

Control架构文档

架构图与概要

以静态逻辑图为主，添加关键数据接口信息；对关键信号和子模块辅以文字解释

概要介绍

该模块在智驾系统中的主要作用

Control作为MAF系统的子模块，主要作用是在多种场景下完成自车的闭环运动控制，如Cruise Pilot（CP），Auto Parking Assist（APA），Highway（HNP）等。

在自动驾驶状态下，control根据planning下发的reference，定位模块输出的ego_pose，自车的反馈信息chassis_report，wheel_report等信息，通过闭环控制的方法使得自车尽可能快速准确的跟踪reference，并且尽可能抵消过程中扰动。

核心算法

描述该模块的核心算法

Control模块从planning获取到reference信息，经过control preprocess的预处理之后转化为自车横纵向控制的参考输入。

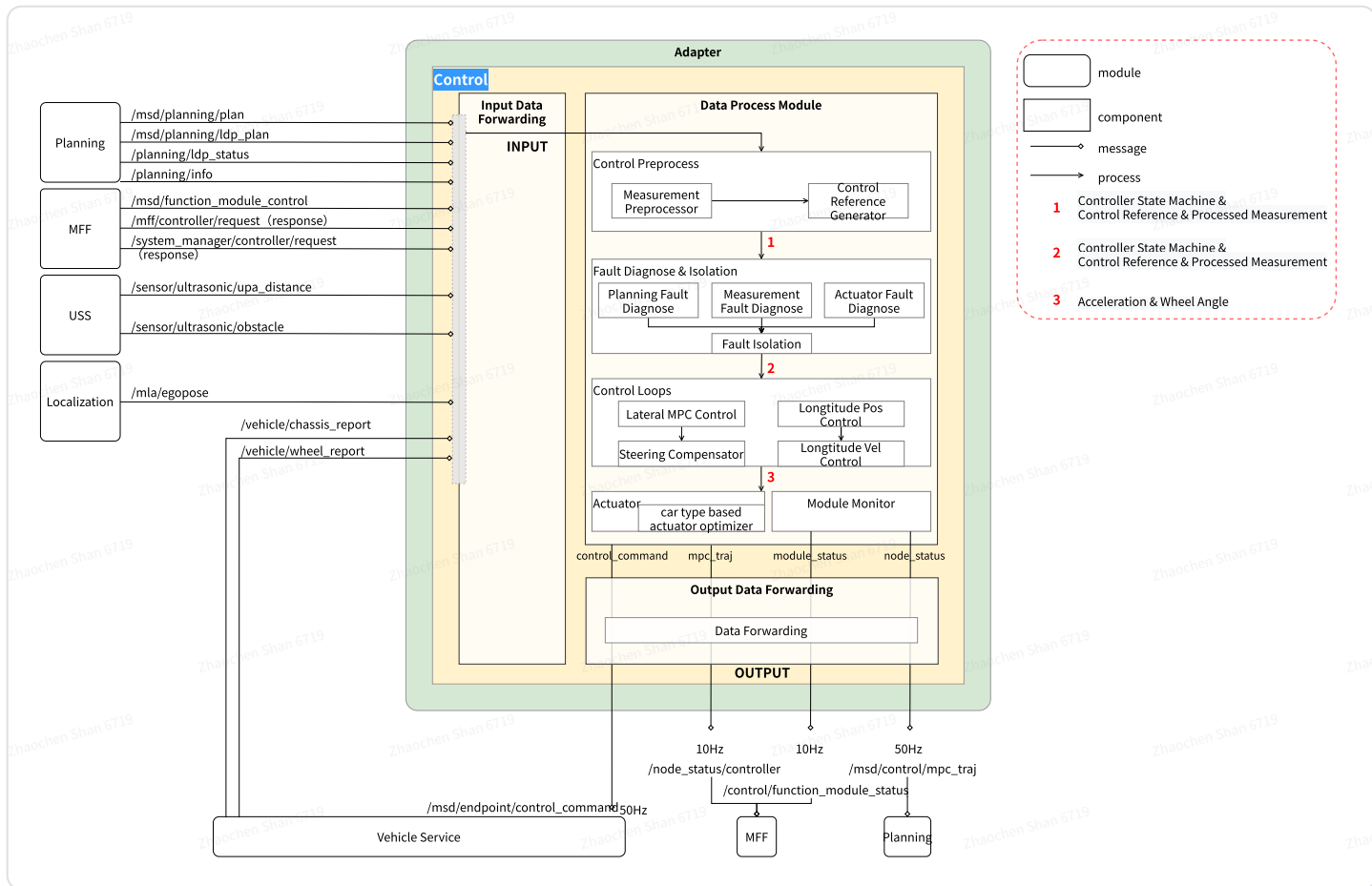
control loop接收预处理得到的轨迹和自车反馈信息进行闭环运动控制。

执行器模块会根据不同的车型将control loop输出的控制指令转化为自车执行器可以直接执行的指令。

Fault diagnose & isolation负责分析上下游的指令以及反馈信息，进行故障诊断与隔离。

静态逻辑图

描述主要子模块的静态逻辑关系



动态过程图

描述模块内部的子模块间的动态交互关系

Control模块的启动

Control模块接受mf_script的统一调度进行启动，启动过程如下。

mf_script (repository)

start master
(start_mfrmaster.sh)

start all module
(start_all.sh)


start sessions
(sessions.sh)

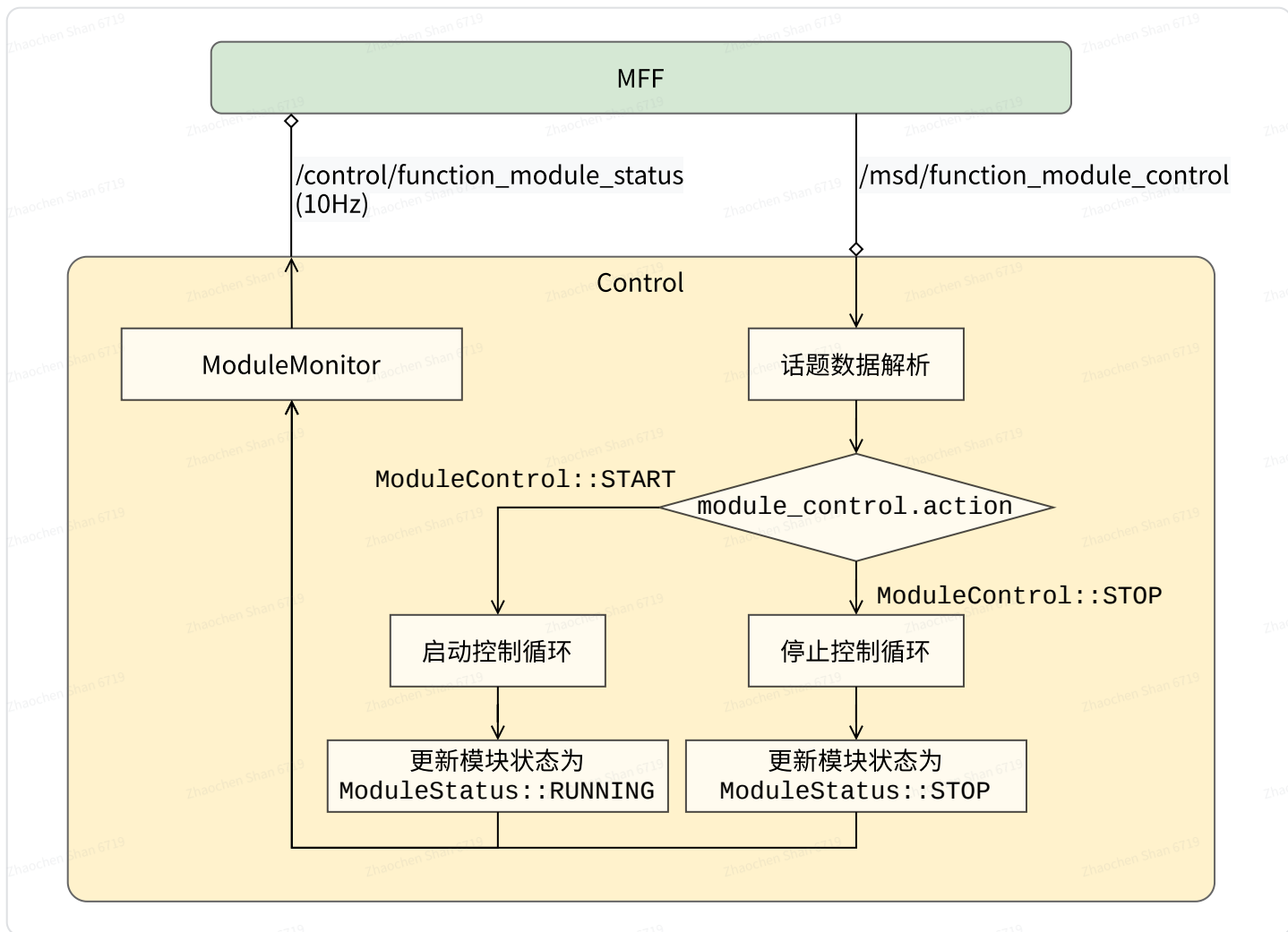
tmuxp load maf.yaml
(sessions.sh)


maf_controller (repository)

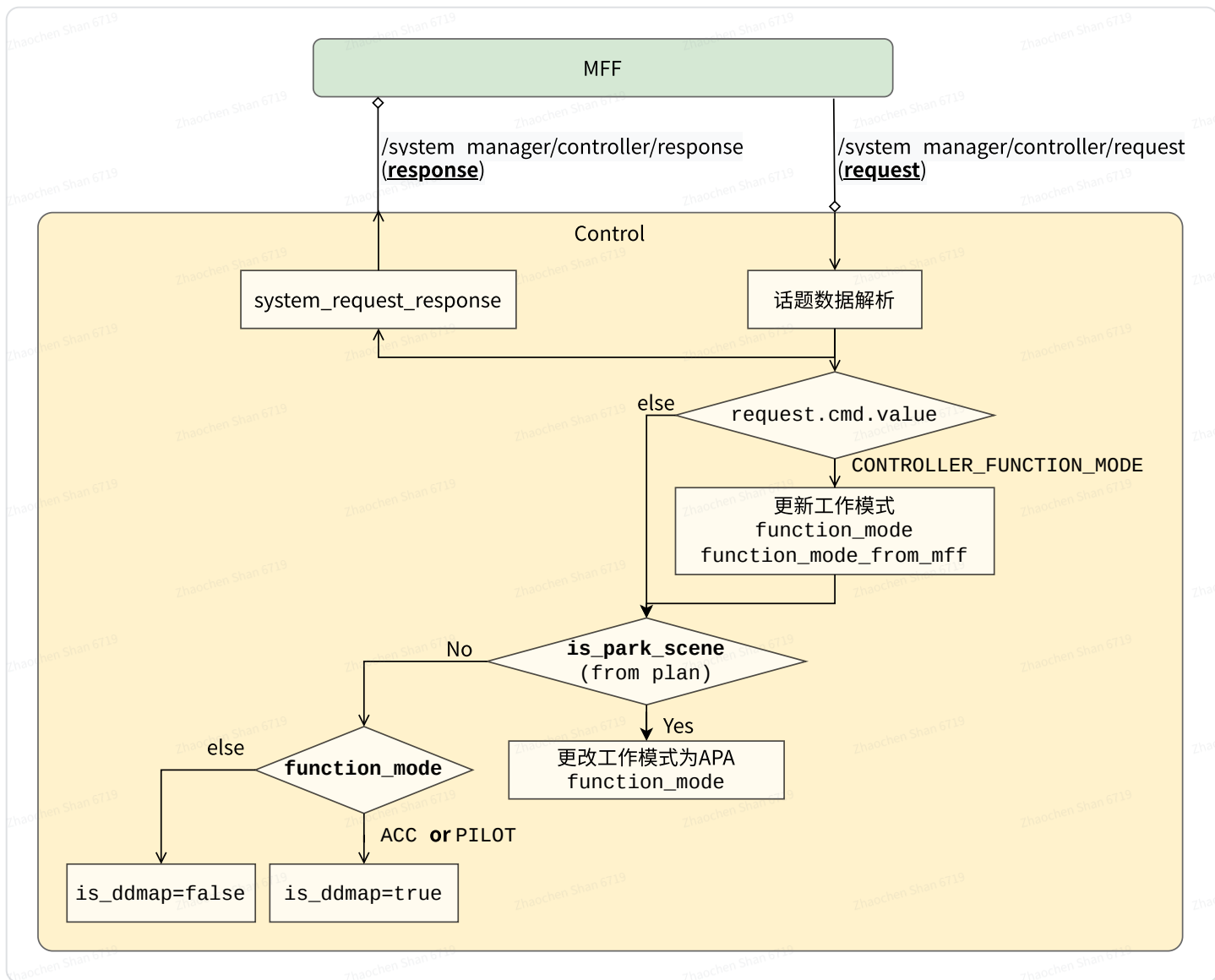
Adapter
(startup.sh)

MFF对Control模块的控制

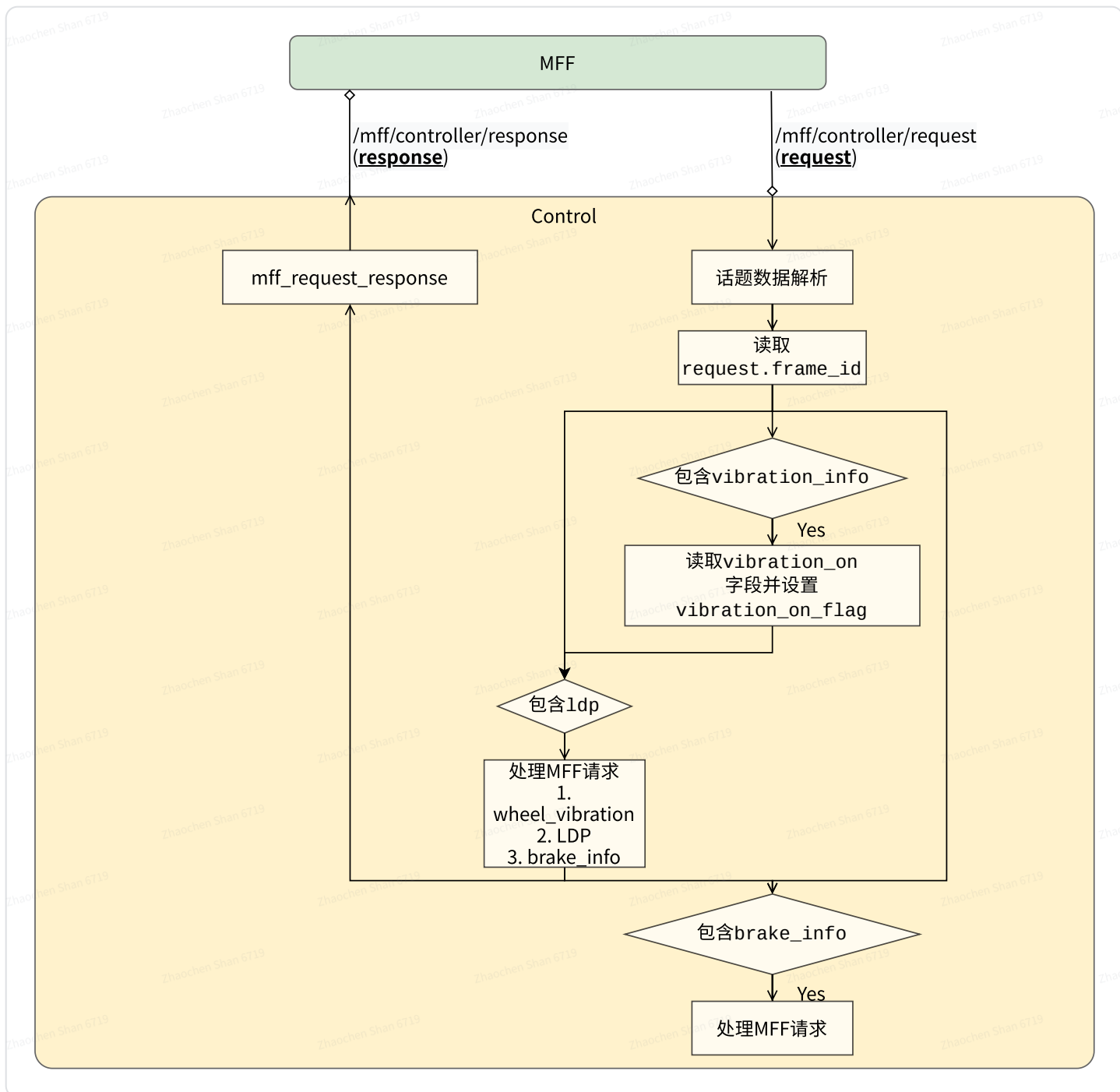
- Control控制回路的开始和停止  Hengbing Wang



- 控制Control模块功能的切换  Hengbing Wang

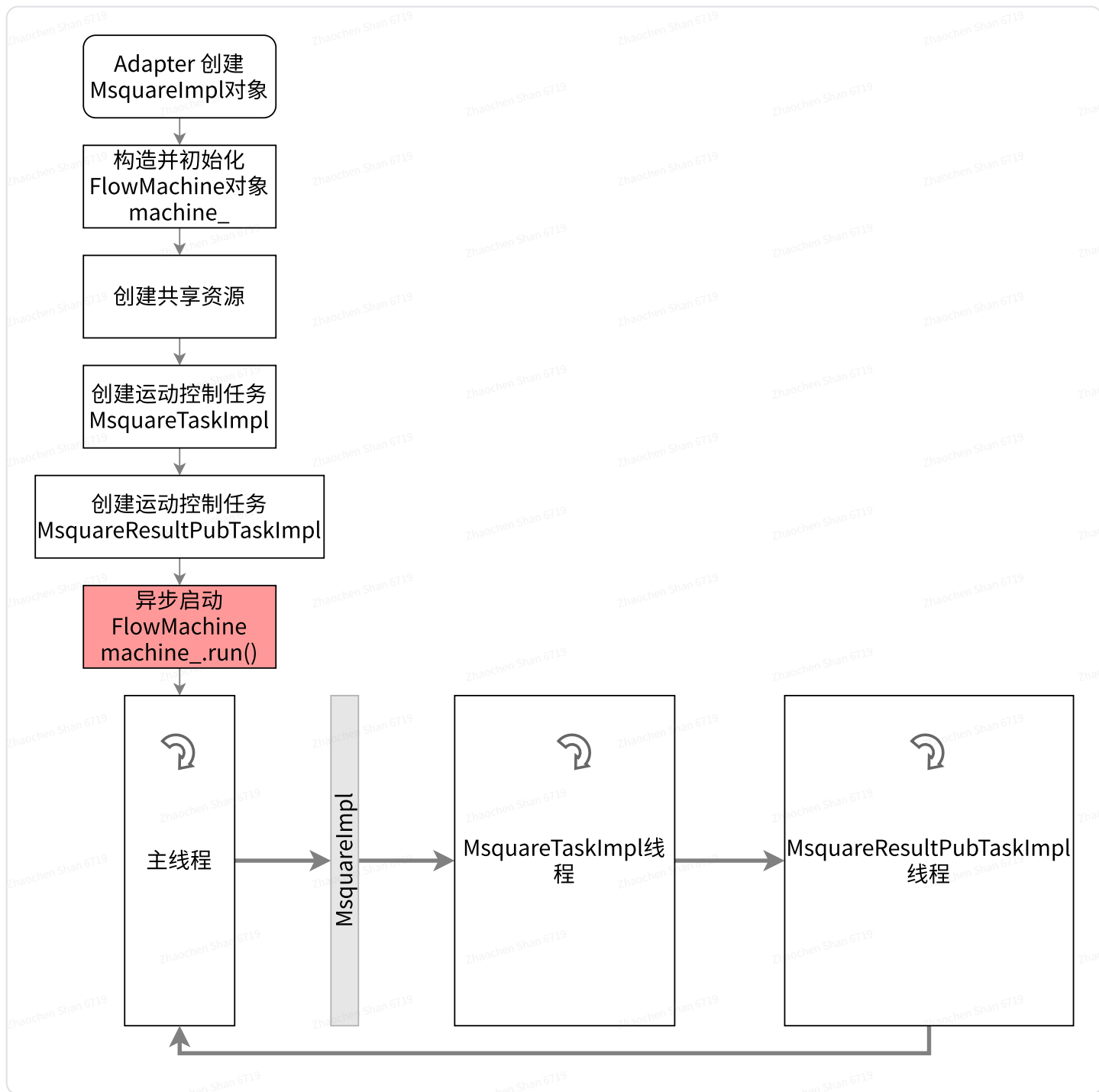


- MFF发送的json字符串指令

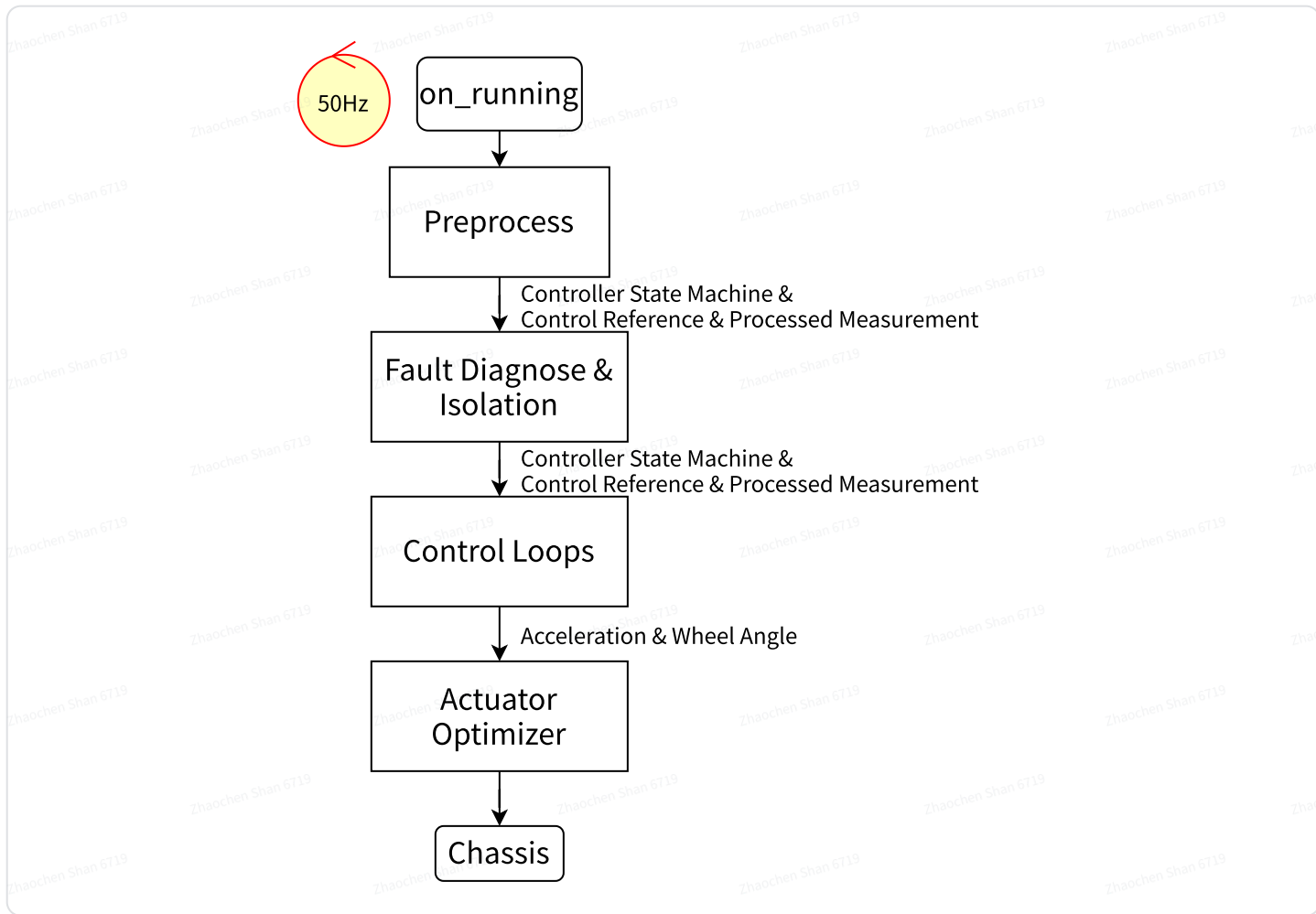


FlowMachine的运行

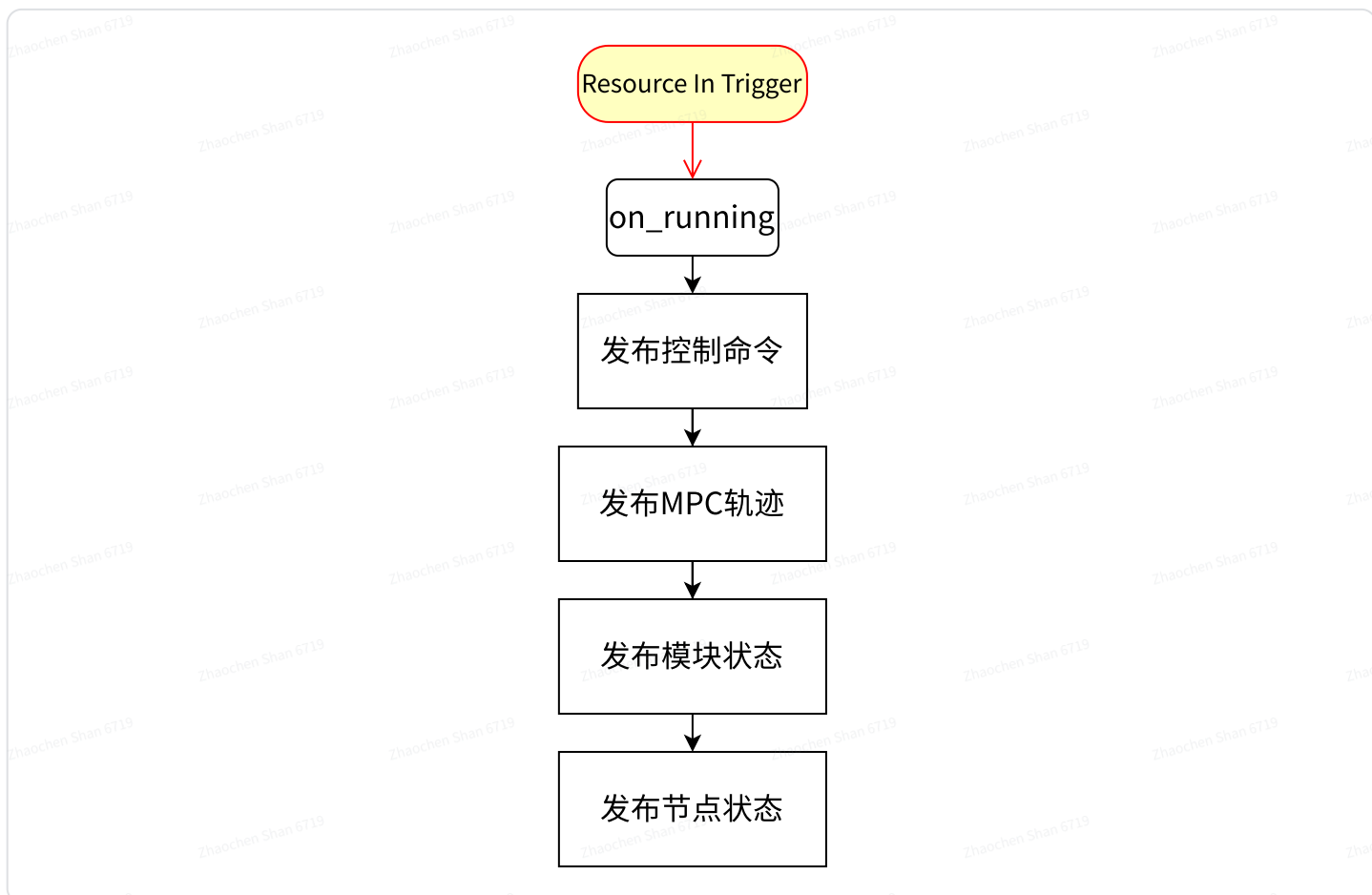
Control总体上有三个线程。主线程负责与外界交互，获取规划、测量、指令等信息；控制任务线程用于完成主要的控制逻辑；转发线程负责将控制指令以及模块状态信息等发到主线程再由主线程发布到外界。



运动控制任务



数据转发任务



名词解释（尽量收录词条百科）

建议对于文中出现的重要名字缩写加以注释，特别是百科中未收录的词条

名字	解释
MPC	Model Predictive Control，模型预测控制

功能需求

这个模块实现了哪些核心功能

功能	描述
Planning数据预处理	将Planning模块下发的规划轨迹转化为控制器的参考输入
测量数据预处理	对测量数据进行滤波，坐标系变换等处理
故障诊断与隔离	对规划、测量数据以及执行器输出结果进行诊断，对数据进行安全约束
横纵向闭环控制	横向通过MPC实现闭环控制，纵向通过位置及速度控制器进行经典闭环控制
执行器适配	将横纵向控制器的输出转化为自车可以直接执行的指令
状态监测	进行节点状态以及模块状态的监控并向外界进行反馈
在线调参	修改Control控制参数配置文件，重新进自动相关配置参数将重新加载

非功能需求

建议包括如下几部分：

- 1. 模块非功能设计需求，如资源占用，耗时，可靠性等
- 2. 安全需求与实现
- 3. 可以补充一些其它信息，如代码设计文档，repo地址，测试方法等

性能需求 Wanli Jiang

设计需求	描述	当前实现
MPC耗时	MPC完成一次计算的时间	< 2ms

纵向误差	自车行驶过程中对期望轨迹的纵向跟踪性能	< 1m
横向误差	自车行驶过程中对期望轨迹的横向跟踪性能	转弯场景0.35m 直行及变道场景0.25m APA 0.1m
横摆角误差	自车行驶过程中横摆角的跟踪性能	CP 4° APA 5°
速度跟踪误差	自车行驶过程中纵向速度跟踪性能	< 1.0m/s
control模块耗时	控制模块单周期耗时（期望值20ms）	17 ~ 23ms
APA剩余距离误差	APA过程终态剩余距离（期望值0m）	-0.1 ~ 0.1m

安全设计/实现

注：该模块如果实现故障错误处理。如果有更详细的设计文档，也可以链接在这

故障诊断与隔离功能实现了Control的安全性设计，详细内容参考 [02-Fault diagnose & isolation算法模块](#)

接口与配置

建议写明模块的关键输入输出、帧率，以及与上下游的联动关系，例如消息定义、通信topic、节点等；建议在源码文件中维护（如manifest中sub/pub的doxygen文档），并在这附上代码地址链接

接口

类型	接口	描述	信号健康状态
输入	EgoPose	提供自车定位信息，Control模块根据该信息进行坐标系变换，构造控制器参考输入	<ul style="list-style-type: none">期望帧率50Hz需始终包含有效 position_local
输入	ChassisReport	提供自车的刹车、油门、档位以及方向盘转角等信息	<ul style="list-style-type: none">期望帧率50Hz
输入	WheelReport	提供自车的轮子位置、速度以及IMU等信息	<ul style="list-style-type: none">期望帧率50Hz
输入	UltrasonicUpaReport	提供自车超声测距信息 (计算泊车过程中的最大剩余距离)	<ul style="list-style-type: none">期望帧率30Hz
输入	UltrasonicObstacle Report	提车自车超声检测到的障碍物信息 (计算泊车过程中的最大剩余距离)	<ul style="list-style-type: none">期望帧率20Hz

输入	Planning	提供自车的期望运动轨迹	<ul style="list-style-type: none">期望帧率10Hz
输入	ModuleControl	控制Control模块的启动和停止	-
输入	SysControllerRequest	切换Control模块的功能模式 <ul style="list-style-type: none">CPAPALPNPACC	请求-应答模式
输入	Header	MFF向Control模块发送请求（wheel vibration） <ul style="list-style-type: none">wheel vibrationLDPbrake info	请求-应答模式
输出	ControlCommand	自行车控制指令，包括油门、刹车、档位、车灯等	<ul style="list-style-type: none">期望帧率50Hz
输出	MpcTrajectoryResult	MPC轨迹 在一个控制周期内MPC控制器输出的自行车位置及横摆角的序列	<ul style="list-style-type: none">期望帧率50Hz
输出	NodesStatus	输出节点状态，包括节点状态字、节点类型、任务类型等信息	<ul style="list-style-type: none">期望帧率10Hz
输出	ModuleStatus	输出模块状态，包括模块类型、模块状态、详细状态等信息	<ul style="list-style-type: none">期望帧率10Hz
输出	Header	回复MFF模块的请求（Header）	请求-应答模式
输出	SysControllerResponse	回复SysControllerRequest的请求	请求-应答模式

配置

写明模块的关键配置项和行为影响；建议在源码文件中维护（如doxygen文档），并在这附上代码地址链接

配置项	描述
任务流配置	Control内部构建了两个子任务，一个子任务负责Control的主逻辑，另一个任务负责完成消息的转发。任务流配置文件（ mtaskflow_config.json ）用于配置各个任务的线程id，任务名称等。

车辆及控制算法参数	Control为不同的车型建立了不同的配置文件，配置文件中存储自车的结构信息，控制算法所需的各种参数等，如EP33对应的配置文件为 controller.msquare.ep33.210520.json
vehicle.yaml	根据文件中的type和architecture字段判断车型并加载相应的control配置文件， vehicle.yaml

重要子模块

针对架构中关键子模块，解释其主要功能，附上代码链接

	功能	描述	代码链接
1	预处理模块	负责处理planning的参考轨迹，整合各类反馈信息，生成控制回路所需要的参考输入、状态机等 01-Control preprocess算法模块	Repo链接
2	诊断与隔离模块	对测量信息，参考轨迹以及执行器状态进行诊断，针对不同的故障类型进行隔离处理 02-Fault diagnose & isolation算法模块	Repo链接
3	闭环控制模块	根据参考输入和测量信息进行车辆横纵向控制，输出响应的控制指令 03-Control loop算法模块	Repo链接
4	执行器适配模块	将横纵向控制器的输出转化为自车可以直接执行的控制指令	Repo链接

其他详细设计材料

注: 链接至模块的其他详细设计材料（如算法组自己的飞书文档链接）