# 第7章 电子数据表格（EDS）

## 7.1 介绍

该章描述了：

* 设备配置的标准化方法
* EDS文件的结构

为以下人员提供了参考资料

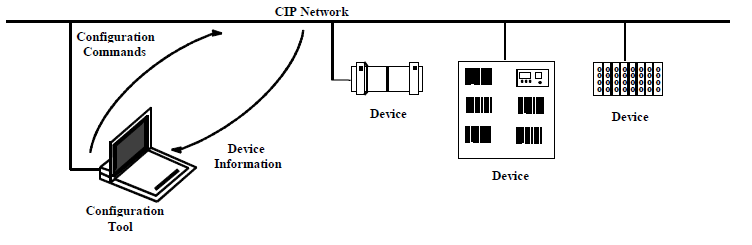
* 产品开发人员
* 配置工具设计人员

### 7.1.1术语和定义

|  |  |
| --- | --- |
| 术语 | 意义 |
| decoded format | The decoded attribute data values in CIP message format |
| EDS | 电子数据表格的英文缩写，一个存储在硬盘上的文件，包含具体设备类型的配置数据 |
| encoded | The encoded attribute data values in the Electronic Data Sheet Format |
| full参数对象 | 设备内的一个对象，包含配置数据，提示字符串，数据转换因子，及其它与设备关联的信息 |
| stub参数对象 | 参数对象的简化形式，只存储配置数据值，且为参数只提供一个标准的访问点 |
| Rack | 通常是包含各种模块的多插槽底板 |
| Adapter | 直接连接到CIP网络的设备，可以插入到Rack的一个或多个槽中 |
| Module | 可以插入到Rack的一个或多个槽中的设备 |
| Adapter Rack Connection | A connection to an adapter whose O->T and T->O application data includes data produced/consumed by one or more modules |

## 7.2 配置选项

CIP标准允许通过网络进行远程设备配置，以及在设备中嵌入配置参数。使用这些功能，你可以为一个特定的应用，选择、修改设备配置项。CIP接口允许访问设备的配置项。



包含只能通过CIP通信接口访问的配置项的设备，需要一个配置工具去改变这些配置项。只通过使用外部开关，跳线组，旋转开关，或者其它专有接口配置的设备，不需要一个配置客户端去修改设备的配置项。但是，设计者必须提供工具访问和确定硬件配置开关的状态。

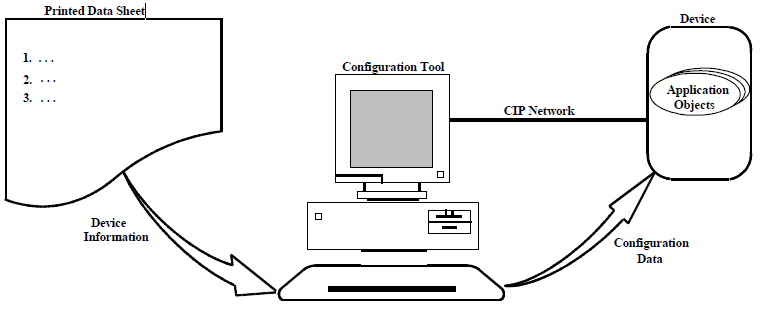
### 7.2.1 配置数据源和方法

This chapter describes possible methods for storing and accessing a device’s configuration data.

1. A printed data sheet
2. An Electronic Data Sheet (EDS)
3. Parameter Objects and Parameter Object Stubs
4. A combination of an EDS and Parameter Object Stubs
5. A Configuration Assembly and any of the above methods

#### 7.2.1.1 使用打印数据表配置

When using configuration information collected on a printed data sheet, configuration tools can only provide prompts for service, class, instance, and attribute data and relay this data to a device. A configuration tool of this type does not determine the context, content, or format of the data. The figure below shows the relationship between configuration information and the tool.



#### 7.2.1.2 使用EDS配置

You can provide configuration support for your device by using a specially formatted ASCII file, referred to as the Electronic Data Sheet (EDS). An EDS provides information about the device configuration data’s:

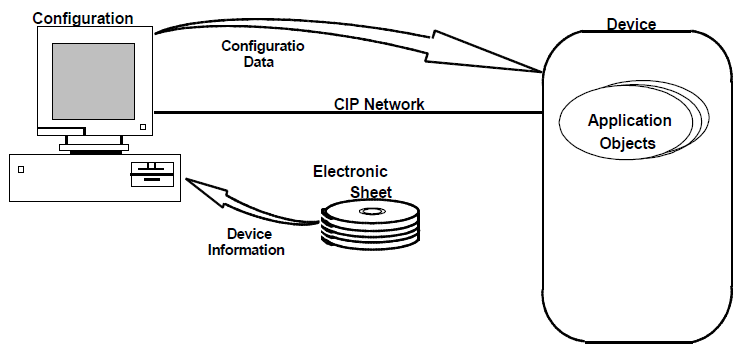
* context
* content
* format

The information in an EDS allows configuration tools to provide informative screens that guide a user through the steps necessary to configure a device. Section 4–3 describes electronic data sheets.

An EDS provides all of the information necessary to access and alter the configurable parameters of a device. This information matches the information provided by instances of the Parameter Object Class. The CIP Object Library describes the Parameter Object Class in detail.

Manufacturers who choose not to supply an EDS on computer-readable media can provide a printed listing of their EDS so that the end user may create the computer-readable EDS using a text editor.

The following figure shows device configuration by a configuration tool that supports an EDS.The application objects in the device represent the destination addresses for configuration data.These addresses are encoded in the EDS.



#### 7.2.1.3 使用参数对象和简化参数对象配置

The device’s public Parameter Object, which is an optional data structure in a device, provides a third method for accessing the configuration data of a device. When a device uses parameter objects, it requires one instance of the Parameter Object class for each supported configuration parameter. Each instance is linked to a configurable parameter that can be an attribute of one of the device’s other objects. Changing the Parameter Value attribute of a Parameter Object causes a corresponding change in the attribute value, as indicated by the Link Path attribute.

**A Full Parameter Object** contains all of the information necessary for device configuration. CIP Object Library, contains a complete definition of the Parameter Object. A partially defined Parameter Object, called a Parameter Object Stub, contains the portion of configuration information needed to perform configuration that excludes user prompts, limit testing, and descriptive text that guides a user through the configuration.

1. 使用完整参数对象

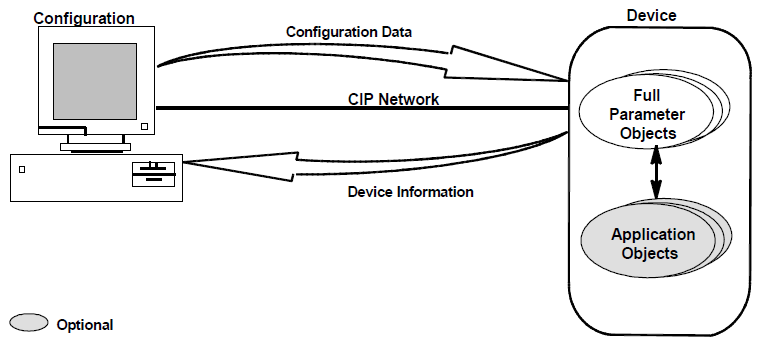
Parameter Objects embed all of the needed configuration information within the device.

Parameter Objects provide:

* a known public interface to a device’s configuration data values
* descriptive text
* data limits, default, minimum, and maximum values

**Important**: When a device contains full Parameter Objects, a configuration tool may derive all of the needed configuration information directly from the device.

The following figure shows the configuration of a device by a configuration tool that supports full Parameter Objects.



2. 使用简化参数对象

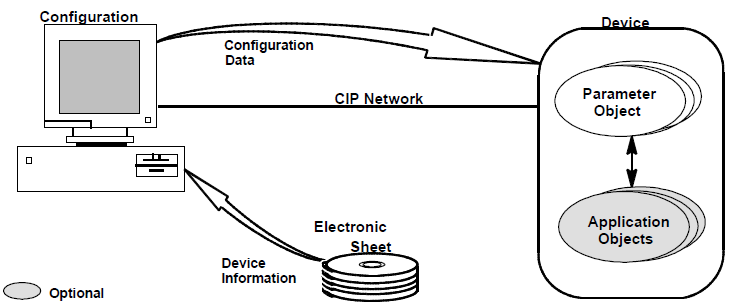
Parameter Object stubs provide an established address to a device’s configuration data values without requiring specification of descriptive text, data limits, and other parameter properties. When a device contains Parameter Object stubs, a configuration tool may obtain additional configuration information from an EDS or provide only a minimal interface to a parameter for modification.

#### 7.2.1.4 使用EDS和简化参数对象

A configuration tool may obtain information from partial Parameter Objects or Parameter Object stubs embedded in a device that provides a companion EDS. The EDS supplies additional parameter information needed by a configuration tool. The Parameter Object stubs can provide a known public interface to a device’s parameter data values, while the EDS supplies descriptive text, data limits, and other parameter properties such as the:

* Data type and size for data validation
* Default data selections
* Descriptive user prompts
* Descriptive help text
* Descriptive parameter names

The figure below shows the configuration of a device by a configuration tool that supports Parameter Object stubs with electronic data sheets.



#### 7.2.1.5 使用配置Assembly

配置Assembly的实例允许批量上传和下载配置数据。如果使用此方法进行设备配置，你必须记录：

* 配置数据块的格式
* 每个可配置属性的地址映射

指定配置Assembly的数据属性时，必须:

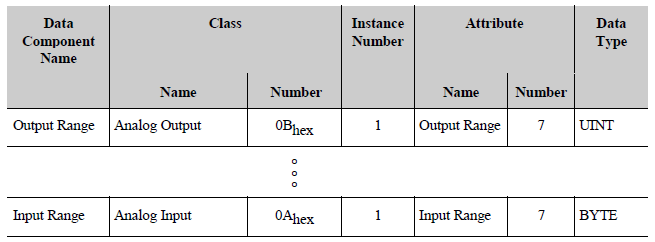
* 按照属性块给出的顺序列出所有数据组件
* 列出大于一个字节的数据组件，从低位字节开始
* 列出所有小于一个字节的数据组件，靠右对齐，从Bit0开始

下表展示了数据块的格式：

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Byte | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
| 0 | 保留 | 保留 | 保留 | 保留 | 保留 | 保留 | 保留 | 保留 |
| ... | ... | | | | | | | |
| n | Input range | | | | | | | |

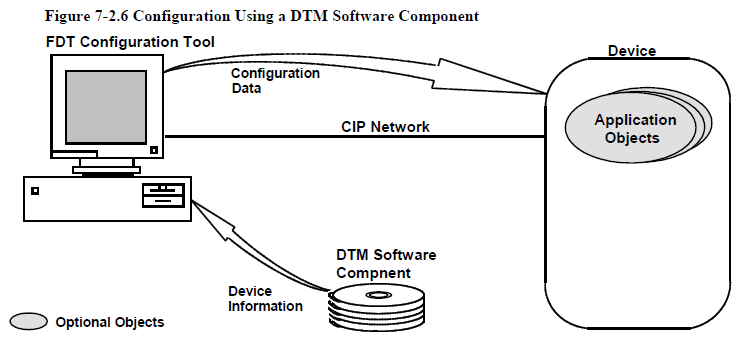
除了指定设备的配置Assembly的数据格式，你还必须把各个配置数据组件映射到其关联的对象。通过指定每个数据组件的类，实例和属性ID来实现。这与指定Assembly对象的Member\_List属性基本相同。

下表显示了配置Assembly数据属性组件的映射示例：



#### 7.2.1.6 使用DTM配置

设备也可以使用设备类型管理（DTM）软件组件进行配置。The DTM is loaded into a configuration software package that supports the Field Device Tool (FDT) interfaces to the DTM. This is demonstrated in below.



Like an EDS file, the DTM contains all of the device specific information for configuring the device. However, while the EDS file is a structured text file describing the data objects, the DTM is a software component that adheres to the FDT interfaces. This allows a greater level of flexibility to add device specific functionality into the configuration interface then can be achieved in an EDS file.

##### 7.2.1.6.1 开发DTM

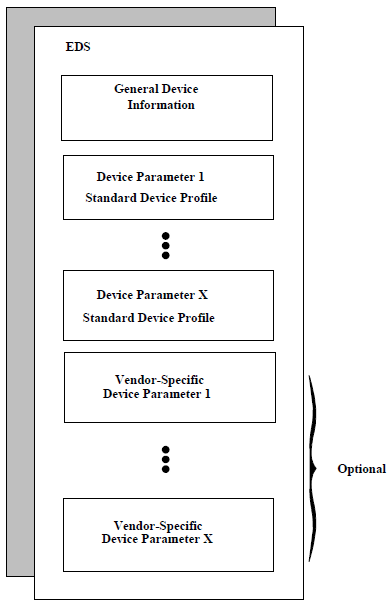
The controlling specifications for the development of a DTM are in the FDT specification and the supporting annex for CIP networks. See www.fdtgroup.org for details.

## 7.3 EDS定义

本节包含EDS文件的正式规范。EDS允许配置工具自动化设备配置过程。EDS规范为所有CIP产品之间的设备配置和兼容性提供了公开标准。

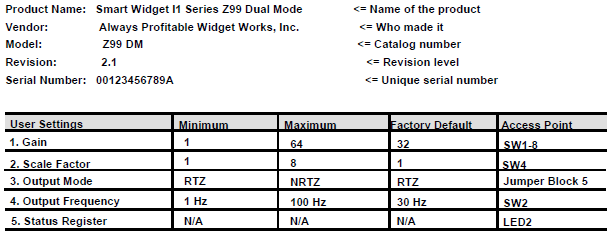
### 7.3.1 EDS

除了本规范定义的所需设备参数信息以外，EDS还包含供应商特有的信息。下图展示了EDS示例的框图。



### 7.3.2 EDS模型

EDS遵循产品数据表隐喻，经过修改以适应CIP要求。通常，产品数据表提供给用户信息，用户用它确定产品功能以及这些功能所选择的用户指定值的范围。



数据表将信息从产品制造商传达给用户。用户解释制造商的数据表，确定必须将哪些设置设置为非默认值，并执行必要的操作以将数据表中的信息导入设备中。

配置工具就是按照这种顺序，通过数据电子表的形式，使用EDS文件和stub参数对象。

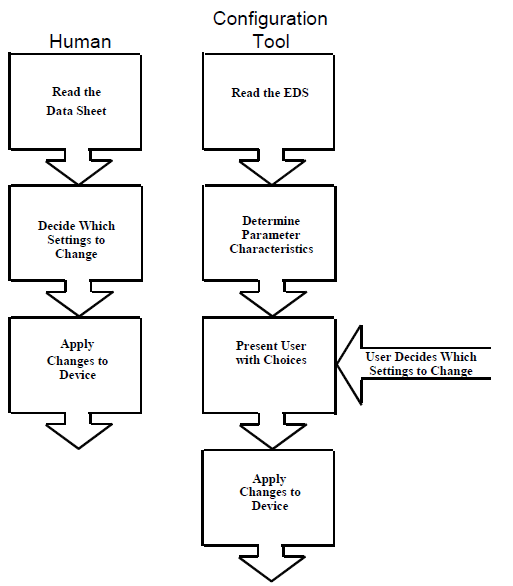
为了执行实际的配置，配置工具使用CIP消息执行设备中的更改。目前，EDS中的文本信息必须是ASCII表示的字符。未来，本规范也将支持非ASCII码的格式，比如UNICODE。

EDS主要有两个目的：

* 描述每一个设备参数，包括它的合法值和默认值
* 为设备中的每一个参数提供给用户一组选项

下图比较了人对打印数据表的解释和配置工具对EDS的解释。至少，CIP配置工具必须：

* 将EDS载入到配置工具的内存中
* 解释EDS的内容，确定每一个参数的特性
* 为用户呈现每个设备参数的选项列表或数据输入字段
* 将用户的参数选择载入设备中的正确参数地址中



产品开发者可以将可选的EDS相关功能添加到配置工具中。对于在CIP环境中执行配置，指定的EDS要求提供了一致性和可兼容的方法。该规范强制执行这些要求，以达到：

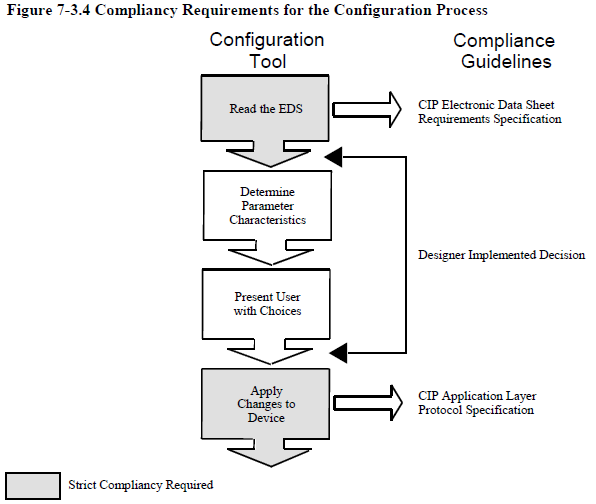
* 为了设备制造商的利益，保持一致性
* 为了工具制造商的利益，保持兼容性

所有的EDS开发者必须遵循这些要求。产品开发者决定其它实现细节。为CIP产品设计的每一个EDS解释器必须有以下能力：

* 读取和解释EDS
* 呈现消息和选项给用户
* 构建必要的消息配置相关的CIP产品

CIP通信模型和协议中的规范，对于通过CIP网络传输的消息施加了严格的要求和协议。下图展示了配置过程的兼容性要求。本章后边的部分描述了使用EDS文件的配置工具所必需满足的要求。这些要求描述了以下内容：

* EDS 编码
* EDS 命令语言
* EDS 语法



### 7.3.3 EDS配合配置工具使用

CIP配置工具：

* 从标准EDS文件中抽取用户提示信息
* 将这些信息以人类可读的格式呈现给用户

工具开发者决定：

* 用户接口格式
* 操作交互方式

#### 7.3.3.1 EDS解释器功能

解释器应该：

* 收集EDS需要的参数选择项
* 构建配置设备所需要的CIP消息
* 包含需要配置的每个设备参数的对象地址

#### 7.3.3.2 EDS文件的组织架构

This section contains the file encoding requirements for EDSs. The EDS encoding requirements define the standard file encoding format to use for CIP products without regard to the configuration tool host platform or file system.

The term “file” as used in this chapter refers to any recognized file format associated with a configuration tool’s file system without regard to the file storage media. An EDS contains an ASCII representation of a device’s Parameter Objects and some additional information required to support object addressing.

##### 7.3.3.2.1 EDS内容

A single file must contain the entire EDS. The following table summarizes the structure of the sections, legal section delimiters, and the placement of sections in an EDS.

表7-3.1 EDS文件的各个段描述

|  |  |  |  |
| --- | --- | --- | --- |
| 段名称 | 分隔符 | Placement | 必要/可选 |
| 文件描述 | [File] | 1 | 必要 |
| 设备描述 | [Device] | 2 | 必要 |
| I/O特征 | [IO\_Info] | 3 or later | DeviceNet专用。查看卷3，CIP的DeviceNet适配 |
| 枚举参数 | [EnumPar] | 3 or later | DeviceNet专用。查看卷3，CIP的DeviceNet适配 |
| 参数 | [Params] | 3 or later | 可选 |
| 参数组 | [Groups] | 3 or later | 可选 |
| 设备分类 | [Device Classification] | 3 or later | 必要 |
| 模块 | [Modular] | 3 or later | 可选 |
| 变体I/O信息 | [Variant\_IO\_Info] | 3 or later | DeviceNet专用。查看卷3，CIP的DeviceNet适配 |
| Capacity | [Capacity] | 3 or later | 可选 |
| 事件枚举 | [EventEnum] | 3 or later | 可选 |
| 符号转换 | [SymbolicTranslation] | 3 or later | 条件可选，查看7.3.6.16节 |
| 公共对象类段 | 参见7.3.6.1.1 | 3 or later | 查看7.3.6.1.1节 |
| 供应商特有对象类段 | 参见7.3.6.1.2 | 3 or later | 查看7.3.6.1.2节 |
| 国际化 | [Internationalization] | 3 or later | 可选 |
| Modbus映射器 | [Modbus\_Mapper] | 3 or later | Modbus专用。查看卷7，Modbus集成 |

当定义EDS文件的时候，遵循下面这些规则:

* EDS 空格 – EDS解释器将会把某些特殊的字符当做空格处理。这些字符，没有以人类可读的字符编码，在文件中空格代替。这些字符有：
  + 新行
  + 回车
  + 换行
  + Tabs，垂直和水平
  + 换页
  + 文件结束符
  + 注释
* 关键词 – 所有的关键词必须使用ASCII码组成，遵循下面的列表：
  + 大写字母A-Z
  + 小写字母a-z
  + 数字0-9
  + 下划线“-”
  + 空格字符（空格只用于段关键字中。空格只能出现在段名称中，多个空格不合法）
* 段 – 必须将EDS文件分成必需和可选段
* 段分隔符 – EDS文件中的每一段必须使用**段关键字**正确分割
* 段关键字 – 使用段关键字分隔符“[”和“]”进行包裹。另外，段关键字也允许非连续地使用字符“/”和“-”。段关键字不区分大小写。有2中类型的段关键字，公共和特定于供应商的关键字
* 段顺序 – 必需的段必须按照要求的顺序排列
* 条目（Entry）- EDS中的每一部分包含一个或多个条目，以条目关键字开头，后跟“=”号。条目关键字的意义依赖于“段”的上下文。分号“;”表示条目的结束，条目可以跨越多行。
* 条目关键字 - 条目关键字由唯一的关键字字符序列组成。 条目关键字不区分大小写。 有2种类型的条目关键字，公共和供应商特定。
* 公共“段”和“条目”关键字 – 这些公共关键字应该由ODVA组织定义，在规范之一的第7章。所有的公共关键字不应该以数字开始。
* 条目字段 – 每一个条目包含一个或多个字段。字段之间使用逗号分隔符“,”进行分割。未提供的可选字段，称之为空字段，应该是下面情况之一：
  + 逗号分隔符之间是空格，或什么也没有（例如：关键字 = x,,x;）
  + 逗号分隔符和分号分隔符之间是空格，或什么也没有（例如：关键字 = x,;）
  + 逗号分隔符和等号之间是空格，或什么也没有（例如：关键字 = ,x;）
  + 分号分隔符和等号之间是空格，或什么也没有（例如：关键字 = ;）
  + 使用分号结束条目，可以忽略掉尾随在其后的可选字段。字段的意义依赖该段的上下文
* 复杂数据字段 - 必须使用逗号分隔符之间的单个值指定的数据指定某些输入字段。 通过使用一组或多组匹配支撑字符“{”和“}”来定义进一步界定输入字段的能力。 大括号字符之间的内容被视为单个项目或条目。 内容可以分组为多个大括号。

### 7.3.4 EDS数据编码要求

#### 7.3.4.1 ASCII字符规约

#### 7.3.4.2 ASCII字符串规约

#### 7.3.4.3 字符串拼接

#### 7.3.4.4 字符串转义符

#### 7.3.4.5 ASCII 无符号整形规约-（USINT，UINT，UDINT，ULINT）

#### 7.3.4.6 ASCII 有符号整形规约-（SINT，INT，DINT，LINT）

#### 7.3.4.7 ASCII字规约-（BYTE，WORD，DWORD，LWORD）

#### 7.3.4.8 CIP Path

#### 7.3.4.9 EDS空格

#### 7.3.4.10 EDS注释

#### 7.3.4.11 EDS 日期

#### 7.3.4.12 EDS 时间

#### 7.3.4.14 ASCII浮点数规约

#### 7.3.4.15 EDS\_URL