Document revision

	T2S Version	Author	Description
22/06/2007	---	NMA	First draft
18/10/2007	Ver 2.0	NMA	- Added password for writing holding registers. - Addition and deprecation of items in miscellaneous information table. - Modification of back plane connector (DB9 → RJ45) - Addition of “How to and examples” section - Displacement of items from bit field “system actions” to bit field “configuration control”
08/02/2008	Ver 2.1	NMA	- Added “Extended Module table” in Input registers elements - Turn “Not used” field into “Reserved” field in “System table action” of the Holding registers element - Added a new flag in “configuration control” bitfield. - Added remark and note regarding “modules action table”
01/03/2010	Ver 2.6	JEA	- Added field 13 and 14 in “Miscellaneous information” table. - Added tasks states in “System Actions” table
20/01/2011	Ver 4.0	JEA	
18/02/2014	Ver 4.1	GMA	- Correction of extended module table index.
1/04/2014	Ver 4.2	PTH	- Update of configurable parameters

Introduction

1. Scope of the document :

The aim of this paper is describe the information that can be retrieved from T2S controler and the action that can be taken using the ModBus protocol (developed by Modicon).

First section describes hardware requirements and configuration capabilities.

Second section is the exhaustive listing of all variables that can be retrieved and written using the ModBus protocol.

Section three is composed of examples and hints to properly operate with T2S controller.

For more information related to the modbus protocol itself, please refer to http://www.modbus.org/ .

SECTION 1: Hardware requirements

1. Cabling :

ModBus is available on the on RJ45 connector located on the back plane of the rack containing the T2S controller. The pinout of this connector is the following :

Name	Description	Pin Number
CANH	CANH pin for CAN bus	1
CANL	CANL pin for CAN bus	2
GND_IAX	Digital communication ground	3
GND_IAX	Digital communication ground	4
12V_IAX	Unregulated +12V	5
COM_TX_A	RS232 Transmit pin (Option RS485: A [+])	6
GND_IAX	Digital communication ground	7
COM_RX_/A	RS232 Receive pin (Option RS485: /A [-])	8

Remark:
Since the 1st January 2011, T2S can use RS485 hardware for serial communication. Additional information in T2S User Manual…

2. Baud rate, parity and mode :

Only RTU mode is supported. Default parameters are 19200 bauds, even parity, one stop bit, and slave address 1. Those parameters can be set in the configuration file. Configurable values are summarized in the table below. Please note that only slave address was configurable in T2S software versions 3.40 and older.

Item	Value
Slave address	From 1 to 247
Baud Rate	9600 or 19200
Parity	Even, odd, none
Stop bits	One, two
Mode	RTU
Electrical interface	RS232 (Option : RS485)

SECTION 2: Database description

1. Typographic convention:

In this document the following naming convention will be used to represent the type of a variable:

The first letter will indicate if the variable is signed (S) or unsigned (U). Then the following digit(s) will indicate the number of bits needed to store the variable.

Thus:

- U8 will represent an unsigned variable stored in a 8-bit wide memory
- U16 will represent an unsigned variable stored in a 16-bit wide memory
- U32 will represent an unsigned variable stored in a 32-bit wide memory

And :

- S8 will represent a signed variable stored in a 8-bit wide memory
- S16 will represent a signed variable stored in a 16-bit wide memory
- S32 will represent a signed variable stored in a 32-bit wide memory

What is more, ModBus register base type is a 16-bit wide variable. This means it is possible to store two 8-bit variables in a register. These two variables will be accessed using the same index in the structure. Thus in order to know if the variable is stored in the upper byte or in the lower byte, letter **H(high)** or **L(low)** is added to the index.

As specified by the protocol, variables longer than 8-bit are always represented in big-Endian format (MSB first).

2. Data types:

ModBus protocol defines four types of variables class described in the table below

Name	Type	Access	Supported by T2S
Discrete input	1-bit wide	Read-only	No
Coil	1-bit wide	Read-write	No
Input register	16-bit wide	Read-only	Yes
Holding register	16-bit wide	Read-write	Yes

Data address mapping and signification is described below.

3. Supported function:

Accordingly to ModBus specification, supported functions by the T2S controler are the following:

- Read holding registers (0x03)
- Read Input registers (0x04)
- Write single register (0x06)
- Write multiple registers (0x10)

INPUT REGISTERS ELEMENTS (Read-Only 16-bit wide)

Module table (0x0000)

The table described below represents the information that can be retrieved regarding a particular module. Maximum amount of module is set to 32. Each of them is identified by an address ranging from 1 to 32.

BASE ADDRESS: 0(0x0000) + 31*(Module address – 1).

Index	Name	Description	Type
0H	eStatusACOut	AC output status number (see A1)	U8
0L	eStatusACIn	AC input status number (see A1)	U8
1H	eStatusDCIn	DC Input status number (see A1)	U8
1L	bAddress	Configured address	U8
2H	bLoadPosition	Position of the load regarding input power sources (0:AC, 100:DC, 50:mixed, 101:unknown)	U8
2L	bLoadRatioW	Loading ratio regarding power in watts (%)	U8
3H	bLoadRatioVA	Loading ratio regarding power in VA (%)	U8
3L	bPhaseNumber	Number of the phase module is belonging to	U8
4	wVout	Output voltage value (0.1V)	U16
5	wIout	Output current value (0.1A)	U16
6	wPoutW	Output power value (W)	U16
7	wPoutVA	Output power value (VA)	U16
8	wVinAC	AC input voltage value (0.1V)	U16
9	wIinAC	AC input current value (0.1A)	U16
10	wPinACW	AC input power value (W)	U16
11	wPinACVA	AC input power value (VA)	U16
12	wACInFreq	AC input frequency value (0.1Hz)	U16
13	wVinDC	DC input voltage value (0.1V)	U16
14	wIinDC	DC input current value (0.1A)	U16

15	wPinDC	DC input power value (W)	U16
16	wTemperature	Temperature value (K)	U16
17	wSoftVersion	Software version number	U16
18	lSerialNumber	Serial number	U32
20	lBusErrorCnt	Counter that is incremented every time a bad checksum is recorded from that module	U32
22H	bStatusMod	Event number of the status related to the output stage and the module internal status	U8
22L	bStatusAC	Event number of the status related to the AC input stage	U8
23H	bStatusDC	Event number of the status related to the DC input stage	U8
23L	bPresent	Flag (true or false) that indicates if module is seen by T2S or not	U8
24H	bGroupAC	AC input group number module is belonging to	U8
24L	bGroupDC	DC input group number module is belonging to	U8
25H	bRestrained	Flag (true or false)that indicates if module cannot cope with more than five other module or not	U8
25L	bNoEPC	Flag (true or false) that indicates if module has an AC input (EPC) or not	U8
26	wPoutNominalW	Nominal output power (W)	U16
27	wPoutNominalVA	Nominal output power (VA)	U16
28	wVinNominalAC	Nominal AC input voltage (0.1V)	U16
29	wVinNominalDC	Nominal DC input voltage (0.1V)	U16
30	wVinNominalFreqAC	Nominal AC frequency (0.1Hz)	U16

Extended Module table (0x03E0)

The table described below represents the extended information that can be retrieved regarding a particular module. Maximum amount of module is set to 32. Each of them is identified by an address ranging from 1 to 32.
Note: the use of extended module table is currently on hold.

BASE ADDRESS: 992(0x03E0) + 19*(Module address – 1).

Index	Name	Description	Type
0	lReserved	Reserved. Always read as 0	U32
2	Not used	Not used. Always read as 0	15*U8

Phase table (0x0640)

The table described below represents the information that can be retrieved regarding a particular phase. Maximum amount of phase is set to 8. Each of them is identified by a label ranging from 1 to 8.

BASE ADDRESS: 1600(0x0640) + 27*(Phase label – 1).

Index	Name	Description	Type
0H	bRatioAvailableW	Ratio between output load and available power in watts (%)	U8
0L	bRatioAvailableVA	Ratio between output load and available power in VA (%)	U8
1H	bRatioInstalledW	Ratio between output load and installed power (Nb modules – redundancy) in watts (%)	U8
1L	bRatioInstalledVA	Ratio between output load and installed power (Nb modules – redundancy) in VA (%)	U8
2	wVout	Output voltage value (0.1V)	U16
3	wIout	Output current value (0.1A)	U16
4H	bNbOndCfg	Number of modules configured in the phase	U8
4L	bRedundancy	Amount of redundancy configured in the phase	U8
5	wACOutFreq	AC output frequency value (0.1Hz)	U16
6	lPinDC	DC input power value (W)	U32
8	lPinACW	AC input power value (W)	U32
10	lPinACVA	AC input power value (VA)	U32
12	lCurrentPowerInVA	Output power value (VA)	U32
14	lCurrentPowerInW	Output power value (W)	U32
16	lInstalledPowerInW	Installed power value (W)	U32
18	lInstalledPowerInVA	Installed power value (VA)	U32
20	lAvailablePowerInW	Available power value (W)	U32
22	lAvailablePowerInVA	Available power value (VA)	U32
24H	bNbInvSeen	Number of module seen by T2S in that phase	U8
24L	bNbInvOK	Number of modules that are delivering output in the phase	U8
25H	bNbInvMO	Number of modules manually off in the phase	U8
25L	bNbInvKO	Number of modules that are not delivering output due to a failure in the phase	U8
26H	bNbInvNT	Number of modules not seen by T2S in the phase (accordingly to bNbOndCfg)	U8

AC group table (0x0730)

The table described below represents the information that can be retrieved regarding a particular AC group. Maximum amount of AC group is set to 4. Each of them is identified by a label ranging from 1 to 4.

BASE ADDRESS: 1840(0x0730) + 10*(AC group label – 1).

Index	Name	Description	Type
0H	bNbInvOK	Number of modules that are delivering output in the group	U8
0L	bNbInvMO	Number of modules manually off in the group	U8
1H	bNbInvKO	Number of modules that are not delivering output due to a failure in the group	U8
1L	bNbInvSeen	Number of module seen by T2S in that group	U8
2	lPinACW	AC input power value (W)	U32
4	lPinACVA	AC input power value (VA)	U32
6	wVinAC	AC input voltage value (0.1V)	U16
7	wIinAC	AC input current value (0.1A)	U16
8	wACInFreq	AC input frequency value (0.1Hz)	U16
9H	bACInOk	Number of modules stating that their AC input stage is fully functional	U8

DC group table (0x076C)

The table described below represents the information that can be retrieved regarding a particular DC group. Maximum amount of DC group is set to 8. Each of them is identified by a label ranging from 1 to 8.

BASE ADDRESS: 1900(0x076C) + 7*(DC group label – 1).

Index	Name	Description	Type
0H	bNbInvOK	Number of modules that are delivering output in the group	U8
0L	bNbInvMO	Number of modules manually off in the group	U8
1H	bNbInvKO	Number of modules that are not delivering output due to a failure in the group	U8
1L	bNbInvSeen	Number of module seen by T2S in that group	U8
2	lPinDC	DC input power value (W)	U32
4	wVinDC	DC input voltage value (0.1V)	U16
5	wIinDC	DC input current value (0.1V)	U16
6H	bDCInOk	Number of modules stating that their DC input stage is fully functional	U8

Miscellaneous information table (0x07BC)

The table described below represents the miscellaneous information that can be retrieved regarding T2S and system.

BASE ADDRESS: 1980(0x07BC)

Index	Name	Description	Type
0H	bOldVersionNumber	Deprecated. Always 0x00	U8
0L	ePhaseNumber	Number of phase configured in the system	U8
1	lSerialNumber	T2S serial number	U32
3	wTempoMajorAl	Temporization of major alarm relay	U16
4	wTempoMinorAl	Temporization of minor alarm relay	U16
5H	bNbMajor	Number of major alarm in the system	U8
5L	bNbMinor	Number of minor alarm in the system	U8
6H	bNbTotalAlarmNumber	Total number of alarm in the system	U8
6L	bACInputPresent	Flag (true or false) that indicates if AC input should be considered as present or not	U8
7H	bSaturationThresh	Value of the ratio over which the saturation alarm will be raised (%)	U8
7L	bNbGroupsDC	Number of DC groups configured in the system	U8
8H	bNbGroupsAC	Number of AC groups configured in the system	U8
8L	bProgRelay	Alarm number on which programmable relay will trigger	U8
9	wSoftMainRevision	Main revision software number of T2S	U16
10	wSoftSubRevision	Sub revision software number of T2S	U16
11H	bSystemLoadPosition	Position of the load at the system level (0:AC, 100:DC, 50:mixed, 101:unknown)	U8
11L	bT2SMaxKnownParameters	Version number of TSI modules configuration parameters	U8
12	wT2SVersionTextError	Version number of TSI event texts	U16
13H	bNbrModConf	Total number of module configured on the installation	U8
13L	bNbrModSeen	Total number of module configured on the installation	U8

14H	bNbrEvent	Number of event present in history log	U8

Date and time table (0x07D0)

The table described below represents the information that can be retrieved regarding date and time.

BASE ADDRESS: 2000(0x07D0)

Index	Name	Description	Type
0	lTime	Time in epoch	U32
2H	bSeconds	Seconds number	U8
2L	bMinutes	Minutes number	U8
3H	bHours	Hours number	U8
3L	bDay	Day of the month	U8
4H	bMonth	Month number	U8
4L	bDaylightSaving	Flag (true or false) that specify if daylight saving is enable or not	U8
5	wYear	Year number	U16

Alarm table (0x07DA)

The table described below represents the information that can be retrieved regarding alarms. Maximum amount of entries is set to 50. A valid entry represents an alarm present in the system. An invalid entry is an entry where all bits of each field are set. All entries following an invalid entry will be invalid.

BASE ADDRESS: 2010(0x07DA) + 2*(Entry number – 1)

Index	Name	Description	Type
0H	bDeviceNumber	Identifier that specifies which device is responsible of this alarm (see A1)	U8
0L	bEventType	Type of the alarm (Major or minor) (see A1)	U8
1	wEventNumber	Alarm number identifier	U16

Rem: See alarm types in the annexes for *bEventType* description.

History log table (0x0870)

The table described below represents the information that can be retrieved regarding history log events. Maximum amount of entries is set to 200. A valid entry represents an event logged by the T2S. An invalid entry is an entry where all bits of each field are set. All entries following an invalid entry will be invalid.

BASE ADDRESS: 2160(0x0870) + 6*(Entry number – 1)

Index	Name	Description	Type
0H	bDeviceNumber	Identifier that specifies which device is responsible of this alarm (see A1)	U8
0L	bEventType	Type of the alarm (Major, minor or no alarm) (see A1)	U8
1	wEventNumber	Event number identifier	U16
2	wYear	Year when the event was logged	U16
3H	bMonth	Month when the event was logged	U8
3L	bDay	Day when the event was logged	U8
4H	bHours	Hour when the event was logged	U8
4L	bMinutes	Minute when the event was logged	U8
5H	bSeconds	Second when the event was logged	U8

Configuration table (0x1040)

The table described below represents the information that can be retrieved regarding parameters that can be configured in the T2S controller. Maximum amount of entries is set to 500. Not all entries are valid. An invalid entry is an entry where all bits of each field is set. Invalid entries might be interleaved with valid entries.

BASE ADDRESS: 4160(0x1040) + 20*(Entry number – 1)

Index	Name	Description	Type
0	swParameter	Configured value of the parameter	S16
1	wValidity	Value indicating if last configured parameter value is valid (see A2)	U16
2	wIdentifier	Unique value identifying the parameter	U16
3	wUnit	Value indicating in which units the parameter is expressed (see A2)	U16
4	strParamDescription	Textual description of the parameter	32*U8

Event string table (0x4114)

The table described below represents the information that can be retrieved regarding event textual description. Maximum amount of entries is set to 300. Each event is identified by an unique number (Event 0 exists !).

BASE ADDRESS: 16660(0x4114) + 8*Event number

Index	Name	Description	Type
0	strEventTxt	Textual description of event	16*U8

Parameter validity string table (0x4CCC)

The table described below gives access to textual description regarding parameter validity. This table should be used in conjunction with annex A2 and wValidity field of "Configuration Table" above. Maximum number of description is set to 20.

BASE ADDRESS: 19660(0x4CCC) + 8*(Description Index – 1)

Index	Name	Description	Type
0	strValidityReasonTxt	Textual description of the parameter validity or invalidity	16*U8

HOLDING REGISTERS ELEMENTS (Read-Write 16-bit wide)

Configuration table (0x0000)

The table described below represents the information that can be retrieved regarding parameters that can be configured in the T2S controller. Maximum amount of entries is set to 500. These values can be written to modify the configuration. Not all entries are valid. Writing to an invalid entry is without effect. It's important to note that writing those fields do not change the current configuration but only alter temporary register. **In order for this table to be written, the "validate password table" should have been written with correct password.**

BASE ADDRESS: 0(0x0000) + (Entry number – 1)

Index	Name	Description	Type
0	swParameter	Configured value of the parameter	S16

Configuration control table (0x05D0)

This register contains values used for controlling the configuration table describes above. Once the new configuration has been written using the "Configuration Table" describe above. Change can be submitted using this register. If configuration is valid, it will be applied. Otherwise nothing would be changed and temporary modification will be cancelled. Reject reason can be obtain by reading wValidity field of "Configuration Table" in the input register variable class. **In order for this table to be written, the "validate password table" should have been written with correct password.**

BASE ADDRESS: 1488(0x05D0)

Index	Name	Description	Type
0H	bBitField	Bit Field (see A3)	U8
0L	Reserved	Reserved. Always write as 0x00	U8

Date and time table (0x05DC)

The table described below represents the information that can be written regarding date and time. **In order for this table to be written, the “validate password table” should have been written with correct password.**

BASE ADDRESS: 1500(0x05DC)

Index	Name	Description	Type
0H	bSeconds	Seconds number	U8
0L	bMinutes	Minutes number	U8
1H	bHours	Hours number	U8
1L	bDay	Day of the month	U8
2H	bMonth	Month number	U8
2L	bDaylightSaving	Flag (true or false) that specify if daylight saving is enable or not	U8
3	wYear	Year number	U16

Module actions table (0x05E6)

The table described below represents the action that can be performed regarding a particular module. Maximum amount of module is set to 32. Each of them is identified by an address ranging from 1 to 32. **In order for this table to be written, the “validate password table” should have been written with correct password.**

BASE ADDRESS: 1510(0x05E6) + 3*(Module address – 1).

Index	Name	Description	Type
0H	bAddress	Configured address	U8
0L	bPhaseNumber	Number of the phase module is belonging to	U8
1H	bGroupAC	AC input group number module is belonging to	U8
1L	bGroupDC	DC input group number module is belonging to	U8
2H	bActionBitField	Bit Field specifying actions to perform on module (see A4)	U8
2L	Not used	Not used (write as 0x00)	U8

Remark: Refer to note 2 at the end of this document to ensure safe use of these functionalities!

Digital Inputs Labels (0x0686)

The table described below represents the textual description that can be written regarding digital inputs. Digital input 1 shorted is event number 227 and Digital input 2 shorted is event number 228. **In order for this table to be written, the “validate password table” should have been written with correct password.**

BASE ADDRESS: 1670(0x0686) + 8*(label number – 1).

Index	Name	Description	Type
0	strDigitalInpLabel	Textual description of the digital input function	16*U8

System Actions Table (0x06A4)

The table described below represents the action that can be performed regarding a the system and the T2S controller. **In order for this table to be written, the “validate password table” should have been written with correct password.**

BASE ADDRESS: 1700(0x06A4)

Index	Name	Description	Type
0H	bActionBitField	Bit Field specifying actions to perform on system (see A4)	U8
0L	Reserved	Reserved. Always write as 0x00	U8
1H	bTaskState	Bit Field specify the state of different tasks (see A4)	U8
1L	Reserved	Reserved. Always write as 0x00	U8

New Modbus write password table (0x07D0)

The table described below represents the password for Modbus write access. **In order for this table to be written, the “validate password table” should have been written with correct password. Reading this table always return “0x00” in all fields.**

BASE ADDRESS: 2000(0x07D0)

Index	Name	Description	Type
0	strNewModbusWritePassword	New password string for Modbus write access	16*U8

Validate password table (0x0834)

The table described below is the field that should be written to validate Modbus write access password. In order to change the configuration or perform action on the system, this field should be written with the correct password to grant access to the desired action mentioned above. As soon as the action is performed this field is invalidated. Thus prior to any action, this field should be written with correct password. Reading this table will always return “0x00” in all fields.

BASE ADDRESS: 2100(0x0834)

Index	Name	Description	Type
0	strModbusWritePassword	Password string for Modbus write access	16*U8

Annexes: Status and constants description

Module status explanation (A1) :

- eStatusACOut :

Name	Description	Value
SBR	Standby running. This means that the module is delivering output	0
SB	Standby. This mean that the module is manually OFF	1
SBWE	Standby with error. This means that the module is not delivering output due to an unrecoverable error	2
SBWRE	Standby with recoverable error. This means that the module is not delivering output due to a recoverable error.	3
UNKNOWN	Unknown. This means status is unknown	4

- eStatusACIn :

Name	Description	Value
OK	OK. This means the AC input is OK for the module	0
SAFE	Safe. This means the AC input is not considered as “good” but some power can still be drawn from it.	1
NOT_SYNC	Not synchronized. This means that the AC input and output are not synchronized together thus invalidating AC input.	2
OFF	Off. This means that the AC input stage of the module has been turned off due to an invalid AC input (maybe not safe).	3
UNKNOWN	Unknown. This means the status is unknown	4

- eStatusDCIn :

Name	Description	Value
OK	OK. This means the AC input is OK for the module	0
FAIL	Fail. This means the DC input voltage is out of valid range.	1
UNKNOWN	Unknown. This means the status is unknown	2

Alarm types :

Name	Description	Value
NO_ALARM	Defines an event that is not considered as an alarm	0
MINOR	Defines an event that is to be considered as a minor alarm	1
MAJOR	Defines an event that is to be considered as a major alarm	2

Alarm sources :

Name	Description	Value
T2S	Device responsible of the alarm is the T2S controller.	0
MOD XX	Device responsible of the alarm is the module number XX where XX is the value	1-32
SYSTEM	Source of the alarm is the whole system (e.g. if all module are sharing the same alarm).	33

Validity and Unit description (A2) :

wValidity should be interpreted as follow :

Name	Description	Value
PARAM_OK	Parameter value is valid	0
PARAM_TOO_LOW	Parameter value is too low	1
HYST_TOO_LOW	Parameter value is in an acceptable range but is too close from another related parameter value	2
PARAM_TOO_HIGH	Parameter value is too high	3
TSI_MUST_BE_OFF	Parameter value can only be changed if TSI modules are not delivering output	4
BAD_VALUE	Parameter value is not acceptable	5
INV_MISMATCH	Parameter cannot be configured for that type of module	6

wUnit is divided in two part :

- High byte is exponent value for parameter conversion (e.g. 2 means to be divided by 10^2 = 100).
- Low byte represents the unit in which the parameter is expressed. This unit can be one of the one represented in the array below.

Name	Description	Value
NO_UNIT	No unit. Represented by a blank character	0
VOLT	Volt. Represented by the “V” character	1
AMPERE	Ampere. Represented by the “A” character	2
HERTZ	Hertz. Represented by the “Hz” characters	3
SECOND	Second. Represented by the “s” character	4
ANGLE	Angle. Represented by the “deg” or “°” characters	5
WATT	Watt. Represented by the “W” character	6
VA	VA. Represented by the “VA” character	7
PERCENT	Percent. Represented by the “%” character	8
DEGREE	Degree. Represented by the “deg” or “°” characters	9
OHM	Ohm. Represented by the “Ohm” character	10

Example: if wUnit value is 0x0201 the parameter is expressed in centiVolts.

Configuration control bit field explanation (A3) :

Field Description	Bit Position
Set this bit to apply the new written configuration	0
Set this bit to cancel modification of configuration table	1
Set this bit if default configuration should be restored	2
Set this bit if factory configuration should be restored	3
Not used (always cleared)	4
This bit is set when configuration parameters are currently checked.	5
This bit is set if temporary configuration is matching current configuration	6
This bit is set if last submitted configuration was valid	7

Action bit field explanation (A4) :

- Module actions :

Action Description	Bit Position
Set this bit if module is to be turned ON (This bit is cleared when action has been transmitted to module)	0
Set this bit if module is to be turned OFF (This bit is cleared when action has been transmitted to module)	1
Set this bit if module is to be identified (Front leds blinking) (This bit is cleared when action has been transmitted to module)	2
Set this bit to notify the module that it’s fan has been replaced (to clear “REPLACE FAN” event)	3
Set this bit to clear lBusErrorCnt counter of each module	4
Not used (always cleared)	5
Not used (always cleared)	6
Not used (always cleared)	7

- System actions :

Action Description	Bit Position
Not used (always cleared)	0
Not used (always cleared)	1
Set this bit if history log should be cleared	2
Set this bit if system should be turned ON	3
Set this bit if system should be turned OFF	4
Set this bit to refresh configuration texts from modules	5
Set this bit to refresh event texts from modules	6

Set this bit to automatically assign module address	7

- Task state :

Action Description	**Bit Position**
This bit is set when a refresh of T2S configuration base on modules memory is pending. This operation can take several minutes (max 20 min).	0
This bit is set while a refresh of event texts from modules is pending. This operation can take several minutes (max 15 min).	1
This bit is set when T2S's configuration is currently saved in modules memory. This operation can take several minutes (max 60 min).	2
This bit is set when a refresh of T2S's configuration base on modules memory is pending. This operation can take several minutes (max 10 min).	3
This bit is set while an action (new address, new group, turn ON, restrain action, …) is pending on the system.	4
This bit is set when configuration parameters are currently checked.	5
Toggle bit which indicate that volatile data have been updated.	6
CE+T usage only - Force custom configuration.	7

SECTION 3: How to and examples

1°) Introduction:

In all the following examples, assumption will be made that T2S controller Modbus slave address is 1 (*0x01*) and that Modbus password for write access is "Modbus_T2S" (*0x4D 0x6F 0x64 0x62 0x75 0x73 0x5F 0x54 0x32 0x53*)

1°) Reading simple variables:

Ex 1: Reading output voltage of module #5

Field	**Value**	**Description**
Function	4 (*0x04*)	Read input register
Address	128 (*0x0080*)	31*(5-1) + 4 = 128 (see module table)
Number of registers	1 (*0x01*)	Vout value is 16-bit wide

Master frame : *0x01 0x04 0x00 0x80 0x00 0x01 0x71 0xE3*

T2S frame : *0x01 0x04 0x02* ***0x09 0x1B*** *0xFF 0x6B*

Received value: 0x091B = 2331 → Output voltage is 233.1V (see module table)

Ex 2: Reading T2S serial number

Field	Value	Description
Function	4 (*0x04*)	Read input register
Address	1989 (*0x07C5*)	1980 + 9 = 1989 (see miscellaneous information table)
Number of registers	2 (*0x02*)	Main and sub software version are both 16-bit wide

Master frame : *0x01 0x04 0x07 0xC5 0x00 0x02 0x60 0x82*
T2S frame : *0x01 0x04 0x04* ***0x00 0x01 0x00 0x07*** *0xEB 0x86*

Received value: 0x0001 and 0x0007 → T2S revision is Vs1.7

2°) Reading alarm and history log :

Reading entry #1

Field	Value	Description
Function	4 (*0x04*)	Read input register
Address	2010 (*0x07D4*)	2010 + 2*(1-1) = 2010 (see alarm table)
Number of registers	2 (*0x02*)	Alarm entry is 2 registers wide

Master frame : *0x01 0x04 0x07 0xDA 0x00 0x02 0x51 0x44*
T2S frame : *0x01 0x04 0x04* ***0x21 0x01 0x00 0xB3*** *0xE1 0xCD*

Alarm entry #1 is a minor (0x01) alarm generated by the system (0x21) and this alarm has the ID 179 (0x00B3)

Reading an invalid entry

Let's assume that there is only 2 alarms present in the system. Then reading alarm entry #3 should return an invalid entry

Field	Value	Description
Function	4 (*0x04*)	Read input register
Address	2014 (*0x07DE*)	2010 + 2*(3-1) = 2014 (see alarm table)
Number of registers	2 (*0x02*)	Alarm entry is 2 registers wide

Master frame : *0x01 0x04 0x07 0xDE 0x00 0x02 0x10 0x85*
T2S frame : *0x01 0x04 0x04* ***0xFF 0xFF 0xFF 0xFF*** *0xFA 0x10*

Conclusion, there is no alarm entry #3 and nor are there further entries. This leads to the conclusion that only 2 alarms are present at the time in the system.

Linking alarm ID to alarm description text :

If we consider the alarm ID #179 of example above, we can get description text for this alarm by reading related entry in the “Event string table”.

Field	**Value**	**Description**
Function	4 (*0x04*)	Read input register
Address	18092 (*0x46AC*)	16660 + 8*179 = 18092
Number of registers	8 (*0x08*)	Event description string is 16 characters long

```
Master frame : 0x01 0x04 0x46 0xAC 0x00 0x08 0x24 0xA5
T2S frame    : 0x01 0x04 0x10 0x56 0x61 0x63 0x5F 0x69 0x6E 0x20
               0x54 0x4F 0x4F 0x20 0x4C 0x4F 0x57 0x20 0x20 0x36
               0x7C
```

String description: Vac_in TOO LOW

3) Reading configuration:

Reading entry #1

Field	**Value**	**Description**
Function	4 (*0x04*)	Read input register
Address	4160 (*0x1040*)	4160 + 20*(1-1) = 4160
Number of registers	20 (*0x14*)	Alarm entry is 20 registers wide

```
Master frame : 0x01 0x04 0x10 0x40 0x00 0x14 0xF5 0x11
T2S frame    : 0x01 0x04 0x28 0x01 0xB8 0x00 0x00 0x01 0x04 0x01
               0x01 0x44 0x43 0x20 0x31 0x20 0x3A 0x20 0x56 0x64
               0x63 0x5F 0x69 0x6E 0x20 0x4C 0x6F 0x77 0x20 0x53
               0x74 0x61 0x72 0x74 0x20 0x20 0x20 0x20 0x20 0x20
               0x20 0x20 0x20 0x64 0x36
```

```
Configured Value   : 0x01B8 → 440
Validity           : 0x0000 → PARAM_OK (see A2)
Parameter ID       : 0x0104 → 260
Units              : 0x0101 → unit is dV (0.1V) (see A2)
String description : DC 1 : Vdc_in Low Start
```

Reading an invalid entry

Let’s assume that entry #189 is invalid

Field	**Value**	**Description**
Function	4 (*0x04*)	Read input register
Address	7920 (*0x1EF0*)	4160 + 20*(189-1) = 7920

Number of registers	20 (*0x14*)	Alarm entry is 20 registers wide

Master frame : *0x01 0x04 0x1E 0xF0 0x00 0x14 0xF6 0x1E*
T2S frame   : *0x01 0x04 0x28* ***0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF***
0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF
0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF
0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF
0xFF 0xFF 0xFF *0xF0 0x04*

Due to internal memory organization, the configuration might have valid entries interleaved with invalid ones. Thus, in order to read all configuration, one should read all entries to determine which ones are valid and which ones are not.

Exceptions: Textual parameter

As one can see, the configured value field is 16 bit wide. Consequently, only integer values can be read (or further configured) using this way. There are 3 parameters that are not integer values but strings. Thus, the values returned in the “Configured value” field of the configuration table for those 3 IDs are dummy values that have no meaning.

Those IDs are the following

ID	Description	Remark
901	Digital input 1 label	Can be read in the digital inputs table in the holding registers (0x0686)
902	Digital input 2 label	Can be read in the digital inputs table in the holding registers (0x0686)
955	Modbus write password	Can’t be read using Modbus

4°) Performing actions on the system

Actions on system are password protected. Prior to any action, the password for Modbus action access should be validated. This password is invalidated as soon as the requested action has been performed.

Let’s suppose we want to turn OFF module #1 and that the password for Mobus action access is “Modbus_T2S” (*0x4D 0x6F 0x64 0x62 0x75 0x73 0x5F 0x54 0x32 0x53*).

Performing action without password validation

Field	Value	Description
Function	6 (*0x06*)	Write single register
Address	1512 (*0x05E8*)	1510 + 3*(1-1) + 2 = 1512
Value	512(*0x0200*)	See A4

Master frame : *0x01 0x06 0x05 0xE8 0x02 0x00 0x08 0x52*
T2S frame   : *0x01* ***0x86 0x04*** *0x43 0xA3*

T2S return an error with SLAVE_DEVICE_FAILURE (0x04)

Performing password validation prior to action

Field	Value	Description
Function	16 (*0x10*)	Write multiple register
Address	2100 (*0x0834*)	2100
Number of register	5 (*0x05*)	Modbus_T2S is 10 character long
Value	*0x4D 0x6F 0x64 0x62 0x75 0x73 0x5F 0x54 0x32 0x53*	Modbus_T2S

```
Master frame : 0x01 0x10 0x08 0x34 0x00 0x05 0x0A 0x4D 0x6F 0x64
               0x62 0x75 0x73 0x5F 0x54 0x32 0x53 0x0A 0xFC
T2S frame    : 0x01 0x10 0x08 0x34 0x00 0x05 0x43 0xA4
```

Warning: There is no way to know if password was correctly validated or not. That is sending an invalid password will produce the same answer …

```
Master frame : 0x01 0x06 0x05 0xE8 0x02 0x00 0x08 0x52
T2S frame    : 0x01 0x06 0x05 0xE8 0x02 0x00 0x08 0x52
```

5°) Writing and modification system configuration :

The configuration table of the holding register section is the companion table of the configuration table of the input register section. That's to say that entries are linking to the same parameter.

Thus entry #1 will still refer to the "`DC 1 : Vdc_in Low Start`" (see example 3)

Writing entry #1:

Let's suppose that I want to change the configured value of 440 (0x01B8) to 420 (0x01A4).

Field	Value	Description
Function	6 (*0x06*)	Write single register
Address	0 (*0x0000*)	0 + (1-1) = 0
Value	420(*0x01A4*)	New value

```
Master frame : 0x01 0x06 0x00 0x00 0x01 0xA4 0x89 0xE1
T2S frame    : 0x01 0x06 0x00 0x00 0x01 0xA4 0x89 0xE1
```

There is no need to validate the password before sending this command because no action has been taken so far. Indeed, only the temporary structure has been modified and thus the modification is still not applied.

In order to apply or cancel modifications made to temporary structure the "Configuration control table" should be written accordingly.

Applying modifications:

Field	Value	Description
Function	6 (*0x06*)	Write single register
Address	1448 (*0x05D0*)	1448
Value	256(*0x0100*)	See A3

```
Master frame : 0x01 0x10 0x08 0x34 0x00 0x05 0x0A 0x4D 0x6F 0x64
                0x62 0x75 0x73 0x5F 0x54 0x32 0x53 0x0A 0xFC
               (Password validation)
T2S frame    : 0x01 0x10 0x08 0x34 0x00 0x05 0x43 0xA4
Master frame : 0x01 0x06 0x05 0xD0 0x01 0x00 0x89 0x6F
T2S frame    : 0x01 0x06 0x05 0xD0 0x01 0x00 0x89 0x6F
```

After sending this command, the T2S will check the configuration located in the temporary structure. If this one is valid, it will be applied and save. Otherwise, the previous configuration will be restored in that temporary structure (same effect as the cancel action). Because the validation process might last during an undetermined amount of time it is useful to exactly know when the process is complete. In order to do so, one should poll the control configuration table. The 6th bit of the bitfield can provide this information.

Reading control table configuration bitfield :

Field	Value	Description
Function	3 (*0x03*)	Read holding register
Address	1448(*0x05D0*)	1448
Number of registers	1 (*0x01*)	Table value is 16-bit wide

- Before any modification :

```
Master frame : 0x01 0x03 0x05 0xD0 0x00 0x01 0x85 0x3F
T2S frame    : 0x01 0x03 0x02 0xC0 0x00 0xE8 0x44
```

Bit 6 is set → Temporary configuration match current configuration
Bit 7 is set → Last submitted configuration is valid

- Modification of entry #1 with new value 420 :

```
Master frame : 0x01 0x06 0x00 0x00 0x01 0xA4 0x89 0xE1
T2S frame    : 0x01 0x06 0x00 0x00 0x01 0xA4 0x89 0xE1
```

- Reading of configuration control table :

```
Master frame : 0x01 0x03 0x05 0xD0 0x00 0x01 0x85 0x3F
T2S frame    : 0x01 0x03 0x02 0x80 0x00 0x09 0x84
```

Bit 6 is cleared → Temporary configuration is not matching current configuration
Bit 7 is set → Last submitted configuration is valid

- Submitting the modification

```
Master frame : 0x01 0x10 0x08 0x34 0x00 0x05 0x0A 0x4D 0x6F 0x64
               0x62 0x75 0x73 0x5F 0x54 0x32 0x53 0x0A 0xFC
              (Password validation)
T2S frame    : 0x01 0x10 0x08 0x34 0x00 0x05 0x43 0xA4
Master frame : 0x01 0x06 0x05 0xD0 0x01 0x00 0x89 0x6F
T2S frame    : 0x01 0x06 0x05 0xD0 0x01 0x00 0x89 0x6F
```

- Reading of configuration control table :

```
Master frame : 0x01 0x03 0x05 0xD0 0x00 0x01 0x85 0x3F
T2S frame    : 0x01 0x03 0x02 0xC0 0x00 0xE8 0x44
```

Bit 6 is set → Temporary configuration match current configuration (indicating that processing is done)
Bit 7 is set → Last submitted configuration is valid (indicating that changes had been applied)

Getting the reason why a parameter is not valid :

Let's suppose that we want to set "`DC 1 : Vdc_in Low Start`" to an invalid value of 10 (0x000A) corresponding to 1 V.

- Modification of entry #1 with new value 10 :

```
Master frame : 0x01 0x06 0x00 0x00 0x00 0x0A 0x09 0xCD
T2S frame    : 0x01 0x06 0x00 0x00 0x00 0x0A 0x09 0xCD
```

- Reading of configuration control table :

```
Master frame : 0x01 0x03 0x05 0xD0 0x00 0x01 0x85 0x3F
T2S frame    : 0x01 0x03 0x02 0x80 0x00 0x09 0x84
```

Bit 6 is cleared → Temporary configuration is not matching current configuration
Bit 7 is set → Last submitted configuration is valid

- Submitting the modification

```
Master frame : 0x01 0x10 0x08 0x34 0x00 0x05 0x0A 0x4D 0x6F 0x64
               0x62 0x75 0x73 0x5F 0x54 0x32 0x53 0x0A 0xFC
              (Password validation)
T2S frame    : 0x01 0x10 0x08 0x34 0x00 0x05 0x43 0xA4
Master frame : 0x01 0x06 0x05 0xD0 0x01 0x00 0x89 0x6F
T2S frame    : 0x01 0x06 0x05 0xD0 0x01 0x00 0x89 0x6F
```

- Reading of configuration control table :

```
Master frame : 0x01 0x03 0x05 0xD0 0x00 0x01 0x85 0x3F
T2S frame    : 0x01 0x03 0x02 0x40 0x00 0x89 0x84
```

Bit 6 is set → Temporary configuration matching current configuration (processing is done)
Bit 7 is cleared → Last submitted configuration is not valid (modification has been cancelled)

In order to know why a parameter has been rejected, one should read again the `wValidity field` of the configuration control in the Input registers section (see example 3).

Field	**Value**	**Description**
Function	4 (*0x04*)	Read input register
Address	4161 (*0x1041*)	4160+20*(1-1)+1 = 4161
Number of registers	1 (*0x01*)	wValidity is 16 bit wide

```
Master frame : 0x01 0x04 0x10 0x41 0x00 0x01 0x65 0x1E
T2S frame    : 0x01 0x04 0x02 0x00 0x02 0x38 0xF1
```

```
Validity: 0x0002 → HYST_TOO_LOW (see A2)
```

Textual description of the parameter validity condition can be obtained by reading the “parameter validity string” table in the input register section.

Note 1:

Setting Modbus password to an empty string is not prohibited. But doing so does not allow user from not validating password before performing actions. The thing is that any password sent to be validated will succeed and grant access to actions.

Note 2:

Caution should be taken while changing module address because it will affect the addresses where to retrieve information regarding this module. What is more there can be a delay between the moment where the change address order is received and the moment where the module address has been physically changed. Moreover, a module address can be changed to a new address that is already assigned to another module! In this case, the modules will swap their addresses.

For all those reasons, the best and secure way to change a module address is the following one:

1°) Get module serial number using “Module informat ion table” using the current address to calculate the index.

2°) Send to the new address for this module using t he “Module action table” using the current address to calculate the index.

3°) Poll the serial number using “Module informatio n table” using the new address as index until there is a match with the serial number collected at point 1.