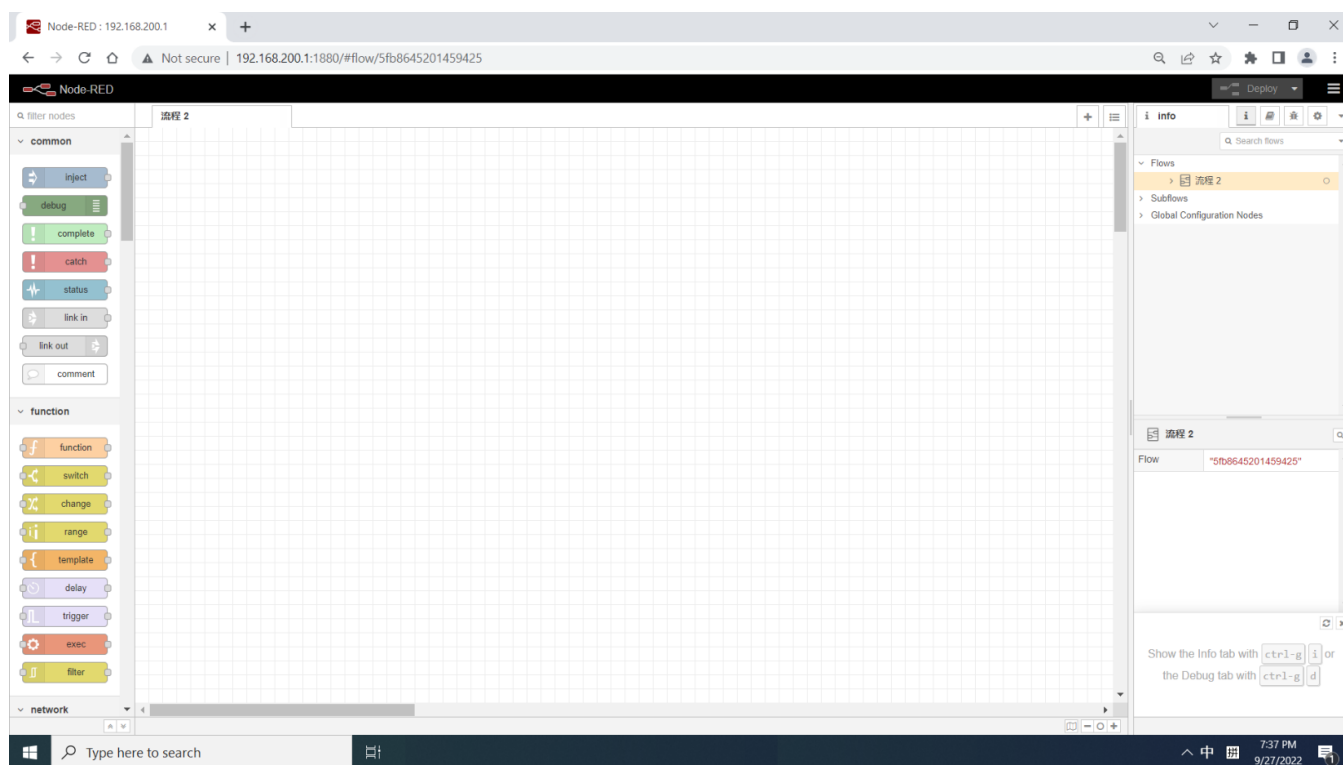


4. Node-Red 应用

Node-RED 是 IBM 开发的一款基于数据流(dataflow)的可视化编程工具，可基于浏览器的流程编辑器连接设备、服务器和 API 应用，在工业物联网和控制中已经被广泛应用。

4.1 启动 Node-Red 应用

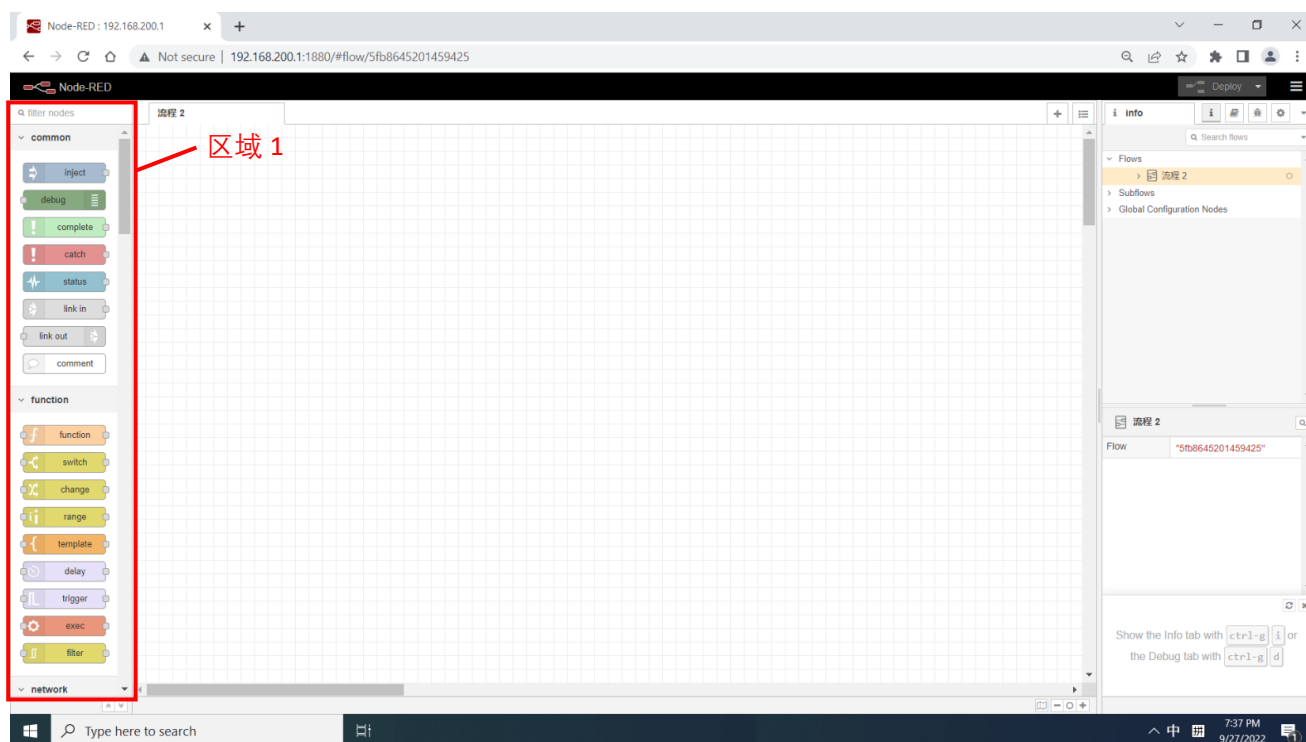
在章节【1.4 IMG 镜像文件说明】中提到过的西门子可提供的两种镜像文件中均带有 Node-RED 应用。连接好设备，IOT2050 正常启动运行后，可以通过访问网址 192.168.200.1:1880 直接访问 Node-RED 应用。



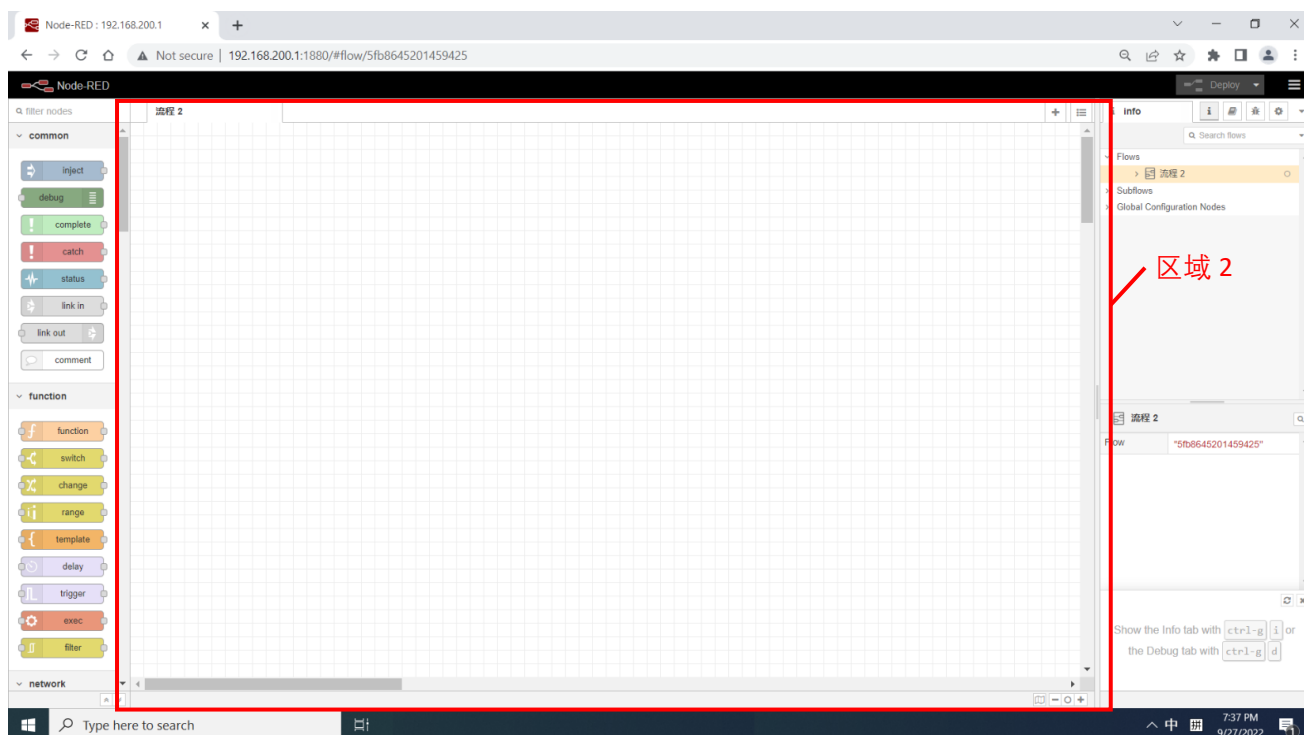
4.2 Node-Red 应用简介

Node-Red 的操作界面包含四个主要区域。区域 1 中包含了多个节点，每个节点可以实现相应的功能，这块区域为节点区。

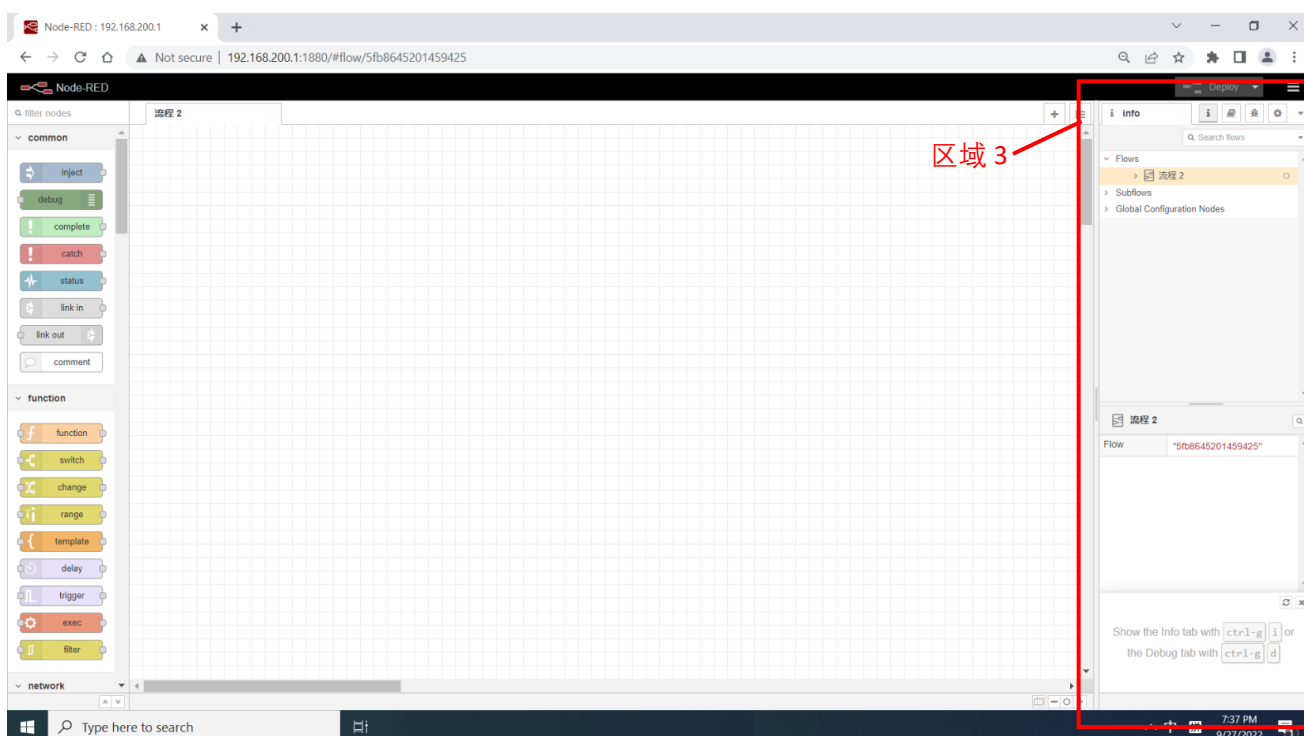
IOT2050 安装的 Node-RED 当中，已经提前为用户安装了开发时常用的不同功能类别的节点，包括通用类节点，功能类节点，协议解析类节点等等。Node-RED 提供了很多开源节点库，用户也可以根据需要自行安装或删除相关的功能节点。



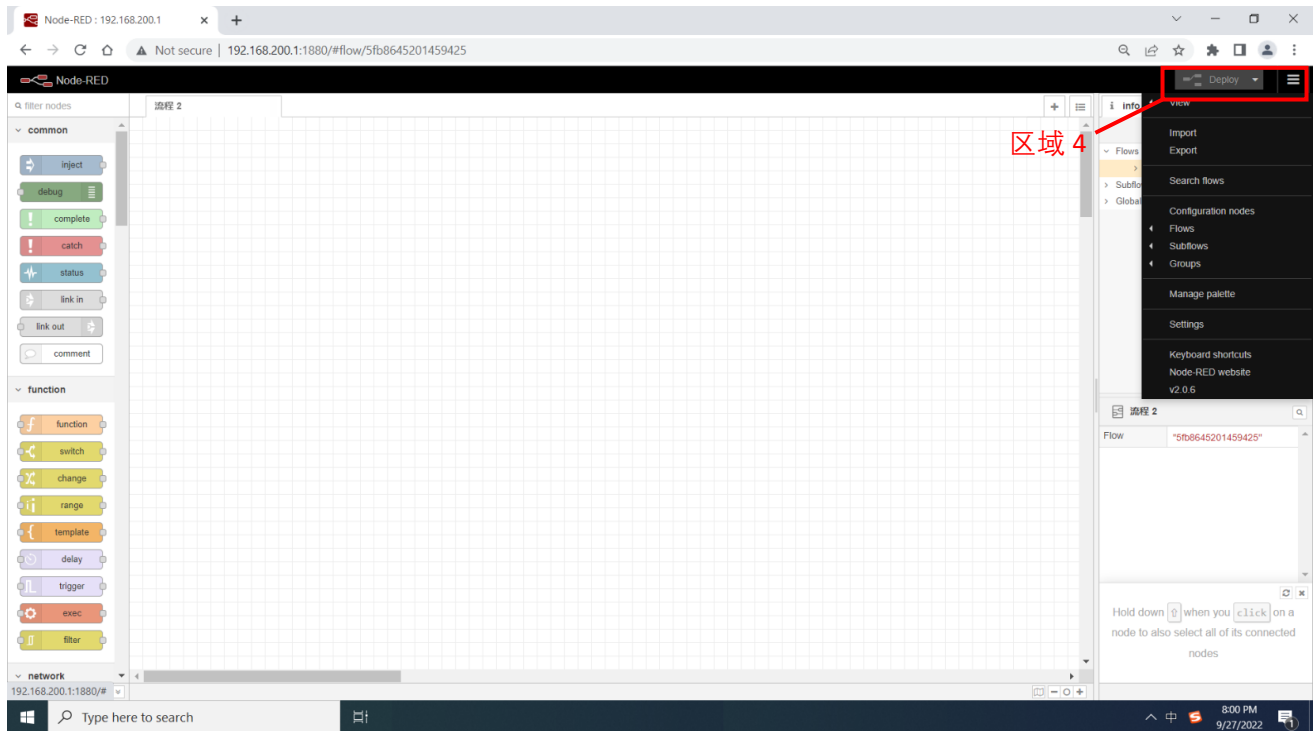
区域 2 为工作区，用户可以将节点区的节点拖拽到工作区，在工作区中对节点进行设置和编辑，就可以实现想要的功能和效果。



区域 3 为信息显示区域，在该区域中，用户可以查看节点，调试，帮助及 UI 设置的相关信息,并进行相关操作。

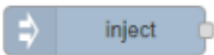


区域 4 为 Node-RED 的工具栏，通过部署按钮用户可以将编辑好的最新的程序部署到 IOT2050 中并开始执行。单机右侧的工具栏图标，会显示出下拉的功能列表，点击后可进行相关的设置和操作。



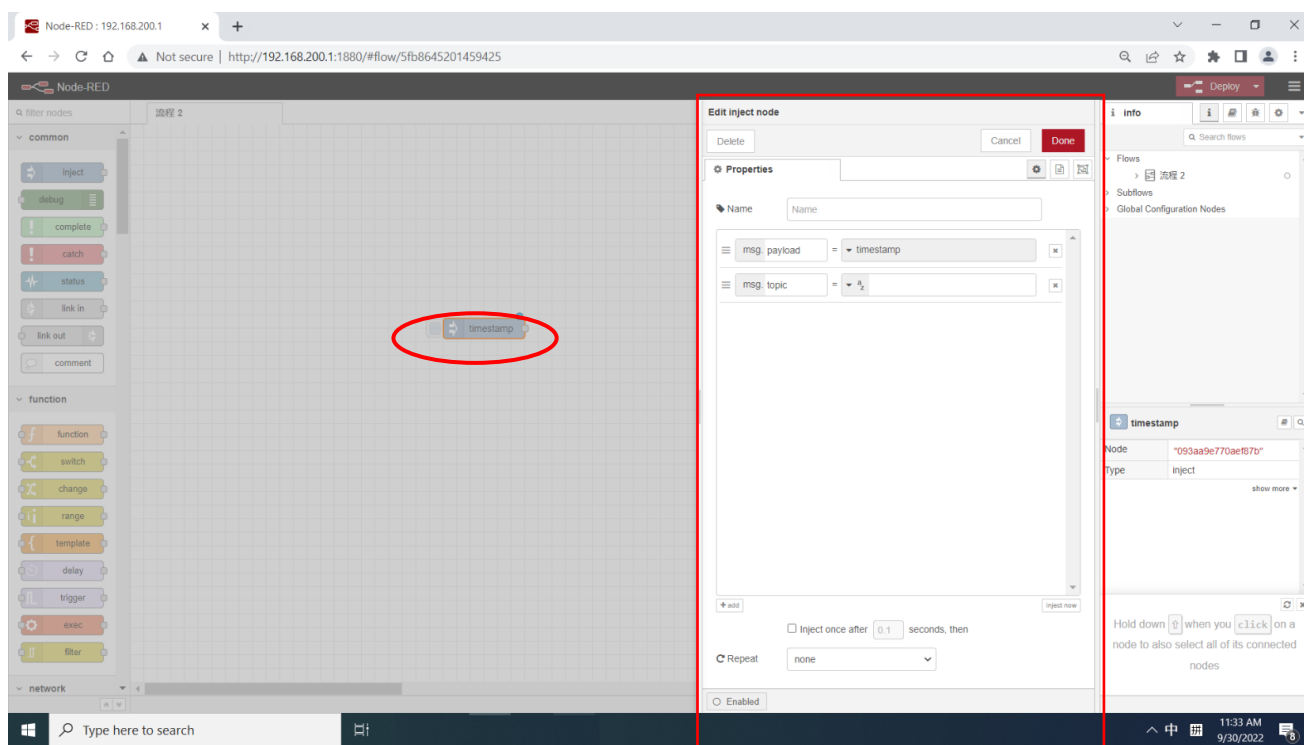
4.3 Node-Red 三个常用节点

在使用 Node-red 进行应用开发时，不同的应用会使用到各种不同功能的节点。其中有三个非常重要的节点，几乎在开发任何程序时，都会使用到，这三个节点分别是：inject 节点，debug 节点和 function 节点。

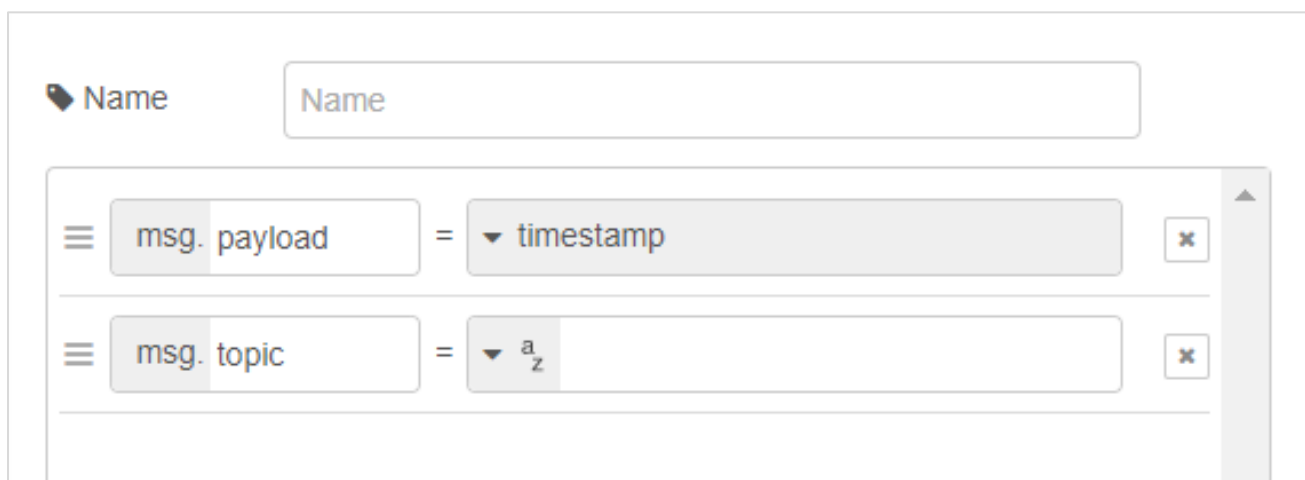
inject（时间戳）节点： 

通过 inject 节点，可以手动或定期地将消息注入到消息流中。

拖动 inject 节点到工作区域中，双击 inject 节点，就会弹出节点的设置界面。

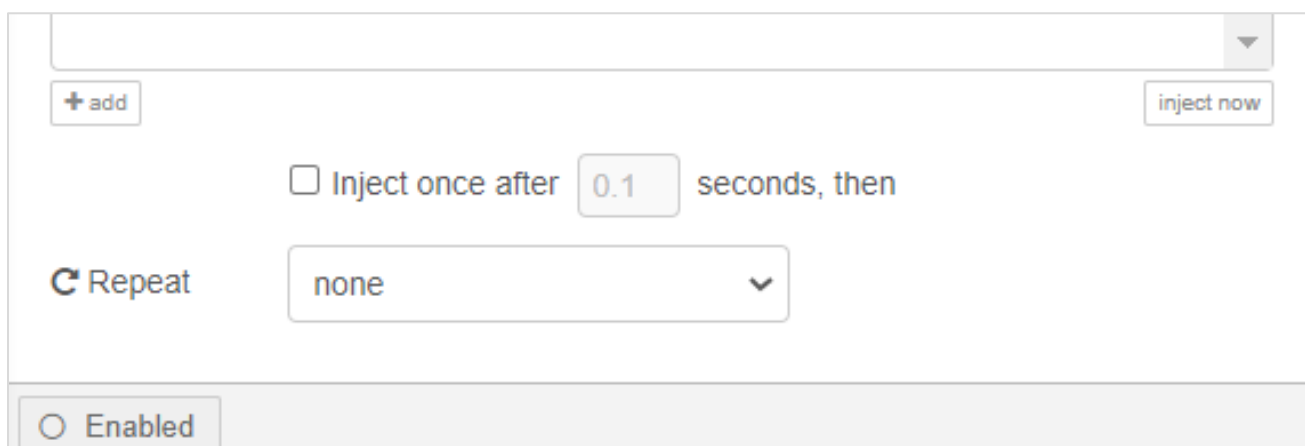


在设置界面中，可以设置节点名称，消息流的属性（默认值为 payload 和 topic），属性所对应的数据类型和数值。

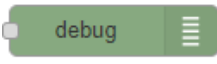


The screenshot shows a configuration window for a node. At the top, there is a 'Name' field with the text 'Name'. Below this, there is a list of message flow attributes. The first attribute is 'msg. payload' set to 'timestamp'. The second attribute is 'msg. topic' set to 'a_z'. Each attribute has a small 'x' icon to its right for removal. The interface is clean and modern with a light gray background.

也可以设置节点是否有效，数据流是否重复发送及发送周期等参数。

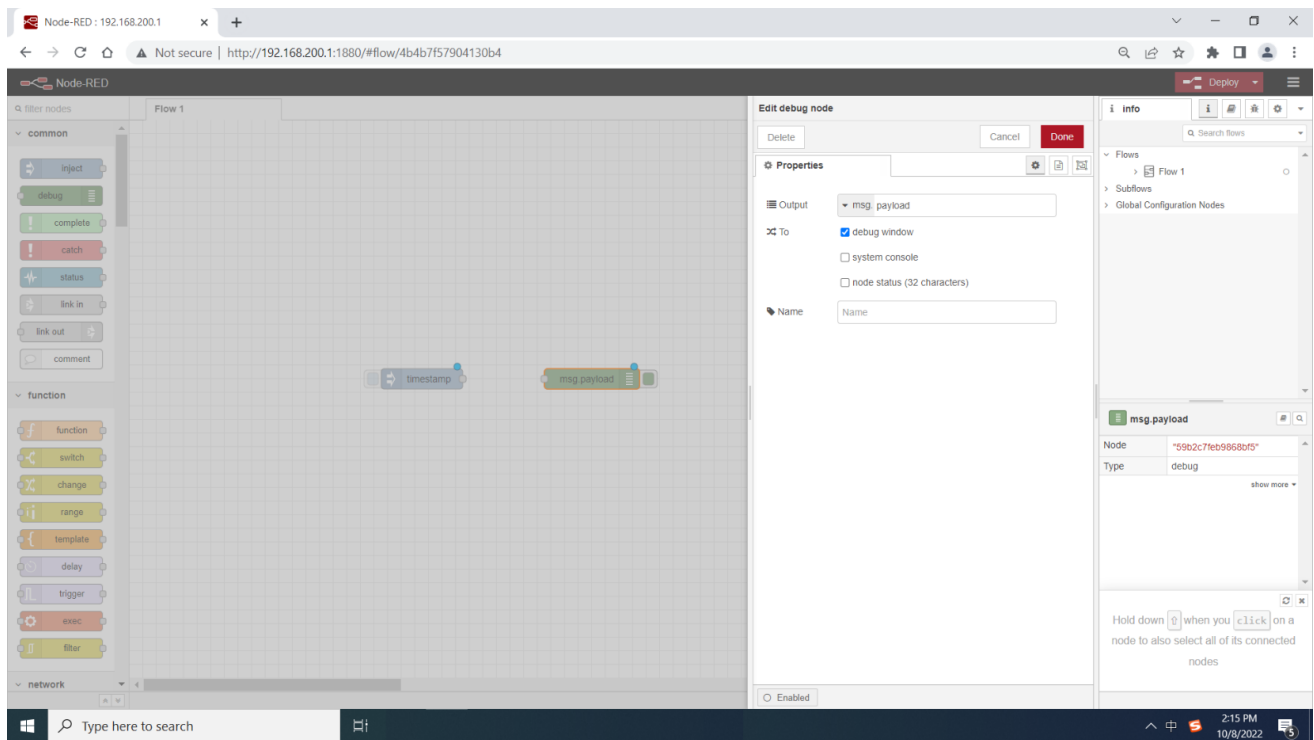


The screenshot shows the lower part of the configuration window. It includes an 'add' button and an 'inject now' button. There is a checkbox for 'Inject once after 0.1 seconds, then'. Below this is a 'Repeat' section with a dropdown menu set to 'none'. At the bottom, there is a radio button labeled 'Enabled'.

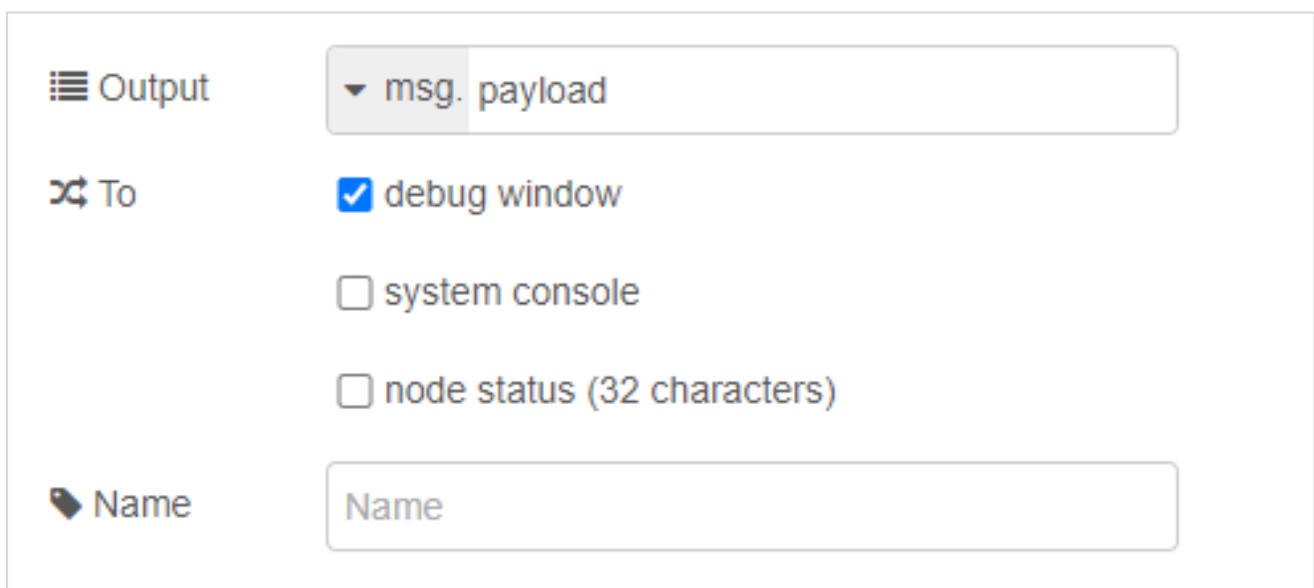
debug（调试）节点： 

通过 Debug 节点，可以在调试窗口中打印出消息流的信息，利用该节点可以十分方便的进行程序调试。

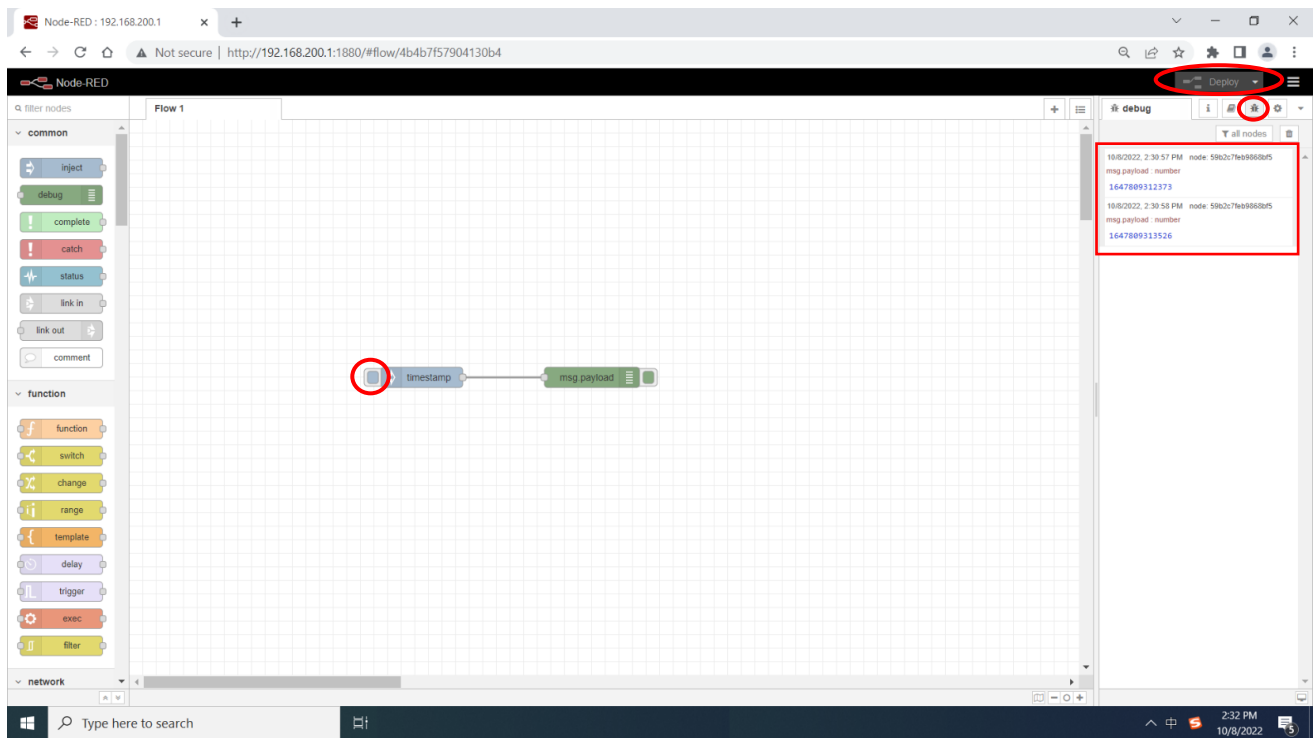
拖动 Debug 节点到工作区域中，双击 Debug 节点，就会弹出节点的设置界面。



在设置界面中，可以设置需要打印的消息流的属性（默认值为 payload），打印信息在哪个窗口显示以及设置节点的名称。



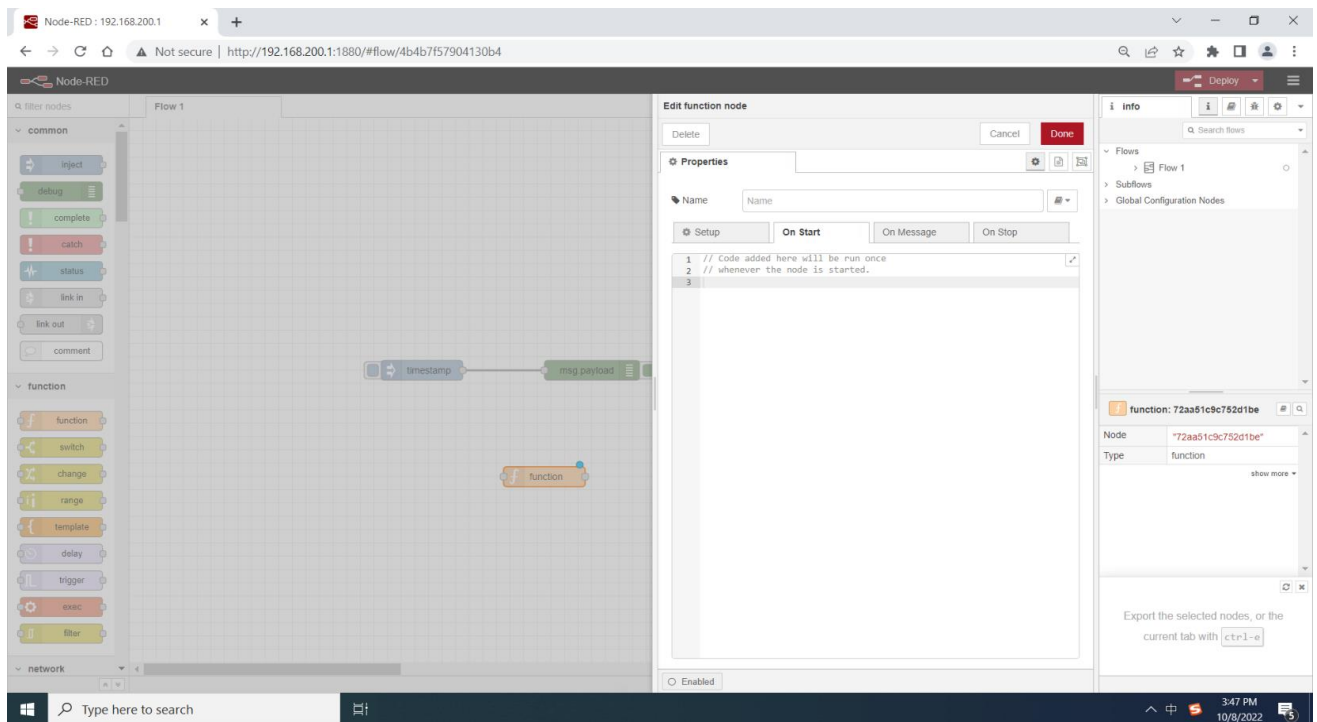
连接 inject 和 debug 节点，就可以完成一个最简单的闭合消息流，点击右上角的部署按钮完成部署。再次点击时间戳节点左侧的触发按钮，在信息显示窗口中点击调试信息按钮，就可以在调试窗口中看到打印出的消息流的信息。



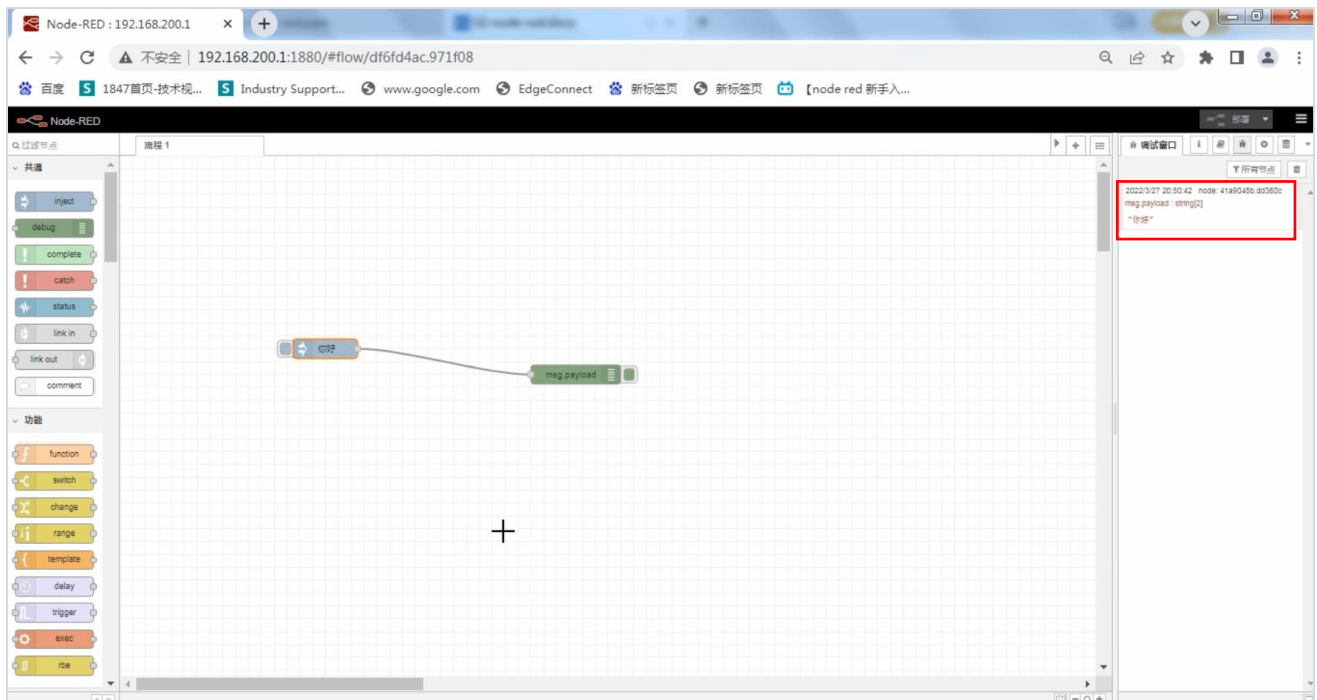
function（功能）节点： 

通过 function 节点，用户可以在节点中编辑 JavaScript 代码，可以对流入节点的消息流执行各种运算，最后可以将运算后的结果返回给消息，以继续消息流。

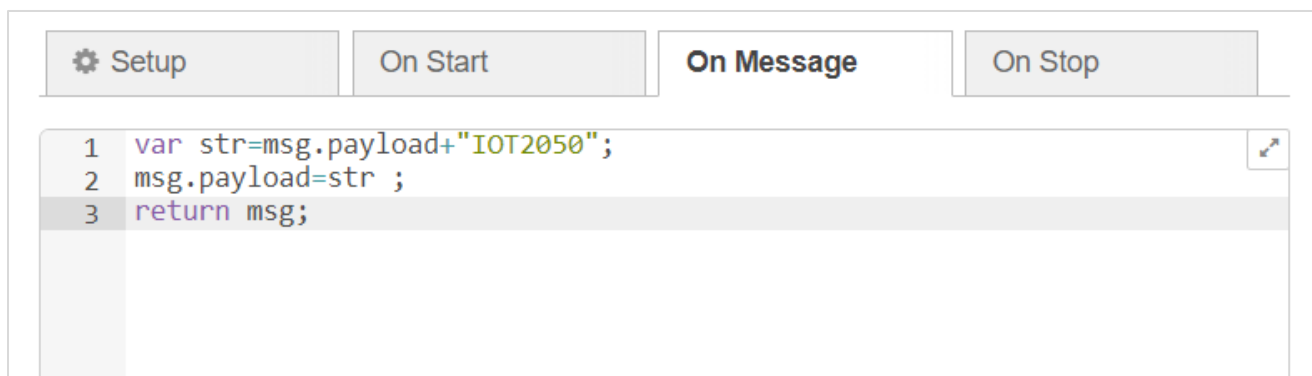
拖动 function 节点到工作区域中，双击 function 节点，就会弹出节点的设置界面。在设置界面中，用户可以设置节点的名称，**编辑**节点在不同的条件下的 JavaScript 程序代码。



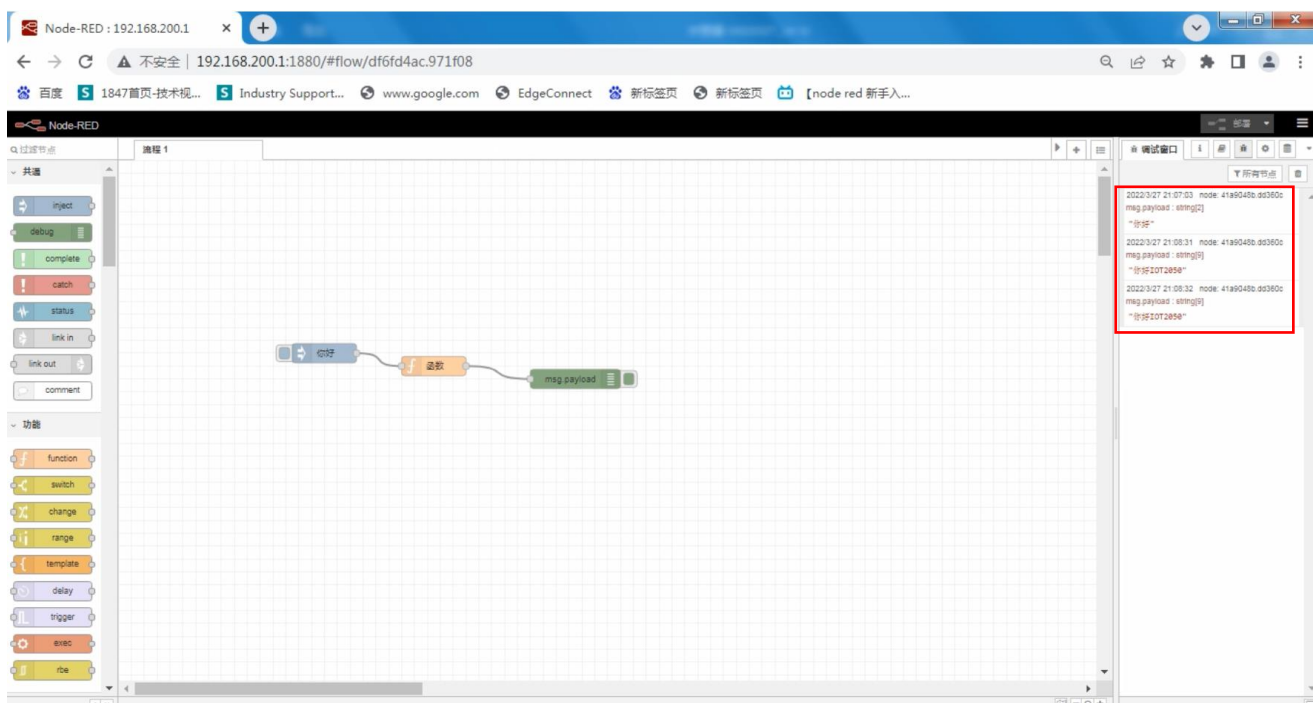
双击 inject 节点，将消息流的内容选择为文字列，输入“你好”后点击完成后部署并触发 inject 节点，可以看到输出值为“你好”。



依次连接 inject，function 和 debug 节点，在 function 节点中输入如下所示 JavaScript 代码，点击右上角的部署按钮完成部署。



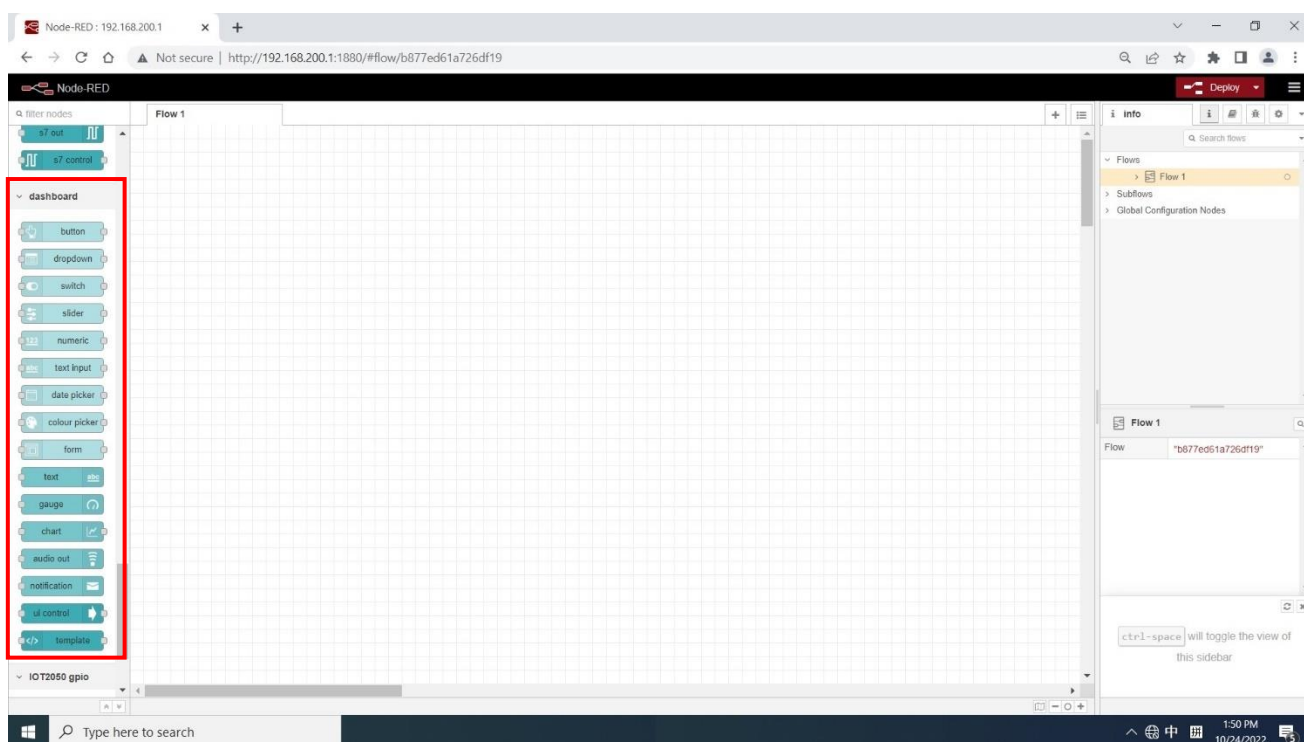
再次点击时间戳节点左侧的触发按钮，可以在调试窗口中看到流经function 节点后输出的消息信息“你好 IOT2050”。



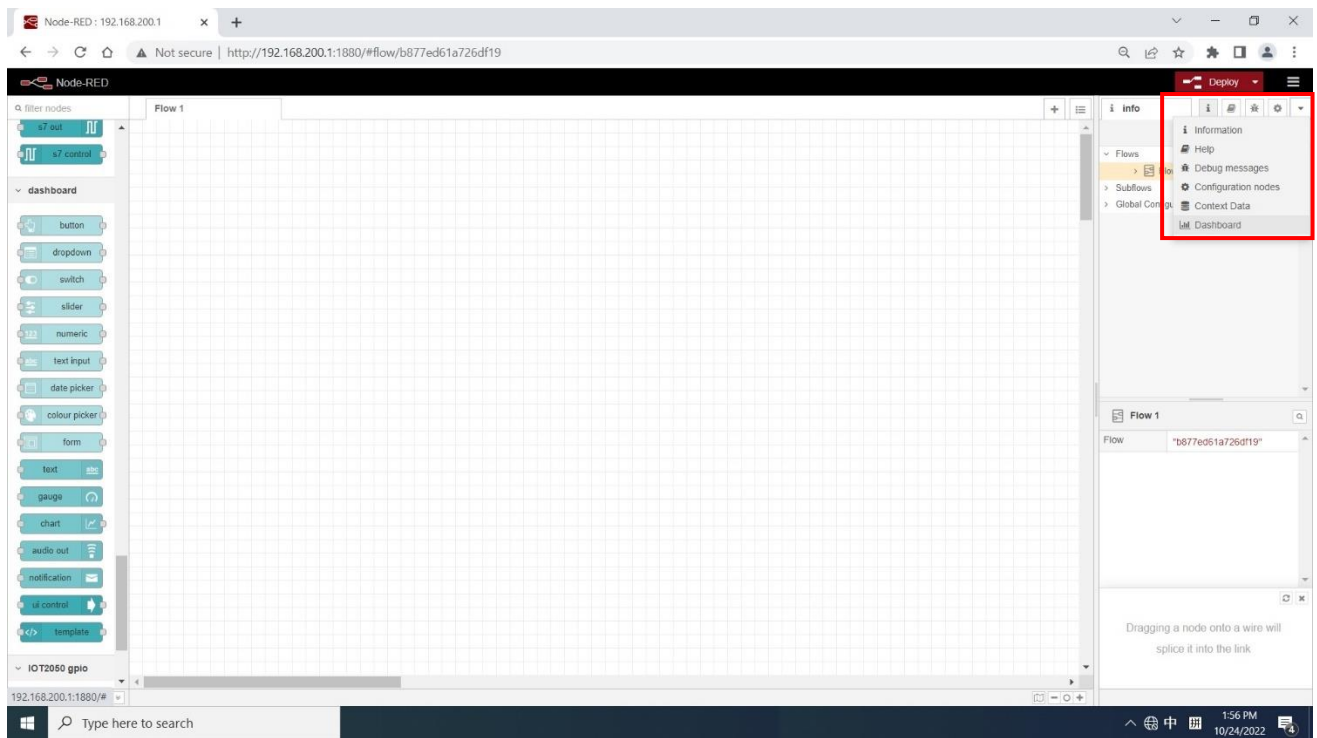
4.4 dashboard 节点

Node-red 应用中除了带有对数据进行分析，运算和处理的功能节点外，还带有能够制作 UI 界面的节点组，其中最常用的就是 dashboard 节点组。SIMATIC IOT2050 镜像中所带的 Node-red 应用中，已经提前安装好了 dashboard 节点组，用户可直接拖拽出相应功能的节点制作满足自身要求的应用界面。

向下拉动节点区域的滚动条至 dashboard 节点区域。再该区域内，可以看到很多我们在 UI 页面中经常使用会使用到的节点，例如图表，仪表盘，选项框等节点。

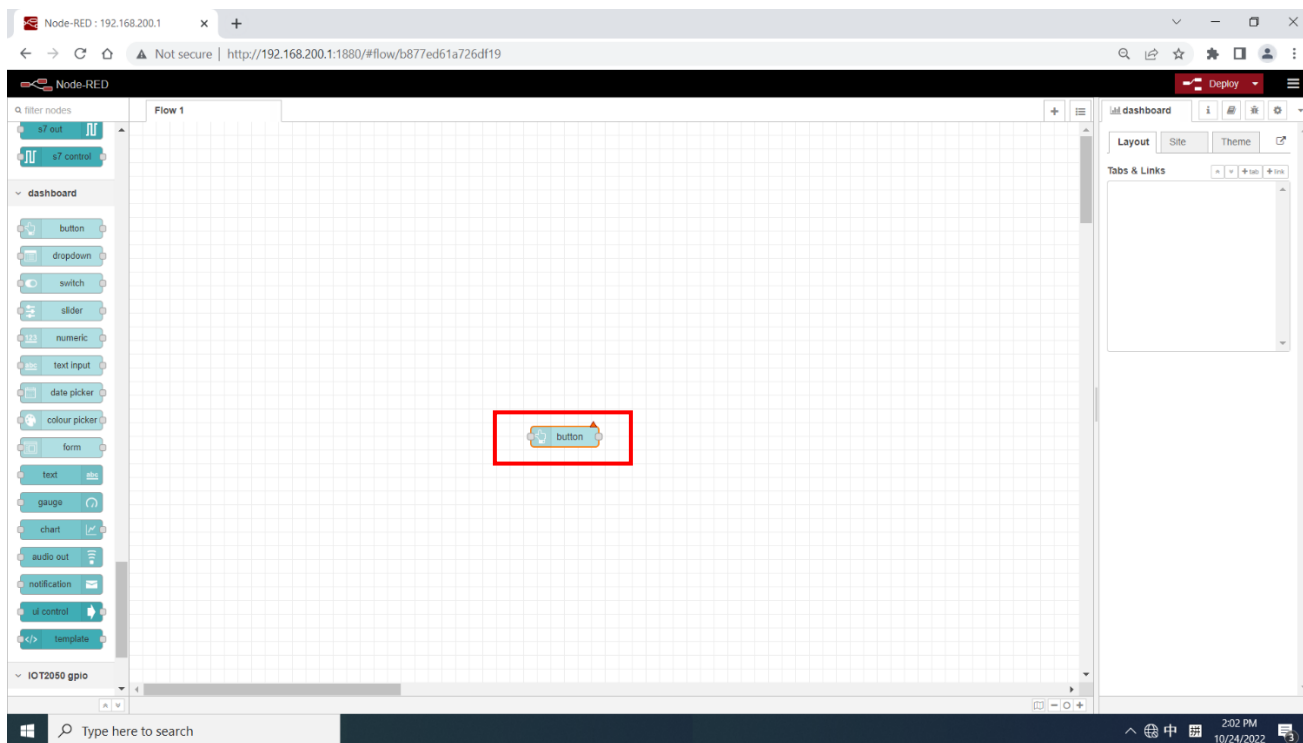


在最右侧的信息显示区，可以选择显示 dashboard 的内容，在这里可以设置 dashboard 中的节点布局，主体，色彩方案等内容。



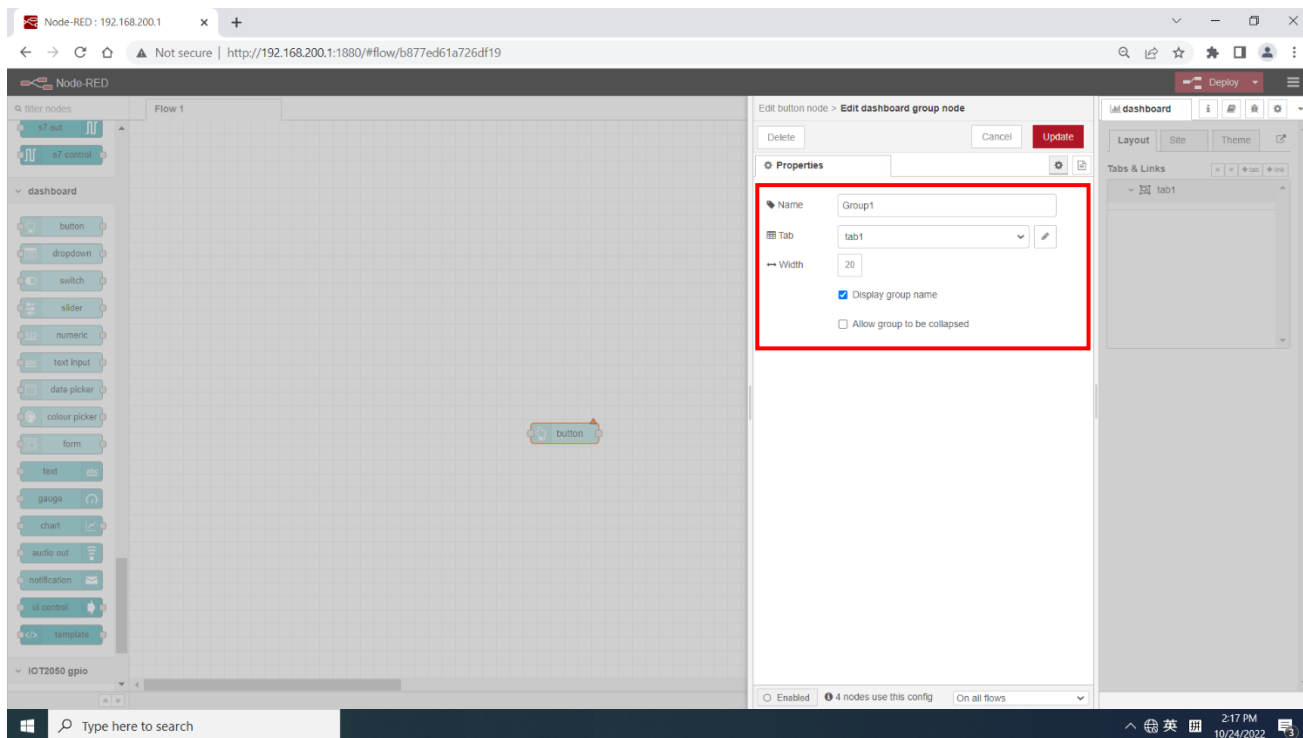
4.4.1 button 节点应用

拖拽一个 button 节点到工作区中，此时该节点上会出现一个红色的三角标志，该标志表明节点缺少相关的参数，需要进行设置。

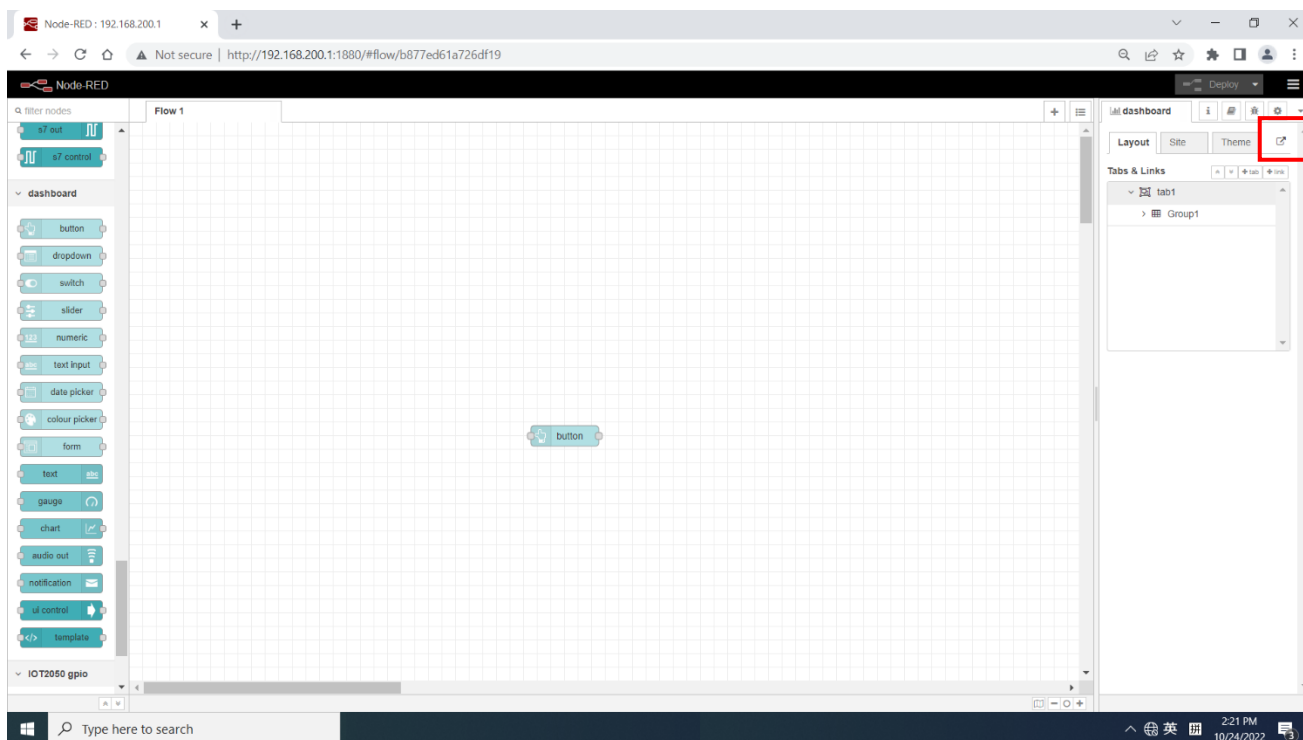


双击 button 节点，在弹出的属性窗口中可以根据自身需要更改 button 节点的对应属性。所有节点都需要被分配在相应的 tab 页面和 group 分组当中的，也可以根据节点的应用分类，将节点设置到不同的 group 分组当中。

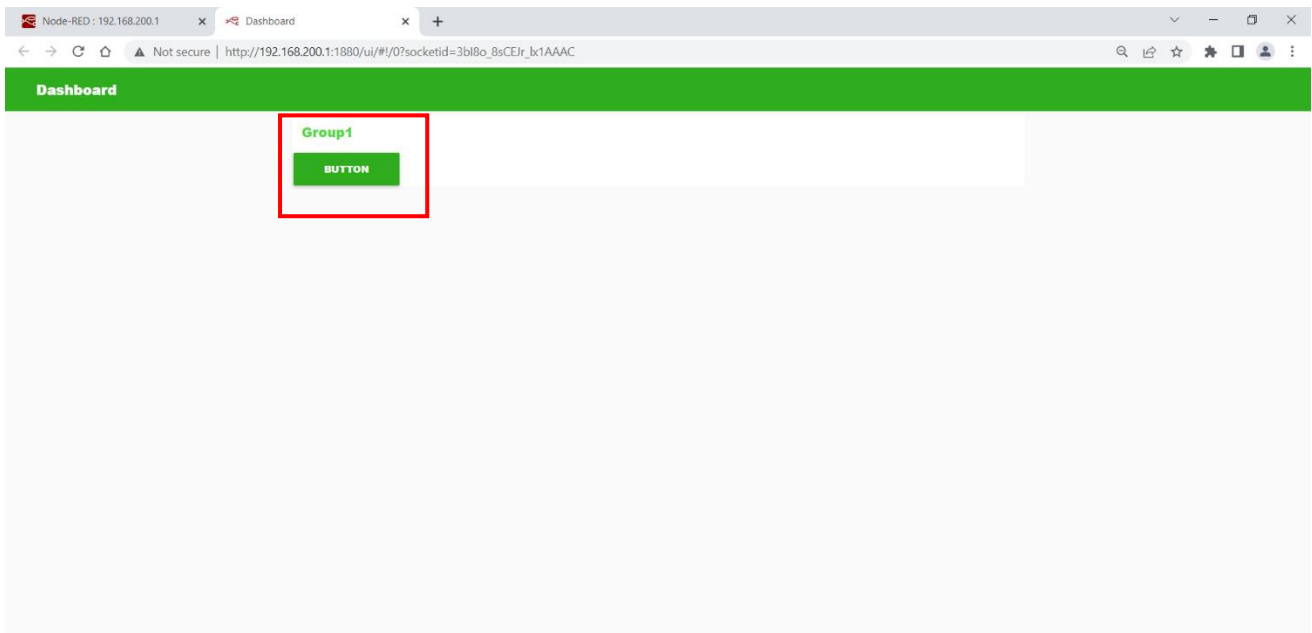
双击 button 节点，在弹出的属性窗口中可以根据自身需要更改 button 节点的对应属性。在 group 下拉列表框中选择“添加新的 ui_group 节点”，点击右侧的按钮，进入 group 属性页面，在这里可以更改 group 的名称为 group1。在 tab 列表框中选择添加新的 ui_tab 节点，点击右侧按钮进入 tab 属性页，name 栏中输入 tab1。



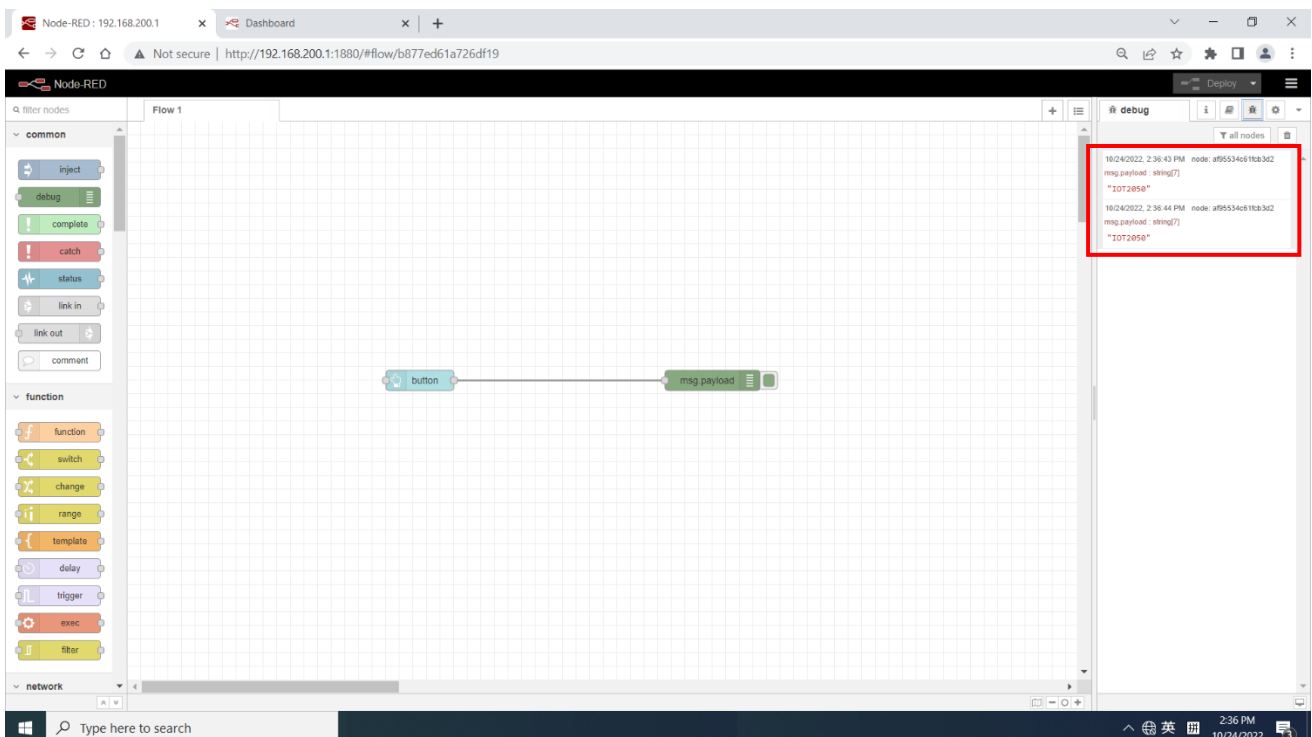
依次点击右上角的红色添加按钮和完成按钮，完成按钮的部署。再次点击右上角 dashboard 的 UI 页面链接图标



此时就可以在 UI 页面中看到我们部署在 tab1 窗口下，group1 分组中的 button 节点。

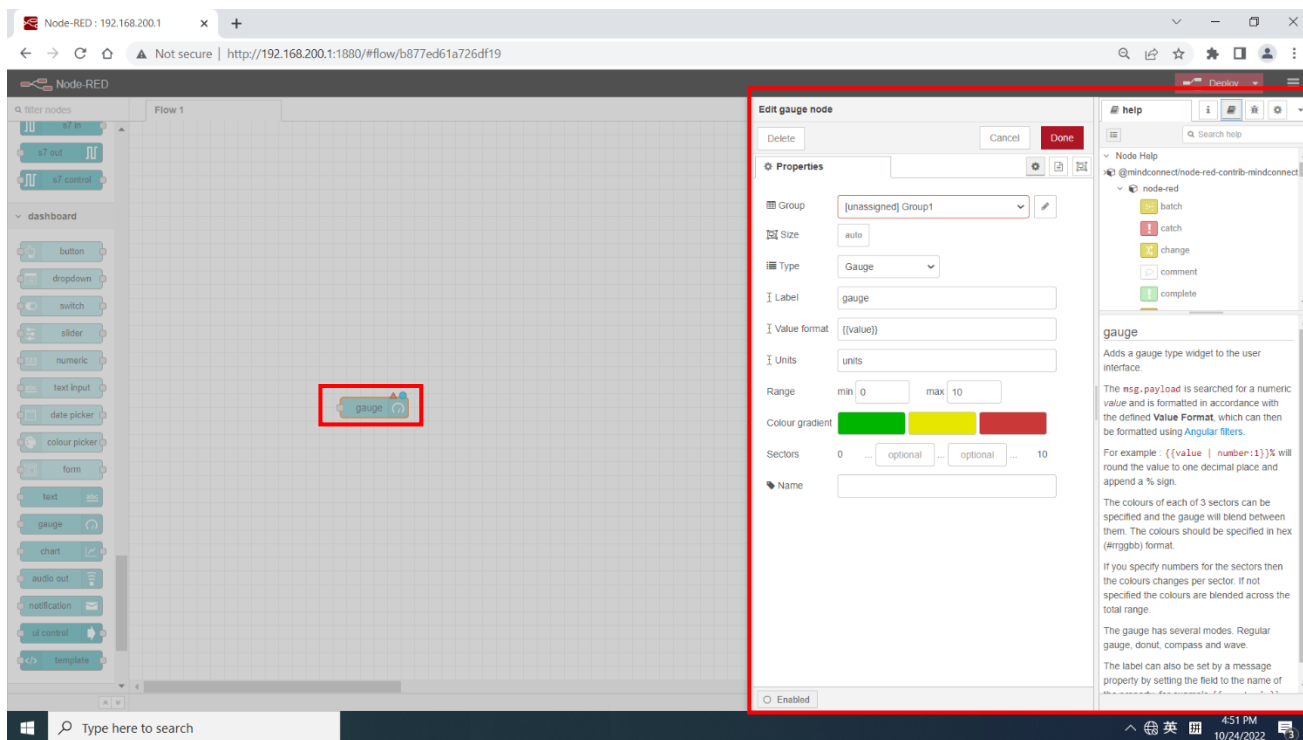


返回应用页面，将 debug 节点拖拽到工作区域中，连接 button 节点和 debug 节点，双击 button 节点，在 payload 下拉列表框中选择输出类型为文字列，内容输入“IOT2050”，点击完成后部署应用。切换至 UI 页面点击按钮后，返回调试窗口，此时可以看到设置的按钮输出内容，已经被打印到调试窗口中了。



4.4.2 gauge 节点应用

拖拽一个 gauge 节点到工作区域，双击 gauge 节点，可以在弹出的属性窗口中设置 gauge 节点的参数。从右侧的帮助栏中也可以查看 gauge 节点对属性性的相关说明。



可在属性设置窗口对节点进行设置：

Size 选项：可以设置 gauge 的大小，gauge 的最大宽度由所设置的 group 的宽度所决定。

type 选项：可以选择不同 gauge 的样式类型，用户可以根据需要选择使用合适的 gauge 样式。

Label 选项：设定 gauge 节点的标签，更改后，UI 页面中节点的标签会被更改为设置的内容。在没有设置 name 选项的情况下，工作区域中节点标签的名称也会被更改。

Value format : gauge 节点数据区域所显示的数值，默认情况下显示 gauge 节点接收到的数值，用户可以将其删除或更改为其他内容，使用时该选项一般会保持默认。

Range, color gradient 和 Sectors 选项 : 只有在节点的样式类型被选择为 Gauge 和 Dount 时才会出现。三个选项配合使用可以在节点接收到不同范围的数值时显示不同的色彩效果。

Range 选项 : 设置 gauge 节点中所能显示数值的最大最小值范围，当此处被设置时，Sectors 选项中的起始和终止数值也会发生相应变化。

sectors 选项 : 将 gauge 节点的显示数值进行分段，最多可设置为 3 个不同的数值区域。

color gradient 选项 : 为不同的数值区域设置不同的显示颜色。

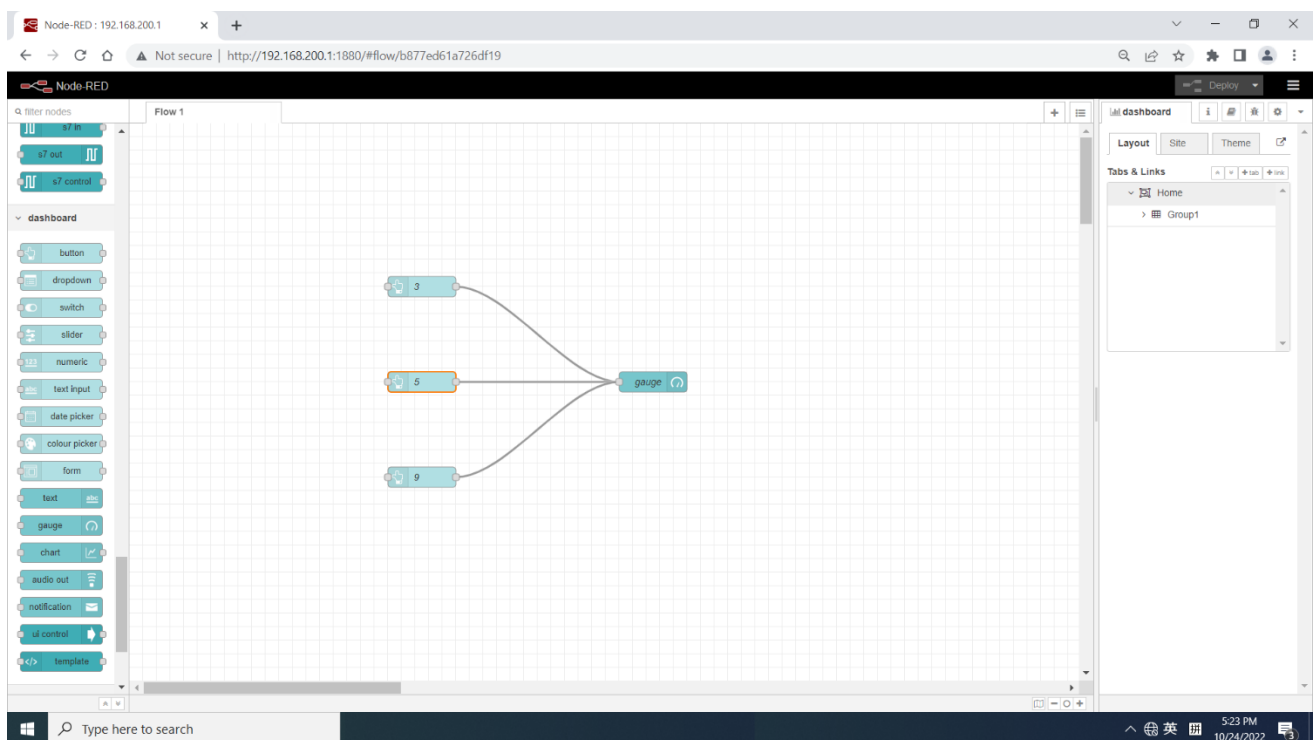
Name 选项 : 设置 gauge 节点在工作区域中所显示的名称，可以看到，当更改 name 选项的设置后，工作区域中 gauge 节点标签的名称也会随之变换。

最后，还可以通过属性页中最下面的按钮来选择节点是否有效，如果选择无效，则整个节点的部分显示为虚线，UI 界面中也不会再出现设置为无效的 gauge 节点的内容

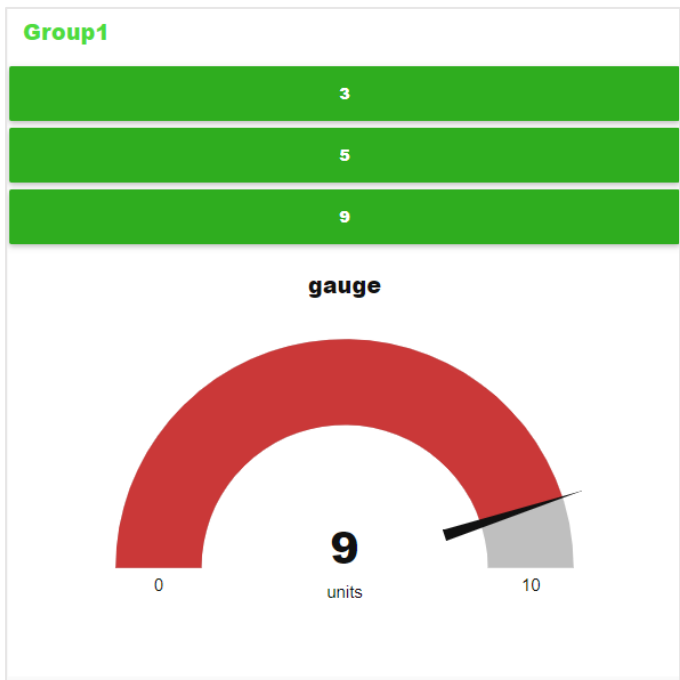
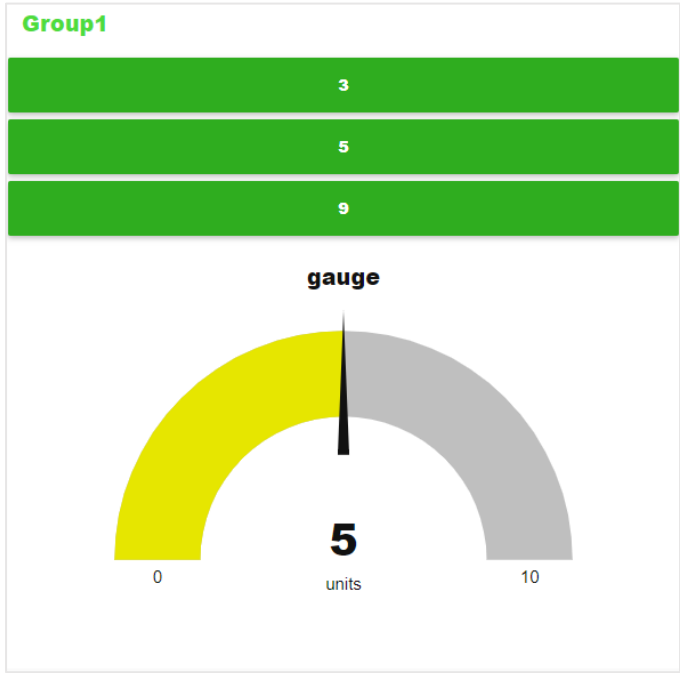
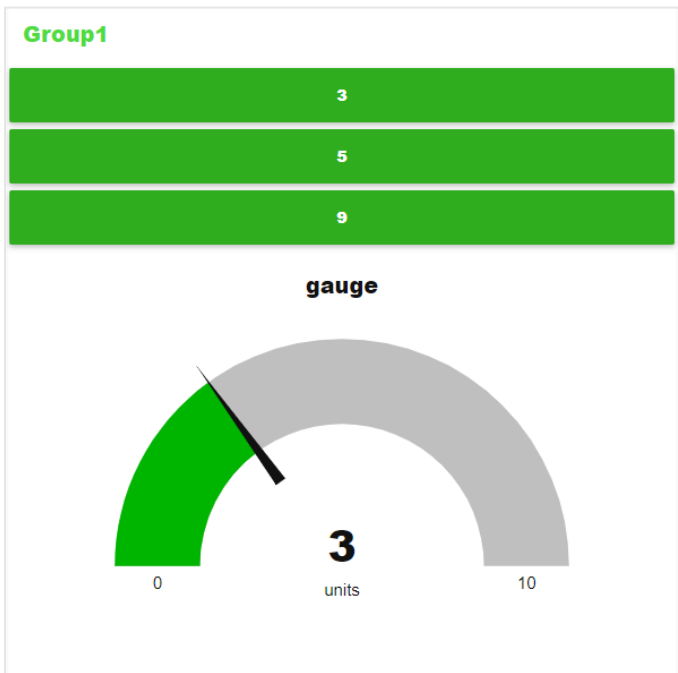
以下图中所示为例对各项参数进行设置后设置。



分别将 3 个按钮和一个 gauge 节点拖动至工作区，双击第一个按钮，设置分组，将 label 选项和 payload 选项都设置为数值 3，使用同样的方法，将另外两个按钮的 label 和 payload 选项分别设置成 5 和 9。将三个按钮节点分别与 gauge 节点进行连接后，点击部署按钮部署设置。



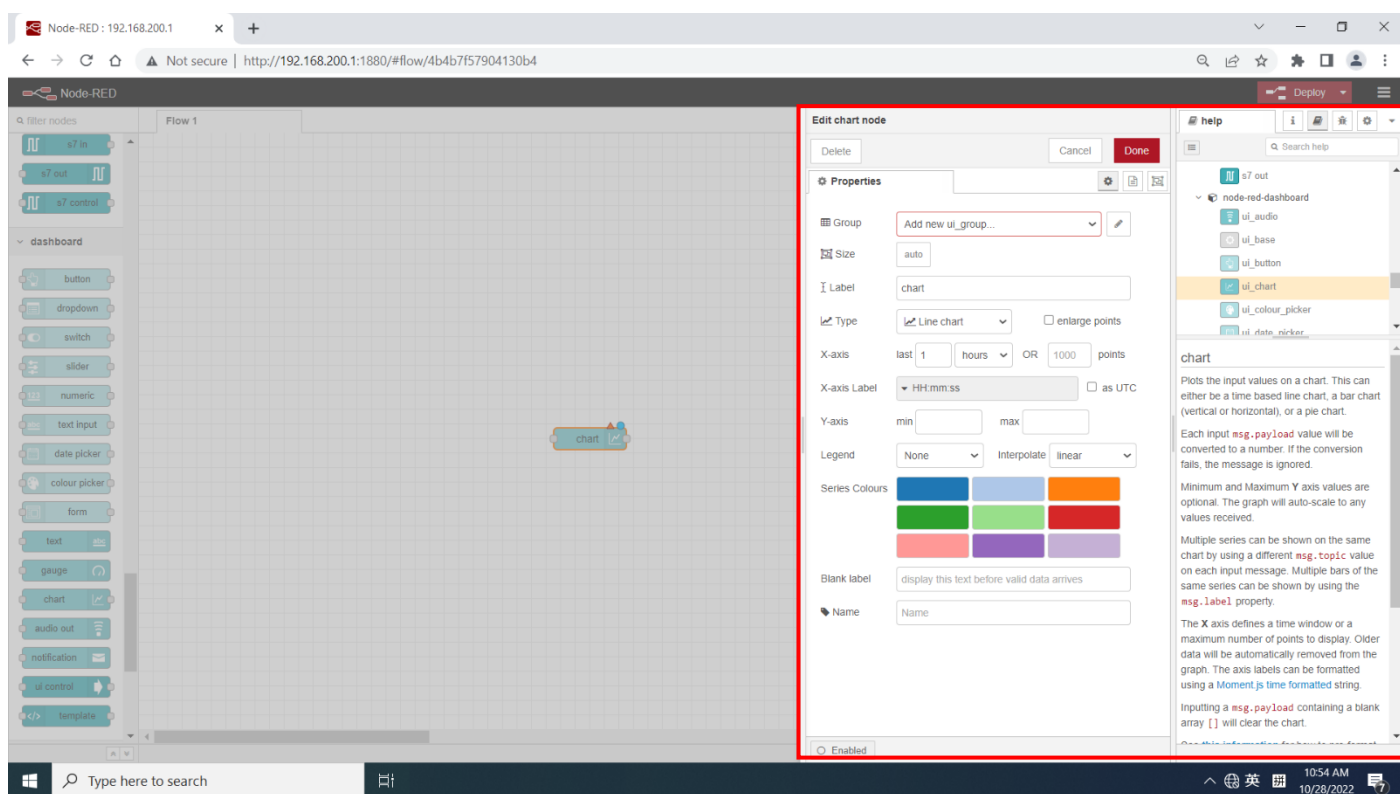
部署成功，进入 UI 页面后，按下发出相应数值的按钮后，gauge 就会显示对应的数值，gauge 中指针区域的颜色也会显示为该数值所在区域对应的颜色



4.4.3 chart 节点用

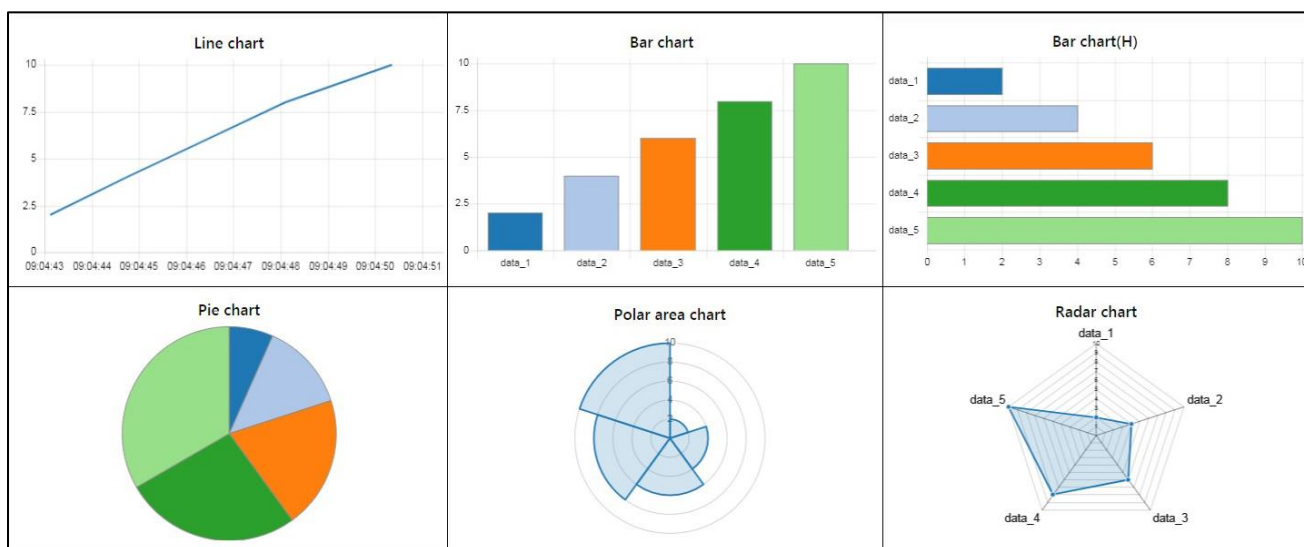
在一般的 UI 页面中，为了更好的展示效果和提高了可读性，页面中通常会带有线性图，柱状图和饼图等等。这些表形效果，都可以通过 chart 节点来实现。

拖拽一个图表控件到工作区域，双击图表控件，可以在对应栏目中查看图表控件的属性，说明以及帮助文档等等

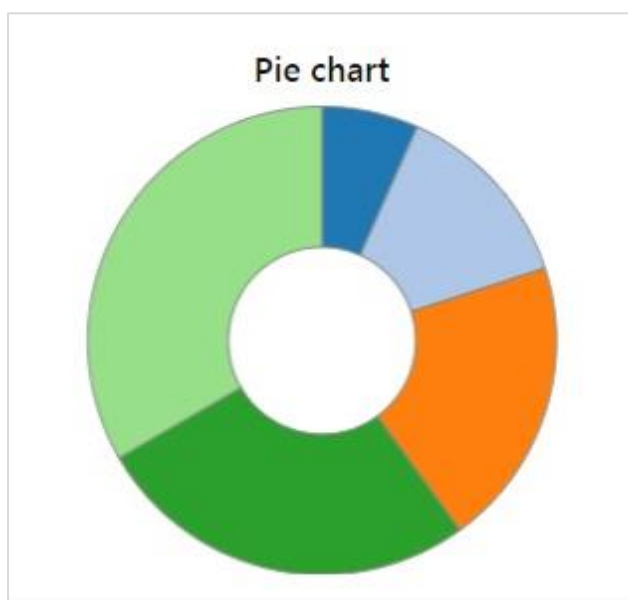


在属性窗口中，可以对图表的分组，大小，类别，颜色等属性进行设置，一些标准属性的设置，可以参考上一章节，一下只对图表独有的一些属性选项进行相关说明：

Type 选项：可以选择不同的图表类型，各种不同的选项对应的图表样式如下图所示。



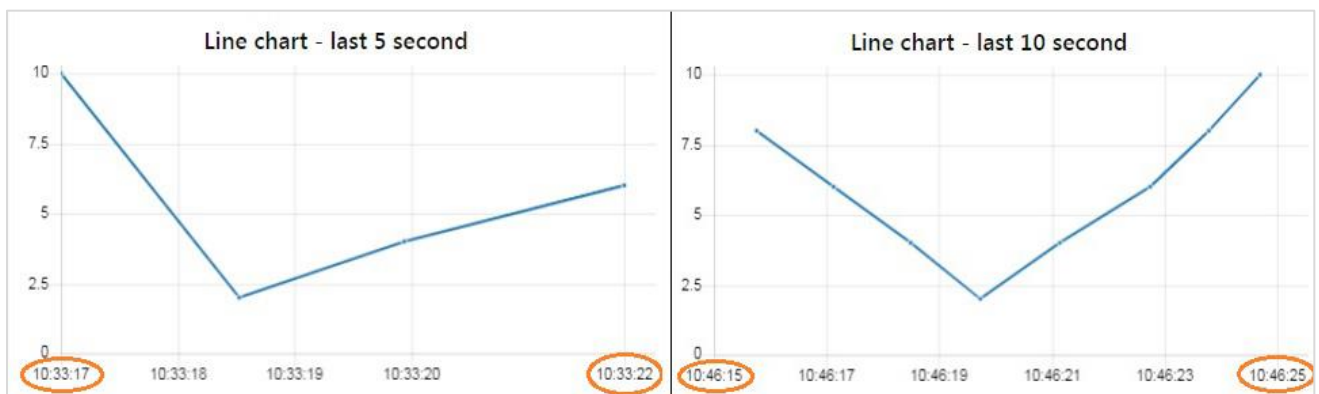
当选择不同图表类型时，后面的可更改的属性选项也会发生相应变化。例如通过设置 Pie chart 的 Cutout 属性值，可以将饼图改变为相应的环形图



当选择 Line chart 图表类型时，可通过勾选 enlarge points 选项来突出显示曲线中各节点的效果。



X-axis 选项：可以设置数据点的保留时长或保留个数，在图例中，当分别将数值设置为 5 秒和 10 秒时，对应的图表就会显示最后 5 秒和 10 秒时间的曲线。



X-axis Lable 选项：可以设置 X 轴标签的格式，用户可通过该选项控制 X 轴标签，如图例所示，当选择以分钟为标签时，即使在一分钟内有多数据点，数据点对应的 X 轴标签也不会发生变化。

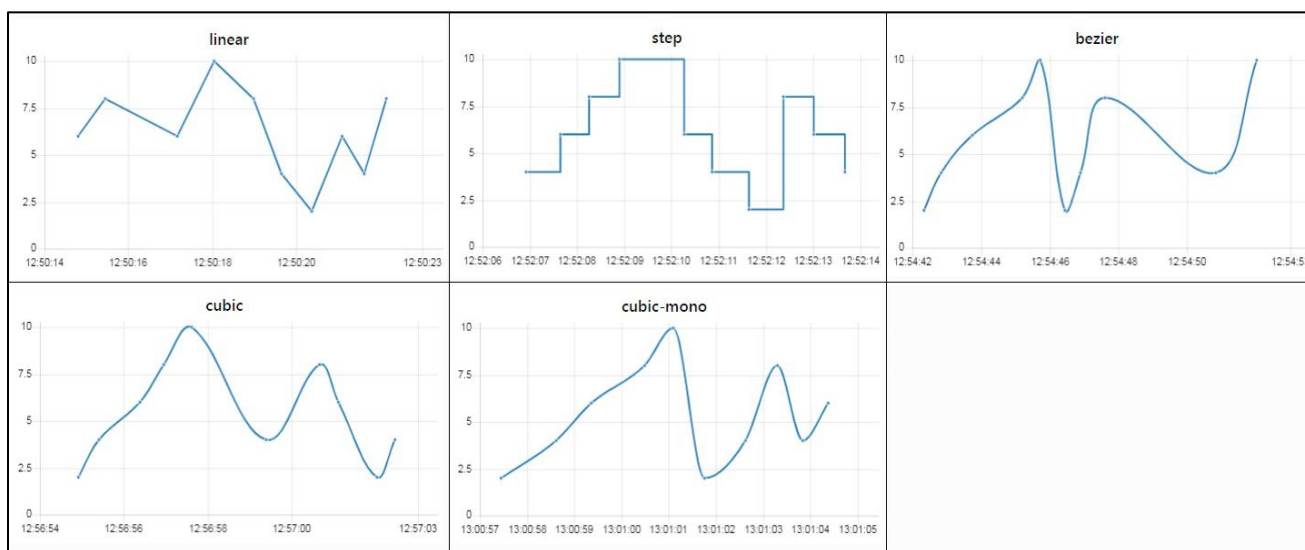
Y-axis 选项：通过 min 和 max 选项可以设置图表 Y 轴的最大和最小值。



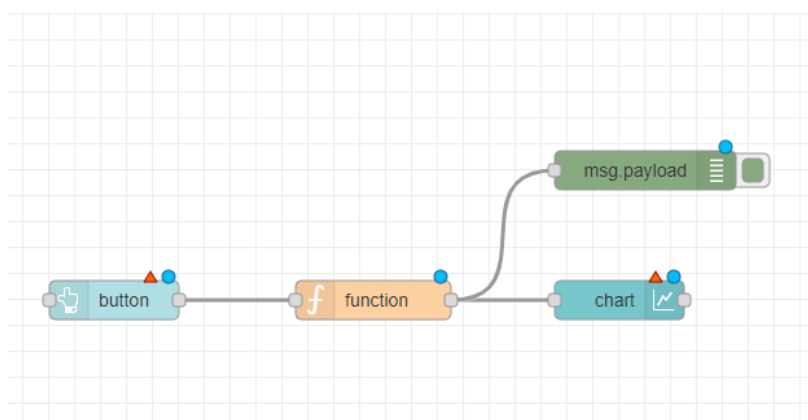
Legend 选项：可以设置是否显示数据标签，如图例中所示，当 Legend 选项设置为 show 时，图表中会显示数据标签



Interpolate 选项：可以设置图表中数据点连接时的过度的方式，选择不同选项时图表的效果如图例中所显示。



分别将一个按钮控件，函数控件，图表控件和 debug 控件拖动至工作区并如图中所示依次连接。



设置按钮属性中的 Group 选项和 payload 选项，将 payload 选项设置为布尔值 true。

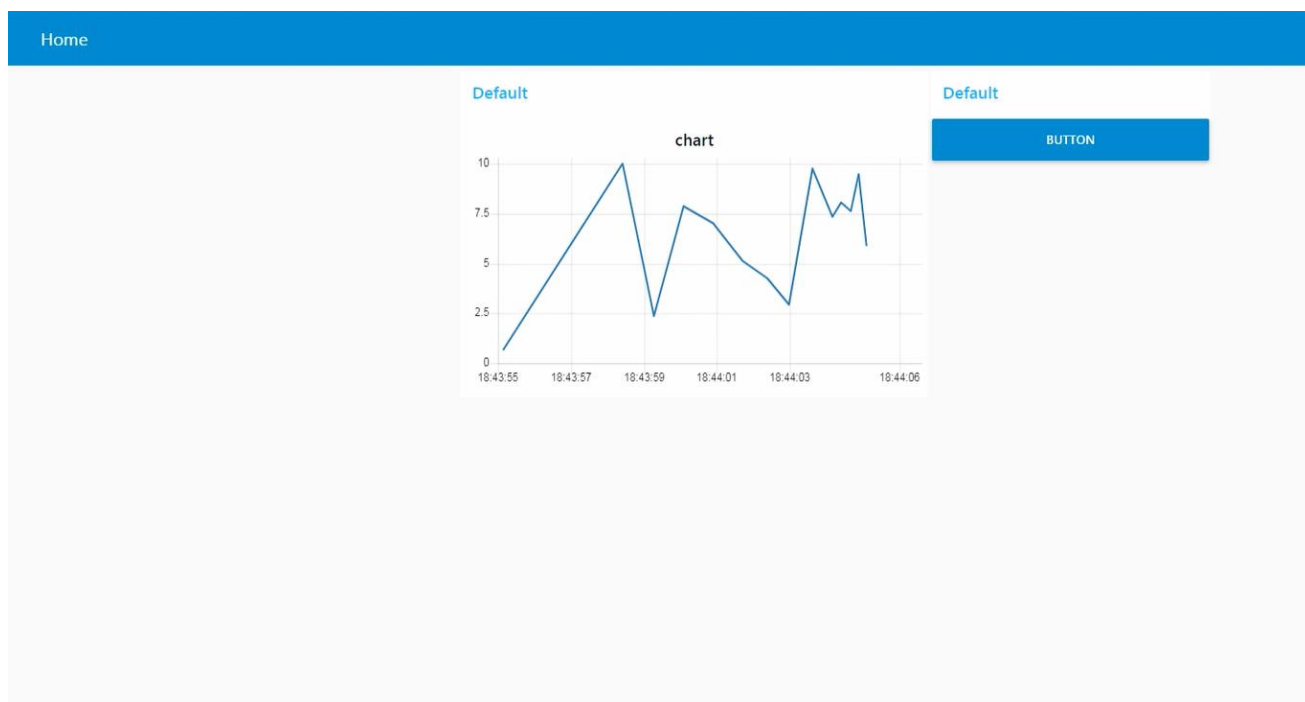
双击函数控件，在弹出的属性框中选中函数库，按照图例所示输入对应函数内容。

```
1 var value1 ;
2 var topic1 ;
3
4 value1=Math.random()*10 ;
5 topic1="data" ;
6
7 msg.payload=value1 ;
8 msg.topic = topic1 ;
9
10 return msg;
```

通过该函数使每次按钮被按下后，都会向图表发送一个 0 至 10 的随机数来模拟从设备采集的数据。

双击图表函数，设定 Group 选项为与按钮相同的 Group，设定 Y-axis 属性，min 值为 0，max 值为 10 后点击完成并部署。

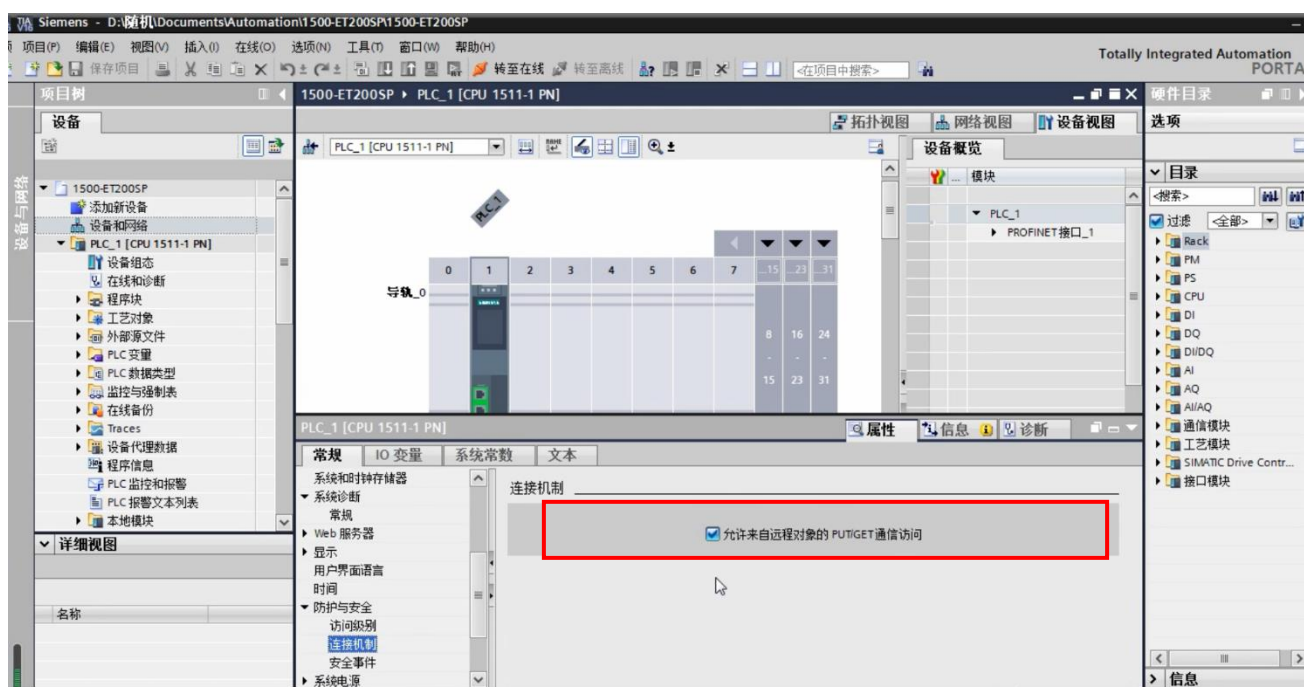
跳转至 UI 页面后，每次按下按钮后，图表中就会根据接收到的数据持续的绘制曲线。



4.5 通过 S7 节点读取 PLC 数据

SIMATIC IOT2050 应用镜像中集成的 Node-red 应用，已经预安装了 S7 节点，经过简单的参数设置，可以通过 S7 节点读取 PLC 中的数据。

在通过 IOT2050 对 PLC 的数据进行采集前，需要先使用博图软件对 PLC 和 ET200SP 进行正确的组态，保证设备正常运行。为了让 IOT2050 访问并读取到 PLC 的数据，组态时需要开启视频中所示 PLC 的“允许来自远程对象的 put/get 通信访问”功能。



设备连接和组态完成后，进入 node-red 应用界面。在节点区域中找到 plc 节点组，将 S7 in 节点拖拽至工作区域中，双击节点进行参数设置。点击 plc 选项后端的铅笔符号设置连接属性。在 Address 处输入 PLC 地址，可在 Cycle time 处设置发送周期。此处的参数，应该和组态时的设置保持一致。通常情况下，只需要设置 PLC 的 IP 地址和更改 PLC 的槽号，其他设置保持默认即可。

编辑 s7 in 节点 > 添加新的 s7 endpoint 配置

取消 添加

属性

Connection Variables

Transport Ethernet (ISO-on-TCP)

Address 192.168.0.1 Port 102

Mode Rack/Slot

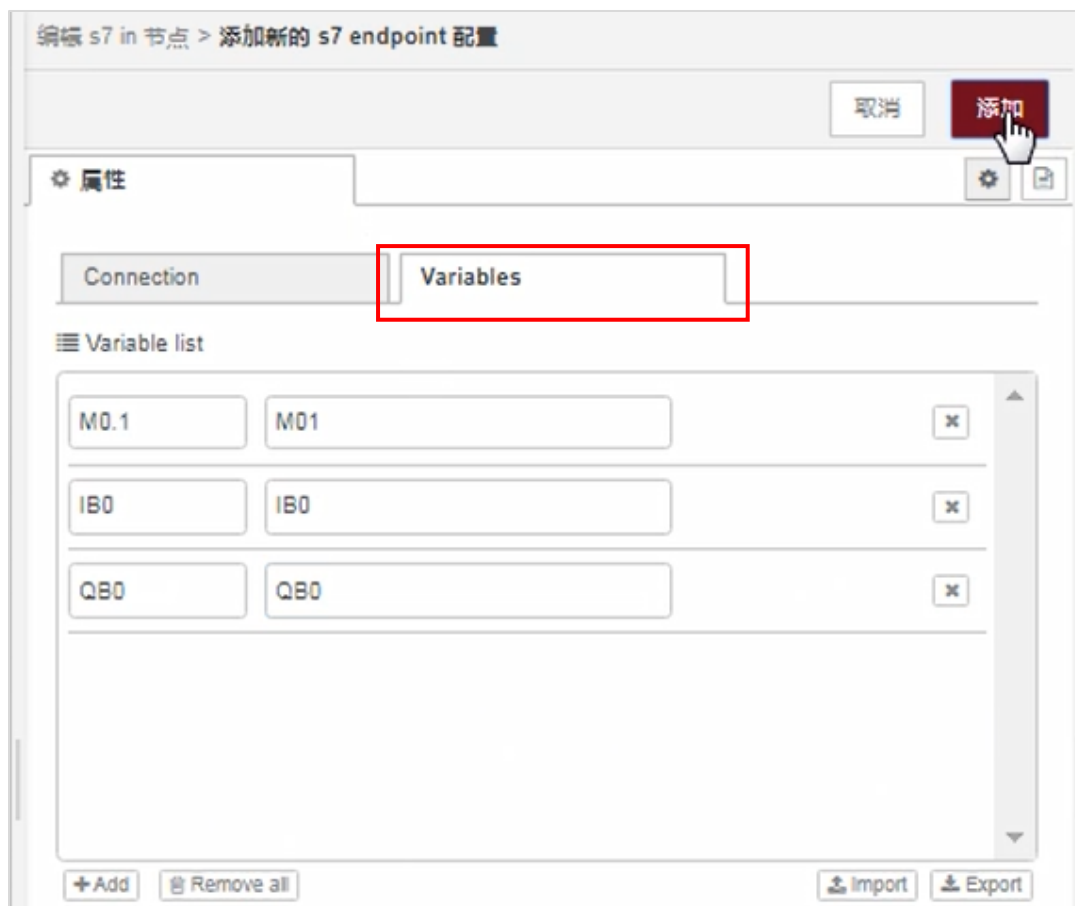
Rack 0 Slot 1

Cycle time 1000 ms

Timeout 2000 ms

Name Name

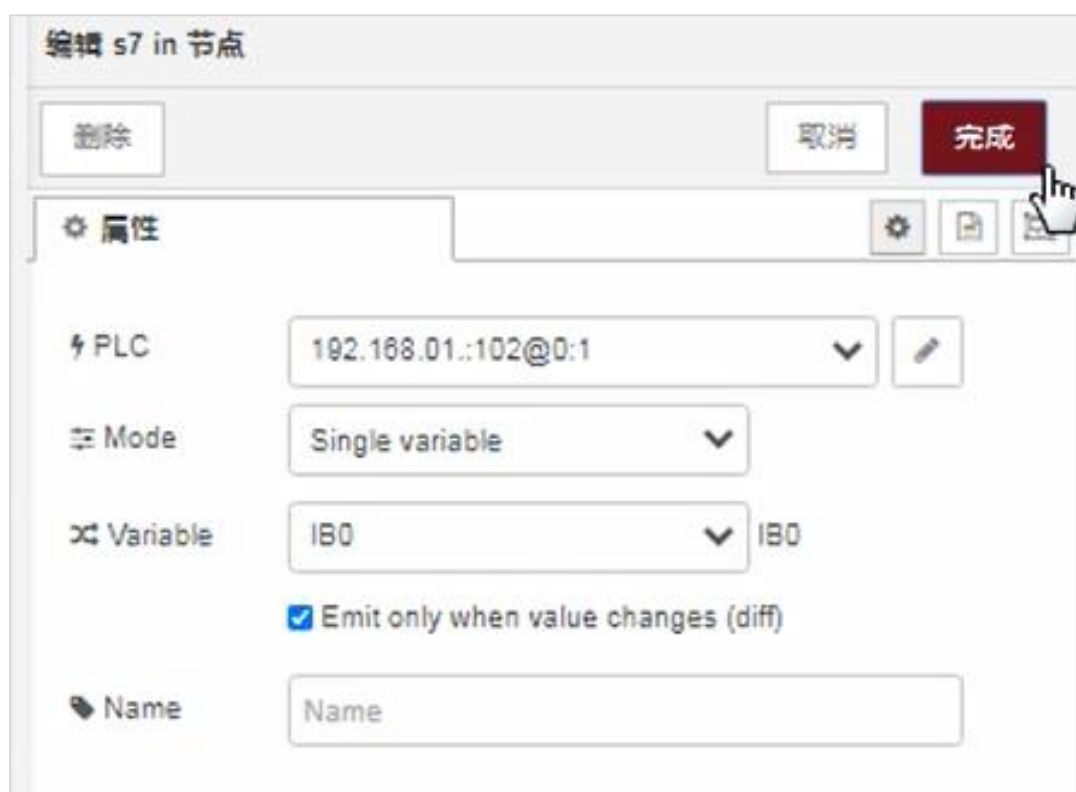
点击变量按钮将设置界面切换至变量列表窗口，在这里可以添加我们想要采集的变量。可以是输入，输出，寄存器或 DB 块等数据。



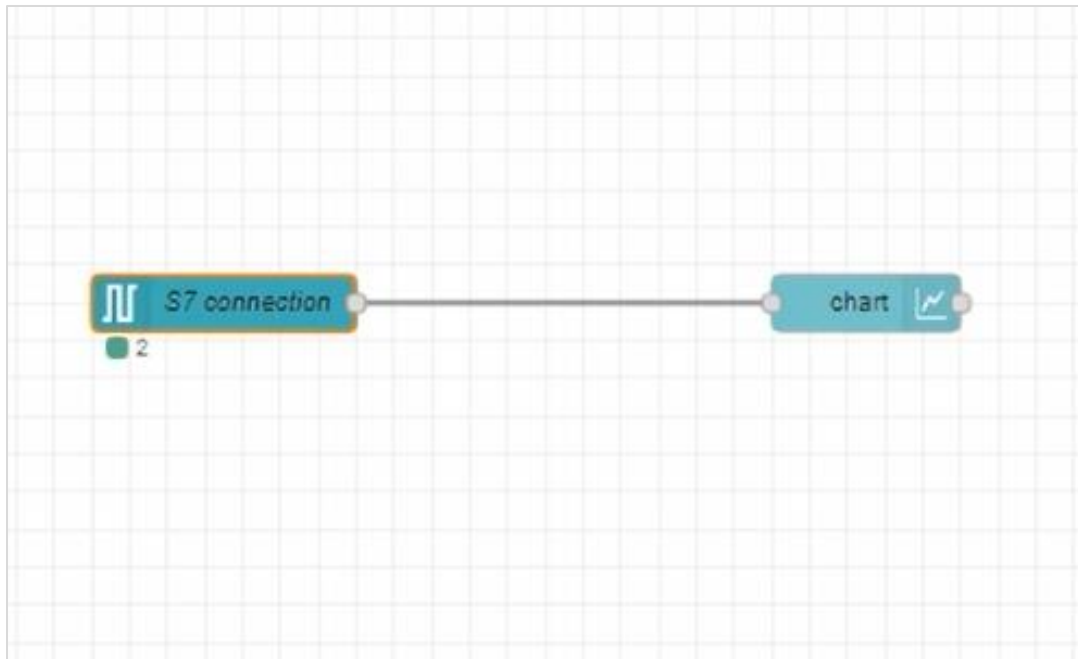
不同变量在 node-red 中的寻址方式，可以下面表格。

	数据区域	数据类型	PLC 寻址方式	Node-red 寻址方式
PLC 变量	I	BOOL	Ix.y	Ix.y
		BYTE,CHAR	IBx	IBx
		WORD,INT	IWx	IWx
		DWORD,DINT, REAL	IDx	IDx
	Q	BOOL	Qx.y	Qx.y
		BYTE,CHAR	QBx	QBx
		WORD,INT	QWx	QWx
		DWORD,DINT, REAL	QDx	QDx
	M	BOOL	Mx.y	Mx.y
		BYTE,CHAR	MBx	MBx
		WORD,INT	MWx	MWx
		DWORD,DINT, ,REAL	MDx	MDx
	T	TIMER (only support S7-300/400)	Tn	
	C	COUNTER (only support S7-300/400)	Cn	
用户数据块	DB	REAL		DBn,REALx
		DWORD		DBn,DWORDx
		DINT		DBn,DINTx
		INT		DBn,INTx
		WORD		DBn,WORDx
		BYTE or B		DBn,Bx
		STRING or S		DBn,S (x - 1).length
		CHAR or C		DBn,Cx

连接和变量设置好后，点击右上角的添加按钮返回属性界面。在 mode 选项框中可以设置读取单个还是全部变量，本示例中我们选择单个变量。在变量选项框中，可以选择采集哪个一变量。Name 选项框可以设置节点在页面中显示的名称。设置好后点击完成退出设置界面。



从节点区域中拖动一个图表节点到工作区，双击节点设置分组及其他属性，设置后点击完成。连接 S7-in 节点和图表节点，点击部署将设置好的程序下载到 IOT2050 中。可以看到 S7-in 节点的左下角会显示连接的状态，连接完成后，可以显示当前采集回来的变量的数值。



跳转至 UI 页面，可以看到从 PLC 采集到的数据已经自动被绘制为图表，当 PLC 中的变量发生变化时，图表上对应的数值也会发生相应的变化。

