

## 2017.6.1更新内容

- 修改位置计算为底盘主控计算，计算结果通过can2传回主控，原变量名不改变
- 重新测量速度，修正为真实速度，单位mm/s
- 添加读取遥控器的值

### 修改文件

- can2.c

全局位置从底盘主控读取，如下

```
void CAN2_RX0_Handler(){
    switch (can2_pRxMsg.StdId)
    ...
    case 0x10:
        Position_real[CH_X] = *(float *)&can2_pRxMsg.Data[0];
        Place_real[CH_X]= *(float *)&can2_pRxMsg.Data[4];
        break;
    case 0x11:
        Position_real[CH_Y] = *(float *)&can2_pRxMsg.Data[0];
        Place_real[CH_Y] = *(float *)&can2_pRxMsg.Data[4];
        break;
    case 0x12:
        Position_real[CH_A] = *(float *)&can2_pRxMsg.Data[0];
        Place_real[CH_A] = *(float *)&can2_pRxMsg.Data[4];
        break;
    ...
}
```

速度输出修正比例如下

```
temp[0] = ch[0] / 2.60 ; //X输出
temp[1] = ch[1] / 2.90 ; //Y输出
temp[2] = ch[2] * 3.10 ; //A输出
```

从底盘主控读取读取遥控器的值

```
void CAN2_RX0_Handler()
{
```

```

...
case 0x01:
    RC_ch1 = *((short *)&can2_pRxMsg.Data[0]);
    RC_s1 = can2_pRxMsg.Data[2];
    RC_s2 = can2_pRxMsg.Data[3];
    RC_ch2 = *((short *)&can2_pRxMsg.Data[4]);
    RC_ch3 = *((short *)&can2_pRxMsg.Data[6]);
    break;
...
}

```

- **can2.h**

添加宏定义控制主控自己计算位置还是从底盘主控读取位置，设置为0则从底盘主控读取位置，设置为1主控自己计算位置

```
#define USE_LOCAL_PLACE 0
```

添加遥控器通道宏定义

```

#define RC_X RC_ch1
#define RC_Y RC_ch2
#define RC_A RC_ch3

```