

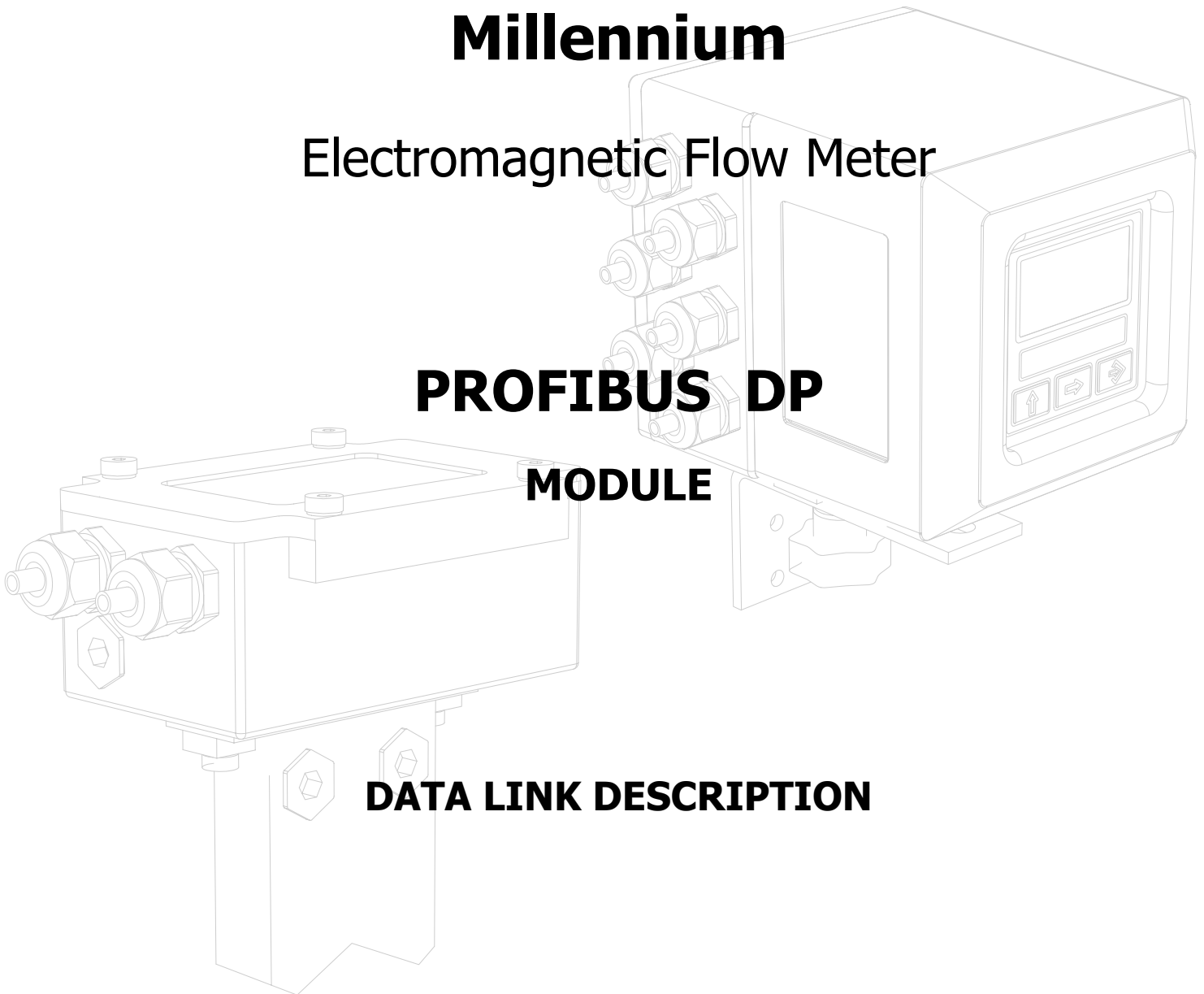
# Millennium

Electromagnetic Flow Meter

## PROFIBUS DP

### MODULE

### DATA LINK DESCRIPTION

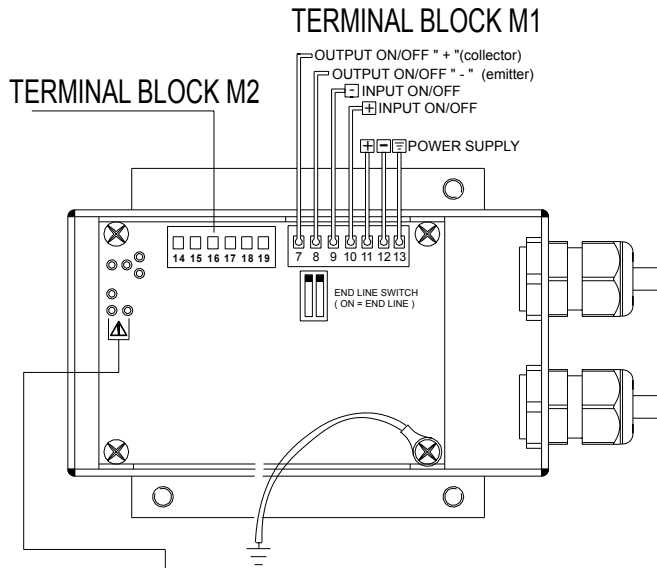


# INDEX

<b>1.1 CHARACTERISTICS OF THE MODULES</b>	3
<b>1.2. PROFIBUS IDENT NUMBER</b>	4
<b>1.3. SLAVE PARAMETERIZATION</b>	4
<b>1.4. SLAVE CONFIGURATION</b>	5
<b>1.5. SLAVE DIAGNOSTIC AND ALARMS</b>	10
<b>1.6. SETTING THE SLAVE ADDRESS</b>	12
<b>1.7. BAUDRATES</b>	12
<b>1.8. GSD FILE</b>	12
<b>1.9. TERMINALS DESIGNATION AND CABLE CONNECTION</b>	12
<b>2. CYCLIC DATA EXCHANGE</b>	13
<b>2.1. FIRST CONFIGURATION: 8 BYTE INPUT</b>	13
<b>2.2. DATA TYPE</b>	14
<b>2.3. SECOND CONFIGURATION: 16 BYTE INPUT</b>	14
<b>2.4. DATA TYPE</b>	15
<b>2.5. THIRD CONFIGURATION: 24 BYTE INPUT</b>	16
<b>2.6. DATA TYPE</b>	17
<b>2.7. FOURTH CONFIGURATION: 16 BYTE INPUT/OUTPUT</b>	17
<b>2.8. FIFTH CONFIGURATION: 24 BYTE INPUT/OUTPUT</b>	28
<b>3.1 DPV1 ACYCLIC DATA EXCHANGE</b>	42
<b>3.2 Acyclic Read Data from the Flow Meter</b>	42
<b>4.1 PROGRAMMING ALGORITHMS</b>	50
<b>4.2 NUMERICAL CONVERSION OF THE DATA RECEIVED</b>	50
<b>4.3 DATE CONVERSION FROM TOTAL MINUTES TO DAY/MONTH/YEAR HOURS:MINUTES</b>	51
<b>4.4 DATE CONVERSION FROM DAY/MONTH/YEAR HOURS:MINUTES TO TOTAL MINUTES</b>	52

**1.1 CHARACTERISTICS OF THE MODULES**

**ME 34 ( profibus DP for ML3F1 converter )**

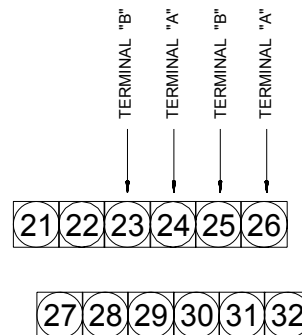
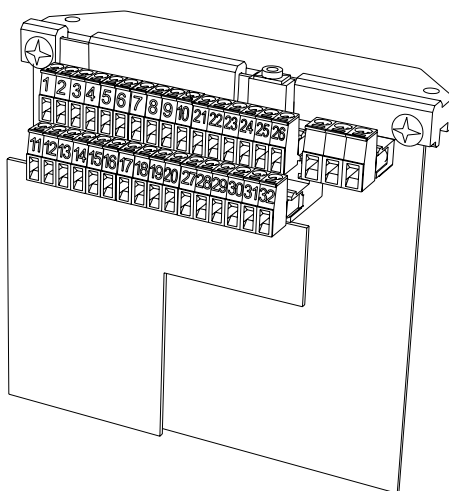



**M2 :**

- 14** = Terminal A PROFIBUS DP
- 15** = Terminal B PROFIBUS DP
- 16** = Shield
- 17** = Terminal - of Power supply
- 18** = Terminal + of Power supply
- 19** = No used

Dangerous voltage:  
 - 60 Vdc max  
 - 250 V max commutation coil's circuit

**ME 100 ( profibus DP for ML 210 converter )**



	TECHNICAL MANUAL		Rev.	Page	4/52
	PROFIBUS DP MODULE- DATA LINK DESCRIPTION		03	Filling By	Checked By
			Document Data	Date	Date
	File name	ProfibusDP_emanual_Burkert_02.doc	20/07/04	12-01-07	12-01-07

The Profibus-DP module mounted in the Electromagnetic Flow Meter is used for to connect the converter of the Flow meter to the Profibus DP fieldbus in accordance with the DIN 19245 Standard as a passive instrument (slave).

When this option is installed it is possible to read the process data from the Flow meter.

The process data consists of **Dynamic Data** and **Static Data**.

With the keyword **Dynamic Data** it is indicated measurement values which are continuously changing e.g. Flow rate, Totalizers, etc..

With the keyword **Static Data** it is indicated parameters value of the Flow meter which do not change during operation for example unit Flow rate, unit Totalizers etc. and other parameters that can be individually written to using the **Acyclic Service** or an offset to **INDEX Output** (see Fourth Configuration with 16 byte **Input/Output** or Fifth Configuration with 24 byte **Input/Output**).

The software of the slave module implement the DP standard/DPV1 functionality for the communication of the slave with a Class 1 (parameterization Master) and a Class 2 Master.

With the data exchange with the **Acyclic Service** and with the **Slot** and **Acyclic Index** it is possible to send and return configuration parameters of the Flow meter with a Class 1 or Class 2 Master.

For the communication with a Class C1 Master the **Slot** is ignored.

Only **Slot 0** is used for the service of acyclic data exchange for the communication with a Class 2 master.

The module support four type of configurations with is possible sets the better combination of type of information of the Flow meter.

It is possible select a configuration with only 8 byte or 16 or 24 byte **Input** and insert or return **Static Data** with the **Acyclic Service** and Class 1 or Class 2 Master.

If the Master doesn't support **Acyclic Service** it is possible select the fourth configuration with 16 byte **Input/Output** or the fifth configuration with 24 byte **Input/Output**. With these last configuration and with two bytes, with the function of indexes, and the **Output** data, it is possible to insert or return **Static Data** in the Flow meter.

With the **Acyclic Service** and with the **Slot** and **Acyclic Index** it is possible to read the data stored in the Internal Data Logger of the Fow Meter.

The Profibus address of the instrument can be set with the converter keypad in the menu Comunication.

Baudrates range supported are from 9.6 kBaud up to 12 MBaud.

The slave module support the Diagnostic and Process Alarms for monitoring Flow rate with four thresholds.

**NOTE:** After the power supply of the Flow meter is switch on, there is an interval time of **5 seconds** during that the slave module remain in a Reset Phase necessary for the initialization of the Flow meter.

The reading of the data stored in the Internal Data Logger and the settings for the thresholds of the Flow rate Control are possible only with the **Acyclic Service** and with the **Slot** and **Acyclic Index**.

## **1.2. PROFIBUS IDENT NUMBER**

The Profibus Identification Number is 0008hex ( 8dec ).

## **1.3. SLAVE PARAMETERIZATION**

The parameters specified in this respect reach the slave via the parameterization message of the parameterization Master.

The parameterization data consists of the following:

- DP standard parameters with a length of 7 bytes according to EN 50170
- DPV1 status bytes with a length of 3 bytes according to extensions of EN 50170 which immediately follow the specified DP standard parameters

- Optional user parameterization data that immediately follow the DPV1 status bytes ( not implemented for the this slave module ).  
Default Parameterization:

Default parameterization makes it possible for old DP standard masters to exchange data with the DPV1 slave despite missing DPV1 status bytes.  
The parameterization message is 7 bytes long and in this case, the slave will use a set of default values for the missing DPV1 status bytes.

**SETTINGS OF THE STATION PARAMETER:**

DP Interrupt Mode DPV0  
 DPV1 Interrupts:  
 Update Interrupt (OB56) No  
 Diagnostic Interrupt(OB82) No  
 Hardware Interrupt(OB40 to OB 47) No  
 General DP Parameters:  
 Fail Safe Yes  
 Startup when expected/actual config diff. No  
 Device-Specific Parameters:  
 DPV1 Mode Enabled  
 Process Alarm Disabled  
 Diagnostic Alarm Disabled  
 Update Alarm Disabled  
 Alarm Type Type mode (Slave is operated alarm type mode: only one alarm of the same type is to be active at the same time)  
 Hex Parameter Assignment:  
 DPV1\_Status (0 to 2) E4,00,00

**1.4. SLAVE CONFIGURATION**

The Configuration possibilities are specified for the Slave module in the device description file (GSD).  
Only a single Configuration is available.

The module support four type of configurations with is possible sets the better combination of types of informations of the Flow meter.

**The First type of Configuration consist of 8 byte Input**

With this type of configuration is possible exchange with few byte ( only 8 ), the main data of the Flow meter.

With the setting of a **control byte**, in the configuration string of the file GSD,it is possible to set the **Data Type** of information returned by **Input** data.

The **control byte** is the third byte of the Configuration string (see below).

It is also possible to change the **Data Type** with a Class 1 and Class 2 Master and the relative **Slot** and **Acyclic Index**, during data exchange.

For details **Acyclic Service** see the relative section in this document.

The reference to the Configuration string in the GSD file is:

-----  
 Module = "8 Byte Input" 0x41,0x07,0x00

1

EndModule-----

The settings in the GSD file for to use this configuration is:

-----  
SlotDefinition

**Slot(1) = "Process Data" 1**

EndSlotDefinition  
-----

The Configuration correspond to 8 bytes Input and the last byte indicate the **Data Type** selected .

**Modifying** the last byte in the GSD file it is possible to set the type of the Flow rate:

With the Configuration 0x41,0x0F,**0x00** **the Flow rate is in % ( 4 byte )**  
 process flags ( 2 byte )  
 dynamic variations ( 1 byte )  
 Data Type = 0 ( 1 byte )

-----  
Number of Input byte exchanged ( 8 byte )

With the Configuration 0x41,0x0F,**0x01** **the Flow rate is in technical unit ( 4 byte )**  
 process flags ( 2 byte )  
 dynamic variations ( 1 byte )  
 Data Type = 1 ( 1 byte )

-----  
Number of Input byte exchanged ( 8 byte )

With the Configuration 0x41,0x0F,**0x02** **totalizers positive ( 4 byte )**  
 process flags ( 2 byte )  
 dynamic variations ( 1 byte )  
 Data Type = 1 ( 1 byte )

-----  
Number of Input byte exchanged ( 8 byte )

With the Configuration 0x41,0x0F,**0x03** **partial totalizers positive ( 4 byte )**  
 process flags ( 2 byte )  
 dynamic variations ( 1 byte )  
 Data Type = 1 ( 1 byte )

-----  
Number of Input byte exchanged ( 8 byte )

With the Configuration 0x41,0x0F,**0x04** **totalizers negative ( 4 byte )**  
 process flags ( 2 byte )  
 dynamic variations ( 1 byte )  
 Data Type = 1 ( 1 byte )

-----  
Number of Input byte exchanged ( 8 byte )

With the Configuration 0x41,0x0F,**0x05** **partial totalizers negative ( 4 byte )**  
 process flags ( 2 byte )  
 dynamic variations ( 1 byte )  
 Data Type = 1 ( 1 byte )

-----  
Number of Input byte exchanged ( 8 byte )

The **Second** type of Configuration consist of **16 byte Input**

With this type of Configuration it is possible exchange with only 16 the same data of the precedent Configuration and in addition there are the values of the Totalizers positive or negative.

With the setting of a **control byte**, in the configuration string of the file GSD,it is possible to set the **Data Type** of information returned by **Input** data.

The **control byte** is the third byte of the Configuration string (see below).

It is also possible to change the **Data Type** with a Class 1 and Class 2 Master and the relative **Acyclic Slot** and **Acyclic Index**, during data exchange.

For details **Acyclic Service** see the relative section in this document.

The reference to the Configuration string in the GSD file is:

```
-----
Module = "16 Byte Input" 0x41,0x0F,0x00
2
EndModule
-----
```

The settings in the GSD file for to use this Configuration is:

```
-----
SlotDefinition
Slot(1) = "Process Data" 2
EndSlotDefinition
-----
```

The Configuration correspond to 16 bytes Input and the last byte indicate the **Data Type** selected .

**Modifying** the last byte in the GSD file it is possible to set the type of the Flow rate and the type of Totalizers:

With the Configuration 0x41,0x0F,**0x00** the Flow rate is in % ( 4 byte )

<b>totalizers positive</b>	( 4 byte )
<b>partial totalizers positive</b>	( 4 byte )
process flags	( 2 byte )
dynamic variations	( 1 byte )
the Data Type = 0	( 1 byte )

-----

Number of Input byte exchanged ( 16 byte )

With the Configuration 0x41,0x0F,**0x01** the Flow rate is in % ( 4 byte )

<b>totalizers negative</b>	( 4 byte )
<b>partial totalizers negative</b>	( 4 byte )
process flags	( 2 byte )
dynamic variations	( 1 byte )
the Data Type = 1	( 1 byte )

-----

Number of Input byte exchanged ( 16 byte )

Whit the Configuration 0x41,0x0F,**0x02** the Flow rate is in tecnicl unit ( 4 byte )

<b>totalizers positive</b>	( 4 byte )
<b>partial totalizers positive</b>	( 4 byte )
process flags	( 2 byte )
dynamic variations	( 1 byte )

the Data Type = 2 ( 1 byte )

-----  
Number of Input byte exchanged ( 16 byte )

Whit the Configuration 0x41,0x0F,**0x03** **the Flow rate is in tecnical unit ( 4 byte )**  
**totalizers negative ( 4 byte )**  
**partial totalizers negative ( 4 byte )**  
 process flags ( 2 byte )  
 dynamic variations ( 1 byte )  
 the Data Type = 3 ( 1 byte )

-----  
Number of Input byte exchanged ( 16 byte )

Whit the Configuration 0x41,0x0F,**0x04** **the Flow rate is in % ( 4 byte )**  
**totalizers positive ( 4 byte )**  
**totalizers negative ( 4 byte )**  
 process flags ( 2 byte )  
 dynamic variations ( 1 byte )  
 the Data Type = 3 ( 1 byte )

-----  
Number of Input byte exchanged ( 16 byte )

Whit the Configuration 0x41,0x0F,**0x05** **the Flow rate is in tecnical unit ( 4 byte )**  
**totalizers positive ( 4 byte )**  
**totalizers negative ( 4 byte )**  
 process flags ( 2 byte )  
 dynamic variations ( 1 byte )  
 the Data Type = 3 ( 1 byte )

-----  
Number of Input byte exchanged ( 16 byte )

**The third type of Configuration consist of 24 byte Input**

This configuration is used when there is necessary to exchange the information about totalizers positive and negative at the same time.

The difference with the precedent configurations is the presence of the totalizers positive and negative contemporary in the block of data exchanged

With the setting of a **control byte**, in the configuration string of the file GSD,it is possible to set the **Data Type** of information returned by **Input** data.

The **control byte** is the third byte of the Configuration string (see below).

It is also possible to change the **Data Type** with a Class 1 and Class 2 Master and the relative **Acyclic Slot** and **Acyclic Index**, during data exchange.

For details **Acyclic Service** see the relative section in this document.

The reference to the Configuration string in the GSD file is:

-----  
Module = "24 Byte Input" 0x41,0x17,0x00  
3  
EndModule  
-----

The settings in the GSD file for to use this configuration is:

-----  
SlotDefinition



**Slot(1) = "Process Data" 3**

EndSlotDefinition

The Configuration correspond to 24 bytes Input and the last byte indicate the **Data Type** selected .

Changing the last byte in the GSD file it is possible to set the type of the Flow rate:

With the Configuration 0x41,0x0F,**0x00** the **Flow rate is in % ( 4 byte )**

totalizers positive	( 4 byte )
partial totalizers positive	( 4 byte )
totalizers negative	( 4 byte )
partial totalizers negative	( 4 byte )
process flags	( 2 byte )
dynamic variations	( 1 byte )
the Data Type = 0	( 1 byte )

---

Number of Input byte exchanged ( 24 byte )

With the Configuration 0x41,0x0F,**0x01** the **Flow rate is in technical unit ( 4 byte )**

totalizers positive	( 4 byte )
partial totalizers positive	( 4 byte )
totalizers negative	( 4 byte )
partial totalizers negative	( 4 byte )
process flags	( 2 byte )
dynamic variations	( 1 byte )
the Data Type = 1	( 1 byte )

---

Number of Input byte exchanged ( 24 byte )

**The fourth type of configuration consist of 16 byte Input/Output**

With this type of configuration is possible to insert or return data from the Flow meter with two indexes exchanged with the 16 byte **Output** data.

With the 16 byte **Output** data it is possible to insert other parameters of the Flow meter in the same way used with the **Acyclic Service** and Class 1 and Class 2 Master.

The reference to the Configuration string in the GSD file is:

```
Module = "16 Byte Input/Output" 0xC0,0x0F,0x0F
4
EndModule
```

The settings in the GSD file for to use this configuration is:

```
SlotDefinition
Slot(1) = "Process Data" 4
EndSlotDefinition
```

The Configuration correspond to 16 bytes **Input/Output**.

For details about the type of the data see the Section 2 **Cyclic Service** below in this document.

**The fifth type of configuration consist of 24 byte Input/Output**

With this type of configuration is possible to insert or return data from the Flow meter with two indexes

exchanged with the 24 byte **Output** data.

With the 24 byte **Output** data it is possible to insert other parameters of the Flow meter in the same way used with the **Acyclic Service** and Class 1 and Class 2 Master.

The reference to the Configuration string in the GSD file is:

```
-----
Module = "24 Byte Input/Output" 0xC0,0x17,0x17
5
EndModule
-----
```

The settings in the GSD file for to use this configuration is:

```
-----
SlotDefinition
Slot(1) = "Process Data" 5
EndSlotDefinition
-----
```

The Configuration correspond to 24 bytes **Input/Output**.

For details about the type of the data see the Section 2 **Cyclic Service** below in this document.

**NOTE:** Any other configuration will not be accepted.

### **1.5. SLAVE DIAGNOSTIC AND ALARMS**

The module support the Diagnostic and Process Alarms for monitoring Flow rate with four thresholds.

With the **Acyclic Service** and a Class 1 or Class 2 Master and the relative **Acyclic Slot** and **Acyclic Index** it is possible insert four thresholds for monitoring the Flow rate.

The thresholds are:

- Flow rate High Alarm
- Flow rate High Warning
- Flow rate Low Warning
- Flow rate Low Alarm

#### **The control is implemented for the Flow rate in technical unit only.**

After the insertion of the four thresholds of the Flow rate in **technical unit**, with the **Acyclic Service** and the relative **Acyclic Slot** and **Acyclic Index**, it is necessary to enable the control of the Flow rate.

With the **Acyclic Service** and the relative **Acyclic Slot** and **Acyclic Index** it is possible to enable the control of the Flow rate and if the value of the Flow rate measured by the Flow meter, exceed one of the thresholds, a diagnostic message in sended to the master.

With the **Acyclic Service** and the realtive **Acyclic Slot** and **Acyclic Index** it is possible to enable the DPV1 Process Alarms for the masters that support this information.

The Process Alarms are set when the value of the Flow rate measured by the Flow meter exceed one of the two thresholds High Alarm or Low Alarm.

Diagnostic message in details:

The standard diagnostic consists of 6 bytes and in addition, the alarm mechanism with extensions is used.

Structure of the Standard Diagnostic:

- byte 0: station\_status\_1
- byte 1: station\_status\_2
- byte 2: station\_status\_3
- byte 3: master\_adress
- byte 4: pno\_ident\_number\_high
- byte 5: pno\_ident\_number\_low

The implemented diagnostic concept supports the Device-related diagnostic (primarily status messages)

According to the diagnostic message of the DPV1 slave consists of the following:

- byte 6: sing\_len
- byte 7: alarm\_type
- byte 8: slot\_number
- byte 9: specifier
- byte 10: **alarm\_data**
- byte 11: 0x00

Details of the information in the byte **alarm\_data**:

- alarm\_data = 0x00 --> Flow rate Normal
- alarm\_data = 0x01 --> Flow rate High Alarm
- alarm\_data = 0x02 --> Flow rate High Warning
- alarm\_data = 0x03 --> Flow rate Low Warning
- alarm\_data = 0x04 --> Flow rate Low Alarm

---

Max Flow Rate value

```

////////////////////////////////////
////////////////////////////////////
//////////////////////////////////// alarm_data = 0x01 //////////////////////////////////
////////////////////////////////////
////////////////////////////////////

```

High Alarm Threshold value

```

////////////////////////////////////
////////////////////////////////////
//////////////////////////////////// alarm_data = 0x02 //////////////////////////////////
////////////////////////////////////
////////////////////////////////////

```

High Warning Threshold value

```

////////////////////////////////////
////////////////////////////////////
//////////////////////////////////// alarm_data = 0x00 //////////////////////////////////
////////////////////////////////////
////////////////////////////////////

```

Low Warning Threshold value

```

////////////////////////////////////
////////////////////////////////////
//////////////////////////////////// alarm_data = 0x03 //////////////////////////////////
////////////////////////////////////
////////////////////////////////////

```

Low Alarm Threshold value

```

////////////////////////////////////

```

TECHNICAL MANUAL		Rev.	Page	12/52
PROFIBUS DP MODULE- DATA LINK DESCRIPTION		03	Filling By	Checked By
		Document Data	Date	Date
File name	ProfibusDP_emanual_Burkert_02.doc	20/07/04	12-01-07	12-01-07

```

////////////////////////////////////
alarm_data = 0x04
////////////////////////////////////
Min Flow Rate value

```

### **1.6. SETTING THE SLAVE ADDRESS**

The Profibus address of the slave can be set with the converter keypad and the menu

7 - Communication --> Address

For the changing of address it's not necessary to switch off the Flow meter by removing the power supply, because the module detect when the address is changed it set the function Autoreset. After the time of 5 seconds the slave module return in data exchange with the new address.

**NOTE:** For reasons of compatibility the menu Address of the Profibus is the same used for inserting the address of the RS-485 communication channel of the Flow meter that can accept values in the range 000-255, but the valid range for the Profibus Address is only 001, 002...126.

**NOTE:** The menu Communication is visible in the screen of the Flow meter only if the parameter RS 485 is set ON in the menu 11-Internal Data.

### **1.7. BAUDRATES**

The Baudrates supported are:

- 9.6 kBaud
- 19.2 kBaud
- 31.25 kBaud
- 45.45 kBaud
- 93.75 kBaud
- 187.5 kBaud
- 500 kBaud
- 1.5 MBaud
- 3 MBaud
- 6 MBaud
- 12 MBaud

The baudrate is automatically recognized by the slave module.

### **1.8. GSD FILE**

The name of the GSD file is ME10034.GSD is included on the shipment packaging .

### **1.9. TERMINALS DESIGNATION AND CABLE CONNECTION**

For the connection of the Profibus cable to the Flow meter see the section 1.1 Characteristics of the module.

Cable specification:

All cables used must meet the following parameters according to the PROFIBUS specification for copper signal cable data wires.

The connectors must allow these connections.

**Parameter**

**DP, Cable type A**

Surge impedance in $\Omega$	135...165 for a frequency of 3...20 MHz
Effective capacitance (pF/m)	$\leq 30$
Loop resistance ( $\Omega$ /km)	$\leq 110$
Core design (solid)	AWG 22/1
Core design (flexible)	$> 0,32 \text{ mm}^2$

The table shows the cable parameters for PROFIBUS copper signal cables with RS-485 transmission standard.

**2. CYCLIC DATA EXCHANGE**

With the **Cyclic Service** of data exchange the slave return **Dynamic Data** that include measurement values which are continuously changing e.g. Flow rate, totalizers, etc..

The module support five type of configurations with is possible sets the better combination of type of information of the Flow meter.

The type of Configuration is set in the file GSD.

Below there are the description of the possible types of configuration with the relative structure of data:

**2.1. FIRST CONFIGURATION: 8 BYTE INPUT**

Below there is a description of the contents of each byte:

bytes 0-3:

(32 bit single precision IEEE floating point, MSB first) Flow rate in % or Flow rate in t.u. or  
 (32 bit long integer, MSB first) totalizer for TOTAL or PARTIAL volume + or - or batch quantity  
 (see section 2.2 Data Type)


bytes 4-5: (16 bit unsigned integer, MSB first) process flags:

- bit 0 =1 if the excitation is too fast for the sensor connected
- bit 1 =1 if the maximum alarm is active
- bit 2 =1 if the minimum alarm is active
- bit 3 =1 if the Flow rate exceeds the scale range value (overflow)
- bit 4 =1 if one or more output impulses are saturated (too many impulses to emit)
- bit 5 =1 if the measurement signal is highly disturbed or if the sensor is disconnected
- bit 6 =1 if the measurement tube is empty
- bit 7 =1 if the circuit powering the coils is not working or the sensor is disconnected
- bit 8 =1 if the second measurement scale is active
- bit 9 =1 if the Flow rate is lower than the cut-off threshold
- bit10=1 if the Flow rate is negative
- bit11=1 if a new measurement value calculated for the display is available
- bit12=1 if the counter block signal is active
- bit13=1 if dosing is in progress
- bit14=1 if a calibration cycle is in progress
- bit15=1 if a Flow rate simulation is in progress

byte 6: (8 bits integer) measurement dynamic variation as a %

byte 7: (8 bits integer) Data Type ( see section 2.2 Data Type )

**NOTE:** The values in the "32 bits single precision IEEE floating point" format are floating point numbers which can be represented during writing by any decimal digits. To keep the same numerical format visible

	TECHNICAL MANUAL		Rev.	Page	14/52
	PROFIBUS DP MODULE- DATA LINK DESCRIPTION		03	Filling By	Checked By
			Document Data	Date	Date
	File name	ProfibusDP_emanual_Burkert_02.doc	20/07/04	12-01-07	12-01-07

on the instrument display however, it is necessary to calculate the decimal figures with a rather complex algorithm which takes account of instrument precision, flow rate measurement unit, etc. For this purpose and to avoid useless calculations, the number of decimals to use to represent the flow rate values is supplied separately. With the acyclic service it is possible to return the value ( see **Acyclic Index Read = 20(dec) Static Process Data of the Flow Meter** byte position 13).

**NOTE:**The counters are expressed with a 32 bit integer. The «counter decimal figures» parameter, indicates the point position starting from the right: 0 = no decimal, 1=1 decimal figure, and so on. ( see **Acyclic Index Read = 20(dec) Static Process Data of the Flow Meter** byte position 12)

## **2.2. DATA TYPE**

The firsts four byte of the **Dynamic Data** Block are the information about the Flow rate or Totalizers.

With the **Control byte** of the Configuration in the GSD file (see the Section Slave Configuration) it is possible to set the type of the Flow rate returned by the slave or the type of the Totalizers. The type of the Flow rate may be in % or in technical unit in base of the value of the **Control byte** inserted in the Configuration string.

The byte 7 indicate the current type of Flow rate selected or the type of the totalizers:

- byte 7 = 0 --> Flow rate in %
- byte 7 = 1 --> Flow rate in technical unit
- byte 7 = 2 --> totalizers positive or batch quantity ( batch quantity only for ML M3F )
- byte 7 = 3 --> partial totalizers positive or batch counter ( batch counter only for ML M3F )
- byte 7 = 4 --> totalizers negative or batch quantity ( data valid only for ML 210 )
- byte 7 = 5 --> partial totalizers negative or batch counter ( data valid only for ML 210 )

There is a second way to set the type of the Flow rate using the **Acyclic Service** during data exchange (see section **Acyclic Service**).

## **2.3. SECOND CONFIGURATION: 16 BYTE INPUT**

Below there is a description of the contents of each byte:

bytes 0-3: (32 bit single precision IEEE floating point, MSB first) Flow rate in % or Flow rate in t.u. (see section 2.4 Data Type)

bytes 4-7: (32 bit long integer, MSB first) totalizer for TOTAL volume + or - or batch quantity (see section 2.4 Data Type)

bytes 8-11: (32 bit long integer, MSB first) totalizer for TOTAL or PARTIAL volume + or - or batch counter (see section 2.4 Data Type)

bytes 12-13: (16 bit unsigned integer, MSB first) process flags:

- bit 0 =1 if the excitation is too fast for the sensor connected
- bit 1 =1 if the maximum alarm is active
- bit 2 =1 if the minimum alarm is active
- bit 3 =1 if the Flow rate exceeds the scale range value (overflow)
- bit 4 =1 if one or more output impulses are saturated (too many impulses to emit)
- bit 5 =1 if the measurement signal is highly disturbed or if the sensor is disconnected
- bit 6 =1 if the measurement tube is empty
- bit 7 =1 if the circuit powering the coils is not working or the sensor is disconnected
- bit 8 =1 if the second measurement scale is active
- bit 9 =1 if the Flow rate is lower than the cut-off threshold
- bit10=1 if the Flow rate is negative

bit11=1 if a new measurement value calculated for the display is available  
 bit12=1 if the counter block signal is active  
 bit13=1 if dosing is in progress  
 bit14=1 if a calibration cycle is in progress  
 bit15=1 if a Flow rate simulation is in progress

byte 14: (8 bits integer) measurement dynamic variation as a %

byte 15: (8 bits integer) Data Type ( see section 2.4 Data Type )

**NOTE:** The values in the "32 bits single precision IEEE floating point" format are floating point numbers which can be represented during writing by any decimal digits. To keep the same numerical format visible on the instrument display however, it is necessary to calculate the decimal figures with a rather complex algorithm which takes account of instrument precision, flow rate measurement unit, etc. For this purpose and to avoid useless calculations, the number of decimals to use to represent the flow rate values is supplied separately. With the acyclic service it is possible to return the value ( see **Acyclic Index Read = 20(dec) Static Process Data of the Flow Meter** byte position 13).

**NOTE:**The counters are expressed with a 32 bit integer. The «counter decimal figures» parameter, indicates the point position starting from the right: 0 = no decimal, 1=1 decimal figure, and so on. ( see **Acyclic Index Read = 20(dec) Static Process Data of the Flow Meter** byte position 12)

## **2.4. DATA TYPE**

The firsts four byte of the **Dynamic Data** Block are the information about the Flow rate. After these bytes there are eight byte for the totalizers.

With the **Control byte** of the Configuration (see the Section Slave Configuration) it is possible to set the type of the Flow rate returned by the slave and the type of totalizers.

There are six possible combination of type of data in base of the value inserted in the **Control byte** of the Configuration string in the GSD file.

The byte 15 indicate the current type of data selected:

- byte 15 = 0 --> Flow rate in %  
totalizers positive or batch quantity ( batch quantity only for ML M3F )  
partial totalizers positive or batch counter ( batch counter only for ML M3F )
- byte 15 = 1 --> Flow rate in %  
totalizers negative or batch quantity ( data valid only for ML 210 )  
partial totalizers negative or batch counter ( data valid only for ML 210 )
- byte 15 = 2 --> Flow rate in technical unit  
totalizers positive or batch quantity ( batch quantity only for ML M3F )  
partial totalizers positive or batch counter ( batch counter only for ML M3F )
- byte 15 = 3 --> Flow rate in technical unit  
totalizers negative or batch quantity ( data valid only for ML 210 )  
partial totalizers negative or batch counter ( data valid only for ML 210 )
- byte 15 = 4 --> Flow rate in %  
totalizers positive or batch quantity ( batch quantity only for ML M3F )  
totalizers negative or batch quantity ( data valid only for ML 210 )

byte 15 = 5 --> Flow rate in technical unit  
 totalizers positive or batch quantity ( batch quantity only for ML M3F )  
 totalizers negative or batch quantity ( data valid only for ML 210 )

For details see the section Slave Configuration.

There is a second way to set the type of the Flow rate using the **Acyclic Service** during data exchange (see section **Acyclic Service**).

**2.5. THIRD CONFIGURATION: 24 BYTE INPUT**

Below there is a description of the contents of each byte:

bytes 0-3: (32 bit single precision IEEE floating point, MSB first) Flow rate in % or Flow rate in t.u. (see section 2.6 Flow Rate Type)

bytes 4-7: (32 bit long integer, MSB first) totalizer for TOTAL volume + or batch quantity ( batch quantity only for ML M3F )

bytes 8-11: (32 bit long integer, MSB first) totalizer for PARTIAL volume + or batch counter ( batch counter only for ML M3F )

bytes 12-15: (32 bit long integer, MSB first) totalizer for TOTAL volume - or batch quantity ( data valid only for ML 210 )

bytes 16-19: (32 bit long integer, MSB first) totalizer for PARTIAL volume - or batch counter ( data valid only for ML 210 )

bytes 20-21: (16 bit unsigned integer, MSB first) process flags:


- bit 0 =1 if the excitation is too fast for the sensor connected
- bit 1 =1 if the maximum alarm is active
- bit 2 =1 if the minimum alarm is active
- bit 3 =1 if the Flow rate exceeds the scale range value (overflow)
- bit 4 =1 if one or more output impulses are saturated (too many impulses to emit)
- bit 5 =1 if the measurement signal is highly disturbed or if the sensor is disconnected
- bit 6 =1 if the measurement tube is empty
- bit 7 =1 if the circuit powering the coils is not working or the sensor is disconnected
- bit 8 =1 if the second measurement scale is active
- bit 9 =1 if the Flow rate is lower than the cut-off threshold
- bit10=1 if the Flow rate is negative
- bit11=1 if a new measurement value calculated for the display is available
- bit12=1 if the counter block signal is active
- bit13=1 if dosing is in progress
  - bit14=1 if a calibration cycle is in progress
  - bit15=1 if a Flow rate simulation is in progress

byte 22: (8 bits integer) measurement dynamic variation as a %

byte 23: (8 bits integer) Data Type ( see section 2.6 Data Type )

**NOTE:** The values in the "32 bits single precision IEEE floating point" format are floating point numbers which can be represented during writing by any decimal digits. To keep the same numerical format visible on the instrument display however, it is necessary to calculate the decimal figures with a rather complex algorithm which takes account of instrument precision, flow rate measurement unit, etc. For this purpose and to avoid useless calculations, the number of decimals to use to represent the flow rate values is supplied separately. With the acyclic service it is possible to return the value ( see **Acyclic Index Read = 20(dec) Static Process Data of the Flow Meter** byte position 13).



	TECHNICAL MANUAL		Rev.	Page	17/52
	PROFIBUS DP MODULE- DATA LINK DESCRIPTION		03	Filling By	Checked By
			Document Data	Date	Date
	File name	ProfibusDP_emanual_Burkert_02.doc	20/07/04	12-01-07	12-01-07

**NOTE:**The counters are expressed with a 32 bit integer. The «counter decimal figures» parameter, indicates the point position starting from the right: 0 = no decimal, 1=1 decimal figure, and so on. ( see **Acyclic Index Read = 20(dec) Static Process Data of the Flow Meter** byte position 12)

## **2.6. DATA TYPE**

The firsts four byte of the **Dynamic Data** Block are the information about the Flow rate.

With the **Control byte** of the Configuration in the GSD file (see the Section Slave Configuration) it is possible to set the type of the Flow rate returned by the slave.

The type of the Flow rate may be in % or in technical unit in base of the value of the **Control byte** inserted in the Configuration string.

The byte 23 indicate the current type of data selected:

byte 23 = 0 --> Flow rate in %

byte 23 = 1 --> Flow rate in technical unit

For details see the Section Slave Configuration.

There is a second way to set the type of the Flow rate using the **Acyclic Service** during data exchange (see section **Acyclic Service**).

## **2.7. FOURTH CONFIGURATION: 16 BYTE INPUT/OUTPUT**

With the precedent configurations, the type of data as **Static Data** and other parameters of the Flow meter, the only possible way for to return the data was with **Acyclic Service** and Class 1 or Class 2 Master.

In the same way the insertion of data in the slave was with **Acyclic Service**.

If the Master don't support the **Acyclic Service**, with this configuration it is possible, with two indexes, read and write **Static Data** and insert data to the slave as Output data.

With the **Output** data it is possible the insertion of the values of the two Indexes used for the selection of the type of data returned with the Input data, for example the **Static Data** or others parameters, and insert other data as the Time of the clock of the Flow meter and other parameters.

The two indexes used for the selection and the insertion of the data are:

INDEX Input

INDEX Output

The two indexes are inserted with the Output data during the cyclic data exchange in the firsts two byte of the 16 Output data ( see the structure of the 16 bytes below in this section ).


The current value of the two index are returned with the first two byte of the 16 bytes Input data ( see the structure of the 16 bytes below in this section ).

The byte INDEX Input in the 16 bytes of the Output data, is used for the selection of the type of data returned with the 16 Input data.

With the value inserted in this byte it is possible to select the type of the **Dynamic Data** returned or the type of the **Static Data** or other internal parameter of the Flow meter or of the Slave as for example the Profibus address etc.

The byte INDEX Output in the 16 bytes of the Output data, is used for indicate the type of information that are inserted with the successive bytes that follow the two indexes ( byte 2 --> 15 ) in the 16 Output data.

For example with the value inserted in this byte it is possible to indicate that the bytes that follow the firsts

	TECHNICAL MANUAL		Rev.	Page	18/52
	PROFIBUS DP MODULE- DATA LINK DESCRIPTION		03	Filling By	Checked By
			Document Data	Date	Date
	File name	ProfibusDP_emanual_Burkert_02.doc	20/07/04	12-01-07	12-01-07

two bytes ( the two indexes ) are the bytes for to update the clock ( to update the clock is possible only for ML 210 ) of the Flow meter.

The firsts two byte of the Input data indicate always the current value of the INDEX Input and Output inserted with the 16 Output data.

**NOTE:** The values in the "32 bits single precision IEEE floating point" format are floating point numbers which can be represented during writing by any decimal digits. To keep the same numerical format visible on the instrument display however, it is necessary to calculate the decimal figures with a rather complex algorithm which takes account of instrument precision, flow rate measurement unit, etc. For this purpose and to avoid useless calculations, the number of decimals to use to represent the flow rate values is supplied separately (see **INDEX Input = 30(dec) Static Process Data of the Flow Meter** byte position 15).

**NOTE:** The counters are expressed with a 32 bit integer. The «counter decimal figures» parameter, indicates the point position starting from the right: 0 = no decimal, 1=1 decimal figure, and so on (see **INDEX Input = 30(dec) Static Process Data of the Flow Meter** byte position 15).

Below there is a description of the contents of each 16 bytes for every type of **INDEX Input** ( this index is inserted with the first byte of the Output data):

#### **INDEX Input = 0(dec) Dynamic Process Data**

- byte 0: (8 bits integer) current INDEX Input
- byte 1: (8 bits integer) current INDEX Output
- bytes 2-5: (32 bit single precision IEEE floating point, MSB first) **Flow rate in %**
- bytes 6-9: (32 bit long integer, MSB first) totalizer for TOTAL volume + or batch quantity ( batch quantity only for ML M3F )
- bytes 10-13: (32 bit long integer, MSB first) totalizer for PARTIAL volume + or batch counter ( batch counter only for ML M3F )
- bytes 14-15: (16 bit unsigned integer, MSB first) process flags:
  - bit 0 =1 if the excitation is too fast for the sensor connected
  - bit 1 =1 if the maximum alarm is active
  - bit 2 =1 if the minimum alarm is active
  - bit 3 =1 if the Flow rate exceeds the scale range value (overflow)
  - bit 4 =1 if one or more output impulses are saturated (too many impulses to emit)
  - bit 5 =1 if the measurement signal is highly disturbed or if the sensor is disconnected
  - bit 6 =1 if the measurement tube is empty
  - bit 7 =1 if the circuit powering the coils is not working or the sensor is disconnected
  - bit 8 =1 if the second measurement scale is active
  - bit 9 =1 if the Flow rate is lower than the cut-off threshold
  - bit10=1 if the Flow rate is negative
  - bit11=1 if a new measurement value calculated for the display is available
  - bit12=1 if the counter block signal is active
  - bit13=1 if dosing is in progress
    - bit14=1 if a calibration cycle is in progress
    - bit15=1 if a Flow rate simulation is in progress

#### **INDEX Input = 1(dec) Dynamic Process Data**

- byte 0: (8 bits integer) current INDEX Input
- byte 1: (8 bits integer) current INDEX Output
- bytes 2-5: (32 bit single precision IEEE floating point, MSB first) **Flow rate in %**
- bytes 6-9: (32 bit long integer, MSB first) totalizer for TOTAL volume - or batch quantity ( data valid only for ML 210 )

bytes 10-13: (32 bit long integer, MSB first) totalizer for PARTIAL volume - or batch counter ( data valid only for ML 210 )

bytes 14-15: (16 bit unsigned integer, MSB first) process flags:

- bit 0 =1 if the excitation is too fast for the sensor connected
- bit 1 =1 if the maximum alarm is active
- bit 2 =1 if the minimum alarm is active
- bit 3 =1 if the Flow rate exceeds the scale range value (overflow)
- bit 4 =1 if one or more output impulses are saturated (too many impulses to emit)
- bit 5 =1 if the measurement signal is highly disturbed or if the sensor is disconnected
- bit 6 =1 if the measurement tube is empty
- bit 7 =1 if the circuit powering the coils is not working or the sensor is disconnected
- bit 8 =1 if the second measurement scale is active
- bit 9 =1 if the Flow rate is lower than the cut-off threshold
- bit10=1 if the Flow rate is negative
- bit11=1 if a new measurement value calculated for the display is available
- bit12=1 if the counter block signal is active
- bit13=1 if dosing is in progress
- bit14=1 if a calibration cycle is in progress
- bit15=1 if a Flow rate simulation is in progress

**INDEX Input = 10(dec) Dynamic Process Data**

byte 0: (8 bits integer) current INDEX Input

byte 1: (8 bits integer) current INDEX Output

bytes 2-5: (32 bit single precision IEEE floating point, MSB first) **Flow rate in t.u.**

bytes 6-9: (32 bit long integer, MSB first) totalizer for TOTAL volume + or batch quantity ( batch quantity only for ML M3F )

bytes 10-13: (32 bit long integer, MSB first) totalizer for PARTIAL volume + or batch counter ( batch counter only for ML M3F )

bytes 14-15: (16 bit unsigned integer, MSB first) process flags:

- bit 0 =1 if the excitation is too fast for the sensor connected
- bit 1 =1 if the maximum alarm is active
- bit 2 =1 if the minimum alarm is active
- bit 3 =1 if the Flow rate exceeds the scale range value (overflow)
- bit 4 =1 if one or more output impulses are saturated (too many impulses to emit)
- bit 5 =1 if the measurement signal is highly disturbed or if the sensor is disconnected
- bit 6 =1 if the measurement tube is empty
- bit 7 =1 if the circuit powering the coils is not working or the sensor is disconnected
- bit 8 =1 if the second measurement scale is active
- bit 9 =1 if the Flow rate is lower than the cut-off threshold
- bit10=1 if the Flow rate is negative
- bit11=1 if a new measurement value calculated for the display is available
- bit12=1 if the counter block signal is active
- bit13=1 if dosing is in progress
- bit14=1 if a calibration cycle is in progress
- bit15=1 if a Flow rate simulation is in progress

**INDEX Input = 11(dec) Dynamic Process Data**

byte 0: (8 bits integer) current INDEX Input

byte 1: (8 bits integer) current INDEX Output

bytes 2-5: (32 bit single precision IEEE floating point, MSB first) **Flow rate in t.u.**

bytes 6-9: (32 bit long integer, MSB first) totalizer for TOTAL volume - or batch quantity ( data valid only

for ML 210 )

bytes 10-13: (32 bit long integer, MSB first) totalizer for PARTIAL volume - or batch counter ( data valid only for ML 210 )

bytes 14-15: (16 bit unsigned integer, MSB first) process flags:

- bit 0 =1 if the excitation is too fast for the sensor connected
- bit 1 =1 if the maximum alarm is active
- bit 2 =1 if the minimum alarm is active
- bit 3 =1 if the Flow rate exceeds the scale range value (overflow)
- bit 4 =1 if one or more output impulses are saturated (too many impulses to emit)
- bit 5 =1 if the measurement signal is highly disturbed or if the sensor is disconnected
- bit 6 =1 if the measurement tube is empty
- bit 7 =1 if the circuit powering the coils is not working or the sensor is disconnected
- bit 8 =1 if the second measurement scale is active
- bit 9 =1 if the Flow rate is lower than the cut-off threshold
- bit10=1 if the Flow rate is negative
- bit11=1 if a new measurement value calculated for the display is available
- bit12=1 if the counter block signal is active
- bit13=1 if dosing is in progress
- bit14=1 if a calibration cycle is in progress
- bit15=1 if a Flow rate simulation is in progress

#### **INDEX Input = 20(dec) Dynamic Process Data**

byte 0: (8 bits integer) current INDEX Input

byte 1: (8 bits integer) current INDEX Output

bytes 2-5: (32 bit single precision IEEE floating point, MSB first) **Flow rate in %**

bytes 6-9: (32 bit long integer, MSB first) totalizer for TOTAL volume + or batch quantity ( batch quantity only for ML M3F )

bytes 10-13: (32 bit long integer, MSB first) totalizer for TOTAL volume - or batch quantity ( data valid only for ML 210 )

bytes 14-15: (16 bit unsigned integer, MSB first) process flags:

- bit 0 =1 if the excitation is too fast for the sensor connected
- bit 1 =1 if the maximum alarm is active
- bit 2 =1 if the minimum alarm is active
- bit 3 =1 if the Flow rate exceeds the scale range value (overflow)
- bit 4 =1 if one or more output impulses are saturated (too many impulses to emit)
- bit 5 =1 if the measurement signal is highly disturbed or if the sensor is disconnected
- bit 6 =1 if the measurement tube is empty
- bit 7 =1 if the circuit powering the coils is not working or the sensor is disconnected
- bit 8 =1 if the second measurement scale is active
- bit 9 =1 if the Flow rate is lower than the cut-off threshold
- bit10=1 if the Flow rate is negative
- bit11=1 if a new measurement value calculated for the display is available
- bit12=1 if the counter block signal is active
- bit13=1 if dosing is in progress
- bit14=1 if a calibration cycle is in progress
- bit15=1 if a Flow rate simulation is in progress

#### **INDEX Input = 21(dec) Dynamic Process Data**

byte 0: (8 bits integer) current INDEX Input

byte 1: (8 bits integer) current INDEX Output

bytes 2-5: (32 bit single precision IEEE floating point, MSB first) **Flow rate in t.u.**

bytes 6-9: (32 bit long integer, MSB first) totalizer for TOTAL volume + or batch quantity ( batch quantity only for ML M3F )

bytes 10-13: (32 bit long integer, MSB first) totalizer for TOTAL volume - or batch quantity ( data valid only for ML 210 )

bytes 14-15: (16 bit unsigned integer, MSB first) process flags:

- bit 0 =1 if the excitation is too fast for the sensor connected
- bit 1 =1 if the maximum alarm is active
- bit 2 =1 if the minimum alarm is active
- bit 3 =1 if the Flow rate exceeds the scale range value (overflow)
- bit 4 =1 if one or more output impulses are saturated (too many impulses to emit)
- bit 5 =1 if the measurement signal is highly disturbed or if the sensor is disconnected
- bit 6 =1 if the measurement tube is empty
- bit 7 =1 if the circuit powering the coils is not working or the sensor is disconnected
- bit 8 =1 if the second measurement scale is active
- bit 9 =1 if the Flow rate is lower than the cut-off threshold
- bit10=1 if the Flow rate is negative
- bit11=1 if a new measurement value calculated for the display is available
- bit12=1 if the counter block signal is active
- bit13=1 if dosing is in progress
- bit14=1 if a calibration cycle is in progress
- bit15=1 if a Flow rate simulation is in progress

**INDEX Input = 30(dec) Static Process Data of the Flow Meter**

- byte 0: (8 bits integer) current INDEX Input
- byte 1: (8 bits integer) current INDEX Output
- bytes 2-5: (32 bit single precision IEEE floating point, MSB first) Flow rate scale range in t.u.
- bytes 6-10: (5 bytes ASCII) flow rate measurement unit
- byte 12: (8 bits integer) number of decimals after the point for flow rate display

**INDEX Input = 31(dec) Static Process Data of the Flow Meter**

- byte 0: (8 bits integer) current INDEX Input
- byte 1: (8 bits integer) current INDEX Output
- bytes 2-4: (3 bytes ASCII) measurement unit of the counters
- byte 6: (8 bits integer) number of decimals after the point for totalizers display
- byte 8: (8 bits integer) measurement samples per second (Hz)

- byte 20=10(dec) measurement frequency = 10 Hz
- byte 20=20(dec) measurement frequency = 20 Hz
- byte 20=50(dec) measurement frequency = 50 Hz
- byte 20=80(dec) measurement frequency = 80 Hz
- byte 20=150(dec) measurement frequency = 150 Hz
- byte 20=44(dec) measurement frequency = 300 Hz
- byte 20=144(dec) measurement frequency = 400 Hz

**INDEX Input = 32(dec) Slave Data**

- byte 0: (8 bits integer) current INDEX Input
- byte 1: (8 bits integer) current INDEX Output
- byte 2: (8 bits integer) slave address
- byte 3: (8 bits integer) pno identification number high
- byte 4: (8 bits integer) pno identification number low
- byte 5: (8 bits integer) DPC31 ASIC Step

bit 0 =1 if DPC31 B

byte 6: (8 bits integer) MAC State Offline / Passive Idle

bit 0 =1 if the MAC is in Passive Idle

byte 7: (8 bits integer) watchdog state

bit 0-1 = 00 --> Baud Search

01 --> Baud Control

10 --> DP\_Control

byte 8: (8 bits integer) baud rate

bit 0-3 = 0000 --> 12 MBd

0001 --> 6 MBd

0010 --> 3 MBd

0011 --> 1.5 MBd

0100 --> 500 kBd

0101 --> 187.5 kBd

0110 --> 93.75 kBd

0111 --> 45.45 kBd

1000 --> 19.2 kBd

1001 --> 9.6 kBd

byte 9: (8 bits integer) DP state machine

bit 0-1 = 11 --> Data\_Exchange

**INDEX Input = 33(dec) Internal Data**

byte 0: (8 bits integer) current INDEX Input

byte 1: (8 bits integer) current INDEX Output

byte 2: (8 bits integer) slave software version (major number)

byte 3: (8 bits integer) slave software version (minor number)

byte 4-9: (8 bits integer) device name («ML 210», «ML M3F»)

byte 10: (8 bits integer) Flow meter software version (major number)

byte 11: (8 bits integer) Flow meter software version (minor number)

byte 12-13: (8 bits integer) (16 bits unsigned integer, byte 13=MSB):hardware / software enabling flags, bit 0 = LSB, bit 15 = MSB:

Bit s	Funzioni ML 210/M3F
0-2	current access level (0..7)
3	channel 1 impulses in use
4	channel 2 impulses in use
5	channel 1 frequency in use
6	channel 2 frequency in use
7	scale range 2 in use

8	specific weight in use
9	additional output 3 present
10	additional output 4 present
11	output 4..20 mA 2 present
12	RS232 present
13	batching functions active
14	output 4..20 mA 2 present
15	RS485 present

byte 14: (8 bits integer) current language set in the Flow meter

Format of the returned data:

- 0(dec) = English
- 1(dec) = Italian
- 2(dec) = French
- 3(dec) = Spanish

**NOTE:** For reading the type of language it is first necessary to send to the slave the relative command, see the **INDEX Output = 10** (below in this section) and then read the type of language in the byte 4.

**INDEX Input = 34(dec) Data and Time of the Flow meter ( only for ML 210)**

- byte 0: (8 bits integer) current INDEX Input
- byte 1: (8 bits integer) current INDEX Output
- bytes 2-5 (32 bit long integer, MSB first) clock expressed in minutes starting from 01-01-1992

**NOTE:** The date and time are expressed with a 32 bit integer containing the number of minutes elapsed since 01-01-1992. To calculate the date starting from this number, see the programming examples in the section 4.0 PROGRAMMING ALGORITMS of this manual.

**INDEX Input = 40(dec) Batch Data**

- byte 0: (8 bits integer) current INDEX Input
- byte 1: (8 bits integer) current INDEX Output
- bytes 2-3: (2 bytes 16 bit unsigned integer, MSB first) number of batch processes done for this batch memory.
- bytes 4-5: (2 bytes 16 bit unsigned integer, MSB first) value of the batch safety timer expressed in tenths of seconds.
- bytes 6-9: (4 bytes 32 bits unsigned long integer, MSB first) batch quantity value expressed in the same units and with the same decimal digits of the volume counters
- byte 10: (8 bits integer) batch status:

Format of the returned data: 1 byte containing the batch process state:

- 0(dec)** = batch process is correctly terminated (pre-set quantity reached).
- 1(dec)** = batch process is running (the valve is opened and the counters are running).
- 2(dec)** = batch process is suspended (the valve is closed before the pre-set quantity is reached).

NOTE: The recipe utilized for this configuration is the 00 in the meter and the name utilized for this recipe is PROFIBUS.

**INDEX Input = 42(dec) Batch Parameters**

- byte 0: (8 bits integer) current INDEX Input

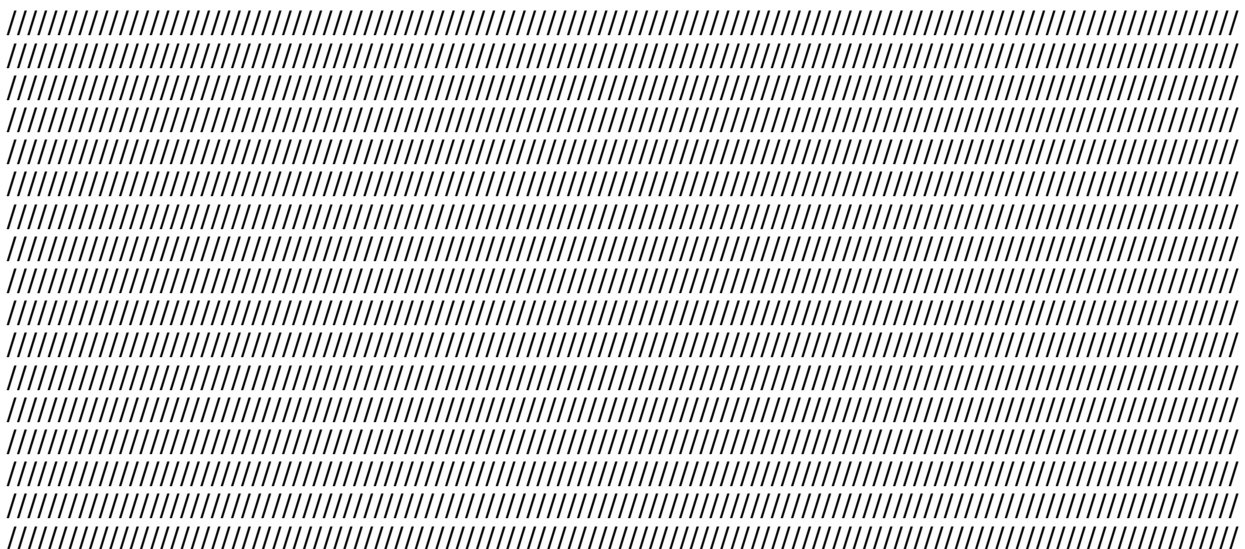
TECHNICAL MANUAL		Rev.	Page	24/52
PROFIBUS DP MODULE- DATA LINK DESCRIPTION		03	Filling By	Checked By
		Document Data	Date	Date
File name	ProfibusDP_emanual_Burkert_02.doc	20/07/04	12-01-07	12-01-07

byte 1: (8 bits integer) current INDEX Output  
 byte 2: (8 bits integer) batch status:

Format of the returned data: 1 byte containing the batch process state:

- 0(dec)** = batch process is correctly terminated (pre-set quantity reached).
- 1(dec)** = batch process is running (the valve is opened and the counters are running).
- 2(dec)** = batch process is suspended (the valve is closed before the pre-set quantity is reached).

byte 3: (8 bits integer) menu Measure – Cut-off es. 1(dec) / 10 = 0.1%, 255(dec) / 10 → 25.5%  
 byte 4: (8 bits integer) menu Batch - N.samples  
 byte 5: (8 bits integer) menu Batch – Diff.thr  
 bytes 6-7: (2 bytes 16 bit unsigned integer, MSB first) menu Batch – V.com (value expressed in the same units and with the same decimal digits of the volume counters)  
 bytes 8-9: (2 bytes 16 bit unsigned integer, MSB first) menu Batch – V.pre (value expressed in the same units and with the same decimal digits of the volume counters)



Below there is a description of the contents of each 16 bytes for every type of **INDEX Output** ( this index is inserted with the second byte of the Output data):

**NOTE:** Setting the commands only in the state OPERATE. In the state CLEAR the commands are not accepted.

**INDEX Output = 10(dec) Flow Meter Commands**

byte 0: (8 bits integer) INDEX Input: the value set the type of the Input data  
 byte 1: (8 bits integer) INDEX Output = 10(dec)  
 byte 2: (8 bits integer) command type

Type of command that is possible to insert in the byte 2:

Byte 2 = **0x06(hex)** command for reset the Totalizers of the Flow Meter ( see the Flow Meter Manual and the relative menu for to enable which totalizers to reset by this command).

Byte 2 = **0x07(hex)** command for read the current Language used in the Flow Meter:  
 The information is returned by the byte 4 of the Input data with Index Input 32.



Byte 2 = **0x08(hex)** command for Reset the slave module. After 5 seconds the module return in the data exchange.

Byte 2 = **0x32(hex)** command for Autozero Calibration

Byte 2 = **0x60(hex)** command for set the type of language in the meter ( Note: It is firstly necessary to read the type of the language of the meter, for to set it with this command )

Byte 3 = type of language:

0x00 = 0(dec) = English

0x01 = 1(dec) = Italian

0x02 = 2(dec) = French

0x03 = 3(dec) = Spanish

es.

Reset Totalizers

byte 0: 0xXX = index input

byte 1: 0x0A = **10(dec)** index output

byte 2: 0x06 → Command

-----

es.

Reading current type language of the Flow Meter

byte 0: 0x21 = **33(dec)** index input → see **INDEX Input = 33(dec)**

byte 1: 0x0A = **10(dec)** index output

byte 2: 0x07 → Command

-----

es.

Writing the type language in the Flow Meter after the firstly reading of the current language set in the meter

byte 0: 0x21 = **33(dec)** index input → see **INDEX Input = 33(dec)**

byte 1: 0x0A = **10(dec)** index output

byte 2: 0x60 = **96(dec)** Command for set language

byte 3: 0x01 → set italian language in the meter

-----

es.

Set Autozero Calibration

byte 0: 0xXX = index input

byte 1: 0x0A = **10(dec)** index output

byte 2: 0x32 → Command

**INDEX Output = 20(dec) Setting Data and Time in the Flow Meter (only for ML 210)**

byte 0: (8 bits integer) INDEX Input: the value set the type of the Input data

byte 1: (8 bits integer) INDEX Output = 20(dec)

byte 2-5: 32 bits long integer containing the date and time expressed in minutes starting from 01-01-1992 the most significant byte must be sent first.

**INDEX Output = 40(dec) Batch Commands**

For to use the batch function it is firstly necessary to set ON the menu Batch in the Flow Meter ( see the Flow Meter Manual )

byte 0: (8 bits integer) INDEX Input: the value set the type of the Input data

byte 1: (8 bits integer) INDEX Output = 40(dec)

byte 2: (8 bits integer) batch **Command**

**0x10(hex)** Read the batch data memory, the data are returned with the **INDEX Input = 40(dec)**

byte 3: (8 bits integer) batch **Code**

Type of code that is possible to insert in the byte 3:

**0x40(hex)** = command for reading the recipe

**0x12(hex)** Set batch with the code in the byte 3

byte 3: (8 bits integer) batch **Code**

Type of command that is possible to insert in the byte 3:

**0x01(hex)** = command for reading the current batch status ( see **INDEX Input = 40(dec) or 42(dec)** for to read the status)

**0x02(hex)** = command for starting the current batch process

**0x03(hex)** = command for suspending the current batch process

**0x04(hex)** = command for resetting the current batch process

**0x13(hex)** Read the batch parameters, the data are returned with the **INDEX Input = 42(dec)**

es. Reading Batch Data:

byte 0: 0x28 = **40(dec)** index input → see **INDEX Input = 40(dec) Batch Data**

byte 1: 0x28 = **40(dec)** index output

byte 2: 0x10 → Command

byte 3: 0x40 → Code

es. Reading Batch Parameters:

byte 0: 0x2A = **42(dec)** index input → see **INDEX Input = 40(dec) Batch Data**

byte 1: 0x28 = **40(dec)** index output

byte 2: 0x13 → Command

es. Batch Commands:

byte 0: 0x28 = **40(dec)** index input → see **INDEX Input = 40(dec) Batch Data**

byte 1: 0x28 = **40(dec)** index output

byte 2: 0x12 → Command

byte 3: 0x01 → Code Read Status

0x02 → Code Start Batch

0x03 → Code Stop Batch

0x04 → Code Reset the current Batch process

**INDEX Output = 41(dec) Setting Batch Data Memory**

With this index it is possible to insert the batch data memory

byte 0: (8 bits integer) INDEX Input: the value set the type of the Input data

byte 1: (8 bits integer) INDEX Output = 40(dec)

bytes 2-3: (2 bytes 16 bit unsigned integer, MSB first) number of batch processes done for this batch memory.

bytes 4-5: (2 bytes 16 bit unsigned integer, MSB first) value of the safety batch timer expressed in tenths of seconds.

bytes 6-9: (4 bytes 32 bits unsigned long integer, MSB first) batch quantity value expressed in the same units and with the same decimal digits of the volume counters.

NOTE: The recipe utilized for this configuration (16 bytes I/O) is the 00 in the meter and the name utilized for this recipe is PROFIBUS.

es.

Insertion of the Batch Data (insertion of the data in the meter is in real time for this type of data):

byte 0: 0x28 = **40(dec)** index input → see **INDEX Input = 40(dec) Batch Data**

byte 1: 0x29 = **41(dec)** index output

byte 2: 0x00

byte 3: 0x00 → 0(dec) value for reset the counter

byte 4: 0x00

byte 5: 0x0F → 15 (dec) safety batch timer

byte 6: 0x00

byte 7: 0x00

byte 8: 0x00

byte 9: 0x01 → 1(dec) batch quantity

**INDEX Output = 42(dec) Setting Batch Parameters**

With this index it is possible to insert the batch parameters

byte 0: (8 bits integer) INDEX Input: the value set the type of the Input data  
 byte 1: (8 bits integer) INDEX Output = 42(dec)  
 byte 2: (8 bits integer) menu Measure – cut-off es.  $1(\text{dec}) / 10 = 0.1\%$ ,  $255(\text{dec}) / 10 \rightarrow 25.5\%$   
 byte 3: (8 bits integer) menu Batch - N.samples  
 byte 4: (8 bits integer) menu Batch – Diff.thr  
 bytes 6-7: (2 bytes 16 bit unsigned integer, MSB first) menu Batch – V.com (value expressed in the same units and with the same decimal digits of the volume counters)  
 bytes 8-9: (2 bytes 16 bit unsigned integer, MSB first) menu Batch – V.pre (value expressed in the same units and with the same decimal digits of the volume counters)

es.

Insertion of the parameters value:

byte 0: 0xXX index input  
 byte 1: 0x2A = **42(dec)** index output  
  
 byte 2: 0x01  $\rightarrow$  cut-off ( menu Measure ) = 0.1%  
  
 byte 3: 0x02  $\rightarrow$  N.samples ( menu Batch ) = 2  
  
 byte 4: 0x03  $\rightarrow$  Diff.thr ( menu Batch ) = 3%  
  
 byte 5: 0x00  $\rightarrow$  byte alignment  
  
 byte 6: 0x00  
 byte 7: 0x04  $\rightarrow$  V.com ( menu Batch ) = 00.004  
  
 byte 8: 0x00  
 byte 9: 0x05  $\rightarrow$  V.pre ( menu Batch ) = 00.005

**2.8. FIFTH CONFIGURATION: 24 BYTE INPUT/OUTPUT**

With the precedent configurations, the type of data as **Static Data** and other parameters of the Flow meter, the only possible way for to return the data was with **Acyclic Service** and Class 1 or Class 2 Master. In the same way the insertion of data in the slave was with **Acyclic Service**.


If the Master don't support the **Acyclic Service**, with this configuration it is possible, with two indexes, read and write **Static Data** and insert data to the slave as Output data. With the **Output** data it is possible the insertion of the values of the two Indexes used for the selection of the type of data returned with the Input data, for example the **Static Data** or others parameters, and insert other data as the Time of the clock of the Flow meter and other parameters.

The two indexes used for the selection and the insertion of the data are:

INDEX Input  
 INDEX Output

The two indexes are inserted with the Output data during the cyclic data exchange in the firsts two byte of the 24 Output data ( see the structure of the 24 bytes below in this section ).

The current value of the two index are returned with the first two byte of the 24 bytes Input data ( see the structure of the 24 bytes below in this section ).

	TECHNICAL MANUAL		Rev.	Page	29/52
	PROFIBUS DP MODULE- DATA LINK DESCRIPTION		03	Filling By	Checked By
			Document Data	Date	Date
	File name	ProfibusDP_emanual_Burkert_02.doc	20/07/04	12-01-07	12-01-07

The byte INDEX Input in the 24 bytes of the Output data, is used for the selection of the type of data returned with the 24 Input data.

With the value inserted in this byte it is possible to select the type of the **Dynamic Data** returned or the type of the **Static Data** or other internal parameter of the Flow meter or of the Slave as for example the Profibus address etc.

The byte INDEX Output in the 24 bytes of the Output data, is used for indicate the type of information that are inserted with the successive bytes that follow the two indexes ( byte 2 --> 23 ) in the 24 Output data.

For example with the value inserted in this byte it is possible to indicate that the bytes that follow the firsts two bytes ( the two indexes ) are the bytes for to update the clock ( to update the clock is possible only for ML 210 ) of the Flow meter.

The firsts two byte of the Input data indicate always the current value of the INDEX Input and Output inserted with the 24 Output data.

**NOTE:** The values in the "32 bits single precision IEEE floating point" format are floating point numbers which can be represented during writing by any decimal digits. To keep the same numerical format visible on the instrument display however, it is necessary to calculate the decimal figures with a rather complex algorithm which takes account of instrument precision, flow rate measurement unit, etc. For this purpose and to avoid useless calculations, the number of decimals to use to represent the flow rate values is supplied separately (see **INDEX Input = 30(dec) Static Process Data of the Flow Meter** byte position 15).

**NOTE:** The counters are expressed with a 32 bit integer. The «counter decimal figures» parameter, indicates the point position starting from the right: 0 = no decimal, 1=1 decimal figure, and so on (see **INDEX Input = 30(dec) Static Process Data of the Flow Meter** byte position 15).

Below there is a description of the contents of each 24 bytes for every type of **INDEX Input** ( this index is inserted with the first byte of the Output data):

### **INDEX Input = 0(dec) Dynamic Process Data**

byte 0: (8 bits integer) current INDEX Input

byte 1: (8 bits integer) current INDEX Output

bytes 2-5: (32 bit single precision IEEE floating point, MSB first) **Flow rate in %**

bytes 6-9: (32 bit long integer, MSB first) totalizer for TOTAL volume + or batch quantity ( batch quantity only for ML M3F )

bytes 10-13: (32 bit long integer, MSB first) totalizer for PARTIAL volume + or batch counter ( batch counter only for ML M3F )

bytes 14-17: (32 bit long integer, MSB first) totalizer for TOTAL volume - or batch quantity ( data valid only for ML 210 )

bytes 18-21: (32 bit long integer, MSB first) totalizer for PARTIAL volume - or batch quantity ( data valid only for ML 210 )

bytes 22-23: (16 bit unsigned integer, MSB first) process flags:

bit 0 =1 if the excitation is too fast for the sensor connected

bit 1 =1 if the maximum alarm is active

bit 2 =1 if the minimum alarm is active

bit 3 =1 if the Flow rate exceeds the scale range value (overflow)

bit 4 =1 if one or more output impulses are saturated (too many impulses to emit)

bit 5 =1 if the measurement signal is highly disturbed or if the sensor is disconnected

bit 6 =1 if the measurement tube is empty

bit 7 =1 if the circuit powering the coils is not working or the sensor is disconnected

bit 8 =1 if the second measurement scale is active

bit 9 =1 if the Flow rate is lower than the cut-off threshold

- bit10=1 if the Flow rate is negative
- bit11=1 if a new measurement value calculated for the display is available
- bit12=1 if the counter block signal is active
- bit13=1 if dosing is in progress
- bit14=1 if a calibration cycle is in progress
- bit15=1 if a Flow rate simulation is in progress

**INDEX Input = 1(dec) Dynamic Process Data**

- byte 0: (8 bits integer) current INDEX Input
- byte 1: (8 bits integer) current INDEX Output
- bytes 2-5: (32 bit single precision IEEE floating point, MSB first) **Flow rate in %**
- bytes 6-9: (32 bit long integer, MSB first) totalizer for TOTAL volume + or batch quantity ( batch quantity only for ML M3F )
- bytes 10-13: (32 bit long integer, MSB first) totalizer for PARTIAL volume + or batch counter ( batch counter only for ML M3F )
- bytes 14-15: (16 bit unsigned integer, MSB first) process flags:
  - bit 0 =1 if the excitation is too fast for the sensor connected
  - bit 1 =1 if the maximum alarm is active
  - bit 2 =1 if the minimum alarm is active
  - bit 3 =1 if the Flow rate exceeds the scale range value (overflow)
  - bit 4 =1 if one or more output impulses are saturated (too many impulses to emit)
  - bit 5 =1 if the measurement signal is highly disturbed or if the sensor is disconnected
  - bit 6 =1 if the measurement tube is empty
  - bit 7 =1 if the circuit powering the coils is not working or the sensor is disconnected
  - bit 8 =1 if the second measurement scale is active
  - bit 9 =1 if the Flow rate is lower than the cut-off threshold
  - bit10=1 if the Flow rate is negative
  - bit11=1 if a new measurement value calculated for the display is available
  - bit12=1 if the counter block signal is active
  - bit13=1 if dosing is in progress
  - bit14=1 if a calibration cycle is in progress
  - bit15=1 if a Flow rate simulation is in progress
- byte 16: (8 bits integer) measurement dynamic variation as a %

**INDEX Input = 2(dec) Dynamic Process Data**

- byte 0: (8 bits integer) current INDEX Input
- byte 1: (8 bits integer) current INDEX Output
- bytes 2-5: (32 bit single precision IEEE floating point, MSB first) **Flow rate in %**
- bytes 6-9: (32 bit long integer, MSB first) totalizer for TOTAL volume - or batch quantity ( data valid only for ML 210 )
- bytes 10-13: (32 bit long integer, MSB first) totalizer for PARTIAL volume - or batch counter ( data valid only for ML 210 )
- bytes 14-15: (16 bit unsigned integer, MSB first) process flags:
  - bit 0 =1 if the excitation is too fast for the sensor connected
  - bit 1 =1 if the maximum alarm is active
  - bit 2 =1 if the minimum alarm is active
  - bit 3 =1 if the Flow rate exceeds the scale range value (overflow)
  - bit 4 =1 if one or more output impulses are saturated (too many impulses to emit)
  - bit 5 =1 if the measurement signal is highly disturbed or if the sensor is disconnected
  - bit 6 =1 if the measurement tube is empty
  - bit 7 =1 if the circuit powering the coils is not working or the sensor is disconnected

- bit 8 =1 if the second measurement scale is active
- bit 9 =1 if the Flow rate is lower than the cut-off threshold
- bit10=1 if the Flow rate is negative
- bit11=1 if a new measurement value calculated for the display is available
- bit12=1 if the counter block signal is active
- bit13=1 if dosing is in progress
- bit14=1 if a calibration cycle is in progress
- bit15=1 if a Flow rate simulation is in progress
- byte 16: (8 bits integer) measurement dynamic variation as a %

**INDEX Input = 10(dec) Dynamic Process Data**

- byte 0: (8 bits integer) current INDEX Input
- byte 1: (8 bits integer) current INDEX Output
- bytes 2-5: (32 bit single precision IEEE floating point, MSB first) **Flow rate in t.u.**
- bytes 6-9: (32 bit long integer, MSB first) totalizer for TOTAL volume + or batch quantity ( batch quantity only for ML M3F )
- bytes 10-13: (32 bit long integer, MSB first) totalizer for PARTIAL volume + or batch counter ( batch counter only for ML M3F )
- bytes 14-17: (32 bit long integer, MSB first) totalizer for TOTAL volume - or batch quantity ( data valid only for ML 210 )
- bytes 18-21: (32 bit long integer, MSB first) totalizer for PARTIAL volume - or batch counter ( data valid only for ML 210 )
- bytes 22-23: (16 bit unsigned integer, MSB first) process flags:
  - bit 0 =1 if the excitation is too fast for the sensor connected
  - bit 1 =1 if the maximum alarm is active
  - bit 2 =1 if the minimum alarm is active
  - bit 3 =1 if the Flow rate exceeds the scale range value (overflow)
  - bit 4 =1 if one or more output impulses are saturated (too many impulses to emit)
  - bit 5 =1 if the measurement signal is highly disturbed or if the sensor is disconnected
  - bit 6 =1 if the measurement tube is empty
  - bit 7 =1 if the circuit powering the coils is not working or the sensor is disconnected
  - bit 8 =1 if the second measurement scale is active
  - bit 9 =1 if the Flow rate is lower than the cut-off threshold
  - bit10=1 if the Flow rate is negative
  - bit11=1 if a new measurement value calculated for the display is available
  - bit12=1 if the counter block signal is active
  - bit13=1 if dosing is in progress
  - bit14=1 if a calibration cycle is in progress
  - bit15=1 if a Flow rate simulation is in progress

**INDEX Input = 11(dec) Dynamic Process Data**

- byte 0: (8 bits integer) current INDEX Input
- byte 1: (8 bits integer) current INDEX Output
- bytes 2-5: (32 bit single precision IEEE floating point, MSB first) **Flow rate in t.u.**
- bytes 6-9: (32 bit long integer, MSB first) totalizer for TOTAL volume + or batch quantity ( batch quantity only for ML M3F )
- bytes 10-13: (32 bit long integer, MSB first) totalizer for PARTIAL volume + or batch counter ( batch counter only for ML M3F )
- bytes 14-15: (16 bit unsigned integer, MSB first) process flags:
  - bit 0 =1 if the excitation is too fast for the sensor connected
  - bit 1 =1 if the maximum alarm is active

- bit 2 =1 if the minimum alarm is active
- bit 3 =1 if the Flow rate exceeds the scale range value (overflow)
- bit 4 =1 if one or more output impulses are saturated (too many impulses to emit)
- bit 5 =1 if the measurement signal is highly disturbed or if the sensor is disconnected
- bit 6 =1 if the measurement tube is empty
- bit 7 =1 if the circuit powering the coils is not working or the sensor is disconnected
- bit 8 =1 if the second measurement scale is active
- bit 9 =1 if the Flow rate is lower than the cut-off threshold
- bit10=1 if the Flow rate is negative
- bit11=1 if a new measurement value calculated for the display is available
- bit12=1 if the counter block signal is active
- bit13=1 if dosing is in progress
- bit14=1 if a calibration cycle is in progress
- bit15=1 if a Flow rate simulation is in progress

byte 16: (8 bits integer) measurement dynamic variation as a %

**INDEX Input = 12(dec) Dynamic Process Data**

- byte 0: (8 bits integer) current INDEX Input
- byte 1: (8 bits integer) current INDEX Output
- bytes 2-5: (32 bit single precision IEEE floating point, MSB first) **Flow rate in t.u.**
- bytes 6-9: (32 bit long integer, MSB first) totalizer for TOTAL volume - or batch quantity ( data valid only for ML 210 )
- bytes 10-13: (32 bit long integer, MSB first) totalizer for PARTIAL volume - or batch counter ( data valid only for ML 210 )
- bytes 14-15: (16 bit unsigned integer, MSB first) process flags:

- bit 0 =1 if the excitation is too fast for the sensor connected
- bit 1 =1 if the maximum alarm is active
- bit 2 =1 if the minimum alarm is active
- bit 3 =1 if the Flow rate exceeds the scale range value (overflow)
- bit 4 =1 if one or more output impulses are saturated (too many impulses to emit)
- bit 5 =1 if the measurement signal is highly disturbed or if the sensor is disconnected
- bit 6 =1 if the measurement tube is empty
- bit 7 =1 if the circuit powering the coils is not working or the sensor is disconnected
- bit 8 =1 if the second measurement scale is active
- bit 9 =1 if the Flow rate is lower than the cut-off threshold
- bit10=1 if the Flow rate is negative
- bit11=1 if a new measurement value calculated for the display is available
- bit12=1 if the counter block signal is active
- bit13=1 if dosing is in progress
- bit14=1 if a calibration cycle is in progress
- bit15=1 if a Flow rate simulation is in progress

byte 16: (8 bits integer) measurement dynamic variation as a %

**INDEX Input = 13(dec) Dynamic Process Data**

- byte 0: (8 bits integer) current INDEX Input
- byte 1: (8 bits integer) current INDEX Output
- bytes 2-5: (32 bit single precision IEEE floating point, MSB first) **Flow rate in %**
- bytes 6-9: (32 bit long integer, MSB first) totalizer for TOTAL volume + or batch quantity ( batch quantity only for ML M3F )



bytes 10-13: (32 bit long integer, MSB first) totalizer for TOTAL volume - or batch quantity ( data valid only for ML 210 )

bytes 14-15: (16 bit unsigned integer, MSB first) process flags:

- bit 0 =1 if the excitation is too fast for the sensor connected
- bit 1 =1 if the maximum alarm is active
- bit 2 =1 if the minimum alarm is active
- bit 3 =1 if the Flow rate exceeds the scale range value (overflow)
- bit 4 =1 if one or more output impulses are saturated (too many impulses to emit)
- bit 5 =1 if the measurement signal is highly disturbed or if the sensor is disconnected
- bit 6 =1 if the measurement tube is empty
- bit 7 =1 if the circuit powering the coils is not working or the sensor is disconnected
- bit 8 =1 if the second measurement scale is active
- bit 9 =1 if the Flow rate is lower than the cut-off threshold
- bit10=1 if the Flow rate is negative
- bit11=1 if a new measurement value calculated for the display is available
- bit12=1 if the counter block signal is active
- bit13=1 if dosing is in progress
  - bit14=1 if a calibration cycle is in progress
  - bit15=1 if a Flow rate simulation is in progress

byte 16: (8 bits integer) measurement dynamic variation as a %

**INDEX Input = 14(dec) Dynamic Process Data**

byte 0: (8 bits integer) current INDEX Input

byte 1: (8 bits integer) current INDEX Output

bytes 2-5: (32 bit single precision IEEE floating point, MSB first) **Flow rate in t.u.**

bytes 6-9: (32 bit long integer, MSB first) totalizer for TOTAL volume + or batch quantity ( batch quantity only for ML M3F )

bytes 10-13: (32 bit long integer, MSB first) totalizer for TOTAL volume - or batch quantity ( data valid only for ML 210 )

bytes 14-15: (16 bit unsigned integer, MSB first) process flags:

- bit 0 =1 if the excitation is too fast for the sensor connected
- bit 1 =1 if the maximum alarm is active
- bit 2 =1 if the minimum alarm is active
- bit 3 =1 if the Flow rate exceeds the scale range value (overflow)
- bit 4 =1 if one or more output impulses are saturated (too many impulses to emit)
- bit 5 =1 if the measurement signal is highly disturbed or if the sensor is disconnected
- bit 6 =1 if the measurement tube is empty
- bit 7 =1 if the circuit powering the coils is not working or the sensor is disconnected
- bit 8 =1 if the second measurement scale is active
- bit 9 =1 if the Flow rate is lower than the cut-off threshold
- bit10=1 if the Flow rate is negative
- bit11=1 if a new measurement value calculated for the display is available
- bit12=1 if the counter block signal is active
- bit13=1 if dosing is in progress
  - bit14=1 if a calibration cycle is in progress
  - bit15=1 if a Flow rate simulation is in progress

byte 16: (8 bits integer) measurement dynamic variation as a %

**INDEX Input = 30(dec) Static Process Data of the Flow Meter**

byte 0: (8 bits integer) current INDEX Input  
 byte 1: (8 bits integer) current INDEX Output  
 bytes 2-5: (32 bit single precision IEEE floating point, MSB first) Flow rate scale range in t.u.  
 bytes 6-10: (5 bytes ASCII) flow rate measurement unit  
 bytes 12-14: (3 bytes ASCII) measurement unit of the counters  
 byte 16: (8 bits integer) number of decimals after the point for totalizers display  
 byte 17: (8 bits integer) number of decimals after the point for flow rate display  
 bytes 18-21: (32 bit long integer, MSB first) clock expressed in minutes starting from 01-01-1992 (only for ML 210)  
 byte 22: (8 bits integer) measurement samples per second (Hz)

byte 20=10(dec) measurement frequency = 10 Hz  
 byte 20=20(dec) measurement frequency = 20 Hz  
 byte 20=50(dec) measurement frequency = 50 Hz  
 byte 20=80(dec) measurement frequency = 80 Hz  
 byte 20=150(dec) measurement frequency = 150 Hz  
 byte 20=44(dec) measurement frequency = 300 Hz  
 byte 20=144(dec) measurement frequency = 400 Hz

**NOTE:** The date and time are expressed with a 32 bit integer containing the number of minutes elapsed since 01-01-1992. To calculate the date starting from this number, see the programming examples in the section 4.0 PROGRAMMING ALGORITMS of this manual.

**INDEX Input = 31(dec) Slave Data**

byte 0: (8 bits integer) current INDEX Input  
 byte 1: (8 bits integer) current INDEX Output  
 byte 2: (8 bits integer) slave address  
 byte 3: (8 bits integer) pno identification number high  
 byte 4: (8 bits integer) pno identification number low  
 byte 5: (8 bits integer) DPC31 ASIC Step

bit 0 =1 if DPC31 B

byte 6: (8 bits integer) MAC State Offline / Passive Idle

bit 0 =1 if the MAC is in Passive Idle

byte 7: (8 bits integer) watchdog state

bit 0-1 = 00 --> Baud Search  
 01 --> Baud Control  
 10 --> DP\_Control

byte 8: (8 bits integer) baud rate

bit 0-3 = 0000 --> 12 MBd  
 0001 --> 6 MBd  
 0010 --> 3 MBd  
 0011 --> 1.5 MBd  
 0100 --> 500 kBd  
 0101 --> 187.5 kBd

0110 --> 93.75 kBd

0111 --> 45.45 kBd

1000 --> 19.2 kBd

1001 --> 9.6 kBd

byte 9: (8 bits integer) DP state machine

bit 0-1 = 11 --> Data\_Exchange

**INDEX Input = 32(dec) Internal Data**

byte 0: (8 bits integer) current INDEX Input

byte 1: (8 bits integer) current INDEX Output

byte 2: (8 bits integer) slave software version (major number)

byte 3: (8 bits integer) slave software version (minor number)

byte 4-9: (8 bits integer) device name («ML 210», «ML M3F»)

byte 10: (8 bits integer) Flow meter software version (major number)

byte 11: (8 bits integer) Flow meter software version (minor number)

byte 12-13: (8 bits integer) (16 bits unsigned integer, byte 13=MSB):hardware / software enabling flags, bit 0 = LSB, bit 15 = MSB:

Bit s	Funzioni ML 210/M3F
0-2	current access level (0..7)
3	channel 1 impulses in use
4	channel 2 impulses in use
5	channel 1 frequency in use
6	channel 2 frequency in use
7	scale range 2 in use
8	specific weight in use
9	additional output 3 present
10	additional output 4 present
11	output 4..20 mA 2 present
12	RS232 present
13	batching functions active
14	output 4..20 mA 2 present
15	RS485 present

byte 14: (8 bits integer) current language set in the Flow meter

Format of the returned data:

0(dec) = English

1(dec) = Italian

2(dec) = French

3(dec) = Spanish

**NOTE:** For reading the type of language it is first necessary to send to the slave the relative command, see the **INDEX Output = 10** (below in this section) and then read the type of language in the byte 4.

**INDEX Input = 40(dec) Batch Data**

TECHNICAL MANUAL		Rev.	Page	36/52
PROFIBUS DP MODULE- DATA LINK DESCRIPTION		03	Filling By	Checked By
		Document Data	Date	Date
File name	ProfibusDP_emanual_Burkert_02.doc	20/07/04	12-01-07	12-01-07

byte 0: (8 bits integer) current INDEX Input  
 byte 1: (8 bits integer) current INDEX Output  
 bytes 2-9: (8 bytes ASCII) memory batch name (allowed characters: 0..9, a..z, A..Z, space (32 DEC, 20 HEX))  
 bytes 10-11: (2 bytes 16 bit unsigned integer, MSB first) number of batch processes done for this batch memory.  
 bytes 12-13: (2 bytes 16 bit unsigned integer, MSB first) value of the batch safety timer expressed in tenths of seconds.  
 bytes 14-17: (4 bytes 32 bits unsigned long integer, MSB first) batch quantity value expressed in the same units and with the same decimal digits of the volume counters  
 byte 18: (8 bits integer) batch status:

Format of the returned data: 1 byte containing the batch process state:  
**0(dec)** = batch process is correctly terminated (pre-set quantity reached).  
**1(dec)** = batch process is running (the valve is opened and the counters are running).  
**2(dec)** = batch process is suspended (the valve is closed before the pre-set quantity is reached).

**INDEX Input = 42(dec) Batch Parameters**

byte 0: (8 bits integer) current INDEX Input  
 byte 1: (8 bits integer) current INDEX Output  
 byte 2: (8 bits integer) batch status:

Format of the returned data: 1 byte containing the batch process state:  
**0(dec)** = batch process is correctly terminated (pre-set quantity reached).  
**1(dec)** = batch process is running (the valve is opened and the counters are running).  
**2(dec)** = batch process is suspended (the valve is closed before the pre-set quantity is reached).

byte 4: (8 bits integer) menu Measure – cut-off es. 1(dec) / 10 = 0.1%, 255(dec) / 10 → 25.5%  
 byte 5: (8 bits integer) menu Batch - N.samples  
 byte 6: (8 bits integer) menu Batch – Diff.thr  
 bytes 8-9: (2 bytes 16 bit unsigned integer, MSB first) menu Batch – V.com (value expressed in the same units and with the same decimal digits of the volume counters)  
 bytes 10-11: (2 bytes 16 bit unsigned integer, MSB first) menu Batch – V.pre (value expressed in the same units and with the same decimal digits of the volume counters)





byte 2: 0x06 → Command

-----

es.

Reading current type language of the Flow Meter

byte 0: 0x20 = **32(dec)** index input → see **INDEX Input = 32(dec)**

byte 1: 0x0A = **10(dec)** index output

byte 2: 0x07 → Command

-----

es.

Writing the type language in the Flow Meter after the firstly reading of the current language set in the meter

byte 0: 0x20 = **32(dec)** index input → see **INDEX Input = 32(dec)**

byte 1: 0x0A = **10(dec)** index output

byte 2: 0x60 = **96(dec)** Command for set language

byte 3: 0x01 → set italian language in the meter

-----

es.

Set Autozero Calibration

byte 0: 0xXX = index input

byte 1: 0x0A = **10(dec)** index output

byte 2: 0x32 → Command

-----

**INDEX Output = 20(dec) Setting Data and Time in the Flow Meter (only for ML 210)**

byte 0: (8 bits integer) INDEX Input: the value set the type of the Input data

byte 1: (8 bits integer) INDEX Output = 20(dec)

byte 2-5: 32 bits long integer containing the date and time expressed in minutes starting from 01-01-1992 the most significant byte must be sent first.

**INDEX Output = 40(dec) Batch Commands**

For to use the batch function it is firstly necessary to set ON the menu Batch in the Flow Meter ( see the Flow Meter Manual )

byte 0: (8 bits integer) INDEX Input: the value set the type of the Input data

byte 1: (8 bits integer) INDEX Output = 40(dec)

byte 2: (8 bits integer) batch **Command**

**0x10(hex)** Read the batch data memory, the data are returned with the **INDEX Input = 40(dec)**

byte 3: (8 bits integer) batch **Code**

bits 0..4 = number of the batch memory to be read (0..15)

bit 5 = must be set to **zero**

bit 6 = **1** if this batch memory must be set as active batch process, **0** if this batch memory is to be read only.

bit 7 = must be set to **zero**

**0x12(hex)** Set batch with the code in the byte 3

byte 3: (8 bits integer) batch **Code**

Type of command that is possible to insert in the byte 3:

**0x01(hex)** = command for reading the current batch status ( see **INDEX Input = 40(dec) or 42(dec)** for to read the status)

**0x02(hex)** = command for starting the current batch process

**0x03(hex)** = command for suspending the current batch process

**0x04(hex)** = command for resetting the current batch process

**0x13(hex)** Read the batch parameters, the data are returned with the **INDEX Input = 42(dec)**

es. Reading Batch Data:

byte 0: 0x28 = **40(dec)** index input → see **INDEX Input = 40(dec) Batch Data**

byte 1: 0x28 = **40(dec)** index output

byte 2: 0x10 → Command

byte 3: 0x40 → Code for read the first recipe and set it as the current

es. Batch Commands:

byte 0: 0x28 = **40(dec)** index input → see **INDEX Input = 40(dec) Batch Data**

byte 1: 0x28 = **40(dec)** index output

byte 2: 0x12 → Command

byte 3: 0x01 → Code Read Status

0x02 → Code Start Batch

0x03 → Code Stop Batch

0x04 → Code Reset the current Batch process

es. Reading Batch Parameters:

byte 0: 0x2A = **42(dec)** index input → see **INDEX Input = 40(dec) Batch Data**

byte 1: 0x28 = **40(dec)** index output

byte 2: 0x13 → Command

**INDEX Output = 41(dec) Setting Batch Data Memory**

With this index it is possible to insert the batch data memory

byte 0: (8 bits integer) INDEX Input: the value set the type of the Input data  
 byte 1: (8 bits integer) INDEX Output = 40(dec)  
 byte 2: (8 bits integer) batch code:

bits 0..4 = number of the batch memory to be written (0..15)

bit 5 = must be set to **one**

bit 6 = **1** if this batch memory must be set as active batch process, **0** if this batch memory is to be written only.

bit 7 = must be set to **zero**

bytes 4-11: (8 bytes ASCII) memory batch name (allowed characters: 0..9, a..z, A..Z, space (32 DEC, 20 HEX))  
 bytes 12-13: (2 bytes 16 bit unsigned integer, MSB first) number of batch processes done for this batch memory.  
 bytes 14-15: (2 bytes 16 bit unsigned integer, MSB first) value of the safety batch timer expressed in tenths of seconds.  
 bytes 16-19: (4 bytes 32 bits unsigned long integer, MSB first) batch quantity value expressed in the same units and with the same decimal digits of the volume counters.  
 es.

Insertion of the Batch Data ( insertion of the data in the meter is in real time for this type of data ):

byte 0: 0x28 = **40(dec)** index input → see **INDEX Input = 40(dec) Batch Data**

byte 1: 0x29 = **41(dec)** index output

byte 2: 0x60 → set the first memory as the current recipe

byte 3: 0x00 → byte alignment

byte 4: 0x41

byte 5: 0x42

byte 6: 0x43

byte 7: 0x44

byte 8: 0x45

byte 9: 0x46

byte 10: 0x47

byte 11: 0x48 → recipe name: "ABCDEFGH"

byte 12: 0x00

byte 13: 0x00 → 0(dec) value for reset the counter

byte 14: 0x00

byte 15: 0x0F → 15 (dec) safety batch timer

byte 16: 0x00

byte 17: 0x00

byte 18: 0x00

byte 19: 0x01 → 1(dec) batch quantity

---

### **INDEX Output = 42(dec) Setting Batch Parameters**

With this index it is possible to insert the batch parameters



byte 0: (8 bits integer) INDEX Input: the value set the type of the Input data  
 byte 1: (8 bits integer) INDEX Output = 42(dec)  
 byte 2: (8 bits integer) menu Measure – cut-off es.  $1(\text{dec}) / 10 = 0.1\%$ ,  $255(\text{dec}) / 10 \rightarrow 25.5\%$   
 byte 3: (8 bits integer) menu Batch - N.samples  
 byte 4: (8 bits integer) menu Batch – Diff.thr  
 bytes 6-7: (2 bytes 16 bit unsigned integer, MSB first) menu Batch – V.com (value expressed in the same units and with the same decimal digits of the volume counters)  
 bytes 8-9: (2 bytes 16 bit unsigned integer, MSB first) menu Batch – V.pre (value expressed in the same units and with the same decimal digits of the volume counters)  
 es.

Insertion of the parameters value:

byte 0: 0xXX index input  
 byte 1: 0x2A = **42(dec)** index output  
  
 byte 2: 0x01  $\rightarrow$  cut-off ( menu Measure ) = 0.1%  
 byte 3: 0x02  $\rightarrow$  N.samples ( menu Batch ) = 2  
 byte 4: 0x03  $\rightarrow$  Diff.thr ( menu Batch ) = 3%  
 byte 5: 0x00  $\rightarrow$  byte alignment  
  
 byte 6: 0x00  
 byte 7: 0x04  $\rightarrow$  V.com ( menu Batch ) = 00.004  
  
 byte 8: 0x00  
 byte 9: 0x05  $\rightarrow$  V.pre ( menu Batch ) = 00.005

**INDEX Output = 43(dec) Setting Batch Data and Batch Parameters with only this index**

With this index it is possible to insert the batch data and batch parameters at the same time.

byte 0: (8 bits integer) INDEX Input: the value set the type of the Input data  
 byte 1: (8 bits integer) INDEX Output = 43(dec)  
  
 bytes 2-3: (2 bytes 16 bit unsigned integer, MSB first) number of batch processes done for this batch memory.  
 bytes 4-5: (2 bytes 16 bit unsigned integer, MSB first) value of the safety batch timer expressed in tenths of seconds.  
 bytes 6-9: (4 bytes 32 bits unsigned long integer, MSB first) batch quantity value expressed in the same units and with the same decimal digits of the volume counters.  
 byte 10: (8 bits integer) menu Measure – cut-off es.  $1(\text{dec}) / 10 = 0.1\%$ ,  $255(\text{dec}) / 10 \rightarrow 25.5\%$   
 byte 11: (8 bits integer) menu Batch - N.samples  
 byte 12: (8 bits integer) menu Batch – Diff.thr  
 bytes 14-15: (2 bytes 16 bit unsigned integer, MSB first) menu Batch – V.com (value expressed in the same units and with the same decimal digits of the volume counters)  
 bytes 16-17: (2 bytes 16 bit unsigned integer, MSB first) menu Batch – V.pre (value expressed in the same units and with the same decimal digits of the volume counters)

NOTE: The only recipe that is possible to use for this index **INDEX Output = 43(dec)** is the 00 in the

meter and the name utilized for this recipe is PROFIBUS.

es.

Insertion of the parameters value:

byte 0: 0xXX index input

byte 1: 0x2A = **43(dec)** index output

byte 2: 0x00

byte 3: 0x00 → 0(dec) value for reset the counter

byte 4: 0x00

byte 5: 0x0F → 15 (dec) safety batch timer

byte 6: 0x00

byte 7: 0x00

byte 8: 0x00

byte 9: 0x01 → 1(dec) batch quantity

byte 10: 0x01 → cut-off ( menu Measure ) = 0.1%

byte 11: 0x02 → N.samples ( menu Batch ) = 2

byte 12: 0x03 → Diff.thr ( menu Batch ) = 3%

byte 13: 0x00 → byte alignment

byte 14: 0x00

byte 15: 0x04 → V.com ( menu Batch ) = 00.004

byte 16: 0x00

byte 17: 0x05 → V.pre ( menu Batch ) = 00.005

### **3.1 DPV1 ACYCLIC DATA EXCHANGE**

The software of the module implement the DP standard/DPV1 functionality for communication of the slave with a Class 1 (parameterization) Master and a Class 2 Master.

With the **Acyclic Service** of data exchange and with the **Acyclic Slot** and **Acyclic Index** it is possible to send and return configuration parameters from the Flow meter with a Class 1 (parameterization) Master and a Class 2 Master.

For the communication with a Class C1 master the **Slot** is **ignored**.


Only **Slot 0** is used for the **Acyclic Service** of data exchange for the communication with a Class 2 master.

### **3.2 Acyclic Read Data from the Flow Meter**

For return data with the Acyclic Read Service and a Class 1 Master the **Acyclic Slot** is ignored and every value is not checked in the slave. The only value used for select the data is the **Acyclic Index**.

If the master is a Class 2 master, it is necessary insert the value of the **Acyclic Slot** = 0 and with the value of the **Acyclic Index** there is the selection of the data returned.

Below there is the description list of the type of the bytes returned with the different value inserted in the **Acyclic Index**.

	TECHNICAL MANUAL		Rev.	Page	43/52
	PROFIBUS DP MODULE- DATA LINK DESCRIPTION		03	Filling By	Checked By
			Document Data	Date	Date
	File name	ProfibusDP_emanual_Burkert_02.doc	20/07/04	12-01-07	12-01-07

### **Acyclic Index Read = 20(dec) Static Process Data of the Flow Meter**

bytes 0-3: (32 bit single precision IEEE floating point, MSB first) Flow rate scale range in t.u.  
bytes 4-8: (5 bytes ASCII) flow rate measurement unit  
bytes 9-11: (3 bytes ASCII) measurement unit of the counters  
byte 12: (8 bits integer) number of decimals after the point for totalizers display  
byte 13: (8 bits integer) number of decimals after the point for flow rate display  
bytes 14-17: (32 bit long integer, MSB first) clock expressed in minutes starting from 01-01-1992  
byte 18: (8 bits integer) measurement samples per second (Hz)

byte 20=10(dec)	measurement frequency = 10 Hz
byte 20=20(dec)	measurement frequency = 20 Hz
byte 20=50(dec)	measurement frequency = 50 Hz
byte 20=80(dec)	measurement frequency = 80 Hz
byte 20=150(dec)	measurement frequency = 150 Hz
byte 20=44(dec)	measurement frequency = 300 Hz
byte 20=144(dec)	measurement frequency = 400 Hz

byte 19: (8 bits integer) current value of Data Type ( see the Sections Data Type and the relative Configuration )

**NOTE:** The values in the "32 bits single precision IEEE floating point" format are floating point numbers which can be represented during writing by any decimal digits. To keep the same numerical format visible on the instrument display however, it is necessary to calculate the decimal figures with a rather complex algorithm which takes account of instrument precision, flow rate measurement unit, etc. For this purpose and to avoid useless calculations, the number of decimals to use to represent the flow rate values is supplied separately (byte position 13).

**NOTE:** The counters are expressed with a 32 bit integer. The «counter decimal figures» parameter, indicates the point position starting from the right: 0 = no decimal, 1=1 decimal figure, and so on (byte position 12).

**NOTE:** The date and time are expressed with a 32 bit integer containing the number of minutes elapsed since 01-01-1992. To calculate the date starting from this number, see the programming examples in the section 4.0 PROGRAMMING ALGORITMS of this manual.

### **Acyclic Index Read = 21(dec) Slave Data**

byte 0: (8 bits integer) enabled/disabled the control of the flow rate by thresholds and sending Diagnostic message (see Section 1.5 Slave Diagnostic Alarms )

bit 0 =1 if enabled the control of the flow rate by thresholds and sending Diagnostic message

byte 1: (8 bits integer) enabled/disabled Alarm message when the control of the flow rate by thresholds is active

bit 0 =1 if enabled the sending of Alarm message when the control of the flow rate by thresholds is active

byte 2: (8 bits integer) slave address

byte 3: (8 bits integer) pno ident number high

byte 4: (8 bits integer) pno ident number low

byte 5: (8 bits integer) DPC31 ASIC Step

bit 0 =1 if DPC31 B

byte 6: (8 bits integer) MAC State Offline / Passive Idle

bit 0 =1 if the MAC is in Passive Idle

byte 7: (8 bits integer) watchdog state

bit 0-1 = 00 --> Baud Search

01 --> Baud Control

10 --> DP\_Control

byte 8: (8 bits integer) baud rate

bit 0-3 = 0000 --> 12 MBd

0001 --> 6 MBd

0010 --> 3 MBd

0011 --> 1.5 MBd

0100 --> 500 kBd

0101 --> 187.5 kBd

0110 --> 93.75 kBd

0111 --> 45.45 kBd

1000 --> 19.2 kBd

1001 --> 9.6 kBd

byte 9: (8 bits integer) DP state machine

bit 0-1 = 11 --> Data\_Exchange

**Acyclic Index Read = 22(dec) Internal Data**

byte 0: (8 bits integer) slave software version (major number)

byte 1: (8 bits integer) slave software version (minor number)

byte 2: (8 bits integer) current language set in the Flow meter

Format of the returned data:

0(dec) = English

1(dec) = Italian

2(dec) = French

3(dec) = Spanish

**NOTE:** For reading the type of language it is first necessary to send to the slave the relative command, see the **Acyclic Index Read = 10** (below in this section) and then read the type of language in the byte 2.

byte 3-8: (8 bits integer) device name («ML 210», «ML M3F»)

byte 9: (8 bits integer) Flow meter software version (major number)

byte 10: (8 bits integer) Flow meter software version (minor number)

byte 11-12: (8 bits integer) (16 bits unsigned integer, byte 13=MSB):

hardware / software enabling flags, bit 0 = LSB, bit 15 = MSB:

bits	Functions ML 210/M3F
0-2	current access level (0..7)
3	channel 1 impulses in use
4	channel 2 impulses in use
5	channel 1 frequency in use
6	channel 2 frequency in use
7	scale range 2 in use
8	specific weight in use
9	additional output 3 present
10	additional output 4 present
11	output 4..20 mA 2 present
12	RS232 present
13	batching functions active
14	output 4..20 mA 2 present
15	RS485 present

**Acyclic Index Read = 30(dec) Batch Functions**

byte 0: (8 bits integer) batch status:

Format of the returned data: 1 byte containing the batch process state:

**0(dec)** = batch process is correctly terminated (pre-set quantity reached).

**1(dec)** = batch process is running (the valve is opened and the counters are running).

**2(dec)** = batch process is suspended (the valve is closed before the pre-set quantity is reached).

bytes 1-8: (8 bytes ASCII) memory batch name (allowed characters: 0..9, a..z, A..Z, space (32 DEC, 20 HEX))

bytes 9-10: (2 bytes 16 bit unsigned integer, MSB first) number of batch processes done for this batch memory.

bytes 11-13: (2 bytes 16 bit unsigned integer, MSB first) value of the safety batch timer expressed in tenths of seconds.

bytes 14-17: (4 bytes 32 bits unsigned long integer, MSB first) batch quantity value expressed in the same units and with the same decimal digits of the volume counters.

**Acyclic Index Read = 50(dec) Data Logger DATA**

byte 0: (8 bits integer) check **DATA** on Internal Data Logger

bit 0 =1 Presence of data in the Internal Data Logger

**Acyclic Index Read = 51(dec) Data Logger DATA**

byte 0: (8 bits integer) number of record requested

byte 1: (8 bits integer) total number of records present in the memory

bytes 2-5: (32 bits long integer, MSB first) data saving time and date expressed in minutes starting from 01-01-1992

bytes 6-9: (32 bits long integer, MSB first) data counted +

bytes 10-13: (32 bits long integer, MSB first) data counted -

bytes 14-17: (32 bit single precision IEEE floating point, MSB first) flow rate in technical unit

bytes 18-20: (3 bytes ASCII) counter measurement unit

byte 21: (8 bits integer) number of decimal figures after the point for counter display  
 bytes 22-26: (5 bytes ASCII) flow rate measurement unit  
 byte 27: (8 bits integer) number of decimals after the point for flow rate display

**Acyclic Index Read = 52(dec) Data Logger EVENTS**

byte 0: (8 bits integer) check **EVENTS** on Internal Data Logger  
 bit 0 = 1 Presence of Events in the Internal Data Logger

**Acyclic Index Read = 53(dec) Data Logger EVENTS**

byte 0: (8 bits integer) number of record requested  
 byte 1: (8 bits integer) total number of records present in the memory  
 es 2-5: (32 bits long integer, MSB first) data saving time and date expressed in minutes starting from 01-01-1992  
 bytes 6-9: (32 bits long integer, MSB first) Events

if bytes 6-9 = 0(dec) then NO ALARMS

if bytes 6-9 = 196863(dec) then SYSTEM STARTUP

if bytes 6-9 = 262143(dec) then SYSTEM STARTUP

else

- bit 0 = 1 Batch alarm
- bit 1 = 1 Alarm Max Flow
- bit 2 = 1 Alarm Min Flow
- bit 3 = 1 Alarm Measure > Full Scale
- bit 4 = 1 Alarm Pulse/Frequency > Full Scale
- bit 5 = 1 Input Noisy
- bit 6 = 1 Empty Pipe
- bit 7 = 1 Excitation Fail
- bit 8 = 1 not used
- bit 9 = 1 not used
- bit 10 = 1 not used
- bit 11 = 1 not used
- bit 12 = 1 not used
- bit 13 = 1 not used
- bit 14 = 1 not used
- bit 15 = 1 not used
- bit 16 = 1 Current Loop Open
- bit 17 = 1 Power Supply Fail

**Acyclic Index Read = 54(dec) Data Logger MIN MAX measurement unit**

bytes 0-4: (5 bytes ASCII) flow rate measurement unit  
 byte 5: (8 bits integer) number of decimals after the point for flow rate display

**Acyclic Index Read = 55(dec) Data Logger MIN MAX data**

bytes 0-4: (32 bit single precision IEEE floating point, MSB first) flow rate MAX  
 bytes 5-8: (32 bit single precision IEEE floating point, MSB first) flow rate MI

**Acyclic Index Read = 100(dec) Current value of the High Alarm Threshold for the Flow rate Control with Diagnostic and Alarm**

bytes 0-3: High Alarm Threshold (32 bit single precision IEEE floating point, MSB first) Flow rate in technical unit

**Acyclic Index Read = 101(dec) Current value of the High Warning Threshold for the Flow rate Control with Diagnostic and Alarm**

bytes 0-3: High Warning Threshold (32 bit single precision IEEE floating point, MSB first) Flow rate in technical unit

**Acyclic Index Read = 102(dec) Current value of the Low Warning Threshold for the Flow rate Control with Diagnostic and Alarm**

bytes 0-3: Low Warning Threshold (32 bit single precision IEEE floating point, MSB first) Flow rate in technical unit

**Acyclic Index Read = 103(dec) Current value of the Low Alarm Threshold for the Flow rate Control with Diagnostic and Alarm**

bytes 0-3: Low Alarm Threshold (32 bit single precision IEEE floating point, MSB first) Flow rate in technical unit

**3.2. Acyclic Write Data to the Flow Meter**

For the insertion of the data with the Acyclic Read Service and a Class 1 Master the **Acyclic Slot** is ignored and every value is not checked in the slave. The only value used for select the type of data inserted is the **Acyclic Index**.

If the master is a Class 2 master, it is necessary insert the value of the **Acyclic Slot** = 0 and with the value of the **Acyclic Index** there is the indication to the slave of the type of data of the data inserted.

Below there is the description list of the type of the bytes that there is possible to insert with the different value inserted in the **Acyclic Index**.

**Acyclic Index Write = 0(dec) Setting of the value of the Data Type**

byte 0: (8 bits integer) Data Type ( see the Sections **Data Type** and the relative Configuration )

**Acyclic Index Write = 10(dec) Flow Meter Commands**

byte 0: (8 bits integer) command type


Type of command that is possible to insert in the byte 0:

Byte 0 = **0x06(hex)** command for reset the Totalizers of the Flow Meter ( see the Flow Meter Manual and the relative menu for to enable which totalizers to reset by this command).

Byte 0 = **0x07(hex)** command for read the current Language used in the Flow Meter:  
The information is returned by the byte 2 of the Input data with Acyclic Index 22.

Byte 0 = **0x08(hex)** command for Reset the slave module. After 5 seconds the module return in the data exchange

Byte 0 = **0x32(hex)** command for Autozero Calibration

	TECHNICAL MANUAL		Rev.	Page	48/52
	PROFIBUS DP MODULE- DATA LINK DESCRIPTION		03	Filling By	Checked By
			Document Data	Date	Date
	File name	ProfibusDP_emanual_Burkert_02.doc	20/07/04	12-01-07	12-01-07

### **Acyclic Index Write = 11(dec) Command for Set Language in the Flow Meter**

Command for set the type of language in the meter ( Note: It is firstly necessary to read the type of the language of the meter, for to set it with this command )

Byte 0 = type of language:

- 0x00 = 0(dec) = English
- 0x01 = 1(dec) = Italian
- 0x02 = 2(dec) = French
- 0x03 = 3(dec) = Spanish

### **Acyclic Index Write = 20(dec) Setting Data and Time in the Flow Meter**

byte 0-4: 32 bits long integer containing the date and time expressed in minutes starting from 01-01-1992 the most significant byte must be sent first.

**NOTE:** The date and time are expressed with a 32 bit integer containing the number of minutes elapsed since 01-01-1992. To calculate the date starting from this number, see the programming examples in the section 4.0 PROGRAMMING ALGORITMS of this manual.

### **Acyclic Index Write = 21(dec) Enable/Disable Flow rate Control with Diagnostic message and Alarms**

byte 0: (8 bits integer) enable/disable Flow meter control with thresholds and Diagnostic message

Type of command that is possible to insert in the byte 0:

- 0** = Control of the Flow rate disabled
- 1** = Control of the Flow rate enabled, if the value of the Flow rate exceed one of the four thresholds the slave send the relative Diagnostic message

byte 1: (8 bits integer) enable/disable Process Alarm message

Type of command that is possible to insert in the byte 1:

- 0** = Alarms disabled
- 1** = Alarms enabled, if the value of the Flow rate exceed one of the four thresholds the slave send Process Alarm message

### **Acyclic Index Write = 30(dec) Batch Commands**

For to use the batch function it is firstly necessary to set ON the menu Batch in the Flow Meter ( see the Flow Meter Manual )

byte 0: (8 bits integer) batch **Command**

- 0x10(hex)** Read the batch data memory, the data are returned with the **Acyclic Index Read 30**
- 0x11(hex)** Write the batch data memory, the data are firstly inserted with **Acyclic Index Write 31**
- 0x12(hex)** Set batch with the code in the byte 1

byte 1: (8 bits integer) batch **Code**

Type of command that is possible to insert in the byte 1:

- 0x01(hex)** = command for reading the current batch status ( see **Acyclic Index Read 30** for to read



the status)

**0x81(hex)** = command for starting the current batch process

**0x82(hex)** = command for suspending the current batch process

**0x83(hex)** = command for resetting the current batch process

**Acyclic Index Write = 31(dec) Setting Batch Data Memory**

With this index it is possible to insert the batch data memory, and then with the Acyclic Index Write = 30 and the batch command 0x11(hex) the data is written in the Flow meter

byte 0: (8 bits integer) batch code:

bits 0..4 = number of the batch memory to be written (0..15)

bit 5 = must be set to **one**

bit 6 = **1** if this batch memory must be set as active batch process, **0** if this batch memory is to be written only.

bit 7 = must be set to **zero**

bytes 1-8: (8 bytes ASCII) memory batch name (allowed characters: 0..9, a..z, A..Z, space (32 DEC, 20 HEX))

bytes 9-10: (2 bytes 16 bit unsigned integer, MSB first) number of batch processes done for this batch memory.

bytes 11-12: (2 bytes 16 bit unsigned integer, MSB first) value of the safety batch timer expressed in tenths of seconds.

bytes 13-16: (4 bytes 32 bits unsigned long integer, MSB first) batch quantity value expressed in the same units and with the same decimal digits of the volume counters.

**Acyclic Index Write = 50(dec) Data Logger Commands**

With this index it is possible to insert the number of record requested for to return the data in the internal Data Logger, and then with the Acyclic Indexes Read = 50,51...55 it is possible read the data returned

byte 0: (8 bits integer) number of record requested

byte 1: (8 bits integer) command type

- byte 1 = 0x20(hex) command for check presence of DATA in the Internal Data Logger
- byte 1 = 0x21(hex) command for reading record of DATA from the Internal Data Logger
- byte 1 = 0x22(hex) command for erasing DATA in the Internal Data Logger
- byte 1 = 0x23(hex) command for check presence of EVENTS in the Internal Data Logger
- byte 1 = 0x24(hex) command for reading record of EVENTS in the Internal Data Logger
- byte 1 = 0x25(hex) command for erasing EVENTS in the Internal Data Logger


byte 1 = 0x26(hex) command for reading measurement unit of the MIN MAX in the Internal Data Logger

byte 1 = 0x27(hex) command for reading values MIN MAX from the Internal Data Logger

byte 1 = 0x28(hex) command for erasing MIN MAX values in the Internal Data Logger

**Acyclic Index Write = 100(dec) Setting the value of the High Alarm Threshold for the Flow rate Control with Diagnostic and Alarm**

bytes 0-3: High Alarm Threshold (32 bit single precision IEEE floating point, MSB first) Flow rate in technical unit

	TECHNICAL MANUAL		Rev.	Page	50/52
	PROFIBUS DP MODULE- DATA LINK DESCRIPTION		03	Filling By	Checked By
			Document Data	Date	Date
	File name	ProfibusDP_emanual_Burkert_02.doc	20/07/04	12-01-07	12-01-07

**Acyclic Index Write = 101(dec) Setting the value of the High Warning Threshold for the Flow rate Control with Diagnostic and Alarm**

bytes 0-3: High Warning Threshold (32 bit single precision IEEE floating point, MSB first) Flow rate in technical unit

**Acyclic Index Write = 102(dec) Setting the value of the Low Warning Threshold for the Flow rate Control with Diagnostic and Alarm**

bytes 0-3: Low Warning Threshold (32 bit single precision IEEE floating point, MSB first) Flow rate in technical unit

**Acyclic Index Write = 103(dec) Setting the value of the Low Alarm Threshold for the Flow rate Control with Diagnostic and Alarm**

bytes 0-3: Low Alarm Threshold (32 bit single precision IEEE floating point, MSB first) Flow rate in technical unit

**4.1 PROGRAMMING ALGORITHMS**


The following algorithms can be applied to IBM™ compatible personal computers. The programs demonstrated are written in BASIC and are supplied as examples only. The program lines are highlighted in bold face and the comments are in italics.

**4.2 NUMERICAL CONVERSION OF THE DATA RECEIVED**

The numerical data contained in the blocks received must be converted into a format compatible with the PC format. The arrangement of bytes used to represent numerical values in IBM™ compatible PCs is in fact inverted with respect to the one in the data received. The following programs illustrate several of the ways that can be used for these conversions.

We presume that there are bytes to convert in the variable - type Bytes\$ string.

- Converting a byte or a group of **ASCII** characters: no conversion is necessary, the data are already in the exact order and format.
- Converting a **16 bit signed word**: Bytes contains the two bytes that constitute the 16 bit number to convert. One 16 bit signed word can represent values from -32768 to 32767. The instructions to use may be the following:  
**NumIntSgn%=CVI(MID\$(Bytes\$,2,1)+MID\$(Bytes\$,1,1))**
- Converting a **16 bit unsigned word**: Bytes contains the two bytes that constitute the 16 bit number to convert. One 16 bit unsigned word can represent values from 0 to 65535. Not all BASIC versions can correctly manage this type of variable; we therefore recommend assigning the converted value to a long integer 32 bit type variable or to a single precision floating point. The instructions to use (for the long integer) are:  
**NumInt&=CVL(MID\$(Bytes\$,2,1)+MID\$(Bytes\$,1,1)+CHR\$(0)+CHR\$(0))**
- Converting a 32 bit long integer: Bytes contains the four bytes that constitute the 32 bit number to convert. A 32 bit integer may represent values from -2147483648 to 2147483647. The instructions to use may be the following:  
**NumInt&=CVL(MID\$(Bytes\$,4,1)+MID\$(Bytes\$,3,1)+MID\$(Bytes\$,2,1)+MID\$(Bytes\$,1,1))**

	TECHNICAL MANUAL		Rev.	Page	51/52
	PROFIBUS DP MODULE- DATA LINK DESCRIPTION		03	Filling By	Checked By
			Document Data	Date	Date
	File name	ProfibusDP_emanual_Burkert_02.doc	20/07/04	12-01-07	12-01-07

- Converting a 32 bit decimal value (**32 bit floating point IEEE standard**): Bytes contains the four bytes that constitute the 32 bit number to convert. A 32 bit decimal number can represent values from  $\pm 1.4E-45$  to  $\pm 3.4E+38$ . The instructions to use may be the following:

**NumFit! = CVS(MID\$(Bytes\$,4,1)+MID\$(Bytes\$,3,1)+MID\$(Bytes\$,2,1)+MID\$(Bytes\$,1,1))**

**NOTE:** Some old versions of BASIC do not support the **CVL** instruction and the 32 bit integer format.

#### **4.3 DATE CONVERSION FROM TOTAL MINUTES TO DAY/MONTH/YEAR HOURS:MINUTES**

The date read by the Millennium series instruments is expressed in minutes starting from 01/01/1992. The following program executes the conversion of the value in minutes contained in the 32 bit integer variable MinTot and prints it in the day/month/year hours:minutes format.

Note the use of suffixes "%" and "&" to respectively indicate the 16 and 32 bit variables. Both types are necessary to optimise the calculations and to contain the values which cannot be expressed in just 16 bits.

*10 REM Length in days for each month in the year*

**20 DATA 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31**

*30 REM Defines a group of twelve elements to memorise the length in days of each month in*

*40 REM the year*

**50 DIM LengthMonths%(12)**

*60 REM Initialises each element of the group*

**70 FOR i% = 1 TO 12: READ LengthMonths%(i%): NEXT i%**

*80 REM Calculates the total number of days*

**90 Days& = INT(MinTot& / 1440)**

*100 REM Calculates the minutes*

**110 n& = MinTot& MOD 1440**

**120 Minutes& = n& MOD 60**

*130 REM Calculates the hours*

**140 Hour& = INT(n& / 60)**

*150 REM Calculates the number of leap years. 4 years are 1460 days + 1*

**160 n& = INT(Days& / 1461)**

*170 REM Calculates the remaining days, not exact multiples for a period of 4 years*

**180 m& = Days& MOD 1461**

*190 REM Determines the p and q correction values for leap years*

**200 IF m& = 0 THEN q% = 1 ELSE q% = 0**

**210 IF m& > 365 THEN p% = 1 ELSE p% = 0**

*220 REM Calculates the total number of years*

**230 Years& = INT(n& \* 4 + (m& - 1 + q%) / 365)**

*240 REM Calculates the remaining days, not exact multiples of one year*

**250 Days& = Days& - Years& \* 365 - n& - p%**

*260 REM Determines if the year calculated is a leap year, in this case m = 0*

**270 m& = Years& AND 3**

*280 REM Calculates the number of the month and removes the sum of the preceding*

*290 REM months from the remaining days.*

**300 FOR i% = 1 TO 12**

**310 Months% = LengthMonths%(i%)**

*320 REM If it is February and the year is a leap year, add one day*

**330 IF i% = 2 AND m& = 0 THEN Months% = Months% + 1**

*340 REM Continues the cycle until the number of days remaining is lower than the*

*350 REM length of the month compared*


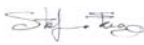
**360 IF Days& >= Months% THEN Days& = Days& - Months%: ELSE GO TO 380**

**370 NEXT**

*380 REM Assigns the number of the month thus calculated to the variable*

**390 Months% = i%**

*400 REM Adjusts the value of the days calculated*

	TECHNICAL MANUAL		Rev.	Page	52/52
	PROFIBUS DP MODULE- DATA LINK DESCRIPTION		03	Filling By	Checked By 
	File name	ProfibusDP_emanual_Burkert_02.doc	Document Date	Date	Date
			20/07/04	12-01-07	12-01-07

**410 Days& = Days& + 1**

*420 REM Determines the last two digits of the year*

**430 Years& = (Years& + 92) MOD 100**

*440 REM Finally prints the complete date*

**450 PRINT Days&; "/" ; Months% ; "/" ; Years&; " " ; Hours&; ":" ; Minutes&**

**NOTE:** Some old versions of BASIC do not support the 32 bit integer format.

#### **4.4 DATE CONVERSION FROM DAY/MONTH/YEAR HOURS:MINUTES TO TOTAL MINUTES**

The date we can set serially in the Millennium series instruments must be expressed in total minutes starting from 01/01/1992. The program that performs the conversion from a string of 14 characters (AsciiData) in the day/month/year hours:minutes format, transforms it into 4 formatted characters ready to be sent to the serial line (MinTot).

Note the use of suffixes "%" and "&" to respectively indicate the 16 and 32 bit variables. Both types are necessary to optimise the calculations and to contain the values which cannot be expressed in just 16 bits.

*10 REM Length in days for each month of the year*

**20 DATA 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31**

*30 REM Defines a group of twelve elements to memorise the length in days of each month in*

*40 REM the year*

**50 DIM LengthMonths%(12)**

*60 REM Initialises each element of the group*

**70 FOR i% = 1 TO 12: READ LengthMonths%(i%): NEXT i%**

*80 REM Takes the number of years starting from 1992*

**90 Years% = (VAL(MID\$(AsciiData\$, 7, 2)) + 8) MOD 100**

*100 REM Determines if the year contained in the entry string is a leap year, in this case, m = 0*

**110 m% = Years% AND 3**

*120 REM Takes the number of the month from the entry string*

**130 Months% = VAL(MID\$(AsciiData\$, 4, 2))**

*140 REM Calculates all the days of the year corresponding to the date of the entry string*

**150 Days& = VAL(MID\$(AsciiData\$, 1, 2))**

**160 FOR i% = 1 TO Mesi% - 1**

**170 Days& = Days& + LengthMonths%(i%)**

*180 REM If it is February and the year is a leap year, it adds one day*

**190 IF i% = 2 AND m% = 0 THEN Days& = Days& + 1**

**200 NEXT i%**

*210 REM Adds a day if the year of the entry string is not a leap year*

**220 IF m% <> 0 THEN Days& = Days& + 1**

*230 REM Calculates the total number of days corresponding to the entry date*

**240 Days& = Days& + (Years% \* 365) + INT(Years% / 4)**

*250 REM Takes the number of hours and minutes from the entry date*

**260 Hours& = VAL(MID\$(AsciiData\$, 10, 2))**

**270 Minutes& = VAL(MID\$(AsciiData\$, 13, 2))**

*280 REM Calculates the total number of minutes to the entry date*

**290 Minutes& = Minutes& + (Days& \* 1440) + (Hours& \* 60)**

*300 REM Finally formats the bytes that represent the number in the right order for sending*

**310 MinTot\$ = MKL\$(Minutes&)**

**320**

**MinTot\$**

**=**

**MID\$(MinTot\$,4,1)+MID\$(MinTot\$,3,1)+MID\$(MinTot\$,2,1)+MID\$(MinTot\$,1,1)**

**NOTE:** Some old versions of BASIC do not support the MKL instruction and the 32 bit integer format.