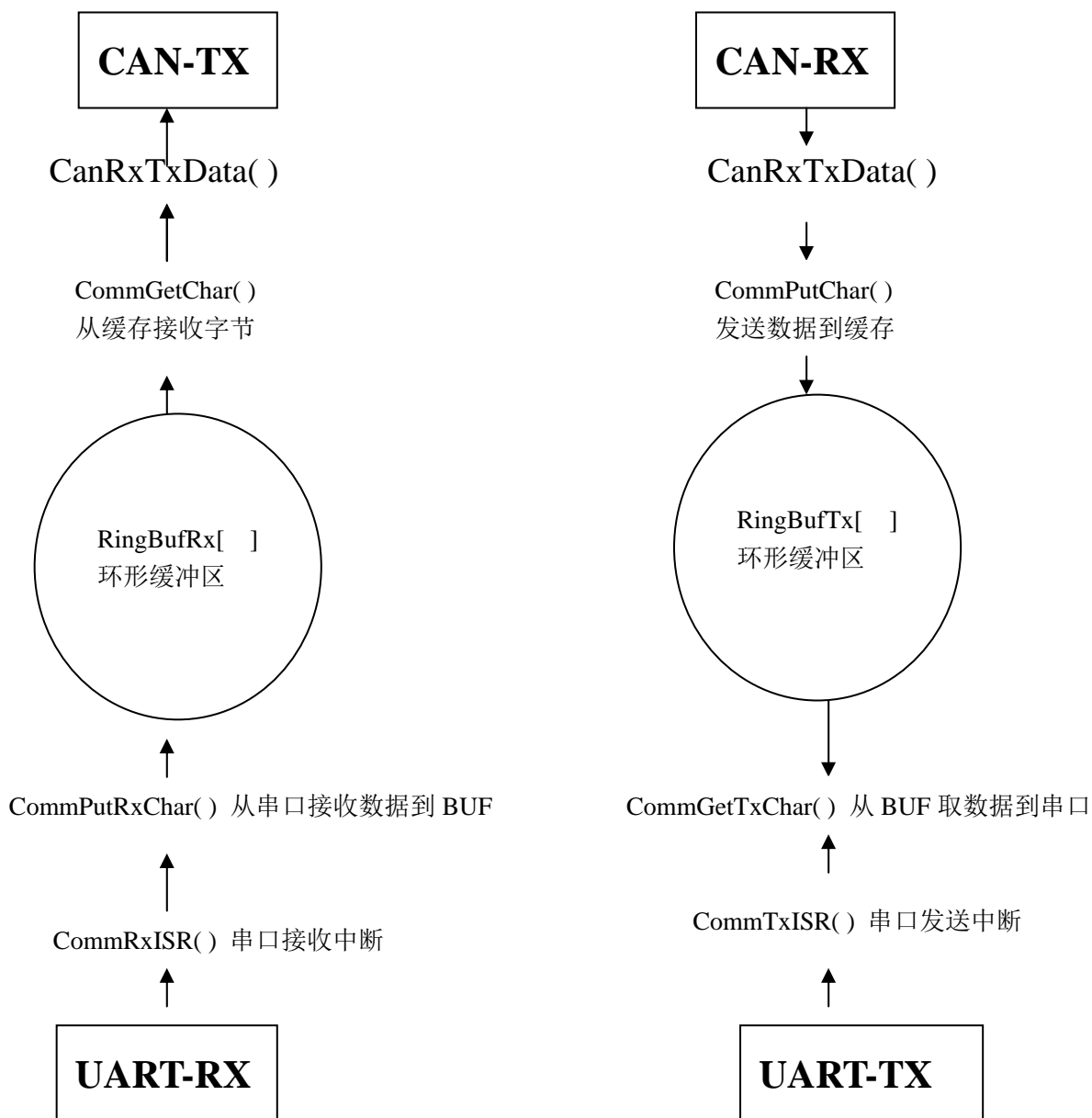


LPC2361 上实现 CAN 转串口

一、概述：

CAN 转串口在很多领域都有应用，也有很多现成的产品可以选择，本文只是通过一个例程介绍一下串行通信透明转换的方法。

二、数据传输：



上面这个图是 UART-CAN 的传输示意，UART 的发送和接收都采用中断方式，CAN 接收用中断方式，发送用直接发送方式，因为两发两收其中三个是中断方式，所以不会有效率问题，整个收发过程直接用串口的两个收发缓冲区。

(1) 如果 CAN 的速率大于 UART,则缓冲区 RingBufTx[]要设的大一些，RingBufRx[]可以相对小一些。

(2) 如果 UART 的速率大于 CAN,则缓冲区 RingBufRx[]要设的大一些，RingBufTx[]可以相对小一些。

三、函数介绍

1、CAN 发送

```
txmsg.len = 1;                //发送数据长度<=8
txmsg.id  = EXP_STD_ID;       //id 用 11 位
txmsg.CANBuf.CANData[0] = PAR(me); //接收串口消息中的数据。
```

2、CAN 接收：

```
void CAN_ISR_Rx1( void )
{
    DWORD * pDest;

    // initialize destination pointer

    pDest = (DWORD *)&CAN1RxMsg;

    *pDest = CAN1RFS; // Frame

    pDest++;

    *pDest = CAN1RID; // ID //change by gongjun
```

```
pDest++;  
  
*pDest = CAN1RDA; // Data A  
  
pDest++;  
  
*pDest = CAN1RDB; // Data B  
  
EvtPostQueISR(2,CAN_RX_SIG,CAN1);  
  
CAN1CMR = 0x04; // release receive buffer  
  
return;  
  
}
```

3、UART 发送和接收:

```
case COMRX_SIG:  
  
    c = CommGetRxChar(COMM2,&err); //从串口接收缓冲区取数据  
  
    EvtPostQue(2,CAN_TX_SIG,c);      //通知 CAN 发送出去  
  
    return (STAT_PTR)0;  
  
case  COMTX_SIG:  
  
    c=me->EventParam;  
  
    CommPutTxChar (COMM2,c);//向串口缓冲区发一个数据  
  
    return (STAT_PTR)0;
```

四、结束语:

用两个缓冲区、中断方式实现 UART 和 CAN 之间数据的透明传输。本例中 CAN 发送和接收只用一个字节，读者可以组帧发送和接收，以充分利用 CAN 的带宽。