

CCU/485-TNMB

Installation and Configuration Manual



TABLE OF CONTENTS

1. GENERAL.....	4
1.1. INTRODUCTION.....	4
1.2. ORDERING CONFIGURATIONS:	5
1.3. SYSTEM REQUIREMENTS FOR CCU-TNMB SETUP PROGRAM	6
2. INSTALLATION.....	7
2.1. SUPPLIED:	7
2.2. YOU WILL NEED:.....	7
2.3. POWER REQUIREMENTS:	7
2.4. BOARD LAYOUT:.....	7
2.5. INSTALLING THE MOUNTING BRACKET:	8
2.6. CONNECTING THE CCU/485-TNMB:	8
2.6.1. <i>Connecting the CCU/485-TNMB to the MINERVA:</i>	9
2.6.2. <i>Connecting the CCU/485-TNMB to the MODBUS Network:</i>	14
2.6.3. <i>Connecting the CCU/485-TNMB to the CCUioNET</i>	15
2.6.4. <i>Connecting the CCU/485-TNMB to Power</i>	16
2.6.5. <i>Setting the MODBUS Address of the CCU/485-TNMB</i>	16
2.7. NORMAL AND FAULT INDICATION	17
2.7.1. <i>Operation of LEDs</i>	17
2.7.2. <i>Supervision Relay</i>	17
3. CONFIGURATION.....	18
3.1. INTERFACING TO THE MINERVA MODBUS.....	18
3.1.1. <i>General</i>	18
3.1.2. <i>The CCU/485-TNMB Global Map</i>	19
3.2. CONFIGURING THE MODBUS MAP	22
3.2.1. <i>The CCU-TNMB Setup program</i>	22
3.2.2. <i>General configuration</i>	23
3.2.3. <i>Opening, Saving and restoring CCU/485-TNMB configurations</i>	23
3.2.4. <i>Adding to the global map</i>	23
3.2.5. <i>Configuring CCUioNET nodes</i>	24
3.2.6. <i>Changing the Submap</i>	24
3.2.7. <i>Setting the PTP parameters</i>	26
3.2.8. <i>Changing the READ and WRITE Registers MODBUS Command</i>	27
3.2.9. <i>Programming the CCU/485-TNMB or CCU/232-TNMB</i>	28
4. COMMUNICATION.....	29
4.1. MODBUS COMMUNICATION PARAMETERS.....	29
4.2. MODBUS COMMANDS.....	29
4.2.1. <i>Read Input Status</i>	29
4.2.2. <i>Read Input Registers</i>	30
4.2.3. <i>Force Multiple Coils</i>	31
4.2.4. <i>Preset Multiple Registers</i>	32
4.2.5. <i>Calculating CRC16</i>	33
5. EXAMPLE CCU/485-TNMB SETUP.....	34
5.0.1. <i>Set the MODBUS address</i>	34
5.1. SET UP THE GLOBAL MAP	35
5.1.1. <i>Add the CCUioNET nodes</i>	35
5.1.2. <i>Add the MINERVA panels</i>	35
5.2. SET UP THE SUBMAP	36
5.2.1. <i>Adding the MINERVA Zone Alarms to the Submap</i>	36
5.2.2. <i>Adding the Blank addresses to the Submap</i>	36
5.2.3. <i>Adding the CCU/IO-485 Active bits to the Submap</i>	37
5.3. SAVE THE PROJECT AND PROGRAM THE CCU/485-TNMB	38

5.3.1.	<i>Saving</i>	38
5.3.2.	<i>Programming</i>	38
APPENDIX A:	CONNECTING THE CCU/485-TNMB TO THE MINERVA:	39
APPENDIX B:	CONNECTING THE CCU/485-TNMB TO THE MODBUS NETWORK:	40

1. General

1.1. Introduction

The CCU/485-TNMB and CCU/232-TNMB provide a MODBUS interface to a number of MINERVA panels on a THORNNET. CCU/IO boards may also be connected to provide general I/O devices accessed through the MODBUS interface.

The CCU/485-TNMB connects to one of the MINERVA panels that are on the THORNNET via a TLT-530 (TPI) interface card using RS232. It connects to MODBUS via an RS485 connection. Another RS485 port allows up to 8 CCU/IO boards to be connected.

The CCU/232-TNMB connects to one of the MINERVA panels that are on the THORNNET via a TLT-530 (TPI) interface card using RS232. It connects to MODBUS via an RS232 connection. Another RS232 port allows 1 CCU/IO board to be connected.

Each CCU/IO has 8 relay outputs, which can be used as inputs to the Minerva. These contacts are controlled via WRITE commands to the MODBUS map. Each CCU/IO also has 8 supervised inputs whose status can be read from the MODBUS map.

The MODBUS map of the CCU/485-TNMB is configured using the **CCU-TNMB Setup** program. The MODBUS map can be accessed directly via the Global Map or through a Submap:

- Global Map– Each MINERVA panel and CCU/IO board is added to the map as a static item, the user gets no choice as to the positions of bits within the panel (viz Modbus addresses). Each MODBUS address points to the corresponding address in the Global Map
- Custom Map Read– Up to 1000 MODBUS addresses are configured individually by the user to point to any address in the Global Map.

1.2. Ordering Configurations:

There are two configurations. The first is as a bare board supplied on a mounting bracket. This is suitable for mounting within existing enclosures (example: Fire Indicator Panel) Part Number: CCU/485-TNMB. The second configuration is within an aluminium box. Part Number CCU/485-TNMB-B. This is also supplied with a mounting bracket to facilitate wall/panel mounting of the box.

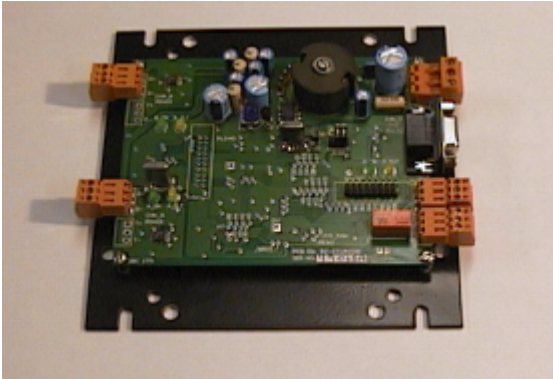


Figure 1:CCU/485-TNMB



Figure 2: CCU/485-TNMB-B

1.3. System Requirements for CCU-TNMB Setup program

The minimum requirements for the computer system are:

- 5 MB of disk space for installation.
- 16MB Ram
- SVGA Monitor
- Windows95® or 98 or NT operating system
- Microsoft® Windows compatible pointing device (such as a mouse)
- A communications port for reprogramming the CCU/485-TNMB.

2. Installation

This section describes the how to physically install the CCU/485-TNMB and provides no information on configuration.

2.1. Supplied:

- 1 x CCU/485 board with firmware for TNMB (MODBUS)
- 1 x Mounting Bracket
- 4 x M4 Machine Screws
- 4 x Flat Washers
- 1 x Power connector for CCU/485
- 1 x Supervision relay connector for CCU/485
- 1 x TPI530 interface cable from TLT-530 to CCU/485-TNMB
- 1 x Floppy disk with TNMB software

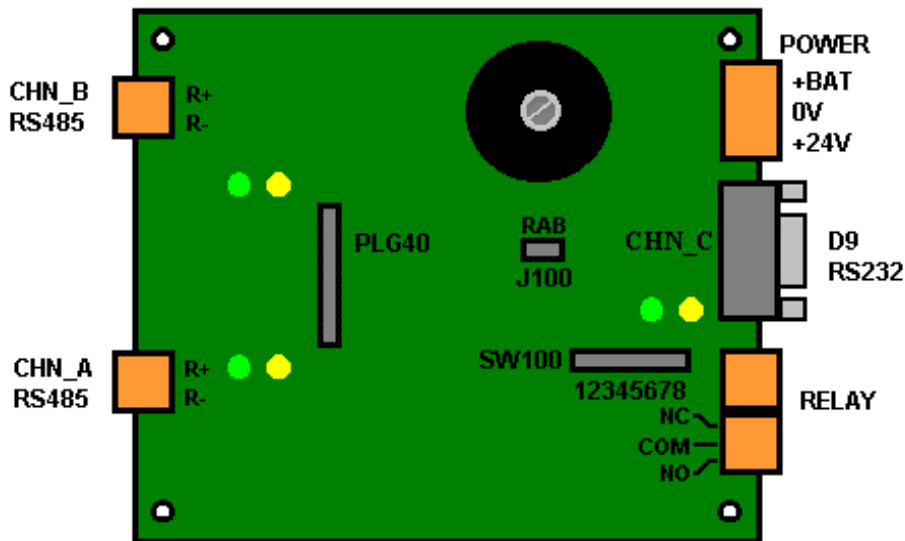
2.2. You will need:

- 1 x 24V DC plug pack or Power Source, optionally 24DC battery backup input

2.3. Power Requirements:

Typical: 150mA at 24VDC Voltage Input: 20-30 VDC

2.4. Board Layout:



2.5. Installing the mounting bracket:

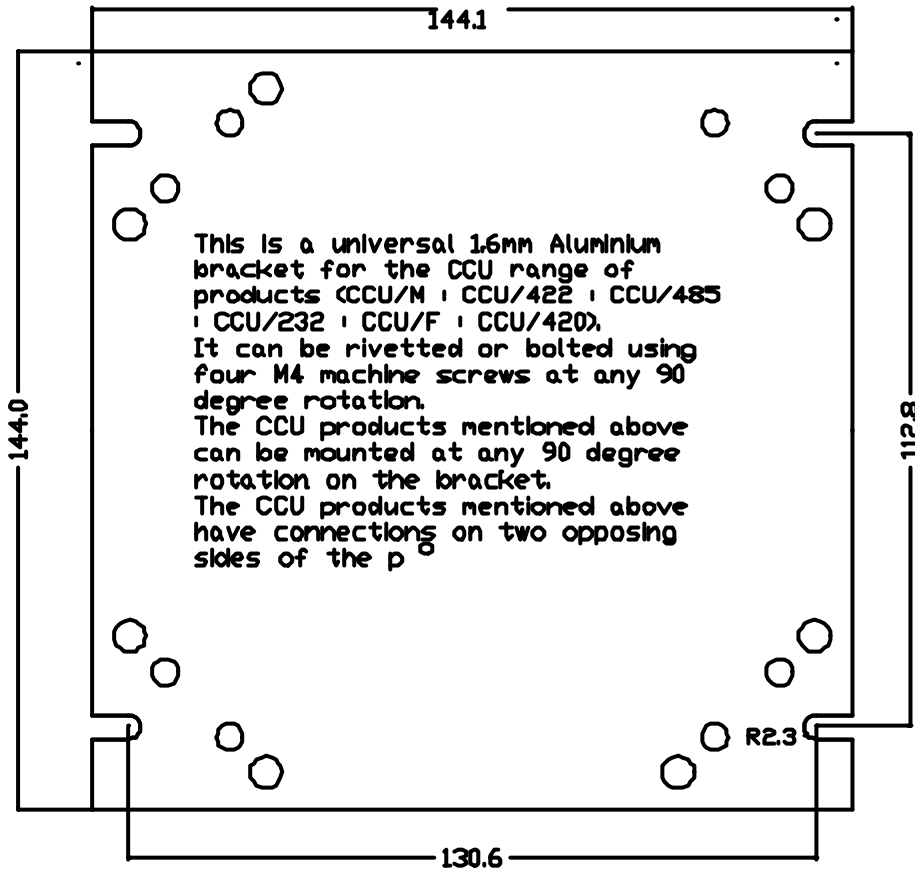


Figure 3: Mounting Bracket

2.6. Connecting the CCU/485-TNMB:

Channel C on the CCU/485-TNMB connects to a TLT-530 (TPI) network interface on the Minerva. Channel B connects to the MODBUS. Channel A may be connected to a CCUioNET or CCU/I boards, for added inputs and outputs.

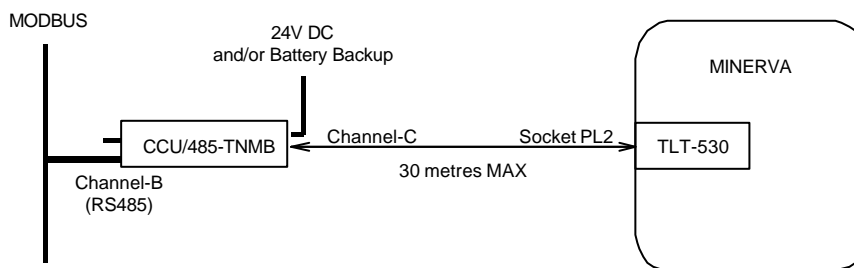


Figure 4: General Configuration

2.6.1. Connecting the CCU/485-TNMB to the MINERVA:

Each MINERVA Fire Panel setup requires at least 2 TLI-530 Network Boards in order to connect to the TNMB. The first TLI-530 board is mounted directly onto the MINERVA MP-500 Main Processor board. It is installed using the 20pin through board connectors from the main processor board to the TLI-530. As seen below these 20 pins align with P1 and P1A on the TLI-530 board. This board is configured to be a Gateway board (SW1-1 ON, SW1-2 ON, SW1-3 ON).

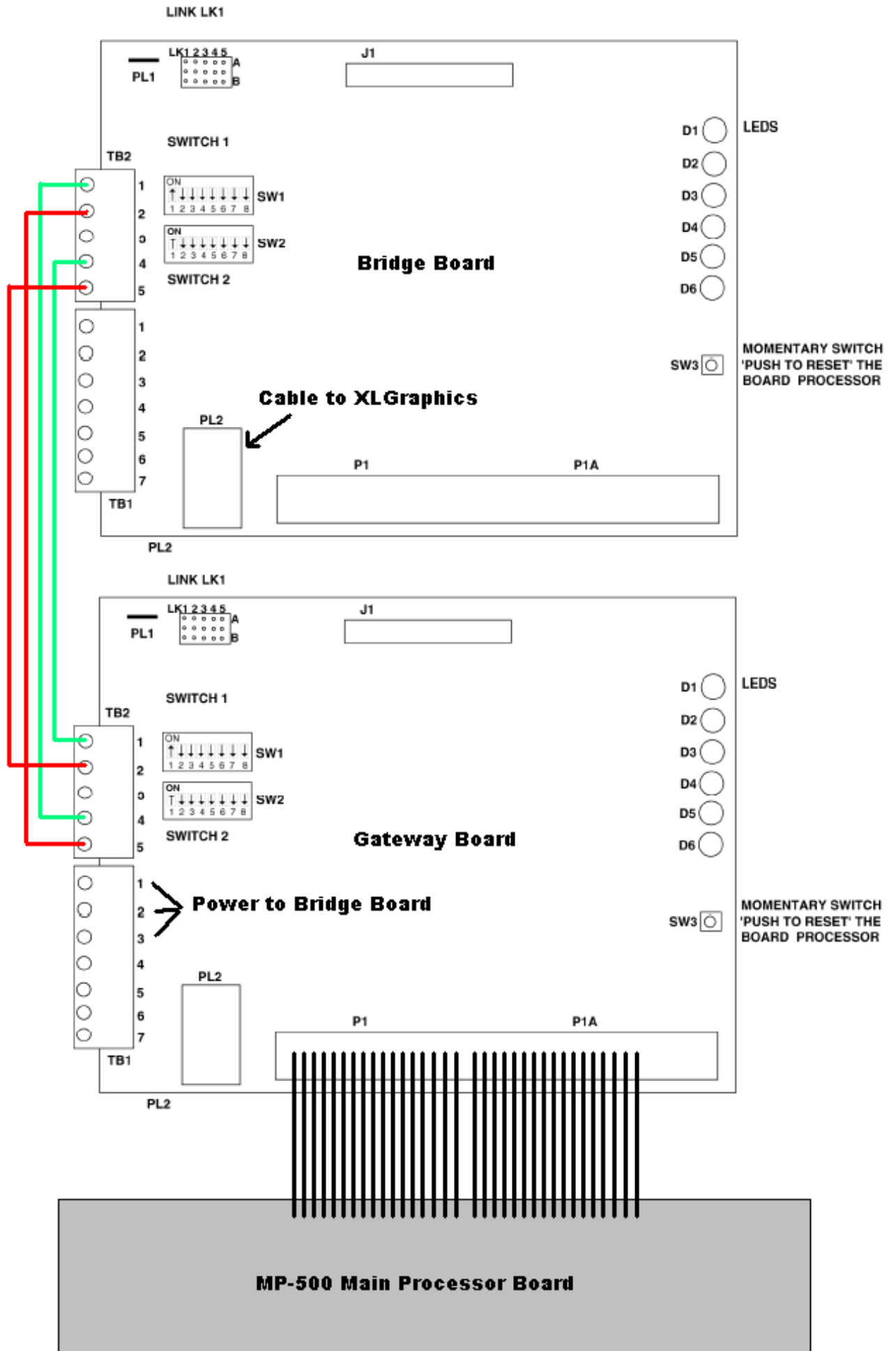
The firmware for this board is the standard ThornNet TLI-530 networking firmware.

A second TLI-530 board is mounted directly on top of the first TLI-530 connected to it by Terminal Blocks 1 and 2 (TB1 and TB2). Please see the TLI-530 Assembly manual, Publication Number 19700344, for the wiring of the 2 terminal blocks. The second board is configured as a Bridge board (SW1-1 ON, SW1-2 OFF, SW1-3 ON).

It is this second board that the telephone connector plugs into.

The firmware for this second board is the TPI Version 1.01 (or later).

Please see next page for image.



Use the supplied adaptor cable, or a cable you have made, to connect the telephone connector into the socket on the TLI-530.

Connect the D9 connector of the TLI network cable to CHN C on the CCU.

ATTN: Before changing any of the DIP switch settings ensure all power to the panel is OFF.

There are 4 settings to be configured on the TLT-530 Interface module. These settings are distributed across the 2 DIP switches on the module, SW1 and SW2. The 4 settings to be configured are:

1. The TLT-530 module's application
2. The Network Baud Rate
3. The Network Topology
4. Port Function

The DIP switch settings have the following options for the TPI configuration setup.

SW1 Switch Position			Application	Assembly	SW1 Switch Position			Network BAUD Rate	SW1 Position		Network Topology
1	2	3			4	5	6		7	8	
ON	ON	ON	TFXnet Gateway	TLI-530	ON	ON	ON	1200	ON	ON	Bus
ON	ON	OFF	TFXnet Hub	TLI-530	OFF	ON	ON	2400	ON	OFF	Star
ON	OFF	ON	TFXnet Bridge	TLI-530	ON	OFF	ON	9600	OFF	ON	Ring
ON	OFF	OFF	Not Used	N/A	OFF	OFF	ON	19200	OFF	OFF	Redundant Star
OFF	ON	ON	TFX-to-XA Normal Operation	TLX-530	ON	ON	OFF	38400			
OFF	ON	OFF	TFX-to-XA Config. Program	TLI-530 TLX-530	OFF	ON	OFF	Reserved			
OFF	OFF	ON	Reserved	N/A	ON	OFF	OFF	Reserved			
OFF	OFF	OFF	Reserved	N/A	OFF	OFF	OFF	Reserved			
ON	ON	ON	Third Party Interface	TLT-530							

SW2 Switch	Value (OFF)	SW2 Position 8	Function
1	1	ON	TFX TTL Port
2	2	OFF	RS232
3	4		
4	8		
5	16		
6	32		
7	64		

The DIP switch settings that can to be used for our needs are the following.

SW1 Pos No:			Applicatio n	SW1 Pos No:			Network BAUD Rate	SW1 Pos No:		Network Topology
1	2	3		4	5	6		7	8	
ON	OFF	ON	Bridge App	OFF	OFF	ON	19200	OFF	ON	Ring

SW2 Pos No:							Network Address	SW2 Pos No:	Host Comms
1	2	3	4	5	6	7		8	
OFF	OFF	ON	ON	ON	OFF	ON	35 (example)	ON	TTL
ON	OFF	ON	ON	ON	ON	ON	2 (example)	OFF	RS232
1	2	4	8	16	32	64	Value when OFF		

As shown above the following settings have been selected. These settings should be used for a MINERVA Panel connected to a CCU/TNMB.

DIP Switch 1

Firstly the TLT has been acknowledged for use in a Bridging Application (DIP 1 - ON, DIP 2 - OFF, DIP 3 - -ON).

The Network BAUD Rate for the TFXnet has been set at 19200 (DIP 4 - OFF, DIP 5 - OFF , DIP 6 - ON)

A Network Topology of a Ring configuration has been set (DIP 7 - OFF and DIP 8 - ON)

DIP Switch 2

The TLT-530 modules address is specified using the first 7 switches. There have been 2 examples provided. Its address must be a unique address on the TFXnet network.

The final switch determines the wiring for the TLT-530. For our application the RS232 communication wiring is used.

When installing the TLI-530 interface be sure to had it to the configuration in the CONSYS MINERVA panel programmer. Below is an outline in how to do this.

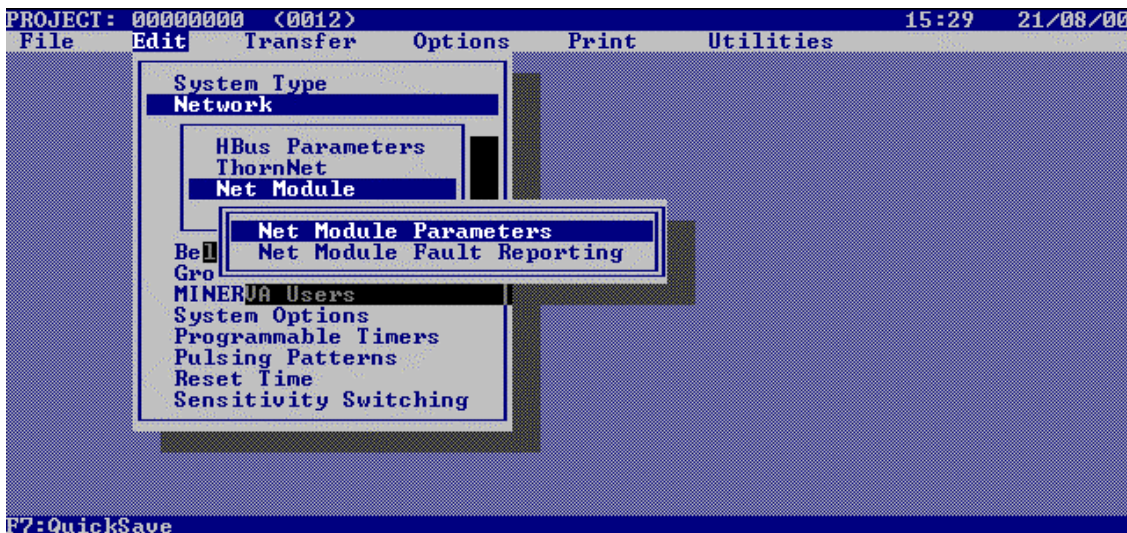


Figure 5: CONSYS Net 1

The image above shows where the TLI-530 Network Module can be added to the panel. In CONSYS, the Network section of the Edit menu will give 3 options that will setup the MINERVA Panel network configuration.

To configure the TPI Module for use with XLGraphics or the CCU/TNMB the following ring setup is used. It is recommended that the settings below be used for the module.

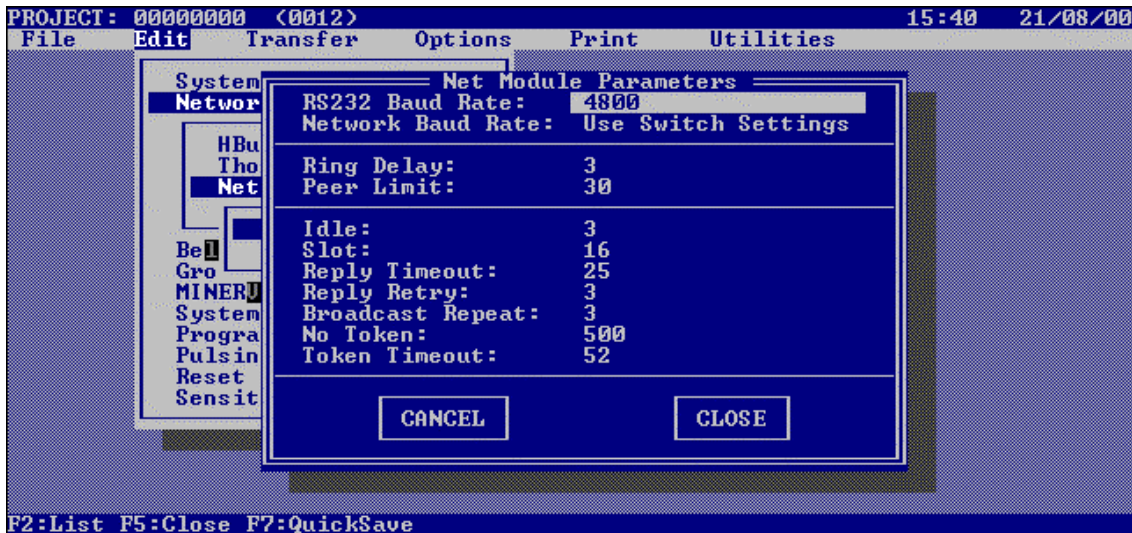


Figure 6: CONSYS Net 2

For the module's Fault Reporting setup the image below shows the parameters that should be used.



Figure 7: CONSYS Net 3

See Appendix A

2.6.2. Connecting the CCU/485-TNMB to the MODBUS Network:

See Appendix B

2.6.3. Connecting the CCU/485-TNMB to the CCUioNET

To connect the CCU/485-TNMB to a chain of CCU/IO-485s or CCU/I-485s connect the RS485 (SKT33) port of the CCU/IO-485 (or CCU/IO-485) to CHN_A CCU/485-TNMB using the following connection diagram (Figure 7). Up to eight CCU/IO-485 boards can be chained off the CCU/485-TNMB. Appendix A contains information on wiring for this configuration. Check the CCU/IO-485 documentation for jumper settings for this configuration.

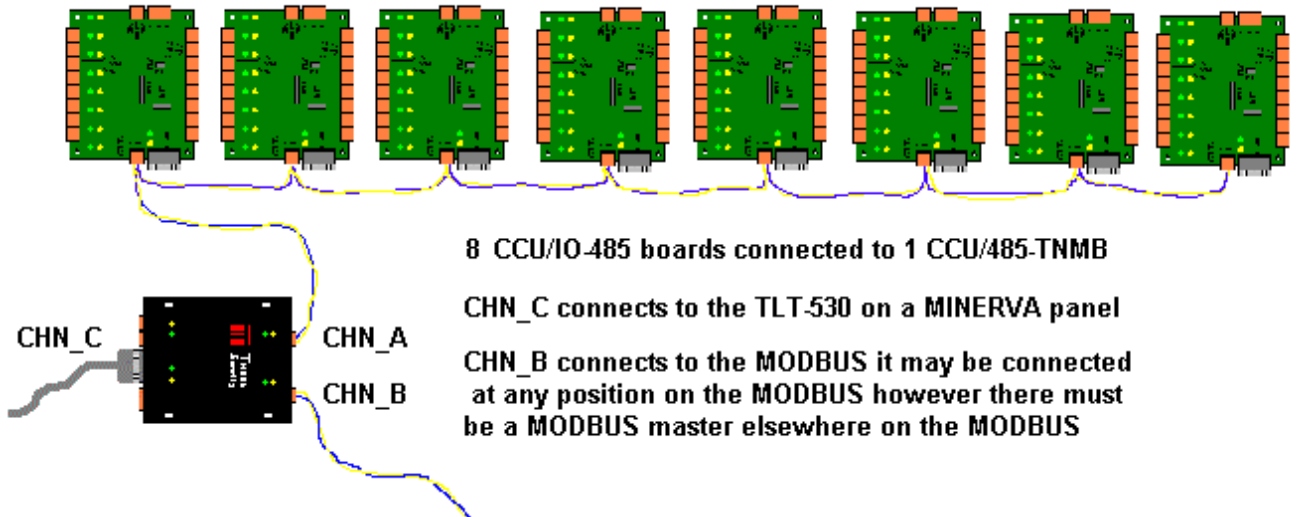


Figure 8: CCUioNET connection on a CCU/485-TNMB

CCU/IO-485 RS485 (SKT33)	CCU/485-IO Channel A
+	R+
-	R-
middle pin	middle pin

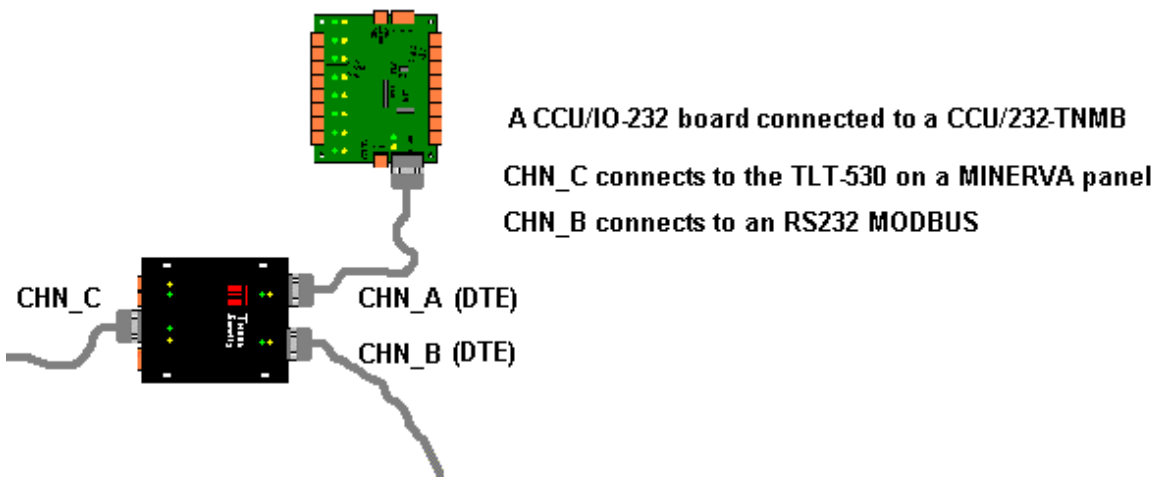


Figure 9: CCUioNET connection on a CCU/232-TNMB

If connecting to a PC MODBUS master use a null modem cable.

2.6.4. Connecting the CCU/485-TNMB to Power

Connect the 24V DC power source to the supplied connector. The power connector is next to the CHN_C port. Pin connections are:

PWR (+24VDC) RETURN/COM (0V) BATTERY (24V)

These connections are written on the board. If the 24V power source is uninterruptible, then there is no need to connect the BATTERY connection.

Pin number	Function
1	24V
2	COM
3	BAT

2.6.5. Setting the MODBUS Address of the CCU/485-TNMB

The MODBUS address of the CCU/485-TNMB is set through the **CCU-TNMB Setup** program.

2.7. Normal and Fault Indication

2.7.1. Operation of LEDs

Each port has a Green and a Yellow LED. The green led flashes when data is transmitted on the corresponding port. The yellow LED glows when there has been supervision failure on the port or it is unable to make a connection. If a port has been disabled in the software then neither LED will glow.

Port A (connected to CCUioNET):

- If CCUioNET is disabled then both LEDs will remain extinguished.
- Green LED Flashes to indicate Data Transmission.
- Yellow LED indicates Supervision failure.

Port B (MODBUS RS485 port):

- Green LED Flashes to indicate Data Transmission.
- Yellow LED indicates Supervision failure.

Port C (TLT530 port):

- Green LED Flashes to indicate Data Transmission.
- Yellow LED indicates Supervision failure.

2.7.2. Supervision Relay

The CCU/485-TNMB will supervise the MODBUS connection. In the event of supervision failure of the MODBUS, the relay on the CCU/485-TNMB will operate. This can optionally be connected to an input on the MINERVA for annunciation of MODBUS fault.

3. Configuration

3.1. Interfacing to the Minerva MODBUS.

3.1.1. General

The CCU/485-TNMB connects to a MINERVA MODBUS as a slave. It recognises 4 MODBUS commands:

Code	Description
2	Read Input Status
4	Read Input Registers
15	Force Multiple Coils
16	Preset Multiple Registers

The CCU/485-TNMB is programmed with a Global map of all of the MINERVA panels and CCUioNET devices (CCU/I or CCU/IOs) connected to it. The only part of this panel that is configurable is the order in which the panels and CCUs appear, the bits within the panels and CCUs are in a fixed order and are of a fixed size.

The CCU/485-TNMB MODBUS map can be accessed by the MODBUS master in two different ways:

1. Direct access to the first 65535 bits of the Global Map.
2. Through a Submap which mirrors 1000 bits of the global map. This Submap is totally configurable and each bit in the Submap can mirror any bit in the Global map.

Note: The Submap always occupies the first 1000 bits of the MODBUS map.

The MODBUS map is made up of two maps a register map and a bit map. The bit map on the CCU/485-TNMB contains the Global map and the Submap and reflects bits on MINERVA panels or on CCUioNET devices. The register map is simply 16, 16-bit registers, which may be used for any purpose, they do not reflect anything on the MINERVA panels or CCUioNET devices.

3.1.2. The CCU/485-TNMB Global Map

The Global Map exists in all CCU/485-TNMBs, however it is not necessary to know the structure of it unless the user is addressing it directly. If the Submap is used then this section will be of no use to the user.

In memory the MODBUS map is arranged in groups of 8192 bits. The first block of 8192 bits is reserved to include the Submap and the CCUioNET map. Each MINERVA panel (such as a MINERVA 16E (TFX-500) or MINERVA 80 (TFX-800)) is given a following block of 8192 bits which fully describe it. This global map can be addressed directly or addressed via a Submap of 1000 bits which points to specific bits out of the global MODBUS map. This must be set up in the **CCU-TNMB Setup**. It is not necessary to understand the layout of the global map if only a Submap is going to be used, as will be the case in most installations.

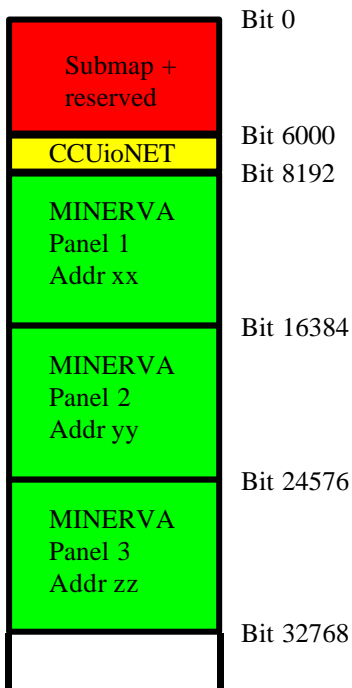


Figure 10: MODBUS map overview

The Submap takes up the first 1000 bits and is configured by the user. From 100 to 6000 is reserved for later use. Within each block of 8192 bits the bits are arranged as such:

CCUioNET block

Bit0-7	CCUioNET Node offline
Bit8-519	128 inputs with 4 bits each Bit 0 – active Bit 1 - supervision fault Bit 2 - reserved Bit 3 - reserved
Bit520-584	64 outputs (relay energise bits)

Panel block

Panel status bits

Bit0	common alarm
Bit1	common fault
Bit2	common isolate
Bit3	
Bit4	
Bit5	
Bit6	
Bit7	
Bit8	printer online
Bit9	keyswitch 1 on
Bit10	keyswitch 2 on
Bit11	
Bit12	
Bit13	
Bit14	
Bit15	

Zone bits

Bit40-363	Zones 0-80 with 4 bits each Bit 0 - alarm Bit 1 - fault Bit 2 - isolated Bit 3 - status
-----------	---

Point bits

Bit400-911	Loop A, Points 0-127 with 4 bits each Bit 0 - alarm Bit 1 - fault Bit 2 - isolated Bit 3 - status
Bit912-1423	Loop B, Points 0-127 with 4 bits each
Bit1424-1935	Loop C, Points 0-127 with 4 bits each
Bit1936-2447	Loop D, Points 0-127 with 4 bits each
Bit2448-2959	Loop E, Points 0-127 with 4 bits each
Bit2960-3471	Loop F, Points 0-127 with 4 bits each
Bit3472-3983	Loop G, Points 0-127 with 4 bits each
Bit3984-4495	Loop H, Points 0-127 with 4 bits each
Bit4496-5007	Loop J, Points 0-127 with 4 bits each
Bit5008-5519	Loop K, Points 0-127 with 4 bits each

Bits 5520-8191 are reserved.

For a MINERVA 16E only Loop A and Loop B are used and Zone 0 to 16 are used, these are still at the same memory locations.

3.2. Configuring the MODBUS Map

The MODBUS map must be configured using the **CCU-TNMB Setup** program. The program allows the user to add MINERVA panels and CCUioNET devices into the Global Map. For most applications the most useful feature is the ability to make a Submap of up to 1000 bits from the bits in the global map.

The Submap allows the user to mirror 1000 bits from the global map and into the Submap from address position 0 to 999. For instance bit 8345 in the global map may be mapped to bit 0 in the Submap. This gives the user flexibility in the layout of the MODBUS map.

3.2.1. The CCU-TNMB Setup program

The **CCU-TNMB Setup** program is designed for use with the CCU/485-TNMB. It allows the user to configure the MODBUS map on the CCU/485-TNMB and download the configuration to the CCU/485-TNMB.

When the **CCU-TNMB Setup** program is executed a screen similar to figure 10 (below) should appear.

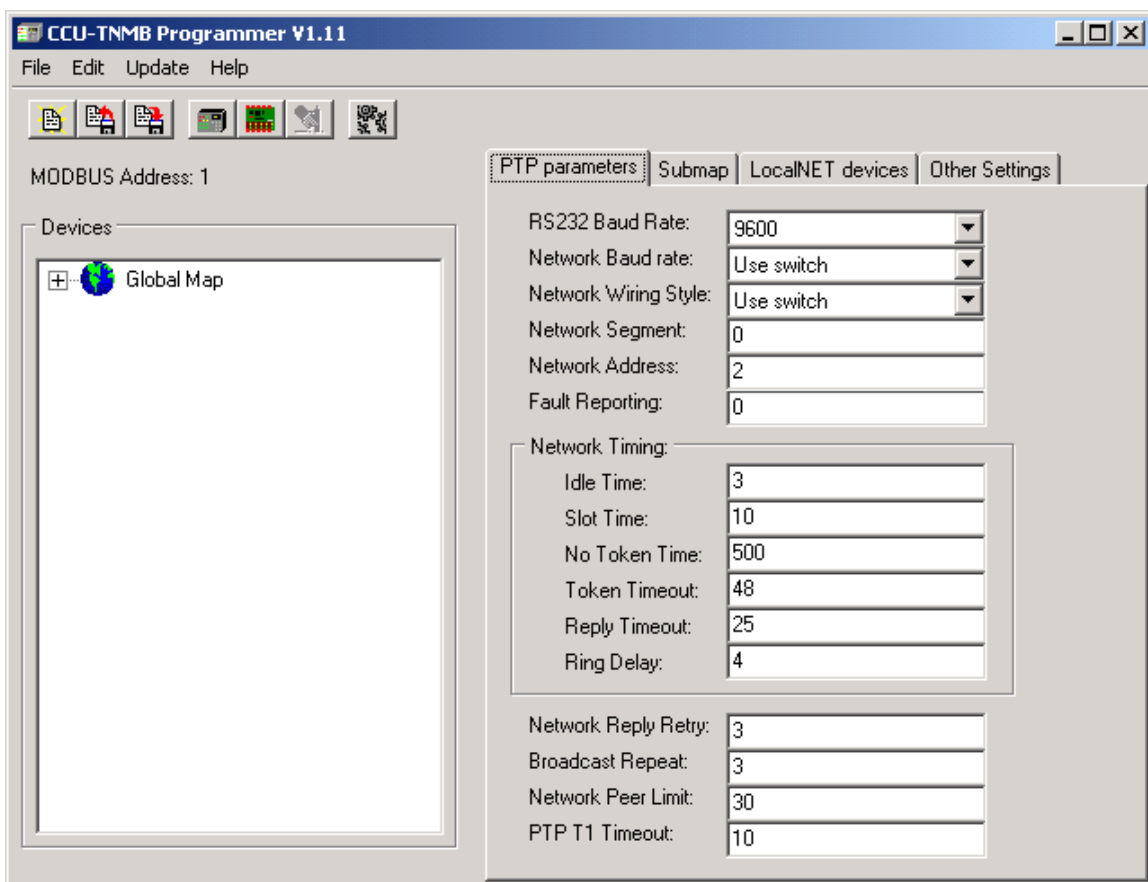


Figure 11 CCU-TNMB Setup main screen

On the left is a tree which is able to be expanded or contracted by clicking the '+' and '-' signs. The tree shows all of the MINERVA panels and all CCUioNET devices that are set up in the CCU/485-TNMB, this list is called the **Global Map Tree**.

The grey bar with the words **File, Edit, Update** and **Help** on it is referred to as the Main Menu for the remainder of this document. Through the Main Menu most options can be accessed. In this text if, for instance, it is mentioned to access **File | New** then this means click on **File** in the Main Menu, wait for a drop down menu to appear and then click on **New**.

The buttons below the Main Menu are equivalent to some items in the Main Menu. Because of their common usage they are available as buttons for easy access.

3.2.2. General configuration

The CCU/485-TNMB is connected to the MODBUS and therefore must be given an address on the MODBUS. The MODBUS address may be changed by selecting **Edit | Change MODBUS Address** from the Main Menu.

3.2.3. Opening, Saving and restoring CCU/485-TNMB configurations



The user may clear the current configuration and start a new configuration by pressing the **New** button or selecting **File | New** from the Main Menu.



A configuration can be saved by pressing the **Save** button or selecting **File | Save** from the Main Menu. All saved files are given the extension .TNM by default.



A configuration can be loaded by pressing the **Load** button or selecting **File | Load** from the Main Menu.

3.2.4. Adding to the global map



To add more MINERVA panels to the global map click the **Add MINERVA panel** button.



To add more CCUioNET devices to the global map click the **Add CCUioNET node** button.

For both the above a new window should appear giving a list of devices to choose from. Select the correct device and fill in a brief description for easy identification. When adding a MINERVA panel the user will also have to fill in the TFXNet address, which is the address of the MINERVA panel on the THORNET.

The figure shows two dialog boxes side-by-side. The left dialog is titled 'Add Minerva Panel' and contains a 'Panel Type' section with two radio buttons: 'Minerva 16E (TFX500)' (selected) and 'Minerva 80 (TFX800)'. Below this is a 'Description' text box and a 'TFXNet Address' text box containing the number '1'. At the bottom are 'Cancel' and 'OK' buttons. The right dialog is titled 'Add CCUioNET Node' and contains a 'Node Type' section with two radio buttons: 'CCU/IO-485' (selected) and 'CCU/I-485'. Below this is a 'Description' text box. At the bottom are 'Cancel' and 'OK' buttons.

Figure 12: Add new device screens



To remove a panel or CCUioNET node select the panel in the **Global Map Tree** the press the **Delete** button or select **Edit | Remove Panel** from the Main Menu

3.2.5. Configuring CCUioNET nodes

The inputs of the CCUioNET nodes must have their supervision set to either supervised or unsupervised, depending on how the CCU/IO-485s are configured. All of the CCUioNET inputs and outputs are configurable, to either read as **set** or **cleared** under normal conditions. Both the inputs and the outputs can operate in normal or reversed mode. In normal mode the relay outputs read **set** when the relays are energised, in reversed mode they would read **cleared** for this case. In normal mode the 'active' bit is **set** for a normal input, in reversed mode the 'active' bit is **cleared** for a normal input. The 'input fault' bit is not affected by the normal or reverse modes.

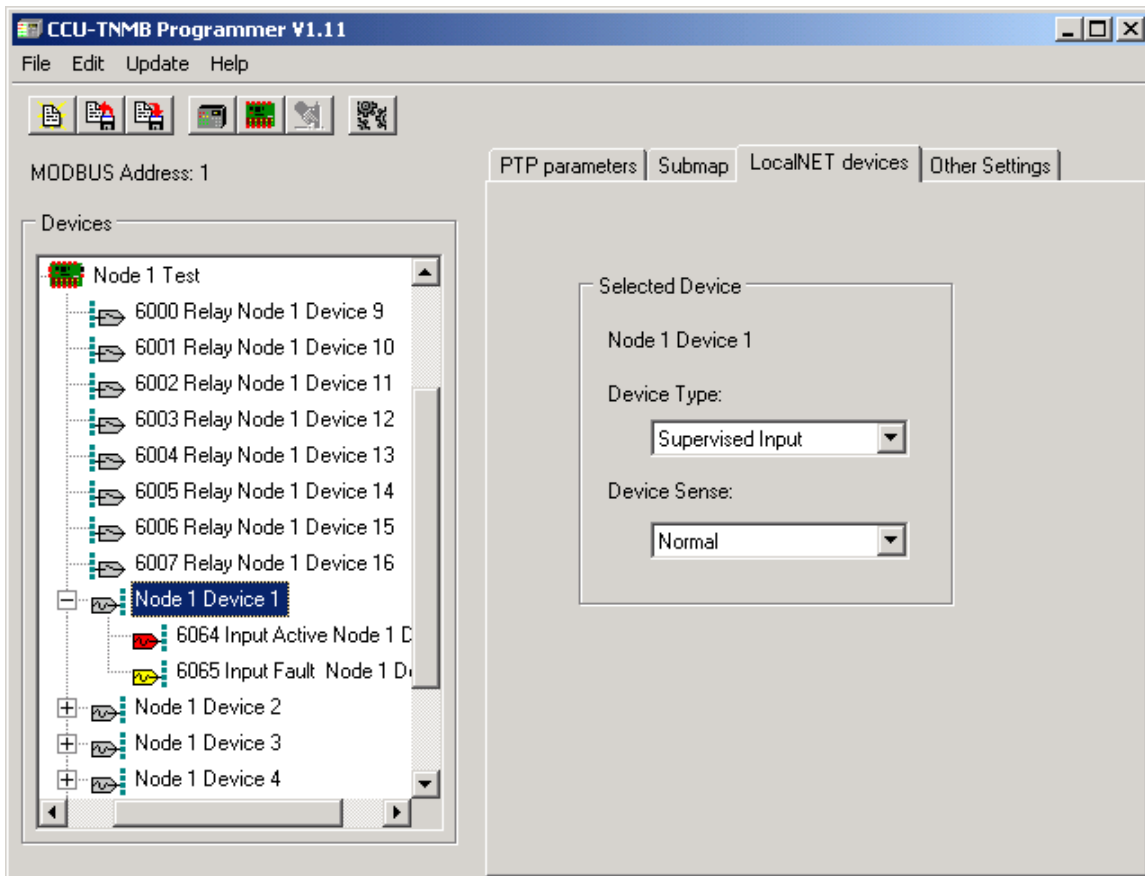


Figure 13: Configuring CCUioNET nodes

3.2.6. Changing the Submap

A Submap is a map of up to 1000 bits which mimics 1000 bits from the global map. The user selects which bits to add to the Submap from the global map and also the order in which they are arranged. The Submap may include any bits from the global map including CCUioNET bits.

To add bits to a Submap go to the Submap page, by clicking on the Submap tab at the top of the window. To add a bit to a Submap find the bit that you wish to add in the Global map on the left hand side of the screen. Remember you can expand and withdraw branches by clicking the '+' and '-' signs. Select the bit then the click **Add** button. The bit should appear in the Submap on the right-hand side of the screen. If the **Add** button is 'greyed out' then a valid bit has not been selected.

If you wish the CCU to store the length of the Submap in the Submap you can use the **Count** button. This will add the relevant 16 bits (1 word) into the Submap. The bits are added into the Submap in Big Endian format (Motorola). With some manipulation they can be changed into Little Endian format. The bits can be added at any position in the Submap.

Only one individual bit may be added to the Submap at once, for instance an Point in a MINERVA panel may not be added to the Submap, because a point is made up of an Alarm bit, a Fault bit and an Isolate Bit. The only exception to this is a useful feature which allows the user to add all of the Zone Alarms, Zone Faults or Zone Isolates to a Submap at once. This is done by **Adding** 'All Zone Alarms', 'All Zone Faults' or 'All Zone Isolates' to the Submap. These three 'multiple' bits are found in the **Global Map Tree** under 'Zones', for each MINERVA panel. If they are added then the Submap shows all of the bits separately and they may be removed separately.

Blank bits, or unused addresses may be added to the Submap by adding the **Unused Entry** in the **Global Map Tree** to the Submap or clicking the **Unused** button.

To remove a bit from the Submap select the bit from the Submap on the right-hand side of the screen and then press the **Remove** button. The bit should disappear from the Submap.

It is possible to add a bit more than once to the Submap.

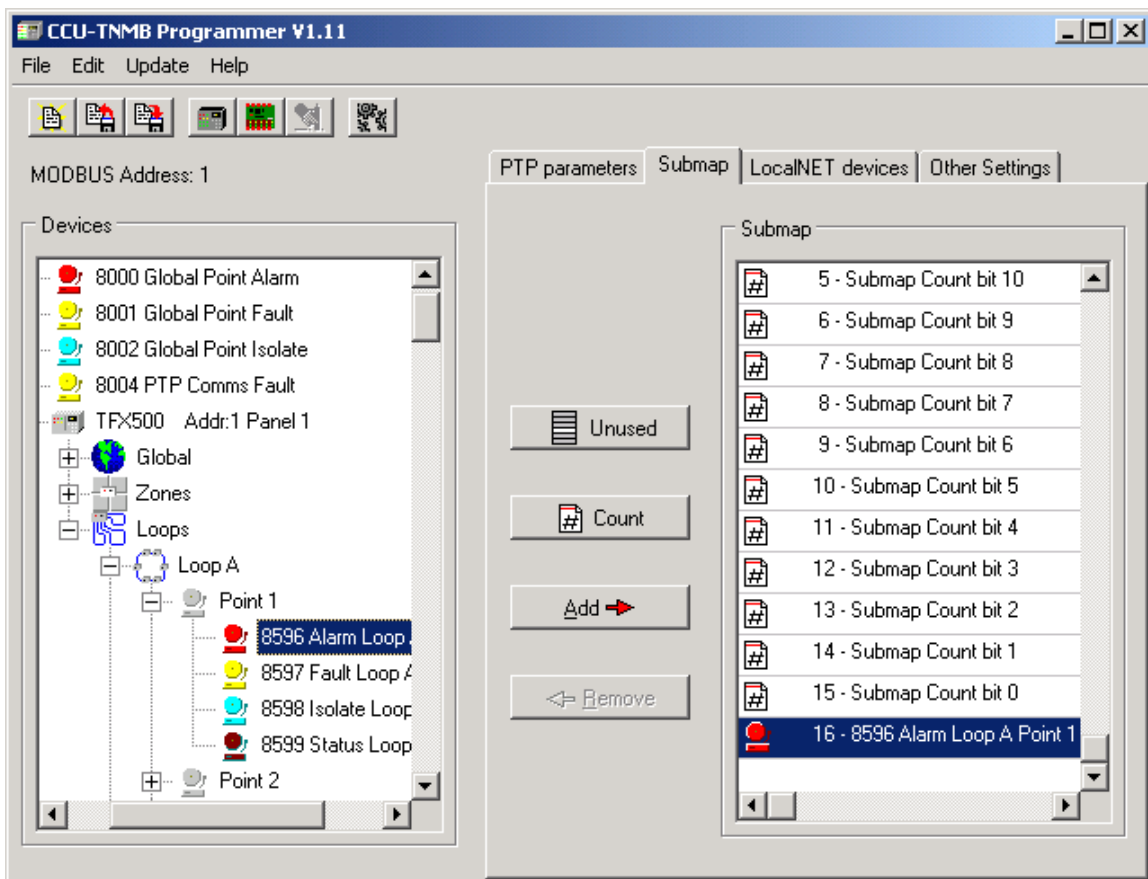


Figure 14: Adding to a Submap

3.2.7. Setting the PTP parameters

These parameters reflect the settings for the communication between the MINERVA panel and the CCU/485-TNMB. The **Network Address** must be set to the jumper address of the TLT-530 and the rest of the settings set to match the network. For further information and help with these settings refer to publication number 19700344 **Network Interface Module** and to **MinervaSpeak Reference Guide for Advanced Third Party Interface Applications**.

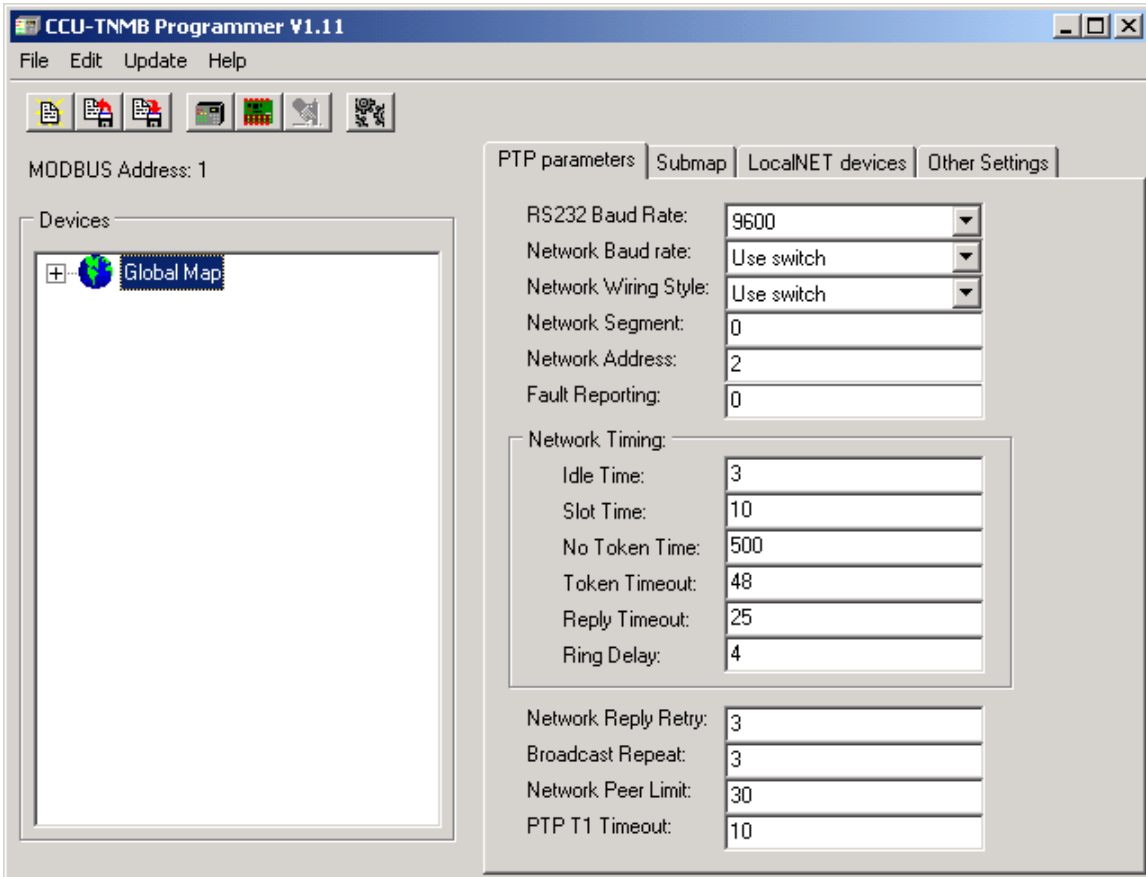


Figure 15: Setting PTP Parameters

3.2.8. Changing the READ and WRITE Registers MODBUS Command.

You may change the Read and Write Registers command function using this tab window. To use the Read and Write Registers command to read and write scratch registers in the CCU/485-TNMB select the 'Use registers' radio button. To use the Read and Write Registers command to read and write from the MODBUS bitmap select the 'Use bitmap' radio button.

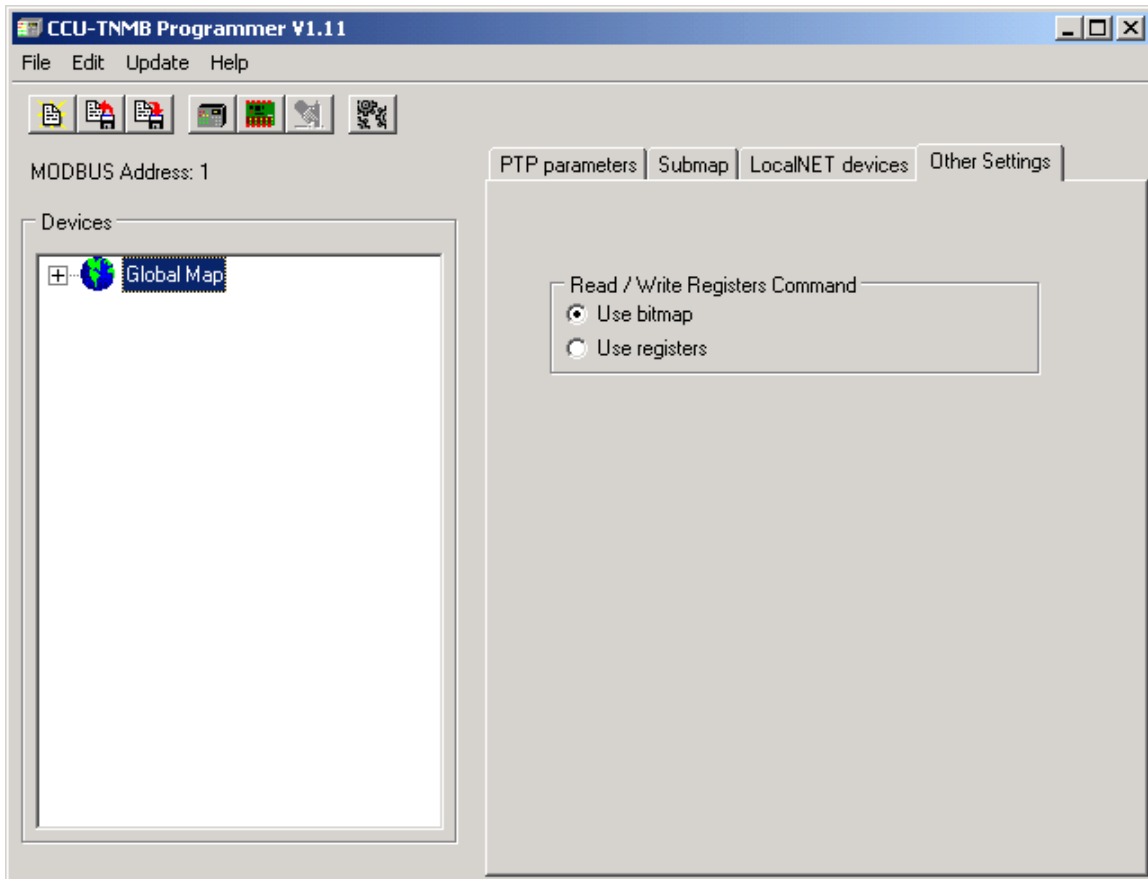


Figure 16: Setting Read and Write MODBUS Function

3.2.9. Programming the CCU/485-TNMB or CCU/232-TNMB

When the configuring process is complete the user should save the work and then program the CCU/485-TNMB.

To plug the computer into the CCU/485-TNMB connect a straight through standard communications port cable (DB9 socket to DB9 plug) from the desired COM port of the computer to CHN_C on the CCU/485-TNMB.

Before programming is commenced the comport must be set in the **CCU-TNMB Setup** program to the port that the CCU/485-TNMB is connected to, also the user must select whether a CCU/232-TNMB or CCU/485-TNMB is to be programmed. To do this select **Update | Programming Option** from the Main Menu.

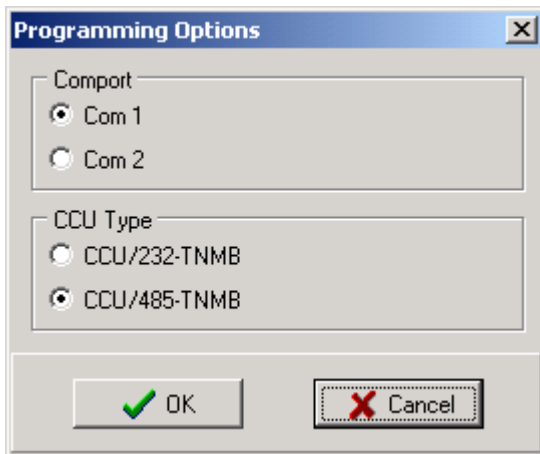


Figure 17: Programming Options

To program:

Make sure the CCU is powered off.

Press the **Update Data** button or select **Update | Data** from the Main Menu, then power on the CCU. Programming should commence.



4. Communication

4.1. MODBUS Communication Parameters

9600,n,8,1

4.2. MODBUS Commands

The CCU/485-TNMB connects to a MINERVA MODBUS as a slave. It recognises the following four MODBUS commands which the MODBUS master can use:

Code	Description
2	Read Input Status
4	Read Input Registers
15	Force Multiple Coils
16	Preset Multiple Registers

4.2.1. Read Input Status

Read Input Status reads a range of bits from the MODBUS map. This command can read up to 65535 bits starting at any address range from 0 to 65535 in the MODBUS map. This command is used to read from the Global map or the Submap on the CCU/485-TNMB. An example of a Read Input Status command:

Field	Example
MODBUS Address	5
MODBUS Function	2
Starting MODBUS Address (High)	1
Starting MODBUS Address (Low)	BF
Number of Bits (High)	0
Number of Bits (Low)	12
CRC16 (Low)	???
CRC16 (High)	???

NB. CRC16 must be calculated using the appropriate formula by the MODBUS master and added in the last word position (it takes up 2 bytes).

This command will read 0012hex (18 decimal) bits from MODBUS device 5 starting at MODBUS address 01BFhex (447 decimal).

The CCU/485-TNMB should return the bits from address 01BFhex to 01D1hex. The bits are packed into bytes in the reply message. Status is indicated as: 1 = ON; 0 = OFF. The first data byte of the reply contains the bits of the lowest 8 addresses, with the lowest or first address being represented by the LSB (Least Significant Bit).

If the number of addresses returned does not divide evenly into eight then the last byte is padded with zeros toward the MSB (Most Significant Bit).

Field	Example
MODBUS Address	5
MODBUS Function	2
Bytes Returned	3
Data (Address 01C6 – 01BF)	E5
Data (Address 01CE – 01C7)	4D
Data (Address 01D1 – 01CF)	02
CRC16 (Low)	???
CRC16 (High)	???

NB. CRC16 is calculated by the CCU/485-TNMB and takes up 2 bytes.

4.2.2. Read Input Registers

The Read Input Registers command allows the MODBUS master to read from the 16, 16 bit registers on the CCU/485-TNMB. This command can read up to 16 registers, each of 16 bits, starting at any address in the range 0 to 15 in the MODBUS map. An example of a Read Input Registers command:

Field	Example
MODBUS Address	5
MODBUS Function	4
Starting register (High) (Always zero)	0
Starting register (Low)	2
Number of Registers (High)(Always zero)	0
Number of Registers (Low)	1
CRC16 (Low)	???
CRC16 (High)	???

NB. CRC16 must be calculated using the appropriate formula by the MODBUS master and added in the last word position (it takes up 2 bytes).

This command will read register 2 from MODBUS device 5.

The CCU/485-TNMB should return the 16 bits with the status indicated as: 1 = ON; 0 = OFF. The first data byte of the reply is contains the higher byte of the first register to return.

Field	Example
MODBUS Address	5
MODBUS Function	4
Bytes Returned	2
Data (Register 2 (High))	C0
Data (Register 2 (Low))	4E
CRC16 (Low)	???
CRC16 (High)	???

NB. CRC16 is calculated by the CCU/485-TNMB and takes up 2 bytes.

4.2.3. Force Multiple Coils

Force Multiple Coils sets a range of bits in the MODBUS map. This command can set up to 65535 bits starting at any address range from 0 to 65535 in the MODBUS map. This command is used to set bits in the Global map or the Submap on the CCU/485-TNMB. Note that setting a bit in the MODBUS map has no effect on the MINERVA panels or any other input device, however it will report back this value until the bit is rewritten by another device. An example of a Force Multiple Coils command:

Field	Example
MODBUS Address	5
MODBUS Function	0F
Starting MODBUS Address (High)	3
Starting MODBUS Address (Low)	3B
Number of Bits (High)	0
Number of Bits (Low)	9
Bytes of Data	2
Data (Address 0342 – 033B)	C2
Data (Address 0345 – 0343)	01
CRC16 (Low)	???
CRC16 (High)	???

NB. CRC16 must be calculated using the appropriate formula by the MODBUS master and added in the last word position (it takes up 2 bytes).

The bits in the data are packed into bytes. Status is indicated as: 1 = ON; 0 = OFF. The first data byte contains the bits of the lowest 8 addresses, with the lowest or first address being represented by the LSB (Least Significant Bit). The last data byte is padded towards the MSB (Most Significant Byte) with zeros.

The CCU/485-TNMB should return the number of coils successfully set.

Field	Example
MODBUS Address	5
MODBUS Function	0F
Starting MODBUS Address (High)	3
Starting MODBUS Address (Low)	3B
Number of Bits (High)	0
Number of Bits (Low)	9
CRC16 (Low)	???
CRC16 (High)	???

NB. CRC16 is calculated by the CCU/485-TNMB and takes up 2 bytes.

4.2.4. Preset Multiple Registers

The Preset Multiple Registers command allows the MODBUS master to write to the 16, 16 bit registers on the CCU/485-TNMB. This command can write up to 16 registers, each of 16 bits, starting at any address in the range 0 to 15 in the MODBUS map. An example of a Read Input Status command:

Field	Example
MODBUS Address	5
MODBUS Function	10(hex)
Starting register (High) (Always zero)	0
Starting register (Low)	3
Number of Registers (High)(Always zero)	0
Number of Registers (Low)	2
Bytes of Data	4
Data (Register 3 (High))	1D
Data (Register 3 (Low))	F0
Data (Register 4 (High))	12
Data (Register 4 (Low))	69
CRC16 (Low)	???
CRC16 (High)	???

NB. CRC16 must be calculated using the appropriate formula by the MODBUS master and added in the last word position (it takes up 2 bytes).

This command will write register 3 on MODBUS device 5.

The CCU/485-TNMB should return the number of registers successfully written, the status is indicated as: 1 = ON; 0 = OFF.

Field	Example
MODBUS Address	5
MODBUS Function	10(hex)
Starting Register (High)(always zero)	0
Starting Register (Low)	3
Number of Registers (High)(always zero)	0
Number of Registers (Low)	2
CRC16 (Low)	???
CRC16 (High)	???

NB. CRC16 is calculated by the CCU/485-TNMB and takes up 2 bytes.

4.2.5. Calculating CRC16

Step 1 - Start with a 16-bit register set to FFFF hex. Call this the CRC16 register.

Step 2 - Exclusive OR the first eight-bit byte of the message with the low byte of the CRC16 register, putting the result in the CRC16 register.

Step 3 - Shift the CRC16 register one bit to the right, shifting in a zero for the MSB.

Step 4 - If the LSB in the CRC16 register is 0, repeat Step 3. If the LSB is 1, Exclusive OR the CRC16 register with A001 hex (1010 0000 0000 0001).

Step 5 - Repeat Steps 3 and 4 until eight shifts have been performed. When this is done, a complete eight-bit byte will have been processed.

Step 6 - Repeat Steps 2 ... 5 for the every byte in the message.

Step 7 – Place the CRC16 value in the message.

5. Example CCU/485-TNMB Setup

This section will step through an example configuration. The example configuration will map five devices (labelled 'A', 'B', 'C', 'D' and 'E'):

- 2 x MINERVA 16E panels connected to the THORNNET ('A' and 'B').
- 1 x MINERVA 80 panel connected to the THORNNET ('C').
- 2 x CCU/IO-485 boards on the CCUioNET ('D' and 'E')

The MODBUS map that we will require for this example project will list all of the zone alarms (excluding zone 0, the system zone) of the three MINERVA panels in the start of the address map. Leave some blank addresses and then have all of the active bits from CCU/IO 'D'. Next we will have 8 more blank addresses then active bits from CCU/IO from CCU/IO 'E'.

Address	
0	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Panel 'A' Zone Alarms	
16	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Panel 'B' Zone Alarms	
32	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Panel 'C' Zone Alarms	
	17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32
	33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48
	49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64
	65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
112	BLANK
128	BLANK
CCU/IO 'D' Active bits	
134	1 2 3 4 5 6 7 8
CCU/IO 'E' Active bits	
	1 2 3 4 5 6 7 8
	BLANK

Figure 18: The desired MODBUS Map

5.0.1. Set the MODBUS address

The first task is to set the MODBUS address of the CCU/485-TNMB. To do this select **Edit | Change MODBUS address** and enter the address.

5.1. Set up the Global Map

5.1.1. Add the CCUioNET nodes

Our example system has two CCUioNET nodes- the CCU/IO-485 boards 'D' and 'E'. To add a node press the **Add CCUioNET Node** button or select **Edit | Add CCUioNET node** from the Main Menu. Select CCU/IO-485 from the list of CCUioNET devices. Press **OK**. The new CCUioNET node should be listed in the **Global Map Tree** on the left hand side of the screen. You may need to expand the tree to see the node, to do this press the '+' next to 'Global Map' and also the '+' next to 'CCUioNET nodes'. To add a description to this node so it is not confused with other nodes double-click on the node in the **Global Map Tree**. The node number in the **Global Map Tree** should match the address on the CCU/IO-485 board. So CCU/IO-485 board 'D' has address 1 on the CCUioNET. Add CCU/IO-485 board 'E' in a similar way.

5.1.2. Add the MINERVA panels

Our example system has three MINERVA panels- two MINERVA 16E panels ('A' and 'B') and one MINERVA 80 panel (device 'C'). To add a panel press the **Add MINERVA panel** button or select **Edit | Add MINERVA panel** from the Main Menu. Select MINERVA 16E from the list of MINERVA devices. The **TFXNet Address** box must be filled in with the address of the MINERVA 16E panel on the TFXNet. Press **OK**. If you have already entered a panel with this address an error message will be displayed, otherwise the new MINERVA 16E should be listed in the **Global Map Tree** on the left hand side of the screen. You may need to expand the tree to see the node, to do this press the '+' next to 'Global Map'. To add a description to this panel double-click on the panel in the **Global Map Tree**.

Add MINERVA panel 'B' and 'C' in a similar way (remembering that device 'C' is a MINERVA 80 instead of MINERVA 16E).

5.2. Set up the Submap

At this stage the Global Map is completely set up. Now all that is left is to create a Submap of points that are useful to us from the global map.

5.2.1. Adding the MINERVA Zone Alarms to the Submap

The appropriate bits from the Global Map must be added to the Submap in the desired order. For us this means that Panel 'A' Zone Alarm 1 should be added first. To do this open the **Submap** page and expand the **Global Map Tree** to be able to see the Zone Alarm 1. It should come under Global Map – TFX500 (Device 'A') – Zones – Zone 1. Select Alarm Zone 1 by clicking on it. The **Add** button should change so it is no longer 'greyed out'. Click the **Add** button, the bit should appear in the Submap on the right-hand side of the screen. Repeat this for all of the Zone Alarms on all three MINERVA panels, making sure that they stay in order in the Submap.

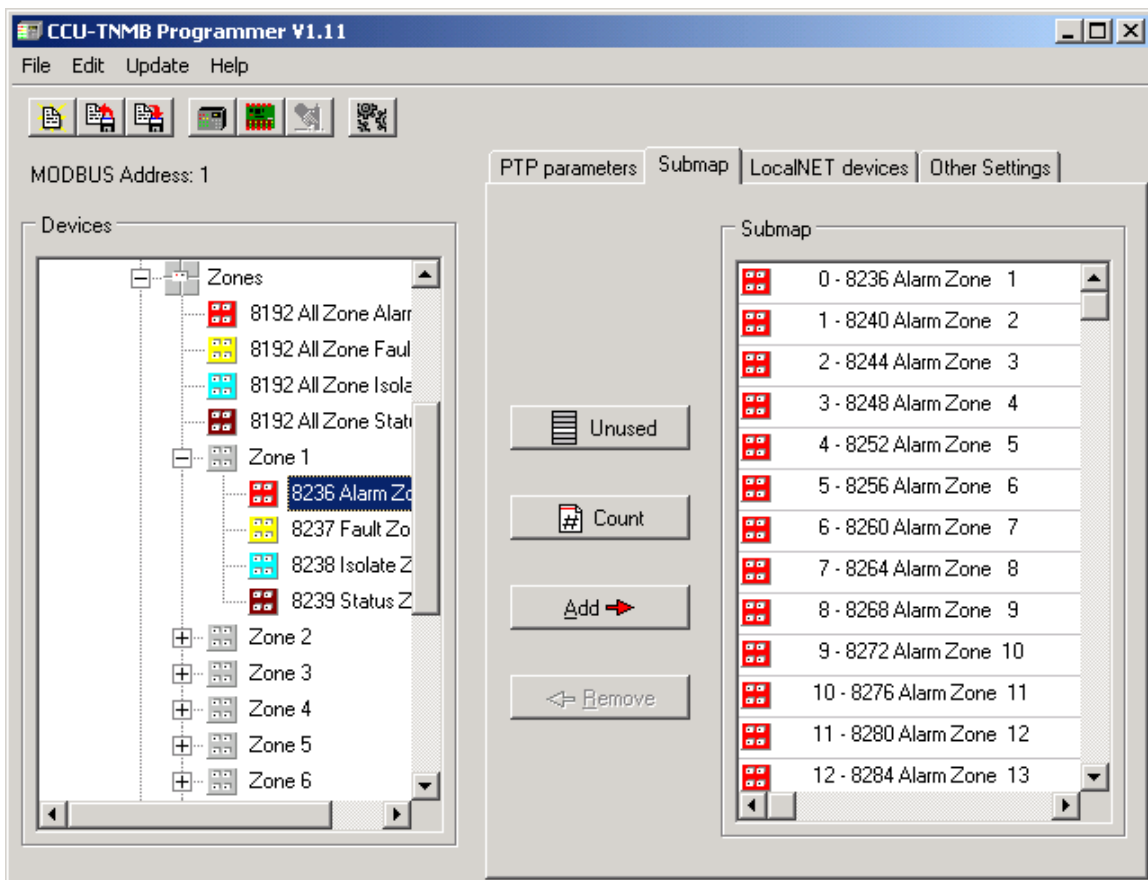


Figure 19: Adding the Zone Alarm bits to the Submap

5.2.2. Adding the Blank addresses to the Submap

In the course of laying out a Submap often there are addresses that are not needed or used and so are left blank. To do this select **Unused Entry** from the **Global Map Tree** and press **Add** to add it to the Submap. As many unused addresses as are required may be added to the Submap. However, unused bits still take up space in the 1000 bit Submap.

Add the desired 16 unused bits to the example project.

5.2.3. Adding the CCU/IO-485 Active bits to the Submap

The CCU/IO-485s have two bits for every input on the board. One indicates a supervision fault on the input and the other indicates that the input has been set to its active state. For the example project the eight Active bits of CCU/IO-485 'D' and 'E' are added to the Submap with eight blank spaces between them.

To add the Active bits to the example project expand the **Global Map Tree** to be able to see the Zone Alarm 1. It should come under Global Map – CCUioNET Nodes – Node 1 (Device 'D') – Node 1 Device 1 – Input Active Node 1 Device. Select Input Active Node 1 Device by clicking on it. The **Add** button should change so it is no longer 'greyed out'. Click the **Add** button, the bit should appear in the Submap on the right-hand side of the screen. Repeat this for all of the Active bits on the CCU/IO-485, making sure that they stay in order in the Submap.

Add eight unused (blank) bits.

Add the eight Active bits for CCU/IO-485 'E'.

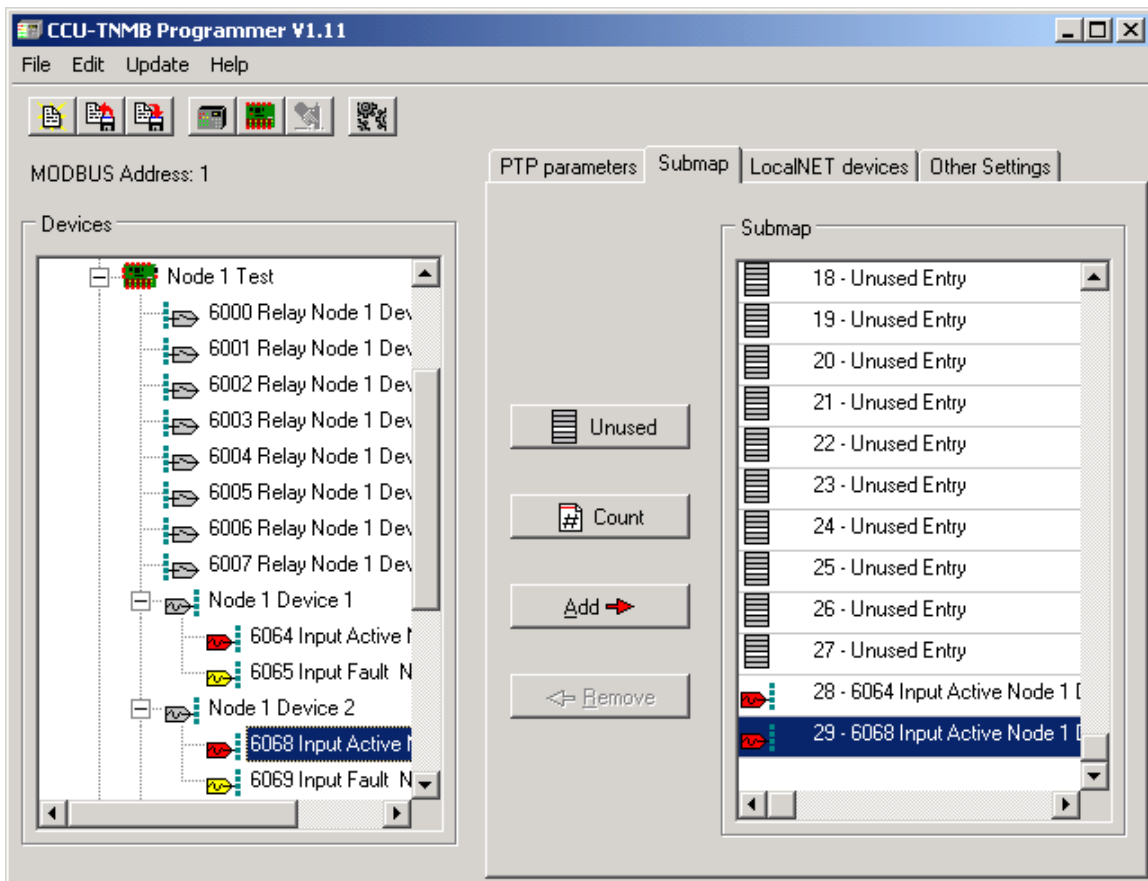


Figure 20: Adding the Input Active bits to the Submap

5.3. Save the project and Program the CCU/485-TNMB

Now both the global map and the Submap are totally set up. Before programming the CCU/485-TNMB it is a good idea to save your work, so that it can be used again at a later date, or for reference purposes.

5.3.1. Saving

To save all of the information that has been entered so far press the **Save** button or select **File | Save** from the Main Menu. To restore it at a later date press the **Load** button or select **File | Load** from the Main Menu.

5.3.2. Programming

Connect the CCU/485-TNMB to the computer using a straight through standard communications port cable (DB9 socket to DB9 plug) from the desired COM port to CHN_C on the CCU/485-TNMB.

Select the **programming** screen and set the **Comport** to the same port that the CCU/485-TNMB is connected to.

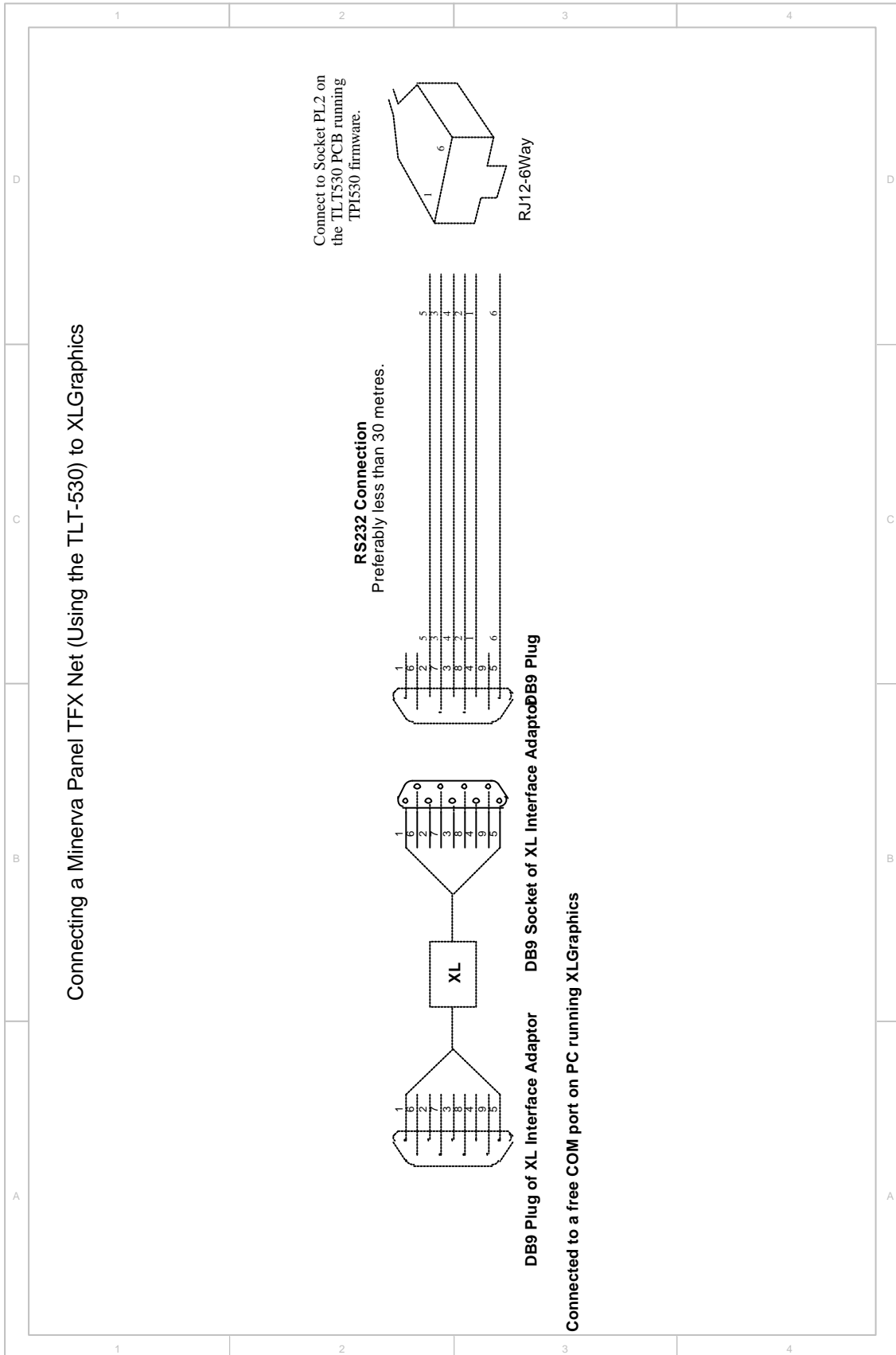
Before programming is commenced the comport must be set in the **CCU-TNMB Setup** program to the port that the CCU/485-TNMB is connected to. To do this select **Update | Select COM Port** from the Main Menu.



Make sure the CCU is powered off

Press the **Update Data** button or select **Update | Data** from the Main Menu, then power on the CCU. Programming should commence.

APPENDIX A: Connecting the CCU/485-TNMB to the MINERVA:



APPENDIX B: Connecting the CCU/485-TNMB to the MODBUS Network:

