

TreeOS.AI 原理图设计命名规则

TreeOS.AI 自动编程系统是通过识别 AD 软件网络表文件实现自动编程，请先下载 AD 工程软件，并学习 AD 工程原理图编辑方法。

TreeOS.AI 需要对元器件标识符 (Designator) 和注释 (Comment) 属性，因此原理图设计命名必须符合命名规则，否则可能导致 TreeOS.AI 无法识别、误识别。

原理图设计主要分为几部分：MCU、功能芯片（或模块）、保护电路、驱动电路、滤波电路等。

一、MCU 命名 (TreeOS.AI 系统目前只支持单 MCU 系统编程)

系统目前支持 MCU 型号请见《TreeOS.AI 系统软件仓库列表》中 MCU 库。

1.1 MCU 属性

器件	标志符	注释	引脚编号
MCU (目前版本只能实现单个 MCU 的代码自动生成!)	不做要求	MCU具体型号+"_主时钟频率MHz", 例如: STM32F103RBT_72MHz。 MCU具体型号必须精确到与器件引脚排列相关的尾缀, 例如: STM32F103 系列 MCU, 103 之后的 "C"、"8"、"T" 与芯片的封装和引脚排列相关, 不可省略, 这三位字符之后的内容则可以省略, 不影响识别。"_主时钟频率"是指 MCU 的运行主时钟频率。如果 "_主时钟频率" 项目空缺, 则按照默认时钟生成代码。	以数据手册为准

• Example:

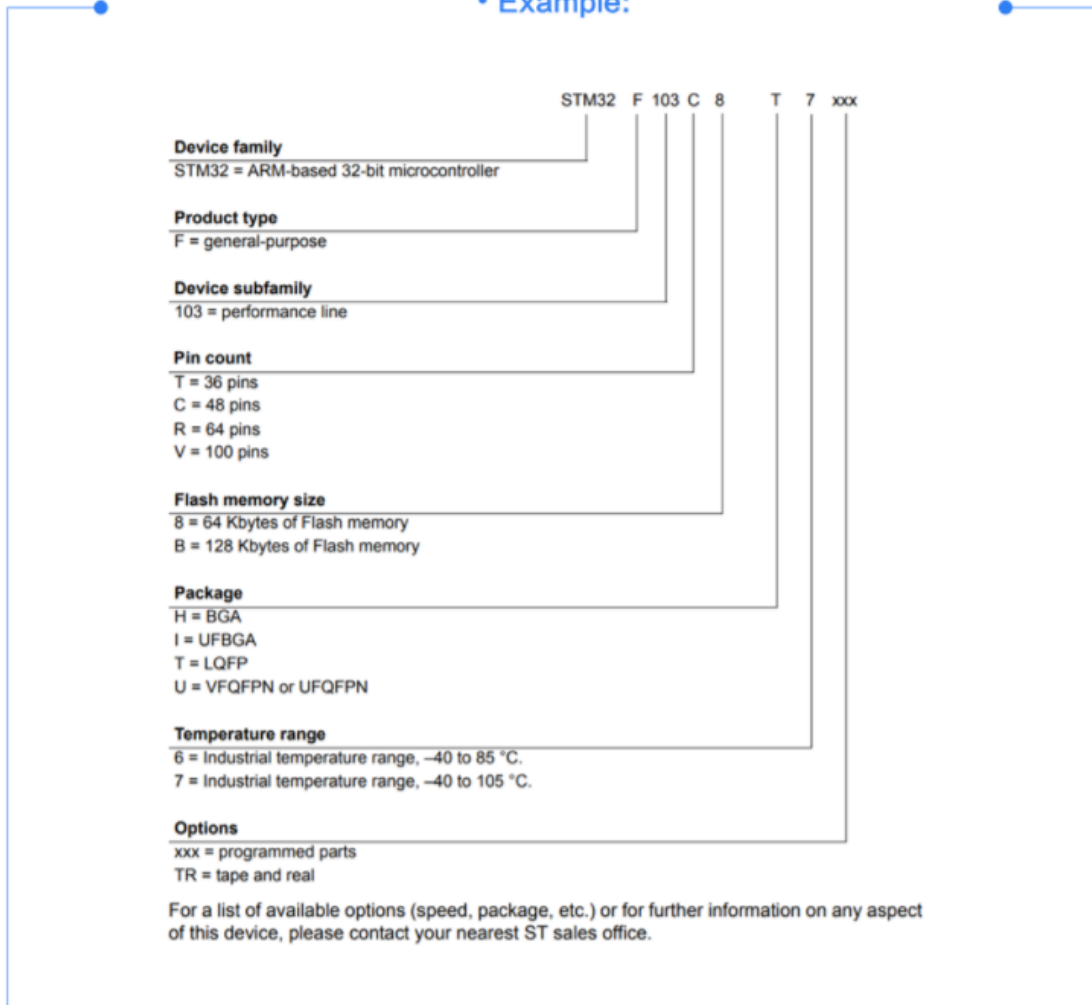


图 1 MCU 命名

1.2 MCU 功能标注

1.2.1 UART 串口功能

如果MCU的串口直接或经电平转换器件（MAX232，MAX485等）连接到某特定功能芯片或模块，则只需正确标识此芯片或模块即可，TreeOS.AI系统会自动识别需要启动的UART串口，无需特别处理；如果MCU的串口直接或经电平转换器件（MAX232，MAX485等）连接到排针或端子等接插件引出，则需将对应的接插件注释设置为“MCU_UART”。

1.2.2 ADC 功能

如果MCU的ADC引脚直接连接到排针或端子等接插件引出，则需将对应的接插件的注释设置为“MCU_ADC”；

如果MCU的ADC引脚连接到模拟信号处理电路，则以下两种情况TreeOS.AI可将对应的MCU引脚识别为ADC功能：

A.MCU引脚与电容相连，

如右图所示：

B.MCU引脚与运算放大器输出端相连



1.2.3 IIC 功能

具有 IIC 功能的芯片，只需芯片标注正确，即可根据芯片与 MCU 连接引脚自动识别，无需特殊标注。例如 AT24C64。

1.2.4 SPI 功能

具有 IIC 功能的芯片，只需芯片标注正确，即可根据芯片与 MCU 连接引脚自动识别，无需特殊标注。例如 RC522。

二、功能芯片（或模块）命名

对于普通功能芯片（或模块），绘制原理图时，应保证期间的注释与芯片（或模块）的名称一致，各引脚标号与芯片（或模块）厂商发布的数据手册（说明书）一致。对于部分器件，需要标注特定内容，通过标识符与注释相结合的方法，达到识别目的。

系统目前支持芯片（或模块）型号请见《TreeOS.AI 系统软件仓库列表》中器件库。

芯片名称及注释方法按《TreeOS.AI 系统软件仓库列表》所列

三、 阻容等常规器件命名

电阻	以“R”或“R”开头	合法字符“0-9”，“R”，“.”，“K”，“K”，“M”，“M”等	设置为1、2
电容	以“C”或“c”开头	合法字符“0-9”，“F”，“.”，“U”，“N”，“P”等	设置为1、2
三极管	不做要求	见本文档第5部分	可设置为E、B、C分别对应三极管的发射机、基极和集电极；也可设置为1、2、3，但此时其与三极的对应关系应以各三极管厂家的数据手册为准，例如：9013和8050三极管手册注明的对对应关系为1-发射极，2-基极，3-集电极
LED指示灯	不做要求	LED	不做要求
矩阵式键盘	不做要求	Keyboard4x4（四乘四键盘） Keyboard3x3（三乘三键盘）	设置为1-8
蜂鸣器	不做要求	BEEP_DC	设置为1、2

四、 电源与接地网络命名

为了保证原理图的完整性与一致性，另外部分芯片需要配置通过电源或接地进行配置（例如 AT24C64），因此采用统一格式的电源与“地”网络号。

- 系统可识别的合法电源网络名（忽略大小写）：
“VCC”，“+5V”，“+3V3”，“+3.3V”，“+3V”，“5V”，“3V”，“3V3”，以“VCC”结尾的网络
- 系统可识别的合法“地”网络名（忽略大小写）：“GND”，“VSS”，以“GND”开始或结尾的网络。

补充说明：

请先确认 MCU 是否在系统 MCU 列表中，若存在可以参照《**TreeOS.AI 原理图设计规范模板**》进行修改。”