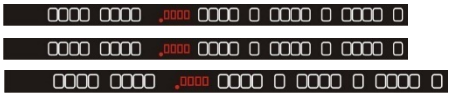
The GeistPDU Driver



The GeistPDU driver connects to a network of up to 30 Geist remotely monitored power distribution units (PDUs). Available for Commander and ObSys.

This document relates to GeistPDU driver version 1.0 to 1.1

Please read the *Commander Manual* or *ObSys Manual* alongside this document, available from *www.northbt.com*

Contents

Compatibility with the GeistPDU System 3
- Equipment 3
- Values 3
- Prerequisites 3

Using the Driver 4
- Starting the Interface 4
- Setting up the Driver 4
- Checking Communications 4

Object Specifications 5
- Example Object Reference 5
- Device Top-Level Objects 5
- PDU Module Setup 6
- Unit Details 7
- PDU System 8
- Parameters 9
- Unit 10
- PDU Information 11
- Sensor Value 12
- Current Value 12

Driver Versions 13

Compatibility with the GeistPDU System

The GeistPDU driver allows North to interface with a network of up to 30 Geist remotely monitored power distribution units (PDUs).

The driver connects to the Geist PDUs via a TCP/IP network (Fig. 1).

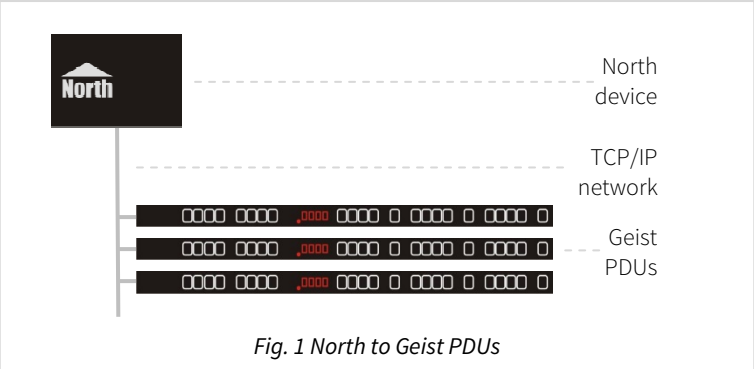


Fig. 1 North to Geist PDUs

Equipment

The driver is compatible with Geist PDUs with an XML data feed, including:

- Monitored power PDUs – RCX, RSM, and RCO
- Switched power PDUs – RCM-O and RCU-O
- Satellite power PDUs – RSS and RSE
- Current monitoring PDUs - EM8 and EM40

Values

Up to 64 values per PDU are available. Depending on the model of PDU, these can typically include:

- Energy (kWh)
- Volts (V)
- Current (A)
- Apparent power (VA)

In addition, the driver can calculate the following values for current monitoring PDUs (EM8 and EM40):

- Estimated Power
- Estimated Energy

Prerequisites

Set the IP address of each Geist PDU. If authentication is enabled, then the ‘Account Username’ and ‘Account Password’ must be set in the driver later.

You can access the configuration settings from the PDUs web page, the default IP address is 192.168.123.1.

If you are connecting via a firewall, then the driver will require access to TCP port 80 on the PDU.

Using the Driver

On ObSys and Commander, the GeistPDU driver is pre-installed. On all of these North devices, you can use the driver to create an interface to a network of Geist PDUs. Once started, you will need to set up the driver before it can communicate with the GeistPDU system.

Starting the Interface

💻 To start an interface using the GeistPDU driver, follow these steps:

- → **Start Engineering** your North device using ObSys
- → Navigate to **Configuration, Interfaces,** and set an unused **Interface** to 'GeistPDU' to start the particular interface
- → Navigate to the top-level of your North device and re-scan it

The driver setup object (Mc), labelled **PDU Setup**, should now be available. If this object is not available, check an interface licence is available and the driver is installed.

Setting up the Driver

💻 To set up the driver, follow these steps:

- → Navigate to the **PDU Setup** object (Mc). For example, if you started interface 1 with the driver earlier, then the object reference will be 'M1'
- → Navigate to **Unit 1 Details** object and set the **IP Address** and **Port Number** for the PDU
- → If basic authentication has been enabled in the PDU, set the **Account Username** and **Account Password**
- → Repeat this configuration for each unit

Checking Communications

You can check that the interface is communicating by reading the **Device Communicating** object (DS). A value of 'Yes' indicates the driver has connected to, and is communicating with the PDU system.

Object Specifications

Once an interface is started, one or more extra objects become available within the top-level object of the device. As with all North objects, each of these extra objects may contain sub-objects, (and each of these may contain sub-objects, and so on) – the whole object structure being a multi-layer hierarchy. It is possible to navigate around the objects using the ObSys Engineering Software.

Each object is specified below, along with its sub-objects.

Example Object Reference

An example of a reference to an object in the same device: the PDU System (S1) contains a Geist Unit (U1), which contains a Volts field (F4), which contains a Value (C) reading. Therefore, the object reference will be 'S1.U1.F4.V'.

An example of a reference to an object in a different device: the IP network object (IP) contains Default Commander object (CDIP), which contains the object above (S1.U1.F4.V) – therefore the complete object reference is 'IP.CDIP.S1.U1.F1.V'

Device Top-Level Objects

When an interface is started using the GeistPDU driver, the objects below become available within the top-level object of the device. For example, if Interface 1 is started, then the object with references 'M1' and 'S1' become available.

Description	Reference	Type
PDU Setup Set up the GeistPDU driver, started on interface *c* (*c* is the interface number)	M*c*	Fixed Container: On the Commander platform this will be *[CDM v20\GeistPDU v11]* On the ObSys platform this will be *[OSM v20\GeistPDU v11]*
PDU System Access Geist PDUs connected to interface *c* (*c* is the interface number)	S*c*	Variable Container: *[GeistPDU v11]*

PDU Module Setup

Object Type: *[OSM v20\GeistPDU v11]*
Object Type: *[CDM v20\GeistPDU v11]*
Object Type: *[OSM v20\GeistPDU v10]*
Object Type: *[CDM v20\GeistPDU v10]*

The PDU Module Setup contains the following objects:

Description	Reference	Type
Device Label	DL	Obj\Text; Max.20 chars; Adjustable
Device Communicating Shows if communication with the Geist system has been established	DS	Obj\NoYes
Poll Rate (min) The driver sets a timer to the Poll Rate when starting to collect data from the connected units. Once the driver has finished collecting data, it will wait for the timer to lapse before collecting data again	PR	Obj\Num; Range: 1…60; Adjustable
Clear Collected Data Write 'Yes' to this object to clear the data collected from the Geist PDUs, and then re-initialize communications	RST	Obj\NoYes; Adjustable
Unit *x* Details Add a PDU to the driver. The unit number, *x*, is in the range 1…30	U*x*	Fixed container: *[OSM v20\GeistPDU v11\Unit]* *[CDM v20\GeistPDU v11\Unit]*

Unit Details

Object Type: *[OSM v20\GeistPDU v11\Unit]*
Object Type: *[CDM v20\GeistPDU v11\Unit]*
Object Type: *[OSM v20\GeistPDU v10\Unit]*
Object Type: *[CDM v20\GeistPDU v10\Unit]*

The Unit Details contains configuration options to add a PDU to the driver.

Description	Reference	Type
IP Address	IA	Obj\IP; Adjustable
Port Number	PN	Obj\Num; Range: 1…65535; Adjustable Default value 80
Account Username Only required if Basic Authentication is enabled within the PDU	UN	Obj\Text; Max 25 chars; Adjustable
Account Password Only required if Basic Authentication is enabled within the PDU	PSW	Obj\Text; Max 25 chars; Adjustable

PDU System

Object Type: *[GeistPDU v11]*
Object Type: *[GeistPDU v10]*

A PDU system is a network of Geist monitored power distribution units. The IP address of each PDU must be configured in the *PDU Module Setup* first.

Description	Reference	Type
Parameters Configure the Power Factor and Voltage for the system	P	Fixed container: *[GeistPDU v11\Params]*
Unit *x* The unit address, *x*, can be in the range 1…30	U*x*	Variable container: *[GeistPDU v11\Unit]*

Parameters

Object Type: *[GeistPDU v11\Params]*
Object Type: *[GeistPDU v10\Params]*

The Parameters object is used to configure a power factor and system voltage. These values are used to provide an estimated power and energy value for EM8 and EM40 PDUs.

Description	Reference	Type
Power Factor	PF	Obj\Float: 0…1; Adjustable Default value 0.90
Voltage (V)	V	Obj\Float: 0…480; Adjustable Default value 230

Unit

Object Type: *[GeistPDU v11\Unit]*
Object Type: *[GeistPDU v10\Unit]*

The driver can store up to 64 data fields per PDU. The objects available here are dependent on the PDU model – scan the unit to find the data fields available.

Description	Reference	Type
PDU Information	I	Fixed container: *[GeistPDU v11\Info]*
Sensor value label The data field index, *a*, is in the range 1…64	F*a*	Fixed container: *[GeistPDU v11\Value]*
Current Transformer *x* The power outlet number, x, is dependent on the PDU	CT*x*	Fixed container: *[GeistPDU v11\Field]*
Circuit *x* The circuit number, x, is in the range 1…6	CI*x*	Fixed container: *[GeistPDU v11\Field]*
Phase *y* The phase letter, y, can be A, B, or C	PH*y*	Fixed container: *[GeistPDU v11\Field]*
Neutral	N	Fixed container: *[GeistPDU v11\Field]*
Total	TOT	Fixed container: *[GeistPDU v11\Field]*

PDU Information

Object Type: *[GeistPDU v11\Info]*
Object Type: *[GeistPDU v10\Info]*

A PDU Information object contains information about the PDU:

Description	Reference	Type
Label PDU description	L	Obj\Text; Max 14 chars.
Online Indicates if PDU is currently responding	S	Obj\NoYes
Last read Time data last retrieved from the PDU	DT	Obj\DateTime
Data fields available Number of data fields retrieved from the PDU	C	Obj\Num: 0…64

Sensor Value

Object Type: *[GeistPDU v11\Value]*

A PDU sensor value contains the following objects.

Description	Reference	Type
Label	L	Obj\Text; Max 20 chars.
Value	V	Obj\Float
Maximum Value Maximum valid range of sensor value	VH	Obj\Num
Minimum Value Minimum valid range of sensor value	VL	Obj\Num

Current Value

Object Type: *[GeistPDU v11\Field]*
Object Type: *[GeistPDU v10\Field]*

A PDU current value reads the electrical current for the monitored power outlets of an EM8 or EM40, and calculates the approximate power and energy used. Configure the supply voltage and power factor within the *Parameters* object.

Description	Reference	Type
Label	L	Obj\Text; Max 20 chars.
Current (A)	C	Obj\Float
Maximum Current (A) Maximum valid range of current	CH	Obj\Num
Minimum Current (A) Minimum valid range of current	CL	Obj\Num
Warning Limit (A) Warning threshold value for current	CW	Obj\Num
Alarm Limit (A) Alarm threshold value for current	CA	Obj\Num
Alarm State	S	Obj\ENum; Range:0…2 Values: 0=OK, 1=Warning, 2=Alarm
Estimated Power (W)	P	Obj\Num
Estimated Power (kW)	PK	Obj\Float
Estimated Energy (Wh)	E	Obj\Num; Adjustable To reset the energy, set the value to ‘0’
Estimated Energy (kWh)	EK	Obj\Num; Adjustable To reset the energy, set the value to ‘0’

Driver Versions

Version	Build Date	Details
1.0	24/11/2011	Driver released
1.1	04/06/2014	Added support for more PDU types – previously only supported EM8 and EM40 Increased number of values per PDU from 32 to 64 Reduced number of PDUs from 55 to 30

Next Steps…

If you require help, contact support on 01273 694422 or visit *www.northbt.com/support*

North Building Technologies Ltd
+44 (0) 1273 694422
support@northbt.com
www.northbt.com

Author: BS
Checked by: JF

Document issued 16/07/2015.