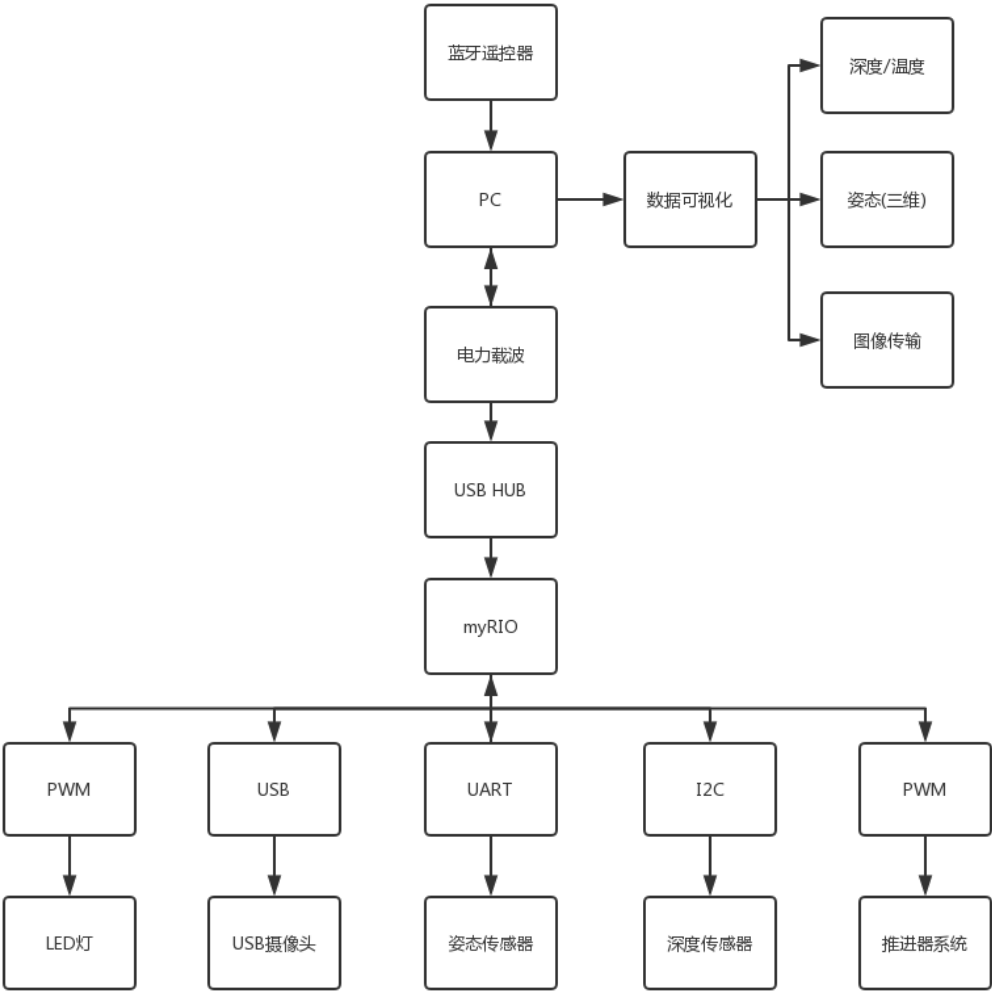


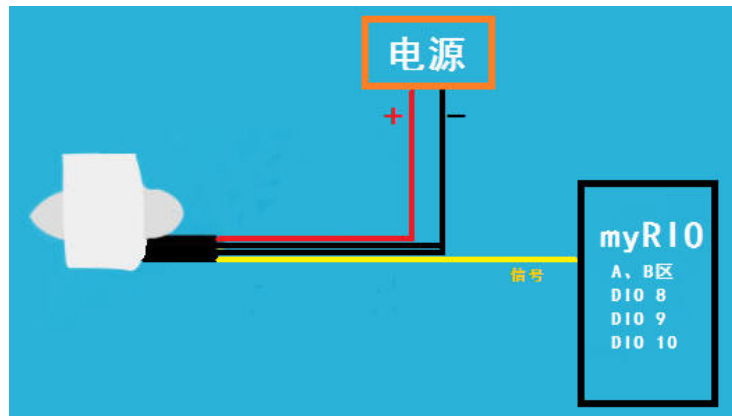
各传感器与 myRIO 的数据通讯



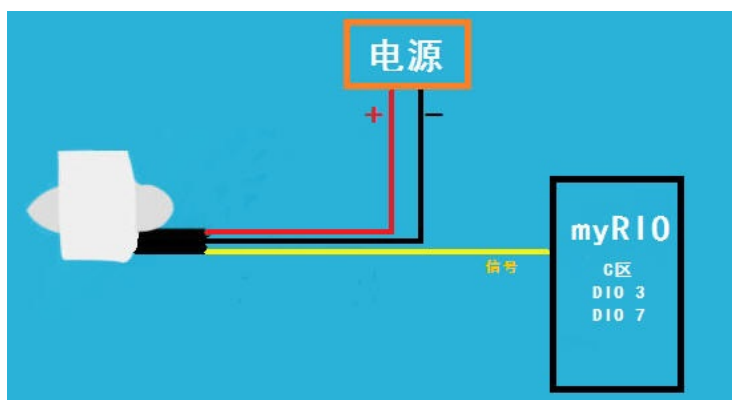
技术路线

推进器和 LED 灯的控制

因为推进器与 LED 照明灯的控制方式是相同的，均采用脉冲宽度调制 (PWM) 方式控制。所以此处将它们放在一起介绍。如图 (a) 动力总成与 myRIO 连接示意图、(b) 照明灯与 myRIO 连接示意图所示，将推进器、电调和桨叶构成的动力总成以及 LED 灯与 myRIO 进行连接。

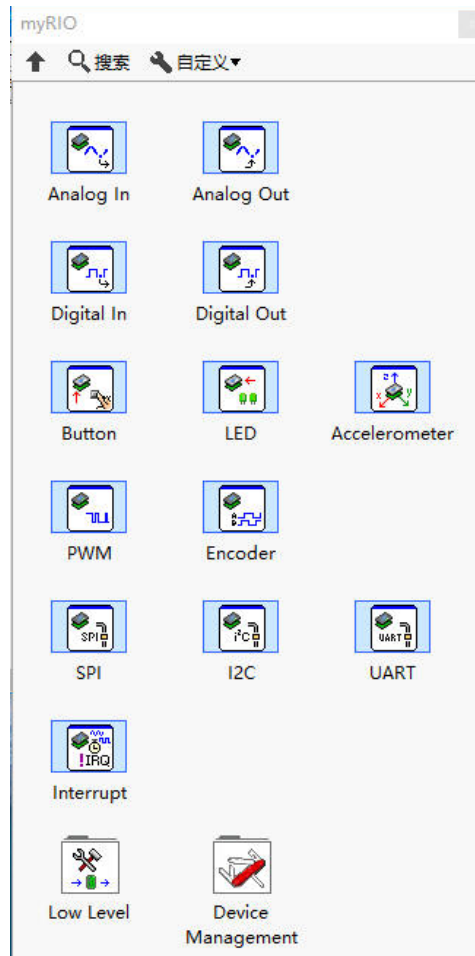


(a) 动力总成与 myRIO 连接示意图



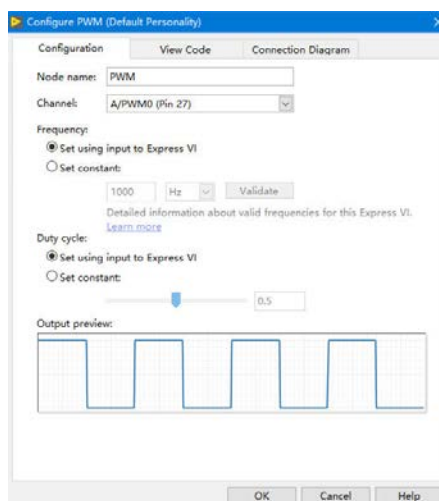
(b) 照明灯与 myRIO 连接示意图

如图在 LabVIEW 中，存在一个为 myRIO 专门编写的模块，用于与 myRIO 上对应的接口进行通讯。前面已经介绍过推进器与 LED 照明灯的控制方式是相同的，均采用脉冲宽度调制 (PWM) 方式控制，所以选择 myRIO 函数模块中的 PWM 函数进行对推进器和 LED 灯的控制。

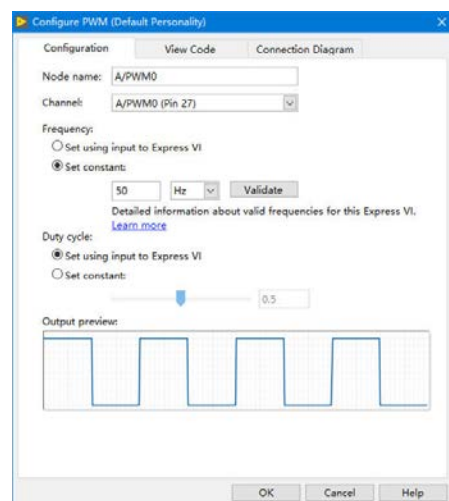


LabVIEW 中 myRIO 模块

如图所示，通过放置 PWM 函数后，进入 PWM 函数配置界面进行配置的过程。



(a) PWM 函数配置前



(b) PWM 函数配置后

PWM 函数配置过程

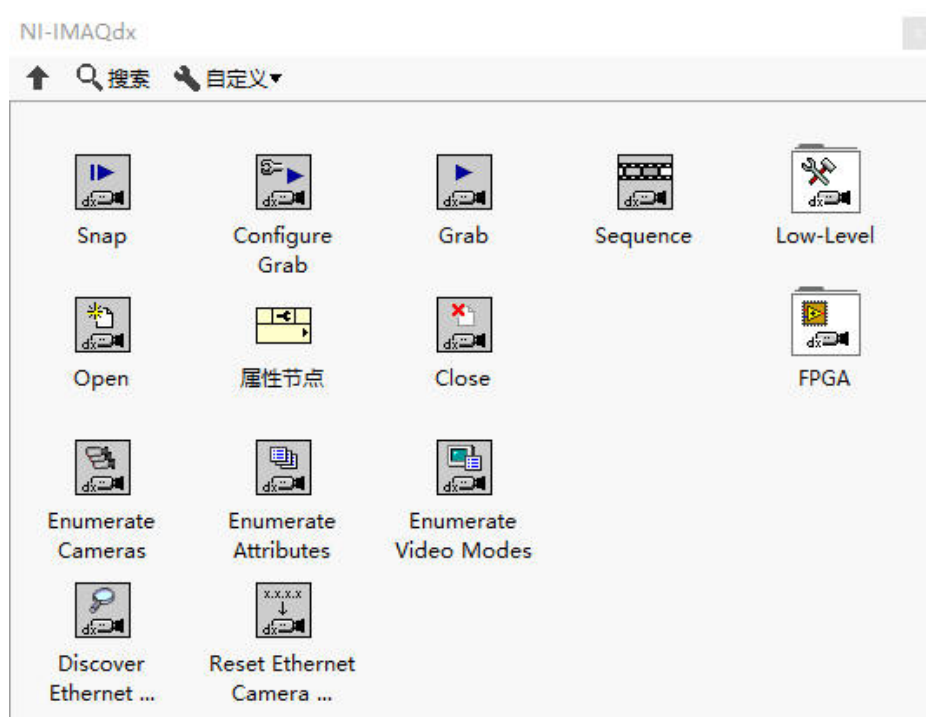
由于推进器和 LED 灯的 PWM 范围有所不同，但是对应的频率是一致的。所

以根据需要设置函数的标签名以及对应的通道名来区分不同的推进器和 LED 灯，设置好一个常数频率，而对应的占空比则有后续的手柄信号来控制。设置好的 PWM 函数配置界面如图所示(由于只是对应的标签名和通道改变，此处只一个推进器的设置为例)。

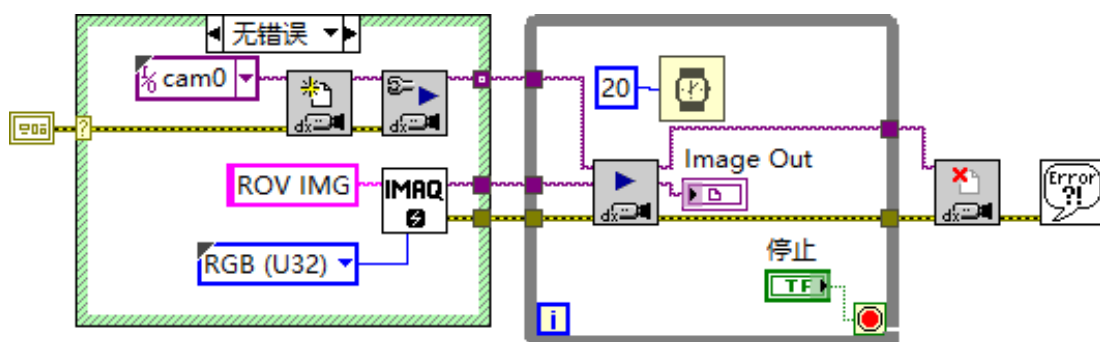
USB 摄像头数据采集

设计所用的观察级水下机器人所示用的是 USB 摄像头。将摄像头的 USB 插头与 myRIO 的 USB 接口连接。如图所示，在 LabVIEW 平台上存在一个专为视频采集的模块 NI-IMAQ，NI-IMAQ 是一款免费软件驱动。借助 NI-IMAQ 模块来配置相关设备，能够将图像采入 LabVIEW。使用 NI-IMAQ 模块可以快速的实现视频采集的任务要求。

如图所示，借助此模块，可以方便的编写出 USB 摄像头的数据采集程序。



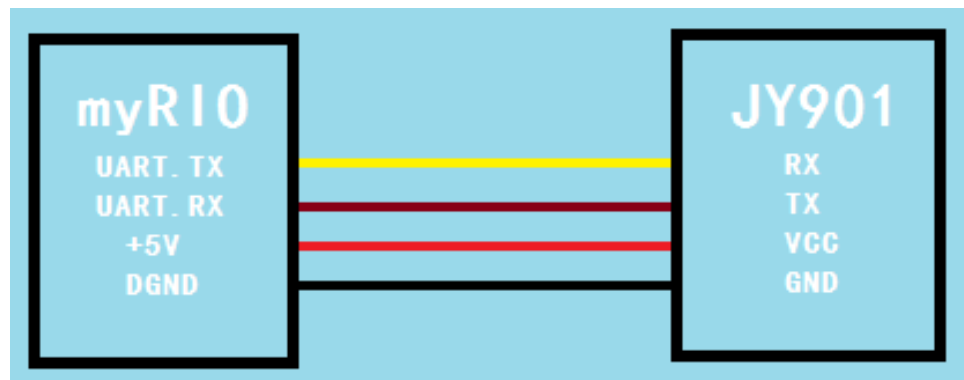
NI IMAQdx 函数面板



USB 摄像头的数据采集程序

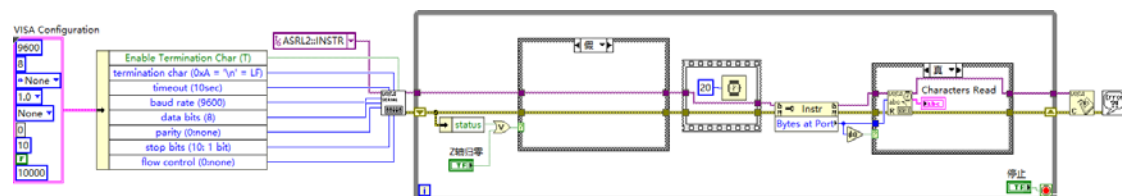
姿态传感器数据采集

设计使用的姿态传感器为 JY901。JY901 可以通过 UART 和 IIC 来和 myRIO 通讯，通过观察 JY901 的手册，发现相较于 IIC 的读取方式，UART 来读取底层数据更加方便，所以使用 myRIO 的 UART 接口与 JY901 连接。如图所示，将 JY901 与 myRIO 进行连接(注意 myRIO 与 JY901 的通讯线是反接的)。



JY901 与 myRIO 连接示意图

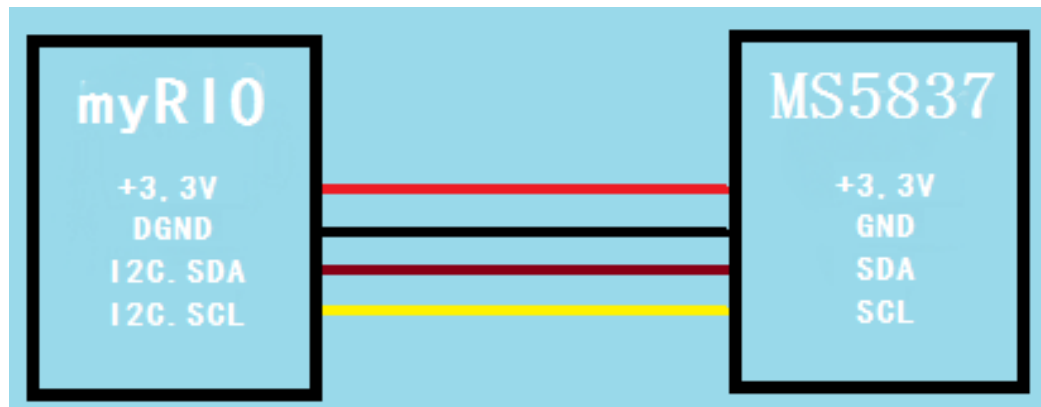
虽然 myRIO 函数模块中有专门封装好的 UART 函数，但是为了提高程序的运行效率。采用底层串口 (VISA) 函数来采集 JY901 接收到的数据。VISA 是仪器编程的标准 I/O API。使用 VISA 函数能够方便的读取到 JY901 的数据。如图所示，即为根据 JY901 的手册编写的程序。



姿态传感器的数据采集程序

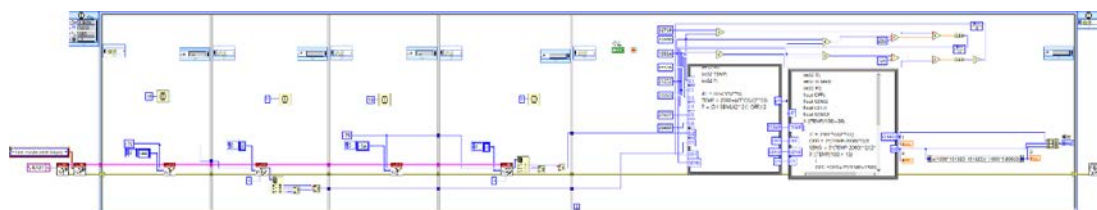
深度传感器数据采集

设计使用深度传感器其核心部分为 MS5837 压力传感器。MS5837 压力传感器是新一代的高分辨率 I2C 接口压力传感器。如图所示，使用 myRIO 的 IIC 接口与 MS5837 压力传感器进行连接。



MS5837 与 myRIO 连接示意图

与姿态传感器一样，虽然 myRIO 函数模块中有专门封装好的 I2C 函数，阅读 MS5837 压力传感器的手册发现，由于读取方式的限制，无法直接使用封装好的 I2C 函数，因此也采用底层函数的方式进行数据的采集。如图所示，即为根据 MS5837 压力传感器的手册编写的程序。



深度传感器数据采集程序

数据传输

由于 myRIO 的性能有限，直接在 myRIO 上观测和处理数据非常的缓慢，不能够达到实时控制的效果。为了避免这个缺陷，需要将 myRIO 上的采集到的底层数据传输到 PC 端，再进行数据的处理。

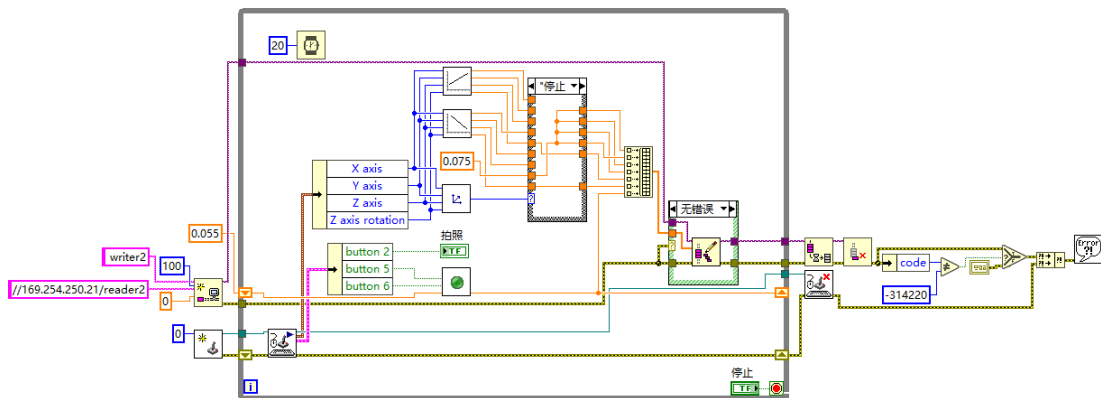
myRIO 和 PC 端之间的数据传输方式有很多种，在已联网的情况下，在两者之间发送数据，避免或丢失数据值，所广泛采用的是两种方式。两者的对比如表所示。

网络流与 TCP 参数对比

方法	属性
网络流	<p>1:1 通信</p> <p>比 TCP 易于实现</p> <p>自动处理未连接和重新连接</p> <p>单向传输</p> <p>仅在 Windows 和 RT 系统上可用。</p>
TCP	<p>广泛适用于需要网络连接至不支持网络流设备的应用</p> <p>低端协议增加了执行的复杂性</p>

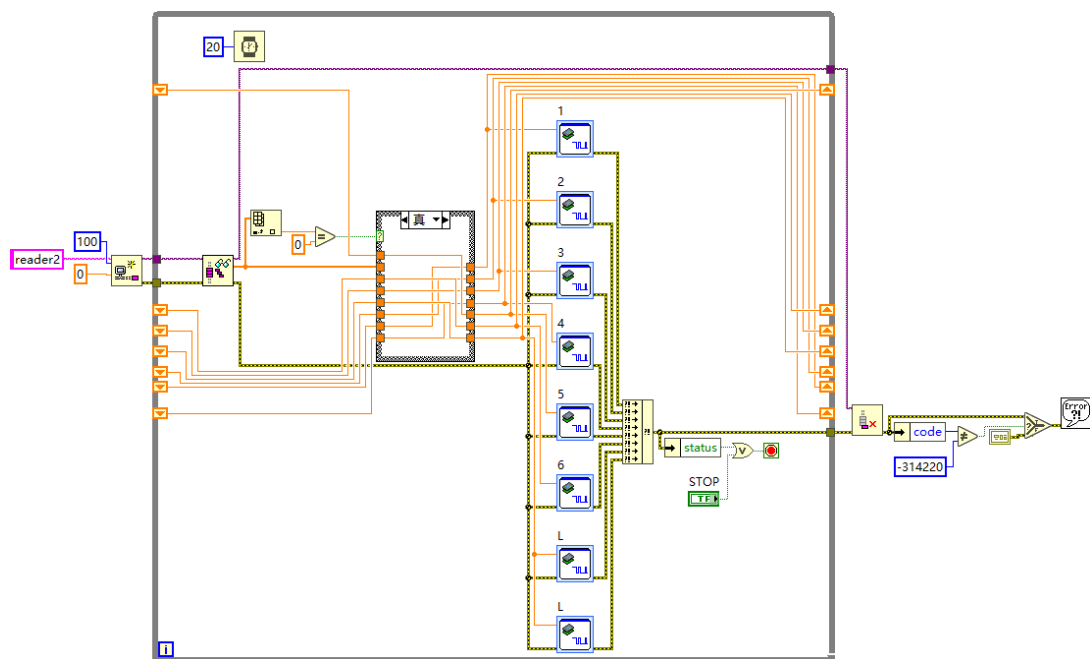
对比发现使用网络流的方式更能够是实现数据传输。因此采用网络流的方式来传输 PS4 手柄以及视频流的数据，而对于像深度传感器等一些简单的数据流则采用网络发布共享变量的方式。因为网络发布共享变量是监控值的理想选择并且用于 1:N 和 N:1 网络配置的内置连接管理器，因此在传输小型数据流方面有这很大的优势。

如图所示，根据网络流的定义方式，将 PS4 手柄数据从 PC 端向 myRIO 发送。



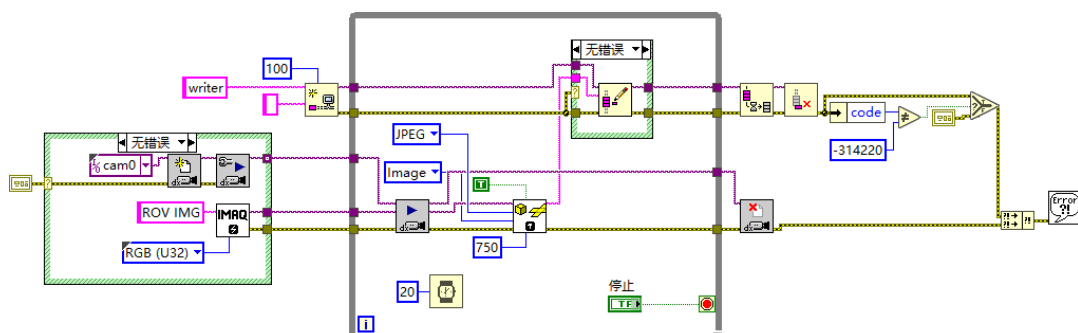
PS4 手柄数据发送端程序

如图所示，网络流由于是单向传输，所以有发送端就有接收端，此段程序表示 myRIO 接收由 PS4 发送的手柄数据。

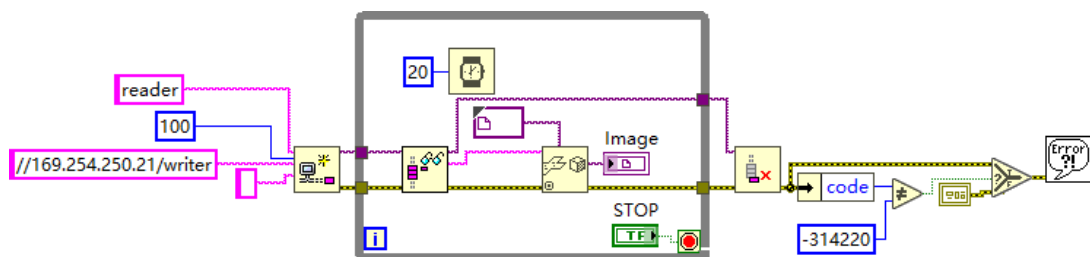


PS4 手柄数据接收端程序

如图所示，与 PS4 手柄的数据传输方式一致，将 myRIO 所采到的视频流信号传递给 PC 端。

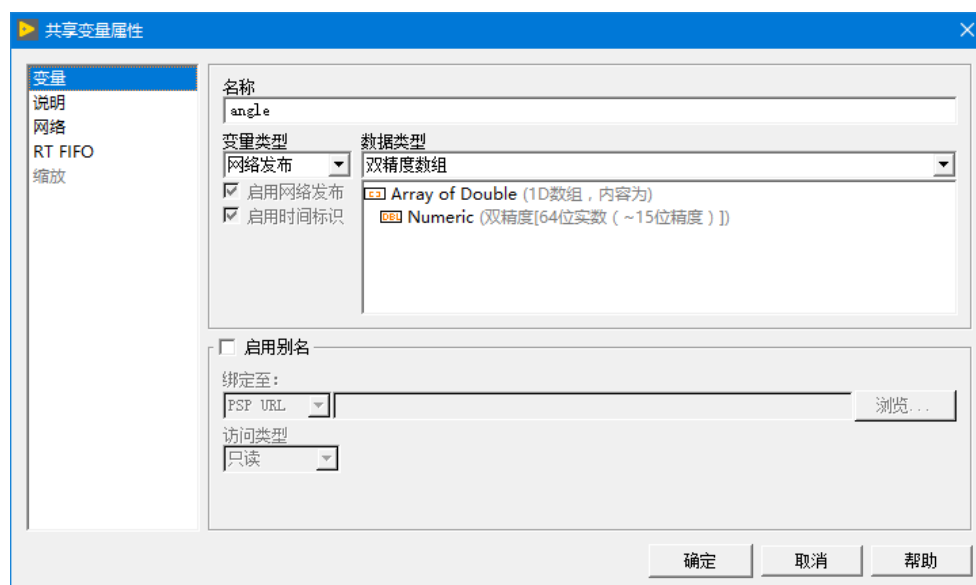


(a) 视频数据发送端程序



(b) 视频数据接收端程序

对于像深度传感器、姿态传感器灯数据流信息，采用网络发布共享变量方式，其配置界面如图所示，根据需要设定数据类型即可实现不同数据的传输。

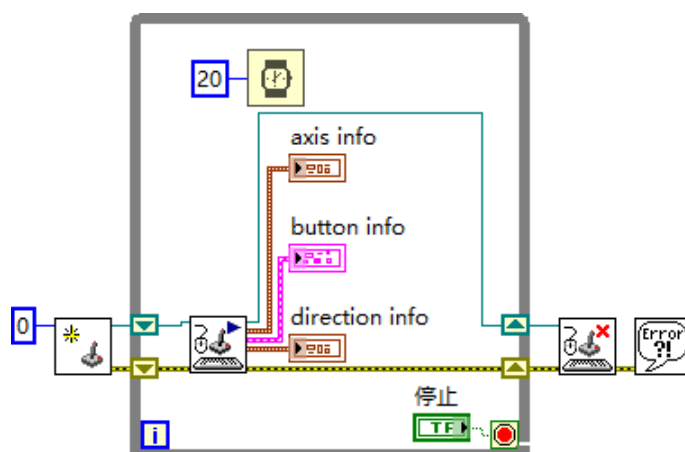


共享变量配置界面

通过这两种方式在已联网的情况下，可轻易的实现 myRIO 与 PC 端之间所有数据的传输。

手柄与 PC 的数据通讯

所使用蓝牙遥控器为 PS4。使用蓝牙作为介质与 PC 端建立连接。在 LabVIEW 中有一个输入设备控制函数模块，使用该模块能够方便的与微软系统中输入设备建立连接。如图所示，使用输入设备控制函数模块中的函数，建立 PS4 手柄与 PC 端的数据通讯。



PS4 手柄与 PC 通讯程序

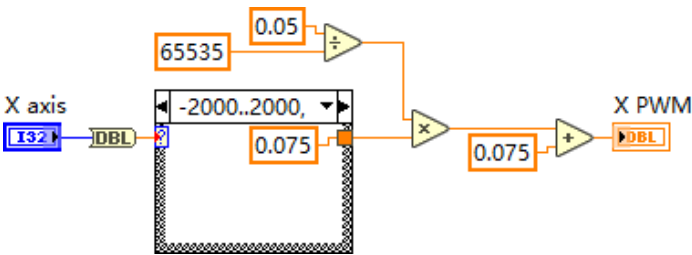
整体控制规划

已经通过输入设备控制函数模块建立了 PS4 与 PC 端的数据通讯。但是还需对手柄的数据进行进一步的细分。如图所示，将控制方式进行定义。



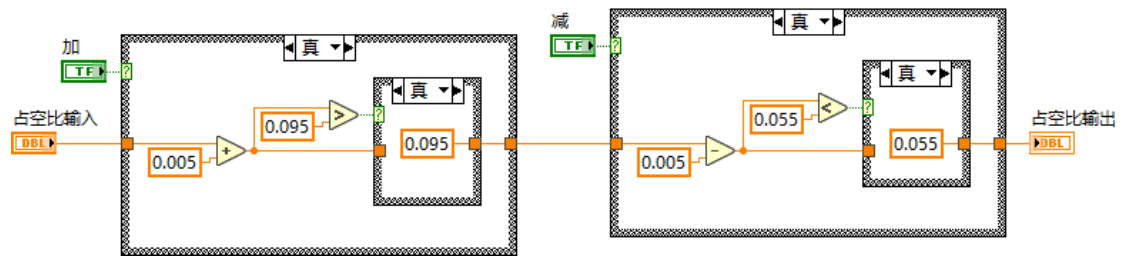
控制方式定义图

考虑到摇杆区间对应着控制动力总成的 PWM 区间。而 PWM 区间又意味着占空比的大小。即最后得到的是摇杆区间和占空比它们之间映射关系。动力总成的 PWM 区间为 $1000\sim 2000\mu s$ ，而频率为 50HZ，也就是一个周期的时间为 0.02s ($20000\mu s$)，占空比的区间即为 0.05-0.1，又摇杆区间为 $-32768\sim +32767$ 。则可列一个一元一次方程来表示摇杆区间和占空比之间的关系。即 $Y=0.05/65535*X+0.075$ 。根据此函数所得到的 VI 程序框图如图所示(为了方式误触，在程序中 $-2000\sim 2000$ 的摇杆区间设为停止状态)。



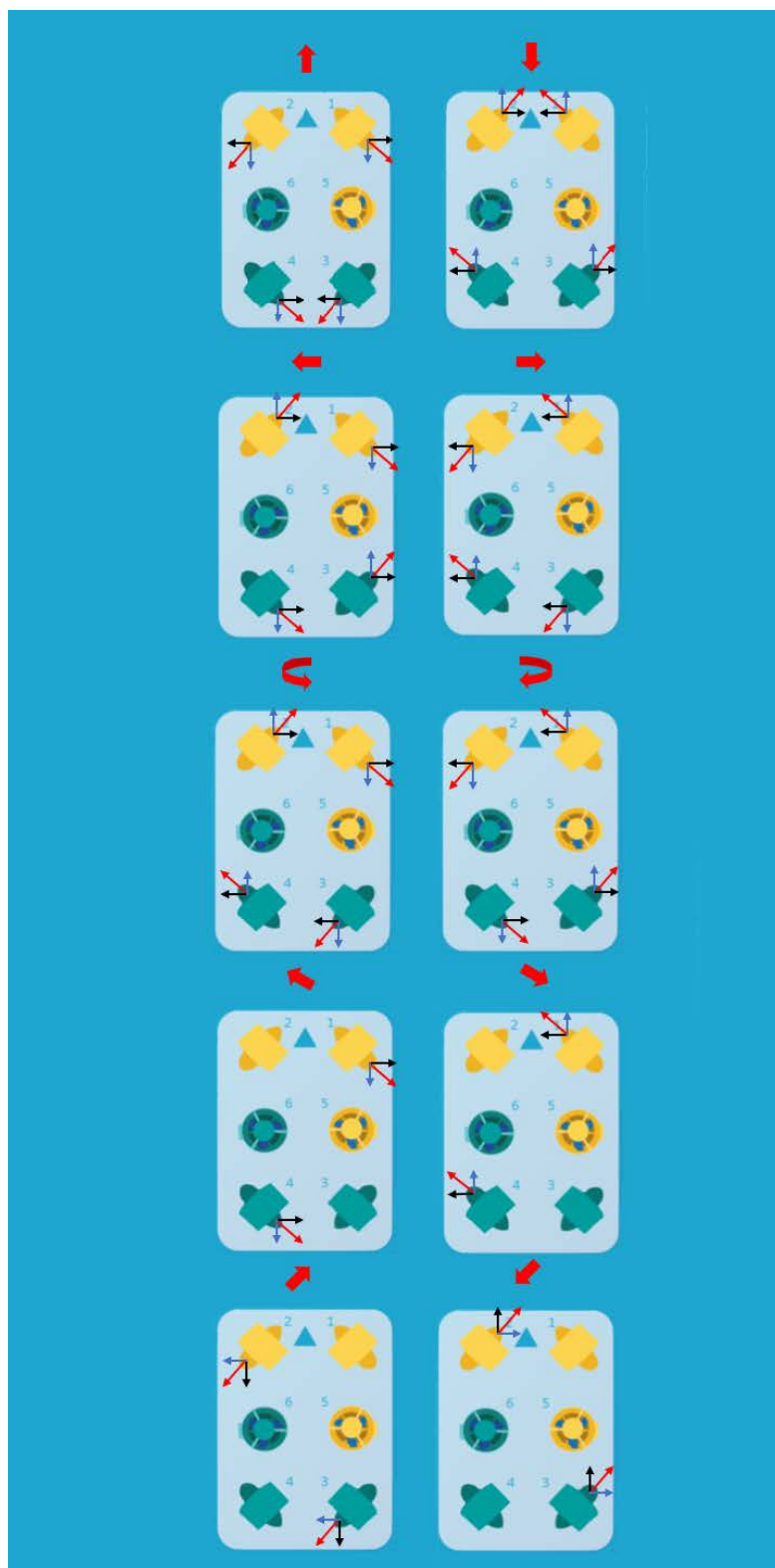
摇杆区间与占空比映射关系程序

与控制动力总成的方式相似，也使用同样的方式来控制 LED 探照灯，但是不同的是控制动力总成的方式是由摇杆区间来映射占空比，而控制 LED 探照灯的方式则采用的是按键的方式。所以如图所示，只需使用编写一个上下极限，再有按键实现对占空比的增减即可。



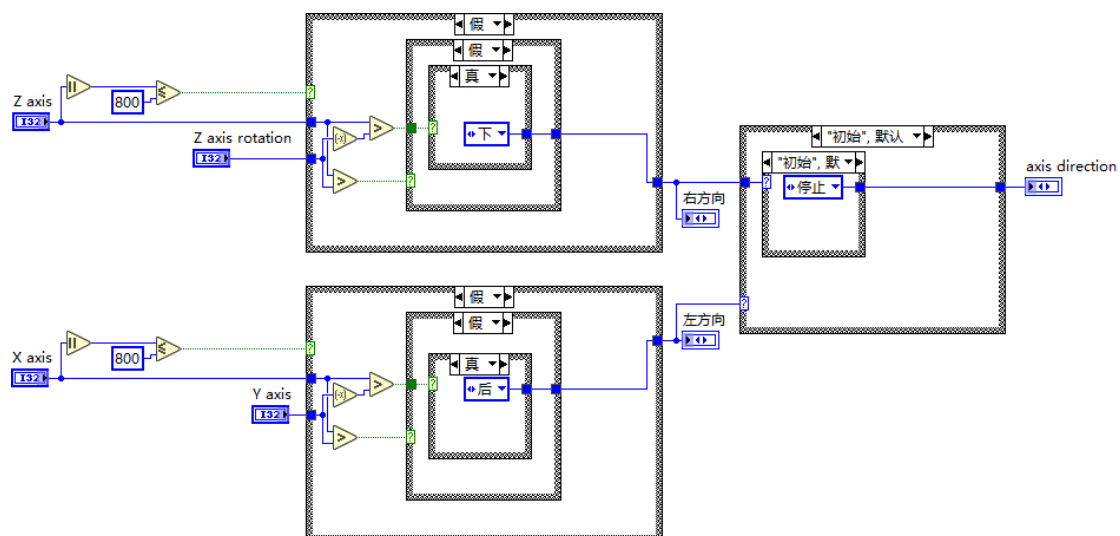
LED 灯占空比控制程序

观察级水下机器人的总体结构，通过对 6 个动力总成的控制来实现观察级水下机器人不同方向的移动。如图所示，为每种运动情况下所对应的动力总成所受力的分析。



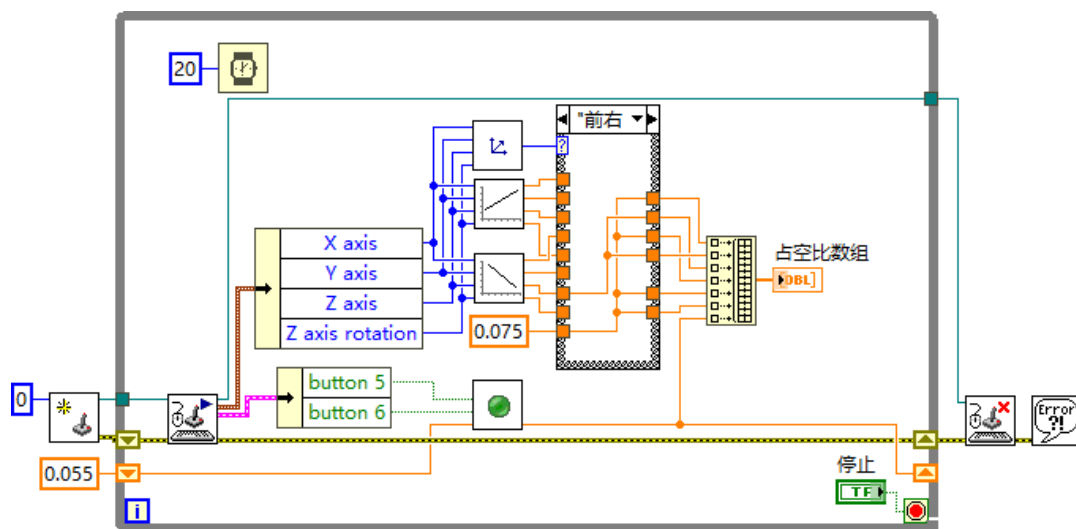
ROV 不同运动时动力总成受力分析

有了各种运动情况下动力总成的受力分析，只需根据 PS4 手柄的动作来控制不同的动力总成运动即可实现不同的运动。如图所示，首先需要对 PS4 手柄的数据进一步的细分，从而确定 PS4 手柄不同的动作所代表的动作含义。



PS4 手柄动作定义程序

确定了对摇杆区间和占空比之间的函数关系以及 PS4 手柄动作的细分，根据这两个条件，就可以确定每个 PS4 手柄动作下 6 个推力总成之间相互关系，从而编写如图所示的程序(为了方便编程，将相应的 VI 封装成子 VI，同时为了判别方便，还封装了一个相反的函数 $Y = -0.05/65535 * X + 0.075$)。

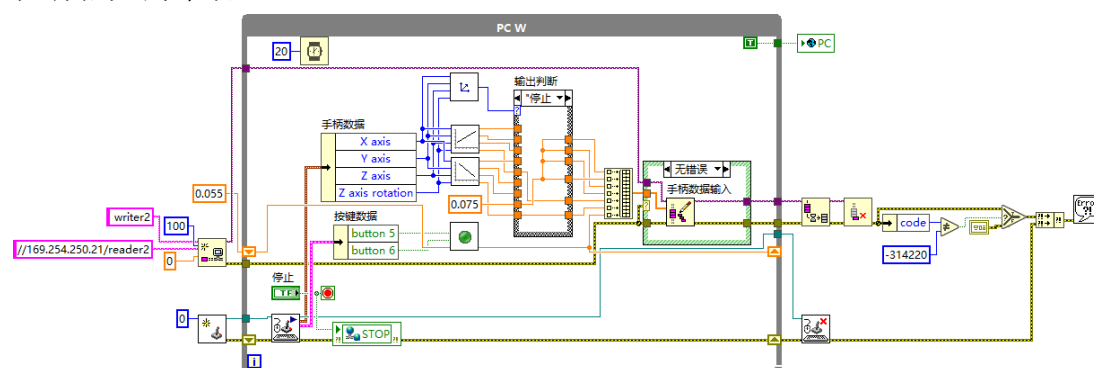


整体控制程序

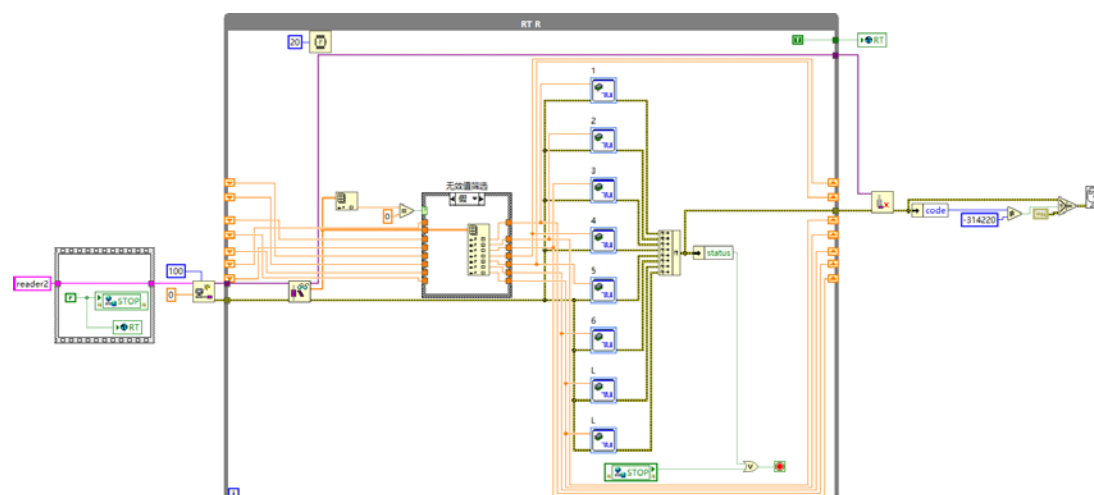
增强 Vi 的可读性

在先前的 Vi 编写过程中，只是单纯的考虑到了代码的可行性，忽略了代码的可读性，为后期的维护调试工作增加了许多的困难。因此需要对相关的代码添加一些标注以及相应的标签，方便以后的调试工作。

由于程序过多，所以只以 PS4 手柄数据在 myRIO 与 PC 端传递的程序为例子做介绍。如图所示，在关键部分添加相应的标签和介绍，对于提高程序的可读性有很大的帮助。



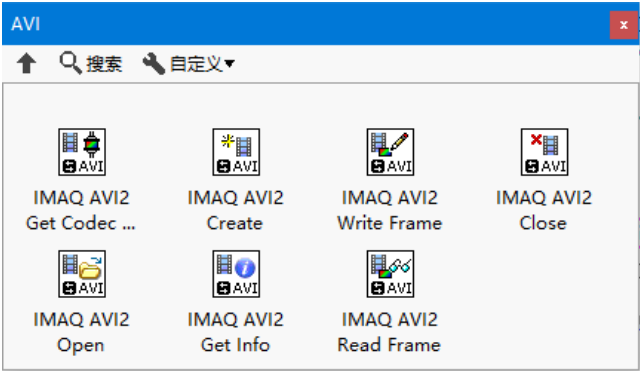
(a) PC 端可读性优化



(b) myRIO 端可读性优化

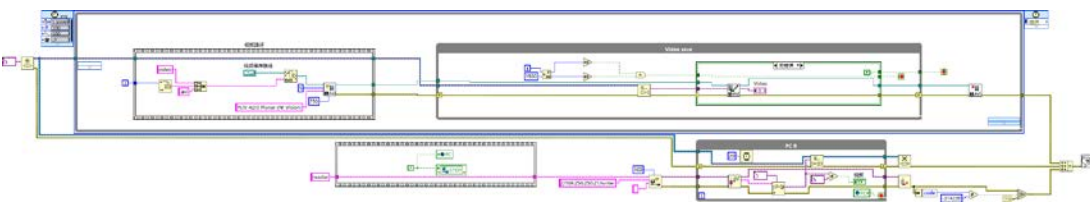
视频保存

观察级水下机器人在水下工作时，除了能够实时将摄像头的数据传递外，其中还有一项重要内容就是能够记录关键信息。如图所示，使用 LabVIEW 中的 AVI 模块，就很方便的存储摄像头中的数据。



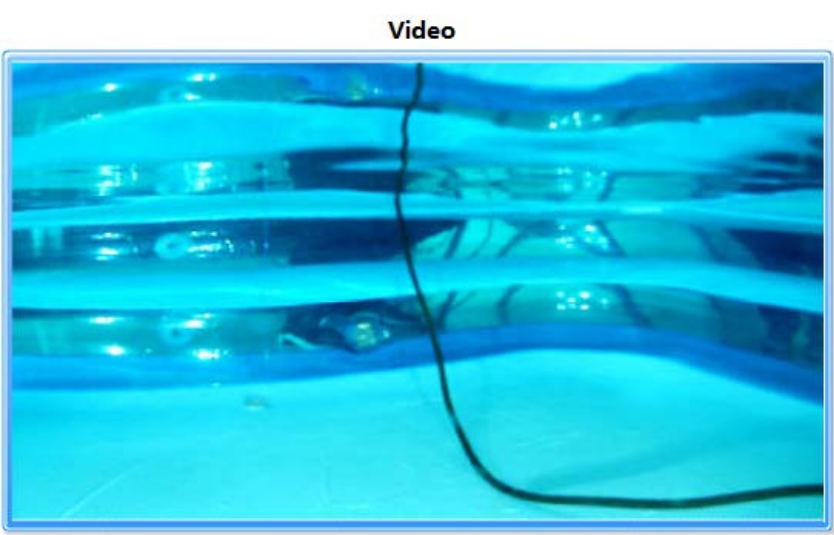
AVI 函数面板

如图所示，将从网络流中的读取到的视频流通过队列函数来传递给 AVI 文件输入端，从而实现对视频文件到保存。



视频保存程序

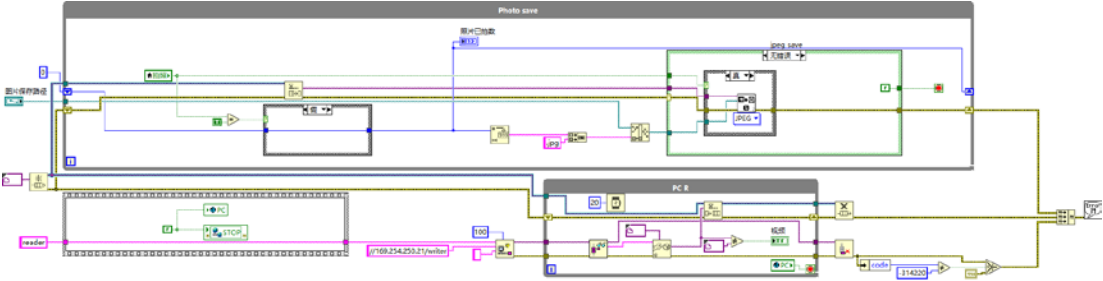
在人机交互界面上，为了能够实时显示视频信息，创建一个视频显示窗口，效果如图所示。



视频信息显示窗口

拍摄照片

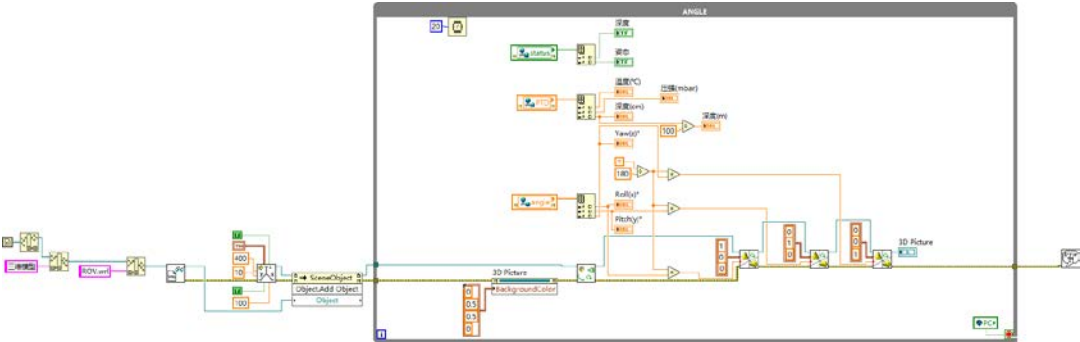
和视频保存一样，为了记录在某一时刻的信息，通常采用拍摄照片的方式。通过使用 IMAQ 函数来将需要的场景保存为 JPEG 格式。同样采用队列函数，将从网络流中的读取到的视频流中的某一时刻转换为照片，具体的程序如图所示。



图片拍摄程序

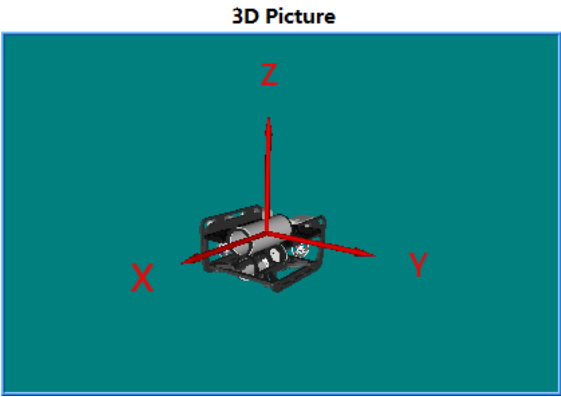
三维模型姿态演示

为了使观察级水下机器人在水下的状态能够得到及时的反馈，通过姿态传感器将相应的数据反馈到三维模型中，以更加直观的三维模型来帮助操作者了解水下机器人的实时情况。将原有的模型转换为 WRL 格式的模型，通过 LabVIEW 中的三维图片控件将具体的信息传递给模型。具体程序如图所示。



三维模型姿态演示

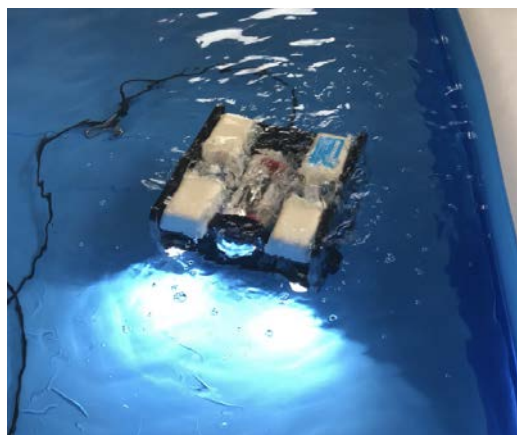
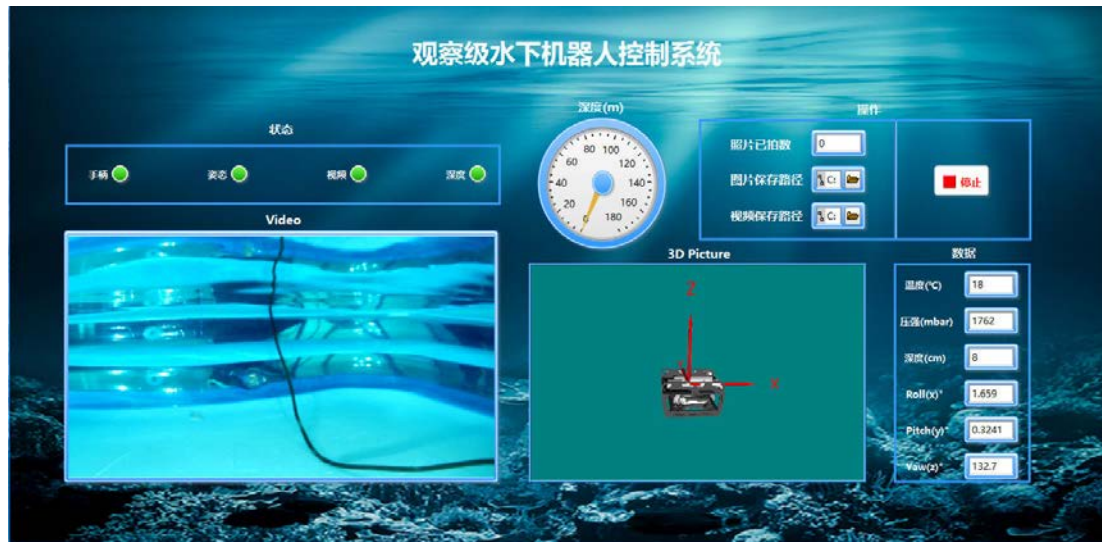
三维模型的姿态实际姿态演示效果如图所示。



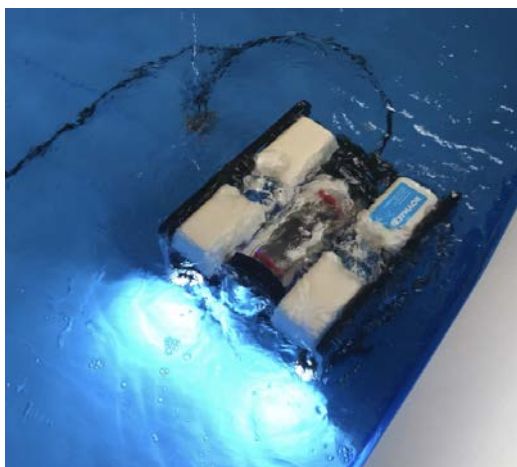
三维模型姿态演示图

美化界面

在前面的章节中已经对界面所需要的内容进行了相关的介绍，但是为了使得界面的显示效果更好，需要对界面进行一定排版和布局，添加相应的标题，同时设定一定的背景和改变的控件的风格也会使得人机交互的界面更加的美观。如图所示，为美化后的界面。



(a)



(b)