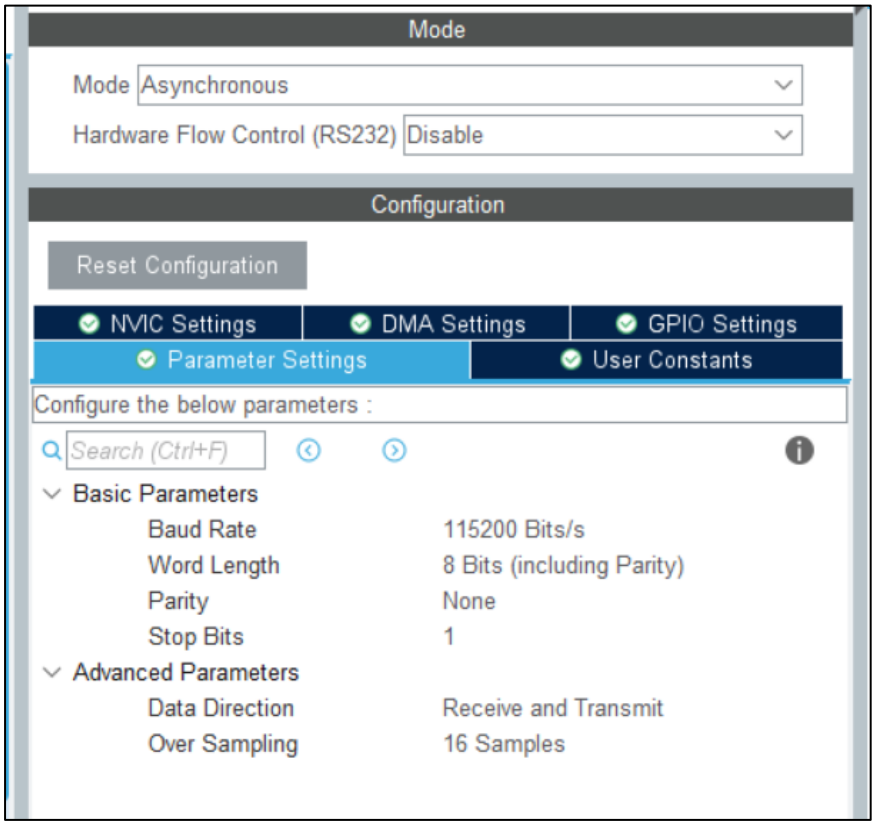
此文档为曹越编写，用于指导各车组移植新的裁判系统数据接收程序。

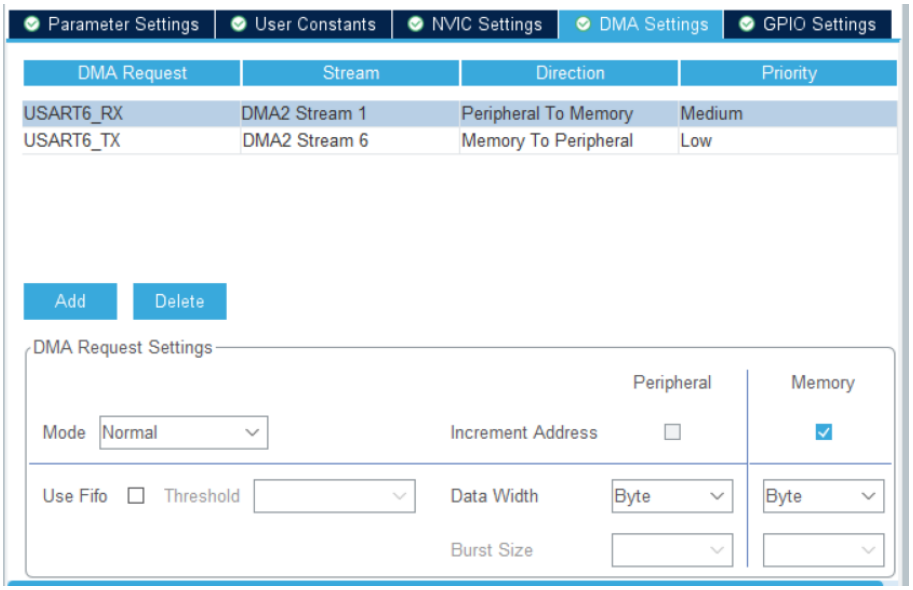
实际使用中根据需求对代码进行了修改，本文档仅做参考

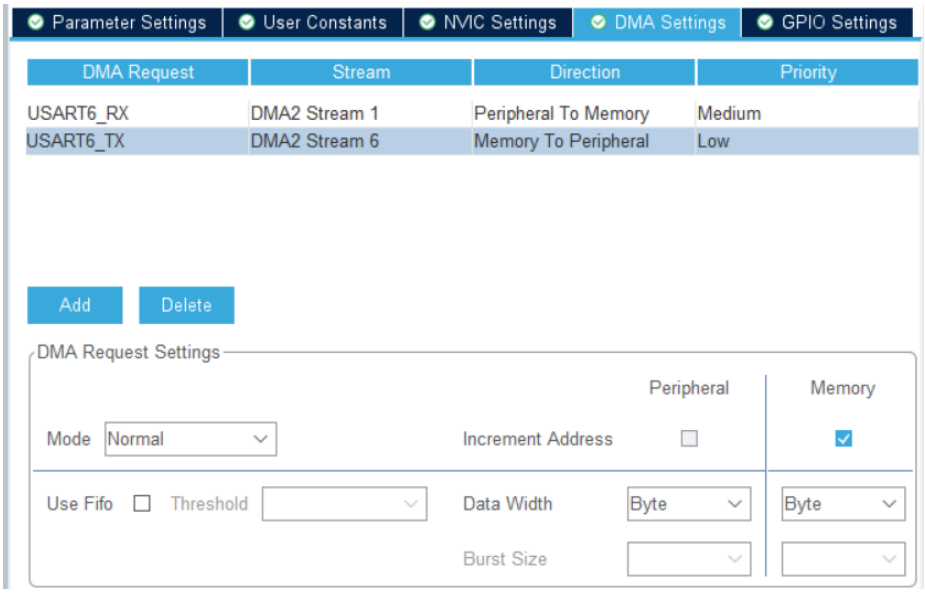
说明

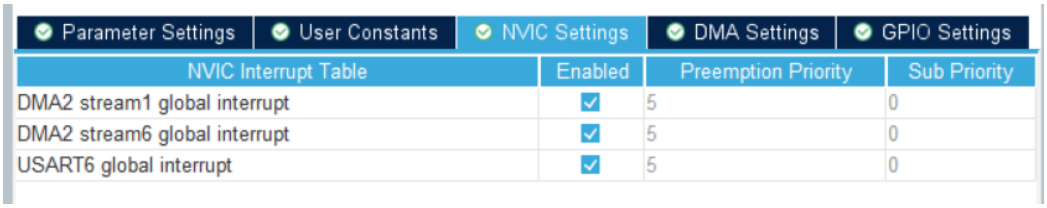
- 战队大多数机器人上的裁判系统数据接收程序均存在丢帧严重的问题，部分机器人上还没有裁判系统接收程序
- 本例程是以官方开发板 C 型例程中的裁判系统接收程序为基础，开发的一套基于裸机编程的，便于移植的裁判系统接收程序，主要解决了丢帧的问题
- 本例程仅实现了裁判系统数据接收的功能，没有实现数据发送的功能，因此功能上不包括绘制 UI 和机间通信
- 本例程基于简易版的电控程序框架开发完成，当中遗留有部分没有用到的文件，可以无视

裁判系统所用串口的 CubeMX 配置









需要移植的文件

CRC8_CRC16.c/.h 是官方 C 板例程中自带的 CRC 校验程序，可以替换为等效的程序
fifo.c/.h 是一个先进先出队列，使用开关中断的方式实现了读写保护
protocol.h 是裁判系统数据处理协议的定义
recv_header.c/.h 是一个时间戳系统，提供 HAL 库时间戳和帧序号两个数据
referee_protocol_v14.h 是裁判系统串口协议，v14 代表 1.4 版本
referee_receiver.c/.h 是裁判系统数据接收器的具体实现

裁判系统数据接收功能使用流程

定义裁判系统数据接收处理器、接收节点和数据结构体

```
11  RRManager_t RRManager;
12  RRNode_t RRNode_0x201;
13  ext_game_robot_status_t game_robot_status;
14  RRNode_t RRNode_0x202;
15  ext_power_heat_data_t power_heat_data;
```

在初始化函数中初始化接收器、接收节点，将接收节点添加到接收器，并启动接收器

```
23  //裁判系统接收初始化
24  RRManagerInit(&RRManager, &huart6);
25  RRNodeInit(&RRNode_0x201, 0x201, (void*)&game_robot_status, sizeof(ext_game_robot_status_t), NULL);
26  RRNodeAddToManager(&RRNode_0x201, &RRManager);
27  RRNodeInit(&RRNode_0x202, 0x202, (void*)&power_heat_data, sizeof(ext_power_heat_data_t), NULL);
28  RRNodeAddToManager(&RRNode_0x202, &RRManager);
29  RRManagerStart(&RRManager);
```

在轮询中调用接收函数，接收数据

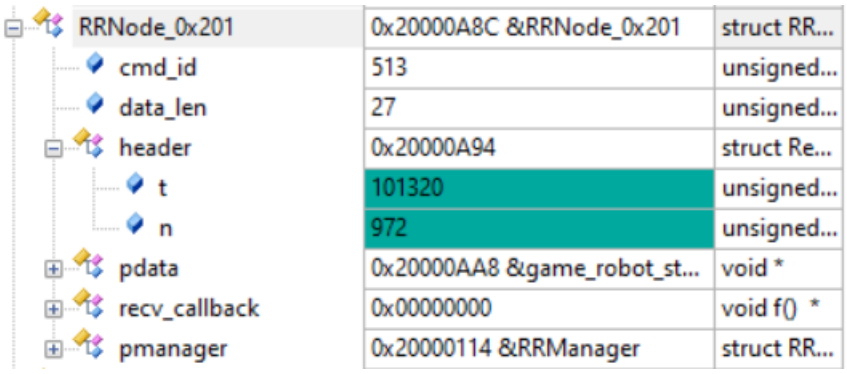
```
38          RRManagerRecv(&RRManager);
```

在中断文件中添加中断处理函数

```
338 void USART6_IRQHandler(void)
339 {
340   /* USER CODE BEGIN USART6_IRQn 0 */
341   RRManager_USART_IRQHandler(&RRManager);
342   /* USER CODE END USART6_IRQn 0 */
343   HAL_UART_IRQHandler(&huart6);
344   /* USER CODE BEGIN USART6_IRQn 1 */
345
346   /* USER CODE END USART6_IRQn 1 */
347 }
```

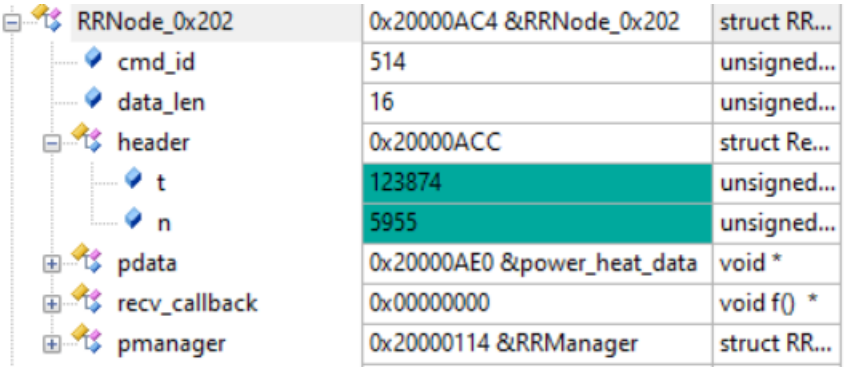
程序的正常表现为：每次调用接收函数 RRManagerRecv 后，如果收到了新的裁判系统数据帧，那么数据结构体里的数据就会更新。

测试



RRNode_0x201	0x20000A8C &RRNode_0x201	struct RR...
cmd_id	513	unsigned...
data_len	27	unsigned...
header	0x20000A94	struct Re...
t	101320	unsigned...
n	972	unsigned...
pdata	0x20000AA8 &game_robot_st...	void *
recv_callback	0x00000000	void f() *
pmanager	0x20000114 &RRManager	struct RR...

ID 为 0x201 的比赛机器人状态信息接收频率为：
972 / (101320/1000) = 9.59Hz
与手册上标注的 10Hz 近似



RRNode_0x202	0x20000AC4 &RRNode_0x202	struct RR...
cmd_id	514	unsigned...
data_len	16	unsigned...
header	0x20000ACC	struct Re...
t	123874	unsigned...
n	5955	unsigned...
pdata	0x20000AE0 &power_heat_data	void *
recv_callback	0x00000000	void f() *
pmanager	0x20000114 &RRManager	struct RR...

ID 为 0x202 的实时功率热量数据接收频率为：
5955 / (123874/1000) = 48.07Hz
与手册上标注的 50Hz 近似