# EMA

## Eurotherm Monitoring & Acquisition Unit

## PROFIBUS-DP & MODBUS Interface

# Contents

## Introduction

### Communication protocol and parameter selection

**The communication protocol and parameters can only be set using the configurator, which runs on a PC (W95, W98 or WNT).**

**Important note! Any modifications to communication parameters are only implemented the next time the equipment is switched on.**

**The configurator can be used to set the following parameters:**

- **Comm Profibus**
- **Comm Modbus**
  - **Baud rate**
  - **Parity**
  - **Enable time out**

**Profibus communication parameters are set automatically.**

# 1. The Profibus DP+ interface

## 1.1. Introduction

**For details of the notions used in these specifications, please refer to the following publications:**

**-Standard DIN 19245 / Part 1: PROFIBUS.**
**- Data transmission technique,**
**- Medium access methods and transmission protocols,**
**- Service interface to the application layer,**
**- Management.**

**- Standard DIN 19245 / Part 3: PROFIBUS DP.**
**- Process Fieldbus: Decentralised Periphery.**

**- Standard ISO/IEC 8802-2: Information processing systems.**
**- Logical Link Control.**

**- Standard ISO/IEC 8802-4: Information processing systems.**
**- Token passing bus access method and physical layer specifications.**

**- Standard CEI 1131-3: programmable controllers**
**- programming Languages**

**- Siemens SPC3 User Description. Version V1.0c - 12/1995.**

## 1.2. General characteristics of the Profibus DP bus

**The purpose of this short section is to give a brief overview of the fundamental principles underlying the Profibus DP system, as described in around 1,100 pages of the Standard EN50170.**

**PROFIBUS DP (Decentralised Peripherals) was developed as a means of providing extremely fast control over decentralised INPUT/OUTPUTS.**
**To this end, the same means must be used to send:**

- **The Parameters**
- **The system Configuration**
- **The measured Process values**
- **Errors and States**

**This type of Communication requires especially fast response times.**
**The table below summarises the main characteristics of the Profibus DP system.**

| Characteristics required | PROFIBUS DP characteristics |
|---|---|
| Fast response times | Over 1,000 digital input/outputs exchanged between 32 different devices in under 10ms (at 12MBds) |
| Single and multiple master operations | Hybrid bus access |
| Simple, inexpensive protocol | Layers 1 and 2 of the OSI model are incorporated into ASICs |
| Diagnostics functions | Diagnostics functions shared between Master and Slaves |
| Simple user interface | Parameters and Configurations are pre-set for the user |
| Existing medium and line tools | PROFIBUS FMS and PROFIBUS DP share the same RS485 medium already installed at most sites. |
| Interoperability | PNO certifies compliance with the Standard. |

## 1.3. Characteristics of EMA's PROFIBUS DP interface

**The PROFIBUS-DP Slave interface has been designed in full compliance with the relevant standard - EN50170 (DIN 19245-3).**

**PNO identification number: 0x00FF. (255 in decimal).**

**The Profibus protocol is controlled by the common motherboard.**

**Layers 1 and 2 of the OSI model are controlled by a dedicated component - SPC3. Most of the specifications presented here are derived from this ASIC's characteristics.**

**The Bus connection uses a shielded twisted pair (RS485) cable.**
**Binary transmission frame: 1 start bit**
**8 data bits**
**1 parity bit**
**1 stop bit.**

**Self-adjusting communication baud rate**
**9.6/19.2/93.75/187.5/500/1,500 Kbaud.**
**The baud rate selection function is activated automatically whenever the interface is Initialised and after a Watchdog Time-Out.**

**The communication state is shown both by the LEDs on the unit's Front Panel and by a symbol on the display.**

**The Unit's Bus Address is either set directly from the Master by the communication or else by the Configurator running on a PC.**

**In the event of an Error on the Bus, a Watchdog informs the User and restarts the unit automatically. The watchdog is reset after each error-free frame sent to the Unit.**
**The Time Out can be set using any Profibus DP Configurator. The Time-Out must be a value between 2ms and 650s.**
**The Time Out is calculated as follows:**

$$T(WD) = Constant * WD\_1 * WD\_2$$

**The constant is either 1ms or 10ms and is set during the configuration phase.**
**The value WD_1*WD_2 =1 is not allowed.**
**Neither WD_1 nor WD_2 can be set to 0.**

**The system does not handle redundancy.**

### 1.4. Data exchange principle - the State Diagram.

**In accordance with the description given in Standard EN50170, the Interface receives Parametrisation and Configuration data. Both types of exchange are required for system initialisation purposes, failing which the system cannot be started.**

- **The device cannot switch to the DATA_EXCHANGE STATE until the PARAMETRISATION and CONFIGURATION sequences have both been completed successfully.**

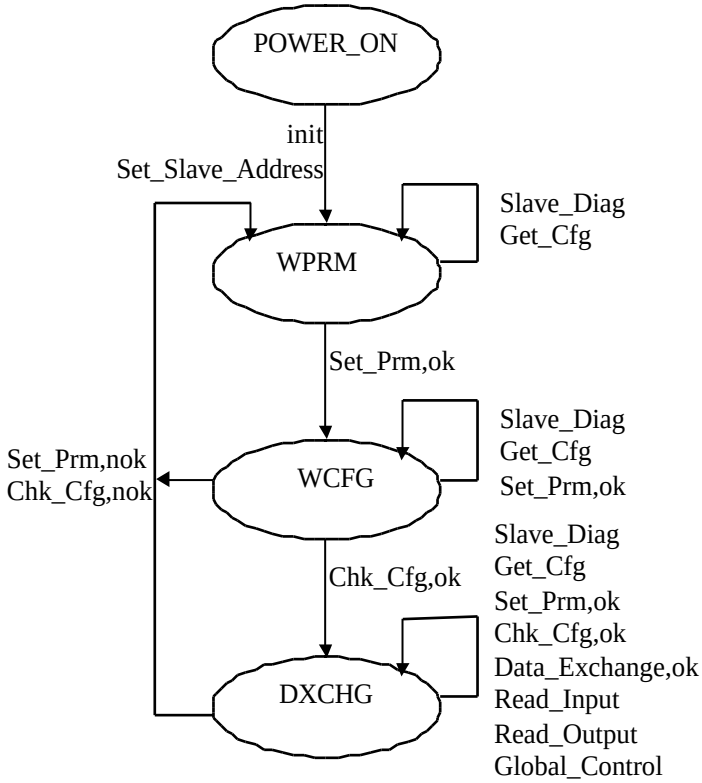
**When the system is started, it is automatically set to a wait state.**
**The simplified State diagram shows the various possible States and the associated Services.**

- **POWER_ON: The system is correctly powered; this is the start of the initialisation phase.**
  - **Note: Addresses can only be changed in this State (using SET_SLAVE_ADD).**
  - **The product's address is factory-set to 32. (decimal).**
- **WPRM = Wait for Parametrisation. The device waits for data in order to be able to start (PNO ID, synchronisation or freeze mode accepted, etc…). Diagnostics queries are also accepted in this State.**
  - **All other message types are rejected while in this state.**

**- WCFG = Wait for Configuration. This kind of message can be used to reserve the necessary space in the Input and Output Buffers of the device and its Master, and to specify the data structure used for transfers. In this State, the Unit waits for a Configuration, Parametrisation or Diagnostics message.**

- **All other message types are rejected while in this state.**

**- DXCHG = Data Exchange. Once the Parametrisation and Configuration sequences have been run and accepted, the Slave is ready to exchange data with the Master that parametrised and configured it (or in some cases with another Master, in accordance with the conditions set out in the Standard).**




**When the System is in the DATA_EXCHANGE_STATE (DXCHG) state, the EMA measurements can be read.**

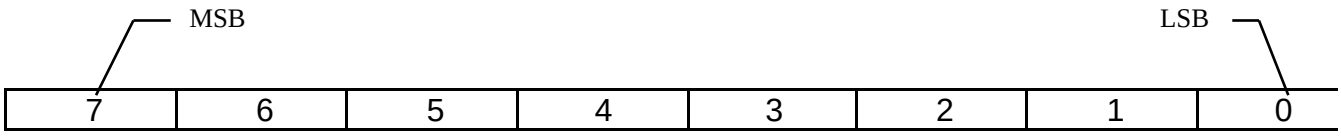
- **NB.: the order "Parametrisation first, then Configuration" MUST be followed.**

## 1.5. Slave diagnostics:

**Diagnostics codes can be read irrespective of the State in the diagram.**

### 1.5.1. Description of the first part of the Diagnostics field

**The first six bytes of the Diagnostics Message are described in chapter 8.3.1 of the Profibus DP Standard.**




**BYTE 00: STATION STATUS 1**
**Bit 7: Master_Lock (Set by the DP Master).**
**Bit 6: Prm_Fault Last Parameter Frame was faulty.**
**Bit 5: Invalid Slave Response (Set by the DP Master).**
**Bit 4: Not Supported Function requested not supported from this DP slave.**
**Bit 3: Ext_Diag A Diag. Entry exists in the Slave specific Diag area (Ext_Diag_Data).**
**Bit 2: Cfg Fault The last received configuration data from the DP-Master are different from these which the DP-Slave has determined (see Check_Cfg).**
**Bit 1: Station Not Ready This Station is not yet ready for data transfer**
**Bit 0: Station Non Existent (Set by the DP Master).**

**BYTE 01: STATION STATUS 2**
**Bit 7: Deactivated (Set by the DP Master).**
**Bit 6: (not used)**
**Bit 5: Sync Mode This station has received the Sync control command.**
**Bit 4: Freeze Mode This station has received the Freeze control command.**
**Bit 3: WD On Set as soon as the WatchDog control has been activated.**
**Bit 2:always set.**
**Bit 1: Stat Diag Set as soon as an External communication fault has been detected (in this case, the Diag LEDs. Indicate an External fault. See chap. 9). The DP-Master shall fetch diagnostic information's as long as this bit is reset again.**
**Bit 0: Prm Req This station shall be reparametrised and reconfigured (This bit has priority on bit 1).**

**BYTE 02: STATION STATUS 3 Not used here.**

**BYTE 03: MASTER ADD Address of the master which has parametrised this station.**

**BYTES 04/05: IDENT NUMBER PNO Ident number (2 bytes).**

### 1.5.2. Description of the second part of the diagnostics field (user field)

**The User field begins at address 6 of the Diagnostics field.**
**This field will contain the flags for any errors or limits exceeded on the device's various channels.**

**BYTE 06: Diag Header.**
**This byte contains the total number of bytes in the User field (including byte 6 itself). This will always be set to Byte 6 = 5, indicating that four User diagnostics bytes are to follow.**

**The Diagnostics Data in the User field are only updated when an alarm is detected (the exception to this rule being the Freeze mode).**
**Important: If Freeze mode is enabled, any diagnostics will also be frozen.**

**The Station state is given by four bytes containing the high and low alarms detected on the Voltage and Current channels.**

**These four bytes are assigned as shown below:**

**Byte 7: Undervoltage alarms. (set to 1 in alarm state)**

| Bit 7: | Bit 6: | Bit 5: | Bit 4: | Bit3: | Bit 2: | Bit1: | Bit 0: |
|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **V5** | **V4** | **V3** | **V2** | **V1** |

**Byte 8: Overvoltage alarms. (set to 1 in alarm state)**

| Bit 7: | Bit 6: | Bit 5: | Bit 4: | Bit3: | Bit 2: | Bit1: | Bit 0: |
|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **V5** | **V4** | **V3** | **V2** | **V1** |

**Byte 9: Undercurrent alarms. (set to 1 in alarm state)**

| Bit 7: | Bit 6: | Bit 5: | Bit 4: | Bit3: | Bit 2: | Bit1: | Bit 0: |
|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **I5** | **I4** | **I3** | **I2** | **I1** |

**Byte 10: Overcurrent alarms. (set to 1 in alarm state)**

| Bit 7: | Bit 6: | Bit 5: | Bit 4: | Bit3: | Bit 2: | Bit1: | Bit 0: |
|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **I5** | **I4** | **I3** | **I2** | **I1** |

### 1.6. Parametrisation:

**The parametrisation sequence is performed by the Master in two phases:**

- **System Parametrisation, using the seven bytes described in Standard EN50170;**
- **User Parametrisation, which is static, always comprising five bytes, all set to zero in normal operation. (These bytes must not be used when the device is functioning normally).**

### 1.7. Configuration:

**The device is configured using a standard Profibus DP Configurator. The configurator specifies how the Input and Output Buffers are to be structured, and how data are to be transferred to and from these buffers.**
**The Station expects between one and five configuration bytes from the Master, depending on the number of voltage/current (V/I) pairs installed.**

**The same configuration byte is always used for each V/I pair installed - 0x51. It indicates that measurements will be sent on two bytes, and naturally as read-only data (no Output buffer).**

### 1.8. Access to measurements – Data Exchange:

**Measurements are read by the Master from the Input Buffer.**
**The Output Buffer is not used in this application.**

**The Input Buffer is organised as shown in the table below:**

| Din | | |
|---|---|---|
| V/I pair 1 Always installed | V1 | 2 bytes |
| | I1 | 2 bytes |
| V/I pair 2 Always installed | V2 | 2 bytes |
| | I2 | 2 bytes |
| V/I pair 3 optional | V3 | 2 bytes |
| | I3 | 2 bytes |
| V/I pair 4 optional | V4 | 2 bytes |
| | I4 | 2 bytes |
| V/I pair 5 optional | V5 | 2 bytes |
| | I5 | 2 bytes |

**Total: Up to 20 bytes**

**In fact, each V/I pair requires four bytes, and all the bytes are stored consecutively in the input buffer in ascending order of the V/I pair numbers.**
**An E.M.A. standard interface will contain one of the following combinations:**

**Pairs V/I 1 and 2, or**
**Pairs V/I 1, 2, 3 and 4, or**
**Pairs V/I 1, 2, 3, 4 and 5. (maximum configuration).**

**The corresponding input buffers will contain 8, 16 and 20 bytes, respectively.**

### 1.9. Global Control:

**The Unit supports SYNC and FREEZE modes.**

### 1.10. Station address selection

**The address can be set via the Profibus DP Bus using the Set_Slave_Address Service.**
**It can also be sent from the Configurator running on a PC.**
**The Station is delivered factory-set with the address 32 (decimal) and a series of standard parameters and Tags to enable the unit to be started "out-of-the-box".**
**If necessary, the User should set another operating address and enter his or her own Tags and parameters.**
**Important note!: Entering address 125 resets the Tags and parameters to their default values. The new settings are implemented the next time the unit is switched on.**

### 1.11. Other Services supported

**The Unit also supports the following Services:**

**Read_Inputs**
**Read_Output**
**Get_Config**

## 1.12. Device Data Base (DDB):

```
; File Name: EURO00FF.GSD
;
; General Informations:
#Profibus_DP
Vendor_Name = "EUROTHERM Automation"
Model_Name = "EMA"
Revision = "1.0"
Ident_Number = 0x00ff
Protocol_Ident = 0                                  ; PROFIBUS DP
Station_Type = 0                                    ; DP-slave
FMS_supp = 0
Hardware_Release = "V1.0"
Software_Release = "V1.0"
9.6_supp = 1                                        ; auto
19.2_supp = 1                                       ; auto
93.75_supp = 1                                      ; auto
187.5_supp = 1                                      ; auto
500_supp = 1                                        ; auto
1.5M_supp = 1                                       ; auto
MaxTsdr_9.6 = 60                                    ; unit = tbit
MaxTsdr_19.2 = 60
MaxTsdr_93.75 = 60
MaxTsdr_187.5 = 60
MaxTsdr_500 = 100
MaxTsdr_1.5M = 150
Redundancy = 0
Repeater_Ctrl_Sig = 0
24V_Pins = 0
;
Implementation_Type = "SPC3"
Slave_Family = 5                ; control group
Bitmap_Device = "EUR00FF"
Bitmap_Diag   = "EUR00FF"
Bitmap_SF     = "EUR00FF"
;
; DP_Slave Informations:
Freeze_Mode_supp = 1
Sync_Mode_supp = 1
Auto_Baud_supp = 1
Set_Slave_Add_supp = 1
User_Prm_Data_Len = 05
User_Prm_Data = 0x00,0x00,0x00,0x00,0x00
Min_Slave_Intervall = 1                  ; 100 micro-second
Modular_Station = 1
Max_Module = 5
Max_Input_Len = 20
Max_output_Len = 0
Max_Data_Len = 20
```

```
Module = " EMA C1 " 0x51
Endmodule
Module = " EMA C2 " 0x51
Endmodule
Module = " EMA C3 " 0x51
Endmodule
Module = " EMA C4 " 0x51
Endmodule
Module = " EMA C5 " 0x51
Endmodule
```

_

# 2. The Modbus interface

## 2.1. Introduction

**For details of the notions used in these specifications, please refer to the following publications:**

- **GOULD MODBUS Protocol Reference Guide PI-MBUS-300 Rev B.**
- **EUROTHERM Group MODBUS Standard TN0998.**
- **RS485 / EIA Standard.**
- **Standard CEI 1131-3: programmable controllers - programming Languages**

## 2.2. General characteristics of the Modbus bus

**The purpose of this section is to give a brief summary of the basic characteristics of the Modbus protocol, as described in the specifications produced by Gould Modicon.**
**Modbus is not, strictly speaking, a field bus, inasmuch as it does not comply with the ISO's OSI specifications. It is generally used with industrial equipment where time is not a critical parameter.**

## 2.3. Protocol type

**This protocol also uses a Master / Slave structure. Various functions (represented by codes) are used to provide access to the values, which can be parameters, blocks, variables or diagnostics. The protocol distinguishes between logical values and analogue values. Each quantity can be accessed using its address.**
**In the E.M.A. interface, only Functions 3-4 (read *n* variables) and 8 (general diagnostics) have been implemented. All 12 variables (10 measurements and 2 diagnostics) can be read in a single transaction.**

## 2.4. Physical bus

**Two-wire RS485 bus.**
**8-bit words with no parity or with even parity.**
**1 stop bit.**

## 2.5. Maximum number of units and addressing system

**The number of units is limited by the characteristics of the RS485 Bus.**
**The medium supports up to 32 units without a repeater, including the Master. If a repeater is used, up to 247 addresses can be used.**
**Each unit's address is set at the same time as the device's other parameters, using the configurator running on a PC (W95, W98 or WNT).**

**Important note! Modifications to communication parameters, and specifically the address, are only implemented the next time the device is switched on.**

## 2.6. Baud rate

**9,600 or 19,200 Baud.**

## 2.7. Implementation

**A UART external to the CPU, manages the bus. The full Slave protocol is implemented in the software.**

## 2.8. Length

**1,200 m at 9,600 baud with no repeater.**

## 2.9. Performances

**The maximum data frame transmission rate is 19,200 Baud. A full transaction (reading all 12 values in a single block) requires 30ms on average. Naturally, the transaction will be longer if parameters are read one by one.**
**The typical polling interval is one second.**

## 2.10. Time out

**A four-second Time Out between valid unit addressing operations can either be set or disabled.**
**After the four-second time limit, the red LED on the front panel will flash. Otherwise, the green LED should remain on constantly.**

## 2.11. Configuration method:

**The Modbus communication characteristics can be set at the same time as the device's other parameters, using the configurator running on a PC (W95, W98 or WNT).**
**Important note! Modifications to communication parameters are only implemented the next time the device is switched on.**

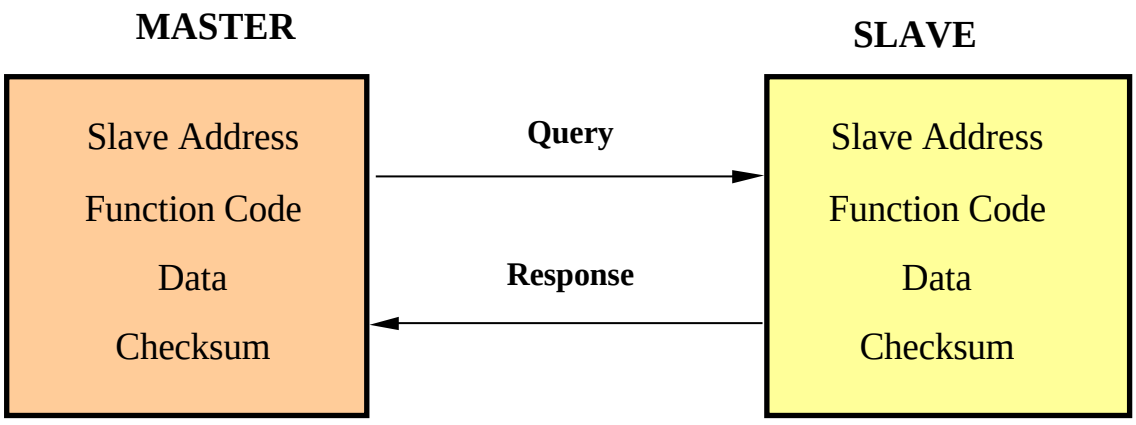
## 2.12. Use of the Modbus protocol

**Only Functions 3-4 (read *n* words) and 8 (diagnostics) have been implemented.**

### 2.12.1. Principle:

**The Modbus RTU protocol is used, i.e.:**
- **data coding: binary bits / NRZ system.**
- **Number of bits: 8 with no parity as standard, although even parity can also be set, in which case, bytes are coded on 9 bits.**
- **Start bit: 1**
- **Stop bit: 1**
- **First bit: Least significant bit first.**
- **Error tracing: CRC16**

**The protocol checks the exchanges between the master and the slaves by tracking the Master's Query then the Slave's Response.**
**This means that each exchange comprises two messages.**




**Only the Master initiates a transaction.**

- **Slave address: specifies which Slave device is concerned by the query.**
- **Function code: specifies the required operation.**
- **Data: all the parameters required by the function.**
- **Checksum: Cyclic Redundancy Code (CRC).**

### 2.12.2. Measurement read frames: Function 3

#### 2.12.2.1. Query

**Measurements and/or diagnostics are read using function 3. Function 4 can also be used, in exactly the same way.**
**The query frame has the following structure:**

| Slave address | Function code = 3 or 4 | Modbus address of the first word | Number of words to read | CRC-16 |
|---|---|---|---|---|
| (1 byte) | (1 byte) | (2 bytes) | (2 bytes) | (2 bytes) |

**The two-byte words are always written with the most significant bit first (Motorola notation).**
**The message will only be accepted by the Slave if the address matches and the CRC16 is correct.**
**With the E.M.A. interface, the number of words to be read cannot exceed 12, i.e. 10 measurement words and two diagnostics words.**

#### 2.12.2.2. Response

| Slave address | Function code = 3 or 4 | Number of bytes read | Value of n words | CRC-16 |
|---|---|---|---|---|
| (1 byte) | (1 byte) | (1 byte) | (n*2 bytes) | (2 bytes) |

### 2.12.2.3.Valid parameter address list

**The Modbus addresses for the various variables used by the E.M.A. interface are set as shown in the table below:**

| | Value | Address | Size |
|---|---|---|---|
| **Card 1 Always installed** | **V1** | **100** | **2 bytes** |
| | **I1** | **101** | **2 bytes** |
| **Card 2 optional** | **V2** | **102** | **2 bytes** |
| | **I2** | **103** | **2 bytes** |
| **Card 3 optional** | **V3** | **104** | **2 bytes** |
| | **I3** | **105** | **2 bytes** |
| **Card 4 optional** | **V4** | **106** | **2 bytes** |
| | **I4** | **107** | **2 bytes** |
| **Card 5 optional** | **V5** | **108** | **2 bytes** |
| | **I5** | **109** | **2 bytes** |
| **Diagnostics** | **Diag V** | **110** | **2 bytes** |
| | **Diag I** | **111** | **2 bytes** |
| | | **Total:** | **Up to 24 bytes** |

**The Diagnostics codes have the following structure:**
**Diag V / address 110**

**MSB bytes: Overvoltage Alarms.**

| Bit 7: | Bit 6: | Bit 5: | Bit 4: | Bit3: | Bit 2: | Bit1: | Bit 0: |
|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **V5** | **V4** | **V3** | **V2** | **V1** |

**LSB bytes: Undervoltage Alarms**

| Bit 7: | Bit 6: | Bit 5: | Bit 4: | Bit3: | Bit 2: | Bit1: | Bit 0: |
|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **V5** | **V4** | **V3** | **V2** | **V1** |

**Diag I / address 111**

**MSB bytes: Overcurrent Alarms.**

| Bit 7: | Bit 6: | Bit 5: | Bit 4: | Bit3: | Bit 2: | Bit1: | Bit 0: |
|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **I5** | **I4** | **I3** | **I2** | **I1** |

**LSB bytes: Undercurrent Alarms.**

| Bit 7: | Bit 6: | Bit 5: | Bit 4: | Bit3: | Bit 2: | Bit1: | Bit 0: |
|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **I5** | **I4** | **I3** | **I2** | **I1** |

### 2.12.2.4. Example

**In this example, the station is set with the address 50 (0x32)**
**The example reads 10 measurements plus the diagnostics bytes:**

**Query:**
**50/3/0/100/0/12/1/211**

**Response:**
**50/3/24/0/0/0/0/0/0/0/1/5/140/0/0/0/0/0/0/86/238/0/14/0/0/0/0/144/252**

**In this example, the bytes in the frame are expressed in decimal.**

### 2.12.2.5. Exchange diagnostics: Function 8

**This function is mainly implemented in order to ensure compatibility with certain masters that use it to check the presence of the slave devices on the bus.**

#### 2.12.2.5.1. Frame structure:

**Query:**

| Slave address (1 byte) | '8' (1 byte) | Sub-code '0' (2 bytes) | Data (2 bytes) | CRC (2 bytes) |
|---|---|---|---|---|

**Note:**
- **Broadcasting is not allowed.**
- **Only sub-code 00 is allowed.**
- **The value of the Data is not verified, merely resent as an echo.**

**Response:**

| Slave address (1 byte) | '8' (1 byte) | Sub-code '0' (2 bytes) | Data (2 bytes) | CRC (2 bytes) |
|---|---|---|---|---|

**Note:**
**The response is simply an echo of the query.**

### 2.12.2.6. Example:

**Query: 0x32, 0x08, 0x00, 0x00, 0x00, 0x00, 0xE5, 0xC8.**

**Response: 0x32, 0x08, 0x00, 0x00, 0x00, 0x00, 0xE5, 0xC8.**

**In this example, the bytes in the frame are expressed in hexadecimal.**

### 2.12.2.7. Error frame

**Frame structure:**

**When the Slave detects an error in the query frame sent by the Master, it responds with an error frame.**
**Error frames comprise the elements shown below.**

| Slave address (1 byte) | Function Code + 128 (1 byte) | Exception code (1 byte) | CRC-16 (2 bytes) |
|---|---|---|---|

**The following exception codes are used:**

| CODE | ERROR |
|---|---|
| 1 | Illegal function |
| 2 | Illegal data address |
| 3 | Illegal data address |
| 4 | Device failure |
| 5 | ACK |
| 6 | Busy |
| 7 | NACK |
| 8 | No data at specified address |
| 9 | No data in query |
| 10 | Too many data in query |

**NB.: these are the standard exception codes. They are not all used with the EMA interface.**

**Example:**

**In the example below, too many variables have been requested:**

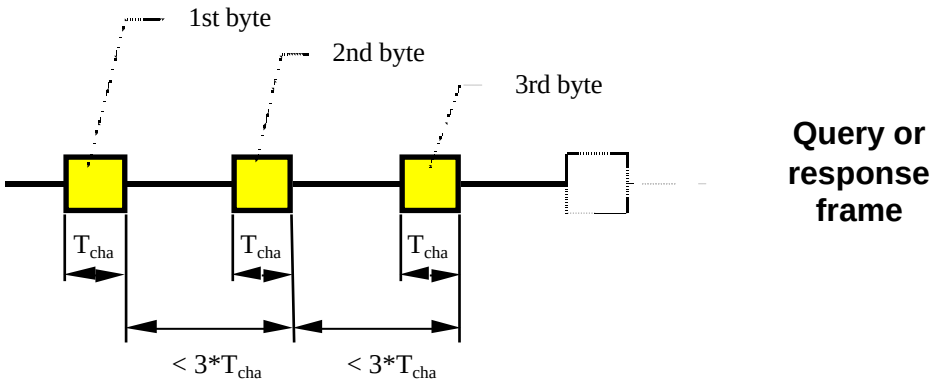
**Query:**
**50/3/0/102/0/12/160/19**

**Response:**
**50/131/10/49/56**

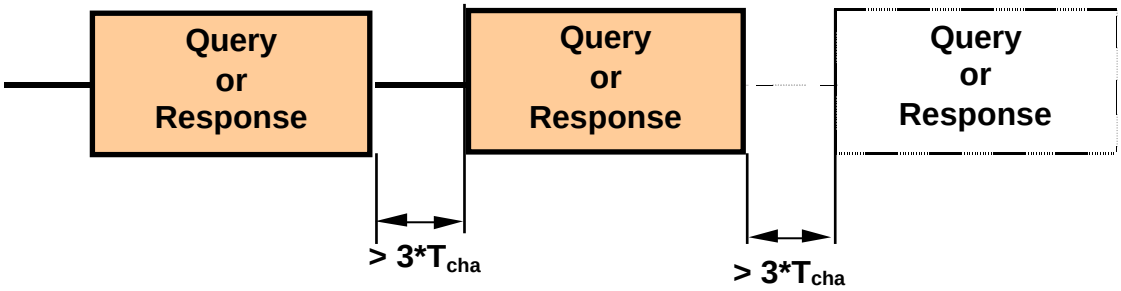
### 2.12.2.8. Frame synchronisation:

**Frames are synchronised using the time elapsed between two consecutive frames. The time between two consecutive characters in a given frame must not exceed the "width" of three characters.**

**Within a frame**




**Between two frames**




**The elapsed time between two consecutive frames must exceed the length of three characters.**

## 3. Troubleshooting using the LEDs on the front panel

**There are two LEDs on the device's front panel:**

- **1 GREEN LED**
- **1 RED LED**

- **Green LED**
  **In Profibus mode, this LED indicates that the unit is in Data Exchange State.**
  **In Modbus mode, it indicates that the unit has received a valid Modbus frame in the last four seconds. This Time Out can be disabled, in which case, the green LED will remain on constantly, once the first valid frame has been received.**

- **Red LED**

**If the red LED is lit (ON steady), a serious error has been detected.**

**The LED will flash (1.2s ON / 1.2s OFF) if the unit is unable to communicate with its Master.**
**Either the Master has never addressed the Slave, or else the Watchdog Time-Out was triggered before the unit received a valid frame.**

**In normal operation, this LED should not be lit, meaning that the Parametrisation and Configuration procedures have been performed correctly (in Profibus mode) and that the device can communicate with its Master in Data Exchange state.**
**If this is not the case, begin by checking the following points:**

**⇒ Profibus Parametrisation sequence;**
**⇒ Profibus Configuration sequence.**
**Then check:**
**⇒ Connections;**
**⇒ Bus cables;**
**⇒ Bus length;**
**⇒ Impedance match;**
**⇒ Address (notably check that no other Slaves on the bus have the same address);**
**⇒ Check that the Watchdog Time-Out is not too short for the Master's polling baud rate.**

**If neither the green nor red LEDs are lit, there is probably no power supply to the unit.**

### 3.1. Indication of the communication state on the display

- **Profibus mode**
-

**The Display constantly shows the Profibus communication state (Cf. State Machine).**
**One of the following states will be displayed:**
- **WPRM** ⇐ **Wait for Parametrisation;**
- **WCFG** ⇐ **Wait for Configuration;**
- **DXCG** ⇐ **Data Exchange.**

- **Modbus mode**
-

**The display constantly shows "MODBUS".**

**This does not appear on the first page.**

##############################