

# **Connected Component Workbench Task-based Quick Start**

## **CN01 – Configure Micro800 for Modbus Communications to a PanelView Component**

## **Hardware & Software Versions Used to Develop This Quick Start**

CCW Release 1

Micro830 – 2080-LC30-16QWB

---

## **Before you begin**

Before beginning this quick start, you should already have a general knowledge of how to use the CCW software and how to create an application for you Micro800 controller. If you do not have this knowledge, please review all the Getting Started Quick Starts for Micro800.

The recommended Modbus RTU network topology for a Micro800 and PanelView Component is to configure the Micro800 controller as a slave device, and the PanelView Component as the master device – therefore that is the configuration that will be discussed and configured in this Quick Start.

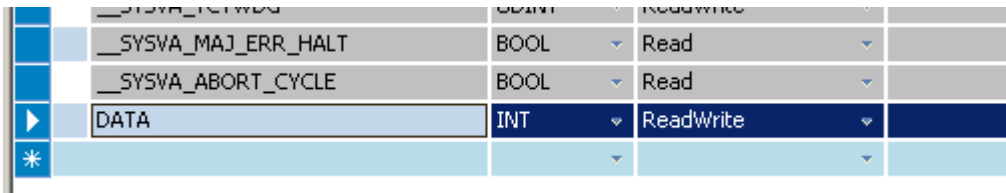
---

## Mapping Variables to Modbus Registers

The Micro800 supports the following Modbus registers.

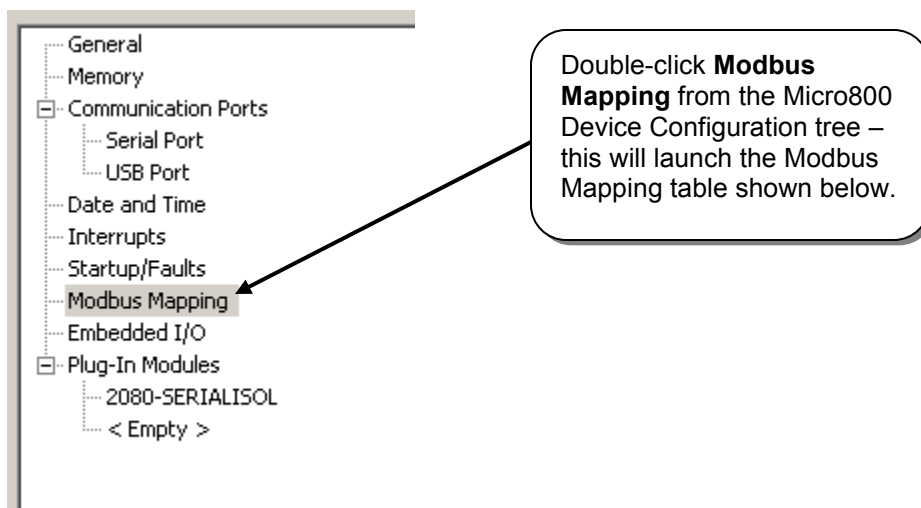
Address	Range	Data Type	Access
Output Coils	000001-065536	Boolean	Read/Write
Input Coils	100001-165536	Boolean	Read Only
Input Registers	300001-365536	Word (16-bit)	Read Only
Holding Registers	400001-465536	Word (16-bit)	Read/Write

1. Create a new CCW project for your Micro800 controller, and create a Global Variable called DATA with data type INT and attribute ReadWrite.



Name	Data Type	Access
__SYSVA_MAJ_ERR_HALT	BOOL	Read
__SYSVA_ABORT_CYCLE	BOOL	Read
DATA	INT	ReadWrite

2. Open the Modbus Mapping table by following the steps below.



Properties

	Variable Name	Data Type	Address	Addresses Used
*				

3. Add a Variable to the Mapping table by following the steps below.

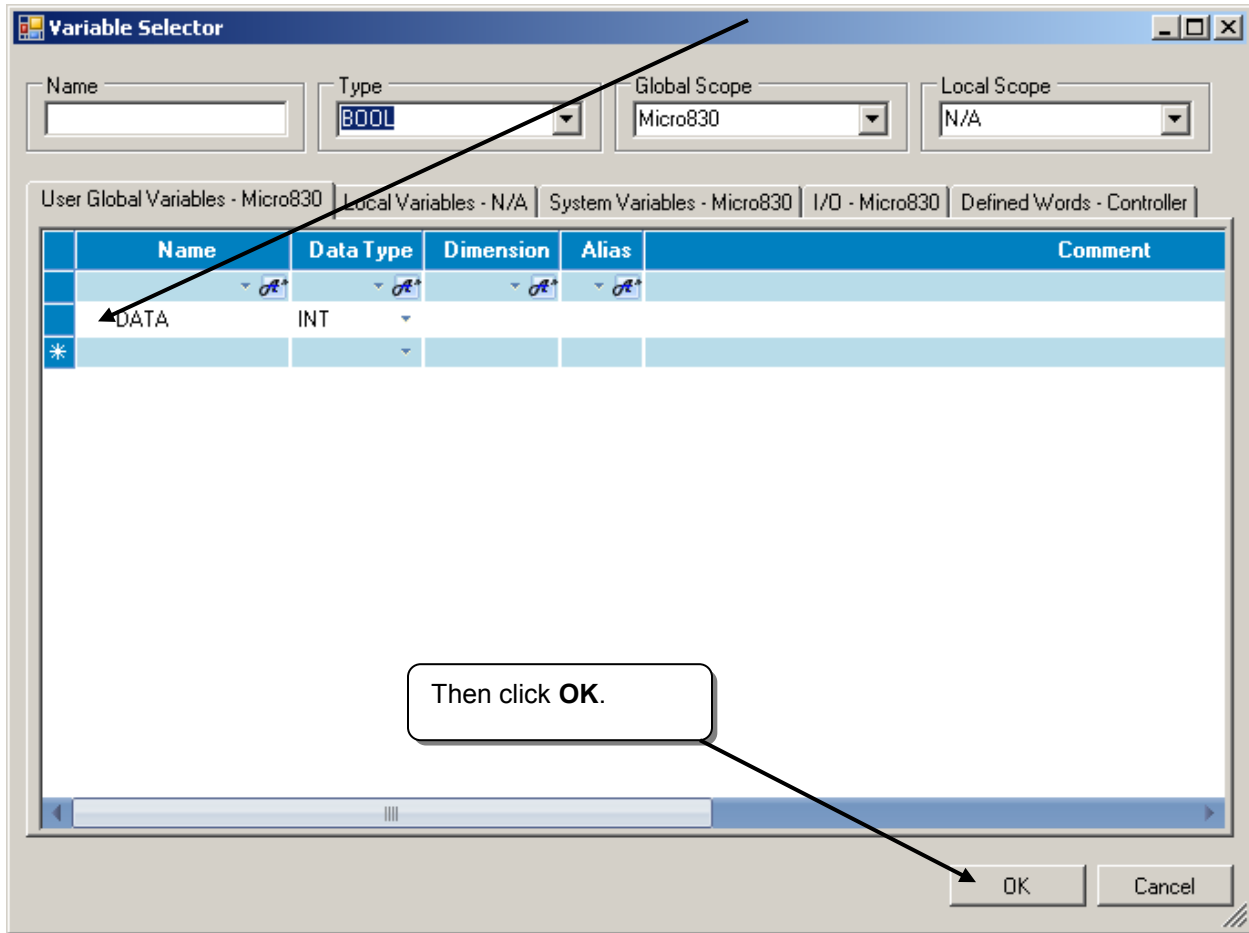
Double-click **here** to launch the **Variable Selector** window.

Properties

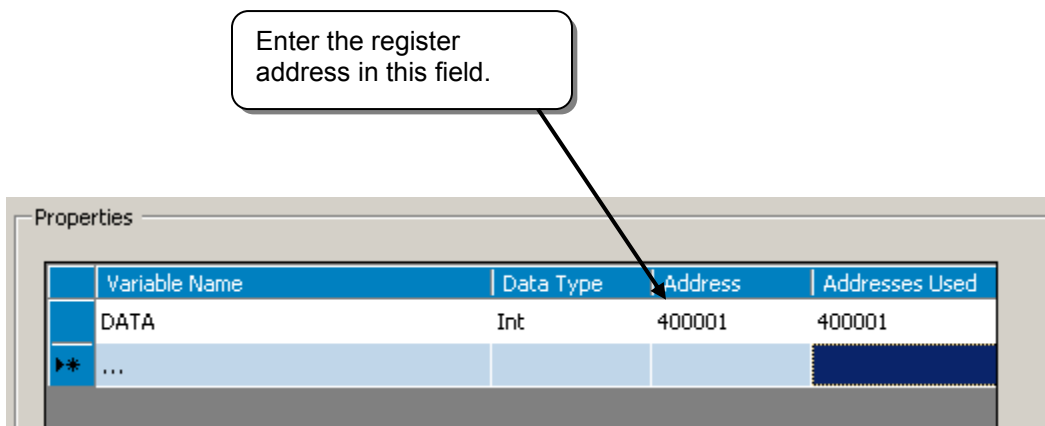
	Variable Name	Data Type	Address	Addresses Used
▶*	...			

Select the **User Global Variables** tab.

Then click **here** to select the **DATA** variable.



- Map the **DATA** variable to register address 400001.



- Repeat steps 3 and 4 for variables, **\_IO\_Embedded\_Digital\_Output\_0** (I/O – Micro830 tab), **\_\_SYSVA\_CYCLECNT** (System Variables – Micro830 tab), and **\_\_SYSVA\_REMOTE** (System Variables – Micro830 tab), and map them to the register addresses as shown below.

Properties

Variable Name	Data Type	Address	Addresses Used
DATA	Int	400001	400001
_IO_Embedded_Digital_Output_0	Bool	000001	000001
__SYSVA_CYCLECNT	Dint	300001	300001 - 300002
__SYSVA_REMOTE	Bool	100001	100001
▶*	...		

Notice that this variable uses two consecutive Modbus registers – this is because it is a 32-bit variable.

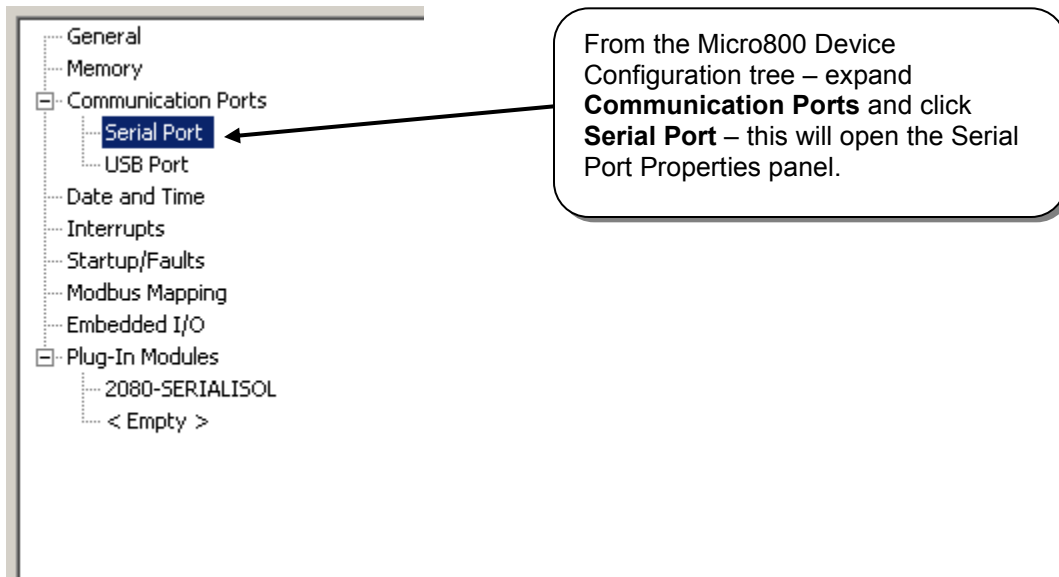
6. You have completed mapping variables to Modbus registers. Save your project.

---

## Configure Micro800 Serial Port

You will be configuring your Micro800 controller as a Modbus RTU slave device. The PanelView Component will be configured as the Modbus RTU Master.

1. Open the Serial Port properties panel.



2. Configure the Serial Port Properties with the following values:

The image shows the 'Properties' dialog box for the Serial Port. The following values are configured:

Driver:	Modbus RTU
Baud Rate:	19200
Parity:	None
Unit Address:	1
Modbus Role:	Modbus RTU Slave

- Expand **Advanced Settings** to configure the **Protocol Control** properties with the following values:

The screenshot shows a dialog box titled "Advanced Settings" with a minus sign icon in the top-left corner. The dialog is divided into two columns. The left column contains the following settings:

- Media: RS232 (dropdown menu)
- Data Bits: 8 (dropdown menu)
- Stop Bits: 1 (dropdown menu)
- Response Timer: 1000 (text input)
- Broadcast Pause: 1000 (text input)
- Inter-Char Timeout: 0 (text input)

The right column contains the following settings:

- RTS Pre-Delay: 0 (text input)
- RTS Post-Delay: 0 (text input)

If you are using RS485, you can set the **Media** property to RS485 and leave the remaining settings the same.

- You have completed configuring your serial port for Modbus. Build and save your project, and then download it to your controller.