

三菱電機 汎用 シーケンサ

MELSEC iQ-R
series

MELSEC iQ-R モーションコントローラ プログラミングマニュアル(Gコード制御編)



-R16MTCPU
-R32MTCPU
-R64MTCPU



安全上のご注意


(ご使用前に必ずお読みください)

本製品のご使用に際しては、本マニュアルおよび本マニュアルで紹介している関連マニュアルをよくお読みいただくと共に、安全に対して十分に注意を払って、正しい取扱いをしていただくようお願いいたします。

本マニュアルで示す注意事項は、本製品に関するもののみについて記載したものです。シーケンサシステムとしての安全上のご注意に関しては、MELSEC iQ-R ユニット構成マニュアルを参照してください。

この「安全上のご注意」では、安全注意事項のランクを「 警告」、「 注意」として区分してあります。

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

なお、 注意に記載した事項でも、状況によっては重大な結果に結びつく可能性があります。

いずれも重要な内容を記載していますので必ず守ってください。

本マニュアルは必要なときに読めるよう大切に保管すると共に、必ず最終ユーザまでお届けいただくようお願いいたします。

[設計上の注意事項]

警告

- 外部電源の異常やシーケンサ本体の故障時でも、システム全体が安全側に働くようにシーケンサの外部で安全回路を設けてください。誤出力または誤動作により、事故の恐れがあります。
 - (1) 非常停止回路、保護回路、正転/逆転などの相反する動作のインタロック回路、位置決めの上限/下限など機械の破損防止のインタロック回路は、シーケンサの外部で構成してください。
 - (2) シーケンサは次の異常状態を検出すると、演算を停止し、出力は下記の状態になります。
 - ・電源ユニットの過電流保護装置または過電圧保護装置が働いたときは全出力をOFFする。
 - ・CPUユニットでウォッチドッグタイマーエラーなどの自己診断機能で異常を検出したときは、パラメータ設定により、全出力を保持またはOFFする。
 - (3) CPUユニットで検出できない入出力制御部分などの異常時は、全出力がONすることがあります。このとき、機械の動作が安全側に働こう、シーケンサの外部でフェールセーフ回路を構成したり、安全機構を設けたりしてください。フェールセーフ回路例については、MELSEC iQ-R ユニット構成マニュアルの「フェールセーフ回路の考え方」を参照してください。
 - (4) 出力回路のリレーやトランジスタなどの故障によっては、出力がONの状態やOFFの状態を保持することがあります。重大な事故につながるような出力信号については、外部で監視する回路を設けてください。
- 出力回路において、定格を超える負荷電流または負荷短絡などによる過電流が長時間継続して流れた場合、発煙や発火の恐れがありますので、外部にヒューズなどの安全回路を設けてください。
- シーケンサ本体の電源立上げ後に、外部供給電源を投入するように回路を構成してください。外部供給電源を先に立ち上げると、誤出力または誤動作により、事故の恐れがあります。
- ネットワークが交信異常になったときの各局の動作状態については、各ネットワークのマニュアルを参照してください。誤出力または誤動作により、事故の恐れがあります。
- CPUユニットまたはインテリジェント機能ユニットに外部機器を接続して、運転中のシーケンサに対する制御(データ変更)を行うときは、常にシステム全体が安全側に働くように、プログラム上でインタロック回路を構成してください。また、運転中のシーケンサに対するその他の制御(プログラム変更、パラメータ変更、強制出力、運転状態変更(状態制御))を行うときは、マニュアルを熟読し、十分に安全を確認してから行ってください。確認を怠ると、操作ミスにより機械の破損や事故の原因になります。

[設計上の注意事項]

警告

- 外部機器から遠隔地のシーケンサに対する制御では、データ通信異常によりシーケンサ側のトラブルにすぐに対応できない場合があります。プログラム上でインタロック回路を構成すると共に、データ通信異常が発生したときのシステムとしての処置方法を外部機器とCPUユニット間で取り決めてください。
 - ユニットのバッファメモリの中で、システムエリアまたは書き込み不可のエリアにはデータを書き込まないでください。また、CPUユニットから各ユニットに対する出力信号の中で、使用禁止の信号を出力(ON)しないでください。システムエリアまたは書き込み不可のエリアに対するデータの書き込み、使用禁止の信号に対する出力を行うと、シーケンサシステムが誤動作する危険性があります。システムエリアまたは書き込み不可のエリア、使用禁止の信号については、各ユニットのユーザーズマニュアルを参照してください。
 - 通信ケーブルが断線した場合は、回線が不安定になり、複数の局でネットワークが通信異常になる場合があります。通信異常が発生しても、システムが安全側に働くようにプログラム上でインタロック回路を構成してください。誤出力または誤動作により、事故の恐れがあります。
 - ネットワーク経由の外部機器からの不正アクセスに対して、シーケンサシステムの安全を保つ必要があるときは、ユーザによる対策を盛り込んでください。また、インターネット経由の外部機器からの不正アクセスに対して、シーケンサシステムの安全を保つ必要があるときは、ファイアウォールなどの対策を盛り込んでください。
 - 外部電源の異常やシーケンサ本体の故障時でも、システム全体が安全側に働くようにシーケンサの外部で安全回路を設けてください。誤出力または誤動作により、事故の恐れがあります。
 - ユニット、サーボアンプ、サーボモータを使用したシステムとしての安全基準(たとえばロボットなどの安全通則など)のあるものは安全基準を満足させてください。
 - ユニット、サーボアンプの異常時動作とシステムとしての安全方向動作が異なる場合はユニット・サーボアンプの外部で対策回路を構成してください。
 - ユニットやサーボアンプの制御電源が投入されているときに、SSCNETⅢケーブルを取りはずさないでください。ユニットやサーボアンプのSSCNETⅢコネクタおよびSSCNETⅢケーブルの先端から発せられる光を直視しないでください。光が目に入ると、目に違和感を感じる恐れがあります。(SSCNETⅢの光源は、JISC6802, IEC60825-1に規定されているクラス1に適合します。)
-

[設計上の注意事項]

注意

- 制御線や通信ケーブルは、主回路や動力線と束線したり、近接させたりしないでください。100mm以上を目安として離してください。ノイズにより、誤動作の原因になります。
 - ランプ負荷、ヒータ、ソレノイドバルブなどの誘導性負荷を制御するときは、出力のOFF→ON時に大きな電流(通常の10倍程度)が流れる場合がありますので、定格電流に余裕のあるユニットをお使いください。
 - CPUユニットの電源OFF→ONまたはリセット時、CPUユニットがRUN状態になるまでの時間が、システム構成、パラメータ設定、プログラム容量などにより変動します。RUN状態になるまでの時間が変動しても、システム全体が安全側に働くように設計してください。
 - 各種設定を登録中に、ユニット装着局の電源OFFおよびCPUユニットのリセットを行わないでください。登録中にユニット装着局の電源OFFおよびCPUユニットのリセットを行うと、フラッシュROM内、SDメモ리카ードのデータ内容が不定となり、バッファメモリへの設定値の再設定、フラッシュROM、SDメモ리카ードへの再登録が必要です。また、ユニットの故障および誤動作の原因になります。
 - 外部機器からCPUユニットに対する運転状態変更(リモートRUN/STOPなど)を行うときは、“ユニットパラメータ”の“オープン方法の設定”を、“プログラムでOPENしない”に設定してください。“オープン方法の設定”が“プログラムでOPENする”に設定されている場合は、外部機器からリモートSTOPを実行すると通信回線がクローズされます。以後はCPUユニット側で再オープンができなくなり、外部機器からのリモートRUNも実行できなくなります。
-

[取付け上の注意事項]

警告

- ユニットの着脱は、必ずシステムで使用している外部供給電源を全相遮断してから行ってください。全相遮断しないと、感電、ユニットの故障や誤動作の原因になります。
-

[取付け上の注意事項]

注意

- シーケンサは、安全にお使いいただくために(ベースユニットに同梱のマニュアル)記載の一般仕様の環境で使用してください。一般仕様の範囲以外の環境で使用すると、感電、火災、誤動作、製品の損傷または劣化の原因になります。
 - ユニットを装着するときは、ユニット下部の凹部をベースユニットのガイドに挿入し、ガイドの先端を支点として、ユニット上部のフックが「カチッ」と音がするまで押してください。ユニットが正しく装着されていないと、誤動作、故障または落下の原因になります。
 - ユニット固定用フックのないユニットを装着するときは、ユニット下部の凹部をベースユニットのガイドに挿入し、ガイドの先端を支点として押し、必ずネジで締め付けてください。ユニットが正しく装着されていないと、誤動作、故障または落下の原因になります。
 - 振動の多い環境で使用する場合は、ユニットをネジで締め付けてください。
 - ネジの締め付けは、規定トルク範囲で行ってください。ネジの締め付けがゆるいと、落下、短絡または誤動作の原因になります。ネジを締め過ぎると、ネジやユニットの破損による落下、短絡または誤動作の原因になります。
 - 増設ケーブルは、ベースユニットの増設ケーブル用コネクタに確実に装着してください。装着後に、浮上りがないか確認してください。接触不良により、誤動作の原因になります。
 - SDメモリカードは、装着スロットに押し込んで確実に装着してください。装着後に、浮上りがないか確認してください。接触不良により、誤動作の原因になります。
 - 拡張SRAMカセットまたはバッテリーレスオプションカセットは、CPUユニットのカセット接続用コネクタに押し込んで確実に装着してください。装着後はカセットカバーを閉め、浮上りがないか確認してください。接触不良により、誤動作の原因になります。
 - ユニット、SDメモリカード、拡張SRAMカセット、バッテリーレスオプションカセットまたはコネクタの、導電部分や電子部品に直接触らないでください。ユニットの故障や誤動作の原因になります。
-

[配線上の注意事項]

警告

- 取付けまたは配線作業は、必ずシステムで使用している外部供給電源を全相遮断してから行ってください。全相遮断しないと、感電、ユニットの故障や誤動作の原因になります。
 - 取付けまたは配線作業後、通電または運転を行う場合は、空きスロットにブランクカバーユニット(RG60)を、増設ケーブル用コネクタに付属の増設コネクタ保護カバーを取り付けてください。ブランクカバーユニット(RG60)や増設コネクタ保護カバーを取り付けないと、感電の恐れがあります。
-

[配線上の注意事項]

注意

- FG端子およびLG端子は、シーケンサ専用のD種接地(第三種接地)以上で必ず接地してください。感電または誤動作の恐れがあります。
 - 圧着端子は適合圧着端子を使用し、規定のトルクで締め付けてください。先開形圧着端子を使用すると、端子ネジがゆるんだ場合に脱落し、故障の原因になります。
 - ユニットへの配線は、製品の定格電圧および信号配列を確認後、正しく行ってください。定格と異なった電源を接続したり、誤配線したりすると、火災または故障の原因になります。
 - 外部機器接続用コネクタは、メーカ指定の工具で圧着、圧接または正しくハンダ付けしてください。接続が不完全な場合、短絡、火災または誤動作の原因になります。
 - コネクタは、確実にユニットに取り付けてください。接触不良により、誤動作の原因になります。
 - 制御線や通信ケーブルは、主回路や動力線と束線したり、近接させたりしないでください。100mm以上を目安として離してください。ノイズにより、誤動作の原因になります。
 - ユニットに接続する電線やケーブルは、必ずダクトに納めるか、またはクランプによる固定処理を行ってください。ケーブルのふらつきや移動、不注意の引っ張りなどによるユニットやケーブルの破損、ケーブルの接続不良による誤動作の原因になります。増設ケーブルには、外皮を取り除いたクランプ処理を行わないでください。ケーブルの特性変化により、誤動作の原因になります。
 - ケーブル接続は、接続するインタフェースの種類を確認の上、正しく行ってください。異なったインタフェースに接続または誤配線すると、ユニットまたは外部機器の故障の原因になります。
 - 端子ネジやコネクタ取付けネジの締め付けは、規定トルク範囲で行ってください。ネジの締め付けがゆるいと、落下、短絡、火災または誤動作の原因になります。ネジを締め過ぎると、ネジやユニットの破損による落下、短絡、火災または誤動作の原因になります。
 - ユニットに接続されたケーブルを取りはずすときは、ケーブル部分を引っ張らないでください。コネクタ付きのケーブルは、ユニットの接続部分のコネクタを持って取りはずしてください。端子台接続のケーブルは、端子台端子ネジを緩めてから取りはずしてください。ユニットに接続された状態でケーブルを引っ張ると、誤動作またはユニットやケーブルの破損の原因になります。
 - ユニット内に、切粉や配線クズなどの異物が入らないように注意してください。火災、故障または誤動作の原因になります。
 - 配線時にユニット内へ配線クズなどの異物混入を防止するため、ユニット上部に混入防止ラベルを貼り付けています。配線作業中は、本ラベルをはがさないでください。システム運転時は、放熱のために本ラベルを必ずはがしてください。
 - シーケンサは、制御盤内に設置して使用してください。制御盤内に設置されたシーケンサ電源ユニットへの主電源配線に関しては、中継端子台を介して行ってください。また、電源ユニットの交換と配線作業は、感電保護に対して、十分に教育を受けたメンテナンス作業者が行ってください。配線方法は、MELSEC iQ-R ユニット構成マニュアルを参照してください。
 - システムで使用するEthernetケーブルは、各ユニットのユーザズマニュアル記載の仕様に従ってください。仕様外の配線では、正常なデータ伝送を保証できません。
-

[立上げ・保守時の注意事項]

警告

- 通電中，端子に触れないでください。感電または誤動作の原因になります。
 - バッテリコネクタは，正しく接続してください。バッテリーに充電，分解，加熱，火中投入，ショート，ハンダ付け，液体を付着させる，強い衝撃を与えることは絶対に行わないでください。バッテリーの取扱いを誤ると，発熱，破裂，発火，液漏れにより，ケガまたは火災の恐れがあります。
 - 端子ネジ，コネクタ取付けネジまたはユニット固定ネジの増し締めや，ユニットの清掃は，必ずシステムで使用している外部供給電源を全相遮断してから行ってください。全相遮断しないと，感電の恐れがあります。
-

[立上げ・保守時の注意事項]

注意

- CPUユニットまたはインテリジェント機能ユニットに外部機器を接続して，運転中のシーケンサに対する制御(データ変更)を行うときは，常にシステム全体が安全側に働くように，プログラム上でインタロック回路を構成してください。また，運転中のシーケンサに対するその他の制御(プログラム変更，パラメータ変更，強制出力，運転状態変更(状態制御))を行うときは，マニュアルを熟読し，十分に安全を確認してから行ってください。確認を怠ると，操作ミスにより機械の破損や事故の原因になります。
 - 外部機器から遠隔地のシーケンサに対する制御では，データ交信異常により，シーケンサ側のトラブルにすぐに対応できない場合があります。プログラム上でインタロック回路を構成すると共に，データ交信異常が発生したときのシステムとしての処置方法を外部機器とCPUユニット間で取り決めてください。
 - ユニットの分解または改造はしないでください。故障，誤動作，ケガまたは火災の原因になります。
 - 携帯電話やPHSなどの無線通信機器は，シーケンサ本体の全方向から25cm以上離して使用してください。誤動作の原因になります。
 - ユニットの着脱は，必ずシステムで使用している外部供給電源を全相遮断してから行ってください。全相遮断しないと，ユニットの故障や誤動作の原因になります。
 - ネジの締め付けは，規定トルク範囲で行ってください。ネジの締め付けがゆるいと，部品や配線の落下，短絡または誤動作の原因になります。ネジを締め過ぎると，ネジやユニットの破損による落下，短絡または誤動作の原因になります。
 - 下記の着脱は，製品ご使用後，50回以内(JIS B 3502，IEC 61131-2に準拠)としてください。
なお，50回を超えた場合は，誤動作の原因となる恐れがあります。
 - ・ ユニットとベースユニット
 - ・ CPUユニットと，拡張SRAMカセットまたはバッテリーレスオプションカセット
 - ・ ユニットと端子台
 - SDメモ리카ードの取付け・取りはずしは，製品使用後，500回以内としてください。500回を超えた場合は，誤動作の原因となる恐れがあります。
 - SDメモ리카ード取扱い時は，剥き出しになっているカード端子に触れないでください。故障や誤動作の原因になります。
 - 拡張SRAMカセットまたはバッテリーレスオプションカセット取扱い時は，基板上のICに触れないでください。故障や誤動作の原因になります。
 - ユニットに装着するバッテリーには，落下・衝撃を加えないでください。落下・衝撃により，バッテリーが破損し，バッテリー液の液漏れがバッテリー内部で発生している恐れがあります。落下・衝撃を加えたバッテリーは使用せずに廃棄してください。
-

[立上げ・保守時の注意事項]

注意

- 制御盤内での立上げ・保守作業は、感電保護に対して、十分に教育を受けたメンテナンス作業者が行ってください。また、メンテナンス作業員以外が制御盤を操作できないよう、制御盤に鍵をかけてください。
 - ユニットに触れる前には、必ず接地された金属などの導電物に触れて、人体などに帯電している静電気を放電させてください。静電気を放電させないと、ユニットの故障や誤動作の原因になります。
 - 試運転は、パラメータの速度制限値を遅い速度に設定し、危険な状態が発生したとき即座に停止できる準備をしてから動作確認を行ってください。
 - 運転前にプログラムおよび各パラメータの確認・調整を行ってください。機械によっては予期しない動きとなる場合があります。
 - 絶対位置システム機能を使用している場合、新規立上げしたとき、またはユニット、絶対位置対応モータ等を交換したときは必ず原点復帰を行ってください。
 - ブレーキ機能を確認してから運転を行ってください。
 - 点検時にメガテスト(絶縁抵抗測定)を行わないでください。
 - 保守・点検終了時、絶対位置検出機能の位置検出が正しいか確認してください。
 - 電気設備に関する教育を受け、十分な知識を有する人のみ制御盤を開けることができるよう、制御盤に鍵をかけてください。
-

[運転時の注意事項]

注意

- インテリジェント機能ユニットにパソコンなどの外部機器を接続して運転中のシーケンサに対する制御(特にデータ変更、プログラム変更、運転状態変更(状態制御))を行うときはユーザーズマニュアルを熟読し、十分に安全を確認してから行ってください。データ変更、プログラム変更、状態制御を誤ると、システムの誤動作、機械の破損や事故の原因になります。
 - ユニット内のフラッシュ ROMへバッファメモリの設定値を登録して使用する場合、登録中はユニット装着局の電源OFFおよびCPUユニットのリセットを行わないでください。登録中にユニット装着局の電源OFFおよびCPUユニットのリセットを行うと、フラッシュ ROM内、SDメモ리카ードのデータ内容が不定となり、バッファメモリへの設定値の再設定、フラッシュ ROM、SDメモ리카ードへの再登録が必要です。また、ユニットの故障や誤動作の原因になります。
 - 補間運転の基準軸速度指定のときは、相手軸(2軸目、3軸目、4軸目)の速度が設定速度より大きく(速度制限値以上)なる場合がありますのでご注意ください。
 - 試験運転やティーチングなどの運転中は機械に近寄らないでください。傷害の原因になります。
-

[廃棄時の注意事項]

注意

- 製品を廃棄するときは、産業廃棄物として扱ってください。
 - バッテリーを廃棄する際は、各地域にて定められている法令に従い分別を行ってください。EU加盟国内でのバッテリー規制の詳細については、MELSEC iQ-R ユニット構成マニュアルを参照してください。
-

[輸送時の注意事項]

注意

- リチウムを含有しているバッテリーの輸送時は、輸送規制に従った取扱いが必要です。規制対象機種の詳細については、MELSEC iQ-R ユニット構成マニュアルを参照してください。
 - 木製梱包材の消毒および除虫対策のくん蒸剤に含まれるハロゲン系物質(フッ素, 塩素, 臭素, ヨウ素など)が当社製品に侵入すると故障の原因になります。残留したくん蒸成分が当社製品に侵入しないようご注意ください。くん蒸以外の方法(熱処理など)で処理してください。なお、消毒および除虫対策は梱包前の木材の段階で実施してください。
-

製品の適用について

- (1) 当社シーケンサをご使用いただくにあたりましては、万一シーケンサに故障・不具合などが発生した場合でも重大な事故にいたらない用途であること、および故障・不具合発生時にはバックアップやフェールセーフ機能が機器外部でシステム的に実施されていることをご使用の条件とさせていただきます。
- (2) 当社シーケンサは、一般工業などへの用途を対象とした汎用品として設計・製作されています。
したがって、以下のような機器・システムなどの特殊用途へのご使用については、当社シーケンサの適用を除外させていただきます。万一使用された場合は当社として当社シーケンサの品質、性能、安全に関する一切の責任（債務不履行責任、瑕疵担保責任、品質保証責任、不法行為責任、製造物責任を含むがそれらに限定されない）を負わないものとさせていただきます。
 - ・ 各電力会社殿の原子力発電所およびその他発電所向けなどの公共への影響が大きい用途
 - ・ 鉄道各社殿および官公庁殿など、特別な品質保証体制の構築を当社にご要求になる用途
 - ・ 航空宇宙、医療、鉄道、燃焼・燃料装置、乗用移動体、有人搬送装置、娯楽機械、安全機械など
生命、身体、財産に大きな影響が予測される用途ただし、上記の用途であっても、具体的に用途を限定すること、特別な品質（一般仕様を超えた品質等）をご要求されないこと等を条件に、当社の判断にて当社シーケンサの適用可とする場合もございますので、詳細につきましては当社窓口へご相談ください。

はじめに

このたびは、三菱電機シーケンサMELSEC iQ-Rシリーズをお買い上げいただき、まことにありがとうございました。
本マニュアルは、モーションコントローラでGコード制御を行う場合に必要な専用信号、パラメータ、データ、機能についてご理解いただくためのマニュアルです。
ご使用前に本マニュアルや関連マニュアルをよくお読みいただき、MELSEC iQ-Rシリーズのシーケンサの機能・性能を十分ご理解の上、正しくご使用くださるようお願いいたします。
また、本マニュアルで紹介するプログラム例を実際のシステムへ流用される場合は、対象システムにおける制御に問題がないことを十分検証ください。
本マニュアルにつきましては最終ユーザまでお届けいただきますよう、宜しく願い申し上げます。

対象ユニット

R16MTCPU, R32MTCPU, R64MTCPU

目次

安全上のご注意	1
製品の適用について	9
はじめに	9
関連マニュアル	14
用語	15
マニュアルの読み方	16
第1章 概要	18
1.1 Gコード制御の概要	18
1.2 性能仕様	19
1.3 Gコード制御のアドオンライブラリ構成	20
Gコード制御のアドオンライブラリ	20
ファイルサイズ／使用メモリ量	20
1.4 バージョンによる機能の制約	20
第2章 システムの立上げ	21
2.1 Gコード制御システムの立上げ	21
2.2 Gコード制御用アドオンライブラリのライセンス認証	23
2.3 Gコード制御の制御周期	23
2.4 Gコード制御システムのデバイス配置方式	24
2.5 Gコード制御システムの始動／終了	24
2.6 Gコード制御システムの停止動作	25
即停止	26
減速停止	26
第3章 Gコード制御専用信号	27
3.1 Gコード制御共通指令信号	28
3.2 Gコード制御共通制御デバイス	29
3.3 Gコード制御共通ステータス	30
3.4 Gコード制御共通モニタデバイス	31
3.5 Gコード制御系統指令信号	33
3.6 Gコード制御系統制御デバイス	37
3.7 Gコード制御系統ステータス	38
3.8 Gコード制御系統モニタデバイス	45
3.9 Gコード制御系統モニタデバイス(拡張)	55
3.10 Gコード制御軸ステータス	61
3.11 Gコード制御軸モニタデバイス	63
3.12 内部リレー (M)／データレジスタ(D)の有効可否	66
内部リレー	66
データレジスタ	69
第4章 Gコード制御用パラメータ	70
4.1 Gコード制御系統パラメータ	71
系統基本設定	73
モータル初期設定	73
制御設定	74
オーバライド設定	75

平面構成	76
法線制御	76
補助機能	77
極座標補間	77
高精度制御	78
マクロ制御	79
4.2 Gコード制御軸パラメータ	80
系統軸情報	81
ストアードストロークリミット	82
速度・時定数	83
回転軸情報	84
タンデム機能	85
高精度制御	86
4.3 Gコード制御加工パラメータ	87
工具径補正	87
工具補正データ	88
ワーク座標オフセット	88
プログラム座標回転	88
 第5章 Gコード制御用プログラム	 89
5.1 Gコード制御プログラムの構成	89
Gコードプログラムのフォーマット	90
5.2 Gコードプログラムファイルの取込み	92
運転中のGコードプログラムファイル取込み	92
5.3 先読みバッファ	94
5.4 小数点入力	94
5.5 座標系について	96
基本機械座標系	96
ワーク座標系	96
ローカル座標系	96
自動座標系設定	97
回転軸用座標系	97
5.6 Gコード	98
Gコード一覧	98
モーダル／アンモーダル	99
Gコードの優先順位	99
G00: 位置決め(早送り)	102
G01: 直線補間	103
G02: 円弧補間CW(中心指定)	104
G03: 円弧補間CCW(中心指定)	106
G02: 円弧補間CW(R指定)	108
G03: 円弧補間CCW(R指定)	110
G04: ドウェル(時間指定)	112
G09: イグザクトストップチェック	113
G12.1: 極座標補間モード開始	114
G13.1: 極座標補間モードキャンセル	118
G17～G19: 平面選択	119
G38: 工具径補正ベクトル指定	120
G39: 工具径補正コーナ円弧	122
G40: 工具径補正キャンセル	123

	G41: 工具径補正 左	124
	G42: 工具径補正 右	125
	G40.1: 法線制御 キャンセル	126
	G41.1: 法線制御 左オン	127
	G42.1: 法線制御 右オン	128
	G43: 工具長補正(+)	129
	G44: 工具長補正(-)	131
	G49: 工具長補正キャンセル	133
	G52: ローカル座標系設定	134
	G53: 基本機械座標系選択	138
	G54～G59: ワーク座標系1選択～ワーク座標系6選択	139
	G61: イグザクトストップチェックモード	141
	G61.1: 高精度制御モード	142
	G62: 自動コーナオーバーライド	143
	G64: 切削モード	146
	G65: マクロ呼出し	147
	G68: プログラム座標回転モード開始	153
	G69: プログラム座標回転モードキャンセル	159
	G90: 絶対値指令	160
	G91: 増分値指令	162
	G94: 毎分送り(非同期送り)	164
5.7	Mコード	165
	M00: プログラムストップ	166
	M01: プログラムストップ	167
	M02: プログラムエンド	168
	M30: プログラムエンド	169
	M98: サブプログラム呼出し	170
	M99: サブ/マクロプログラム復帰	172
5.8	変数指令	173
	変数の記述	175
	変数の引用	176
	変数の応用	177
	変数(デバイス)の読み込み／書き込みタイミング	177
	コモン変数	181
	ローカル変数	182
5.9	演算指令	184
	演算の優先順位	185
	定義/置換(=)	186
	加法形演算(+, -)	187
	乗法形演算(*, /, MOD)	188
	論理演算(OR, XOR, AND)	189
	三角関数(SIN, COS, TAN, ASIN, ACOS, ATAN)	190
	関数(SQRT, ABS, BIN, BCD, ROUND, FIX, FUP, LN, EXP, POW)	191
5.10	制御指令	193
	分岐(IF, GOTO)	193
	分岐(IF, THEN, ELSE, ENDIF)	195
	繰り返し(WHILE, DO, END)	198

第6章 補助／応用機能 200

6.1	Gコード制御と各機能の関係	200
-----	---------------------	-----

6.2	補助機能(M機能)	202
	M単独出力	202
	補助機能完了	204
6.3	送り機能	206
	早送り速度	206
	切削送り速度	207
	送り速度の指定と各制御軸への効果	208
	減速チェック	211
6.4	工具補正機能	215
	工具補正	215
	工具長補正	216
	工具径補正	216
6.5	操作支援機能	238
	自動運転起動(サイクルスタート)	238
	自動運転休止(フィードホールド)	239
	リセット	240
	シングルブロック	241
6.6	法線制御機能	242
6.7	高精度制御	255
	補間前加減速	257
	最適速度制御	259
	ベクトル精補間	265
	円弧入口/出口速度制御	265
6.8	タンデム機能	266
6.9	GOTによるGコードプログラムの操作	267
	GOTのプログラム入出力機能	267
	GOTのプログラム編集機能	267
6.10	マクロに関する機能	268
	マクロ指令を使用する際の注意事項	268
	マクロ指令の使用例	269
付録		270
付1	Gコード制御のエラー詳細コード	270
	Gコード制御エラー詳細コード	270
	アドオンライセンスエラー詳細コード	275
付2	Gコード制御のイベント詳細コード	276
	Gコード制御イベント詳細コード	276
	改訂履歴	278
	保証について	279
	購入に関するお問い合わせ	280
	サービスのお問い合わせ	280
	商標	280

関連マニュアル

最新のe-ManualおよびマニュアルPDFは、三菱電機FAサイトからダウンロードできます。

www.MitsubishiElectric.co.jp/fa

マニュアル名称[マニュアル番号]	内容	提供形態	価格
MELSEC iQ-R モーションコントローラプログラミングマニュアル (Gコード制御編) [IB-0300370](本マニュアル)	Gコード制御を行うためのGコード制御パラメータ、およびGコードプログラムについて説明しています。	製本物	3,000円
		e-Manual PDF	—
MELSEC iQ-R モーションコントローラユーザーズマニュアル [IB-0300234]	モーションCPUユニット、SSCNETⅢケーブルおよびシリアルABS同期エンコーダケーブル、トラブルシューティングなどについて説明しています。	製本物	1,500円
		e-Manual PDF	—
MELSEC iQ-R モーションコントローラプログラミングマニュアル (共通編) [IB-0300236]	マルチCPUシステム構成、性能仕様、共通パラメータ、補助／応用機能およびエラーリストなどについて説明しています。	製本物	4,000円
		e-Manual PDF	—
MELSEC iQ-R モーションコントローラプログラミングマニュアル (プログラム設計編) [IB-0300238]	モーションSFCの機能、プログラミング、およびデバッグなどについて説明しています。	製本物	3,000円
		e-Manual PDF	—
MELSEC iQ-R モーションコントローラプログラミングマニュアル (位置決め制御編) [IB-0300240]	サーボパラメータ、位置決め命令、およびデバイス一覧などについて説明しています。	製本物	4,000円
		e-Manual PDF	—
MELSEC iQ-R モーションコントローラプログラミングマニュアル (アドバンス同期制御編) [IB-0300242]	同期制御を行うための同期制御パラメータ、およびデバイス一覧などについて説明しています。	製本物	3,000円
		e-Manual PDF	—
MELSEC iQ-R モーションコントローラプログラミングマニュアル (マシン制御編) [IB-0300308]	マシン制御を行うためのマシン制御パラメータ、マシン位置決めデータ、およびデバイス一覧などについて説明しています。	製本物	3,000円
		e-Manual PDF	—

Point

e-Manualとは、専用のツールを使用して閲覧できる三菱電機FA電子書籍マニュアルです。

e-Manualには下記のような特長があります。

- 探したい情報を複数のマニュアルから一度に検索可能(マニュアル横断検索)
- マニュアル内のリンクから他マニュアルを参照可能
- 製品のイラストの各パーツから知りたいハードウェア仕様を閲覧可能
- 頻繁に参照する情報をお気に入り登録可能

用語

本マニュアルでは、特に明記する場合を除き、下記の用語を使用して説明します。

用語	内容
R64MTCPU/R32MTCPU/R16MTCPU もしくはモーションCPU(ユニット)	MELSEC iQ-Rシリーズ モーションコントローラの略称
MR-J4(W)-□B	MR-J4-□B/MR-J4W-□B形サーボアンプ
MR-J3(W)-□B	MR-J3-□B/MR-J3W-□B形サーボアンプ
AMPもしくはサーボアンプ	MR-J4-□B/MR-J4W-□B/MR-J3-□B/MR-J3W-□B形サーボアンプシリーズの総称
RnCPUもしくはシーケンサCPU	MELSEC iQ-Rシリーズ CPUユニットの略称
マルチCPUシステム もしくはモーションシステム	RシリーズシーケンサマルチCPUシステムの略称
CPU _n	マルチCPUシステムにおけるn号機のCPUユニット(n=1～4)の略称
本体OSソフトウェア	SW10DNC-RMTFWの総称
エンジニアリングソフトウェアパッ ケージ	MT Developer2/GX Works3の総称
MELSOFT MT Works2	モーションコントローラエンジニアリングソフトウェアSW1DND-MTW2の総称製品名
MT Developer2	モーションコントローラエンジニアリングソフトウェア「MELSOFT MT Works2」に含まれるプログラミングソフト ウェアの略称
GX Works3	MELSECシーケンサソフトウェアパッケージSW1DND-GXW3の総称製品名
手動パルス	手動パルス発生器の略称
シリアルABS同期エンコーダ もしくはQ171ENC-W8	シリアルABS同期エンコーダ(Q171ENC-W8)の略称
SSCNETⅢ/H ^{*1}	モーションコントローラ⇄サーボアンプ間高速同期ネットワーク
SSCNETⅢ ^{*1}	
SSCNETⅢ(H)	SSCNETⅢ/H、SSCNETⅢの総称
絶対位置システム	絶対位置対応のサーボモータおよびサーボアンプを使用したシステムの総称
インテリジェント機能ユニット	A/D、D/A変換ユニットなど、入出力以外の機能を持つユニットの総称
SSCNETⅢ/Hヘッドユニット ^{*1}	MELSEC-Lシリーズ SSCNETⅢ/Hヘッドユニット(LJ72MS15)の略称
光分岐ユニットもしくはMR-MV200	SSCNETⅢ/H対応光分岐ユニット(MR-MV200)の略称
センシングユニット	SSCNETⅢ/H対応センシングユニットMR-MT2000シリーズの総称
センシングSSCNETⅢ/Hヘッドユニッ ト ^{*1} もしくはMR-MT2010	SSCNETⅢ/Hヘッドユニット(MR-MT2010)の略称
センシング拡張ユニット	入出力ユニット(MR-MT2100)、パルス入出力ユニット(MR-MT2200)、アナログ入出力ユニット(MR-MT2300)、エン コーダI/Fユニット(MR-MT2400)の総称
センシング入出力ユニットもしくは MR-MT2100	入出力ユニット(MR-MT2100)の略称
センシングパルス入出力ユニットもし くはMR-MT2200	パルス入出力ユニット(MR-MT2200)の略称
センシングアナログ入出力ユニットも しくはMR-MT2300	アナログ入出力ユニット(MR-MT2300)の略称
センシングエンコーダI/Fユニットも しくはMR-MT2400	エンコーダI/Fユニット(MR-MT2400)の略称

*1 SSCNET: Servo System Controller NETwork

マニュアルの読み方

本マニュアルに使われている数値の表示について

■軸No.の表示について

位置決め専用信号の説明において、M3200+20nなどのnは、下表の通り軸No.に対応する数値を示しています。

軸No.	n	軸No.	n	軸No.	n	軸No.	n	軸No.	n	軸No.	n	軸No.	n	軸No.	n
1	0	9	8	17	16	25	24	33	32	41	40	49	48	57	56
2	1	10	9	18	17	26	25	34	33	42	41	50	49	58	57
3	2	11	10	19	18	27	26	35	34	43	42	51	50	59	58
4	3	12	11	20	19	28	27	36	35	44	43	52	51	60	59
5	4	13	12	21	20	29	28	37	36	45	44	53	52	61	60
6	5	14	13	22	21	30	29	38	37	46	45	54	53	62	61
7	6	15	14	23	22	31	30	39	38	47	46	55	54	63	62
8	7	16	15	24	23	32	31	40	39	48	47	56	55	64	63

- ・ R16MTCPUでは軸No.1～16の範囲(n=0～15)，R32MTCPUでは軸No.1～32の範囲(n=0～31)が有効です。
- ・ 各軸に対応するデバイスNo.は、下記のように計算してください。

例

Q互換配置方式で軸No.32の場合

M3200+20n([Rq.1140]停止指令)=M3200+20×31=M3820

M3215+20n([Rq.1155]サーボOFF指令)=M3215+20×31=M3835

ただし、M10440+10nなど同期エンコーダ軸ステータス，同期エンコーダ軸指令信号，同期エンコーダ軸モニタデバイス，同期エンコーダ軸制御デバイスのnは、下表の通り同期エンコーダ軸No.に対応する数値を示しています。

同期エンコーダ軸No.	n	同期エンコーダ軸No.	n	同期エンコーダ軸No.	n
1	0	5	4	9	8
2	1	6	5	10	9
3	2	7	6	11	10
4	3	8	7	12	11

- ・ 各同期エンコーダに対応するデバイスNo.は、下記のように計算してください。

例

Q互換配置方式で同期エンコーダ軸No.12の場合

M10440+10n([St.320]同期エンコーダ軸設定有効フラグ)=M10440+10×11=M10550

D13240+20n([Md.320]同期エンコーダ軸現在値)=D13240+20×11=D13460

■マシンNo.の表示について

位置決め専用信号の説明において、M43904+32mなどのmは、下表の通りマシンNo.に対応する数値を示しています。

マシンNo.	m	マシンNo.	m
1	0	5	4
2	1	6	5
3	2	7	6
4	3	8	7

- ・ 各マシンに対応するデバイスNo.は、下記のように計算してください。

例

R標準配置方式でマシンNo.8の場合

M43904+32m([St.2120]マシンエラー検出)=M43904+32×7=M44128

D53168+128m([Md.2020]マシンタイプ)=D53168+128×7=D54064

■Gコード制御における系統No.の表示について

位置決め専用信号の説明において、D54496+128sなどのsは、下表の通り系統No.に対応する数値を示しています。

系統No.	s
1	0
2	1

- 各系統に対応するデバイスNo.は、下記のように計算してください。

例

R標準配置方式で系統No.2の場合

$D54440.0 + 4s$ ([St.3208]Gコード制御中) = $D54440.0 + 4 \times 1 = D54444.0$

$D54496 + 128s$ ([Md.3016]系統内軸数) = $D54496 + 128 \times 1 = D54624$

■Gコード制御における系統No.と軸No.の表示について

位置決め専用信号の説明において、D54278+16snなどのsnは、下表の通り系統No.と軸No.に対応する数値を示しています。

系統No.	軸No.	sn	系統No.	軸No.	sn
1	1	0	2	1	8
	2	1		2	9
	3	2		3	10
	4	3		4	11
	5	4		5	12
	6	5		6	13
	7	6		7	14
	8	7		8	15

- 各系統に対応するデバイスNo.は、下記のように計算してください。

例

R標準配置方式で系統No.2，軸No.8の場合

$D54448.0 + 2sn$ ([St.3076]スレービングゼロ) = $D54448.0 + 2 \times 15 = D54478.0$

$D54754 + 32sn$ ([Md.3146]回転軸設定状態) = $D54754 + 32 \times 15 = D55234$

本マニュアルに使われているデバイス番号の表示について

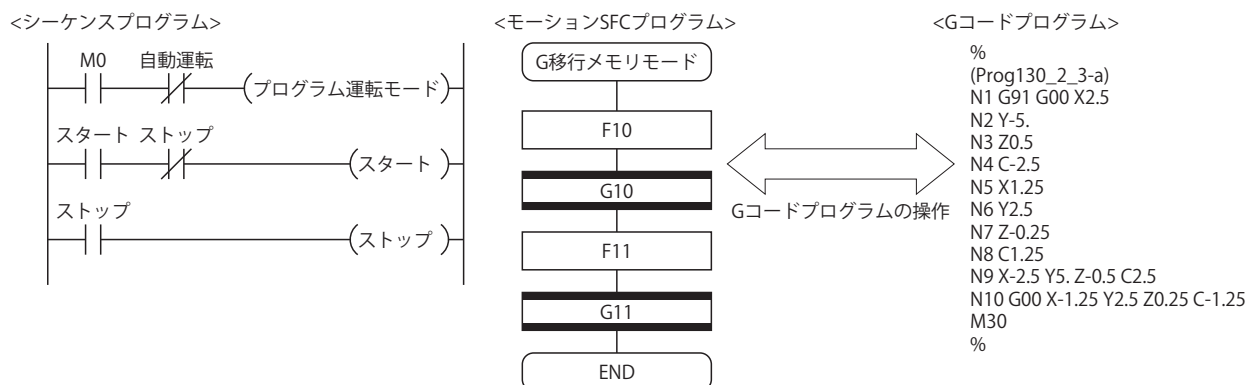
位置決め専用信号のデバイス番号に記載されている，“[Rq.1140]停止指令(R: M34480+32n/Q: M3200+20n)”などのRとQは、下記の通り、使用するデバイス配置方式のデバイス番号を示しています。RとQが記載されていない場合は、デバイス配置方式で共通のデバイス番号となります。

記号	デバイス配置方式
R	R標準配置方式
Q	Q互換配置方式

1 概要

1.1 Gコード制御の概要

- Gコード制御は、GコードのプログラムをモーションCPUで解析して制御を行います。工作機械などの切削加工や多様な加工制御に適応できます。
- Gコード制御を使用するには、Gコード制御用アドオンライブラリをインストールする必要があります。
- 制御に必要なパラメータを設定することで、最大2系統のシステムを構築することができます。1系統内は最大8軸、同時制御軸数は最大4軸まで可能です。プログラム制御は、系統ごとに独立して同時に制御することができます。(系統とは、Gコード制御を実行する軸構成のグループのことで、サーボネットワークの系統とは異なります。)
- Gコードプログラムは、パソコン上にてテキスト形式で作成します。作成したプログラムファイルは、MT Developer2を使用して、モーションCPUへの書き込み／読出しを行います。また、モーションCPUに挿入したSDメモ리카ードからでも、ブート時ファイル転送機能を使用して、Gコードプログラムを転送することができます。
- Gコードプログラムの始動は、シーケンスプログラムやモーションSFCプログラムからGコード制御用デバイス进行操作することで、Gコード制御モード(系統ごと)へ移行し、プログラムを自動起動／停止、またはMコードなどの補助機能まで自動的に制御を行うことができます。軸の制御は、パラメータで各軸に割当てたアルファベットの座標語(X, Y, Z, A, B, C, U, V, W)をGコードプログラムの各命令で指定します。



- モーションCPUにおける他の機能との組合せも可能です。例えば、アドバンス同期制御を使用し、任意の制御軸に同じ指令を送ることで、多軸同期などの高度な動作が可能となります。他の機能を併用することで、加工制御のみでなく多様なシステムを構築することができます。

1.2 性能仕様

Gコード制御仕様

項目		仕様
Gコード制御構成軸数(系統内)		最大8軸
同時輪郭制御軸数		最大4軸
系統数		最大2系統
運転モード		メモリ運転
プログラムサイズ/1プログラム		最大512k/バイト
プログラム数		最大256本(O001～O256)
プログラム合計サイズ		最大2M/バイト
Gコード		G00, G01, G02, G03, G04, G09, G12.1, G13.1, G17, G18, G19, G38, G39, G40, G40.1, G41, G41.1, G42, G42.1, G43, G44, G49, G52, G53, G54, G55, G56, G57, G58, G59, G61, G61.1, G62, G64, G65, G68, G69, G90, G91, G94
Mコード		M-99999999～M99999999(1ブロック内に最大4組指令可能)
特定のMコード		M00, M01, M02, M30, M98, M99
演算指令	演算子	+, -, *, /, =, OR, XOR, MOD, AND
	関数	SIN, COS, TAN, ASIN, ACOS, ATAN, SQRT, ABS, BIN, BCD, ROUND, FIX, FUP, LN, EXP, POW
制御指令		IF～GOTO, IF～THEN～ELSE～ENDIF, WHILE～DO～END
Gコード制御の制御単位		直線軸: mm, pulse 回転軸: degree, pulse
データ単位	最小指令単位	直線軸: 0.0001[mm] 回転軸: 0.0001[degree]
補間機能		直線補間, 円弧補間, 極座標補間
送り機能		早送り速度, 切削速度, 減速チェック
オーバーライド機能		早送りオーバーライド, 切削送りオーバーライド, オーバライドキャンセル, 自動コーナオーバーライド
Gコード機能の補助機能	Mコード単独出力	M00, M01, M02, M30
	補助機能完了	補助機能完了1(FIN1), 補助機能完了2(FIN2)
工具補正機能	補正メモリ	工具長補正, 工具径補正
	補正組数	最大40組
座標系		基本機械座標系, ワーク座標系, ローカル座標系, 自動座標系, 回転軸用座標系
支援機能		自動運転起動, 自動運転休止, シングルブロック, Gコード制御のリセット, プログラム座標回転, 法線制御
高精度制御		補間前加減速, 最適速度制御, ベクトル精補間, 円弧入口/出口速度制御
タンデム機能		あり

1.3 Gコード制御のアドオンライブラリ構成

Gコード制御のアドオンライブラリ

Gコード制御を行うために使用するアドオンライブラリを下記に示します。

用途	形名 ^{*1}	アドオンライブラリ名
Gコード制御	SW10DND-GCDO	Gcode_Ctrl.adm

^{*1} □はライセンス本数を示す。(01, 05, 10, 15, 20)

ファイルサイズ／使用メモリ量

アドオンライブラリのファイルサイズ、およびメモリ使用量を下記に示します。

アドオンライブラリ名	ファイルサイズ[バイト]	メモリ使用量[バイト]
Gcode_Ctrl.adm	193818(約189k/バイト)	5697152(約5.4M/バイト)

1.4 バージョンによる機能の制約

アドオンライブラリ、本体OS ソフトウェア、およびエンジニアリングソフトウェアのバージョンにより、使用できる機能に制約があります。各バージョンと機能の組合せを示します。

機能	アドオンライブラリバージョン		本体OSソフトウェアバージョン	エンジニアリングソフトウェア
	メジャーバージョン	マイナーバージョン		MELSOFT MT Works2
ローカル座標系設定	01	02	14	1.140W
極座標補間	01	02	14	1.140W
高精度制御モード	01	02	14	1.140W
プログラム座標回転	01	02	14	1.140W
タンデム制御	01	02	14	1.140W
Gコード制御パラメータのパラメータ変更機能	01	02	14	1.140W
アドオンライブラリライセンス認証機能	01	02	14	1.140W
サブプログラム制御機能	01	03	15	1.145B
変数指令	01	03	15	1.145B
演算指令	01	03	15	1.145B
制御指令	01	03	15	1.145B
マクロ呼出し機能	01	04	16	1.150G

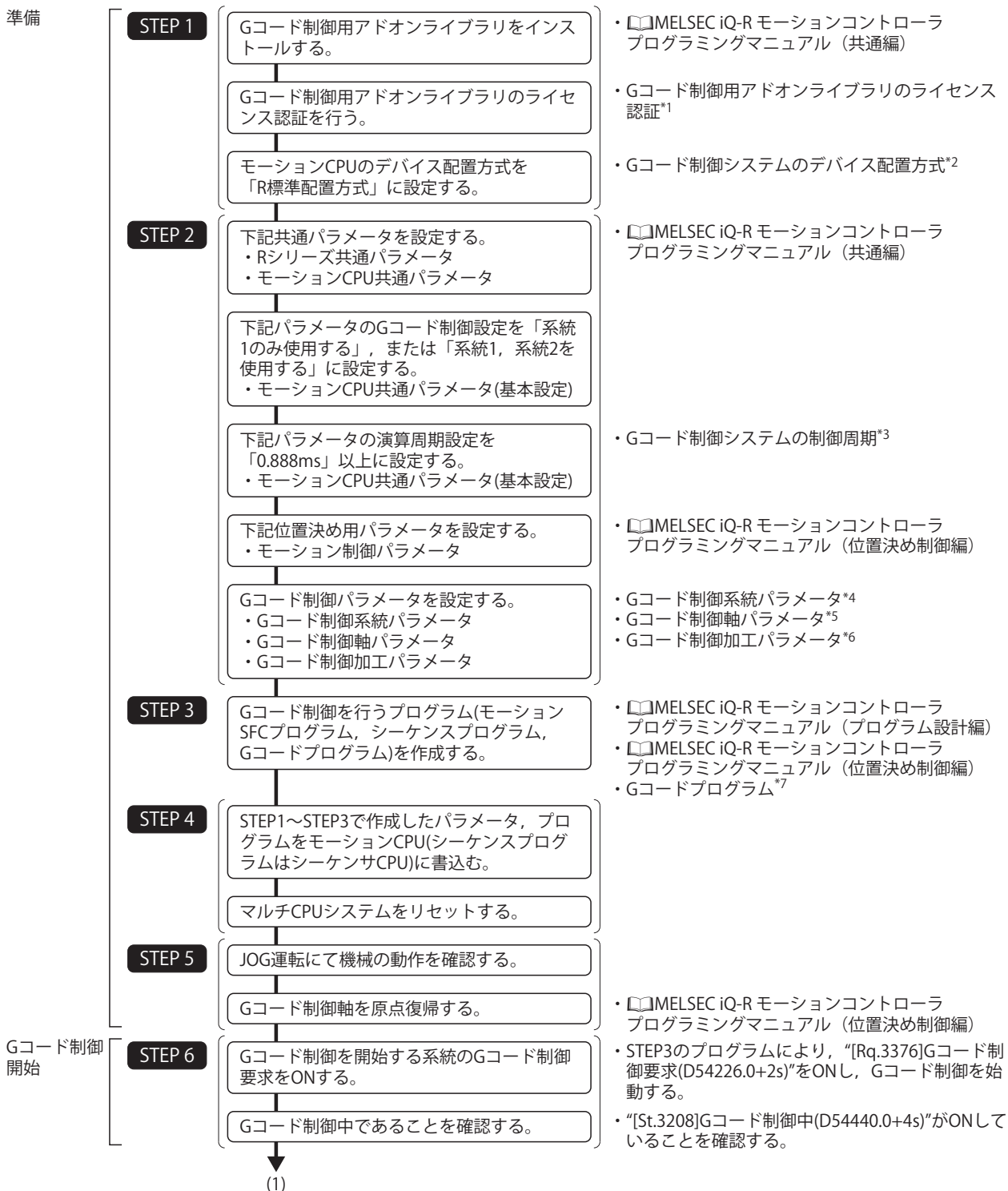
2 システムの立上げ

Gコード制御を行うための手順を、下記に示します。

2.1 Gコード制御システムの立上げ

Gコード制御のシステム立上げ手順を、下記に示します。

準備



Gコード制御
開始



*1 Gコード制御用アドオンライブラリは, ライセンス認証が必要です。(☞ 23ページ Gコード制御用アドオンライブラリのライセンス認証)

*2 Gコード制御システムのデバイス配置方式(☞ 24ページ Gコード制御システムのデバイス配置方式)

*3 Gコード制御の制御周期(☞ 23ページ Gコード制御の制御周期)

*4 Gコード制御系統パラメータ(☞ 71ページ Gコード制御系統パラメータ)

*5 Gコード制御軸パラメータ(☞ 80ページ Gコード制御軸パラメータ)

*6 Gコード制御加工パラメータ(☞ 87ページ Gコード制御加工パラメータ)

*7 Gコードプログラム(☞ 89ページ Gコード制御用プログラム)

2.2 Gコード制御用アドオンライブラリのライセンス認証

Gコード制御アドオンライブラリは、ライセンス認証が必要なアドオンライブラリです。ライセンスを購入しないでGコード制御用アドオンライブラリをインストールすると、ライセンス期限が設定されます。ライセンス期限オーバーとなると、中度エラー (エラーコード: 3081H) となり、Gコード制御用アドオンライブラリは使用できなくなります。

ライセンス認証の詳細は、下記を参照してください。

📖 MELSEC iQ-R モーションコントローラプログラミングマニュアル(共通編)

ライセンス期限

Gコード制御用アドオンライブラリをインストールした場合のライセンス期限を下記に示します。

ライセンス期限	備考
1500時間	稼働時間(マルチCPUシステム電源ON中の時間)に応じて、ライセンス期限(ライセンス認証が必要になるまでの残り時間)が減っていきます。 マルチCPUシステム電源OFF中の時間はカウントしません。

2.3 Gコード制御の制御周期

- ・ [モーションCPU共通パラメータ]⇒[基本設定]⇒“演算周期設定”にて、演算周期を「0.888ms」以上に設定してください。
Gコード制御は“モーション設定演算周期(SD523)”が0.888[ms]以上の場合に下記の周期で動作し、この周期は“[Md.3000]Gコード制御設定演算周期(D54480)”に格納されます。(Gコード制御以外の機能は、“モーション設定演算周期(SD523)”の周期で動作します。) “モーション設定演算周期(SD523)”が0.888[ms]より短いと、中度エラー (エラーコード: 30FEH) となり、Gコード制御は使用できません。

モーション設定演算周期(SD523)	[Md.3000]Gコード制御設定周期(D54480)	備考
0.888[ms]	3.555[ms]	モーション設定演算周期の4倍の間隔で動作します。
1.777[ms]	7.111[ms]	
3.555[ms]	14.222[ms]	
7.111[ms]	28.444[ms]	

- ・ [モーションCPU共通パラメータ]⇒[基本設定]⇒“低速演算周期倍率設定”にて、低速演算周期倍率を「使用しない」に設定してください。低速演算周期倍率設定を設定すると中度エラー (エラーコード: 30F9H) となります。
- ・ Gコード制御の開始から完了するまでの時間は、Gコード設定演算周期ごとに“[Md.3001]Gコード制御演算周期(D54481)”へ格納されます。格納値の最大値は“[Md.3002]Gコード制御最大演算周期(D54482)”に格納されます。
- ・ “[Md.3001]Gコード制御演算周期(D54481)”の時間が“[Md.3000]Gコード制御設定演算周期(D54480)”を超えると、“[St.3272]Gコード制御演算周期オーバー発生フラグ(D54438.0)”がONします。また、“[Md.3001]Gコード制御演算周期(D54481)”の時間に余裕がある状態でも、モーション演算にかかった時間の合計(モーション演算周期(SD522))が、“モーション設定演算周期(SD523)”を超えた場合、“[St.1046]演算周期オーバーフラグ(R: M30054/Q: M2054)”がONします。
- ・ “[St.3272]Gコード制御演算周期オーバー発生フラグ(D54438.0)”，または “[St.1046]演算周期オーバーフラグ(R: M30054/Q: M2054)”がONした場合は、下記の対策を行ってください。
 - ・ [モーションCPU共通パラメータ]⇒[基本設定]⇒“演算周期設定”にて、演算周期を大きい値に変更する。
 - ・ モーションSFCプログラムにて、イベントタスク、NMI タスクの命令実行数を減らす。

2.4 Gコード制御システムのデバイス配置方式

Gコード制御を使用する場合のデバイス配置方式は「R標準配置方式」を推奨します。デバイス配置方式の詳細は、下記を参照してください。

📖 MELSEC iQ-R モーションコントローラプログラミングマニュアル(共通編)

Gコード制御で使用するデバイスについては、Gコード制御専用信号を参照してください。(📖 27ページ Gコード制御専用信号)

注意事項

R32MTCPU/R16MTCPUにてデバイス配置方式を「Q互換配置方式」でGコード制御を行う場合、[Rシリーズ共通パラメータ]⇒[モーションCPUユニット]⇒[CPUパラメータ]⇒「デバイス関連設定」⇒「デバイス点数／ラッチ設定」にてデータレジスタ(D)を下記の点数以上に設定してください。デフォルトのデバイス点数のままで、アドオンライブラリのインストールを行うと、中度エラー(エラーコード: 308FH(詳細コード: 0500H))となり、Gコード制御は使用できません。

デバイス	Gコード制御使用時の最小設定点数
データレジスタ(D)	56320点(55k)

デバイス点数／ラッチ設定の詳細は、下記を参照してください。

📖 MELSEC iQ-R モーションコントローラプログラミングマニュアル(共通編)

2.5 Gコード制御システムの始動／終了

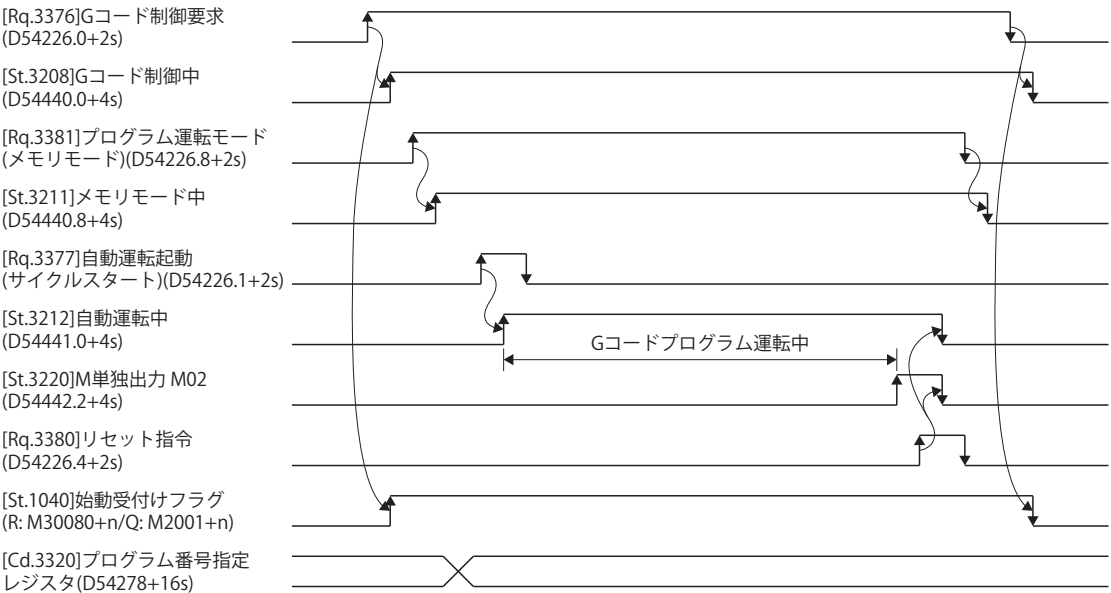
Gコード制御軸パラメータで設定した軸を、Gコードプログラムで制御します。

系統ごとに「[Rq.3376]Gコード制御要求(D54226.0+2s)」のON/OFFでGコード制御を始動／終了します。Gコード制御の状態は「[St.3208]Gコード制御中(D54440.0+4s)」で確認できます。

「[Rq.3381]プログラム運転モード(メモリモード)(D54226.8+2s)」をONすることにより、自動運転のメモリモードを選択します。「[Rq.3381]プログラム運転モード(メモリモード)(D54226.8+2s)」をOFFすることにより、メモリモードを解除します。

「[Rq.3376]Gコード制御要求(D54226.0+2s)」をOFFするときは、メモリモードを解除してから行ってください。

「[Cd.3320]プログラム番号指定レジスタ(D54278+16s)」に始動するプログラム番号をセットし、「[Rq.3377]自動運転起動(サイクルスタート)(D54226.1+2s)」をOFF→ONすることで、メモリ運転を起動できます。メモリ運転の運転中は「[St.3212]自動運転中(D54441.0+4s)」がONします。



2.6 Gコード制御システムの停止動作

Gコード制御中、Gコード制御軸に以下の停止要因が発生した場合、“[St.3215]Gコード制御終了中(D54441.3+4s)”がONし、系統内全軸の停止処理後、“[St.3208]Gコード制御中(D54440.0+4s)”， “[St.3215]Gコード制御終了中(D54441.3+4s)”， “[St.1040]始動受け付けフラグ(R: M30080+n/Q: M2001+n)”がOFFとなり、Gコード制御は終了します。終了後はGコード制御関連のデバイス更新が停止します。

停止要因	停止処理
マルチCPUシステムリセット操作	即停止
マルチCPUシステム電源OFF	
モーションCPU WDTエラー	
緊急停止(モーションコントローラの緊急停止(デバイス))	
強制停止(サーボアンプの強制入力端子)	
サーボエラー発生	
サーボアンプの制御電源ON→OFF	
外部入力信号(STOP/FLS/RLS)の入力	減速停止
モーションCPUのRUN→STOP	
“[Rq.3376]Gコード制御要求(D54226.0+2s)”ON→OFF	

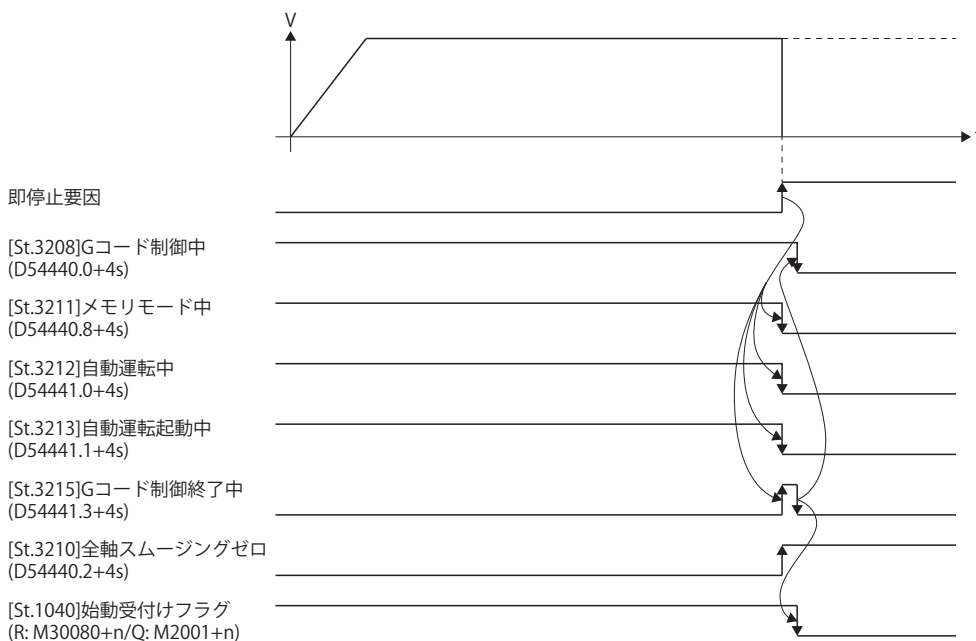
Point

- 停止要因は演算周期ごとにチェックを行いますが、停止処理はGコード制御演算周期で実行します。
- Gコード制御中、パラメータブロックの設定は無効となります。
- 停止要因発生時は “[St.3215]Gコード制御終了中(D54441.3+4s)” がONするタイミングで、下記の信号がOFFします。
 - [St.3211]メモリモード中(D54440.8+4s)
 - [St.3212]自動運転中(D54441.0+4s)
 - [St.3213]自動運転起動中(D54441.1+4s)
 - [St.3214]自動運転休止中(D54441.2+4s)
 - [St.3216]リセット中(D54441.8+4s)
 - [St.3217]リセット完了(D54441.9+4s)

- Gコード制御中は、下記の停止要因が発生しても処理は無効となり、停止しません。また、エラーも発生しません。
 - 各軸ソフトウェアストロークリミットの検出
 - 停止指令の入力
 - 急停止指令の入力
 - サーボモータ最大回転速度チェック

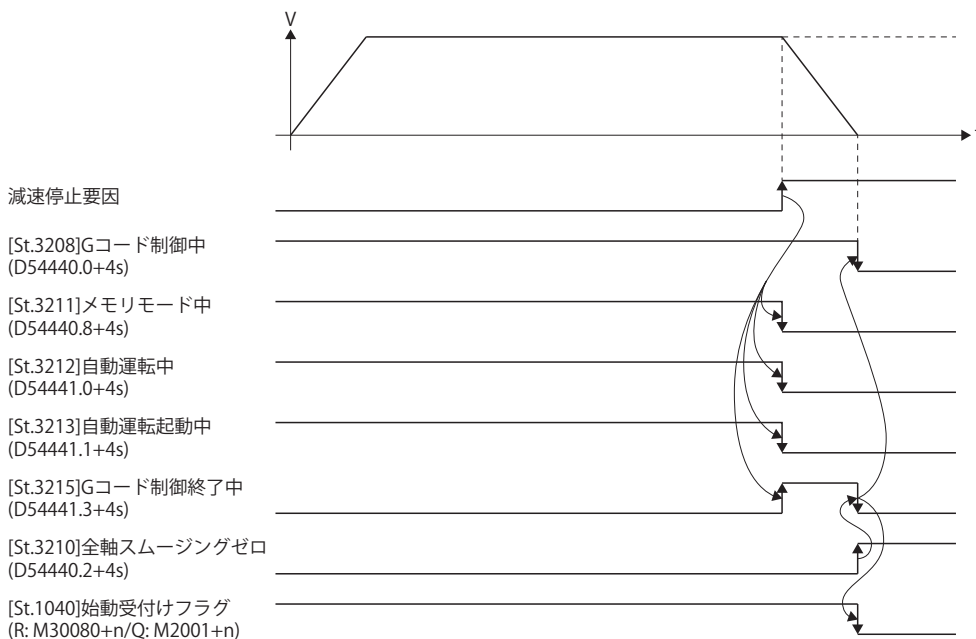
即停止

減速処理を行わない停止です。モーションCPUは指令を即停止しますが、サーボアンプは偏差カウンタの溜まりパルス分償行します。即停止時のタイミングチャートを以下に示します。



減速停止

Gコード制御パラメータの加減速時定数に従って系統内の全軸が減速を開始し、停止後にGコード制御が終了します。減速停止時のタイミングチャートを以下に示します。




3 Gコード制御専用信号

Gコード制御で使用するGコード制御用デバイスを以下に示します。

デバイス名	デバイス範囲	
	R標準配置方式	Q互換配置方式
データレジスタ(D)	D54192～D55583(1392点)	








3

Point

- R32MTCPU/R16MTCPUにてQ互換配置方式で使用する場合は、デバイス点数／ラッチ設定にてデバイス範囲をGコード制御使用時の最小設定範囲以上に変更する必要があります。デフォルトのデバイス点数のままで、アドオンライブラリのインストールを行うと、中度エラー (エラーコード: 308FH(詳細コード: 0500H))となります。(24ページ Gコード制御システムのデバイス配置方式)
- 本マニュアルは、Gコード制御時に使用する内部リレー、データレジスタの内容のみ記載しています。記載されていないデバイスについては、下記を参照してください。
 MELSEC iQ-R モーションコントローラプログラミングマニュアル(位置決め制御編)
- Gコード制御用デバイスは、Gコード制御中のみ有効です。(Gコード制御用デバイスの一部は、Gコード制御中以外でも有効なデバイスがあります。)

データレジスタ一覧

■R標準配置方式/Q互換配置方式

デバイス番号	記号	用途分類	参照
D54192 ～	—	ユーザ使用不可 (32点)	—
D54224 ～	[Rq.3344]	Gコード制御共通指令信号 (2点)	 28ページ Gコード制御共通指令信号
D54226 ～	[Rq.3376]～[Rq.3385]	Gコード制御系統指令信号 (4点)	 33ページ Gコード制御系統指令信号
D54230 ～	—	ユーザ使用不可 (32点)	—
D54262 ～	[Cd.3305]	Gコード制御共通制御デバイス (16点)	 29ページ Gコード制御共通制御デバイス
D54278 ～	[Cd.3320]～[Cd.3322]	Gコード制御系統制御デバイス (32点)	 37ページ Gコード制御系統制御デバイス
D54310 ～	—	ユーザ使用不可 (128点)	—
D54438 ～	[St.3272]	Gコード制御共通ステータス (2点)	 30ページ Gコード制御共通ステータス
D54440 ～	[St.3208]～[St.3225], [St.3234]	Gコード制御系統ステータス (8点)	 38ページ Gコード制御系統ステータス
D54448 ～	[St.3076]	Gコード制御軸ステータス (32点)	 61ページ Gコード制御軸ステータス
D54480 ～	[Md.3000]～[Md.3004]	Gコード制御共通モータデバイス (16点)	 31ページ Gコード制御共通モータデバイス
D54496 ～	[Md.3016]～[Md.3070], [Md.3074]	Gコード制御系統モータデバイス (256点)	 45ページ Gコード制御系統モータデバイス
D54752 ～	[Md.3144]～[Md.3150], [Md.3152]～[Md.3154]	Gコード制御軸モータデバイス (512点)	 63ページ Gコード制御軸モータデバイス
D55264 ～ D55583	[Md.3178]～[Md.3180]	Gコード制御系統モータデバイス(拡張) (320点)	 55ページ Gコード制御系統モータデバイス (拡張)

3.1 Gコード制御共通指令信号

デバイス番号		記号	信号名称	リフレッシュ 周期	取込周期	信号種別
R標準配置方式	Q互換配置方式					
D54224.0		—	ユーザ使用不可	—	—	—
D54224.1						
D54224.2						
D54224.3						
D54224.4						
D54224.5						
D54224.6						
D54224.7						
D54224.8						
D54224.9						
D54224.A						
D54224.B						
D54224.C						
D54224.D						
D54224.E						
D54224.F						
D54225.0		Rq.3344	運転中プログラム取込み要求	—	メイン周期	指令信号
D54225.1		—	ユーザ使用不可	—	—	—
D54225.2						
D54225.3						
D54225.4						
D54225.5						
D54225.6						
D54225.7						
D54225.8						
D54225.9						
D54225.A						
D54225.B						
D54225.C						
D54225.D						
D54225.E						
D54225.F						

[Rq.3344]運転中プログラム取込み要求(D54225.0)

プログラム実行中にGコードプログラムファイルを取込むための要求を行う信号です。

設定値	内容
ON	取込み要求する
OFF	取込み要求しない

3.2 Gコード制御共通制御デバイス

デバイス番号		記号	信号名称	リフレッシュ 周期	取込周期	信号種別
R標準配置方式	Q互換配置方式					
D54262		—	ユーザ使用不可	—	—	—
D54263						
D54264		Cd.3305	運転中プログラム取込みプログラム番号	—	運転中プログラム取込み要求ON時	指令デバイス
D54265		—	ユーザ使用不可	—	—	—
D54266						
D54267						
D54268						
D54269						
D54270						
D54271						
D54272						
D54273						
D54274						
D54275						
D54276						
D54277						

3

[Cd.3305]運転中プログラム取込みプログラム番号(D54264)

“[Rq.3344]運転中プログラム取込み要求(D54225.0)”にて指定するプログラム番号のGコードプログラムを取り込むために設定するレジスタです。

項目	設定範囲
プログラム番号	1～256

3.3 Gコード制御共通ステータス

デバイス番号		記号	信号名称	リフレッシュ 周期	取込周期	信号種別
R標準配置方式	Q互換配置方式					
D54438.0		St.3272	Gコード制御演算周期オーバ発生フラグ	Gコード制御演算周期	—	ステータス信号
D54438.1		—	ユーザ使用不可	—	—	—
D54438.2						
D54438.3						
D54438.4						
D54438.5						
D54438.6						
D54438.7						
D54438.8						
D54438.9						
D54438.A						
D54438.B						
D54438.C						
D54438.D						
D54438.E						
D54438.F						
D54439.0						
D54439.1						
D54439.2						
D54439.3						
D54439.4						
D54439.5						
D54439.6						
D54439.7						
D54439.8						
D54439.9						
D54439.A						
D54439.B						
D54439.C						
D54439.D						
D54439.E						
D54439.F						

[St.3272]Gコード制御演算周期オーバ発生フラグ(D54438.0)

Gコード制御演算処理が、Gコード制御演算周期以内に完了しなかった場合にONするフラグです。
OFFさせる場合には、下記の操作を行ってください。

- マルチCPUシステムの電源OFF→ON
- マルチCPUシステムのリセット

■演算周期オーバ時の対策

- [モーションCPU共通パラメータ]⇒[基本設定]⇒“演算周期設定”にて、演算周期を大きい値に変更する。
- モーションSFCプログラムにて、イベントタスク、NMIタスクの命令実行数を減らす。

3.4 Gコード制御共通モニタデバイス

デバイス番号		記号	信号名称	リフレッシュ 周期	取込周期	信号種別
R標準配置方式	Q互換配置方式					
D54480		Md.3000	Gコード制御設定演算周期	STOP→RUN時	—	モニタデバイス
D54481		Md.3001	Gコード制御演算周期	Gコード制御演算周期		
D54482		Md.3002	Gコード制御最大演算周期			
D54483		—	ユーザ使用不可	—	—	—
D54484						
D54485						
D54486						
D54487						
D54488						
D54489						
D54490						
D54491						
D54492		Md.3003	運転中プログラム取込み状態	メイン周期	—	モニタデバイス
D54493		Md.3004	運転中プログラム取込みエラー情報			
D54494		—	ユーザ使用不可	—	—	—
D54495						

[Md.3000]Gコード制御設定演算周期(D54480)

Gコード制御の演算周期を[μs]単位で格納します。

Gコード制御の制御周期は，“モーション設定演算周期(SD523)”により以下の周期で動作します。

“モーション設定演算周期(SD523)”が0.888[ms]より短いと，中度エラー（エラーコード: 30FEH）となり，Gコード制御は使用できません。

モーション設定演算周期(SD523)	[Md.3000]Gコード制御設定演算周期(D54480)
0.888[ms]	3.555[ms]
1.777[ms]	7.111[ms]
3.555[ms]	14.222[ms]
7.111[ms]	28.444[ms]

[Md.3001]Gコード制御演算周期(D54481)

Gコード制御演算周期ごとにGコード制御演算処理にかかる時間を[μs]単位で格納します。

[Md.3002]Gコード制御最大演算周期(D54482)

マルチCPUシステムの電源投入後，Gコード制御演算周期ごとにGコード制御演算処理にかかる最大時間を[μs]単位で格納します。

[Md.3003]運転中プログラム取込み状態(D54492)

運転中プログラム取込みの状態が格納されます。

モニタ値	内容
0	要求なし
1	実行中
2	正常完了
3	異常完了

[Md.3004]運転中プログラム取込みエラー情報(D54493)

“[Md.3003]運転中プログラム取込み状態(D54492)”が異常完了時に，エラー情報が格納されます。

モニタ値	内容
0	エラーなし
1	該当プログラム運転中
2	プログラムファイルなし
3	ファイルフォーマット異常
4	取込みエリアサイズオーバ
5	プログラム番号範囲外
6	STOP中取込み要求
7	システム異常

3.5 Gコード制御系統指令信号

デバイス番号		信号名称
R標準配置方式	Q互換配置方式	
D54226～D54227		系統1のGコード制御系統指令信号
D54228～D54229		系統2のGコード制御系統指令信号

・各系統の詳細内容

デバイス番号	記号	信号名称	リフレッシュ 周期	取込周期	信号種別
R標準配置方式	Q互換配置方式				
D54226.0+2s	Rq.3376	Gコード制御要求	—	メイン周期/ Gコード制御演 算周期	指令信号
D54226.1+2s	Rq.3377	自動運転起動(サイクルスタート)		Gコード制御演 算周期	
D54226.2+2s	Rq.3378	自動運転休止(フィードホールド)			
D54226.3+2s	Rq.3379	シングルブロック			
D54226.4+2s	Rq.3380	リセット指令			
D54226.5+2s	—	ユーザ使用不可	—	—	—
D54226.6+2s					
D54226.7+2s					
D54226.8+2s	Rq.3381	プログラム運転モード(メモリモード)	—	Gコード制御演 算周期	指令信号
D54226.9+2s	—	ユーザ使用不可	—	—	—
D54226.A+2s					
D54226.B+2s					
D54226.C+2s	Rq.3384	マクロシングル	—	Gコードプログ ラム始動時	指令信号
D54226.D+2s	—	ユーザ使用不可	—	—	—
D54226.E+2s					
D54226.F+2s					
D54227.0+2s	Rq.3382	補助機能完了1(FIN1)	—	Gコード制御演 算周期	指令信号
D54227.1+2s	Rq.3383	補助機能完了2(FIN2)			
D54227.2+2s	—	ユーザ使用不可	—	—	—
D54227.3+2s					
D54227.4+2s	Rq.3385	G65引数初期化	—	Gコードプログ ラム始動時	指令信号
D54227.5+2s	—	ユーザ使用不可	—	—	—
D54227.6+2s					
D54227.7+2s					
D54227.8+2s					
D54227.9+2s					
D54227.A+2s					
D54227.B+2s					
D54227.C+2s					
D54227.D+2s					
D54227.E+2s					
D54227.F+2s					

[Rq.3376]Gコード制御要求(D54226.0+2s)

該当する系統の“[Rq.3376]Gコード制御要求(D54226.0+2s)”をONすることで、この系統のGコード制御を始動する信号です。“[St.3208]Gコード制御中(D54440.0+4s)”がONの状態のときに、“[Rq.3376]Gコード制御要求(D54226.0+2s)”をOFFすると、Gコード制御を終了します。詳細は、Gコード制御システムの始動／終了を参照ください。(24ページ Gコード制御システムの始動／終了)

設定値	内容	取込周期
ON	Gコード制御要求	メイン周期
OFF	Gコード制御終了	Gコード制御演算周期

[Rq.3377]自動運転起動(サイクルスタート)(D54226.1+2s)

- Gコードプログラムの起動、自動運転休止後の再起動、ブロック停止の再起動を行うための指令です。
- “[Rq.3377]自動運転起動(サイクルスタート)(D54226.1+2s)”の立上り(OFF→ON)により、“[Rq.3378]自動運転休止(フィードホールド)(D54226.2+2s)”をONするまで、または“[Rq.3379]シングルブロック(D54226.3+2s)”のONによりブロック停止するまで、“[St.3213]自動運転起動中(D54441.1+4s)”がONします。
- 下記の場合、“[Rq.3377]自動運転起動(サイクルスタート)(D54226.1+2s)”をONしても無効となります。
 - すでに“[St.3213]自動運転起動中(D54441.1+4s)”がONのとき。
 - “[Rq.3378]自動運転休止(フィードホールド)(D54226.2+2s)”がONのとき。
 - “[St.3216]リセット中(D54441.8+4s)”がONのとき。
 - エラーが発生しているとき。
- 下記の場合、自動運転休止、または停止、ブロック停止になります。
 - “[Rq.3378]自動運転休止(フィードホールド)(D54226.2+2s)”をOFF→ONしたとき。
 - “[Rq.3380]リセット指令(D54226.4+2s)”をONしたとき。
 - プログラム運転モードでなくなったとき。
 - “[Rq.3379]シングルブロック(D54226.3+2s)”がONした後、実行中のブロックが終了したとき。

[Rq.3378]自動運転休止(フィードホールド)(D54226.2+2s)

- 自動運転により移動中、途中で減速停止を行うための指令です。
- 再起動は、“[Rq.3378]自動運転休止(フィードホールド)(D54226.2+2s)”をOFFした後、“[Rq.3377]自動運転起動(サイクルスタート)(D54226.1+2s)”の立上り(OFF→ON)により行います。“[Rq.3378]自動運転休止(フィードホールド)(D54226.2+2s)”がONの状態では、“[Rq.3377]自動運転起動(サイクルスタート)(D54226.1+2s)”をOFF→ONすると軽度エラー(エラーコード: 1FC2H(詳細コード: 0202H))となります。
- “[Rq.3378]自動運転休止(フィードホールド)(D54226.2+2s)”の立上り(OFF→ON)により、“[St.3213]自動運転起動中(D54441.1+4s)”がONの場合、自動運転休止となり、“[St.3214]自動運転休止中(D54441.2+4s)”がONします。

[Rq.3379]シングルブロック(D54226.3+2s)

- 自動運転のGコードプログラムを1ブロックずつ実行させるための指令です。
- 自動運転を実行中の場合、実行中のブロックを終了した後、停止します。次のブロックの指令を実行するには、再度 “[Rq.3377]自動運転起動(サイクルスタート)(D54226.1+2s)”をOFF→ONします。
- 1ブロック動作した後に停止するため、Gコードプログラムを1ブロックずつ実行することができます。

[Rq.3380]リセット指令(D54226.4+2s)

Gコード制御のリセットを行うための指令です。“[Rq.3380]リセット指令(D54226.4+2s)”をOFF→ONすると下記のように動作します。

- “[St.3216]リセット中(D54441.8+4s)”がONし、移動中の制御軸は減速停止します。
- 停止後、リセットが完了すると “[St.3216]リセット中(D54441.8+4s)”がOFFし、“[St.3217]リセット完了(D54441.9+4s)”がONします。
- モーダル状態を初期化します。
- 工具補正データをキャンセルします。(軸動作はしません。)
- 下記エラーのGコード制御用デバイスをクリアします。ただし、LED表示やGX Works3, MT Developer2のエラー表示、特殊リレー、特殊レジスタのエラー状態は解除されません。解除の方法は下記を参照してください。

📖 MELSEC iQ-R モーションコントローラプログラミングマニュアル(共通編)

- [St.3209]Gコード制御エラー検出(D54440.1+4s)
- [Md.3019]Gコード制御エラーコード(D54504+128s)
- [Md.3020]Gコード制御エラー詳細コード1(D54505+128s)
- [Md.3021]Gコード制御エラー詳細コード2(D54506+128s)
- Mコード出力は保持します。(ストローブ信号はOFFとなります。)
- M単独出力(M00, M01, M02, M30)は、OFFとなります。

[Rq.3381]プログラム運転モード(メモリモード)(D54226.8+2s)

- 自動運転のメモリモードを行うための指令です。
- “[Rq.3381]プログラム運転モード(メモリモード)(D54226.8+2s)”をONすると、自動運転のメモリモードが選択され、指定したGコードプログラムの自動運転を行います。(Gコードプログラムの起動は、“[Rq.3377]自動運転起動(サイクルスタート)(D54226.1+2s)”で行います。)
- “[Rq.3381]プログラム運転モード(メモリモード)(D54226.8+2s)”をOFFすると、メモリモードを解除します。ただし、プログラム起動中にOFFすると、軽度エラー(エラーコード: 1FC2H(詳細コード: 0222H))となり、ブロック停止します。
- “[Rq.3376]Gコード制御要求(D54226.0+2s)”をOFFする場合は、“[Rq.3381]プログラム運転モード(メモリモード)(D54226.8+2s)”をOFFし、メモリモードを解除してから行ってください。

[Rq.3384]マクロシングル(D54226.C+2s)

- Gコード制御中に “[Rq.3384]マクロシングル(D54226.C+2s)”をON、かつ “[Rq.3379]シングルブロック(D54226.3+2s)”がONのときに、演算指令、制御指令のGコードプログラムを1ブロックずつ実行させるための指令です。
- “[Rq.3384]マクロシングル(D54226.C+2s)”は、リセット状態^{*1}で “[Rq.3377]自動運転起動(サイクルスタート)(D54226.1+2s)”をOFF→ONしたときに有効となります。Gコード制御中でない場合、またはリセット状態以外の場合は、“[Rq.3384]マクロシングル(D54226.C+2s)”の取込みを無視します。
- マクロシングルを無効にするときは、自動運転中に “[Rq.3380]リセット指令(D54226.4+2s)”をONし、Gコード制御をリセットしてください。再度マクロシングルを使用するときは、リセットした後に “[Rq.3384]マクロシングル(D54226.C+2s)”をOFF→ONしてください。

^{*1} リセット状態は、下記のデバイスがすべてOFF状態のとき。

- [St.3212]自動運転中(D54441.0+4s)
- [St.3213]自動運転起動中(D54441.1+4s)
- [St.3214]自動運転休止中(D54441.2+4s)

[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)

補助機能(M機能)が指令され、モーションSFCプログラムやシーケンスプログラムにて、処理および完了シーケンスを作成することで、完了信号を待つ次のブロックに進ませる指令です。“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”の立下り(ON→OFF)時、次のブロックに進みます。詳細は、補助機能完了を参照してください。(📖 204ページ 補助機能完了)

[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)

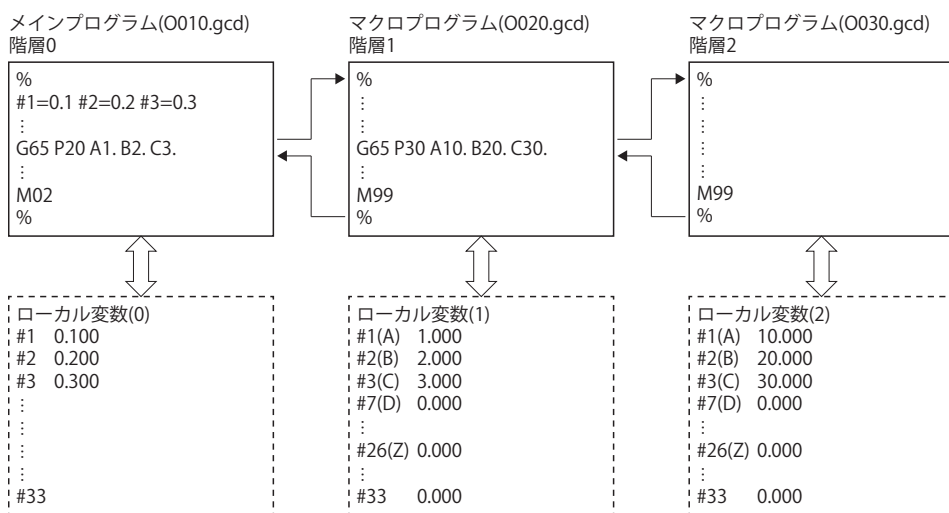
補助機能(M機能)が指令され、モーションSFCプログラムやシーケンスプログラムにて、処理および完了シーケンスを作成することで、完了信号を待つ次のブロックに進ませる指令です。“[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”の立上り(OFF→ON)時、次のブロックに進みます。詳細は、補助機能完了を参照してください。(📖 204ページ 補助機能完了)

[Rq.3385]G65引数初期化(D54227.4+2s)

- “[Rq.3385]G65引数初期化(D54227.4+2s)”を操作し、マクロ呼出し指定引数以外を初期化(0クリア)するための指令です。
- 初期化(0クリア)の対象は、マクロ呼出しを行う階層の引数です。
- “[Rq.3385]G65引数初期化(D54227.4+2s)”は、リセット状態^{*1}で “[Rq.3377]自動運転起動(サイクルスタート)(D54226.1+2s)”をOFF→ONしたときに有効となります。
- 指定できる範囲を下記に示します。

設定値	内容
ON	マクロ呼出し時、指定しない引数を保持します。
OFF	マクロ呼出し時、指定しない引数を初期化(0クリア)します。

- “[Rq.3385]G65引数初期化(D54227.4+2s)”がOFFの状態でもマクロ呼出しを行った場合、指定していない引数は初期化(0クリア)されます。



*1 リセット状態は、下記のデバイスがすべてOFF状態のとき。

- [St.3212]自動運転中(D54441.0+4s)
- [St.3213]自動運転起動中(D54441.1+4s)
- [St.3214]自動運転休止中(D54441.2+4s)

3.6 Gコード制御系統制御デバイス

デバイス番号		信号名称
R標準配置方式	Q互換配置方式	
D54278～D54293		系統1のGコード制御系統制御デバイス
D54294～D54309		系統2のGコード制御系統制御デバイス

・各系統の詳細内容

デバイス番号		記号	信号名称	リフレッシュ 周期	取込周期	信号種別
R標準配置方式	Q互換配置方式					
D54278+16s		Cd.3320	プログラム番号指定レジスタ	—	Gコード制御演 算周期	指令信号
D54279+16s		—	ユーザ使用不可	—	—	—
D54280+16s		Cd.3321	シーケンス番号指定レジスタ	—	Gコード制御演 算周期	指令信号
D54281+16s						
D54282+16s		Cd.3322	ブロック番号指定レジスタ	—	—	—
D54283+16s						
D54284+16s		—	ユーザ使用不可	—	—	—
D54285+16s						
D54286+16s						
D54287+16s						
D54288+16s						
D54289+16s						
D54290+16s						
D54291+16s						
D54292+16s						
D54293+16s						

[Cd.3320]プログラム番号指定レジスタ(D54278+16s)

指定するプログラム番号のGコードプログラムを起動するために設定するレジスタです。

詳細は、自動運転起動(サイクルスタート)を参照してください。(P.238ページ 自動運転起動(サイクルスタート))

項目	設定範囲
プログラム番号	1～256

[Cd.3321]シーケンス番号指定レジスタ(D54280+16s, D54281+16s)

指定するシーケンス番号のブロックからGコードプログラムを起動するために設定するレジスタです。

詳細は、自動運転起動(サイクルスタート)を参照してください。(P.238ページ 自動運転起動(サイクルスタート))

項目	設定範囲
シーケンス番号	0～99999

[Cd.3322]ブロック番号指定レジスタ(D54282+16s, D54283+16s)

指定するブロック番号のブロックからGコードプログラムを起動するために設定するレジスタです。

詳細は、自動運転起動(サイクルスタート)を参照してください。(P.238ページ 自動運転起動(サイクルスタート))

項目	設定範囲
ブロック番号	0～99999

3.7 Gコード制御システムステータス

デバイス番号		信号名称
R標準配置方式	Q互換配置方式	
D54440～D54443		系統1のGコード制御システムステータス
D54444～D54447		系統2のGコード制御システムステータス

・各系統の詳細内容

デバイス番号		記号	信号名称	リフレッシュ 周期	取込周期	信号種別
R標準配置方式	Q互換配置方式					
D54440.0+4s		St.3208	Gコード制御中	メイン周期	—	ステータス信号
D54440.1+4s		St.3209	Gコード制御エラー検出	即時		
D54440.2+4s		St.3210	全軸スムージングゼロ	Gコード制御演算周期		
D54440.3+4s		—	ユーザ使用不可	—	—	—
D54440.4+4s						
D54440.5+4s						
D54440.6+4s						
D54440.7+4s						
D54440.8+4s		St.3211	メモリモード中	Gコード制御演算周期	—	ステータス信号
D54440.9+4s		—	ユーザ使用不可	—	—	—
D54440.A+4s						
D54440.B+4s						
D54440.C+4s						
D54440.D+4s						
D54440.E+4s						
D54440.F+4s						
D54441.0+4s		St.3212	自動運転中	Gコード制御演算周期	—	ステータス信号
D54441.1+4s		St.3213	自動運転起動中			
D54441.2+4s		St.3214	自動運転休止中			
D54441.3+4s		St.3215	Gコード制御終了中			
D54441.4+4s		—	ユーザ使用不可	—	—	—
D54441.5+4s						
D54441.6+4s						
D54441.7+4s						
D54441.8+4s		St.3216	リセット中	Gコード制御演算周期	—	ステータス信号
D54441.9+4s		St.3217	リセット完了			
D54441.A+4s		—	ユーザ使用不可	—	—	—
D54441.B+4s						
D54441.C+4s						
D54441.D+4s						
D54441.E+4s						
D54441.F+4s		St.3234	マクロシングル有効中	Gコード制御演算周期	—	ステータス信号
D54442.0+4s		St.3218	M単独出力 M00			
D54442.1+4s		St.3219	M単独出力 M01			
D54442.2+4s		St.3220	M単独出力 M02			
D54442.3+4s		St.3221	M単独出力 M30			
D54442.4+4s		St.3222	補助機能ストローブ1			
D54442.5+4s		St.3223	補助機能ストローブ2			
D54442.6+4s		St.3224	補助機能ストローブ3			
D54442.7+4s		St.3225	補助機能ストローブ4			

デバイス番号		記号	信号名称	リフレッシュ 周期	取込周期	信号種別
R標準配置方式	Q互換配置方式					
D54442.8+4s		—	ユーザ使用不可	—	—	—
D54442.9+4s						
D54442.A+4s						
D54442.B+4s						
D54442.C+4s						
D54442.D+4s						
D54442.E+4s						
D54442.F+4s						
D54443.0+4s						
D54443.1+4s						
D54443.2+4s						
D54443.3+4s						
D54443.4+4s						
D54443.5+4s						
D54443.6+4s						
D54443.7+4s						
D54443.8+4s						
D54443.9+4s						
D54443.A+4s						
D54443.B+4s						
D54443.C+4s						
D54443.D+4s						
D54443.E+4s						
D54443.F+4s						

[St.3208]Gコード制御中(D54440.0+4s)

Gコード制御中にONする信号です。

[St.3209]Gコード制御エラー検出(D54440.1+4s)

- Gコード制御に関するエラーの検出でONする信号で、エラーの有／無の判別に使用できます。Gコード制御に関するエラー検出で該当するエラーコードが、“[Md.3019]Gコード制御エラーコード(D54504+128s)”， “[Md.3020]Gコード制御エラー詳細コード1(D54505+128s)”， “[St.3021]Gコード制御エラー詳細コード2(D54506+128s)”に格納されます。エラー検出時のエラーコードは、下記を参照してください。

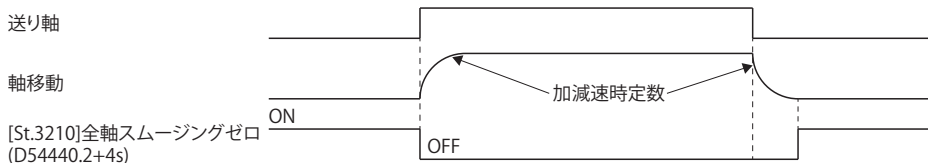
📖 MELSEC iQ-R モーションコントローラプログラミングマニュアル(共通編)

- “[Rq.3380]リセット指令(D54226.4+2s)”をONするとOFFします。

[St.3210]全軸スージングゼロ(D54440.2+4s)

制御軸の遅れ(加減速時定数による遅れ)が全軸ゼロのときにONする信号です。

自動運転で指令した移動量を、加減速時定数の遅れ量も含め、すべての出力処理が完了するとONします。移動指令を実行中、および加減速時定数の遅れ量があるとOFFします。



Point

非常に低速で移動時は、移動中であっても “[St.3210]全軸スージングゼロ(D54440.2+4s)” がONすることがあります。

[St.3211]メモリモード中(D54440.8+4s)

メモリモード中のときにONする信号です。

[St.3212]自動運転中(D54441.0+4s)

自動運転中ONする信号で、“[Rq.3377]自動運転起動(サイクルスタート)(D54226.1+2s)”のONにより、自動運転起動してからリセット状態になるまでONします。

自動運転の状態は，“[St.3212]自動運転中(D54441.0+4s)”， “[St.3213]自動運転起動中(D54441.1+4s)”， “[St.3214]自動運転休止中(D54441.2+4s)”のON/OFFにより下記となります。

状態	[St.3212]自動運転中(D54441.0+4s)	[St.3213]自動運転起動中(D54441.1+4s)	[St.3214]自動運転休止中(D54441.2+4s)
リセット	OFF	OFF	OFF
自動運転停止	ON	OFF	OFF
自動運転休止	ON	OFF	ON
自動運転起動	ON	ON	OFF

各状態について下記に説明します。

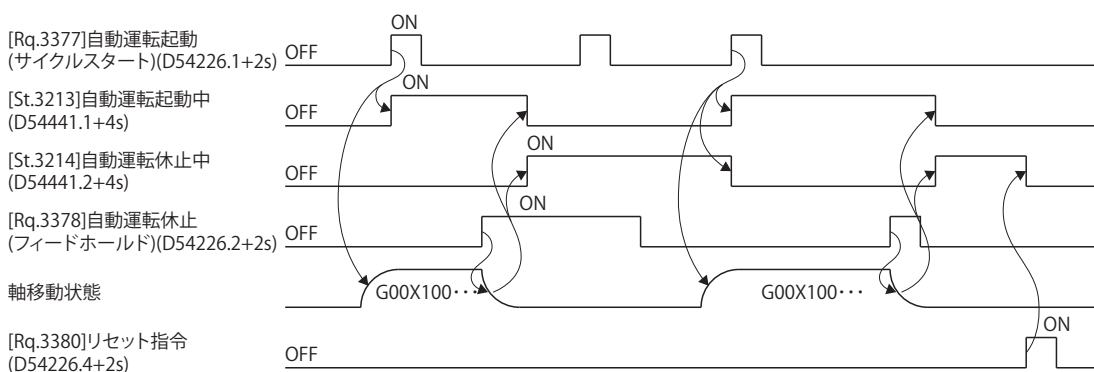
- リセット状態
リセット後の状態も含み、自動運転を停止している状態です。自動運転を行っていない状態はすべてリセット状態となります。
- 自動運転停止状態
1つのブロックの実行を終了し、自動運転を停止している状態です。“[Rq.3379]シングルブロック(D54226.3+2s)”のONによる停止時は自動運転停止状態になります。
- 自動運転休止状態
1つのブロックの実行途中で自動運転を停止している状態です。“[Rq.3378]自動運転休止(フィードホールド)(D54226.2+2s)”のONにより、自動運転休止状態となります。
- 自動運転起動状態
実際に自動運転を実行している状態です。

[St.3213]自動運転起動中(D54441.1+4s)

自動運転起動中ONする信号で、“[Rq.3377]自動運転起動(サイクルスタート)(D54226.1+2s)”のONにより、自動運転起動してから自動運転休止、ブロック停止、リセット状態になるまでONします。移動指令やM処理を実行中であることを確認できます。

[St.3214]自動運転休止中(D54441.2+4s)

- 自動運転が休止した場合にONする信号で、移動指令、または補助機能指令実行中に、“[Rq.3378]自動運転休止(フィードホールド)(D54226.2+2s)”のONなどの要因で休止中であることを確認できます。
- “[Rq.3378]自動運転休止(フィードホールド)(D54226.2+2s)”がONしたときにONします。また、補助機能(M機能)指令中でもONします。
- リセットしたとき、または“[Rq.3377]自動運転起動(サイクルスタート)(D54226.1+2s)”をONしたときにOFFします。ただし、“[Rq.3378]自動運転休止(フィードホールド)(D54226.2+2s)”をOFFに戻していないときや、メモリモードではないときは無効となります。



[St.3215]Gコード制御終了中(D54441.3+4s)

Gコード制御の停止要因が発生した場合にONする信号です。停止処理後に“[St.3215]Gコード制御終了中(D54441.3+4s)”はOFFします。停止要因については、Gコード制御システムの停止動作を参照してください。(25ページ Gコード制御システムの停止動作)

[St.3216]リセット中(D54441.8+4s)

“[Rq.3380]リセット指令(D54226.4+2s)”によるリセット処理中にONします。
リセットがすぐに完了する場合、リセット中の状態を検出できないことがあります。

[St.3217]リセット完了(D54441.9+4s)

- “[Rq.3380]リセット指令(D54226.4+2s)”にてリセットが完了するとONする信号で、リセットが完了したことを確認できます。
- “[St.3217]リセット完了(D54441.9+4s)”は自動でOFFしません。再度リセットしたときにリセットの完了を確認する場合は、“[St.3217]リセット完了(D54441.9+4s)”をOFFしてから、“[Rq.3380]リセット指令(D54226.4+2s)”をONしてください。

[St.3234]マクロシングル有効中(D54441.F+4s)

- マクロシングルが使用可能のときONする信号です。
- “[St.3234]マクロシングル有効中(D54441.F+4s)”がONの状態では、“[Rq.3379]シングルブロック(D54226.3+2s)”をONすると、マクロシングルが有効状態となります。
- “[St.3234]マクロシングル有効中(D54441.F+4s)”は下記の場合にON/OFFします。

状態	内容
ON	<ul style="list-style-type: none"> • リセット状態^{*1}で“[Rq.3384]マクロシングル(D54226.C+2s)”をONした後、“[Rq.3377]自動運転起動(サイクルスタート)(D54226.1+2s)”をOFF→ONしたとき。
OFF	<ul style="list-style-type: none"> • 自動運転中に“[Rq.3380]リセット指令(D54226.4+2s)”をONしたことにより、Gコード制御のリセットが完了したとき。 • Gコード制御の停止要因^{*2}が発生したとき。

*1 リセット状態は、下記のデバイスがすべてOFF状態のとき。

- [St.3212]自動運転中(D54441.0+4s)
- [St.3213]自動運転起動中(D54441.1+4s)
- [St.3214]自動運転休止中(D54441.2+4s)

*2 Gコード制御の停止要因については、Gコード制御システムの停止動作を参照してください。(25ページ Gコード制御システムの停止動作)

[St.3218]M単独出力 M00(D54442.0+4s)

- 特定の補助機能M00が出力されたときにONする信号です。なお特定の補助機能M00でも、通常の“[St.3222]補助機能ストローブ1(D54442.4+4s)”～“[St.3225]補助機能ストローブ4(D54442.7+4s)”, “[Md.3058]Mコードデータ1(D54554+128s, D54555+128s)”～“[Md.3061]Mコードデータ4(D54560+128s, D54561+128s)”は出力されます。
- 自動運転による運転中によりM00が指令されるとONし、“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”, “[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”, または“[Rq.3380]リセット指令(D54226.4+2s)”によりOFFします。詳細は、M単独出力を参照してください。(202ページ M単独出力)

[St.3219]M単独出力 M01(D54442.1+4s)

- 特定の補助機能M01が出力されたときにONする信号です。なお特定の補助機能M01でも、通常の“[St.3222]補助機能ストローブ1(D54442.4+4s)”～“[St.3225]補助機能ストローブ4(D54442.7+4s)”, “[Md.3058]Mコードデータ1(D54554+128s, D54555+128s)”～“[Md.3061]Mコードデータ4(D54560+128s, D54561+128s)”は出力されます。
- 自動運転による運転中によりM01が指令されるとONし, “[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”, “[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”, または“[Rq.3380]リセット指令(D54226.4+2s)”によりOFFします。詳細は、M単独出力を参照してください。(202ページ M単独出力)

[St.3220]M単独出力 M02(D54442.2+4s)

- 特定の補助機能M02が出力されたときにONする信号です。なお特定の補助機能M02でも、通常の“[St.3222]補助機能ストローブ1(D54442.4+4s)”～“[St.3225]補助機能ストローブ4(D54442.7+4s)”, “[Md.3058]Mコードデータ1(D54554+128s, D54555+128s)”～“[Md.3061]Mコードデータ4(D54560+128s, D54561+128s)”は出力されます。
- 自動運転による運転中によりM02が指令されるとONし, “[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”, “[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”, または“[Rq.3380]リセット指令(D54226.4+2s)”によりOFFします。詳細は、M単独出力を参照してください。(202ページ M単独出力)

[St.3221]M単独出力 M30(D54442.3+4s)

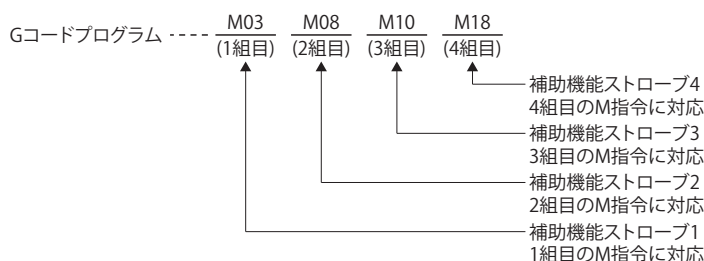
- 特定の補助機能M30が出力されたときにONする信号です。なお特定の補助機能M30でも、通常の“[St.3222]補助機能ストローブ1(D54442.4+4s)”～“[St.3225]補助機能ストローブ4(D54442.7+4s)”, “[Md.3058]Mコードデータ1(D54554+128s, D54555+128s)”～“[Md.3061]Mコードデータ4(D54560+128s, D54561+128s)”は出力されます。
- 自動運転による運転中によりM30が指令されるとONし, “[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”, “[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”, または“[Rq.3380]リセット指令(D54226.4+2s)”によりOFFします。詳細は、M単独出力を参照してください。(202ページ M単独出力)

[St.3222]補助機能ストローブ1(D54442.4+4s)

- 1組目の補助機能(M機能)が指令されたときにONする信号です。
- 自動運転のGコードプログラムにより1組目の補助機能(M機能)が指令されるとONします。
- 下記の場合にOFFします。
 - “[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”, または “[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)” がONしたとき。
 - リセット状態となったとき。

■動作

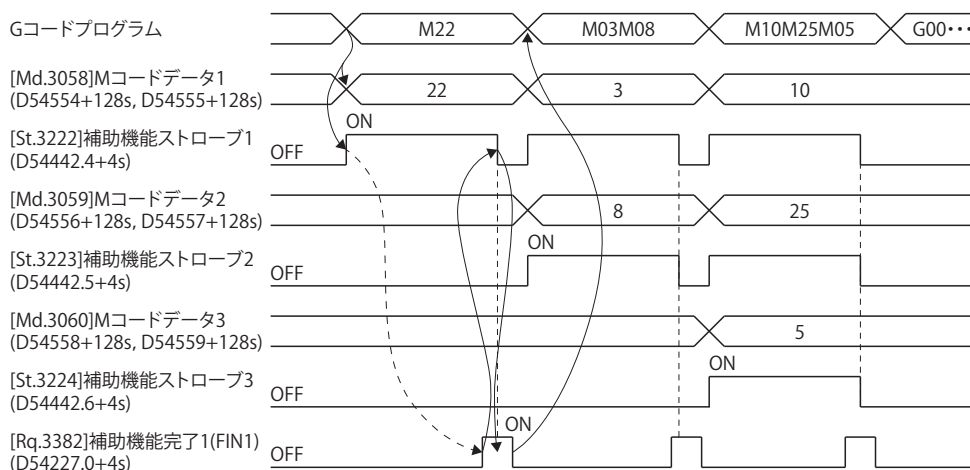
- 補助機能(M機能)は1ブロックで同時に4個まで指令できます。



*1 下記のMコードデータは、補助機能ストローブを出力しません。

- M98(サブプログラム呼出し)
- M99(サブ/マクロプログラム復帰)

- “[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”, または “[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)” がONしている間は、M機能が実行されても “[St.3222]補助機能ストローブ1(D54442.4+4s)” ~ “[St.3225]補助機能ストローブ4(D54442.7+4s)” はONしません。



- モーションSFCプログラム、シーケンスプログラムは、以下の処理となります。

- 補助機能(M機能)を指令すると、“[St.3222]補助機能ストローブ1(D54442.4+4s)” ~ “[St.3225]補助機能ストローブ4(D54442.7+4s)” と “[Md.3058]Mコードデータ1(D54554+128s, D54555+128s)” ~ “[Md.3061]Mコードデータ4(D54560+128s, D54561+128s)” が出力します。
- モーションSFCプログラム、またはシーケンスプログラム処理において、必ず “[St.3222]補助機能ストローブ1(D54442.4+4s)” ~ “[St.3225]補助機能ストローブ4(D54442.7+4s)” をトリガとして補助機能(M機能)の処理を開始します。
- 補助機能(M機能)の処理が完了したら、“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”, または “[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)” をONします。
- モーションCPUは、“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”, または “[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)” の立上りを見て、“[St.3222]補助機能ストローブ1(D54442.4+4s)” ~ “[St.3225]補助機能ストローブ4(D54442.7+4s)” をOFFします。
- モーションSFCプログラム、またはシーケンスプログラム処理においては、“[St.3222]補助機能ストローブ1(D54442.4+4s)” ~ “[St.3225]補助機能ストローブ4(D54442.7+4s)” のOFFを確認して、“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”, または “[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)” をOFFします。

[St.3223]補助機能ストローク2(D54442.5+4s)

- 2組目の補助機能(M機能)が指令されたときにONする信号です。
- 自動運転により、同一ブロックに補助機能(M機能)が2つ以上指令されるとONします。
- 詳細動作は、“[St.3222]補助機能ストローク1(D54442.4+4s)”と同様です。(43ページ [St.3222]補助機能ストローク1(D54442.4+4s))

[St.3224]補助機能ストローク3(D54442.6+4s)

- 3組目の補助機能(M機能)が指令されたときにONする信号です。
- 自動運転により、同一ブロックに補助機能(M機能)が3つ以上指令されるとONします。
- 詳細動作は、“[St.3222]補助機能ストローク1(D54442.4+4s)”と同様です。(43ページ [St.3222]補助機能ストローク1(D54442.4+4s))

[St.3225]補助機能ストローク4(D54442.7+4s)

- 4組目の補助機能(M機能)が指令されたときにONする信号です。
- 自動運転により、同一ブロックに補助機能(M機能)が4つ以上指令されるとONします。
- 詳細動作は、“[St.3222]補助機能ストローク1(D54442.4+4s)”と同様です。(43ページ [St.3222]補助機能ストローク1(D54442.4+4s))

3.8 Gコード制御システムモニタデバイス

デバイス番号		信号名称
R標準配置方式	Q互換配置方式	
D54496～D54623		系統1のGコード制御システムモニタデバイス
D54624～D54751		系統2のGコード制御システムモニタデバイス

・各系統の詳細内容

デバイス番号		記号	信号名称	リフレッシュ 周期	取込周期	信号種別
R標準配置方式	Q互換配置方式					
D54496+128s		Md.3016	系統内軸数	STOP→RUN時	—	モニタデバイス
D54497+128s		—	ユーザ使用不可	—	—	—
D54498+128s		Md.3017	Gコード制御軸構成	STOP→RUN時	—	モニタデバイス
D54499+128s						
D54500+128s						
D54501+128s						
D54502+128s						
D54503+128s		Md.3018	速度	Gコード制御演 算周期	—	
D54504+128s						
D54505+128s		Md.3019	Gコード制御エラーコード	即時		
D54506+128s		Md.3020	Gコード制御エラー詳細コード1			
D54507+128s		Md.3021	Gコード制御エラー詳細コード2			
D54508+128s		—	ユーザ使用不可	—	—	—
D54509+128s		Md.3022	実行中プログラム番号(メイン)	Gコードプログラ ム始動時	—	モニタデバイス
D54510+128s		—	ユーザ使用不可	—	—	—
D54511+128s		Md.3023	実行中シーケンス番号(メイン)	Gコード制御演 算周期	—	モニタデバイス
D54512+128s						
D54513+128s		Md.3024	実行中ブロック番号(メイン)			
D54514+128s						
D54515+128s		Md.3025	実行中プログラム番号(サブ/マクロ)			
D54516+128s		—	ユーザ使用不可	—	—	—
D54517+128s		Md.3026	実行中シーケンス番号(サブ/マクロ)	Gコード制御演 算周期	—	モニタデバイス
D54518+128s						
D54519+128s		Md.3027	実行中ブロック番号(サブ/マクロ)			
D54520+128s						
D54521+128s		Md.3028	グループ01モータ状態			
D54522+128s		Md.3029	グループ02モータ状態			
D54523+128s		Md.3030	グループ03モータ状態			
D54524+128s		—	ユーザ使用不可	—	—	—
D54525+128s						
D54526+128s						
D54527+128s		Md.3034	グループ07モータ状態	Gコード制御演 算周期	—	モニタデバイス
D54528+128s		Md.3035	工具径補正番号			
D54529+128s		Md.3036	工具径補正量			
D54530+128s		—	ユーザ使用不可	—	—	—
D54531+128s						
D54532+128s		Md.3038	グループ08モータ状態	Gコード制御演 算周期	—	モニタデバイス
D54533+128s		Md.3039	工具長補正番号			
D54534+128s		Md.3040	工具長補正量			
D54535+128s		—	ユーザ使用不可	—	—	—
D54536+128s						
D54537+128s						

デバイス番号		記号	信号名称	リフレッシュ 周期	取込周期	信号種別
R標準配置方式	Q互換配置方式					
D54538+128s		Md.3042	工具長補正軸No.	Gコード制御演 算周期	—	モニタデバイス
D54539+128s		—	ユーザ使用不可	—	—	—
D54540+128s						
D54541+128s						
D54542+128s		Md.3046	グループ12モーダル状態	Gコード制御演 算周期	—	モニタデバイス
D54543+128s		Md.3047	グループ13モーダル状態			
D54544+128s		—	ユーザ使用不可	—	—	—
D54545+128s		Md.3049	グループ15モーダル状態	Gコード制御演 算周期	—	モニタデバイス
D54546+128s		Md.3050	グループ16モーダル状態			
D54547+128s		—	ユーザ使用不可	—	—	—
D54548+128s						
D54549+128s						
D54550+128s						
D54551+128s		Md.3055	グループ21モーダル状態	Gコード制御演 算周期	—	モニタデバイス
D54552+128s		—	ユーザ使用不可	—	—	—
D54553+128s						
D54554+128s		Md.3058	Mコードデータ1	Gコード制御演 算周期	—	モニタデバイス
D54555+128s						
D54556+128s		Md.3059	Mコードデータ2			
D54557+128s						
D54558+128s		Md.3060	Mコードデータ3			
D54559+128s						
D54560+128s		Md.3061	Mコードデータ4			
D54561+128s						
D54562+128s		—	ユーザ使用不可	—	—	—
D54563+128s						
D54564+128s						
D54565+128s						
D54566+128s						
D54567+128s						
D54568+128s						
D54569+128s						
D54570+128s						
D54571+128s						
D54572+128s						
D54573+128s						
D54574+128s						
D54575+128s						
D54576+128s						
D54577+128s						
D54578+128s						
D54579+128s						
D54580+128s						
D54581+128s						
D54582+128s		Md.3074	ローカル変数深さ	Gコード制御演 算周期	—	モニタデバイス
D54583+128s		—	ユーザ使用不可	—	—	—
D54584+128s						
D54585+128s						
D54586+128s						
D54587+128s						

デバイス番号		記号	信号名称	リフレッシュ 周期	取込周期	信号種別
R標準配置方式	Q互換配置方式					
D54588+128s		Md.3070	実行中プログラムコメント	Gコードプログラ ム始動時	—	モニタデバイス
D54589+128s						
D54590+128s						
D54591+128s						
D54592+128s						
D54593+128s						
D54594+128s						
D54595+128s						
D54596+128s		—	ユーザ使用不可	—	—	—
D54597+128s						
D54598+128s						
D54599+128s						
D54600+128s						
D54601+128s						
D54602+128s						
D54603+128s						
D54604+128s						
D54605+128s						
D54606+128s						
D54607+128s						
D54608+128s						
D54609+128s						
D54610+128s						
D54611+128s						
D54612+128s						
D54613+128s						
D54614+128s						
D54615+128s						
D54616+128s						
D54617+128s						
D54618+128s						
D54619+128s						
D54620+128s						
D54621+128s						
D54622+128s						
D54623+128s						

[Md.3016]系統内軸数(D54496+128s)

[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“系統基本設定”⇒“系統内軸数”にて設定した軸数が格納されます。

[Md.3017]Gコード制御軸構成(D54498+128s～D54501+128s)

[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“系統軸情報”⇒“軸No.”にて設定した、軸No.がビットデータで格納されます。系統内の軸がサーボネットワーク設定で割付けたどの軸No.を設定しているかを確認できます。

[Md.3017]Gコード制御軸構成		b15 b14 b13 b12 b11 b10 b9 b8 b7 b6 b5 b4 b3 b2 b1 b0																
		D54498+128s	軸16	軸15	軸14	軸13	軸12	軸11	軸10	軸9	軸8	軸7	軸6	軸5	軸4	軸3	軸2	軸1
		D54499+128s	軸32	軸31	軸30	軸29	軸28	軸27	軸26	軸25	軸24	軸23	軸22	軸21	軸20	軸19	軸18	軸17
		D54500+128s	軸48	軸47	軸46	軸45	軸44	軸43	軸42	軸41	軸40	軸39	軸38	軸37	軸36	軸35	軸34	軸33
		D54501+128s	軸64	軸63	軸62	軸61	軸60	軸59	軸58	軸57	軸56	軸55	軸54	軸53	軸52	軸51	軸50	軸49

*1: Gコード制御構成軸は0/1を格納する。

0: 非構成軸

1: 構成軸

*2: R16MTCPUは軸1～16, R32MTCPUは軸1～32の範囲が有効です。

[Md.3018]速度(D54502+128s, D54503+128s)

補間動作中、または非補間動作中の速度($\times 0.01$ [mm/min])が格納されます。

項目	内容
補間動作中	現在移動中のベクトル方向の速度を格納します。
非補間動作中	移動中の軸において、最も速度が大きい軸の速度を格納します。

[Md.3019]Gコード制御エラーコード(D54504+128s)

- Gコード制御に関するエラー発生時に、該当するエラーコードが格納されます。エラーコード格納後に、他のエラーが発生した場合は、上書きされ新しいエラーコードになります。
- Gコード制御エラーコードのクリアは、“[Rq.3380]リセット指令(D54226.4+2s)”, または“エラー解除(SM50)”で行います。

Point

エラーコードの詳細は、下記を参照してください。

📖 MELSEC iQ-R モーションコントローラプログラミングマニュアル(共通編)

[Md.3020]Gコード制御エラー詳細コード1(D54505+128s)

- Gコード制御に関するエラー発生時に、該当するエラー詳細コード1が格納されます。エラーコード格納後に、他のエラーが発生した場合は、上書きされ新しいエラーコードになります。
- Gコード制御エラーコードのクリアは、“[Rq.3380]リセット指令(D54226.4+2s)”, または“エラー解除(SM50)”で行います。

Point

エラー詳細コードの詳細は、Gコード制御エラー詳細コードを参照してください。(📖 270ページ Gコード制御エラー詳細コード)

[Md.3021]Gコード制御エラー詳細コード2(D54506+128s)

- Gコード制御に関するエラー発生時に、該当するエラー詳細コード2が格納されます。エラーコード格納後に、他のエラーが発生した場合は、上書きされ新しいエラーコードになります。
- Gコード制御エラーコードのクリアは、“[Rq.3380]リセット指令(D54226.4+2s)”, または“エラー解除(SM50)”で行います。

Point

エラー詳細コードの詳細は、Gコード制御エラー詳細コードを参照してください。(📖 270ページ Gコード制御エラー詳細コード)

[Md.3022]実行中プログラム番号(メイン)(D54508+128s)

Gコードプログラム運転実行時に、該当するプログラム番号が格納されます。

[Md.3023]実行中シーケンス番号(メイン)(D54510+128s, D54511+128s)

Gコードプログラム運転中に、実行しているシーケンス番号が格納されます。

[Md.3024]実行中ブロック番号(メイン)(D54512+128s, D54513+128s)

Gコードプログラム運転中に、実行しているブロック番号が格納されます。

シーケンス番号を変更した場合、ブロック番号は「0」から開始します。

1ブロック実行ごとに+1します。

■動作例

- GコードプログラムNo.12を実行した場合

プログラム	[Md.3022]実行中プログラム番号(メイン)(D54508+128s)	[Md.3023]実行中シーケンス番号(メイン)(D54510+128s, D54511+128s)	[Md.3024]実行中ブロック番号(メイン)(D54512+128s, D54513+128s)
G90 G00 X0. Y0.	12	0	1
G91 X100. Y100.	12	0	2
N100 X-50. Y-25.	12	100	0
N110 G01 X250. F300.	12	110	0
Y-225.	12	110	1
X-50.	12	110	2
Y-25.	12	110	3
N120 Y-125.	12	120	0
N130 G00 X-100.	12	130	0
N140 G01 X-200.	12	140	0
Y-175.	12	140	1
X-100.	12	140	2
Y-75.	12	140	3
N150 G90 G00 X0. Y0.	12	150	0
N160 M02	12	160	0

- 分岐(IF, THEN, ELSE, ENDIF)命令を実行した場合

条件分岐により実行されないブロックは、ブロック番号としてカウントしません。

プログラム	[Md.3023]実行中シーケンス番号(メイン)(D54510+128s, D54511+128s)	[Md.3024]実行中ブロック番号(メイン)(D54512+128s, D54513+128s)	
		#100=0, #101=0	#100≠0, #101≠0
G90 G00 X0. Y0.	0	1	1
G91 G01 X100. Y100. F2000.	0	2	2
X200. Y200.	0	3	3
IF [#100 EQ 0] THEN	0	4	4
G01 Y300. F1500.	0	5	—
X300.	0	6	—
ENDIF	0	7	5
G02 X50. Y50. I0. J50. F800.	0	8	6
G01 X100. Y500. F2000.	0	9	7
IF [#101 EQ 0] THEN	0	10	8
G00 X10. Y100.	0	11	—
ELSE	0	12	9
G90 G00 X0 Y0.	0	—	10
ENDIF	0	13	11
M02	0	14	12

- ・繰り返し(WHILE, DO, END)命令を実行した場合

条件式が成立している間に実行されたブロック数分(WHILE～DOブロックからENDブロックまで)、ブロック番号がカウントアップします。

プログラム	[Md.3023]実行中シーケンス番号(メイン)(D54510+128s, D54511+128s)	[Md.3024]実行中ブロック番号(メイン)(D54512+128s, D54513+128s)			
		#100=0	#100=1	#100=2	#100=3
G90 G00 X0. Y0.	0	1			
#100=0	0	2			
G91 G00 X25. Y50.	0	3			
WHILE [#100 LT 3] DO1	0	4	8	12	16
G03 X0. Y0. I50. J0. F100.	0	5	9	13	—
#100=#100+1	0	6	10	14	—
END1	0	7	11	15	—
G90 G00 X0 Y0.	0	17			
M02	0	18			

[Md.3025]実行中プログラム番号(サブ/マクロ)(D54514+128s)

Gコードプログラム運転中に、実行しているサブプログラム、またはマクロプログラムのプログラム番号が格納されます。サブプログラム、またはマクロプログラムが実行中でない場合は、「0」が格納されます。

[Md.3026]実行中シーケンス番号(サブ/マクロ)(D54516+128s, D54517+128s)

Gコードプログラム運転中に、実行しているサブプログラム、またはマクロプログラムのシーケンス番号が格納されます。サブプログラム、またはマクロプログラムが実行中でない場合は、「0」が格納されます。

[Md.3027]実行中ブロック番号(サブ/マクロ)(D54518+128s, D54519+128s)

Gコードプログラム運転中に、実行しているサブプログラム、またはマクロプログラムのブロック番号が格納されます。サブプログラム、またはマクロプログラムが実行中でない場合は、「0」が格納されます。また、シーケンス番号を変更した場合、ブロック番号は「0」から開始します。

1ブロック実行ごとに+1します。

[Md.3028]グループ01モーダル状態(D54520+128s)

グループ01のモーダル状態が格納されます。

Gコードモーダル状態	格納値
G00 位置決め	0
G01 直線補間	10
G02 円弧補間CW	20
G03 円弧補間CCW	30

[Md.3029]グループ02モーダル状態(D54521+128s)

グループ02のモーダル状態が格納されます。

Gコードモーダル状態	格納値
G17 平面選択 X-Y	170
G18 平面選択 Z-X	180
G19 平面選択 Y-Z	190

[Md.3030]グループ03モーダル状態(D54522+128s)

グループ03のモーダル状態が格納されます。

Gコードモーダル状態	格納値
G90 絶対値指令	900
G91 増分値指令	910

[Md.3034]グループ07モーダル状態(D54526+128s)

グループ07のモーダル状態が格納されます。

Gコードモーダル状態	格納値
G40 工具径補正 キャンセル	400
G41 工具径補正 左	410
G42 工具径補正 右	420

[Md.3035]工具径補正番号(D54527+128s)

工具径補正番号が格納されます。

[Md.3036]工具径補正量(D54528+128s, D54529+128s)

工具径補正量が格納されます。

[Md.3038]グループ08モーダル状態(D54532+128s)

グループ08のモーダル状態が格納されます。

Gコードモーダル状態	格納値
G43 工具長補正(+)	430
G44 工具長補正(-)	440
G49 工具長補正キャンセル	490

[Md.3039]工具長補正番号(D54533+128s)

工具長補正番号が格納されます。最後に指令した補正番号を格納するため、複数の軸に対して補正をかけた場合でも、最後に指令した軸の値が格納されます。

[Md.3040]工具長補正量(D54534+128s, D54535+128s)

工具長補正量が格納されます。最後に指令した補正量を格納するため、複数の軸に対して補正をかけた場合でも、最後に指令した軸の値が格納されます。

[Md.3042]工具長補正軸No.(D54538+128s)

工具長補正を実行している系統内の軸No.がビットデータで格納されます。

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
[Md.3042]工具長補正軸No. (D54538+128s)	未使用								軸8	軸7	軸6	軸5	軸4	軸3	軸2	軸1

*1: 実行中の軸は0/1を格納する。
0: 未実行
1: 実行中

[Md.3046]グループ12モーダル状態(D54542+128s)

グループ12のモーダル状態が格納されます。

Gコードモーダル状態	格納値
G54 ワーク座標系1選択	540
G55 ワーク座標系2選択	550
G56 ワーク座標系3選択	560
G57 ワーク座標系4選択	570
G58 ワーク座標系5選択	580
G59 ワーク座標系6選択	590

[Md.3047]グループ13モーダル状態(D54543+128s)

グループ13のモーダル状態が格納されます。

Gコードモーダル状態	格納値
G61 イグザクトストップチェックモード	610
G61.1 高精度制御モード	611
G62 自動コーナオーバーライド	620
G64 切削モード	640

[Md.3049]グループ15モーダル状態(D54545+128s)

グループ15のモーダル状態が格納されます。

Gコードモーダル状態	格納値
G40.1 法線制御 キャンセル	401
G41.1 法線制御 左オン	411
G42.1 法線制御 右オン	421

[Md.3050]グループ16モーダル状態(D54546+128s)

グループ16のモーダル状態が格納されます。

Gコードモーダル状態	格納値
G68 プログラム座標回転モード開始	680
G69 プログラム座標回転モードキャンセル	690

[Md.3055]グループ21モーダル状態(D54551+128s)

グループ21のモーダル状態が格納されます。

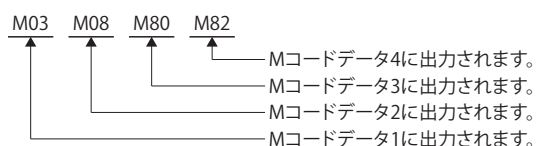
Gコードモーダル状態	格納値
G12.1 極座標補間モード開始	121
G13.1 極座標補間モードキャンセル	131

[Md.3058]Mコードデータ1(D54554+128s, D54555+128s)

- 補助機能(M機能)が指令されたとき、補助機能用アドレスMに続く数値が格納されます。出力するMコードデータは、Gコード制御システムパラメータの「Mバイナリ」で、「8桁BCDコード」、「符号なし32ビットバイナリデータ」、「符号付き32ビットバイナリデータ」を選択できます。
- 自動運転により補助機能(M機能)を指令したときに、“[Md.3058]Mコードデータ1(D54554+128s, D54555+128s)”は更新されます。“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”，または“[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”をONしてもデータは更新されません。また，“[Rq.3380]リセット指令(D54226.4+2s)”をONしてもクリアされません。

Point

- アドレスMに続く数値を、1ブロック内に最大4組まで指令できます。1ブロックに複数個のMをプログラムした場合、プログラムした順番により出力するインターフェースが決まります。



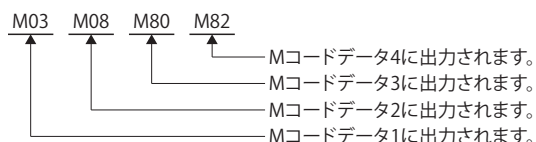
- M98(サブプログラム呼出し)、M99(サブ/マクロプログラム復帰)などは、モーションCPU内部で処理され、Mコードデータとしては出力されません。

[Md.3059]Mコードデータ2(D54556+128s, D54557+128s)

- 補助機能(M機能)が指令されたとき、補助機能用アドレスMに続く数値が格納されます。出力するMコードデータは、Gコード制御システムパラメータの「Mバイナリ」で、「8桁BCDコード」、「符号なし32ビットバイナリデータ」、「符号付き32ビットバイナリデータ」を選択できます。
- 自動運転により、同一ブロックに補助機能(M機能)を2個以上指令したときに、“[Md.3059]Mコードデータ2(D54556+128s, D54557+128s)”は更新されます。“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”，または“[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”をONしてもデータは更新されません。また，“[Rq.3380]リセット指令(D54226.4+2s)”をONしてもクリアされません。

Point

- アドレスMに続く数値を、1ブロック内に最大4組まで指令できます。1ブロックに複数個のMをプログラムした場合、プログラムした順番により出力するインターフェースが決まります。



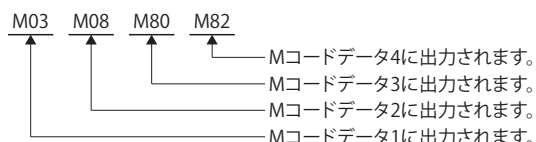
- M98(サブプログラム呼出し), M99(サブ/マクロプログラム復帰)などは、モーションCPU内部で処理され、Mコードデータとしては出力されません。

[Md.3060]Mコードデータ3(D54558+128s, D54559+128s)

- 補助機能(M機能)が指令されたとき、補助機能用アドレスMに続く数値が格納されます。出力するMコードデータは、Gコード制御システムパラメータの「Mバイナリ」で、「8桁BCDコード」、「符号なし32ビットバイナリデータ」、「符号付き32ビットバイナリデータ」を選択できます。
- 自動運転により、同一ブロックに補助機能(M機能)を3個以上指令したときに、“[Md.3060]Mコードデータ3(D54558+128s, D54559+128s)”は更新されます。“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”，または“[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”をONしてもデータは更新されません。また，“[Rq.3380]リセット指令(D54226.4+2s)”をONしてもクリアされません。

Point

- アドレスMに続く数値を、1ブロック内に最大4組まで指令できます。1ブロックに複数個のMをプログラムした場合、プログラムした順番により出力するインターフェースが決まります。



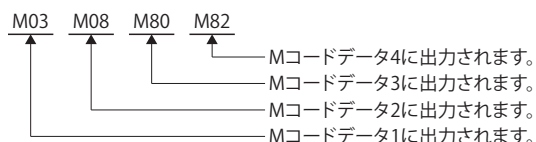
- M98(サブプログラム呼出し), M99(サブ/マクロプログラム復帰)などは、モーションCPU内部で処理され、Mコードデータとしては出力されません。

[Md.3061]Mコードデータ4(D54560+128s, D54561+128s)

- 補助機能(M機能)が指令されたとき、補助機能用アドレスMに続く数値が格納されます。出力するMコードデータは、Gコード制御系統パラメータの「Mバイナリ」で、「8桁BCDコード」、「符号なし32ビットバイナリデータ」、「符号付き32ビットバイナリデータ」を選択できます。
- 自動運転により、同一ブロックに補助機能(M機能)を4個以上指令したときに、“[Md.3061]Mコードデータ4(D54560+128s, D54561+128s)”は更新されます。“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”，または“[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”をONしてもデータは更新されません。また，“[Rq.3380]リセット指令(D54226.4+2s)”をONしてもクリアされません。

Point

- アドレスMに続く数値を、1ブロック内に最大4組まで指令できます。1ブロックに複数個のMをプログラムした場合、プログラムした順番により出力するインターフェースが決まります。



- M98(サブプログラム呼出し)，M99(サブ/マクロプログラム復帰)などは、モーションCPU内部で処理され、Mコードデータとしては出力されません。

[Md.3074]ローカル変数深さ(D54582+128s)

マクロ呼出し(G65)中のローカル変数深さ(0~4)が格納されます。

“[Md.3074]ローカル変数深さ(D54582+128s)”は、下記の場合に更新されます。

- メインプログラム，またはサブプログラムからマクロプログラムを呼出したとき。
- マクロプログラムからメインプログラム，またはサブプログラムへ戻ったとき。

また，リセット指令を行うと，“[Md.3074]ローカル変数深さ(D54582+128s)”は，0クリアされます。

[Md.3070]実行中プログラムコメント(D54588+128s~D54595+128s)

実行中プログラムのプログラムコメントをASCIIコードで8ワード分格納します。

プログラムスタートに設定したエンドオブレコード(%)の次ブロックのコメント(「(」から「)」までの文字列)を15文字+NULL文字で格納します。

3.9 Gコード制御システムモニタデバイス(拡張)

デバイス番号		信号名称
R標準配置方式	Q互換配置方式	
D55264～D55423		系統1のGコード制御システムモニタデバイス(拡張)
D55424～D55583		系統2のGコード制御システムモニタデバイス(拡張)

・各系統の詳細内容

デバイス番号		記号	信号名称	リフレッシュ 周期	取込周期	信号種別
R標準配置方式	Q互換配置方式					
D55264+160s		Md.3178	実行中プログラムモニタ(1行目)	ブロック切り 替時	—	モニタデバイス
D55265+160s						
D55266+160s						
D55267+160s						
D55268+160s						
D55269+160s						
D55270+160s						
D55271+160s						
D55272+160s						
D55273+160s						
D55274+160s						
D55275+160s						
D55276+160s						
D55277+160s						
D55278+160s						
D55279+160s						
D55280+160s						
D55281+160s						
D55282+160s						
D55283+160s						
D55284+160s						
D55285+160s						
D55286+160s						
D55287+160s						
D55288+160s						
D55289+160s						
D55290+160s						
D55291+160s						
D55292+160s						
D55293+160s						
D55294+160s						
D55295+160s						

デバイス番号		記号	信号名称	リフレッシュ 周期	取込周期	信号種別
R標準配置方式	Q互換配置方式					
D55296+160s		Md.3179	実行中プログラムモニタ(2行目)	ブロック切り 替時	—	モニタデバイス
D55297+160s						
D55298+160s						
D55299+160s						
D55300+160s						
D55301+160s						
D55302+160s						
D55303+160s						
D55304+160s						
D55305+160s						
D55306+160s						
D55307+160s						
D55308+160s						
D55309+160s						
D55310+160s						
D55311+160s						
D55312+160s						
D55313+160s						
D55314+160s						
D55315+160s						
D55316+160s						
D55317+160s						
D55318+160s						
D55319+160s						
D55320+160s						
D55321+160s						
D55322+160s						
D55323+160s						
D55324+160s						
D55325+160s						
D55326+160s						
D55327+160s						

デバイス番号		記号	信号名称	リフレッシュ 周期	取込周期	信号種別
R標準配置方式	Q互換配置方式					
D55328+160s		Md.3180	実行中プログラムモニタ(3行目)	ブロック切り 替時	—	モニタデバイス
D55329+160s						
D55330+160s						
D55331+160s						
D55332+160s						
D55333+160s						
D55334+160s						
D55335+160s						
D55336+160s						
D55337+160s						
D55338+160s						
D55339+160s						
D55340+160s						
D55341+160s						
D55342+160s						
D55343+160s						
D55344+160s						
D55345+160s						
D55346+160s						
D55347+160s						
D55348+160s						
D55349+160s						
D55350+160s						
D55351+160s						
D55352+160s						
D55353+160s						
D55354+160s						
D55355+160s						
D55356+160s						
D55357+160s						
D55358+160s						
D55359+160s						
D55360+160s		—	ユーザ使用不可	—	—	—
D55361+160s						
D55362+160s						
D55363+160s						
D55364+160s						
D55365+160s						
D55366+160s						
D55367+160s						
D55368+160s						
D55369+160s						
D55370+160s						
D55371+160s						
D55372+160s						
D55373+160s						
D55374+160s						
D55375+160s						
D55376+160s						
D55377+160s						
D55378+160s						
D55379+160s						

デバイス番号		記号	信号名称	リフレッシュ 周期	取込周期	信号種別
R標準配置方式	Q互換配置方式					
D55380+160s		—	ユーザ使用不可	—	—	—
D55381+160s						
D55382+160s						
D55383+160s						
D55384+160s						
D55385+160s						
D55386+160s						
D55387+160s						
D55388+160s						
D55389+160s						
D55390+160s						
D55391+160s						
D55392+160s						
D55393+160s						
D55394+160s						
D55395+160s						
D55396+160s						
D55397+160s						
D55398+160s						
D55399+160s						
D55401+160s						
D55402+160s						
D55403+160s						
D55404+160s						
D55405+160s						
D55406+160s						
D55407+160s						
D55408+160s						
D55409+160s						
D55410+160s						
D55411+160s						
D55412+160s						
D55413+160s						
D55414+160s						
D55415+160s						
D55416+160s						
D55417+160s						
D55418+160s						
D55419+160s						
D55420+160s						
D55421+160s						
D55422+160s						
D55423+160s						

[Md.3178]実行中プログラムモニタ(1行目)(D55264+160s~D55295+160s)

起動しているGコードプログラムの実行中ブロック(1行目)の1ブロック分をASCII文字列として格納します。
1ブロックを最大64文字(63文字+¥0)で格納します。

例

系統1の実行中Gコードプログラムの場合

プログラムブロック	プログラム
1行目(実行中)	G91 G00 X100. Y100. Z100. C50.
2行目	G90 G94 G17 G00 X1000.0000 Y1000.0000 Z1000.0000 C180.0000 F10000.00 M100 M200 M300
3行目	G04 P1

• [Md.3178]実行中プログラムモニタ(1行目)に格納されるデバイス値

項目	デバイス番号															
	D55264		D55265		D55266		D55267		D55268		D55269		D55270		D55271	
	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位
ASCII文字列	G	9	1	G	0	0	X	1	0	0	.	Y	1	0	0	.

項目	デバイス番号															
	D55272		D55273		D55274		D55275		D55276		D55277		D55278		D55279	
	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位
ASCII文字列	Z	1	0	0	.	C	5	0	.	¥0	¥0	¥0	¥0	¥0	¥0	¥0

項目	デバイス番号															
	D55280		D55281		D55282		D55283		D55284		D55285		D55286		D55287	
	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位
ASCII文字列	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0

項目	デバイス番号															
	D55288		D55289		D55290		D55291		D55292		D55293		D55294		D55295	
	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位
ASCII文字列	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0

• [Md.3179]実行中プログラムモニタ(2行目)に格納されるデバイス値

項目	デバイス番号															
	D55296		D55297		D55298		D55299		D55300		D55301		D55302		D55303	
	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位
ASCII文字列	G	9	0	G	9	4	G	1	7	G	0	0	X	1	0	0

項目	デバイス番号															
	D55304		D55305		D55306		D55307		D55308		D55309		D55310		D55311	
	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位
ASCII文字列	0	.	0	0	0	0	Y	1	0	0	0	.	0	0	0	0

項目	デバイス番号															
	D55312		D55313		D55314		D55315		D55316		D55317		D55318		D55319	
	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位
ASCII文字列	Z	1	0	0	0	.	0	0	0	0	C	1	8	0	.	0

項目	デバイス番号															
	D55320		D55321		D55322		D55323		D55324		D55325		D55326		D55327	
	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位
ASCII文字列	0	0	0	F	1	0	0	0	0	.	0	0	M	1	0	¥0

- [Md.3180]実行中プログラムモニタ(3行目)に格納されるデバイス値

項目	デバイス番号															
	D55328		D55329		D55330		D55331		D55332		D55333		D55334		D55335	
	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位
ASCII文字列	G	0	4	P	1	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0

項目	デバイス番号															
	D55336		D55337		D55338		D55339		D55340		D55341		D55342		D55343	
	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位
ASCII文字列	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0

項目	デバイス番号															
	D55344		D55345		D55346		D55347		D55348		D55349		D55350		D55351	
	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位
ASCII文字列	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0

項目	デバイス番号															
	D55352		D55353		D55354		D55355		D55356		D55357		D55358		D55359	
	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位	下位	上位
ASCII文字列	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0	¥0

■注意事項

- Gコードプログラム内のスペースは削除してブロックを表示するため、スペースは格納されません。
- コメント行は格納されません。
- 1ブロックの文字数が64文字以上の場合、63文字目までの文字と64文字目に「¥0」を格納します。

[Md.3179]実行中プログラムモニタ(2行目)(D55296+160s~D55327+160s)

起動しているGコードプログラムの実行中ブロック(1行目)から2行目の1ブロック分をASCII文字列として格納します。

1ブロックは最大64文字(63文字+¥0)を格納します。

格納するプログラムの例は、“[Md.3178]実行中プログラムモニタ(1行目)(D55264+160s~D55295+160s)”を参照してください。(☞ 59ページ [Md.3178]実行中プログラムモニタ(1行目)(D55264+160s~D55295+160s))

[Md.3180]実行中プログラムモニタ(3行目)(D55328+160s~D55359+160s)

起動しているGコードプログラムの実行中ブロック(1行目)から3行目の1ブロック分をASCII文字列として格納します。

1ブロックは最大64文字(63文字+¥0)を格納します。

格納するプログラムの例は、“[Md.3178]実行中プログラムモニタ(1行目)(D55264+160s~D55295+160s)”を参照してください。(☞ 59ページ [Md.3178]実行中プログラムモニタ(1行目)(D55264+160s~D55295+160s))

3.10 Gコード制御軸ステータス

デバイス番号		信号名称
R標準配置方式	Q互換配置方式	
D54448, D54449		系統1 軸1のGコード制御軸ステータス
D54450, D54451		系統1 軸2のGコード制御軸ステータス
D54452, D54453		系統1 軸3のGコード制御軸ステータス
D54454, D54455		系統1 軸4のGコード制御軸ステータス
D54456, D54457		系統1 軸5のGコード制御軸ステータス
D54458, D54459		系統1 軸6のGコード制御軸ステータス
D54460, D54461		系統1 軸7のGコード制御軸ステータス
D54462, D54463		系統1 軸8のGコード制御軸ステータス
D54464, D54465		系統2 軸1のGコード制御軸ステータス
D54466, D54467		系統2 軸2のGコード制御軸ステータス
D54468, D54469		系統2 軸3のGコード制御軸ステータス
D54470, D54471		系統2 軸4のGコード制御軸ステータス
D54472, D54473		系統2 軸5のGコード制御軸ステータス
D54474, D54475		系統2 軸6のGコード制御軸ステータス
D54476, D54477		系統2 軸7のGコード制御軸ステータス
D54478, D54479		系統2 軸8のGコード制御軸ステータス

・各系統の詳細内容

デバイス番号		記号	信号名称	リフレッシュ 周期	取込周期	信号種別
R標準配置方式	Q互換配置方式					
D54448.0+2sn		St.3076	スムージングゼロ	Gコード制御演 算周期	—	ステータス信号
D54448.1+2sn		—	ユーザ使用不可	—	—	—
D54448.2+2sn						
D54448.3+2sn						
D54448.4+2sn						
D54448.5+2sn						
D54448.6+2sn						
D54448.7+2sn						
D54448.8+2sn						
D54448.9+2sn						
D54448.A+2sn						
D54448.B+2sn						
D54448.C+2sn						
D54448.D+2sn						
D54448.E+2sn						
D54448.F+2sn						
D54449.0+2sn						
D54449.1+2sn						
D54449.2+2sn						
D54449.3+2sn						
D54449.4+2sn						
D54449.5+2sn						
D54449.6+2sn						
D54449.7+2sn						
D54449.8+2sn						
D54449.9+2sn						
D54449.A+2sn						
D54449.B+2sn						

デバイス番号		記号	信号名称	リフレッシュ 周期	取込周期	信号種別
R標準配置方式	Q互換配置方式					
D54449.C+2sn		—	ユーザ使用不可	—	—	—
D54449.D+2sn						
D54449.E+2sn						
D54449.F+2sn						

[St.3076]スムージングゼロ(D54448.0+2sn)

- 加減速処理が終了し，指令が残っていないとONする信号です。
- 自動運転で指令した移動量を加減速における遅れ量も含め，すべての出力処理が完了するとONします。
- 移動指令を実行中および加減速における遅れ量があるとOFFします。

Point

非常に低速で移動時は，移動中であっても“[St.3076]スムージングゼロ(D54448.0+2sn)”がONすることがあります。

3.11 Gコード制御軸モニタデバイス

デバイス番号		信号名称
R標準配置方式	Q互換配置方式	
D54752～D54783		系統1 軸1のGコード制御軸モニタデバイス
D54784～D54815		系統1 軸2のGコード制御軸モニタデバイス
D54816～D54847		系統1 軸3のGコード制御軸モニタデバイス
D54848～D54879		系統1 軸4のGコード制御軸モニタデバイス
D54880～D54911		系統1 軸5のGコード制御軸モニタデバイス
D54912～D54943		系統1 軸6のGコード制御軸モニタデバイス
D54944～D54975		系統1 軸7のGコード制御軸モニタデバイス
D54976～D55007		系統1 軸8のGコード制御軸モニタデバイス
D55008～D55039		系統2 軸1のGコード制御軸モニタデバイス
D55040～D55071		系統2 軸2のGコード制御軸モニタデバイス
D55072～D55103		系統2 軸3のGコード制御軸モニタデバイス
D55104～D55135		系統2 軸4のGコード制御軸モニタデバイス
D55136～D55167		系統2 軸5のGコード制御軸モニタデバイス
D55168～D55199		系統2 軸6のGコード制御軸モニタデバイス
D55200～D55231		系統2 軸7のGコード制御軸モニタデバイス
D55232～D55263		系統2 軸8のGコード制御軸モニタデバイス

・各系統の詳細内容

デバイス番号		記号	信号名称	リフレッシュ 周期	取込周期	信号種別	
R標準配置方式	Q互換配置方式						
D54752+32sn		Md.3144	軸No.	STOP→RUN時	—	モニタデバイス	
D54753+32sn		Md.3145	軸名称				
D54754+32sn		Md.3146	回転軸設定状態				
D54755+32sn		Md.3153	タンデム機能有効情報	STOP→RUN時	—	—	
D54756+32sn		Md.3154	ローカル座標オフセット	Gコード制御移行時/G52指令実行時/G54～G59指令実行時			
D54757+32sn							
D54758+32sn		—	ユーザ使用不可	—	—	—	
D54759+32sn							
D54760+32sn							
D54761+32sn							
D54762+32sn							
D54763+32sn							
D54764+32sn							
D54765+32sn							
D54766+32sn							
D54767+32sn							
D54768+32sn		Md.3147	機械位置	演算周期	—	モニタデバイス	
D54769+32sn							
D54770+32sn		Md.3148	機械目標位置	Gコード制御演算周期			
D54771+32sn							
D54772+32sn		Md.3149	相対位置	演算周期			
D54773+32sn							
D54774+32sn		Md.3150	相対目標位置	Gコード制御演算周期	—	—	
D54775+32sn							
D54776+32sn		—	ユーザ使用不可	—			
D54777+32sn							

デバイス番号		記号	信号名称	リフレッシュ 周期	取込周期	信号種別
R標準配置方式	Q互換配置方式					
D54778+32sn		Md.3152	プログラム目標位置	Gコード制御演 算周期	—	モニタデバイス
D54779+32sn						
D54780+32sn		—	ユーザ使用不可	—	—	—
D54781+32sn						
D54782+32sn						
D54783+32sn						

Point

- 固定パラメータの単位設定にかかわらず、Gコード制御用デバイスのアドレスには少数点以下4桁で格納します。“[Md.20]送り現在値(R: D32000+48n, D32001+48n/Q: D0+20n, D1+20n)”や“[Md.101]実現在値(R: D32002+48n, D32003+48n/Q: D2+20n, D3+20n)”などのデバイスモニタ値と小数点以下の桁数が異なりますので、“[Rq.3376]Gコード制御要求(D54226.0+2s)”のOFF→ON時に確立される“[Md.3147]機械位置(D54768+32n, D54769+32n)”のモニタ値とアドレスを比較する際は注意してください。
- Gコード制御軸パラメータの「回転軸」を設定し、回転軸タイプに「近回り有効」、または「近回り無効」を設定すると、固定パラメータの「単位設定」にかかわらず、Gコード制御用デバイスのアドレスには、0～359.9999[degree]の範囲で格納されます。回転軸を使用する場合の単位設定には、「degree」を推奨します。

[Md.3144]軸No.(D54752+32sn)

[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“系統軸情報”⇒“軸No.”にて設定した軸No.が格納されます。

[Md.3145]軸名称(D54753+32sn)

[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“系統軸情報”⇒“軸名称”にて設定した軸名称の番号が格納されます。

軸名称の番号には、下記の値が格納されます。

モニタ値	軸名称
0	設定なし
1	X
2	Y
3	Z
4	A
5	B
6	C
7	U
8	V
9	W

[Md.3146]回転軸設定状態(D54754+32sn)

[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“系統軸情報”⇒“回転軸”にて設定した情報が格納されます。

モニタ値	内容
0	直線軸
1	回転軸

[Md.3153] タンデム機能有効情報(D54755+32sn)

タンデム機能でのマスタ制御軸No.が格納されます。

モニタ値	内容
0	タンデム機能未使用(マスタ制御軸の場合も0となります。)
1	タンデム機能使用時マスタ制御軸No.

[Md.3154] ローカル座標オフセット(D54756+32sn, D54757+32sn)

現在選択されている座標系における、ローカル座標系設定値が格納されます。

[Md.3147] 機械位置(D54768+32sn, D54769+32sn)

“[St.3208]Gコード制御中(D54440.0+4s)”がONの状態にて、制御軸の機械位置が格納されます。

“[Rq.3376]Gコード制御要求(D54226.0+2s)”のOFF→ONにより、“[Md.20]送り現在値(R: D32000+48n, D32001+48n/Q: D0+20n, D1+20n)”の位置より確立します。

[Md.3148] 機械目標位置(D54770+32sn, D54771+32sn)

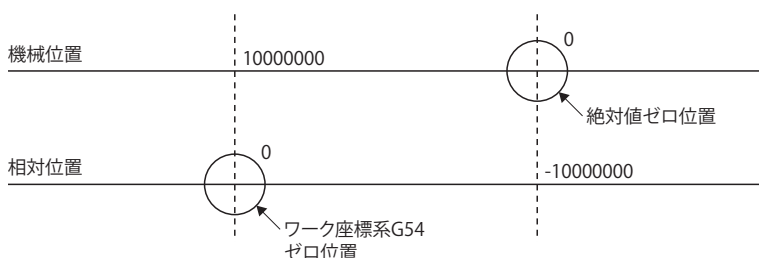
実行中のブロックにおける機械位置の終点が格納されます。

[Md.3149] 相対位置(D54772+32sn, D54773+32sn)

ワーク座標系選択(G54～G59)の現在値を格納します。

例

G54に1000.を設定している場合



機械位置で10000000の位置にいる場合は、「相対位置=0」となります。

相対位置は、ワーク座標系選択(G54～G59), またはローカル座標系設定(G52)によりシフトします。

[Md.3150] 相対目標位置(D54774+32sn, D54775+32sn)

実行中のブロックにおける相対位置の終点が格納されます。

[Md.3152] プログラム目標位置(D54778+32sn, D54779+32sn)

実行中のブロックにおけるプログラム位置の終点が格納されます。

3.12 内部リレー (M)／データレジスタ(D)の有効可否

Gコード制御中における内部リレー (M)とデータレジスタ(D)の共通デバイス、各軸デバイスの有効可否を下記に示します。

内部リレー

共通デバイス

○: 有効, ×: 無効

デバイス番号		記号	信号名称	Gコード 制御中	備考
R標準配置方式	Q互換配置方式				
M30000	M2000	Rq.1120	シーケンサレディフラグ	○	
M30001 ～ M30037	—	—	ユーザ使用不可	—	
M30038	M2038	St.1041	モーションSFCデバッグモード中フラグ	○	
M30039	M2039	—	ユーザ使用不可	—	
M30040	M2040	Rq.1122	速度切換えポイント指定フラグ	×	
M30041	M2041	—	ユーザ使用不可	—	
M30042	M2042	Rq.1123	全軸サーボON指令	×	Gコード制御中は、サーボOFFしません。(Gコード制御は終了しません。) Gコード制御を終了したとき、その時点の指令状態が有効になります。
M30043 ～ M30047	M2043 ～ M2047	—	ユーザ使用不可	—	
M30048	M2048	Rq.1124	JOG運転同時始動指令	×	
M30049	M2049	St.1045	全軸サーボON受付けフラグ	○	
M30050	M2050	—	ユーザ使用不可	—	
M30051	M2051	Rq.1125	手動パルス1許可フラグ	×	
M30052	M2052	Rq.1126	手動パルス2許可フラグ	×	
M30053	M2053	Rq.1127	手動パルス3許可フラグ	×	
M30054	M2054	St.1046	演算周期オーバーフラグ	○	
M30055 ～ M30079	—	—	ユーザ使用不可	—	
M30080 ～ M30143	M2001 ～ M2032	St.1040	始動受付けフラグ	○	Gコード制御開始時にONします。 Gコード制御終了時にOFFします。
M30144 ～ M30207	M2061 ～ M2092	St.1047	速度変更受付け中フラグ	×	
M30208 ～ M30271	M2128 ～ M2159	St.1048	自動減速中フラグ	×	Gコード制御開始時にOFFします。
M30272 ～ M30335	M2240 ～ M2271	St.1049	速度変更「0」受付け中フラグ	×	
M30336 ～ M30399	M2272 ～ M2303	St.1050	制御ループモニタステータス	○	
M30400 ～ M30639	—	—	ユーザ使用不可	—	

各軸ステータス

○: 有効, ×: 無効

デバイス番号		記号	信号名称		Gコード 制御中	備考
R標準配置方式	Q互換配置方式					
M32400+32n	M2400+20n	St.1060	位置決め始動完了		×	Gコード制御開始時にOFFします。
M32401+32n	M2401+20n	St.1061	位置決め完了		×	
M32402+32n	M2402+20n	St.1062	インポジション		○	演算周期ごとに更新します。
M32403+32n	M2403+20n	St.1063	指令インポジション		×	Gコード制御開始時にOFFします。
M32404+32n	M2404+20n	St.1064	速度制御中		×	
M32405+32n	M2405+20n	St.1065	速度・位置切換えラッチ		×	
M32406+32n	M2406+20n	St.1066	零点通過		○	
M32407+32n	M2407+20n	St.1067	エラー検出		○	Gコード制御で検出したエラーは更新されません。Gコード制御以外で検出したエラーは更新されます。
M32408+32n	M2408+20n	St.1068	サーボエラー検出		○	
M32409+32n	M2409+20n	St.1069	原点復帰要求		○	ONしていた場合、Gコード制御要求時に軽度エラー (エラーコード: 19A6H) となります。
M32410+32n	M2410+20n	St.1070	原点復帰完了		○	ONしていた場合、Gコード制御開始時にOFFします。
M32411+32n	M2411+20n	St.1071	外部信号	FLS	○	
M32412+32n	M2412+20n	St.1072		RLS	○	
M32413+32n	M2413+20n	St.1073		STOP	○	
M32414+32n	M2414+20n	St.1074		DOG/CHANGE	○	
M32415+32n	M2415+20n	St.1075	サーボレディ		○	
M32416+32n	M2416+20n	St.1076	トルク制限中		○	
M32417+32n	M2417+20n	—	ユーザ使用不可		—	
M32418+32n	M2418+20n					
M32419+32n	M2419+20n	St.1079	Mコード出力中		×	
M32420+32n ～ M32431+32n		—	ユーザ使用不可		—	

各軸指令信号

○: 有効, ×: 無効

デバイス番号		記号	信号名称	Gコード 制御中	備考
R標準配置方式	Q互換配置方式				
M34480+32n	M3200+20n	Rq.1140	停止指令	×	
M34481+32n	M3201+20n	Rq.1141	急停止指令	×	
M34482+32n	M3202+20n	Rq.1142	正転JOG始動指令	×	
M34483+32n	M3203+20n	Rq.1143	逆転JOG始動指令	×	
M34484+32n	M3204+20n	Rq.1144	完了信号OFF指令	×	
M34485+32n	M3205+20n	Rq.1145	速度・位置切換え許可指令	×	
M34486+32n	M3206+20n	—	ユーザ使用不可	—	
M34487+32n	M3207+20n	Rq.1147	エラーリセット指令	○	Gコード制御に関するエラーはリセットしません。Gコード制御以外の制御で発生している軸のエラーに対して有効です。
M34488+32n	M3208+20n	Rq.1148	サーボエラーリセット指令	○	
M34489+32n	M3209+20n	Rq.1149	始動時の外部STOP入力無効指令	×	
M34490+32n	M3210+20n	—	ユーザ使用不可	—	
M34491+32n	M3211+20n				
M34492+32n	M3212+20n	Rq.1152	送り現在値更新指令	×	
M34493+32n	M3213+20n	—	ユーザ使用不可	—	
M34494+32n	M3214+20n				
M34495+32n	M3215+20n	Rq.1155	サーボOFF指令	×	Gコード制御中は、サーボOFFしません。(Gコード制御は終了しません。) Gコード制御を終了したとき、その時点の指令状態が有効になります。
M34496+32n	M3216+20n	Rq.1156	ゲイン切換え指令	○	
M34497+32n	M3217+20n	Rq.1157	PI-PID切換え指令	○	
M34498+32n	M3218+20n	Rq.1158	制御ルー プ切換え指令	○	
M34499+32n	M3219+20n	Rq.1159	FIN信号	×	
M34500+32n ～ M34511+32n		—	ユーザ使用不可	—	

データレジスタ

各軸モニタデバイス

○: 有効, ×: 無効

デバイス番号		記号	信号名称	Gコード 制御中	備考
R標準配置方式	Q互換配置方式				
D32000+48n	D0+20n	Md.20	送り現在値	○	演算周期ごとに更新されます。
D32001+48n	D1+20n				
D32002+48n	D2+20n	Md.101	実現在値	○	
D32003+48n	D3+20n				
D32004+48n	D4+20n	Md.102	偏差カウンタ値	○	
D32005+48n	D5+20n				
D32006+48n	D6+20n	Md.1003	警告コード	○	Gコード制御で検出したエラーは更新されません。Gコード制御以外で検出したエラーは更新されます。
D32007+48n	D7+20n	Md.1004	エラーコード	○	
D32008+48n	D8+20n	Md.1005	サーボエラーコード	○	
D32009+48n	D9+20n	Md.1006	原点復帰再移動量	×	
D32010+48n	D10+20n	Md.34	近点ドグON後の移動量	×	
D32011+48n	D11+20n				
D32012+48n	D12+20n	Md.1008	実行プログラムNo.	×	Gコード制御開始時にクリアします。実行中のGコードプログラムNo.は格納しません。
D32013+48n	D13+20n	Md.25	Mコード	×	
D32014+48n	D14+20n	Md.35	トルク制限値	○	
D32015+48n	D15+20n	Md.1011	連続軌跡制御用データセットポインタ	×	
D32016+48n	D16+20n	—	ユーザ使用不可	—	
D32017+48n	D17+20n				
D32018+48n	D18+20n	Md.1012	STOP入力時の実現在値	×	
D32019+48n	D19+20n				
D32020+48n	#8001+20n	Md.104	モータ電流値	○	
D32021+48n	#8017+20n	—	ユーザ使用不可	—	
D32022+48n	#8002+20n	Md.103	モータ回転数	○	
D32023+48n	#8003+20n				
D32024+48n	#8004+20n	Md.28	指令速度	○	
D32025+48n	#8005+20n				
D32026+48n	#8006+20n	Md.100	原点復帰再移動量	×	
D32027+48n	#8007+20n				
D32028+48n	#8008+20n	Md.1019	サーボアンプ表示サーボエラーコード	○	
D32029+48n	#8009+20n	Md.107	パラメータエラー番号	○	
D32030+48n	#8000+20n	Md.1014	サーボアンプ種別	○	
D32031+48n	#8016+20n	Md.1027	サーボアンプベンダID	○	
D32032+48n	#8010+20n	Md.108	サーボステータス1	○	
D32033+48n	#8011+20n	Md.1022	サーボステータス2	○	
D32034+48n	#8012+20n	Md.125	サーボステータス3	○	
D32035+48n	#8013+20n	—	ユーザ使用不可	—	
D32036+48n	#8014+20n				
D32037+48n	#8015+20n				
D32038+48n	#8018+20n	Md.500	サーボステータス7	○	
D32039+48n	#8019+20n	—	ユーザ使用不可	—	
D32040+48n					
D32047+48n					

4 Gコード制御用パラメータ

Gコード制御で使用するパラメータについて説明します。

Rシリーズ共通パラメータ、モーションCPU共通パラメータについては、下記を参照してください。

📖 MELSEC iQ-R モーションコントローラプログラミングマニュアル(共通編)

モーション制御パラメータについては、下記を参照してください。

📖 MELSEC iQ-R モーションコントローラプログラミングマニュアル(位置決め制御編)

Point

Gコード制御を使用する場合、以下の設定を行ってください。以下のように設定されていないとGコード制御を行えません。

- [モーションCPU共通パラメータ]⇒[基本設定]⇒“演算周期設定”にて「0.888ms」以上に設定する。(📖 23 ページ Gコード制御の制御周期)
 - [モーションCPU共通パラメータ]⇒[基本設定]⇒“Gコード制御設定”にて使用する系統により、「系統1のみ使用する」、または「系統1, 系統2を使用する」に設定する。
-

4.1 Gコード制御システムパラメータ

Gコード制御システムごとにシステム内で使用する軸数などのパラメータを設定します。

🔗 [モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御システムパラメータ]

No.	項目		初期値	設定範囲	直接設定 ^{*1}	間接設定 ^{*2}		詳細説明項
					可否	可否(使用サイズ)	取込周期	
1	システム基本設定	システム内軸数	1	1～8: システム内軸数	○	×	—	📖 73 ページ
2	モーダル初期設定	補間モード ^{*3}	0	0: 直線補間(G01指令状態) 1: 位置決め(G00指令状態)	○	×	—	📖 73 ページ
3		平面選択 ^{*3}	0	0: X-Y平面(G17指令状態) 1: X-Y平面(G17指令状態) 2: Z-X平面(G18指令状態) 3: Y-Z平面(G19指令状態)	○	×	—	📖 73 ページ
4		絶対値設定 ^{*3}	0	0: 増分値設定(G91指令状態) 1: 絶対値設定(G90指令状態)	○	×	—	📖 73 ページ
5	制御設定	G00非補間 ^{*3}	0	0: 終点に向け、直線で移動する。(補間タイプ) 1: 各軸の終点に向け、各軸の早送り速度で移動する。(非補間タイプ)	○	×	—	📖 74 ページ
6		減速チェック ^{*3}	0000H	<div> <div>H 0 0 □ □</div> <div> <div>→ G00の減速チェック方式</div> <div>0: 指令減速方式</div> <div>1: インポジションチェック方式</div> <div>2: スムージングチェック方式</div> </div> <div>→ G01の減速チェック方式</div> <div>0: 指令減速方式</div> <div>1: インポジションチェック方式</div> <div>2: スムージングチェック方式</div> </div>	○	×	—	📖 74 ページ
7		G1→G1減速チェック ^{*3}	1	0: 減速チェック実施しない 1: 減速チェック実施する	○	×	—	📖 75 ページ
8		円弧誤差 ^{*3}	100(×10 ⁻⁴ [mm])	1～1000(×10 ⁻⁴ [mm])	○	×	—	📖 75 ページ
9		円弧中心誤差修正許容値 ^{*3}	0(×10 ⁻⁴ [mm])	-1000～100(×10 ⁻⁴ [mm])	○	×	—	📖 75 ページ
10		自動コーナオーバーライド ^{*3}	0[%]	0～100[%]	○	×	—	📖 75 ページ
11		自動コーナオーバーライド最大角度 ^{*3}	0[degree]	0～180[degree]	○	×	—	📖 75 ページ
12		自動コーナオーバーライドコーナ前長さ ^{*3}	0(×10 ⁻⁴ [mm])	0～99999999(×10 ⁻⁴ [mm])	○	×	—	📖 75 ページ
13	オーバーライド設定	早送りオーバーライド ^{*3}	—	—	×	○ (1ワード)	Gコード制御演算周期	📖 75 ページ
14		切削送りオーバーライド ^{*3}	—	—	×	○ (1ワード)		📖 75 ページ
15		オーバーライドキャンセル ^{*3}	—	—	×	○ (1ビット)		📖 75 ページ
16	平面構成	基本軸I ^{*3}	1	0: 設定なし 1: X 2: Y 3: Z 4: A 5: B 6: C 7: U 8: V 9: W	○	×	—	📖 76 ページ
17		基本軸J ^{*3}	2		○	×	—	
18		基本軸K ^{*3}	3		○	×	—	

No.	項目		初期値	設定範囲	直接設定 ^{*1}	間接設定 ^{*2}		詳細説明項
					可否	可否(使用サイズ)	取込周期	
19	法線制御	法線制御軸 ^{*3}	0	0: 法線制御しない 1: X 2: Y 3: Z 4: A 5: B 6: C 7: U 8: V 9: W	○	×	—	76 ページ
20		法線制御タイプ ^{*3}	0	0: 法線制御タイプI 1: 法線制御タイプII	○	×	—	76 ページ
21		旋回最小角 ^{*3}	0(×10 ⁻⁴ [degree])	0~3600000(×10 ⁻⁴ [degree])	○	×	—	76 ページ
22		法線制御軸旋回速度 ^{*3}	2000[degree/min]	1~100000[degree/min]	○	×	—	76 ページ
23		旋回最小移動量 ^{*3}	0(×10 ⁻⁴ [mm])	0~99999999(×10 ⁻⁴ [mm])	○	×	—	76 ページ
24		法線制御軸旋回径 ^{*3}	0(×10 ⁻⁴ [mm])	0~99999999(×10 ⁻⁴ [mm])	○	×	—	77 ページ
25		法線制御軸挿入径 ^{*3}	0(×10 ⁻⁴ [mm])	0~99999999(×10 ⁻⁴ [mm])	○	×	—	77 ページ
26	補助機能	M/バイナリ ^{*3}	0	0: BCD 1: 符号なしバイナリ -1: 符号付きバイナリ	○	×	—	77 ページ
27	極座標補間	極座標補間直線軸 ^{*3}	0	0: 設定なし 1: X 2: Y 3: Z 4: A 5: B 6: C 7: U 8: V 9: W	○	×	—	77 ページ
28		極座標補間回転軸 ^{*3}	0		○	×	—	77 ページ
29	高精度制御	補間前加減速-最高速度 ^{*3}	2000[mm/min]	1~999999[mm/min]	○	×	—	78 ページ
30		補間前加減速-時定数 ^{*3}	100[ms]	1~5000[ms]	○	×	—	78 ページ
31		各軸許容加速度制御有効 ^{*3}	0	0: 最適コーナ減速 1: 各軸許容加速度制御	○	×	—	78 ページ
32		コーナ減速角度 ^{*3}	5[degree]	1~89[degree]	○	×	—	78 ページ
33		コーナ精度係数 ^{*3}	0[%]	-1000~99[%]	○	×	—	78 ページ
34		曲線精度係数 ^{*3}	0[%]	-1000~99[%]	○	×	—	78 ページ
35		円弧減速速度切換 ^{*3}	0	0: 減速しない 1: 減速する	○	×	—	78 ページ
36		円弧減速速度 ^{*3}	1[mm/min]	1~999999[mm/min]	○	×	—	78 ページ

No.	項目		初期値	設定範囲	直接設定 ^{*1}	間接設定 ^{*2}		詳細説明項
					可否	可否(使用サイズ)	取込周期	
37	マクロ制御	系統共通コモン変数点数	0	0~900: 系統共通コモン変数点数 ^{*4}	○	×	—	79 ページ
38		系統別コモン変数点数	0	0~900: 系統別コモン変数点数 ^{*4}	○	×	—	79 ページ
39		系統共通コモン変数先頭デバイス番号	—	ワードバイス(D, W, #, U3E□¥G ^{*5} , U3E□¥HG ^{*5})*6*7*8	○	×	—	79 ページ
40		系統別コモン変数先頭デバイス番号	—	ワードバイス(D, W, #, U3E□¥G ^{*5} , U3E□¥HG ^{*5})*6*7*8	○	×	—	79 ページ
41		ローカル変数先頭デバイス番号	—	ワードバイス(D, W, #, U3E□¥G ^{*5} , U3E□¥HG ^{*5})*6*7*8	○	×	—	79 ページ

*1 MT Developer2で直接設定する場合は、指数形式を小数点形式に置き換えて設定してください。

*2 間接設定に使用するデバイスの範囲は、下記を参照してください。

📖 MELSEC iQ-R モーションコントローラプログラミングマニュアル(共通編)

*3 本設定は省略可能です。

*4 系統共通コモン変数点数と系統別コモン変数点数を合わせて、合計900点まで設定できます。

*5 自号機のデバイスのみ設定可能です。他号機のデバイスを設定すると、中度エラー (エラーコード: 2221H)となります。

*6 デバイス番号は、偶数で設定してください。

*7 使用可能なデバイスの設定範囲は、下記を参照してください。

📖 MELSEC iQ-R モーションコントローラプログラミングマニュアル(共通編)

*8 ワードデバイスの「ユニットアクセスデバイス(U□¥G)」は設定できません。「ユニットアクセスデバイス(U□¥G)」を設定すると、中度エラー (エラーコード: 2221H)となります。

系統基本設定

系統内軸数

系統内で使用するGコード制御軸数を設定します。

Gコード制御軸パラメータでは、本設定で設定した軸数分を軸No.1から順番に設定してください。

例

系統1の系統内軸数が「4」の場合

Gコード制御における系統1に軸No.1, No.2, No.3, No.4と順番に割り当てられます。Gコード制御軸パラメータで、軸No.1~軸No.4まで使用されるため、軸No.1~軸No.4の設定を行う必要があります。

モーダル初期設定

補間モード

マルチCPUシステムの電源投入時、およびリセット時の直線指令のモードを選択します。

平面選択

マルチCPUシステムの電源投入時、およびリセット時の平面を選択します。

「0」を設定した場合、「1: X-Y平面(G17指令状態)」が設定されたものとみなします。

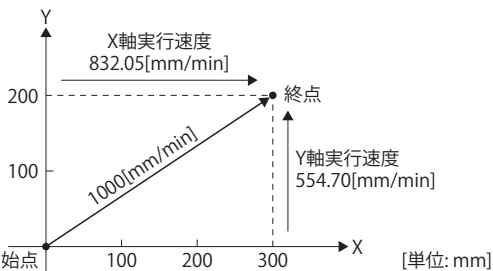
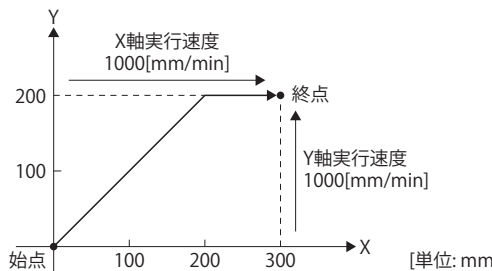
絶対値設定

マルチCPUシステムの電源投入時、およびリセット時の絶対設定/増分設定のモードを選択します。

制御設定

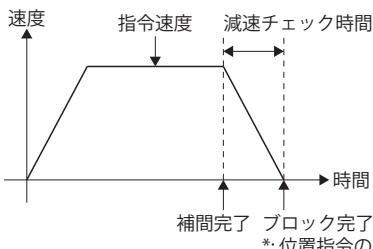
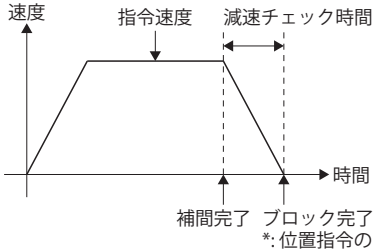
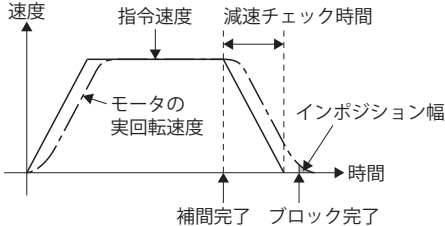
G00非補間

G00の動作経路のタイプを選択します。

0: 終点に向け、直線で移動する。(補間タイプ)	1: 各軸の終点に向け、各軸の早送り速度で移動する。(非補間タイプ)
<ul style="list-style-type: none">補間速度として扱います。位置決めにおける工具の移動経路は、始点と終点を結ぶ最短経路になります。位置決め速度は指令された各軸の速度がその早送り速度を越えない範囲で最短の分配時間が得られるように自動的に計算されます。 <p>(例) X軸とY軸の早送り速度がともに1000[mm/min]のとき G91 G00 X300. Y200. ,F1000</p> 	<ul style="list-style-type: none">軸ごとの指令速度として扱います。位置決めにおける工具の移動経路は、始点から終点まで各軸の早送り速度で移動します。 <p>(例) X軸とY軸の早送り速度がともに1000[mm/min]のとき G91 G00 X300. Y200. ,F1000</p> 

減速チェック

実行ブロックが早送り指令(G00)と切削指令(G01, G02, G03)の減速チェック方式を選択します。減速チェック方式には、3方式があり、早送り指令(G00)と切削指令(G01, G02, G03)とでそれぞれ下記の値を設定します。

設定値	減速チェック方式	内容
0	指令減速チェック方式	<p>補間完了後に減速チェック時間*1が経過したとき、ブロック完了とします。</p>  <p>補間完了 ブロック完了 *: 位置指令の目標位置到達をチェックしない。</p>
1	スムージングチェック方式	<p>補間完了後に減速チェック時間*1が経過し、系統内の全軸が下記の条件を満たしたとき、ブロック完了とします。</p> <ul style="list-style-type: none">位置指令が目標位置に到達したとき。  <p>補間完了 ブロック完了 *: 位置指令の目標位置到達をチェックする。</p>
2	インポジションチェック方式	<p>補間完了後に減速チェック時間*1が経過し、系統内の全軸が下記の条件を満たしたとき、ブロック完了とします。</p> <ul style="list-style-type: none">位置指令が目標位置に到達したとき。モータの実位置がインポジション幅内に到達したとき。  <p>補間完了 ブロック完了</p>

*1 減速チェック時間は加減速モードと加減速時定数により自動的に算出されます。

G1→G1減速チェック

G1→G1の移動方向反転時、減速チェックを実施するか、実施しないかを選択します。

G1→G1減速チェック	内容
0: 減速チェック実施しない	補間が完了した後に減速チェックを行わず、次ブロックの移動を開始します。
1: 減速チェック実施する	補間が完了した後に減速チェックを行ってから、次ブロックの移動を開始します。

円弧誤差

円弧指令で、終点と中心座標のずれがあったときの許容誤差範囲を設定します。

円弧中心誤差修正許容値

R指定円弧補間の中心座標値の計算誤差補正許容値を設定します。「始点と終点を結ぶ線分」と「指令半径×2」との誤差が許容値以下の場合は始点と終点を結ぶ線分の中点が円弧中心となるように修正します。

- ・ 設定値<0の場合: 0(中心誤差を補正しない)
- ・ 設定値=0の場合: 2×最小設定単位(0.0001)
- ・ 設定値>0の場合: 設定した値

オーバライド設定

自動コーナオーバライド

自動コーナオーバライドのオーバライド値を設定します。

「0[%]」、または「100[%]」を設定すると自動コーナオーバライドは「無効」となります。

自動コーナオーバライド最大角度

自動コーナオーバライドで自動的に減速するコーナの最大開き角を設定します。

「0[degree]」、または「180[degree]」を設定すると自動コーナオーバライドは「無効」となります。

自動コーナオーバライドコーナ前長さ

自動コーナオーバライドのコーナで減速を開始する位置を設定します。

コーナの手前、どれだけの長さの点から減速するかを設定します。

早送りオーバライド

早送りオーバライドの比率を設定するデバイスを設定します。

早送り速度に対して、1[%]単位でオーバライドをかけることができます。

指定したデバイスには、以下の値を設定します。

設定範囲
0～100[%]

切削送りオーバライド

切削送りオーバライドの比率を設定するデバイスを設定します。

Gコードプログラムに設定した送り速度指令に対して、1[%]単位でオーバライドをかけることができます。

指定したデバイスには、以下の値を設定します。

設定範囲
0～327[%]

オーバライドキャンセル

オーバライドをキャンセルするためのデバイスを設定します。

設定したデバイスのOFF→ONで、自動運転における切削送りオーバライドに対して、オーバライドが「100[%]」となります。自動コーナオーバライド、早送りオーバライドに対しては「無効」となります。

平面構成

基本軸I/基本軸J/基本軸K

平面を構成する基本軸の軸名称を設定します。

[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“系統軸情報”⇒“軸名称”にて設定した同じ軸名称を設定してください。

2軸で使用する場合など、使用しない項目(基本軸I, 基本軸J, 基本軸K)には「0」を設定してください。

通常、「基本軸I」、「基本軸J」、「基本軸K」にそれぞれX, Y, Zを設定することで下記の関係が成立しますが、下記以外の軸名称を設定する場合は任意の軸名称を設定してください。

- G17: X-Y
- G18: Z-X
- G19: Y-Z

「基本軸I」、「基本軸J」、「基本軸K」に設定した軸名称が重複している場合、または設定した軸名称の軸が[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“系統軸情報”⇒“軸名称”にて設定されていない場合、中度エラー (エラーコード: 30FDH(詳細コード: 0010H))となります。

法線制御

法線制御軸

法線制御を行う軸の軸名称を設定します。

指定した軸名称の軸が設定されていない場合、中度エラー (エラーコード: 30FDH(詳細コード: 0011H))となります。

法線制御軸は、回転軸の軸名称を設定してください。直線軸の軸名称を設定した場合、中度エラー (エラーコード: 30FDH(詳細コード: 0012H))となります。

「基本軸I」、「基本軸J」、「基本軸K」で設定した軸名称の軸は、法線制御軸には設定できません。法線制御軸に設定した場合、中度エラー (エラーコード: 30FDH(詳細コード: 0012H))となります。

法線制御タイプ

法線制御のタイプを設定します。

法線制御タイプ	内容		
	旋回方向	ブロック継ぎ目の旋回速度	円弧補間時の旋回速度
0: 法線制御タイプI	180[degree]以下になる方向(近回り方向)	パラメータの設定速度	プログラム経路が指令速度 に従った速度
1: 法線制御タイプII	■開始時 180[degree]以下になる方向(近回り方向) ■開始時以外 指令方向	工具刃先が指令速度に従った速度	

旋回最小角

法線制御中、ブロックの継ぎ目での法線制御軸旋回最小角度を設定します。

法線制御軸旋回速度

法線制御中、ブロック継ぎ目の法線制御軸旋回速度を設定します。

法線制御軸旋回速度は、法線制御タイプIの場合に有効です。

法線制御軸旋回速度で設定した値が、法線制御軸の[Gコード制御軸パラメータ]⇒“速度・時定数”⇒“切削送りクランプ速度”にて設定した値を超える場合は、中度エラー (エラーコード: 30FDH(詳細コード: 0012H))となります。

旋回最小移動量

法線制御中、直前の継ぎ目で旋回動作を行うブロックの最小移動量を設定します。

法線制御軸旋回径

法線制御軸中心から工具先端までの長さを設定します。
ブロック継ぎ目、円弧補間中の旋回速度計算で使用します。
法線制御軸旋回径は、法線制御タイプⅡの場合に有効です。

法線制御軸挿入径

法線制御中、コーナに自動挿入する円弧の半径を設定します。法線制御軸挿入径は、法線制御タイプⅠの場合に有効です。

補助機能

Mバイナリ

補助機能(M機能)をBCD出力するか、バイナリ(符号付き、符号なし)出力するかを設定します。

極座標補間

極座標補間直線軸

極座標補間を行う直線軸の軸名称を設定します。

指定した軸名称の軸が設定されていない場合、中度エラー (エラーコード: 30FDH(詳細コード: 0013H))となります。

極座標補間直線軸は、直線軸の軸名称を設定してください。回転軸の軸名称を設定した場合、中度エラー (エラーコード: 30FDH(詳細コード: 0014H))となります。

「基本軸I」、「基本軸J」、「基本軸K」で設定した軸名称を設定してください。設定した軸名称が、「基本軸I」、「基本軸J」、「基本軸K」に設定されていない場合、中度エラー (エラーコード: 30FDH(詳細コード: 0014H))となります。

極座標補間直線軸と極座標補間回転軸は、合わせて設定してください。どちらかのみ設定した場合、中度エラー (エラーコード: 30FDH(詳細コード: 0013H))となります。

極座標補間回転軸

極座標補間を行う回転軸の軸名称を設定します。

指定した軸名称の軸が設定されていない場合は、中度エラー (エラーコード: 30FDH(詳細コード: 0013H))となります。

極座標補間回転軸は、回転軸の軸名称を設定してください。直線軸の軸名称を設定した場合、中度エラー (エラーコード: 30FDH(詳細コード: 0014H))となります。

下記で設定されていない軸名称を設定してください。設定した軸名称が、下記に設定されている場合、中度エラー (エラーコード: 30FDH(詳細コード: 0014H))となります。

- 基本軸I, 基本軸J, 基本軸K
- 法線制御軸

極座標補間直線軸と極座標補間回転軸は、合わせて設定してください。どちらかのみ設定した場合、中度エラー (エラーコード: 30FDH(詳細コード: 0013H))となります。

高精度制御

補間前加減速-最高速度

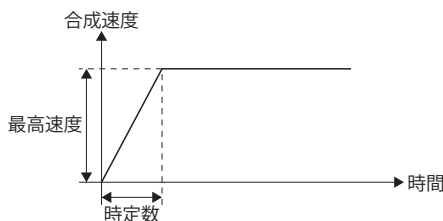
補間前加減速時の切削送り速度を設定します。

Gコード制御軸パラメータの「高精度制御モード用切削送りクランプ速度」、またはGコード制御系統パラメータの「補間前加減速-最高速度」が、Gコード制御軸パラメータの「切削送りクランプ速度」より大きい値を設定した場合、中度エラー（エラーコード: 30FDH(詳細コード: 0015H)）となります。

補間前加減速-時定数

補間前加減速時の切削送り時定数を設定します。

合成速度波形と「補間前加減速-最高速度」、「補間前加減速-時定数」の関係は、下図のようになります。



各軸許容加速度制御有効

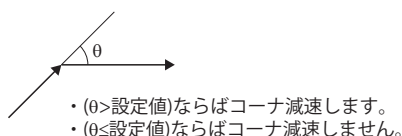
高精度制御有効時のブロック間のコーナ減速速度の計算方法を選択します。

設定値	内容
0: 最適コーナ減速	Gコード制御系統パラメータの「補間前加減速-最高速度」と「補間前加減速-時定数」から決まる全軸共通の許容加速度を用いて減速速度を計算します。
1: 各軸許容加速度制御	Gコード制御軸パラメータの「各軸補間前切削送り-最高速度」と「各軸補間前切削送り-時定数」から決まる各軸の許容加速度を用いて減速速度を計算します。

コーナ減速角度

コーナとみなすべき角度(外角)の最小値を設定します。

高精度制御モード中のブロック間角度(外角)が設定値より大きい場合、コーナであると判定し、エッジを出すために減速します。



コーナ精度係数

高精度制御モード中のコーナでの丸みをさらに小さくする場合、または大きくする場合の補正係数を設定します。設定値が大きいほど理論上の軌跡誤差は小さくなりますが、コーナ減速速度が低くなるためサイクルタイムが増加します。

曲線精度係数

高精度制御モード中の曲線(円弧)における半径減少量をさらに小さくする場合、または大きくする場合の補正係数を設定します。設定値が大きいほど理論上の軌跡誤差は小さくなりますが、円弧クランプ速度が低くなるためサイクルタイムが増加します。

円弧減速速度切換

高精度制御モード中、円弧入口と出口の進入時に減速するか、減速しないを選択します。

円弧減速速度

円弧入口と出口の進入時の減速速度を設定します。

マクロ制御

系統共通コモン変数点数

系統共通コモン変数の点数を設定します。「0」を設定した場合、系統共通コモン変数は使用できません。
系統共通コモン変数点数と系統別コモン変数点数を合わせて、合計900点まで設定できます。

系統別コモン変数点数

系統別コモン変数の点数を設定します。「0」を設定した場合、系統別コモン変数は使用できません。
系統共通コモン変数点数と系統別コモン変数点数を合わせて、合計900点まで設定できます。

系統共通コモン変数先頭デバイス番号

系統共通コモン変数の先頭デバイス番号を設定します。デバイスは、系統共通コモン変数1点に対して4ワード単位で割り当てられ、データ型は64ビット浮動小数点型となります。

デバイス番号は、偶数で設定してください。

系統共通コモン変数点数が「0」の場合、本設定は省略可能です。

使用可能なワードデバイスの設定範囲は、下記を参照してください。

📖 MELSEC iQ-R モーションコントローラプログラミングマニュアル(共通編)

系統別コモン変数先頭デバイス番号

系統別コモン変数の先頭デバイス番号を設定します。デバイスは、系統別コモン変数1点に対して4ワード単位で割り当てられ、データ型は64ビット浮動小数点型となります。

デバイス番号は、偶数で設定してください。

系統別コモン変数点数が「0」の場合、本設定は省略可能です。

使用可能なワードデバイスの設定範囲は、下記を参照してください。

📖 MELSEC iQ-R モーションコントローラプログラミングマニュアル(共通編)

ローカル変数先頭デバイス番号

ローカル変数の先頭デバイス番号を設定します。デバイスは、ローカル変数1点に対して4ワード単位で割り当てられ、データ型は64ビット浮動小数点型となります。

デバイス番号は、偶数で設定してください。

ローカル変数は、マクロ呼出しの深さ(0~4重)ごとに#1~#33までの33点を使用するため、系統ごとに165点使用します。

そのため、ローカル変数を使用する場合、1点に対して4ワード単位で割り当てられるため、系統ごとにワードデバイスを660ワード固定で使用します。

Gコード制御プログラム中にローカル変数を使用しない場合、本設定は省略可能です。


ローカル変数の詳細は、ローカル変数を参照してください。(📖 182ページ ローカル変数)

使用可能なワードデバイスの設定範囲は、下記を参照してください。

📖 MELSEC iQ-R モーションコントローラプログラミングマニュアル(共通編)

4.2 Gコード制御軸パラメータ

Gコード制御の系統ごとの各軸に軸の情報などのパラメータを設定します。

 [モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]

No.	項目		初期値	設定範囲	直接設定*1	間接設定		詳細説明項
					可否	可否(使用サイズ)	取込周期	
1	系統軸情報	軸No.	1	0: 設定なし 1～64:軸No.	○	×	—	 81 ページ
2		軸名称	0	0: 設定なし 1: X 2: Y 3: Z 4: A 5: B 6: C 7: U 8: V 9: W	○	×	—	 81 ページ
3		回転軸*2	0	0: 直線軸 1: 回転軸	○	×	—	 81 ページ
4	ストアードストロークリミット	ソフトリミット-	$-99999999(\times 10^{-4}[\text{mm}])$	■直線軸設定時 $-99999999\sim 99999999(\times 10^{-4}[\text{mm}])$ ■回転軸設定時 $-99999999\sim 99999999(\times 10^{-4}[\text{degree}])$	○	×	—	 82 ページ
5		ソフトリミット+	$99999999(\times 10^{-4}[\text{mm}])$		○	×	—	
6	速度・時定数	早送り速度	2000[mm/min]	■直線軸設定時 1～1000000[mm/min] ■回転軸設定時 1～1000000[degree/min]	○	×	—	 83 ページ
7		切削送りクランプ速度	2000[mm/min]	■直線軸設定時 1～1000000[mm/min] ■回転軸設定時 1～1000000[degree/min]	○	×	—	 83 ページ
8		G0時定数(リニア)	1000[ms]	1～4000[ms]	○	×	—	 83 ページ
9		G1時定数(リニア)	1000[ms]	1～4000[ms]	○	×	—	 83 ページ
10	回転軸情報	回転軸タイプ*2	0	0: 近回り無効 1: 近回り有効 3: 全座標直線型	○	×	—	 84 ページ
11	タンデム機能	マスタ制御軸名称*2	0	0: 設定なし 1: X 2: Y 3: Z 4: A 5: B 6: C 7: U 8: V 9: W	○	×	—	 85 ページ

No.	項目		初期値	設定範囲	直接設定 ^{*1}	間接設定		詳細説明項
					可否	可否(使用サイズ)	取込周期	
12	高精度制御	高精度制御モード用早送り速度 ^{*2}	0[mm/min]	■直線軸設定時 0~1000000[mm/min] ■回転軸設定時 0~1000000[degree/min]	○	×	—	86ページ
13		高精度制御モード用切削送りクランプ速度 ^{*2}	0[mm/min]	■直線軸設定時 0~999999[mm/min] ■回転軸設定時 0~999999[degree/min]	○	×	—	86ページ
14		各軸補間前切削送り-最高速度 ^{*2}	0[mm/min]	■直線軸設定時 0~999999[mm/min] ■回転軸設定時 0~999999[degree/min]	○	×	—	86ページ
15		各軸補間前切削送り-時定数 ^{*2}	0[ms]	0~5000[ms]	○	×	—	86ページ
16		各軸精度係数 ^{*2}	0[%]	-1000~99[%]	○	×	—	86ページ

*1 MT Developer2で直接設定する場合は、指数形式を小数点形式に置き換えて設定してください。

*2 本設定は省略可能です。

系統軸情報

軸No.

Gコード制御で制御する軸No.を設定します。

サーボネットワーク設定にて設定した軸No.を設定してください。また、設定する軸No.は重複して設定できません。

サーボネットワーク設定にて設定されていない軸No., または軸No.を重複して設定した場合、中度エラー(エラーコード: 30FDH(詳細コード: 0020H))となります。

軸名称

各軸の軸名称をアルファベット(X, Y, Z, A, B, C, U, V, W)で設定します。

同一系統内では、同じ軸名称を設定できません。同一系統内で、同じ軸名称を設定した場合、中度エラー(エラーコード: 30FDH(詳細コード: 0021H))となります。ただし、別の系統で設定されている軸名称と同じ軸名称は設定できます。

回転軸

制御する軸が直線軸か、回転軸かを設定します。

回転軸を指定した場合、軸は回転座標系で制御します。

回転軸の種類は、「回転軸タイプ」で設定します。(84ページ 回転軸タイプ)

設定した回転軸により、軸No.で設定した軸の単位を[モーション制御パラメータ]⇒[軸設定パラメータ]⇒「固定パラメータ」⇒「単位設定」にて下記の単位に設定してください。下記以外を設定した場合、中度エラー(エラーコード: 30FDH(詳細コード: 0022H))となります。

回転軸	単位設定
0: 直線軸	mm, pulse
1: 回転軸	degree, pulse

ストアードストロークリミット

ソフトリミット-／ソフトリミット+

ストアードストロークリミットの移動範囲の-方向(下限値), +方向(上限値)を設定します。

下記の場合, 中度エラー (エラーコード: 30FDH(詳細コード: 0024H))となります。

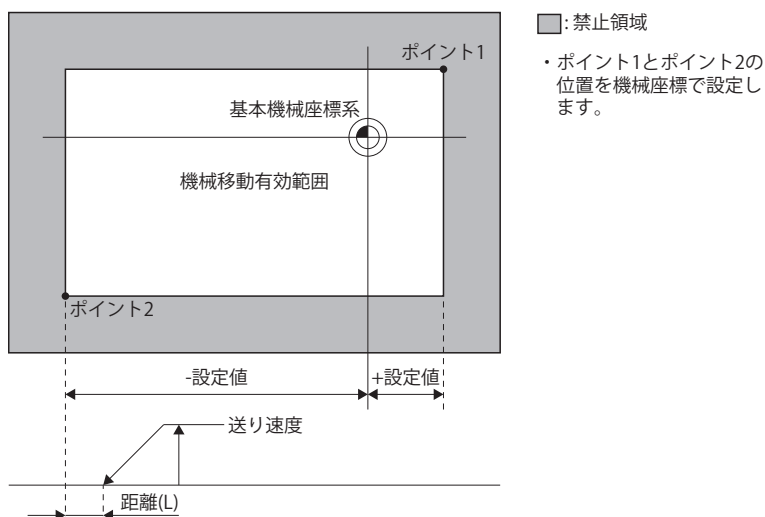
回転軸	回転軸タイプ	ストアードストロークリミット有効／無効	エラーとなる条件
0: 直線軸	—	有効	$(\text{ソフトリミット-}) \geq (\text{ソフトリミット+})$
1: 回転軸	0: 近回り無効	無効	$(\text{ソフトリミット-}) \neq (\text{ソフトリミット+})$
	1: 近回り有効	無効	$(\text{ソフトリミット-}) \neq (\text{ソフトリミット+})$
	3: 全座標直線型	有効	$(\text{ソフトリミット-}) \geq (\text{ソフトリミット+})$

ストアードストロークリミットには下記の種類があります。

■ストアードストロークリミットI

設定した境界の外側を進入禁止領域とします。

制御装置の電源を入れてすぐには有効とならず, Gコード制御移行後から有効となります。なお, Gコード制御中, 固定パラメータのストロークリミット設定は無効となります。Gコード制御移行時, 機械移動有効範囲外の軸があると, Gコード制御に移行せず, 軽度エラー (エラーコード: 1FC1H(詳細コード: 0101H))となります。



自動運転においては, 禁止領域に進入しないように, 軽度エラー (エラーコード: 1FC2H(詳細コード: 0221H))となります。1軸でもエラーが発生すると, 全軸減速停止となります。停止位置は必ず禁止領域の手前となります。禁止領域と停止位置の距離(L)は送り速度により異なります。

Point

Gコード制御への移行前に, 現在の機械位置に問題ないか, “[Md.20]送り現在値(R: D32000+48n, D32001+48n/Q: D0+20n, D1+20n)”にて確認してください。軸の単位設定が[degree]の場合, “[Md.20]送り現在値(R: D32000+48n, D32001+48n/Q: D0+20n, D1+20n)”は回転型の範囲(0~359.99999[degree])となりますが, 機械位置を直線型の範囲(-2147483648~2147483647)で確認する場合は, 任意データモニタ設定のデータ種別で「累積現在値」を設定してください。設定した任意の格納デバイスで確認できます。

速度・時定数

早送り速度

軸ごとの早送り速度を設定します。詳細は、早送り速度を参照してください。(📖 206ページ 早送り速度)

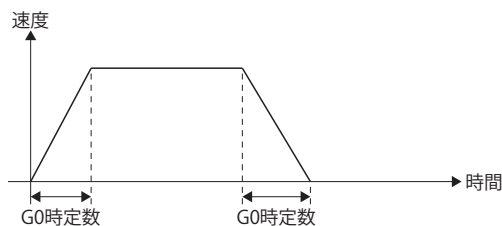
切削送りクランプ速度

軸ごとの切削送り最高速度を設定します。

G01での送り速度が切削送りクランプ速度の値を超えて指令されても、切削送りクランプ速度で設定した速度でクランプします。詳細は、切削送り速度を参照してください。(📖 207ページ 切削送り速度)

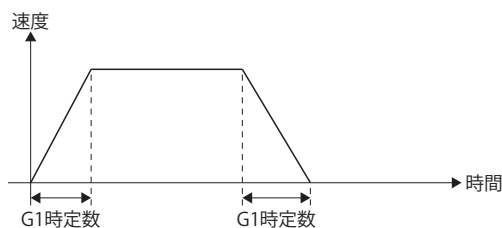
G0時定数(リニア)

早送り加減速における直線制御の時定数を設定します。詳細は、早送り速度を参照してください。(📖 206ページ 早送り速度)



G1時定数(リニア)

切削送り加減速における直線制御の時定数を設定します。詳細は、切削送り速度を参照してください。(📖 207ページ 切削送り速度)



回転軸情報

回転軸タイプ

回転軸のタイプが回転型(近回り有効/無効)か、直線型かを設定します。

回転軸にて「1: 回転軸」を設定した場合に有効となります。

設定する回転軸タイプにより、移動の方法が異なります。

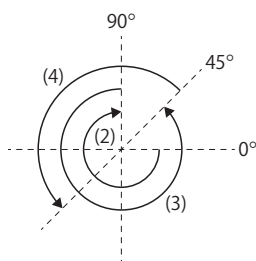
■回転型(0: 近回り無効)

- 機械位置, ワーク座標位置, 相対位置のいずれも, 0~359.9999[degree]の範囲となります。
- 絶対値指令の場合は, 360.0000[degree] で割った余りだけ符号に従って移動します。

例

「0: 近回り無効」にて下記Gコードプログラムを実行した場合

動作	プログラム	[Md.3149]相対位置(D54772+32sn, D54773+32sn)[degree]	[Md.3147]機械位置(D54768+32sn, D54769+32sn)[degree]
(1)	G90 C0.	0.0000	0.0000
(2)	G90 C-270.	90.0000	90.0000
(3)	C405.	45.0000	45.0000
(4)	G91 C180.	225.000	225.000



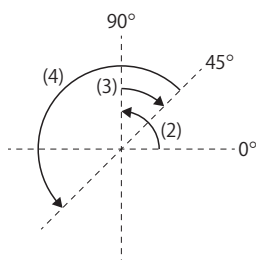
■回転型(1: 近回り有効)

- 機械位置, ワーク座標位置, 相対位置のいずれも, 0~359.9999[degree]の範囲となります。
- 絶対値指令の場合は, 終点までの移動量が少ない方向に移動します。ただし, 正負の移動量が同じ(180.0000[degree]移動する)場合は, 正方向に移動します。

例

「1: 近回り有効」にて下記Gコードプログラムを実行した場合

動作	プログラム	[Md.3149]相対位置(D54772+32sn, D54773+32sn)[degree]	[Md.3147]機械位置(D54768+32sn, D54769+32sn)[degree]
(1)	G90 C0.	0.0000	0.0000
(2)	G90 C-270.	90.0000	90.0000
(3)	C405.	45.0000	45.0000
(4)	G91 C180.	225.000	225.000



■直線型(3: 全座標位置直線型)

- 全座標位置直線型は、-9999.9999～9999.9999[degree]の範囲となります。
- 直線軸と同様の動作をします。
- Gコード制御をいったん終了後、再度Gコード制御に移行した場合、機械座標位置をもとにワーク座標位置が設定されます。

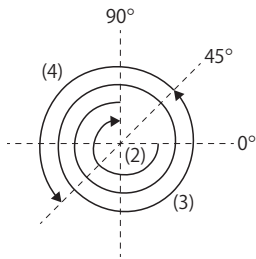
例

「3: 全座標位置直線型」にて下記Gコードプログラムを実行した場合

動作	プログラム	[Md.3149]相対位置(D54772+32sn, D54773+32sn)[degree]	[Md.3147]機械位置(D54768+32sn, D54769+32sn)[degree]
(1)	G90 C0.	0.0000	0.0000
(2)	G90 C-270.	-270.0000	-270.0000
(3)	C405.	405.0000	405.0000
(4)	G91 C180.	585.0000	585.0000

- Gコード制御を終了し、再度Gコード制御に移行した座標位置

ワーク[degree]	機械位置[degree]
585.0000	585.0000



タンデム機能

マスタ制御軸名称

タンデム機能を使用するためのマスタ制御軸名称を設定します。

以下の場合、中度エラー (エラーコード: 30FDH(詳細コード: 0023H))となります。

- Gコード制御システムパラメータの下記に設定されている軸で本設定を使用したとき。
 - 基本軸I, 基本軸J, 基本軸K
 - 法線制御軸
 - 極座標補間直線軸, 極座標補間回転軸
- Gコード制御軸パラメータの「軸名称」が「U」, 「V」, 「W」以外の軸で本設定を使用したとき。
- 指定した軸名称の軸が設定されていないとき。
- 指定した軸名称の軸がスレーブ制御軸に設定されているとき。
- 指定した軸名称の軸と下記パラメータが一致していないとき。
 - Gコード制御軸パラメータの「回転軸」
 - Gコード制御軸パラメータの「回転軸タイプ」

高精度制御

高精度制御モード用早送り速度

高精度制御モード中の軸ごとの早送り速度を設定します。

「0」の場合は、Gコード制御軸パラメータの「早送り速度」の速度を使用します。

高精度制御モード用切削送りクランプ速度

高精度制御モード中の軸ごとの切削送り最高速度を設定します。

「0」の場合は、Gコード制御軸パラメータの「切削送りクランプ速度」の速度を使用します。

各軸補間前切削送り-最高速度

各軸許容加速度制御有効時、各軸の許容加速度を計算するための最高速度を設定します。

「0」の場合は、Gコード制御軸パラメータの「早送り速度」の速度を使用します。

各軸補間前切削送り-時定数

各軸許容加速度制御有効時、各軸の許容加速度を計算するための時定数(最高速度に到達するまでの時間)を設定します。

「0」の場合は、Gコード制御軸パラメータの「G0時定数(リニア)」の時定数を使用します。

各軸精度係数

各軸許容加速度制御有効時、高精度制御モード中のコーナ部の軌跡誤差やクランプ速度を各軸で調整するための補正係数を設定します。

設定値が大きいほどエッジ精度は向上しますが、コーナでの速度が低くなるためサイクルタイムが増加します。

4.3 Gコード制御加工パラメータ

Gコード制御で加工を行うために使用する工具のパラメータを設定します。

☞ [モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御加工パラメータ]

No.	項目		初期値	設定範囲	直接設定 ^{*1}	間接設定		詳細説明項
					可否	可否(使用サイズ)	取込周期	
1	工具径補正	工具径補正タイプ ^{*2}	0	0: タイプA 1: タイプB	○	×	—	87 ページ
2		干渉チェック ^{*2}	0	0: 干渉チェックアラーム 1: 干渉チェック無効 2: 干渉チェック回避	○	×	—	87 ページ
3		径補正直径指定有効 ^{*2}	0	0: 半径補正量 1: 直径補正量	○	×	—	87 ページ
4	工具補正データ	径補正量 ^{*2}	0(×10 ⁻⁴ [mm])	-99999999~99999999(×10 ⁻⁴ [mm])	○	×	—	88 ページ
5		長補正量 ^{*2}	0(×10 ⁻⁴ [mm])	-99999999~99999999(×10 ⁻⁴ [mm])	○	×	—	88 ページ
6	ワーク座標オフセット	座標オフセット ^{*2}	X	■直線軸設定時 -99999999~99999999(×10 ⁻⁴ [mm]) ■回転軸設定時 -99999999~99999999(×10 ⁻⁴ [degree])	○	×	—	88 ページ
7			Y		○	×	—	
8			Z		○	×	—	
9			A		○	×	—	
10			B		○	×	—	
11			C		○	×	—	
12			U		○	×	—	
13			V		○	×	—	
14			W		○	×	—	
15	プログラム座標回転	座標回転タイプ ^{*2}	0	0: 座標回転タイプ0 1: 座標回転タイプ1	○	×	—	88 ページ

*1 MT Developer2で直接設定する場合は、指数形式を小数点形式に置き換えて設定してください。

*2 本設定は省略可能です。

工具径補正

工具径補正タイプ

工具径補正におけるスタートアップ、キャンセル指令動作時の交点演算処理方法を設定します。

工具径補正タイプ	内容
0: タイプA	スタートアップ、キャンセル指令ブロックを交点演算処理の対象とせず、指令の直角方向へのオフセットベクトルとします。
1: タイプB	指令ブロックと次指令ブロックとの交点演算処理を行います。

干渉チェック

工具径補正で、工具の径によるワークへの干渉(食い込み)制御を設定します。

干渉チェック	内容
0: 干渉チェックアラーム	干渉すると判定した時点でアラーム停止します。
1: 干渉チェック無効	干渉チェックを行いません。
2: 干渉チェック回避	干渉しないように経路を変更します。

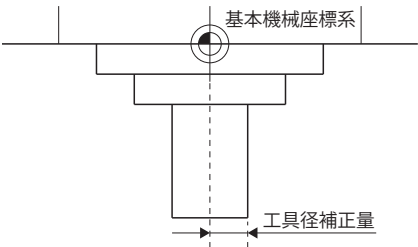
径補正直径指定有効

工具径補正量の指定方法を設定します。

工具補正データ

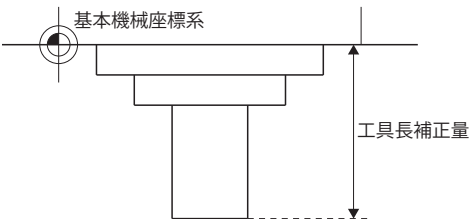
径補正量

工具径に関する形状補正量を設定します。
系統間共通で最大40組設定できます。



長補正量

工具長に関する形状補正量を設定します。
系統間共通で最大40組設定できます。



ワーク座標オフセット

座標オフセット(X~W)

加工するワークの基準点を原点とするために、機械座標系からのオフセット量を設定します。
各軸名称に対応するオフセット量を、ワーク座標系1~6ごとにそれぞれ設定します。

プログラム座標回転

座標回転タイプ

プログラム座標回転指令後、初回移動指令での始点を選択します。

座標回転タイプ	内容
0: 座標回転タイプ0	始点は座標回転に伴って回転せず、回転前のローカル座標系上の現在の位置から終点位置を計算します。
1: 座標回転タイプ1	座標回転に伴って始点が回転すると仮定して終点位置を計算します。

5 Gコード制御用プログラム

5.1 Gコード制御用プログラムの構成

Gコードプログラムのファイルは、テキスト形式で作成します。1つのファイルに対して1つのGコードプログラムを作成します。1ファイルに複数のGコードプログラムを作成することはできません。

Gコードプログラムのファイルの格納先については下記を参照してください。

📖 MELSEC iQ-R モーションコントローラプログラミングマニュアル(共通編)

Gコードプログラムの作成、およびモーションCPUへの書込みは、以下の方法で行います。

MT Developer2によるGコードプログラムの操作

パソコン上で作成したテキストファイルをMT Developer2を使用し、モーションCPUへの書込み/読出し/照合/削除を行います。操作方法の詳細は、下記を参照してください。

📖 MT Developer2のヘルプ

GOTによるGコードプログラムの操作

GOTからGコードプログラムを作成してモーションCPUへの書込み/読出しを行います。

詳細は、GOTによるGコードプログラム操作を参照してください。(📖 267ページ GOTによるGコードプログラムの操作)

FTPサーバ機能経由ファイル転送にてモーションCPUへ操作

パソコン上で作成したテキストファイルをFTPサーバ機能経由ファイル転送にてモーションCPUへの書込み/読出しを行います。FTPサーバ機能経由ファイル転送の詳細は、下記を参照してください。

📖 MELSEC iQ-R モーションコントローラプログラミングマニュアル(共通編)

Gコードプログラムのフォーマット

Gコードプログラムのファイルフォーマットを下記に示します。

項目	内容
ファイル名	O***.gcd*1
拡張子	.gcd
改行コード	CRLF(0x0D, 0x0A)
文字コード	ASCII コード
ファイルサイズ	最大512kバイト

*1 ***=プログラム番号(001～256)

Gコードプログラム

Gコードプログラムは、機械の一つの動作(シーケンス)を指令する「ブロック」と呼ばれる単位の集まりです。これらのブロックは、工具を実際に動かす順序に記述します。

% ←	(1)
(Proc Prog1) ←	(2)
G91 G64 G00 X50. Y100. Z150.	}
G01 X100. Y200. Z300. F15000.	
G04 P1	
G09 G01 X-100. Y-200. Z-300. F15000.	
G91 C500.	(3)
M02	}
% ←	
	(4)

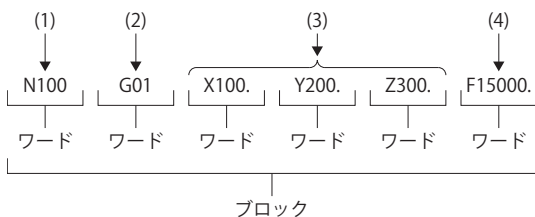
No.	名称	内容
(1)	プログラムスタート	Gコードプログラムの先頭行には、必ずエンドオブレコード(%)を記述します。 エンドオブレコード(%)からプログラムエンドのエンドオブレコード(%)までを1つのプログラムとして読み込みます。
(2)	コメント	ブロックの先頭にコントロールアウト「(」があると、そのブロックは無視されコメントとして扱われます。プログラム名称やコメントなどの情報(ASCIIコード)を記述します。 また、(1)のエンドオブレコード(%)の次行に記述したコメントは、該当プログラム実行中、15文字分(スペースを除く)の情報が「[Md.3070]実行中プログラムコメント(D54588+128s～D54595+128s)」に格納されます。
(3)	プログラムブロック	動作に必要なプログラムブロックが、制御順に複数のブロックで構成されます。 プログラムブロックの終わりを改行により、ブロックの終端として扱われます。
(4)	プログラムエンド	Gコードプログラムの最終行には、必ずエンドオブレコード(%)を記述します。 Gコードプログラムの終わりにエンドオブレコード(%)がないと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0309H))となります。

Point

- プログラムスタート、およびプログラムエンドは、ブロックの先頭に記述してください。ブロックの先頭以外に「%」を記述した場合、プログラムスタート、またはプログラムエンドとして扱われません。
- コントロールアウト「(」は、ブロックの先頭に記述してください。ブロックの先頭以外に「(」を記述した場合、コメントとして扱われないため、プログラム実行時に軽度エラー (エラーコード: 1FC3H(詳細コード: 0306H))となります。
- プログラムブロックの先頭が「アルファベット(A～Z)」、「#」、「[」以外の場合、軽度エラー (エラーコード: 1FC0H(詳細コード: 0042H))となります。ただし、ブロックが改行のみの場合は、そのブロックは無視され、ブロック番号もカウントされません。
- Gコードプログラムに記述するアドレスのアルファベットは、大文字(半角)で記述してください。小文字で記述した場合、軽度エラー (エラーコード: 1FC3H(詳細コード: 0305H))となります。

ブロック

ブロックは一つの作業に対する命令を含む「ワード」と呼ばれる単位の集まりです。機械の特定動作を実行するのに必要な情報を含み、ブロック単位で完全な指令になります。また、改行することで、改行コードをブロックの終端として扱います。



No.	名称	内容
(1)	シーケンス番号	アドレスNとそれに続く数字(最大5桁)で構成されます。 プログラム上で必要なブロックを探す指標(分岐の飛び先など)として使用します。動作には影響しません。
(2)	準備機能(Gコード, G機能)	アドレスGとそれに続く2桁, または3桁(小数点以下1桁を含む場合もある)の数字で構成されます。 Gコードは軸の移動や座標系の設定などの機能を指定します。
(3)	座標語	各軸の座標位置や移動量を指定します。 それぞれの軸を表すアドレスとそれに続く数値情報(正負の符号および数字)で構成されます。 アドレスには, X, Y, Z, A, B, C, U, V, Wなどが用いられます。
(4)	送り機能(F機能)	工作物に対する工具の相対速度を指定します。 アドレスFとそれに続く数字で構成されます。

■注意事項

同一ブロック上に、同一のアドレスを複数記述した場合、最後に記述したアドレスの指令が有効になります。

例

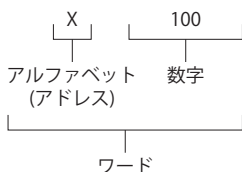
「G01 X10. X1. F1000.」のブロックを記述した場合

アドレスXは、X1.が有効となり、1[mm]になります。

ワード

ワードはある順序に並べられた命令コードや数値, 記号の集まりです。ワードはアドレスと呼ばれるアルファベットと数字(数値情報)で構成されます。数値情報の意味と有効桁数は、アドレスにより異なります。

数字はすべてリーディングゼロ(先行するゼロ)の省略が可能です。



5.2 Gコードプログラムファイルの取込み

Gコードプログラムは、モーションCPU内の標準ROMにGコードプログラムファイルとして格納されます。標準ROM内の格納先については、下記を参照してください。

📖 MELSEC iQ-R モーションコントローラプログラミングマニュアル(共通編)

ファイルの取込みタイミングは、下記のタイミングで取込みます。

- マルチCPUシステムの電源ON時
- マルチCPUシステムのリセット時
- STOP→RUN時

運転中のGコードプログラムファイル取込み

Gコードのプログラム運転中に、モーションCPU内のプログラム運転中取込み用エリアにファイルを転送し、プログラム取込み操作を行うことで、RUN→STOP操作によりモーションCPUの動作を停止しなくてもプログラムを更新(Gコードプログラムファイルを取込み)できます。

運転中のGコードプログラムファイル取込みは、下記の方法により実行します。

- MT Developer2のGコードプログラム操作(🔗 89ページ MT Developer2によるGコードプログラムの操作)
- GOTプログラム入出力/GOTプログラム編集(🔗 267ページ GOTによるGコードプログラムの操作)
- FTPサーバ機能経由ファイル転送によるGコードプログラムの転送(🔗 93ページ FTPサーバ機能経由ファイル転送を使用した運転中のプログラムファイル取込み)

注意事項

- 下記の場合、Gコードプログラムの取込みが失敗します。原因と処置方法について示します。

異常内容と原因	処置方法
該当Gコードプログラムの運転中にプログラム取込み要求を行ったとき。	該当Gコードプログラムが運転中でないことを確認してください。
Gコードプログラムファイルのフォーマットが異常なとき。	Gコードプログラムファイルのフォーマットを見直してください。
Gコードプログラム運転中取込み用エリアのサイズをオーバーしたとき。	いったんGコード制御を終了し、RUN→STOPにして、Gコードプログラムを再度取り込んでください。
指定したプログラムNo.が範囲外のとき。	<ul style="list-style-type: none">• “[Cd.3305]運転中プログラム取込みプログラム番号(D54264)”を確認してください。• 標準ROM内の「\$MMTPRJ\$/gcode/prog/temp」フォルダ内に該当プログラムが格納されていることを確認してください。
STOP中に運転中のプログラム取込み要求を行ったとき。	STOP中は、運転中のプログラム取込み要求を行わないでください。通常プログラムの格納フォルダに格納することで、STOP→RUN時に反映されます。
M98(サブプログラム呼出し)、G65(マクロ呼出し)のブロックを解析中に、サブプログラム、またはマクロプログラムのプログラム取込み要求を行ったとき。	M98(サブプログラム呼出し)、G65(マクロ呼出し)の前ブロックでM00(プログラムストップ)、またはM01(プログラムストップ)を実行し、M98(サブプログラム呼出し)、G65(マクロ呼出し)のブロックの読み込みを停止してください。

- プログラム取込み実行中、該当Gコードプログラムに対してプログラム始動を行った場合、軽度エラー (エラーコード: 1FC3H(詳細コード: 031FH))となります。

FTPサーバ機能経由ファイル転送を使用した運転中のプログラムファイル取込み

FTPサーバ機能経由ファイル転送を使用した運転中のプログラムファイル取込みの手順について説明します。FTPサーバ機能経由ファイル転送の詳細は、下記を参照してください。

📖 MELSEC iQ-R モーションコントローラプログラミングマニュアル(共通編)

操作手順

1. Gコードプログラムファイルの格納フォルダ(\$MMTPRJ\$/gcode/prog/temp)にGコードプログラムファイルを格納する。
2. “[Cd.3305]運転中プログラム取込みプログラム番号(D54264)”に、取込みを行うGコードプログラムのプログラム番号(1～256)を設定する。
3. “[Rq.3344]運転中プログラム取込み要求(D54225.0)”をOFF→ONする。
4. 正常に完了した場合、“[Md.3003]運転中プログラム取込み状態(D54492)”に「2: 正常完了」が格納されます。異常完了した場合は、「3: 異常完了」が格納されます。
5. プログラムファイルを連続で取込む場合は、手順2.～4.を実行する。

■使用デバイス

FTPサーバ機能経由ファイル転送によるGコードプログラムの転送では下記のデバイスを使用します。デバイスの詳細はGコード制御専用信号を参照してください。(📖 27ページ Gコード制御専用信号)

- [Rq.3344]運転中プログラム取込み要求(D54225.0)
- [Cd.3305]運転中プログラム取込みプログラム番号(D54264)
- [Md.3003]運転中プログラム取込み状態(D54492)
- [Md.3004]運転中プログラム取込みエラー情報(D54493)

5.3 先読みバッファ

通常、自動運転時はプログラム解析処理を円滑に行うために1ブロックの先読みを行います。また、工具径補正中は干渉チェックを含む交点計算のために移動指令を3ブロック(移動指令が3ブロックなければ最大5ブロック)の先読みを行います。先読みの内容を下記に示します。

- 先読みは、1ブロックのデータを記憶します。
- コメントは、先読みバッファに読み込まれません。
- 先読みバッファの内容は、“[Rq.3380]リセット指令(D54226.4+2s)”をONするとクリアされます。
- 連続運転中に“[Rq.3379]シングルブロック(D54226.3+2s)”をONしたとき、先読みバッファは次のブロックデータを記憶して停止します。

5.4 小数点入力

工具の軌跡、距離、速度を定義するGコードプログラム入力情報で、[mm]単位の零ポイントを指令する小数点入力が可能です。小数点を使用しない指令の最小桁は、最小入力指令単位となります。

- 小数点指令はGコードプログラムにおける距離、角度、時間、速度の指令に対して有効です。
- 位置指令、速度指令、時間指令の指令単位は、指令入力方法、および四則演算の有/無により下表のようになります。

指令入力方法	四則演算	位置指令	速度指令	時間指令
定数	無効	0.0001[mm]	1.00[mm/min]	0.001[s]
	有効*1	1.0000[mm]	1.00[mm/min]	1.000[s]
変数	—	1.0000[mm]	1.00[mm/min]	1.000[s]

*1 四則演算子を使用する場合は、演算指令を参照してください。(184ページ 演算指令)

- 小数点指令における有効指令値範囲を下記に示します。Gコードプログラムにて有効指令範囲を超えた指令は実行しないでください。

指令	有効指令値範囲
移動指令(直線)	-9999.9999～9999.9999[mm]
移動指令(回転)	-9999.9999～9999.9999[degree]
送り速度	0.01～1000000.00[mm/min]
ドウェル	0～99999.999[s]
R指定円弧の半径/回転角度	-99999.9999～99999.9999[degree]

- 小数点指令はサブプログラム、またはマクロプログラムなどで用いられる変数データの定義指令に対しても有効です。

小数点指令一覧

小数点指令の一覧を下記に示します。小数点無効のアドレスに対する小数点指令は、軽度エラー (エラーコード: 1FC3H(詳細コード: 031AH)) となります。ただし、変数指令では、すべて小数点付きデータとして扱います。小数点無効アドレスに小数点ありの変数を使用した場合は、小数点以下は四捨五入となります。

○: 有効, ×: 無効

アドレス	小数点指令	用途	備考
A	○	座標位置データ	
B	○	座標位置データ	
C	○	座標位置データ	
D	×	工具径補正番号	
F	○	切削送り速度	
	○	早送り速度	,Fと指定
G	○	準備機能コード	
H	×	工具長補正番号	
	×	サブプログラム呼出しシーケンス番号	
I	○	円弧中心	
	○	工具径補正のベクトル成分	
J	○	円弧中心	
	○	工具径補正のベクトル成分	
K	○	円弧中心	
	○	工具径補正のベクトル成分	
L	×	サブプログラム繰り返し回数	
	×	マクロプログラム繰り返し回数	
M	×	補助機能コード	
N	×	シーケンス番号	
O	×	プログラム番号	
P	×	ドウェル時間	
	×	サブプログラム呼出しプログラム番号	
	×	マクロプログラム呼出しプログラム番号	
	×	復帰先シーケンス番号	
R	○	R指定円弧の半径	
	○	回転角度	
U	○	座標位置データ	
V	○	座標位置データ	
W	○	座標位置データ	
X	○	座標位置データ	
	○	ドウェル時間	
Y	○	座標位置データ	
Z	○	座標位置データ	

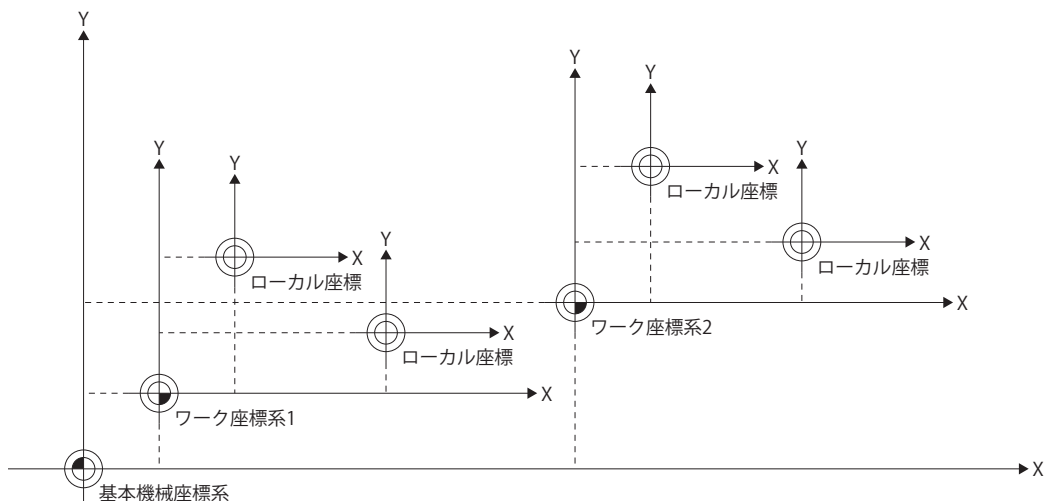
Point

- 小数点指令が有効のアドレスで小数点を使用しない指令の場合、指令データの最下位桁が指令単位に一致します。
(例) 「X1」と指定したときは、「X0.0001」指令と同等になります。
- マクロ呼出しに使用する引数の小数点はすべて有効となります。詳細は、マクロ呼出し(G65)を参照してください。(P147ページ G65: マクロ呼出し)

5.5 座標系について

座標系について説明します。

座標系を表す座標原点シンボルを下記に示します。



座標原点シンボル	名称	内容
	機械座標原点	機械固有に決められた位置
	ワーク座標原点	ワークの加工に使用する座標系の原点
	ローカル座標原点	ワーク座標系を一時的に変更したときの座標系の原点

基本機械座標系

機械固有の座標系で、機械固有に決められた位置(工具交換位置、ストロークエンド位置など)を表す座標系です。

G53指令とこれに続く座標指令により工具を機械座標系上の指令位置に移動させます。

G53指令と同一ブロック内の座標指令のみ、機械座標系への指令となります。

ワーク座標系

Gコードプログラムを作成する際に使用する座標系で、ワーク上の基準点を座標原点として設定する座標系です。ワーク座標系は、基本機械座標系からのオフセット量で位置を指定します。オフセット設定量は、機械座標系原点(0)からの距離で設定します。ワーク座標系は最大6個(ワーク座標1～6)設定できます。Gコード制御加工パラメータの「ワーク座標オフセット」、およびワーク座標系1選択(G54)～ワーク座標系6選択(G59)にて設定します。

ローカル座標系

現在選択中のワーク座標系の中でさらに座標系を指定するための座標系です。これにより、一時的にワーク座標系を変更できます。ローカル座標系は、ワーク座標系1～6(G54～G59)の各ワーク座標系上に独立して指定します。

ローカル座標系の原点位置は、指定したワーク座標系の原点からの距離で指令します。

増分値モードのときは、現在指定されているローカル座標オフセット量に加算した位置がローカル座標原点となります。

ワーク座標系の指定がないときは、現在選択されているワーク座標系上にローカル座標系を作成します。

自動座標系設定

Gコード制御移行時に、あらかじめ設定されているパラメータに従い、基本機械座標系、ワーク座標系が自動的に設定されます。

自動座標系設定により作られる座標系は下記の通りです。

- 基本機械座標系
- ワーク座標系(G54～G59)
- ワーク座標系(G54～G59)の下に作られるローカル座標系(G52)

座標関連のパラメータは、すべて基本機械座標系の原点からの距離で設定します。

自動座標系設定が実行されると、G52指令によるローカル座標系設定はキャンセルされます。

回転軸用座標系

Gコード制御軸パラメータの「回転軸」で「1: 回転軸」を指定した軸は、回転軸の座標系で制御されます。

回転軸の種類には、回転型(近回り有効/無効)と直線型があり、軸ごとに[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒「回転軸情報」⇒「回転軸タイプ」にて設定します。

機械位置、相対位置はパラメータにより異なります。

回転軸は[degree]単位で指令します。

動作範囲を制限する場合は、Gコード制御軸パラメータの「回転軸タイプ」に「3: 全座標直線型」を指定し、ストアードストロークリミットを設定してください。動作範囲を制限しない場合は、Gコード制御軸パラメータの「回転軸タイプ」に「0: 近回り無効」、または「1: 近回り有効」を指定し、ストアードストロークリミットの上限値と下限値を同じ値に設定してください。ストアードストロークリミットの詳細は、ストアードストロークリミットを参照してください。(P82ページストアードストロークリミット)

	回転軸			直線軸
	回転型回転軸		直線型回転軸	
	近回り無効	近回り有効	全座標直線型	
ワーク座標位置	0～359.9999[degree]		-9999.9999～9999.9999 [degree]	-9999.9999～9999.9999[mm]
機械位置/相対位置	0～359.9999[degree]		-9999.9999～9999.9999 [degree]	-9999.9999～9999.9999[mm]
ABS指令	終点から現在位置を引いた増 分量を360[degree]で割った余 りの分だけ、符号に従って移 動します。	終点まで近回りで移動しま す。	通常の直線軸と同様に、終点から現在位置を引いた移動量分だ け360[degree]で丸めず、符号に従って移動します。	
INC指令	現在位置を起点に指令された増分量だけ指令された符号の方向に移動します。			
ストアードストロークリ ミット設定範囲	上限値と下限値を同じ値に設定してください。		-9999.9999～9999.9999 [degree]	-9999.9999～9999.9999[mm]

5.6 Gコード

Gコードはプログラム内の各ブロックでの動作モードを指定する指令です。

Gコード一覧

Gコードプログラムで使用するGコードを下記に示します。

Gコード	グループ	内容	参照
G00 ^{*1}	01	位置決め(早送り)	📖 102ページ G00: 位置決め(早送り)
G01 ^{*1}		直線補間	📖 103ページ G01: 直線補間
G02		円弧補間CW(中心指定)	📖 104ページ G02: 円弧補間CW(中心指定)
		円弧補間CW(R指定)	📖 108ページ G02: 円弧補間CW(R指定)
G03		円弧補間CCW(中心指定)	📖 106ページ G03: 円弧補間CCW(中心指定)
		円弧補間CCW(R指定)	📖 110ページ G03: 円弧補間CCW(R指定)
G04	00	ドウェル(時間指定)	📖 112ページ G04: ドウェル(時間指定)
G09		イグザクトストップチェック	📖 113ページ G09: イグザクトストップチェック
G12.1	21	極座標補間モード開始	📖 114ページ G12.1: 極座標補間モード開始
G13.1 ^{*2}		極座標補間モードキャンセル	📖 118ページ G13.1: 極座標補間モードキャンセル
G17 ^{*1}	02	平面選択 X-Y	📖 119ページ G17～G19: 平面選択
G18 ^{*1}		平面選択 Z-X	
G19 ^{*1}		平面選択 Y-Z	
G38	00	工具径補正ベクトル指定	📖 120ページ G38: 工具径補正ベクトル指定
G39		工具径補正コーナ円弧	📖 122ページ G39: 工具径補正コーナ円弧
G40 ^{*2}	07	工具径補正キャンセル	📖 123ページ G40: 工具径補正キャンセル
G41		工具径補正 左	📖 124ページ G41: 工具径補正 左
G42		工具径補正 右	📖 125ページ G42: 工具径補正 右
G40.1 ^{*2}	15	法線制御 キャンセル	📖 126ページ G40.1: 法線制御 キャンセル
G41.1		法線制御 左オン	📖 127ページ G41.1: 法線制御 左オン
G42.1		法線制御 右オン	📖 128ページ G42.1: 法線制御 右オン
G43	08	工具長補正(+)	📖 129ページ G43: 工具長補正(+)
G44		工具長補正(-)	📖 131ページ G44: 工具長補正(-)
G49 ^{*2}		工具長補正キャンセル	📖 133ページ G49: 工具長補正キャンセル
G52	00	ローカル座標系設定	📖 134ページ G52: ローカル座標系設定
G53		基本機械座標系選択	📖 138ページ G53: 基本機械座標系選択
G54 ^{*2}	12	ワーク座標系1選択	📖 139ページ G54～G59: ワーク座標系1選択～ワーク座標系6選択
G55		ワーク座標系2選択	
G56		ワーク座標系3選択	
G57		ワーク座標系4選択	
G58		ワーク座標系5選択	
G59		ワーク座標系6選択	
G61	13	イグザクトストップチェックモード	📖 141ページ G61: イグザクトストップチェックモード
G61.1		高精度制御モード	📖 142ページ G61.1: 高精度制御モード
G62		自動コーナオーバーライド	📖 143ページ G62: 自動コーナオーバーライド
G64 ^{*2}		切削モード	📖 146ページ G64: 切削モード
G65	00	マクロ呼出し	📖 147ページ G65: マクロ呼出し
G68	16	プログラム座標回転モード開始	📖 153ページ G68: プログラム座標回転モード開始
G69 ^{*2}		プログラム座標回転モードキャンセル	📖 159ページ G69: プログラム座標回転モードキャンセル
G90 ^{*1}	03	絶対値指令	📖 160ページ G90: 絶対値指令
G91 ^{*1}		増分値指令	📖 162ページ G91: 増分値指令
G94 ^{*1}	05	毎分送り(非同期送り)	📖 164ページ G94: 毎分送り(非同期送り)

*1 パラメータよりイニシャル状態で選択するコード、または選択されているコードを示す。

*2 イニシャル状態で選択するコード、または選択されているコードを示す。

- ・同一グループのGコードを2個以上指令した場合は、最後のGコードが有効になります。
- ・Gコード制御要求の解除、またはリセット入力でモーダルは初期化されます。

モーダル／アンモーダル

Gコードには、モーダル指令とアンモーダル指令があります。

分類	グループ	内容
モーダルGコード	01, 02, 03, 05, 07, 08, 12, 13, 15, 16, 21	グループ内のGコードのうち、常に1つを動作モードとして指定する指令です。キャンセル指令、または同一グループのGコードが指令されるまで動作モードを保持します。
アンモーダルGコード	00	指令されたブロックのみ動作モードとなる指令です。次のブロックに対しては無効となります。

Gコードの優先順位

G指令を組合せた場合(同一ブロックに指令した場合、および各モーダル中に指令した場合)の動作について以下に示します。

同一ブロックに指令した場合

○: 両方の指令が同時に実行, △: 最後の指令が有効, ×: 指令不可

グループ	Gコード	グループ									
		01	02	03	07	08	12	13	15	16	21
00	G04	○*1	○	○	×*2	×*2	○*1	○	○*1	○*2	×*3
	G09	○	○	○	○	○	○	○	○	○	×*3
	G38	○*4	×*5	○	○	○	○	○	○	○	×*3
	G39	○	×*5	○	○	○	○	○	○	○	×*3
	G52	○	○	○	×*2	×*2	○	○	○	○*2	×*3
	G53	○*6	○	○	×*2	×*2	○	○	○	○*2	×*3
	G65	○*7	○*7	○*7	○*7	○*7	○*7	○*7	○*7	○*7	○*7
01	G00	△	○	○	○	○	○	○	×*8	○	×*3
	G01	△	○	○	○	○	○	○	○	○	×*3
	G02	△	○	○	×*9	○	○	○	○	○	×*3
	G03	△	○	○	×*9	○	○	○	○	○	×*3
02	G17	○	△	○	○	○	○	○	○	○*10	×*3
	G18	○	△	○	○	○	○	○	○	○*10	×*3
	G19	○	△	○	○	○	○	○	○	○*10	×*3
03	G90	○	○	○	○	○	○	○	○	○	×*3
	G91	○	○	○	○	○	○	○	○	○*11	×*3
05	G94	○	○	○	○	○	○	○	○	○	×*3
07	G40	○*9	○	○	△	○	○	○	○	○	×*12
	G41	○*9	○	○	△	○	○	○	○	○	×*12
	G42	○*9	○	○	△	○	○	○	○	○	×*12
08	G43	○	○	○	○	△	○	○	○	○	×*3
	G44	○	○	○	○	△	○	○	○	○	×*3
	G49	○	○	○	○	△	○	○	○	○	×*3
12	G54	○	○	○	○	○	△	○	○	○	×*3
	G55	○	○	○	○	○	△	○	○	○	×*3
	G56	○	○	○	○	○	△	○	○	○	×*3
	G57	○	○	○	○	○	△	○	○	○	×*3
	G58	○	○	○	○	○	△	○	○	○	×*3
	G59	○	○	○	○	○	△	○	○	○	×*3

グループ	Gコード	グループ									
		01	02	03	07	08	12	13	15	16	21
13	G61	○	○	○	○	○	○	△	○	○	× ^{*3}
	G61.1	○	○	○	○	○	○	△	○	○	× ^{*3}
	G62	○	○	○	○	○	○	△	○	○	× ^{*3}
	G64	○	○	○	○	○	○	△	○	○	× ^{*3}
15	G40.1	○ ^{*13}	○	○	○	○	○	○	△	○	× ^{*12}
	G41.1	○ ^{*13}	○	○	○	○	○	○	△	○	× ^{*12}
	G42.1	○ ^{*13}	○	○	○	○	○	○	△	○	× ^{*12}
16	G68	○	○	○	○	○	○	○	○	△	× ^{*3}
	G69	○	× ^{*14}	○	○	○	○	○	○	△	× ^{*3}
21	G12.1	× ^{*3}	× ^{*3}	× ^{*3}	× ^{*12}	× ^{*3}	× ^{*3}	× ^{*3}	× ^{*3}	× ^{*15}	△
	G13.1	× ^{*3}	× ^{*3}	× ^{*3}	× ^{*12}	× ^{*3}	× ^{*3}	× ^{*3}	× ^{*3}	× ^{*15}	△

*1 モーダルを更新し、G04を実行する。

*2 軽度エラー (エラーコード: 1FC3H(詳細コード: 030FH))となる。

*3 軽度エラー (エラーコード: 1FC3H(詳細コード: 0306H))となる。

*4 円弧とG38を指令した場合は、軽度エラー (エラーコード: 1FC3H(詳細コード: 0306H))となる。

*5 軽度エラー (エラーコード: 1FC3H(詳細コード: 0316H))となる。

*6 円弧は直線として動作する。

*7 モーダルを更新し、G65を実行する。

*8 G00(非補間)では旋回動作しない。

*9 円弧とG40、G41、G42を指令した場合は、軽度エラー (エラーコード: 1FC3H(詳細コード: 0315H))となる。

*10 G69を指令した場合は、軽度エラー (エラーコード: 1FC3H(詳細コード: 0326H))となる。

*11 指令直後の移動指令がG91の場合、軽度エラー (エラーコード: 1FC3H(詳細コード: 0327H))となる。

*12 軽度エラー (エラーコード: 1FC3H(詳細コード: 0306H, または0324H))となる。

*13 G00(非補間)では開始しない。

*14 プログラム座標回転モード中は、軽度エラー (エラーコード: 1FC3H(詳細コード: 0326H))となる。

*15 G68を指令した場合は、軽度エラー (エラーコード: 1FC3H(詳細コード: 0324H)), G69を指令した場合は、軽度エラー (エラーコード: 1FC3H(詳細コード: 0306H))となる。

各モーダル中に指令した場合

○: 両方の指令が同時に実行, △: 最後の指令が有効, ×: 指令不可

グループ	Gコード	モーダル中グループ									
		01	02	03	07	08	12	13	15	16	21
00	G04	○	○	○	○	○	○	○	○	○	○
	G09	○	○	○	○	○	○	○	○	○	○
	G38	○ ^{*1}	× ^{*2}	○	○	○	○	○	○	○	○
	G39	○	× ^{*2}	○	○	○	○	○	○	○	○
	G52	○	○	○	○	○	○	○	× ^{*3}	○	× ^{*4}
	G53	○ ^{*5}	○	○	○	○	○	○	○	○	× ^{*4}
	G65	○	○	○	○	○	○	○	○	○	○
01	G00	○	○	○	○	○	○	○	× ^{*6}	○	○
	G01	○	○	○	○	○	○	○	○	○	○
	G02	○	○	○	○	○ ^{*7}	○	○	○	○	○
	G03	○	○	○	○	○ ^{*7}	○	○	○	○	○
02	G17	○	○	○	× ^{*2}	○	○	○	× ^{*8}	○ ^{*9}	× ^{*4}
	G18	○	○	○	× ^{*2}	○	○	○	× ^{*8}	○ ^{*9}	× ^{*4}
	G19	○	○	○	× ^{*2}	○	○	○	× ^{*8}	○ ^{*9}	× ^{*4}
03	G90	○	○	○	○	○	○	○	○	○	○
	G91	○	○	○	○	○	○	○	○	○ ^{*10}	○
05	G94	○	○	○	○	○	○	○	○	○	○
07	G40	○ ^{*11}	○	○	○	○	○	○	○	○	○
	G41	○ ^{*11}	○	○	○	○	○	○	○	○	○
	G42	○ ^{*11}	○	○	○	○	○	○	○	○	○

グループ	Gコード	モーダル中グループ									
		01	02	03	07	08	12	13	15	16	21
08	G43	○	○	○	○	○	○	○	○	○	× ^{*4}
	G44	○	○	○	○	○	○	○	○	○	× ^{*4}
	G49	○	○	○	○	○	○	○	○	○	× ^{*4}
12	G54	○	○	○	○	○	○	○	× ^{*3}	○	× ^{*4}
	G55	○	○	○	○	○	○	○	× ^{*3}	○	× ^{*4}
	G56	○	○	○	○	○	○	○	× ^{*3}	○	× ^{*4}
	G57	○	○	○	○	○	○	○	× ^{*3}	○	× ^{*4}
	G58	○	○	○	○	○	○	○	× ^{*3}	○	× ^{*4}
	G59	○	○	○	○	○	○	○	× ^{*3}	○	× ^{*4}
13	G61	○	○	○	○	○	○	○	○	○	○
	G61.1	○	○	○	○	○	○	○	○	○	× ^{*4*12}
	G62	○	○	○	○	○	○	○	○	○	× ^{*4}
	G64	○	○	○	○	○	○	○	○	○	○
15	G40.1	○ ^{*6}	○	○	○	○	× ^{*3}	○	○	○	× ^{*4}
	G41.1	○ ^{*6}	○	○	○	○	× ^{*3}	○	○	○	× ^{*4}
	G42.1	○ ^{*6}	○	○	○	○	× ^{*3}	○	○	○	× ^{*4}
16	G68	○	○	○	○	○	○	○	○	○	× ^{*4}
	G69	○	○	○	○	○	○	○	○	○	○
21	G12.1	○	○	○	× ^{*13}	○	○	○ ^{*12*14}	× ^{*13}	× ^{*13}	× ^{*4}
	G13.1	○	○	○	× ^{*13}	○	○	○ ^{*12*14}	× ^{*13}	× ^{*13}	○

*1 円弧とG38は軽度エラー (エラーコード: 1FC3H(詳細コード: 0306H))となる。

*2 軽度エラー (エラーコード: 1FC3H(詳細コード: 0316H))となる。

*3 軽度エラー (エラーコード: 1FC3H(詳細コード: 0304H))となる。

*4 軽度エラー (エラーコード: 1FC3H(詳細コード: 0322H))となる。

*5 円弧は直線として動作する。

*6 G00(非補間)では旋回動作しない。

*7 円弧モーダル中のG49の動作はG01で動作する。

*8 軽度エラー (エラーコード: 1FC3H(詳細コード: 031CH))となる。

*9 G68を指令した場合、軽度エラー (エラーコード: 1FC3H(詳細コード: 0326H))となる。

*10 指令直後の移動指令がG91の場合、軽度エラー (エラーコード: 1FC3H(詳細コード: 0327H))となる。

*11 円弧補間中のG40, G41, G42は軽度エラー (エラーコード: 1FC3H(詳細コード: 0315H))となる。

*12 G61.1の場合、Gコード制御系統パラメータの「各軸許容加速度制御有効」に「1:各軸許容加速度制御」を設定してください。

*13 軽度エラー (エラーコード: 1FC3H(詳細コード: 0324H))となる。

*14 G62中に指令した場合、軽度エラー (エラーコード: 1FC3H(詳細コード: 0324H))となる。

G00: 位置決め(早送り)

現在点を始点とし、指令された終点へ高速位置決めを行います。

コード	フォーマット
G00	$G00 _X _x _Y _y _Z _z _ _ _ ,F _f$ <div style="display: flex; justify-content: flex-end; align-items: center; margin-top: -40px;"> <div style="margin-right: 20px;"> 早送り速度 座標指令 *: □は付加軸 </div> </div>

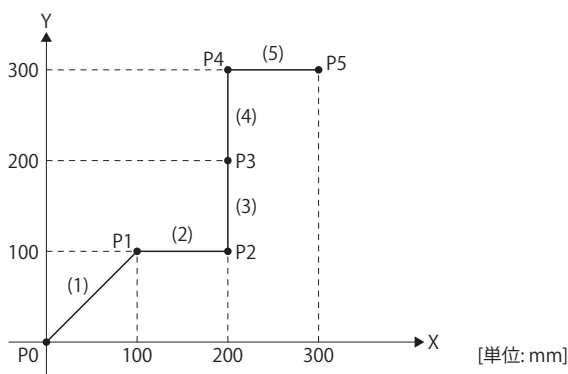
機能

- G00指令はモーダル指令です。同一グループのGコードを使用するまで有効となります。G00指令が連続する場合、次のブロック以降は座標語だけで指令できます。
- 「,F」指令は、指定したブロック内のみ有効です。G00指令と同一ブロックに「,F」指令がある場合は、「,F」指令の速度で位置決めします。「,F」指令がない場合は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“速度・時定数”⇒“早送り速度”にて設定した速度で位置決めを行います。
- 高精度制御モード中は、「,F」指令がない場合、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“高精度制御”⇒“高精度制御モード用早送り速度”にて設定した送り速度で位置決めを行います。ただし、高精度制御モード用早送り速度の設定値が「0」の場合は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“速度・時定数”⇒“早送り速度”にて設定した速度で位置決めを行います。
- 指定した座標軸位置へ移動する経路(直線、または非直線)の設定は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御システムパラメータ]⇒“制御設定”⇒“G00非補間”にて設定します。直線、非直線のどちらも、位置決めの時間は変わりません。
- G00指令では、ブロックの始点、終点で常に加速、減速を行います。終点では減速チェックを行い、各システム内のすべての移動軸の移動完了を確認してから次のブロックへ進みます。
- 減速チェック方式には、指令減速チェック方式、スムージングチェック方式、インポジションチェック方式の3種類があります。早送り/切削送りで使用する減速チェック方式は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御システムパラメータ]⇒“制御設定”⇒“減速チェック”にて設定します。減速チェックについては、減速チェックを参照してください。(P211ページ 減速チェック)

プログラム例

■P0→P1→P2→P3→P4→P5の位置決めを行うプログラム(絶対値指令時)

動作	プログラム	備考
(1)	G00 X100. Y100.	G00で移動
(2)	X200.	
(3)	Y200.	
(4)	G01 Y300. F100.	G01で移動
(5)	X300.	



G01: 直線補間

現在位置より指令された終点へ、指令された速度で直線補間します。
指令される送り速度は、進行方向に対する線速度(合成速度)で指定します。

コード	フォーマット
G01	G01 <u>X x</u> <u>Y y</u> <u>Z z</u> <u>□ □</u> <u>F f</u> <div style="display: flex; justify-content: flex-end; align-items: center;"> <div style="margin-right: 20px;">送り速度</div> <div>座標指令</div> </div> <div style="display: flex; justify-content: flex-end; align-items: center;"> <div>*: □は付加軸</div> </div>

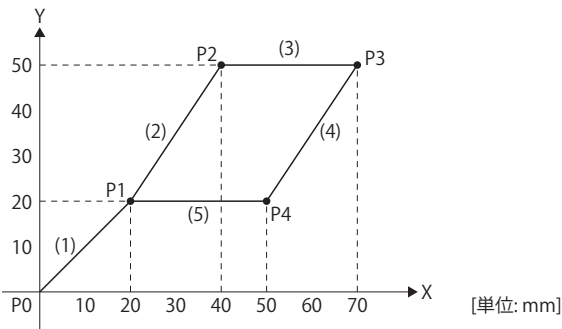
機能

- G01指令はモーダル指令です。同一グループのGコードを使用するまで有効となります。G01指令が連続する場合、次のブロック以降は座標語だけで指令します。
- 最初のG01指令にF指令が設定されていない場合は、軽度エラー (エラーコード: 1FC3H(詳細コード: 0311H))となります。
- 回転軸の送り速度は、[degree/min](小数点位置の単位)で指令します。(F300=300[degree/min])
- G01指令を連続して行う場合、減速チェックは行いません。G01指令において、減速チェックが有効となるのは下記の場合です。
 - 同一ブロックにG09(イグザクトストップチェック)が指令されたとき。
 - G61(イグザクトストップチェックモード)が選択されているとき。
- 減速チェック方式には、指令減速チェック方式、スムージングチェック方式、インポジションチェック方式の3種類があります。早送り/切削送りで使用する減速チェック方式は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御システムパラメータ]⇒“制御設定”⇒“減速チェック”にて設定します。減速チェックについては、減速チェックを参照してください。(211ページ 減速チェック)

プログラム例

■P0→P1→P2→P3→P4→P1の位置決めを行うプログラム(増分値指令時)

動作	プログラム	備考
(1)	G91 G00 X20. Y20.	G00で移動(工具の位置決め)
(2)	G01 X20. Y30. F300.	G01で移動 (送り速度300[mm/min]で移動)
(3)	X30.	
(4)	X-20. Y-30.	
(5)	X-30.	



G02: 円弧補間CW(中心指定)

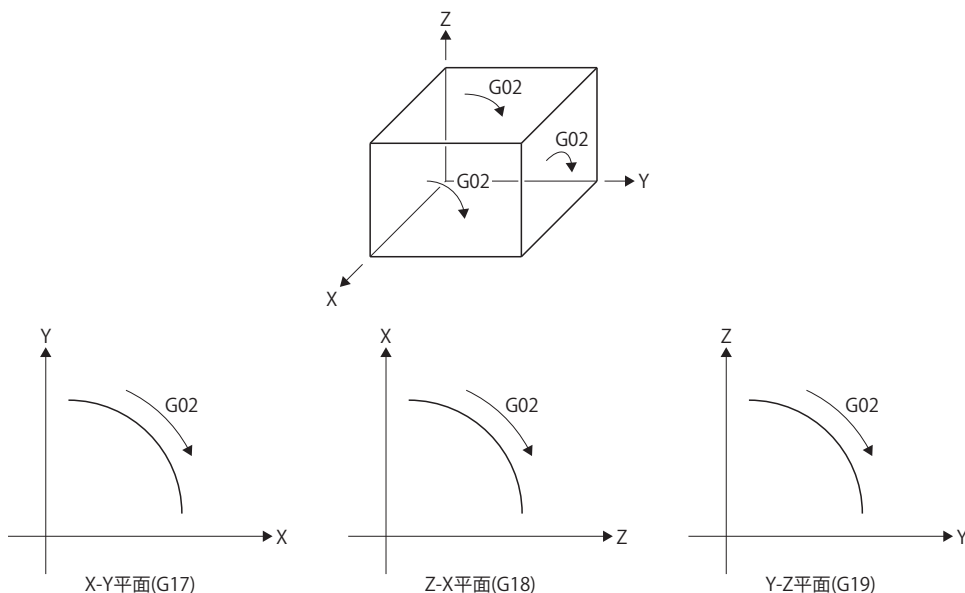
現在位置(始点)から指定された座標位置(終点)へ円弧(CW)で移動します。

移動速度は、指定された送り速度です。

コード	フォーマット
G02	<p>G02 <u>X x</u> <u>Y y</u> <u>I i</u> <u>J j</u> <u>F f</u></p> <p>送り速度 円弧中心座標 座標指令</p>

機能

- G02(CW)は円弧の終点座標をアドレスX, Y(またはZ)で与え、円弧の中心座標をアドレスI, J(またはK)で指定します。円弧の終点座標値の指令は、絶対値、増分値の併用が可能です。円弧の中心座標は必ず始点からの増分値で指令してください。
- 中心座標の基本軸は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“平面選択”⇒“基本軸I〜K”にて設定した軸名称となります。
- G02指令はモーダル指令です。同一グループのGコードを使用するまで有効となります。G02指令が連続する場合、次のブロック以降は座標語だけで指令することができます。円弧の回転方向は、CW(時計回り)です。



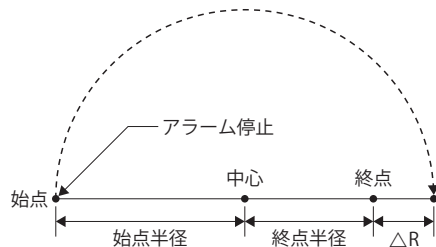
- ・多象限にまたがる円弧を1ブロック指令で実行できます。
- ・円弧の存在する平面には、下記の平面選択があり、Gコードを指令し選択されます。平面選択されていない軸を指定した場合は、軽度エラー（エラーコード: 1FC3H(詳細コード: 030AH))となります。

平面選択	Gコード
X-Y平面	G17
Z-X平面	G18
Y-Z平面	G19

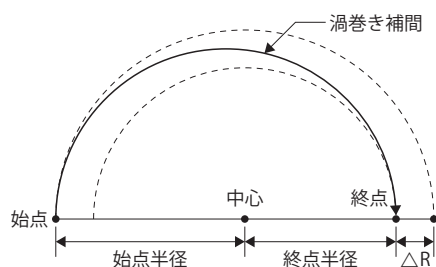
- 円弧補間指令時、中心指定がないと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0306H))となります。
- 減速チェック方式には、指令減速チェック方式、スムージングチェック方式、インポジションチェック方式の3種類があります。早送り/切削送りで使用する減速チェック方式は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“制御設定”⇒“減速チェック”にて設定します。減速チェックについては、減速チェックを参照してください。(👉 211ページ 減速チェック)

注意事項

- 下記の場合は、360°の円弧(真円)となります。
 - 終点座標をすべて省略し、アドレスI, J, Kを使用して中心を指令した場合
 - 終点と始点が同じ位置の場合
- 円弧指令にて、始点半径と終点半径が一致しない場合は、下記ようになります。
 - 誤差 ΔR が[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“制御設定”⇒“円弧誤差”にて設定した値より大きい場合、円弧始点において軽度エラー(エラーコード: 1FC3H(詳細コード: 0313H))となります。



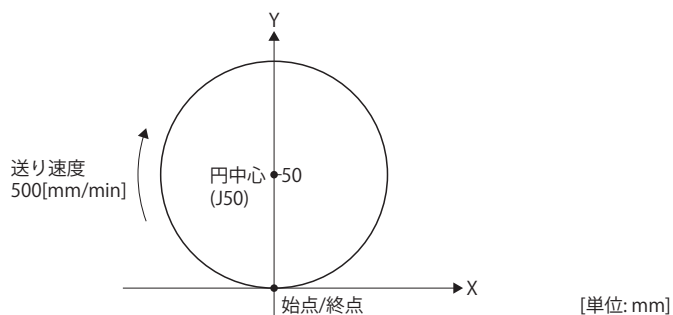
- 誤差 ΔR が[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“制御設定”⇒“円弧誤差”にて設定した値が範囲以内の場合は、指令終点に向かう渦巻き状の補間となります。



プログラム例

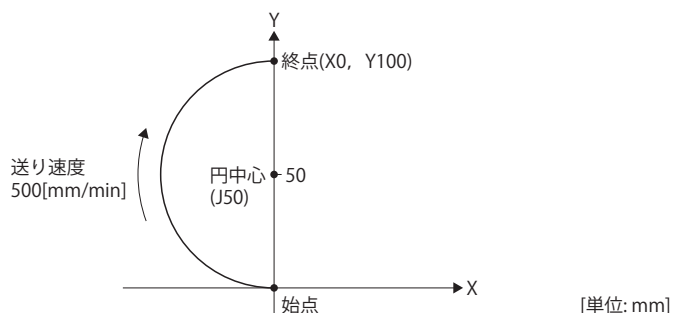
■現在位置から円弧補間を行い、真円を描くプログラム

プログラム	備考
G02 J50. F500.	真円指令



■現在位置から円弧補間を行い、半円を描くプログラム

プログラム	備考
G91 G02 X0. Y100. J50. F500.	—



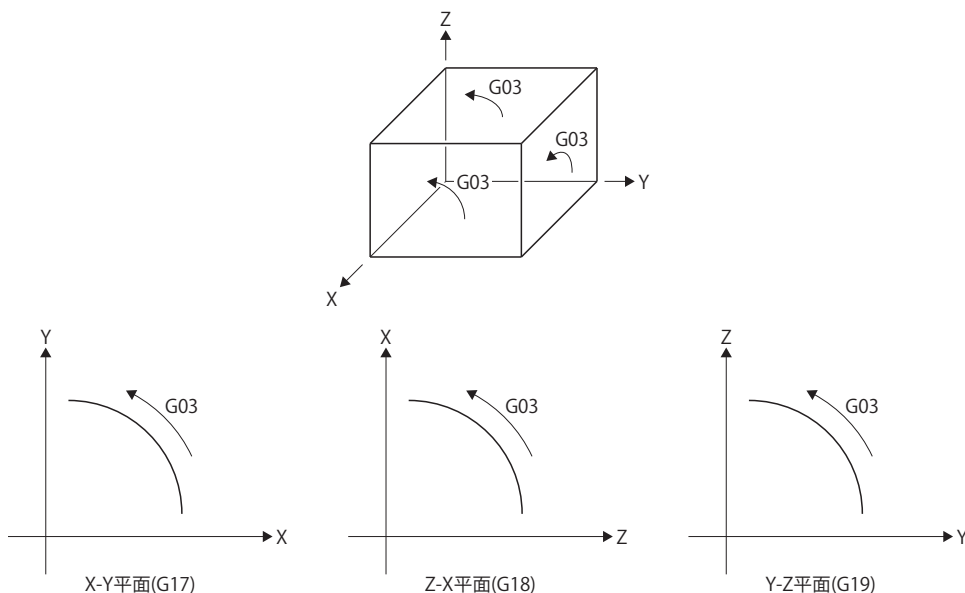
G03: 円弧補間CCW(中心指定)

現在位置(始点)から指定された座標位置(終点)へ円弧(CCW)で移動します。
移動速度は、指定された送り速度です。

コード	フォーマット
G03	G03_ X x _ Y y _ I i _ J j _ F f <div style="display: flex; justify-content: flex-end; margin-top: -20px;"> <div style="margin-right: 20px;">送り速度</div> <div style="margin-right: 20px;">円弧中心座標</div> <div>座標指令</div> </div>

機能

- G03(CCW)は円弧の終点座標をアドレスX, Y(またはZ)で与え、円弧の中心座標をアドレスI, J(またはK)で指定します。円弧の終点座標値の指令は、絶対値、増分値の併用が可能です。円弧の中心座標は必ず始点からの増分値で指令してください。
- 中心座標の基本軸は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御システムパラメータ]⇒“平面選択”⇒“基本軸I〜K”にて設定した軸名称となります。
- G03指令はモーダル指令です。同一グループのGコードを使用するまで有効となります。G03指令が連続する場合、次のブロック以降は座標語だけで指令することができます。円弧の回転方向は、CCW(反時計回り)です。



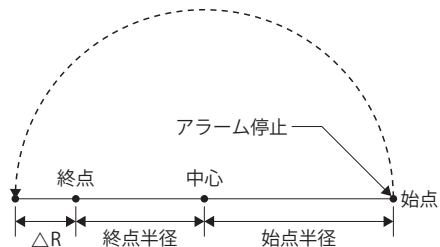
- 多象限にまたがる円弧を1ブロック指令で実行できます。
- 円弧の存在する平面には、下記の平面選択があり、Gコードを指令し選択されます。平面選択されていない軸を指定した場合は、軽度エラー(エラーコード: 1FC3H(詳細コード: 030AH))となります。

平面選択	Gコード
X-Y平面	G17
Z-X平面	G18
Y-Z平面	G19

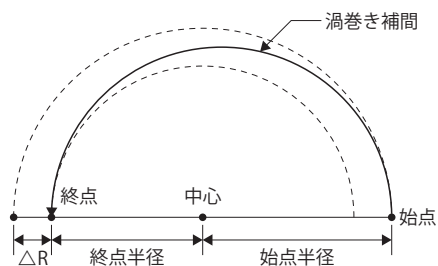
- 円弧補間指令時、中心指定がないと、軽度エラー(エラーコード: 1FC3H(詳細コード: 0306H))となります。
- 減速チェック方式には、指令減速チェック方式、スムージングチェック方式、インポジションチェック方式の3種類があります。早送り/切削送りで使用する減速チェック方式は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御システムパラメータ]⇒“制御設定”⇒“減速チェック”にて設定します。減速チェックについては、減速チェックを参照してください。(P211ページ 減速チェック)

注意事項

- 下記の場合は、360°の円弧(真円)となります。
 - 終点座標をすべて省略し、アドレスI, J, Kを使用して中心を指令した場合
 - 終点と始点が同じ位置の場合
- 円弧指令にて、始点半径と終点半径が一致しない場合は、下記ようになります。
 - 誤差 ΔR が[モーション制御/パラメータ]⇒[Gコード制御/パラメータ]⇒[Gコード制御システム/パラメータ]⇒“制御設定”⇒“円弧誤差”にて設定した値より大きい場合、円弧始点において軽度エラー(エラーコード: 1FC3H(詳細コード: 0313H))となります。



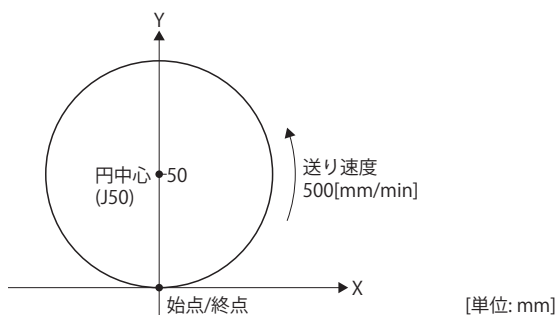
- 誤差 ΔR が[モーション制御/パラメータ]⇒[Gコード制御/パラメータ]⇒[Gコード制御システム/パラメータ]⇒“制御設定”⇒“円弧誤差”にて設定した値が範囲以内の場合は、指令終点に向かう渦巻き状の補間となります。



プログラム例

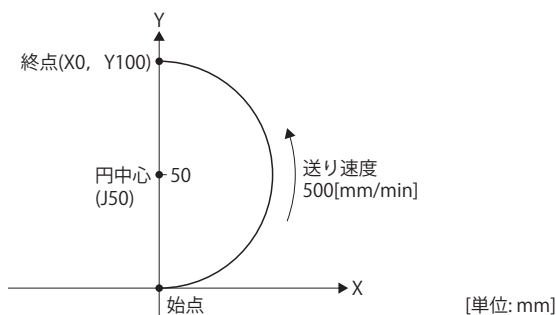
■現在位置から円弧補間を行い、真円を描くプログラム

プログラム	備考
G03 J50. F500.	真円指令



■現在位置から円弧補間を行い、半円を描くプログラム

プログラム	備考
G91 G03 X0. Y100. J50. F500.	—



G02: 円弧補間CW(R指定)

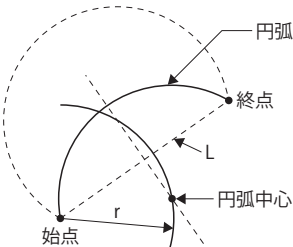
現在位置(始点)から指定された座標位置(終点)へ指定された半径の円弧(CW)で移動します。
移動速度は、指定された送り速度です。

コード	フォーマット
G02	G02_ X x _ Y y _ R r _ F f
	送り速度 円弧半径 座標指令

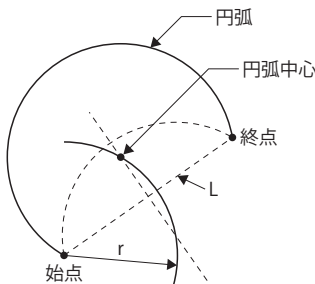
機能

- 円弧中心は、始点と終点を結ぶ線分に直交する2等分線上にあり、始点を中心とする指定半径の円との交点が指令された円弧指令の中心座標となります。
- 円弧半径(R)の値が正のときは180°以下の円弧、負のときは180°以上の円弧となります。

<半径値が正(R>0)のとき>



<半径値が負(R<0)のとき>

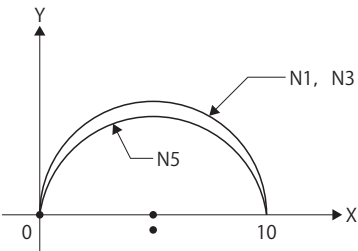


- R指定円弧補間指令は、下記の計算式となるように指定してください。「始点と終点を結ぶ線分(L)/2×円弧半径(r)>円弧誤差(パラメータ値)」となる場合は、軽度エラー (エラーコード: 1FC3H(詳細コード: 0313H))となります。

$$\frac{\text{始点と終点を結ぶ線分(L)}}{2 \times \text{円弧半径(r)}} \leq 1$$

- 計算誤差により円弧とならないときに、「始点と終点を結ぶ線分(L)」と「2×円弧半径(r)」の誤差が設定値以下の場合は、始点と終点を結ぶ線分の中点が円弧中心となるように補正します。
円弧誤差の設定値は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“制御設定”⇒“円弧中心誤差修正許容値”にて設定します。

円弧中心誤差修正許容値の設定値	円弧中心誤差修正許容値
設定値<0	0(中心誤差を補正しない)
設定値=0	2×最小設定単位(0.0001)
設定値>0	設定した値



```
G90 X0. Y0.  
N1 G02 X10. R5.0000  
N2 G00 X0.  
N3 G02 X10. R5.0001 .....(a)  
N4 G00 X0.  
N5 G02 X10. R5.0002 .....(b)  
N6 G00 X0.  
M02
```

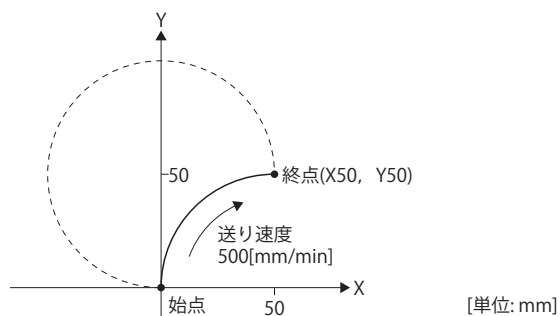
- (a)「中心座標補正する」=N1と同じ軌跡を通る。
(b)「中心座標補正しない」=N1より少し内側の軌跡を通る。
円弧中心誤差補正許容値: 0.0002mm
始点から終点の線分 : 10.0000
・ N3: 半径×2=10.0002 「誤差0.0002 → 許容値以下のため、補正する」
・ N5: 半径×2=10.0004 「誤差0.0004 → 許容値を超えているため、補正しない」

- G02(CW)で同一ブロック内にR指定と中心指定(I, J, K)を同時に指定した場合は、R指定による円弧指令が優先されます。
- R指定円弧補間において真円指令(始点と終点が同じ)を指定すると、R指定円弧指令は即完了し動作しません。
- 減速チェック方式には、指令減速チェック方式、スムージングチェック方式、インポジションチェック方式の3種類があります。早送り/切削送りで使用する減速チェック方式は、[モーション制御パラメータ]⇒[Gコード制御/パラメータ]⇒[Gコード制御システムパラメータ]⇒“減速チェック”⇒“制御設定”にて設定します。減速チェックについては、減速チェックを参照してください。(211ページ 減速チェック)

プログラム例

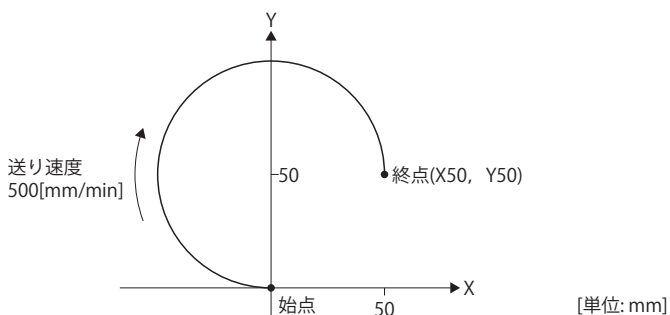
■円弧半径の値が正の場合、180°以下の円弧を描くプログラム

プログラム	備考
G91 G02 X50. Y50. R50. F500.	—



■円弧半径の値が負の場合、180°以上の円弧を描くプログラム

プログラム	備考
G91 G02 X50. Y50. R-50. F500.	—



G03: 円弧補間CCW(R指定)

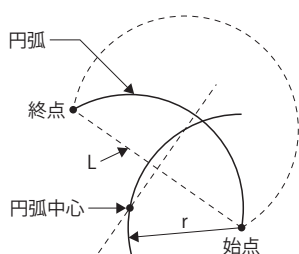
現在位置(始点)から指定された座標位置(終点)へ指定された半径の円弧(CCW)で移動します。
移動速度は、指定された送り速度です。

コード	フォーマット
G03	$G03 _X _x _Y _y _R _r _F _f$ <div style="display: flex; justify-content: flex-end; margin-top: -20px;"> <div style="margin-left: 20px;">送り速度</div> <div style="margin-left: 20px;">円弧半径</div> <div style="margin-left: 20px;">座標指令</div> </div>

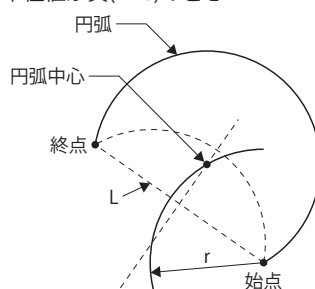
機能

- 円弧中心は、始点と終点を結ぶ線分に直交する2等分線上にあり、始点を中心とする指定半径の円との交点が指令された円弧指令の中心座標となります。
- 円弧半径(R)の値が正のときは180°以下の円弧、負のときは180°以上の円弧となります。

<半径値が正(R>0)のとき>



<半径値が負(R<0)のとき>

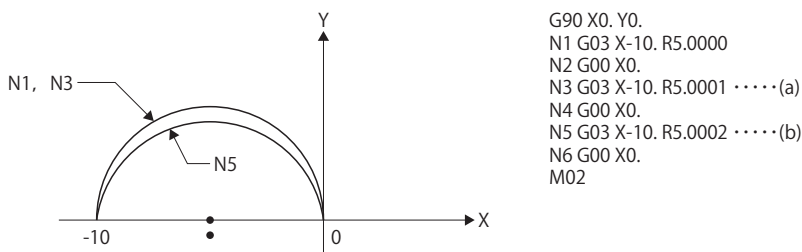


- R指定円弧補間指令は、下記の計算式となるように指定してください。「始点と終点を結ぶ線分(L)/2×円弧半径(r)>円弧誤差(パラメータ値)」となる場合は、軽度エラー(エラーコード: 1FC3H(詳細コード: 0313H))となります。

$$\frac{\text{始点と終点を結ぶ線分}(L)}{2 \times \text{円弧半径}(r)} \leq 1$$

- 計算誤差により円弧とならないときに、「始点と終点を結ぶ線分(L)」と「2×円弧半径(r)」の誤差が設定値以下の場合には、始点と終点を結ぶ線分の中点が円弧中心となるように補正します。
円弧誤差の設定値は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御システムパラメータ]⇒“制御設定”⇒“円弧中心誤差修正許容値”にて設定します。

円弧中心誤差修正許容値の設定値	円弧中心誤差修正許容値
設定値<0	0(中心誤差を補正しない)
設定値=0	2×最小設定単位(0.0001)
設定値>0	設定した値



```

G90 X0. Y0.
N1 G03 X-10. R5.0000
N2 G00 X0.
N3 G03 X-10. R5.0001 .....(a)
N4 G00 X0.
N5 G03 X-10. R5.0002 .....(b)
N6 G00 X0.
M02

```

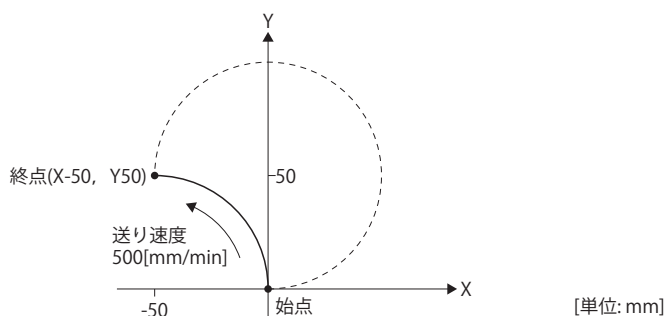
- (a)「中心座標補正する」=N1と同じ軌跡を通る。
(b)「中心座標補正しない」=N1より少し内側の軌跡を通る。
円弧中心誤差補正許容値: 0.0002mm
始点から終点の線分 : 10.0000
・N3: 半径×2=10.0002 「誤差0.0002 → 許容値以下のため、補正する」
・N5: 半径×2=10.0004 「誤差0.0004 → 許容値を超えているため、補正しない」

- G03(CCW)で同一ブロック内にR指定と中心指定(I, J, K)を同時に指定した場合は, R指定による円弧指令が優先されます。
- R指定円弧補間において真円指令(始点と終点が同じ)を指定すると, R指定円弧指令は即完了し動作しません。
- 減速チェック方式には, 指令減速チェック方式, スムージングチェック方式, インポジションチェック方式の3種類があります。早送り/切削送りで使用する減速チェック方式は, [モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御システムパラメータ]⇒“制御設定”⇒“減速チェック”にて設定します。減速チェックについては, 減速チェックを参照してください。(211ページ 減速チェック)

プログラム例

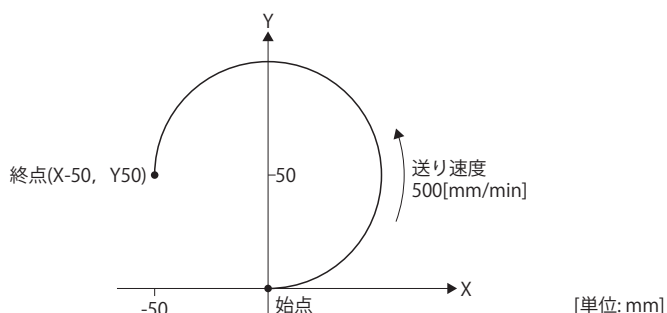
■円弧半径の値が正の場合, 180°以下の円弧を描くプログラム

プログラム	備考
G91 G03 X-50. Y50. R50. F500.	—



■円弧半径の値が負の場合, 180°以上の円弧を描くプログラム

プログラム	備考
G91 G03 X-50. Y50. R-50. F500.	—



G04: ドウェル(時間指定)

指定された時間、次のブロックへの実行を待ちます。

コード	フォーマット
G04	<div>■X指定時</div> <div>G04_ X x</div> <div>└─ ドウェル時間(0～99999.999[s])</div> <div>■P指定時</div> <div>G04_ P p</div> <div>└─ ドウェル時間(0～99999999[ms])</div>

機能

- Xによるドウェル時間の指定範囲は、「0～99999.999[s]」で小数点指令が有効です。小数点を省略した場合は、0.001[s]単位となります。
- Pによるドウェル時間の指定範囲は、「0～99999999[ms]」で小数点指令が無効です。Pによる小数点以下の値を指令した場合は、軽度エラー (エラーコード: 1FC3H(詳細コード: 031AH))となります。
- ドウェル指令は、前のブロックに切削指令があると減速停止完了後、ドウェル時間の計算を開始します。なお、M指令と同一ブロックに指令したときは、同時にスタートします。
- ドウェル実行中にフィードホールド信号が入力された場合、ドウェルをいったん停止し、再起動後に次ブロック実行待ち残り時間分のドウェルを実行します。

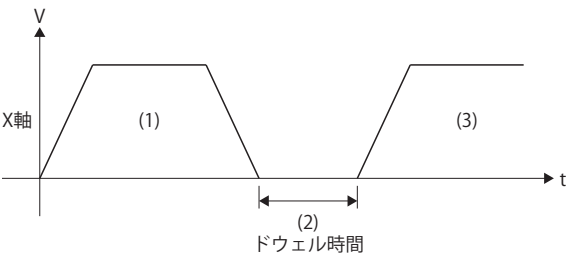
Point

移動指令とドウェル指令を同一のブロックに指定しないでください。移動指令は実行せず、G04のみ実行します。

プログラム例

■位置決め動作命令の間にドウェル時間を入れたプログラム

動作	プログラム	備考
(1)	G01 X100.F10.	位置決め
(2)	下表参照	ドウェル時間
(3)	G01 X200.	位置決め



(2)の指令	ドウェル時間[s]
G04 X500	0.5
G04 X5000	5
G04 X5.	5
G04 P5000	5
G04 P12.345	軽度エラー (エラーコード: 1FC3H(詳細コード: 031AH))

G09: イグザクトストップチェック

指定ブロックのみ減速チェックを行います。

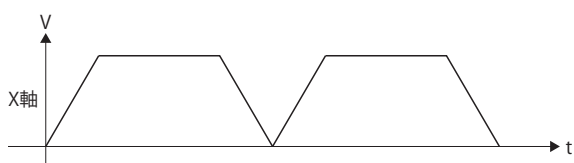
コード	フォーマット
G09	G09 <u> </u> G01 <u> </u> X <u> </u> x <u> </u> F <u> </u> f
	<div style="text-align: center;">└──────────┘ G01, G02, G03の指令のみ使用可能</div>

機能

- G09指令は、指定したブロックのみ減速チェックを行います。
- G09指令は指定された座標位置で減速チェックを行ってから、次ブロックを実行します。
- G09指令はアンモーダル指令です。指定ブロックのみ有効です。
- 減速チェックについての詳細は、減速チェックを参照してください。(P211ページ 減速チェック)

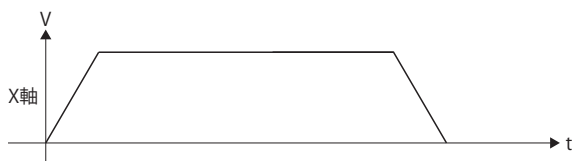
■イグザクトストップチェックを使用した場合

G09 G01 X100. F300.
X200.



■イグザクトストップチェックを使用しない場合

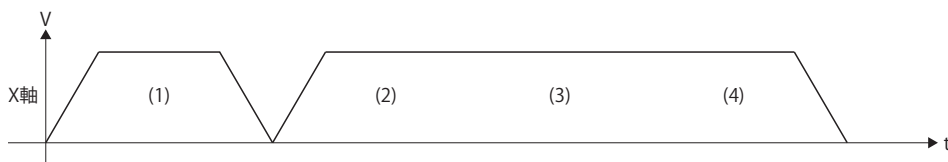
G01 X100. F300.
X200.



プログラム例

■イグザクトストップチェックにより、位置決めするプログラム

動作	プログラム	備考
(1)	G09 G01 X100. F500.	イグザクトストップチェックによる位置決め
(2)	X200.	位置決め
(3)	X300.	位置決め
(4)	G09 G01 X400.	イグザクトストップチェックによる位置決め



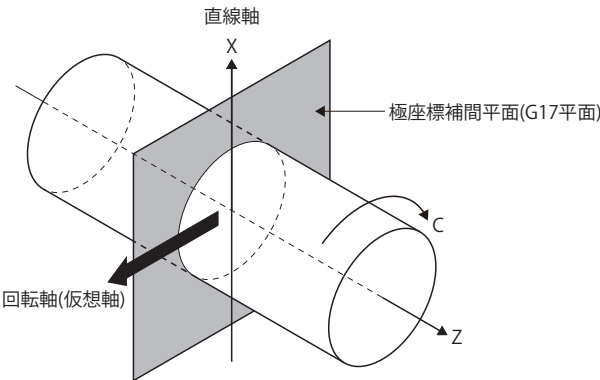
G12.1: 極座標補間モード開始

直線軸の移動(工具の移動)と回転軸の移動(ワークの回転)に変換して輪郭制御する極座標補間モードを開始します。
ワーク外径に直線上の切欠部を切削する場合、およびカムシャフトの研削等に有効です。

コード	フォーマット
G12.1	G12.1

機能

- G12.1指令は、直線軸を平面第1軸目、直交する仮想軸(極座標補間回転軸)を平面2軸目とする極座標補間平面が選択され、極座標補間はこの平面上で行います。また、極座標補間ではローカル座標系の原点を座標系の原点とします。



- G12.1指令はモーダル指令です。極座標補間モードキャンセル(G13.1)されるまで、極座標補間モードが保持されます。
- G12.1指令は、単独ブロックで指令してください。他の指令を同一ブロックに指令すると、軽度エラー (エラーコード: 1FC3H(詳細コード: 0306H, または0324H))となります。ただし、シーケンス番号(N), および速度(F, F)については、同一ブロックに指定できます。
- 極座標補間を行う直線軸と回転軸は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“極座標補間”⇒“極座標補間直線軸”, および“極座標補間回転軸”にて設定します。“極座標補間直線軸”, および“極座標補間回転軸”を設定せず、G12.1を指令すると、軽度エラー (エラーコード: 1FC3H(詳細コード: 0325H))となります。
- Gコード制御移行時、およびリセット時は、極座標補間モードキャンセル状態になります。
- 極座標補間モード開始指令ブロック(G12.1)では、減速チェックを行います。
- 極座標補間モード中、直線軸の可動領域は、極座標補間指令時の直線軸の位置により決定します。極座標補間モード中、極座標平面の直線軸の可動領域をプラス側で制御する場合、極座標補間指令前に直線軸が0以上のプラスの領域に移動している必要があります。逆に、直線軸の可動領域をマイナス側で制御する場合、極座標指令前に0を含まないマイナスの領域に移動している必要があります。

プラス側領域	マイナス側領域

■各機能との組合せによる極座標補間モードの動作

各機能による極座標補間モードの動作を下記に示します。

機能	極座標補間モード動作																																									
極座標補間でのプログラム指令	<div><div><div>・極座標補間モード中のプログラム指令は、極座標補間平面で、直線軸と回転軸(仮想軸)の直交座標値により指令します。平面第2軸(仮想軸)を指令する軸アドレスは、極座標補間回転軸の軸アドレスを使用します。このとき、指令の単位は、[degree]ではなく、平面第1軸(直線軸)と同じ単位[mm])で指令します。また、プログラム指令では、極座標補間平面の2軸(極座標補間直線軸、極座標補間回転軸)、および高さ軸のみ指令可能です。その他の軸を指令した場合、軽度エラー(エラーコード: 1FC3H(詳細コード: 0323H))となります。高さ軸を示す軸アドレスは、Gコード制御系統パラメータの「極座標補間直線軸」にて設定した軸名称により、下記ようになります。(基本軸Iに「1:X」、基本軸Jに「2:Y」、基本軸Kに「3:Z」を設定した場合)</div><table><tr><th>極座標補間直線軸の設定値</th><th>みなし平面</th><th>高さ</th></tr><tr><td>1: X</td><td>G17(XY平面)</td><td>Z</td></tr><tr><td>2: Y</td><td>G18(YZ平面)</td><td>X</td></tr><tr><td>3: Z</td><td>G19(ZX平面)</td><td>Y</td></tr></table><div>・極座標補間モード中の回転軸の回転方向は、Gコード制御軸パラメータの「回転軸タイプ」の設定によらず、近回りとなります。</div><div>・極座標補間モード開始指令(G12.1)が指令された時点での仮想軸座標値は、「0」となります。すなわちG12.1が指令された位置を「角度=0」とみなし極座標補間を開始します。また、極座標補間開始指令(G12.1)時、極座標補間キャンセル指令(G13.1)時のモニタデバイスは下記となります。</div><table><tr><th>モニタデバイス</th><th>内容</th></tr><tr><td>[Md.3147]機械位置 (D54768+32sn, D54769+32sn)</td><td rowspan="4">変化しません。</td></tr><tr><td>[Md.3148]機械目標位置 (D54770+32sn, D54771+32sn)</td></tr><tr><td>[Md.3149]相対位置 (D54772+32sn, D54773+32sn)</td></tr><tr><td>[Md.3150]相対目標位置 (D54774+32sn, D54775+32sn)</td></tr><tr><td>[Md.3152]プログラム目標位置 (D54778+32sn, D54779+32sn)</td><td>極座標補間モード開始指令(G12.1)時に「0」となります。 極座標補間モードキャンセル指令(G13.1)時は変化しません。</td></tr></table></div></div> <tr><td>位置決め(G00)</td><td><div>極座標補間モード中でも指令できます。</div><div>Gコード制御系統パラメータの「G00非補間」にて「非補間タイプ」を設定しても補間動作となります。</div><div>極座標補間モード中は、G1→G1移動方向反転時の減速チェック(Gコード制御系統パラメータの「G1→G1減速チェック」)は無効となります。</div></td></tr> <tr><td>直線補間(G01)</td><td><div>極座標補間モード中でも指令できます。</div><div>極座標補間モード中は、G1→G1移動方向反転時の減速チェック(Gコード制御系統パラメータの「G1→G1減速チェック」)は無効となります。</div></td></tr> <tr><td>円弧補間(G02, G03)</td><td><div>極座標補間モード中でも指令できます。中心点指定、半径指定が使用できます。</div><div>中心点指定円弧補間を行うときの中心点を指定するアドレスは、Gコード制御系統パラメータの「極座標補間直線軸」にて設定した軸名称により、下記ようになります。(基本軸Iに「1:X」、基本軸Jに「2:Y」、基本軸Kに「3:Z」を設定した場合)</div><table><tr><th>極座標補間直線軸の設定値</th><th>中心指定指令</th></tr><tr><td>1: X</td><td>I, J(極座標平面をXY平面とみなす)</td></tr><tr><td>2: Y</td><td>J, K(極座標平面をYZ平面とみなす)</td></tr><tr><td>3: Z</td><td>K, I(極座標平面をZX平面とみなす)</td></tr></table><div>極座標補間モード中は、G1→G1移動方向反転時の減速チェック(Gコード制御系統パラメータの「G1→G1減速チェック」)は無効となります。</div></td></tr> <tr><td>ドウェル(G04)</td><td rowspan="5"><div>極座標補間モード中でも指令できます。</div></td></tr> <tr><td>イクザクトストップチェック(G09)</td></tr> <tr><td>工具径補正ベクトル指定(G38)</td></tr> <tr><td>工具径補正コーナ円弧(G39)</td></tr> <tr><td>マクロ呼出し(G65)</td></tr>	極座標補間直線軸の設定値	みなし平面	高さ	1: X	G17(XY平面)	Z	2: Y	G18(YZ平面)	X	3: Z	G19(ZX平面)	Y	モニタデバイス	内容	[Md.3147]機械位置 (D54768+32sn, D54769+32sn)	変化しません。	[Md.3148]機械目標位置 (D54770+32sn, D54771+32sn)	[Md.3149]相対位置 (D54772+32sn, D54773+32sn)	[Md.3150]相対目標位置 (D54774+32sn, D54775+32sn)	[Md.3152]プログラム目標位置 (D54778+32sn, D54779+32sn)	極座標補間モード開始指令(G12.1)時に「0」となります。 極座標補間モードキャンセル指令(G13.1)時は変化しません。	位置決め(G00)	<div>極座標補間モード中でも指令できます。</div> <div>Gコード制御系統パラメータの「G00非補間」にて「非補間タイプ」を設定しても補間動作となります。</div> <div>極座標補間モード中は、G1→G1移動方向反転時の減速チェック(Gコード制御系統パラメータの「G1→G1減速チェック」)は無効となります。</div>	直線補間(G01)	<div>極座標補間モード中でも指令できます。</div> <div>極座標補間モード中は、G1→G1移動方向反転時の減速チェック(Gコード制御系統パラメータの「G1→G1減速チェック」)は無効となります。</div>	円弧補間(G02, G03)	<div>極座標補間モード中でも指令できます。中心点指定、半径指定が使用できます。</div> <div>中心点指定円弧補間を行うときの中心点を指定するアドレスは、Gコード制御系統パラメータの「極座標補間直線軸」にて設定した軸名称により、下記ようになります。(基本軸Iに「1:X」、基本軸Jに「2:Y」、基本軸Kに「3:Z」を設定した場合)</div> <table><tr><th>極座標補間直線軸の設定値</th><th>中心指定指令</th></tr><tr><td>1: X</td><td>I, J(極座標平面をXY平面とみなす)</td></tr><tr><td>2: Y</td><td>J, K(極座標平面をYZ平面とみなす)</td></tr><tr><td>3: Z</td><td>K, I(極座標平面をZX平面とみなす)</td></tr></table> <div>極座標補間モード中は、G1→G1移動方向反転時の減速チェック(Gコード制御系統パラメータの「G1→G1減速チェック」)は無効となります。</div>	極座標補間直線軸の設定値	中心指定指令	1: X	I, J(極座標平面をXY平面とみなす)	2: Y	J, K(極座標平面をYZ平面とみなす)	3: Z	K, I(極座標平面をZX平面とみなす)	ドウェル(G04)	<div>極座標補間モード中でも指令できます。</div>	イクザクトストップチェック(G09)	工具径補正ベクトル指定(G38)	工具径補正コーナ円弧(G39)	マクロ呼出し(G65)
極座標補間直線軸の設定値	みなし平面	高さ																																								
1: X	G17(XY平面)	Z																																								
2: Y	G18(YZ平面)	X																																								
3: Z	G19(ZX平面)	Y																																								
モニタデバイス	内容																																									
[Md.3147]機械位置 (D54768+32sn, D54769+32sn)	変化しません。																																									
[Md.3148]機械目標位置 (D54770+32sn, D54771+32sn)																																										
[Md.3149]相対位置 (D54772+32sn, D54773+32sn)																																										
[Md.3150]相対目標位置 (D54774+32sn, D54775+32sn)																																										
[Md.3152]プログラム目標位置 (D54778+32sn, D54779+32sn)	極座標補間モード開始指令(G12.1)時に「0」となります。 極座標補間モードキャンセル指令(G13.1)時は変化しません。																																									
位置決め(G00)	<div>極座標補間モード中でも指令できます。</div> <div>Gコード制御系統パラメータの「G00非補間」にて「非補間タイプ」を設定しても補間動作となります。</div> <div>極座標補間モード中は、G1→G1移動方向反転時の減速チェック(Gコード制御系統パラメータの「G1→G1減速チェック」)は無効となります。</div>																																									
直線補間(G01)	<div>極座標補間モード中でも指令できます。</div> <div>極座標補間モード中は、G1→G1移動方向反転時の減速チェック(Gコード制御系統パラメータの「G1→G1減速チェック」)は無効となります。</div>																																									
円弧補間(G02, G03)	<div>極座標補間モード中でも指令できます。中心点指定、半径指定が使用できます。</div> <div>中心点指定円弧補間を行うときの中心点を指定するアドレスは、Gコード制御系統パラメータの「極座標補間直線軸」にて設定した軸名称により、下記ようになります。(基本軸Iに「1:X」、基本軸Jに「2:Y」、基本軸Kに「3:Z」を設定した場合)</div> <table><tr><th>極座標補間直線軸の設定値</th><th>中心指定指令</th></tr><tr><td>1: X</td><td>I, J(極座標平面をXY平面とみなす)</td></tr><tr><td>2: Y</td><td>J, K(極座標平面をYZ平面とみなす)</td></tr><tr><td>3: Z</td><td>K, I(極座標平面をZX平面とみなす)</td></tr></table> <div>極座標補間モード中は、G1→G1移動方向反転時の減速チェック(Gコード制御系統パラメータの「G1→G1減速チェック」)は無効となります。</div>	極座標補間直線軸の設定値	中心指定指令	1: X	I, J(極座標平面をXY平面とみなす)	2: Y	J, K(極座標平面をYZ平面とみなす)	3: Z	K, I(極座標平面をZX平面とみなす)																																	
極座標補間直線軸の設定値	中心指定指令																																									
1: X	I, J(極座標平面をXY平面とみなす)																																									
2: Y	J, K(極座標平面をYZ平面とみなす)																																									
3: Z	K, I(極座標平面をZX平面とみなす)																																									
ドウェル(G04)	<div>極座標補間モード中でも指令できます。</div>																																									
イクザクトストップチェック(G09)																																										
工具径補正ベクトル指定(G38)																																										
工具径補正コーナ円弧(G39)																																										
マクロ呼出し(G65)																																										

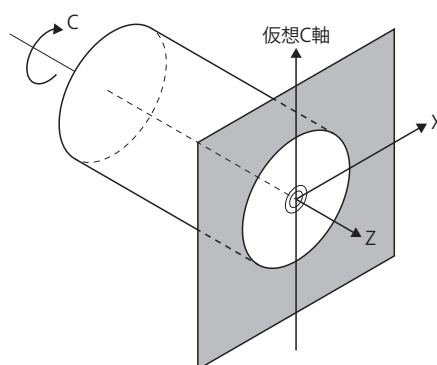
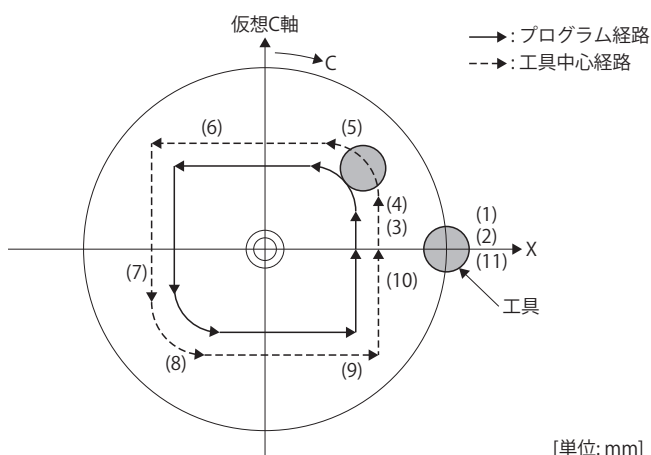
機能	極座標補間モード動作								
平面選択	<p>極座標補間モード開始指令(G12.1)で、Gコード制御系統パラメータの「極座標補間直線軸」にて設定した軸名称により、下表のように、極座標補間を行うとみなし平面(極座標補間平面)を決定します。</p> <p>極座標補間モードキャンセル指令(G13.1)で、極座標補間開始前に選択していた平面に戻ります。(基本軸Iに「1:X」、基本軸Jに「2:Y」、基本軸Kに「3:Z」を設定した場合)</p> <table border="1"> <thead> <tr> <th>極座標補間直線軸の設定値</th><th>みなし平面</th></tr> </thead> <tbody> <tr> <td>1:X</td><td>G17(XY平面)</td></tr> <tr> <td>2:Y</td><td>G18(YZ平面)</td></tr> <tr> <td>3:Z</td><td>G19(ZX平面)</td></tr> </tbody> </table> <p>極座標補間モード中に平面選択指令(G17～G19)を行うと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0322H))となります。</p>	極座標補間直線軸の設定値	みなし平面	1:X	G17(XY平面)	2:Y	G18(YZ平面)	3:Z	G19(ZX平面)
極座標補間直線軸の設定値	みなし平面								
1:X	G17(XY平面)								
2:Y	G18(YZ平面)								
3:Z	G19(ZX平面)								
絶対値指令/増分値指令	座標指令は、絶対値指令、増分値指令ともに使用できます。								
速度指令	<p>送り速度は、極座標補間平面(直交座標系)上の補間速度になりますので、極座標補間平面における接線速度を「F」により指令してください。(極座標変換により工具との相対速度が変化します。)</p> <p>Fの単位は、[mm/min]です。</p> <p>極座標補間モード中にF指令を指定しない場合は、直前のF指令の送り速度をそのまま使用します。極座標補間モードキャンセル後のF指令のモーダル値は、極座標補間モード開始、または極座標補間モード中に設定した最後のF指令の送り速度のままとなります。</p> <p>極座標補間平面(直交座標系)で、回転軸の中心付近を通過する場合、極座標補間後の回転軸側の送り速度は、非常に大きくなります。回転軸の送り速度が、切削送りクランプ速度より大きくなる場合、回転軸が切削送りクランプ速度を超えないように、回転軸および直線軸の速度をクランプします。</p>								
工具径補正	<ul style="list-style-type: none"> 極座標補間モード中に、プログラム指令に対して、工具径補正が使用できます。工具径補正後の経路に対して、極座標補間が行われます。 極座標補間平面に対して、工具径補正を使用できます。工具径補正を行う場合は、極座標補間モードの中でスタートアップ、キャンセルを行ってください。工具径補正中に極座標補間モード開始指令(G12.1)、または極座標補間モードキャンセル指令(G13.1)を行うと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0324H))となります。 工具径補正をG40単独指令でキャンセル後に移動指令がないままG12.1指令、またはG13.1指令を指令すると、G12.1指令ブロック、またはG13.1指令ブロックの軸の位置を工具補正キャンセル後の位置とみなして以降の動作を行います。 								
工具長補正	<ul style="list-style-type: none"> 極座標補間モード中に工具長補正を行うと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0322H))となります。 <p><例></p> <pre> : : G43 H12 極座標補間を行う前の工具長補正: 有効 G00 X100. Z0. G12.1 : G43 H11 極座標補間モード中の工具長補正: 軽度エラー : G13.1 </pre> <ul style="list-style-type: none"> 極座標補間モード開始前に工具補正動作(工具長補正量の移動)を完了してください。極座標補間モード開始指令時に工具補正動作が未完了の場合は、下記ようになります。 <ul style="list-style-type: none"> G12.1指令を実行しても、機械座標は変化しません。 G12.1指令を実行すると、ワーク座標は工具長補正動作後の値になります。 <p>また、極座標補間モード中は、工具長補正量の更新を行いません。そのため、工具長補正一時キャンセル中に極座標補間となった場合、工具長補正量はキャンセルされたままとなり、極座標補間モードキャンセル後の移動指令から再度工具長補正がかかります。</p> <p><例></p> <pre> : : G43 H1 Y0. Y軸を補正軸とする : : G53 X0. Y0. Z0. .. 工具長補正を一時キャンセルする G12.1 (工具長補正をキャンセルしたまま) : : G13.1 (工具長補正をキャンセルしたまま) Y10. 工具長補正が再開される : </pre>								
ローカル座標系設定(G52)	極座標補間モード中に指令した場合、軽度エラー (エラーコード: 1FC3H(詳細コード: 0322H))となります。								
基本機械座標系選択(G53)									
ワーク座標系選択	<p>極座標補間が指令される前に、回転軸の中心が座標系の原点となるようにワーク座標系を設定してください。</p> <p>極座標補間モード中に指令した場合、軽度エラー (エラーコード: 1FC3H(詳細コード: 0322H))となります。</p>								
イクザクトストップチェックモード	<p>極座標補間モード中に、イクザクトストップチェックモードへ切換えることができます。</p> <p>イクザクトストップチェックモード中に、極座標補間モード開始指令(G12.1)を行えます。</p>								
自動コーナオーバーライド	<p>自動コーナオーバーライド中に、極座標補間モード開始指令(G12.1)を行うと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0324H))となります。</p> <p>極座標補間モード中に、自動コーナオーバーライドを指令した場合、軽度エラー (エラーコード: 1FC3H(詳細コード: 0322H))となります。</p>								

機能	極座標補間モード動作
高精度制御モード	<p>極座標補間モード中に、高精度制御モードへ切換えることができます。 高精度制御モード中に、極座標補間モード開始指令(G12.1)を行えます。 ただし、高精度制御モードと組合せて使用する場合は、各軸許容加速度制御が有効となるようなパラメータを設定してください。 各軸許容加速度制御が無効の場合、極座標補間モード中に高精度制御モード(G61.1)を指令すると、軽度エラー (エラーコード: 1FC3H(詳細コード: 0322H))となります。また、高精度制御モード中に極座標補間モード開始指令 (G12.1)を指令すると、軽度エラー (エラーコード: 1FC3H(詳細コード: 0324H))となります。 極座標補間モード中は、円弧入口/出口速度制御は有効となりません。 高精度制御モードの詳細は、高精度制御を参照してください。(P255ページ 高精度制御)</p> <p>■注意事項 高精度制御モードと組合せて使用し、極座標補間平面の中心付近を通過するプログラムを動作させた場合、速度がばらつくことがあります。軌跡精度に影響はありません。速度変動を抑えたい場合は、指令速度を小さい値にしてください。</p>
切削モード	<p>極座標補間モード中に、切削モードへ切換えることができます。 切削モード中に、極座標補間モード開始指令(G12.1)を行えます。</p>
法線制御	<ul style="list-style-type: none"> 極座標補間モードと法線制御は同時に使用できません。 法線制御中に極座標補間モード開始指令(G12.1)を行うと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0324H))となります。 極座標補間モード中に、法線制御指令(G41.1, G42.1)を行うと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0322H))となります。
プログラム座標回転	<ul style="list-style-type: none"> 極座標補間モードとプログラム座標回転モードは、同時に使用できません。 プログラム座標回転モード中に極座標補間モード開始指令(G12.1)を行うと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0324H))となります。 極座標補間モード中にプログラム座標回転モード指令(G68)を行うと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0322H))となります。

プログラム例

■極座標補間モードにより、位置決めするプログラム

動作	プログラム	備考
(1)	G17 G90 G00 X40.0 C0 Z0	開始位置の決定
(2)	G12.1	極座標補間モード: 開始
(3)	G01 G42 X20.0 F2000	実加工開始
(4)	C10.0	形状プログラム (X-C 平面での直交座標値による。)
(5)	G03 X10.0 C20.0 R10.0	
(6)	G01 X-20.0	
(7)	C-10.0	
(8)	G03 X-10.0 C-20.0 I10.0 J0	
(9)	G01 X20.0	
(10)	C00	実加工終了
(11)	G40 X40.0	終了位置に移動



G13.1: 極座標補間モードキャンセル

極座標補間モードを終了します。

コード	フォーマット
G13.1	G13.1

機能

- G13.1指令は、設定した極座標補間モード開始(G12.1)をキャンセルして極座標補間モードを終了します。
- G13.1指令は、単独ブロックで指令してください。他の指令を同一ブロックに指令すると、軽度エラー (エラーコード: 1FC3H(詳細コード: 0306H, または0324H))となります。ただし、シーケンス番号(N), および速度(F, ,F)については、同一ブロックに指定できます。
- 極座標補間モードキャンセル指令ブロック(G13.1)では、減速チェックを行います。

プログラム例

■極座標補間モードをキャンセルするプログラム

動作	プログラム	備考
(1)	G17 G90 G00 X40.0 C0 Z0	開始位置の決定
(2)	G12.1	極座標補間モード: 開始
(3)	G01 G42 X20.0 F2000	実加工開始
(4)	C10.0	形状プログラム (X-C 平面での直交座標値による。)
(5)	G03 X10.0 C20.0 R10.0	
(6)	G01 X-20.0	
(7)	C-10.0	
(8)	G03 X-10.0 C-20.0 I10.0 J0	
(9)	G01 X20.0	
(10)	C00	実加工終了
(11)	G40 X40.0	終了位置に移動
(12)	G13.1	極座標補間モード: キャンセル
(13)	M30	プログラムの終了

G17～G19: 平面選択

円弧補間を行う平面、および工具径補正を行う平面を指定します。

コード	フォーマット
G17, G18, G19	G17 G18 G19

機能

- 平面選択は、円弧補間、または工具径補正を行うための平面を指定します。

Gコード	内容
G17	X-Y平面選択
G18	Z-X平面選択
G19	Y-Z平面選択

- 平面を構成する基本軸は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“平面構成”⇒“基本軸I～K”にて設定します。詳細は、基本軸I～Kを参照してください。(P76ページ 基本軸I/基本軸J/基本軸K)
- 平面選択指令(G17, G18, G19)はモーダル指令です。同一グループのGコードを使用するまで有効となり、指令されないブロックでは平面を切り換えません。
- 平面選択指令(G17, G18, G19)と同一ブロックに軸アドレスを指令すると、指令した軸は移動します。
- 平面選択指令(G17, G18, G19)により決まる平面に存在しない軸指令は、平面選択と無関係です。

例

下表のように登録し、G17 X100. Z100.を指令した場合。
XY平面が選択され、Zは平面と無関係に移動します。

基本軸	設定
I	1: X
J	2: Y
K	3: Z

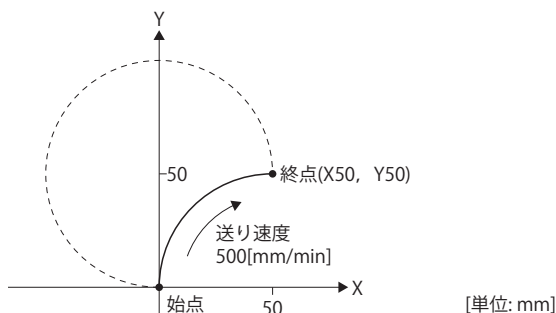
Point

Gコード制御開始時、およびリセット時は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“モーダル初期設定”⇒“平面選択”にて設定されている平面が指定されます。

プログラム例

■X-Y平面を選択し、円弧を描くプログラム

動作	プログラム	備考
(1)	G17	X-Y平面を選択
(2)	G91 G02 X50. Y50. R50. F500.	円弧補間の位置決め



G38: 工具径補正ベクトル指定

工具径補正中に補正ベクトルの変更，または保持を行います。

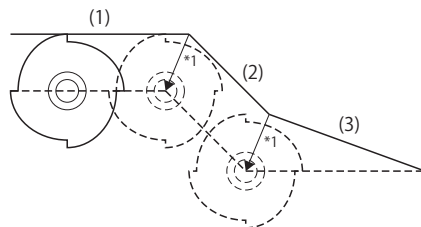
コード	フォーマット
G38	<p>G38 <u>l i l J j</u></p> <p>補正ベクトル方向</p>

機能

- G38指令を実行すると、工具径補正中に補正ベクトルの変更、または保持を行うことができます。
- G38指令はアンモーダル指令です。指定ブロックのみ有効です。
- 工具径補正についての詳細は、工具径補正を参照してください。(📖 216ページ 工具径補正)

■補正ベクトルの保持

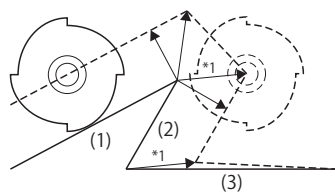
- ・ G38を移動指令のあるブロックで指令すると、そのブロックの終点では交点計算せずに、前のブロックのベクトルが保持されます。
- ・ 補正ベクトル保持の動作を下記に示します。
 - ・ 内側補正のベクトル保持



- (1) G01 Xx1
- (2) G38 Xx2 Yy2
- (3) G40 Xx3

*1: (1)-(2)ブロックの交点計算でのベクトル

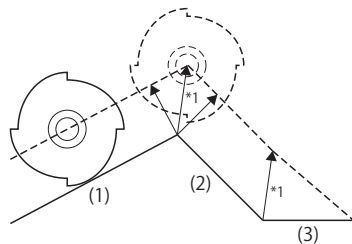
- ・外側補正鋭角のベクトル保持



- (1) G01 Xx1 Yy1
- (2) G38 Xx2 Yy2
- (3) G40 Xx3

*1: (1)-(2)ブロックの交点計算でのベクトル

- ・外側補正鈍角のベクトル保持

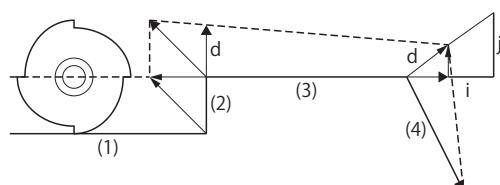


- (1) G01 Xx1 Yy1
- (2) G38 Xx2 Yy2
- (3) G40 Xx3

*1: (1)-(2)ブロックの交点計算でのベクトル

■補正ベクトルの変更

- 新たな補正ベクトルの方向を「I, J, K」、補正量を「D」にて指定できます。
- 移動指令と同一ブロックに指令できます。
- 「I, J, K」で有効なアドレスは、選択されている平面により異なります。
- 円弧指令の「I, J」の指令と同じブロックにG38を指令した場合、「I, J」はG38指令のベクトル方向として扱われるため、軽度エラー（エラーコード: 1FC3H(詳細コード: 0306H))となります。
- 補正ベクトル変更の動作を下記に示します。
 - ベクトルの変更



- (1) G01 Xx1
- (2) Yy2
- (3) G38 Xx3 li Jj Dd
- (4) G40 Xx4 Yy4

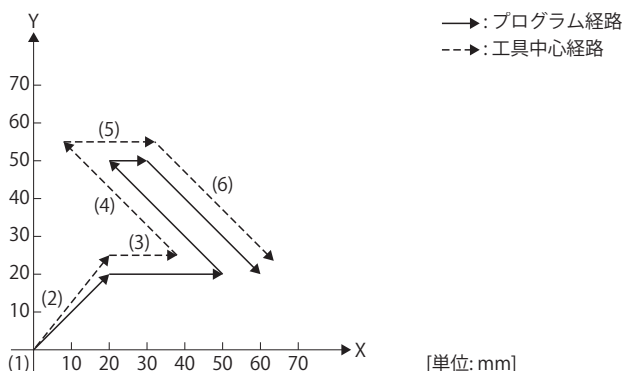
プログラム例

■補正ベクトルを保持し、位置決めするプログラム

工具径補正量は「補正番号:01」,「補正量:5[mm]」の場合

動作	プログラム	備考 ^{*1}
(1)	G90 X0. Y0.	絶対値指令により「X0, Y0」へ移動
(2)	G91 G01 G41 X20. Y20. D01 F500.	工具径補正開始動作で「X20, Y20」に移動
(3)	G01 X30.	「X50」に直線で移動
(4)	G01 G38 X-30. Y30.	「X20, Y50」に直線で移動
(5)	X10.	「X30」に直線で移動
(6)	X30. Y-30.	「X60, Y20」に直線で移動

*1 プログラム位置について説明しています。

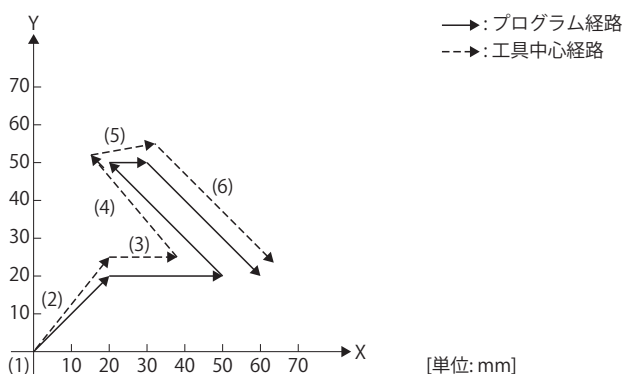


■補正ベクトルを変更し、位置決めするプログラム

工具径補正量は「補正番号:01」,「補正量:5[mm]」の場合

動作	プログラム	備考 ^{*1}
(1)	G90 X0. Y0.	絶対値指令により「X0, Y0」へ移動
(2)	G91 G01 G41 X20. Y20. D01 F500.	工具径補正開始動作で「X20, Y20」に移動
(3)	G01 X30.	「X50」に直線で移動
(4)	G01 G38 X-30. Y30. I-2. J2.	「X20, Y50」に直線で移動
(5)	X10.	「X30」に直線で移動
(6)	X30. Y-30.	「X60, Y20」に直線で移動

*1 プログラム位置について説明しています。



G39: 工具径補正コーナ円弧

ワークコーナでの交点計算を行わず，補正量を半径とする円弧を挿入します。

コード	フォーマット
G39	<div> <div>G39</div><div> <div>X</div><div>x</div> <div>Y</div><div>y</div> </div> <div>座標指令</div> </div>

機能

- G39指令を実行すると，ワークコーナでの交点計算を行わず，補正量を半径とする円弧を挿入します。

補正	G39指令あり	G39指令なし
外側補正		
内側補正		

- G39指令はアンモーダル指令です。指定ブロックのみ有効です。
- 工具径補正についての詳細は，工具径補正を参照してください。(P216ページ 工具径補正)

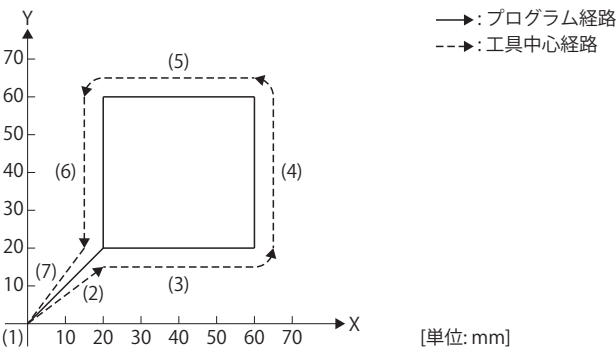
プログラム例

■指令位置に補正量を半径として位置決めするプログラム

工具径補正量は「補正番号: 01」，「補正量: 5[mm]」の場合

動作	プログラム	備考 ^{*1}
(1)	G90 X0. Y0.	絶対値指令により「X0, Y0」へ移動
(2)	G91 G01 G42 X20. Y20. D01 F100.	工具径補正開始動作で「X20, Y15」に移動
(3)	G39 X40.	「X60, Y15」に直線で移動後，「X65, Y20」に円弧で移動
(4)	G39 Y40.	「X65, Y60」に直線で移動後，「X60, Y65」に円弧で移動
(5)	G39 X-40.	「X20, Y65」に直線で移動後，「X15, Y60」に円弧で移動
(6)	Y-40.	「X15, Y20」に直線で移動
(7)	G40 X-20. Y-20.	工具径キャンセル動作で「X0, Y0」に移動
(8)	M02	プログラムエンド

^{*1} 工具中心位置について説明しています。



G40: 工具径補正キャンセル

設定した工具径補正量(G41, G42)をキャンセルします。

コード	フォーマット
G40	G40 <u> </u> X <u> </u> x <u> </u> Y <u> </u> y 座標指令

機能

- G40指令は、設定した工具径補正量(G41, G42)をキャンセルして指定の位置決めを行います。
- 工具径補正についての詳細は、工具径補正を参照してください。(216ページ 工具径補正)

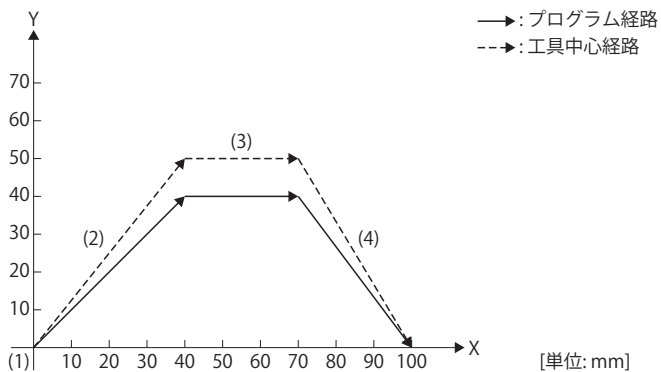
プログラム例

■工具径補正(左方向)のキャンセルを行うプログラム

工具径補正量は「補正番号: 10」, 「補正量: 10[mm]」の場合

動作	プログラム	備考 ^{*1}
(1)	G90 X0. Y0.	絶対値指令により「X0, Y0」へ移動
(2)	G91 G01 G41 X40. Y40. D10 F500.	工具径補正開始動作で「X40, Y50」に移動
(3)	G01 X30.	「X70」に直線で移動
(4)	G40 X30. Y-40.	工具径補正をキャンセルし「X0, Y0」に移動

*1 工具中心位置について説明しています。



G41: 工具径補正 左

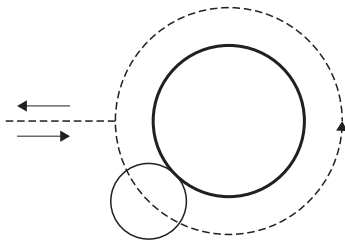
プログラム軌跡に対し、実際の工具中心軌跡を工具半径量だけ補正します。交点演算方式にて軌跡計算を行いますので、コーナ内側における切り込みすぎを防ぐことができます。

コード	フォーマット
G41	G41 <u>X x</u> <u>Y y</u> <u>D d</u> <div><div></div><div></div><div>工具番号</div><div>座標指令</div></div>

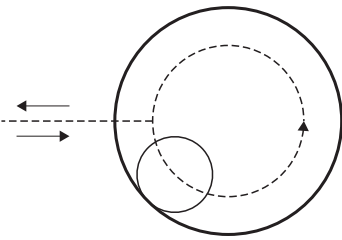
機能

- G41指令を実行すると、移動指令の終点位置を[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御加工パラメータ]⇒“工具補正データ”⇒“径補正量”にて設定した工具径補正量を加えて外側、または内側に補正し移動します。
- G41指令はモーダル指令です。工具径補正キャンセル(G40)されるまで、補正量が保持されます。
- 工具径補正において、H指令は無視され、D指令のみ有効となります。
- 工具番号に「0」を指定すると、工具径補正指令がキャンセルされます。
G41 D0 (工具番号0で、工具径補正キャンセル)
- 工具径補正は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“モーダル初期選択”⇒“平面選択”にて指定した平面内において行われます。指定した平面に含まれる軸以外は、補正を行いません。
- 補正量の値を正(+), または負(-)に設定した場合、動作は下記ようになります。

<補正量が正(+)の場合>



<補正量が負(-)の場合>



→: プログラム経路
--→: 工具中心経路

- 工具径補正についての詳細は、工具補正径を参照してください。(216ページ 工具径補正)

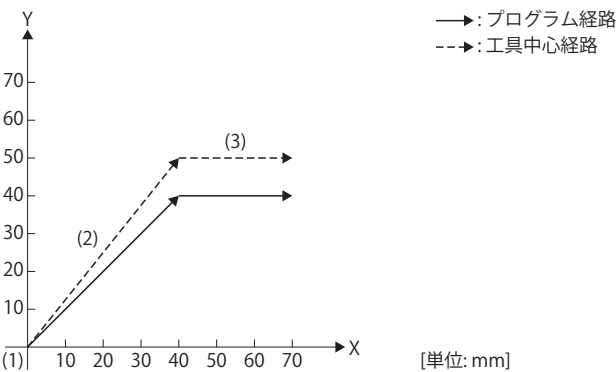
プログラム例

■工具径補正(左方向)を開始し、位置決めを行うプログラム

工具径補正量は「補正番号: 10」, 「補正量: 10[mm]」の場合

動作	プログラム	備考 ^{*1}
(1)	G90 X0. Y0.	絶対値指令により「X0, Y0」へ移動
(2)	G91 G01 G41 X40. Y40. D10 F500.	工具径補正開始動作で「X40, Y50」に移動
(3)	G01 X30.	「X70」に直線で移動

*1 工具中心位置について説明しています。



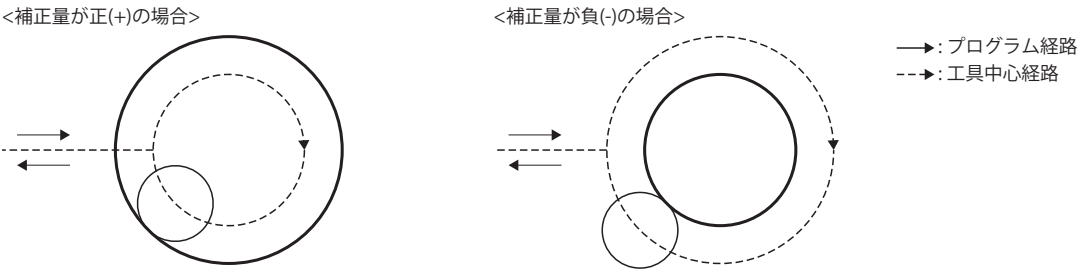
G42: 工具径補正 右

プログラム軌跡に対し、実際の工具中心軌跡を工具半径量だけ補正します。交点演算方式にて軌跡計算を行いますので、コーナ内側における切り込みすぎを防ぐことができます。

コード	フォーマット
G42	<div> <div>G42</div><div>X</div><div>x</div><div>Y</div><div>y</div><div>D</div><div>d</div> <div> <div></div><div></div><div></div><div></div><div></div><div></div> <div> <div></div><div></div> <div>工具番号</div> <div>座標指令</div> </div> </div> </div>

機能

- G42指令を実行すると、移動指令の終点位置を[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御加工パラメータ]⇒“工具補正データ”⇒“径補正量”にて設定した工具径補正量を加えて外側、または内側に補正し移動します。
- G42指令はモーダル指令です。工具径補正キャンセル(G40)されるまで、補正量が保持されます。
- 工具径補正において、H指令は無視され、D指令のみ有効となります。
- 工具番号に「0」を指定すると、工具径補正指令がキャンセルされます。
G42 D0 (工具番号0で、工具径補正キャンセル)
- 工具径補正は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“モーダル初期選択”⇒“平面選択”にて指定した平面内において行われます。指定した平面に含まれる軸以外は、補正を行いません。
- 補正量の値を正(+), または負(-)に設定した場合、動作は下記ようになります。



- 工具径補正についての詳細は、工具径補正を参照してください。(P216ページ 工具径補正)

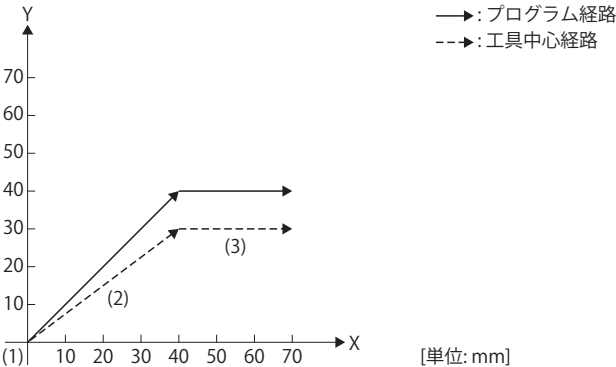
プログラム例

■工具径補正(右方向)を開始し、位置決めを行うプログラム

工具径補正量は「補正番号: 10」, 「補正量: 10[mm]」の場合

動作	プログラム	備考 ^{*1}
(1)	G90 X0. Y0.	絶対値指令により「X0, Y0」へ移動
(2)	G91 G01 G42 X40. Y40. D10 F500.	工具径補正開始動作で「X40, Y30」に移動
(3)	G01 X30.	「X70」に直線で移動

^{*1} 工具中心位置について説明しています。



G40.1: 法線制御 キャンセル

設定した法線制御(G41.1, G42.1)をキャンセルします。

コード	フォーマット
G40.1	G40.1 X x Y y F f <div><div></div>送り速度 <div></div>座標指令</div>

機能

- G40.1指令は、設定した法線制御(G41.1, G42.1)をキャンセルして指定の位置決めを行います。
- 法線制御についての詳細は、法線制御機能を参照してください。(242ページ 法線制御機能)

プログラム例

■法線制御をキャンセルするプログラム

動作	プログラム	備考
(1)	G41.1	法線制御左オン
(2)	G91 G01 X100. Y50. F500.	旋回軸が旋回後、「X100, Y50」に直線で移動
(3)	G40.1	法線制御キャンセル
(4)	X50. Y50.	旋回軸の位置を保持し、「X50, Y50」に直線で移動

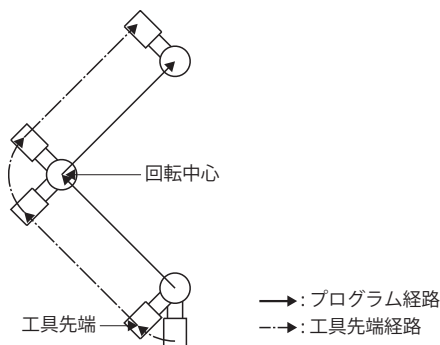
G41.1: 法線制御 左オン

平面選択中の軸の移動に対して、工具が常に法線方向に向くように回転軸を旋回制御します。

コード	フォーマット
G41.1	G41.1 \square X x \square Y y \square F f <div style="display: flex; justify-content: center; align-items: center; gap: 20px;"> <div style="border-top: 1px solid black; width: 100px; margin-left: 100px;"></div> <div style="border-top: 1px solid black; width: 100px; margin-left: 100px;"></div> <div style="border-top: 1px solid black; width: 100px; margin-left: 100px;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; gap: 100px; margin-top: -10px;"> <div></div> <div>送り速度</div> <div>座標指令</div> </div>

機能

- 平面選択中の軸移動に対して、法線制御を行います。(G17: 平面 X-Y軸, G18: 平面 Z-X軸, G19: 平面 Y-Z 軸)



- G41.1指令はモーダル指令です。法線制御キャンセル(G40.1)されるまで、法線制御が保持されます。
- 法線制御で旋回制御する軸は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“法線制御”⇒“法線制御軸”にて軸名称を設定します。法線制御軸に「0: 法線制御しない」が設定されている場合、G41.1指令を実行すると、軽度エラー (エラーコード: 1FC3H(詳細コード: 031BH))となり、法線制御を行いません。
- 法線制御中の法線制御軸旋回方式により、法線制御タイプⅠ、法線制御タイプⅡの2種類があります。[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“法線制御”⇒“法線制御タイプ”にて設定します。
- 法線制御についての詳細は、法線制御機能を参照してください。(P.242ページ 法線制御機能)

プログラム例

■法線制御(左側)を行うプログラム

動作	プログラム	備考
(1)	G41.1	法線制御左オン
(2)	G91 G01 X-50. Y50. F500.	旋回軸が旋回後、「X-50, Y50」に直線で移動
(3)	X50. Y50.	旋回軸が旋回後、「X50, Y50」に直線で移動

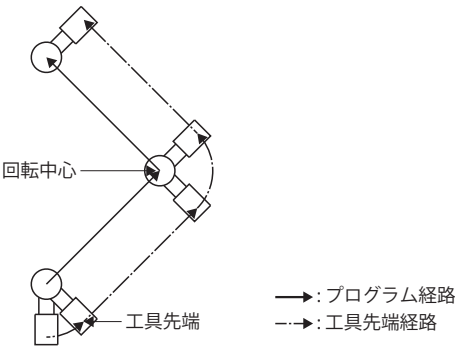
G42.1: 法線制御 右オン

平面選択中の軸の移動に対して、工具が常に法線方向に向くように回転軸を旋回制御します。

コード	フォーマット
G42.1	G42.1 <u> </u> X <u> </u> x <u> </u> Y <u> </u> y <u> </u> F <u> </u> f
	<div>送り速度</div> <div>座標指令</div>

機能

- 平面選択中の軸移動に対して、法線制御を行います。(G17: 平面 X-Y軸, G18: 平面 Z-X軸, G19: 平面 Y-Z 軸)



- G42.1指令はモーダル指令です。法線制御キャンセル(G40.1)されるまで、法線制御が保持されます。
- 法線制御で旋回制御する軸は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“法線制御”⇒“法線制御軸”にて軸名称を設定します。法線制御軸に「0: 法線制御しない」が設定されている場合、G42.1指令を実行すると、軽度エラー (エラーコード: 1FC3H(詳細コード: 031BH))となり、法線制御を行いません。
- 法線制御中の法線制御軸旋回方式により、法線制御タイプⅠ、法線制御タイプⅡの2種類があります。[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“法線制御”⇒“法線制御タイプ”にて設定します。
- 法線制御についての詳細は、法線制御機能を参照してください。(242ページ 法線制御機能)

プログラム例

■法線制御(右側)を行うプログラム

動作	プログラム	備考
(1)	G42.1	法線制御右オン
(2)	G91 G01 X50. Y50. F500.	旋回軸が旋回後、「X50, Y50」に直線で移動
(3)	X-50. Y50.	旋回軸が旋回後、「X-50, Y50」に直線で移動

G43: 工具長補正(+)

移動指令に設定した補正量を加えて移動します。工具長の値と実際のズレを補正量として設定することにより工具長を意識せず、プログラムを作成できます。

コード	フォーマット
G43	G43 <u> </u> Z <u> </u> z <u> </u> H <u> </u> h <div style="display: inline-block; vertical-align: middle; margin-left: 100px;"> — 工具番号 — 座標指令 </div>

機能

- G43指令を実行すると、移動指令の終点位置を[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御加工パラメータ]⇒“工具補正データ”⇒“長補正量”にて設定した補正量を加えた位置に移動します。
- G43指令はモーダル指令です。工具長補正キャンセル(G49)されるまで、補正量が保持されます。
- 工具番号に「0」を指定すると、工具長補正指令がキャンセルされます。ただし、同一ブロックに軸アドレスを指定した場合は、指定した軸アドレスの工具長補正のみキャンセルされます。
G43 H0 (工具番号0で、工具長補正キャンセル)
- G43指令は、下記の計算式により移動量の計算を行います。

プログラム	Z軸移動量	動作
G43 Z10. H01	10+(工具番号01の工具長補正量)	工具補正量だけ+方向に補正

- 工具長補正モード中に再度G43指令を行った場合は、補正番号の差分量だけの補正を行います。

プログラム	Z軸移動量
G43 Z10. H01	10+(工具番号01の工具長補正量)
⋮	⋮
G43 Z10. H02	10-(工具番号02の工具長補正量-工具番号01の工具長補正量)

- 機械座標系への移動指令(G53指令)のときは、工具補正量をキャンセルした状態で機械位置に移動します。ワーク座標系(G54～G59)に戻ったとき、再び工具補正量だけ加えた位置に戻ります。
- Gコード制御移行時、おおよりセット実行後は、G49(工具長補正キャンセル)モードとなります。
- 工具長補正は、G43と同一ブロックに指令される軸アドレスに対して有効になります。ただし、有効となる軸アドレスは、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“平面構成”⇒“基本軸I～K”にて指定された軸アドレスのみとなります。(以降、基本軸Iに「X」、基本軸Jに「Y」、基本軸Kに「Z」を指定した場合としています。)
- G43と同一ブロックに軸アドレスの指定がない場合は、Z軸に対して有効となります。ただし、基本軸Kを設定していない場合は、工具長補正を行いません。
- 工具長補正は、1軸のみ有効な指令です。2軸以上同時に指令した場合、優先順位は「Z>Y>X」となります。

例

下記プログラムで有効となる補正対象軸

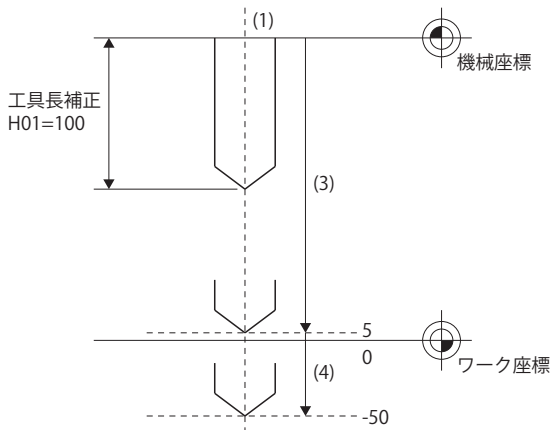
プログラム	補正対象軸
G43 X10. H01	X軸が有効となる。
G43 X10. Y10. Z10. H01	Z軸が有効となる。
G43 H01	Z軸が有効となる。

プログラム例

■指令位置に補正量を加えて位置決めするプログラム(絶対値指令時)

工具長補正量は「補正番号01」,「補正量: 100」の場合

動作	プログラム	備考
(1)	G90 G53 Z0.	機械座標系のZ軸「0」の位置へ移動
(2)	G90	絶対値指令
(3)	G43 Z5. H01	工具番号01の工具長補正量を使用し, Z軸「5」の位置へ移動
(4)	G01 Z-50. F500	Z軸「-50」の位置へ移動



G44: 工具長補正(-)

移動指令に設定した補正量を引いて移動します。工具長の値と実際のズレを補正量として設定することにより工具長を意識せず、プログラムを作成できます。

コード	フォーマット
G44	G44 <u> </u> Z <u> </u> z <u> </u> H <u> </u> h <div style="margin-left: 150px;"> 工具番号 座標指令 </div>

機能

- G44指令を実行すると、移動指令の終点位置を[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御加工パラメータ]⇒“工具補正データ”⇒“長補正量”にて設定した補正量を引いた位置に移動します。
- G44指令はモーダル指令です。工具長補正キャンセル(G49)されるまで、補正量が保持されます。
- 工具番号に「0」を指定すると、工具長補正指令がキャンセルされます。ただし、同一ブロックに軸アドレスを指定した場合は、指定した軸アドレスの工具長補正のみキャンセルされます。
G44 H0 (工具番号0で、工具長補正キャンセル)
- G44指令は、下記の計算式により移動量の計算を行います。

プログラム	Z軸移動量	動作
G44 Z10. H01	10-(工具番号01の工具長補正量)	工具補正量だけ-方向に補正

- 工具長補正モード中に再度G44指令を行った場合は、補正番号の差分量だけの補正を行います。

プログラム	Z軸移動量
G44 Z10. H01	10-(工具番号01の工具長補正量)
⋮	⋮
G44 Z10. H02	10+(工具番号02の工具長補正量-工具番号01の工具長補正量)

- 機械座標系への移動指令(G53指令)のときは、工具補正量をキャンセルした状態で機械位置に移動します。ワーク座標系(G54～G59)に戻ったとき、再び工具補正量だけ引いた位置に戻ります。
- Gコード制御移行時、およびリセット実行後は、G49(工具長補正キャンセル)モードとなります。
- 工具長補正は、G44と同一ブロックに指令される軸アドレスに対して有効になります。ただし、有効となる軸アドレスは、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“平面構成”⇒“基本軸I～K”にて指定された軸アドレスのみとなります。(以降、基本軸Iに「X」、基本軸Jに「Y」、基本軸Kに「Z」を指定した場合としています。)
- G44と同一ブロックに軸アドレスの指定がない場合は、Z軸に対して有効となります。ただし、基本軸Kを設定していない場合は、工具長補正を行いません。
- 工具長補正は、1軸のみ有効な指令です。2軸以上同時に指令した場合、優先順位は「Z>Y>X」となります。

例

下記プログラムで有効となる補正対象軸

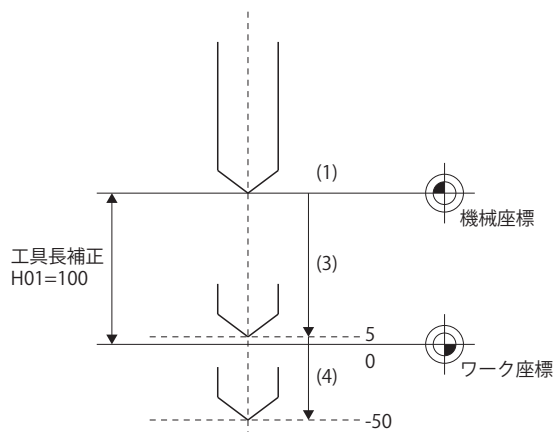
プログラム	補正対象軸
G44 X10. H01	X軸が有効となる。
G44 X10. Y10. Z10. H01	Z軸が有効となる。
G44 H01	Z軸が有効となる。

プログラム例

■指令位置に補正量を引いて位置決めするプログラム(絶対値指令時)

工具長補正量は「補正番号01」,「補正量: 100」の場合

動作	プログラム	備考
(1)	G90 G53 Z0.	機械座標系のZ軸「0」の位置へ移動
(2)	G90	絶対値指令
(3)	G44 Z5. H01	工具番号01の工具長補正量を使用し, Z軸「5」の位置へ移動
(4)	G01 Z-50. F500	Z軸「-50」の位置へ移動



G49: 工具長補正キャンセル

設定した工具長補正量(G43, G44)をキャンセルします。

コード	フォーマット
G49	G49 <u> </u> Z <u> </u> z └── 座標指令

機能

- G49指令は、設定した工具長補正量(G43, G44)をキャンセルして指定の位置決めを行います。
- G49指令は、下記の計算式により移動量の計算を行います。

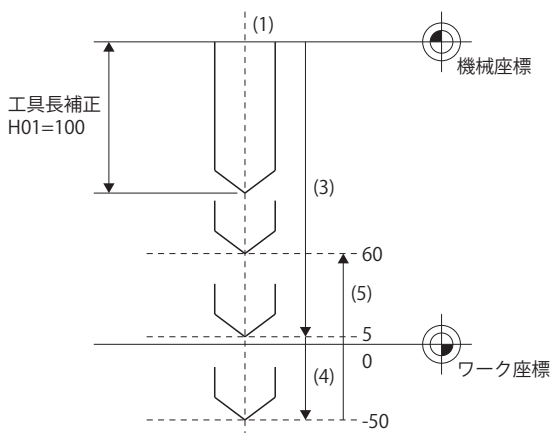
プログラム	Z軸移動量	動作
G49 Z10.	10-(+)(工具番号01の工具長補正量)	工具補正量キャンセル

プログラム例

■工具長補正による位置決め実行後、補正量をキャンセルして指定の位置決めを行うプログラム(絶対値指令時)

工具長補正量は「補正番号01」,「補正量:100」の場合

動作	プログラム	備考
(1)	G90 G53 Z0.	機械座標系のZ軸「0」の位置へ移動
(2)	G90	絶対値指令
(3)	G43 Z5. H01	工具番号01の工具長補正量を使用し、Z軸「5」の位置へ移動
(4)	G01 Z-50. F500	Z軸「-50」の位置へ移動
(5)	G49 Z40.	Z軸「60」の位置へ移動



G52: ローカル座標系設定

選択したワーク座標系上にローカル座標系オフセットの位置を設定します。

コード	フォーマット
G52	G54_ X x _Y y _Z z └─┬─┘ G59 ワーク座標系上のローカル座標オフセットの位置

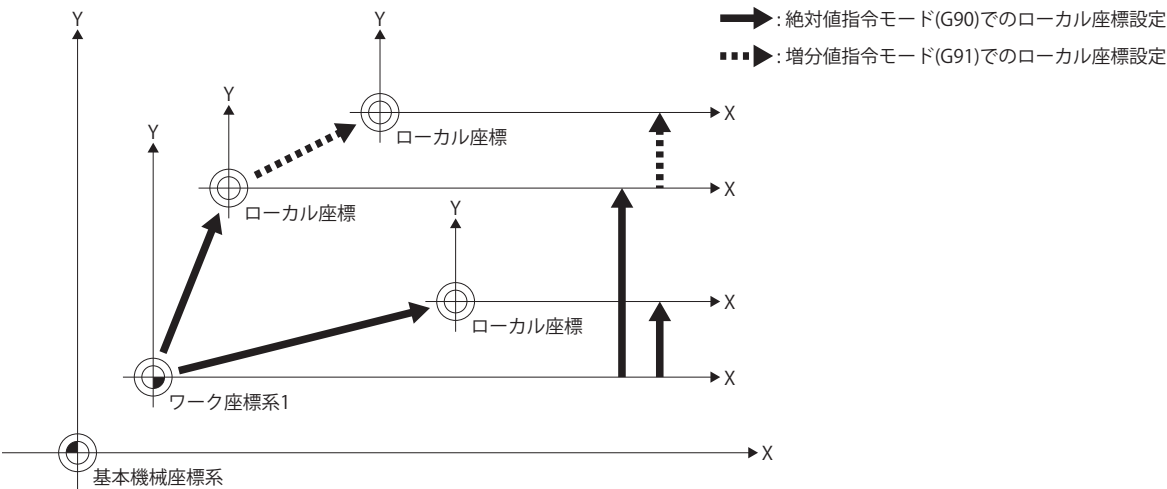
機能

- ローカル座標系は、ワーク座標系(G54～G59)上に指令した位置がプログラム原点となるように設定できる座標系です。加工プログラム原点と加工ワーク原点とのずれを変更する場合に使用できます。
- G52指令はアンモーダル指令ですが、次のG52指令が実行されるまで有効です。
- G52指令は座標系の移動は行いません。ワーク座標系(G54～G59)の原点位置を変更せず、さらに別の座標系を使用できます。
- ローカル座標系は、Gコード制御移行時、リセット時、またはローカル座標系設定を行った座標に対して、絶対値指令で「0」を指令したときにキャンセルされます。

例

ワーク座標系1(G54)に絶対値指令で「0」を指令した場合
G90 G54 G52 X0 Y0 Z0

- G52指令は、絶対値指令モード(G90)のときは、現在選択されているワーク座標系の原点位置から絶対位置でローカル座標系を設定します。増分値指令モード(G91)のときは、現在のローカル座標系の原点位置から相対位置で新たにローカル座標系を設定します。



- G52指令では、軸は移動しません。
- 絶対値指令(G90)での座標指令は、ローカル座標系の位置への移動となります。
- “[Md.3149]相対位置(D54772+32sn, D54773+32sn)”に“[Md.3154]ローカル座標オフセット(D54756+32sn, D54757+32sn)”を減算した位置がローカル座標系での位置となります。
- ローカル座標系と各機能を組合せた場合の動作を下記に示します。

機能	動作
工具径補正	工具径補正指令(G41, G42)のブロック、および工具径補正キャンセル指令(G40)のブロックと同一ブロックにG52を指令した場合、軽度エラー (エラーコード: 1FC3H(詳細コード: 030FH))となります。
工具長補正	工具長補正指令(G43, G44)のブロック、および工具長補正キャンセル指令(G49)のブロックと同一ブロックにG52を指令した場合、軽度エラー (エラーコード: 1FC3H(詳細コード: 030FH))となります。
法線制御	法線制御中はローカル座標系の変更はできません。変更を行うと軽度エラー (エラーコード: 1FC3H(詳細コード: 0304H))となります。

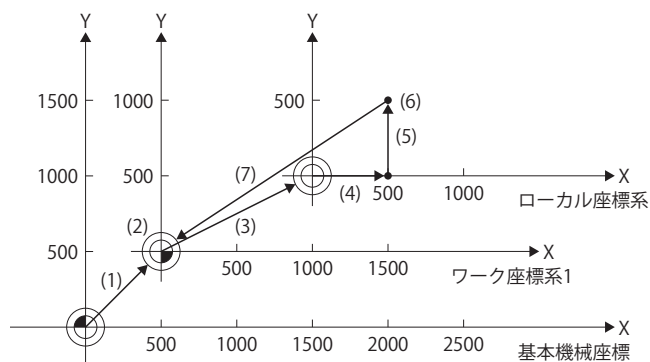
プログラム例

■絶対値モードでワーク座標系のローカル座標系オフセットの位置から指定位置へ位置決めを行うプログラム

- ワーク座標系のオフセット設定量

アドレス	オフセット設定量
	ワーク座標系1(G54)
X	500
Y	500

動作	プログラム	備考
(1)	G90 G54 X0. Y0. F100.	絶対値指令によりワーク座標系1の「X0, Y0」へ移動
(2)	G52 X1000. Y500.	ローカル座標系オフセット「X1000, Y500」を設定(軸移動なし)
(3)	G00 X0. Y0.	ローカル座標系の「X0, Y0」へ移動
(4)	G01 X500.	ローカル座標系の「X500, Y0」へ移動
(5)	Y500.	ローカル座標系の「X500, Y500」へ移動
(6)	G52 X0. Y0.	ローカル座標系オフセットをキャンセル(軸移動なし)
(7)	G00 X0. Y0.	ワーク座標系1の「X0, Y0」へ移動



[単位: mm]

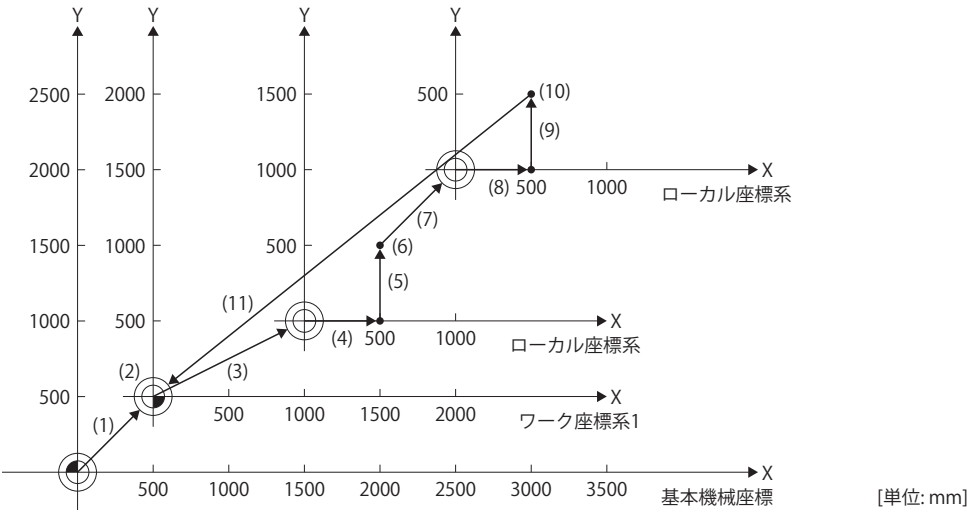
- ローカル座標は(2)によって作成されます。
- (6)にてローカル座標がキャンセルされ、(1)の座標系と一致します。

■増分値指令でワーク座標系のローカル座標系オフセットの位置から指定位置へ位置決めを行うプログラム

• ワーク座標系のオフセット設定量

アドレス	オフセット設定量
	ワーク座標系1(G54)
X	500
Y	500

動作	プログラム	備考
(1)	G90 G54 X0. Y0. F100.	絶対値指令によりワーク座標系1の「X0, Y0」へ移動
(2)	G91 G52 X1000. Y500.	増分値指令によりローカル座標系オフセット「X1000, Y500」を設定(軸移動なし)
(3)	G90 G00 X0. Y0.	絶対値指令によりローカル座標系の「X0, Y0」へ移動
(4)	G01 X500.	ローカル座標系の「X500, Y0」に移動
(5)	Y500.	ローカル座標系の「X500, Y500」に移動
(6)	G91 G52 X1000. Y1000.	増分値指令によりローカル座標系オフセット「X1000, Y1000」を設定(軸移動なし)
(7)	G90 G00 X0. Y0.	絶対値指令によりローカル座標系の「X0, Y0」へ移動
(8)	G01 X500.	ローカル座標系の「X500, Y0」へ移動
(9)	Y500.	ローカル座標系の「X500, Y500」へ移動
(10)	G91 G52 X-2000. Y-2000.	増分値指令によりローカル座標系オフセット「X-2000, Y-2000」を設定(軸移動なし)
(11)	G90 G00 X0. Y0.	絶対値指令によりローカル座標系(ワーク座標1)の「X0, Y0」へ移動



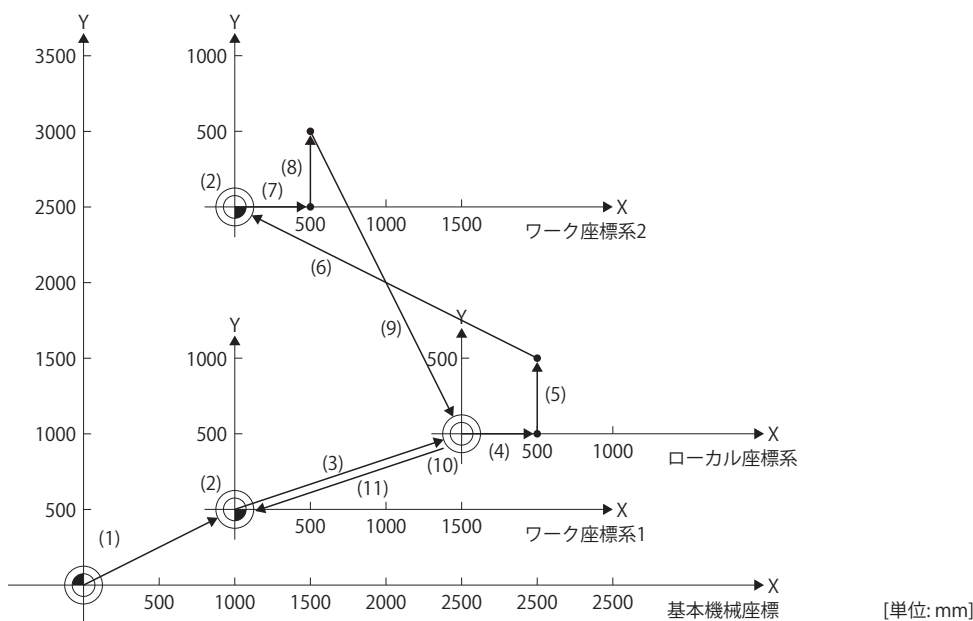
- (1)により、「500, 500」の位置にワーク座標系1が作成されます。
- (2)により、「X1000, Y500」の位置にローカル座標系を作成します。
- (6)により、「X2000, Y1500」の位置にローカル座標系を作成します。
- (10)により、「X500, Y500」の位置にローカル座標系を作成します。(ローカル座標系とワーク座標系1が一致し、ローカル座標系をキャンセルしたことになります。)

■複数のワーク座標系を併用してローカル座標系オフセットの位置から指定位置へ位置決めを行うプログラム

- ワーク座標系のオフセット設定量

アドレス	オフセット設定量	
	ワーク座標系1(G54)	ワーク座標系2(G55)
X	1000	1000
Y	500	2500

動作	プログラム	備考
(1)	G90 G54 X0. Y0. F100.	絶対値指令によりワーク座標系1の「X0, Y0」へ移動
(2)	G52 X1500. Y500.	ローカル座標系オフセット「X1500, Y500」を設定(軸移動なし)
(3)	G00 X0. Y0.	ローカル座標系の「X0, Y0」へ移動
(4)	G01 X500.	ローカル座標系の「X500, Y0」へ移動
(5)	Y500.	ローカル座標系の「X500, Y500」へ移動
(6)	G55 G00 X0. Y0.	ワーク座標系2の「X0, Y0」へ移動
(7)	G01 X500.	ワーク座標系2の「X500, Y0」へ移動
(8)	Y500.	ワーク座標系2の「X500, Y500」へ移動
(9)	G54 G00 X0. Y0.	ワーク座標系1のローカル座標系の「X0, Y0」へ移動
(10)	G52 X0. Y0.	ローカル座標系オフセットをキャンセル(軸移動なし)
(11)	G00 X0. Y0.	ワーク座標系1の「X0, Y0」へ移動



- (2)により、ワーク座標系1の「X1500, Y500」の位置にローカル座標系を作成します。(ワーク座標系2には、ローカル座標系は作成されません。)
- (9)により、ワーク座標系1のローカル座標系の「X0, Y0」へ移動します。
- (10)により、ローカル座標オフセットをキャンセルし、(1)の座標系と一致します。

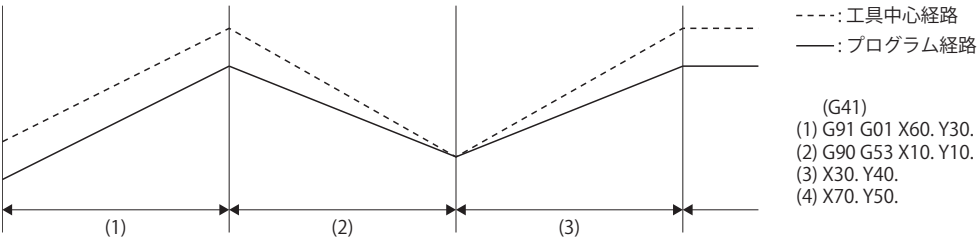
G53: 基本機械座標系選択

基本機械座標系上の指令位置へ移動します。

コード	フォーマット
G53	G53 <u> X x Y y Z z</u> 基本機械座標系上の座標

機能

- 基本機械座標系とは、機械固有に決められた位置を示す座標系です。
- G53指令は、アンモーダル指令です。指定したブロックでのみ有効です。
- G53指令は、増分値指令モード(G91)のときは、選択中の座標系を基準に増分値で移動します。基本機械座標系での位置に移動させる場合は、絶対値指令モード(G90)にて行ってください。前ブロックまで、増分値指令モード(G91)で動作させていた場合は、G53指令の前に絶対値指令モード(G90)を設定してください。
- G53指令は、指令モーダル(G00, G01)に従い、切削送り、または早送りで移動します。G53指令と円弧補間指令(G02, G03)を同一ブロックで指令した場合、終点座標値まで直線補間されますが、モーダルは円弧モーダルとなります。
- G53指令を実行した場合、当該ブロックにおいて工具径の補正ベクトルはなくなりますが、以降のポイントで指令された軸の工具径の補正量はキャンセルされません。

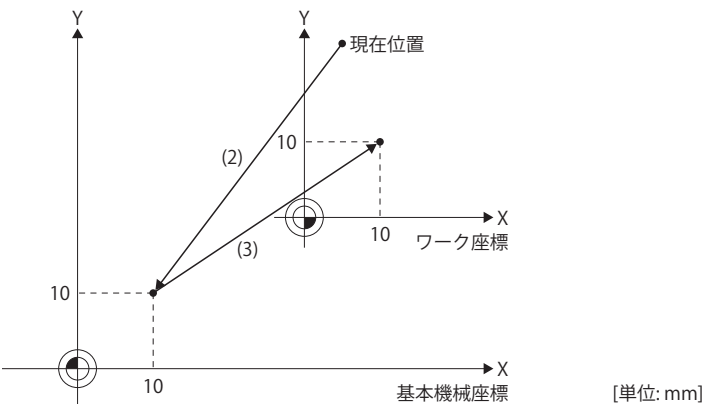


- G53指令ブロックの座標指令は、前のブロックが減速停止完了後に移動を開始します。
- 工具長補正(G43, G44)で設定された工具長補正量は、一時キャンセルされます。

プログラム例

■絶対値モードで基本機械座標系の指定位置へ位置決め後、ワーク座標系の指定位置に位置決めするプログラム

動作	プログラム	備考
(1)	G90	絶対値指令
(2)	G53 X10. Y10.	基本機械座標のX10, Y10へ移動
(3)	G01 X10. Y10. F20.	ワーク座標のX10, Y10へ移動



G54～G59: ワーク座標系1選択～ワーク座標系6選択

ワーク座標系を選択し、ワーク座標系の指定位置へ指定された送り速度で移動します。

コード	フォーマット
G54, G55, G56, G57, G58, G59	$G54 \text{ } \underline{\text{X } x \text{ } Y y \text{ } Z z}$ <div style="display: flex; align-items: center; margin-top: 5px;"> 指定ワーク座標の位置づけ位置 </div>

機能

- ワーク座標系は、プログラムする際に使用する座標系で、ワーク座標系1～6(G54～G59)のワーク座標系があります。ワーク座標系1～6により、現在選択されているワーク座標系において、工具の現在位置が指令された座標値になるようにワーク座標系が再設定されます。(工具の現在位置は、工具径、工具長および工具位置の補正量を含みます。)
- Gコード制御移行時は、ワーク座標系1(G54)が選択されます。
- ワーク座標系1～6選択はモーダル指令です。次のワーク座標系1～6選択指令が指令されるまで有効となります。
- ワーク座標系のオフセット設定量は、基本機械座標系原点からの距離で設定します。ワーク座標系のオフセット設定量は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御加工パラメータ]⇒“ワーク座標オフセット”にて設定します。
- Gコード制御移行時は、パラメータの設定に従った基本機械座標系、およびワーク座標系が自動座標系設定により自動的に設定されます。

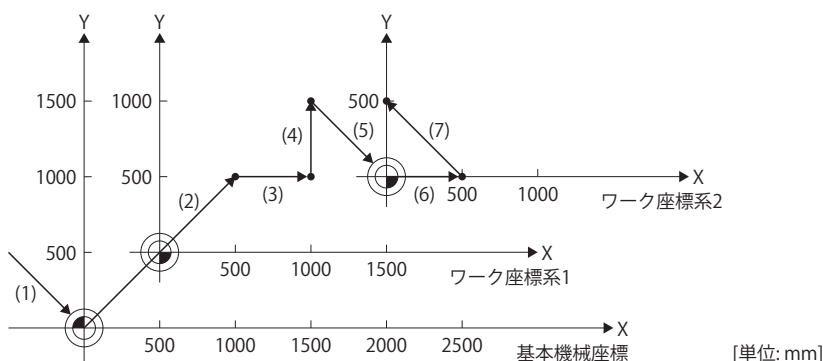
プログラム例

■複数のワーク座標系の指定位置へ位置決めを行うプログラム

- ワーク座標系のオフセット設定量

アドレス	オフセット設定量	
	ワーク座標系1(G54)	ワーク座標系2(G55)
X	500	2000
Y	500	1000

動作	プログラム	備考
(1)	G90 G00 G53 X0. Y0.	絶対値指令により基本機械座標系の「X0, Y0」へ移動
(2)	G54 X500. Y500. F100.	ワーク座標系1の「X500, Y500」へ移動
(3)	G01 G91 X500. F100.	現在位置から増分値指令により「X500」移動
(4)	Y500.	現在位置から増分値指令により「Y500」移動
(5)	G90 G00 G55 X0. Y0.	絶対値指令によりワーク座標系2の「X0, Y0」へ移動
(6)	G01 X500.	ワーク座標系2の「X500, Y0」へ移動
(7)	X0. Y500.	ワーク座標系2の「X0, Y500」へ移動

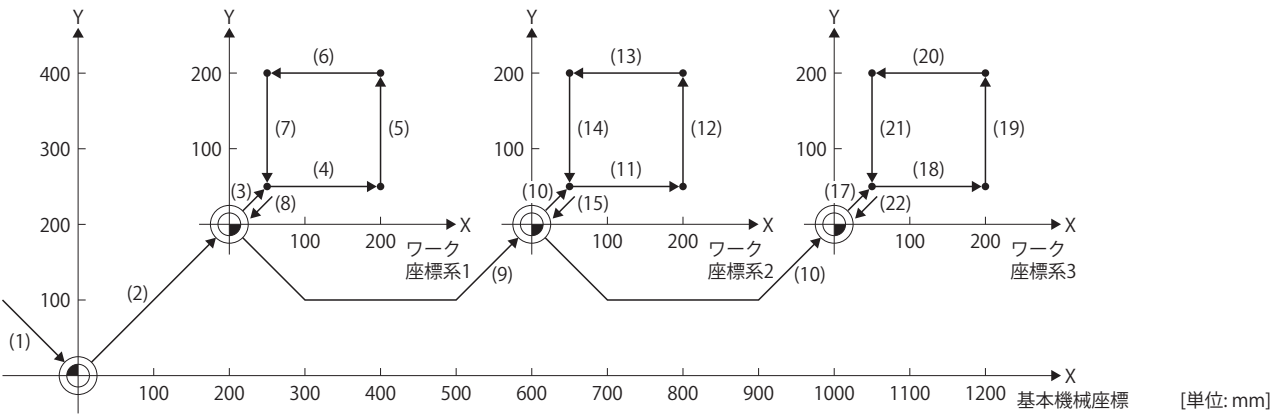


■G54～G56の座標上に同一のワークを3個置き、位置決めを行うプログラム

・ワーク座標系のオフセット設定量

アドレス	オフセット設定量		
	ワーク座標系1(G54)	ワーク座標系2(G55)	ワーク座標系3(G56)
X	200	600	1000
Y	200	200	200

動作	プログラム	備考
(1)	G90 G00 G53 X0. Y0.	絶対値指令により基本機械座標系の「X0, Y0」へ移動
(2)	G90 G54 X0. Y0. F50.	ワーク座標系1の「X0, Y0」へ移動
(3)	G90 G01 X50. Y50. F50.	現在位置から絶対値指令により「X50, Y50」へ移動
(4)	X200.	「X200, Y50」へ移動
(5)	Y200.	「X200, Y200」へ移動
(6)	X50.	「X50, Y200」へ移動
(7)	Y50.	「X50, Y50」へ移動
(8)	X0. Y0.	「X0, Y0」へ移動
(9)	G90 G00 G55 X0. Y0.	ワーク座標系2の「X0, Y0」へ移動
(10)	G90 G01 X50. Y50. F50.	現在位置から絶対値指令により「X50, Y50」へ移動
(11)	X200.	「X200, Y50」へ移動
(12)	Y200.	「X200, Y200」へ移動
(13)	X50.	「X50, Y200」へ移動
(14)	Y50.	「X50, Y50」へ移動
(15)	X0. Y0.	「X0, Y0」へ移動
(16)	G90 G00 G56 X0. Y0.	ワーク座標系3の「X0, Y0」へ移動
(17)	G90 G01 X50. Y50. F50.	現在位置から絶対値指令により「X50, Y50」へ移動
(18)	X200.	「X200, Y50」へ移動
(19)	Y200.	「X200, Y200」へ移動
(20)	X50.	「X50, Y200」へ移動
(21)	Y50.	「X50, Y50」へ移動
(22)	X0. Y0.	「X0, Y0」へ移動



G61: イグザクトストップチェックモード

減速チェックを行います。

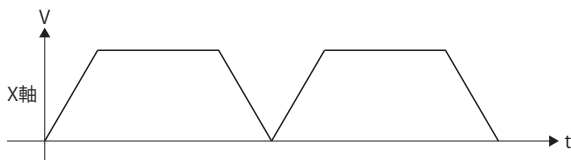
コード	フォーマット
G61	G61

機能

- G61指令は、切削指令のブロックにおいて、すべての終点で減速チェックを行います。
- G61指令は指定座標ごとに減速チェックを行ってから、次ブロックを実行します。
- G61指令はモーダル指令です。同一グループの高精度制御モード(G61.1)、自動コーナオーバーライド(G62)、切削モード(G64)のいずれかが指令されるまで有効となります。
- 減速チェックについての詳細は、減速チェックを参照してください。(P211ページ 減速チェック)

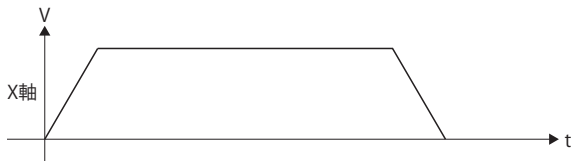
■イグザクトストップチェックモード中の場合

G61 G01 X100.F500.
X200.



■イグザクトストップチェックモードでない場合

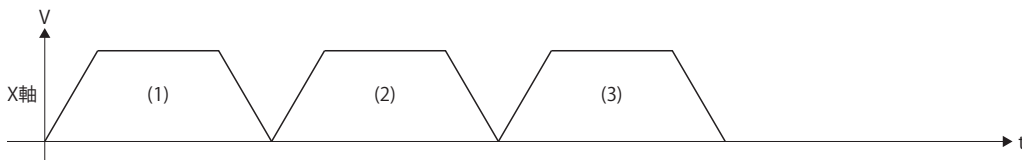
G01 X100.F500.
X200.



プログラム例

■イグザクトストップチェックモードにより、位置決めするプログラム

動作	プログラム	備考
(1)	G61 G01 X100.F500.	イグザクトストップチェックモードによる位置決め
(2)	X200.	イグザクトストップチェックモードによる位置決め
(3)	X300.	イグザクトストップチェックモードによる位置決め



G61.1: 高精度制御モード

制御系の遅れに起因する加工誤差を抑制します。

コード	フォーマット
G61.1	G61.1

機能

- G61.1指令はモーダル指令です。同一グループのイグザクトストップチェックモード(G61)、自動コーナオーバーライド(G62)、切削モード(G64)のいずれかが指令されるまで有効となります。
- 高精度制御モード中の早送り速度は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“高精度制御”⇒“高精度制御モード用早送り速度”にて設定した早送り速度が有効です。高精度制御モード用早送り速度に「0」が設定されている場合は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“速度・時定数”⇒“早送り速度”にて設定した早送り速度が有効です。
高精度制御モード中、G00指令は時間一定加減速方式(補間後加減速)で加減速処理します。
- 高精度制御モード中の送り速度指令(F)は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“高精度制御”⇒“高精度制御モード用切削送りクランプ速度”にて設定した送り速度でクランプします。高精度制御モード用切削送りクランプ速度に「0」が設定されている場合は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“速度・時定数”⇒“削送りクランプ速度”にて設定した送り速度でクランプします。
高精度制御モード中、G01指令、G02指令、G03指令は傾き一定加減速方式(補間前加減速)で加減速処理します。
- 他の同一グループのモード(G61、G62、G64)から高精度制御モード(G61.1)に切替わるとき、または高精度制御モード(G61.1)から他の同一グループのモード(G61、G62、G64)に切替わるときは、減速チェックを行います。
- Gコード制御移行時、およびリセット時は、切削モード(G64)になります。
- 高精度制御の詳細については、高精度制御を参照してください。(P255ページ 高精度制御)

プログラム例

■切削モードと高精度制御モードを切替えて位置決めするプログラム

動作	プログラム	備考
(1)	G64 G91 G01 X100. F1000.	切削モードによる位置決め
(2)	G61.1 X100.	高精度制御モードによる位置決め
(3)	Y100.	高精度制御モードによる位置決め
(4)	G64 X100.	切削モードによる位置決め

G62: 自動コーナオーバーライド

工具径補正において、内側コーナ切削時、または自動コーナRの内側切削時の工具負荷の軽減を、自動的に送り速度にオーバーライドをかけることにより行います。

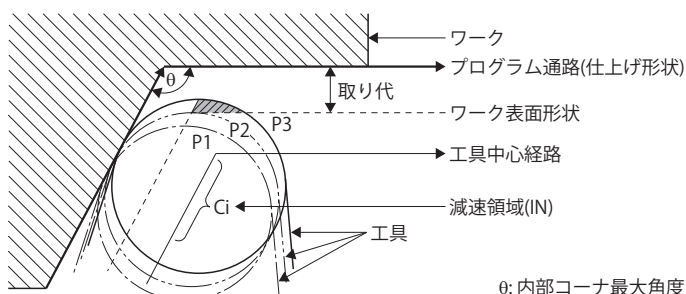
コード	フォーマット
G62	G62

機能

- G62指令は、補間指令と複合で使用し、実行すると、自動コーナオーバーライドを行います。G62指令と複合できる補間指令コードは、G01、G02、G03のみです。G00を指定した場合は、無効となります。また、コーナにおいて、G00からG01/G02/G03へ、またはG01/G02/G03からG00へ切り換わる場合、そのコーナでG00のブロックは自動コーナオーバーライドがかかりません。
- G62指令はモーダル指令です。同一グループのイグザクトストップチェックモード(G61)、高精度制御モード(G61.1)、切削モード(G64)のいずれかが指令されるまで有効となります。
- 下記の場合、自動コーナオーバーライドはかかりません。
 - 自動コーナオーバーライドモード中でも、工具径補正モードになっていないとき。
 - 工具径補正の開始、またはキャンセルがあるコーナのとき。
 - 工具径補正の、Jベクトル指令があるコーナのとき。
 - 交点演算が行えないとき。(移動指令ブロックが4回以上不連続の場合)
- 円弧指令時の減速領域は弧の長さとなります。
- Gコード制御システムパラメータの設定で、内側コーナの角度は、プログラム通路上での角度です。下記のようにパラメータが設定されている場合は、自動コーナオーバーライドが無効となります。
 - Gコード制御システム/パラメータの「自動コーナオーバーライド」の設定が「0」、または「100」のとき。
 - Gコード制御システム/パラメータの「自動コーナオーバーライド最大角度」の設定が「0」、または「180」のとき。
 - Gコード制御システム/パラメータの「自動コーナオーバーライドコーナ前長さ」の設定が「0」のとき。

■内側コーナの場合

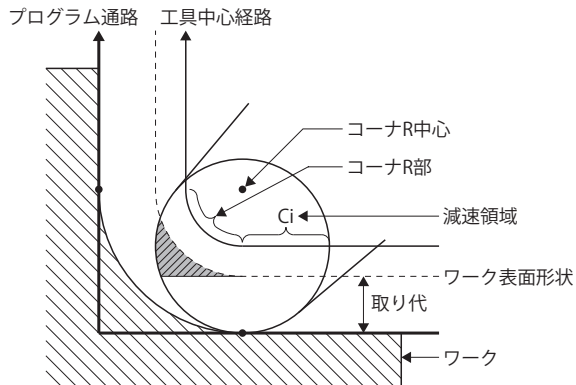
内側コーナを切削する場合、取り代の量が多くなり工具にかかる負荷が増加します。そのため、コーナの設定された範囲内で自動的にオーバーライドをかけて送り速度を下げ、負荷の増加を抑えて良好な切削が行えるようにします。



- G62 指令なしの場合
上図で工具がP1→P2→P3の順に移動するとき、P3はP2より斜線の面積だけ取り代が多いため、工具の負荷が増加します。
- G62指令ありの場合
上図で内側コーナの角度 θ がパラメータに設定された角度以下の場合、減速領域Ciで自動的にパラメータで設定されたオーバーライドがかかります。

■自動コーナRの場合

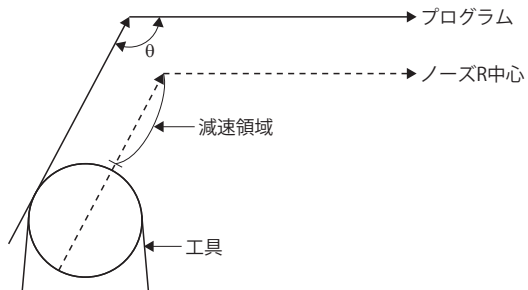
自動コーナRで内側補正のとき、減速領域(Ci)とコーナR部で、自動的にパラメータに設定されたオーバーライドがかかります。(角度のチェックはしません。)



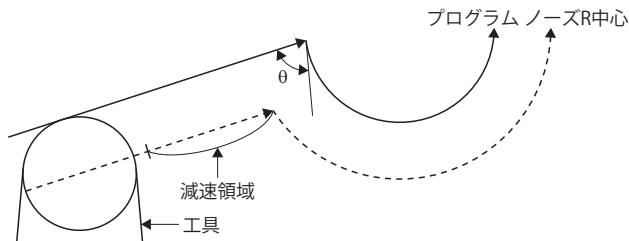
■実行例

自動コーナオーバーライドの実行例を以下に示します。減速領域でパラメータに設定されたオーバーライドがかかります。

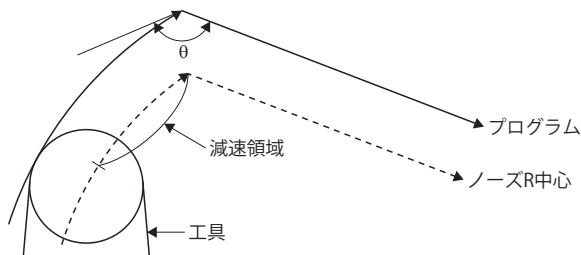
- 直線→直線コーナ



- 直線→円弧(外側補正)コーナ

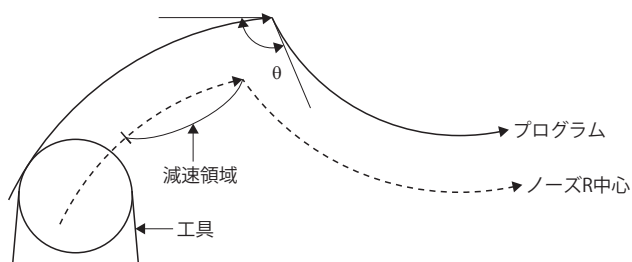


- 円弧(外側補正)→直線コーナ



*: オーバライドがかかる減速領域は、円弧指令のときは弧の長さとなります。

- 円弧(内側補正)→円弧(外側補正)コーナ



*: オーバライドがかかる減速領域は、円弧指令のときは弧の長さとなります。

■各機能との組合せによる自動コーナオーバーライドの動作

各機能による自動コーナオーバーライドの動作を下記に示します。

機能	自動コーナオーバーライド動作
位置決め(G00)	位置決め指令では、自動コーナオーバーライドはかかりません。
直線補間(G01)	直線補間は、自動コーナオーバーライドがかかります。
円弧補間(G02, G03)	円弧補間は、自動コーナオーバーライドがかかります。
切削送りオーバーライド	自動コーナオーバーライドに切削送りオーバーライドがかかります。
オーバーライドキャンセル	オーバーライドキャンセルでは、自動コーナオーバーライドはキャンセルされません。
速度クランプ	切削送り速度に自動コーナオーバーライドがかかった後でクランプ速度になります。
工具径補正	工具径補正モードになるまでは、自動コーナオーバーライドはかかりません。

プログラム例

■自動コーナオーバーライドを行うプログラム

動作	プログラム	備考
(1)	G91 G01 G42 X10. Y10. D10 F2000.	位置決め
(2)	G62	自動コーナオーバーライドモード
(3)	G01 X20. Y40.	位置決め
(4)	X50.	位置決め

G64: 切削モード

切削送りブロック間で減速停止せず連続して次のブロックを実行します。

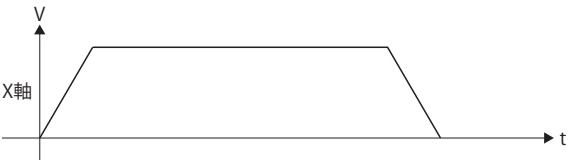
コード	フォーマット
G64	G64

機能

- G64指令は、イグザクトストップチェックモード(G61)とは逆に、切削送りブロック間で減速チェックを行わずに連続してブロックを実行します。G64指令と複合できる補間指令コードは、G01、G02、G03のみです。
- マルチCPUシステムの電源投入時は、切削モードとなります。
- G64指令はモーダル指令です。同一グループのイグザクトストップチェックモード(G61)、高精度制御モード(G61.1)、自動コーナオーバーライド(G62)のいずれかが指令されるまで有効となります。

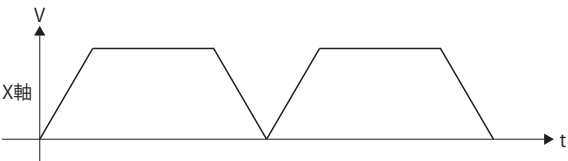
■切削モード中の場合

G64 G01 X100. F500.
X200.



■切削モードでない場合

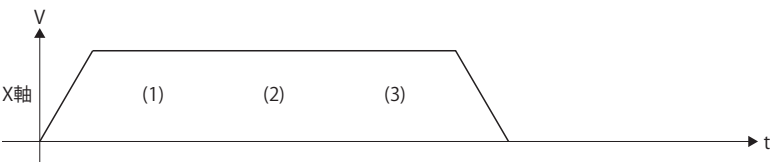
G61 G01 X100. F500.
X200.



プログラム例

■切削モードにより、位置決めするプログラム

動作	プログラム	備考
(1)	G64 G01 X100. F500.	切削モードによる位置決め
(2)	X200.	切削モードによる位置決め
(3)	X300.	切削モードによる位置決め

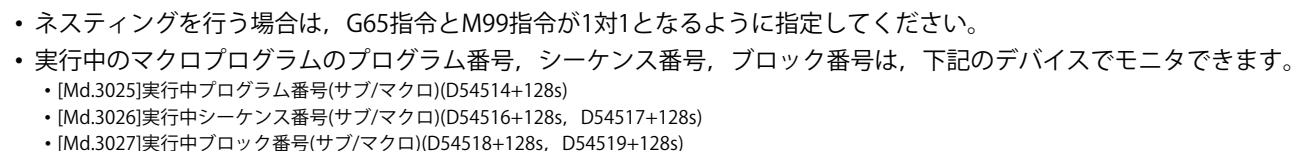


5

コード	フォーマット
G65	<p>G65 □ P p □ L l □ A a</p> <p>□ 引数(変数データ指定)</p> <p>□ マクロプログラム繰り返し回数(0~9999)</p> <p>□ マクロプログラム呼出しプログラム番号(1~256)</p>

- 製品の形状が同じで寸法のみ異なる場合に、メインのプログラムとは別にマクロプログラムを作成し、同じプログラムを流用して呼出せます。
- マクロプログラムは必要なときにメインプログラム、またはサブプログラムより呼出して使用できます。
- マクロプログラムを呼出すごとに引数をローカル変数としてプログラムに使用できます。
- マクロプログラムからの復帰はM99指令で行います。
- マクロプログラム呼出しプログラム番号、マクロプログラム繰り返し回数、引数の動作を下記に示します。

- マクロプログラムから他のマクロプログラムを呼出すことができます。これをネスティングと呼びます。マクロプログラムの呼出し深さ(ネスティング深さ)は、最大4重まで可能です。



■引数指定I

- 引数指定Iで使用可能なアドレスと、マクロプログラム内での変数番号の対応表を下記に示します。

○:使用可, ×:使用不可, —:アドレスなし

引数指定Iのアドレス	マクロプログラム内の変数	マクロ呼出し使用可否
A	#1	○
B	#2	○
C	#3	○
D	#7	○
E	#8	○
F	#9	○
—	#10	×
H	#11	○
I	#4	○
J	#5	○
K	#6	○
—	#12	×
M	#13	○
—	#14	×
—	#15	×
—	#16	×
Q	#17	○
R	#18	○
S	#19	○
T	#20	○
U	#21	○
V	#22	○
W	#23	○
X	#24	○
Y	#25	○
Z	#26	○

例

引数指定Iを指定したプログラムの場合

メインプログラム

```
%
:
G65 P100 A10. M20. Z30.
:
M02
%
```

マクロ呼出し

マクロプログラム

```
%
G90 G00 X#1
      Y#13
      Z#26
M99
%
```

→ X10.
→ Y20.
→ Z30.

- 引数指定Iは、「A～Z」のアドレスを引数に指定できます。ただし、「G, L, N, O, P」のアドレスは下記のように扱うため、引数に指定できません。


アドレス	内容
G	Gコード指令
L	G65指令フォーマット
N	シーケンス番号
O	指令しても無視される
P	G65指令フォーマット

- 「I, J, K」のアドレスはアルファベット順に指定してください。ただし、「I, J, K」以外のアドレスはアルファベット順に指定する必要はありません。

例

下記プログラムを指定した場合

プログラム	変数データ		内容
	ローカル変数(アドレス)	受渡すデータ	
G65 P100 I40. J50. K60.	#4(I)	40.	引数指定Iのアドレスとして扱われます。
	#5(J)	50.	
	#6(K)	60.	
G65 P100 A10. I40. B20. J50. C30. K60.	#1(A)	10.	
	#4(I)	40.	
	#2(B)	20.	
	#5(J)	50.	
	#3(C)	30.	
	#6(K)	60.	
G65 P100 J50. I40. K60.	#5(J)	50.	引数指定Ⅱのアドレス(1組目(J)と2組目(I, K))として扱われるため、目的の変数に値が設定されません。
	#7(I)	40.	
	#9(K)	60.	
G65 P100 A10. J50. B20. I40. C30. K60.	#1(A)	10.	
	#5(J)	50.	
	#2(B)	20.	
	#7(I)	40.	
	#3(C)	30.	
	#9(K)	60.	

- 指定する必要のないアドレスは省略できます。
- アドレスの後に指定する値は、数値以外に変数も使用できます。変数についての詳細は、変数指令を参照してください。
( 173ページ 変数指令)

■引数指定Ⅱ

- ・ 引数指定Ⅱで使用可能なアドレスと、マクロプログラム内での変数番号の対応表を下記に示します。

引数指定Ⅱのアドレス		マクロプログラム内の変数
A		#1
B		#2
C		#3
I	1組目	#4
J		#5
K		#6
I	2組目	#7
J		#8
K		#9
I	3組目	#10
J		#11
K		#12
I	4組目	#13
J		#14
K		#15
I	5組目	#16
J		#17
K		#18
I	6組目	#19
J		#20
K		#21
I	7組目	#22
J		#23
K		#24
I	8組目	#25
J		#26
K		#27
I	9組目	#28
J		#29
K		#30
I	10組目	#31
J		#32
K		#33

例

引数指定Ⅱを指定したプログラムの場合

メインプログラム

```
%
:
G65 P100 I10. J20. K30. I-10. J-20. K-30.
:
M02
%
```

マクロ呼出し

マクロプログラム

```
%
G90 G00 X#4
      Y#5
      Z#6
G91 G00 X#7
      Y#8
      Z#9
M99
%
```

→ X10.
→ Y20.
→ Z30.
→ X-10.
→ Y-20.
→ Z-30.


- ・ 引数指定Ⅱは、「A, B, C」のアドレスと「I, J, K」のアドレスを1組とする引数を最大10組まで設定できます。引数の組数が10組を超えた場合、軽度エラー (エラーコード: 1FC3H(詳細コード: 0336H))となります。

- 同じアドレス(I, J, K)が重複するときは、定められた順番に指定してください。順番に指定しないと、引数の組数の順序がずれてしまい、目的の変数に値が設定されません。

例

下記プログラムを指定した場合

プログラム	変数データ		内容
	ローカル変数(アドレス)	受渡すデータ	
G65 P100 I10. J20. K30. I40. J50. K60.	#4(I)	10.	1組目(I, J, K), 2組目(I, J, K)のアドレスとして扱われます。
	#5(J)	20.	
	#6(K)	30.	
	#7(I)	40.	
	#8(J)	50.	
	#9(K)	60.	
G65 P100 I10. J20. K30. I40. K60. J50.	#4(I)	10.	1組目(I, J, K), 2組目(I, K), 3組目(J)のアドレスとして扱われます。
	#5(J)	20.	
	#6(K)	30.	
	#7(I)	40.	
	#9(K)	60.	
	#11(J)	50.	

- アドレスの後に指定する値は、数値以外に変数も使用できます。変数についての詳細は、変数指令を参照してください。
( 173ページ 変数指令)

■G65(マクロ呼出し)とM98(サブプログラム呼出し)の相違点

G65指令とM98指令の相違点を下記に示します。

項目		G65(マクロ呼出し)	M98(サブプログラム呼出し)
呼出し深さ(ネスティング)		最大4重	最大10重 ^{*1}
引数指定		指定可	指定不可
シーケンス番号指定		指定不可	指定可
同一ブロックに指令フォーマット以外のアドレスがあるとき	シングルブロック	停止する	停止する
	アドレス指令	G65指令前のアドレスを実行し、その後にマクロプログラム呼出し	実行後、サブプログラム呼出し
ローカル変数深さ		変動(呼出し深さに応じる) ^{*2}	固定

*1 G65指令と合わせて最大10重です。

*2 マクロ呼出しを行うと、呼出し深さ1重とみなし、ローカル変数の深さも1つ大きくなります。

注意事項

- モーダル情報はメインプログラム、サブプログラム、マクロプログラムの区別なく実行の順序に従って書換えられます。マクロ呼出しを実行した後は、モーダルデータの状態に注意してプログラムしてください。
- マクロプログラムでM02指令、M30指令を実行するとメインプログラム、サブプログラム、マクロプログラムが共にプログラムの実行を終了します。
- G65指令のブロックは、シングルブロックにより停止できます。
- マクロ呼出しの引数は、末尾に小数点が記述されているものとして扱います。

例

G65 P110 X100の場合

「X」のローカル変数(#24)の値は、「100.」となります。

- マクロ呼出しの引数に同一のアドレスを複数指定した場合、最後に記述したアドレスの指令が有効になります。

例

G65 P110 A1. A2.の場合

アドレスAは、「A2.」が引数として有効となります。

- ・マクロ呼出しで引数を指定する場合、アドレスはG65指令の後に指令してください。

例

X100. G65 P110の場合

X軸が100.の位置へ移動した後、マクロ呼出しを実行します。

- ・引数指定Ⅰと引数指定Ⅱを同一ブロックに指定し、同一の変数に対応するアドレスを指定した場合、最後に記述したアドレスの指令が有効となります。

例

G65 P100 A10. B20. D30. I40. I70.の場合

#7の変数に対して「D30.」と「I70.」の引数が指令された場合、最後の「I70.」が有効となります。

アドレス	変数	指定方法
A10.	#1	引数指定Ⅰ
B20.	#2	
D30.	#7	
I40.	#4	
I70.	#7	引数指定Ⅱ

- ・ローカル変数は、「[Rq.3376]Gコード制御要求(D54226.0+2s)」がONしたときに初期化(0クリア)します。

プログラム例

■指定したマクロプログラムを実行後、メインプログラムに復帰して動作完了するプログラム

メインプログラム
O100.gcd

```
%
:
:
:
G65 P200 A60. S100. F800
:
:
M02
%
```

サブプログラム
O200.gcd

```
%
G91
G01 X[#19*COS [#1]]
Y[#19*SIN [#1]] F#9
:
:
M99
%
```



*: ローカル変数
#1(A) =60.000
#9(F) =800
#19(S)=100.000

G68: プログラム座標回転モード開始

座標系に対して回転角度を指定し、回転した形状を位置決めします。

コード	フォーマット
G68	$G68 _X _x _Y _y _R _r$ <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="border-bottom: 1px solid black; width: 100px; margin-right: 10px;"></div> <div style="border-bottom: 1px solid black; width: 100px; margin-right: 10px;"></div> <div style="border-bottom: 1px solid black; width: 100px; margin-right: 10px;"></div> <div style="border-bottom: 1px solid black; width: 100px; margin-right: 10px;"></div> <div style="border-bottom: 1px solid black; width: 100px;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 10px;"> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black; margin-right: 10px;"></div> <div style="width: 100px; border-bottom: 1px solid black;"></div> </div>

- ・ G68指令直後は、基本的に回転平面内の2軸を絶対値で指令してください。座標回転後の初回移動指令で1軸のみ指令した場合、Gコード制御加工パラメータの「座標回転タイプ」の設定により、動作が異なります。

Gコードプログラム	0: 座標回転タイプ0	1: 座標回転タイプ1
(1) G90 G53 X0. Y0. (2) G68 X40. Y0. R90. (3) X50. (4) Y50. (5) G69 (6) M02	(3)で指令された軸(X軸)のみ移動します。 始点は座標回転に伴って回転せず、回転前のローカル座標系上の現在位置から終点位置を計算します。	(3)で「X50. Y0.」と指令した場合と同様の動作となります。 座標回転に伴って始点が回転すると仮定して終点位置を計算します。

- ・ G68指令直後は、基本的に回転平面内の2軸を絶対値の位置決め/直線補間で指令してください。座標回転後の初回移動指令で円弧補間を指令した場合は、移動が発生しない場合でもいったん、始点の位置への位置決め/直線補間での移動指令を行ってください。なお、座標回転の直後に円弧補間を指令した場合は、Gコード制御加工パラメータの「座標回転タイプ」の設定により、動作が異なります。

Gコードプログラム	0: 座標回転タイプ0	1: 座標回転タイプ1
(1) G90 G53 X0. Y0. (2) G68 X40. Y0. R90. (3) G03 X50. R-25. F500 (4) G00 Y50. (5) G69 (6) M02	円弧補間の始点は、座標回転に伴って回転せず、基本機械座標系から見て座標回転前と同じ位置になります。従って、この始点から終点へ円弧補間で動作します。	座標回転に伴って回転すると仮定した始点の位置と実際の軸の位置が、機械座標系から見て異なった位置になります。このため、始点から終点への円弧補間の軌跡が軸が移動できずに軽度エラー (エラーコード: 1FC3H(詳細コード: 0313H)) となります。

- ・ プログラム座標回転中の機械座標系選択は、Gコード制御加工パラメータの「座標回転タイプ」の設定により、動作を選択できます。

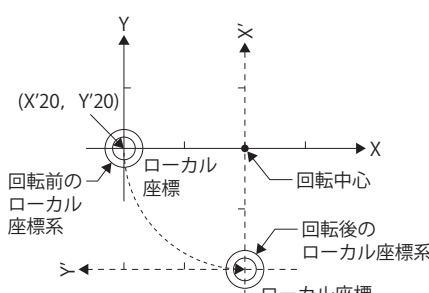
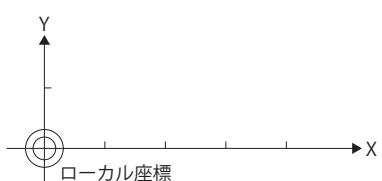
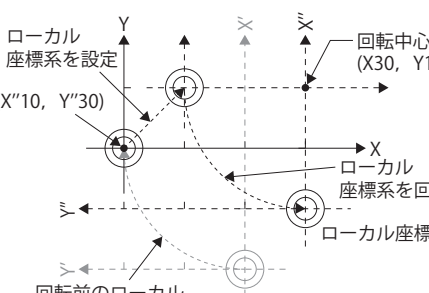
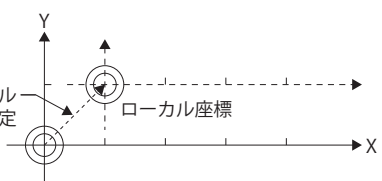
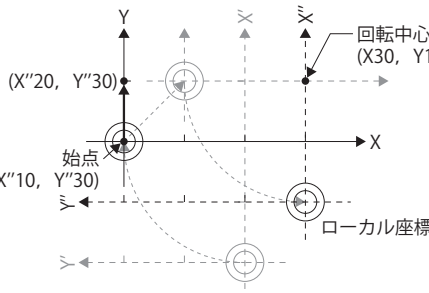
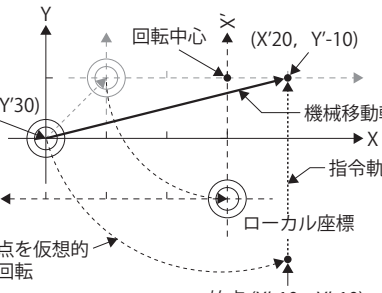
Gコードプログラム	0: 座標回転タイプ0	1: 座標回転タイプ1
(1) G90 G53 X0. Y0. (2) G68 X20. Y0. R90. (3) G53 X20. Y20. (4) M02	G53指令にて基本機械座標系での位置(X20, Y20)に移動します。 プログラム座標は、基本機械座標系での位置を座標回転させた位置(X'40, Y'0)(回転後のローカル座標系)となります。	G53指令にて座標回転を一時キャンセルし、基本機械座標系の位置(X20, Y20)に移動します。 プログラム座標は、座標回転が一時キャンセルとなるため、基本機械座標系と同じ位置(X20, Y20)となります。

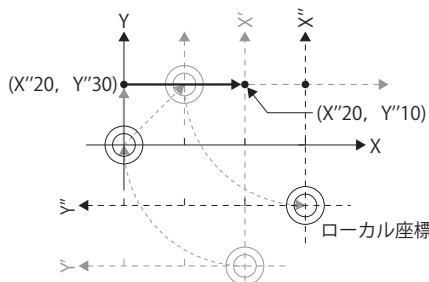
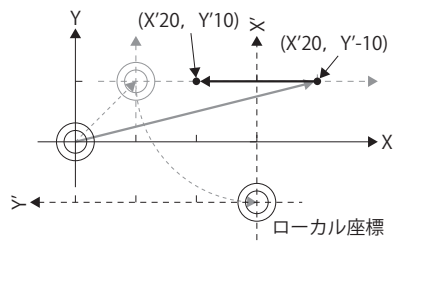
- ・プログラム座標回転中のローカル座標系設定は、Gコード制御加工パラメータの「座標回転タイプ」の設定により、動作を選択できます。

座標回転タイプ	動作
座標回転タイプ0	プログラム座標回転後の座標系上で指令された位置がローカル座標原点となります。ただし、補正の方向は回転前の座標系となります。
座標回転タイプ1	<ul style="list-style-type: none"> ・G68指令からG52指令までの間に平面選択軸の移動指令がない場合 プログラム座標回転前の座標系上で指令された位置がローカル座標原点となり、その座標系を回転します。ただし、補正の方向は回転前の座標系となります。 ・G68指令からG52指令までの間に平面選択軸の移動指令がある場合 プログラム座標回転後の座標系上で指令された位置がローカル座標原点となり、その座標系を回転します。ただし、補正の方向は回転前の座標系となります。

例

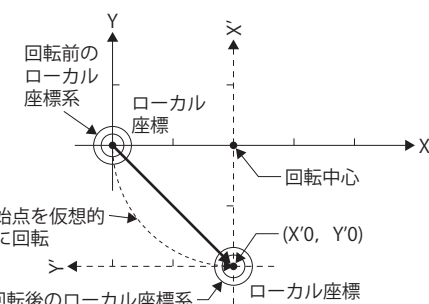
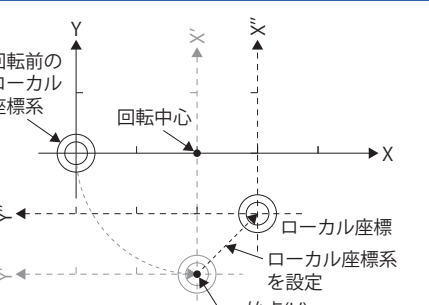
座標回転タイプ0と座標回転タイプ1(G68指令からG52指令までの間に平面選択軸の移動指令がない場合)を選択した場合

動作	プログラム	0: 座標回転タイプ0	1: 座標回転タイプ1(G68指令からG52指令までの間に平面選択軸の移動指令がない場合)
(1)	G90 G53 X0. Y0.	—	—
(2)	G68 X20. Y0. R90.	 <p>始点は座標回転に伴って回転せず、回転前のローカル座標系上の現在位置から計算します。</p>	 <p>ローカル座標系は回転しません。</p>
(3)	G52 X10. Y10.	 <ol style="list-style-type: none"> 1. 座標回転前のローカル座標系に戻します。 2. ワーク座標系上にローカル座標系を設定します。 X方向に10、Y方向に10シフトした位置がローカル座標原点となります。 3. ローカル座標系を回転します。回転中心座標は、2.で設定したローカル座標系となります。 	 <p>ワーク座標系上にローカル座標系を設定します。 *: 補正の方向は回転前の座標系となります。</p>
(4)	X20.	 <p>指令したX軸が回転座標系上で移動します。 *: 移動指令がない軸は、移動しません。</p>	 <p>座標回転に伴い始点を仮想的に回転させた後、指令したX軸が回転座標系上で移動します。 *: 移動指令がない軸は、回転座標系上の位置へ移動します。</p>

動作	プログラム	0: 座標回転タイプ0	1: 座標回転タイプ1(G68指令からG52指令までの間に平面選択軸の移動指令がない場合)
(5)	Y10.	 <p>指令したY軸が回転座標系上で移動します。 *: 移動指令がない軸は、移動しません。</p>	 <p>指令したY軸が回転座標系上で移動します。 *: 移動指令がない軸は、移動しません。</p>
(6)	G69	—	—
(7)	M02	—	—

例

座標回転タイプ1(G68指令からG52指令までの間に平面選択軸の移動指令がある場合)を選択した場合

動作	プログラム	1: 座標回転タイプ1(G68指令からG52指令までの間に平面選択軸の移動指令がある場合)
(1)	G90 G53 X0. Y0.	—
(2)	G68 X20. Y0. R90. X0. Y0.	 <p>座標回転に伴い始点を仮想的に回転させて、終点位置へ移動します。</p>
(3)	G52 X10. Y10.	 <p>回転後のローカル座標系上の原点位置から新たにローカル座標系を設定します。 *1: 補正の方向は回転前の座標系となります。 *2: 始点(L')の座標は、新たに設定した座標系(X''Y''平面)上のローカル座標原点を基準とした回転前の座標系(XY平面)上の座標となります。</p>

■各機能との組合せによるプログラム座標回転モードの動作

機能	動作
回転軸	回転軸に対しても座標回転が行えます。この場合、角度[degree]を長さ[mm]として回転処理を行います。
位置決め(早送り)	1軸のみのG00指令に対し座標回転を行うと、2軸が移動することになります。このとき、G00非補間の設定が「非補間タイプ」であれば、各軸がそれぞれの軸の早送り速度で独立して移動します。 始点から終点まで直線的に移動(補間)させる場合は、必ずG00非補間の設定を「補間タイプ」で使用してください。 このときの送り速度は、各軸の早送り速度の合成速度となりますので、1軸のみ移動する場合(座標回転前)よりも移動速度が速くなります。
ワーク座標系選択	座標回転モード中にワーク座標を切換えた場合(G54からG55に切換えなど)、プログラム座標回転の回転中心は、指令を行った座標系での位置となります。(基本座標系から見て同じ位置のままになります。)
プログラム目標位置	位置表示はすべて、座標回転後の位置を座標回転前のローカル座標系基準で表示します。
工具長補正	プログラム座標回転中は、工具長の補正量は機械座標に対して補正します。

プログラム例

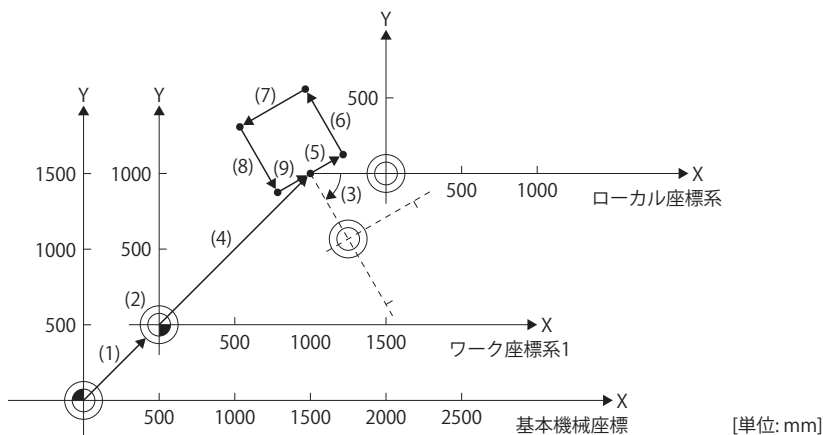
■プログラム座標回転モードにより、位置決めするプログラム

- ワーク座標系のオフセット設定量

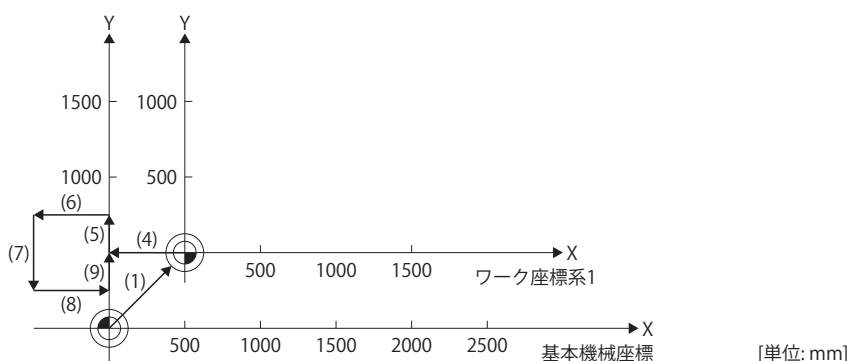
アドレス	オフセット設定量
	ワーク座標系1(G54)
X	500
Y	500

動作	プログラム	備考
(1)	G90 G54 X0. Y0. F100.	絶対値指令によりワーク座標系1の「X0, Y0」へ移動
(2)	G52 X1500. Y1000.	ローカル座標系オフセット「X1500, Y1000」を設定(軸移動なし)
(3)	G68 X-500. Y0. R-60.	ローカル座標系の「X-500, Y0」を回転中心とし、-60[degree]を反時計方向に座標回転
(4)	G01 X-500. Y0.	ローカル座標系の「X-500, Y0」を(3)の通り座標回転した位置へ移動
(5)	Y250.	ローカル座標系の「X-500, Y250」を(3)の通り座標回転した位置へ移動
(6)	X-1000.	ローカル座標系の「X-1000, Y250」を(3)の通り座標回転した位置へ移動
(7)	Y-250.	ローカル座標系の「X-1000, Y-250」を(3)の通り座標回転した位置へ移動
(8)	X-500.	ローカル座標系の「X-500, Y-250」を(3)の通り座標回転した位置へ移動
(9)	Y0.	ローカル座標系の「X-500, Y0」を(3)の通り座標回転した位置へ移動

- ローカル座標(2)とプログラム座標回転(3)を設定した場合の経路



- ローカル座標(2)とプログラム座標回転(3)を設定しない場合の経路



G69: プログラム座標回転モードキャンセル

設定したプログラム座標回転オフセットをキャンセルします。

コード	フォーマット
G69	G69

機能

- G69指令は、設定したプログラム座標回転モード開始(G68)をキャンセルしてプログラム座標回転モードを終了します。
- G69指令直後の移動指令は、必ず絶対値指令モードで指令してください。増分値指令モードで指令すると、軽度エラー（エラーコード: 1FC3H(詳細コード: 0327H)）となります。

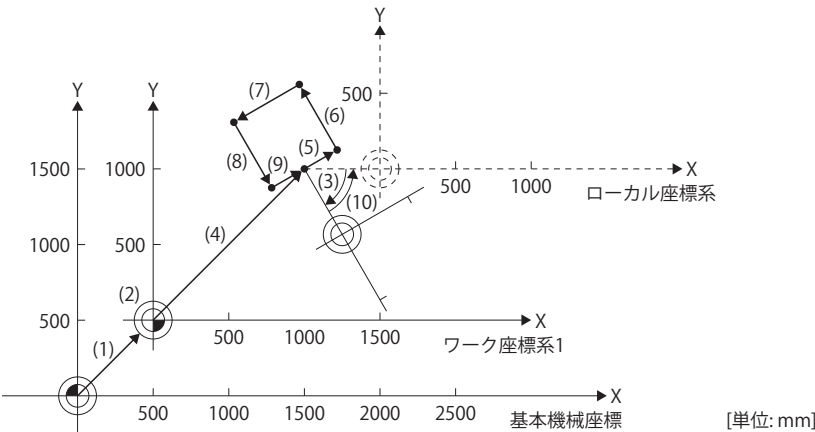
プログラム例

■プログラム座標回転モードをキャンセルするプログラム

- ワーク座標系のオフセット設定量

アドレス	オフセット設定量
	ワーク座標系1(G54)
X	500
Y	500

動作	プログラム	備考
(1)	G90 G54 X0. Y0. F100.	絶対値指令によりワーク座標系1の「X0, Y0」へ移動
(2)	G52 X1500. Y1000.	ローカル座標系オフセット「X1500, Y1000」を設定(軸移動なし)
(3)	G68 X-500. Y0. R-60.	ローカル座標系の「X-500, Y0」を回転中心とし、-60[degree]を反時計方向に座標回転
(4)	G01 X-500. Y0.	ローカル座標系の「X-500, Y0」を(3)の通り座標回転した位置へ移動
(5)	Y250.	ローカル座標系の「X-500, Y250」を(3)の通り座標回転した位置へ移動
(6)	X-1000.	ローカル座標系の「X-1000, Y250」を(3)の通り座標回転した位置へ移動
(7)	Y-250.	ローカル座標系の「X-1000, Y-250」を(3)の通り座標回転した位置へ移動
(8)	X-500.	ローカル座標系の「X-500, Y-250」を(3)の通り座標回転した位置へ移動
(9)	Y0.	ローカル座標系の「X-500, Y0」を(3)の通り座標回転した位置へ移動
(10)	G69	プログラム座標回転モードをキャンセル
(11)	G54 G52 X0. Y0.	ローカル座標系オフセットをキャンセル(軸移動なし)



G90: 絶対値指令

座標指令を絶対値指令として実行します。

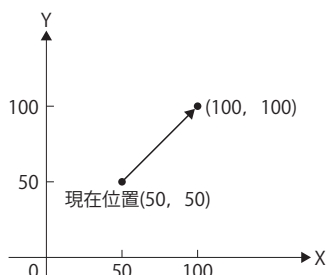
コード	フォーマット
G90	G90 <u>X x</u> <u>Y y</u> <u>Z z</u> 座標指令

機能

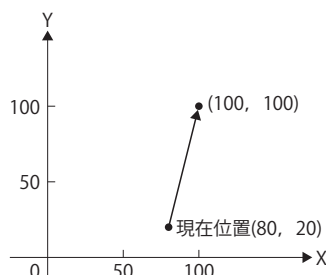
- 絶対値モードでは、現在の位置に関係なくプログラムにより指定された座標位置へ移動します。G90指令実行後に設定する位置決め指令は、原点座標からの絶対値で動作します。

例

G90 X100. Y100.



現在位置座標 X50, Y50のとき



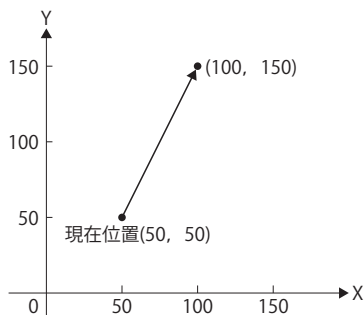
現在位置座標 X80, Y20のとき

[単位: mm]

- 同一ブロック内で、特定のアドレスだけを絶対値、または増分値として指令できます。ブロック内の最後に実行した指令がモーダルとして残ります。

例

G90 X100. G91 Y100.



現在位置座標 X50, Y50のとき

[単位: mm]

- G90指令はモーダル指令です。増分値指令モード(G91)が指令されるまで有効となります。
- Gコード制御開始時、またはリセット時のモードは、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御システムパラメータ]⇒“モーダル初期値”⇒“絶対値設定”にて設定します。

プログラム例

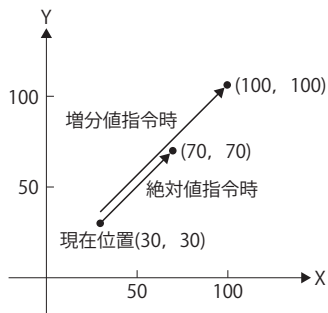
■絶対値指令時と増分値指令時の比較例

• 絶対値指令例

プログラム	備考
G90 X70. Y70.	絶対値指令により「X70, Y70」へ移動

• 増分値指令例

プログラム	備考
G91 X70. Y70.	増分値指令により「X100, Y100」へ移動



[単位: mm]

G91: 増分値指令

座標指令を増分値指令として実行します。

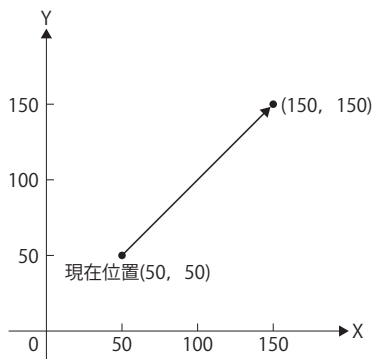
コード	フォーマット
G91	G91 <u>X x</u> <u>Y y</u> <u>Z z</u> 座標指令

機能

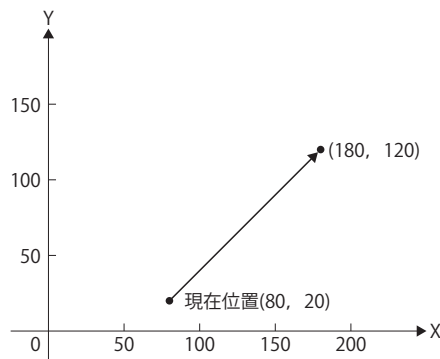
- 増分値モードでは、現在の位置を始点(0)として指定された値だけ相対値として移動します。G91指令実行後に設定する位置決め指令は、現在位置からの増分値で動作します。

例

G91 X100. Y100.



現在位置座標 X50, Y50のとき



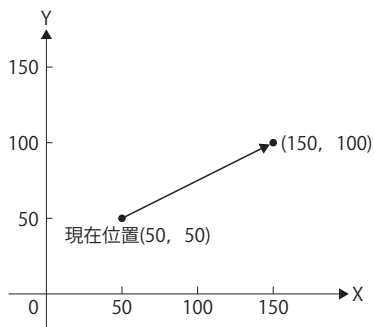
現在位置座標 X80, Y20のとき

[単位: mm]

- 同一ブロック内で、特定のアドレスだけを絶対値、または増分値として指令できます。ブロック内の最後に実行した指令がモーダルとして残ります。

例

G91 X100. G90 Y100.



現在位置座標 X50, Y50のとき

[単位: mm]

- G91指令はモーダル指令です。絶対値指令モード(G90)が指令されるまで有効となります。
- Gコード制御開始時、またはリセット時のモードは、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御システムパラメータ]⇒“モーダル初期値”⇒“絶対値設定”にて設定します。

プログラム例

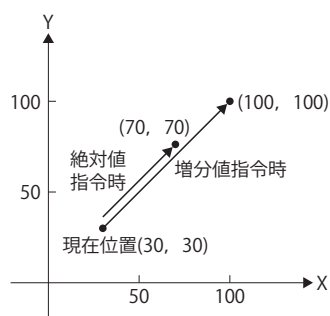
■増分値指令時と絶対値指令時の比較例

・増分値指令例

プログラム	備考
G91 X70. Y70.	増分値指令により「X100, Y100」へ移動

・絶対値指令例

プログラム	備考
G90 X70. Y70.	絶対値指令により「X70, Y70」へ移動



[単位: mm]

G94: 毎分送り(非同期送り)

ブロックからの指令を1分間当たりの送り速度[mm/min]として指令します。

コード	フォーマット
G94	G94

機能

- G94を指令することで、Fに続く数値を1分間当たりの送り速度[mm/min]で指令します。
- G94指令はモーダル指令です。Gコード制御移行時、およびリセット後はG94モードになります。

5.7 Mコード

アドレスMに続く数値8桁(M-99999999～M99999999)で指定し、1ブロック内に最大4組指令することができます。Mコードデータが指令されると“[Md.3058]Mコードデータ1(D54554+128s, D54555+128s)”～“[Md.3061]Mコードデータ4(D54560+128s, D54561+128s)”と“[St.3222]補助機能ストローブ1(D54442.4+4s)”～“[St.3225]補助機能ストローブ4(D54442.7+4s)”を出力します。

出力するMコードデータの出力方法には、下記の3種類があります。

[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“補助機能”⇒“Mバイナリ”にて設定します。

- BCDコード
- 符号なし32ビットバイナリデータ
- 符号付き32ビットバイナリデータ

Point

1ブロックに5組以上指令したときは、最後の4組が有効となります。

5

特定の補助機能(M機能)

- 下表のMコードデータは、特定の目的に使用するため、一般的な補助機能(M機能)として割当ててはできません。

補助機能(M機能)	内容	参照
M00	プログラムストップ	166ページ M00: プログラムストップ
M01	プログラムストップ	167ページ M01: プログラムストップ
M02	プログラムエンド	168ページ M02: プログラムエンド
M30	プログラムエンド	169ページ M30: プログラムエンド
M98	サブプログラム呼出し	170ページ M98: サブプログラム呼出し
M99	サブ/マクロプログラム復帰	172ページ M99: サブ/マクロプログラム復帰

- M98指令、およびM99指令を除くM指令は、“[Md.3058]Mコードデータ1(D54554+128s, D54555+128s)”～“[Md.3061]Mコードデータ4(D54560+128s, D54561+128s)”と“[St.3222]補助機能ストローブ1(D54442.4+4s)”～“[St.3225]補助機能ストローブ4(D54442.7+4s)”以外に“[St.3218]M単独出力 M00(D54442.0+4s)”～“[St.3221]M単独出力 M30(D54442.3+4s)”の単独信号も出力します。詳細は、M単独出力を参照してください。(172ページ M単独出力)
- M98指令、およびM99指令を除く、すべての補助機能(M機能)に対して、モーションSFCプログラムやシーケンスプログラムにて、処理および完了シーケンスを作成する必要があります。詳細は、補助機能完了を参照してください。(170ページ 補助機能完了)
- M00指令、M01指令、M02指令、M30指令は、先読み禁止のため、次のブロックは先読みバッファに読み込みません。

M00: プログラムストップ

プログラムの実行を停止します。

コード	フォーマット
M00	M00

機能

- M00指令を実行すると、次のブロックの実行を行わず、プログラムを停止します。停止後、“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”をONすることにより、次のブロックから実行を開始します。
- モーションSFCプログラム、またはシーケンスプログラムにより、プログラムストップ読み込み時、Gコードプログラムをブロック停止させずに継続して実行できます。詳細は、M単独出力を参照してください。(202ページ M単独出力)

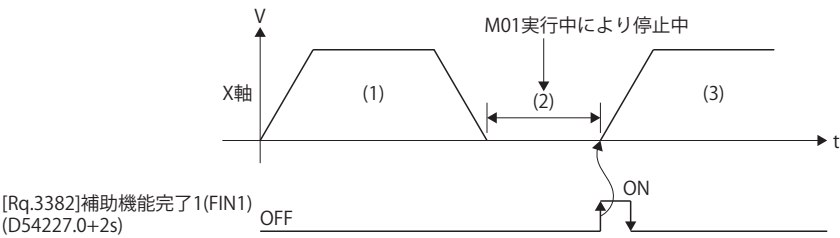
Point

M00とM01は、同一機能です。

プログラム例

■位置決め動作中にプログラムを停止し、再始動するプログラム

動作	プログラム	備考
(1)	G01 X100.F10.	位置決め
(2)	M00	プログラム停止
(3)	G01 X200.	補助機能完了1(FIN1)により、位置決め再開



[Rq.3382]補助機能完了1(FIN1)
(D54227.0+2s)

M01: プログラムストップ

プログラムの実行を停止します。

コード	フォーマット
M01	M01

機能

- M01指令を実行すると、次のブロックの実行を行わず、プログラムを停止します。停止後、“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”をONすることにより、次のブロックから実行を開始します。
- モーションSFCプログラム、またはシーケンスプログラムにより、プログラムストップ読み込み時、Gコードプログラムをブロック停止させずに継続して実行できます。詳細は、M単独出力を参照してください。(202ページ M単独出力)

Point

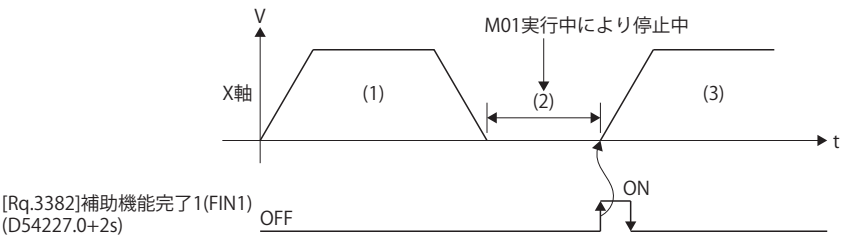
M01とM00は、同一機能です。

5

プログラム例

■位置決め動作中にプログラムを停止し、再始動するプログラム

動作	プログラム	備考
(1)	G01 X100.F10.	位置決め
(2)	M01	プログラム停止
(3)	G01 X200.	補助機能完了1(FIN1)により、位置決め再開



[Rq.3382]補助機能完了1(FIN1)
(D54227.0+2s)

M02: プログラムエンド

プログラムを終了します。

コード	フォーマット
M02	M02

機能

M02指令を実行すると、プログラムの実行を終了します。

プログラムの最終ブロックには、M02指令(またはM30指令)が必要です。

プログラム例

■位置決め完了後、プログラムを終了する

プログラム	備考
G90	絶対値指令
G01 X100. Y200. F100.	位置決め
X200. Y300.	位置決め
G00 X0. Y0.	位置決め
M02	プログラム終了
%	

Point

M02とM30は、同一機能です。

M30: プログラムエンド

プログラムを終了します。

コード	フォーマット
M30	M30

機能

M30指令を実行すると、プログラムの実行を終了します。
プログラムの最終ブロックには、M30指令(またはM02指令)が必要です。

プログラム例

■位置決め完了後、プログラムを終了する

プログラム	備考
G90	絶対値指令
G01 X100. Y200. F100.	位置決め
X200. Y300.	位置決め
G00 X0. Y0.	位置決め
M30	プログラム終了
%	

Point

M30とM02は、同一機能です。

M98: サブプログラム呼出し

指定したサブプログラムを呼出します。

コード	フォーマット
M98	M98 P p H h L l サブプログラム繰り返し回数(0~9999) サブプログラム呼出しシーケンス番号(0~99999) サブプログラム呼出しプログラム番号(1~256)

機能

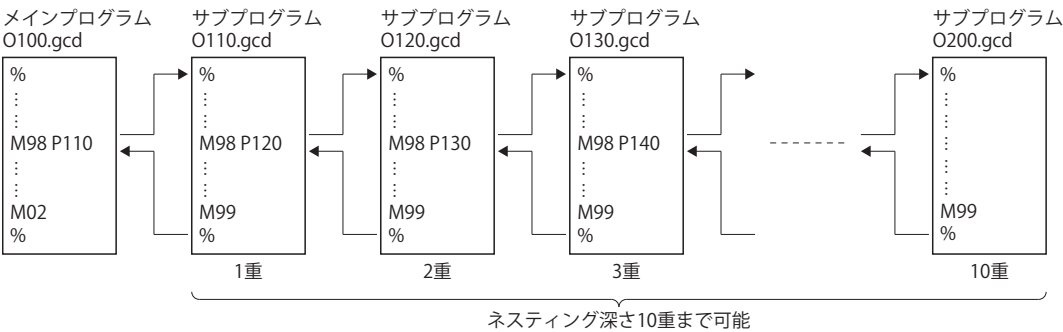
- ・ 繰り返し使用する運転パターンをサブプログラムとして作成し、必要に応じてメインプログラムより呼出します。
- ・ サブプログラム呼出しプログラム番号、サブプログラム呼出しシーケンス番号、サブプログラム繰り返し回数の動作を下記に示します。

項目	動作
サブプログラム呼出しプログラム番号	・ プログラム番号を省略すると、M98指令を実行したプログラム(自プログラム)を呼出します。 ・ 指定した番号のプログラムが存在しないと、軽度エラー (エラーコード: 1FC3H(詳細コード: 031FH))となります。
サブプログラム呼出しシーケンス番号	・ シーケンス番号を省略、または「0」を指定すると、M98指令で呼出したプログラムの先頭ブロックを呼出します。 ・ プログラム内に指定したシーケンス番号が存在しないと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0320H))となります。
サブプログラム繰り返し回数	・ 繰り返し回数を省略すると、プログラムを1回(L1を指令)繰り返しとなります。 ・ 繰り返し回数に「0」を指定した場合は、サブプログラムは実行しません。 ・ 範囲外の値を設定すると、軽度エラー (エラーコード: 1FC3H(詳細コード: 0328H))となります。

- ・ メインプログラムとサブプログラムとの違いを下記に示します。Gコードプログラムのフォーマットの詳細は、Gコードプログラムのフォーマットを参照してください。(90ページ Gコードプログラムのフォーマット)

プログラム	相違点
メインプログラム	最終ブロックにプログラムエンド指令(M02, またはM30)を単独ブロックとして記述します。
サブプログラム	最終ブロックにサブプログラム復帰指令(M99)を単独ブロックとして記述します。

- ・ サブプログラムから他のサブプログラムを呼出することができます。これをネスティングと呼びます。サブプログラムの呼出し深さ(ネスティング深さ)は、最大10重まで可能です。



- ・ ネスティングを行う場合は、M98指令とM99指令が1対1となるように指定してください。
- ・ サブプログラムからプログラムを呼出すときに、ネスティング深さが10重を超えた場合、軽度エラー (エラーコード: 1FC3H(詳細コード: 0329H))となります。
- ・ 実行中のサブプログラムのプログラム番号、シーケンス番号、ブロック番号は、下記のデバイスでモニタできます。
 - ・ [Md.3025]実行中プログラム番号(サブ/マクロ)(D54514+128s)
 - ・ [Md.3026]実行中シーケンス番号(サブ/マクロ)(D54516+128s, D54517+128s)
 - ・ [Md.3027]実行中ブロック番号(サブ/マクロ)(D54518+128s, D54519+128s)

注意事項

- ・ モーダル情報は、メインプログラム、サブプログラムの区別なく実行の順序に従って書換えられます。サブプログラム呼出しを実行した後は、モーダルデータの状態に注意してプログラムしてください。
- ・ 「M98 P□」が記述されているブロックは、シングルブロックにより停止しません。ただし、N, P, L, H以外のアドレスがあるときは、シングルブロックにより停止できます。

例

X100. M98 P100の場合

「X100.」を実行後、「100」のプログラム番号を呼出します。

- ・ サブプログラムでM02指令、またはM30指令を実行すると、メインプログラムとサブプログラムが共にプログラムの実行を終了します。
- ・ M98指令と同一ブロック上のアドレス(P, H, L)は、M98指令で指定したものとして扱います。

例

G04 P5 M98 H1の場合

「P5」はG04指令のドウェル時間として扱われず、M98指令のプログラム番号として扱われ、「G04 M98 P5 H1」の記述と同等になります。

プログラム例

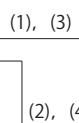
■指定したサブプログラムを2回繰り返し実行後、メインプログラムに復帰して動作完了するプログラム

メインプログラム
O100.gcd

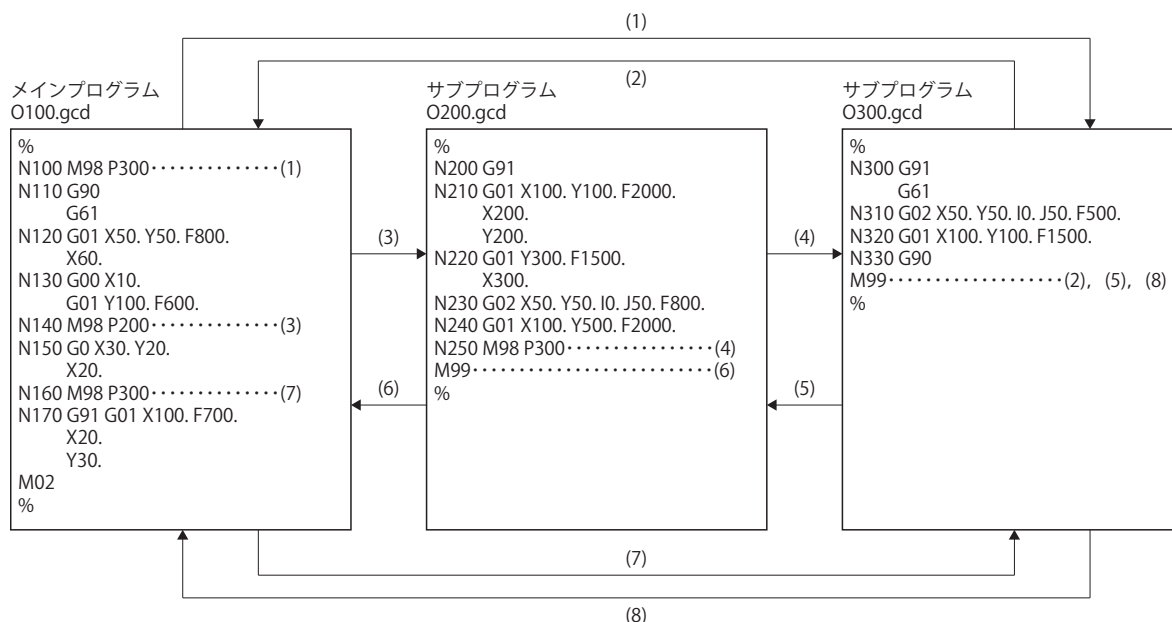
```
%
N100 G90
      G61
N110 G01 X50. Y50. F800.
      X60.
N120 M98 P200 H220 L2
N130 G00 X10.
      G01 Y100. F600.
N140 G0 X30. Y20.
      X20.
N150 G91 G01 X100. F700.
      X20.
      Y30.
M02
%
```

サブプログラム
O200.gcd

```
%
N200 G91
N210 G01 X100. Y100. F2000.
      X200.
      Y200.
N220 G01 Y300. F1500.
      X300.
N230 G02 X50. Y50. I0. J50. F800.
N240 G01 X100. Y500. F2000.
M99 P130
%
```



■サブプログラムからサブプログラムを呼び出すプログラム



M99: サブ/マクロプログラム復帰

サブプログラム、またはマクロプログラムからメインプログラムへ復帰します。

コード	フォーマット
M98	M99 <u> </u> P <u> </u> p └── 復帰先シーケンス番号(0～99999)

機能

- M98指令にて呼出したサブプログラム、またはG65指令にて呼出したマクロプログラムからメインプログラムの呼出しブロックの次ブロックへ復帰します。
- 復帰先シーケンス番号の動作を下記に示します。

項目	動作
復帰先シーケンス番号	<ul style="list-style-type: none">• シーケンス番号を省略、または「0」を指定すると、呼出しブロックの次ブロックへ復帰します。• プログラム内に指定したシーケンス番号が存在しないと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0320H))となります。

- ネスティングを行う場合は、M98指令、またはG65指令とM99指令が1対1となるように指定してください。
- ネスティング対象になる指令を下記に示します。
 - M98(サブプログラム呼出し)
 - G65(マクロ呼出し)

注意事項

- 「M99」が記述されているブロックは、シングルブロックにより停止しません。ただし、N、P、L、H以外のアドレスがあるときは、シングルブロックにより停止できます。

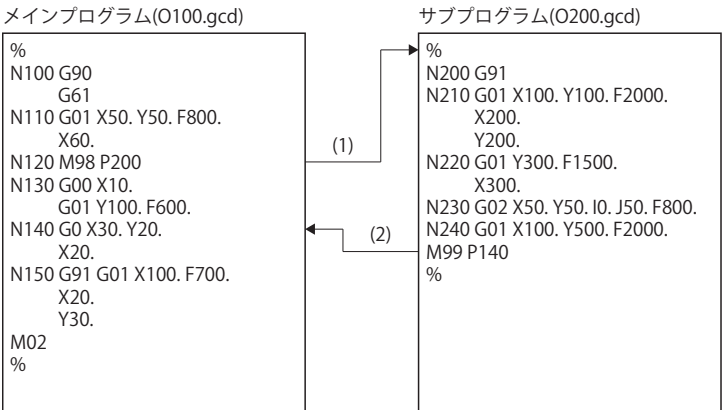
例

X100. M99の場合
「X100.」を実行後、呼出しブロックの次ブロックへ復帰します。

- メインプログラムでM99指令を実行すると先頭に戻ります。ただし、シーケンス番号を指定した場合は、M98指令、またはG65指令の次ブロックから指定されたシーケンス番号をサーチします。
- 「M99 P□」により指定したシーケンス番号は、M98指令、またはG65指令の次ブロックからプログラムエンドまでをサーチし、指定したシーケンス番号が存在しないと、プログラムの先頭からM98指令、またはG65指令の前ブロックまでをサーチします。そのため、プログラムの流れと逆の方向へサーチするため、プログラムの実行時間がかかることがあります。

プログラム例

■サブプログラムの復帰時に、シーケンス番号の指定がある場合



5.8 変数指令

アドレスに定数を与える代わりに変数を指定することで、プログラムに汎用性を持たせることができます。定数の場合は、プログラム実行中に数値の変更が行えないため、運転パターンごとにプログラムを用意する必要がありますが、変数を指定することで、プログラム実行中に数値の変更が行えるため、1つのプログラムで複数の運転パターンに対応することができます。変数には、下記の種類があります。

種類	機能
系統共通コモン変数	全系統のメインプログラム、サブプログラム、マクロプログラムを通じて共通に使用できます。
系統別コモン変数	系統ごとのメインプログラム、サブプログラム、マクロプログラムを通じて共通に使用できます。
ローカル変数	系統ごとのメインプログラム、サブプログラム、マクロプログラム内でのみ使用できます。 “[Rq.3376]Gコード制御要求(D54226.0+2s)”がONしたときに初期化(0クリア)します。

系統共通コモン変数/系統別コモン変数/ローカル変数

変数を使用する場合は、[モーション制御パラメータ]⇒[Gコード制御系統パラメータ]⇒[マクロ制御]にて、下記のパラメータを設定します。パラメータの詳細については、Gコード制御系統パラメータを参照してください。(P71ページ Gコード制御系統パラメータ)

- ・ 系統共通コモン変数点数
- ・ 系統別コモン変数点数
- ・ 系統共通コモン変数先頭デバイス番号
- ・ 系統別コモン変数先頭デバイス番号
- ・ ローカル変数先頭デバイス番号

■変数番号の設定

系統共通コモン変数点数と、系統別コモン変数点数の設定により、変数番号の設定の概略を下記に示します。

系統別コモン変数は、#100から系統別コモン変数点数に設定した点数分の連番となります。

系統共通コモン変数は、系統別コモン変数の最終番号以降から系統共通コモン変数点数に設定された点数分の連番となります。

項目	変数番号		備考
	系統1	系統2	
系統別コモン変数	#100	#100	系統別コモン変数点数に設定した点数分
	#101	#101	
	～	～	
	#i	#i	
系統共通コモン変数	#(i+1)		系統共通コモン変数点数に設定した点数分
	～		
	#j		

例

系統共通コモン変数「100点」、系統別コモン変数「400点」、ローカル変数「165点」を使用する場合

パラメータ項目	系統1	系統2
系統共通コモン変数点数	100	
系統別コモン変数点数	400	
系統共通コモン変数先頭デバイス番号	D6000	
系統別コモン変数先頭デバイス番号	D2000	D4000
ローカル変数先頭デバイス番号	D7000	D8000

*1 各先頭デバイスは、ユーザデバイスに割り当ててください。

*2 ローカル変数のエリアは、シーケンスプログラムやモーションSFCプログラムから書込みできません。書込みを行う場合は、コモン変数を使用してください。

各コモン変数、ローカル変数の設定を行った場合、デバイスは下記のように割り当てられます。

項目				変数	デバイス
系統共通コモン変数				#500～#599	D6000～D6399
系統別コモン変数	系統1			#100～#499	D2000～D3599
	系統2			#100～#499	D4000～D5599
ローカル変数	系統1	マクロ呼出しの深さ	0重	#1～#33	D7000～D7131
			1重	#1～#33	D7132～D7263
			2重	#1～#33	D7264～D7395
			3重	#1～#33	D7396～D7527
			4重	#1～#33	D7528～D7659
	系統2	マクロ呼出しの深さ	0重	#1～#33	D8000～D8131
			1重	#1～#33	D8132～D8263
			2重	#1～#33	D8264～D8395
			3重	#1～#33	D8396～D8527
			4重	#1～#33	D8528～D8659

Point

デバイスは、変数1点に対して4ワード単位で割り当てられ、データ型は64ビット浮動小数点型となります。

変数の記述

変数はGコードプログラム中で、下記のように記述します。

フォーマット


#i₁ = ₂ <式>

- 変数(#i)の「i」は、#に続く任意の式を示します。「i」には、下記のように記述できます。

i	記述例
数値	#123
変数	#[#543]
式 演算子 式	#[#110 + #119]
-(マイナス) 式	#[-#120]
[式]	#[[#119]]
関数 [式]	#[ACOS[#110]]

- 下記のように記述した場合は、変数が正しく認識されません。

記述例		説明
誤	正	
#206/2	#[206/2]	「#206/2」は、正しく解釈されず、「[#206]/2」と解釈されます。
#--105	#[[-105]]	「#--105」は、軽度エラー (エラーコード: 1FC3H(詳細コード: 0306H))となります。
#-[#100]	#[-#100]	「#-[#100]」は、軽度エラー (エラーコード: 1FC3H(詳細コード: 032AH))となります。

- <式>は、定数、変数、関数、または演算子による結合です。関数、演算子については、演算指令を参照してください。
( 184ページ 演算指令)

注意事項

- 変数(#i)は、モーションレジスタ(#)とは異なります。
- 標準演算子は、+、-、*、/の4種類です。
- 各変数番号に範囲外の番号を指定した場合、軽度エラー (エラーコード: 1FC3H(詳細コード: 032AH))となります。
- 変数の値は、64ビット浮動小数点(倍精度実数)値となります。64ビット浮動小数点値データの有効桁数は約15桁となるため、これをオーバーすると演算結果を正しく算出しない可能性があります。
- 系統間で系統共通コモン変数の受渡しを行う場合、M00指令、M01指令を使用して系統間のインターロック条件を入れてください。

変数の引用

変数は、O(プログラム番号)、N(シーケンス番号)を除くすべてのアドレスに対して使用できます。
変数の使用例を下記に示します。

項目	変数	内容
変数の値を直接使用する場合	#100 = 10 X#100	Xの値として、#100の値(10)を使用します。
変数の値の補数を使用する場合	X-#102	Xの値として、#102の符号を変えた値を使用します。
変数を定義する場合	#103 = #105	#103の値として、変数#103と等価な変数#105の値を使用します。
	#101 = 1000	#101の値として、「1000」を使用します。
変数演算式を定義する場合	#101 = #103 + #102 - 100	#101の値として、「#103 + #102 - 100」の演算結果の値を使用します。
	X[#101 + #103 + 1000]	Xの値として、「#101 + #103 + 1000」の演算結果の値を使用します。

注意事項

- 変数は、アドレスと同一ブロックには定義できません。分割して定義してください。同一ブロックに変数を定義すると、軽度エラー (エラーコード: 1FC3H(詳細コード: 0305H))となります。

例

X#101を使用する場合

誤	正
X#101 = #103 + 100	#101 = #103 + 100 X#101

- 「[」, 「]」は、5重まで使用できます。6重以上使用すると、軽度エラー (エラーコード: 1FC3H(詳細コード: 032DH))となります。

#543 = -[[[[[#120] / 2 + 15.] * 3 - #100] / #520 + #125 + #128] * #130 + #132]



- 「[」, 「]」は、ペアになるように使用してください。「[」, 「]」の数がペアになっていないと、軽度エラー (エラーコード: 1FC3H(詳細コード: 032EH))となります。
- 変数定義時に「=」がないと、軽度エラー (エラーコード: 1FC3H(詳細コード: 032BH))となります。
- 変数の引用は、末尾に小数点が記述されているものとして扱います。

例

#100 = 10の場合

X#100はX10となります。

- 変数を定義したとき、左辺の変数番号に小数を設定すると、軽度エラー (エラーコード: 1FC3H(詳細コード: 032AH))となります。ただし、変数番号に引用した変数の値、演算式の演算結果が小数の場合、または右辺の変数番号に小数が設定されている場合、小数点以下は切り捨てになります。
- 実行文、または制御指令(IF~GOTO, WHILE~DO~END)と同一ブロックに、変数を定義しないでください。

変数の応用

変数の多重化

変数は、変数番号を変数化(多重化)したり、<式>で置き換えることができます。変数の使用例を下記に示します。

項目	変数	内容
変数の多重化	#101 = 110 #110 = 120 #120 = 30 #105 = #[#101]	#101 = 110より, #[#101] = #[110]となります。 #110 = 120より, #[110] = #120となります。 したがって, #105 = #120, すなわち#105 = 30となります。
	#101 = 110 #110 = 120 #120 = 30 #105 = 1000 #[#101] = #105	#101 = 110より, #[#101] = #[110]となります。 #110 = 120より, #[110] = #120となります。 したがって, #120 = #105, すなわち#120 = 1000となります。
	#110 = 105 ##110 = 100	<式>##110 = 100は, #[#110] = 100と同じ扱いとなります。 #105 = 100となります。
変数番号を<式>で置換え	#110 = 210 #[#110 + 1] = 1000 #[#110 - 1] = -1000 #[#110 * 3] = 100 #[#110 / 2] = -100 ^{*1}	#211 = 1000となります。 #209 = -1000となります。 #630 = 100となります。 #105 = -100となります。

*1 演算結果が小数の場合、小数点以下は切り捨てになります。
(例)「#110 = 201」の場合、「#[#110 / 2]」は、「#100」と同じ扱いになります。

変数(デバイス)の読み込み／書き込みタイミング

変数の値は、下記のタイミングで読み込み、書き込みを行います。

システムのデバイスアクセスタイミング

○: デバイスの値をシステム側が読み込む

△: デバイスへシステム側が出力する

—: アクセスしない

操作／タイミング	コモン変数	ローカル変数
Gコード制御要求ON	—	△ ^{*1}
Gコードプログラムの開始時	○	—
シングルブロック運転の自動運転起動時	○	—
M00指令, M01指令後の再始動時	○	—
ブロック切り換え時 (連続運転中の変数指令, 演算指令, 制御指令のブロックを除く)	△	△
リセット時	△	△

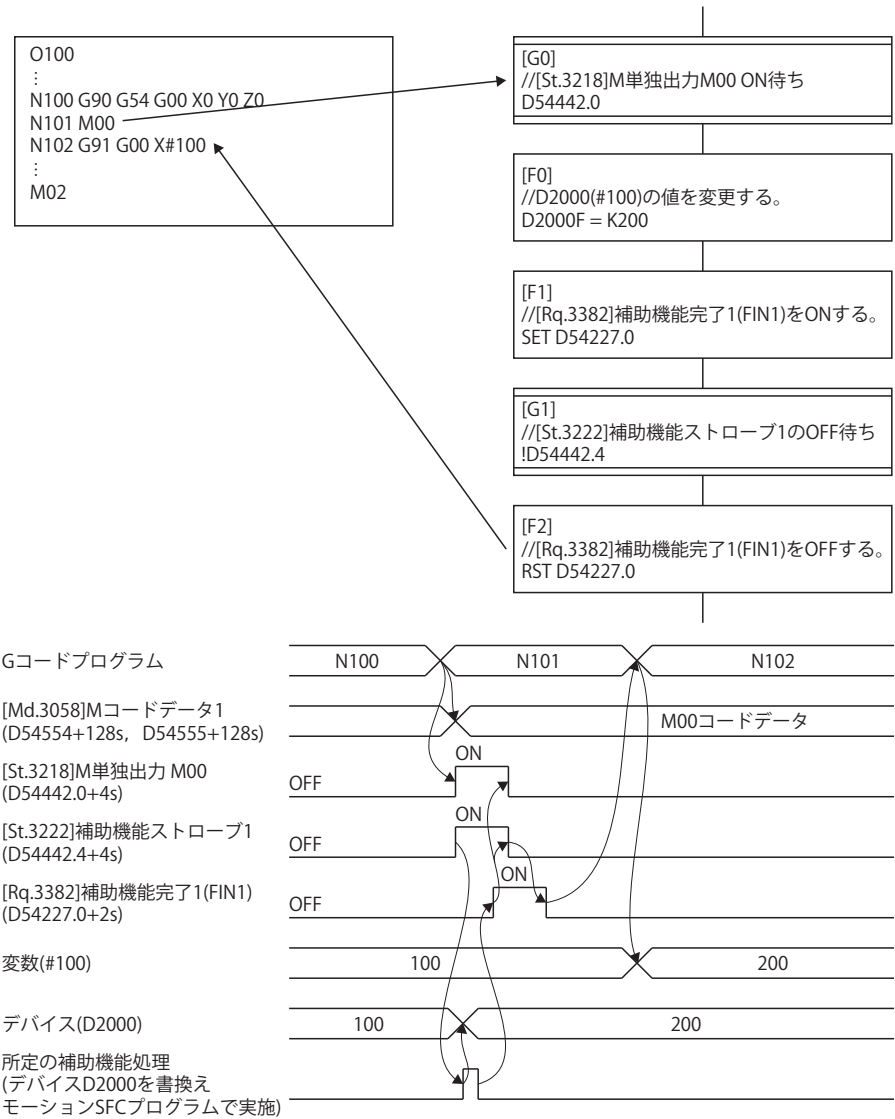
*1 初期化(0クリア)します。

GコードプログラムとモーションSFCプログラム間で変数の受渡しを行う場合

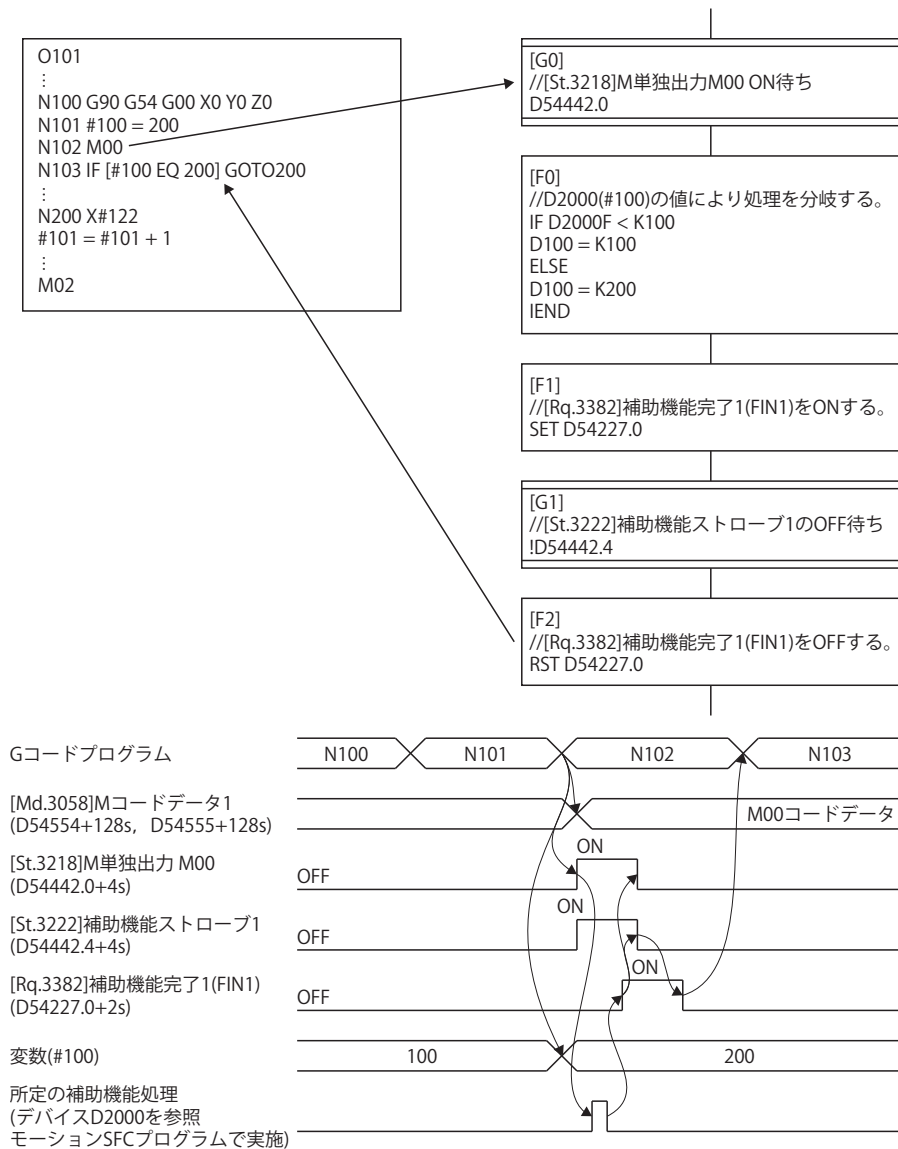
GコードプログラムとモーションSFCプログラム間で変数の受渡しを行う場合は、M00指令、M01指令を使用し、“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”のOFF、または“[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”のONにて次のブロックを実行するようにしてください。

モーションSFCプログラムにて書換えた変数の値をGコードプログラムで参照する場合、およびGコードプログラムで書換えた変数の値をモーションSFCプログラムで参照する場合の例を下記に示します。

■モーションSFCプログラムで書換えた変数#100(D2000に割付け)の値をGコードプログラムで参照する場合(系統1を使用)



■Gコードプログラムで書換えた変数#100(D2000に割付け)の値をモーションSFCプログラムで参照する場合(系統1を使用)

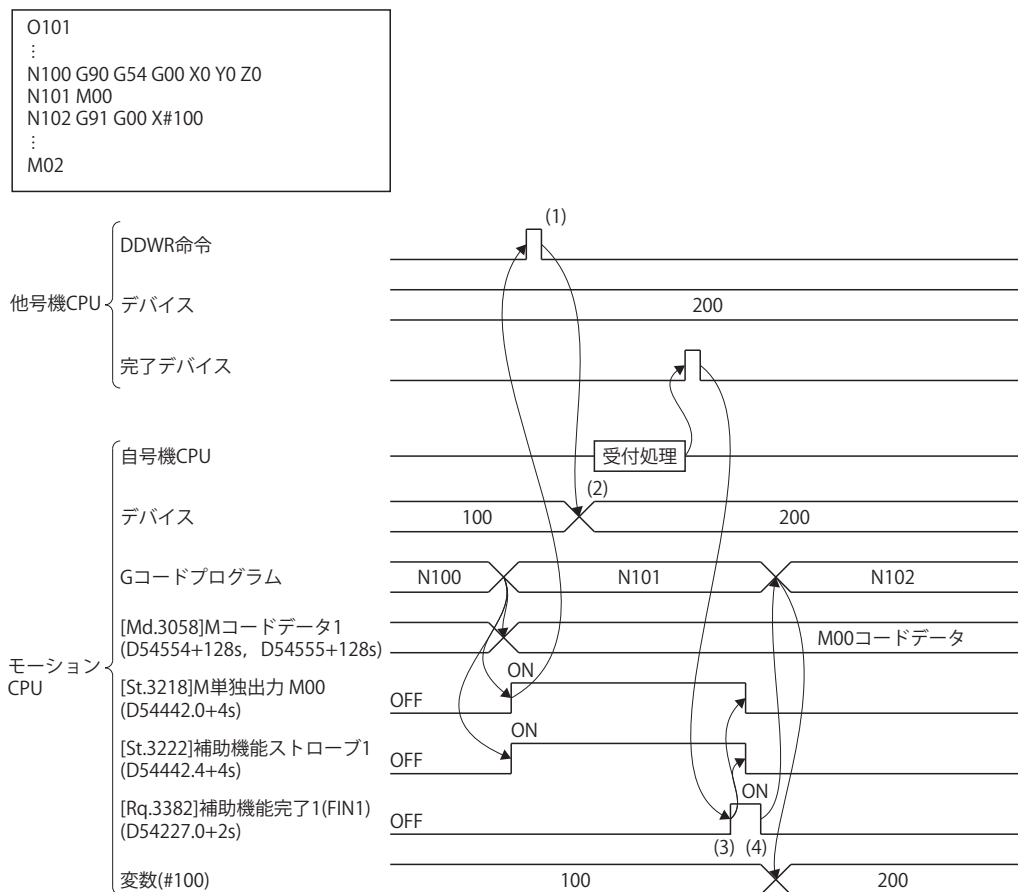


モーションCPUと他号機との間で変数の受渡しを行う場合

モーションCPUと他号機CPUとの間で変数の受渡しを行う場合は、M00指令、M01指令を使用し、DDWR/DDR命令にて共通変数に割り当てたデバイスの書込み、または読出しを行った後、“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”のOFF、または“[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”のONにて次のブロックを実行するようにしてください。

他号機CPUのデバイスデータをGコードプログラムで参照する場合、およびGコードプログラムで書換えた変数の値を他号機CPUで参照する場合の例を下記に示します。

■他号機CPUのデバイスデータをGコードプログラム(変数: #100)で参照する場合(系統1を使用)



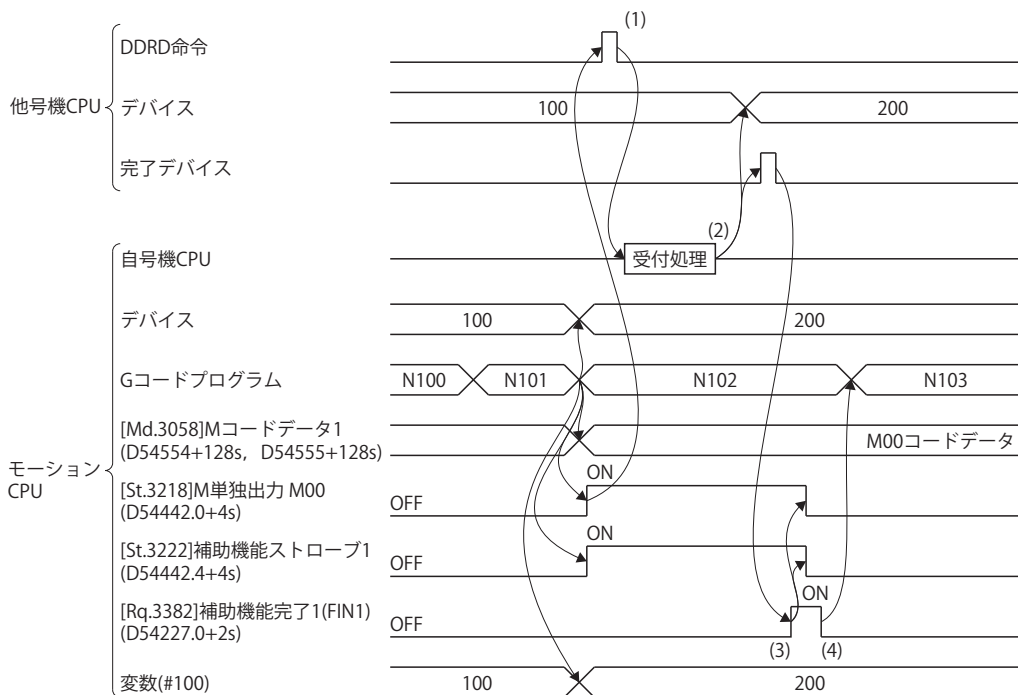
- (1) “[St.3218]M単独出力 M00(D54442.0+4s)”のON確認後、DDWR命令を実行する。
- (2) 他号機CPUのデバイスデータを自号機CPUの共通変数に割り当てたデバイスに書込む。
- (3) DDWR命令完了確認後、“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”をONする。
- (4) “[St.3222]補助機能ストローブ1(D54442.4+4s)”のOFF確認後、“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”をOFFし、次ブロックへ移行する。

■Gコードプログラムで書換えた変数#100の値を他号機CPUで参照する場合(系統1を使用)

```

O101
:
N100 G90 G54 G00 X0 Y0 Z0
N101 #100 = 200
N102 M00
N103 IF [#100 EQ 200] GOTO200
:
N200 X#122
#101 = #101 + 1
:
M02

```



- (1) "[St.3218]M単独出力 M00(D54442.0+4s)"のON確認後、DDR命令を実行する。
- (2) 自号機CPUの共通変数に割り当てたデバイスを他号機CPUのデバイスに読出す。
- (3) DDR命令完了確認後、"[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)"をONする。
- (4) "[St.3222]補助機能ストロープ1(D54442.4+4s)"のOFF確認後、"[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)"をOFFし、次ブロックへ移行する。

共通変数

任意の位置から共通に利用される変数です。詳細は、変数指令を参照してください。(173ページ 変数指令)

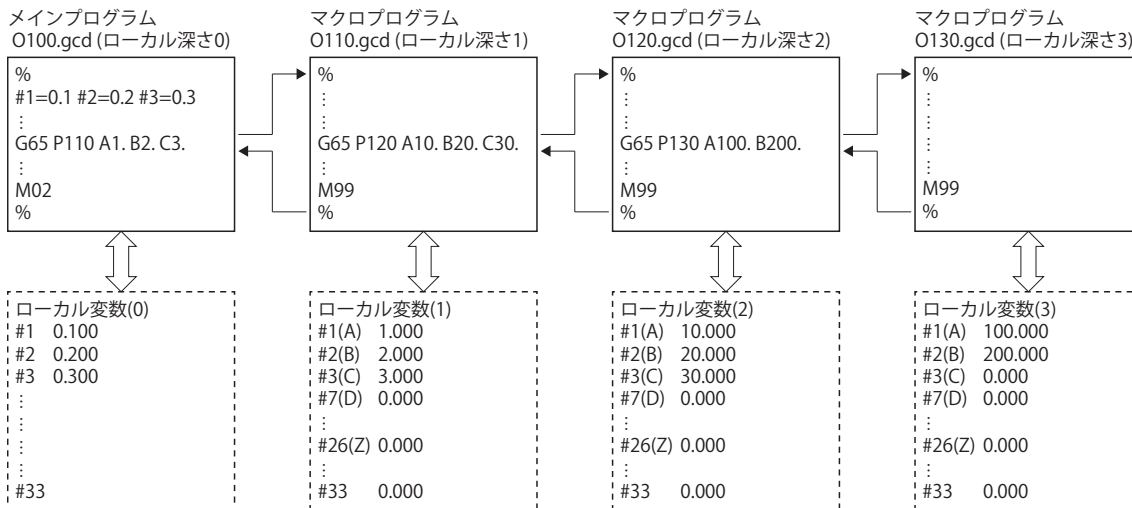
ローカル変数

マクロ呼出し時に、引数として定義できるほか、メインプログラム、サブプログラム、マクロプログラム内でのみ利用する変数です。マクロプログラム間での関係はなく4重まで重複可能です。重複が4重を超えると、軽度エラー(エラーコード: 1FC3H(詳細コード: 0335H))となります。

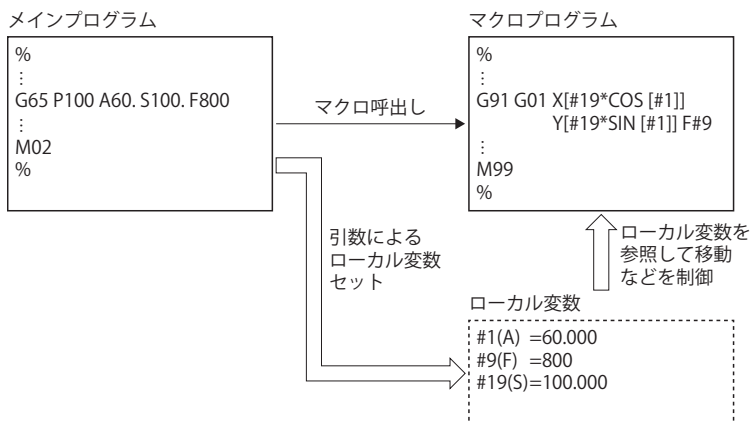
ローカル変数先頭デバイス番号が未設定の状態ではローカル変数を指令した場合、軽度エラー(エラーコード: 1FC3H(詳細コード: 032AH))となります。ローカル変数を使用しない場合、本設定は省略可能です。

マクロ呼出しについては、マクロ呼出し(G65)を参照してください。(147ページ G65: マクロ呼出し)

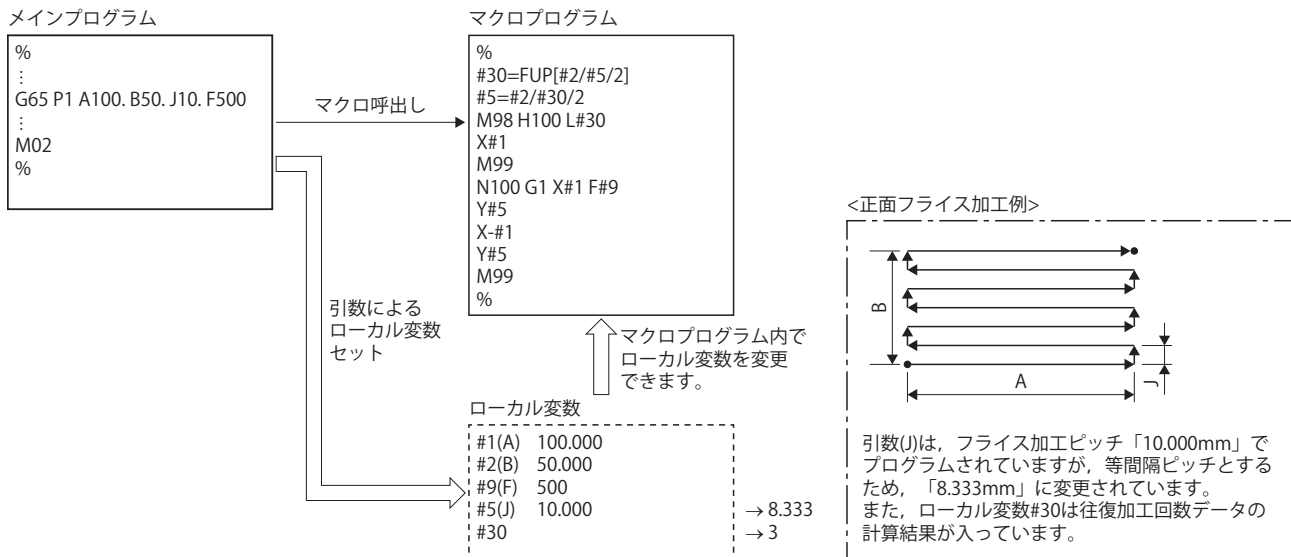
- ローカル変数はマクロ呼出し(4重)の各深さごとに独立して使用できます。メインプログラム(ローカル深さ0)にも、ローカル変数が独立して用意されています。実行中のローカル変数深さは、“[Md.3074]ローカル変数深さ(D54582+128s)”でモニタ可能です。



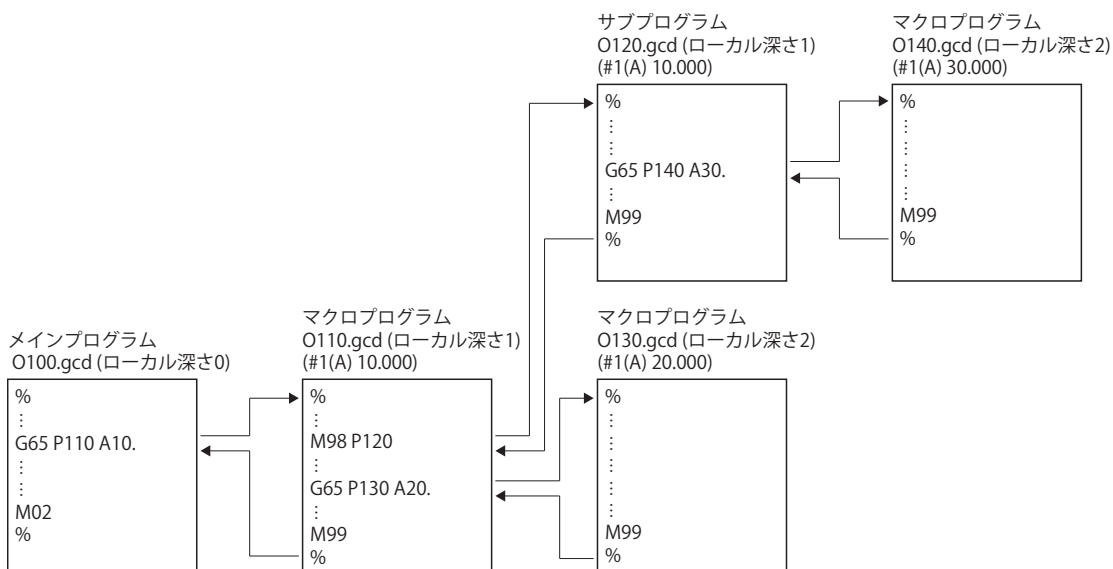
- マクロ呼出し時、引数指定することにより、マクロプログラムのローカル変数を定義できます。



- ローカル変数はマクロプログラム内で自由に使用できます。



- M98指令とG65指令を組み合わせたときのローカル変数深さは、下記ようになります。



- マクロ呼出し時、引数に指定がないローカル深さの変数と、Gコード制御系統パラメータの「ローカル変数先頭デバイス番号」の設定にて割り当てられたデバイスは、「[Rq.3385]G65引数初期化(D54227.4+2s)」の設定により下記ようになります。Gコードプログラム運転中、マクロ呼出しでローカル深さを繰返し往復する場合、マクロ呼出しごとに「[Rq.3385]G65引数初期化(D54227.4+2s)」の設定に準じた動作となります。

設定値	内容
ON	マクロ呼出し時、指定しない引数を保持します。
OFF	マクロ呼出し時、指定しない引数を初期化(0クリア)します。

5.9 演算指令

演算指令は、変数間で種々の演算を行うことができます。対応している演算指令を下記に示します。

種類	演算指令	式 ^{*1*2}	内容	参照
変数の定義, 置換	=	#i = #j	定数, 置換	☞ 186ページ 定義/置換(=)
加法形演算	+	#i = #j + #k	加算	☞ 187ページ 加法形演算(+, -)
	-	#i = #j - #k	減算	
乗法形演算	*	#i = #j * #k	乗算	☞ 188ページ 乗法形演算(*, /, MOD)
	/	#i = #j / #k	除算	
	MOD	#i = #j MOD #k	余り	
論理演算	OR	#i = #j OR #k	論理和(32ビットのビットごと, 小数点以下切り捨て)	☞ 189ページ 論理演算(OR, XOR, AND)
	XOR	#i = #j XOR #k	排他的論理和(32ビットのビットごと, 小数点以下切り捨て)	
	AND	#i = #j AND #k	論理積(32ビットのビットごと, 小数点以下切り捨て)	
関数	SIN	#i = SIN[#k]	正弦(#kの単位: degree)	☞ 190ページ 三角関数(SIN, COS, TAN, ASIN, ACOS, ATAN)
	COS	#i = COS[#k]	余弦(#kの単位: degree)	
	TAN	#i = TAN[#k]	正接(#kの単位: degree)	
	ASIN	#i = ASIN[#k]	逆正弦	
	ACOS	#i = ACOS[#k]	逆余弦	
	ATAN	#i = ATAN[#k]	逆正接	
	SQRT	#i = SQRT[#k]	平方根	
	ABS	#i = ABS[#k]	絶対値	☞ 191ページ 関数(SQRT, ABS, BIN, BCD, ROUND, FIX, FUP, LN, EXP, POW)
	BIN	#i = BIN[#k]	BCDからBIN変換(#kの値: BCD, 小数点以下切り捨て)	
	BCD	#i = BCD[#k]	BINからBCD変換(#kの値: BIN, 小数点以下切り捨て)	
	ROUND	#i = ROUND[#k]	四捨五入	
	FIX	#i = FIX[#k]	小数点以下切り捨て	
	FUP	#i = FUP[#k]	小数点以下切り上げ	
	LN	#i = LN[#k]	自然対数	
	EXP	#i = EXP[#k]	e(=2.718.....)を底とする指数	
	POW	#i = POW[#j, #k]	べき乗(#jの#k乗)	

*1 #i, #j, #kは, 変数を示します。

*2 #j, #kの代わりに定数を使用することもできます。

注意事項

- 小数点が記述されていない値は、末尾に小数点が記述されているものとして扱います。

例

「100」と記述した場合

記述した値を「100.」として扱います。

- 関数の後に続く式は、必ず「[」,「]」で囲う必要があります。囲われていないと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0306H))となります。
- 四則演算子(+, -, *, /)を記述した場合、小数点付きデータとして扱います。

例

G00 X123 + 0

X軸は、「123mm」の指令となります。「12.3μm」にはなりません。

- 演算時に丸め誤差が生じることがあります。特に比較演算に使用する場合は、丸め誤差の影響により、意図しない動作となることがありますので注意してください。

例

丸め誤差の影響で、#200の値によっては比較演算の結果が真にならない場合

#100 = SQRT[#200]

#300 = #100 * #100

IF [#200 EQ #300]

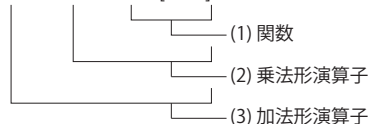
- 軽度エラー発生時は、演算により変数の値は更新されません。

演算の優先順位

- 演算の優先順位は関数、乗法形演算、加法形演算、論理演算の順です。

例

#101 = #111 + #112 * SIN[#113]



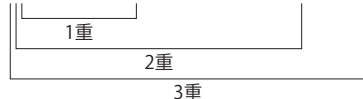
(1) 「SIN[#113]」を計算します。

(2) 「#112 * [(1)の計算結果]」を計算します。

(3) 「#111 + [(2)の計算結果]」を計算します。

- 演算順位を優先させたい部分を「[」,「]」で括ることができます。「[」,「]」は、関数の「[」,「]」を含めて5重まで可能です。6重以上使用すると、軽度エラー (エラーコード: 1FC3H(詳細コード: 032DH))となります。

#101 = SQRT[[[#111 + #112] * SIN[#113] + #114] * #115]



定義/置換(=)

定義/置換(=)を行います。

コード	フォーマット
=	<div><div>n1</div><div>演算子</div><div>n2</div><div>変数または定数</div><div>演算子(=)</div><div>変数</div></div>

機能

- 指定した演算子の計算を行います。
- 小数点が記述されていない値は，末尾に小数点が記述されているものとして扱います。

プログラム例

演算子	プログラム	結果
=	#201 = 1000 #202 = 1000.	#201 = 1000.0000 #202 = 1000.0000
	#101 = 100 #102 = 200 #203 = #101 #204 = #102	#203 = 100.0000 #204 = 200.0000

加法形演算(+, -)

加算(+), 減算(-)を行います。

コード	フォーマット
+, -	$n1 \quad \text{演算子} \quad n2$ <p>変数または定数 演算子(+, -) 変数または定数</p>

機能

- 指定した演算子の計算を行います。
- 四則演算子(+, -)を記述した場合, 小数点付きデータとして扱います。
- 小数点が記述されていない値は, 末尾に小数点が記述されているものとして扱います。

プログラム例

演算子	プログラム	結果
+	<pre>#201 = 100 #202 = 1000 #211 = #201 + 1000 #213 = #201 + #202</pre>	<pre>#211 = 1100.0000 #213 = 1100.0000</pre>
-	<pre>#201 = 100 #202 = 1000 #212 = #202 - 50. #213 = #201 - #202</pre>	<pre>#212 = 950.0000 #213 = -900.0000</pre>

乗法形演算(*, /, MOD)

乗算(*), 除算(/), 余り(MOD)を行います。

コード	フォーマット
*, /, MOD	n1 演算子 n2
	変数または定数 演算子(*, /, MOD) 変数または定数

機能

- 指定した演算子の計算を行います。
- 四則演算子(*, /)を記述した場合、小数点付きデータとして扱います。
- 小数点が記述されていない値は、末尾に小数点が記述されているものとして扱います。

プログラム例

関数	プログラム	結果
*	#221 = 100 * 100	#221 = 10000.0000
	#222 = 100. * 100	#222 = 10000.0000
	#223 = 100 * 100.	#223 = 10000.0000
	#224 = 100. * 100.	#224 = 10000.0000
	#101 = 100 #102 = 200 #225 = #101 * #102	#225 = 20000.0000
/	#226 = 100 / 100	#226 = 1.0000
	#227 = 100. / 100	#227 = 1.0000
	#228 = 100 / 100.	#228 = 1.0000
	#229 = 100. / 100.	#229 = 1.0000
	#101 = 100 #102 = 200 #230 = #101 / #102	#230 = 0.5000
	#231 = 100 / 0	軽度エラー (エラーコード: 1FC3H(詳細コード: 0319H))
MOD	#219 = 48 #220 = 9 #231 = #219 MOD #220	#219/#220 = 48/9 = 5 余り 3 #231 = 3
	#232 = 100 MOD 0	軽度エラー (エラーコード: 1FC3H(詳細コード: 0319H))

論理演算(OR, XOR, AND)

論理和(OR), 排他的論理和(XOR), 論理積(AND)を行います。

コード	フォーマット
OR, XOR, AND	<div><div>n1</div><div>論理演算子</div><div>n2</div><div>変数または定数</div><div>演算子(OR, XOR, AND)</div><div>変数または定数</div></div>

機能

- 指定した論理演算子の計算を行います。
- 小数点が記述されていない値は、末尾に小数点が記述されているものとして扱います。
- 論理和(OR), 排他的論理和(XOR), 論理積(AND)は、32ビット整数型の範囲内で演算を行ってください。演算を行うデータが32ビット整数型の範囲外の場合、想定した演算結果にはなりません。

プログラム例

演算子	プログラム	結果
OR*1	#203 = 100 #204 = #203 OR 14	#203 = 64H 14 = 0EH #204 = 6EH = 110
	#205 = 100.1234 OR 14	100.1234 = 64H 14 = 0EH #205 = 6EH = 110
	#206 = -100 OR 14	-100 = 9CH 14 = 0EH #206 = 9EH = -98
XOR*1	#203 = 100 #204 = #203 XOR 14	#203 = 64H 14 = 0EH #204 = 6AH = 106
	#205 = 100.1234 XOR 14	100.1234 = 64H 14 = 0EH #205 = 6AH = 106
	#206 = -100 XOR 14	-100 = 9CH 14 = 0EH #206 = 92H = -110
AND*1	#209 = 100 #210 = #209 AND 15	#209 = 64H 15 = 0FH #210 = 04H = 4
	#211 = 100.1234 AND 15	100.1234 = 64H 15 = 0FH #211 = 04H = 4
	#212 = -100 AND 15	-100 = 9CH 15 = 0FH #212 = 0CH = 12

*1 小数点以下は切り捨てになります。

三角関数(SIN, COS, TAN, ASIN, ACOS, ATAN)

SIN(正弦), COS(余弦), TAN(正接), ASIN(逆正弦), ACOS(逆余弦), ATAN(逆正接)の演算を行います。

コード	フォーマット
SIN, COS, TAN, ASIN, ACOS, ATAN	$\text{function } \angle [n]$ <div style="display: flex; justify-content: space-around; margin-top: 10px;"> <div style="text-align: center;"> \swarrow 変数または定数 </div> <div style="text-align: center;"> \searrow 関数(SIN, COS, TAN, ASIN, ACOS, ATAN) </div> </div>

機能

- 指定した三角関数の演算を行います。
- 小数点が記述されていない値は、末尾に小数点が記述されているものとして扱います。
- 関数の後に続く式は、必ず「[」, 「]」で囲う必要があります。囲われていないと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0306H))となります。
- TAN(正接)にて、引数にcosθが「0」となる数を設定した場合、軽度エラー (エラーコード: 1FC3H(詳細コード: 0318H))となります。

プログラム例

関数	プログラム	結果
SIN*1	#501 = SIN [60]	#501 = 0.8660
	#502 = SIN [60.]	#502 = 0.8660
	#503 = 1000 * SIN [60]	#503 = 866.0254
	#504 = 1000. * SIN [60.]	#504 = 866.0254
COS*1	#541 = COS [45]	#541 = 0.7071
	#542 = COS [45.]	#542 = 0.7071
	#543 = 1000 * COS [45]	#543 = 707.1068
	#544 = 1000. * COS [45.]	#544 = 707.1068
TAN*1	#551 = TAN [60]	#551 = 1.7321
	#552 = TAN [60.]	#552 = 1.7321
	#553 = 1000 * TAN [60]	#553 = 1732.0508
	#554 = 1000. * TAN [60.]	#554 = 1732.0508
	#555 = TAN [90.]	軽度エラー (エラーコード: 1FC3H(詳細コード: 0318H))
ASIN*2	#531 = ASIN[100.5000 / 201.]	#531 = 30.0000
	#532 = ASIN[100.5000 / 201]	#532 = 30.0000
	#533 = ASIN[0.5000]	#533 = 30.0000
	#534 = ASIN[-0.5000]	#534 = -30.0000
	#535 = ASIN[1.1000]	軽度エラー (エラーコード: 1FC3H(詳細コード: 0318H))
ACOS*2	#521 = ACOS [100 / 141.4210]	#521 = 45.0000
	#522 = ACOS [100. / 141.4210]	#522 = 45.0000
	#523 = ACOS [1.1000]	軽度エラー (エラーコード: 1FC3H(詳細コード: 0318H))
ATAN	#561 = ATAN [173205 / 100000]	#561 = 60.0000
	#562 = ATAN [173205 / 100000.]	#562 = 60.0000
	#563 = ATAN [173.2050 / 100]	#563 = 60.0000
	#565 = ATAN [1.732050]	#565 = 60.0000

*1 引数は「-360.~360.」の範囲で指定してください。

*2 引数は「-1~1」の範囲で指定してください。

関数(SQRT, ABS, BIN, BCD, ROUND, FIX, FUP, LN, EXP, POW)

SQRT(平方根), ABS(絶対値), BIN(BCDからBIN変換), BCD(BINからBCD変換), ROUND(四捨五入), FIX(小数点以下切り捨て), FUP(小数点以下切り上げ), LN(自然対数), EXP(eを底とする指数), POW(べき乗)の演算を行います。

コード	フォーマット
SQRT, ABS, BIN, BCD, ROUND, FIX, FUP, LN, EXP, POW	<div> <div>function</div> <div>└─[n]</div> <div>└─変数または定数</div> <div>└─関数(SQRT, ABS, BIN, BCD, ROUND, FIX, FUP, LN, EXP, POW)</div> </div>

機能

- 指定した関数の演算を行います。
- 小数点が記述されていない値は、末尾に小数点が記述されているものとして扱います。
- 関数の後に続く式は、必ず「[」, 「]」で囲う必要があります。囲われていないと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0306H))となります。
- 下記の場合は、軽度エラー (エラーコード: 1FC3H(詳細コード: 0318H))となります。
 - 平方根指令(SQRT)にて、引数に負の数を設定したとき。
 - 自然対数指令(LN)にて、引数に負の数を設定したとき。
 - べき乗指令(POW)にて、引数1に「0」、かつ引数2に「0」以下の数を設定したとき。
 - べき乗指令(POW)にて、引数1に負の数、かつ引数2に整数でない数を設定したとき。
- EXP(eを底とする指数), POW(べき乗)の演算結果が64ビット浮動小数点型のデータ範囲を超えると、オーバーフローが発生しますが、軽度エラーとはなりません。
- BIN(BCDからBIN変換), BCD(BINからBCD変換)の引数は、「0～99999999」の範囲で指定してください。引数が「0～99999999」の範囲外の場合、想定した演算結果にはなりません。
- FIX(小数点以下切り捨て), FUP(小数点以下切り上げ)にて、引数に負の数を設定した場合、下記の動作となります。モーションSFC プログラム(演算制御プログラム)とは演算結果が異なります。

関数	動作	
	Gコード制御	モーションSFCプログラム
FIX	絶対値が小さくなります。	絶対値が大きくなります。
FUP	絶対値が大きくなります。	絶対値が小さくなります。

プログラム例

関数	プログラム	結果
SQRT ^{*1}	#571 = SQRT [1000]	#571 = 31.6228
	#572 = SQRT [1000.]	#572 = 31.6228
	#573 = SQRT [10. * 10. + 20. * 20]	#573 = 22.3607
	#574 = SQRT [-1.0000]	軽度エラー (エラーコード: 1FC3H(詳細コード: 0318H))
ABS	#576 = -1000	#576 = -1000.0000
	#577 = ABS [#576]	#577 = 1000.0000
	#203 = 70. #204 = -50.	#580 = 120.0000
	#580 = ABS [#204 - #203]	
BIN ^{*2*3}	#201 = 100	#211 = 64
	#211 = BIN [#201]	
	#213 = BIN [100.5]	#213 = 64
	#215 = BIN [99999999]	#215 = 6564165
BCD ^{*2}	#201 = 100	#212 = 256
	#212 = BCD [#201]	
	#214 = BCD [100.5]	#214 = 256

関数	プログラム	結果
ROUND	#221 = ROUND [14 / 3]	#221 = 5
	#222 = ROUND [14. / 3]	#222 = 5
	#223 = ROUND [14. / 3.]	#224 = 5
	#224 = ROUND [-14 / 3]	#225 = -5
	#225 = ROUND [-14 / 3.]	#226 = -5
	#226 = ROUND [14 / 6]	#229 = 2
FIX	#221 = FIX [14 / 3]	#221 = 4.0000
	#222 = FIX [14. / 3]	#222 = 4.0000
	#223 = FIX [14. / 3.]	#224 = 4.0000
	#224 = FIX [-14 / 3]	#225 = -4.0000
	#225 = FIX [-14 / 3.]	#227 = -4.0000
	#226 = FIX [14 / 6]	#229 = 2.0000
FUP	#221 = FUP [14 / 3]	#221 = 5.0000
	#222 = FUP [14. / 3]	#222 = 5.0000
	#223 = FUP [14. / 3.]	#224 = 5.0000
	#224 = FUP [-14 / 3]	#225 = -5.0000
	#225 = FUP [-14 / 3.]	#227 = -5.0000
	#226 = FUP [14 / 6]	#229 = 3.0000
LN	#101 = LN [5]	#101 = 1.6094
	#102 = LN [0.5]	#102 = -0.6931
	#103 = LN [-5]	軽度エラー (エラーコード: 1FC3H(詳細コード: 0318H))
EXP	#104 = EXP [2]	#104 = 7.3891
	#105 = EXP [1]	#105 = 2.7183
	#106 = EXP [-2]	#106 = 0.1353
POW	#107 = POW [2, 3]	#107 = 8.0000
	#108 = POW [2, -3]	#108 = 0.1250
	#109 = POW [2.5, 3.5]	#109 = 24.7053
	#110 = POW [0, -1]	軽度エラー (エラーコード: 1FC3H(詳細コード: 0318H))
	#111 = POW [-2, 2.5]	

*1 括弧内で演算することにより演算精度が向上しますので、可能な限り括弧内で演算を行ってください。

*2 小数点以下は切り捨てになります。

*3 BINの変換結果は、10進数に変換され、変数に設定されます。

(例) BIN [99999999] = 5F5E0FFHの場合

$(5 \times 1000000) + (15 \times 100000) + (5 \times 10000) + (14 \times 1000) + (0 \times 100) + (15 \times 10) + (15 \times 1) = 6564165$

5.10 制御指令

制御指令により、プログラムの流れを制御できます。対応している制御指令を下記に示します。

指令	制御指令	参照
分岐	IF, GOTO	☞ 193ページ 分岐(IF, GOTO)
	IF, THEN, ELSE, ENDIF	☞ 195ページ 分岐(IF, THEN, ELSE, ENDIF)
繰り返し	WHILE, DO, END	☞ 198ページ 繰り返し(WHILE, DO, END)

分岐(IF, GOTO)

実行プログラムの流れを条件により制御します。

コード	フォーマット
IF, GOTO	IF [条件式] GOTO n └─ シーケンス番号

5

機能

- 条件が成立したときに、GOTOで指定したシーケンス番号へ分岐します。不成立のときは、次のブロックを実行します。
- 条件式で使用する命令には、下記の種類があります。

命令	条件式 ^{*1}	内容
EQ	#i EQ #j	#iと#jが等しいとき(=)
NE	#i NE #j	#iと#jが等しくないとき(≠)
GT	#i GT #j	#iが#jを超えているとき(>)
LT	#i LT #j	#iが#j未満のとき(<)
GE	#i GE #j	#iが#j以上のとき(≥)
LE	#i LE #j	#iが#j以下のとき(≤)

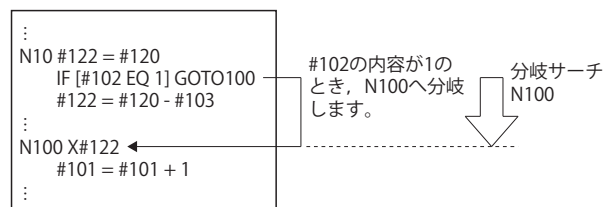
*1 #i, #jは、変数を示します。

- 条件式の#i, #j, およびn(シーケンス番号)の代わりに式、または変数を使用できます。
- 条件式は、必ず「[」, 「]」で囲う必要があります。
- GOTOで指定するシーケンス番号は、必ず同一プログラム内になければなりません。プログラム内に指定するシーケンス番号がないと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0320H))となります。
- GOTO nのみ指定した場合は、無条件に指定したシーケンス番号へ分岐します。

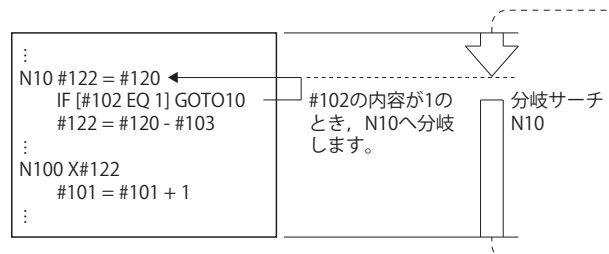
注意事項

- 分岐先のシーケンス番号をサーチするとき、IF文の次ブロックからプログラムエンドまでをサーチし、指定したシーケンス番号が存在しないと、プログラムの先頭からIF文の前ブロックまでをサーチします。そのため、プログラムの流れと逆の方向へのサーチは、順方向へのサーチに比べてプログラムの実行時間がかかることがあります。

・順方向にサーチする場合



・逆方向にサーチする場合



- 1ブロックの先読みを行っているため、IF文の前ブロックの処理中に変数の値を変更しても、変更前の値で条件判定を行います。このような場合は、M00指令、M01指令にて先読み禁止にしてください。

```
#102 = 0
:
N10 #122 = #120
  #102 = 1
  M100 ←
  IF [#102 EQ 1] GOTO 100
  #122 = #120 - #103
:
N100 X#122
  #101 = #101 + 1
:
```

M100の実行中にモーションSFCプログラムから#102の値を変更しても、先読み処理を行っているため、N100へジャンプします。

- 条件式に「EQ」、および「NE」を使用しての比較は、整数のみ使用できます。小数点以下の数値を使用する場合の比較は、「GT」、「LT」、「GE」、「LE」を使用してください。

プログラム例

■条件を満たした場合、指定されたシーケンス番号にジャンプするプログラム

動作	プログラム	備考
(1)	N200 G91	
(2)	N210 G01 X100. Y100. F2000.	
(3)	X200.	
(4)	Y200.	
(5)	IF [#100 GT #102] GOTO230	#100が#102を超えると、N230へジャンプ((8)へジャンプ)
(6)	N220 G01 Y300. F1500.	
(7)	X300.	
(8)	N230 G02 X50. Y50. I0. J50. F800.	
(9)	N240 G01 X100. Y500. F2000.	
(10)	IF [#110 EQ 180] GOTO260	#110が180のとき、N260へジャンプ((13)へジャンプ)
(11)	N250 G00 X10.	
(12)	Y100.	
(13)	N260 G00 X0. Y0.	

分岐(IF, THEN, ELSE, ENDIF)

実行プログラムの流れを条件により制御します。

コード	フォーマット
IF, THEN, ELSE, ENDIF	IF [条件式] THEN : ELSE : ENDIF

機能

- 条件が成立したときは、THEN命令(ELSEまでのブロック)の処理を実行します。不成立のときは、ELSE命令(ENDIFまでのブロック)の処理を実行します。
- 条件式で使用する命令には、下記の種類があります。

命令	条件式*1	内容
EQ	#i EQ #j	#iと#jが等しいとき(=)
NE	#i NE #j	#iと#jが等しくないとき(≠)
GT	#i GT #j	#iが#jを超えているとき(>)
LT	#i LT #j	#iが#j未満のとき(<)
GE	#i GE #j	#iが#j以上のとき(≥)
LE	#i LE #j	#iが#j以下のとき(≤)

*1 #i, #jは、変数を示します。

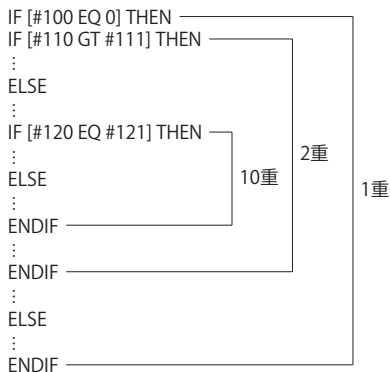
- 条件式の#i, #jの代わりに式、または変数を使用できます。
- 条件式は、必ず「[」, 「]」で囲う必要があります。
- IF/THEN/ELSE/ENDIF命令は、下記のフォーマットで記述できます。

	フォーマット	内容
(1)	IF [条件式] THEN マクロ文, または実行文 : ELSE マクロ文, または実行文 : ENDIF	<ul style="list-style-type: none"> 実行命令に実行文があるとき、または複数の命令があるときは、IF/THEN/ELSE/ENDIF命令のブロックで囲んで指令してください。 ENDIF命令がないと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0330H))となります。 (1)と(2)のフォーマットを組み合わせ使用できます。 <p>(例)</p> <pre>IF [#100 EQ 0] THEN #101 = 2 G00 X#101 ELSE #110 = 10 ENDIF</pre>
(2)	IF [条件式] THEN 演算指令 ELSE 演算指令	<ul style="list-style-type: none"> 実行命令が演算命令の場合は、THEN/ELSE命令に続いて記述できます。 ENDIF命令を記述することもできます。
(3)	IF [条件式] THEN 演算指令 ELSE 演算指令	<ul style="list-style-type: none"> IF文を入れ子で使用しているときは、ENDIF命令を記述してください。 <p>(例)</p> <pre>IF [#100 EQ 0] THEN.....(a) IF [#110 EQ 1] THEN #120=10.....(b) ENDIF.....(c) ELSE #120 = 20 ENDIF ENDIF命令(c)ありの場合: (a)のIF条件が偽のとき、ELSE処理を実行する。 ENDIF命令(c)なしの場合: (b)のIF条件が偽のとき、ELSE処理を実行する。</pre>

- THEN命令(ELSEまでのブロック)の処理、またはELSE命令(ENDIFまでのブロック)の処理を省略できます。

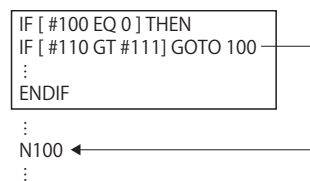
命令	プログラム
ELSE文の処理を省略した場合	IF[#100 EQ 0] THEN #101 = 2 G00 X#101 ENDIF
THEN文の処理を省略した場合	IF [#100 EQ 0] ELSE #101 = 10

- IF命令がない状態で、THEN/ELSE/ENDIF命令を指令すると、軽度エラー (エラーコード: 1FC3H(詳細コード: 0330H))となります。
- IF命令と同一ブロックにTHEN/ELSE命令ともない(IF文が単独で命令されている)と、軽度エラー (エラーコード: 1FC3H(詳細コード: 0331H))となります。
- IF文の深さは、10重までです。IF文の深さが10重を超えると、軽度エラー (エラーコード: 1FC3H(詳細コード: 032FH))となります。

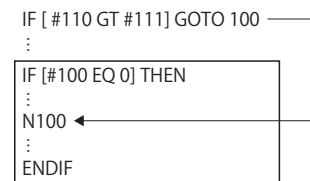


- IF～ENDIF文の範囲内からIF～ENDIF文の外へ分岐できます。ただし、IF～ENDIF文の範囲内(ENDIFブロックを含む)への分岐は行わないでください。分岐を行うと、読み飛ばしたIF命令は無効となり、それに対応するENDIFまでの指令をすべて実行します。

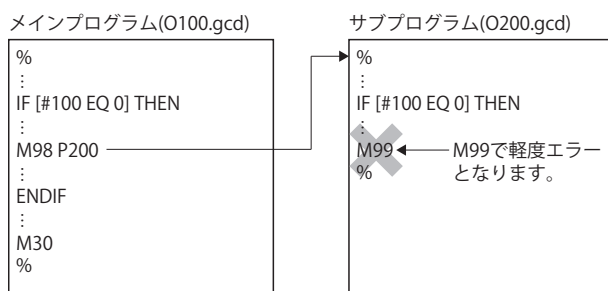
- IF～ENDIF文の範囲内からの分岐



- IF～ENDIF文の範囲内への分岐

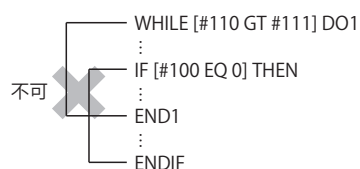


- IF～ENDIF文の範囲内からサブプログラム呼出し(M98), またはマクロ呼出し(G65)を行えます。また、サブプログラム, またはマクロプログラム内でIF/THEN/ELSE/ENDIF命令を行うこともできます。サブプログラム, またはマクロプログラム内でもIF文の深さは、10重までです。(IF文の深さは、プログラムごとに10重まで指令できます。)
- IF～ENDIF文は、同一プログラム内で終了してください。IF～ENDIF文が同一プログラム内で終了しない(ENDIF命令がない)と、軽度エラー (エラーコード: 1FC3H(詳細コード: 0330H))となります。

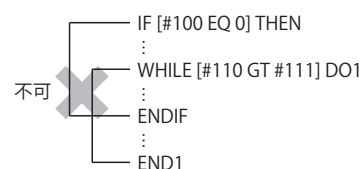


- IF～ENDIF文とWHILE～END文を交差させて使用することはできません。

- WHILE～END文とIF～ENDIF文が交差している場合



- IF～ENDIF文とWHILE～END文が交差している場合



プログラム例

■条件を満たした場合，THEN側，またはELSE側の処理を実行するプログラム

動作	プログラム	備考
(1)	G91	
(2)	G01 X100. Y100. F2000.	
(3)	X200.	
(4)	Y200.	
(5)	IF [#100 EQ 0] THEN	#100が0のとき，THEN～ENDIFまでを実行((5)～(8))
(6)	G01 Y300. F1500.	
(7)	X300.	
(8)	ENDIF	
(9)	G02 X50. Y50. I0. J50. F800.	
(10)	G01 X100. Y500. F2000.	
(11)	IF [#110 GT #120] THEN	#110が#120を超えるとき，THEN～ELSEを実行((11)～(14))
(12)	G00 X10.	
(13)	Y100.	
(14)	ELSE	#110が#120以内のとき，ELSE～ENDIFを実行((14)～(16))
(15)	G28 X0. Y0.	
(16)	ENDIF	

繰り返し(WHILE, DO, END)

実行プログラムの流れを条件により制御します。

コード	フォーマット
WHILE, DO, END	WHILE [条件式] DOm ⋮ ENDm <div style="position: absolute; left: 450px; top: 150px;"> 繰り返し識別番号(1~127) </div>

機能

- 条件が成立している間、次のブロックからENDブロックまで繰り返し実行し、不成立のときは、ENDの次のブロックへ実行が移ります。
- WHILE～DOとENDはペアで使用します。同一プログラム内でWHILE～DOとENDがペアになっていないと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0334H))となります。
- WHILE～DOよりも前にENDを指定することはできません。WHILE～DOよりも前にENDを指定すると軽度エラー (エラーコード: 1FC3H(詳細コード: 0334H))となります。
- WHILE[条件式]を省略すると、DOm～ENDmを無限に繰り返します。
- 繰り返し識別番号mの範囲は、「1～127」です。繰り返し識別番号mに範囲外の値を指定すると、軽度エラー (エラーコード: 1FC3H(詳細コード: 0332H))となります。
- 繰り返し識別番号は、何度でも自由に使用できます。また、繰り返し識別番号は、順番に使用する必要はありません。

```

WHILE [#110 GT #111] DO1
:
END1
:
WHILE [#120 EQ #121] DO3
:
END3
:
WHILE [#130 GT #131] DO1
:
END1

```

- WHILE文の深さは、27重までです。WHILE文の深さが27重を超えると、軽度エラー (エラーコード: 1FC3H(詳細コード: 0333H))となります。また、入れ子の場合、入れ子内で一度使用した繰り返し識別番号は、使用できません。

```

WHILE[#120 GT #121] DO1
:
WHILE[#120 LT #121] DO2
:
:
WHILE[#120 EQ #121] DO27
:
END27
:
END2
:
END1

```

27重 2重 1重

- WHILE～END文の範囲内からWHILE～END文の外へ分岐できます。ただし、WHILE～END文の範囲内への分岐は行えません。WHILE～END文の範囲内への分岐を行うと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0306H))となります。

• WHILE～END文の範囲内からの分岐

```

WHILE [#100 EQ 0] DO1
:
IF [#110 GT #111] GOTO 100
:
END1
:
N100
:

```

• WHILE～END文の範囲内への分岐

```

:
IF [#110 GT #111] GOTO 100
:
WHILE [#100 EQ 0] DO1
:
N100
:
END1

```

不可

• WHILE～END文の範囲内からWHILE～END文の範囲内への分岐

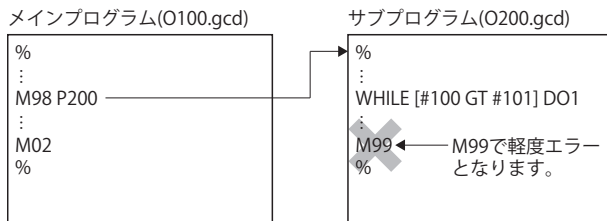
```

WHILE [#100 EQ 0] DO1
:
IF [#110 GT #111] GOTO 100
:
END1
:
WHILE [#100 EQ 0] DO1
:
N100
:
END1

```

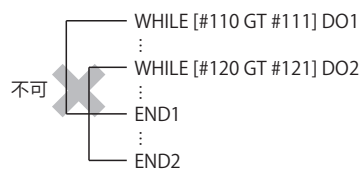
不可

- WHILE～END文の範囲内からサブプログラム呼出し(M98), またはマクロ呼出し(G65)を行うことができます。また、サブプログラム、またはマクロプログラム内でWHILE/DO/END命令を行うこともできます。メインプログラム、およびサブプログラム、またはマクロプログラムを含めWHILE文の深さは、27重までです。
- WHILE～END文は、サブプログラム、またはマクロプログラム内でWHILEとENDがペアとなっていないと、M99実行時に軽度エラー(エラーコード: 1FC3H(詳細コード: 0334H))となります。

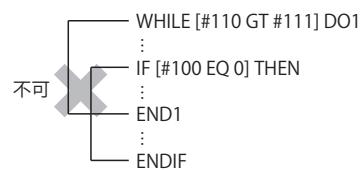


- WHILE～END文とWHILE～END文、またはIF～ENDIF文とWHILE～END文を交差させて使用することはできません。IF～ENDIF文、WHILE～END文を交差させると、軽度エラー(エラーコード: 1FC3H(詳細コード: 0334H))となります。

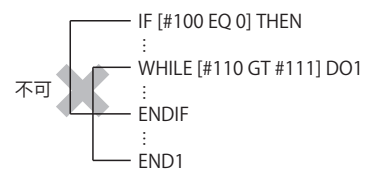
- WHILE～END文とWHILE～END文が交差している場合



- WHILE～END文とIF～ENDIF文が交差している場合



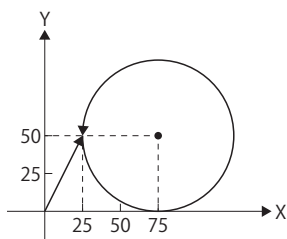
- IF～ENDIF文とWHILE～END文が交差している場合



プログラム例

■条件を満たした場合、指定ブロックを繰り返すプログラム

動作	プログラム	備考
(1)	#100 = 0	
(2)	G91 G00 X25. Y50.	
(3)	WHILE [#100 LT 3] DO1	#110が3未満のとき、(3)～(6)を繰り返す
(4)	G03 X0. Y0. I50. J0. F100.	
(5)	#100 = #100 + 1	1回実行するごとに、変数#100に1を加算
(6)	END1	
(7)	G00 X0. Y0.	



*1 上記Gコードプログラムは、円を3回描きます。

6 補助／応用機能

6.1 Gコード制御と各機能の関係

Gコード制御と各機能の関係を以下に示します。

○: 有効, —: 無効

機能	Gコード制御	内容
トルク制限機能	○	トルク制限値変更命令(M(P)CHGT/D(P).CHGT, CHGT)でトルク制限値の変更が可能です。
外部入力信号(STOP/FLS/RLS)の入力	○	位置決め制御と同様に制御します。 ただし、パラメータブロックの「STOP時の減速処理」の設定が「急停止」の場合でも、停止処理は減速停止となります。「STOP入力時の実現在値」は格納されません。
緊急停止	○	他の制御方式の場合と同様に制御します。
停止指令	—	無視します。
急停止指令	—	無視します。
各軸ストロークリミット	—	無視します。
制御変更	現在値変更	— 無視します。
	速度変更	— 無視します。
	トルク制限値変更	○ 位置決め制御と同様に制御します。
	目標位置変更	— 無視します。
絶対位置システム	○	他の制御方式の場合と同様に制御します。
Mコード出力機能	—	Gコード制御軸に割り付けた軸のMコードに格納されません。
バックラッシュ補正機能	○	他の制御方式の場合と同様に制御します。
停止指令を無視するスキップ機能	—	無視します。
速度・トルク制御	—	<ul style="list-style-type: none"> Gコード制御中、速度・トルク制御を始動できません。軽度エラー (エラーコード: 192AH) となります。 速度・トルク制御中、Gコード制御を開始できません。軽度エラー (エラーコード: 1FC1H(詳細コード: 0117H)) となります。
圧力制御	—	<ul style="list-style-type: none"> Gコード制御中、圧力制御を始動できません。軽度エラー (エラーコード: 192AH) となります。 圧力制御中、Gコード制御を開始できません。軽度エラー (エラーコード: 1FC1H(詳細コード: 0117H)) となります。
ブート時ファイル転送機能	○	Gコード制御システムパラメータ, Gコード制御軸パラメータ, Gコード制御加工パラメータ, Gコードプログラムのブート時ファイル転送が可能です。
パラメータ変更機能	○	Gコード制御システムパラメータ, Gコード制御軸パラメータ, Gコード制御加工パラメータのパラメータ変更が可能です。
オーバライド機能	—	Gコード制御軸に割り付けた軸のオーバライドは無視します。 Gコード制御用のオーバライドを使用します。
制振指令フィルタ	○	他の制御方式の場合と同様に制御します。
サーボモータ最大回転速度チェック	—	無視します。
degree時ABS方向設定	—	無視します。
サーボプログラムによる位置決め制御	—	<ul style="list-style-type: none"> Gコード制御中、他の位置決め制御を始動できません。軽度エラー (エラーコード: 192AH) となります。 他の位置決め制御中、Gコード制御を開始できません。軽度エラー (エラーコード: 1FC1H(詳細コード: 0117H)) となります。
モーション専用シーケンス命令 (M(P).SVSTD/D(P).SVSTD)によるダイレクト位置決め制御	—	
JOG運転	—	
手動パルス運転	—	
アドバンス同期制御	—	
マシンプログラム運転	—	
マシンJOG運転	—	
リミットスイッチ機能	○	他の制御方式の場合と同様に制御します。
マーク検出機能	○	他の制御方式の場合と同様に制御します。
演算周期混在機能	—	使用できません。 Gコード制御と演算周期混在を使用する設定とした場合、マルチCPUシステムの電源投入時に、中度エラー (エラーコード: 30F9H) となります。
SSCNET通信を使用する機能	○	他の制御方式の場合と同様に制御します。
デジタルオシロ	○	Gコード制御用デバイスを使用可能です。

機能	Gコード制御	内容
軸モニタ	○	Gコード制御中の位置決め専用信号をモニタします。 なお、目標速度、指令アドレスはモニタしません。
モーションCPUエラー一括モニタ	○	Gコード制御で検出したエラーを登録します。
スクロールモニタ	—	Gコード制御中の履歴は登録しません。
現在値履歴モニタ	○	他の制御方式の場合と同様に制御します。
テストモード	—	テストモードによるGコード制御はできません。

6.2 補助機能(M機能)

補助機能はM機能とも呼ばれ、Gコード制御の補助的な機能を指令する機能です。

M単独出力

M00, M01, M02, M30が指令されると、“[Md.3058]Mコードデータ1(D54554+128s, D54555+128s)”～ “[Md.3061]Mコードデータ4(D54560+128s, D54561+128s)”, および “[St.3222]補助機能ストロープ1(D54442.4+4s)”～ “[St.3225]補助機能ストロープ4(D54442.7+4s)”とは別に, “[St.3218]M単独出力 M00(D54442.0+4s)”～ “[St.3221]M単独出力 M30(D54442.3+4s)”を出力します。

M単独出力を使用する場合、Gコード制御は, “[St.3218]M単独出力 M00(D54442.0+4s)”～ “[St.3221]M単独出力 M30(D54442.3+4s)”を出力するだけで、動作および完了信号の処理は、ユーザ側で作成する必要があります。

“[St.3218]M単独出力 M00(D54442.0+4s)”～ “[St.3221]M単独出力 M30(D54442.3+4s)”は、補助機能完了後、または “[Rq.3380]リセット指令(D54226.4+2s)”をOFF→ONするとOFFします。同一ブロックに移動指令、ドウェルがある場合は、移動指令、ドウェルが完了したときに出力します。M単独出力指令を単独で指令した場合でも、前の指令ブロックの軸移動が完了後に出力します。ただし、移動指令、ドウェルの完了前に “[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”, または “[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”をONしたときは出力しません。

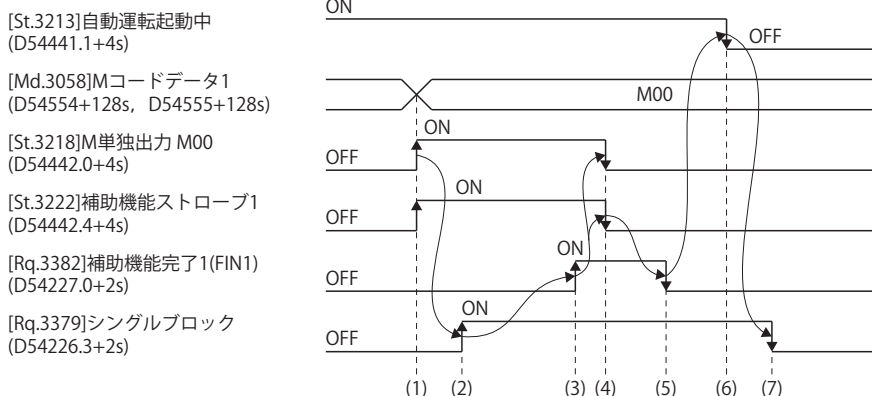
Mコード	M単独出力信号	操作
M00	[St.3218]M単独出力 M00(D54442.0+4s)	“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”, または “[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”をOFF→ON
M01	[St.3219]M単独出力 M01(D54442.1+4s)	“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”, または “[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”をOFF→ON
M02	[St.3220]M単独出力 M02(D54442.2+4s)	“[Rq.3380]リセット指令(D54226.4+2s)”をOFF→ON (“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”, または “[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”はONしないでください。)
M30	[St.3221]M単独出力 M30(D54442.3+4s)	“[Rq.3380]リセット指令(D54226.4+2s)”をOFF→ON (“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”, または “[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”はONしないでください。)

Point

特定の補助機能(M00, M01, M02, M30)でも通常の “[St.3222]補助機能ストロープ1(D54442.4+4s)”～ “[St.3225]補助機能ストロープ4(D54442.7+4s)”, “[Md.3058]Mコードデータ1(D54554+128s, D54555+128s)”～ “[Md.3061]Mコードデータ4(D54560+128s, D54561+128s)”は出力されます。

使用例

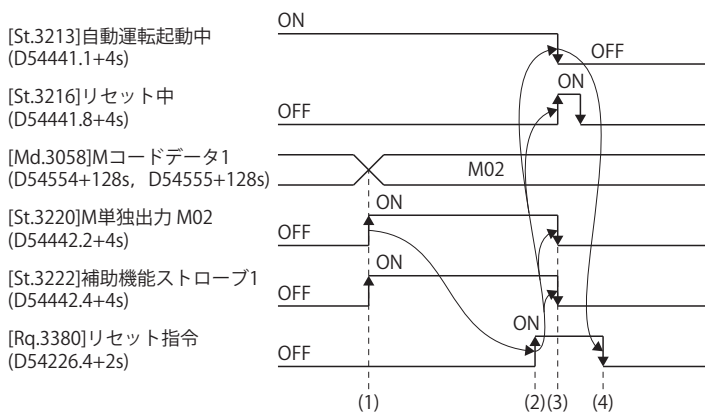
■M00指令の状態によりブロック停止／継続実行を切替える場合



【動作説明】

- モーションCPUにて，“[Md.3058]Mコードデータ1(D54554+128s, D54555+128s)”と“[St.3218]M単独出力 M00(D54442.0+4s)”，および“[St.3222]補助機能ストロープ1(D54442.4+4s)”を出力する。
- ユーザにて，“[St.3218]M単独出力 M00(D54442.0+4s)”がONしたことを確認し，“[Rq.3379]シングルブロック(D54226.3+2s)”をONする。
- (2)の処理が完了後，ユーザにて，“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”をONする。
- モーションCPUにて，“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”のONを確認し，“[St.3218]M単独出力 M00(D54442.0+4s)”，および“[St.3222]補助機能ストロープ1(D54442.4+4s)”をOFFする。
- ユーザにて，“[St.3222]補助機能ストロープ1(D54442.4+4s)”のOFFを確認し，“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”をOFFする。
- モーションCPUにて，“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”のOFF，“[Rq.3379]シングルブロック(D54226.3+2s)”の状態，および現在のブロックの完了を確認して，“[St.3213]自動運転起動中(D54441.1+4s)”をOFFする。
- ユーザにて，“[St.3213]自動運転起動中(D54441.1+4s)”のOFFを確認し，次の自動運転起動前に，“[Rq.3379]シングルブロック(D54226.3+2s)”をOFFする。

■M02指令でリセットを行う場合



【動作説明】

- モーションCPUにて，“[Md.3058]Mコードデータ1(D54554+128s, D54555+128s)”と“[St.3220]M単独出力 M02(D54442.2+4s)”，および“[St.3222]補助機能ストロープ1(D54442.4+4s)”を出力する。
- ユーザにて，“[St.3220]M単独出力 M02(D54442.2+4s)”がONしたことを確認し，所定の動作を行う。所定の動作完了後，“[Rq.3380]リセット指令(D54226.4+2s)”をOFF→ONする。
- モーションCPUにて，“[Rq.3380]リセット指令(D54226.4+2s)”のONを確認し，“[St.3220]M単独出力 M02(D54442.2+4s)”，および“[St.3222]補助機能ストロープ1(D54442.4+4s)”をOFFする。
- ユーザにて，“[St.3213]自動運転起動中(D54441.1+4s)”のOFFを確認し，“[Rq.3380]リセット指令(D54226.4+2s)”をOFFする。

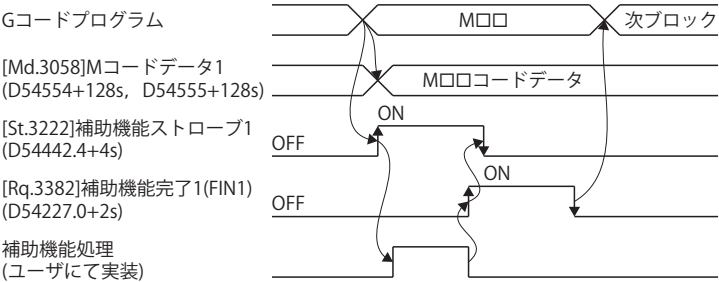
補助機能完了

補助機能(M機能)が指令され、モーションSFCプログラムやシーケンスプログラムにて、処理および完了シーケンスを作成することで、完了信号を待ってから次のブロックに進ませる機能です。完了信号には、下記に示す2種類があります。目的の動作に合わせ使い分けことができます。

補助機能完了信号	動作
[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)	完了信号(FIN1)の立下り時、次のブロックに進みます。
[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)	完了信号(FIN2)の立上り時、次のブロックに進みます。

補助機能完了1(FIN1)

FIN1信号を使用した場合の動作を下記に示します。モーションSFCプログラムやシーケンスプログラムでは、補助機能ストローブ信号がOFFとなったことを確認して、“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”をOFFする処理としてください。



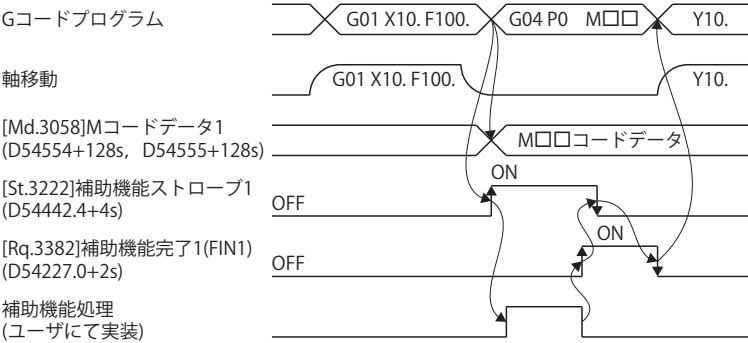
■注意事項

- “[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”は、補助機能(M機能)共通で使用するため、すべての機能動作が完了したという条件でONしてください。
- 補助機能(M機能)の指令前に“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”がONしている場合、補助機能(M機能)関連データは出力されません。
- M02指令、またはM30指令を行うときは、“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”ではなく、“[Rq.3380]リセット指令(D54226.4+2s)”をOFF→ONしてください。Gコードプログラム最後のM02指令、またはM30指令で“[Rq.3382]補助機能完了1(FIN1)(D54227.0+2s)”をONした場合、軽度エラー(エラーコード: 1FC3H(詳細コード: 0309H))となります。
- 切削指令の次ブロックにM指令した場合、切削指令の減速が完了する前にMコードデータを出力します。軸の移動が完了した後に、M指令を実行させるときは、切削指令とM指令の間にブロック(遅延時間なしのG04)を挿入してください。

■プログラム例

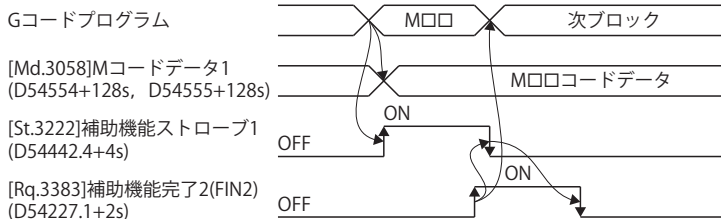
- 軸の移動が完了した後にM指令を実行する場合

プログラム
⋮
G01 X10. F100.
G04 P0 M□□
Y10.
⋮



補助機能完了2(FIN2)

FIN2信号を使用した場合の動作を下記に示します。モーションSFCプログラムやシーケンスプログラムでは、補助機能ストロープ信号がOFFとなったことを確認して、“[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”をOFFする処理としてください。

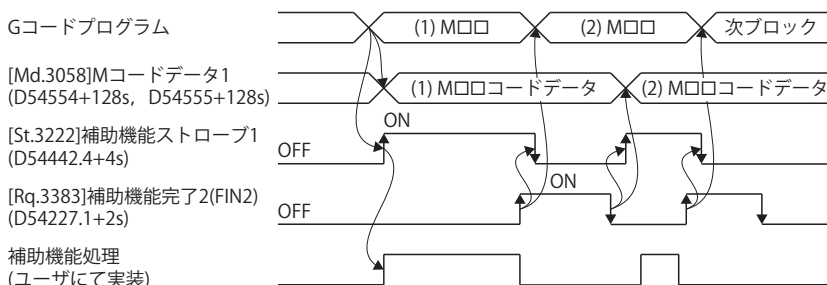


■注意事項

- “[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”は、補助機能(M機能)共通で使用するため、すべての機能動作が完了したという条件でONしてください。
- 補助機能(M機能)の指令前に“[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”がONしている場合、補助機能(M機能)関連データは出力されません。

例

M指令が連続する場合



- ブロック切り後、“[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”がOFFしたタイミングで次ブロックの“[Md.3058]Mコードデータ1(D54554+128s, D54555+128s)”～“[Md.3061]Mコードデータ4(D545560+128s, D545561+128s)”と“[St.3222]補助機能ストロープ1(D54442.4+4s)”～“[St.3225]補助機能ストロープ4(D54442.7+4s)”が出力されます。
- M02指令、またはM30指令を行うときは、“[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”ではなく、“[Rq.3380]リセット指令(D54226.4+2s)”をOFF→ONしてください。Gコードプログラム最後のM02指令、またはM30指令で“[Rq.3383]補助機能完了2(FIN2)(D54227.1+2s)”をONした場合、軽度エラー(エラーコード: 1FC3H(詳細コード: 0309H))となります。

6.3 送り機能

早送り速度

早送り速度は、G00指令による位置決めを行う際の位置決め速度です。

早送り速度は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“速度・時定数”⇒“早送り速度”にて設定します。

位置決め時の経路には、始点から終点までを直線で補間する「補間型」と各軸の早送り速度で移動する「非補間型」があります。補間型/非補間型の設定は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“制御設定”⇒“G00 非補間”にて設定します。どちらを設定しても位置決め時間は同じです。

加減速時定数は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“速度・時定数”⇒“G0時定数(リニア)”にて設定します。

早送林中、“[Md.3018]速度(D54502+128s, D54503+128s)”に格納される速度は、Gコード制御系統パラメータの「G00非補間」の設定により異なります。

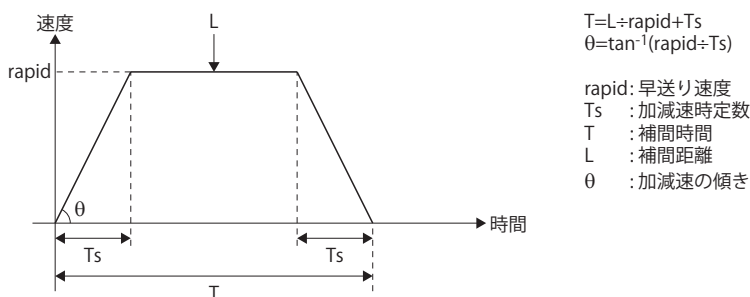
G00非補間	[Md.3018]速度(D54502+128s, D54503+128s)
0: 終点に向け、直線で移動する。(補間タイプ)	現在移動中のベクトル方向の速度を格納します。
1: 各軸の終点に向け、各軸の早送り速度で移動する。(非補間タイプ)	移動中の軸において、最も速度が大きい軸の速度を格納します。

高精度制御モード中の送り速度は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“高精度制御”⇒“高精度制御モード用早送り速度”にて設定します。ただし、高精度制御モード用早送り速度の設定値が「0」の場合は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“速度・時定数”⇒“早送り速度”にて設定した速度で移動します。

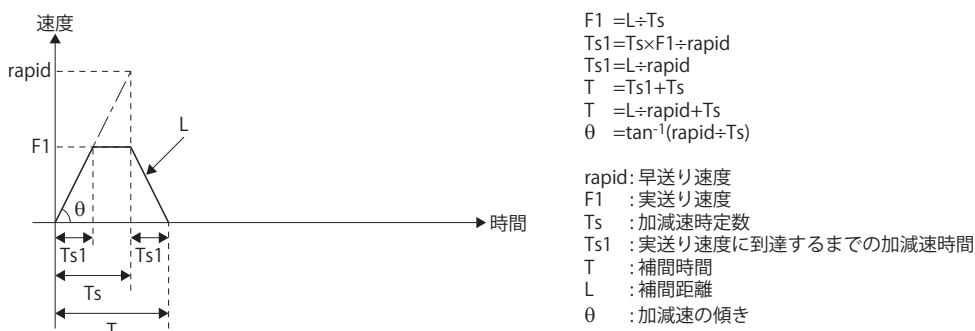
高精度制御モード用早送り速度は、各軸ごとに設定できます。

G00指令は、切削モード/高精度制御モードに関わらず、時間一定加減速方式で加減速処理します。

補間距離が長く、速度が早送り速度に到達する場合



補間距離が短く、速度が早送り速度に到達しない場合



切削送り速度

切削送り速度は、アドレスFと1分間当たりの送り速度[mm/min]を設定します。

切削送り速度は、G01指令、G02指令、G03指令に対して有効となります。

切削送り速度は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“速度・時定数”⇒“切削送りクランプ速度”にて設定した値でクランプします。

高精度制御モード中の切削送り速度は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“高精度制御”⇒“高精度制御モード用切削送りクランプ速度”にて設定した値でクランプします。ただし、高精度制御モード用切削送りクランプ速度の設定値が「0」の場合は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“速度・時定数”⇒“切削送りクランプ速度”にて設定した値でクランプされます。

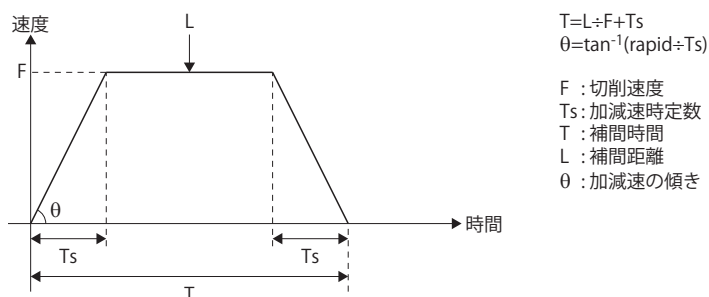
切削送り中、“[Md.3018]速度(D54502+128s, D54503+128s)”には、補間軸の合成速度が格納されます。

イグザクトストップチェックモード/自動コーナオーバーライド/切削モード中の加減速処理

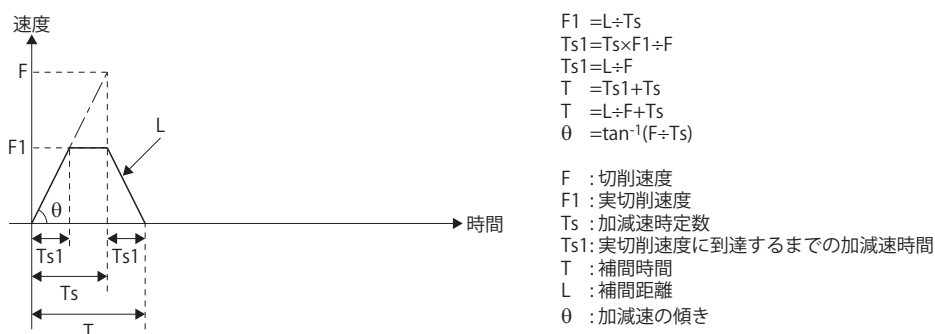
イグザクトストップチェックモード/自動コーナオーバーライド/切削モード中のG01指令、G02指令、G03指令は、時間一定加減速方式で加減速処理します。

加減速時定数は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“速度・時定数”⇒“G1時定数(リニア)”にて設定します。

■補間距離が長く、速度が切削速度に到達する場合



■補間距離が短く、速度が切削速度に到達しない場合



高精度制御モード中の加減速処理

高精度制御モード中のG01指令、G02指令、G03指令は、傾き一定加減速方式で加減速処理します。

高精度制御モード中の切削送り加減速処理についての詳細は、補間前加減速を参照してください。(257ページ 補間前加減速)

送り速度の指定と各制御軸への効果

制御軸は、直線運動を制御する直線軸と回転運動を制御する回転軸に分けられます。送り速度は、これらの軸の変位速度を指定するものであり、直線軸を制御する場合と回転軸を制御する場合で、切削において問題となる工具移動速度におよぼす効果が異なります。また、各軸の変位量は軸ごとに指定しますが、送り速度は軸ごとに指定するのではなく、1つの数値で指定します。2軸以上を補間する場合、各軸への作用の仕方を以下に説明します。

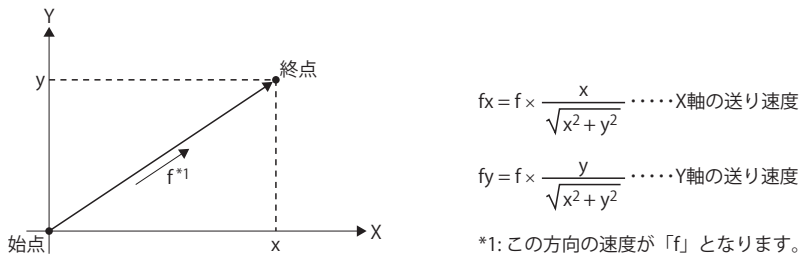
直線軸を制御する場合

1軸のみ制御する場合、および2軸以上の制御軸が同時に制御される場合でも、Fで指定された送り速度は工具進行方向の線速度として作用します。

■直線軸のみ制御

例

送り速度の指定を「f」とし、直線軸(X, Y軸)を制御する場合

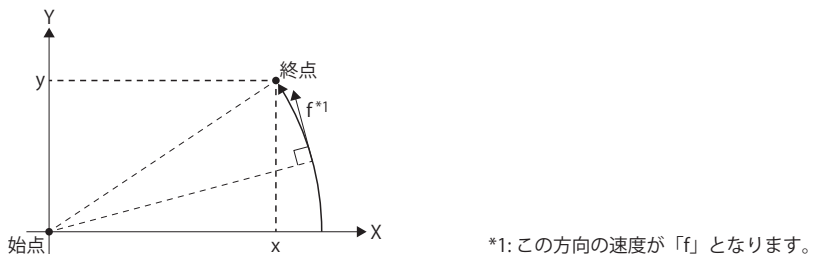


直線軸のみ制御する場合、プログラムでは切削速度を指定すればよいことになります。各軸の送り速度は、指定された送り速度を移動量に対応した成分に分解したものとなります。

■円弧補間を用いて直線軸を制御

例

送り速度の指定を「f」とし、円弧補間を用いて直線軸(X, Y軸)を制御する場合
工具進行方向(接線方向)の速度がプログラム指定された送り速度となります。



この場合、X, Yそれぞれの軸の送り速度は、工具の移動とともに変化します。ただし、合成された速度は常に一定値fに保たれます。

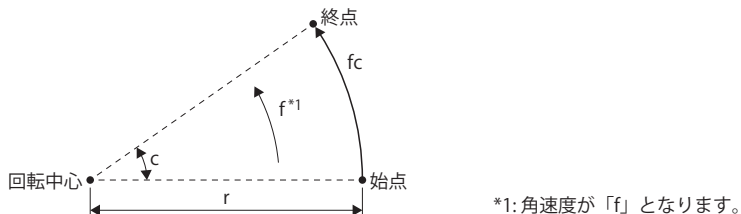
回転軸を制御する場合

回転軸を制御する場合、指定された送り速度は、回転軸の回転速度(角速度)として作用します。

従って、工具進行方向の切削速度(線速度)は、回転中心と工具との距離により変化します。プログラムで指定する速度は、この距離を考慮してください。

例

送り速度の指定を「f」とし、回転軸(C軸)を制御する場合(fの単位: [degree/min])



この場合、工具進行方向の切削速度(線速度)を「fc」とするためには、

$$f_c = f \times \frac{\pi \times r}{180}$$

となります。

従って、プログラムで指定する送り速度は、下記のようにします。

$$f = f_c \times \frac{180}{\pi \times r}$$

直線軸制御と回転軸制御が同時に行われる場合

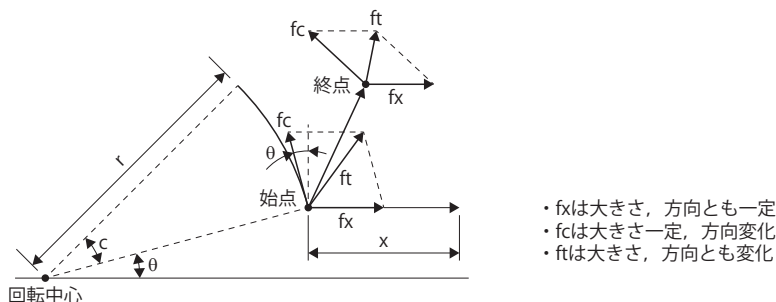
制御装置は直線軸を制御する場合、および回転軸を制御する場合も、すべて同じ扱いをします。

回転軸を制御する場合、座標語(A, B, C)で与えられる数値は角度であり、送り速度(F)で与えられる数値はすべて線速度として扱います。すなわち、回転軸の1[degree]は、直線軸1[mm]と等価なものとして扱います。

従って、直線軸と回転軸が同時に制御される場合、Fで与えられた数値の各軸に対応する成分は、「直線軸を制御する場合」のときと同様になります。ただし、この場合の直線軸制御による速度成分の大きさと方向はともに変化しませんが、回転軸制御による速度成分の方向が工具の移動とともに変化する(大きさは変化しない)ため、結果として合成された工具進行方向の送り速度は、工具の移動とともに変化します。

例

送り速度の指定をfとし、直線軸(X軸)と回転軸(C軸)を同時制御する場合
X軸増分指令値を「x」、C軸増分指令値を「c」とすると、



X軸の送り速度(線速度)「fx」および、C軸の送り速度(角速度)「ω」は、

$$f_x = f \times \frac{x}{\sqrt{x^2 + c^2}} \quad \dots\dots(1)$$

$$\omega = f \times \frac{c}{\sqrt{x^2 + c^2}} \quad \dots\dots(2)$$

となります。

C軸制御による線速度「fc」は、

$$f_c = \omega \times \frac{\pi \times r}{180} \quad \dots\dots(3)$$

となります。

始点における工具進行方向の速度を「ft」とし、そのX軸方向の分速度を「ftx」、Y軸方向の分速度を「fty」とすると、

$$f_{tx} = -r \sin\left(\frac{\pi}{180} \theta\right) \times \frac{\pi}{180} \omega + f_x \quad \dots\dots(4)$$

$$f_{ty} = -r \cos\left(\frac{\pi}{180} \theta\right) \times \frac{\pi}{180} \omega \quad \dots\dots(5)$$

となります。

ここで「r」は回転中心と工具との距離(単位: mm)、「θ」は回転中心において、始点とX軸間の角度(単位: °)となります。

(1)～(5)の式により合成速度「ft」は、

$$f_t = \sqrt{f_{tx}^2 + f_{ty}^2}$$

$$= f \times \frac{\sqrt{x^2 - x \times c \times r \sin\left(\frac{\pi}{180} \theta\right) \frac{\pi}{90} + \left(\frac{\pi \times r \times c}{180}\right)^2}}{x^2 + c^2} \quad \dots\dots(6)$$

となります。

従って、プログラムで指定する送り速度「f」は、

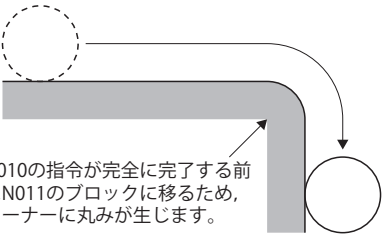
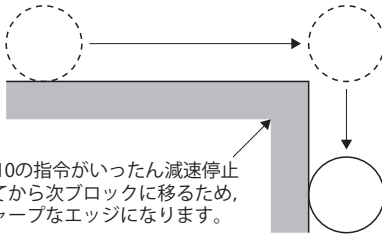
$$f = f_t \times \frac{x^2 + c^2}{\sqrt{x^2 - x \times c \times r \sin\left(\frac{\pi}{180} \theta\right) \frac{\pi}{90} + \left(\frac{\pi \times r \times c}{180}\right)^2}} \quad \dots\dots(7)$$

とします。

ただし、(6)の式の「ft」は、始点における速度であり、C軸が回転して行くに従い、θの値は変化し、それにより「ft」の値が変化します。従って、切削速度(ft)をできるだけ一定に保つためには、1ブロックで指定する回転角度をできるだけ小さくし、θの値の変化幅を小さくしてください。

減速チェック

減速チェック機能は、軸移動ブロックの継ぎ目において、いったん減速停止を行ってから次ブロックを実行することにより、制御軸の送り速度が急激に変化するときの機械ショックを緩和し、コーナ切削時における丸みを防止することができます。

減速チェックなし	減速チェックあり
<p>N010 G01 X100. N011 G01 Y-50.</p>  <p>N010の指令が完全に完了する前にN011のブロックに移るため、コーナーに丸みが生じます。</p>	<p>N010 G09 G01 X100. N011 G01 Y-50.</p>  <p>N010の指令がいったん減速停止してから次ブロックに移るため、シャープなエッジになります。</p>

減速チェックの実行条件

■早送り時の減速チェック

早送りモードの場合は、ブロックの移動完了時に必ず減速チェックを行ってから次ブロックを実行します。

■切削送り時の減速チェック

切削送りモードの場合は、下記条件のいずれかを満たすときに減速チェックを実行し、チェック完了後に次ブロックの移動を開始します。

- ・ G61(イグザクトストップチェックモード)が選択されているとき。
- ・ 同一ブロックにG09(イグザクトストップチェック)が指令されたとき。
- ・ 現ブロックと次ブロックの移動方向が反転する軸があり、移動指令の組合せの条件を満たすとき。(詳細は、移動方向反転時の減速チェックを参照(213ページ 移動方向反転時の減速チェック))

減速チェック方式

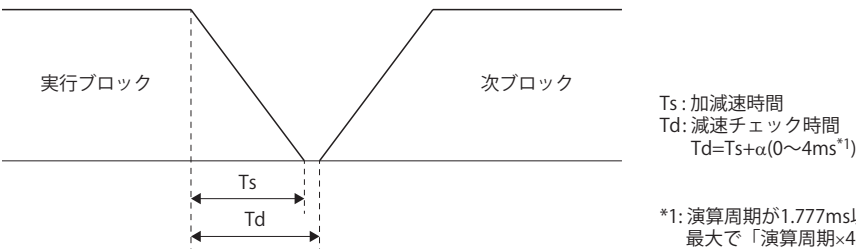
減速チェック方式には、指令減速チェック方式、スムージングチェック方式、インポジションチェック方式の3つの方式があります。

減速チェック方式は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“制御設定”⇒“減速チェック”にて設定します。

■指令減速チェック方式

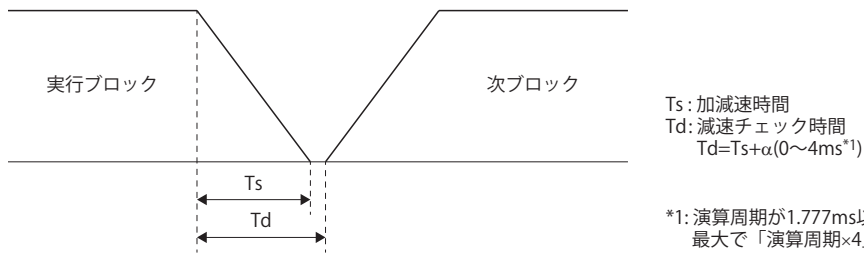
1ブロックの補間が完了した後、指令系の減速が完了したことを確認してから次ブロックの実行を開始します。以降は早送り実行中のブロックから次ブロックへの移行の例を説明します。

減速チェックに要する時間は、同時に指令された軸の加減速モード、および加減速時定数によって決まる各軸の減速チェック時間のうち最も長いものとなります。



■スレーシングチェック方式

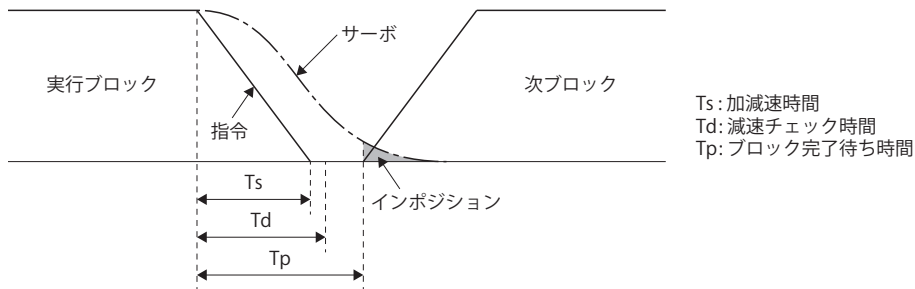
指令減速チェックを行った後、系統内全軸の位置指令が目標位置に到達したことを確認してから、次ブロックの実行を開始します。



*1: 演算周期が1.777ms以上の場合は、
最大で「演算周期×4」になります。

■インポジションチェック方式

指令減速チェックを行った後、さらに系統内全軸のモータ実位置の残距離が一定値以下になったことを確認してから、次ブロックを実行します。残距離の確認は、インポジション幅により行います。サーボパラメータの「インポジション範囲 (PA10)」をインポジション幅とします。



インポジション幅は上図に示すように、次ブロック開始時の前ブロックの残距離です。(■部分の面積)

減速チェックは、位置決め時間の短縮を目的としています。インポジション幅の設定値を大きくすれば短縮時間が大きくなりますが、次ブロック開始時の前ブロックの残距離も大きくなり、実加工に支障が生じることがあります。

残距離のチェックは、Gコード制御演算周期ごとに行います。そのためインポジション幅の設定値どおりの位置決め時間短縮効果がでないことがあります。

移動方向反転時の減速チェック

切削送りで移動方向が反転する場合、減速チェックは下記ようになります。

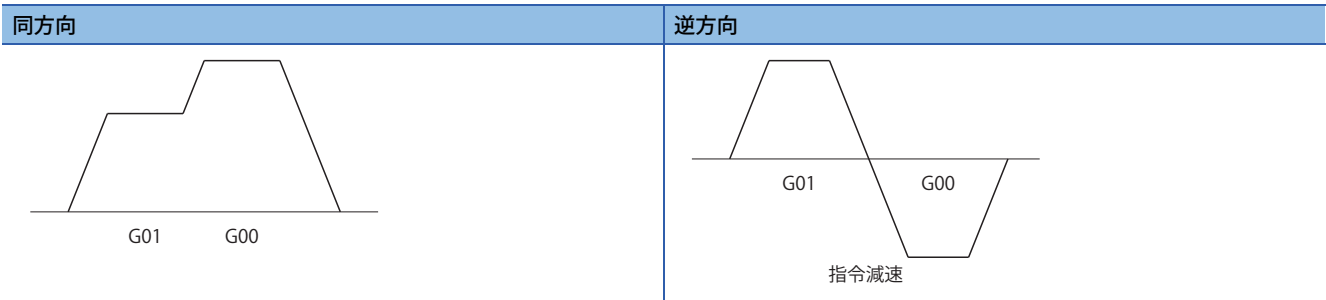
○: 減速チェックする, ×: 減速チェックしない

現ブロック	次ブロック		
	G00	G01	G00/G01以外
G01	○	○/×*1	×
G02	×	×	×
G03	×	×	×

*1 パラメータの設定により異なります。詳細はG1→G1の移動方向を参照してください。(214ページ G1->G1の移動方向反転)

G1->G0の移動方向反転

G1→G0の移動方向反転時、補間が完了し減速チェックを行った後、次ブロックの移動を開始します。速度指令が重ならず、加速度が過大となることを回避します。複数軸で補間する場合、1軸でも逆方向に移動反転すれば、減速チェックを行います。



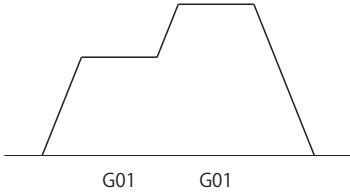
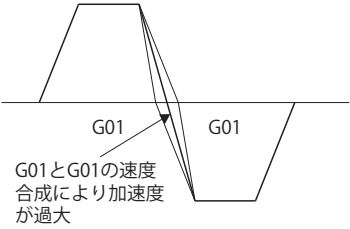
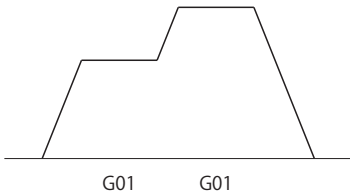
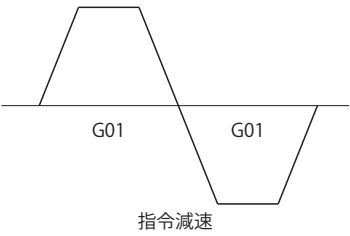
例

複数軸を移動させる場合のプログラム例を下記に示します。

移動方向	プログラム	内容
X軸(正方向→負方向) Y軸(正方向→正方向)	G91 G01 X100. Y100. F4000G00 X-100. Y120.	X軸が逆方向への移動のため、減速チェックを行います。
X軸(正方向→正方向) Y軸(負方向→正方向)	G91 G01 X100. Y-100. F4000G00 X80. Y100.	Y軸が逆方向への移動のため、減速チェックを行います。
X軸(正方向→負方向) Y軸(正方向→正方向)	G90 G01 X100. Y100. F4000G00 X80. Y120.	X軸が逆方向への移動のため、減速チェックを行います。(プログラム開始位置がX0, Y0の時)
X軸(正方向→正方向) Y軸(正方向→正方向)	G91 G01 X100. Y100. F4000G00 X100. Y100.	X軸、およびY軸ともに同方向の移動のため、減速チェックは行いません。
X軸(正方向→正方向) Y軸(正方向→移動なし)	G91 G01 X100. Y80. F4000G00 X80.	X軸が同方向の移動でY軸の移動指令がないため、減速チェックは行いません。

■G1->G1の移動方向反転

G1→G1の移動方向反転時、減速チェックする/減速チェックしないを、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“制御設定”⇒“G1→G1減速チェック”で設定します。

G1->G1減速チェック	内容	同方向	逆方向
0: 減速チェック実施しない	移動方向反転時、補間が完了した後に減速チェックを行わず、次ブロックの移動を開始します。 速度指令が重なり、加速度が過大となることがあります。		
1: 減速チェック実施する	移動方向反転時、補間が完了した後に減速チェックを行ってから、次ブロックの移動を開始します。 速度指令が重ならず、加速度が過大となることを回避します。 複数軸で補間する場合、1軸でも逆方向に移動反転すれば、減速チェックを行います。		

例

「1: 減速チェック実施する」を設定時、複数軸を移動させる場合のプログラム例を下記に示します。

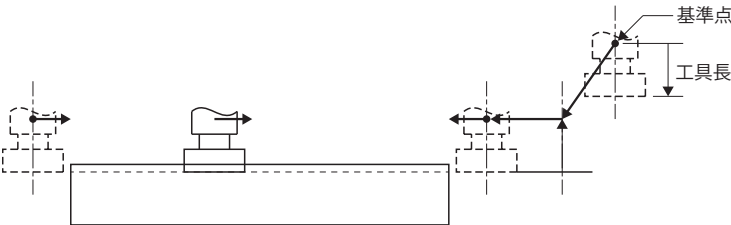
移動方向	プログラム	内容
X軸(正方向→負方向) Y軸(正方向→正方向)	G91 G01 X100. Y100. F4000G01 X-100. Y120.	X軸が逆方向への移動のため、減速チェックを行います。
X軸(正方向→正方向) Y軸(負方向→正方向)	G91 G01 X100. Y-100. F4000G01 X80. Y100.	Y軸が逆方向への移動のため、減速チェックを行います。
X軸(正方向→負方向) Y軸(正方向→正方向)	G90 G01 X100. Y100. F4000G01 X80. Y120.	X軸が逆方向への移動のため、減速チェックを行います。(プログラム開始位置がX0, Y0の時)
X軸(正方向→正方向) Y軸(正方向→正方向)	G91 G01 X100. Y100. F4000G01 X100. Y100.	X軸、およびY軸ともに同方向の移動のため、減速チェックは行いません。
X軸(正方向→正方向) Y軸(正方向→移動なし)	G91 G01 X100. Y80. F4000G01 X80.	X軸が同方向の移動でY軸の移動指令がないため、減速チェックは行いません。

6.4 工具補正機能

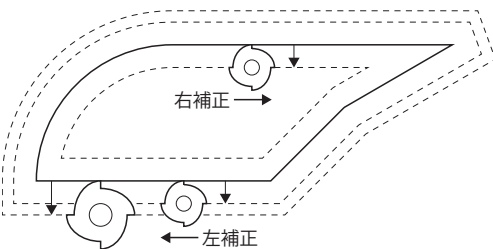
工具補正

工具補正機能には、下図に示すように工具長補正、工具径補正の2種類があります。各補正量は[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御加工パラメータ]⇒“工具補正データ”にて設定します。使用するときは、工具補正番号で指定します。

- 工具長補正



- 工具径補正



工具補正メモリ

工具補正量には、形状補正量を設定します。

1つの補正番号に対して工具長に関する形状補正量、工具径に関する形状補正量を独立して設定します。

Gコードプログラムでは、「H」で工具長の補正量、「D」で工具径の補正量を指定します。

■工具補正量

工具長、および工具径に関する形状補正量は、下図の位置に関しての補正量となります。

補正量の設定範囲は、「-9999.9999～9999.9999[mm]」となります。

工具長補正	工具径補正

工具補正番号(H/D)

工具補正番号を指定するアドレスは、「H」、および「D」を使用します。

工具補正番号は、系統間共通で最大40組あります。

Hは工具長補正、Dは工具径補正に使用します。

工具補正	設定範囲
工具長補正	H01～H40
工具径補正	D01～D40

■注意事項

- 工具補正番号はモダルデータです。一度指定されると、新たに「H」、または「D」が指令されるまで有効となります。
- 工具補正番号指令は、1ブロックに1回のみです。1ブロックに2回以上指定した場合は、最後の補正番号指令が有効になります。
- 工具補正番号が範囲外の場合、軽度エラー (エラーコード: 1FC3H(詳細コード: 030DH))となります。
- H0、またはD0を指定した場合、工具長補正、または工具径補正はキャンセルされます。

工具長補正

軸ごとに移動指令の終点位置を、工具補正量で設定した補正量分だけ補正した位置へ変更して移動制御を行うことができます。G43指令、G44指令、G49指令を使用することで、工具の長さによるプログラム座標位置と機械の工具先端位置との距離差を補正できます。

工具径補正

工具の半径を補正する機能で、Gコードの指令およびD指令により、プログラム軌跡に対し、工具補正量で設定した補正量分だけ任意のベクトル方向に補正することができます。

工具径補正の開始動作、および終了動作には、タイプAとタイプBの2通りがあります。工具径補正のタイプは、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御加工パラメータ]⇒“工具径補正”⇒“工具径補正タイプ”にて設定します。

工具径補正開始動作

補正キャンセル状態のとき、下記の条件をすべて満たしたとき、工具径補正を開始します。補正開始時は、シングルブロック運転に関わらず、必ず移動指令を3ブロック、または移動指令が3ブロックなければ最大5ブロック先読みを行ってから実行します。また、補正モード中も同様に最大5ブロック先読みして補正演算を行います。

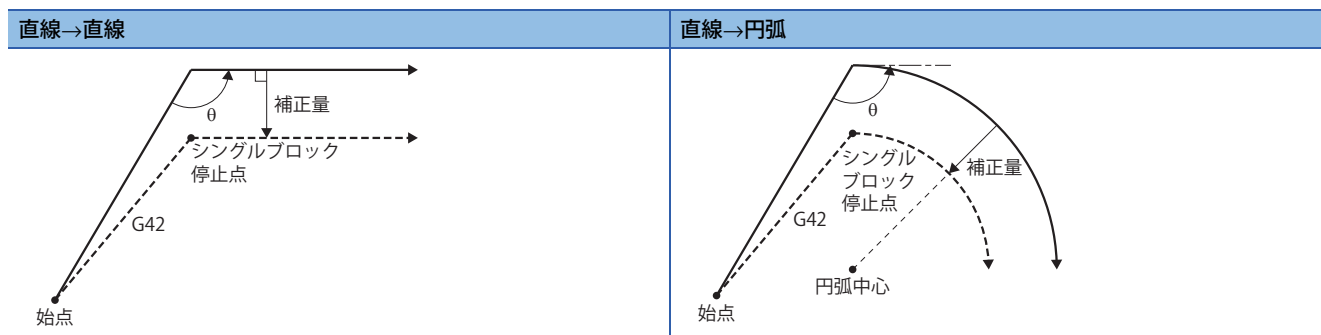
- 補正指令(G41、G42)を指令したとき。
- 工具径補正の補正番号が範囲内のとき。
- 移動指令が、位置決め(G00)、または直線補間(G01)のとき。

Point

補正指令(G41、G42)と同一ブロック内に移動指令がない場合は、次の移動ブロックの方向と垂直に補正動作を行います。

■コーナ内側の場合($\theta \leq 180^\circ$)

—→: プログラム経路, - - -→: 工具中心経路, ———: 交点での円弧の接線, - - - - : プログラム経路の平行線



■コーナ外側(鈍角)の場合($90^\circ \leq \theta < 180^\circ$)

——→: プログラム経路, - - - ->: 工具中心経路, ———: 交点での円弧の接線, - - - ->: プログラム経路の平行線

工具径補正タイプ	直線→直線	直線→円弧
タイプA		
タイプB		

■コーナ外側(鋭角)の場合($\theta < 90^\circ$)

——→: プログラム経路, - - - ->: 工具中心経路, ———: 交点での円弧の接線, - - - ->: プログラム経路の平行線

工具径補正タイプ	直線→直線	直線→円弧
タイプA		
タイプB		

補正モード中の動作

移動指令(G00, G01, G02, G03)にて指令されたプログラム経路に対し、直線/円弧から工具中心経路を求め、補正を行います。

Point

- 補正モード中に同一の補正指令(G41/G42)を指令しても無視します。
- 補正モード中に移動を伴わないブロックを4ブロック以上連続して指令すると、切り込み過ぎや切り込み不足が発生します。
- 補正モード中にM00, またはM01を指令した場合は、先読み禁止になります。

■コーナ内側(鈍角)の場合($90^\circ \leq \theta < 180^\circ$)

—→: プログラム経路, - - -→: 工具中心経路, ———: 交点での円弧の接線, - - - - : プログラム経路の平行線

直線→直線	直線→円弧
円弧→直線	円弧→円弧

■コーナ内側(鋭角)の場合($\theta < 90^\circ$)

—→: プログラム経路, - - -→: 工具中心経路, ———: 交点での円弧の接線, - - - - : プログラム経路の平行線

直線→直線	直線→円弧
円弧→直線	円弧→円弧

■コーナ外側(鈍角)の場合($90^\circ \leq \theta < 180^\circ$)

——→: プログラム経路, - - - ->: 工具中心経路, ———: 交点での円弧の接線, - - - - : プログラム経路の平行線

直線→直線	直線→円弧
円弧→直線	円弧→円弧

■コーナ外側(鋭角)の場合($\theta < 90^\circ$)

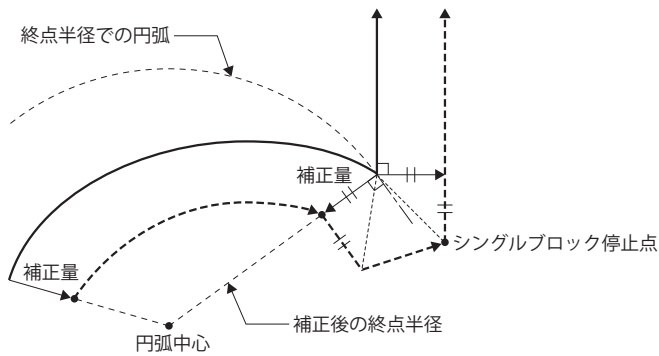
——→: プログラム経路, - - - ->: 工具中心経路, ———: 交点での円弧の接線, - - - - : プログラム経路の平行線

直線→直線	直線→円弧
円弧→直線	円弧→円弧

■円弧の終点が円弧上にない場合

始点半径と終点半径の差が円弧誤差の範囲内のときは、指令終点へ渦巻状になめらかにつなぐ円弧補間で動作します。円弧誤差範囲外の場合は、軽度エラー (エラーコード: 1FC3H(詳細コード: 0313H))となります。

——▶: プログラム経路, - - -▶: 工具中心経路, ———: 交点での円弧の接線, ———: プログラム経路の平行線



■内側の交点が存在しなかった場合

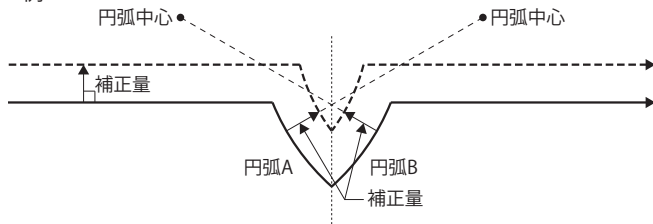
補正量によっては、円弧Aと円弧Bの交点が存在しなくなることがあります。(下図の例2)

その場合、前ブロックの終点で、軽度エラー (エラーコード: 1FC3H(詳細コード: 031DH))となります。

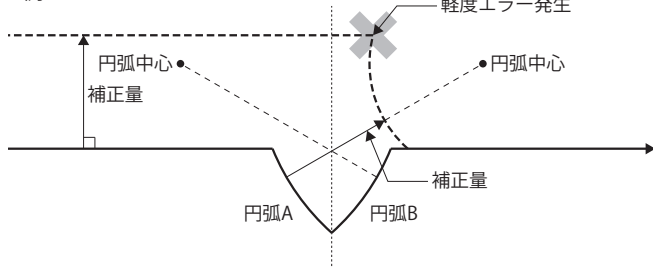
下図の例1は、補正量が小さいため、交点が存在し、加工できます。ただし、例2は、補正量が大きく交点が存在しないため、加工できません。

——▶: プログラム経路, - - -▶: 工具中心経路

<例1>



<例2>



工具径補正キャンセル動作

下記の条件下では、工具径補正は補正キャンセルモードとなります。補正キャンセルモードでは、補正ベクトルは「0」で、工具中心経路はプログラム経路と一致します。工具補正を含むGコードプログラムは、必ず補正キャンセル状態で終了させてください。

- Gコード制御移行後のとき。
- リセット(リセット機能付きのM02, M30を含む)を行ったとき。
- 工具径補正キャンセル指令(G40)を指令したとき。
- 補正番号にD00を指定したとき。

Point

- 工具径補正をキャンセルするときは、円弧指令以外の移動指令で行ってください。円弧指令で補正キャンセル指令を行うと、軽度エラー(エラーコード: 1FC3H(詳細コード: 0315H))となります。
- 補正キャンセル指令が読み込まれた後は、補正キャンセルモードとなり、5ブロックの先読みを中止し、1ブロックの先読みとなります。

■コーナ内側の場合

—→: プログラム経路, ----→: 工具中心経路, ———: 交点での円弧の接線, - - - - -: プログラム経路の平行線

直線→直線	直線→円弧

■コーナ外側(鈍角)の場合($90^\circ \leq \theta < 180^\circ$)

—→: プログラム経路, ----→: 工具中心経路, ———: 交点での円弧の接線, - - - - -: プログラム経路の平行線

工具径補正タイプ	直線→直線	直線→円弧
タイプA		
タイプB		

■コーナ外側(鋭角)の場合($\theta < 90^\circ$)

—→: プログラム経路, ----→: 工具中心経路, ———: 交点での円弧の接線, - - - - : プログラム経路の平行線

工具径補正タイプ	直線→直線	直線→円弧
タイプA		
タイプB		

工具径補正(G41, G42)とI, J, K指令(I, Jタイプベクトル指令)

工具径補正指令(G41, G42)と「I, J, K」を同一ブロックで指令することにより、補正方向を意図的に変えることができます。工具径補正指令(G41, G42)により作成されるベクトルについて、「I, Jタイプベクトル(G17平面)」を例に説明します。「I, Jタイプベクトル」は、プログラムされた経路の交点演算は行わず、「I, J」で指定された方向に垂直で補正量分の大きさのベクトルを補正ベクトルとします。I, Jベクトルは補正開始時でも補正モード中でも指令できます。補正量はI, J指定のあったブロックの補正番号(モーダルデータ)で決まります。

■補正開始時に「I, J」を指令した場合

—→: プログラム経路, ----→: 工具中心経路

・移動指令がある場合

I, Jタイプベクトル	プログラム	動作
「I, J」を指令した場合	(G40) : N100 G91 G41 X100 Y100 I100 J150 D01 N110 G04 P1000 N120 G01 F1000 N130 X150	
「I, J」を指令しない場合(タイプA)	(G40) : N100 G91 G41 X100 Y100 D01 N110 G04 P1000 N120 G01 F1000 N130 X150	

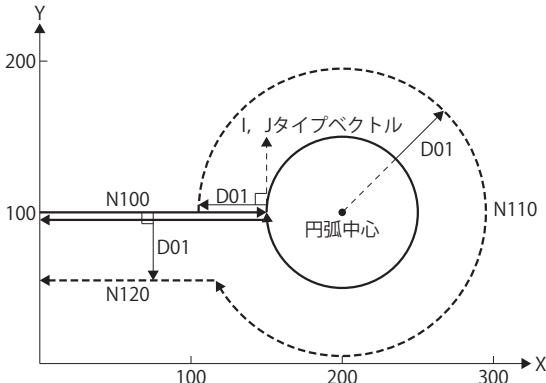
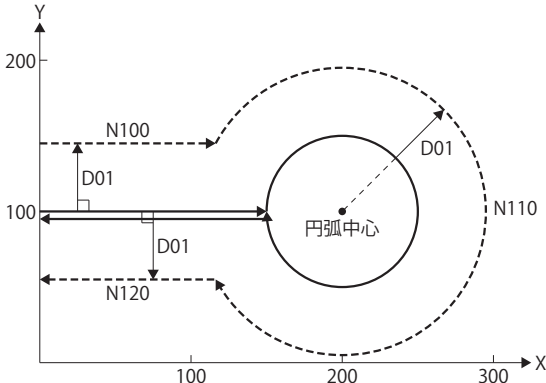
・移動指令がない場合

I, Jタイプベクトル	プログラム	動作
「I, J」を指令した場合	(G40) : N1 G41 I150 D01 N2 G91 X100 Y100 N3 X150	
「I, J」を指令しない場合(タイプA)	(G40) : N1 G41 D01 N2 G91 X100 Y100 N3 X150	

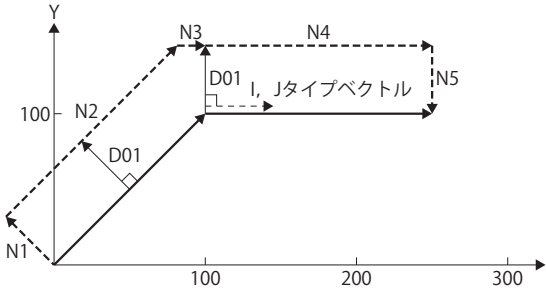
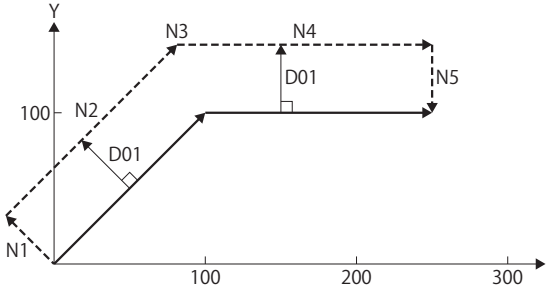
■補正モード中に「I, J」を指令した場合

—→: プログラム経路, ----→: 工具中心経路

・移動指令がある場合

I, Jタイプベクトル	プログラム	動作
「I, J」を指令した場合	(G17 G41 G91) : N100 G41 G00 X150 J50 D01 N110 G02 I50 N120 G00 X-150	
「I, J」を指令しない場合(タイプA)	(G17 G41 G91) : N100 G41 G00 X150 D01 N110 G02 I50 N120 G00 X-150	

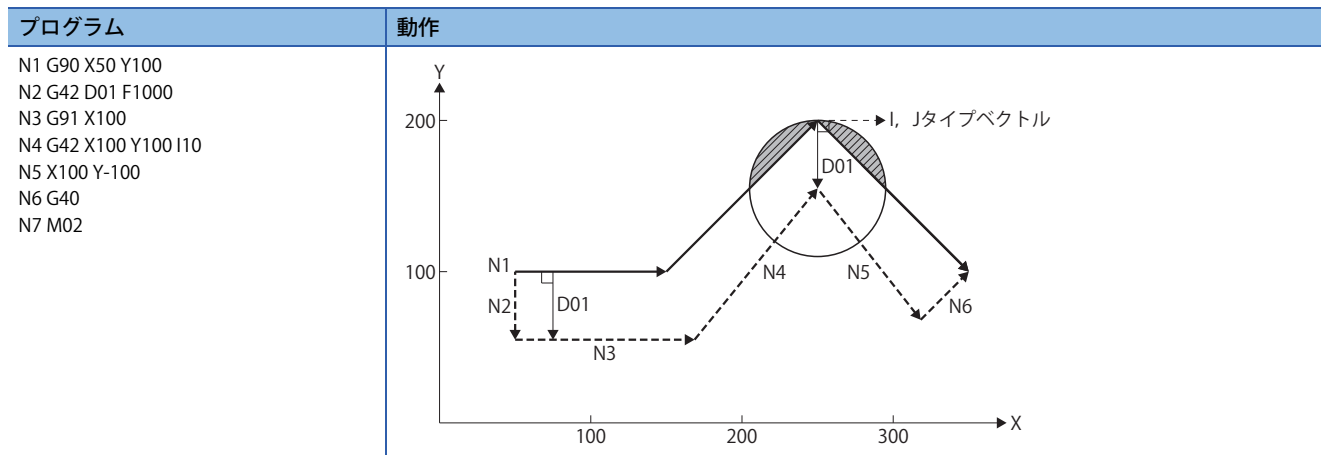
・移動指令がない場合

I, Jタイプベクトル	プログラム	動作
「I, J」を指令した場合	N1 G41 D01 G01 F1000 N2 G91 X100 Y100 N3 G41 I50 N4 X150 N5 G40	
「I, J」を指令しない場合(タイプA)	N1 G41 D01 G01 F1000 N2 G91 X100 Y100 N3 G41 N4 X150 N5 G40	

■注意事項

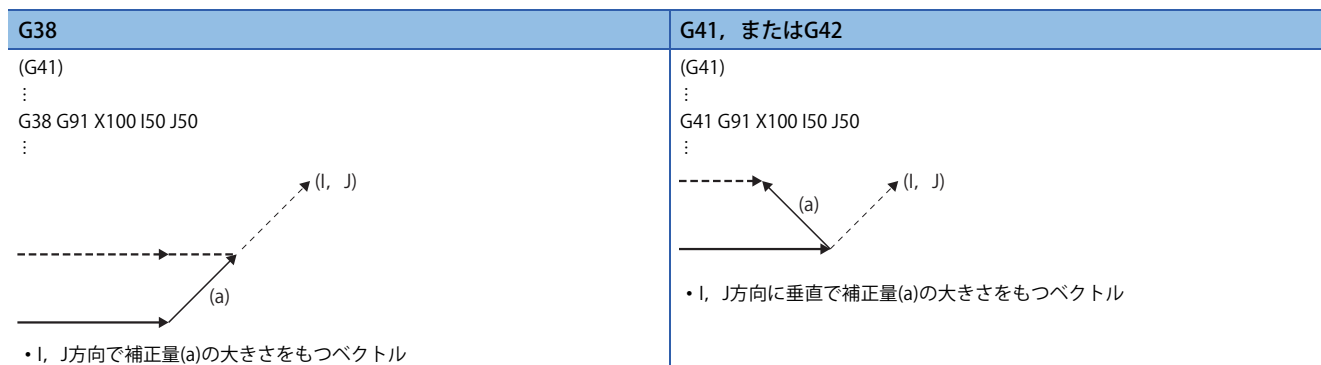
- I, Jタイプベクトルを指令するときは直線モード(G00, G01)で行ってください。補正開始時に円弧モードの場合は、軽度エラー(エラーコード:1FC3H(詳細コード:0315H))となります。補正モード中、円弧モードのときのI, J指定は、円弧の中心指定となります。
- I, Jタイプベクトルを指定した場合は、干渉があってもベクトルの消去(干渉の回避)は行いません。そのため、下図のように切り込み過ぎが発生することがあります。(下図の斜線部分)

——→: プログラム経路, ----→: 工具中心経路



- G38指令とG41/G42指令は補正するベクトルが異なります。

——→: プログラム経路, ----→: 工具中心経路



- G41/G42指令とI, J指令の組合せによる補正方法は、下表のようになります。

○: 指令あり, ×: 指令なし

G41/G42指令	I, J指令	補正方法
×	×	交点演算タイプベクトル
×	○	
○	×	
○	○	I, Jタイプベクトル ^{*1}

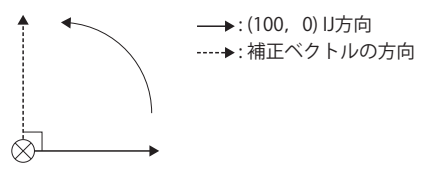
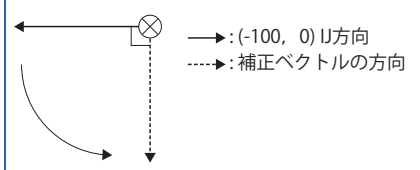
*1 I, Jタイプベクトルの場合、挿入ブロックはありません。

■補正ベクトルの方向

各工具径補正モードにおけるI, Jタイプベクトルの補正ベクトルの方向を以下に示します。

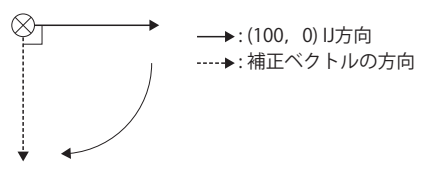
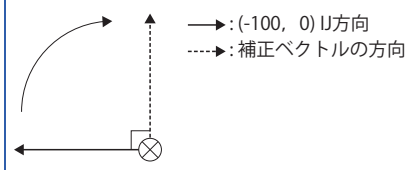
- 工具径補正(左方向)モードのとき(G41)

I, Jで指定された方向をZ軸(第3軸)の正方向から原点を見て左に90°回転した方向

I100のとき	I-100のとき
	

- 工具径補正(右方向)モードのとき(G42)

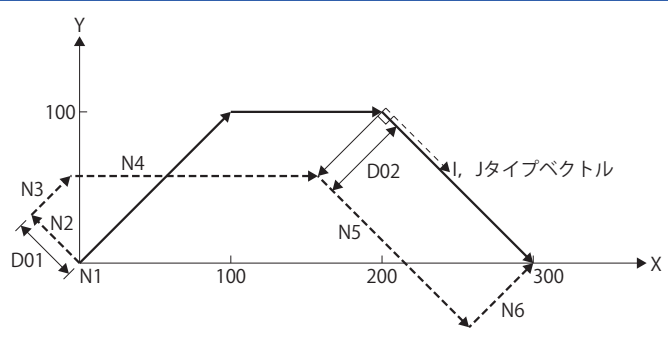
I, Jで指定された方向をZ軸(第3軸)の正方向から原点を見て右に90°回転した方向

I100のとき	I-100のとき
	

■補正のモーダル切換え

補正モードの途中で補正方向(G41/G42指令のモーダル)を切換えることができます。

——→: プログラム経路, ----→: 工具中心経路

プログラム	動作
N1 X0 Y0 N2 G41 D01 F1000 N3 G01 G91 X100. Y100. N4 G42 X100. I100. J -100. D2 N5 X100. Y-100. N6 G40 N7 M02 %	

干渉チェック

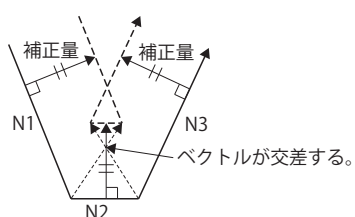
干渉チェックとは、プログラム経路に対して工具径が大きい場合に、工具径補正で補正された工具が素材に切り込むこと(干渉)があり、この動作を未然に防ぐための機能です。干渉チェックには、下記の3種類があり、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御加工パラメータ]⇒“工具径補正”⇒“干渉チェック”にて設定します。

干渉チェック	動作
干渉チェックアラーム	切り込みが発生するブロックの実行前に、軽度エラー (エラーコード: 1FC3H(詳細コード: 031EH))となり停止します。
干渉チェック無効	切り込みが発生してもそのまま切削します。(微小線分プログラムで使します。)
干渉チェック回避	切り込みが発生しないように経路を変更します。変更ができない場合は、軽度エラー (エラーコード: 1FC3H(詳細コード: 031EH))となり停止します。

■干渉チェックの条件

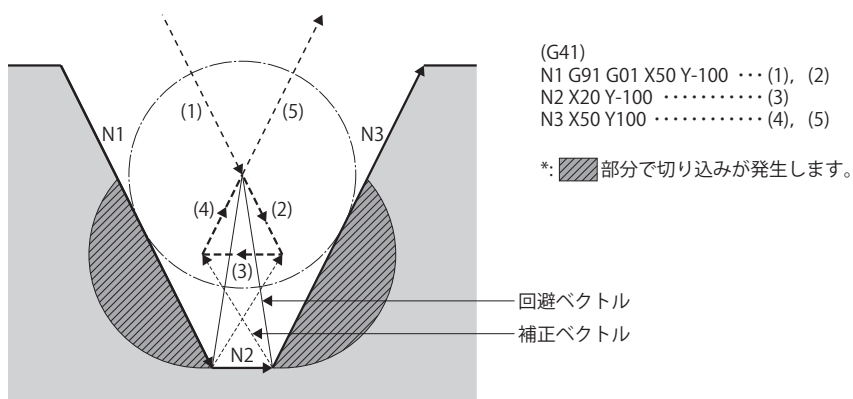
先読み5ブロック中に3ブロックの移動指令がある場合、各々の移動指令の設定に作った補正ベクトルが交差するときを干渉とみなします。

——▶: プログラム経路, ----▶: 工具中心経路,▶: 補正ベクトル



例

径の大きい工具で短い線分を含むプログラムを加工した場合



設定した干渉チェックにより下記の動作となります。

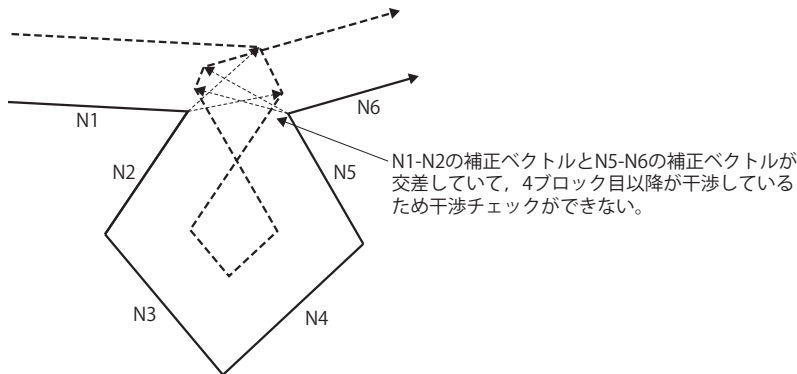
干渉チェック	動作
干渉チェックアラーム	N1実行前にアラームとなります。
干渉チェック無効	N1とN3の直線を切り込みながら通過します。工具中心経路は、(1)→(2)→(3)→(4)→(5)となります。
干渉チェック回避	N1とN3の交点演算を行い、回避ベクトルを作ります。工具中心経路は、(1)→(5)となります。

■干渉チェック不可の条件

以下の場合、干渉チェックが行えないため、切り込みが発生することがあります。

- ・移動指令ブロックを3ブロック先読みできない場合。(先読み5ブロック中に移動のないブロックが3ブロック以上ある場合)
- ・移動指令の4ブロック目以降が干渉する場合

——→: プログラム経路, - - - ->: 工具中心経路, ·····→: 補正ベクトル



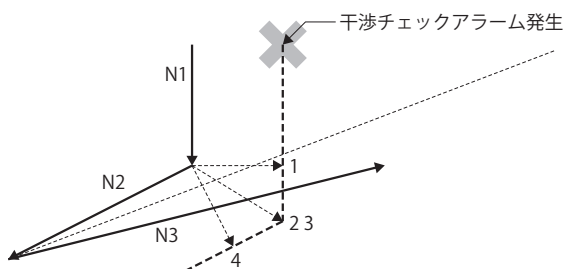
■干渉チェックアラーム

以下の場合、干渉チェックアラームとなります。

- ・自ブロックの終点でのベクトルがすべて消去された場合

N1ブロックの終点でのベクトル1~4がすべて消去された場合、N1実行前に軽度エラー (エラーコード: 1FC3H(詳細コード: 031EH))となります。

——→: プログラム経路, - - - ->: 工具中心経路, ·····→: 補正ベクトル



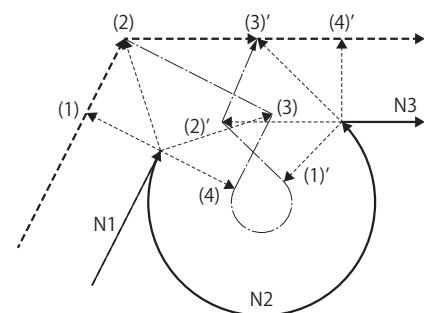
Point

干渉チェックアラームは、基本的に干渉するブロックの実行前に軽度エラー (エラーコード: 1FC3H(詳細コード: 031EH))となり、停止させる機能ですが、以下の場合、干渉チェックアラームとせず、回避動作を行います。

- ・円弧の始点と終点の補正ベクトルがすべて消去されない場合

(例)

径の大きい工具で半径の小さい円弧を含むプログラムを加工した場合



<動作>

- ・干渉チェック処理

1. ベクトル(1)(4)'チェック → 干渉しない
2. ベクトル(2)(3)'チェック → 干渉しない
3. ベクトル(3)(2)'チェック → 干渉する, ベクトル(3)(2)'を消去
4. ベクトル(4)(1)'を消去

1. 干渉チェックアラーム機能の場合

上記干渉チェック処理により, (1), (2), (3)', (4)'が有効ベクトルとして残ります。

工具中心経路は, (1)→(2)→(3)'→(4)'となります。

(- - - ->の経路)

この場合、干渉チェックアラームとなりません。

2. 干渉チェック回避機能の場合

上記干渉チェック処理により, (1), (2), (3)', (4)'が有効ベクトルとして残ります。

工具中心経路は, (1)→(2)→(3)'→(4)'となります。

(- - - ->の経路)

3. 干渉チェック無効機能の場合

工具中心経路は, (1)→(2)→(3)→(4)→(1)'→(2)'→(3)'→(4)'となります。

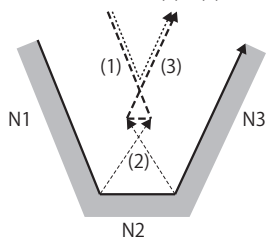
(- - - ->の経路)

■干渉チェック回避

干渉チェック回避の有効時，以下の動作となります。

<例1>

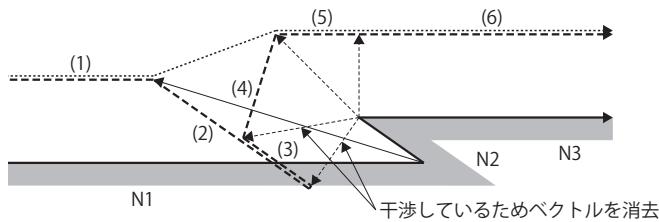
・工具中心経路は，(1)→(3)となります。



→: プログラム経路
 ---: 干渉チェックなし工具中心経路
: 干渉回避工具中心経路
 -----: 補正ベクトル

<例2>

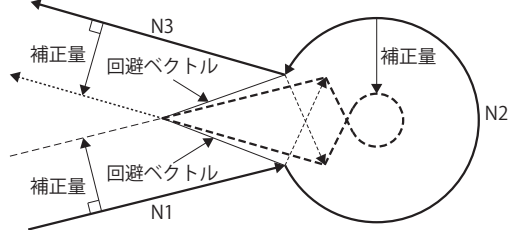
・工具中心経路は，(1)→(5)→(6)となります。



→: プログラム経路
 ---: 干渉チェックなし工具中心経路
: 干渉回避工具中心経路
 →: N1-N2の補正ベクトル
 -----: N2-N3の補正ベクトル

干渉回避の線ベクトルがすべて消去されたときは下図のように新しい回避ベクトルを作り，干渉を回避します。

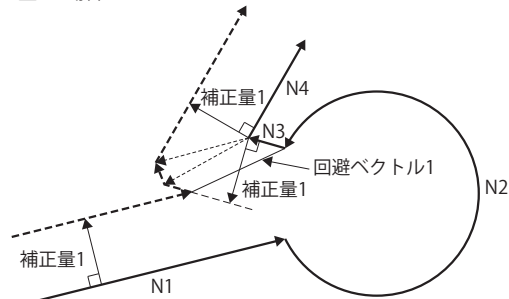
<例1>



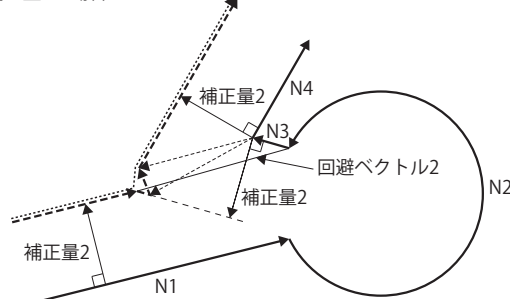
→: プログラム経路
 ---: 干渉チェックなし工具中心経路
: 干渉回避工具中心経路
 -----: 補正ベクトル

<例2>

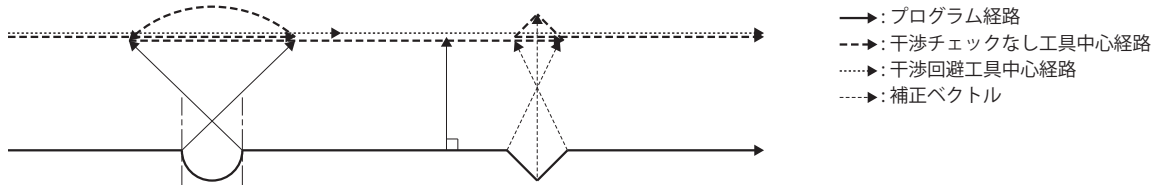
・補正量1の場合



・補正量2の場合



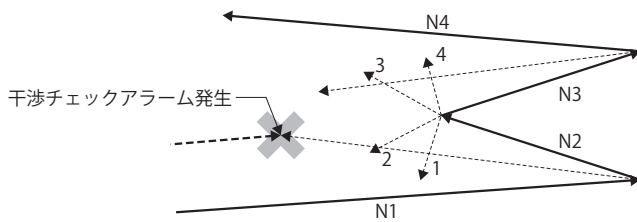
下図の場合，溝は削り残します。



以下の場合、経路の変更が行えず、干渉チェックアラームとなります。

- 自ブロックの終点ベクトルがすべて消去されても、次ブロックの終点ベクトルに有効なものがある場合
N2の干渉チェックを行うとN2の終点ベクトルはすべて消去されますがN3の終点ベクトルは有効とみなされます。このとき、N1の終点で軽度エラー (エラーコード: 1FC3H(詳細コード: 031EH)) となり停止します。

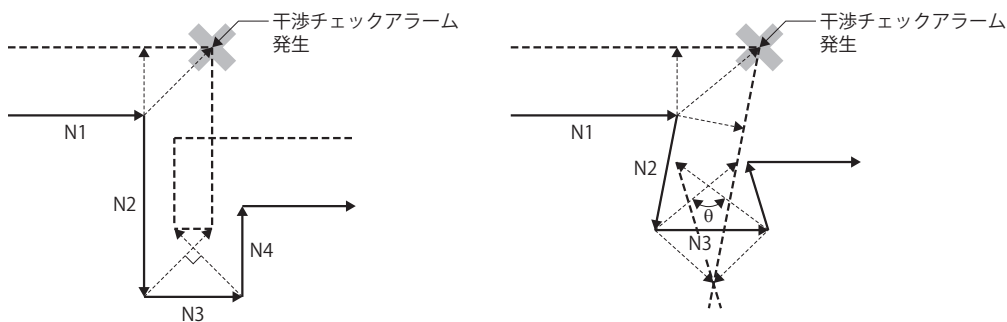
——▶: プログラム経路, - - - -▶: 工具中心経路, ·····▶: 補正ベクトル



- 回避ベクトルを作成できない場合

回避ベクトルの作成条件が満たされている場合でも、回避ベクトルが作成できなかったり、回避ベクトルがN3に干渉したりすることがあります。

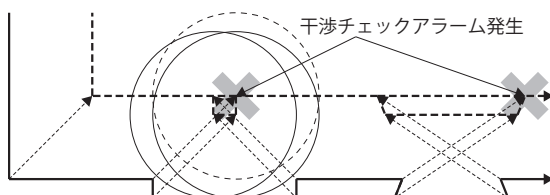
——▶: プログラム経路, - - - -▶: 工具中心経路, ·····▶: 補正ベクトル



- プログラム進行方向と、補正後の進行方向が逆になる場合

工具直径より狭く平行、または底広がり溝をプログラムした場合、実際には干渉しなくても干渉するとみなされることがあります。

——▶: プログラム経路, - - - -▶: 工具中心経路, ·····▶: 補正ベクトル



- 一時的に補正ベクトルをなくす指令の直前ブロックの終点ベクトルが干渉する場合

一時的に補正ベクトルをなくす指令の直前ブロックでも、補正ベクトルがなくなる場合と同様の終点ベクトルにて干渉チェックを行います。そのため、実際には干渉しなくても干渉したとみなされることがあり、干渉するとみなしたときは、軽度エラー (エラーコード: 1FC3H(詳細コード: 031EH)) となります。

径補正直径指定

[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御加工パラメータ]⇒“工具径補正”⇒“径補正直径指定”にて、工具径補正を工具直径で指定します。径補正直径指定を設定すると、指令工具番号に設定されている補正量を直径補正量と認識し、補正時に半径補正量に変換されます。

工具径補正の指令，動作に関する注意事項

■工具径補正中の補正方向の変更

- 補正方向は、工具径補正指令(G41, G42)と補正量の符号により決定します。補正モード中に工具径補正キャンセル指令を行わないで工具径補正指令を変更すると、補正方向を変更できます。

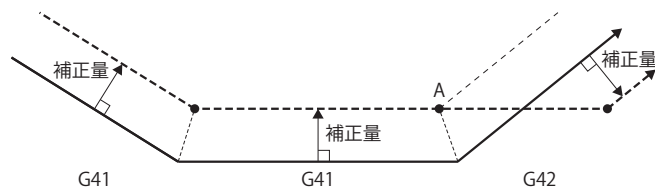
工具径補正指令	補正量	
	符号(+)	符号(-)
G41	左側補正	右側補正
G42	右側補正	左側補正

- 補正量の変更は、通常、工具径補正キャンセルモードで別の工具が選択されたときに行いますが、補正モード中に変更された場合、ブロックの終点におけるベクトルは、そのブロックで指定されている補正量を用いて計算します。

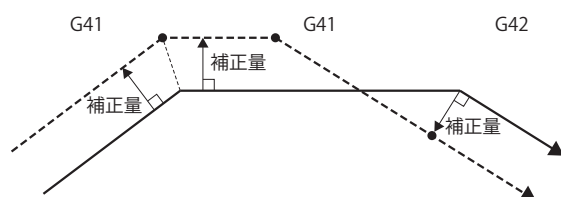
→: プログラム経路, ---→: 工具中心経路

- 直線→直線

- 補正方向変更時に交点(A)がある場合

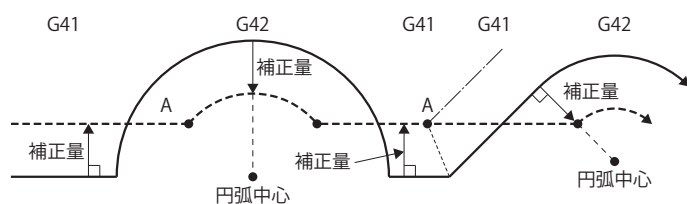


- 補正方向変更時に交点がない場合

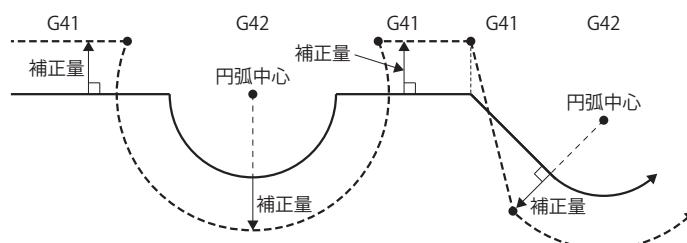


- 直線→円弧

- 補正方向変更時に交点(A)がある場合

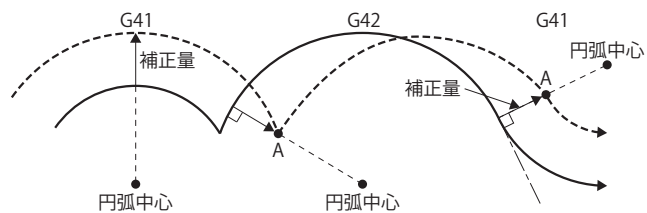


- 補正方向変更時に交点がない場合

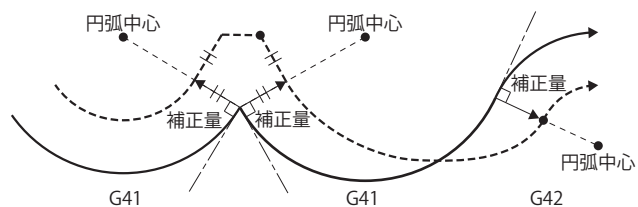


• 円弧→円弧

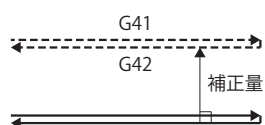
• 補正方向変更時に交点(A)がある場合



• 補正方向変更時に交点がない場合

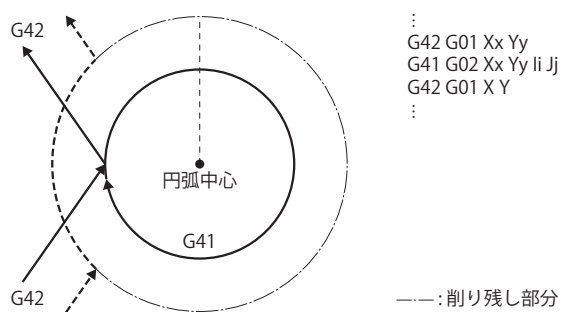


• 直線で往復する場合



• G41, G42の切換えにより、円弧が360°以上になる場合

下図のように補正され、削り残しが発生します。



■工具径補正中の補正番号の変更

補正モード中は、補正番号の変更を行わないでください。補正モード中に変更した場合は、以下の動作となります。

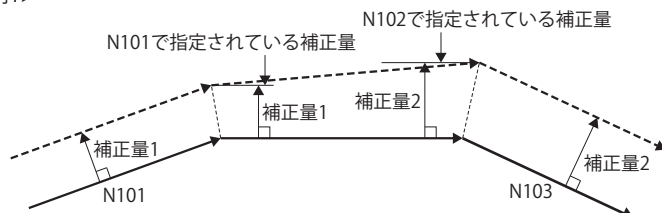
プログラム

```
G41 G01 ..... D01
(α=0, 1, 2, 3)
N101 G0α Xx1 Yy1
N102 G0α Xx2 Yy2 D02
```

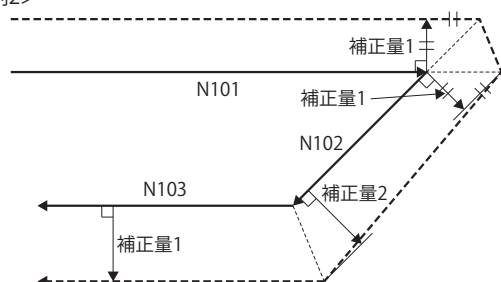
——→: プログラム経路, - - - ->: 工具中心経路

・直線→直線の場合

<例1>

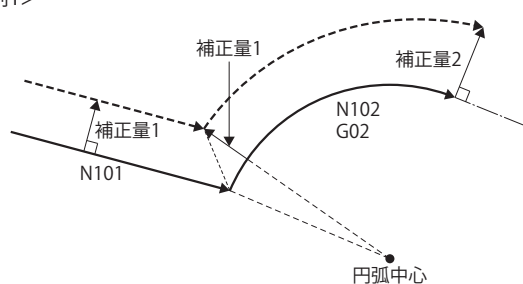


<例2>

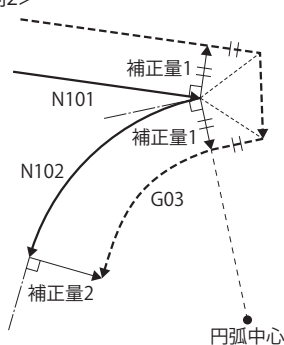


・直線→円弧の場合

<例1>

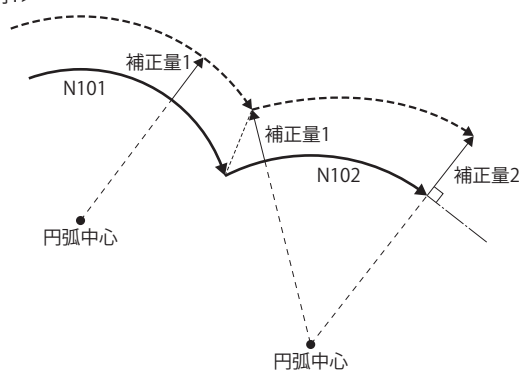


<例2>

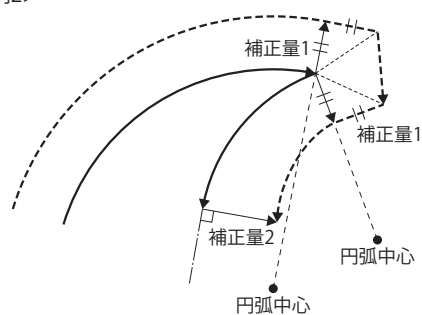


・円弧→円弧の場合

<例1>



<例2>



■一時的に補正ベクトルをなくす指令

補正モード中に基本機械座標系選択(G53)を指令すると、一時的に補正ベクトルがなくなり、その後、自動的に補正モードに戻ります。この場合、補正キャンセルの動作にはならず、交点ベクトルから直後にベクトルのない点(プログラムの指令点)に移動します。補正モードに戻るときも直接交点へ移動します。

■移動のないブロック

—→: プログラム経路, ----→: 工具中心経路

- 補正開始時に指令された場合

移動のないブロックが4ブロック以上連続するとき、および先読み禁止M指令のときは、補正ベクトルは作成しません。

プログラム	動作
N1 G91 X50. Y130. N2 G41 D10 N3 G04 X1000 N4 F100 N5 M500 N6 M3 N7 X50. Y-80. N8 X100. Y-30. *: N2~N6は移動なしブロック	

- 補正モード中に指令された場合(移動のないブロックが4ブロック未満)

移動のないブロックが補正モード中に4ブロック以上連続していないとき、または先読み禁止M指令がないときは、通常の交点ベクトルを作成します。

プログラム	動作
N6 G91 X50. Y100. N7 G04 P1000 N8 X200. *: N7は移動なしブロック	

- 補正モード中に指令された場合(移動のないブロックが4ブロック以上)

移動のないブロックが補正モード中に4ブロック以上連続するとき、および先読み禁止M指令があるときは、前ブロックの終点に垂直に補正ベクトルを作成します。このとき、切り込みが発生することがあります。

プログラム	動作
N6 G91 X50. Y100. N7 G04 X1000 N8 F100 N9 M500 N10 M4N11 X200. *: N7~N10は移動なしブロック	

- 工具径補正キャンセルと共に指令された場合

移動のないブロックがG40と共に指令されたときは、補正ベクトルのみキャンセルします。

プログラム	動作
N6 G91 X50. Y100. N7 G40 M5 N8 X150. Y30.	

■工具径補正キャンセル(G40)に「I, J, K」が指令されたとき

工具径補正キャンセル(G40)と共に「I, J, K」が指令された場合、以下のような動作となります。

——→: プログラム経路, ----→: 工具中心経路, - - - ->: 仮想工具中心経路

- G40ブロックの手前4ブロック中の最後の移動指令ブロックが工具径補正(G41, G42)モードのとき、最後の移動指令の終点からベクトル(I, J, K)の方向に移動指令したと扱い、それに対する仮想工具中心経路との交点まで補正後キャンセルを行います。このとき、補正方向は変わりません。

プログラム	動作
<例1> (G41) : N1 G91 G01 X200. N2 G40 X-100. Y100. I100. J50.	<p>この場合、補正方向にかかわらず例2のように指令ベクトルを間違えた場合でも、必ず交点を求めますので注意してください。</p> <p><例2>I, Jの符号を間違えた場合</p> <p>ただし、「I, J, K」で指定したベクトル方向とG40指令前のベクトル方向が平行に近い場合は、G40の前ブロックに垂直なベクトルを作成します。</p> <p><例3></p>

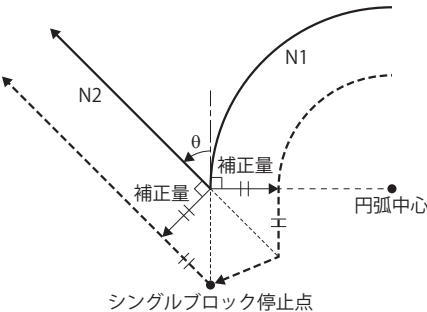
- 円弧指令後のG40で「I, J, K」の内容により円弧が360°以上になると、削り残しが発生しますので注意してください。

プログラム	動作
(G42, G91) : N1 G01 X200. N2 G03 J50. N3 G40 G01 X100. Y-100. I-100. J100.	<p>———: 削り残し部分</p>

■コーナの移動

移動指令ブロックのつなぎ目で補正ベクトルが複数個作成されるときは、そのベクトル間を直線で移動します。
ベクトルが一致していない場合は、コーナを回るための移動を行います。
シングルブロック運転の動作は、前ブロックとコーナの移動を1つのブロックとして実行し、次の起動で残りの移動と次ブロックを1つのブロックとして実行します。

——→: プログラム経路, ----→: 工具中心経路



■工具径補正開始とZ軸切り込み動作

切削の開始時に、ワークから離れた位置であらかじめ径補正をかけておき(通常XY平面)、その後、Z軸で切り込むという方法を一般的に行いますが、このとき、Z軸の動作を早送りとワークに接近してからの切削送りとの2段階に分けて行う場合は、以下のことに注意してプログラムしてください。

下記Gコードプログラム(プログラム1)の場合、N1の補正開始時に、N6ブロックまで読み込むことができ、N1とN6の関係を判断して、下図のように正しく補正が行われます。

——→: プログラム経路, ----→: 工具中心経路

プログラム1	動作
N1 G91 G00 G41 X500. Y500. D01 N2 M1000 N3 M3 N4 G01 Z-150. F1 N6 Y200. F2	

Gコードプログラム(プログラム1)のN4ブロックを2つに分けた場合、N2～N5とXY平面内に指令のないブロックが4ブロックあるため、N1の補正開始時に、N6のブロックまで読み込むことができません。そのため、N1のブロックの情報だけを基にして補正を行うことになり、補正開始時に補正ベクトルが作成されず、切り込み過ぎとなります。

——▶: プログラム経路, ----▶: 工具中心経路

プログラム2	動作
N1 G91 G00 G41 X500. Y500. D01 N2 M1000 N3 M3 N4 Z-100. N5 G01 Z-50. F1 N6 Y200. F2	

Gコードプログラム(プログラム2)の場合、内部の計算を考慮して、下記Gコードプログラム(プログラム3)のようにZ軸が下降した後に進行する方向と同じ方向の指令を、あらかじめZ軸の切り込み直前に指令しておけば、切り込み過ぎを防ぐことができます。N2で、N6の進行方向と同じ方向を指令しているため、正しく補正します。

——▶: プログラム経路, ----▶: 工具中心経路

プログラム3	動作
N1 G91 G00 G41 X500. Y400. D01 N2 Y100. M1000 N3 M3 N4 Z-100. N5 G01 Z-50. F1 N6 Y200. F2	

6.5 操作支援機能

自動運転起動(サイクルスタート)

自動運転を行うGコードプログラムのプログラム番号、シーケンス番号、ブロック番号を指定し、“[Rq.3377]自動運転起動(サイクルスタート)(D54226.1+2s)”の立上りにより、指定したブロックの頭出しを行い、Gコードプログラムを起動できます。また、自動運転休止中やブロック停止中のGコードプログラムに対し、再起動できます。自動運転起動は、系統ごとに行います。

Gコードプログラムの優先度

指定したプログラム番号、シーケンス番号、ブロック番号の優先度を下記に示します。

条件			動作
[Cd.3320]プログラム番号指定レジスタ (D54278+16s)	[Cd.3321]シーケンス番号指定レジスタ (D54280+16s, D54281+16s)	[Cd.3322]ブロック番号指定レジスタ (D54282+16s, D54283+16s)	
指定なし(=0)	—	—	軽度エラー (エラーコード: 1FC3H(詳細コード: 031FH))となります。
指定あり	指定なし(=0)	指定なし(=0)	Gコードプログラムの先頭から起動します。
指定あり	指定なし(=0)	指定あり	Gコードプログラムの先頭からシーケンス番号が指令されるまでのブロック内での、指定したブロック番号の位置から起動します。
指定あり	指定あり	指定なし(=0)	Gコードプログラムの指定したシーケンス番号の位置から起動します。
指定あり	指定あり	指定あり	指定したGコードプログラム内のシーケンス番号の位置から、次のシーケンス番号が指令されるまでのブロック内での、指定したブロック番号の位置から起動します。

Gコードプログラムの起動条件

“[Rq.3377]自動運転起動(サイクルスタート)(D54226.1+2s)”にて、プログラム番号、シーケンス番号、ブロック番号で指定したGコードプログラムが動作する条件を下記に示します。

条件			動作
[St.3212]自動運転中 (D54441.0+4s)	[St.3213]自動運転起動中 (D54441.1+4s)	[St.3214]自動運転休止中 (D54441.2+4s)	
OFF	OFF*1	OFF*1	プログラム番号、シーケンス番号、ブロック番号で指定したGコードプログラムが起動します。
ON	OFF	OFF	自動運転停止状態のGコードプログラムが再起動します。
	OFF	ON	自動運転休止状態のGコードプログラムが再起動します。
	ON	OFF	自動運転起動中のため、自動運転起動信号は無視されます。
	ON	ON	状態として存在しません。

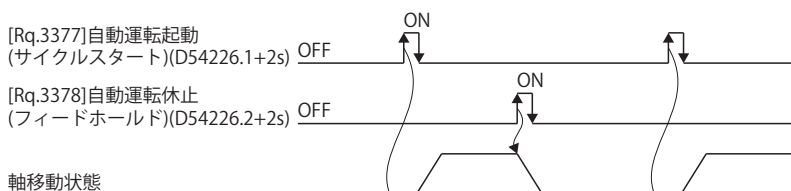
*1 “[St.3212]自動運転中(D54441.0+4s)”がOFFの場合、ONにはなりません。

注意事項

- シーケンス番号、ブロック番号を指定して起動した場合、起動ブロックをサーチする時間があるため、起動までに時間がかかることがあります。
- 1つのプログラム内に同じシーケンス番号を複数指定した場合、先頭から最初に検索されたシーケンス番号の位置から起動します。
- 指定したプログラム番号のGコードプログラムが存在しないと、軽度エラー (エラーコード: 1FC3H(詳細コード: 031FH))となります。
- Gコードプログラム内に対応するシーケンス番号がないと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0320H))となります。
- Gコードプログラム内に対応するブロック番号がないと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0321H))となります。
- 極座標補間モード、プログラム座標回転モード、および法線制御などを行うGコードプログラムの途中から、指定したブロックの頭出しを行いGコードプログラムを起動すると、意図した動作とならないことがありますので指定しないでください。
- 検索対象のシーケンス番号が複数指定されているGコードプログラムに対して、シーケンス番号、ブロック番号を指定して起動した場合、先頭から最初に検索されたシーケンス番号から、次の異なるシーケンス番号が指定されるまでの間、起動ブロックのサーチを行います。次の異なるシーケンス番号が指定されるまでの間に対応するブロック番号がないと、軽度エラー (エラーコード: 1FC3H(詳細コード: 0321H))となります。

自動運転休止(フィードホールド)

自動運転中、“[Rq.3378]自動運転休止(フィードホールド)(D54226.2+2s)”をONすることにより、軸を途中で減速停止させます。再起動は、“[Rq.3377]自動運転起動(サイクルスタート)(D54226.1+2s)”により行います。



Point

自動運転中にフィードホールドとなったとき、送りは直ちに減速停止しますが、同一ブロック中のM指令はそのまま継続して実行します。

リセット

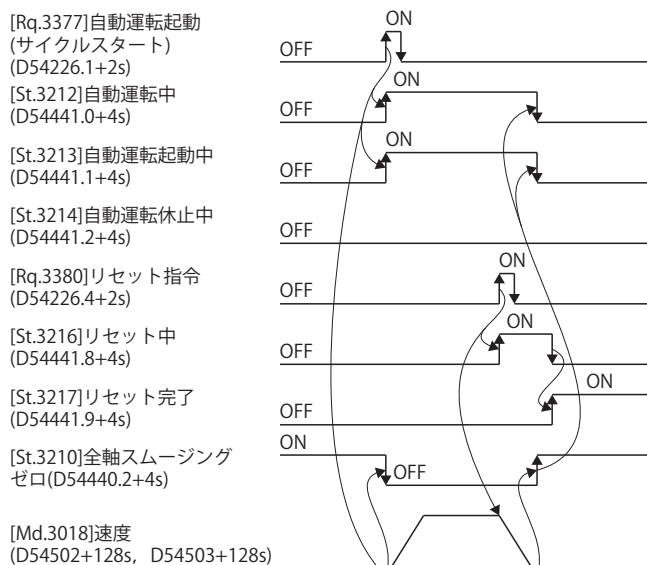
“[Rq.3380]リセット指令(D54226.4+2s)”の立上りによりGコード制御の運転状態をリセットします。リセットを行ったときの各機能の動作を下記に示します。

機能	動作	関連デバイス	
Gコードモダル	初期化	<ul style="list-style-type: none"> • [Md.3028]グループ01モダル状態(D54520+128s) • [Md.3029]グループ02モダル状態(D54521+128s) • [Md.3030]グループ03モダル状態(D54522+128s) • [Md.3034]グループ07モダル状態(D54526+128s) • [Md.3038]グループ08モダル状態(D54532+128s) • [Md.3046]グループ12モダル状態(D54542+128s) • [Md.3047]グループ13モダル状態(D54543+128s) • [Md.3049]グループ15モダル状態(D54545+128s) • [Md.3050]グループ16モダル状態(D54546+128s) • [Md.3055]グループ21モダル状態(D54551+128s) 	各グループの初期状態に合わせて設定
工具補正データ	キャンセル	<ul style="list-style-type: none"> • [Md.3035]工具径補正番号(D54527+128s) • [Md.3036]工具径補正量(D54528+128s, D54529+128s) • [Md.3039]工具長補正番号(D54533+128s) • [Md.3040]工具長補正量(D54534+128s, D54535+128s) • [Md.3042]工具長補正軸No.(D54538+128s) 	0クリア
プログラム	停止	<ul style="list-style-type: none"> • [Md.3022]実行中プログラム番号(メイン)(D54508+128s) • [Md.3023]実行中シーケンス番号(メイン)(D54510+128s, D54511+128s) • [Md.3024]実行中ブロック番号(メイン)(D54512+128s, D54513+128s) • [Md.3070]実行中プログラムコメント(D54588+128s～D54595+128s) 	保持
		<ul style="list-style-type: none"> • [Md.3025]実行中プログラム番号(サブ/マクロ)(D54514+128s) • [Md.3026]実行中シーケンス番号(サブ/マクロ)(D54516+128s, D54517+128s) • [Md.3027]実行中ブロック番号(サブ/マクロ)(D54518+128s, D54519+128s) 	0クリア
エラー	リセット*1	• [St.3209]Gコード制御エラー検出(D54440.1+4s)	OFF
		<ul style="list-style-type: none"> • [Md.3019]Gコード制御エラーコード(D54504+128s) • [Md.3020]Gコード制御エラー詳細コード1(D54505+128s) • [Md.3021]Gコード制御エラー詳細コード2(D54506+128s) 	0クリア
補助機能	保持	<ul style="list-style-type: none"> • [Md.3058]Mコードデータ1(D54554+128s, D54555+128s) • [Md.3059]Mコードデータ2(D54556+128s, D54557+128s) • [Md.3060]Mコードデータ3(D54558+128s, D54559+128s) • [Md.3061]Mコードデータ4(D54560+128s, D54561+128s) 	保持
	OFF	<ul style="list-style-type: none"> • [St.3218]M単独出力 M00(D54442.0+4s) • [St.3219]M単独出力 M01(D54442.1+4s) • [St.3220]M単独出力 M02(D54442.2+4s) • [St.3221]M単独出力 M30(D54442.3+4s) • [St.3222]補助機能ストローブ1(D54442.4+4s) • [St.3223]補助機能ストローブ2(D54443.5+4s) • [St.3224]補助機能ストローブ3(D54444.6+4s) • [St.3225]補助機能ストローブ4(D54445.7+4s) 	OFF
マクロシングル	リセット	• [St.3234]マクロシングル有効中(D54441.F+4s)	OFF
制御軸移動	減速停止	—	—
出力信号	ON	• [St.3216]リセット中(D54441.8+4s)	ON
ローカル変数深さ	リセット	• [Md.3074]ローカル変数深さ(D54582+128s)	0クリア

*1 エラー発生時に“[Rq.3380]リセット指令(D54226.4+2s)”のOFF→ONにより、Gコード制御の制御状態がリセットされ、関連デバイスがクリアされますが、LED表示やGX Works3、MT Developer2のエラー表示、特殊リレー、特殊レジスタのエラー状態は解除されません。エラーの解除方法については下記を参照してください。

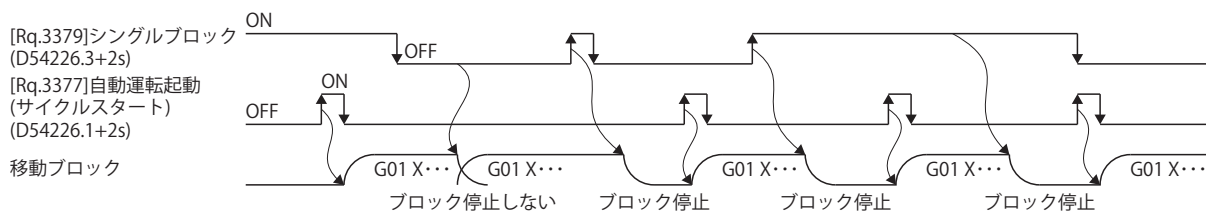
📖 MELSEC iQ-R モーションコントローラプログラミングマニュアル(共通編)

- リセット中は, “[St.3216]リセット中(D54441.8+4s)”がONし, 制御中の軸は, 減速停止を開始します。リセットが完了すると “[St.3216]リセット中(D54441.8+4s)”がOFFし, “[St.3217]リセット完了(D54441.9+4s)”がONします。自動運転の状態はリセット状態となり, “[St.3212]自動運転中(D54441.0+4s)”がOFFします。軸移動中にリセットを実行した場合の動作を下記に示します。



シングルブロック

“[Rq.3379]シングルブロック(D54226.3+2s)”をONすることにより, 自動運転における指令を1ブロックずつ実行(ブロック停止)することができます。また, 連続運転中に “[Rq.3379]シングルブロック(D54226.3+2s)”をONすると, 運転中のブロックを実行後に停止します。



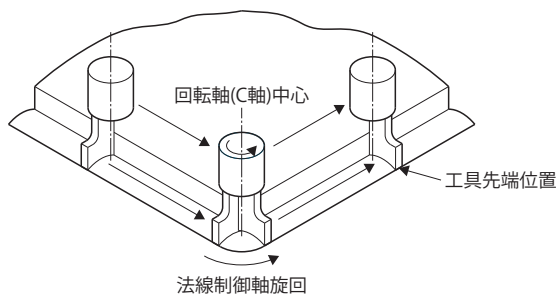
Point

- 自動運転実行中の場合, “[Rq.3379]シングルブロック(D54226.3+2s)”をONすると, 実行中のブロックが終了後, 停止します。次のブロックを実行するためには, 再度 “[Rq.3377]自動運転起動(サイクルスタート)(D54226.1+2s)”をOFF→ONしてください。
- “[Rq.3379]シングルブロック(D54226.3+2s)”がONの状態では自動運転起動を行うと, 1ブロック動作した後に停止するため, Gコードプログラムを1ブロックずつ実行することができます。

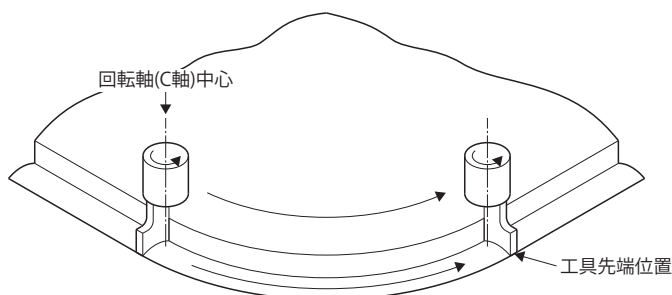
6.6 法線制御機能

プログラム運転で平面選択中の軸移動に対して、工具が常に法線方向に向くように回転軸(C軸)を旋回制御する機能です。

- ブロックの継ぎ目では、次ブロックの始点で工具が法線方向に向くように回転軸(C軸)を旋回する制御をします。



- 円弧補間中は、円弧補間の動作に同期して回転軸を旋回する制御をします。



法線制御の角度

法線制御の角度は、工具が水平軸(+方向)の方向を向いているとき、0[degree]とします。反時計周りの旋回を+(プラス)、時計周りの旋回を-(マイナス)とします。

平面選択	内容	動作
G17平面(I軸-J軸)	工具がI軸の+方向を向いているときを0[degree]とします。	
G18平面(K軸-I軸)	工具がK軸の+方向を向いているときを0[degree]とします。	
G19平面(J軸-K軸)	工具がJ軸の+方向を向いているときを0[degree]とします。	

移動指令に対する法線制御の旋回動作

■法線制御開始時

法線制御指令ブロックの始点で進行方向と直角になるように法線制御軸が旋回した後、平面選択中の軸の移動が行われます。スタートアップ時の法線制御軸旋回方向は、法線制御タイプIと法線制御タイプII共に、180[degree]以下になるような方向(近回り方向)に旋回します。ただし、補間しないG00指令のブロックでは法線制御しません。次のブロック以降に法線制御を開始します。

—→: プログラム経路, —: 工具先端経路

ブロック	プログラム	動作
単独ブロック	⋮ N1 G01 Xx1 Yy1 Ff1 N2 G41.1 N3 Xx2 Yy2 ⋮ *: N2は移動なし	
同一ブロック	⋮ N1 G01 Xx1 Yy1 Ff1 N2 G41.1 Xx2 Yy2 ⋮	

■法線制御モード中

• ブロック中の動作

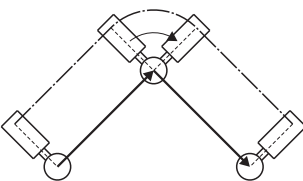
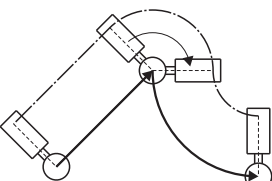
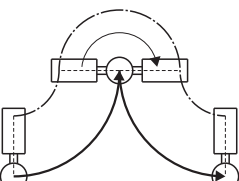
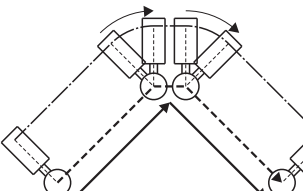
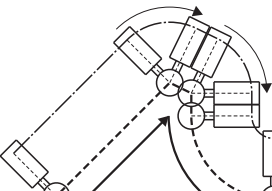
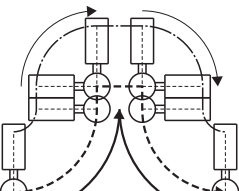
直線指令時、法線制御の旋回軸の角度は一定に保たれ、法線制御軸は旋回しません。円弧指令時、法線制御の旋回軸は円弧補間の動作に同期して旋回します。ただし、円弧補間許容誤差以内の誤差がある円弧補間に同期する場合、法線制御中の旋回角は法線方向から補間誤差分ずれることがあります。

—→: プログラム経路, —: 工具先端経路

プログラム	動作
⋮ G41.1 N1 G02 Xx1 Yy1 Ii1 Jj1 ⋮	

・ブロックの継ぎ目

——▶: プログラム経路, - - - -▶: 工具径補正経路, ———▶: 工具先端経路

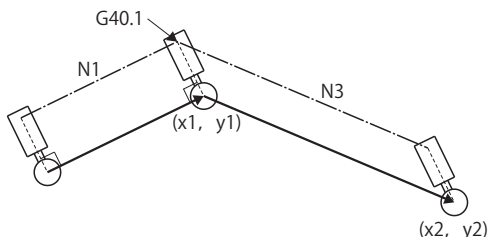
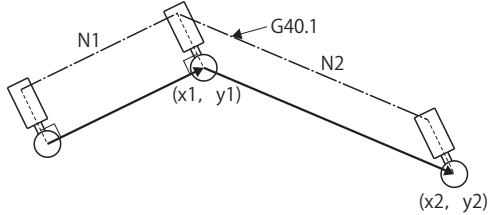
工具径補正	内容	動作		
		直線→直線	直線→円弧	円弧→円弧
なし	次ブロックの平面選択中の移動方向に直角になるように、法線制御軸を旋回させた後そのブロックを移動させます。*1			
あり	工具径補正をかけると、工具径補正のかかった経路に沿って法線制御を行います。*1			

*1 継ぎ目直後のブロックが補間しないG00指令の場合、法線制御軸は旋回しません。

■法線制御終了時

G40.1指令以降のブロックでは、法線制御軸の旋回動作をしません。法線制御軸(C軸)は指令による軸移動を行います。G40.1指令のブロックで減速停止完了後、次ブロックの座標指令は移動を開始します。

——▶: プログラム経路, ———▶: 工具先端経路

ブロック	プログラム	動作
単独ブロック	⋮ N1 G01 Xx1 Yy1 Ff1 N2 G40.1 N3 Xx2 Yy2 ⋮ *: N2は移動なし	
同一ブロック	⋮ N1 G01 Xx1 Yy1 Ff1 N2 G40.1 Xx2 Yy2 ⋮	

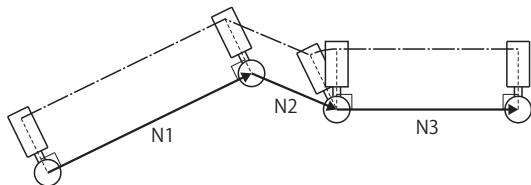
法線制御一時キャンセル

法線制御中、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“法線制御”⇒“旋回最小移動量”で設定した移動量より、小さい移動量のブロックとその前ブロックとの継ぎ目では、法線制御軸の旋回動作は行いません。

■直線ブロックの場合

N2ブロックの移動量がGコード制御系統パラメータの「旋回最小移動量」より小さい場合、N1ブロックとN2ブロックとの継ぎ目で法線制御軸の旋回は行いません。N1ブロックでの方向のままとなります。

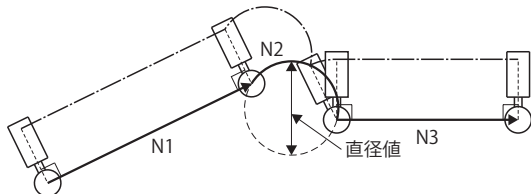
- N2ブロックの移動量 < 旋回最小移動量



■円弧ブロックの場合

N2ブロックの直径値がGコード制御系統パラメータの「旋回最小移動量」より小さい場合、N1ブロックとN2ブロックとの継ぎ目で法線制御軸の旋回は行いません。N1ブロックでの方向のままとなります。また、N2ブロックでの円弧補間中も法線制御軸は円弧補間の動作に同期して旋回しません。

- N2ブロックの直径値 < 旋回最小移動量

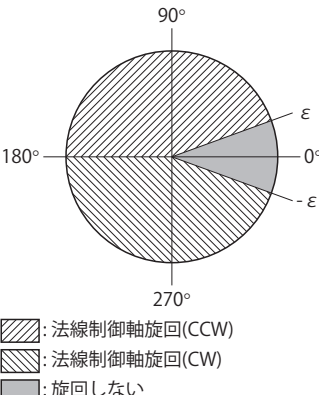



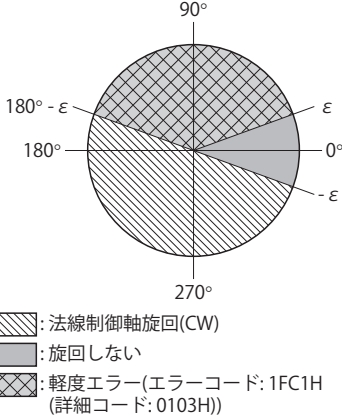



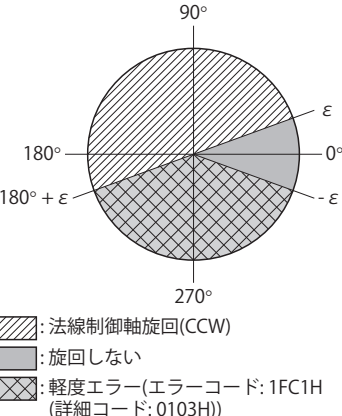





Point

工具径補正中、2線分の交点計算により演算端数が発生するため、Gコード制御系統パラメータの「旋回最小移動量」と線分の長さが等しいとき、旋回する場合と旋回しない場合があります。

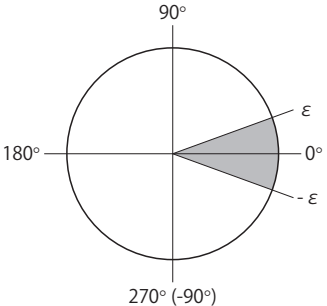
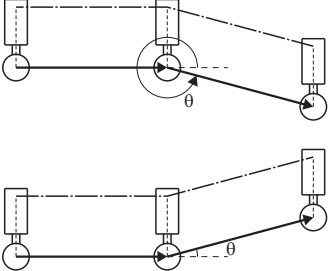
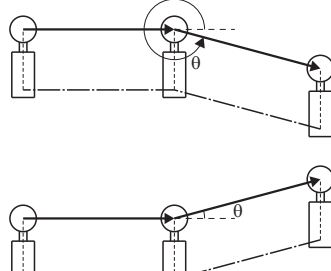
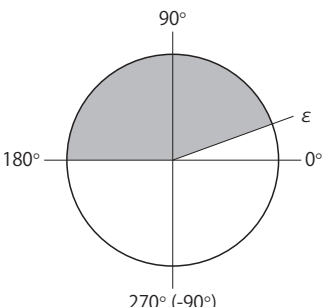
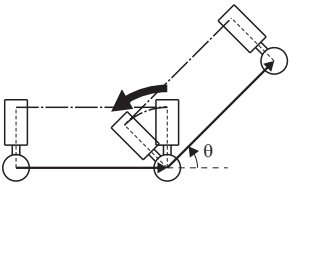
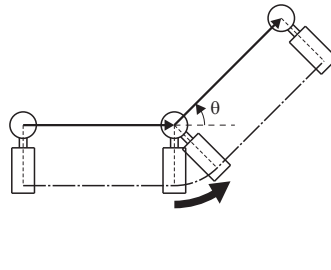
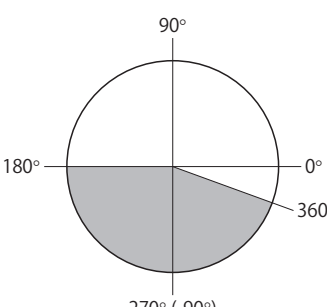
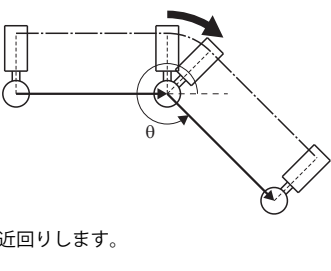
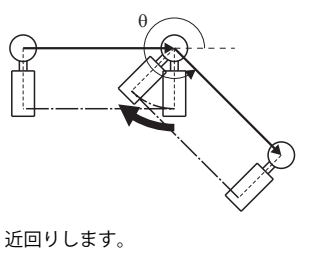
ブロック継ぎ目における法線制御軸旋回方向

ブロック継ぎ目における法線制御軸旋回方向は、法線制御タイプⅠと法線制御タイプⅡにより異なります。また、旋回角度は[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御系統パラメータ]⇒“法線制御”⇒“旋回最小角”で設定した角度 ε により制限を受けます。

項目	法線制御タイプⅠ	法線制御タイプⅡ
ブロック継ぎ目の法線制御軸旋回方向	180°以下になる方向(近回り方向)	<ul style="list-style-type: none"> • G41.1: -方向(CW) • G42.1: +方向(CCW)
ブロック継ぎ目の法線制御軸旋回角度	<p>「$\theta < \varepsilon$」の場合は旋回しません。 θ: 旋回角度 ε: 旋回最小角</p> <p>旋回角度=180°の場合、指令モードに関係なく旋回方向は不定となります。</p> <p><G41.1/G42.1法線制御軸が0°にある場合></p>  <p> : 法線制御軸旋回(CCW) : 法線制御軸旋回(CW) : 旋回しない </p>	<p>「$\theta < \varepsilon$」の場合は旋回しません。 θ: 旋回角度 ε: 旋回最小角</p> <p>下記の場合は、軽度エラー (エラーコード: 1FC1H(詳細コード: 0103H))となります。</p> <ul style="list-style-type: none"> • G41.1の場合 $\varepsilon \leq \theta < 180^\circ - \varepsilon$ • G42.1の場合 $180^\circ + \varepsilon < \theta \leq 360^\circ - \varepsilon$ <p><G41.1法線制御軸が0°にある場合></p>  <p> : 法線制御軸旋回(CW) : 旋回しない : 軽度エラー(エラーコード: 1FC1H (詳細コード: 0103H)) </p> <p><G42.1法線制御軸が0°にある場合></p>  <p> : 法線制御軸旋回(CCW) : 旋回しない : 軽度エラー(エラーコード: 1FC1H (詳細コード: 0103H)) </p>

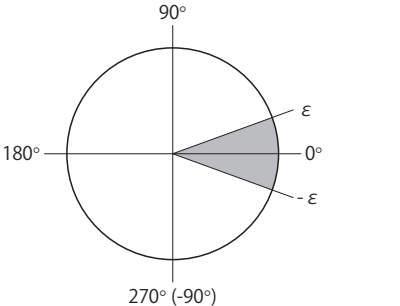
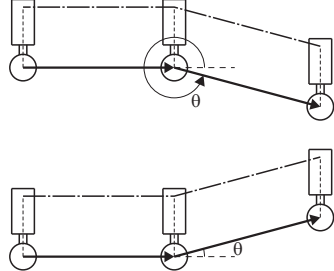
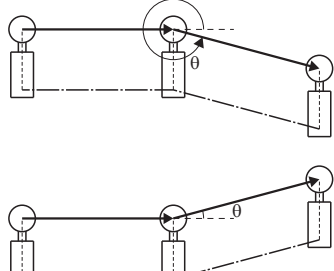
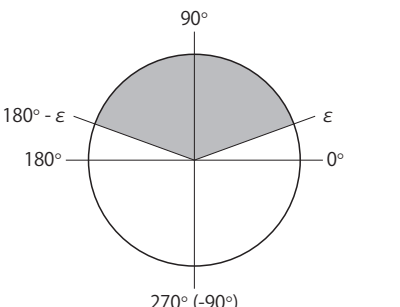
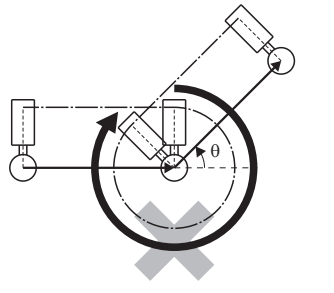
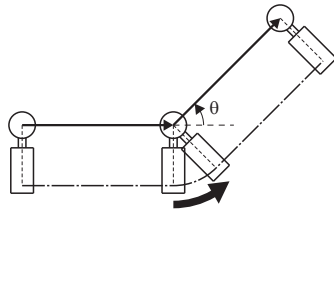
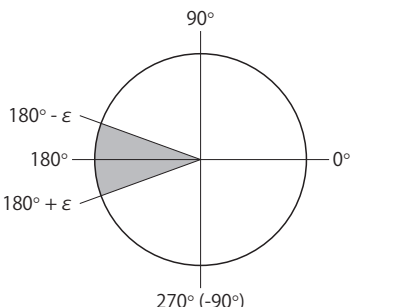
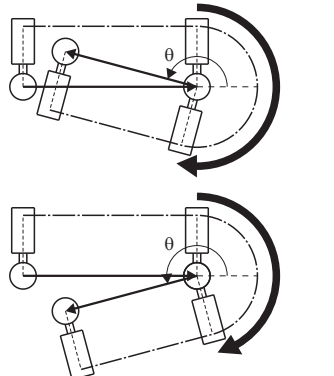
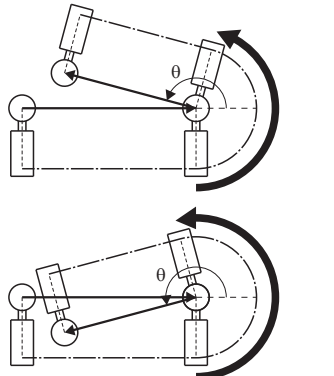
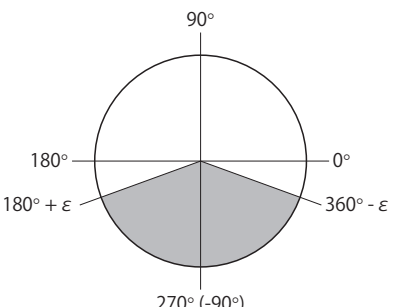
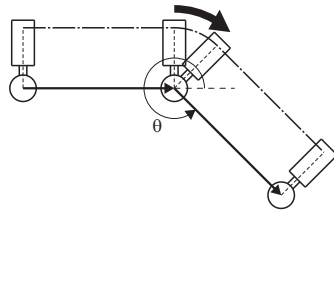
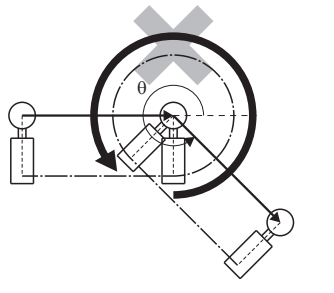
■法線制御タイプ

→: プログラム経路, ---: 工具先端経路

ブロック継ぎ目の法線制御軸旋回角度: θ $-\varepsilon < \theta < \varepsilon$	G41.1	G42.1
	 <p>旋回しません。</p>	 <p>旋回しません。</p>
	 <p>近回りします。</p>	 <p>近回りします。</p>
	 <p>近回りします。</p>	 <p>近回りします。</p>

■法線制御タイプII

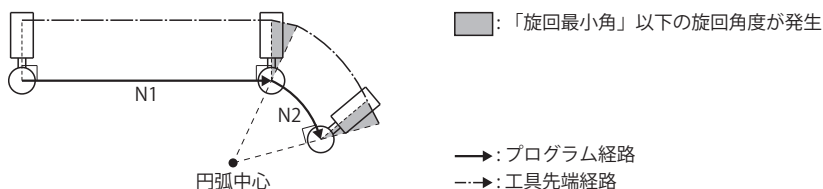
→: プログラム経路, —: 工具先端経路

ブロック継ぎ目の法線制御軸回転角度: θ $-\varepsilon < \theta < \varepsilon$	G41.1	G42.1
	 <p>旋回しません。</p>	 <p>旋回しません。</p>
	 <p>軽度エラー (エラーコード: 1FC3H(詳細コード: 0103H)となります。</p>	
	 <p>旋回しません。</p>	 <p>旋回しません。</p>
		 <p>軽度エラー (エラーコード: 1FC3H(詳細コード: 0103H)となります。</p>

旋回角度が旋回最小角以下となったときの動作

法線制御中、ブロック継ぎ目の旋回角度が旋回最小角以下となった場合、旋回動作が挿入されません。挿入されなかった分の旋回角度はブロック継ぎ目直後に補間しません。

- ブロック継ぎ目直後が直線補間の場合
旋回角度を補間しません。
- ブロック継ぎ目直後が円弧補間の場合
挿入されなかった分の旋回角度を円弧終点までに補間しません。



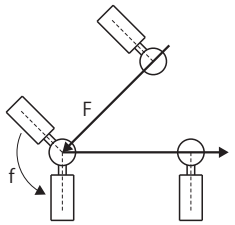
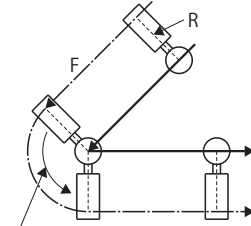
法線制御旋回軸の速度

法線制御旋回軸での速度は、法線制御タイプⅠと法線制御タイプⅡにより異なります。

■ブロック継ぎ目の法線制御旋回速度

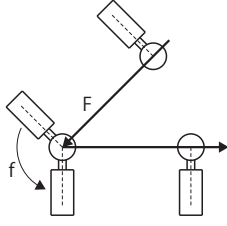
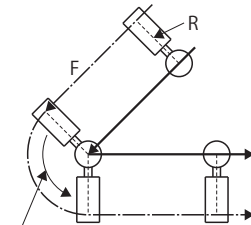
- ブロック継ぎ目直後が早送りの場合

——→: プログラム経路, ———: 工具先端経路

法線制御タイプ	法線制御旋回軸の速度
法線制御タイプⅠ	<p>[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“速度・時定数”⇒“早送り速度”にて設定した法線制御軸旋回速度となります。</p>  <p>法線制御軸旋回速度f = 早送り速度 × (早送りオーバーライド) [degree/min]</p>
法線制御タイプⅡ	<p>工具刃先での送り速度が早送り速度になります。法線制御軸旋回速度は、この早送り速度に従った法線制御軸速度となります。</p>  <p>$f = F \times 180 \div (\pi \times R)$</p> <p>法線制御軸旋回速度$f = F \times 180 \div (\pi \times R) \times (\text{早送りオーバーライド})$ [degree/min]</p> <p>R=0の場合は以下の式となります。 法線制御軸旋回速度$f = F \times (\text{早送りオーバーライド})$ [degree/min]</p> <p>F: Gコード制御軸パラメータの「早送り速度」[mm/min] R: Gコード制御系統パラメータの「法線制御軸旋回径」[mm](法線制御軸中心から工具刃先までの長さ)</p> <p>*: 法線制御軸旋回速度が早送り速度を超える場合は、早送り速度になります。</p>

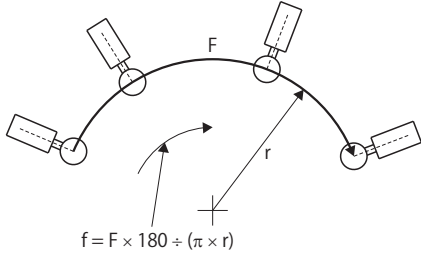
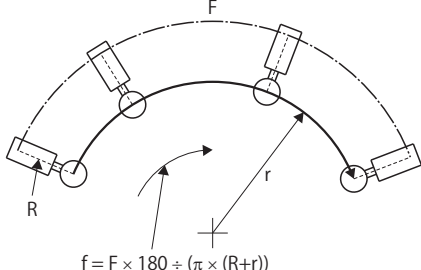
・ブロック継ぎ目直後が切削送りの場合

→: プログラム経路, —: 工具先端経路

法線制御タイプ	法線制御軸回転速度
法線制御タイプⅠ	<p>[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御システムパラメータ]⇒“法線制御”⇒“法線制御軸回転速度”にて設定した法線制御軸回転速度となります。</p>  <p>法線制御軸回転速度f = 法線制御軸回転速度 × (切削送りオーバライド)[degree/min]</p> <p>*: Gコード制御システムパラメータの「法線制御軸回転速度」は、法線制御軸の[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“速度・時定数”⇒“切削送りクランプ速度”にて設定した値以下で設定してください。法線制御軸の切削送りクランプ速度よりも大きい速度を設定すると、マルチCPUシステムの電源ON時、またはSTOP→RUN時に、中度エラー(エラーコード: 30FDH(詳細コード: 0012H))となります。</p>
法線制御タイプⅡ	<p>工具刃先での送り速度がF指令になります。法線制御軸回転速度は、このF指令に従った法線制御軸速度となります。</p>  <p>$f = F \times 180 \div (\pi \times R)$</p> <p>法線制御軸回転速度$f = F \times 180 \div (\pi \times R) \times$ (切削送りオーバライド)[degree/min]</p> <p>R=0の場合は以下の式となります。 法線制御軸回転速度$f = F \times$ (切削送りオーバライド)[degree/min]</p> <p>F: 送り速度指令[mm/min] R: Gコード制御システムパラメータの「法線制御軸回転径」[mm](法線制御軸中心から工具刃先までの長さ)</p> <p>*: 法線制御軸回転速度が切削送りクランプ速度を超える場合は、切削送りクランプ速度になります。</p>

■円弧補間中の法線制御旋回速度

→: プログラム経路, —: 工具先端経路

法線制御タイプ	法線制御軸旋回軸の速度
法線制御タイプⅠ	<p>法線制御軸旋回速度は、送り速度Fに従った回転速度となります。</p>  $f = F \times 180 \div (\pi \times r)$ <p>法線制御軸旋回速度 $f = F \times 180 \div (\pi \times r) \times (\text{切削送りオーバーライド})$ [degree/min]</p> <p>F: 送り速度指令 [mm/min] r: 円弧半径 [mm]</p>
法線制御タイプⅡ	<p>工具刃先での送り速度がF指令になります。法線制御軸旋回速度は、このF指令に従った回転速度となります。</p>  $f = F \times 180 \div (\pi \times (R + r))$ <p>法線制御軸旋回速度 $f = F \times 180 \div (\pi \times (R + r)) \times (\text{切削送りオーバーライド})$ [degree/min]</p> <p>F: 送り速度指令 [mm/min] R: Gコード制御系統パラメータの「法線制御軸旋回径」 [mm] (法線制御軸中心から工具刃先までの長さ) r: 円弧半径 [mm]</p>

Point

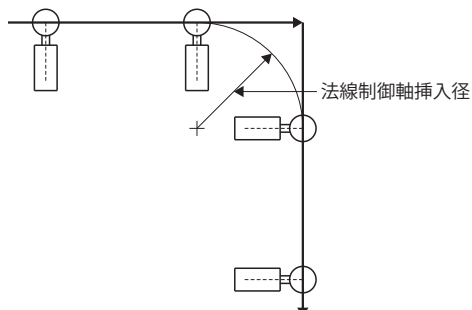
法線制御軸旋回速度がGコード制御軸パラメータの「切削送りクランプ速度」を超える場合は、以下となります。

- 法線制御軸旋回速度 = 切削送りクランプ速度
- 円弧補間中の軸の移動速度 = 法線制御軸の旋回速度に追従した速度

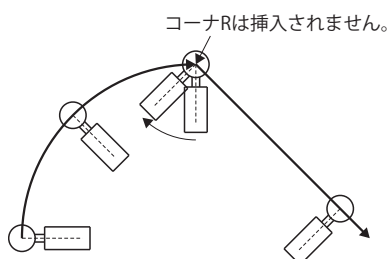
コーナ自動円弧挿入

法線制御中にて、平面選択している軸移動のコーナに自動で円弧を挿入します。コーナ自動円弧挿入は、法線制御タイプIで行います。挿入円弧の半径は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御システムパラメータ]⇒“法線制御”⇒“法線制御軸挿入径”にて設定します。Gコード制御システムパラメータの「法線制御軸」に「0: 法線制御しない」が設定されている場合は、コーナ自動円弧挿入は無効です。

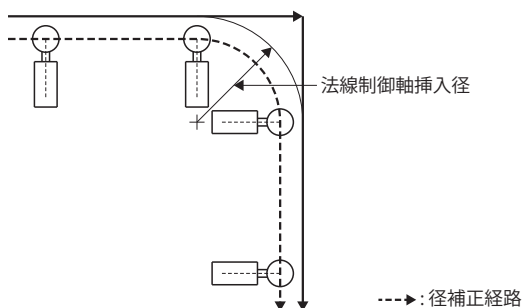
挿入円弧の補間中も法線制御を行います。



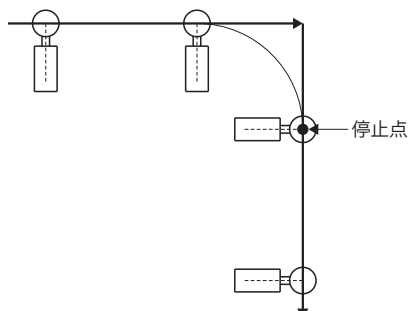
- 「直線→円弧」、「円弧→円弧」、「直線→移動ブロックなしのブロック」、「移動ブロックなしのブロック→直線、および挿入円弧の半径より短い直線」にはコーナ円弧を挿入しません。



- 径補正中は、コーナ円弧を挿入した経路に径補正がかかります。



- シングルブロックの停止点は下記ようになります。



- ・法線制御タイプI(連続する円弧補間の形状を法線制御で加工する場合)

- ・法線制御タイプⅡ(法線制御軸旋回径=15, プログラム経路の右側を法線制御で加工する場合)

6

各機能による法線制御の動作を下記に示します。

6 補助／応用機能 253
6.6 法線制御機能

注意事項

- 法線制御中、法線制御軸の移動に伴いプログラム座標を更新します。そのため、法線制御もプログラム座標系でプログラムしてください。
- シングルブロックの場合、法線制御軸は旋回開始位置で停止します。
- 法線制御中、法線制御軸(C軸)への移動指令は無視します。
- 法線制御中、法線制御軸(C軸)の目標位置に関する下記のデバイスは更新されません。法線制御を終了すると更新されます。
 - [Md.3148]機械目標位置(D54770+32sn, D54771+32sn)
 - [Md.3150]相対目標位置(D54774+32sn, D54775+32sn)
 - [Md.3152]プログラム目標位置(D54778+32sn, D54779+32sn)
- Gコード制御システムパラメータの「法線制御軸」には、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“系統軸情報”⇒“軸名称”で設定した軸名称を設定してください。設定した軸名称の軸がGコード制御軸パラメータの「軸名称」で設定されていない場合、マルチCPUシステムの電源ON時、またはSTOP→RUN時に、中度エラー (エラーコード: 30FDH(詳細コード: 0011H))になります。
- Gコード制御システムパラメータの「法線制御軸」には、回転軸を指定し、基本軸と重ならないように指定してください。法線制御が行えない不正な軸を指定すると、マルチCPUシステムの電源ON時、またはSTOP→RUN時に、中度エラー (エラーコード: 30FDH(詳細コード: 0012H))となります。
- 法線制御軸の移動も同時輪郭制御軸数の1軸としてカウントされます。法線制御軸の移動により同時輪郭制御軸数が仕様の軸数を超える場合は、軽度エラー (エラーコード: 1FC3H(詳細コード: 0302H))となります。

6.7 高精度制御

高精度制御では、制御系の遅れに起因する加工誤差を抑制できます。コーナ部でのエッジ出しが必要な加工や、曲線形状での内回り誤差を小さくする必要のある加工に有効です。複数ブロックを先読みすることで加工誤差が発生しない加減速を行うとともに、加工形状に応じた減速制御を自動的にを行い、加工時間の延長を最小にして加工誤差を抑制します。

形状	高精度制御	
	無効	有効
コーナ形状		
曲線形状		

高精度制御には、下記の機能があります。

- 補間前加減速(257ページ 補間前加減速)
- 最適速度制御(259ページ 最適速度制御)
- ベクトル精補間(265ページ ベクトル精補間)
- 円弧入口/出口速度制御(265ページ 円弧入口/出口速度制御)

Point

高精度制御モード中は、移動のないブロックを単独ブロックで指令しないでください。移動のないブロックを単独ブロックで指令すると、ブロックの切替わりに時間がかかることがあります。

各機能との組合せによる高精度制御の動作

各機能による高精度制御の動作を下記に示します。

機能	動作
法線制御	<div><ul style="list-style-type: none">法線制御と高精度制御モードを組合わせて使用する場合は、各軸許容加速度制御が有効となるようにパラメータを設定してください。各軸許容加速度制御が無効の場合、法線制御中にG61.1を指令すると、軽度エラー (エラーコード: 1FC3H(詳細コード: 0304H))となります。また、高精度制御モード中にG41.1指令、G42.1指令により法線制御を開始すると、軽度エラー (エラーコード: 1FC3H(詳細コード: 0304H))となります。法線制御中、旋回軸の単独動作は、補間後加減速で動作します。このため、旋回軸が動作するブロック間のつなぎでは、補間前加減速と補間後加減速が切替るため、減速チェックを行います。ブロック間動作は、ブロック間で旋回軸が単独動作しない場合^{*1}、旋回軸は円弧補間の動作に同期して動作するため、ブロックの継ぎ目で発生する加速度が、旋回軸の許容加速度以下になるように、最適な減速速度を算出し、その速度まで予め減速し、ブロック切替わり後、再び指令の速度まで加速します。<p>[動作例]</p><div><div><プログラム> (G61.1) : N1 G41.1 G90 G01 X10. F3000 N2 G02 X20. Y-10. I0. J-10.</div><div><動作></div><div><速度パターン></div><div></div></div><p>■注意事項</p><p>法線制御中の円弧補間時は、旋回軸の許容加速度にて速度がクランプされないため、円弧半径が小さいと旋回軸の加速度が旋回軸許容加速度より大きくなる場合があります。円弧半径が小さい場合は、旋回軸の許容加速度と比べて、全軸共通許容加速度が小さい値となるように設定してください。</p></div>
極座標補間	<div><ul style="list-style-type: none">極座標補間モードと高精度制御モードを組合わせて使用する場合は、各軸許容加速度制御が有効となるパラメータを設定してください。各軸許容加速度制御が無効の場合、極座標補間モード中に高精度制御モード指令(G61.1)を指令すると、軽度エラー (エラーコード: 1FC3H(詳細コード: 0322H))となります。また、高精度制御モード中に極座標補間モード開始指令(G12.1)を指令すると、軽度エラー (エラーコード: 1FC3H(詳細コード: 0324H))となります。極座標補間モード中は、円弧入口/出口速度制御は有効となりません。</div>

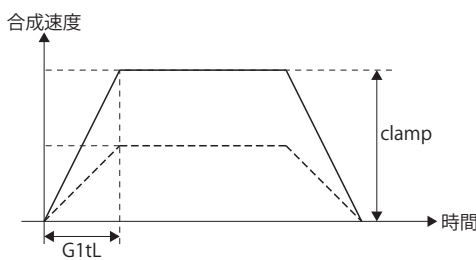
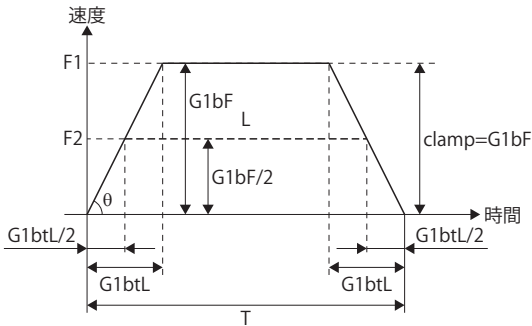
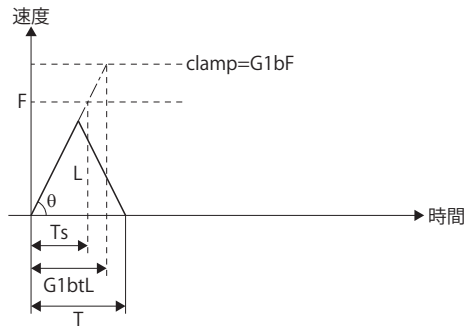
^{*1} 下記に示すいずれかの場合

- ブロック継ぎ目の旋回角度がGコード制御系統パラメータの「旋回最小角」以下。
- 後ろのブロックが直線補間で移動量がGコード制御系統パラメータの「旋回最小移動量」より小さい。
- 後ろのブロックが円弧補間で直径値がGコード制御系統パラメータの「旋回最小移動量」より小さい。

補間前加減速

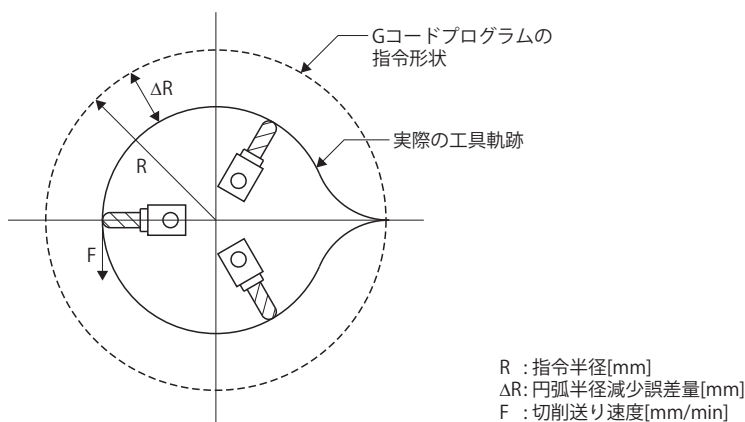
機械の移動開始/停止時の衝撃を抑えるため、移動指令に対し加減速制御を行い、速度波形を滑らかにしますが、高精度制御無効時の加減速処理は、補間後に行うためブロックでのつなぎ目のコーナが丸くなったり、指令の形状に対して軌跡誤差が発生したりします。高精度制御モードでは、以上の問題を解決するために、加減速制御を補間前に行います。この補間前加減速により、Gコードプログラムの指令形状に、忠実な加工軌跡で加工が可能となります。また、補間前加減速では傾き一定加減速を行うため加減速時間の短縮が可能となります。

直線補間指令における加減速制御の基本パターン

制御モード	加減速波形パターン	動作説明
通常モード (補間後加減速)	 <p>clamp: 切削送りクランプ速度 G1tL: G1時定数(リニア)</p>	<ul style="list-style-type: none"> 指令速度到達までの加速時間を一定にする加減速(時間一定加減速)のため、指令速度が遅いほど緩やかな加速/減速になります(加減速時間は変化しません)。 指令速度到達までの時間(G1時定数(リニア))は、各軸独立に設定できます。ただし、基本軸の時定数が異なると円弧形状にひずみが生じます。
高精度制御モード (補間前加減速)	<p>・ 補間距離が長く、速度が送り速度に到達する場合</p>  $T = \frac{L}{F1} + \frac{Ts \times F1}{G1bF} \quad \theta = \tan^{-1} \left(\frac{G1bF}{Ts} \right)$ <p>*: Tsは常にG1btL</p> <p>・ 補間距離が短く、速度が送り速度に到達しない場合</p>  $T = 2 \times \sqrt{Ts \times L / F} \quad TS = \frac{F \times G1btL}{G1bF} \quad \theta = \tan^{-1} \left(\frac{G1bF}{G1btL} \right)$ <p>F: 送り速度 T: 補間時間 Ts: 送り速度に到達する加減速時間 L: 補間距離 θ: 加減速の傾き clamp: 高精度制御モード用切削送りクランプ速度 G1bF: 補間前加減速-最高速度 G1btL: 補間前加減速-時定数</p>	<ul style="list-style-type: none"> パラメータで設定した最高速度(補間前加減速-最高速度)までの加速時間を一定にする加減速(傾き一定加減速)のため、指令速度が遅いほど加減速時間は短縮されます。 加減速時定数は1系統に対して、1つの値(各軸共通)となります。 <p>*: 補間前加減速-最高速度、補間前加減速-時定数は、加減速時の傾きを定めるためのもので、実際の切削送り最高速度は、Gコード制御軸パラメータの「高精度制御モード用切削送りクランプ速度」によりクランプされます。</p>

円弧補間指令における軌跡制御

円弧補間指令の場合、従来の補間後加減速制御方式では、加減速制御のスムージング処理が影響し、モーションCPUからサーボへ出力する軌跡が指令より内側になり、円弧半径が小さくなります。補間前加減速制御方式では、加減速制御の後に補間を行うため、モーションCPUからサーボへ出力する軌跡が指令通りになります。補間後の加減速制御と高精度制御モードにおける補間前加減速制御の各制御方式における円弧半径減少誤差量の比較を下記に示します。



上図のように、Gコードプログラムの指令形状が円弧形状の場合、実際の工具軌跡にはGコードプログラムの指令形状に対して ΔR の誤差が生じます。通常モード(補間後加減速)では、 ΔR はモーションCPUの加減速、およびサーボ系の遅れにより生じますが、高精度制御モード(補間前加減速)では、モーションCPUの加減速による誤差を0にすることができ、さらにフィードフォワード制御を使用することにより、サーボ系の遅れによる誤差も小さくすることが可能です。 ΔR はサーボ系の遅れにより生じます。

サーボ系の遅れは、サーボ系の位置ループ時定数(T_p [s])で表され、 T_p はサーボパラメータの「モデル制御ゲイン(PB07)」の逆数($T_p=1/PG1$)です。

フィードフォワード制御は、フィードフォワード係数(K_f)で表され、 K_f はサーボパラメータの「フィードフォワードゲイン(PB04)」です。

円弧半径減少誤差量 ΔR は、下表のように計算します。

制御モード	計算式
補間後加減速制御 (通常モード)	$\Delta R = \frac{1}{2R} \left(\frac{1}{12} T_s^2 + T_p^2 \left(1 - \left(\frac{K_f}{100} \right)^2 \right) \right) \left(\frac{F}{60} \right)^2$
補間前加減速制御 (高精度制御モード)	$\Delta R = \frac{1}{2R} T_p^2 \left(1 - \left(\frac{K_f}{100} \right)^2 \right) \left(\frac{F}{60} \right)^2$

*1 T_s : Gコード制御軸/パラメータの「G1時定数(リニア)」

*2 T_p : サーボ系の位置ループ時定数[s](サーボパラメータの「モデル制御ゲイン(PB07)」)

*3 K_f : フィードフォワード係数(サーボパラメータの「フィードフォワードゲイン(PB04)」)

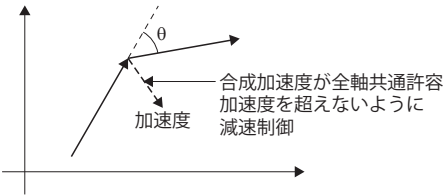
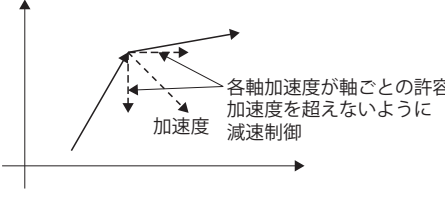
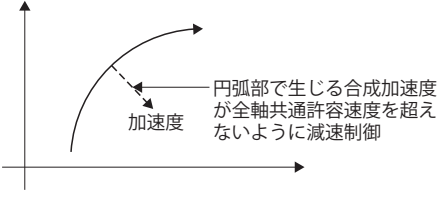
*4 F : 切削送り速度[mm/min]

円弧半径減少誤差量 ΔR を小さくするには、切削送り速度 F を小さくする必要があります。

最適速度制御

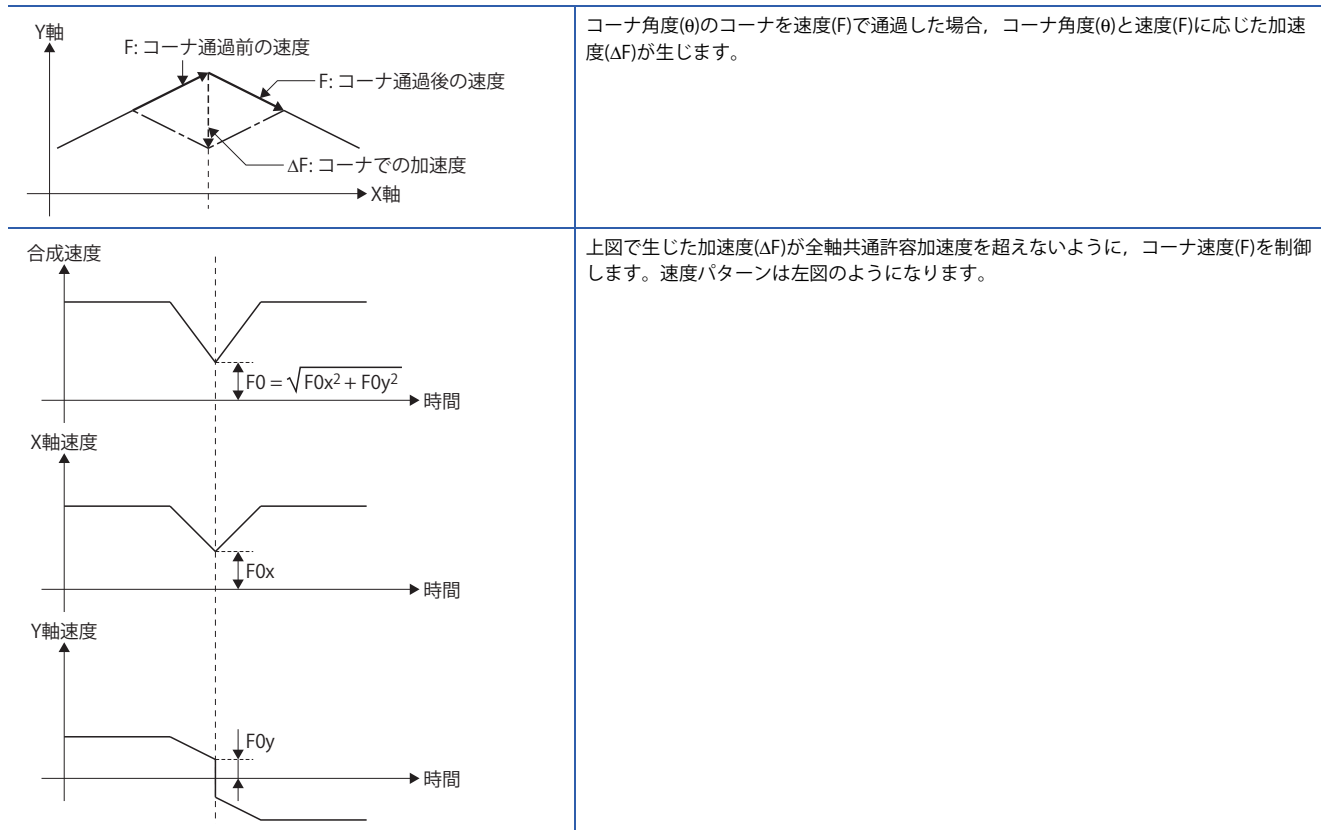
コーナや円弧など移動方向が変化するとき、その変化量と送り速度に応じた加速度が生じます。この加速度が大きいと、機械が振動し加工面に縞模様が残ることがあります。高精度制御モードでは、この問題を解決するために、発生する加速度がパラメータで設定した許容加速度以下になるように減速制御(最適速度制御)を行います。最適速度制御により、サイクルタイムの増加を最小限に抑えながら機械振動を抑制した高精度な加工が可能となります。

最適速度制御には、下記の機能があります。

機能	動作	内容
コーナ減速	最適コーナ減速	 <p>ブロックとブロックの継ぎ目で生じる合成加速度が、全軸共通許容加速度以下になるように減速制御します。これにより、エッジ精度の高い加工を行うことができます。</p> <p>最適コーナ減速を行うには、Gコード制御システムパラメータの「各軸許容加速度制御有効」にて「0: 最適コーナ減速」を設定します。</p>
	各軸許容加速度制御	 <p>ブロックとブロックの継ぎ目で生じる各軸の加速度が軸ごとに設定した許容加速度以下になるように減速制御します。</p> <p>各軸許容加速度制御は、各軸で特性(加速度)が異なる場合に有効な機能です。各軸の特性が同じであれば最適コーナ減速と同様の減速制御となります。</p> <p>各軸許容加速度制御を行うには、Gコード制御システムパラメータの「各軸許容加速度制御有効」にて「1: 各軸許容加速度制御」を設定します。</p>
円弧速度クランプ		<p>円弧部で生じる合成加速度が全軸共通許容加速度以下になるように減速制御します。これにより、円弧部に生じる軌跡誤差(円弧半径減少誤差量)を一定値以下に抑えることができます</p>

最適コーナ減速

ブロックとブロックの継ぎ目で生じる合成加速度が、Gコード制御システムパラメータの「補間前加減速-最高速度」と「補間前加減速-時定数」、および精度係数から決まる全軸共通許容加速度以下となるように減速制御を行うと、エッジ精度の高い加工が行えます。コーナ進入時、次のブロックとの角度(コーナ角度)と全軸共通許容加速度から最適な速度(最適コーナ速度)を算出して、その速度まであらかじめ減速し、コーナ通過後に再び指令の速度まで加速します。



ブロックとブロックが滑らかにつながっている場合は、減速する必要がないため最適コーナ減速を行いません。滑らかとなっているかの基準は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御システムパラメータ]⇒“高精度制御”⇒“コーナ減速角度”にて設定します。コーナ角度が設定したコーナ減速角度以下の場合は、滑らかであると判定し最適コーナ減速を行いません。更にエッジ精度を改善する場合は、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御システムパラメータ]⇒“高精度制御”⇒“コーナ精度係数”にて精度係数を設定します。

精度係数に上限値に近い値を設定すると、エッジ精度は向上しますが、最適コーナ速度が低下するため、サイクルタイムが伸びます。精度係数に負の値を設定すると、最適コーナ速度が上がり、サイクルタイムを短縮することができますが、エッジ精度は低下します。

全軸共通許容加速度は下記の式により求めることができます。

$$\text{全軸共通許容加速度}[\text{mm/s}^2] = \frac{\text{補間前加減速-最高速度}[\text{mm/min}] \div 60}{\text{補間前加減速-時定数}[\text{ms}] \div 1000} \times \frac{100 - \text{コーナ精度係数}}{100}$$

各軸許容加速度制御(最適加速度制御)

ブロックとブロックの継ぎ目で発生する加速度を各軸で評価し、ブロックの継ぎ目を最適な速度で通過する減速制御を行います。これにより、エッジ精度の高い加工が行えます。ブロックの継ぎ目で発生する各軸の加速度が、各軸許容加速度以下になるように、最適な減速速度を算出して、その速度まであらかじめ減速し、コーナ通過後に再び指令の速度まで加速します。ブロック間が滑らかにつながっている場合(各軸で発生する加速度が各軸許容加速度以下の場合)は減速しません。

各軸許容加速度制御を使用することで、特定の軸(回転軸)の許容加速度が低く機械振動を生じやすい場合でも、各軸の特性に応じた減速速度で減速するため、許容加速度の高い軸のみに加速度が生じるコーナでは、減速速度を高くすることができ、サイクルタイムを短縮できます。

下表のようにX軸(直線軸)、およびC軸(回転軸)に加速度が生じる場合のコーナ形状は、X軸、およびC軸で生じる加速度がX軸許容加速度、およびC軸許容加速度を超えないようにそれぞれコーナ速度(F)を制御します。

X軸の許容加速度がC軸の許容加速度よりも高い場合、X軸にのみ加速度が生じる経路では、C軸にのみ加速度が生じる経路よりも減速速度を高くすることができ、速度パターンは下表のようになります。

	X軸(直線軸)に加速度が生じる場合	C軸(回転軸)に加速度が生じる場合
コーナ形状		
速度パターン		

更にエッジ精度を改善する場合は、精度係数を大きくしてください。ただし、精度係数を大きくすると、最適コーナ速度が低下するため、サイクルタイムが伸びます。

精度係数に負の値を設定すると、最適コーナ速度が上がり、サイクルタイムを短縮できます。ただし、エッジ精度は低下します。

許容速度は、軸ごとに調整でき、下記の式により算出します。

$$\text{各軸共通許容加速度}[\text{mm/s}^2] = \frac{\text{各軸補間前切削送り-最高速度}[\text{mm/min}]^{*1} \div 60}{\text{各軸補間前切削送り-時定数}[\text{ms}]^{*2} \div 1000} \times \frac{100 - \text{コーナ精度係数}}{100} \times \frac{100 - \text{各軸精度係数}}{100}$$

*1 「0」が設定されている場合は、Gコード制御軸パラメータの「早送り速度」の速度を使用します。

*2 「0」が設定されている場合は、Gコード制御軸パラメータの「G0時定数(リニア)」の時定数を使用します。

■注意事項

- 基本軸I, 基本軸J, 基本軸Kに設定された3軸のGコード制御軸パラメータの「各軸補間前切削送り最高速度」と「各軸補間前切削送り時定数」がすべて「0」の場合, 基本軸I, 基本軸J, 基本軸Kの各軸許容加速度は, 基本軸I, 基本軸J, 基本軸Kのうち最も低い許容加速度に統一されます。

例

下表の値を設定した場合

基本軸I〜Kに設定された3軸(X, Y, Z)の各軸許容加速度は, 基本軸I〜Kのうち最も低いZ軸の許容加速度(8.3333[mm/s²])となります。

基本軸I〜Kに設定されていない軸(C軸)の許容加速度は, Gコード制御軸パラメータの「各軸補間前切削送り最高速度」と「各軸補間前切削送り時定数」から算出した許容加速度(83.3333[mm/s²])となります。(基本軸I〜Kのうち最も低い許容加速度には統一されません。)

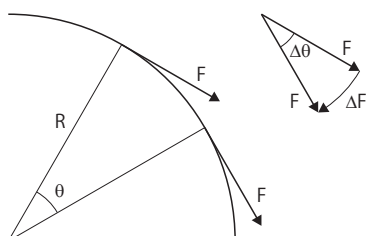
項目	X軸(基本軸I)	Y軸(基本軸J)	Z軸(基本軸K)	C軸
早送り速度	2000[mm/min]	2000[mm/min]	1000[mm/min]	2000[degree/min]
G0時定数(リニア)	1000[ms]	1000[ms]	2000[ms]	1000[ms]
各軸補間前切削送り-最高速度	0(未設定)	0(未設定)	0(未設定)	1000[degree/min]
各軸補間前切削送り-時定数	0(未設定)	0(未設定)	0(未設定)	200[ms]
各軸精度係数	0(未設定)	0(未設定)	0(未設定)	0(未設定)

- 基本軸I〜Kで許容加速度が異なる場合, 円弧形状でひずみが生じるため, 基本軸I〜Kの許容加速度は同じ値になるように設定してください。

円弧速度クランプ

円弧補間時は, 一定速度で移動中の場合でも, 常に進行方向が変化するため, 加速度が生じます。円弧半径が指令速度に対して十分大きいときは, 指令通りの速度で制御しますが, 円弧半径が比較的小さいときには, 発生する加速度がパラメータより算出される補間前加減速許容加速度を超えないように速度をクランプします。これにより, 円弧半径に応じた適正な送り速度での円弧切削が可能となります。

下図は半径(R)[mm]の円弧を一定速度(F)[mm/min]で移動した場合の加速度(ΔF)[mm/s²]を示します。ここで, 加速度(ΔF)が全軸共通許容加速度(Ac)[mm/s²]よりも小さくなる円弧クランプ速度(F')[mm/min]は, 下記の式から求めることができます。



F : 指令速度[mm/min]
R : 指令円弧半径[mm]
Δθ: 補間単位あたりの角度変化
ΔF: 補間単位あたりの速度変化

Fが全軸共通許容加減速(Ac)[mm/s²]を超えないような円弧クランプ速度で送ります。

$$F' \leq \sqrt{R \times Ac} \times 60 \dots\dots\dots (1)$$

$$\Delta F' = \frac{\text{補間前加減速-最高速度[mm/min]}}{\text{補間前加減速-時定数[ms]}} \dots\dots\dots (2)$$

(1)の式を補間前加減速の最大理論円弧半径減少誤差量(ΔR)を表す式(258ページ 円弧補間指令における軌跡制御)のFに代入すると, 指令半径(R)が消去され, ΔRはRに依存しなくなります。

サーボ系の位置ループ時定数(Tp[s])は, サーボパラメータの「モデル制御ゲイン(PB07)」の逆数(Tp=1/PG1)です。

フィードフォワード係数(Kf)は, サーボパラメータの「フィードフォワードゲイン(PB04)」です。

$$\Delta R = \frac{1}{2R} T_p^2 \left(1 - \left(\frac{K_f}{100} \right)^2 \right) \left(\frac{F}{60} \right)^2 \Rightarrow \Delta R = \frac{Ac}{2} T_p^2 \left(1 - \left(\frac{K_f}{100} \right)^2 \right)$$

すなわち, 円弧クランプ速度でクランプされる円弧指令では, 指令半径(R)に関係なく, 常に一定値以内の半径減少誤差量(ΔR)で加工できます。更に真円度を改善する場合には, 精度係数を大きくしてください。ただし, 精度係数を大きくすると, 円弧クランプ速度が低下するため, サイクルタイムが伸びます。

精度係数に負の値を設定すると, 円弧クランプ速度が上がり, サイクルタイムを短縮することができます。ただし, 真円度は低下します。

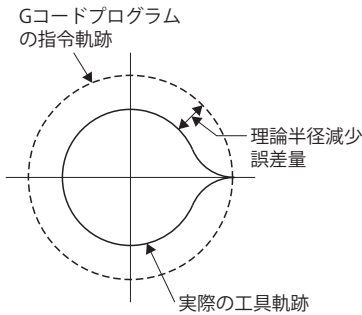
全軸共通許容加速度は下記の式により求めることができます。

$$\text{全軸共通許容加速度[mm/s}^2] = \frac{\text{補間前加減速-最高速度[mm/min]} \div 60}{\text{補間前加減速-時定数[ms]} \div 1000} \times \frac{100 - \text{曲線精度係数}}{100}$$

■高精度制御の各パラメータと速度/加速度の関係

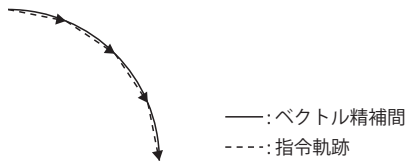
高精度制御の各パラメータと速度/加速度の関係を下表に示します。

項目	内容	使用パラメータ	
切削送り加速度 [m/s ²]	使用するパラメータのデータに基づき計算します。	■Gコード制御システムパラメータ ・補間前加減速-最高速度 ・補間前加減速-時定数	
各軸切削送り加速度 [m/s ²]	使用するパラメータのデータに基づき計算します。	■Gコード制御軸パラメータ ・各軸補間前切削送り-最高速度 ・各軸補間前切削送り-時定数	
各軸コーナ許容加速度 [m/s ²]	使用するパラメータのデータに基づき計算します。	■Gコード制御システムパラメータ ・コーナ精度係数 ■Gコード制御軸パラメータ ・各軸補間前切削送り-最高速度 ・各軸補間前切削送り-時定数 ・各軸精度係数	
各軸曲線許容加速度 [m/s ²]	使用するパラメータのデータに基づき計算します。	■Gコード制御システムパラメータ ・曲線精度係数 ■Gコード制御軸パラメータ ・各軸補間前切削送り-最高速度 ・各軸補間前切削送り-時定数 ・各軸精度係数	
コーナ減速速度 [mm/min]	角度(外角) θ [degree]のコーナに対するコーナ減速速度は、使用するパラメータのデータに基づき計算します。	<最適コーナ減速> ■Gコード制御システムパラメータ ・補間前加減速-最高速度 ・補間前加減速-時定数 ・コーナ減速角度 ・コーナ精度係数	<各軸許容加速度制御> ■Gコード制御システムパラメータ ・コーナ減速角度 ・コーナ精度係数 ■Gコード制御軸パラメータ ・各軸補間前切削送り-最高速度 ・各軸補間前切削送り-時定数 ・各軸精度係数
直角コーナ減速速度 [mm/min]	角度(外角)90[degree]のコーナに対するコーナ減速速度は、使用するパラメータのデータに基づき計算します。	<最適コーナ減速> ■Gコード制御システムパラメータ ・補間前加減速-最高速度 ・補間前加減速-時定数 ・コーナ精度係数	<各軸許容加速度制御> ■Gコード制御システムパラメータ ・コーナ精度係数 ■Gコード制御軸パラメータ ・各軸補間前切削送り-最高速度 ・各軸補間前切削送り-時定数 ・各軸精度係数
理論コーナダレ量 [mm]	角度(外角) θ [degree]のコーナに対するコーナダレ量[mm]は、使用するパラメータのデータに基づき計算します。	<最適コーナ減速> ■Gコード制御システムパラメータ ・補間前加減速-最高速度 ・補間前加減速-時定数 ・コーナ減速角度 ・コーナ精度係数 ■サーボパラメータ(制御軸1の値) ・フィードフォワードゲイン(PB04) ・モデル制御ゲイン(PB07)	<各軸許容加速度制御> ■Gコード制御システムパラメータ ・コーナ減速角度 ・コーナ精度係数 ■Gコード制御軸パラメータ ・各軸補間前切削送り-最高速度 ・各軸補間前切削送り-時定数 ・各軸精度係数 ■サーボパラメータ(制御軸1の値) ・フィードフォワードゲイン(PB04) ・モデル制御ゲイン(PB07)
理論直角ダレ量 [mm]	角度(外角)90[degree]のコーナに対するコーナダレ量[mm]は、使用するパラメータのデータに基づき計算します。	<最適コーナ減速> ■Gコード制御システムパラメータ ・補間前加減速-最高速度 ・補間前加減速-時定数 ・コーナ減速角度 ・コーナ精度係数 ■サーボパラメータ(制御軸1の値) ・フィードフォワードゲイン(PB04) ・モデル制御ゲイン(PB07)	<各軸許容加速度制御> ■Gコード制御システムパラメータ ・コーナ減速角度 ・コーナ精度係数 ■Gコード制御軸パラメータ ・各軸補間前切削送り-最高速度 ・各軸補間前切削送り-時定数 ・各軸精度係数 ■サーボパラメータ(制御軸1の値) ・フィードフォワードゲイン(PB04) ・モデル制御ゲイン(PB07)

項目	内容	使用パラメータ	
理論半径減少誤差量 [mm]	<p>半径Rの円弧に対する理論半径減少誤差量[mm]は、使用するパラメータのデータに基づき計算します。</p> 	<p><最適コーナ減速></p> <p>■Gコード制御系統パラメータ</p> <ul style="list-style-type: none"> ・補間前加減速-最高速度 ・補間前加減速-時定数 ・曲線精度係数 <p>■サーボパラメータ(制御軸1の値)</p> <ul style="list-style-type: none"> ・フィードフォワードゲイン(PB04) ・モデル制御ゲイン(PB07) 	<p><各軸許容加速度制御></p> <p>■Gコード制御系統パラメータ</p> <ul style="list-style-type: none"> ・曲線精度係数 <p>■Gコード制御軸パラメータ</p> <ul style="list-style-type: none"> ・各軸補間前切削送り-最高速度 ・各軸補間前切削送り-時定数 ・各軸精度係数 <p>■サーボパラメータ(制御軸1の値)</p> <ul style="list-style-type: none"> ・フィードフォワードゲイン(PB04) ・モデル制御ゲイン(PB07)
R5mm円弧クランプ速度 [mm/min]	半径5mmの円弧に対する円弧減速速度[mm/min]は、使用するパラメータのデータに基づき計算します。	<p><最適コーナ減速></p> <p>■Gコード制御系統パラメータ</p> <ul style="list-style-type: none"> ・補間前加減速-最高速度 ・補間前加減速-時定数 ・曲線精度係数 	<p><各軸許容加速度制御></p> <p>■Gコード制御系統パラメータ</p> <ul style="list-style-type: none"> ・曲線精度係数 <p>■Gコード制御軸パラメータ</p> <ul style="list-style-type: none"> ・各軸補間前切削送り-最高速度 ・各軸補間前切削送り-時定数 ・各軸精度係数
R1mm円弧クランプ速度 [mm/min]	半径1mmの円弧に対する円弧減速速度は、使用するパラメータのデータに基づき計算します。	<p><最適コーナ減速></p> <p>■Gコード制御系統パラメータ</p> <ul style="list-style-type: none"> ・補間前加減速-最高速度 ・補間前加減速-時定数 ・曲線精度係数 	<p><各軸許容加速度制御></p> <p>■Gコード制御系統パラメータ</p> <ul style="list-style-type: none"> ・曲線精度係数 <p>■Gコード制御軸パラメータ</p> <ul style="list-style-type: none"> ・各軸補間前切削送り-最高速度 ・各軸補間前切削送り-時定数 ・各軸精度係数

ベクトル精補間

微小線分指令時、ブロックとブロックのつながりの角度が非常に小さく滑らかな場合(最適コーナ減速をしない場合)、ベクトル精補間機能により更に滑らかに補間します。



円弧入口/出口速度制御

直線→円弧、円弧→直線のつながり目では、加速度変動が生じ機械振動となることがあります。円弧入口/出口速度制御は、円弧に侵入する前、および円弧から出る場合に減速速度まで減速し、機械振動を低減させるための機能です。ただし、最適コーナ減速、または各軸許容加速度制御(最適加速度制御)と重なった場合は、減速速度の低い方が有効となります。円弧入口/出口速度制御の有効/無効は、Gコード制御システムパラメータの「円弧減速速度切換」で設定します。また、減速速度は、Gコード制御システムパラメータの「円弧減速速度」で設定します。

動作例

■コーナ減速ではない場合

プログラム	動作	速度パターン
(G61.1) ∴ N1 G01 X-10. F3000. N2 G02 X-5. Y-5. J-2.5 N3 G01 Y10. ∴		

■コーナ減速となる場合

プログラム	動作	速度パターン
(G61.1) ∴ N1 G01 X-10. F3000. N2 G02 X-5. Y-5. I-2.5 N3 G01 X10. ∴		

6.8 タンデム機能

タンデム機能は、Gコード制御中にて、指定した軸(マスタ軸)の移動指令と同一の指令を別の軸(スレーブ軸)に出力する機能です。タンデム機能にて使用する軸を、[モーション制御パラメータ]⇒[Gコード制御パラメータ]⇒[Gコード制御軸パラメータ]⇒“タンデム機能”⇒“マスタ制御軸名称”にて設定します。タンデム機能の有効情報は、“[Md.3153]タンデム機能有効情報(D54755+32sn)”にて確認できます。

注意事項

- Gコードプログラムでは、スレーブ軸に設定した軸アドレスを指定することはできません。スレーブ軸に設定した軸アドレスを指定した場合、その指令は無視されます。
- スレーブ制御軸のステータス、およびモニタデバイスの有効/無効の内容について下表に示します。

デバイス	有効/無効の内容
[St.3076]スムージングゼロ(D54448.0+2sn)	ONのままとなります。
[Md.3153]タンデム機能有効情報(D54755+32sn)	マスタ制御軸No.を格納します。 *: タンデム機能設定処理を行ったときにのみ更新します。
[Md.3147]機械位置(D54768+32sn, D54769+32sn)	更新します。
[Md.3148]機械目標位置(D54770+32sn, D54771+32sn)	更新しません。
[Md.3149]相対位置(D54772+32sn, D54773+32sn)	更新します。
[Md.3150]相対目標位置(D54774+32sn, D54775+32sn)	更新しません。
[Md.3152]プログラム目標位置(D54778+32sn, D54779+32sn)	更新しません。

- タンデム機能を使用するときは、Gコード制御要求をONする前に、マスタ制御軸とスレーブ制御軸の基準位置を一致させてください。
- Gコード制御軸パラメータの「早送り速度」、「切削送りクランプ速度」、「G0時定数(リニア)」、「G1時定数(リニア)」は、スレーブ制御軸No.に設定した設定値は無視され、マスタ制御軸No.の設定値に従います。
- スレーブ制御軸のストアードストロークリミットは、下記の動作となりますので注意してください。

Gコード制御	動作
Gコード制御要求時	スレーブ軸の設定に基づきエラーチェックを実行
Gコード制御中	無効

- マスタ軸とスレーブ軸に設定する軸の単位は、[モーション制御パラメータ]⇒[軸設定パラメータ]⇒“固定パラメータ”⇒“単位設定”にて同一の単位に合わせてください。

6.9 GOTによるGコードプログラムの操作

GOTとモーションCPUを接続して、GOTの画面上でGコードプログラムの入出力、プログラムの編集を行うことができます。GOTでGコードプログラムを操作可能な対応機種、エンジニアリングソフトウェアバージョンを以下に示します。

対応機種

品名	形名
GOT2000シリーズ	解像度がSVGA, XGAのGT25, GT27 ・GT25□□-S ・GT27□□-S ・GT27□□-X

エンジニアリングソフトウェア

■表示器画面作成ソフトウェア

品名	形名	エンジニアリングソフトウェアバージョン
MELSOFT GT Works3 ・GT Designer3 Version1(GOT2000)	SW1DND-GTWK3-J	1.185T

■システムアプリケーション(拡張機能)

品名	機能名
システムアプリケーション	モーションプログラム入出力 モーションプログラム編集

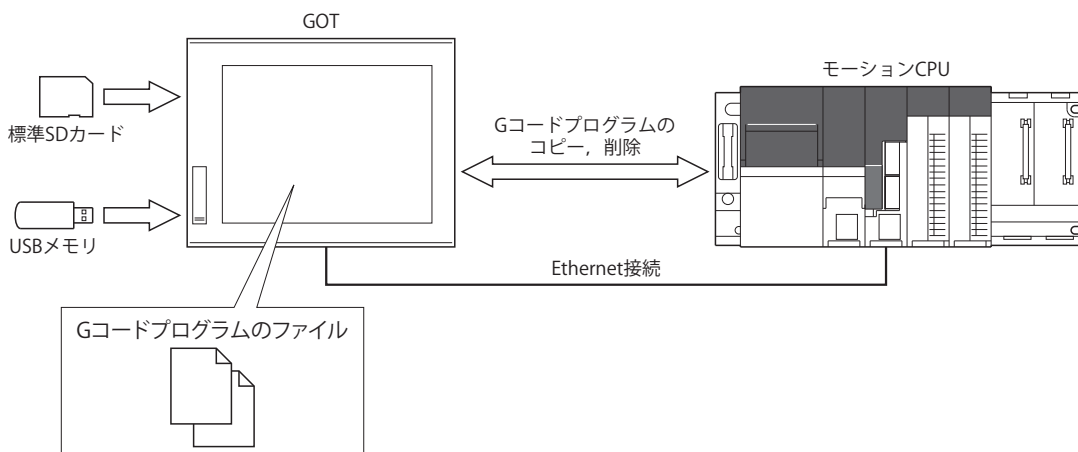
GOTのプログラム入出力機能

GOTと接続したモーションCPUとメモリカード間で、Gコードプログラムの入出力をGOTの画面上で行えます。

- ・GOTからテキストファイル形式でモーションGコードプログラムを個別に、入出力できます。
- ・Gコードプログラムの個別、または一括での削除が行えます。

プログラム入出力の詳細は、下記を参照してください。

📖 GOT2000シリーズ本体取扱説明書(モニタ編)



GOTのプログラム編集機能

GOTと接続したモーションCPUのGコードプログラムをGOTの画面上で編集を行います。

- ・プログラム一覧画面からモーションCPU内のGコードプログラムを個別に選択し、編集できます。ただし、Gコードプログラムの新規登録や削除はできません。
- ・Gコードプログラムの表示、編集、行挿入、行削除、文字列検索が行えます。

プログラム編集の詳細は、下記を参照してください。

📖 GOT2000シリーズ本体取扱説明書(モニタ編)

6.10 マクロに関する機能

変数指令、演算指令、制御指令を使用してマクロ指令のプログラムを作成できます。

変数指令、演算指令、制御指令については、下記を参照してください。

- 変数指令(173ページ 変数指令)
- 演算指令(184ページ 演算指令)
- 制御指令(193ページ 制御指令)

マクロ指令を使用する際の注意事項

ユーザマクロ指令を使用すると従来の移動指令、M指令などの制御指令と演算、判定、分岐などのマクロ指令とを組合せて加工プログラムが作成できます。従来の移動指令、M指令などの制御指令を実行文、演算、判定、分岐などのマクロ指令をマクロ文とします。

マクロ文の処理は、直接機械制御に関係がないため、通常加工時はマクロ文を一括して処理し、加工時間を短縮しています。マクロ文を1ブロックごとに実行させる場合は、“[Rq.3384]マクロシングル(D54226.C+2s)”をONしてください。各マクロ文を1ブロックごとに実行させることで、マクロ文の動作確認が容易になります。マクロシングル設定有効時は、“[St.3234]マクロシングル有効中(D54441.F+4s)”がONします。

Point

- マクロ文は、以下の指令を指します。
 - 演算指令(=を含むブロック)
 - 制御指令(GOTO, DO~END 等を含むブロック)
 - マクロ呼出し指令(G65)
- 実行文は、マクロ文以外を指します。

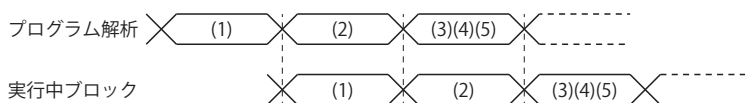
プログラム例

プログラムによる処理を下記に示します。

動作	プログラム	備考
(1)	N1 G91 G28 X0 Y0	実行文
(2)	N2 G00 X-100. Y-100.	実行文
(3)	N3 #101 = 100. * COS[210.]	マクロ文
(4)	N4 #103 = 100. * SIN[210.]	マクロ文
(5)	N5 G01 X#101 Y#103 F800	実行文

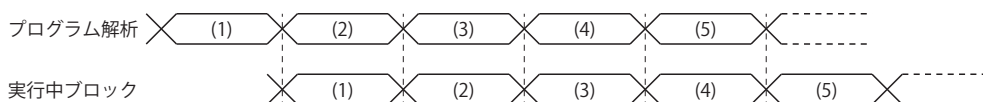
■マクロシングルOFF

N2の制御と並行してN3, N4, N5を処理します。N2の制御中にN3, N4, N5の解析が終了すれば、機械制御は連続して動作します。



■マクロシングルON

N2の制御と並行してN3を処理します。N2の制御が終了してからN4, N5を解析しN5を実行するため、N4, N5の解析時間だけ機械制御が待ち状態となります。



マクロ指令の使用例

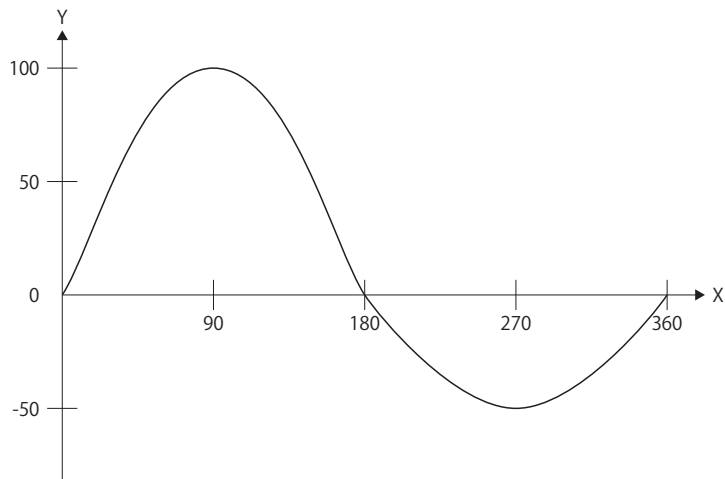
角度に応じて振幅を変化させる(SINカーブ)場合のプログラム例を下記に示します。

メインプログラム(O102.gcd)

```
%  
:  
#101=0  
#102=360  
#103=100  
#109=100  
M98 P103  
:  
M02  
%
```

サブプログラム(O103.gcd)

```
%  
WHILE [#101 LE #102] DO1  
#104=#103*SIN [#101]  
IF [#101 GT 180] THEN #104=#104/2  
G90 G01 X#101 Y#104 F#109  
#101=#101+10  
END1  
M99  
%
```



*1 上記は軌跡のイメージです。本プログラム例は10[degree]刻みのため、実際の軌跡は上記よりも粗くなります。

付録

付1 Gコード制御のエラー詳細コード

Gコード制御エラー詳細コード

Gコード制御エラー検出時の詳細コードについて示します。

詳細情報1

■Gコード制御パラメータエラー (軽度エラー (エラーコード: 1FC0H)), Gコード制御構成エラー (中度エラー (エラーコード: 30FDH))

Gコード制御パラメータエラー (軽度エラー (エラーコード: 1FC0H)), Gコード制御構成エラー (中度エラー (エラーコード: 30FDH))検出時の詳細コードについて示します。

詳細コード	エラー名称	異常内容と原因	処置方法
0010H	基本軸設定不正	<ul style="list-style-type: none">基本軸I, 基本軸J, 基本軸Kで軸名が重複している。Gコード制御系統内に指定した軸名がない。	<ul style="list-style-type: none">基本軸I, 基本軸J, 基本軸Kは, 軸名が重複しないよう設定してください。Gコード制御軸パラメータの「軸名称」で設定された軸名を指定してください。
0011H	法線制御軸名称不正	法線制御軸名称で指定した軸名称の軸が設定されていない。	Gコード制御軸パラメータの「軸名称」で設定された軸名を指定してください。
0012H	法線制御軸設定不正	<ul style="list-style-type: none">法線制御軸が回転軸でない。法線制御軸が基本軸I, 基本軸J, 基本軸Kに設定されている。法線制御軸旋回速度が切削送りクランプ速度より大きい。	<ul style="list-style-type: none">法線制御軸で使用する軸は, 回転軸を設定してください。法線制御軸で使用する軸は, 基本軸I, 基本軸J, 基本軸Kに設定しないように設定を見直してください。法線制御軸旋回速度は, 切削送りクランプ速度以下の値に設定してください。
0013H	極座標補間軸名称不正	<ul style="list-style-type: none">極座標補間直線軸で指定した軸名称の軸が設定されていない。極座標補間回転軸で指定した軸名称の軸が設定されていない。極座標補間直線軸と極座標補間回転軸のどちらかのみ設定されている。	<ul style="list-style-type: none">Gコード制御軸パラメータの「軸名称」で設定された軸名を指定してください。極座標補間を使用する場合は, 極座標補間直線軸と極座標補間回転軸の両方を設定してください。
0014H	極座標補間軸設定不正	<ul style="list-style-type: none">極座標補間直線軸が直線軸でない。極座標補間回転軸が回転軸でない。極座標補間直線軸が基本軸I, 基本軸J, 基本軸Kに設定されていない。極座標補間回転軸が基本軸I, 基本軸J, 基本軸Kに設定されている。極座標補間回転軸が法線制御軸に設定されている。	<ul style="list-style-type: none">極座標補間直線軸で使用する軸は, 直線軸を設定してください。極座標補間回転軸で使用する軸は, 回転軸を設定してください。極座標補間直線軸で使用する軸は, 基本軸I, 基本軸J, 基本軸Kに設定した軸を使用してください。極座標補間回転軸で使用する軸は, 基本軸I, 基本軸J, 基本軸Kに設定しないように設定を見直してください。極座標補間回転軸で使用する軸は, 法線制御軸に設定しないように設定を見直してください。
0015H	高精度制御設定不正	<ul style="list-style-type: none">補間前加減速-最高速度が切削送りクランプ速度より小さい。補間前加減速-最高速度が高精度制御モード用切削送りクランプ速度より小さい	<ul style="list-style-type: none">補間前加減速-最高速度がG切削送りクランプ速度以上となるようにしてください。補間前加減速-最高速度が高精度制御モード用切削送りクランプ速度以上となるようにしてください。

詳細コード	エラー名称	異常内容と原因	処置方法
0016H	マクロ制御設定不正	<ul style="list-style-type: none"> ・系統共通コモン変数点数と系統別コモン変数の合計点数が900点を超えた。 ・系統共通コモン変数点数を設定している状態で、系統共通コモン変数先頭デバイス番号が設定されていない。 ・系統別コモン変数点数を設定している状態で、系統別コモン変数先頭デバイス番号が設定されていない。 ・系統共通コモン変数と系統別コモン変数で使用するデバイスが重複している。 ・系統別コモン変数で使用するデバイスが系統間で重複している。 ・ローカル変数先頭デバイス番号+(660-1)がデバイス点数設定範囲を超えている。 ・ローカル変数と系統共通コモン変数で使用するデバイス番号が重複している。 ・ローカル変数と系統別コモン変数で使用するデバイス番号が重複している。 ・ローカル変数で使用するデバイス番号が系統間で重複している。 	<ul style="list-style-type: none"> ・系統共通コモン変数点数と系統別コモン変数の合計点数は、900点以下となるように設定してください。 ・系統共通コモン変数を使用する場合は、系統共通コモン変数先頭デバイス番号を設定してください。 ・系統別コモン変数を使用する場合は、系統別コモン変数先頭デバイス番号を設定してください。 ・系統共通コモン変数と系統別コモン変数で使用するデバイスが重複しないように設定を見直してください。 ・系統別コモン変数で使用するデバイスが系統間で重複しないように設定を見直してください。 ・ローカル変数先頭デバイス番号+(660-1)がデバイス点数設定範囲内になるよう設定を見直してください。 ・ローカル変数と系統共通コモン変数で使用するデバイス番号が重複しないように設定を見直してください。 ・ローカル変数と系統別コモン変数で使用するデバイス番号が重複しないように設定を見直してください。 ・ローカル変数で使用するデバイス番号が系統間で重複しないように設定を見直してください。
0020H	軸No.設定不正	<ul style="list-style-type: none"> ・サーボネットワーク設定にない軸を設定している。 ・Gコード制御内で軸No.が重複している。 	<ul style="list-style-type: none"> ・サーボネットワーク設定に設定済みの軸No.を指定してください。 ・軸No.が重複しないように設定してください。
0021H	軸名称不正	Gコード制御系統内で軸名称が重複している。	Gコード制御系統内で、軸名称が重複しないように設定してください。
0022H	制御単位不正	<ul style="list-style-type: none"> ・回転軸を設定している状態で、軸No.で指定した軸の単位設定が[mm], [inch]となっている。 ・直線軸を設定している状態で、軸No.で指定した軸の単位設定が[inch], [degree]となっている。 	<ul style="list-style-type: none"> ・回転軸を設定する場合は、固定パラメータの単位設定に[mm], [inch]以外を設定してください。 ・直線軸を設定する場合は、固定パラメータの単位設定に[inch], [degree]以外を設定してください。
0023H	タンデム機能設定不正	<ul style="list-style-type: none"> ・基本軸I, 基本軸J, 基本軸Kに割当てられている軸をスレーブ制御軸として設定した。 ・法線制御軸に割当てられている軸をスレーブ制御軸として設定した。 ・極座標補間直線軸, 極座標補間回転軸に割当てられている軸をスレーブ制御軸として設定した。 ・軸名称が「U」, 「V」, 「W」以外の軸をスレーブ制御軸として設定した。 ・マスタ制御軸名称で指定した軸名称の軸が設定されていない。 ・マスタ制御軸がスレーブ制御軸に設定されている。 ・マスタ制御軸とスレーブ制御軸の下記パラメータが一致していない。 Gコード制御軸パラメータの「回転軸」 Gコード制御軸パラメータの「回転軸タイプ」 	<ul style="list-style-type: none"> ・Gコード制御軸パラメータの「軸名称」で設定された軸名を指定してください。 ・スレーブ制御軸で使用する軸は、基本軸I, 基本軸J, 基本軸Kに設定しないように設定を見直してください。 ・スレーブ制御軸で使用する軸は、法線制御軸に設定しないように設定を見直してください。 ・スレーブ制御軸で使用する軸は、極座標補間直線軸, 極座標補間回転軸に設定しないように設定を見直してください。 ・スレーブ制御軸で使用する軸の軸名称は「U」, 「V」, 「W」としてください。 ・マスタ制御軸がスレーブ制御軸とならないよう設定を見直してください。 ・マスタ制御軸とスレーブ制御軸の「回転軸」, 「回転軸タイプ」が一致するよう設定を見直してください。
0024H	ソフトリミット設定不正	<ul style="list-style-type: none"> ・ソフトリミット+よりもソフトリミット-の方が大きい。 ・回転軸タイプが、近回り有効、もしくは近回り無効のとき、ソフトリミット-とソフトリミット+が一致していない。 	<ul style="list-style-type: none"> ・ソフトリミット+よりもソフトリミット-の方が小さくなるように設定値を見直してください。 ・回転軸タイプが、近回り有効もしくは近回り無効のときは、ソフトリミット-とソフトリミット+の設定を一致させてください。
0040H	Gコードプログラムファイル異常	Gコードプログラムのファイルが異常である。	Gコードプログラムを再度書込みしてください。 再度同じエラーを表示した場合は、モーションコントローラのハードウェア異常の恐れがあります。最寄りの三菱電機システムサービス株式会社、または当社の支社、代理店に症状を説明し、ご相談ください。
0041H	Gコードプログラム容量オーバー	<ul style="list-style-type: none"> ・Gコードプログラムがファイルサイズ上限を超えている。 ・Gコードプログラムの合計サイズが、プログラム容量を超えた。 	<ul style="list-style-type: none"> ・Gコードプログラムがファイルサイズ上限に収まるように修正してください。 ・Gコードプログラムの合計サイズがプログラム容量に収まるように修正してください。(ファイルサイズ上限とプログラム容量は、性能仕様を参照してください。〔P.19ページ 性能仕様〕)
0042H	Gコードプログラムフォーマット不正	プログラムスタート、またはプログラムエンドにエンドオブコード(%)がない。	プログラムスタートとプログラムエンドにエンドオブコード(%)を設定してください。

■Gコード制御オペレーションエラー (軽度エラー (エラーコード: 1FC1H))

Gコード制御オペレーションエラー (軽度エラー (エラーコード: 1FC1H))検出時の詳細コードについて示します。

詳細コード	エラー名称	異常内容と原因	処置方法
0101H	ストアードストロークリミットオーバ	ストアードストロークリミット範囲外の状態となった。	<ul style="list-style-type: none"> Gコード制御をOFFした状態で、リミット範囲内に機械を移動させてください。 Gコード制御軸パラメータの「ストアードストロークリミット」を見直してください。
0102H	運転モードなし	運転モードの指定がない。	運転モードを設定してください。
0103H	法線制御中ブロック継ぎ目旋回停止	<ul style="list-style-type: none"> 法線制御中のブロック継ぎ目の旋回角度が制限を超えた。 法線制御タイプⅡにて円弧の内側を旋回する場合、Gコード制御系統パラメータの「法線制御軸旋回径」の設定値が円弧半径より大きくなっている。 	<ul style="list-style-type: none"> Gコード系統パラメータの「旋回最小角」の設定値、または旋回角度を見直してください。 Gコード系統パラメータの「法線制御軸旋回径」の設定値を見直してください。 “[St.3208]Gコード制御中”がONの状態のまま運転を開始する場合は、“[Rq.3380]リセット指令”をOFF→ONしてください。
0104H	リセット処理中	“[St.3216]リセット中”がONしている状態で“[Rq.3377]自動運転起動(サイクルスタート)”をONした。	“[St.3217]リセット完了”がONするのを待ってから、“[Rq.3377]自動運転起動(サイクルスタート)”をONしてください。
0111H	外部停止信号ON	<ul style="list-style-type: none"> Gコード制御要求時、外部停止信号がONしていた。 Gコード制御中、外部停止信号がONした。 	外部停止信号をOFFの状態で、“[Rq.3376]Gコード制御要求”をONしてください。
0112H	ハードウェアストロークリミット+	<ul style="list-style-type: none"> Gコード制御要求時、外部信号FLS(上限LS)がOFFしていた。 Gコード制御中、外部信号FLS(上限LS)がOFFした。 	逆方向へ移動し外部リミット範囲内の状態で、“[Rq.3376]Gコード制御要求”をONしてください。
0113H	ハードウェアストロークリミット-	<ul style="list-style-type: none"> Gコード制御要求時、外部信号RLS(下限LS)がOFFしていた。 Gコード制御中、外部信号RLS(下限LS)がOFFした。 	正方向へ移動し外部リミット範囲内の状態で、“[Rq.3376]Gコード制御要求”をONしてください。
0114H	シーケンサレディ OFF	<ul style="list-style-type: none"> Gコード制御要求時、“[Rq.1120]シーケンサレディフラグ”、または“PCPU準備完了フラグ”がOFFしていた。 Gコード制御中、“[Rq.1120]シーケンサレディフラグ”、または“PCPU準備完了フラグ”がOFFした。 	<ul style="list-style-type: none"> モーションCPUをRUNにしてください。 “[Rq.1120]シーケンサレディフラグ”をONにしてください。 上記後、“[Rq.3376]Gコード制御要求”をONしてください。
0115H	始動受け付けON	“[St.1040]始動受け付けフラグ”がONしていた。	“[St.1040]始動受け付けフラグ”がOFFしている状態で“[Rq.3376]Gコード制御要求”をONしてください。
0116H	原点復帰未完	“[St.1069]原点復帰要求”がONしていた。	原点復帰を実施した後に、“[Rq.3376]Gコード制御要求”をONしてください。
0117H	サーボエラー検出信号ON	<ul style="list-style-type: none"> Gコード制御要求時、“[St.1068]サーボエラー検出”がONしていた。 Gコード制御中、“[St.1068]サーボエラー検出”がONした。 	サーボ側のエラーを除去した後、“[Rq.1148]サーボエラーリセット指令”により、“[St.1068]サーボエラー検出”をリセット後、“[Rq.3376]Gコード制御要求”をONしてください。
0118H	サーボレディ OFF	<ul style="list-style-type: none"> Gコード制御要求時、“[St.1075]サーボレディ”がOFFしていた。 (1) サーボアンプの電源OFF (2) サーボアンプの電源ONによるイニシャル処理中 (3) サーボアンプ未実装 (4) サーボエラー発生 (5) ケーブル不良 (6) “[Rq.1155]サーボOFF指令”がON Gコード制御中、サーボアンプ電源OFFとなった。(サーボ未実装検知、ケーブル異常など) 	<ul style="list-style-type: none"> サーボアンプ電源ONにしてください。または、サーボアンプとの接続ケーブルを確認してください。 “[St.1075]サーボレディ”がONしている状態で“[Rq.3376]Gコード制御要求”をONしてください。
0119H	Gコード制御終了中のシーケンサレディ ON	Gコード制御終了中に、“[Rq.1120]シーケンサレディフラグ”が再度OFF→ONとなった。	全系統の“[St.3208]Gコード制御中”がOFF後、“[Rq.1120]シーケンサレディフラグ”をONにした状態で、“[Rq.3376]Gコード制御要求”をONしてください。

■Gコード制御停止エラー (軽度エラー (エラーコード: 1FC2H))

Gコード制御停止エラー (軽度エラー (エラーコード: 1FC2H))検出時の詳細コードについて示します。

詳細コード	エラー名称	異常内容と原因	処置方法
0201H	リセット指令ON	“[Rq.3380]リセット指令”がONしているため、自動運転起動(サイクルスタート)ができない。	“[Rq.3380]リセット指令”がOFFの状態を確認してから操作してください。
0202H	自動運転休止(フィードホールド)ON	“[Rq.3378]自動運転休止(フィードホールド)”がONしている。	“[Rq.3378]自動運転休止(フィードホールド)”をOFFしてください。
0203H	運転モードなし	運転モードが選択されていない。	“[Rq.3381]プログラム運転モード(メモリモード)”をONしてください。
0221H	ストアードストロークリミット	ストアードストロークリミットになった。	<ul style="list-style-type: none"> プログラムを修正してください。 Gコード制御軸パラメータの「ストアードストロークリミット」を見直してください。 “[St.3208]Gコード制御中”がONの状態のまま運転を開始する場合は、“[Rq.3380]リセット指令”をOFF→ONしてください。
0222H	運転モード変化	<ul style="list-style-type: none"> 自動運転中に運転モードを解除した。 自動運転中に他の運転モードに変えた。 	自動運転終了後に運転モードを解除してください。

■Gコード制御プログラムエラー (軽度エラー (エラーコード: 1FC3H))

Gコード制御プログラムエラー (軽度エラー (エラーコード: 1FC3H))検出時の詳細コードについて示します。

詳細コード	エラー名称	異常内容と原因	処置方法
0301H	前計算不能	先読みを必要とする命令の組合せにより先読みブロック数が制限を超えた。	先読みを必要とする命令の組合せを少なくするか、もしくはなくしてください。
0302H	同時軸数オーバ	同一ブロックに指令した軸アドレス数が仕様より多い。	<ul style="list-style-type: none"> エラーが発生しているブロックの指令を2つに分割してください。 軸数の仕様を確認してください。
0303H	軸名称設定不正	プログラム指令の軸アドレス名とパラメータ設定の軸アドレス名が異なる。	プログラムの軸名称を見直してください。
0304H	指令不可状態	<ul style="list-style-type: none"> 法線制御できないモーダル中に法線制御指令(G40.1, G41.1, G42.1)を指令した。 法線制御中に変更できないモーダルを変更しようとした。 	プログラムを修正してください。
0305H	不正アドレス	仕様にないアドレスを使用した。	仕様を確認してプログラムのアドレスを修正してください。
0306H	フォーマットエラー	プログラム上の指令フォーマットが正しくない。	プログラムを修正してください。
0307H	不正Gコード	仕様にないGコードを指令した。	プログラムのGコードアドレスを修正してください。
0308H	指令値オーバ	各アドレスの設定範囲をオーバーした。	プログラムを修正してください。
0309H	プログラムエンドエラー	プログラム運転中に“EOF”を読み込んだ。	プログラムの最後にM02, またはM30を入れてください。
030AH	ヘリカル動作不可	円弧補間指令で平面選択されていない軸を指令している。	円弧補間指令で平面選択されていない軸を指定する場合は、指令を次のブロックに移動してください。
030DH	補正番号なし	補正(G41, G42, G43, G44)指令時、補正番号(D□□, H□□)の指令がない。または補正番号が仕様の組数より大きい。	<ul style="list-style-type: none"> 補正指令ブロックに補正番号の指令を付け加えてください。 補正番号組数を確認して、補正組数以内の補正番号の指令に修正してください。
030EH	割り出しエラー	指令単位で割り切れない軸指令を行った。	プログラムを修正してください。
030FH	Gコード組合せ不正	同一ブロックで指令できないGコードを指令した。	同一ブロックに指令できないGコードを別ブロックに分割してください。
0310H	補間長さオーバ	指令移動距離が大きいく。(2 ⁻³¹ を超えている)	軸アドレスの指令範囲を修正してください。
0311H	F指令なし	最初の切削指令(G01, G02, G03)においてF指令がない。	F指令で送り速度を指令してください。
0312H	F指令値オーバ	F指令, または,F指令が指令範囲を超えています。	F指令, または,F指令が指令範囲内となるように修正してください。
0313H	円弧半径差オーバ	円弧の始点, 終点および円弧中心が間違っている。	<ul style="list-style-type: none"> プログラムの始点, 終点, 円弧中心および半径指定アドレスの数値を修正してください。 Gコード制御システムパラメータの「円弧誤差」を見直してください。
0314H	円弧中心計算不能	R指定円弧補間時, 円弧の中心が求まらない。	プログラムの各アドレスの数値を修正します。
0315H	円弧モーダル中径補正	円弧モーダル(G02, G03)中に補正指令(G40, G41, G42)を指令した。	径補正指令ブロックまたは, キャンセルブロックでは, 直線指令(G01)または, 早送り指令(G00)を指令してください。(モーダルを直線補間にしてください。)

詳細コード	エラー名称	異常内容と原因	処置方法
0316H	工具径補正中平面選択	工具径補正(G41, G42)中に平面選択指令(G17, G18, G19)を指令した。	工具径補正指令を完了(G40のキャンセル指令以降に軸移動指令を指令)してから、平面選択指令を指令してください。
0317H	平面選択不正	円弧指令軸と選択平面とが違っている。	正しい平面選択で円弧指令を行ってください。
0318H	演算不可	演算式が正しくない。	プログラムを修正してください。
0319H	ゼロ割り	割り算の分母がゼロである。	プログラムを修正し、演算式の割り算の分母がゼロにならないようにしてください。
031AH	小数点指令不可	小数点指令が使用できないアドレスで小数点指令を行った。	プログラムを修正してください。
031BH	法線制御軸不正	法線制御軸が未設定である。	法線制御軸を修正してください。
031CH	法線制御中平面選択	法線制御中に、平面選択指令(G17, G18, G19)を指令した。	法線制御中のプログラムから平面選択指令(G17, G18, G19)を削除してください。
031DH	交点なし	工具径補正(G41, G42)実行時、干渉ブロック処理において交点が算出できない。	<ul style="list-style-type: none"> プログラムを修正してください。 Gコード制御加工パラメータの「干渉チェック」を見直してください。
031EH	補正干渉不正	工具径補正(G41, G42)実行時に干渉エラーが発生した。	プログラムを修正してください。
031FH	プログラム番号なし	<ul style="list-style-type: none"> Gコードプログラムが登録されていない。 Gコードプログラムの取込み実行中に「[Rq.337]自動運転起動(サイクルスタート)」をONした。 	<ul style="list-style-type: none"> プログラム番号を修正してください。 該当プログラム番号のGコードプログラムを書き込んでください。 Gコードプログラムの取込み完了後に「[Rq.337]自動運転起動(サイクルスタート)」をONしてください。
0320H	シーケンス番号なし	指定したシーケンス番号がプログラム中に設定されていない。	<ul style="list-style-type: none"> シーケンス番号を修正してください。 適当なブロックにシーケンス番号を設定してください。
0321H	ブロック番号なし	指定したブロック番号がプログラム中にない。	ブロック番号を修正してください。
0322H	不正Gコード(極座標補間)	極座標補間中に、極座標補間と組合せできないGコードを使用した。	プログラムを修正してください。
0323H	不正軸指令(極座標補間)	極座標補間中に指令できない軸指令を行った。	プログラムを修正してください。
0324H	不正モード(極座標補間)	<ul style="list-style-type: none"> 法線制御中に極座標補間モード開始指令を行った。 工具径補正中に極座標補間モード開始指令を行った。 プログラム座標回転モード中に極座標補間モード開始指令を行った。 高精度制御モード中に極座標補間モード開始指令を行った。(各軸許容加速度制御が無効な場合) 極座標補間中の工具径補正中に自動コーナオーバーライドを指令した。 	<ul style="list-style-type: none"> 法線制御を終了してから極座標補間モードを開始してください。 工具径補正キャンセル状態で極座標補間モードを開始してください。 プログラム座標回転キャンセル状態で極座標補間モードを開始してください。 高精度制御を終了してから極座標補間モードを開始してください。 各軸許容加速度制御を有効にしてください。 極座標補間中の工具径補正中は自動コーナオーバーライドを指令しないでください。
0325H	極座標補間軸不正	極座標補間軸が未設定の状態で極座標補間モード開始指令を行った。	極座標補間軸を設定してください。
0326H	座標回転中平面選択	座標回転指令中に平面選択指令(G17, G18, G19)を指令した。	座標回転指令後、必ず座標回転キャンセル指令を実行してから、平面選択指令(G17, G18, G19)を指令してください。
0327H	座標回転後増分値指令	プログラム座標回転モード開始指令(G68)直後、およびプログラム座標回転モードキャンセル指令(G69)直後の移動指令を増分値指令で指令した。	プログラム座標回転モード開始指令(G68)直後、およびプログラム座標回転モードキャンセル指令(G69)直後の移動指令は絶対値指令で指令してください。
0328H	サブ/マクロプログラム繰り返し回数オーバ	サブ/マクロプログラムの繰り返し回数が範囲外の値を指定した。	サブ/マクロプログラムの繰り返し回数を修正してください。
0329H	サブ/マクロプログラム深さオーバ	サブ/マクロプログラムからサブプログラムを順次呼出す回数が10回を超えた。	サブ/マクロプログラムの呼出し回数が10回を超えないようにプログラムを修正してください。
032AH	変数番号なし	範囲外の変数番号を指定した。	変数番号を修正してください。
032BH	変数定義=なし	変数を定義するとき、「=」を指令していない。	プログラムの変数定義に「=」を設定してください。
032CH	演算指令/制御指令フォーマットエラー	同一ブロックに演算指令(=のあるブロック)、または制御指令(分岐、繰り返し)と、その他の指令(移動指令等)が混在している。	プログラムを修正し、演算指令(=のあるブロック)、または(分岐、繰り返し)と、その他の指令(移動指令等)を別ブロックに記述してください。
032DH	括弧深さオーバ	1ブロック中の括弧の数が5重よりも多くなった。	括弧の数が5重以下になるようにプログラムを修正してください。
032EH	括弧数不一致	1ブロックに指令された括弧の数が異なる。	括弧の数がペアになるようにプログラムを修正してください。

詳細コード	エラー名称	異常内容と原因	処置方法
032FH	IF深さオーバ	IF文の深さが10回を越えた。	IF文の深さが10回を越えないようにプログラムを修正してください。
0330H	IF文不一致	<ul style="list-style-type: none"> IFとENDIFがペアになっていない。 IF命令がない状態でTHEN/ELSEを指令した。 	<ul style="list-style-type: none"> IFとENDIFがペアになるようプログラムを修正してください。 THEN/ELSEの前に、IFを指令してください。
0331H	IF文不正	<ul style="list-style-type: none"> 条件式が括弧で括られていない。 THEN, またはELSEを指令していない。 	<ul style="list-style-type: none"> 条件式を括弧で括ってください。 THEN, またはELSEを指令してください。
0332H	WHILE文不正	<ul style="list-style-type: none"> 条件式が括弧で括られていない。 繰り返し識別番号に範囲外の値を指定した。 	<ul style="list-style-type: none"> 条件式を括弧で括ってください。 繰り返し識別番号を修正してください。
0333H	DO-END深さオーバ	WHILE[条件式]DOロ～ENDロ文の深さが27回を超えた。	DO～END文の深さが27回を越えないようにプログラムを修正してください。
0334H	DO-END不一致	DOとENDがペアになっていない。	DO～ENDがペアになるようにプログラムを修正してください。
0335H	マクロ呼出し深さオーバ	マクロプログラムを順次呼出す回数が4回を超えた。	マクロ呼出し回数が4回を超えないようにプログラムを修正してください。
0336H	マクロ引数組数オーバ	マクロ呼出しで引数指定Ⅱを使用したとき、引数の組数が10組を超えた。	マクロ呼出しの引数指定Ⅱの組数が10組を超えないようにプログラムを修正してください。

アドオンライセンスエラー詳細コード

Gコード制御にてアドオンライセンスエラー検出時の詳細コードについて示します。

詳細情報1

■アドオンライセンスワーニング(警告(エラーコード:0EF1H)), アドオンライセンス軽度エラー (軽度エラー (エラーコード:1FF1H)), アドオンライセンス中度エラー (中度エラー (エラーコード:3081H))

アドオンライセンスワーニング(警告(エラーコード:0EF1H)), アドオンライセンス軽度エラー (軽度エラー (エラーコード:1FF1H)), アドオンライセンス中度エラー (中度エラー (エラーコード:3081H))検出時の詳細コードについては、下記を参照してください。

📖MELSEC iQ-R モーションコントローラプログラミングマニュアル(共通編)

付

付2 Gコード制御のイベント詳細コード

Gコード制御イベント詳細コード

Gコード制御イベント検出時の詳細コードについて示します。

詳細情報1

■Gコード制御システム情報(情報(イベントコード: 07FAH))

Gコード制御システム情報(情報(イベントコード: 07FAH))検出時の詳細コードについて示します。

詳細コード	イベント名称	内容
0401H	Gコード制御要求ON	"[Rq.3376]Gコード制御要求"をOFF→ONした。
0402H	Gコード制御要求OFF	"[Rq.3376]Gコード制御要求"をON→OFFした。
0403H	プログラム始動中のリセット指令ON	プログラム始動中に"[Rq.3380]リセット指令"をOFF→ONした。
0404H	自動運転休止(フィードホールド)ON	"[Rq.3378]自動運転休止(フィードホールド)"をONした。 ("[Rq.3377]自動運転起動(サイクルスタート)"をOFF→ONで自動運転を再開できます。)
0405H	シングルブロック停止	シングルブロックにより停止した。 ("[Rq.3377]自動運転起動(サイクルスタート)"をOFF→ONで自動運転を再開できます。)
0406H	切削オーバーライドゼロ	切削送りオーバーライドに「0」, または範囲外の値を指定した。
0407H	早送りオーバーライドゼロ	早送りオーバーライドに「0」, または範囲外の値を指定した。

MEMO

付

改訂履歴

*取扱説明書番号は、本説明書の裏表紙の左下に記載してあります。

改訂年月	*取扱説明書番号	改訂内容
2017年7月	IB(名)-0300370-A	初版
2017年12月	IB(名)-0300370-B	■追加機能 極座標補間モード(G12.1, G13.1), ローカル座標系設定(G52), 高精度制御モード(G61.1), プログラム座標回転モード(G68, G69), タンデム機能, GOTによる機能 ■追加・修正箇所 2.1節, 2.2節, 2.4節, 3章, 3.1節, 3.2節, 3.4節, 3.8節, 3.9節, 3.11節, 4.1節, 4.2節, 4.3節, 5.2節, 5.4節, 5.5節, 5.6節, 6.3節, 6.5節, 6.5節, 6.7節, 6.8節, 6.9節, 付1, 付2
2018年6月	IB(名)-0300370-C	■追加機能 サブプログラム呼出し(M98), サブプログラム復帰(M99), 変数指令, 演算指令, 制御指令 ■追加・修正箇所 安全上のご注意, 1.1節, 1.2節, 1.4節, 3.5節, 3.7節, 3.8節, 3.12節, 4.1節, 4.2節, 5.1節, 5.2節, 5.4節, 5.5節, 5.6節, 5.7節, 5.8節, 5.9節, 5.10節, 6.1節, 6.5節, 6.6節, 6.7節, 6.8節, 6.10節, 付1
2018年11月	IB(名)-0300370-D	■追加機能 マクロ呼出し(G65) ■追加・修正箇所 安全上のご注意, 1.2節, 1.3節, 3章, 3.5節, 3.7節, 3.8節, 4.1節, 5.2節, 5.3節, 5.4節, 5.6節, 5.7節, 5.8節, 5.10節, 6.5節, 6.10節, 付1

本書によって、工業所有権その他の権利の実施に対する保証、または実施権を許諾するものではありません。また本書の掲載内容の使用により起因する工業所有権上の諸問題については、当社は一切その責任を負うことができません。

© 2017 MITSUBISHI ELECTRIC CORPORATION

保証について

ご使用に際しましては、以下の製品保証内容をご確認いただきますよう、よろしくお願いいたします。

1. 無償保証期間と無償保証範囲

無償保証期間中に、製品に当社側の責任による故障や瑕疵（以下併せて「故障」と呼びます）が発生した場合、当社はお買い上げいただきました販売店または当社サービス会社を通じて、無償で製品を修理させていただきます。

ただし、国内および海外における出張修理が必要な場合は、技術者派遣に要する実費を申し受けます。

また、故障ユニットの取替えに伴う現地再調整・試運転は当社責務外とさせていただきます。

【無償保証期間】

製品の無償保証期間は、お客様にてご購入後またはご指定場所に納入後 36 ヶ月とさせていただきます。

ただし、当社製品出荷後の流通期間を最長 6 ヶ月として、製造から 42 ヶ月を無償保証期間の上限とさせていただきます。

また、修理品の無償保証期間は、修理前の無償保証期間を超えて長くなることはありません。

【無償保証範囲】

(1) 一次故障診断は、原則として貴社にて実施をお願い致します。

ただし、貴社要請により当社、または当社サービス網がこの業務を有償にて代行することができます。この場合、故障原因が当社側にある場合は無償と致します。

(2) 使用状態・使用方法、および使用環境などが、取扱説明書、ユーザーズマニュアル、製品本体注意ラベルなどに記載された条件・注意事項などに従った正常な状態で使用されている場合に限定させていただきます。

(3) 無償保証期間内であっても、以下の場合には有償修理とさせていただきます。

① お客様における不適切な保管や取扱い、不注意、過失などにより生じた故障およびお客様のハードウェアまたはソフトウェア設計内容に起因した故障。

② お客様にて当社の了解なく製品に改造などの手を加えたことに起因する故障。

③ 当社製品がお客様の機器に組み込まれて使用された場合、お客様の機器が受けている法的規制による安全装置または業界の通念上備えられているべきと判断される機能・構造などを備えていれば回避できたと認められる故障。

④ 取扱説明書などに指定された消耗部品が正常に保守・交換されていれば防げたと認められる故障。

⑤ 消耗部品（バッテリー、リレー、ヒューズなど）の交換。

⑥ 火災、異常電圧などの不可抗力による外部要因および地震、雷、風水害などの天変地異による故障。

⑦ 当社出荷当時の科学技術の水準では予見できなかった事由による故障。

⑧ その他、当社の責任外の場合またはお客様が当社責任外と認めた故障。

2. 生産中止後の有償修理期間

(1) 当社が有償にて製品修理を受け付けることができる期間は、その製品の生産中止後 7 年間です。

生産中止に関しましては、当社テクニカルニュースなどにて報じさせていただきます。

(2) 生産中止後の製品供給（補用品も含む）はできません。

3. 海外でのサービス

海外においては、当社の各地域 FA センターで修理受付をさせていただきます。ただし、各 FA センターでの修理条件などが異なる場合がありますのでご了承ください。

4. 機会損失、二次損失などへの保証責務の除外

無償保証期間の内外を問わず、以下については当社責務外とさせていただきます。

(1) 当社の責に帰すことができない事由から生じた障害。

(2) 当社製品の故障に起因するお客様での機会損失、逸失利益。

(3) 当社の予見の有無を問わず特別の事情から生じた損害、二次損害、事故補償、当社製品以外への損傷。

(4) お客様による交換作業、現地機械設備の再調整、立上げ試運転その他の業務に対する補償。

5. 製品仕様の変更

カタログ、マニュアルもしくは技術資料などに記載の仕様は、お断りなしに変更させていただく場合がありますので、あらかじめご承知おきください。

以 上

購入に関するお問い合わせ

製品の購入のご検討やご相談はこちらからお問い合わせください。

三菱電機株式会社

本社機器営業部	〒110-0016	東京都台東区台東1-30-7 (秋葉原アイマークビル)	(03) 5812-1430
北海道支社	〒060-8693	札幌市中央区北二条西4-1 (北海道ビル)	(011) 212-3793
東北支社	〒980-0013	仙台市青葉区花京院1-1-20 (花京院スクエア)	(022) 216-4546
関東支社	〒330-6034	さいたま市中央区新都心11-2 (明治安田生命さいたま新都心ビル)	(048) 600-5835
新潟支店	〒950-8504	新潟市中央区東大通2-4-10 (日本生命ビル)	(025) 241-7227
神奈川支社	〒220-8118	横浜市西区みなとみらい2-2-1 (横浜ランドマークタワー)	(045) 224-2623
北陸支社	〒920-0031	金沢市広岡3-1-1 (金沢パークビル)	(076) 233-5502
中部支社	〒450-6423	名古屋市中村区名駅3-28-12 (名古屋ビルヂング)	(052) 565-3326
豊田支店	〒471-0034	豊田市小坂本町1-5-10 (矢作豊田ビル)	(0565) 34-4112
関西支社	〒530-8206	大阪市北区大深町4-20 (グランフロント大阪タワーA)	(06) 6486-4120
中国支社	〒730-8657	広島市中区中町7-32 (ニッセイ広島ビル)	(082) 248-5445
四国支社	〒760-8654	高松市寿町1-1-8 (日本生命高松駅前ビル)	(087) 825-0055
九州支社	〒810-8686	福岡市中央区天神2-12-1 (天神ビル)	(092) 721-2251

サービスのお問い合わせ

修理・サービスに関するお問い合わせはこちらにお問い合わせください。

三菱電機システムサービス株式会社

北日本支社	(022) 353-7814	関西支社	(06) 6458-9728
北海道支店	(011) 890-7515	京滋機器サービスステーション	(075) 611-6211
東京機電支社	(03) 3454-5521	姫路機器サービスステーション	(079) 269-8845
神奈川機器サービスステーション	(045) 938-5420	中四国支社	(082) 285-2111
関東機器サービスステーション	(048) 859-7521	岡山機器サービスステーション	(086) 242-1900
新潟機器サービスステーション	(025) 241-7261	四国支店	(087) 831-3186
中部支社	(052) 722-7601	九州支社	(092) 483-8208
静岡機器サービスステーション	(054) 287-8866	長崎機器サービスステーション	(095) 818-0700
北陸支店	(076) 252-9519		

商標

Ethernetは、富士ゼロックス株式会社の日本における登録商標です。

Microsoft, Microsoft Access, Excel, SQL Server, Visual Basic, Visual C++, Visual Studio, Windows, Windows NT, Windows Server, Windows Vista, およびWindows XP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。

本文中における会社名、システム名、製品名などは、一般に各社の登録商標または商標です。

本文中で、商標記号(™, ®)は明記していない場合があります。

三菱電機株式会社 〒100-8310 東京都千代田区丸の内2-7-3 (東京ビル)

仕様・機能に関するお問い合わせ

製品ごとにお問い合わせを受け付けております。

●電話技術相談窓口 受付時間※1 月曜～金曜 9:00～19:00、土曜・日曜・祝日 9:00～17:00

対象機種		電話番号
自動窓口案内		052-712-2444
エッジコンピューティング製品	産業用PC MELIPC (MI5000/2000/1000)	052-712-2370※2
	Edgecross対応ソフトウェア (MTConnectデータコレクタを除く)	
シーケンサ	MELSEC iQ-R/Q/L/QnAS/AnSシーケンサ一般	052-711-5111
	MELSEC iQ-F/FXシーケンサ全般	052-725-2271※3
	ネットワークユニット/シリアルコミュニケーションユニット	052-712-2578
	MELSOFT シーケンサプログラミングツール	052-711-0037
	MELSOFT 統合エンジニアリング環境	052-799-3591※2
	iQ Sensor Solution	
	MELSOFT 通信支援ソフトウェアツール	MELSOFT MXシリーズ
	MELSEC/パソコンボード	052-712-2370※2
	C言語コントローラ	
	MESインタフェースユニット/高速データロガーユニット	052-799-3592※2
	MELSEC計装/iQ-R/Q二重化	プロセスCPU/二重化CPU (MELSEC-Qシリーズ)
		プロセスCPU/二重化機能 SIL2プロセスCPU (MELSEC iQ-Rシリーズ)
		MELSOFT PXシリーズ
	MELSEC Safety	安全シーケンサ (MELSEC iQ-R/QSシリーズ)
		安全コントローラ (MELSEC-WSシリーズ)
	電力計測ユニット/絶縁監視ユニット	QEシリーズ/REシリーズ
FAセンサ MELSENSOR		レーザ変位センサ
GOT表示器		ビジョンセンサ
SCADA MC Works64		GOT2000/1000シリーズ
		MELSOFT GTシリーズ
		052-712-2417
		052-712-2962※2※6
サーボ/位置決めユニット/シンプルモーションユニット/ モーションコントローラ/センシングユニット/ 組込み型サーボシステムコントローラ		MELSERVOシリーズ
		位置決めユニット (MELSEC iQ-R/Q/L/AnSシリーズ)
		シンプルモーションユニット (MELSEC iQ-R/iQ-F/Q/Lシリーズ)
		モーションCPU (MELSEC iQ-R/Q/AnSシリーズ)
		センシングユニット (MR-MTシリーズ)
		シンプルモーションボード
		C言語コントローラインタフェースユニット (Q173SCCF)/ポジジョンボード
センサレスサーボ		MELSOFT MTシリーズ/MRシリーズ/EMシリーズ
インバータ		FR-E700EX/MM-GKR
三相モータ		FREQROLシリーズ
産業用ロボット		三相モータ225フレーム以下
電磁クラッチ・ブレーキ/テンションコントローラ		MELFAシリーズ
データ収集アナライザ		052-722-2182
低圧開閉器		052-722-2182
低圧遮断器		0536-25-0900※2※4
電力管理用計器		052-721-0100
省エネ支援機器		052-712-5430※5
小容量UPS (5kVA以下)		052-712-5440※5
		052-719-4170
		052-719-4559
		052-719-4556
		052-719-4557※2※3
		052-799-9489※2※6

お問い合わせの際には、今一度電話番号をお確かめの上、お掛け間違いのないようお願い致します。

※1：春季・夏季・年末年始の休日を除く ※2：土曜・日曜・祝日を除く ※3：金曜は17:00まで ※4：月曜～木曜の9:00～17:00と金曜の9:00～16:30

※5：受付時間9:00～17:00 (土曜・日曜・祝日・当社休日を除く) ※6：月曜～金曜の9:00～17:00

●FAX技術相談窓口 受付時間 月曜～金曜 9:00～16:00 (祝日・当社休日を除く)

対象機種		FAX番号
電力計測ユニット/絶縁監視ユニット (QEシリーズ/REシリーズ)		084-926-8340
三相モータ225フレーム以下		0536-25-1258※7
低圧開閉器		0574-61-1955
低圧遮断器		084-926-8280
電力管理用計器/省エネ支援機器/小容量UPS (5kVA以下)		084-926-8340

三菱電機FAサイトの「仕様・機能に関するお問い合わせ」もご利用ください。

※7：月曜～木曜の9:00～17:00と金曜の9:00～16:30 (祝日・当社休日を除く)

三菱電機 FA

検索

www.MitsubishiElectric.co.jp/fa

メンバー
登録無料!

インターネットによる情報サービス「三菱電機FAサイト」

三菱電機FAサイトでは、製品や事例などの技術情報に加え、トレーニングスクール情報や各種お問い合わせ窓口をご提供しています。また、メンバー登録いただくとマニュアルやCADデータ等のダウンロード、eラーニングなどの各種サービスをご利用いただけます。

本マニュアルは、輸出する場合、経済産業省への役務取引許可申請は不要です。

IB(名)-0300370-D(1811)MEE

形名: RMT-P-GCD

2018年11月作成
標準価格 3,000円

本マニュアルは、お断りなしに仕様を変更することがありますのでご了承ください。

この標準価格には消費税は含まれておりません。ご購入の際には消費税が付加されますのでご承知置き願います。