

---

---

# Lab7-1. DeviceNet 网络组态

---

---

情景模拟：工厂新添置了一台设备并带 DeviceNet 接口，电气工程主管找到你，说他需要你通过 DeviceNet 网络控制该设备。

在本实验中，我们假设该设备为 AB 的 1336F 变频器，我们将利用 RSLogix5000 编程软件在 ControlLogix 处理器中创建一个项目，添加 1336F 变频器（带有 1336-GM6 DeviceNet 通讯卡）并编辑简单的梯形图阶梯，通过组态 DeviceNet 网络来演示实际的控制过程。

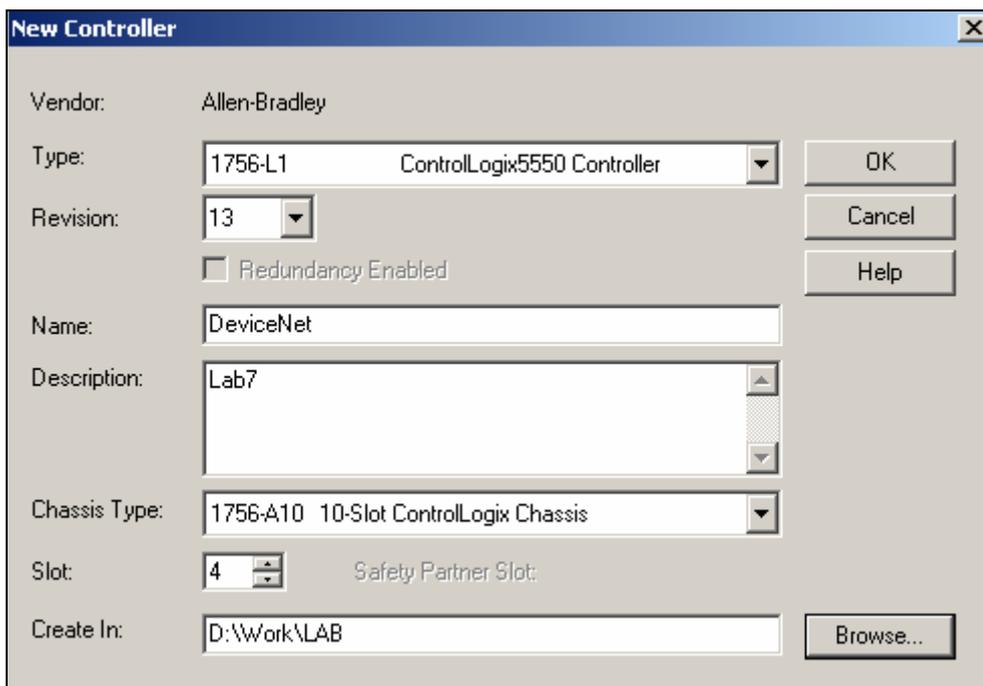
## 实验主题：

- 创建一个 ControlLogix 项目
- 组态 1336F 变频器
- 观察 RSLogix5000 自动生成的对象数据模型
- 添加控制调节频率的梯形图逻辑
- 通过 RSNetWorx For DeviceNet 软件组态 DeviceNet 网络
- PLC 控制调节变频器频率

## 请按步骤：

### 一、创建工程

1. 打开 RSLogix5000 编程软件，点击 File(文件) → New(新建)，打开 New Controller（新建控制器）画面。如下图所示填写控制器的名称、描述（可选），选择控制器类型、版本和所在槽位（起始槽号从 0 开始），指定框架类型、工程保存目录等相关信息，然后按 OK。



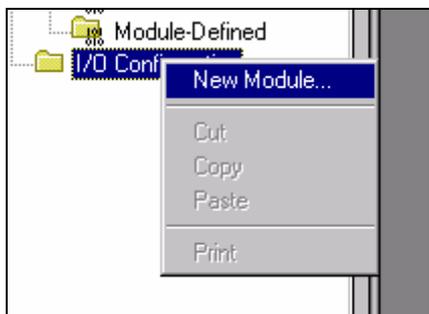
现在我们已经创建了一个 **ControlLogix** 项目。此时我们还没有与项目相关的任何 I/O 模块，项目中也没有可执行的代码（如梯形图），你正在离线工作。所作的任何改变都只限于软件中，并存储在计算机的硬盘驱动中。在进入在线操作以前，这些变化并不能反映到控制器中。

接下来是要确认我们想用在这个项目中的、插在本地背板上的模块。在机架中，有如下设备：

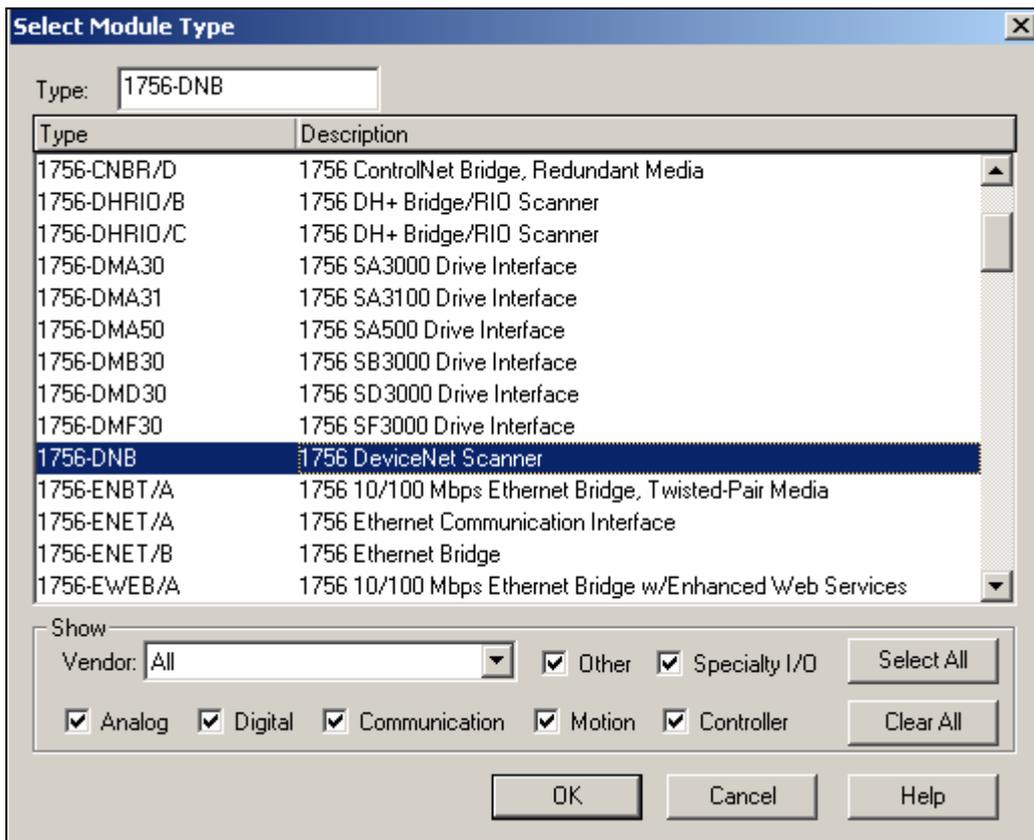
- 0 号槽： 1756-ENBT 以太网通讯模块，IP 地址为 192.168.1.108
- 4 号槽： 1756-L1 Logix5550 处理器模块
- 5 号槽： 1756-DNB DeviceNet 网络通讯模块

注意：所有模块都可带电插拔，如有需要可以另行配置其他模块。

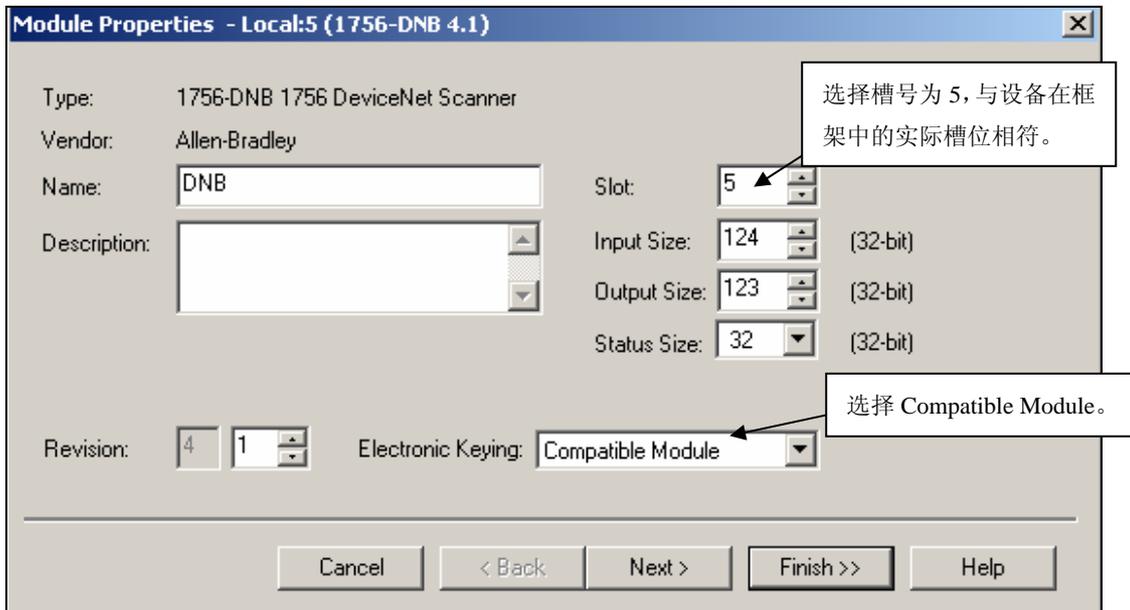
2. 添加本机架的 DNB 通讯模块。在项目管理器窗口中，鼠标右键点击 I/O Configuration (I/O 组态文件夹)，然后选择 New Module(新建模块)。



在模块列表中选中 1756-DNB，然后按 OK。



在模块属性对话框中填写模块名称、描述（可选），选择模块所在的槽位，其他的内容接受默认设置，然后点击 Finish。



Electronic Keying（电子锁）允许你在 online（在线）之前确定一个物理模块与软件组态之间达到何种匹配程度。这种特性可以避免在不经意中将错误的模块插入错误的槽中。它有

如下三种选择:

**Compatible Module** —— 物理模块的模块类型(Module Types)、目录号(Catalog Number)以及主要版本号(Major Revision)必须与软件组态匹配,次要版本号(Minor Revision)必须大于等于软件指定的数值,否则 RSLogix 5000 将不接受所插模块。

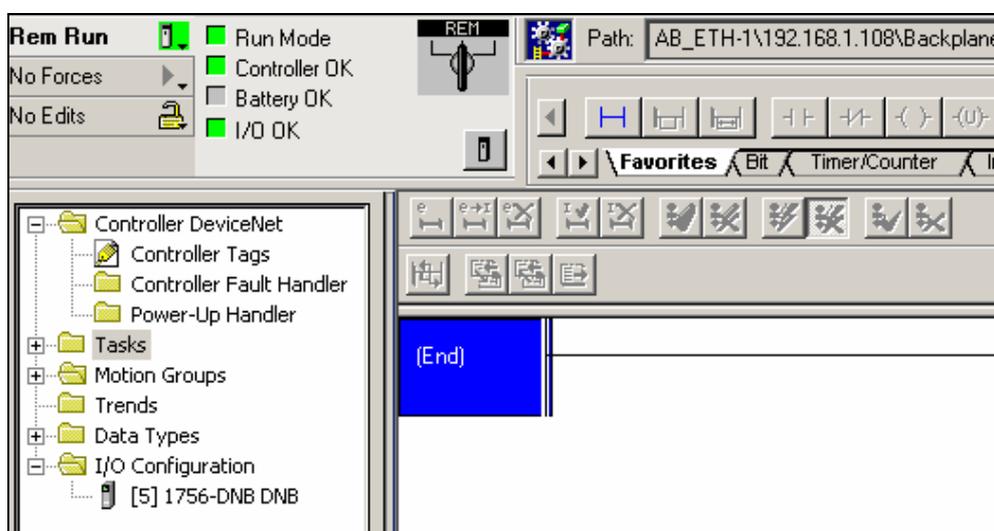
**Disable Keying** —— RSLogix 5000 不会检查模块版本的匹配情况。

**Exact Match** —— 物理模块的下列五个参数必须与软件组态匹配,否则 RSLogix 5000 将不接受所插模块:

Vendor, Product Type, Catalog Number, Major Revision, Minor Revision

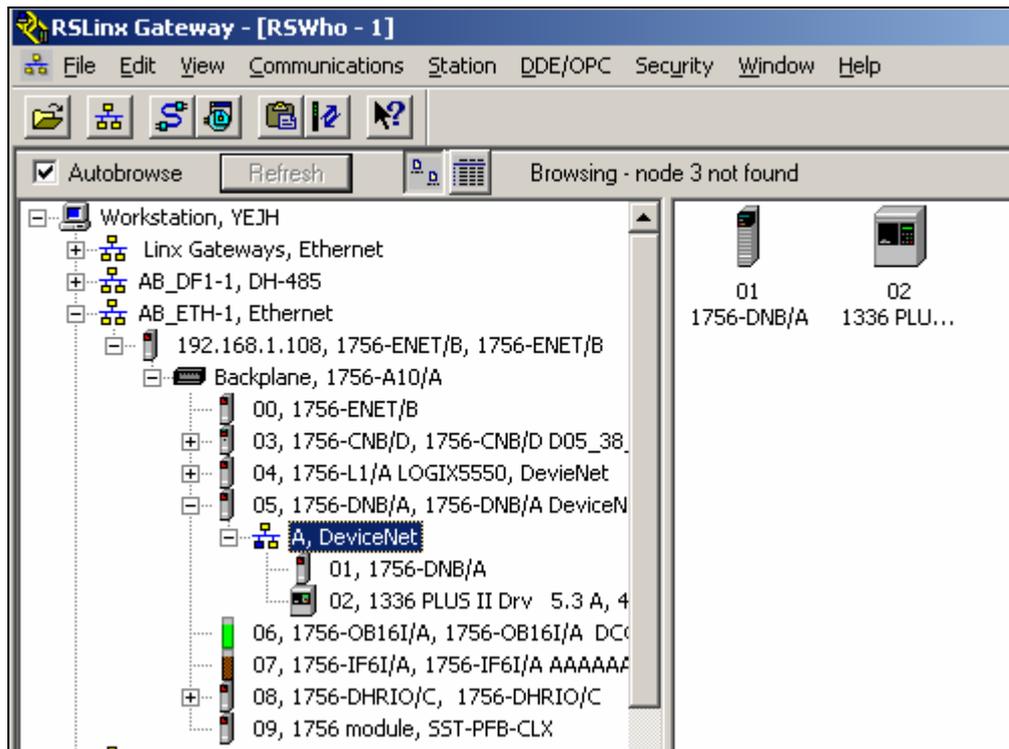
(供应商、产品类型、目录号、主要版本号、次要版本号)

3. 将工程下载到 CPU, 运行一下, 看看有没有问题, 见下图:

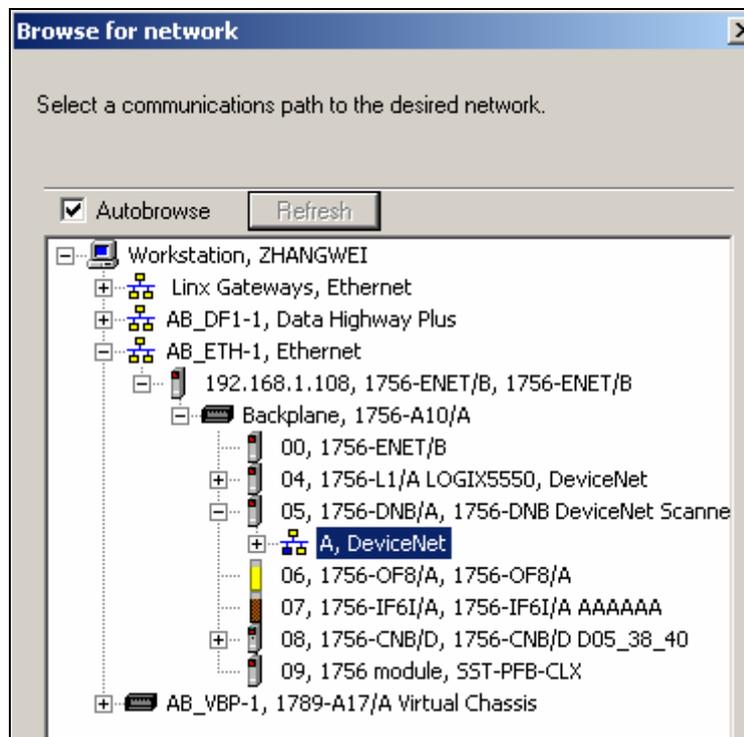


## 二、组态 DeviceNet 网络

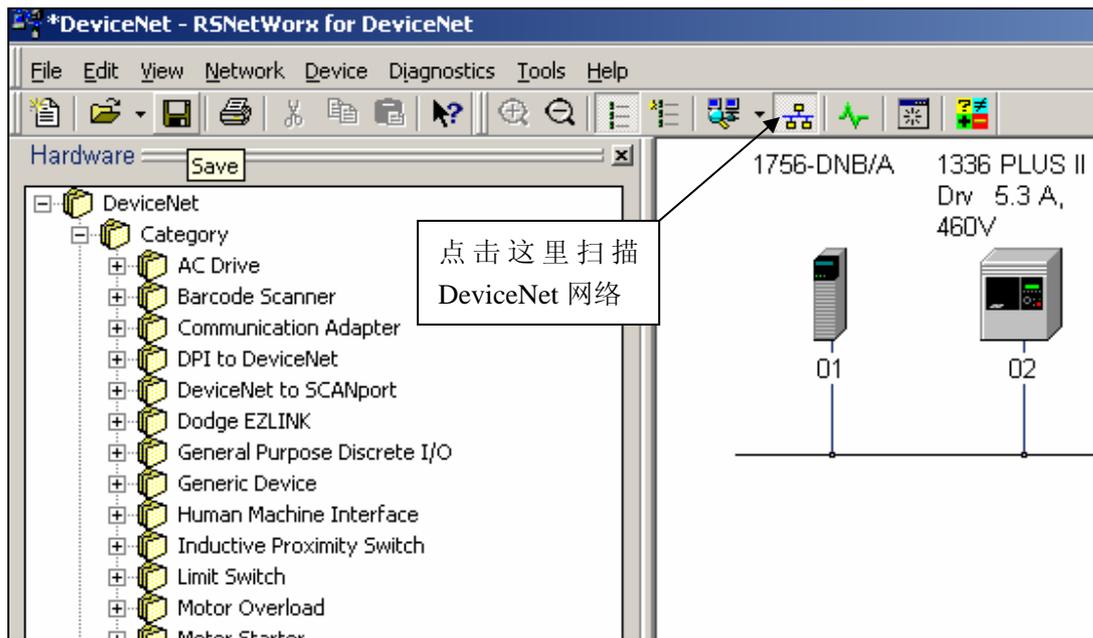
4. 使用 RSNetWorx For DeviceNet 来配置 DeviceNet 网络之前, 首先确认到变频器的通讯网络已连接好, 在 RSLinx 的网络浏览窗口里先找到 1336 PLUS II 变频器, 如下图所示:



5. 打开 RSNetwork for DevicNet 软件, 点击 online 按钮 , 选择通讯路径。选中 DeviceNet 网络, 然后点击 OK。



这样, 组态软件便会自动扫描所有接入 DeviceNet 网络的硬件设备, 并显示在右边的窗口中。



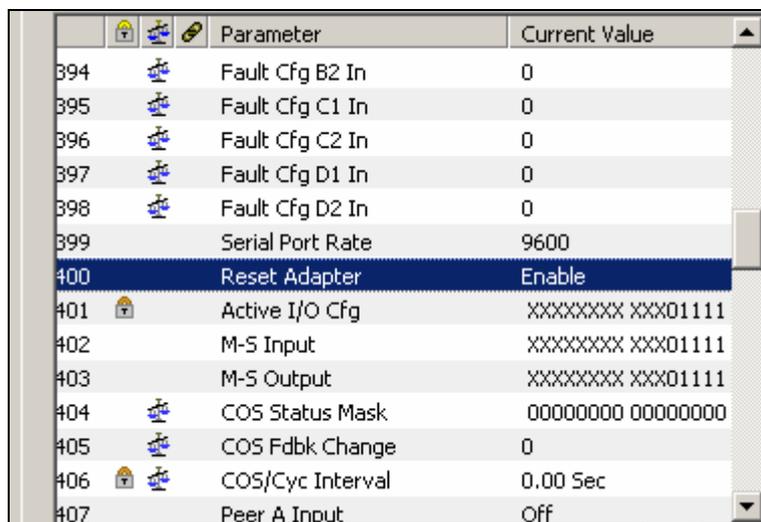
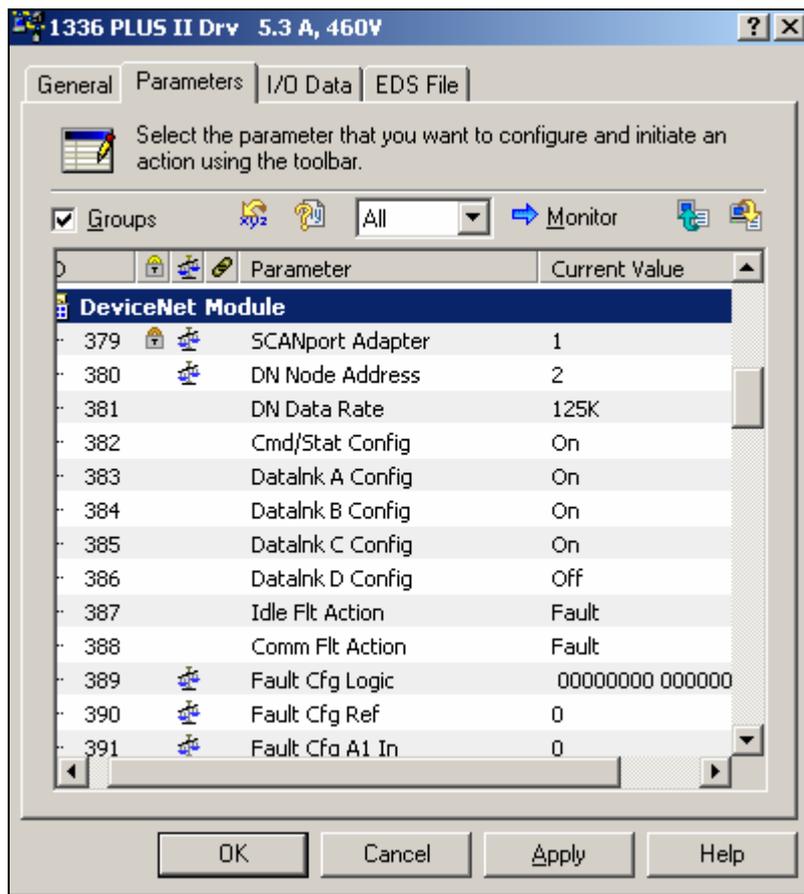
1756-DNB 的 Node 地址为 01；1336F 变频器的 Node 地址为 02，该地址可以在设备属性里更改。注意地址不能重复，且不要大于 63。

6. 双击变频器设备图标，设置变频器的属性：

① 参数设置

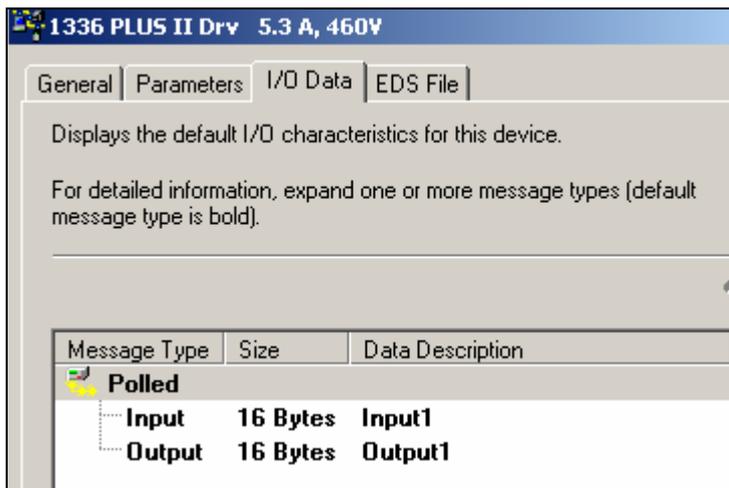
我们设定 381 号参数（波特率）为 125K，它必须和 1756-DNB 的波特率一样；382~385 号参数都设定为 On，与 402 号参数 M-S Input 和 403 号参数 M-S Output 设定一致。（如果是 1336 Impact 变频器，则还要和在变频器里的 Link 设定一致）。

400~403 号参数如图设置即可。



## ② 通讯设置

我们定义通讯格式为 4 个 DINT 进，4 个 DINT 出，所以在建立 EDS 文件（变频器没有 EDS 文件的时候）时要注意为 16 Bytes In， 16 Bytes Out，见下图：

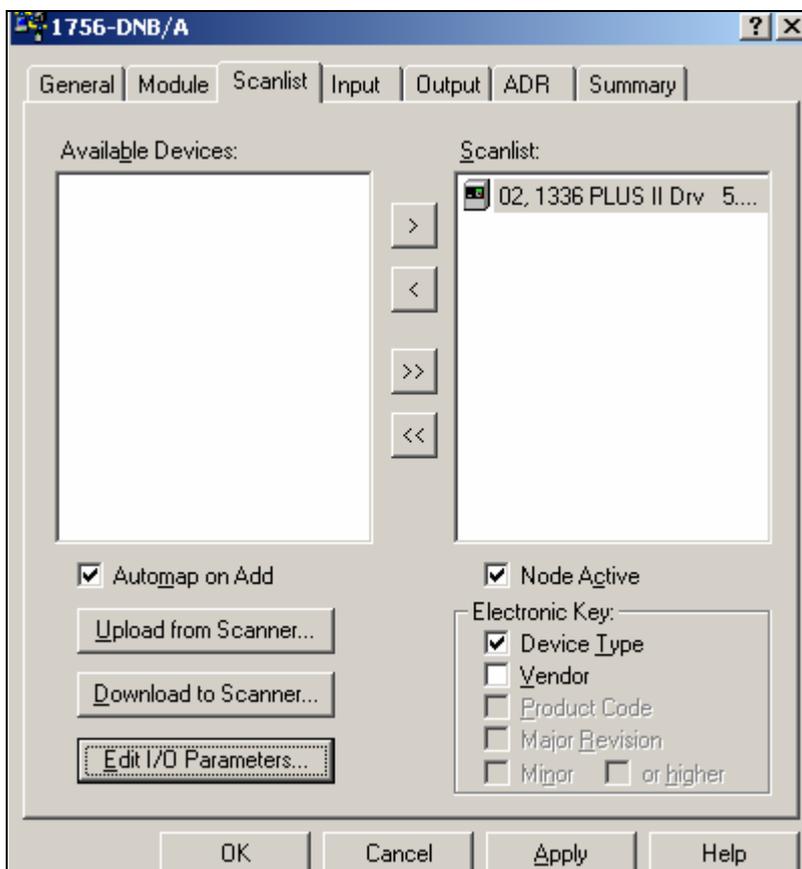


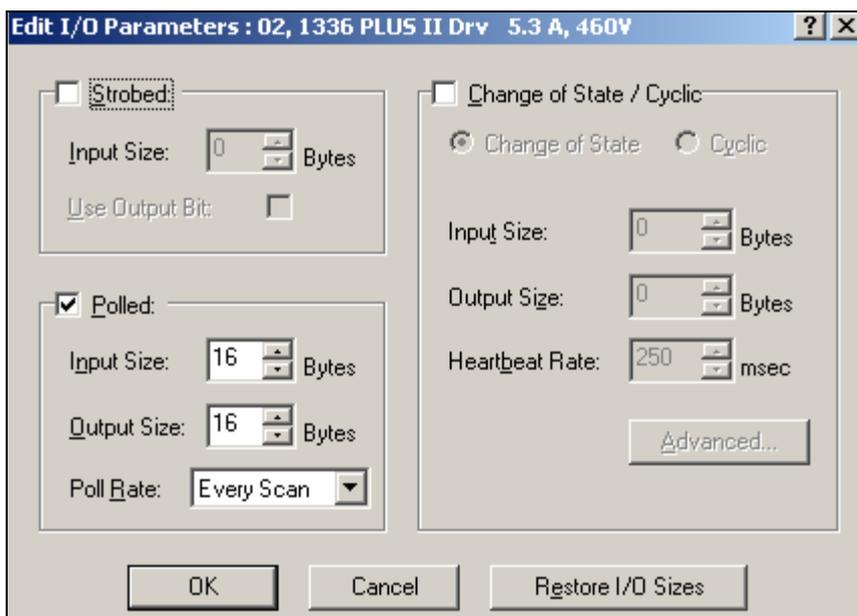
最后 Apply 一下，就可以了！

7. 双击 1756-DNB 模块图标，配置 DeviceNet 网络通讯模块。

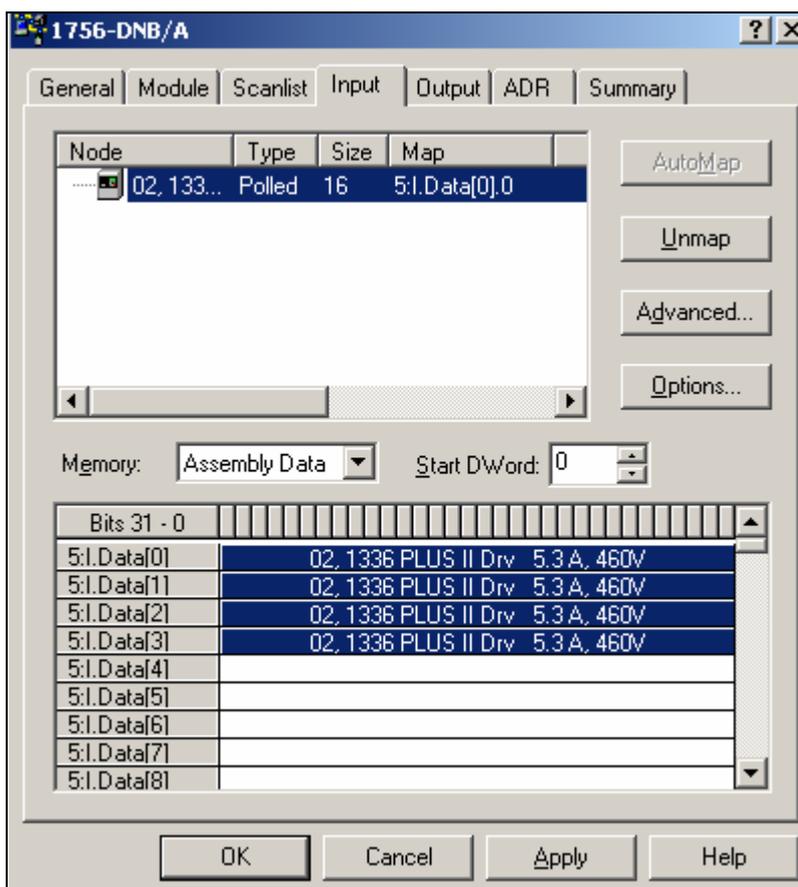
在 Scanlist 里选择你要控制的变频器，点击 Edit I/O Parameters 按钮，进去配置通讯格式。选择 Polled 方式，并且设置 Input Size 和 Output Size 分别为 16 个 Bytes。

另外，注意 Scanlist 中的 Electronic Key 选项，Vendor、Product Code 等都一定不要选中，这是从以后维护方便的角度考虑。





在下图中可以看到输入数据的寻址方式，输出数据也是一样的。最后 Apply 一下就可以了。



### 三、编程

- 完成上述的工作后，就可以编程了，程序的执行效果是能够读取变频器的状态字，启动

和停止变频器。



首先通过上图所示的程序，置位运行位，将 1756-DNB 模块激活，。

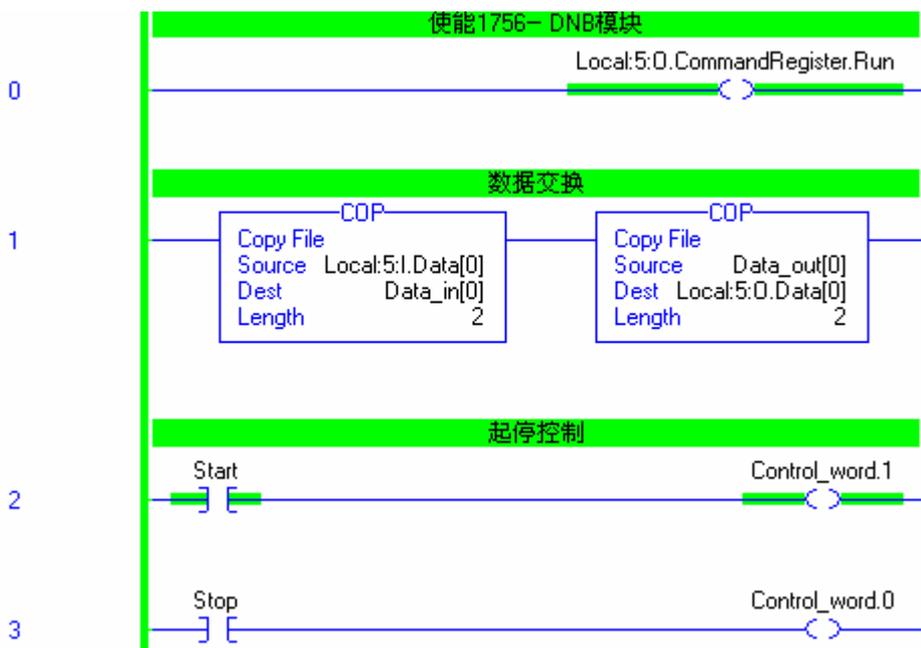
在这里说明一下 1336 变频器数据交换的具体格式：

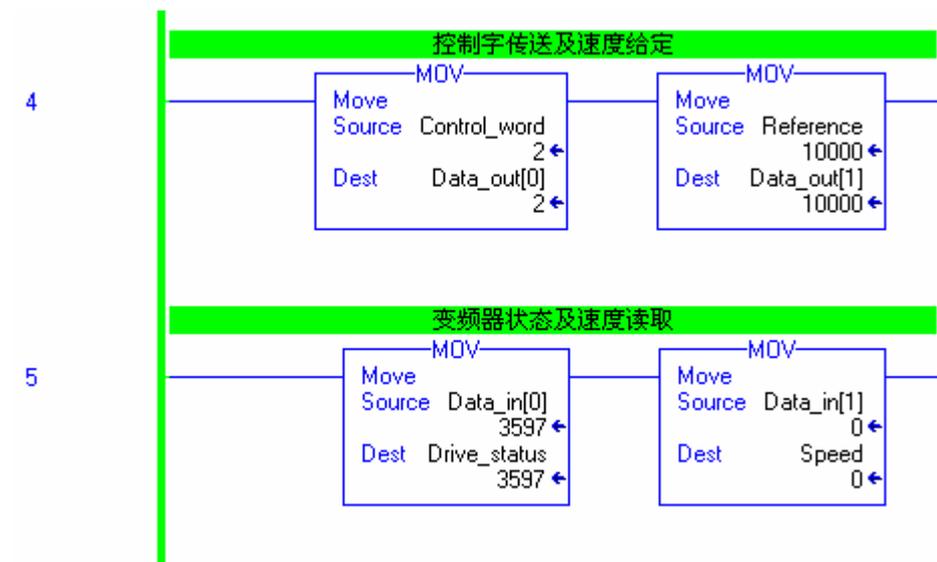
1756-DNB 模块从变频器上读上来 4 个 DINT，第一个 DINT（Local:5:I.Data[0]）的低 16 位的数据为变频器的状态字（每个位的意思可以参考变频器的说明书），高 16 位数据表示变频器的输出频率；第二个 DINT（Local:5:I.Data[1]）为 DataLink A 的数据（包括 A1 和 A2）；第三个 DINT（Local:5:I.Data[2]）为 DataLink B 的数据（包括 B1 和 B2）；第四个 DINT（Local:5:I.Data[3]）为 DataLink C 的数据（包括 C1 和 C2）。

1756-DNB 写入变频器 4 个 DINT，第一个 DINT（Local:5:O.Data[0]）的低 16 位数据为变频器的控制字（每个位的意思可以参考变频器的说明书），高 16 位数据表示变频器的给定频率。第二个 DINT（Local:5:O.Data[1]）为 DataLink A 的数据（包括 A1 和 A2）；第三个 DINT（Local:5:O.Data[2]）为 DataLink B 的数据（包括 B1 和 B2）；第四个 DINT（Local:5:O.Data[3]）为 DataLink C 的数据（包括 C1 和 C2）。

理解了上述的数据交换结构之后，就不难看懂下列的程序了：

将变频器设置为网络给定频率源，Freq Select 1 设定为 Adapter 1，即 DeviceNet 给定速度源！





恭喜你，你已经完成本实验！

**Note:**

---