CANMcu1为通信中心，设置为正常模式，接收来自其他Mcu的信息

CANMcu2为小车通信，负责发送小车速度状态 CAN总线状态为换环回状态

CANMcu3为环境通信，负责发送温度，湿度，光强 CAN总线状态为换环回状态

CAN通信采用PA11 PA12引脚

CANMcu2

tim2定时器2为小车提供动力，复用在PA0端口

定时器4为计时器，负责计时

定时器3为电机测速,使用TIM编码器接口测速，复用PA6 PA7

CANMcu3

使用DAC获取温度，光照

PA0引脚通过ad转换获取电池电压

PA1-3通过ad与dma的结合采集温度，光照，距离,电压

**问题1:**adc数据产生之后是通过那个函数传给dma外设地址

* 解决方法：ADC在完成一次转换后，将结果存储在其数据寄存器中（例如，ADC1->DR）。这个寄存器是一个固定的内存地址，DMA控制器会从这个地址读取数据。

**问题2：**在CAN发送前AD0 = AD\_Value[0];OLED\_ShowNum(1,1,AD0,4);显示3766（十进制），但是在CAN接受后，OLED\_ShowHexNum(4,6,RxMsg.Data[0],2);显示0E，因为AD1是uint16\_t而RxMsg.Data[0]是uint8\_t故在发送时会舍弃高八位，

解决方法：

发送时TxMsgArray[0].Data[0]= AD0>>8;

TxMsgArray[0].Data[1]= AD1;

一次CAN传输可以传32位数据，所以用一次CAN只能来传输两个数据了

**问题3：**CAN通信时一次只能发送一条CAN，里面只能装入4个8字节的数据，但是我有4个16字节的数据要发送就必须要用2个CAN，CAN接收时一次只能接一条，现在要做到在只有一条CAN接收结构体下保存两条Can的数据

解决方法:if(RxMsg.StdId==0x100)

{

Can\_RxADx[0] = RxMsg.Data[0]<<8 | RxMsg.Data[1];

Can\_RxADx[1] = RxMsg.Data[2]<<8 | RxMsg.Data[3];

}

if(RxMsg.StdId==0x200)

{

Can\_RxADx[2] = RxMsg.Data[0]<<8 | RxMsg.Data[1];

Can\_RxADx[3] = RxMsg.Data[2]<<8 | RxMsg.Data[3];

}

在接收时判断报文ID，然后赋值给uint16\_t Can\_RxADx[4]; 这样定义的接收结构体