DMPU-PS

USER MANUAL

Version

rev. 0.4.

Contents

Introduction

Foreword

DMPU is a modular electronic motor protection relay that provides protection, monitoring and metering functions for 3-phase, constant or dual speed, AC induction motors. The modular housing is for DIN-rail mounting with IP20 protection degree. The device, in its basic configuration, is able to measure the electrical motor variables (current, voltage, harmonic distortion, etc.), to monitor the thermal image of the motor, and also its load, operational status (start-stop, star-delta starting, 2 speeds, alarm set-point and other functions adjustable by the user), motor temperature and includes an event datalogger. The current measurement is carried out by means of 3 external current transformers, or by built-in split-core transformers up to 5A. With serial communication, it is possible to gather all the relevant instantaneous values and transmit them to a host control system for data collection and process control. Profibus and Modbus TCP/IP protocols are available for a high connectivity to the most used fieldbus systems. Through the optional remote operator interface (for panel mounting) it is possible to see the instantaneous values and status. The whole programming of the unit is to be performed via configuration software. Optional modules allow the collection of additional PTC and PT100 values for coils and bearing temperature control, and additional input/outputs are used for some local on-board logic functions.

Product inspection

Please check the followings when receiveng and unpacking DMPU:

The modules of DMPU are the ones specified in your purchase order.

- Check if there are damages caused by transportation. Please do not install the product, and contact Carlo Gavazzi sales representatives if there is any problem.
- We suggest to keep the original packing in case it is necessary to return the instrument to our After Sales Department. In order to achieve the best results with your instrument, we recommend to read this instruction manual carefully. If the instrument is used in a way not specified by the producer, the protection provided by the instrument may be impaired.

Precautions

For your safety, the following symbol is to remind you to pay attention to safety instructions on configuring and installing DMPU. Be sure to follow the instructions for higher safety.

This symbol indicates a particularly important subject or information

Safety precautions

General Information

Please read this manual thoroughly before using the device. Should there be any problem using the product which cannot be solved with the information provided in the manual, contact your nearest Carlo Gavazzi distributor or our sales representatives to help you.

Check that the device is installed in accordance with the procedures as described in this manual.

The manufacturer accepts no liability for any consequence resulting from inappropriate, negligent or incorrect installation or adjustment of the optional parameters of the equipment. The contents of this guide are believed to be correct at the time of printing. In the interests of commitment to a policy of continuous development and improvement, the manufacturer reserves the right to change the specification of the product or its performance, or the content of the guide without notice.

The device is only for qualified personnel who takes responsibility for the use. For your safety pay attention to safety instructions on handling, installing, operating, and checking the device.

Installation

Ambient environment

The environment will directly affect the proper operation and the life span of the device, so install the device in an environment complying with the following conditions:

- Ambient temperature: -25°C ~ +55°C (-13°F ~ +131°F)
- Avoid exposure to rain or moisture
- Avoid direct sunlight
- Avoid oil mist and salinity
- Avoid erosive liquid and gas
- Keep away from radioactive and flammable materials
- Avoid electromagnetic interference
- Avoid vibrations

All the modules have IP20 protection degree. Don't place the devices in an environment where they can be damaged electrically or mechanically.

Mounting

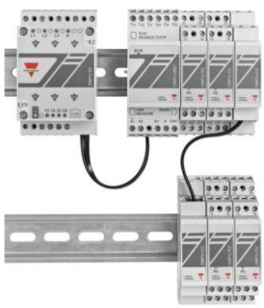
Mounting description

DIN-rail mounting

Mount the modules on the DIN-rail (before placing the modules on the DIN rail move down the plastic tab on the back of the module and secure the modules to the DIN rail by repositioning the tab up); connect them (except DMPU-05) side by side according to the order defined in the configuration of the device decided for your application (see the software configuration chapter).

The available modules are the following:

- DMPU-MBT or DMPU-PRB (base unit Modbus TCP/IP or Profibus)
- DMPU-05 (measurement unit)
- DMPU-R2 (2I/2O unit)
- DMPU-CC (bus extension unit)

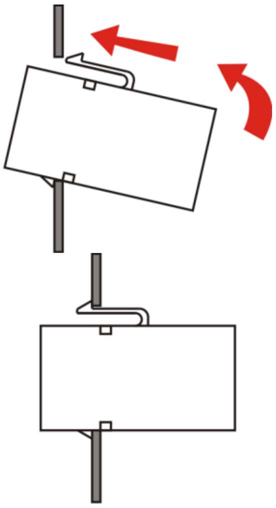


The first module mounted side by side on the left must be DMPU-MBT or DMPU-PRB (main module). Side by side mounting allows to communicate among and to supply the other modules from the main one. Use DMPU-R2 modules to increase the available inputs/outputs and the internal bus adaptor DMPU-CC if more than one DIN-rail is needed. The maximum connectable modules are:

- 1 DMPU-MBT or DMPU-PRB
- 1 DMPU-05
- 10 expansion modules (DMPU-R2)

If the order of the modules is changed after the definition of the software

Mounting example



DMPU-CPAN mounting

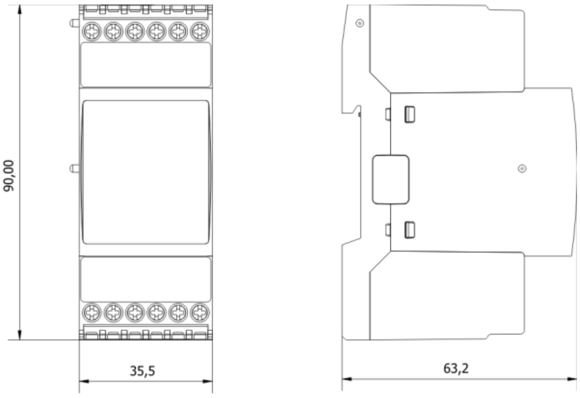
configuration, check the congruency between the modules mounting and the software settings; in case of incongruity the application may not work properly.

Connection and disconnection of the modules must be done when the system is OFF.

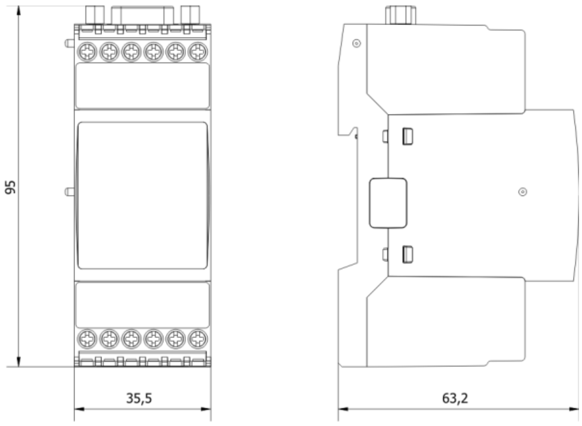
If DMPU-HMI display is used, mount it on the front of the panel.

Dimensions

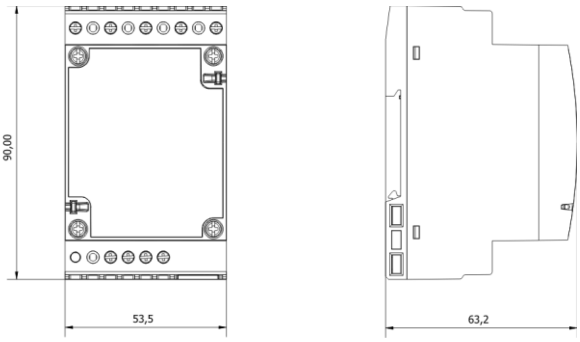
DMPU-MBT



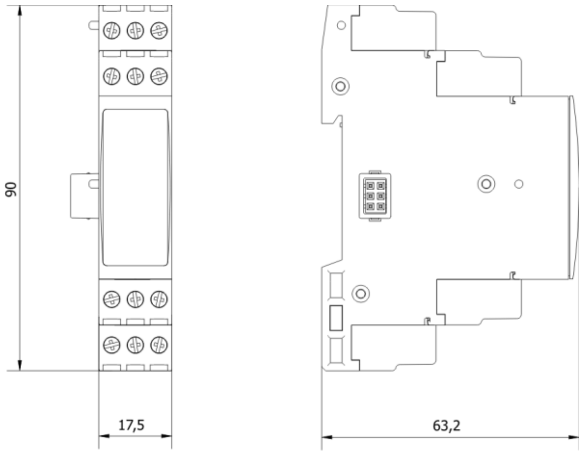
DMPU-PRB



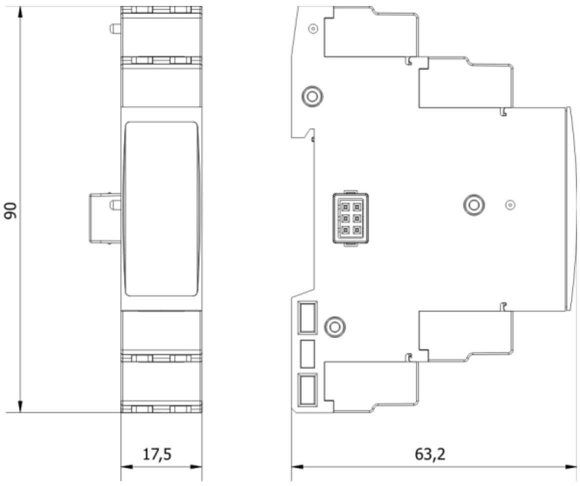
DMPU-05



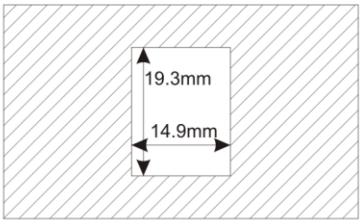
DMPU-R2



DMPU-CC



DMPU-CPAN panel cut-out



Wiring

Connect the wires to the terminals taking care of the correct torque:

Terminal torque

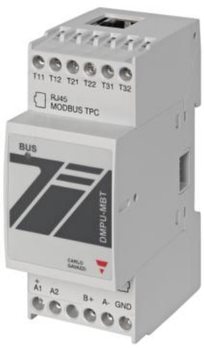
Module	Screwdriver torque
DMPU-MBT and DMPU-PRB	0.4Nm/0.8Nm
DMPU-05	0.4Nm/0.8Nm
DMPU-R2	0.4Nm/0.8Nm

Use cables with the following section area:

Cable section area

Module	Max cable section area	
DMPU-MBT and DMPU-PRB	power supply	2 x 1.5 mm^2
	inputs	6 x 1.5 mm^2
	RS485 communication	3 x 1.5 mm^2
DMPU-05	outputs	4 x 1.5 mm^2
	voltage inputs	4 x 1.5 mm^2
	currents	ø 9mm
DMPU-R2	input and output	4 x 1.5 mm^2

DMPU-MBT and DMPU-PRB



DMPU-MBT

Connect power supply voltage according to the scheme taking care of the polarity.
Connect max 3 temperature sensors (2-wire PT100 or PTC) or 3 digital inputs or a mix of them.

If DMPU-HMI display is used, connect the serial port of the display to the RS485 port. It is the same to use the screw or the RJ11 terminals.

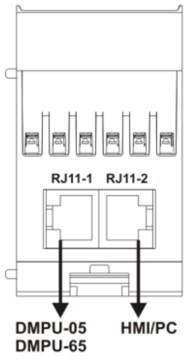
⚠ To reduce the noise on the RS485 communication cable use a shielded cable and connect the shield to GND terminal and to the ground (one point only).

⚠ While programming DMPU-MBT or DMPU-PRB with DMPU-PS software (using RS485 port), DMPU-HMI display must be disabled (see the display instructions to enable this mode), otherwise the connection may not work properly (it's not possible to have two master devices in the same bus).

On the bottom of the module there are two RJ11 connectors for DMPU-05 measurement module (on the left) and the DMPU-CPAN or DMPU-CPC cable (on the right); these cables are used for PC connection and the configuration by programming software.

On the top of the module there is an auxiliary communication port which depends on the main module type:

- on DMPU-MBT there is a RJ45 connector for Modbus TCP/IP communication
- on DMPU-PRB there is a DB9 connector for Profibus communication



Bottom view of DMPU-MBT and DMPU-PRB

DMPU-05

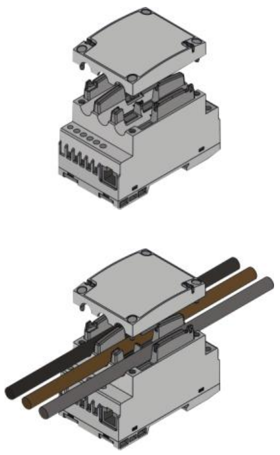
These modules are connected to DMPU-PRB or DMPU-MBT through the dedicated cable with RJ11 connector on the bottom side of the module.

DMPU-05

Connect the measurement module when the device is not powered.

It's possible to use only one measurement module (DMPU-05) for each main module.

DMPU-05 doesn't require auxiliary power supply, being provided via RJ11 connector from the main module.



Removing the cover for current measurement

To measure the currents remove the 4 screws on the front to open the cover of DMPU-05 and place the 3 motor cables or the current transformer cables in the holes to allow the current measurement through the 3 phase split-core current transformers (the hole diameter for the wire is 9 mm) taking care of the sequence and the current directions. Close the cover of DMPU-05 with a screwdriver avoiding damages to the cable. Connect the voltages (directly or through voltage transformers) to the terminals including neutral if available. Voltage connection isn't mandatory but it is recommended to increase the accuracy of the product.

Pay attention to the reference of the voltages and currents: the voltage terminal L1 refers to the current hole I1, the voltage terminal L2 refers to the current hole I2, the voltage terminal L3 refers to the current hole I3. Observe the currents directions according to the arrows reported on the front label (the arrowhead indicates the motor position).

Connect up to 2 relay outputs.

The output present on the measurement module have an intrinsic delay < 100ms. The ones on the DMPU-R2 modules < 1000ms. Use the former ones for the most time critical tasks (e.g.: start/stop, star delta, …).

The suggested current transformers for the most common motor rating and power supply voltage are the following:

Current transformer				
Motor power [kW]	Items			
	@230V	@400V	@480V	@600V
1.5	CTD3X1505A	-	-	-
2.2	CTD3X1505A	-	-	-
3.7	CTD3X1505A	CTD3X1505A	CTD3X1505A	-
5.5	CTD3X1505A	CTD3X1505A	CTD3X1505A	CTD3X1505A
7.5	CTD3X2005A	CTD3X1505A	CTD3X1505A	CTD3X1505A
11	CTD3X2505A	CTD3X1505A	CTD3X1505A	CTD3X1505A
15	CTD3X4005A	CTD3X2005A	CTD3X1505A	CTD3X1505A
18.5	CTD3X5005A	CTD3X2505A	CTD3X2005A	CTD3X1505A
22	CTD3X6005A	CTD3X3005A	CTD3X2505A	CTD3X2005A
30	CTD3X7005A	CTD3X4005A	CTD3X3005A	CTD3X2505A
37	CTD3X10005A	CTD3X5005A	CTD3X4005A	CTD3X3005A
45	CTD3X12005A	CTD3X6005A	CTD3X5005A	CTD3X4005A
55	CTD4X15005A	CTD3X7005A	CTD3X6005A	CTD3X5005A
75	CTD8V20005A	CTD3X10005A	CTD3X7505A	CTD3X6005A
90	CTD8V25005A	CTD4X15005A	CTD3X10005A	CTD3X7505A
110	CTD8V30005A	CTD4X16005A	CTD4X15005A	CTD3X10005A

DMPU-R2

This module doesn't require auxiliary power supply, which is via internal bus from main module.

Connect it when the device is not powered.

Connect up to 2 temperature sensors (2 or 3 wires PT100 or PTC) or 2 digital inputs or a mix of them.

Connect up to 2 relay outputs.

DMPU-R2

DMPU-CPAN or DMPU-CPC

DMPU-CPAN and DMPU-CPC cables are used for communication with the PC:

- Use DMPU-CPC cable to connect DMPU-MBT or DMPU-PRB directly to the PC RS485 port. If this port is not available on your PC please use an adapter.
- Use DMPU-CPAN cable to relocate the RJ11 socket from the main module to the panel and connect the PC RS 485 port to the panel through DMPU-CPC cable. This accessory is of great help when the device is installed in a drawer system to allow re-programming,

DMPU-CPAN

datalogger downloading and troubleshooting without powering the device OFF.

DMPU-CPAN or DMPU-CPC cable are connected to the RJ11 socket on the right in the bottom side of DMPU-PRB or DMPU-MBT.

DMPU-CPC

During the connection with DMPU-PS software DMPU-HMI display must be disabled (see the display instructions to enable this mode), otherwise the connection won't work properly (it's not possible to have two master devices in the same RS485 Modbus network).

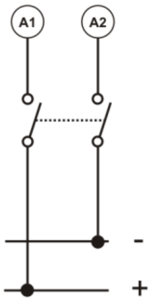
DMPU-HMI

DMPU-HMI

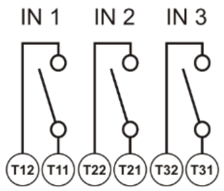
Connect DMPU-HMI display to the main module through the screw terminals, taking care of the RS485 polarity. To reduce the noise use a shielded cable and connect the shield to main module GND terminal and to the ground.

Wiring diagrams

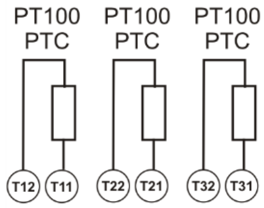
DMPU-MBT and DMPU-PRB



Power supply

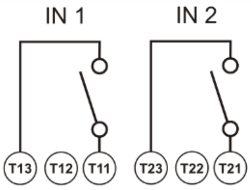


Digital inputs

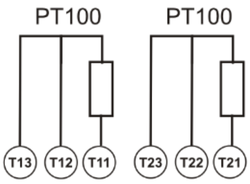


PT100 and PTC temperature sensors

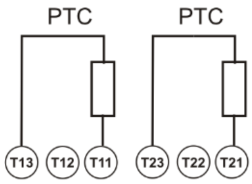
DMPU-R2



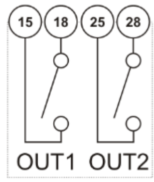
Digital inputs



PT100 temperature sensors

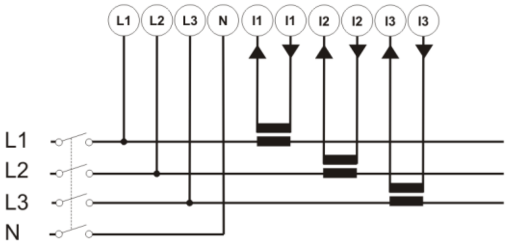


PTC temperature sensors

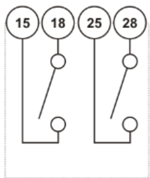


Relay outputs

DMPU-05

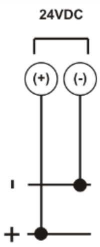


Measurement connection

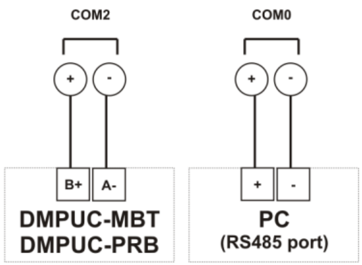


Relay output

DMPU-HMI



Power supply



RS485 ports connection

DMPU-PS software configuration

Introduction

The factory settings release the product with no function programmed. Use DMPU-PS software to create the desired configuration and upload it to the device through the communication cable.

Please power off motor while configuration is sent.

The setup of the device is defined through a guided tour where the user inputs step by step the main characteristics of the motor, the type of main module, the additional modules, the communication parameters, all the alarms and warnings for motor protection and the setup of inputs and outputs. In the main window the user can see the present configurations and modify them or upload/download a specific configuration from/to the device. The user can also load predefined configurations of parameters to fast set-up the most common operating functions as motor start/stop, reversing, star/delta starting and 2 speeds. DMPU-PS has also a section to read the data logger registers of DMPU.

Start-up

Install the software DMPU-PS provided by Carlo Gavazzi in your PC. Connect the PC RS485 port to the main module (DMPU-MBT or DMPU-PRB) through DMPU-CPC or DMPU-CPAN cable. Raw cables can also be used (using the screw terminals of the RS485 port).

At start-up configure the main parameters of the software in the main menu: “File” -> “Setup”

- Configure the RS485 communication parameters between the PC and DMPU-MBT or DMPU-PRB. The factory settings of the main modules are: speed “9600bps”, data bits “8”, parity “none”, stop bit “1” and device ID “1”.
- Define the software language.

The status of communication appears in the bottom of the main window: “offline” if the device is not correctly connected or the communication parameters are wrong, “online” if the device is ready for communication.

System requirements

- Minimum monitor resolution: 800x600 pixel
- Compatibility: Windows XP, Windows Vista, Windows 7

Software structure

The software main window has the following parts:

- Menu bar: a menu to access most common functions and basic software setup

- Toolbar: buttons to choose among the main functions of the software, used to configure and monitor the device
- Configuration list: a list of configurations created by the user

The functions of the software in the toolbar are the following:

Toolbar buttons	
Button	**Function description**
New	Add a new configuration to the configuration list. A window to enter the configuration name is shown.
Open	Load previously saved configuration from an external file.
Imp/Exp	Import the configuration from the device to the software or export the selected configuration in the configurations list to the device.
Save	Save the parameters of the highlighted configuration (in the list of configurations) to an external file.
Modify	A wizard to insert all the parameters of the configuration.
Remove	The selected configuration (in the list of configurations) is removed.
Monitoring	Monitor the instantaneous values of the variables and the virtual alarms status in the device.
Data logging	Download the data loggers and save them to an external Microsoft Excel 97-2003 file.
Commands	Execute the commands of DMPU.
Reset	Restore the device to the factory configuration.
Print	Print a document reassuming the configuration.
Close	Close the software.

Main functions

New

Press the button “New” in the toolbar to add a new configuration on the configuration list and then use the button "Modify" to enter the configuration parameters (see the relevant chapter).

Open

Press the button “Open” in the toolbar to load a configuration setup for DMPU saved previously in an external file. This configuration is added to the list of configurations.

Imp/Exp

This button is activate only if an item of the configuration list is selected. A window to choose if import the configuration from the device to the software (overwriting the data in the selected configuration) or export the configuration selected in the configuration list from software to the device (overwriting the configuration previously present in the device) is shown. An alert window is shown if the selected modules on the exported configuration don't match to the modules physically present.

Save

Saves the parameters of the selected configuration (in the list of configurations) to an external file. If there is no selected item in the list of configuration this button is disabled. Use this file to store the configuration of the plant for future maintenance and assistance.

Modify

Starts a wizard with the parameters values of the selected configuration entered last time or the default values in case of new configuration.

Through the wizard insert all the parameters of the configuration that are divided on the following 3 groups:

1. Basic parameters: to select the main module, add expansion and measurement modules, set the communication parameters between the DMPU and PC or supervision system, input the motor features.
2. Functions configuration: to define functions of inputs/outputs and the internal monitoring variable functions (ex. ANSI, counters/timers, instantaneous variables, etc.). It's also possible to add predefined configuration from external file without programming every time from scratch.
3. Data logger: to add the variables to store in the dataloggers.

Every window of the wizard has a cancel button (which allows to close the wizard) and the next and previous buttons (allowing to move forward and backward among the wizard windows). See "Wizard Description" for more details about the 3 step groups.

Remove

Removes the selected configuration after being prompted for confirmation. The removed configuration is eliminated from the software memory, not the device.

Monitoring

This window allows monitoring the instantaneous values of the variables and the virtual alarms status in the device. The window is divided into 5 sheets showing the following 5 groups of variables:

Monitoring variables	
Current sheet	
I_1	Phase 1 current
I_2	Phase 2 current
I_3	Phase 3 current
I_+	Positive motor sequence current
I_-	Negative motor sequence current
I_{IMB}	Current imbalance
TCU	Thermal Capacity Used [%]

$I_{EARTH\ 64}$	Earth fault current
THD I_1	Total harmonic distorsion of I_1
THD I_2	Total harmonic distorsion of I_2
THD I_3	Total harmonic distorsion of I_3
Voltage sheet	
V_{1-N}	L1-N voltage
V_{2-N}	L2-N voltage
V_{3-N}	L3-N voltage
$V_{L-N\Sigma}$	Average value of phase-neutral voltages
V_{1-2}	L1-L2 voltage
V_{2-3}	L2-L3 voltage
V_{3-1}	L3-L1 voltage
$V_{L-L\Sigma}$	Average value of phase-phase voltages
$AsyV_{L-N}$	Asymmetry L-N%
$AsyV_{L-L}$	Asymmetry L-L%
THD V_{1-N}	Total harmonic distorsion of V_{1-N}
THD V_{2-N}	Total harmonic distorsion of V_{2-N}
THD V_{3-N}	Total harmonic distorsion of V_{3-N}
THD V_{1-2}	Total harmonic distorsion of V_{1-2}
THD V_{2-3}	Total harmonic distorsion of V_{2-3}
THD V_{3-1}	Total harmonic distorsion of V_{3-1}
Hz	Frequency
Digital/Temperature sheet	
IN_1 to IN_{23}	Digital inputs associated to main or expansion modules (up to 23 available)
TIN_1 to TIN_{23}	Temperature inputs associated to main or expansion modules (up to 23 available)
VIN_1 to VIN_9	Virtual input associated to Modbus or Profibus input (up to 9 available)
Power sheet	
W_1	Phase1 active power
W_2	Phase 2 active power
W_3	Phase 3 active power
W_{TOT}	Total active power
VA_1	Phase 1 apparent power
VA_2	Phase 2 apparent power
VA_3	Phase 3 apparent power
VA_{TOT}	Total apparent power
VAR_1	Phase 1 reactive power
VAR_2	Phase 2 reactive power
VAR_3	Phase 3 reactive power
VAR_{TOT}	Total reactive power
PF_1	Phase 1 power factor
PF_2	Phase 2 power factor
PF_3	Phase 3 power factor
PF_{TOT}	Total power factor

Operating variables sheet	
kWh_{TOT}	Active energy [kWh]
$kVARh_{TOT}$	Reactive energy [kVArh]
N_S	Total number of Starts
N_{SH}	Number of starts per hour
T_{RTOT}	Total running hours
T_{RPAR}	Partial running hours
T_{BT}	Time estimation before trip (associated with ANSI 49)
T_{BR}	Time estimation before restart (associated with ANSI 66)

The virtual alarms status is also shown (tripped or not).

Data logging

Through this function it is possible to download the data loggers from DMPU. The data tables are saved to a Microsoft Excel 97-2003 file. When saving the data loggers to the PC, it's possible to reset the values stored in the device.

Commands

This function allows to execute the DMPU commands which are the following:

Commands list	
Command	**Description**
Reset total kWh	Reset to zero the Active energy counter
Reset total kVArh	Reset to zero the Reactive energy counter
Reset total numbers of starts	Reset to zero the Number of starts counter
Reset partial running hours	Reset to zero the Partial running hours counter
Reset Max motor start time	Reset to zero the Max motor start time counter
Reset Max start currents	Reset to zero the Max start currents counter
Reset datalogger	Reset the memory of Datalogger
Reset data event	Reset the memory of Dataevent
Reset fast datalogger	Reset the memory of Fast datalogger

Reset

This function restores the device to factory configuration.

Print

Prints a document reassuming the configuration. This document includes the following contents:

- The synoptic of the connections (inserted in the graphic connection panel during the configuration).
- The synoptic of the connections for the “logic function blocks”.
- The parameter list values of each block used.

Use this document to store the configuration of the plant for better maintenance and assistance.

Close

The software is closed.

Wizard description

Basic parameters

The wizard, regarding the basic parameters, is divided in 4 windows that are described below.

Modules configuration

Defines the main module type, the measurement module and adds the expansion modules in use: use the buttons to add modules in the used module list or to delete them from the list; the max number of connectable modules being 10.

⚠ The modules in the list of modules used must be listed in the same order as physically mounted; otherwise the device may not work properly.

The types of available expansion modules are as follows:

Expansion modules

Expansion module	Description
DMPU-R2	I/O module (with digital/temperature inputs and relay outputs)

Select the temperature unit ("Celsius" or "Fahrenheit") which will be used for all the measured temperatures.

Communication

Set the communication port parameters of the main module:

Main module communication parameter

DMPU-MBT	
Ethernet parameters:	“IP address”, the “Subnet mask”, the “Default gateway” and the “Modbus TCP/IP port”; IP address is fixed (DHCP isn't available)
Modbus RTU parameters:	Instrument address, bps rate, parity and stop bit.
DMPU-PRB	
Profibus parameter:	Profibus address
Modbus RTU parameters:	Instrument address, bps rate, parity and stop bit.

⚠ The communication parameters become effective when the device is turned OFF and ON.

CT and VT parameters

If DMPU-05 module is used, input the R_{CT} (current transformer ratio) and R_{VT} (voltage transformer ratio) parameters. All the electric variables that are used already take these ratios into consideration. Set the ratios to 1 if CT's and VT's aren't used.

Motor features

Define the following general parameters:

General parameters	
Parameter	**Function**
I_N	Nominal current of the motor (from the nameplate of the motor or from its datasheet)
t_{ST}	Nominal motor start time (depends on the application and the type of load)
I_{S49-LR}, t_{S49-H}, t_{S49-C}, k_{49}, K_{49-R}, K_{49-S}	See the ANSI 49 thermal image description

Function configurations

The digital/temperature inputs, internal functions (instantaneous variables alarms, ANSI functions, counters/timers, logic functions) and relay output are represented as graphic blocks. All of them (except the outputs) are named "virtual alarms": they are internal alarms of DMPU and can be connected to the relay outputs or read from host control system for process control (see the two words in the communication protocol representing the virtual alarms status). The functions and the status of these alarms (trip or not trip) are described in the "Block description" chapter.

All these blocks have input pins (on the left of the block) and/or output pins (on the right of the block). Connect the pins among different blocks by clicking and dragging if a block function depends on another virtual alarm status: each connection between the blocks links the function block input to the function status of another block.

The user can connect the blocks directly to the output relay or through logic function or ANSI functions, counters/timers, internal counters or instantaneous variables blocks. The user can also add a block without connection to the output relay (for instance to monitor it from the supervision system on the Ethernet or Profibus port).

Functions setup is made through a graphic tool where all the above elements are represented by blocks divided in the following groups:

Blocks list	
Inputs	
Main.1, Main.2, Main.3	3 digital or temperature inputs associated to main modules

From 1.R2.1 to 10.R2.2	Up to 20 digital or temperature inputs associated to expansion modules
From Virtual.1 to Virtual.9	Virtual input associated to Modbus or Profibus input (up to 9 available)
Instantaneous variable functions	
V_{1-N}	L1-N voltage
V_{2-N}	L2-N voltage
V_{3-N}	L3-N voltage
V_{L-N}	Average value of phase-neutral voltages
V_{1-2}	L1-L2 voltage
V_{2-3}	L2-L3 voltage
V_{3-1}	L3-L1 voltage
V_{L-L}	Average value of phase-phase voltages
I_1	Phase 1 current
I_2	Phase 2 current
I_3	Phase 3 current
I_{EARTH}	Earth fault current
W_1	Phase1 active power
W_2	Phase 2 active power
W_3	Phase 3 active power
W_{TOT}	Total active power
VA_1	Phase 1 apparent power
VA_2	Phase 2 apparent power
VA_3	Phase 3 apparent power
VA_{TOT}	Total apparent power
VAR_1	Phase 1 reactive power
VAR_2	Phase 2 reactive power
VAR_3	Phase 3 reactive power
VAR_{TOT}	Total reactive power
PF_1	Phase 1 power factor
PF_2	Phase 2 power factor
PF_3	Phase 3 power factor
PF_{TOT}	Total power factor
Hz	Frequency
$AsyV_{L-N}$	Asymmetry L-N%
$AsyV_{L-L}$	Asymmetry L-L%
Phase sequence	L1-L2-L3 or L1-L3-L2 phase sequence
I_+	Positive sequence component of motor current
I_-	Negative sequence component of motor current
THD V_{1-N}	Total harmonic distorsion of V_{1-N}
THD V_{2-N}	Total harmonic distorsion of V_{2-N}
THD V_{3-N}	Total harmonic distorsion of V_{3-N}
THD V_{1-2}	Total harmonic distorsion of V_{1-2}

THD V_{2-3}	Total harmonic distorsion of V_{2-3}
THD V_{3-1}	Total harmonic distorsion of V_{3-1}
THD I_1	Total harmonic distorsion of I_1
THD I_2	Total harmonic distorsion of I_2
THD I_3	Total harmonic distorsion of I_3
TCU	Thermal Capacity Used [%]
ANSI functions	
ANSI 49	Thermal image
ANSI 46	Inverse sequence current
ANSI 50	Overcurrent
ANSI 64	Earth fault
ANSI 66_{SH}	Starts per hour
ANSI 66_{MTBS}	Time between starts
ANSI 66_{MTFLS}	Time from last stop
ANSI 37	Undercurrent
ANSI 27S	Undervoltage
ANSI 59	Overvoltage
ANSI 47	Phase sequence
ANSI 27D	Phase loss
I_{IMB}	Current imbalance
Counters/timers	
Counter 1, Counter 2	Auxiliary counter: one input to increment and one input to reset the counter
Timer 1, Timer 2	Auxiliary timer: one input to activate and one input to reset the timer
Internal counters	
N_{SH}	Starts per hour
T_{BT}	Estimated time before trip (associated with ANSI 49)
T_{BR}	Estimated time before restart (associated with ANSI 66s)
Digital outputs	
Main.1, Main.2	2 relay outputs associated to measurement module
From 1.R2.1 to 10.R2.2	Up to 20 relay outputs associated to expansion modules
Latch reset	Internal output to reset all the active functions which have been set for latching (functions set as "Enabled and latched")
Logic functions	
From Truth table 1 to Truth table 9	6IN/1OUT truth table: AND/OR/NOT relationships and their combinations between up to 6 input to deliver 1 output

Drag the required blocks from the tree view (on the left) and drop them to the graphic connection panel (on the right) to be then connected with other blocks.

Notes:

- The maximum number of blocks (except from the output blocks) is 32.
- The number of digital/temperature input and output relay blocks available (in the block list) depends on the number and types of physical modules

used (ex. if one DMPU-R2 is used; 2 additional inputs and outputs are available).
- Each digital/temperature input and digital output blocks has label *on the bottom* of the block to describe the module type (the label prefix is a number which identifies the physical location of installation) and the number of input in the module above the pin (to identify the physical input/output in the module) to uniquely identify the physical input/output in the connection panel.
- The “Logic function” blocks available are 9.
- It's possible to use up to 2 timers and 2 counters.
- The instantaneous variables blocks can be used more than once to set different alarm trip levels.

Every block has a label name on the top set by the user (stored in the device) and a label at the bottom to uniquely identify of block type.

Every block group has parameters described in the specific chapter to set through dedicated pop-up windows. When the user right clicks an existing block, the respective pop-up window is shown automatically.

Using the button on the bottom of the window it's possible to add predefined configurations to the graphic connections panel; it allows to set-up typical motor settings without programming every time from scratch. These functions are saved in external files containing the graphic connections among the blocks and the parameters' values of the blocks. Only one pre-defined function can be added. It's possible to use the Carlo Gavazzi functions or to save personal predefined configurations and use them in the same way.

Data logger

The data logger function stores some variables of the device; there are 3 data loggers:

Dataloggers list

Name	Description	Variable type
Database logging	Max. 9999 data with date/hour reference based on FIFO storage	Average values on programmable time windows (max. 20 variables)
Data event logging	Max. 9999 data with date/hour reference based on FIFO storage	By event
Fast data logger	Max. 9999 data with progressive number based on stack storage	Instantaneous value from the start event (max. 20 variable, fixed time resolution 100ms)

The datalogger features are defined in three steps:

- Database logging: the variables to store are added to the list through the "add" button (the "delete" one cancels from the list) among the available ones. The database enabling and the time base is set through the proper box. If the database isn't activated the variables aren't stored on the device.
- Data event logging: the variables are selected from the list. Each state

change of the variables is stored in data logger. The data event logging enabling is set through the proper box. If it isn't activated the events aren't stored on the device. The available variables are the following:

- Modules conn. error: communication failure with expansion modules.
- Meas. modules conn. error: communication failure with measurement module.
- Too long start
- Start/Stop motor: the motor was started or stopped.
- Module configuration error
- Data base logging reset: the database logging was reset.
- Fast data logger reset: the fast data logger was reset.
- Data event reset: the data event logging was reset.
- DMPU power OFF: the device was powered off.
- DMPU power ON: the device was powered on.
- Latch reset command: the latch virtual alarm reset command is executed
- Used virtual inputs: the virtual inputs changed status.
- Digital inputs status: the digital inputs changed status.
- Digital outputs status: the outputs changed status.

- Fast data logger: this datalogger works just as the database logger apart from the time base which is 100 ms during each motor start. The data fast logger enabling is set through the proper box. If it isn't activated the events aren't stored on the device.

Blocks description

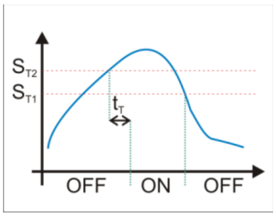
Every block in the graphic connection panel has a pop-up window where the values of the respective parameters are inserted to define the block function. Right-click the block to show the pop-up window.

Latch function

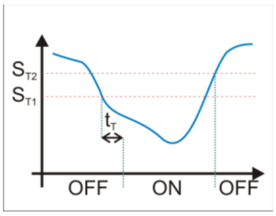
For each block the function can be latched or not: if the block is latched the function maintains the alarm state until the "internal latch reset" block is activated (all blocks functions with this feature enabled are released by "internal latch reset" activation; the alarms status aren't released after DMPU power OFF), otherwise the alarm remains active until the monitored value goes back to non-alarm values.

If the "Internal latch reset" is activated but the latched alarm is in tripping condition the alarm remains active regardless of the time delay. If the alarm is set with hysteresis it's released only when the variables goes under S_{T1}/S_{P1} (defined below) setpoint in case of "over level" or goes over S_{T2}/S_{P2} set-point in case of "under level".

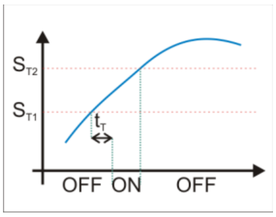
Digital/temperature input



Over level function



Under level function



Input types are the following:

- 2 or 3 wires PT100: 3 wires PT100 is available only in DMPU-R2 modules, 2 wires PT100 is available in DMPU-MBT/DMPU-PRB.
- 2 wires PTC.
- Toggle digital input: at each push the alarm status changes state (DMPU stores the previous state even if the device is turned OFF and ON).
- Switch digital input: when the input is activated the block status is ON; when the input is de-activated the block status is OFF.

If PT100 is used, the device measures the temperature and compares the value against two set-points (S_{T1} and S_{T2}). Set the two levels (upper level must be greater than lower level), the delay time (t_T) and the type of comparison that is one of the following:

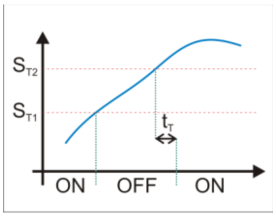
- Alarm over level (with hysteresis): when the measured value goes above the “over level” for all “delay time” the alarm trips. The alarm is released when the measured value goes below the “under level”.
- Alarm under level (with hysteresis): when the measured value goes below the “under level” for all “delay time” the alarm trips. The alarm is released when the measured value goes above the “over level”.
- Alarm in window (without hysteresis): when the measured value goes between the “over level” and the “under level” for all “delay time” the alarm trips. The alarm is released when the measured value goes below the “under level” or above the “over level”.
- Alarm out window (without hysteresis): when the measured value goes

below the “under level” or above the “over level” for all “delay time” the alarm trips. The alarm is released when the measured value goes between the “over level” and the “under level”.

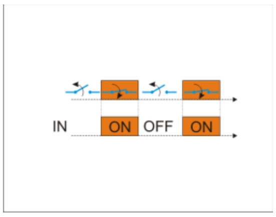
If switch digital input is used, choose one of the following input type:

- Active when closed: when the physical input is closes the alarm trips after the set delay time, it is released when the physical input is open.
- Active when open: when the physical input is open the alarm trips after the set delay time; it is released when the physical input is closed.

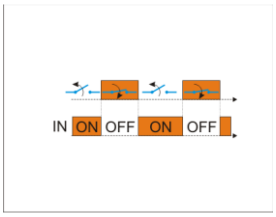
In window level function



Out window level function

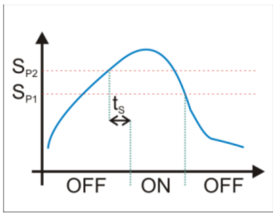


Active when closed input

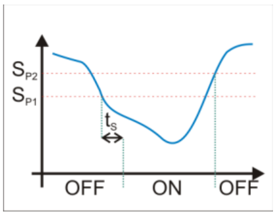


Active when open input

Instantaneous variables



Over level function



Under level function

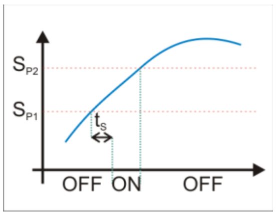
The device measures the variable and compares the value against two set-points (S_{P1} and S_{P2}). Set the two levels (upper level must be greater than lower level), the delay time (t_S) and the type of comparison that is one of the following:

- Alarm over level (with hysteresis): when the measured value goes above the “over level” for all “delay time” the alarm trips. The alarm is released when the measured value goes below the “under level”.
- Alarm under level (with hysteresis): when the measured value goes below the “under level” for all “delay time” the alarm trips. The alarm is released when the measured value goes above the “over level”.
- Alarm in window (without hysteresis): when the measured value goes between the “over level” and the “under level” for all “delay time” the alarm trips. The alarm is released when the measured value goes below the “under level” or above the “over level”.
- Alarm out window (without hysteresis): when the measured value goes below the “under level” or above the “over level” for all “delay time” the alarm trips. The alarm is released when the measured value goes

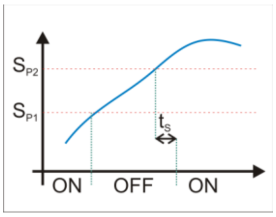
between the “over level” and the “under level”.

The formulas to determine the variables which aren't directly measured are:

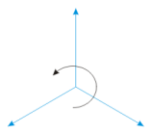
- $V_{L\text{-}N\Sigma} = (V_{1\text{-}N} + V_{2\text{-}N} + V_{3\text{-}N})/3$
- $V_{L\text{-}L\Sigma} = (V_{1\text{-}2} + V_{2\text{-}3} + V_{3\text{-}1})/3$
- $W_{TOT} = W_1 + W_2 + W_3$
- $VA_{TOT} = VA_1 + VA_2 + VA_3$
- $VAR_{TOT} = VAR_1 + VAR_2 + VAR_3$
- $PF_{TOT} = W_{TOT}/VA_{TOT}$
- $AsyV_{L\text{-}N} = (V_{L\text{-}N\ max} - V_{L\text{-}Nmin})/V_{L\text{-}N\Sigma}$
  - $V_{L\text{-}N\ max}$ is the maximum value among phase-neutral voltages
  - $V_{L\text{-}Nmin}$ is the minimum value among phase-neutral voltages
- $AsyV_{L\text{-}L} = (V_{L\text{-}L\ max} - V_{L\text{-}Lmin})/V_{L\text{-}L\Sigma}$
  - $V_{L\text{-}N\ max}$ is the maximum value among phase-phase voltages
  - $V_{L\text{-}Nmin}$ is the minimum value among phase-phase voltages
- I_{IMB} calculation (I_{MAX} is the maximum value of three phase curren):
  - When the average current (I_{AV}) is greater than the rated motor current:
    $I_{IMB} = (I_{MAX} - I_{AV})/I_{AV}$
  - When the average current (I_{AV}) is less than the rated motor current:
    $I_{IMB} = (I_{MAX} - I_{AV})/I_N$
- I_+, I_- and I_0: the three-phase current system is decomposed into a direct or positive-sequence, inverse or negative-sequence and homopolar or zero-sequence system. The vector sum of these components is the motor current. The direct sequence is associated with a positively rotating vector whereas the inverse sequence is associated with a negative rotating vector. The homopolar component has the identical phase angles.



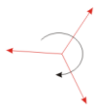
In window level function



Out window level function

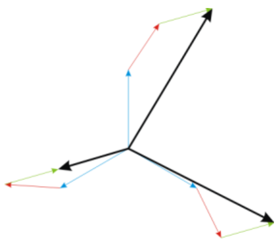


Positive sequence component



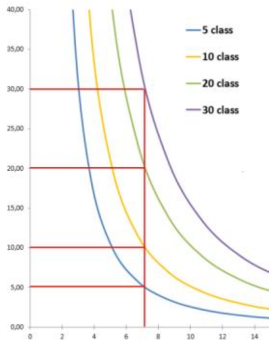
Negative sequence component

Zero sequence component

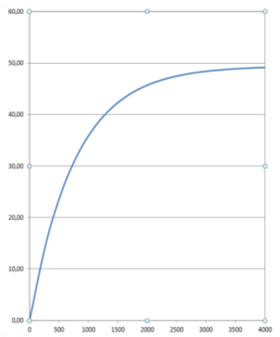


Vector sum (3 motor currents)

ANSI functions



ANSI 49 trip classes



TCU growing path example

ANSI 49 Thermal image

This function allows to protect the motor against damages due to thermal effects taking place in overload conditions, starting from current measurement.

The protection function trips when motor heating, i.e. the heat quantity in the motor, (represented by the TCU parameter – Thermal Capacity Used) reaches 100% of the maximum one for that specific motor. Some of the motor plate parameters allow to estimate how long an overload condition can be sustained before motor overheats.

The estimation of motor heating begins from the equivalent motor current calculation (I_{EQ}) that is derived from the following values:

- I_+: the direct sequence motor current.
- I_-: the inverse sequence motor current.
- I_{S49-LR}: locked rotor current (expressed as number of times against nominal current). This parameter changes the contribution of the inverse current component on the I_{EQ} calculation; the I_{EQ} current is greater as the I_{S49-LR} is smaller.

The overload condition depends on I_{EQ} value and the following parameters:

- I_N: nominal current of the motor
- k_{49}: the motor service factor: the typical value is 1.15

k_{49} x I_N defines the current value that the motor can absorb for an indefinite time without problems. Set these parameters to define the maximum current that never generates an alarm. When I_{EQ} is greater than k_{49} x I_N the motor is in overload condition and the TCU value grows until 100% unless the motor goes back to standard load condition. The time to reach the 100% value depends from the following parameters:

- K_{49-R}: time constant when the motor is already running
- K_{49-S}: time constant when the motor is just started

In case of uncertainty the user can input the values suggested by the software according to the trip classes (refer to IEC 60947-4-1): class 5, class 10, class 20 or class 30. In this case the tripping curve is calculated to observe the time to trip of the IEC standard (the class indicates the maximum tripping time within which DMPU must trip cold at 7.2 times the nominal current I_N):

Trip classes

Class	Tripping time T_p for 7.2 x I_{EQ}
5	$3<T_p\leq5$
10	$4<T_p\leq10$
20	$6<T_p\leq20$
30	$9<T_p\leq30$

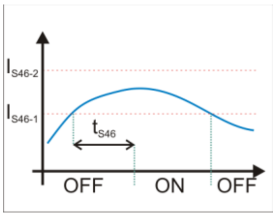
The following parameters are used to estimate the TCU value when not in overload condition:

- t_{S49-H}: max locked rotor time with hot motor
- t_{S49-C}: max locked rotor time with cold motor

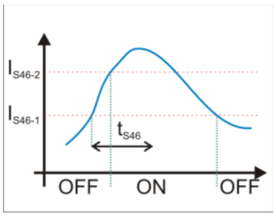
t_{S49-H} and t_{S49-C} values affect the TCU estimation during standard load condition (I_{EQ} < k_{49} x I_N). The reached TCU value depends on the ratio between t_{S49-H} and t_{S49-C}; the reached TCU is greater as the t_{S49-H} and t_{S49-C} ratio is smaller.

Modify the suggested values according to the motor plate parameters or datasheet given by motor manufacturer.

ANSI 46 Inverse sequence current



ANSI 46 delayed trip



ANSI 46 immediate function trip

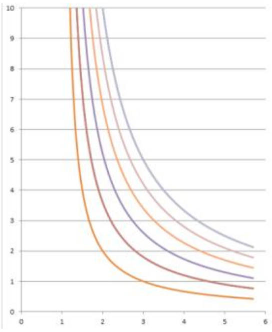
This function monitors the negative component of motor current (I_-) which is one of the main causes of motor heating. It is based on two set-points:

- I_{S46-1}: the reference for the delayed trip
- I_{S46-2}: the reference for immediate trip

I_{S46-2} must be greater than I_{S46-1}. When the measured value goes above I_{S46-1} for all t_{S46} ("delay time") the alarm trips. If negative current goes above I_{S46-2} value alarm trips immediately.

Select the dependent or independent time function to calculate the delay time:

- Time dependent function: t_{S46} depends on K_{46} and the value of negative current (the delay time is reduced as negative current increases). The delay increases increasing K_{46}.
- Time independent function: t_{S46} equals K_{46} (constant time).



ANSI 46 time dependent function

ANSI 27D Phase loss

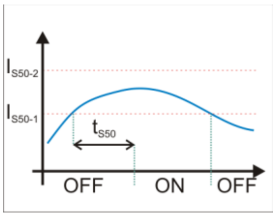
This function monitors if any of the phase-phase voltages measured goes below 70% of the mains voltage. Set V_{S27D} nominal voltage.

ANSI 47 Phase sequence

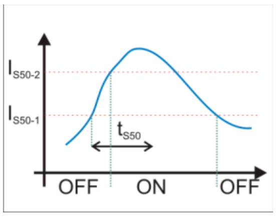
This function monitors the voltage phase sequence:

- if the phase sequence is L1-L2-L3 alarm is OFF
- if the phase sequence is L1-L3-L2 alarm is ON

ANSI 50 Overcurrent (max phase current)



ANSI 50 delayed trip



ANSI 50 immediate function trip

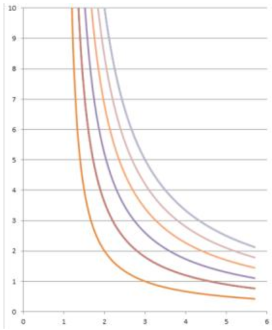
This function monitors if any of the phase currents measured is too high. It is based on two set-points:

- I_{S50-1}: for delayed alarm trip
- I_{S50-2}: for immediate alarm trip

I_{S50-2} must be greater than I_{S50-1}. When one of the currents goes above I_{S50-1} for all t_{S50} (delay time) alarm trips. If the one of the currents goes above I_{S50-2} value alarm trips immediately.

Select the dependent or independent time function to calculate t_{S50} time:

- Time dependent function: t_{S50} depends on K_{50} and the value of overcurrent (the delay time is reduced as current increases). The delay increases increasing K_{50}.
- Time independent function: t_{S50} equals K_{50} (constant time).



ANSI 50 time dependent function

ANSI 66 Starts per hour

This function monitors that the number of starts during the set time period is lower than the set point. Set the following parameters:

- P_{Thours}: observation period (0 to 360 minutes)
- N_a: max number of starts during the observation period (1 to 100)

Start condition is detected when current goes above 10% of nominal value.

When the number of starts during the last P_{Thours} time goes above the N_a number the alarm trips. As the measurement time window scrolls, when the number of starts becomes lower than N_a the alarm is released. If the alarm is active and the motor is restarted, it is automatically released. The time to restart appears among the internal counters and it is the longest time among all ANSI66s time of reentry alarm: it becomes zero when all ANSI66 alarms are deactivated.

ANSI 66 Minimum time between starts

This function monitors the time since previous start. Set the following parameter:

- $t_{StartStart}$: minimum time between starts (0 to 5400 seconds)

The alarm is active as long as time from previous start is below $t_{StartStart}$ time. In case the motor is started again before this alarm is released the alarm condition cancels automatically. If the alarm is active and the motor is restarted, the alarm is automatically released. The time to restart appears among the internal counters and it is the longest time among all ANSI66s time of reentry alarm: it becomes zero when all ANSI66 alarms are deactivated.

ANSI 66 Minimum time from last stop

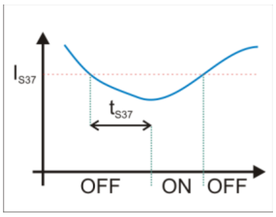
This function monitors the time since the previous stop through the parameter:

- $t_{StopStart}$: minimum time from last stop (0 to 5400 seconds)

Stop event is detected when current goes below 10% of nominal current.

The alarm is active as long as time from previous stop is below $t_{StopStart}$ time. In case the motor is started again before this alarm is released the alarm condition cancels automatically. The time to restart appears among the internal counters and it is the longest time among all ANSI66s time of reentry alarm: it becomes zero when all ANSI66 alarms are deactivated.

ANSI 37 Undercurrent (minimum phase current)



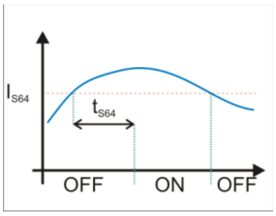
ANSI 37 delay time function trip

This function monitors if any of the currents is below the set-point current during the set time. Set the following parameters:

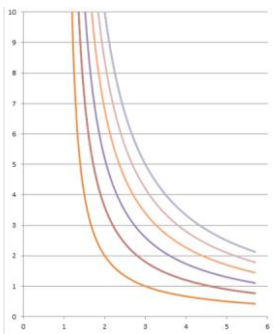
- I_{S37}: minimum current set point (range 0.1 I_N - I_N)
- t_{S37}: time delay (1 to 300 seconds)

If all motor currents are below 10% of I_N the motor is supposed to be off and there is no alarm. If they are above 10% of I_N, the function is blanked during the start-up period, after which alarm trips if at least one of the currents drops below the I_{S37} value for t_{S37} time.

ANSI 64 Earth fault



ANSI 64 delayed trip



ANSI 64 time dependent

The function measures the vector sum of the three phase currents and assumes that neutral isn't present. The sum is the earth fault current. It uses one set-point:

- I_{S64}: current set point for alarm tripping (0 to 200% of I_N).

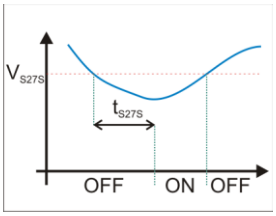
Select the time dependent or independent function to calculate the t_{S64} delay time:

- Time dependent function: t_{S64} depends on K_{64} and the value of earth fault current (the delay time is reduced as current increases). The delay increases increasing K_{46}.
- Time independent function: t_{S64} equals to K_{64} (constant time).

When the measured value goes above the I_{S64} for all t_{S64} the alarm trips.

function

ANSI 27S Undervoltage



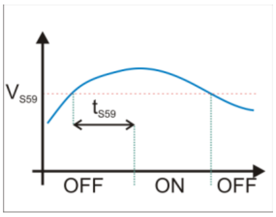
ANSI 27S delayed trip

This function monitors if any of the phase-phase voltages is too low for the delay time. Set the following parameters:

- V_{S27S}: minimum voltage set point (if TV is used this value refers to the motor voltage value)
- t_{S27S}: time delay

Alarm trips if at least one of the phase-phase voltages is below V_{S27S} for time t_{S27S}.

ANSI 59 Overvoltage



ANSI 59 delay time function trip

This function monitors if any of the phase-phase voltages is too high during the set time. Set the following parameters:

- V_{S59}: maximum voltage set point
- t_{S59}: time delay for maximum voltage (1-30000 seconds)

Alarm trips if at least one of the phase-phase voltages is above V_{S59} for the time t_{S59}.

Counters/timers

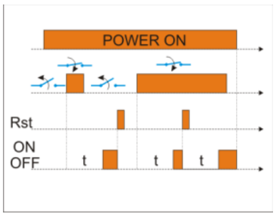
Counters

Two internal incremental counters are available. Set the following parameter:

- N_c: set point of the counter (0 to 6500 counts)

Each time the input (connected to the input pin of the block) becomes active, the counter is incremented; when it reaches N_c the alarm trips. It's possible to reset the counter through the reset input.

Timers



Timer function

Two internal timers are available. Set the following parameter:

- T_t: set point time of the timer (0 to 6500 seconds)

The timer is activated or reset by the inputs (connected to the input pins of the block). The timer alarm becomes active after the set point time from the trigger input activation and it maintains this status as long as the reset command is activated.

Internal counters

Starts per hour

Monitors the number of start during P_{Thours} observation period (see the ANSI 66 function). The user can activate an alarm on this counter to pre-warn a ANSI 66 trip. This counter is available if the ANSI 66 starts per hour is used.

Estimated time before trip

It is the estimation of time before ANSI 49 trips; this counter is very useful for pre-warning as it gives the forecast of the trip time. This value is available if ANSI 49 is used.

Estimated time before restart

It is the longest time needed by the ANSI 66 functions to be released. This counter is available if ANSI 66s are used.

Digital outputs

Use these blocks to setup the output relays of the modules. If more than one block is connected at the pin of output block, the relay is activated when at least one block connected to the pin is active (OR logic). Select if to use a normally open or normally close contact.

Latch reset

This function refers to all latched alarms. Activate the status of this block to release the active status of the used blocks with latch function enabled. If more than one block is connected to the pin, the block is activated when at least one block connected to the pin is active (OR logic).

Logic functions

There are 9 truth tables. They summarize the relationship among up to 6 inputs and 1 output. When a block is added to the graphic table the user has to choose if the logic function is described through a graph or a table:

- Graph: the logic function is represented by a combination of elementary logic functions (OR, AND, NOT); the elementary logic function, the 6 inputs and 1 output are represented as graphic blocks. Connect these blocks among them to define the desired function.
- Truth table: the logic function is represented by a table that summarizes all combinations of inputs states and its output state ("1" enable, "0" disable): activate or deactivate the output state for each combination setting the output to "1" or to "0". Clicking the cell corresponding to the output state to switch the value.

If the configuration is loaded directly from the device, the truth table is seen (not the graphic connections of AND/OR/NOT logic functions).

If the graph is used the truth table is available to show the output, but can't be modified.

Introduction to Modbus

Introduction

The Modbus RTU is available through RS485 serial port which is supplied in both DMPU main modules (DMPU-MBT and DMPU-PRB); the Modbus TCP/IP is only available on DMPU-MBT through ethernet port (RJ45 connector).

Modbus RTU protocol is a messaging structure used to establish master-slave communication between devices in which only one device (called master) can initiate transactions (called queries); the other devices (called slaves) respond with the requested data to the master. Modbus device includes a registers map which shows the internal variables/parameters; the modbus functions operate on the register map to monitor, configure and control the device (for the DMPU register map see the relevant chapter). The register map is the same for Modbus RTU and Modbus TCP/IP communication.

For a complete description of the MODBUS protocol please refer to the documentation available in the www.modbus.org web site.

Modbus functions

These functions are available on DMPU:

1. Reading of n holding registers (code 03h)
2. Reading of n input register (code 04h)
3. Writing of one holding registers (code 06h)
4. Writing of multiple registers (code 10h)
5. Diagnostic (code 08h with sub-function code 00h)
6. Reading of a record file (code 14h with sub-code 06h)
7. Reading of n special registers (code 42h)
8. Broadcast mode (writing instruction on address 00h)

1. In this document the Modbus address field is indicated in two modes:
    - Modicon address: it is the “6 digit Modicom” representation with Modbus function code 04 (Read Input Registers) . It is possible to read the same values with function code 03 (Read Holding Register) substituting the first digit with number “4”.
    - Physical address: it is the word address value included in the communication frame.
2. The functions 03h and 04h have exactly the same effect with DMPU.

Function 03h (Read holding registers)

This function is used to read the contents of a contiguous block of holding registers (words). The request frame specifies the starting register address and the number of registers to be read. It is possible to read maximum 125 registers (words) with a single request.

The register data in the response message are packed as two bytes per register (word), with the binary contents right justified within each byte. For each register, the first byte contains the high order bits (MSB) and the second contains the low order bits (LSB).

Request frame

Description	Length	Value	Note
Physical Address	1 byte	1 to F7 (1 to 247)	
Function code	1 byte	03h	
Starting Address	2 bytes	0000h to FFFFh	Byte order: MSB, LSB
Quantity of Registers (N word)	2 bytes	1 to 7Dh (1 to 125)	Byte order: MSB, LSB
CRC	2 bytes		

Response frame (correct action)

Description	Length	Value	Note
Physical Address	1 byte	1 to F7 (1 to 247)	
Function code	1 byte	03h	
Byte count	1 byte	N word * 2	
Register value	N*2 bytes		Byte order: MSB, LSB
CRC	2 bytes		

Response frame (incorrect action)

Description	Length	Value	Note
Physical Address	1 byte	1 to F7 (1 to 247)	Possible exception: 01h: illegal function 02h: illegal data address 03h: illegal data value 04h: slave device failure
Function code	1 byte	83h	
Exception code	1 byte	01h, 02h, 03h, 04h	
CRC	2 bytes		

Function 04h (Read input registers)

This function is used to read the contents of a contiguous block of input registers (words). The request frame specifies the starting register address and the number of registers to be read. It is possible to read maximum 125 register (word) with a single request.

The register data in the response message are packed as two bytes per register (word), with the binary contents right justified within each byte. For each register, the first byte contains the high order bits (MSB) and the second contains the low order bits (LSB).

Request frame

Description	Length	Value	Note
Physical Address	1 byte	1 to F7 (1 to 247)	
Function code	1 byte	04h	
Starting Address	2 bytes	0000h to FFFFh	Byte order: MSB, LSB
Quantity of Registers (N word)	2 bytes	1 to 7Dh (1 to 125)	Byte order: MSB, LSB
CRC	2 bytes		

Response frame (correct action)

Description	Length	Value	Note
Physical Address	1 byte	1 to F7 (1 to 247)	

Function code	1 byte	04h	
Byte count	1 byte	N word * 2	
Register value	N*2 bytes		Byte order: MSB, LSB
CRC	2 bytes		

Response frame (incorrect action)

Description	Length	Value	Note
Physical Address	1 byte	1 to F7 (1 to 247)	Possible exception: 01h: illegal function 02h: illegal data address 03h: illegal data value 04h: slave device failure
Function code	1 byte	84h	
Exception code	1 byte	01h, 02h, 03h, 04h	
CRC	2 bytes		

Function 06h (Write single holding register)

This function is used to write a single holding register. The request frame specifies the address of the register (word) to be written and its contents.

The correct response is an echo of the request, returned after the register contents have been written.

Request frame

Description	Length	Value	Note
Physical Address	1 byte	1 to F7 (1 to 247)	
Function code	1 byte	06h	
Starting Address	2 bytes	0000h to FFFFh	Byte order: MSB, LSB
Register value	2 bytes	0000h to FFFFh	Byte order: MSB, LSB
CRC	2 bytes		

Request frame (correct action)

Description	Length	Value	Note
Physical Address	1 byte	1 to F7 (1 to 247)	
Function code	1 byte	06h	
Starting Address	2 bytes	0000h to FFFFh	Byte order: MSB, LSB
Register value	2 bytes	0000h to FFFFh	Byte order: MSB, LSB
CRC	2 bytes		

Response frame (incorrect action)

Description	Length	Value	Note
Physical Address	1 byte	1 to F7 (1 to 247)	Possible exception: 01h: illegal function 02h: illegal data address 03h: illegal data value 04h: slave device failure
Function code	1 byte	86h	
Exception code	1 byte	01h, 02h, 03h, 04h	
CRC	2 bytes		

Function 10h (Write multiple registers)

This function is used to write a block of contiguous registers (maximum 120). The requested values to be written are specified in the request data field. Data is packed as two bytes per register.

The correct response returns the function code, starting address, and the quantity of written registers.

Request frame			
Description	**Length**	**Value**	**Note**
Physical Address	1 byte	1 to F7 (1 to 247)	
Function code	1 byte	10h	
Starting Address	2 bytes	0000h to FFFFh	Byte order: MSB, LSB
Quantity of Registers (N word)	2 bytes	0001h to 0078h	Byte order: MSB, LSB
Byte count	1 byte	N word * 2	
Register value	N * 2 bytes	value	Byte order: MSB, LSB
CRC	2 bytes		

Request frame (correct action)			
Description	**Length**	**Value**	**Note**
Physical Address	1 byte	1 to F7 (1 to 247)	
Function code	1 byte	10h	
Starting Address	2 bytes	0000h to FFFFh	Byte order: MSB, LSB
Quantity of Registers (N word)	2 bytes	0001h to 0078h	Byte order: MSB, LSB
CRC	2 bytes		

Response frame (incorrect action)			
Description	**Length**	**Value**	**Note**
Physical Address	1 byte	1 to F7 (1 to 247)	Possible exception: 01h: illegal function 02h: illegal data address 03h: illegal data value 04h: slave device failure
Function code	1 byte	90h	
Exception code	1 byte	01h, 02h, 03h, 04h	
CRC	2 bytes		

Function 08h (Diagnostic with sub-function code 00h)

MODBUS function 08h provides a series of tests to check the communication system between a client (Master) device and a server (Slave), or to check various internal error conditions within a server.

DMPU supports only 0000h sub-function code (Return Query Data). With this sub-function the data passed in the request data field are returned (looped back) in the response. The entire response message should be identical to the request.

Request frame			
Description	**Length**	**Value**	**Note**
Physical Address	1 byte	1 to F7 (1 to 247)	
Function code	1 byte	08h	
Sub-function	2 bytes	0000h	
Data (N word)	2 bytes	N word * 2	Byte order: MSB, LSB
CRC	2 bytes		

Request frame (correct action)			
Description	**Length**	**Value**	**Note**
Physical Address	1 byte	1 to F7 (1 to 247)	
Function code	1 byte	08h	
Sub-function	2 bytes	0000h	
Data (N word)	2 bytes	N word * 2	Byte order: MSB, LSB

CRC	2 bytes		

Response frame (incorrect action)

Description	Length	Value	Note
Physical Address	1 byte	1 to F7 (1 to 247)	Possible exception: 01h: illegal function 02h: illegal data address 03h: illegal data value 04h: slave device failure
Function code	1 byte	88h	
Exception code	1 byte	01h, 02h, 03h, 04h	
CRC	2 bytes		

Function 14h with sub-function 06h (Reading a record file)

This function is used to perform a record file read. All the request data lengths are provided in terms of number of bytes and all the record lengths are provided in terms of registers.

A file is set of records. Each file contains 10000 records, addressed from 0 to 9999.

The function can read multiple records using one sub-function for every record. Each sub-function is defined by 7 bytes:

- The reference type: 1 byte (must be specified as 6)
- The file number: 2 bytes
- The starting record number within the file: 2 bytes
- The length of the record to be read: 2 bytes.

Request frame

Description	Length	Value	Note
Physical address	1 byte	1 to F7 (1 to 247)	
Function code	1 byte	14h	
Byte count	1 byte	07h to F5h bytes	
1° Sub-function code	1 byte	06h	Byte order: MSB, LSB
1° Sub-function file number	2 bytes	0h to FFFFh	
1° Sub-function record number	2 bytes	0h to 270Fh	Byte order: MSB, LSB
1° Sub-function number of word (N)	2 bytes	N	Byte order: MSB, LSB
2° Sub-function code	1 byte	06h	Byte order: MSB, LSB
2° Sub-function file number	2 bytes	0h to FFFFh	
2° Sub-function record number	2 bytes	0h to 270Fh	Byte order: MSB, LSB
2° Sub-function number of word (N)	2 bytes	N	Byte order: MSB, LSB
...			
CRC	2 bytes		

The normal response is a series of sub-responses, one for each sub-function request. The response data length field is the total combined count of bytes in all sub-responses. In addition, each sub-response contains a field that shows its own byte count.

Request frame (correct action)

Description	Length	Value	Note

Physical address	1 byte	1 to F7 (1 to 247)	
Function code	1 byte	14h	
Resp. data length	1 byte	0x07 to 0xF5	
1°Sub-func. response data length	1 byte	07h to 0F5h	
1°Sub-function code	1 byte	06h	
1°Sub-func. data (N word)	2 bytes	N word * 2	Byte order: MSB, LSB
2°Sub-func. response data length	1 byte	07h to 0F5h	
2°Sub-function code	1 byte	06h	
2°Sub-func. data (N word)	2 bytes	N word * 2	Byte order: MSB, LSB
...			
CRC	2 bytes		

Response frame (incorrect action)

Description	Length	Value	Note
Physical address	1 byte	1 to F7 (1 to 247)	Possible exception: 01h: illegal function 02h: illegal data address 03h: illegal data value 04h: slave device failure
Function code	1 byte	88h	
Exception code	1 byte	01h, 02h, 03h, 04h	
CRC	2 bytes		

The request frame and the response frame must not exceed the allowable length of the MODBUS PDU: 253 bytes.

Broadcast mode

In broadcast mode the master can send a request (command) to all the slaves. No response is returned to broadcast requests sent by the master. It is possible to send the broadcast message only with function code 06h and 10h and using address 00h.

Profibus GSD file

Use the GSD file for Profibus cyclic data exchange. The following modules are inserted within GSD file: CAGA0D6E.GSD

GSD Module 1: Minimum values		
Variables	**Length (words)**	**Data Format**
Output		
9 virtual digital input	1	UINT16
Input		
Virtual alarm from 1 to 16	1	UINT16
Virtual alarm from 17 to 32	1	UINT16
TCU (thermal image)	2	32 bit IEEE 754
Status of 9 virtual digital input	2	UINT16

GSD Module 2: Phase Voltage values		
Variables	**Length (words)**	**Data Format**
Input		
L1-N voltage	2	32 bit IEEE 754
L2-N voltage	2	32 bit IEEE 754
L3-N voltage	2	32 bit IEEE 754
Avarage value of phase-neutral voltages	2	32 bit IEEE 754

GSD Module 3: Phase-Phase Voltage values		
Variables	**Length (words)**	**Data Format**
Input		
L1-L2 voltage	2	32 bit IEEE 754
L2-L3 voltage	2	32 bit IEEE 754
L3-L1 voltage	2	32 bit IEEE 754
Avarage value of phase-phase voltages	2	32 bit IEEE 754

GSD Module 4: General Voltage values		
Variables	**Length (words)**	**Data Format**
Input		
L1-N voltage	2	32 bit IEEE 754
L2-N voltage	2	32 bit IEEE 754
L3-N voltage	2	32 bit IEEE 754
Total power factor	2	32 bit IEEE 754
Total harmonic distortion of V_{1-N}	2	32 bit IEEE 754
Total harmonic distortion of V_{2-N}	2	32 bit IEEE 754
Total harmonic distortion of V_{3-N}	2	32 bit IEEE 754

GSD Module 5: Current Values		
Variables	**Length (words)**	**Data Format**
Input		
Phase 1 current	2	32 bit IEEE 754
Phase 2 current	2	32 bit IEEE 754

Phase 3 current	2	32 bit IEEE 754
Earth fault current	2	32 bit IEEE 754

GSD Module 6: Current extension values

Variables	Length (words)	Data Format
Input		
Phase 1 current	2	32 bit IEEE 754
Phase 2 current	2	32 bit IEEE 754
Phase 3 current	2	32 bit IEEE 754
Earth fault current	2	32 bit IEEE 754
Homopolar sequence component of motor current	2	32 bit IEEE 754
Positive sequence component of motor current	2	32 bit IEEE 754
Negative sequence component of motor current	2	32 bit IEEE 754

GSD Module 7: Active power values

Variables	Length (words)	Data Format
Input		
Phase 1 active power	2	32 bit IEEE 754
Phase 2 active power	2	32 bit IEEE 754
Phase 3 active power	2	32 bit IEEE 754
Total active power	2	32 bit IEEE 754

GSD Module 8: Reactive power values

Variables	Length (words)	Data Format
Input		
Phase 1 reactive power	2	32 bit IEEE 754
Phase 2 reactive power	2	32 bit IEEE 754
Phase 3 reactive power	2	32 bit IEEE 754
Total reactive power	2	32 bit IEEE 754

GSD Module 9: Apparent power values

Variables	Length (words)	Data Format
Input		
Phase 1 apparent power	2	32 bit IEEE 754
Phase 2 apparent power	2	32 bit IEEE 754
Phase 3 apparent power	2	32 bit IEEE 754
Total apparent power	2	32 bit IEEE 754

GSD Module 10: Power factor values

Variables	Length (words)	Data Format
Input		
Phase 1 power factor	2	32 bit IEEE 754
Phase 2 power factor	2	32 bit IEEE 754
Phase 3 power factor	2	32 bit IEEE 754
Total power factor	2	32 bit IEEE 754

GSD Module 11: Total power values

Variables	Length (words)	Data Format
Input		

Total active power	2	32 bit IEEE 754
Total reactive power	2	32 bit IEEE 754
Total apparent power	2	32 bit IEEE 754
Total power factor	2	32 bit IEEE 754

GSD Module 12: Harmonic distortion values

Variables	Length (words)	Data Format
	Input	
Total harmonic distortion of V_{1-N}	2	32 bit IEEE 754
Total harmonic distortion of V_{2-N}	2	32 bit IEEE 754
Total harmonic distortion of V_{3-N}	2	32 bit IEEE 754
Total harmonic distortion of I_1	2	32 bit IEEE 754
Total harmonic distortion of I_2	2	32 bit IEEE 754
Total harmonic distortion of I_3	2	32 bit IEEE 754

GSD Module 13: Electrical extension values

Variables	Length (words)	Data Format
	Input	
Frequency	2	32 bit IEEE 754
Asymmetry L-N %	2	32 bit IEEE 754
Asymmetry L-L %	2	32 bit IEEE 754
Total harmonic distortion of V_{1-2}	2	32 bit IEEE 754
Total harmonic distortion of V_{2-3}	2	32 bit IEEE 754
Total harmonic distortion of V_{3-1}	2	32 bit IEEE 754

GSD Module 14: Base running motor values

Variables	Length (words)	Data Format
	Input	
Active energy (LSW)	2	UINT32
Active energy (MSW)	2	UINT32
Reactive energy (LSW)	2	UINT32
Reactive energy (MSW)	2	UINT32
Estimated time before trip	2	UINT32
Estimated time before restart	2	UINT32
Partitial running hours	2	UINT32
Max motor start time	2	UINT32

GSD Module 15: Running motor extension values

Variables	Length (words)	Data Format
	Input	
Total number of starts	2	UINT32
Starts per hour	2	UINT32
Total running hours	2	UINT32
Total running seconds	2	UINT32
Hours before greasing	2	UINT32
Starts before contact maintenance	2	UINT32
Partitial running hours	2	UINT32

GSD Module 16: Auxiliary function		
Variables	**Length (words)**	**Data Format**
Input		
Counter #1	2	UINT32
Counter #2	2	UINT32
Timer #1	2	UINT32
Timer #2	2	UINT32

GSD Module 17: Thermal base values		
Variables	**Length (words)**	**Data Format**
Input		
Temperature/digital input 1	1	INT16
Temperature/digital input 2	1	INT16
Temperature/digital input 3	1	INT16
TCU (thermal image)	2	32 bit IEEE 754

GSD Module 18: External module variables		
Variables	**Length (words)**	**Data Format**
Input		
Module code	1	UINT16
Module status	1	UINT16
Word 1	1	INT16
Word 2	1	INT16
Word 3	1	INT16
Word 4	1	INT16
Word 5	1	INT16
Word 6	1	INT16

DMPU registers map

Data format representation

In this chapter is reported the registers map of DMPU. These registers are formatted according to one of the following data representations:

Register data formats

Format	IEC data type	Description	Bits	Range
INT16	INT	Integer	16	-32768 .. 32767
UINT16	UINT	Unsigned integer	16	0 .. 65535
INT32	DINT	Double integer	32	-2^{31} .. 2^{31}
UINT32	UDINT	Unsigned double int	32	0 .. 2^{32}-1
UINT64	ULINT	Unsigned long integer	64	0 .. 2^{64}-1
IEEE754 SP		Single-precision floating-point	32	$-(1+[1 - 2^{-23}])x2^{127}$ .. 2^{128}
ASCII		ASCII char extended	8	0..255

The IEEE754 representation of a 32-bit floating-point number as an integer is defined as follows:

32 bit floating point

Bits		
31	30 ... 23	22 ... 0
Sign	Exponent	Mantissa

Formula to calculate the number: $-1^{sign} * 2^{Exponent-127} * 1.Mantissa$

The byte order in the MODBUS (and ANSI) frame is:

- 1st byte = Bits 15 … 8 of the 32-bit floating-point number in standard IEEE-754
- 2nd byte = Bits 7 … 0 of the 32-bit floating-point number in standard IEEE-754
- 3rd byte = Bits 31 … 24 of the 32-bit floating-point number in standard IEEE-754
- 4th byte = Bits 23 … 16 of the 32-bit floating-point number in standard IEEE-754

The integers are represented in UINT16 (16 bit) or UINT64 (64 bit) format without sign (the byte order inside the single word is MSB->LSB while the word order is LSW->MSW).

The byte order in the Profibus frame is big endian for every variable.

Read/write the registers listed in this chapter using the Modbus functions or

Profibus acyclic data exchange references (slot and index) shown in the bottom of the tables.

Variables Map

Instantaneous variables

Instantaneous variables from measurement module					
Modicon address	Physical address	Length (words)	Instantaneous variable		Data format
400081	0050h	2	V_{1-N}	L1-N voltage	32 bit IEEE754
400083	0052h	2	V_{2-N}	L2-N voltage	32 bit IEEE754
400085	0054h	2	V_{3-N}	L3-N voltage	32 bit IEEE754
400087	0056h	2	$V_{L-N\Sigma}$	Average value of phase-neutral voltages	32 bit IEEE754
400089	0058h	2	V_{1-2}	L1-L2 voltage	32 bit IEEE754
400091	005Ah	2	V_{2-3}	L2-L3 voltage	32 bit IEEE754
400093	005Ch	2	V_{3-1}	L3-L1 voltage	32 bit IEEE754
400095	005Eh	2	$V_{L-L\Sigma}$	Average value of phase-phase voltages	32 bit IEEE754
400097	0060h	2	I_1	Phase 1 current	32 bit IEEE754
400099	0062h	2	I_2	Phase 2 current	32 bit IEEE754
400101	0064h	2	I_3	Phase 3 current	32 bit IEEE754
400103	0066h	2	$I_{EARTH\ 64}$	Earth fault current	32 bit IEEE754
400105	0068h	2	W_1	Phase1 active power	32 bit IEEE754
400107	006Ah	2	W_2	Phase 2 active power	32 bit IEEE754
400109	006Ch	2	W_3	Phase 3 active power	32 bit IEEE754
400111	006Eh	2	W_{TOT}	Total active power	32 bit IEEE754
400113	0070h	2	VA_1	Phase 1 apparent power	32 bit IEEE754
400115	0072h	2	VA_2	Phase 2 apparent power	32 bit IEEE754
400117	0074h	2	VA_3	Phase 3 apparent power	32 bit IEEE754
400119	0076h	2	VA_{TOT}	Total apparent power	32 bit IEEE754
400121	0078h	2	VAR_1	Phase 1 reactive power	32 bit IEEE754
400123	007Ah	2	VAR_2	Phase 2 reactive power	32 bit IEEE754
400125	007Ch	2	VAR_3	Phase 3 reactive power	32 bit IEEE754
400127	007Eh	2	VAR_{TOT}	Total reactive power	32 bit IEEE754
400129	0080h	2	PF_1	Phase 1 power factor	32 bit IEEE754
400131	0082h	2	PF_2	Phase 2 power factor	32 bit IEEE754
400133	0084h	2	PF_3	Phase 3 power factor	32 bit IEEE754
400135	0086h	2	PF_{TOT}	Total power factor	32 bit IEEE754
400137	0088h	2	Hz	Frequency	32 bit IEEE754
400139	008Ah	2	$AsyV_{L-N}$	Asymmetry L-N%	32 bit IEEE754
400141	008Ch	2	$AsyV_{L-L}$	Asymmetry L-L%	32 bit IEEE754
400143	008Eh	2	PSQ	Phase sequence	32 bit IEEE754

400147	0092h	2	I_0	Homopolar sequence component of motor current	32 bit IEEE754
400149	0094h	2	I_+	Positive sequence component of motor current	32 bit IEEE754
400151	0096h	2	I_-	Negative sequence component of motor current	32 bit IEEE754
400153	0098h	2	THD V_{1-N}	Total harmonic distorsion of V_{1-N}	32 bit IEEE754
400155	009Ah	2	THD V_{2-N}	Total harmonic distorsion of V_{2-N}	32 bit IEEE754
400157	009Ch	2	THD V_{3-N}	Total harmonic distorsion of V_{3-N}	32 bit IEEE754
400159	009Eh	2	THD V_{1-2}	Total harmonic distorsion of V_{1-2}	32 bit IEEE754
400161	00A0	2	THD V_{2-3}	Total harmonic distorsion of V_{2-3}	32 bit IEEE754
400163	00A2h	2	THD V_{3-1}	Total harmonic distorsion of V_{3-1}	32 bit IEEE754
400165	00A4h	2	THD I_1	Total harmonic distorsion of I_1	32 bit IEEE754
400167	00A6h	2	THD I_2	Total harmonic distorsion of I_2	32 bit IEEE754
400169	00A8h	2	THD I_3	Total harmonic distorsion of I_3	32 bit IEEE754
400171	00AAh	2	TCU	Thermal Capacity Used [%]	32 bit IEEE754
400173	00ACh	2	I_{IMB}	Current imbalance	32 bit IEEE754

Read only mode with function code 03 and 04. PROFIBUS acyclic data exchange: slot 8 - index 0.

Variables from modules

DMPU can have up to 10 espansion module mounted side by side from main module. The instantaneous variables of the modules are organized in 11 groups of 8 words. Every group identify a module according to the physical location of installation (first group always represents the main module). The words references are the following:

Instantaneous variables form main and expansion modules				
Module	**Description**	**Modicon address**	**Physical address**	**Length (words)**
Main module (DMPU-MBT or DMPU-PRB)	Module code	400769	0300h	1
	Module status	400770	0301h	1
	Word #1	400771	0302h	1
	Word #2	400772	0303h	1
	Word #3	400773	0304h	1
	Word #4	400774	0305h	1
	Word #5	400775	0306h	1
	Word #6	400776	0307h	1
Espansion module #1	Module code	400777	0308h	1
	Module	400778	0309h	1

	status			
	Word #1	400779	0310h	1
	Word #2	400780	0311h	1
	Word #3	400781	0312h	1
	Word #4	400782	0313h	1
	Word #5	400783	0314h	1
	Word #6	400784	0315h	1
Espansion module #2	Module code	400785	0316h	1
	Module status	400786	0317h	1
	Word #1	400787	0318h	1
	Word #2	400788	0319h	1
	Word #3	400789	0320h	1
	Word #4	400790	0321h	1
	Word #5	400791	0322h	1
	Word #6	400792	0323h	1
Espansion module #3	Module code	400793	0324h	1
	Module status	400794	0325h	1
	Word #1	400795	0326h	1
	Word #2	400796	0327h	1
	Word #3	400797	0328h	1
	Word #4	400798	0329h	1
	Word #5	400799	0330h	1
	Word #6	400800	0331h	1
Espansion module #4	Module code	400801	0332h	1
	Module status	400802	0333h	1
	Word #1	400803	0334h	1
	Word #2	400804	0335h	1
	Word #3	400805	0336h	1
	Word #4	400806	0337h	1
	Word #5	400807	0338h	1
	Word #6	400808	0339h	1
Espansion module #5	Module code	400809	0340h	1
	Module status	400810	0341h	1
	Word #1	400811	0342h	1
	Word #2	400812	0343h	1
	Word #3	400813	0344h	1
	Word #4	400814	0345h	1
	Word #5	400815	0346h	1
	Word #6	400816	0347h	1
Espansion module #6	Module code	400817	0348h	1

	Module status	400818	0349h	1
	Word #1	400819	0350h	1
	Word #2	400820	0351h	1
	Word #3	400821	0352h	1
	Word #4	400822	0353h	1
	Word #5	400823	0354h	1
	Word #6	400824	0355h	1
Espansion module #7	Module code	400825	0356h	1
	Module status	400826	0357h	1
	Word #1	400827	0358h	1
	Word #2	400828	0359h	1
	Word #3	400829	0360h	1
	Word #4	400830	0361h	1
	Word #5	400831	0362h	1
	Word #6	400832	0363h	1
Espansion module #8	Module code	400833	0364h	1
	Module status	400834	0365h	1
	Word #1	400835	0366h	1
	Word #2	400836	0367h	1
	Word #3	400837	0368h	1
	Word #4	400838	0369h	1
	Word #5	400839	0370h	1
	Word #6	400840	0371h	1
Espansion module #9	Module code	400841	0372h	1
	Module status	400842	0373h	1
	Word #1	400843	0374h	1
	Word #2	400844	0375h	1
	Word #3	400845	0376h	1
	Word #4	400846	0377h	1
	Word #5	400847	0378h	1
	Word #6	400848	0379h	1
Espansion module #10	Module code	400849	0380h	1
	Module status	400850	0381h	1
	Word #1	400851	0382h	1
	Word #2	400852	0383h	1
	Word #3	400853	0384h	1
	Word #4	400854	0385h	1
	Word #5	400855	0386h	1
	Word #6	400856	0387h	1

Read only mode with function code 03 and 04. PROFIBUS acyclic data exchange: slot 8 - index 1.

The first word of the group indicates the type of module, the others words have different structure according to the modules:

Organisation of the instantaneous variables vs. module type				
DMPU-MBT or DMPU-PRB				
Address	**Description**	**Length (words)**	**Data format**	**Value**
base + 0h	Module code *1	1	UINT16	1=DMPU-MBT or DMPU-PRB
base + 1h	Reserved			
base + 2h	Input channel #1	1	INT16	PT100 input: -50.0 to 850.0 (°C or °F). Digital input or PTC: 0=ch open (OFF), 1=ch close (ON)
base + 3h1	Input channel #2	1	INT16	
base + 4h	Input channel #3	1	INT16	
base + 5h	Reserved			
base + 6h	Reserved			
base + 7h	Reserved			
DMPU-R2				
Address	**Description**	**Length (words)**	**Data format**	**Value**
base + 0h	Module code *1	1	UINT16	2=DMPU-R2
base + 1h	Reserved			
base + 2h	Input channel 1	1	INT16	PT100 input: -50.0 to 850.0 (°C or °F). Digital input or PTC (bit0=ch.1; bit1=ch.2): 0=ch open (OFF), 1=ch close (ON)
base + 3h	Input channel 2	1	INT16	
base + 4h	Output channel 1 and 2	1	INT16	Bit0=ch.1, Bit1=ch.2: 0=not activated (OFF), 1=activated (ON)
base + 5h	Reserved			
base + 6h	Reserved			
base + 7h	Reserved			

**1: if the value is 0, the module is not present.*

Communication parameters

RS485 communication parameters are read/written from/to the device using the following words:

RS485 Modbus communication parameters					
Modicom address	**Address**	**Description**	**Length (words)**	**Data format**	**Value**
304401	1130h	Instrument address	1	UINT16	From 1 to 247 If the value is outside the limits the device considers

					the value equal to 1
304402	1131h	Baud rate	1	UINT16	0 = 9600bps 1 = 19200bps 2 = 38400bps 3 = 115200bps All other values are considered as 0
304403	1132h	Parity	1	UINT16	0 = No parity 1 = Odd parity 2 = Even parity All other values are considered as 0
304404	1133h	Bit stop	1	UINT16	1 or 2

Read and write mode. PROFIBUS acyclic data exchange: slot 0 - index 4. Note: values are update only when the command “update communication setting” is sent or switch off and on the instrument

To write RS485 communication parameters: first write the new parameters values then execute the "External serial communication update" command (see the table below) to make actual these parameters.

Wait at least 6 seconds before communicate with the new parameters. The "External serial communication update" command is equivalent to switch OFF and ON the device.

Table of clock command

Modicom address	Address	Description	Length (words)	Data format	Value
312371	3052h	External serial communication update	1	UINT16	1 = command is executed; other values = no effect.

Write only mode. PROFIBUS acyclic data exchange: slot 5 - from index 0 to index 2.

Profibus communication parameter is read/written from/to the device using the following word:

RS485 Profibus communication parameters

Modicom address	Address	Description	Length (words)	Data format	Value
304405	1134h	Instrument address	1	UINT16	From 2 to 126 (default = 126)

Read and write mode. PROFIBUS acyclic data exchange not available

Ethernet communication parameters are read/written from/to the device using the following words:

Modbus TCP/IP communication parameters

Modicom address	Address	Description	Length (words)	Data format	Value
304433	1150h	IP address (**A**.B.C.D)	1	UINT16	From 0 to 255 All other values
304434	1151h	IP address (A.**B**.C.D)	1	UINT16	

304435	1152h	IP address (A.B.**C**.D)	1	UINT16	are considered as 255
304436	1153h	IP address (A.B.C.**D**)	1	UINT16	
304437	1154h	Subnet mask (**A**.B.C.D)	1	UINT16	
304438	1155h	Subnet mask (A.**B**.C.D)	1	UINT16	
304439	1156h	Subnet mask (A.B.**C**.D)	1	UINT16	
304440	1157h	Subnet mask (A.B.C.**D**)	1	UINT16	
304441	1158h	Default gatway (**A**.B.C.D)	1	UINT16	
304442	1159h	Default gatway (A.**B**.C.D)	1	UINT16	
304443	115Ah	Default gatway (A.B.**C**.D)	1	UINT16	
304444	115Bh	Default gatway (A.B.C.**D**)	1	UINT16	
304445	115Ch	Modbus TCP/IP port	1	UINT16	From 1 to 9999 (default = 502)

Read and write mode. PROFIBUS acyclic data exchange not available. Note: to activate the new configuration of the ethernet interface it is necessary to send the updating of ethernet configuration command or switch off and on the device.

To write TCP/IP communication parameters: first write the new parameters values then execute the "Ethernet communication configuration update" command (see the table below) to make actual these parameters.

Wait at least 6 seconds before communicate with the new parameters. The "Ethernet communication configuration update" command is equivalent to switch OFF and ON the device.

Table of ethernet parameters updating command

Modicom address	Address	Description	Length (words)	Data format	Value
312372	3053h	Ethernet communication configuration update	1	UINT16	1 = command is executed; other values = no effect.

Write only mode. PROFIBUS acyclic data exchange not available.

Internal counters

Table of time command

Modicon address	Address	Description	Length (words)	Data format	Value
401281	0500h	Active energy *1	4	UINT64	kWh
401285	0504h	Reactive energy *1	4	UINT64	kVARh
401289	0508h	Number of starts *1	2	UINT32	Counter value
401293	050Ah	Starts per hours (ANSI 66) *1	2	UINT32	Number of starts during the observation period (see ANSI 66)
401297	050Ch	Total running hours	2	UINT32	Hours
401301	050Eh		2	UINT32	Seconds
401305	0510h	Reserved			
401309	0512h	Reserved			

401317	0514h	Estimated time before trip (ANSI 49)	2	UINT32	Seconds
401321	0516h	Estimated time before restart (ANSI 66) [*1]	2	UINT32	Seconds
401325	0518h	Partitial running hours [*1]	2	UINT32	Hours
401329	051Ah		2	UINT32	Seconds
401333	051Ch	Reserved	2	UINT32	
401337	051Eh	Reserved			
401341	0520h	Counter #1	2	UINT32	Counter #1 value
401345	0522h	Counter #2	2	UINT32	Counter #2 value
401349	0524h	Timer #1	2	UINT32	Timer #1 value
401353	0526h	Timer #2	2	UINT32	Timer #2 value

Write only mode. PROFIBUS acyclic data exchange: slot 8 - index 3.
**1: the counter is resettable by command (see "Table of internal reset command").*

Maximum variables

Modicom address	Physical address	Length (words)	Instantaneous variable		Data format
401321	0528h	2	I_{1SMAX}	Max start current 1	32 bit IEEE754
400323	052Ah	2	I_{2SMAX}	Max start current 2	32 bit IEEE754
400325	052Ch	2	I_{3SMAX}	Max start current 3	32 bit IEEE754

Read only mode with function code 03 and 04. PROFIBUS acyclic data exchange: slot 8 - index 2.

Table of internal reset commands

Modicon address	Address	Description	Length (words)	Data format	Value
313569	3500h	Active energy reset	1	UINT16	1 = command is executed; other values = no effect.
313570	3501h	Reactive energy reset	1	UINT16	
313571	3502h	Number of starts reset	1	UINT16	
313572	3503h	Partitial running hours reset	1	UINT16	
313573	3504h	Reserved	1	UINT16	
313574	3505h	TCU reset [*1]	1	UINT16	
313575	3506h	Starts per hours (ANSI 66) reset	1	UINT16	
313576	3507h	Minimum time between starts ANSI 66 reset	1	UINT16	
313577	3508h	Minimum time from last stop ANSI 66 reset	1	UINT16	
313578	3509h	Max. start phase 1, 2 and 3 currents reset [*2]	1	UINT16	
313579	350Ah	Latched virtual alarm reset [*3]	1	UINT16	

Write only mode. PROFIBUS acyclic data exchange: slot 5 - from index 39 to index 49.
**1: this command resets to zero the TCU variable value; *2: this command resets to zero "Max. start current 1", "Max. start current 2" and "Max. start current 3"; *3: this command is equivalent to digital reset (latch) function.*

Date and time management

Date and time are read/written from/to the device using the following temporary words:

Clock setup

Modicon address	Address	Description	Length (words)	Data format	Value
304355	1102h	Clock calendar: year	1	UINT16	From 2009 to 2099
304356	1103h	Clock calendar: month	1	UINT16	From 1 to 12
304357	1104h	Clock calendar: day	1	UINT16	From 1 to 31
304358	1105h	Clock time: hours	1	UINT16	From 0 to 23
304359	1106h	Clock time: minutes	1	UINT16	From 0 to 59
304360	1107h	Clock time: seconds	1	UINT16	From 0 to 59

Read and write only mode. PROFIBUS acyclic data exchange: slot 0 - index 2.

To read data and time: first execute the "Get clock values" command (see the table below) then read the 6 temporany words: when the command is executed, the device stores the actual date and time in these words.

To write data and time: first write the new date and time in the 6 temporany words then execute the "Set clock values" or "Set clock values with hour and minutes" command to make actual date/time.

Table of clock command

Modicon address	Address	Description	Length (words)	Data format	Value
312369	3050h	Get clock values	1	UINT16	1 = command is executed; other values = no effect.
312370	3051h	Set clock values *1	1	UINT16	1 = set data & time; 2 = set only time

Write only mode. PROFIBUS acyclic data exchange: slot 5 - from index 0 to index 2.
**1: use this command for sync without generate any events.*

Table of time command

Modicon address	Address	Description	Length (words)	Data format	Value
312417	3080h	Set clock values with hour and minute *1	1	UINT16	1 = command is executed; other values = no effect.

Write only mode. PROFIBUS acyclic data exchange: slot 5 - index 10.
**1: use this command for sync without generate any events;*

Virtual inputs

Virtual input registers						
Modicon address	**Physical address**	**Length (words)**	**Description**		**Value**	**Data format**
412801	3200h	1	VIN_1	Virtual input 1	0 = OFF; 1 = ON (other value don't take effect)	UINT16
412802	3201h	1	VIN_2	Virtual input 2		UINT16
412803	3202h	1	VIN_3	Virtual input 3		UINT16
412804	3203h	1	VIN_4	Virtual input 4		UINT16
412805	3204h	1	VIN_5	Virtual input 5		UINT16
412806	3205h	1	VIN_6	Virtual input 6		UINT16
412807	3206h	1	VIN_7	Virtual input 7		UINT16
412808	3207h	1	VIN_8	Virtual input 8		UINT16
412809	3208h	1	VIN_9	Virtual input 9		UINT16

Write only mode. PROFIBUS acyclic data exchange: slot 5 - from index 29 to 38

Virtual alarms status

Virtual alarm status registers				
Modicon address	**Physical address**	**Length (words)**	**Description**	**Data format**
316385	4000h	1	Block function status from 1 to 16: each bit from 1 to 16 represents the alarm status	UINT16
316386	4001h	1	Block function status from 17 to 32: each bit from 17 to 32 represents the alarm status	UINT16

Read only mode. PROFIBUS acyclic data exchange not available, please use cyclic data exchange (see GSD file)

Datalogger system

Database logging

The data base is a file with 10000 records (from index 0000 to 9999). The record is organized in 45 words as illustrated in table "Database record organisation". The data base has a FIFO management system and uses two reference record numbers to identify the first record available (Ref_A) and the last record stored (Ref_B). If $Ref_B > Ref_A$, the records valid are from Ref_A+1 to Ref_B, if $Ref_A > Ref_B$, the records valid are from Ref_A+1 to 9999 and from 0 to Ref_B.

Database with $Ref_B > Ref_A$

0	1	2	3	4	5	...	...	...	...	...	...	...	9996	9997	9998	9999
			Ref_A	Valid records												
														Ref_B		

Database with $Ref_B < Ref_A$

0	1	2	3	4	5	...	...	...	...	...	...	9996	9997	9998	9999
Valid records													Ref_A	Valid records	
			Ref_B												

The data base file is readable in Modbus or Profibus as described below:

- Modbus mode:
    1. Read the reference of the first record available (Ref_A) and the reference of the last record stored (Ref_B) using Modbus function code 03h or 04h.
    2. Read the valid records using Modbus function code 14h and sub-function code 06h. The identification file number for the data base is 0.
    3. When all records are read, write the reference number Ref_A with the value of Ref_B (Modbus function code 06h). This action executes an equivalent reset function (the new records that are added during the database reading are lost).
- Profibus mode:
    1. Read the reference of the first record available (Ref_A) and the reference of the last record stored (Ref_B) using Profibus acyclic data excenge (slot 0 index 9).
    2. Write the record number to read at slot 6 and index 0 using the acyclic data exchange (it is possible to read only one record for every request).
    3. Read the record at slot 6 and index 0 using Profibus acyclic data exchange.

Database Ref_A and Ref_B references

Modicon address	Physical address	Length (words)	Description	Data format	Values
408193	2000h	1	First record available (Ref_A)	INT16	0 … 9999 (read/write)
408194	2001h	1	Last record available (Ref_B)	INT16	0 … 9999 (read only)

Database record organisation

Address	Description	Length (words)	Data format	Value
base + 0h	Record index	1	INT16	0 … 9999
base + 1h	Date: year and month	1	INT16	LSB=Month (1…12); MSB=Year (08…50)
base + 2h	Date: day and hour	1	INT16	LSB=Hour (0 … 23); MSB=Day (01 … 31)
base + 3h	Date: minute and second	1	INT16	LSB=Second (0 … 59); MSB=Minute (0 …59)
base + 4h	Number of variables	1	INT16	Number of variables
base + 5h	Variable #1	2	32 bit IEEE 754	
base + 6h	Variable #2	2	32 bit IEEE 754	
base + 7h	Variable #3	2	32 bit IEEE 754	
base + 8h	Variable #4	2	32 bit IEEE 754	
base + 9h	Variable #5	2	32 bit IEEE 754	
base + 10h	Variable #6	2	32 bit IEEE 754	
base + 11h	Variable #7	2	32 bit IEEE 754	
base + 12h	Variable #8	2	32 bit IEEE 754	

base + 13h	Variable #9	2	32 bit IEEE 754	
base + 14h	Variable #10	2	32 bit IEEE 754	
base + 15h	Variable #11	2	32 bit IEEE 754	
base + 16h	Variable #12	2	32 bit IEEE 754	
base + 17h	Variable #13	2	32 bit IEEE 754	
base + 18h	Variable #14	2	32 bit IEEE 754	
base + 19h	Variable #15	2	32 bit IEEE 754	
base + 20h	Variable #16	2	32 bit IEEE 754	
base + 21h	Variable #17	2	32 bit IEEE 754	
base + 22h	Variable #18	2	32 bit IEEE 754	
base + 23h	Variable #19	2	32 bit IEEE 754	
base + 24h	Variable #20	2	32 bit IEEE 754	

Dataevent logging

The data event is a file with 10000 records (from index 0000 to 9999). The record is organized in 11 words as illustrated in table "Dataevent record organisation". The data event has a FIFO management system and uses two reference record numbers to identify the first record available (Ref$_A$) and the last record stored (Ref$_B$). If Ref$_B$ > Ref$_A$, the records valid are from Ref$_A$+1 to Ref$_B$, if Ref$_A$ > Ref$_B$, the records valid are from Ref$_A$+1 to 9999 and from 0 to Ref$_B$.

Dataevent with Ref$_B$ > Ref$_A$

0	1	2	3	4	5	...	...	...	...	...	...	...	9996	9997	9998	9999
			Ref$_A$	Valid records												
														Ref$_B$		

Dataevent with Ref$_B$ < Ref$_A$

0	1	2	3	4	5	...	...	...	...	...	...	9996	9997	9998	9999
Valid records													Ref$_A$	Valid records	
			Ref$_B$												

The data base file is readable in Modbus or Profibus as described below:

- Modbus mode:
  1. Read the reference of the first record available (Ref$_A$) and the reference of the last record stored (Ref$_B$) using Modbus function code 03h or 04h.
  2. Read the valid records using Modbus function code 14h and sub-function code 06h. The identification file number for the data event is 1.
  3. When all records are read, write the reference number Ref$_A$ with the value of Ref$_B$ (Modbus function code 06h). This action executes an equivalent reset function (the new records that are added during the database reading are lost).
- Profibus mode:
  1. Read the reference of the first record available (Ref$_A$) and the reference of the last record stored (Ref$_B$) using Profibus acyclic

data excenge (slot 0 index 10).

2. Write the record number to read at slot 6 and index 1 using the acyclic data exchange (it is possible to read only one record for every request).
3. Read the record at slot 6 and index 1 using Profibus acyclic data exchange.

Dataevent Ref_A and Ref_B references

Modicon address	Physical address	Length (words)	Description	Data format	Values
408195	2002h	1	First record available (Ref_A)	INT16	0 … 9999 (read/write)
408196	2003h	1	Last record available (Ref_B)	INT16	0 … 9999 (read only)

Dataevent record organisation

Address	Description	Length (words)	Data format	Value
base + 0h	Record index	1	INT16	0 … 9999
base + 1h	Date: year and month	1	INT16	LSB=Month (1…12); MSB=Year (08…50)
base + 2h	Date: day and hour	1	INT16	LSB=Hour (0 … 23); MSB=Day (01 … 31)
base + 3h	Date: minute and second	1	INT16	LSB=Second (0 … 59); MSB=Minute (0 …59)
from base + 4h to base + 00Ah	Record fields	1	INT16	See the table dataevent record fields

Dataevent record fiels

Input event				
Address	**Description**	**Length (words)**	**Data format**	**Value**
base + 4h	Type of event	1	UINT16	1=Input
base + 5h	Number of input channel	1	UINT16	From 1 to 23
base + 6h	New status	1	UINT16	**Switch, toggle or PTC input** 0 = release, 1 = activate **PT100 input** 1 = error (probe break), 0 = probe OK
Output event				
Address	**Description**	**Length (words)**	**Data format**	**Value**
base + 4h	Type of event	1	UINT16	2=Relay output
base + 5h	Number of output channel	1	UINT16	From 1 to 23
base + 6h	New status	1	UINT16	1 (OFF) = open, 0 (ON) = close
Generic event				
Address	**Description**	**Length (words)**	**Data format**	**Value**

base + 4h	Type of event	1	UINT16	3=Generic event
base + 5h	Number of output channel	1	UINT16	0=Modules connection error 1=Measurement module connection error 5=Start/stop motor 9=Module configuration error 10=Block hours greasing 11=Block hours maintenance 16=Data base logging reset 17=Fast data logger reset 18=Data event reset 19=Power OFF 20=Power ON 21=Latch reset command
base + 6h	New status	1	UINT16	**Modules conn. error** 1 = error; 0 = OK **Measurement modules conn. error** 1 = error; 0 = OK **Start/stop motor:** 1 = start; 0 = stop **Module configuration error** 1 = error; 0 = OK **Block hours greasing** 1 = locked; 0 = unlocked **Block hours maintenance** 1 = locked; 0 = unlocked **Data base logging reset** 0 = reset **Fast data logger reset** 0 = reset **Data event reset** 0 = reset **Power OFF** 0 = reset **Power ON** 0 = reset **Latch reset command** 0="Active energy" reset 1="Reactive energy" reset 2="Number of starts" reset 3="Partitial running hours" reset 4="Max motor start time (ANSI 51LR)" reset 5="TCU" reset 6="Starts per hours (ANSI 66" reset 7=Minimum time between starts ANSI 66 reset 8=Minimum time from last stop ANSI 66 reset 9=Max. start phase 1, 2 and 3 reset 10=Latched virtual alarm reset

Max/min events				
Address	**Description**	**Length (words)**	**Data format**	**Value**
base + 4h	Type of event	1	UINT16	5=Max/min event
base + 5h	Number of max/min event	1	UINT16	4=Too long start 6=Max current phase 1 7=Max current phase 2 8=Max current phase 3
base + 6h	Value	2	32 bit IEEE 754	Actual value
Virtual alarm events				
Address	**Description**	**Length (words)**	**Data format**	**Value**
base + 4h	Type of event	1	UINT16	6=Virtual alarm event
base + 5h	Number of virtual alarm	1	UINT16	From 1 to 32
base + 6h	New status	1	UINT16	1 = Alarm active, 0 = Alarm not active

Fast data logger

The data fast logger is a file with 10000 records (from index 0000 to 9999). The record is organized in 45 words as illustrated in table "Fast datalogger record organisation". The data fast logger has a FIFO management system and uses two reference record numbers to identify the first record available (Ref_A) and the last record stored (Ref_B). If $Ref_B > Ref_A$, the records valid are from Ref_A+1 to Ref_B, if $Ref_A > Ref_B$, the records valid are from Ref_A+1 to 9999 and from 0 to Ref_B.

Fast datalogger with $Ref_B > Ref_A$

0	1	2	3	4	5	...	...	...	...	...	...	...	9996	9997	9998	9999
			Ref_A	Valid records												
														Ref_B		

Fast datalogger with $Ref_B < Ref_A$

0	1	2	3	4	5	...	...	...	...	...	...	9996	9997	9998	9999
Valid records													Ref_A	Valid records	
			Ref_B												

The data fast logger file is readable in Modbus or Profibus as described below:

- Modbus mode:
  1. Read the reference of the first record available (Ref_A) and the reference of the last record stored (Ref_B) using Modbus function code 03h or 04h.
  2. Read the valid records using Modbus function code 14h and sub-function code 06h. The identification file number for the data fast logger is 2.
  3. When all records are read, write the reference number Ref_A with

the value of Ref_B (Modbus function code 06h). This action executes an equivalent reset function (the new records that are added during the database reading are lost).

- Profibus mode:
  1. Read the reference of the first record available (Ref_A) and the reference of the last record stored (Ref_B) using Profibus acyclic data excenge (slot 0 index 11).
  2. Write the record number to read at slot 6 and index 2 using the acyclic data exchange (it is possible to read only one record for every request).
  3. Read the record at slot 6 and index 2 using Profibus acyclic data exchange.

Fast datalogger Ref_A and Ref_B references

Modicon address	Physical address	Length (words)	Description	Data format	Values
408197	2004h	1	First record available (Ref_A)	INT16	0 … 9999 (it is possible the write and read mode access)
408198	2005h	1	Last record available (Ref_B)	INT16	0 … 9999 (it is possible only the read mode access)

Fast datalogger record organisation

Address	Description	Length (words)	Data format	Value
base + 0h	Record index	1	INT16	0 … 9999
base + 1h	Date: year and month	1	INT16	LSB=Month (1…12); MSB=Year (08…50)
base + 2h	Date: day and hour	1	INT16	LSB=Hour (0 … 23); MSB=Day (01 … 31)
base + 3h	Date: minute and second	1	INT16	LSB=Second (0 … 59); MSB=Minute (0 …59)
base + 4h	Number of variables	1	INT16	Number of variables
base + 5h	Variable #1	2	32 bit IEEE 754	
base + 6h	Variable #2	2	32 bit IEEE 754	
base + 7h	Variable #3	2	32 bit IEEE 754	
base + 8h	Variable #4	2	32 bit IEEE 754	
base + 9h	Variable #5	2	32 bit IEEE 754	
base + 10h	Variable #6	2	32 bit IEEE 754	
base + 11h	Variable #7	2	32 bit IEEE 754	
base + 12h	Variable #8	2	32 bit IEEE 754	
base + 13h	Variable #9	2	32 bit IEEE 754	
base + 14h	Variable #10	2	32 bit IEEE 754	
base + 15h	Variable #11	2	32 bit IEEE 754	
base + 16h	Variable #12	2	32 bit IEEE 754	
base + 17h	Variable #13	2	32 bit IEEE 754	
base + 18h	Variable #14	2	32 bit IEEE 754	
base + 19h	Variable #15	2	32 bit IEEE 754	

base + 20h	Variable #16	2	32 bit IEEE 754	
base + 21h	Variable #17	2	32 bit IEEE 754	
base + 22h	Variable #18	2	32 bit IEEE 754	
base + 23h	Variable #19	2	32 bit IEEE 754	
base + 24h	Variable #20	2	32 bit IEEE 754	

Datalogger reset

Use the following commands to reset the three databases:

Table of datalogger reset command

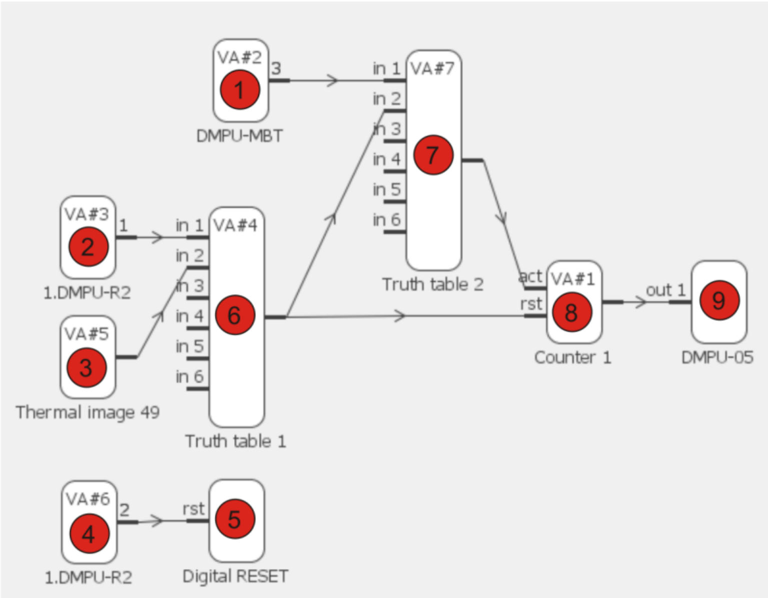
Modicon address	Address	Description	Length (words)	Data format	Value
313825	3600h	Reset database	1	UINT16	1 = command is executed; other values = no effect.
313826	3601h	Reset data event	1	UINT16	
313827	3602h	Reset data fast	1	UINT16	

Write only mode. PROFIBUS: slot 5 - from index 50 to index 52.

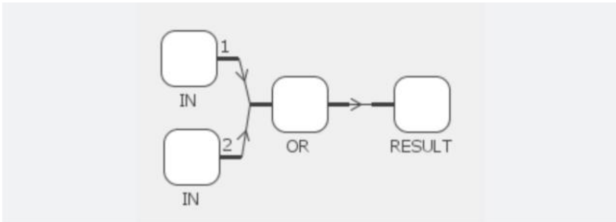
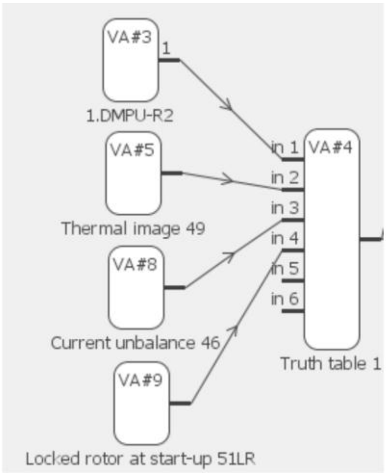
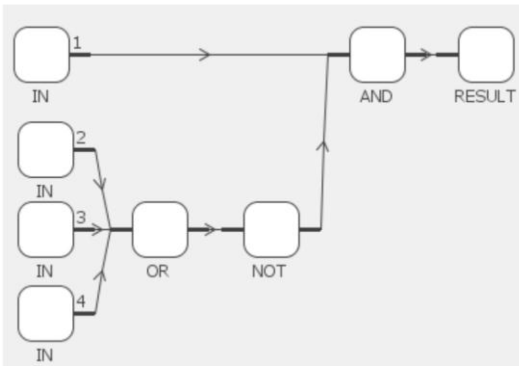
Configuration Examples

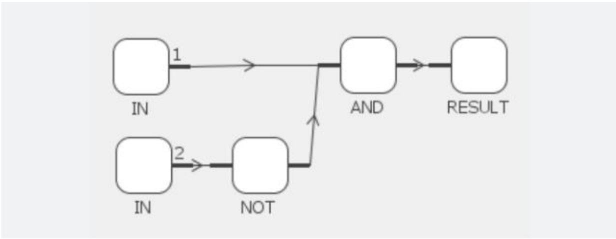
Direct starter

In this example a basic DOL starter with start and stop pushbuttons and a thermal image 49 alarm is built. The used blocks and connections are the following:



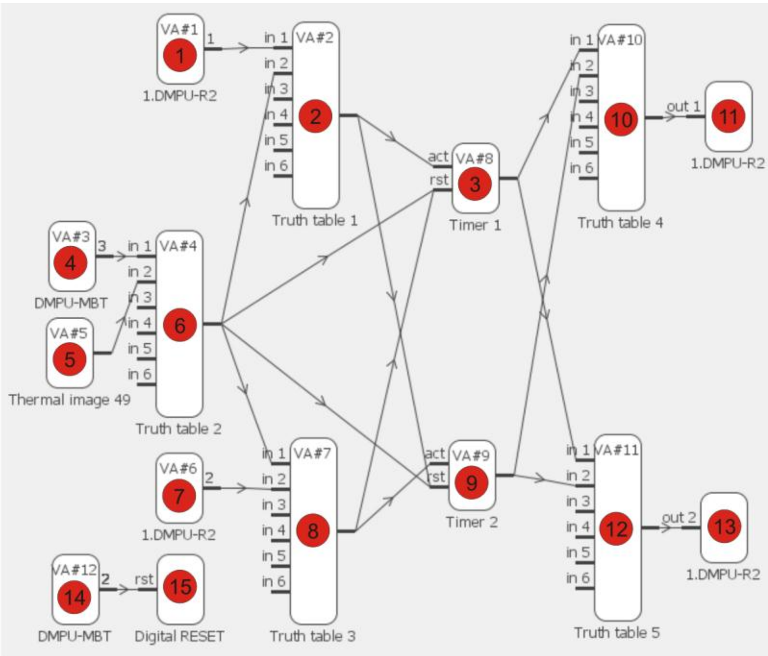
ID	Block	Description
1	Digital input	Start input: • Alarm # enable: enabled • Input type: switch • Type of input: active when closed • On delay Al. #: 0 s
2	Digital input	Stop input: • Alarm # enable: enabled • Input type: switch • Type of input: active when closed • On delay Al. #: 0 s
3	Thermal image 49	Thermal image alarm: • Enable function ANSI49: enabled and latched. This kind of alarm is usually latched otherwise it may cause oscillations due to the fact that the motor cools down when stopped so alarm might release. • set the other values according to the motor features
4	Digital input	Used to release the ANSI 49 Thermal image alarm (reset alarm as it is latched). • Alarm # enable: enabled

		• Input type: switch • Type of input: active when closed • On delay Al. #: 0 s
5	Digital reset	Internal output to control the latch reset
6	Logic function	To stop the motor in case of stop input command or ANSI 49 Thermal image alarm trip. The graphic representation is the following:  It is possible to use other protection functions as thermal image adding them to this logic function:  The logic function is as follows:  In this case the motor is stopped when at least one protection function trips.
7	Logic function	To inhibit the start input if the thermal image alarm is tripped or the stop motor input is activated. The graphic representation is the following:

		
8	Counter	It is used as a flip-flop. The setpoint is set to 1 so if a signal comes from the act terminal it activates the output (motor starts), while if it comes from the rst terminal it de-activates the output (motor stops). • Alarm # enable: enabled • Set counts of Al. #: 1 count
9	Relay output	Output relay for motor start/stop • O# working mode: NO

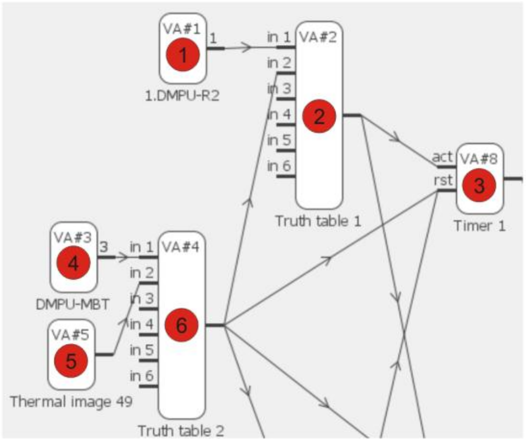
Reversing starters

In this example a reversing starter with forward start, reverse start and stop pushbuttons and a thermal image 49 alarm is built. The used blocks and connections are the following:

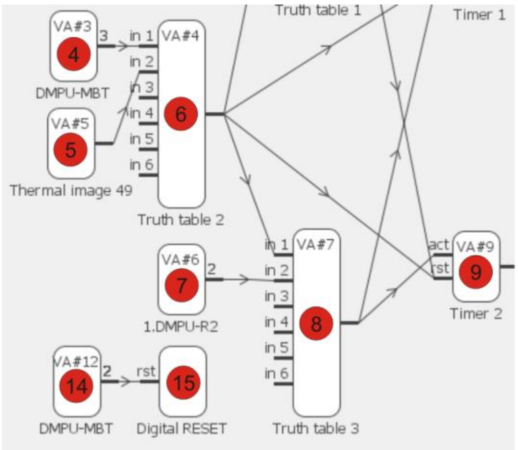


The used blocks are grouped in three groups with the following features:

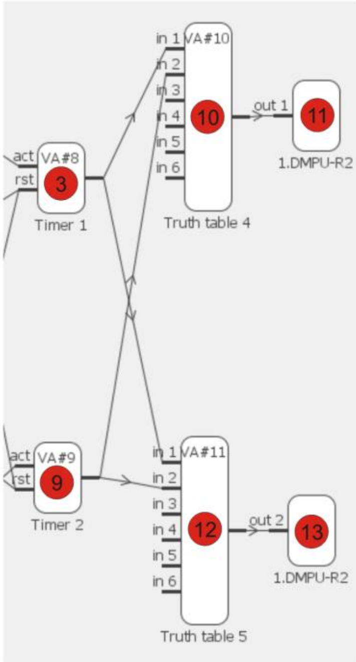
1. Forward starter group: this group is used for forward start-stop motor function. The blocks are the following:



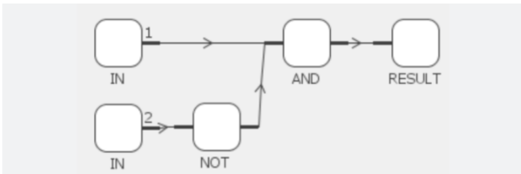
2. Reverse starter group: this group is used for reverse start-stop motor function. The blocks are the following:

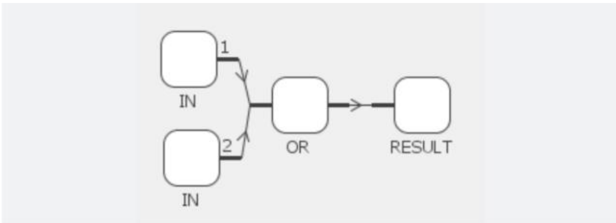
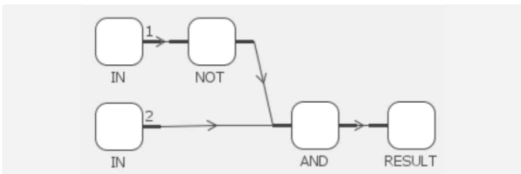


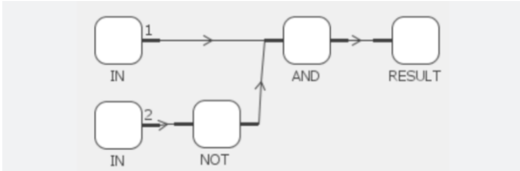
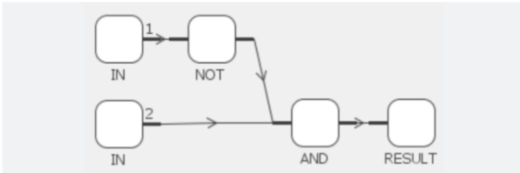
3. Forward-reverse interlock group: this group is used to avoid the simultaneous activation of forward and reverse outputs. The blocks are the following:



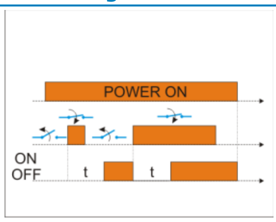
Group blocks description:

ID	Block	Description
1	Digital input	Start forward input: • Alarm # enable: enabled • Input type: switch • Type of input: active when closed • On delay Al. #: 0 s
2	Logic function	To inhibit the start forward input if the thermal image alarm is tripped or the stop motor input is activated. The graphic representation is the following: 
3	Timer	It is used as forward motor start introducing a delay time. The delay time is neccessary during the reversing operation (moving from reverse to forward rotating without using the motor stop) to minimize the dinamic overloads deriving from motion reversals. If a signal comes from the "act" terminal it activates the output (motor forward starts) after the set time, while if it comes from the "rst" terminal it de-activates the output (motor stops). • Alarm # enable: enabled • Set time of Al. #: 3 seconds (set this value according to the motor features)
4	Digital input	Stop input:

		• Alarm # enable: enabled • Input type: switch • Type of input: active when closed • On delay Al. #: 0 s
5	Thermal image 49	Thermal image alarm: • Enable function ANSI49: enabled and latched. This kind of alarm is usually latched otherwise it may cause oscillations due to the fact that the motor cools down when stopped so alarm might release. • set the other values according to the motor features
6	Logic function	To control the motor stop in case of stop input command or ANSI 49 Thermal image alarm trip. The graphic representation is the following: 
7	Digital input	Start reverse input: • Alarm # enable: enabled • Input type: switch • Type of input: active when closed • On delay Al. #: 0 s
8	Logic function	To inhibit the start reverse input if the thermal image alarm is tripped or the stop motor input is activated. The graphic representation is the following: 
9	Timer	It is used as reverse motor start introducing a delay time. The delay time is neccessary during the reversing operation (moving from forward to reverse rotating without using the motor stop) to minimize the dinamic overloads deriving from motion reversals. If a signal comes from the "act" terminal it activates the output (motor reverse starts) after the set time, while if it comes from the "rst" terminal it de-activates the output (motor stops). • Alarm # enable: enabled • Set time of Al. #: 3 seconds (set this value according to the motor features)
10	Logic function	To ensure the interlock function between forward and reverse operations: forward operations is inhibited if the motor is stopped or running in reverse condition. The graphic representation is the following:

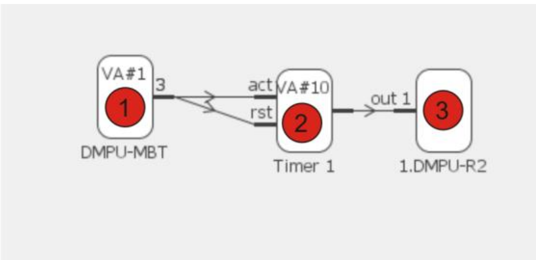
		
11	Relay output	Output relay for motor forward operating • O# working mode: NO
12	Logic function	To ensure the interlock function between forward and reverse operations: reverse operations is inhibited if the motor is stopped or running in forward condition. The graphic representation is the following: 
13	Relay output	Output relay for motor reverse operating • O# working mode: NO
14	Digital input	Used to release the ANSI 49 Thermal image alarm (reset alarm as it is latched). • Alarm # enable: enabled • Input type: switch • Type of input: active when closed • On delay Al. #: 0 s
15	Digital reset	Internal output to control the latch reset

ON delay timer



Timer ON delay function

In this example an ON delay timer function is built. The used blocks and connections are the following:



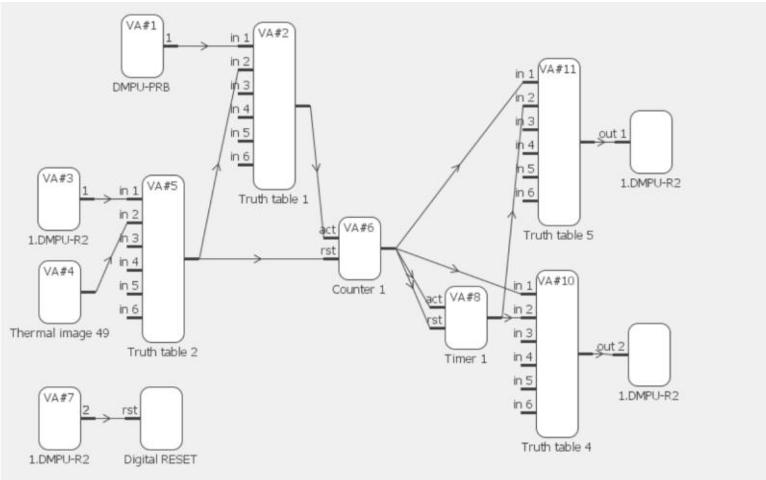
ID	Block	Description
1	Digital input	Trigger input: • Alarm # enable: enabled • Input type: toggle • Type of input: active when closed

		• On delay Al. #: 0 s
2	Timer	To enable the delay time for ON delay function at the trigger command and obtain the function activation after the set delay time. • Alarm # enable: enabled • Set time of Al. #: 5 s (it is the actual delay time and will be set according to the application needs)
3	Relay output	Output relay for timer ON delay function. • O# working mode: NO

Every time the trigger input is activated the timer is reset and restarted; when the delay time (5s) is expired the timer turns OFF but the logic function turns ON until the trigger input is reactivated.

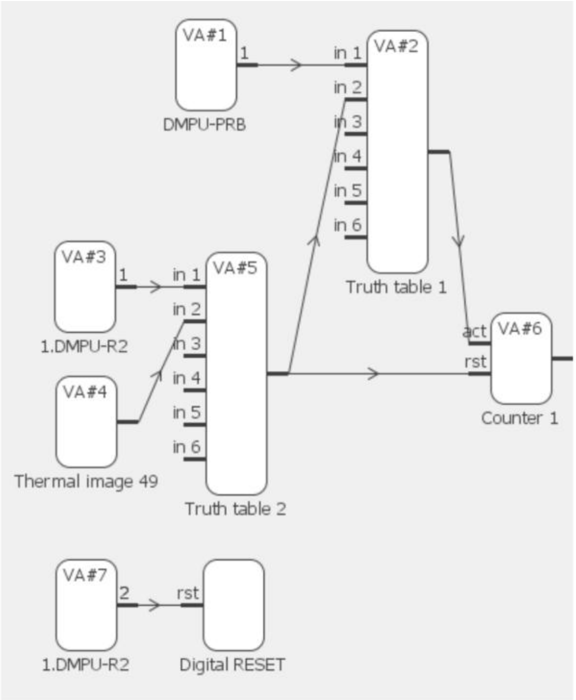
Star-delta starter

In this example a star-delta starter with start/stop pushbuttons and a thermal image 49 alarm is built. The used blocks and connections are the following:

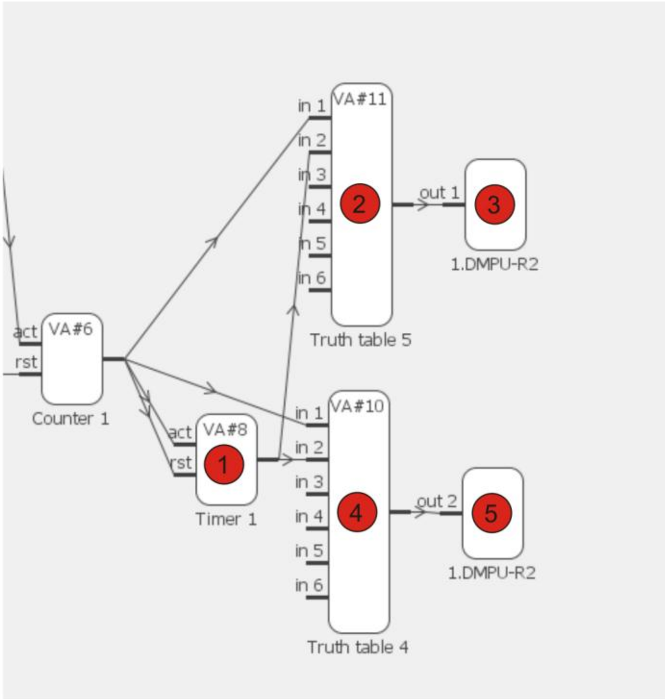


The used blocks are grouped in two groups with the following features:

1. Direct starter group: see the direct starter example for the block description; this group is used for start-stop motor function. The blocks are the following:

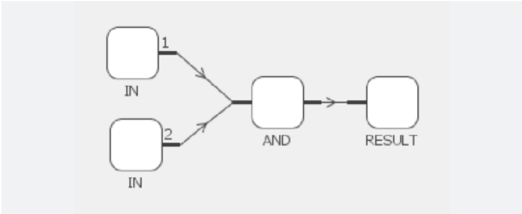
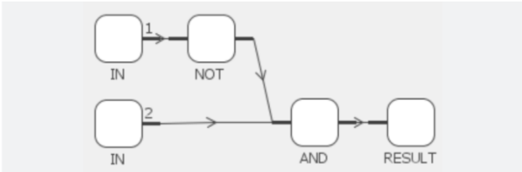


2. Start-delta interlock group: see the following table for the blocks description; this group is used to avoid the simultaneous activation of the star and delta outputs. The blocks are the following:



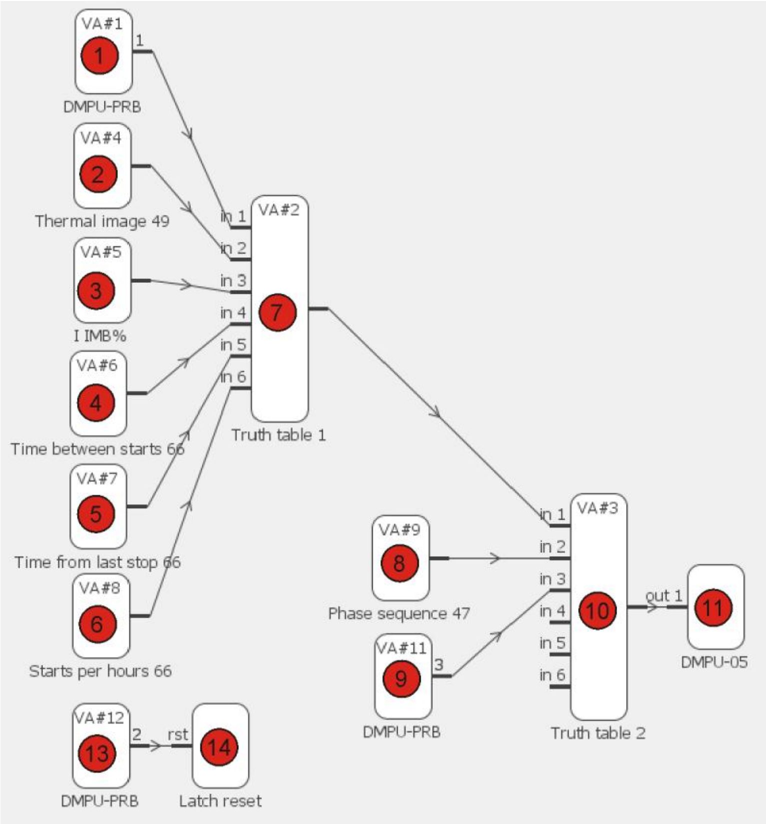
Start-delta interlock group blocks description:

ID	Block	Description
1	Timer	It is used for star-delta switching in a set time. Set this time according to the motor load.

2	Logic function	To ensure the interlock function between star and delta operations: delta is inhibited if the motor is stopped or running in star condition. The graphic representation is the following: 
3	Relay output	Output relay for motor delta operation • O# working mode: NO
4	Logic function	To ensure the interlock function between star and delta operating: the star operating is inhibited if the motor is stopped or running in delta condition. The graphic representation is the following: 
5	Relay output	Output relay for motor star operating • O# working mode: NO

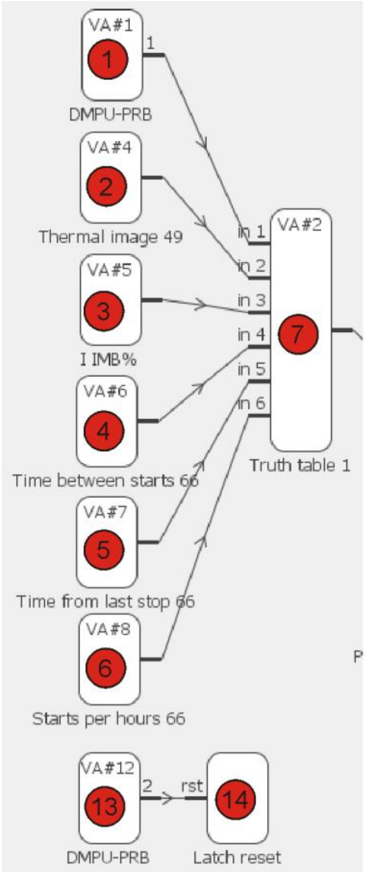
Emergency start

In this example an emergency start with auxiliary start pushbuttons (emergency start) is built. The used blocks and connections are the following:

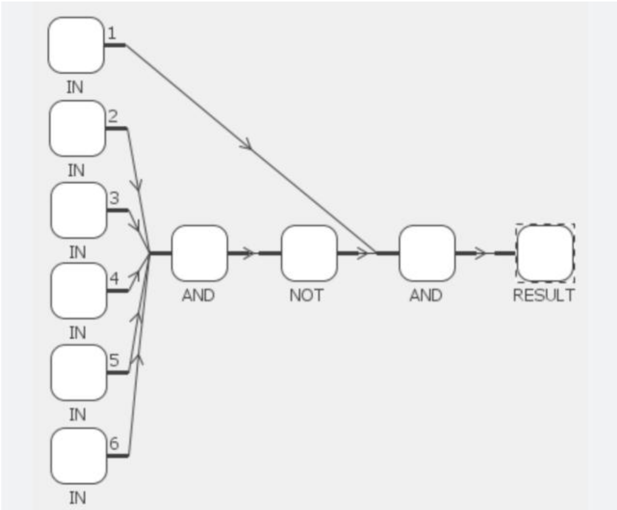


The used blocks are grouped in two groups with the following features:

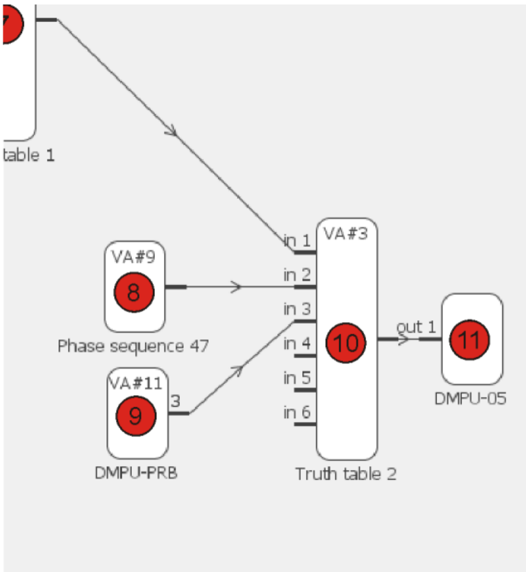
- 1° alarm group: all the alarms, which can trip during normal running condition, are summarized on this group. Normal running condition is performed when the motor is started by the normal start input. A connection example is the following:

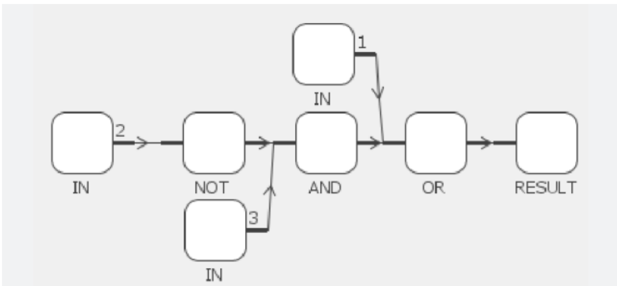


ID	Block	Description
1	Digital input	Normal start input: • Alarm # enable: enabled • Input type: toggle • Type of input: active when closed • On delay Al. #: 0 s
2	Thermal image 49	Thermal image alarm: • Enable function ANSI49: enabled and latched. This kind of alarm is usually latched otherwise it may cause oscillations due to the fact that the motor cools down when stopped so alarm might release. • set the other values according to the motor features
3	Current imbalance	Current imbalance alarm: set the other values according to the motor features
4	Time between starts 66	Time between starts alarm: set the minimum time between two starts according to the motor features. Use this function to lock the starts too close.
5	Time from last stop 66	Time from last stop alarm: set the minimum time between last stop and next start according to the motor features. Use this function to lock the restart too close to last stop.
6	Starts per hours 66	Starts per hours alarm: set the allowed number of starts and the observation

		interval time according to the motor features. Use this function to limit the frequent starts (that can cause the motor overheating).
7	Logic function	When the Thermal image 49, Inverse sequence current 46, Time between starts 66, Time before last stop 66 or Starts per hours 66 alarms trip, the motor is stopped and the normal start is inhibited; if the motor re-starting is required (by passing the previous alarms), use the emergency start as described in second group alarm. The logic function appears as follows (the normal start is available when the start input is activated and all other alarms are de-active): 
13	Digital input	Used to release the latched alarms (reset alarm as it is latched). • Alarm # enable: enabled • Input type: switch • Type of input: active when closed • On delay Al. #: 0 s
14	Digital reset	Internal output to control the latch reset

- 2° alarm group: all the alarms, which can trip during special start condition, are summarized on this group. The special start condition is performed when the motor is stopped by an alarm of the first group and immediate restart is requied. An connection example is the following:



ID	Block	Description
8	Phase sequence 47	Phase sequence alarm: set the values according to the motor features. Use this configuration when the motor reverse rotation is not allowed.
9	Digital input	Emergency start input: • Alarm # enable: enabled • Input type: toggle • Type of input: active when closed • On delay Al. #: 0 s
10	Logic function	The emergency start is inhibited only if Phase sequence or Locked rotor at start-up alarms are tripped: for example use this configuration when the motor reverse rotation is not allowed during both normal and special condition. The logic function appears as follows: 
11	Relay output	Output relay for motor start/stop • O# working mode: NO