



---

编号:

## 智能驾驶开放平台 API

# 执行机构逻辑说明

比亚迪汽车工业有限公司

2019 年 01 月 03 日



文件修改履历表

版本号	修改内容描述	生效日期	修改人
N/0	逻辑说明初版	2019-1-13	高上添



## 目 录

1	引言 .....	4
1.1	编写目的 .....	4
1.2	执行机构分类说明 .....	4
2	ACU 的 CAN 通信说明 .....	5
2.1	CAN 发送说明 .....	5
2.2	CAN 接收说明 .....	6
3	秦 Pro 执行机构逻辑说明 .....	7
3.1	组合开关逻辑说明 .....	7
3.2	电子手刹逻辑说明 .....	7
3.3	档位控制器逻辑说明 .....	8
3.4	电机控制器逻辑说明 .....	8
3.5	EPS 逻辑说明 .....	8
3.6	ESP 逻辑说明 .....	9
3.7	ACU 智能驾驶请求、退出说明 .....	9
3.8	ACU 异常退出情况 .....	9



## 1 引言

### 1.1 编写目的

比亚迪智能驾驶开发者平台的 SDK 开发包是基于秦 Pro 平台的，所以有必要对秦 Pro 的相关的智能驾驶模块进行逻辑说明，可以结合 SDK 中的应用例程，使开发者更好的理解从而进行智能驾驶开发。

### 1.2 执行机构分类说明

开发者要实现秦 Pro 智能驾驶就必须让 6 大模块（电机控制器、ESP、档位控制器、电子手刹、组合开关、EPS）进入智能驾驶模式当中，为了兼容秦 EV 的 API 版本，API 库中将 EPS 请求进入智能驾驶的方式与电机控制器、档位控制器、电子手刹、组合开关的请求方式划分到一类中，而 ESP 仍然继续保持单独的请求进入智能驾驶的方式。后续的内容将先对各个执行器的细节逻辑进行描述以此让用户了解在开发过程中出现的正常情况；而后从整体控制请求进行逻辑说明，开发者可以结合用户例程中的 main.c、api\_test.c、api\_test\_autodrive.c 请求进入自动驾驶和退出智能驾驶的逻辑，当然开发者可以增加或者改进自己的判断逻辑使得整车控制更加合理。

## 2 ACU 的 CAN 通信说明

ACU 开发者要实现 API 的设置和获取必须先实现 ACU 端同安全网关之间的通信，比亚迪智能驾驶开发者平台 API 的 CAN 的发送和接收依赖于 ACU 的 CAN 设备，所以要靠 ACU 开发者自己编程，如下图 Receive 虚线框和 Transmit 虚线框，其中 Receive 可以通过线程来实现；而 Transmit 发送报文周期比较严格，最好通过稳定的定时器来实现（可以参考《智能驾驶开发者平台 API 用户手册》3.2 章节）。

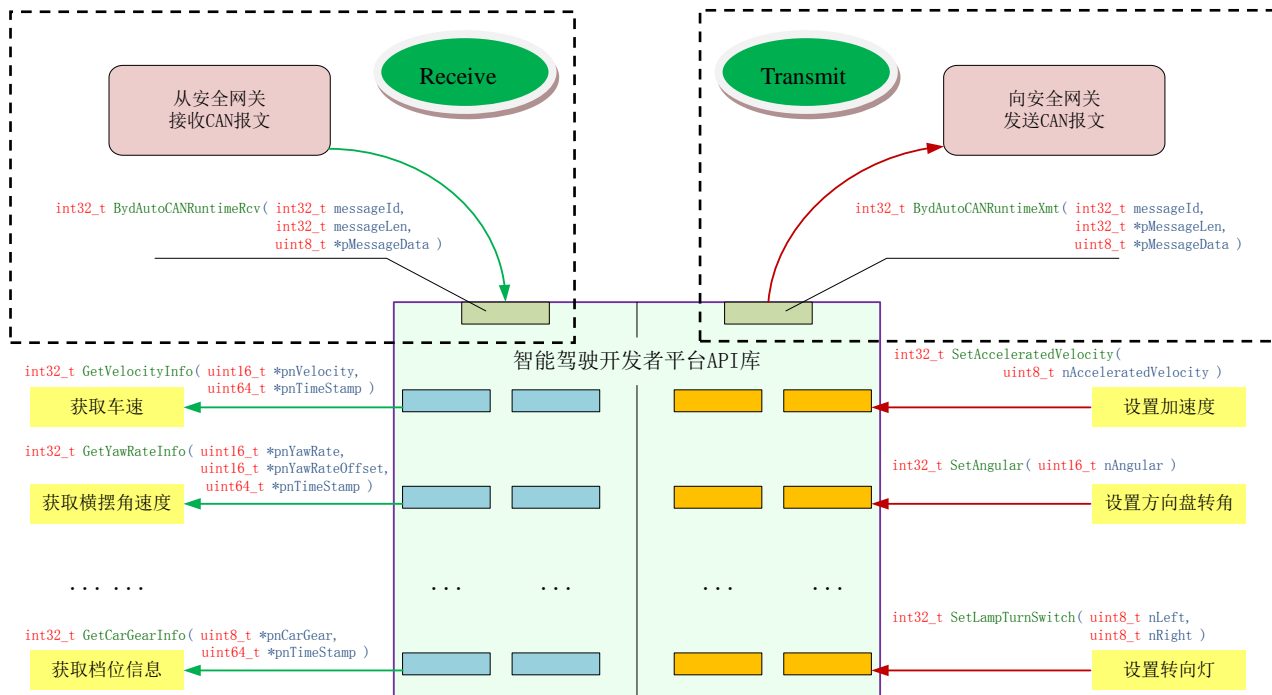


图 1 API 及 CAN 通信

### 2.1 CAN 发送说明

ACU 的 CAN 发送是实现 ACU、安全网关和整车通信的关键部分，所以 ACU 开发者要设计好自己的 CAN 发送的功能部分，在 SDK\Developer\_Platform\_Release\include\下有个 idriving\_api.h 文件，其中定义了几个报文 ID，如下：

```
#define BYD_AUTO_IDRIVING_CMD_0 (0x12F82340u) // 周期, 10ms
#define BYD_AUTO_IDRIVING_CMD_1 (0x12F83A30u) // 周期, 20ms
#define BYD_AUTO_IDRIVING_CMD_2 (0x12F832E0u) // 周期, 20ms
// #define BYD_AUTO_IDRIVING_CMD_3 (0x12F832D0u) // 周期, 20ms
// #define BYD_AUTO_IDRIVING_CMD_4 (0x12F832F0u) // 周期, 20ms
#define BYD_AUTO_IDRIVING_CMD_5 (0x12F83380u) // 事件周期, 100ms
```

目前 BYD\_AUTO\_IDRIVING\_CMD\_3 和 BYD\_AUTO\_IDRIVING\_CMD\_4 暂未使用，可以屏蔽以此减少整车网络的负载；同时 ACU 开发者必须要满足上述 ID 的发送周期要求，误差保持在 20% 以内，如 BYD\_AUTO\_IDRIVING\_CMD\_0 报文周期保持在 8~12ms 之内，而 BYD\_AUTO\_IDRIVING\_CMD\_1 和 BYD\_AUTO\_IDRIVING\_CMD\_2 报文要保持在 16~24ms 之内，否则无法激活安全网关，接收不到整车报文，即 API 无法获取整车数据！可以参考用户例程中 main.c 的 CAN 定时回调函数 void\* can\_xmt(void)，注意发送相关 ID 报文前一定要调用 API 函数 BydAutoCANRuntimeXmt(uint32\_t messageId, uint32\_t \*pMessageLen, uint8\_t \*pMessageData)，否则无法实现图 1 右侧的控制类 Set-API；同时



为了确定周期是否满足要求,最好在 ACU 与秦 Pro 后备箱的 CAN 线连接处用相关的 CAN 设备获取相关报文,分析报文周期是否满足要求,如果不满足,则需要修改 CAN 发送的机制。

## 2.2 CAN 接收说明

ACU 通过图 1 左侧的获取类 Get-API 获取整车的状态信息(轮速、方向盘角度等等),用户必须先设计 ACU 上的 CAN 设备接收函数,然后将获取的 CAN 报文数据通过 API 函数 `BydAutoCANRuntimeRcv(uint32_t messageID,uint32_t messageLen,uint8_t *pMessageData)` 写入到 API 数据库中,这样获取类 Get-API 就可以读取到相关的整车信息。参考代码在用户例程 `man.c` 的线程函数 `void *can_rcvThread(void *data)` 中。

## 3 秦 Pro 执行机构逻辑说明

以下逻辑经过本文作者亲自编码测试并且通过，开发者可以先使用“智能驾驶-ACU模拟平台.exe”工具连接安全网关进行测试，在“车辆状态”界面中查看“ESP进入允许”（通过调用GetESPAutoDrivePermissionInfo(&nAutoDrivePermission,&nTimeStamp)，当nAutoDrivePermission为1时，表示ESP允许进入智能驾驶）、“EPS进入允许”（通过调用GetDriveAutoPermissionInfo(&nDriveAutoPermission,&nTimeStamp)，当nDriveAutoPermission返回值为1时表示EPS可以请求进入智能驾驶）满足后，然后在“命令设置”界面点击右侧的“请求自动驾驶”，此时可以在“车辆状态”界面下的“驾驶模式反馈”板块查看各个执行机构是否进入智能驾驶模式。

### 3.1 组合开关逻辑说明

开发者在ACU端调用SendAutoDriveModeRequest(BYD\_AUTO\_DRIVE\_MODE\_OPEN)将请求组合开关进入智能驾驶模式，通过调用GetSwitchDriveModeFeedbackInfo(&nSwitchDriveModeFeedback,&nTimeStamp)获取组合开关模式反馈，如果nSwitchDriveModeFeedback反馈的值为2(BYD\_AUTO\_DRIVE\_MODE\_SWITCH\_FEEDBACK\_AUTO)，则表示组合开关进入智能驾驶模式当中，之后发送保持命令SendAutoDriveModeRequest(BYD\_AUTO\_DRIVE\_MODE\_HOLDING)；之后分别调用如下API进行控制，具体控制部件的宏定义以及开启关闭宏定义可以从idiving\_api.h中查找。

SetHornSwitch(BYD\_AUTO\_HORN\_OFF)控制喇叭

SetLampBrakeSwitch(BYD\_AUTO\_LAMP\_OFF)控制制动灯

SetRainWiper(BYD\_AUTO\_RAIN\_WIPER\_SHORT\_PRESS)控制雨刮

SetLampAutoSwitch(BYD\_AUTO\_LAMP\_AUTO\_INVALID)控制自动灯光

SetLampEmergencyWarningSwitch(BYD\_AUTO\_LAMP\_OFF)控制紧急警告灯

SetLampTurnSwitch(BYD\_AUTO\_LAMP\_OFF, BYD\_AUTO\_LAMP\_OFF)控制转向灯

SetLampOutSideSwitch(BYD\_AUTO\_LAMP\_SMALL, combobox\_lamp\_ofoff\_switch)控制小灯、近光灯、远光灯（前后雾灯虽已定义但暂不支持）。

当要退出智能驾驶则调用SendAutoDriveModeRequest(BYD\_AUTO\_DRIVE\_MODE\_CLOSE)，那么组合开发收到控制命令后，如果组合开关的反馈值为1(BYD\_AUTO\_DRIVE\_MODE\_SWITCH\_FEEDBACK\_NORMAL)，则将退出智能驾驶回到人工驾驶模式，上述控制API将不起作用，之后发送保持命令SendAutoDriveModeRequest(BYD\_AUTO\_DRIVE\_MODE\_HOLDING)。开发者可以根据自己的开发的智能驾驶思路对整车的组合开发进行逻辑控制。

### 3.2 电子手刹逻辑说明

开发者在ACU端调用SendAutoDriveModeRequest(BYD\_AUTO\_DRIVE\_MODE\_OPEN)将请求电子手刹EPB模块进入智能驾驶模式，通过调用GetEleBrakeDriveModeFeedbackInfo(&nEleBrakeDriveModeFeedback,&nTimeStamp)的nEleBrakeDriveModeFeedback反馈值为2(BYD\_AUTO\_DRIVE\_MODE\_ELE\_BRAKE\_FEEDBACK\_AUTO)，则表示EPB已经进入智能驾驶当中，此时可以根据实际的情景调用，之后发送保持命令SendAutoDriveModeRequest(BYD\_AUTO\_DRIVE\_MODE\_HOLDING)。

SetEpbState(BYD\_AUTO\_EPB\_SET\_INVALID)对电子手刹进行拉起、释放、无效等控制。值得注意的是在P档下EPB是无法响应SetEpbState(BYD\_AUTO\_EPB\_SET\_RELEASED)释放控制的。

在车辆运动过程中ACU调用SetEpbState(BYD\_AUTO\_EPB\_SET\_APPLYING)进行EPB拉起操作（车速高于6km/h）是无法起到效果的，目前只支持车辆静止状态下的拉起情况，所以遇到突发比如减速度失效，人工进行刹车失效，此时可以人工强制持续拉起EPB，那么EPB在持续人工拉起过程中将退出自动驾驶并保持人工驾驶模式下同时产生制动效果，所以在这个过程中SendAutoDriveModeRequest(BYD\_AUTO\_DRIVE\_MODE\_OPEN)将不起作用，直到手动放开，EPB才会重新进入智能驾驶模式当中。

开发者要让电子手刹EPB退出智能驾驶模式有2种方法，一种就是前面描述的人工持续性拉起EPB，另外一种就是调用



SendAutoDriveModeRequest (BYD\_AUTO\_DRIVE\_MODE\_CLOSE)，当nEleBrakeDriveModeFeedback的反馈值为1 (BYD\_AUTO\_DRIVE\_MODE\_ELE\_BRAKE\_FEEDBACK\_NORMAL) 则EPB回到人工驾驶模式当中，之后发送保持命令SendAutoDriveModeRequest (BYD\_AUTO\_DRIVE\_MODE\_HOLDING)。

### 3.3 档位控制器逻辑说明

开发者在ACU端调用SendAutoDriveModeRequest (BYD\_AUTO\_DRIVE\_MODE\_OPEN) 将请求档位控制器SCU模块进入智能驾驶模式，调用GetGearDriveModeFeedbackInfo (&nGearDriveModeFeedback, &nTimeStamp) 的nGearDriveModeFeedback反馈值为1 (BYD\_AUTO\_DRIVE\_MODE\_GEAR\_FEEDBACK\_AUTO) 则表示SCU已经进入智能驾驶模式当中，之后发送保持命令SendAutoDriveModeRequest (BYD\_AUTO\_DRIVE\_MODE\_HOLDING)，此时可以通过调用SetCarGear (BYD\_AUTO\_CAR\_GEAR\_P) 进行档位的P、D、R、N档控制。

开发者在ACU端调用SendAutoDriveModeRequest (BYD\_AUTO\_DRIVE\_MODE\_CLOSE)，当nGearDriveModeFeedback反馈值为BYD\_AUTO\_DRIVE\_MODE\_GEAR\_FEEDBACK\_NORMAL (即0) 则SCU进入人工驾驶模式当中，之后发送保持命令SendAutoDriveModeRequest (BYD\_AUTO\_DRIVE\_MODE\_HOLDING)。

### 3.4 电机控制器逻辑说明

在没有踩油门踏板和制动踏板下，在ACU端调用SendAutoDriveModeRequest (BYD\_AUTO\_DRIVE\_MODE\_OPEN) 将请求电机控制器进入智能驾驶模式，此时调用GetMCDriveModeFeedbackInfo (&nMCDriveModeFeedback, &nTimeStamp)，当nMCDriveModeFeedback反馈值为1 (BYD\_AUTO\_DRIVE\_MODE\_MC\_FEEDBACK\_AUTO) 则电机控制器进入智能驾驶模式，之后发送保持命令SendAutoDriveModeRequest (BYD\_AUTO\_DRIVE\_MODE\_HOLDING)。

当人工踩制动踏板深度大于6% (GetBrakeDeepnessInfo (&nBrakeDeepness, &nTimeStamp) 的nBrakeDeepness大于6)，电机控制器将退出智能驾驶，此时将不响应SendAutoDriveModeRequest (BYD\_AUTO\_DRIVE\_MODE\_OPEN) 的请求。

当人工踩油门深度大于10% (GetAccelerateDeepnessInfo (&nAccelerateDeepness, &nTimeStamp) 的nAccelerateDeepness大于10)，则电机控制器将退出智能驾驶，此时也不响应BYD\_AUTO\_DRIVE\_MODE\_OPEN进入智能驾驶命令请求。

开发者在ACU端调用SendAutoDriveModeRequest (BYD\_AUTO\_DRIVE\_MODE\_CLOSE)，当nMCDriveModeFeedback的反馈值是BYD\_AUTO\_DRIVE\_MODE\_MC\_FEEDBACK\_NORMAL (即0)，则电机控制器进入人工驾驶模式，之后发送保持命令SendAutoDriveModeRequest (BYD\_AUTO\_DRIVE\_MODE\_HOLDING)。

### 3.5 EPS 逻辑说明

开发者在ACU端通过调用GetDriveAutoPermissionInfo (&nDriveAutoPermission, &nTimeStamp) 来查看nDriveAutoPermission是否为1 (BYD\_AUTO\_EPS\_PERMISSION\_OK)，是表示EPS无内部故障，表示允许ACU发送进入智能驾驶模式请求，此时调用SetAutoDriveLateralState (BYD\_AUTO\_DRIVE\_MODE\_EPS\_OPEN) 将请求EPS进入智能驾驶模式，然后调用GetEPSDriveModeFeedbackInfo (&nEPSDriveModeFeedback, &nTimeStamp) 判断nEPSDriveModeFeedback为2 (即BYD\_AUTO\_DRIVE\_MODE\_EPS\_FEEDBACK\_AUTO) 则进入智能驾驶模式当中。

先通过GetAngularInfo (&nAngular, &nTimeStamp) 获取当前方向盘的角度，然后发送设置角度SetAngular (Angular)，其中Angular=7800+ nAngular。这样保证进入智能驾驶后，控制方向盘的角度为车辆当前的角度。后续根据路径规划和相关算法计算角度值，然后通过SetAngular (Angular) 进行整车横向控制。

与秦EV不同是手动拨动进入智能驾驶模式的EPS，EPS是不会退出智能驾驶模式的，所以在测试过程中，主驾驶可以很快介入方向盘的控制，但是在人为用力左打死 (+532°) 或者右打死 (-532°)，则将退出智能驾驶，此时不响应智能驾驶请求，解除左打死或者右打死EPS可以重新响应智能驾驶请求，进行控制。

目前秦Pro的EPS不支持方向盘角速度控制 (SetAngularVelocity (nAngularVelocity))。

如果整车断电，则需要等待5分钟，才能保证EPS完全退电，否则直接重启，那么nDriveAutoPermission的反馈值将





是2 (BYD\_AUTO\_EPS\_NOT\_PERMISSION) 此时, EPS无法响应智能驾驶请求。

开发者在ACU端调用SetAutoDriveLateralState(BYD\_AUTO\_DRIVE\_MODE\_EPS\_CLOSE), EPS则将退出智能驾驶模式, 此时EPS的驾驶模式反馈nEPSDriveModeFeedback为1 (BYD\_AUTO\_DRIVE\_MODE\_EPS\_FEEDBACK\_NORMAL)。

### 3.6 ESP 逻辑说明

开发者在ACU端调用GetESPAutoDrivePermissionInfo(&nAutoDrivePermission, &nTimeStamp), 若nAutoDrivePermission的返回值为1 (BYD\_AUTO\_DRIVE\_ESP\_PERMISSION\_OK), 如果没有踩制动踏板, 那么表示ESP响应智能驾驶纵向控制请求SetDriveAutoState(BYD\_AUTO\_DRIVE\_MODE\_AUTO\_STATE\_OPEN), 通过调用获取ESP的驾驶模式反馈函数GetESPDriveModeFeedbackInfo(&nESPDriveModeFeedback, &nTimeStamp), 如果nESPDriveModeFeedback的返回值为1 (BYD\_AUTO\_DRIVE\_MODE\_ESP\_FEEDBACK\_OPEN) 那么ESP已经进入智能驾驶模式中了。通过如下API进行加减速度控制:

SetAcceleratedVelocity( nAcceleratedVelocity ), 当nAcceleratedVelocity=100时, 实际对应的加速度为零。具体的对应关系可以查用户手册的API说明。

在ESP进入智能驾驶模式下, 人工踩制动踏板大于6% (GetBrakeDeepnessInfo (&nBrakeDeepness, &nTimeStamp) 的nBrakeDeepness大于6), 那么ESP将退出智能驾驶模式, 并且不响应BYD\_AUTO\_DRIVE\_MODE\_AUTO\_STATE\_OPEN命令请求, 而踩油门踏板ESP不会退出智能驾驶模式。

开发者调用SetDriveAutoState(BYD\_AUTO\_DRIVE\_MODE\_AUTO\_STATE\_CLOSE), 则ESP将退出智能驾驶, 此时nESPDriveModeFeedback的返回值将是0 (BYD\_AUTO\_DRIVE\_MODE\_ESP\_FEEDBACK\_CLOSE)。

### 3.7 ACU 智能驾驶请求、退出说明

开发者通过调用SetDriveModeFeedback(BYD\_AUTO\_DRIVE\_MODE\_AUTO) 向整车发送当前ACU状况并在仪表的右侧中上部显示, 如下说明, 其主要作用是用于通过图标显示当前ACU状况, 让驾驶者做出是否人工介入的决策。

BYD\_AUTO\_DRIVE\_MODE\_AUTO\_ABNORMAL\_1: 表示仪表没有显示ACU状态

BYD\_AUTO\_DRIVE\_MODE\_AUTO: 在仪表上显示绿色的方向盘脱手标志

BYD\_AUTO\_DRIVE\_MODE\_NORMAL: 在仪表上显示白色的手握方向盘标志

BYD\_AUTO\_DRIVE\_MODE\_AUTO\_ABNORMAL\_2: 在仪表上显示红色的方向盘脱手标志

#### A、智能驾驶开关

在方向盘上左侧的操控按钮的右下角有个智能驾驶开关, 开发者可以设计长按或者短按来表示请求智能驾驶, 通过调用GetAutoDriveKeyInfo(&nAutoDriveKey, &nTimeStamp), 获取nAutoDriveKey值, 具体设计由开发者自由发挥, 当满足某个条件, 则请求智能驾驶。

#### B、紧急退出按钮

在前排车内灯光按钮的后面有个紧急退出按钮, 驾驶员可以根据当前紧急情况, 关闭ACU, 人工介入, 之后整车将广播一个“紧急退出成功”。ACU端通过调用GetEmergencyExitKeyInfo(&nEmergencyExitKey, &nTimeStamp) 获取nEmergencyExitKey的值, 在没有按紧急退出按钮, 返回值一直是零, 当按下时, 网关将转发3帧给ACU, 然后网关停止转发任何报文, 并且仪表盘上显示“智能驾驶故障, 请人工接管”; 此时ACU可以设计关闭进入休眠, 但是网关需要断电重启后才会被ACU重新激活转发报文。

综合前面的各个执行机构的逻辑, ACU端进入智能驾驶请求和退出逻辑如图2所示。

### 3.8 ACU 异常退出情况

A、当 ACU 停止发送相关的 ID 报文, 则安全网关将停止转发报文, 并且整车相关执行机构将退出智能驾驶, 并且仪表盘闪烁红色的方向盘脱手标志, 并在仪表盘显示“智能驾驶故障, 请人工接管”。

B、当 ACU 与安全网关的 CAN 先断线, 则安全网关将停止转发报文, 并且整车相关执行机构将退出智

能驾驶，并在仪表盘上闪烁红色的方向盘脱手标志，仪表盘提示“智能驾驶故障，请人工接管”。

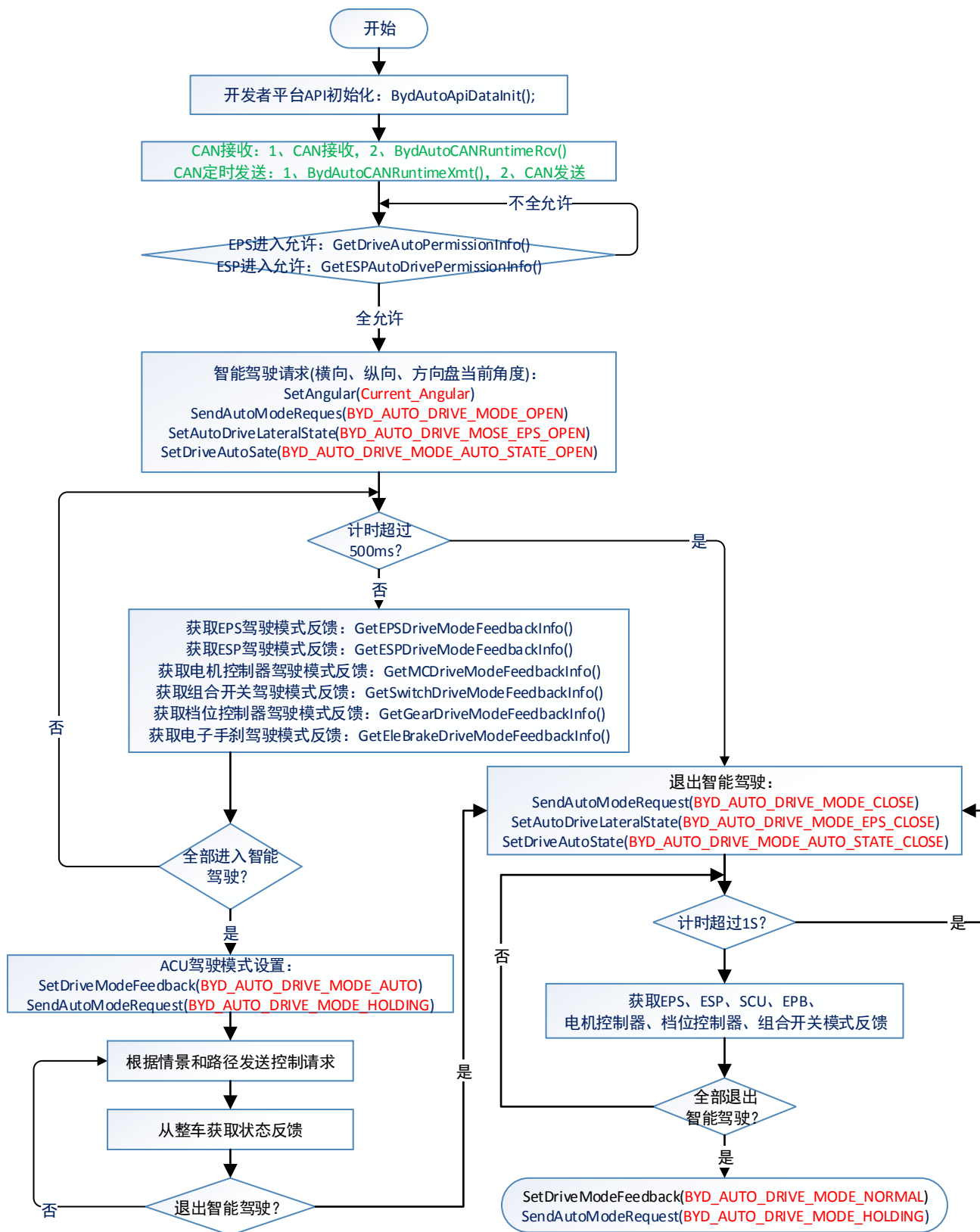


图 2 ACU 智能驾驶请求、退出逻辑图