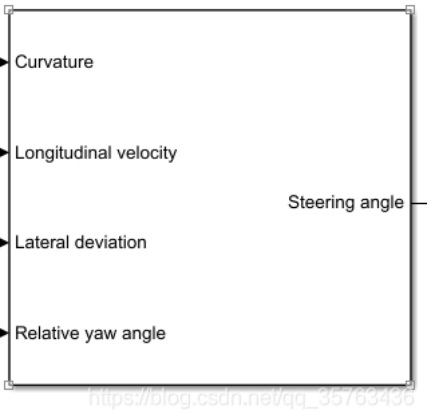
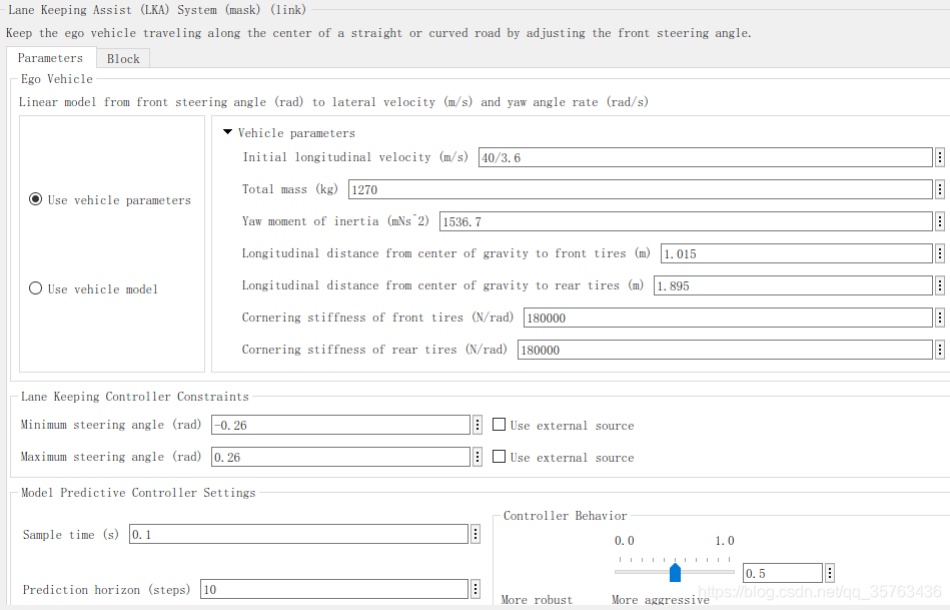


# 基于MPC的LKA算法设计

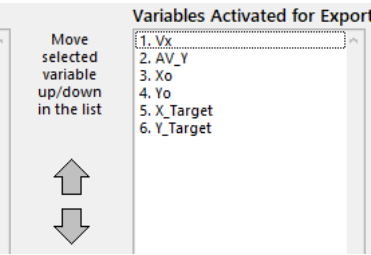
Matlab 2018版本中的simulink，有个自带的车道辅助保持模块，可以直接在帮助文档中搜索lane keeping assist system，如下，该模块是以MPC控制器为基础的，通过道路曲率、纵向速度、侧向路径偏移、横摆角误差作为控制输入，输出车轮转角信号。现在分享一下通过Carsim与simulink联合仿真出来的效果。



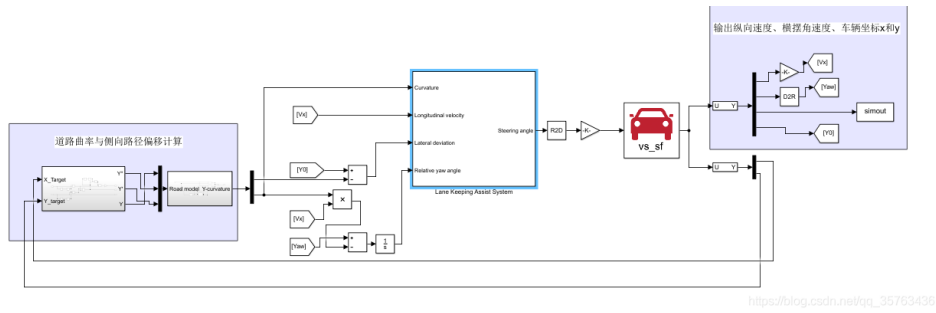
首先看一下这个模块里面的信息，给定车辆信息作为MPC预测模块的输入，车辆信息在Carsim中可以查到，给定转角约束，预测步长以及采样时间。右下角有个controller behavior的设置，范围在0-1之间，该值越大，控制器响应越快，但可能出现超调过大的情况，越小，控制器效果越平滑，这里取0.5。



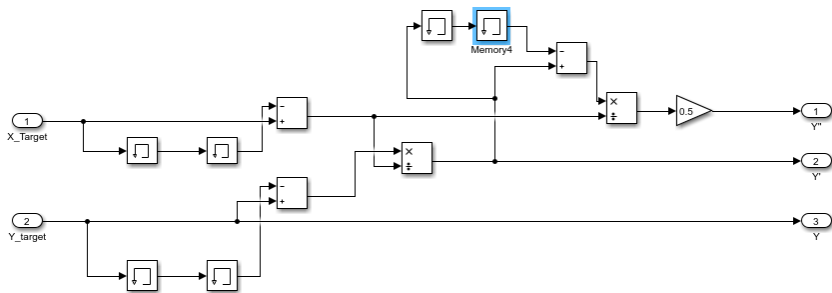
Carsim中的设置与上一篇LQR中类似，只在Carsim的输出模块中第一项更换为纵向速度



simulink中模型搭建如下：



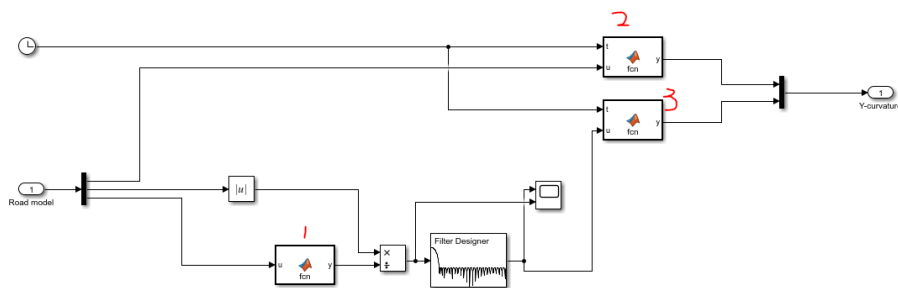
首先看一下右边carsim的输出，选择其中四个输出并将纵向速度和横摆角速度转换单位，这里车辆坐标x没有用到，我们先输出到工作空间。左边是道路曲率和侧向路径偏移计算模块，分别如下：



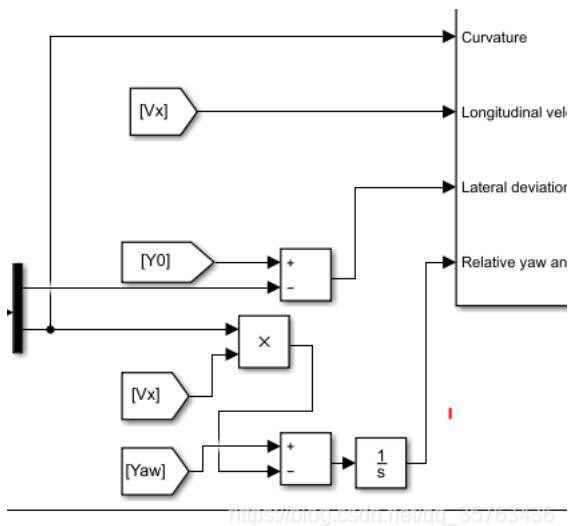
由于道路曲率的计算如下：

$$K = \frac{|y''|}{(1 + y'^2)^{3/2}}$$

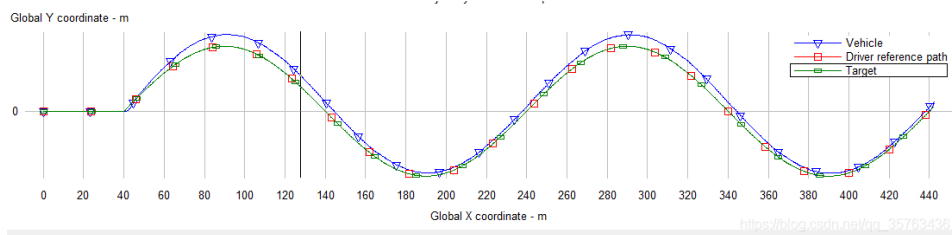
通过memory延时模块，利用导数定义的方式分别求取期望路径y的两阶和一阶导数。再通过如下模块计算出道路曲率，下面的fuction模块1即为上述曲率计算公式，function2和3是使得信号更加平滑。



通过车辆横摆角、车辆y坐标与期望的路径横摆角、期望y坐标做差，如下：得到控制器模块输入。



最后看一下carsim中的仿真效果：



车辆大致还是跟踪了期望路径，后续可以尝试在控制器设计中加入预瞄特性，以获得更好的跟踪效果。