**KY-IMU102N-A0/B0使用SPI通讯操作指南**

1. **SPI定义以及简单说明**

SPI 协议是由摩托罗拉公司提出的通讯协议(Serial Peripheral Interface)，即串行外围设备接口，是一种高速全双工的通信总线。它被广泛地使用在 ADC、 LCD 等设备与 MCU 间，要求通讯速率较高的场合。

SPI 通讯使用 3 条总线及片选线， 3 条总线分别为 SCK、 MOSI、 MISO，片选线为SS，它们的作用介绍如下：

（1）SS （Slave Select）：从设备选择信号线，常称为片选信号线，也称为 NSS、 CS，以下用 NSS 表示。

而每个从设备都有独立的这一条 NSS 信号线，本信号线独占主机的一个  
引脚，即有多少个从设备，就有多少条片选信号线。

而 SPI 协议中没有设备地址，它使用 NSS 信号线来寻址，当主机要选择从设备时，把该从设备的 NSS 信号线设置为低电平，该从设备即被选中，即片选有效，接着主机开始与被选中的从设备进行 SPI 通讯。所以SPI 通讯以 NSS 线置低电平为开始信号，以 NSS 线被拉高作为结束信号。

（2）SCK (Serial Clock)： 时钟信号线，用于通讯数据同步。它由通讯主机产生，决定了通讯的速率，不同的设备支持的最高时钟频率不一样，两个设备之间通讯时，通讯速率受限于低速设备。

（3）MOSI (Master Output，Slave Input)： 主设备输出/从设备输入引脚。主机的数据从这条信号线输出，从机由这条信号线读入主机发送的数据，即这条线上数据的方向为主机到从机。  
 （4）MISO(Master Input，Slave Output)： 主设备输入/从设备输出引脚。主机从这条信号线读入数据，从机的数据由这条信号线输出到主机，即在这条线上数据的方向为从机到主机。

1. **KY-IMU102N-A0/B0关于SPI的相关说明**

表1 KY-IMU102N-A0/B0针脚定义

|  |  |  |  |
| --- | --- | --- | --- |
| 管脚序号 | 名称 | 类型 | 描述 |
| 10，11，12 | VDD | 电源 |  |
| 13，14，15 | GND | 电源 |  |
| 7 | DIO1 | 输入/输出 | 通用IO，可配置 |
| 9 | DIO2 | 输入/输出 |
| 1 | DIO3 | 输入/输出 |
| 2 | DIO4 | 输入/输出 |
| 3 | SPI-CLK | 输入/输出 | SPI，主从模式可配置，默认为从模式 |
| 4 | SPI-MISO | 输入/输出 |
| 5 | SPI-MOSI | 输入/输出 |
| 6 | SPI-/CS | 输入/输出 |
| 19 | UART-TXD | 输出 | UART，波特率可配置，默认为230400bps |
| 21 | UART-RXD | 输入 |
| 18 | CAN-T | 输出 |  |
| 20 | CAN-R | 输入 |
| 8 | RST | 输入 | 复位 |
| 23 | VDDRTC | 电源 |  |
| 其它 | NC | 备用 | 厂家保留 |

从表1可以看出，我们主要使用KY-IMU102N-A0/B0的3456针脚。从图1以及图2可以看出，KY-IMU102N-A0/B0的针脚排序方式：

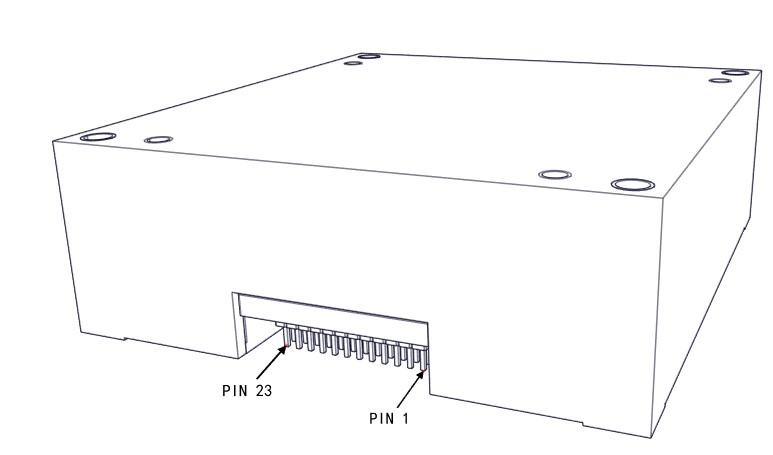


图1 KY-IMU102N-A0/B0的针脚排序方式

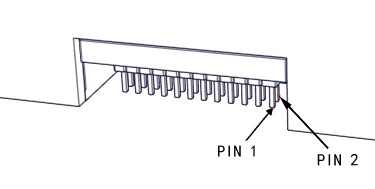


图2 KY-IMU102N-A0/B0的针脚排序方式

当配合KY-EVK-01底板使用时,底板的连接注意事项如图三所示：

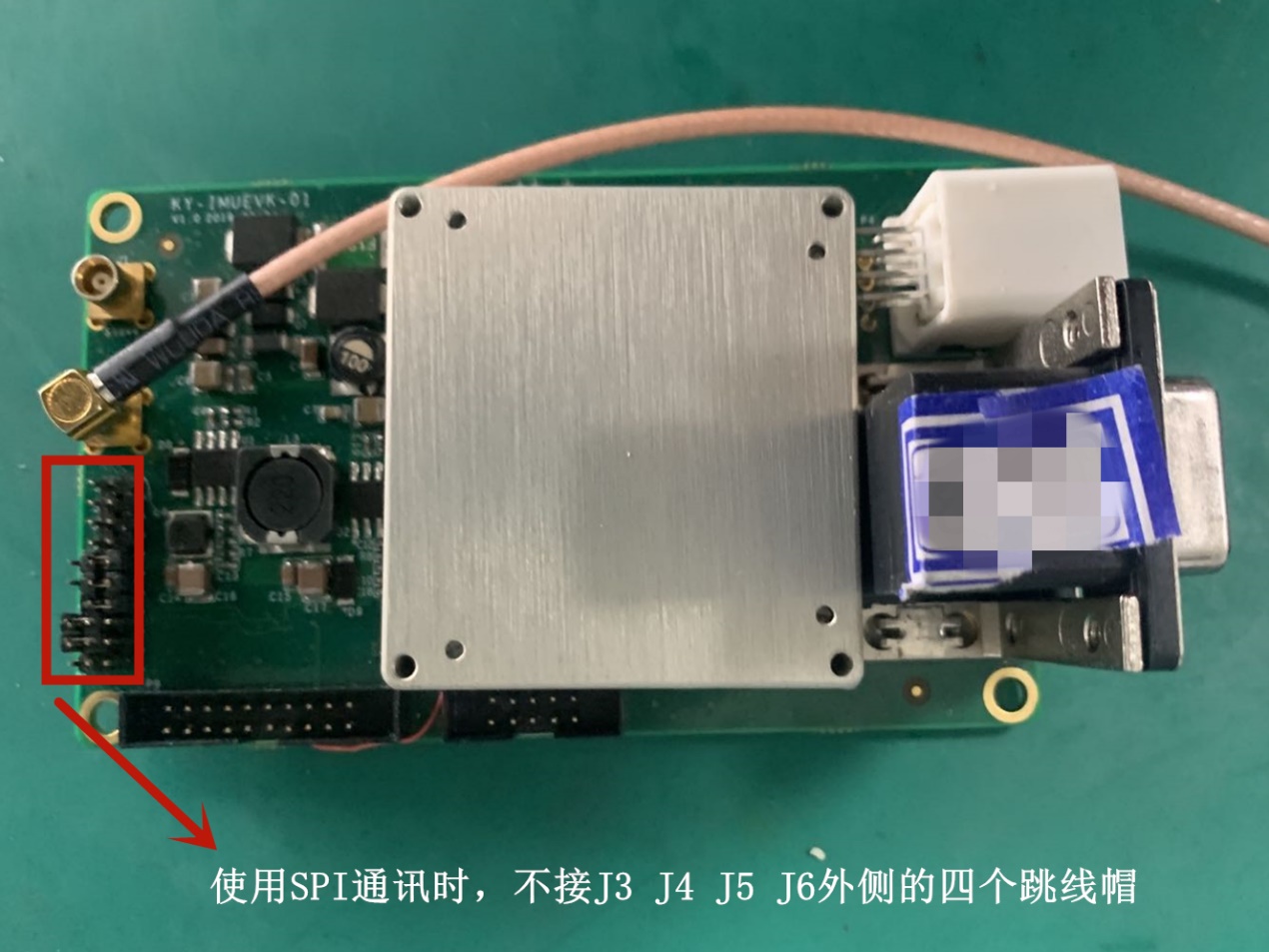


图3 KY-EVK-01的注意事项

1. **SPI转USB模块的相关说明**

我们使用的是如下图所示的SPI转USB模块（志明电子，参考购买链接：https://item.taobao.com/item.htm?id=540102939542），主要使用右侧SCK,MOSI,MISO,SS四个针脚和下排的GND针脚，其他针脚无需使用，转换模块USB端插在电脑上即可电脑供电，无需外部供电。



图4 SPI转USB模块示意图

SPI转换模块与KY-EVK-01的硬件连接示意图如图5所示：

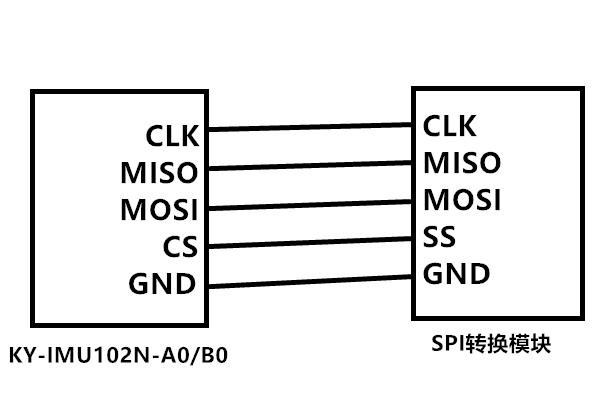


图5 SPI与KY-EVK-01的硬件连接示意图

1. **SPI转USB模块软件的使用说明**

我们使用的是USB转SPI模块的配套软件，打开后界面如图6：



图6 SPI转USB模块的配套软件

请注意红框圈出的部分，勾选使能，选择HEX模式，其他配置如下：

主从模式选择：主模式

传输模式选择：模式11

数据长度选择：16位

频率选择：4M

软件界面左下方为命令发送区域，左上方为数据接收区。可以发送的指令如表2所示：

表2用户寄存器存储器映射

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名称** | **R/W** | **PAGE\_ID** | **地址** | **默认** | **寄存器描述** |
| TEMP\_OUT | R | 0x00 | 0x0E | N/A | 温度 |
| X\_GYRO\_LOW | R | 0x00 | 0x10 | N/A | x轴陀螺仪输出，低位字 |
| X\_GYRO\_OUT | R | 0x00 | 0x12 | N/A | x轴陀螺仪输出，高位字 |
| Y\_GYRO\_LOW | R | 0x00 | 0x14 | N/A | y轴陀螺仪输出，低位字 |
| Y\_GYRO\_OUT | R | 0x00 | 0x16 | N/A | y轴陀螺仪输出，高位字 |
| Z\_GYRO\_LOW | R | 0x00 | 0x18 | N/A | z轴陀螺仪输出，低位字 |
| Z\_GYRO\_OUT | R | 0x00 | 0x1A | N/A | z轴陀螺仪输出，高位字 |
| X\_ACCL\_LOW | R | 0x00 | 0x1C | N/A | x轴加速度计输出，低位字 |
| X\_ACCL\_OUT | R | 0x00 | 0x1E | N/A | x轴加速度计输出，高位字 |
| Y\_ACCL\_LOW | R | 0x00 | 0x20 | N/A | y轴加速度计输出，低位字 |
| Y\_ACCL\_OUT | R | 0x00 | 0x22 | N/A | y轴加速度计输出，高位字 |
| Z\_ACCL\_LOW | R | 0x00 | 0x24 | N/A | z轴加速度计输出，低位字 |
| Z\_ACCL\_OUT | R | 0x00 | 0x26 | N/A | z轴加速度计输出，高位字 |
| PROD\_ID | R | 0x00 | 0x7E | 102 | 产品标识(102)输出 |

我们以最后一行产品标识为例，发送指令7E00，反馈的为0066，直接将16进制转换为10进制，结果为102，反馈内容正确。（请见图6）

数据解析以及转换公式如下图所示：

当前温度= 25+ TEMP\_OUT\* 0.00565

x轴陀螺值=0.02\* X\_GYRO\_OUT

y轴陀螺值=0.02\* Y\_GYRO\_OUT

z轴陀螺值=0.02\* Z\_GYRO\_OUT

x轴加速度计值= (long)(X\_ACCL\_OUT \*65536+ X\_ACCL\_LOW) \*0.00001220703125\*0.001

y轴加速度计值= (long)(Y\_ACCL\_OUT \*65536+ Y\_ACCL\_LOW) \*0.00001220703125\*0.001

z轴加速度计值= (long)(Z\_ACCL\_OUT \*65536+ Z\_ACCL\_LOW) \*0.00001220703125\*0.001

1. **SPI数据读取注意事项**

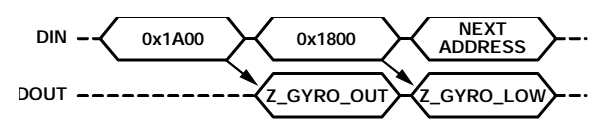


图7 SPI读操作示例

通过图7的SPI数据请求和读取示例我们可以看出，当请求第一次时，请求的结果会在第二次请求后出现。也就是本次指令发送后反馈的内容，将在下一次指令发送后回馈。

如图6所示，假设我们连续发送三次7E00指令，在接收框出现了三次0066，我分别标注了123，则发送第一次7E00后对应的结果为第2次出现的0066，第二次发送7E00后对应的结果为第3次出现的0066。那么上电后首次发指令后反馈的内容无效。