

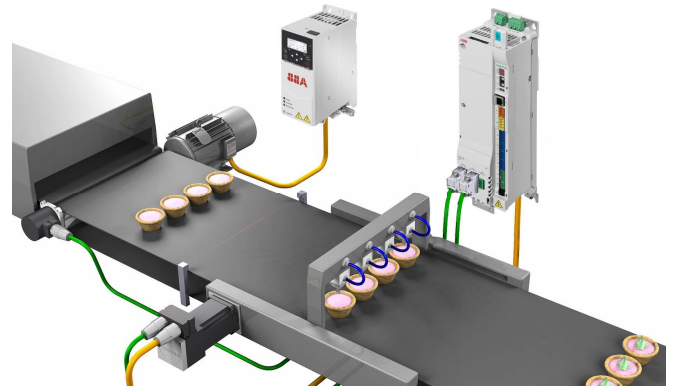
应用说明

过程与产品的同步

AN00180

Rev C (CN)

Mint 是一种易于使用的高级编程语言。它具有丰富的功能，包括编写模块化、结构化的程序。Mint 语言包括子例程、函数、任务、结构体（用户自定义的数据类型）、条件语句、循环语句等。



引言

Mint 还提供了广泛的专用功能，可与 ABB 控制器的硬件连接。它还提供丰富的关键字，大大简化了复杂运动控制的编程。Mint 是一种顺序语言（即：前面的代码执行完后，才会执行后面的代码），它还为用户提供并行处理（通过多任务）和事件/中断驱动操作的能力（例如，通过数字输入事件），使语言整体非常灵活，特别适用于机器和运动控制应用。

本文档介绍了一些技术。通过这些技术，可轻松地操作需要将过程与送进的产品同步的机器，如 <https://new.abb.com/drives/low-voltage-ac/motion> 上的动画所示。

产品同步动画

产品同步动画具有以下特性：

- 伺服控制的分配器轴（无刷交流伺服电机）
- MotiFlex e180 交流伺服驱动器
- MotiFlex e180 配备了一个可编程卡（+ N8020）。它允许使用 Mint 和通过驱动器增量编码器输入和数字 I/O 对轴进行编程
- ACS 变频器控制输送机速度
- 对准传感器连接到驱动器的快速输入通道
- 辅助编码器连接到驱动器上的辅助编码器输入通道

示例代码显示了轴如何与另一个轴上的点同步，如本示例中的蛋糕。使用快速锁定功能通过读取连接到输送机的编码器的值来捕获蛋糕的位置。当蛋糕通过定位传感器时，触发该快速锁定器。从这个测量位置开始，我们可以设置一个位置触发值。在该位置，我们的轴将开始移动并与蛋糕同步。一旦从轴与蛋糕同步，就可以给蛋糕分配糖霜。用于实现此目的的移动类型称为飞剪。

轴的配置

示例应用使用以下轴...

- (a) 产品输送机（用于输送蛋糕，变频器控制。输送机有一个辅助编码器连接到 MotiFlex e180 的编码器输入通道。）
- (b) 糖霜分配器（MotiFlex e180 控制带有糖霜分配器的轴。糖霜在 MotiFlex e180 上的数字输出打开时分配）

糖霜分配器的轴在 Mint 的 Startyup 模块中配置。

Mint 程序

程序常量

为了使代码易于阅读、便于扩展，使用了一些常量来定义某些属性。糖霜分配器轴、输送机编码器通道、输入和输出编号、糖霜持续时间和停顿距离都被声明为常数。利用这些常数代替“硬编码”，可以通过更改这些声明中的值，来方便地更新应用程序的参数。

```
Const _axDisp As Integer      = 0      ' 带糖霜分配器的轴
Const _EncChannel As Integer  = 1      ' 编码器输入通道
Const _ipProdReg As Integer   = 2      ' 产品对准输入
Const _opIcing As Integer     = 0      ' 糖霜分配器的输出
Const _IcingTime As Integer   = 500   ' 分配持续时间应该以 ms 为单位
Const _DwellDistance As Float = 200   ' 从对准传感器到轴必须开始移动时的位置的距离
```

变量声明

飞剪中使用的各种主距离被声明为变量，按蛋糕间距的百分比来计算（分数）。

```
Dim fMinDist As Float = 1000          ' mm 蛋糕之间的最小间距
' 主距离
Dim fMSD1 As Float   = fMinDist * 0.15 '150mm
Dim fMSD2 As Float   = fMinDist * 0.35 '350mm
Dim fMSD3 As Float   = fMinDist * 0.15 '150mm
```

比例因子

比例因子允许将每个轴缩换算为工程单位，以便于使用。比例因子是应用于轴的所有运动变量（速度、加速度、移动距离等）的除法因子，使用 Mint SCALEFACTOR 关键字进行设置。

由反馈设备生成的用于移动一个用户单位（uu）的脉冲计数定义为 SCALEFACTOR。默认情况下，SCALEFACTOR 为 1，表示所有轴运动参数均为编码器正交计数。

糖霜分配器轴的机械结构决定了每 200 个脉冲计数对应 1mm 距离。因此，将 SCALEFACTOR 设置为 200 允许以线性 mm 为单位进行移动。

可以使用 ENCODERSCALE 关键字换算输送机编码器的距离。输送机的机械结构决定每 400 个编码器计数对应 1mm，因此，将 ENCODERSCALE（_EncChannel）设置为 400 将允许以 mm 为单位对与输送机（主轴）行走的距离编程。

主从配置

为了能够使用飞剪，我们必须建立主（输送机）和从（糖霜分配器）轴之间的关系。可以在 Mint 程序的 Startup 模块中添加如下代码：

```
MASTERSOURCE(_axDisp)      = _msENCODER
MASTERCHANNEL(_axDisp)     = _EncChannel
```

此代码设置从轴将与之同步的主轴。在本应用中，糖霜分配器轴将跟随通道 1 上的编码器。这是测量输送机位置的编码器，连接到驱动器的编码器输入通道。

触发配置

通过使用触发，我们不使用 GO 命令触发运动。以下代码包含在 mint 程序的 Startup 模块中：

```
TRIGGERMODE(_axDisp)       = _trFWD_MOTION
TRIGGERSOURCE(_axDisp)     = _tsENCODER
TRIGGERCHANNEL(_axDisp)    = _EncChannel
TRIGGERVALUE(_axDisp)      = -9999      ' 首先禁用触发
```

当编码器通道 2 的值在正向上通过-9999 个用户单位时，此代码设置轴 0 以触发运动。

在 Mint 启动模块中，我们还设置了：

```
ENCODERWRAP(_EncChannel) = 1000
```

这是编码器在返回（回退）到零之前将达到的最大值。这可以确保此编码器的操作值只能在 0 到 999 的范围内。因此，设置-9999 的 TRIGGERVALUE（编码器永远无法达到的值）会禁止从编码器触发运动。

快速锁存配置

要在 e180 产品上设置快速锁存，可以使用一系列 LATCH关键字。在使用 MotiFlex e180 时，锁存通道 0-3 用于锁存与驱动器相关的硬件（例如基本编码器或轴位置）。

在伺服驱动器上锁存编码器的值时，用于触发锁存的输入或输出也必须位于该伺服驱动器上。DI1&DI2 是 e180 上的快速输入，DO0&DO1 是快速输出。

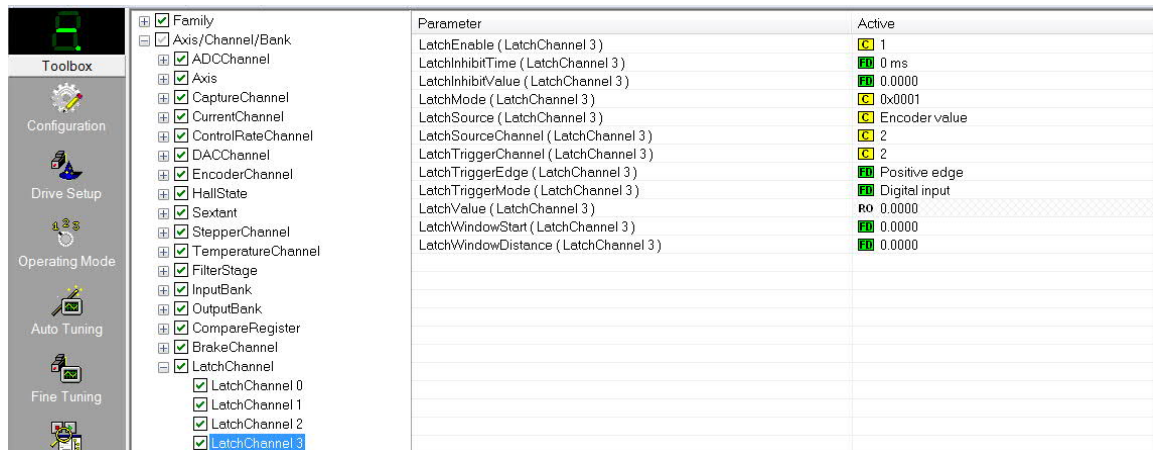
在 Startup 模块中添加以下代码来配置锁存通道 3，以在数字输入 2 的上升沿捕获输送机轴的位置。

```
LATCHTRIGGERCHANNEL(3)    = _ipProdReg
LATCHTRIGGERMODE(3)       = _ltmINPUT
LATCHTRIGGEREDGE(3)      = _ltePOSITIVE_EDGE
LATCHSOURCE(3)            = _lsENCODER
LATCHSOURCECHANNEL(3)    = _EncChannel
LATCHMODE(3)              = _lmAUTO_ENABLE
LATCHENABLE(3)            = _true
```

锁存被设置为自动重新使能，并且在最开始时处于使能状态以备使用。

在 Mint Workbench 的参数查看页中可查看锁存通道的设置。最初仅显示锁存通道 0-3，因为这些都是与驱动器相关联的通道。如果您设置了额外的锁存通道，并且在参数查看器中看不到它，则需要断开连接，并从 workbench 重新连接驱动器以更新参数查看器。为此，请单击位于 Mint Workbench 左上角的“Connect/Disconnect”图标，按一次将断开连接，再次按下它将重新连接。

设置完成后，参数查看器将显示锁存通道的以下设置，请参见下图。



主程序

在主程序中，LOOP 触发被禁用，飞剪定位被加载到定位缓冲区中，然后程序等待定位被触发。这可以通过以下代码实现。

```

TRIGGERVALUE(_axDisp) = -9999          ' 禁用触发
doLoadFlys                          ' 将飞剪加载到定位缓冲区
' 等到触发定位或轴闲置
Pause ((MOVESTATUS(_axDisp) & 256) = 256) Or (IDLE(_axDisp))

```

我们首先将 TRIGGERVALUE 设置为 -9999，以禁用触发（有关原因，请参阅上一节关于触发器配置的部分）。然后，调用子例程 doLoadFlys，将飞剪定位载入到定位缓冲区中，同时通过输出打开分配糖霜开关。稍后我们将进一步研究这个子例程。然后，程序等待触发定位。当一批蛋糕通过对准传感器时，将发生一个设置 TRIGGERVALUE 的事件。在设置该事件后，当输送机到达传感器的某个位置时，该 TRIGGERVALUE 将使糖霜分配器轴开始飞剪。然后对下一行蛋糕重复整个过程。为了提高程序的速度（以及机器的整体速度），我们可以在检测到当前一组定位被触发后开始加载下一组飞剪数据（MOVESTATUS 为 256）。

当检测到糖霜分配器的轴处于 IDLE 状态时，也会开始加载下一组飞剪。要使 IDLE 为真，定位缓冲区中不能加载定位，轴必须是静止的，并且状态为“in position”。

子例程 - 加载飞剪段

doLoadFlys 子例程将飞剪定位加载到定位缓冲区中，准备好在需要时触发。

```

MASTERDISTANCE(_axDisp) = fMSD1          ' 沿斜坡加速到同步速度
FLY(_axDisp)           = fMSD1/2

MOVEPULSEOUTX(_axDisp, _opIcing) = _IcingTime ' 分配糖霜

MASTERDISTANCE(_axDisp) = fMSD2          ' 与产品同步
FLY(_axDisp)           = fMSD2

MASTERDISTANCE(_axDisp) = fMSD1          ' 沿斜坡减速到停止
FLY(_axDisp)           = fMSD1/2

MASTERDISTANCE(_axDisp) = fMSD3          ' 返回起始位置
FLY(_axDisp)           = -(fMSD1+fMSD2) /2

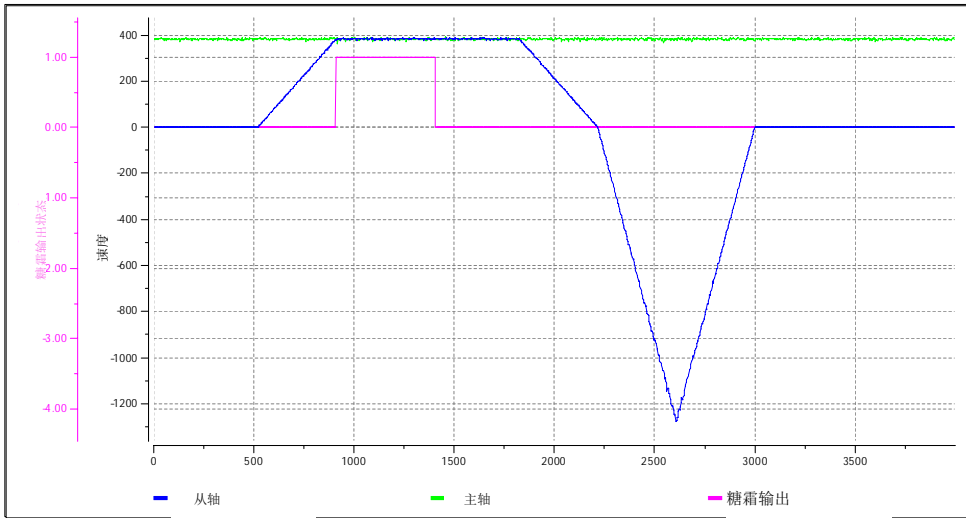
MASTERDISTANCE(_axDisp) = fMSD3          ' 沿斜坡减速，回到起始位置
FLY(_axDisp)           = -(fMSD1+fMSD2) /2

```

每个定位段都由 MASTERDISTANCE 和 FLY 命令组成。MASTERDISTANCE 的值是主轴（输送机）上的距离。从轴（糖霜分配器）将在该距离内行进一段。FLY 的值是对应主轴（输送机）行进 MASTERDISTANCE 的距离上，从轴（糖霜分配器）行进的距离。

通过使用 FLY 命令，我们可以实现两个轴的同步运动，本示例中是指输送机和糖霜分配器轴。一旦轴同步，就可以把糖霜分配到蛋糕上。然后，糖霜分配器的轴减速到停止，然后返回到起始位置以等待下一批蛋糕。

下图来自 workbench，它显示了在缓冲区中配置的定位的主轴和从轴的速度。此外，图中还显示了糖霜输出的状态，即在轴同步时打开输出的状态。



MOVEPULSEOUTX 关键字使我们能够在前一个定位段完成后打开输出一段时间。打开输出不会影响轴的运动，并且将在不停止轴的情况下施加下一个飞剪段。在这个程序中，DO0 的输出时间为 500ms。

图中显示的负数区域表示轴已改变方向并返回到起始位置，以等待下一批蛋糕。曲线与 x 轴包围的面积在 x 轴的上方和下方应该相同，表示向前行进的距离与向后返回的距离相同。

完整的循环在代码中定义的所有 MASTERDISTANCE 的总和上发生。这相当于 $0.95 * fMinDist$ ，从而确保循环始终能在下一组蛋糕到达循环开始位置之前完成。

锁存事件

在调用锁存事件时，执行以下代码行。

```
TRIGGERVALUE(_axDisp) = ((LATCHVALUE(3) + _DwellDistance) % ENCODERWRAP(_EncChannel))
```

当下一批蛋糕通过对准传感器时，调用锁存事件。此时，我们从主编码器（连接到输送机）获取锁存值并添加已知的偏移距离（_DwellDistance），以确定糖霜分配器轴何时启动。我们使用常量值来定义该偏移距离。但是，该值通常是可调节的（例如，通过 HMI），以允许操作者调整系统的定时并由此调整糖霜的位置。该组合值被分配给 TRIGGERVALUE。当主编码器处于设定值时，该值将触发在定位缓冲区中加载的定位。为确保分配给 TRIGGERVALUE 的值不大于主编码器返回的最大值（最大值由 ENCODERWRAP 设置），我们使用 ENCODERWRAP 获取值的模数（由 % 符号表示）。这将返回 TRIGGERVALUE 除以 ENCODERWRAP 的余数。

联系我们

要了解更多信息，请联系您的当地的 ABB 代表，或以以下方式：

new.abb.com/drives/low-voltage-ac/motion

new.abb.com/drives

new.abb.com/channel-partners

new.abb.com/plc

© ABB 公司，2019 年，版权所有。保留所有权利。
技术规格如有变更，恕不另行通知。