

MCX514 使用説明書

2014. 4. 11 暫定版

NOVA electronics

株式会社 ノヴァ電子

1. 概要

1.1 主要機能及特徵

MCX514 是舊的 MCX314As/MCX314AL 在機能上大寬度提升的 4 軸運動控制 IC。

補間機能上、在已有的直線補間、圓弧補間、位元補間再加上XY平面圓弧補間時同步做Z軸垂直方向移動的螺旋補間機能。

MCX500系列的運動控制 IC、已經捨棄速度倍率設定。因此速度控制驅動可以從1pps的超低速到最高8Mpps的驅動速度做線性控制。

上位 CPU 的連結上在舊有的 16 位元/8 位元匯流排外增加 I²C 串列介面匯流排。因此可以連接無並列介面匯流排的 CPU。

■ 螺旋補間

MCX514 可以在已有的直線・圓弧補間做螺旋補間驅動。

螺旋補間為XY平面(直交座標)在做圓弧補間驅動的同一時間其他軸也一起移動的動作方式。如下圖所示，配合XY平面的圓弧補間下、Z軸做+方向的移動例。圖1為圓弧補間 1 圈以內的螺旋補間、圖 2 為複數圈數的示意。MCX514 可以做這兩種補間方式。

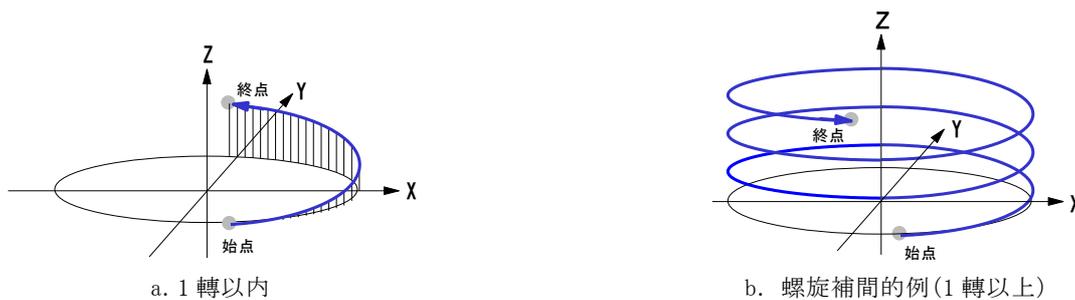


圖 1.1-1 螺旋補間例

另外下面為螺旋補間的一個應用例、配合XY平面圓弧補間的同時另一軸依迴轉軸座一定角度的迴轉、在圓弧的法線方向進行動作控制。圖 3 為XY平面上做圓弧補間的機台上加裝一迴轉軸設計、機台上的相機或是噴嘴可以隨時對正圓弧補間中心的動作方式。

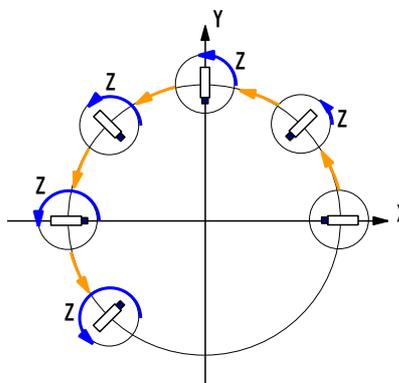


圖 1.1-2 X Y 軸圓弧補間及 Z 軸的法線控制例

■ 連續補間可用的 8 個暫存器

MCX514 為達高速連續補間驅動的所需、備有可存入 8 段終點資料的預設緩衝暫存器。

早期只有 1 段暫存器的 MCX314A 在做連續補間時、各補間區段需要在移動結束前內寫入下一區段位置資料，所以限制了各區段的長度。因此個區段的最小移動脈衝數會限制補間驅動速度。例如、CPU 資料寫入時間 $T_{DS}=80 \mu sec$ 、補間驅動速度 $V=100Kpps$ 時，最小移動脈衝數需 8 個脈衝以上必要、因此無法執行比此脈衝數還少的區段移動。

MCX514 因為緩衝暫存器增加為 8 個、所以大寬度改善此限制條件。如右圖所示執行連續補間的時候、如 Seg3 一樣短的區段存在也沒有關係、只要包含 Seg3 的 8 個區段的平均移動時間較下一區段寫入時間長就可以做連續補間。

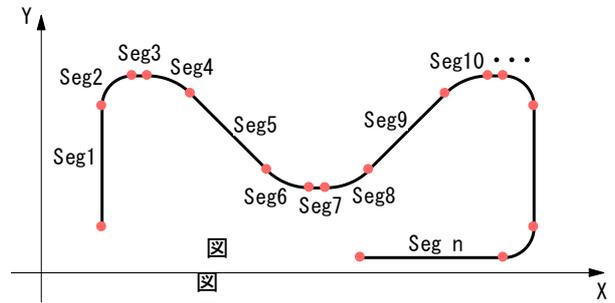


圖 1.1-3 連續補間例

■ 多晶片補間

MCX514 可以將複數晶片做連結來達到 5 軸以上的多軸直線補間。各晶片間將 8 條多晶片補間的訊號線並連即可。

一般多軸直線補間執行時補間的全部軸其終點最大值需經過補間演算、而 MCX514 則不需要特別去設定最大值。上位 CPU 將各軸的終點資料寫入對應的 IC 後、這些資料會經由多晶片補間訊號線送到各 IC、IC 內會自動產生終點最大值。

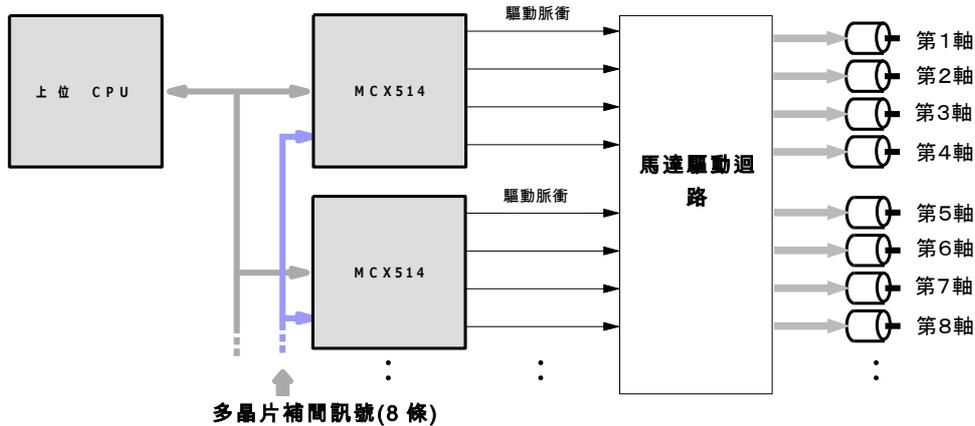


圖 1.1-4 多晶片補間的例

■ 補間短軸脈衝均一模式

一般補間驅動時、進行補間的全部軸在驅動中必須隨保持均一的周期下輸出驅動脈衝。例如下圖所示、2 軸直線補間時、移動量(脈衝數)多的一方為(長軸)保持脈衝連續輸出、少的一方(短軸)因補間演算結果有不會有脈衝輸出。在 MCX514 下、短軸因新增的脈衝均一化機能的原因。即使移動量少的軸也會儘可能將脈衝輸出周期均一、以相近的驅動脈衝輸出。並同線速一定模式使用時、可提高線速一定的精度。

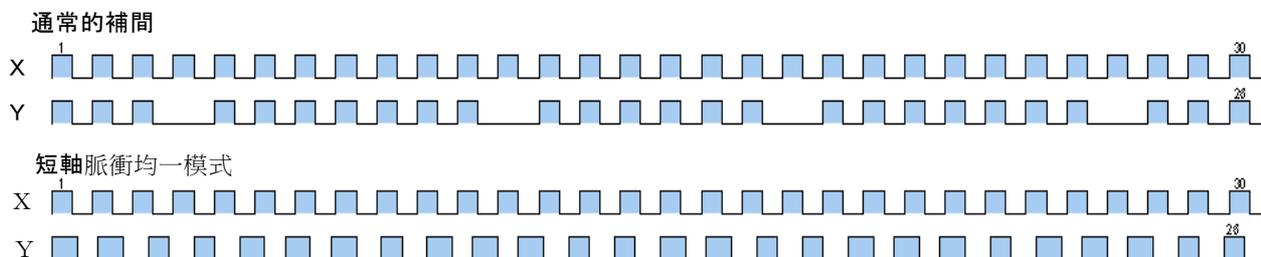


圖 1.1-5 移動量 X 30 脈衝 Y 26 脈衝的 2 軸直線補間驅動脈衝輸出

■ 2 軸高精度線速一定模式

線速為補間驅動時其軌跡先端的移動速度、亦稱為頭速。以補間驅動進行工件加工、塗布…等動作時、保持線速一定是相當重要的機能。

本 IC 在舊有的線速一定模式下、大寬度提升線速一定精度，獲得 2 軸高精度線速一定模式。在 2 軸直線補間、圓弧補間、螺旋補間驅動時、前述的短軸脈衝均一模式及 2 軸高精度線速一定模式組合使用時、線速的速度偏差可以在 $\pm 0.2\%$ 以下、期使補間驅動時大寬度提高速度精度。

下圖是半徑 10,000 脈衝的圓弧補間驅動、早期線速一定模式執行、與 MCX514 的 2 軸高精度線速一定模式執行時、其速度偏差圖形比較。

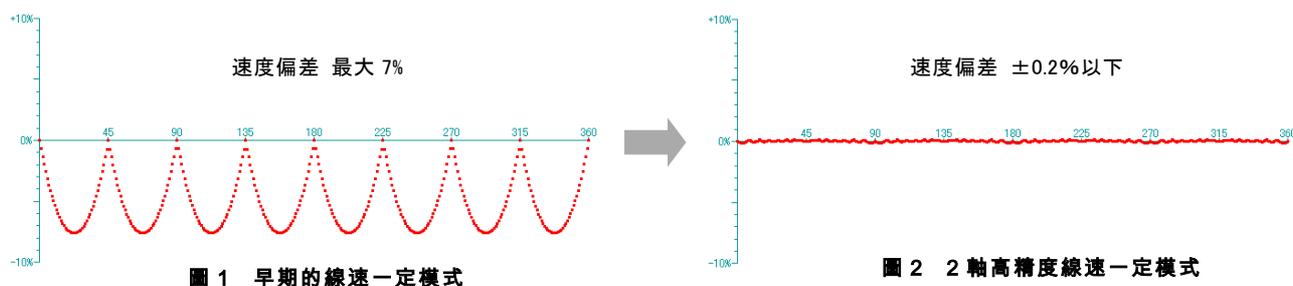


圖 1.1-6 線速一定模式速度偏差

■ 無段變速

MCX514 是前所未有的一顆不需設定速度倍率的運動控制 IC。因為、本 IC 在輸出速度範圍可由 1pps 到 8Mpps 為止、全部速度可以 1pps 為單位來設定。

早期使用速度倍率來設定速度時、

- ... 為了可以設定低的細小速度、速度倍率需設小 → 無法設定高速驅動
- 為了可以設定高的速度驅動、速度倍率需設大 → 無法設定細小驅動速度

如以上的限制。MCX514 解除速度倍率的設定不便、驅動中如 1pps、2pps 的低速到 1Mpps 一樣的高速脈衝均可直接變更所需速度。

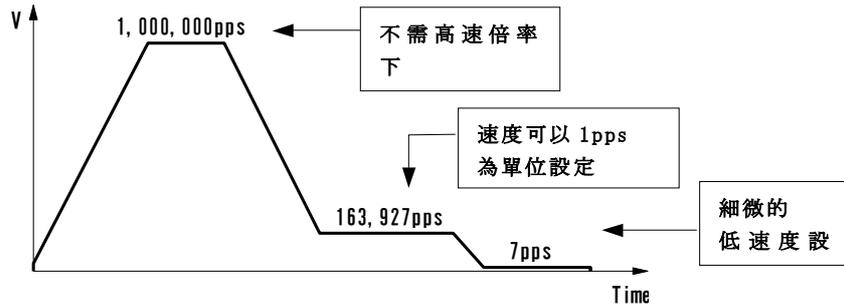


圖 1.1-7 無段變速

■ 簡單精確的速度設定

不需設定速度倍率之故、輸出驅動脈衝速度可以參數的方式設定。(CLK=16MHz 標準時)

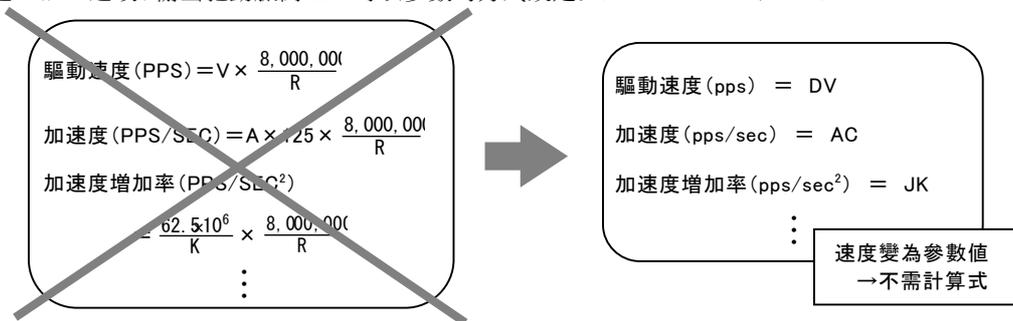


圖 1.1-8 速度設定

1pps 到 8Mpps 止的範圍下、設定的驅動速度以相當的精度輸出施初。輸出驅動脈衝速度精度可達到設定值的±0.1%以下。此為輸入 CLK 的時脈沒有誤差的情況下。實際上因輸入的 CLK 時脈會有誤差的關係，所以精度會受其影響。

■ 豐富加減速驅動模式

● 加減速驅動種類

加減速驅動可以、
等速驅動

直線加減速驅動(對稱/不對稱)

S 形

加減速驅動(對稱/不對稱)
...等方式進行。

● 位置驅動的自動減速

直線加減速(對稱/不對稱)、S形加減速驅動(對稱)的位置驅動...等、減速時減速開始位置由IC算出、自動開始減速。
(非對稱 S 形加減速驅動無此機能)

● S 形加減速曲線

S 形加減速在加速度或減速度時以一次直線增加/減少的方式進行之故、速度曲線為2次拋物線加速/減速。此外、本公司還以特有的方法來防止 S 形加減速中產生三角波形。

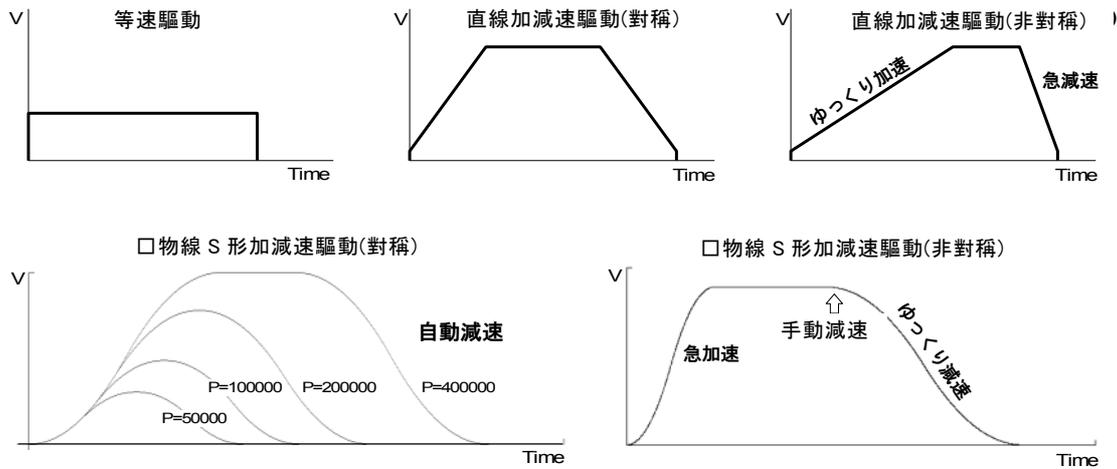


图 1.1-9 加減速驅動模式

■ 位置管理機能

驅動脈衝管理在IC內部有輸出計數的理論位置計數器及外部編碼器傳回的脈衝計數的實際位置計數器、共2個 32 位位置計數器。

另用資料讀取的命令可隨時讀取現在位置。

與同步動作搭配後、在通過指定位置時可以開始/停止計時器...等、可做為以位置資訊為起動要素的控制。

■ 軟體極限機能

具有驅動中、位置計數器超過指定範圍時驅動停止的軟體極限機能。超過軟體極限時驅動停止方式可選擇減速停止、或是立即停止。

■ 豐富的同步動作

同步動作是指、所定的起動要素發生後、執行預設連結動作的機能。因為不需要藉由 CPU 的處理、所以欲設的指定動作可以高速精確的執行。

同步動作每軸有 4 組可以設定。同步動作 1 組是指、欲設 1 個起動要素所指定的 1 個動作所構成。

起動要素可以是、指定位置通過、驅動開始・結束、外部輸入訊號觸發向上・觸發向下、計時結束...等。而指定的動作可以是驅動開始・停止、現在位置計數器值寫入多目的暫存器、寫入驅動速度書...等。

或是某軸的 1 組起動要素發生時、也可以連同該軸其他 3 組動作、及其他軸的 1 組動作、合計 7 組動作同時起動。利用同步動作的複數組合使用時、可以規畫豐富的應用程式。

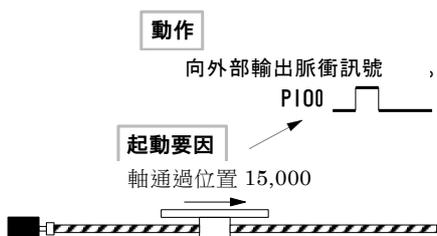


圖 1.1-10 同步期動作

- 驅動中指定位置通過時、外部訊號輸出。
- 驅動中外部訊號輸入時、現在位置存入指定的暫存器。
- 驅動中通過指定位置時輸出N個分割脈衝。
- …等

■ 4個多目的暫存器

每軸具有 32 位元長的多目的暫存器4個。搭配同步動作、驅動中的位置資料及現在速度…等參數值存入暫存器、或讀取欲設於暫存器的值多可以使用。

各軸多有 32 位元長的多目的暫存器4個。

多目的暫存器是、現在位置及速度、計時器與大小比較時、大小關係由狀態讀取後、可做為訊號輸出外、大小關係的變化啟動同步動作、觸發中斷。

或與同步動作搭配、將驅動途中現在位置及現在速度值儲存於多目的暫存器、再由多目的暫存器中保存的值載輸出脈衝數及驅動速度。

■ 計時器機能

每 1 軸多有 1 個計時器。可在 1 ~ 2,147,483,647 μ sec 範圍、以 1 μ sec 為單位設定 (CLK=16MHz 時)。搭配同步動作使用時、可以達到如下所述的各式動作精度。



圖 1.1-11 計時器機能

- 驅動結束、指定時間後驅動再開始。
- 外部訊號輸入、指定時間後驅動開始。
- 連續驅動在指定時間後停止。
- 測量從位置A到位置B所需移動時間。
- …等

■ 分割脈衝輸出

驅動中輸出分割脈衝的機能。軸移動中在所定的間隔下採取各式的同步動作時所使用的機能。可以設定的內容有分割間隔、脈衝寬、脈衝數。配合同步動作、可在指定位置分割脈衝開始/停止、變更外部訊號輸入分割間隔及寬度...等。也可以在做補間驅動時對任意軸做分割脈衝輸出。

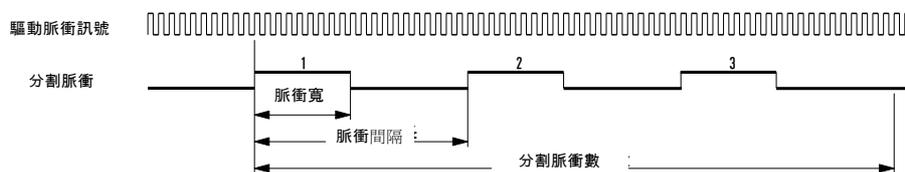


圖 1.1-12 分割脈衝輸出

■ 自動原點復歸機能

本 IC 可以不經過 CPU 的處理、自動執行做高速原點搜尋 → 低速原點貼近 → 編碼器 Z 相搜尋 → 偏移...等一連串原點復歸順序的機能。也可以輸出伺服馬達的偏差計數器清除訊號。各步驟間的停止時間也可以設定。旋轉軸的原點復歸也有適當的動作模式。

■ 各種伺服馬達用的訊號

2相編碼器訊號、定位完成、異常...等伺服馬達驅動器的訊號輸入接收。或、偏差計數器清除所需的輸出多有。

■ 中斷發生機能

加減速驅動中等速開始時、等速結束時、驅動結束時、位置計數器與多目的暫存器的大小關係變化...等、可依各軸所需做為中斷發生的要素。

連續補間驅動時也可以在緩衝暫存器的狀態變化時發生中斷。例如緩衝暫存器 8→7 變化時、4→3 變化時多可以發生中斷。

■ 外部訊號操作驅動

可以利用外部訊號、做相對位置驅動、連續驅動、手動脈衝產生器驅動。依此機能、即使在手動教導進給下也可以減輕上位 CPU 的多工處理、達到動作的順暢。

■ 內含輸入訊號濾波

IC內部、各輸入訊號的輸入段內含積分型濾波器。可設定各輸入訊號的的濾波機能有效或、訊號直接通過。另外、濾波器的時間常數共有 16 種 (500nsec~16msec) 可以選擇。

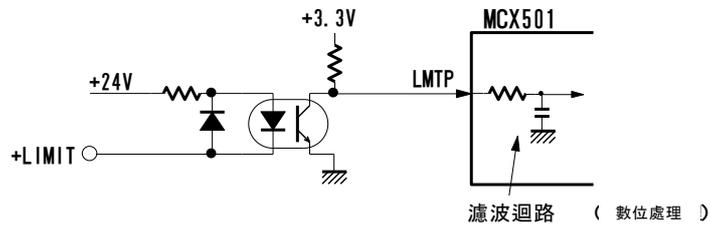


圖 1.1-13 輸入訊號濾波

■ 即時監控機能

驅動中現在理論位置、實際位置、驅動速度、加速度、加減速狀態(加速中、等速中、減速中、加速度增加中、加速度一定、加速度減少中)、計時器...等可以即時讀取。

■ CPU 介面

本 IC 連接上位 CPU 的介面上在舊有的 8 位元/16 位元資料匯流排上增加了 I²C 串列介面匯流排。I²C 串列介面匯流排是、將必要的匯流排線由串列·資料線(SDA) 及串列·時脈線(SCL) 整合為 2 條之故、因端子數少，所以可以使用在 PIC™ 微處理器...等 CPU 上。I²C 匯流排的特性上可在同一匯流排上連接複數 MCX514 或 EEPROM 的周邊。

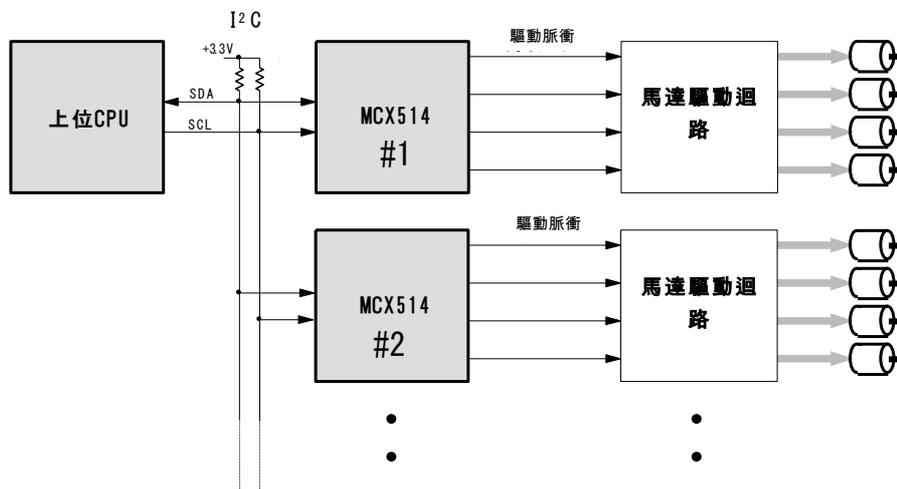


圖 1.1-14 I²C 串列介面匯流排例

1.2 機能方塊圖

图1.2-1 所示為本IC的機能方塊圖。X,Y,Z,U 4軸的控制部全部持有相同機能、由補間演算回路區塊所構成。補間驅動時、依指定的主軸(AX1)為基本脈衝發振時序做補間演算。等速驅動或加減速驅動多可進行。图1.2-2為、各軸軸控制部的機能方塊圖。

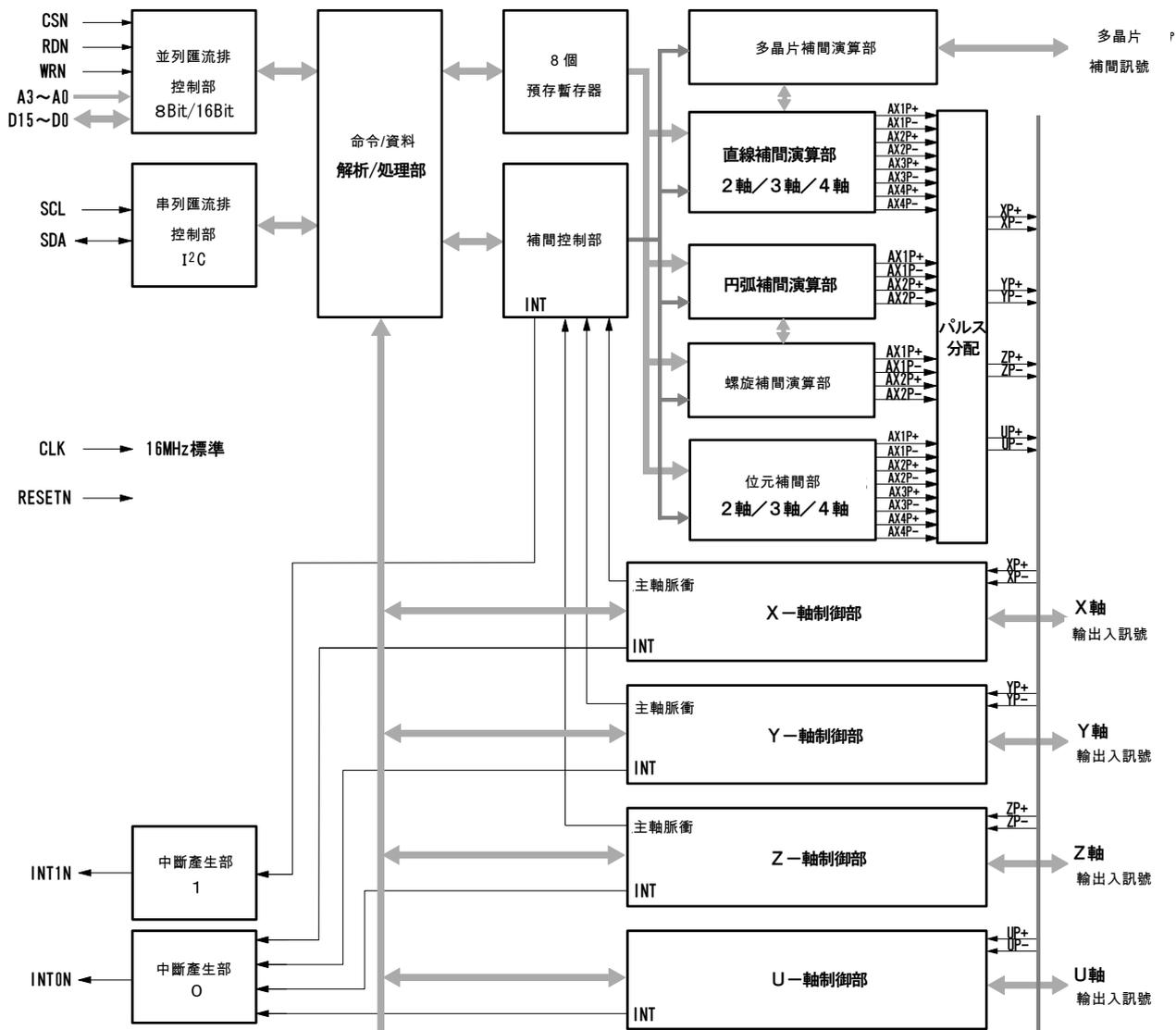
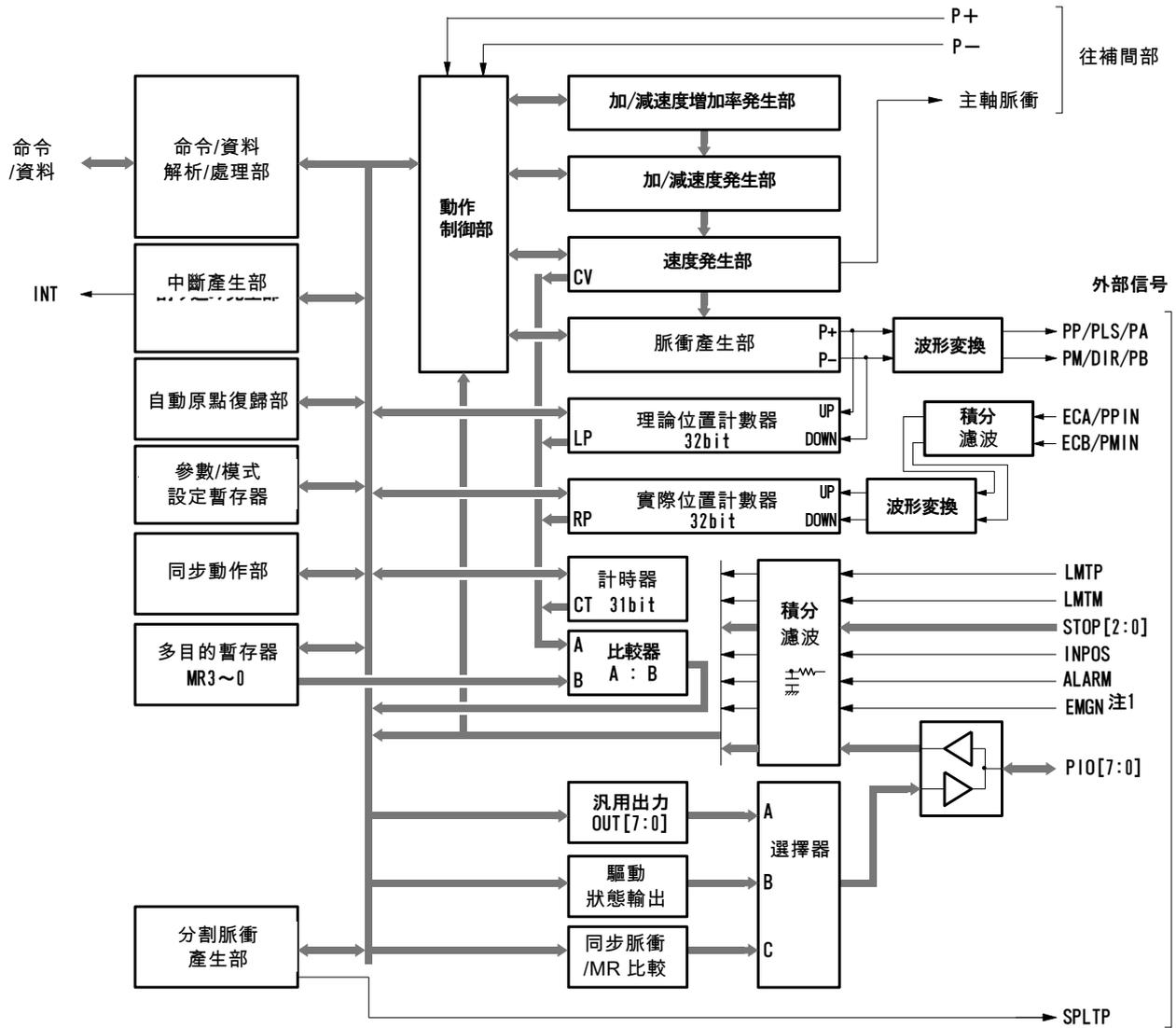


图 1.2-1 MCX514 全体機能方塊圖



注1. EMGN 為全軸共用

图 1.2-2 X、Y、Z、U 軸控制部內的方塊圖 (1 軸分)

1.3 規格一覽

(CLK = 16MHz 時)

項目	小項目	內容	補足
控制軸數		4 軸	
CPU 並列匯流排接統		16 位元或 8 位元匯流排選擇可能	
CPU 串列匯流排接統		I2C 串列介面匯流排	
補間機能	補間命令	2 軸/3 軸/4 軸直線補間、CW/CCW 円弧補間、 2 軸/3 軸/4 軸位元補間、CW/CCW 螺旋補間	
	補間範圍	各軸 -2,147,483,646 ~ 2,147,483,646 drive pulse	
	補間速度	1 pps ~ 8,000,000 pps	
	補間位置精度	±0.5LSB 以下(直線補間) ±1LSB 以下(円弧補間)	
	其他補間相關機能	<ul style="list-style-type: none"> ・任意軸選擇可能 ・短軸脈衝均一 ・線速一定(2 軸/3 軸簡單模式、2 軸高精度模式選擇可能) ・連續補間 ・8 段緩衝暫存器，資料緩衝連結 ・補間單步進給 ・多晶片多軸直線補間 ・直線補間最大值手動設定 ・円弧補間軸反轉 	
驅動脈衝輸出	驅動速度範圍	1 pps ~ 8,000,000 pps (CLK = 20MHz 時:最高 10,000,000pps)	
	初速度範圍	1 pps ~ 8,000,000 pps	
	輸出速度精度	±0.1%以下(設定值)	
	加速度範圍	1 pps/sec ~ 536,870,911pps/sec	
	加速度增加・減少率範圍	1 pps/sec ² ~ 1,073,741,823 pps/sec ²	*1
	加減速曲線	等速、對稱 / 非對稱直線加減速、對稱 / 非對稱 S 形加減速	
	驅動脈衝範圍	<ul style="list-style-type: none"> ・相對位置驅動 : -2,147,483,646 ~ 2,147,483,646 drive pulse ・絕對位置驅動 : -2,147,483,646 ~ 2,147,483,646 drive pulse 	*2
	位置驅動減速停止模式	自動減速停止 / 手動減速停止	*3
	覆寫	驅動中輸出脈衝數、驅動速度變更可能	*4
	驅動命令種類	相對位置、絕對位置、+ 方向連續、- 方向連續	
	三角防止機能	直線加減速、S 形加減速多有	
	驅動脈衝輸出方式	獨立 2 脈衝、1 脈衝・方向、2 相 4 通倍、2 相 2 通倍 選擇可能	
	驅動脈衝輸出理論	正理論 / 負理論輸出 選擇可能	
	驅動脈衝輸出端子	可替換端子	
編碼器輸入	輸入脈衝輸入方式	2 相 4 通倍 / 2 相 2 通倍 / 2 相 1 通倍 / 上升下降脈衝	
	輸入脈衝端子	可替換端子	
位置計數器	理論位置計數器	計數範圍: -2,147,483,648 ~ +2,147,483,647 drive pulse	*5
	實際位置計數器	計數範圍: -2,147,483,648 ~ +2,147,483,647 drive pulse	*5
	可變連結	可設定各計數器的計數最大值	
軟體極限	設定範圍	-2,147,483,648 ~ +2,147,483,647 drive pulse	
	停止模式	減速停止 / 即停止 選擇可能	
多目的暫存器	位元長・個數	32 位元長・4 個 / 軸	
	用途	位置・速度・計時器比較、位置・速度的登錄、驅動中現在位置・速度・計時器的儲存	
計時器	機能數	各軸 1 個	
	設定範圍	1 ~ 2,147,483,647 μ sec	
分割脈衝	訊號點數	各軸 1 點	
	分割間隔	1 ~ 65,535 drive pulse	*6
	分割脈衝寬	1 ~ 65,535 drive pulse	
	分割脈衝數	1 ~ 65,535、或無限	
自動原點復歸	順序	STEP1 高速原點搜尋 → STEP2 低速原點貼近 → STEP3 編碼器 Z 相搜尋 → STEP4 偏移量移動 ・可以選擇各步驟有效 / 無效、檢出訊號、檢出方向。	
	清除偏差計數器的輸出	清除脈衝寬在 10 μ ~ 20msec 內選擇、可以選擇理論位準	
	各步驟間等待時間	在 1msec ~ 1,000msec 內選擇	

同步動作	組數	各軸 4 組	*7
	起動要素	<ul style="list-style-type: none"> 多目的暫存器比較變化時 <ul style="list-style-type: none"> 比較對象:理論/實際位置計數器值、現在速度值、現在計時值 比較條件: \geq, $>$, $=$, $<$ 驅動開始/結束時、加減速驅動等速域開始/結束時 觸發動作開始/結束時、分割脈衝輸出時 PIO_n 訊號 \uparrow/\downarrow、PIO_{n+4} 訊號 Low 且 PIO_n 訊號 \uparrow、PIO_{n+4} 訊號 Hi 且 PIO_n 訊號 \uparrow、PIO_{n+4} 訊號 Low 且 PIO_n 訊號 \downarrow、PIO_{n+4} 訊號 Hi 且 PIO_n 訊號 \downarrow (n:0,1,2,3) 起動命令 	
	動作	<ul style="list-style-type: none"> 值的設定 (MR_n→設定值): 驅動速度、移動脈衝數(終點)、分割間隔、分割脈衝寬度、理論位置計數器值、實際位置計數器值、初速度、加速度 值的取得 (MR_n←現在值): 理論位置計數器值、實際位置計數器值、現在計時值、現在驅動速度、現在加速度 外部同步脈衝輸出 相對/絕對位置驅動起動、+/-方向連續驅動起動、MR_n 中設定的位置資料以相對/絕對位置驅動起動 驅動減速停止/即停止、速度增加/減少、計時開始/停止、分割脈衝動作開始/停止 	
	他 SYNC 起動	可設定其他 3 組的動作起動	
	他軸 SYNC0 起動	可設定其他軸 SYNC0 的動作起動	
	重複	可以設定同步動作為單一/重複	
中斷	訊號數	INTON、INT1N 2 條	
	中斷發生要素	<ul style="list-style-type: none"> 多目的暫存器比較變化時 <ul style="list-style-type: none"> 比較對象:理論/實際位置計數器值、現在速度值、現在計時口 比較條件: \geq, $>$, $=$, $<$ 驅動開始/結束時、加減速驅動等速域開始/結束時 自動原點出_し結束時、計時結束時 分割脈衝輸出時、觸發動作結束時 同步動作 0/1/2/3 起動時 8 段緩衝暫存器狀態變化時(連續補間驅動) 	
	有效/無效	各中斷要素有效/無效 選擇可能	
由外部訊號控制 驅動操作		<ul style="list-style-type: none"> 以 EXPP, EXPM 訊號做相對位置驅動、連續驅動 手動脈衝口生器(編碼器輸入:2 相 1 通倍) EXPLSN 訊號做補間單步進給 	*8
外部停止訊號	訊號點數	3 點 (STOP0~2)/軸	
	有效/無效	可以選擇停止訊號機能有效/無效	*9
	理論位準	可以選擇 Low 動作/Hi 動作	
	停止模式	動作時、驅動減速停止 (初速度以下驅動時只可以即停止)	
伺服馬達用輸出入訊號	訊號種類	ALARM(異常)、INPOS(定位完成)、DCC(偏差計數器清除)	
	有效/無效	可以選擇訊號的有效/無效	
	理論位準	可以選擇 Low 動作/Hi 動作	
汎用入/輸出訊號	訊號點數	8 點/軸 <ul style="list-style-type: none"> 同步輸入、外部驅動輸入訊號端子共用 同步動作輸出、多目的暫存器比較輸出、驅動狀態輸出訊號端子共用 	
驅動狀態輸出 訊號	訊號種類	<ul style="list-style-type: none"> 驅動中、錯誤中、加速/等速/減速中、加速度增加/一定/減少中 驅動狀態也可以由狀態暫存器讀取 	*10
極限開關訊號	訊號點數	2 點 (+方向、-方向 各 1 點)	
	有效/無效	可以選擇極限機能的有效/無效	*9
	理論位準	可以選擇 Low 動作/Hi 動作	
	停止模式	可以選擇動作時的驅動即停止/減速停止	
	輸入脈衝端子	可替換端子	
緊急停止訊號		全軸 EMGN 1 點 Low 為準時驅動脈衝輸出停止 (理論位準不能設定)	

積分型濾波器	輸入訊號濾波器	各訊號輸入段有積分濾波器	
	時間常數	有 16 種可以選擇 (500n, 1 μ , 2 μ , 4 μ , 8 μ , 16 μ , 32 μ , 64 μ , 128 μ , 256 μ , 512 μ , 1m, 2 m, 4 m, 8 m, 16 m[sec])	
	有效/無效	可以選擇濾波機能有效/無效	
電氣的特性	動作溫度範圍	-40°C ~ +85°C	
	動作電源電壓	+3.3V \pm 10%	
	消費電流	150mA (平均), 204mA (最大) CLK=16MHz 時	
	輸入時脈頻率	16MHz (標準) 20MHz (最大)	
	輸入訊號位準	TTL 位準 (5V tolerant)	
	輸出訊號位準	3.3V CMOS 位準 (5V 或僅可以連接 TTL)	
封裝		<ul style="list-style-type: none"> • 144 pin 塑膠 QFP, 口距 0.5mm RoHS 指令對應品 • 封裝尺寸 20 \times 20 \times 1.4 mm 	

< 補充說明 >

記号	
*1	S 形加減速驅動所使用的參數。
*2	指定之脈衝數輸出可以設定脈衝範圍。連續驅動時時可無限脈衝輸出。
*3	自動減速停止是指預設的移動脈衝量其減速開始位置由 IC 內部自動計算來做減速停止。手動減速停止是指由上位設定減速開始位置來做減速停止。本 IC 除非對稱 S 形加減速以外, 多可以做自動減速停止。
*4	驅動開始後變更輸出脈衝數、僅可以在同一方向的相對位置驅動可行。
*5	理論位置計數器做為驅動脈衝輸出的計數。實際位置計數器做為編碼器輸入脈衝的計數。
*6	驅動中與驅動脈衝同步、指定的脈衝間隔下輸出分割脈衝。
*7	同步動作 1 組是指、指定的 1 個起動要素搭配指定的 1 個動作組成。
*8	外部訊號輸入端子與汎用輸出入端子共用。
*9	不使用該機能時、可以做為汎用輸入使用。
*10	驅動狀態輸出端子與汎用輸出入端子共用。

2. 機能說明

2.1 定量脈衝驅動與連續脈衝驅動

驅動脈波輸出包括預訂輸出脈波數進行的定量脈波驅動，與發出停止指令，或停止訊號輸入之前，持續輸出脈波的連續脈波驅動。定量脈波驅動包括相對位置驅動、絕對位置驅動、反相對位置驅動。連續脈波驅動則有＋方向連續脈波驅動、－方向連續脈波驅動。

- 定量脈衝驅動
 - 相對位置驅動
 - 絕對位置驅動
 - 反相對位置驅動
- 連續脈衝驅動
 - ＋方向連續脈衝驅動
 - －方向連續脈衝驅動

2.1.1 相對位置驅動

相對位置驅動是由現在位置，指定移動脈波數驅動。在現在位置欲往＋方向移動時，移動脈波數設為正的脈波值，欲往－方向移動時，則設為負的脈波值。

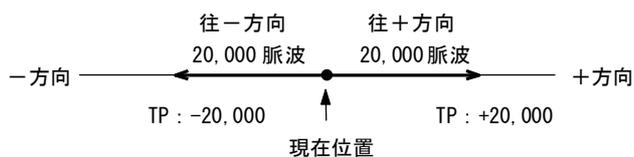


圖 2.1-1 相對位置驅動之移動脈波數 (TP) 設定例

相對位置驅動以等速或加減速驅動進行。加速度與減速度相等，以加減速的相對位置驅動之動作如圖 2.1-2 所示，剩餘移動脈波在加速時若消耗到小於脈波數，即開始自動減速，輸出指定的移動脈波數結束即結束驅動。

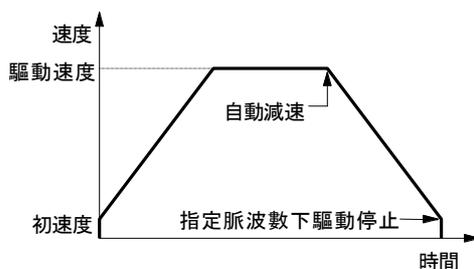


圖 2.1-2 相對位置驅動之自動減速及停止

相對位置驅動的指令碼為 50h。以直線加減速進行相對位置驅動，必須設定以下參數。

表 2.1-1 設定參數：相對位置驅動

參數名	記号	備註
加/減速度	AC/DC	加減速相等時，不需減速設定
初速度	SV	
驅動速度	DV	
移動脈波數/終點	TP	指定往＋方向的驅動為＋脈波數、 往－方向的驅動為－脈波數

2.1.2 絕對位置驅動

絕對位置驅動是指定原點（理論位置計數值：0）為基準的移動目標位置驅動。不拘現在位置何在，指定絕對座標，即可得出移動目標位置。於 IC 內部根據指定移動目標位置與現在位置的差距，計算出驅動方向與輸出脈波數驅動。

因絕對位置驅動為以移動空間座標指定移動目標，所以必須預先做原點復歸，確定理論位置計數器後驅動。

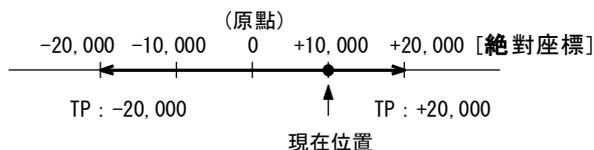


圖 2.1-3 絕對位置驅動之終點 (TP) 指定例

絕對位置驅動同相對位置驅動，以等速或加減速驅動。

絕對位置驅動的指令碼為 54h。以直線加減速進行絕對位置驅動，必須設定以下參數。

表 2.1-2 設定參數：絕對位置驅動

參數名	記号	備註
加/減速度	AC/DC	加速與減速相等時，不需減速設定
初速度	SV	
驅動速度	DV	
移動脈波數/終點	TP	指定移動目標之絕對座標

2.1.3 反相對位置驅動

反相對位置驅動是以現在位置為基準，往移動目標位置的方向與進行指定移動脈波數驅動。不同於相對位置驅動，與移動脈波數（TP）所設定的脈波數符號呈反向驅動。

預先在移動脈波數設定特定的正脈波值，以驅動指令決定驅動方向時使用。

移動脈波數設定負脈波值時，反相對位置驅動是在 + 方向上驅動。

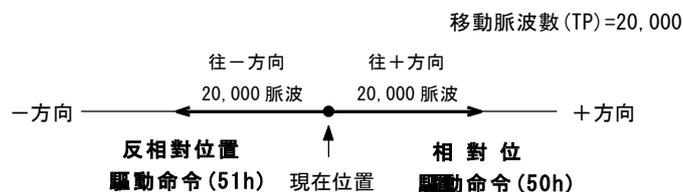


圖 2.1-4 相對/反相對位置驅動指令來決定驅動方向

反相對位置驅動，除與在移動脈波數設定的脈波值符號逆向驅動外的動作，與相對位置驅動一樣。反相對位置驅動指令碼為 51h。

A. 驅動途中變更移動脈波數(覆載)

於相對位置驅動與反相對位置驅動途中，可變更改移動脈波數（TP）。但移動脈波數在變更前與變更後的驅動方向必須相同。欲設成改變驅動方向的值時，將無法變更移動脈波數。

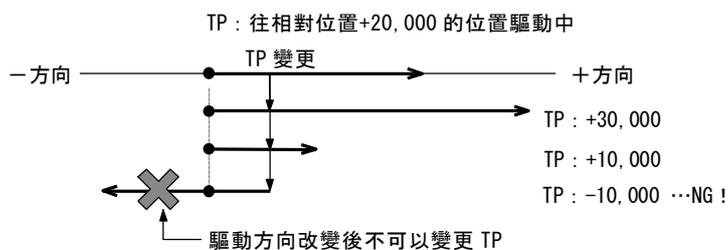


圖 2.1-5 相對位置驅動的移動脈波數 (TP) 覆載

以加減速驅動中，剩餘的輸出脈波，變成較加速時脈波為少，而進入減速時，當移動脈波數 (TP) 被變更後，即再度開始加速(圖 2.1-7)。此外，若變更後移動脈波數(TP)的輸出脈波數，比已經輸出的脈波數少時，驅動將立即停止(圖 2.1-8)。在 S 形加減速中，若於減速時變更移動脈波數 (TP)，無法畫出正確的 S 形曲線，敬請注意。

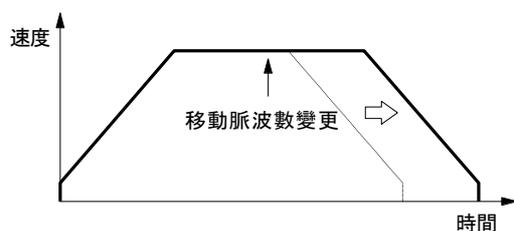


圖 2.1-6 驅動中移動脈波數變更

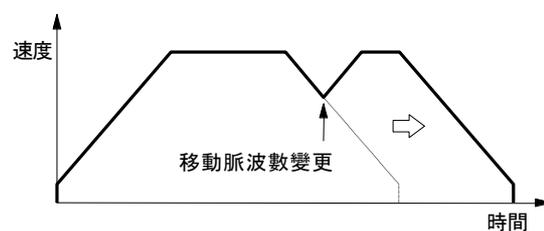


圖 2.1-7 減速時移動脈波數變更

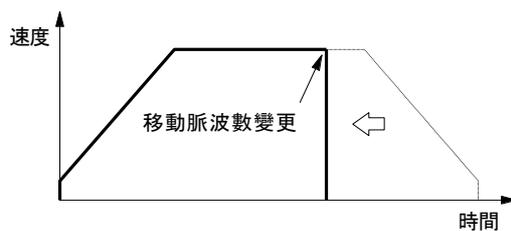


圖 2.1-8 比已輸出脈波還少的移動脈波數變更

【注意】

- 絕對位置驅動在驅動途中，無法變更移動脈波數 (TP)。

B. 於加減速定量脈波驅動時手動減速

定量脈波驅動於加減速驅動時（相對位置驅動、絕對位置驅動、反相對位置驅動），一般為如圖 2.1-2 所示，由 IC 已計算出的減速點開始自動減速，但該減速點亦可手動指定。

在以下的情況中，自動減速點會偏低，無法正確算出，因此必須手動指定減速點。

- 直線加減速之定量脈波驅動中，多次在驅動途中變更速度。
- 非對稱梯形加減速、與 S 形加減速之定量脈波驅動中，在驅動途中變更速度。
- S 形加減速定量脈波驅動，個別設定加速度與減速度、加速度增加率與減速度增加率。
（非對稱 S 形加減速）

設定成手動減速模式，是將 WR3 暫存器的 D0 位元設為 1，以手動減速點設定指令(07h)設定減速點。其他操作則與一般的定量脈波驅動相同。

C. 加減速定量脈波驅動之加速計數器偏移

加減速定量脈波驅動之動作中，加速時會利用加速計數器，計算加速所消耗脈波。於加減速度相等的加減速驅動中，設定的剩餘輸出脈波數，少於加速計數值時便開始減速，在減速時，會輸出與加速時相同的脈波數。

加速計數器偏移，是在該加速計數器上，加計指定的偏移量。如圖 2.1-9 所示，偏移量的正值越大，則自動減速點會往前移動，減速結束時，初速度拖拉變長。此外，偏移量若設定為負值，則有未減到初速度即斷尾的現象。

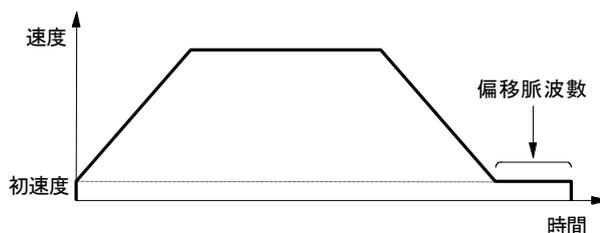


圖 2.1-9 加速計數器偏移

加速計數器偏移量在重置後設為 0。在一般直線加減速驅動時，幾乎不需要重設此參數。在非對稱梯形加減速和 S 形加減速的定量脈波驅動下，因初速度設低，在驅動結束時有拖拉脈波和斷尾問題發生時，可設定適當的加速計數器偏移值來補正。

2.1.4 連續脈波驅動

連續脈波驅動在上位發出停止指令，或來自外部的停止訊號有效之前，會連續輸出驅動脈波。於原點搜尋、掃描教導進給，或以速度控制驅動馬達旋轉時使用。

停止指令包括減速停止指令與即停止指令。此外，外部的減速停止（驅動速度在初速度以下時即停止）訊號，備有 STOP0 ~ STOP2 等 3 點。各個訊號可設定有效/無效、或動作有效位準模式設定。

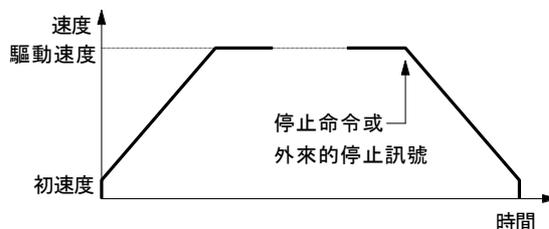


圖 2.1-10 連續脈波驅動

連續脈波驅動指令備有 + 方向連續脈波驅動指令 (52h) 與 - 方向連續脈波驅動指令 (53h)。連續脈波驅動以加減速進行，除移動脈波數 (TP) 以外，必須與定量脈波驅動設定相同的參數。

表 2.1-3 設定參數：連續脈波驅動

參數名	記号	備註
加/減速度	AC/DC	加速與減速相等時，不需減速度設定
初速度	SV	
驅動速度	DV	

■ 驅動途中變更驅動速度(覆載)

連續脈波驅動於驅動途中可自由變更驅動速度。驅動速度變更方法包括變更驅動速度參數(DV)方法，與利用速度增加、減少指令的方法。

但 S 形加減速驅動，在加減速中變更速度也無效。

於定量脈波驅動之對稱直線加減速與等速驅動，驅動途中可變更驅動速度。但於直線加減速的定量脈波驅動中，若變更驅動速度，多少會發生斷尾的情況，因此，使用較低的初速度設定時，請多加注意。

而定量脈波驅動(自動減速模式時)之非對稱直線加減速與 S 形加減速，於驅動途中無法變更驅動速度。

<利用變更驅動速度參數變更速度>

若利用驅動速度設定指令(05h)變更驅動速度參數(DV)，設定會立即被反映，若有加減速驅動，即加減速到指定的驅動速度。

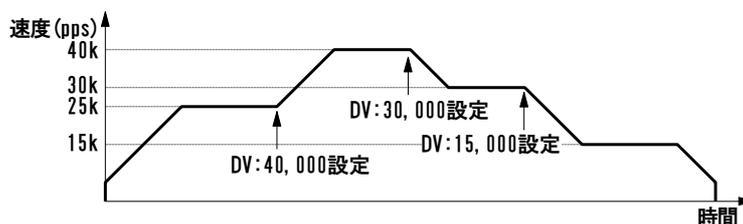


圖 2.1-11 驅動途中驅動速度變更例

<利用速度增加、減少指令變更速度>

預先設定速度增減值參數(IV)上的增減速度。驅動途中發出速度增加指令(70h)、速度減少指令(71h)，立即反映於設定，若為加減速驅動，即由現在速度，加減速度增減值參數程度的速度。

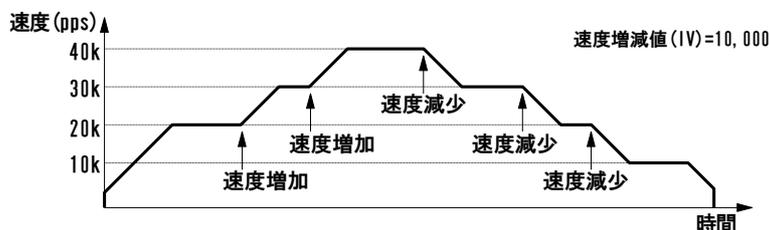


圖 2.1-12 速度增加，減少指令之速度變更例

■ 利用連續脈波驅動做原點搜尋動作

將近原點訊號、原點訊號、編碼器 Z 相訊號等分配到 STOP0~STOP2。(編碼器 Z 相訊號請接到 STOP2。)在 WR2 暫存器設定各訊號之有效/無效、及邏輯準位。高速搜尋時以加減速連續脈波驅動。輸入訊號若設為有效，且變成有效準位時減速停止。低速搜尋時，以等速執行連續脈波驅動。輸入訊號若設為有效，且變成有效準位時立即停止。本 IC 亦備有自動原點復歸機能。自動原點復歸機能請參 2.5 節。

2.2 加減速

驅動脈波輸出的速度曲線，包括不執行加減速的等速驅動、朝目標速度直線加減速的直線加減速驅動、朝目標速度以流暢曲線加減速的 S 形加減速驅動。

加減速驅動分別備有加速度與減速度相等的對稱加減速，與加速度與減速度需個別設定的非對稱加減速。

- 等速驅動
- 加減速驅動
 - 直線加減速驅動
 - 直線加減速(對稱)
 - 非對稱直線加減速
 - S形加減速驅動
 - S形加減速(對稱)
 - 非對稱S形加減速

2.2.1 等速

等速驅動是不進行加減速，持續以固等速度輸出驅動脈波。等速驅動的驅動速度需設定在初速度以下的值（亦即初速度高於驅動速度的值）。等速驅動低於初速度之驅動速度時，驅動並不加減速。停止動作為即停方式。搜尋原點和搜尋編碼器 Z 相等訊號後欲即停時，則不執行加減速驅動，而自始即以低速等速驅動。

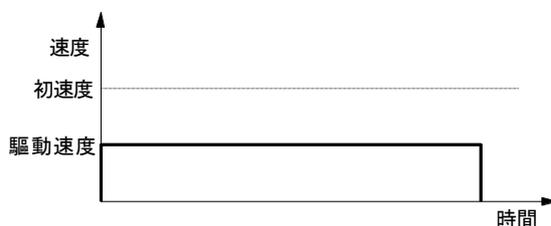


圖 2.2-1 等速驅動

等速驅動必須設定以下參數。

表 2.2-1 設定參數：等速驅動

參數名	記号	備註
初速度	SV	設定高於驅動速度 (DV) 之速度值
驅動速度	DV	
移動脈波數/終點	TP	連續脈波驅動不需設定

■ 參數設定例

驅動速度 980pps 之等速驅動例。此例中在移動脈波值 2,450 的相對位置驅動。

初速度	SV = 980	設定初速度 \geq 驅動速度 值
驅動速度	DV = 980	
移動脈波數	TP = 2450	

各參數相關內容請參 5.2 節。

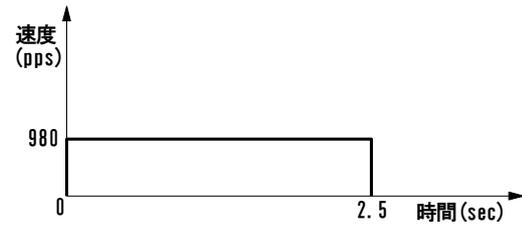


圖 2.2-2 等速驅動例

2.2.2 直線加減速(對稱)

直線加減速驅動是以指定加速度斜率，用一次直線加速方式，自初速度開始驅動，加速到驅動速度。直線加減速驅動可自動減速，不需手動設定減速點。於加速度與減速度相等的對稱直線加減速定量脈波驅動中，會計算加速中消耗的脈波數，當剩餘輸出脈波數少於加速中的脈波數時，即自動開始減速。減速與加速度以具相同斜率的一次直線，減速到初速度，至所有輸出脈波數輸出完畢便停止。

在加速中發生減速停止的情況，如圖 2.2-3 所示於加速途中開始減速。

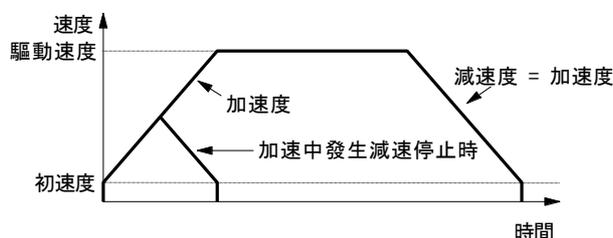


圖 2.2-3 直線加減速驅動 (對稱)

以自動減速執行對稱直線加減速驅動，WR3 暫存器的 D2~0 位元如下設定。並必須設定以下參數。

表 2.2-2 模式設定：直線加減速 (對稱)

模式設定位元	記号	設定值	備註
WR3/D0	MANLD	0	自動減速
WR3/D1	DSNDE	0	減速時使用加速度設定值 (對稱)
WR3/D2	SACC	0	直線加減速

表 2.2-3 設定參數：直線加減速 (對稱)

參數名	記号	備註
加速度	AC	減速時以此值減速
初速度	SV	
驅動速度	DV	
移動脈波數	TP	連續脈波驅動不需要

■ 參數設定例

初速度：500pps、驅動速度:15,000pps 之前以 0.3 秒進行直線加減速之參數設定如下所示。

加速度	AC = 48333	$(15000-500)/0.3$ = 48333pps/sec
初速度	SV = 500	
驅動速度	DV = 15000	

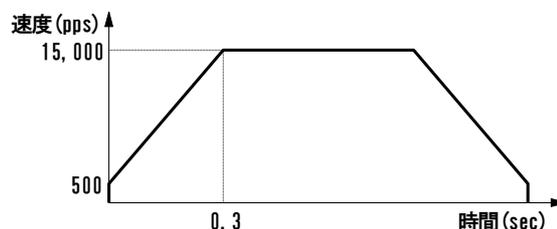


圖 2.2-4 直線加減速驅動 (對稱) 例

各參數相關內容請參 5.2 節。

■ 直線加減速之三角波形防止（定量脈波驅動）

防止三角波形機能是即便於直線加減速之定量脈波驅動中，輸出脈波數未達到加速至驅動速度所需脈波數時，也能防止三角波形的機能。三角波形是指於直線加減速驅動，加減速轉變的速度曲線。

藉由防止三角波形機能，在加速中的加速時及減速時所消耗脈波數之合計，若超過定量驅動總輸出脈波數的 1/2 立即停止加速，維持該速度驅動後自動減速。從而，即便定量脈波驅動的輸出脈波數很少，輸出脈波數的 1/2 都會進入等速域，亦可將三角波形變成梯形波形。

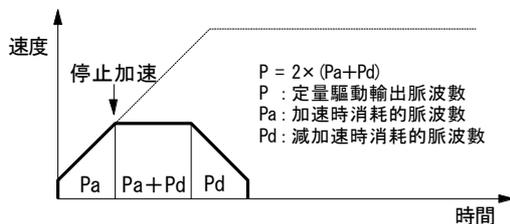


圖 2.2-5 直線加減速驅動之角波形防止

防止直線加減速定量脈波驅動的三角波形機能，在重置後自動變為有效。將 WR3 的 D13 位元設定為 1，即可解除本機能。

於直線加減速驅動中，在加速中遇到減速停止訊號時，無法防止產生三角波形。如圖 2.2-3 所示的發生減速停止時即開始減速。

2.2.3 非對稱直線加減速

有時在各種材料堆疊設備中，包括於垂直方向移動標的物的情況，對標的物施加重力加速度時，欲改變上下移動的加速度與減速度的情況。

本 IC 在這種加速度與減速度不同的非對稱直線加減速定量脈波驅動中，也可自動減速。不需事先計算再設定手動減速點。圖 2.2-6 為減速度比加速度大之實例、圖 2.2-7 為加速度比減速度大之實例。於此類非對稱直線加減速狀態下，也是於 IC 內部計算出定量脈波驅動的輸出脈波數，及根據各速度參數值算出自動減速開始點。

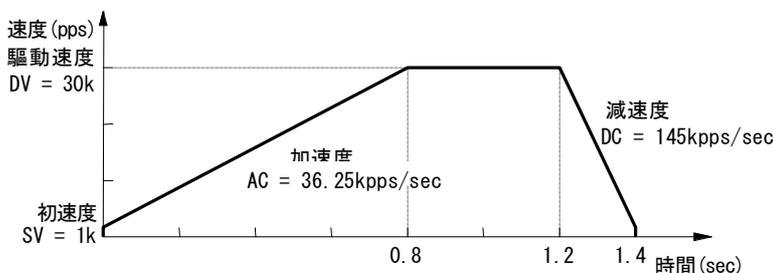


圖 2.2-6 非對稱直線加減速驅動（加速度 < 減速度）

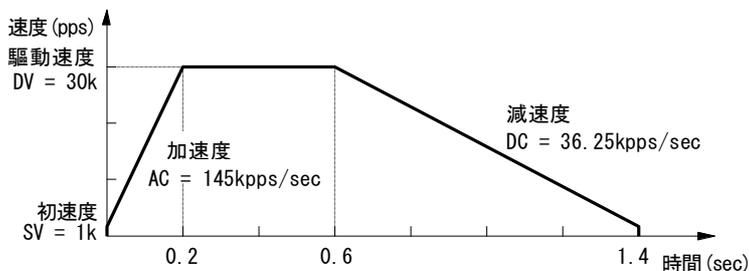


圖 2.2-7 非對稱直線加減速驅動（加速度 > 減速度）

非對稱直線加減速驅動中，執行自動減速時，WR3 暫存器的 D2~0 位元應做如下設定。或必須設定以下參數。

表 2.2-4 模式設定：非對稱直線加減速

模式設定位元	記号	設定值	備註
WR3/D0	MANLD	0	自動減速
WR3/D1	DSNDE	1	在減速時使用減速度設定值
WR3/D2	SACC	0	直線加減速

表 2.2-5 設定參數：非對稱直線加減速

參數名	記号	備註
加速度	AC	
減速度	DC	
初速度	SV	
驅動速度	DV	
移動脈波數／終點	TP	連續脈波驅動不需要

【注意】

- 於非對稱直線加減速驅動，加速度 > 減速度（圖 2.2-7）的情況，加速度與減速度的比率需符合以下條件。

$$DC > AC \times \frac{DV}{8 \times 10^6}$$

DC：減速度 (pps/sec)

AC：加速度 (pps/sec)

DV：驅動速度 (pps)

其中 CLK = 16MHz

例如，若驅動速度 DV = 100kpps，減速度 D 的值，必須大於加速度 A 值之 1/80。不得小於 1/80。

- 於非對稱直線加減速驅動中，加速度 > 減速度（圖 2.2-7）的情況，加速度 AC 與減速度 DC 的比率變得越大，拖拉脈波會變多（AC/DC = 10 倍之最大 10 脈波程度）。若拖拉脈波造成問題時，可提升初速度，將加速計數器偏移設為負值等方式因應。

■ 參數設定例

前述如圖 2.2-6 所示的非對稱直線加減速（加速度 < 減速度）相對位置驅動之參數設定如下。

模式設定	WR3←0002h	WR3 暫存器的模式設定
加速度	AC = 36250	(30000-1000)/0.8 = 36250pps/sec
減速度	DC = 145000	(30000-1000)/0.2 = 145000pps/sec
初速度	SV = 1000	
驅動速度	DV = 30000	
移動脈波數	TP = 27500	相對位置驅動

2.2.4 S 形加減速（對稱）

S 形加減速驅動為到達指定驅動速度之前，是以 2 次曲線（拋物線）式流暢速度曲線方式加減速。

本 IC 於驅動速度在加減速時，將加速度／減速度以一次直線式增加／減少，產生 S 形曲線的速度。加速及減速呈對稱 S 形加減速驅動之動作如圖 2.2-8 所示。

- 區間 a. 驅動開始即開始加速，加速度由 0 開始，以指定加速度增加率，直線增加。此時的速度曲線形成 2 次曲線。
- 區間 b. 設定的驅動速度與現在速度之差，若少於在加速度增加中所消耗的速度，則加速度即會以指定加速度增加率，呈直線減少。加速度減少的比例，與加速度增加時相同。此時的速度曲線變成反向拋物線狀。
- 區間 c. 速度到達指定的驅動速度或加速度到達 0，則維持該速度驅動。
- 加速與減速呈對稱 S 形加減速之定量脈波驅動，剩餘輸出脈波數若小於加速消耗的脈波數，即開始減速。(自動減速)
- 區間 d.e. 於減速時也和加速時同樣，令減速度呈一次直線增加／減少，即產生速度的 S 形曲線。

此外，於連續脈波驅動途中變更驅動速度的情況，即便於加速／減速中，也是執行同樣動作。但於 S 形加減速驅動中，在加減速中變更速度也無效。

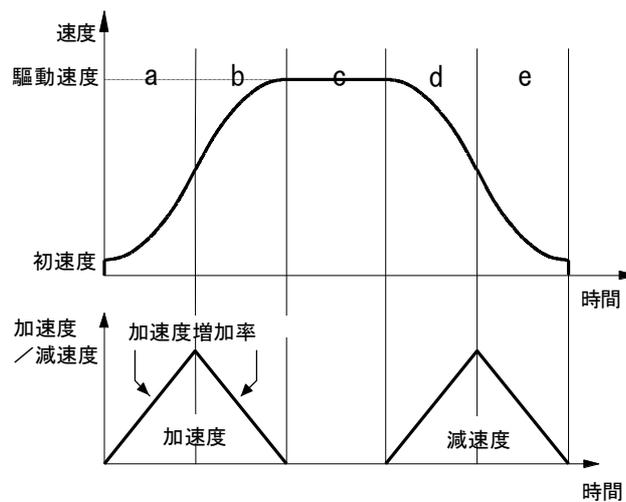


圖 2.2-8 S 形加減速驅動 (對稱)

於對稱 S 形加減速驅動中進行自動減速時。WR3 暫存器的 D2~0 位元做如下設定。並且必須設定以下參數。

表 2.2-6 模式設定：S 形加減速 (對稱)

模式設定位元	記号	設定值	備註
WR3/D0	MANLD	0	自動減速
WR3/D1	DSNDE	0	減速時使用加速度設定值、加速度增加率設定值
WR3/D2	SACC	1	S 形加減速

表 2.2-7 設定參數：S 形加減速 (對稱)

參數名	記号	備註
加速度增加率	JK	
加速度	AC	最大值：設為 536,870,911 (1FFF FFFFh)
初速度	SV	
驅動速度	DV	
移動脈波數／終點	TP	連續脈波驅動不需要

■ S 形加減速的三角波形防止

於 S 形加減速驅動為保持速度曲線流暢度，亦配備有三角波形防止機能。S 形加減速驅動的三角波形防止機能為，加速及減速對稱的 S 形加減速定量脈波驅動之輸出脈波數，不足加速到達驅動速度所需脈波數的情況時，將 S 形加速轉為減

速停止，使仍能保持雙方的機能及滑順的速度曲線。

< 定量脈波驅動之三角波形防止 >

於加速與減速為對稱的 S 形加減速定量脈波驅動中，輸出脈波數在到達驅動速度之前，不足加速所需脈波的情況，採用以下方式。

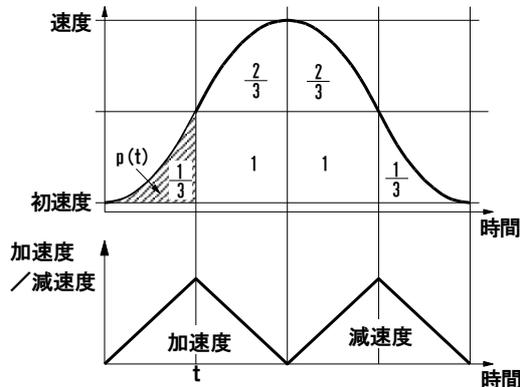


圖 2.2-9 S 形加減速的 1/12 則

初速度為 0 時，令加速度以某一加速度增加率，在時間 t 之前增加。在該加速度的增加區間中，時間 t 的速度 $v(t)$ 以下式顯示。

$$v(t) = at^2 \quad a: \text{速度相關係數}$$

據此，由時間 0 到 t 之間所消耗脈波數 $p(t)$ ，為速度 $v(t)$ 由時間 0 到 t 之積分值如下式。

$$p(t) = \frac{1}{3} \times at^3$$

該值和加速度增加率的值無關，用以顯示是 $at^2 \times t$ （圖中 1 格脈波數）的 1/3。

於定量脈波驅動中，由時間 0 到 t 的加速度，令其以某一加速度增加率增加，由時間 t 開始，以相同加速度增加率來減少加速度。當加速度變成 0 之後，在減速時也一樣，以相同的加速度增加率增加/減少減速度，則定量脈波驅動全體消耗的脈波數，如圖 2.2-9 所示，成為下式之脈波數。

$$\frac{1}{3} + \frac{2}{3} + 1 + 1 + \frac{2}{3} + \frac{1}{3} = 4 \text{ 格}$$

因而，驅動開始的加速度增加區間，由時間 0 到 t 之消耗脈波數(1/3 格)，變成定量脈波驅動全體被消耗的脈波數之 1/12。

基於上述理由，本 IC 在 S 形加減速定量脈波驅動中，加速度增加時的脈波若大於總輸出脈波的 1/12，即轉變成減少加速度，可畫出如圖 2.2-9 所示的速度曲線。[1/12 法則]

此方式於初速度 = 0 時為最理想曲線。由於初速度實際上不會是 0，因此，圖中速度由 0 到初速度的脈波數會多出來，該脈波數在速度峰值時會輸出。

< 減速停止之三角波形防止 >

於直線加減速驅動中，在加速中發生減速停止訊號時，速度曲線呈三角波形。但 S 形加減速驅動時，因相當重視速度曲線之滑順度，在發生圖 2.2-10 之加速時減速停止的情況，並不立即轉換成減速，而是將加速度暫且減至 0 後，再轉換成減速。

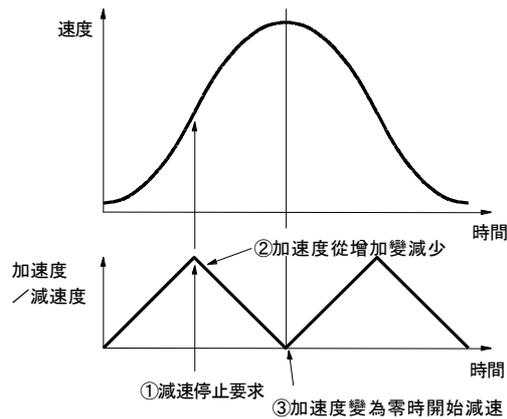


圖 2.2-10 S 形加減速中減速停止之三角波形防止

■ S 形加減速驅動的注意事項

- 於 S 形加減速定量脈波驅動中，不可在驅動途中變更驅動速度。
- 於 S 形加減速定量脈波驅動中，在減速時變更移動脈波數後，無法正確畫出 S 形曲線。
- 於 S 形加減速定量脈波驅動中，將初速度設成極低，則在減速時可能發生斷尾停止（降至初速度之前，已結束輸出指定驅動脈波而結束的現象）和拖拉現象（即便達到初速度，仍無法結束輸出指定驅動脈波，而以初速度輸出剩餘驅動脈波的現象）的情況。
- 於 S 形加減速之連續脈波驅動中，可於驅動途中變更速度。
然而，於加減速途中，驅動速度變更指令無效，無法變更速度。
於 S 形加減速之連續脈波驅動欲變更速度時，請務必在等速域（RR0 暫存器 CNST=1）進行。
利用速度增加、減少指令（70h,71h）與同步動作變更速度，亦同樣無效。

■ 參數設定例（對稱 S 形加減速）

用 0.4 秒由初速度 100pps 開始用 S 形加速到驅動速度 40kpps 實例。

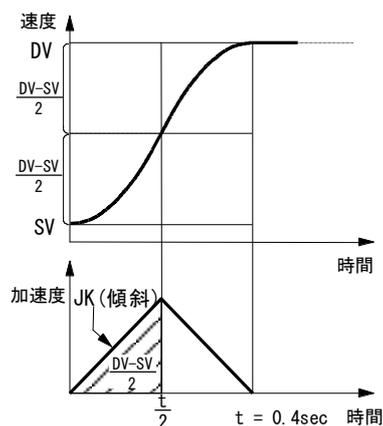


圖 2.2-11 S 形加減速驅動（對稱）例

加速時按固定加速度增加率（JK），使加速度呈線性增加。此時，加速度的線性積分值（斜線面積）為自初速度 SV 開始之速度增加值。

以全體加速時間（ $t = 0.4\text{sec}$ ）的一半時間（ $t/2$ ），求出加速度增加率（JK），速度剛好是初速度（SV）開始之驅動速度（DV）的一半速度（ $(DV-SV)/2$ ）。根據加速度增加率（JK）所增加的加速度，由時間 0 到 $t/2$ 的積分值（斜線部面積），

等於自時間 t/2 的速度之初速度上升值，因此，求 JK 的公式如下所示。

$$\frac{1}{2} \times JK \times \left(\frac{t}{2}\right)^2 = \frac{DV - SV}{2}$$

$$JK = \frac{4(DV - SV)}{t^2}$$

$$JK = \frac{4(40000 - 100)}{0.4^2} = 997,500 \text{ pps/sec}^2$$

加速度增加率	JK [pps/sec ²]
驅動速度	DV [pps]
初速度	SV [pps]
加速時間	t [sec]

因此，具有圖 2.2-11 所示加速的對稱 S 形加減速驅動參數應如下設定。

模式設定	WR3 ← 0004h	WR3 暫存器的模式設定
加速度增加率	JK = 997500	
加速度	AC = 536870911	設為最大值 (1FFF FFFFh)
初速度	SV = 100	
驅動速度	DV = 40000	
移動脈波數	TP = 27500	定量脈波驅動時設定

■ 局部 S 形加減速

針對加減速驅動，於直線加速和減速區間，僅加速和減速之開始與結束的部分，也可以使用 S 形滑順的速度曲線進行加減速。速度參數中加速度、減速度設定不要使用最大值，而是設為直線加減速區間之加速度、減速度。如圖 2.2-12 之區間 b,f 是直線加減速區間、區間 a,c,e,g 為 S 形加減速區間。

於區間 a，加速度由 0 到指定的加速度設定值為直線增加，速度曲線呈 2 次拋物線。若達到指定加速度，加速度即維持該值，此時速度曲線變成區間 b 的線性加速。指定的驅動速度與現在速度之差，比在加速度增加中消耗的速度部分為少，加速度是以指定加速度增加率減少，區間 c 的速度曲線呈反向拋物線。減速時也一樣，產生局部 S 形減速曲線。

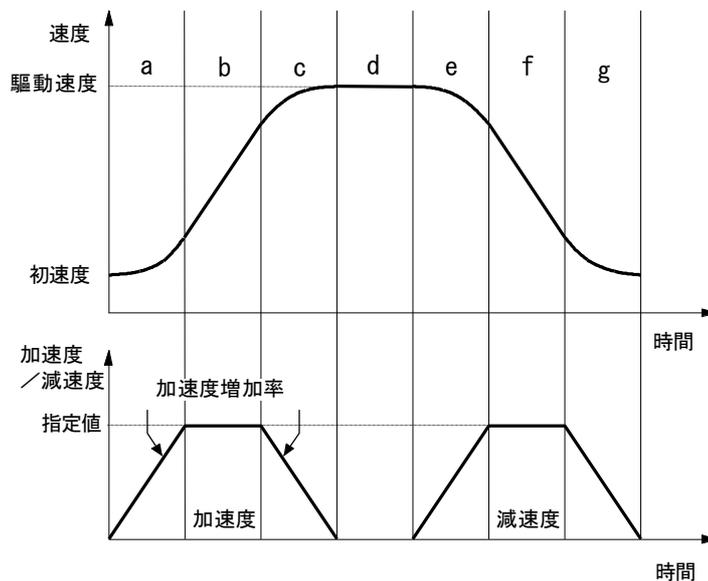


圖 2.2-12 局部 S 形加減速驅動

■ 參數設定例 (局部 S 形加減速)

下圖所示為以 0.2 秒在 10kpps 之前拋物線加速，在 0.2 秒間由 10kpps 到 30kpps 線性加速，以 0.2 秒將剩餘的 30kpps 到 40kpps，呈現拋物線加速之局部 S 形加速例。

為便於計算，忽視初速度 0。

於一開始到 10kpps 的拋物線加速中，加速度到 0.2 秒之前呈線性增加。此時的加速度積分值（斜線面積），相當於以一開始的拋物線加速拉升的速度 10kpps。

因而，0.2 秒時點的加速度，變成 $10k \times 2 / 0.2 = 100kpps/sec$ ，加速度增加率變成 $100k / 0.2 = 500kpps/sec^2$ 。

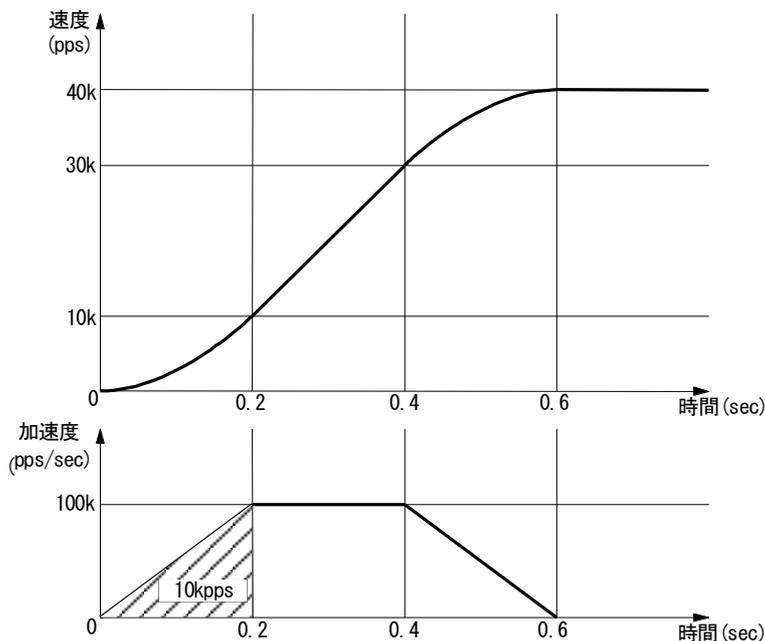


圖 2.2-13 局部 S 形加減速驅動例

因實際初速度不能設定為 $SV=0$ ，故初速度 SV 設定大於 0 的值。部分 S 形加減速之初速度 SV 設定值，以加速度 AC 設定值的平方根以上為目標。

據此，具備如圖 2.2-13 所示具有加速之對稱部分 S 形加減速驅動參數設定例如下所示。

模式設定	WR3 ← 0004h	WR3 暫存器的模式設定
加速度增加率	JK = 500000	拋物線 (S 形) 加速區間的加速度增加率設定
加速度	AC = 100000	直線加速區間的加速度設定
初速度	SV = 400	
驅動速度	DV = 40000	
移動脈波數	TP = 40000	定量脈波驅動時設定

2.2.5 非對稱 S 形加減速

S 形加減速驅動時，藉由個別設定加速度增加率與減速度增加率，可作出非對稱 S 形曲線。但非對稱 S 形加減速之定量脈波驅動的情況，不同於對稱 S 形加減速驅動，無法自動減速，因此，必須手動指定減速點。此外，定量脈波驅動的三角波形防止機能（1/12 法則）也無法發揮作用，故必須視加/減速度增加率、定量脈波驅動的輸出脈波數，設定驅動速

度。

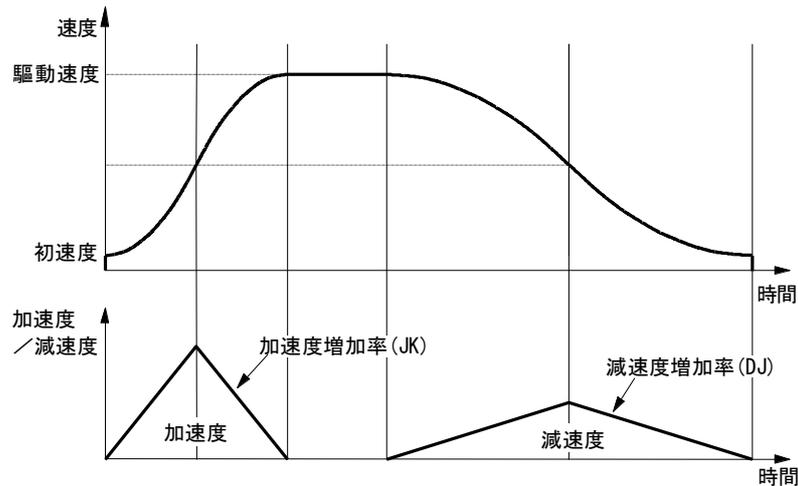


圖 2.2-14 非對稱 S 形加減速驅動

執行非對稱 S 形加減速驅動時，WR3 暫存器的 D2~0 位元應如下設定。並必須設定以下參數。

表 2.2-8 模式設定：非對稱 S 形加減速

模式設定位元	記号	設定值	備註
WR3/D0	MANLD	1	手動減速
WR3/D1	DSNDE	1	於減速時使用減速度設定值、減速度增加率設定值
WR3/D2	SACC	1	S 形加減速

表 2.2-9 設定參數：非對稱 S 形加減速

參數名	記号	備註
加速度增加率	JK	
減速度增加率	DJ	
加速度	AC	設為最大值：536, 870, 911 (1FFF FFFFh)
減速度	DC	設為最大值：536, 870, 911 (1FFF FFFFh)
初速度	SV	
驅動速度	DV	
輸出脈波數	TP	連續脈波驅動不需要
手動減速點	DP	<ul style="list-style-type: none"> 設定值為定量驅動的輸出脈波數減去減速消費脈波數 連續脈波驅動不需要

■ 參數設定例（非對稱 S 形加減速）

例如加速時是於 0.2 秒由初速度 (SV) 100pps 加速到驅動速度 (DV) 40kpps，減速時是於 0.4 秒由驅動速度 (DV) 40kpps 減速到初速度 (SV) 100pps 的非對稱 S 形加減速例。本驅動是移動脈波數 (TP) 為 20,000 脈波的相對位置驅動。

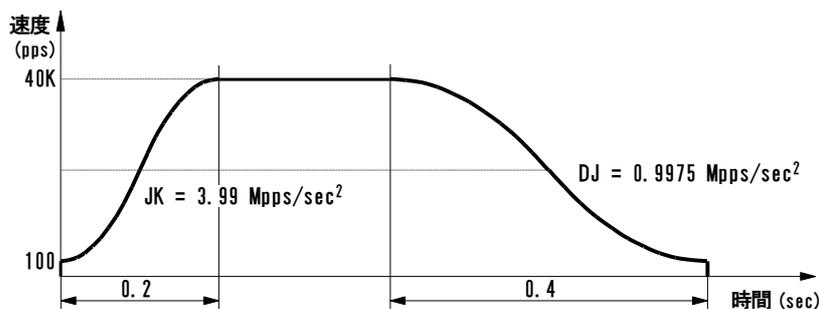


圖 2.2-15 非對稱 S 形加減速驅動例

使用前述對稱 S 形加減速參數設定例公式，求出加速度增加率、減速度增加率。

$$\text{加速度增加率 } JK = \frac{4(40000 - 100)}{0.2^2} = 3.99 \text{ Mpps/sec}^2$$

$$\text{減速度增加率 } DJ = \frac{4(40000 - 100)}{0.4^2} = 0.9975 \text{ Mpps/sec}^2$$

其次，因非對稱 S 形加減速無法自動減速，故手動設定減速點(DP)。手動減速點是設定定量驅動開始驅動起，至開始減速時點之輸出脈波數。相對位置驅動中，變成由移動脈波數 (TP) 扣掉減速時消耗脈波 (Pd) 後的值，故先求出減速時消耗脈波 (Pd)。

$$\text{減速消費脈波 } Pd = (DV + SV) \sqrt{\frac{DV - SV}{DJ}} = (40000 + 100) \sqrt{\frac{40000 - 100}{0.9975 \times 10^6}} = 8020$$

移動脈波數 (TP) 在 20,000 脈波相對位置驅動的減速時消耗脈波 (Pd) 為 8,020 時，手動減速點 (DP) 如下。

$$\text{手動減速點 } DP = TP - Pd = 20000 - 8020 = 11980$$

因而，本 IC 之參數設定如下。

模式設定	WR3←0007h	WR3 暫存器的模式設定
加速度增加率	JK = 3990000	
減速度增加率	DJ = 997500	
加速度	AC = 536870911	設為最大值 (1FFF FFFFh)
減速度	DC = 536870911	設為最大值 (1FFF FFFFh)
初速度	SV = 100	
驅動速度	DV = 40000	
移動脈波數	TP = 20000	
手動減速點	DP = 11980	

【注意】

- 上述求減速消耗脈波公式為理想公式，實際會依 IC 的參數設定值，產生拖拉或斷尾現象。

2.2.6 驅動脈波寬及速度精度

■ 驅動脈波的脈波比率

於 + 方向 / - 方向之驅動脈波，由驅動速度所決定的脈波週期時間，運算上會有 ±1CLK (CLK=16MHz 時為 ±62.5nsec) 的誤差，但基本上高位準與低位準會各分配到 50%。

例如下圖所示，若設定 DV = 1000pps，驅動脈波會輸出高位準寬 = 500µsec、低位準寬 = 500µsec、週期 = 1.00 msec 的

脈波。

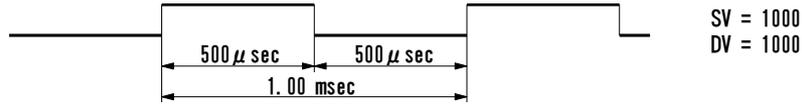


圖 2.2-16 驅動脈波输出的 Hi/Low 位準寬 (1000pps)

在加減速驅動時，加速時在輸出單一驅動脈波之同時，驅動速度也在上升，因此低位準之脈波寬會短於高位準。反之於減速時，低位準脈波寬會長於高位準。

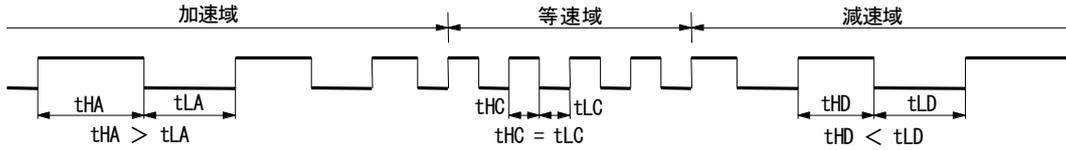


圖 2.2-17 加減速驅動時的驅動脈波寬比較

■ 驅動速度的精度

本 IC 用以產生驅動脈波的迴路，均是利用輸入時脈訊號 (CLK) 動作。CLK 輸入若為標準的 16MHz，欲產生特定頻率的驅動脈波的情況，萬一擬製作無抖動的均一式頻率驅動脈波，只能產生 CLK 頻率整數倍的週期的頻率。例如只能輸出 2 倍週期：8.000 MHz、3 倍：5.333 MHz、4 倍：4.000 MHz、5 倍：3.200 MHz、6 倍：2.667 MHz、7 倍：2.286 MHz、8 倍：2.000 MHz、9 倍：1.778 MHz、10 倍：1.600 MHz、... 的頻率，無法輸出非整數倍週期的頻率。因此無法任意設定驅動速度。故本 IC 藉由以下實例方式，使可輸出任意驅動速度。

例如，若欲輸出驅動速度 DV = 980kpps 驅動脈波，由於該週期並非 CLK 週期的整數倍，無法以均一頻率輸出 980kpps。於是如下圖所示般，將 CLK 週期的 16 整數倍 1000kpps 之頻率，與 17 整數倍 941kpps 之頻率，合成輸出。980kpps 週期是 CLK (16MHz) 週期的 16.326 倍，故將 CLK 的 16 倍週期脈波，與 17 倍週期的脈波，以 674 : 326 之比率輸出，使平均單位時間之平均週期變成 CLK 的 16.326 倍。



圖 2.2-18 CLK 周期 980kpps 驅動脈波的周期

藉此方式，可輸出高精確度的指等速度驅動脈波。針對指等速度實際輸出驅動脈波，其速度精確度可控制在±0.1%以下。

若以示波器觀測驅動脈波，在驅動脈波週期並非 CLK 週期的整數倍時，如上圖所示，脈波週期會產生 1CLK (62.5nsec) 的時間差，雖看起來像是抖動，但本 IC 是藉此 1CLK 的時間差，製作出正確的驅動速度。在馬達轉動的情況，該 1CLK 時間差會被負載慣性所吸收，使用上幾無問題。

2.3 位置管理

本 IC 配備 2 個管理現在位置所需的 32 位元上下計數器（理論位置計數器、實際位置計數器）。預先在多目的暫存器設定值，即可與現在位置進行大小比較。此外，針對理論位置計數器、實際位置計數器，可設定軟體極限和可變環機能。

2.3.1 理論位置計數器及實際位置計數器

理論位置計數器在 IC 內部，是計算 + 方向 / - 方向之驅動輸出脈波。+ 方向以 1 脈波 1 計數累加，- 方向以 1 脈波 1 計數遞減。

實際位置計數器是計算由編碼器等外部輸入脈波。輸入脈波可在 2 相訊號和獨立 2 個脈波上下觸發訊號模式中選擇。請參 2.12.3 項。

兩計數器均隨時可由 CPU 寫入 / 讀取資料。計數範圍是 -2,147,483,648 ~ +2,147,483,647。負值取 2 的補數顯示。理論位置計數器、實際位置計數器在重置時其值為亂數。

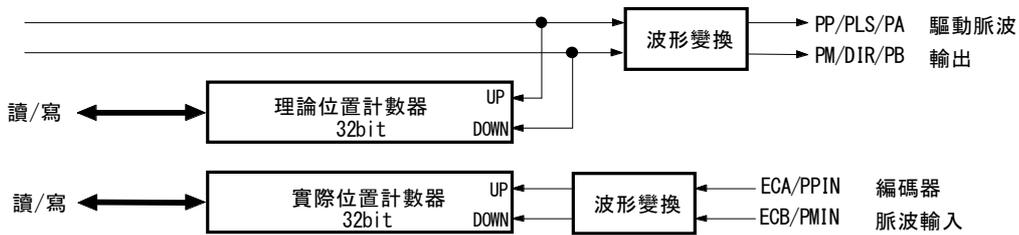


圖 2.3-1 位置計數器部機能方塊圖

2.3.2 位置比較

本 IC 配備 4 個比較理論位置計數器與實際位置計數器，其現在位置與大小的多目的暫存器。於驅動中亦可讀取多目的暫存器與理論 / 實際位置計數器大小關係之狀態。比較條件符合時，輸出訊號可產生中斷或同步動作起動。

多目的暫存器之比較機能請參 2.4 節。

2.3.3 軟體極限

理論位置計數器與實際位置計數器，可設定軟體極限。軟體極限之設定標的是在 WR2 暫存器的 D14 位元設定。設定軟體極限的 2 個 32 位元暫存器 (SLMT+, SLMT-)，分別設在 + 方向 / - 方向的軟體極限位置。

軟體極限設定標的所在的理論 / 實際位置計數值，若大於 SLMT+ 暫存器值，則驅動減速停止 / 即停，RR2 暫存器的 D0 位元上升變成 1。此錯誤狀態，在執行 - 方向之驅動指令，理論 / 實際位置計數值小於 SLMT+ 暫存器值時，即可解除。SLMT- 暫存器 - 方向也一樣。

+ 方向軟體極限在「位置計數值 \geq SLMT+ 暫存器值」時出現軟體極限錯誤，- 方向軟體極限在「位置計數值 $<$ SLMT- 暫存器值」時出現軟體極限錯誤。

圖 2.3-2 是軟體極限機能為有效，設定 SLMT+ 暫存器值=10000、SLMT- 暫存器值=-1000 實例。

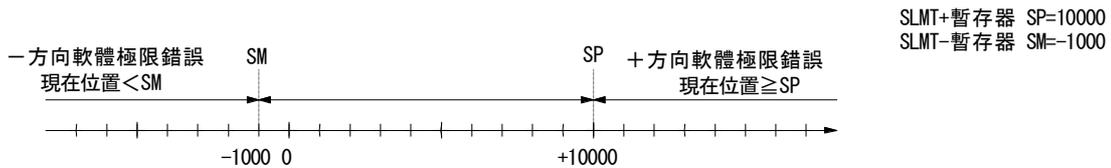


圖 2.3-2 軟體極限值設定及軟體極限錯誤

軟體極限機能可設定有效作用／無效。在 WR2 暫存器的 D13 位元進行設定。軟體極限的停止動作可選擇減速停止／即停。在 WR2 暫存器的 D15 位元設定。可隨時寫入 SLMT+暫存器與 SLMT-暫存器。
軟體極限機能在重置時無效。重置時的 SLMT+暫存器、SLMT-暫存器值為亂數。

2.3.4 位置計數器的可變環

理論位置計數器與實際位置計數器，是 32 位元長度的 UP/DOWN 環形計數器。因而，由一般 32 位元長度最大值 FFFF FFFFh 往+方向上數，則值回到 0。且由 0 值往-方向下數，即回到 FFFF FFFFh。

可變環機能是可將此環形計數器環圈最大值，設成任意值的機能。在定位軸直線運動時，欲對迴轉 1 圈會返回原位的迴轉軸，進行位置管理是很方便的機能。

可變環的大小，亦即理論／實際位置計數器的最大值，可在 $\sim 2,147,483,647$ ($1\sim 7FFF\ FFFFh$) 的範圍設定任意值。可變環機能是將理論位置計數器最大值 (LX)，以理論位置計數器最大值寫入指令 (0Eh) 設定，將實際位置計數器最大值 (RX)，以實際位置計數器最大值寫入指令 (0Fh) 設定供使用。

重置時的理論位置計數器最大值 (LX)、實際位置計數器最大值 (RX) 為 FFFF FFFFh。不使用可變環機能時維持其初值即可。

■ 可變環形計數器設定例

以 10,000 脈波迴轉 1 次的迴轉軸如下設定。

- ① 理論位置計數器最大值 (LX) 設為 9,999(270Fh)。
- ② 同時使用實際位置計數器時，實際位置計數器最大值 (RX) 設為 9,999(270Fh)。

此時的計數動作

- 往+方向上數時 : ...→9998→9999→0→1→...
- 往-方向下數時 : ...→1→0→9999→9998→...

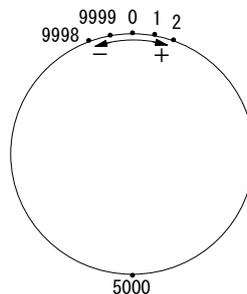


圖 2.3-3 位置計數器環最大值 9999 的動作

【注意】

- 可設為可變環機能位置計數器的最大值，是在 $1\sim 2,147,483,647$ ($1\sim 7FFF\ FFFFh$) 的範圍。附符號的 32 位元暫存器不能設成負值 ($8000\ 0000h\sim FFFF\ FFFEh$)。
- 設定理論位置計數值 (LP) 與實際位置計數值 (RP) 時，均無法設成理論位置計數器最大值 (LX、實際位置計數器最大值 (RX) 範圍外的值。

2.4 補間

本IC可選擇在任意4軸中的軸做直線補間、圓弧補間、螺旋補間、位元補間驅動。
或是、本IC也可以在複數晶片中使用多軸直線補間。
有關補間機能的詳細說明請參考第3章。

2.5 多目的暫存器

本IC備有4個(MR3~0)帶符號的32位元多目的暫存器。

多目的暫存器可作為現在位置與速度等之比較標的，可用於與該些值比較。可將比較結果以訊號輸出，比較後的大小關係變化，作為同步動作的起動要素，和作為產生中斷要素使用。此外，就同步動作的動作而言，可將多目的暫存器預設值，設成新的速度和移動脈波數的載入，或在多目的暫存器上取得現在位置和現在速度的儲存。

藉由可隨時對多目的暫存器進行資料寫入／讀取，4個多目的暫存器分別利用多目的暫存器設定指令(10h~13h)、多目的暫存器讀取指令(34h~37h)執行。
重置時多目的暫存器值為亂數。

2.5.1 比較對象及比較條件

就多目的暫存器MR3~0之比較標的而言，可設定理論位置計數值、實際位置計數值、現在速度值、現在計時器值。與比較標的之比較條件式可由□，>，=，<4種中選擇。

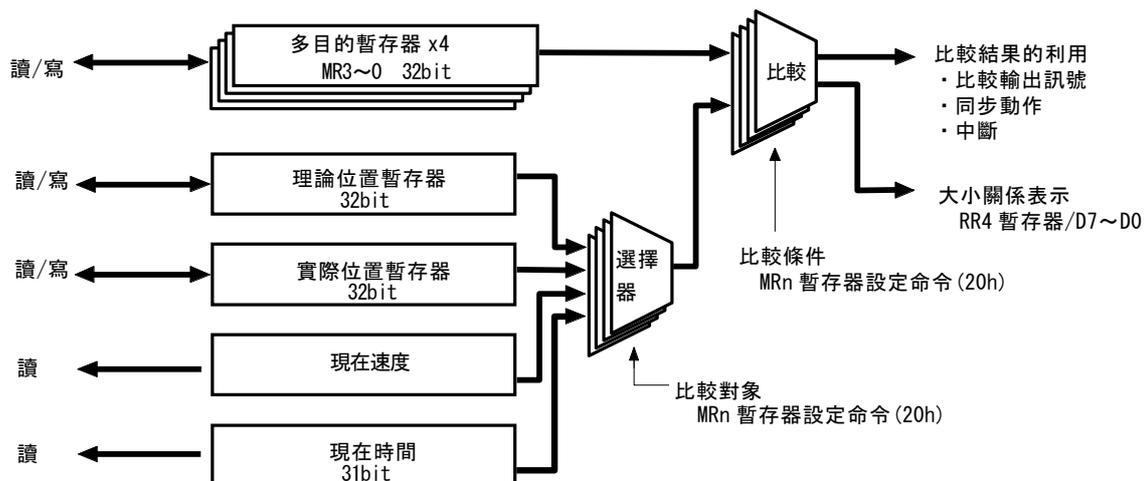


圖 2.5-1 多目的暫存器及比較機能

針對4個多目的暫存器，可分別選設比較標的與比較條件。比較標的、比較條件之設定，是藉由多目的暫存器模式指令(20h)設定。設定WR6寫入資料暫存器指定的位元，並設定將多目的暫存器模式設定指令碼(20h)寫入WR0暫存器。多目的暫存器模式設定狀態，可以多目的暫存器模式設定讀取指令(40h)。

多目的暫存器模式設定指令 (20h)

WR6	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
	M3C1	M3C0	M3T1	M3T0	M2C1	M2C0	M2T1	M2T0	M1C1	M1C0	M1T1	M1T0	M0C1	M0C0	M0T1	M0T0
	MR3 比較條件		MR3 比較對象		MR2 比較條件		MR2 比較對象		MR1 比較條件		MR1 比較對象		MR0 比較條件		MR0 比較對象	

表 2.5-1 比較對象的設定 (n:0~3)

MnT1 位元	MnT0 位元	MRn 比較對象
0	0	理論位置計數器 (LP)
0	1	實際位置計數器 (RP)
1	0	現在速度值 (CV)
1	1	現在計時器值 (CT)

表 2.5-2 比較條件的設定 (n:0~3)

MnC1 位元	MnC0 位元	MRn 比較條件
0	0	比較對象 \geq MRn
0	1	比較對象 $>$ MRn
1	0	比較對象 = MRn
1	1	比較對象 $<$ MRn

【注意】

設定比較標的為「現在速度值 (CV)」、比較條件為「比較標的 = MRn」時，在以直線或 S 形加減速驅動之加速度、減速度逾 4,194,304 (400000h) pps/sec 的情況，比較結果有時不為真 (有效)。

比較標的在以「現在速度值 (CV)」加速度，減速度超過該值時，就比較條件而言，請勿使用「比較標的 = MRn」，應使用「比較標的 \square MRn」等其他條件。

■ 設定例：理論位置計數器之比較

理論位置計數值在 500,000 以上時，比較結果為真之比較設定如下。

本例中與理論位置計數器進行大小比較的暫存器使用 MR0。

WR6 \leftarrow A120h		
WR7 \leftarrow 0007h	MR0 值 : 500,000	\hookrightarrow 設定 MR0 的比較值
WR0 \leftarrow 0010h		
WR6 \leftarrow 0000h	D3, D2 : 0, 0 比較條件 : \geq D1, D0 : 0, 0 比較對象 : 理論位置計數器 (LP)	\hookrightarrow MR0 的比較對象、比較條件
WR0 \leftarrow 0020h	多目的暫存器模式設定寫入	

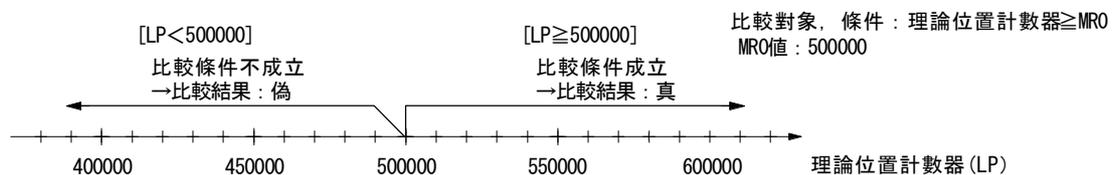


圖 2.5-2 多目的暫存器及理論位置計數器的比較例

2.5.2 比較結果的用途

多目的暫存器與比較標的之比較結果，可供作比較輸出訊號、同步動作之起動要素、產生中斷要素使用。比較結果利用機能與其動作如下表所示。

表 2.5-3 比較結果的利用及動作

機能	對象	動作
比較輸出訊號	PI07~4 輸出訊號	比較結果為真時輸出訊號 Hi
同步動作的起動要因	同步動作 SYNC3~0	比較結果為真變化時同步動作起動
中斷發生要因	中斷發生機能	比較結果為真變化時 \square 中斷

■ 比較輸出訊號

將多目的暫存器之比較結果，作為比較輸出訊號輸出。多目的暫存器與比較標的之關係，符合指定的比較條件時，比較輸出訊號高位準輸出，未符合比較條件時，比較輸出訊號低位準輸出。

多目的暫存器 MR3~0 之比較輸出，分別對應比較輸出訊號 PIO7~4 輸出。PIO7~4 訊號兼作泛用輸出入訊號等其他訊號之用。為將這些供作比較輸出端子使用，必須預先在 PIO 訊號設定 1 指令(21h)，所使用的 PIO7~4 訊號機能，設定於比較輸出訊號。

表 2.5-4 多目的暫存器相對應之比較輸出訊號及設定位元

多目的暫存器	比較輸出訊號	PIO 訊號設定 1 指令 (21h) WR6 暫存器的設定位元
MR0	PI04	WR6/D9 , 8 : 1, 1
MR1	PI05	WR6/D11, 10 : 1, 1
MR2	PI06	WR6/D13, 12 : 1, 1
MR3	PI07	WR6/D15, 14 : 1, 1

泛用輸出入 PIO 訊號詳細內容請參 2.9 節。

■ 設定例 比較輸出訊號

設定驅動中的現在速度超過 5,000pps 時，PIO5 輸出訊號輸出 Hi、在 5,000pps 以下時，PIO5 輸出訊號輸出 Low。

WR6 ← 1388h		
WR7 ← 0000h	MR1 值 : 5,000	↔ 設定 MR1 的比較值
WR0 ← 0011h		
WR6 ← 0060h	D7, D6 : 0, 1 比較條件 : > D5, D4 : 1, 0 比較對象 : 現在速度 (CV)	↔ 設定 MR1 的比較對象、比較條件
WR0 ← 0020h	多目的暫存器模式設定寫入	
WR6 ← 0C00h	D11, D10 : 1, 1 PIO5 機能 : MR1 比較輸出	↔ PIO5 訊號的機能設定
WR0 ← 0021h	PIO 訊號設定 1 寫入	

■ 同步動作起動

藉由多目的暫存器比較結果，可起動同步動作。多目的暫存器與比較標的之關係，在符合指定的比較條件產生變化時，起動同步動作。於設定同步動作有效的瞬間，若所有比較條件均已符合的情況下，同步動作於當時並不起動，等到有進入不符合比較條件的狀態，再變成符合比較條件時，才會起動同步動作。

藉由多目的暫存器 MR3~0 比較起動的同步動作，可作為其分別對應的同步動作組 SYNC3~0 之起動要素設定。為將多目的暫存器之比較結果，作為同步動作之起動要素使用，必須預先以同步動作 SYNC0,1,2,3 設定指令 (26h,27h,28h,29h)，將所使用的同步動作組之起動要素，設定為「MRn 比較變為真」(起動要素碼：01h)，並將以同步動作有效指令 (81h~8Fh) 所使用的同步動作集，設定為有效。

表 2.5-5 多目的暫存器對應的同步動作組與設定指令

多目的暫存器	同步動作組	設定起動要因 同步動作組指令
MR0	SYNC0	同步動作 SYNC0 設定指令 (26h)
MR1	SYNC1	同步動作 SYNC1 設定指令 (27h)
MR2	SYNC2	同步動作 SYNC2 設定指令 (28h)
MR3	SYNC3	同步動作 SYNC3 設定指令 (29h)

同步動作 SYNC0,1,2,3 設定指令，除起動要素以外，亦可進行動作和重複設定等同步動作相關設定。同步動作的機能和設定的詳細內容，請參 2.7 節。

■ 設定例 同步動作起動

在 10 秒計時當中，於計時開始 5 秒後，可以同步動作 SYNC2，設定起動相對位置之驅動。
據此設定的計時器，於計時開始 5 秒後起動同步動作，10 秒後計時到的計時器。

WR6 ← 4B40h			
WR7 ← 004Ch	MR2 值 : 5,000,000		↩ 設定 MR2 比較值
WRO ← 0012h	(5 秒 = 5,000,000 μ sec)		
WR6 ← 9680h			
WR7 ← 0098h	計時器值 : 10,000,000		↩ 計時器時間設定為 10 秒
WRO ← 0016h	(10 秒 = 10,000,000)		
WR6 ← 0300h	D11, D10 : 0, 0 比較條件 : \geq		
	D9, D8 : 1, 1 比較對象 : 現在計時器值 (CT)		↩ 設定 MR2 的比較對象、比較條件
WRO ← 0020h	多目的暫存器模式設定寫入		
WR6 ← 00A1h	起動要因編碼 01h : MR2 比較變真 動作編碼 0Ah : 相對位置驅動起動		↩ 同步動作 SYNC2 的機能設定
WRO ← 0028h	同步動作 SYNC2 設定寫入		
WRO ← 0084h	同步動作 SYNC2 有效設定指令		↩ 設定同步動作 SYNC2 有效

- 相對位置驅動所需參數必須預先設定妥。
相對位置驅動請參 2.1.1 項。

■ 中斷發生

藉由多目的暫存器之比較結果，即可做到產生中斷。多目的暫存器與比較標的之關係，在變成符合指定比較條件時產生中斷。於設定中斷動作有效的瞬間，若所有比較條件均已符合的情況下，中斷動作於當時並不起動，等到有進入不符合比較條件的狀態，再變成符合比較條件時，才會起動中斷動作。

為能藉由比較多目的暫存器 MR3~0 產生中斷，預先針對 WR1 模式暫存器 1 的各中斷要素之許可/禁止位元，將使用的多目的暫存器比較，設為許可中斷。產生中斷時產生中斷要素之確認，以 RR1 狀態暫存器 1 之各產生中斷要素位元確認。

表 2.5-6 多目的暫存器對應之比較中斷許可、確認位元

多目的暫存器	中斷許可設定位元	中斷發生要因確認位元
MR0	WR1/D0 : 1	RR1/D0 : 1
MR1	WR1/D1 : 1	RR1/D1 : 1
MR2	WR1/D2 : 1	RR1/D2 : 1
MR3	WR1/D3 : 1	RR1/D3 : 1

中斷之詳細內容請參 2.11 節。

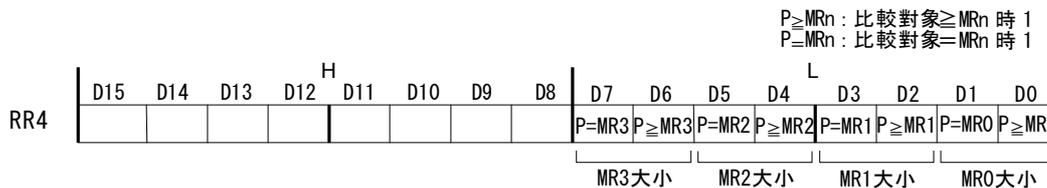
■ 設定例 中斷發生

設定實際位置計數值若通過 30,000 即讓產生中斷。

WR6 ← 7530h			
WR7 ← 0000h	MR3 值 : 30,000		↩ 設定 MR3 比較值
WRO ← 0012h			
WR6 ← 0060h	D15, D14 : 1, 0 比較條件 : =		
	D13, D12 : 0, 1 比較對象 : 實際位置計數器 (RP)		↩ 設定 MR3 的比較對象、比較條件
WRO ← 0020h	多目的暫存器模式設定寫入		
WR1 ← 0008h	中斷發生要因 : 許可 MR3 比較變真		↩ 設定中斷發生要因

2.5.3 比較大小狀態的顯示

以多目的暫存器 MR3~0 與多目的暫存器模式設定指令(20h)，所設定的各比較標的大小之比較結果，可利用 RR4 暫存器確認。與以多目的暫存器模式設定指令(20h)設定的比較條件無關，而是顯示比較標的與多目的暫存器現在的大小關係。



按照 RR4 暫存器位元讀取值，多目的暫存器與比較標的之大小關係如下表。

表 2.5-7 比較大小關係及 RR4 暫存器位元的狀態 (n:0~3)

P=MRn (D7, 5, 3, 1)	P ≥ MRn (D6, 4, 2, 0)	比較對象及 MRn 的大小關係
0	1	比較對象 > MRn
x	1	比較對象 ≥ MRn
1	x	比較對象 = MRn
x	0	比較對象 < MRn

於 2.4.1 節 ■設定例 進行比較設定的狀態，若讀取 RR4，則比較標的所在理論位置計數值之 RR4 的位元狀態如下圖所示。

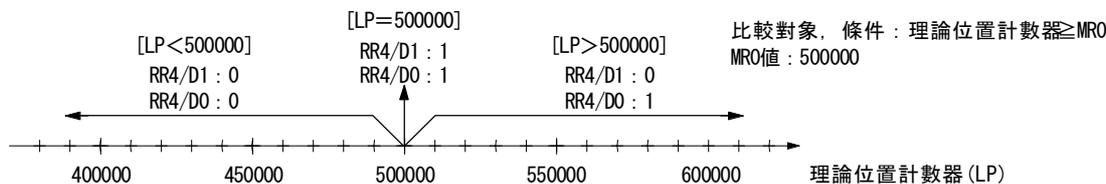


圖 2.5-3 比較大小關係及 RR4 暫存器位元的狀態 例

2.5.4 利用同步動作載入／儲存參數值

藉使用同步動作將多目的暫存器的預設值，作為新速度和移動脈波數之設定（載入），或在多目的暫存器上取得（儲存）現在位置和速度。

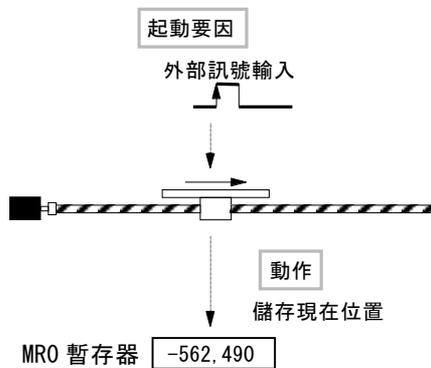


圖 2.5-4 參數值的儲存使用例

經由同步動作可由多目的暫存器載入的參數值有 7 種，可儲存在多目的暫存器的參數值有 5 種。針對起動的同步動作 SYNC3~0 所對應的多目的暫存器 MR3~0，載入／儲存參數值。

為執行藉由同步動作載入／儲存參數值，以同步動作 SYNC0,1,2,3 設定指令（26h,27h,28h,29h），必須設定所使用的同步動作組動作上，載入／儲存各參數值所需的動作碼。此外，於同步動作有效指令（81h~8Fh）所使用的同步動作組，必須設為有效。

表 2.5-8 藉同步動作載入／儲存可能的參數

動作編碼 (Hex)	參數值的設定（載入）	動作編碼 (Hex)	現在值的取得（儲存）
01	驅動速度 (DV)	05	理論位置計數器 (LP)
02	移動脈波數／終點 (TP)	06	實際位置計數器 (RP)
03	分割脈波設定 1 (SP1)	07	現在計時器 (CT)
04	理論位置計數器 (LP) (SYNC0)	08	現在速度 (CV) (SYNC0)
	實際位置計數器 (RP) (SYNC1)		現在加減速度 (CA) (SYNC1)
	初速度 (SV) (SYNC2)		
	加速度 (AC) (SYNC3)		
0F	移動脈波數 (TP) 設定、 且起動相對位置驅動		
10	終點 (TP) 設定、 且起動絕對位置驅動		

動作編碼 (Hex)：設定同步動作 SYNC0, 1, 2, 3 設定指令的資料寫入暫存器編碼

使用同步動作載入／儲存參數值於多目的暫存器之詳細內容請參 2.7 節。

2.6 自動原點復歸

本 IC 無需透過 CPU，即具有自動執行高速原點搜尋 → 低速原點貼近搜尋 → 編碼器 Z 相搜尋 → 偏移移動等一連串原點復歸順序的機能。自動原點復歸是依下表所示之步驟 1 到步驟 4 依序執行。各步驟可選擇執行／不執行。選擇不執行時，該步驟即不執行，移至下一步驟。此外，在各步驟進行搜尋方向、搜尋訊號之模式設定。步驟 1、4 為驅動速度設定高速的搜尋動作或移動。步驟 2、3 是搜尋原點速度設定低速的搜尋動作。此外，步驟 2、3 可設定訊號檢出時，DCC(清除偏差計數器)輸出及清除實際位置/理論位置計數器。於各步驟結束時，步驟間可插入計時器。

表 2.6-1 自動原點復歸的內容

步驟編號	動作	搜尋速度	檢出訊號
步驟 1	高速原點搜尋	驅動速度 (DV)	可選擇STOP0, STOP1, 極限
步驟 2	低速原點搜尋	原點檢出速度 (HV)	可選擇STOP1, 極限
步驟 3	低速 Z 相搜尋	原點檢出速度 (HV)	STOP2
步驟 4	高速偏移移動	驅動速度 (DV)	無

一般原點復歸所使用的訊號搜尋動作包羅萬象。如下例所示，使用近原點訊號與原點訊號這 2 種感測器進行時，和單獨使用原點訊號、或僅使用單方向的極限訊號的方法。

(1) 使用近原點訊號(STOP0)與原點訊號(STOP1)進行時實例

在指定方向以高速尋找近原點的訊號，找到近原點訊號後減速停止。再以低速搜尋原點訊號，找到原點訊號後立即停止。

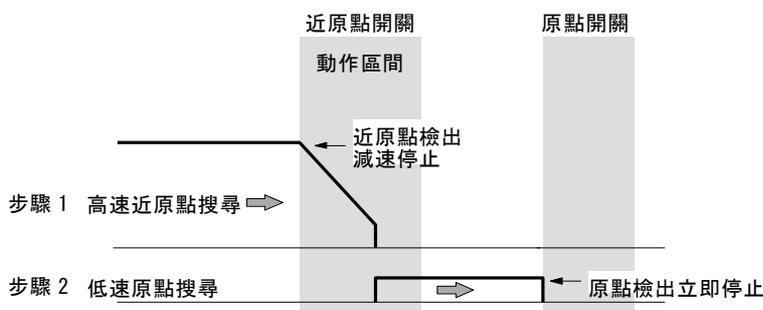


圖 2.6-1 自動原點復歸 例 1

(2) 僅以原點訊號(STOP1)或極限訊號(LMTP/LMTM)進行時實例

在指定方向以高速搜尋原點或極限訊號，找到訊號後減速停止。再以反方向移動到離開訊號有效區間。然後再以低速回到原點訊號位置，找到原點訊號後立即停止。搜尋訊號選擇極限訊號時，檢出方向為極限訊號。

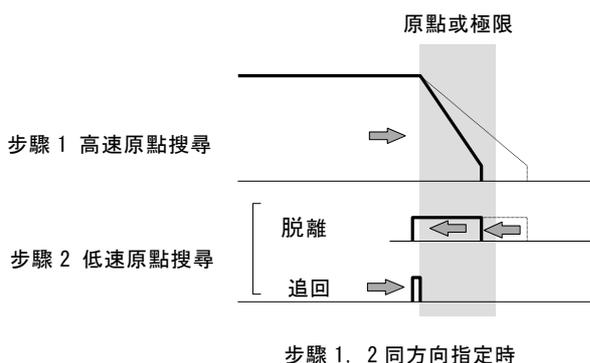


圖 2.6-2 自動原點復歸 例 2

本 IC 為也能支援此類各式原點復歸，備有若干種模式設定。

2.6.1 各步驟動作

各步驟均能以模式設定方式，指定執行與否、或搜尋的+/-方向、或檢出的訊號。若指定不執行，即不執行該步驟，進入下個步驟。

■ 步驟1 高速原點搜尋

以驅動速度（DV）所設定的速度，往指定方向，搜尋指定訊號直到訊號有效之前，輸出驅動脈波。搜尋的訊號可選擇STOP0、STOP1、極限訊號。若選擇極限訊號時，變成搜尋方向的極限訊號。為執行高速搜尋動作所需，將驅動速度（DV）設定在高於初速度（SV）的值。進行加減速驅動，指定訊號若有效，即減速停止。

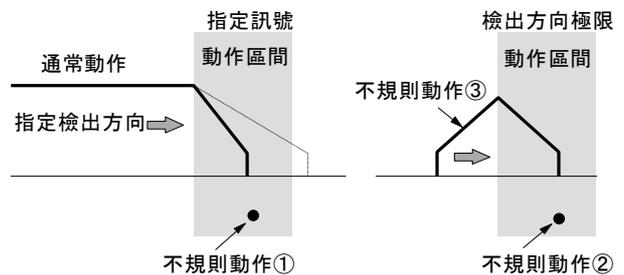


圖 2.6-3 步驟 1 的動作

不規則動作

- 步驟1開始前所指定的訊號已經有效。 → 進入步驟2。
- 搜尋訊號指定STOP0、STOP1時，步驟1開始前，搜尋方向的極限訊號已經有效。 → 進入步驟2。
- 搜尋訊號指定STOP0、STOP1時，搜尋方向的極限訊號在執行中有效。 → 停止驅動，進入步驟2。

步驟1的其他動作

步驟1結束時，可起動步驟間計時器。詳細內容請參 2.6.3 項。

【注意】

- 步驟1係進行高速搜尋，因此，搜尋訊號指定為極限訊號時，將極限停止模式設為減速停止(WR2/D12:1)。WR2暫存器請參 6.6 節。

■ 步驟2 低速原點搜尋

步驟2的正常動作，是以原點檢出速度（HV）所設定的速度，往指定方向，搜尋指定訊號，直到訊號有效前，輸出驅動脈波。搜尋訊號由STOP1、極限訊號中選擇。選定極限時，變成搜尋方向的極限訊號。為進行低速搜尋動作，將原點檢出速度（HV），設為低於初速度（SV）的值。進行等速驅動，指定訊號找到後會立即停止。

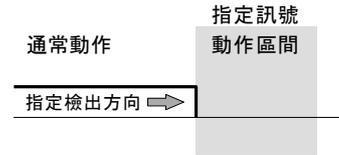


圖 2.6-4 步驟 2 的動作

不規則動作

□在步驟2開始前指定訊號已經有效。

〔動作〕在指定訊號無效為止，往指定方向之反方向以原點檢出速度（HV）移動。當指定訊號變成無效，開始執行步驟2的正常動作。

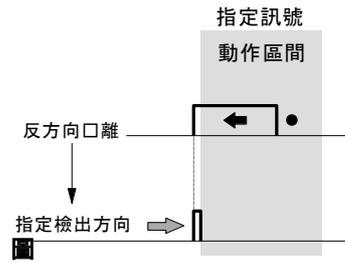


圖 2.6-5 步驟 2 不規則動作①

□指定搜尋STOP1訊號，在步驟2開始前，搜尋方向的極限訊號已經有效。

〔動作〕STOP1訊號有效前，以指定搜尋方向之反方向用驅動速度（DV）移動。直到STOP1訊號有效後，再以原點檢出速度（HV）移動，直到STOP1訊號再無效為止。當STOP1訊號再次變成無效後，開始執行步驟2的正常動作。

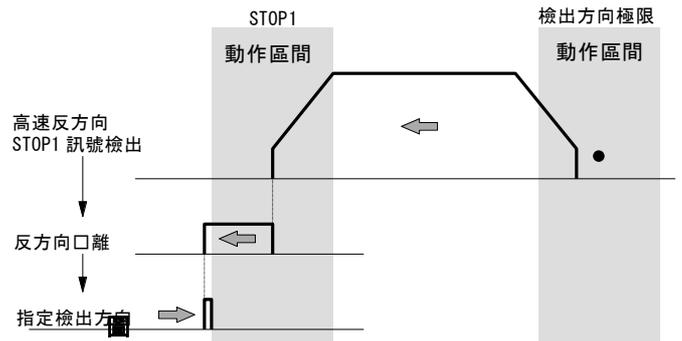


圖 2.6-6 步驟 2 不規則動作②

□指定搜尋STOP1訊號，執行中搜尋方向的極限訊號有效。

〔動作〕停止驅動，進行不規則動作□所示動作。

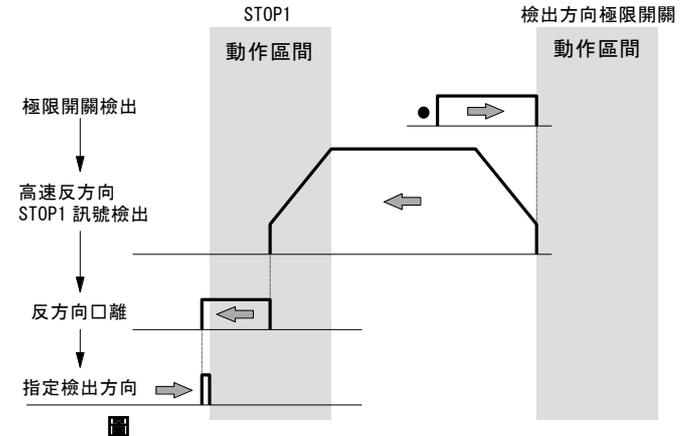


圖 2.6-7 步驟 2 不規則動作③

④步驟1與步驟2之搜尋訊號為同一個，且步驟1與步驟2之搜尋方向相同的情況，在步驟2開始前，指定訊號為無效。

〔動作〕進行不規則□所示動作。

本動作適用於迴轉軸之原點復歸。

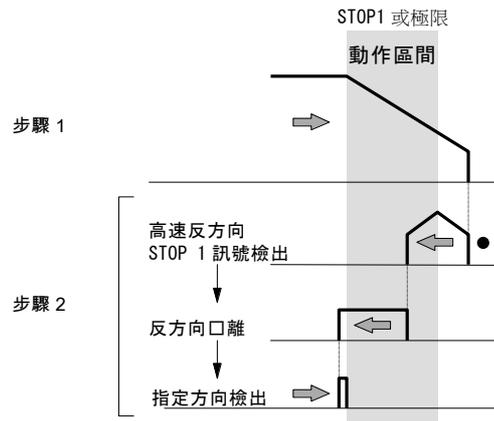


圖 2.6-8 步驟2 不規則動作④

步驟2的其他動作

在指定方向搜尋中，步驟2之搜尋訊號由無效變成有效時，可以產生清除偏差計數器(DCC)訊號的輸出，或清除實際位置計數器，理論位置計數器。但不規則動作中，在指定與反方向移動時，即使搜尋訊號變成有效，該些動作亦不作動。有關清除偏差計數器(DCC)輸出請參考 2.5.2 項。

此外，不規則動作□~□在反向脫離後，與在步驟2結束時，可起動步驟間計時器。

■ 步驟3 低速Z相搜尋

以原點檢出速度(HV)所設定的速度，於指定方向，編碼器Z相訊號(STOP2)在變成有效之前，輸出驅動脈波。為執行低速搜尋動作，將搜尋原點速度(HV)設成低於初速度(SV)的值。進行等速驅動，若編碼器Z相訊號(STOP2)有效則立即停止。

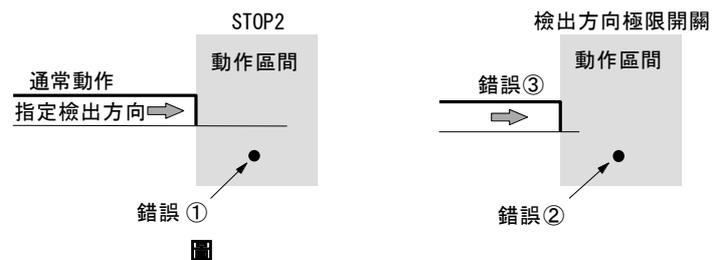


圖 2.6-9 步驟3的動作

就搜尋條件而言，亦可利用編碼器Z相訊號(STOP2)與原點訊號(STOP1)之AND條件停止。

步驟3之其他動作

編碼器Z相訊號(STOP2)在進入有效變化時，可清除實際位置計數器，理論位置計數器。實際位置計數器的清除，在STOP2進入有效，不需CPU介入，即可清除實際位置計數器。這在搜尋Z相驅動速度設為低速時，因伺服系統或機械系統反應遲鈍，致使Z相搜尋位置偏差問題發生之虞的情況中使用，相當方便。

同樣在編碼器Z相訊號(STOP2)在有效變化時，可輸出清除偏差計數器(DCC)的訊號。此外，在步驟3結束時，可起動步驟間計時器。

【注意】

- 在步驟3開始時，編碼器Z相訊號(STOP2)若已有效，就會產生錯誤，RR2暫存器的D6位元上升變為1。自動原點復歸結束。為使步驟3務必在編碼器Z相訊號(STOP2)穩定的無效狀態開始，請調整機械系統。
- 在步驟3開始前，搜尋方向的極限訊號若有效，就會產生錯誤，RR2暫存器搜尋方向的極限值錯誤位元(D2或D3)上升變為1。自動原點復歸結束。
- 在執行中，搜尋方向極限訊號若有效，搜尋動作即中斷，RR2暫存器搜尋方向的極限錯誤位元(D2或D3)上升變為1。自動原點復歸結束。

■ 步驟4 高速偏移移動

將驅動速度（DV）設定的速度，移動脈波數（TP）設定的脈波數，以相對位置驅動輸出脈波。此步驟4，一般是在欲由機械原點位置，移動到工作原點的情況使用。此外，在搜尋訊號選擇極限訊號時，將作業原點略遠離極限位置時使用。在步驟4開始前或執行中，移動方向的極限訊號變成有效，則錯誤結束，RR2 暫存器搜尋方向的極限值錯誤位元（D2或D3）上升變為1。自動原點復歸即結束。

2.6.2 輸出偏差計數器清除

步驟2或步驟3動作時，指定的搜尋訊號（步驟3在STOP2上固定）往有效正緣觸發時，可輸出清除偏差計數器(DCC)訊號。此外，可指定清除偏差計數器(DCC)訊號的脈波邏輯準位、脈波寬。設定方法詳細內容請參2.5.4項。

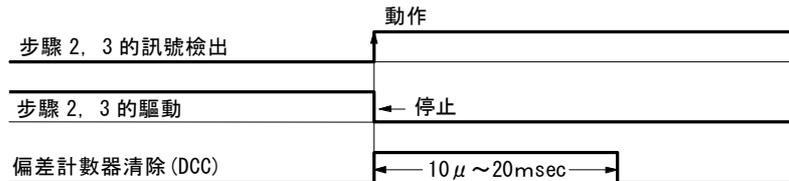


圖 2.6-10 偏差計數器清除輸出

清除偏差計數器輸出，與步驟2或步驟3的搜尋動作結束同時有效，待清除偏差計數器(DCC)脈波輸出結束，再開始下個步驟。

2.6.3 步驟間計時器

自動原點復歸的各步驟，有馬達軸逆轉設定。馬達若急劇逆轉，可能會對機械系統造成很大負載。步驟間計時器是為減輕該施加於機械系統的負載而設計。

本IC在各步驟結束時，可起動步驟間計時器。步驟2方面，在特定的不規則動作後，亦可起動步驟間計時器。可指定有無步驟間計時器與計時器值。設定方法的詳細內容請參2.6.4項。

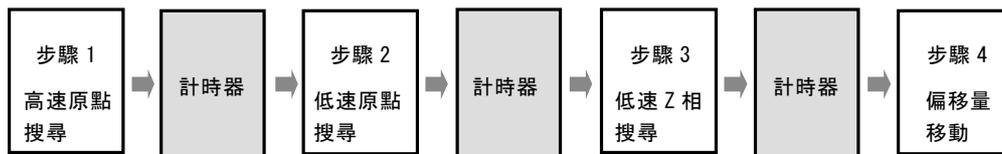


圖 2.6-11 步驟間計時器

步驟間計時器有效的情況，在各步驟結束時，步驟間計時器起動，於計時器動作後，開始下個步驟。而步驟2在發生特定不規則動作時，步驟間計時器亦起動，於計時器動作後，展開步驟2的正常動作。有關步驟2的不規則動作，請參2.6.1項。

【注意】

- 步驟間計時器無法在各步驟逐一設定。有效的情况，各步驟間與步驟 2 特定不規則動作後，所有的步驟間計時器即變成有效，指定計時器值的步驟間計時器即起動。設定為無效的情况，所有的步驟間計時器即變成無效。

2.6.4 設定搜尋速度模式的

為進行自動原點復歸，必須進行以下所述之速度參數與模式設定。

■ 速度參數的設定

表 2.6-2 速度參數的設定

速度參數	指令編碼(hex)	說明
驅動速度 (DV)	0 5	以步驟1、4之高速搜尋、移動之速度。 但於步驟2不規則動作，搜尋指定與逆向搜尋訊號時，變成該驅動速度。 為進行加減速驅動，加速度(AC)、初速度(SV)亦均需設定成合宜的值。 請參2.2.2項。
原點檢出速度 (HV)	1 4	以步驟2、3之低速搜尋之速度。 搜尋訊號有效時，為使之即停，設定成比初速度(SV)為低的值。請參2.2.1項。

■ 自動原點復歸模式設定1

自動原點復歸模式設定1，如下所示設定WR6暫存器的各位元，將自動原點復歸模式設定1指令(23h)，寫入WR0暫存器與進行模式設定。設定各步驟之執行/不執行、搜尋訊號之指定、搜尋方向與清除偏差計數器(DCC)輸出與理論位置計數器、實際位置計數器清除。

WR6	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
	S4EN	S3LC	S3RC	S3DC	S3DR	S3EN	S2LC	S2RC	S2DC	S2SG	S2DR	S2EN	S1G1	S1G0	S1DR	S1EN

① 各步驟執行/不執行的設定

各步驟執行指定位元設為1、不執行設為0。

各步驟執行/不執行之指定位元如下表。

表 2.6-3 各步驟執行/不執行的指定位元

	步驟 1	步驟 2	步驟 3	步驟 4	
執行/不執行	D0 位元	D4 位元	D10 位元	D15 位元	0: 不執行
指定位元	S1EN	S2EN	S3EN	S4EN	1: 執行

② 各步驟檢出方向的設定

各步驟搜尋訊號之搜尋方向為+方向指定位元設為0、-方向設為1。

各步驟搜尋方向之指定位元如下表。

表 2.6-4 各步驟檢出方向的指定位元

	步驟 1	步驟 2	步驟 3	步驟 4	
檢出方向	D1 位元	D5 位元	D11 位元	—	0: +方向
指定位元	S1DR	S2DR	S3DR		1: -方向

③ 各步驟檢出訊號的設定

步驟1可選擇 STOP0、STOP1 與極限訊號。步驟 2 可選擇 STOP1 與極限訊號。步驟 3 則固定為 STOP2 訊號。
步驟1及步驟2可設成相同訊號。

步驟1及步驟2之搜尋訊號指定如下表。

表 2.6-5 步驟 1 及步驟 2 檢出訊號的指定

步驟 1			步驟 2	
D3 位元 S1G1	D2 位元 S1G0	檢出訊號	D6 位元 S2SG	檢出訊號
0	0	STOP0	0	STOP1
0	1	STOP1	1	極限訊號
1	0	極限訊號		
1	1	(不可設定)		

若指定極限訊號作為搜尋訊號，步驟 1 選擇 D1 位元(S1DR)、步驟 2 選擇 D5 位元(S2DR)指定的搜尋方向面極限訊號。
搜尋方向為 + 方向的情況為 LMTP 訊號、- 方向時為 LMTM 訊號。

搜尋的輸入訊號動作是高位準或低位準之邏輯設定，必須在WR2 暫存器中設定。有關WR2 暫存器，請參4.5節。

④ 偏差計數器清除(DCC)輸出及實際位置/理論位置計數器清除的設定

在步驟2及3中，指定的搜尋訊號為無效，在往有效正緣觸發時，可指定是否輸出清除偏差計數器(DCC)訊號。輸出清除偏差計數器(DCC)訊號將指定位元設為 1、不輸出設為 0。

此外，步驟 2、3、4 結束時，可清除實際位置計數器/理論位置計數器。

清除實際位置計數器/理論位置計數器指定位元設為 1、不清除設為 0。

各步驟之清除偏差計數器(DCC)輸出、實際位置計數器/理論位置計數器清除之指定位元如下表。

表 2.6-6 各步驟的 DCC 輸出及實際位置·理論置計數器清除的指定位元

	步驟 1	步驟 2	步驟 3	步驟 4	
偏差計數器清除訊號 (DCC) 輸出	—	D7 位元 S2DC	D12 位元 S3DC	—	0: 不輸出 1: 輸出
實際位置計數器清除	—	D8 位元 S2RC	D13 位元 S3RC	(※1)	0: 不清除 1: 清除
理論位置計數器清除	—	D9 位元 S2LC	D14 位元 S2LC	(※1)	

(※1)步驟 4 結束時(設定執行步驟 4 時)之實際位置計數器/理論位置計數器清除，是在自動原點復歸模式設定 2 指令(24h)之自動原點復歸結束時，設定清除。請參下述「■ 自動原點復歸模式設定 2」。

■ 自動原點復歸模式設定2

自動原點復歸模式設定2，如下所示設定WR6暫存器的各位元，將自動原點復歸模式設定2指令(24h)，寫入WR0暫存器進行模式設定。清除偏差計數器(DCC)輸出之脈波邏輯與脈波寬、步驟間計時器有效/無效與計時器時間、自動原點復歸結束時之實際位置計數器/理論位置計數器清除、以編碼器Z相訊號(STOP2)與原點訊號(STOP1)之AND條件，進行停止之指定。

WR6	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
					HTM2	HTM1	HTM0	HTME	DCP2	DCP1	DCP0	DCPL	LCLR	RCLR	SAND	

① 偏差計數器清除(DCC)輸出脈波理論及脈波寬的指定

在各步驟輸出清除偏差計數器(DCC)訊號時，可設定脈波邏輯與脈波寬。

脈波邏輯以 D3 位元(DCPL)指定。如下圖，指定 0 為高脈波、指定 1 為低脈波。

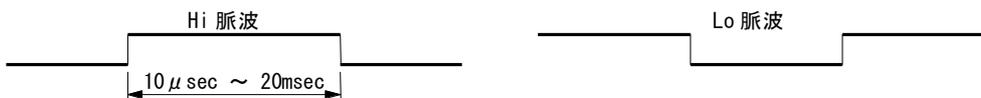


圖 2.6-12 偏差計數器清除輸出脈波理論位準

脈波寬是以 D6~4 位元(DCP2~DCP0)的 3 位元指定。

可指定脈波寬如下表。

表 2.6-7 偏差計數器清除輸出脈波寬

WR6/D6 DCP2	WR6/D5 DCP1	WR6/D4 DCP0	脈波寬 (CLK=16MHz 時)
0	0	0	10 μ sec
0	0	1	20 μ sec
0	1	0	100 μ sec
0	1	1	200 μ sec
1	0	0	1 msec
1	0	1	2 msec
1	1	0	10 msec
1	1	1	20 msec

② 步驟間計時器的指定

設定步驟間計時器之有效/無效及計時器時間。

有效/無效是以 D7 位元(HTME)指定。指定 1 有效、指定 0 無效。

計時器時間是以 D10~7 位元(HTM2~HTM0)指定。

可指定的計時器時間如下表。

表 2.6-8 步驟間計時器的時間寬指定

WR6/D10 HTM2	WR6/D9 HTM1	WR6/D8 HTM0	計時器時間 (CLK=16MHz 時)
0	0	0	1 msec
0	0	1	2 msec
0	1	0	10 msec
0	1	1	20 msec
1	0	0	100 msec
1	0	1	200 msec
1	1	0	500 msec
1	1	1	1000 msec

③ 自動原點復歸結束時實際位置計數器·理論位置計數器的清除

自動原點復歸結束時，可設定實際位置計數器/理論位置計數器之清除。

實際位置計數器之清除是以 D1 位元(RCLR)指定。清除設為 1、不清除設為 0。

理論位置計數器之清除是以 D2 位元(LCLR)指定。清除設為 1、不清除設為 0。

④ 編碼器 Z 相訊號 (STOP2) 及原點訊號 (STOP1) 的 AND 條件下停止

步驟 3 動作中，原點訊號 (STOP1) 有效，且編碼器 Z 相訊號 (STOP2) 變成有效時，停止驅動的機能。D0 位元(SAND) 若設為 1，在原點訊號 (STOP1) 有效下，編碼器 Z 相訊號 (STOP2) 同時變成有效時停止。

【注意】

- 本設定請在步驟 2 搜尋訊號設定為 STOP1 時使用。在步驟 2 的搜尋訊號上選擇極限訊號時，請務必指定 0。在步驟 2 之搜尋訊號選擇極限訊號，本設定若設為 1，則無法正確動作。

2.6.5 自動原點復歸的執行及狀態

■ 自動原點復歸的執行

自動原點復歸是根據自動原點復歸執行指令（5Ah）設定進行。正確設定自動原點復歸模式與速度參數後，將指令碼5Ah寫入WR0暫存器即開始。

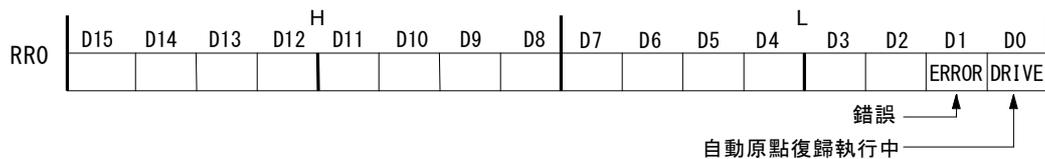
■ 自動原點復歸的中斷

欲中途中斷自動原點復歸時，寫入驅動減速停止指令（56h）、或驅動立即停止指令（57h）。當前執行中的步驟被中斷，未執行的步驟不再執行，自動原點復歸結束。

步驟間計時器進入有效狀態時，在步驟間計時器作動中，發出停止指令的情況，步驟間計時器動作亦會中斷，結束自動原點復歸。

■ 狀態暫存器

主狀態暫存器RR0的D0 位元，是顯示驅動執行中的位元，自動原點復歸執行時，也是以此位元顯示執行中狀態。自動原點復歸若開始進行，此位元變為1，在步驟1動作開始到步驟4動作結束之間，顯示1。步驟4結束即回歸0。

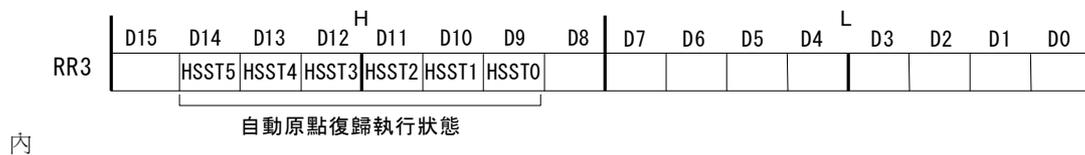


RR0暫存器的D1位元若在自動原點復歸中產生錯誤即變成1。產生錯誤原因由如下RR2 暫存器之D6~D0位元顯示。



各錯誤要素請參4.11節。

RR3暫存器的D14~D9 位元是將自動原點復歸之執行狀態以編號顯示。可得知自動原點復歸執行中，當前執行中的動作



內

表 2.6-9 自動原點復歸的執行狀態

執行狀態	執行步驟	動作內容
0		等待自動原點復歸執行指令
3	步驟 1	等待指定搜尋方向的搜尋訊號成為有效
6		步驟1與步驟2間之計時器動作中
1 1	步驟 2	等待指定搜尋方向反向的搜尋訊號為有效（不規則動作）
1 5		等待指定搜尋方向反向的搜尋訊號進入無效（不規則動作）
1 8		不規則動作後計時器動作中
2 0		等待指定搜尋方向的搜尋訊號有效
2 3		步驟2與步驟3間之計時器動作中、或清除偏差計數器輸出中
2 8	步驟 3	等待指定搜尋方向的STOP2訊號有效
3 2		步驟3與步驟4間之計時器動作中、或清除偏差計數器輸出中
3 6	步驟 4	於指定搜尋方向偏移移動中

2.6.6 自動原點復歸時的錯誤

自動原點復歸執行中，可能產生下表中的錯誤。

表 2.6-10 自動原點復歸時的錯誤

錯誤發生要因	錯誤發生後 I C 的動作	結束時的顯示
步驟 1 ~ 4 中 ALARM 訊號動作。	搜尋驅動立即停止、以下的步驟不執行即結束。	RR0/D1 : 1, RR2/D4 : 1
步驟 1 ~ 4 中 EMGN 訊號動作。	搜尋驅動立即停止、以下的步驟不執行即結束。	RR0/D1 : 1, RR2/D5 : 1
步驟 3 進行方向的極限訊號 (LMTP/M) 動作。 (注)	搜尋驅動立即停止/減速停止、以下的步驟不執行即結束。	RR0/D1 : 1, RR2/D3或D2 : 1
步驟 4 進行方向的極限訊號 (LMTP/M) 動作。 (注)	偏移移動即停/減速停止、結束。	RR0/D1 : 1, RR2/D3或D2 : 1
步驟 3 開始時 STOP2 訊號動作。	以下的步驟不執行即結束。	RR0/D1 : 1, RR2/D6 : 1

自動原點復歸結束後，請務必確認主狀態暫存器的錯誤位元 (RR0/D1)。錯誤位元上升變成1時，無法正確進行自動原點復歸。

(註) 於步驟1、2中，進行方向的極限若為有效，減速停止或立即停止，不會產生錯誤。

■ 感測器故障時的症狀

原點訊號和極限訊號等感測器迴路經常性故障之症狀描述。但配線周圍雜訊、配線鬆脫、元件不穩定動作等要素，導致間歇性故障，因分析困難，並不適合納入此處說明。此外，該些症狀在客戶開發系統時，訊號準位的邏輯設定錯誤、訊號配線錯誤時，也會發生。

表 2.6-11 感測器故障時的症狀

故障要因	症狀	
極限開關 及配線迴路的故障	常 ON	不會朝該方向驅動，結束時極限值錯誤 位元 (RR2/D3或D2) 變成1。
	常 OFF	會碰到該方向的機械終點，但原點復歸動作不會結束。
步驟 1 檢出訊號 (STOP0, 1) 感測器及配線迴路的故障	常 ON	步驟1設定為有效，不拘訊號是否由OFF位置開始進行自動原點復歸，都不會執行步驟1 (高速原點搜尋) 即進入步驟2。
	常 OFF	於步驟1 (高速原點搜尋) 碰到極限值停止後，進入步驟2的不規則動作。原點復歸結果正確但不是正常動作。
步驟 2 檢出訊號 (STOP1 指定 時) 感測器及配線迴路的故障	常 ON	於步驟2 (低速原點搜尋) 朝反方向驅動，碰到反方向的極限值立即停止。結束時反方向極限值的錯誤 位元 (RR2/D3或D2) 變成1。
	常 OFF	於步驟2 (低速原點搜尋) 碰到指定方向的極限值後，開始朝反方向移動，碰到反方向的極限值即結束。結束時反方向極限值的錯誤 位元 (RR2/D3或D2) 變成1。
Z 相 (STOP2) 感測器及配線 迴路的故障	常 ON	於步驟3 (低速Z相搜尋) 結束錯誤。RR2/D6變成1。
	常 OFF	於步驟3 (低速Z相搜尋)，碰到指定方向的極限值立即停止。結束時指定方向極限值的錯誤 位元 (RR2/D3或D2) 變成1。

2.6.7 自動原點復歸的注意點

■ 搜尋速度

為提升原點復歸位置精確度，搜尋原點速度(HV)必須設低。設定低於初速度的值，使輸入訊號為有效後立即停止。此外，執行步驟3的Z相編碼器搜尋時，Z相訊號延遲與搜尋原點速度(HV)的關係很重要。例如，Z相訊號路徑的光電耦合器延遲時間與IC內建積分濾波器延遲時間，合計長為500 μ sec時，為讓編碼器Z相輸出維持在1msec以上的ON狀態，須設定搜尋原點速度。

■ 步驟3(Z相搜尋)開始位置

步驟3的Z相搜尋，Z相(STOP2)訊號由無效狀態變成有效時，立即停止搜尋驅動。因而，步驟3的開始位置(亦即步驟2的停止位置)必須穩定，且並非該變化點。一般是進行機械式調整，以便讓步驟3的開始位置，是在編碼器Z相位置的反向180°處。

■ 軟體極限

自動原點復歸執行中，請將軟體極限設為無效。若將軟體極限設為有效，自動原點復歸無法正確進行。自動原點復歸正常結束後，請正確設定理論位置計數器、實際位置計數器後，再設定軟體極限。

■ 各輸入訊號的理論設定

自動原點復歸所使用的輸入訊號(STOP0,1,2)之動作有效邏輯設定，是在WR2暫存器的位元(WR2/D0,D2,D4)設定。自動原點復歸時，各訊號有效/無效之位元(WR2/D1,D3,D5)設定內容會被忽視。

2.6.8 自動原點復歸的實例

■ 例 1 使用原點訊號做原點復歸例

以1個原點訊號，執行高速原點復歸與低速貼近時，不進行搜尋編碼器Z相的實例。原點訊號務必在STOP1輸入。

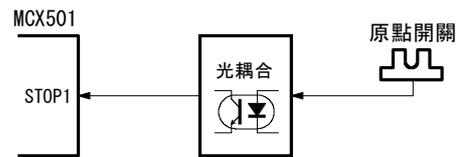


圖 2.6-13 自動原點復歸 例 1 的接線

自動原點復歸之動作順序如下表所示。

表 2.6-12 自動原點復歸 例 1 的動作

步驟	動作	執行/不執行	檢出訊號	訊號位準	檢出方向	檢出速度
1	高速搜尋	執行	STOP1	Low 動作	-方向	20,000pps
2	低速搜尋	執行			-方向	500pps
3	Z相搜尋	不執行	-	-	-	-
4	偏移移動	執行	-	-	+方向	20,000pps

步驟1是以20,000pps的高速速度，在一方向移動，直到STOP1訊號變成Low，當檢出Low位準後即減速停止。移至步驟2，STOP1訊號若Low位準(動作)時，以不規則動作□500pps的低速速度，向指定方向之反方向(此時為+方向)移動，直到STOP1訊號變成Hi位準，亦即STOP1離開有效區間後停止。然後再以500pps的低速度，依照步驟2的指定方向移動，STOP1訊號再度進入Low位準時停止。

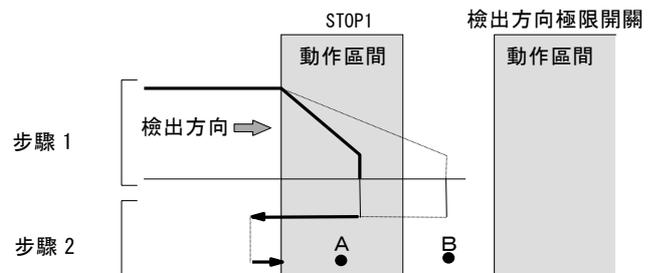


圖 2.6-14 自動原點復歸 例 1 的動作

作

步驟1中，即使STOP1訊號超過有效區間減速停止的情況，如上圖虛線所示，會再以反方向退回並離開STOP1之有效區間後，繼續執行步驟2的指定方向貼近原點。此動作僅限於步驟1、步驟2之搜尋訊號相同，且在指定相同搜尋方向之情況。

自動原點復歸開始位置若在上圖A點的情況，不執行步驟1，而是執行步驟2的不規則□。此外，當位於圖中B點時，則於步驟1碰到搜尋方向的極限後，再執行步驟2的不規則動作□。不規則動作□內容請參2.6.1項。

本例中假設步進馬達軸不使用編碼器做原點復歸，所以不執行步驟3的Z相搜尋。而步驟4在+方向偏移移動3500脈波設為工作原點。

【程式例】

```

// WR2暫存器設定
WR2 ← 0800h 寫 // 原點訊號理論設定：STOP1:Low動作
                // 硬體極限有效

// 輸入訊號濾波模式設定
WR6 ← 0A0Fh 寫 // D11~D8 1010 濾波延遲:512 μ sec
                // D2 1 STOP1訊號：濾波有效
WRO ← 0025h 寫 // 指令寫入

// 自動原點復歸模式設定 1
WR6 ← 8037h 寫 // D15 1 步驟 4 執行/不執行 執行
                // D14 0 步驟 3 LP清除 無效
                // D13 0 步驟 3 RP清除 無效
                // D12 0 步驟 3 DCC輸出 無效
                // D11 0 步驟 3 檢出方向 一
                // D10 0 步驟 3 執行/不執行 不執行
                // D9 0 步驟 2 LP清除 無效
                // D8 0 步驟 2 RP清除 無效
                // D7 0 步驟 2 DCC輸出 無效
                // D6 0 步驟 2 檢出訊號 STOP1
                // D5 1 步驟 2 檢出方向 一方向
                // D4 1 步驟 2 執行/不執行 執行
                // D3, 2 0, 1 步驟 1 檢出訊號 STOP1
                // D1 1 步驟 1 檢出方向 一方向
                // D0 1 步驟 1 執行/不執行 執行
WRO ← 0023h 寫 // 指令寫入

// 自動原點復歸模式設定 2
WR6 ← 0000h 寫 // D15 0
                // D14 0
                // D13 0
                // D12 0
                // D11 0
                // D10~8 0 計時器值
                // D7 0 步驟間計時器 無效
                // D6~4 0 DCC脈波寬
                // D3 0 DCC脈波理論
                // D2 0 原點復歸結束時LP清除 無效
                // D1 0 原點復歸結束時RP清除 無效
                // D0 0 步驟 2 & 3 無效
WRO ← 0024h 寫 // 指令寫入

// 高速原點搜尋及低速原點搜尋速度的設定
WR6 ← 7318h 寫 // 加減速度：95,000 PPS/SEC
WR7 ← 0001h 寫
WRO ← 0002h 寫

WR6 ← 03E8h 寫 // 初速度：1000 PPS
WR7 ← 0000h 寫
WRO ← 0004h 寫

WR6 ← 4E20h 寫 // 步驟 1, 4 的速度：20000 PPS
WR7 ← 0000h 寫
WRO ← 0005h 寫

WR6 ← 01F4h 寫 // 步驟 2 的速度：500 PPS
WR7 ← 0000h 寫
WRO ← 0014h 寫

// 偏移脈波的設定
WR6 ← 0DACH 寫 // 偏移移動脈波量：3500
WR7 ← 0000h 寫
WRO ← 0006h 寫

// 自動原點復歸執行開始
WRO ← 005Ah 寫

```

■ 例 2 使用極限訊號做原點復歸例

以某一方向的極限訊號替代原點訊號，做原點復歸的方法。在此實例為以一方向極限訊號，取代原點訊號。使用極限訊號做原點復歸時，有以下2項條件。

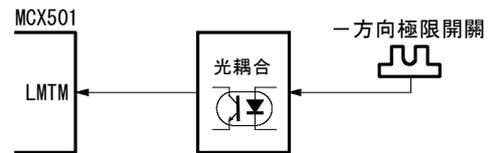


圖 2.6-15 自動原點復歸 例 2 的接統

- 執行步驟1高速搜尋動作時，極限訊號變成有效的位置開始，到機械硬體極限間的距離，需可以完全減速停止。
- 自動原點復歸開始的位置，往搜尋方向進行前，該位置並不在超過極限訊號有效區間處(圖2.5-16 B的位置)。

本例的自動原點復歸動作，依下表所示順序執行。於步驟1、2之模式設定中，若指定一方向為搜尋方向，指定極限訊號為搜尋訊號，即可決定一方向之極限訊號(LMTM)。

表 2.6-13 自動原點復歸 例 2 的動作

步驟	動作	執行/不執行	檢出訊號	訊號位準	檢出方向	檢出速度
1	高速搜尋	執行	LMTM	Low	-方向	20,000pps
2	低速搜尋	執行		動作	-方向	500pps
3	Z相搜尋	不執行	-	-	-	-
4	偏移移動	執行	-	-	+方向	20,000pps

步驟1到步驟4的動作，同前述原點訊號(STOP1)的情況。

自動原點復歸開始位置在右圖A點時，步驟1不執行，執行步驟2的不規則動作□，一旦極限訊號反向離開有效區間後，即進行指定方向之搜尋動作。

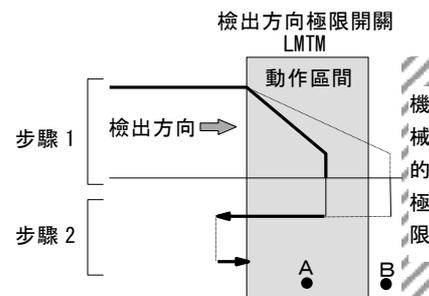


圖 2.6-16 自動原點復歸 例 2 的動作

【程式例】

```
// WR2暫存器設定
WR2 ← 1800h 寫 // 極限訊號理論設定：LMTM:Low動作
                // 硬體極限有效 減速停止 注1

// 輸入訊號濾波模式設定
WR6 ← 0A0Fh 寫 // D11~D8 1010 濾波延遲:512 μ sec
                // D1 1 LMTM訊號：濾波有效
WR0 ← 0025h 寫 // 指令寫入

// 自動原點復歸模式設定 1
WR6 ← 807Bh 寫 // D15 1 步驟 4 執行/不執行 執行
                // D14 0 步驟 3 LP清除 無效
                // D13 0 步驟 3 RP清除 無效
                // D12 0 步驟 3 DCC輸出 無效
                // D11 0 步驟 3 檢出方向 -
                // D10 0 步驟 3 執行/不執行 不執行
                // D9 0 步驟 2 LP清除 無效
                // D8 0 步驟 2 RP清除 無效
                // D7 0 步驟 2 DCC輸出 無效
                // D6 1 步驟 2 檢出訊號 LMTM
                // D5 1 步驟 2 檢出方向 一方向
                // D4 1 步驟 2 執行/不執行 執行
                // D3,2 1,0 步驟 1 檢出訊號 LMTM
                // D1 1 步驟 1 檢出方向 一方向
                // D0 1 步驟 1 執行/不執行 執行
WR0 ← 0023h 寫 // 指令寫入
```

```

// 自動原點復歸模式設定 2
WR6 ← 0000h 寫 // D15 0
// D14 0
// D13 0
// D12 0
// D11 0
// D10~8 0 計時器值
// D7 0 步驟間計時器 無効
// D6~4 0 DCC脈波寬
// D3 0 DCC脈波理論
// D2 0 原點復歸結束時LP清除 無効
// D1 0 原點復歸結束時RP清除 無効
// D0 0 步驟 2 & 3 無効
WR0 ← 0024h 寫 // 指令寫入

// 高速原點搜尋及低速原點搜尋速度的設定
WR6 ← 7318h 寫 // 加減速度：95,000 PPS/SEC
WR7 ← 0001h 寫
WR0 ← 0002h 寫

WR6 ← 03E8h 寫 // 初速度：1000 PPS
WR7 ← 0000h 寫
WR0 ← 0004h 寫

WR6 ← 4E20h 寫 // 步驟 1, 4 的速度：20000 PPS
WR7 ← 0000h 寫
WR0 ← 0005h 寫

WR6 ← 01F4h 寫 // 步驟 2 的速度：500 PPS
WR7 ← 0000h 寫
WR0 ← 0014h 寫

// 偏移脈波的設定
WR6 ← 0DACh 寫 // 偏移移動脈波量：3500
WR7 ← 0000h
WR0 ← 0006h 寫

// 自動原點復歸執行開始
WR0 ← 005Ah 寫

```

注1：WR2 暫存器的D10是極限訊號的邏輯、D11是極限有效、D12是設定極限動作的位元，但如本例所示，使用極限訊號作為搜尋訊號時，該步驟的動作，與D11的設定無關，極限訊號自動有效(D11之設定資料，在將極限訊號使用於搜尋訊號的步驟下，不會對動作有所影響)。D12請務必設為減速停止有效。D10則請配合使用狀況設定。

【極限訊號使用時的注意】

步驟1,2之搜尋方向務必為同向。此外，若要執行步驟3（Z相搜尋）動作時，其搜尋方向與步驟1、2方向相反。步驟4（偏移移動）動作方向也和步驟1,2相反，請務必能在離開極限有效區間的位置，結束自動原點復歸。

■ 例 3 伺服馬達軸的原點復歸例

接受脈波輸入之伺服馬達，一般由驅動器（伺服放大器）輸出編碼器Z相訊號。執行位置高精度原點復歸的時候，配合該Z相編碼器的輸出時序，必須清除驅動器內部的偏差計數器，及輸入清除偏差計數器的訊號。這些訊號的原點復歸接線方式如下所示。

如下圖所示，由設置在軸上的原點感測器，透過介面迴路輸入原點訊號(STOP1)。伺服驅動器方面，透過介面迴路連接Z相編碼器輸入(STOP2)、清除偏差計數器輸出(DCC)。

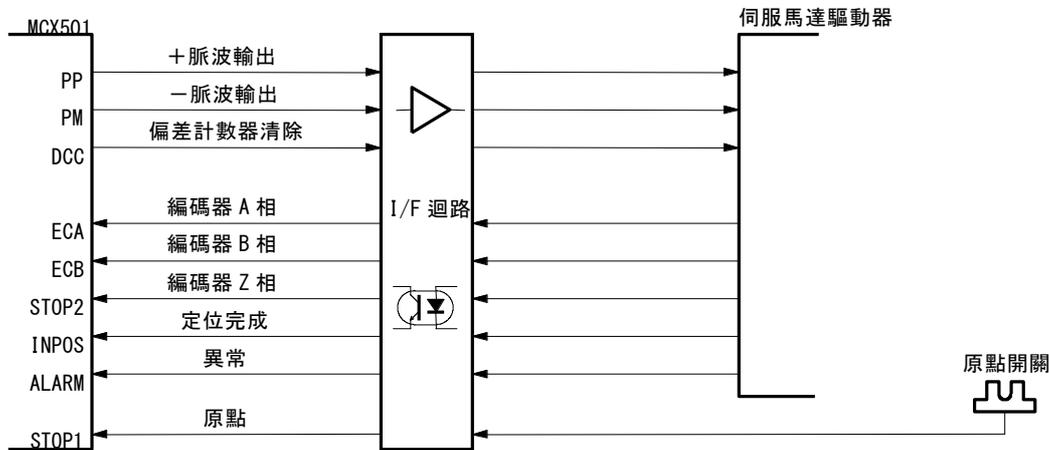


圖 2.6-17 自動原點復歸 例 3 的接線

注意：Z相編碼器輸入請務必接在本IC的STOP2上。因響應性必須要快，介面迴路使用線性迴授或高速光耦合器較合適。

表 2.6-14 自動原點復歸 例 3 的動作

步驟	動作	執行/不執行	檢出訊號	訊號位準	檢出方向	檢出速度
1	高速搜尋	執行	STOP1	Low 動作	-方向	20,000pps
2	低速搜尋	執行			-方向	500pps
3	Z相搜尋	執行	STOP2	Low	-方向	500pps
4	偏移移動	執行	-	-	+方向	20,000pps

由步驟1到步驟2的動作，與前述原點訊號(STOP1)例的情況一樣。在步驟2的STOP1輸入變成Low，步驟2即告結束，進入步驟3。在步驟3以500pps的速度，於一方向搜尋到STOP2(Z相)訊號為Low前移動，搜尋到Low後即停。以STOP2輸入訊號的↓，輸出DCC(清除偏差計數器)。在此例中，DCC訊號是設定成能以100μsec的寬，輸出Hi脈波。

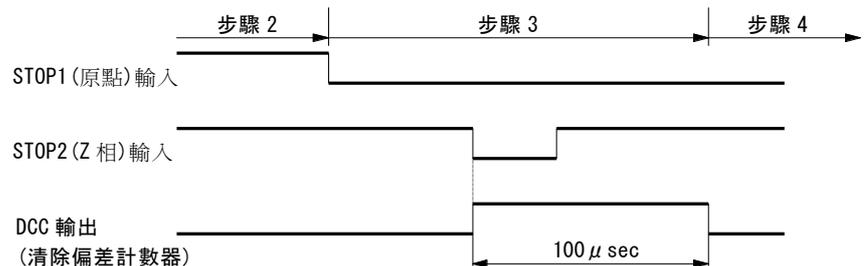


圖 2.6-18 自動原點復歸 例 3 的動作

此外，此處步驟3的STOP2(Z相)訊號變成低位準時，設定成可清除實際位置計數器與理論位置計數器。

【程式例】

```

// WR2暫存器設定
WR2 ← 0800h 寫 // 原點訊號理論設定：STOP1,2:Low動作
                // 硬體極限有效

// 輸入訊號濾波模式設定
WR6 ← 0ACFh 寫 // D15~D12 0000 濾波FE6,7延遲:500nsec
                // D11~D8 1010 濾波FE0-5延遲:512μsec
                // D6 1 STOP2訊號：濾波有效
                // D2 1 STOP1訊號：濾波有效
WR0 ← 0025h 寫 // 指令寫入

// 自動原點復歸模式設定 1
WR6 ← FC37h 寫 // D15 1 步驟 4 執行/不執行 執行
                // D14 1 步驟 3 LP清除 有效
                // D13 1 步驟 3 RP清除 有效
                // D12 1 步驟 3 DCC輸出 有效
                // D11 1 步驟 3 檢出方向 一方向
                // D10 1 步驟 3 執行/不執行 執行
                // D9 0 步驟 2 LP清除 無效
                // D8 0 步驟 2 RP清除 無效
                // D7 0 步驟 2 DCC輸出 無效
                // D6 0 步驟 2 檢出訊號 STOP1
                // D5 1 步驟 2 檢出方向 一方向
                // D4 1 步驟 2 執行/不執行 執行
                // D3,2 0,1 步驟 1 檢出訊號 STOP1
                // D1 1 步驟 1 檢出方向 一方向
                // D0 1 步驟 1 執行/不執行 執行
WR0 ← 0023h 寫 // 指令寫入

// 自動原點復歸模式設定 2
WR6 ← 0020h 寫 // D15 0
                // D14 0
                // D13 0
                // D12 0
                // D11 0
                // D10~8 0 計時器值
                // D7 0 步驟間計時器 無效
                // D6~4 010 DCC脈波寬 100μsec
                // D3 0 DCC脈波理論 Hi脈波
                // D2 0 原點復歸結束時LP清除 無效
                // D1 0 原點復歸結束時RP清除 無效
                // D0 0 步驟 2 & 3 無效
WR0 ← 0024h 寫 // 指令寫入

// 高速原點搜尋及低速原點搜尋速度的設定
WR6 ← 7318h 寫 // 加減速度：95,000 PPS/SEC
WR7 ← 0001h 寫
WR0 ← 0002h 寫

WR6 ← 03E8h 寫 // 初速度：1000 PPS
WR7 ← 0000h 寫
WR0 ← 0004h 寫

WR6 ← 4E20h 寫 // 步驟 1, 4 的速度：20000 PPS
WR7 ← 0000h 寫
WR0 ← 0005h 寫

WR6 ← 01F4h 寫 // 步驟 2, 3 的速度：500 PPS
WR7 ← 0000h 寫
WR0 ← 0014h 寫

// 偏移脈波的設定
WR6 ← 0DACH 寫 // 偏移移動脈波量：3500
WR7 ← 0000h 寫
WR0 ← 0006h 寫

// 自動原點復歸執行開始
WR0 ← 005Ah 寫

```

2.7 同步動作

本 IC 之同步動作是指於軸驅動途中，在指定位置輸出外部訊號，或相反地利用外部訊號，將現在通過位置儲存在指定暫存器等，在 IC 以外的裝置之間，執行各種連動動作的機能。例如可執行以下動作。

例1 驅動中通過指定位置即向外部輸出訊號。

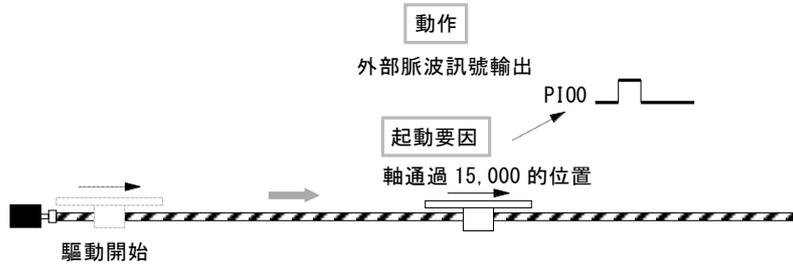


圖 2.7-1 同步動作 例 1

例2 驅動中外部有訊號輸入時，將現在位置儲存在規定的暫存器。

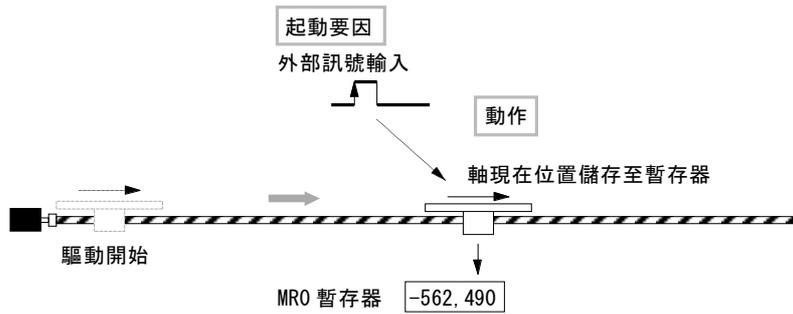


圖 2.7-2 同步動作 例 2

例3 驅動中由指定位置向外部輸出N個分割脈波。

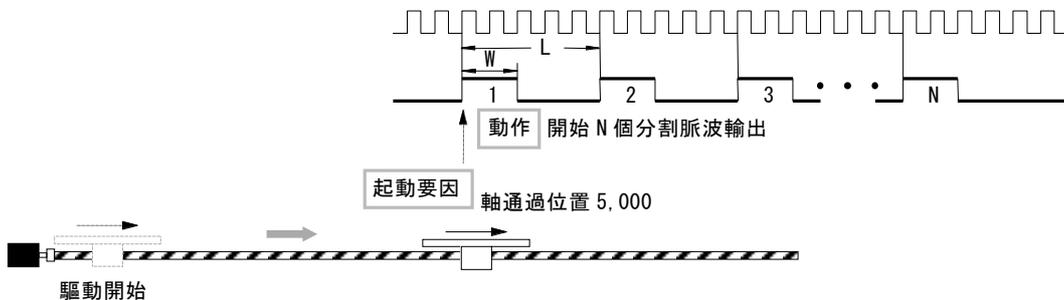


圖 2.7-3 同步動作 例 3

例4 驅動中量測由指定位置A到指定位置B的經過時間。

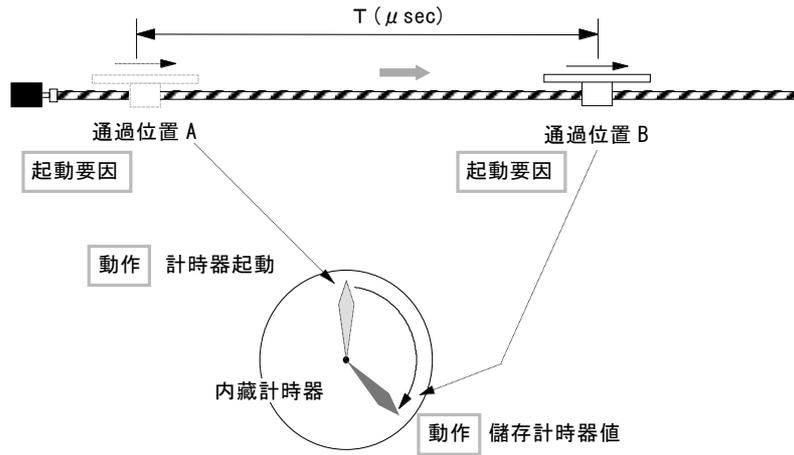


圖 2.7-4 同步動作 例 4

通常這種同步動作，藉 CPU 端的程式亦可執行。但採用 CPU 中斷處理時間和程式執行時間，在不容許時間延遲的情況下，使用本機能就很方便。本 IC 的同步動作是在指定的起動要素發生，會立即執行指定動作的機能。該連鎖動作並不需透過 CPU 執行，因此可做到高精確度的同步控制。

將執行指定起動要素有效的指定動作，設為1個同步動作組，MCX501具有4個獨立的同步動作組。

並且，4組同步動作組除可分別獨立動作之外，亦可4組連鎖動作。

各同步動作組SYNC0~3中，備有15種起動要素。由其中選取1個來設定編碼。此外，起動的動作有24種。

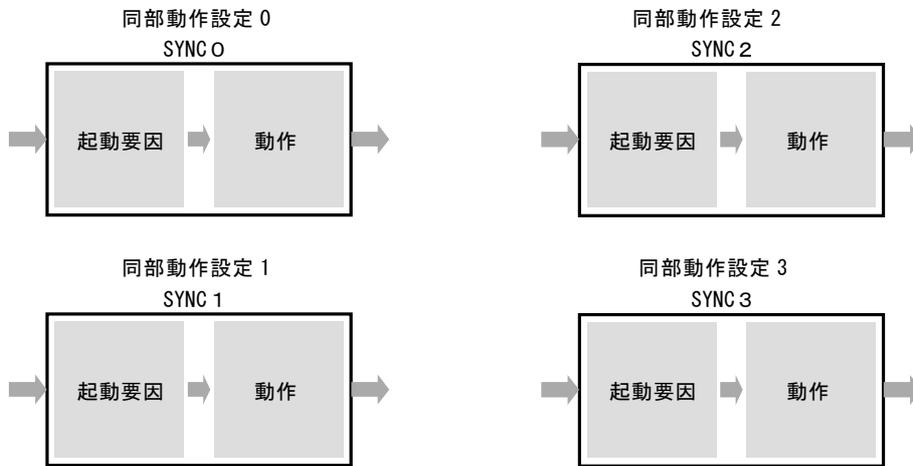


圖 2.7-5 同步動作組

2.7.1 起動要因

起動同步動作的起動要素，如下表所示從0h到Fh共16個指定碼指定。

表 2.7-1 起動要因一覽

設定編碼 (Hex)	同步動作組 0 SYNC0	同步動作組 1 SYNC1	同步動作組 2 SYNC2	同步動作組 3 SYNC3	詳細
1	MR0對象變為真	MR1對象變為真	MR2對象變為真	MR3對象變為真	說明 1
2	內藏計時器計時器到時				說明 2
3	驅動開始				說明 3
4	加減速驅動的等速域開始				說明 3
5	加減速驅動的等速域結束				說明 3
6	驅動結束				說明 3
7	分割脈波開始				說明 4
8	分割脈波結束				說明 4
9	分割脈波輸出				說明 4
A	PI00輸入訊號 ↑	PI01輸入訊號 ↑	PI02輸入訊號 ↑	PI03輸入訊號 ↑	說明 5
B	PI00輸入訊號 ↓	PI01輸入訊號 ↓	PI02輸入訊號 ↓	PI03輸入訊號 ↓	說明 6
C	PI04輸入Low 且PI00輸入 ↑	PI05輸入Low 且PI01輸入 ↑	PI06輸入Low 且PI02輸入 ↑	PI07輸入Low 且PI03輸入 ↑	說明 7
D	PI04輸入Hi 且PI00輸入 ↑	PI05輸入Hi 且PI01輸入 ↑	PI06輸入Hi 且PI02輸入 ↑	PI07輸入Hi 且PI03輸入 ↑	說明 8
E	PI04輸入Low 且PI00輸入 ↓	PI05輸入Low 且PI01輸入 ↓	PI06輸入Low 且PI02輸入 ↓	PI07輸入Low 且PI03輸入 ↓	說明 9
F	PI04輸入Hi 且PI00輸入 ↓	PI05輸入Hi 且PI01輸入 ↓	PI06輸入Hi 且PI02輸入 ↓	PI07輸入Hi 且PI03輸入 ↓	說明 10
0	NOP				說明 11

說明 1： MRn對象變真

多目的暫存器(以下稱MRn暫存器)比較標的在符合比較條件時起動。如表中所示，4個同步動作組所對應的MRn暫存器已固定。比較標的與比較條件是以多目的暫存器模式設定指令(20h)設定。例如，MR0暫存器之比較標的設定為理論位置計數器(LP)，比較條件設定為“比較標的□MRn”的情況，在理論位置計數器的值在等於或大於MR0的值時起動。

同步動作進入有效狀態時，比較條件業已為真的情況，一旦進入非真狀態後，再度進入真的狀態時，同步動作即起動。

說明 2： 內建計時器計時到達

內建計時器計時到時起動。計時器的值以計時器值設定指令(16h)設定。計時器可藉由寫入計時器起動指令(73h)、或以其他同步動作組開始。

說明 3： 驅動狀態變化

如下圖所示，於驅動中之速度狀態變化時起動。

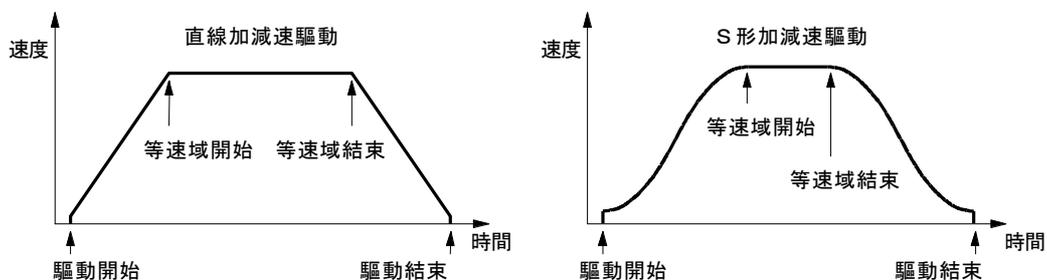


圖 2.7-6 驅動狀態之相關起動要因

【注意】

- 加減速驅動中之等速域(以固等速度執行驅動的區間)，在驅動結束時亦會有微小產生的情況。

說明 4： 分割脈波

“分割脈波開始”是以分割脈波開始指令(75h)，或其他的同步動作組，於分割脈波開始時起動同步動作。

“分割脈波結束”是在最終分割脈波輸出結束時起動同步動作。

“分割脈波輸出”是輸出分割脈波（上升或下降到有效準位）時起動同步動作。同步動作若設定為重覆時，則各分割脈波均會起動同步動作。

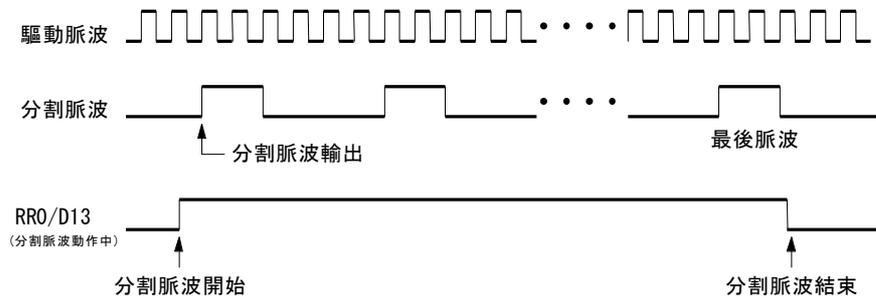


圖 2.7-7 分割脈波的起動要因

說明 5： 汎用輸入訊號之正緣觸發變化

“PIOn輸入訊號↑”是PIOn(n=0~3)輸入訊號由低位準上升到高位準時起動。

如表中所示，4個同步動作集所對應的PIOn訊號為固定。

同步動作設為有效時，輸入訊號已經在高位準的時候，會等到訊號一旦降到低位準，再度上升到高位準時起動同步動作。

說明 6： 汎用輸入訊號之負緣觸發變化

“PIOn輸入訊號↓”是在PIOn(n=0~3)輸入訊號由高位準下降到低位準時起動。

如表中所示，4個同步動作集所對應的PIOn訊號為固定。

同步動作設為有效時，輸入訊號已經在低位準的時候，會等到訊號一旦上到高位準，再度降到低位準時起動同步動作。

說明 7： 汎用輸入訊號的Low位準及正緣觸發變化

“PIOm輸入Low且PIOn輸入↑”是PIOm(m=4~7)輸入訊號為低位準，且PIOn(n=0~3)輸入訊號由低位準上升到高位準時起動。

如表中所示，4個同步動作集所對應的PIOn、PIOm訊號為固定。

同步動作設為有效時，PIOm輸入訊號已為Low、且PIOn輸入訊號進入高位準時其動作如同說明5。

說明 8： 汎用輸入訊號之Hi位準及正緣觸發變化

“PIOm輸入Hi且PIOn輸入↑”是PIOm(m=4~7)輸入訊號為高位準，且PIOn(n=0~3)輸入訊號由低位準上升到高位準時起動。

如表中所示，4個同步動作集所對應的PIOn、PIOm訊號為固定。

同步動作設為有效時，PIOm輸入訊號已為Hi，且PIOn輸入訊號進入高位準時其動作如同說明5。

說明 9： 汎用輸入訊號之Low位準及負緣觸發變化

“PIOm輸入Low且PIOn輸入↓”是PIOm(m=4~7)輸入訊號為低位準，且PIOn(n=0~3)輸入訊號由高位準下降到低位準時起動。

如表中所示，4個同步動作集所對應的PIOn、PIOm訊號為固定。

同步動作設為有效時，PIOm輸入訊號已為Low、且PIOn輸入訊號進入低位準時其動作如同說明6。

說明 10： 汎用輸入訊號之Hi位準及負緣觸發變化

“PIOm輸入Hi且PIOn輸入↓”是PIOm(m=4~7)輸入訊號為高位準，且PIOn(n=0~3)輸入訊號由高位準下降到低位準時起動。

如表中所示，4個同步動作集所對應的PIOn、PIOm訊號為固定。

同步動作設為有效時，PIOm輸入訊號已為Hi、且PIOn輸入訊號進入低位準時其動作如同說明6。

說明11: NOP

未設定起動要素的條件時使用。

例如在模式設定中，使用其他SYNC起動時，起動的同步動作組之起動要素設為NOP。

2.7.2 動作(Action)

下表所示為起動的動作 (Action)。指令碼01~09h,0Fh,10h依同步動作組0到同步動作組4動作有所不同。

表 2.7-2 動作 (Action) 一覽

設定碼 (Hex)	同步動作組 0 SYNC0	同步動作組 1 SYNC1	同步動作組 2 SYNC2	同步動作組 3 SYNC3	詳細
01	MR0 → DV	MR1 → DV	MR2 → DV	MR3 → DV	說明 1
02	MR0 → TP	MR1 → TP	MR2 → TP	MR3 → TP	說明 1
03	MR0 → SP1	MR1 → SP1	MR2 → SP1	MR3 → SP1	說明 1
04	MR0 → LP	MR1 → RP	MR2 → SV	MR3 → AC	說明 1
05	LP → MR0	LP → MR1	LP → MR2	LP → MR3	說明 2
06	RP → MR0	RP → MR1	RP → MR2	RP → MR3	說明 2
07	CT → MR0	CT → MR1	CT → MR2	CT → MR3	說明 2
08	CV → MR0	CA → MR1	—	—	說明 2
09	PI00訊號脈波輸出	PI01訊號脈波輸出	PI02訊號脈波輸出	PI03訊號脈波輸出	說明 3
0A	相對位置驅動開始				
0B	反相對位置驅動開始				
0C	絕對位置驅動開始				
0D	+方向連續脈波驅動開始				
0E	-方向連續脈波驅動開始				
0F	MR0值相對位置 驅動開始	MR1值相對位置 驅動開始	MR2值相對位置 驅動開始	MR3值相對位置 驅動開始	說明 4
10	MR0值絕對位置 驅動開始	MR1值絕對位置 驅動開始	MR2值絕對位置 驅動開始	MR3值絕對位置 驅動開始	說明 4
11	驅動減速停止				
12	驅動立即停止				
13	驅動速度增加				說明 5
14	驅動速度減少				說明 5
15	計時器始動				
16	計時器停止				
17	分割脈波開始				說明 6
18	分割脈波停止				說明 6
00	NOP				說明 7

說明 1： 參數值的載入

將多目的暫存器MRn(n=0~3)的值，載入各參數中。

表 2.7-3 參數值的載入

(n = 0~3)

表記	說明
MRn → DV	MRn暫存器的值載入驅動速度 (DV)。
MRn → TP	MRn暫存器的值載入輸出脈波數 (TP)。
MRn → SP1	MRn暫存器的值載入分割脈波資料 1 (分割長及脈波寬)。
MRO → LP	MRO暫存器的值載入理論位置計數器 (LP)。
MR1 → RP	MR1暫存器的值載入實際位置計數器 (RP)。
MR2 → SV	MR2暫存器的值載入初速度 (SV)。
MR3 → AC	MR3暫存器的值載入加速度 (AC)。

按照同步動作組的編號，所使用的MRn暫存器為固定。

動作碼04h按照同步動作組編號，MRn暫存器的值載入，參數即改變。

說明 2： 參數值的儲存

將各參數的值儲存在多目的暫存器MRn(n=0~3)。

表 2.7-4 參數值的儲存

(n = 0~3)

表記	說明
LP → MRn	理論位置計數器 (LP) 的值儲存到MRn暫存器。
RP → MRn	實際位置計數器 (RP) 的值儲存到MRn暫存器。
CT → MRn	現在計時器值的值儲存到MRn暫存器。
CV → MRO	現在驅動速度的值儲存到MRO暫存器。
CA → MR1	現在加減速度的值儲存到MR1暫存器。

按照同步動作組的編號，使用固定的MRn暫存器。

動作碼08h僅在同步動作組1與2有效，MRn暫存器的值儲存，參數即改變。

說明 3： 同步脈波訊號輸出

由PIO_n(n=0~3)訊號輸出脈波訊號。

4個同步動作組編號使用固定的PIO_n訊號。

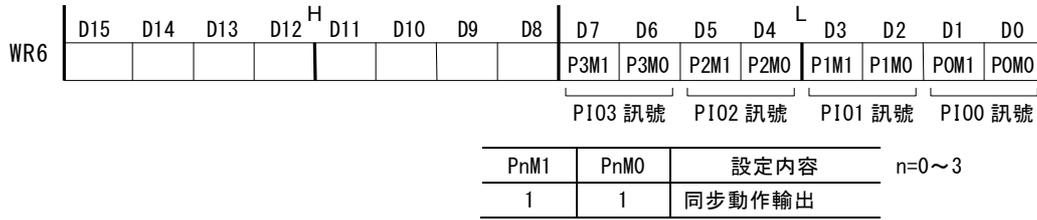
為執行本動作，必須設定以下項目。

- ① PIO_n訊號的同步脈波輸出設定
- ② 輸出脈波訊號的理論及脈波寬設定

同步動作的脈波訊號欲對外部輸出，必須用同步脈波輸出模式設定泛用輸出入訊號。其次，必須設定將此訊號以Hi脈波輸出或以Low脈波輸出的邏輯設定與脈波寬。這些是以PIO訊號設定1指令(21h)與PIO訊號設定2暨其他設定指令(22h) 設定。

① PIO_n(n=0~3)訊號的同步脈波輸出設定

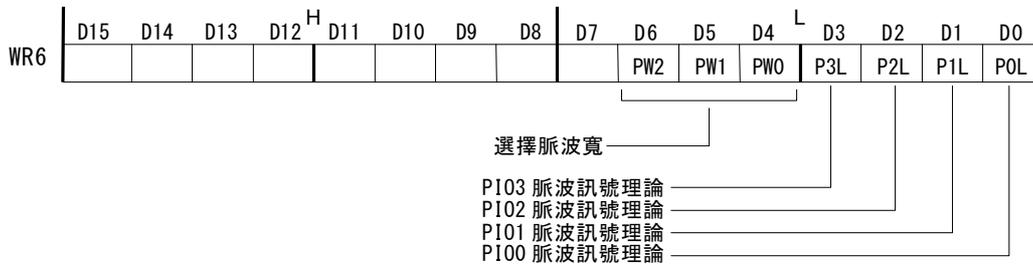
同步脈波輸出用模式設定PIO_n訊號，以PIO訊號設定1指令(21h)設定。
必須做以下設定。



使用的PIO_n訊號所對應的WR6暫存器，以2位元供同步脈波訊號輸出用設為1,1。例如使用PIO2訊號時，將WR6暫存器的D5,D4 位元(P2M1,P2M0)設為1,1之後，將PIO訊號設定1指令(21h) 寫入WR0暫存器。

② 輸出脈波訊號的理論及脈波寬的設定

設定輸出脈波訊號之邏輯與脈波寬，以PIO訊號設定2暨其他設定(22h)進行。
必須做以下設定。



PnL (n=0~3)	脈波訊號的理論		
0	輸出正理論脈波。		
1	輸出負理論脈波。		

PW2	PW1	PW0	脈波寬 (CLK=16MHz時)
0	0	0	125nsec
0	0	1	312nsec
0	1	0	1 μ sec
0	1	1	4 μ sec
1	0	0	16 μ sec
1	0	1	64 μ sec
1	1	0	256 μ sec
1	1	1	1msec

WR6暫存器的D0到D3位元(P0L~P3L)指定所使用的PIO訊號之脈波邏輯。0為輸出正邏輯脈波、1為輸出負邏輯脈波。未使用訊號所對應的位元為0或1均可。此外，WR6暫存器的D4到D6位元(PW0~PW3)設定上表所示的脈波寬。若在WR0暫存器寫入PIO訊號設定2·其他指令(22h)，則WR6暫存器指定內容即被設定。

【注意】

- 脈波寬之設定和 PIO0~PIO3 訊號全部共通。各訊號每個脈波寬不能個別設定。
- 同步脈波輸出之動作連續起動的情況，若下個起動發生在同步脈波輸出途中，同步脈波不會進入無效狀態，由起動時點開始，再度輸出指定脈波寬。

說明 4： MRn值相對／絕對位置驅動起動

驅動開始時，將MRn暫存器的值設定在移動脈波數(TP)值中，據此起動相對／絕對位置驅動。

因MRn暫存器的值寫入移動脈波數(TP)中，若執行該動作(Action)，移動脈波數(TP)的設定內容即改變。可確認以移動脈波數／終點設定值讀取指令(46h)所變更的移動脈波數(TP)值。

說明 5： 驅動速度增加／減少

使驅動中的現在驅動速度增加／減少。增減值必須以速度增減值設定指令(15h)預設。

S形驅動之加減速中，本動作(Action)進入無效狀態。

說明 6： 分割脈波開始／停止

分割脈波開始會以預設內容開始分割脈波。根據起動要素發生時序，決定分割脈波之開始驅動脈波。分割脈波停止會使現在動作中的分割脈波停止。根據起動要素發生時序，決定分割脈波的停止時序。詳細內容請參2.7節。

說明 7： NOP

即便起動要素有效，就動作(Action)而言，在不做任何動作時指定。

例如可在只因某起動要素而發生中斷時使用。

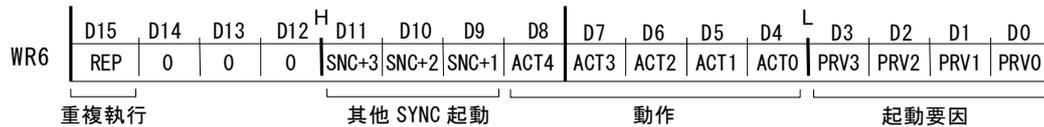
2.7.3 同步動作組

同步動作的設定分為SYNCn設定、有效設定及無效設定、依此設定進行同步動作。

■ SYNCn 設定

依據同步動作SYNCn設定指令(26h, 27h, 28h, 29h)可設定、起動要因、動作(Action)、其他同步動作同時起動及同步動作的單一／重複...等四種同步動作。

WR6 暫存器中寫入設定內容後、再寫入同步動作組指令。



⑤ 起動要因的設定

起動要因由 D3~0 位元(PRV3~PRV0)的 4 位元指定。

例如、要設定起動要因之「驅動開始」時、需指定編碼為 3h、所以 D3~0 設為 0011。

有關詳細起動要因之設定請參考 2.6.1 項之說明。

⑥ 動作(Action)的設定

動作(Action)由 D8~4 位元(ACT4~ACT0)的 5 位元指定。

例如、要設定動作(Action)之「分割脈波開始」時、需指定編碼為 17h、所以 D8~4 設為 10111。

有關詳細動作(Action)之設定請參考 2.6.2 項之說明。

⑦ 其他同步動作同時起動的設定

同步動作組在起動要因動作時、想連同其他同步動作組的動作(Action)同時起動時就需設定此位元。

指定 D11~9 位元(SNC+3~SNC+1)。

起動其他同步動作組的動作(Action)時設定為 1、不起動時設定為 0。

位元的指定及起動其他的同步動作組的對應如下表所示。

表 2.7-5 其他同步動作同時起動

自同步動作組	D11 (SNC+3)	D10 (SNC+2)	D9 (SNC+1)
SYNC0	SYNC3 起動	SYNC2 起動	SYNC1 起動
SYNC1	SYNC0 起動	SYNC3 起動	SYNC2 起動
SYNC2	SYNC1 起動	SYNC0 起動	SYNC3 起動
SYNC3	SYNC2 起動	SYNC1 起動	SYNC0 起動

善用此機能時、可在單一起動要因下、同時起動複數動作(Action)、因此可以進行更複雜的同步動作。

設定為、例如本身的同步動作組為 SYNC0。此時、SYNC0 的起動要因若動作時、也像要讓 SYNC1、2 的動作(Action)同時起動時、如上表所示、D9 及 D10 位元設為 1。依此設定、SYNC0 的起動要因若動作時、除 SYNC0 的動作(Action)外、SYNC1,2 的動作(Action)亦同時起動。此時、SYNC1,2 的起動要因設為 NOP，僅將設定動作(Action)即可。另需以同步動作有效設定指令將其狀態設為有效。

⑧ 同步動作組的重複設定

可設定同步動作組的有效狀態、在同步動作一經起動後要不要變為無效狀態。

重複有效時 D15 位元(REP)設為 1、僅 1 次有效時設為 0。

重複有效時、起動要因在動作的時候會重複起動同步動作。僅 1 次有效時、起動要因在初次動作時僅執行 1 次同步動作。

【注意】

- 重複設定有效時、起動要因做驅動停止、動作(Action)做相對位置驅動起動時、驅動結束→驅動開始的動作將不結束的情況下繼續執行。以停止指令無法停止。須以同步動作無效設定指令停止。

■ 有效設定

以同步動作有效設定指令(81h~8Fh)設定各同步動作組之有效狀態。

同步動作組有效狀態的時候、起動要因若動作時動作(Acition)即起動。

4個同步動作組各有其對應之指令編碼、同步動作組 SYNC0 為 81h、同步動作組 SYNC1 為 82h、同步動作組 SYNC2 為 84h、同步動作組 SYNC3 為 88h。

這些指令的組合下可以讓複數同時有效。例如、83h 指令執行時 SYNC0,1 成為有效狀態。指令編碼的組合方式請參考表 2.6-6。

SYNCn 設定中 REP=0 的時候、一經同步動作執行後、其同步動作組會變為無效狀態、即使起動要因再次動作同步動作也不再起動。REP=1 的時候、即使同步動作執行過、其同步動作組的有效狀態不會改變。

依據同步動作執行、一旦變為無效狀態的同步動作組要再讓狀態有效時、需再一次用同步動作有效設定指令設定。

PIO 訊號設定 2・其他指令(22h)之、ERRDE=1、當錯誤發生時(主狀態暫存器 RR0 的 D1 變為 1 的狀態)、全部同步動作組均變為無效狀態。此狀態下、若不清除錯誤狀態、即使發出同步動作有效設定指令、狀態也不會有效。要清除錯誤狀態時、請使用錯誤・結束狀態清除指令(79h)。

4個同步動作組的有效/無效狀態、可由主狀態暫存器 RR0 的 D11~D8 位元(SYNC3~SYNC0)確認。

■ 無效設定

以同步動作無效設定指令(91h~9Fh)設定各同步動作組為無效狀態。

同步動作組若處於無效狀態時、起動要因即使發生動作(Acition)也不會起動。

重置時、4個同步動作組全部變為無效狀態。

4個同步動作組各有相對應的指令編碼、同步動作組 SYNC0 為 91h、同步動作組 SYNC1 為 92h、同步動作組 SYNC2 為 94h、同步動作組 SYNC3 為 98h。

與同步動作有效設定指令一樣、這些指令的組合下可以讓複數同時無效。指令編碼的組合方式請參考表 2.6-6。

改變同步動作的無效狀態有、「發出同步動作無效指令時」、「PIO 訊號設定 2・其他設定指令(22h)下錯誤發生時同步動作無效設定(D7:ERRDE)設為有效、而錯誤發生時」、「同步動作 1 次(重複無效)設定下、同步動作起動後」的 3 個。

4個同步動作組的有效/無效狀態可經由主狀態暫存器 RR0 的 D11~D8 位元(SYNC3~SYNC0)確認。

表 2.7-6 同步動作有效，無效，起動指令碼及所對應的同步動作組

指令碼 (Hex)			同步動作組			
有效設定	無效設定	起動	同步動作組 3 SYNC3	同步動作組 2 SYNC2	同步動作組 1 SYNC1	同步動作組 0 SYNC0
81	91	A1	—	—	—	○
82	92	A2	—	—	○	—
83	93	A3	—	—	○	○
84	94	A4	—	○	—	—
85	95	A5	—	○	—	○
86	96	A6	—	○	○	—
87	97	A7	—	○	○	○
88	98	A8	○	—	—	—
89	99	A9	○	—	—	○
8A	9A	AA	○	—	○	—
8B	9B	AB	○	—	○	○
8C	9C	AC	○	○	—	—
8D	9D	AD	○	○	—	○
8E	9E	AE	○	○	○	—
8F	9F	AF	○	○	○	○

○：有效設定指令執行時為有效狀態、無效設定指令執行時為無效狀態、起動指令執行時為起動。

—：有效、無效設定指令執行時也不會改變狀態。起動指令執行時亦不起動。

2.7.4 同步動作執行

■ 同步動作的執行步驟

同步動作如以下步驟進行。

- ① 依同步動作 SYNCn 設定指令(26h~29h)、設定起動要因及動作(Action)。
- ② 依同步動作有效設定指令(81h~8Fh)、設定同步動作有效。
- ③ 設定起動要因發生時、同步動作作動。

■ 同步動作依起動指令而起動

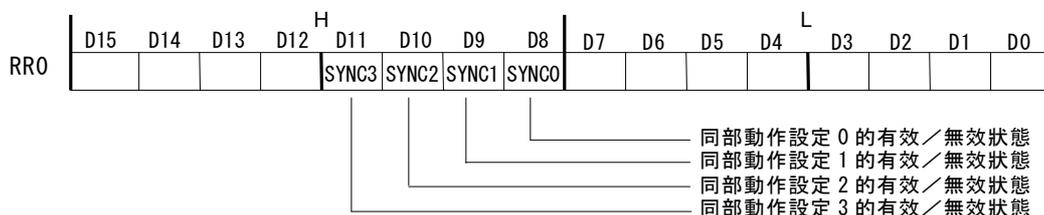
同步動作可依指令起動。同步動作依據起動指令(A1h~Ah)起動。依據指令編碼複數的同步動作組可同時起動。指令編碼及起動之同步動作 SYNC3~0 的對應請參考如表 2.7-6。

依據同步動作起動指令來起動同步動作時、需以同步動作有效設定指令指定同步動作組有效。

■ 主狀態暫存器

主狀態暫存器 RR0 的 D11~D8 位元(SYNC3~SYNC0)可用來確認同步動作組的狀態。

位元1的時候同步動作組為有效狀態、0的時候為無效狀態。



2.7.5 同步動作發出中斷

同步動作起動時可發出中斷。

設定 WR1 暫存器的 D15~D12 位元(SYNC3~SYNC0)。

這些位元設為1的時候、位元對應之同步動作組的起動要因動作時會發出中斷。

有關中斷機能請參考 2.10 節。

2.7.6 同步動作實例

■ 例 1 驅動中通過指定位置 15000 時同步脈波透過 PIO0 輸出

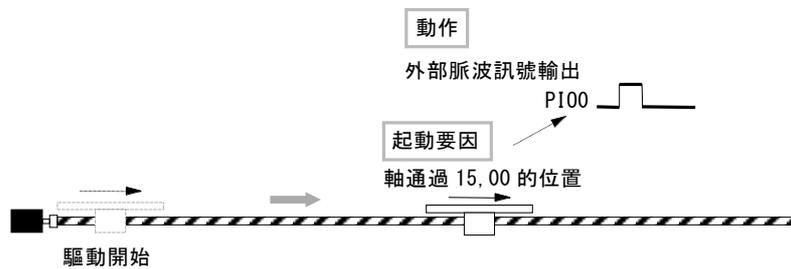


圖 2.7-8 同步動作實例 1

【程式例】

```

// 驅動設定 (1000 PPS 的等速驅動設定)
WR6 ← 1200h 寫 // 初速度 8M PPS (規格最大)
WR7 ← 007Ah 寫
WR0 ← 0004h 寫

WR6 ← 03E8h 寫 // 驅動速度 1000 PPS
WR7 ← 0000h 寫
WR0 ← 0005h 寫

WR6 ← 0000h 寫 // 理論位置計數器 0
WR7 ← 0000h 寫
WR0 ← 0009h 寫

// MRO 的設定
WR6 ← 3A98h 寫 // MRO 15000
WR7 ← 0000h 寫
WR0 ← 0010h 寫

// 多目的暫存器模式設定
WR6 ← 0000h 寫 // D1, D0 00 MOT1, 0 : MRO 的比較對象 理論位置計數器
WR0 ← 0020h 寫 // D3, D2 00 MOC1, 0 : MRO 的比較條件 ≥

// PIO 訊號設定 1
WR6 ← 0003h 寫 // D1, D0 11 POM1, 0 : PIO0 訊號 同步動作輸出
WR0 ← 0021h 寫

// PIO 訊號設定 2
WR6 ← 0070h 寫 // D0 0 POL : PIO0 脈波訊號的邏輯 正理論
WR0 ← 0022h 寫 // D6~D4 111 PW2~0 : 脈波寬 1msec (時脈 16MHz 時)

// 同步動作組
// 同步動作 SYNC0 設定
WR6 ← 0091h 寫 // D3~D0 0001 PREV3~0 : 同步動作的起動要因 MRO 比較為真
WR0 ← 0026h 寫 // D8~D4 01001 ACT4~0 : 同步動作的動作 同步脈波輸出

// SYNC0 有效
WR0 ← 0081h 寫

// 驅動開始
WR0 ← 0052h 寫 // +方向連續脈波驅動開始

```

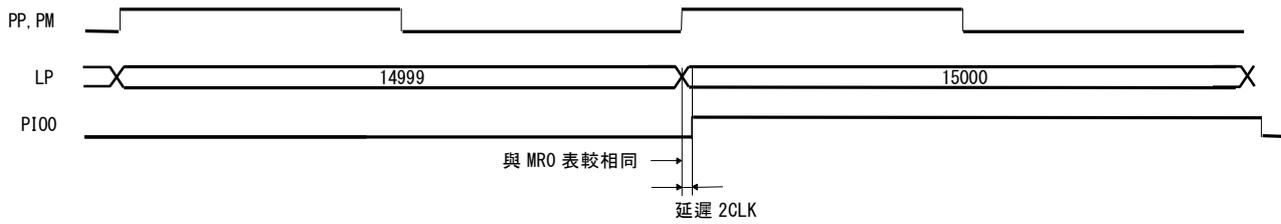


圖 2.7-9 同步動作實例 1 的時序

依照 2.6.7 項的起動要因發生的延遲時間為 1CLK、動作(Action)的延遲時間為 1CLK 之故、此同步動作的延遲時間為 2CLK (125nsec)。

■ 例 2 驅動中外部訊號輸入時儲存位置情報

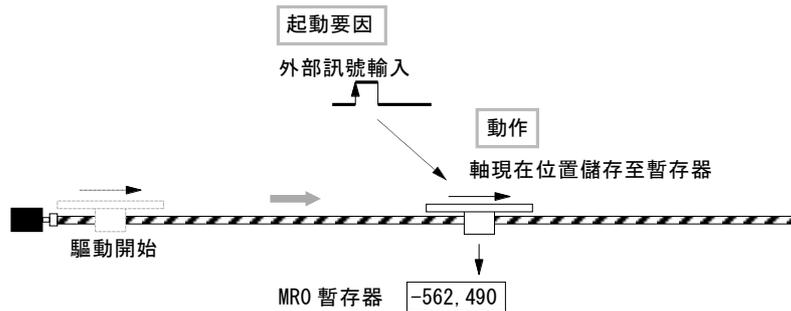


圖 2.7-10 同步動作實例 2

【程式例】

```
// 驅動的設定 (1000 PPS 的等速驅動的設定)
WR6 ← 1200h 寫 // 初速度 8M PPS (規格最大)
WR7 ← 007Ah 寫
WR0 ← 0004h 寫

WR6 ← 03E8h 寫 // 驅動速度 1000 PPS
WR7 ← 0000h 寫
WR0 ← 0005h 寫

WR6 ← 0000h 寫 // 理論位置計數器 0
WR7 ← 0000h 寫
WR0 ← 0009h 寫

// PIO 訊號設定 1
WR6 ← 0000h 寫 // D1, D0 00 POM1, 0 : PIO0 訊號 汎用・同步輸入
WR0 ← 0021h 寫

// 中斷設定
WR1 ← 1000h 寫 // D12 1 SYNC0 : 同步動作 SYNC0 起動時

// 同步動作組
// 同步動作 SYNC0 設定
WR6 ← 005Ah 寫 // D3~D0 1010 PREV3~0 : 同步動作的起動要因 PIOn 輸入 ↑
WR0 ← 0026h 寫 // D8~D4 00101 ACT4~0 : 同步動作的動作 儲存 LP → MRn

// SYNC0 有効
WR0 ← 0081h 寫

// 驅動開始
WR0 ← 0052h 寫 // +方向連續脈波驅動開始
```

↓

SYNC0 起動, 中斷發生

↓

```
//的讀取 MR0 中儲存的理論位置計數器值
WR0 ← 0034h 寫
RR6 → 讀
RR7 → 讀
```

依照 2.6.7 項起動要因發生的延遲時間最小為 0、最大為 1CLK、動作(Action)的延遲時間為 1CLK 之故、此同步動作的延遲時間最小為 1CLK (62.5nsec)、最大為 2CLK (125nsec)。

■ 例 3 驅動中、指定位置 A(10000)到指定位置 B(55000)止經過的時間

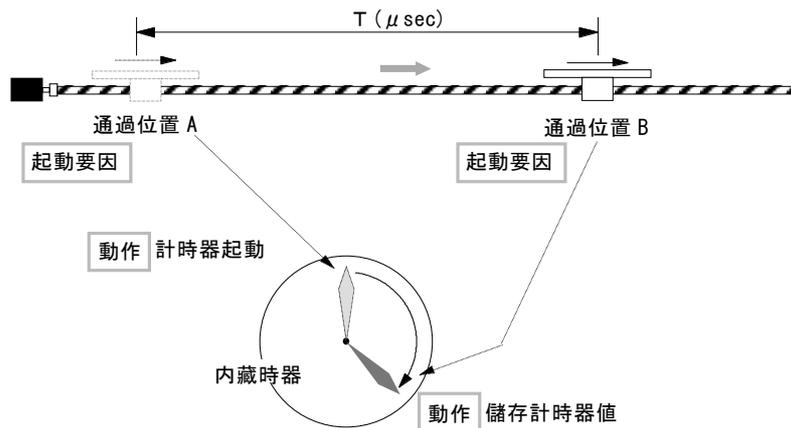


圖 2.7-11 同步動作實例 3

【程式例】

```
// 驅動設定 (10K PPS 等速驅動設定)
WR6 ← 1200h 寫 // 初速度 8M PPS (規格最大)
WR7 ← 007Ah 寫
WR0 ← 0004h 寫

WR6 ← 2710h 寫 // 驅動速度 10K PPS
WR7 ← 0000h 寫
WR0 ← 0005h 寫

WR6 ← 0000h 寫 // 理論位置計數器 0
WR7 ← 0000h 寫
WR0 ← 0009h 寫

// MRn 暫存器設定指定位置
// 設定 MR0 (指定位置 A : 10000)
WR6 ← 2710h 寫 // MR0 10000
WR7 ← 0000h 寫
WR0 ← 0010h 寫

// 設定 MR1 (指定位置 B : 55000)
WR6 ← D6D8h 寫 // MR1 55000
WR7 ← 0000h 寫
WR0 ← 0010h 寫

// 設定計時器值
WR6 ← FFFFh 寫 // 計時器值 2147483647 (最大值)
WR7 ← 7FFFh 寫
WR0 ← 0016h 寫

// 設定中斷
WR1 ← 2000h 寫 // D13 1 SYNC1 : 同步動作 SYNC1 起動時

// 設定多目的暫存器模式
WR6 ← 0000h 寫 // D1, D0 00 MOT1, 0 : MR0 的比較對象 理論位置計數器
// D3, D2 00 MOC1, 0 : MR0 的比較條件 ≥
// D5, D4 00 MIT1, 0 : MR1 的比較對象 理論位置計數器
// D7, D6 00 MIC1, 0 : MR1 的比較條件 ≥

WR0 ← 0020h 寫

// 設定同步動作
// 設定同步動作 SYNC0
WR6 ← 0151h 寫 // D3~D0 0001 PREV3~0 : 同步動作之起動要因 MRn 比較為真
// D8~D4 10101 ACT4~0 : 同步動作的動作 計時器始動

WR0 ← 0026h 寫

// 設定同步動作 SYNC1
WR6 ← 0071h 寫 // D3~D0 0001 PREV3~0 : 同步動作的起動要因 MRn 比較為真
// D8~D4 00111 ACT4~0 : 同步動作的動作 儲存 CT → MRn

WR0 ← 0027h 寫

// SYNC0, 1 有効
WR0 ← 0083h 寫
```

```
// 驅動開始  
WRO ← 0052h 寫
```

```
// +方向連續脈波驅動開始
```



SYNC0 起動, 中斷發生



```
// 讀取 MR1 儲存的計時器值  
WRO ← 0035h 寫  
RR6 → 讀  
RR7 → 讀
```

```
// 計時器停止  
WRO ← 0074h 寫
```

2.7.7 同步動作的延遲時間

同步動作的延遲時間合計為、下表所示從起動要因發生開始的延遲到動作(Action)為止的延遲。

■ 起動要因發生開始的延遲

1CLK=62.5nsec (CLK=16MHz 時)

表 2.7-7 起動要因發生開始的延遲

起動要因		延遲開始的定義	延遲時間 (CLK)		
			最小	標準	最大
MRn 比較為真	理論位置計數器	理論位置計數器與 MRn 值之比較條件為一致時的驅動脈波的↑開始		1	
	實際位置計數器	實際位置計數器與 MRn 值之比較條件為一致時的 ECA/B 輸入訊號的↓開始	2		3
	現在速度	現在速度與 MRn 值之比較條件為一致時開始		1	
	現在計時器值	現在計時器值與 MRn 值之比較條件為一致時開始		1	
計時器到時		現在計時器值到指定值時開始		0	
驅動開始		驅動指令寫入時的 WRN 訊號↑開始	2		3
加減速驅動的等速域開始		CNST 訊號↑開始		0	
加減速驅動的等速域結束		CNST 訊號↓開始		0	
驅動結束		最終驅動脈波的 Low 位準結束開始		1	
分割脈波開始		第 1 SPLTP 訊號↑開始 (開始脈波有效時)		0	
分割脈波結束		最終 SPLTP 訊號↓開始 (正理論時)		2	
分割脈波輸出		SPLTP 訊號↑開始 (正理論時)		0	
PIO _n 輸入↑		PIO _n 訊號↑開始 (內藏濾波無效時)	0		1
PIO _n 輸入↓		PIO _n 訊號↓開始 (內藏濾波無效時)	0		1
PIO _n 輸入 Low 且 PIO _(n+4) ↑		PIO _(n+4) 訊號↑開始 (內藏濾波無效時)	0		1
PIO _n 輸入 Hi 且 PIO _(n+4) ↑		PIO _(n+4) 訊號↑開始 (內藏濾波無效時)	0		1
PIO _n 輸入 Low 且 PIO _(n+4) ↓		PIO _(n+4) 訊號↓開始 (內藏濾波無效時)	0		1
PIO _n 輸入 Hi 且 PIO _(n+4) ↓		PIO _(n+4) 訊號↓開始 (內藏濾波無效時)	0		1
起動指令		同步動作起動指令寫入時 WRN 訊號↑開始	1		2

■ 動作(Action)為止的延遲

1CLK=62.5nsec (CLK=16MHz 時)

表 2.7-8 動作(Action)為止的延遲

動作	延遲結束的定義	延遲時間 (CLK)
載入 MRn → DV	MRn 值載入 DV 為止	1
載入 MRn → TP	MRn 值載入 TP 為止	1
載入 MRn → SP1	MRn 值載入 SP1 為止	1
載入 MRn → LP (SYNC0), RP (SYNC1), SV (SYNC2), AC (SYNC3)	MRn 值載入 LP (SYNC0), RP (SYNC1), SV (SYNC2), AC (SYNC3) 為止	1
儲存 LP → MRn	LP 的值存入 MRn 為止	1
儲存 RP → MRn	RP 的值存入 MRn 為止	1
儲存 CT → MRn	CT 的值存入 MRn 為止	1
儲存 CV (SYNC0), CA (SYNC1) → MRn	CV (SYNC0), CA (SYNC1) 的值存入 MRn 為止	1
同步脈波 PION 輸出	同步脈波 PION 訊號 ↑ 為止	1
相對位置驅動起動	第 1 驅動脈波 ↑ 為止	3
反相對位置驅動起動	第 1 驅動脈波 ↑ 為止	3
絕對位置驅動起動	第 1 驅動脈波 ↑ 為止	3
+方向連續脈波驅動起動	第 1 驅動脈波 ↑ 為止	3
-方向連續脈波驅動起動	第 1 驅動脈波 ↑ 為止	3
MRn 值的移動脈波數做相對位置驅動	第 1 驅動脈波 ↑ 為止	4
往 MRn 值的終點做絕對位置驅動	第 1 驅動脈波 ↑ 為止	4
驅動減速停止	開始減速為止	(※1)
驅動立即停止	停止驅動為止	(※1)
驅動速度增加	開始往變更後的速度增加速度為止	1
驅動速度減少	開始往變更後的速度減少速度為止	1
計時器始動	計時器開始為止	1
計時器停止	計時器停止為止	1
分割脈波開始	SPLTP 訊號 ↑ 為止 (有開始脈波的時候)	(※2)
分割脈波停止	SPLTP 訊號 ↓ 為止	(※3)
中斷	INTN 訊號 ↓ 為止	1

(※1)現在輸出中的1驅動脈波結束為止的時間

(※2)分割脈波因和驅動脈波同步之故、最大為1驅動脈波周期的延遲。

(※3)現在輸出中的分割脈波結束為止的時間

■ 延遲計算例

例如、起動要因「PION 輸入 ↑」到動作(Action)「存入 LP → MRn」為止的延遲時間為、起動要因「PION 輸入 ↑」延遲時間(0 ~ 1CLK)及動作(Action)「存入 LP → MRn」延遲時間(1CLK)的合計、最小 1CLK 到最大 2CLK。CLK=16MHz 時、最小 62.5nsec 到最大 125nsec。

■ 其他 SYNC 起動的延遲

執行其他 SYNC 起動時、自同步動作組的動作(Action)起動相比、1CLK 的延遲下動作(Action)起動。

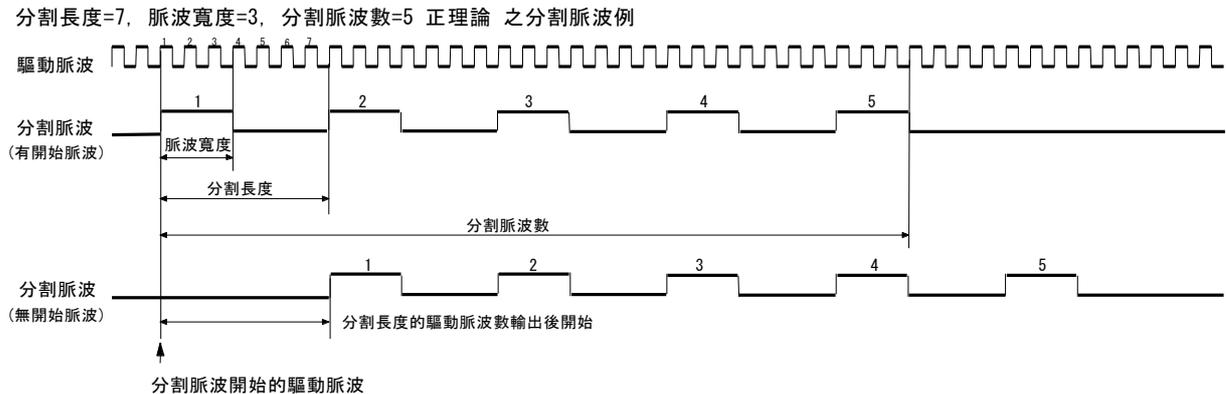
2.8 分割脈波

驅動中驅動脈波與同步下的分割脈波輸出機能。

馬達的回轉或軸移動中、想要在一定的脈波間隔下同時做別的工作時的便利機能。

上述需求可設定分割脈波的脈波寬、分割長(周期)、分割脈波數來達成。另、可以指定脈波的理论位準、開始脈波有/無。分割脈波訊號由 SPLTP(端子編號:64)輸出。

驅動中分割脈波的開始是依指令、例如以同步動作執行。藉由同步動作時、可由指定的位置計數器的值來開始執行、外部訊號的↑來開始執行。



2.8.1 分割脈波的設定

欲執行分割脈波時、需設定以下的參數及模式。

■ 分割長、脈波寬的設定

分割長、脈波寬的設定是、以分割脈波設定 1 指令(17h)來進行。設定 WR6 暫存器為分割長、WR7暫存器為脈波寬。分割長及脈波寬的設定單位為驅動脈波數。

分割脈波的機能需求、請設定分割長 > 脈波寬。

可以設定的分割長為 2~65535、脈波寬為 1~65534 的範圍。

設定後的內容、可用分割脈波設定 1 的讀取指令(47h)來確認。

分割脈波動作中也可以變更分割長(周期)、脈波寬的設定內容。

■ 分割脈波數的設定

分割脈波數的設定是、以分割脈波設定 2 指令(18h)來進行。設定 WR6 暫存器。

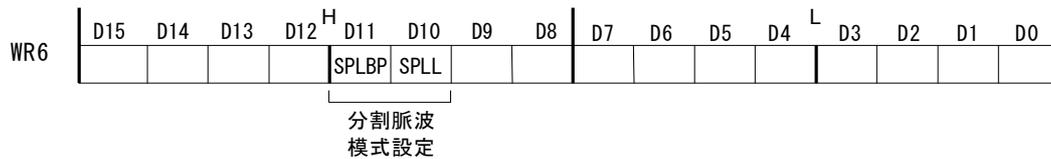
可以設定範圍在 0~65535。設定值為 0 時變為無限。開始後、分割脈波停止指令或驅動停止為止、分割脈波會持續輸出。

分割脈波數在分割脈波動作中其設定內容仍可以變更。

■ 分割脈波的模式設定

分割脈波的動作模式由PIO訊號設定2·其他設定指令(22h)來設定。

分割脈波開始時的開始脈波有或無、及脈波的輸出理論、可由 WR6 暫存器的 D10、D11 位元來設定。



分割脈波的脈波理論由 D10 位元(SPLL)來設定。

如下圖所示、0 指定為正理論、1 指定為負理論。

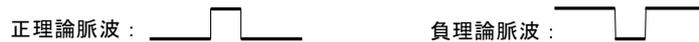


圖 2.8-2 分割脈波的脈波理論

分割脈波的開始脈波的有或無由 D11 位元(SPLBP)來設定。

D11 位元(SPLBP)中1指定為開始脈波有、設為 0 則開始脈波無。

開始脈波設定為有的時候、分割脈波開始後、下一個驅動脈波開始輸出分割脈波。開始脈波設定為無的時候、分割脈波開始後、分割長份的驅動脈波數輸出結束後、最初的分割脈波輸出。

2.8.2 分割脈波開始／停止

■ 分割脈波開始

分割脈波開始指令(75h)及依靠同步動作開始。

指令寫入當中、或同步動作的動作(Action)已經開始的時候，下一個驅動脈波會成為分割脈波的開始驅動脈波。

■ 分割脈波的停止

分割脈波輸出是、以下 3 個要因中、哪一個發生而停止。

- 指定分割脈波數全不輸出完畢
- 執行分割脈波停止指令、或同步動作的動作(Action)停止時
- 驅動停止時

指定之分割脈波數輸出完成而停止時、指定之分割脈波數最後的分割脈波變為 OFF 的狀態時停止。

以分割脈波停止指令(76h)或因同步動作而停止分割脈波時、分割脈波若處於 ON 的狀態下、會在該分割脈波寬輸出完成後停止。停止執行後、分割脈波會在 OFF 狀態、分割脈波停止指令及同步動作執行的時序下停止。

因驅動停止而停止分割脈波輸出時、無關於分割脈波的輸出狀態、因驅動停止的時序使分割脈波變為 OFF 的狀態而停止。

■ 主狀態暫存器

主狀態暫存器 RR0 的 D13 位元(分割)可確認分割脈波式否動作中。

D13 位元(分割)為1的時候分割脈波在動作中、為 0 的時候分割脈波在停止中。

2.8.3 同步動作之分割脈波

分割脈波的動作可伴隨同步動作。

同步動作的起動要因、可指定為「分割脈波開始時」、「分割脈波輸出時」、「分割脈波結束時」的3種。

同步動作的動作(Action)、可指定「分割脈波開始」、「分割脈波停止」、「多目的暫存器資料載入分割脈波資料(分割長、脈波寬)」、的3種。

這些機能的詳細、請參考 2.6 節。

2.8.4 分割脈波與中斷發生

相關的分割脈波動作可以用來產生中斷。

設定 WR1 暫存器的 D10 及 D11 位元。

D10 位元(SPLTP)設為1時、分割脈波的脈波↑時發生中斷。(分割脈波正理論設定時)

D11 位元(SPLTE)設為1時、分割脈波的動作結束時發生中斷。

有關中斷機能、請參考 2.10 節。

2.8.5 分割脈波的注意點

(1) 分割脈波有開始脈波的時候、僅第一脈波的輸出時序不同。詳細請參考9.5節。

(2) 分割脈波動作中、在指定的分割脈波數完成前因停止指令等而停止時、停止後再重新開始分割脈波時、分割脈波數的計數由1開始。

2.8.6 分割脈波的實例

■ 例1 驅動開始到分割脈波開始

分割脈波開始指令發出後、開始驅動、驅動的同時將分割脈波輸出例。

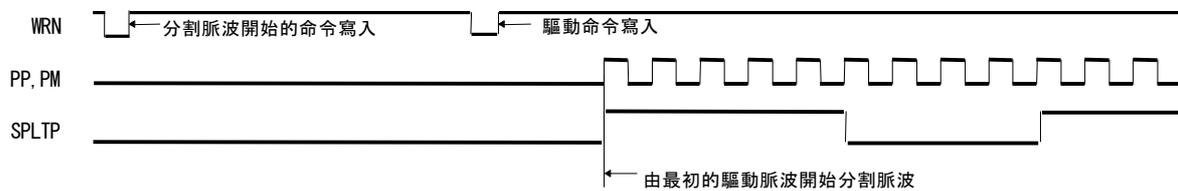


圖 2.8-3 以驅動開始輸出分割脈波的時序

【程式例】

```
// 驅動的設定(1000 PPS 的等速驅動的設定)
WR6 ← 1200h 寫 // 初速度 8M PPS (規格最大)
WR7 ← 007Ah 寫
WR0 ← 0004h 寫

WR6 ← 03E8h 寫 // 驅動速度 1000 PPS
WR7 ← 0000h 寫
WR0 ← 0005h 寫

WR6 ← 0000h 寫 // 理論位置計數器 0
WR7 ← 0000h 寫
WR0 ← 0009h 寫

// 分割脈波的設定
// 分割長、脈波寬的設定
WR6 ← 0009h 寫 // 分割長 9
WR7 ← 0005h 寫 // 脈波寬 5
WR0 ← 0017h 寫

// 分割脈波數的設定
WR6 ← 000Ah 寫 // 分割脈波數 10
WR0 ← 0018h 寫

// 分割脈波理論、開始脈波的設定
WR6 ← 0800h 寫 // D10 0 SPLL : 脈波理論 正
// D11 1 SPLBP : 開始脈波 有
WR0 ← 0022h 寫

// 分割開始(於驅動開始前發出分割脈波開始指令)
WR0 ← 0075h 寫

// 驅動開始
WR0 ← 0052h 寫 // +方向連續脈波驅動開始
```

驅動開始後、最初的驅動脈波成為分割脈波的開始驅動脈波。

發出分割脈波開始指令後、驅動開始前分割脈波不會輸出、主狀態暫存器 RR0 的 D13 位元(分割)為發出分割脈波開始指令的時候變為1。

■ 例 2 指定位置 5000 起開始分割脈波

驅動開始後、理論位置到達 5000 時開始分割脈波的實例。使用同步動作機能執行。

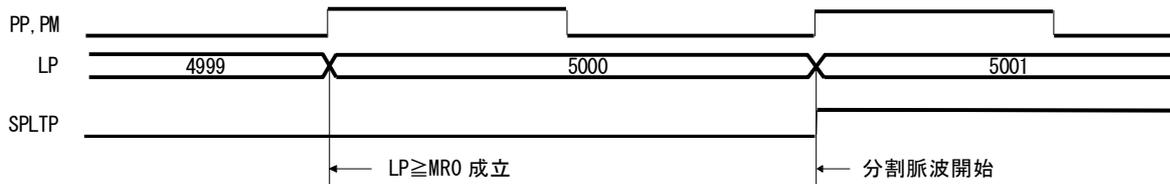


圖 2.8-4 以 MRn 比較輸出分割脈波的時序

【程式例】

```
// 驅動設定 (設定 1000 PPS 等速驅動)
WR6 ← 1200h 寫 // 初速度 8M PPS (規格最大)
WR7 ← 007Ah 寫
WR0 ← 0004h 寫

WR6 ← 03E8h 寫 // 驅動速度 1000 PPS
WR7 ← 0000h 寫
WR0 ← 0005h 寫

WR6 ← 0000h 寫 // 理論位置計數器 0
WR7 ← 0000h 寫
WR0 ← 0009h 寫

// 設定分割脈波
// 設定分割長、脈波寬
WR6 ← 0008h 寫 // 分割長 8
WR7 ← 0005h 寫 // 脈波寬 5
WR0 ← 0017h 寫

// 設定分割脈波數
WR6 ← 000Ah 寫 // 分割脈波數 10
WR0 ← 0018h 寫

// 設定分割脈波理論、開始脈波
WR6 ← 0800h 寫 // D10 0 SPLL : 脈波理論 正
// D11 1 SPLBP : 開始脈波 有
WR0 ← 0022h 寫

// 設定多目的暫存器
// MRO 的設定
WR6 ← 1388h 寫 // MRO 5000
WR7 ← 0000h 寫
WR0 ← 0010h 寫

// 設定多目的暫存器模式
WR6 ← 0000h 寫 // D1, D0 00 MOT1, 0 : MRO 的比較對象 理論位置計數器
// D3, D2 00 MOC1, 0 : MRO 的比較條件 ≥
WR0 ← 0020h 寫

// 設定同步動作
// 同步動作 SYNC0 設定
WR6 ← 0171h 寫 // D3~D0 0001 PREV3~0 : 同步動作的起動要因 MRn 比較為真
// D8~D4 10111 ACT4~0 : 同步動作的動作 分割脈波開始
WR0 ← 0026h 寫

// SYNC0 有效
WR0 ← 0081h 寫

// 驅動開始
WR0 ← 0052h 寫 // + 方向連續脈波驅動開始
```

比較值為 5000、比較條件為 \geq 時、開始分割脈波，理論位置計數器的值如圖到達 5001。因比較條件為真之故下一個驅動脈波成為分割脈波的開始驅動脈波。

■ 例 3 S 形加減速驅動的等速區間中、輸出分割脈波

S 形加減速驅動的等速區間中、輸出分割脈波的實例。使用同步動作機能執行。

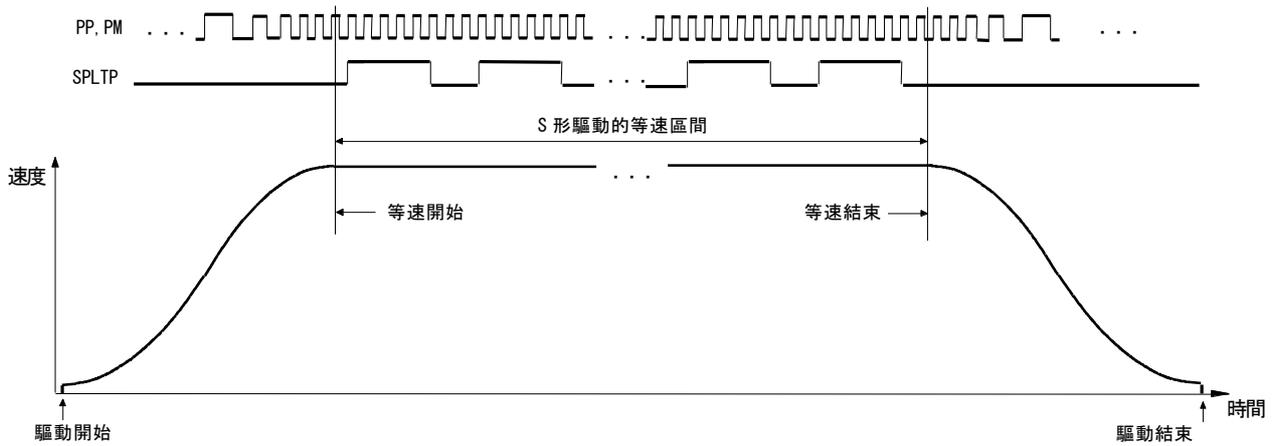


圖 2.8-5 S 形加減速驅動的等速區間中輸出分割脈波

【程式例】

```
//設定 S 形加減速驅動
WR6 ← 000Ah 寫           // 初速度 10 PPS
WR7 ← 0000h 寫
WR0 ← 0004h 寫

WR6 ← 0FA0h 寫           // 驅動速度 4000 PPS
WR7 ← 0000h 寫
WR0 ← 0005h 寫

WR6 ← FFFFh 寫           // 加速度 536870911(最大值)
WR7 ← 1FFFh 寫
WR0 ← 0002h 寫

WR6 ← A048h 寫           // 加速度增加率 893K PPS/SEC2
WR7 ← 000Dh 寫
WR0 ← 0000h 寫

WR6 ← 9C40h 寫           // 移動脈波數 40000
WR7 ← 0000h 寫
WR0 ← 0006h 寫

WR6 ← 0000h 寫           // 理論位置計數器 0
WR7 ← 0000h 寫
WR0 ← 0009h 寫

WR3 ← 0004h 寫           // D3      1 SACC : S 形加減速

//設定分割脈波
//設定分割長、脈波寬
WR6 ← 0008h 寫           // 分割長 8
WR7 ← 0005h 寫           // 脈波寬 5
WR0 ← 0017h 寫

//設定分割脈波數
WR6 ← 0000h 寫           // 分割脈波數 無限
WR0 ← 0018h 寫

//設定分割脈波理論、開始脈波
WR6 ← 0800h 寫           // D10    0 SPLL : 脈波理論 正
WR0 ← 0022h 寫           // D11    1 SPLBP : 開始脈波 有

//設定同步動作
//設定同步動作 SYNC0
WR6 ← 0174h 寫           // D3~D0  0100 PREV3~0 : 同步動作的起動要因 驅動等速域開始
WR0 ← 0026h 寫           // D8~D4  10111 ACT4~0 : 同步動作的動作 分割脈波開始
// D15    0 REP      : 同步動作的重複 務必設為無效
```

```
// 同步動作 SYNC1 設定
WR6 ← 0185h 寫

// D3~D0 0101 PREV3~0 : 同步動作的起動要因 驅動等速域結束
// D8~D4 11000 ACT4~0 : 同步動作的動作 分割脈波停止
// D15 0 REP : 同步動作的重複 務必設為無効

WRO ← 0027h 寫

// SYNC0, 1 有効
WRO ← 0083h 寫

// 驅動開始
WRO ← 0050h 寫

// 相對位置驅動開始
```

■ 例 4 由位置 5000 開始分割脈波、位置 10000 起改變分割長、脈波寬輸出分割

從理論位置 5000 開始分割脈波、理論位置 10000 起變更分割長、脈波寬，輸出剩下的脈波數、分割脈波的實例。使用同步動作機能執行。

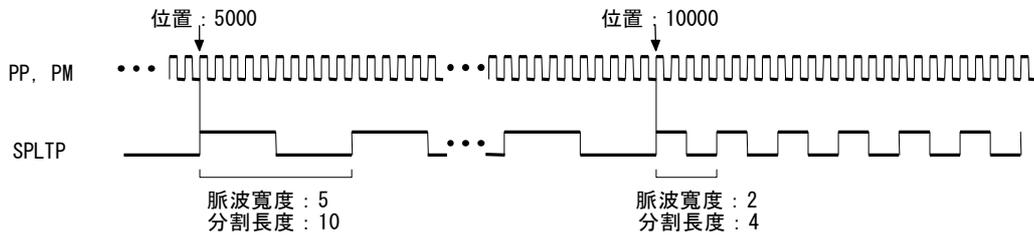


圖 2.8-6 驅動中在指定位置變更分割長度、脈波寬度

【程式例】

```
// 驅動的設定(設定 1000 PPS 的等速驅動)
WR6 ← 1200h 寫 // 初速度 8M PPS (最大值)
WR7 ← 007Ah 寫
WR0 ← 0004h 寫

WR6 ← 03E8h 寫 // 驅動速度 1000 PPS
WR7 ← 0000h 寫
WR0 ← 0005h 寫

WR6 ← 0000h 寫 // 理論位置計數器 0
WR7 ← 0000h 寫
WR0 ← 0009h 寫

WR6 ← 2EE0h 寫 // 移動脈波數 12000
WR7 ← 0000h 寫
WR0 ← 0006h 寫

// 設定分割脈波
// 設定分割長、脈波寬
WR6 ← 000Ah 寫 // 分割長 10
WR7 ← 0005h 寫 // 脈波寬 5
WR0 ← 0017h 寫

// 設定分割脈波數
WR6 ← 0320h 寫 // 分割脈波數 800
WR0 ← 0018h 寫

// 設定分割脈波理論、開始脈波
WR6 ← 0800h 寫 // D10 0 SPLL : 脈波理論 正
// D11 1 SPLBP : 開始脈波 有
WR0 ← 0022h 寫

// 設定多目的暫存器
// MRO 的設定
WR6 ← 1387h 寫 // MRO 4999
WR7 ← 0000h 寫
WR0 ← 0010h 寫

// 設定 MR1
WR6 ← 2710h 寫 // MR1 10000
WR7 ← 0000h 寫
WR0 ← 0011h 寫

// 設定 MR2
WR6 ← 0004h 寫 // 分割長 4
WR7 ← 0002h 寫 // 脈波寬 2
WR0 ← 0012h 寫

// 設定多目的暫存器模式
WR6 ← 0000h 寫 // D1, D0 00 MOT1, 0 : MRO 的比較對象 理論位置計數器
// D3, D2 00 MOC1, 0 : MRO 的比較條件 ≥
// D5, D4 00 MIT1, 0 : MR1 的比較對象 理論位置計數器
// D7, D6 00 M1C1, 0 : MR1 的比較條件 ≥

WR0 ← 0020h 寫
```

```

//設定同步動作
//設定同步動作 SYNC0
WR6 ← 0171h 寫          // D3~D0    0001 PREV3~0 : 同步動作的起動要因 MRn 比較為真
                          // D8~D4    10111 ACT4~0  : 同步動作的動作 分割脈波開始

WR0 ← 0026h 寫

//設定同步動作 SYNC1
WR6 ← 0201h 寫          // D3~D0    0001 PREV3~0 : 同步動作的起動要因 MRn 比較為真
                          // D8~D4    00000 ACT4~0 : 同步動作的動作  NOP
                          // D11~D9   001  SNC+3, 2, 1 : 他 SYNC 起動  SNC+1 起動

WR0 ← 0027h 寫

//設定同步動作 SYNC2
WR6 ← 0030h 寫          // D3~D0    0001 PREV3~0 : 同步動作的起動要因  NOP
                          // D8~D4    00011 ACT4~0 : 同步動作的動作  載入 MRn → SP1
                          // D11~D9   001  SNC+3, 2, 1 : 他 SYNC 起動  SNC+1 起動

WR0 ← 0027h 寫

// SYNC2~0 有效
WR0 ← 0087h 寫

// 驅動開始
WR0 ← 0050h 寫          // 相對位置驅動開始

```

本例設定為、位置到 4999 的時候開始分割脈波、實際上脈波是從 5000 開始。

【注意】

- 如本例所示、應用同步動作在分割脈波動作中變更分割長或脈波數的時候一定要注意。因變更的時間點關係、會有變更前後的分割脈波不依循設定值動作的情形。

2.9 汎用輸出入訊號

本IC有、8點汎用輸出入訊號(PIO7~0)。

另、若滿足以下條件時具特定機能的輸入訊號也可以用來做汎用輸入訊號使用。

- ① 步使用輸入訊號的特定機能時、該機能設為無効後可用於汎用輸入訊號
- ② 本IC使用8位元的資料匯流排的時候、高位元之D15~D8不做資料匯流排使用即可以作為汎用輸入訊號

2.9.1 PIO_n訊號

PIO_n訊號如下所示可使用於各種目的輸出入訊號使用。

- 1) 汎用輸入訊號
- 2) 汎用輸出訊號
- 3) 同步動作的起動要因之輸入訊號
- 4) 同步動作的動作(Action)同步脈波輸出訊號
- 5) 驅動狀態輸出訊號
- 6) 多目的暫存器之比較結果輸出訊號
- 7) 外部訊號操作驅動之輸入訊號

■ PIO_n訊號的機能設定

PIO_n訊號的機能設定是、以PIO訊號設定1指令(21h)進行。

WR6	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
	P7M1	P7M0	P6M1	P6M0	P5M1	P5M0	P4M1	P4M0	P3M1	P3M0	P2M1	P2M0	P1M1	P1M0	P0M1	P0M0
	PIO7 訊號		PIO6 訊號		PIO5 訊號		PIO4 訊號		PIO3 訊號		PIO2 訊號		PIO1 訊號		PIO0 訊號	

WR6暫存器之各PIO_n訊號依使用目的設定所對應的2位元。

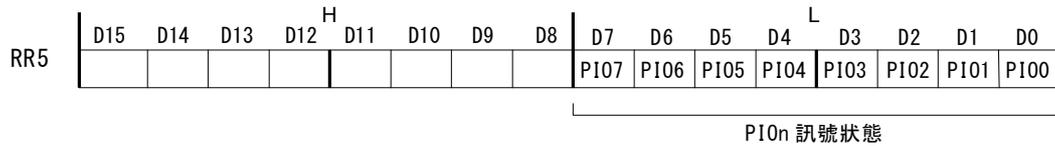
各PIO_n訊號的2位元的設定值之對應機能如下顯示。

表 2.9-1 PIO_n 訊號的機能設定

		(n:0~7)
PnM1 位元	PnM0 位元	機能
0	0	汎用輸入 PIO7~0 訊號變為輸入狀態。 同步動作是在、可以在訊號↑或↓起動同步動作。 外部驅動操作是在、可以利用 PIO4,5 訊號做相對位置驅動或連續脈波驅動的起動。
0	1	汎用輸出 PIO7~0 訊號變為輸出狀態。
1	0	驅動狀態輸出 PIO7~0 訊號變為輸出狀態、驅動狀態輸出。
1	1	同步脈波・MR _n 比較輸出 PIO7~0 訊號變為輸出狀態。輸出 PIO3~0 為同步脈波、PIO7~4 為 MR _n 的比較值。

■ PIO_n訊號的讀取

PIO_n訊號的訊號位準是、不拘於輸出入、隨時可以從RR5暫存器的D7~D0位元(PIO7~PIO0)讀取。訊號為Low位準的時候為0、Hi位準的時候為1。



■ 汎用輸入

輸入訊號的機能分為、汎用輸入訊號、同步輸入訊號、依外部訊號驅動操作的輸入訊號3種。
使用的PIO_n訊號將該2位元設為0,0、再設定PIO訊號設定1指令(21h)。

汎用輸入訊號使用

PIO7~0訊號的訊號位準顯示於RR5暫存器的D7~D0位元(PIO7~PIO0)。訊號在Low位準時為0、Hi位準時為1。

同步輸入訊號使用

可將PIO_n訊號的輸入變化做為同步動作的起動要因。
有關同步動作請參考2.6節。

外部訊號用來做驅動操作的輸入訊號使用

相對位置驅動或連續脈波驅動、不用指令、可藉由PIO_n訊號輸入來起動。
利用PIO4訊號、PIO5訊號的輸入狀態及輸入變化來起動驅動。
有關以外部訊號來操作驅動的說明請參考2.12.1項。

■ 汎用輸出

使用的PIO_n訊號該2位元設為0,1、在設定PIO訊號設定1指令(21h)。

往PIO_n訊號寫入的資料、請寫入WR4暫存器。WR4暫存器的D7~0位元寫入值會在、PIO7~0訊號上輸出。在D7~0位元寫入0時為Low位準、寫入1時為Hi位準輸出。

■ 驅動狀態輸出

驅動中的狀態可以藉由PIO_n訊號輸出。
使用到的PIO_n訊號該2位元設為1,0、再設定PIO訊號設定1指令(21h)。

驅動中、加速中、減速中等驅動狀態經由PIO_n訊號輸出。
有關驅動狀態的輸出說明請參考2.12.7項。

■ 同步脈波・MR_n比較輸出

使用的PIO_n訊號該2位元設為1,1、在設定PIO訊號設定1指令(21h)。

做為同步脈波輸出訊號使用

作為同步動作的動作(Action), PIO0~PIO3訊號可接受將同步脈波輸出。
有關同步動作之□明請參考2.6節。

做為MR_n比較輸出訊號使用

MR_n暫存器之比較結果可藉由PIO_n訊號輸出。
MR0~MR3比較輸出、由PIO4~PIO7訊號輸出。
有關MR_n暫存器的說明請參考2.4節。

2.9.2 其他輸入訊號

下表所示、有關PIOn訊號以外的輸入訊號、在不使用其訊號特有功能時、可以用做汎用輸入訊號。輸入訊號的訊號位準由RR3暫存器來顯示。訊號為Low位準的時候0、Hi位準的時候為1。

下表中詳列可用於汎用輸入訊號的輸入訊號。

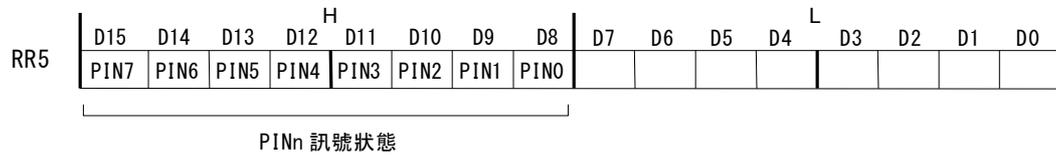
表 2.9-2 可用於汎用輸入訊號的輸入訊號

輸入訊號 (端子編號)	輸入訊號持有的機能	狀態顯示 RR3暫存器的位元	RR3 狀態顯示
STOP0 (42)	驅動停止訊號	D0位元 (STOP0)	0:Low位準
STOP1 (40)	驅動停止訊號	D1位元 (STOP1)	
STOP2 (39)	驅動停止訊號	D2位元 (STOP2)	1:Hi位準
ECA (37)	編碼器A相訊號	D3位元 (ECA)	
ECB (38)	編碼器B相訊號	D4位元 (ECB)	
INPOS (48)	伺服用定位完成訊號	D5位元 (INPOS)	
ALARM (49)	伺服用異常訊號	D6位元 (ALARM)	
LMTP (43)	+方向硬體極限訊號	D7位元 (LMTP)	
LMTM (44)	-方向硬體極限訊號	D8位元 (LMTM)	

2.9.3 8位元資料匯流排時的上位資料訊號

本IC在8位元資料匯流排模式(H16L8=Low)下使用時、不使用的上位資料匯流排D15~D8訊號可以當成汎用輸入訊號使用。

訊號的位準在RR5暫存器的D15~D8位元(PIN7~PIN0)顯示。訊號Low位準時為0、Hi位準時為1。



2.10 計時器

本 IC 有內建一個計時器。可以用 $1 \mu \text{sec}$ 的單位在 $1 \sim 2,147,483,647 \mu \text{sec}$ (CLK=16MHz 時) 的範圍下設定。依據使用本 IC 具有的同步動作機能、搭配馬達・驅動及計時器機能可達到各式各樣良好精度的動作。如以下實例所示。

■ 驅動結束、指定時間後驅動開始

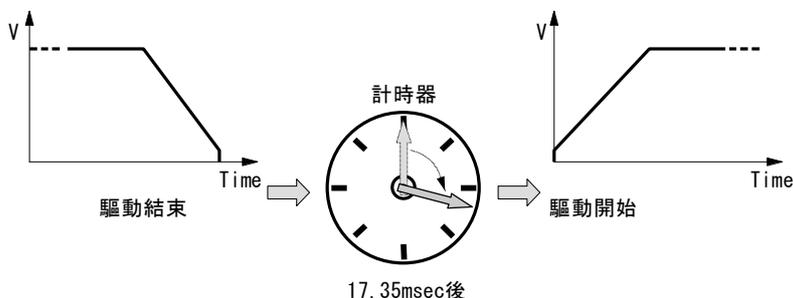


圖 2.10-1 計時器動作 例 1

■ 正確的時間周期下輸出驅動脈波。



圖 2.10-2 計時器動作 例 2

■ 加減速驅動時僅指定的時間下進行等速驅動後、減速停止。

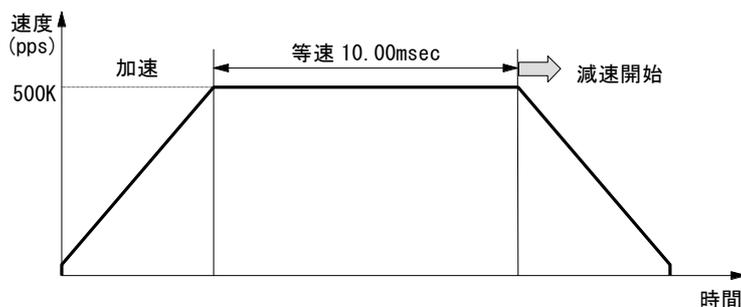


圖 2.10-3 計時器動作 例 3

2.10.1 計時器的動作

本 IC 持有 31 位元長的計時器計數器。計時器起動時、計時器計數器每 $1 \mu \text{sec}$ 間隔下、從 0 開始計時累積、當指定的計時器值到達後 (時間到) 停止。計時器動作模式若指定 "1 次" 則時間到了以後計時器動作就會停止。動作模式若設為 "重複"、時間到了以後計數器會再從 0 開始計時。計時器停止指令若由同步動作來的停止，則直到停止前動作會持續重複。計時器的時間到可設為同步動作的起動要因、驅動的開始、外部訊號的輸出等各式各樣的動作進行，有關同步動作的說明請參考 2.6 節。

另、時間到的時候也可以發出中斷訊號、也可以配合 CPU 側的時序進行處理。

2.10.2 計時器的設定

要使計時器動作、需設定計時器值及動作模式(1次/重複)。

■ 計時器值的設定

計時器值是以計時器值設定指令(16h)設定。把值設在WR6,7暫存器後、再將計時器值設定指令(16h)寫入WR0暫存器即可。計時器值的單位是 μsec 、設定範圍在1~2,147,483,647(參考5.2.22項)。

計時器動作中，計時器值在也可以變更。

■ 計時器的模式設定

計時器的動作模式是、設定 WR3 暫存器的 D14 位元(TMMD)。D14 位元 (TMMD) 設定 0 為一次、設定1為返複動作。

2.10.3 計時器的起動及停止

■ 計時器的始動

計時器的起動是、發出計時器始動指令(73h)、同步動作的動作(Action)中設定計時器始動編碼、依所定的同步動作的起動執行。

■ 計時器的停止

計時器在、動作模式為1次時，指定的計時器值到達後(時間到)停止。計時器動作中、發出計時器停止指令(74h)時、可依同步動作狀況停止。

動作模式為重複時、發出計時器停止指令(74h)後、依同步動作的情況停止。

2.10.4 計時器與同步動作

計時器可於同步動作中使用。

可指定「時間到」為同步動作的起動要因。「現在計時器值存入多目的暫存器」、「計時器始動」、「計時器停止」3種可指定為同步動作的動作(Action)、。這些機能的詳細說明請參考2.7節。

2.10.5 計時器動作狀態及現在計時器值的讀取

■ 現在計時器值的讀取

現在計時器值讀取指令(38h)可讀取動作中的現在計時器值。

計時器計數由0開始計數累積進行。動作當中隨時可以讀取計時器計數值。

計時器計數在計時器動作停止時清除為0。計時器結束、或是發出計時器停止指令後讀取現在計時器值時、讀到的值為0。

■ 主狀態暫存器

主狀態暫存器 RR0 的 D12 位元(計時器)可確認計時器的動作狀態。計時器起動時、D12 位元為1、顯示計時器動作中。

2.10.6 計時器發出中斷

計時器在時間到的時候可發出中斷訊號。WR1暫存器D9(計時器)位元設為1。

有關中斷機能說明請參考2.11節。

2.10.7 計時器的實例

■ 例 1 驅動結束、17.35msec 後驅動開始

相對位置驅動結束、在 17.35msec 後、再次起動相同的相對位置驅動的實例。使用同步動作機能執行。

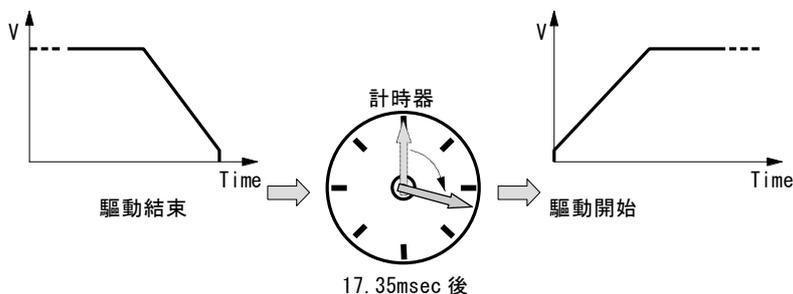


圖 2.1010-4 計時器動作 實例 1

【程式例】

```
//設定加減速驅動
WR6 ← 0190h 寫          // 初速度 400 PPS
WR7 ← 0000h 寫
WR0 ← 0004h 寫

WR6 ← 9C40h 寫          // 驅動速度 40K PPS
WR7 ← 0000h 寫
WR0 ← 0005h 寫

WR6 ← E848h 寫          // 加速度 125K PPS/SEC
WR7 ← 0001h 寫
WR0 ← 0002h 寫

WR6 ← 9C40h 寫          // 移動脈波數 40000
WR7 ← 0000h 寫
WR0 ← 0006h 寫

//設定計時器
// 單次計時器
WR3 ← 0000h 寫          // D14      0 TMMD : 計時器動作 1次

//設定計時器值
WR6 ← 43C6h 寫          // 計時器值 17350 μ sec
WR7 ← 0000h 寫
WR0 ← 0016h 寫

//設定同步動作
//設定同步動作 SYNC0
WR6 ← 0156h 寫          // D3~D0    0110 PREV3~0 : 同步動作的起動要因 驅動停止
WR0 ← 0026h 寫          // D8~D4    10101 ACT4~0 : 同步動作的動作 計時器始動

//設定同步動作 SYNC1
WR6 ← 00A2h 寫          // D3~D0    0010 PREV3~0 : 同步動作的起動要因 時間到
WR0 ← 0027h 寫          // D8~D4    01010 ACT4~0 : 同步動作的動作 相對位置驅動起動

// SYNC1~0 有効
WR0 ← 0083h 寫

// 驅動開始
WR0 ← 0050h 寫          // 相對位置驅動開始
```

■ 例 2 每 1msec 輸出已定的驅動脈波

每 1msec 重複相對位置驅動(20kpps×10 脈波的等速驅動)的實例。使用同步動作機能執行。

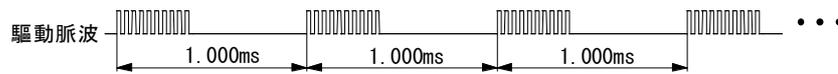


圖 2.10-5 計時器動作 實例 2

【程式例】

```
// 驅動設定 (設定 1000 PPS 等速驅動)
WR6 ← 1200h 寫 // 初速度 8M PPS (規格最大)
WR7 ← 007Ah 寫
WR0 ← 0004h 寫

WR6 ← 4E20h 寫 // 驅動速度 20K PPS
WR7 ← 0000h 寫
WR0 ← 0005h 寫

WR6 ← 000Ah 寫 // 移動脈波數 10
WR7 ← 0000h 寫
WR0 ← 0006h 寫

WR6 ← 0000h 寫 // 理論位置計數器 0
WR7 ← 0000h 寫
WR0 ← 0009h 寫

// 設定計時器
// 返複計時器
WR3 ← 4000h 寫 // D14 1 TMMD : 計時器動作 返複

// 計時器值的設定
WR6 ← 03E8h 寫 // 計時器值 1000 μ sec
WR7 ← 0000h 寫
WR0 ← 0016h 寫

// 設定同步動作
// 設定同步動作 SYNC0
WR6 ← 0153h 寫 // D3~D0 0011 PREV3~0 : 同步動作的起動要因 驅動開始
// D8~D4 10101 ACT4~0 : 同步動作的動作 計時器始動
// D15 0 REP : 同步動作的返複 無效

WR0 ← 0026h 寫

// 設定同步動作 SYNC1
WR6 ← 80A2h 寫 // D3~D0 0010 PREV3~0 : 同步動作的起動要因 時間到
// D8~D4 01010 ACT4~0 : 同步動作的動作 驅動開始
// D15 1 REP : 同步動作的返複 有效

WR0 ← 0027h 寫

// SYNC1~0 有效
WR0 ← 0083h 寫

// 驅動開始
WR0 ← 0050h 寫 // 相對位置驅動開始

:
:
:

// 計時器停止
WR0 ← 0074h 寫 // 計時器停止

// 同步動作 SYNC1 狀態設為無效
WR0 ← 0092h 寫 // 同步動作 SYNC1 狀態設為無效
```

■ 例 3 加減速驅動下、等速驅動 10msec 後、減速停止

加減速驅動開始後、等速開始後起動 10msec 的計時器、時間到後減速驅動停止。使用同步動作機能執行。

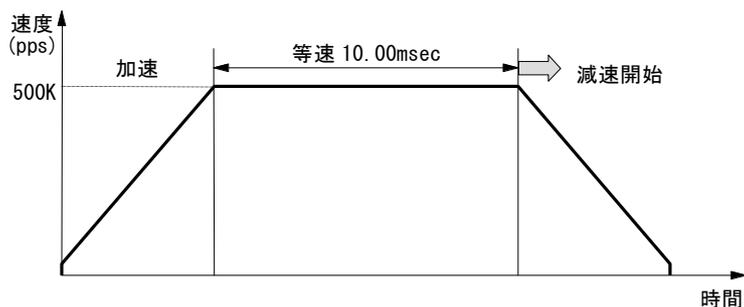


圖 2.10-6 計時器動作 實例 3

【程式例】

```
//設定加減速驅動
WR6 ← 0064h 寫           // 初速度 100 PPS
WR7 ← 0000h 寫
WR0 ← 0004h 寫

WR6 ← A120h 寫           // 驅動速度 500K PPS
WR7 ← 0007h 寫
WR0 ← 0005h 寫

WR6 ← E848h 寫           // 加速度 125K PPS/SEC
WR7 ← 0001h 寫
WR0 ← 0002h 寫

// 設定計時器
// 單一計時器
WR3 ← 0000h 寫           // D14      0 TMMD : 計時器動作 單一

//設定計時器值
WR6 ← 2710h 寫           // 計時器值 10000 μ sec
WR7 ← 0000h 寫
WR0 ← 0016h 寫

// 設定同步動作
//設定同步動作 SYNC0
WR6 ← 0154h 寫           // D3~D0    0100 PREV3~0 : 同步動作的起動要因 驅動等速域開始
WR0 ← 0026h 寫           // D8~D4    10101 ACT4~0 : 同步動作的動作 計時器始動

//設定同步動作 SYNC1
WR6 ← 0112h 寫           // D3~D0    0010 PREV3~0 : 同步動作的起動要因 時間到
WR0 ← 0027h 寫           // D8~D4    10001 ACT4~0 : 同步動作的動作 驅動減速停止

// SYNC1~0 有効
WR0 ← 0083h 寫

// 驅動開始
WR0 ← 0052h 寫           // +方向連續脈波驅動開始
```

2.11 中斷

中斷發生機能為、驅動有關的中斷或、運用多目的暫存器發出的中斷、依同步動作發出的中斷等、可在各式各樣要因下發出中斷。

對CPU的中斷訊號有、INTN訊號1點。

中斷要因全不可以設定其中斷許可／禁止。重置時時全部變為禁止狀態。

2.11.1 X, Y, Z, U軸的中斷

X, Y, Z, U軸口生的中斷要素如下所示。

表 2.11-1 X, Y, Z, U軸口生的中斷要素

許可/禁止的設定 WR1暫存器	發生的確認 RR1暫存器	中斷發生要因
D0 (CMR0)	D0 (CMR0)	多目的暫存器MR0與比較對象的比較結果、滿足比較條件時
D1 (CMR1)	D1 (CMR1)	多目的暫存器MR1與比較對象的比較結果、滿足比較條件時
D2 (CMR2)	D2 (CMR2)	多目的暫存器MR2與比較對象的比較結果、滿足比較條件時
D3 (CMR3)	D3 (CMR3)	多目的暫存器MR3與比較對象的比較結果、滿足比較條件時
D4 (D-STA)	D4 (D-STA)	驅動開始後
D5 (C-STA)	D5 (C-STA)	加減速驅動、等速域的脈波輸出開始後
D6 (C-END)	D6 (C-END)	加減速驅動、等速域的脈波輸出結束後
D7 (D-END)	D7 (D-END)	驅動結束後
D8 (H-END)	D8 (H-END)	自動原點復歸結束後
D9 (計時器)	D9 (計時器)	計時器時間到後
D10 (SPLTP)	D10 (SPLTP)	分割脈波輸出後 (正理論時、分割脈波↑時發生)
D11 (SPLTE)	D11 (SPLTE)	分割脈波結束後
D12 (SYNC0)	D12 (SYNC0)	同步動作SYNC0起動後
D13 (SYNC1)	D13 (SYNC1)	同步動作SYNC1起動後
D14 (SYNC2)	D14 (SYNC2)	同步動作SYNC2起動後
D15 (SYNC3)	D15 (SYNC3)	同步動作SYNC3起動後

■ 中斷的設定及讀取

各種中斷要因如上表所示在WR1暫存器設定許可(1)／禁止(0)。設定許可後中斷要因成真時，RR1暫存器的該要因對應的位元會變為1、中斷輸出訊號(INTN)變為Low位準。上位CPU讀取RR1暫存器時、RR1暫存器中位元為1的會被清除為0、中斷輸出訊號(INTON)回到Hi-Z。因此、RR1暫存器的讀取動作會自動解除中斷訊號。另、通知發生中斷的情報是在、中斷發生後初次讀取RR1暫存器時僅此一次會傳給CPU、此後再次讀取RR1暫存器也只會讓下個中斷要因無法成真而已、該位元成為0(讀取重置方式)。

■ 複數的中斷

容許發生複數中斷要因時、初始中斷要因成真時中斷訊號變為Low、RR1暫存器相對應的位元變為1。此後、CPU讀取RR1暫存器前另外的要因也變成真時、該別的要因所對應的位元變為1。讀取RR1暫存器變成1的複數位元、了解每個發生中斷的要因。

■ 8位元資料匯流排的中斷

8位元資料匯流排、WR1H暫存器、WR1L暫存器設定許可(1)／禁止(0)。中斷發生(中斷訊號Low)後也個別讀取RR1H暫存器及RR1L暫存器。無論如何僅設定許可單方暫存器的時候、另一個暫存器就不需要去讀取。RR1H暫存器讀1次後RR1H內顯示中斷發生的位元會清除為0。RR1L暫存器也一樣。兩暫存器全部位元清除後中斷訊號(INTON)回到Hi-Z。

■ I²C串列通訊匯流排的中斷

I²C串列通訊匯流排的時候、WR1H暫存器、WR1L暫存器可分別設為許可(1)／禁止(0)。也可以在WR1暫存器整合設定許可

(1)／禁止(0)。中斷發生(中斷信號(INT0N)Low)時、讀取RR1暫存器。即使單方暫存器設為許可、也不要只讀取單方暫存器，請務必完整讀取RR1暫存器。RR1暫存器讀取後、RR1暫存器1的地方會全部清除為0、中斷輸出訊號(INT0N)回到Hi-Z。有關I²C串列通訊匯流排的說明請參考第4章。

有關WR1暫存器的說明請參考6.5節、有關RR1暫存器的說明請參考6.12節。

CPU 讀取時序的注意

CPU讀／寫循環的時序如8.2.2項所示。讀循環下RDN訊號在Low位準的區間時務必確認位址訊號A[3:0]。tAR最小=0及tRA最小=3nsec。違反此條件時、RDN訊號在Low位準區間下非有效位址資料穿插進來時、其他暫存器的讀取動作下RR1的資料被清除、中斷訊號有可能被解除。使用中斷訊號時請務必注意CPU的讀取時序。

2.11.2 連續補間的中斷

補間模式設定命令(2Ah)設定許可(1)／禁止(0)。設定許可而發生中斷時、連續補間中斷輸出訊號(INT1N)變為Low位準。

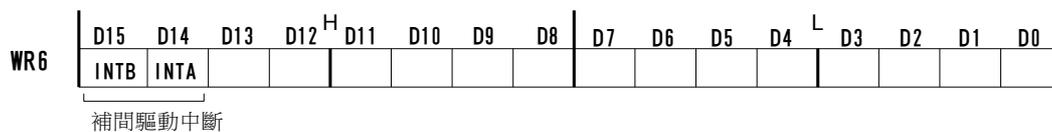


表 2.11-2 連續補間時中斷口生要素

許可/禁止設定 補間模式	中斷發生要素
D14 (INTA)	預存暫存器的狀態計數器 4 → 3 變化
D15 (INTB)	預存暫存器的狀態計數器 8 → 7 變化

補間中斷訊號(INT1N)、在已下條件成立時清除，回復Hi-Z。

- ① 補間中斷清除命令(6Fh)執行後
- ② 下一區段補間命令執行後
- ③ 連續補間驅動結束

二個中斷要素多設為許可時、先發生的中斷要素會把中斷訊號(INT1N)設為Low。此後清除前另一個中斷要素發生時中斷訊號(INT1N)保持Low的狀態，此狀態清除後補間中斷訊號(INT1N)返回Hi-Z。

2.12 輸入訊號濾波

本IC為、於IC內部的各輸入訊號的輸入段備有積分型濾波。圖2.11-1標示各輸入訊號的濾波構造。濾波的時間常數為、依據圖中的T振盪迴路決定。本IC有2個濾波的時間常數A及B、依照輸入訊號的種類決定使用濾波時間常數A或B。濾波的有効／無効及濾波的時間常數是以、輸入訊號濾波模式設定指令(25h)來設定。

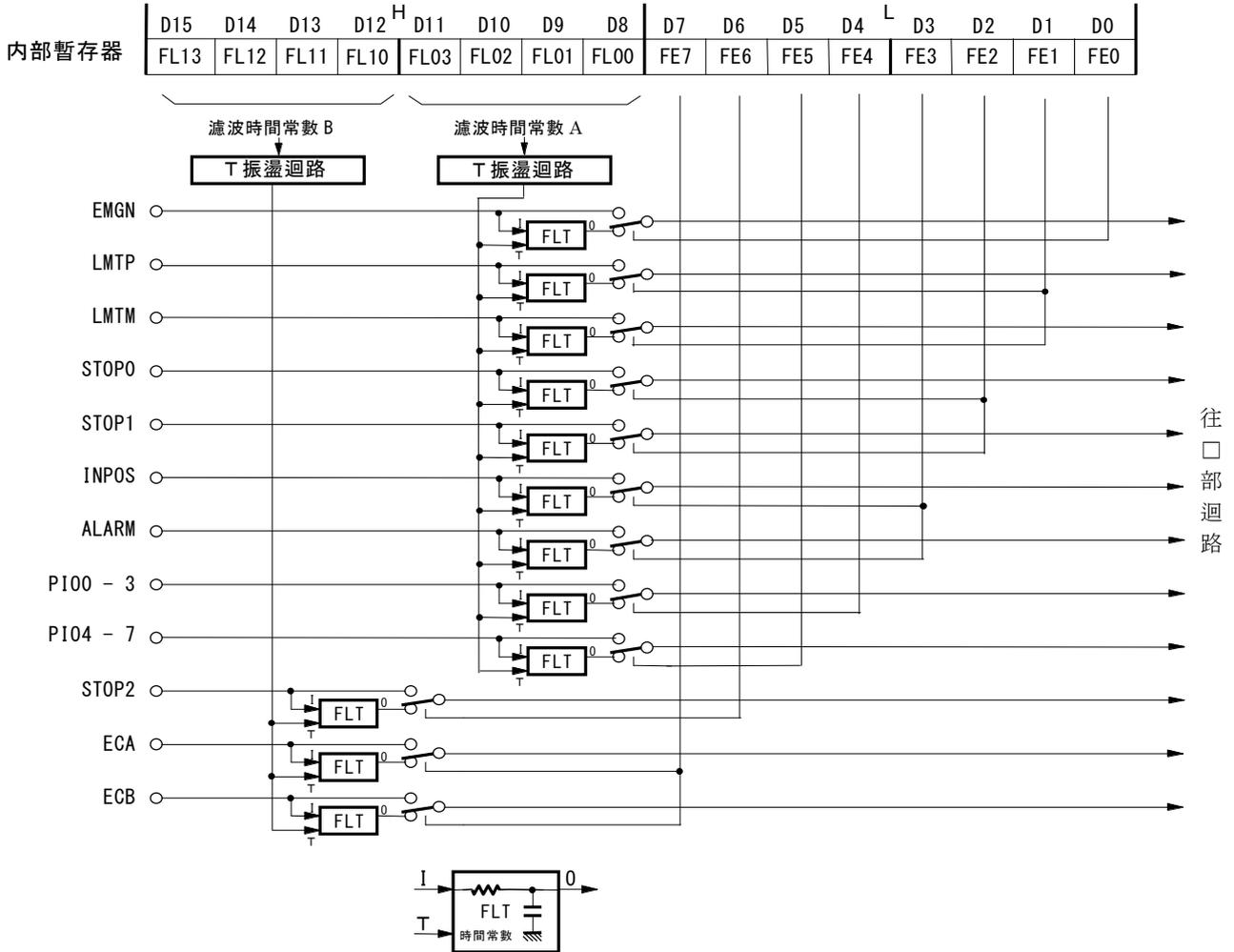
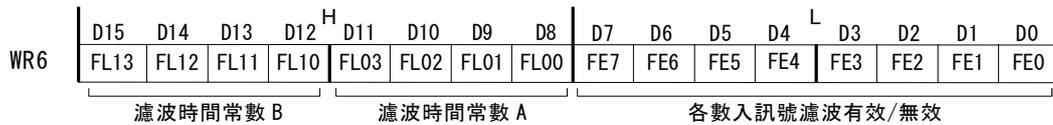


圖 2.12-1 輸入訊號濾波迴路 概念圖

2.12.1 輸入訊號濾波機能的設定

各輸入訊號的濾波機能設定、以輸入訊號濾波模式設定指令(25h)進行。



對於各輸入訊號、IC 內建的濾波機能有效或無效可由 D7~0 位元(FE7~FE0)設定。

濾波機能有效時設定時設定1、無效時設定0。

各位元所對應的輸入訊號如表 2.11-1 所示。決定適用於各輸入訊號的濾波時間常數 A, B。

表 2.12-1 輸入訊號及對應的時間常數

指定位元	輸入訊號	適用的時間常數
D0 (FE0)	EMGN	濾波時間常數 A
D1 (FE1)	LMTp, LMTM	
D2 (FE2)	STOP0, STOP1	
D3 (FE3)	INPOS, ALARM	
D4 (FE4)	PI03~0	
D5 (FE5)	PI07~4	
D6 (FE6)	STOP2	濾波時間常數 B
D7 (FE7)	ECA, ECB	

濾波的時間常數 A 的設定在 D11~8 位元(FL03~FL00)、濾波時間常數 B 的設定在 D15~D12 位元(FL13~FL10)進行。

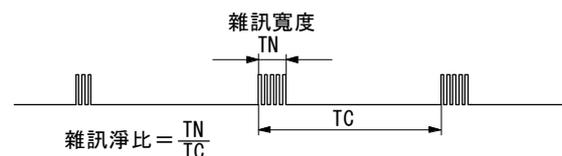
可以設定的時間常數值如表 2.11-2 所示，由 16 段階中選擇。時間常數提高時可以去除的最大雜訊寬會相對提高、但訊號的延遲時間會變大、請設定適切值。通常建議、時間常數 A(FL03~00)設為 Ah 或 Bh。時間常數 B(FL13~10)內定由編碼器輸入訊號使用的時間常數。

表 2.12-2 時間常數及可過濾之最大雜訊寬度

(CLK=16MHz 時)

時間常數 (Hex)	可過濾之最大雜訊寬度 * 1	輸入訊號延遲時間
0	437.5 n sec	500 n sec
1	875 n sec	1 μ sec
2	1.75 μ sec	2 μ sec
3	3.5 μ sec	4 μ sec
4	7 μ sec	8 μ sec
5	14 μ sec	16 μ sec
6	28 μ sec	32 μ sec
7	56 μ sec	64 μ sec
8	112 μ sec	128 μ sec
9	224 μ sec	256 μ sec
A	448 μ sec	512 μ sec
B	896 μ sec	1.024 msec
C	1.792 msec	2.048 msec
D	3.584 msec	4.096 msec
E	7.168 msec	8.192 msec
F	14.336 msec	16.384 msec

* 1 : 雜訊寬度



雜訊淨比為(訊號中雜訊發生時間比率)、無論如何 1/4 以下為所需條件。

重置時、所有的輸入訊號濾波機能全部變為無效。

2.12.2 輸入訊號濾波的設定例

濾波的時間常數 A 所屬的輸入訊號、EMGN 及 LMTP,LMTM,STOP0, STOP1 輸入訊號設定 128 μ sec 延遲的濾波、其他輸入訊號為濾波無効。

濾波的時間常數 B 所屬的 ECA,ECB,STOP2 輸入訊號為濾波無効。

【程式例】

//輸出訊號濾波模式設定

```
WR6 ← 0807h 寫      // D15~D12 0000 濾波時間常數B 濾波延遲:500nsec
                    // D11~D8  1000 濾波時間常數A 濾波延遲:128 $\mu$ sec
                    // D7      0   ECA, ECB訊號(濾波時間常數B) : 濾波無効
                    // D6      0   STOP2訊號(濾波時間常數B) : 濾波無効
                    // D5      0   PI04-7訊號(濾波時間常數A) : 濾波無効
                    // D4      0   PI00-3訊號(濾波時間常數A) : 濾波無効
                    // D3      0   INPOS, ALARM訊號(濾波時間常數A) : 濾波無効
                    // D2      1   STOP0, 1訊號(濾波時間常數A) : 濾波有効
                    // D1      1   LMTP, LMTM訊號(濾波時間常數A) : 濾波有効
                    // D0      1   EMGN訊號(濾波時間常數A) : 濾波有効
```

WR0 ← 0025h 寫

2.13 其他機能

2.13.1 外部訊號操作驅動

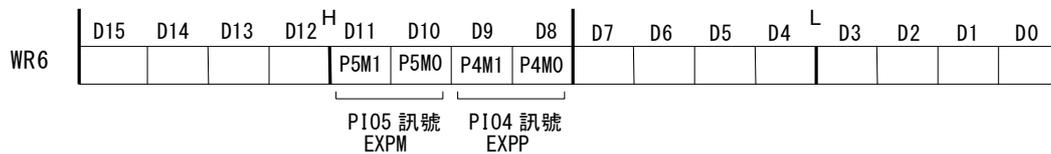
相對位置驅動或連續脈波驅動不使用指令、由訊號輸入 (EXPP, EXPM) 來起動的機能。系統控制的馬達教導進給等手動操作、用本機能來進行時可以減輕CPU的負擔。也可以輸入手動脈波產生器的編碼器2相訊號、進行教導進給。EXPP, EXPM訊號分配在汎用輸出入訊號的PIO4,5訊號。

以外部訊號來操作驅動時、需要做以下的設定。

- ① 用PIO訊號設定1指令(21h)、設定 PIO4,5 為輸入訊號
- ② 用PIO訊號設定2·其他設定指令(22h)、設定驅動操作的模式

■ PIO_n 訊號的外部驅動操作機能設定

以外部訊號進行驅動操作時、汎用輸入訊號PIO4,5訊號設定為外部驅動操作的輸入訊號 (EXPP, EXPM)。設定PIO訊號設定1指令(21h)的 D11~8位元。



PIO4訊號的機能使用於外部驅動操作的輸入訊號 (EXPP) 時、D9,8位元需設為0,0。同樣的、PIO5訊號的D11,10位元也要設為0,0。

■ 驅動操作的模式設定

設定外部驅動的模式。

設定PIO訊號設定2·其他設定指令(22h)的D9,8位元。



外部輸入訊號 (EXPP, EXPM) 之驅動操作模式是設定D9,8位元的2位元。
對應驅動操作模式的各位元的值如下表所示。

表 2.13-1 外部訊號之驅動操作模式

D9 (EXOP1)	D8 (EXOP0)	外部訊號之驅動操作模式
0	0	外部訊號之驅動操作無効
0	1	連續脈波驅動模式
1	0	相對位置驅動模式
1	1	手動脈波產生器模式

■ 相對位置驅動模式

PIO訊號設定2·其他設定指令(22h)的D9,8位元設為1,0、設定驅動所需要的速度參數、移動脈波數(正值)。EXPP訊號由Hi位準往Low位準降下來的時候、其↓起動+方向的相對位置驅動。EXPM訊號的情況也一樣、Hi位準往Low位準降下來的時候、其↓起動-方向的相對位置驅動。各輸入操作訊號的Low位準寬、最小需有4CLK循環以上。在驅動結束前、再次觸發向下訊號也無效。

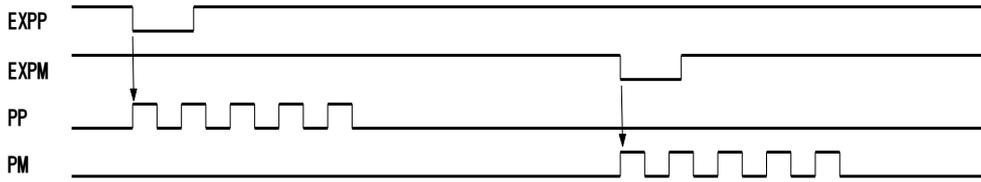


圖 2.13-1 以外部操作訊號驅動 5 個移動脈波的相對位置驅動例

■ 連續脈波驅動模式

PIO訊號設定2·其他設定指令(22h)的D9,8位元數為0,1、設定驅動所需的的速度參數。把EXPP訊號從Hi位準往Low位準降下來的時候、Low位準的期間、輸出連續+方向的驅動脈波。EXPP訊號由Low往Hi位準回復時、加減速驅動的時候會減速停止、等速驅動的時候會立即停止。EXPM訊號的情況也一樣、同樣的、輸出連續-方向的驅動脈波。驅動途中其他的輸入訊號由Hi位準往Low位準降下來的時候、結束現在方向的驅動且立刻開始另一方向的驅動。

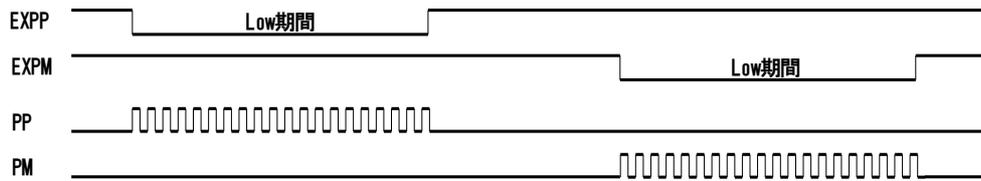


圖 2.13-2 以外部操作訊號控制連續脈波驅動例

■ 手動脈波產生器模式

PIO訊號設定2·其他設定指令(22h)的D9,8位元設為1,1、設定驅動所需的速度參數、移動脈波數。編碼器的A相訊號接在EXPP輸入、B相訊號接在EXPM輸入。EXPM訊號在Low位準的時候EXPP訊號的↑會起動+方向的相對位置驅動。或是、EXPM訊號在Hi位準的時候EXPP訊號的↑會起動-方向的相對位置驅動。移動脈波數的設定若為1時EXPP訊號的↑會送出1個驅動脈波。移動脈波數若設定為TP時會送出TP個驅動脈波。

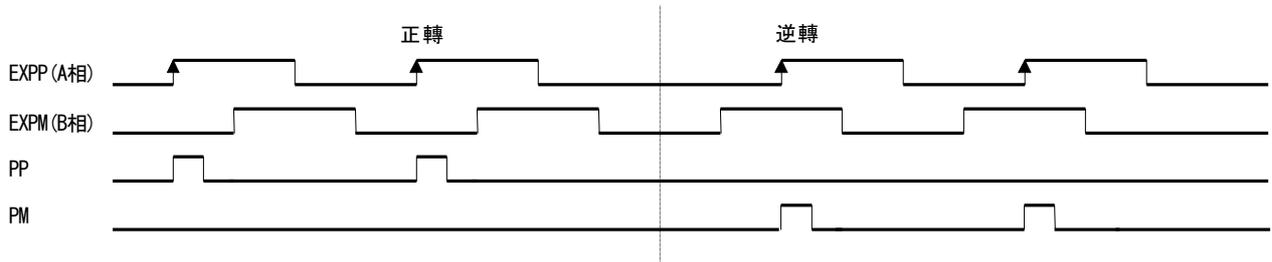


圖 2.13-3 以手輪控制 1 個移動脈波的驅動例

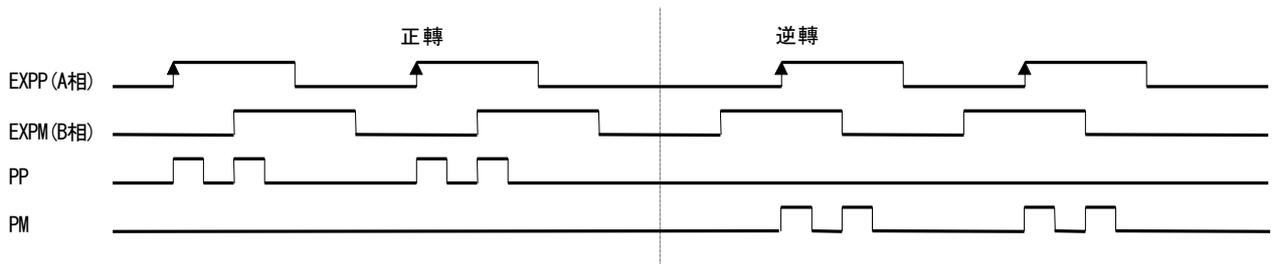


圖 2.13-4 以手輪控制 2 個移動脈波的驅動例

EXPP訊號的↑到下一個↑間需結束TP個驅動脈波的輸出之故、速度參數需如下條件設定。

$$DV \geq F \times TP \times 2$$

DV: 驅動速度(pps)

TP: 移動脈波數

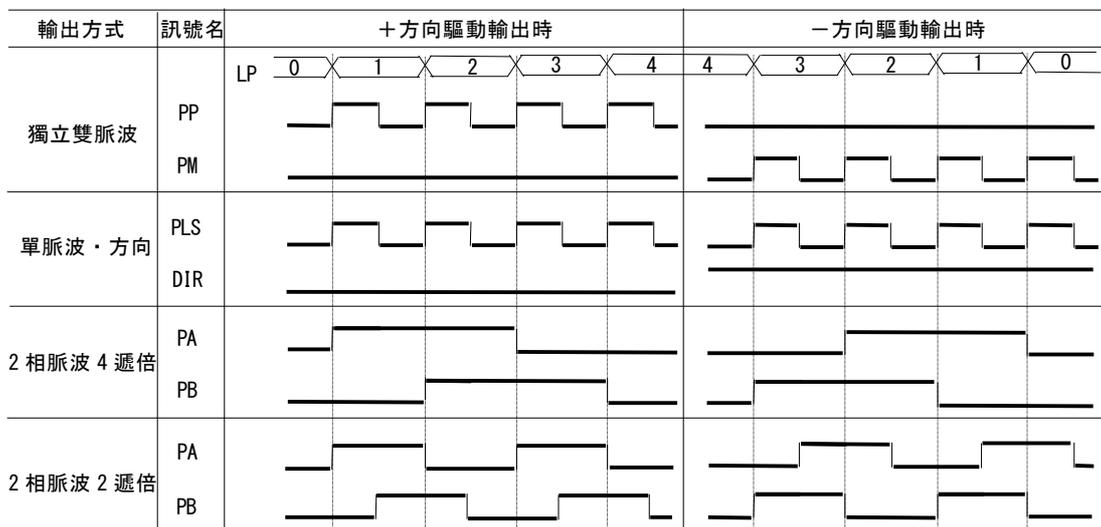
F: 手動脈波產生器編碼器最高速時的周波數(Hz)

例如、手動脈波產生器的最高速周波數F=500Hz、移動脈波數TP=1的時候、驅動速度需設定DV=1000pps以上的值。另外、因不做加減速驅動之故、初速度SV需設定在驅動速度DV以上的值。但式、驅動馬達若為步進馬達的情況下、請在不超過馬達的自起動周波數範圍內設定驅動速度。

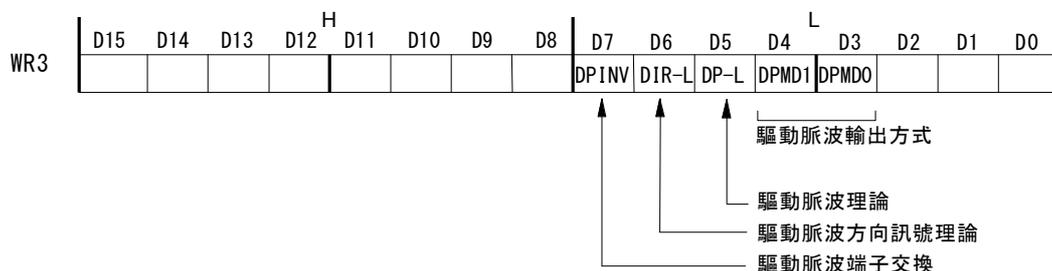
2.13.2 驅動脈波輸出方式的選擇

驅動脈波輸出訊號為PP/PLS/PB(35)及PM/DIR/PB(36)。下表所示有4個脈波輸出方式可供選擇。獨立2脈波方式是在、+方向驅動時PP、-方向驅動時PM輸出驅動脈波。或是、1脈波·方向方式是在、PLS輸出驅動脈波、DIR輸出方向訊號。選擇2相脈波方式的時候，輸出訊號PA為2相脈波的A相訊號、輸出訊號PB為2相脈波的B相訊號。2相4遞倍式在、PA、PB脈波的輸出有變化發生時理論位置計數器遞增(遞減)。2相2遞倍式在、PA脈波的輸出有變化時理論位置計數器遞增(遞減)。

表 2.13-2 驅動脈波輸出方式



驅動脈波輸出方式、設定WR3暫存器的D4,3位元(DPMD1,0)。



各位元值所對應的驅動操作模式如下表。

表 2.13-3 驅動脈波輸出方式

D4 (DPMD1)	D3 (DPMD0)	驅動脈波輸出方式
0	0	獨立2脈波方式
0	1	1脈波·方向方式
1	0	2相脈波4遞倍方式
1	1	2相脈波2遞倍方式

1脈波·方向方式的情況式、脈波訊號PLS及方向訊號DIR的輸出時序請由11.2節確認。在驅動前想要設定DIR訊號的情況是、發出方向訊號+設定指令(58h)或方向訊號-設定指令(59h)。

或是、可以設定驅動脈波訊號·D5位元(DP-L)為驅動脈波的脈波理論、D6位元(DIR-L)為驅動方向訊號(DIR)理論、D7位元(DPINV)為端子替換。

2.13.3 編碼器輸入脈波方式的選擇

實際位置計數器的遞增/遞減計數輸入訊號若為編碼器脈波輸入(ECA/PPIN、ECB/PMIN)時、可選擇2相脈波輸入及上下脈波輸入。

或、可以設定 D10 位元(PI-L)為編碼器輸入訊號的理論、D11 位元(PIINV)為編碼器輸入訊號的端子替換。編碼器輸入訊號的端子替換時實際位置計數器的增減如下所示。

表 2.13-5 編碼器輸入訊號的端子替換時實際位置計數器的增減

WR3/D11 (PIINV)	輸入脈波模式	實際位置計數器的增減
0	2 相模式	A 相行進時計數遞增。 B 相行進時計數遞減。
	上下脈波模式	PPIN 脈波輸入時計數遞增。 PMIN 脈波輸入時計數遞減。
1	2 相模式	B 相行進時計數遞增。 A 相行進時計數遞減。
	上下脈波模式	PMIN脈波輸入時計數遞增。 PPIN 脈波輸入時計數遞減。

2.13.4 硬體極限訊號

硬體極限訊號(LMTP,LMTM)為、抑止+方向、一方向驅動脈波的訊號輸入。

可以選擇極限訊號的有効／無効、極限訊號的理論位準、極限訊號動作時減速停止或立即停止、以及極限訊號的端子替換。

極限訊號的有効／無効、理論位準、停止方式是經由WR2暫存器的D12～10位元設定。有關WR2暫存器的設定說明請參考4.5節。

極限訊號端子替換的選擇是經由 WR3 暫存器的 D12 位元(LMINV)設定。有關 WR3 暫存器的設定說明請參考 4.6 節。

極限訊號的狀態隨時可由RR3暫存器讀取。

2.13.5 伺服馬達驅動器對應的訊號

■ INPOS 訊號、ALARM 訊號

具備伺服馬達驅動器接線的輸入訊號、如輸入定位完成訊號的INPOS、及輸入伺服異常訊號的ALARM。

各訊號有効／無効及理論位準都可以定。設定時可藉由WR2暫存器的D9～6位元來完成。有關WR2暫存器的設定請參考4.5節說明。

INPOS輸入訊號是、對應伺服馬達驅動器的定位完成訊號。設為有効後、驅動結束後、等到INPOS輸入訊號動作時、RR0主狀態暫存器的D0位元(驅動狀態)會變為0。

ALARM輸入訊號是、接收伺服馬達驅動器來的異常訊號。設為有効後、驅動中會一直監控ALARM輸入訊號、若變為動作狀態時驅動會立即停止。此時RR2暫存器的D4(ALARM)位元及D14(ALARM)位元位變為1。

這些伺服馬達驅動器用輸入訊號可隨時由RR3暫存器讀取其狀態。

■ 偏差計數器清除輸出訊號

具備伺服馬達驅動器用輸出訊號的偏差計數器清除輸出訊號(DCC)。

偏差計數器清除輸出訊號(DCC)的理論、及脈波寬都可以設定。設定時藉由自動原點復歸模式設定2指令(24h)的D3～6位元來完成。有關自動原點復歸模式設定2指令(24h)的說明請參考5.3.5項說明。

發出偏差計數器清除輸出指令(72h)時、於自動原點復歸模式設定2指令(24h)設定的、脈波理論位準、脈波寬就會經由、DCC輸出端子輸出偏差計數器清除脈波。

自動原點復歸中使用偏差計數器清除輸出訊號(DCC)的情況、請參考2.5.2項、2.5.4項的說明。

2.13.6 緊急停止

本IC具備、緊急停止驅動所需的輸入訊號EMGN。EMGN訊號通常為Hi位準。當它變為Low位準時、驅動會立即停止、RR2暫存器的D5(EMG)位元及D15(EMG)位元變為1。請注意EMGN訊號的理論位準是不可以選擇的。

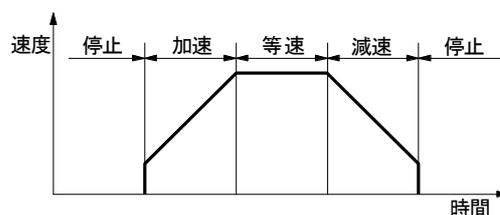
EMGN訊號是可隨時由RR3暫存器讀取其狀態。

由CPU側發起緊急停止、有以下的方法。

- a. 發出立即停止指令
WR0暫存器寫入立即停止指令(57h)。
- b. 發出重置指令
WR0暫存器寫入00FFh後重置。

2.13.7 驅動狀態的輸出

驅動中／停止的狀態由、RR0暫存器D0(DRIVE)位元及PIO0訊號輸出。
驅動中驅動速度加速／等速／減速的狀態由RR0暫存器的D2(ASND), D3(CNST), D4(DSND)位元及PIO2/ASND, PIO3/CNST, PIO4/DSND訊號輸出。



圖

2.13-7 驅動狀態

表 2.13-6 對應驅動狀態的 RR0 暫存器及 PIO_n 訊號

驅動狀態	主狀態暫存器 (RR0)				PIO _n 訊號			
	D0/DRIVE	D2/ASND	D3/CNST	D4/DSND	PIO0/DRIVE	PIO2/ASND	PIO3/CNST	PIO4/DSND
停止	0	0	0	0	Low	Low	Low	Low
加速	1	1	0	0	Hi	Hi	Low	Low
等速	1	0	1	0	Hi	Low	Hi	Low
減速	1	0	0	1	Hi	Low	Low	Hi

另、S形加減速驅動的加速度、減速度的增加／一定／減少的狀態也是由RR0暫存器的D5(AASND), D6(ACNST), D7(ADSND)位元及PIO5/AASND, PIO6/ACNST, PIO7/ADSND訊號來輸出。

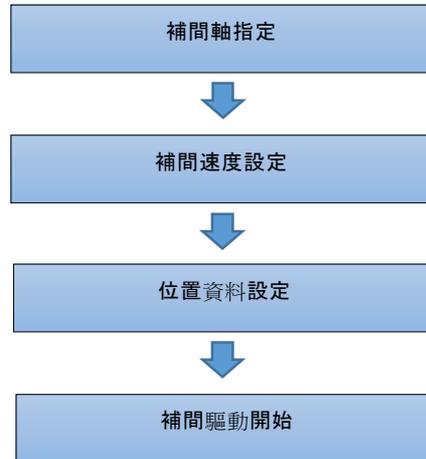
PIO_n訊號把驅動狀態輸出的時候、以PIO訊號設定1指令(21h)來進行。請參考7.3.2項說明。

3. 補間

補間驅動為 2 軸以上的軸各軸在每一驅動脈衝位置間做插補移動的動作。

本IC可任選4軸中的軸做直線補間、圆弧補間、螺旋補間、位元補間驅動。或是複數本IC使用下也可做5軸以上的多軸直線補間。

補間動作的進行基本的操作順序如下所示。



■ 補間軸設定

指定進行補間的軸。執行補間模式設定命令(2Ah)。
有關補間模式設定命令(2Ah)請參考7. 3. 8項說明。

【注意】

- 補間軸指定(補間模式設定命令(2Ah)執行)、請務必在補間動作設定的最初執行。若此命令在補間速度或位置資料設定後進行時、補間驅動會不確實。

■ 補間速度設定

補間驅動速度設定。速度設定在指定軸中以X>Y>Z>U的優先順位自動決定的主軸設定。例如、指定補間執行軸X, Z, U 3軸時、主軸為X軸。初速度、驅動速度...等速度參數設定於主軸。主軸是、補間驅動開始後補間演算部對主軸輸出脈衝。補間演算部則依主軸脈衝的時序執行演算循環、執行補間產生各軸的驅動脈衝。請參考圖1. 2-1 MCX514全体機能方塊圖。主軸脈衝僅止於補間演算部使用、並非選取主軸來設定驅動脈衝速度。

補間演算在、直線補間、圆弧補間、位元補間最高可達8MPPS。但是連續補間時最高4MPPS為止。螺旋補間最高2MPPS。另、螺旋補間無法在連續補間中執行。

補間速度請一定要設定。特別是以下的情況、也務必要設定。

- 一般驅動後執行補間驅動時、速度參數與一般驅動相同時
- 補間驅動後、速度及位置參數相同僅改變補間模式設定的補間驅動

【注意】

- 請不要將驅動速度設為 0 做補間驅動。

■ 位置資料設定

設定2·3·4軸直線補間下各軸的終點位置、圆弧補間下圆弧中心位置及終點位置。2·3·4軸位元補間下各軸+方向/-方向

位元資料。位元補間下補間驅動開始前各軸128個位元資料。螺旋補間下圆弧中心位置及終點位置、還有Z(U)方向移動量。

【注意】

- 位置資料一樣繼續執行補間驅動時、位置資料需重新設定。

■ 補間驅動開始

補間所要的速度・位置參數設定後、執行補間驅動命令後補間驅動開始。位元補間下、補間驅動中隨時補充位元資料時可以無限的描繪連續任意的驅動軌跡。

■ 補間驅動狀態

補間驅動中、RR0 (主狀態暫存器)在補間進行時所有軸的n-DRV位元變為1。補間驅動結束後該位元回到0。

■ 補間驅動時錯誤發生

補間驅動進行中驅動的各軸硬體或軟體極限動作中、任一軸的極限動作時、補間驅動停止。錯誤發生停止時、RR0(主狀態暫存器)的補間指定軸錯誤位元設為1。且讀取該軸的RR2(錯誤暫存器)後可以了解發生錯誤的原因。

【注意】

- 圆弧補間、螺旋補間、及位元補間下、+方向／-方向各方向的硬體極限或軟體動作時也會使補間驅動停止。所以無法用圆弧補間、螺旋補間、及位元補間方式脫離領域。請单独驅動該軸脫離。

■ 伺服馬達定位完成訊號對應

補間驅動時、補間驅動的各軸定位完成訊號(nINPOS)有效設定時、補間驅動結束後、等到所有軸的nINPOS訊號全部動作後、RR0(主狀態暫存器)補間進行的全部軸驅動位元返回0。

■ 補間軸的設定

以補間模式設定命令(2Ah)指定補間軸後、請勿再對該軸做一般的驅動。

例如有個軸在通常驅動及補間驅動反覆執行、通常驅動前、WR6暫存器設為0後、請執行補間模式設定命令(2Ah)將補間軸的設定情報清除。而做補間驅動前、請執行補間模式設定命令(2Ah)把補間軸及補間模式做適切的設定。

■ 依同步動作狀況停止補間驅動

以同步動作來停止補間驅動時、請務必對補間軸執行錯誤・結束狀態清除命令(79h)。以同步動作確認驅動結束時、可透過RR2暫存器的D8位元進行。

有關同步動作的說明請參考2.7節、RR2暫存器相關說明請參考6.13節。

3.1 直線補間

任意選擇2軸、3軸或全4軸、進行直線補間驅動。直線補間為、相對於現在座標所設定的終點座標、支援補間執行的軸數在寫入直線補間命令後開始執行。

終點座標是以現在位置起的相對值寫入對應軸的輸出脈衝數。

圖3.1-1為2軸補間例、由現在座標往終點座標做直線補間。如圖所示理想直線的相對誤差演算精度可控制在全補間 ± 0.5 LSB範圍內。

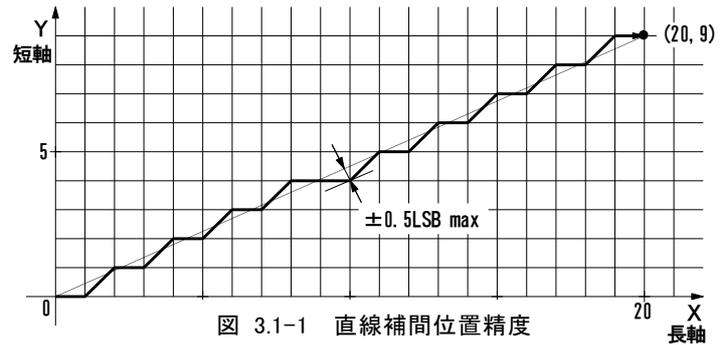


圖 3.1-1 直線補間位置精度

右圖3.1-2為直線補間驅動脈衝輸出例。以設定的終點值中絕對值最大的軸為長軸、補間驅動進行中、隨時保持有脈衝輸出(長軸的終點值為終點最大值稱之)。其他剩下的軸為短軸、依據直線補間演算結果、脈衝輸出會時有時無的情況。線速一定模式設為無効時、長軸驅動脈衝的速度為已設定的主軸驅動速度。

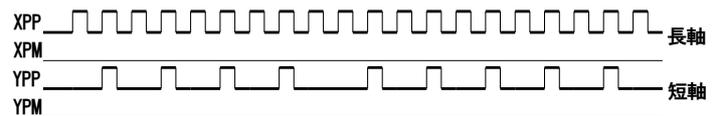


圖 3.1-2 終點(X:20,Y:9)的驅動脈衝輸出例

直線補間的座標範圍為帶符號的32位元長。各軸均從現在位置起 $-2,147,483,646 \sim +2,147,483,646$ (帶符號32bit-2LSB)的範圍內做補間。

3.1.1 終點最大值

IC重置後初期狀態的終點最大值为自動算出、也可以用補間模式設定命令(2Ah)做人工設定。人工設定時、可設定任意值為終點最大值使用。有關補間模式設定命令(2Ah)說明請參考7.3.8項

■ 終點最大值的設定及清除

以補間模式設定命令(2Ah)指定補間軸、讓直線補間最大值自動設定時、對補間軸指定軸寫入移動脈衝數/終點的命令(06h)寫入後本IC即自動設定終點最大值。因此有可能設定意外的終點最大值。

且一旦終點最大值設定後就無法回復。所以

- ① 補間下使用直線補間時、補間模式設定後請執行終點最大值清除命令(7Ch)
- ② 終點輸入錯誤、重新輸入修正時、請用終點最大值讀取命令(39h)做終點最大值確認。

①不一定每次多要做、但在直線補間前做其他動作時做一次會較安全。

②讀取值若不是想要的時候、請以終點最大值清除命令(7Ch)將終點最大值清除後、請再一次輸入全軸的終點。但是、終點最大值讀取命令(39h)所讀取的值、請注意在補間驅動執行前與執行中內容會有差異。終點最大值讀取命令(39h)的詳細說明請參考、7.4.10項。

■ 人工設定多晶片補間的終點最大值

多晶片補間人工設定終點最大值時、請針對全部晶片做以下設定。

表 3.1-1 人工設定多晶片補間時終點最大值

設定項目	設定詳細
人工設定補間終點	以補間模式設定命令(2Ah)對WR6/D12位元設為1。
寫入終點最大值	執行直線補間驅動命令前、以終點最大值寫入命令(19h)把值寫入。

3.1.2 直線補間實例

■ 2軸直線補間驅動的例

X、Y 軸從現在位置起到終點座標(X:+300,Y:-200)進行直線補間。補間驅動速度為1000PPS 的等速驅動。

```

WR6 ← 0003h 寫入      ; 補間軸X,Yを指定
WR0 ← 002Ah 寫入

WR6 ← 03E8h 寫入      ; 初速度 : 1000 PPS
WR7 ← 0000h 寫入
WR0 ← 0104h 寫入

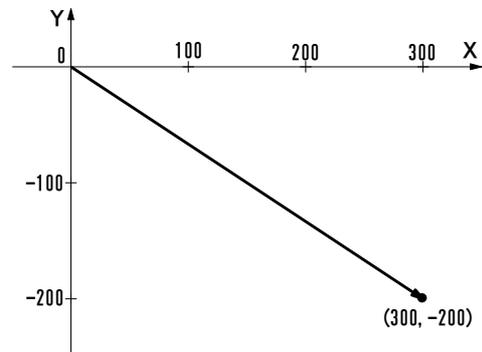
WR6 ← 03E8h 寫入      ; 驅動速度 : 1000 PPS
WR7 ← 0000h 寫入
WR0 ← 0105h 寫入

WR6 ← 012Ch 寫入      ; 終点 X 軸 : 300
WR7 ← 0000h 寫入
WR0 ← 0106h 寫入

WR6 ← FF38h 寫入      ; 終点 Y 軸 : -200
WR7 ← FFFFh 寫入
WR0 ← 0206h 寫入

WR0 ← 0061h 寫入      ; 2 軸直線補間驅動

```



■ 3軸直線補間驅動的例

X、Y、Z 軸從現在位置起到終點座標(X:15000,Y:16000,Z:20000)進行三軸直線補間。補間驅動速度為初速度:500PPS、加減速度:40,000PPS/SEC、驅動速度: 5,000PPS 的梯形加減速驅動。

```

WR6 ← 0007h 寫入      ; 指定補間軸X,Y,Z
WR0 ← 002Ah 寫入

WR6 ← 9C40h 寫入      ; 40,000 PPS/SEC
WR7 ← 0000h 寫入
WR0 ← 0102h 寫入      ; 設定主軸加速度

WR6 ← 01F4h 寫入      ; 500 PPS
WR7 ← 0000h 寫入
WR0 ← 0104h 寫入      ; 設定主軸初速度

WR6 ← 1388h 寫入      ; 5000 PPS
WR7 ← 0000h 寫入
WR0 ← 0105h 寫入      ; 設定主軸驅動速度

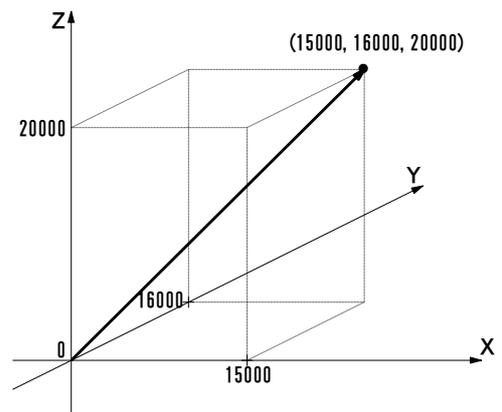
WR6 ← 3A98h 寫入      ; 終点 X : 15,000
WR7 ← 0000h 寫入
WR0 ← 0106h 寫入

WR6 ← 3E80h 寫入      ; 終点 Y : 16,000
WR7 ← 0000h 寫入
WR0 ← 0206h 寫入

WR6 ← 4E20h 寫入      ; 終点 Z : 20,000
WR7 ← 0000h 寫入
WR0 ← 0406h 寫入

WR0 ← 006Dh 寫入      ; 減速有効
WR0 ← 0062h 寫入      ; 3 軸直線補間驅動

```



3.2 圆弧補間

4軸中、可以選擇任意2軸做圆弧補間驅動。

如右圖所示直交座標系、選擇的2軸以、 $X > Y > Z > U$ 的先後順序，優先度高的軸為ax1軸(橫軸)、低的軸為ax2軸(縱軸)。ax1軸(橫軸)的右方向為+方向、ax2軸(縱軸)的上方向為+方向。X軸及Y軸選擇確定後、X軸為ax1軸(橫軸)、Y軸為ax2軸(縱軸)。也可以用補間模式設定方式替換軸。

圆弧補間是以現在座標(始点)作為起點，等圆弧中心座標、及終点座標設定後、寫入CW圆弧補間命令、或是CCW圆弧補間命令後開始執行。中心座標、及終点座標的指定方式是以現在座標(始点)為基準設定其相對值。CW圆弧補間是由現在座標起往終点座標方向，以中心座標為中心做順時針方向、或、CCW圆弧補間為逆時針方向做圆弧。終点如果設為(0, 0)時、可以畫出真円。

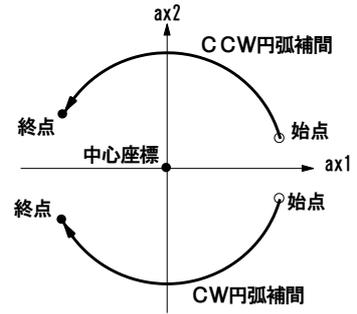


圖 3.2-1 CW/CCW圆弧補間

本IC內部圆弧補間的演算如圖3.2-2所示、在第1軸(ax1)及第2軸(ax2)所成的平面上、以中心座標為中心、分成0~7共8個象限。如圖所示、0象限上圆弧移動時補間座標(ax1, ax2)、一直是ax2的絕對值小於ax1的絕對值。絕對值小的一方做為短軸、因此在1、2、5、6象限中第1軸(ax1)為短軸、0、3、4、7象限中第2軸(ax2)為短軸。短軸在該象限內驅動脈衝持續輸出、長軸則依據圆弧補間演算結果有時不會輸出脈衝。

圖3.2-3、為以現在座標開始，同時指定中心(-11, 0)、終点(0, 0)後描繪半径11的全真円參考例。另外、圖3.2-4為此時輸出的驅動脈衝。

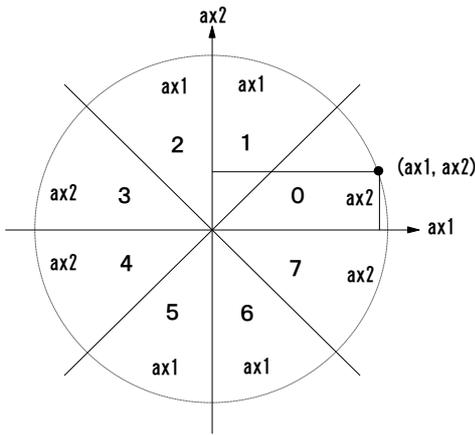


圖 3.2-2 圆弧補間演算的0~7象限及短軸

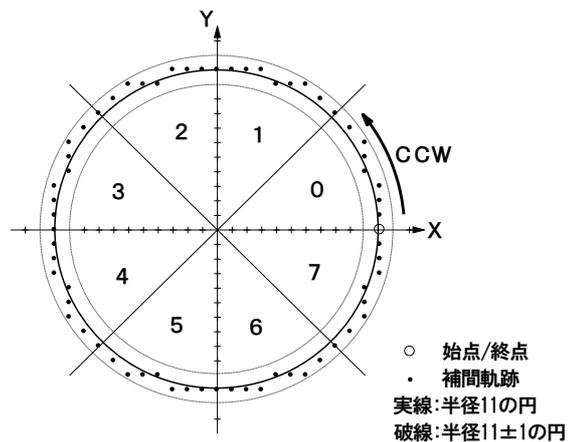


圖 3.2-3 圆弧補間例

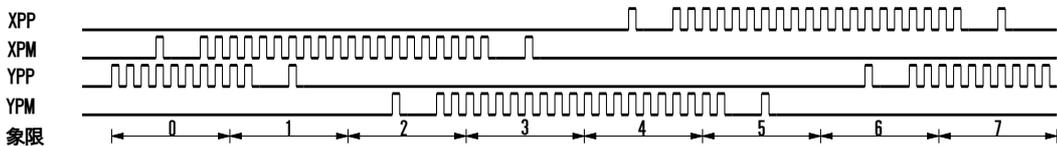


圖 3.2-4 圆弧補間驅動脈衝輸出例

中心座標及終点座標的指定範圍、現在位置起-2, 147, 483, 646~+2, 147, 483, 646(帶符號32bit - 2LSB)。相對於指定圆弧曲線的位置誤差在全補間範圍內±1 LSB。補間速度1PPS~8MPPS。

3.2.1 終點判定

圆弧補間在補間驅動開始前現在座標作為(0, 0)時、依中心座標值可決定半徑大小、及描繪的圆弧軌跡。圆弧演算誤差在補間座標範圍內、±1LSB之故、指定的終點不一定要在圆弧軌跡上。因此本IC是以終點所在象限下當終點與短軸值相等或是超過時作為圆弧補間結束。終點所在象限下即使終點不能達到短軸值時在該象限結束的地方圆弧補間結束。

圖3.2-5是、現在位置(0, 0)起中心(-200, 500)、終點(-702, 299)的情況下做CCW圆弧補間的例。現在位置(0, 0)與中心(-200, 500)所決定的半徑做CCW方向補間。指定的終點(-702, 299)、與中心的位置關係可以看出在第4象限。補間再進入第4象限後、第2軸(ax2)變為短軸、因此第2軸的值到達終點(-702, 299)的299時判定補間結束。

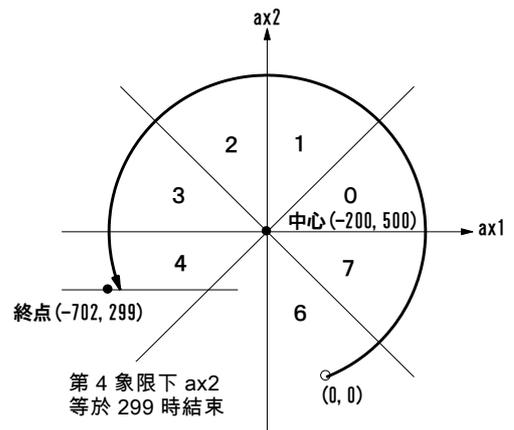


圖 3.2-5 圆弧補間終了判定例

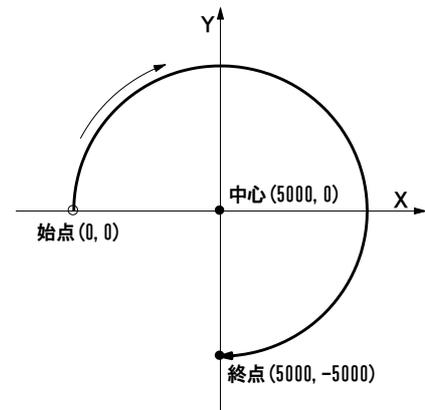
3.2.2 短軸脈衝均一化使用時注意

使用短軸脈衝均一化時、圆弧的始點及終點若不在X軸、Y軸的軸上時、終點位置在兩軸多有±1脈衝的偏差。圆弧始點及終點都不在X軸、Y軸的軸上且要使用短軸脈衝均一化的機能時、請務必慎重檢討此問題可否接受。圆弧始點及終點若在X軸或Y軸的軸上時、此問題不會發生。有關短軸脈衝均一化說明請參考3.6節。

3.2.3 CW圆弧補間驅動例

對X、Y軸、從現在位置(始點)開始、中心(X:5000, Y:0)、終點(X:5000, Y:-5000)做CW圆弧補間。補間驅動速度為1000PPS等速驅動、2軸簡易線速一定模式補間。

WR6 ← 0043h 寫入	; ax1:X軸、ax2:Y軸指定、2軸簡易線速一定
WR0 ← 002Ah 寫入	
WR6 ← 03E8h 寫入	; 1000 PPS
WR7 ← 0000h 寫入	
WR0 ← 0104h 寫入	; 設定主軸初速度
WR6 ← 03E8h 寫入	; 1000 PPS
WR7 ← 0000h 寫入	
WR0 ← 0105h 寫入	; 設定主軸驅動速度
WR6 ← 1388h 寫入	; 中心 X : 5000
WR7 ← 0000h 寫入	
WR0 ← 0108h 寫入	
WR6 ← 0000h 寫入	; 中心 Y : 0
WR7 ← 0000h 寫入	
WR0 ← 0208h 寫入	
WR6 ← 1388h 寫入	; 終點 X : 5000
WR7 ← 0000h 寫入	
WR0 ← 0106h 寫入	
WR6 ← EC78h 寫入	; 終點 Y : -5000
WR7 ← FFFFh 寫入	
WR0 ← 0206h 寫入	
WR0 ← 0064h 寫入	; CW圆弧補間驅動



3.3 螺旋補間

螺旋補間是、XY平面(直交座標)做圆弧補間驅動的時候其他軸同步移動的動作。如下圖配合XY平面圆弧補間的同時、Z軸做+方向移動的例。圖 3.3-1 所示為圆弧補間 1 轉以內的螺旋補間、圖 3.3-2 所示為複數回轉例。本 IC 可做這兩種補間動作。

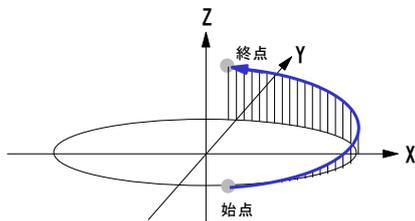


圖 3.3-1 螺旋補間例 (1 轉以內)

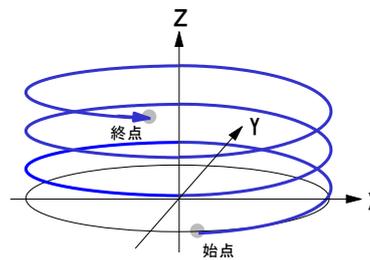


圖 3.3-2 螺旋補間例 (1 轉以上)

另、螺旋補間搭配XY平面上的圆弧補間、其他的回轉軸做一定回轉角的回轉、可以做圆弧上的法線動作。圖 3.3-3 所示為XY平面上做圆弧補間時設置一回轉軸的台座、台座上的攝影機或噴嘴可以隨時保持面向圓心動作。

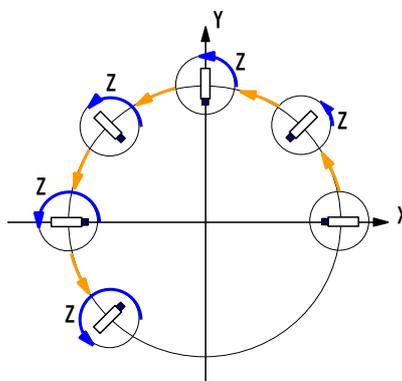


圖 3.3-3 X Y 軸圆弧補間及 Z 軸的法線控制例

本 IC 做螺旋補間的順序說明如下。這裡說明的是圆弧補間搭配 Z 軸移動的螺旋補間。本 IC 為求、Z 軸移動均一的關係會使用圆弧補間的總輸出脈衝數及 Z 軸的移動脈衝數。Z 軸的移動脈衝數可以完全確定、但 XY 平面上動作的圆弧補間總輸出脈衝數僅使用一開始的圆弧中心點、終點值來做精確的計算非常困難。因此本 IC 在做螺旋補間前、會算出圆弧補間總輸出脈衝數來求取螺旋補間演算。實際上做螺旋補間時會有下的操作順序。

表 3.3-1 螺旋補間的操作順序

項次	操作項目	內容
1	設定補間軸	指定螺旋補間軸。
2	設定補間速度資料	設定圆弧補間速度。
3	設定螺旋補間回轉數	複數回轉時設定回轉數。
4	位置資料設定	設定圆弧補間中心點，終點、Z、U 軸的進給量。
5	執行螺旋補間演算	計算圆弧補間的總輸出脈衝數。
6	執行螺旋補間	執行螺旋補間。

#1(設定補間軸)請務必一開始就做此動作。接這進行#2~#4(#2~#4 設定順序可以改變)、然後進行#5(執行螺旋補間演算)。最後#6(執行螺旋補間)。未依此順序執行、或是省略操作項目時補間動作會不正常。

連續做完全一樣的螺旋補間時，#1 及#5 的設定或執行可以省略、但其他項目必須再設定或執行，不可以省略。

3.3.1 補間軸設定

螺旋補間時、円弧補間軸固定為X及Y軸。不可以用其他軸做円弧補間。而搭配円弧補間移動軸可以選擇指定Z軸或是U軸。Z軸及U軸僅其中1軸移動、或兩軸移動(比如回轉)都可以。例如攝影機及噴嘴，刃物等，Z軸在円弧補間平面與垂直方向做螺旋補間、U軸的台座回轉(U軸)控制鏡頭做法線控制移動。

補間軸的設定、以補間模式設定命令(2Ah)來執行。設定以下WR6暫存器的D0到D3位元。針對補間軸對應的位元設為1。X軸與Y軸位元請務必設為1、Z或U軸可依需求設定。

WR6	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
													U-EN	Z-EN	Y-EN	X-EN

D3 U-EN	D2 Z-EN	D1 Y-EN	D0 X-EN	軸動作
0	1	1	1	配合X, Y軸円弧補間Z軸移動。
1	0	1	1	配合X, Y軸円弧補間U軸移動。
1	1	1	1	配合X, Y軸円弧補間Z, U軸移動。

WR6暫存器的其他位元(D15~D4)為補間相關機能設定位元。請參考7.3.8項說明依需求做適當值設定。

3.3.2 補間速度資料設定

螺旋補間因為主軸為X軸、所以速度設定在X軸。通常螺旋補間是等速(不加減速)進行、所以X軸只要設定初速度及驅動速度。設定後的速度在XY平面的円弧補間動作。配合円弧補間移動(回轉)的Z或U軸的速度會配合円弧補間速度自動決定軸進給量、不需設定。

因補間速度一定的關係、可以使用“短軸脈衝均一化模式”及“線速一定模式”。有關之詳細說明請參考、3.6節及3.5節。

3.3.3 螺旋補間回轉數設定

螺旋超過1周以上時需要設定回轉數。不滿1周時回轉數設定0。請將0 ~ 65,535範圍的回轉數寫入WR6暫存器、螺旋回轉數設定命令(1Ah)寫入後回轉數就被設定。此寫入命令不需要指定軸。

■ 真円螺旋補間時的回轉數設定

始点X, Y軸一起設為0的時候就會是真円。此時、螺旋回轉數不管設為0或1也會轉1圈。設定2以上時僅會轉設定圈數。

3.3.4 位置資料設定

設定XY平面上移動的圆弧補間中心(X值, Y值)與終點後。如果Z軸或U軸有配合的動作時再設定其進給量。

表 3.3-2 螺旋補間位置資料的設定

設定資料	設定內容
圆弧補間中心	中心點 X 軸值, Y 軸值、相對於現在位置(螺旋補間開始前的位置)的脈衝數設定。設定值寫入 WR6,7 暫存器、軸指定連同圆弧中心寫入命令(08h)寫入 WR0 暫存器。
圆弧補間終點	終點 X 軸值, Y 軸值、相對於現在位置脈衝數設定。設定值寫入 WR6,7 暫存器、軸指定連同移動脈衝數/終點寫入命令(06h)寫入 WR0 暫存器。
Z U 軸進給量	<ul style="list-style-type: none"> · 搭配圆弧補間進給軸的進給量以現在位置的相對脈衝值設定。+方向進給時正值、-方向進給場合負值設定。設定值寫入 WR6,7 暫存器、軸指定連同移動脈衝數/終點寫入命令(06h)寫入 WR0 暫存器。 · 圆弧補間未滿 1 轉時設定終點為止的進給量(參考圖 3.3-4 (a))。圆弧補間 1 轉以上的時候、設定圆弧補間 1 轉的進給量(圖 3.3-4 (b)參照)。 · 設定 Z 軸或 U 軸的進給量時、一定要設定圆弧補間的總輸出脈衝數(螺旋演算式的螺旋演算結果值),設定值一定要小於總輸出脈衝數。目標值條件是設定值為小於圆弧補間圆弧長的值。

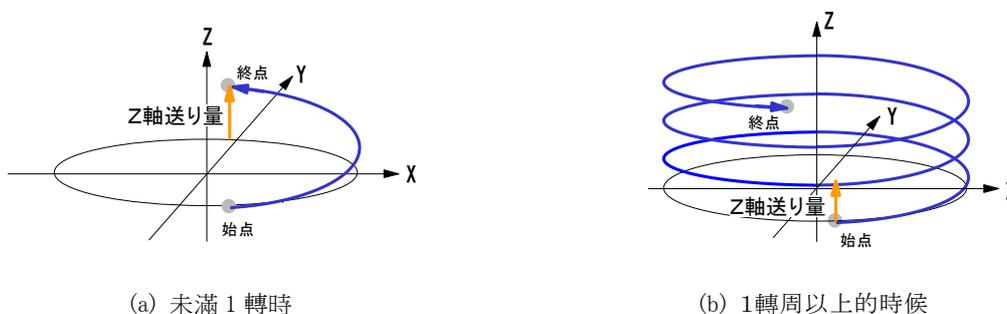


圖 3.3-4 Z、U 軸進給量設定

圆弧補間中心點、終點的設定方法和通常圆弧補間的做法相同。剛好 1 轉的時候終點設為(0,0)。複數轉、回轉的始點位置為結束位置時,終點設為(0,0)。

3.3.5 螺旋演算執行

本節開始所述、本 IC 做螺旋補間時Z軸的移動為求均一的關係需要先知知道圓弧補間的總輸出脈衝數。而螺旋演算命令為依此目的產生的求取總輸出脈衝數的命令。

螺旋演算執行前、前項1~4 的補間軸、補間速度資料、螺旋回轉數、位置資料(圓弧補間中心點·終點)一定要先設定。螺旋演算才可以把這些參數值代入執行。

螺旋演算可分為 CW 螺旋演算與CCW螺旋演算。使用時務必執行與螺旋補間相同的圓弧回轉方向演算命令。回轉方向不同時補間無法正常執行。

WR0 暫存器中寫入 CW 螺旋演算命令(6Bh)、或 CCW 螺旋演算命令(6Ch)後執行時。

表 3.3-3 螺旋演算命令

螺旋演算命令碼	執行螺旋演算
6Bh	CW螺旋演算
6Ch	CCW螺旋演算

演算執行中 RR0 暫存器的 D0,D1(XDRV、YDRV)位元為1、演算結束後該位元回到0。演算結束時也可以利用驅動結束發出中斷的方式獲取訊息。有關中斷的資訊請參考、2.11 節。

■ 螺旋演算結果讀取及寫入

執行螺旋演算結束後、可以獲取螺旋演算結果(圓弧補間總輸出脈衝數)。此值可以螺旋演算值讀取命令(3Bh)讀取。WR0 暫存器寫入螺旋演算值讀取命令(3Bh)、再由 RR6,RR7 暫存器讀取。

完全相同的螺旋補間(螺旋回轉數及圓弧補間中心點·終點值及Z·U軸進給量相同)執行多次時、不需每次做補間前的螺旋演算。獲取1次演算結果由螺旋演算值讀取命令(3Bh)讀取、下次只要將該值設定就可以進行螺旋補間。螺旋演算結果的寫入是以螺旋演算值寫入命令(1Bh)進行。螺旋演算結果寫入 WR6,7 暫存器、再將螺旋演算值寫入命令(1Bh)寫入 WR0 暫存器後 IC 內部暫存器的值就被設定。

【注意】

- 螺旋演算及螺旋補間、補間模式設定命令(2Ah)的全部位元內容如果不同，動作會不正確。

■ 螺旋演算執行時間

螺旋演算執行時間如下表所示。螺旋演算執行時間因螺旋補間時的XY軸圓弧半徑大小而異。演算時間最多不超過圓弧 1 轉時間。螺旋回轉數 1 轉以上的時候、無論幾轉多是下表所示值。不滿 1 轉的時候因回轉角度大小其值會比下表小。

表 3.3-4 螺旋演算執行時間

圓弧補間的半徑 r (脈衝)	螺旋演算執行時間 t (msec)	
	短軸脈衝均一化模式無効	短軸脈衝均一化模式有効
1,000	0.7	5.6
10,000	7	56
100,000	70	565
1,000,000	707	5,656

圓弧補間的半徑可以由本 IC 設定圓弧中心點(xc,yc)獲得。圓弧半徑及執行時間關係如下式算出。

$$\text{圓弧半徑 } r = \sqrt{(x_c^2 + y_c^2)}$$

$$\text{執行時間 } t = (1 \times 10^{-6} \times r) / \sqrt{2}$$

【注意】

- 短軸脈衝均一化模式有無設定其執行時間差為 8 倍。

3.3.6 螺旋補間執行

螺旋補間可以 CW 螺旋補間驅動命令(69h)或 CCW 螺旋補間驅動命令(6Ah)來執行。XY 平面上的圆弧補間以 CW 回轉時執行 CW 螺旋補間驅動命令(69h)、CCW 回轉時執行 CCW 螺旋補間驅動命令(6Ah)，當命令寫入 WR0 暫存器後螺旋補間開始。

表 3.3-5 螺旋補間命令

螺旋補間命令碼	執行的螺旋補間
69h	CW螺旋補間
6Ah	CCW螺旋補間

CW回轉為、XY直交平面(X軸橫軸、Y軸縱軸、X軸+方向右、Y軸+方向上方向的平面)，在此平面順時針方向上描繪的圆弧補間。相反的逆時針回轉時為 CCW 回轉。

螺旋補間執行前、補間所需資料一定要先設定。詳細參考本節 3.3.1 項到 3.3.5 項為止的說明。

3.3.7 讀取現在螺旋回轉數

螺旋補間執行中、可以現在螺旋回轉數讀取命令(3Ah)讀取現在回轉數。

螺旋回轉數為遞增計數器、圆弧補間轉1圈回到起點時計數。

3.3.8 螺旋補間的位置變動

螺旋補間在XY平面上做圆弧補間時、與圆弧補間同步進行 Z 軸或U軸的移動。理想上相對於圆弧補間中心的回轉角增加量與Z·U軸進給增加量如圖 3.3-5 所示、不成正比例不可。但、本 IC 圆弧補間在XY直交平面上進行的關係、輸出 X 軸及Y軸的驅動脈衝量增加與圆弧補間中心的回轉角增加量不完全為正比例的關係。圆弧補間進行時 X 軸Y軸的驅動脈衝量是由演算所得，因此會影響到Z·U軸的進給而成非完全正比例。圆弧補間的象限改變時會發生周期性的變動。

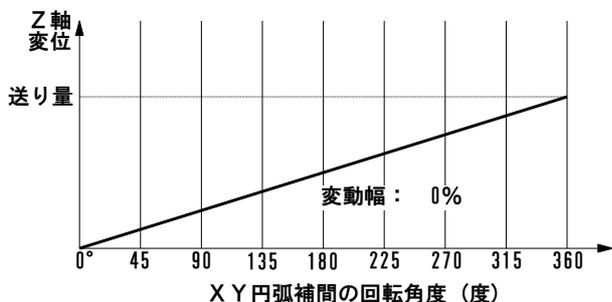


圖 3.3-5 理想的螺旋補間 Z 軸進給

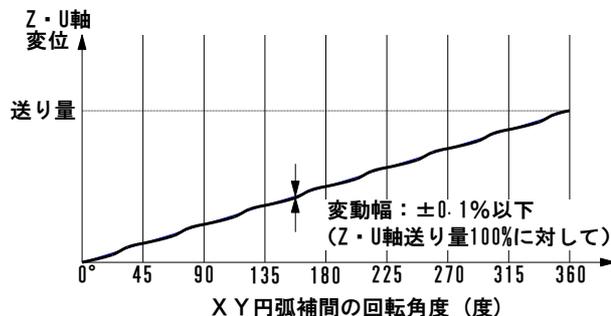


圖 3.3-6 MCX514 螺旋補間 Z·U 軸進給變動

本 IC 如圖 3.3-6 所示、Z或U軸的位置在圆弧補間象限改變時會發生周期的變動(飄移)。與理想位置的變動寬度因動作條件的關係如下所示。

表 3.3-6 理想進給位置的變動寬

動作条件	理想位置變動寬
短軸脈衝均一化+線速一定 2 軸高精度	±0.1%以下
無短軸脈衝均一化、線速一定	±0.4%以下

有關短軸脈衝均一化請參考 3.6 節。

有關線速一定請參考 3.5 節。

【注意】

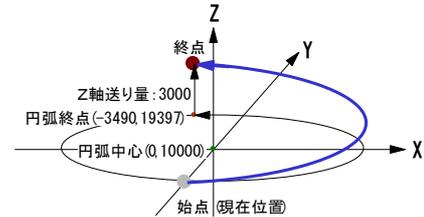
- 使用脈衝均一化時、圆弧的始点與終点若不在 X 軸、Y 軸的軸上時、一轉時終点位置會在兩軸間產生 ±1 脈衝的誤差。此誤差會有累積現象。因此圆弧始点與終点不在 X 軸、Y 軸的軸上時，短軸脈衝均一化使用是否會產生問題請務必確實檢討是否可用。圆弧始点與終点如果有在 X 軸或 Y 軸的軸上時，此誤差不會發生。

3.3.9 螺旋補間實例

螺旋補間以下幾個參考實例、示範MCX514的設定方法。

(1) 1 轉未滿的螺旋補間例 (X,Y,Z 軸)

如右圖所示、始點(現在位置)到(X:0,Y:10000)的相對位置為中心做CCW 圓弧補間、終點在(X:-3490,Y:19397)時圓弧補間結束。此時 Z 軸配合圓弧補間由現在位置移動+3000。圓弧補間速度為等速 1000pps。



```

WR6 ← 01C7h 寫入 ; X Y 軸圓弧 + Z 軸指定、2軸高精度線速一定
WR0 ← 002Ah 寫入 ; 短軸脈衝均一化有効

WR6 ← 03E8h 寫入 ; 1000 PPS
WR7 ← 0000h 寫入
WR0 ← 0104h 寫入 ; 設定主軸 X 初速度

WR6 ← 03E8h 寫入 ; 1000 PPS
WR7 ← 0000h 寫入
WR0 ← 0105h 寫入 ; 設定主軸 X 驅動速度

WR6 ← 0000h 寫入 ; 螺旋回轉數 : 0
WR0 ← 001Ah 寫入

WR6 ← 0000h 寫入 ; 圓弧中心 X : 0
WR7 ← 0000h 寫入
WR0 ← 0108h 寫入

WR6 ← 2710h 寫入 ; 圓弧中心 Y : 10000
WR7 ← 0000h 寫入
WR0 ← 0208h 寫入

WR6 ← F25Eh 寫入 ; 圓弧終點 X : -3490
WR7 ← FFFFh 寫入
WR0 ← 0106h 寫入

WR6 ← 4BC5h 寫入 ; 圓弧終點 Y : 19397
WR7 ← 0000h 寫入
WR0 ← 0206h 寫入

WR6 ← 0BB8h 寫入 ; Z 進給量 : 3000
WR7 ← 0000h 寫入
WR0 ← 0406h 寫入

WR0 ← 006Ch 寫入 ; C C W螺旋演算 (演算時間 : 約56ms)

RR0 → 讀取 ; 等待演算結束(等待D0位元= 0)

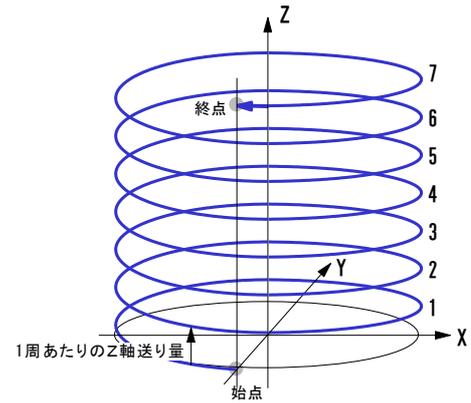
WR0 ← 006Ah 寫入 ; C C W螺旋補間驅動開始

RR0 → 讀取 ; 等待補間結束(等待D0位元= 0)

```

(2) 複数回転螺旋補間例 (X,Y,Z 軸)

半径 10000 の円弧補間 1 圏時 Z 軸進給量 3000 脈衝、總共 7 圏。
螺旋補間複数圏執行時、Z 軸進給量以円弧補間 1 轉の進給量設定。始点(現在位置)到 (X:0,Y:10000) 的相对位置為中心做 CW 円弧補間。円弧 1 轉時 Z 軸前進 3000 脈衝、円弧 7 周後結束。円弧補間速度為等速 1000pps。



```

WR6 ← 00C7h 寫入      ; X Y 軸円弧 + Z 軸指定、2軸高精度線速一定
WR0 ← 002Ah 寫入      ; 短軸脈衝均一化：無効

WR6 ← 03E8h 寫入      ; 1000 PPS
WR7 ← 0000h 寫入
WR0 ← 0104h 寫入      ; 設定主軸 X 初速度

WR6 ← 03E8h 寫入      ; 1000 PPS
WR7 ← 0000h 寫入
WR0 ← 0105h 寫入      ; 設定主軸 X 驅動速度

WR6 ← 0007h 寫入      ; 螺旋回転数：7
WR0 ← 001Ah 寫入

WR6 ← 0000h 寫入      ; 円弧中心 X：0
WR7 ← 0000h 寫入
WR0 ← 0108h 寫入

WR6 ← 2710h 寫入      ; 円弧中心 Y：10000
WR7 ← 0000h 寫入
WR0 ← 0208h 寫入

WR6 ← F25Eh 寫入      ; 円弧終点 X：0
WR7 ← FFFFh 寫入
WR0 ← 0106h 寫入

WR6 ← 4BC5h 寫入      ; 円弧終点 Y：0
WR7 ← 0000h 寫入
WR0 ← 0206h 寫入

WR6 ← 0BB8h 寫入      ; Z 進給量：3000
WR7 ← 0000h 寫入
WR0 ← 0406h 寫入

WR0 ← 006Bh 寫入      ; CW螺旋演算 (演算時間：約56ms)

RR0 → 讀取           ; 等待演算結束(等待D0位元=0)

WR0 ← 0069h 寫入      ; CW螺旋補間驅動開始

RR0 → 讀取           ; 等待補間結束(等待D0位元=0)

```

(3) Z,U 兩軸螺旋補間例 (X,Y,Z,U 軸)

半径 10000 的円弧補間做 CCW 方向 1 周。円弧補間 1 轉間時、Z 軸進給 3000 脈衝、回轉軸 U 軸進給 400 脈衝。円弧補間速度等速 1000pps。

```

WR6 ← 00CFh 寫入      ; X Y 軸円弧 + Z, U 軸指定、2 軸高精度線速一定
WR0 ← 002Ah 寫入      ; 短軸脈衝均一化：無効

WR6 ← 03E8h 寫入      ; 1000 PPS
WR7 ← 0000h 寫入
WR0 ← 0104h 寫入      ; 設定主軸 X 初速度

WR6 ← 03E8h 寫入      ; 1000 PPS
WR7 ← 0000h 寫入
WR0 ← 0105h 寫入      ; 設定主軸 X 驅動速度

WR6 ← 00007h 寫入     ; 螺旋回轉數：1
WR0 ← 001Ah 寫入

WR6 ← 0000h 寫入     ; 円弧中心 X：0
WR7 ← 0000h 寫入
WR0 ← 0108h 寫入

WR6 ← 2710h 寫入     ; 円弧中心 Y：10000
WR7 ← 0000h 寫入
WR0 ← 0208h 寫入

WR6 ← F25Eh 寫入     ; 円弧終點 X：0
WR7 ← FFFFh 寫入
WR0 ← 0106h 寫入

WR6 ← 4BC5h 寫入     ; 円弧終點 Y：0
WR7 ← 0000h 寫入
WR0 ← 0206h 寫入

WR6 ← 0BB8h 寫入     ; Z 軸進給量：3000
WR7 ← 0000h 寫入
WR0 ← 0406h 寫入

WR6 ← 0190h 寫入     ; U 軸進給量：400
WR7 ← 0000h 寫入
WR0 ← 0806h 寫入

WR0 ← 006Ch 寫入     ; C C W 螺旋演算 (演算時間：約 56ms)

RR0 → 讀取           ; 等待演算結束 (等待 D0 位元 = 0)

WR0 ← 006Ah 寫入     ; C C W 螺旋補間驅動開始

RR0 → 讀取           ; 等待補間結束 (等待 D0 位元 = 0)

```

3.4 位元補間

本 IC 的位元補間是以 1 驅動脈衝單位做 + 方向或 - 方向脈衝輸出與否的指定後進行複數軸的補間動作。補間軸數可以 2 軸到最大 4 軸。

由 CPU 控制各軸 + 方向或 - 方向的驅動脈衝，設定上 1 位元 1 脈衝。驅動脈衝輸出時設 "1"、不輸出設為 "0"。例如右圖 3.4-1 描繪軌跡時、X+ 方向、X- 方向、Y+ 方向、Y- 方向依序設定驅動脈衝輸出 "1"、不輸出 "0"、位元資料如下所示。

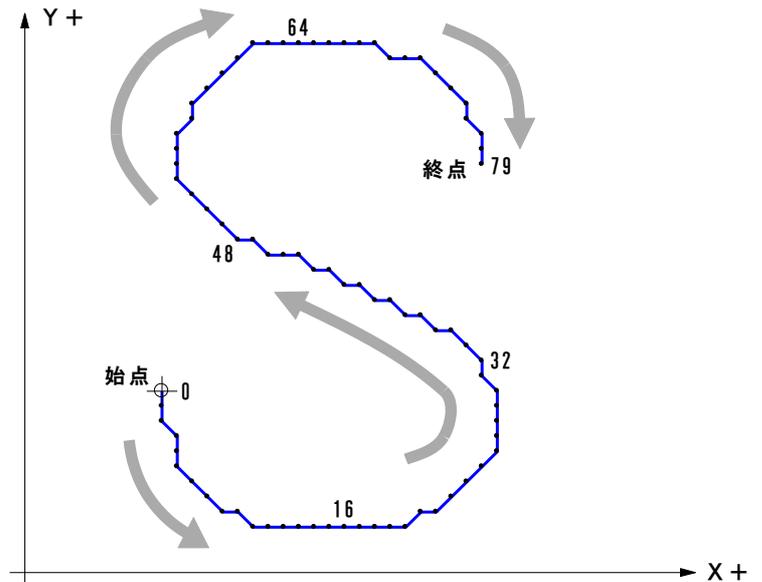


圖 3.4-1 X Y 軸位元補間例

79	64	48	32	16	0	
10010111	11111111	11111110	10000000	00000000	00000000	00000011 11111111 11111111 11100100 : XPP (X+方向パルス)
10000000	00000000	00000000	00001111	11111111	11111111	01000000 00000000 00000000 00000000 : XPM (X-方向パルス)
00000000	00000000	00011111	11111111	01001010	10101011	11111111 11010000 00000000 00000000 : YPP (Y+方向パルス)
01111111	00100000	00000000	00000000	00000000	00000000	00000000 00000000 00000011 11111111 : YPM (Y-方向パルス)

位元補間動作進行時其操作順序如下所示。

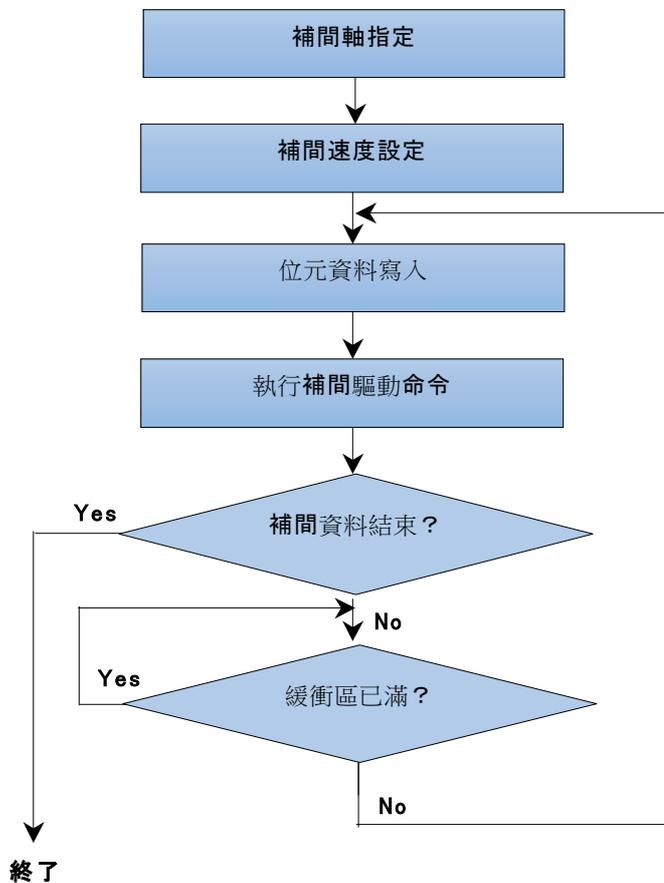
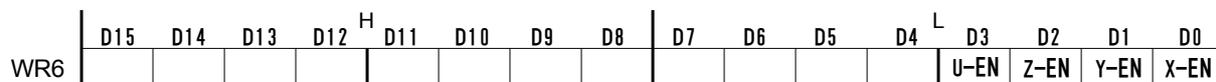


圖 3.4-2 位元補間操作順序

3.4.1 補間軸指定

執行補間模式設定命令(2Ah)設定補間軸。如下所示設定WR6暫存器的D0到D3位元。要補間的軸該位元設1。可以設定2軸到4軸全軸位元補間。不能只設1軸。



WR6暫存器的其他位元(D15~D4)為補間相關機能的設定位元。詳細請參考7.3.8項做適當值設定。

3.4.2 補間速度設定

針對補間軸中的主軸設定位元補間驅動速度。

BP(位元圖樣)的補間驅動速度最高4MHz。但、位元數超過128位元時、CPU需要在補間驅動中補充後續的BP資料、所以補間驅動速度與CPU在BP資料設定所要時間有依存關係。

例如執行2軸位元補間時、CPU側設定BP資料之故、需有足夠時間寫入(16位元資料×2+16位元命令)×2軸+補間驅動命令。假設此時間需要100μSEC、則16位元(=16驅動脈衝)輸出所需的時間要比它長。因此補間驅動速度需設定在1/(100μSEC/16)=160KPPS以下。若設速度超過此速度、則BP資料補充會來不及。

3.4.3 位元資料寫入

位元資料寫入補間軸。

WR6暫存器寫入16位元分的+方向位元資料、WR7暫存器寫入16位元分-方向位元資料。16位元資料由D0位元往高位位元的順序輸出驅動脈衝。

軸指定以BP資料設定命令(06h)寫入WR0暫存器後、BP資料會收納在預存緩衝暫存器。此操作在補間進行軸進行。

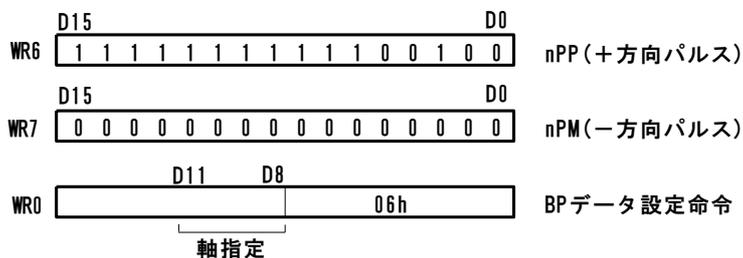


図 3.4-3 位元資料寫入

3.4.4 執行補間驅動命令

全部位元資料寫入後、位元補間驅動命令寫入 WR0 暫存器。
2軸到4軸為止的補間開始驅動、補間驅動命令編碼如下所示。

表 3.4-1 位元補間命令

補間命令	命令碼
2 軸位元補間	66h
3 軸位元補間	67h
4 軸位元補間	68h

不需指定軸。命令寫入 WR0 暫存器後，預存緩衝暫存器就會更新1段(堆疊計數器加1)、補間驅動開始執行。要使用預存緩衝暫存器中的位元資料開始做補間時、事先先對主軸設定驅動開始保持命令(77h)。等到位元資料及補間命令的複數段寫入後再對主軸下驅動開始釋放命令(78h)，這樣補間驅動就開始執行。

【注意】

- 全軸位元資料寫入後一定要接著寫入位元補間命令。因為預存緩衝暫存器資料更新是以位元補間命令寫入來進行。

3.4.5 補間結束

位元補間由以下2種方法結束。

① 寫入補間軸位元資料結束碼。

位元補間軸不管軸+方向、-方向其位元資料多寫入"1"時，判定為位元補間結束。結束碼以降的位元資料無効。



図 3.4-4 以結束碼做位元補間結束

② 資料寫入中止。

中止寫入位元資料後、預存緩衝暫存器中堆疊有的資料處理完後補間驅動結束。

3.4.6 確認預存緩衝暫存器是否空的

本IC各軸具有8段預存緩衝暫存器來做連續補間驅動用。位元補間的時候補間的全部軸可利用此8段16位元儲存 $16 \times 8 = 128$ 位元的資料。如果補間資料超過128位元時、需確認預存緩衝暫存器是否為空的再做補間。這時候可以讀取RR0暫存器D12~D15的4位元、這4位元被當成預存緩衝暫存器的狀態計數值顯示。4位元值0為完全空狀態，8為滿的狀態、這以上的BP資料寫入不可。位元補間命令寫入時狀態計數器值會增加1，同時補間驅動開始。16位元輸出結束後狀態計數器值減少1。

另外、RR0暫存器D11(CNEXT)位元為連續補間驅動下一筆資料寫入許可位元。補間驅動開始後、預存緩衝暫存器狀態計數器在1到7間CNEXT位元為1。上位CPU判斷此位元1時、下一筆資料可以寫入。



3.4.7 補間驅動停止

■ 執行停止命令

位元補間驅動時對主軸下達即停止命令或減速停止命令後補間驅動停止。
預存緩衝暫存器的狀態計數器強制的歸0。預存緩衝暫存器內的位元資料全部無効。

■ 硬體極限、軟體極限

補間驅動中、不管是軸的硬體極限、或是軟體極限動作時補間驅動都會停止。
位元補間時+方向/-方向各方向的硬體極限、或是軟體極限動作時補間也會停止。因此位元補間時請勿超出極限領域使用。

3.4.8 位元補間驅動例

X軸及Y軸做 $m \times 16$ 位元的位元補間時。如圖3. 4-1位元圖樣圖形例共有79位元，所以 $m=5$ 。補間驅動速度1000PPS的等速驅動、2軸簡易線速一定模式補間。X軸及Y軸補間以X軸為主軸、所以驅動速度對X軸設定。

位元資料如下所示。

	m = 5	m = 4	m = 3	m = 2	m = 1
X 軸 + 方向資料 X_PlusBPdata (m)	1001011111111111 97FFh	1111111010000000 FE80h	0000000000000000 0000h	0000011111111111 03FFh	1111111111100100 FFE4h
X 軸 - 方向資料 X_MinusBPdata (m)	1000000000000000 8000h	0000000000001111 000Fh	1111111111111111 FFFFh	0100000000000000 4000h	0000000000000000 0000h
Y 軸 + 方向資料 Y_PlusBPdata (m)	0000000000000000 0000h	0001111111111111 1FFFh	0100101010101011 4AABh	1111111110100000 FFD0h	0000000000000000 0000h
Y 軸 - 方向資料 Y_MinusBPdata (m)	0111111100100000 7F20h	0000000000000000 0000h	0000000000000000 0000h	0000000000000000 0000h	0000011111111111 03FFh

WR6 ← 0043h 寫入 ; 指定 X 軸及 Y 軸、2 軸簡易線速一定
WR0 ← 002Ah 寫入 ; 補間模式設定

WR6 ← 03E8h 寫入 ; X 軸 (主軸) 速度參數設定
WR7 ← 0000h 寫入 ; 初速度 : 1000 PPS
WR0 ← 0104h 寫入

WR6 ← 03E8h 寫入 ; 驅動速度 : 1000 PPS
WR7 ← 0000h 寫入
WR0 ← 0105h 寫入

m ← 1 ; 資料指標 = 1

Loop:

WR6 ← X_PlusBPdata (m) 寫入 ; X 軸 + 方向BP資料
WR7 ← X_MinusBPdata (m) 寫入 ; X 軸 - 方向BP資料
WR0 ← 0106h 寫入

WR6 ← Y_PlusBPdata (m) 寫入 ; Y 軸 + 方向BP資料
WR7 ← Y_MinusBPdata (m) 寫入 ; Y 軸 - 方向BP資料
WR0 ← 0206h 寫入

WR0 ← 0066h 寫入 ; 2 軸 B P 補間命令 此步驟執行後補間驅動開始。

m ← m + 1 ; 資料指標增加
m = 6 的時候結束

RR0 → 讀取 ; 確認預存緩衝暫存器空?
RR0/D11 = 1 的時候 Loop 跳躍。= 0的時候讀取RR0

■ 使用中斷訊號做位元補間驅動

連續補間專用中斷訊號 (INT1N) 在預存緩衝暫存器的狀態計數器由8變為7、或是4變為3的時候其位準會動作 (Low位準)。上位CPU可依此中斷訊號發生時、狀態計數器變為8為止 (= CNEXT位元1的時候) 寫入下筆BP資料。其中選擇8到7時可以寫入1段、選擇4到3時可以連續寫入5段BP資料。

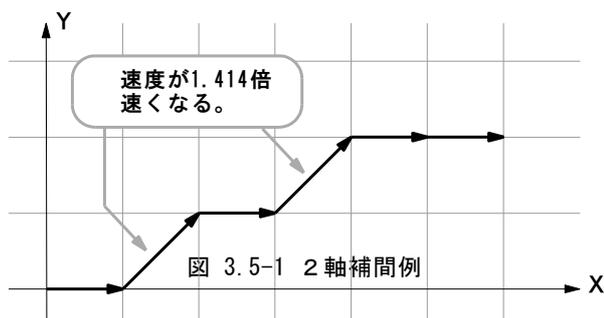
中斷訊號 (INT1N) 在位元資料或補間命令 (2/3/4軸位元補間命令等) 寫入後回到未動作狀態。另外、補間驅動結束後強制的回到非動作的狀態。

3.5 線速一定

線速為補間驅動時軌跡先端的移動速度、也可以稱之為前頭速度。補間驅動做工件加工或塗布等動作時、此線速保持一定為其重要的條件。本 IC 在 2 軸補間上、可以選擇 2 軸簡易線速一定模式及 2 軸高精度線速一定模式。其他 3 軸補間也有 3 軸簡易線速一定模式可以使用。

圖 3.5-1 所示為 XY 直行平面 2 軸補間的軌跡。依據主軸基本脈衝下各軸驅動脈衝輸出時、從圖上可以看出 X、Y 軸兩方都有驅動脈衝輸出時、相較於只有 1 軸驅動脈衝輸出的移動距離為 1.414 倍長。

線速一定機能沒有啟動時、兩軸以相同的周期輸出驅動脈衝、而移動距離卻是 1.414 倍的情況下、移動速度快了 1.414 倍。



如下圖 3.5-2、XY 直交平面下做直線補間時、X 軸線與補間直線所形成的角度在 0 到 90 度範圍、顯示其線速度的偏差。圖中記載在 0-90 度的範圍、其他 90-180、180-270、270-360 的範圍也一樣。

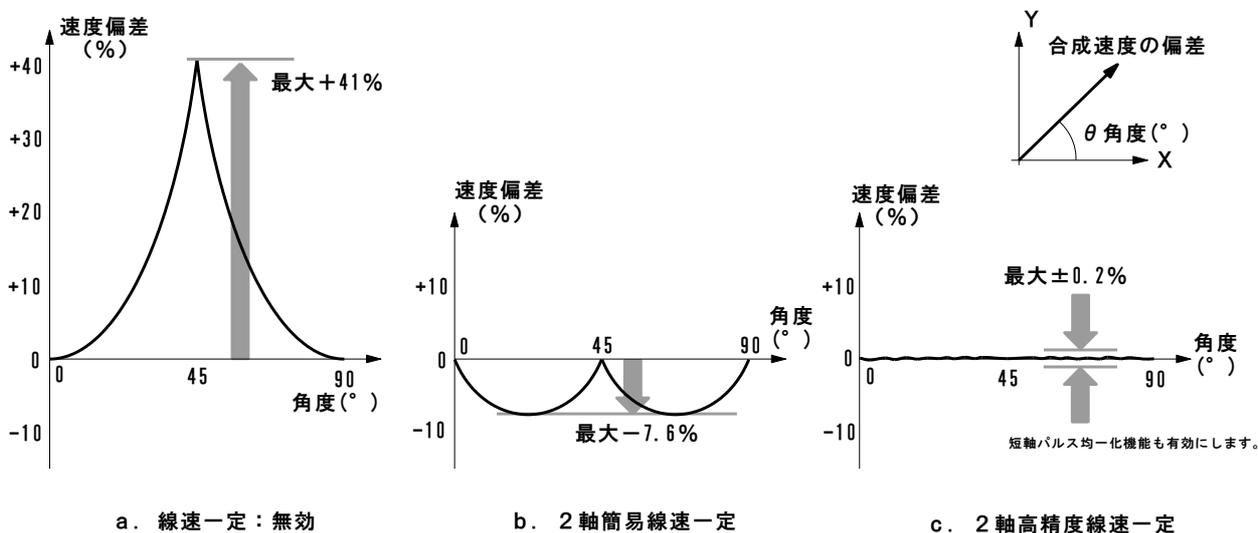


圖 3.5-2 直線補間驅動的設等速度に対する線速度的偏差

圖 3.5-2 a 顯示線速一定機能無效時設定的驅動速度與線速度的偏差。與 X 軸成 45 度角的時候速度偏差最大、速度約增加 +41%。

圖 3.5-2 b 顯示 2 軸簡易線速一定模式時的速度偏差。2 軸簡易線速一定模式下、兩軸同時輸出驅動脈衝時、其脈衝周期為降為 $1/1.414$ 倍以改善速度偏差的方法。

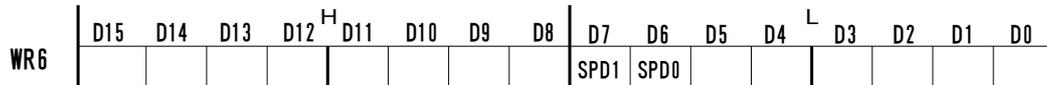
圖 3.5-2 c 顯示 2 軸高精度線速一定的模式設定下的速度偏差。全部的角度範圍下其速度偏差控制在 $\pm 0.2\%$ 以下*1。

*1：短軸脈衝均一化機能需設為有效。

3 軸直線補間時可使用 3 軸簡易線速一定模式。3 軸簡易線速一定模式為、3 軸中只要同時有 2 軸的驅動脈衝輸出時、其脈衝周期就降為 $1/1.414$ 倍、3 軸全部有驅動脈衝輸出時其脈衝周期就降為 $1/1.732$ 倍以改善速度偏差。

3.5.1 線速一定設定

線速一定為補間模式設定(2Ah)的 D6,D7 位元設定。



各式線速一定模式所對應的D6,D7位元設定如下表。

表 3.5-1 線速一定模式設定

D7 (SPD1) 位元	D6 (SPD0) 位元	線速一定模式
0	0	無效
0	1	2 軸簡易線速一定
1	0	3 軸簡易線速一定
1	1	2 軸高精度線速一定

■ 高精度2軸線速一定直線補間驅動例

X軸 Y 軸直線補間以驅動速度 1000PPS 等速驅動、且以2軸高精度線速一定模式進行例。短軸脈衝均一化機能也有効時。

// 補間模式設定

WR6 ← 01C3h 寫入

WR0 ← 002Ah 寫入

// X, Y軸補間、2 軸高精度線速一定模式、短軸脈衝均一化有効

// 補間模式設定命令

// 主軸驅動設定

WR6 ← 03E8h 寫入

WR7 ← 0000h 寫入

WR0 ← 0104h 寫入

// 初速度 : 1000pps

WR6 ← 03E8h 寫入

WR7 ← 0000h 寫入

WR0 ← 0105h 寫入

// 驅動速度 : 1000pps

// 終点設定

WR6 ← 03E8h 寫入

WR7 ← 0000h 寫入

WR0 ← 0106h 寫入

// 終点 X值 : 1000

WR6 ← 0190h 寫入

WR7 ← 0000h 寫入

WR0 ← 0206h 寫入

// 終点 Y值 : 400

// 補間驅動開始

WR0 ← 0061h 寫入

// 2軸直線補間驅動

3.6 短軸脈衝均一化

通常補間驅動在補間進行時，全部軸在驅動中、不會隨時保持均一周期的驅動脈衝輸出。例如下圖 3.6-1 a. 所示、2 軸直線補間進行時、移動量(脈衝)多的軸(長軸)會持續有脈衝輸出、少的軸(短軸)會因補間演算結果而脈衝輸出會時有時無。使用步進馬達的時候、此短軸脈衝的現象可能引起機械的振動增大。

短軸脈衝均一化是為求改善此問題的機能。即使移動量少的軸也盡力讓其脈衝周期接近均一的輸出驅動脈衝。下圖 3.6-1 b. 為短軸脈衝均一化機能有效下輸出驅動脈衝的波形。

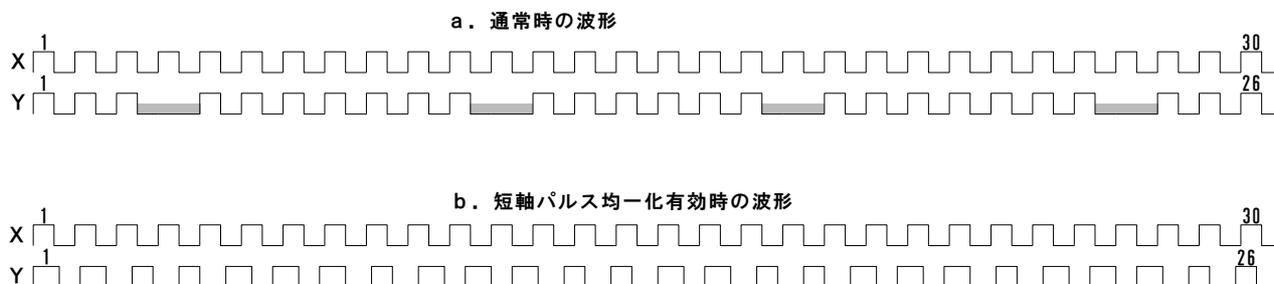


圖 3.6-1 2 軸直線補間（終点 X30, Y26）的脈衝波形

短軸脈衝均一化在 IC 內部的補間演算通常以高於數倍進行。因此可設定的參數範圍如表所示的 1/8 為其限制。短軸脈衝均一化有效時、請務必在下表所示設定可能範圍內進行補間驅動。

表 3.6-1 短軸脈衝均一化的參數設定範圍

設定參數	記号	設定可能範圍	
		短軸脈衝均一化：有效	通常
驅動速度	V	1~1,000,000	1~8,000,000
初速度	S V	1~1,000,000	1~8,000,000
加速度, 減速度	A, D	1~67,108,863	1~536,870,911
終点	P	-268,435,456~268,435,456	-2,147,483,646~2,147,483,646
圆弧中心点	C	-268,435,456~268,435,456	-2,147,483,646~2,147,483,646

【注意】

- S形加減速驅動不能使用短軸脈衝均一化機能。

3.6.1 短軸脈衝均一化的設定

短軸脈衝均一化請使用は補間模式設定命令(2Ah)的D8位元設定。

WR 6	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
								LMDF								

設定 1 時短軸脈衝均一化有效、設定 0 時短軸脈衝均一化無效。

【注意】

- 多晶片補間時短軸脈衝均一化機能無法使用。
- 補間單步進給時短軸脈衝均一化機能無法使用。

3.7 連續補間

連續補間為直線補間→凹弧補間→直線補間→…等等、各種補間區段驅動在不停止的情況下、連續執行補間動作。連續補間的執行其補間軸數要相同為其前提、如下表是可以執行的連續補間。

表 3.7-1 可以執行的連續補間

執行可能的連續補間	動作
2軸直線補間連續	2軸直線 → 2軸直線 → 2軸直線 → ……
2軸直線補間及凹弧補間的連續	2軸直線 → 凹弧補間 → 2軸直線 → 2軸直線 → 凹弧補間 → ……
3軸直線補間的連續	3軸直線 → 3軸直線 → 3軸直線 → ……
4軸直線補間的連續	4軸直線 → 4軸直線 → 4軸直線 → ……

連續補間是使用預存緩衝暫存器來實現。補間驅動開始前、或是執行中、將補間資料寫入預存緩衝暫存器來實現連續補間驅動。預存緩衝暫存器最多可以寫入8段補間資料。

3.7.1 連續補間的實施方法

執行連續補間時、預存緩衝暫存器要先寫入補間資料後才開始補間驅動。補間驅動開始前最多 8 段補間資料可放在預存緩衝暫存器。補間驅動開始後、確認預存緩衝暫存器的狀態計數器值後再寫入下個補間資料來執行連續補間。

連續補間操作順序如下。

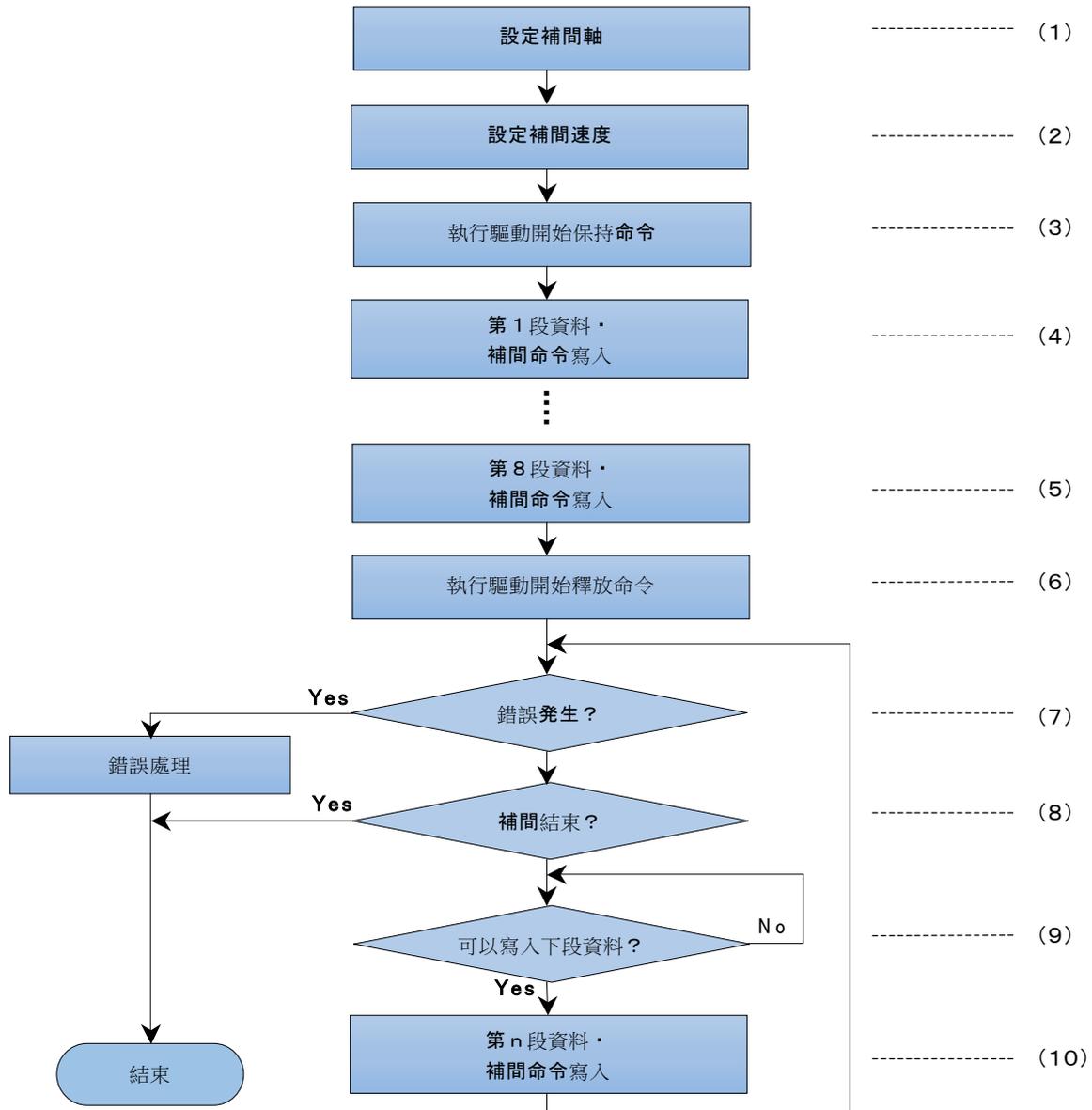


圖 3.7-1 連續補間流程

(1) 補間軸設定

補間模式設定命令(2Ah) 設定補間軸。如下所述設定 WR6 暫存器 D0 到 D3 位元。將要做補間的軸對應位元設為1。

WR6	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
													U-EN	Z-EN	Y-EN	X-EN

- 連續補間驅動開始後，補間軸設定不能變更。
- WR6暫存器其他位元(D15~D4)也是與補間相關機能的設定位元。請參考7.3.8項說明做適當值的設定。

(2) 補間速度設定

針對補間進行軸中的主軸、設定補間驅動的速度。可以設的最高速度為4Mpps。

連續補間中全部區段都以等速驅動時、請針對主軸設定驅動速度及初速度。以初速度為等速驅動動作之故，驅動速度請設定相同值。

(3) 執行驅動開始保持命令

對主軸發出驅動開始保持命令(77h)。驅動開始保持命令執行後、即使執行補間驅動命令也不會開始驅動。因此預存緩衝暫存器在補間開始前可以設定最多8段補間資料。

(4) 第1段資料・補間命令寫入

第1段為直線補間的時候、對各補間軸寫入終點、然後寫入直線補間驅動命令。圓弧補間的時候對各軸寫入圓弧中心點及終點、然後寫入圓弧補間命令。

1個區段情報寫入時、終點、圓弧中心點、補間軸的寫入順序沒有先後差異，但補間驅動命令請務必在最後階段寫入。

(5) 第8段資料・補間命令寫入

第2區段到第8區段為止、資料及補間驅動命令的寫入與第1區段相同。

中途的區段想要變更驅動速度時、請在補間驅動命令寫入前針對主軸寫入要變更的驅動速度。如果沒有要變更的話，驅動速度不需要再次寫入。

預存緩衝暫存器有8段構成。確認RR0暫存器的D12~D15中顯示的狀態計數器值、可以在補間驅動開始前將8個區段的資料寫入。

(6) 執行驅動開始釋放命令

預存緩衝暫存器存入所要的區段補間資料資料後，針對主軸發出驅動開始釋放命令(78h)後。補間驅動開始執行。

(7) 錯誤的確認

RR0 暫存器的 D4~D7(X~UERR)位元為顯示驅動進行軸有無錯誤發生的狀態。錯誤發生時該當位元變1、補間驅動停止。請確認這些位元沒有錯誤發生後再處理下一動作。RR0 暫存器的錯誤位元詳細請參考 6.13 節。

RR0	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
	HSTC3	HSTC2	HSTC1	HSTC0	CNEXT	ZONE2	ZONE1	ZONE0	UERR	ZERR	YERR	XERR	UDRV	ZDRV	YDRV	XDRV

(8) 補間結束的確認

確認全部區段資料寫入完成否，還沒有的話繼續處理下一個動作。

(9) 確認下筆資料寫入許可

RR0 暫存器 D12~D15 (HSTC0~3) 位元為 8 段預存緩衝暫存器的狀態計數器值分配狀態顯示、顯示緩衝暫存器的儲存量。4 位元值都為 0 時，暫存器為完全空的狀態。8 的時候為用完的狀態、此時無法再寫入新的區段資料。補間驅動命令寫入時狀態計數器值增加 1、現在輸出中的區段驅動結束後狀態計數器值減 1。

另、RR0 暫存器的 D11 (CNEXT) 位元為連續補間驅動可以寫入下一筆資料的狀態位元。此 CNEXT 位元在補間驅動開始後、預存緩衝暫存器的狀態計數器在 1 到 7 之間時為 1。上位 CPU 可以用此位元做為判斷依據，當其值為 1 時可以寫入下一筆資料的。

(10) 第 n 段資料・補間命令寫入

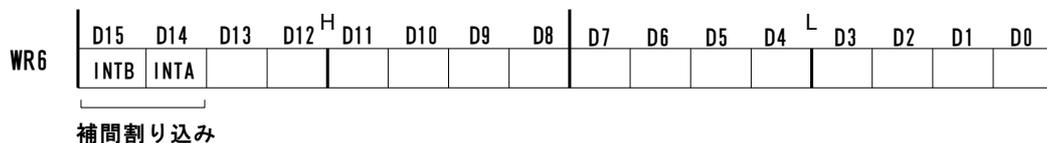
第 9 區段以後視補間驅動執行狀態再寫入區段資料。寫入的內容在 (4) 及 (5) 有說明，與第 1~8 區段相同。補間驅動命令寫入後回到 (7)。

3.7.2 利用中斷訊號做連續補間

連續補間已可以利用中斷訊號來實現。本 IC 的 INT1N 訊號 (端子番号:34) 在預存緩衝暫存器空的時候，訊號位準變為 Low、針對上位通知下一區段資料可以寫入。空狀態的中斷通知方式可以選擇 2 種設定。

補間中斷設定

預存緩衝暫存器空的時候發出中斷、補間模式設定 (2Ah) 的 D14, D15 的 2 位元設定。



D14 (INTA) 位元變為 1 時、預存緩衝暫存器的狀態計數器由 4 變為 3 時，INT1N 訊號變為 Low 的動作位準。通知 8 段預存緩衝暫存器已經有一半空出來。適合連續補間驅動速度較緩慢的地方使用。

D15 (INTB) 位元變為 1 時、預存緩衝暫存器的狀態計數器由 8 變為 7 時，INT1N 訊號變為 Low 的動作位準。通知預存緩衝暫存器已經空出 1 個空間。適合連續補間驅動速度高速的地方使用。

中斷處理

INT1N 訊號產生的中斷訊號、上位 CPU 在中斷處理迴圈中寫入需要的下筆區段資料。書入內容如上記 (4) 及 (5) 的說明，與第 1~8 區段做法相同。每個區段資料最後一定要藉由補間驅動命令寫入。寫入時機可以讀取 RR0 暫存器 D15~D12 位元 (HSTC3~0) 的狀態計數器值確認。

中斷訊號 (INT1N) 的解除

INT1N 訊號在下一次補間驅動命令寫入時自動清除、回到 hi-Z。另外也可以用以下方法清除。

- 執行補間中斷清除命令 (6Fh)
- 結束連續補間驅動

3.7.3 連續補間時錯誤發生

連續補間錯誤發生、可分為超過極限或補間資料寫入錯誤 2 種。

超過極限等錯誤

連續補間驅動途中超過極限等錯誤發生時、現在驅動中的補間區段下停止。因錯誤發生停止時預存緩衝暫存器的狀態計數器清為 0、全部已經寫入的區段資料及補間命令回全部無效。錯誤解除後不會繼續執行。

資料寫入錯誤

補間資料寫入錯誤是現在補間驅動中的區段的下一筆資料來不及寫入時發生。

連續補間時在預存緩衝暫存器設定的最後區段的補間驅動最終脈衝下降前(正理論時)寫入下一筆區段資料及補間驅動命令的動作完成時就不會有此問題、最終脈衝已降下後此區段驅動結束時、下一筆區段的補間驅動命令再寫入、此資料不會被處理。此時、因設定的資料區段沒有執行之故、預存緩衝暫存器的狀態計數器也沒變化。主軸的 RR2 暫存器的 D7 位元(補間錯誤)會成為 1、補間驅動因錯誤而結束。解除此錯誤請對補間軸全部發出錯誤結束·狀態清除命令(79h)。

3.7.4 連續補間注意事項

連續補間注意事項如以。

- 各補間區段為終點等必要資料設定同時設定補間驅動命令。順序相反時動作會不正確。
- 連續補間驅動速度最高 4MHz (CLK=16MHz 時)。
- 預存緩衝暫存器設定後全部補間區段驅動的時間、補間軸錯誤檢查的時間需大於下一筆補間區段資料及命令設定的時間。若時間來不及，現在補間區段驅動結束後下一筆補間區段的驅動命令再寫入時、驅動會暫時停止後再繼續做連續補間的動作。但是此時補間命令的寫入錯誤(補間錯誤)若發生時連續補間就會結束。
- 連續補間在直線補間時全軸終點為 0、圆弧補間時兩軸的中心點都是 0 時、哪一軸多不要有驅動脈衝輸出的資料設定無法寫入。因這些資料寫入時補間動作會不正常。
- 連續補間中有圆弧補間的時候、圆弧補間終點的短軸值會與真值有 $\pm 1\text{LSB}$ 的偏差、為求各區段誤差不累積的關係、請確認每一個圆弧補間終點後、再組成連續補間。另外、使用短軸脈衝均一化模式的時候、依據終點及中心點的設定、圆弧的始點及終點在軸上以外的點時、圆弧補間的終點不管是長軸或短軸多會有 ± 1 脈衝的偏差。
- 2 軸補間到 3 軸補間等、因軸數不同連續補間不能執行。
- 連續補間途中、補間軸指定變更不可。
- 因錯誤而停止時、務必確認錯誤的種別、然後發出錯誤結束·狀態清除命令(79h)解除錯誤的動作。錯誤狀態沒有清除時補間驅動無法進行。
- 連續補間驅動中因停止命令而停止時、預存緩衝暫存器已設定的資料會全部無效。
- 位元補間及螺旋補間無法與其他補間驅動組合成連續補間使用。

3.7.5 連續補間の実施例

図3.7-2は、(0, 0)作為起點、區段S1到區段S21為止做連續補間例。2軸直線補間與円弧補間的組合使用。円弧補間半径500及半径1000の1/4円。補間速度1000PPS等速驅動、2軸高精度線速一定模式下進行。區段S1(X0, Y6000)的位置開始執行。下表為各區段補間命令及設定資料值。

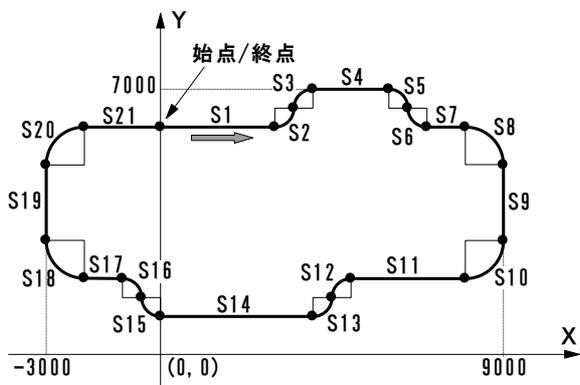


図 3.7-2 連續補間例

區段番号	補間命令	終点 X	終点 Y	円弧中心 X	円弧中心 Y
S1	2軸直線	3000	0		
S2	CCW円弧	500	500	0	500
S3	CW円弧	500	500	500	0
S4	2軸直線	2000	0		
S5	CW円弧	500	-500	0	-500
S6	CCW円弧	500	-500	500	0
S7	2軸直線	1000	0		
S8	CW円弧	1000	-1000	0	-1000
S9	2軸直線	0	-2000		
S10	CW円弧	-1000	-1000	-1000	0
S11	2軸直線	-3000	0		
S12	CCW円弧	-500	-500	0	-500
S13	CW円弧	-500	-500	-500	0
S14	2軸直線	-4000	0		
S15	CW円弧	-500	500	0	500
S16	CCW円弧	-500	500	-500	0
S17	2軸直線	-1000	0		
S18	CW円弧	-1000	1000	0	1000
S19	2軸直線	0	2000		
S20	CW円弧	1000	1000	1000	0
S21	2軸直線	2000	0		

```
//---補間軸指定・モード設定 -----
WR6 ← 00C3h 寫入 ; X, Y 軸指定、2軸高精度線速一定
WR0 ← 002Ah 寫入

//---補間驅動速度設定 -----
WR6 ← 03E8h 寫入 ; 1000 PPS
WR7 ← 0000h 寫入
WR0 ← 0104h 寫入 ; 主軸(X)初速度設定

WR6 ← 03E8h 寫入 ; 1000 PPS
WR7 ← 0000h 寫入
WR0 ← 0105h 寫入 ; 主軸驅動速度設定

//---驅動開始保持 -----
WR0 ← 0177h 寫入 ; 主軸驅動開始保持(HOLD)

//--- 區段 1 2軸直線 設定 -----
WR6 ← 0BB8h 寫入 ; 終点 X : 3000
WR7 ← 0000h 寫入
WR0 ← 0106h 寫入 ; 終点設定命令
WR6 ← 0000h 寫入 ; 終点 Y : 0
WR7 ← 0000h 寫入
WR0 ← 0206h 寫入
WR0 ← 0061h 寫入 ; 2軸直線補間命令

//--- 區段 2 CCW 円弧 設定 -----
WR6 ← 01F4h 寫入 ; 終点 X : 500
WR7 ← 0000h 寫入
WR0 ← 0106h 寫入
WR6 ← 01F4h 寫入 ; 終点 Y : 500
WR7 ← 0000h 寫入
WR0 ← 0206h 寫入
WR6 ← 0000h 寫入 ; 円弧中心 X : 0
WR7 ← 0000h 寫入
WR0 ← 0108h 寫入
WR6 ← 01F4h 寫入 ; 円弧中心 Y : 500
WR7 ← 0000h 寫入
WR0 ← 0208h 寫入
WR0 ← 0065h 寫入 ; CCW円弧補間命令

//--- 區段 3 CW 円弧 設定 -----
WR6 ← 01F4h 寫入 ; 終点 X : 500
WR7 ← 0000h 寫入
WR0 ← 0106h 寫入
WR6 ← 01F4h 寫入 ; 終点 Y : 500
WR7 ← 0000h 寫入
WR0 ← 0206h 寫入
WR6 ← 01F4h 寫入 ; 円弧中心 X : 500
WR7 ← 0000h 寫入
WR0 ← 0108h 寫入
WR6 ← 0000h 寫入 ; 円弧中心 Y : 0
WR7 ← 0000h 寫入
WR0 ← 0208h 寫入
```

WR0 ← 0064h 寫入 ; CW円弧補間命令

區段4~8設定相同。

//---驅動開始釋放-----

WR0 ← 0178h 寫入 ; 主軸に驅動開始釋放命令発行
; 補間驅動開始。

區段計數器設為9。

Loop:

//---錯誤確認-----

RR0 → 讀取 ;
RR0/D4或是D5若為1時有錯誤發生。 錯誤解析處理。

//---補間結束確認-----

區段計數器如果是22，連續補間結束。

//---次區段資料寫入許可確認-----

RR0 → 讀取 ;
RR0/D11若為1就可以寫入下一資料。若為0則不可寫入之故再次讀取RR0。

//---次區段資料寫入-----

區段計數器所示區段資料及補間命令寫入。

//---返回 Loop-----

區段計數器為加1後跳到Loop。

3.8 加減速度驅動的補間

補間通常在等速驅動進行、本IC可以在直線加減速驅動或是S形加減速驅動(僅直線補間)中進行。

使用減速有效命令(6Dh)、減速無效命令(6Eh)，可以設定補間驅動即使是連續補間做加減速驅動。減速有效命令為補間驅動自動減速或手動減速有效的命令、減速無效命令為無效的命令。重置時預設為無效。加減速單獨的補間驅動時、驅動開始前請務必設減速有效狀態。驅動途中再寫入減速有效命令時不會有效。

3.8.1 直線補間加減速驅動

直線補間時可以做直線加減速驅動或是S形加減速驅動。而減速可以自動減速或手動減速。

手動減速的時候時、終點座標的各軸的值中絕對值最大的主軸設定手動減速點。例、主軸:X, 第2軸:Y, 第3軸:Z軸、終點(X:-20000,Y:30000,Z:-50000)的3軸直線補間、減速所需脈衝數假設為5000時、Z軸終點絕對值最大之故、所以將50000-5000=45000的手動減速點寫入主軸X軸。

直線補間加減速驅動例請參考、3.1節的3軸直線補間驅動例。

【注意】

- 短軸脈衝均一化模式下、S形加減速驅動不能使用。

3.8.2 圆弧補間、位元補間的加減速驅動

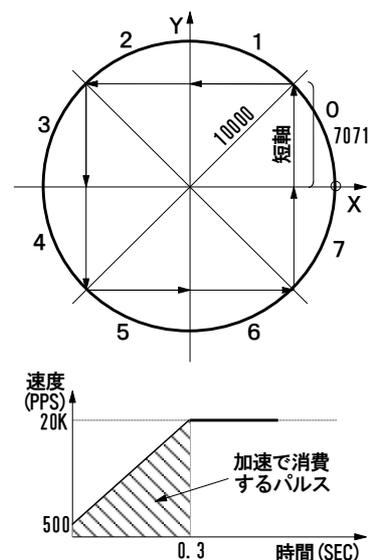
圆弧補間、位元補間手動減速止渴以在直線加減速驅動使用。S形加減速驅動及自動減速不能使用。

如右圖所示、以半徑10000的真圆弧軌跡做直線加減速驅動的描繪為例。

圆弧補間無法自動減速之故、需要設定手動減速點。

半徑10000的圆弧由0到7象限全部通過。各象限中短軸保持脈衝輸出之故、短軸側1象限有 $10000/\sqrt{2} = 7071$ 脈衝輸出。因此、主軸輸出基本脈衝數全圆弧為、 $7071 \times 8 = 56568$ 。

另外、初速度500PPS、驅動速度20000PPS止0.3秒做直線加速、加速度為 $(20000 - 500)/0.3 = 65000\text{PPS/SEC}$ 、加速時消耗的脈衝數如右下圖斜線部面積為 $(500 + 20000) \times 0.3 / 2 = 3075$ 。因此減速度與加速度如果相同的話手動減速點為 $56568 - 3075 = 53493$ 設定就可以。



【注意】

- 線速一定模式、以上計算式不成立。

```

WR0 ← 011Fh 寫入      ; X軸選擇
WR3 ← 0001h 寫入      ; 減速開始點：手動

WR6 ← 0003h 寫入      ; 補間模式設定：X, Y軸指定
WR0 ← 002Ah 寫入

WR6 ← FDE8h 寫入      ; 加速度：65000 PPS/SEC
WR7 ← 0000h 寫入
WR0 ← 0102h 寫入

WR6 ← 01F4h 寫入      ; 初速度：500 PPS
WR7 ← 0000h 寫入
WR0 ← 0104h 寫入

WR6 ← 4E20h 寫入      ; 驅動速度：20000 PPS
WR0 ← 0105h 寫入

WR6 ← D8F0h 寫入      ; 中心 X：-10000
WR7 ← FFFFh 寫入
WR0 ← 0108h 寫入

WR6 ← 0000h 寫入      ; 中心 Y：0
WR7 ← 0000h 寫入
WR0 ← 0208h 寫入

WR6 ← 0000h 寫入      ; 終點 X：0
WR7 ← 0000h 寫入
WR0 ← 0106h 寫入

```

```
WR6 ← 0000h 寫入 ; 終点 Y : 0
WR7 ← 0000h 寫入
WR0 ← 0206h 寫入

WR6 ← D0F5h 寫入 ; 手動減速点 : 53493
WR7 ← 0000h 寫入
WR0 ← 0107h 寫入

WR0 ← 006Dh 寫入 ; 減速有效
WR0 ← 0065h 寫入 ; C C W 円弧補間驅動
```

3.8.3 螺旋補間的加減速驅動

螺旋補間無法做自動減速或手動減速。

螺旋補間如果要減速時、可利用同步動作參考以下方法做。

- ① 減速開始位置變更驅動速度來減速
- ② 減速開始位置寫入減速停止命令來減速

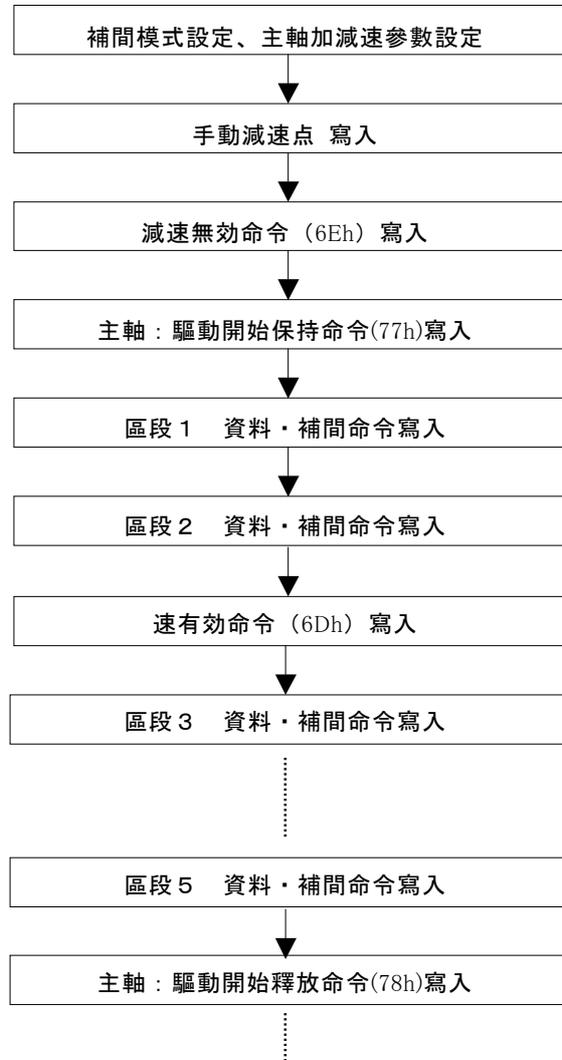
任一方法多需要先計算減速開始位置。

①可用3.8.5項的方法減速。②是起動要素下取得減速開始位置情報、動作(Action)時發出減速停止命令減速。

3.8.4 連續補間的加減速驅動

連續補間也是只可以手動減速或直線加減速驅動。S形加減速驅動也無法自動減速。連續補間需預先設定手動減速點，此手動減速點在減速開始區段的主軸輸出基本脈衝值設定。連續補間在減速開始區段的補間資料設定時、補間命令寫入前寫入減速有效命令。進入減速有效設定的區段後驅動減速有效、該區段開始計數主軸基本脈衝脈衝數，等到超過手動減速點值的時候減速開始。

例、區段1到5為止連續補間、區段3加入手動減速的時候、流程如下。



請注意此例是將手動減速點放在區段3開始的主軸基本脈衝的脈衝數的值。

3.8.5 補間驅動中變更驅動速度

補間驅動中要變更驅動速度時、以同步動作方式進行。起動要素以驅動速度變更為要素設定、當動作(Action)時「參數值載入」來變更驅動速度。同步動作起動後、變更後的驅動速度對應於加減速驅動或等速驅動執行。同步動作請參考、2.7節。

【注意】

- 連續補間途中區段要變更驅動速度時、可以用驅動速度寫入命令(05h)來變更驅動速度、除此情況外驅動速度寫入命令(05h)無法變更驅動速度。請使用本項所示的同步動作方法變更。

3.9 補間單步進給

補間驅動單步脈衝進給的方式。可以分為指令執行或是外部訊號控制的方法。使用外部訊號控制時不使用主軸基本脈衝、而可以利用外部訊號做同步補間驅動。

單步進給時、補間主軸設定為等速驅動。各軸輸出驅動脈衝的Hi為準寬依補間主軸設定的驅動速度決定，脈衝周期為1/2。Low位準寬依一下依指令或外部訊號到達時間而決定。圖3.9-1為、外部訊號做補間單步進給例。主軸初速度為500PPS、驅動速度為500PPS的等速驅動、輸出驅動脈衝的Hi寬為1mSEC。(驅動脈衝為正理論的時候)

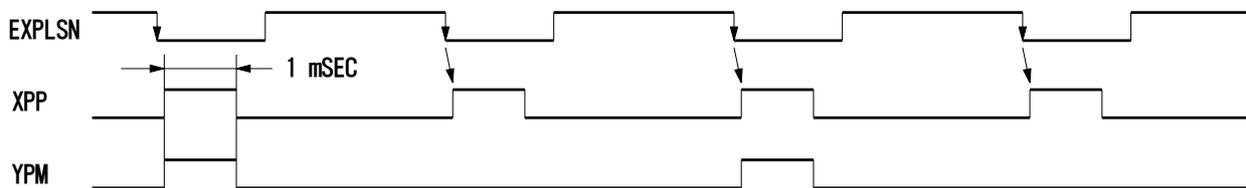


圖 3.9-1 外部信号(EXPLSN)做補間單步進給例(驅動速度:500PPS)

使用補間單步進給時、補間模式設定命令(2Ah)的D9位元寫入1即可做補間單步進給模式。



3.9.1 指令方式補間單步進給

補間驅動以單步進給指令處理時、可用補間單步(6Fh)命令。操作步驟如下。

- 補間模式設定命令(2Ah)D9位元設為1。
執行補間單步進給模式。
- 補間主軸初速度及驅動速度相同值設定。
初速度與驅動速度相同時做等速驅動。此時設定的速度值需比單步命令寫入時間快。例如、單步寫入最快需1mSEC的時間的話、初速度與驅動速度需設定高於1000PPS。
- 補間資料設定。(終点、中心点...等)
- 補間命令寫入。
補間命令寫入因指令差異，單步補間模式下、各軸驅動脈衝還不會輸出。
- 單步補(6Fh)命令寫入。
補間演算結果的驅動脈衝由各軸輸出。補間驅動結束後、再寫入單步補間(6Fh)命令。

補間單步進給途中要中止時、可對主軸寫入即停止命令(57h)、驅動速度在1脈衝周期以上的時間延遲後、再次補間單步命令(6Fh)寫入時驅動停止。

補間驅動結束後再寫入的補間單步進給命令無效。

3.9.2 外部訊號補間單步進給

EXPLSN端子(30)為補間驅動單步進給的外部輸入訊號。

EXPLSN輸入訊號通常為Hi。外部訊號補間單步模式下、Low的↓時，補間單步進給執行。

以下為操作步驟。

a. 補間模式設定(2Ah)的D9位元設為1。

補間單步模式設定。

b. 補間主軸的初速度及驅動速度設定同值。

初速度與驅動速度相同的等速驅動。此時速度值與單步進給指令處理時相同、需要比EXPLSN的Low脈衝循環速度快才可以。

c. 補間資料設定。(終點、中心點...等)

d. 補間命令寫入。

補間命令寫入因指令差異，單步補間模式下、各軸驅動脈衝還不會輸出。

e. EXPLSN Low準位的脈衝輸入。

脈衝觸發向下起2~5CLK後、補間驅動脈衝由各軸輸出。

EXPLSN的Low位準脈衝寬需要4CLK以上。另外、EXPLSN的脈衝周期要比主軸設定驅動速度的周期長才可以。

補間驅動結束為止、EXPLSN的Low為準脈衝返復執行。

補間單步進給途中要中止時、對主軸寫入即停止命令(57h)、驅動速度在1脈衝周期以上的時間延遲後、再度、EXPLSN的Low位準脈衝輸入時、驅動停止。(更快的方法是讓軟體極限動作的方法也可以。)

補間驅動結束後EXPLSN的Low脈衝輸入無效。

3.9.3 補間單步進給注意事項

補間單步進給注意事項如以下所示。

■ 外部訊號EXPLSN

EXPLSN訊號無濾波機能。EXPLSN Low脈衝如果是機械接點所產生的話、請注意突波誤動作的處理。

■ 短軸脈衝均一化

補間單步進給不能使用短軸脈衝均一化機能。

3.10 多晶片補間

本IC可使用複數晶片做多軸直線補間。

圖3.10-1為3個晶片12軸直線補間的接線例。主晶片會對副晶片送出補間驅動的同步脈衝訊號之故、因此補間驅動速度參數只要設定在主晶片的主軸即可。

如圖所示、8條多晶片補間用訊號(MPLS,MCLK,MERR,MINP,MDT3~0)連結各晶片、用(3.3KΩ左右)電阻提升。這些訊號與汎用輸入訊號(PIN7~0)共用之故、汎用輸入訊號就無法使用。

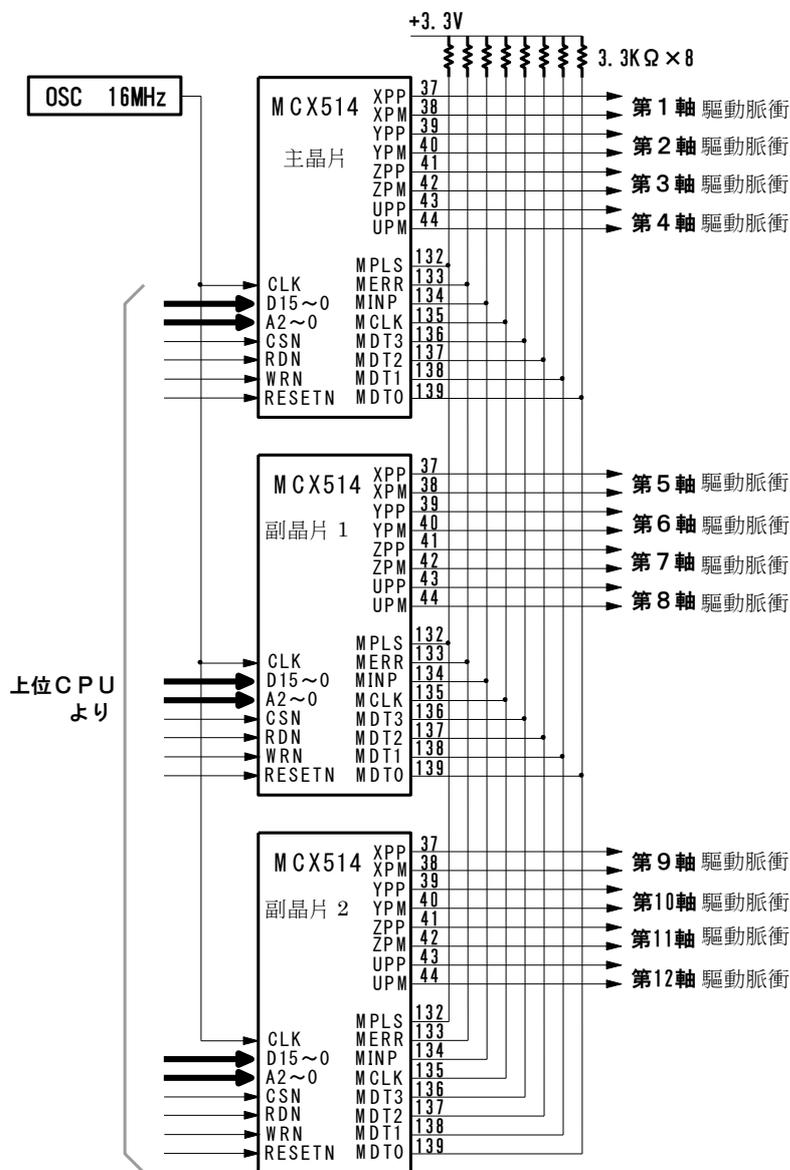


圖 3.10-1 多晶片多軸補間接線例

各訊號作用如下。

表 3.10-1 多晶片補間各訊號的作用

訊號名(端子番号)	訊號機能	通信方向	兼用汎用輸入訊號
MPLS (132)	補間驅動的同步脈衝	主→副	PIN7
MERR (133)	錯誤発生/主晶片的停止	主↔副	PIN6
MINP (134)	等待定位完成	主←副	PIN5
MCLK (135)	MD3~0 資料傳送區塊	主↔副	PIN4
MDT3~0 (136~139)	各晶片終点的資料傳送	主↔副	PIN3~0

3.10.1 執行步驟

以下為複數晶片多軸直線補間的步驟。

(1) 多晶片主・副的指定および補間軸指定

依據補間模式設定命令(2Ah) 指定主晶片・副晶片、及各晶片內補間執行軸。補間模式設定命令在 WR6 暫存器的所定位元設定後、WR0 暫存器寫入命令碼 2Ah 後執行。WR6 暫存器其他補間模式指定位元依需要設定。

WR6	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
	INTB	INTA	0	MAXM	MLT1	MLT0	STEP	LMDF	SPD1	SPD0	0	CXIV	U-EN	Z-EN	Y-EN	X-EN
	主・副晶片指定								補間軸指定							

主晶片・副晶片指定為 D10,11 位元(MLT0,1)設定。

表 3.10-2 多晶片補間的晶片指定

D11 (MLT1)	D10 (MLT0)	主・副晶片指定
0	0	不執行多晶片補間
0	1	多晶片補間之主晶片
1	0	多晶片補間之副晶片
1	1	無効 (設定不可)

各晶片內的補間軸在 WR6/D3~D0 位元指定。対応位元設 1 時補間軸有効。

表 3.10-3 多晶片補間軸指定

WR6 位元	補間軸	
D0 (X-EN)	X 軸	0 : 補間無効
D1 (Y-EN)	Y 軸	1 : 補間有効
D2 (Z-EN)	Z 軸	
D3 (U-EN)	U 軸	

多晶片補間可以只有 1 軸設定。

【注意】

- 多晶片補間、不能使用短軸脈衝均一化機能。D8 位元請務必設為 0。

(2) 主晶片的主軸驅動速度參數設定

補間驅動的速度參數設定在主晶片的主軸進行。主晶片其他補間軸及副晶片的補間軸不需設定。主晶片主軸依據加減速動作狀況請參考下表所示設定必要的速度參數。

表 3.10-4 主晶片主軸速度參數設定

加減速動作	主晶片主軸速度參數設定				
	加速度增加率	加速度	減速度	初速度	驅動速度
等速驅動				○	○
台形加減速驅動		○		○	○
非对称台形加減速驅動		○	○	○	○
S 形加減速驅動	○	○		○	○

○ 必須設定

【注意】

- 多晶片補間的驅動速度最高 4Mpps。驅動速度請設定在 4Mpps 以下。

- 多晶片多軸直線補間的全部軸當中、驅動速度為移動脈衝數最多的軸。不是主晶片的主軸設定的驅動速度。
- 台形及S形加減速驅動時、請務必在補間驅動命令發行前、務必對主晶片發出減速有效命令(6Dh)。

(3) 各軸終点的設定

主・副全部晶片的補間執行各軸、寫入的終点資料為現在值起的相對值。

多晶片補間終点資料範圍為帶符號28位元。WR6,7暫存器中寫入終点資料、WR0暫存器中軸指定連同命令碼06h寫入後設定。

一般、執行多軸直線補間時、全部軸中終点資料最大值在各軸的直線補間演算要使用到。本IC因可以做高速連續直線補間的關係、各軸終点資料設定後自動的產生最大值。不需要CPU算出最大值，所以各軸最大值不需設定。

所有晶片的軸將終点資料寫入後、由寫入的晶片起、透過多晶片補間用訊號(MCLK,MDT3~0)、傳送到其他晶片做終点資料最大值的絕對值比較、當值較大時就會更新終点最大值。

終点資料傳送時間約需要 $2\mu\text{sec}$ (CLK=16MHz時)必要。因此、各軸終点資料寫入間隔需比此時間短才可以。高速演算CPU在終点資料寫入循環下若彼此時間快的時候需要做軟體上的遲延。

終点最大值在重置及補間驅動命令開始瞬間清為0。或、終点最大值清除命令(7Ch)也可以清除。或是終点最大值、可以終点最大值讀取命令(39h)讀取之故、全軸終点資料寫入後、是否產生正確的最大值可以做確認。

【注意】

- 終点最大值讀取命令(39h)讀取值時、補間驅動執行前即執行中的內容不同。終点最大值讀取命令(39h)的詳細請參考、7.4.10 項。

■ 終点資料傳送錯誤

終点資料受信側的各晶片、資料接收及傳送會檢查是否異常。若接收不正常則是為錯誤、RR2暫存器的D7位元(CERR)及RR3暫存器第一頁的D12位元(MCERR)變為1。另外、RR0暫存器的補間軸全部錯誤位元(D7~4:n-ERR的該位元)變為1。副晶片受信錯誤發生時、訊號會經由多晶片補間用訊號(MERR)將錯誤傳到主晶片、主晶片RR0暫存器的主軸錯誤位元變為1。

(4) 補間命令寫入

針對每個副晶片、寫入對應的補間軸數的直線補間命令(60h~63h)。接著對主晶片對應的補間軸數寫入直線補間命令(60h~63h)。寫入順序若主晶片比副晶片先的話動作會不正常。加減速驅動時、務必在補間驅動命令發行前對主晶片寫入減速有效命令(6Dh)。

表 3.10-5 多晶片補間命令

補間命令	命令碼
1 軸直線驅動	60h
2 軸直線補間驅動	61h
3 軸直線補間驅動	62h
4 軸直線補間驅動	63h

主晶片在直線補間命令(60h~63h)寫入後、主晶片立刻送出MPLS訊號到各副晶片，補間驅動同步脈衝開始送出後、全軸直線補間開始。

(5) 驅動結束、錯誤確認

補間驅動中各晶片RR0暫存器的補間有效軸驅動位元(D3~0:n-DRV)變為1。補間驅動結束後主晶片主軸的驅動位元回到0為判斷依據。

各軸定位完成若設為有效時、等到所有軸的nINPOS訊號有效後主晶片RR0暫存器的主軸驅動位元(D3~0:n-DRV的主軸的位元)回到0。

補間驅動中、主晶片各軸發生錯誤時、各軸RR2暫存器的D5~0位元會變為1、RR0暫存器錯誤位元(該軸的D7~4:n-ERR)變為1。

另外、副晶片各軸發生錯誤時、與主晶片相同各軸RR2暫存器D5~0位元變為1、RR0暫存器錯誤位元(該軸的D7~4:n-ERR)變為1。另外、副晶片的補間用訊號MERR訊號變為Low位準，錯誤發生傳到主晶片。主晶片接收到錯誤後、RR0暫存器的主軸錯誤位元(該軸D7~4:n-ERR)變為1。主晶片在錯誤發生時停止對副晶片送出補間驅動同步脈衝之故、全軸立即

停止。

因此、補間驅動中及補間結束時的錯誤的確認、僅監視主晶片主軸的RR0暫存器錯誤位元(D7~4:n-ERR)就可以。檢查有錯誤(位元資料 =1)時、再確認補間實施的各軸RR2暫存器(錯誤顯示暫存器)內容後，做錯誤原因解析處理。

3.10.2 補間驅動途中停止

補間驅動途中要停止時、對主晶片的主軸發出驅動停止命令即可。主軸驅動停止後、主晶片其他補間實施軸及副晶片的補間實施軸驅動回同時停止、RR0暫存器的驅動位元(D3~0:n-DRV的該當位元)回到0。

3.10.3 連續補間

多晶片補間也可以做直線補間的連續補間。連續補間執行方法與單一晶片的連續補間相同。驅動開始保持、驅動開始釋放命令對主晶片的主軸寫入。主晶片的預存緩衝暫存器空的話、全部補間軸可以寫入終點資料。副晶片預存緩衝暫存器的空狀態與主晶片同時增減。

3.10.4 多晶片補間注意事項

多晶片補間注意事項如以下所示。

- 多晶片補間用訊號(MPLS,MCLK,MERR,MINP,MDT3~0)需用電阻將 3.3V 電源提升。阻值範圍在 1K~5.1K Ω 、推薦使用 3.3K Ω 左右。
- 多晶片補間用訊號(MPLS,MCLK,MERR,MINP,MDT3~0)的的線路盡可能縮短距離、不要與其他訊號交叉。回路系統上避免用短路套切換方式做汎用輸入訊號兼用設計。
- 多晶片補間僅主晶片的軸可以做線速一定。
- 連續補間使用時定為完成訊號請設為無效。

3.10.5 多晶片補間的實施例

主晶片與2個副晶片的實施例。

■ 例 1 各晶片 X 軸,Y 軸的 2 軸做多晶片補間

【程式例】

```
// 主晶片、副晶片的補間模式設定
//寫入主晶片
WR6 ← 0403h 寫入           // 主晶片 X,Y 補間軸指定
WRO ← 002Ah 寫入
//寫入副晶片 1
WR6 ← 0803h 寫入           // 副晶片 X,Y 補間軸指定
WRO ← 002Ah 寫入
//寫入副晶片 2
WR6 ← 0803h 寫入           // 副晶片 X,Y 補間軸指定
WRO ← 002Ah 寫入

// 主晶片主軸、驅動関連參數設定(設定 2M PPS 等速驅動)
WR6 ← 1200h 寫入           // 初速度 8M PPS (規格最大)
WR7 ← 007Ah 寫入
WRO ← 0104h 寫入

WR6 ← 8480h 寫入           // 驅動速度 2M PPS
WR7 ← 001Eh 寫入
WRO ← 0105h 寫入

//寫入終點資料及檢查受信錯誤
//寫入主晶片
WR6 ← 0014h 寫入           // 終點 1 X 20
WR7 ← 0000h 寫入
WRO ← 0106h 寫入
// 檢查副晶片 1 的受信錯誤           A 處理
RRO / D4, D5 讀取
D4, D5=1 時跳到 ERROR 處理(副晶片) // 到錯誤處理
// 檢查副晶片 2 受信錯誤           B 處理
RRO / D4, D5 讀取
D4, D5=1 時跳到 ERROR 處理(副晶片) // 到錯誤處理
WR6 ← 000Ah 寫入           // 終點 1 Y 10
WR7 ← 0000h 寫入
WRO ← 0206h 寫入
// 執行 A 處理
// 執行 B 處理

//寫入副晶片 1
WR6 ← FFF6h 寫入           // 終點 1 X -10
WR7 ← FFFh 寫入
WRO ← 0106h 寫入
// 檢查主晶片的受信錯誤           C 處理
RRO / D4, D5 讀取
D4, D5=1 時跳到 ERROR 處理(主晶片) // 到錯誤處理
// 執行 B 處理
WR6 ← 0005h 寫入           // 終點 1 Y 5
WR7 ← 0000h 寫入
WRO ← 0206h 寫入
// 執行 C 處理
// 執行 B 處理

// 寫入副晶片 2
WR6 ← 0019h 寫入           // 終點 1 X 25
WR7 ← 0000h 寫入
WRO ← 0106h 寫入
// 執行 C 處理
// 執行 A 處理
WR6 ← FFF4h 寫入           // 終點 1 Y -12
WR7 ← FFFh 寫入
WRO ← 0206h 寫入
// 執行 C 處理
// 執行 A 處理

// 依副晶片、主晶片的順序發出補間命令
// 寫入副晶片 1 的
WRO ← 0061h 寫入           // 2 軸直線補間
//寫入副晶片 2
WRO ← 0061h 寫入           // 2 軸直線補間
// 寫入主晶片
WRO ← 0061h 寫入           // 2 軸直線補間
```

// ERROR 處理 (主晶片)

WR0 ← 011Fh 寫入
 RR0 / D4, 5 讀取
 RR2 / D7 讀取
 WR0 ← 017Bh 寫入
 RR3 / D12 讀取

WR0 ← 0179h 寫入
 RR0 / D4, 5 讀取
 RR2 / D7 讀取
 RR3 / D12 讀取

// ERROR 處理 (副晶片)

WR0 ← 011Fh 寫入
 RR0 / D4, 5 讀取
 RR2 / D7 讀取
 WR0 ← 017Bh 寫入
 RR3 / D12 讀取
 //讀取主晶片 RR0 暫存器
 RR0 / D4 讀取

WR0 ← 0179h 寫入
 RR0 / D4, 5 讀取
 RR2 / D7 讀取
 RR3 / D12 讀取
 // 讀取主晶片的 RR0 暫存器
 RR0 / D4 讀取

// 對補間軸任一軸 (例如 X 軸)、RR2 暫存器的補間錯誤確認
 // 補間軸錯誤確認
 // 補間錯誤確認
 // 對補間軸任一軸 (例如 X 軸)、RR3 的第一頁顯示命令發行
 // 多晶片補間傳送錯誤確認

// 對補間軸發出錯誤清除命令
 // 確認補間軸的錯誤清除
 // 補間錯誤清除確認
 // 多晶片補間傳送錯誤清除確認

// 對補間軸任一軸 (例如 X 軸)、RR2 暫存器的補間錯誤確認
 // 補間軸錯誤確認
 // 補間錯誤確認
 // 對補間軸任一軸 (例如 X 軸)、RR3 的第一頁顯示命令發行
 // 多晶片補間傳送錯誤確認

// 主晶片的主軸錯誤確認

// 對補間軸發出錯誤清除命令
 // 確認補間軸的錯誤清除
 // 補間錯誤清除確認
 // 多晶片補間傳送錯誤清除確認

// 主晶片的主軸錯誤清除確認

■ 例 2 多晶片補間的連續補間

下例A、B、C處理與ERROR處理焊例1相同。

【程式例】

```
// 主晶片、副晶片補間模式設定
// 寫入主晶片
WR6 ← 0403h 寫入          // 主晶片 X,Y 補間軸指定
WRO ← 002Ah 寫入
// 寫入副晶片 1
WR6 ← 0803h 寫入          // 副晶片 X,Y 補間軸指定
WRO ← 002Ah 寫入
// 寫入副晶片 2
WR6 ← 0803h 寫入          // 副晶片 X,Y 補間軸指定
WRO ← 002Ah 寫入

// 對主晶片主軸設定驅動関連參數(2M PPS 等速驅動)
WR6 ← 1200h 寫入          // 初速度 8M PPS (規格最大)
WR7 ← 007Ah 寫入
WRO ← 0104h 寫入

WR6 ← 8480h 寫入          // 驅動速度 2M PPS
WR7 ← 001Eh 寫入
WRO ← 0105h 寫入

//對主晶片主軸、発行驅動開始保持命令
WRO ← 0177h 寫入

// 終点資料寫入及檢査受信錯誤
// Seg1
// 寫入主晶片
WR6 ← 0014h 寫入          // 終点 1 X 20
WR7 ← 0000h 寫入
WRO ← 0106h 寫入
// 執行A 處理
// 執行B 處理
WR6 ← 000Ah 寫入          // 終点 1 Y 10
WR7 ← 0000h 寫入
WRO ← 0206h 寫入
// 執行A 處理
// 執行B 處理

// 寫入副晶片 1
WR6 ← FFF6h 寫入          // 終点 1 X -10
WR7 ← FFFFh 寫入
WRO ← 0106h 寫入
// 執行C 處理
// 執行B 處理
WR6 ← 0005h 寫入          // 終点 1 Y 5
WR7 ← 0000h 寫入
WRO ← 0206h 寫入
// 執行C 處理
// 執行B 處理

// 寫入副晶片 2
WR6 ← 0019h 寫入          // 終点 1 X 25
WR7 ← 0000h 寫入
WRO ← 0106h 寫入
// 執行C 處理
// 執行A 處理
WR6 ← FFF4h 寫入          // 終点 1 Y -12
WR7 ← FFFFh 寫入
WRO ← 0206h 寫入
// 執行C 處理
// 執行A 處理

// 依副晶片、主晶片的順序寫入補間命令
// 寫入副晶片 1
WRO ← 0061h 寫入          // 2 軸直線補間
// 寫入副晶片 2
WRO ← 0061h 寫入          // 2 軸直線補間
// 寫入主晶片
WRO ← 0061h 寫入          // 2 軸直線補間
```

```

// Seg2
// 寫入主晶片
WR6 ← 000Ah 寫入           // 終点 1 X 10
WR7 ← 0000h 寫入
WRO ← 0106h 寫入
// 執行 A 處理
// 執行 B 處理
WR6 ← 0014h 寫入           // 終点 1 Y 20
WR7 ← 0000h 寫入
WRO ← 0206h 寫入
// 執行 A 處理
// 執行 B 處理

// 寫入副晶片 1
WR6 ← 0005h 寫入           // 終点 1 X 5
WR7 ← 0000h 寫入
WRO ← 0106h 寫入
// 執行 C 處理
// 執行 B 處理
WR6 ← 000Ah 寫入           // 終点 1 Y 10
WR7 ← 0000h 寫入
WRO ← 0206h 寫入
// 執行 C 處理
// 執行 B 處理

// 寫入副晶片 2
WR6 ← 0019h 寫入           // 終点 1 X 25
WR7 ← 0000h 寫入
WRO ← 0106h 寫入
// 執行 C 處理
// 執行 A 處理
WR6 ← 000Ch 寫入           // 終点 1 Y 12
WR7 ← 0000h 寫入
WRO ← 0206h 寫入
// 執行 C 處理
// 執行 A 處理

// 依副晶片、主晶片的順序寫入補間命令
// 寫入副晶片 1
WRO ← 0061h 寫入           // 2 軸直線補間
// 寫入副晶片 2
WRO ← 0061h 寫入           // 2 軸直線補間
// 寫入主晶片
WRO ← 0061h 寫入           // 2 軸直線補間

// 以下、最大 Seg8 為止必要的數重複
.
.
.

// 對主晶片主軸發出驅動開始釋放命令
WRO ← 0178h 寫入           // 連續補間驅動開始

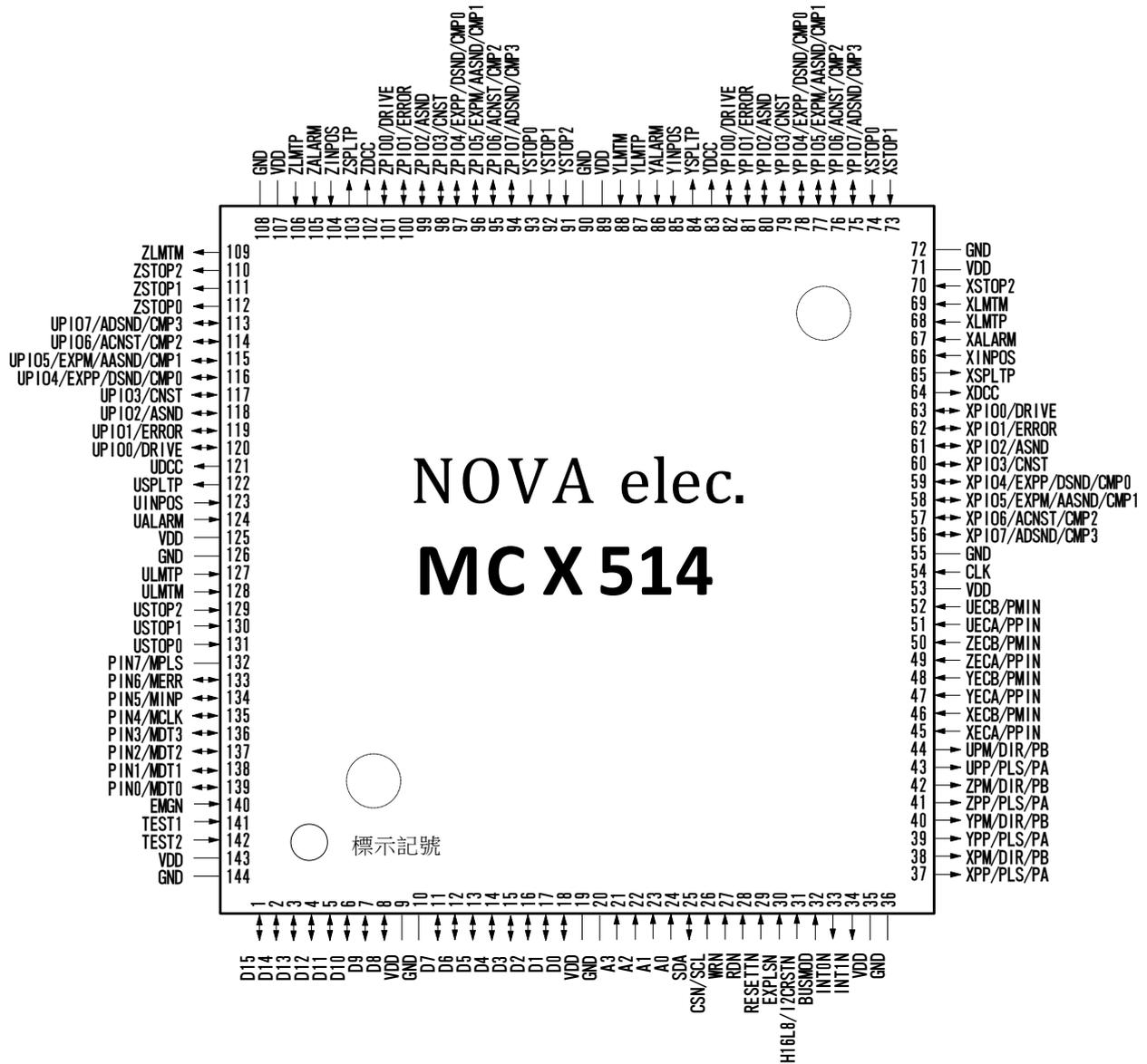
```

4. I²C 串列介面匯流排機能

現在作成中です。

5. 端子配置及各訊號說明

5.1 端子配置



144PIN 塑膠 QFP

封裝外形 20×20mm, 最外形 22×22mm, 端子腳距 0.5mm
外形寸法在第 10 章有記載。

5.2 各訊號說明

輸出入回路請參考5.3節說明。有－F－記號的地方其輸入訊號為IC內部輸入段有積分濾波回路。

訊號名	端子番号	輸出入回路	訊號的說明
CLK	54	輸入 A	Clock : 提供本 I C 內部同步迴路動作的時脈訊號。輸入時脈為周波數 16.000MHz。驅動速度、加／減速度、加／減速度增加率與此時脈的周波數相依存。輸入 16MHz 以外的周波數的情況時速度設定值、加減速設定值會有所不同。
D15~D0	1~8, 11~18	雙方向 A	資料匯流排 (D15~D0) : 3 狀態雙方向的 16 位元資料匯流排。與系統資料匯流排連接。在 CSN=Low 且 RDN=Low 的時候變成輸出狀態。除此以外高阻抗時候為輸入狀態。 使用 8 位元資料匯流排的情況、不使用 D15~D8 時、請以高阻抗 (10k~100k Ω 左右) 電阻連接 VDD 或是 GND。
A3~A0	21~24	輸入 A	Address : CPU 選擇對本 I C 讀／寫暫存器時的位址訊號。 資料匯流排為 16 位元的時候、因 A3 不使用、請與 GND 連接。 I ² C 模式下、A2~A0 為晶片選擇設定端子使用。
SDA	25	雙方向 D	I ² CSDA : I ² C 模式時 SDA 訊號。
CSN/SCL	26	輸入 A	Chip Select / I ² C SCL : 選擇本 I C 為 I / O 周邊之輸入訊號。對本 I C 讀／寫時、訊號為 Low 位準。 I ² C 模式下做為 SCL 訊號使用。
WRN	27	輸入 A	Write Strobe : 本 I C 要對寫入暫存器做寫入動作時使其為 Low。WRN 在 Low 的期間下 CSN 及 A3~A0 需先確定。WRN↑ 的時候、資料匯流排的內容的栓鎖暫存器會鎖住的關係、WRN↑ 的前後 D15~D0 的值需先確定。
RDN	28	輸入 A	Read Strobe : 本 I C 要從讀取暫存器讀取資料時使其為 Low。CSN 在 Low 的時候 RDN 將 Low 時、只有在 RDN 為 Low 的這段時間、由 A3~A0 的位址訊號所選擇的讀取暫存器的資料在資料匯流排輸出。
RESETN	29	輸入 A	重置 : 本 I C 的重置 (初始化) 訊號。CLK 為 8 個循環以上的時間下將 RESETN 置為 Low 時就可以重置。電源投入時時、請務必重置接本 I C 的 RESETN 訊號重置。 【注意】CLK 入沒有輸入時即使將 RESETN 置於 Low 也不回會重置。
EXPLSN	30	輸入 B	External Pulse : 補間驅動外部訊號單步進給模式時的脈衝輸入。補間驅動外部訊號單步進給模式的補間驅動為、EXPLSN 在 ↓ 時候 1 脈衝分的補間演算起動、各軸補間脈衝以 1 脈衝輸出。 EXPLSN 的 Low 位準寬、最小需要 4 CLK 以上。 【注意】EXPLSN 無濾波機能
H16L8 /I2CRSTN	31	輸入 B	Hi=16 位元、Low=8 位元 : 選擇 16 位元資料匯流排 / 8 位元資料匯流排。 Hi 位準時為 16 位元資料匯流排、I C 內的讀寫暫存器以 16 位元存取。 若為、Low 位準時、資料匯流排僅 D7~D0 的 8 位元有效、內部讀／寫暫存器以 8 位元存取。 I ² C 模式下 I ² C 做為重置使用。訊號 Low 時、IC 內部僅 I ² C 控制部重置。
BUSMOD	32	輸入 B	Bus Mode : 選擇 CPU 匯流排連接。Hi 位準時 16 位元 / 8 位元並列資料匯流排連接。Low 位準時、I ² C 串列介面匯流排連接。
INTON	33	輸出 B	Interrupt : 對上位 CPU 中斷要求訊號。補間緩衝暫存器以外的各中斷要素發生中斷時 INTON 變為 Low 位準。中斷解除後、回到 Hi-Z。
INT1N	34	輸出 B	Interrupt : 對上位 CPU 中斷要求訊號。補間緩衝暫存器發生的中斷要素 INT1N 變為 Low 位準。中斷解除後、回到 Hi-Z。
XPP/PLS/PA YPP/PLS/PA ZPP/PLS/PA UPP/PLS/PA	37 39 41 43	輸出 A	Pulse + / Pulse / Pulse Phase A : 輸出 + 方向的驅動脈波。重置時的狀態變為 Low 位準、進入驅動動作時、輸出工作週期 50% (等速時) 的正脈波。 模式選擇為 1 脈波方式的時候、由本端子輸出驅動脈波。 模式選擇為 2 相脈波方式的時候、由本端子輸出 A 相訊號。
XPM/DIR/PB YPM/DIR/PB ZPM/DIR/PB UPM/DIR/PB	38 40 42 44	輸出 A	Pulse - / Direction / Pulse Phase B : 輸出一方向的驅動脈波。重置時的狀態變為 Low 位準、進入驅動動作時、輸出工作週期 50% (等速時) 的正脈波。 模式選擇為 1 脈波方式的時候、由本端子輸出方向訊號。 模式選擇為 2 相脈波方式的時候、由本端子輸出 B 相訊號。

訊號名	端子番号	輸出入回路	訊號的說明
XECA/PPIN YECA/PPIN ZECA/PPIN UECA/PPIN	45 47 49 51	輸入 B — F —	Encoder-A / Pulse+in : 編碼器 A 相訊號的輸入。B 相訊號也一樣、I C 內部轉換為上下脈波、變為實際位置計數器的計數輸入。 模式選擇為上下脈波輸入時、本端子變為上脈波輸入、輸入脈波↑的時候、實際位置計數器的計數遞增。
XECB/PMIN YECB/PMIN ZECB/PMIN UECB/PMIN	46 48 50 52	輸入 B — F —	Encoder-B / Pulse-in : 編碼器 B 相訊號的輸入。A 相訊號也一樣、I C 內部轉換為上下脈波、變為實際位置計數器的計數輸入。 模式選擇為上下脈波輸入時、本端子變為下脈波輸入、輸入脈波↑的時候、實際位置計數器的計數遞減。
XSTOP2~0 YSTOP2~0 ZSTOP2~0 USTOP2~0	70, 73, 74 91, 92, 93 110, 111, 112 129, 130, 131	輸入 B — F —	Stop2~0 : 驅動途中做減速停止或是立即停止的輸入訊號。做為搜尋動作的輸入訊號使用。濾波機能無效的時候、需 2CLK 以上的動作脈波寬。STOP2~STOP0 均可以設定其有效/無效、理論位準。 自動原點復歸時、可分配 STOP0 當原點近傍訊號、STOP1 當原點訊號、STOP2 當編碼器 Z 相訊號。 另、這些訊號狀態隨時可以由 RR3 暫存器第 0 頁讀取。
XLMTMP YLMTMP ZLMTMP ULMTMP	68 87 106 127	輸入 B — F —	Over Run Limit + : + 方向的過行程極限訊號。+ 方向的驅動脈波輸出中、此訊號若動作時驅動會減速停止或是立即停止。濾波機能若無效的時候、需 2CLK 以上的動作脈波寬。有效/無效、減速停止/立即停止、理論位準可由模式選擇。 極限訊號設為有效時、+ 方向驅動中此訊號若變為動作位準時、RR2 暫存器的 HLMT+位元變為 1。 此訊號狀態可隨時由 RR3 暫存器第 0 頁讀取。 另、此訊號可分配給自動原點復歸的檢出訊號。
XLMTM YLMTM ZLMTM ULMTM	69 88 109 128	輸入 B — F —	Over Run Limit - : - 方向的過行程極限訊號。- 方向的驅動脈波輸出中、此訊號若動作時驅動會減速停止或是立即停止。濾波機能若無效的時候、需 2CLK 以上的動作脈波寬。有效/無效、減速停止/立即停止、理論位準可由模式選擇。 極限訊號設為有效時、+ 方向驅動中此訊號若變為動作位準時、RR2 暫存器的 HLMT-位元變為 1。 此訊號狀態可隨時由 RR3 暫存器第 0 頁讀取。 另、此訊號可分配給自動原點復歸的檢出訊號。
XINPOS YINPOS ZINPOS UINPOS	66 85 104 123	輸入 B — F —	Inposition : 與伺服馬達驅動器定位完成輸出對應的輸入訊號。有效/無效、理論位準可由模式選擇。設為有效時、驅動結束後、等到此訊號動作為止、主狀態暫存器的 DRIVE 位元才會變為 0。 此訊號狀態可由 RR3 暫存器第 0 頁隨時讀取。
XALARM YALARM ZALARM UALARM	67 86 105 124	輸入 B — F —	Servo Alarm : 與伺服馬達驅動器的異常輸出對應的輸入訊號。有效/無效、理論位準可由模式選擇。設為有效時、驅動中此訊號的動作位準變化時 RR2 暫存器的 ALARM 位元變為 1、驅動會立即停止。 此訊號狀態可由 RR3 暫存器第 0 頁隨時讀取。
XPI07/ADSND/CMP3 YPI07/ADSND/CMP3 ZPI07/ADSND/CMP3 UPI07/ADSND/CMP3	56 75 94 113	雙方向 B — F —	Universal Input Output7 / Acceleration Descend / Compare MR3 : 汎用輸出入訊號 (PI07)、加速度減少狀態輸出訊號 (ADSND)、MR3 比較輸出 (CMP3) 的共用端子。使用的訊號可由模式選擇。 汎用輸入訊號 (PI07) 的狀態、XPI07, YPI07 可由 RR4 暫存器、ZPI07, UPI07 可由 RR5 暫存器隨時讀取。 汎用輸出訊號 (PI07) XPI07, YPI07 可由 WR4 暫存器、ZPI07, UPI07 可由 WR5 暫存器寫入 1/0 資料設定其 Hi/Low。 同步動作可做為起動要素的輸入訊號使用。 加速度減少狀態輸出 (ADSND) 時、驅動命令執行中、加速度減少狀態時 Hi。 MR3 比較輸出 (CMP3) 為多目的暫存器 MR3 的比較條件滿足時 Hi。
XPI06/ACNST/CMP2 YPI06/ACNST/CMP2 ZPI06/ACNST/CMP2 UPI06/ACNST/CMP2	57 76 95 114	雙方向 B — F —	Universal Input Output6 / Acceleration Constant / Compare MR2 : 汎用輸出入訊號 (PI06)、加速度一定狀態輸出訊號 (ACNST)、MR2 比較輸出 (CMP2) 的共用端子。使用的訊號可由模式選擇。 有關汎用輸出入訊號 (PI06)、與 PI07 相同。 同步動作可做為起動要素的輸入訊號使用。 加速度減少狀態輸出 (ACNST) 時、驅動命令執行中、加速度減少狀態時 Hi。 MR2 比較輸出 (CMP2) 為多目的暫存器 MR3 的比較條件滿足時 Hi。

訊號名	端子番号	輸出入回路	訊號的說明
XPI05/EXPM/AASND/CMP1 YPI05/EXPM/AASND/CMP1 ZPI05/EXPM/AASND/CMP1 UPI05/EXPM/AASND/CMP1	58 77 96 115	雙方向 B — F —	Universal Input Output5 / External Operation- / Acceleration Ascend / Compare MR1 : 汎用輸出入訊號 (PI05)、外部操作輸入 (EXPM)、加速度增加狀態輸出訊號 (AASND)、MR1 比較輸出 (CMP1) 的共用端子。 使用的訊號可由模式選擇。 有關汎用輸出入訊號 (PI05)、與 PI07 相同。 同步動作可做為起動要素的輸入訊號使用。 外部操作輸入 (EXPM) 為由外部起動一方向驅動的訊號。外部位置驅動模式下、本訊號↓時起動一方向的相對位置驅動。外部連續脈衝驅動模式下、本訊號在 Low 位準時、執行連續一方向連續脈衝驅動。手動脈衝口生器模式的時候下、本端子連接編碼器的 B 相訊號輸入。 加速度增加狀態輸出 (AASND) 為、驅動命令執行中、加速度增加狀態下 Hi。 MR1 比較輸出 (CMP1) 為多目的暫存器 MR1 的比較條件滿足時 Hi。
XPI04/EXPP/DSND/CMPO YPI04/EXPP/DSND/CMPO ZPI04/EXPP/DSND/CMPO UPI04/EXPP/DSND/CMPO	59 78 97 116	雙方向 B — F —	Universal Input Output4 / External Operation+ / Descend / Compare MRO : 汎用輸出入訊號 (PI04)、外部操作輸入 (EXPP)、減速狀態輸出訊號 (DSND)、MRO 比較輸出 (CMPO) 的共用端子。 使用的訊號可由模式選擇。 有關汎用輸出入訊號 (PI04)、與 PI07 相同。 同步動作可做為起動要素的輸入訊號使用。 外部操作輸入 (EXPM) 為由外部起動+方向驅動的訊號。外部位置驅動模式下、本訊號↓時起動+方向的相對位置驅動。外部連續脈衝驅動模式下、本訊號在 Low 位準時、執行連續+方向連續脈衝驅動。手動脈衝口生器模式的時候下、本端子連接編碼器的 A 相訊號輸入。 減速狀態輸出訊號 (DSND) 為、驅動命令執行中、減速狀態下 Hi。 MRO 比較輸出 (CMPO) 為多目的暫存器 MRO 的比較條件滿足時 Hi。
XPI03/CNST YPI03/CNST ZPI03/CNST UPI03/CNST	60 79 98 117	雙方向 B — F —	Universal Input Output3 / Constant : 汎用輸出入訊號 (PI03)、等速狀態輸出訊號 (CNST) 的共用端子。 使用的訊號可由模式選擇。 有關汎用輸出入訊號 (PI03)、與 PI07 相同。 同步動作可做為起動要素的輸入訊號使用。或動作的同步脈衝輸出訊號使用。同步脈衝的理論位準、可由脈衝寬的模式選擇。 等速狀態輸出訊號 (CNST) 在驅動命令執行中、等速狀態下 Hi。
XPI02/ASND YPI02/ASND ZPI02/ASND UPI02/ASND	61 80 99 118	雙方向 B — F —	Universal Input Output2 / Ascend : 汎用輸出入訊號 (PI02)、加速狀態輸出訊號 (ASND) 的共用端子。 使用的訊號可由模式選擇。 有關汎用輸出入訊號 (PI02)、與 PI07 相同。 有關同步動作與 PI03 相同。 加速狀態輸出訊號 (ASND) 是、驅動命令執行中、加速狀態時 Hi。
XPI01/ERROR YPI01/ERROR ZPI01/ERROR UPI01/ERROR	62 81 100 119	雙方向 B — F —	Universal Input Output1 / Error : 汎用輸出入訊號 (PI01)、錯誤狀態輸出訊號 (ERROR) 的共用端子。 使用的訊號可由模式選擇。 有關汎用輸出入訊號 (PI01)、與 PI07 相同。 有關同步動作與 PI03 相同。 錯誤狀態輸出 (ERROR) 在錯誤發生期間、Hi。
XPI00/DRIVE YPI00/DRIVE ZPI00/DRIVE UPI00/DRIVE	63 82 101 120	雙方向 B — F —	Universal Input Output0 / Drive : 汎用輸出入訊號 (PI00)、驅動狀態輸出訊號 (DRIVE) 的共用端子。 使用的訊號可由模式選擇。 汎用輸出入訊號 (PI00)、與 PI07 相同。 有關同步動作與 PI03 相同。 驅動狀態顯示輸出 (DRIVE) 為驅動脈衝輸出期間、Hi。自動原點復歸執行時、執行間、本訊號 Hi。模式選擇下、伺服馬達用的 INPOS 訊號設為有效時、INPOS 訊號動作為止、DRIVE 訊號 Hi。
XDCC YDCC ZDCC UDCC	64 83 102 121	輸出 A	Deviation Counter Clear : 偏差計數器清除輸出訊號。針對伺服馬達驅動器使用的輸出訊號。可以自動原點復歸模式設定來設定輸出。另、也可以用指令的方式使其輸出。
XSPLTP YSPLTP ZSPLTP USPLTP	65 84 103 122	輸出 A	Split Pulse : 分割脈衝輸出。分割脈衝輸出開始/停止、可由同步動作及指令操作。 分割脈衝長度、脈衝寬度、脈衝數可由指令設定。 另外、輸出理論、開始脈衝有/沒有可以用模式選擇。
PIN7/MPLS	132	雙方向 C	Universal Input7/ : 汎用輸入訊號。可以用汎用輸入值讀取命令 (48h) 取得輸入值。Low 位準為 0、Hi 位準為 1。 多晶片多軸補間執行時、此訊號在晶片間連接、3.3kΩ 的電阻對 VDD (+3.3V) 提升。

訊號名	端子番号	輸出入回路	訊號的說明
PIN6/MERR	133	雙方向 E	Universal Input6/ : 汎用輸入訊號。讀取方式與PIN7相同。 多晶片多軸補間執行時、此訊號在晶片間連接、3.3kΩ的電阻對VDD(+3.3V)提升。
PIN5/MINP	134	雙方向 E	Universal Input5/ : 汎用輸入訊號。讀取方式與PIN7相同。 多晶片多軸補間執行時、此訊號在晶片間連接、3.3kΩ的電阻對VDD(+3.3V)提升。
PIN4/MCLK	135	雙方向 C	Universal Input4/ : 汎用輸入訊號。讀取方式與PIN7相同。 多晶片多軸補間執行時、此訊號在晶片間連接、3.3kΩ的電阻對VDD(+3.3V)提升。
PIN3/MDT3	136	雙方向 C	Universal Input3/ : 汎用輸入訊號。讀取方式與PIN7相同。 多晶片多軸補間執行時、此訊號在晶片間連接、3.3kΩ的電阻對VDD(+3.3V)提升。
PIN2/MDT2	137	雙方向 C	Universal Input2/ : 汎用輸入訊號。讀取方式與PIN7相同。 多晶片多軸補間執行時、此訊號在晶片間連接、3.3kΩ的電阻對VDD(+3.3V)提升。
PIN1/MDT1	138	雙方向 C	Universal Input1/ : 汎用輸入訊號。讀取方式與PIN7相同。 多晶片多軸補間執行時、此訊號在晶片間連接、3.3kΩ的電阻對VDD(+3.3V)提升。
PIN0/MDT0	139	雙方向 C	Universal Input0/ : 汎用輸入訊號。讀取方式與PIN7相同。 多晶片多軸補間執行時、此訊號在晶片間連接、3.3kΩ的電阻對VDD(+3.3V)提升。
EMGN	140	輸入 B — F —	Emergency Stop : 驅動要做緊急停止的時候的輸入訊號。驅動中本訊號若為 Low 位準時驅動會立即停止、RR2 暫存器的 EMG 位元變為 1。濾波機能無效的情況、需 2CLK 以上的 Low 位準脈波寬。 【注意】此訊號不能選擇理論位準。
TEST1 TEST2	141, 142	—	Test : 內部回路動作輸入端子。Hi 的時候、IC 內部的測試回路動作。 兩端子請務必空口或與 GND 連接。 IC 內部用 50kΩ 與 GND 接在一起。
GND	10, 20, 36, 55, 72, 90, 108, 126, 144		接地 (0V) 端子。請務必全部端子連接。
VDD	10, 19, 35, 53, 71, 89, 107, 125, 143		電源端子。請供給 +3.3V。請務必全部端子連接。

5.3 輸出入回路

輸入 A	LVTTTL 位準的史密特觸發電路輸入。I C 內部為高阻抗迴路。 本輸入為、可以接受 5V 直接輸入的接口。3.3V 系輸出、及 5V 系輸出 (CMOS 位準、TTL 位準) 的任何輸出多可以連接。 不使用的時候、請務必連接 GND 或是 VDD。 有一 F 一記号的訊號是、本 I C 內部具有積分濾波迴路的輸入。
輸入 B	LVTTTL 位準的史密特觸發電路輸入。I C 內部以 50K Ω 做提升。 本輸入為、可以接受 5V 直接輸入的接口。3.3V 系輸出、及 5V 系輸出 (CMOS 位準、TTL 位準) 的任何輸出多可以連接。 不使用的時候、請務必空口或是與 VDD 連接。 有一 F 一記号的訊號、本 I C 內部輸入段有積分濾波回路。
輸出 A	3.3V 系 CMOS 位準的輸出。6mA 驅動緩衝 (Hi 位準輸出電流 IOH=-6mA 時 VOH=2.6Vmin、Low 位準輸出電流 IOL=6mA 時 VOL=0.4Vmax)。 Hi 位準輸出時時請不要在外掛超過輸出電壓以上的電壓。 5V 系雙方向 IC 之連接時、對接輸入若為 TTL 位準時可以連接。對接輸入為 5V 系 CMOS 位準的時候不能連接。(*注 1)
輸出 B	漏極開路輸出。12mA 驅動緩衝 (Low 位準輸出電流 IOL=12mA 時 VOL=0.4Vmax)。 使用時、請用高阻抗電阻將 VDD (+3.3V) 提升。也可以連接 TTL 位準的 5V 系 IC。
雙方向 A	輸入側為、可以接受 5V 直接輸入的 LVTTTL 位準的史密特觸發電路輸入。I C 內部為高阻抗迴路。資料訊號是、訊號線請不要有高阻抗發生、系統整體在資料匯流排以高阻抗做提升。 D15~D8、PIN6、5 不使用時、高阻抗 (10k~100k Ω 左右) 串聯 VDD (+3.3V) 或與 GND 連接。因雙方向的原因、因直接提升/下拉而加入高阻抗連接的方式安全的。 Hi 位準輸出時時請不要在外掛超過輸出電壓以上的電壓。 輸出側為、3.3V 系 CMOS 位準的輸出。12mA 驅動緩衝 (Hi 位準輸出電流 IOH=-12mA 時 VOH=2.6Vmin、Low 位準輸出電流 IOL=12mA 時 VOL=0.4Vmax)。 5V 系雙方向 IC 之連接時、對接輸入若為 TTL 位準時可以連接。對接輸入為 5V 系 CMOS 位準的時候不能連接。(*注 1)
雙方向 B	輸入側為、可以接受 5V 直接輸入的 LVTTTL 位準的史密特觸發電路輸入。I C 內部以 50K Ω 做提升。 Hi 位準輸出時時請不要在外掛超過輸出電壓以上的電壓。 輸出側為、3.3V 系 CMOS 位準輸出。6mA 驅動緩衝 (Hi 位準輸出電流 IOH=-6mA 時 VOH=2.6Vmin、Low 位準輸出電流 IOL=6mA 時 VOL=0.4Vmax)。 5V 系雙方向 IC 之連接時、對接輸入若為 TTL 位準時可以連接。對接輸入為 5V 系 CMOS 位準的時候不能連接。(*注 1)
雙方向 C	輸入側為、可以接受 5V 直接輸入的 LVTTTL 位準的史密特觸發電路輸入。I C 內部以 100K Ω 做提升。 多晶片多軸補間時輸出側作動。多晶片多軸補間下晶片間訊號連接時、儘量將配線縮短、且不要和其他訊號線交叉。 不使用的話空口就可以。
雙方向 D	I2C 的 SDA 訊號專用。 輸入側為、可以接受 5V 直接輸入的 LVTTTL 位準的史密特觸發電路輸入。I C 內部為高阻抗迴路。 輸出側為、6mA 驅動緩衝的漏極開路輸出。作為 SDA 訊號使用時串接外部電阻以 VDD 提升。 不使用時、串接高阻抗電阻 (10k~100k Ω) 與 VDD 或 GND 連接。
雙方向 E	輸入側為、可以接受 5V 直接輸入的 LVTTTL 位準的史密特觸發電路輸入。I C 內部為高阻抗迴路。 輸出側為漏極開路輸出 多晶片多軸補間時輸出側作動。晶片間訊號連接時、儘量將配線縮短、且不要和其他訊號線交叉。 不使用時、串接高阻抗電阻 (10k~100k Ω) 與 VDD 或 GND 連接。

注 1: 輸出 A 及雙方向 A,B 的輸出訊號在外部介入電阻提升為 5V 也一樣、無法將 Hi 位準輸出電壓提高到 5V 系 CMOS 的 Hi 位準輸入電壓。請勿這樣的回路結構。

5.4 回路設計上注意

a. TEST1,2 端子的処置

TEST1,2(141,142)端子請務必接 GND。Hi 位準下內部測試回路作動、動作會完全不正常。

b. 未使用輸入端子的処置

不使用的輸入 A 端子務必與 GND 或 VDD 連接。不使用輸入端子若空腳時、端子訊號位準會不安定、造成 IC 機能不良。
輸入 B 端子可以空腳。

c. 未使用雙方向端子的処置

雙方向端子 (雙方向 A,D,E) 不使用的端子、請串接高阻抗 (10k~100k Ω 左右) 與 VDD 或 GND 連接。直接 GND 或 VDD 連接時、万一程式有問題發生輸出狀態時、過電流流過 IC 會破損。雙方向 B,C 端子可以空腳。

d. 去耦電容

本IC在 VDD 與 GND 間請加入 2~3 個高周波特性的 $0.1\mu\text{F}$ 左右的去耦電容。

e. 端子電感引起的振鈴雜訊

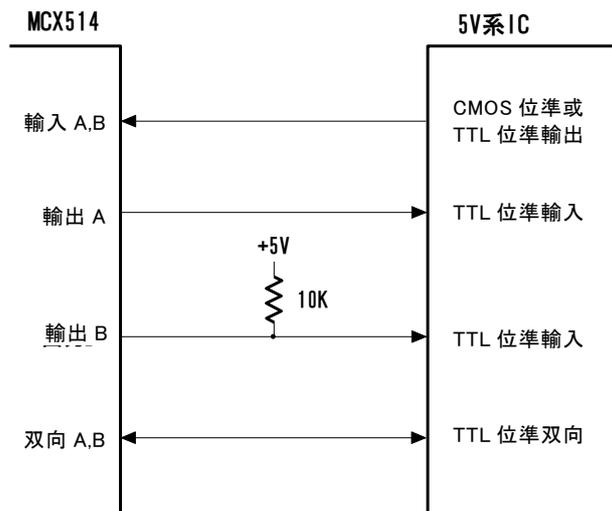
輸出端子的電感與輸出連接的負荷容量共振、輸出訊號觸發向上及向下有時會產生的振鈴雜訊。造成連接的下一段回路誤動作，振鈴雜訊大的時候可以用 $10\sim 100\text{pF}$ 左右的負荷容量連接、降低振鈴雜訊。

f. 傳送路的反射

輸出A, B及雙方向A~E的輸出、負荷容量 $20\sim 50\text{pF}$ 的時候、訊號觸發向上/向下的時間約 $3\sim 4\text{nsec}$ 、配線超過 60cm 長度時、反射的影響越來越顯著。可以的話請縮短配線長度。

g. 5V 系IC的連接例

本IC輸出入回路可以直接接 5V、但輸出回路僅可以連接 TTL 位準的輸入。CMOS 位準輸入不可以連接



6. 讀／寫暫存器

本章說明CPU對各軸控制時的讀／寫暫存器內容。

6.1 16位元資料匯流排暫存器位址

下表所示為使用16位元資料匯流排、16位元的讀／寫暫存器存取位址有8個。

■ 16位元資料匯流排寫入暫存器

全部暫存器為 16 位元長。

位址 A2 A1 A0	暫存器記号	暫存器名	內 容
0 0 0	WR0	指令暫存器	・ 軸指定、命令碼的設定
0 0 1	XWR1	X 軸模式暫存器 1	・ 中斷許可／禁止設定
	YWR1	Y 軸模式暫存器 1	
	ZWR1	Z 軸模式暫存器 1	
	UWR1	U 軸模式暫存器 1	
0 1 0	XWR2	X 軸模式暫存器 2	・ 外部減速停止訊號的理論位準、有効／無効設定 ・ 伺服馬達用訊號的理論位準、有効／無効設定 ・ 極限訊號的模式設定、軟體極限的模式設定
	YWR2	Y 軸模式暫存器 2	
	ZWR2	Z 軸模式暫存器 2	
	UWR2	U 軸模式暫存器 2	
0 1 1	XWR3	X 軸模式暫存器 3	・ 自動減速／手動減速設定 ・ 加減速的模式設定（對稱／非對稱、直線加減速／S 形加減速） ・ 驅動脈衝輸出的模式、端子設定 ・ 編碼器輸入訊號的模式、端子設定
	YWR3	Y 軸模式暫存器 3	
	ZWR3	Z 軸模式暫存器 3	
	UWR3	U 軸模式暫存器 3	
1 0 0	WR4	輸出暫存器 1	・ X 軸汎用輸出入訊號 XP107~0 輸出值設定 ・ Y 軸汎用輸出入訊號 YP107~0 輸出值設定
1 0 1	WR5	輸出暫存器 2	・ Z 軸汎用輸出入訊號 XP107~0 輸出值設定 ・ U 軸汎用輸出入訊號 YP107~0 輸出值設定
1 1 0	WR6	寫入資料暫存器 1	・ 寫入資料下位 16 位元 (D15~D0) 的設定
1 1 1	WR7	寫入資料暫存器 2	・ 寫入資料上位 16 位元 (D31~D16) 的設定

● 上表所示、各軸都持有、WR1、WR2、WR3(模式暫存器 1, 2, 3)。這些暫存器在同一位址寫入。哪一軸的模式暫存器寫入是在執行前指定寫入命令的軸所決定。另外、軸指定後NOP命令寫入前、選擇要寫入的軸。

● 重置時、WR1,WR2,WR3,WR4,WR5 暫存器全部位元清為 0。

■ 16位元資料匯流排讀取暫存器

全部的暫存器為16位元長。

位址 A2 A1 A0	暫存器記号	暫存器名	內 容
0 0 0	RR0	主狀態暫存器	<ul style="list-style-type: none"> 驅動狀態、錯誤狀態顯示 連續補間次資料可、円弧補間的象限、連續補間預存緩衝暫存器狀態計數器(SC)的顯示
0 0 1	XRR1 YRR1 ZRR1 URR1	X軸狀態暫存器 1 Y軸狀態暫存器 1 Z軸狀態暫存器 1 U軸狀態暫存器 1	<ul style="list-style-type: none"> 中斷發生要因的顯示
0 1 0	XRR2 YRR2 ZRR2 URR2	X軸狀態暫存器 2 Y軸狀態暫存器 2 Z軸狀態暫存器 2 U軸狀態暫存器 2	<ul style="list-style-type: none"> 錯誤發生要因的顯示 結束狀態的顯示
0 1 1	XRR3 YRR3 ZRR3 URR3	X軸狀態暫存器 3 Y軸狀態暫存器 3 Z軸狀態暫存器 3 U軸狀態暫存器 3	<ul style="list-style-type: none"> ● 頁 0 <ul style="list-style-type: none"> 輸入訊號的狀態顯示 自動原点復歸執行狀態的顯示 ● 頁 1 <ul style="list-style-type: none"> 同步動作組的有效/無效狀態顯示 加/減速狀態、加/減速度的增加/減少狀態顯示 計時器作動、分割脈衝作動的狀態顯示 多晶片補間時的終點資料傳送錯誤顯示
1 0 0	RR4	P I O讀取暫存器 1	<ul style="list-style-type: none"> X軸汎用輸出入訊號的狀態顯示 Y軸汎用輸出入訊號的狀態顯示
1 0 1	RR5	P I O讀取暫存器 2	<ul style="list-style-type: none"> Z軸汎用輸出入訊號的狀態顯示 U軸汎用輸出入訊號的狀態顯示
1 1 0	RR6	讀取資料暫存器 1	<ul style="list-style-type: none"> 讀取資料下位 16 位元 (D15~D0) 的顯示
1 1 1	RR7	讀取資料暫存器 2	<ul style="list-style-type: none"> 讀取資料上位 16 位元 (D31~D16) 的顯示

● 寫入暫存器一樣、各軸持有、RR1、RR2、RR3(各軸狀態暫存器 1, 2, 3)。這些暫存器都在同一位址讀取。那軸的狀態暫存器讀取是在讀取前指定軸來決定。

● 有關 RR3 暫存器、存在頁 0 及頁 1 的 2 種。RR3 頁顯示命令(7Ah、7Bh)寫入時指定頁來讀取。重置時預設為頁 0。

6.2 8 位元資料匯流排的暫存器位址

8 位元資料匯流排的存取時、是將 16 位元暫存器分成上位位元組、下位位元組來存取。

下表所示、****L 為 16 位元暫存器****的下位位元組(D7~D0)、****H 是 16 位元暫存器****的上位位元組(D15~D8)。其中只有指令暫存器(WROL, WROH)一定要先寫入上位位元組(WROH)後再寫入下位位元組(WROL)。

■ 8 位元資料匯流排寫入暫存器

位址				寫入暫存器
A3	A2	A1	A0	
0	0	0	0	WROL
0	0	0	1	WROH
0	0	1	0	WR1L
0	0	1	1	WR1H
0	1	0	0	WR2L
0	1	0	1	WR2H
0	1	1	0	WR3L
0	1	1	1	WR3H
1	0	0	0	WR4L
1	0	0	1	WR4H
1	0	1	0	WR5L
1	0	1	1	WR5H
1	1	0	0	WR6L
1	1	0	1	WR6H
1	1	1	0	WR7L
1	1	1	1	WR7H

■ 8 位元資料匯流排讀取暫存器

位址				讀取暫存器
A3	A2	A1	A0	
0	0	0	0	RR0L
0	0	0	1	RR0H
0	0	1	0	RR1L
0	0	1	1	RR1H
0	1	0	0	RR2L
0	1	0	1	RR2H
0	1	1	0	RR3L
0	1	1	1	RR3H
1	0	0	0	RR4L
1	0	0	1	RR4H
1	0	1	0	RR5L
1	0	1	1	RR5H
1	1	0	0	RR6L
1	1	0	1	RR6H
1	1	1	0	RR7L
1	1	1	1	RR7H

6.3 I²C 串列介面匯流排模式的暫存器位址

本 IC 使用 I²C 串列介面匯流排時、以位址控制的方式指定暫存器的存取。暫存器位址的指定方法與 6.2 節所述相同、將 16 位元暫存器分成上位位元組、下位位元組。

I²C 串列介面匯流排請參考第 4 章。

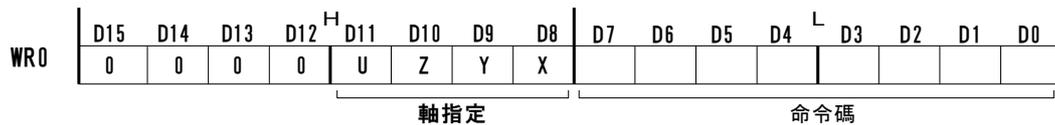
6.4 WRO 指令暫存器

針對IC內的各軸、軸指定後、寫入命令的暫存器。暫存器由軸指定位元及命令碼位元所組成。

此暫存器寫入命令碼後、該命令立刻執行。驅動速度的設定...等資料寫入命令須先將資料寫入 WR6,7 暫存器。或是、資料讀取命令此指令暫存器寫入後、內部回路會將要讀取的資料放在 RR6,7 暫存器。

8位元資料匯流排的時請務必先寫上位位元組(H)後再寫下位位元組(L)。下位位元組寫入後先前指定的軸命令立刻執行。

全部的命令碼的命令處理所要時間、最多 125nsec (CLK=16MHz 的時候)。在此時間內請勿重複寫入下一命令。



D7~0 命令碼設定。

D11~8 指定命令執行軸。各軸的位元設1後該軸被指定。軸的指定不限1軸。可同時對複數軸下達相同命令、相同參數值。但、資料讀取命令的時候僅可以指定1軸。

補間關係的命令、軸指定的位元請全部設為0。

其他位元請務必設為0。設1的時候、IC內部回路的測試命令起動、會發生無法憶測的動作易其。

6.5 WR1 模式暫存器 1

模式暫存器 1 是 4 軸個別持有。哪軸的模式暫存器寫入、在寫入前由命令指定。

模式暫存器1為各中斷發生要因的許可/禁止設定暫存器。各位元設為1時中斷許可、0時中斷禁止。



D3~0 CMR3~0 多目的暫存器 MR3~0 與比較對象的比較結果、滿足比較條件時、中斷發生。
MR3~0 的比較對象及比較條件的設定、以多目的暫存器模式設定命令(20h)進行。

D4 D-STA 驅動開始時、中斷發生。

D5 C-STA 加減速驅動時、等速域脈衝輸出開始時、中斷發生。

D6 C-END 加減速驅動時、等速域的脈衝輸出結束時、中斷發生。

D7 D-END 驅動結束時、中斷發生。

D8 H-END 自動原點復歸結束、中斷發生。

D9 TIMER 計時器時間到時、中斷發生。

D10 SPLTP 分割脈衝的脈衝↑時中斷發生。(分割脈衝理論:Hi 脈衝設定時)

D11 SPLTE 分割脈衝結束時、中斷發生。

D15~12 SYNC3~0 同步動作 SYNC3~0 起動時、中斷發生。

重置時、D15~D0 全部設為0。

6.6 WR2 模式暫存器2

模式暫存器2是4軸個別持有。哪軸的模式暫存器寫入、在寫入前由命令指定。

模式暫存器2為、驅動途中減速停止/即停止的輸入訊號 STOP2~STOP0 的模式設定、伺服馬達用輸入訊號的模式設定、極限輸入訊號的模式設定、及軟體極限的模式設定的暫存器。

WR2	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	SLM-M	SLM-O	SLM-E	HLM-M	HLM-E	HLM-L	ALM-E	ALM-L	INP-E	INP-L	SP2-E	SP2-L	SP1-E	SP1-L	SP0-E	SP0-L

- D4, 2, 0 SPn-L** 驅動停止輸入訊號 STOPn (n:2~0) 的有效的理論位準設定位元。
0:Low 動作、1:Hi 動作
自動原点復歸時、使用 STOPn 訊號的理論位準在這些位元設定。
- D5, 3, 1 SPn-E** 驅動停止輸入訊號 STOPn (n:2~0) 的有効/無効設定位元。
0:無効、1:有効
STOP2~STOP0 設有効後開始驅動時、指定的 STOP 訊號輸入動作時、驅動減速停止或即停止。加減速驅動時減速停止、等速驅動時即停止。
自動原点復歸時、使用的 STOPn 有効/無効位元做設定。
- D6 INP-L** 伺服馬達定位完成輸入訊號 nINPOS 的理論位準設定。
0:Low 動作、1:Hi 動作
- D7 INP-E** nINPOS 輸入訊號的有効/無効設定。
0:無効、1:有効
有効に設定すると、驅動結束後、等到 nINPOS 訊號動作後 RR0(主狀態)暫存器的 DRIVE 位元才回到 0。
- D8 ALM-L** 伺服馬達異常用輸入訊號 nALARM 的理論位準設定。
0:Low 動作、1:Hi 動作
- D9 ALM-E** nALARM 輸入訊號的有効/無効設定。
0:無効、1:有効
設定有効後、驅動中 nALARM 輸入訊號動作時 RR2 暫存器的 D4(ALARM)位元變為1。驅動即停止。
- D10 HLM-L** 硬體極限輸入訊號 nLMTP, nLMTM 的理論位準設定。
0:Low 動作、1:Hi 動作
- D11 HLM-E** nLMTP, nLMTM 極限輸入訊號的有効/無効設定。
0:無効、1:有効
有効設定時、+方向驅動中 nLMTP 極限輸入訊號動作時 RR2 暫存器的 D2(HLMT+)變為1、一方向驅動中 nLMTM 極限輸入訊號動作時、RR2 暫存器的 D3(HLMT-)變為1。驅動停止。
- D12 HLM-M** nLMTP, nLMTM 極限輸入訊號動作時驅動停止方式設定。
0:即停止、1:減速停止
自動原点復歸的停止訊號以極限訊號使用時、1:減速停止設定。
- D13 SLM-E** 軟體極限機能的有效/無効設定。
0:無効、1:有効
設定有効時、+方向驅動中+方向的軟體極限錯誤狀態時、RR2 暫存器的 D0(SLMT+)變為1、一方向驅動中-方向的軟體極限錯誤狀態時、RR2 暫存器的 D1(SLMT-)變為1。
・+方向的軟體極限:対象的位置計數 \geq SLMT+ 值時錯誤、驅動停止
・-方向的軟體極限:対象的位置計數 $<$ SLMT- 值時錯誤、驅動停止
軟體極限錯誤發生方向的驅動命令寫入時不會執行。
- D14 SLM-O** 軟體極限的設定対象理論位置數器、實際位置計數器設定。
0:理論位置計數器、1:實際位置計數器
- D15 SLM-M** 軟體極限時的驅動停止方式設定。
0:減速停止、1:即停止

(請注意設定與硬體極限訊號的停止方式的設定位元的0/1相反。)

重置時、D15~D0 全部0。

6.7 WR3 模式暫存器3

模式暫存器3是4軸個別持有。哪軸的模式暫存器寫入、在寫入前由命令指定。

模式暫存器3為、手動減速、加減速模式(對稱/非對稱、直線加減速/S形加減速)、驅動脈衝輸出模式、編碼器輸入模式、極限訊號端子交換、台形三角防止機能、計時器重覆的設定暫存器。

WR3	D15	D14	D13	H				L				D3	D2	D1	D0
	0	TMMD	AVTRI	LMINV	PIINV	PI-L	PIMD1	PIMD0	DPINV	DIR-L	DP-L	DPMD1	DPMD0	SACC	DSNDE

- D0 MANLD 加減速定量脈衝驅動時減速為自動減速或手動減速設定。
0:自動減速、1:手動減速
手動減速模式時、手動減速點(DP)須先設定。
- D1 DSNDE 直線加減速驅動的減速時的減速度、用加速度設定值(對稱)、或是個別的減速度設定值(非對稱)的設定。另、S形加減速驅動的減速時的減速度增加率使用加速度增加率設定值(對稱)、或個別的減速度增加率設定值(非對稱)的設定。
0:對稱加減速、1:非對稱加減速
非對稱S形加減速做定量脈衝驅動時、不會自動減速之故、D0(MANLD)位元設為1時、手動減速點(DP)就需要設定。
- D2 SACC 加減速驅動時的速度曲線為直線加減速或S形加減速的設定。
0:直線加減速、1:S形加減速
S形加減速的時候時、需要設定加速度增加率(JK)、(減速度增加率(DJ))。
- D4, 3 DPMD1, 0 驅動脈衝的輸出方式設定。

D4 (DPMD1)	D3 (DPMD0)	驅動脈衝輸出方式
0	0	獨立2脈衝方式
0	1	1脈衝・方向方式
1	0	2相脈衝4通倍方式
1	1	2相脈衝2通倍方式

獨立2脈衝方式時、輸出訊號PP為+方向脈衝、輸出訊號PM為-方向脈衝輸出。
單1脈衝・方向方式時、輸出訊號PLS為+/-兩方向的驅動脈衝、輸出訊號DIR為脈衝的方向訊號輸出。
2相脈衝方式時輸出訊號PA為2相脈衝的A相訊號、輸出訊號PB為2相脈衝的B相訊號輸出。

- D5 DP-L 驅動脈衝的理論位準設定。
0:正理論脈衝、1:負理論脈衝

正理論脈衝： 負理論脈衝：

- D6 DIR-L 驅動脈衝輸出方式為1脈衝・方向方式設定時、驅動脈衝的方向輸出訊號的理論位準設定。
此位元的值如下表 DIR 輸出訊號的電壓位準輸出。

D6 (DIR-L)	+方向脈衝輸出時	-方向脈衝輸出時
0	Low	Hi
1	Hi	Low

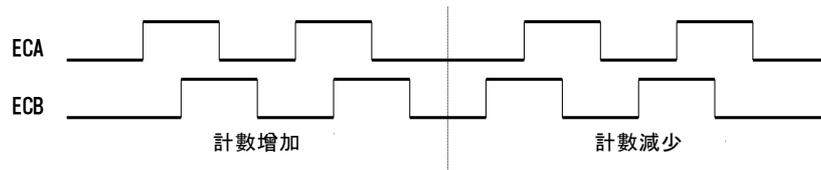
- D7 DPINV 驅動脈衝輸出的PP/PLS/PA訊號與PM/DIR/PB訊號的輸出端子交換。
0:初期狀態、1:端子交換

此位元設1時、驅動脈衝輸出方式為獨立2脈衝方式的時候、+方向的驅動在 PM 訊號輸出驅動脈衝、-方向的驅動在 PP 訊號輸出驅動脈衝。其他驅動脈衝輸出方式也一樣訊號輸出端子交換。

- D9, 8 PIMD1, 0 編碼器輸入脈衝方式設定。
編碼器輸入訊號、實際位置計數器計數增/減。

D9 (PIMD1)	D3 (PIMD0)	編碼器輸入脈衝方式
0	0	2 相脈衝輸入 4 通倍
0	1	2 相脈衝輸入 2 通倍
1	0	2 相脈衝輸入 1 通倍
1	1	上/下脈衝輸入

此位元設為2相脈衝輸入的模式時、正理論脈衝在A相前進時計數增加、B相前進時計數減少。4 通倍設定時兩訊號的↑、↓時計數增加、減少。2 通倍設定時 A 相訊號的↑、↓時計數增加、減少。1 通倍設定時は B 相訊號 Low 時的 A 相訊號的↑時計數增加、B 相訊號 Low 時的 A 相訊號↓時計數減少。



此位元增加/減少脈衝輸入的模式設定時、PPIN 訊號計數增加輸入、PMIN 訊號計數減少輸入。各訊號在正脈衝的↑計數。

- D10 PI-L 編碼器輸入訊號的正理論/負理論設定。
0: 正理論、1: 負理論
此設定編碼器輸入脈衝方式的增/減脈衝方式在負脈衝的↓時計數。
- D11 PIINV 編碼器輸入脈衝的 nECA/PPIN 訊號與 nECB/PMIN 訊號的輸入端子交換。
0: 初期狀態、1: 端子交換
因此設的的關係、以下的實際位置計數的增減相反。

D11 (PIINV)	編碼器輸入脈衝方式	實際位置計數 (RP) 的增減
0	2 相脈衝輸入	A 相前進時計數增加。 B 相前進時計數減少。
	增/減脈衝輸入	PPIN 脈衝輸入時計數增加。 PMIN 脈衝輸入時計數減少。
1	2 相脈衝輸入	B 前進時計數增加。 A 相前進時計數減少。
	增/減脈衝輸入	PMIN脈衝輸入時計數增加。 PPIN 脈衝輸入時計數減少。

- D12 LMINV 硬體極限輸入訊號 nLMTP, nLMTM 的輸入端子交換。
0: 初期狀態、1: 端子交換
此位元為1時、nLMTP 訊號為-方向的極限訊號、nLMTM 訊號為+方向的極限訊號。
- D13 AVTRI 直線加減速的定量脈衝驅動三角波形防止機能的有效/無效設定。重置時、三角波形防止機能有效。
0: 有效、1: 無效
- D14 TMMD 計時器動作單次/重覆的設定。
0: 單次、1: 重覆

重置時、D15~D0 全部0。D15 位元請設定為0。

6.8 WR4 輸出暫存器

X軸汎用輸出入訊號 XPIO7~0 及Y軸汎用輸出入訊號 YPIO7~0 做為汎用輸出使用時、輸出設定暫存器。各位元0時 Low 位準、1設定時 Hi 位準輸出。

WR4	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
	YP107	YP106	YP105	YP104	YP103	YP102	YP101	YP100	XPIO7	XPIO6	XPIO5	XPIO4	XPIO3	XPIO2	XPIO1	XPIO0

重置時、D15~D0 全部0。

6.9 WR5 輸出暫存器

Z軸汎用輸出入訊號 ZPIO7~0 及U軸汎用輸出入訊號 UPIO7~0 做為汎用輸出使用時、輸出設定暫存器。各位元0時 Low 位準、1設定時 Hi 位準輸出。

WR5	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
	UP107	UP106	UP105	UP104	UP103	UP102	UP101	UP100	ZPIO7	ZPIO6	ZPIO5	ZPIO4	ZPIO3	ZPIO2	ZPIO1	ZPIO0

重置時、D15~D0 全部0。

6.10 WR6, 7 資料寫入暫存器 1, 2

資料寫入命令的資料設定暫存器。WR6 暫存器為寫入資料下位 16 位元 (WD15~WD0)、WR7 暫存器為寫入資料上位 16 位元 (WD31~WD16) 設定。

WR6	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
	WD15	WD14	WD13	WD12	WD11	WD10	WD9	WD8	WD7	WD6	WD5	WD4	WD3	WD2	WD1	WD0

WR7	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
	WD31	WD30	WD29	WD28	WD27	WD26	WD25	WD24	WD23	WD22	WD21	WD20	WD19	WD18	WD17	WD16

資料寫入命令是、首先、各々の命令所指定資料長的資料寫入此寫入資料暫存器。寫入資料暫存器 WR6,7 (8位元資料匯流排的時候 WR6L,WR6H,WR7L,WR7H)、高低位元組無先後寫入順序的規定。之後、命令碼寫入指令暫存器時、寫入資料暫存器的內容就會被取用。

寫入數值資料全部二進制。而負數以2的補數寫入。

各々の命令的資料請務必依照指定資料長設定。

重置時、WR6,WR7 暫存器的內容為亂數。

6.11 RRO 主狀態暫存器

主狀態暫存器顯示各軸的驅動、錯誤狀態。或是連續補間下筆資料許可、凹弧補間的象限、連續補間的預存緩衝暫存器狀態計數器(SC)的顯示。

RRO	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
	HSTC3	HSTC2	HSTC1	HSTC0	CNEXT	ZONE2	ZONE1	ZONE0	U-ERR	Z-ERR	Y-ERR	X-ERR	U-DRV	Z-DRV	Y-DRV	X-DRV

D3~0 n-DRV 顯示各軸的驅動狀態。此位元1時、該軸為驅動脈衝輸出中。0時該軸驅動結束。另、自動原点復歸執行時、執行期其間此位元1。

伺服馬達定位完成用輸入訊號 nINPOS 有效設定時、驅動脈衝輸出後、nINPOS 訊號動作後、此位元0。

D7~4 n-ERR 各軸的錯誤發生狀態整合顯示。也就是各軸的 RR2 暫存器的錯誤位元(D7~D0)有位元式1時、此位元就會設為1。多晶片補間時、副晶片錯誤發生時主晶片的主軸的錯誤位元變為1。

補間驅動以外的驅動時(含自動原復歸)、錯誤・結束狀態清除命令(79h)、或是下次驅動開始時、此位元回到0。補間驅動時、請務必以錯誤・結束狀態清除命令(79h)將錯誤解除。錯誤沒有解除時、此後的補間驅動會不正常。

D10~8 ZONEm 凹弧補間驅動時顯示現在驅動中的象限。

D10	D9	D8	現在驅動中的象限
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

D11 CNEXT 顯示連續補間下一筆資料寫入許可。連續補間驅動、此位元1時、下一筆區段的補間資料(參數資料及補間命令)可以開始寫入。

D15~12 HSTC3~0 顯示連續補間的預存緩衝暫存器狀態計數器(SC)的值。

D15	D14	D13	D12	狀態計數器(SC)的值
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8

連續補間的驅動中、SC=8時、預存緩衝暫存器的堆疊為上限狀態。SC在7以下的時、可以寫入下一筆區段的補間資料(參數資料及補間命令)。SC=0為補間資料全部輸出完畢、連續補間的驅動結束。

6.12 RR1 狀態暫存器1

狀態暫存器1為4軸個別持有。讀取哪一軸的狀態暫存器是在命令的軸指定。

狀態暫存器1為中斷發生要因顯示的暫存器。中斷發生時、對應的中斷發生要因的位元變為1。
中斷發生需先在由 WR1 暫存器設定各要因中斷許可。

RR1	D15	D14	D13	D12	H				D7	D6	D5	D4	L			
	SYNC3	SYNC2	SYNC1	SYNC0	SPLTE	SPLTP	TIMER	H-END	D-END	C-END	C-STA	D-STA	CMR3	CMR2	CMR1	CMR0
中斷發生要因																

- D3~0 CMR3~0 顯示多目的暫存器 MR3~0 與比較對象的比較結果、滿足比較條件時中斷發生。
MR3~0 的比較對象及比較條件的設定、請以多目的暫存器模式設定命令(20h)。
- D4 D-STA 顯示驅動開始。
- D5 C-STA 顯示加減速驅動時、等速域的脈衝輸出的開始。
- D6 C-END 顯示加減速驅動時、等速域的脈衝輸出的結束。
- D7 D-END 顯示驅動結束。
- D8 H-END 顯示自動原點復歸結束。
- D9 TIMER 顯示計時器的時間到。
- D10 SPLTP 顯示分割脈衝的脈衝的↑。
(分割脈衝理論:Hi 脈衝設定時)
- D11 SPLTE 顯示分割脈衝結束。
- D15~12 SYNC3~0 顯示同步動作 SYNC3~0 的起動。

所有中斷要因的中斷發生時、此暫存器的位元變1、中斷輸出訊號(INT0N)變 Low 位準。CPU讀取 RR1 暫存器後、RR1 暫存器的位元清為0、中斷輸出訊號(INT0N)回到未動作位準。

【注意】

- 8位元資料匯流排的時候 RR1L 暫存器的讀取 RR1L 清除、RR1H 的讀取 RR1H 清除。RR1L 的讀取時不會清除 RR1H。RR1H 的讀取也不會清除 RR1L。
- I²C 串列介面匯流排的時候時、RR1L 暫存器與 RR1H 暫存器不可以分開讀取、請務必再一次通訊情況下完整讀取 RR1 暫存器。

6.13 RR2 狀態暫存器2

狀態暫存器2為4軸個別持有。讀取哪一軸的狀態暫存器是在命令的軸指定。

狀態暫存器2為錯誤情報、及驅動結束狀態顯示暫存器。錯誤情報(D7~D0)各位元1的時候、顯示驅動中該位元的錯誤發生。此RR2暫存器的D7~D0的不管哪個位元變1時、RR0主狀態暫存器的該當軸的n-ERR位元變1。

RR2暫存器的各位元1時、錯誤要因、或驅動結束要因解消依舊保持為1。補間驅動以外驅動(含自動原点復歸)的錯誤、用錯誤・結束狀態清除命令(79h)、或是下次驅動開始時全部位元會回到0。補間驅動時的錯誤請務必用錯誤・結束狀態清除命令(79h)將錯誤位元歸0。

RR2	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
		EMG	ALARM	LMTM	LMT+	STOP2	STOP1	STOP0	SYNC	CERR	HOME	EMG	ALARM	HLMT-	HLMT+	SLMT-

└──────────────────────────────────┘
└──────────────────────────────────┘

驅動結束狀態
錯誤資訊

- D0 SLMT+ 軟體極限機能有效時、+方向驅動中比較對象的位置計數超過SLMT+值以上時變1、驅動停止。
- D1 SLMT- 軟體極限機能有效時、-方向驅動中比較對象的位置計數小於SLMT-值時變1、驅動停止。
- D2 HLMT+ 硬體極限訊號有效時、+方向驅動中、nLMT+極限訊號動作時變1、驅動停止。
- D3 HLMT- 硬體極限訊號有效時、-方向驅動中、nLMT-極限訊號動作時變1、驅動停止。
- D4 ALARM 伺服馬達異常用輸入訊號有效時、驅動中nALARM訊號動作時變1、驅動停止。
- D5 EMG 驅動中、緊急停止訊號EMGN為Low位準時變為1、驅動停止。
- D6 HOME 自動原点復歸執行時的錯誤。復歸步驟3開始時、編碼器Z相訊號STOP2已經動作時變為1。
- D7 CERR 補間錯誤、連續補間的驅動中下一筆區段的補間資料的來不及寫入而停止、另外多晶片補間時的終點資料轉送錯誤的時變為1。多晶片補間時的資料轉送錯誤發生時、RR3暫存器頁1的D12位元也會變為1。

【注意】

D7位元1的時候、請務必以錯誤・結束狀態清除命令(79h)清除。錯誤無解除時其後的補間驅動會不正常動作。

驅動中進行方向的硬體／軟體極限作動時、驅動會減速停止或是即停止。停止後停止要因解消前、同方向驅動命令在發行也再錯誤發生、驅動命令不會執行。

錯誤情報位元、在驅動停止狀態下、各要因即使動作也不會變為1。
軟體極限、硬體極限、反方向驅動時即使要因動作也不會發生錯誤。

- D8 SYNC 驅動中因同步動作SYNC3~0停止、變為1。
- D11~9 STOP2~0 驅動中因外部停止訊號STOP2~0停止、變為1。
- D12 LMT+ 驅動中因+方向極限訊號nLMT+停止、變為1。
- D13 LMT- 驅動中因-方向極限訊號nLMT-停止、變為1。
- D14 ALARM 驅動中因伺服馬達用アラーム訊號nALARM停止、變為1。
- D15 EMG 驅動中因緊急停止訊號EMGN停止、變為1。

驅動結束狀態(D15~D8)顯示驅動結束的要因位元。即使驅動結束要因為驅動結束狀態(D15~D8)所示要因以外、有以下3個要因。

- a. 定量脈衝驅動輸出脈衝全部送出。
- b. 減速停止、即停止命令寫入時。
- c. 軟體極限有效設定動作時。

驅動結束狀態 (D15~D8) 的確認請務必在 RR0 主狀態暫存器的 n-DRV 位元的驅動結束確認後進行。

6.14 RR3 狀態暫存器3

狀態暫存器3為4軸個別持有。讀取哪一軸的狀態暫存器是在命令的軸指定。

狀態暫存器3有頁0及頁1的2種存在。頁0為輸入訊號的狀態直接顯示的暫存器。或自動原点復歸執行時的執行狀態顯示。頁1為、同步動作的有效/無效設定、驅動中的加減速驅動的加減速狀態、S形加減速的加速度增加/減少狀態、計時器作動狀態、分割脈衝作動狀態顯示。另、多晶片補間時的傳送錯誤狀態也顯示。
重置時は頁0となります。

RR3 頁 0	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
	0	HSST5	HSST4	HSST3	HSST2	HSST1	HSST0	LMTM	LMTM	ALARM	INPOS	ECB	ECA	STOP2	STOP1	STOP0
RR3 頁 1	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
	1	0	0	MCERR	SPLIT	TIMER	ADSND	ACNST	AASND	DSND	CNST	ASND	SYNC3	SYNC2	SYNC1	SYNC0

■ 頁 0

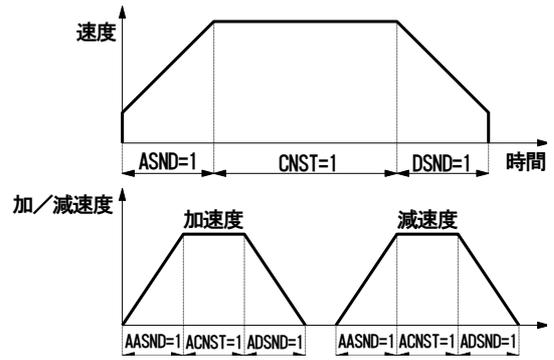
各訊號的輸入訊號狀態位元、輸入訊號 Low 位準時0、Hi 位準の時1。D8~D0 的輸入訊號功能未使用時可以作為汎用輸入訊號使用。以下的說明中訊號名的後的括弧內為 X 軸到 U 軸順序的端子編號。

- D2~0 STOP2~0 外部停止訊號 STOP2 (70,91,110,129)、STOP1 (73,92,111,130)、STOP0 (74,93,112,131) 的輸入狀態顯示。
- D3 ECA 編碼器輸入脈衝訊號 ECA/PPIN (45,47,49,51) 的輸入狀態顯示。
即使編碼器輸入端子交換設定 (WR3/D11:PIINV)、本位元為輸入狀態顯示端子番号不變。
- D4 ECB 編碼器輸入脈衝訊號 ECB/PMIN (46,48,50,52) 的輸入狀態顯示します。
即使編碼器輸入端子交換設定 (WR3/D11:PIINV) 本位元為輸入狀態顯示端子番号不變。
- D5 INPOS 伺服馬達定位完成用輸入訊號 INPOS (66,85,104,123) 的輸入狀態顯示。
- D6 ALARM 伺服馬達異常用輸入訊號 ALARM (67,86,105,124) 的輸入狀態顯示。
- D7 LMTM 硬體極限輸入訊號 LMTM (68,87,106,127) 的輸入狀態顯示。
即使硬體極限輸入端子交換設定 (WR3/D12:LMINV) 本位元為輸入狀態顯示端子番号不變。
- D8 LMTM 硬體極限輸入訊號 LMTM (69,88,109,128) 的輸入狀態顯示。
即使硬體極限輸入端子交換設定 (WR3/D12:LMINV) 本位元為輸入狀態顯示端子番号不變。
- D14~9 HSST5~0 自動原点復歸執行狀態在自動原点復歸執行中顯示現在執行的動作內容。
請參考 2.6.5 項。
- D15 PAGE 頁 0 顯示時 0。

■ 頁 1

- D3~0 SYNC3~0 同步動作 SYNC3~0 有效狀態の時 1。
同步動作有效時、發出行同步動作有效命令 (8F~81h)。同步動作無效時、不動作時發出同步動作無效命令 (9F~91h)。

D4	ASND	加減速驅動中加速狀態的時候1。
D5	CNST	加減速驅動中、等速狀態的時候1。
D6	DSND	加減速驅動中、減速狀態的時候1。
D7	AASND	S形加減速驅動中、加速度／減速度增加狀態的時候1。
D8	ACNST	S形加減速驅動、速度一定狀態的時候1。
D9	ADSND	S形加減速驅動中、速度減少狀態的時候1。
D10	TIMER	計時器動作中變為1。
D11	SPLIT	分割脈衝動作中變為1。
D12	MCERR	多晶片補間時的資料傳送錯誤發生的時候1。此時候、RR2 暫存器的 D7 位元也會1。
D15	PAGE	頁 1 顯示時 1。



6.15 RR4 PIO讀取暫存器1

PIO 讀取暫存器1為、X軸的汎用輸出入訊號 XPIO7~0、及Y軸的汎用輸出入訊號 YPIO7~0 的訊號狀態顯示暫存器。訊號 Low 位準的時候0、Hi 位準的時候1。

RR4	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
	YPIO7	YPIO6	YPIO5	YPIO4	YPIO3	YPIO2	YPIO1	YPIO0	XPIO7	XPIO6	XPIO5	XPIO4	XPIO3	XPIO2	XPIO1	XPIO0

D7~0 XPIO7~0 X軸的汎用輸出入訊號 XPIO7~0 的狀態顯示。
XPIO7~0 訊號設定為輸入時顯示輸入狀態、設定為輸出時顯示輸出狀態。

D15~8 YPIO7~0 Y軸的汎用輸出入訊號 YPIO7~0 的狀態顯示。
YPIO7~0 訊號設定為輸入時顯示輸入狀態、設定為輸出時顯示輸出狀態。

6.16 RR5 PIO讀取暫存器2

PIO 讀取暫存器1は、Z軸的汎用輸出入訊號 ZPIO7~0、及U軸的汎用輸出入訊號 UPIO7~0 的訊號狀態顯示暫存器。。訊號 Low 位準的時候0、Hi 位準的時候1。

RR5	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
	UPI07	UPI06	UPI05	UPI04	UPI03	UPI02	UPI01	UPI00	ZPIO7	ZPIO6	ZPIO5	ZPIO4	ZPIO3	ZPIO2	ZPIO1	ZPIO0

D7~0 ZPIO7~0 Z軸的汎用輸出入訊號 ZPIO7~0 的狀態顯示。
ZPIO7~0 訊號設定為輸入時顯示輸入狀態、設定為輸出時顯示輸出狀態。

D15~8 UPIO7~0 U軸的汎用輸出入訊號 UPIO7~0 的狀態顯示。
UPI07~0 訊號設定為輸入時顯示輸入狀態、設定為輸出時顯示輸出狀態。

6.17 RR6, 7 讀取資料暫存器1, 2

資料讀取命令下、內部暫存器的資料設定到此暫存器。RR6 暫存器為讀取資料下位 16 位元 (RD15~RD0)、RR7 暫存器為讀取資料上位 16 位元 (RD31~RD16)。

RR6	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
	RD15	RD14	RD13	RD12	RD11	RD10	RD9	RD8	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0

RR7	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
	RD31	RD30	RD29	RD28	RD27	RD26	RD25	RD24	RD23	RD22	RD21	RD20	RD19	RD18	RD17	RD16

資料全部為二進位。負的值为 2 的補數。

7. 命令

7.1 命令一覽

■ 資料寫入命令

編碼	命令	參數記号	資料範圍	資料長 (位元組)
00h	加速度增加率 設定	J K	1 ~ 1,073,741,823 [pps/sec ²]	4
01	減速度增加率 設定	D J	1 ~ 1,073,741,823 [pps/sec ²]	4
02	加速度 設定	A C	1 ~ 536,870,911 [pps/sec]	4
03	減速度 設定	D C	1 ~ 536,870,911 [pps/sec]	4
04	初速度 設定	S V	1 ~ 8,000,000 [pps]	4
05	驅動速度 設定	D V	1 ~ 8,000,000 [pps]	4
06	移動脈衝數/終點 設定	T P	-2,147,483,646 ~ +2,147,483,646	4
07	手動減速點 設定	D P	0 ~ 4,294,967,292	4
08	円弧中心點 設定	C T	-2,147,483,646 ~ +2,147,483,646	4
09	理論位置計數 設定	L P	-2,147,483,648 ~ +2,147,483,647	4
0A	實際位置計數 設定	R P	-2,147,483,648 ~ +2,147,483,647	4
0B	軟體極限+ 設定	S P	-2,147,483,648 ~ +2,147,483,647	4
0C	軟體極限- 設定	S M	-2,147,483,648 ~ +2,147,483,647	4
0D	加速計數偏移 設定	A O	-32,768 ~ +32,767	2
0E	理論位置計數最大值 設定	L X	1 ~ 2,147,483,647 (7FFF FFFFh) または FFFF FFFFh	4
0F	實際位置計數最大值 設定	R X	1 ~ 2,147,483,647 (7FFF FFFFh) または FFFF FFFFh	4
10	多目的暫存器0 設定	M R 0	-2,147,483,648 ~ +2,147,483,647	4
11	多目的暫存器1 設定	M R 1	-2,147,483,648 ~ +2,147,483,647	4
12	多目的暫存器2 設定	M R 2	-2,147,483,648 ~ +2,147,483,647	4
13	多目的暫存器3 設定	M R 3	-2,147,483,648 ~ +2,147,483,647	4
14	原点檢出速度 設定	H V	1 ~ 8,000,000 [pps]	4
15	速度增減值 設定	I V	1 ~ 1,000,000 [pps]	4
16	計時器值 設定	T M	1 ~ 2,147,483,647 [μ sec]	4
17	分割脈衝設定1	S P 1	分割脈衝長度 : 2 ~ 65,535 脈衝寬度 : 1 ~ 65,534	4
18	分割脈衝設定2	S P 2	分割脈衝數 : 0 ~ 65,535	2
19	補間・終點最大值	T X	0 ~ 2,147,483,646	4
1A	螺旋回轉數	H N	0 ~ 65535	2
1B	螺旋演算值	H V	1 ~ 2,147,483,646	4

【注意】

- 資料寫入請務必依據指定的資料長寫入。
- 速度參數值、計時器值的記載的單位為、輸入脈衝(CLK)在16MHz的時候。輸入時序(CLK)若16MHz以外的時候的參數計算式請參考付録B。

■ 模式寫入命令

編碼	命令	參數記号	資料長 (位元組)
2 0h	多目的暫存器模式設定	MRM	2
2 1	P I O 訊號設定 1	P 1 M	2
2 2	P I O 訊號設定 2 · 其他設定	P 2 M	2
2 3	自動原点復歸模式設定 1	H 1 M	2
2 4	自動原点復歸模式設定 2	H 2 M	2
2 5	輸入訊號濾波模式設定	F L M	2
2 6	同步動作 S Y N C 0 設定	S 0 M	2
2 7	同步動作 S Y N C 1 設定	S 1 M	2
2 8	同步動作 S Y N C 2 設定	S 2 M	2
2 9	同步動作 S Y N C 3 設定	S 3 M	2
2 A	補間模式設定	I P M	2

【注意】

- 資料寫入請務必依據指定的資料長寫入。

■ 資料讀取命令

編碼	命令	參數記号	資料範圍	資料長 (位元組)
3 0h	理論位置計數 讀取	L P	-2, 147, 483, 648 ~ +2, 147, 483, 647	4
3 1	實際位置計數 讀取	R P	-2, 147, 483, 648 ~ +2, 147, 483, 647	4
3 2	現在驅動速度 讀取	C V	0 ~ 8, 000, 000 [pps]	4
3 3	現在加減速度 讀取	C A	0 ~ 536, 870, 911 [pps/sec]	4
3 4	多目的暫存器 0 讀取	M R 0	-2, 147, 483, 648 ~ +2, 147, 483, 647	4
3 5	多目的暫存器 1 讀取	M R 1	-2, 147, 483, 648 ~ +2, 147, 483, 647	4
3 6	多目的暫存器 2 讀取	M R 2	-2, 147, 483, 648 ~ +2, 147, 483, 647	4
3 7	多目的暫存器 3 讀取	M R 3	-2, 147, 483, 648 ~ +2, 147, 483, 647	4
3 8	現在計時器值 讀取	C T	0 ~ 2, 147, 483, 647 [μ sec]	4
3 9	補間 · 終点最大值	T X	0 ~ 2, 147, 483, 646	4
3 A	現在螺旋回轉數	C H	0 ~ 65535	2
3 B	螺旋演算值	H V	1 ~ 2, 147, 483, 646	4
3 D	WR 1 設定值 讀取	W R 1	(位元資料)	2
3 E	WR 2 設定值 讀取	W R 2	(位元資料)	2
3 F	WR 3 設定值 讀取	W R 3	(位元資料)	2
4 0	多目的暫存器模式設定 讀取	M R M	(位元資料)	2
4 1	P I O 訊號設定 1 讀取	P 1 M	(位元資料)	2
4 2	P I O 訊號設定 2 · 其他設定 讀取	P 2 M	(位元資料)	2
4 3	加速度設定值 讀取	A C	1 ~ 536, 870, 911 [pps/sec]	4
4 4	初速度設定值 讀取	S V	1 ~ 8, 000, 000 [pps]	4
4 5	驅動速度設定值 讀取	D V	1 ~ 8, 000, 000 [pps]	4
4 6	移動脈衝數 / 終点設定值 讀取	T P	-2, 147, 483, 646 ~ +2, 147, 483, 646	4
4 7	分割脈衝設定 1 讀取	S P 1	分割脈衝長度 : 2 ~ 65, 535 脈衝寬度 : 1 ~ 65, 534	4
4 8	汎用輸入值	U I	RR7: 下位 1 位元組 (PIN) RR6: 下位 2 位元組 (I2C 通信時的 D15~0)	4

■ 驅動命令

編碼	命 令
5 0 h	相對位置驅動
5 1	反相對位置驅動
5 2	+方向連續脈衝驅動
5 3	-方向連續脈衝驅動
5 4	絕對位置驅動
5 6	驅動減速停止
5 7	驅動即停止
5 8	方向訊號+設定
5 9	方向訊號-設定
5 A	自動原点復歸執行

■ 補間命令

編碼	命 令
6 0 h	1 軸直線驅動(多晶片)
6 1	2 軸直線補間驅動
6 2	3 軸直線補間驅動
6 3	4 軸直線補間驅動
6 4	CW 圓弧補間驅動
6 5	CCW 圓弧補間驅動
6 6	2 軸位元補間驅動
6 7	3 軸位元補間驅動
6 8	4 軸位元補間驅動
6 9	CW 螺旋補間驅動
6 A	CCW 螺旋補間驅動
6 B	CW 螺旋演算
6 C	CCW 螺旋演算
6 D	減速有效
6 E	減速無效
6 F	補間中斷清除/補間單步

■ 同步動作操作命令

編碼	命 令
8 1 ~ 8 F h	同步動作 有效設定
9 1 ~ 9 F	同步動作 無效設定
A 1 ~ A F	同步動作 起動

■ 其他命令

編碼	命 令
7 0 h	速度增加
7 1	速度減少
7 2	偏差計數清除輸出
7 3	計時器始動
7 4	計時器停止
7 5	分割脈衝開始
7 6	分割脈衝停止
7 7	驅動開始保持
7 8	驅動開始釋放
7 9	錯誤・結束狀態清除
7 A	RR 3 0 頁顯示
7 B	RR 3 1 頁顯示
7 C	終点最大值清除
1 F	NO P
0 0 F F	指令重置

【注意】

- 這些以外的命令碼請勿寫入指令暫存器。有可能啟動IC內部回路的測試命令、造成無法意料的動作。

7.2 資料寫入命令

資料寫入命令為陪同寫入資料的命令。驅動時、加速度、驅動速度、移動脈衝數等的動作參數的設定。複數的軸指定時、相同資料可以指定軸全部全部同時設定。但是、補間・終點最大值、螺旋回轉數、螺旋演算值設定不需指定軸不要。

資料寫入命令在、指定的資料長有2位元組的時用 WR6 暫存器、資料長有4位元組的時 WR6,7 暫存器中設定數值。然後再用 WR0 暫存器指定軸及命令碼寫入。

WR6,7 寫入資料暫存器數值資料全部二進位。負的值以2的補數處理。

各々の資料請務必在資料範圍內的值設定。設定超出範圍外的值時動作會不正常。

【注意】

- 資料寫入命令的命令處理所要時間為最多 125nsec(CLK=16MHz 的時候)。命令寫入在此時間內請勿重複寫入命令。
- 加速計數偏移(AO)、理論位置計數最大值(LX)、實際位置計數最大值(RX)以外其他全部的動作參數在重置後為亂數。所以驅動時有關的參數在驅動前請務必設定適當值。
- 各速度參數值、計時器值記載的單位為輸入時脈(CLK)在 16MHz 的時候。輸入時脈(CLK)若不是 16MHz 時參數計算式請參考付錄B。

7.2.1 加速度增加率設定

命令碼	命令	參數記號	資料範圍	資料長(位元組)
0 0h	加速度增加率 設定	J K	1 ~ 1,073,741,823	4

加速度增加率設定值在S形加減速下加速度的單位時間的增加／減少率的參數。設定值的單位為 pps/sec²。

$$\text{加速度增加率} = JK \text{ [pps/sec}^2\text{]}$$

加速與減速對稱的S形加減速驅動(WR3/D1=0)時、減速時也使用此加速度增加率的值。

7.2.2 減速度增加率設定

命令碼	命令	參數記號	資料範圍	資料長(位元組)
0 1h	減速度增加率 設定	D J	1 ~ 1,073,741,823	4

減速度增加率設定值為加速及減速以非對稱S形加減速驅動(WR3/D1= 1)時減速度的單位時間減少率的參數。設定值的單位 pps/sec²。

$$\text{減速度增加率} = DJ \text{ [pps/sec}^2\text{]}$$

加速與減速為對稱S形加減速驅動(WR3/D1=0)時、減速度增加率的值不使用。

7.2.3 加速度 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
0 2h	加速度 設定	A C	1 ~ 536,870,911	4

直線加減速驅動的加速時的加速度參數。設定值的單位 pps/sec。

$$\text{加速度} = AC \text{ [pps/sec]}$$

加速及減速為對稱直線加減速驅動(WR3/D1=0)時、減速時也使用此加速度的值。

S形加減速驅動、此參數最大值 536,870,911 (1FFF FFFFh)。

部分S形加減速驅動、此參數作為直線加速部分的加速度設定。

加速與減速為對稱部分S形加減速驅動(WR3/D1=0)時、減速時也使用此加速度的值。

驅動中的現在加速度值可以由現在加減速度讀取命令(33h)讀取。

設定的加速度值可以由加速度設定值讀取命令(43h)讀取。

7.2.4 減速度 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
0 3h	減速度 設定	D C	1 ~ 536,870,911	4

非對稱直線加減速驅動(WR3/D1=1)時、減速時的減速度參數。設定值的單位 pps/sec。

$$\text{減速度} = DC \text{ [pps/sec]}$$

非對稱S形加減速驅動時、此參數最大值 536,870,911 (1FFF FFFFh)。

非對稱部分S形加減速驅動時、此參數做為直線減速部分的減速度設定。

7.2.5 初速度 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
0 4h	初速度 設定	S V	1 ~ 8,000,000	4

加減速驅動的加速開始的速度及減速結束時的速度。設定值的單位 pps。

$$\text{初速度} = SV \text{ [pps]}$$

對象馬達為步進馬達的時候、自起動周波數內的值設定。有機械的共振周波數的時候、請迴避設定為初速度。

定量脈衝驅動下、初速度極端低的值設定時、會有斷尾或拖拉的情況。

- 直線加減速驅動的時候以加速度設定值的平方根的值以上設定作目標。
- S形加減速驅動的時候以加速度增加率的平方根的 1/10 倍值以上設定作目標。
- 部分S形加減速驅動的時候以加速度設定值的平方根的值以上設定作目標。

$$\text{直線加減速驅動 } SV \geq \sqrt{AC}, \text{ S形加減速驅動 } SV \geq \sqrt{JK} \times 1/10, \text{ 部分S形加減速驅動 } SV \geq \sqrt{AC}$$

設定的初速度值可以用初速度設定值讀取命令(44h)讀取。

7.2.6 驅動速度 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
05h	驅動速度 設定	DV	1 ~ 8,000,000	4

加減速驅動到等速域時的速度。等速驅動時一開始為此速度。設定值的單位 pps。

$$\text{驅動速度} = DV \text{ [pps]}$$

此驅動速度設在初速度以下時不做加減速驅動、一開始就已等速驅動。編碼器的Z相搜尋等、低速驅動、訊號檢出後即停止的時候、驅動速度請設在初速度以下。

驅動速度在驅動途中可以自由變更。加減速度驅動的等速域驅動速度再設定時、開始往再設定的速度方向做加速或減速、再設定速度到達後再等速驅動。

補間驅動中以下狀況以外、本命令的驅動速度的不可以變更。詳細請參考、3.8.5。

- ・ 連續補間的途中的區段驅動速度變更時

自動原点復歸在此驅動速度為步驟1的高速檢出速度、及步驟4的高速移動速度。

【注意事項】

- S形加減速的定量脈衝驅動(自動減速模式時)、及非對稱直線加減速的定量脈衝驅動(自動減速模式時)、驅動途中驅動速度的變更不可。
- S形加減速的連續脈衝驅動在等速域時驅動速度可以變更、加減速中的驅動速度變更設定無效。
- 對稱直線加減速的定量脈衝驅動、驅動途中驅動速度可以變更、但因變更頻度太頻繁而有若干斷尾、拖拉發生的可能。

驅動中的現在驅動速度值可以用現在驅動速度讀取命令(32h)讀取。

設定後的驅動速度值可以用驅動速度設定值讀取命令(45h)讀取。

7.2.7 移動脈衝數／終點 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
06h	移動脈衝數／終點 設定	TP	-2,147,483,646 ~ +2,147,483,646	4

相對位置驅動、設定現在位置起的移動脈衝數。移動脈衝數正的脈衝數設定時驅動方向為+方向、負的脈衝數設定時驅動方向-方向。

反相對位置驅動時移動脈衝數正的脈衝數指定後、驅動方向為-方向。

絕對位置驅動以原点(理論位置計數=0)為基準設定帶符號 32 位元值的移動目標終點。

移動脈衝數可以在相對位置驅動及反相對位置驅動的途中變更。但是引起驅動方向改變的值不能變更。另外、已通過的位置變更時、驅動即停止。

絕對位置驅動的途中、終點不能變更。

直線、圓弧補間驅動設定各軸的終點。終點座標以帶符號 32 位元相對於現在位置的值設定。

位元補間下設定各軸的位元資料。位元資料在 32 位元的中下位 16 位元為+方向的位元資料、上位 16 位元為-方向的位元資料設定。

螺旋補間驅動時設定 X、Y 軸的終點及 Z、U 軸的移動量。終點座標為 32 位元相對於現在位置的帶符號相對值。移動量為回轉數 0 的時候的總移動量、1 以上的時候為 1 回轉時的移動量以帶符號值設定。

7.2.8 手動減速點 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
07h	手動減速點 設定	DP	0 ~ 4,294,967,292	4

手動減速模式(WR3/D0=1)的加減速定量脈衝驅動、手動減速點設定。手動減速點為、定量脈衝驅動的輸出脈衝數與減速所要消費脈衝數的值設定。

$$\text{手動減速點} = \text{輸出脈衝數} - \text{減速消費脈衝數}$$

<輸出脈衝數>

輸出脈衝數為定量脈衝驅動實際輸出脈衝數。

相對位置驅動時、輸出脈衝數 P 為移動脈衝數 TP 設定值的絕對值。

絕對位置驅動時、輸出脈衝數 P 為移動脈衝數 TP 設定值減去驅動開始前的理論位置計數值 LP 值的絕對值。

$$\text{相對位置驅動：輸出脈衝數 } P = | TP |$$

$$\text{絕對位置驅動：輸出脈衝數 } P = | TP - LP |$$

7.2.9 円弧中心點 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
08h	円弧中心點 設定	CT	-2,147,483,646 ~ +2,147,483,646	4

円弧、螺旋補間驅動的中心點設定。中心座標為現在位置的帶符號相對值設定。

7.2.10 理論位置計數 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
09h	理論位置計數 設定	LP	-2,147,483,648 ~ +2,147,483,647	4

理論位置計數的值設定。

理論位置計數為、+方向/-方向的驅動輸出脈衝增/減計數。

理論位置計數的值隨時可以寫入。隨時可以理論位置計數讀取命令(30h)讀取。

7.2.11 實際位置計數 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
0Ah	實際位置計數 設定	RP	-2,147,483,648 ~ +2,147,483,647	4

實際位置計數的值設定。

實際位置計數為編碼器輸入脈衝增／減計數。

實際位置計數的值隨時可以寫入。隨時可以實際位置計數讀取命令(31h)讀取。

7.2.12 軟體極限+ 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
0 Bh	軟體極限+ 設定	S P	-2, 147, 483, 648 ~ +2, 147, 483, 647	4

+方向軟體極限 SLMT+暫存器的值設定。
軟體極限的有効／無効、設定對象、停止模式的設定以 WR2 暫存器設定。

軟體極限 SLMT+暫存器的值隨時可以寫入。

7.2.13 軟體極限- 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
0 Ch	軟體極限- 設定	S M	-2, 147, 483, 648 ~ +2, 147, 483, 647	4

-方向軟體極限 SLMT-暫存器的值設定。
軟體極限的有効／無効、設定對象、停止模式的設定的設定以 WR2 暫存器設定。

軟體極限 SLMT-暫存器的值隨時可以寫入。

7.2.14 加速計數偏移 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
0 Dh	加速計數偏移 設定	A O	-32, 768 ~ +32, 767	2

加速計數的偏移值設定。
加速計數的偏移值重置時為 0。通常不需變更。

有關加速計數偏移請參考 2.1 節的 C. 項。

本資料寫入命令的資料長 2 位元組。設定值僅 WR6 暫存器寫入。

7.2.15 理論位置計數最大值 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
0 Eh	理論位置計數最大值 設定	L X	1 ~ 2, 147, 483, 647 (7FFF FFFFh) 或FFFF FFFFh	4

理論位置計數的可變連接機能下、理論位置計數最大值的正值設定。

重置時的值為 FFFF FFFFh。可變連接機能不使用時初始值的不用改變。

7.2.16 實際位置計數最大值 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
0 Fh	實際位置計數最大值 設定	R X	1 ~ 2, 147, 483, 647 (7FFF FFFFh) 或FFFF FFFFh	4

實際位置計數的可變連接機能下、實際位置計數最大值以正的值設定。

重置時的值 FFFF FFFFh。可變連接機能不使用時初始值的不用改變。

7.2.17 多目的暫存器0 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
1 0h	多目的暫存器0 設定	M R 0	-2, 147, 483, 648 ~ +2, 147, 483, 647	4

多目的暫存器 MR0 的值設定。

多目的暫存器、位置、速度、計時器的值做大小比較、同步動作時各種參數值的載入／儲存等使用。比較結果做比較輸出訊號的輸出及、同步動作起動及中斷發生使用。

多目的暫存器 MR0 的值隨時可以寫入。隨時可以多目的暫存器0讀取命令(34h)讀取。

7.2.18 多目的暫存器1 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
1 1h	多目的暫存器1 設定	M R 1	-2, 147, 483, 648 ~ +2, 147, 483, 647	4

多目的暫存器 MR1 的值設定。

多目的暫存器、位置、速度、計時器的值做大小比較、同步動作時各種參數值的載入／儲存等使用。比較結果做比較輸出訊號的輸出及、同步動作起動及中斷發生使用。

多目的暫存器 MR1 的值隨時可以寫入。隨時可以多目的暫存器1讀取命令(35h)讀取。

7.2.19 多目的暫存器2 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
1 2h	多目的暫存器2 設定	M R 2	-2, 147, 483, 648 ~ +2, 147, 483, 647	4

多目的暫存器 MR2 的值設定。

多目的暫存器、位置、速度、計時器的值做大小比較、同步動作時各種參數值的載入／儲存等使用。比較結果做比較輸出

訊號的輸出及、同步動作起動及中斷發生使用。

多目的暫存器 MR2 的值隨時可以寫入。隨時可以多目的暫存器2讀取命令 (36h) 讀取。

7.2.20 多目的暫存器3 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
1 3 h	多目的暫存器3 設定	MR 3	-2, 147, 483, 648 ~ +2, 147, 483, 647	4

多目的暫存器 MR3 的值設定。

多目的暫存器、位置、速度、計時器的值做大小比較、同步動作時各種參數值的載入／儲存等使用。比較結果做比較輸出訊號的輸出及、同步動作起動及中斷發生使用。

多目的暫存器 MR3 的值隨時可以寫入。隨時可以多目的暫存器3讀取命令 (37h) 讀取。

7.2.21 原点檢出速度 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
1 4 h	原点檢出速度 設定	H V	1 ~ 8,000,000	4

自動原点復歸的步驟2, 3的低速搜尋速度設定。設定值的單位 pps。

$$\text{原点檢出速度} = HV \text{ [pps]}$$

檢出訊號動作時即停止之故、設定值需比初速度 (SV) 低。

自動原点復歸請參考 2.6 節。

7.2.22 速度增減值 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
1 5 h	速度增減值 設定	I V	1 ~ 1,000,000	4

依據速度增加命令 (70h)、速度減少命令 (71h) 設定、驅動中的現在速度做增減速度值設定。設定值的單位 pps。

$$\text{速度增減值} = IV \text{ [pps]}$$

加減速驅動的等速域速度增加／減少命令發行後、此速度增減值分因增加／減少而往新速度做加速或減速、速度到達後再等速驅動。

7.2.23 計時器值 設定

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
1 6 h	計時器值 設定	T M	1 ~ 2, 147, 483, 647	4

計時器的計時時間設定。設定值的單位 μsec 。

$$\text{計時器值} = TM [\mu sec]$$

計時器動作中的現在計時器值隨時可由現在計時器值讀取命令(38h)讀取。

7.2.24 分割脈衝設定 1

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
1 7 h	分割脈衝設定 1	S P 1	WR6 分割脈衝長度 : 2 ~ 65, 535	4
			WR7 脈衝寬度 : 1 ~ 65, 534	

分割脈衝的分割脈衝長度與脈衝寬度設定。分割脈衝長度、脈衝寬度為驅動脈衝單位。
寫入資料在 WR6 設定分割脈衝長度、WR7 設定脈衝寬度。

分割脈衝長度、脈衝寬度在分割脈衝輸出中可以變更。分割脈衝長度、脈衝寬度再設定後、再設定的值、分割脈衝繼續輸出。

本資料寫入命令的資料長為 4 位元組、分割脈衝長度與脈衝寬度的任一方的值變更，請寫入時都務必對 WR6 及 WR7 的兩暫存器寫入適當值。

設定後分割脈衝設定 1 (SP1) 的值、可以由分割脈衝設定 1 讀取命令(47h)讀取。

7.2.25 分割脈衝設定 2

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
1 8 h	分割脈衝設定 2	S P 2	分割脈衝數 : 0, 1 ~ 65, 535	2

輸出分割脈衝數設定。分割脈衝數設為 0 時、依據指令或同步動作控制分割脈衝輸出停止前、持續輸出分割脈衝。

分割脈衝數、分割脈衝輸出中不能變更。

本資料寫入命令的資料長為 2 位元組。設定值從 WR6 暫存器寫入。

7.2.26 補間・終点最大値

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
1 9h	補間・終点最大値	T X	1 ~ 2,147,483,646	4

直線補間時的終点的最大値設定。本命令軸指定不要。終点最大値為無符号無 32 位元值。本命令設定值作為直線補間的演算用。

本命令使用時、以補間模式設定命令(2Ah) 人工設定直線補間最大値。

7.2.27 螺旋回轉數

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
1 Ah	螺旋回轉數	H N	0~65, 535	2

螺旋補間時的螺旋回轉數設定。本命令軸指定不要。

螺旋補間驅動中的現在螺旋回轉數可由現在螺旋回轉數讀取命令(3Ah)讀取。

7.2.28 螺旋演算值

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
1 Ah	螺旋演算值	H V	1 ~ 2, 147, 483, 646	4

螺旋補間時的螺旋演算值設定。本命令軸指定不要。

進行螺旋補間時、本 IC 需設定螺旋演算值。
本 IC 螺旋演算值的設定步驟如下所示。

- ① 螺旋演算命令(6Bh,6Ch) 執行螺旋演算
- ② 演算結果以螺旋演算值讀取命令(3Bh)讀取
- ③ 讀取值用本命令設定

螺旋補間有關說明請參考、3.3 節。

7.3 模式寫入命令

模式寫入命令為隨同寫入資料的命令。多目的暫存器、自動原点復歸、同步動作、補間驅動等動作模式的設定。複數的軸指定時、相同資料可以指定軸全部同時設定。但是、補間模式設定軸指定不要。

模式寫入命令、全部寫入資料長為2位元組。WR6暫存器的各位元將適當值設定、WR0暫存器命令碼寫入後、WR6暫存器的內容就會設定到IC內部的各模式設定暫存器。

重置時IC內部的各模式設定暫存器的全部位元清為0。

【注意】

- 資料寫入命令的命令處理所需時間、最多125nsec (CLK=16MHz的時候)。命令寫入在此時間內請勿重複寫入。

7.3.1 多目的暫存器模式設定

命令碼	命令	參數記號	資料長(位元組)
2 0h	多目的暫存器模式設定	M R M	2

多目的暫存器 MR3~0 的值比較對象、及比較條件的設定。MR3~0 的可以個別設定比較對象、比較條件。比較結果做為、同步動作起動要因、中斷發生要因、比較訊號輸出使用。



D1, 0 MOT1, 0 MR0 的比較對象。

(n:0~3)

D3, 2 M0C1, 0 MR0 的比較條件。

MnT1 位元	MnT0 位元	MRn 比較對象
0	0	理論位置計數 (LP)
0	1	實際位置計數 (RP)
1	0	現在速度值 (CV)
1	1	現在計時器值 (CT)

D9, 8 M2T1, 0 MR2 的比較對象。

(n:0~3)

D11, 10 M2C1, 0 MR2 的比較條件。

MnC1 位元	MnC0 位元	MRn 比較條件
0	0	比較對象 ≥ MRn
0	1	比較對象 > MRn
1	0	比較對象 = MRn
1	1	比較對象 < MRn

D13, 12 M3T1, 0 MR3 的比較對象。

D15, 14 M3C1, 0 MR3 的比較條件。

多目的暫存器模式設定所設定的比較條件 (MnC1, 0 位元)、MR3~0 的比較對象的大小比較結果可以由 RR4 暫存器確認。多目的暫存器、詳細請參考 2.5 節。

【注意】

比較對象為「現在速度值 (CV)」、比較條件為「比較對象 = MRn」設定時、加減速驅動在加速度、減速度超過 4,194,304 (400000h)pps/sec 時、比較結果會不動作。

比較對象為「現在速度 (CV)」在加速度、減速度超過此值以上時、比較條件請不要用「比較對象 = MRn」、改用「比較對象 ≥ MRn」等其他條件使用。

重置時、D15~D0、全部歸0。

7.3.2 PIO訊號設定1

命令碼	命 令	參數記號	資料長(位元組)
2 1h	PIO訊號設定1	P1M	2

nPIO7~0 訊號的機能設定。nPIO7~0 訊號可在、汎用輸出入訊號、同期輸入訊號、同步脈衝輸出訊號、驅動狀態輸出訊號、MRn 比較輸出訊號、外部訊號驅動操作的輸入訊號使用。

WR6	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
	P7M1	P7M0	P6M1	P6M0	P5M1	P5M0	P4M1	P4M0	P3M1	P3M0	P2M1	P2M0	P1M1	P1M0	P0M1	P0M0
	PIO7信号		PIO6信号		PIO5信号		PIO4信号		PIO3信号		PIO2信号		PIO1信号		PIO0信号	

D1, 0 P0M1, 0 nPIO0 訊號的機能設定。

D3, 2 P1M1, 0 nPIO1 訊號的機能設定。

D5, 4 P2M1, 0 nPIO2 訊號的機能設定。

D7, 6 P3M1, 0 nPIO3 訊號的機能設定。

D9, 8 P4M1, 0 nPIO4 訊號的機能設定。

D11, 10 P5M1, 0 nPIO5 訊號的機能設定。

D13, 12 P6M1, 0 nPIO6 訊號的機能設定。

D15, 14 P7M1, 0 nPIO7 訊號的機能設定。

各機能如下表所示。

		(n:0~7)	
PnM1 位元	PnM0 位元	機能	
0	0	汎用輸入 nPIO7~0 訊號為輸入狀態。訊號位準、X 軸在 RR4 暫存器的 D7~0、Y 軸在 RR4 暫存器的 D15~8 讀取。同樣的、Z 軸在 RR5 暫存器的 D7~0、U 軸在 D15~8 讀取。 同步動作時、訊號的↑及↓時可以起動同步動作。 外部驅動操作時、依據 nPIO4, 5 訊號的狀態可以起動相對位置驅動及連續脈衝驅動。	
0	1	汎用輸出 nPIO7~0 訊號為輸出狀態。WR4 暫存器的 D7~0 在 X 軸的 PIO7~0、D15~8 在 Y 軸的 PIO7~0 輸出。同樣、WR5 暫存器的 D7~0 在 Z 軸的 PIO7~0、D15~8 在 U 軸的 PIO7~0 輸出。值為 0 Low 位準、1 為 Hi 位準輸出。	
1	0	驅動狀態輸出 nPIO7~0 訊號的輸出狀態、各訊號為下表所示驅動狀態輸出。	
1	1	同步脈衝・MRn 比較輸出 nPIO7~0 訊號為輸出狀態。nPIO3~0 為同步脈衝、nPIO7~4 為 MRn 比較值輸出。比較對象與比較條件再多目的暫存器模式設定命令(20h)設定	

各 nPIOm 訊號的機能如下表所示。

(n:0~7)				
nPIOm 訊號	PnM1, 0 = 0, 0	PnM1, 0 = 0, 1	PnM1, 0 = 1, 0	PnM1, 0 = 1, 1

(端子番号)	汎用輸入 [*注]	汎用輸出	驅動狀態輸出 (真で Hi)	同步脈衝輸出, MRn 比較輸出
XP100 (63) YP100 (82) ZP100 (101) UP100 (120)	X 軸 : RR4/D0 Y 軸 : RR4/D8 Z 軸 : RR5/D0 U 軸 : RR5/D8 訊號位準讀取	X 軸 : WR4/D0 Y 軸 : WR4/D8 Z 軸 : WR5/D0 U 軸 : WR5/D8 值輸出	驅動中	SYNC0 同步脈衝輸出
XP101 (62) YP101 (81) ZP101 (100) UP101 (119)	X 軸 : RR4/D1 Y 軸 : RR4/D9 Z 軸 : RR5/D1 U 軸 : RR5/D9 訊號位準讀取	X 軸 : WR4/D1 Y 軸 : WR4/D9 Z 軸 : WR5/D1 U 軸 : WR5/D9 值輸出	錯誤発生	SYNC1 同步脈衝輸出
XP102 (61) YP102 (80) ZP102 (99) UP102 (118)	X 軸 : RR4/D2 Y 軸 : RR4/D10 Z 軸 : RR5/D2 U 軸 : RR5/D10 訊號位準讀取	X 軸 : WR4/D2 Y 軸 : WR4/D10 Z 軸 : WR5/D2 U 軸 : WR5/D10 值輸出	加速中	SYNC2 同步脈衝輸出
XP103 (60) YP103 (79) ZP103 (98) UP103 (117)	X 軸 : RR4/D3 Y 軸 : RR4/D11 Z 軸 : RR5/D3 U 軸 : RR5/D11 訊號位準讀取	X 軸 : WR4/D3 Y 軸 : WR4/D11 Z 軸 : WR5/D3 U 軸 : WR5/D11 值輸出	等速中	SYNC3 同步脈衝輸出
XP104 (59) YP104 (78) ZP104 (97) UP104 (116)	X 軸 : RR4/D4 Y 軸 : RR4/D12 Z 軸 : RR5/D4 U 軸 : RR5/D12 訊號位準讀取	X 軸 : WR4/D4 Y 軸 : WR4/D12 Z 軸 : WR5/D4 U 軸 : WR5/D12 值輸出	減速中	MRO 比較輸出 (真 Hi)
XP105 (58) YP105 (77) ZP105 (96) UP105 (115)	X 軸 : RR4/D5 Y 軸 : RR4/D13 Z 軸 : RR5/D5 U 軸 : RR5/D13 訊號位準讀取	X 軸 : WR4/D5 Y 軸 : WR4/D13 Z 軸 : WR5/D5 U 軸 : WR5/D13 值輸出	加速度增加中	MR1 比較輸出 (真 Hi)
XP106 (57) YP106 (76) ZP106 (95) UP106 (114)	X 軸 : RR4/D6 Y 軸 : RR4/D14 Z 軸 : RR5/D6 U 軸 : RR5/D14 訊號位準讀取	X 軸 : WR4/D6 Y 軸 : WR4/D14 Z 軸 : WR5/D6 U 軸 : WR5/D14 值輸出	加速度一定中	MR2 比較輸出 (真 Hi)
XP107 (56) YP107 (75) ZP107 (94) UP107 (113)	X 軸 : RR4/D7 Y 軸 : RR4/D15 Z 軸 : RR5/D7 U 軸 : RR5/D15 訊號位準讀取	X 軸 : WR4/D7 Y 軸 : WR4/D15 Z 軸 : WR5/D7 U 軸 : WR5/D15 值輸出	加速度減少中	MR3 比較輸出 (真 Hi)

nPIO7~0 訊號的詳細使用方法請參考、2.9 節。

*注 nPIO7~0 訊號作為汎用輸入模式 (PnM1,0 = 0,0) 時、可以使用在同步動作的起動要因。詳細請參考 2.7 節。
nPIO4,5 訊號作為汎用輸入模式 (PnM1,0 = 0,0) 時、可以用外部訊號來做驅動操作的輸入訊號 (EXPP, EXPM 輸入)。
詳細請參考 2.13.1 項。

重置時、D15~D0 全部 0。

7.3.3 PIO訊號設定2・其他設定

命令碼	命 令	參數記號	資料長(位元組)
-----	-----	------	----------

2 2 h	P I O 訊號設定 2 · 其他設定	P 2 M	2
-------	---------------------	-------	---

同步脈衝輸出的理論、脈衝寬度設定。另外、錯誤發生時時設定同步動作無效、由外部訊號來控制驅動操作的模式設定、及分割脈衝輸出的理論、開始脈衝有無的設定。



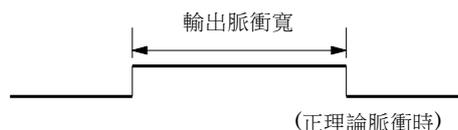
D3~0 PnL nPIOm(m:3~0)作為同步脈衝輸出訊號使用時、脈衝的理論設定。
 0:正理論脈衝、1:負理論脈衝



D6~4 PW2~0 同步脈衝輸出訊號的輸出脈衝寬度設定。

(CLK=16MHz 時)

D6~4 (PW2~0)	輸出脈衝寬度
0	125 n sec
1	312 n sec
2	1 μ sec
3	4 μ sec
4	16 μ sec
5	64 μ sec
6	256 μ sec
7	1msec



D7 ERRDE 同步動作 SYNC3~0 的有効狀態、錯誤發生 (RR0/D7~4:n-ERR = 1)時無効否的設定。
 0:錯誤時有効、1:錯誤時無効
 此位元1設定時、RR0 暫存器的 n-ERR 位元若變為 1、同步動作 SYNC3~0 立刻全部無効設定。

同步動作要再設為有効時、RR0 暫存器的 n-ERR 位元若為 1、同步動作 SYNC3~0 無法設定為有効。
 請以錯誤・結束狀態清除命令 (79h)將 n-ERR 位元清除後再設定同步動作有効。
 錯誤發生狀況、及同步動作 SYNC3~0 的有効/無効設定狀況可由 RR3 暫存器 頁 1 確認。

D9, 8 EXOP1, 0 外部輸入訊號 (EXPP, EXPM) 做驅動操作模式設定。

D9 (EXOP1)	D8 (EXOP0)	外部訊號驅動操作模式
0	0	外部訊號驅動操作無効
0	1	連續脈衝驅動模式
1	0	相對位置驅動模式
1	1	手動脈衝口生器模式

D10 SPLL 分割脈衝輸出的脈衝理論設定。
 0:正理論脈衝、1:負理論脈衝



D11 SPLBP 分割脈衝輸出的開始脈衝有無設定。
 0:開始脈衝無、1:開始脈衝有

重置時、D15～D0 全部0。D15～D12 位元請隨時設為0。

7.3.4 自動原点復歸模式設定 1

命令碼	命 令	參數記號	資料長(位元組)
2 3 h	自動原点復歸模式設定 1	H 1 M	2

自動原点復歸的動作模式設定。自動原点復歸各步驟的有效／無效、檢出方向、停止訊號的選擇、偏差計數清除訊號輸出的有效／無效、位置計數清除的設定。

WR6	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4	D3 ^L	D2	D1	D0
		S4EN	S3LC	S3RC	S3DC	S3DR	S3EN	S2LC	S2RC	S2DC	S2SG	S2DR	S2EN	S1G1	S1G0	S1DR
	步驟 4				步驟 3				步驟 2				步驟 1			

D0 S1EN 自動原点復歸步驟1的動作「高速原点搜尋」執行否的設定。
0:不執行、1:執行

D1 S1DR 步驟1的檢出方向設定。
0:+方向、1:-方向

D3, 2 S1G1, 0 步驟1的檢出訊號設定。
檢出輸入訊號的理論設定在 WR2 暫存器進行。

D3 (S1G1)	D2 (S1G0)	檢出訊號
0	0	nSTOP0
0	1	nSTOP1
1	0	極限訊號 *
1	1	(設定不可)

* 極限訊號指定時、D1 (S1DR) 指定的檢出方向側的選擇極限訊號。

D4 S2EN 自動原点復歸步驟2的動作「低速原点搜尋」執行否的設定。
0:不執行、1:執行

D5 S2DR 設定步驟2的檢出方向。
0:+方向、1:-方向

D6 S2SG 步驟2的檢出訊號設定。
檢出輸入訊號的理論設定在 WR2 暫存器進行。

D6 (S2SG)	檢出訊號
0	nSTOP1
1	極限訊號 *

* 極限訊號指定時、D5 (S2DR) 指定的檢出方向側的選擇極限訊號。

D7 S2DC 步驟2的訊號檢出、偏差計數清除 (nDCC) 訊號輸出否的設定。
0:不輸出、1:輸出

D8 S2RC 步驟2的訊號檢出下、實際位置計數清除否的設定。
0:不清除、1:清除

D9 S2LC 步驟2的訊號檢出、理論位置計數清除否的設定。
0:不清除、1:清除

D10	S3EN	自動原点復歸步驟3的動作「低速Z相搜尋」執行否的設定。 0:不執行、1:執行
D11	S3DR	步驟3的檢出方向設定。 0:+方向、1:-方向
D12	S3DC	步驟3的 nSTOP2 訊號、偏差計數清除 (nDCC) 訊號輸出否的設定。 0:不輸出、1:輸出
D13	S3RC	步驟3的 nSTOP2 訊號檢出、實際位置計數清除否的設定。 0:不清除、1:清除
D14	S3LC	步驟3的 nSTOP2 訊號檢出、理論位置計數清除否的設定。 0:不清除、1:清除
D15	S4EN	步驟4的動作「高速偏移移動」執行否的設定。 0:不執行、1:執行

自動原点復歸的詳細請參考 2.6 節、及 2.6.4 項。

重置時、D15~D0、全部0。

7.3.5 自動原点復歸模式設定2

命令碼	命 令	參數記號	資料長(位元組)
2 4 h	自動原点復歸模式設定 2	H 2 M	2

自動原点復歸的動作模式設定。自動原点復歸步驟 3 的停止條件、位置計數清除、偏差計數清除輸出、步驟間計時器的設定。

WR 6	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
	0	0	0	0	0	HTM2	HTM1	HTM0	HTME	DCP2	DCP1	DCP0	DCPL	LCLR	RCLR	SAND
									<div style="display: flex; justify-content: space-around; width: 100%;"> 步驟間計時 偏差計數器清除輸出 </div>							

D0	SAND	1設定後、步驟3動作在 nSTOP1 訊號動作後和 nSTOP2 訊號動作變化時停止。 僅步驟2的檢出訊號選取 nSTOP1 訊號時有效。選取極限訊號時無法設有効。
D1	RCLR	自動原点復歸結束時實際位置計數清除否的設定。 0:不清除、1:清除
D2	LCLR	自動原点復歸結束時理論位置計數清除否的設定。 0:不清除、1:清除
D3	DCPL	偏差計數清除 (nDCC) 輸出脈衝的理論設定。 0:正理論脈衝、1:負理論脈衝

正理論脈衝



負理論脈衝 :



D6~4 DCP2~0 偏差計數清除 (nDCC) 輸出的脈衝寬度設定。

(CLK=16MHz 時)	
D6~4 (DCP2~0)	輸出脈衝寬度
0	10 μ sec
1	20 μ sec
2	100 μ sec
3	200 μ sec
4	1msec
5	2msec
6	10msec
7	20msec



D7 HTME 步驟間計時器有效。
0:無効、1:有效

D10~8 HTM2~0 步驟間計時器的時間指定。

(CLK=16MHz 時)	
D10~8 (HTM2~0)	計時器時間
0	1msec
1	2msec
2	10msec
3	20msec
4	100msec
5	200msec
6	500msec
7	1000msec

自動原点復歸的詳細請參考 2.6 節、及 2.6.4 項。

重置時、D15~D0 全部0。D15~D11 位元請隨時保持為0設定。

7.3.6 輸入訊號濾波模式 設定

命令碼	命 令	參數記號	資料長(位元組)
2 5 h	輸入訊號濾波模式設定 2	F L M	2

輸入訊號濾波的有効／無効、及 2 個濾波的時間常數設定。

WR6	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
		FL13	FL12	FL11	FL10	FL03	FL02	FL01	FL00	FE7	FE6	FE5	FE4	FE3	FE2	FE1

濾波時間常數 B 濾波時間常數 A ; 各輸入訊號濾波有效/無效

D7~0 FE7~0 下表輸入訊號在、IC內藏的濾波機能有効或無効(通過)設定。
0:無効、1:有効

指定位元	輸入訊號	適用時間常數
D0 (FE0)	EMGN	濾波時間常數 A
D1 (FE1)	nLMTP, nLMTM	
D2 (FE2)	nSTOP0, nSTOP1	
D3 (FE3)	nINPOS, nALARM	
D4 (FE4)	nPI03~0	
D5 (FE5)	nPI07~4	濾波時間常數 B
D6 (FE6)	nSTOP2	
D7 (FE7)	nECA, nECB	

D11~8 FL03~00 濾波時間常數A在 D5~D0 (FE5~0) 指定的輸入訊號濾波的時間常數設定。

D15~12 FL13~10 濾波時間常數B在 D7, D6 (FE7, 6) 指定的輸入訊號濾波的時間常數設定。

(CLK=16MHz 時)

時間常數 (Hex)	可消除的最大雜訊寬度	輸入訊號遲延時間
0	437.5 nsec	500 nsec
1	875 nsec	1 μsec
2	1.75 μsec	2 μsec
3	3.5 μsec	4 μsec
4	7 μsec	8 μsec
5	14 μsec	16 μsec
6	28 μsec	32 μsec
7	56 μsec	64 μsec
8	112 μsec	128 μsec
9	224 μsec	256 μsec
A	448 μsec	512 μsec
B	896 μsec	1.024msec
C	1.792msec	2.048msec
D	3.584msec	4.096msec
E	7.168msec	8.192msec
F	14.336msec	16.384msec

輸入訊號 EXPLSN、PIN7~0 無濾波機能。
輸入訊號濾波機能的詳細請參考 2.12 節。

重置時、D15~D0 全部0。

7.3.7 同步動作 SYNC0, 1, 2, 3 設定

命令碼	命令	參數記號	資料長(位元組)
26h	同步動作 SYNC0 設定	S0M	2
27h	同步動作 SYNC1 設定	S1M	2
28h	同步動作 SYNC2 設定	S2M	2
29h	同步動作 SYNC3 設定	S3M	2

同步動作 SYNC0, 1, 2, 3 的動作模式設定。各同步動作組的起動要因、動作、他同步動作組起動、他軸 SYNC0 起動、同步動作的單一／重複的設定。



D3~0 PRV3~0 同步動作的起動要因編碼指定。

(m : 0, 1, 2, 3)

編碼 (Hex)	SYNCn 起動要因	編碼 (Hex)	SYNCn 起動要因
0	NOP	8	分割脈衝結束
1	MRn 比較為變為真	9	分割脈衝輸出
2	計時器的時間到	A	nPIOm 輸入 ↑
3	驅動開始	B	nPIOm 輸入 ↓
4	驅動等速域開始	C	nPIO(m+4) 輸入 Low 和 nPIOm 輸入 ↑
5	驅動等速域結束	D	nPIO(m+4) 輸入 Hi 和 nPIOm 輸入 ↑
6	驅動結束	E	nPIO(m+4) 輸入 Low 和 nPIOm 輸入 ↓
7	分割脈衝開始	F	nPIO(m+4) 輸入 Hi 和 nPIOm 輸入 ↓

同步動作的起動要因、設定編碼的詳細請參考、2.7.1 項。

D8~4 ACT4~0 同步動作的動作編碼指定。

(n : 0, 1, 2, 3)

編碼 (Hex)	SYNCn 動作	編碼 (Hex)	SYNCn 動作
00	NOP	0C	絕對位置驅動起動
01	載入 MRn → DV	0D	+方向連續脈衝驅動起動
02	載入 MRn → TP	0E	-方向連續脈衝驅動起動
03	載入 MRn → SP1	0F	MRn 值的移動脈衝數相對位置驅動
04	載入 MRn → LP (SYNC0), RP (SYNC1), SV (SYNC2), AC (SYNC3)	10	MRn 值的終點絕對位置驅動
		11	驅動減速停止
05	儲存 LP → MRn	12	驅動即停止
06	儲存 RP → MRn	13	驅動速度增加
07	儲存 CT → MRn	14	驅動速度減少
08	儲存 CV (SYNC0), CA (SYNC1) → MRn	15	計時器始動
09	同步脈衝 nPIOm 輸出	16	計時器停止
0A	相對位置驅動起動	17	分割脈衝開始
0B	反相對位置驅動起動	18	分割脈衝停止

DV : 驅動速度	TP : 移動脈衝數/終點	SP1 : 分割脈衝設定 1
LP : 理論位置計數	RP : 實際位置計數	SV : 初速度
AC : 加速度	CT : 現在計時器值	CV : 現在驅動速度
CA : 現在加速度		

同步動作的動作、設定編碼的詳細請參考、2.7.2 項。

D11~9 SNC+3~1 同步動作起動時、連同其他的同步動作啟動的設定。

0:無効、1:有効

同步動作組	D11 (SNC+3)	D10 (SNC+2)	D9 (SNC+1)
SYNC0	SYNC3 起動	SYNC2 起動	SYNC1 起動
SYNC1	SYNC0 起動	SYNC3 起動	SYNC2 起動
SYNC2	SYNC1 起動	SYNC0 起動	SYNC3 起動
SYNC3	SYNC2 起動	SYNC1 起動	SYNC0 起動

D14~13 AXIS3~1 同步動作的起動、指定由其他軸 SYNC0 來啟動。

0:無効、1:有効

自軸	D14 (AXIS3)	D13 (AXIS2)	D12 (AXIS1)
X	U 軸 SYNC0 起動	Z 軸 SYNC0 起動	Y 軸 SYNC0 起動
Y	X 軸 SYNC0 起動	U 軸 SYNC0 起動	Z 軸 SYNC0 起動
Z	Y 軸 SYNC0 起動	X 軸 SYNC0 起動	U 軸 SYNC0 起動
U	Z 軸 SYNC0 起動	Y 軸 SYNC0 起動	X 軸 SYNC0 起動

D15 REP 同步動作組的有効狀態、同步動作一旦起動後無効否的設定。

0:無効(執行一次)、1:有効(重複)

此位元 0 時、起動要因動作後啟動 1 次同步動作。此位元 1 的時候、起動要因動作時重複同步動作起動。

無効後的同步動作要再有効時、請從新發行同步動作有効命令。同步動作 SYNC3~0 的有効/無効設定狀況可由 RR0 暫存器確認。

同步動作的詳細請參考、2.7 節。

重置時、D15~D0 全部 0。

7.3.8 補間模式 設定

命令碼	命 令	參數記號	資料長(位元組)
2 Ah	補間模式設定	I P M	2

補間驅動時的模式設定。本命令軸指定不要。

WR6	D15	D14	D13	D12 ^H	D11	D10	D9	D8	D7	D6	D5	D4 ^L	D3	D2	D1	D0
	INTB	INTA	0	MAXM	MLT1	MLT0	STEP	LMDF	SPD1	SPD0	0	CXIV	U-EN	Z-EN	Y-EN	X-EN

D3~0 U-EN~X-EN 補間驅動軸設定。下表所示位元為對應補間軸。

0:補間軸不使用、1:補間軸使用

軸	指定位元
X	D0
Y	D1
Z	D2
U	D3

主軸為X-EN > Y-EN > Z-EN > U-EN 的優先順、從設定1的位元的軸選取。

D4 CXIV 円弧補間進行時、指定補間軸反轉補間。

0:補間軸不反轉、1:補間軸反轉

D7,6 SPD1,0 補間驅動的線速一定模式設定。

D7 (SPD1)	D6 (SPD0)	線速一定的種類
0	0	無効
0	1	2軸簡易
1	0	3軸簡易
1	1	2軸高精度

D8 LMDF 補間驅動的短軸脈衝均一化模式設定。

0:無効、1:有効

D9 STEP 補間驅動的外部訊號・命令單步進給設定。

0:無効、1:有効

此位元1的時候、補間驅動由外部訊號(EXPLSN)，或是補間步驟命令(6Fh)做單步進給的模式。

D11,10 MLT1,0 多晶片補間設定。

D11 (MLT1)	D10 (MLT0)	多晶片補間的動作
0	0	無効
0	1	主晶片
1	0	副晶片
1	1	設定不可

多晶片補間時、主晶片設為01、其他(副晶片)設為10。

多晶片補間不使用時設為00。

D12 MAXM 直線補間最大值的設定方法指定。

0:自動設定、1:人工設定
人工設定時、用補間・終点最大值寫入命令(19h)做終点最大值設定。

D15, 14 INTA, B 連續補間驅動時的中斷設定。
預存緩衝暫存器的狀態計數器變化要產生中斷時設定。

D15 (INT1)	D14 (INT0)	補間中斷的發生
0	0	無効
0	1	狀態計數器 4 → 3 變化時
1	0	狀態計數器 8 → 7 變化時
1	1	狀態計數器 8 → 7 變化時 狀態計數器 4 → 3 變化時

中斷發生後、補間中斷輸出訊號(INT1N)變為 Low 位準。補間中斷清除命令發行、下一區段的補間實効命令發行、或連續補間驅動結束的時候清除、補間中斷輸出訊號回復為未動作的位準。

重置時、D15~D0 全部0。D5,13 位元請保持為0。

7.4 資料讀取命令

資料讀取命令為各軸的暫存器的內容由讀取資料暫存器讀取的命令。

WR0 暫存器寫入資料讀取命令碼、指定的資料就會設定在 RR6,7 暫存器。CPU再由 RR6,7 暫存器讀取指定的資料。指定資料資料長為2位元組的時候用 RR6 暫存器、資料長4位元組的時候 RR6,7 暫存器。

讀取資料全部二進位。另、負值為2的補數。

【注意】

- 資料讀取命令的命令處理所需時間最多 125nsec (CLK=16MHz 的時候)。請命令寫入後此時間內讀取 RR6,7 暫存器。
- 各速度參數值、計時器值記載的單位為輸入時脈 (CLK) 16MHz 的時候。輸入時脈 (CLK) 為 16MHz 以外的時候的參數計算式請參考付錄B。
- 軸指定請務必僅1軸。

7.4.1 理論位置計數 讀取

命令碼	命令	參數記號	資料範圍	資料長(位元組)
3 0h	理論位置計數 讀取	L P	-2, 147, 483, 648 ~ +2, 147, 483, 647	4

理論位置計數的現在值、RR6,7 讀取資料暫存器設定。

7.4.2 實際位置計數 讀取

命令碼	命令	參數記號	資料範圍	資料長(位元組)
3 1h	實際位置計數 讀取	R P	-2, 147, 483, 648 ~ +2, 147, 483, 647	4

實際位置計數的現在值、RR6,7 讀取資料暫存器設定。

7.4.3 現在驅動速度 讀取

命令碼	命令	參數記號	資料範圍	資料長(位元組)
3 2h	現在驅動速度 讀取	C V	0 ~ 8,000,000	4

驅動中的現在驅動速度的值、RR6,7 讀取資料暫存器設定。驅動停止時讀取值為0。資料的單位與驅動速度設定值 (DV) 相同為 pps。

7.4.4 現在加減速度 讀取

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
3 3 h	現在加減速度 讀取	C A	0 ~ 536,870,911	4

加減速驅動、加速中現在加速度、減速中現在減速度、RR6,7 讀取資料暫存器設定。
 驅動停止中讀取值為 0。
 資料的單位與加速度設定值(AC)、減速度設定值(DC)相同 pps/sec。

【注意】

直線加減速驅動(對稱)、驅動中隨時可以讀取設定的加速度。
 S 字加減速驅動時、等速域下現在加減速度讀取值無效。

7.4.5 多目的暫存器0 讀取

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
3 4 h	多目的暫存器 0 讀取	M R 0	-2,147,483,648 ~ +2,147,483,647	4

多目的暫存器 MR0 的值、RR6,7 讀取資料暫存器設定。
 同步動作儲存在 MR0 的現在位置及現在計時器值、現在速度值的讀取時使用。

7.4.6 多目的暫存器 1 讀取

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
3 5 h	多目的暫存器 1 讀取	M R 1	-2,147,483,648 ~ +2,147,483,647	4

多目的暫存器 MR1 的值、RR6,7 讀取資料暫存器設定。
 同步動作儲存在 MR1 的現在位置及現在計時器值、現在速度值的讀取時使用。

7.4.7 多目的暫存器2 讀取

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
3 6 h	多目的暫存器 2 讀取	M R 2	-2,147,483,648 ~ +2,147,483,647	4

多目的暫存器 MR2 的值、RR6,7 讀取資料暫存器設定。
 同步動作儲存在 MR2 的現在位置及現在計時器值、現在速度值的讀取時使用。

7.4.8 多目的暫存器3 讀取

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
3 7 h	多目的暫存器3 讀取	M R 3	-2, 147, 483, 648 ~ +2, 147, 483, 647	4

多目的暫存器 MR3 的值、RR6,7 讀取資料暫存器設定。
同步動作儲存在 MR3 的現在位置及現在計時器值、現在速度值的讀取時使用。

7.4.9 現在計時器值 讀取

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
3 8 h	現在計時器值 讀取	C T	0 ~ 2, 147, 483, 647	4

計時器動作中的現在計時器值、RR6,7 讀取資料暫存器設定。計時器停止時讀取值為0。
資料的單位與計時器設定值(TM)相同 μsec 。

7.4.10 補間・終点最大值 讀取

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
3 9 h	補間・終点最大值 讀取	T X	1 ~ 2,147,483,646	4

直線補間的各軸的終点設定時点的終点最大值，在 RR6,7 讀取資料暫存器設定。本命令不要指定軸。

依補間驅動狀態的情況讀取值會有差異。
補間驅動執行前、讀取值為輸入中的補間區段的終点最大值。補間驅動執行中讀取值為現在執行中的補間區段的終点最大值。

7.4.11 現在螺旋回轉數 讀取

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
3 A h	現在螺旋回轉數 讀取	C H	0~65, 535	2

螺旋補間動作中的現在螺旋回轉數設定於 RR6 讀取資料暫存器。本命令不要指定軸。

7.4.12 螺旋演算值 讀取

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
3 Bh	螺旋演算值 讀取	H V	1 ~ 2, 147, 483, 646	4

讀取用螺旋演算命令(6Bh,6Ch)做螺旋演算的結果時使用。
演算結果設定在 RR6,7 讀取資料暫存器。本命令不要指定軸。

螺旋補間進行時、需要用本命令讀取螺旋演算值設定給 IC。
螺旋演算值設定到本 IC 為所需的步驟如下。

- ① 螺旋演算命令(6Bh,6Ch)的螺旋演算執行
- ② 演算結果由本命令讀取
- ③ 讀取值以螺旋演算值寫入命令(1Ah)設定

螺旋補間請參考、3.3 節。

7.4.13 WR1設定值 讀取

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
3 Dh	WR 1 設定值 讀取	WR 1	(位元資料)	2

WR1 暫存器的設定值由 RR6 讀取資料暫存器讀取。
即使讀取 WR1 暫存器位址也讀不到 WR1 暫存器的設定資料。要確認 WR1 暫存器的設定值時、請用本命令讀取。

RR7 讀取暫存器設定為 0。

7.4.14 WR2設定值 讀取

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
3 Eh	WR 2 設定值 讀取	WR 2	(位元資料)	2

WR2 暫存器的設定值由 RR6 讀取資料暫存器讀取。
即使讀取 WR2 暫存器位址也讀不到 WR2 暫存器的設定資料。要確認 WR2 暫存器的設定值時、請用本命令讀取。

RR7 讀取暫存器設定為 0。

7.4.15 WR3設定值 讀取

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
3 Fh	WR 3 設定值 讀取	WR 3	(位元資料)	2

WR3 暫存器的設定值由 RR6 讀取資料暫存器讀取。

即使讀取 WR3 暫存器位址也讀不到 WR3 暫存器的設定資料。要確認 WR3 暫存器的設定值時、請用本命令讀取。

RR7 讀取暫存器設定為 0。

7.4.16 多目的暫存器模式設定 讀取

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
4 0h	多目的暫存器模式設定 讀取	MRM	(位元資料)	2

多目的暫存器模式設定命令(20h)的設定值、RR6 讀取資料暫存器中設定。

RR7 讀取暫存器設定為 0。

7.4.17 PIO訊號設定1 讀取

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
4 1h	P I O訊號設定 1 讀取	P 1 M	(位元資料)	2

PIO 訊號設定1命令(21h)設定值、RR6 讀取資料暫存器設定。

RR7 讀取暫存器設定為 0。

7.4.18 PIO訊號設定2 讀取

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
4 2h	P I O訊號設定 2 · 其他設定 讀取	P 2 M	(位元資料)	2

PIO訊號設定2·其他設定命令(22h)設定值、RR6 讀取資料暫存器設定。

RR7 讀取暫存器設定為 0。

7.4.19 加速度設定値 讀取

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
4 3 h	加速度設定 讀取	A C	1 ~ 536,870,911	4

加速度設定命令(02h)設定値、RR6,7 讀取資料暫存器設定。
資料的單位 pps/sec。

依據同步動作 MR3 值載入加速度値(AC)時、該値被讀取。

7.4.20 初速度設定値 讀取

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
4 4 h	初速度設定値 讀取	S V	1 ~ 8,000,000	4

初速度設定命令(04h)設定値、RR6,7 讀取資料暫存器設定。
資料的單位 pps。

依據同步動作 MR2 值載入初速度値(SV)時、該値被讀取。

7.4.21 驅動速度設定値 讀取

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
4 5 h	驅動速度設定値 讀取	D V	1 ~ 8,000,000	4

驅動速度設定命令(05h)設定値、RR6,7 讀取資料暫存器設定。
資料的單位 pps。

依據同步動作 MRn 值載入驅動速度値(DV)時、該値被讀取。

7.4.22 移動脈衝數／終点設定値 讀取

命令碼	命 令	參數記號	資料範圍	資料長(位元組)
4 6 h	移動脈衝數／終点設定値 讀取	T P	-2,147,483,646 ~ +2,147,483,646	4

移動脈衝數／終点速度設定命令(06h)設定値、RR6,7 讀取資料暫存器設定。

依據同步動作 MRn 值載入移動脈衝數／終点値(TP)時、該値被讀取。

7.4.23 分割脈衝設定1 讀取

命令碼	命 令	參數記號	資料範圍		資料長(位元組)
4 7 h	分割脈衝設定1 讀取	S P 1	RR6	分割脈衝長度：2 ~ 65,535	4
			RR7	脈衝寬度：1 ~ 65,534	

分割脈衝設定1命令(17h)設定值、RR6,7 讀取資料暫存器設定。
RR6 暫存器為分割脈衝長度、RR7 暫存器為脈衝寬度設定。

依據同步動作 MRn 值載入分割脈衝設定1(SP1)時、該值被讀取。

7.4.24 汎用輸入值 讀取

命令碼	命 令	參數記號	資料範圍		資料長(位元組)
4 8 h	汎用輸入值 讀取	U I	RR7	下位 1 位元組 (PIN)	4
			RR6	下位2位元組 (I2C通信時的D15~0)	

本命令不要指定軸。

PIN7~0 做汎用輸入使用時、RR7 讀取暫存器的下位 8 位元設定。上位 8 位元組 0。

I²C 串列介面匯流排模式時、D15~0 做汎用輸入使用時、RR6 讀取暫存器設定。不是 I²C 串列介面匯流排模式時、RR6 讀取暫存器 0 設定。

7.5 驅動命令

驅動命令為、驅動脈衝輸出命令和它一起的命令。

無寫入資料、WR0 指令暫存器寫入命令碼後立刻執行。

驅動中、RR0 主狀態暫存器的 DRIVE 位元變為1。驅動結束時、DRIVE 位元變為0。

伺服馬達驅動器用的 nINPOS 訊號設定有效時、驅動結束後、等待 nINPOS 輸入訊號動作後、RR0 主狀態暫存器的 DRIVE 位元會變為0。

【注意】

- 驅動命令的命令處理所需時間最多 125nsec (CLK=16MHz 的時候)。下一次的命令寫入時請等待此時間的過後進行。

7.5.1 相對位置驅動

命令碼	命令
5 0h	相對位置驅動

設定的帶符號移動脈衝數由+方向驅動脈衝訊號 (PP)、或一方向驅動脈衝訊號 (PM) 輸出脈衝。移動脈衝數的值為正的時候由PP輸出訊號輸出脈衝、負的時候由PM輸出訊號輸出脈衝。(驅動脈衝輸出方式: 獨立2脈衝方式時)

驅動中、+方向的驅動脈衝輸出1脈衝時理論位置計數加1、一方向的驅動脈衝輸出1脈衝時理論位置計數減1。

驅動命令寫入前、輸出速度曲線所需參數、移動脈衝數一定要先正確設定。

○ : 必須設定

參數	輸出速度曲線				
	等速	對稱直線加減速	非對稱直線加減速	對稱S形加減速	非對稱S形加減速
加速度增加率 (JK)				○	○
減速度增加率 (DJ)					○
加速度 (AC)		○	○	○ *	○ *
減速度 (DC)			○		○ *
初速度 (SV)	○	○	○	○	○
驅動速度 (DV)	○	○	○	○	○
移動脈衝數/終點 (TP)	○	○	○	○	○
手動減速點 (DP)					○

*注: 最大值 536, 870, 911 (1FFF FFFFh) 設定。但部分S形加減速為、直線加速/減速部分的加速度/減速度設定。

7.5.2 反相对位置驅動

命令碼	命 令
5 1 h	反相对位置驅動

設定帶符號的移動脈衝數由+方向驅動脈衝訊號(PP)、或是一方向驅動脈衝訊號(PM)輸出脈衝。移動脈衝數的值為正的時候由PM輸出訊號輸出脈衝、負的時候由PP輸出訊號輸出脈衝。(驅動脈衝輸出方式:獨立2脈衝方式時)

本驅動命令是在已確定的移動脈衝數根據驅動命令改變輸出方向時使用。
通常移動脈衝數(TP)設定為正的脈衝量、要做+方向移動時寫入相对位置驅動命令(50h)、要做一方向移動寫入反相对位置驅動命令(51h)。

驅動中、+方向的驅動脈衝輸出1脈衝時理論位置計數加1、一方向的驅動脈衝輸出1脈衝時理論位置計數減1。

驅動命令寫入前、輸出速度曲線所需的參數及移動脈衝數要先確實設定。

7.5.3 +方向連續脈衝驅動

命令碼	命 令
5 2 h	+方向連續脈衝驅動

停止指令或指定的外部訊號動作為止、連續PP輸出訊號輸出脈衝。(脈衝輸出方式:獨立2脈衝方式時)

驅動中、驅動脈衝輸出1脈衝時理論位置計數加1

驅動命令寫入前、輸出速度曲線所需的參數要先確實設定。

7.5.4 -方向連續脈衝驅動

命令碼	命 令
5 3 h	-方向連續脈衝驅動

停止指令或指定的外部訊號動作為止、連續PM輸出訊號輸出脈衝。(脈衝輸出方式:獨立2脈衝方式時)

驅動中、驅動脈衝輸出1脈衝時理論位置計數減1

驅動命令寫入前、輸出速度曲線所需的參數要先確實設定。

7.5.5 絕對位置驅動

命令碼	命 令
5 4 h	絕對位置驅動

現在座標到終點座標為止驅動。

驅動前、原点(理論位置計數=0)為基準設定移動後終點、以移動脈衝數/終點設定命令(06h)設定帶符號 32 位元值。

驅動命令寫入前、輸出速度曲線所需的參數及終點要先確實設定。

7.5.6 驅動減速停止

命令碼	命 令
5 6 h	驅動減速停止

驅動脈衝輸出、途中減速停止。驅動中的速度若低於初速度時立即停止。

7.5.7 驅動即停止

命令碼	命 令
5 7 h	驅動即停止

驅動脈衝輸出、途中立即停止。加減速驅動也是即停止。

7.5.8 方向訊號+設定

命令碼	命 令
5 8 h	方向訊號+設定

驅動脈衝輸出方式以 1 脈衝·方向方式設定時、驅動開始以前方向訊號DIR+方向的動作位準設定命令。

11.2 節所示 1 脈衝·方向方式驅動開始後、方向訊號確定後的 1CLK 後驅動脈衝的第 1 脈衝輸出。

7.5.9 方向訊號－設定

命令碼	命 令
5 9 h	方向訊號－設定

驅動脈衝輸出方式以 1 脈衝・方向方式設定時、驅動開始以前方向訊號DIR－方向的動作位準設定命令。
11.2 節所示 1 脈衝・方向方式驅動開始後、方向訊號確定後的 1CLK 後驅動脈衝的第 1 脈衝輸出。

7.5.10 自動原点復歸執行

命令碼	命 令
5 A h	自動原点復歸執行

自動原点復歸的執行。
執行前、自動原点復歸模式及各參數要先確實設定。自動原点復歸的詳細請參考 2.6 節。

7.6 補間命令

補間命令在、2軸／3軸／4軸直線補間、CW／CCW圆弧補間、2軸／3軸／4軸位元補間、CW／CCW螺旋補間及補間驅動關聯的命令組成。補間命令、WRO指令暫存器的D11～8位元的軸指定不需要。請設定0。

各種補間進行時、補間驅動開始前需共同做以下2点。

- a. 指定補間軸。（補間模式設定。）
- b. 設定主軸的速度參數。

補間驅動中、RRO主狀態暫存器的補間軸的n-DRV位元變為1、驅動結束時變為0。

【注意】

- 補間命令的命令處理所需時間、最多 125nSEC (CLK=16MHz 的時候)。下個命令寫入時請等待此時間後進行。

7.6.1 1軸直線驅動(多晶片補間)

命令碼	命 令
6 0 h	1 軸直線驅動(多晶片補間)

僅多晶片補間時可以使用。主或副晶片的補間軸僅設定為1軸時使用。

7.6.2 2軸直線補間驅動

命令碼	命 令
6 1 h	2 軸直線補間驅動

現在座標到終點座標做2軸直線補間。

驅動前、補間的2軸的每個的終點相對值以輸出脈衝(P)設定。

7.6.3 3軸直線補間驅動

命令碼	命 令
6 2 h	3 軸直線補間驅動

現在座標到終點座標為止做3軸直線補間。

驅動前、補間的3軸的每個的終點相對值以輸出脈衝(P)設定。

7.6.4 4軸直線補間驅動

命令碼	命 令
6 3 h	4 軸直線補間驅動

現在座標到終點座標為止做4軸直線補間。

驅動前、補間的4軸的每個的終點相對值以輸出脈衝(P)設定。

7.6.5 CW圓弧補間驅動

命令碼	命 令
6 4 h	CW圓弧補間驅動

指定的中心座標為中心、現在座標到終點座標為止做順時針方的圓弧補間。

驅動前補間的2軸相對於現在位置的中心點作為圓弧中心點(C)、相對於現在位置的終點為輸出脈衝(P)做相對值的設定。

終點座標(0, 0)設定時、描繪出全圓。

7.6.6 CCW圓弧補間驅動

命令碼	命 令
6 5 h	CCW圓弧補間驅動

指定的中心座標為中心、現在座標到終點座標為止做逆時針方的圓弧補間。

驅動前補間的2軸相對於現在位置的中心點作為圓弧中心點(C)、相對於現在位置的終點為輸出脈衝(P)做相對值的設定。

終點座標(0, 0)設定時、描繪出全圓。

7.6.7 2軸位元補間驅動

命令碼	命 令
6 6 h	2 軸位元補間驅動

執行2軸位元補間。

驅動前、補間的2軸的+方向/—方向的位元資料設定。驅動前可以設定的位元資料、各軸各方向多有 $16 \times 8 = 128$ 位元。超過的部分在驅動中做補充。

7.6.8 3軸位元補間驅動

命令碼	命 令
6 7 h	3 軸位元補間驅動

執行3軸位元補間。

驅動前、補間的3軸的+方向/—方向的位元資料設定。驅動前可以設定的位元資料、各軸各方向多有 $16 \times 8 = 128$ 位元。超過的部分在驅動中做補充。

7.6.9 4軸位元補間驅動

命令碼	命 令
6 8 h	4 軸位元補間驅動

執行4軸位元補間。

驅動前、補間的4軸的+方向/—方向的位元資料設定。驅動前可以設定的位元資料、各軸各方向多有 $16 \times 8 = 128$ 位元。超過的部分在驅動中做補充。

7.6.10 CW螺旋補間驅動

命令碼	命 令
6 9 h	CW螺旋補間驅動

執行順時針方向螺旋補間。

驅動前以下的值需做設定。

- ① X、Y軸、現在位置的終點及中心點
- ② Z、U軸的移動量
- ③ 螺旋演算值
- ④ 螺旋回轉數

螺旋補間不能做連續補間。

螺旋補間請參考、3.3 節。

7.6.11 CCW螺旋補間驅動

命令碼	命 令
6 A h	CCW螺旋補間驅動

執行逆時針方向螺旋補間。

驅動前必須定的參數與CW螺旋補間驅動相同。

7.6.12 CW螺旋演算

命令碼	命 令
6 Bh	CW螺旋演算

執行順時針方向的螺旋演算。

本IC的螺旋補間為求Z或U軸的移動均一的關係，會用到凹弧補間的總輸出脈衝數與Z或U軸的移動脈衝數。但是、XY平面上動作的凹弧補間的總輸出脈衝數由凹弧的中心點、終點的值做精度計算非常困難。因此螺旋補間進行前、凹弧補間的總輸出脈衝數用螺旋演算來取得。

演算前以下參數值需設定。

- ① X、Y軸的終點及中心點的設定
- ② 螺旋圈數的設定

螺旋演算中、驅動脈衝雖不輸出、但RR0暫存器的D0,1(XDRV、YDRV)位元會ON。

另、利用驅動結束中斷可以判定演算結束(X軸指定)。

螺旋演算執行後、以螺旋演算值讀取命令(3Bh)讀取演算結果、再用螺旋演算值寫入命令(1Ah)設定。

螺旋補間請參考3.3節。

7.6.13 CCW螺旋演算

命令碼	命 令
6 Ch	CCW螺旋演算

執行逆時針方向的螺旋演算。

螺旋演算的必要設定與CW螺旋演算相同。

7.6.14 減速有效

命令碼	命 令
6 Dh	減速有效

加減速補間驅動時自動減速或手動減速設為有效狀態。

单独的補間驅動作加減速時、驅動前一定要用執行本命令。連續補間時、開始減速補間區段的補間命令寫入的前、先寫入減速有效命令。

重置時為減速無效狀態。以本命令使減速成位有效狀態時、補間驅動結束、減速無效命令(3C)寫入或是重置為止為有效狀態。

減速有效/無效只在補間驅動時有用。各軸獨自驅動時、隨時為自動減速或手動減速有效狀態。

7.6.15 減速無効

命令碼	命 令
6 E h	減速無効

加減速補間驅動時自動減速或手動減速為無効狀態。

7.6.16 補間中斷清除/單步補間

命令碼	命 令
6 F h	補間中斷清除/單步補間

補間中斷清除為清除連續補間所發生的中斷 (INT1N訊號)。

單步補間是補間驅動以單一脈衝方式做單步輸出。

請參考中斷相關為2.11節、單步補間進給相關3.9節。

7.7 同步動作操作命令

同步動作操作命令為、將同步動作設為有効、無効或是起動的命令。

4 個同步動作組 SYNC3~0 之中、可設定任意的同步動作組同時有効、無効、起動。

同步動作操作命令在 WR0 指令暫存器的 D7~D4 的 4 位元中操作命令碼、D3~D0 的 4 位元指定要設定的同步動作組。即 D7~D4 設定在同步動作有効時為 8h、無効時為 9h、起動時 Ah。或是、D3~D0 4 個同步動作組 SYNC3、SYNC2、SYNC1、SYNC0 所對應的位元、欲操作的同步動作組在相對位元設為 1。



不需寫入資料、WR0 指令暫存器內指定軸及命令碼寫入後執行。

【注意】

- 同步動作操作命令的命令處理所需時間、最多 125nsec (CLK=16MHz 的時候)。下一個命令寫入前請等待此時間後進行。

7.7.1 同步動作 有効設定

命令碼	命 令
8 1h ~ 8 Fh	同步動作 有効設定

命令碼的下位 4 位元中位元指定後各同步動作組為有効。

同步動作有効設定前、執行同步動作 SYNC3~0 設定命令 (29h~26h) 來設定有效的同步動作組的模式設定。

■ 設定例: X 軸的同步動作組 SYNC0 と SYNC2 又設為有効時、WR0 寫入 0185h。

同步動作 SYNC3~0 的有效/無効狀態は可由 RR3 暫存器 頁 1 確認。

重置時、SYNC3~0 全部無効設定。

【注意】

PIO 訊號設定 2・其他設定命令 (22h) 下、錯誤發生時同步動作設定為無効 (D7:ERRDE 位元=1)、因此錯誤發生狀態 (RR0 暫存器的 n-ERR 位元為 1) 的時候、執行本命令也無法讓同步動作設定有効。以錯誤・結束狀態清除命令 (79h) 等將 n-ERR 位元清除後、在執行同步動作有効設定命令。

7.7.2 同步動作 無効設定

命令碼	命 令
9 1 h ~ 9 F h	同步動作 無効設定

命令碼の下位 4 位元中位元指定後各同步動作組設為無効。
設定無効後の同步動作組は、起動要因発生及同步動作起動命令不能起動。

■ 設定例: X 軸の同步動作組 SYNC1 及 SYNC3 設為無効時在、WR0 寫入 019Ah 即可

同步動作 SYNC3~0 の有効/無効設定狀況可以由 RR3 暫存器 頁 1 來確認。
重置時、SYNC3~0 全部設定為無効。

7.7.3 同步動作 起動

命令碼	命 令
A 1 h ~ A F h	同步動作 起動

命令碼の下位 4 位元中位元指定後各同步動作組可由命令起動。
同步動作起動前、要起動的同步動作組的模式設定須由同步動作 SYNC3~0 設定命令 (29h~26h) 執行。另、依據同步動作有效設定命令、要起動的同步動作組須先設為有效。
同步動作 SYNC3~0 の有効/無効設定狀況は RR3 暫存器 頁 1 にて確認することが出来ます。

■ 設定例: X 軸の同步動作組 SYNC0 起動時、WR0 寫入 01A1h
X 軸の同步動作組 SYNC3~0 全部起動時、WR0 寫入 01AFh。

7.8 其他的命令

不需寫入資料、軸指定字 WR0 指令暫存器中寫入命令碼後開始執行。

【注意】

- 命令處理所需時間、最多 125nSEC (CLK=16MHz 的時候)。下個命令寫入時請等待此時間後進行。

7.8.1 速度增加

命令碼	命 令
7 0 h	速度增加

驅動中速度以速度增減值增加。

速度增減值 (IV) 為速度增減值設定命令 (15h) 在事前設定。

本命令在連續脈衝驅動中使用。定量脈衝驅動中請勿使用。定量脈衝驅動中使用太頻繁時、驅動結束時會有斷尾・拖拉的情況發生。

另外、S形加減速驅動、加速中及減速中本命令執行時無效。務必等速中 (RR3 頁 0/D5:CNST=1) 進行。

本命令的執行後、不會更新驅動速度設定值 (DV)。

補間驅動下本命令不可以使用。

7.8.2 速度減少

命令碼	命 令
7 1 h	速度減少

驅動中速度以速度增減值分減少。

速度增減值 (IV) 為速度增減值設定命令 (15h) 在事前設定。

本命令在連續脈衝驅動中使用。定量脈衝驅動中請勿使用。定量脈衝驅動中使用太頻繁時、驅動結束時會有斷尾・拖拉的情況發生。

另外、S形加減速驅動、加速中及減速中本命令執行時無效。務必等速中 (RR3 頁 0/D5:CNST=1) 進行。

本命令的執行後、不會更新驅動速度設定值 (DV)。

補間驅動下本命令不可以使用。

7.8.3 偏差計數清除輸出

命令碼	命 令
7 2 h	偏差計數清除輸出

nDCC 輸出端子輸出偏差計數清除脈衝。

此命令執行前、需以自動原点復歸模式設定2命令 (24h) 設定脈衝的理論位準、脈衝寬度。詳細請參考 2.6.2 項、2.6.4 項。

7.8.4 計時器始動

命令碼	命 令
7 3 h	計時器始動

開始計時。

本命令執行後開始計時，現在計時器值 (CT) 由 0 開始遞增、及時到指定的計時器值 (TM) 後時間到。時間到後、計時器可以重複動作。要重複動作時在 WR3 暫存器的 D14 位元 (TMMD) 設為 1。

計時器動作的詳細請參考、2.10 節。

7.8.5 計時器停止

命令碼	命 令
7 4 h	計時器停止

停止計時。

計時途中停止時遞增的現在計時器值 (CT) 回到 0。下次計時開始時由 0 開始。

7.8.6 分割脈衝開始

命令碼	命 令
7 5 h	分割脈衝開始

開始分割脈衝的輸出。

分割脈衝於驅動中由 nSPLTP 輸出端子輸出。

分割脈衝開始命令執行後、分割脈衝動作中在 RR3 暫存器 頁1的 SPLIT 位元變為 1。

本命令執行前、分割脈衝長等各參數須先正確的設定。

分割脈衝的各參數的詳細請參考 2.8 節。

7.8.7 分割脈衝停止

命令碼	命 令
7 6 h	分割脈衝停止

停止分割脈衝的輸出。

分割脈衝停止命令執行後、分割脈衝動作中在 RR3 暫存器 頁1的 SPLIT 位元變為 0。

分割脈衝停止命令執行時、分割脈衝輸出訊號在 Hi 位準時，確保指定的脈衝寬度的 Hi 位準處理後結束。(正理論設定時)

7.8.8 驅動開始保持

命令碼	命 令
7 7 h	驅動開始保持

驅動的開始暫時停止。

複數的軸的驅動要使其同時開始時使用。要同時啟動的軸先執行本命令後、當各軸驅動命令寫入完成。在同時寫入驅動開始釋放命令(25h)後、全軸同時驅動開始。

驅動中本命令寫入時驅動不會停止。下次的驅動命令會保持住。

另外、連續補間下驅動開始前將必要的區段補間資料寫入預存緩衝暫存器時也會使用本命令。連續補間請參考 3.7 節。

7.8.9 驅動開始釋放

命令碼	命 令
7 8 h	驅動開始釋放

依據驅動開始保持命令(77h)設定解除驅動開始(含補間驅動)保持狀態。

7.8.10 錯誤・結束狀態清除

命令碼	命 令
7 9 h	錯誤・結束狀態清除

RR2 暫存器的全部的錯誤情報位元、全部的驅動結束狀態位元、及 RR0 暫存器的錯誤位元(D7~4:n-ERR)清為 0。

補間驅動錯誤發生時也使用本命令做錯誤清除。

7.8.11 RR3 頁0顯示

命令碼	命 令
7 Ah	RR3 頁0顯示

RR3 暫存器的顯示內容設定為頁0。
頁0顯示時、RR3 暫存器的 D15 位元為0。

7.8.12 RR3 頁1顯示

命令碼	命 令
7 Bh	RR3 頁1顯示

RR3 暫存器的顯示內容設定為頁1。
頁1顯示時、RR3 暫存器的 D15 位元為1。

7.8.13 終点最大值清除

命令碼	命 令
7 Ch	終点最大值清除

直線補間、清除現在寫入中的補間終点自動算出終点最大值。
本命令軸指定不要。
本命令執行後，全部補間軸的終点須再次寫入。

7.8.14 NOP

命令碼	命 令
1 Fh	NOP

執行後此命令什麼多不會做。

7.8.15 指令重置

命令碼	命 令
00FFh	指令重置

重置本 IC。

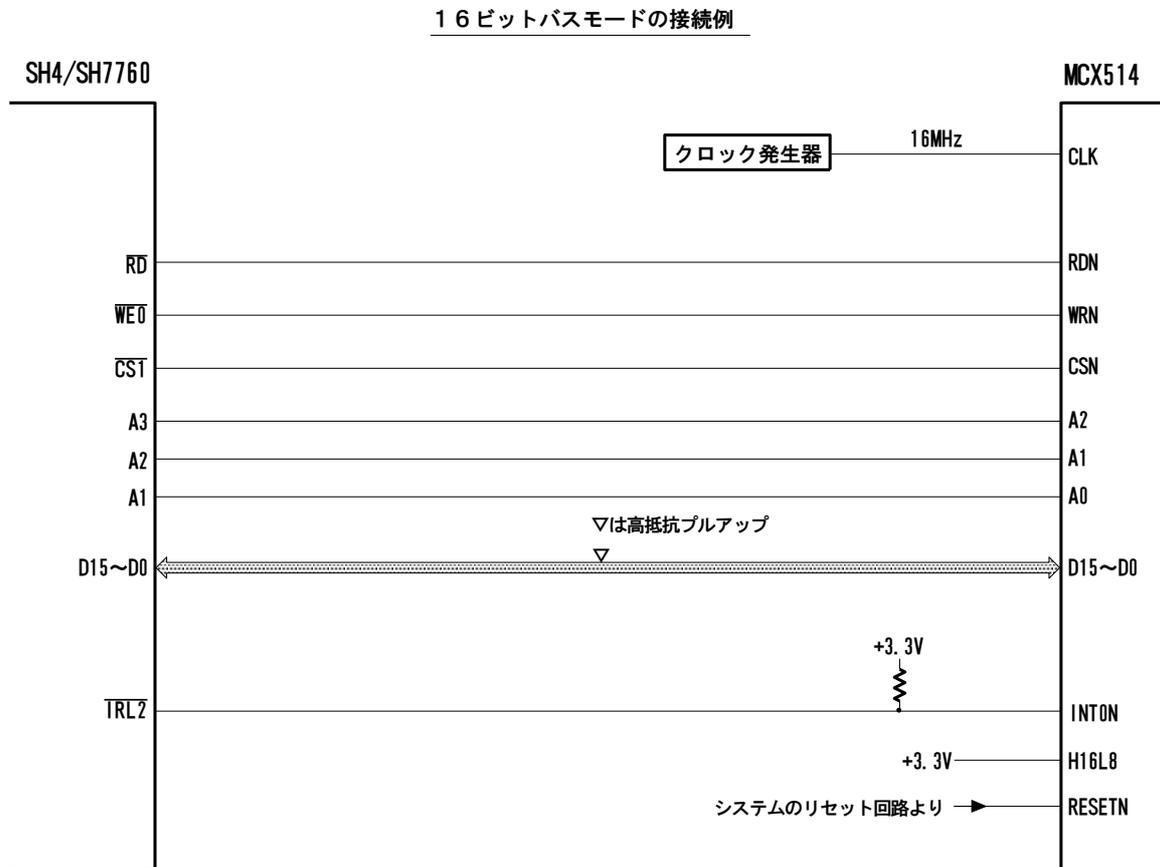
本命令請務必將 WR0 暫存器的上位 8 位元 (D15~D8) 全部設為 0。
命令寫入後 8CLK (500nsec:CLK=16MHz 時) 的間隔下、本 IC 無法存取。

本命令使用 8 位元資料匯流排、或是 I²C 串列介面匯流排、上位位元組 (WR0H) 也務必寫入。
上位位元組 (WR0H) 務必先寫入 00h、下位位元組 (WR0L) 在寫入 FFh。下位位元組寫入後、立刻執行直重置。

8. 輸出入訊號接續例

8.1 16 位元／8 位元匯流排模式的接續例

■SH-4CPU と的 16 位元匯流排模式接續例



SH-4/SH7760 等待控制例

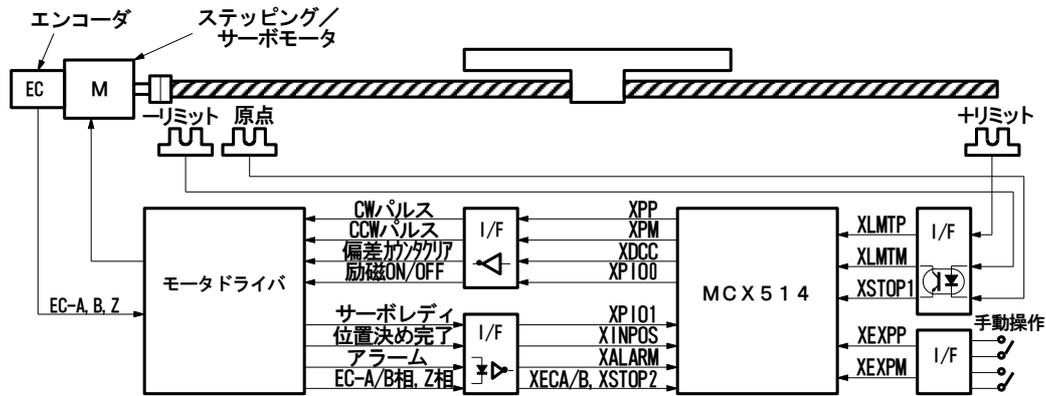
匯流排時脈	66.664MHz	—
設定上升等待	1 循環挿入	暫存器設定 : WCR3/A1S0=1
存取等待	2 循環挿入	暫存器設定 : WCR2/A1W2, A1W1, A1W0 = 010
保持等待	1 循環挿入	暫存器設定 : WCR3/A1H1, A1H0 = 01

8.2 I²C 串列介面匯流排模式的接續例

現在作成中です。

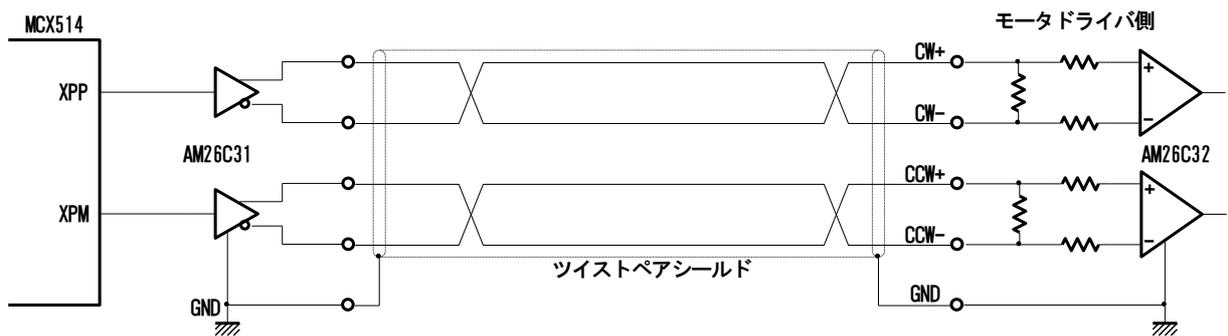
8.3 運動制御系統構成例

下図為、運動控制系統的X軸例。4軸全部可以用相同的構造。

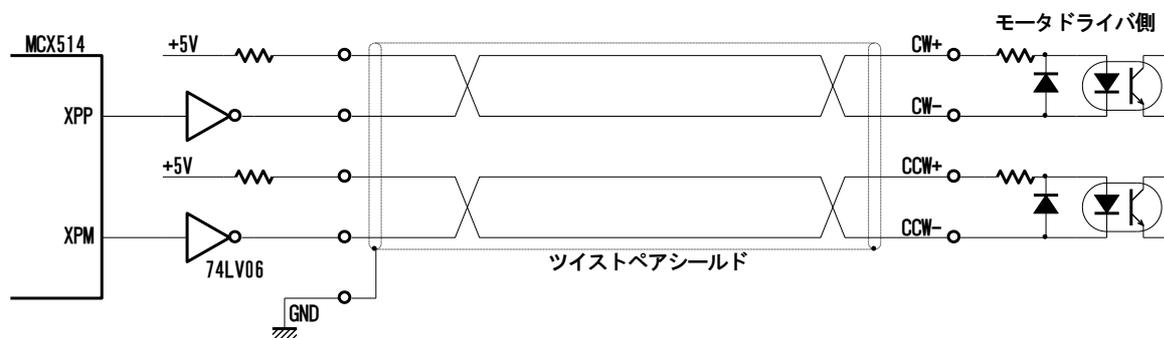


8.4 驅動脈衝輸出回路例

■ 差動線性驅動輸出



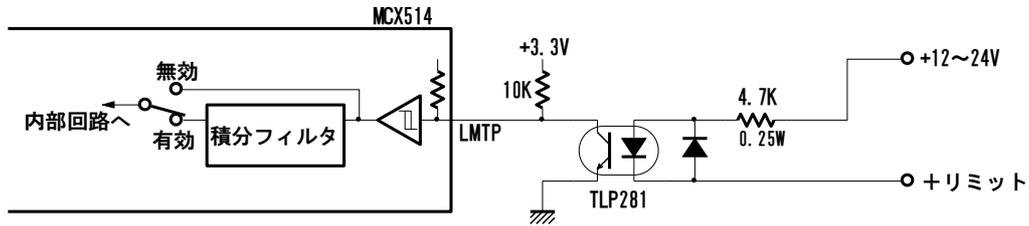
■ 集極開路 TTL 輸出



驅動脈衝輸出訊號考慮EMC影響時建議使用對絞的遮蔽線。

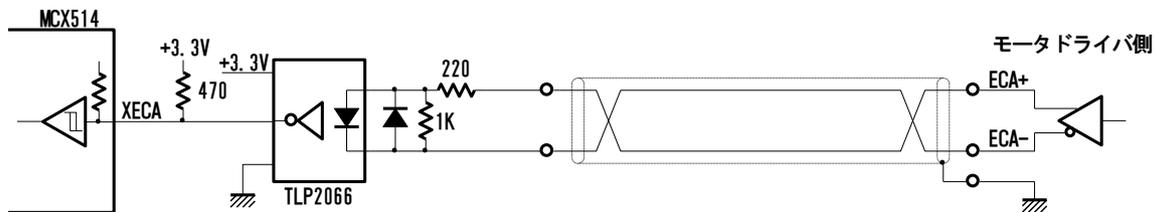
8.5 極限等の輸入訊號的接続例

極限訊號等、通常配線相當複雜、雜訊容易附著。單純光耦合無法吸收全部的雜訊。有効IC內濾波機能時、請設定適當時間常數(FL=Ah,Bh)。



8.6 編碼器輸入訊號的接続例

下圖為差動線性驅動輸出的編碼器訊號以高速光耦合IC接收後傳送給MCX514的輸入回路例。



9. 控制程式例

函數名など、一部、作成中のものがあります。

此章以C言語做MCX514的控制程式例。16位元匯流排構成的程式。

此程式可從本公司網頁 (<http://www.novaelec.co.jp/>) 下載。檔名為:MCX514Apl.c

```

////////////////////////////////////
// 命令碼的定義
////////////////////////////////////
////////////////////////////////////
// 資料寫入命令
////////////////////////////////////
#define          MCX514_CMD00_JK          0x0000          // 加速度增加率 設定
#define          MCX514_CMD01_DJ          0x0001          // 減速度增加率 設定
#define          MCX514_CMD02_AC          0x0002          // 加速度 設定
#define          MCX514_CMD03_DC          0x0003          // 減速度 設定
#define          MCX514_CMD04_SV          0x0004          // 初速度 設定
#define          MCX514_CMD05_DV          0x0005          // 驅動速度 設定
#define          MCX514_CMD06_TP          0x0006          // 移動脈衝數/終点 設定
#define          MCX514_CMD07_DP          0x0007          // 手動減速点 設定
#define          MCX514_CMD09_LP          0x0009          // 理論位置計數 設定
#define          MCX514_CMD0A_RP          0x000A          // 實際位置計數 設定
#define          MCX514_CMD0B_SP          0x000B          // 軟體極限+ 設定
#define          MCX514_CMD0C_SM          0x000C          // 軟體極限- 設定
#define          MCX514_CMD0D_AO          0x000D          // 加速計數偏移 設定
#define          MCX514_CMD0E_LX          0x000E          // 理論位置計數最大值 設定
#define          MCX514_CMD0F_RX          0x000F          // 實際位置計數最大值 設定
#define          MCX514_CMD10_MR0         0x0010          // 多目的暫存器 0 設定
#define          MCX514_CMD11_MR1         0x0011          // 多目的暫存器 1 設定
#define          MCX514_CMD12_MR2         0x0012          // 多目的暫存器 2 設定
#define          MCX514_CMD13_MR3         0x0013          // 多目的暫存器 3 設定
#define          MCX514_CMD14_HV          0x0014          // 原点檢出速度 設定
#define          MCX514_CMD15_IV          0x0015          // 速度增減值 設定
#define          MCX514_CMD16_TM          0x0016          // 計時器值 設定
#define          MCX514_CMD17_SP1         0x0017          // 分割脈衝設定 1
#define          MCX514_CMD18_SP2         0x0018          // 分割脈衝設定 2
#define          MCX514_CMD19_TX          0x0019          // 補間・終点最大值設定
#define          MCX514_CMD1A_HN          0x001A          // 螺旋回轉數設定
#define          MCX514_CMD1B_HV          0x001B          // 螺旋演算值設定

////////////////////////////////////
// 模式寫入命令
////////////////////////////////////
#define          MCX514_CMD20_MRM         0x0020          // 多目的暫存器模式設定
#define          MCX514_CMD21_P1M         0x0021          // PIO 訊號設定 1
#define          MCX514_CMD22_P2M         0x0022          // PIO 訊號設定 2・其他設定
#define          MCX514_CMD23_H1M         0x0023          // 自動原点復歸模式設定 1
#define          MCX514_CMD24_H2M         0x0024          // 自動原点復歸模式設定 2
#define          MCX514_CMD25_FLM         0x0025          // 輸入訊號濾波模式設定
#define          MCX514_CMD26_SOM         0x0026          // 同步動作 SYNC0 設定
#define          MCX514_CMD27_S1M         0x0027          // 同步動作 SYNC1 設定
#define          MCX514_CMD28_S2M         0x0028          // 同步動作 SYNC2 設定
#define          MCX514_CMD29_S3M         0x0029          // 同步動作 SYNC3 設定
#define          MCX514_CMD2A_XXX         0x002A          // 補間模式設定

////////////////////////////////////
// 資料讀取命令
////////////////////////////////////
#define          MCX514_CMD30_LP          0x0030          // 理論位置計數 讀取
#define          MCX514_CMD31_RP          0x0031          // 實際位置計數 讀取
#define          MCX514_CMD32_CV          0x0032          // 現在驅動速度 讀取
#define          MCX514_CMD33_CA          0x0033          // 現在加減速度 讀取
#define          MCX514_CMD34_MR0         0x0034          // 多目的暫存器 0 讀取
#define          MCX514_CMD35_MR1         0x0035          // 多目的暫存器 1 讀取
#define          MCX514_CMD36_MR2         0x0036          // 多目的暫存器 2 讀取
#define          MCX514_CMD37_MR3         0x0037          // 多目的暫存器 3 讀取
#define          MCX514_CMD38_CT          0x0038          // 現在計時器值 讀取

```

```

#define          MCX514_CMD39_XX          0x0039          // 補間・終点最大值 讀取
#define          MCX514_CMD3A_XX          0x003A          // 現在螺旋回轉數 讀取
#define          MCX514_CMD3B_XX          0x003B          // 螺旋演算値 讀取
#define          MCX514_CMD3D_WR1         0x003D          // WR1 設定値 讀取
#define          MCX514_CMD3E_WR2         0x003E          // WR2 設定値 讀取
#define          MCX514_CMD3F_WR3         0x003F          // WR3 設定値 讀取
#define          MCX514_CMD40_MRM         0x0040          // 多目的暫存器模式設定 讀取
#define          MCX514_CMD41_P1M         0x0041          // P10 訊號設定 1 讀取
#define          MCX514_CMD42_P2M         0x0042          // P10 訊號設定 2・其他設定 讀取
#define          MCX514_CMD43_AC          0x0043          // 加速度設定値 讀取
#define          MCX514_CMD44_SV          0x0044          // 初速度設定値 讀取
#define          MCX514_CMD45_DV          0x0045          // 驅動速度設定値 讀取
#define          MCX514_CMD46_TP          0x0046          // 移動脈衝數/終点設定値 讀取
#define          MCX514_CMD47_SP1         0x0047          // 分割脈衝設定 1 讀取
#define          MCX514_CMD48_XX          0x0048          // 汎用輸入値 讀取

////////////////////////////////////
// 驅動命令
////////////////////////////////////
#define          MCX514_CMD50_DRVRL       0x0050          // 相對位置驅動
#define          MCX514_CMD51_DRVNR       0x0051          // 反相對位置驅動
#define          MCX514_CMD52_DRVVP       0x0052          // +方向連續脈衝驅動
#define          MCX514_CMD53_DRVVM       0x0053          // -方向連續脈衝驅動
#define          MCX514_CMD54_DRVAB       0x0054          // 絕對位置驅動
#define          MCX514_CMD56_DRVSBRK     0x0056          // 驅動減速停止
#define          MCX514_CMD57_DRVFBRK     0x0057          // 驅動即停止
#define          MCX514_CMD58_DIRCP       0x0058          // 方向訊號+設定
#define          MCX514_CMD59_DIRCM       0x0059          // 方向訊號-設定
#define          MCX514_CMD5A_HMSRC       0x005A          // 自動原点復歸執行

////////////////////////////////////
// 補間命令
////////////////////////////////////
#define          MCX514_CMD60_LHK1         0x0060          // 1 軸直線驅動(多晶片用)
#define          MCX514_CMD61_LHK2         0x0061          // 2 軸直線補間驅動
#define          MCX514_CMD62_LHK3         0x0062          // 3 軸直線補間驅動
#define          MCX514_CMD63_LHK4         0x0063          // 4 軸直線補間驅動
#define          MCX514_CMD64_CHKCW        0x0064          // CW 円弧補間驅動
#define          MCX514_CMD65_CHKCCW       0x0065          // CCW 円弧補間驅動
#define          MCX514_CMD66_BHK2         0x0066          // 2 軸位元補間驅動
#define          MCX514_CMD67_BHK3         0x0067          // 3 軸位元補間驅動
#define          MCX514_CMD68_BHK4         0x0068          // 4 軸位元補間驅動
#define          MCX514_CMD69_HLCW         0x0069          // CW 螺旋補間驅動
#define          MCX514_CMD6A_HLCCW        0x006A          // CCW 螺旋補間驅動
#define          MCX514_CMD6B_HLPCW        0x006B          // CW 螺旋演算
#define          MCX514_CMD6C_HLPCCW       0x006C          // CCW 螺旋演算
#define          MCX514_CMD6D_XXX          0x006D          // 減速有効
#define          MCX514_CMD6E_XXX          0x006E          // 減速無効
#define          MCX514_CMD6F_XXX          0x006F          // 補間中斷清除/補間步驟

////////////////////////////////////
// 同步動作操作命令
////////////////////////////////////
#define          MCX514_CMD81_SYNCOEN      0x0081          // 同步動作 SYNC0 有効設定
#define          MCX514_CMD82_SYNC1EN     0x0082          // 同步動作 SYNC1 有効設定
#define          MCX514_CMD84_SYNC2EN     0x0084          // 同步動作 SYNC2 有効設定
#define          MCX514_CMD88_SYNC3EN     0x0088          // 同步動作 SYNC3 有効設定
#define          MCX514_CMD91_SYNCODIS     0x0091          // 同步動作 SYNC0 無効設定
#define          MCX514_CMD92_SYNC1DIS    0x0092          // 同步動作 SYNC1 無効設定
#define          MCX514_CMD94_SYNC2DIS    0x0094          // 同步動作 SYNC2 無効設定
#define          MCX514_CMD98_SYNC3DIS    0x0098          // 同步動作 SYNC3 無効設定
#define          MCX514_CMDA1_SYNCOACT     0x00A1          // 同步動作 SYNC0 起動
#define          MCX514_CMDA2_SYNC1ACT     0x00A2          // 同步動作 SYNC1 起動
#define          MCX514_CMDA4_SYNC2ACT     0x00A4          // 同步動作 SYNC2 起動
#define          MCX514_CMDA8_SYNC3ACT     0x00A8          // 同步動作 SYNC3 起動

////////////////////////////////////
// 其他命令
////////////////////////////////////
#define          MCX514_CMD70_VINC         0x0070          // 速度增加

```

```

#define MCX514_CMD71_VDEC 0x0071 // 速度減少
#define MCX514_CMD72_DCC 0x0072 // 偏差計數清除輸出
#define MCX514_CMD73_TMSTA 0x0073 // 計時器始動
#define MCX514_CMD74_TMSTP 0x0074 // 計時器停止
#define MCX514_CMD75_SPSTA 0x0075 // 分割脈衝開始
#define MCX514_CMD76_SPSTP 0x0076 // 分割脈衝停止
#define MCX514_CMD77_XXX 0x0077 // 驅動開始ホールド
#define MCX514_CMD78_XXX 0x0078 // 驅動開始フリー
#define MCX514_CMD79_R2CLR 0x0079 // 錯誤・結束狀態清除
#define MCX514_CMD7A_XXX 0x007A // RR3 0 頁顯示
#define MCX514_CMD7B_XXX 0x007B // RR3 1 頁顯示
#define MCX514_CMD1F_NOP 0x001F // NOP
#define MCX514_CMDFF_RST 0x00FF // 指令重置

```

```

//////////////////////////////////////////////////////////////////
// 軸的定義

```

```

//////////////////////////////////////////////////////////////////
#define MCX514_AXIS_X 0x01 // X 軸
#define MCX514_AXIS_Y 0x02 // Y 軸
#define MCX514_AXIS_Z 0x04 // Z 軸
#define MCX514_AXIS_U 0x08 // U 軸
#define MCX514_AXIS_ALL 0x0f // 全軸指定
#define MCX514_AXIS_NONE 0x00 // 無軸指定

```

```

//////////////////////////////////////////////////////////////////
// 位址的定義

```

```

//////////////////////////////////////////////////////////////////
#define REG_ADDR 0x0000 // 基本位址

```

```

// 寫入暫存器、讀取暫存器的定義

```

```

#define MCX514_WR0 0x00
#define MCX514_WR1 0x02
#define MCX514_WR2 0x04
#define MCX514_WR3 0x06
#define MCX514_WR4 0x08
#define MCX514_WR6 0x0c
#define MCX514_WR7 0x0e
#define MCX514_RR0 0x00
#define MCX514_RR1 0x02
#define MCX514_RR2 0x04
#define MCX514_RR3 0x06
#define MCX514_RR4 0x08
#define MCX514_RR5 0x0a
#define MCX514_RR6 0x0c
#define MCX514_RR7 0x0e

```

```

unsigned short reg_read (unsigned short n);

```

```

#define reg_write(n,c) (*(volatile unsigned short *)n = ((volatile)c))
#define reg_read(n) (*(volatile unsigned short *)n)

```

```

//////////////////////////////////////////////////////////////////
// 共通函數的宣言

```

```

//////////////////////////////////////////////////////////////////
int WriteReg(volatile unsigned short *Adr, unsigned short Data); // WR 暫存器寫入共通函數
int ReadReg(volatile unsigned short *Adr, unsigned short *Data); // RR 暫存器讀取共通函數
int SetData(unsigned short Cmd, int Axis, long Data); // 資料寫入命令共通函數
int SetModeData(unsigned short Cmd, int Axis, unsigned short Data); // 模式寫入命令共通函數
int GetData(unsigned short Cmd, int Axis, long *Data); // 資料讀取命令共通函數
int ExeCmd(unsigned short Cmd, int Axis); // 命令執行共通函數

```

```

//////////////////////////////////////////////////////////////////
// WR 暫存器寫入函數

```

```

//////////////////////////////////////////////////////////////////
int WriteReg0(unsigned short Data) { // WR0 暫存器寫入
    return(WriteReg((volatile unsigned short*)(REG_ADDR + MCX514_WR0), Data));
}
int WriteReg1(int Axis, unsigned short Data) { // WR1 暫存器寫入
    WriteReg0(((Axis << 8) + MCX514_CMD1F_NOP)); // 軸指定
}

```

```

        return(WriteReg((volatile unsigned short*)(REG_ADDR + MCX514_WR1), Data));
    }

int WriteReg2(int Axis, unsigned short Data) {                // WR2 暫存器寫入
    WriteReg0(((Axis << 8) + MCX514_CMD1F_NOP));            // 軸指定
    return(WriteReg((volatile unsigned short*)(REG_ADDR + MCX514_WR2), Data));
}
int WriteReg3(int Axis, unsigned short Data) {                // WR3 暫存器寫入
    WriteReg0(((Axis << 8) + MCX514_CMD1F_NOP));            // 軸指定
    return(WriteReg((volatile unsigned short*)(REG_ADDR + MCX514_WR3), Data));
}
int WriteReg4(unsigned short Data) {                          // WR4 暫存器寫入
    return(WriteReg((volatile unsigned short*)(REG_ADDR + MCX514_WR4), Data));
}
int WriteReg6(unsigned short Data) {                          // WR6 暫存器寫入
    return(WriteReg((volatile unsigned short*)(REG_ADDR + MCX514_WR6), Data));
}
int WriteReg7(unsigned short Data) {                          // WR7 暫存器寫入
    return(WriteReg((volatile unsigned short*)(REG_ADDR + MCX514_WR7), Data));
}

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
// RR 暫存器讀取函數
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
int ReadReg0(unsigned short *Data) {                          // RR0 暫存器讀取
    return(ReadReg((volatile unsigned short*)(REG_ADDR + MCX514_RR0), Data));
}
int ReadReg1(int Axis, unsigned short *Data) {                // RR1 暫存器讀取
    WriteReg0(((Axis << 8) + MCX514_CMD1F_NOP));            // 軸指定
    return(ReadReg((volatile unsigned short*)(REG_ADDR + MCX514_RR1), ((Axis << 8) + Data)));
}
int ReadReg2(int Axis, unsigned short *Data) {                // RR2 暫存器讀取
    WriteReg0(((Axis << 8) + MCX514_CMD1F_NOP));            // 軸指定
    return(ReadReg((volatile unsigned short*)(REG_ADDR + MCX514_RR2), ((Axis << 8) + Data)));
}
int ReadReg3(int Page, int Axis, unsigned short *Data) {     // RR3 暫存器讀取
    if (Page == 0) {                                         // 0 頁指定
        WriteReg0(((Axis << 8) + MCX514_CMD7A_XXX));
    }
    else {                                                    // 1 頁指定
        WriteReg0(((Axis << 8) + MCX514_CMD7B_XXX));
    }
    WriteReg0(((Axis << 8) + MCX514_CMD1F_NOP));            // 軸指定
    return(ReadReg((volatile unsigned short*)(REG_ADDR + MCX514_RR3), ((Axis << 8) + Data)));
}
int ReadReg4(unsigned short *Data) {                          // RR4 暫存器讀取
    return(ReadReg((volatile unsigned short*)(REG_ADDR + MCX514_RR4), Data));
}
int ReadReg5(unsigned short *Data) {                          // RR5 暫存器讀取
    return(ReadReg((volatile unsigned short*)(REG_ADDR + MCX514_RR5), Data));
}
int ReadReg6(unsigned short *Data) {                          // RR6 暫存器讀取
    return(ReadReg((volatile unsigned short*)(REG_ADDR + MCX514_RR6), Data));
}
int ReadReg7(unsigned short *Data) {                          // RR7 暫存器讀取
    return(ReadReg((volatile unsigned short*)(REG_ADDR + MCX514_RR7), Data));
}

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
// 資料寫入命令 函數
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
int SetStartSpd(int Axis, long Data) {                        // 初速度 設定
    return(SetData(MCX514_CMD04_SV, Axis, Data));
}
int SetSpeed(int Axis, long Data) {                           // 驅動速度 設定
    return(SetData(MCX514_CMD05_DV, Axis, Data));
}
int SetJerk(int Axis, long Data) {                            // 加速度增加率 設定

```

```

        return(SetData(MCX514_CMD00_JK, Axis, Data));
    }

    int SetDJerk(int Axis, long Data) { // 減速度增加率 設定
        return(SetData(MCX514_CMD01_DJ, Axis, Data));
    }
    int SetAcc(int Axis, long Data) { // 加速度 設定
        return(SetData(MCX514_CMD02_AC, Axis, Data));
    }
    int SetDec(int Axis, long Data) { // 減速度 設定
        return(SetData(MCX514_CMD03_DC, Axis, Data));
    }
    int SetPulse(int Axis, long Data) { // 移動脈衝數/終點 設定
        return(SetData(MCX514_CMD06_TP, Axis, Data));
    }

    int SetDecP(int Axis, long Data) { // 手動減速點 設定
        return(SetData(MCX514_CMD07_DP, Axis, Data));
    }
    int SetLp(int Axis, long Data) { // 理論位置計數 設定
        return(SetData(MCX514_CMD09_LP, Axis, Data));
    }
    int SetRp(int Axis, long Data) { // 實際位置計數 設定
        return(SetData(MCX514_CMD0A_RP, Axis, Data));
    }
    int SetCompP(int Axis, long Data) { // 軟體極限+ 設定
        return(SetData(MCX514_CMD0B_SP, Axis, Data));
    }
    int SetCompM(int Axis, long Data) { // 軟體極限- 設定
        return(SetData(MCX514_CMD0C_SM, Axis, Data));
    }
    int SetAccOfst(int Axis, long Data) { // 加速計數偏移 設定
        return(SetData(MCX514_CMD0D_AO, Axis, Data));
    }
    int SetHomeSpd(int Axis, long Data) { // 原點檢出速度 設定
        return(SetData(MCX514_CMD14_HV, Axis, Data));
    }
    int SetLpMax(int Axis, long Data) { // 理論位置計數最大值 設定
        return(SetData(MCX514_CMD0E_LX, Axis, Data));
    }
    int SetRpMax(int Axis, long Data) { // 實際位置計數最大值 設定
        return(SetData(MCX514_CMD0F_RX, Axis, Data));
    }
    int SetMR0(int Axis, long Data) { // 多目的暫存器 0 設定
        return(SetData(MCX514_CMD10_MR0, Axis, Data));
    }
    int SetMR1(int Axis, long Data) { // 多目的暫存器 1 設定
        return(SetData(MCX514_CMD11_MR1, Axis, Data));
    }
    int SetMR2(int Axis, long Data) { // 多目的暫存器 2 設定
        return(SetData(MCX514_CMD12_MR2, Axis, Data));
    }
    int SetMR3(int Axis, long Data) { // 多目的暫存器 3 設定
        return(SetData(MCX514_CMD13_MR3, Axis, Data));
    }
    int SetSpeedInc(int Axis, long Data) { // 速度增減值 設定
        return(SetData(MCX514_CMD15_IV, Axis, Data));
    }
    int SetTimer(int Axis, long Data) { // 計時器值 設定
        return(SetData(MCX514_CMD16_TM, Axis, Data));
    }
    int SetSplit1(int Axis, unsigned short Data1, unsigned short Data2) { // 分割脈衝設定 1
        long Data;

        Data = ((Data1 << 16) | Data2);
        return(SetData(MCX514_CMD17_SP1, Axis, Data));
    }
    int SetSplit2(int Axis, long Data) { // 分割脈衝設定 2
        return(SetData(MCX514_CMD18_SP2, Axis, Data));
    }

```

```

}
int SetTPMax(long Data) { // 補間・終点最大値設定
    return(SetData(MCX514_CMD39_XX, MCX514_AXIS_NONE, Data));
}
int SetHLNumber(unsigned short Data) { // 螺旋回轉數設定
    return(SetData(MCX514_CMD3A_XX, MCX514_AXIS_NONE, (long )Data));
}
int SetHLValue(long Data) { // 螺旋演算値設定
    return(SetData(MCX514_CMD3B_XX, MCX514_AXIS_NONE, Data));
}

////////////////////////////////////
// 模式寫入命令 函數
////////////////////////////////////
int SetModeMRn(int Axis, unsigned short Data) { // 多目的暫存器模式設定
    return(SetModeData(MCX514_CMD20_MRM, Axis, Data));
}
int SetModePIO1(int Axis, unsigned short Data) { // PIO 訊號設定 1
    return(SetModeData(MCX514_CMD21_P1M, Axis, Data));
}
int SetModePIO2(int Axis, unsigned short Data) { // PIO 訊號設定 2・其他設定
    return(SetModeData(MCX514_CMD22_P2M, Axis, Data));
}
int SetModeHMSrch1(int Axis, unsigned short Data) { // 自動原点復歸模式設定 1
    return(SetModeData(MCX514_CMD23_H1M, Axis, Data));
}
int SetModeHMSrch2(int Axis, unsigned short Data) { // 自動原点復歸模式設定 2
    return(SetModeData(MCX514_CMD24_H2M, Axis, Data));
}
int SetModeFilter(int Axis, unsigned short Data) { // 輸入訊號濾波模式設定
    return(SetModeData(MCX514_CMD25_FLM, Axis, Data));
}
int SetModeSync0(int Axis, unsigned short Data) { // 同步動作 SYNC0 設定
    return(SetModeData(MCX514_CMD26_S0M, Axis, Data));
}
int SetModeSync1(int Axis, unsigned short Data) { // 同步動作 SYNC1 設定
    return(SetModeData(MCX514_CMD27_S1M, Axis, Data));
}
int SetModeSync2(int Axis, unsigned short Data) { // 同步動作 SYNC2 設定
    return(SetModeData(MCX514_CMD28_S2M, Axis, Data));
}
int SetModeSync3(int Axis, unsigned short Data) { // 同步動作 SYNC3 設定
    return(SetModeData(MCX514_CMD29_S3M, Axis, Data));
}
int SetModeXXX(unsigned short Data) { // 補間模式設定
    return(SetModeData(MCX514_CMD2A_XXX, MCX514_AXIS_NONE, Data));
}

////////////////////////////////////
// 資料讀取函數 函數
////////////////////////////////////
int GetLp(int Axis, long *Data) { // 理論位置計數 讀取
    return(GetData(MCX514_CMD30_LP, Axis, Data));
}
int GetRp(int Axis, long *Data) { // 實際位置計數 讀取
    return(GetData(MCX514_CMD31_RP, Axis, Data));
}
int GetCV(int Axis, long *Data) { // 現在驅動速度 讀取
    return(GetData(MCX514_CMD32_CV, Axis, Data));
}
int GetCA(int Axis, long *Data) { // 現在加減速度 讀取
    return(GetData(MCX514_CMD33_CA, Axis, Data));
}
int GetCT(int Axis, long *Data) { // 現在計時器值 讀取
    return(GetData(MCX514_CMD38_CT, Axis, Data));
}
int GetMRO(int Axis, long *Data) { // 多目的暫存器 0 讀取
    return(GetData(MCX514_CMD34_MRO, Axis, Data));
}
int GetMR1(int Axis, long *Data) { // 多目的暫存器 1 讀取

```

```

        return(GetData(MCX514_CMD35_MR1, Axis, Data));
    }

int GetMR2(int Axis, long *Data) { // 多目的暫存器 2 讀取
    return(GetData(MCX514_CMD36_MR2, Axis, Data));
}
int GetMR3(int Axis, long *Data) { // 多目的暫存器 3 讀取
    return(GetData(MCX514_CMD37_MR3, Axis, Data));
}
int GetXXX(long *Data) { // 補間・終点最大值 讀取
    return(GetData(MCX514_CMD39_XXX, MCX514_AXIS_NONE, Data));
}
int GetXXX(long *Data) { // 現在螺旋回轉數 讀取
    return(GetData(MCX514_CMD3A_XXX, MCX514_AXIS_NONE, Data));
}
int GetXXX(long *Data) { // 螺旋演算值 讀取
    return(GetData(MCX514_CMD3B_XXX, MCX514_AXIS_NONE, Data));
}

int GetWR1(int Axis, long *Data) { // WR1 設定值 讀取
    return(GetData(MCX514_CMD3D_WR1, Axis, Data));
}
int GetWR2(int Axis, long *Data) { // WR2 設定值 讀取
    return(GetData(MCX514_CMD3E_WR2, Axis, Data));
}
int GetWR3(int Axis, long *Data) { // WR3 設定值 讀取
    return(GetData(MCX514_CMD3F_WR3, Axis, Data));
}
int GetMRM(int Axis, long *Data) { // 多目的暫存器模式設定 讀取
    return(GetData(MCX514_CMD40_MRM, Axis, Data));
}
int GetP1M(int Axis, long *Data) { // PIO 訊號設定 1 讀取
    return(GetData(MCX514_CMD41_P1M, Axis, Data));
}
int GetP2M(int Axis, long *Data) { // PIO 訊號設定 2・其他設定 讀取
    return(GetData(MCX514_CMD42_P2M, Axis, Data));
}
int GetAc(int Axis, long *Data ) { // 加速度設定值 讀取
    return(GetData(MCX514_CMD43_AC, Axis, Data));
}
int GetStartSpd(int Axis, long *Data ) { // 初速度設定值 讀取
    return(GetData(MCX514_CMD44_SV, Axis, Data));
}
int GetSpeed(int Axis, long *Data ) { // 驅動速度設定值 讀取
    return(GetData(MCX514_CMD45_DV, Axis, Data));
}
int GetPulse(int Axis, long *Data ) { // 移動脈衝數/終点設定值 讀取
    return(GetData(MCX514_CMD46_TP, Axis, Data));
}
int GetSplit(int Axis, long *Data ) { // 分割脈衝設定 1 讀取
    return(GetData(MCX514_CMD47_SP1, Axis, Data));
}
int GetXXX(long *Data ) { // 汎用輸入值 讀取
    return(GetData(MCX514_CMD48_XXX, MCX514_AXIS_NONE, Data));
}

}

////////////////////////////////////
// 驅動命令 函數
////////////////////////////////////
int ExeDRVRL(int Axis) { // 相對位置驅動
    return (ExeCmd(MCX514_CMD50_DRVRL, Axis));
}
int ExeDRVNR(int Axis) { // 反相對位置驅動
    return (ExeCmd(MCX514_CMD51_DRVNR, Axis));
}
int ExeDRVVP(int Axis) { // 十方向連續脈衝驅動
    return (ExeCmd(MCX514_CMD52_DRVVP, Axis));
}
int ExeDRVVM(int Axis) { // 一方向連續脈衝驅動

```

```

        return (ExeCmd(MCX514_CMD53_DRVVM, Axis));
    }

int ExeDRVAB(int Axis) { // 絕對位置驅動
    return (ExeCmd(MCX514_CMD54_DRVAB, Axis));
}
int ExeDRVSBRK(int Axis) { // 驅動減速停止
    return (ExeCmd(MCX514_CMD56_DRVSBRK, Axis));
}
int ExeDRVFBRK(int Axis) { // 驅動即停止
    return (ExeCmd(MCX514_CMD57_DRVFBRK, Axis));
}
int ExeDIRCP(int Axis) { // 方向訊號+設定
    return (ExeCmd(MCX514_CMD58_DIRCP, Axis));
}
int ExeDIRCM(int Axis) { // 方向訊號-設定
    return (ExeCmd(MCX514_CMD59_DIRCM, Axis));
}
int ExeHMSRC(int Axis) { // 自動原点復歸執行
    return (ExeCmd(MCX514_CMD5A_HMSRC, Axis));
}

////////////////////////////////////
// 補間命令 函數
////////////////////////////////////
int ExeLHK1(void) { // 1 軸直線驅動(多晶片用)
    return (ExeCmd(MCX514_CMD60_LHK1, MCX514_AXIS_NONE));
}
int ExeLHK2(void) { // 2 軸直線驅動
    return (ExeCmd(MCX514_CMD61_LHK2, MCX514_AXIS_NONE));
}
int ExeLHK3(void) { // 3 軸直線驅動
    return (ExeCmd(MCX514_CMD62_LHK3, MCX514_AXIS_NONE));
}
int ExeLHK4(void) { // 4 軸直線驅動
    return (ExeCmd(MCX514_CMD63_LHK4, MCX514_AXIS_NONE));
}
int ExeCHKCW(void) { // CW 円弧補間驅動
    return (ExeCmd(MCX514_CMD64_CHKCW, MCX514_AXIS_NONE));
}
int ExeCHKCCW(void) { // CCW 円弧補間驅動
    return (ExeCmd(MCX514_CMD64_CHKCCW, MCX514_AXIS_NONE));
}
int ExeBHK2(void) { // 2 軸位元補間驅動
    return (ExeCmd(MCX514_CMD66_BHK2, MCX514_AXIS_NONE));
}
int ExeBHK3(void) { // 3 軸位元補間驅動
    return (ExeCmd(MCX514_CMD67_BHK3, MCX514_AXIS_NONE));
}
int ExeBHK4(void) { // 4 軸位元補間驅動
    return (ExeCmd(MCX514_CMD68_BHK4, MCX514_AXIS_NONE));
}
int ExeHLCW(void) { // CW 螺旋補間驅動
    return (ExeCmd(MCX514_CMD69_HLCCW, MCX514_AXIS_NONE));
}
int ExeHLCCW(void) { // CCW 螺旋補間驅動
    return (ExeCmd(MCX514_CMD6A_HLCCW, MCX514_AXIS_NONE));
}
int ExeHLPCW(void) { // CW 螺旋演算
    return (ExeCmd(MCX514_CMD6B_HLPCW, MCX514_AXIS_NONE));
}
int ExeHLPCCW(void) { // CCW 螺旋演算
    return (ExeCmd(MCX514_CMD6C_HLPCCW, MCX514_AXIS_NONE));
}
int ExeXXX(void) { // 減速有効
    return (ExeCmd(MCX514_CMD6D_XXX, MCX514_AXIS_NONE));
}
int ExeXXX(void) { // 減速無効
    return (ExeCmd(MCX514_CMD6E_XXX, MCX514_AXIS_NONE));
}

```

```

}
int ExeXXX(void ) { // 補間中斷清除／補間步驟
    return (ExeCmd(MCX514_CMD6F_XXX, MCX514_AXIS_NONE));
}
////////////////////////////////////////////////////
// 同步動作操作命令 函數
////////////////////////////////////////////////////
int ExeSYNC(unsigned short Cmd) { // 同步動作関連命令
    return (ExeCmd(Cmd));
}

////////////////////////////////////////////////////
// 其他命令 函數
////////////////////////////////////////////////////
int ExeVINC(int Axis) { // 速度増加
    return (ExeCmd(MCX514_CMD70_VINC, Axis));
}
int ExeVDEC(int Axis) { // 速度減少
    return (ExeCmd(MCX514_CMD71_VDEC, Axis));
}
int ExeDCC(int Axis) { // 偏差計數清除輸出
    return (ExeCmd(MCX514_CMD72_DCC, Axis));
}
int ExeTMSTA(int Axis) { // 計時器始動
    return (ExeCmd(MCX514_CMD73_TMSTA, Axis));
}
int ExeTMSTP(int Axis) { // 計時器停止
    return (ExeCmd(MCX514_CMD74_TMSTP, Axis));
}
int ExeSPSTA(int Axis) { // 分割脈衝開始
    return (ExeCmd(MCX514_CMD75_SPSTA, Axis));
}
int ExeSPSTP(int Axis) { // 分割脈衝停止
    return (ExeCmd(MCX514_CMD76_SPSTP, Axis));
}
int ExeXXX(int Axis) { // 驅動開始保持
    return (ExeCmd(MCX514_CMD77_XXX, Axis));
}
int ExeXXX(int Axis) { // 驅動開始釋放
    return (ExeCmd(MCX514_CMD78_XXX, Axis));
}
int ExeR2CLR(int Axis) { // 錯誤・結束狀態清除
    return (ExeCmd(MCX514_CMD79_R2CLR, Axis));
}
int ExeXXX(int Axis) { // RR3 0 頁顯示
    return (ExeCmd(MCX514_CMD7A_XXX, Axis));
}
int ExeXXX(int Axis) { // RR3 1 頁顯示
    return (ExeCmd(MCX514_CMD7B_XXX, Axis));
}
int ExeNOP(int Axis) { // NOP
    return (ExeCmd(MCX514_CMD1F_NOP, Axis));
}
int ExeSRST(void ) { // 指令重置
    return (ExeCmd(MCX514_CMDFF_RST, MCX514_AXIS_NONE));
}

////////////////////////////////////////////////////
// 共通函數
////////////////////////////////////////////////////
// WR 暫存器寫入共通函數(I/O 位址存取部分。以下の程式例は、SH 微處理器的例。)
int WriteReg(volatile unsigned short *Adr, unsigned short Data) {
    reg_write(Adr, Data);
    return 0;
}
// RR 暫存器讀取共通函數(I/O 位址存取部分。以下の程式例は、SH 微處理器的例)
int ReadReg(volatile unsigned short *Adr, unsigned short *Data) {
    *Data = reg_read(Adr);
}

```

```
    return 0;  
}
```

```

// 資料寫入共通函數
// WR6、WR7 資料寫入、然後、在 WRO 命令寫入後資料同時寫入。
int SetData(unsigned short Cmd, int Axis, long Data) {

    long mask_data = 0x0000ffff;
    unsigned short write_data;

    // WR6 的資料的下位 16bit 寫入
    write_data = (unsigned short)(Data & mask_data);
    WriteReg6(write_data);

    // WR7 的資料的上位 16bit 寫入
    write_data = (unsigned short)(Data >> 16);
    WriteReg7(write_data);

    // 命令寫入(WRO 的寫入)
    WriteReg0(((Axis << 8) + Cmd));

    return 0;
}

// 模式寫入命令 共通函數
// WR6 資料寫入、然後、在 WRO 命令寫入後資料同時寫入。
int SetModeData(unsigned short Cmd, int Axis, unsigned short Data) {

    // WR6 的資料的下位 16bit 寫入
    WriteReg6(Data);

    // 命令寫入(WRO 寫入)
    WriteReg0(((Axis << 8) + Cmd));

    return 0;
}

// 資料讀取共通函數
// WRO 命令寫入、然後、讀取 RR6、RR7、取得資料。
int GetData(unsigned short Cmd, int Axis, long *Data) {

    unsigned short rdata1, rdata2;
    long retdata = 0x00000000;

    if (Data == NULL) return 0;

    // 命令寫入(WRO 的寫入)
    WriteReg0(((Axis << 8) + Cmd));

    // RR7 讀取
    ReadReg7(&rdata1);

    // RR6 讀取
    ReadReg6(&rdata2);

    // 讀取資料的處理
    retdata = (long)rdata1; // RR7 的值設定為上位 16bit
    *Data = (retdata << 16);
    retdata = (long)rdata2; // RR6 的值設定為下位 16bit
    *Data = *Data + retdata;

    return 0;
}

// 命令執行共通函數
int ExeCmd(unsigned short Cmd, int Axis) {

    // 命令寫入(WRO 的寫入)
    WriteReg0(((Axis << 8) + Cmd));

    return 0;
}

```

```

// 等待驅動結束
void waitdrive(int Axis) {

    unsigned short rrData;

    ReadReg0(&rrData);           // RR0 的讀取
    while ((rrData & Axis)) {    // 驅動中
        ReadReg0(&rrData);      // RR0 的讀取
    }
}

// 等待分割脈衝結束
void waitsplit(int Axis) {

    unsigned short rrData;

    ReadReg3(1, Axis, &rrData); // RR3 頁 1 的讀取
    while ((rrData & 0x0800)) {  // 分割脈衝動作中
        ReadReg0(&rrData);      // RR0 的讀取
    }
}

////////////////////////////////////
// 動作例函數
////////////////////////////////////
// 自動原点復歸
// MCX514 手動的「2.5.8 自動原点復歸实例」的「例 1 使用原点訊號做原点復歸例」。
void homesrch(void) {

    WriteReg2(AXIS_X, 0x0800); // 原点訊號理論設定 STOP1 Low 動作
                                // 硬體極限有效
    SetModeFilter(AXIS_X, 0x0A0F); // STOP1 濾波有效
                                    // 濾波遲延 512 μsec
    SetModeHMSrch1(AXIS_X, 0x8037); // 步驟 4 執行
                                    // 步驟 3 不執行
                                    // 步驟 2 執行
                                    //          檢出訊號 STOP1
                                    //          檢出方向 一方向
                                    //          LP, RP 清除 無效
                                    //          DCC 清除 無效
                                    // 步驟 1 執行
                                    //          檢出訊號 STOP1
                                    //          檢出方向 一方向
    SetModeHMSrch2(AXIS_X, 0x0000); // 步驟間計時器 無效
                                    // 原点復歸結束時 LP, RP 清除 無效
    SetAcc(AXIS_X, 95000);          // 加速度 95,000 pps/sec
    SetStartSpd(AXIS_X, 1000);      // 初速度 1000pps
    SetSpeed(AXIS_X, 20000);        // 步驟 1、4 的速度 20000pps
    SetHomeSpd(AXIS_X, 500);        // 步驟 2 的速度 500pps
    SetPulse(AXIS_X, 3500);         // 偏移移動脈衝量 3500
    ExeHMSRC(AXIS_X);               // 自動原点復歸執行
    waitdrive(AXIS_X);              // 等待驅動結束
}

// 全軸 S 字加減速驅動
void drive(void) {

    SetStartSpd(AXIS_ALL, 10);      // 初速度 10pps
    SetSpeed(AXIS_ALL, 40000);      // 驅動速度 40Kpps
    SetAcc(AXIS_ALL, 536870911);    // 加速度 規格最大
    SetJerk(AXIS_ALL, 89300);        // 加速度增加率 89.3Kpps/sec2
    SetPulse(AXIS_ALL, 70000);      // 移動脈衝數 70000
    SetLp(AXIS_ALL, 0);             // 理論位置計數 清除
    WriteReg3(AXIS_ALL, 0x0004);    // S 字加減速驅動指定
    ExeDRVRL(AXIS_ALL);             // 相對位置驅動
    waitdrive(AXIS_ALL);            // 等待驅動結束
}

```

```

// 同步動作
// MCX514 手動的「2.6.6 同步動作的实例」的「例 3 驅動中、求指定位置 A 到指定位置 B 為止所需時間」。
void sync(void) {

    // 10kpps 的等速驅動
    SetStartSpd(8000000); // 初速度 8Mpps(規格最大)
    SetSpeed(10000); // 驅動速度 10Kpps
    SetLp(0); // 理論位置計數 0
    SetPulse(60000); // 移動脈衝數 60000
    SetMR0(10000); // MR0 10000
    SetMR1(55000); // MR1 55000
    SetTimer(2147483647); // 計時器值(規格最大)
    WriteReg1(0x2000); // WR1 同步動作組 1 起動
    SetModeMRn(0x0000); // MR0 與 LP 比較。比較條件 ≥
    // MR1 與 LP 比較。比較條件 ≥
    SetModeSync0(0x0151); // SYNC0 設定
    // 起動要因 MRn 比較為真變化時
    // 動作 計時器始動
    SetModeSync1(0x0071); // SYNC1 設定
    // 起動要因 MRn 比較為真變化時
    // 動作 儲存 CT→MRn
    ExeSYNC((MCX514_CMD81_SYNC0EN | MCX514_CMD82_SYNC1EN));
    // SYNC0,1 有効
    ExeDRVRL(); // 相對位置驅動
    waitdrive(); // 等待驅動結束
}

// 分割脈衝
// MCX501 手動的「2.7.6 分割脈衝的实例」的「例 1 驅動開始時分割脈衝開始」。
void split(void) {

    // 1000pps 的等速驅動
    SetStartSpd(8000000); // 初速度 8Mpps(規格最大)
    SetSpeed(1000); // 驅動速度 1000pps
    SetLp(0); // 理論位置計數
    SetSplit1(9,5); // 分割脈衝長度 9 脈衝寬度 5
    SetSplit2(10); // 脈衝數 10
    SetModePIO2(0x0800); // 脈衝理論正 開始脈衝有
    ExeSPSTA(); // 分割脈衝開始
    ExeDRVVP(); // 十方向連續脈衝驅動
    waitsplit(); // 等待分割脈衝結束
    ExeDRVFBRK(); // 驅動即停止
    waitdrive(); // 等待驅動結束
}

// 主函數
void main(void) {

    ExeSRST(); // 指令重置

    homesrch(); // 自動原点復歸

    drive(); // S 字加減速驅動

    sync(); // 同步動作

    split(); // 分割脈衝
}

```

10. 電氣的特性

10.1 DC 特性

■ 絶対最大定格

項目	記号	条件	定格	単位
電源電圧	V_{DD}	—	-0.3 ~ +4.0	V
輸入電圧	V_I	$V_I < V_{DD} + 3.0V$	-0.3 ~ +7.0	V
輸出電圧	V_O	$V_O < V_{DD} + 3.0V$	-0.3 ~ +7.0	V
輸出電流	I_O	—	±30	mA
保存温度	T_{STG}	—	-65 ~ +150	°C

■ 推奨動作条件

項目	記号	定格	単位
電源電圧	V_{DD}	3.3 ± 0.3	V
周囲温度	T_a	-40 ~ +85	°C

■ DC特性

($T_a = -40 \sim +85^\circ\text{C}$, $V_{DD} = 3.3\text{V} \pm 0.3\text{V}$)

項目	記号	条件	最小	標準	最大	単位	備考
高位準輸入電圧	V_{IH}		2.0		5.5	V	
低位準輸入電圧	V_{IL}		-0.3		0.8	V	
高位準輸入電流	I_{IH}	$V_{IN} = V_{DD}$			1.0	μA	
低位準輸入電流	I_{IL}	$V_{IN} = 0V$			-1.0	μA	
高位準輸出電圧	V_{OH}	$I_{OH} = 0\text{mA}$	$V_{DD} - 0.2$			V	注1
		$I_{OH} = -12\text{mA}$	$V_{DD} - 0.4$			V	D15~D0 訊號
		$I_{OH} = -6\text{mA}$	$V_{DD} - 0.4$			V	上記以外の訊號
低位準輸出電圧	V_{OL}	$I_{OL} = 0\text{mA}$			0.1	V	
		$I_{OL} = 12\text{mA}$			0.4	V	D15~D0 訊號 INTON, INT1N 訊號
		$I_{OL} = 6\text{mA}$			0.4	V	上記以外の訊號
輸出泄漏口流	I_{OZ}	$V_{OUT} = V_{DD} \text{ or } GND$	-1		1	μA	D15~D0 訊號 PIN6, PIN5 訊號 INTON, INT1N 訊號 SDA 訊號
施米特觸口器 磁滯口口	V_H		0.1			V	
消費電流	I_{DD}	$I_{IO} = 0\text{mA}$, CLK=16MHz		150	204	mA	
		$I_{IO} = 0\text{mA}$, CLK=20MHz		190	252		

注1: INTON, INT1N 輸出訊號及 PIN6, 5, SDA 訊號為、因為是漏極開路輸出、所以沒有高位準輸出電圧の項目。

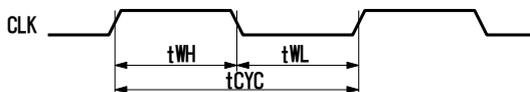
■ 端子容量

項目	記号	条件	最小	標準	最大	単位	備考
輸出入容量	C_{IO}	$T_a = 25^\circ\text{C}$, $f = 1\text{MHz}$			10	pF	D15~D0 訊號 nPI07~nPI00 訊號 PIN7~PIN0 訊號 SDA 訊號
輸入容量	C_I				10	pF	其他的輸入端子

10.2 AC 遅延特性 (Ta = -40~+85°C, V_{DD} = +3.3V±10%, 輸出負荷条件: D15~D0,INTN: 85pF,SDA: 400pF,其他: 50pF)

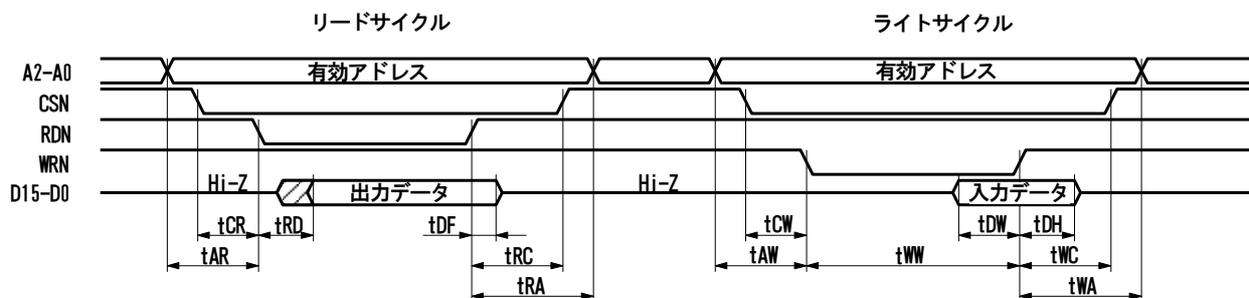
10.2.1 時脈

■ CLK入力信号



記号	項目	最小	標準	最大	単位
tCYC	CLK 周期	50	62.5		nS
tWH	CLK Hi 位準寬度	15			nS
tWL	CLK Low 位準寬度	15			nS

10.2.2 CPU 讀／寫循環

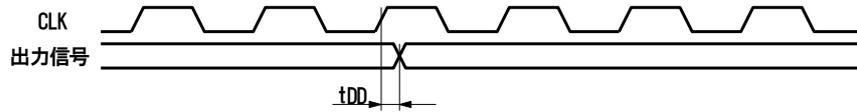


上図は16位元資料匯流排(H16L8=Hi)の訊號。8位元資料匯流排(H16L8=Low)時、図中位址訊號為A3~A0、資料訊號為D7~D0。

記号	項目	最小	最大	単位
tAR	位址設定上升時間 (to RDN ↓)	0		nS
tCR	CSN 設定上升時間 (to RDN ↓)	0		nS
tRD	輸出資料遅延時間 (from RDN ↓)		21	nS
tDF	輸出資料保持時間 (from RDN ↑)	0	12	nS
tRC	CSN 保持時間 (from RDN ↑)	0		nS
tRA	位址保持時間 (from RDN ↑)	3		nS
tAW	位址設定上升時間 (to WRN ↓)	0		nS
tCW	CSN 設定上升時間 (to WRN ↓)	0		nS
tWW	WRN Low 位準脈衝寬度	30		nS
tDW	輸入資料設定上升時間 (to WRN ↑)	10		nS
tDH	輸入資料保持時間 (from WRN ↑)	0		nS
tWC	CSN 保持時間 (from WRN ↑)	0		nS
tWA	位址保持時間 (from WRN ↑)	4		nS

10.2.3 CLK／輸出訊號遲延

以下的輸出訊號隨時與CLK訊號同步。CLK的↑的時候位準變化。



輸出訊號: nPP, nPM, nDCC, nSPLTP, nPIO7~0 (依機能選擇)

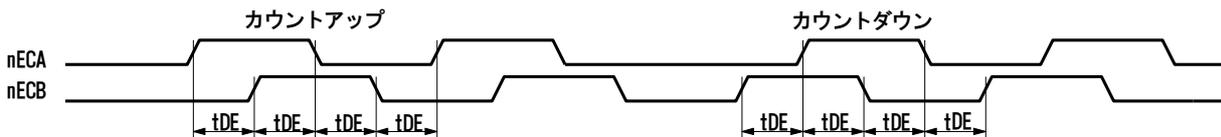
記号	項目	最小	最大	単位
tDD	CLK ↑ → 輸出訊號 ↑ ↓ 遲延時間	7	30	nS

輸出訊號: INTON, INT1N

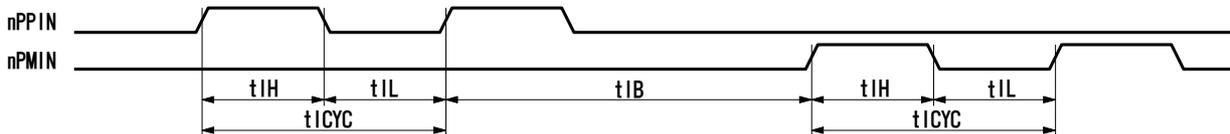
記号	項目	最小	最大	単位
tDD	CLK ↑ → INTON, INT1N 訊號 ↓ 遲延時間	12	22	nS

10.2.4 輸入脈衝

■ 2相脈衝輸入模式



■ 上下脈衝輸入模式



- 2相脈衝輸入模式時、nECA, nECB輸入變化的時候、最多CLK 4循環後、實際位置計數後更新為變化後的值。
- 上下脈衝輸入模式時、nPPIN, nPMIN 輸入的↑起最多 CLK 4循環後、實際位置計數後更新為變化後的值。

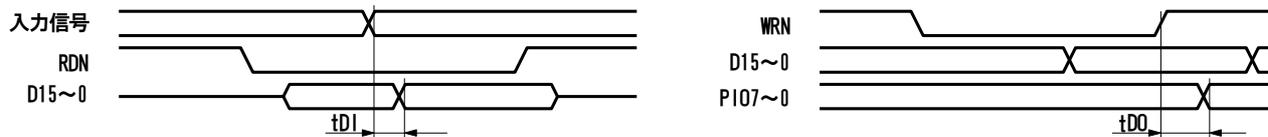
記号	項目	最小	最大	単位
tDE	nECA, nECB 位相差時間	tCYC + 20		nS
tIH	nPPIN, nPMIN Hi 位準寬度	tCYC + 20		nS
tIL	nPPIN, nPMIN Low 位準寬度	tCYC + 20		nS
tICYC	nPPIN, nPMIN 周期	tCYC × 2 + 20		nS
tIB	nPPIN ↑ ↔ nPMIN ↑ 時間	tCYC × 2 + 20		nS

tCYC 為 CLK 的周期。

10.2.5 汎用輸出入訊號 (nPIO7~0)

左下圖為、nPIO7~0輸入訊號在、RR4、5暫存器讀取時的遲延時間。IC內藏濾波無效的時候。

右下圖為、nPIO7~0輸出訊號資料在、WR4,5暫存器寫入的遲延時間。



記号	項目	最小	最大	単位
tDI	輸入訊號 → 資料 遲延時間		17	nS
tDO	WRN ↑ → 資料設定 上升時間		23	nS

10.2.6 分割脈衝

分割脈衝開始，驅動脈衝觸發向上到分割脈衝變為 Hi 為止的遲延時間(分割脈衝為正理論時)。

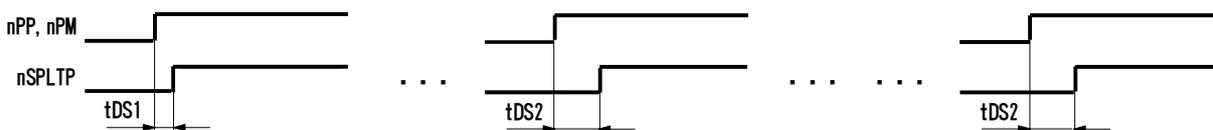
開始脈衝有的時候、只有第 1 個分割脈衝與驅動脈衝同時輸出、第 2 個以後的分割脈衝比驅動脈衝遲 1CLK 輸出。

開始脈衝無的時候、全部的分割脈衝比驅動脈衝遲 1CLK 輸出。

■ 分割脈衝的模式設定開始脈衝有的時候

分割脈衝的模式設定開始脈衝有的時候開始驅動脈衝觸發向上到分割脈衝變 Hi 為止的遲延時間。

tDS1 為第 1 分割脈衝的遲延時間。tDS2 為第 2 分割脈衝以後的遲延時間。第 2 脈衝以後遲 1CLK。

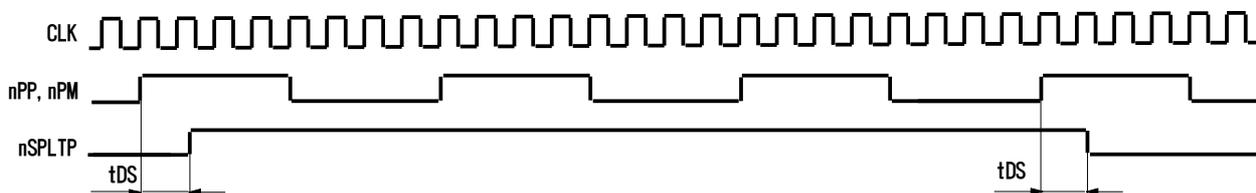


記号	項目	最小	最大	単位
tDS1	nPP, nPM ↑ → nSPLTP ↑ 遲延時間		20	nS
tDS2	nPP, nPM ↑ → nSPLTP ↑ 遲延時間		tCYC + 20	nS

tCYC 為 CLK 的周期。

■ 分割脈衝的模式設定開始脈衝無的時候

分割脈衝的模式設定開始脈衝無的時候開始驅動脈衝觸發向上到分割脈衝變 Hi 為止的遲延時間。



記号	項目	最小	最大	単位
tDS	nPP, nPM ↑ → nSPLTP ↑ 遲延時間		tCYC + 20	nS

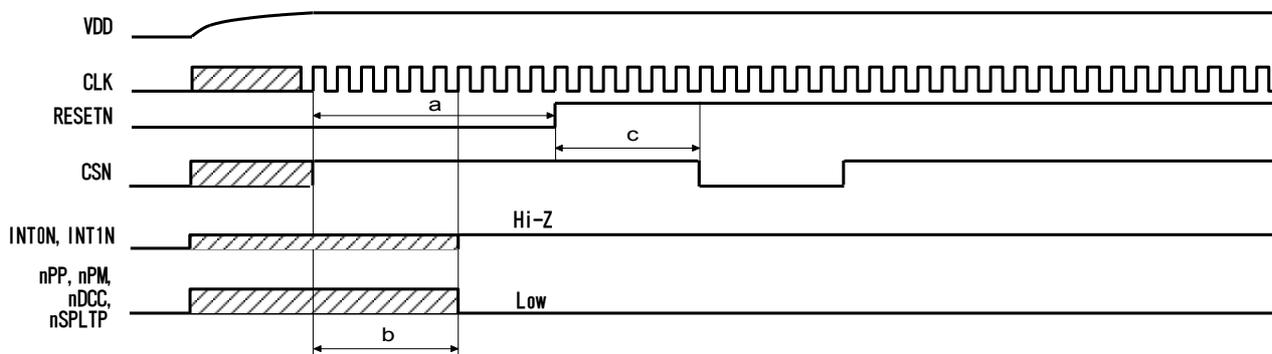
tCYC 為 CLK 的周期。

10.2.7 I²C 串列介面匯流排

現在、作成中です。

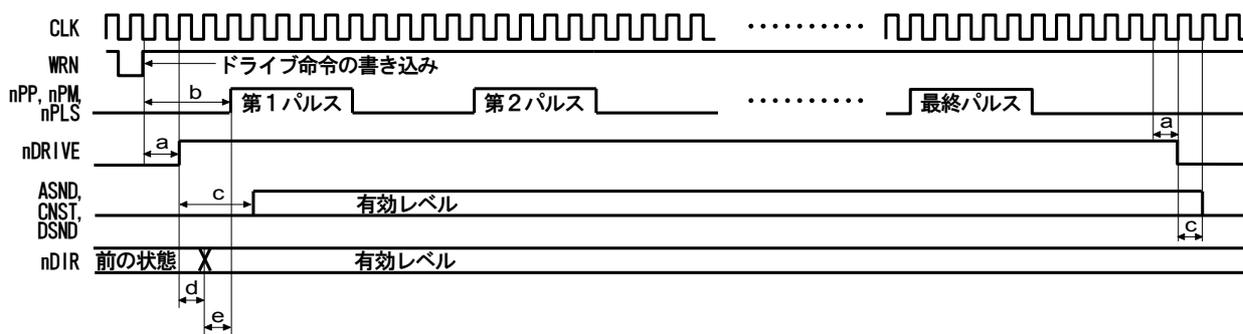
11. 輸出入訊號時序

11.1 電源開啟時序



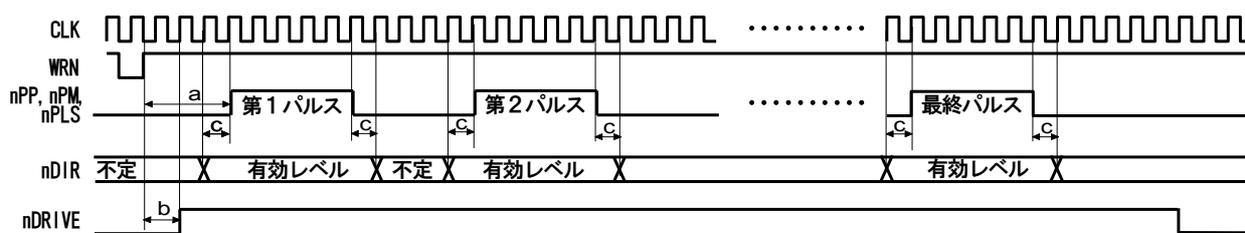
- 重置輸入訊號 (RESETN) 需要在CLK輸入後、CLK×8循環以上Low位準。
- 電源投入時的輸出訊號、RESETN為Low位準、CLK有輸入的狀態下、最多CLK×6循環後、上図所示位準確定。
- RESETN變為Hi位準上開始、最多CLK×4循環內、不能對本IC讀／寫。

11.2 驅動開始／結束時



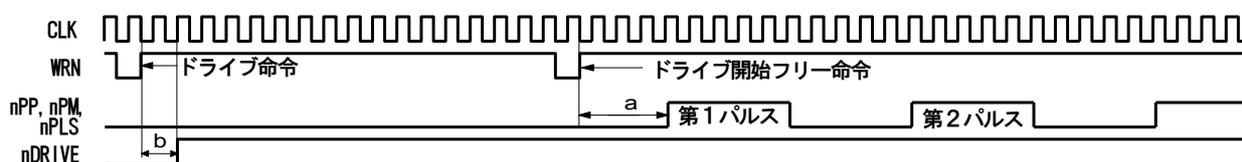
- 驅動狀態輸出訊號(nDRIVE)在驅動命令寫入時、WRN的↑起最多CLK2循環後變為Hi位準、最終脈衝的Low期間後凱使CLK1循環後回到Low位準。
- 驅動脈衝(nPP, nPM, nPLS)在本圖為正脈衝的時候。驅動命令寫入時、WRN的↑起最多CLK4循環後第1個脈衝輸出。
- ASND,CNST,DSND在、nDRIVE的↑開始CLK3循環後變為有効位準、DRIVE的↓開始CLK1循環後回到Low位準。
- 驅動輸出脈衝方式設定為1脈衝方式時nDIR(方向)訊號在nDRIVE的↑開始CLK1循環後變為有効位準。即使驅動結束後、下一個驅動命令寫入為止其位準會保持。
- nDIR(方向)訊號變為有効位準開始CLK1循環後輸出驅動脈衝(nPLS)的第1脈衝。

11.3 補間驅動開始



- 補間駆動時の駆動脈衝 (nPP, nPM, nPLS)、在驅動命令寫入時、WRN的↑開始最多CLK4循環後輸出第1個脈衝。
- nDRIVE在WRN的↑開始最多CLK2循環後變為Hi位準。
- 驅動輸出脈衝方式設定為1脈衝方式時 nDIR(方向)訊號在補間驅動時、驅動脈衝的 Hi 位準範圍及其前後 CLK1循環內、變為有效位準。(驅動脈衝: 正理論脈衝的時候)

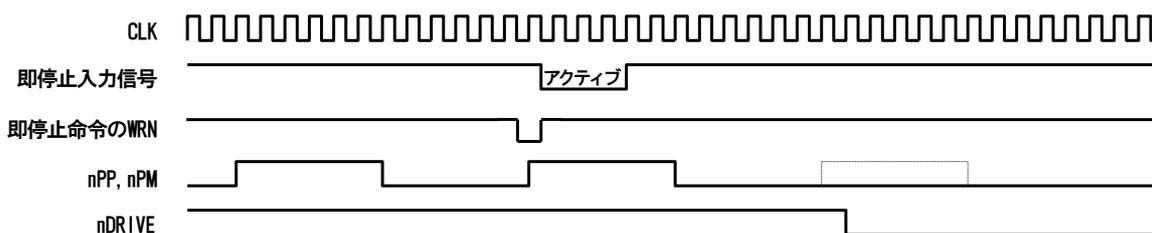
11.4 驅動開始釋放



- 各軸の驅動脈衝 (nPP, nPM, nPLS) 在驅動開始釋放命令寫入時、WRN的↑開始最多CLK4循環後輸出第1個脈衝。
- nDRIVE 在各軸的驅動命令寫入時、WRN 的↑開始最多 CLK2循環後變為 Hi 位準。

11.5 驅動即停止

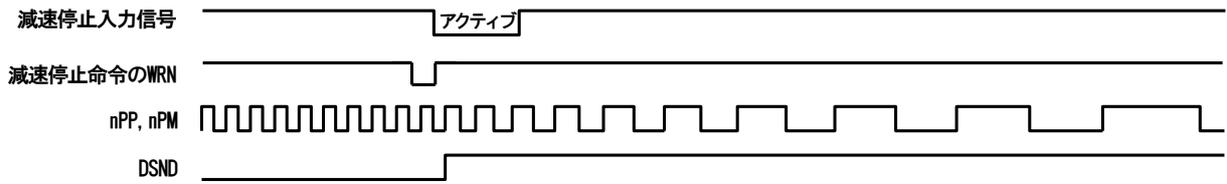
即停止輸入訊號時、即停止命令的動作時間。即停止輸入訊號為、EMGN、nLMTP/M(設定即停止模式時)、nALARM。
即停止輸入訊號變為動作位準、或是即停止命令寫入、現在輸出中的驅動脈衝在輸出後立刻脈衝輸出停止。



即停止輸入訊號即使在輸入訊號濾波無效的情況下、也需要CLK2循環以上的脈衝寬度。
輸入訊號濾波有效時、濾波的時間常數的值相對的輸入訊號遲延。

11.6 驅動減速停止

減速停止輸入訊號時減速停止命令的動作時間。減速停止輸入訊號為、nSTOP2~0、nLMTP/M(減速停止模式設定時)。
減速停止輸入訊號變為動作位準、或減速停止命令寫入時、現在輸出中的驅動脈衝輸出後開始減速。

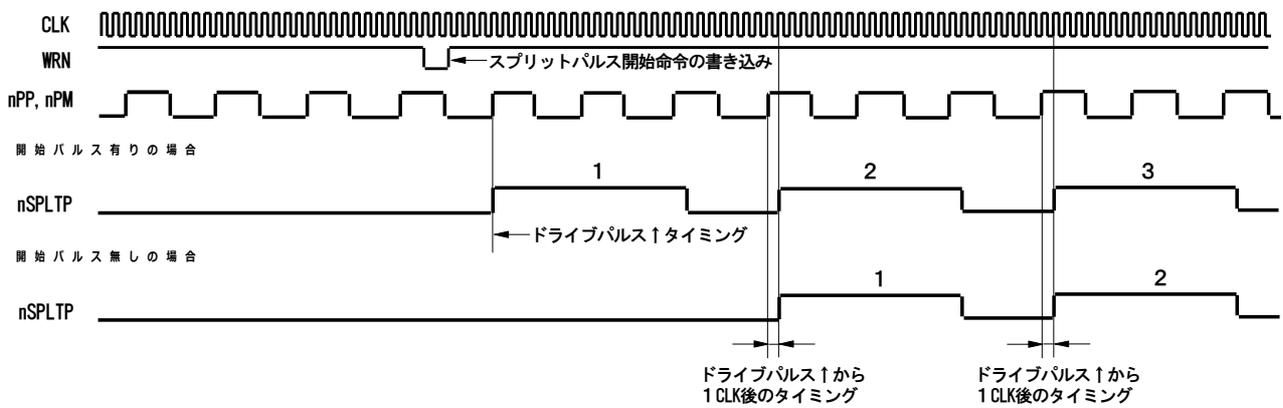


輸入訊號濾波有効時、濾波的時間常數的値相對應輸入訊號遲延。

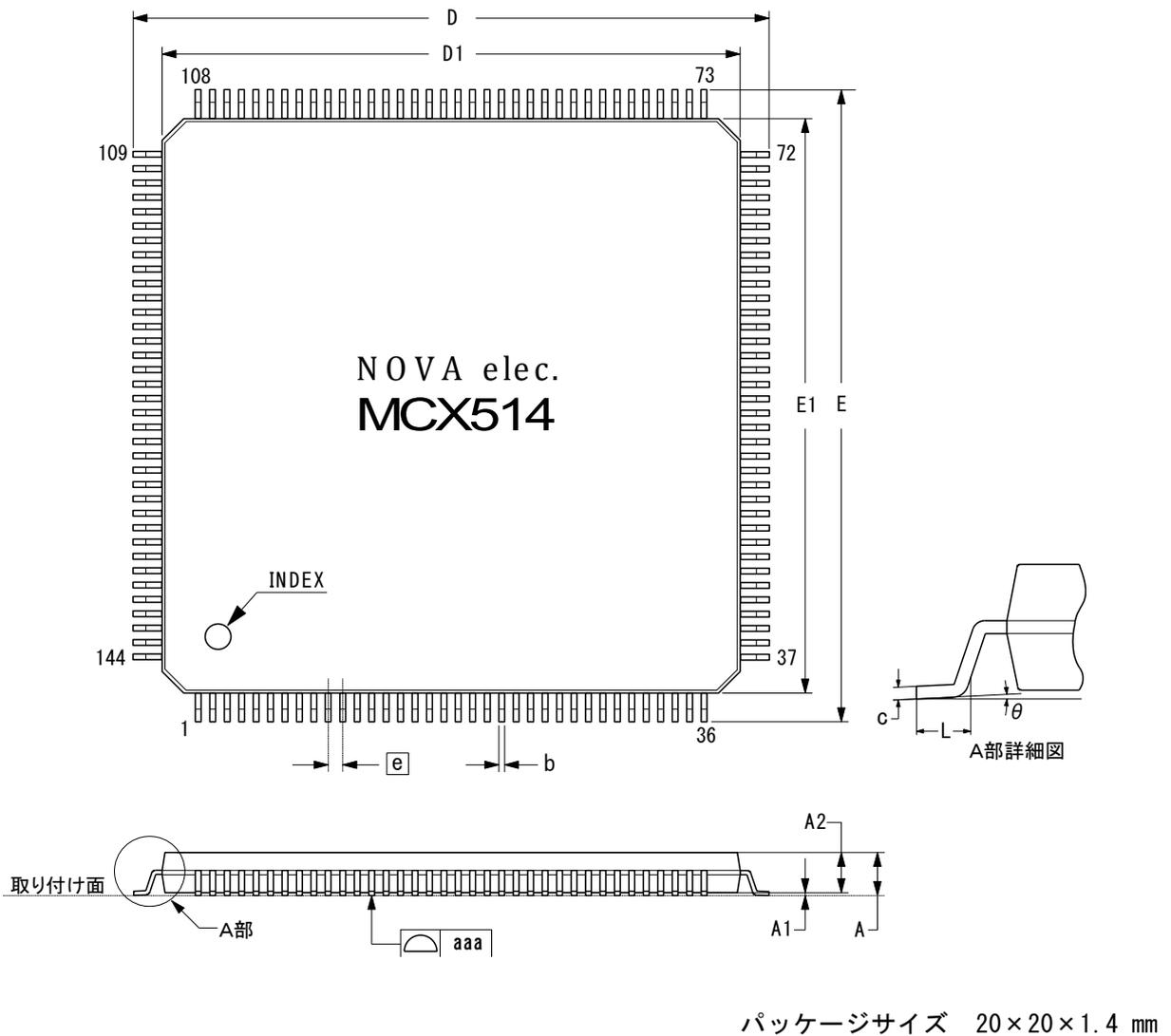
11.7 分割脈衝的詳細時間

分割脈衝的動作模式開始脈衝設為有的時候、僅第1個分割脈衝、驅動脈衝↑的時候分割脈衝變為Hi位準。第2個分割脈衝以後、驅動脈衝↑開始1CLK後分割脈衝變為Hi位準。因此、第1個分割脈衝的Hi位準寬度比第2個分割脈衝以後長1 CLK。

動作模式設定無開始脈衝時、全部的分割脈衝在驅動脈衝↑開始1CLK後分割脈衝變為Hi位準(正理論脈衝設定時)。



12. 外形寸法



記号	寸法 mm			説明
	最小	標準	最大	
A	—	—	1.7	焊錫面に封裝本体最上端部為止の高度
A1	—	0.1	—	焊錫面に封裝本体下端為止の高度
A2	—	1.4	—	封裝本体の上端到下端為止の高度
b	0.17	—	0.27	端子の寬度
c	0.09	—	0.2	端子の厚度
D	21.8	22	22.2	含端子封裝長度方向の最大長
D1	19.8	20	20.2	不含端子封裝本體の長度
E	21.8	22	22.2	含端子封裝寬度方向の最大長
E1	19.8	20	20.2	不含端子封裝本體の寬度
e	0.5			端子間距基準寸法
L	0.3	—	0.75	與焊錫面接觸の端子平坦部長度
θ	0°	—	10°	焊錫面與端子平坦部角度
aaa	0.08			端子最下面の均一性 (垂直方向の許容値)

13. 保管条件及推薦実装条件

13.1 本 IC の保管

本IC保管時請注意以下的項目。

- (1) 請勿拋投或掉落。包裝材破損後氣密性有可能損壞。
- (2) 保管時、防潮梱包未開封的狀態為30℃、85%RH 以下的環境、請12個月以內使用。
- (3) 超過有効期限時、排湿處理以125℃±5℃烘20小時以上36小時以內。烘烤次數請以2次為限。另、即使在有効期限內，防潮梱包的氣密有破損時也請做排湿處理。
- (4) 排湿處理的實施時、請注意防止靜電對元件的破壞。
- (5) 防潮梱包開封後、30℃、70%RH 以下的環境條件下保管、請7日以內の実装。如果超過上述許容放置期間時，請実装前務必做烘烤處理。

13.2 烙鐵標準実装条件

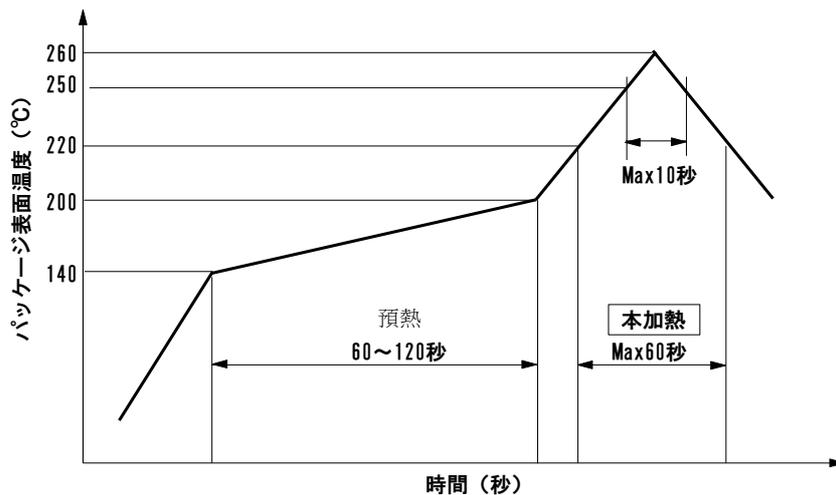
本IC的烙鐵的標準実装条件如下。

- (1) 実装方法： 烙鐵(端子部分加熱)
- (2) 実装条件： 端子温度:350℃以下、時間:5秒以內、回数:2回以下

13.3 迴風爐標準実装条件

本IC的迴風爐標準実装条件如下。

実装方法	(1) 赤外線 (2) 温風 (3) 赤外線・温風併用
最高迴風爐温度(封裝表面温度)	260℃以下
250℃以上の時間	10秒以內
220℃以上の時間	60秒以內
預熱的溫度 140~200℃的溫度	60~120秒
迴風爐次數	2回以內



MCX514迴風爐実装条件