Ewaybot\_Core 说明文档

版本控制

|  |  |  |
| --- | --- | --- |
| 修改日期 | 说明 | 修订人 |
| 2015-10-22 | 文档建立 | 史皓天 |

***类名***：Core

描述：管理各个模块进程，监听控制信号， 数据处理与传输

私有变量描述：

1. motion\_control 实例化运动控制对象

2. vision 实例化视觉对象

3. voice 实例化语音对象

4. ui 实例化UI控制对象

5. arm\_control 实例化手臂控制对象

6. active\_system 启动的控制系统

类型：int[5]

数据说明： 0-5 分别为 vison voice UI arms wheels

7. listen\_flag 信号端口监听指示位

类型：bool

数据说明：False，不监听数据；True，监听数据

8. num\_sources 控制源数目

类型：int

9. t\_listen 监听线程

10. t\_vision 视觉线程

11. t\_ui ui线程

12. t\_voice 语音线程

13. ser\_wheels 轮子驱动端口

14. ser\_arms 手臂驱动端口

…

外部接口描述：

1. system\_check() 检测机电系统状态

2. start\_all\_command\_system() 启动全部控制源线程

3. start\_local\_vision\_system() 启动本地视觉线程

4. start\_ui\_system() 启动UI线程

5. start\_voice\_system() 启动语音线程

6. start\_listen() 启动监听线程

7. start\_mechanical\_system() 启动机械系统

8. start\_mechanical\_arms() 启动手臂系统

9. start\_mechanical\_wheels() 启动轮子系统

10. kill\_all\_system() 关闭全部系统

11. kill\_ui\_system() 关闭ui系统

12. kill\_mechanical\_system() 关闭机械系统

13. kill\_voice\_system() 关闭语音系统

14. kill\_local\_vision\_system() 关闭视觉系统

15. get\_data\_old\_gui() 传输数据给本地GUI，目前废弃不用

16. stop\_listen() 停止监听信号源

功能示例：

1.启动系统

core = Core() #实例化对象

if core.system\_check: #确认机械系统无故障

core.start\_all\_command\_system #开启控制系统

core.start\_listen #开启监听模块

2.关闭某一系统

core = Core() #实例化对象

core.start\_all\_command\_system #开启控制系统

core.start\_listen #开启监听模块

…

…

core.kill\_local\_vision\_system #关闭本地视觉系统

…

…

3.停止核心监听线程

core = Core() #实例化对象

…

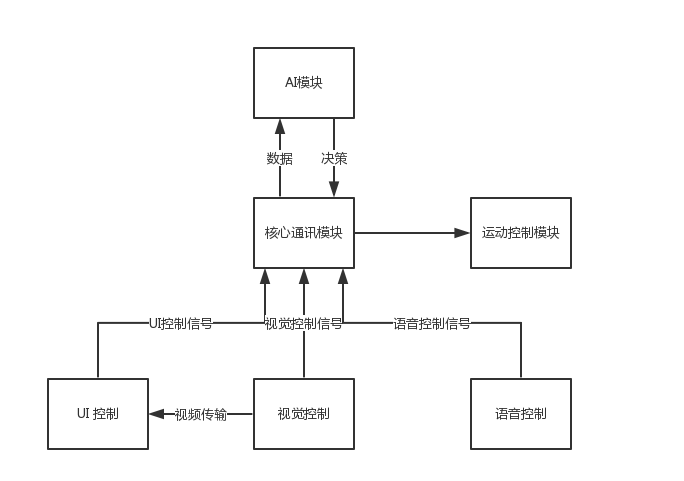
…

core.stop\_listen #关闭本地视觉系统

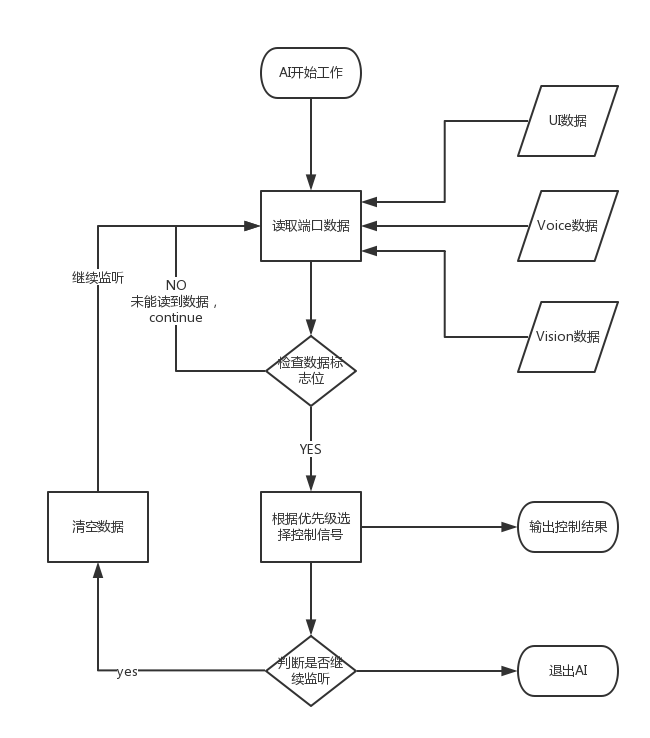
注意事项：

1. 初始化MotionControl 和 UI的端口信息，都在Core内部更改

2. 软件构架



3. AI 设计



具体函数定义：

***函数名***：system\_check()

描述：检查系统状态

输入参数定义：无

输出参数定义：全部系统正常，返回True反之，返回false；

调用实例：

***函数名***：start\_all\_command\_system()

描述：启动全部控制进程

输入参数定义：无

输出参数定义：无

调用实例：

***函数名***：start\_local\_vision\_system()

描述：启动本地视觉线程

输入参数定义：无

输出参数定义：无

调用实例：

***函数名***：start\_ui\_system()

描述：启动UI线程

输入参数定义：无

输出参数定义：无

调用实例：

***函数名***：start\_voice\_system()

描述：启动语音线程

输入参数定义：无

输出参数定义：无

调用实例：

***函数名***：start\_mechanical\_system()

描述：启动机械系统

输入参数定义：无

输出参数定义：无

调用实例：

***函数名***：start\_mechanical\_arms()

描述：启动手臂系统

输入参数定义：无

输出参数定义：无

调用实例：

***函数名***：start\_mechanical\_wheels()

描述：启动轮子系统

输入参数定义：无

输出参数定义：无

调用实例：

***函数名***：kill\_all\_system()

描述：关闭全部系统

输入参数定义：无

输出参数定义：无

调用实例：

***函数名***：kill\_ui\_system()

描述：关闭ui系统

输入参数定义：无

输出参数定义：无

调用实例：

***函数名***：kill\_mechanical\_system()

描述：关闭机械系统

输入参数定义：无

输出参数定义：无

调用实例：

***函数名***：kill\_voice\_system()

描述：关闭语音系统

输入参数定义：无

输出参数定义：无

调用实例：

***函数名***：kill\_local\_vision\_system()

描述：关闭本地视觉系统

输入参数定义：无

输出参数定义：无

调用实例：

***函数名***：stop\_listen()

描述：停止监听

输入参数定义：无

输出参数定义：无

调用实例：***函数名***：\_listen\_command()

描述：循环监听各个信号口数据，调用于start\_listen

输入参数定义：无

输出参数定义：无，信号经过处理后直接由motion\_control模块传到至下位机

调用实例：

t\_listen **=** threading.Thread(target**=**\_listen\_commend)

t\_listen.daemon **=** True

t\_listen.start()

***函数名***：\_simple\_decision(flag,data\_w,data\_a)

描述：根据优先级（UI>voice>vision）输出控制信号，调用于\_listen\_command

输入参数定义：flag(bool),控制位；data\_w(int(6)),轮子信号数据；data\_a(int(7)),手臂信号数据

输出参数定义：flag(bool),控制位；data\_w(int(6)),轮子信号数据；data\_a(int(7)),手臂信号数据

调用实例：

flag, data\_wheels, data\_arms **=** self.\_simple\_decision(flag, data\_wheels, data\_arms)