

埃夫特智能装备股份有限公司

简单码垛说明书

版本号：v0.4

软件二组

2019-12-11

修订历史

版本	日期（年/月/日）	原因（创建/修订）	创建或修订人
V0.1	2019.07.05	创建	王连兵
V0.2	2019.08.10	修订	王连兵
V0.3	2019.09.26	修订	王连兵
V0.4	2019.12.11	修订	王连兵

目录

简单码垛.....	3
1.1 系统组成及功能说明.....	3
1.2 界面说明.....	3
1.2.1 用户坐标系的声明和标定.....	3
1.2.2 码垛基本信息设置.....	3
1.3 操作说明.....	8
1.3.1 垛盘信息设置.....	8
1.3.2 rpl 程序调用码垛功能.....	10

简单码垛

1.1 系统组成及功能说明

本码垛功能为普通码垛，码垛工件形状一致，摆放方向相同，行列分布均匀。普通码垛操作流程分为三大部分：码盘坐标系的建立、堆垛基础信息设置和码垛执行程序编写。

- 1、垛盘坐标系的建立。该步骤为码垛操作流程建立以垛盘为基准的用户坐标系。
- 2、码垛基本信息的设置。该步骤主要设置堆垛物品矩阵的行、列、高信息以及码垛中机器人的路径信息。系统会将保存的码垛信息直接生成用户可调用的机器人动作指令，方便用户调用。
- 3、码垛执行程序的编写。该部分需要用户自身需求将码垛动作指令嵌入工艺程序中。

1.2 界面说明

若用户在编辑码垛程序前，已经对码垛建立用户坐标，即码垛程序使用现有的用户坐标系，则可以跳过该部分。

1.2.1 用户坐标系的声明和标定

如若使用新的用户坐标系，需在用户程序中声明“REFSYS”用户参考坐标系变量，并使用桌面“用户坐标系”app 对新建的用户坐标系进行标定；若使用未标定的已声明的坐标系，请先行标定该坐标系。

1.2.2 码垛基本信息设置

1.2.2.1 进入码垛 APP

1. 码垛 APP 的首页

在桌面，点击“码垛”按钮，进入码垛功能页面，操作如图 2.1 所示；图 2.2 为码垛 APP 的首页：

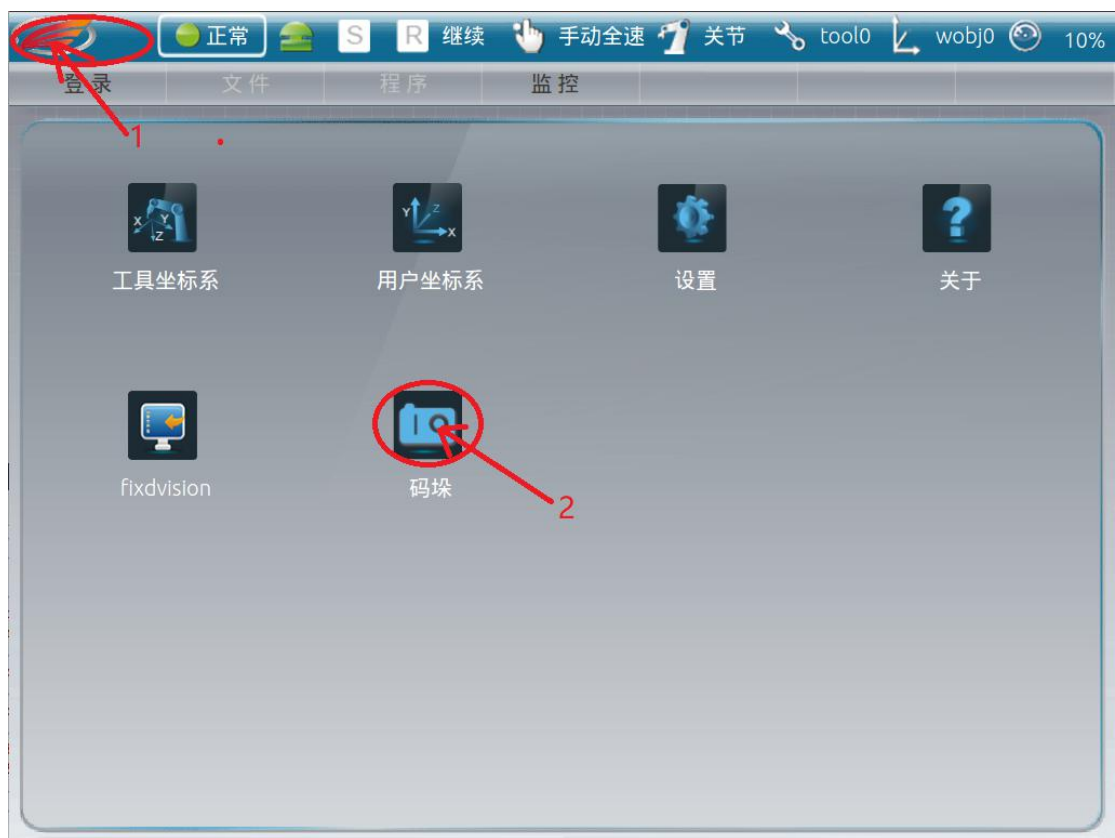


图 2.1

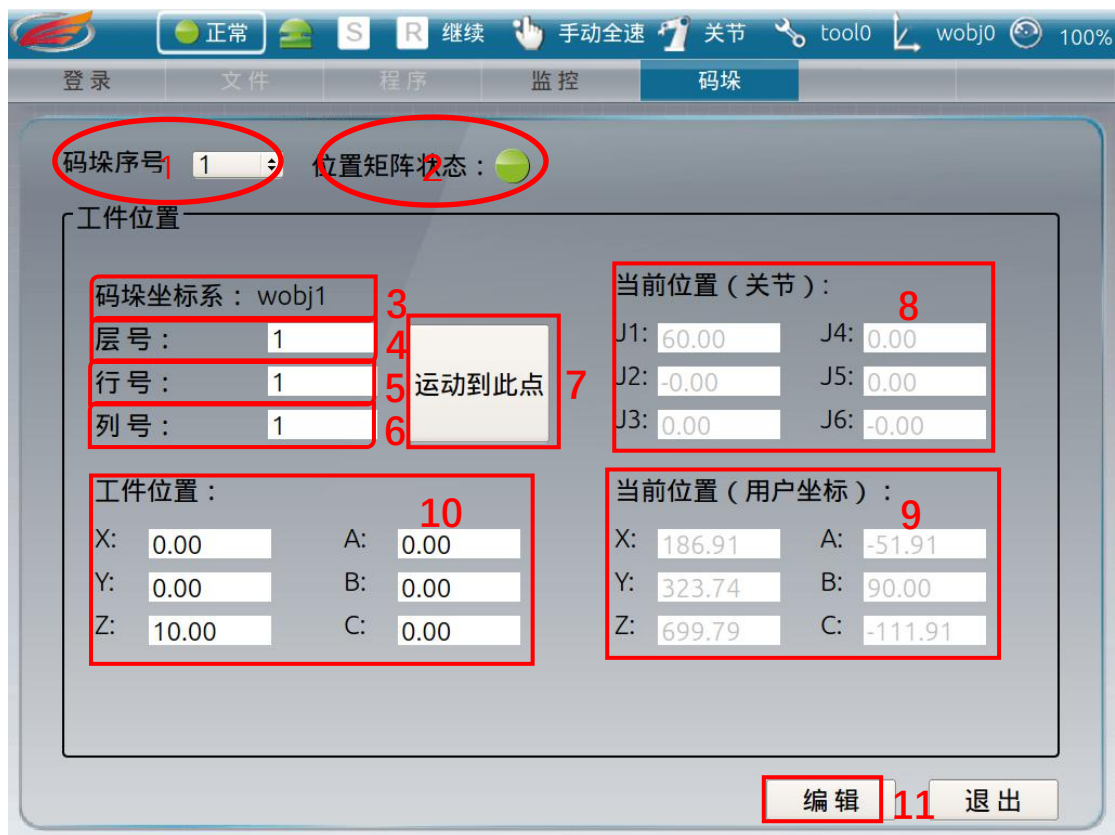


图 2.2

图中信息说明:

1.码垛序号: 堆/垛的序号, 码垛程序最大支持 8 个堆/垛。

- 2.位置矩阵状态：当位置矩阵所有信息完备，且当前无修改的情况下，会显示绿色；当位置矩阵信息不全，或对当前位置矩阵信息做了修改而未保存时，会显示红色。
- 3.码垛坐标系：显示本堆/垛的位置信息所基于的坐标系名称。
- 4.层号：此处“层号”代表的是排列方式的序号，目前在同一堆垛中只支持一种排列方式，因此此处只允许填“1”；，更多码垛方式的支持需后续扩展。
- 5.行号：当前点的行号。
- 6.列好：当前点的列好。
- 7.运动到此点：上伺服状态下，长按此按钮，机器人将向当前码垛点的位置做直线运动，当松开按钮时，机器人会立刻停止运动！如果要将机器人运动到当前码垛点位置，请长按该按钮直到机器人运动结束。当用户按压该按钮时间<1 秒时，将有提醒框，提示用户长按按钮已实现运动
- 8.当前位置（关节）：显示当前机器人的关节坐标。
- 9.当前位置（用户坐标）：显示在当前用户坐标系下机器人的直角坐标信息。
- 10.工件位置：显示当前工件的点的坐标，及放置时的姿态。
- 11.编辑：点击编辑按钮，对码垛信息进行修改，将进入码垛信息修改第一页，如图 2.3 所示。

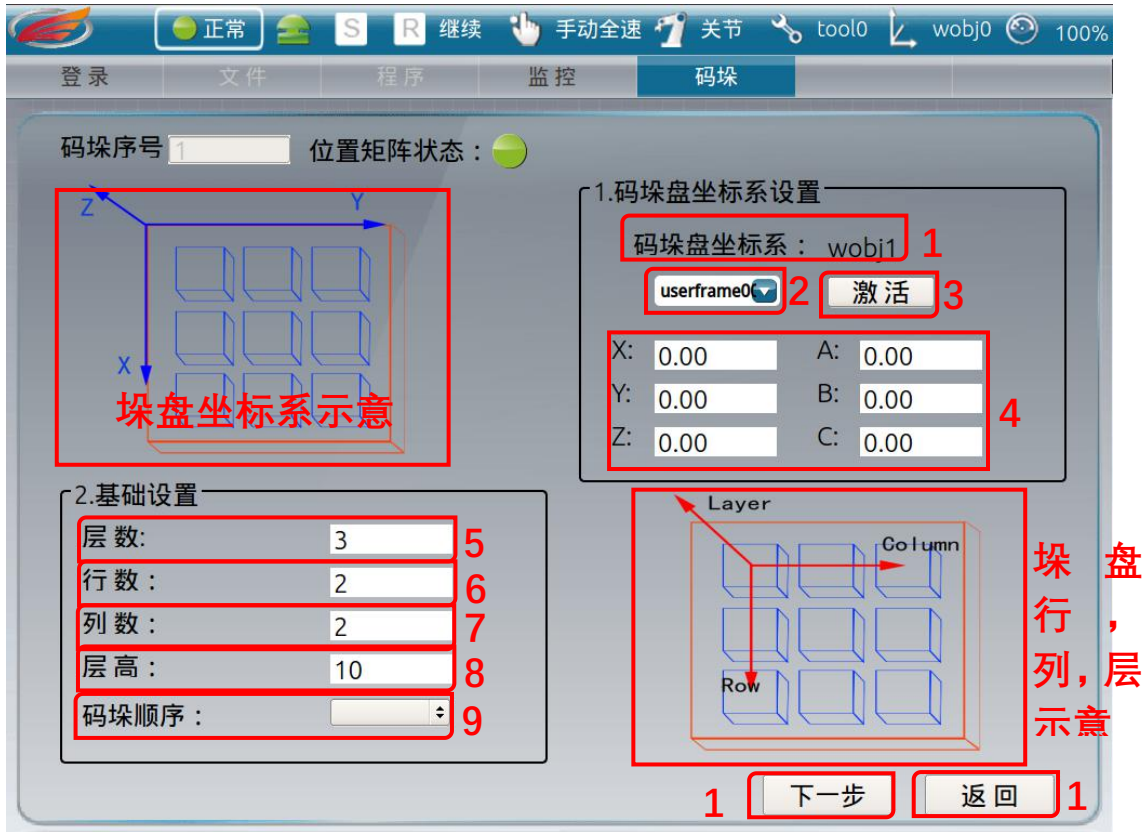


图 2.3

2. 码垛信息修改第一页
码垛盘坐标系设置

1.码垛盘坐标系：显示码垛盘使用的坐标系。说明：该坐标系的 z 轴正方向必须为垛层增加方向相同（通常为竖直向上），且为右手坐标系。

2.下拉框：显示系统中所有的用户坐标系，能够生效的用户坐标系仅为系统默认的“wobj0-wobj5”6 个系统坐标系和声明为 module 变量的用户坐标系。

着重说明：垛盘的用户坐标系必须是右手坐标系，且 z 轴正方向必须为垛层增加方向相

同（通常为竖直向上）！！！！

3.激活按钮：会将机器人的当前坐标系设置为当前选择的坐标系

4.为码垛盘坐标系的位置与姿态信息。

基础设置：

5.层数：堆/垛的层数

6.行数：堆/垛工件矩阵的行数

7.列数：堆/垛工件矩阵的列数

8.层高：堆/垛每层的高度

9.码垛顺序：目前不可用

10.返回按钮:返回上一页面，即码垛首页。

11.下一步按钮：进入码垛信息设置的下一个页面，其页面详情如图 2.4 所示。

3. 码垛信息修改第二页

图 2.4

码垛进入点设置

1.记录按钮：将当前机器人的位置信息，记录为码垛进入点的坐标信息。

2.X、Y、Z、A、B、C：显示进入点的坐标和工具姿态信息。

过渡点设置：

3.过渡点 1(approachPos1)：进入点到工件上方点间的第一停顿点。

4.过渡点 2(approachPos2)：进入点到工件上方点间的第二停顿点。

5.工具上放点点 1(overPiecePos1)：进入点到工件点间的第一个位于工具正上方的点。

6.工具上放点点 2(overPiecePos2)：进入点到工件点间的第二个位于工具正上方的点。

说明：码垛中放置每个工件时，最多允许添加两个过渡点和两个工件上方点。过渡

点为从进入点到工件上方的过渡点，可以有 X、Y、Z 三个坐标轴上的偏移量（该偏移量是以码垛盘用户坐标系为准）；工件上方点为工件上方的点，与工件的位置只允许有 Z 轴方向的偏移。

7.返回按钮：返回到码垛信息修改第一页

8.下一页按钮：进入码垛信息修改第三页，页面详细信息如图 2.5 所示

4. 码垛信息修改第三页



图 2.5

示教点设置

1.示教点：码垛程序需要示教四个点来计算码垛盘上工件矩阵的每个工件的位置及垛层的高度，四个需要示教的点分别为：

第一个点 P1：码垛起始位置，工件序号为 1 的位置

第二个点 P2：第一行最后一个，列方向结束点位置，姿态与 P1 相同。如果只有一列，P2 不用记录

第三个点 P3：最后一行第一个，行方向结束点位置，姿态与 P1 相同，如果只有一行，P3 与 P1 重合

根据前三个点，可以算出码垛盘上所有工件基于码垛盘坐标系的码垛点位置矩阵。

第四个点 P4：P1 沿着摆放高度方向一个点，姿态与 P1 相同，用来标定摆放上升高度方向。

2.码垛坐标系：显示码垛过程所采用的用户坐标系

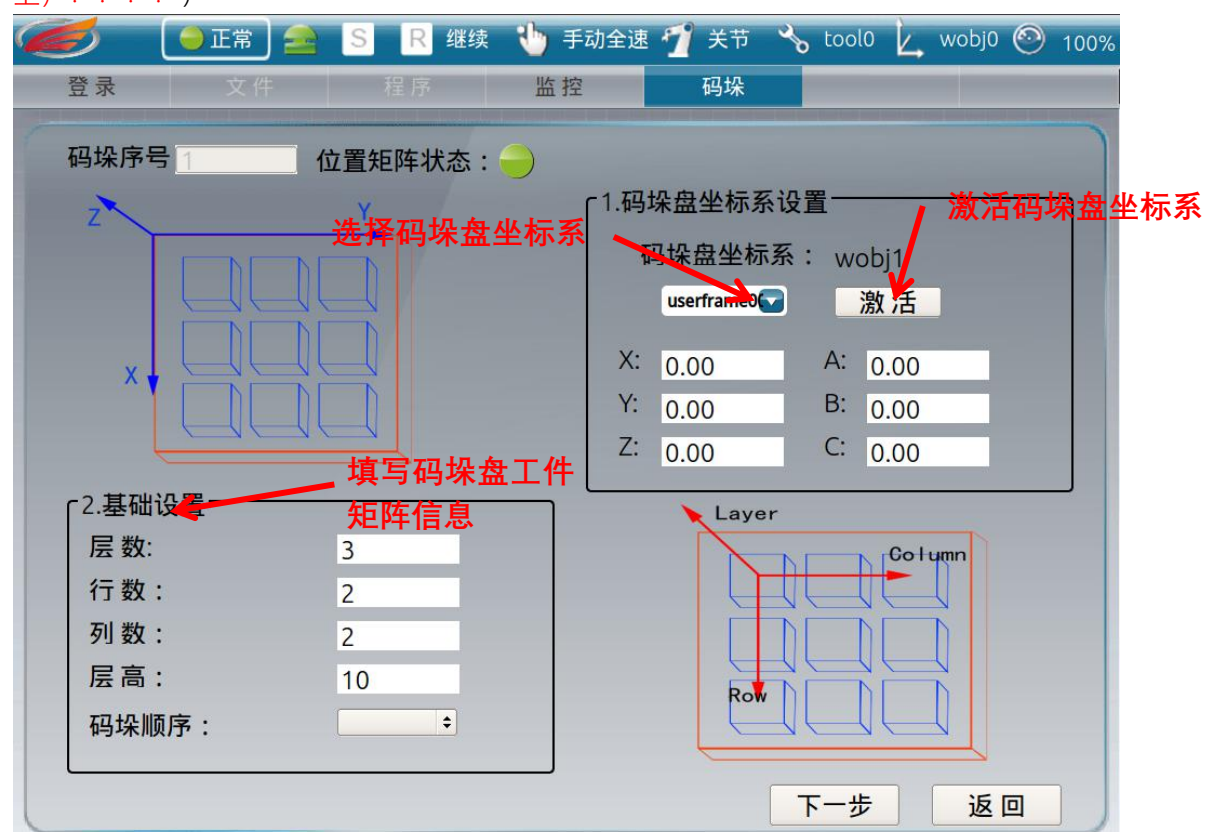
3.工具坐标系：显示码垛过程中采用的工具坐标系；

- 4.X、Y、Z、A、B、C：为示教点的具体位置和姿态信息，可以接输入
- 5.当前位置：显示机器人的当前位置
- 6.记录按钮：记录当前机器人的位置作为示教点的坐标
- 7.运动到此点按钮：上伺服状态下，**长按此按钮**，机器人将向选定的示教点位置做直线运动，当松开按钮时，机器人会立刻停止运动！如果要将机器人运动到选定的示教点位置，请长按该按钮直到机器人运动结束。**该功能只在用户“计算/保存”后才有效。**
- 8.计算/保存 按钮：将计算并保存，码垛中所有点的坐标信息，并保存进系统文件中。
- 9.返回 按钮：返回到码垛信息修改第二页

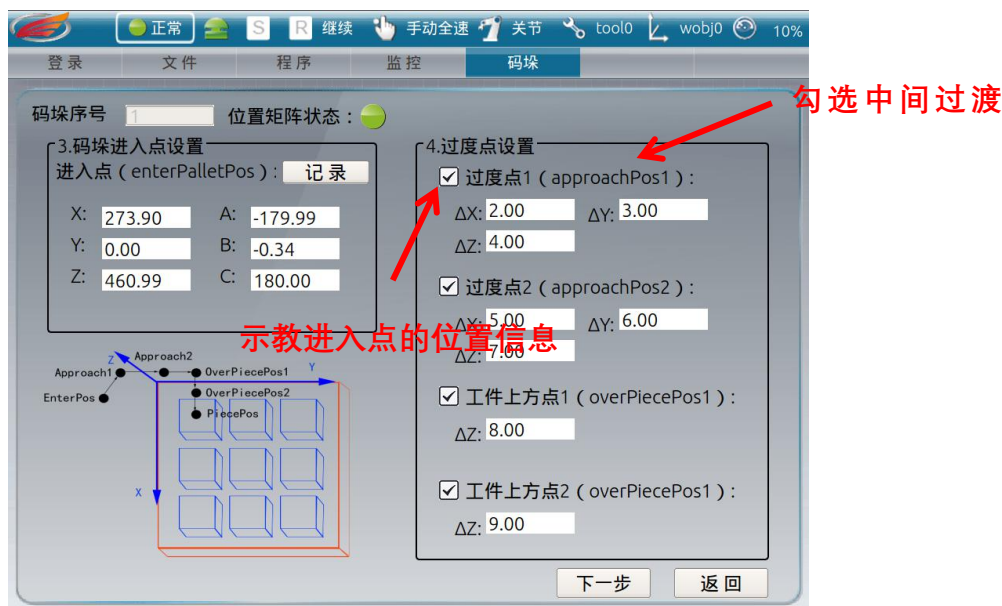
1.3 操作说明

1.3.1 垛盘信息设置

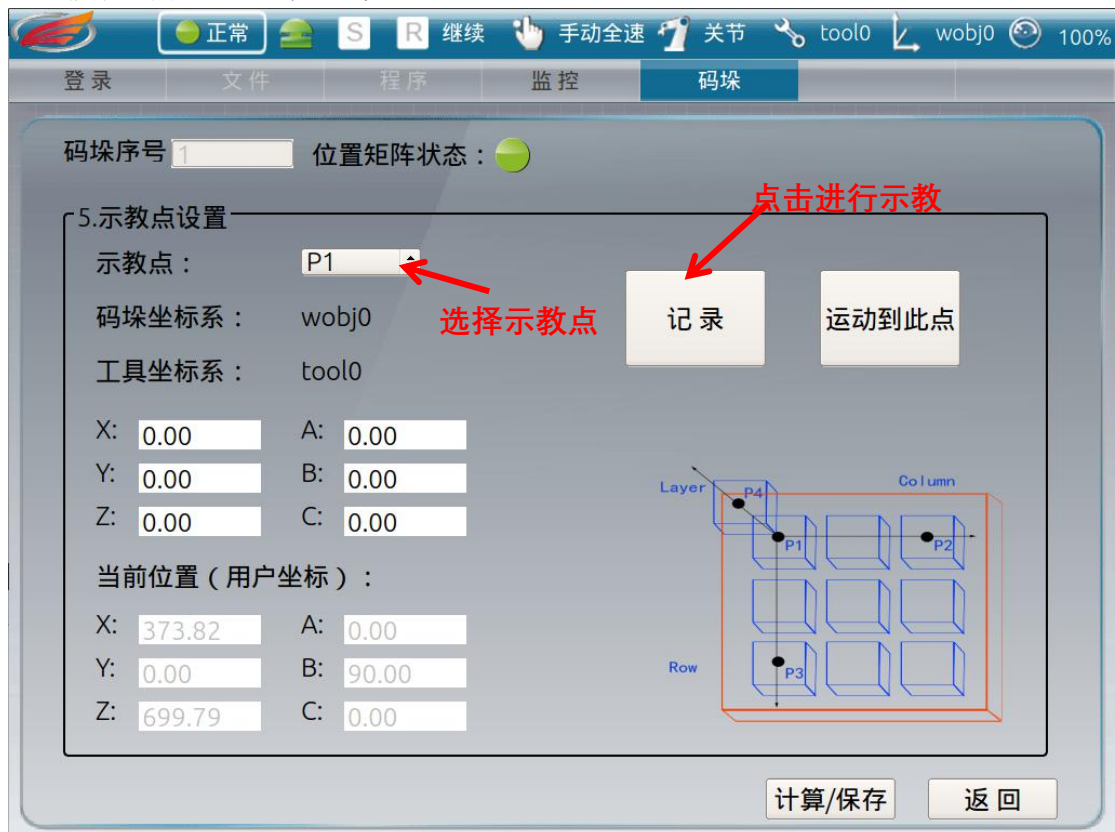
- 1.码垛首页点击“编辑”按钮，进入编辑第一页；
- 2.编辑第一页“1.码垛盘坐标系设置”中，点击下拉框，选择码垛盘坐标系，并点击“激活”。在“2.基础设置”中，输入码垛盘中工件的排列矩阵的相关信息，完毕后点击“下一步”。（**垛盘的用户坐标系必须是右手坐标系，且z轴正方向必须为垛层增加方向相同（通常为竖直向上）！！！！**）



- 3.先将机器人移动至码垛盘中第一层第一个工件的码垛进入点位置，点击“3.码垛进入点设置”中“记录”按钮；根据路径需要设置“4.过渡点设置”中的生效的点，这些点的位置是相对于工件点的偏移来确定的；设置完毕后点击“下一步”



4. 示教点 P1 为码垛第一层的第一工件放置点，位于工件矩阵第一层的第一行第一列，将机器人移动至该工件点后，点击“记录”进行示教；



5. 选取下一示教点 P2，P2 为第一行最后一个工件放置点，将机器人移动至该工件点后，点击“记录”进行示教；

6. 选取下一示教点 P3，P3 为第一列最后一个工件放置点，将机器人移动至该工件点后，点击“记录”进行示教；

7. 选取下一示教点 P4，P4 为第二层中位于 P1 正上方的工件放置点，将机器人移动至该工件点后，点击“记录”进行示教；

8. 点击“计算/保存”按钮完成码垛盘信息设置。

1.3.2 rpl 程序调用码垛功能

1. 常用码垛指令介绍

常用函数

1) pallet.loadCfg()函数，用来更新码垛参数设置，当用户使用简单码垛工艺包在线修改了堆垛信息后，该函数将修改后的信息更新到运行程序中，**建议在码垛程序开始时调用**。

2) pallet.update(DINT PalletID, DINT Type, DINT PieceID)函数，用于更新当前工件点的位置信息

参数说明：

DINT PalletID：垛盘的序号。

DINT Type：值为 0 代表码垛，值为 1 代表取垛。

DINT PieceID：工件点的序号，**从 1 开始**。

特别备注：在程序使用任何码垛工艺包相关变量前，请先行调用 pallet.update()函数，否则，码垛工艺包相关变量不会被赋值。

常用变量

pallet.enterPalletPos 当前工件点的进入点

pallet.approachPos1、pallet.approachPos2 当前工件点的中间过渡点

pallet.overPiecePos1、pallet.overPiecePos2 当前工件点的上方点

pallet.piecePos 当前工件点

pallet.maxPieces 当前垛盘中的工件点的数量

pallet.maxPallets 有效垛盘的数量

2. 程序示例

1) 变量声明



2) 主函数体

PALLET.XPLrobot

变量 代码 子程序 日志Main

```
1 pallet.loadCfg() ;
2 pieceid := 1 ;
3 pallet.update(1, 0, pieceid) ;
4 WHILE pallet.maxPieces >= pieceid DO
5     pallet.update(1, 0, pieceid) ;
6     MLIN (pallet.enterPalletPos, v500, fine, tool0, wobj1) ;
7     MLIN (pallet.approachPos1, v500, fine, tool0, wobj1) ;
8     MLIN (pallet.approachPos2, v500, fine, tool0, wobj1) ;
9     MLIN (pallet.overPiecePos1, v500, fine, tool0, wobj1) ;
10    MLIN (pallet.overPiecePos2, v50, fine, tool0, wobj1) ;
11    MLIN (pallet.piecePos, v500, fine, tool0, wobj1) ;
12    DWELL (0.5) ;
13    MLIN (pallet.overPiecePos2, v500, fine, tool0, wobj1) ;
14    MLIN (pallet.overPiecePos1, v500, fine, tool0, wobj1) ;
15    MLIN (pallet.approachPos2, v500, fine, tool0, wobj1) ;
16    MLIN (pallet.approachPos1, v500, fine, tool0, wobj1) ;
17    pieceid := 1 + pieceid ;
18 END_WHILE ;
```