主要类别Controller,SmartMap,Interactor和Navigator。

Navigator用于管理智能小车的传感器数据获取和处理，同时对小车周围的环境进行检测并处罚异常。

Navigator包括霍尔传感器(SensorHall),加速度传感器(SensorAcc),陀螺仪传感器(SensorGyro),超声波传感器(SensorUltrasonic),二维码检测(QRCode)和磁场传感器(SensorMagnetic)。每一个传感器的设计如下：

例：超声波传感器包括SensorUltrasonic(主类)实现SensorUltrasonicIf接口,维护的数 据类为SensorUltrasonicData，与超声波有关的数据（包括传感器原始数据和处理后 的数据）。在SensorUltrasonic中我们会使用Timer定时获取传感器数据（注：一个 Timer对象占用一个线程）

这几个传感器类将作为Navigator类的成员类出现，他们之间的数据传输采用两种方式：监听者模式和主动函数调用。

监听者模式的Event和Listener为SensorEvent和SensorListener。SensorEvent中包含两个字段，type和data。type为整数类型，用于指明data的类型，data为Object,需要根据type还原为适当的数据类型，SensorListener中包含了事件处理函数。

在Navigator中，我们会维护x,y,vx,vy,ax,av,angular数据。

下面为各传感器的数据汇总处理：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 传感器类 | 采样频率 | 数据更新 | 用途 | 优先级 |
| SensorUltrasonic | 20-40Hz ? | 配合其他传感器，更新x,y | 疑似障碍物判断 | 1 |
| SensorHall | 10 Hz ? | 更新x,y,vx,vy |  | 2 |
| SensorAcc | 100 Hz ? | 更新vx,vy,ax,av | 控制行走速度 | 3 |
| SensorGyro | 100 Hz ? | 更新angular | 控制转向角度 | 3 |
| SensorMagnetic | 100 Hz ? | 更新angular |  | 2? 3? |
| QRCode | 1-2 Hz ? | 更新位置 |  | Infinite |

其中的优先级会用于数据的权值处理。

在Navigator中我们同样会有NavigatorEvent和NavigatorListener，event有检测到疑似障碍物等。

主动函数，用于Controller中的一些特殊策略，实现的功能有直接控制传感器数据的获取。

Interactor：

这个类用于人机交互，在项目的前期我们把此类的功能考虑完善，方便在后期我们添加手机或pc端的远程控制。

Interactor产生的命令通过相同的监听者机制以InteractorCMDEvent和InteractorCMDListener向Controller传递命令。目前我所想到的命令有如下：马达控制（前进，后退，行走），行驶到地图中的某个位置，获取数据等。

SmartMap:

此类用于维护地图信息，进行实时路径规划。

Controller：

这是我们项目中最核心的地方。在Controller中我会使用状态机进行控制，关于状态的设置目前尚未完善。

Controller中有三大块软件：InteractorCMDEvent，NavigatorEvent的处理程序和Controller中的例行循环语句（后面会讲清楚）。

InteractorCMDEvent的处理比较简单，直接响应即可。

NavigatorEvent的处理，目前只有障碍物的处理，处理策略尚未完善。

例行循环语句，可以理解为一个while(true)循环，这个循环目前做两件事情：

1、更新位置信息（从Navigator中获取），更新Motor的控制

2、查询关键点（QRCode），并检测