系统数据流和任务调度

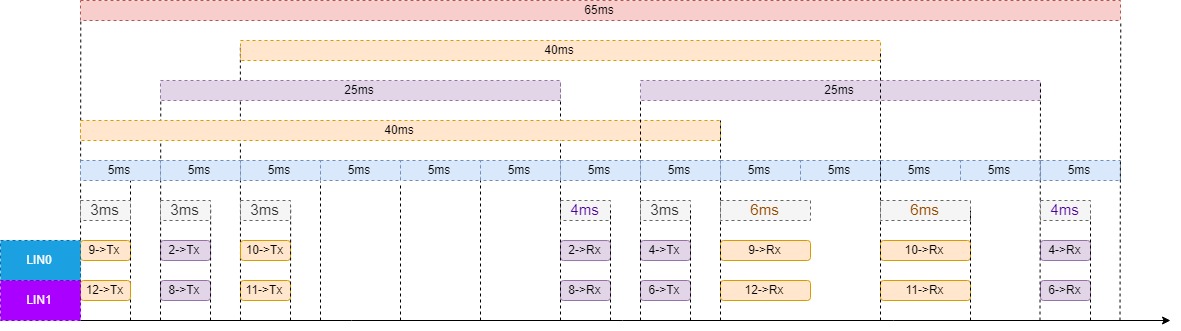
# 数据流

## 超声波信息

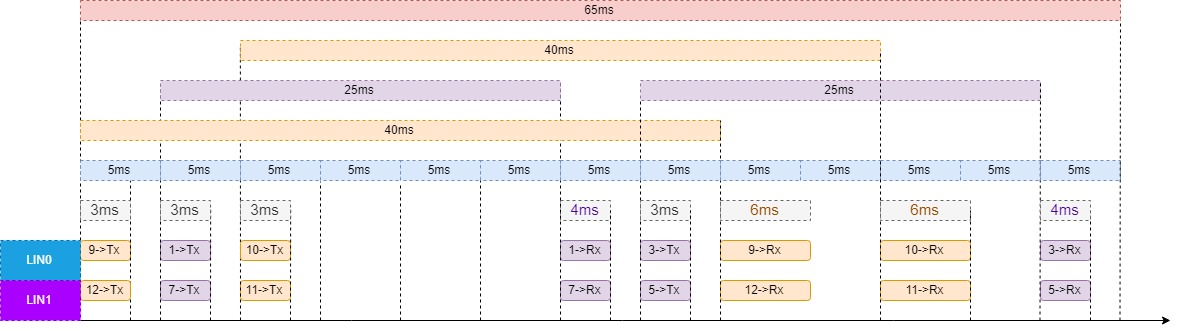
如图一、图二所示，长距超声波的更新周期为65ms，短距超声波的更新周期为130ms。数据更新通过内部的DMA模块自动搬运到内部指定的RAM区域。在此过程中，几乎不占用CPU的时间，从而可以让有限的CPU资源更多的用在算法计算和实时控制上。

在车位检测阶段，检测算法的数据获取周期可以设定为65ms，这样可以保证每次获取的数据都是最新值。

在超声波避障阶段，避障算法的数据更新周期可以设为130ms，由于8个短距超声波的数据并不同步更新，因此每次激发时都会记录相应的系统时间戳，在实际使用时可以通过系统时间进行相应的距离补偿。



图一



图二

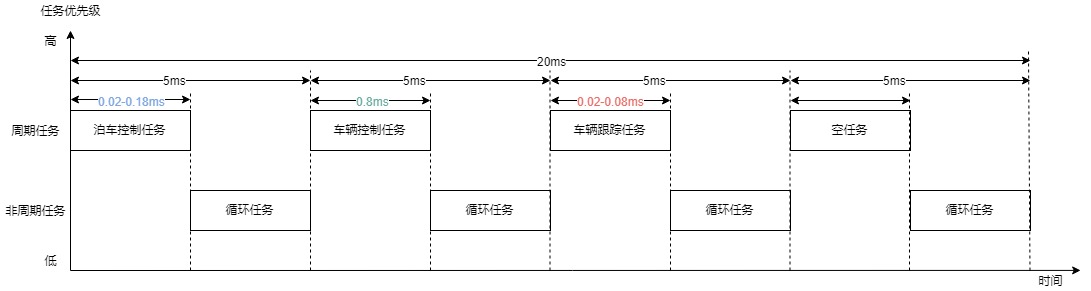
## 车身信息

车身信息的更新周期为20ms，嵌入式平台通过CAN接收中断及时获取车身信息。同时CAN接收模块开启ID滤波器，可以有效滤除无效的帧ID，提升CAN总线的接收效率。

# 任务调度

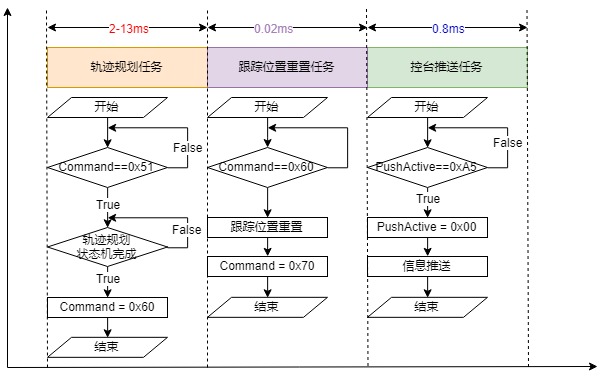
## 总体任务时间分配

如下图所示，系统任务以20ms为总执行周期，在该周期内分为4个时间段，每个时间段各自执行相应的子任务。泊车控制任务根据不同控制阶段，大概需要0.02-0.18ms的执行时间；车辆控制任务需要0.8ms的执行时间；车辆位置跟踪任务需要0.02-0.08ms的执行时间。其余时间执行循环任务，循环任务内的任务都是对时间不敏感的任务。



## 循环任务

如下图所示，在循环任务内部分配了轨迹规划、跟踪位置重置和控制台推送任务。轨迹规划任务会根据库位状态的不同，大概需要2-13ms的规划时间，跟踪位置重置任务需要0.02ms的执行时间，控制台推送任务需要0.8ms的执行时间。



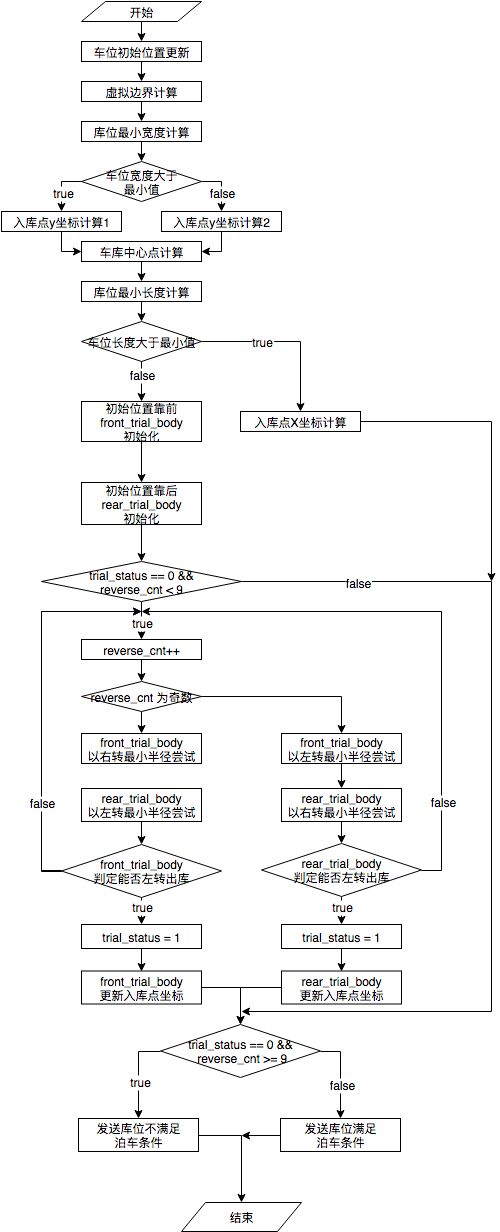
### 轨迹规划任务

轨迹规划任务在收到控制台发送的0x51命令后，开始进行相应的轨迹规划，规划成功后，将Command命令设置为0x60，从而激活跟踪位置重置任务。如果规划失败，说明车位不满足泊车条件，车辆需重新寻找车位。

轨迹规划任务可以分为反向尝试出库阶段、入库曲线规划阶段和转向点计算阶段。经过这三个阶段的规划，最终可以计算出车辆入库点坐标和行驶过程中的转向点坐标和对应的转向角度。

#### 反向尝试出库阶段

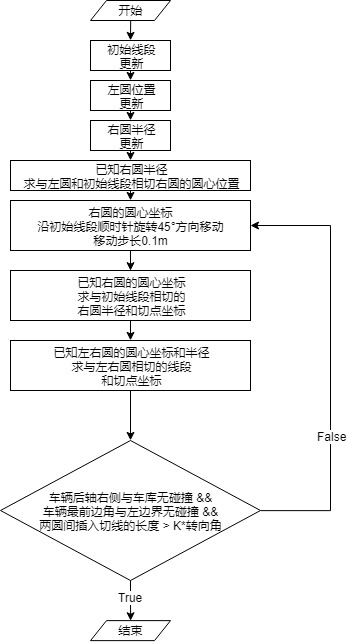
如下图所示，是车辆反向尝试出库阶段的流程图。首先根据设定的车位余量，进行车位虚拟边界的计算。接下来就是根据实际车身状态，计算理论的最小车位宽度和长度。如果实际车位宽度大于理论计算的最小宽度，则可以确定入库点y方向的坐标，使得车辆与车位左边界对齐。如果实际车位宽度小于最小宽度，则车辆入库点需向左移动，移动距离依据实际车位宽度与最小车位宽度的差值，最终确定入库点的y方向坐标。入库点y方向确定后，就可以进行泊车中心点的计算。根据最小车位长度，如果实际车位长度大于理论最小车位长度，则可直接确定入库点的x轴方向的坐标 。如果实际车位长度小于理论最小车位长度，则分别初始化前向尝试和后向尝试对象，如果当前尝试次数为奇数，判断前向尝试对象是否可以出库，如果尝试次数为偶数，判断后向尝试对象是否可以出库。如果某次尝试对象可以出库，则置位状态变量并将该尝试对象的状态更新到入库点坐标中。最后如果尝试状态变量为0并且尝试次数大于等于9，则说明车位不满足泊车条件，否则满足泊车条件。



#### 入库曲线规划阶段

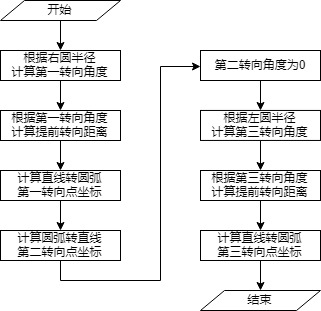
该阶段主要根据车辆初始位置和入库点坐标规划一条曲线，使得车辆可以从初始位置到达入库点。首先根据初始位置更新初始线段信息，然后根据入库点坐标更新左圆位置信息和右圆半径信息。

接下来在右圆半径已知的情况下，移动右圆使得与左圆和初始线段相切，确定右圆心的圆心坐标。右圆圆心沿初始线段顺时针旋转45°方向以0.1m的步长移动。此时右圆圆心坐标已知，求与初始线段相切圆的半径和切点坐标。当左右圆信息确定后，求与左右圆相切的直线和相应的切点坐标。最后进行相关条件的判断，条件满足后则结束规划，否则继续移动圆心坐标，直到条件全都满足。



#### 转向点计算阶段

如下图所示，是转向点计算的流程图。由于入库曲线阶段已经算出各个切点得坐标，根据相切圆的曲率可以计算出提前转向点的坐标。



### 跟踪位置重置任务

跟踪位置重置任务在轨迹规划任务成功后进行，主要根据车位检测后的停车点坐标，重新更新车辆坐标系信息。

### 控制台信息推送任务

控制台推送任务每隔5ms会被激活一次，然后会发送相应时间段的推送信息。推送的信息包括超声波信息、车辆实时跟踪信息、车辆控制信息、车辆状态信息和泊车状态信息。如下图所示，是控制台信息推送任务的执行流程。从图中可以看出，不同的时间段会向控制台推送不同的信息。

