



# TOTALFLOW

## *Technical Bulletin 118*

### Configuring XSeries and $\mu$ FLO FCU's and RTU's as Modbus Slaves

#### Totalflow Technical Bulletin

Version 1.0, Revision AB (10 June 2004)

---

#### ABB Inc.

ABB Inc.  
7051 Industrial Blvd  
Bartlesville, Oklahoma  
74006 USA

Telephone  
Domestic 800 442-3097  
International (918) 338-  
4880  
Telefax (918) 338-4607

Internet  
[www.abb.com/totalflow](http://www.abb.com/totalflow)

## **Purpose**

The purpose of this document is to help the end user in setting up and programming an XFC / XRC /  $\mu$ FLO device for operation in a Modbus communications environment.

We will also cover the special case of using the “Enron Modbus Protocol” with the FCU.

## **Description**

There are several steps involved in programming the newest generation of Totalflow Flow Computers to communicate in a Modbus environment. These steps include:

- 1) Setting the Communications Port Parameters – See page 2
- 2) Notes on setting the Modbus Address – See page 5
- 3) Creating Modbus Register Maps – See page 6
- 4) Downloading the Modbus Register Maps to the FCU – See page 13
- 5) Saving the FCU configuration – See page 13

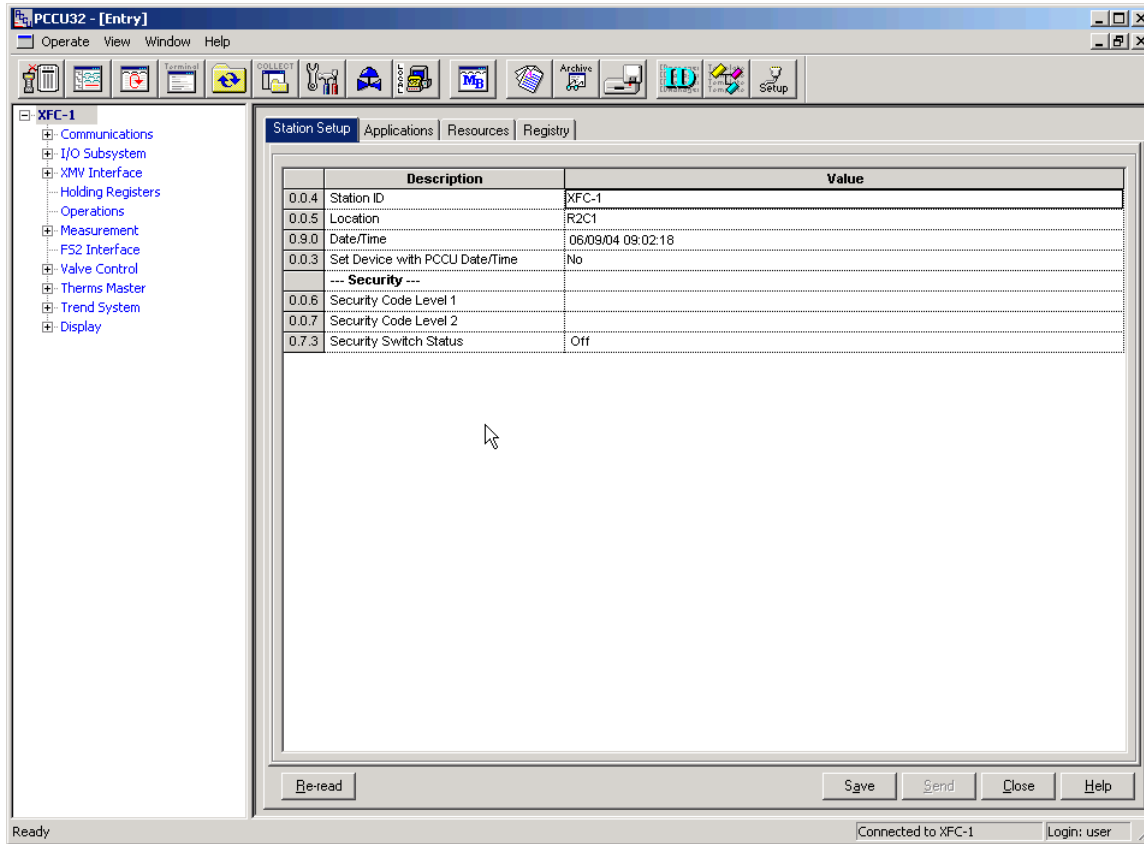
## **Operation**

All of these operations require a user to have a Laptop computer running WinPCCU32 version 4.50 or higher.

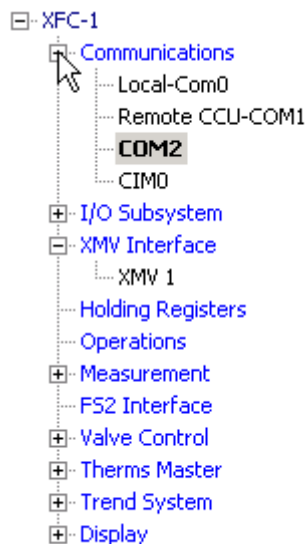
### **1) Setting the Communications Port Parameters:**

Connect to the device using WinPCCU32 and Click the “Enter” button

You should see a screen similar to the following:



(The items in the list on the left are based on those tasks that are running on your FCU, not all meters will display the same list.)



If you click the + sign to the left of “Communications” in the tree on the left you will receive a list of all recognized serial ports in the FCU.

**Never change the programming on the Local-Com0 Port or you will lose the ability to use the PCCU32 software.**

Click on the port that you will be using for Modbus communications to begin programming the port.



You will receive a screen that looks similar to the following:

	Description	Value
0.4.4	Port Name	COM2
3.0.6	Protocol	None

You may make the “Port Name” more descriptive to make it easier to tell what the port is being used for. (i.e. PLC – COM2 might be used to remind you that this port is configured to talk to a PLC wired to COM2)

Select the appropriate Protocol (Either Modbus Slave (ASCII) or Modbus Slave (RTU) as appropriate for the host device, then click the “Send” button at the bottom of the screen.

After clicking “Send” the PCCU will “reconnect” and several items will be added to the screen as shown below.

	Description	Value
0.4.4	Port Name	COM2
3.0.6	Protocol	Modbus Slave (ASCII)
3.0.2	Baud Rate	9600
3.0.11	Slave Address	1
3.0.12	Register Format	32 Bit Totalflow

Set the Baud rate to match the host device and pick the appropriate Register Format from the list. \*

The “Slave Address” may be set on this screen as well, a further discussion of slave addresses will occur on page 5.

#### \* Register Format

32 Bit Totalflow - 32 bit values are transferred as a single 32 bit register. Register list entries are biased one greater than standard Modbus (7002 = 7001 Standard)

32 Bit - 32 bit values are transferred as a single 32 bit register.

16 Bit Modicon - 32 Bit values are transferred as two consecutive 16 bit registers.

16 Bit Word Swapped - 32 Bit values are transferred as two consecutive 16 bit registers and swaps the order of the registers.

16 Bit ROS Modulo 10000 - Multiplies the floating-point number by 1000 and then divides by 10000. The integer portion (before the decimal point) goes into the first 16 bit register and the fractional portion (after the decimal point) goes in the second 16 bit register.

After this screen is programmed click the “Send” button again, then click on the tab at the top labeled “Advanced” and you will receive the following screen.



	Description	Value
3.0.1	Interface	Rs232
3.0.3	Data Bits	8
3.0.4	Parity	None
3.0.5	Stop Bits	2
3.1.1	Xmit Key Delay (milliseconds)	420
3.1.2	Unkey Delay (milliseconds)	40
3.1.3	Timeout(milliseconds)	0
3.0.17	Trailing Pad	None
3.3.0	Directory	R:\COM-2\Modbus
3.0.15	Switched V-Batt/Operate	Disable

The interface should be set to match the COM module that is inserted into the appropriate slot in the board. If using a TFIO communications (CIM) module changing the interface actually reprograms the Module for RS-232 or RS-485/422 operation.

Set the Data Bits, Parity, and Stop Bits to match the host device.

Set the Xmit Key Delay to 10, the Unkey Delay to 20, and the Timeout to 1500 (we suggest these values as a starting point, however, if using Modbus over a radio system you may need to keep the default 420 Xmit Key Delay and 40 Unkey delay for proper communications).

The “Trailing Pad” allows you to add some zeroes to the end of a transmission sequence to make sure the transmission is done before the remote device tries to respond. This is used for devices that have a quick turnaround time, and is usually not required. This can be used in place of having an Unkey Delay when there is no hardware handshaking available on the communications cable. Select 2 bytes, 3 bytes or 4 bytes from the drop down list.

Take note of the “Directory” as we will need to know this in a future step.

Switched V-Batt/Operate should normally be set to Disable on ports using Modbus.

Send these changes to the FCU.

### Setting the Slave Address

A new feature in the X-Series devices allows each communications port running Modbus to have it's own address, if programming multiple ports for Modbus you must set the slave address in each port. Multiple ports can use the same address as long as that address is unique on that communications channel.

Modbus protocol requires that each device in a system have a unique address, however, due to the limited number of available addresses (approximately 250) most systems only require the address to be unique to a particular communications chain. Totalflow devices do support “Modbus Groups”, however, the user must enter manual commands in terminal mode to implement the groups. If you need to use Modbus Groups call Technical Support at 1-800-442-3097 for assistance.



## Creating Modbus Register Maps

In previous Totalflow equipment the list of Modbus registers was pre-programmed into the unit and not changeable \*\* by the end user. The newest Totalflow devices give the end user full control over Modbus register assignments allowing greater flexibility in customizing the device to your system.

\*\* Some PROMS in older 64xx and 67xx did give the end user limited ability to change base registers for different preset groups of variables and to build a small list of items that could be read with a single read multiple register command.

Before we discuss building a custom map, Totalflow does have available a Modbus map that is preconfigured to look much like the Modbus register lists that were available in the older PROM based units. These map files require that you have certain versions of flash installed in your FCU that have a variation on the AGA3 equation with specific support for these systems. If you need your unit to look as much like the older devices as possible contact Totalflow Tech Support at 1-800-442-3097 and ask about "Enron Modbus" support for your device. The programming features mentioned above are not available on the new meters as these register list changes are more easily implemented by just using customized register maps.

To build your own customized register map you will need a list of the information that you would like to be able to access, and the internal App/Array/Register\*\*\* address for that information. The following worksheet might help you in building this list.

Modbus Address	Totalflow Address	Description



Of course if you need more than 9 items you might want to build this table on a separate worksheet. After completing your table you might have something like this:

Modbus Address	Totalflow Address	Description
7001	11.3.0	Tube 1 Static Pressure
7002	11.7.0	Tube 1 Differential Pressure
7003	11.3.3	Tube 1 Temperature
7004	11.7.19	Tube 1 Flow Rate
7011	12.3.0	Tube 2 Static Pressure
7012	12.7.0	Tube 2 Differential Pressure
7013	12.3.3	Tube 2 Temperature
7014	12.7.19	Tube 2 Flow Rate
7021	7.3.5	Battery Voltage
7022	7.3.6	Charger Voltage
7031-7065	11.225.0 through 11.225.34	Historical volumes Today, yesterday, two days ago etc. for tube 1.

\*\*\* Totalflow App/Array/Register addresses are used to indicate which variables are to be made available via Modbus protocol. A discussion on these addresses is located in Appendix A at the end of the document.

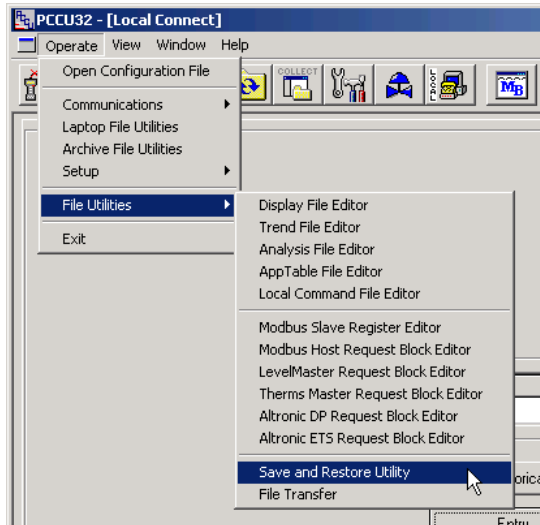
Notice that we have four groups of registers: 7001-7004 for Tube 1, 7011-7014 for Tube 2, 7021-7022 for battery and charger voltages and then a block from 7031-7064 for accessing the volume history.

We will have to create a separate map for each of these groups of registers as the addresses are not contiguous. This is done using the Modbus Register Map editor.

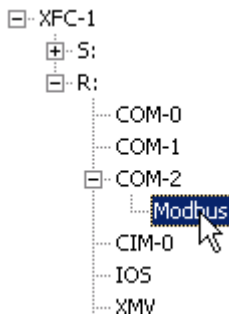
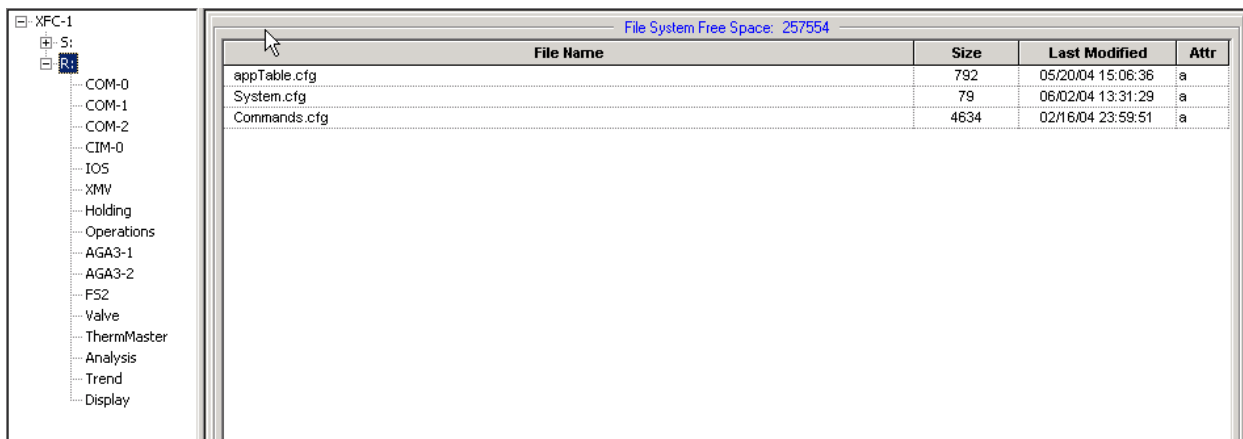
Connect the PCCU32 cable to the FCU and click the Connect to Totalflow button to start the process. After the Connected to device screen comes up start the Save and Restore Utility by clicking the



appropriate icon on the toolbar or by selecting Operate->File Utilities->Save and Restore from the menus as shown below.



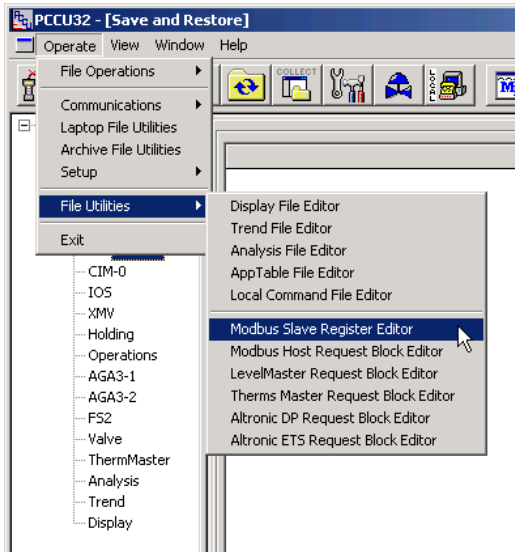
After the Save and Restore utility starts you will see a tree on the left that looks similar to the following:



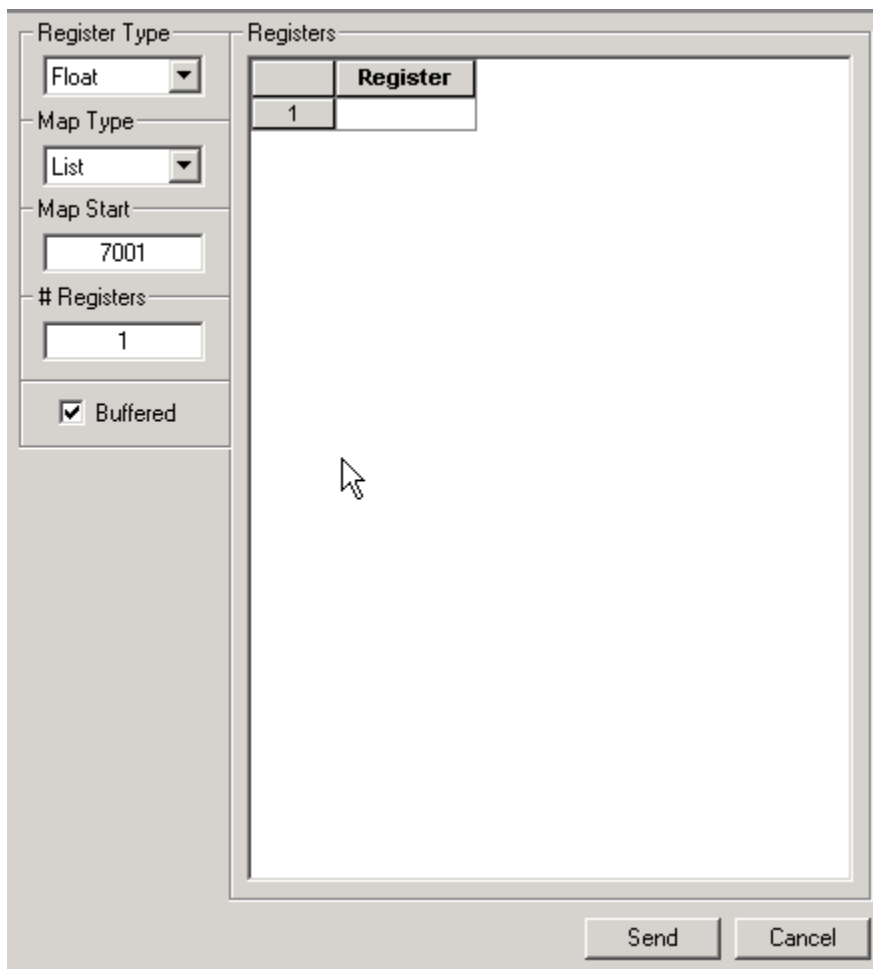
Click on the entry for the communications port you are using for Modbus and you will see a new entry appear underneath it labeled Modbus, click on this new entry and you should see a blank screen on the right.

It is important that the word Modbus actually be highlighted before we continue, please verify that it appears as it does in the example shown at the left, then start the "Modbus Slave Register Editor" by selecting it from the File Utilities Menu on the Operate menu as shown below.





The following screen will appear.





We will use this screen to build the tables necessary to tell the FCU what registers we wish to access via Modbus.

Start by verifying that the "Register Type" is correct. As the default is Floating Point and all of the variables we are planning on accessing are also Floating Point we won't need to make any changes on this configuration.

There are two map types available, List and Array. We will be using both types as we create the Register Maps for this example.

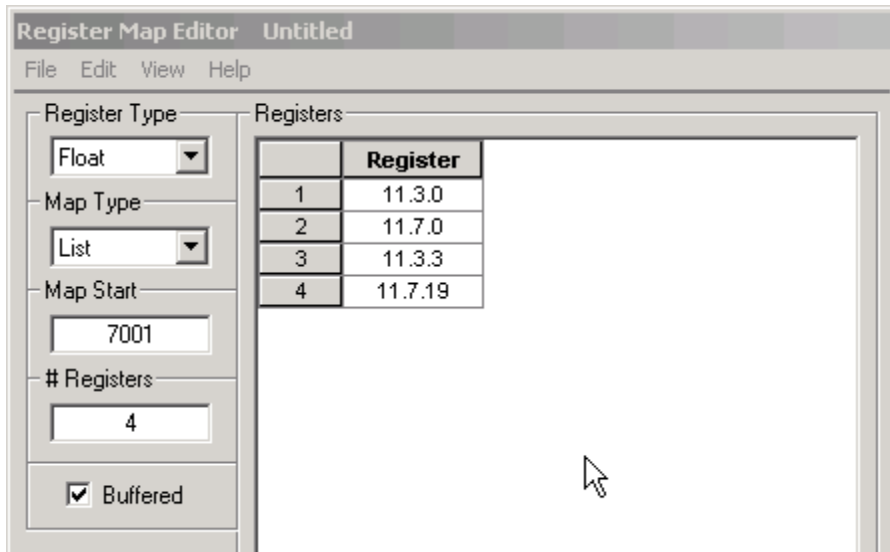
The first group of variables that we want to map were status data for Tube 1 as shown in the table below:

Modbus Address	Totalflow Address	Description
7001	11.3.0	Tube 1 Static Pressure
7002	11.7.0	Tube 1 Differential Pressure
7003	11.3.3	Tube 1 Temperature
7004	11.7.19	Tube 1 Flow Rate

Since the Totalflow Address' of these variables are not contiguous we will set the Map Type to List.

The Map Start should be set to the starting Modbus Address for this group, or 7001 and the # Register to the number of items in this map, or 4 for this example.

Fill in the Totalflow Addresses from the table above into the Registers list on the left and you should end up with a screen that looks like the following.

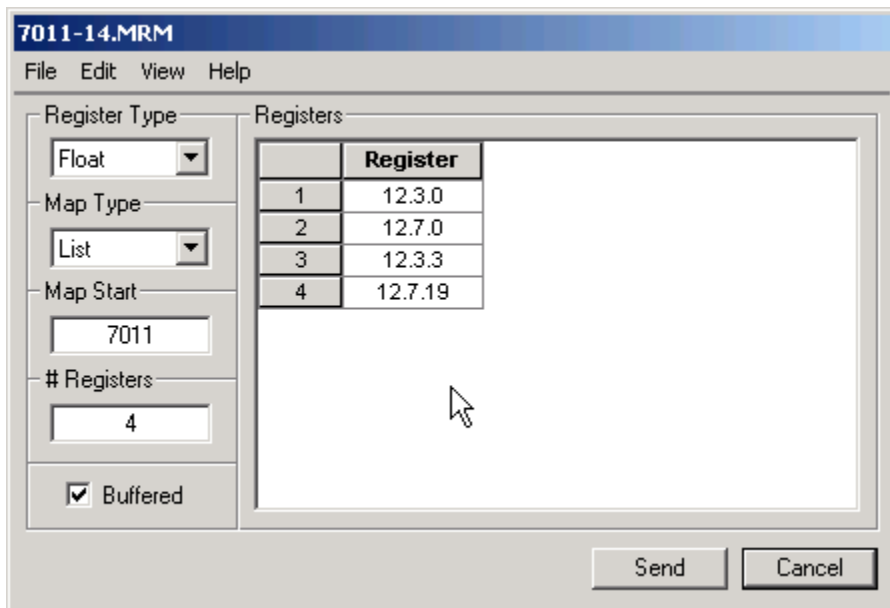


If you are using 4.53 or higher PCCU32 there will be a “Send As” option on the File menu that will send this file to the unit and place a copy in the folder that was highlighted when we started the editor (that’s why it was important above.) If using 4.50 to 4.52 we will transfer the files after we finish building the rest of them. When you send the file name it 7001-04.mrm (the file name shows us the registers that are used and the .mrm stands for modbus register map.)

At this point I recommend (this step is mandatory for PCCU 4.50 – 4.52) that you save this file for future use, to do this Select “Save As” from the file menu and name the file 7001-04.mrm.

To create the map for tube 2 we can make a few changes to this screen. Simply change the Map Start to 7011 and change the 11. that is the first part of each register address to a 12.

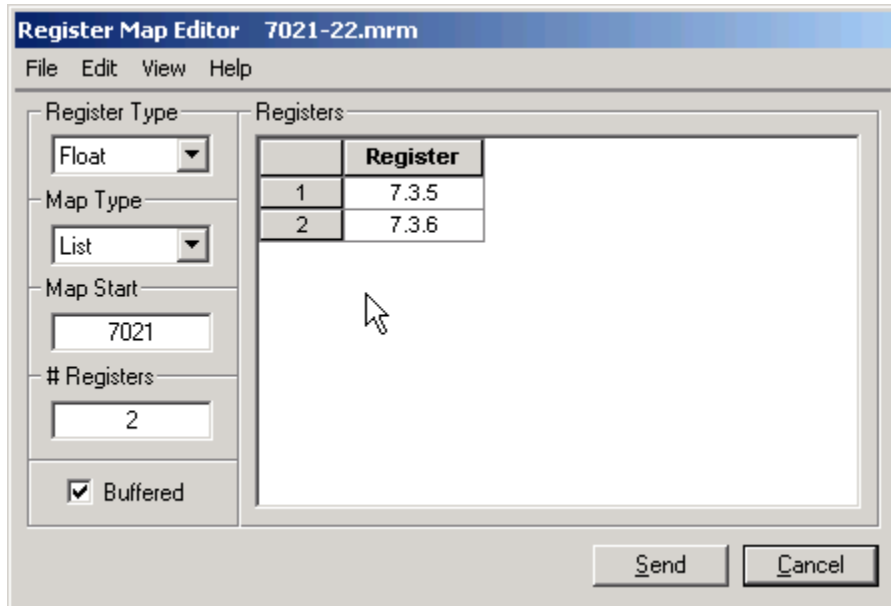
You should see the following screen.



Use the procedure outlined above to send / save the file, this time using the name 7011-14.mrm for the file.

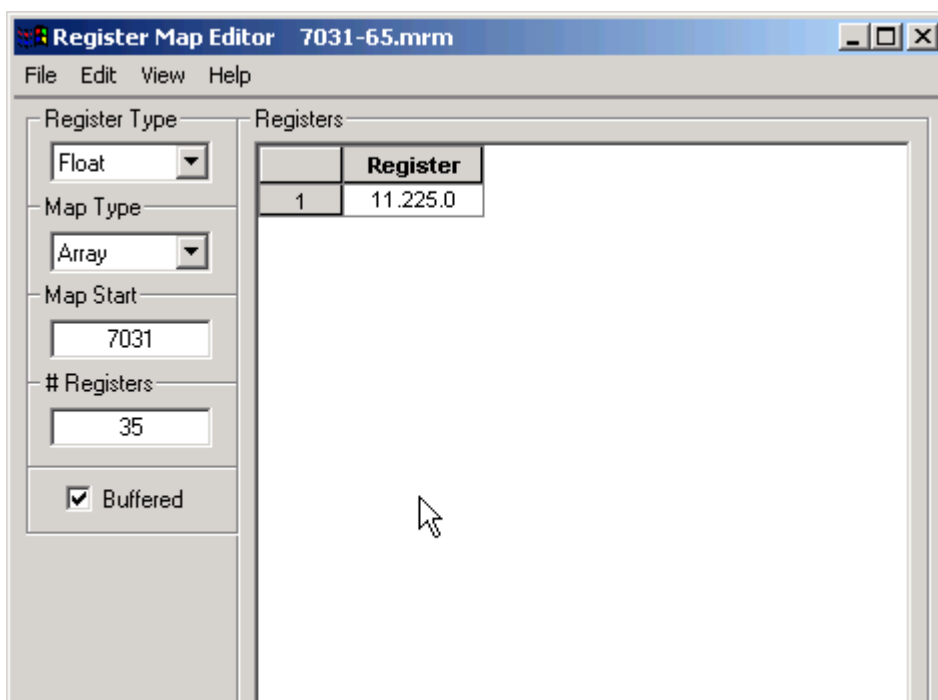
For the next block set the Map Start to 7021, the # registers to 2, and the addresses to 7.3.5 and 7.3.6 as shown in the original table.

You should end up with the following screen.



Send / Save this file as 7021-22.mrm

The last item in our table was the historical data. This is a list of sequential variables (note that they App.Array is the same on each variable, and the .Register changes and they are sequential.) To create this register map we will need to make several changes.



First change the number of registers to 1 (we need to reduce the register list to a single entry before proceeding.)

Second Change the Map Type to Array.

Third change the Map Start to 7031.

Fourth set the # Registers to 35 (we want a list of current day

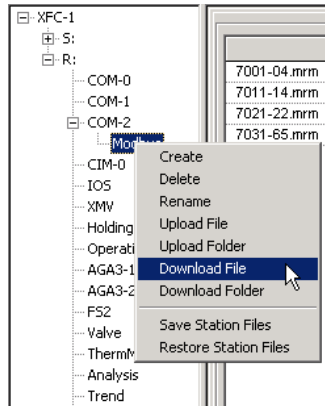
plus 34 previous days of volume information).

Fifth set the register number to 11.225.0 (the first register in the list we wish to access.)

Send / Save this file as 7031-65.mrm

Click the Close / Cancel button on the Modbus slave register editor.

### Downloading Modbus Register Maps

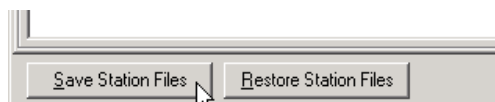


If you have been able to send the files all along we are done, and the unit should respond to any of the modbus registers we have programmed. For PCCU 4.50- 4.52 we need to transfer the files we created. This is done by clicking on the “Modbus” folder on the tree with the right mouse button then picking “Download File” from the menu.

Do this once for each file and send the four files we created during this exercise.

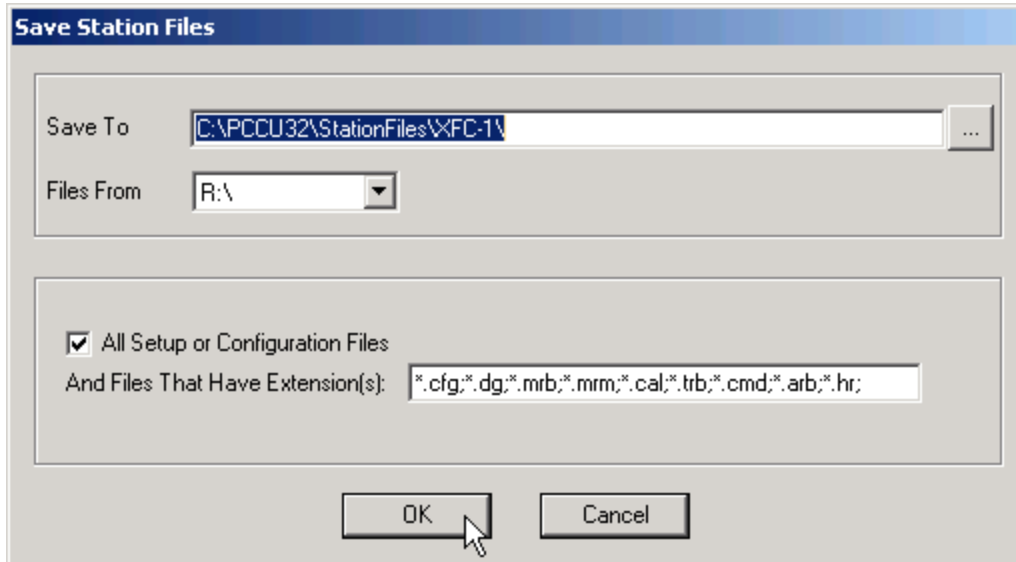
### Saving the FCU configuration

We recommend that anytime changes have been made to the FCU programming that you create a backup copy and “restore” the files to the S: drive in the unit so that if it is ever required that the unit be cold booted it will return to normal operation and won’t have to be reprogrammed.



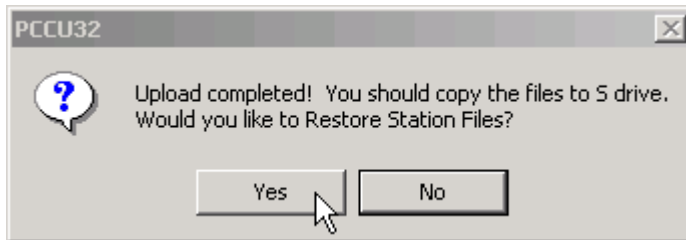
This is done by clicking the “Save Station Files” button and the bottom of the Save and Restore Utility screen. Since we are already in the Save and Restore Utility while building these files it’s simply a matter of clicking the button.

When you click on this button you will see the following screen.



The system will automatically pick the proper folder, based on the stations device ID, to place the files in, just click OK to start the process. (If asked select "Delete all existing files and save")

After all the files have been read and a copy stored on your laptops hard disk you will see the following screen.



Click the Yes button to save a copy of the configuration back to the S: drive of the flow computer. This copy is used to automatically configure the device in the event of a cold boot. By saving the files now we know that should the unit ever be cold booted it will come back running the Modbus configuration we just created.

# TOTALFLOW

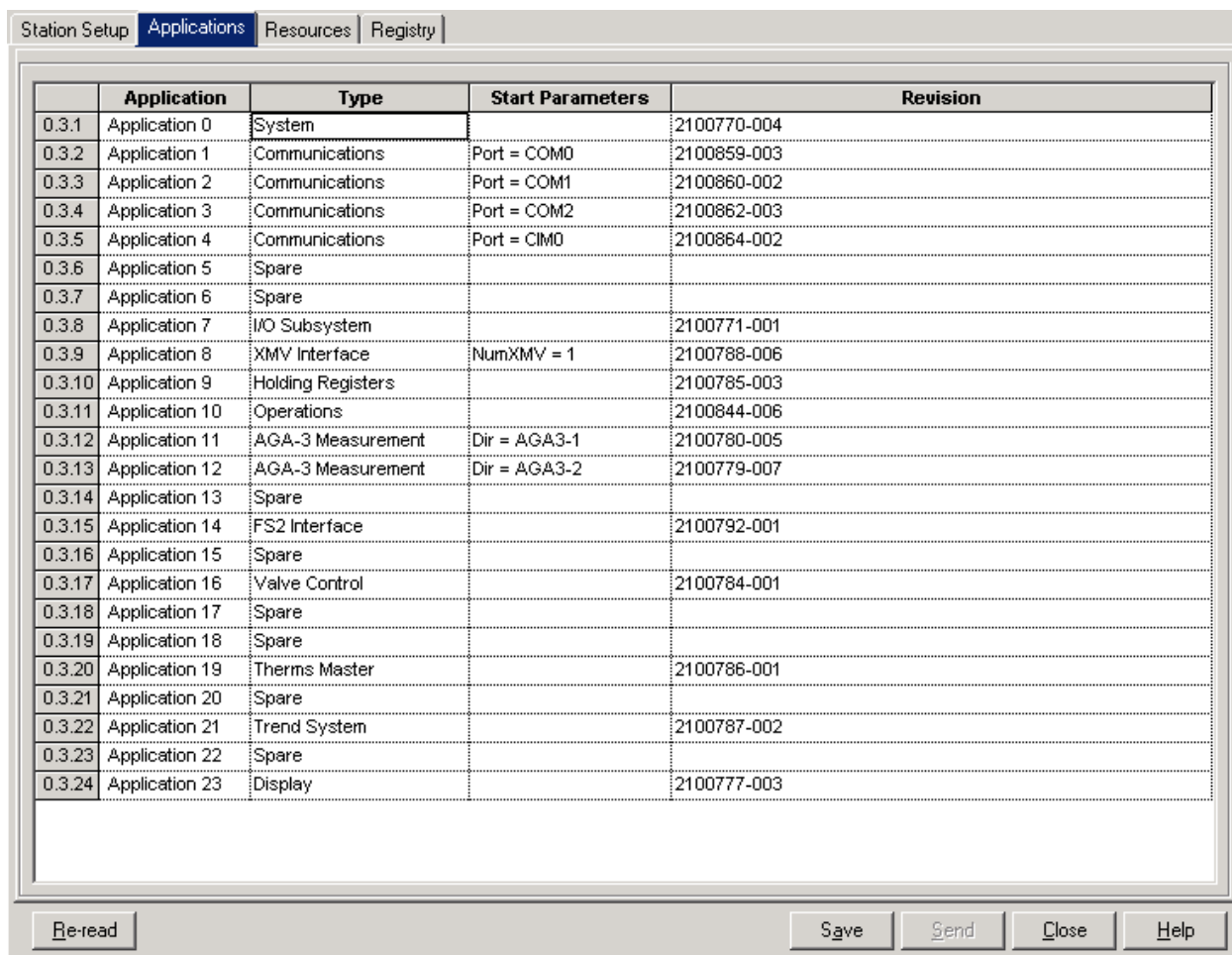
## Appendix A

App/Array/Register Addresses.

The Totalflow FCU's use a tiered structure to store data and use a three part address to uniquely identify the data.

Each address starts first by identifying which application it is associated with. When connected to an X-Series device these applications ID numbers and types can be seen by clicking on the tab marked applications on the opening screen of "Entry" mode.

You will see a screen that looks similar to the following.



	Application	Type	Start Parameters	Revision
0.3.1	Application 0	System		2100770-004
0.3.2	Application 1	Communications	Port = COM0	2100859-003
0.3.3	Application 2	Communications	Port = COM1	2100860-002
0.3.4	Application 3	Communications	Port = COM2	2100862-003
0.3.5	Application 4	Communications	Port = CIM0	2100864-002
0.3.6	Application 5	Spare		
0.3.7	Application 6	Spare		
0.3.8	Application 7	I/O Subsystem		2100771-001
0.3.9	Application 8	XMV Interface	NumXMV = 1	2100788-006
0.3.10	Application 9	Holding Registers		2100785-003
0.3.11	Application 10	Operations		2100844-006
0.3.12	Application 11	AGA-3 Measurement	Dir = AGA3-1	2100780-005
0.3.13	Application 12	AGA-3 Measurement	Dir = AGA3-2	2100779-007
0.3.14	Application 13	Spare		
0.3.15	Application 14	FS2 Interface		2100792-001
0.3.16	Application 15	Spare		
0.3.17	Application 16	Valve Control		2100784-001
0.3.18	Application 17	Spare		
0.3.19	Application 18	Spare		
0.3.20	Application 19	Therms Master		2100786-001
0.3.21	Application 20	Spare		
0.3.22	Application 21	Trend System		2100787-002
0.3.23	Application 22	Spare		
0.3.24	Application 23	Display		2100777-003

In this case we can see that several spaces are available (marked spare) for additional tasks, but this list is important when trying to access data as we always need to look here to identify the first part of the address. In this screen note that there are two tubes (application 11 and application 12) as indicated by the type being AGA-3 Measurement.



Within each application, Arrays are used to organize data by the type of variable (floating point, integer, Boolean etc.) We also support having multiple arrays of the same type within a single application, this allows us to group similar related data into a single array.

The final part of the address is an index into the array to say which of the individual items in the array we are wanting to address. These variables are entered into the system by combining the three numbers using periods to separate them.

As an example, the AGA-3 Measurement application uses array 225 to hold daily volumes. Variable 0 holds today's volume, variable 1 holds yesterday's volume, variable 2 holds the volume from three days ago... To access this data we identify the application (11 or 12 in our case as these are the two places in the application table where we see AGA-3 Measurement), the array 225, and then the index for the appropriate day. 11.225.0 would be today's volume on tube 1, 12.225.1 would be yesterday's volume on tube 2.

You will notice if you look at any of the screens in entry mode in PCCU32 4.50 or higher that it shows you this variable address for each field that it displays. If you were to look under Current Values in the measurement section you will see a screen similar to the following:

	Description	Value	
11.3.0	Static Pressure	98.626	PSIA
11.7.0	Diff. Pressure	86.060	InH2O
11.3.3	Temperature	51.303	Deg F
11.7.19	Flow Rate	591.022	MCF/DAY
11.3.98	Energy Rate	25956.000	MBTU/HOUR
11.7.22	Today's Volume	234.349	MCF
11.7.23	Yesterday's Volume	590.963	MCF
11.7.21	Accumulated Volume	15997.638	MCF
11.7.20	Last Calc Period Volume	410.430	SCF
11.3.96	Today's Energy	247017.703	MBTU
11.3.97	Yesterday's Energy	623779.188	MBTU

For any item you wish to access via Modbus if you can find it somewhere on the PCCU's screens it will show you the register number to access.

Totalflow has also developed a set of Microsoft Word documents that have a complete list of all variables in each application type, these documents are available upon request by calling Totalflow Technical Support at 1-800-442-3097.