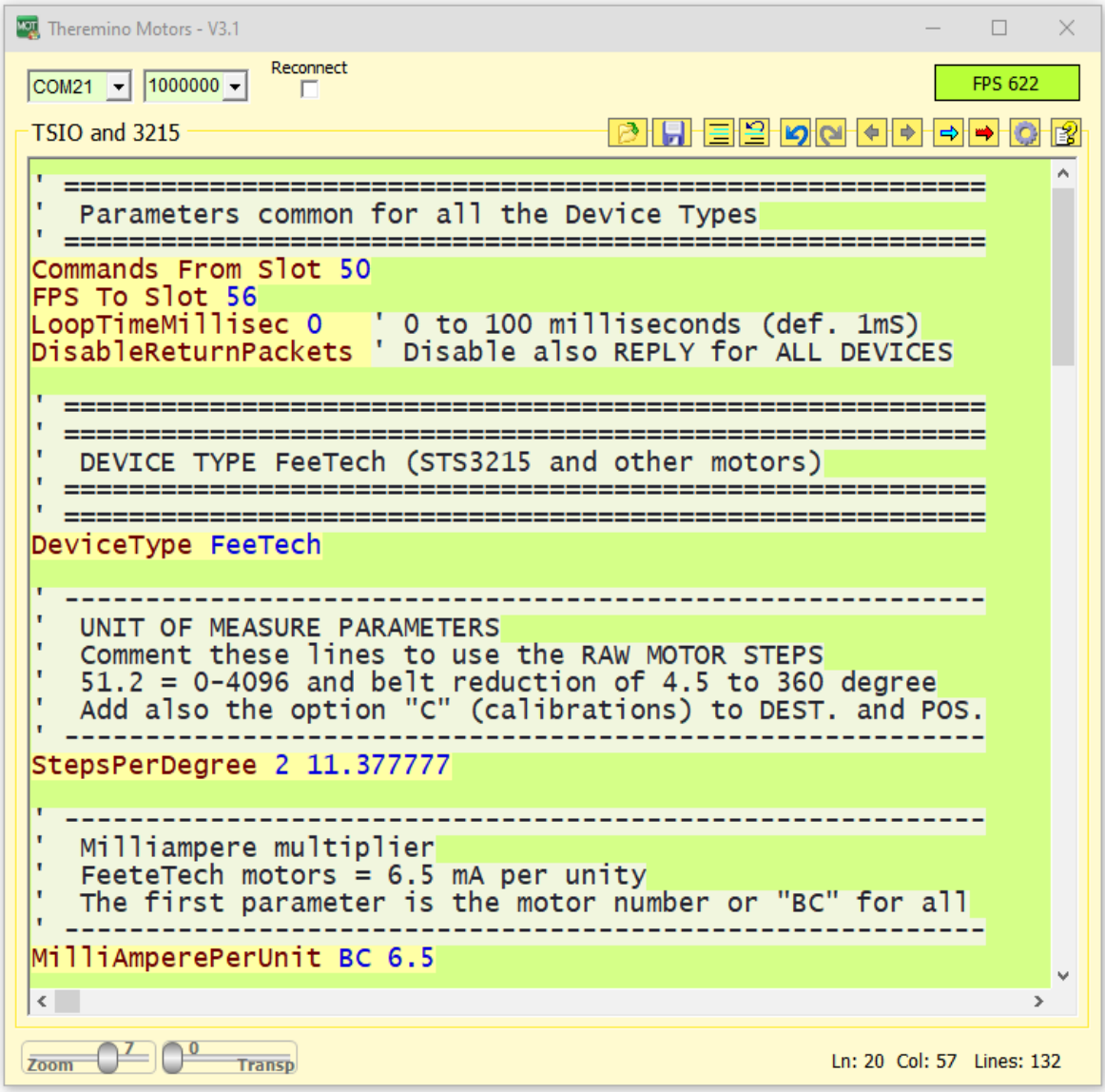


Theremino Motors V3.1

Theremino_Motors 应用程序

此应用程序将 theremino 系统的插槽与连接到 ***Half-Duplex*** 类型的串行线路的设备进行通信（RS485 与两条平衡线，或 TTL 与单线）。

这些设备通过四线电缆（两个信号加上电源和 GND）或在 TTL 版本中甚至仅三线连接成链。

在此图像中，您可以看到一些 ***FeeTech*** 和 ***Dynamixel*** 电机以及打开的 TSIO 模块以显示连接。

智能电机包含所有控制电子设备、一个 4096 步编码器和一个可配置的 PID 算法，可让您以 0.09 度的精度控制旋转（以及带双编码器的 TMOT 高达 0.003 度）。

还可以调整速度、加速度和扭矩，以及读取到达的位置、温度、电流（与扭矩有关）和许多其他参数。

不同类型的设备可以连接在同一条通信线上，例如底座、肩部和肘部三个 TMOT 能够在一米处举起 1 千克，然后顶部有三个 FeeTech 的 3215，体积小，重量轻，移动钳子和螺丝刀，最后是尖端的 TSIO 来控制工具。

(2022 年) TSIO (Theremino 智能输入输出) 模块已经可用，它提供超过二十个 ADC、PWM、DigIN 和 DigOUT 输入输出引脚。 TSIO 可以轻松连接到与 Feetech、Dynaminxel 和 TMOT 电机相同的数据和电源线上

(2023 年) TMOT (Theremino Motors) 正在准备中，真正的 SmartMotors 用于工业控制，但价格极具竞争力。 TMOT 比 FeeTech 和 Dynamixel 具有更好的扭矩控制。 此外，无需打开机器人并将它们与通信链断开连接，就可以识别和重新编程 TMOT。

伺服电机的特性

在此页面上，我们报告了一些特别有趣的伺服电机的特性。

我们尝试了此列表中的前五名，并绝对推荐前三名（3032、3215 和 3046），除了非常经济外，它还可以平稳，安静地运行。

Motor	Voltage	Gears	Speed (RPM)	Torque (kg-cm)	Encoder bits	Size (mm)	Weight (g)	Euro	Processor
STS3215	4V-7.4V	345:1	54	19	12	45*25*35	55	10	GD32F130F8P6TR
STS3032	4V-7.4V	205:1	111	4	12	32*12*27	21	20	GD32F130F8P6TR
STS3046	6-7.4V	378:1	52	40	12	40*20*43	89	25	
SM29BL	12-24V	241:1	110	24	12	40*28*42	102	50	STM32F030K6T6
SM45BL	12-24V	353:1	110	25	12	46*28*34	100	90	STM32F030K6T6
SM120BL	9-25V	232:1	50	120	12	*78*43*65*	485	420	
eRob70	48V		40	367	19	70*81	1000	550	

该表中的数据为近似值，应将其作为粗略比较电动机的粗略指示。 毫米四舍五入到最接近的整数，价格是欧洲的价格。 中国的价格低 30..40％

这里考虑的所有电动机都是无芯或无刷的，编码器是磁性的或光学的。 我们以前已经放弃了所有带有电刷电动机和电位计反馈的型号。

指示的“转矩”值是失速值。 在正常工作条件下，必须将其保持在较低水平。

通讯协议

该应用程序使用的协议是 ***Dynamixel 1.0***（缩写为 DXP1），使您可以与所有伺服电机通信 ***Dynamixel*** 是 ***费科技***。

还有一个协议 ***Dynamixel 2.0*** 但是我们没有实现它，因为它仅适用于某些 Dynamixel 模型（MX 和 PRO），并且不适用于 FeeTechs。此外，协议 2.0 不包含任何特定的优点。唯一的主要改进就是 SyncRead 指令，该指令已被我们的***缓冲传输***

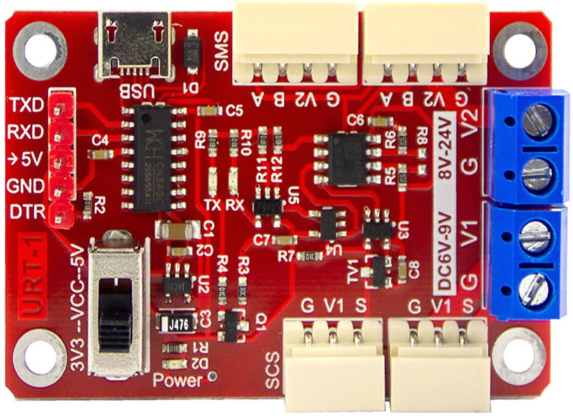
使用 DXP1 或 Modbus 协议

可以将 FeeTech 伺服器编程为使用 DXP1 协议或 ModBus 协议。后者是 PLC 用户所熟悉的，但是具有较低的速度性能和较少的命令。

为了使用 Modbus 设备，我们编写了一个与此应用程序非常相似的应用程序，但是将其称为 ***Theremino_Modbus*** 在其 ZIP 中，您还将找到相关的文档文件。

用于 DXP1 或 Modbus 的 Program FeeTech 伺服电机

我们已经准备好了 ***这一页*** 一个 ZIP 存档，其中包含使用两种协议对 FeeTech 伺服器进行重新编程所需的所有内容。您还可以在 ZIP 文件中找到编程说明。



要对 FeeTech 电动机进行重新编程，您将需要一个 USB 连接模块（如右图所示），一条 USB 电缆和一条用于电动机的连接电缆。

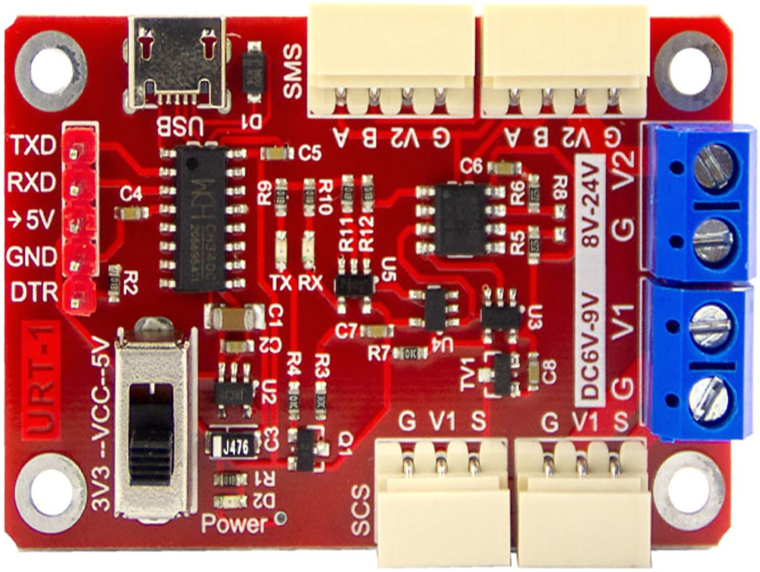
您还需要 24 伏电源 它的负极连接到端子“ G”，正极连接到 V2。

电源必须在 PC 键盘和鼠标附近有一个方便的电源开关。使用开关，您必须在完全按下软件上的编程按钮的同时通电。如果未同步执行此操作，则编程将不会开始。

要使 Dynamixels 工作，您可以使用相同的适配器，但不能 我们仍然尝试。

Feetech URT1 开发板

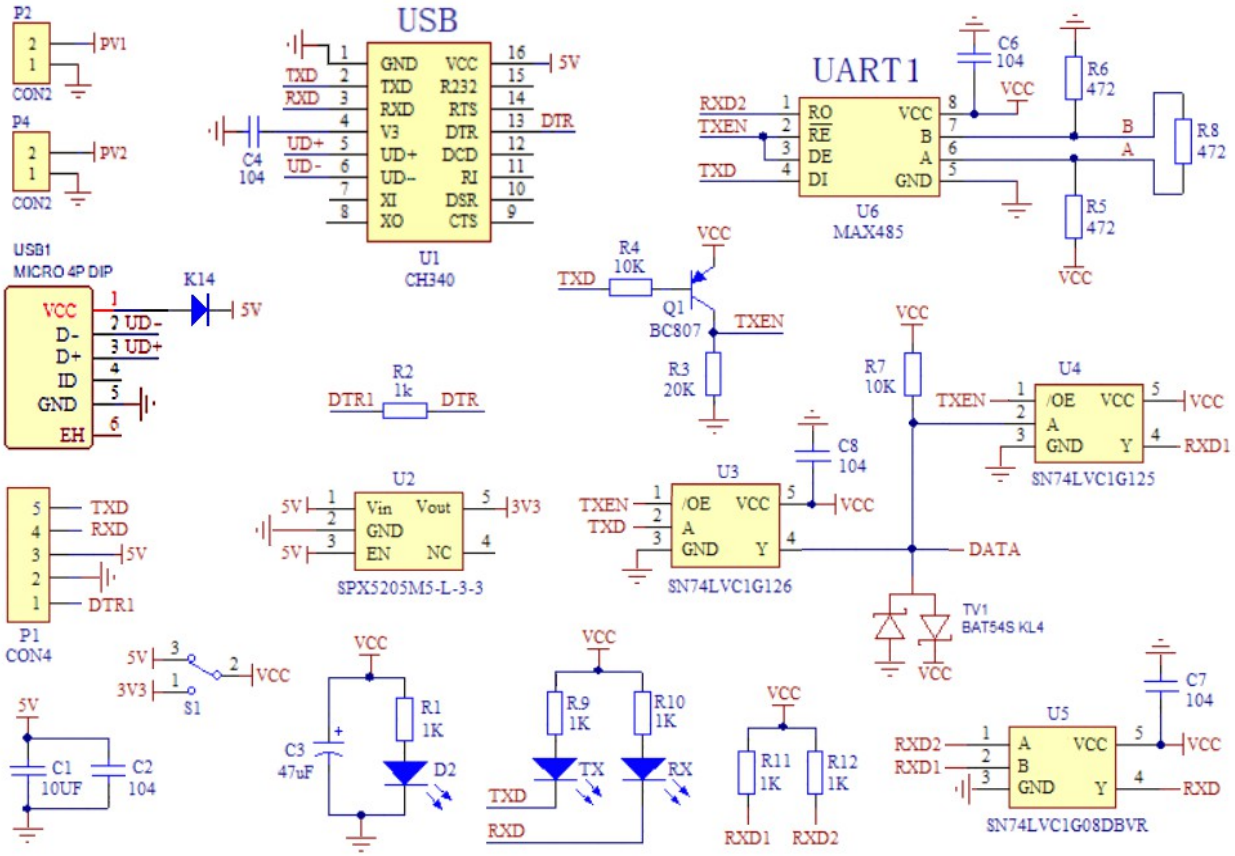
使用此卡，您可以将电动机连接到 USB 端口，并使用右侧的蓝色连接器为其供电。



四线电机连接到顶部连接器，并在连接器 G 和 V2 上馈入 8 至 24 伏电压三芯电机插入下面的连接器，并在连接器 G 和 V1 上提供 5 至 7 伏特的电压

可以通过将来自第二个 USB 端口（最好是 USB3）的 5 伏电压连接到 G 和 V1，或使用 5 伏或 6 伏电源和至少 2 伏或 5 伏电源，为低压电动机（我们推荐出色的 3032 和 3215）供电。 3 安培

接线图



编辑命令列表

在应用程序的中央部分有命令列表。黄色背景上的彩色部分处于活动状态，没有错误。浅绿色的部件带有注释，不起作用。

```
' ==========================================================
' ==========================================================
'  DEVICE TYPE TSIO  (Theremino Smart InOut module)
' ==========================================================
' ==========================================================
DeviceType TSIO1

0 To DXP1 BC 152-1   ' ReplyMode (0=NoReturnPackets)

' ----------------------------------------------------------
'  Exchange list (executed repeatedly)
' ----------------------------------------------------------

' ----------------------------------------------------------
'  TSIO INPUTS (TMOT BASE DATA)
' ----------------------------------------------------------
DXP1 1 100-34 To Buffer       ' Bytes 100 to 133 >>> Buffer
Buffer 100-4 To Slot 100      ' Serial number
Buffer 104-1 To Slot 104      ' Motor  version
Buffer 105-1 To Slot 105      ' Firmware   Version
Buffer 106-1 To Slot 106      ' Firmware   Subversion
Buffer 107-1 To Slot 107      ' Bootloader Version

' ----------------------------------------------------------
'  TSIO INPUTS (PINS)
' ----------------------------------------------------------
Buffer 110-2 To Slot 110 F03' ADC1   PA07
Buffer 112-2 To Slot 112 F03' ADC2   PA06
Buffer 114-2 To Slot 114 F03' ADC3   PA04
Buffer 116-2 To Slot 116 F03' ADC4   PA03
Buffer 118-2 To Slot 118 F03' ADC5   PA02
Buffer 120-2 To Slot 120 F03' ADC6   PA09
Buffer 122-2 To Slot 122 F03' ADC7   PA10
Buffer 124-2 To Slot 124 F03' ADC8   PA11
Buffer 126-2 To Slot 126      ' DigIn1 PA25
```

此列表中写入的所有内容均立即可用。每次甚至更改一个字符时，都会再次检查整个列表，并立即激活没有错误的零件。

为此 编辑时要小心 命令不要意外地创建可写入所需寄存器以外的寄存器的命令。
或禁用通讯 如下页所述。

```
' ------------------
'  Write with option
' ------------------
Slot 11 To DXP1 0 42-2  E    ' Write Destination
```

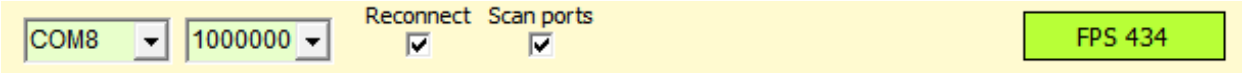
如果发生错误，该命令会变为红色，在此处您会看到包含错误的行，该行不存在“ E”选项。

禁用通讯

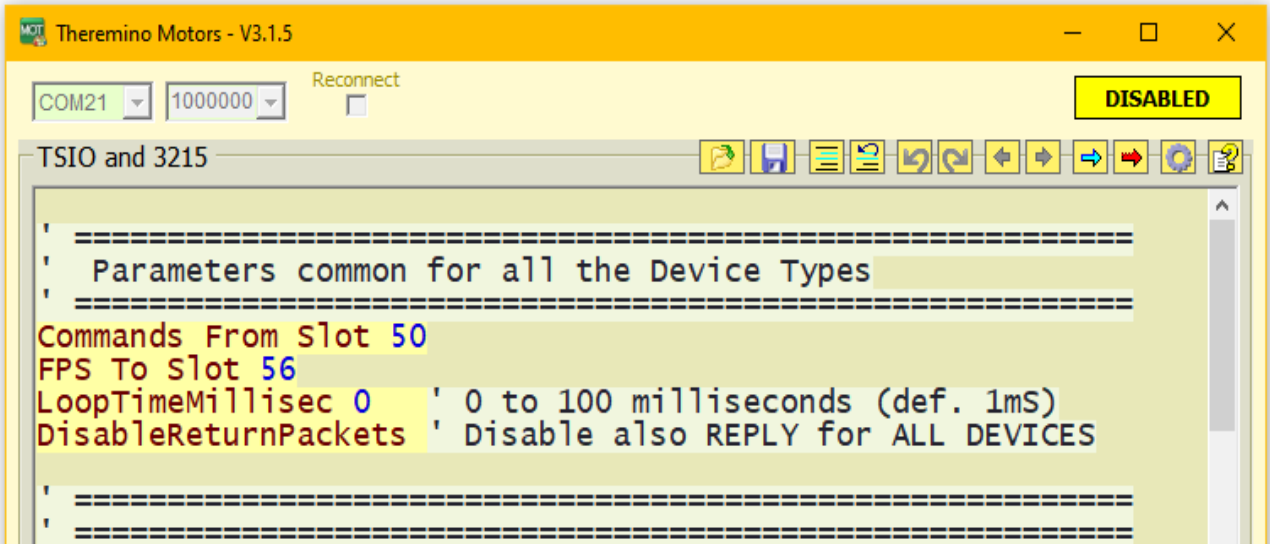
您在命令列表中编写的所有内容均可立即运行。每次即使更改一个字符，都将再次检查整个列表，并立即激活没有错误的零件。

这对于快速测试变化的影响非常方便，例如在调整 PID 值时。但是，如果您不小心，可能会无意中写入其他寄存器。例如，如果您写入标识符寄存器，则引擎将不再工作，并且您将不得不断开它们并一次重置一个标识符。

因此，在进行实质性更改时，建议禁用通信。您可以断开电动机的电源，但是使用鼠标以 FPS 指示通信速度可以更方便地按下右上角的按钮。



当按下按钮时，消息“ DISABLED”在黄色背景上出现，并且通讯端口命令被禁用。



命令列表也会更改颜色以突出显示禁用。

完成更改后，再次按按钮，该按钮将以绿色背景标记速度。

命令列表DXP1 协议

初始化命令　在启动时和编辑程序文本时运行一次

LoopTimeMillisec nnn	' 延迟以减慢执行循环
SectionSelectorSlot s	' 设置分区槽
nnn To DXP1 d r-b	' “立即”编号到寄存器
StepsPerUnit d n.nnn	' 请参阅下一页的特殊命令

设备类型初始化

它必须位于每个部分的开头，并且位于与特定设备相关的所有其他命令之前

DeviceType Feetech	' 设备类型

写命令　尽可能连续地进行

Slot s To DXP1 d r-b	' 读取插槽并写入寄存器
Slot s To DXP1 d r-b　A	' 总是写（A =总是）
Slot s To DXP1 BC r-b	' 写入所有设备（BC =广播）

读取命令　尽可能连续地进行

DXP1 d r-b To Slot s	' 读取 1 到 4 个字节并将其发送到插槽
DXP1 d r-b To Buffer	' 从 1 到 253 字节读取并将其发送到缓冲区
Buffer r-b To Slot s	' 从缓冲区到插槽的 1 到 4 个字节
Buffer r-b To Slot s	' 从缓冲区到另一个插槽的 1 到 4 个字节

段设置命令，也连续执行

Section nnn	' 部分的开始标记（可能还有小数点后）

缩写的含义

nnn　=数字
s　=插槽（1 到 999）
d　=设备标识符（0 到 199）（BC =“广播”）
[R　=设备寄存器（0 至 255）
b　=组成寄存器的字节数（从 1 到 4）

特殊命令

这些命令用于以固定值初始化设备，它们主要用于调整串行端口“波特率”的速度，最小和最大移动限制以及 PID 参数。

这些命令仅在启动应用程序时发送一次，并在每次更改程序中的任何字符时再次发送。

这些转移 它们不影响每秒的交换次数（FPS），因为它们仅作用一次且时间很短。 因此，您不必担心会像其他所有传输那样小心地使用它们，并将它们的使用限制在必需品上。

特殊命令示例

```
DeviceType Feetech          ' 设备类型
LoopTimeMillisec nnn        ' 延迟以减慢执行循环
SectionSelector_Slot s      ' 设置分区槽
StepsPerUnit d n.nnn        ' 测量单位设置
StepsPerDegree d n.nnn      ' 测量单位设置
StepsPerMM d n.nnn          ' 测量单位设置
Commands From Slot s        ' 外部指令接收槽
FPS To Slot s               ' 用于测量 FPS 值的插槽
DisableReturnPackets        ' 禁用等待回复的说明
```

如果您不编写或注释设置插槽的行，则不会执行相关功能。

在仅执行一次的特殊命令中，还有“立即”数字的传送，将在下一页中更详细地说明。

“立即”转移的示例

```
nnn To DXP1 d r-b           ' “立即”编号到寄存器
```

缩写的含义

nnn =数字
s =插槽（1 到 999）
d =设备标识符（0 到 199）（BC =“广播”）
[R =设备寄存器（0 至 255）
b =组成寄存器的字节数（从 1 到 4）

测量单位和C”选项

如果不使用此页面上的命令，则与电机交换的位置数据是原始值，由位于电机中的角度编码器测量，即非常大的整数。

通过以下命令，可以使用更方便和更重要的单位，例如度或毫米，具体取决于使用的是角运动还是线性运动。

如果需要，您还可以使用自己喜欢的测量单位，例如公里、英寸、码、英尺等。在这些情况下，您将使用通用的“StepsPerUnit”命令并在注释中注明您正在使用的单位.

以下三个命令具有不同的名称以突出显示您使用的度量单位，但它们执行完全相同的计算。 实际测量单位仅取决于数字字段“n.nnn”中设置的数字。

```
StepsPerUnit      d n.nnn
StepsPerDegree  d n.nnn
StepsPerMM        d n.nnn
```

第一个值“d”定义了电机，可以从 1 到 199，这样每个电机可以有不同的减速比。第二个值“n.nnn”可以是整数或小数。 为了获得所需的测量单位，必须根据电机的分辨率和机械减速来计算该数字。

“C”选项

要使用测量单位，所有读取或写入位置值的命令（通常称为“目标位置”和“实际位置”）都必须标有字母“C”（单位）。

```
nnn    To DXP1 d r-b    C      ' 引擎寄存器的“立即”编号
Slot s To DXP1 d r-b    C      ' 从插槽到引擎寄存器
DXP1  d r-b To Slot s   C      ' 从引擎寄存器到插槽
Buffer r-b    To Slot s C      ' 从缓冲区到插槽
```

如果没有“C”选项，则“StepsPerXXX”命令无效。

“C”选项在下一页解释。

“C”和 “M”选项

选项“C”（校准）

要使用零位校准（和测量单位），您必须用字母“C”（校准）标记所有读取或写入位置值的命令（通常称为“目标”和“实际位置”）。

“M”（镜像）选项

“M”选项（mirror，在意大利语中意为“镜像”）用于反转电机的旋转方向。当其中一台电机安装在另一台电机的另一侧时，或者当使用不同电机型号的电机以与所需电机相反的方向旋转时，将使用此选项。

仅在位置命令上使用“C”和“M”选项。
速度、温度、加速度、启用、调整等命令。
他们不能有“C”，否则你会得到不可预测的行为。

“X”选项（乘法）

“X”选项在将值发送到设备或将其写入插槽之前将其相乘。 您还可以使用带小数点的数字进行除法，如 X0.123 或者使用带有指数的数字，如 X2.7e2（意味着乘以 270）。

从“F00”到“F99”的选项（特殊功能）

F00 到 F06 和 F12 的功能是当前定义的，所有其他直到 F99 的功能都可以自由使用以备将来使用。

F00、F01、F02 函数转换 FeeTech 电机的转速、转矩、电流的数值格式

功能 F03 转换 TSIO 模块 ADC 的数字格式

功能 F04 转换 TSIO 模块温度的数值格式

函数 F05 和 F06 转换 TSIO 模块 PWM 的数值格式

F12 函数将 FeeTech 电机的位置值转换为多转速，否则只能从 0 到 4095，然后在每个新转速从零重新启动。

特殊功能的详细信息在以下几页中说明。

F00~F02 命令选项

目前功能指令选项只有 F00, F01, F02, F03, F04, F05, F06, F12

F00 函数

此函数解码 FeeTech 电机的速度参数，尽管它有两个字节长，但不遵循从 32768 到 65535 的负数的正常编码。

仅将 F00 添加到读取“速度”值的 FeeTech 引擎行。".

F01 功能

该函数对 FeeTech 电机的 Torque 参数进行解码，该参数虽然是两个字节长，但并没有按照 32768 到 65535 的负数进行正常编码，并对 1024 到 2047 的数字进行了进一步的反演。

仅将 F01 添加到读取“扭矩”值的 FeeTech 电机线路。

F02 功能

此函数重新计算电机的电流参数并使其以毫安为单位可用。只有在代码中以 MilliAmperePerUnit 命令开头时，此函数才有效，该命令的编写方式如下例所示：

MilliAmperePerUnit 1 6.5

此示例使用 FeeTech 电机，通过将电机 1 的电流值乘以 6.5 得到电机消耗的毫安。

仅将 F02 添加到读取值“Current”的 FeeTech 引擎行。

从 F03 开始的命令选项

F03 功能

此函数解码来自 TSIO 模块的 ADC 的值。 ADC 是 12 位，产生从 0 到 4095 的整数值，但通过添加此选项，值成为 Theremino 系统的经典浮点值，范围从 0 到 1000。

仅将 F03 添加到读取 ADC 的 TSIO 模块行。

F04 功能

此函数解码 TSIO 模块的温度值。 TSIO 模块以表示摄氏度乘以一百的整数形式发送温度，但添加此选项后，您会得到一个方便的摄氏度值，小数点可达百分之一度。

仅将 F04 添加到读取温度的 TSIO 模块行。

F05 功能

此函数将 theremino 系统的经典值从 0 到 1000 编码到 TSIO 模块的 PWM。 用于 PWM 的 TSIO 模块需要从 0 到 65535 的整数，但通过添加此选项，您可以使用从 0 到 1000 的 theremino 系统的经典值。

仅将 F05 添加到 TSIO 模块的 PWM 线。

F06 功能

此功能与之前的 F06 相同，但执行对数转换，允许 LED 更缓慢地打开。

TSIO 模块的 PWM 线只加 F05　用于逐渐打开 LED

F12 功能

此函数解码 FeeTech 电机的实际位置参数，尽管它有两个字节长，但不会返回多圈旋转，但仅限于从 0 到 4095 的一圈旋转，然后使用模块重复负片和正片 4096（12 位）。

仅将 F12 添加到读取当前位置的行。

日志表

每个设备具有不同的特性，甚至控制表也不尽相同。因此，始终建议查阅各个设备的文档。

FeeTech 伺服电机

幸运的是，FeeTech 伺服电机只有一张适用于所有人的表格。各个电动机的行为差异并不重要，并且寄存器始终相同。

因此，我们能够将所有表格收集到一个方便的 PDF 文件中，您可以从中下载 ***这一页***

我们必须用以下公式纠正所有两个 Byte FeeTech 寄存器的负值，包括读取和写入：**n = -32768 - n**

您将在下一页找到所有示例 使用 FeeTech 引擎注册方案。

对于 Dynamixel 伺服器，必须修改示例 到单个引擎的表中，更正寄存器的数量。

TMOT 伺服电机及 TSIO 模组（2023 年筹备中）

TMOT (theremino 电机) 和 TSIO (输入输出模块) 设备具有不同于 Feetech 的寄存器表。 ***您可以在此页面上找到它们。***

Dynamixel 伺服电机

Dynamixels 的可控性不如 FeeTechs，而且成本更高。

我们还没有全部检查它们，但是来自同一（XL，XC，XM，XH，XW，AX，EX，DX，RX，MX，PH，PM，L，M，H）的 Dynamixel 应该具有几乎相同的表格。

Dynamixel 文档非常方便，并且针对每个引擎都有明确的说明，因此我们还没有准备任何文档。

转到咨询用于每个引擎的特性和寄存器，您可以在此页面上找到所有这些信息：
https://emanual.robotis.com/docs/en/dxl

在左侧，有一个引擎类别的菜单，单击以将其打开，然后在内部可以找到所有带有其图像的引擎。

然后单击单个引擎将打开一个页面，其中包含有关该引擎的所有信息，技术规格，要使用的注册信息以及交流建议。

立即转账

这些命令用于以固定值初始化设备，它们主要用于调整“ Return-Delay”响应时间，最小和最大运动极限以及 PID 参数。

即时传输仅在启动应用程序时发送一次，并在程序中的任何字符更改时再次发送。

这些转移 它们不影响每秒的交换次数（FPS），因为它们仅作用一次且时间很短。 因此，我们不必担心会小心使用它们并将它们的使用限制在必要范围之内，就像我们必须对所有其他传输一样。

立即转账示例

0 To DXP1 BC 7-1	'中间值到返回延迟
0 To DXP1 BC 9-2	'MinPositionLimit 的即时价值
0 To DXP1 BC 11-2	'MaxPositionLimit 的直接价值
32 To DXP1 BC 21-1	'对 P 具有立即价值
32 To DXP1 BC 21-1	'对 D 的直接价值
32 To DXP1 BC 21-1	'对我来说是直接的价值
500 To DXP1 BC 48-2	'立即价值 MaxTorque

请注意，在这些示例中，目标设备为 **BC** ，即“广播”，它向所有连接的设备发送相同的值。使用“广播”方法，您无需为所有设备重复说明，从而获得了以下好处： 即使添加或删除设备也不必更改程序。

通过立即传输，您还可以控制单个伺服电机的目的地。这通常没有多大意义，但是在某些情况下，可以使用它来确保电动机在启动时位于预定点

初始化零位设备位置的命令示例

1000 To DXP1 0 42-2	'对目的地的直接价值

转账

这些命令将数值从 Theremino 系统插槽传送到设备寄存器，反之亦然。

以下示例读取插槽 12 的数值，并将其写入从设备 3 的地址 42 开始的四个连续字节中。

Slot 12 To DXP1 3 42-4

- 第一部分 插槽 12 定义广告位（可以从 0 到 999）
- 这个单词 至 指示方向（从插槽到 DXP1）
- DXP1 定义使用 Dynamixel V1 协议的设备
- DXP1 之后的数字定义设备。在此示例中，设备是 3 但可能是 0 到 199。
- 数字 200 .. 254 不能用于指示设备，因为它将指定“广播”发送，即同时发送到所有连接的设备。
- 不能使用数字 255，因为协议将其用作传输开始信号。
- 最后一部分 42-4 表示从地址 42 开始写入 4 个连续字节。

从插槽到设备的传输示例

Slot 22 To DXP1 0 22-2 从插槽 22 到设备零的字节 22 和 23。

Slot 22 To DXP1 BC 22-2 从插槽 22 到“ **BC**”，即所有连接的设备。

Slot 22 To DXP1 0 22-2 A 发送特殊选项“ A”（始终）

从设备到插槽的传输示例

DXP1 2 22-4 To Slot 22 设备 2 到插槽 22 的字节 22、23、24 和 25

DXP1 3 40-2 To Slot 22 朝向插槽 22 的设备 3 的字节 40 和 41

缓中传输

传输速度减慢最多的是日志读数。

通常，您感兴趣的寄存器很多，例如位置，速度，运动指示器，温度，电压，电流等。如果您逐一读取所有这些寄存器，则每次都必须等待答案，这可以以不可接受的方式降低速度。而且，如果有多个伺服器，您甚至可以达到每秒几次交换，并且动作变得零散，并不断停止和启动。

写入寄存器时，可以使用 SYNC 方法解决此问题，该方法将数据发送到多个寄存器而无需等待响应，但是 Dinamixel 协议没有提供用于读取寄存器的有效方法。

因此，我们增加了使用单个响应包读取多个字节的功能。字节累积在一个缓冲区中，可以以非常高的速度从中提取各个寄存器的数据，而无需花费时间进行通信。

要从缓冲区中提取寄存器的数据，必须指出寄存器的地址及其大小（以字节（1、2、3 或 4 字节）为单位）

缓冲读取示例

```
DXP1 0 56-15 To Buffer          '读取 15 个字节（56 到 70）以缓冲

Buffer 56-2 To Slot 100         '位置 从缓冲区到插槽 100 的字节 56 和 57
Buffer 58-2 To Slot 101         '速度 从缓冲区到插槽 101 的字节 58 和 59
Buffer 60-2 To Slot 102         '扭矩 从缓冲区到插槽 102 的字节 60 和 61
Buffer 62-1 To Slot 103         '电压  字节 62     从缓冲区到插槽 103
Buffer 63-1 To Slot 104         '温度 字节 63      从缓冲区到插槽 104
Buffer 64-1 To Slot 105         '同步标志    字节 64     从缓冲区到插槽 105
Buffer 65-1 To Slot 106         '硬错误      字节 65     从缓冲区到插槽 106
Buffer 66-1 To Slot 107         '移动  字节 66     从缓冲区到插槽 107
Buffer 69-2 To Slot 108         '当前 字节 69 和 70 从缓冲区到插槽 108
```

“广播”节目

要使广播命令起作用，必须已经定义了 **DeviceType** 命令。 因此，最好始终在与特定类型设备相关的每个部分的开头指定 DeviceType。

您可以将具有相同值的命令发送到所有连接的设备，这些命令称为“广播”-

这些命令可以节省代码行以及传输时间。它们也是不需要响应的命令，因此它们实际上是瞬时的。

“广播”命令主要用于初始化，但也可以在连续交换信息期间使用，例如，同时更改所有设备上的相同设置。

“广播”命令只是写入命令，因为同时读取许多设备会导致数据冲突。

为了向所有设备发送“广播”命令，使用特殊数字 BC 代替设备标识符。

广播传输的例子

```
0 To DXP1 BC  7-1        '中间值到返回延迟
0 To DXP1 BC  9-2        '立即价值达到 MinPos 限制
0 To DXP1 BC 11-2        '立即价值到 MaxPos 限制
32 To DXP1 BC 21-1       'PID 成比例的中间值
32 To DXP1 BC 22-1       'PID 导数的中间值
32 To DXP1 BC 23-1       'PID 积分的直接价值
```

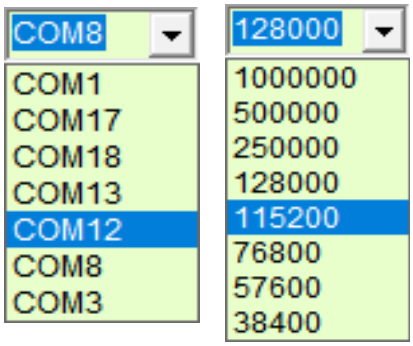
使用这一系列指令（在启动时仅执行一次），所有连接的设备都以相同的方式初始化。

移动所有伺服器的广播传输示例

Slot 100 To DXP1 BC 42-2

该指令移动所有连接的伺服器的目的地。通过更改插槽 100 的值，所有伺服器将一起移动。

COM 端口设置



使用左上方的两个框，您可以选择串行端口和通信速度。

通常使用最大速度，但是如果电缆很长，则会受到限制。

要找到端口，请断开连接并重新连接 USB 电缆，每次关闭并重新打开左侧的框。

如果端口不起作用或已在使用中，则两个框将变为红色。

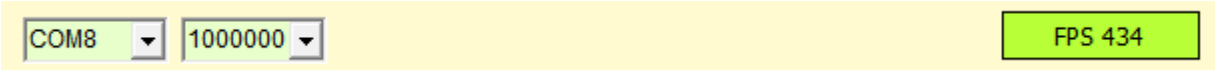
- - - - -

```
' ==========================================================
' ==========================================================
'  DEVICE TYPE FeeTech (STS3215 and other motors)
' ==========================================================
' ==========================================================
DeviceType FeeTech
```

要将波特率发送到设备，必须在与特定类型设备相关的每个部分的开头定义 **DeviceType** 命令。 始终记得在开头指定 DeviceType。

- - - - -

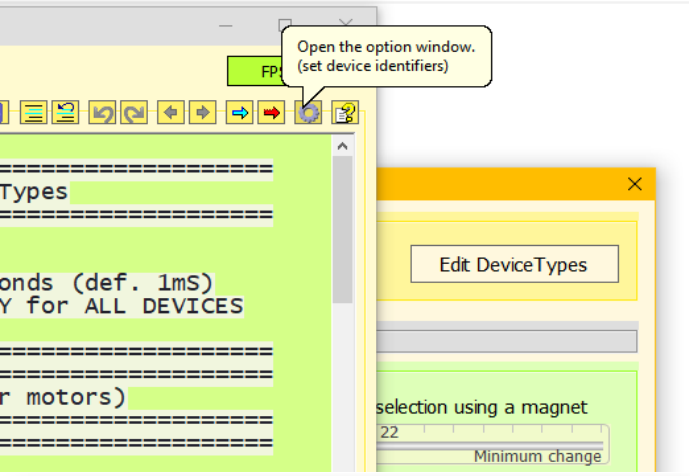
在每次启动时，以及每次更改程序中的任何字符时，通讯速度都会再次发送到所有伺服器。因此，右侧的框应变为绿色，并指示良好的 FPS 速度。



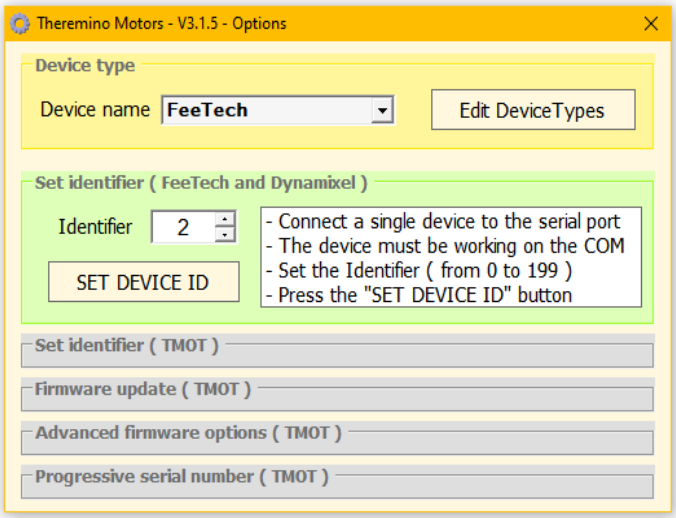
如果右侧的框没有变为绿色，或者闪烁并表明 FPS 太低，则您需要检查与伺服器的通讯线路，还可能需要编程伺服器的标识符，如下页所述。

请注意，至少必须使用一个期望响应数据包的命令，否则将显示文本 "Disconnected"，而不是 FPS。

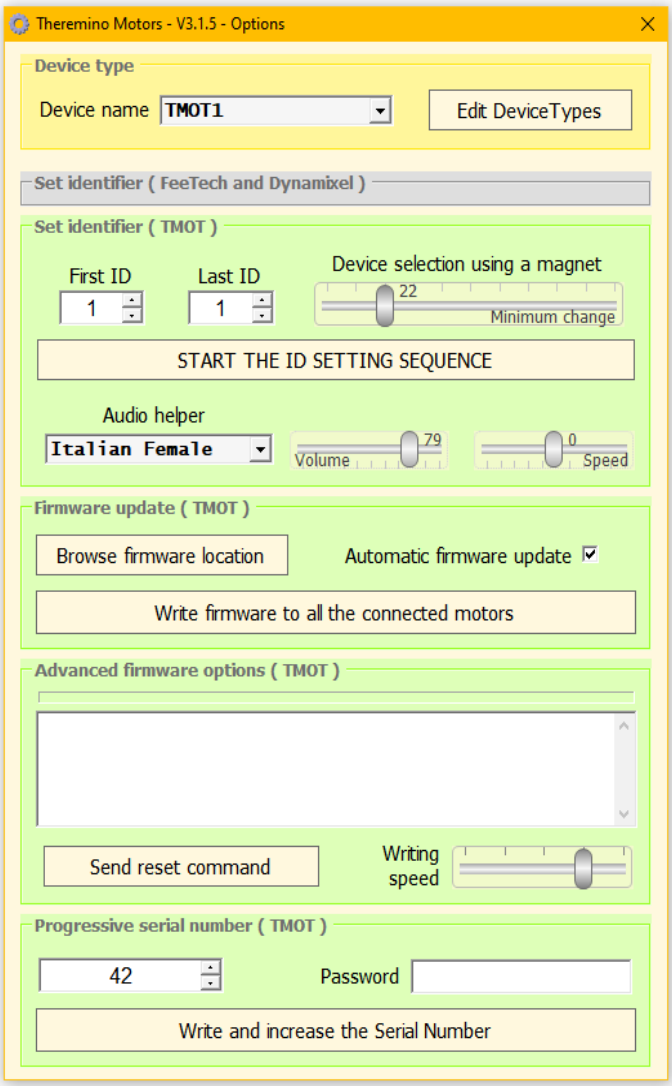
选项面板



齿轮打开选项窗口，将标识符分配给所有设备，并更新固件并将序列号分配给 theremino 系统设备（TMOT 和 TSIO）。



顶部的第一个面板设置设备类型。



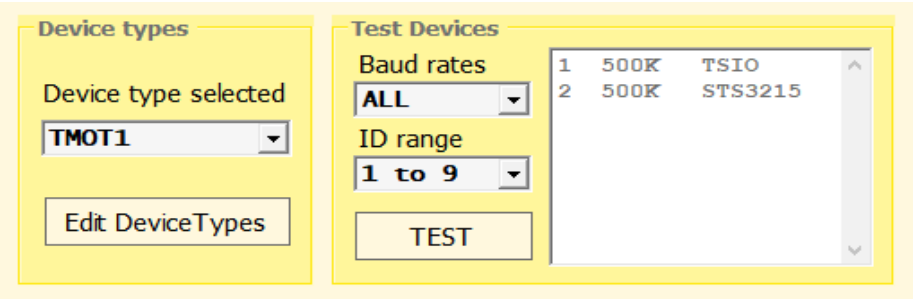
如果您选择 FeeTech 或 Dynamixel，则只有第一个面板可以为这些引擎提供标识符。 对于此操作，您必须将它们从线路上断开并一次连接一个。

相反，如果您选择 TMOT 或 TSIO，您将启用四个面板，您可以使用这些面板分配标识符、更新固件并将序列号设置为 theremino 系统的设备（TMOT 和 TSIO）。 所有这一切都非常方便，无需断开设备与公共线路的连接。

找到连接的设备

组装或修改由多个电机组成的设备的任何人经常会遇到困难，因为他们不知道所连接电机的标识符和波特率。 而且，可悲的是，不了解它们，就不可能用适当的标识符和波特率对它们进行重新编程。

到目前为止，获取此信息的唯一可能方法是使用 FeeTech 的“FT 伺服调试”应用程序，设置不同的波特率并反复按“打开”、“搜索”、“停止”和“关闭”，直到随机识别一个配置或更多引擎并将它们显示在列表中。

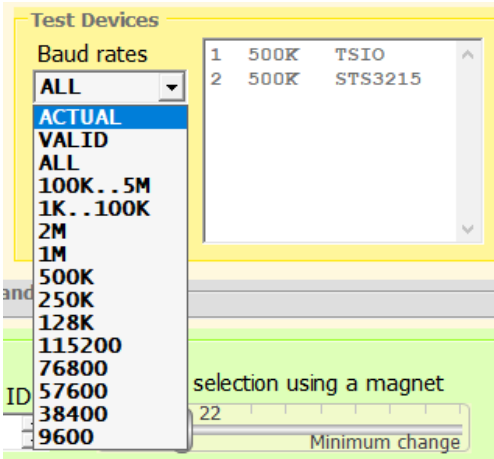


为了方便这些操作，最新版本的电机包含新的“测试设备”面板。

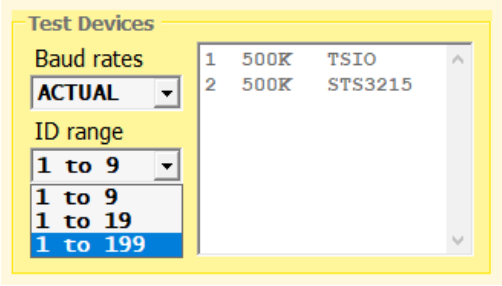
只需按下“测试”按钮，几秒钟后所有设备都会被识别并出现在列表中，每个设备都有自己的 ID 和波特率。

对所有设备执行所有可能的波特率的完整搜索可能需要几分钟时间，因此我们准备了许多选项来将搜索范围缩小到最常见的区域，并将搜索时间缩短到几秒钟。

使用“ACTUAL”时，仅使用当前使用的波特率搜索电机的 ID。 使用“VALID”仅测试当前配置电机的有效波特率。 使用“ALL”测试所有可能的波特率。 对于“100K..5M”，我们尝试从 100K 开始的那些，对于“1K..100K”，我们只尝试从 100K 开始的那些。 最后使用 2M、1M、500K、250K、128K、115200、76800、57600、38400 和 9600 测试各个速度。



将搜索限制在前九个设备会大大减少搜索时间。

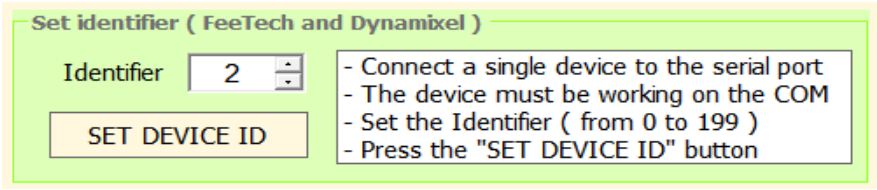


在所有常见情况下，可通过以下方式获得快速有效的搜索：“波特率 = ALL”和“ID 范围 = 1 到 9”

设置设备标识符

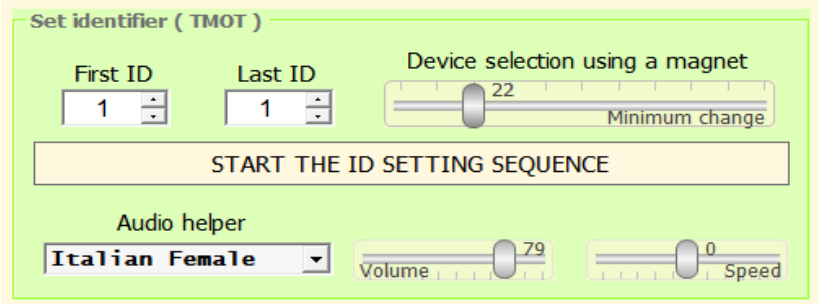
如果在同一条线路上连接两个或多个具有相同 ID 的设备，则会发生通信错误。 在这些情况下，通信速度可能会显着下降，并且指示灯可能会导致“断开连接”字样闪烁。因此，要使用多个设备，必须通过一次连接一个设备并为每个设备分配不同的标识符来准备它们。

Feetech 和 Dynamixel 发动机



- 连接单个设备并确保它正在通信。 如果不通信，用飞泰和舵机软件设置。
- 为每个引擎选择一个从 0 到 199 的不同标识符。
- 按设置设备 ID 按钮。

TMOT 和 TSIO 设备

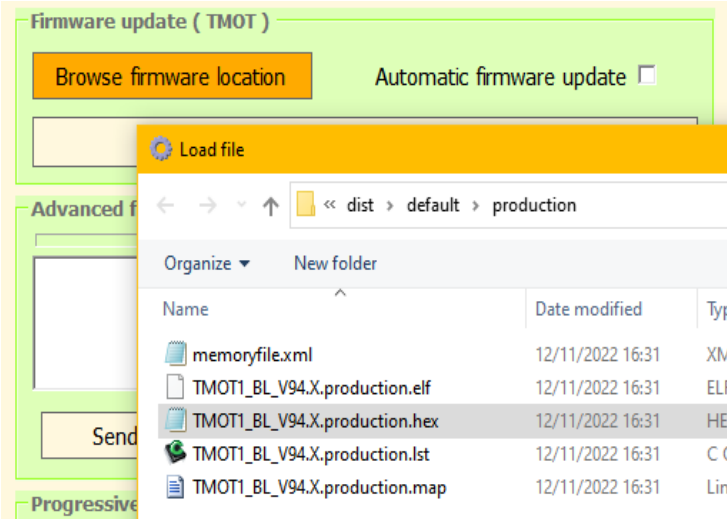


- 所有 TMOT 和 TSIO 都可以保持连接到公共线路。
- 设置 FirstID 和 LastID，然后按“开始 ID 设置序列”。
- 按照语音提示识别设备，当出现提示时，将一块磁铁靠近每个设备。
- 如果灵敏度太低，您必须将磁铁靠得太近，而如果灵敏度太高，则可能会错误地选择另一个设备。
- 如果你犯了一个错误，你可以通过再次按下按钮来停止程序并重新开始。

更新设备的固件

与 FeeTech 和 Dynamixel 相比，TSIO 模块和 TMOT 引擎具有更多的可能性。 其中最重要的一项是能够通过串行线路更新固件。 而且无需拆卸机器人即可一次连接一个设备，这非常方便。

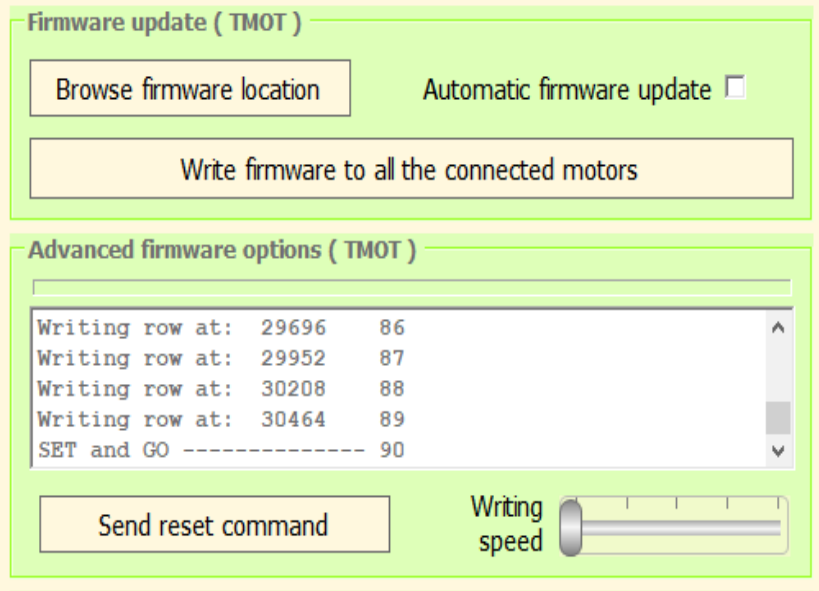
- 按“浏览固件位置”
- 为要编程的设备选择“十六进制”文件。
- 按下“写入固件...”按钮
- 如果启用“自动固件更新”，则每次更改 hex 文件时都会自动开始写入。 因此，程序员可以在每次编译程序时立即尝试更改。



所有连接到数据线（RS485 或 TTL）的同类型设备将被同时编程。 要用相同的固件给其他人编程，只需连接它们并按下按钮，无需每次都重新选择文件。

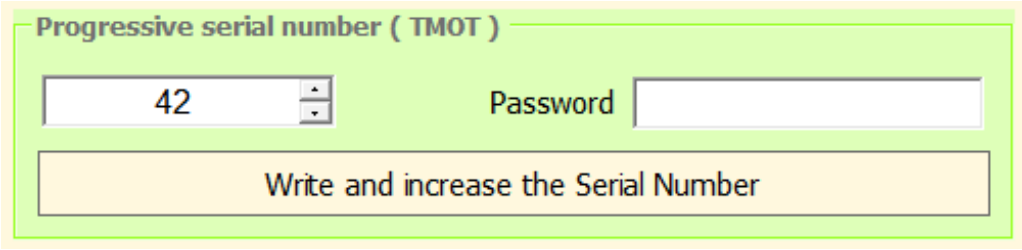
在编程期间，列表显示正在发送的块。

我们建议使用编程速度“4”，因为在某些情况下，速度“5”可能会发生通信错误。
“发送重置命令”按钮可用于测试。
请记住，TSIO 和 TMOT 设备以 115200 波特率重启，因此在重置后您将需要使用波特率框或启用重新连接选项。



设置序列号

与 FeeTech 和 Dynamixel 相比，TSIO 模块和 TMOT 引擎具有更多的可能性。
一个方便有用的选项是通过串行通信线路设置序列号。
当然这个设置不会由用户来做，而是在生产阶段，所以也有设置密码的可能，以防止后续更改。
知道密码的维修人员或制造商可以更改序列号。
如果您不知道密码，更改序列号的唯一方法是擦除所有程序存储器并从头开始，由程序员编写 Boot-Loader。



这个面板的操作很简单：

- 一次连接一台设备。
- 您设置序列号。
- 如有必要，还会设置密码。
- 按下按钮。

序列号会自动递增，因此每次您只需插入设备并按下按钮即可。

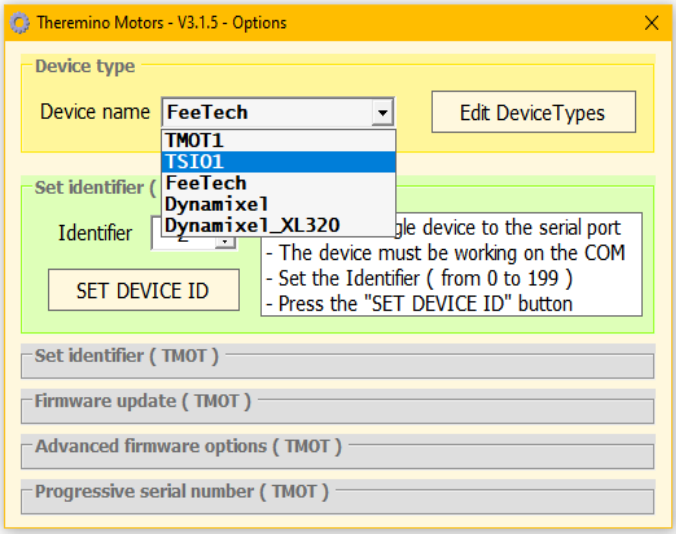
这些操作在生产过程中也很有用，可以检查设备是否正常工作和通信。

设备配置

"Theremino_Motors_CONFIG" 文件必须包含所使用的每种设备的有效值。

使用“编辑设备类型”按钮，您可以检查文件中写入的值，并可能修改它们或添加新设备。 如果使用此按钮，则文件会自动从电机应用程序重新加载。

选项面板命令使用文件中的值。 请记住选择要向其发送命令的设备，如右图所示。



主程序的一些命令（Broadcast 和 BaudRate）也使用配置文件中写入的值。 对于它们的功能，每个程序区域都必须以设备名称开头，如下图所示：

```
' ==========================================================
' ==========================================================
'  DEVICE TYPE TSIO  (Theremino Smart InOut module)
' ==========================================================
' ==========================================================
DeviceType TSIO1
```

FeeTech 和 TSIO 设备的配置示例

```
DeviceName            FeeTech ' 3215 and other motors
Broadcast             254
RegisterID            5
RegisterBaudRate      6
RegisterWriteEnable   55      ' Enable EEPROM writing
RegisterWorkingMode   ---     ' Reset / SetID
RegisterIDtolerance   ---
RegisterSerialAndPwd  ---
BaudRates 1M 500K 250K 128K 115200 76800 57600 38400

DeviceName            TSIO1   ' Theremino Smart IO
Broadcast             250
RegisterID            55
RegisterBaudRate      54
RegisterWriteEnable   50
RegisterWorkingMode   51      ' Reset / SetID
RegisterIDtolerance   56
RegisterSerialAndPwd  80
BaudRates 1M 500K 250K 128K 115200 76800 57600 38400 2M
```

启用EEPROM 写入

FeeTech 和 Dynamixel 伺服器均具有一个存储位置，可在其中写入零以实现对 EEPROM 的写入。通常该位置的值为“ 1”，并且 EEPROM 被阻塞。

如果 EEPROM 被阻止，则在修改某些参数后，它仅在 RAM 存储器中起作用，并且一旦电源电压关闭，此设置便会丢失。

自动启用 EEPROM 的命令

通信速度设置框 (波特率) 和位于选项面板中的 "标识符设置", 自动启用 EEPROM 并在最后禁用它, 从文件中读取要使用的寄存器 "Theremino_Motors_CONFIG.txt"

不自动启用 EEPROM 的命令

除非在主列表中写有启用 EEPROM 的行，否则它们仅在临时存储器中起作用。

建议在开始时将要写入 EEPROM 的所有命令分组，并在它们前面加上使能线。

并且建议在这组命令后加上 EEPROM 禁用线。

请参见下一页中的示例。

启用和禁用 EEPROM

在此示例中，我们看到了通常使用的第一行。这些行设置所有引擎以加快通信速度（零返回延迟和无返回数据包）。此外，在这些行中，它们还确保运动范围是所需的（在这种情况下为 0 到 4095）。可以在这些行之间添加其他命令。

```
    0 To DXP1 BC 55-1        ' Zero to enable EEPROM writing
    0 To DXP1 BC  7-1        ' Zero to Return Delay
    1 To DXP1 BC  8-1        ' ReplyMode 0=NoReturnPackets
    0 To DXP1 BC  9-2        '    0 to MinPos
 4095 To DXP1 BC 11-2        ' 4095 to MaxPos (0=NoLimit)
    1 To DXP1 BC 55-1        ' 1 to disable EEPROM writing
```

重要的是要注意， **0 To DXP1 BC 55-1** 的第一行 0 启用对 EEPROM 的写入，至 **1 To DXP1 BC 55-1** 的最后一行 1 再次禁用它。

如果您将 ReplyMode 设置为零，您还必须添加一个命令， 如下页所述。

重要的是在必须写入最后一条命令之后禁用 EEPROM。这降低了错误更改重要参数的风险。例如，如果您更改了电动机 ID，则必须断开所有电动机并逐一重置 ID。

- - -

本页上的示例对所有具有 EEPROM 锁定寄存器在位置 55 的 FeeTech 伺服器均有效。

另一方面，对于 Dynamixel 伺服器，有必要参考各个设备的特性。在接下来的几行中，我们收集了一些 Dynamixel 伺服器的寄存器号。但是，我们建议您始终检查电动机的特性表。

Dynamixel 伺服器使用寄存器 24： *XL-320，AX-12，AX-18，EX-106 DX-113，DX-116，DX-117，RX-10，RX24F，RX-28，RX-64。MX-12W，MX-28，MX-64，MX-106*

Dynamixel 伺服器使用寄存器 64： *XL-430，XC430，XM430，XM540，XH430，XH540*

最后，需要注意的是，Dynamixel 使用相同的存储位置来启用 EEPROM（值为零）和启用引擎本身（值为一）。因此，在设置标识符之后，还需要将值重置为“1”以使电动机再次工作。

回复模式

FeeTech 伺服器具有“重播模式”选项，可用于加快通信速度。

要使用它，必须首先启用对 EEPROM 的写入，然后发送“ ReplyMode = 1”命令，该命令禁止发送确认数据包。

在下一个序列中，我们将看到通常发送给伺服器以初始化它们的命令。

```
    0 To DXP1 BC 55-1        ' Zero to enable EEPROM writing
    0 To DXP1 BC  7-1        ' Zero to Return Delay
    0 To DXP1 BC  8-1        ' ReplyMode 0=NoReturnPackets
    0 To DXP1 BC  9-2        '    0 to MinPos
 4095 To DXP1 BC 11-2        ' 4095 to MaxPos (0=NoLimit)
    1 To DXP1 BC 55-1        ' 1 to disable EEPROM writing

DisableReturnPackets ' <- If ReplyMode = 0 add this line !
```

始终建议在 EEPROM 上写上注释，以免意外更改设备 ID。

*实际上，可能会错误地在 ID 寄存器上发送带有标识符 **BC** 的命令（因此被广播），然后在所有伺服器上写入相同的 ID。在这种情况下，您将必须断开所有电缆的连接，并通过一次连接一台电动机来重置 ID。*

- - - - -

丢弃返回数据包的行如下所示：

```
    0 To DXP1 BC  8-1        ' ReplyMode=0

DisableReturnPackets ' <- If ReplyMode = 0 add this line !
```

通过消除它们，使用四个伺服器可以达到 130 FPS 的通讯速度，使用带有六个电机和机械手的整个 COBOT 臂可以达到 60 FPS 的通讯速度。

！！！ 注意力 ！！！

如果将 RepyMode 设置为零，则还必须添加一行 使用

DisableReturnPackets

命令 否则随着每一个动作 会出现通讯错误 FPS 框会变红片刻。

Slot To DXP 命令的选项

您可以通过在命令末尾添加一些字母（A、M、C、Fxx 和 Xnnn）来修改 **Slot x To DXPn x xx-x** 类型命令的操作。

“M”和“C”选项在页面上有解释：**"C" and "M"** 选项以及在页面上：**测量单位和“C”选项|**

选项 A（英语中的“始终”）

仅当要发送的值至少改变一个单位时，才会执行将值从插槽发送到设备的命令。但使用 A 选项时，它们始终会执行。通信速度略有下降，但可以获得连续的数据交换。这在某些情况下很有用，例如有时会丢失命令的电机。这种情况有时会发生在 STS3215 丢失最后一个目的地并因此停止执行 Cobot 应用程序的情况下。

例：**Slot 1 To DXP1 1 11-2 A** '总是（也可以不更改）

DXP to Slot 和 Buffer to Slot 的选项

您可以通过在命令末尾添加选项（M、C、F01、F02、F12 和 Xnnn）来更改类型为 **DXP1 x xx-x To Slot x** 和类型：**Buffer xx-x To Slot x** 的命令的行为。

如果需要，它们也可以用连字符（减号）字符或下划线字符分隔，如下例所示：

DXP1 x xx-x To Slot x MCF12
DXP1 x xx-x To Slot x C_F12M
DXP1 x xx-x To Slot x F12-MC
DXP1 x xx-x To Slot x M-F12_C
Buffer xx-x To Slot x CM-F12
Buffer xx-x To Slot x F12-MC

- M 和 C 选项表示：镜像（旋转反转）和校准（测量单位和归零）。
- F00 ~ F99 选项为数值转换功能。

页面上解释了选项和功能
“C”和“M”选项 以及接下来的几页。

节和节槽

当所连接的电动机数量众多（例如超过四个，或者可能是十二个，甚至一百个）时，可能需要使用部分。在这些情况下，通信速度可能会下降太多，从而导致明显的问题。如果降到 30 FPS 以下，则运动会变得不稳定并且摇摆，甚至会发生无法控制的振荡。

写入命令（从软件到电动机）几乎是瞬时的，因此始终可以在每个通信周期将其发送给所有电动机。此外，仅当值更改时才发送这些命令（请在前几页中检查要使用哪些命令以获得最大速度）。

另一方面，就时间而言，读取命令非常昂贵。在使用最佳命令时，每个引擎必须做出响应，并且至少需要几毫秒的时间。因此，如果查询数十个电动机，则通信速度太低。

为了解决这个问题，我们增加了仅激活程序某些部分的可能性。因此，可以始终保持运行电机的指令始终处于运行状态，并且仅偶尔读取电机数据，或者更好的是一次读取它们。因此，该方法对于偶尔检查电动机的温度或电源电压可能有用。

要使用各节，请建立一个控制槽并用以下语句进行设置 **SectionSelectorSlot**。如果缺少该指令，或者将其注释掉，或者如果插槽号无效，则将执行程序的所有部分，并执行指令 **Section** 它们不再起作用。

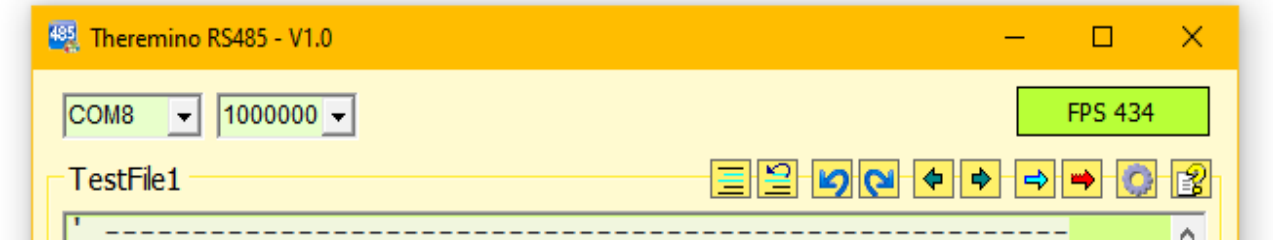
使用节的示例

```
SectionSelectorSlot 11          ' 控制部分的插槽设置
始终遵循的说明
Section 1                       ' 段开始标记 1
第 1 节中的说明
Section 2                       ' 区段开始标记 2
第 2 节中的说明
```

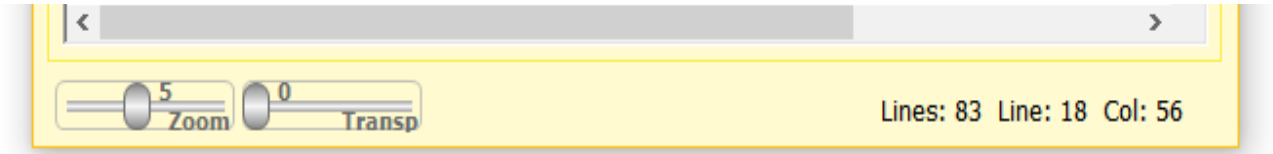
根据插槽 11 的当前数值，仅执行相应部分中的指令。

命令之前的所有语句 **Section 1**，始终执行。如果您想写一个 **Section 0** 在开始时，但这不是必需的。

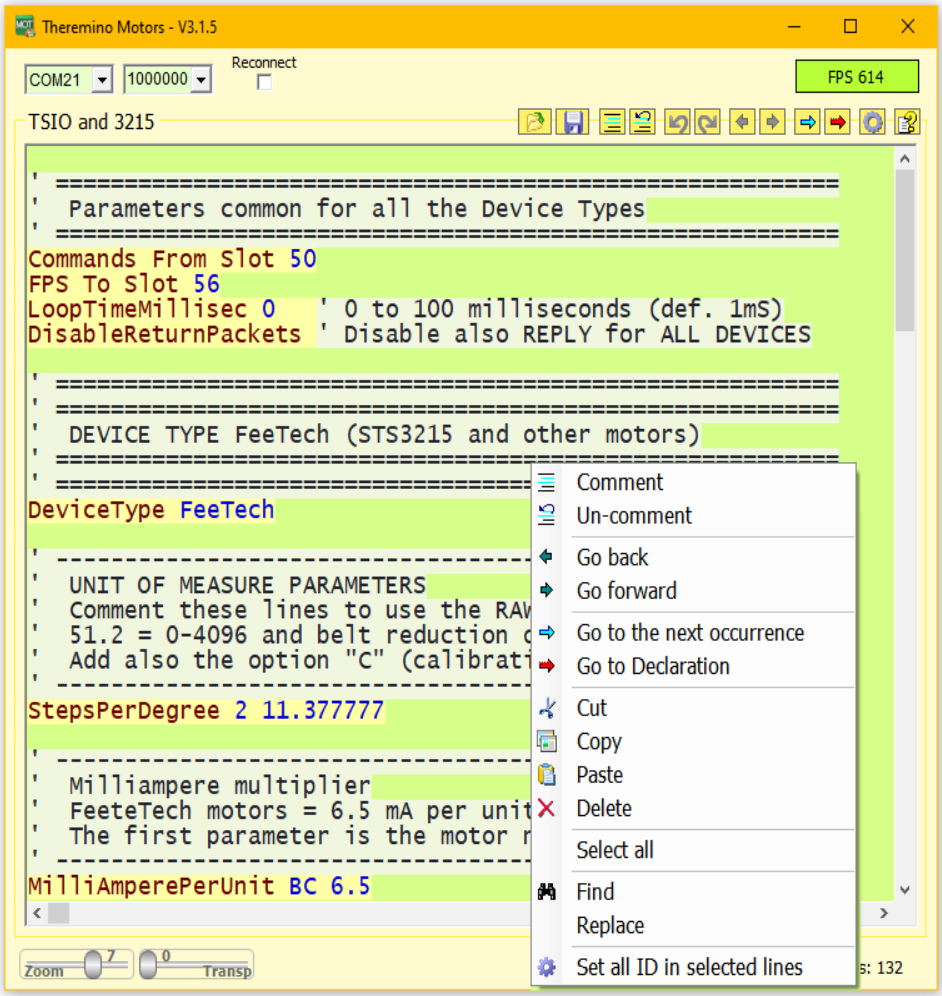
应用程序控件



在上方区域中，我们可以找到通讯控件和仪器按钮。



在下面的区域中，我们找到大小和透明度控件以及有关光标位置的信息。

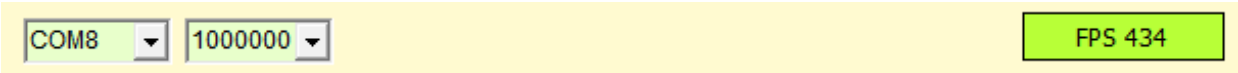


通过在程序区域上单击鼠标右键（或通过触摸屏两秒钟），将显示以下菜单。

这些检查将一一说明在下一页上。

顶部栏上的控件

使用左侧的两个控件，您可以设置通信端口及其速度（波特/秒）。使用右侧的框，您可以检查是否没有错误发生，并以 FPS（每秒帧数）为单位检查通讯速度。“框架”表示发送和接收所有已编程的命令行。

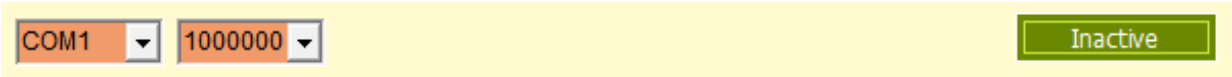


使用正确的设置，您应该以 400 ... 500 FPS 的速度写入寄存器并读取响应。在某些情况下，可以向缓冲区读取高达 900 FPS 或更高的多个寄存器。

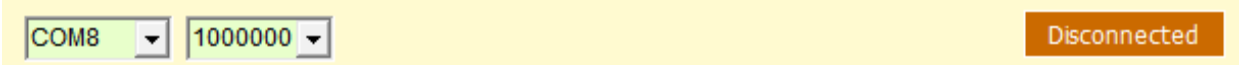
当使用多台电动机时，速度必然会下降，但是如果使用正确的说明，则可以检查一打设备，然后降至低于 50 FPS 并开始出现问题。

通讯错误

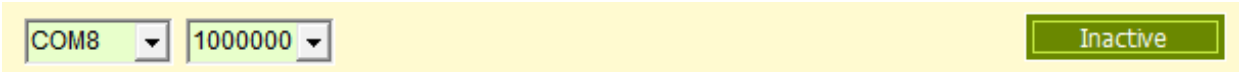
错误分为多种类型：



在此图像中，左侧带有红色背景的两个框表示 COM1 端口不起作用。



但是，在这里，我们看到已连接 COM8 端口，但是一个或多个设备没有响应有效数据包。如果您对 ID 错误的设备，未连接的设备或没有电源的设备进行讯问，则可能会发生这种情况。

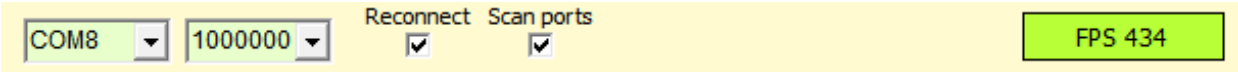


在这种情况下，已连接 COM8 端口，但单词“无效”表示未接收到任何返回数据包。

“不活动”条件也可能不是由于错误而只是由于没有命令需要响应这一事实。 如果只使用广播命令而不使用读取命令，就会发生这种情况。

自动重新连接选项

从 1.2 版开始，我们增加了自动重新连接 COM 端口和与其连接的设备的可能性。



“重新连接”选项会连续检查与设备的通信，如果它们没有响应，它将关闭并重新打开 COM 端口，并重新初始化连接的设备。

“扫描端口”选项还添加了 COM 端口更改。然后依次对所有端口进行测试，直到发现一个端口正常工作并且设备响应为止。

谨慎使用自动重新连接

在某些情况下，设备可能被编程为不响应，并且任何通信命令都不会期望响应。

在这些情况下，自动重新连接可能会认为存在通信错误并导致持续的重新连接，从而阻止了通信。

仔细使用端口扫描

在某些情况下，“扫描端口”选项可能会减慢系统启动速度，并在找到正确的端口之前先检查所有端口。

因此，如果您知道通信端口并且它永远不会更改，则最好不要启用“扫描端口”选项。

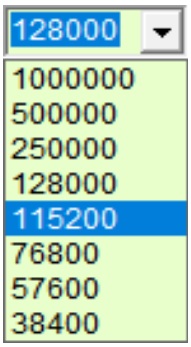
提高通讯速度

为了获得平稳的移动，每秒的交易数（FPS）必须至少为 20，但如果可能的话，最好超过 50。

此外，如果应用程序进行实时检查和更改，则最好将反应时间减至最少，并且交换速率至少为 100 FPS。

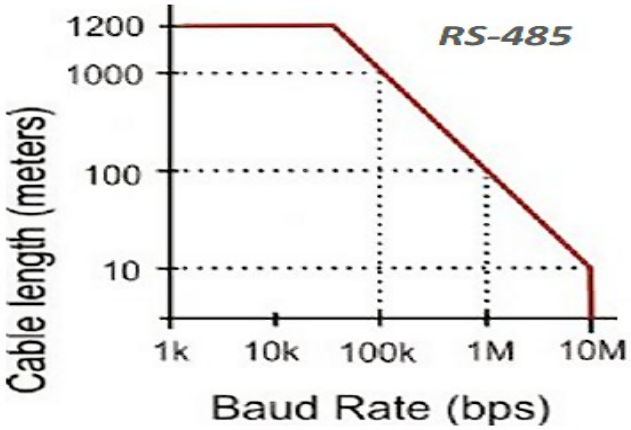
控制一台设备没有问题，但是要使用三个或更多连接的设备来达到这些速度，需要仔细编程。

首先最好将波特率提高到最大（请参阅上一页“COM 端口设置”）



FeeTech 伺服器的工作速率高达 1 兆位，而 TMOT 的工作速率高达 2 兆位。

右图表示限制 仅当电缆超过 100 米时（或 4 Mb Dynamixel 超过 30 米）时，才可以提高速度，



然后，您必须将 LoopTime 和 Return Delay 降低为零

例： **LoopTimeMillisec 0**　　'LoopTime = 0

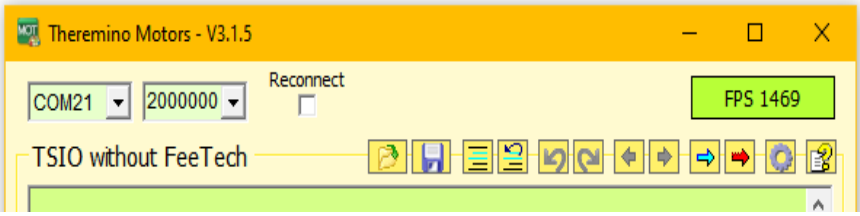
例： **0 To DXP1 BC 7-1** '中间值到返回延迟

（还要记得添加 **DisableReturnPackets** 语句）

最后，仅需使用通讯指令，即可进行快速数据交换

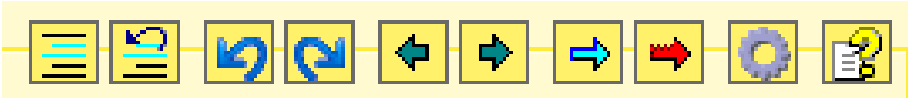
仅读取到缓冲区的指令将用于读取。这些指令使用单个返回包从设备读取整个字节块（甚至数十个字节）。

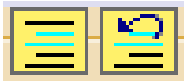
例： **DXP1 1 50-8 To Buffer**



使用新版本的 FeeTech 卡（USB 写入连接器左侧而不是下方），最大数据交换速率约为 1500 FPS。

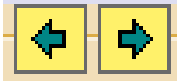
工具列





前两个按钮注释和注释掉所选文本。

两个蓝色箭头 它们用于返回程序更改并重建已删除的更改。

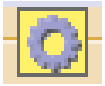


两个暗箭头将光标以及可见页面移动到先前访问的程序部分。

蓝色箭头搜索所有出现的所选单词，甚至只是由文本光标指示。

红色的箭头仅搜索活动单词（不在注释区域中）。

搜索功能非常方便，只需选择一个单词或将光标放在该单词上，然后反复按箭头即可。



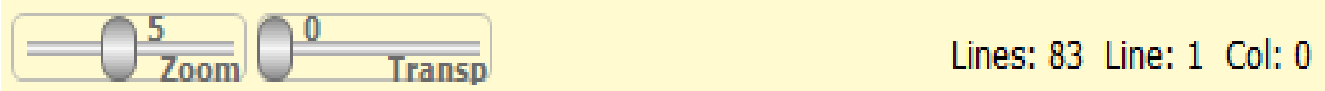
齿轮打开选项窗口，将标识符分配给所有设备，并更新固件并将序列号分配给 theremino 系统设备（TMOT 和 TSIO）。

问号以所选语言打开说明文件（帮助）。为了使此命令生效，您必须将首选语言的帮助文件复制到“文档”文件夹中。最新的帮助文件从以下位置下载 *这一页*。

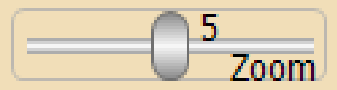
如果找不到帮助文件，则会出现一条消息，提示您打开 Docs 文件夹并将其复制到该文件夹中。

或者，您可以选择首选语言的“帮助”文件，该文件位于“文档”文件夹或任何其他文件夹中。 *要更改所选文件，请单击鼠标右键。* 。

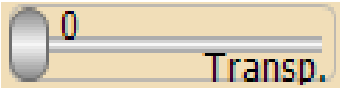
下栏的控件



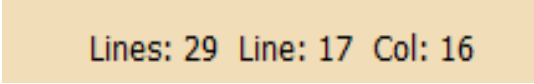
可用鼠标调整光标，并使用鼠标右键返回到默认位置。



缩放滑块可确定文本的大小。



此滑块可调整主窗口的透明度，并允许您在其下方看到。



底部栏的右侧显示有关程序的信息：

-总行数

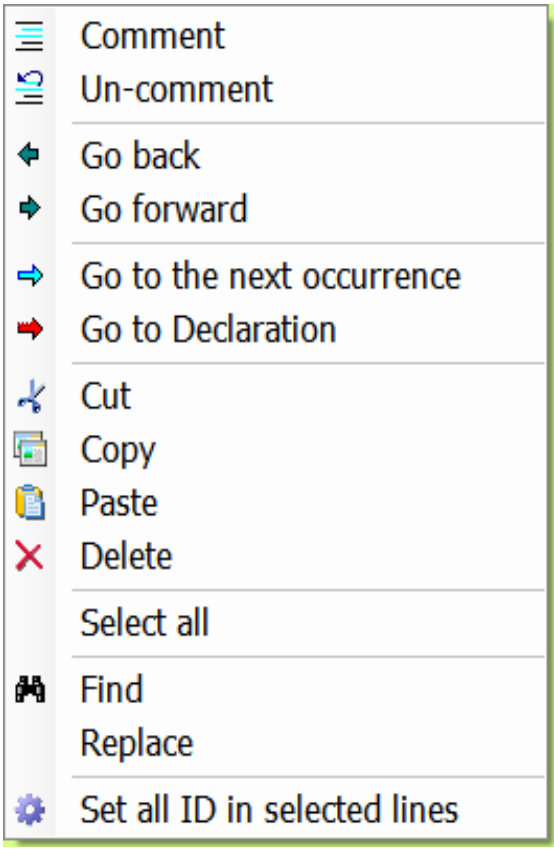
-光标所在的行（从第 1 行开始）

-光标所在的列（从第 1 列开始）

上下文菜单

该菜单显示了一些工具按钮（右上角的按钮）已经可用的命令，但是将它们与其他有用的命令集成在一起。

通过使用鼠标右键单击程序区域（或在不松开手指的情况下触摸触摸屏两秒钟），将显示以下菜单。



评论 是 **取消评论**它们用于对程序的整个区域进行注释（添加初始上标）。或删除评论。

回去 是 **向前** 他们将光标和可见页面移动到以前访问的程序区域。

转到下一个事件 搜索所选单词的其他出现。

前往申报 搜索选定的单词，但仅在活动区域（未注释的部分）中搜索。

切， **复制** 是 **糊状** 剪切，复制和粘贴文本的选定部分。

删除 删除文本的选定部分。

全选 选择所有文本。

找 是 **更换**， 他们打开一个窗口来搜索和替换单词和短语。

Set all ID in selected lines è spiegato nella prossima pagina.

通过使用 CONTROL 键和一些字母，也可以使用键盘来访问此菜单中的某些命令。

CTRL-X， **CTRL-C**， **CTRL-V** 用于剪切，复制和粘贴

CANC 用于删除

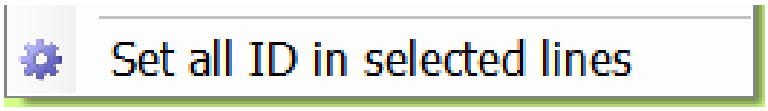
CTRL-A 对于全选

CTRL-F 寻找

CTRL-R 更换

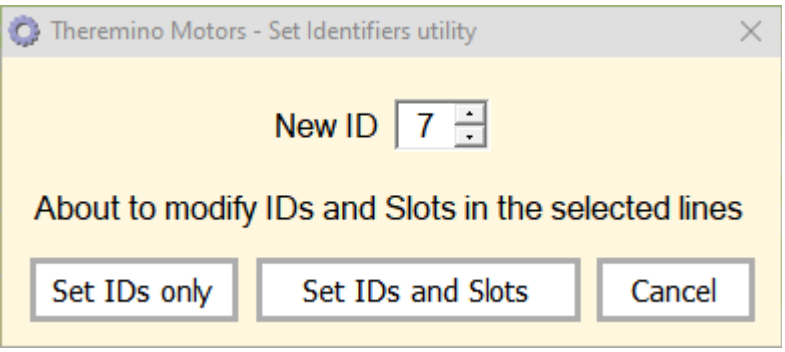
Set all ID in selected lines

此命令修改所有 ID（设备标识符）和在选定文本区域中找到的所有相关插槽。



要使用命令：

- 选择仅包含一个设备的文本区域。
- 打开右键菜单，选择“Set all ID...”
- 从 1 到 9 选择“新 ID”
- 按“仅设置 ID”以仅更改 ID 而不是插槽。
- 按“设置 ID 和插槽”以更改 ID 和插槽。



只需单击一下，您就可以获得需要花费大量时间进行仔细手动操作才能获得的结果。

这是一个将插槽从 2xx 更改为 7xx 并将 ID 从 2 更改为 7 的示例

之前	之后
Slot 201 To DXP1 2 42-2	Slot 701 To DXP1 7 42-2
Slot 220 To DXP1 2 48-2	Slot 720 To DXP1 7 48-2
Slot 221 To DXP1 2 41-1	Slot 721 To DXP1 7 41-1
Slot 222 To DXP1 2 46-2	Slot 722 To DXP1 7 46-2
Slot 223 To DXP1 2 40-1	Slot 723 To DXP1 7 40-1

根据程序的组成方式，在某些情况下，可以选择从头到尾构成整个设备的许多页面，但在其他情况下，可以只选择部分行，因此命令将具有以不同的 ID 重复多次。

请非常小心地使用此命令。 如果出现错误，请使用 UNDO 命令消除更改。

从自动化到电机的命令

要向 Motors 应用程序发送命令，请设置 Slot_CommandsToMotors 变量，然后将其与命令一起用于在文本槽中写入和读取。

Calibrate Motor x To nn.n

该命令使电机 x 的当前位置对应于值 nn.n

nnn.nn 值是一个浮点数，它考虑了所有选项：

- 多速（FeeTech 发动机和其他带有 4096 模块的发动机的选项“F12”）
- 测量单位（例如毫米或度）
- 镜子（选项“M”，即电机旋转方向的反转）

ResetAllCalibrationValues

该命令通过重置包含所有电机校准值的数组来作用于所有电机。

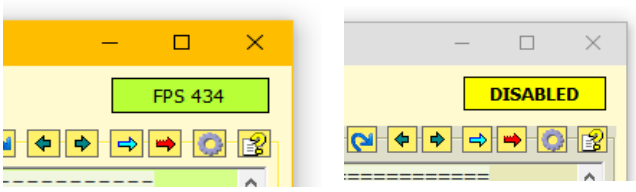
ResetAllMultiturnValues

该命令通过重置考虑已累积的转数的阵列来作用于所有电机。

两个重置命令用于自动化中，如本例所示：

EnableConnection

DisableConnection



当您在右上角的按钮上单击鼠标时，这些命令会启用和禁用通信，这表示 FPS 中的通信速度。

- - - -

有关使用这些命令的示例，请打开包含自动化应用程序的文件夹中的“Commands_to_MOTORS.txt”示例：
Theremino_Automation \ Programs \ Demo Programs \ SlotText Commands