

保密级别	秘密级
版本号	V0.1
归档部门	

设计文档

【车联网】

硬件研发部

二〇一九年一月

一、模块框架

模块采用底层驱动和上层应用的方式实现软件编写。

需实现内容：

- 获取车速
- 获取里程数
- 获取位置信息
- 获取故障信息
- 信号灯指示
- 离线存储
- 参数配置

1.1 底层驱动

底层驱动分为：

- Mc20 驱动
- CAN 驱动
- LED 驱动
- 系统驱动

1.2 上层应用

- 配置参数
- 上报数据
- 离线记录
- 状态检测

二、底层实现

2.1 MC20 驱动

2.1.1 包含函数

1. mc20 硬件初始化
2. mc20 指令集
3. mc20 GPRS 开启
4. mc20 GPS 开启
5. mc20 消息队列

2.1.2 初始化内容

1. Void Mc20_Init(void);

2.1.3 运行事件

1. Void MC20_Core_Loop_Run(void);
此函数 20ms 调用一次
2. Void MC20_GPRS_Start_Process(void);
此函数为触发事件
3. Void MC20_GPS_Start_Process(void);
此函数为触发事件
4. Void MC20_GPS_GetLocation_Process(void);
此函数为触发事件

2.1.4 结构体

Typedef struct

```
{  
    Uint8_t status_machine;  
    Uint8_t status_Rev;  
}Status_Machine_t;
```

Typedef stuct

```
{  
    Status_Machine_t GPRS_Status_Machine;
```

```
Status_Machine_t GPS_Status_Machine;  
}MC20_CoreStatus_t;
```

2.1.5 接口

```
Void MC20_GPRS_Start(void);  
Void MC20_GPS_Start(void);  
Void MC20_GPS_GetLocation(uint16 longitude,uint16_t latitude);
```

2.2 CAN 驱动

2.2.1 包含函数

- CAN 初始化
- CAN 获取车辆信息
- CAN 消息队列

2.2.2 初始化内容

```
Void CAN_Init(void)
```

2.2.3 运行事件

1. Void CAN_Data_Loop_Run(void);
此函数 1s 调用一次
2. Void CAN_Data_Analysis_Porcess(void);
此函数为触发事件

2.2.4 结构体

```
Typedef struct  
{  
    Uint16_t Value;  
    Uint8_t status;  
}CAN_Car_Info_t;  
Typedef struct  
{  
    CAN_Car_Info_t mileage;  
    CAN_Car_Info_t Speed;
```

```
}CAN_Kind_info_t;
```

2.1.5 接口

```
Void CAN_Start_Get_Mileage(void);  
Void CAN_Start_Get_Speed(void);  
Uint16_t CAN_Get_Mileage(void);  
Uint16_t CAN_Get_Speed(void);.
```

2.3 LED 驱动

2.3.1 包含函数

- LED 初始化
- LED 频率设置
- LED 运行

2.3.2 初始化内容

```
Void LED_Init(void)
```

2.3.3 运行事件

```
Void
```

2.3.4 结构体

2.3.5 接口

2.4 系统驱动

2.4.1 包含函数

参数保存修改
离线记录

2.4.2 初始化内容

```
Void System_Paramter_Load(void);
```

2.4.3 运行事件

2.4.4 结构体

2.4.5 接口

```
Void Offline_Record_Info(uint8_t * buf,uint16_t len);  
Void Offline_Read_Info(uint8_t num,uint8_t *buf,uint16_t len);
```

三、上层实现

3.1 初始化内容

Void User_Open_Func(void)

3.2 运行事件

Void User_Scan_Status_Loop(void);