

L6470モータードライバモジュール (秋月)

<http://akizukidenshi.com/catalog/g/gM-06365/>

L6470モータードライバモジュール (ストロベリリナックス)

<https://strawberry-linux.com/catalog/items?code=12023>

ストロベリリナックス製のマニュアル

<https://strawberry-linux.com/pub/l6470-manual.pdf>

データシートのサイト

http://www.st.com/content/st_com/ja/products/motor-drivers/stepper-motor-drivers/l6470.html

データシートの一部翻訳

<http://spinelify.blog.fc2.com/blog-entry-79.html>

下の表は、ストロベリリナックス製のマニュアルからのコピーです、怒られるかも？

レジスタの概要

アドレス	名称	機能	長さ(bit)	初期値	#12026 モータの例
0x01	ABS_POS	モータの絶対位置を保持しています。	22	0	←
0x02	EL_POS	マイクロステップの位置を保持	9	0	←
0x03	MARK	マークしている絶対位置	22	0	←
0x04	SPEED	現在の回転スピード	20	0	←
0x05	ACC	加速係数(大きくすると加速が速くなります)	12	0x8A	←
0x06	DEC	減速係数(大きくすると減速が速くなります)	12	0x8A	←
0x07	MAX_SPEED	最大回転スピード	10	0x41	0x20
0x08	MIN_SPEED	最低回転スピード	13	0	←
0x09	KVAL_HOLD	モータ停止中の電圧設定	8	0x29	0xFF
0x0A	KVAL_RUN	モータ定速回転時の電圧設定	8	0x29	0xFF
0x0B	KVAL_ACC	加速中の電圧設定	8	0x29	0xFF
0x0C	KVAL_DEC	減速中の電圧設定	8	0x29	0xFF
0x0D	INT_SPD	補間を開始する回転数	14	0x408	←
0x0E	ST_SLP	補間の傾き	8	0x19	←
0x0F	FN_SLP_ACC	加速最終時の傾き	8	0x29	←
0x10	FN_SLP_DEC	減速最終時の傾き	8	0x29	←
0x11	K_THERM	高温時の補正	4	0	←
0x12	ADC_OUT	ADCの結果(読み取り専用)	5	—	
0x13	OCD_TH	オーバーカレントの電流スレッシュホールド	4	0x8	0xF
0x14	STALL_TH	ストールの電流スレッシュホールド	7	0x40	0x7F
0x15	FS_SPD	フルステップ駆動に切り替える回転数	10	0x27	←
0x16	STEP_MODE	フルステップ、ハーフステップ、1/4、1/8、 …、1/128ステップの設定	8	0x7	←
0x17	ALARM_EN	アラーム機能のON/OFF	8	0xFF	←
0x18	CONFIG	システム設定レジスタ	16	0x2E88	←
0x19	STATUS	現在のステータス(読み取り専用)	16	—	
0x1A	RESERVED	<予約済み>			
0x1B	RESERVED	<予約済み>			

ABS_POS(0x01) 22bit

ABS_POS レジスタは、選択されたステップモードと一致して、現在のモータの絶対位置が含まれています。

格納された値の単位は、選択されたステップモード（フル、ハーフ、マイクロステップなど）に等しいです。

値は2の補数形式であり、それは $-2^{21} \sim +2^{21}-1$ の範囲です。

電源オンレジスタ「0」（ホームポジション）に初期化されます。

モータが動作している時にレジスタの書き込みをしようとすると、コマンドは無視されます、STATUS レジスタ NOTPERF_CMD フラグが上昇します。

MARK(0x03) 22bit

MARK レジスタは、選択したステップモードに応じ MARK と呼ばれる絶対位置が含まれています。

格納された値の単位は、選択されたステップモード（フル、ハーフ、マイクロステップなど）に等しいです。これは、2の補数形式であり、それは $-2^{21} \sim +2^{21}-1$ の範囲です。

EL_POS(0x02) 9bit

EL_POS レジスタは、モータの電流電氣的位置 (マイクロステップ位置)が含まれています。
2 MSbits は、現在のステップを示し、他のビット (0-6) は、ステップ内 (ステップ /128 で表される) 現在のマイクロステップを示しています。

```
 8 7 6 5 4 3 2 1 0  
[STEP][ MICROSTEP ]
```

EL_POS レジスタがユーザによって書き込まれると、新たな電氣的位置が瞬時に課せられます。
EL_POS レジスタが書き込まれると、その値は一致させるためにマスクされなければなりません
間違ったマイクロ値の生成を回避するためにSTEP_MODE レジスタで選択されたステップ・モードで
(セクション 9.1.19 を参照のこと)。

そうでなければ、得られたマイクロシーケンスが正しくありません。

モータが動作している時にレジスタの書き込みをしようとすると、コマンドは無視されます、
STATUS レジスタ NOTPERF_CMD フラグが上昇します。

SPEED(0x04) 20bit

SPEED レジスタは、step/tick (形式符号なし固定小数点 0.28) で表され、現在のモータ速度が含まれています。

ここで、SPEED は、レジスタに格納された整数であり、1tick は 250 ns です。

範囲は0.015step/s の分解能で 0-104166(15625step/s) のステップです。

モータが動作している時にレジスタの書き込みをしようとすると、コマンドは無視されます、
STATUS レジスタ NOTPERF_CMD フラグが上昇します。

ACC(0x05) 12bit

ACC レジスタは、step/ tick² (形式符号なし固定小数点 0.40) で表した加速度時の速度プロファイルが含まれています。

設定可能な範囲は 14.55 step/s² の解像度で 0(14.55) から 4095(59590step/s²) にあります。

ACC の値が 0xFFF に設定されている場合、デバイスは無限の加減速モードで動作します。
(無限の加減速モード：加速および減速フェーズが完全にスキップされます)

モータが動作している時にレジスタの書き込みをしようとすると、コマンドは無視されます、
STATUS レジスタ NOTPERF_CMD フラグが上昇します。

DEC(0x06) 12bit

DEC レジスタは、step/ tick² (形式符号なし固定小数点 0.40) で表した減度時の速度プロファイルが含まれています。

設定可能な範囲は 14.55 step/s² の解像度で 0(14.55) から 4095(59590step/s²) にあります。

デバイスは無限の加減速モードで動作している場合、この値は無視されます。

モータが動作している時にレジスタの書き込みをしようとすると、コマンドは無視されます、
STATUS レジスタ NOTPERF_CMD フラグが上昇します。

MAX_SPEED(0x07) 10bit

MAX スピードレジスタは、step/ tick (形式符号なし固定小数点 0.18) で表した最大速度時の速度プロファイルが含まれています。速度指定時はこの値を超えてはいけません。

設定可能な範囲は15.25 step/sの解像度で 0(15.25) から 1023(15610step/s) にあります。

MIN_SPEED(0x08) 13bit

MINスピードレジスタは、step/ tick（形式符号なし固定小数点 0.24）で表した最少速度時の速度プロファイルが含まれています。

設定可能な範囲は0.238 step/sの解像度で0 から4102(976.3step/s)にあります。

```
      12      11 10 9 8 7 6 5 4 3 2 1 0  
[LSPD_OPT][  MIN_SPEED      ]
```

LSPD_OPT ビットが” 1”の場合は、低速の最適化機能が有効になり、MIN_SPEED 値の速度閾値以下の速度が指定されるとこの場合、速度プロファイルの最低速度はゼロに設定されます。

モータが動作している時にレジスタの書き込みをしようとすると、コマンドは無視されます、STATUS レジスタ NOTPERF_CMD フラグが上昇します。

FS_SPD(0x15) 10bit

FS_SPD レジスタは、閾値速度が含まれています。

実際の速度がこの値を超えると、ステップモードが自動的にフルステップモードへと二段階に切り替えられます。（マイクロステップで動作中に）

その値は、step/ tick（形式符号なし固定小数点 0.18）で表した値であり、

設定可能な範囲は15.25 step/sの解像度で0(7.63) から 1023(15625step/s) にあります。

FS_SPD 値が 0x3FF（最大）に設定されている場合システムは、常に、マイクロステップモードで動作します。

FS_SPD 値の0は約 7.63 ステップ / 秒の速度閾値に相当します。

KVAL_HOLD(0x09) 8bit

モータ停止中時の供給電圧を設定するレジスタ

KVAL_RUN(0x0A) 8bit

モータ定速動作時の供給電圧を設定するレジスタ

KVAL_ACC(0x0B) 8bit

モータ加速中時の供給電圧を設定するレジスタ

KVAL_DEC(0x0C) 8bit

モータ減速中時の供給電圧を設定するレジスタ

設定可能範囲は 0.004xVs の解像度で0(0)から 255(0.996xVs) にあります。

例えばモータ電源Vs=12VでKVAL_HOLD=255なら 0.996x12=11.952V が停止中時の電圧となります。

INT_SPEED(0x0D) 14bit

INT_SPEED レジスタは BEMF 補償曲線で2段階のスロープ変化を行うが、その変化する速度閾値が含まれています。

設定可能な範囲は0.0596 step/s の解像度で0 から16383(976.5 step/s)にあります。

モータが動作している時にレジスタの書き込みをしようとすると、コマンドは無視されます、STATUS レジスタ NOTPERF_CMD フラグが上昇します。

ST_SLP(0x0E) 8bit

ST_SLP レジスタは、INT_SPEED 速度閾値より低い場合 (スロープの1段目) に使用される BEMF 補償曲線の傾きが含まれています。

利用可能な範囲は、0.000015 step/s の解像度で 0 から 255(0.004 step/s) です。

ST_SLP 、 FN_SLP_ACC と FN_SLP_DEC パラメータがゼロに設定されている場合、何の BEMF 補償は行われません。

モータが動作している時にレジスタの書き込みをしようとすると、コマンドは無視されます、STATUS レジスタ NOTPERF_CMD フラグが上昇します。

FN_SLP_ACC(0x0F) 8bit

FN_SLP_ACC レジスタは、INT_SPEED 速度閾値より高い場合 (スロープの2段目) に使用され、加速中時の速度 BEMF 補償曲線の傾きが含まれています。

利用可能な範囲は、0.000015 step/s の解像度で 0 から 255(0.004 step/s) です。

ST_SLP 、 FN_SLP_ACC と FN_SLP_DEC パラメータがゼロに設定されている場合、何の BEMF 補償は行われません。

モータが動作している時にレジスタの書き込みをしようとすると、コマンドは無視されます、STATUS レジスタ NOTPERF_CMD フラグが上昇します。

FN_SLP_DEC(0x10) 8bit

FN_SLP_DEC レジスタは、INT_SPEED 速度閾値より高い場合 (スロープの2段目) に使用され、減速中時の速度 BEMF 補償曲線の傾きが含まれています。

利用可能な範囲は、0.000015 step/s の解像度で 0 から 255(0.004 step/s) です。

ST_SLP 、 FN_SLP_ACC と FN_SLP_DEC パラメータがゼロに設定されている場合、何の BEMF 補償は行われません。

モータが動作している時にレジスタの書き込みをしようとすると、コマンドは無視されます、STATUS レジスタ NOTPERF_CMD フラグが上昇します。

K_THERM(0x11) 4bit

K_THERM レジスタは巻線抵抗、熱ドリフト補償システムによって使用される値 (巻線抵抗熱ドリフト補償係数) を含んでいます。

利用可能な範囲は、0.03125 の解像度で0(1)から 255(1.46875) です。

ADC_OUT(0x12) 5bit 読み取り専用

ADC_OUT レジスタはアナログ ADCIN ピンの電圧をデジタル変換 ; その結果が含まれている、電源電圧補償が無効になっている場合でも使用可能です。

レジスタの書き込みをしようとすると、コマンドは無視されます、STATUS レジスタ NOTPERF_CMD フラグが上昇します。

OCD_TH(0x13) 4bit

OCD_TH は、レジスタは、過電流閾値をが含まれています。

利用可能な範囲は、375 ミリアンペアの段階で0(375mA)から 15(6 A) です。

この値を超えた場合に STATUS レジスタのOCD フラグがLOW (0) に強制されます。

過電流イベントが発生した場合には MOSFET ブリッジがOFF (ハイインピーダンス) になり、

STATUS レジスタのOCD フラグがHIGH (1) になります。 (34 ページのパラグラフ 9.1.22 を参照)

この過電流イベントは無効にすることもできます。過電流イベントの設定先は CONFIG レジスタの OC_SD ビットです。

STALL_TH(0x14) 7bit

STALL_TH は、レジスタは、ストール検出閾値を含んでいます。

利用可能な範囲は、31.25 ミリアンペアの段階で0(31.25mA)から 127(4 A) です。

モータ速度と負荷角特性に応じて、L6470 は、プログラマブル電流コンパレータを用いてモータ失速状態の検出を提供しています。

ストールイベントが発生すると、(セクション 9.2.20 を参照)、それぞれのフラグ (STEP_LOSS_A またはSTEP_LOSS_B) はシステムリセットが発生するか GetStaus コマンドまでローに強制します。

STEP_MODE(0x16) 8bit

STEP_MODE レジスタは、以下の構造を有しています

7 6 5 4 3 2 1 0
[SYNC_EN] [SYNC_SEL] [0] [STEP_SEL]

STEP_SEL[2 : 0]

パラメータは、8 つの可能なステップモードのいずれかを選択します。

0:Full-step 1:Half-step 2:1/4 microstep 3:1/8 microstep 4:1/16 microstep 5:1/32 microstep
6:1/64 microstep 7:1/128 microstep

ステップモードが変更されるたびに、電気的位置は、最初のマイクロにリセットされます。

STEP_SEL が変更され、ABS_POS レジスタの値は意味を失い、リセットされるべきです。

モータが動作している時にレジスタの書き込みをしようとすると、コマンドは無視されます、

STATUS レジスタ NOTPERF_CMD フラグが上昇します。

SYNC_EN[7]

SYNC_EN ビットが” 1”に設定されると、BUSY/SYNC 出力はSYNC_SEL パラメータに従って信号を提供します。

SYNC_EN ビットが” 0”設定されている場合は、BUSY/SYNC 出力は、コマンド実行時に強制的にローにされます、

SYNC_SEL[6 : 4]

信号は EL_POS のポジションに同期します、同期周波数はSYNC_SEL の選択と STEP_SEL により選ばれます。

ALARM_EN(0x17) 8bit

ALARM_EN レジスタは、アラーム信号の ON/OFF 選択を可能にします。

ALARM_EN レジスタの各ビットがハイに設定されている場合に、アラーム状態が発生すれば

FLAG 端子の出力を LOW に強制します。

0bit : 過電流検出

1bit : サーマルシャットダウン検出

2bit : 熱警告検出

3bit : 定電圧検出

4bit : A相ストール検出

5bit : B相ストール検出

6bit : スイッチイベント検出

7bit : 間違った、または非実行可能コマンドの発行

CONFIG(0x18) 16bit

CONFIG レジスタの構造は次のとおりです。

15	14	13	12	11	10	9	8	7	6	5	4	3
[F_PWM_INT]	[F_PWM_DEC]	[POW_SR]	[OC_SD]	[]	[EN_VSCOMP]	[SW_MODE]	[EXT_CLK]					
2	1	0										
[OSC_SEL]												

OSC_SEL[2 : 0] / EXT_CLK[3]

OSC_SEL とEXT_CLKビットは、システムクロックソースを設定します。
デフォルトは内部発信器の16MHz (2MHzの OSCOUT ピン出力有り)

SW_MODE[4]

“1”なら、SW ピンのスイッチを押せばモータが停止する (ハード停止モード)。
“0”なら、ユーザにお任せモード (GoUntil/ReleaseSW コマンドで使用する)

EN_VSCOMP[5]

EN_VSCOMPビットは、モータの電源電圧補償を有効にするかどうかを設定します。
“1”で有効 ” 0”で無効 (ADCIN/ADC_OUT はユーザにお任せ)

OC_SD[7]

“1”なら、過電流発生時にブリッジ回路をシャットダウンする様に設定
“0”なら、過電流発生してもブリッジ回路はシャットダウンしない様に設定

POW_SR[9 : 8]

モータ出力のスルーレート値の設定
320・75・110・260V/us の設定が出来ます。
(デフォルトは110V/us?)(読み出したら 260V/us だった)

F_PWM_INT[15 : 13] / F_PWM_DEC[12 : 10]

PWM出力周波数の設定を行う。(デフォルトは15.6KHz)

STATUS(0x19) 16bit 読み取り専用

STATUS レジスタの構造は次のとおりです。

15	14	13	12	11	10	9	8
[SCK_MOD]	[STEP_LOSS_B]	[STEP_LOSS_A]	[OCD]	[TH_SD]	[TH_WRN]	[UVLO]	[WRONG_CMD]
7	6	5	4	3	2	1	0
[NOTPERF_CMD]	[MOT_STATUS]	[DIR]	[SW_EVN]	[SW_F]	[BUSY]	[HiZ]	

HiZ[0]

HiZが” 1” 場合には、ブリッジがハイインピーダンス状態であることを示しています。任意のモーションコマンドは、ハイインピーダンス状態（ハードストップが含まれる）からデバイスは終了になります。ハイインピーダンス状態を強制的にエラーフラグがない限り有効です

BUSY[1]

BUSYですが、BUSYピンの状態を反映しています。一定速度コマンド、絶対位置指定コマンド、移動コマンドが実行中の時、ピン出力がLOW（0）になります。コマンドが実行されたとき（目標速度、目標座標に到達）、この BUSY ピンは解放されます。

SW_F[2]

SW_F フラグはSWピンの入力状態を報告します。

SW_EVN[3]

SW_EVN フラグはスイッチのターンオンイベント（立下りエッジ）を検出すると HIGH（1）になる。このフラグはGetStatusコマンドでフラグは解除される。

DIR[4]

現在のモーターの回転方向を示しています。
0：逆転 1：正転

MOT_STATUS[6 : 5]

MOT_STATUS は、現在のモータの状態を示します。
0：停止 1：加速中 2：減速中 3：定速度

NOTPERF_CMD[7]

“1”で有れば、コマンドが実行できない。

WRONG_CMD[8]

“1”で有れば、コマンドが全く存在しない。

UVLO[9]

UVLOフラグは電源投入時およびロジックリセット時にも強制的に LOW(0) になります。モータ電源電圧が Vsthoff 電圧を下回ると、STATUS レジスタの UVLO フラグが強制的に LOW(0) になります。このフラグはGetStatusコマンドでフラグは解除される。

TH_WRN[10]

熱警告閾値温度に達すると、以下に温度が低下するまで、およびStatusコマンドが送信されて来るまで、STATUSレジスタのTH_WRNビットがLOW(0)になります。

TH_SD[11]

サーマルシャットダウン閾値に達すると、デバイスはシャットダウン状態になります。

シャットダウン状態では、STATUSレジスタのTH_SDビットがLOW(0)になり、パワーブリッジがOFF（ハイインピーダンス）になり、STATUSレジスタのHIZビットがHIGH(1)になる。

OCD[12]

電流が過電流閾値（OCD_TH）を超えた場合にSTATUSレジスタのOCDビットがLOW(0)になる。デバイスはシャットダウン状態になり、パワーブリッジがOFF（ハイインピーダンス）になり、STATUSレジスタのHIZビットがHIGH(1)になります。

STEP_LOSS_A[13]/STEP_LOSS_B[14]

ストールイベントが発生すると、GetStausコマンドが送信されるか、システムリセットされるまで、各フラグ（STEP_LOSS_A又はSTEP_LOSS_B）がLOW(0)になる。

SCK_MOD[15]

STCK入力ピンを介してデバイスが、ステップ・クロック・モードで動作していることを示すハイアクティブフラグです。

このレジスタに書き込みをしようとする、コマンドは無視されます、STATUSレジスタNOTPERF_CMDフラグが上昇します。

コマンドの概要

SPIのデータは 8bit(1byte) 単位に送る、例えば 20bit なら上位4bitは0で埋めて 3byte 送る事になる。

NOP ()

何も実行しない。

```
 7 6 5   4   3   2 1   0  
[ 0 0 0 ][ 0 ][ 0 ][ 0 0 ][ 0 ]
```

SetParam (PARAM, VALUE)

レジスタにデータを書き込む。

```
 7 6 5   4 3 2 1 0  
[ 0 0 0 ][ PARAM ] [VALUE Byte 2][ VALUE Byte 1][ VALUE Byte 0]  
PARAM   書き込み先のレジスタのアドレス (0x01-0x1B)。  
VALUE   レジスタにより 1~3バイトのデータ値を送る。
```

一部のレジスタは、特定の条件下で書くことができます

条件が満たされないときに、これらのレジスタの1つを記述しようとする、最後の引数バイトの終わりに上昇し、無視されるコマンドと NOTPERF_CMD フラグを引き起こし

GetParam (PARAM,)

レジスタからデータを読み込む。

```
 7 6 5   4 3 2 1 0  
[ 0 0 1 ][ PARAM ] [0x00][0x00][0x00]  
PARAM   読み込み先のレジスタのアドレス (0x01-0x1B)。  
応答   レジスタにより 1~3バイトのデータ値がかえされる。
```

すべてのレジスタは、いつでも読むことができます。

Run (DIR, SPD)

モータ回転指示をおこなう。(一定速度コマンド)

プログラムされた最低速度、加速度、減速度に従い、目標速度に到達するための動きを実行します。

```
 7 6 5   4   3   2 1   0       23 22 21 20   19 - 0  
[ 0 1 0 ][ 1 ][ 0 ][ 0 0 ][ DIR ] [ 0 0 0 0   SPD ]  
DIR   モータの回転方向を設定する。 0 : 逆転 1 : 正転  
SPD   モータの目標速度を設定する。(20bit で指定 SPEED レジスタを参照)  
       MAX_SPEED より低く、MIN_SPEED 大きくなければなりません。
```

目標速度に到達するまで、このコマンドを実行すると、BUSYフラグをLowに保持します。

このコマンドは、いつでも与えることができ、すぐに実行されます。

StepClock (DIR)

モータの動作をステップ・クロック・モードにする。

回転指示は、STCK入力ピンを介してクロックを入力する事で行う。

7 6 5 4 3 2 1 0

[0 1 0] [1] [1] [0 0] [DIR]

DIR モータの回転方向を設定する。 0 : 逆転 1 : 正転

ステップ信号の立ち上がりエッジで、モーターがプログラムされた方向に1マイクロステップ移動し、絶対位置が結果的に更新されます。

デバイスは、ステップクロックモードである場合、STATUSレジスタのSCK_MODフラグが上昇し、常に考慮されているモーターは停止します。

一定の速度、絶対位置やモーションコマンドは、SPIを介して送信された場合、デバイスは、ステップ・クロック・モードから抜けません。

ブリッジOFFイベントは、ステップ・クロック・モードを残すために、ハイインピーダンス状態（過熱、過電流など）になります。

このモード実行時は、SPEEDレジスタの値は（常に）0になり、STCKピンにどのようなstep信号が入力されても、モータは停止していると判断される。（STATUSレジスタのMOT_STATUSパラメータは00になる）

Move (DIR,N_STEP)

モータを指定ステップ数まで回転させる。（移動コマンド）

プログラムされた最低速度、加速度、減速度に従い、目標に到達するための動きを実行します。

7 6 5 4 3 2 1 0 23 22 21 - 0

[0 1 0] [0] [0] [0 0] [DIR] [0 0 N_STEP]

DIR モータの回転方向を設定する。 0 : 逆転 1 : 正転

N_STEP ステップの目標数設定する。(22bit)

ステッピングモードにより目標ステップ数の単位値（回転角度）は変わります。

ステップの目標数が実行されるまで、このコマンドはBUSYフラグをLowに保持します。

モータが停止している場合、このコマンドは、実行することができます。

GoTo(ABS_POS)

モータを絶対位置指示で回転（最短経路）させる。（絶対位置指定コマンド）

目標位置を与えると、現在の位置から最短方向に目標位置まで回転します。

プログラムされた最低 / 最高速度、加速度、減速度に従い、目標に到達するための動きを実行します。

7 6 5 4 3 2 1 0 23 22 21 - 0

[0 1 1] [0] [0] [0 0] [0] [0 0 ABS_POS]

ABS_POS 目標位置を設定する。(22bit $-2^{21} \sim +2^{21}-1$)

目標位置に到達するまでこのコマンドはBUSYフラグをLowに保持します。

このコマンドは、以前のモーションコマンドは（BUSYフラグが解除）完了した場合にのみ与えることができます。

GoTo_DIR(DIR,ABS_POS)

モータを絶対位置指示で回転させる。(絶対位置指定コマンド)

目標位置を与えると、現在の位置から指定方向に目標位置まで回転します。

プログラムされた最低 / 最高速度、加速度、減速度に従い、目標に到達するための動きを実行します。

```
7 6 5   4   3   2 1   0           23 22 21 - 0  
[ 0 1 1 ][ 0 ][ 1 ][ 0 0 ][ DIR ] [ 0 0 ABS_POS ]
```

DIR モータの回転方向を設定する。 0 : 逆転 1 : 正転

ABS_POS 目標位置を設定する。(22bit $-2^{21} \sim +2^{21}-1$)

目標位置に到達するまでこのコマンドはBUSYフラグをLowに保持します。

このコマンドは、以前のモーションコマンドは (BUSYフラグが解除) 完了した場合にのみ与えることができます。

GoHome

モータを現在の位置から最短方向でホームポジションまで回転させます。(絶対位置指定コマンド)

```
7 6 5   4   3   2 1   0  
[ 0 1 1 ][ 1 ][ 0 ][ 0 0 ][ 0 ]
```

このコマンドは “GoTo(0)” コマンドと等価であることに注意してください。

モータの回転方向が必須の場合、GoTo_DIR コマンドを使用する必要があります。

ホームポジションに到達するまでGOHOMEコマンドはBUSYフラグをLowに保持します。

GoMark

モータを現在の位置から最短方向で MARK 位置まで回転させます。(絶対位置指定コマンド)

```
7 6 5   4   3   2 1   0  
[ 0 1 1 ][ 1 ][ 1 ][ 0 0 ][ 0 ]
```

GoUntil(ACT,DIR, SPD)

モータの位置を初期化 (原点復帰) する為に回転させる。(一定速度コマンド)

```
7 6 5   4   3   2 1   0           23 22 21 20   19 - 0  
[ 1 0 0 ][ 0 ][ ACT ][ 0 1 ][ DIR ] [ 0 0 0 0   SPD ]
```

ACT スイッチの立ち下がリエッジ検出時の動作を指示する

0 : ABS_POS レジスタの値を 0 (ホームポジション) に設定し、モータは減速して停止します。

1 : ABS_POS レジスタの値を MARK レジスタにコピーし、モータは減速して停止します。

DIR モータの回転方向を設定する。 0 : 逆転 1 : 正転

SPD モータの目標速度を設定する。(20bit で指定 SPEED レジスタを参照)

MAX_SPEED より低く、MIN_SPEED 大きくなければなりません。

Config レジスタのSW_MODE ビットが '0' に設定されていた場合は、モータは減速せずに即時停止します。

このコマンド実行は、スイッチの入カイベントが発生するとモータが停止するまで、BUSYフラグをLowに保持します。

このコマンドは、いつでも与えることができ、すぐに実行されます。

ReleaseSW(ACT,DIR)

モータの位置を初期化（原点復帰）する為に回転させる。（一定速度コマンド）

モータの速度は、プログラムされた最低速度で回転させます。

7 6 5 4 3 2 1 0 23 22 21 20 19 - 0

[1 0 0][1][ACT][0 1][DIR] [0 0 0 0 SPD]

ACT スwitchの立ち上がりエッジ検出時の動作を指示する

0 : ABS_POSレジスタの値を 0（ホームポジション）に設定し、モータは即時停止します。

1 : ABS_POSレジスタの値を MARK レジスタにコピーし、モータは即時停止します。

DIR モータの回転方向を設定する。 0 : 逆転 1 : 正転

もし、プログラムされた最低速度が5step/sより低い場合は、モータは5step/sで駆動します。スイッチ入力が解除され、モータが停止するまでリリースコマンドは、BUSYフラグをLowに保持します。

ResetPos

ABS_POSレジスタを0で初期化し、ホームポジションとする。

7 6 5 4 3 2 1 0

[1 1 0][1][1][0 0][0]

ResetDevice

デバイスをリセットして、電源投入時の状態にします。（6. 1章を参照）

7 6 5 4 3 2 1 0

[1 1 0][0][0][0 0][0]

GetStatus

STATUSレジスタの内容を返します。

7 6 5 4 3 2 1 0

[1 1 0][1][0][0 0][0] [0x00][0x00]

GetStatusコマンドは、STATUSレジスタの警告フラグをリセットします。

コマンドは、すべてのエラー状態から復帰するためのシステムを強制します。

GetStatusコマンドはHizフラグをリセットしません。

SoftStop

減速しながらモータを停止させます。（ストップコマンド）

プログラムされた最低速度に到達するまでモータがプログラムされた減速度で減速させ、

その後、回転子位置を維持したままモータを停止させます。（保持トルクあり）

7 6 5 4 3 2 1 0

[1 0 1][1][0][0 0][0]

このコマンドは、いつでも与えることができ、すぐに実行されます。

モータが高インピーダンス状態にある場合、このコマンドは、高インピーダンス状態から復帰するためにブリッジを強制します。

モーターが停止されるまで、このコマンドはBUSYフラグをLowに保持します。

HardStop

即時にモータを停止させます。(ストップコマンド)

モータが瞬時に減速制約を無視して回転子位置を維持したままモータを停止させます。

(保持トルクあり)

```
7 6 5   4   3   2 1   0  
[ 1 0 1 ][ 1 ][ 1 ][ 0 0 ][ 0 ]
```

このコマンドは、いつでも与えることができ、すぐに実行されます。

モータが高インピーダンス状態にある場合、このコマンドは、高インピーダンス状態から復帰するためにブリッジを強制します。

モーターが停止されるまで、このコマンドはBUSYフラグをLowに保持します。

SoftHiz

減速しながらモータを停止させます。(ストップコマンド)

プログラムされた最低速度に到達するまでモータがプログラムされた減速度で減速させ、その後、高インピーダンス状態にさせます。(保持トルクなし)

```
7 6 5   4   3   2 1   0  
[ 1 0 1 ][ 0 ][ 0 ][ 0 0 ][ 0 ]
```

このコマンドは、いつでも与えることができ、すぐに実行されます。

モーターが停止されるまで、このコマンドはBUSYフラグをLowに保持します。

HardHiz

即時にモータを停止させます。(ストップコマンド)

モータが瞬時に減速制約を無視して停止させます、その後、高インピーダンス状態にさせます。

(保持トルクなし)

```
7 6 5   4   3   2 1   0  
[ 1 0 1 ][ 0 ][ 1 ][ 0 0 ][ 0 ]
```

このコマンドは、いつでも与えることができ、すぐに実行されます。

モーターが停止されるまで、このコマンドはBUSYフラグをLowに保持します。