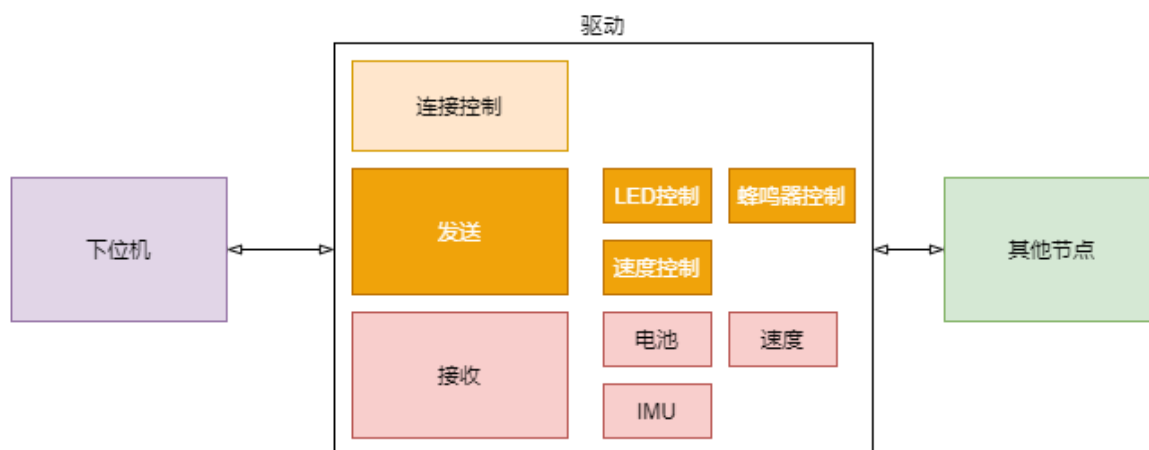


构建设计



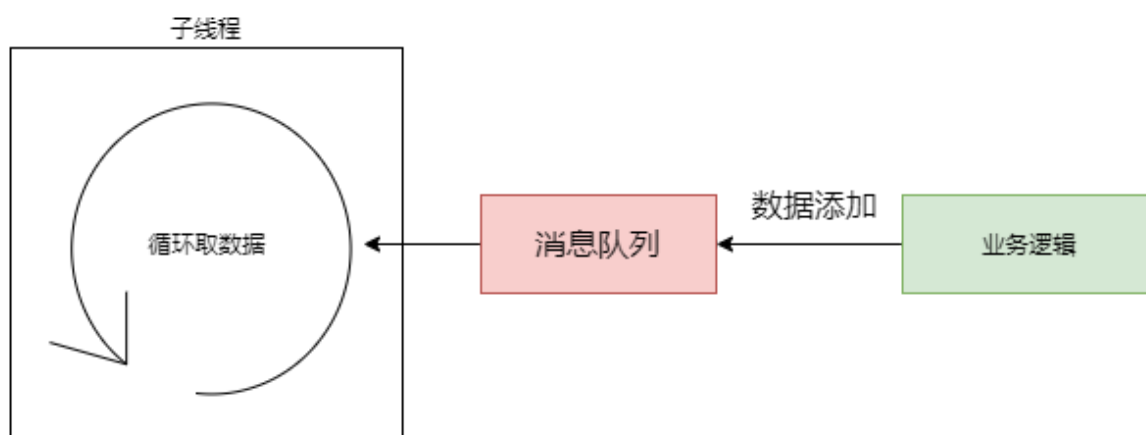
面向对象

将驱动面向对象化，从功能的角度而言，提供了以下功能：

- 连接下位机，断开下位机
- 给下位机发送数据
- 接收下位机发送的数据
- LED控制，蜂鸣器控制，速度控制
- 电池数据，速度数据，imu数据读取

消息队列

消息队列通常帮助我们来处理大量数据的，能让数据统一进行管理。



通过消息队列的使用分为两个部分：

- 一方负责放数据
- 一方负责取数据

python中，`Queue` 就是我们的消息队列。

- `queue.put`是放数据
- `queue.get`是取数据
- `queue.get`是一个阻塞式的函数，如果queue中没有数据，则会阻塞住代码

python中，通常用线程配合消息队列来使用，取数据会在子线程中，进行阻塞式读取。

回调设计

回调在开发中经常出现，在driver设计过程中，我们采用面向对象的操作，分为 `Driver` 类和节点。

两者要采取数据和事件交互时，可以通过回调来解决。

`Driver` 知道什么时候接收到数据，数据是什么。

节点想要把对应的数据发布出去。

`Driver` 负责定义回调函数，负责在适当的时机，调用回调函数。

节点则只需要实现回调函数即可。这样两者就能有效的进行数据传递。