# 一、SPI协议详解

1、SPI相比I2C最大的优势有两点：一个是速度快，最高可以大几十M，甚至上百MHz，第二个就是SPI是个全双工。

2、SPI接口和I2C一样，一个SPI接口可以连接多个SPI外设，SPI通过CS引脚/数据线，片选引脚来选择和哪个SPI外设通信。SPI通信前先将指定的SPI外设对应的CS引脚拉低来选中此设备。

3、ALPHA开发板上通过ECSPI3接口连接了一个6轴传感器，引脚如下：

ECSPI3\_SCLK : UART2\_RX

ECSPI3\_MOSI：UART2\_CTS

ECSPI3\_SS0：UART2\_TXD

ECSPI3\_MISO: UART2\_RTS

6ULL一个SPI主接口有4个硬件片选，分别为SS0~SS3。

4、根据CPOL和CPHA可以设置四种工作模式，一般使用CPOL=0、CPHA=0。

# 二、6ULL SPI接口详解

1、6ULL的SPI接口叫做ECSPI，支持全双工、主丛可配置。

2、4个硬件片选信号，可以使用软件片选，这样一个SPI接口所能连接的外设就无限制了。

1、RXDATA寄存器为接收到的数据。

2、TXDATA寄存器为发送数据寄存器。

3、CONREG寄存器为配置寄存器，bit0置1，使能SPI。Bit3置1，表示当向TXFIFO写入数据以后马上开启SPI突发访问，也就是发送数据。Bit7:4设置SPI通道主从模式，bit7为通道3，bit4为通道0，我们使用到了SS0，也就是通道0，因此需要设置bit4为1。Bit19：18设置为00，我们使用到SS0，也就是通道0。Bit31:30设置突发访问长度，我们设置为7，也就是8bit突发长度，一个字节。

4、CONFIGREG寄存器的bit0为PHA，设置为0，表示 串行时钟的第一个跳变沿开始采集数据。设置bit4为PO，设置为0，表示SCLK空闲的时候为低电平。Bit8设置0。Bit12设置 为0。Bit16设置为0，表示空闲的时候数据线为高。Bit20设置为0，表示SCLK空闲的时候为低。

5、STATREG寄存器，bit0表示TXFIFO为空，我们在发送数据之前要等待TXFIFO为空，也就是等待bit0为1。Bit3表示RXFIFO是否有数据，为1的时候示RXFIFO至少有1个字的数据，我们在接收数据的时候要等到bit3为1。

6、PERIODREG寄存器，bit14:0设置wait states时间，我们设置为0X2000。Bit15设置wait states的时钟源为SPI CLK，将此位设置0。Bit21:16表示片选信号的延时，可设置0-63，这里设置为0.

7、SPI时钟设置！

SPI时钟源最终来源于pll3\_sw\_clk=480MHz/8=60MHz，设置CSCDR2寄存器的bit18为0，也就是ECSPI时钟源为60MHz。bit24:19设置为0，表示1分频，因此最终进入到SPI外设的时钟源为60MHz

ECSPI模块还需要对时钟进行两级分频，由ECSPI\_CONREG寄存器设置。Bit15:12设置前级分频，可以设置0~0xf，表示1~16分频。Bit11:8设置2级分频，设置2^n分频，n=0~15.

# 三、ICM20608简介

# 四、实验程序编写

1、我们在使用浮点计算的时候程序卡死了，因为我们没有开始6UL的硬件浮点运算。我们在编译的时候没有使用浮点。解决此问题需要两点：

①、开启6UL的硬件浮点单元

②、编译的是时候指定硬件浮点。