PMEUCM DTM

Installation Manual

This manual covers the PMEUCM DTM development and installation procedures.

Effective: August 11, 2017



Niobrara Research & Development Corporation P.O. Box 3418 Joplin, MO 64803 USA

Telephone: (800) 235-6723 or (417) 624-8918 Facsimile: (417) 624-8920 http://www.niobrara.com All trademarks and registered trademarks are the property of their respective owners. Subject to change without notice.

© Niobrara Research & Development Corporation 2017. All Rights Reserved.

Contents

1	Introduction	7
2	DIO Overview	9
	CPU DIO Limitations	10
	DIO Connection Limits	10
	DIO Byte Count Limits	10
	NOC DIO	12
	Hot Standby Implications	12
	Inputs Not Bumpless	12
	Outputs are Bumpless	12
	Remote Rack Mounting Only	12
3	SE PME DTM Library	13
	Installation	14
4	Adding the PMEUCM to Unity Pro	19
	Adding the PMEUCM to Unity Hardware Catalog Manager	20
5	Designing the TXT File	25
	Header Information	26
	Assembly Information	27
	Rules for assemblyID	27
	Rules for assemblyPath	27
	Rules for RPI	28
	Example assemblyPath for single Ethernet/IP half-duplex	29
	Example assemblyPath for single Ethernet/IP full-duplex	30
	Example assemblyPath for dual Ethernet/IP full-duplex	31
	Language Tags Rules	32
	Examples of Language tagnames	32
	Input Data	32
	Rules for Input Data	32
	Input Data Tagnames (Required)	34
	Input Data Tagnames (Optional)	35
	Output Data	36
	Output Data Tagnames (Required)	36
	Output Data Tagnames (Optional)	37
	Configuration Data	37
	Automatic Incremental Variables	37

	VarLoopStart, x, y	37
	VarName	38
	VarLoopEnd	38
6	NRD PTK DTM UTIL	39
	Editing the .TXT file	39
	Start Notepad	39
	Open the file	40
	Open the NRD DTM Tool	42
	Installing a new file	42
	Making Changes to a File	46
	Removing an Entry	49
	Restore after DTM Upgrade	50
	Rebuild .txt from Unity Pro .xys	50
7	QLOAD the Example1 UCM Application	57
	Determine the Installed Application	57
	Halting a running Application	58
	Erasing the Installed Application	58
	Factory Default	59
	QLOAD Example1.qcc	60
8	Unity Pro Operations	63
	New Project	63
	DTM Hardware Catalog Update	69
	Unity Variables	77
	Steps for Modifying the Installed DTM	83
	Link the DTM to the PMEUCM Hardware	83
	Build All	86
	Transfer Project to PLC	87
	PLC Set Address	87
	PLC Connect	88
	Transfer Project to PLC	89
	Transfer to FDR Server	91
	Cycle Power on the PMEUCM	92
9	Using Example1	95
	LED Panel	95
	PTK Board Controlled Lights	95
	UCM OS Controlled Lights	96
	USER Controlled Lights	96
	Example code for blinking lights	97
	LCD and Joystick Operation	98
	Backlight	99
	Menus	99
	Data Echo Config	99
	PLCOUT Data	99
	PLCIN Data	101

Modbus Registers	
Modbus 6x Files	109

1 Introduction

The Niobrara PMEUCM is a user programmable communication card for the Schneider Electric x80 PAC platform. It is capable of running a custom application for performing communication translations between serial and/or Ethernet protocols for the Modicon M580 Automation platform. The exchange of data between the M580 PAC and the PMEUCM across the Ethernet bacplane is controlled by a DTM. This document provides on overview of the UCM DTM development process and installation procedure.

DTM stands for Device Type Manager. DTMs are special files that provide information concerning the configuration and operation of some sort of 'field device'. Typical field devices may be motor drives, electric meters, or the PMEUCM.

DTMs are used with a software package called a FDT (Field Device Tool). Unity Pro includes a built-in FDT. DTMs are installed into the FDT (Unity Pro) and then used to configure the remote device. The DTMs conform to a standard so field devices from different vendors may all be configured within a single FDT.

The M580 PAC relies heavily on DTMs. Many of the Schneider-Electic x80 cards use DTMs for their configuration. All CAPP member x80 cards (like the PMEUCM) use a DTM to tell the M580 which variables are exchanged with the card, static and dynamic configuration, etc.

The PMEUCM is different from other x80 cards because it does not use just a single DTM. Products like the SCAIME PMESWT0100 have a single DTM that uniquely defines all configuration aspects of the card. The PMEUCM is 'user programmable' which means that the data structure that needs to be exchanged with the M580 will be dependent upon the 'user program'.

A customer may have a PMEUCM application that communicates with an inkjet printer. This program may need to exchange data like a string for the label, commands to start/stop printing, feedback about ink levels. A specific DTM for this application will be needed.

A different customer may have a PMEUCM application that communicates with an automotive dynamometer. This application would require variables like wheel speed,

power, start/stop/coast, etc. Obviously, this application needs a different DTM than the printer app.

Developing a custom DTM for each PMEUCM application would be cost prohibitive. Fortunately, Unity Pro includes a 'Generic PTK DTM' which allows a specific type of XML (DDXML) file to be imported into the Unity Pro package and be used as a DTM.

Building one of these DDXML files by hand is tedious, so Niobrara has developed a utility that will generate a proper DDXML file from a simple comma separated text file (.txt) and automatically installs it into the location required by Unity Pro. A simple update of the DTM Hardware Catalog inside Unity Pro is all that is needed to bring a custom PMEUCM DTM to the M580.

2 DIO Overview

All PME devices operate as an Ethernet/IP DIO device to the M580 CPU. This means that the I/O data communication between the M580 CPU and the PMEUCM is done by means of an Ethernet/IP data exchange across the backplane.

When the PMEUCM located in the CPU rack, this data exchange happens completely across the Ethernet backplane between the CPU and the PMEUCM backplane interface.



In the case of the PMEUCM located in an eRIO rack, the exchange happens across the RIO ring from the CPU through the CRA and across the Ethernet backplane to the PMEUCM backplane interface.



Each Ethernet/IP message from the CPU is transported via the RIO ring, through the CRA, and across the Ethernet backplane to the PMEUCM.

The update time of the I/O Data between the CPU and PMEUCM is controlled by the Request Packet Interval (RPI) The RPI is typically 10mS and may be adjusted using the DTM of the PMEUCM.

CPU DIO Limitations

There are several limiting factors concerning DIO use with the M580 PLC. The two most common limits are the number of device connections and the number of bytes exchanged.

DIO Connection Limits

The different M580 CPU models feature limits on the number of Ethernet/IP (EIP) connections, Modbus/TCP (MB) connections and combined EIP+MB connections. Table 2.1 shows the connection limits for M580 CPUs.

DIO Byte Count Limits

Along with limitations on the number of possible DIO devices that may be connected to a single M580 CPU, there are limits on the total amount of bytes of data exchanged.

The actual limiting factor will usually be the total number of PLC Input or Output

bytes used by the DIO. Exceeding the PLC's limit on any of these items will result in a failure to build the proposed system inside Unity Pro.

Table 2.1: CPU DIO Limits

CPU	Max EIP DIO	Max Modbus DIO	Max DIO EIP+MB	Total Bytes DIO INPUT	Total Bytes DIO OUTPUT
BMEP581020	64	64	64	2048	2048
BMEP582020	128	128	128	4096	4096
BMEP582040	64	64	64	2048	2048
BMEP583020	128	128	128	4096	4096
BMEP583040	64	64	64	2048	2048
BMEP584020	128	128	128	8192	8192
BMEP584040	64	64	64	2048	2048
BMEP585040	64	64	64	2048	2048
BMEP586040	64	64	64	2048	2048

For example, the SCAIME PMESWT0100 Weighing module's DTM configures 118 bytes of PLC input data and 32 bytes of PLC output data per card. The maximum number of PMESWT0100 cards that may be installed in a 581020 CPU is limited to 17 cards. This is because having 18 cards would exceed the maximum allowed number of DIO input bytes for the CPU.

18 modules x 118 bytes/module = 2124 bytes

The PMEUCM setup includes several simple example applications with DTM files. EXAMPLE6 includes 1960 bytes of PLC inputs and 1960 bytes of PLC outputs. A 581020 CPU would not be able to include a both the SCAIME PMESWT0100 card and a PMEUCM running EXAMPLE6.

118 (SWT bytes) + 1960 (UCM bytes) = 2078 bytes total

A 582020 CPU would need to be used because it has 4096 bytes available for DIO.

Notice that the byte limitation includes all DIO connections from the CPU.

PME modules + Modbus/TCP slaves + Ethernet/IP slaves

Consider a 584040 CPU that is scanning 30 Holding Registers from each of 15 Modbus/TCP electric meters. This action uses 900 bytes of the 2048 PLC inputs available.

```
(15 slaves) x (30 register/slave) x (2 bytes/register) = 900 bytes total
```

This PLC would then be able to include 9 PMESWT0100 weighing modules.

PMEUCM DTM Manual

2 DIO Overview 11

2048 - 900 = (1148 available bytes) / (118 byes/module) = 9.73 modules

NOC DIO

One option to consider when running into DIO byte count restrictions would be to move some of the DIO to an BMENOC3*1 module. The NOC is capable of controlling its own DIO on its own Ethernet ports. The eNOC Ethernet backplane must be enabled for the DIO to function.

Hot Standby Implications

The PMEUCM may be used in M580 Hot Standby systems (HSBY) remote Ethernet rack.

Inputs Not Bumpless

Like most DIO connections, the PLC Input data from a PME module is not "bumpless" during a transfer from Secondary to the new Primary. The Ethernet/IP connections will be closed and the module "Freshness" will drop to 0 during during the transfer. The data may not be available for several PLC scans after the transfer as the Ethernet/IP connections are re-established.

Outputs Not Bumpless

The PMEUCM will mark the PLC offline when the transfer occurs.

Remote Rack Mounting Only

The PMEUCM may only be used in a remote Ethernet rack in a M580 HSBY system. The PMEUCM may not be mounted in either HSBY CPU rack.

3 SE PME DTM Library

The latest version of Schneider Electric's PME DTM Library must be installed before attempting to use the PMEUCM. The PMEUCM requires many newly added features to the PME DTM.

The latest version is available at Niobrara's web site:

http://www.niobrara.com/programs/PME_DTM_Library_SETUP_1_0_30.EXE



Installation

NOTE: Unity Pro must be closed before installing the PME DTM Library.

NOTE: Unity Pro V11.1 and V12.0 are shipped with an older version of the PME_DTM_Library already installed. The installed version must be updated to allow the PMEUCM0302 to operate properly.

NOTE: Installing a new version of the PME_DTM_LIBRARY will remove all currently installed DDXML files from the repository.

Use the File>Restore feature of the NRDPTKDDXMLUTIL program to recover the previously installed DTMs.

NOTE: This example shows version 1.0.30. The actual file downloaded may not be this version but the procedure is the same.

After downloading the PME_DTM_Library_SETUP_1_0_30.EXE file, run it to begin the installation.

🧬 PME_DTM_Library_SETUP Setup: Installation Fo —	
This will install the Schneider PME DTM Library on your directory	computer. Choose a
Destination Folder	
C: Wiobrara	Browse
Space required: 29.0MB Space available: 688.5GB	
Cancel Nullsoft Install System v2,46	Install

The self extractor will install the setup files into the <u>c:\Niobrara</u>\ folder and automatically start the S-E installation wizard.

📕 🐙 PME DTM Library SET	TUP Setup: Completed — 🗆 🗙
Show details	
Cancel Nullsoft	Install System v2,46 < <u>B</u> ack Close
Schneider Electric PME DI	M Library - 1.0.30
	Walcome to the Installation Wizzud for Eshneider
	Welcome to the Installation Wizard for Schneider Electric PME DTM Library
	Welcome to the Installation Wizard for Schneider Electric PME DTM Library The Installation Wizard will allow you to modify, repair, or
	Welcome to the Installation Wizard for Schneider Electric PME DTM Library The Installation Wizard will allow you to modify, repair, or remove Schneider Electric PME DTM Library. To continue, clic Next.
	Welcome to the Installation Wizard for Schneider Electric PME DTM Library The Installation Wizard will allow you to modify, repair, or remove Schneider Electric PME DTM Library. To continue, did Next.
	Welcome to the Installation Wizard for Schneider Electric PME DTM Library The Installation Wizard will allow you to modify, repair, or remove Schneider Electric PME DTM Library. To continue, did Next.
	Welcome to the Installation Wizard for Schneider Electric PME DTM Library The Installation Wizard will allow you to modify, repair, or remove Schneider Electric PME DTM Library. To continue, dic Next.
	Welcome to the Installation Wizard for Schneider Electric PME DTM Library The Installation Wizard will allow you to modify, repair, or remove Schneider Electric PME DTM Library. To continue, dic Next.
Schneider	Welcome to the Installation Wizard for Schneider Electric PME DTM Library The Installation Wizard will allow you to modify, repair, or remove Schneider Electric PME DTM Library. To continue, dic Next.
Schneider Electric	Welcome to the Installation Wizard for Schneider Electric PME DTM Library The Installation Wizard will allow you to modify, repair, or remove Schneider Electric PME DTM Library. To continue, clic Next.
Scheider Electric	Welcome to the Installation Wizard for Schneider Electric PME DTM Library The Installation Wizard will allow you to modify, repair, or remove Schneider Electric PME DTM Library. To continue, did Next. <

If you have a previous version of the PME DTM Libraray installed, you may be prompted to Repair or Remove the previous installation. Select Repair.

🔀 Schneider Ele	ctric PME DTM Library - 1.0.30	×
Program Main Modify, repair,	tenance , or remove the program.	Schneider GElectric
	Repair installation errors in the program. This option fixes corrupt files, shortcuts, and registry entries.	missing or
○ <u>R</u> emove	Remove Schneider Electric PME DTM Library from your co	mputer.
	< <u>B</u> ack <u>N</u> ext >	Cancel

When finished a screen like the following should be displayed.



After selecting "Finish", the S-E Wizard will close. Now close the PME_TCM_Libraray_Setup window to complete the setup.

PME_DTM_Library_SETUP Setup: Completed	—	\times
Completed		
Show <u>d</u> etails		
Cancel Nullsoft Install System v2.46	< <u>B</u> ack	se

If you have not installed the PMEUCM_SETUP.EXE program and added the PMEUCM0302 to the Unity Hardware Catalog, proceed immediately to the next chapter before starting Unity Pro.

PMEUCM_SETUP.EXE

The latest version of Niobrara's PMEUCM_SETUP must be installed before attempting to use the PMEUCM. This setup installs many utilities needed to configure the PMEUCM. The user may access this file at:

http://www.niobrara.com/html/pmeucm_cut.html



Download and run PMEUCM_SETUP.EXE. A box will appear prompting the user to choose a directory in which to install. The default is <u>C:\Niobrara</u>, as shown below.

PMEUCM_SETUP Setup: Installation Folder	—		×
This will install important PMEUCM-related files on directory	i your con	nputer. Cho	oose a
Destination Folder			
C: Wiobrara		Browse	
Space required: 9.4MB			
Space available: 688.5GB			
Cancel Nullsoft Install System v3,01		Insta	

4 Adding the PMEUCM to Unity Pro

Unity Pro versions 8.1 and higher provide a method for adding third party modules to their hardware catalog. Niobrara provides the necessary .cpx file as part of PMEUCM_SETUP.EXE. The user may access this file at:

http://www.niobrara.com/html/pmeucm_cut.html



Adding the PMEUCM to Unity Hardware Catalog Manager

NOTE: Unity Pro must be not be running to access the Hardware Catalog Manager.

After the setup program is finished, start the Hardware Catalog Manager, located at Start>All Programs>Schneider Electric. In the File menu, click on Import User Devices, as shown below.

Hardware Catalog Manager	- 🗆 X
<u>File Edit View Service H</u> elp	
Import User Devices Ctrl+I]
Export User Devices Ctrl+E	Build Catalog
Exit	Abort Modifications
Safety	
Third party products	Close
\Build \A Import/Export \ADDA Log /	
Add one or several devices from archive file	

Choose the folder where PMEUCM_SETUP.EXE installed the .cpx file. This is normally the 'c:\Niobrara\PMEUCM\DTM\' folder.

Inside the DTM folder is a file for Unity V11, V11.1, and V12: M580_PME_UCM_0302_for_V11_and_V12.cpx Also present is a cpx file for Unity V13: M580_PME_UCM_0302_for_V13.cpx

Choose the appropriate file for the installed version of Unity Pro, then click 'Open'.



Import User Dev	vices		×
User devices			
PME_UCM			
Select All			
	OK	Cancel	

The Hardware Catalog Manager will show a dialog box displaying its progress.

Import User Devices	
er devices	
ME_UCM	
Select All	
OK Cancel	
Hardware Catalog Manager	- L X
File Edit View Service Help	
⊡ CANopen drop	Build Catalog
Distributed I/Os	
	All and Man BC and Server
	Abort Modifications
Inird party products	Close
Building	
Please wait when building the catalog	
Catalog building started at 14:12:12.	
Initializing database for update	
Process database Finalize database	

When it is finished, it will appear as below.

For Help, press F1



Close the Hardware Catalog Manager, and start Unity Pro. The PMEUCM0302 can now be chosen from the Hardware Catalog under the "Third party products" section.

🔁 PLC	Dus			
Bus:		0 BME P58 1020 02.10	~	
0		CPS 2000 New Device		×
		Topological Address:	0.2	OK Cancel
	Ц	Part Number	Description	Help
l L		Modicon M580 local drop		
		🕀 🛄 Analog		
		Communication		
		Counting		
		Discrete		
		Motion		
J		Third party products		
		PME SWT 0100	Partner Module Ethernet System Weighing Transmitter 1 channel	
		PME UCM 0202	Partner Module Ethemet UCM	
		PME UCM 0302	User Programmable Module for customer Serial and Ethemet networks	

5 Designing the TXT File

A comma separated variable text file (TXT) is used by the NRDPTKDTMUTIL.EXE program to generate the custom DDXML file used by Unity Pro's Generic PTK DTM. This txt file is simple to configure using a standard text editor like MS Notepad.

The following rules are enforced by the compiler utility:

- The filename of the txt file must have a single underscore "_".
 - The filename should be of the format "PME UCM 0202_name.txt" where "name" is the project name.
 - More than one underscore is not allowed.
 - Decimal points and commas are not allowed.
 - Spaces are allowed.
- The format of each line is tagname, tagdata followed by a new line.
- Only one tagname is allowed per line.
- Leading spaces and Tab characters before the tagname are ignored.
- Tab characters between the comma and the tagdata are ignored.
- Leading spaces between the comma and the tagdata are ignored.
- Trailing spaces between the tagdata and new line are ignored.
- Empty lines are ignored.
- Some tagnames include modifiers that are separated by commas.
- Commas are not allowed inside tagdata. Use a semi-colon instead.
- Tagnames are not case sensitive.
- Tagdata may be case sensitive.
- assemblyID and assemblyPath tagnames are required as a pair. These tagnames must be adjacent to each other.
- There must be at least two assemblyID and assemblyPath pairs.
- The varType 'STRING' must be preceded by the varLength tagname.

- Language fields must be presented in the same order throughout the document. For example, if 'En-En' is listed first, followed by 'Fr-Fr' then this order must be followed whenever both EN and FR are referenced by a description.
- If the very first character of a line is #, the line is ignored. This may be used to add comments.
- The 'Header' SW is the only required header value.

Header Information

The first block of information includes data about the application.

Tagname	Description	Default Value if Ommited	
fileCreator	Name of the person or company building the file.	NRDDTMUTIL	
fileModifiedBy	Name of person or company modifying the file.	NRDDTMUTIL	
fileCreationDate	Hardcoded date or 'now' to use the PC's time/date.	'now'	
fileModifiedDate	Hardcoded date or 'now' to use the PC's time/date.	'now'	
MinCompiler	Datecode of required DTM Utility	Current version of the DTM Utility	
productID	Must start with 'PME UCM 0202' followed by a unique identifier for this DTM.	'PME UCM 0202 name' where 'name' is the text following the underscore in the source file name.	
productText	Usually the text following the 0202 in the productID	'name' i.e. the text following the underscore in the source file name.	
SW	Number of the software version. This is the value shown in the Unity Pro DTM Browser as the 'version'.	Required. Should be of the form xx.xx normally 01.00	
FW Set to match the Firmware Version of the UCM board. (Ignored at this time)		01.00	
HW	Set to match the Hardware Version of the UCM board. (Ignored at this time)	01.00	

Assembly Information

The data is transported between the M580 CPU and the PMEUCM using an Ethernet/IP explicit data connection. Each of these connections is limited to a total of around 1400 bytes each in and out. This 1400 byte limit includes the header information tacked on by the DTM. Sometimes it is desirable to have more than one Ethernet/IP data connection between the M580 and the PMEUCM. The assemblyID and assemblyPath tagnames are used to define these connections. The update timing of these assemblies may be set using the assemblyDefaultRPI tagname.

Rules for assemblyID

- The first group of InputVars must be assemblyID = 1.
- The first group of OutputVars must be assembly ID = 2.
- assemblyIDs must be sequential with InputVars EVEN numbers while OutputVars must be ODD numbers
 - InputVars follow the sequence 1, 3, 5, 7
 - OutputVars follow the sequence 2, 4, 6, 8
 - The maximum assemblyID is 8.
- assemblyID settings are valid in a top-down fashion. Variables defined after an assemblyID declaration are included in that assembly until the next assemblyID statement is reached.

Rules for assemblyPath

- The first InputVars assemblyPath must be 101.
- The first OutputVars assemblyPath must be 101.
- assemblyPath settings are valid in a top-down fashion. Variables defined after an assemblyPath declaration are included in that assembly until the next assemblyPath statement is reached.

Rules for RPI

- The RPI for the assembly group may be optionally set using assemblyDefaultRPI.
- The minimum supported RPI is 4mS.
- The maximum supported RPI is 500mS.
- The tagname assemblyMinRPI can be used to limit the lowest settable RPI on a connection. This may be useful for restricting the bandwidth used by a secondary connection.
- The tagname assemblyMaxRPI may be used to limit the highest settable RPI for a connection. This may be useful for restricting the update time of the Primary Connection.
- RPI settings apply to both input and output channels. It is only necessary to set RPI commands on the INPUT assemblies. The DTM Utility will automatically apply RPI settings to the OUTPUT assembly.
- The following default settings are used if the RPI tagnames are omitted:
 - Assembly 1,2
 - Default = 10 mS
 - Min = 4 mS
 - Max = 500mS
 - Assembly 3,4
 - Default = 50mS
 - Min = 4mS
 - Max = 500mS
 - Assembly 5,6
 - Default = 100mS
 - Min = 20mS
 - Max = 500mS
 - Assembly 7,8
 - Default = 300mS
 - Min = 40mS
 - Max = 500mS

Example assemblyPath for single Ethernet/IP half-duplex

SW,	1.00
assemblyID,	1
assemblyPath,	101
varHeading,	InputVars
varType,	WORD
varName,	UCM_Runtime_Status
varLabel,en-us,	.15=Run; LSB=Runtime Error
V	
varHeading,	InputVars
varType,	UINT
varName,	UCM_Halt_Line_Number
varLabel,en-us,	UCM Runtime Error Halt Line Number
V	
varHeading,	InputVars
varType,	INT
varName,	In_01
varLabel,en-us,	In_01
assemblyID,	2
assemblyPath,	102
varHeading,	outputVars
varType,	REAL
varName,	out_01
varLabel,en-us,	out_01
varHeading,	outputVars
varType,	DINT
varName,	out_02
varLabel,en-us,	out_02
varHeading,	outputVars
varType,	UINT
varName,	out_03
varLabel,en-us,	out_03
varHeading,	outputVars
varType,	UINT
varName,	out 04
varLabel,en-us,	out 04
	_

Example assemblyPath for single Ethernet/IP full-duplex

SW,	1.00
assemblyID,	1
assemblyPath,	101
varHeading,	InputVars
varType,	WORD
varName,	UCM_Runtime_Status
varLabel,en-us,	.15=Run; LSB=Runtime Error
varHeading,	InputVars
varType,	UINT
varName,	UCM_Halt_Line_Number
varLabel,en-us,	UCM Runtime Error Halt Line Number
varHeading,	InputVars
varType,	INT
varName,	In_01
varLabel,en-us,	In_01
assemblyID,	2
assemblyPath,	101
varHeading,	outputVars
varType,	REAL
varName,	out_01
varLabel,en-us,	out_01
varHeading,	outputVars
varType,	DINT
varName,	out_02
varLabel,en-us,	out_02
varHeading,	outputVars
varType,	UINT
varName,	out_03
varLabel,en-us,	out_03
varHeading,	outputVars
varType,	UINT
varName,	out_04
varLabel,en-us,	out_04

Notice that the both the first and second assemblyPaths are set to 101.

Example assemblyPath for dual Ethernet/IP full-duplex

SW,	1.00
assemblyID, assemblyPath, varHeading, varType, varTyme, varName, varLabel,en-us,	1 101 InputVars WORD UCM_Runtime_Status .15=Run; LSB=Runtime Error
varHeading,	InputVars
varType,	UINT
varName,	UCM_Halt_Line_Number
varLabel,en-us,	UCM Runtime Error Halt Line Number
varHeading,	InputVars
varType,	INT
varName,	In_01
varLabel,en-us,	In_01
assemblyID,	3
assemblyPath,	102
varHeading,	InputVars
varType,	INT
varName,	In_02
varLabel,en-us,	In_02
assemblyID,	2
assemblyPath,	101
varHeading,	outputVars
varType,	REAL
varName,	out_01
varLabel,en-us,	out_01
varHeading,	outputVars
varType,	DINT
varName,	out_02
varLabel,en-us,	out_02
assemblyID,	4
assemblyPath,	102
varHeading,	outputVars
varType,	UINT
varTyme,	out_03
varLabel,en-us,	out_03
varHeading,	outputVars
varType,	UINT
varName,	out_04
varLabel,en-us,	out_04

Language Tags Rules

- Tagnames that require language tags must include a single language tag.
- Tagnames requiring language tags must be inserted in the same order throughout the txt file.
- Language tags are not case sensitive.
- Language tags follow the tagname with a comma.
- The following Language Tags are valid:

Tag	Language	
En or en-us	English (United States)	
Fr or fr-fr	French (standard)	
De or de-de	German (standard)	
It or it-it	Italian (standard)	
pt-br	Portuguese (Brazil)	
Es or es-es	Spanish (standard)	

Examples of Language tagnames

varLabel,en-us,	Gross Measurement
varLabel,fr-fr,	Mesure en Brut

Input Data

Rules for Input Data

- VarHeading must be the first tagname
- varType STRING must be preceded by varLength
- varTypes REAL, DINT, UDINT, and DWORD must be aligned on a 4byte boundary starting from the first byte assigned.
- VarTypes INT, UINT, and WORD must be aligned on a 2-byte boundary starting from the first byte assigned.
- A maximum of 428 bytes of Input data may be defined in the Primary (first) Ethernet/IP assembly. The 72 byte PTK header is always included in the first input assembly.
- A maximum of 500 bytes of Input data may be define for assemblies 2, 4, 6, and 8.
- The first variable must be UCM_Runtime_Status. The UCM operating system always places this WORD variable into the PLC INPUT data

structure.

• The second variable must be UCM_Halt_Line_Number. Again, the OS places this UINT value into the PLC INPUT data structure.

Input Data Tagnames (Required)

Value		Notes
InputVars		Must be the first field for a variable.
Length of follows	string that	Required for varType=STRING
Туре	Byte Length	
BOOL	1	
BYTE	1	
WORD	2	
DWORD	4	
REAL	4	
STRING	variable	
INT	2	
DINT	4	
UINT	2	
UDINT	4	
Text Field		
Text Field		Requires Language tag
Text Field		Requires Language tag Defaults to varLabel if ommited.
Туре		Defaults to 'Read' for Input, 'Read/Write' for Output if ommited.
Read		
Write		
ReadWrite		
ConditionalReadWrite		
noAccess		
Туре		Defaults to 'True' for Input and 'False for Output if ommited.
True		
False		
	VaInputVarsLength of followsTypeBOOLBYTEWORDDWORDREALSTRINGINTUINTUDINTText FieldText FieldText FieldText FieldConditionalnoAcccessTrueFalse	ValueInputVarsLength of string that followsTypeByte LengthBOOL1BYTE1WORD2DWORD4REAL4STRINGvariableINT2DINT4UINT2UDINT4Text FieldText FieldText FieldImpleReadViriteReadWriteConditionalReadWritenoAcccessTrueFalseImple

Input Data Tagnames (O	ptional)
------------------------	----------

Tagname	Value	Notes
varDefault	Signed Decimal number	Sets the default value for the variable.
varLowLimit	Signed Decimal number	Sets the lowest allowed value.
varHiLimit	Signed Decimal number	Sets the highest allowed value.
varWarning	Text Field	Requires Language tag
varDescription	Text Field	Requires Language tag
varUnit	Text Field	

Output Data

Output Data Tagnames (Required)

Tagname	me Value		Notes	
varHeading	InputVars		Must be the first field for a variable.	
varLength	rLength Length of string that follows		Required for varType=STRING	
varType	Туре	Byte Length		
	BOOL	1		
	BYTE	1		
	WORD	2		
	DWORD	4		
	REAL	4		
	STRING	variable		
	INT	2		
	DINT	4		
	UINT	2		
	UDINT	4		
varName	Text Field			
varLabel	Text Field		Requires Language tag	
varDescriptio n	ptio Text Field		Requires Language tag Defaults to varLabel if ommited.	
varAccess	Туре		Defaults to 'Read' for Input, 'Read/Write' for Output if ommited	
	Read			
	Write			
	ReadWrite			
	ConditionalReadWrite			
	noAcccess			
varPersistent	Ту	pe	Defaults to 'True' for Input and	
	True		'False for Output if ommited.	
	False			
Tagname	Value	Notes		
----------------	-----------------------	--		
varDefault	Signed Decimal number	Sets the default value for the variable.		
varLowLimit	Signed Decimal number	Sets the lowest allowed value.		
varHiLimit	Signed Decimal number	Sets the highest allowed value.		
varWarning	Text Field	Requires Language tag		
varDescription	Text Field	Requires Language tag		
varUnit	Text Field			

Output Data Tagnames (Optional)

Configuration Data

Variable Arrays

Certain variable types may be defined as arrays. To declare a variable as an array, simply add a ',' followed by the starting index '0', then another ',' followed by the ending index.

NOTE: The current version of the DTM only supports a starting index of 0.

Array Variable Type	Example	Result
BYTE	VarType, BYTE,0,99	Array of BYTE[099]
WORD	VarType, WORD,0,9	Array of WORD[09]
DWORD	VarType, DWORD,0,4	Array of DWORD[04]
INT	VarType, INT,0,9	Array of INT[09]
DINT	VarType, DINT,0,9	Array of DINT[09]
UINT	VarType, UINT,0,3	Array of UINT[03]
UDINT	VarType, UDINT,0,9	Array of UDINT[09]
REAL	VarType, REAL,0,10	Array of REAL[010]

Automatic Incremental Variables

It can become tedious to define a number of variables that share the same parameters and only differ by variable name. The DTM Utility supports automatic incremental naming of a variable where the name includes a three character number that varies from a starting point to a finish point.

```
# The following loop creates PLC_OUT_B000 through PLC_OUT_B009
# 0=Start, 9=End
varLoopStart, 0, 9
varHeading, outputVars
varType, BYTE
varName, PLC_OUT_B%
varLabel,en-us,Process data from PLC
varLoopEnd
```

VarLoopStart, x, y

VarLoopStart marks the beginning of an automatic variable generation. It includes two parameters. The first parameter is the starting value and the second is the ending value. The first value must be smaller than the second value. The maximum number of variables generated is 500.

VarName

The varName must include a '%' character at the end of the string. This % character is replaced by the 3 digit number generated by the varLoop. The number is always 3 digits and includes leading zeros.

VarLoopEnd

VarLoopEnd denotes the end of a VarLoopStart segment. The trailing comma and any parameter are ignored and may be ommitted.

6 NRD PTK DTM UTIL

Editing the .TXT file

Any text editor (like MS Notepad) may be used to edit the .TXT file used to generate the DDXML for the PMEUCM.

For this exmple, the file c:\Niobrara\DTM\PME UCM 0202_Example1.txt is used.

Start Notepad

The first step is to open Notepad. It is usually found under Start > Programs > Accessories > Notepad.



Next, it is a good idea to turn on the "Status Bar". This setting is found under "View, Status Bar". The Status Bar shows the line number and column location of the cursor. The line number is helpful if there is an error compiling the .txt file.

2	Untitled - Notepad -	×
<u>File Edit Fo</u> rmat <u>V</u> iew	Help	
 Status Bar 		^
<		>
	Ln 1, Col 1	
Open the file		

Open the file

Next select File, Open and then browse to c:\Niobrara\DTM\ and open the file: "PME UCM 0202_Example1.txt"

1	Open		×
🔄 🏵 🗉 🕇 🚺 « Nio	brara → PMEUCM → DTM	✓ 🖒 Search DTM	م
Organize 🔻 New folder			
Desktop ^	Name	Date modified Type	e
Documents	PME UCM 0202_EXAMPLE1.txt	11/19/2015 3:40 PM TXT	File
Uownloads	PME UCM 0202_EXAMPLE2.txt	11/18/2015 12:44 TXT	File
Music	PME UCM 0202_EXAMPLE3.txt	11/18/2015 12:43 TXT	File
Videos	PME UCM 0202_EXAMPLE4.txt	11/23/2015 1:03 PM TXT	File
🊢 TI10685100A (C:)			
🖵 common (\\fsrv1			
→ My Book (H:)	C		
File <u>n</u> ar	me: PME UCM 0202_EXAMPLE1.txt	✓ Text Documents (*.txt)	~
	Encoding: ANSI	Open C	Cancel

	PME UCM 0202_EXAMPLE1.txt - Notepad	- 🗆 🗙
File Edit Format View Help		
SW,	01.00	^
	4	
assemblyID,	101	
assemblyDefaultRPT.	15	
assemblyMinRPI,	5	
assemblyMaxRPI,	200	
varHeading,	InputVars	
varlype,	WORD UCM Ruptimo Status	
varlabel.en-us.	15=Run: SB=Runtime Error	
var Eaberjen abj	.is han, is hancine in or	
varHeading,	InputVars	
varType,	UINT	
varName,	UCM_Halt_Line_Number	
varLabel,en-us,	UCM Runtime Error Halt Line Number	
vanHeading	TaputVans	
varType.	TNT	
varName,	In 01	
varLabeĺ,en-us,	In_01	
varHeading,	InputVars	
varlype,	1NI T- 02	
varName,	In_02 In_02	
var Laber, en-us,	11-02	
varHeading,	InputVars	
varType,	INT	
varName,	In_03	
varLabel,en-us,	In_03	
varHeading.	InputVars	
varType,	INT	
varName,	In_04	
varLabel,en-us,	In_04	
assemblyTD	2	
assemblyPath.	102	
varHeading,	outputVars	
varType,	INT	
varName,	out_01	~
<		>
		Ln 1, Col 1

Open the NRD DTM Tool

The next step is to open the Niobrara DTM Tool. Select Programs > Niobrara > PMEUCM > DTM > DTM Utility.



The tree on the left of the screen shows the PTK DTMs installed in Unity Pro. In this case, it shows the PME SWT 0100 Weighing Module from SCAIME.

Installing a new file

Select File > "Install new .txt..." and then browse to the c:NiobaraPMEUCMDTM folder and select the file to install "PME UCM 0202_Example1.txt"

		NRD PTK DTM Too	I Rev. 24NOV2015 – 🗖
	File	Help	
Install new .bt Remove from Repository Restore after DTM upgrade Rebuild .bt from Unity Pro .xys Exit		▶ PME SWT 0100	NRDDDXMLUTIL.EXE Rev. 24NOV2015 Niobrara R&D Corp. Joplin, MO USA Technical Support at +1 417-624-8918 www.niobrara.com This program is used to convert comma separated formate into the DDXML file used by Schneider-Electric Unity Pro for with the PMEUCM0202 module. The DTM tree on the left is taken from the PME DDXML_Re It is necessary to 'Update' the DTM section of the Hardward

NR&D DDXML Utility							
🔄 🌛 🔹 🕇 📕 «	Niobrara → PMEUCM → DTM v C	Search DTM	٩				
Organize 🔻 New 1	older	!≡ ▼ □	0				
Desktop	A Name	Date modified Type					
Uownloads	PME UCM 0202_EXAMPLE1.txt 1	11/19/2015 3:40 PM TXT File					
Recent places	PME UCM 0202_EXAMPLE2.txt	11/18/2015 12:44 TXT File					
The DC	PME UCM 0202_EXAMPLE3.txt 1	11/18/2015 12:43 TXT File					
Parlitan	PME UCM 0202_EXAMPLE4.txt	11/23/2015 1:03 PM TXT File					
Desktop							
Documents							
Downloads	v <		>				
	0202 EXAMPLE1.txt Date modified: 11/19/2015 3:40 P	M					
TXT File	Size: 1.29 KB						
Fi	e <u>n</u> ame: PME UCM 0202_EXAMPLE1.txt v	Text Files, (*.TXT)	*				
		<u>O</u> pen ▼ Cancel					

After selecting "Open", the main screen should now change to show a new entry in the tree.

NRD PTK D	TM Tool Rev. 24NOV2015 – 🗆 🗙
<u>File</u> <u>H</u> elp	
PME SWT 0100	Building: C:\Niobrara\PMEUCM\DTM\PME UCM 0202_EXAMPLE Success! INPUT Byte Count = 12 [84] OUTPUT Byte Count = 14 [40] Inserting: C:\Niobrara\PMEUCM\DTM\PME UCM 0202_EXAMPL No Match, Added. Remember to Update the Unity Pro Hardware DTM Catalog.
	*

Status information is displayed on the right side of the screen. If there is an error during the compile, the error description and source code line number will be displayed.

PMEUCM DTM Manual

NRD PTK DTM To	ol Rev. 24NOV2015 – 🗆 🗙
<u>F</u> ile <u>H</u> elp	
PME SWT 0100 PME UCM 0202_EXAMPLE1 PME v0 1.00 Input Byte Count = 12 (84) UCM_Runnine_Status, WORD, 1000, 0 UCM_UDI, INT, 1003, 6 In_03, INT, 1004, 8 UCM_UDI, INT, 1005, 10 UCM_UDI, INT, 1005, 0 UCM_UDI, INT, 1007, 2 UCM_UDI, INT, 1007, 2 UCM_UDI, INT, 1008, 4 UCM_UDI, INT, 1010, 8 UCM_UDI, INT, 1011, 10 UCM_UDI, INT, 1011, 10 UCM_UDI, INT, 1012, 12	NRDDDXMLUTILEXE Rev. 24NOV2015 Niobrara R&D Corp. Joplin, MO USA Technical Support at +1 417-624-8918 www.niobrara.com This program is used to convert comma separated formated files into the DDXML file used by Schneider-Electric Unity Pro for use with the PMEUCM0202 module. The DTM tree on the left is taken from the PME DDXML_Repository in Unit It is necessary to 'Update' the DTM section of the Hardware Catalog in Un
J	

Expanding the tree for the bptest3in4out entry shows quite a few details.

The SW Version number is shown:



This is the value from the txt file:



Next is the structure of the PLC INPUT data.



The heading shows the total number of bytes of PLC INPUT data. In this case there are 12 bytes of data. The number following (84) gives the total number of byes including the PTK header information.

The next segment is the assemblyID information. The assemblyID = 1 and Path = 101 are from the .txt file. The Instance = x271B, File = 58100, and Byte Count = 12 values are useful in configuring the UCM application. AssemblyIDs 1 and 2 also include the total byte count with the header information (+72 bytes for inputs and +26 bytes for outputs).

The variables start with the UCM_Runtime_Status. This is a WORD variable. The number 1000 that follows is the Ethernet/IP reference number assigned to this variable. These numbers always start at 1000 and are automatically generated by the DTM Utility. The value 0 that follows the 1000 is the byte offset from the beginning of the structure. The offset value is useful when doing the UCM programming.

The PLC OUTPUT data structure follows.



Again there is the byte count of outputs 14, followed by the total byte count including the PTK overhead of 26 bytes (40). The assemblyID = 2 and Path = 102 are from the .txt file. The Ethenet/IP index 1006 is next followed by the byte offset from the start of the data structure.

Making Changes to a File

Making a change to the installed DTM is a simple as modifying the txt file, saving it, and then performing the Import function again.

							PME UCM 0202_EXAMPLE1.txt - Notepad		- 🗆 🗙
		File	Edit	Format	View	Help			
New	Ctrl	+N				1	01.01		^
Open	Ctrl	+0							
Save	Ctrl	+ S	embly	yID, vDath			1		
Save As			embly	vDefaul	+RPT		15		
Page Setup			embly	MinRPI	[,	•	5		
Print	Ctrl	+ P	embly	yMaxRP1	Ι,		200		
			Unadi				TanutVana		
Exit	_	_	neau.	varTvr	ne.		WORD		
				varNar	ne,		UCM_Runtime_Status		
				varLat	pel,er	n-us,	.15=Run; LSB=Runtime Error		
							T		
		Var	Head:	varTvr	10		INDUTVARS		
				varNar	ne.		UCM Halt Line Number		
				varLat	peĺ,er	n-us,	UCM Runtime Error Halt Line Number		
		var	Head	ing,		1			
				varNar	ne.		In 01		
				varLat	bel,er	1-us,	In_01		
		var	Head	ing,			InputVars		
				variyp	, no				
				varLat	pel,er	n-us,	In_02		
							-		
		var	Head	ing,			InputVars		
				variyp	e,		INI To 03		
				varLat	pel,er	ı-us,	In 03		
						,	-		
							T		
		var	Head	ing,			InputVars		
				varNar	ne.		INI In 04		
				varLat	bel,er	n-us,	In_04		
					-				
							2		
		ass	emp1)	vIU, vPath			∠ 102		
				,,					
		var	Head	ing,			outputVars		
				varTyp	ре,		INT		
				varNar	ne,		out_01		~
		<							>
								Ln 20, Col 16	

In this example, the SW version is changed from 1.00 to 1.01. Also, the varType of In_01 is changed from an UINT to a UDINT. The new version is saved with the same filename.

Now, back in the NRD DTM Utility, do a File > "Install new .txt..." and select the same filename.

	File Help
Install new .txt	PME SWT 0100
Remove from Repository	PME UCM 0202_EXAMPLE1
Restore after DTM upgrade	
Rebuild .bxt from Unity Pro .xys	
Exit	

NR&D DDXML Utility									
🔄 🏵 🔻 🕇 퉬 « Niol	() ✓ ↑ ↓ ≪ Niobrara → PMEUCM → DTM ✓ C Search DTM								
Organize 🔻 New folder	Organize 👻 New folder								
Desktop ^	Name	Date modified	Туре						
Documents	PME UCM 0202_EXAMPLE1.txt	11/23/2015 11:30	TXT File						
Downloads	PME UCM 0202_EXAMPLE2.txt	11/18/2015 12:44	TXT File						
Music	PME UCM 0202_EXAMPLE3.txt	11/18/2015 12:43	TXT File						
Videos	PME UCM 0202_EXAMPLE4.txt	11/23/2015 1:03 PM	TXT File						
LI10685100A (C:)									
🖵 common (\\fsrv1									
My Dook (H)				>					
PME UCM 020	2_EXAMPLE1.txt Date modified: 11/23/2015 11: Size: 1.29 KB	30 PM							
File <u>n</u> ar	ne: PME UCM 0202_EXAMPLE1.txt V	Text Files, (*.TXT) <u>O</u> pen ▼	Cancel	 ✓ : 					

Note: It is not necessary to remove the installed DTM to make modifications. Simply Install the same file again.

Note: The INTPUT Byte Count has changed from 12 to 14 because the INT was changed into a DINT.

PMEUCM DTM Manual

NRD PTK DTM Tool Rev. 24NOV2015 - 🗆 🗙			
<u>F</u> ile <u>H</u> elp			
PME SWT 0100 → PME UCM 0202_EXAMPLE1 → Ver. 01.01 → ↓ Input Byte Count = 14 (86) → ↓ Output Byte Count = 14 (40)	NRDDDXMLUTIL.EXE Rev. 24N0V2015 Niobrara R&D Corp. Joplin, MO USA Technical Support at +1 417-624-8918 www.niobrara.com This program is used to convert comma separated f into the DDXML file used by Schneider-Electric Unity with the PMEUCM0202 module.		

Expanding the tree shows the new structure for the PLC INPUT data:



Notice that the bytes offset of In_02 is now 8 because In_01 is a UDINT (4 bytes).

Removing an Entry

Highlight one of the DTM names in the tree and select File > "Remove from Repository".



A prompt will appear asking for conformation of the removal.

Confirm Remove ×
'OK' to Remove PME UCM 0202_EXAMPLE1.XML
OK Cancel

Select "OK" to remove the DTM entry.

Select "Cancel" to keep the DTM entry.

	NRD PTK DTM Tool Rev. 24NOV2015 – 🗆 🗙		
<u>F</u> ile <u>H</u> elp			
E		Removing: PME UCM 0202_EXAMPLE1 Match, Removed. Remember to Update the Unity Pro Har	XML
		<	>

PMEUCM DTM Manual

6 NRD PTK DTM UTIL 51

Restore after DTM Upgrade...

The File > "Restore after DTM Upgrade" menu item is used to recover installed DDXML files from the repository after an update to the DTM has been performed.

Rebuild .txt from Unity Pro .xys...

The File > "Rebuild .txt from Unity Pro .xys..." menu item is used to recover a .txt source file from an active Unity Pro project. This may be necessary if the original .txt file is lost and the only information available about the application variable structure is the Unity Pro project.

NOTE: The file rebuilding function makes the following assumptions:

- All PLC_INPUTS are placed in assemblyID = 1 with PATH = 101
- All PLC_OUTPUTS are placed in assemblyID = 2 with PATH = 102
- RPI is set to 10mS

NOTE: It is necessary to inspect the PMEUCM source file to determine if more assemblyID definitions or PATH adjustements are required.

The first step is to open the Project Browser in the Unity Pro project and right click on the Variables & FB Instances to select "OPEN".

roject Browser				
E Structural view				
Example 1 Configuration Configurat				
	Open Export Import Add User Directory Add Hyperlink			
Communication	Zoom in Zoom out Expand all Collapse all			

The Data Editor will open. Narrow down the displayed items to only include "Device DDT".

		Data Edi	itor					- 0	x
Variables	DDT Types Function Blocks	DFB Types							
Filter	🍸 🍇 Name =			EDT		01	DDT 🗹	Device DD	
Name		• .	Туре			• ,	Address	-	Valu
• • • • • • • • • • • • • • • • • • •	BMEP58_ECPU		T_BMEP58	_ECPU					
B	PME_UCM_0202_EXAMPLE1		T_PME_UC	M_0202_E	XAMPLE	1			
									_
									-

Now right click on the PME_UCM_0202_EXAMPLE1 structure and select "Export Selected".

	Data Editor			
Variables DDT Types Function Blocks DFB Types				
Filter 🚺 🐝 Name 🗄	•		DDT 🔽 Device DDT	
Name	🔻 🔤 Туре	▼ ,	Address 👻 Valu	
BMEP58_ECPU	T_BMEP58	_ECPU		
	Cut	Ctrl+X		
	Сору	Ctrl+C		
	Paste	Ctrl+V		
	Insert	Ins		
	Delete	Del		
<	Select All	Ctrl+A	>	
	Expand All	NumPad *		
	Collapse	NumPad -		
	Analyze	Ctrl+Shift+B		
	Customize Columns			
	Initialize Search	Ctrl+U		
	Hyperlink			
	Go to type definition			
	Initialize Animation Table	Ctrl+T		
	Initialize New Animation Tabl	e Ctrl+Shift+T		
	Purge Unused Variables			
	Export Filtered			
	Export Selected			
	Data Properties	Ctrl+Enter		
	Unlock			

Now Export the file with the .xsy extension into the c:NiobraraPMEUCMDTM folder.

8		Export		×
Save in:	🐌 DTM	~	G 🤌 📂 🛄 -	
ea	Name	*	Date modified	Туре
Recent places		No items match your s	earch.	
Desktop				
Libraries				
Mail This PC				
	<		_	>
Network	File name:	Example 1 xsy	¥ (Export
	Save as type:	Data (*.XSY)	~	Cancel
- Options				
With DDT		Vith DFB		

Now select File > "Rebuild .txt from Unity Pro .xsy" and select the new file.

NR&D DDXML Utility				
🔄 🏵 🔻 🕇 🕌 « Nio	brara → PMEUCM → DTM	✓ ♂ Search DTM	,p	
Organize 🔻 New folder	r	8== 🔻 🔟	0	
Desktop ^	Name	Date modified Type		
Documents	example1.xsy	11/24/2015 12:00 XSY File		
Music Pictures Videos Ti10685100A (C:) Common (\\fsrv1	<		>	
example1.xsy I XSY File	Date modified: 11/24/2015 12:00 AM Size: 12.6 KB	Date created: 11/24/2015 12:00 AM Availability: Available offline		
File <u>n</u> ar	me: example1.xsy	✓ Unity Pro Variable Files, (*.xsy) Open ▼ Cancel	 ✓ 	

NOTE: If prompted "OK to overwrite PME UCM 0202_EXAMPLE1.txt", select Cancel because the .txt file is already present and need not be rebuilt.

NOTE: The generator will make the new txt file with the name "PME UCM 0202_name.txt" format.

Now the .txt file may be opened in Notepad and altered if necessary.

2	PME UCM 0202_EXAMPLE1.txt - Notepad	- 🗆 🗙
<u>F</u> ile <u>E</u> dit F <u>o</u> rmat <u>V</u> iew <u>H</u> elp		
# Generated by NRDPTKDD	XMLUTIL Rev.24NOV2015	^
SW, 1.00		
assemblyID, 1 assemblyPath, 101		
varHeading, varType, varName, varLabel,en-us,	InputVars WORD UCM_Runtime_Status .15=Run; LSB=Runtime Error	
varHeading, varType, varName, varLabel,en-us,	InputVars UINT UCM_Halt_Line_Number UCM Runtime Error Halt Line Number	
varHeading, varType, varName, varLabel,en-us,	InputVars INT In_01 In_01	
varHeading, varType, varName, varLabel,en-us,	InputVars INT In_02 In_02	
varHeading, varType, varName, varLabel,en-us,	InputVars INT In_03 In_03	
varHeading, varType, varName, varLabel,en-us,	InputVars INT In_04 In_04	
assemblyID, 2 assemblyPath, 102		
varHeading, varType, varName, varLabel,en-us,	OutputVars INT out_01 out_01	
varHeading,	OutputVars	~
<		>
		Ln 1, Col 1

When satisfied that the new .txt file will work, it may be now be installed like normal using FILE > "Install new.txt".

PMEUCM DTM Manual

6 NRD PTK DTM UTIL 57

7 QLOAD the Example1 UCM Application

The standard PMEUCM is shipped from the factory with the Example1 application preloaded. The following chapters use the Example1 application and this chapter explains how to install this application.

NOTE: It is good practice to follow these procedures even if the Example1 application is already installed in the PMEUCM.

Determine the Installed Application

The first step is to determine what (if any) application is currently installed and possibly running in the test PMEUCM.

For the purposes of this manual, it is best to stop and erase any installed application, reset the module to factory defaults, and qload the Example1.qcc file.

Install the module in an Ethernet slot of a powered M580 rack. The module will beep, blink some lights at the top, and eventually show something on the front LCD screen.



If the screen shows "Example1" at the top then the proper application is already loaded, but it is a good idea to qload a new copy anyway.

If the screen shows the OS default splash screen with App:Er=0, then the currently loaded application is halted.

If the screens shows something else, some other application may be loaded and should be

halted and erased at this time.

Halting a running Application

Most PMEUCM applications written by Niobara will include some type of screen driver with standardized menus. Use the joystick to navigate the menu to exit to the OS.

NOTE: Push "in" on the joystick for the "Enter" selection.



If the application does not include this type of screen driver but shows the OS screens, navigate to the Main > App > Switch > Halt screen.



If all else fails, push the joystick "IN" for "Enter" and "UP" at the same time and hold for 10 seconds. This will cause the application to halt and exit to the OS.

Erasing the Installed Application

Once the screen is under the UCM OS control, it is simple to erase the installed application.



Notice that the App:Er=8 after the erase command is processed. Error 8 means that the checksum for the application is bad and the app will not run.

Factory Default

Now reset the module to factory defaults.

NOTE: Factory defaults does not erase a user application. It simply sets the Ethernet and serial ports to their default settings.



The module will reboot and come up with the following settings:

Setting	Factory Default Value
E1 IP Address	10.10.10
E2 IP Address	10.10.11
E1 and E2 Subnet Mask	255.0.0.0
E1 and E2 Default Gateway	0.0.0.0
OS Modbus/TCP Port	502
S1 and S2 Mode	Modbus RTU Slave
S1 and S2 Baud Rate	9600
S1 and S2 Parity	EVEN
S1 and S2 Data Bits	8
S1 and S2 Stop Bits	1
S1 and S2 Driver	RS-232 (Fixed)

NOTE: Rebooting the UCM does not reboot the PTK board inside the PMEUCM0202. Cycling power on the rack or the PMEUCM will reboot both the UCM board and PTK board.

QLOAD Example1.qcc

The QLOAD utility is used to load applications into the PMEUCM. Start QLOAD by Start > Programs > Niobrara > QLOAD.

The first time QLOAD is started, it should look something like this:

🐚 QUCM File Downloader - 06Jun 🗕 🗆 🗙
<u>File Advanced Configure Help</u>
Load File Browse
Modbus Serial Modbus TCP Program Info Module Info
▼ 9600 ▼ Baud
255 Modbus Drop C 7 Bits
Application 1 C Application 2 S Bits
Even Parity ASCII
Set Defaults
Query Start Download Cancel

PMEUCM DTM Manual

7 QLOAD the Example1 UCM Application 62

Click on the Browse button and select this file: c:\Niobrara\PMEUCM\DTM\Example1.qcc

Ð	File to Transfer to QUCM	×
🔄 🏵 🔻 🕇 🕌 « PMI	EUCM → DTM v C	Search DTM 🔎
Organize 🔻 New folder		III 🔻 🔟 🔞
A	Name	Date modified Type
P This PC	Example1.qcc	11/24/2015 7:34 AM QCC File
Desktop	Example2.qcc	11/24/2015 7:34 AM QCC File
Documents	Example3.qcc	11/24/2015 7:34 AM QCC File
Music	Example4.qcc	11/24/2015 7:35 AM QCC File
Dictures		
Videos		
LI10685100A (C:)		
common (\\fsrv1		
	C	>
QCC File	Date modified: 11/24/2015 7:34 AM Date Size: 110 KB Av	e created: 11/24/2015 7:34 AM /ailability: Available offline
File <u>n</u> ar	me: Example1.qcc 🗸 🗸	QUCM Object 🗸
		Open Cancel

Now select the ModbusTCP tab.

🐴 QUCM File Downloader - 06Jun 🗕 🗖 🗙
File Advanced Configure Help
Load File C:\Niobrara\PMEUCM\DTM\Example IBrowse
10.10.10.10 502 TCP Port
255 Modbus Drop
Application 1 C Application 2
Set Defaults
Query Start Download Cancel

Make sure that the IP Address is set to match the PMEUCM E1 port of 10.10.10.10, the TCP Port is set to 502, Modbus Drop is 255, and Application 1 radio button is set.



Connect the Ethernet port of the computer to E1 on the PMEUCM with a standard CAT5/6 cable.

Set the Ethernet port of the computer to be on the same 10.10.10.x subnet as the PMEUCM.

Press "Start Download" to begin the loading of the program into the PMEUCM.

Example1.qcc - Downloading File			
Downloading C:\Niobrara\PMEUCM\DTM\Example1.qcc 10.10.10.10:502 255			
17721 words transmitted.			
Cancel			

When the download is finished, the program should automatically start and the screen should look something like this:

Example1
INACTIVE PLC: STOP V1.02.007 PVL:00.01
B 10.10. P 210.6
E 10.10. 1 10.10
E 10.10. 2 10.11

8 Unity Pro Operations

NOTICE: The newest version of the PTK_DTM_Libraray must be installed before attempting to use the PMEUCM. See Chapter 3.

New Project

This example starts with a new project in Unity Pro XL V8.1.

- The PME UCM 0202 will be installed in the CPU rack slot 2.
- The M580 P581020 is the chosen CPU.
- All of the IP Addresses will be left at their default settings.
 - The CPU will be at default IP Addresses of 192.168.10.1 and 192.168.11.1
 - The PME UCM backplane (PTK) will be at 192.168.10.3
 - The PME UCM E1 and E2 ports will be at 10.10.10.10 and 10.10.10.11

The BME P58 1020 CPU is chosen, along with a BME XBP 0400 four slot Ethernet backplane.

		New Project	×
Show all versions			ОК
PLC	Min OS Version	Description	Cancel
Medicen M340		boonpaon	Cancer
Modicon M540			<u>H</u> elp
BME P58 1020	01 10	CPU 580-1 ETH distributed IO	
BME P58 2020	01.10	CPU 580-2 ETH distributed IO	
BME P58 2040	01.10	CPU 580-2 ETH remote and distributed IO	
BME P58 3020	01.10	CPU 580-3 ETH distributed IO	
BME P58 3040	01.10	CPU 580-3 ETH remote and distributed IO	
BME P58 4020	01.10	CPU 580-4 ETH distributed IO	
BME P58 4040	01.10	CPU 580-4 ETH remote and distributed IO	
Momentum Unity			
• Premium			
Quantum			
L			
Rack	Description	^	
Back			
BME XBP 0400	4 SLOTS BACKPI	LANE	
BME XBP 0800	8 SLOTS BACKPI	LANE	
BME XBP 1200	12 SLOTS BACK	PLANE	
BMX XBP 0400	4 SLOTS BACKPI	LANE	
Project Settings			
Settings File:	ault settings>		
Li octungs rile.			

After selecting the "PLC Bus" in the Structural View Tree, right click on the Ethernet ports of the CPU to open the configuration submodule.



The FTP server must be enabled in the PLC for any PTK Partner module to function. The easy way to enable this server is to select "Unlock Security".

After unlocking the security, click the check box in the tool bar to accept the change.

-	Unity Pro XL : <no name="">* - [0.0 : EIO : CommHeadDIO] – 🗖</no>	×
Eile Edit View Services	s <u>T</u> ools <u>B</u> alld <u>N</u> C <u>D</u> ebug <u>W</u> indow <u>H</u> elp 🗗	×
1a 🕞 🗖 🚑 11a x r	a 🖕 🔽 🖂 🖉 🔲 🗛 🗄 🚓 🕮 🚟 🚍 🚍 📾 🖓 🐐 🗛 👘	
DIO Communicator Head		
CommHeadDIO	10 Sec 10 IPC 10 RSTP 10 SNMP 10 NTP 10 Sw 10 QoS 10 Ser 10 Adv	-
Channel 0		11
	FTP : Enabled V	
	TETP : Earlying M	
	HTTP : Enabled V	
	Access Control : Disabled	
	Enforce Security Unlock Security	
	Authorized addresses	
	IP Address Subnet Subnet mask	
	192.168.10.1 Yes 💟 255.255.0.0	
	No 🔽	
	No U	
Euroption:	No 🔽	Ш
Punction.	No 🔽	Ш
010	K Nn	Ш
	-	<u> </u>
PLC bus 0.0 : EIO : 0	C	
×		_
	port/export λ User errors λ FDT log event λ Search/Replace /	

Now close the submodule.



After right clicking on slot 2, a "New Device" is added to slot 2.

The PME UCM 0202 is selected from the "Third Party products" section.

	New Device	
Topological Address:	0.2	OK Cancel
Part Number	Description	<u>H</u> elp
Modicon M580 local drop		
🗄 ······ Analog		
Communication		
Counting		
🗄 Discrete		
Motion		
Third party products		
PME SWT 0100	Partner Module Ethemet System Weighing Transmitter 1 channel	
PME UCM 0202	Partner Module Ethernet UCM	

The UCM will now appear in the rack.



The PLC rack window may now be closed.

DTM Hardware Catalog Update

The next step is to force an update of the DTM Catalog. The DTM Catalog is accessed through Tools > Hardware Catalog.

				Unit	Pro VI + «N	0.00005*			
File Edit View St	ervices Tools	Build Pl	C Debug W	indow Hel	FIU AL . SI				
 ✓ Project <u>B</u>rowser 	Alt+1		🔎 🔍 🔎		۵ 🗠 🛃	i	l 🖻 🕅	9 M	▼ 128 26 111 모
<u>H</u> ardware Catalog	Alt+2								
Types Library Browser	Alt+3								
Operator Screen Library	Alt+4		<u>^</u>						
<u>S</u> earch / Replace	Alt+5								
Diagnostic Viewer	Alt+6								
PLC Screen	Alt+7								
Variable Window	Alt+8								
Data Editor	Alt+9								
DTM Browser	Alt+Shift+1	es							
Bookmarks	Alt+Shift+2								
Irending tool									
Convert Partially		ices							
Network Inspector		ľ							
🔁 Types Library Manager									
<u>C</u> ustomize									
Options									
Project Settings									
Animat Operat B B, Docum	tion Tables tor Screens eentation	1							
<u> </u>									
	iild / Import/	export λ	User errors	FDT log e	ent	arch/Replace	_/		
Opens the Hardware C	Catalog editor					HMI R/W mod			USB:SYS
opens are riardware e	canog canor						- Official		

The Hardware Catalog Window should appear and look something like this:

 m Device types m Vendors m Groups m Protocols 	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	140NOC77100 (from 140NOC77101 (from 140NOC78000 (from 140NOC78100 (from Advanced Generic E ALTIVAR61 Revision ALTIVAR61 Revision ALTIVAR61 Revision ALTIVAR71 Revision ALTIVAR71 Revision BME AHI 0812 BME AHI 0812 BME AHI 0412 BME AHI 0412	Device Device Device Device Device Device Device Device Device Device Gateway Gateway	Si Si
	<	BMENOCUSUI	Communication	>

Click on the "DTM catalog" tab at the bottom.

Then Click on the "Update" button.

A message box should pop up asking if it is ok to update the catalog. Select "Yes".

Note: This box opens every time the Update button is clicked – even if the catalog is updated twice in a row with no other changes.

Unity Pro XL	
? The DTM Catalog seems to be up of date.\Would you like to update it?	
<u>Y</u> es <u>N</u> o	

A progress window pops open.

Updating Dtm catalog
62%
GenericEDSDTM.DTMCore.1
Cancel

After the catalog update is complete, the new Niobrara DTM device should be listed in the hardware catalog. Also, the "User Errors" display should show "Information: The update of the Dtm catalog is finished"

®	Unity Pro XL : EXAMPLE1*		
File Edit View Services Tools Build PLC Debug Window Help	Ø □ ■ ■ ■ ■ ■ ■ ■ ■ ■ ₩ ₩ ₩ :		() () 4
Project provser Structural view Struct	Advanced More Catalog Vendos Advanced More Correct Advanced More Correct Advanced More Correct Advanced More Correct Advanced More Correct HINS Industrial Networks Rockwell Automation/Rev Correct Rockwell Automation/Rev Correct Rock	Device Type ME UCM 0202 EXAMPLE1 Device alog /	Vendo Nebrara
Information: The Update of the Dtm catalog is finished information: The Update of the Dtm catalog is finished information: The Update of the Dtm catalog is finished	FOLL a control A Search (Backers		
	rs A FDT log event A Search/Heplace /		

Now, Open the DTM browser by selecting Tools > DTM Browser.

8	Unity Pro XL : <no name="">* Z</no>
File Edit View Services Tools	Build PLC Debug Window Help
✓ Project <u>B</u> rowser Alt+1	p 🗠 🔽 🥔 🖾 🔲 🛃 🔯 🕮 🚟 🔚 🔚 🖶 🐐 🗰 🗛
Hardware Catalog Alt+2	
In Types Library Browser Alt+3	
Operator Screen Library Alt+4	
Search / Replace Alt+5	—
Diagnostic Viewer Alt+6	
PLC Screen Alt+7	
Variable Window Alt+8	
Data Editor Alt+9	
DTM Browser Alt+Shift+1	
Bookmarks Alt+Shift+2	
Trending tool	
Convert Partially	nces
Network Inspector	
🔁 Types Library Manager	
<u>C</u> ustomize	
<u>O</u> ptions	
Project Settings	
Animation Tables Operator Screens Documentation	
Information: The Update of t	the Dtm catalog is finished
	export λ User errors / FDT log event λ Search/Replace /
Opens DTM Browser	HMI R/W mode OFFLINE

PMEUCM DTM Manual

The DTM Browser will open and show a tree with the CPU at 192.168.10.1. Right click on the CPU and select "Add".

Eile Edit View Services Teel	Unity Pro XL : No name>^
DTM Browser	
₩ Host PC	
	Open
	Add
	Delete Del
-	
	Sort by address
Project Browser	Connect
E Structural view	Disconnect
Project	Load data from device
E B Conngeration	Store data to device
2 : EIO Bus	Copy
Derived FB Types	Paste
Variables & FB inst	Device menu
Derived Variables	Properties ALT+Enter
IO Derived Variat	Print device
Elementary FB In	Zoom out
Motion	Expand all
Communication	Collapse all
<	>
Information: The Update of the Dtm catalog is finished	
UNITED A Import/export A User errors A FDT log event A Search/Replace /	
Add new device.	HMI R/W mode OFFLINE
A window will pop up showing all of the installed DTMs. Scroll down until you reach the PME UCM 0202 EXAMPLE1 device by Niobrara.

NOTE: This installation is using the original version of the EXAMPLE1 file with all variables being INTs.

Notice that it has the version 01.00 which matches the SW version in the txt file.

	Ad	dd			
Device	Туре	Vendor	Version	Date	~
Anybus Communicator - Slave R	Device	HMS Industrial N	1.60		
Anybus Communicator - Slave R	Device	HMS Industrial N	2.3		
Anybus Communicator CAN (from	Device	HMS Industrial N	1.3		
Anybus-C EtherNet/IP (from EDS)	Device	HMS Industrial N	1.27		
Anybus-S EtherNet/IP Revision	Device	HMS Industrial N	1.26		
Anybus-S EtherNet/IP Revision	Device	HMS Industrial N	2.1		
PME UCM 0202 EXAMPLE1	Device	Niobrara	01.00	2015-11-24	
1305 AC Drive Revision 6.1 (from	Device	Rockwell Autom	6.1		
1305 AC Drive Revision 7.1 (from	Device	Rockwell Autom	7.1		
1336 IMPACT Drive Revision 1.1	Device	Rockwell Autom	1.1		
1336 IMPACT Drive Revision 2.1	Device	Rockwell Autom	2.1		
1336 IMPACT Drive Revision 3.1	Device	Rockwell Autom	3.1		
1336 IMPACT Drive Revision 4.1	Device	Rockwell Autom	4.1		
1336 PLUS Drive Revision 1.1 (fr	Device	Rockwell Autom	1.1		
1336 PLUS Drive Revision 1.1 (fr	Device	Rockwell Autom	1.1		
1336 PLUS Drive Revision 2.1 (fr	Device	Rockwell Autom	2.1		
1336 PLUS Drive Revision 2.1 (fr	Device	Rockwell Autom	2.1		
1336 PLUS Drive Revision 3.1 (fr	Device	Rockwell Autom	3.1		
1336 PLUS Drive Revision 3.1 fr	Device	Rockwell Autom	31		¥
Add DTM				Close	

Press Enter or "Add DTM" to load the DTM for the PMEUCM. A window will pop up with information about the DTM.

At this point the "Alias name" may be modified. The can be very handy for shortening the variable names created by the DTM.

	Properties of device	x
General	Device information DTM information Protocol information	
- DTM r	name management	
Alias	name : PME_UCM_0202_EXAMPLE1	
Tagr	name :	
Defau	t I/O vision management	
Defa	ult input I/O vision	
Va	ariable name : PME_UCM_0202_EXAMPLE1_IN	
Defa	ult output I/O vision	
Va	ariable name : PME_UCM_0202_EXAMPLE1_OUT	
	OK Cancel Help)

Pressing "OK" will add the DTM device to the DTM Browser.

Add new device.
100% - PME_UCM_0202_EXAMPLE1

The PMEUCM is now added to the tree below the PLC.

🔞 Unity Pro XL : EXAMPLE1* 🗕 🕻	⊐ ×
<u>File Edit View S</u> ervices <u>T</u> ools <u>B</u> uild <u>P</u> LC <u>D</u> ebug <u>W</u> indow <u>H</u> elp	
🎦 🎜 🖶 🎒 🕺 🛍 🗤 🖓 🔽 🖉 💭 💷 🏄 🎉 🎬 🛗 🔂 🛪 🗰 🗛 🗰 🗛	
DTM Browser	1
Host PC P Image: State of the state	
Project Browser	
B Structural view	
Example1 Configuration O : PLC bus Derived Data Types Derived FB Types Elementary Variables	
Information: The Update of the Dtm catalog is finished Information: The Update of the Dtm catalog is finished Information: The Update of the Dtm catalog is finished I	
Ready HMI R/W mode OFFLINE	TCPIP

Unity Variables

At this point the variables defined in the DTM are not yet present in the Unity Project. This is because the project has not been Analyzed or Built. Opening the "Variables & FB instances" item will verify that there are no variables in the system.





Now select Build > Analyze Project. A window will pop up showing the progress.

After the completion of the Analyze Project, two variable structures will appear. One for the CPU and one for the PMEUCM.

Notice that the PMEUCM structure includes Freshness BOOLs and both Input and Output structures.

The Freshness BOOL values are 0 if the PLC is unable to connect with the PMEUCM via Ethernet/IP across the backplane. If the data connection is working, the Freshness variables will have the value 1.

The Input structure includes 76 bytes of PTK header information. Most of this can be ignored in normal operation. The UCM data starts after the "Free2" BYTE. Variables named FreeX are place holder variables that are used to align certain variable types to 2, 4, or 8 byte boundaries in the PLC memory.

A closer look at the UCM variables shows they match the names, types, and descriptions from the txt file.

PMEUCM DTM Manual



	BOOL
🗝 🐣 Free2	BYTE
	WORD
🗝 😓 UCM_Halt_Line_Number	UINT
	INT
	INT
	INT
	INT
📩 🚇 🗂 Outouto	T DME LICM

		PME UCM 0202_EXAMPLE1.txt - Notepad	-	×
<u>F</u> ile <u>E</u> dit	F <u>o</u> rmat <u>V</u> iew <u>H</u> elp			
SW,		01.00		^
	TD			
assembl	yID, vPath	1		
assembl	vDefaultRPT	15		
assembl	vMinRPT.	5		
assembl	yMaxRPI.	200		
varHead	ing,	InputVars		
	varType,	WORD		
	varName,	UCM_Runtime_Status		
	varLabel,en-us,	.15=Kun; LSB=Kuntime Error		
varHead	ing.	InputVars		
	varTvpe.	UINT		
	varName,	UCM Halt Line Number		
	varLabel,en-us,	UCM Runtime Error Halt Line Number		
varHead	ing,	InputVars		
	varType,	INT		
	varName,	In_01		
	varLadel,en-us,	11_01		
varHead	ing,	InputVars		
	varType,	INT		
	varName,	In_02		
	varLabel,en-us,	In_02		
		T		
varHead	ing,	Inputvars		
	variype,			
	varlabel.en-us.	In_03		
	var Eubergen ubg	11_05		
varHead	ing,	InputVars		
	varType,	INT		
	varName,	In_04		
	varLabel,en-us,	In_04		
assembl	yID,	2		
assembl	yPath,	102		
varHead	ing,	outputVars		
	varType,	INT		
	varName,	OUT_01		~
<				>
			Ln 1, Col 1	

The Output variable structure includes the 26 byte PTK header information followed by the UCM data.

	IN I
🖮 🚜 🗇 Outputs	T_PME_UCM_0202_EXAMPLE1_O
	REAL
	UINT
	BYTE
🕀 👝 📕 📕 Free3	ARRAY[06] OF BYTE
	INT

		PME UCM 0202_EXAMPLE1.txt - Notepad	-		
<u>F</u> ile <u>E</u> dit	F <u>o</u> rmat <u>V</u> iew <u>H</u> elp				٦
varHead.	ing, varType, varName, varName,	InputVars INT In_04 In_04		^	
assembl assembl	yID, yPath,	2 102			
varHead:	ing, varType, varName, varLabel,en-us,	outputVars INT out_01 out_01			
varHead:	ing, varType, varName, varLabel,en-us,	outputVars INT out_02 out_02			
varHead:	ing, varType, varName, varLabel,en-us,	outputVars INT out_03 out_03			
varHead:	ing, varType, varName, varLabel,en-us,	outputVars INT out_04 out_04			
varHead:	ing, varType, varName, varLabel,en-us,	outputVars INT out_05 out_05			
varHead:	ing, varType, varName, varLabel,en-us,	outputVars INT out_06 out_06			
varHead	ing, varType, varName, varLabel,en-us,	outputVars INT out_07 out_07		,	
<				>	
			Ln 41, Col 15		

PMEUCM DTM Manual

Steps for Modifying the Installed DTM

- 1. Edit the txt file.
- 2. Save the changes in the txt file.
- 3. "Install" the txt file with the NRD DTM Util.
- 4. Perform an UPDATE on the DTM Hardware Catalog in Unity Pro.
- 5. Delete the UCM DTM item from the DTM Browser.
- 6. Add the UCM DTM item back into the DTM Browser.
- 7. BUILD or ANALYZE the Project to generate the new variables.

NOTE: Step 5 is vitally important. Simply updating the DTM Hardware Catalog does not affect installed DTMs. Installed DTMs must be removed and then Added for the changes to the DTM to occur.

Link the DTM to the PMEUCM Hardware

Once the variable lists have been finalized, it is time to actually associate the DTM instance with the actual PMEUCM device. This is done inside the DTM Browser window.

Right click on the CPU and select Open.

*	Unity Pro XL : <i< th=""><th>No name>* _ 🗆 🗙</th></i<>	No name>* _ 🗆 🗙
File Edit View Services Tools	Build PLC Debug Window H	Help
12 🚅 🗖 🗇 🛛 🖿 🖉 🕞	v v 🖂 🖉 🦀 🕼 🔲 🖥	5 🖉 🛗 📻 🔚 🖼 🗛 🗰 🗛
DTM Browser	×	1
Nost PC		
⊟	Open	
	Add	
<	Delete Del	
Project Browser	Field bus discovery	
E Structural view	Sort by address	
Project	Connect	
Configuration	Disconnect	
	Load data from device	
Derived Data Types	Store data to device	
🖃 🔄 Variables & FB insta	Сору	
Elementary Variable	Paste	
Device DDT Variabl	Device menu	Data Editor
Analyzing	Properties ALT+Enter	
Project settings disabled : Process succeeded : 0 Err	Print device	DIEO EF (EN=V)
Analyze Proje	Zoom in	À FDT log event À Search/Replace /
Open device editor.	Zoom out	HMI R/W mode OFFLINE .:
	Expand all	
	Collapse all	

NOTE: The "Source IP Address" is a pull-down listing of all of the IP Addresses of the Unity Pro PC. Make sure to select an address that is on the same subnet as the M580 PLC. In this case the IP Address of 192.168.100.200 is selected since the PLC is at 192.168.10.1.

BMEP58_ECPU Communication BME P58 1020		
Channel Properties TCP/IP Services Address Server EtherNet/IP Local Slaves Local Slave 1	Source Address: Source IP Address: Sub-Network Mask:	192.168.10.200 V 255 . 255 . 255 . 0
Items Iccal Slave 2 Local Slave 3 Local Slave 3	EtherNet/IP Network Detection: Begin detection range address: End detection range address:	192 . 168 . 10 . 1 192 . 168 . 10 . 254
	Modbus Network Detection: Begin detection range address: End detection range address:	192 . 168 . 10 . 1 192 . 168 . 10 . 254
< >> Help	OK	Cancel Apply
에 Disconnected 이 Data set		

After Setting the Source IP Address, click on the PME_UCM_0202_... entry in the list on the left.

Select the "Address Setting" Tab.

BMEP58_ECPU		
Communication		Schneider
BME P58 1020		U Electric
- Channel Properties	Properties Address Setting	
TCP/IP		
😑 Services	IP Configuration	
Address Server	The difference	102 169 10 2
EtherNet/IP Local Slaves	IP Address:	152 . 105 . 10 . 5
E Local Slave 1		
ltems	Subnet Mask:	255.255.0.0
Local Slave 2		
Items	Gateway:	192 . 168 . 10 . 1
ia Local Slave 3		
Items		
Device List	Address Server	
□ [040] PME_UCM_0202_EXAMPLET <eip: 192.168.1<="" td=""><td></td><td></td></eip:>		
) DHCP for this device:	Enabled V
items		
Logging	Identified by:	Device Name 🗸 🗸
	Identifier:	Mx80_01_PMEUCM02
< >		
Help		OK Cancel Apply
-⊅D Disconnected ① Data set		
BMEP58_EC		

The "Identifier" must be modified to define the exact Rack and Slot occupied by

the PMEUCM.

In this example, the PMEUCM is located in the CPU rack, Slot 3. Therefore, the Identifier must be set for "Mx80 02 PMEUCM02".

Enabled	The format MUST be :- Mx80_XX_PMEUCM02 or M58A_XX_PMEUCM02 or M58B_XX_PMEUCM02 or C000_XX_PMEUCM02
Device Name	XX = Slot number(range 01-11) YYY = CRA rack no.(range 000-159
Mx80_02_PMEUCM02	

NOTE: If the PMEUCM is located in a remote rack, the YYY value is the thumbwheel (rotary switches) setting of the eCRA, not necessarily the logical rack number.

Build All

After setting the Identifier, it is time to do a Build All of the Project.

Select "Build > Rebuild All Project"



Transfer Project to PLC

After a successful Build, it is time to transfer the project to the M580. Connect the Ethernet port of the PC to the Service Port of the M580.

PLC Set Address

Select PLC > Set Address and choose TCPIP for the Media and set the Address of the M580 (192.168.10.1).

	Set Address	? ×
✓ PLC Address 192.168.10.1 √	Simulator Address 127.0.0.1 V 💼 Media	Bandwidth
TCPIP	TCPIP <u>Communication Parameters</u>	OK Cancel
\checkmark Speed rate auto-adaptation at the end of dow	vnload	<u>H</u> elp

It is usually a good idea to try the "Test Connection" button to make sure that the PC can connect with the M580.

PLC Connect

Now select PLC > Connect to open a connection to the M580 CPU.

Connect Ctrl X Set Address Stadaddress B'stadad Mode DDT Types Function Blocks DFB Types Compare DDT Types Function Blocks DFB Types Compare Image Project from PLC Transfer Project from PLC Ctrl - Shift-L Transfer Project from PLC Ctrl - Shift-L Transfer Project from PLC Ctrl - Shift-L See blaf from FIC to Ele Ser VPR SERVICE_B1 Restore Data from FIC to Ele Ctrl - Shift-L Restore Data from FIC to Ele Ctrl - Shift-L Mun/Stop Ctrl - Shift-L Jift Ctrl - Shift-L Update Lybioad Information Ctrl - Shift-L Compare BOOL Compare Ctrl - Shift-L Compare Ctrl - Shift-L Data Edit Mode Sate y/Maintenace Ctrl - Shift-L Dide Lybioad Information Ctrl - Shift-L Compare Ctrl - Shift-L Compare <th>Eile Edit View Services Tools Build PLC</th> <th><u>D</u>ebug <u>W</u>indow <u>H</u>elp</th> <th></th> <th>- 6</th>	Eile Edit View Services Tools Build PLC	<u>D</u> ebug <u>W</u> indow <u>H</u> elp		- 6
Set Address 3) Standard Mode is Simulation Mode Cgmpare If Inder Project to PLC Carl I framder Project from PLC or Cirl - Shift-Lirl Trander Project from PLC to Elle Restore Data from PLC to Elle Restore Data from PLC to Elle Restore Data from PLC to Elle Init. Update Upload Information Update Upload Information Update Upload Information Update Upload Information Init. Update Upload Information Cost Dort With PLC Init Values. Update Upload Information Init. Update Upload Information It is the Ram Viewer Is Elley Cost Difference Init. Update Local Init Values. Init. Init. Update Upload Information Init. Init.	<u>C</u> onnect Ctrl+K	🔍 🖾 🛑 🔛 🔚 🔚	🗄 🔲 🖶 🛪 🗱 🗛	
33 Standard Mode 2 Simulation Mode 2 Transfer Project to PLC C Crit-Shift-L 7 and For Project from Plinary to StandBy PLC Save Data from PLC to File Restore Data from File to PLC Save Data from File to PLC Save Data from File to PLC Save Data from PLC to File Restore Data from File to PLC Save Data from PLC to File Restore Data from File to PLC Save Data from PLC to File Restore Data from File to PLC Save Data from PLC to File Restore Data from File to PLC Save Data from PLC Christian Christian Bata from PLC Christian Christian Save Data from PLC Christian Save Data from PLC Save Data from PLC Christian Save Data from PLC Save Data from PLC Save Data from PLC Christian	Set <u>A</u> ddress			
Simulation Mode Compare Inansfer Project to PLC Ctrl+L Transfer Project from PLC bit Transfer Project from PLC Save Data from PLC bit Save Data from PLC S	🖞 Standard Mode			
Compare Torsder Project ton PLC Cort-L Transfer Project from PLC Cort-Shift-L Transfer Project from PLC Cort-Shift-L Transfer Project from PLC Cort-Shift-L Save Data from File to PLC Save Data from File to PLC Cort-Shift-M Rum/Stop Cort-R Dat Cort-Shift-M Cort-	Simulation Mode	DDT Types Function Blocks DFB Types		
Innifer Project to PLC Ctri+L Trander Project to PLC Ctri+Shift-L Trander Project to PLC to File Restore Data from PLC Ctri-Shift-M Run/Stop R	Compare			т "
Transfer Project from PLC Ctrl-Shift-L Transfer Project from PLC SNMP_SERVICE Save Data from PLC to File BOOL Relation PLC to File POR_SERVICE B2 BOOL Save Data from PLC to File POR_SERVICE B2 BOOL Save Data from PLC to File POR_SERVICE B2 BOOL Save Data from PLC to File POR_SERVICE B3 BOOL Save Data from PLC to File POR_SERVICE B4 BOOL Save Data from PLC to File POR_SERVICE B4 BOOL Save Data from PLC to File POR_SERVICE B4 BOOL Save Data from PLC POR_SERVICE B4 BOOL Int POR_SERVICE B4 BOOL Update Load Information Ctrl+R POR_SERVICE B4 BOOL Update Load Init Values with PLC Init Values. POR Init PARCE UNINT POR_SERVICE B4 Memory Consumption SIGNATURE UNINT POR_SERVICE B4 POR_SERVICE B4 Derived Variable PI	Transfer Project to PLC Ctrl+L	Name =		. "
Transfer Project from Primary to StandBy PLC Transfer Project from Primary to StandBy PLC Save Data from PLC to Ele Restore Data from PLC to Ele PERSERVICE B3 BOOL Save Data Edata PLC TV PERSERVICE B3 BOOL Save Data Edata PLC TV PERSERVICE B3 BOOL Save Data Edata PLC TV PERSON UNIT Program PERSERVICE B3 BOOL Save Data Edata PLC TV PERSON UNIT PRESERVICE B3 PLC TV PERSON UNIT PRESERVICE B3 PLC TV PERSON UNIT PRESERVICE B3 PLC TV PERSON UNIT PLC TV PERSON	Transfer Project from PLC Ctrl+Shift+L	-	Type 👻	Adc ^
Save Data Froget Notificity O Jambury PLC Image: Service_B1 Bool Save Data from PLC to File FOR, SERVICE_B2 Bool Retore_Bat from File to PLC Image: Service_B1 Bool Safety/Maintenance Ctrl-Shift-M Image: Service_B1 Bool Run/Stop Ctrl-R Image: Service_B1 Bool Juit Image: Service_B1 Bool Juit Image: Service_B1 Bool Jupdate Local Init Values. Image: Service_B1 Bool Update Local Init Values. Image: Service_B1 UNIT Update Local Init Values. Image: SignATUS UNIT Image: Service B1 Bool Image: SignATUS Image: Service B2 Biol Image: SignATUS Update Local Init Values. Image: SignATUS UNIT Image: Service B1 Image: SignATUS UNIT Image: SignATUS Image: SignATUS UNIT Image: SignATUS Image: SignATUS UNIT Image: SignATUS Image: SignATUS Image: SignATUS Image: SignATUS Imag	Transfer Project from Drimany to StandBy DLC	SNMP_SERVICE	BOOL	
Save Data from PIC to Elie Image: Star Vice_B2 BOOL Restore Data from File to PIC FOR, SERVICE_B3 BOOL Safety/Maintenance Ctrl+Shift+M FOR, SERVICE_B4 BOOL KuryStop Ctrl+R ETH-PORTI_FUNCTION_B1 BOOL Jint Ctrl+R ETH-PORTI_FUNCTION_B1 BOOL Update Upload Information Ctrl+R ETH-PORTI_FUNCTION_B1 BOOL Update Upload Information Ctrl+R UMR true_Pathing WORO Update Upload Information Ctrl+R Image: Ctrl+R Image: Ctrl+R Update Upload Information Image: Ctrl+R Image: Ctrl+R Image: Ctrl+R Update Upload Information Image: Ctrl+R Image: Ctrl+R Image: Ctrl+R Update Local Init Values with PLC Init Values. Image: Ctrl+R Image: Ctrl+R Image: Ctrl+R Image: Ctrl+R Image: Ctrl+R Image: Ctr	mansfel Project from Primary to Standby PCC	FDR_SERVICE_B1	BOOL	
Restore Data from File to PLC Image: Straw CE B3 BOOL Safety/Maintenance Ctrl-Shift-M Image: Straw CE B4 BOOL Safety/Maintenance Ctrl-Shift-M Image: Straw CE B4 BOOL Maintenance Ctrl-Shift-M Image: Straw CE B4 BOOL Maintenance Ctrl-Shift-M Image: Straw CE B4 BOOL Image: Straw CE B4 Ctrl-R Image: Straw CE B4 BOOL Image: Straw CE B4 Ctrl-R Image: Straw CE B4 BOOL Image: Straw CE B4 Ctrl-R Image: Straw CE B4 BOOL Image: Straw CE B4 Ctrl-R Image: Straw CE B4 BOOL Image: Straw CE B4 Ctrl-R Image: Straw CE B4 BOOL Image: Straw CE B4 UCM, Partime Stata WORD Image: Straw CE B4 Image: Straw CE B4 Image: Straw CE B4 Image: Straw CE B4 Image: Straw CE B4 Image: Straw CE B4 Image: Straw CE B4 Image: Straw CE B4 Image: Straw CE B4 Image: Straw CE B4 Image: Straw CE B4 Image: Straw CE B4 Image: Straw CE B4 Image: Straw CE B4 Image: Straw CE B4 Image: Straw CE B4 Image: Straw CE B4	Save Data from PLC to File		BOOL	
Safety/Maintenance Ctrl+Shift-M ^A FOR, SERVICE_84 ^B BOOL ^A FOR, SERVICE_84 ^B BOOR ^A FOR, SERVICE_84 ^A FOR, SERVICE_84 ^A FOR, SERVICE_84 ^B BOOR ^A FOR, SERVICE_84 ^A	Restore Data from File to PLC		BOOL	
Safety/Maintenance Ctri-Shift-M & & ETH-PORTI_INFO Run/Stop Ctri-R & & ETH-PORTI_INFO Prior Ctri-R & & ETH-PORTI_FUNCTION_B1 BOOL pit & & ETH-PORTI_FUNCTION_B2 BOOL & UCM_Hat_Line_Number UDINT & Device With PLC Init Values. & In_02 INIT & BLOCK_LENGTH & BLOCK_LENGTHER UNIT & BLOCK_STATUS UNIT & BLOCK_STATUS UNIT & BLOCK_STATUS UNIT & BLOCK_STATUS UNIT & BLOCK_STATUS UNIT & Deviced Wanable & PLC_TYPE_IO BYTE & Deviced B Instance & PLC_TYPE_IO BYTE & Deviced B Instance & PLC_TYPE_IO BYTE & PLC_TYPE_IO BYTE & Device DITYTE INTER & PLC_TYPE_IO BYTE & Device DITYTE INTER & Device DITYTE INTER & Device DITYTE INTER & Device DITYTE INTER & PLC_TYPE_IO BYTE & Device DITYTE INTER & DITYTE INTER & DITYTE INTER & DITYTE INTER & DITYTE INT	Notore Bata Holl Hie to Fee	_ R SERVICE_B4	BOOL	
Burry Stop Ctrl - R	Safety/Maintenance Ctrl+Shift+M	ETH_PORT1_INFO	BYTE	
Init	Run/Stop Ctrl+R	ETH_PORT1_FUNCTION_B1	BOOL	
Image: Second	Init	ETH_PORT1_FUNCTION_B2	BOOL	
Update Upload Information Image: Status W0/U Update Lipload Information Image: Status W0/U Update Local Init Values. Image: Status UNIT Update Local Init Values. Image: Status UNIT Update Local Init Values. Image: Status UNIT Project Backup Image: Status UNIT Memory Consumption Image: Status UNIT State Ram Viewer BLOCK, IDENTIFIER UNIT Image: Derived Variable Image: Status UNIT Image: Derived Variable Image: Derived Variable Image: Derived Variable Image: Derived Variable Image: Derived Variable Image: Derived Variable Image: Derived Variable		Hree2	BYIE	_
Update link Values with Current Values. 0 0.0.1 mg Line Values 0.0.1 mg Line Values Update link Values with PLC Init Values. 0 0.0.1 mg Line Values 0.0.1 mg Line Values Project Backup 0 0.0.2 mg Link Values 0 0.0.1 mg Link Values Project Backup 0 0.0.2 mg Link Values 0 0.0.2 mg Link Values Memory Consumption 0 0.0.0.7 mg Link Values 0 0.0.0.7 mg Link Values State Ram Viewer 0 0.0.0.7 mg Link Values 0 0.0.0.7 mg Link Values Derived Variable 0 0.0.0.7 mg Link Values 0 0.0.1 mg Link Values Derived PB Instance 0 0.0.2 mg Link Values 0 0.0.1 mg Link Values Website Communication 0 0.0.2 mg Link Values 0 0.0.1 mg Link Values Whotom 0 0.0.1 mg Link Values 0 0.0.1 mg Link Values	Update Upload Information	CM_Runtime_Status	WORD	_
Update Local Init Values with PLC init Values.		Com_hat_une_number	LIDINT	_
Update Local (nit Values with PLC init Values. Image: Nit Market Marke	Update Init Values with Current Values.		INT	
Project Backup Project Back	Update Local Init Values with PLC Init Values.		UINT	
Project gackup ASS OUT TAG PEL Memory Consumption BLOCK_LEMSTH UINT State Ram Viewer BLOCK_LEMSTH UINT BLOCK_LEMSTH UINT BLOCK_CENSTH BLOCK_STATUS UINT BLOCK_STATUS UINT BLOCK_STATUS UINT BLOCK_STATUS UINT BLOCK_STATUS UINT BLOCK_STATE BYTE BLOCK_DOTWORD BInstance PLC_STATE BTE PLC_TYPE_ID BYTE PLC_TYPE_ID BYTE Image: Communication Image: Communication Image: Communication		Cutouts	T PME UCM 0202 botest3in4out	
Memory Consumption Image: BioCK, LENGTH UNIT State Ram Viewer BLOCK, DENTHER UNIT Image: Device DOT Variable BLOCK, DENTHER UNIT Image: Device DOT Variable BLOCK, DENTHER UNIT Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device DOT Variable Image: Device	Project Backup	ASS OUT TAG	REAL	
BioCK JDENTIFIER UINT BioCK JDENTIFIER UINT Device DOT Variable Dot Variable Dot Variable Device Variable Dot Variable Device Variabl	Memory Consumption	BLOCK LENGTH	UINT	
State Kam Viewer Device DOT Variable UNT Do Device DOT Variable SiGNATURE UNT Do Device Variable SiGNATURE UNT Do Device Variable PIC, START BYTE Device dFB Instance PIC, START BYTE Program PIC, START BYTE Data Edtor State Edtor	Memory consumption	BLOCK IDENTIFIER	UINT	
Convector Variable SigNaTure UNIT Society Variable Society Variable Society Version UNIT Overweid PB Instarc Over	State Ram Viewer	BLOCK_STATUS	UINT	
ID Derived Variable Pic, FW, VERSION UINT Pic, START	Device DDT Variable	🛁 🔶 SIGNATURE	UINT	
Benentary FB Insta Derived FB Instance Motion Communication Plc_START BYTE Derived FB Instance Motion Ormanication Derived FB Instance Derived FB Instanc	IO Derived Variable:	Reference PLC_FW_VERSION	UINT	
Communication Communi	Elementary FB Insta	A PLC_STATE	BYTE	
Motion Communication	Derived FB Instance		BYTE	
Communication Program Linking Process succeeded : 0 Error(s), 0 Warning(s)	Motion		BYTE	~
Program Program Data Editor Data Editor Process succeeded : 0 Error(s) , 0 Warning(s)	Communication			>
Cata Estor Data Estor Deta Estor Process succeeded : 0 Error(s) , 0 Warning(s)	🔋 📄 👘 Program 🛛 👻 🛛 🖵			
JLinking Process succeeded : 0 Error(s) , 0 Warning(s)	< > Dat	a Editor		
ی Linking Process succeeded : 0 Error(s) , 0 Warning(s)		/		
Process succeeded : 0 Error(s) , 0 Warning(s)	≚Linking			
	Process succeeded : 0 Error(s) . 0 Warnir	naís)		
		-3(-)		

Transfer Project to PLC

After connecting, transfer the project to the PLC.

Eile Edit View Services Tools Build PL	C <u>D</u> ebug <u>W</u> indow <u>H</u> elp		_
Disconn <u>e</u> ct Ctrl+K	- 4 d - 1		
Set <u>A</u> ddress			
l Standard Mode			
simulation Mode		7	
~	DDT Types Function Blocks DFB Types		
Compare	🝸 🦓 Name = *	EDT UDDT UDD)T »
in Iransfer Project to PLC Ctrl+L		- Tune	Ade
Iransfer Project from PLC Ctrl+Shift+L		ROOL	1000
Transfer Project from Primary to StandBy PLC		ROOL	-
		BOOL	
Save Data from PLC to File		BOOL	
Restore <u>D</u> ata from File to PLC		BOOL	
		BYTE	-
Safety/Maintenance Ctrl+Shift+M		BOOL	-
Stop Ctrl+R		BOOL	
Init		BYTE	
	UCM Buntime Statue	WORD	<u> </u>
Update <u>U</u> pload Information	- A UCM Halt Line Number	LIINT	
		UDINT	
Update Init <u>V</u> alues with Current Values.		INT	
Update Local Init Values with PLC Init Values.		UINT	
		T PME UCM 0202 botest3in4out	
Project <u>B</u> ackup	ASS OUT TAG	BEAL	
Manage Commention	BLOCK LENGTH	UINT	
Memory Consumption		UINT	
State Ram Viewer	BLOCK STATUS	UINT	
Device DDT Variable	SIGNATURE	UINT	
IO Derived Variable:	R PLC FW VERSION	UINT	
Elementary FB Insta	A PLC STATE	BYTE	
Derived FB Instance	A PLC START	BYTE	
Motion	PLC TYPE ID	BYTE	
Communication <		· · · · · · · · · · · · · · · · · · ·	>
🛛 🖯 🗁 🦳 Program 🗸 🗍 🖵			
	ta Editor		
, 3			
× Linking			
Process succeeded : 0 Error(s) 0 Warsi	ng(e)		
TIOCCOS SUCCECUEU . O ETIOT(S), O YYAMII	ng(ə)		
Import Import	t∕export ∧ Useremons ∧ FDT log even	nt 🔥 Search/Replace /	

The Transfer Project to PLC window should look something like this:

Transfer Project to PLC					
PC Project	Overwritten PLC Project				
Name: Version:	Example1	Name: Version:	Example1 0.0.4		
Last Build:	11/24/2015 12:39:54 AM	Last Build:	11/24/2015 12:39:54 AM		
PLC Ru	n after Transfer Transfer		Cancel		

It is usually convenient to check the PLC Run after Transfer box.

PMEUCM DTM Manual

If the PLC is in RUN, you will be prompted to Stop the M580.

WARINING: Stopping a running PLC may result in injury or death. Make sure that you understand the consequences of halting a running program.

Stop	×
PLC Project: Name: Example1 Version: 0.0.4 Last Build: 11/24/2015 12:39:54 AM Confirm Stop on this Project?	
OK Cancel	

The transfer should look like this:

Project transfer
25% - Wait for PLCReady after download (test 1)
Cancel

The Run confirmation screen will be shown if the "Run after Transfer" was selected.



Selecting "OK" will start the PLC.

Transfer to FDR Server

An additional step that must be followed after any configuration change within the DTM is the "Transfer to FDR Server".

This is accomplished by right clicking on the UCM's entry in the DTM Browser, select "Device Menu", "Additional Functions", and finally "Transfer to FDR Server"

	Eile Edit View	Services Iools Build PLC De	window Help □ □ ▲ ◎ ─ ─ ─ ■		-
	DTM Browser	Variables DE Filter Filter	T Types Function Blocks DFB Types		IDT »
		Open Add Delete Del	SNMP_SERVICE FDR_SERVICE_B1 FDR_SERVICE_B2	Type BOOL BOOL BOOL	Adc A
	< Project Browser	Field bus discovery Sort by address Connect	FDR_SERVICE_B3 FDR_SERVICE_B4 ETH_PORT1_INFO ETH_PORT1_FUNCTION_B1 ETH_PORT1_FUNCTION_B2	BOOL BOOL BYTE BOOL BOOL	
	Configure	Disconnect Load data from device Store data to device	Free2 UCM_Runtime_Status UCM_Halt_Line_Number In_01	BYTE WORD UINT UDINT	
	2 : E Derived	0: TO Copy Da Paste	In_02 In_03 Itputs ASS_OUT_TAG	INT UINT T_PME_UCM_0202_bptest3in4out REAL	
	Online Parameter Online Parameter	Properties ALT+Enter	BLOCK_LENGTH BLOCK_IDENTIFIER BLOCK_STATUS	UINT UINT UINT	
	Configuration Observe	Print device Zoom out		UINT UINT BYTE BYTE	<u>+</u>
Transfer to FDR Server	Diagnosis Additional functions	Expand all Coliapse all	PLC_TYPE_ID	BYTE	>
Help	▲ Linking Process succee	ded : 0 Error(s) , 0 Warning(s)) liser errors) EDT los avec	t). Search/Replace /	

A progress box will be shown during the transfer.

BMEP58_ECPU
100% - FDR transfer completed (100%)

A confirmation box will be shown to show the success of the transfer.



A box will then be shown to remind the user of the requirement to cycle power on the PTK board (entire PMEUCM module) before the new settings will be used.

Reset Device	×
The module must be restarted to apply new configuration change.	
ОК	

Cycle Power on the PMEUCM

There are three methods of cycling power on the PMEUCM.

1. The quickest is to press the "Reset" button on the power supply of the PMEUCM's rack.

WARTING: Cycling power on the rack will also restart every device in that rack which may result in injury or death.

2. Remove the mounting screw on the PMEUCM and remove it from the powered rack. Re-install the PMEUCM into the rack to cycle power on the entire device.

3. Remote resetting of the PMEUCM may be done through the DTM Browser. Right click on the PMEUCM entry in the DTM Broweser and select "Connect".

DTM Browse	er X	es DDT Types Function Blocks DEB Types	1
Host PC	168, 10, 1 > BMEP58_ECPU	T Name *	EDT UDDT VIODDT »
	Open	e 🗸	Type 👻 Adc 🔨
	Add		BOOL
	Delete Del		BOOL
	Delete Del	B FDR_SERVICE_B2	BOOL
	Field bus discovery	FDR_SERVICE_B3	BOOL
	Sort by address	FDR_SERVICE_B4	BOOL
Project Broy			BOOL
	Connect		BOOL
E Structura	Disconnect	Ree2	BYTE
Proj	Level data form de las	😓 🔶 UCM_Runtime_Status	WORD
📄 🛱 🖓 👘	Load data from device	🚗 🔶 UCM_Halt_Line_Number	UINT
₿ ₽ • ₹	Store data to device		UDINT
÷,	Сору		INT
	Dacte	- 10 ln_03	UINT TRAFF LICK 0000 L to 12" 4 -1
			I_PME_UCM_U2U2_pptest3in4out
	Device menu 🕨		
			UINT
	Properties ALT+Enter	BLOCK_STATUS	UINT
	Print device	SIGNATURE	UINT
		Reference PLC_FW_VERSION	UINT
	Zoom out	B PLC_STATE	BYTE
	France of all	PLC_START	BYTE
	Expand an	PLC_ITPE_ID	BTIE
	Collapse all		>
	>	ta Editor	
J			
× Linking			
		<i>(</i>)	<u>^</u>

If the connection is successful you will be able to select: Device Menu > Additional Functions > Reset Device.

		N/I NI A				
	Unity Pro XL : <ino name="">*</ino>					
<u>File Edit View Services To</u>	Eile Edit View Services Iools Build PLC Debug Window Help					
	○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○					
き ⊕ □ ⊟ Q, -						
DTM Browser	DTM Browser					
Host PC		Data Editor				
□ ⊕ 🕎 < 192.168.10.1 > BM	EP58_E Variables DDT Types	Function Blocks DFB Types				
the second secon	Open	Name = *	EDT DDT VIODDT »			
	Add	•	Type 👻 🖌 ۸			
	Delete Del		BYTE			
	Field bus discovery		BYTE			
<	Cost law address	STATUS	BYTE			
	Sort by address	1_LINK	BOOL			
Project Browser	Connect	F_IN_PROGRESS	BOOL			
P	Discount	INDANCY_OWNER	BOOL			
Ca Structural view	Disconnect	AL_STATUS	BOOL			
Project	Load data from device	ICE_STATUS	BYTE			
📄 👘 📩 Configuration	Change dista tra discission	_SERVICE	BOOL			
😑 🚽 📴 🖓 🛛 🖓 🔁	store data to device	SERVICE	BOOL			
ш III 0 : ВМЕ ХВР		SERVICE_B1	BOOL			
2 : EIO Bus	Deste	SERVICE_B2	BOOL			
📄 Derived Data Type	Paste	SERVICE_B3	BOOL			
Offline Parameter	Device menu	SERVICE_B4	BOOL			
Online Parameter	beneemens	PORT1_INFO	BYTE			
Online Parameter	Properties ALT+Enter	PORT1_FUNCTION_B1	BOOL			
Compare		PORT1_FUNCTION_B2	BOOL			
Configure	Print device		BTIE			
Configuration	7 .	Nuntime_Status	WORD			
Observe	Zoom out	Hatt_une_number	UINT			
Diagnosis	Expand all		INT			
	Callenae all		1111			
I ransfer to FDK Server Additional functions	Conapse an					
Reset Device Program	· · · · · · · · · · · · · · · · · · ·					
Help	Data Editor					
DTM message: PME_UCI ##WRONG_ADDRESS	M_0202_bptest3in4out roject λ Import/export λUse	er errors / FDT log event	∧ ∧ Search/Replace /			
Reset Device		HMI R/V	V mode EQUAL RUN UPLOAD INFO OK TCP			

Otherwise this windows will be shown:

	BMEP58_ECPU	×
A No Prop has	Ethernet/IP device at the specified address. Check Channel perties configuration screen and verify that the Source IP Addr been configured correctly. All online functions are disabled.	ress
		ок
	Unity Pro XL 🛛 🗙	
	Impossible to go in on-line mode.	
	ОК	

Cycling power on the PMEUCM is the most reliable method for forcing the PTK card to reboot.

9 Using Example1

LED Panel

Most of the LED indicators on the top panel are controlled by the user application. Example1 blinks many of these lights at different rates just for fun. Other LEDs are controlled by the PTK board or the UCM operating system.



PTK Board Controlled Lights

The top three lights are controlled by the PTK board. The meaning of these lights is described in the following table.

Label	Color		Description					
RUN	Green	ON – The PTK data across the NOTE: This is PLC.	board is properly configured and exchanging backplane with the M580 PLC. NOT an indication of the run/halt state of the					
ERR	Red	The blink patte backplane inter	The blink pattern of the ERR light indicates the state of the PTK packplane interface.					
		Blink Rate	Meaning					
		Steady OFF	If RUN light is ON – No errors present If RUN light is OFF – No backplane configuration					
		Steady ON	UCM has not started backplane operation or invalid MAC address or no Ethernet link					
		2 blinks ON	No IP Address					
		3 blinks ON	Invalid configuration					
		4 blinks ON	Duplicate IP Address					
		5 blinks ON	Awaiting served IP Address					
		6 blinks ON	IP Address invalid					
		7 blinks ON	Error on UCM board					
PWR	Green	The PMEUCM backplane when	has proper 24Vdc power from the Ethernet n lit.					

UCM OS Controlled Lights

The ETHLINK1 and ETHLINK2 lights are controlled by the UCM operating system.

Label	Color	Description
ETHLINK1	Green	ON – Link OK on port E1 OFF – NO Ethernet Link on port E1
ETHLINK2	Green	ON – Link OK on port E2 OFF – NO Ethernet Link on port E2

USER Controlled Lights

The LEDs labeled 1 through 6 on the top panel are under control of the running application. Lights 1, 2, 3, 4, and 6 are blinked at various timed rates as an example for controlling these lights. Light 5 is used to indicate the UCM's status of the communication with the M580 PLC.

NOTE: While the label on the top panel shows the number 1 through 6, the UCM programming language labels these lights as 3 through 8. This is because UCM lights 1 and 2 are the red LEDs behind the LCD to maintain compatibility with other Niobrara UCM products with these two LEDs behind the LCD (MUCM, QUCM, DUCM, and CUCM).

Panel Label	UCM Programming Reference	Color	Example1 Functional Description
1	3	Green	Toggles at 50mS rate
2	4	Yellow	Toggles at 100mS rate
3	5	Green	Toggles at 200mS rate
4	6	Yellow	Toggles at 400mS rate
5	7	Red	ON – Data comms with PLC BAD OFF – Data comms with PLC GOOD
6	8	Red	Toggles at 800mS rate

Example code for blinking lights

At the end of Thread 1 of the program Example1.ucm2 are several timers that expire to change on/off state of the blinking lights.



LCD and Joystick Operation

The front panel LCD provides status information about the PMEUCM and user interaction with the setup and operation of the card/application.

Most of the UCM code for controlling the screen and handling the joystick input is in Thread 2 of Example1.ucm2. In fact, most of the code for this program is the screen driver.

The information displayed on the "splash" screen varies depending on the configuration and state of the PTK board.



The text "ACTIVE" or "INACTIVE" describes the backplane operation. ACTIVE means that the UCM is exchanging data with the M580 while INACTIVE means that it is not.

PLC: RUN or STOP shows the UCMs understanding of the state of the M580 CPU.

NOTE: RUN is only possible if the connection is ACTIVE. STOP will always be displayed if the connection is INACTIVE. This description may not actually represent the RUN/STOP state of the CPU if the link is INACTIVE.

V1.02.007 shows the software version of the PTK board. This value may not be displayed if the UCM is unable to configure the PTK board. The most likely reason is that the PMEUCM is installed in an unconfigured slot of an active PLC system.

PVL:00.01 shows the PVL version of the SPI interface between the UCM board and the PTK board.

BP IP Address indicates the actual IP Address on the Ethernet backplane. In the above screen shots, the PTK board is at 192.168.10.3 or 0.0.0.0. If the PMEUCM is mounted in a powered slot of a rack without a PLC, the PTK board will revert to an IP address determined by its MAC address.

E1 IP Address shows the IP Address of the E1 port on the PMEUCM. By default it is 10.10.10.10.

E2 IP Address shows the IP Address of the E2 port on the PMEUCM. By default it is 10.10.10.11.

Backlight

The backlight time is controlled user code. In this case there is a timer that keeps the backlight on for 60000 mS when there is no activity of the joystick. At the end of this timer, the UCM code changes the screen back to the splash screen.

Menus

Moving the joystick will cause the application to show various menus to access status or setup screens. Move the highlighted cursor around with the joystick. Typically a right press will act as "Enter" while a left press will act as "Escape". Sometimes a push in "Enter" is needed (Factory Default for example).

Data Echo Config

The Example1 performs one of three different tasks:

- 1. Echo PLCOUT data to PLCIN data
- 2. Echo the inversion of PLCOUT to PLCIN
- 3. Allow the user to manually edit the PLCIN data from the front panel

The mode of operation is set in the Data menu screen by moving selecting the bottom entry and moving the joystick to the right to cycle through the options.



PLCOUT Data

The PLCOUT data may be vied by selecting the PLCOUT menu item. The radix of the displayed value may be changed between SIGNED, UNSIGNED and HEX by moving the joystick right while on the PLCOUT data screen.

In the following example, an animation table has been built to allow modification of the PLCOUT data to the UCM.

🗲 Table[Data Editor]			
Modification Force	<u>v</u> 5 v		> 国 列 🖡
Name 🔹	Value	Туре 👻	Comment
		T_PME_UCM_0	
🔶 Freshness	1	BOOL	Global Freshness
Freshness_1	1	BOOL	Freshness of Object
		T_PME_UCM_0	Input Variables
🖃 🗐 Outputs		T_PME_UCM_0	Output Variables
ASS_OUT_TAG	0.0	REAL	
BLOCK_LEN	0	UINT	
BLOCK_IDEN	0	UINT	
BLOCK_STAT	0	UINT	
SIGNATURE	0	UINT	
	0	UINT	
PLC_STATE	0	BYTE	
PLC_START	0	BYTE	
PLC_TYPE_ID	0	BYTE	
PLC_AB_ADD	0	BYTE	
DATA_FRES	0	BYTE	
🕀 🖬 📘 Free3		ARRAY[06] OF	Unused Variable
	-15373	INT	out_01
	123	INT	out_02
	-500	INT	out_03
	333	INT	out_04
	0	INT	out_05
- 🔶 out_06	12345	INT	out_06
🔷 🔶 out_07	9876	INT	out_07
b-			

Now, the PLCOUT data may be viewed on the PMEUCM screen:



PLCIN Data

The data returned to the M580 by the UCM depends upon the setting of the ECHO, INVERTED, or EDITABLE menu item on the Data menu.

ECHO – The data is copied directly so $In_01 = Out_01$ and $In_02 = Out_02$.

🗲 Table[Data Editor]			
Modification Force	고 고 그	- i - i - i - i - i - i - i - i - i - i	ジ 川 永 📘 📄
Name 👻	Value	Туре 💌	Comment
Free2	0	BYTE	Unused Variable
UCM_Runtime_S	22768	WORD	.15=Run; LSB=Runtime Error
UCM_Halt_Line	0	UINT	UCM Runtime Error Halt Line Number
🖌 🔶 🔤 01	-15373	INT	In_01
• In_02	123	INT	ln_02
- In_03	-500	INT	ln_03
In_04	333	INT	In_04
📄 🗇 Outputs 📃		T_PME_UCM_0	Output Variables
ASS_OUT_TAG	0.0	REAL	
BLOCK_LEN	0	UINT	
BLOCK_IDEN	0	UINT	
BLOCK_STAT	0	UINT	
SIGNATURE	0	UINT	
PLC_FW_VE	0	UINI	
	0	BYIE	
	0	BYIE	
	0	BTIE	
	0	BTIE	
			Libuard Variable
	-15272		out 01
	123	INT	out 02
	-500	INT	out 03
	333	INT	out 04
out 05	0	INT	out 05
• out 06	12345	INT	out 06
• out 07	9876	INT	out 07
· · · · ·		-	

Viewing the PLCIN screens:



🗲 Table[Data Editor]						
Modification Force	[] <u>E</u> <u></u>	N 5 3		> 14 名		
Name 🔻	Value	Туре	.	Comment		
Free2	0	BYTE		Unused Variable		
UCM_Runtime_S	32768	WORD		.15=Run; LSB=R	untime Error	
UCM_Halt_Line_	. U	UINT		UCM Runtime En	ror Halt Line Number	
• In_01	15373	INT		In_01		
In_02	-123	INT		In_02		
🗣 ln_03	500	INT		In_03		
📃 👘 🔶 ln_04	-333	INT		In_04		
🖻 🗐 Outputs		T_PME_	UCM_0	Output Variables		
ASS_OUT_TA	G 0.0	REAL				
BLOCK_LEN	. 0	UINT				
BLOCK_IDEN.	0	UINT				
BLOCK_STAT	0	UINT				
SIGNATURE	0	UINT				
PLC_FW_VE	. 0	UINT				
PLC_STATE	0	BYTE				
PLC_START	0	BYTE				
PLC_TYPE_ID	0	BYTE				
PLC_AB_ADD	0	BYTE				
DATA_FRES,.	. 0	BYTE				
Free3		ARRAY[0)6] OF	Unused Variable		
	-153/3	IN		out_01		
out_02	123	INT		out_02		
out_03	-500			out_03		
out_04	333	INT		out_04		
	U			CU_00		
	12345			out_06		
	30/0			001_07		

INVERTED – The data is copied and negated so $in_01 = -out_01$ and $in_02 = -out_02$.



EDITABLE – The PLCIN data is set from the front panel of the PMEUCM.



In this example, the value in register 502 is changed from 500 to 5034. Now the value in the M580 has changed to 5034 while the Out_03 stays at -500.

• Table[Data Editor]			
Modification Force	[군 고] 군	- 14 - 14 - 14 - 14 - 14 - 14 - 14 - 14	> 川 卯 🖥 🔟
Name -	Value	Туре 👻	Comment
Free2	0	BYTE	Unused Variable
UCM_Runtime_S	32768	WORD	.15=Run; LSB=Runtime Error
UCM_Halt_Line_	0	UINT	UCM Runtime Error Halt Line Numb
🔶 ln_01	15373	INT	In_01
🔶 In_02 📊	-123	INT	ln_02
In_03	5034	INT	In_03
🔜 🔶 ln_04	-333	INT	In_04
🚊 🗐 Outputs		T_PME_UCM_0	Output Variables
ASS_OUT_TA	G 0.0	REAL	
BLOCK_LEN	0	UINT	
BLOCK_IDEN.	0	UINT	
BLOCK_STAT.	0	UINT	
SIGNATURE	0	UINT	
PLC_FW_VE	. 0	UINT	
PLC_STATE	0	BYTE	
PLC_START	0	BYTE	
PLC_TYPE_ID	0	BYTE	
PLC_AB_ADD.	0	BYTE	
DATA_FRES	. 0	BYTE	
🕢 🖪 Free3		ARRAY[06] OF	Unused Variable
out_01	-15373	INT	out_01
out_02	123	INT	out_02
out_03	-500	INT	out_03
out_04	333	INT	out_04
out_05	0	INT	out_05
out_06	12345	INT	out_06
out_07	9876	INT	out_07
b			

PMEUCM DTM Manual

9 Using Example1 105

Modbus Registers

A close inspection of Thread 1 in Example1.ucm2 shows that the PLCOUT data is copied to output[200] through output[206]. These output registers may be accessed as Modbus Holding registers through the built-in Modbus/TCP server of the PMEUCM OS.

The PLCIN data is copied to output[500] through output[503].

The NRDTOOL program may be used to inspect/modify these Modbus registers.

NRDTOOL may be started by Start > Programs > Niobrara > NRDTOOL

ſ	😰 N	iobrara Desl	ktop Tool		×
	<u>F</u> ile	<u>W</u> indows	⊻iew		<u>H</u> elp
Open Connection					
Open Recent	>				
Close					
Preserve Desktop					
Restore Desktop					
Quit					

To open a new Modbus/TCP connection to the PMEUCM, Select File >Open Connection.

File Windows View Help Image: Connection Image: Connection Settings Modbus TCP Meximum Asyncronous Reads Host Image: Connection Settings Host Default Slave Address Z55 Starting Register Ox Registers Slave Starting Register Starting Register Variangesters Starting Register Starting Register Starting Register Disable multi-register writes Disable multi-register writes Disable ingle-register writes Do not process XML	🔞 Niobrara Desktop Tool	—		×
Image: Connection Settings Host Image: Connection Settings Maximum Asyncronous Reads 1 Max Timeout Image: Connection Settings Image: Connection Settings Image: Connection Settings Image: Connection Settings Slave Eddress Image: Connection Settings Image: Connection Settings Image: Connection Settings Connect	<u>File W</u> indows <u>V</u> iew			<u>H</u> elp
Image: Connection Modbus TCP Modbus RTU Modbus ASCII Sy/Max Host Maximum Asyncronous Reads Concet Maximum Asyncronous Reads Connect				
Modbus TCP Mabus ASCII Sy/Max Connection Settings Port 502 Host 1 Port 502 Maximum Asyncronous Reads 1 Maximum Asyncronous Reads 1 Max Timeout 5000 Default Slave Address 255 Register Editor Settings Government 1 Ox Registers Slave Address 255 C tx Registers Slave Address 255 C tx Registers Starting Register 1 • 4x Registers Read Count 0 • 4x Registers Read Count 0 • 50 not process XML Connect	Deep connection		×	
Connection Settings Port 502 Host 10,10,10,10 Maximum Asyncronous Reads 1 Max Timeout 500 Default Slave Address 255 Register Editor Settings 1 O x Registers Slave Fideress Ax Registers Slave Fideress X Registers Starting Register X Registers Read Count X Register 6x File Diable multi-register writes Diable single-register writes Diable single-register writes Connect	Modbus TCP Modbus RTU Modbus ASCII Sy/Max			
Maximum Asyncronous Reads Max Timeout 5000 Default Slave Address 255 Register Editor Settings Ox Registers Starting Register A Registers Starting Register Read Count Concut Connect Do not process XML Connect	Connection Settings Host 10.10.10.10			
Max Timeout 5000 Default Slave Address 255 Register Editor Settings 1 Ox Registers Slave Address 255 1 x Registers Slave Address 255 1 x Registers Starting Register 1 3 x Registers Starting Register 1 • 3 x Registers Read Count 0 • 4x Registers Read Count 0 • ownegister S 6x File 0 • Isable Register Editor 6x File 0 • Disable multi-register writes Disable single-register writes 0 • Do not process XML Connect	Maximum Asyncronous Reads 1			
Register Editor Settings Ox Registers Slave Indress 1 x Registers Starting Register 1 x Registers Starting Register 1 x Registers Starting Register 1 x Registers Read Count 0 * ox Registers * ox Register Editor Disable Register Editor Disable single-register writes Do not process XML Connect	Max Timeout 5000 Default Slave Address 255			
○ 0x Registers Slave iddress 255 ○ 1x Registers Starting Register 1 ○ 3x Register Read Count 0 ○ 4x Registers Read Count 0 ○ ox negister S 6x File 0 ○ bisable multi-register writes □ □ □ Do not process XML Connect	Register Editor Settings			
C 1x Registers Starting Register 1 Dx Register Read Count 0 C 4x Registers Read Count 0 C 4x Register 6x File 0 Image: Starting Register Writes 0 0 Image: Do not process XML Connect	C 0x Registers Slave Address 255			
SX Registers Read Count • 4x Registers Read Count • ox registers 6x File • Disable Register Editor 0 • Disable multi-register writes 0 • Disable single-register writes 0 • Do not process XML Connect	C 1x Registers Starting Register 1			
Image: Connect 6x File 0 Image: Connect 0 0 Image: Connect <td>Ax Registers Read Count</td> <td></td> <td></td> <td></td>	Ax Registers Read Count			
Disable multi-register writes Disable single-register writes Do not process XML Connect				
Do not process XML Connect	Disable multi-register writes			
Do not process XML Connect	Disable single register writes			
	Do not process XML Connect			

Select the Modbus TCP tab.

Enter the IP Address of E1 (10.10.10.10) for the Host.

Make sure that the Port is set to match the OS Port of the UCM (502 default).

Make sure that the radio button is set to 4x.

Make sure that the Slave Address is set to 255.

Now click "Connect"

The screen should now show the first block of 4x Holding Registers.

	Niobrara De	sktop Tool							—		×
<u>F</u> ile	<u>W</u> indows	<u>V</u> iew									<u>H</u> elp
)
	🚺 Modb	us TCP 10.1	10.10.10:502:	Register Ed	itor - Editing	255				×	
	Register	Editor									
	Modbus	TCP 10	.10.10.1	0:502 -	Editing	255					
	Aunnin 4x RE	g Norma GISTER	HEX	VALUE	SIGNED		BII	NARY			
		1	0	0	0	0000	_0000_0000_0	0000			
		3	0 0	0 0	0 0	0000	_0000_00000_(3000 3000			
		4	Ø	0	Ø	0000	_0000_0000_0	0000			
		5	Ю	0 0	0 0	0000	_0000_00000_(3000 3000			
		2	Ō	Ø	Ø	0000	_0000_0000_	0000			
		8	и 0	0 0	и 0	0000	_0000_00000_0	1000 1000			
		10	õ	õ	õ	0000	_0000_00000_0	0000			
		11	0	Ø	0	0000	_0000_0000_0	0000			
		13	Ø	Ø	0 0	0000	_0000_0000_0	3000			
		14	0	0	0	0000	_0000_0000_(0000			
		15 16	И	0 0	ы 10	0000	_0000_00000_0 0000_00000_0	1000 1000			
		17	õ	õ	õ	0000	_0000_00000_0	0000			
		18	0	Ø	0	0000	_0000_0000_0	0000			
		20	И	0 0	0 0	0000	_0000_00000_(0000 1000			
		21	Ø	Ø	ē	0000	_0000_0000_	0000			
		22	0 0	0 0	ы 10	0000	_0000_00000_0	4000 1000			
		24	õ	õ	õ	0000	_0000_00000_0	0000			
		25	0	0	0	0000	_0000_0000_0	0000			
		25	И	0 0	0 0	0000	_0000_00000_0	0000			
		28	õ	Ø	ŏ	0000	_0000_0000_	0000			
		29	0	0	0	0000	_0000_00000_(0000 2000			
		10	U	ย	ย	0000	_0000_00000_0	0000			
	-										

Typing in a value in the blue cursor enters the new value. Typing in 200 in the Register number will cause the viewer to jump to register 200.

👔 Niobrara Desktop Tool	- 0	×
<u>File W</u> indows <u>V</u> iew		<u>H</u> elp
Moibrara Desktop Tool File Windows View Image: Construct of the state o	255 BINARY 1100_0011_1111_0011 0000_0000_011_1011 1111_110_0000_01101 0000_0000_0111_001 1111_1110_0000_01101 0000_0000_0000_0000 0011_0000_0011_1001 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000	× Help
220 0 0 0 221 0 0 0 222 0 0 0 223 0 0 0 224 0 0 0	0000_0000_0000_0000 0000_0000_0000 0000_0000_0000 0000_0000_0000 0000_0000_0000 0000_0000_0000	
225 0 0 0 226 0 0 0 227 0 0 0 228 0 0 0 229 0 0 0	0000_0000_0000_0000 0000_0000_0000 0000_0000_0000 0000_0000_0000 0000_0000_0000 0000_0000_0000	

Notice that now the PLCOUT data is shown.

Type in the number 500 in the REGISTER column and the screen will jump to the location of the PLCIN data.

	Niobrara De	sktop Tool							—	×
<u>F</u> ile	<u>W</u> indows	<u>V</u> iew								<u>H</u> elp
Eile	Niobrara Des <u>W</u> indows Modbus Register Modbus Runn in 4× RE	ktop Tool View us TCP 10. Editor g Norma GISTER 501 502 503 504 505 503 504 505 503 504 505 503 504 505 509 510 511 512 513 514 515 516 517 518	10.10.10:502: 1.10.10.1 11y HEX 3c0d ff85 13aa feb3 0 0 0 0 0 0 0 0 0 0 0 0 0	Register Edi 0:502 - UALUE 15373 5034 65203 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	tor - Editing SIGNED 15373 -123 5034 -333 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	255 255 0011 1111 0000 1111 1111 0000 0000	1100 1111 0011 0000 0000 0000 0000 000	BINARY 0000 1101 1000 0101 1010 101 1011 001 0000 0000 0000 0000		× Help
		519 520 521 522 523 524	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0000 0000 0000 0000 0000 0000	_0000 _0000 _0000 _0000 _0000 _0000	_0000_0000 _0000_0000 _0000_0000 _0000_0000 _0000_0000 _0000_0000		
		525 526 527 528 529	0 0 0 0	0 0 0 0 0	0 0 0 0 0	0000 0000 0000 0000	0000 0000 0000 0000 0000	_0000_0000 _0000_0000 _0000_0000 _0000_0000 _0000_0000		
]

Notice that register 502 has the value 5034 that was entered from the front panel.

Because the DATA configuration is still set for EDITABLE, we can manually enter data in this Modbus viewer and see the new data show up in the M580.

Move the cursor into the SIGNED column on register 503 and change the -333 value to 1234 and press ENTER.

e	<u>W</u> indows	<u>V</u> iew							<u>H</u> elj
	_	-							_
ſ	Modbi	TCP 10	10 10 10 502	Register Ed	itor - Editing	255			
	w would us for 10.10.10.10.22: Register Editor - Editing 255								
	Register I	Editor							
	Modbus	TCP 10	0.10.10.1	.0:502 -	Editing	255			
	4x RE	g Norma Gister	HEX H	UALIE	SIGNED		BINARY		
		500	3cØd	15373	15373	0011_1100	_0000_1101		
		501	ff85	65413	-123	1111_1111	_1000_0101		
		502	13aa	5034	5034	0001_0011	_1010_1010		
		503 504	4az Ø	1234	1234	0000_0100	_1101_0010		
		505	õ	ดี	õ	0000_0000	0000 0000		
		506	Ø	Ø	Ø	0000_0000	_0000_0000		
		507	0	Ø	0	0000_0000	_0000_0000		
		508	N	N	N	0000_0000	-0000-0000		
		510	6	0 0	6	0000_0000	_0000_00000		
		511	õ	õ	õ	0000 0000	0000 0000		
		512	Ø	Ø	Ø	0000_0000	_0000_0000		
		513	Ø	0	0	0000_0000	_0000_0000		
		514 C1C	0	0	0	0000_0000	0000_0000		
		516	ñ	ดี	Й	0000_0000	_0000_0000		
		517	õ	õ	Õ	0000_0000	_0000_0000		
		518	0	Ø	Ø	0000_0000	_0000_0000		
		519	0	Ø	0	0000_0000	_0000_0000		
		520 591	0	9 0	0 0	0000_0000	_0000_00000		
		522	ă	ดี	й	0000_0000	_0000_0000		
		523	Ō	ø	Ō	0000_0000	_0000_0000		
		524	0	Ø	Ø	0000_0000	_0000_0000		
		525	0	Ø	0	0000_0000	_0000_0000		
		526	0	0 0	0 0	0000_0000	_0000_00000		
		528	ă	ดี	й	0000_0000	0000_0000		
		529	Ø	0	0	0000_0000	_0000_0000		
L									

Now look in Unity Pro and see that In_04 has changed to 1234.

Table[Data Editor]							
Modification Force	2 <u>7</u> <u>7</u>		メ 川 永 🖪 📄				
Name 🔹	Value	Туре 💌	Comment				
Free2	0	BYTE	Unused Variable				
UCM_Runtime_S	32768	WORD	.15=Run; LSB=Runtime Error				
UCM_Halt_Line	0	UINT	UCM Runtime Error Halt Line Number				
h_ 01	15373	INT	In_01				
h_02	-123	INT	ln_02				
🔶 In_03	5034	INT	In_03				
In_04	1234	INT	In_04				
🖻 🗇 🗇 Outputs		T_PME_UCM_0	Output Variables				
ASS_OUT_TAG	0.0	REAL					
BLOCK_LEN	0	UINT					
BLOCK_IDEN	0	UINT					
BLOCK_STAT	0	UINT					
SIGNATURE	0	UINT					
PLC_FW_VE	0	UINT					
PLC_STATE	0	BYTE					
PLC_START	0	BYTE					
PLC_TYPE_ID	0	BYTE					
PLC_AB_ADD	0	BYTE					
DATA_FRES	0	BYTE					
		ARRAY[06] OF	Unused Variable				
out_01	-15373	INT	out_01				
out_02	123	INT	out_02				
• out_03	-500	INT	out_03				
out_04	333	INI	out_04				
• out_05	0	INI	out_05				
• out_06	12345	INI	out_06				
	9876	INT	out_0/				
······ •							

PMEUCM DTM Manual

Modbus 6x Files

NRDTOOL may be used to directly inspect the data in the Modbus 6x files used for passing data between the PMEUCM and the PTK board.

Inspecting Thread 1 of Examle1.ucm2 shows this line:

read file 58201 offset 0 BPOutputs { Copy outputs from backplane block 1}

This instruction copies the PLCOUT data from backplane block 1 located in 6x file 58201 into the array of bytes variable BPOutputs. The offset 0 means to start the file read at byte offset 0 (the beginning of the file).

Note: File 58201 is used because this is the file specified by the DTM. Looking at the DTM Utility, we see file 58201 listed in the Output assembly.



Before we look at the 6x file, change the starting register back to 200 to see the PLCOUT data:

🔣 Niobrara Desktop Tool — 🗆 🗙							
<u>F</u> ile <u>W</u> indows <u>V</u> iew						<u>H</u> elp	
Modbus TCP 10 Register Editor Modbus TCP 11 Running Norm. 4x REGISTER 200 201 202 203 204 205 206 207 208 209 209 210 211 212 213 214 215	10.10.10:502: Register 0.10.10.10:502 ally HEX UALU c3f3 5016 7b 12 fe0c 6503 14d 3039 1234 2694 987 0 0 0 0 0 0 0 0 0 0 0 0 0	Editor - Editing E SIGNED 3 -15373 3 123 6 -500 6 3333 6 2345 6 9876 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	255 255 255 BINARY 1100_0011_1111_0011 0000_0000_0111_001 0000_0000_0011_000 0000_0001_0100_01101 0000_0001_0100_01100 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000 0000_0000_0000_0000		~		

PMEUCM DTM Manual
Now Open a new connection in NRDTOOL by selecting File >Open connection.

👔 Niobrara Desktop Tool — 🗆 🗙								
<u>File Windows</u> <u>View</u>			<u>H</u> elp					
U Open connection								
Re Modbus TCP Modbus RTU Modbus ASCII Sy/Max Hor Connection Settings Host 10.10.10.10 Port								
Maximum Asyncronous Reads 1 Max Timeout 5000 Default Slave Address 255 Register Editor Settings Cox Registers Slave Address 255 1 X.Registers Slave Address 255 2 X.Registers Starting Register 1 3 X.Registers Read Count 0 1 M.Kegisters Read Count 0 4 crusser kegister 6x File 58201 1 Disable multi-register writes 1 1								
Do not process XML Connect								

Select 6x Registers then enter 58201 in the 6x File and press "Connect"

Move the cursor up to register 0.

	Niobrar	a Desktop Too	I						-		×
<u>F</u> ile	<u>W</u> ind	ows <u>V</u> iew									<u>H</u> elp
	И М	odbus TCP 10.	10.10.10:502: R	egister Editor -	Editing 2	55				8	
	Regi	ster Editor									
	Mod	bus TCP 10	.10.10.10	:502 - Ed:	iting 2	55				~	
	4x	REGISTER	HEX	VALUE SI	GNED		BIN	RY			
		200	c3f3 7b	50163 -1 123	123	1100_00		911 911			
		202	14d	333	333	0000_00	1_0100_1	100 101			
		204	3039	12345 1	2345	0000_000 0011_000	00_0000_0	900 301			
		205	2674	9876 Ø	9876	0010_01 0000_00	0_0000_0	100 100			
		208	0	0	0	0000 <u>0</u> 000	00_0000_0	300 300			
		210	<u></u>	<u>ل</u> ا	U	0000_000	00_0000_0	000			
		Modbus TCP 10.10.10.10:502: Register Editor - Editing 255							×=		
		Register Editor	 P 10 10 10	10-502 -	Editir	w 955 f	110 59201				
L.		Running N	ormally TED UEY	.10.302 IIALIIE	SIGN	19 200 I 70	110 30201	BINGRU			<u>^</u>
			f3c3	62403	-313		_0011_110	0_0011			
		2	cfe	3326	332	6 0000		1_1110			
		4	-1001 0 2020	14649	146	0 0000		0_0000			
		6	9426	37926	-2761	0 1001		0_0110			
		8	0	6		0 0000	_0000_000	0_0000			
		10	0	2 2		0 0000	-0000-000	0_0000			
		11 12	0	6		0 0000	_0000_000	0_0000			
		13	0			0 000	_0000_000	เด_ดดดด		_	~

Look at the data in HEX and notice that the data is byte swapped. This is

because the data in the 6x file is straight from the Ethernet/IP data handed to the UCM by the PTK board. This explains the code in Thread 1 that swaps the byte order of the data as it copies to/from the OUTPUT[] registers.

We can look at the PLCINPUT data 6x file as well.

R N	liobrara De	sktop Tool					—		×
<u>F</u> ile	<u>W</u> indows	<u>V</u> iew							<u>H</u> elp
	Modbu Register E Modbus Running 4x RE	TCP 10.10 ditor TCP 10.10 Gittor 9 Normal GISTER 500 501 502 503 504 505 504 505 506 507 508 509 509 509 509 509 509 511 512	0.10.10:502: R 10.10.10 1y HEX 3c0d ff85 ff85 f13aa 4d2 0 0 0 0 0 0 0 0 0 0 0 0 0	egister Editor : 502 - Ed UALUE S 15373 65413 5034 1234 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	- Editing liting IGNED 15373 -1234 5034 1234 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	255 255 BINARY 0011_1100_0000_1101 1111_1100_0000_101 1001_0011_1010_1010		*	3
	6×	514 515 REGIST 1 2 3 4 5 6 7 8 9 10 11 12 13	0 0 ER HEX f3c3 7b00 cfe 4d01 3930 3930 9426 9426 0 0 0 0 0 0 0 0	0 0 0 3148 332 1971 1464 3792	0 0 E SIG 3 -3: 8 31: 6 3: 3 19 0 14: 6 -27: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	ODD ODD <thodd< th=""> <thodd< th=""> <thodd< th=""></thodd<></thodd<></thodd<>		~	~

First change the start register from 200 back to 500.

Inspecting Thread 1 again shows the following line of code:

write file 58100 offset 4 BPInputs { Copy inputs to backplane block 0}

This procedure copies the data from the byte array BPInputs to 6x file 58100. Again looking back at the DTM shows why file 58100 is chosen.



Also notice that the first 4 bytes of this file are the UCM Runtime Status and Halt Line Number. The UCM operating system automatically fills in these two values so the UCM application must start its data with an offset of 4 bytes as shown in the WRITE FILE line of code above.

Open a new connection in NRDTOOL with 6x file 58100.

📊 Niobrara Desktop Tool		×
<u>File W</u> indows <u>V</u> iew		<u>H</u> elp
File Windows View Image: Modbus TCP 10:10:10:502: Register Editor - Editing 255 Image: Modbus TCP Modbus RTU Modbus TCP Modbus ASCII Image: Modbus TCP Modbus RTU Modbus TCP Modbus ASCII Image: Modbus TCP Maximum Asyncronous Reads Image: Modbus TCP Maximum Asyncronous Reads 1 Image: Modbus TCP Maximum Asyncronous Reads 1 Image: Modbus TCP Maxing Registers 1 I		Help

Again notice that the hex values are byte swapped because of the Ethernet/IP data

PMEUCM DTM Manual

structures. Also notice that the data starting in register 500 is in register 2 of the 6x file because of the 4 byte offset of the write file command.

Using the 6x file register viewer can be very handy when byte aligning the data in a UCM application.

NOTE: Accessing the 6x files associated with the DIO interface has a big impact on the performance of the backplane. These registers should only be accessed during troubleshooting and not normal operation.