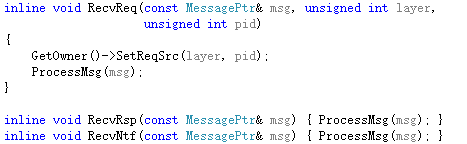
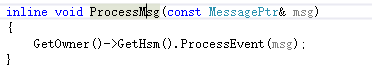
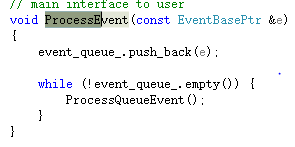
**Sap中**的处理函数，最终交给HSM中的函数

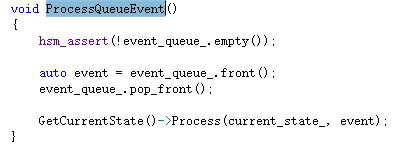


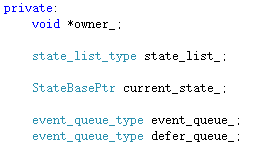


**状态机大类**中的函数：（插入队列中并一个个处理完）

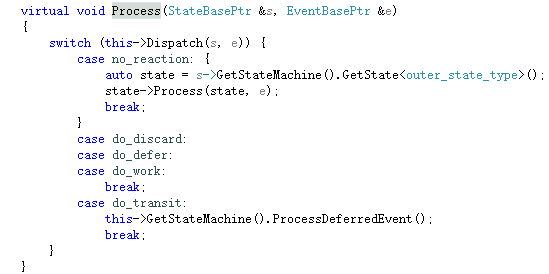


**状态机大类**中的处理队列函数：首先获得当前状态，然后调用该状态中的处理函数Process。（每个状态机都有一个私有成员，是当前的状态，找到当前状态就可以调用状态中的函数）

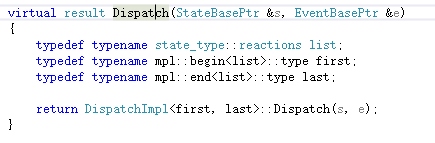




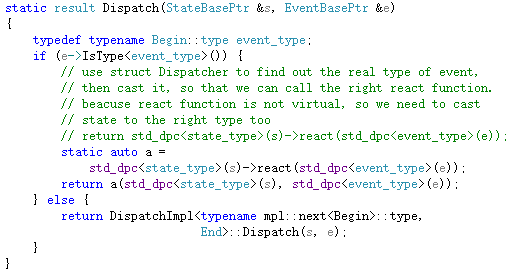
**状态中**的处理函数，之后交给Dispatch函数（父类StateImpl中的一个函数）



状态类父类（StateImpl）中的一个函数：

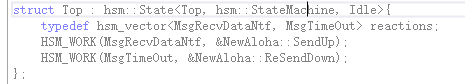


然后调用状态StateImpl中的，一个结构体名为DispatchImpl中的函数（与StateImpl中的Dispatch函数不同，只是同名而已）

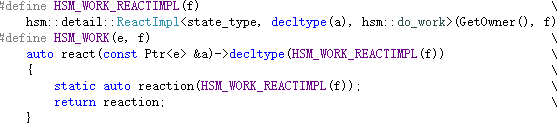


这里面，传入的状态s强转成当前类型，然后调用函数（react函数已经定义成了一些宏，协议栈中只需要用宏来写就行了。运行到这里的时候，编译器应该会自动找到匹配的react函数，然后调用函数）

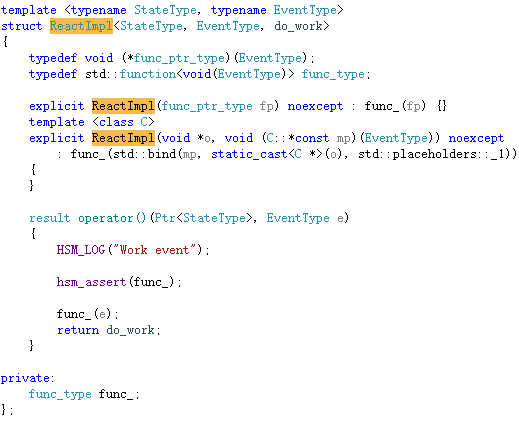
例如：（协议栈中）



HSM中的宏定义：（注意，这里的a表示传入的事件）



注意：这里通过react函数，返回的reaction是其中一种reactImpl结构体：（返回了这个结构体之后，再传入模板，这个模板应该是一种状态）



通过这个结构体重载的()运算符，在StateImpl中的一个结构体的Dispatch函数中，返回reactImpl，然后调用运算符重载，调用在协议栈中设置好的回调函数：（其中，传入的参数是s和e，也就是状态和事件）

