータリ：Motor[x].AbsPhasePosSf = 2048 * NoOfPolePairs / 2^{20} リニア：Motor[x].AbsPhasePosSf = 2048 * RESmm / ECLmm Motor[x].pAbsPhasePos = ACC84E[i].Chan[j].SerialEncDataA.a Motor[x].AbsPhasePosFormat = $0000140C Motor[x].AbsPhasePosForce = テスト中の{**(ACC84E[i].Chan[j].SerialEncDataA & $FFFFF0) >> 4**}の値 Motor[x].AbsPhasePosOffset = –Motor[x].AbsPhasePosForce * Motor[x].AbsPhasePosSf

例	絶対位相基準設定
3	シリアルデータ A と B レジスタに連続して配置された 36 ビットのシングルターン（またはリニアスケール）位置データがある絶対シリアルエンコーダ • Gate3  ロータリ：Motor[x].AbsPhasePosSf = 2048 * NoOfPolePairs / 2^{32} リニア：Motor[x].AbsPhasePosSf = 2048 * RESmm * 2^4 / ECLmm Motor[x].pAbsPhasePos = Gate3[i].Chan[j].SerialEncDataA.a Motor[x].AbsPhasePosFormat = $00002004（下位 4 ビットを無視し、上位 32 ビットを使用して MSB を含めます） Motor[x].AbsPhasePosForce = テスト中の{**Gate3[i].Chan[j].SerialEncDataA>>4+(Gate3[i].Chan[j].SerialEncDataB&$F)<<28**}の値 Motor[x].AbsPhasePosOffset = –Motor[x].AbsPhasePosForce * Motor[x].AbsPhasePosSf • ACC-84E  ロータリ：Motor[x].AbsPhasePosSf = 2048 * NoOfPolePairs / 2^{32} リニア：Motor[x].AbsPhasePosSf = 2048 * RESmm * 2^4 / ECLmm Motor[x].pAbsPhasePos = ACC84E[i].Chan[j].SerialEncDataA.a Motor[x].AbsPhasePosFormat = $0008200C（下位 4 ビットを無視し、上位 32 ビットを使用して MSB を含めます） Motor[x].AbsPhasePosForce = テスト中の {**ACC84E[i].Chan[j].SerialEncDataA>>4+(ACC84E[i].Chan[j].SerialEncDataB&$FFF)<<20**}の値 Motor[x].AbsPhasePosOffset = –Motor[x].AbsPhasePosForce * Motor[x].AbsPhasePosSf
4	シリアルデータ A レジスタのビット#0 から始まり、ビット#28 まで継続的に拡張する、17 ビットのシングルターンおよび 12 ビットのマルチターン位置データがある 29 ビットのバイナリ絶対シリアルエンコーダ • Gate3  Motor[x].AbsPhasePosSf = 2048 * NoOfPolePairs / 2^{17} Motor[x].pAbsPhasePos = Gate3[i].Chan[j].SerialEncDataA.a Motor[x].AbsPhasePosFormat = $00001100（レジスタ A で 17 ビットシングルターンデータだけを使用） Motor[x].AbsPhasePosForce = テスト中の{**Gate3[i].Chan[j].SerialEncDataA & $1FFFF**}の値 Motor[x].AbsPhasePosOffset = –Motor[x].AbsPhasePosForce * Motor[x].AbsPhasePosSf • ACC-84E  ロータリ：Motor[x].AbsPhasePosSf = 2048 * NoOfPolePairs / 2^{17} Motor[x].pAbsPhasePos = ACC84E[i].Chan[j].SerialEncDataA.a Motor[x].AbsPhasePosFormat = $00001108（レジスタ A で 17 ビットのシングルターンデータだけを使用） Motor[x].AbsPhasePosForce = テスト中の{**ACC84E[i].Chan[j].SerialEncDataA & $1FFFF**}の値 Motor[x].AbsPhasePosOffset = –Motor[x].AbsPhasePosForce * Motor[x].AbsPhasePosSf

A-2-5 モータ動作方向の反転

安全上の要点

- モータが通電している間は、以下の設定変更を適用しないでください。
- 運転方向変更を実施した後は、モータの位相を再調整する必要があります。
- モータの動作方向を変更する場合は、それに応じてオーバートラベルリミットとソフトリミットを変更する必要があります。

多くの場合、特定のアプリケーションでは、設定が完了した後にモータの動作方向を反転することが望まれます。ブラシレスモータの場合、この変更を適用するために必要な構造体要素の「新しい」設定を以下に示します。これはエンコーダの種類によって異なります。

パルスおよびアナログ正弦波エンコーダ付きのモータ

構造体要素	新しい値
Motor[x].PhaseOffset	= –現在値
Motor[x].PwmSf	= –現在値
Gate3[i].Chan[j].EncCtrl	= 3 または 7（反対のデコード値）
Motor[x].AbsPhasePosSf（ホール使用の場合）	= –現在値
Motor[x].AbsPhasePosOffset（ホール使用の場合）	= 2048 –現在値

シリアルエンコーダ付きのモータ

構造体要素	新しい値
Motor[x].PhaseOffset	= –現在値
Motor[x].PwmSf	= –現在値
Motor[x].PhasePosSf	= –現在値
EncTable[n].ScaleFactor	= –現在値
Motor[x].PhasePosSf（絶対値エンコーダの場合）	= –現在値
Motor[x].AbsPhasePosSf（絶対値エンコーダの場合）	= –現在値
Motor[x].AbsPhasePosOffset（絶対値エンコーダの場合）	= 2048 –現在値

A-3　Gate3「スクリプト」モータ設定サンプル

ここでは、参照用として、Power PMAC ACC-24E3、ACC-84E、および CK3M　AX ユニットありのブラシレスモータの設定サンプルを提供します。

安全上の要点

ここで提供するサンプルは参照用であり、モータを直ちに動作させない可能性があります。以下の項目に注意する必要があります。

- エンコーダとモータの仕様パラメータ
- 電流ループチューニングを実行する必要があります
- 初期位相を実行する必要があります
- 開ループテストを実行する必要があります
- 位置ループチューニングを実行する必要があります
- 絶対位置の設定（該当する場合）を確認する必要があります
- 絶対位相基準の設定（該当する場合）を確認する必要があります
- 一般的なモーションパラメータ（ジョグ速度など）を確認する必要があります

参考

簡単にするために、ここのサンプルは、Motor#1、Gate3、index0、channel0 を参照しています。

A-3-1　パルスエンコーダ付きロータリサーボモータ

ハードウェア構成

項目	内容
コントローラ	このサンプルは、UMAC ACC-24E3 および CK3M AX に適用
アンプ	形 CK3A-G310L（三相、AC208V、主電源電圧）
モータ	ロータリサーボモータ、AC200V、11/23A、4 極ペア
エンコーダ	パルス、2,000 ライン（x4 デコードで 1 回転あたり 8,000 カウント）

システムゲート

```
Sys.WpKey = $AAAAAAAA
Sys.MaxCoords = 2
Sys.MaxMotors = 5

Gate3[0].PhaseFreq = 10000
Gate3[0].ServoClockDiv = 1
Sys.RtIntPeriod = 1
Sys.ServoPeriod = 1000 * (Gate3[0].ServoClockDiv + 1) / Gate3[0].PhaseFreq
Sys.PhaseOverServoPeriod = 1 / (Gate3[0].ServoClockDiv + 1)

Gate3[0].Chan[0].PwmFreqMult = 1
Gate3[0].Chan[1].PwmFreqMult = 1
Gate3[0].Chan[2].PwmFreqMult = 1
Gate3[0].Chan[3].PwmFreqMult = 1

Gate3[0].Chan[0].PackInData = 0
Gate3[0].Chan[1].PackInData = 0
Gate3[0].Chan[2].PackInData = 0
Gate3[0].Chan[3].PackInData = 0

Gate3[0].Chan[0].PackOutData = 0
Gate3[0].Chan[1].PackOutData = 0
Gate3[0].Chan[2].PackOutData = 0
Gate3[0].Chan[3].PackOutData = 0

Gate3[0].AdcAmpStrobe = $901001
Gate3[0].AdcAmpClockDiv = 5
Gate3[0].AdcAmpHeaderBits = 4
Gate3[0].Chan[0].PwmDeadTime = 3 / 0.0533
```

書き込み保護の無効
最大座標系+1（効率性）
最大モータ数+1（効率性）

10kHz のフェーズ周波数
5kHz のサーボ周波数
2.5kHz の RTI 周波数
典型的な計算
典型的な計算

CH1～4 10kHz の PWM 周波数

CK3A の場合、CH1～4 入力データが 0 である

CK3A の場合、CH1～4 出力データが 0 である

CH1～4 ストローブワード（CK3A の場合）
CH1～4 3.125MHz ADC クロック
CH1～4 は、4 である（CK3A の場合）
G310 の場合、CH1 が最小 3μSEC である

エンコーダ変換テーブル（ECT）

```
EncTable[1].Type = 1
EncTable[1].pEnc = Gate3[0].Chan[0].ServoCapt.a
EncTable[1].pEnc1 = Sys.Pushm
EncTable[1].index1 = 0
EncTable[1].index2 = 0
EncTable[1].index3 = 0
EncTable[1].index4 = 0
EncTable[1].index5 = 0
EncTable[1].index6 = 0
EncTable[1].ScaleFactor = 1 / 256
EncTable[1].MaxDelta = 0

EncTable[2].Type = 0
```

典型的なパルスエンコーダエントリ

ECT の終了（効率性のため）

モータの設定と試運転

```
GLOBAL Ck3a1MaxAdc = 31.470                                  CK3A の最大 ADC
GLOBAL Mtr1CtsPerRev = 8000                                  1 回転あたりのカウント数
GLOBAL Mtr1PolePairs = 4                                     極ペア数
GLOBAL Mtr1MaxRmsCur = 20                                    最大 RMS 電流(MTR>CK3A)
GLOBAL Mtr1RatedRmsCur = 10                                  定格 RMS 電流(MTR>CK3A)
GLOBAL Mtr1TimeAtMaxCur = 2                                  最大電流での許容可能時間

Gate3[0].Chan[0].EncCtrl = 7                                 エンコーダカウント方向
Motor[1].ServoCtrl = 1                                       チャネルをアクティブ
Motor[1].PosUnit = 11                                        IDE ウィンドウ単位:度(deg)
Motor[1].PosSf = 360 / Mtr1CtsPerRev                         スケール位置:度(deg)
Motor[1].Pos2Sf = 360 / Mtr1CtsPerRev                        スケール速度:度(deg)
Motor[1].pLimits = Gate3[0].Chan[0].Status.a                 リミット有効、=0 の場合、無効

Motor[1].AbortTa = 150                                       減速度中断、150MSEC
Motor[1].AbortTs = 0                                         S 字カーブ中断、なし

Motor[1].JogTa = 300                                         ジョグ加速度、300MSEC
Motor[1].JogTs = 0                                           ジョグ S 字カーブ、なし
Motor[1].JogSpeed = 4.5                                      ジョグ速度、4.5 度/MSEC

Motor[1].FatalFeLimit = 5                                    フォローエラーリミット:5 度
Motor[1].WarnFeLimit = 0                                     警告フォローエラーリミット、なし

Motor[1].InPosBand = 0.05                                    位置帯:0.05 度
Motor[1].InPosTime = 20 / Sys.ServoPeriod                    位置時間:20MSEC

Motor[1].pPhaseEnc = Gate3[0].Chan[0].PhaseCapt.a            直交の典型例
Motor[1].PhaseCtrl = 4                                       ブラシレスモータの典型例
Motor[1].PhaseEncLeftShift = 0                               直交、なし
Motor[1].PhaseEncRightShift = 0                              直交、なし
Motor[1].PhasePosSf = 2048 * Mtr1PolePairs / (256 * Mtr1CtsPerRev)   直交の典型例

Motor[1].AmpFaultLevel = 1                                   CK3A の場合は、=1
Motor[1].AdcMask = $FFFF0000                                 CK3A の場合は、=$FFFF0000
Motor[1].PhaseOffset = 683                                   ブラシレスモータの典型例
Motor[1].PwmSf = 0.95 * 16384                                MTR 電圧>入力の場合の典型例

Motor[1].MaxDac = Mtr1MaxRmsCur * 32768 * COSD(30) * SQRT(2) / Ck3a1MaxAdc   ブラシレスの典型的な I2T
Motor[1].I2TSet = Mtr1RatedRmsCur * 32768 * COSD(30) * SQRT(2) / Ck3a1MaxAdc   ...
Motor[1].I2tTrip =(POW(Motor[1].MaxDac,2)-POW(Motor[1].I2TSet,2))*Mtr1TimeAtMaxCur   ...

Motor[1].CurrentNullPeriod = 0                               CK3A の場合は、=0
Motor[1].IaBias = 0                                          正常動作中、=0
Motor[1].IbBias = 0                                          正常動作中、=0

Motor[1].IiGain = 0                                          電流ループゲイン
Motor[1].IpfGain = 0                                         (調整が必要)
Motor[1].IpbGain = 0                                         ...

Motor[1].Servo.Kp = 0                                        位置サーボループゲイン
Motor[1].Servo.Kvfb = 0                                      (調整が必要)
Motor[1].Servo.Kvifb = 0                                     ...
Motor[1].Servo.Kvff = 0                                      ...
Motor[1].Servo.Kviff = 0                                     ...
Motor[1].Servo.Ki = 0                                        ...
Motor[1].Servo.Kaff = 0                                      ...
Motor[1].Servo.Kfff = 0                                      ...
Motor[1].Servo.SwZvInt = 0                                   ...

Motor[1].PhaseFindingTime = 8000 / (2 * Sys.ServoPeriod * (Sys.RtIntPeriod + 1))   ステッパー位相(8 秒)
Motor[1].PhaseFindingDac = 0.8 * Motor[1].I2tSet             I2TSET の 80%
Motor[1].AbsPhasePosOffset = 2048 / 5                        典型的な設定
Motor[1].PowerOnMode = 0                                     電源投入時の絶対位相がなし
```

A-3-2　オムロン G5 サーボモータ

ハードウェア構成

項目	内容
コントローラ	この設定サンプルは、CK3MAX または UMAC　ACC-24E3 に適用
アンプ	形 CK3A-G305L（AC110V 主電源電圧）
モータ	G5 サーボモータ形 R88M-K40030H-S2、AC200V、2.4/10.2A、5 極ペア
エンコーダ	1 回転あたり G5 20 ビットアブソリュート（初期マーク後）

システムゲート

```
Sys.WpKey = $AAAAAAAA
Sys.MaxCoords = 2
Sys.MaxMotors = 5

Gate3[0].PhaseFreq = 10000
Gate3[0].ServoClockDiv = 1
Sys.RtIntPeriod = 1
Sys.ServoPeriod = 1000 * (Gate3[0].ServoClockDiv + 1) / Gate3[0].PhaseFreq
Sys.PhaseOverServoPeriod = 1 / (Gate3[0].ServoClockDiv + 1)

Gate3[0].Chan[0].PwmFreqMult = 1
Gate3[0].Chan[1].PwmFreqMult = 1
Gate3[0].Chan[2].PwmFreqMult = 1
Gate3[0].Chan[3].PwmFreqMult = 1

Gate3[0].Chan[0].PackInData = 0
Gate3[0].Chan[1].PackInData = 0
Gate3[0].Chan[2].PackInData = 0
Gate3[0].Chan[3].PackInData = 0

Gate3[0].Chan[0].PackOutData = 0
Gate3[0].Chan[1].PackOutData = 0
Gate3[0].Chan[2].PackOutData = 0
Gate3[0].Chan[3].PackOutData = 0

Gate3[0].AdcAmpStrobe = $901001
Gate3[0].AdcAmpClockDiv = 5
Gate3[0].AdcAmpHeaderBits = 4
Gate3[0].Chan[0].PwmDeadTime = 2 / 0.0533

Gate3[0].SerialEncCtrl = $01000008
Gate3[0].Chan[0].SerialEncCmd = $521000
Gate3[0].Chan[0].SerialEncEna = 1
```

書き込み保護の無効
最大座標系+1（効率性）
最大モータ数+1（効率性）

10kHz のフェーズ周波数
5kHz のサーボ周波数
2.5kHz の RTI 周波数
典型的な計算
典型的な計算

CH1～4 10kHz の PWM 周波数

CK3A の場合、CH1～4 入力データが 0 である

CK3A の場合、CH1～4 出力データが 0 である

CH1～4 ストローブワード（CK3A の場合）
CH1～4 3.125MHz ADC クロック
CH1～4 は、4 である（CK3A の場合）
G310 の場合、CH1 が最小 3μSEC である

CH1～4 シリアルエンコーダ制御 20 ビット G5
CH1～4 シリアルエンコーダコマンド
CH1～4 シリアルエンコーダ有効

エンコーダ変換テーブル（ECT）

```
EncTable[1].Type = 1
EncTable[1].pEnc = Gate3[0].Chan[0].SerialEncDataA.a
EncTable[1].pEnc1 = Sys.Pushm
EncTable[1].index1 = 12
EncTable[1].index2 = 0
EncTable[1].index3 = 0
EncTable[1].index4 = 0
EncTable[1].index5 = 0
EncTable[1].index6 = 0
EncTable[1].ScaleFactor = 1 / EXP2(EncTable[1].index1)
EncTable[1].MaxDelta = 0

EncTable[2].Type = 0
```

32 ビットの読み出し

ロールオーバーの場合は、MSB への 12 ビット左シフト

$1/2^{INDEX1}$ スケールから LSB へ

ECT の終了（効率性のため）

モータの設定と試運転

```
GLOBAL Ck3a1MaxAdc = 15.735
GLOBAL Mtr1SingleTurn = 20
GLOBAL Mtr1CtsPerRev = EXP2(Mtr1SingleTurn)
GLOBAL Mtr1PolePairs = 5
GLOBAL Mtr1MaxRmsCur = 10
GLOBAL Mtr1RatedRmsCur = 2.5
GLOBAL Mtr1TimeAtMaxCur = 2

Motor[1].ServoCtrl = 1
Motor[1].PosUnit = 11
Motor[1].PosSf = 360 / Mtr1CtsPerRev
Motor[1].Pos2Sf = 360 / Mtr1CtsPerRev
Motor[1].pLimits = Gate3[0].Chan[0].Status.a

Motor[1].AbortTa = 100
Motor[1].AbortTs = 0

Motor[1].JogTa = 100
Motor[1].JogTs = 0
Motor[1].JogSpeed = 3

Motor[1].FatalFeLimit = 1
Motor[1].WarnFeLimit = 0

Motor[1].InPosBand = 0.050
Motor[1].InPosTime = 5 / Sys.ServoPeriod

Motor[1].pPhaseEnc = Gate3[0].Chan[0].SerialEncDataA.a
Motor[1].PhaseCtrl = 4
Motor[1].PhaseEncLeftShift = 12
Motor[1].PhaseEncRightShift = 0
Motor[1].PhasePosSf = 2048 * Mtr1PolePairs / EXP2(32)

Motor[1].AmpFaultLevel = 1
Motor[1].AdcMask = $FFFF0000
Motor[1].PhaseOffset = 683
Motor[1].PwmSf = 0.95 * 16384

Motor[1].MaxDac = Mtr1MaxRmsCur * 32768 * COSD(30) * SQRT(2) / Ck3a1MaxAdc
Motor[1].I2TSet = Mtr1RatedRmsCur * 32768 * COSD(30) * SQRT(2) / Ck3a1MaxAdc
Motor[1].I2tTrip =(POW(Motor[1].MaxDac,2)-POW(Motor[1].I2TSet,2))*Mtr1TimeAtMaxCur

Motor[1].CurrentNullPeriod = 0
Motor[1].IaBias = 0
Motor[1].IbBias = 0

Motor[1].IiGain = 0.144
Motor[1].IpfGain = 2.7
Motor[1].IpbGain = 0

Motor[1].Servo.Kp = 1200
Motor[1].Servo.Kvfb = 13000
Motor[1].Servo.Kvifb = 0
Motor[1].Servo.Kvff = 13000
Motor[1].Servo.Kviff = 0
Motor[1].Servo.Ki = 0.016
Motor[1].Servo.Kaff = 80000
Motor[1].Servo.Kfff = 0
Motor[1].Servo.SwZvInt = 0

Motor[1].pAbsPos = Gate3[0].Chan[0].SerialEncDataA.a
Motor[1].AbsPosFormat = $1400
Motor[1].AbsPosSF = Motor[1].PosSf
Motor[1].HomeOffset = 0

Motor[1].pAbsPhasePos = Gate3[0].Chan[0].SerialEncDataA.a
Motor[1].AbsPhasePosFormat = $1400
Motor[1].AbsPhasePosSf = 2048 * Mtr1PolePairs / Mtr1CtsPerRev
//Motor[1].AbsPhasePosForce = Gate3[0].Chan[0].SerialEncDataA & $FFFFF
Motor[1].AbsPhasePosForce = 69614
Motor[1].AbsPhasePosOffset =-Motor[1].AbsPhasePosForce * Motor[1].AbsPhasePosSf
Motor[1].PowerOnMode = 0
```

CK3A の最大 ADC
シングルターン ST ビット
1 回転あたりのカウント数=2ST
ペア極数
最大 RMS 電流（CK3A<MTR）
定格 RMS 電流（MTR<CK3A）
最大電流での許容可能時間

チャネルをアクティブ
IDE ウィンドウ単位：度(deg)
スケール位置：度(deg)
スケール速度：度(deg)
リミット有効、=0 の場合、無効

減速度の中止、100MSEC
S 字カーブ中断、なし

ジョグ加速度、100MSEC
ジョグ S 字カーブ、なし
ジョグ速度、3 度/MSEC

フォローエラーリミット：1 度
警告フォローエラーリミット、なし

位置帯：[deg]
位置時間：5MSEC

シリアルエンコーダ
ブラシレスモータの典型例
MSB に左シフト
右シフトなし
=2048*極ペア/$2^{ST+左シフト}$

CK3A の場合は、=1
CK3A の場合は、=$FFFF0000
ブラシレスモータの典型例
MTR 電圧>入力の場合の典型例

ブラシレスの典型的な I2T
…
…

CK3A の場合は、=0
正常動作中、=0
正常動作中、=0

電流ループゲイン
…
…

位置サーボループゲイン
…
…
…
…
…
…
…
…

絶対シリアルエンコーダの典型例
符号なし ST ビット
位置スケールと同じ
ユーザ設定可能な位置オフセット

絶対シリアルエンコーダの典型例
絶対位相には ST データを使用
=2048*モータ極ペア/2ST
ST ビットをマスクし、値をテスト
テスト値を保存
典型的な設定
絶対位相読み出し時の電源 ON がなし

モータ/エンコーダアセンブリ用特別 PLC

この G5 エンコーダに初めて電力が供給されるときは、絶対位置(および位相基準)を読み出す前に、インデックスマークに移動するための特別な手順が必要です。以下は、この手順を実行するサンプル PLC です。

```
GLOBAL PhaseHomeFlag = 0

OPEN PLC PhaseHomePLC
IF(PhaseHomeFlag == 1)
{
 LOCAL EncStatusWord
 LOCAL StepAngle = 1
 LOCAL StepMag = 1000

 // エンコーダステータスワードの読み出し
 EncStatusWord = (Gate3[0].Chan[0].SerialEncDataA & $F00000) >> 20
 CALL DelayTimer.msec(10)

 // 初期化されていません?
 IF(EncStatusWord > 0)
 {
     WHILE(EncStatusWord > 0 && Motor[1].AmpFault == 0)
     {
         IF(Motor[1].AmpEna == 0)
         {
             COUT 1:0
             WHILE(Motor[1].AmpEna == 0 && Motor[1].AmpFault == 0){}
         }
         Motor[1].IaBias = StepMag * SIND(StepAngle + 120)
         Motor[1].IbBias = StepMag * SIND(StepAngle)
         EncStatusWord = (Gate3[0].Chan[0].SerialEncDataA & $F00000) >> 20
         StepAngle ++
     }
     CALL DelayTimer.msec(50)

     Motor[1].IaBias = 0
     Motor[1].IbBias = 0
     KILL 1
     CALL DelayTimer.msec(50)
 }

 // 絶対位相の読み出し
 Motor[1].PhaseFindingStep = 1
 CALL DelayTimer.msec(100)
 WHILE(Motor[1].PhaseFound == 0){}

 // 絶対位置の読み出し
 HOMEZ 1
 CALL DelayTimer.msec(100)
 WHILE(Motor[1].HomeComplete == 0){}

 PhaseHomeFlag = 0
}
CLOSE
```

A-3-3　オムロン 1S サーボモータ（ACC-84E）

ハードウェア構成

項目	内容
コントローラ	この設定サンプルは、ACC-84E ありの UMAC ACC-24E3（1S プロトコル）に適用
アンプ	形 CK3A-G305L（AC110V 主電源電圧）
モータ	1S サーボモータ形 R88M-1M10030S、AC100V、1.5/4.7A
エンコーダ	1S シリアルアブソリュート 23 ビットシングルターン、16 ビットマルチターン

システムゲート

```
Sys.WpKey = $AAAAAAAA
Sys.MaxCoords = 2
Sys.MaxMotors = 5

Gate3[0].PhaseFreq = 10000
Gate3[0].ServoClockDiv = 1
Sys.RtIntPeriod = 1
Sys.ServoPeriod = 1000 * (Gate3[0].ServoClockDiv + 1) / Gate3[0].PhaseFreq
Sys.PhaseOverServoPeriod = 1 / (Gate3[0].ServoClockDiv + 1)

Gate3[0].Chan[0].PwmFreqMult = 1
Gate3[0].Chan[1].PwmFreqMult = 1
Gate3[0].Chan[2].PwmFreqMult = 1
Gate3[0].Chan[3].PwmFreqMult = 1

Gate3[0].Chan[0].PackInData = 0
Gate3[0].Chan[1].PackInData = 0
Gate3[0].Chan[2].PackInData = 0
Gate3[0].Chan[3].PackInData = 0

Gate3[0].Chan[0].PackOutData = 0
Gate3[0].Chan[1].PackOutData = 0
Gate3[0].Chan[2].PackOutData = 0
Gate3[0].Chan[3].PackOutData = 0

Gate3[0].AdcAmpStrobe = $901001
Gate3[0].AdcAmpClockDiv = 5
Gate3[0].AdcAmpHeaderBits = 4
Gate3[0].Chan[0].PwmDeadTime = 2 / 0.0533

ACC84E[0].SerialEncCtrl = $E
ACC84E[0].Chan[0].SerialEncCmd = $1400
```

書き込み保護の無効
最大座標系+1（効率性）
最大モータ数+1（効率性）

10kHz のフェーズ周波数
5kHz のサーボ周波数
2.5kHz の RTI 周波数
典型的な計算
典型的な計算

CH1～4 10kHz の PWM 周波数

CK3A の場合、CH1～4 入力データが 0 である

CK3A の場合、CH1～4 出力データが 0 である

CH1～4 ストローブワード（CK3A の場合）
CH1～4 3.125MHz ADC クロック
CH1～4 は、4 である（CK3A の場合）
G310 の場合、CH1 が最小 3μSEC である

1S の場合、ACC-84E CH1～4
1S の場合、ACC-84E CH1 シリアルエンコーダコマンド

エンコーダ変換テーブル（ECT）

```
EncTable[1].Type = 1
EncTable[1].pEnc = Gate3[0].Chan[0].SerialEncDataA.a
EncTable[1].pEnc1 = Sys.Pushm
EncTable[1].index1 = 8
EncTable[1].index2 = 8
EncTable[1].index3 = 0
EncTable[1].index4 = 0
EncTable[1].index5 = 0
EncTable[1].index6 = 0
EncTable[1].ScaleFactor = 1 / EXP2(EncTable[1].index1)
EncTable[1].MaxDelta = 0

EncTable[2].Type = 0
```

32 ビットの読み出し

ACC-84（左シフト）
ACC-84（右シフト）

1/2^{INDEX1} スケールから LSB へ

ECT の終了（効率性のため）

モータの設定と試運転

```
GLOBAL Ck3a1MaxAdc = 15.735
GLOBAL Mtr1SingleTurn = 23
GLOBAL Mtr1MultiTurn = 16
GLOBAL Mtr1CtsPerRev = EXP2(Mtr1SingleTurn)
GLOBAL Mtr1PolePairs = 5
GLOBAL Mtr1MaxRmsCur = 4.7
GLOBAL Mtr1RatedRmsCur = 1.5
GLOBAL Mtr1TimeAtMaxCur = 2

Motor[1].ServoCtrl = 1
Motor[1].PosUnit = 11
Motor[1].PosSf = 360 / Mtr1CtsPerRev
Motor[1].Pos2Sf = 360 / Mtr1CtsPerRev
Motor[1].pLimits = Gate3[0].Chan[0].Status.a

Motor[1].AbortTa = 100
Motor[1].AbortTs = 0

Motor[1].JogTa = 100
Motor[1].JogTs = 0
Motor[1].JogSpeed = 3

Motor[1].FatalFeLimit = 1
Motor[1].WarnFeLimit = 0

Motor[1].InPosBand = 0.050
Motor[1].InPosTime = 5 / Sys.ServoPeriod

Motor[1].pPhaseEnc = ACC84E[0].Chan[0].SerialEncDataA.a
Motor[1].PhaseCtrl = 4
Motor[1].PhaseEncLeftShift = 9
Motor[1].PhaseEncRightShift = 8
Motor[1].PhasePosSf = 2048 * Mtr1PolePairs / EXP2(32)

Motor[1].AmpFaultLevel = 1
Motor[1].AdcMask = $FFFF0000
Motor[1].PhaseOffset = 683
Motor[1].PwmSf = 0.95 * 16384

Motor[1].MaxDac = Mtr1MaxRmsCur * 32768 * COSD(30) * SQRT(2) / Ck3a1MaxAdc
Motor[1].I2TSet = Mtr1RatedRmsCur * 32768 * COSD(30) * SQRT(2) / Ck3a1MaxAdc
Motor[1].I2tTrip =(POW(Motor[1].MaxDac,2)-POW(Motor[1].I2TSet,2))*Mtr1TimeAtMaxCur

Motor[1].CurrentNullPeriod = 0
Motor[1].IaBias = 0
Motor[1].IbBias = 0

Motor[1].IiGain = 0
Motor[1].IpfGain = 0
Motor[1].IpbGain = 0

Motor[1].Servo.Kp = 0
Motor[1].Servo.Kvfb = 0
Motor[1].Servo.Kvifb = 0
Motor[1].Servo.Kvff = 0
Motor[1].Servo.Kviff = 0
Motor[1].Servo.Ki = 0
Motor[1].Servo.Kaff = 0
Motor[1].Servo.Kfff = 0
Motor[1].Servo.SwZvInt = 0

Motor[1].pAbsPos = ACC84E[0].Chan[0].SerialEncDataA.a
Motor[1].AbsPosFormat = $01082708
Motor[1].AbsPosSF = Motor[1].PosSf
Motor[1].HomeOffset = 0

Motor[1].pAbsPhasePos = ACC84E[0].Chan[0].SerialEncDataA.a
Motor[1].AbsPhasePosFormat = $1708
Motor[1].AbsPhasePosSf = 2048 * Mtr1PolePairs / Mtr1CtsPerRev
//Motor[1].AbsPhasePosForce = ACC84E[0].Chan[0].SerialEncDataA & $7FFFFF
Motor[1].AbsPhasePosForce = 1269945
Motor[1].AbsPhasePosOffset =-Motor[1].AbsPhasePosForce * Motor[1].AbsPhasePosSf
Motor[1].PowerOnMode = $6
```

CK3A の最大 ADC
シングルターン ST ビット
マルチターン MT ビット
1 回転あたりのカウント数=2ST
ペア極数
最大 RMS 電流(MTR<CK3A)
定格 RMS 電流(MTR<CK3A)
最大電流での許容可能時間

チャネルをアクティブ
IDE ウィンドウ単位:度(deg)
スケール位置:度(deg)
スケール速度:度(deg)
リミット有効、=0 の場合、無効

減速度の中止、100MSEC
S 字カーブ中断、なし

ジョグ加速度、100MSEC
ジョグ S 字カーブ、なし
ジョグ速度、3 度/MSEC

フォローエラーリミット:1 度
警告フォローエラーリミット、なし

位置帯: [deg]
位置時間:5MSEC

ACC-84 シリアルエンコーダ
ブラシレスモータの典型例
MSB に左シフト
典型的な ACC-84
=2048*極ペア/$2^{ST+左シフト}$

CK3A の場合は、=1
CK3A の場合は、=$FFFF0000
ブラシレスモータの典型例
MTR 電圧>入力の場合の典型例

ブラシレスの典型的な I2T
…
…

CK3A の場合は、=0
正常動作中、=0
正常動作中、=0

電流ループゲイン
(調整が必要)
…

位置サーボループゲイン
(調整が必要)
…
…
…
…
…
…
…

絶対シリアルエンコーダの典型例
符号付き ST+MT ビット
位置スケールと同じ
ユーザ設定可能な位置オフセット

絶対シリアルエンコーダの典型例
絶対位相には ST データを使用
=2048*モータ極ペア/2ST
ST ビットをマスクし、値をテスト
テスト値を保存
典型的な設定
電源 ON 絶対位相と位置

A-3-4 正弦波エンコーダ付きのリニアサーボモータ

ハードウェア構成

項目	内容
コントローラ	このサンプルは、UMAC ACC-24E3（標準正弦波）および CK3M AX（正弦波）に適用
アンプ	形 CK3A-G310L（三相、AC208V、主電源電圧）
モータ	リニアサーボモータ、AC200V、2.6/8.3A、60.96mm 電気サイクル長さ
エンコーダ	正弦波、20μm（分解能）の正弦波サイクル長さ

システムゲート

```
Sys.WpKey = $AAAAAAAA
Sys.MaxCoords = 2
Sys.MaxMotors = 5

Gate3[0].PhaseFreq = 10000
Gate3[0].ServoClockDiv = 1
Sys.RtIntPeriod = 1
Sys.ServoPeriod = 1000 * (Gate3[0].ServoClockDiv + 1) / Gate3[0].PhaseFreq
Sys.PhaseOverServoPeriod = 1 / (Gate3[0].ServoClockDiv + 1)

Gate3[0].Chan[0].PwmFreqMult = 1
Gate3[0].Chan[1].PwmFreqMult = 1
Gate3[0].Chan[2].PwmFreqMult = 1
Gate3[0].Chan[3].PwmFreqMult = 1

Gate3[0].Chan[0].PackInData = 0
Gate3[0].Chan[1].PackInData = 0
Gate3[0].Chan[2].PackInData = 0
Gate3[0].Chan[3].PackInData = 0

Gate3[0].Chan[0].PackOutData = 0
Gate3[0].Chan[1].PackOutData = 0
Gate3[0].Chan[2].PackOutData = 0
Gate3[0].Chan[3].PackOutData = 0

Gate3[0].AdcAmpStrobe = $901001
Gate3[0].AdcAmpClockDiv = 5
Gate3[0].AdcAmpHeaderBits = 4
Gate3[0].Chan[0].PwmDeadTime = 3 / 0.0533

Gate3[0].EncClockDiv = 3
Gate3[0].AdcEncClockDiv = 3
//Gate3[0].AdcEncCtrl = $3FFFC000
```

書き込み保護の無効
最大座標系+1（効率性）
最大モータ数+1（効率性）

10kHz のフェーズ周波数
5kHz のサーボ周波数
2.5kHz の RTI 周波数
典型的な計算
典型的な計算

CH1～4 10kHz の PWM 周波数

CK3A の場合、CH1～4 入力データが 0 である

CK3A の場合、CH1～4 出力データが 0 である

CH1～4 ストローブワード（CK3A の場合）
CH1～4 3.125MHz ADC クロック
CH1～4 は、4 である（CK3A の場合）
G310 の場合、CH1 が最小 3μSEC である

エンコーダサンプリング 12.5MHz
エンコーダ ADC サンプリング 12.5MHz
CK3M AX の場合、コメント解除

エンコーダ変換テーブル（ECT）

```
EncTable[1].Type = 1
EncTable[1].pEnc = Gate3[0].Chan[0].ServoCapt.a
EncTable[1].pEnc1 = Sys.Pushm
EncTable[1].index1 = 0
EncTable[1].index2 = 0
EncTable[1].index3 = 0
EncTable[1].index4 = 0
EncTable[1].index5 = 0
EncTable[1].index6 = 0
EncTable[1].ScaleFactor = 1
EncTable[1].MaxDelta = 0

EncTable[2].Type = 0
```

典型的な正弦波エンコーダエントリ

ECT の終了（効率性のため）

モータの設定と試運転

```
GLOBAL Ck3a1MaxAdc = 31.470
GLOBAL Mtr1Res = 0.020
GLOBAL Mtr1CtsPerMm = 16384 / Mtr1Res
GLOBAL Mtr1Ecl = 60.96
GLOBAL Mtr1MaxRmsCur = 8.3
GLOBAL Mtr1RatedRmsCur = 2.6
GLOBAL Mtr1TimeAtMaxCur = 2

Motor[1].EncType = 6
Gate3[0].Chan[0].AtanEna = 1
Gate3[0].Chan[0].EncCtrl = 7

Motor[1].ServoCtrl = 1
Motor[1].PosUnit = 3
Motor[1].PosSf = 1 / Mtr1CtsPerMm
Motor[1].Pos2Sf = 1 / Mtr1CtsPerMm
Motor[1].pLimits = Gate3[0].Chan[0].Status.a

Motor[1].AbortTa = 100
Motor[1].AbortTs = 0

Motor[1].JogTa = 100
Motor[1].JogTs = 0
Motor[1].JogSpeed = 0.050

Motor[1].FatalFeLimit = 0.500
Motor[1].WarnFeLimit = 0

Motor[1].InPosBand = 0.000100
Motor[1].InPosTime = 10 / Sys.ServoPeriod

Motor[1].pPhaseEnc = Gate3[0].Chan[0].PhaseCapt.a
Motor[1].PhaseCtrl = 4
Motor[1].PhaseEncLeftShift = 0
Motor[1].PhaseEncRightShift = 0
Motor[1].PhasePosSf = 2048 * Mtr1Res / (Mtr1Ecl * 16384)

Motor[1].AmpFaultLevel = 1
Motor[1].AdcMask = $FFFF0000
Motor[1].PhaseOffset = 683
Motor[1].PwmSf = 0.95 * 16384

Motor[1].MaxDac = Mtr1MaxRmsCur * 32768 * COSD(30) * SQRT(2) / Ck3a1MaxAdc
Motor[1].I2TSet = Mtr1RatedRmsCur * 32768 * COSD(30) * SQRT(2) / Ck3a1MaxAdc
Motor[1].I2tTrip =(POW(Motor[1].MaxDac,2)-POW(Motor[1].I2TSet,2))*Mtr1TimeAtMaxCur

Motor[1].CurrentNullPeriod = 0
Motor[1].IaBias = 0
Motor[1].IbBias = 0

Motor[1].IiGain = 0
Motor[1].IpfGain = 0
Motor[1].IpbGain = 0

Motor[1].Servo.Kp = 0
Motor[1].Servo.Kvfb = 0
Motor[1].Servo.Kvifb = 0
Motor[1].Servo.Kvff = 0
Motor[1].Servo.Kviff = 0
Motor[1].Servo.Ki = 0
Motor[1].Servo.Kaff = 0
Motor[1].Servo.Kfff = 0
Motor[1].Servo.SwZvInt = 0

Motor[1].pAbsPhasePos = Gate3[0].Chan[0].Status.a
Motor[1].AbsPhasePosFormat = $400030C
Motor[1].AbsPhasePosSf = 2048 / 12
Motor[1].AbsPhasePosOffset = 0
Motor[1].PowerOnMode = 2
```

CK3A の最大 ADC
分解能、エンコーダピッチ［mm］
MM あたりのカウント数
電気サイクル長さ［mm］
最大 RMS 電流（MTR<CK3A）
定格 RMS 電流（MTR<CK3A）
最大電流での許容可能時間

正弦波の典型例
正弦波の典型例
エンコーダカウント方向

チャネルをアクティブ
IDE ウィンドウ単位［mm］
スケール位置：mm
スケール速度：mm
リミット有効、=0 の場合、無効

減速度の中止、100MSEC
S 字カーブ中断、なし

ジョグ加速度、100MSEC
ジョグ S 字カーブ、なし
ジョグ速度、50mm/秒

フォローエラーリミット：500μm
警告フォローエラーリミット、なし

位置帯：100nm
位置時間：10MSEC

正弦波の典型例
ブラシレスモータの典型例
正弦波、なし
正弦波、なし
正弦波の典型例

CK3A の場合は、=1
CK3A の場合は、=$FFFF0000
ブラシレスモータの典型例
MTR 電圧>入力の場合の典型例

ブラシレスの典型的な I2T
…
…

CK3A の場合は、=0
正常動作中、=0
正常動作中、=0

電流ループゲイン
（調整が必要）
…

位置サーボループゲイン
（調整が必要）
…
…
…
…
…
…
…

ホールの場合の絶対位相

ホールごとの手順を±設定する必要がある
ホールごとの手順を見つける必要がある
電源投入時の ABS ホール位相

A-3-5 BiSS エンコーダ付きのリニアサーボモータ（ACC-84E）

ハードウェア構成

項目	内容
コントローラ	このサンプルは、ACC-84E（BiSS）ありの UMAC ACC-24E3 に適用
アンプ	形 CK3A-G310L（三相、AC208V、主電源電圧）
モータ	リニアサーボモータ、AC200V、2.6/8.3A、60.96mm 電気サイクル長さ
エンコーダ	BiSS、32 ビット、1nm 分解能

システムゲート

```
Sys.WpKey = $AAAAAAAA
Sys.MaxCoords = 2
Sys.MaxMotors = 5

Gate3[0].PhaseFreq = 10000
Gate3[0].ServoClockDiv = 1
Sys.RtIntPeriod = 1
Sys.ServoPeriod = 1000 * (Gate3[0].ServoClockDiv + 1) / Gate3[0].PhaseFreq
Sys.PhaseOverServoPeriod = 1 / (Gate3[0].ServoClockDiv + 1)

Gate3[0].Chan[0].PwmFreqMult = 1
Gate3[0].Chan[1].PwmFreqMult = 1
Gate3[0].Chan[2].PwmFreqMult = 1
Gate3[0].Chan[3].PwmFreqMult = 1

Gate3[0].Chan[0].PackInData = 0
Gate3[0].Chan[1].PackInData = 0
Gate3[0].Chan[2].PackInData = 0
Gate3[0].Chan[3].PackInData = 0

Gate3[0].Chan[0].PackOutData = 0
Gate3[0].Chan[1].PackOutData = 0
Gate3[0].Chan[2].PackOutData = 0
Gate3[0].Chan[3].PackOutData = 0

Gate3[0].AdcAmpStrobe = $901001
Gate3[0].AdcAmpClockDiv = 5
Gate3[0].AdcAmpHeaderBits = 4
Gate3[0].Chan[0].PwmDeadTime = 3 / 0.0533

ACC84E[0].SerialEncCtrl = $31000B
ACC84E[0].Chan[0].SerialEncCmd = $2114A0
```

書き込み保護の無効
最大座標系+1（効率性）
最大モータ数+1（効率性）

10kHz のフェーズ周波数
5kHz のサーボ周波数
2.5kHz の RTI 周波数
典型的な計算
典型的な計算

CH1～4 10kHz の PWM 周波数

CK3A の場合、CH1～4 入力データが 0 である

CK3A の場合、CH1～4 出力データが 0 である

CH1～4 ストローブワード（CK3A の場合）
CH1～4 3.125MHz ADC クロック
CH1～4 は、4 である（CK3A の場合）
G310 の場合、CH1 が最小 3μSEC である

シリアルエンコーダ制御（BISS）
シリアルエンコーダコマンド（32 ビット）

エンコーダ変換テーブル（ECT）

```
EncTable[1].Type = 1
EncTable[1].pEnc = ACC84E[0].Chan[0].SerialEncDataA.a
EncTable[1].pEnc1 = Sys.Pushm
EncTable[1].index1 = 8
EncTable[1].index2 = 8
EncTable[1].index3 = 0
EncTable[1].index4 = 0
EncTable[1].index5 = 0
EncTable[1].index6 = 0
EncTable[1].ScaleFactor = EXP2(EncTable[1].index1)
EncTable[1].MaxDelta = 0

EncTable[2].Type = 0
```

典型的なシリアル ACC-84 エントリ

ECT の終了（効率性のため）

モータの設定と試運転

```
GLOBAL Ck3a1MaxAdc = 31.470
GLOBAL Mtr1Res = 0.000001
GLOBAL Mtr1CtsPerMm = 1 / Mtr1Res
GLOBAL Mtr1Ecl = 60.96
GLOBAL Mtr1MaxRmsCur = 8.3
GLOBAL Mtr1RatedRmsCur = 2.6
GLOBAL Mtr1TimeAtMaxCur = 2

Motor[1].ServoCtrl = 1
Motor[1].PosUnit = 3
Motor[1].PosSf = 1 / Mtr1CtsPerMm
Motor[1].Pos2Sf = 1 / Mtr1CtsPerMm
Motor[1].pLimits = Gate3[0].Chan[0].Status.a

Motor[1].AbortTa = 100
Motor[1].AbortTs = 0

Motor[1].JogTa = 100
Motor[1].JogTs = 0
Motor[1].JogSpeed = 0.050

Motor[1].FatalFeLimit = 0.500
Motor[1].WarnFeLimit = 0

Motor[1].InPosBand = 0.000100
Motor[1].InPosTime = 10 / Sys.ServoPeriod

Motor[1].pPhaseEnc = ACC84E[0].Chan[0].SerialEncDataA.a
Motor[1].PhaseCtrl = 4
Motor[1].PhaseEncLeftShift = 8
Motor[1].PhaseEncRightShift = 8
Motor[1].PhasePosSf = 2048 * Mtr1Res / (EXP2(Motor[1].PhaseEncLeftShift) * Mtr1Ecl)

Motor[1].AmpFaultLevel = 1
Motor[1].AdcMask = $FFFF0000
Motor[1].PhaseOffset = 683
Motor[1].PwmSf = 0.95 * 16384

Motor[1].MaxDac = Mtr1MaxRmsCur * 32768 * COSD(30) * SQRT(2) / Ck3a1MaxAdc
Motor[1].I2TSet = Mtr1RatedRmsCur * 32768 * COSD(30) * SQRT(2) / Ck3a1MaxAdc
Motor[1].I2tTrip =(POW(Motor[1].MaxDac,2)-POW(Motor[1].I2TSet,2))*Mtr1TimeAtMaxCur

Motor[1].CurrentNullPeriod = 0
Motor[1].IaBias = 0
Motor[1].IbBias = 0

Motor[1].IiGain = 0
Motor[1].IpfGain = 0
Motor[1].IpbGain = 0

Motor[1].Servo.Kp = 0
Motor[1].Servo.Kvfb = 0
Motor[1].Servo.Kvifb = 0
Motor[1].Servo.Kvff = 0
Motor[1].Servo.Kviff = 0
Motor[1].Servo.Ki = 0
Motor[1].Servo.Kaff = 0
Motor[1].Servo.Kfff = 0
Motor[1].Servo.SwZvInt = 0

Motor[1].pAbsPos = ACC84E[0].Chan[0].SerialEncDataA.a
Motor[1].AbsPosFormat = $00082008
Motor[1].AbsPosSF = Motor[1].PosSf
Motor[1].HomeOffset = 0

Motor[1].pAbsPhasePos = ACC84E[0].Chan[0].SerialEncDataA.a
Motor[1].AbsPhasePosFormat = $00082008
Motor[1].AbsPhasePosSf = 2048 * Mtr1Res / Mtr1Ecl
//Motor[1].AbsPhasePosForce =
//ACC84E[0].Chan[0].SerialEncDataA+(ACC84E[0].Chan[0].SerialEncDataB&$FF)*16777216
Motor[1].AbsPhasePosForce = 253109861
Motor[1].AbsPhasePosOffset =-Motor[1].AbsPhasePosForce * Motor[1].AbsPhasePosSf
Motor[1].PowerOnMode = $6
```

CK3A の最大 ADC
1nm 分解能［mm］
MM あたりのカウント数
電気サイクル長さ［mm］
最大 RMS 電流（MTR<CK3A）
定格 RMS 電流（MTR<CK3A）
最大電流での許容可能時間

チャネルをアウクティブ
IDE ウィンド単位［mm］
スケール位置：mm
スケール速度：mm
リミット有効、=0 の場合、無効

減速度の中止、100MSEC
S 字カーブ中断、なし

ジョグ加速度、100MSEC
ジョグ S 字カーブ、なし
ジョグ速度、50mm/秒

フォローエラーリミット：500μm
警告フォローエラーリミット、なし

位置帯：100nm
位置時間：10MSEC

シリアルエンコーダの典型例
ブラシレスモータの典型例
ACC-84
ACC-84
シリアルエンコーダの典型例

CK3A の場合は、=1
CK3A の場合は、=$FFFF0000
ブラシレスモータの典型例
MTR 電圧>入力の場合の典型例

ブラシレスの典型的な I2T
…
…

CK3A の場合は、=0
正常動作中、=0
正常動作中、=0

電流ループゲイン
（調整が必要）
…

位置サーボループゲイン
（調整が必要）
…
…
…
…
…
…
…

絶対シリアルエンコーダの典型例
符号なし 32 ビット
位置スケールと同じ
ユーザ設定可能な位置オフセット

絶対シリアルエンコーダの典型例
絶対位相には 32 ビットを使用
絶対シリアルエンコーダの典型例
32 ビットを読み出し、値をテスト

テスト値を保存
典型的な設定
電源投入時の絶対位相と位置

A-4　Gate1「スクリプト」モータ設定サンプル

ここでは、参照用として、Gate1（UMAC ACC-24E2）を用いたブラシレスモータの設定サンプルを提供します。

安全上の要点

ここで提供するサンプルは参照用であり、モータを直ちに動作させない可能性があります。以下の項目に注意する必要があります。

- エンコーダとモータの仕様パラメータ
- 電流ループチューニングを実行する必要があります
- 初期位相を実行する必要があります
- 開ループテストを実行する必要があります
- 位置ループチューニングを実行する必要があります
- 絶対位置の設定（該当する場合）を確認する必要があります
- 絶対位相基準の設定（該当する場合）を確認する必要があります
- 一般的なモーションパラメータ（ジョグ速度など）を確認する必要があります

参考

簡単にするために、ここのサンプルは、Power PMAC について Motor#1、Gate1、index4、channel0 を参照し、Turbo PMAC について$78200 ベースアドレスを参照しています。

A-4-1 Gate1(ACC-24E2)ありの Power PMAC

ハードウェア構成

項目	内容
コントローラ	このサンプルは、ACC-24E2 ありの Power PMAC UMAC に適用
アンプ	形 CK3A-G305L（単相 AC110V 主電源電圧）
モータ	ロータリサーボモータ、DC48V、3/9A、2 極ペア
エンコーダ	パルス、500 ライン（x4 デコードで 1 回転あたり 2,000 カウント）

システムゲート

```
Sys.WpKey = $AAAAAAAA
Sys.MaxCoords = 2
Sys.MaxMotors = 5

Gate1[4].PwmPeriod = 2948
Gate1[4].PhaseClockDiv = 1
Gate1[4].ServoClockDiv = 1
Sys.RtIntPeriod = 1
Sys.ServoPeriod = 0.2
Sys.PhaseOverServoPeriod = 0.5

Gate1[4].AdcStrobe = $901001
Gate1[4].HardwareClockCtrl = 2258
Gate1[4].Chan[0].PwmDeadTime = 2 / 0.135
```

書き込み保護の無効
最大座標系+1（効率性）
最大モータ数+1 （効率性）

20kHz の最大フェーズ周波数
10kHz の位相および PWM 周波数
5kHz のサーボ周波数
2.5kHz の RTI 周波数
サーボ期間［MSEC］
位相オーバーサーボ比率

CH1～4 ストローブワード（CK3A の場合）
CH1～4 2.4576MHz ADC クロック（初期値）
G310 の場合、CH1～4 2μSEC

エンコーダ変換テーブル（ECT）

```
EncTable[1].Type = 3
EncTable[1].pEnc = Gate1[4].Chan[0].ServoCapt.a
EncTable[1].pEnc1 = Gate1 [4].Chan[0].TimeBetweenCts.a
EncTable[1].index1 = 0
EncTable[1].index2 = 0
EncTable[1].index3 = 0
EncTable[1].index4 = 0
EncTable[1].index5 = 0
EncTable[1].index6 = 0
EncTable[1].ScaleFactor = 1 / 512
EncTable[1].MaxDelta = 0

EncTable[2].Type = 0
```

典型的なパルスエンコーダエントリ

ECT の終了（効率性のため）

モータの設定と試運転

```
GLOBAL Ck3a1MaxAdc = 15.735
GLOBAL Mtr1CtsPerRev = 2000
GLOBAL Mtr1PolePairs = 4
GLOBAL Mtr1MaxRmsCur = 9
GLOBAL Mtr1RatedRmsCur = 3
GLOBAL Mtr1TimeAtMaxCur = 2

Gate1[4].Chan[0].EncCtrl = 7
Motor[1].ServoCtrl = 1
Motor[1].PosUnit = 11
Motor[1].PosSf = 360 / Mtr1CtsPerRev
Motor[1].Pos2Sf = 360 / Mtr1CtsPerRev
Motor[1].pLimits = Gate1[4].Chan[0].Status.a

Motor[1].AbortTa = 150
Motor[1].AbortTs = 0

Motor[1].JogTa = 300
Motor[1].JogTs = 0
Motor[1].JogSpeed = 3

Motor[1].FatalFeLimit = 5
Motor[1].WarnFeLimit = 0

Motor[1].InPosBand = 0.01
Motor[1].InPosTime = 20 / Sys.ServoPeriod

Motor[1].pPhaseEnc = Gate1[4].Chan[0].PhaseCapt.a
Motor[1].PhaseCtrl = 4
Motor[1].PhaseEncLeftShift = 0
Motor[1].PhaseEncRightShift = 0
Motor[1].PhasePosSf = 2048 * Mtr1PolePairs / (256 * Mtr1CtsPerRev)

Motor[1].AmpFaultLevel = 1
Motor[1].AdcMask = $FFFF0000
Motor[1].PhaseOffset =-683
Motor[1].PwmSf =-Gate1[4].PwmPeriod * 48 / 155

Motor[1].MaxDac = Mtr1MaxRmsCur * 32768 * COSD(30) * SQRT(2) / Ck3a1MaxAdc
Motor[1].I2TSet = Mtr1RatedRmsCur * 32768 * COSD(30) * SQRT(2) / Ck3a1MaxAdc
Motor[1].I2tTrip =(POW(Motor[1].MaxDac,2)-POW(Motor[1].I2TSet,2))*Mtr1TimeAtMaxCur

Motor[1].CurrentNullPeriod = 0
Motor[1].IaBias = 0
Motor[1].IbBias = 0

Motor[1].IiGain = 0
Motor[1].IpfGain = 0
Motor[1].IpbGain = 0

Motor[1].Servo.Kp = 0
Motor[1].Servo.Kvfb = 0
Motor[1].Servo.Kvifb = 0
Motor[1].Servo.Kvff = 0
Motor[1].Servo.Kviff = 0
Motor[1].Servo.Ki = 0
Motor[1].Servo.Kaff = 0
Motor[1].Servo.Kfff = 0
Motor[1].Servo.SwZvInt = 0

Motor[1].PhaseFindingTime = 4000 / (2 * Sys.ServoPeriod * (Sys.RtIntPeriod + 1))
Motor[1].PhaseFindingDac = 0.5 * Motor[1].I2tSet
Motor[1].AbsPhasePosOffset = 2048 / 5
Motor[1].PowerOnMode = 0
```

CK3A の最大 ADC
1 回転あたりのカウント数
極ペア数
最大 RMS 電流（MTR>CK3A）
定格 RMS 電流（MTR>CK3A）
最大電流での許容可能時間

エンコーダカウント方向
チャネルをアクティブ
IDE ウィンドウ単位：度(deg)
スケール位置：度(deg)
スケール速度：度(deg)
リミット有効、=0 の場合、無効

減速度中断、150MSEC
S 字カーブ中断、なし

ジョグ加速度、300MSEC
ジョグ S 字カーブ、なし
ジョグ速度、4.5 度/MSEC

フォローエラーリミット：5 度
警告フォローエラーリミット、なし

位置帯：0.05 度
位置時間：20MSEC

直交の典型例
ブラシレスモータの典型例
直交、なし
直交、なし
直交の典型例

CK3A の場合は、=1
CK3A の場合は、=$FFFF0000
ブラシレスモータの典型例
MTR 電圧>入力の場合の典型例

ブラシレスの典型的な I2T
…
…

CK3A の場合は、=0
正常動作中、=0
正常動作中、=0

電流ループゲイン
（調整が必要）
…

位置サーボループゲイン
（調整が必要）
…
…
…
…
…
…
…

ステッパー位相（4 秒）
I2TSET の 50%
典型的な設定
電源投入時の絶対位相がなし

A-4-2　Gate1（ACC-24E2）ありの Turbo PMAC

ハードウェア構成

項目	内容
コントローラ	このサンプルは、ACC-24E2 ありの Turbo PMAC UMAC に適用
アンプ	形 CK3A-G305L（単相 AC110V 主電源電圧）
モータ	ロータリサーボモータ、DC48V、3/9A、2 極ペア
エンコーダ	パルス、500 ライン（x4 デコードで 1 回転あたり 2,000 カウント）

システムゲート

```
END GAT
DEL GAT
CLOSE

I7200 = 2947
I7201 = 1
I7202 = 1
I10 = 1677368

I7206 = $901001
I7203 = 2258
I7204 = 2 / 0.135

#define MaxPhaseFreq   P8000 ; 最大フェーズクロック[kHz]
#define PWMClk         P8001 ; PWM クロック[kHz]
#define PhaseClk       P8002 ; フェーズクロック[kHz]
#define ServoClk       P8003 ; サーボクロック[kHz]
MaxPhaseFreq = 117964.8 / (2 * I7200 + 3)
PWMClk = 117964.8 / (4 * I7200 + 6)
PhaseClk = MaxPhaseFreq / (I7201 + 1)
ServoClk = PhaseClk / (I7202 + 1)
```

ギャザーバッファの終了
ギャザーバッファの削除
開いているバッファをすべてクローズします

20kHz の最大フェーズ周波数
10kHz の位相および PWM 周波数
5kHz のサーボ周波数
サーボ割込み時間

CH1～4 ストローブワード（CK3A の場合）
CH1～4 2.4576MHz ADC クロック（初期値）
G310 の場合、CH1～4 2μSEC

エンコーダ変換テーブル（ECT）

```
I8000 = $78200
I8001 = 0
```

典型的なパルスエンコーダエントリ
ECT の終了（効率性のため）

モータの設定と試運転

```
I7210 = 3                                                    エンコーダカウント方向
I100 = 1                                                     サーボ制御有効
I101 = 1                                                     位相制御有効
I124 = $800001                                               フラグ制御

I184 = $FFFF00                                               ADC マスク（CK3A の場合）
I172 = 683                                                   位相オフセット、ブラシレスの典型例
I166 = I7200 * 48 / 155                                      PWM スケール値、モータ<供給電圧

I170 = 2                                                     極ペア数
I171 = 2000                                                  1回転あたりのカウント数

I119 = 0.5                                                   基本モーション設定
I120 = 100                                                   …
I121 = 0
I122 = 32
I123 = 16
I115 = 0.5
I111 = 55 * 16
I112 = 0
I128 = 1 * 16

#define ContCurrent    3                                     I2T 計算
#define PeakCurrent    9
#define MaxADC         15.735
#define I2TOnTime      2
I157 = INT(32767 * (ContCurrent * 1.414 / MaxADC) * COS(30))
I169 = INT (32767 * (PeakCurrent * 1.414 / MaxADC) * COS (30) )
I158 = INT((I169*I169-I157*I157)*ServoClk*1000*I2TOnTime/(32767*32767))

I182 = $ 78216                                               電流ループアドレス
I129 = 0
I179 = 0                                                     電流ループの調整
I161 = 0
I162 = 0
I176 = 0

I130 = 0                                                     位置ループの調整
I131 = 0
I132 = 0
I133 = 0
I134 = 0
I135 = 0
I136 = 0
I137 = 0
I138 = 0
I139 = 0                                                     4 つの推測位相方法
I168 = 0

I180 = 6
I173 = 1000
I174 = 60
```

A-5　Gate1 用 CK3A サンプル PLC

A-5-1　Gate1 ありの Power PMAC

CK3A 定義 4 モータ

```
// GATE3[0]、CH1～4 アンプ機能 ====================== //
GLOBAL Ck3a0Clrf, Ck3a0FanCtrl, Ck3a0BusDisCtrl, Ck3a0DbCtrl
PTR Gate30DataCtrl      ->U.IO:$6000B0.16.4
PTR Ck3a0BusDisState    ->U.IO:$6000B0.13.1
PTR Ck3a0ClrfBit        ->U.IO:$6000B0.24.1
PTR Ck3a0FanState       ->U.IO:$6000B0.25.1
PTR Ck3a0DbState        ->U.IO:$6000B0.28.1

// CH1～4 ADC B レジスタ [15:08] ========================== //
PTR Ck3a1AdcBData ->U.IO:$600018.8.8 // Gate3[0].Chan[0].AdcAmp[1]
PTR Ck3a2AdcBData ->U.IO:$600058.8.8 // Gate3[0].Chan[1].AdcAmp[1]
PTR Ck3a3AdcBData ->U.IO:$600098.8.8 // Gate3[0].Chan[2].AdcAmp[1]
PTR Ck3a4AdcBData ->U.IO:$6000D8.8.8 // Gate3[0].Chan[3].AdcAmp[1]

// CH1～4 アンプデータ ============================== //
GLOBAL Ck3a1DcBus, Ck3a1Temp, Ck3a1FwVer, Ck3a1CurRating
GLOBAL Ck3a2DcBus, Ck3a2Temp, Ck3a2FwVer, Ck3a2CurRating
GLOBAL Ck3a3DcBus, Ck3a3Temp, Ck3a3FwVer, Ck3a3CurRating
GLOBAL Ck3a4DcBus, Ck3a4Temp, Ck3a4FwVer, Ck3a4CurRating

// CH1、Gate3[0].Chan[0].AdcAmp[0]、ステータスセット 1 ============== //
PTR Ck3a1PwrFlt         ->U.IO:$600014.8.1
PTR Ck3a1SoftStartFlt   ->U.IO:$600014.9.1
PTR Ck3a1StoStatus      ->U.IO:$600014.10.1
PTR Ck3a1ShuntShortFlt  ->U.IO:$600014.11.1
PTR Ck3a1OverVoltFlt    ->U.IO:$600014.12.1
PTR Ck3a1I2tFlt         ->U.IO:$600014.13.1
PTR Ck3a1ShortFlt       ->U.IO:$600014.14.1
PTR Ck3a1OverLoadFlt    ->U.IO:$600014.15.1

// CH2、Gate3[0].Chan[1].AdcAmp[0]、ステータスセット 1 ============== //
PTR Ck3a2PwrFlt         ->U.IO:$600054.8.1
PTR Ck3a2SoftStartFlt   ->U.IO:$600054.9.1
PTR Ck3a2StoStatus      ->U.IO:$600054.10.1
PTR Ck3a2ShuntShortFlt  ->U.IO:$600054.11.1
PTR Ck3a2OverVoltFlt    ->U.IO:$600054.12.1
PTR Ck3a2I2tFlt         ->U.IO:$600054.13.1
PTR Ck3a2ShortFlt       ->U.IO:$600054.14.1
PTR Ck3a2OverLoadFlt    ->U.IO:$600054.15.1

// CH3、Gate3[0].Chan[2].AdcAmp[0]、ステータスセット 1 ============== //
PTR Ck3a3PwrFlt         ->U.IO:$600094.8.1
PTR Ck3a3SoftStartFlt   ->U.IO:$600094.9.1
PTR Ck3a3StoStatus      ->U.IO:$600094.10.1
PTR Ck3a3ShuntShortFlt  ->U.IO:$600094.11.1
PTR Ck3a3OverVoltFlt    ->U.IO:$600094.12.1
PTR Ck3a3I2tFlt         ->U.IO:$600094.13.1
PTR Ck3a3ShortFlt       ->U.IO:$600094.14.1
PTR Ck3a3OverLoadFlt    ->U.IO:$600094.15.1

// CH4、Gate3[0].Chan[3].AdcAmp[0]、ステータスセット 1 ============== //
PTR Ck3a4PwrFlt         ->U.IO:$6000D4.8.1
PTR Ck3a4SoftStartFlt   ->U.IO:$6000D4.9.1
PTR Ck3a4StoStatus      ->U.IO:$6000D4.10.1
PTR Ck3a4ShuntShortFlt  ->U.IO:$6000D4.11.1
PTR Ck3a4OverVoltFlt    ->U.IO:$6000D4.12.1
PTR Ck3a4I2tFlt         ->U.IO:$6000D4.13.1
PTR Ck3a4ShortFlt       ->U.IO:$6000D4.14.1
PTR Ck3a4OverLoadFlt    ->U.IO:$6000D4.15.1

// CH1、SYS.UDATA[256000] ステータスセット 2 ======================== //
PTR Ck3a1PwrStatus      ->U.USER:$FA000.0.1
PTR Ck3a1OverTempFlt    ->U.USER:$FA000.2.1
PTR Ck3a1ShuntOverLFlt  ->U.USER:$FA000.3.1
PTR Ck3a1PwmFreqFlt     ->U.USER:$FA000.4.1
PTR Ck3a1PwmCmdFlt      ->U.USER:$FA000.5.1
PTR Ck3a1AmpEna         ->U.USER:$FA000.6.1
```

$6000B00 は Gate1[i].AdcStrobe、
アドレスは以下のように：
Gate1[i].AdcStrobe.a - Sys.piom

$600xx8 は ADC B
Gate1[i].Chan[j].Adc[1]、
アドレスは以下のように：
Gate1[i].Chan[j].Adc[1].a - Sys.piom

$600014 は ADC A
Gate1[i].Chan[j].Adc[0]、
アドレスは以下のように：
Gate1[i].Chan[j].Adc[0].a - Sys.piom

$600054 は ADC A
Gate1[i].Chan[j].Adc[0]、
アドレスは以下のように：
Gate1[i].Chan[j].Adc[0].a - Sys.piom

$600094 は ADC A
Gate1[i].Chan[j].Adc[0]、
アドレスは以下のように：
Gate1[i].Chan[j].Adc[0].a - Sys.piom

$6000D4 は ADC A
Gate1[i].Chan[j].Adc[0]、
アドレスは以下のように：
Gate1[i].Chan[j].Adc[0].a - Sys.piom

$FA000 は Sys.Udata[256000]、
アドレスは以下のように：
Sys.Udata[i].a – Sys.pushm

```
// CH2、SYS.UDATA[256001] ステータスセット 2 ======================== //
PTR Ck3a2PwrStatus      ->U.USER:$FA004.0.1
PTR Ck3a2OverTempFlt    ->U.USER:$FA004.2.1
PTR Ck3a2ShuntOverLFlt ->U.USER:$FA004.3.1
PTR Ck3a2PwmFreqFlt     ->U.USER:$FA004.4.1
PTR Ck3a2PwmCmdFlt      ->U.USER:$FA004.5.1
PTR Ck3a2AmpEna         ->U.USER:$FA004.6.1

// CH3、SYS.UDATA[256002] ステータスセット 2 ======================== //
PTR Ck3a3PwrStatus      ->U.USER:$FA008.0.1
PTR Ck3a3OverTempFlt    ->U.USER:$FA008.2.1
PTR Ck3a3ShuntOverLFlt ->U.USER:$FA008.3.1
PTR Ck3a3PwmFreqFlt     ->U.USER:$FA008.4.1
PTR Ck3a3PwmCmdFlt      ->U.USER:$FA008.5.1
PTR Ck3a3AmpEna         ->U.USER:$FA008.6.1

// CH4、SYS.UDATA[256003] ステータスセット 2 ======================== //
PTR Ck3a4PwrStatus      ->U.USER:$FA00C.0.1
PTR Ck3a4OverTempFlt    ->U.USER:$FA00C.2.1
PTR Ck3a4ShuntOverLFlt ->U.USER:$FA00C.3.1
PTR Ck3a4PwmFreqFlt     ->U.USER:$FA00C.4.1
PTR Ck3a4PwmCmdFlt      ->U.USER:$FA00C.5.1
PTR Ck3a4AmpEna         ->U.USER:$FA00C.6.1

// CH1、SYS.UDATA[256040] ステータスセット 3 ======================== //
PTR Ck3a1I2tint         ->U.USER:$FA0A0.2.1
PTR Ck3a1AdcOffsetFlt   ->U.USER:$FA0A0.4.1
PTR Ck3a1Ready          ->U.USER:$FA0A0.6.1

// CH2、SYS.UDATA[256041] ステータスセット 3 ======================== //
PTR Ck3a2I2tint         ->U.USER:$FA0A4.2.1
PTR Ck3a2AdcOffsetFlt   ->U.USER:$FA0A4.4.1
PTR Ck3a2Ready          ->U.USER:$FA0A4.6.1

// CH3、SYS.UDATA[256042] ステータスセット 3 ======================== //
PTR Ck3a3I2tint         ->U.USER:$FA0A8.2.1
PTR Ck3a3AdcOffsetFlt   ->U.USER:$FA0A8.4.1
PTR Ck3a3Ready          ->U.USER:$FA0A8.6.1

// CH4、SYS.UDATA[256043] ステータスセット 3 ======================== //
PTR Ck3a4I2tint         ->U.USER:$FA0AC.2.1
PTR Ck3a4AdcOffsetFlt   ->U.USER:$FA0AC.4.1
PTR Ck3a4Ready          ->U.USER:$FA0AC.6.1
```

$FA004 は Sys.Udata[256001]、アドレスは以下のように：Sys.Udata[i].a – Sys.pushm

$FA008 は Sys.Udata[256002]、アドレスは以下のように：Sys.Udata[i].a – Sys.pushm

$FA00C は Sys.Udata[256003]、アドレスは以下のように：Sys.Udata[i].a – Sys.pushm

$ FA0A0 は Sys.Udata[256040]、アドレスは以下のように：Sys.Udata[i].a – Sys.pushm

$FA0A4 は Sys.Udata[256041]、アドレスは以下のように：Sys.Udata[i].a – Sys.pushm

$FA0A8 は Sys.Udata[256042]、アドレスは以下のように：Sys.Udata[i].a – Sys.pushm

$FA0AC は Sys.Udata[256043]、アドレスは以下のように：Sys.Udata[i].a – Sys.pushm

CK3A サンプル PLC 4 モータ

```
OPEN PLC Ck3aPLC
// CH1～4 ステータスビットセット 2
Gate30DataCtrl = 0
CALL DelayTimer.msec(1)
Sys.Udata[256000] = Ck3a1AdcBData
Sys.Udata[256001] = Ck3a2AdcBData
Sys.Udata[256002] = Ck3a3AdcBData
Sys.Udata[256003] = Ck3a4AdcBData
CALL DelayTimer.msec(1)
// CH1～4 ステータスビットセット 3
Gate30DataCtrl = 1
CALL DelayTimer.msec(1)
Sys.Udata[256040] = Ck3a1AdcBData
Sys.Udata[256041] = Ck3a2AdcBData
Sys.Udata[256042] = Ck3a3AdcBData
Sys.Udata[256043] = Ck3a4AdcBData
CALL DelayTimer.msec(1)
// CH1～4 DC バス電圧
Gate30DataCtrl = 2
CALL DelayTimer.msec(1)
IF(Ck3a1Ready == 0) Ck3a1DcBus = Ck3a1AdcBData * 4
IF(Ck3a2Ready == 0) Ck3a2DcBus = Ck3a2AdcBData * 4
IF(Ck3a3Ready == 0) Ck3a3DcBus = Ck3a3AdcBData * 4
IF(Ck3a4Ready == 0) Ck3a4DcBus = Ck3a4AdcBData * 4
CALL DelayTimer.msec(1)
// CH1～4 温度
Gate30DataCtrl = 3
CALL DelayTimer.msec(1)
IF(Ck3a1Ready == 0) Ck3a1Temp = Ck3a1AdcBData
IF(Ck3a2Ready == 0) Ck3a2Temp = Ck3a2AdcBData
IF(Ck3a3Ready == 0) Ck3a3Temp = Ck3a3AdcBData
IF(Ck3a4Ready == 0) Ck3a4Temp = Ck3a4AdcBData
CALL DelayTimer.msec(1)
// CH1～4 ファームウェアバージョン
Gate30DataCtrl = 6
CALL DelayTimer.msec(1)
IF(Ck3a1Ready == 0) Ck3a1FwVer = Ck3a1AdcBData
IF(Ck3a2Ready == 0) Ck3a2FwVer = Ck3a2AdcBData
IF(Ck3a3Ready == 0) Ck3a3FwVer = Ck3a3AdcBData
IF(Ck3a4Ready == 0) Ck3a4FwVer = Ck3a4AdcBData
CALL DelayTimer.msec(1)
// CH1～4 電流定格
Gate30DataCtrl = 7
CALL DelayTimer.msec(1)
IF(Ck3a1Ready == 0) Ck3a1CurRating = Ck3a1AdcBData
IF(Ck3a2Ready == 0) Ck3a2CurRating = Ck3a2AdcBData
IF(Ck3a3Ready == 0) Ck3a3CurRating = Ck3a3AdcBData
IF(Ck3a4Ready == 0) Ck3a4CurRating = Ck3a4AdcBData
CALL DelayTimer.msec(1)
// GATE3[0]、CH1～4 バス放電制御
IF(Ck3a0BusDisCtrl == 1 && Ck3a0BusDisState == 0) Ck3a0BusDisState = 1
IF(Ck3a0BusDisCtrl == 0 && Ck3a0BusDisState == 1) Ck3a0BusDisState = 0
// GATE3[0]、CH1～4 ファン制御
IF(Ck3a0FanCtrl == 1 && Ck3a0FanState == 0) Ck3a0FanState = 1
IF(Ck3a0FanCtrl == 0 && Ck3a0FanState == 1) Ck3a0FanState = 0
// GATE3[0]、CH1～4 ダイナミックブレーキ制御
IF(Ck3a0DbCtrl == 1 && Ck3a0DbState == 1) Ck3a0DbState = 0
IF(Ck3a0DbCtrl == 0 && Ck3a0DbState == 0) Ck3a0DbState = 1
// GATE3[0]、CH1～4 異常解除
IF(Ck3a0Clrf == 1)
{
    Ck3a0ClrfBit = 1
    CALL DelayTimer.msec(10)
    Ck3a0ClrfBit = 0
    Ck3a0Clrf = 0
}
// CH1～4 ロジック電源 OFF?
IF (Ck3a1Ready == 1) Ck3a1DcBus、4 = 0
IF(Ck3a2Ready == 1) Ck3a2DcBus,4 = 0
IF (Ck3a3Ready == 1) Ck3a3DcBus、4 = 0
IF (Ck3a4Ready == 1) Ck3a4DcBus、4 = 0
CLOSE
```

A-5-2 Gate1 ありの Turbo PMAC

CK3A 定義 4 モータ

```
// CH1～4 アンプ機能
#define Ck3a0Clrf                P1000
#define Ck3a0BusDisCtrl          P1001
#define Ck3a0FanCtrl             P1002
#define Ck3a0DbCtrl              P1003

#define Ck3a0BusDisState         P1004
#define Ck3a0FanState            P1005
#define Ck3a0DbState             P1006

// CH1～4 ADC B データレジスタ
#define Ck3a1AdcBData            M1000
#define Ck3a2AdcBData            M1001
#define Ck3a3AdcBData            M1002
#define Ck3a4AdcBData            M1003

Ck3a1AdcBData->Y:$078206,0,8
Ck3a2AdcBData->Y:$07820E,0,8
Ck3a3AdcBData->Y:$078216,0,8
Ck3a4AdcBData->Y:$07821E,0,8

// CH1～4 データ
#define Ck3a1DcBus               P1010
#define Ck3a1Temp                P1011
#define Ck3a1FwVer               P1012
#define Ck3a1CurRating           P1013

#define Ck3a2DcBus               P1014
#define Ck3a2Temp                P1015
#define Ck3a2FwVer               P1016
#define Ck3a2CurRating           P1017

#define Ck3a3DcBus               P1018
#define Ck3a3Temp                P1019
#define Ck3a3FwVer               P1020
#define Ck3a3CurRating           P1021

#define Ck3a4DcBus               P1022
#define Ck3a4Temp                P1023
#define Ck3a4FwVer               P1024
#define Ck3a4CurRating           P1025

// CH1～4 ステータスビットセット 1、ADC A（任意のストローブ）
#define Ck3a1PwrFlt              M1004
#define Ck3a1SoftStartFlt        M1005
#define Ck3a1StoStatus           M1006
#define Ck3a1ShuntShortFlt       M1007
#define Ck3a1OverVoltFlt         M1008
#define Ck3a1I2tFlt              M1009
#define Ck3a1ShortFlt            M1010
#define Ck3a1OverLoadFlt         M1011
Ck3a1PwrFlt->Y:$78205,0,1
Ck3a1SoftStartFlt->Y:$78205,1,1
Ck3a1StoStatus->Y:$78205,2,1
Ck3a1ShuntShortFlt->Y:$78205,3,1
Ck3a1OverVoltFlt->Y:$78205,4,1
Ck3a1I2tFlt->Y:$78205,5,1
Ck3a1ShortFlt->Y:$78205,6,1
Ck3a1OverLoadFlt->Y:$78205,7,1

#define Ck3a2PwrFlt              M1012
#define Ck3a2SoftStartFlt        M1013
#define Ck3a2StoStatus           M1014
#define Ck3a2ShuntShortFlt       M1015
#define Ck3a2OverVoltFlt         M1016
#define Ck3a2I2tFlt              M1017
#define Ck3a2ShortFlt            M1018
#define Ck3a2OverLoadFlt         M1019
Ck3a2PwrFlt->Y:$7820D,0,1
Ck3a2SoftStartFlt->Y:$7820D,1,1
Ck3a2StoStatus->Y:$7820D,2,1
Ck3a2ShuntShortFlt->Y:$7820D,3,1
Ck3a2OverVoltFlt->Y:$7820D,4,1
Ck3a2I2tFlt->Y:$7820D,5,1
Ck3a2ShortFlt->Y:$7820D,6,1
Ck3a2OverLoadFlt->Y:$7820D,7,1
```

```
#define Ck3a3PwrFlt              M1020
#define Ck3a3SoftStartFlt        M1021
#define Ck3a3StoStatus           M1022
#define Ck3a3ShuntShortFlt       M1023
#define Ck3a3OverVoltFlt         M1024
#define Ck3a3I2tFlt              M1025
#define Ck3a3ShortFlt            M1026
#define Ck3a3OverLoadFlt         M1027
Ck3a3PwrFlt->Y:$78215,0,1
Ck3a3SoftStartFlt-> Y : $ 78215,1,1
Ck3a3StoStatus->Y:$78215,2,1
Ck3a3ShuntShortFlt->Y:$78215,3,1
Ck3a3OverVoltFlt->Y:$78215,4,1
Ck3a3I2tFlt->Y:$78215,5,1
Ck3a3ShortFlt->Y:$78215,6,1
Ck3a3OverLoadFlt->Y:$78215,7,1

#define Ck3a4PwrFlt              M1028
#define Ck3a4SoftStartFlt        M1029
#define Ck3a4StoStatus           M1030
#define Ck3a4ShuntShortFlt       M1031
#define Ck3a4OverVoltFlt         M1032
#define Ck3a4I2tFlt              M1033
#define Ck3a4ShortFlt            M1034
#define Ck3a4OverLoadFlt         M1035
Ck3a4PwrFlt->Y:$7821D,0,1
Ck3a4SoftStartFlt->Y:$7821D,1,1
Ck3a4StoStatus-> Y : $ 7821D、2,1
Ck3a4ShuntShortFlt->Y:$7821D,3,1
Ck3a4OverVoltFlt->Y:$7821D,4,1
Ck3a4I2tFlt-> Y : $ 7821D、5,1
Ck3a4ShortFlt-> Y : $ 7821D、6,1
Ck3a4OverLoadFlt->Y:$7821D,7,1

// CH1～4 ステータスビットセット 2、ADC B（ストローブ［11 : 8］ = 0）
#define Ck3a1PwrStatus           M1036
#define Ck3a1OverTempFlt         M1037
#define Ck3a1ShuntOverLFlt       M1038
#define Ck3a1PwmFreqFlt          M1039
#define Ck3a1PwmCmdFlt           M1040
#define Ck3a1AmpEna              M1041
Ck3a1PwrStatus->X:$10F0,0,1
Ck3a1OverTempFlt->X:$10F0,2,1
Ck3a1ShuntOverLFlt->X:$10F0,3,1
Ck3a1PwmFreqFlt->X:$10F0,4,1
Ck3a1PwmCmdFlt->X:$10F0,5,1
Ck3a1AmpEna-> X : $ 10F0,6,1

#define Ck3a2PwrStatus           M1042
#define Ck3a2OverTempFlt         M1043
#define Ck3a2ShuntOverLFlt       M1044
#define Ck3a2PwmFreqFlt          M1045
#define Ck3a2PwmCmdFlt           M1046
#define Ck3a2AmpEna              M1047
Ck3a2PwrStatus->X:$10F1,0,1
Ck3a2OverTempFlt->X:$10F1,2,1
Ck3a2ShuntOverLFlt->X:$10F1,3,1
Ck3a2PwmFreqFlt->X:$10F1,4,1
Ck3a2PwmCmdFlt->X:$10F1,5,1
Ck3a2AmpEna-> X : $ 10F1,6,1

#define Ck3a3PwrStatus           M1048
#define Ck3a3OverTempFlt         M1049
#define Ck3a3ShuntOverLFlt       M1050
#define Ck3a3PwmFreqFlt          M1051
#define Ck3a3PwmCmdFlt           M1052
#define Ck3a3AmpEna              M1053
Ck3a3PwrStatus->X:$10F2,0,1
Ck3a3OverTempFlt->X:$10F2,2,1
Ck3a3ShuntOverLFlt->X:$10F2,3,1
Ck3a3PwmFreqFlt->X:$10F2,4,1
Ck3a3PwmCmdFlt->X:$10F2,5,1
Ck3a3AmpEna-> X : $ 10F2,6,1
```

```
#define Ck3a4PwrStatus          M1054
#define Ck3a4OverTempFlt        M1055
#define Ck3a4ShuntOverLFlt      M1056
#define Ck3a4PwmFreqFlt         M1057
#define Ck3a4PwmCmdFlt          M1058
#define Ck3a4AmpEna             M1059
Ck3a4PwrStatus->X:$10F3,0,1
Ck3a4OverTempFlt->X:$10F3,2,1
Ck3a4ShuntOverLFlt->X:$10F3,3,1
Ck3a4PwmFreqFlt->X:$10F3,4,1
Ck3a4PwmCmdFlt->X:$10F3,5,1
Ck3a4AmpEna-> X : $ 10F3,6,1

// CH1～4 ステータスビットセット 3、ADC B（ストローブ［11 : 8］ = 1）
#define Ck3a1I2tint             M1060
#define Ck3a1AdcOffsetFlt       M1061
#define Ck3a1Ready              M1062
Ck3a1I2tint-> Y : $ 10F0,2,1
Ck3a1AdcOffsetFlt->Y:$10F0,4,1
Ck3a1Ready-> Y : $ 10F0,6,1

#define Ck3a2I2tint             M1063
#define Ck3a2AdcOffsetFlt       M1064
#define Ck3a2Ready              M1065
Ck3a2I2tint-> Y : $ 10F1,2,1
Ck3a2AdcOffsetFlt->Y:$10F1,4,1
Ck3a2Ready-> Y : $ 10F1,6,1

#define Ck3a3I2tint             M1066
#define Ck3a3AdcOffsetFlt       M1067
#define Ck3a3Ready              M1068
Ck3a3I2tint->Y:$10F2,2,1
Ck3a3AdcOffsetFlt->Y:$10F2,4,1
Ck3a3Ready->Y:$10F2,6,1

#define Ck3a4I2tint             M1069
#define Ck3a4AdcOffsetFlt       M1070
#define Ck3a4Ready              M1071
Ck3a4I2tint->Y:$10F3,2,1
Ck3a4AdcOffsetFlt->Y:$10F3,4,1
Ck3a4Ready->Y:$10F3,6,1

// CH1～4 ADC B ミラーワード
#define Ch1Mirror2              M1072
#define Ch1Mirror3              M1073
Ch1Mirror2->X:$10F0,0,8
Ch1Mirror3->Y:$10F0,0,8

#define Ch2Mirror2              M1074
#define Ch2Mirror3              M1075
Ch2Mirror2->X:$10F1,0,8
Ch2Mirror3->Y:$10F1,0,8

#define Ch3Mirror2              M1076
#define Ch3Mirror3              M1077
Ch3Mirror2->X:$10F2,0,8
Ch3Mirror3->Y:$10F2,0,8

#define Ch4Mirror2              M1078
#define Ch4Mirror3              M1079
Ch4Mirror2->X:$10F3,0,8
Ch4Mirror3->Y:$10F3,0,8
```

CK3A サンプル PLC 4 モータ

```
// モータ 1～4 CK3A PLC
OPEN PLC 1 CLEAR
I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW
// ステータスビットセット 2
I7206 = (I7206 & $FFF0FF) | $0
I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW
Ch1Mirror2 = Ck3a1AdcBData
Ch2Mirror2 = Ck3a2AdcBData
Ch3Mirror2 = Ck3a3AdcBData
Ch4Mirror2 = Ck3a4AdcBData
I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW

// ステータスビットセット 3
I7206 = (I7206 & $FFF0FF) | $100
I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW
Ch1Mirror3 = Ck3a1AdcBData
Ch2Mirror3 = Ck3a2AdcBData
Ch3Mirror3 = Ck3a3AdcBData
Ch4Mirror3 = Ck3a4AdcBData
I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW

// DC バス電圧
I7206 = (I7206 & $FFF0FF) | $200
I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW
IF(Ck3a1Ready = 0) Ck3a1DcBus = Ck3a1AdcBData * 4 ENDIF
IF(Ck3a2Ready = 0) Ck3a2DcBus = Ck3a2AdcBData * 4 ENDIF
IF(Ck3a3Ready = 0) Ck3a3DcBus = Ck3a3AdcBData * 4 ENDIF
IF(Ck3a4Ready = 0) Ck3a4DcBus = Ck3a4AdcBData * 4 ENDIF
I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW

// 温度
I7206 = (I7206 & $FFF0FF) | $300
I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW
IF(Ck3a1Ready = 0) Ck3a1Temp = Ck3a1AdcBData ENDIF
IF(Ck3a2Ready = 0) Ck3a2Temp = Ck3a2AdcBData ENDIF
IF(Ck3a3Ready = 0) Ck3a3Temp = Ck3a3AdcBData ENDIF
IF(Ck3a4Ready = 0) Ck3a4Temp = Ck3a4AdcBData ENDIF
I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW

// ファームウェアバージョン
I7206 = (I7206 & $FFF0FF) | $600
I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW
IF(Ck3a1Ready = 0) Ck3a1FwVer = Ck3a1AdcBData ENDIF
IF(Ck3a2Ready = 0) Ck3a2FwVer = Ck3a2AdcBData ENDIF
IF(Ck3a3Ready = 0) Ck3a3FwVer = Ck3a3AdcBData ENDIF
IF(Ck3a4Ready = 0) Ck3a4FwVer = Ck3a4AdcBData ENDIF
I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW

// 電流定格
I7206 = (I7206 & $FFF0FF) | $700
I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW
IF(Ck3a1Ready = 0) Ck3a1CurRating = Ck3a1AdcBData ENDIF
IF(Ck3a2Ready = 0) Ck3a2CurRating = Ck3a2AdcBData ENDIF
IF(Ck3a3Ready = 0) Ck3a3CurRating = Ck3a3AdcBData ENDIF
IF(Ck3a4Ready = 0) Ck3a4CurRating = Ck3a4AdcBData ENDIF
I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW

// バス放電制御
Ck3a0BusDisState = (I7206 & $20) / 32
I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW
IF(Ck3a0BusDisCtrl = 1 AND Ck3a0BusDisState = 0)
     I7206 = (I7206 & $FFFFDF) | $20
ENDIF
IF(Ck3a0BusDisCtrl = 0 AND Ck3a0BusDisState = 1)
     I7206 = (I7206 & $FFFFDF) | $0
ENDIF
I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW

// ファン制御
Ck3a0FanState = (I7206 & $20000) / 131072
I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW
IF(Ck3a0FanCtrl = 1 AND Ck3a0FanState = 0)
     I7206 = (I7206 & $FDFFFF) | $20000
ENDIF
IF(Ck3a0FanCtrl = 0 AND Ck3a0FanState = 1)
     I7206 = (I7206 & $FDFFFF) | $0
ENDIF
I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW
```

```
// ダイナミックブレーキ制御
Ck3a0DbState = (I7206 & $100000) / 1048576
I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW
IF(Ck3a0DbCtrl = 1 AND Ck3a0DbState = 1)
     I7206 = (I7206 & $EFFFFF) | $0
ENDIF
IF(Ck3a0DbCtrl = 0 AND Ck3a0DbState = 0)
     I7206 = (I7206 & $EFFFFF) | $100000
ENDIF
I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW

// 異常解除
IF(Ck3a0Clrf = 1)
     I7206 = (I7206 & $FEFFFF) | $10000
     I5111 = 10 * 8388608 / I10 WHILE (I5111 > 0) ENDW
     I7206 = (I7206 & $FEFFFF) | $0
     I5111 = 1 * 8388608 / I10 WHILE (I5111 > 0) ENDW
     Ck3a0Clrf = 0
ENDIF

// 準備ができていません（電源 OFF）？
IF(Ck3a1Ready = 1)
     Ck3a1DcBus = 0
     Ck3a1Temp = 0
     Ck3a1FwVer = 0
     Ck3a1CurRating = 0
ENDIF

IF(Ck3a2Ready = 1)
     Ck3a2DcBus = 0
     Ck3a2Temp = 0
     Ck3a2FwVer = 0
     Ck3a2CurRating = 0
ENDIF

IF(Ck3a3Ready = 1)
     Ck3a3DcBus = 0
     Ck3a3Temp = 0
     Ck3a3FwVer = 0
     Ck3a3CurRating = 0
ENDIF

IF(Ck3a4Ready = 1)
     Ck3a4DcBus = 0
     Ck3a4Temp = 0
     Ck3a4FwVer = 0
     Ck3a4CurRating = 0
ENDIF
CLOSE
```

A-6　STO 安全上の注意

A-6-1　危険分析

はじめに

通常動作中、アンプはコントローラに接続され、コントローラから指示されない限り、出力を生成しません。したがって、モータを停止する必要がある場合、アンプはモータへの電流出力を停止する必要があります。さらに、アンプを(電流を出力しない)この無効モードのままにしておくことが望ましい場合があります。この結果は、次の 2 つの方法で実現できます。

1. コントローラがアンプにモータへの電流出力を停止するように指示する
2. 安全装置がアンプに直接指示して、出力電流をゼロにする

この 2 番目の方法は、安全トルク OFF (STO) 機能を利用して実現されます。

リスクアセスメント

CK3A DirectPWM アンプには、UL SIL 3 要件を満たし、このドキュメントで説明されているように安全機能として機能することができる 安全トルク OFF (STO) 機能が搭載されています。STO は、損傷、怪我、または死亡を防ぐためにさまざまなメカニズムと方法が使用される安全システムの一部にすぎないことを覚えておくことが重要です。STO 自体は、システムの唯一の安全機能として使用しないでください。アンプが使用される特定のシステムまたはマシンを設計するエンドユーザに、包括的な安全システムの設計と実装が求められます。アンプが使用されるシステムを設計するときは、すべての潜在的な危険、各危険の深刻度、および各危険の発生確率を特定するために、徹底的なリスクアセスメントを実行する必要があります。リスクアセスメントでは、人が各危険にさらされる頻度と、個人がそれらを効果的に回避する方法についても検討する必要があります。アンプを使用したシステムを実装する前に、資格のある個人またはチームがリスクアセスメントを実行する必要があります。

CK3A の潜在的なリスク

● 全般

このユーザマニュアルに記載されている方法から逸脱したり、ここに記載されている警告を無視したりすると、損傷、重傷、または死亡につながる可能性があります。

● 感電

アンプのメイン入力は、1 相または 3 相で最大 AC 240V です。この電圧レベルは危険であり、重傷または死亡につながる可能性があります。アンプの出力も AC 電圧であり、触れると危険です。アンプを取り扱い、使用する際には、このマニュアルのすべての警告を考慮する必要があります。アンプのカバーは絶対に取り外さないでください。取り外すと、感電の重大な可能性が生じます。

● 操作またはメンテナンス中のリスク

主な安全上の問題は、アンプが予期しないまたは望ましくない状況で出力電流を生成する可能性があることです。アンプが使用されるアプリケーションによっては、この現象により物的損害や傷害または死亡が発生する可能性があります。メンテナンス中は、主電源とロジック電源の両方をオフにして、それをオフのまま保持する必要があります。

この付録の残りの部分では、STO 機能の使用例について詳しく説明します。

● STO の使用によって軽減される可能性のある特定のリスク

緊急事態が発生した場合:

- コントローラが損傷や怪我を防ぐために時間内に停止コマンドを送信できない
- コントローラがアンプに停止コマンドをまったく送信できない

アンプの出力をシャットダウンしたままにする必要がある場合 (システムの取り扱いやメンテナンス中など):

コントローラがアンプを有効にして電流を出力するために誤ったコマンドを生成する

安全上のご注意

システムのメンテナンスを実行するときは、ユーザはアンプへのメイン AC 入力の電源を切るか、取り外す必要があります。

意図したアンプ出力	コントローラコマンド	STO コマンド	結果
OFF	OFF	X	安全
	X	OFF	安全
	ON	ON	危険

OFF: 出力電流を阻止する状態の信号
ON: 出力電流を許可する状態の信号
X: ON または OFF のいずれか

危険と軽減

STO 機能は、アンプの出力がオフになり、オフのままであることを確実にするためにシステム設計者が利用できるツールの 1 つです。この機能は、アンプ内部の IGBT のゲート電圧を除去することでこれを実現し、出力が生成されないようにします。

STO が期待どおりに動作しない可能性があるケースには、次のものがあります。

- 誤配線
- 外部安全装置の故障
- STO 回路内のコンポーネントの故障

誤配線

人為的ミスの可能性を減らすために、次のものを提供します。

- このユーザマニュアル上での詳細な配線図
- 安全コネクタに適切なラベルを貼付
- このマニュアル全体にわたる、推奨配線図に従わないと危険な結果につながる可能性があることを示す警告

さらに、設置とシステム設計は、どちらも適切な資格を持つ個人が行う必要があります。

外部安全装置の故障

外部安全装置は STO を制御する役割を担っています。この安全装置に問題が発生すると、STO の機能と信頼性が損なわれる可能性があります。このリスクを最小限に抑えるには、アンプの適切な安全機能を保証するために、最低でも SIL 3 の定格を持つ安全装置が必要です。さらに、ユーザには、安全装置が適切に機能することを確認するために、適切な頻度と方法で安全装置をテストすることをお勧めします。

STO 回路内の部品故障

他の電子機器と同様に、STO 機能は部品故障による異常の影響を受けます。この主題は、SIL 3 要件への準拠の一環として調査、分析、および文書化されています。

A-6-2 STO 仕様

このセクションでは、STO 機能の一般的な仕様について説明します。

全般

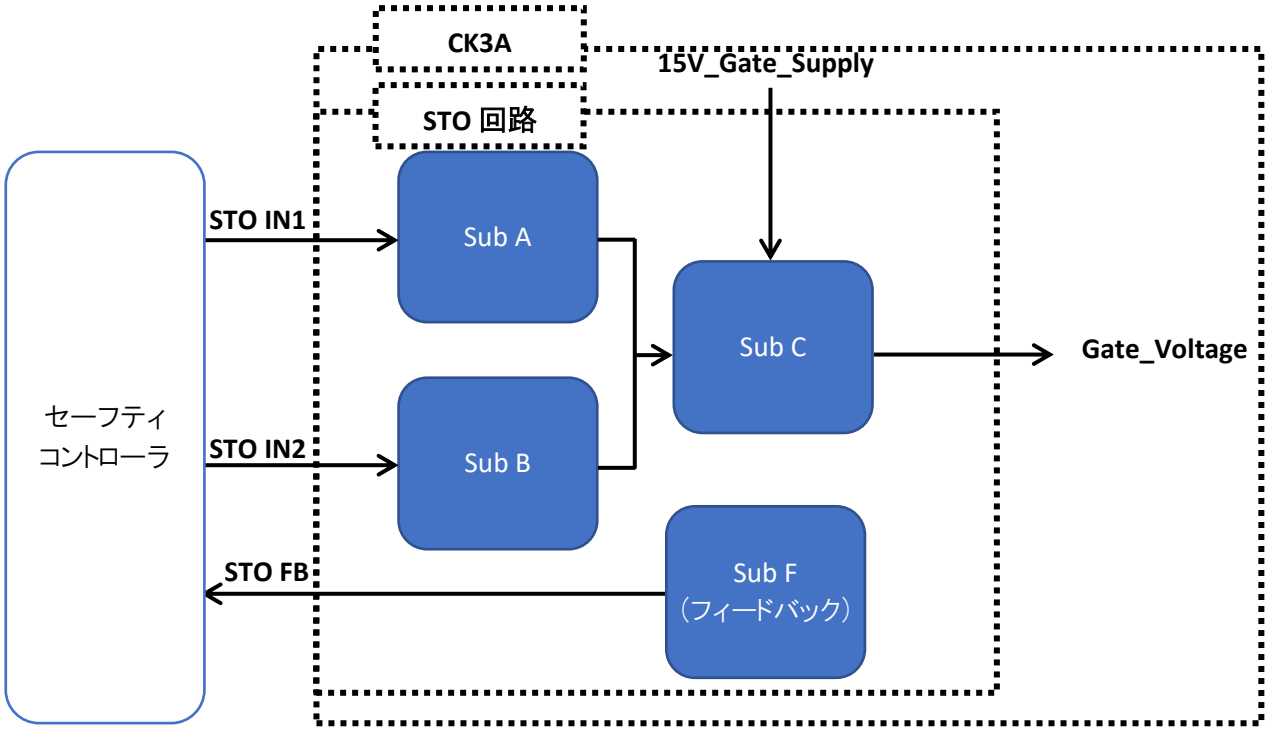
STO 反応時間:「2-1-4 性能」を参照ください。
STO 反応時間: STO をアクティブ化してからアンプの出力がオフになるまでの時間。

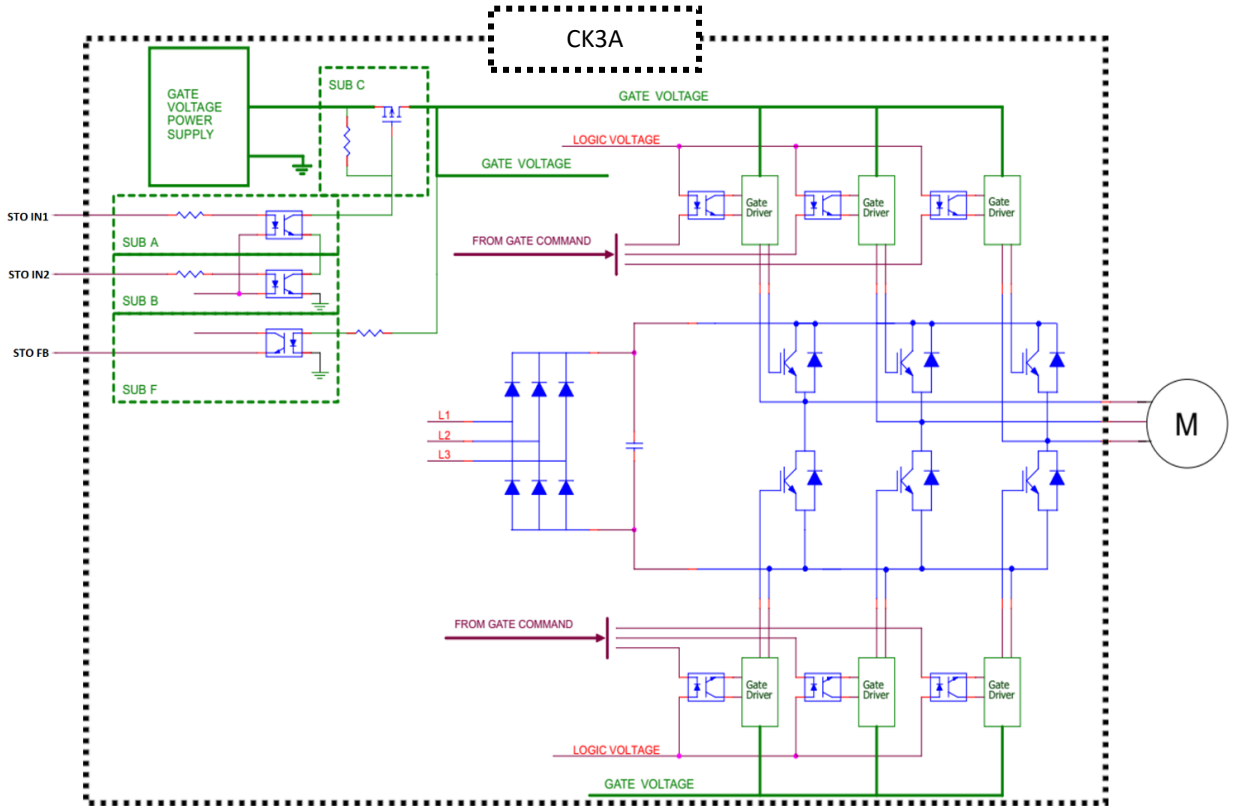
STO 機能	PFH(1/h)
安全入力信号による STO 機能	1.08×10^{-8}

A-6-3　STO の動作原理

IEC 60204-1. 安全トルク OFF (STO) 機能は、モータが力やトルクを発生しないようにします。言い換えると、モータが停止している場合、この安全機能によりモータは動きません。また、STO 機能が有効になっているときにモータが動いている場合は、IEC 60204-1 の停止カテゴリに従って制御不能な停止状態が続きます。

STO 回路は、次のブロック図に示すようにアンプとともに使用されます。モータが電力を生成しないようにするために、STO 回路はアンプの IGBT 回路への電源 (最初のブロック図では Gate_Voltage、2 番目のブロック図では GATE VOLTAGE) を除去します。この電源がない場合、IGBT は開いたスイッチとして機能し、コマンドされた電流がモータに流れ込むことはありません。




CK3A
GATE VOLTAGE POWER SUPPLY
SUB C
GATE VOLTAGE
LOGIC VOLTAGE
GATE VOLTAGE
STO IN1
SUB A
STO IN2
SUB B
STO FB
SUB F
FROM GATE COMMAND
Gate Driver
L1
L2
L3
M
FROM GATE COMMAND
LOGIC VOLTAGE
GATE VOLTAGE

オムロン株式会社 インダストリアルオートメーションビジネスカンパニー

製品に関するお問い合わせ先

お客様相談室

フリー通話 **0120-919-066**（クイック オムロン）

携帯電話の場合、

055-982-5015（有料）をご利用ください。

受付時間：9:00～17:00（土・日・12/31～1/3を除く）

オムロンFAクイックチャット

www.fa.omron.co.jp/contact/tech/chat/

技術相談員にチャットでお問い合わせいただけます。（I-Webメンバーズ限定）

受付時間：平日9:00～12:00 / 13:00～17:00（土日祝日・年末年始・当社休業日を除く）

※受付時間、営業日は変更の可能性がございます。最新情報はリンク先をご確認ください。

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

オムロン制御機器の最新情報をご覧いただけます。緊急時のご購入にもご利用ください。 **www.fa.omron.co.jp**

●本誌に記載のない条件や環境での使用、および原子力制御・鉄道・航空・車両・燃焼装置・医療機器・娯楽機械・安全機器、その他人命や財産に大きな影響が予測されるなど、特に安全性が要求される用途に使用される際には、当社の意図した特別な商品用途の場合や特別の合意がある場合を除き、当社は当社商品に対して一切保証をいたしません。

●本製品の内、外国為替及び外国貿易法に定める輸出許可、承認対象貨物（又は技術）に該当するものを輸出（又は非居住者に提供）する場合は同法に基づく輸出許可、承認（又は役務取引許可）が必要です。

オムロン商品のご用命は

Man. No. SBCE-518B 2025年2月現在

©OMRON Corporation 2022-2025 All Rights Reserved.
お断りなく仕様などを変更することがありますのでご了承ください