RAK439 SPI 接口调试

1. 查看 SPI 通信 log

当 RAK439 无法初始化成功时，可以在 SDK 代码中打开 SPI 通信 log 查看，
NOS 修改.\platform\ rw_lib_platform.c 里的_spi_io_buffer 函数，
OS 修改.\platform\ rw_lib_platform_os.c 里的_spi_io_buffer 函数。

```
static void _spi_io_buffer(uint8_t* write, uint8_t* read, uint16_t len)
{
    uint8_t   dummy;
    int       i = 0;

    SPI_SetSS(WIFI_SPI, SPI_SS0);
#ifdef RW_SPI_DMA

#else
    if (read == NULL){
      for (i = 0; i < len; i++) {
        while (WIFI_SPI->STATUS & SPI_STATUS_TX_FULL);
        WIFI_SPI->TX0 = write[i];
//      printf("send=%x      ",write[i]);
        while (WIFI_SPI->STATUS & SPI_STATUS_RX_EMPTY);
        dummy = WIFI_SPI->RX0;
//      printf("recv dummy=%x\r\n",dummy);
      }
    } else {
      for (i = 0; i < len; i++) {
        while (WIFI_SPI->STATUS & SPI_STATUS_TX_FULL);
        if (write == NULL) {
          WIFI_SPI->TX0 = dummy;
//      printf("send dummy=%x  ",dummy);
        } else {
          WIFI_SPI->TX0 = write[i];
//      printf("send=%x  ",write[i]);
        }
        while (WIFI_SPI->STATUS & SPI_STATUS_RX_EMPTY);
        read[i] = WIFI_SPI->RX0;
//      printf("recv=%x\r\n",read[i]);
      }
    }
#endif

    SPI_ClrSS(WIFI_SPI, SPI_SS0);
}
```

执行 rw_sysDriverInit 时的 SPI 起始数据：

```
send=44 recv=b4
send=0 recv=b4
send=0 recv=b4
send=80 recv=b4
send=c2 recv=b4
send=0 recv=b4
send=0 recv=c
send=0 recv=5b
```

接收到 5b 表示 SPI 接口工作正常，SPI 接口接触不良或者供电不足则收不到 5b。
不同的模块接收的前 7 个字节可能不一样，但同一模块每次初始化的时候接收的前 7 个字节都是固定值。

2. 示波器查看波形

如果 RAK439 初始化时，SPI 接收数据不正确，可能是 SPI 设置不对，可以用示波器查看波形是否正常。

1. 修改代码降低 SPI 时钟，这里 MCU 使用 STM32F411（使用 SPI1，时钟源为 APB2），APB2=SYSCLK(96MHz)，SPI CLK=APB2/256=384KHz，方便抓取波形。

```
/*!< SPI configuration */
SPI_InitStructure.SPI_Direction = SPI_Direction_2Lines_FullDuplex;
SPI_InitStructure.SPI_Mode = SPI_Mode_Master;
SPI_InitStructure.SPI_DataSize = SPI_DataSize_8b;
SPI_InitStructure.SPI_CPOL = SPI_CPOL_High;
SPI_InitStructure.SPI_CPHA = SPI_CPHA_2Edge;
SPI_InitStructure.SPI_NSS = SPI_NSS_Soft;
SPI_InitStructure.SPI_BaudRatePrescaler = SPI_BaudRatePrescaler_256;
```

2. SPI 收到 8 个字节以后，进入 while(1)循环，方便抓取波形。

```
if(read == NULL) {
  for(i=0;i<len;i++) {
    while((WIFI_SPI->SR&SPI_FLAG_TXE)==RESET) ;
    if(write == NULL) {
      WIFI_SPI->DR = dummy;
    }else {
      WIFI_SPI->DR = write[i];
    }
        printf("send=%x      ",write[i]);
    while((WIFI_SPI->SR&SPI_FLAG_RXNE)==RESET);
    recv = WIFI_SPI->DR;
        printf("recv dummy=%x\r\n",dummy);
  }
}
else {
  for(i=0;i<len;i++) {
    while((WIFI_SPI->SR&SPI_FLAG_TXE)==RESET);
    if(write == NULL) {
      WIFI_SPI->DR = dummy;
          printf("send dummy=%x  ",dummy);
    }else {
      WIFI_SPI->DR = write[i];
           printf("send=%x  ",write[i]);
    }
    while((WIFI_SPI->SR&SPI_FLAG_RXNE)==RESET);
    read[i] = WIFI_SPI->DR;
        printf("recv=%x\r\n",read[i]);
    cnt++;
    if(cnt > 8) {
      while(1);
    }
  }
}
```

3. 串口上电打印如下

```
tcpudp_test.c:68  Host platform init...success
send=44  recv=d6
send=0  recv=d6
send=0  recv=d6
send=80  recv=d6
send=c2  recv=d6
send=0  recv=d6
send=0  recv=c
send=0  recv=5b
send=53  recv=5b
```

4. 示波器使用 single 模式，触发类型为边沿触发，触发源选择 CLK，下降沿触发。

图 1

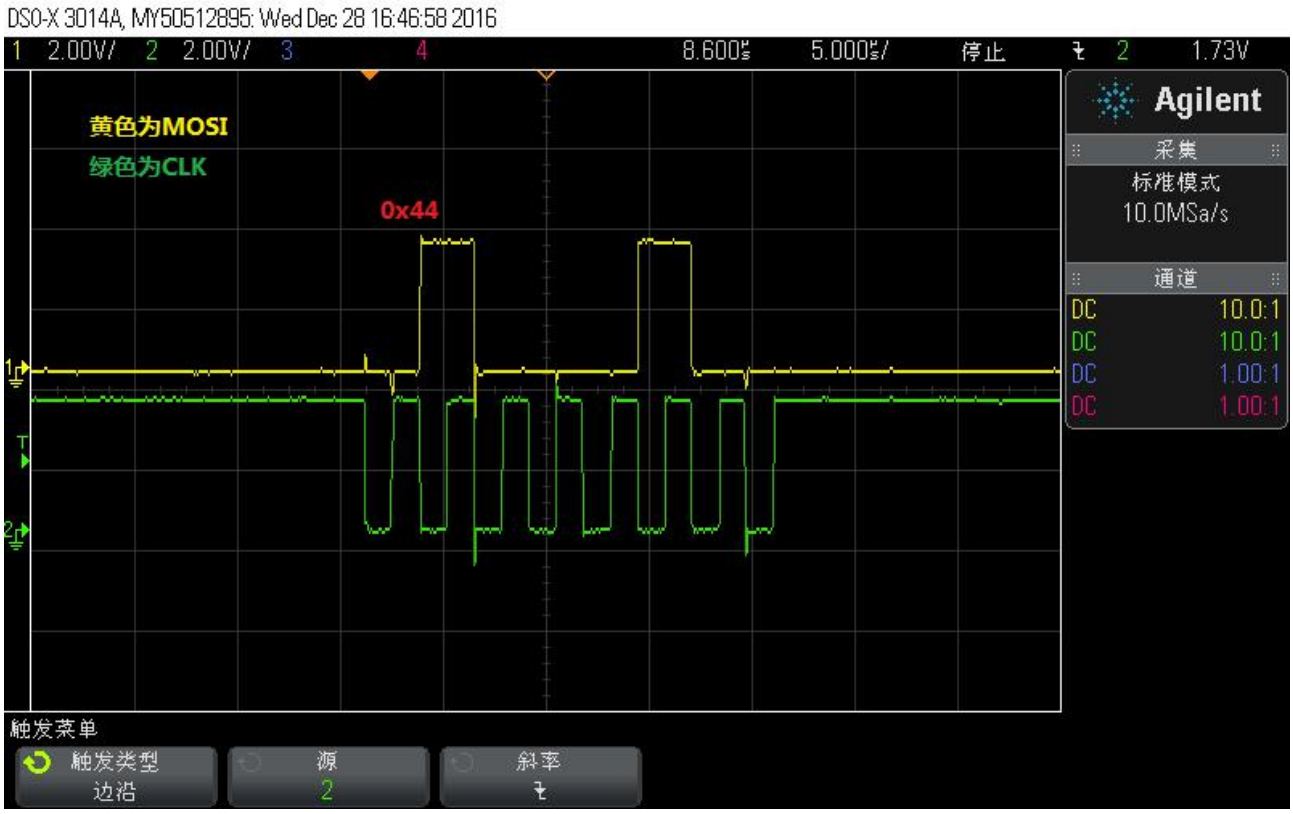

图 2

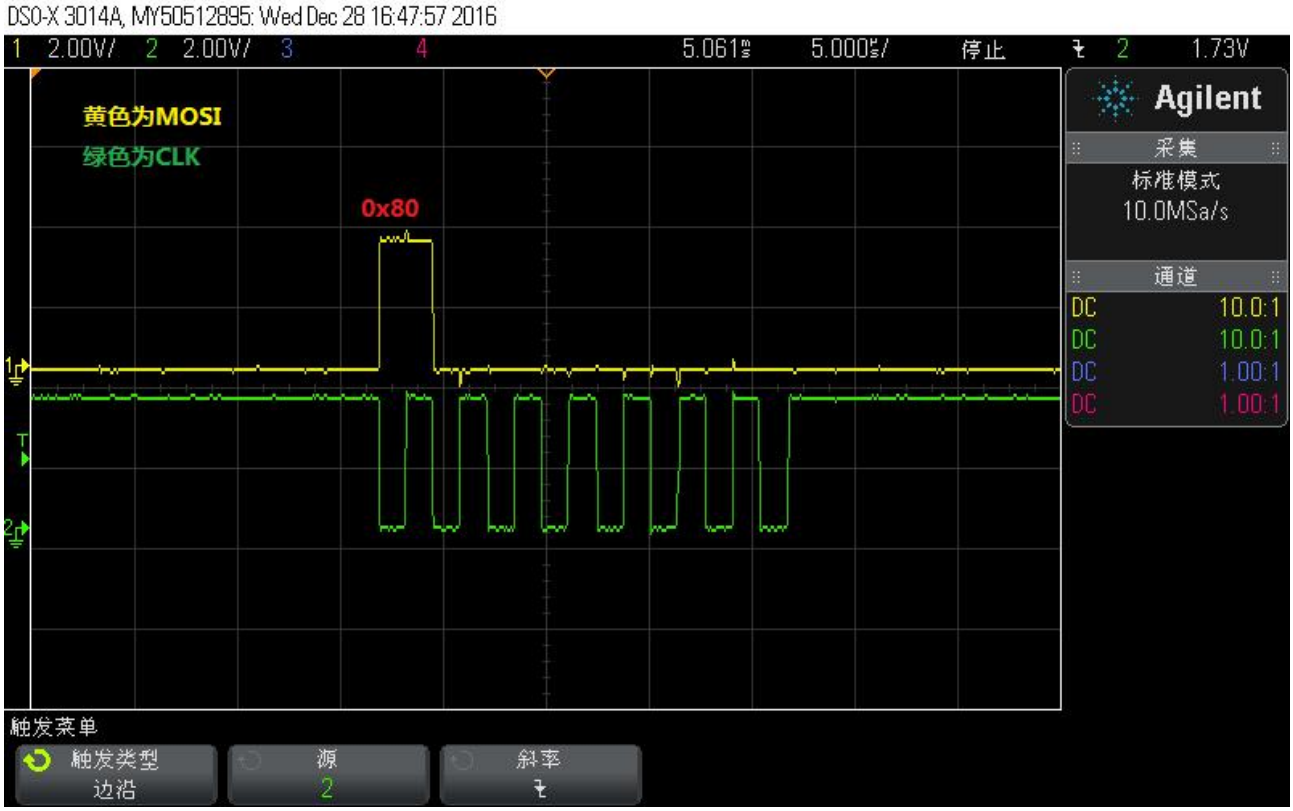

图 3

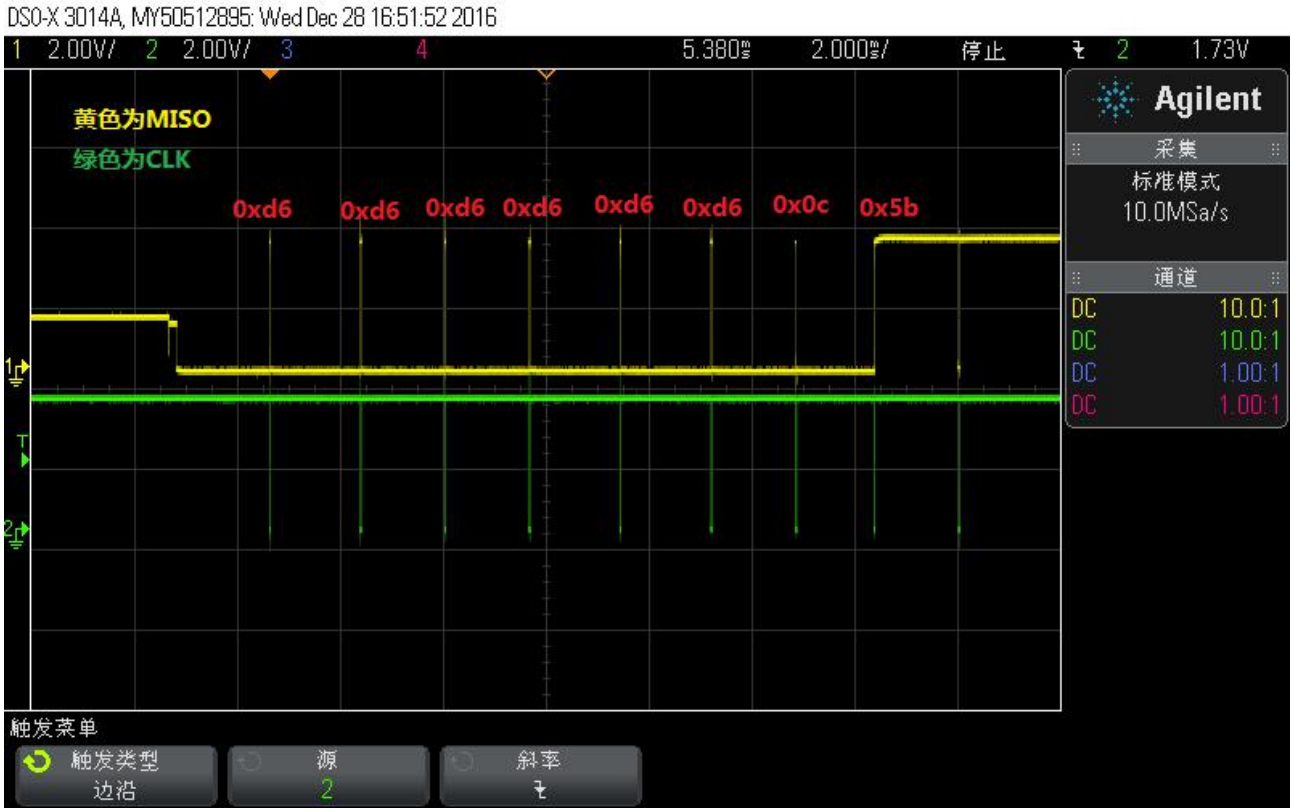

图 4

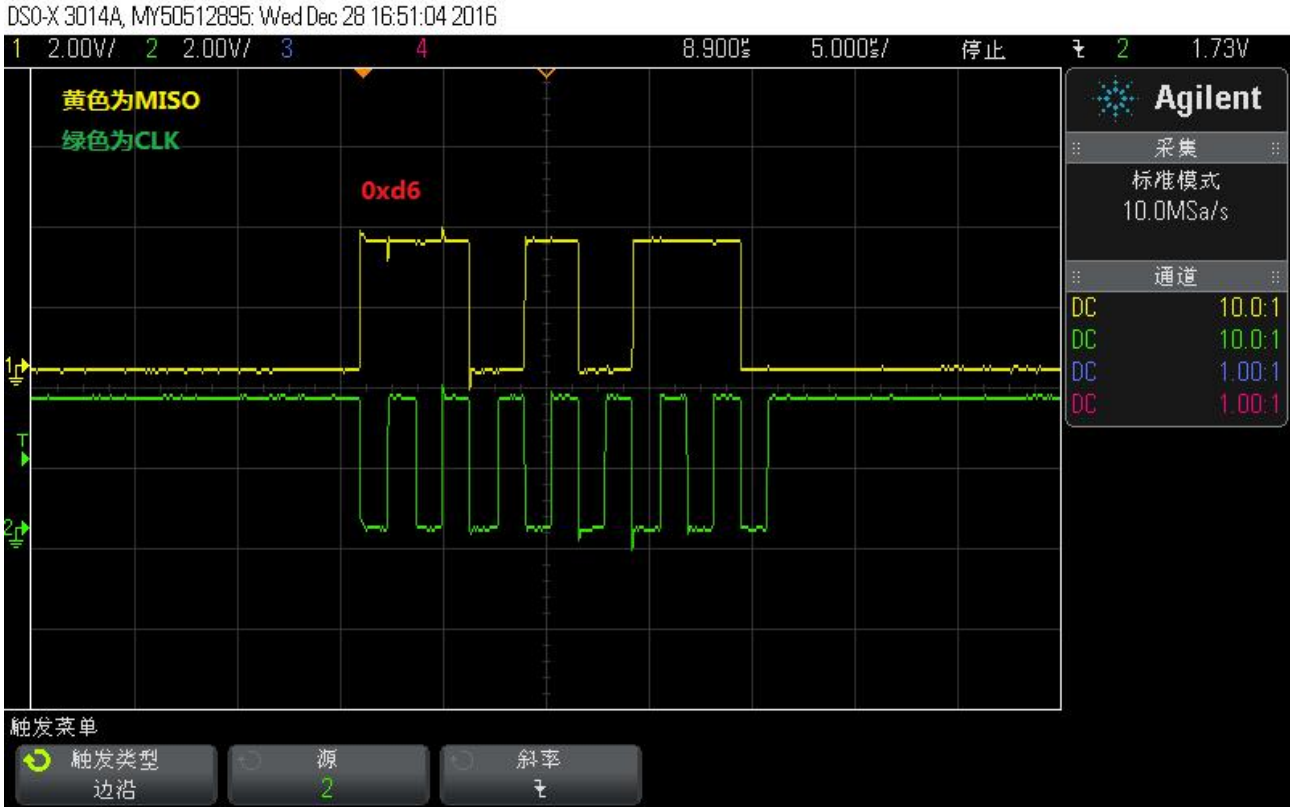

图 5

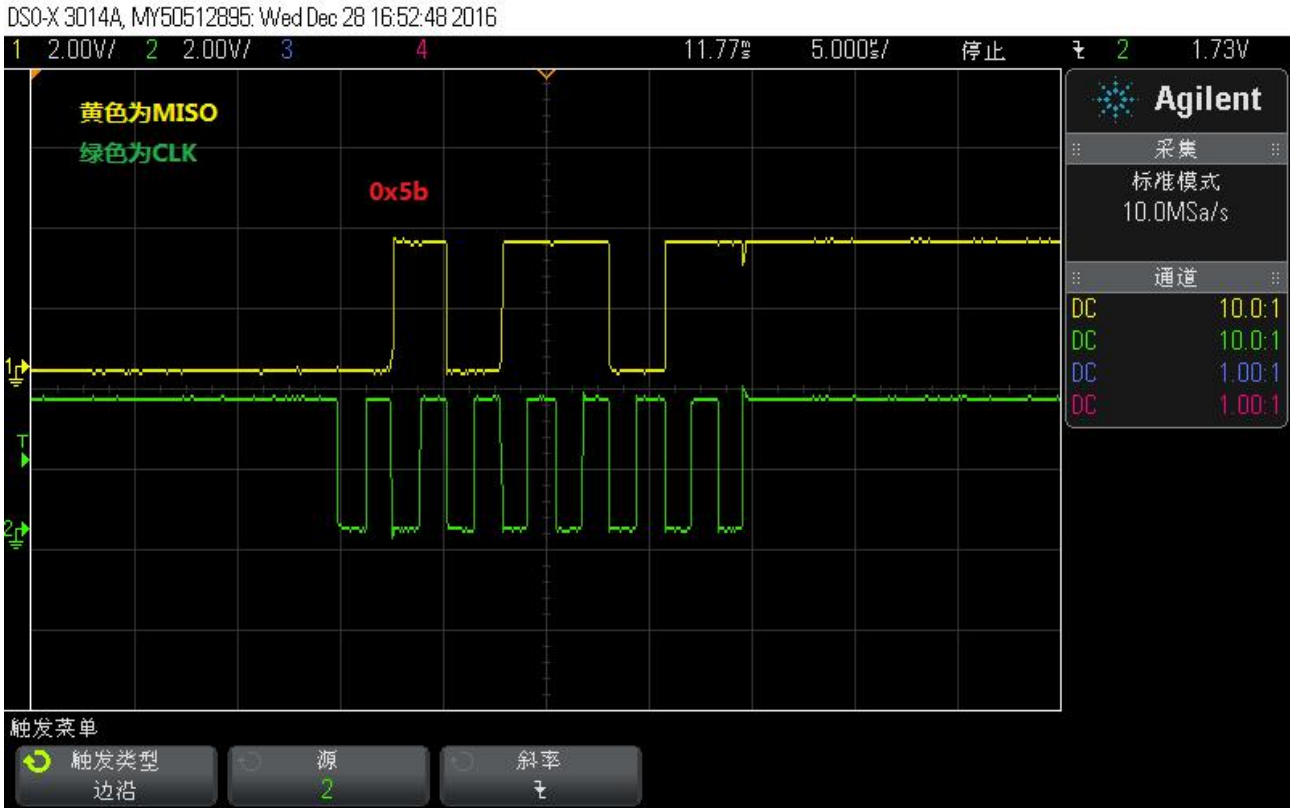

图 6

3. 修改记录

版本	作者	时间	修改内容
V1.0	harry	2016/12/28	创建文档