



SYNTEC
TECHNOLOGY CO.,LTD.

桌上型应用手册(旧)

匯出日期: 2023-05-10

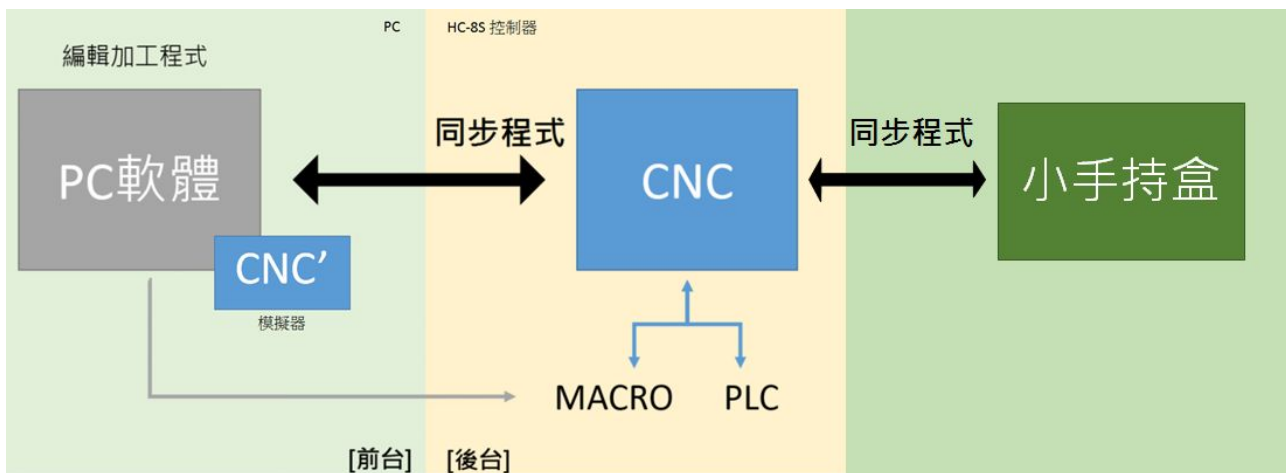
修改日期: 2021-02-23

版本记录

项次	更改内容	更改日期	作者	版本
01	初版草稿	2015/12/09	张景嫻	V1.0
02	初版定稿	2015/12/11	张景嫻	V1.1
03	章节顺序更换, 内容、图片补详	2015/12/25	商庭祯	V1.2
04	新增第七章	2016/2/2	黄右年	V1.3
05	统整修改 (editing)	2016/11/??	张景嫻	v1.4 (editing)

- 零、点胶机软体架构
 - 点胶机的运作流程:
 - PC操作画面客制
 - 透过PLC呼叫MACRO
 - 3. 使用扩充G码
 - 3-1 eHMI呼叫G码
 - 3-2 在MACRO直接呼叫G码
 - 4. MACRO与PLC警报使用
 - 5. PC软体客制步骤
 - 6. 副程序呼叫
 - 7. 单/双Y视觉功能修改
 - 7-1 视觉页面对应修改档案
 - 7-2 相机数量比较
 - 7-3 目前光学点分配比较
 - 7-4 操作流程
 - 7-5 单/双Y MACRO比较
 - 7-6 单/双Y R值对照

1 零、点胶机软体架构

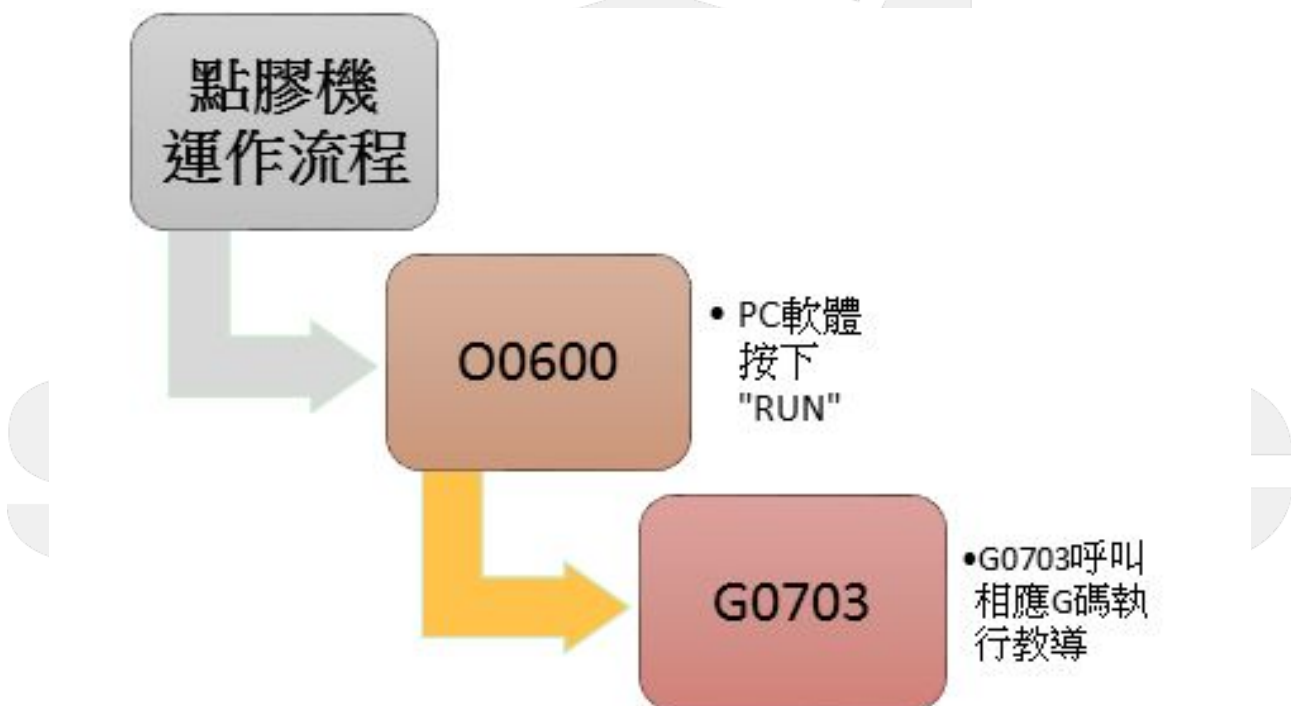


PC端：PC软体DispensingMachine.exe 可进行步骤表单编辑，和控制器连线後可控制CNC执行对应的动作。

CNC端：主要由MACRO与PLC构成，另外有同步程序在PC与控制器之间进行沟通，PC与CNC之间透过R值进行沟通。

小手持盒：透过同步程序与控制器沟通，可进行步骤表单编辑以及轴控等功能，操作方便简易。

1.1 点胶机的运作流程：



1. 开启控制器後开启PC软体，在PC软体提供的操作介面上，进行动作的编程以及参数的设定。

动作编程：提供4000行的步骤表单可以编辑，每行步骤代表一个动作(每个动作有独立的OPCode数字)，

且每个步骤有最多9个引数可以设定，所有的步骤设定值会用R值存取。

因此一行步骤需要使用1个OPCode与9个引数共10个R值，总共4000行会需要共 $4000 \times 10 = 40000$ 个R值，摆放於R20000~R59999。

参数设定：包含机构校正、输出状态、光学点学习等参数可以在PC软体介面上设定，这些参数会同步到控制器上。

PLC辅助：以上参数设定有些需要绑定使用流程，而流程的处理会透过控制器上的PLC执行，例如寻原点的流程就是在PLC上处理各轴回原点的顺序，以及到达原点后会导通跳脱讯号，让轴动停止。其他包含光学点学习、IO讯号、Modbus通讯等功能无法在MACRO内处理，就会在PLC设计执行流程。

* 以上提到的使用R值请参考附录1的R值定义表

2. PC软体按下执行(RUN)会导通PLC加工状态，让控制器执行主加工程序。

主加工程序：00600这只MACRO是控制器的主加工程序，在主加工程序里会进行机台的初始化，并呼叫G703解译前项设定的步骤R值(R20000~R59999)，执行对应的动作MACRO。

动作MACRO：每一个步骤动作会有一个对应的执行MACRO，亦提供客制MACRO可以自定义动作内容。

* MACRO列表请参考附录2，客制方式会在後面的章节作说明

以上的运作流程，在操作画面、PLC与MACRO动作中开放很大的客制空间，以下的章节会分别针对这三种工具的使用与规格作说明。

1.2 PC操作画面客制

1.3 透过PLC呼叫MACRO

呼叫流程如下：

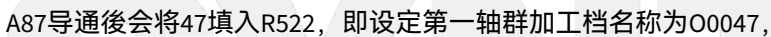
- 决定使用轴群
- 参照R值定义表找到对应的主程序号码(例如第一轴群对应到R522，下图为第一轴群的R值定义表节录)

SYNTEC

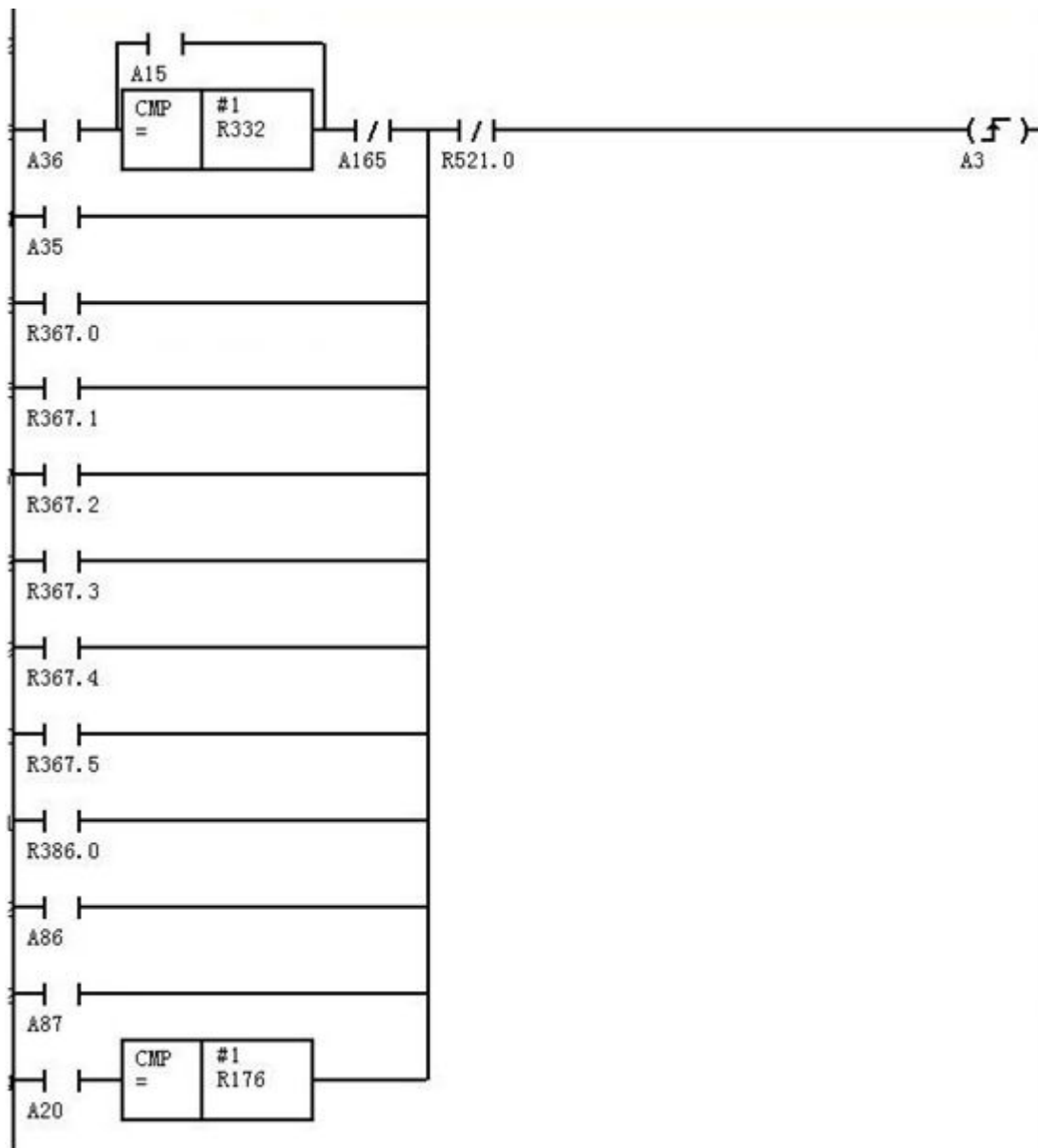
- 将主程序号码设成欲呼叫的MACRO号码，输入的号码将对应到MACRO主程序名称(必须为0开头後面帶四码数字)
- 於PLC内使用上缘触发R520即可执行呼叫

首先点胶机执行时会呼叫主程序O0600，下图PLC功能是判定第四轴移动方式，若是固定式，则设定第一轴群执行档为O0047。

先判断R176(第四轴移动方式)是否为1, 若符合则导通A87。



零、点胶机软体架构-5



当A3、A160~164(马达状态)都满足时，将上缘导通R520.0，执行R522设定的加工档，此例中即为O0047。



1.4 3. 使用扩充G码

当控制器内建的基本G码无法满足应用需求时，使用者可以自行撰写MACRO作为扩充G码以满足自身需求。G码的撰写格式以及指令与函式表请参考Help中的MACRO发展工具操作手册。

扩充G码命名规则：

目前支援G200~G999为扩充G码

G200所对应之MACRO，其档名为G0200；

G200.1所对应之MACRO，其档名为G0200001；

G200.10所对应之MACRO，其档名为G0200010；

扩充G码程序储存路径：

DiskC:\OpenCNC\MACRO

使用G码的方式有两种：

1. 在eHMI页面中定义动作去呼叫G码
2. 在MACRO直接呼叫G码

1.4.1 3-1 eHMI呼叫G码

在eHMI页面作画规划时，可以动作设置里设定G码执行指令，页面整合到机台上启动程序时，标准人机(CNCMon)会接收到此动作，并执行指定的G码。

设定方式：开启动作设定视窗，功能选项总表选择==加工档==分类下的"执行扩充G码"，在输入栏中输入G码代号并点击加入按钮後左边动作列表会产生相应的指令，最後点击确定完成设定。

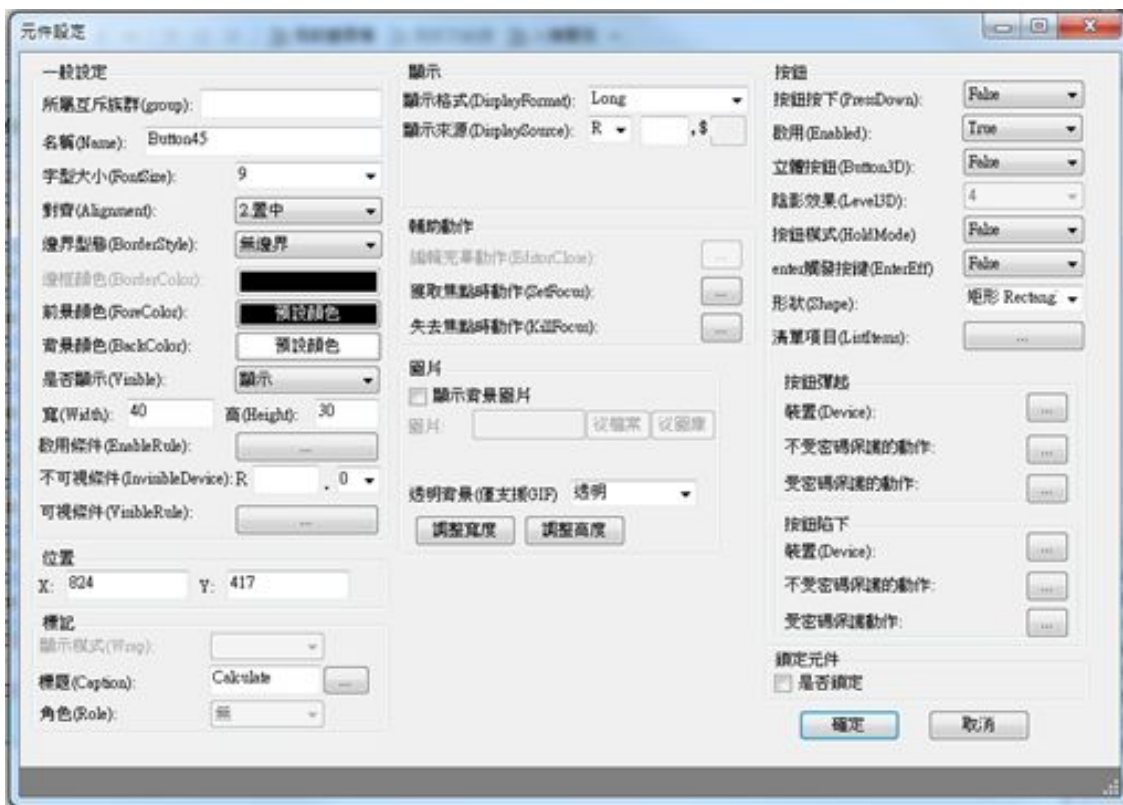
范例



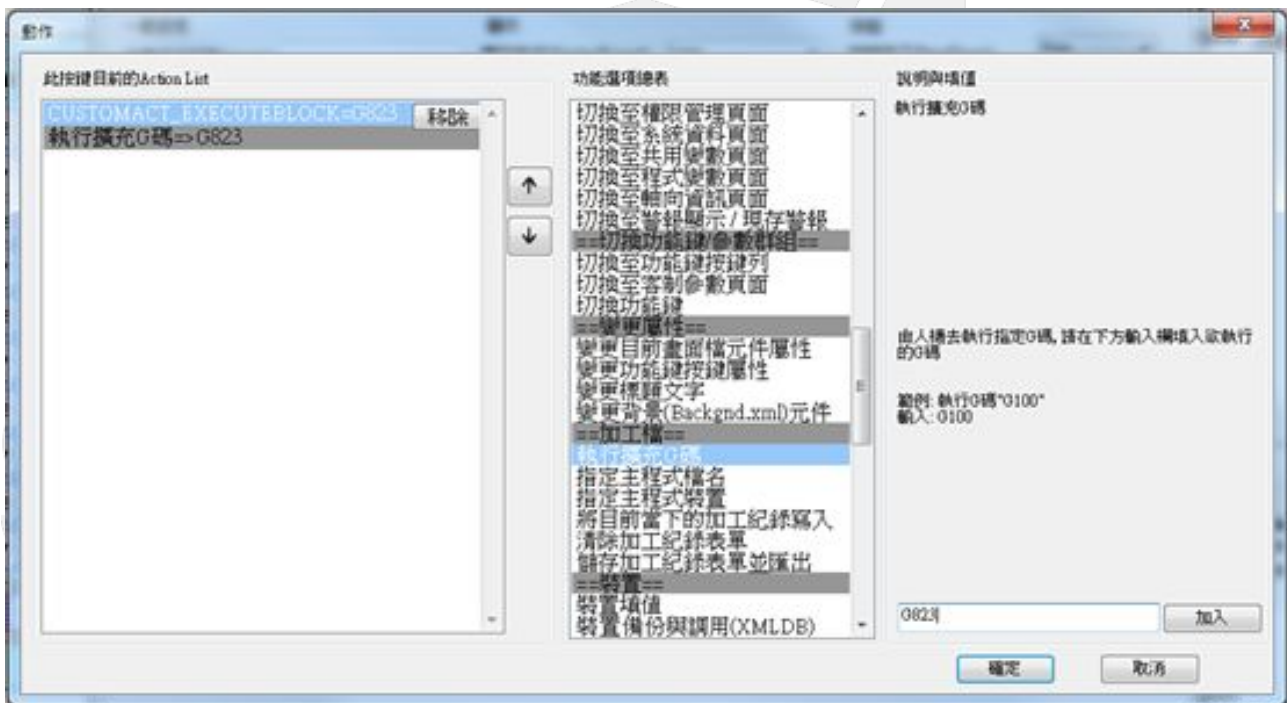
上图为测高功能页面，下方"针头vs表头关系设定"栏位中，可以分别设置针头与表头的座标值，设定完後程序要计算两者的位置差值，但eHMI没有运算指令，因此必须设定G码呼叫MACRO作计算，设置的方式如下：

--右键点击或双击"计算"按钮进入元件设定页面

SYNTEC



--此例利用按钮弹起设置动作，点选"不受密码保护的行动"进行设置：



选择==加工档==分类下的"执行扩充G码"，在输入栏中输入G码代号并点击加入按钮，确认左方动作列表有将此新增指令列入即可按确定完成设定。

设定完后当此页面的计算按钮被点击时，就会呼叫G823执行内部的计算指令。

1.4.2 3-2 在MACRO直接呼叫G码

直接写入G码名称呼叫，如上图MACRO中判定为单点点胶的情形，则呼叫G601扩充G码运作。

```
%@MACRO

//G703 教導步驟判斷

CASE @[120000+@100110*10] OF

    //不做事
    0:
        #91:=#91+1; //執行空單節記數，避免核心q節爆掉
        ;

    //單點點膠
    1:
        G601;
```

1.5 4. MACRO与PLC警报使用

I MACRO警报使用方式

在MACRO内使用ALARM指令并指定相应的警报编号。

输入型式如下：

ALARM(10016,"区域涂布找不到直线点胶终点");

ALARM括号内输入警报编号与显示字串

I PLC警报使用方式

警报对应到的R值如下表

SYNTEC

R44	PLC Alarm Message #65 ~ #80 #315 ~ #330	由使用者自行定義的警告訊息，對應警報號碼65號到80號。 Bit 0~15 #65 ~ #80 PLC Alarm Message, save to history. Bit16~31 #315 ~ #330 PLC Warning Message, no save to history.	唯寫
R45	PLC Alarm Message #81 ~ #96 #331 ~ #346	由使用者自行定義的警告訊息，對應警報號碼81號到96號。 Bit 0~15 #81 ~ #96 PLC Alarm Message, save to history. Bit16~31 #331 ~ #346 PLC Warning Message, no save to history.	唯寫
R46	PLC Alarm Message #97 ~ #112 #347 ~ #362	由使用者自行定義的警告訊息，對應警報號碼97號到112號。 Bit 0~15 #97 ~ #112 PLC Alarm Message, save to history. Bit16~31 #347 ~ #362 PLC Warning Message, no save to history.	唯寫
R47	PLC Alarm Message #113 ~ #128 #363 ~ #378	由使用者自行定義的警告訊息，對應警報號碼113號到128號。 Bit 0~15 #113 ~ #128 PLC Alarm Message, save to history. Bit16~31 #363 ~ #378 PLC Warning Message, no save to history.	唯寫
R48	PLC Alarm Message #129 ~ #144 #379 ~ #394	由使用者自行定義的警告訊息，對應警報號碼129號到144號。 Bit 0~15 #129 ~ #144 PLC Alarm Message, save to history. Bit16~31 #379 ~ #394 PLC Warning Message, no save to history.	唯寫
R49	PLC Alarm Message #145 ~ #160 #395 ~ #410	由使用者自行定義的警告訊息，對應警報號碼145號到160號。 Bit 0~15 #145 ~ #160 PLC Alarm Message, save to history. Bit16~31 #395 ~ #410 PLC Warning Message, no save to history.	唯寫

新代提供96个PLC Alarm、96个PLC Warning以及319个PLC Hint供使用者自行定义，其中各Alarm/Warning/Hint编号与R bit对应关系请参阅下表：

型態	R 位元(Bit)	警報編號
PLC 警報訊息 (Alarm Message)	R44.0 ~ R44.15	65 ~ 80
	R45.0 ~ R45.15	81 ~ 96
	R46.0 ~ R46.15	97 ~ 112
	R47.0 ~ R47.15	113 ~ 128
	R48.0 ~ R48.15	129 ~ 144
	R49.0 ~ R49.15	145 ~ 160
PLC 警告訊息 (Warning Message)	R44.16 ~ R44.31	315 ~ 330
	R45.16 ~ R45.31	331 ~ 346
	R46.16 ~ R46.31	347 ~ 362
	R47.16 ~ R47.31	363 ~ 378
	R48.16 ~ R48.31	379 ~ 394
	R49.16 ~ R49.31	395 ~ 410
PLC 提示訊息 (Hint Message)	R590.1 ~ R590.31	1 ~ 31
	R591.0 ~ R591.31	32 ~ 63
	R592.0 ~ R592.31	64 ~ 95
	R593.0 ~ R593.31	96 ~ 127
	R594.0 ~ R594.31	128 ~ 159
	R595.0 ~ R595.31	160 ~ 191
	R596.0 ~ R596.31	192 ~ 223
	R597.0 ~ R597.31	224 ~ 255
	R598.0 ~ R598.31	256 ~ 287
	R599.0 ~ R599.31	288 ~ 319
備註	1. PLC警報(Alarm)可儲存於歷史警報中；但PLC警告(Warning)與PLC提示(Hint)則無法儲存。 2. PLC提示(Hint)的識別方式有別於PLC警報(Alarm)與PLC警告(Warning)，因此警報編號重複無妨。 3. PLC提示(Hint)不需自保，警報視窗即可一直存在；但PLC警報(Alarm)與PLC警告(Warning)則必須透過自保才行。	

范例

下图PLC中用R370控制警報内容，当R379被填入不同的值时会导通不同的警報R值，此例中当R370等於1时将显示R44.0警報(螺旋型矩形涂胶 前无 线段点胶终点)。



1.6 5. PC软体客制步骤

若现有的步骤功能无法满足需求，可以二次开发PC软体客制步骤，开发方式依照以下流程进行。

--设定步骤画面，画面名称为TEACH_0x5 (为一码整数)。注意步骤编号5为16进制，在MACRO中呼叫时，必需转为10进制。

Ex. 步骤编号52则画面名称为TEACH_0x52.xml，在macro中呼叫时则使用82。

--画面中引数输入框依照引数编号命名为P1、P2...以此类推。

--於步骤画面新增确定按键，连结新增步骤 (TEACH_0x50 ~ TEACH_0x5F)。

--DMMMain.xml主画面档加入相应步骤按钮。

--将新增步骤资讯加入Utility.dat中。

Utility.dat存在两个不同位置，两者之间没有关联，所以在更新时，请务必於下列两个位置的Utility.dat都做更新：

1.	PC软体	DiskC\OpenCNC\Bin
1.	控制器	DiskC\Shared\dotnet

范例:

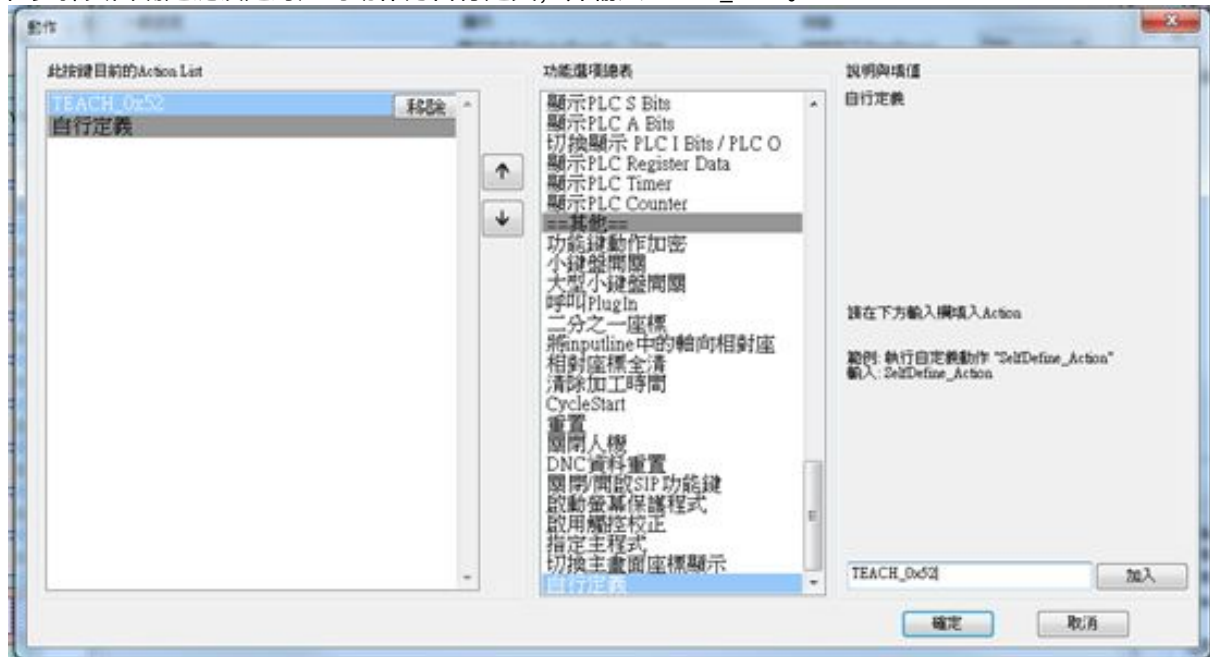
考量到加工时会有高度误差存在，因此需要客制一测高步骤，让机台去量测高度误差并作对应的补偿。其步骤客制方式如下：

1. 设定此步骤编号为52，新增一参数画面档，并依照步骤编号命名为TEACH_0x52。



2. 如上图，此步骤设计有两个引数，分别为测高点位编号以及补偿步骤位置。在画面中设计两个引数的输入框，并分别命名为P1与P2，程序运作时才能读取到对应的值。

- 在步骤页面确定键设定对应的动作为自行定义，并输入TEACH_0x52。



- 将新增步骤加入主画面DMMMain.xml中，并设定好页面连结。



此例中测高步骤为进阶功能，设定在进阶功能的下拉选单中，将此步骤加入下拉式选单项目。



并设定动作为呼叫第一个步骤所设计的步骤页面，选取自行定义并输入 PopUpXml_TEACH_0x52.xml_X=200_Y=200，XY设定视窗尺寸。



5. 最後將设计的步骤加入Utility.dat当中。

Utility.dat 客制方式

Utility.dat档案中以逗号分隔每一个参数，以分号做为下拉式选单选项区隔，写法如下：

步骤後两位数,步骤名称,参数型态,参数P1所显示的数值型态(单位),参数P2为下拉式选单中选项1;选项2;选项3

1. 参数型态有NONE跟FULLAXIS两种，差别如下：

型态	可用引数(个)	使用时机
NONE	9	步骤无需纪录当下座标(如时间、加速度)
FULLAXIS	4	步骤需纪录当下座标(如单点点胶)

1. 参数显示的数值型态

- 若为整数，单位为A --> 0(A)
- 若为小数後两位数，无单位 --> 2()
- 下拉式选单需与eHMI参数画面档中的对应值相互对应。
即eHMI下拉式选项说明Yes，对应值为1，Utility.dat档案中若资料显示字串为Y，参数写法为1=Y
- Utility.dat除了PC软体需要使用外，若手持盒也需要同步此客制步骤，需将此档案置放於
DiskC\Shared\dotnet资料夹中。

此例中的测高客制步骤使用测高点位编号以及补偿步骤位置两个引数，精度都为整数且没有单位，因此在Utility.dat输入以下资讯：

- ü 步骤编号52
- ü 步骤名称"测高(步骤)"
- ü 没有使用轴向引数输入NONE
- ü 第一引数精度为整数，输入0，无单位故括号内空白
- ü 第二引数原理同第一引数

```

48 52,测高(步骤),NONE,0(),0()
49 53,测高(標記),NONE,0(),0(＃)

```

1.7 6. 副程序呼叫

在步骤编辑时，有时会有常用步骤会需要重复编辑导致耗时增加，因此有一步骤为副程序呼叫，供使用者预先编辑好常用的步骤区段，在主程序编辑时可以随时呼叫。此步骤使用方式如下：

- 在程序的後段行数(如第3000行)编辑常用步骤
- 使用位於进阶功能中的呼叫副程序步骤，操作页面如下：



设定呼叫副程序的加工位置，并设定副程序所处的行数，设定完後，当执行到副程序呼叫步骤时，就会移到设定的加工位置执行副程序内容。

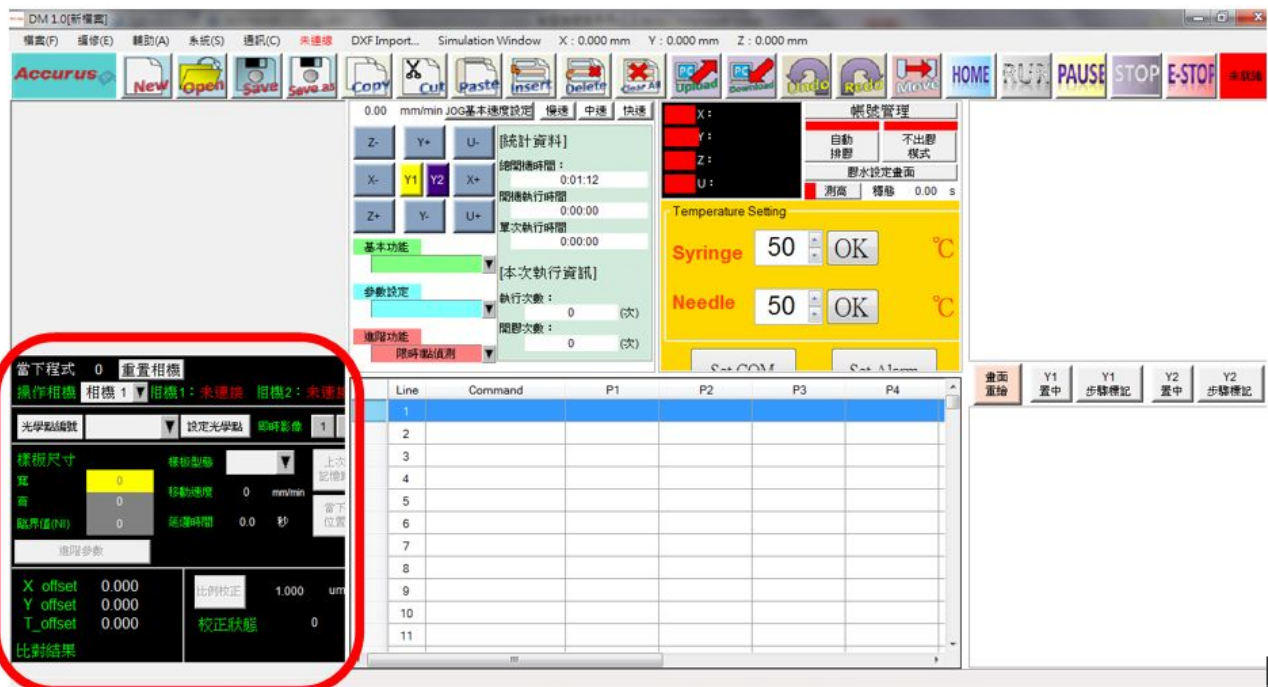
1.8 7. 单/双Y视觉功能修改

(以下介绍有关视觉功能的开发流程。)

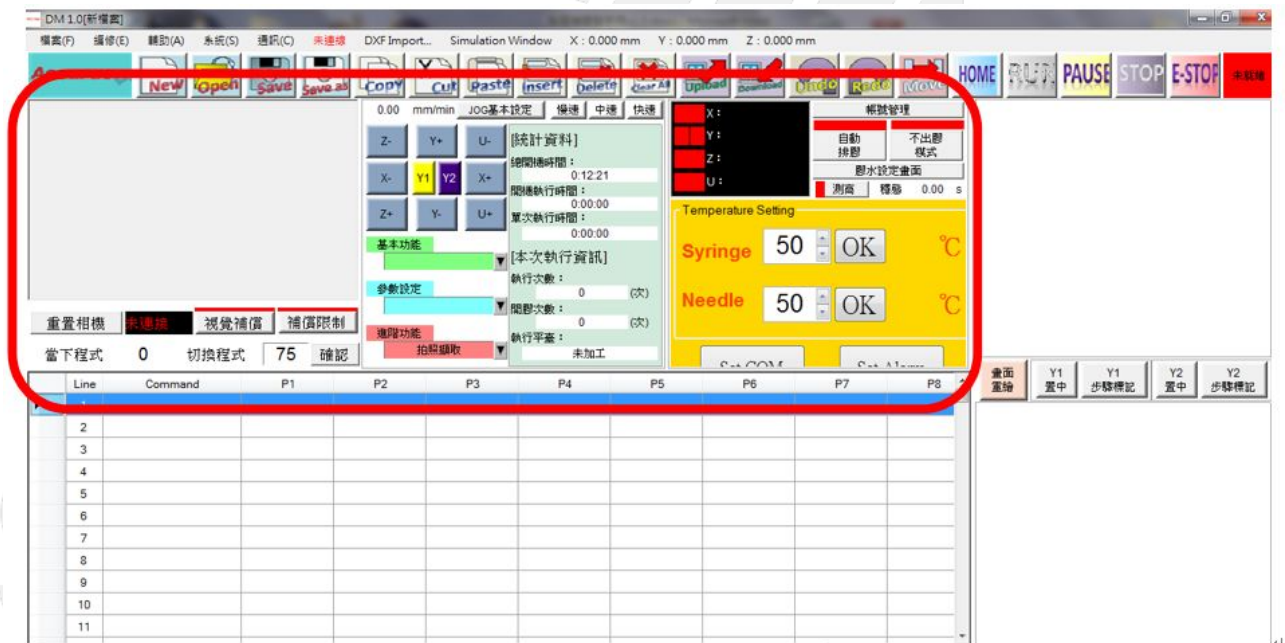
1.8.1 7-1 视觉页面对应修改档案

在使用单/双Y点胶机PC程序时，如果有客制修改视觉画面的需求，使用者可以从DiskC\OpenCnc Shared\PrdMMIRes\VisionRes\Common\AppData中的Manipulate_one修改。(下图为双Y程序范例)

SYNTEC



若是想在主页面新增功能按钮，则至DiskC\OpenCNC\Bin\OCRes\Common\AppData中的DMMMain修改。(下图为双Y程序范例)



如果想新增或更改UTILITY中的功能与页面，也可以从DiskC\OpenCNC\Bin\OCRes\Common\AppData中新增或修改 (TEACH_0xXX)。

1.8.2 7-2 相机数量比较

点胶机可以根据相机数量分成单CCD(只有一台)、多CCD(一台以上)，目前的PC程序中，单Y的点胶机预设成单CCD，而双Y则是多CCD，包括CCD1、CCD2。

單 Y PC 程式→單 CCD



雙 Y PC 程式→雙 CCD



在使用点胶机与PC软体之前，先确认控制器的option环境，单CCD的4、16需要ON，而双CCD则是4、17需要ON。

1.8.3 7-3 目前光学点分配比较

每台相机都能拍照纪录光学点，换言之，每台CCD都有分配到一段空间储存光学点。使用者可以依据需求分配一台相机的光学点分配，但要注意一台相机最多对应到20个光学点。

以目前的PC程序来说：

- 单Y的CCD1拥有20个视觉对位的光学记忆点。

- 双Y的CCD1拥有16个视觉对位的光学记忆点，

1~8是记忆Y1轴上的光学点，

9~16是记忆Y2轴上的光学点，

另外还有4个单点拍照的光学记忆点。

1、2是拍摄Y1轴上的光学点，

3、4是拍摄Y2轴上的光学点。

CCD2则拥有4个单点拍照的光学记忆点。

1、2是拍摄Y1轴上的光学点，

3、4是拍摄Y2轴上的光学点。

(小提醒:单点拍照只是单纯拍照，视觉对位则是使用两个光学点来计算物件的偏移量。)

单Y的20个视觉对位的光学记忆点。



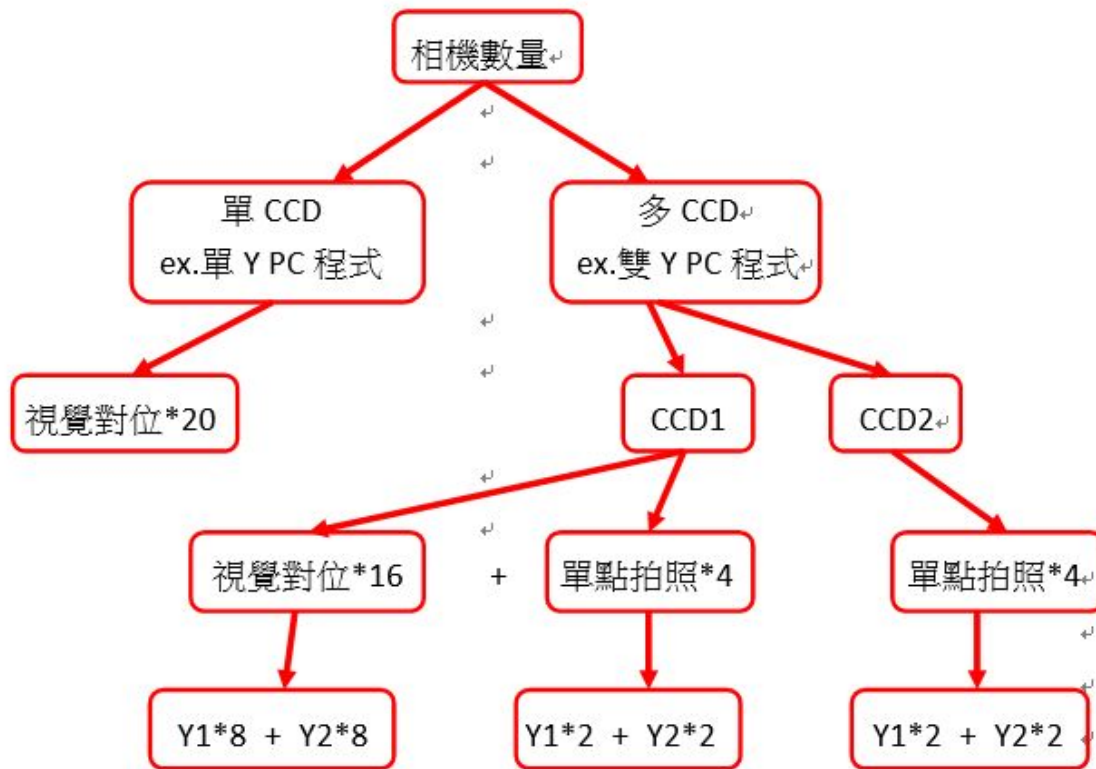
双Y的CCD1中有16视觉对位的光学记忆点+4个单点拍照的光学记忆点。



而双Y的CCD2则只有4个单点拍照的光学记忆点。(如上右图)

光学点分配示意图:

SYNTEC



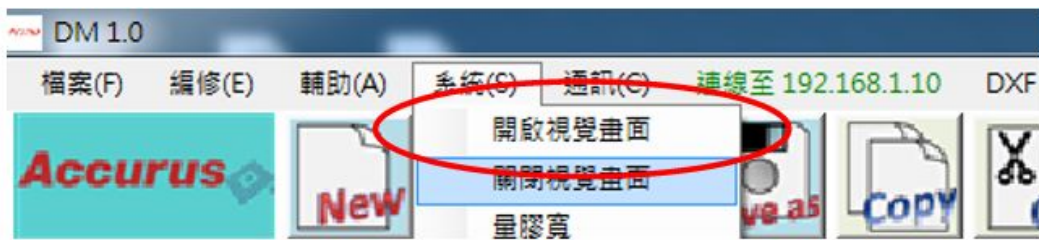
1.8.4 7-4 操作流程

以下为双Y PC程序在视觉画面的操作流程范例。

首先，开启视觉核心PC_VLocate.exe

接着点击系统中的开启视觉画面

SYNTEC



可以看到起始畫面相機狀態顯示未連接



可以看到起始画面相机状态显示未连接



遇到这种情况可以先选择相机à按下复位相机à点选即时影像

就可以看到相机的画面在视觉的部分出现，而相机状态也会依序变成「闲置」à「操作中」



接着选择光学点编号a按下设定光学点就可以看到下面的选项可以启用

此时选择的光学点，如果是CCD1则是纪录至R15012，值根据选择的光学点依序是0~19(0~15：视觉对位，16~19：单点拍照)。如果选择CCD2则是纪录至R15013，值根据选择的光学点依序是16~19(单点拍照)。

另一方面，按下设定光学点则会呼叫MACRO O0060，读取光学点性质。

(7-5、7-6会有更多的MACRO、R值整理介绍)

SYNTEC



接著按下JOG面板控制按鈕將鏡頭移動至想要拍照(對位)的位置à設定樣板型態、移動速度(>0)、延遲時間(>0)等參數

此時選擇的樣板型態，如果是CCD1則是紀錄至R15008，值根據選擇的樣板有所不同（單點拍照：6）。如果選擇CCD2則是紀錄至R15009，只能選擇單點拍照(6)。

SYNTEC



在学习之前，先按下复位相机，因为相机要在闲置的状态下才能学习。
 流程：按下复位相机→按下当下位置(上次记忆点)→按下学习→按下比例校正
 (如果接下来纪录的拍照对位光学点都在同有相同的z值，则只需校正一次。)

SYNTEC



按下当下位置时会呼叫MACRO O0066，强制开启学习模式，若是想在上次记忆点学习则按下上次记忆点，同样会呼叫MACRO O0066，但座标的值会有所不同。

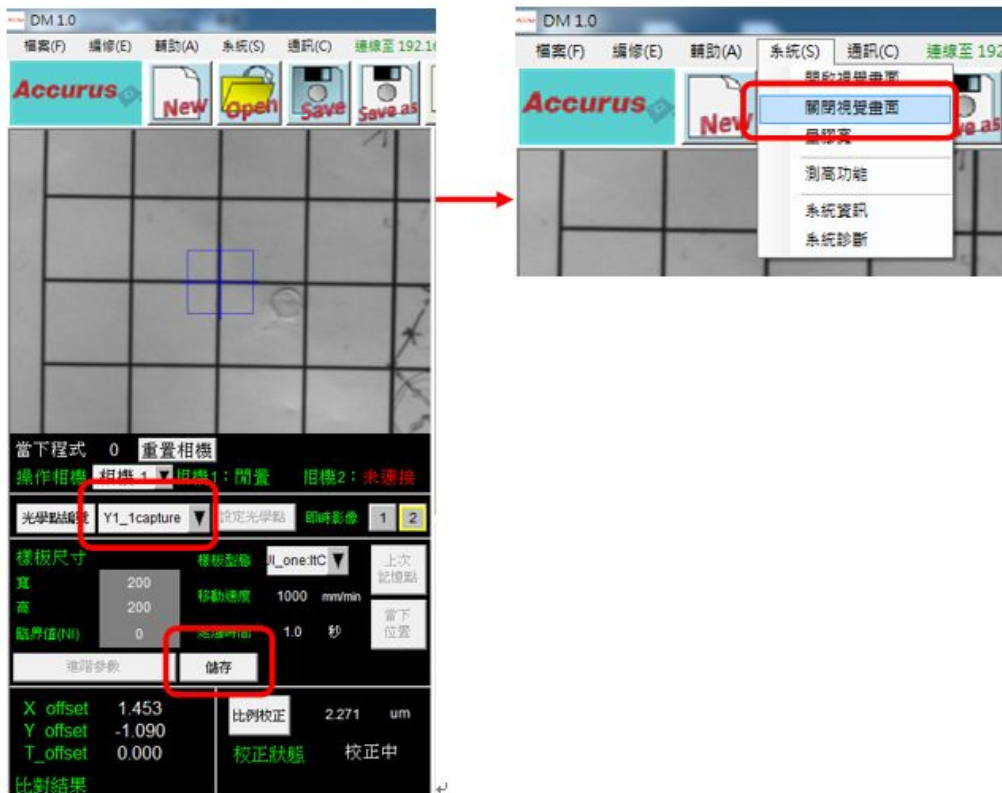
按下学习则会呼叫MACRO O0061写光学点参数值。

按下比例校正则呼叫MACRO O0062执行轴耦合校正动做。

若是要执行单点拍照而不是对位拍照，选择第17~20个光学点，执行同样的步骤，但最後是选择储存，因为单点拍照不需要对位学习或是校正。

设定完所有的光学点后，点击系统中的关闭视觉画面。

SYNTEC



最後补充执行程式的部分

首先點選左上角的NEW

如果要先安排單點拍照，則選擇進階功能中的拍照擷取，並設定各參數。

按下確定後會呼叫0x2B(G643)

最後可以看到流程顯示在表格中

接著若是要執行視覺對位，選擇進階功能中的視覺對位(拍照) 呼叫0x2C(G644)

設定完成後可以再選擇進階功能中的視覺對位(補償) 呼叫0x2D(G645)

設定好總流程後加上基本功能中的結束程序，並前往程序2組(原本在第0組)

選擇基本功能中的雙Y平台加工程序，設定執行第0組。

最後登入系統，按下RUN，就能夠執行程式了。

1.8.5 7-5 單/雙Y MACRO比較

以下說明、對照幾個單/雙Y視覺功能使用的MACRO程序。

按下設定光學點：

觸發00060(讀光學點參數值)，此程序賦予R300值(手持盒功能碼及參數)以讀光學點參數資料，在此設定移動速度、延遲時間等等。

按下当下位置:

触发**00066**(轴动到拍照点後强制开启学习模式), 在单Y程序中会先判断是否按下上次记忆点(R1142=0), 如果有则先把座标改程上次记忆点座标, 接着使用**G0130**(学习模式, 相机拍照); 而在双Y的程序中, **00066**会先判断始用哪一台相机, 接着用R15012、R1142判断是否为视觉对位

(需要先做轴耦合 ex. G10 L1000 P1036 R1)

按下学习:

触发**00061** (写光学点参数值), 在单Y程序中把程序座标@100721...标准化後写入R 106410...(分别有X、Y、Z、U、V), 双Y程序则是多判断始用哪一台相机後写入。

按下比例校正:

触发**00062** (执行比例校正), 呼叫**G132**执行比例校正。 **G132**我在双Y程序中需要多判断校正的位置是Y1或是Y2, 才能够有正确的校正值。

G643: **UTILITY**中的0x2B, 对应拍照撷取的功能, 在主画面中选择单点拍照後会触发, 目前预设只有双Y PC程序中有这个功能。程序会读取座标值及相机选择再执行拍照。

G644: **UTILITY**中的0x2C, 对应视觉对位(拍照)的功能, 在主画面中选择视觉对位(拍照)会触发, 单/双Y都有此功能。

G645: **UTILITY**中的0x2D, 对应视觉对位(补偿)的功能, 在主画面中选择视觉对位(补偿)会触发, 单/双Y都有此功能, 会由两个拍照点计算物件偏差的值。

1.8.6 7-6 单/双Y R值对照

以下说明几个重要R值功能、意义。

座标存取:

单Y程序:

视觉对位1~20(编号) 光学点的X、Y、Z、U、V储存於R6410~R6509

Ex. 第一个光学点的Xà6410 Yà6411 Zà6412 Uà6413 Và6414

第二个光学点的Xà6415 Yà6416 Zà6417 Uà6418 Và6419

双Y程序:

CCD1:

视觉对位1~16(编号) 光学点的X、Y、Z、U、V储存於R6410~R6489

单点拍照17~20(编号) 光学点的X、Y、Z、C、B储存於R6490~R6509

CCD2:

单点拍照1~4(编号) 光学点的X、Y、Z、C、B储存於R6381~R6400

光学点编号存取:

单Y程序:

只有一台相机, 光学点编号1~20储存於R15012, 对应值为0~19。

双Y程序:

CCD1:

视觉对位1~16(编号) 储存於R15012, 对应值为0~15。

单点拍照17~20(编号) 储存於R15012, 对应值为16~19。

CCD2:

单点拍照1~4(编号) 储存於R15013, 对应值为16~19。

样板编号存取:

单Y程序:

只有一台相机, 样板编号储存於R15008。

双Y程序:

CCD1:

视觉对位1~16(编号) 储存於R15008, 此时无单点拍照选项, 值没有6。

单点拍照17~20(编号) 储存於R15008, 此时只能选择单点拍照, 值为6。

CCD2:

单点拍照1~4(编号) 储存於R15009, 此时只能选择单点拍照, 值为6。

误差相关:

相机状态:

SYNTEC