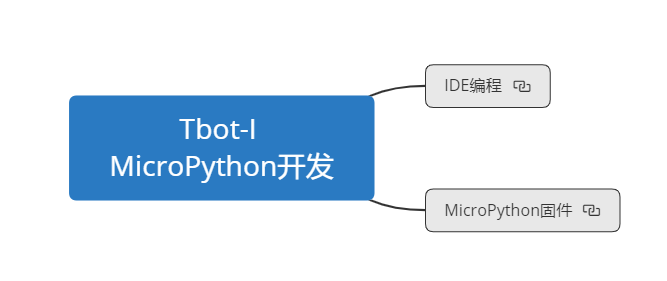
# 雷霆Tbot-I MicroPython实施方案

## 方案架构

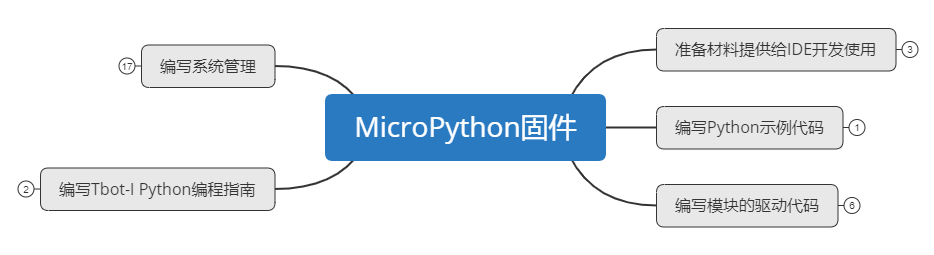
雷霆Tbot-I 进行MicroPython开发，需要进行固件开发和IDE开发。固件开发是要实现产品的系统功能和运行库等等，IDE开发是实现产品应用的编程功能、调试功能等等。



产品简要：Tbot-I运行MicroPython系统，MicroPython系统会解释运行Tbot-I Python脚本代码，用户编程不再需要编译生成系统二进制文件，将Python代码文件下载到Tbot-I MicroPython的文件系统即可。IDE也不再需要集成Arduino编译环境，只需要实现串口通信协议、语句块编程、Python代码编程、交互调试。

1. **固件开发**

基于ESP32 IDF开发MicroPython系统，需要熟悉ESP32 IDE开发环境和MicroPython系统开发，还有各方面的工作。



1. 固件下载更新需要使用到esptool进行固件程序烧录，与原本的模式一样；
2. 需要设计功能库、接口库（由于接口的独特性，普通库硬件控制会影响到设备硬件，要考虑提醒用户不能未经过考虑就使用通用MicroPython库）
3. 代码文件的下载，需要下载到MicroPython的文件系统里面。

* Adafruit MicroPython Tool (ampy) - Utility to interact with a CircuitPython or MicroPython board over a serial connection.（安装命令：pip install adafruit-ampy）；
* 可参考uPyCraft IDE的代码文件下载功能，源码地址：

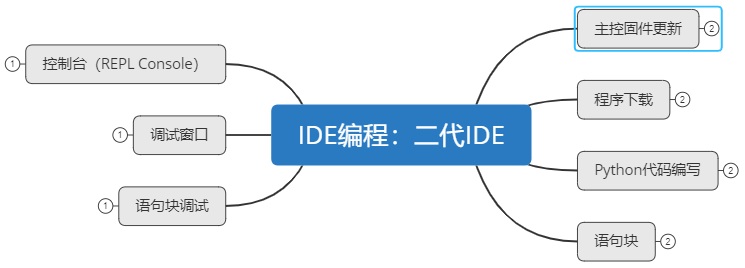
https://github.com/DFRobot/uPyCraft\_src

* uPyLoader源码地址：

https://github.com/BetaRavener/uPyLoader

1. IDE在线调试设备的方法：使用REPL console（通信接口是USB-UART，可以参考pycom的VScode插件工具Pymakr），执行相关代码调试和功能调试。
2. 需要在系统上编程实现交互逻辑，这里的代码可能使用底层系统的C语言编写。需要访问MicroPython的文件系统，实现多程序程序选择执行，所以要能通过底层C语言控制MicroPython的运行来代替命令行的执行功能，还要能修改启动脚本boot.py或main.py，甚至自定义启动逻辑。
3. 编写MicroPython的驱动程序。
4. 编写Tbot-I Python示例代码。
5. 编写Tbot-I Python编程指南。
6. **IDE开发**

IDE开发主要涉及：固件更新、编程功能、调试功能。



1. 主控固件更新：版本管理，固件下载脚本；
2. 程序下载：程序下载到指定程序卡槽（相应的python文件名），控制程序运行/停止；
3. Python代码编写：代码提示，从语句块生成的代码中导入Python代码；
4. 语句块：需要设计一套Python语句块，语句块生成Tbot-I Python代码；
5. 控制台（REPL Console）：REPL是MicroPython的控制台，可以在这里方便调试Python语句，也有利于Python教学用途
6. 调试窗口：继承一代IDE的电机、舵机调试窗口，通过REPL传输调试代码到Tbot-I主控；
7. 语句块调试：每个语句块都是会生成一段python代码，对语句块进行调试时，也就是通过REPL传输这一段生成的代码到Tbot-I 主控；
8. **工作安排**
9. 固件：原来的硬件设备驱动程序都是在Arduino上编写，移植到基于ESP-IDF的MicroPython固件，需要大概4周的时间；
10. 固件：底层系统的分析和优化，并做相关改动以适应我们产品的需求，需要约2周时间；
11. 固件：编写MicroPython的虚拟机底层代码，并测试相关命令，需要约2周时间；
12. 固件：如果要兼容APP 蓝牙遥控功能，需要2周时间；
13. 产品：重新优化相关功能，设计产品应用、系统交互，并编码实现，需要2周时间；
14. 产品：需要2周时间进行产品调整；
15. IDE：搭建平台环境需要2周时间，实现相关固件烧录、运行脚本、烧录脚本；
16. IDE：代码调试功能，搭建和实现功能需要2周时间；
17. IDE：重写语句块生成代码，需要2周时间；
18. IDE：实现代码编程、图形编程的切换功能，需要1周时间；

整体时间安排：

|  |  |  |  |
| --- | --- | --- | --- |
| 固件开发 | | IDE开发 | |
| 系统更新、程序下载、代码调试功能 | 10月第3周 | 搭建基本框架 | 10月第3周 |
| 扬声器、屏幕、LED的驱动程序 | 11月第2周 | 系统更新、程序下载 | 10月第4周 |
| 系统开机启动 | 11月第3周 | 语句块设计，完成基础语句块 | 11月第2周 |
| 电机控制、舵机控制驱动程序 | 11月第4周 | 代码调试 | 11月第3周 |
| 超声波、触碰、彩色传感驱动程序 | 12月第1周 | 电机、舵机、传感器语句块 | 12月第2周 |
| 交由应用测试 | 12月第2周 | 交由应用测试 | 12月第2周 |
| 所有完成的传感器的驱动程序 | 12月第4周 | 所有传感器的语句块 | 12月第4周 |

**五、方案展望**

产品同时优化了Tbot-I固件系统和IDE编程，计划在2019年12月前提交应用测试，提交的产品在编程上要比Arduino版本的更强，提供更多语句，更灵活的编程方式；在固件升级、程序下载方面做到更快捷；IDE交互方面更迎合用户使用体验；编程调试方面可以做到更加智能。