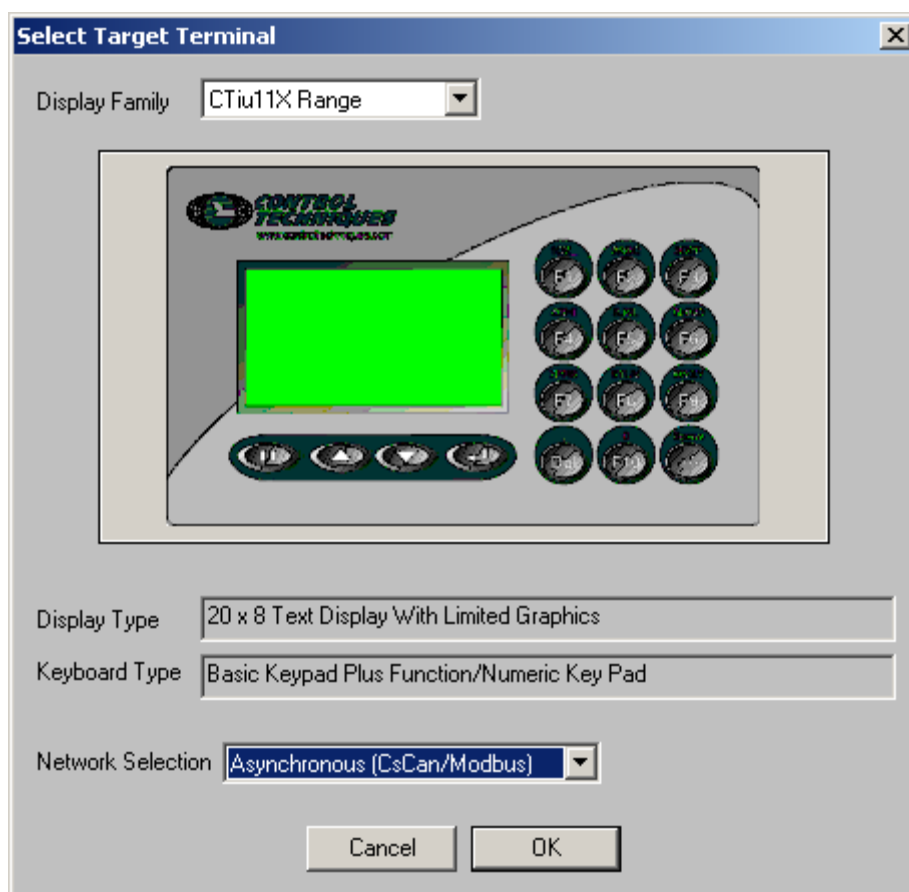


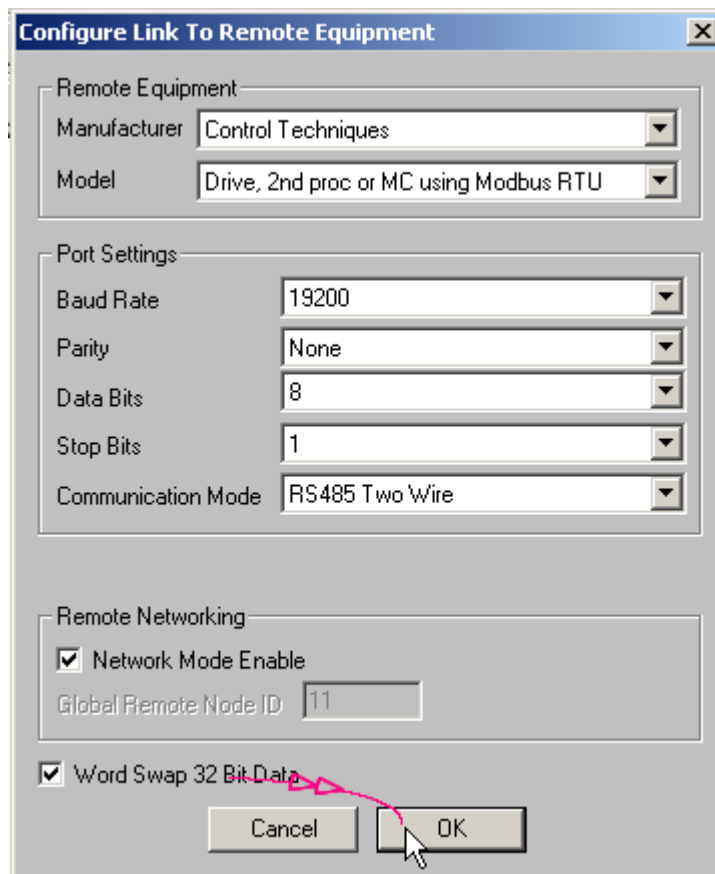
The Application Note is specific the CTIU HMI and our Unidrive SP

## Establishing Communications between the CTIU and UnidriveSP using Modbus RTU

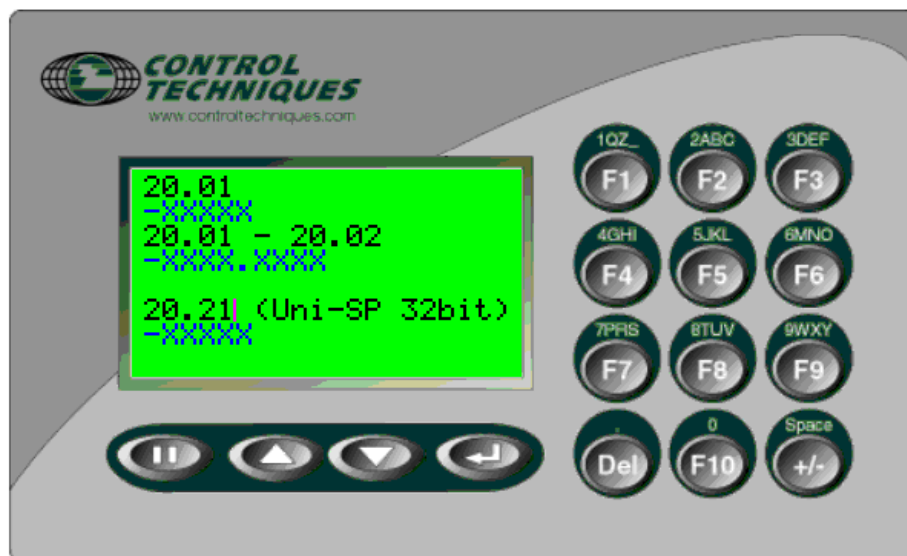
Start by selecting the proper CTIU for the application. For this application the CTIU 110 terminal will be implemented. Different CTIU terminals will require different Operating Systems in order to run. These operating systems are generally loaded before the customer receives his CTIU so this software will not have to be loaded manually.



Select "Configure", "Communications Settings" to select the proper driver for communications. Choose the following settings to ensure proper communications between the CTIU and the three products.



The first page of this application note will detail communication with three different types of data. The Unidrive SP includes specific registers that are 32-bits in length. These should not be confused with 2 consecutive 16 bit registers that make up a 32-bit value. The latter is very common method for passing 32 bit data over Modbus. Unlike consecutive 16 bit registers, 32 bit registers have one address and store 32 bits within that one address. One example of a 32-bit register is 20.21. Although it is not common for HMI manufacturers to include a driver that handles 32-bit Modbus, the CTIU does include one.



To add a register such as 20.01, double click in the area that you would like this information to reside on the screen. Choose to add a piece of numeric embedded data and configure it for the following to read and write 20.01. This is a 16 bit register in the Unidrive SP.

**Configure Numeric Embedded Data**

Remote Data Source

Remote Node ID: 1

Data Type: 16 Bit Drive Parameters

Location: 20.1

Display Format

Decimal | XXX.XXX

☒ 16-Bit ☐ Unsigned

☐ 32-Bit ☒ Signed

Edit/Write

☒ Enabled

Range Checking

☐ Enable

Minimum: 0.000

Maximum: 65.535

Scaling

☐ Enable

TIU Minimum: 0.000

TIU Maximum: 65.535

Remote Minimum: 0

Remote Maximum: 65535

Attributes

☐ Flashing

Change Type Cancel OK

If it is desirable to write a 32 bit number to two consecutive 16-bit registers then simply select a data type of "16 bit drive parameters" but a display format of "32 bit". The following screen capture demonstrates the setup required to read 20.02 and 20.03 from the Unidrive SP and form these two 16 bit registers to display as one 32 bit register.

**Configure Numeric Embedded Data** [X]

Remote Data Source

Remote Node ID

Data Type

Location

Display Format

☐ 16-Bit ☐ Unsigned  
☒ 32-Bit ☒ Signed

Edit/Write

☒ Enabled

Range Checking

☐ Enable

Minimum

Maximum

Scaling

☐ Enable

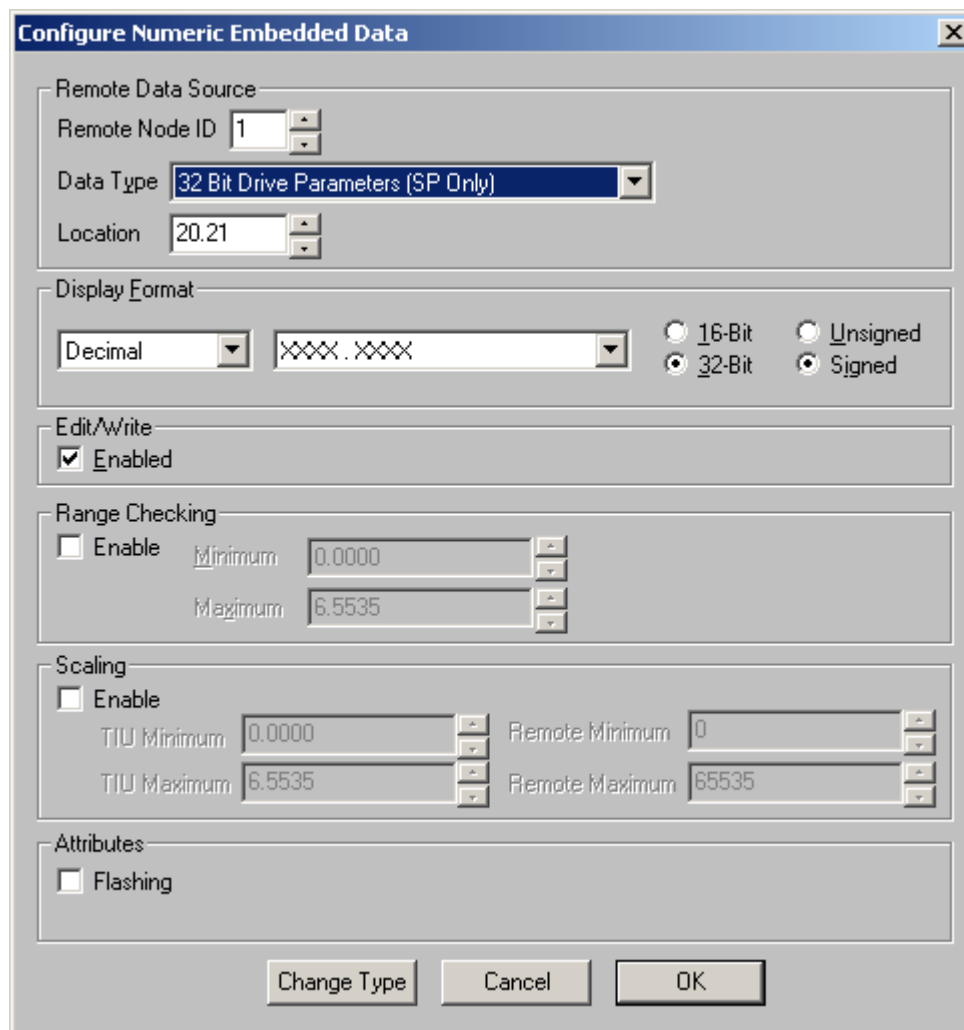
TIU Minimum  Remote Minimum

TIU Maximum  Remote Maximum

Attributes

☐ Flashing

To write to the Unidrive SP's 32 bit parameters, simply choose the data type to be "32 Bit Drive Parameters (SP Only)". Since 20.21 is a 32 bit parameter, the following Numeric Embedded Data Format should be used for data transfer.



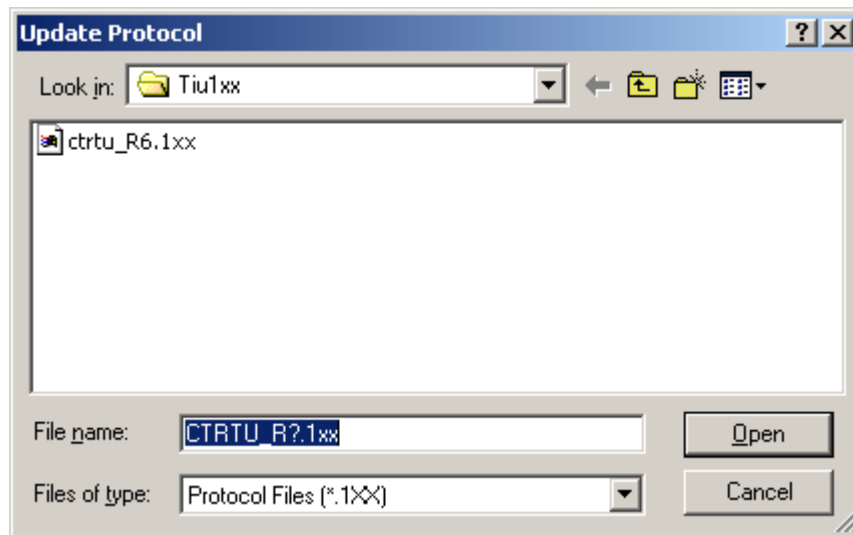
The image shows a Windows-style dialog box titled "Configure Numeric Embedded Data". It contains several sections for configuring data transfer parameters:

- Remote Data Source:** Includes a "Remote Node ID" spinner set to 1, a "Data Type" dropdown menu set to "32 Bit Drive Parameters (SP Only)", and a "Location" spinner set to 20.21.
- Display Format:** Includes a "Decimal" dropdown, a pattern field showing "XXXX.XXXX", and radio buttons for "16-Bit", "32-Bit" (selected), "Unsigned", and "Signed" (selected).
- Edit/Write:** A checkbox labeled "Enabled" which is checked.
- Range Checking:** A checkbox labeled "Enable" which is unchecked. Below it are "Minimum" and "Maximum" fields with values 0.0000 and 6.5535 respectively.
- Scaling:** A checkbox labeled "Enable" which is unchecked. Below it are four fields: "TIU Minimum" (0.0000), "TIU Maximum" (6.5535), "Remote Minimum" (0), and "Remote Maximum" (65535).
- Attributes:** A checkbox labeled "Flashing" which is unchecked.

At the bottom of the dialog are three buttons: "Change Type", "Cancel", and "OK".

Now download to the unit.

If the project does not download to the CTIU and an error is encountered which details an incorrect PLC protocol selection, the correct protocol needs to be downloaded prior to a successful download. The correct PLC protocol is determined based on the driver selected for communications. Here "Control Techniques", Drive, 2<sup>nd</sup> proc or MC using Modbus RTU, which requires ctrtu\_Rx.1xx protocol.



Click on "File", "Update Tiu Protocol", to load the correct operating protocol. The file selection area will default to the exact protocol that is needed based on the communications configuration that was selected.

After this file is chosen, the download will begin. This download will take a couple of minutes to complete.

After the protocol download is complete, download the actual program to the unit.

#### Questions ?? Ask the Author:

**Author:** Tom Schrauth  
952-995-8168

**e-mail :** [Tom.Schrauth@emersonct.com](mailto:Tom.Schrauth@emersonct.com)



**SCIGATE AUTOMATION (S) PTE LTD**  
No.1 Bukit Batok Street 22 #01-01 Singapore 659592  
Tel: (65) 6561 0488 Fax: (65) 6562 0588  
Email: [sales@scigate.com.sg](mailto:sales@scigate.com.sg) Web: [www.scigate.com.sg](http://www.scigate.com.sg)  
Business Hours: Monday - Friday 8.30am - 6.15pm