**NAME**

motion − accepts NML motion commands, interacts with HAL in realtime

**SYNOPSIS**

**loadrt motmod [base\_period\_nsec=***period***] [base\_thread\_fp=***0 or 1***] [servo\_period\_nsec=***period***][traj\_period\_nsec=***period***] [num\_joints=***[1-9]***] [num\_dio=***[1-64]***] [num\_aio=***[1-64]***] [num\_spindles=***[1-8]***] [unlock\_joints\_mask=***jointmask***]**

次の項目の制限は、コンパイル時の設定です。

Number of joints available (num\_joints) is set by EMCMOT\_MAX\_JOINTS.

Maximum number of digital inputs (num\_dio) is set by EMCMOT\_MAX\_DIO.

Maximum number of analog inputs (num\_aio) is set by EMCMOT\_MAX\_AIO.

Maximum number of spindles (num\_spindles) is set by EMCMOT\_MAX\_SPINDLES

**DESCRIPTION**

デフォルトでは、ベーススレッドは浮動小数点をサポートしていません。 ソフトウェアステッピング、ソフトウェアエンコーダカウント、およびソフトウェアpwmは浮動小数点を使用しません。 base\_thread\_fpを使用して、ベーススレッドの浮動小数点を有効にすることができます（たとえば、ブラシレスDCモーター制御の場合）。

　これらのピンとパラメーターは、リアルタイムmotmodモジュールによって作成されます。 このモジュールは、LinuxCNCのモーションプランナーにHALインターフェースを提供します。 基本的に、motmodはウェイポイントのリストを取り込んで、モータードライブに供給されるジョイント位置の適切にブレンドされた制約制限付きストリームを生成します。

　オプションで、デジタルI / Oの数はnum\_dioで設定されます。 アナログI / Oの数はnum\_aioで設定されます。デフォルトはそれぞれ4です。

　「joint」または「axis」で始まるピン名は、モーションコントローラ機能によって読み取られ、更新されます。

**MOTION PINS**

**motion-command-handler.time** OUT S32

モーションモジュールmotion-command-handlerの時間（CPUクロック単位）

**motion-controller.time** OUT S32

モーションモジュールモーションコントローラの時間（CPUクロック単位）

**motion.adaptive−feed** IN FLOAT

M52 P1でアダプティブフィードを有効にすると、指令速度にこの値が乗算されます。この効果は、NMLレベルのフィードオーバーライド値とmotion.feed-holdで乗算されます。 負の値は有効であり、Gコードパスを逆に実行します。

**motion.analog−in−***NN* IN FLOAT

これらのピンは、M66Ennの入力待機モードで使用されます。

**motion.analog−out−***NN* OUT FLOAT

これらのピンはM67-68で使用されます。

**motion.coord−error** OUT BIT

ソフト制限を超えるなど、モーションでエラーが発生した場合はTRUE

**motion.coord−mode** OUT BIT

モーションが「テレオプモード」ではなく「協調モード」の場合はTRUE

**motion.current−vel** OUT FLOAT

現在のデカルト速度

**motion.digital−in−***NN* IN BIT

これらのピンは、M66Pnn入力待機モードで使用されます。

**motion.digital−out−***NN* OUT BIT

これらのピンは、M62〜M65ワードによって制御されます。

**motion.distance−to−go** OUT FLOAT

現在の移動で残っている距離

**motion.enable** IN BIT

このビットがFALSEに駆動されると、モーションが停止し、マシンは「マシンオフ」状態になり、オペレーターにメッセージが表示されます。 通常の動作の場合、このビットをTRUEに駆動します。

**motion.eoffset-active** OUT BIT

外部オフセットがアクティブ（ゼロ以外）であることを示します

**motion.eoffset-limited** OUT BIT

外部オフセットのあるモーションがソフト制限制約（[AXIS\_L] MIN\_LIMIT、MAX\_LIMIT）によって制限されたことを示します。

**motion.feed−hold** IN BIT

M53 P1で送り停止制御が有効になっていて、このビットがTRUEの場合、送り速度は0に設定されます。

注：フィードホールドは、ジョグではなく、gcodeコマンドに適用されます。

**motion.feed−inhibit** IN BIT

このピンがTRUEの場合、gcodeコマンドのマシンモーションは禁止されます。

このピンがTRUEになったときに機械が主軸同期移動を実行している場合、主軸同期動作は終了し、それ以降の移動は禁止されます（これは、機械、工具、またはワークの損傷を防ぐためです）。

このピンがTRUEになったときにマシンが（スピンドル同期されていない）移動の途中にある場合、マシンは最大許容加速速度で停止するまで減速します。

このピンがFALSEになると、モーションが再開されます。

注：feed-inhibitは、ジョグではなく、gcodeコマンドに適用されます。

**motion.homing−inhibit** IN BIT

このビットがTRUEの場合、ジョイントホーミング移動（「ホームオール」を含む）の開始は許可されず、エラーが報告されます。 デフォルトでは、モーションが有効になっている場合は常に、ジョイントモードでホーミングが許可されます。

**motion.in−position** OUT BIT

マシンが所定の位置にある場合（つまり、現在コマンド位置に向かって移動していない場合）はTRUE。

**motion.motion−enabled** OUT BIT

**motion.motion−type** OUT S32

これらの値は、src / emc / nml\_intf /motion\_types.hからのものです。

0: Idle (no motion)

1: Traverse

2: Linear feed

3: Arc feed

4: Tool change

5: Probing

6: Rotary unlock for traverse

**motion.on−soft−limit** OUT BIT

**motion.probe−input** IN BIT

G38.nは、このピンの値を使用して、プローブがいつ接触したかを判断します。 プローブ接点が閉じている（接触している）場合はTRUE、プローブ接点が開いている場合はFALSE。

**motion.program−line** OUT S32

実行中の現在のプログラム行。 実行されていない場合、またはシングルステップ中に行間でゼロ。

**motion.requested−vel** OUT FLOAT

現在要求されている速度（ユーザー単位/秒）。 この値は、GコードファイルのFワード設定であり、マシンの速度と加速の制限に対応するために削減される可能性があります。 このピンの値は、フィードオーバーライドまたはその他の調整を反映していません。

**motion.servo.last−period** OUT U32

サーボスレッドの呼び出し間のCPUクロックの数。 通常、この数値をCPU速度で割ると、時間が秒単位で示され、リアルタイムモーションコントローラーがタイミング制約を満たしているかどうかを判断するために使用できます。

**motion.teleop−mode** OUT BIT

モーションモードはteleopです（軸座標ジョギングが利用可能）。

**motion.tooloffset.L** OUT FLOAT

各軸の現在の工具オフセット（Lは軸の文字、次のいずれか：x y z a b c u v w）

**motion.tp−reverse** OUT BIT

軌道計画が逆になります（逆実行）

**AXIS PINS**

(**L** is the axis letter, one of: **x y z a b c u v w**)

**axis.***L***.eoffset** OUT FLOAT

現在の外部オフセット。

**axis.***L***.eoffset-clear** IN BIT

外部オフセット要求のクリア

**axis.***L***.eoffset-counts** IN S32

外部オフセットの入力をカウントします。 eoffset-countsは内部レジスタに転送されます。 適用される外部オフセットは、レジスタカウントとeoffset-scale値の積です。 レジスタは、マシンの起動ごとにゼロにリセットされます。 外部オフセットがアクティブな状態でマシンの電源がオフになっている場合は、再起動する前にeoffset-countsピンをゼロに設定する必要があります。

**axis.***L***.eoffset-enable** IN BIT

外部オフセットを有効にします（[AXIS\_L] OFFSET\_AV\_RATIOのiniファイル設定も必要です）

**axis.***L***.eoffset-request** OUT FLOAT

要求された外部オフセットのデバッグピン。

**axis.***L***.eoffset-scale** IN FLOAT

外部オフセットのスケール。

**axis.***L***.jog−accel−fraction** IN FLOAT

ホイールジョギングの加速度を、軸のinimax\_accelerationの一部に設定します。 1より大きい値または0より小さい値は無視されます。

**axis.***L***.jog−counts** IN S32

物理的なジョグホイールを使用するには、外部エンコーダの「カウント」ピンに接続します。

**axis.***L***.jog−enable** IN BIT

TRUEの場合（および手動モードの場合）、「ジョグカウント」を変更するとモーションが発生します。 falseの場合、「jog-counts」は無視されます。

**axis.***L***.jog−scale** IN FLOAT

「ジョグカウント」のカウントごとに移動距離を機械単位で設定します。

**axis.***L***.jog−vel−mode** IN BIT

FALSE（デフォルト）の場合、ジョグホイールは位置モードで動作します。 軸は、所要時間に関係なく、カウントごとに正確にジョグスケールの単位で移動します。 TRUEの場合、ホイールは速度モードで動作します。つまり、コマンドされたモーションが完了していない場合でも、ホイールが停止するとモーションが停止します。

**axis.***L***.kb−jog−active** OUT BIT

（無料プランナー軸ジョギングアクティブ（キーボードまたはハルイ））

**axis.***L***.pos−cmd** OUT FLOAT

軸が指示した位置。 軸とモーターの座標の間には、バックラッシュ補正、ねじ誤差補正、ホームオフセットなどのオフセットがいくつかある場合があります。 外部オフセットは個別に報告されます（axis.L.eoffset）。

**axis.***L***.teleop−pos−cmd** OUT FLOAT

**axis.***L***.teleop−tp−enable** OUT BIT

この軸で「テレオププランナー」が有効になっている場合はTRUE

**axis.***L***.teleop−vel−cmd** OUT FLOAT

軸の指令速度

**axis.***L***.teleop−vel−lim** OUT FLOAT

レオププランナーの速度制限

**axis.***L***.wheel−jog−active** OUT BIT

**JOINT PINS**

Nはジョイント番号（0 ... num\_joints-1））

（注：（DEBUG）とマークされたピンはデバッグ支援として機能し、いつでも変更または削除される可能性があります。）

**joint.***N***.joint−acc−cmd** OUT FLOAT **(DEBUG)**

関節の命令された加速

**joint.***N***.active** OUT BIT **(DEBUG)**

このジョイントがアクティブな場合はTRUE

**joint.***N***.amp−enable−out** OUT BIT

このジョイントのアンプを有効にする必要がある場合はTRUE

**joint.***N***.amp−fault−in** IN BIT

このジョイントのアンプで外部障害が検出された場合は、TRUEで駆動する必要があります

**joint.***N***.backlash−corr** OUT FLOAT **(DEBUG)**

バックラッシュまたはねじ補正の生の値

**joint.***N***.backlash−filt** OUT FLOAT **(DEBUG)**

バックラッシュまたはねじ補正のフィルター値（モーション制限を尊重）

**joint.***N***.backlash−vel** OUT FLOAT **(DEBUG)**

バックラッシュまたはねじ補正速度

**joint.***N***.coarse−pos−cmd** OUT FLOAT **(DEBUG)**

**joint.***N***.error** OUT BIT **(DEBUG)**

このジョイントでリミットスイッチが閉じるなどのエラーが発生した場合はTRUE

**joint.***N***.f−error** OUT FLOAT **(DEBUG)**

実際の次のエラー

**joint.***N***.f−error−lim** OUT FLOAT **(DEBUG)**

次のエラー制限

**joint.***N***.f−errored** OUT BIT **(DEBUG)**

このジョイントが次のエラー制限を超えた場合はTRUE

**joint.***N***.faulted** OUT BIT **(DEBUG)**

**joint.***N***.free−pos−cmd** OUT FLOAT **(DEBUG)**

「フリープランナー」はこの関節の位置を命じました。

**joint.***N***.free−tp−enable** OUT BIT **(DEBUG)**

このジョイントで「フリープランナー」が有効になっている場合はTRUE

**joint.***N***.free−vel−lim** OUT FLOAT **(DEBUG)**

フリープランナーの速度制限

**joint.***N***.home−state** OUT S32 **(DEBUG)**

ホーミングステートマシンの状態

**joint.***N***.home−sw−in** IN BIT

このジョイントのホームスイッチが閉じている場合は、TRUEで駆動する必要があります

**joint.***N***.homed** OUT BIT **(DEBUG)**

ジョイントがホームになっている場合はTRUE

**joint.***N***.homing** OUT BIT

ジョイントが現在ホーミングしている場合はTRUE

**joint.***N***.in−position** OUT BIT **(DEBUG)**

ジョイントが「フリープランナー」を使用していて停止した場合はTRUE

**joint.***N***.index−enable** IO BIT

インデックスパルスへのホーミングを可能にするには、ジョイントのエンコーダのインデックスイネーブルピンに接続する必要があります

**joint.N.is−unlocked** IN BIT

ジョイントがロック解除されていることを示します（JOINT UNLOCK PINSを参照）。

**joint.***N***.jog−accel−fraction** IN FLOAT

ホイールジョギングの加速度を、ジョイントのinimax\_accelerationの一部に設定します。 1より大きい値または0より小さい値は無視されます。

**joint.***N***.jog−counts** IN S32

物理的なジョグホイールを使用するには、外部エンコーダの「カウント」ピンに接続します。

**joint.***N***.jog−enable** IN BIT

TRUEの場合（および手動モードの場合）、「ジョグカウント」を変更するとモーションが発生します。 falseの場合、「jog-counts」は無視されます。

**joint.***N***.jog−scale** IN FLOAT

「ジョグカウント」のカウントごとに移動距離を機械単位で設定します。

**joint.***N***.jog−vel−mode** IN BIT

FALSE（デフォルト）の場合、ジョグホイールは位置モードで動作します。 ジョイントは、所要時間に関係なく、カウントごとに正確にジョグスケールの単位を移動します。 TRUEの場合、ホイールは速度モードで動作します。つまり、コマンドされたモーションが完了していない場合でも、ホイールが停止するとモーションが停止します。

**joint.***N***.kb−jog−active** OUT BIT **(DEBUG)**

（無料プランナージョイントジョギングアクティブ（キーボードまたはハルイ））

**joint.***N***.motor−offset** OUT FLOAT **(DEBUG)**

ジョイントがホームになっているときに確立されるジョイントモーターオフセット。

**joint.***N***.motor−pos−cmd** OUT FLOAT

このジョイントのコマンド位置。

**joint.***N***.motor−pos−fb** IN FLOAT

このジョイントの実際の位置。

**joint.***N***.neg−hard−limit** OUT BIT **(DEBUG)**

ジョイントの負のハード制限

**joint.***N***.neg−lim−sw−in** IN BIT

このジョイントの負のリミットスイッチが作動した場合は、TRUEで駆動する必要があります。

**joint.***N***.pos−cmd** OUT FLOAT

ジョイント（モーターではなく）のコマンド位置。 ジョイント座標とモーター座標の間には、バックラッシュ補正、ねじエラー補正、ホームオフセットなどのオフセットがいくつかある場合があります。

**joint.***N***.pos−fb** OUT FLOAT

ジョイントフィードバック位置。 この値は、実際のモーター位置からジョイントオフセットを引いたものから計算されます。 機械の視覚化に役立ちます。

**joint.***N***.pos−hard−limit** OUT BIT **(DEBUG)**

ジョイントの正のハード制限

**joint.***N***.pos−lim−sw−in** IN BIT

このジョイントの正のリミットスイッチが作動した場合は、TRUEで駆動する必要があります。

**joint.***N***.unlock** OUT BIT

軸がロックされたジョイント（通常はロータリー）であり、移動がコマンドされている場合はTRUE（JOINT UNLOCK PINSを参照）。

**joint.***N***.joint−vel−cmd** OUT FLOAT **(DEBUG)**

ジョイントの指令速度

**joint.***N***.wheel−jog−active** OUT BIT **(DEBUG)**

**JOINT UNLOCK PINS**

ジョイントのロック解除に使用されるジョイントピン（joint.N.unlock、joint.N.is-unlocked）は、motmodのunlock\_joints\_mask = jointmaskパラメーターに従って作成されます。 これらのピンは、インデクサー（通常はロータリージョイント）をロックするために必要になる場合があります。

ジョイントマスクビットは次のとおりです：（lsb）0：joint0、1：joint1、2：joint2、..。

例：loadrt motmod ... unlock\_joints\_mask = 0x38は、ジョイントのロック解除ピンを作成します3,4,5

**SPINDLE PINS**

（Mはスピンドル番号（0 ... num\_spindles-1））

**spindle.M.amp−fault−in** IN BIT

このスピンドルのアンプで外部障害が検出された場合は、TRUEで駆動する必要があります

**spindle.M.at−speed** IN BIT

次の条件下では、このピンがTRUEになるまでモーションが一時停止します。各スピンドルの開始または速度変更後の最初の送りの移動前。 スピンドル同期移動のすべてのチェーンの開始前。 CSSモードの場合、すべての高速->フィード遷移で。

**spindle.M.brake** OUT BIT

スピンドルブレーキをかける必要がある場合はTRUE

**spindle.M.forward** OUT BIT

スピンドルが前方に回転する必要がある場合はTRUE

**spindle.M.index−enable** I/O BIT

スピンドル同期移動を正しく動作させるには、この信号をスピンドルエンコーダのインデックスイネーブルピンにフックする必要があります。

**spindle.M.inhibit** IN BIT

TRUEの場合、スピンドル速度は0に設定および保持されます。

**spindle.M.is−oriented** IN BIT

スピンドル方向の確認ピン。 オリエントサイクルを完了します。 主軸方向がアサートされたときに主軸方向が真の場合、主軸方向ピンがクリアされ、主軸ロックピンがアサートされます。 また、スピンドルブレーキピンがアサートされます。

**spindle.M.locked** OUT BIT

スピンドルは完全なピンを方向付けます。 M3、M4、M5のいずれかによってクリアされます。

**spindle.M.on** OUT BIT

スピンドルが回転する必要がある場合はTRUE

**spindle.M.orient** OUT BIT

スピンドルオリエントサイクルの開始を示します。 M19によって設定されます。 M3、M4、M5のいずれかによってクリアされます。

スピンドルオリエントがtrueのときにスピンドルオリエントフォールトがゼロでない場合、M19コマンドはエラーメッセージで失敗します。

**spindle.M.orient−angle** OUT FLOAT

M19に必要なスピンドルの向き。 M19Rワードパラメータの値と[RS274NGC] ORIENT\_OFFSETiniパラメータの値。

**spindle.M.orient−fault** IN S32

オリエントサイクルの故障コード入力。 ゼロ以外の値を指定すると、オリエントサイクルが中止されます。

**spindle.M.orient−mode** OUT BIT

必要なスピンドル回転モード。 M19Pパラメータワードを反映します。

**spindle.M.reverse** OUT BIT

スピンドルが後方に回転する必要がある場合はTRUE

**spindle.M.revs** IN FLOAT

主軸同期移動を正しく動作させるには、この信号を主軸エンコーダの位置ピンに接続する必要があります。

**spindle.M.speed−cmd−rps** FLOAT OUT

毎秒回転数の単位での指令スピンドル速度

**spindle.M.speed−in** IN FLOAT

1秒あたりの回転数で表した実際のスピンドル速度フィードバック。 G96（一定の表面速度）およびG95（1回転あたりの送り）モードに使用されます。

**spindle.M.speed−out** OUT FLOAT

1分あたりの回転数での望ましいスピンドル速度

**spindle.M.speed−out−abs** OUT FLOAT

1分あたりの回転数で表した望ましいスピンドル速度。スピンドルの方向に関係なく常に正です。

**spindle.M.speed−out−rps** OUT FLOAT

1秒あたりの回転数で表した望ましいスピンドル速度

**spindle.M.speed−out−rps−abs** OUT FLOAT

スピンドルの方向に関係なく常に正の、1秒あたりの回転数で表した望ましいスピンドル速度。

**MOTION PARAMETERS**

パラメータの多くはデバッグの補助として機能し、いつでも変更または削除される可能性があります。

**motion−command−handler.tmax** RW S32

これらのHAL関数の実行時間に関する情報をCPUクロックで表示します

**motion−command−handler.tmax−increased** RO S32

**motion−controller.tmax** RW S32

これらのHAL関数の実行時間に関する情報をCPUクロックで表示します

**motion−controller.tmax−increased** RO BIT

**motion.debug−***\**

これらの値は、デバッグの目的で使用されます。

**FUNCTIONS**

通常、これらの機能は両方とも、示されている順序でサーボスレッドに追加されます。

**motion−command−handler**

ユーザースペースからのモーションコマンドを処理します。 motion-command-handlerという名前のピン。 時間とパラメータmotion-command-handler.tmax、tmax-increasedがこの関数用に作成されます。

**motion−controller**

LinuxCNCモーションコントローラーを実行します。motion-controller.timeという名前のピンとmotion-controller.tmax、tmax-increasedパラメーターがこの関数用に作成されます。

**BUGS**

このマニュアルページは不完全です

（DEBUG）で分類されたピンの識別は疑わしいです。

**SEE ALSO**

iocontrol(1), milltask(1)