PROFINET MANUAL

VECTOR STEP POSITIONERS

PROFI NET®

Member of PROFIBUS • PROFINET

Subject to technical and layout alterations.

Index

NOTES ON SAFETY

Products for automation manufactured by AEC must be handled, installed and maintained only by skilled and authorized personnel, that must be qualified and instructed to install components for automation. Devices must be installed only for the purposes described in the user's guide. The installer should pay particular attention to potential risks caused by mechanical and electrical hazards.

It is very important that all applications and installations meet all applicable safety requirements.

The installers must take responsibility to verify their knowledge and understanding of all applicable safety standards.

Installations which are not complying with safety requirements can damage equipment and injure the user.

AEC s.r.l. will not be liable and will not take any responsability for damages caused by products handled or installed improperly, or if the customer have given permission or performed modifications and/or repairs not authorized from AEC s.r.l.

AEC's motion control equipment are high-performances devices for automation, able to producing high forces and rapid movements.

Pay high attention, in particular during installation and development of applications.

Use properly sized equipments for the type of application.

AEC's devices must be considered as components for automation. They are sold as end-user products, and must be installed only by qualified personnel, in accordance with all applicable safety requirements.

Skilled staff must be able to recognize possible dangers that may result from programming, modifying parameter's values and, generally, that may result from using mechanical, electric and electronic equipment.

The drive must be installed in closed cabinets, so that any parts thereof is not reachable while system is powered on.

AEC s.r.l strongly recommends to always follow safety requirements and security rules. Failure to follow this instruction may cause and/or injuries.

General precautions

- The images contained in this manual are for demonstration purposes, and may differ from the products received.
- This manual is subject to changes due to improvement of the products, modification of specifications, or manual thereof improvement.
- AEC s.r.l. is not responsible for any damage to property or injury that could result from improper installation and/or not authorized modification to products.

AEC's drive systems are products for general use that conform to the state of the art in technology and are designed to prevent any dangers. However, drives and drive controllers that are not specifically designed for safety functions are not approved for applications where the functioning of the drive could endanger persons. ***The possibility of unexpected or unbraked movements can never be totally excluded without additional safety equipment****. For this reason personnel must never be in the danger zone of the drives unless additional suitable safety equipment prevents any personal danger. This applies to operation of the machine during production and also to all service and maintenance work on drives and the machine. The machine design must ensure personal safety. Suitable measures for prevention of property damage are also required.*

To prevent personal injury and damage to property, damaged drive systems must not be installed. Changes and modifications of the drive systems are not permitted, and if made all no warranty and liability will be accepted.

Maintenance and Inspection

To ensure a proper and satisfactory performace of the drives and the motors, equipments and installations need periodic inspections and checks.

Notes for maintenance pesonnel

After shutdown, the internal capacity will remain charged, at high voltages, for a short period of time.
Wait at least 10 minutes after PWR led goes off, before working on the device.

Drives and motors can reach high temperatures during functioning, therefor it is recommended to wait for them to cool before touching any of their surfaces. in all cases, be careful.

Never plug or unplug any connector when power is connected.

Control cycles checklist

Correct operating conditions:

Operating temperature : 30° C (annual average)

Hours of work : 24 hours per day

Periodically check the correct operation of the equipments by following this checklist:

Type of inspection	Frequency	Checklist
Weekly check	Weekly	1. Operating temperature, humidity, dust, particles or foreign matter 2. Vibrations or not standard noises 3. Main and auxiliary supply voltage 4. Odors 5. Obstruction of ventilation slots 6. Cleaning of drive and connectors 7. Correct insertion of connectors 8. Integrity of the cables
Periodically check	Annual	1. Verify the correct closure of the fixing screws 2. Signal malfunction or overheating

In case that operating conditions are different from the recommended ones, carry out inspections more frequently.

REVISIONS

Version	Date	Notes
Rev. 01	November 2021	First draft

COPYRIGHTS

TRADEMARKS

INFORMATION ON PROFINET

OVERVIEW

The drives Profinet SMD1204xIN, SMD2204xIN e SMD5206xIN support several application profiles based on cyclic and acyclic communication services:

- PROFIdrive v.4.1 - Standard Telegram 9
- PROFIdrive v.4.1 - Base Mode Parameter Access (Acyclic Data Exchange)

CYCLIC AND ACYCLIC SERVICES

Normally, the data exchange uses cyclic and acyclic services.
For the cyclic data, the application profiles define:

- data indipendent from the manufacturers
- specific data for the manufacturer

The fixed setting and the use of the indipendent data from the manufacturer, permit to switch between them masters of different brand.

ACYCLIC READ/WRITE SERVICES

The acyclic Read/Write services provide access to data or parameters which cannot be accessed with cyclic data exchange.

ELECTRONIC FILE DESCRIPTION

The drives mod. SMD1204xIN, SMD2204xIN e SMD5206xIN are described by a GSDML file, used by Profinet configuration tools to obtain information on the devices themselves
GSDML files and icon files of the AEC's drives can be downloaded from the website www.aec-smd.it
The GSDML file and the icons are compressed into a .zip file, that has to be decompressed in the same folder of the hard disk.

THE DEVICES MAY PUT THEMSELVES INTO OPERATION WITHOUT NOTICE
Do not alter in any way the GSDML file. The alteration of the GSDML file may cause unexpected behaviour of the drives.
Failure to observe this precaution may cause injuries or damages to devices.

CAUTION!!! Any alteration to the GSDML file will void the AEC guarantee with immediate effect.

PROFINET CHARACTERISTICS

INTRODUCTION

The following table is an overview for the PROFINET features supported by SMDyyyyxIN drives

PROFINET RT	✓
PROFINET IRT (RT_CLASS_3)	✓
Advanced Startup	✓
Legacy Startup	✓
Minimum cycle time, RT	250us (motor controlled every 1ms)
Minimum cycle time, IRT	250us (motor controlled every 1ms)
Enhanced Configuration Support	✓
Support of I&M5	✓
Simple Network Management Protocol (SNMP)	✓
MRP Client (Media Redundancy Protocol)	✓
Number of ARs / Shared Device capable	1
IO Supervisor AR	1 (only device access)
Acyclic communication	Read/Write Record
Alarm Types	Process Alarm, Diagnostic Alarm, Plug Alarm, Pull Alarm, Return of Submodule Alarm
Identification & Maintenance	I&M0-5
Topology recognition	LLDP, SNMP V1, MIB2, PDEV
Media Redundancy	MRP client
Additional supported features	DCP, 802.1q Priority
Data rate / duplex	100 MBit/s, Full Duplex
Data transport layer	Ethernet II, IEEE 802.3
PROFINET IO specification	V2.35

TECHNICAL DATA

Type	Ethernet network
Cable	Ethernet CAT. 5e
Function	Real-time motion control, setup and parameterization, programming, diagnostics
Protocol	PROFIdrive according to Profile Drive Technology version 4.1, May 2006 (IEC 61800-7)
Error checking	Checksum
Supported Masters	Class 1, Class 2
Application Class	3 (Single axis positioning drive, with local motion control)
Number of port	2

Restrictions

Following restriction apply:

- RT over UDP not supported
- DHCP is not supported
- Fast Startup iso not supported
- Shared Inputs are not supported
- Multicast cammunication not supported
- Only 1 Input-CR and 1 Output-CR per AR is supported
- System Redundancy (SR-AR) and Configuration-in-Run (CiR) are not supported
- The amount of configured IO-data influences the minimum cycle time that can be reached.

Certification

The SMDyyyyxIN device was tested with the official PROFINET IO Test Bundle of PI (Release 2017-04-05) at ComDeC test lab (Würzburger Straße 121, 90766 Fürth, Germany).

Configuration file

GSDML file	Interface	Profile
GSDML-V2.42-AECsrl-ProfiNetDrives-20220218.xml	Profinet	PROFIdrive V4.1

SMD2204xIN 3 axis board, ProfiNet Slots Structure

Slot 0 (API=0)	Slot 1 (API = 0x3A00 PROFIDrive)			Slot 2 (API = 0x3A00 PROFIDrive)			Slot 3 (API = 0x3A00 PROFIDrive)		
Subslot 0	Subslot 0	Subslot 1	Subslot 2	Subslot 0	Subslot 1	Subslot 2	Subslot 0	Subslot 1	Subslot 2
		Module Access Point (MAP)	Standard Telegram x (submodule ID = PROFIDrive telegram number)		Module Access Point (MAP)	Standard Telegram x (submodule ID = PROFIDrive telegram number)		Module Access Point (MAP)	Standard Telegram x (submodule ID = PROFIDrive telegram number)
		Contains parameters Access Point and alarm			Contains parameters Access Point and alarm			Contains parameters Access Point and alarm	
P-Device	Drive Object 1			Drive Object 2			Drive Object 3		

SMD1204xIN
SMD5206xIN **1AxLE board, PrifiNet slots structure**

GSDML FILE TELEGRAMS

Overview

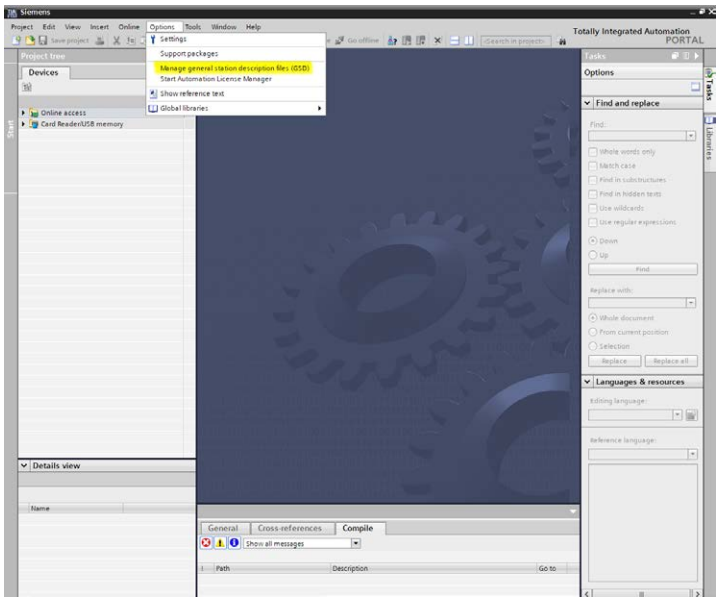
On Profinet, the AEC's devices permit to configure the cyclic communication telegrams.

There are several types of messages, which can be used according to the type of control that must be obtained:

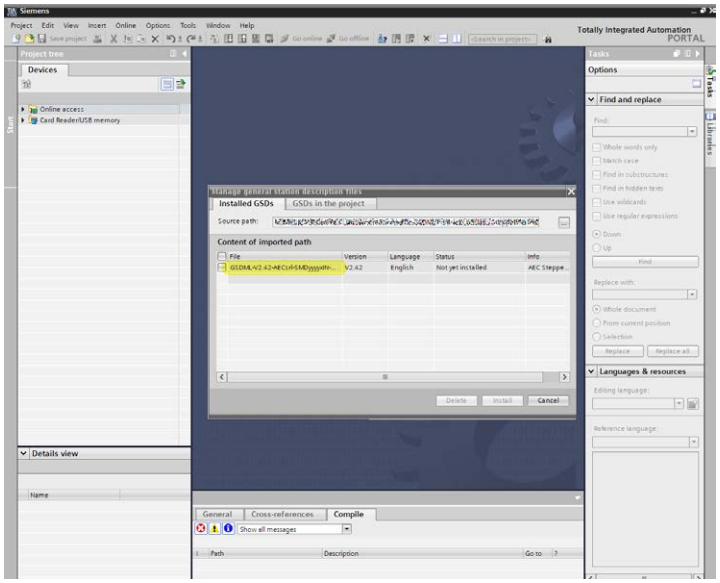
- Positioning interface (Program submode)
- x set-interface, 32 bit (position setpoint interface)
- Positioning interface (Program submode + MDI submode)

During the configuration, it is necessary to choose according to what is stated the following table.

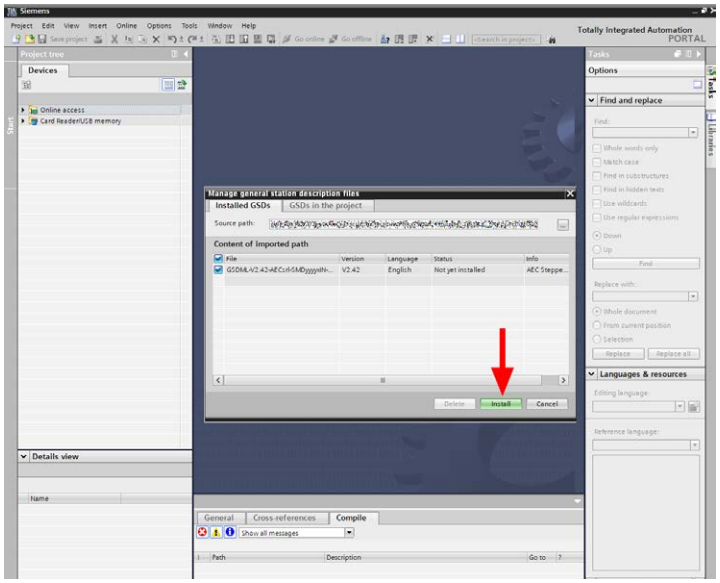
Telegram type	Description
Telegram 9	Positioning interface (Program submode plus MDI submode) *The MDI permits to access directly to the motion command interface to achieve direct movements that are not configurated in the traversing blocks.* *Using the signals MDI_TARPOS, MDI_ACC, MDI_DEC e MDI_VELOCITY it is possible to parameterize a new movement.*
Telegram 109 + External Encoder + IO interface + Mot_Enc + Analog IO	Positioning interface (Program submode plus MDI submode) *The MDI permits to access directly to the motion command interface to achieve direct movements that are not configurated in the traversing blocks.* *Using the signlas: MDI_TARPOS, MDI_ACC, MDI_DEC e MDI_VELOCITY it is possible to parameterize a new movement.* *In addition to the standard signals it is possible to read in cyclic way the current position of the external encoder and the input status, and to set the output status.*



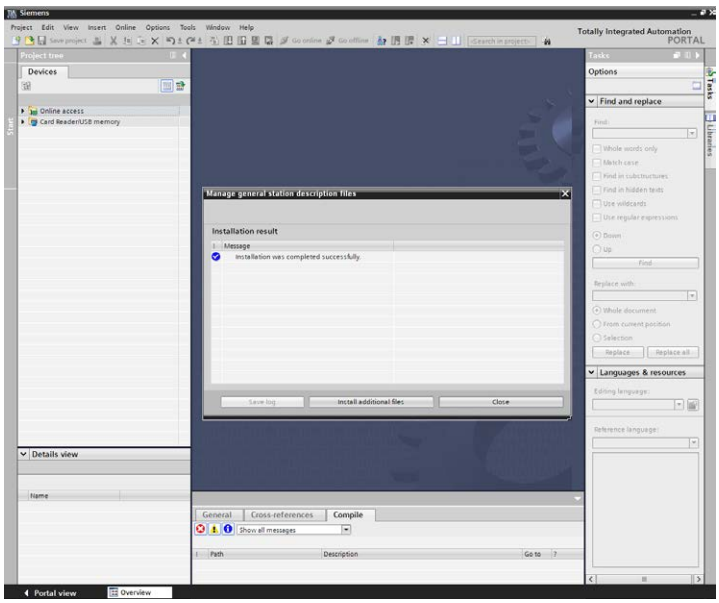
To install the GSDML file, use the command “Manage general station description files (GSD)” from the menu “Options” of the Hardware configurator.



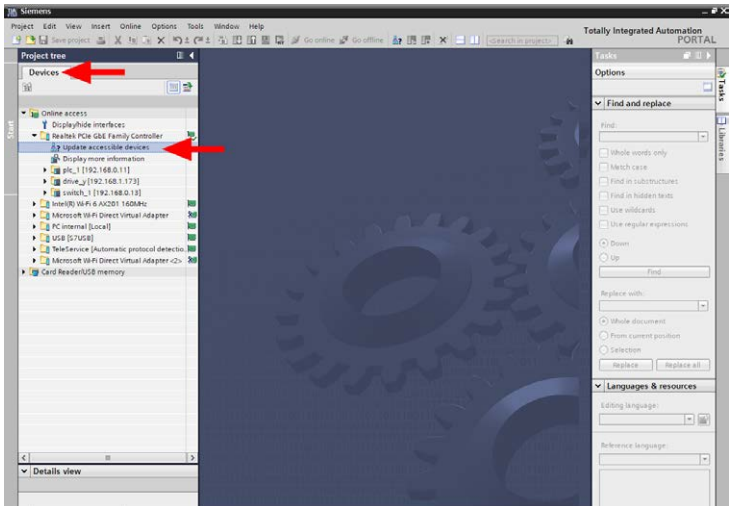
Select the local directory where the GSDML file is stored.



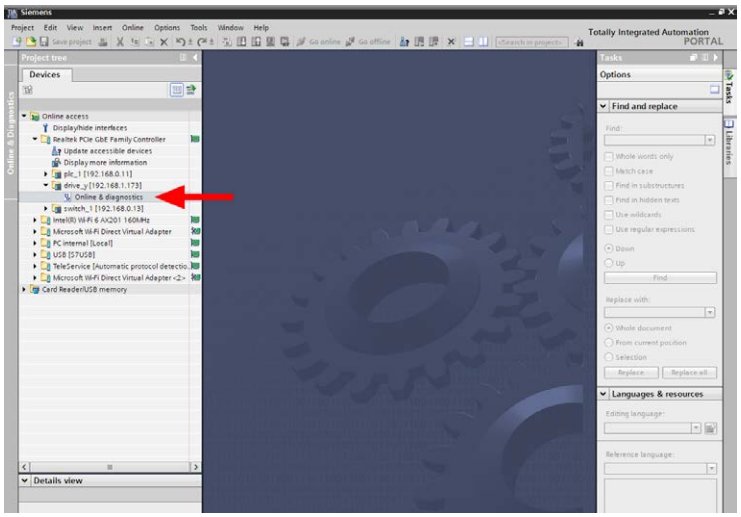
Flag the file and click on “Install”.



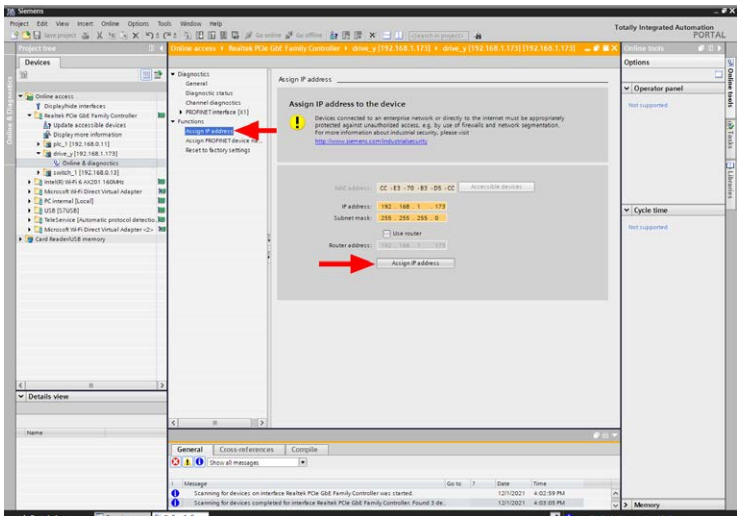
Wait until the message “Installation was completed successfully” appears.



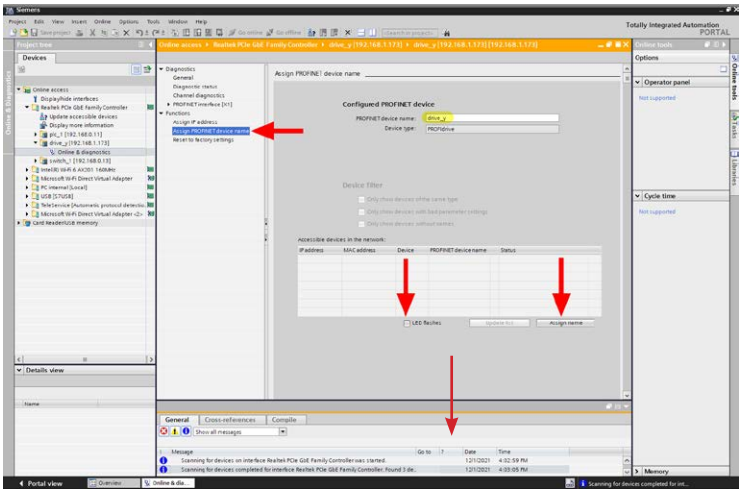
From the tree menu on the left, open the item “Online access” and select the network interface used by the PC, then click on “Update accessible devices”.



All the accessible devices on the interface are shown. Using the MAC address of the drive, select the device and click on “Online & diagnostics”



Select the item “Functions” - “Assign IP address”.
Enter the new data and click on “Assign IP address”.



Select the item “Assign PROFINET device name”.
Enter the new “PROFINET device name” and click on “Assign name”.
In order to verify and identify the device, it is possible to command the LED flashing by flagging “LED flashes”.

Device output behavior

Description of device output behavior

This chapter describes the device outputs behavior in certain cases.

Case	Output behavior
IOPS=bad (controller in STOP state)	Keep last value
Connection to controller is lost	Set output to zero
Power-on of the device	Set substitute value

ACYCLIC COMMUNICATION

INTRODUTION TO ACYCLIC COMMUNICATION PROFINET

This chapter describes the functions and the procedures to use AEC's drives in Profinet.
Please refer to Profinet Nutzerorganisation e.V. or visit the website www.profibus.com for further information on acyclic communication Profinet.

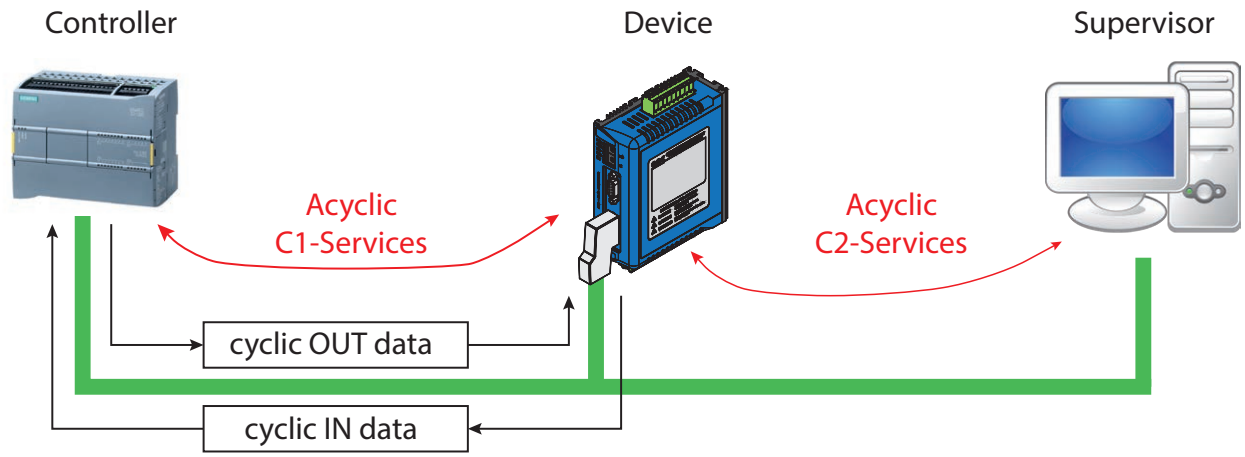
Profinet introduces a new service of acyclic read/write ; these communication services are embedded into special telegrams, that are inside the normal cyclic operativity of the bus.

The acyclic service permits to exchange volumes of data greater than the ones allowed by the cyclic service. At the same time, the communication will not be overloaded, because the acyclic communication telegram is added to the bus cycle only on request.

The Acyclic communication permits many features to the user:

- the master C1 can access in read/write to all the configuration and statuts parameters: registers, variables and tasks of the slave, and not only to the data contained in the cyclic process;
- the master C2 can access in read/write to all the configuration and statuts parameters: registers, variables and tasks of the slave;
- permits the access to the I&M (Information & Maintenance) of the drive

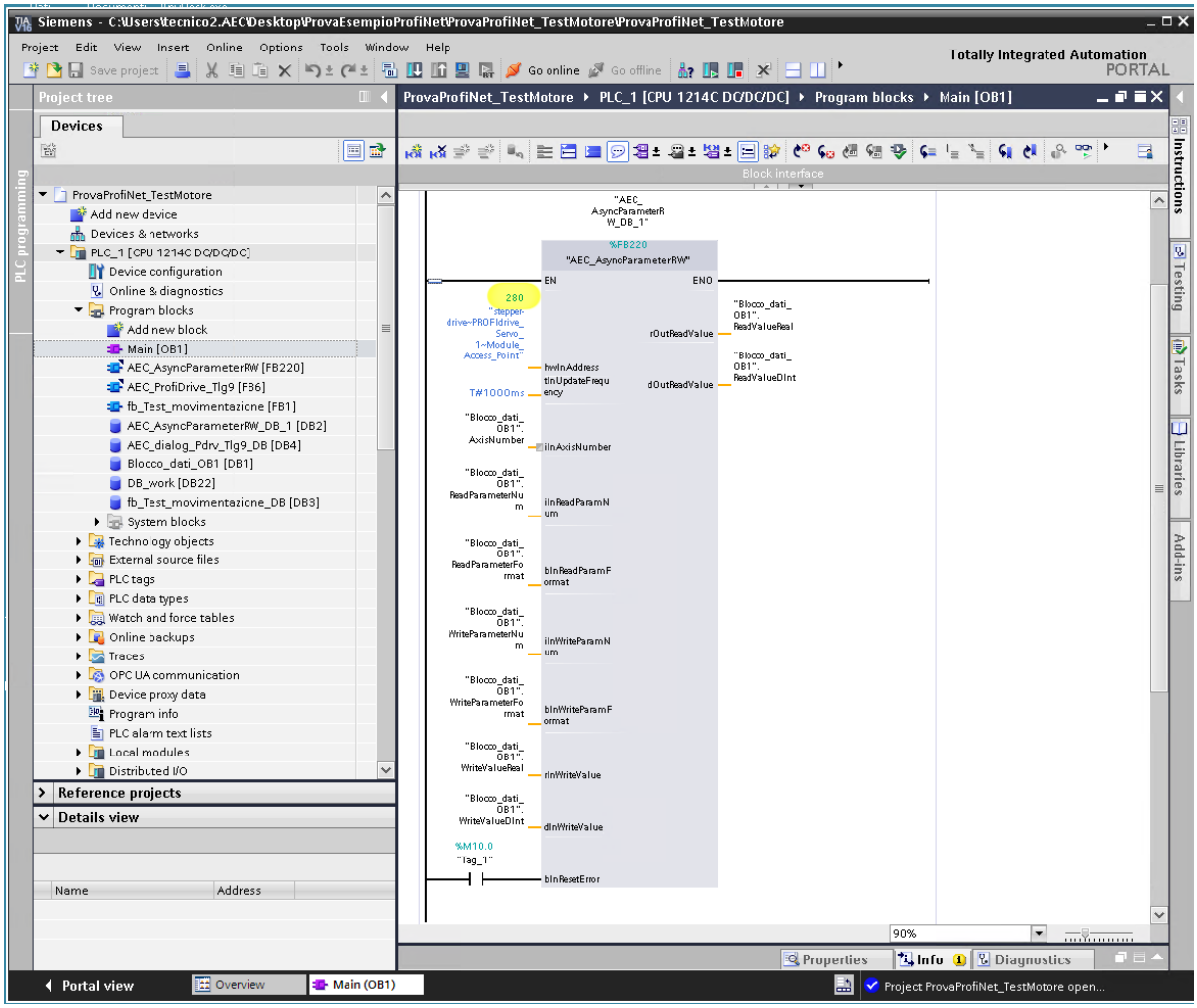
The following scheme summarizes the features of Profinet.



Nome	Descrizione
Controller	In a Profinet network, different classes of masters can coexist. The Controller manages the cyclic exchange with the slaves. Normally is the Controller (PLC), that manages the system automation. In case that the acyclic communication functionalities are enabled through the GSDML file, the acyclic connection between the Controller and Device is automatically enabled, in conjunction with the activation of the cyclic connection. In a Profinet network it is possible to use only one Controller.
Supervisor	The Supervisor are not able to execute cyclic exchange data with the Device. Normally, the Supervisor are visualization systems (eg. HMI) or analyses systems (network analyzers, notebook, PC), used only to monitor the state of the slaves or to alter some of their parameters.
Device	Stepper drive

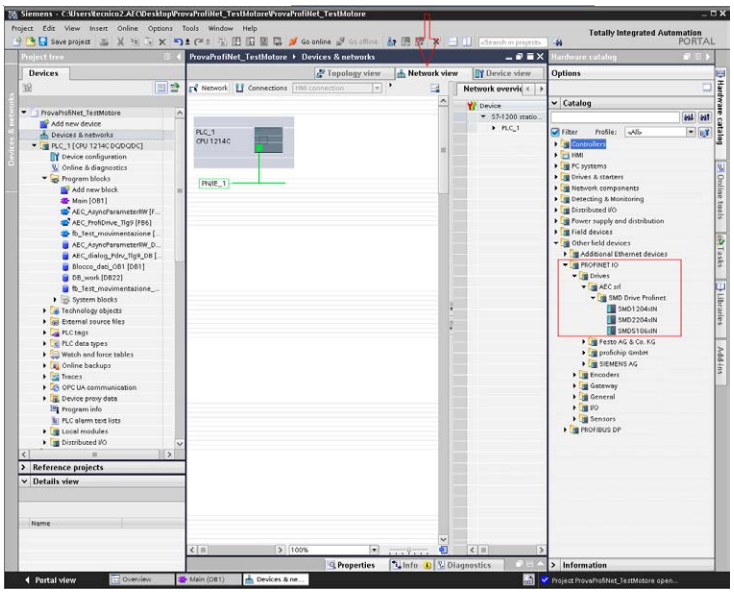
In order to access to the parameters not included in the standard telegams, it is possible to use the acyclic communication services.
This communication system permits to request, independently from the cyclic data exchange, any register or variable of the drive.
The read/write of the data is based on an addressing mechanism through slot/index and length.

Example of a segment for the writing and reading of acyclic data



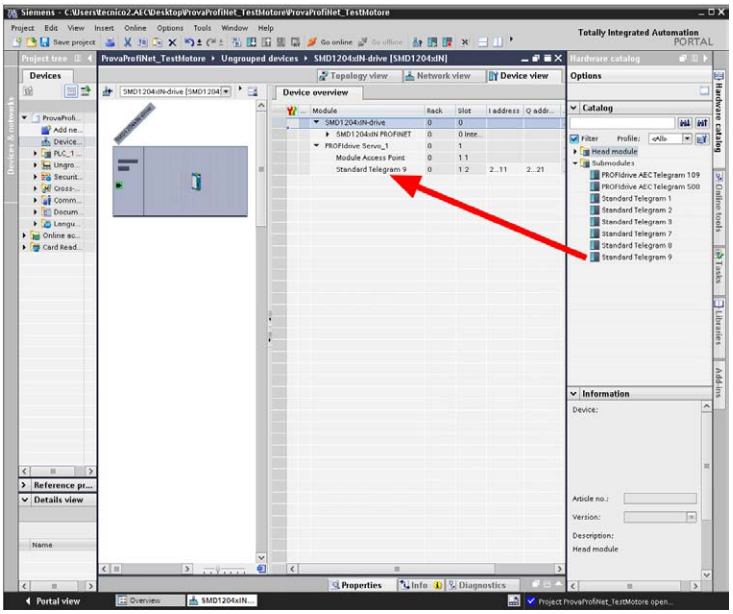
The highlighted value must be obtained from the hardware definition pages, as described below.

On PROFINET bus it is possible to communicate with the drives SMDyyyyxIN using the Modbus TCP protocol.
The IP address of the drive must coincide to the PROFINET IP address.
The communication port used is the 502.

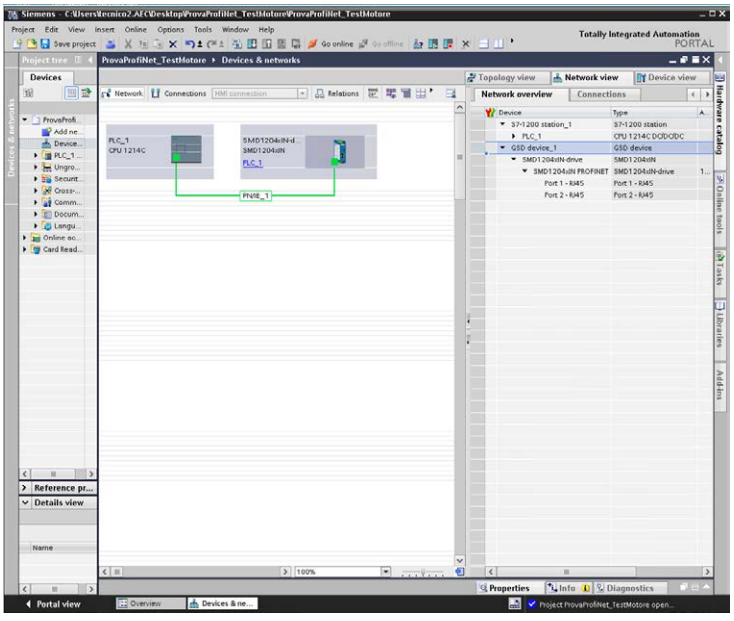


Select “Devices & networks” from the menu on the left, then select the tab “Network view”.
From the “Hardware catalog” on the right column, select the branch “Other field devices” - “Additional Ethernet devices” - “PROFINET IO” - “Drives” - “AEC srl” - “SMD Drive Profinet”.

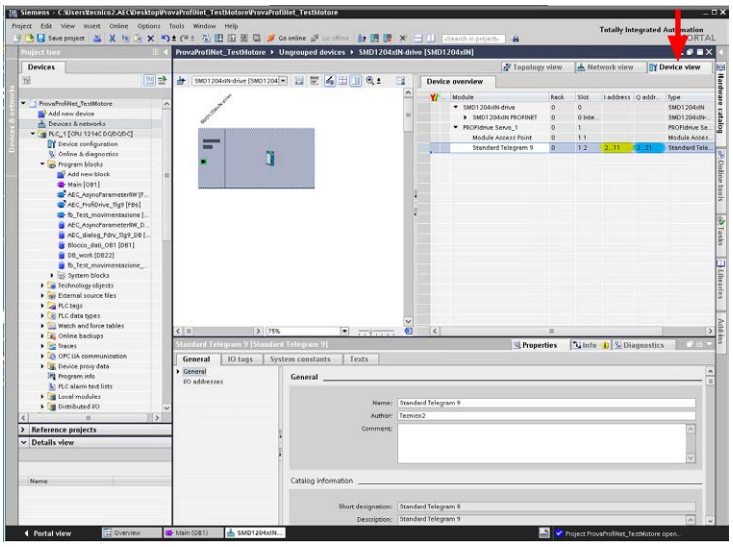
Drag the selected model in the “Network view” section.



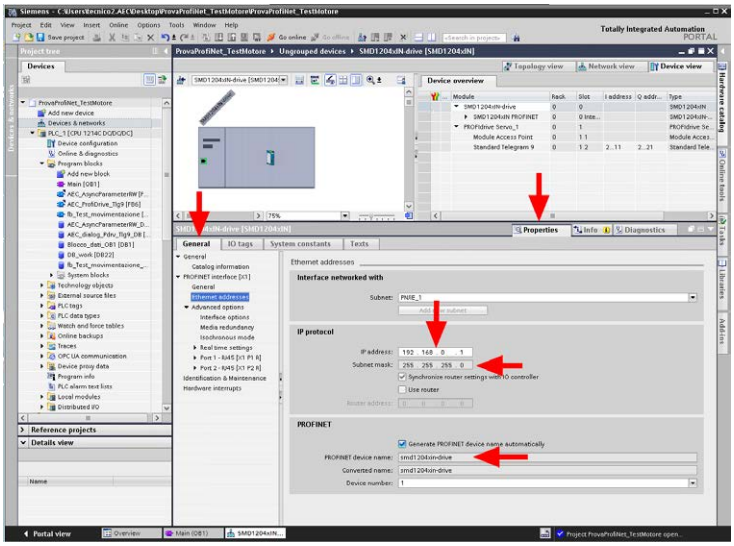
Click on the device and select the tab “Device view”.
Insert into “Device overview” column the selected telegram for the cyclic communication.



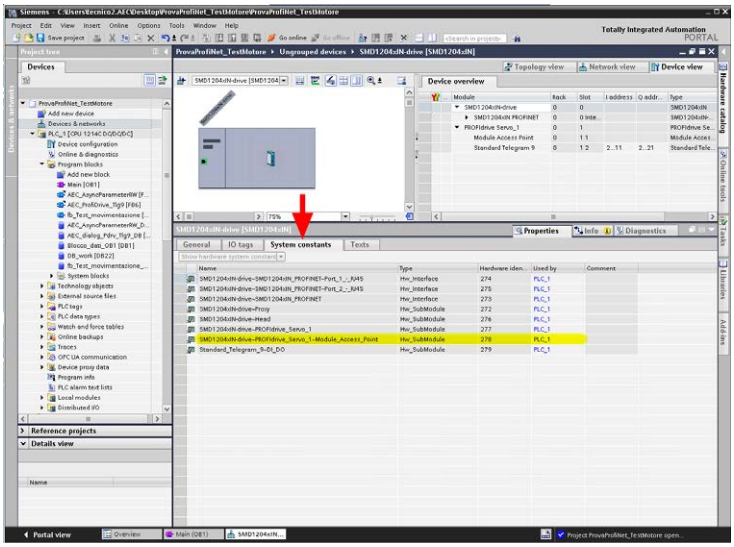
Associate the device to a PROFINET controller.



Return to the tab “Device view” and set the input/output addresses for the acyclic communication. It is also possible to use the default settings.
This values will be necessary for the configuration of the of the blocks of PLC program.



Open the tab “Properties” - “General” and set the IP address of the device, the net mask and the desired PROFINET device name.



Open the tab “System constants” and take note of hardware ID of the slot “Module access”, that will be necessary to configure the program block of the acyclic communication.

PROFIDRIVE SIGNALS

The following table contains the DO signals (Drive Objects signals) supported by the AEC drives. These data are transferred in cyclic way, using the service “Cyclic Data Exchange DP V0”. Data representation is Big-Endian type (MSWord - LSWord).

Nr. of signal	Name	Abbreviation	Format	Description
1	Controlword 1	STW1	Unsigned16	see Controlword 1
2	Statusword 1	ZSW1	Unsigned16	see Statusword 1
3	Controlword 2	STW2	Unsigned16	
4	Statusword 2	ZSW2	Unsigned16	
5	Speed setpoint A	NSOLL_A	Signed16	
6	Speed actual value A	NIST_A	Signed16	
27	Position setpoint value A	XSOLL_A	Signed32	
28	Position actual value A	XIST_A	Signed32	
32	Traversing block selection	SATZANW	Unsigned16	
33	Actual traversing block	AKTSATZ	Unsigned16	
34	MDI target position	MDI_TARPOS	Signed32	
35	MDI velocity	MDI_VELOCITY	Unsigned16	
36	MDI acceleration	MDI_ACC	Unsigned16	
37	MDI deceleration	MDI_DEC	Unsigned16	
38	MDI mode	MDI_MOD	Unsigned16	

CAUTION!!!

The direct access to the words in the periphery of a Profinet node, does not guarantee the data consistency, because the Profinet DATA-EXCHANGE cycle is not synchronous, or linked to the PLC cycle.

To read or write consistent data it is reccomended to use SFC14 and SFC15.

Structure of the Controlword 1.

Bit	Value	Significance	Notes
0	1	ON	Condition “Switched ON”; enable the drive.
	0	OFF (OFF 1)	Power-down (the drive comes back to the stauts “Ready to switch on”); stop the drive with the maximum possible ramp, if not already stopped. During the deceleration, the bit 1 of ZSW1 remains high. The command OFF is not interruptable.
1	1	No Coast Stop (no OFF 2)	Cancel all “Coast Stop” commands.
	0	Coast Stop (OFF 2)	Disable the drive (current off) and change the status to “Switching On Inhibited”; the motor turns free until stop.
2	1	No Quick Stop (no OFF 3)	Cancel all “Quick Stop” commands.
	0	Quick Stop (OFF 3)	If necessary, enable the bit “operation enable”; the drive is stopped with the maximum possible ramp and the status is changed to “Switching On Inhibited”. The command “Quick Stop “ is not interruptable.
3	1	Enable Operation	Enable the drive (Current ON).
	0	Disable operation	Disable the drive (Current OFF).
4	1	Do Not Reject Traversing Task	Active a task on the rising edge of the bit 6.
	0	Reject Traversing Task	Command an emergency ramp stop (Rdeceme parameter); the motor remains stopped in current. The active task is canceled.
5	1	No Intermediate Stop	Must always be high during the execution of a ramp.
	0	Intermediate Stop	The motor brakes with the set ramp and remains stopped in current. The task in execution remains active, and it is re-enabled when the bit 5 comes back to 1.
6	1	Activate Traversing Task (0 -> 1)	A rising edge enables the execution of a task or of a MDI setpoint.
7	1	Fault Acknowledge (0 -> 1)	The rising edge acknowledges and accepts the last fault contained in the buffer. The behaviour of the drive when a fault occurs depends on the type of alarm.
	0	No significance	
8	1	JOG CW ON (JOG 1 ON)	Prerequisites: the drive is in the status “Operation Enable”, and there is no ongoing positioning. Enable the clockwise movement.
	0	JOG CW OFF (JOG 1 OFF)	Stop the motor in ramp; the drive comes back to the status “Operation Enable” when the motor stops.
9	1	JOG CCW ON (JOG 2 ON)	Prerequisites: the drive is in the status “Operation Enable”, and there is no ongoing positioning. Enable the counterclockwise movement.
	0	JOG CCW OFF (JOG 2 OFF)	Stop the motor in ramp; the drive comes back to the status “Operation Enable” when the motor stops.
10	1	Reserved	Reserved.
11	1	Start Homing Procedure	Prerequisites: drive in “Operation Enable” The rising edge enables the homing procedure. The start of the home procedure resets the bit 11 of the Statusword.
	0	Stop Homing Procedure	The homing procedure is cancelled and the motor stops in ramp.
12 - 13		Device Specific	
14	1	Continuous Update	Force the continuous update of the movement parameters (speed, acc, dec) during JOG movements.
	0	No Continuous Update	Parameters are not updated during the movement.
15	1	Update Moving parameter	Force the setting of the parameters (speed, acc, dec) at each movement (jog, home).
	0	No Update Moving parameter	The movement parameters are updated at the activation of a traversing task.

Statusword 1 (ZSW1)

Structure of the Statusword 1.

Bit	Value	Significance	Notes
0	1	Ready To Switch On	Drive is enabled and initialized. Output current to the motor is disabled.
	0	Not Ready To Switch On	
1	1	Ready To Operate	Drive is power supplied.
	0	Not Ready To Operate	
2	1	Operation Enable	Drive is enabled to execute positionings.
	0	Operation Disable	Drive is disabled to execute positionings.
3	1	Fault Present	In the Fault buffer there are unread alarms and accepted by the master.
	0	No fault	
4	1	Coast Stop Not Activated (No OFF 2)	
	0	Coast Stop Activated (OFF 2)	Coast Stop command (OFF 2) is activated.
5	1	Quick Stop Not Activated (No OFF 3)	
	0	Quick Stop Activated (OFF 3)	Quick Stop command (OFF 3) is activated.
6	1	Switching On Inhibited	The drive can change status to “Switched ON” only with the condition “No Coast Stop AND No Quick Stop”, followed by the command “ON”.
	0	Switching On Not Inhibit	
7	1	Warning Present	There are warning messages in the parameter “I&M”.
	0	No Warning	
8	1	Reserved	
9		Reserved	
10	1	Target Position Reached	Axis to the target position.
	0	Not at Target Position	Axis not in position.
11	1	Home Position Set	Axis is homed.
	0	Home Position Not Yet Set	None home position is valid.
12	Edge	Traversing Task Aknowledgment (0 ->1)	On the rising edge, Traversing Task accepted.
13	1	Drive Stopped	Axis stopped.
	0	Drive Moving	Axis is moving.
14	1	Motor direction CCW	Forward direction
	0	Motor direction CW	Backward direction
15	1	Not In Position	Motor not in position.
	0	In Position	Motor in position.

STATUSWORD 2 (ZSW2)

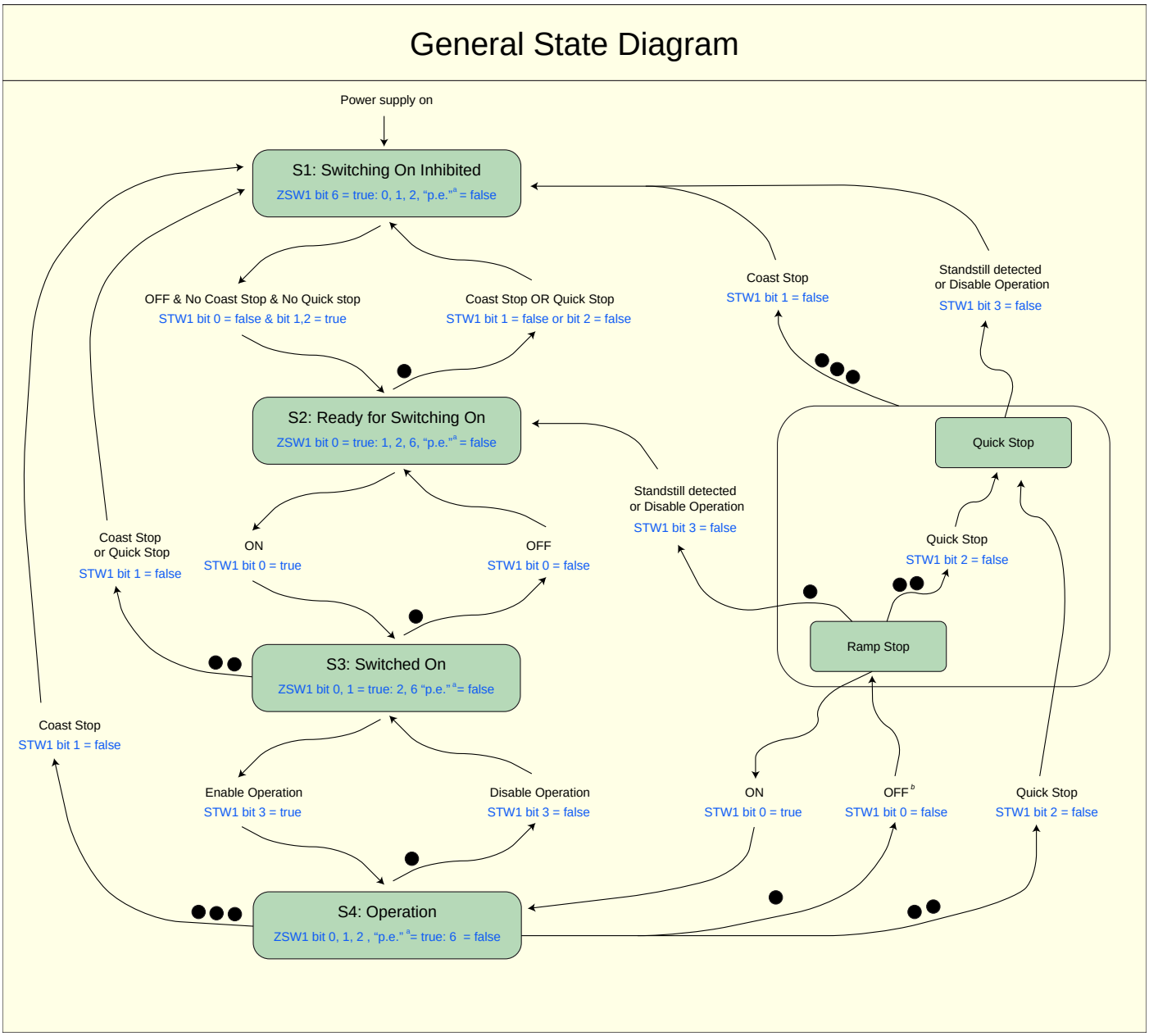
Structure of the Statusword 2, when telegrams 9 and 109 are used (Implemeted with firmware 606 or higher).

Bit	Value	Significance	Notes
0	1	Drive Moving	Axis in movement
	0	Drive Stopped	Axis stopped
1	1	On acceleration ramp	Motor in acceleration ramp
	0		
2	1	Motor at costant velocity	Motor at constant speed
	0		
3	1	On deceleration ramp	Motor in deceleration ramp
	0		
4		Reserved	
5		Reserved	
6		Reserved	
7		Reserved	
8		Reserved	
9		Reserved	
10		Reserved	
11	1	Positioning enable	Positioner enabled
	0		
12		Reserved	
13		Reserved	
14		Reserved	
15		Reserved	

POSITIONING MODE

PROFIBUS GENERAL STATE MACHINE

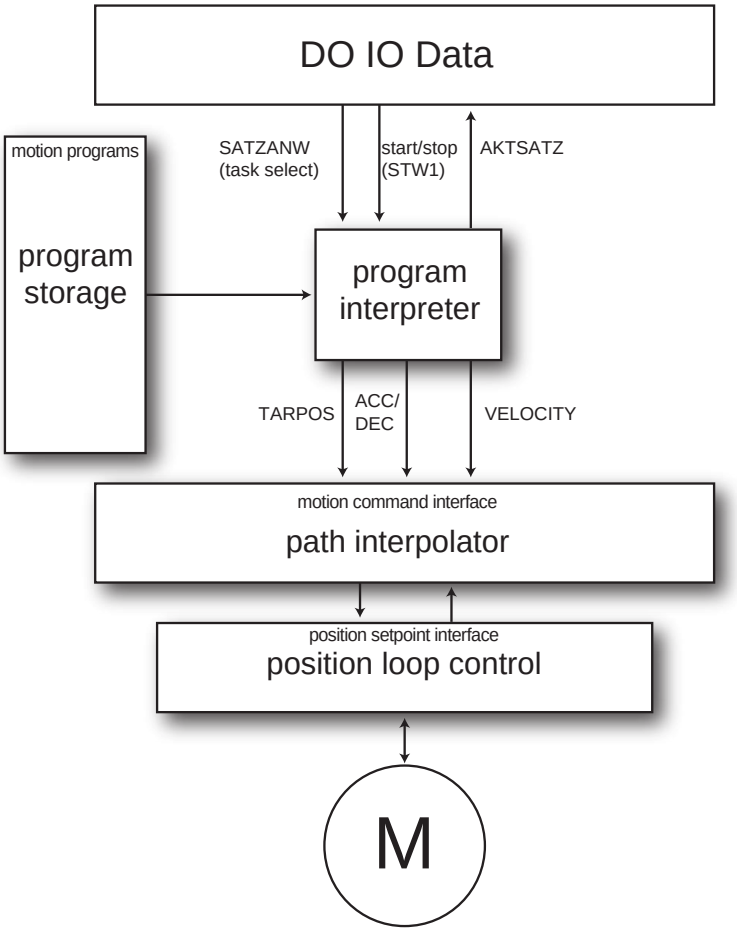
The Profinet general state machine, for the management of the Drive Objects, is composed by 4 states, common to all operation modes.
The green blocks indicate the states, and the arrows indicate the transitions from a state to another. Since, in some states, are allowed multiple transitions with different priorities, the points indicate the priority of the transition. Greater is the number of points, higher is the priority (the transitions without points have the lowest priority).
The transitions can be activated only from the master, by acting on the bit of the Controlword (STW1).
The Statusword indicates the current state of the drive.
The internal conditions of the connected drive may generate some transitions indipendent from the master.



The motion controller of this mode is composed of four blocks:

1. position loop controller
2. path interpolator
3. program interpreter
4. program storage

The interpolator generates, cyclically, the setpoint position for the position controller of the axis.
The input data to the interpolator is a movement command, composed of a target position; a speed setpoint; the acceleration and deceleration ramps; that are used to calculate the new positioning profile
Using the function “Program Submode”, the movement commands are stored into a memory (program storage) and manipulated from the interpreter.
Through the Controlword1 and the Traversing block selection, it is possible to check the program interpreter, and to indicate which command has to be loaded and run.



The execution of a movement program, or the switch to a new program, during the execution of a positioning block, can be obtained by selecting the new block through SATZANW (while the bit 15 is 0) and by enabling the bit 6 of STW1

Description of signal SATZANW (Traversing block selection).

Bit	Descrizione
0 - 9	Number of the block, of the program storage, to start. (values range 0 .. 63) Bits from 0 to 9 are significant only in the Program submode status.
10 - 14	Reserved
15	Selection of Submode (Modeswitch): = 1 Activation of the Submode MDI. If a positioning is ongoing (Extended state machine not in Basic State), the MDI Submode will be activated at the end of it (STW1 bit4). = 0 Disactivation of the Submode MDI. If there is a MDI command ongoing, a stop will be commanded, and the command will be rejected.

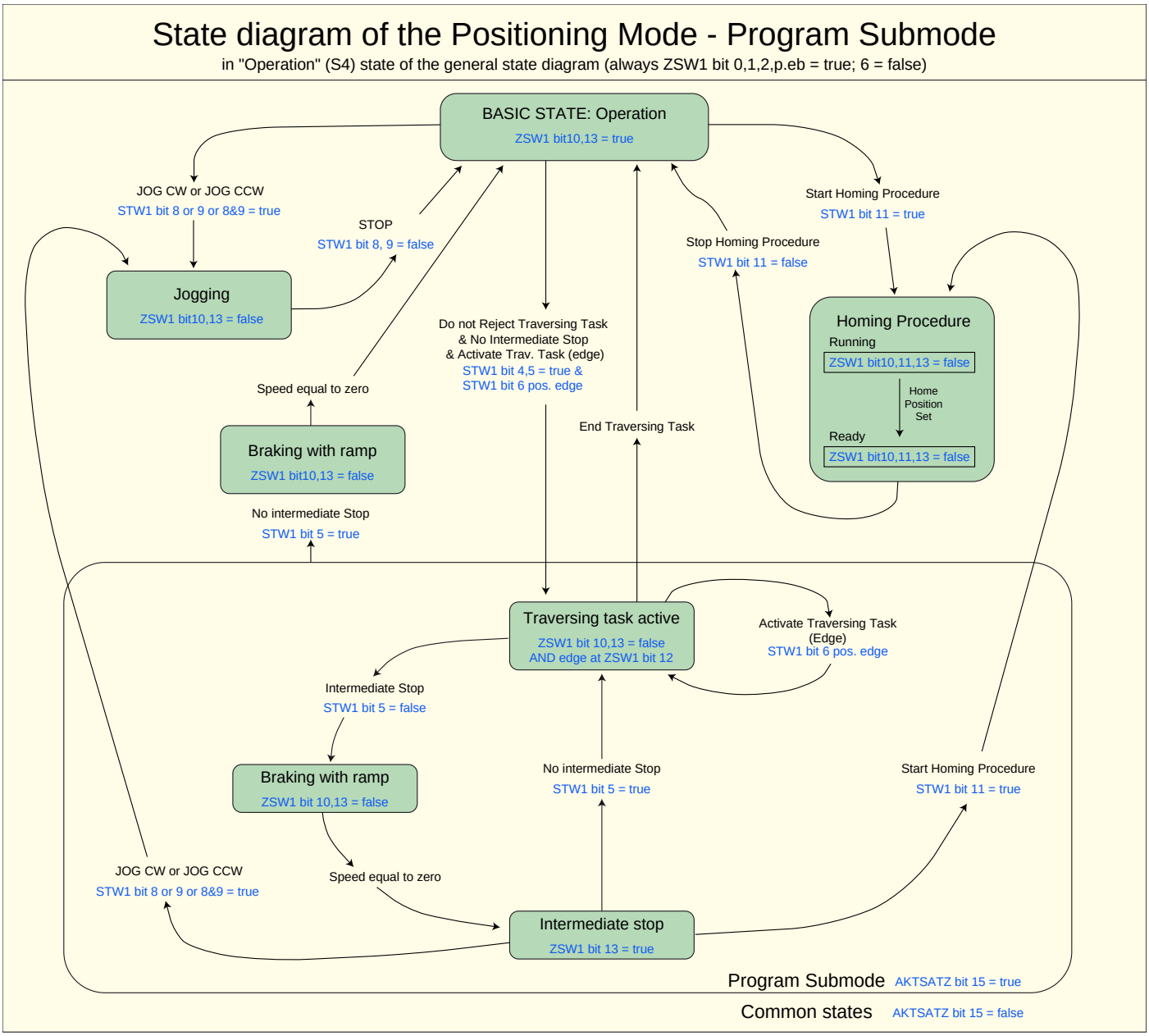
Description of signal AKTSATZ (Actual traversing block).

Bit	Descrizione
0 - 9	Number of the currently active block. (values range 0 .. 63) Bits from 0 to 9 are significant only in the Program submode status. (In MDI-submode and when there are no Traversing blocks running, the value will be 0).
10 - 14	Reserved
15	Status of Modeswitch: = 1 MDI submode active. Movement parameters are difined by the signals MDI_TARPOS, MDI_VELOCITY, MDI_ACC and MDI_DEC. = 0 Program submode active. Movement parameters are difined by the motion task indicated by bits 0 - 9.

Description of signal MDI_MOD (MDI mode).

Bit	Descrizione
0	Absolute/relative positioning mode = 1 Absolute positioning. The target quota indicated in TARPOS defines the absolute quota for the movement (referring to the zero). = 0 Relative positioning. The target quota indicated in TARPOS defines the relative quota for the movement (referring to the current quota).
1	Reserved
2	Reserved
3 - 15	Reserved

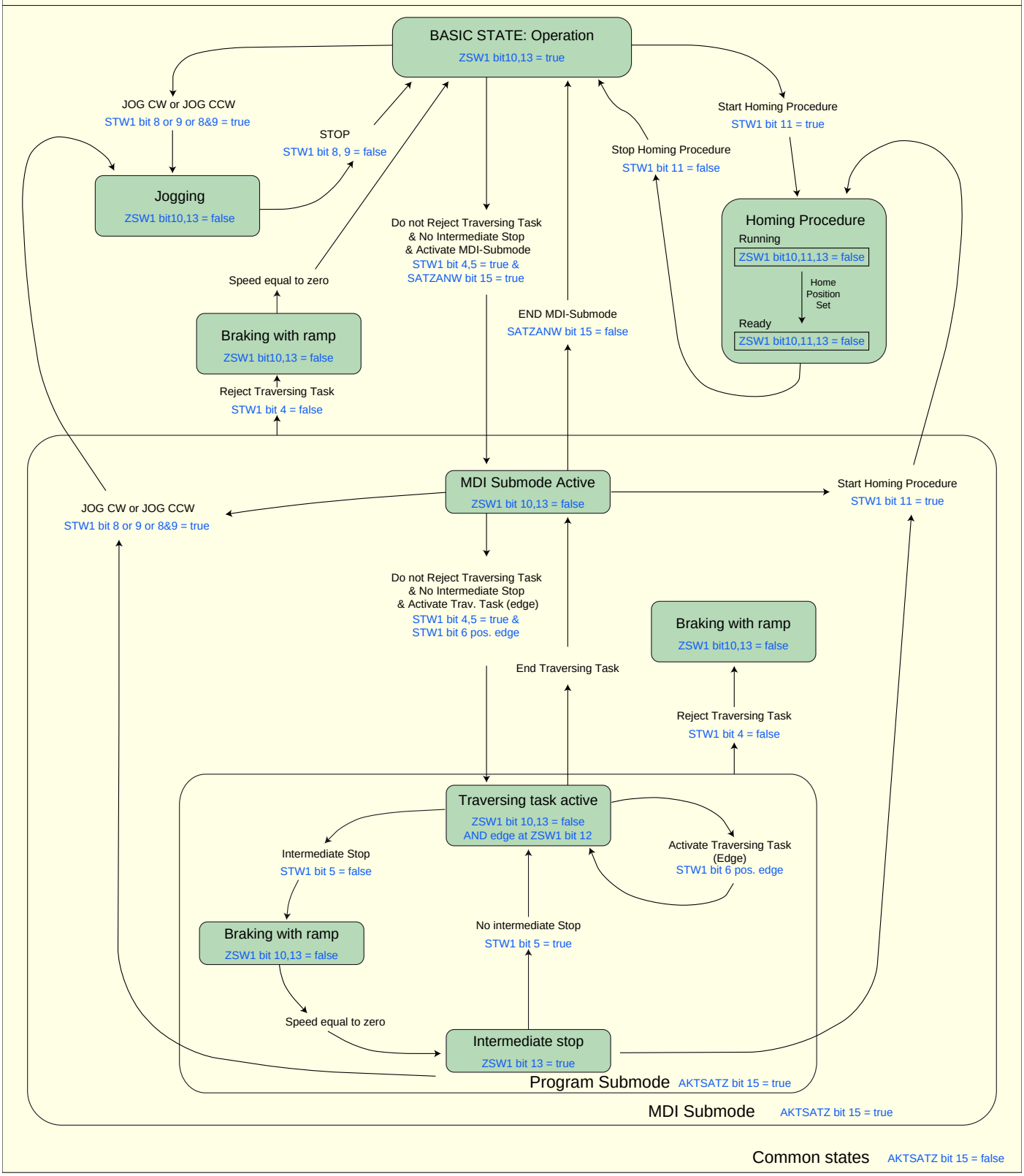
The extended state machine Positioning mode is composed of 3 modules: the Common State, Program Submode and the MDI Submode.
The first two states are represented below.
The Activate Traversing Task transition runs the motion task indicated by the bits from 0 to 9 of the signal SATZANW.



The complete state machine is represented below.

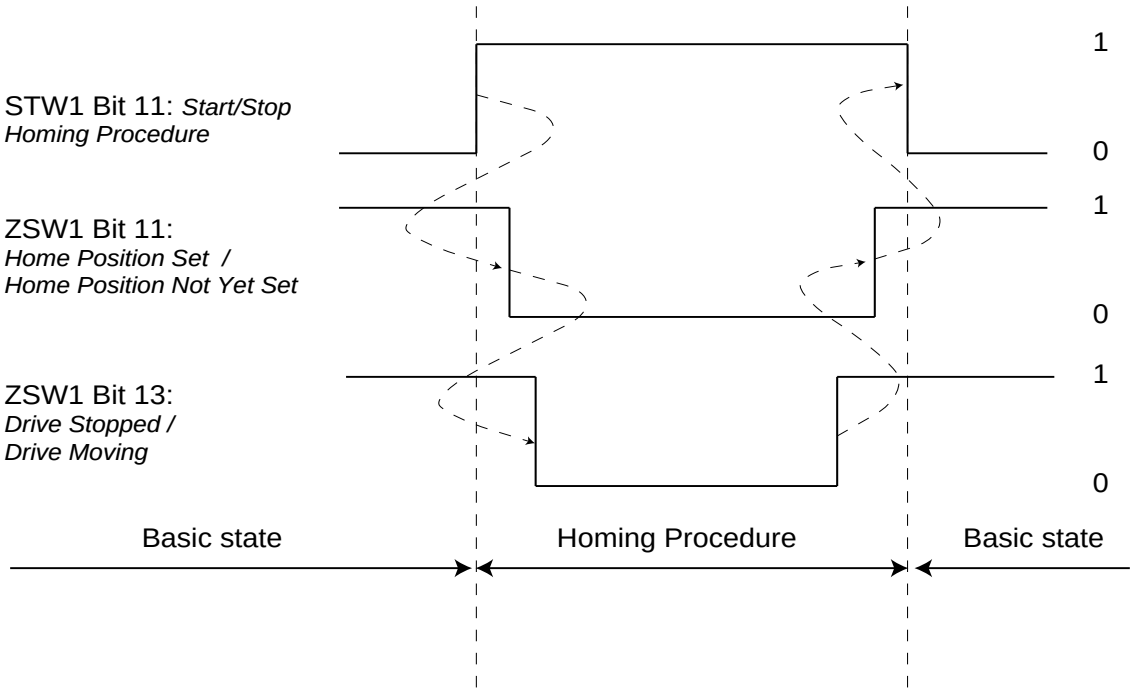
State diagram of the Positioning Mode - MDI Submode

in "Operation" (S4) state of the general state diagram (always ZSW1 bit 0,1,2,p.eb = true; 6 = false)



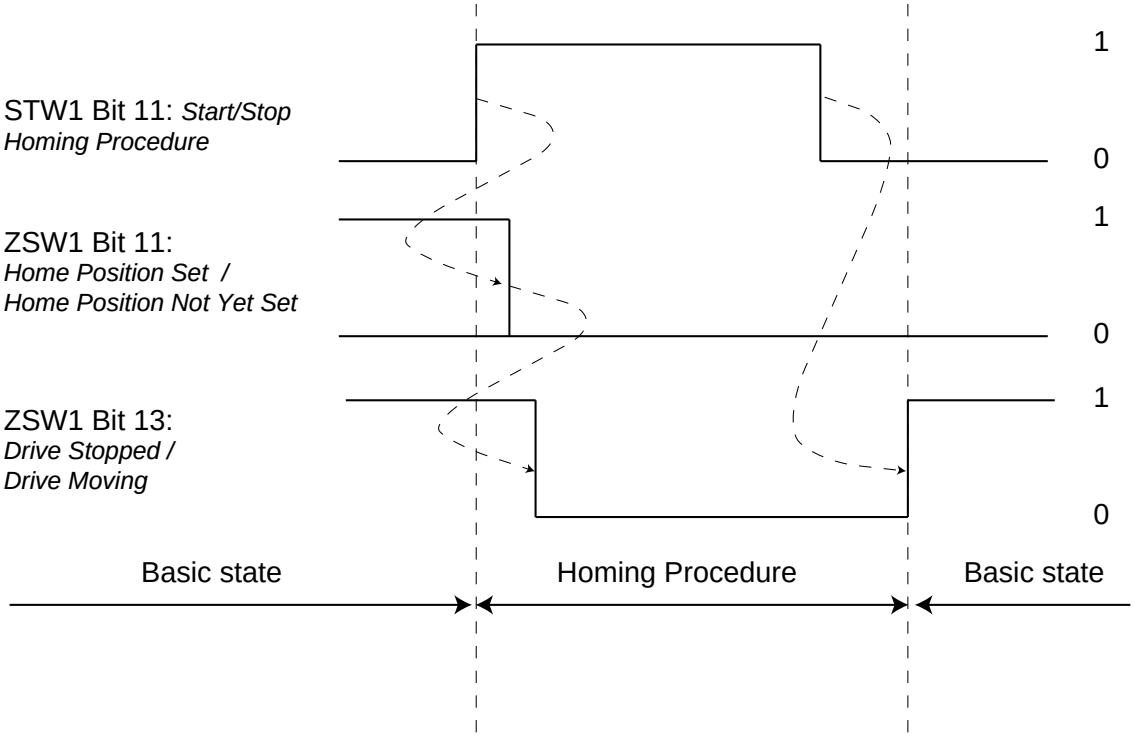
The following diagram shows the bit sequence to activate the homing procedure.

Homing procedure: Home Position Set



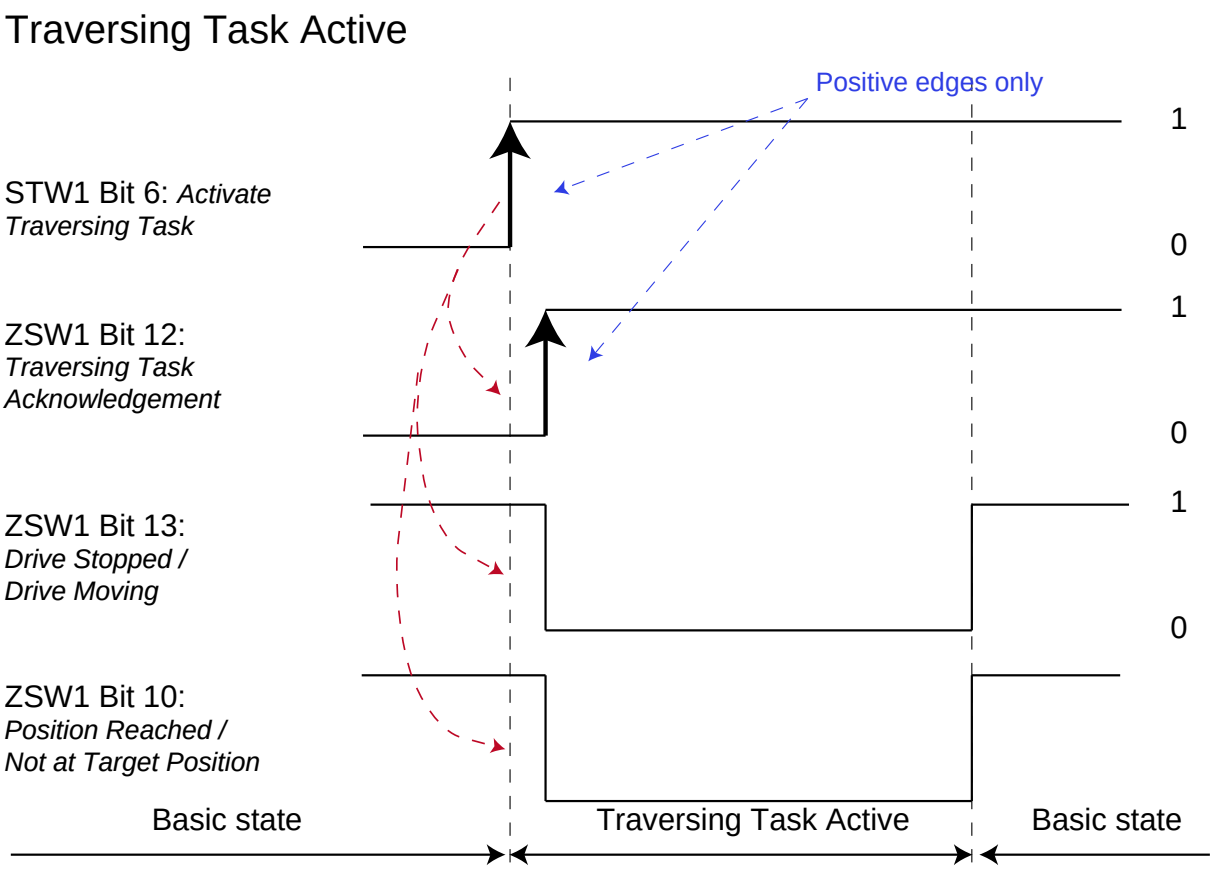
Activation of the homing sequence

Homing procedure: Stop Homing Procedure by the Controller

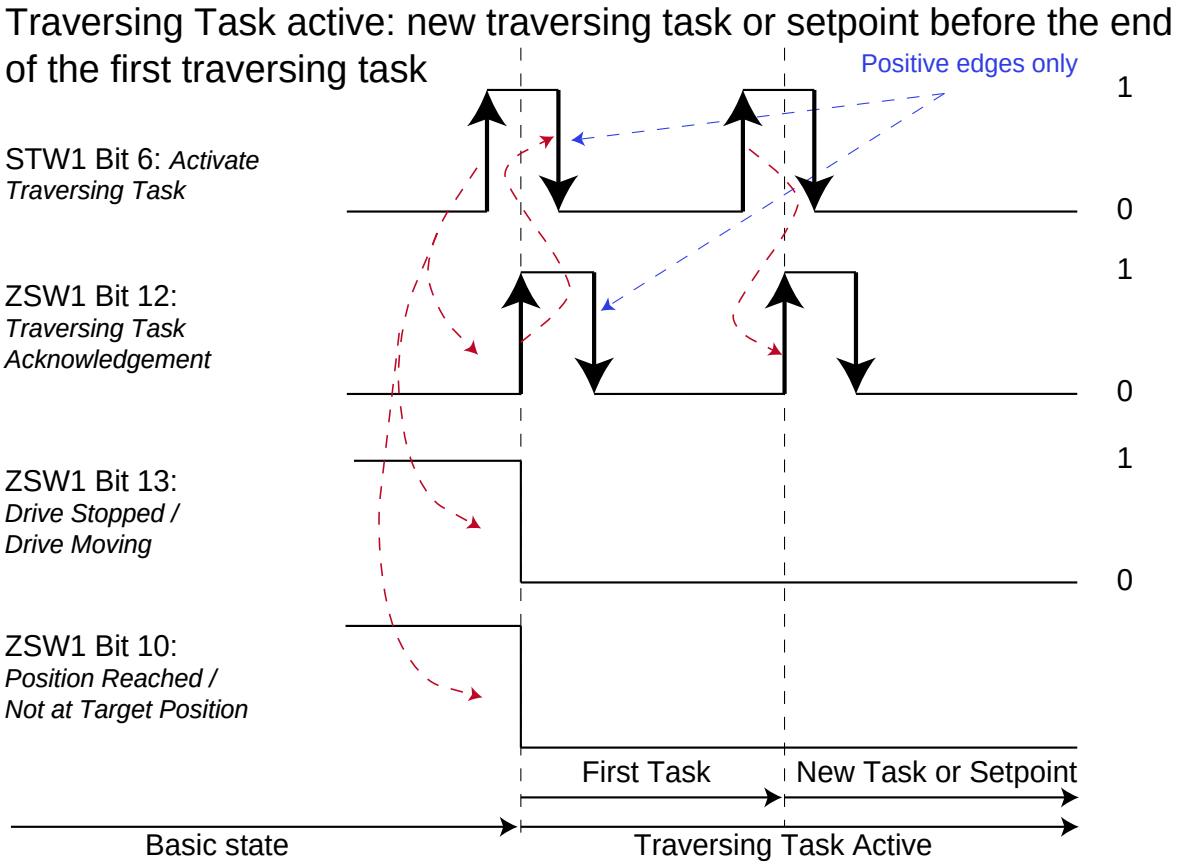


Abort of the homing sequence

The following diagram shows the bit sequence to activate a traversing task.



The following diagram shows the bit sequence to activate a new traversing task or a new setpoint before the end of the previous task.



Standard Telegram 9

Positioning interface (Program submode plus MDI submode)

IO Data Number	Setpoint	Actual value
1	STW1	ZSW1
2	SATZANW	AKTSATZ
3	STW2	ZSW2
4	MDI_TARPOS	XIST_A
5		
6	MDI_VELOCITY	
7		
8	MDI_ACC	
9	MDI_DEC	
10	MDI_MOD	

Standard Telegram 109 + External_encoder + motor _encoder + IO

Positioning interface (Program submode plus MDI submode)

IO Data Number	Setpoint	Actual value
1	STW1	ZSW1
2	SATZANW	AKTSATZ
3	STW2	ZSW2
4	MDI_TARPOS	XIST_A
5		
6	MDI_VELOCITY	EXT_ENC
7		
8	MDI_ACC	MOT_ENC
9	MDI_DEC	
10	MDI_MOD	SERVICE INPUTS
11	DIGITAL OUTPUTS	DIGITAL INPUTS
12	ANALOG OUTPUT	ANALOG INPUTS
13		
14		

PROPRIETARY REGISTERS

REGISTERS

The registers are memory locations inside the drives, each one with specific functions. When a determined value is written in these locations, a function defined by the correspondent register is carried out.
Some of the registers are at 32 bit, and the access to these registers is done by reading or writing two consecutive registers, according to Modbus protocol.

N.B.: ALL THE INTERNAL RESOURCES, REGISTERS E VARIABLES OF THE AEC DRIVES ARE MAPPED IN THE MEMORY LOCATION 4 "HOLDING REGISTER".

N.B.: Verify if the Modbus addresses of the master start from 0 or 1. AEC Modbus addresses start from 0, in case of using a master in which they start from 1 (e.g. Siemens or Weintek) it is necessary to add 1 to the AEC Modbus address.
Example: Rposact= 0000+1= 0001-2

POSITION REGISTERS

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0000-1	0x2000	**Rposact**	Actual position of the axis	Step	0x80000000	0x7FFFFFFF	Signed32	LS – MSWORD MAP READ
0004-5	0x2004	**Rposactreq**	Theoric position os the axis	Step	0x80000000	0x7FFFFFFF	Signed32	LS -MSWORD MAP-READ
0008-9	0x2008	**Rpostarg**	Target position for GO/GOR functions	Step	0x80000000	0x7FFFFFFF	Signed32	LS – MSWORD ANA_T MAP WRITE
0012-13	0x200C	**Rupplim**	Upper limit quota	Step	0x80000000	0x7FFFFFFF	Signed32	LS – MSWORD SAVE MAP WRITE
0016-17	0x2010	**Rlowlim**	Lower limit quota	Step	0x80000000	0x7FFFFFFF	Signed32	LS – MSWORD SAVE MAP WRITE
0055-56	0x2037	**Rposactsa-ved**	Saved quota at switch-off of the drive	Step			Signed32	LS – MSWORD SAVE NO MAP RO
0057	0x2039	**Rposactsa-vedflag**	B0: 1= Quota saved correctly				Unsig-ned16	WORD RO

Automatic storing of the actual quota at the switch off of the drive

The drive is equipped with a circuit that automatically detect when the supply voltage of the logic stage drops below about 20Vdc.

When this event occurrs, the firmware interpretes it like a voltage loss, immediately cuts the current to the motor and, thanks to the residual load of the condensers, tries to store the actual quota (register Rposact) in the non-volatile memory.

When the voltage is again supplied to the logic stage, the firmware restarts and a dedicated function checks if the quota saved in the non-volatile memory is valid by executing some controls on data congruency.
If the stored quota is valid, this value is loaded in the register Rposactsaved (register 55) and the bit 0 of the register Rposactsavedflag (register 57) is set to 1.

If the stored quota is not valid, both Rposactsaved and Rposactsavedflag registers are load with the value 0.

The user has the possibility to check if there is an available valid switch off quota, and so pass the value of Rposactsaved in Rposact.

Please note that the motor must be in standstill at the moment of the voltage loss, else the saved quota will be not valid, because it may be affected by possible movements due to the inertia.

In case of voltage oscillations, the save of the quota is made only during the first signal of voltage loss. The saving is enabled again after the drive is powered again. In case the drive hasn't been completely switched off, if the quota Rposact is varied with respect to the saved quota, it is assumed that the control program had been able to put the motor in current and to move it in a controlled way.

It is essential to take account that this function uses the residual load of the condensers to execute the operations necessary to write the data into the non-volatile memory.
It may occur that the residual load of the condensers is not sufficient to complete the saving of the data.
In this case, when the drive is powered again, the data saved will be invalid. It will be necessary to decide the operations to be carried out, like an example an homing of the axis.

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0199	0x20C7	**Rstsflg**	Flag status register Bit 0: Drive enabled Bit 1: Drive in alarm Bit 2: Axis homed Bit 3: Motor in movement (theoric) Bit 4: Motor in acceleration Bit 5: Motor at constant speed Bit 6: Motor in deceleration Bit 7: Information contained in register Rstscllp Bit 8: Home executed with errors Bit 9: Status of current (0=CurOff/1=CurON) Bit 10: 1=Motor in position Bit 11: Following error Bit 12: Motor moved while in “disable” state (only with encoder). The range is given by the value of the Rdeadpos register. Bit 13: Counterclockwise rotation direction Bit 14: Actual quota out of software limits range Bit 15: Home in progress				Unsig-ned16	WORD MAP-READ RO
0200	0x20C8	**Rstsflg1**	Flag status 1 register Bit 0: Alarms are present in the buffer Bit 1: Warning is present Bit 2: Power OFF signal Bit 3: STOP in progress Bit 4: Task in progress Bit 5: Lower SW intervention limit Bit 6: Upper SW intervention limit Bit 7: BLS intervention (memory) Bit 8: FLS intervention (memory) Bit 9: Operation in voltage limit Bit 10: Saturated regulators Bit 11: Current limit is active Bit 12: Encoder phasing in progress Bit 13: 1 = Register 51 (Rextencvel) updated Bit 14: 1 = Register 89 (Rextenctopvel) updated Bit 15: SSP active				Unsig-ned16	WORD MAP READ RO
0203	0x20CB	**Rstscllp**	Closed-loop flag status register Bit 0: Phased encoder Bit 1: Motor in theoric movement Bit 2: Motor in position Bit 3: Positioning alarm Bit 4: Following alarm Bit 5: Reserved Bit 6: Positioning time out Bit 7: Motor in real movement Bit 8: Current limit Bit 9: Speed limit Bit 10: Acceleration limit B11: B12: B13: B14: B15: N.B. bits 0-1-2 are used to set the bit 7 of Rstsflg				Unsig-ned16	WORD RO

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0058	0x203A	**Rflag1**	Operation mode flag 1 register (not savable) B0 = Wait end of movement B1 = Update JOG immediately B2= Enable the CAM table for function of positioning from table with strart from master quota + digital outputs. B3: B4: B5: B6: B7: B8: B9: B10: B11: B12: On the rising edge, resets quotas in closed loop (real quota and theoric quota) (equal to bit 12 of STW1 of Profibus) B13: B14: B15:		0	65535	Unsig-ned16	WORD

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0102-03	0x2066	**Rflag**	Operation mode flag register B0 = Wait end of movement B1 = Update velocity JOG immediately (see also BIT1 of Rflag1) B2 = Enable SW low limit B3 = Enable SW high limit B4 = Enable Back Limit Switch B5 = Enable Forward Limit Switch B6 = Reset Posact after Home B7 = Reset Motenc after Home B8 = Reset Motext after Home B9 = Reset Posact after Home offset B10 = Reset Motenc after Home offset B11 = Reset Motext after Home offset (0 = Reset quota / 1 = No reset quota) B12 = 1: Enable automatic offset correction in closed loop B13 = 1: Enable automatic PID current correction in closed loop B14 = 1: Unidirectional gear B15 = 1: Forward only gear B16 = 1: Back only gear B17 = 1: In Smart Mode, at “Current ON”, update the requested actual position with the actual quota obtained by the encoder B18 = 1: In Smart Mode, at “Current OFF”, doesn’t update the requested actual position with the actual quota obtained by the encoder, in order to permit the recovery of the quota at “Current ON” B19 = 0: USB port without slave address and fixed parameters at 9600,N,8,1 /1: USB port with parameters set by registers but without slave address. B20 = B21 = B22 = B23 = B24 = B25 = B26 = B27 = B28 = B29 = B30 = B31 =		0	0x7FFFFFFF	Unsig-ned32	LS – MSWORD SAVE NO MAP
0278	0x2116	**Rhwconfig**	Hardware configuration B0 (1): 0= Not used B1 (2): Motor encoder direction B2 (4): Motor rotation direction B3 (8): Phased encoder in FC reset flag B4 (16): Configure An. Inp 0 SMD1104 B5 (32): 0=Motor encoder in quadrature / 1=Motor encoder pulse/direction B6 (64): 1=Preset encoder SSI B7 (128): 0=Normal SSI encoder / 1=Complement SSI encoder B8 (256): 0=Gray SSI encoder / 1=Binary SSI encoder B9-10: 00=SSI encoder with right alignment data / 01=SSI encoder with left alignment data / 10=SSI encoder with centered data B11 (2048): 0=DMD with PWM 50% / 1=DMD with PWM 0-100% B12 (4096): 1=Realign entering quota in interpolation mode (Master CanOpen Nardi) B13 (8096): Rialign quotas Rposact Rposactreq in interpolation mode (Master CanOpen Sipro)		0	65535	Unsig-ned16	WORD SAVE

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0063-64	0x203F	**Rvel**	Maximum translation speed	rev*100 /s	-10000	10000	Signed32	LS – MSWORD ANA MAP WRITE
0065-66	0x2041	**Rvss**	Start and stop speed (Start/Stop)	rev*100 /s	0	10000	Signed32	LS – MSWORD MAP WRITE
0067-68	0x2043	**Racc**	Acceleration ramp. If =0, ramp is disabled.	rev*10/ s²	0	200000	Unsig-ned32	LS – MSWORD ANA MAP WRITE
0069	0x2045	**Raccpro**	Acceleration profile 0=S 10=Linear		0	10	Unsig-ned16	WORD SAVE NO MAP
0070-71	0x2046	**Rdec**	Deceleration ramp. If =0, ramp is disabled.	rev*10/ s²	0	200000	Unsig-ned32	LS – MSWORD ANA MAP WRITE
0072	0x2048	**Rdecpro**	Deceleration profile 0=S 10=Linear		0	10	Unsig-ned16	WORD SAVE NO MAP
0073-74	0x2049	**Rdeceme**	Emergency deceleration ramp	rev*10/ s²	0	200000	Unsig-ned32	LS – MSWORD SAVE MAP WRITE
0075-76	0x204B	**Rvelact**	Actual speed of the motor	rev*100 /s			Signed32	LS – MSWORD MAP READ RO
0077-78	0x204D	**Rvelactreq**	Actual speed requested by the motor	rev*100 /s			Signed32	LS – MSWORD MAP READ RO
0079-80	0x204F	**Rvelmax**	Maximum speed during quota recovery	rev*100 /s	0	10000	Signed32	LS – MSWORD SAVE NO MAP
0100	0x2064	**Rdefum**	Define the velocity and acceleration divider. The following values can be taken: 1= 1:1 Ratio between the set value and the real value 10= 10:1 Ratio between the set value and the real value 100= 100:1 Ratio between the set value and the real value		1	100	Unsig-ned16	WORD NO MAP SAVE
0327	0x2147	**Rpwmacc**	Only for DMD. Acc/dec PWM ramp	bit/s²			Unsig-ned16	WORD
0328	0x2148	**Rpwm**	Only for DMD. PWM opening in PWM mode, expres-sed in bit. It can range from 0 (100%) to 1250 (100%)	%	0	1250	Unsig-ned16	WORD
0366	0x216E	**Rveladjpicur**	Maximum velocity for current PI reduction at low velocity		0	32767	Unsig-ned16	WORD SAVE
0406	0x2196	**Rdlyadjpicur**	PI current correction activation delay (0=disable correction)	ms	0	32767	Unsig-ned16	WORD SAVE
0410	0x219A	**Rzerovellim**	Minimum value of the motor encoder steps for axis in movement. Used to detect the motor in stop in homing in mecha-nical stop (hard-stop) in SmartMode	Enc. Pulses	0	65535	Unsig-ned16	WORD SAVE NO MAP
0411	0x219B	**Rzeroveltim**	Motor encoder sample time for stopped motor reading. Used to detect the motor in stop in homing in mecha-nical stop in SmartMode	ms	0	65535	Unsig-ned16	WORD SAVE NO MAP
0412	0x219C	**Rcurlimtim**	Filter time for the signal of current limit in FOC_CLO-SE	ms	0	65535	Unsig-ned16	WORD SAVE NO_MAP

0493-494	0x21ED	**Rveltarg**	Velocity target in function mode 9 (CSV). Only for protocols CANopen and EtherCAT.		-20000	+20000	Signed32	D WORD MAP WRITE

MOVEMENT COMMANDS

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0059	0x203B	**Rcmdwr**	PC writing/indexer reading command register B0 (1) : DISABLE DRIVE B1 (2) : ENABLE DRIVE B2 (4) : ABORT B3 (8) : STOP B4 (16) : ESTOP B5 (32) : JOG CW B6 (64) : JOG CCW B7 (128) : New setpoint GO B8 (256) : New setpoint GOR B9 (512) : HOME B10 (1024) : GEAR B11 (2048) : CAM Start Stop B12 (4096) : CAM B13 (8192) : TASK B14 (16384): BESTOP B15 (32768): SHIFT STOP		0	65535	Unsig-ned16	WORD MAP WRITE WO
0060	0x203C	**Rcmd1wr**	PC writing/indexer writing command 1 register B0: Current pulse on phase A B1: Motor phases wiring test B2: Winding mode B3: Smit mode B4: Realign quotas in SmartMode (Remove possible thrust of the motor) B5: Not used B6: Set step/dir cam mode (function as step/dire mode, but without the need to pass from “Position” to “Step Dir”) B7: Recover the motor quota in SmartMode (if the motor has been shifted with the drive disabled) B8: B9: B10: B11: B12: B13: B14: B15:		0	65535	Unsig-ned16	WORD MAP WRITE WO

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0061	0x203D	**Rcmdrd**	PC reading/indexer writing command register B0: B1: B2: Positioning ABORT B3: Ramp movement STOP B4: ESTOP active B5: JOG + B6: JOG - B7: New setpoint GO B8: New setpoint GOR B9: Home B10: GEAR B11: CAM Start Stop Mode B12: CAM mode B13: TASK mode B14: BESTOP active B15:				Unsig-ned16	WORD MAP READ RO
0062	0x203E	**Rcmd1rd**	PC reading/indexer writing command 1 register B0: 1=Current step executed B1: 1=Motor phases wiring test executed (reset by writing 0 on Rcmd1wr or at the start of another test) B3: 1=Winding mode active B4: 1= Smit mode active B5: 1= Smart mode quotas realignment executed B6: Not used B7: 1= Step/dir cam mode active B8: 1= SmartMode motor quota recover executed B9: B10: B11: B12: B13: B14: B15:				Unsig-ned16	WORD MAP READ RO
0106	0x206A	**Rtasknum**	Number of the task to be executed with serial command		0	63	Unsig-ned16	WORD
0450	0x21C2	**Rptroldcmd**	Executed commands buffer pointer 0= Newest 7= Oldest		0	7	Unsig-ned16	WORD NO MAP
0451	0x21C3	**Rbufcmdsta-tus**	Status of the last command. Indicates where the command came from: User program Serial Fieldbus				Unsig-ned16	WORD NO MAP RO
0452-53	0x21C4	**Rbufcmdcom-mand**	Last command executed				Unsig-ned32	H-LWORD NO MAP RO
0454-55	0x21C6	**Rbufcmdpara-meter**	Parameter of the last command executed				Signed32	H-LWORD NO_MAP RO

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0032-33	0x2020	**Rhmaxspc**	Maxumum space in Home	Step	0	0x7FFFFFFF	Unsig-ned32	LS – MSWORD SAVE NO MAP
0034	0x2022	**Rdefinpbls**	Definition of input number for BLS function (back limit switch) 255 = Standard BLS input 254 = BLS input disabled 0-15 = Digital input used for BLS (if the input is not associated to a specific function)		0	255	Unsig-ned16	WORD SAVE
0035	0x2023	**Rdefinpfls**	Definition of input number for FLS function (forward limit switch) 255 = Standard FLS input 254 = FLS input disabled 0-15 = Digital input used for FLS (if the input is not associated to a specific function)		0	255	Unsig-ned16	WORD
0036-37	0x2024	**Rhofs**	Homing offset (Shift of the axis after home routine)	Step	0x80000000	0x7FFFFFFF	Signed32	LS – MSWORD SAVE NO MAP
0038-39	0x2026	**Rhpos**	Axis quota forced after the execution of homing function	Step	0x80000000	0x7FFFFFFF	Signed32	WORD SAVE
0040	0x2028	**Rhcurcoll**	% of nominal current to detect a collision during ho-ming sequence with mechanical limit (FOC_CLOSE)	%	1	100	Unsig-ned16	WORD SAVE MAP-WRITE
0041	0x2029	**Rhtimcoll**	Filter time to detect a collision during homing sequen-ce with mechanical limit (FOC_CLOSE)	ms	0	10000	Unsig-ned16	WORD NOMAP SAVE
0081	0x2051	**Rhtinv**	Stop time of the axis before inverting direction during homing in mS (default=512mS)	ms	0	32767	Unsig-ned16	WORD SAVE NO MAP

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0082	0x2052	**Rhmode**	Homing method (Type of home routine): -16 = Homing on FLS + motor encoder TOP, positive direction -15 = Homing on FLS + motor encoder TOP, negative direction -14 = Homing on FLS, positive direction -13 = Homing on FLS, negative direction -12 = Homing with forward mechanical limit + encoder TOP (only SmartMode and Closed Loop) -11 = Homing with backward mechanical limit + encoder TOP (only SmartMode and Closed Loop) -10 = Homing with forward mechanical limit -9 = Homing with backward mechanical limit -8 = Homing with forward mechanical limit + axis measure (Resets the registers Rlowlim and Rupplim) -7 = Homing with backward mechanical limit + axis measure (Resets the registers Rlowlim and Rupplim) -6 = Homing only with TOP in positive direction -5 = Homing only with TOP in negative direction -4 = Homing with BLS + TOP rising edge, positive direction -3 = Homing with BLS + TOP rising edge, negative direction -2 = Homing only with BLS in positive direction -1 = Homing only with BLS in negative direction 0 = Homing on place 35 = Homing on place (only in CanOpen, for DS402 compatibility) 37 = Homing on place (only in CanOpen, for DS402 compatibility)		-16	37	Signed16	WORD MAP WRITE
0083-84	0x2053	**Rhvh**	Homing speed during the limit switch search	rev*100 /s	0	10000	Signed32	LS– MSWORD MAP WRITE
0085-86	0x2055	**Rhvl**	Homing speed during the 0 point search (Must be a speed in the start/stop range)	rev*100 /s	0	10000	Signed32	LS – MSWORD MAP WRITE
0087-88	0x2057	**Rhacc**	Acceleration/deceleration during homing sequence	rev*10/ s^2	0	200000	Unsig-ned32	LS – MSWORD MAP WRITE
0202	0x20CA	**Rhsts**	Homing sequence error code Bit 0: Home in progress Bit 1: Drive not enabled Bit 2: Maximum space for homing sequence Bit 3: BLS intervention error Bit 4: FLS intervention error Bit 5: Home interrupted Bit 6: Homing sequence not recognized Bit 7: Homing with mechanical limit (with FOC_OPEN) Bit 8: Bit 9: Bit 10: Bit 11: Bit 12: Bit 13: Bit 14: Bit 15:				Unsig-ned16	WORD NO MAP RO

STOP INSTRUCTION

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0024-25	0x2018	**Rshstop**	Shift space in STOP instruction	Step	0	0x7FFFFFFF	Unsig-ned32	LS– MSWORD SAVE NO MAP
0028-29	0x201C	**Rspcstop**	Fixed stop space	Step	0	0x7FFFFFFF	Unsig-ned32	LS – MSWORD SAVE NO MAP
0030-31	0x201E	**Rspcstopcalc**	Stop space of the last stop	Step			Unsig-ned32	LS – MSWORD NO MAP

BESTOP INSTRUCTION (STOP ON BIT EVENT)

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0049	0x2031	**Rbestpflg**	BESTOP command enabling [B0 ..B4]: Bit Number B5: Condition (0 = Low, 1 = High) B6: Type of data (0 = Variable, 1= Register) B7 = Stop on the rising edge of TOP mot B8 = Stop on the falling edge of TOP mot B9 = Stop on the rising edge of TOP ext B10= Stop on the falling edge of TOP ext				Unsig-ned16	WORD MAP WRITE
0050	0x2032	**Rbestppar**	BESTOP parameter, contains the register or variable that generates ESTOP				Unsig-ned16	WORD MAP WRITE

N.B. In order to activate the BEstop functions, it is necessary to raise the bit 14 of the register Rcmdwr (address 0059).

ESTOP INSTRUCTION (STOP ON REGISTER OR VARIABLE VALUE)

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0020-21	0x2014	**Rshestop**	Shift space in ESTOP instruction	Step	0	0x7FFFFFFF	Unsig-ned32	LS – MSWORD ANA_T - SAVE NO MAP
0045	0x202D	**Restpflg**	ESTOP command enabling B7: Type of destination data (0 = Variabie, 1= Regi-ster) B6-B5: Source: 00=Var / 01=Reg / 10=direct value / 11=not allowed. B4-B3-B2-B1: Jump condition 0000 = Equal 0001 = Not Equal 0010 = Higher 0011 = Lower 0100 = Higher or Same 0101 = Lower or Same B0: Free				Unsig-ned16	WORD MAP WRITE
0046-47	0x202E	**Restppar1**	Parameter 1 for ESTOP function				Unsig-ned32	WORD MAP WRITE
0048	0x2030	**Restppar2**	Parameter 2 for ESTOP function				Unsig-ned16	WORD MAP WRITE

N.B. In order to activate the Estop functions, it is necessary to raise the bit 4 of the register Rcmdwr (address 0059).

GEAR INSTRUCTION

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0104	0x2068	**Rgearmul**	Reduction ratio for GEAR instruction (multiplier)		1	32767	Unsig-ned16	WORD SAVE NO MAP
0105	0x2069	**Rgeardiv**	Reduction ratio for GEAR instruction (divider)		1	32767	Unsig-ned16	WORD SAVE NO MAP

ALARMS AND WARNINGS

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0224	0x20E0	**Rpostimeout**	Time for positioning Time-out	ms	0	65535	Unsig-ned16	WORD SAVE
0225	0x20E1	**Rdeadpos**	Dead band in position	Step motore	0	32767	Unsig-ned16	WORD SAVE NO MAP
0226	0x20E2	**Rsettim**	Settling time in closed loop. Time in the range of motor position with requested theoric speed = 0, before the signal of motor in posi-tion.	ms	0	1000	Unsig-ned16	WORD SAVE NO MAP
0227	0x20E3	**Ralarm**	Drive alarms Bit 0: Overcurrent HW (not maskable) Bit 1: Overcurrent SW (not maskable) Bit 2: I2T Bit 3: Positioning error (Closed Loop. Position out of the DeadBand for time Rposalmtime with requested velocity=0) (disabled by default) Bit 4: Following error (Open Loop. Encoder pulses – normalized motor pul-ses greater than Rflwmax) (disabled by default) (Closed Loop. With requested velocity <>0 (Encoder pulses – normalized motor pulses) > Rflwmax for the time Rposalmtime) Bit 5: Overload digital output (not maskable) Bit 6: Overtemperature (not maskable) Bit 7: Overvoltage (not maskable) Bit 8: Undervoltage Bit 9: Motor encoder phasing error (not maskable) Bit 10: Motor phase A disconnected (not maskable) Bit 11: Motor phase B disconnected (not maskable) Bit 12: Positioning Timeout Bit 13: Homing Error Bit 14: Inverted encoder in Smart Mode or Closed Loop Bit 15: Encoder anomaly Note: write 0 into this register to reset the alarms.		0	65535	Unsig-ned16	WORD MAP READ

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0228	0x20E4	**Rwarning**	Drive pre-alarm notifications Bit 0: Overcurrent HW (= alarm) (not maskable) Bit 1: Overcurrent SW (not maskable) Bit 2: I2T Bit 3: Positioning error (Closed Loop. Position out of the DeadBand for time Rposalmtime with requested velocity=0) (disabled by default) Bit 4: Following error (Open Loop. Encoder pulses – normalized motor pulses greater than Rflwwrn) (disabled by default) (Closed Loop. With requested velocity <>0 (Encoder pulses – normalized motor pulses) > Rflwmax for the time Rposwrntime) Bit 5: Overload digital output (not maskable) Bit 6: Overtemperature (not maskable) Bit 7: Overvoltage (not maskable) Bit 8: Undervoltage Bit 9: Current limitated by voltage Bit 10: Saturated regulator Bit 11: Current limit is active Bit 12: Positioning Timeout Bit 13: Free Bit 14: Free Bit 15: Free		0	65535	Unsig-ned16	WORD MAP READ RO
0229	0x20E5	**Rbufalm0**	Alarm buffer 0				Unsig-ned16	WORD NO MAP RO
0230	0x20E6	**Rbufalm1**	Alarm buffer 1				Unsig-ned16	WORD NO MAP RO
0231	0x20E7	**Rbufalm2**	Alarm buffer 2				Unsig-ned16	WORD NO MAP RO
0232	0x20E8	**Rbufalm3**	Alarm buffer 3				Unsig-ned16	WORD NO MAP RO
0233	0x20E9	**Rbufalm4**	Alarm buffer 4				Unsig-ned16	WORD NO MAP RO
0234	0x20EA	**Rbufalm5**	Alarm buffer 5				Unsig-ned16	WORD NO MAP RO
0235	0x20EB	**Rbufalm6**	Alarm buffer 6				Unsig-ned16	WORD NO MAP RO
0236	0x20EC	**Rbufalm7**	Alarm buffer 7				Unsig-ned16	WORD NO MAP RO
0237	0x20ED	**Ralmcnt**	Fault counter		0	65535	Unsig-ned16	WORD NO MAP RO
0238	0x20EE	**Ralmack**	Last alarm acknowledge Bit 0: Alarm Acknowledge Bit 1: Reset alarm counter		0	3	Unsig-ned16	WORD NO MAP
0239	0x20EF	**Rtempalm**	Temperature limit to be reached to activate the alarm. Exceeded this value, an alarm is generated.	°C	0	150	Unsig-ned16	WORD SAVE NO MAP
0240	0x20F0	**Rtensmax**	Maximum voltage limit. Exceeded this value, an alarm is generated.	Volt	0	200	Unsig-ned16	WORD SAVE NO MAP
0241	0x20F1	**Rtensmin**	Minimum voltage limit. Below this value, an alarm is generated.	Volt	0	200	Unsig-ned16	WORD SAVE NO MAP
0242	0x20F2	**Rcurmax**	Maximum current limit.	mA	0	20000	Unsig-ned16	WORD SAVE NO MAP

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0243-244	0x20F3	**Rflwmax**	Maximum delta for the generation of a following alarm.	Imp. Enco-der	0	32000	Signed32	LS – MSWORD SAVE NO MAP
0245-246	0x20F5	**Rmaxi2t**	I2T for alarm		0	9999999	Unsig-ned32	LSWORD – MSWORD SAVE NO MAP
0247	0x20F7	**Rmaskalm**	Alarms mask. If bit=1 the corresponding alarm is masked. Bit 0: Not maskable Bit 1: Not maskable Bit 2: I2T Bit 3: Positioning error (Closed Loop. Position out of the DeadBand for time Rposalmtime with requested velocity=0) (disabled by default) Bit 4: Following error (Open Loop. Encoder pulses – normalized motor pulses greater than Rflwmax) (disabled by default) (Closed Loop. With requested velocity <>0 (Encoder pulses – normalized motor pulses) > Rflwmax for the time Rposalmtime) Bit 5: Not maskable Bit 6: Not maskable Bit 7: Not maskable Bit 8: Sottotensione Bit 9: Not maskable Bit 10: Not maskable Bit 11: Not maskable Bit 12: Positioning Timeout Bit 13: Homing Error Bit 14: Inverted encoder in Smart Mode or Closed Loop Bit 15: Encoder anomaly				Unsig-ned16	WORD SAVE
0248	0x20F8	**Rposalmti-me**	Time before positioning alarm in Closed Loop.	ms	0	32000	Unsig-ned16	WORD SAVE
0249	0x20F9	**Rtempwrn**	Value for overtemperature warning	°C	0	150	Unsig-ned16	WORD SAVE NO MAP
0250	0x20FA	**Rovvwrn**	Value for overvoltage warning	Volt	0	200	Unsig-ned16	WORD SAVE NO MAP
0251	0x20FB	**Runvwrn**	Value for undervoltage warning	Volt	0	200	Unsig-ned16	WORD SAVE NO MAP
0252	0x20FC	**Rovcwrn**	Value for overcurrent warning	mA	0	20000	Unsig-ned16	WORD SAVE NO MAP
0253-254	0x20FD	**Rflwwrn**	Value for following warning	Step	0	32000	Unsig-ned32	LSWORD - MSWORD SAVE NO MAP
0255-256	0x20FF	**Rwrni2t**	Value for I2T warning		0	9999999	Unsig-ned32	LSWORD - MSWORD SAVE NO MAP

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0257	0x2101	**Rmaskwrn**	Warning mask. If bit=1 the corresponding warning is masked. Bit 0: Not maskable Bit 1: Not maskable Bit 2: I2T Bit 3: Positioning error (Closed Loop. Position out of the DeadBand for time Rposalmtime with requested velocity=0) (disabled by default) Bit 4: Following error (Open Loop. Encoder pulses – normalized motor pulses greater than Rflwwrn) (disabled by default) (Closed Loop. With requested velocity <>0 (Encoder pulses – normalized motor pulses) > Rflwmax for the time Rposwrntime) Bit 5: Not maskable Bit 6: Not maskable Bit 7: Not maskable Bit 8: Undervoltage Bit 9: Current limitated by voltage Bit 10: Saturated regulator Bit 11: Current limit is active Bit 12: Positioning Timeout Bit 13: Free Bit 14: Free Bit 15: Free				Unsig-ned16	WORD SAVE
0258	0x2102	**Rposwrn-time**	Time before positioning warning in Closed Loop.	ms	0	32000	Unsig-ned16	WORD SAVE
0259-260	0x2103	**Rflwdisp**	Display the absolute following error. Write 0 in this register to reset the following alarm.	Imp. Enco-der			Unsig-ned32	LSWORD – MSWORD MAP READ
0261-262	0x2105	**Rflwmem**	Display the maximum saved following error. Write 0 in this register to reset.	Imp. Enco-der			Unsig-ned32	LSWORD – MSWORD NO MAP
0263	0x2107	**Rflwtim**	Filter time before the signal of following error.	ms	0	32000	Unsig-ned16	WORD SAVE NO MAP
0266	0x210A	**Rflwmemp**	Store the maximum positive following error at 16 bit. Value goes from 0 to 65535. Write 0 in this location to reset the memory.	Imp. Enco-der	0	65535	Unsig-ned16	WORD NO MAP
0267	0x210B	**Rflwmemn**	Store the maximum negative following error at 16 bit. Value goes from 0 to 65535. Write 0 in this location to reset the memory.	Imp. Enco-der	0	65535	Unsig-ned16	WORD NO MAP
0407-08	0x2197	**Rflwencerr**	Maximum error of counter difference between motor encoder and normalized motor steps to generate an error. This function is similar to the followng error, but it doesn't have a time filter, and it's always active. It's used to detect the encoder absence. If used, it's usually set at a value of 1 or 2 turns of the encoder.	Imp. Enco-der	0	0x7FFFFFFF	Unsig-ned32	WORD
0409	0x2199	**Rposti-meoutwrn**	Time for positioning timeout warning	ms	0	65535	Unsig-ned16	WORD SAVE

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0559-60	0x222F	**Ralarm32**	Drive alarms Bit 0: Overcurrent HW (not maskable) Bit 1: Overcurrent SW (not maskable) Bit 2: I2T Bit 3: Positioning error (Closed Loop. Position out of the DeadBand for time Rposalmtime with requested velocity=0) (disabled by default) Bit 4: Following error (Open Loop. Encoder pulses – normalized motor pulses greater than Rflwmax) (disabled by default) (Closed Loop. With requested velocity <>0 (Encoder pulses – normalized motor pulses) > Rflwmax for the time Rposalmtime) Bit 5: Overload digital output (not maskable) Bit 6: Overtemperature (not maskable) Bit 7: Overvoltage (not maskable) Bit 8: Undervoltage Bit 9: Motor encoder phasing error (not maskable) Bit 10: Motor phase A disconnected (not maskable) Bit 11: Motor phase B disconnected (not maskable) Bit 12: Positioning Timeout Bit 13: Homing Error Bit 14: Inverted encoder in Smart Mode or Closed Loop Bit 15: Encoder anomaly Bit 16: Reserved Bit 17: Reserved Bit 18: Missing Sync (CANopen or EtherCAT) Bit 19: SSP anomaly (Incongruent inputs) Bit 20: VLogic undervoltage alarm Bit 21: SSP anomaly (test on SSP A failed) Bit 22: SSP anomaly (test on SSP B failed) Bit 23: Anomaly on output pin “Drive safety state” Note: write 0 into this register to reset the alarms.		0	0xFFFFFFF	Unsig-ned32	D WORD
0561-62	0x2231	**Rwarning32**	Drive pre-alarm notifications Bit 0: Overcurrent HW (= alarm) (not maskable) Bit 1: Overcurrent SW (not maskable) Bit 2: I2T Bit 3: Positioning error (Closed Loop. Position out of the DeadBand for time Rposalmtime with requested velocity=0) (disabled by default) Bit 4: Following error (Open Loop. Encoder pulses – normalized motor pulses greater than Rflwwrn) (disabled by default) (Closed Loop. With requested velocity <>0 (Encoder pulses – normalized motor pulses) > Rflwmax for the time Rposwrntime) Bit 5: Overload digital output (not maskable) Bit 6: Overtemperature (not maskable) Bit 7: Overvoltage (not maskable) Bit 8: Undervoltage Bit 9: Current limitated by voltage Bit 10: Saturated regulator Bit 11: Current limit is active Bit 12: Positioning Timeout Bit 13: Free Bit 14: Free Bit 15: Free Bit 16: Reserved Bit 17: Reserved Bit 18: Missing Sync (CANopen or EtherCAT) Bit 19: SSP anomaly					

Management of following control in closed loop

When the motor works in closed loop, the motor must be equipped with an encoder, so it is possible to execute a following control between the motor and the encoder.
In closed loop mode, there are two types of alarms generated by the comparison between the encoder position (real)

and the position requested by the program (target).
If a movement of the motor is requested (speed different than 0) the following control in enabled.
When the theoric positioning profile ends, and the requested speed is equal to 0, it is necessary to wait the motor enters in the positioning zone indicated in the register Rdeadpos.
When the motor remains in this positioning zone for the time indicated in the register Rsettim, the flag of positioned motor is activated.
At this point, the positioning control enters in function and the following control is disabled.

The positioning control checks that the motor doesn't exit the zone indicated in the registers Rdeadpos.
In the motor exits this zone, a "motor out of position" timer starts.
After the time set in the register Rposwrntime, the bit 3 of the register Rwarning is activated to notify the waring of "motor out of position". If the motor returns to the positioning zone, the warining is automatically reset.
By setting to 1 the bit 3 of the register Rmaskwrn, this notification is disabled.
If the "motor out of position" timer reaches the value set in the register Rposalmtime, the bit 3 of the register Ralarm is activated, the drive enters into alarm state and is disabled. At this stage, an action must be taken to reset the alarm and to restart drive.
By setting to 1 the bit 3 of the register Rmaskalm, this alarm is disabled.

The following control is a continuous comparison between the actual target quota and the real quota of the encoder.
If the difference between these two values exceeds the value set in the register Rflwwrn, the bit 4 of the register Rwarning is set to 1. If the motor returns inside the range, the warining is automatically reset.
By setting to 1 the bit 4 of the register Rmaskwrn, this notification is disabled.
If the difference between the two values exceeds the value set in the register Rflwmax, a filter time is activated for the time set in the register Rflwtim. After this filter time, if the error still exceeds the value of Rflwmax, the bit 4 of the register Ralarm is activated, the drive enters into alarm state and is disabled. At this stage, an action must be taken to reset the alarm and to restart drive.
By setting to 1 the bit 4 of the register Rmaskalm, this alarm is disabled.

The waring is just an anomaly notification, and it doesn't perform any action.

When the drive is disabled, the following error is reset.
To disable the control of the positioning warning, set to 0 the register Rposwrntime.
To disable the control of the positioning error, set to 0 the register Rposalmtime.
To disable the control of the following warning, set to 0 the register Rflwwrn.
To disable the control of the following error, set to 0 the register Rflwmax.

The following registers must be set in order to use the function of following control.
Setting:
Rmotenc : Motor encoder pulses/revolution
Rflwwrn : Following error absolute maximum value in encoder pulses, to activate the warning notification. The value of this register is stored into the drive.
Rflwmax : Following error absolute maximum value in encoder pulses, to activate the error notification (after the filter time). The value of this register is stored into the drive.
Rflwtim : Filter time before the notification of the following error. The warning is immediately displyed , without filter time. The value of this register is stored into the drive.
Rposalmtime : Time before the notification positioning alarm.
Rposwrntime : Time before the notification positioning warning.
Rmaskwrn (b3) : Disable the notification of the positioning warning in the register Rwarning.
Rmaskalm (b3) : Disable the notification of the positioning alarm in the register Ralarm. Avoids the drive to enter the alarm mode, and the consequent deactivation of the motor.
Rmaskwrn (b4) : Disable the notification of the following warning in the register Rwarning.
Rmaskalm (b4) : Disable the notification of the following alarm in the register Ralarm. Avoids the drive to enter the alarm mode, and the consequent deactivation of the motor.

Displaying:
Rflwdisp : Displays the actual following error in encoder pulses.
Rflwmem : Displays the maximum following error in encoder pulses (absolute value). To reset this value, write 0 in the register.
Rflwmemp : Displays the positive maximum following error in encoder pulses (displayed in positive value). To reset this value, write 0 in the register.
Rflwmemn : Displays the negative maximum following error in encoder pulses (displayed in positive value). To reset this value, write 0 in the register.

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0107	0x206B	**Rhslsi (50us)**	High Speed Limit switch input. Bit 0: High Speed FLS limit switch (forward) Bit 1: High Speed BLS limit switch in (backward) Bit 2: High Speed TOP motor encoder Bit 3: High Speed CH.A motor encoder Bit 4: High Speed CH.B motor encoder Bit 5: High Speed TOP external encoder Bit 6: High Speed CH.A external encoder Bit 7: High Speed CH.B external encoder				Unsig-ned16	WORD MAP READ RO
0108	0x206C	**Rhsinp (50us)**	High speed digital input. Bit 0: High speed digital Input 0 Bit 1: High speed digital Input 1 Bit 2: High speed digital Input 2 Bit 3: High speed digital Input 3 Bit 4: High speed digital Input 4 Bit 5: High speed digital Input 5 Bit 6: High speed digital Input 6 Bit 7: High speed digital Input 7 Bit 8: High speed digital Input 8 Bit 9: High speed digital Input 9 Bit 10: High speed digital Input 10 Bit 11: High speed digital Input 11 Bit 12: High speed digital Input 12 Bit 13: High speed digital Input 13 Bit 14: High speed digital Input 14 Bit 15: High speed digital Input 15				Unsig-ned16	WORD MAP READ RO
0109	0x206D	**Rlsi**	Limit switch input. Bit 0: FLS overtravel (forward) Bit 1: BLS overtravel (backward) Bit 2: TOP motor encoder Bit 3: CH.A motor encoder Bit 4: CH.B motor encoder Bit 5: TOP external encoder Bit 6: CH.A external encoder Bit 7: CH.B external encoder				Unsig-ned16	WORD MAP READ RO
0110	0x206E	**Rdeflsi**	Service inputs “active state” definition (0=active high; 1=active low). Bit 0: FLS overtravel (forward) Bit 1: BLS overtravel (backward) Bit 2: TOP motor encoder. This input is ALWAYS active on the rising edge in case it is used with HOME, ESTOP instructions. Else, it works like other inputs. Bit 3: CH.A motor encoder Bit 4: CH.B motor encoder Bit 5: TOP external encoder Bit 6: CH.A external encoder Bit 7: CH.B external encoder				Unsig-ned16	WORD SAVE NO MAP
0111	0x206F	**Rfillsi**	Limit switch input digital filter time	ms	1	16	Unsig-ned16	WORD SAVE NO MAP
0112	0x2070	**Renflsi**	Limit switch input digital filter time enabling				Unsig-ned16	WORD SAVE NO MAP
0113	0x2071	**Rmemlsi**	Limit switch input memory				Unsig-ned16	WORD NO MAP

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0114	0x2072	**Rinp**	Digital input. Bit 0: Input 0 Bit 1: Input 1 Bit 2: Input 2 Bit 3: Input 3 Bit 4: Input 4 Bit 5: Input 5 Bit 6: Input 6 Bit 7: Input 7 Bit 8: Input 8 Bit 9: Input 9 Bit 10: Input 10 Bit 11: Input 11 Bit 12: Input 12 Bit 13: Input 13 Bit 14: Input 14 Bit 15: Input 15				Unsig-ned16	WORD MAP READ RO
0115	0x2073	**Rdefinp**	Digital input “active state” definition (0=active high; 1=active low). Bit 0: Input 0 .... Bit 15: Input 15				Unsig-ned16	WORD SAVE NO MAP
0116	0x2074	**Rfilinp**	Digital input digital filter time	ms	1	16	Unsig-ned16	WORD SAVE NO MAP
0117	0x2075	**Renfinp**	Filter enabling on digital input Bit 0: Input 0 .... Bit 15: Input 15				Unsig-ned16	WORD SAVE NO MAP
0118	0x2076	**Rmeminp**	Digital input memory Bit 0: Input 0 .... Bit 15: Input 15				Unsig-ned16	WORD NO MAP
0119	0x2077	**Rout**	Digital output Bit0: Output0 Bit1: Output1 Bit2: Output2 Bit3: Output3 Bit4: Output4 Bit5: Output5 Bit6: Output6 Bit7: Output7				Unsig-ned16	WORD MAP WRITE
0120	0x2078	**Rdefout**	Digital output “active state” definition (0=active high; 1=active low). Bit0: Output0 .... Bit7: Output7				Unsig-ned16	WORD SAVE NO MAP
0121	0x2079	**Rfuno0**	Digital output 0 function definition register 0) Normal digital output 1) Drive enabled 2) Drive alarm 3) Synchronized motor (Home executed) 4) Motor in movement 5) Task in progress 6) Alarm I^2/T 7) Motor in position 8) Motor in actual movement (for closed loop) 9) Motore in theoric+actual movement (for closed loop) 10) Command for external brake 11) Signal of changed quota while the drive was disabled (only with encoder) 12) Signal of changed quota while the drive was disabled (only with encoder) + motor in position (When the drive is enabled: output=0 if motor not in position or moved while the drive was disabled / =1 if motor in position and not been moved while the drive was disabled). 13) Virtual motor step signal output 14) Motor direction signal output 15) SSP output		0	14	Unsig-ned16	WORD SAVE NO MAP

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0122	0x207A	**Rfuno1**	Digital output 1 function definition register (See Rfun0 for details)		0	14	Unsig-ned16	WORD SAVE NO MAP
0123	0x207B	**Rfuno2**	Digital output 2 function definition register (See Rfun0 for details)		0	14	Unsig-ned16	WORD SAVE NO MAP
0124	0x207C	**Rfuno3**	Digital output 3 function definition register (See Rfun0 for details)		0	14	Unsig-ned16	WORD SAVE NO MAP
0125	0x207D	**Rfuno4**	Digital output 4 function definition register (See Rfun0 for details)		0	14	Unsig-ned16	WORD SAVE NO MAP
0126	0x207E	**Rfuno5**	Digital output 5 function definition register (See Rfun0 for details)		0	14	Unsig-ned16	WORD SAVE NO MAP
0127	0x207F	**Rfuno6**	Digital output 6 function definition register (See Rfun0 for details)		0	14	Unsig-ned16	WORD SAVE NO MAP
0128	0x2080	**Rfuno7**	Digital output 7 function definition register (See Rfun0 for details)		0	14	Unsig-ned16	WORD SAVE NO MAP
0129	0x2081	**Rfuni0**	Digital input 0 function definition register 0) Normal digital input 1) Enable/disable drive 2) JOG forward 3) JOG backward 4) GO (Quota set in Rpostarg) 5) GOR 6) HOME 7) Bit 0 number of task to be enabled 8) Bit 1 number of task to be enabled 9) Bit 2 number of task to be enabled 10) Bit 3 number of task to be enabled 11) Bit 4 number of task to be enabled 12) Bit 5 number of task to be enabled 13) Bit 6 number of task to be enabled 14) Start task (enable selected task) 15) Alarms reset 16) Quota Line Up 17) Current reduction 18) ABORT 19) STOP 20) GEAR 21) Direction (reverse the JOG direction) 22) Position recovery (only with encoder)		0	22	Unsig-ned16	WORD SAVE NO MAP
0130	0x2082	**Rfuni1**	Digital input 1 function definition register (See Rfuni0 for details)		0	22	Unsig-ned16	WORD SAVE NO MAP
0131	0x2083	**Rfuni2**	Digital input 2 function definition register (See Rfuni0 for details)		0	22	Unsig-ned16	WORD SAVE NO MAP
0132	0x2084	**Rfuni3**	Digital input 3 function definition register (See Rfuni0 for details)		0	22	Unsig-ned16	WORD SAVE NO MAP
0133	0x2085	**Rfuni4**	Digital input 4 function definition register (See Rfuni0 for details)		0	22	Unsig-ned16	WORD SAVE NO MAP
0134	0x2086	**Rfuni5**	Digital input 5 function definition register (See Rfuni0 for details)		0	22	Unsig-ned16	WORD SAVE NO MAP
0135	0x2087	**Rfuni6**	Digital input 6 function definition register (See Rfuni0 for details)		0	22	Unsig-ned16	WORD SAVE NO MAP
0136	0x2088	**Rfuni7**	Digital input 7 function definition register (See Rfuni0 for details)		0	22	Unsig-ned16	WORD SAVE NO MAP

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0297	0x2129	**Rdiginplev**	Selection of digital inputs trigger level. 0=Trigger at 12V (for inputs at 24V) 1=Trigger at 2.5V (for inputs at 5V) The thresholds of the levels selected with this register are set in the registers 306 Rtrginp5v and 307 Rtrginp24v		0	1	Unsigned16	WORD
0298	0x212A	**Rstpoutmaxfreq**	Maxumim frequency in digital output with step function (Theoric rotation step frequency of the motor) If the real frequency is higher than the maximum frequency, the output stops at the maximum frequency. The accumulated pulses are given in output when the motor is already in stop, or when its speed drops to a theoric frequency lower than the maximum frequency.	Hz	0	10000	Unsigned16	WORD
0306	0x2132	**Rtrginp5v**	Set the voltage threshold for inputs at 5V. The save is performed with the command 145 in Rloadsav.	Volt *100	0	1755	Unsigned16	WORD SAVE NO MAP NO RES
0307	0x2133	**Rtrginp24v**	Set the voltage threshold for inputs at 24V. The save is performed with the command 145 in Rloadsav.	Volt *100	0	1755	Unsigned16	WORD SAVE NO MAP NO RES
0442	0x21BA	**Rfuni8**	Digital input function definition		0	22	Unsigned16	WORD SAVE NO MAP
0443	0x21BB	**Rfuni9**	Digital input function definition		0	22	Unsigned16	WORD SAVE NO MAP
0444	0x21BC	**Rfuni10**	Digital input function definition		0	22	Unsigned16	WORD SAVE NO MAP
0445	0x21BD	**Rfuni11**	Digital input function definition		0	22	Unsigned16	WORD SAVE NO MAP
0446	0x21BE	**Rfuni12**	Digital input function definition		0	22	Unsigned16	WORD SAVE NO MAP
0447	0x21BF	**Rfuni13**	Digital input function definition		0	22	Unsigned16	WORD SAVE NO MAP
0448	0x21C0	**Rfuni14**	Digital input function definition		0	22	Unsigned16	WORD SAVE NO MAP
0449	0x21C1	**Rfuni15**	Digital input function definition		0	22	Unsigned16	WORD SAVE NO MAP

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0137	0x2089	**Ranainp**	12 bit analog input register				Signed16	WORD MAP READ RO
0138	0x208A	**Rdefanainp**	Analog input definition register		0	558	Unsig-ned16	WORD SAVE NO MAP
0139	0x208B	**Rmulanainp**	Analog input multiplier register		1	32767	Unsig-ned16	WORD SAVE NO MAP
0140	0x208C	**Rdivanainp**	Analog input divider register		1	32767	Unsig-ned16	WORD SAVE NO MAP
0141	0x208D	**Roffsanainp**	Analog input offset		-32768	32767	Signed16	WORD SAVE NO MAP
0142	0x208E	**Rdeada-nainp**	Analog input dead band	Bit	0	4095	Unsig-ned16	WORD SAVE NO MAP
0143	0x208F	**Ranaout**	10 bit analog output register	Bit	0	1023	Unsig-ned16	WORD MAP WRITE
0144	0x2090	**Rdefanaout**	Parameter definition to be used for Ranaout register		0	558	Unsig-ned16	WORD SAVE NO MAP
0145	0x2091	**Rmulanaout**	Multiplier of the value to be set in the analog output		1	32767	Unsig-ned16	WORD SAVE NO MAP
0146	0x2092	**Rdivanaout**	Division of the value to be set in the analog output		1	32767	Unsig-ned16	WORD SAVE NO MAP
0147	0x2093	**Roffsanaout**	Offset to be added to the value to be set in the analog output		-32768	32767	Signed16	WORD NO MAP SAVE
0313	0x2139	**Rfloatingrol-lerperc**	Only for DMD. Dancer roller percentage (-100/+100)	%	0	100	Unsig-ned16	WORD SAVE
0429	0x21AD	**Ranainpdi-sable**	Disable the analog inputs 0-1-2 with respective bits B0, B1 and B2 high				Unsig-ned16	
0430	0x21AE	**Ranainp1**	Analog input 1 value (only for SMD10.04 and SMD1104)				Signed16	WORD RO MAP READ
0431	0x21AF	**Rdefanainp1**	Analog input 1 function definition (only for SMD10.04 and SMD1104)		0	558	Unsig-ned16	WORD SAVE
0432	0x21B0	**Rmula-nainp1**	Analog input 1 multiplier (only for SMD1004 and SMD1104)		1	32767	Unsig-ned16	WORD SAVE
0433	0x21B1	**Rdivanainp1**	Analog input 1 divider (only for SMD1004 and SMD1104)		1	32767	Unsig-ned16	WORD SAVE
0434	0x21B2	**Roffsa-nainp1**	Analog input 1 offset register (only for SMD10.04 and SMD1104)		-32768	32767	Signed16	WORD SAVE
0435	0x21B3	**Rdeada-nainp1**	Analog input 1 dead band (only for SMD10.04 and SMD1104)	Bit	0	4095	Unsig-ned16	WORD SAVE
0436	0x21B4	**Ranainp2**	Analog input 2 value (only for SMD10.04 and SMD1104)				Signed16	WORD RO MAP READ
0437	0x21B5	**Rdefanainp2**	Analog input 2 function definition(only for SMD10.04 and SMD1104)		0	558	Unsig-ned16	WORD SAVE
0438	0x21B6	**Rmula-nainp2**	Analog input 2 multiplier (only for SMD10.04 and SMD1104)		1	32767	Unsig-ned16	WORD SAVE
0439	0x21B7	**Rdivanainp2**	Analog input 2 divider (only for SMD10.04 and SMD1104)		1	32767	Unsig-ned16	WORD SAVE
0440	0x21B8	**Roffsa-nainp2**	Analog input 2 offset register (only for SMD10.04 and SMD1104)		-32768	32767	Signed16	WORD SAVE

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0441	0x21B9	**Rdeada-nainp2**	Analog input 2 dead band (only for SMD10.04 and SMD1104)	Bit	0	4095	Unsig-ned16	WORD SAVE

Motor encoder

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0151-152	0x2097	**Rmotenc**	Motor encoder quota register	Enc. Pulses			Signed32	LSWORD – MSWORD MAP READ
0163	0x20A3	**Rmotencper**	Period read by the pulses of the motor encoder				Unsig-ned16	WORD
0269	0x210D	**Rmo-tencpuls**	Motor encoder pulses per revolution	Enc. Pulses	1	32767	Unsig-ned16	WORD SAVE NO MAP

External encoder

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0051	0x2033	**Rextencvel**	External encoder actual speed				Unsig-ned16	WORD MAP READ RO
0052	0x2034	**Rextencsmp**	External encoder speed reading sample time	ms	0	10000	Unsig-ned16	WORD SAVE MAP WRITE
0053	0x2035	**Rextencvel-mul**	External encoder speed reading multiplier		0	32767	Unsig-ned16	WORD SAVE MAP WRITE
0054	0x2036	**Rextencvel-div**	External encoder speed reading divider		1	32767	Unsig-ned16	WORD SAVE MAP WRITE
0089	0x2059	**Rextenco-topvel**	Speed detected from TOP input of external encoder. Sample time is set in the register Rextencsmp (52).				Unsig-ned16	WORD
0090	0x205A	**Rexten-ctopvelmul**	Multiplier of the speed read by external encoder TOP input				Unsig-ned16	WORD
0091	0x205B	**Rexten-ctopveldiv**	Divider of the speed read by external encoder TOP input				Unsig-ned16	WORD
0092-93	0x205C	**Rexten-ctopcnt**	External encoder TOP pulses counter				Unsig-ned32	WORD
0101	0x2065	**Rextencmo-de**	Auxiliary encoder operation mode 0 = Forward quadrature 1 = Back quadrature 2 = Step-dir (Steps multiplied by 1) 3 = Step-dir (Steps multiplied by 2)		0	3	Unsig-ned16	WORD SAVE NO MAP
0153-154	0x2099	**Rextenc**	External encoder quota register	Enc. Pulses			Signed32	LSWORD – MSWORD MAP READ
0164	0x20A4	**Rextencper**	Period read by the pulses of the external encoder				Unsig-ned16	WORD
0165	0x20A5	**Rnewencper**	Signal of reading of new encoder pulses period B0= Motor encoder new period B1= External encoder new period				Unsig-ned16	WORD
0268	0x210C	**Rextencpuls**	External encoder pulses per revolution	Enc. Pulses	0	32767	Unsig-ned16	WORD SAVE

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0044	0x202C	**Rssiencframelen**	SSI encoder frame length. Default value is 25.	Bit	1	32	Unsigned16	WORD SAVE
0094	0x205E	**Rssiencturnbit**	Bits of the frame reserved for encoder turns count	Bit	0	16	Unsigned16	WORD RW SAVE
0095	0x205F	**Rssisencountsbit**	Bits of the frame reserved for encoder position inside the turn count	Bit	0	16	Unsigned16	WORD RW SAVE
0096	0x2060	**Rssiencounts**	SSI encoder position on the turn count register N.B. Preset and complement bits are in the Rhwconfig register: BIT6 = Preset BIT7 = Complement		-32768	32767	Unsigned16	WORD MAP READ RO
0097	0x2061	**Rssiencturns**	SSI absolute encoder turns counter register N.B. Preset and complement bits are in the Rhwconfig register: BIT6 = Preset BIT7 = Complement		-32768	32767	Unsigned16	WORD MAP READ RO

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0194	0x20C2	**Rloadsav**	Depending on the value written in this register, the registers block or the variables block will be loaded from or saved into the EEPROM memory The register is reset as the command has been executed. 01 (01h) = Load registers from the Eeprom 02 (02h) = Load variables from the Eeprom 03 (03h) = Load tasks from the Eeprom 04 (04h) = Load current sensors offset from the Eeprom 05 (05h) = Load MAC address from the Eeprom 06 (06h) = Load absolute encoder offset from the Eeprom 07 (07h) = Load current signal amplification from the Eeprom 08 (08h) = Load the trigger levels for digital inputs 09 (09h) = Load the register Rpwmmode 129 (81h) = Save registers in the Eeprom 130 (82h) = Save variables in the Eeprom 131 (83h) = Save tasks in the Eeprom *132 (84h) = Reset the registers block at the default parameters *133 (85h) = Reset the variables block at 0 *134 (86h) = Reset the tasks *135 (87h) =Initialize the EEPROM **137 (89h) = Save current sensors offset **138 (8Ah) = Aquire and save current sensors offset 139 (8Bh) = Apply Ethernet network parameters 141 (8Dh) = Save absolute encoder offset in the Eeprom 142-143 (8Eh-8Fh) = Restart the drive. The commands must be written in sequence consecutively. 144 (90h) = Save current signal amplification in the Eeprom. 145 (91h) = Save the trigger levels for digital inputs * Not executed if the user program is in RUN. ** Executed only if the drive is not enabled, also if the user program is in RUN.		0	65535	Unsig-ned16	WORD NO MAP
0195	0x20C3	**Rmemvar**	By writing in this register, it is possible to load or save single variables in the EEPROM memory. Bit 15 = 1 Load / 0= Save Bit 7-0 =Number of the variable to be loaded or saved (1..128)				Unsig-ned16	WORD NO MAP

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0196	0x20C4	**Reepsts**	Eeprom status 0 = OK B0 = R/W in progress B1 = Not valid command B2 = Error in reading registers B3 = Error in writing registers B4 = Error in reading variables B5 = Error in writing variables B6 = Error in reading program B7 = Error in writing program B8 = Error in reading password B9 = Error in writing password B10 = Error in reading task B11 = Error in writing task B12 = Error in reading alarms buffer B13 = Error in writing alarms buffer B14 = Error in reading current sensors offset B15 = Error in writing current sensors offset				Unsig-ned16	WORD NO MAP RO
0201	0x20C9	**Rindex**	MIL program variables indexing register		1	128	Holding Register (16bit) Unsig-ned16	WORD NO MAP
0329-30	0x2149	**Reepsts32**	32bit EEPROM status (From SMD5106 on) The low 16 bit are equal to the register Reepsts (196).				Unsig-ned32	LS – MSWORD MAP READ

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0216	0x20D8	**Rstrtmode**	Operation mode at power-on 0 = Load the registers 1 = Load the registers and set the operation mode 2 = Load the registers, set the operation mode and enable the drive 3 = Load the registers, set the operation mode, enable the drive and RUN the program		1	3	Unsig-ned16	WORD SAVE NO MAP
0217	0x20D9	**Rstrtconf**	Configuration at power-on When a register saving command is given, if bit 15 of Rconfig =0, the value of Rconfig is copied into this register.		0	13	Unsig-ned16	WORD SAVE NO MAP
0389-90	0x2185	**Rstrtvel**	Value of Rvel set at the power-on	rev*100 /s	-10000	10000	Signed32	LS-MSWORD SAVE NO MAP
0391-92	0x2187	**Rstrtvss**	Value of Rvss set at the power-on	rev*100 /s	0	10000	Unsig-ned32	LS-MSWORD SAVE NO MAP
0393-94	0x2189	**Rstrtacc**	Value of Racc set at the power-on	rev*10 /s²	1	200000	Unsig-ned32	LS-MSWORD SAVE NO MAP
0395-96	0x218B	**Rstrtdec**	Value of Rdec set at the power-on	rev*10 /s²	1	200000	Unsig-ned32	LS-MSWORD SAVE NO MAP
0397-98	0x218D	**Rstrtpostarg**	Value of Rpostarg set at the power-on	Step	0x80000000	0x7FFFFFFF	Signed32	LS-MSWORD SAVE NO MAP
0399-400	0x218F	**Rstrthvh**	Value of Rhvh set at the power-on	rev*100 /s	1	10000	Signed32	LS-MSWORD SAVE NO MAP
0401-02	0x2191	**Rstrthvl**	Value of Rhvl set at the power-on	rev*100 /s	1	10000	Signed32	LS-MSWORD SAVE NO MAP
0403-04	0x2193	**Rstrthacc**	Value of Rhacc set at the power-on	rev*10 /s²	1	200000	Unsig-ned32	LS-MSWORD SAVE NO MAP

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0405	0x2195	**Rstrthmode**	Value of Rhmode set at the power-on -16 = Homing on FLS + motor encoder TOP, positive direction -15 = Homing on FLS + motor encoder TOP, negative direction -14 = Homing on FLS, positive direction -13 = Homing on FLS, negative direction -12 = Homing with forward mechanical limit + encoder TOP (only SmartMode and Closed Loop) -11 = Homing with backward mechanical limit + encoder TOP (only SmartMode and Closed Loop) -10 = Homing with forward mechanical limit (only SmartMode and Closed Loop) -9 = Homing with backward mechanical limit (only SmartMode and Closed Loop) -8 = Homing with forward mechanical limit + axis measuring (Resets the registers Rlowlim and Rupplim) (only SmartMode and Closed Loop) -7 = Homing with backward mechanical limit + axis measuring (Resets the registers Rlowlim and Rupplim) (only SmartMode and Closed Loop) -6 = Homing only with TOP in positive direction -5 = Homing only with TOP in negative direction -4 = Homing with BLS + TOP rising edge, positive direction -3 = Homing with BLS + TOP rising edge, negative direction -2 = Homing only with BLS in positive direction -1 = Homing only with BLS in negative direction 0 = Homing on place		-16	0	Signed16	WORD SAVE NO MAP

If the drive is powered and the rotary switches are set to 00, the communication parameters of the fieldbuses are set to AEC default values.

In detail:

Modbus RS232/RS485:
Address: 125
BaudRate: 9600
Parity: Even
StopBit 1
Modalita': Intel

CanOpen:
Address: 125
BaudRate: 125Kbit

Profibus:
Address: 125
BaudRate: Auto

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0099	0x2063	**Rcu-ronramptime**	Cuttent ramp time at current on	ms	0	65535	Unsig-ned16	WORD SAVE NO MAP
0209	0x20D1	**Rcuract**	Actual current	mA			Signed16	WORD MAP READ RO
0210	0x20D2	**Rcurnom**	Nominal current supplied to the motor	mA	-8500	8500	Unsig-ned16	WORD SAVE MAP WRITE
0211	0x20D3	**Rcurred**	Reduced current supplied to the motor	mA	0	8500	Unsig-ned16	WORD SAVE MAP WRITE
0212	0x20D4	**Rcurboost**	Boost current during ramps	mA	0	10000	Unsig-ned16	WORD SAVE MAP WRITE
0213	0x20D5	**Rcurtorque**	Requested current in TORQUE mode	mA	-10000	10000	Unsig-ned16	WORD MAP WRITE ANA
0214	0x20D6	**Rtboost**	Maximum boost time	ms	0	5000	Unsig-ned16	WORD SAVE NO MAP
0215	0x20D7	**Rtcred**	Time frame before switching to reduced current	ms	0	10000	Unsig-ned16	WORD SAVE NO MAP
0219	0x20DB	**Rfocmode**	Actual configuration of the drive Actual status of the control 0= Not initialized 1 = Open loop 2 = Closed loop 3 = Smart mode (smart closed loop)		0	3	Unsig-ned16	WORD NO MAP RO
0220	0x20DC	**Rcurmode-act**	Actual current level 0 = No current 1 = Reduced current 2 = Nominal current 3 = Current boost 4 = Automatic current reduction		0	4	Unsig-ned16	WORD RO MAP READ
0221	0x20DD	**Rcurmode**	Current control modes 0 = No current 1 = Reduced current 2 = Nominal current 3 = Current boost		0	3	Unsig-ned16	WORD SAVE NO MAP
0222	0x20DE	**Rconfig**	Drive operation mode configuration 0= Not configured 1= Reserved 2= Open Loop / Speed 3= Open Loop / Position 4= Open Loop / Step-Direction 5= Closed Loop / Torque 6= Closed Loop / Speed (with encoder) 7= Closed Loop / Position 8= Closed Loop / Step-Direction 9= Closed Loop / Speed (with tachometer – Only DMD) 10= Reserved 11= Smart Loop / Speed 12= Smart Loop / Position 13= Smart Loop / Step-Direction 14= Reserved 15= Open loop PWM (Only DMD)		0	15	Unsig-ned16	WORD NO MAP

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0223	0x20DF	**Renmask**	Drive enabling control mask Bit 0: Rconfig is set Bit 1: Bit 2: Bit 3: Bit 4: Bit 15: Control mask error				Unsig-ned16	WORD NO MAP RO
0264-265	0x2108	**Rtrqdisp**	Display the torque				Unsig-ned32	LS – MSWORD MAP READ RO
0270	0x210E	**Rstpres**	Motor step resolution		1	1024	Unsig-ned16	WORD SAVE NO MAP
0279	0x2117	**Rkpiq**	Kp PI current Iq		0	32767	Unsig-ned16	WORD SAVE NO MAP
0280	0x2118	**Rkiiq**	Ki PI current Iq		0	32767	Unsig-ned16	WORD SAVE NO MAP
0281	0x2119	**Rkpid**	Kp PI current Id		0	32767	Unsig-ned16	WORD SAVE NO MAP
0282	0x211A	**Rkiid**	Ki PI current Id		0	32767	Unsig-ned16	WORD SAVE NO MAP
0283	0x211B	**Rkpvel**	Kp PI velocity in FOC Close		0	32767	Unsig-ned16	WORD SAVE NO MAP
0284	0x211C	**Rkivel**	Ki PI velocity in FOC Close		0	32767	Unsig-ned16	WORD SAVE NO MAP
0285	0x211D	**Rkcvel**	Kc PI velocity in FOC Close		0	32767	Unsig-ned16	WORD SAVE NO MAP
0286	0x211E	**Rkppos**	Kp PI position in FOC Close		0	32767	Unsig-ned16	WORD SAVE NO MAP
0287	0x211F	**Rkipos**	Ki PI position in FOC Close		0	32767	Unsig-ned16	WORD SAVE NO MAP
0288	0x2120	**Rkcipos**	Kci PI position in FOC Close		0	32767	Unsig-ned16	WORD SAVE NO MAP
0289	0x2121	**Rkffpos**	Kff PI position in FOC Close		0	32767	Unsig-ned16	WORD SAVE NO MAP
0290	0x2122	**Rkafpos**	Kaf PI position in FOC Close		0	32767	Unsig-ned16	WORD SAVE NO MAP
0291	0x2123	**Rswacfw**	Acc. Forward switch choice 0 = Current 1 = Speed		0	1	Unsig-ned16	WORD SAVE NO MAP
0292	0x2124	**Radjpicur**	Percentage of current PI correction from maximum to minimum current level. This permits to keep the PI reactive on the whole range of current without going in oscillation.		-100	100	Unsig-ned16	WORD SAVE NO_MAP
0299	0x212B	**Rstpressd**	Step resolution in step/dir mode The function of this register has been assigned to the register Rstpres from the firmware 3.44 onwards. For firmwares after 3.44, this register has no functions.		1	1024	Unsig-ned16	WORD SAVE NO MAP

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0300	0x212C	**Rkpstpdir**	Kp PI increments management in step/dir mode		0	4096	Unsig-ned16	WORD SAVE NO MAP
0301	0x212D	**Rampfrq**	Amplification of the increment step/dir in FOC close mode from 1 to 10 (1 = 1:1 / 10 = Multiplies the input steps by 10)		0	32767	Unsig-ned16	WORD SAVE NO MAP
0309	0x2135	**Rdefluxen**	Enable FOCOPEN deflux		0	1	Unsig-ned16	WORD SAVE NO MAP
0310	0x2136	**Rdefluxmin**	FOCOPEN minimum deflux current		0	5000	Unsig-ned16	WORD SAVE NO MAP
0311	0x2137	**Rphamode**	Phaase advance FOCOPEN mode		0	9	Unsig-ned16	WORD SAVE NO MAP
0312	0x2138	**Rphagain**	Phase advance FOCOPEN gain		0	8192	Unsig-ned16	WORD SAVE NO MAP
0314	0x213A	**Rveladjo-penmode**	Only for DMD. Speed correction in open loop (armature feedback)		-4096	4096	Unsig-ned16	WORD
0351	0x215F	**Radjcura**	Adjustment of signal amplification read by the current sensor of the phase A		-128	127	Unsig-ned16	WORD SAVE
0352	0x2160	**Radjcurb**	Adjustment of signal amplification read by the current sensor of the phase B		-128	127	Unsig-ned16	WORD SAVE
0370	0x2172	**Rla**	Current read from phase A	mA			Unsig-ned16	WORD MAP RO
0371	0x2173	**Rlb**	Current read from phase B	mA			Unsig-ned16	WORD MAP RO
0372	0x2174	**Rlan**	Filtered current read from phase A	mA			Unsig-ned16	WORD MAP RO
0373	0x2175	**Rlbn**	Filtered current read from phase B	mA			Unsig-ned16	WORD MAP RO
0374	0x2176	**Rla_offs**	Phase A calculated offset	Bit			Unsig-ned16	WORD NO MAP RO
0375	0x2177	**Rlb_offs**	Phase B calculated offset	Bit			Unsig-ned16	WORD NO MAP RO
0376	0x2178	**Rla_offsEl**	Phase A offset manual correction (*) Saved with command 137 in the register Rloadsav. Calculated and saved with command 138 in the regi-ster Rloadsav.	Bit	-32768	32767	Signed16	WORD SAVE (*) NO MAP
0377	0x2179	**Rlb_offsEl**	Phase B offset manual correction (*) Saved with command 137 in the register Rloadsav. Calculated and saved with command 138 in the regi-ster Rloadsav.	Bit	-32768	32767	Signed16	WORD SAVE (*) NO MAP
0378	0x217A	**Rid**	Direct current	mA			Unsig-ned16	WORD NO MAP RO
0379	0x217B	**Riq**	Current in quadrature	mA			Unsig-ned16	WORD MAP RO
0495	0x21EF	**Rpwmmode**	Motor pwm current generation mode: 0=Centered 1=Left aligned Save the data with the command 146 in Rloadsav, shut down and power-on the drive.		0	1	Unsig-ned16	WORD

MOTOR PARAMETERS

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0218	0x20DA	**Rmottype**	Index of the configured motor. Indicates the index of the configured motor in the motors database.				Unsig-ned16	WORD SAVE NO MAP
0271	0x210F	**Rmotres**	Phase resistance of the motor	ohm *10	1	32767	Unsig-ned16	WORD SAVE NO MAP
0272	0x2110	**Rmotind**	Phase inductance of the motor	mHe *10	1	32767	Unsig-ned16	WORD SAVE NO MAP
0273	0x2111	**Rmotkfm**	F.c.e.m. constant L(mH)*Inom(mA)*100/1000	mHA *100	1	32767	Unsig-ned16	WORD SAVE NO MAP
0274	0x2112	**Rmottens-nom**	Only for DMD. Nominal voltage of the motor.	Volt	1	32767	Unsig-ned16	WORD SAVE
0275	0x2113	**Rmotiph**	Motor nominal phase current	mA	1	32767	Unsig-ned16	WORD SAVE NO MAP
0276	0x2114	**Rmotktq**	Motor torque constant	mNm /A	1	65535	Unsig-ned16	WORD SAVE NO MAP
0277	0x2115	**Rmotvelnom**	Only for DMD. Nominal speed of the motor.	rev/s			Unsig-ned16	WORD SAVE

TIMER

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0155-156	0x209B	**Rtim0**	Register decreased by 1 down to 0 every 1 mS	ms			Unsig-ned32	LSWORD – MSWORD NO MAP
0157-158	0x209D	**Rtim1**	Register decreased by 1 down to 0 every 1 mS	ms			Unsig-ned32	LSWORD – MSWORD NO MAP
0159-160	0x209F	**Rtim2**	Register decreased by 1 down to 0 every 1 mS	ms			Unsig-ned32	LSWORD – MSWORD NO MAP
0161-162	0x20A1	**Rtim3**	Register decreased by 1 down to 0 every 1 mS	ms			Unsig-ned32	LSWORD – MSWORD NO MAP
0536-37	0x2218	**Rtim4**	Register decreased by 1 down to 0 every 1 mS	ms	0		Unsig-ned32	WORD
0538-39	0x221A	**Rtim5**	Register decreased by 1 down to 0 every 1 mS	ms	0		Unsig-ned32	WORD
0540-41	0x221C	**Rtim6**	Register decreased by 1 down to 0 every 1 mS	ms	0		Unsig-ned32	WORD
0542-43	0x221E	**Rtim7**	Register decreased by 1 down to 0 every 1 mS	ms	0		Unsig-ned32	WORD

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Mini-mum value	Maximum value	Type	Note	Note
0148	0x2094	**Rtempact**	Actual temperature of the drive		°C			Signed16	WORD MAP RO
0149	0x2095	**Rtensact**	Actual voltage of the drive CC bus		Volt			Unsig-ned16	WORD MAP RO
0150	0x2096	**Rcurdcact**	Actual current requested by the drive to the DC power supply		mA			Unsig-ned16	WORD NO MAP RO
0190	0x20BE	**Rswrev**	Software revision					Unsig-ned16	WORD NO MAP RO
0191	0x20BF	**Rhwrev**	Hardware revision					Unsig-ned16	WORD NO MAP RO
0192-193	0x20C0	**Rserial**	Serial number of the device					Unsig-ned32	LS – MSWORD SAVE NO MAP
0197	0x20C5	**Rrotsw**	Copy of the value read by the rotary switch			0	99	Unsig-ned16	WORD NO MAP RO
0343	0x2157	**Rhwoptions**	Active hardware options B1= SSP present					Unsig-ned16	WORD RO

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Mini-mum value	Maximum value	Type	Note	Note
0198	0x20C6	**Rindtype**	Tipo indexer 0: SMD30.06LIM 1: SMD30.06LIC 2: SMD30.06LIP 3: SMD50.06LIM 4: SMD50.06LIC 5: SMD50.06LIP 6: SMD30.06HIM 7: SMD30.06HIC 8: SMD30.06HIP 9: SMD50.06HIM 10: SMD50.06HIC 11: SMD50.06HIP 12: SMD10.04LIM 13: SMD10.04LIC 14: SMD10.04LIP 15: SMD10.04HIM 16: SMD10.04HIC 17: SMD10.04HIP 18: SMD50.06LS 19: SMD50.06HS 20: SMD30.06LS 21: SMD30.06HS 22: SMD10.04LS 23: SMD10.04HS 24: SMD10.04LUM 25: SMD10.04HUM 26: SMD104u 27: SMD30.06LIE 28: SMD50.06LIE 29: SMD30.06HIE 30: SMD50.06HIE 31: SMD10.04LIE 32: SMD10.04HIE 33: SMD11.04LIM 34: SMD11.04LIC 35: SMD11.04LIP 36: SMD11.04LS 37: SMD11.04LUM 38: SMD11.04LIE 39: SMD11.04HIM 40: SMD11.04HIC 41: SMD11.04HIP 42: SMD11.04HS 43: SMD11.04HUM 44: SMD11.04HIE 45: SMD114u 46: SMD51.06LIM 47: SMD51.06LIC 48: SMD51.06LIP 49: SMD51.06HIM 50: SMD51.06HIC 51: SMD51.06HIP 52: SMD51.06LS 53: SMD51.06HS 54: SMD51.06LIE 55: SMD51.06HIE 56: SMD51.06LUM 57: SMD51.06HUM 58: SMD51.06LIT 59: SMD51.06HIT 60: SMD11.04LIT 61: SMD11.04HIT 62: SMD31.06LIM 63: SMD31.06LIC 64: SMD31.06LIP 65: SMD31.06HIM 66: SMD31.06HIC 67: SMD31.06HIP 68: SMD31.06LS 69: SMD31.06HS 70: SMD31.06LIE 71: SMD31.06HIE 72: SMD31.06LUM 73: SMD31.06HUM 74: SMD31.06LIT 75: SMD31.06HIT 76:SMD12.04LIM 77: SMD12.04LIC	78: SMD12.04LIP 79: SMD12.04LS 80: SMD12.04LUM 81: SMD12.04LIE 82: SMD12.04HIM 83: SMD12.04HIC 84: SMD12.04HIP 85: SMD12.04HS 86: SMD12.04HUM 87: SMD12.04HIE 88: SMD12.04LIT 89: SMD12.04HIT 90: SMD22.04LIM 91: SMD22.04LIC 92: SMD22.04LIP 93: SMD22.04HIM 94: SMD22.04HIC 95: SMD22.04HIP 96: SMD22.04LS 97: SMD22.04HS 98: SMD22.04LIE 99: SMD22.04HIE 100: SMD22.04LUM 101: SMD22.04HUM 102: SMD22.04LIT 103: SMD22.04HIT 104: SMD1204HIN 105: SMD1204LIN 106: SMD5106HIN 107: SMD5106LIN 108: SMD2204HIN 109: SMD2204LIN				Holding Register (16bit) Unsig-ned16	WORD NO MAP RO

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0042-43	0x202A	**Rtestintmot**	Register for the control of the communication mode (32bit Intel or Motorola). The fixed value of this register is 1234567890 (0x499602D2) By executing a read of this register at 32bit, it is possible to know if the communication is configured in Inter or Motorola mode, depending ont he result.		1234567890 0x499602D2	1234567890 0x499602D2	Holding Register (16bit)	WORD RO
0166	0x20A6	**Rprofists**	Profibus communication status 0 = Disabled 1 = Parameterization 2 = Configuration 3 = Data Exchange FF = Fatal Error				Unsig-ned16	WORD NO MAP RO
0167	0x20A7	**Rprofibaud**	0: Auto baud-rate 1: 9.6KB 2: 19.2KB 3: 31.25KB 4: 45.45KB 5: 93.75KB 6: 187.5KB 7: 500KB 8: 1500KB 9: 3000KB 10: 6000KB 11: 12000KB		0	11	Unsig-ned16	WORD SAVE NO MAP
0168	0x20A8	**Rprofiaddr**	Profibus Address This value is added to the hardware address.		0	127	Unsig-ned16	WORD SAVE NO MAP
0169	0x20A9	**Rcanbaud**	Can Baud Rate 0 = 10Kb 1 = 20Kb 2 = 50Kb 3 = 125Kb 4 = 250Kb 5 = 500Kb 6 = 800Kb 7 = 1Mb		0	8	Unsig-ned16	WORD SAVE NO MAP
0170	0x20AA	**Rcanaddr**	Can Address This value is added to the hardware address.		0	127	Unsig-ned16	WORD SAVE NO MAP
0171-172	0x20AB	**Rcantx**	Number of messages transmitted in Can				Unsig-ned32	LS – MSWORD NO MAP RO
0173-174	0x20AD	**Rcanrx**	Number of messages received in Can				Unsig-ned32	LS – MSWORD NO MAP RO
0175	0x20AF	**Rcanovr**	Number of overrun errors in Can				Unsig-ned16	WORD NO MAP RO
0176	0x20B0	**Rcanerr**	Number of errors in Can				Unsig-ned16	WORD NO MAP RO
0177	0x20B1	**Rcanidx**	Index of the CANopen object to be read or written		0	65535	Unsig-ned16	WORD NO MAP
0178	0x20B2	**Rcansub**	Sub-index of the CANopen object to be read or written		0	255	Unsig-ned16	WORD NO MAP
0179-180	0x20B3	**Rcannew**	New value to be written in CanOpen object				Unsig-ned32	LS – MSWORD NO MAP

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0181	0x20B5	**Rcancmd**	CANopen object Read/Write command 1 = Write value 2 = Read value		0	2	Unsig-ned16	WORD NO MAP
0182-183	0x20B6	**Rcanact**	Read value from CANopen object				Unsig-ned32	LS – MSWORD NO MAP RO
0184	0x20B8	**Rcansts**	CanOpen status Bit 0..3 status LED RED b3 b2 b1 b0 0 0 0 0 Disabled 0 0 1 1 Warning Limit reached 0 1 0 0 Error control event 0 1 0 1 Sync Error 0 1 1 0 Event timer error 0 1 1 1 Bus OFF Bit 4..7 status LED GREEN b7 b6 b5 b4 0 0 0 1 Pre-Operational 0 0 1 0 Stopped 0 1 0 1 Oper				Unsig-ned16	WORD NO MAP RO
0185	0x20B9	**Rserbaud**	Serial port Baud Rate 0 = 1200 1 = 2400 2 = 4800 3 = 9600 4 = 19200 5 = 38400 6 = 57600 7 = 115200		0	7	Unsig-ned16	WORD SAVE NO MAP
0186	0x20BA	**Rserpar**	Serial port parameter (par, stop bit) 0 = NONE, 1 1 = EVEN, 1 2 = ODD, 1 3 = NONE, 2 4 = EVEN, 2 5 = ODD, 2		0	5	Unsig-ned16	WORD SAVE NO MAP
0187	0x20BB	**Rserdly**	Serial Reply delay	ms	0	16	Unsig-ned16	WORD SAVE NO MAP
0188	0x20BC	**Rseraddr**	Serial port address (this value is added to the hardware address)		0	31	Unsig-ned16	WORD SAVE NO MAP
0189	0x20BD	**Rintmot**	Intel/Motorola mode selection for 32bit registers RS232/RS485 serial ports		0	1	Unsig-ned16	WORD SAVE NO MAP
0302	0x212E	**Rfinsenable**	Enable FINS/TCP protocol in drives with Ethernet communication		0	1	Unsig-ned16	WORD SAVE NO MAP
0303	0x212F	**Rfinsnode**	FINS/TCP station number		1	254	Unsig-ned16	WORD SAVE NO MAP
0304	0x2130	**Rmdbport**	TCP port number for Modbus/TCP protocol (Default = 502)		0	65535	Unsig-ned16	WORD SAVE NO MAP
0305	0x2131	**Rethintmot**	Definition of Intel or Motorola mode for registers at 32bit in Modbus/TCP communication or Ethernet based protocols 0= Intel 1=Motorola		0	1	Unsig-ned16	WORD SAVE NO MAP

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0308	0x2134	**Rextmdbti-me**	Modbus RTU inter-message additional time, expres-sed in ms. It's used in case of slow remote commuincation. Default value is 0. If a value <>0 is set, the communication speed slows down.	ms	0	10000	Holding Register (16bit)	WORD SAVE NO MAP
0353	0x2161	**Rstatuscan**	Controlword-Statusword congruence errors in CanO-pen				Unsig-ned16	WORD NO MAP
0354	0X2162	**Rcanmodeo-foperation**	Copy of the object 0x6060				Unsig-ned16	WORD NO MAP
0355	0x2163	**Rswrevan-taios**	Antaios firmware version				Unsig-ned16	WORD NO MAP
0356	0x2164	**Rhwrevan-taios**	Antaios hardware firmware				Unsig-ned16	WORD NO MAP
0357	0x2165	**Rethercatid**	Explicit board ID Ethercat				Unsig-ned16	WORD NO MAP
0358	0x2166	**Rstrtmode-sofoperation**	Setting of the object 0x6060 at power-on, for CANo-pen and EtherCAT protocols		-1	8	Unsig-ned16	WORD SAVE
0365	0x216D	**Rds402com-patibility**	Bit used to adapt the DS402 stack to the Master Bit 0: 1= Current off with Controlword xxx7 (OMRON motion blocks) Bit 1: 1= CSP and CSV active with Controlword xxxF (OMRON motion blocks)					
0515	0x2203	**Rethlocipad-dr32**	Byte 3 - Byte 2 Ethernet local IP address		0	65535	Unsig-ned16	WORD SAVE
0516	0x2204	**Rethlocipad-dr10**	Byte 1 - Byte 0 Ethernet local IP address		0	65535	Unsig-ned16	WORD SAVE
0517	0x2205	**Rethsub-net32**	Byte 3 - Byte 2 Subnet ethernet		0	65535	Unsig-ned16	WORD SAVE
0518	0x2206	**Rethsub-net10**	Byte 1 - Byte 0 Subnet ethernet		0	65535	Unsig-ned16	WORD SAVE
0519	0x2207	**Rethgwad-dr32**	Byte 3 - Byte 2 Gateway address ethernet		0	65535	Unsig-ned16	WORD SAVE
0520	0x2208	**Rethgwad-dr10**	Byte 1 – Byte 0 Gateway address ethernet		0	65535	Unsig-ned16	WORD SAVE
0521	0x2209	**Rethmacad-dr054**	Byte 5 - Byte 4 MAC address 0 ethernet		0	65535	Unsig-ned16	WORD EEPROM SAVE
0522	0x220A	**Rethmacad-dr032**	Byte 3 - Byte 2 MAC address 0 ethernet		0	65535	Unsig-ned16	WORD EEPROM SAVE
0523	0x220B	**Rethmacad-dr010**	Byte 1 - Byte 0 MAC address 0 ethernet		0	65535	Unsig-ned16	WORD EEPROM SAVE
0524	0x220C	**Rethmacad-dr154**	Byte 5 - Byte 4 MAC address 1 ethernet (Profinet port 1)		0	65535	Unsig-ned16	WORD
0525	0x220D	**Rethmacad-dr132**	Byte 3 - Byte 2 MAC address 1 ethernet (Profinet port 1)		0	65535	Unsig-ned16	WORD
0526	0x220E	**Rethmacad-dr110**	Byte 1 - Byte 0 MAC address 1 ethernet (Profinet port 1)		0	65535	Unsig-ned16	WORD
0527	0x220F	**Rethmacad-dr254**	Byte 5 - Byte 4 MAC address 2 ethernet (Profinet port 2)		0	65535	Unsig-ned16	WORD
0528	0x2210	**Rethmacad-dr232**	Byte 3 - Byte 2 MAC address 2 ethernet (Profinet port 2)		0	65535	Unsig-ned16	WORD
0529	0x2211	**Rethmacad-dr210**	Byte 1 - Byte 0 MAC address 2 ethernet (Profinet port 2)		0	65535	Unsig-ned16	WORD
0544	0x2220	**Rethprot-sock1**	Protocol for socket 1 0=TCP / 1=UDP		0	1	Unsig-ned16	WORD SAVE
0545	0x2221	**Rethprot-sock2**	Protocol for socket 2 0=TCP / 1=UDP		0	1	Unsig-ned16	WORD SAVE
0546	0x2222	**Rethprot-sock3**	Protocol for socket 3 0=TCP / 1=UDP		0	1	Unsig-ned16	WORD SAVE

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0547	0x2223	**Rethprot-sock4**	Reserved		0	1	Unsig-ned16	WORD SAVE
0548	0x2224	**Rethprot-sock5**	Reserved		0	1	Unsig-ned16	WORD SAVE
0549	0x2225	**Rethprot-sock6**	Reserved		0	1	Unsig-ned16	WORD SAVE
0550	0x2226	**Rethprot-sock7**	Reserved		0	1	Unsig-ned16	WORD SAVE
0551	0x2227	**Rethport-sock1**	Socket port 1		0	65535	Unsig-ned16	WORD SAVE
0552	0x2228	**Rethport-sock2**	Socket port 2		0	65535	Unsig-ned16	WORD SAVE
0553	0x2229	**Rethport-sock3**	Socket port 3		0	65535	Unsig-ned16	WORD SAVE
0554	0x222A	**Rethport-sock4**	Socket port 4 (only for SMD2204)		0	65535	Unsig-ned16	WORD SAVE
0555	0x222B	**Rethport-sock5**	Socket port 5 (only for SMD2204)		0	65535	Unsig-ned16	WORD SAVE
0556	0x222C	**Rethport-sock6**	Socket port 6 (only for SMD2204)		0	65535	Unsig-ned16	WORD SAVE
0557	0x222D	**Rethport-sock7**	Socket port 7 (only for SMD2204)		0	65535	Unsig-ned16	WORD SAVE
0558	0x222E	**Rethtcpti-meout**	TCP socket timeout time in absence of activities. With value=0 the timeout is disabled.	s	0	65535	Unsig-ned16	WORD SAVE

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
8200		**Rfieldbusdatarx1**	Word 1 Data received via fieldbus				Unsigned16	WORD
8201		**Rfieldbusdatarx2**	Word 2 Data received via fieldbus				Unsigned16	WORD
8202		**Rfieldbusdatarx3**	Word 3 Data received via fieldbus				Unsigned16	WORD
8203		**Rfieldbusdatarx4**	Word 4 Data received via fieldbus				Unsigned16	WORD
8204		**Rfieldbusdatarx5**	Word 5 Data received via fieldbus				Unsigned16	WORD
8205		**Rfieldbusdatarx6**	Word 6 Data received via fieldbus				Unsigned16	WORD
8206		**Rfieldbusdatarx7**	Word 7 Data received via fieldbus				Unsigned16	WORD
8207		**Rfieldbusdatarx8**	Word 8 Data received via fieldbus				Unsigned16	WORD
8208		**Rfieldbusdatarx9**	Word 9 Data received via fieldbus				Unsigned16	WORD
8209		**Rfieldbusdatarx10**	Word 10 Data received via fieldbus				Unsigned16	WORD
8210		**Rfieldbusdatarx11**	Word 11 Data received via fieldbus				Unsigned16	WORD
8211		**Rfieldbusdatarx12**	Word 12 Data received via fieldbus				Unsigned16	WORD
8212		**Rfieldbusdatarx13**	Word 13 Data received via fieldbus				Unsigned16	WORD
8213		**Rfieldbusdatarx14**	Word 14 Data received via fieldbus				Unsigned16	WORD
8214		**Rfieldbusdatarx15**	Word 15 Data received via fieldbus				Unsigned16	WORD
8215		**Rfieldbusdatarx16**	Word 16 Data received via fieldbus				Unsigned16	WORD
8216		**Rfieldbusdatarx17**	Word 17 Data received via fieldbus				Unsigned16	WORD
8217		**Rfieldbusdatarx18**	Word 18 Data received via fieldbus				Unsigned16	WORD
8218		**Rfieldbusdatarx19**	Word 19 Data received via fieldbus				Unsigned16	WORD
8219		**Rfieldbusdatarx20**	Word 20 Data received via fieldbus				Unsigned16	WORD
8220		**Rfieldbusdatarx21**	Word 21 Data received via fieldbus				Unsigned16	WORD
8221		**Rfieldbusdatarx22**	Word 22 Data received via fieldbus				Unsigned16	WORD
8222		**Rfieldbusdatarx23**	Word 23 Data received via fieldbus				Unsigned16	WORD
8223		**Rfieldbusdatarx24**	Word 24 Data received via fieldbus				Unsigned16	WORD
8224		**Rfieldbusdatarx25**	Word 25 Data received via fieldbus				Unsigned16	WORD
8225		**Rfieldbusdatarx26**	Word 26 Data received via fieldbus				Unsigned16	WORD
8226		**Rfieldbusdatarx27**	Word 27 Data received via fieldbus				Unsigned16	WORD
8227		**Rfieldbusdatarx28**	Word 28 Data received via fieldbus				Unsigned16	WORD
8228		**Rfieldbusdatarx29**	Word 29 Data received via fieldbus				Unsigned16	WORD

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
8229		**Rfieldbusdatarx30**	Word 30 Data received via fieldbus				Unsigned16	WORD
8230		**Rfieldbusdatarx31**	Word 31 Data received via fieldbus				Unsigned16	WORD
8231		**Rfieldbusdatarx32**	Word 32 Data received via fieldbus				Unsigned16	WORD
8232		**Rfieldbusdatatx1**	Word 1 Data sent to fieldbus				Unsigned16	WORD
8233		**Rfieldbusdatatx2**	Word 2 Data sent to fieldbus				Unsigned16	WORD
8234		**Rfieldbusdatatx3**	Word 3 Data sent to fieldbus				Unsigned16	WORD
8235		**Rfieldbusdatatx4**	Word 4 Data sent to fieldbus				Unsigned16	WORD
8236		**Rfieldbusdatatx5**	Word 5 Data sent to fieldbus				Unsigned16	WORD
8237		**Rfieldbusdatatx6**	Word 6 Data sent to fieldbus				Unsigned16	WORD
8238		**Rfieldbusdatatx7**	Word 7 Data sent to fieldbus				Unsigned16	WORD
8239		**Rfieldbusdatatx8**	Word 8 Data sent to fieldbus				Unsigned16	WORD
8240		**Rfieldbusdatatx9**	Word 9 Data sent to fieldbus				Unsigned16	WORD
8241		**Rfieldbusdatatx10**	Word 10 Data sent to fieldbus				Unsigned16	WORD
8242		**Rfieldbusdatatx11**	Word 11 Data sent to fieldbus				Unsigned16	WORD
8243		**Rfieldbusdatatx12**	Word 12 Data sent to fieldbus				Unsigned16	WORD
8244		**Rfieldbusdatatx13**	Word 13 Data sent to fieldbus				Unsigned16	WORD
8245		**Rfieldbusdatatx14**	Word 14 Data sent to fieldbus				Unsigned16	WORD
8246		**Rfieldbusdatatx15**	Word 15 Data sent to fieldbus				Unsigned16	WORD
8247		**Rfieldbusdatatx16**	Word 16 Data sent to fieldbus				Unsigned16	WORD
8248		**Rfieldbusdatatx17**	Word 17 Data sent to fieldbus				Unsigned16	WORD
8249		**Rfieldbusdatatx18**	Word 18 Data sent to fieldbus				Unsigned16	WORD
8250		**Rfieldbusdatatx19**	Word 19 Data sent to fieldbus				Unsigned16	WORD
8251		**Rfieldbusdatatx20**	Word 20 Data sent to fieldbus				Unsigned16	WORD
8252		**Rfieldbusdatatx21**	Word 21 Data sent to fieldbus				Unsigned16	WORD
8253		**Rfieldbusdatatx22**	Word 22 Data sent to fieldbus				Unsigned16	WORD
8254		**Rfieldbusdatatx23**	Word 23 Data sent to fieldbus				Unsigned16	WORD
8255		**Rfieldbusdatatx24**	Word 24 Data sent to fieldbus				Unsigned16	WORD
8256		**Rfieldbusdatatx25**	Word 25 Data sent to fieldbus				Unsigned16	WORD
8257		**Rfieldbusdatatx26**	Word 26 Data sent to fieldbus				Unsigned16	WORD
8258		**Rfieldbusdatatx27**	Word 27 Data sent to fieldbus				Unsigned16	WORD

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
8259		**Rfieldbusdatx28**	Word 28 Data sent to fieldbus				Unsigned16	WORD
8260		**Rfieldbusdatx29**	Word 29 Data sent to fieldbus				Unsigned16	WORD
8261		**Rfieldbusdatx30**	Word 30 Data sent to fieldbus				Unsigned16	WORD
8262		**Rfieldbusdatx31**	Word 31 Data sent to fieldbus				Unsigned16	WORD
8263		**Rfieldbusdatx32**	Word 32 Data sent to fieldbus				Unsigned16	WORD

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0413	0x219D	**Rmodulcmd**	Operation mode with module cyclic quota B0 : Forward module B1 : Backward module B2 : Minimum distance module B3 : Motor encoder module B4 : External encoder module		0	65535	Unsig-ned16	WORD SAVE NO_MAP
0414-415	0x219E	**Rmodulpos**	Module quota value for positioner	Step	1	0x7FFFFFFF	Unsig-ned32	H- LWORD SAVE NO_MAP
0416-417	0x21A0	**Rmodulmo-tenc**	Module quota value for motor encoder	Enc. Pulses	1	0x7FFFFFFF	Unsig-ned32	H-LWORD SAVE NO_MAP
0418-419	0x21A2	**Rmodu-lextenc**	Module quota value for external encoder	Enc. Pulses	1	0x7FFFFFFF	Unsig-ned32	H-LWORD SAVE NO_MAP

Management of module quota function (cyclic or Rollover)

With the register Rmodulcmd it is possible to enable the management of the quota in cyclic mode (or rollover).
The quota of the positioner is closed in a loop between a minimum value of 0 and the maximum limit set in the registrer Rmodulpos. Through the bits 3 and 4 of the module Rmodulcmd it is possible to do the same for the motor encoder (Rmotenc) or the external encoder (Rextenc).

The register Rmodulcmd is amanged at bit. The bits have the followin meanings:
B0: Positioner quota in module with always forward direction.
B1: Positioner quota in module with always backward direction.
B2: Positioner quota in module with direction of the movement calculated to perform the minimum distance.
B3: Module quota value for motor encoder.
B4: Module quota value for external encoder.

The quota in module is used in the management of rotary tables, where there is an accurate angular position of the table and the positions cyclically repeat themselves.
With this mode it is possible to indicate an absolute quota inside the module range, and that quota is reached, even if the table must execute a “rollover” of the quota, both forward or backward.

By using the mode “always forward direction”(BIT0=1), each quota is reached by letting the motor (or the table) rotate always in forward direction. As an example, if we are at quota 500 and we give a GO at quota 400, the new position is reached by arriving at the maximum quota of the module, then by resetting it at 0 when the 0 point is reached, and then by advancing until the quota reaches out 400.

By using the mode “always backward direction” (BIT1=1), the sequence is the same as above, but with the rotation direction always set “backward”. So, if we are at quota 500 and we give a go at 600, the new position is reached by reaching quota 0, then by executing the rollover on the maximum quota of the module, and then by proceeding with the back direction until quota 600 is reached.
The mode “minumim distance” (BIT2=1), before starting the movement, executes a check of which is the direction that brigs to the achievement of the desired quota by following the shortest path.

Once the control has decidedthe sense of rotation of the motor (or the table), the operation mode is restored to mode “always forward” or “always backward”.
Regarding encoders, the range of the quotas is always between 0 and a maximum value indicated in the preset encoder module register.
The quotas of he motor encoder and of the external encoder are recorded in the usual registers Rmotenc and Rextenc.

The registers used in this operation mode are:
Rmodulcmd: Module opration mode selection register.
Rmodulpos: Module quota for the positioner. It’s the maximum reachable quota from the positioner before executing the rollover of the quota.
Rmodulmotenc: Module quota for the motor encoder. It’s the maximum reachable quota from the register Rmotenc before executing the rollover of the quota.
Rmodulextenc: Module quota for the motor encoder. It’s the maximum reachable quota from the register Rextenc before executing the rollover of the quota.

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0420	0x21A4	**Rlineupcmd**	Quotas realignment command B0 : Realign positioner quota B1 : Realign motor encoder quota B2 : Realign external encoder quota				Unsig-ned16	WORD MAP WRITE
0421-422	0x21A5	**Rlineuppos**	Positioner realign quota value	Step			Signed 32	H-LWORD SAVE NO_MAP
0423-424	0x21A7	**Rlineupmo-tenc**	Motor encoder realign quota value	Enc. Pulses			Signed32	H-LWORD SAVE NO_MAP
0425-426	0x21A9	**Rlineu-pextenc**	External encoder realign quota value	Enc. Pulses			Signed 32	H-LWORD SAVE NO_MAP
0427	0x21AB	**Rlineupdef**	Enable quotas realignament from an external com-mand (input)When the external digital command arri-ves, these bit are copied in the register Rlineupcmd. B0 : Rialign positioner quota B1 : Rialign motor encoder quota B2 : Rialign external encoder quota				Unsig-ned16	WORD SAVE MAP WRITE

Quota realignment management

It is possible to realign the actual requested quota, the motor encoder quota and the external encoder quota to a prefi-xed values by acting on the register Rlineupcmd.

Rlineupcmd is managed at bit. The bits have the followin meanings:
B0: Realign positioner quota to register Rposactreq.
B1: Realign motor encoder quota to register Rmotenc.
B2: Realign external encoder quota to register Rextenc.

Once the realignment is completed, the register Rlineupcmd is reset to 0, and it waits for another realignment com-mand.

The realignment bits must be transferred all together, in order to avoid the possible loss of realignment requests.

The registers which contain the realignment quota must be already set before giving the realignment.

Essentially, the realignment function copies the lineup registers on the work registers:

Rlineuppos ⟶ Rposactreq
Rlineupmotenc ⟶ Rmotenc
Rlineupextenc ⟶ Rextenc

During the realignment, the interrupts of the DSP are blocked, so in case of realignment of more than one quota, these will be executed in the same moment.

It is advised not to use the realignment quota during the deceleration ramp.

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0331	0x214B	**Rtouchpro-befunc**	Controlword of the Touch Probe function. B0: 0=Switch off touch probe 1 1=Enable touch probe 1 B1: 0=Trigger first event 1=Continuous B2: 0=Trigger with t.p.1 input 1=Trigger with TOP encoder B3: Reserved B4: 0=Switch off sampling at t.p.1 1=Enable sampling at t.p.1 B5: Not supported B7-6: 00=Touch Rposact 01=Touc+E323h Motor Encoder 10=Touch External Encoder 11=Touch Absolute Encoder B8: 0=Switch off touch probe2 1=Enable touch probe 2 B9: 0=Trigger first event 1=Continuous B10: 0=Trigger with t.p.2 input 1=Trigger with TOP encoder B11: Reserved B12: 0=Switch off sampling at t.p.2 1=Enable sampling at t.p.2 B13: Not supported B15-14: 00=Touch Rposact 01=Touch Motor Encoder 10=Touch External Encoder 11=Touch Absolute Encoder		0	65535	Unsig-ned16	WORD SAVE MAP WRITE
0332	0x214C	**Rtouchpro-bestatus**	Statusword of the Touch Probe function. B0: 0=Touch Probe 1 is switched off 1=Touch Probe 1 is enabled B1: 0=T.P. 1 no value stored 1=Touch Probe 1 value stored B2: Not supported B3: Reserved B4: Reserved B5: Reserved B6: Reserved B7: Shall toggle with every update of T.P.1 value stored B8: 0=Touch Probe 2 is switched off 1=Touch Probe 2 is enabled B9: 0=T.P. 2 no value stored 1=Touch Probe 2 value stored B10: Not supported B11: Reserved B12: Reserved B13: Reserved B14: Reserved B15: Shall toggle with every update of T.P.2 value stored				Unsig-ned16	WORD RO MAP READ
0333-34	0x214D	**Rtouchpro-bepos1pos**	Touch Probe 1 position value at positive edge of t.p.1 touch signal				Signed32	WORD RO MAP READ
0335-36	0x214F	**Rtouchpro-bepos1neg**	Touch Probe 1 position value at negative edge of t.p.1 touch signal				Signed32	WORD RO MAP READ
0337-38	0x2151	**Rtouchpro-bepos2pos**	Touch Probe 2 position value at positive edge of t.p.2 touch signal				Signed32	WORD RO MAP READ
0339-40	0x2153	**Rtouchpro-bepos2neg**	Touch Probe 2 position value at negative edge of t.p.2 touch signal				Signed32	WORD RO MAP READ
0341	0x2155	**Rtouchpro-be1inpdef**	Setting of the digital input to be used for the touch probe 1 function		0	255	Unsig-ned16	WORD RO MAP WRITE

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0342	0x2156	**Rtouchpro-be2inpdef**	Setting of the digital input to be used for the touch probe 2 function		0	255	Unsig-ned16	WORD RO MAP WRITE

Touch probe function

The touch probe function records an axis position at the point in time of an input digital signal. Since the position is usually not recorded directly in the PLC, but via an external hardware latch, it is highly accurate and independent of cycle time. The touch probe function controls this mechanism and determines the externally recorded position.

Register 0331 (CanOpen Object 60B8h) – Touch probe function

Modbus register	CanOpen Index	CanOpen SubIndex	Parameter Name	Data type	Access type	Default value	PDO map-ping
4x0331	60B8h	0	**Rtouchprobefunc** Touch probe function	Unsigned 16	RW	0	Yes

This object indicates the configured function of the touch probe.

Notes: Bit 0 to 7: for touch probe 1
Bit 8 to 15: for touch probe 2
Bit2/10 cannot be changed after 60B8h Bit4/12 was set to 1.

Bit No.	Value	Definition
0	0	Switch off touch probe 1
	1	Enable touch probe 1
1	0	Trigger first event
	1	continuous
2	0	Trigger with touch probe 1 input
	1	Trigger with zero signal of position encoder
3	-	Reserved
4	0	Switch off sampling at touch probe 1
	1	Enable sampling at touch probe 1
5	-	not supported
6, 7	0	User-defined (not used)
8	0	Switch off touch probe 2
	1	Enable touch probe 2
9	0	Trigger first event
	1	continuous
10	0	Trigger with touch probe 2 input
	1	Trigger with zero signal of position encoder
11	0	Reserved
12	0	Switch off sampling at touch probe 2
	1	Enable sampling at touch probe 2
13	0	not supported
14, 15	0	User-defined (not used)

Register 0332 (CanOpen Object 60B9h) - Touch probe status

Modbus register	CanOpen Index	CanOpen SubIndex	Parameter Name	Data type	Access type	Default value	PDO mapping
4x0332	60B9h	0	**Rtouchprobestatus** Touch probe Status	Unsigned 16	RO	0	Yes

This object provides the status of the touch probe.

Value range: Unsigned16

Bit No.	Value	Definition
0	0	Touch probe 1 is switched off
	1	Touch probe 1 is enabled
1	0	Touch probe 1 no value stored
	1	Touch probe 1 value stored
2	0	not supported
3 to 6	0	Reserved
7	0,1	Shall toggle with every update of Touch probe 1 value stored *1
8	0	Touch probe 2 is switched off
	1	Touch probe 2 is enabled
9	0	Touch probe 2 no value stored
	1	Touch probe 2 value stored
10	0	not supported
11 to 14	0	Reserved
15	0,1	Shall toggle with every update of Touch probe 2 value stored *1

Notes: Bit 0 to 7: for touch probe 1
Bit 8 to 15: for touch probe 2

*1) If the continuous latch is enabled (object 60B8 bit 1 = 1, or bit 9 = 1), bit 7 or bit 15 of object 60B9h is toggled with every stored update of the touch probe value.

Register 0333-34 (CanOpen Object 60BAh) - Touch probe position 1 positive value

Modbus register	CanOpen Index	CanOpen SubIndex	Parameter Name	Data type	Access type	Default value	PDO mapping
4x0333-34	60BAh	0	**Rtouchprobepos1pos** Touch probe position 1 positive value	Integer 32	RO	0	Yes

This object provides the position value of the touch probe 1.
The value shall be given in user-defined position units.

Value range: Integer32

Units: Pos units

Register 0335-36 (CanOpen Object 60BBh) - Touch probe position 1 negative value

Modbus register	CanOpen Index	CanOpen SubIndex	Parameter Name	Data type	Access type	Default value	PDO mapping
4x0335-36	60BBh	0	**Rtouchprobepos1neg** Touch probe position 1 negative value	Integer 32	RO	0	Yes

This object provides the position value of the touch probe 1.
The value shall be given in user-defined position units.

Value range: Integer32

Units: Pos units

Register 0337-38 (CanOpen Object 60BCh) - Touch probe position 2 positive value

Modbus register	CanOpen Index	CanOpen SubIndex	Parameter Name	Data type	Access type	Default value	PDO mapping
4x0337-38	60BCh	0	**Rtouchprobepos2pos** Touch probe position 2 positive value	Integer 32	RO	0	Yes

This object provides the position value of the touch probe 2.
The value shall be given in user-defined position units.

Value range: Integer32

Units: Pos units

Register 0339-40 (CanOpen Object 60BDh) - Touch probe position 2 negative value

Modbus register	CanOpen Index	CanOpen SubIndex	Parameter Name	Data type	Access type	Default value	PDO mapping
4x0339-40	60BDh	0	**Rtouchprobepos2neg** Touch probe position 2 negative value	Integer 32	RO	0	Yes

This object provides the position value of the touch probe 2.
The value shall be given in user-defined position units.

Value range: Integer32

Units: Pos units

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0345	0x2159	**Rcaptselinp**	Selection of the digital input to be used for “capture” function.		0	15	Unsig-ned16	WORD SAVE
0346-47	0x215A	**Rcaptvel-max**	Maximum value reachable by the timer in “capture” function. (Time base 20us)				Unsig-ned32	WORD SAVE
0348-49	0x215C	**Rcaptval**	Value of the period registerd by the “capture” function between two edges of the digital input. (Time base 20us)				Unsig-ned32	WORD
0350	0x215E	**Rcaptcnt**	Number of pulses received on the digital input asso-ciated to the “capture” function. It ranges from 0 to 65535. Once the upper limit is reached, it restarts from 0.		0	65535	Unsig-ned16	WORD

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0380	0x217C	**Rcammstpul-se**	Master encoder pulses per revolution (for cams Type1 and Type2)	Enc. Pulses	1	32767	Unsig-ned16	WORD SAVE NO MAP
0381	0x217D	**Rcammst-maxfrq**	Master encoder maximum pulses/sec (for cam Type2)	Enc. Pulses /s	1	32767	Unsig-ned16	WORD SAVE NO MAP
0382	0x217E	**Rcamstrtpha-se**	Number of the task with the first phase to be executed (for cams Type1 and Type2)		0	63	Unsig-ned16	WORD SAVE MAP READ
0383	0x217F	**Rcammstpo-sact**	Master encoder actual position(for cams Type1 and Type2)	Enc. Pulses			Unsig-ned16	WORD MAP READ RO
0384	0x2180	**Rcamphase-act**	Actual phase (task) in progress (for cams Type1 and Type2)				Unsig-ned16	WORD NO MAP RO
0385	0x2181	**Rcamflgwr**	Cam writing flag Bit 0: Cam reset Bit 1: Only one revolution of the cam. then stops at the maximum value. Bit 2: Bit 3: Bit 4: Bit 5: Bit 6: Bit 7: Bit 8: Bit 9: Bit 10: Bit 11: Bit 12: Bit 13: Bit 14: Bit 15: (for cams Type1 and Type2)				Unsig-ned16	WORD NO MAP
0386	0x2182	**Rcamflgrd**	Cam reading flag Bit 0: Cam reset Bit 1: Cam in movement Bit 2: Camma in acceleration Bit 3: Camma at constant speed Bit 4: Camma in deceleration Bit 5: Bit 6: Bit 7: Bit 8: Bit 9: Bit 10: Bit 11: Bit 12: Bit 13: Bit 14: Bit 15: (for cams Type1 and Type2)				Unsig-ned16	WORD MAP READ RO
0387	0x2183	**Rcamkp**	Kp PI increments management in cam mode (for cams Type1, Type2, STSP and Winding)		0	32767	Unsig-ned16	WORD SAVE NO MAP

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0456	0x21C8	**Rcamst-spmstmm-pulse**	Millimeters/pulse master shift in start/stop cam (mm*1000/pulse)	mm/ Imp.	0	65535	Unsig-ned16	WORD SAVE
0457	0x21C9	**Rcamst-spslvmm-step**	Millimeters/pulse slave shift in start/stop cam (mm*1000/pulse)	mm/ Imp.	0	65535	Unsig-ned16	WORD SAVE
0458	0x21CA	**Rcamst-spcor**	Master/slave centesimal correction in start/stop cam (-100/+100)	%	-1000	1000	Signed16	WORD SAVE
0459	0x21CB	**Rcamst-spaccspace**	Slave acceleration space on master space in start/ stop cam	mm*10	0	65535	Unsig-ned16	WORD SAVE
0460	0x21CC	**Rcamstspc-trl**	Start/stop cam control B0: 0=Stop/1=Run B1: 1=Disable start delay B2: 1=Disable stop delay B3: 1=Simulated encoder B4: B5: B6: B7:				Unsig-ned16	WORD MAP WRITE
0461	0x21CD	**Rcamst-spstatus**	Start/stop cam status B1: Cam enabled B2: Cam in RUN B3: Cam in acceleration B4: Cam at constant acceleration B5: Cam in decelaration B5: B6: B7:				Unsig-ned16	WORD RO MAP READ
0462-63	0x21CE	**Rcamst-splowquo-teslv**	Minimum slave quota in start/stop cam	Step	0	0x7FFFFFFF	Unsig-ned32	H-LWORD SAVE
0464-65	0x21D0	**Rcamstsphi-ghquoteslv**	Maximum slave quota in start/stop cam	Step	0	0x7FFFFFFF	Unsig-ned32	H-LWORD SAVE
0466-67	0x21D2	**Rcamst-splowquo-temst**	Minimum master quota in start/stop cam	Enc. Pulses	0	0x7FFFFFFF	Unsig-ned32	H-LWORD SAVE
0468-69	0x21D4	**Rcamstsphi-ghquotemst**	Maximum master quota in start/stop cam	Enc. Pulses	0	0x7FFFFFFF	Unsig-ned32	H-LWORD SAVE
0470	0x21D6	**Rcammode-select**	Cam mode selection 0=Type1 cam (Blk) 1=Type2 cam (Bert)		0	1	Unsig-ned16	WORD SAVE
0496	0x21F0	**Rcamst-spsimvel**	Speed in mm/s of the simulated encoder in start/stop cam function	mm/s			Unsig-ned16	WORD SAVE
0497	0x21F1	**Rcamst-spdecspace**	Slave deceleration space on master space in start/ stop cam	mm*10			Unsig-ned16	WORD SAVE
0498	0x21F2	**Rcamst-spstartdly**	Mater space before the slave start in start/stop cam mode	mm*10			Unsig-ned16	WORD SAVE
0499	0x21F3	**Rcamst-spstopdly**	Slave space from the stop signal to the start of the deceleration ramp	mm*10			Unsig-ned16	WORD SAVE

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0471	0x21D7	**Rwindspoolimpturn**	Spool encoder pulses per revolution in winding mode	Enc. Pulses	1	65535	Unsigned16	WORD SAVE
0472	0x21D8	**Rwindyarnguidemmstep**	Wire guide shift in mm for stepper mode in winding mode	mm* 1000	0	65535	Unsigned16	WORD SAVE
0473	0x21D9	**Rwindaccspoolturn**	Wire guide acceleration turns to reach the requested speed in winding mode	1/10 turn	0	65535	Unsigned16	WORD SAVE
0474	0x21DA	**Rwindyarnshift**	Wire guide shift for spool revolution in winding mode	mm* 100	0	65535	Unsigned16	WORD SAVE
0475	0x21DB	**Rwindtyingshift**	Wire guide shift for spool revolution during the binding phase in winding mode	mm* 100	0	65535	Unsigned16	WORD SAVE
0476	0x21DC	**Rwindctrl**	Control word winding mode B0: 1=Enable winding cam B1: 1=Enable stop sequence B2: 1=Enable binding sequence B3: 1=Disable encoder reading B4: 1=Winding cam error reset B5: Start direction of the deposit B6: B7: B8: B9: B10: B11: B12: B13: B14: B15:				Unsigned16	WORD MAP WRITE
0477	0x21DD	**Rwindstatus**	Status word winding mode B0: 1=Winding cam enabled B1: 1=Stop sequence request enabled B2: 1=Binding sequence request enabled B3: 1=Encoder disabled B4: Cam stepper motor direction flag B5: 1=End of cam sequence B6: 1=Cam error B7: 1=Binding sequence executed B8: B9: B10: B11: B12: B13: Cam in acceleration B14: Cam in constant speed B15: Cam in deceleration				Unsigned16	WORD RO MAP READ
0478-79	0X21DE	**Rwindquoteleft**	Quota toward wire guide machine in winding mode	mm*10	0	0x7FFFFFFF	Unsigned32	H-LWORD
0480-81	0x21E0	**Rwindquoteright**	Quota toward wire guide user in winding mode	mm*10	0	0x7FFFFFFF	Unsigned32	H-LWORD
0482-83	0x21E2	**Rwindquotestop**	Wire guide stop quota in winding mode	mm*10	0	0x7FFFFFFF	Unsigned32	H-LWORD
0484-85	0x21E4	**Rwindquotetyingleft**	Quota toward wire guide machine for binding in winding mode	mm*10	0	0x7FFFFFFF	Unsigned32	H-LWORD
0486-87	0x21E6	**Rwindquotetyingright**	Quota toward wire guide user for binding in winding mode	mm*10	0	0x7FFFFFFF	Unsigned32	H-LWORD
0488-89	0x21E8	**Rwindyarnguidemmturnmotor**	Wire guide shift in mm per stepper motor revolution	mm* 100	0	0x7FFFFFFF	Unsigned32	H-LWORD
0490	0x21EA	**Rwindrevcnt**	Reversals counter in winding mode				Unsigned16	WORD MAP READ
0491-92	0X21EB	**Rwindrevpause**	Pulses of master encoder with stepper motor in pause during the reversal of the direction of motion	Enc. Pulses	0	0x7FFFFFFF	Unsigned32	H-LWORD

FUNCTIONAL TASK WITH DIGITAL OUTPUTS FUNCTION

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0530	0x2212	**Rtskfmstfw1**	Experimental function for movement from task table + outputs management				Unsig-ned16	WORD
0531	0x2213	**Rtskfmstfw2**	Experimental function for movement from task table + outputs management				Unsig-ned16	WORD
0532	0x2214	**Rtskfmst-macvelact**	Experimental function for movement from task table + outputs management				Unsig-ned16	WORD
0533	0x2215	**Rtskfmst-macvelmax**	Experimental function for movement from task table + outputs management				Unsig-ned16	WORD
0534	0x2216	**Rtskfmst-spcphdis**	Experimental function for movement from task table + outputs management				Unsig-ned16	WORD
0535	0x2217	**Rtskfmstphinp**	Experimental function for movement from task table + outputs management				Unsig-ned16	WORD

EXTERNAL BRAKE

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0098	0x2062	**Rbrakedlyo-pen**	Opening delay of external brake, commanded by digital output	ms	0	65535	Unsig-ned16	WORD SAVE
0344	0x2158	**Rbra-kedlyclose**	Closing delay of external brake, commanded by digital output	ms	0	65535	Unsig-ned16	WORD SAVE

PID FUNCTION

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0500	0x21F4	**Rusrpidkp**	User PID proportional gain				Unsig-ned16	WORD SAVE
0501	0x21F5	**Rusrpidki**	User PID integral gain				Unsig-ned16	WORD SAVE
0502	0x21F6	**Rusrpidkd**	User PID derivative gain				Unsig-ned16	WORD SAVE
0503	0x21F7	**Rusrpidkt**	User PID integration time in mS	ms			Unsig-ned16	WORD SAVE
0504	0x21F8	**Rusrpidflag**	User PID control flag B0: 0=Stop PID / 1=Start PID B1: 1=Reset PID B2: B3: B4: B5:				Unsig-ned16	WORD
0505-06	0x21F9	**Rusrpidma-xout**	User PID maximum output value				Signed32	WORD SAVE
0507-08	0x21FB	**Rusrpidmi-nout**	User PID minimum output value				Signed32	WORD SAVE
0509-10	0x21FD	**Rusr-pidsetpoint**	User PID Setpoint				Signed32	WORD SAVE MAP WRITE
0511-12	0x21FF	**Rusrpidpro-cessvalue**	User PID process value				Signed32	WORD MAP READ
0513-14	0x2201	**Rusrpidout**	User PID output value				Signed32	WORD MAP READ

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0315	0x213B	**Rfgapplyto**	Data to which the internal function generator signal must be applied: 0: None 1: Current Phase A 2: Current Phase B 3: Speed 4: Position 5: Profile		0	5	Unsig-ned16	WORD
0316-17	0x213C	**Rfgvalmin**	Function generator minimum value	Bit	0x80000000	0x7FFFFFFF	Signed32	LS – MSWORD NO MAP RO
0318-19	0x213E	**Rfgvalmax**	Function generator maximum value	Bit	0x80000000	0x7FFFFFFF	Signed32	LS – MSWORD NO MAP RO
0320	0x2140	**Rfgfreq**	Frequency of the generated signal	Hz	1	10000	Unsig-ned16	WORD NO MAP
0321	0x2141	**Rfgperiod**	Period of the generated signal	ms	1	1000	Unsig-ned16	WORD NO MAP
0322	0x2142	**Rfgcmd**	Function generator command 0 = FG stop 1 = FG start		0	1	Unsig-ned16	WORD SAVE NO MAP
0323	0x2143	**Rfgmode**	Type of signal generated 0 = sinusoidal 1= square wave 2 = pulse		0	2	Unsig-ned16	WORD
0324-25	0x2144	**Rfgout32**	Functions generator output	Bit			Signed32	LS – MSWORD NO MAP RO
0326	0x2146	**Rfgout16**	Functions generator output	Bit			Signed16	WORD

Indirizzo Modbus	Nome Registro	Descrizione	U.M.	Valore Minimo	Valore Massimo	Tipo	
7000		Oscilloscope data 0 channel 1				Unsigned16	WORD
7001-7498		Oscilloscope data 1-498 channel 1				Unsigned16	WORD
7499		Oscilloscope data 499 channel 1				Unsigned16	WORD
7500		Oscilloscope data 0 channel 2				Unsigned16	WORD
7501-7998		Oscilloscope data 1-498 channel 2				Unsigned16	WORD
7999		Oscilloscope data 499 channel 2				Unsigned16	WORD
8000		Channel 1 value divider Nuber of right Shifts				Unsigned16	WORD
8001		Channel 2 value divider Nuber of right Shifts				Unsigned16	WORD
8002		Trigger value divider Nuber of right Shifts				Unsigned16	WORD
8003		Ch1 data Modbus address				Unsigned16	WORD
8004		Ch2 data Modbus address				Unsigned16	WORD
8005		Trigger data Modbus address				Unsigned16	WORD
8006		Value for trigger				Unsigned16	WORD
8007		Trigger type Bit 0: 0=Single / 1=Continuous Bit 1: 1=Arm trigger in single mode Bit 2: 1=Immediate start Bit 3: 1=Start if trigger > Bit 4: 1=Start if trigger < Bit 5: 1=Acquisition stop Bit 6: Bit 7:				Unsigned16	WORD
8008		Pretrigger points				Unsigned16	WORD
8009		Times base Number of 250uS temporal quantum				Unsigned16	WORD
8010		Oscilloscope commands B0: Enable oscilloscope B1: B2: B3: B4: B5: B6: B7:				Unsigned16	WORD
8011		Oscilloscope Status B0: Oscilloscope enabled B1: Copy of the bit “Arm trigger in single mode” B2: Trigger armed B3: Acquisition in progress B4: Data buffer ready B5: B6: B7:				Unsigned16	WORD

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
0204	0x20CC	**Rstsprg**	MIL program status register Bit 0: RUN (Program in RUN if 1) Bit 1: PRG_OK (Program not valid if 0) Bit 2: Bit 3: Bit 4: Bit 5: Bit 6: Bit 7: Bit 8:				Unsig-ned16	WORD NO MAP RO
0205	0x20CD	**Rprgrunmo-de** **Run mode User Pro-grammare** **0 = Stop** **1 = Single step** **2 = Run**	Run mode User Program 0 = Stop 1 = Single step 2 = Run		0	2	Unsig-ned16	WORD NO MAP
0206	0x20CE	**Rprgstope-vent**	Stop event status register (value reading) 0: No stop request 1: Stop from manual command 2: Not valid program 3: Program Counter over the end of the program 4: Program Counter over the end of the program memory 5: Not valid instruction 6: Interrupt event manager overflow stack pointer 7: END Instruction 8: Excessive number of subroutines 9: Excessive number of returns from subroutine(corrupted stack) 10: Excessive number of interrupt ONH events (ONH+ONL<10) 11: Excessive number of interrupt ONL events (ONH+ONL<10) 12: Excessive number of returns from interrupt events (corrupted stack)				Holding Register (16bit)	WORD RO
0207	0x20CF	**Rprgcnt**	Program counter User program N.B. it can be written only if user program is in stop.		0	4998	Unsig-ned16	WORD NO MAP
0208	0x20D0	**Rprgccflag**	Condition code flag B0: Carry B1: Overflow B2: Zero B3: Negative B4: Interrupt (1 ONH and ONL instruction mask)				Unsig-ned16	WORD NO MAP
2000		**Rprgusrpsw**	Password entered for the enabling of the access to the user program memory location. It is not stored in the eeprom. If not equal to the value in the eeprom, the user program memory location is not accessible neither in read nor in write.				Unsig-ned16	WORD
2001		**Rprgcmd**	Coomands 0x01: Save program in EEPROM 0x02: Load program from EEPROM 0x81: Delete password (delete the whole program) 0x82: Set new password 0x83: Delete program in RAM				Unsig-ned16	WORD
2002		**Rprgcmdsts**	Command status Last command output status				Unsig-ned16	WORD
2003		**Rprgnew-psw**	New Password				Unsig-ned16	WORD
2004		**Rprgpswact**	Password active 0: Program unlocked 1: Program protected by password				Unsig-ned16	WORD

Modbus Profibus address	CAN EtherCAT Profinet address	Register name	Description	U.M.	Minimum value	Maximum value	Type	Note
2005		**Rprgbpena-ble**	Enable the software breakpoint management The value is expressed in byte				Unsig-ned16	WORD
2006		**Rprgbpnum**	Number of the breakpoint to be managed in read/write with Rprgbpval 0= Number of instructions to be executed in single step mode 1= Break point number 1 address 2= Break point number 2 address 3= Break point number 3 address 4= Break point number 4 address				Unsig-ned16	WORD
2007		**Rprgbpval**	Value read or to be written in the pointed braskpoint.				Unsig-ned16	WORD
2008		**Rprglen**	Length of the user program				Unsig-ned16	WORD
2009		**Rprgprgcks**	User program checksum				Unsig-ned16	WORD
2010		**Rprgusrprg**	User program location 0				Unsig-ned16	WORD
2011-5008			User program locations nnn				Unsig-ned16	WORD
6999			User program location 4998				Unsig-ned16	WORD

VARIABLES

Variables are 32 bits memory locations, at the disposal of the programmer.
VectorStep drives are provided with a maximum of 128 varialbes, nembered from 1 to 128.
The transfer of the data is done with the instruction MOVE. Mathematical, comparison, increment and decrement instructions can act on a variable..

V1 : Variable 1 (Signed 32 bit)
.
.
V128 : Variable 128 (Signed 32 bit)

MODBUS protocol only manages 16 bit values. To perform a reading or a writing of the variables, you must transfer 2 MODBUS registers at a time. The calculation of the location of a variable can be made using the following formula:

Number of variable + 4 1001

Example:
To read variable V7, you must access to the following MODBUS locations:
(7 * 2) + 4 1000 = 4 1014 (least significant word of the variable, bit 0-15)
(7 * 2) + 4 1000 + 1 = 4 1015 (most significant word of the variable, bit 16-31)

N.B.: Verify if the Modbus addresses of the master start from 0 or 1. AEC Modbus addresses start from 0, in case of using a master in which they start from 1 (e.g. Siemens or Weintek) it is necessary to add 1 to the AEC Modbus address.
Example: V1= 1002+1= 1003

AREA MODBUS	ID	Name	Format	Access	Description
4	1002-3	V1	Unsigned32	RW	V1 general purpose program variable
4	1004-5	V2	Unsigned32	RW	V2 general purpose program variable
4	1006-7	V3	Unsigned32	RW	V3 general purpose program variable
4	1008-9	V4	Unsigned32	RW	V4 general purpose program variable
4	1010-11	V5	Unsigned32	RW	V5 general purpose program variable
4	1012-13	V6	Unsigned32	RW	V6 general purpose program variable
4	1014-15	V7	Unsigned32	RW	V7 general purpose program variable
4	1016-17	V8	Unsigned32	RW	V8 general purpose program variable
4	1018-19	V9	Unsigned32	RW	V9 general purpose program variable
4	1020-21	V10	Unsigned32	RW	V10 general purpose program variable
4	1022-23	V11	Unsigned32	RW	V11 general purpose program variable
4	1024-25	V12	Unsigned32	RW	V12 general purpose program variable
4	1026-27	V13	Unsigned32	RW	V13 general purpose program variable
4	1028-29	V14	Unsigned32	RW	V14 general purpose program variable
4	1030-31	V15	Unsigned32	RW	V15 general purpose program variable
4	1032-33	V16	Unsigned32	RW	V16 general purpose program variable
4	1034-35	V17	Unsigned32	RW	V17 general purpose program variable
4	1036-37	V18	Unsigned32	RW	V18 general purpose program variable
4	1038-39	V19	Unsigned32	RW	V19 general purpose program variable
4	1040-41	V20	Unsigned32	RW	V20 general purpose program variable
4	1042-43	V21	Unsigned32	RW	V21 general purpose program variable
4	1044-45	V22	Unsigned32	RW	V22 general purpose program variable
4	1046-47	V23	Unsigned32	RW	V23 general purpose program variable
4	1048-49	V24	Unsigned32	RW	V24 general purpose program variable
4	1050-51	V25	Unsigned32	RW	V25 general purpose program variable
4	1052-53	V26	Unsigned32	RW	V26 general purpose program variable
4	1054-55	V27	Unsigned32	RW	V27 general purpose program variable

AREA MODBUS	ID	Nome	Formato	Accesso	Descrizione
4	1056-57	V28	Unsigned32	RW	V28 general purpose program variable
4	1058-59	V29	Unsigned32	RW	V29 general purpose program variable
4	1060-61	V30	Unsigned32	RW	V30 general purpose program variable
4	1062-63	V31	Unsigned32	RW	V31 general purpose program variable
4	1064-65	V32	Unsigned32	RW	V32 general purpose program variable
4	1066-67	V33	Unsigned32	RW	V33 general purpose program variable
4	1068-69	V34	Unsigned32	RW	V34 general purpose program variable
4	1070-71	V35	Unsigned32	RW	V35 general purpose program variable
4	1072-73	V36	Unsigned32	RW	V36 general purpose program variable
4	1074-75	V37	Unsigned32	RW	V37 general purpose program variable
4	1076-77	V38	Unsigned32	RW	V38 general purpose program variable
4	1078-79	V39	Unsigned32	RW	V39 general purpose program variable
4	1080-81	V40	Unsigned32	RW	V40 general purpose program variable
4	1082-83	V41	Unsigned32	RW	V41 general purpose program variable
4	1084-85	V42	Unsigned32	RW	V42 general purpose program variable
4	1086-87	V43	Unsigned32	RW	V43 general purpose program variable
4	1088-89	V44	Unsigned32	RW	V44 general purpose program variable
4	1090-91	V45	Unsigned32	RW	V45 general purpose program variable
4	1092-93	V46	Unsigned32	RW	V46 general purpose program variable
4	1094-95	V47	Unsigned32	RW	V47 general purpose program variable
4	1096-97	V48	Unsigned32	RW	V48 general purpose program variable
4	1098-99	V49	Unsigned32	RW	V49 general purpose program variable
4	1100-101	V50	Unsigned32	RW	V50 general purpose program variable
4	1102-103	V51	Unsigned32	RW	V51 general purpose program variable
4	1104-105	V52	Unsigned32	RW	V52 general purpose program variable
4	1106-107	V53	Unsigned32	RW	V53 general purpose program variable
4	1108-109	V54	Unsigned32	RW	V54 general purpose program variable
4	1110-111	V55	Unsigned32	RW	V55 general purpose program variable
4	1112-113	V56	Unsigned32	RW	V56 general purpose program variable
4	1114-115	V57	Unsigned32	RW	V57 general purpose program variable
4	1116-117	V58	Unsigned32	RW	V58 general purpose program variable
4	1118-119	V59	Unsigned32	RW	V59 general purpose program variable
4	1120-121	V60	Unsigned32	RW	V60 general purpose program variable
4	1122-123	V61	Unsigned32	RW	V61 general purpose program variable
4	1124-125	V62	Unsigned32	RW	V62 general purpose program variable
4	1126-127	V63	Unsigned32	RW	V63 general purpose program variable
4	1128-129	V64	Unsigned32	RW	V64 general purpose program variable
4	1130-131	V65	Unsigned32	RW	V65 general purpose program variable
4	1132-133	V66	Unsigned32	RW	V66 general purpose program variable
4	1134-135	V67	Unsigned32	RW	V67 general purpose program variable
4	1136-137	V68	Unsigned32	RW	V68 general purpose program variable
4	1138-139	V69	Unsigned32	RW	V69 general purpose program variable
4	1140-141	V70	Unsigned32	RW	V70 general purpose program variable
4	1142-143	V71	Unsigned32	RW	V71 general purpose program variable
4	1144-145	V72	Unsigned32	RW	V72 general purpose program variable
4	1146-147	V73	Unsigned32	RW	V73 general purpose program variable
4	1148-149	V74	Unsigned32	RW	V74 general purpose program variable
4	1150-151	V75	Unsigned32	RW	V75 general purpose program variable
4	1152-153	V76	Unsigned32	RW	V76 general purpose program variable
4	1154-155	V77	Unsigned32	RW	V77 general purpose program variable
4	1156-157	V78	Unsigned32	RW	V78 general purpose program variable
4	1158-159	V79	Unsigned32	RW	V79 general purpose program variable

AREA MODBUS	ID	Nome	Formato	Accesso	Descrizione
4	1160-161	V80	Unsigned32	RW	V80 general purpose program variable
4	1162-163	V81	Unsigned32	RW	V81 general purpose program variable
4	1164-165	V82	Unsigned32	RW	V82 general purpose program variable
4	1166-167	V83	Unsigned32	RW	V83 general purpose program variable
4	1168-169	V84	Unsigned32	RW	V84 general purpose program variable
4	1170-171	V85	Unsigned32	RW	V85 general purpose program variable
4	1172-173	V86	Unsigned32	RW	V86 general purpose program variable
4	1174-175	V87	Unsigned32	RW	V87 general purpose program variable
4	1176-177	V88	Unsigned32	RW	V88 general purpose program variable
4	1178-179	V89	Unsigned32	RW	V89 general purpose program variable
4	1180-181	V90	Unsigned32	RW	V90 general purpose program variable
4	1182-183	V91	Unsigned32	RW	V91 general purpose program variable
4	1184-185	V92	Unsigned32	RW	V92 general purpose program variable
4	1186-187	V93	Unsigned32	RW	V93 general purpose program variable
4	1188-189	V94	Unsigned32	RW	V94 general purpose program variable
4	1190-191	V95	Unsigned32	RW	V95 general purpose program variable
4	1192-193	V96	Unsigned32	RW	V96 general purpose program variable
4	1194-195	V97	Unsigned32	RW	V97 general purpose program variable
4	1196-197	V98	Unsigned32	RW	V98 general purpose program variable
4	1198-199	V99	Unsigned32	RW	V99 general purpose program variable
4	1200-201	V100	Unsigned32	RW	V100 general purpose program variable
4	1202-203	V101	Unsigned32	RW	V101 general purpose program variable
4	1204-205	V102	Unsigned32	RW	V102 general purpose program variable
4	1206-207	V103	Unsigned32	RW	V103 general purpose program variable
4	1208-209	V104	Unsigned32	RW	V104 general purpose program variable
4	1210-211	V105	Unsigned32	RW	V105 general purpose program variable
4	1212-213	V106	Unsigned32	RW	V106 general purpose program variable
4	1214-215	V107	Unsigned32	RW	V107 general purpose program variable
4	1216-217	V108	Unsigned32	RW	V108 general purpose program variable
4	1218-219	V109	Unsigned32	RW	V109 general purpose program variable
4	1220-221	V110	Unsigned32	RW	V110 general purpose program variable
4	1222-223	V111	Unsigned32	RW	V111 general purpose program variable
4	1224-225	V112	Unsigned32	RW	V112 general purpose program variable
4	1226-227	V113	Unsigned32	RW	V113 general purpose program variable
4	1228-229	V114	Unsigned32	RW	V114 general purpose program variable
4	1230-231	V115	Unsigned32	RW	V115 general purpose program variable
4	1232-233	V116	Unsigned32	RW	V116 general purpose program variable
4	1234-235	V117	Unsigned32	RW	V117 general purpose program variable
4	1236-237	V118	Unsigned32	RW	V118 general purpose program variable
4	1238-239	V119	Unsigned32	RW	V119 general purpose program variable
4	1240-241	V120	Unsigned32	RW	V120 general purpose program variable
4	1242-243	V121	Unsigned32	RW	V121 general purpose program variable
4	1244-245	V122	Unsigned32	RW	V122 general purpose program variable
4	1246-247	V123	Unsigned32	RW	V123 general purpose program variable
4	1248-249	V124	Unsigned32	RW	V124 general purpose program variable
4	1250-251	V125	Unsigned32	RW	V125 general purpose program variable
4	1252-253	V126	Unsigned32	RW	V126 general purpose program variable
4	1254-255	V127	Unsigned32	RW	V127 general purpose program variable
4	1256-257	V128	Unsigned32	RW	V128 general purpose program variable

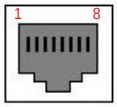
CONNECTIONS

EMC IMMUNITY

To prevent them from being created EMI disturbances caused by cables or devices contained in the same power panel, the drive must be properly connected to protective earth as described in the in the manual.
AEC does not guarantee proper EMC behavior unless thes PE requirements are fulfilled

The shield of the RJ45 connector is not connected directly to PE. As all nodes in a Profinet network have to share earth connection, the Profinet cable shield has to be connected to the earth ai each node in the network.
For further information, see “PROFINET Installation Guideline for Cabling and Assembly, no. 8072” available to download at www.profinet.com

CONNECTOR PINOUT



Pin no	Description
1	TD+
2	TD-
3	RD+
4, 5, 7, 8	Connected to ground over serial RC circuit
6	RD-
Housing	Cable Shield

ALERTS

LED STATUS

In order to simplify the diagnostic of the drive, there are 7 bicolor LEDs that permit to obtain information on the current status of the drive.

LED	*Led name*	Color	Status	Description
PWR	*PWR*	GREEN	ON	Drive is powered correctly
			OFF	No voltage or low voltage
		ORANGE	X	Reserved
		RED	X	Reserved
CUR	*CUR*	X	OFF	Output current to the motor disabled (no current)
		GREEN	ON	Nominal current
		ORANGE	ON	Reduced current
		RED	ON	Current BOOST
STS	*STS*	GREEN	ON	Drive OK
		ORANGE	ON	Drive in thermal protection
		RED	ON	Drive in alarm, fail or BOOT
			BLINK	No power supply
SER	*SER*	GREEN	X	Reserved
		ORANGE	ON	Serial port in transmission
		RED	X	Reserved
FLD 1	*DATA EXCH.*	GREEN	ON	Data Exchange
			BLINK	No Data Exchange
FLD 2	*BUS FAIL.*	RED	ON	Non initialized stack or Fatal Error
FLD 3	*SF*	YELLOW	ON	Service mode

FAULT ALERTS

In case an alarm occurs, the drive disables the power output and the motor will stop in non-controlled way.
The presence of errors is indicated from the bit 3 of the Statusword 1 (ZSW1.3), and from the STS LED.
The type of error that occurred is stored in a 8 locations buffer (P947), and the maximum number of errors occurred is stored in the parameter P952.
In case that number of errors is greater then 8, the last occurred error will overwrite the eighth position of the buffer.
How to delete the errors:

1. eliminate the cause of the error;
2. send the error acknoledge by changing to 1 the bit 7 of the Controlword 1 (STW1.7 Toggle);
3. repeat the previous operation until there are errors;
4. at the acknoledge of the last active error, the drive is in the status *Switching On Inhibited*;
5. change to 0 and then newly change to 1 the bit 0 of the Controlword 1 (STW1.0 Toggle), in order to return back to the status *Ready For Switching On;*
6. repeat the previous operation with the bit 1 of the Controlword 1 (STW1.1 Toggle), in order to return back to the status *Switched On;*
7. repeat the previous operation with the bit 2 of the Controlword 1 (STW1.2 Toggle), in order to return back to the status *Operation;*