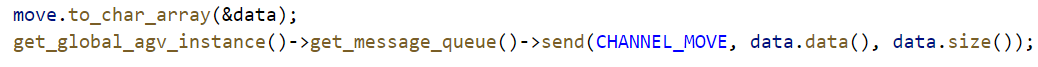
消息队列的接口定义在msg\_queue.h

消息队列的实例位于agv.h，类AGVInstance中

通过函数get\_global\_agv\_instance()->get\_message\_queue()获取全局消息队列指针

如：



## 发送消息的方法：

**消息队列支持的数据类型只有 char数组**

使用函数MsgQueue::send(int channel, char \* buf, const int size)

发送过程中会将消息内容复制到共享内存中，在send函数返回之后可以释放buf

频道的定义在文件msg\_channel\_index.h中

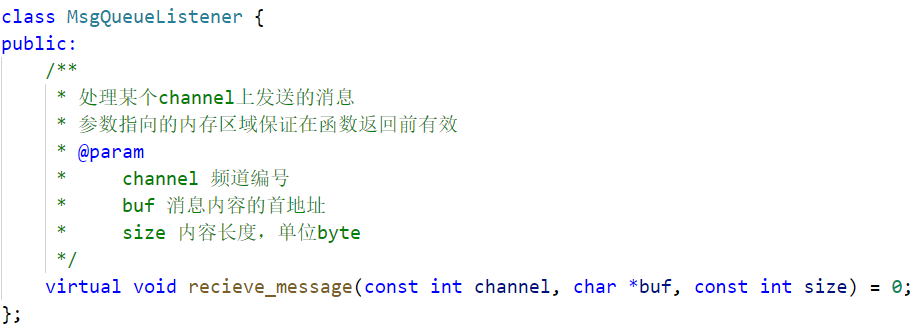


其中7，8，9，10暂未实现

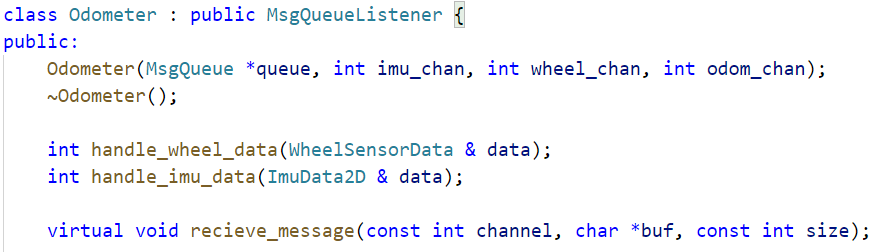
send函数不会等待消息处理结束再返回。send函数只负责为每个该频道的监听者启动一个处理消息的线程，不参与处理消息的过程。

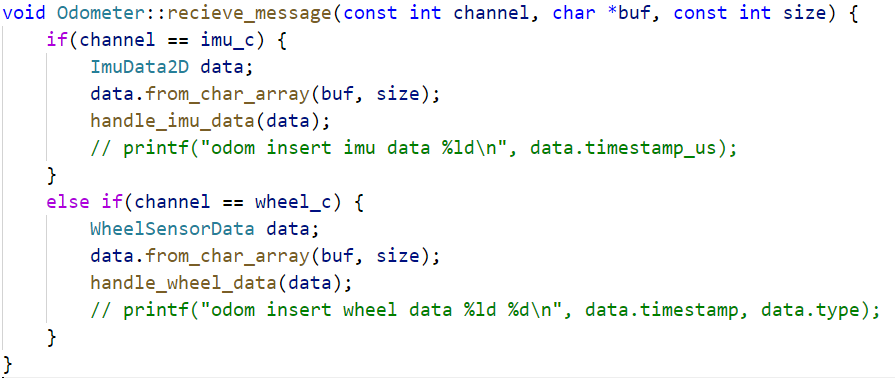
## 监听消息的方法：

将一个类继承类MsgQueueListener，并实现纯虚函数recieve\_message(const int channel, char \*buf, const int size)



如：





之后，可以使用函数

MsgQueue::add\_listener(const int channel, MsgQueueListener \* p\_listener)

将一个继承了类MsgQueueListener的指针绑定到某个频道上

函数add\_listener会返回一个整数，表示本次绑定的唯一id，可用于之后删除本次绑定

在监听完成后，使用函数MsgQueue::remove\_listener (const int listener\_id)

或MsgQueue::remove\_listener(const int channel, MsgQueueListener \* p\_listener)

删除绑定。