

製品不具合情報

第 2 版 ----- 2.2(1)を追加

株式会社ノヴァ電子
技術部

対称製品: モーションコントロール IC MCX302, MCX312, MCX304
モーションコントロールボード MC8080P, MC8082P

MCX305, MCX314As では本不具合は起きません。

1. 不具合の現象

S字加減速の定量パルスドライブにおいて、ドライブ終了間際に下記の①から④のいずれかの動作が行なわれた場合、パラメータの設定する値によっては、連続してパルスを出し続ける現象が起きる場合があります。

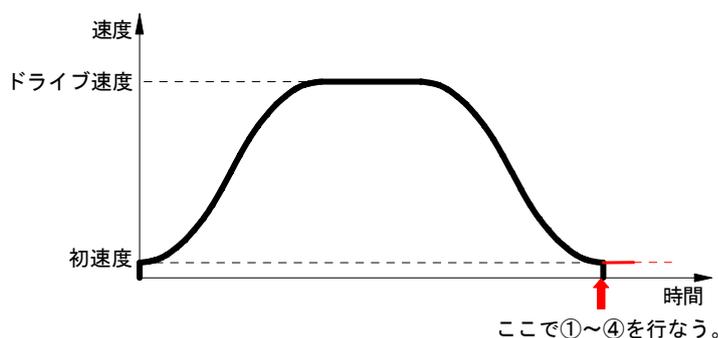


図 1 S字加減速 定量パルスドライブの速度波形

- ① ドライブ終了間際に減速停止命令(26h)を発行したとき。
- ② ハードウェアリミット (nLMTP/M 信号) の停止モードを減速停止に設定し(WR2/D2=1)、ドライブを開始させ、ドライブ終了間際に進行方向のリミットがアクティブになったとき。
- ③ ソフトウェアリミットを有効にし(WR2/D0,1=1)、ドライブを開始させ、ドライブ終了間際に進行方向のソフトウェアリミットがアクティブになったとき。
- ④ nSTOP(2~0)信号 (MCX314As は nIN(2~0)) を有効にし(WR1/D5,3,1)、定量パルスドライブを開始させ、ドライブ終了間際にこれらの信号がアクティブになったとき。

- ・ 台形(直線)加減速ドライブ、定速ドライブでは、本不具合は起きません。
- ・ S字加減速の連続パルスドライブでは、本不具合は起きません。
- ・ 即停止命令、EMGN 信号、即停止モードの LMT 信号、ALARM 信号では、本不具合は起きません。

S字加減速の定量パルスドライブは、ドライブが終了する時、すなわち出力パルスを出し終えるときに、現在速度が初速度に到達するように、また現在加速度が0に到達するようにしています。しかしながら演算精度の問題からすべてのパラメータの組み合わせにおいて、速度＝初速度、加速度＝0にすることができません。この不具合現象は、図2に示すように現在速度が初速度に到達していないで加速度が0の到達してしまった時に、たまたま上記の①から④のいずれかの減速停止要因が働くと発生します。

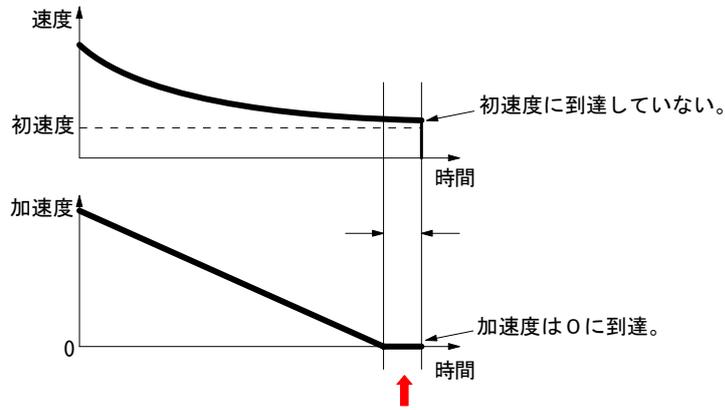


図2 ドライブ終了間際の速度と加速度

ICのRR1レジスタからは、加速中(ASND)、定速中(CNST)、減速中(DSND)の加減速状態を読み取ることができますが、この時の加減速状態は下の図3ようになります。

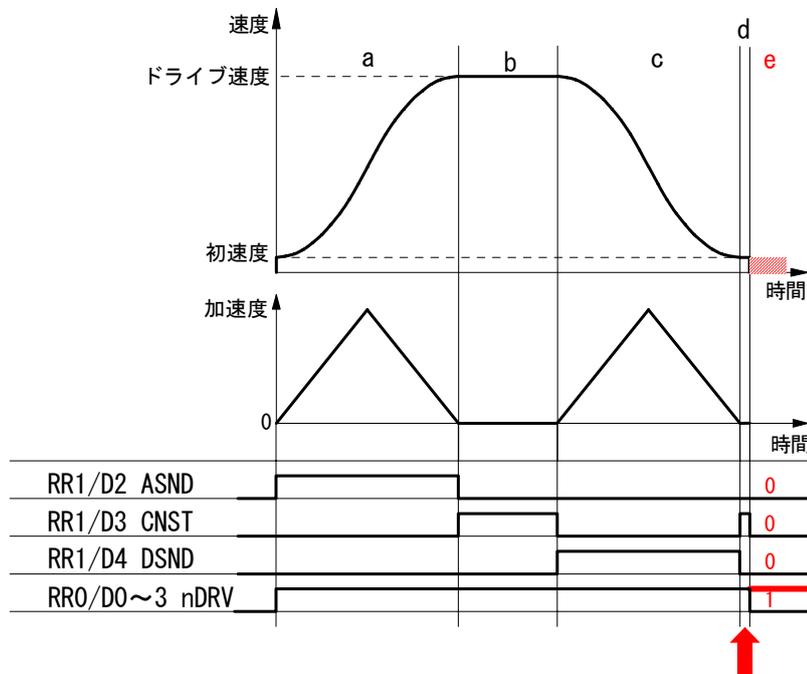


図3 RR1レジスタが示す加減速状態

減速停止要求が入ると不具合が発生する可能性がある区間は図3中dの区間で、この時の加減速状態は定速(CNST=1)を示しています。また不具合(連続してパルスを出す)が発生した場合には、ドライブ中(RR0/nDRV=1)にもかかわらず、ASND,CNST,DSNDはいずれも0となります。

2. 回避方法

2.1 減速停止命令(26h)を発行する場合

基本的には減速が始まったら、減速停止をさせる必要がないわけですから、減速停止命令(26h)の発行を禁止するようにします。減速中であることを知るには通常 nRR1/D4(DSND)を見ますが、図 3 に示すように、減速停止命令を書くと不具合が発生するのは図 3 中 d の区間です。この区間では DSND ビットは 0 になり CNST ビットが 1 になります。従って、回避策として次の 2 つの方法を提案させていただきます。

(1) IC からの割込みを使用できる場合

減速が始まった時に割込みを発生させて、これ以降ドライブが終了するまでの間、減速停止命令(26h)の発行を禁止する方法です。減速停止命令禁止フラグを用意し、ドライブ開始前にクリアしておきます。IC の定速域終了割込みを有効にします(WR1/D13(C-END)=1)。定量パルスドライブを開始し、割り込みが入ったならば割込み処理ルーチン内で RR3/D5(C-END)を読みこのビットが 1 であれば定速域終了=減速開始ですので、減速停止命令禁止フラグを 1 にします。さらにドライブ終了間際にも CNST (定速域) が現れる可能性がありますので、ここで WR1/D13=0 に戻し、以降はこの割込みが発生しないようにします。一方タスク内ではこのフラグをみて 1 ならば減速停止命令の書き込みをしないようにします。

(2) 割込みを使用しない場合

減速停止をかけたい区間は加速時及び定速時(図 3 の a と b)ですが、図 3 に示すように、不具合を起こす区間 d も定速区間 b と同じ定速状態を示します。しかし現在速度に違いがあり定速区間 b では設定ドライブ速度に近く、不具合を起こす区間 d では初速度に近い値です。従って、ドライブ開始前に初速度とドライブ速度の中間の速度を判定速度 $((\text{ドライブ速度} - \text{初速度})/2 + \text{初速度})$ として求めておき、ドライブ中減速停止命令を発行する時には、加速中(ASND=1)または定速中(CNST=1)で且つ現在速度 \geq 判定速度ならば減速停止命令を発行するようにします。

2.2 減速停止モードのハードウェアリミット(nLMTP/M 信号)

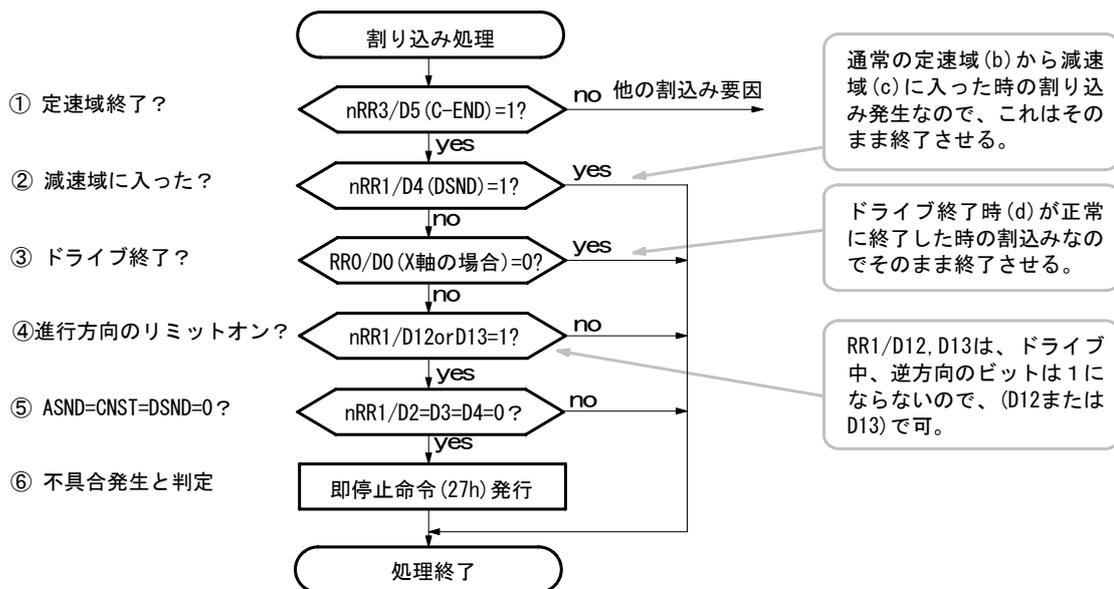
基本的には、S 字加減速の定量パルスドライブでは、ハードウェアリミット (nLMTP/M 信号) は即停止モードでのご使用をお願いします。

止もう得ず、即停止モードでの使用ができず、減速停止モードで行なう場合には次の対策をお願いします。多軸を同時制御する場合には、(1)の割込みによる方法が効果的です。

(1) IC からの割込みを使用できる場合

割込みの発生要因として、図 3 に示す S 字加減速の定速域(b 区間)終了を設定することができます。しかしこの割込みは、図 3 に示すように、ドライブ終了間際に d 区間がある場合、すなわち終了間際に初速度に到達した場合、あるいは加速度が 0 になった場合にも発生します。不具合はドライブ終了間際に、速度が初速度に到達しないで加速度が 0 に落ちた状態で、減速停止要求が発生すると起きますので、必ず d 区間が現れます。d 区間終了時の割込み発生タイミングで不具合か否かを判定する方法です。

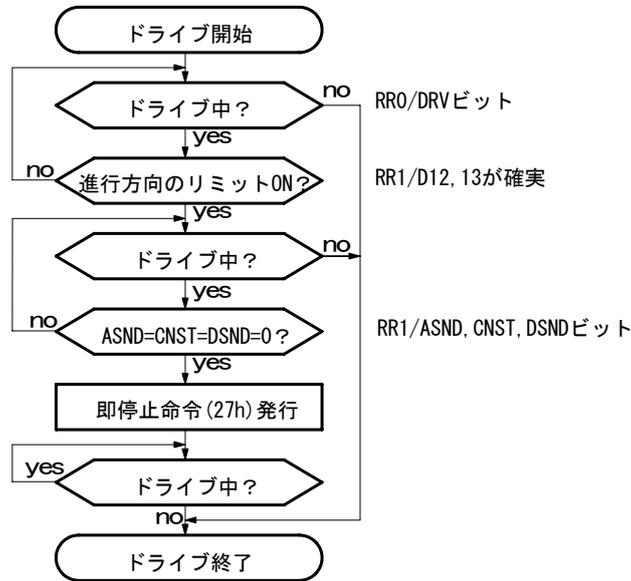
IC の定速域終了割込みを有効にします(WR1/D13(C-END)=1)。S 字加減速で定量パルスドライブを開始し、割込みが発生したら、下記の示す割込み処理を行いません。



- ① 定速域終了を確認します。ドライブしている軸の RR3/D5(C-END)ビットが0の場合は、他の割り込み要因ですので、そちらの割り込み処理を行いません。
- ② 減速域に入ったか確認します。RR1/D4(DSND)=1の場合は、図3のb区間からc区間にはいった時ですから、そのまま処理を終了します。=0の場合はd区間終了であることを示していますので、③の処理に移ります。
- ③ ドライブの終了を確認します。もしドライブが終了していれば、d区間も正常に終了したことになるので、そのまま処理を終了します。ドライブが終了していない場合は、d区間が終了したにもかかわらずドライブ状態であることですので、不具合が発生したことがほぼ確定できます。即停止命令をかける前に、次の④と⑤は安全のために確認します。
- ④ ハードリミットがオンしていることを確認します。RR1/D12,D13 ビットはそれぞれ+方向、-方向のリミットが作動すると1を示しますので、D12=1またはD13=1ならば、進行方向のリミットがオンしたと判断します。
- ⑤ 不具合発生時にはASND=CNST=DSND=0となるので、これを確認します。
- ⑥ 即停止命令を発行します。

(2) IC からの割込みを使用しない場合

図 3 の d 区間において進行方向のリミットがアクティブになると不具合が発生します(正確には、まれに発生する場合があります)。これを事前に回避する方策はありませんので、不具合が発生したら直ちに停止させる方法を取ることになります。図 3 に示すように不具合が発生すると(e 区間)、ドライブ状態のまま(RR0/nDRV=1)、加減速状態は ASND、CNST、DSND ともに 0 になります。この状態は正常なドライブでは起きません。よって、次のような対策のための処理例を示します。



S字加減速の定量パルスドライブを開始したら、タイマー割込みなどで進行方向のリミット信号の状態(RR1/D12,D13)を常に読み出して、リミット信号がアクティブになった場合には、RR1のASND、CNST、DSNDビットを更に常に読み出して、これら3つのビットがすべて0の場合には即停止コマンド(27h)を一回だけ発行してやります。

2.3 ソフトウェアリミット

定量パルスドライブでは、ドライブ前に、現在位置(論理位置カウンタ値)と出力パルス数の値から目的位置を計算することができます。目的位置がソフトウェアリミットの値を越えている場合にはドライブを行なわないようにして回避します。

2.4 STOP(2~0)信号(MCX314As は nIN(2~0))による減速停止

STOP(2~0)信号 (MCX314As は nIN(2~0)) による減速停止は、通常、連続パルスドライブで行なわれます。

しかしながら、止もう得ず、S字加減速の定量パルスドライブで STOP(IN)信号による減速停止を行なう場合には、2.2 と同様に不具合の発生を事前に回避する方策がありません。次のような対策のための処理例を示します。または、2.2(1)に示すような割り込みを用いる方法も有効です。

