F600 Series – Modbus RTU
Manual

Table of Contents

Safety advisory / Warranty

Preamble

Hardware installation

Configuration of the ATEQ device (slave)

Configuration of the master

Frame construction

Functional description of an ATEQ device

ATEQ Manufacturer Plants – Measurement Solution, Global Leader

📍	🌐	☎
ATEQ 15, rue des Dames, Z.I. 78340 LES CLAYES-SOUS-BOIS FRANCE	info@ateq.com ateq.com	T.: +33 1 30 80 1020 F.: +33 1 30 54 1100
ATEQ K.K. 3 – 41 ATEQ Building, Ikehata Chiryu-city, Aichi-pref JAPAN	info@ateq.co.jp ateq.co.jp	T.: +81 566-84-4670 F.: +81 566-84-4680
ATEQ China 98 Jian Peng Lu Shanghai CHINA	shanghai@ateq.com.cn ateq.com.cn	T.: +86 21 6763 9508 F.: +86 21 6763 9528
ATEQ SYSTEMS ANALYSIS TAIWAN CO., LTD. NO. 3, LAN 223, San Jia Dong Street 40642, TAICHUNG TAIWAN	ateqtaiwan@ateq.com.tw ateq.com.tw	T.: +886 4 2437 5278 F.: +886 4 2437 3675
ATEQ CORP. 35980 Industrial Road Suite L Livonia MI 48150 UNITED STATES	leaktest@atequsa.com atequsa.com	T.: +1 734-838-3100 F.: +1 734-838-0644

ℹ We continuously work on improving our products. This is why inforamtion contained in this manual, the device and the technical specifications may be modified without prior notification.

ℹ Pictures and figures in this manual are non contractual

Safety advisory / Warranty

GOOD PRACTICES AND SAFETY INSTRUCTIONS

Safety recommendations

If the device is supplied with 100 / 240 V AC, it is mandatory to connect it to the ground with a good link to the ground, to protect against electric hazard or electrocution.

It is dangerous to change the status of the outputs.
They can control power actuators or other equipment (mechanical, pneumatic, hydraulic, electrical or other) which can cause serious personal injury and damage to surrounding material.

For safety and quality measurement reasons, it is important, before powering on the device, to ensure that it is air supplied with a minimum operating pressure (0.6 MPa ± 15%).

Recommendations for the test environment

Keep the test area as clean as possible.

Recommendations for operators

ATEQ recommends that the operators who use the devices have training and a level of qualification that correspond to the job to perform.

General recommendations

- Read the user manual before using the device.
- All electrical connections to the device must be equipped with safety systems (fuses, circuit breakers, etc.) adapted to the needs and in accordance with the applicable standards and rules.
- To avoid electromagnetic interference, electrical connections to the device must be shorter than 2 meters.
- Power supply plug must be grounded.
- Disconnect the device from the mains before performing any maintenance work.
- Shut off the compressed air supply when working on the pneumatic assembly.
- Do not open a connected device.
- Avoid splashing water on the device.

ATEQ is at your disposal for any information concerning the use of the device under maximum safety conditions.
We draw your attention to the fact that ATEQ cannot be held responsible for any accident related to a misuse of the measuring instrument, the workstation or non-compliance of the installation with safety rules.
In addition, ATEQ declines any responsibility for the calibration or the fitting of their instruments that is not done by ATEQ.
ATEQ also declines any responsibility for any modification (program, mechanical or electrical) of the device done without their written consent.

Preamble

INTRODUCTION

This manual intends to help you for the configuration and the use of your ATEQ F600 device on the Modbus RTU network.

For more information on your ATEQ equipment, refer to the Quick Start Manual.

BASIC NOTIONS

The numerical values used in the ATEQ device are coded on a **Long** format.

> ATEQ devices are configured in Little Endian format. It means that the Least Significant Byte is sent first on the network.

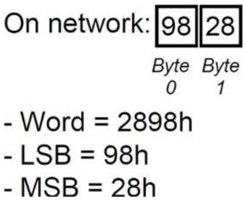
Word

A word is a 16-bit data. It is coded with two bytes (8bits):

- The first byte is the Least Significant Byte (**LSB**)
- The second byte is the Most Significant Byte (**MSB**)

Example of a word:

> Reminder: "**h**" indicates a hexadecimal code, "**(d)**" indicates a decimal code.



On network: 98 28

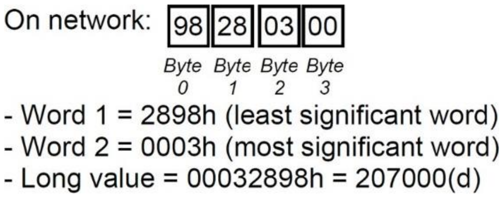
Byte 0 Byte 1

- Word = 2898h
- LSB = 98h
- MSB = 28h

Long format (Signed Double word)

A **Long** format data is coded with two words (of 16 bits).
In the memory range of the ATEQ device or when they are transmitted, both words are coming in the following order:

- The first word is the least significant word
- The second word is the most significant word
- Example of a **Long** format:



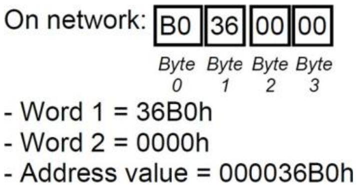
On network: 98 28 03 00

Byte 0 Byte 1 Byte 2 Byte 3

- Word 1 = 2898h (least significant word)
- Word 2 = 0003h (most significant word)
- Long value = 00032898h = 207000(d)

Address value

All address values are treated with the **Long** format.
Example – address of the "millibar" unit in the Unit table (see Unit table):



On network: B0 36 00 00

Byte 0 Byte 1 Byte 2 Byte 3

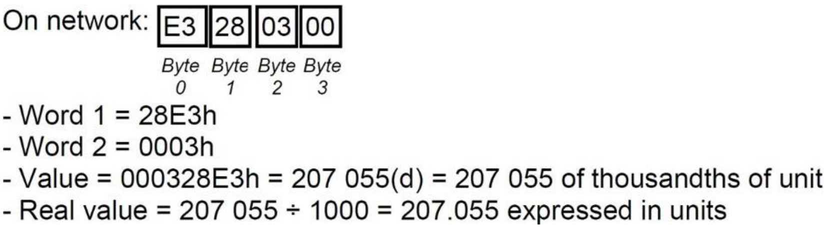
- Word 1 = 36B0h
- Word 2 = 0000h
- Address value = 000036B0h

Numerical value

All the numerical values are treated with the **Long** format with fixed comma (10^{-3}).
Thus, their value is expressed in thousandths of unit. So, this value must be multiplied by 1000 to get the value in units.
For example, a value of 207055 represents 207.055. So, any numerical value must be divided by 1000 to get the real value:

- 207.055 = 207055 ÷ 1000

Example – Pressure:



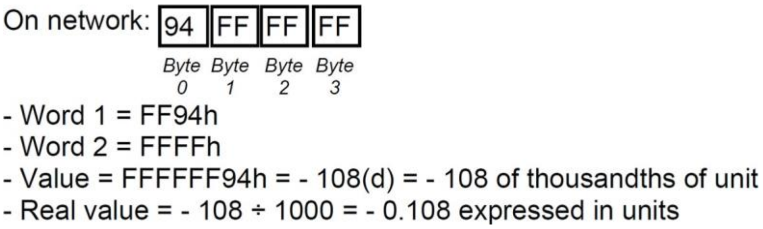
On network:

E3	28	03	00
Byte 0	*Byte 1*	*Byte 2*	*Byte 3*

- Word 1 = 28E3h
- Word 2 = 0003h
- Value = 000328E3h = 207 055(d) = 207 055 of thousandths of unit
- Real value = 207 055 ÷ 1000 = 207.055 expressed in units

Negative numerical value

All the negative numerical values are treated with **Signed long** format with fixed comma (10^{-3}).
Thus, they must be multiplied by 1000 to get the value in units.
Example – Leak value (signed long):



On network:

94	FF	FF	FF
Byte 0	*Byte 1*	*Byte 2*	*Byte 3*

- Word 1 = FF94h
- Word 2 = FFFFh
- Value = FFFFFF94h = - 108(d) = - 108 of thousandths of unit
- Real value = - 108 ÷ 1000 = - 0.108 expressed in units

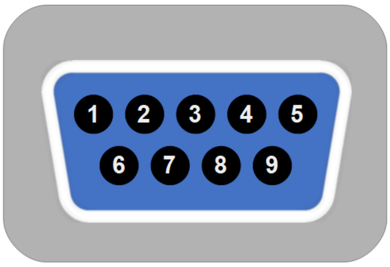
Hardware installation

HARDWARE CONFIGURATION

Connect your ATEQ equipment to the Modbus RTU network using its Modbus RTU connectors and compatible cables.
Your device has one Modbus RTU connector.

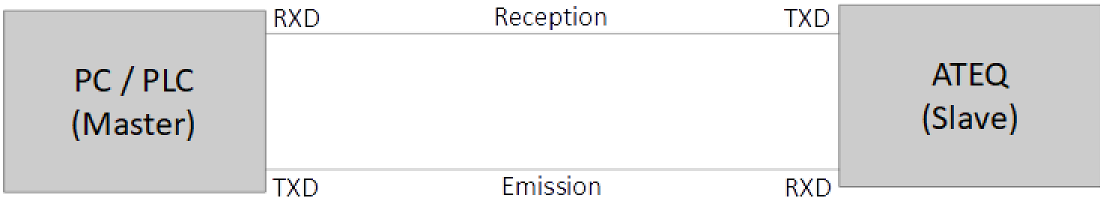
For more information on your ATEQ equipment, refer to the Quick Start Manual.

Modbus RTU connector – 9 pins male connector

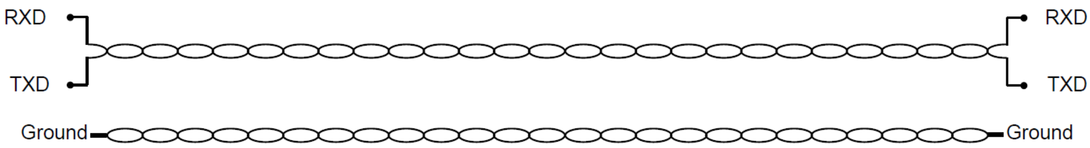


Pin number	Signal
1	-
2	RXD (receive data)
3	TXD (transmit data)
4	-
5	Ground
6	-
7	RTS (request to send)
8	CTS (clear to send)
9	-

Architecture of the Modbus RTU network



The network is built on the basis of a cable composed of two pairs of entwined and shielded wires. One pair is for the signals and the other is for the ground.



Configuration of the ATEQ device (slave)

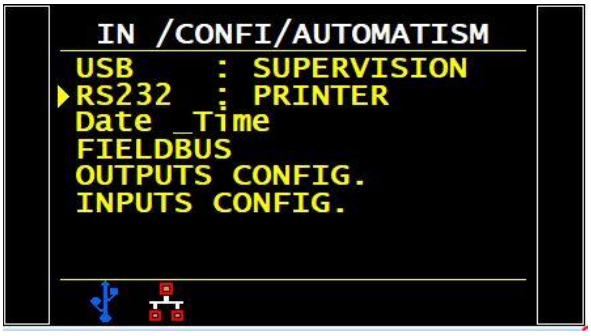
Use this procedure to configure your device.

This configuration can be done with the front panel of your ATEQ device

The Modbus RTU configuration on an ATEQ device is **always 8 bits long with one stop bit**.

SETUP OF THE RS232 MODE

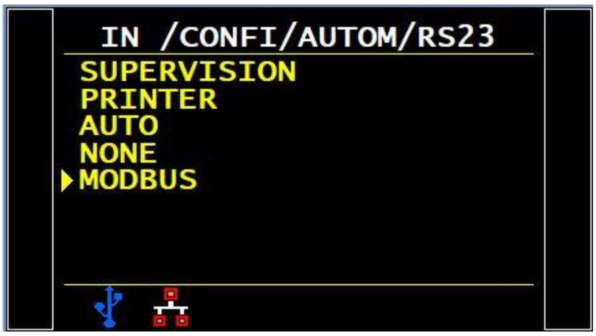
From the ATEQ device



```
 IN /CONFI/AUTOMATISM
 USB      : SUPERVISION
►RS232    : PRINTER
 Date _Time
 FIELDBUS
 OUTPUTS CONFIG.
 INPUTS CONFIG.
```

From the **MAIN MENU** screen of your ATEQ device:

- ➢ **CONFIGURATION**
- ➢ **AUTOMATISM**
- ➢ **RS232**



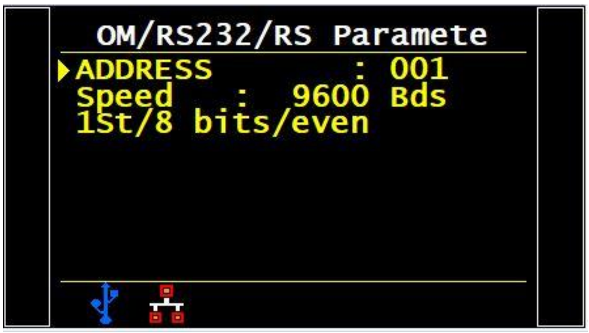
```
 IN /CONFI/AUTOM/RS23
 SUPERVISION
 PRINTER
 AUTO
 NONE
►MODBUS
```

Choose **MODBUS** value in the new window.
It will also give you access to the serial parameters.

SETUP OF THE STATION NUMBER

ℹ The **station number** must be the same on slave and master.

From the ATEQ device



```
OM/RS232/RS Paramete
▶ADDRESS        :  001
 Speed   :  9600 Bds
 1St/8 bits/even
```

From the **MAIN MENU** screen of your ATEQ device:

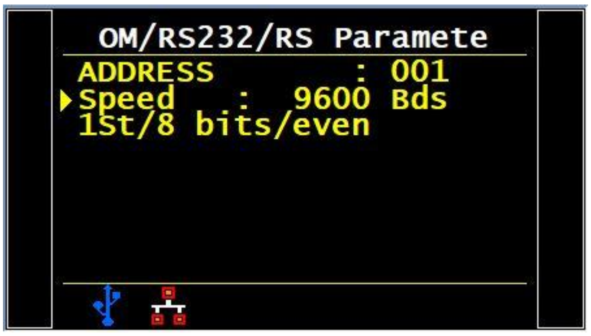
- ➢ **CONFIGURATION**
- ➢ **AUTOMATISM**
- ➢ **RS232: MODBUS**
- ➢ **ADDRESS**

The station number can be equal to a value between **1 and 255**.

SETUP OF THE COMMUNICATION SPEED

The **speed** must be the same on slave and master.

From the ATEQ device



```
OM/RS232/RS Paramete
ADDRESS         : 001
▶Speed    :  9600 Bds
1St/8 bits/even
```

From the **MAIN MENU** screen of your ATEQ device:

- ➢ **CONFIGURATION**
- ➢ **AUTOMATISM**
- ➢ **RS232: MODBUS**
- ➢ **Speed**

The speed can be equal to:

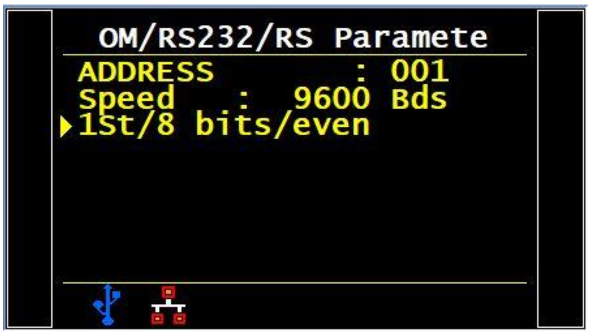
- — **4800 bauds**
- — **9600 bauds**
- — **19200 bauds**
- — **28800 bauds**
- — **38400 bauds**
- — **57600 bauds**

SETUP OF THE PARITY

The **parity** must be the same on slave and master.

The Modbus RTU configuration on an ATEQ device is **always 8 bits long with one stop bit**.

From the ATEQ device



```
OM/RS232/RS Paramete
ADDRESS      : 001
Speed    :   9600 Bds
▸1St/8 bits/even
```

From the **MAIN MENU** screen of your ATEQ device:

- ➢ **CONFIGURATION**
- ➢ **AUTOMATISM**
- ➢ **RS232: MODBUS**

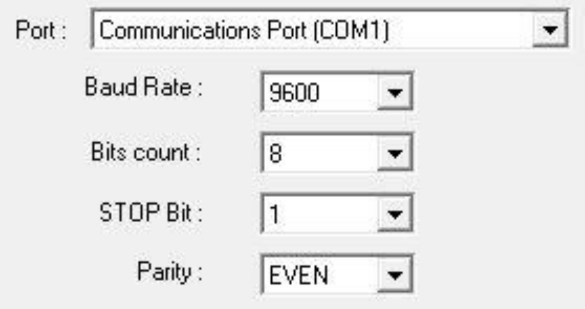
Select the last line in this menu to change the parity.

The parity can be equal to:

- **None**
- **0**
- **1**
- **Even**
- **Odd**

Configuration of the master

SETUP OF THE COMMUNICATION PORT



Port : Communications Port (COM1)

Baud Rate : 9600

Bits count : 8

STOP Bit : 1

Parity : EVEN

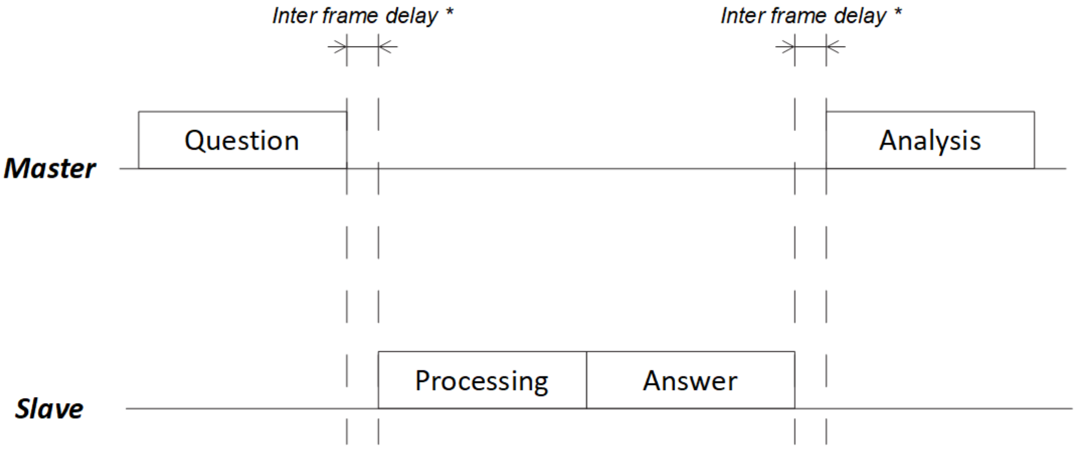
Select the connected communication port and go into its properties.

Then adjust the different settings according to those of your ATEQ device.

Frame construction

DIALOG MECHANISM (ASYNCHRONOUS LINK)

The Modbus RTU data frames do not include delimiters.
The synchronization is achieved by using a delay 3.5 times longer than the transmission time of a byte.
At the end of this delay, the first byte received is considered as the start of a new frame.



* 3,5 times the transmission time of a byte

COMMANDS

Standard access

Reminder: a **byte is 8 bits long** and a **word is 16 bits long**

Reminder: "h" indicates a hexadecimal code, "(d)" indicates a decimal code.

The **Standard access** allows the user to read/write **multiple items** in a single frame.
The ATEQ 6th series instruments support three different functions in **Standard access**.

Writing N*words: 10h

Question:

Slave address	Function number (10h)	Word address	Number of words to write	Number of bytes to write	Data 0	...	Data N	CRC
Byte	*Byte*	*Word*	*Word*	*Byte*	*N*words*			*Word*

Answer:

Slave address	Function number (10h)	Word address	Number of written words	CRC
Byte	*Byte*	*Word*	*Word*	*Word*

Reading N*words: 03h

Question:

Slave address	Function number (03h)	Word address	Number of words to read	CRC
Byte	*Byte*	*Word*	*Word*	*Word*

Answer:

Slave address	Function number (03h)	Number of read bytes	Data 0	...	Data N	CRC
Byte	*Byte*	*Byte*	*N*words*			*Word*

Writing a bit: 05h

Question:

Slave address	Function number (05h)	Bit address	Bit value Force bit to 1: FF00h Force bit to 0: 0000h	CRC
Byte	*Byte*	*Word*	*Word*	*Word*

Answer (identical to the question):

Slave address	Function number (05h)	Bit address	Bit value Force bit to 1: FF00h Force bit to 0: 0000h	CRC
Byte	*Byte*	*Word*	*Word*	*Word*

Direct access

> Reminder: a **byte is 8 bits long** and a **word is 16 bits long**

> Reminder: "h" indicates a hexadecimal code, "(d)" indicates a decimal code.

The **Direct access** allows the user to read/write **directly only one item** in a single frame.
The ATEQ 6th series instruments support two different functions in **Direct access**.

Writing N*words: 10h

Question:

Slave address	Function number (10h)	Direct access address	Number of words to write	Number of bytes to write	Data 0	…	Data N	CRC
Byte	*Byte*	*Word*	*Word*	*Byte*	*N*words*			*Word*

Answer:

Slave address	Function number (10h)	Direct access address	Number of written words	CRC
Byte	*Byte*	*Word*	*Word*	*Word*

Reading N*words: 03h

Question:

Slave address	Function number (03h)	Direct access address	Number of words to read	CRC
Byte	*Byte*	*Word*	*Word*	*Word*

Answer:

Slave address	Function number (03h)	Number of read bytes	Data 0	…	Data N	CRC
Byte	*Byte*	*Byte*	*N*words*			*Word*

Command error handling

> Reminder: a **byte is 8 bits long** and a **word is 16 bits long**

> Reminder: "h" indicates a hexadecimal code, "(d)" indicates a decimal code.

Error frame

The errors are handled in the answer of the slave to a request of the master.
When an error occurs, the slaves add **80h** to the **Function number** followed by the error code:

— Error on a **Writing N*words (10h)** request

Slave address	Function number + 80h (90h)	Error code	CRC
Byte	*Byte*	*Byte*	*Word*

— Error on a **Reading N*words (03h)** request

Slave address	Function number + 80h (83h)	Error code	CRC
Byte	*Byte*	*Byte*	*Word*

Error codes

Hexa code	Item	Meaning
02	ILLEGAL DATA ADDRESS	Address out of range
03	ILLEGAL DATA VALUE	Value out of limit / value not valid / parameter or bit unavailable

CRC CALCULATION

Definition

In Modbus RTU, the **Cyclic Redundancy Check** is calculated on 16 bits. It is therefore called **CRC16**.

The CRC16 is a calculation based on the binary value of each character composing the frame. This function translates the frame into a 16-bit binary word; this binary word is inserted at the end of the frame.

When the master or the slave receives a frame, it calculates the CRC16 of this frame and compares the result with the value of CRC16 contained in the frame (last word), in order to check that the exchange has been correctly undertaken:

- If the CRC16 corresponds, the slave responds.
- If the CRC16 is false:
  - The slave that receives the erroneous frame does not respond,
  - The master having not received a response restarts the same request for the slave.

> If the exchange is not accomplished after 2 attempts, the master declares a communication error in the network and stops the exchanges.

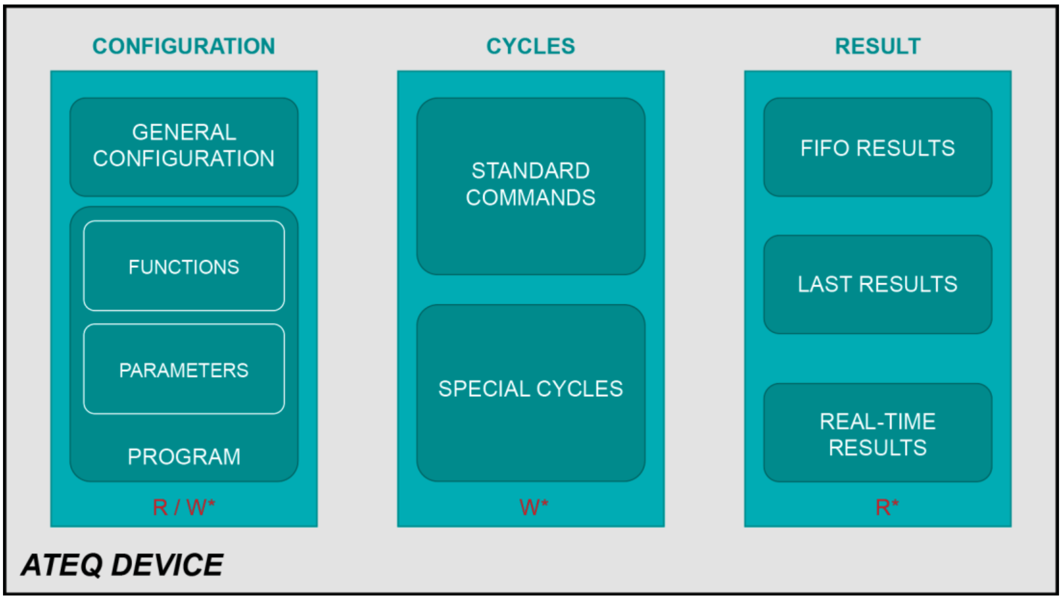
CRC16 calculation algorithm

```
CRC16 = 0FFFFh                    // Initialization at the start of each new data frame

As long as (NO(End of frame))
        CRC16=(CRC16 OR exclusive character received)
        for (i=0;i<8;i++)
        {
                CRC16=CRC16/2
                If there are remainders to the division then
                        CRC16= (CRC16 XOR 0A001h)
        }
FTQ
```

Functional description of an ATEQ device

INTRODUCTION



- R/W*: reading and writing
- W*: writing only
- R*: reading only

Commands – Standard access

Reminder: "h" indicates a hexadecimal code, "(d)" indicates a decimal code.

The **Standard access** allows the user to read/write **multiple items** in a single frame.
The ATEQ 6th series instruments support four different functions in **Standard access**:

- **Writing N*words: 10h**

Slave address	Function number (10h)	Word address	Number of words to write	Number of bytes to write	Word 0	…	Word N	CRC
Byte	*Byte*	*Word*	*Word*	*Byte*		*N*words*		*Word*

- **Reading N*words: 03h**

Slave address	Function number (03h)	Word address	Number of words to read	CRC
Byte	*Byte*	*Word*	*Word*	*Word*

- **Writing a bit: 05h**

Slave address	Function number (05h)	Bit address	Bit value Force bit to 1: FF00h Force bit to 0: 0000h	CRC
Byte	*Byte*	*Word*	*Word*	*Word*

- **Writing a word: 06h**

Slave address	Function number (06h)	Word address	Word value	CRC
Byte	*Byte*	*Word*	*Word*	*Word*

Word addresses

These addresses are used with the **Writing N*words (10h)** or the **Reading N*words (03h)** functions:

Hexa address	Item	Read	Write
0000	Read parameters	Y	N
0010	FIFO result	Y	N
0011	Last result	Y	N
0020	Step code in progress	Y	N
0030	Real time result (real time information)	Y	N
007F	Write parameters	N	Y
0100	Extended menu bits	Y	Y
0110	Function bits	Y	Y
0120	Personalization	Y	Y
0130	Number of results in FIFO	Y	N
0200	Program to be selected	N	Y
0201	Special cycle	N	Y
0202	Selected program	Y	N
3004	Program in edition mode	Y	Y

Bit addresses

These addresses are used with the **Writing a bit (05h)** function:

Hexa address	Item
0000	Reset
0001	Start
0002	FIFO reset

Commands – Direct access

Reminder: "h" indicates a hexadecimal code, "(d)" indicates a decimal code.

The **Direct access** allows the user to read/write **directly only one item** in a single frame.
The ATEQ 6th series instruments support two different functions in **Direct access**:

— **Writing N*words: 10h** (limited to 2 maximum words)

Slave address	Function number (10h)	Direct access address	Number of words to write	Number of bytes to write	Word 0	…	Word N	CRC
Byte	*Byte*	*Word*	*Word*	*Byte*	*N*words*			*Word*

— **Reading N*words: 03h** (limited to 2 maximum words)

Slave address	Function number (03h)	Direct access address	Number of words to read	CRC
Byte	*Byte*	*Word*	*Word*	*Word*

Direct access addresses

Read hexa address	Write hexa address	Item
2000	6000	Program in edition mode
2001 … 2200	6001 … 6200	Parameters
2201 … 220D	-	Status and real time measurement
2301 … 2328	-	Last result
2401 … 24FF	6401 … 64FF	Extended menu bits
2601 … 26FF	6601 … 66FF	Function bits

Error Handling

Reminder: "h" indicates a hexadecimal code, "(d)" indicates a decimal code.

The errors are handled in the answer of the slave to a request of the master.
When an error occurs, the slaves add **80h** to the **Function number** followed by the error code:

— Error on a **Writing N*words (10h)** request

Slave address	Function number + 80h (90h)	Error code	CRC
Byte	*Byte*	*Byte*	*Word*

— Error on a **Reading N*words (03h)** request

Slave address	Function number + 80h (83h)	Error code	CRC
Byte	*Byte*	*Byte*	*Word*

Error codes

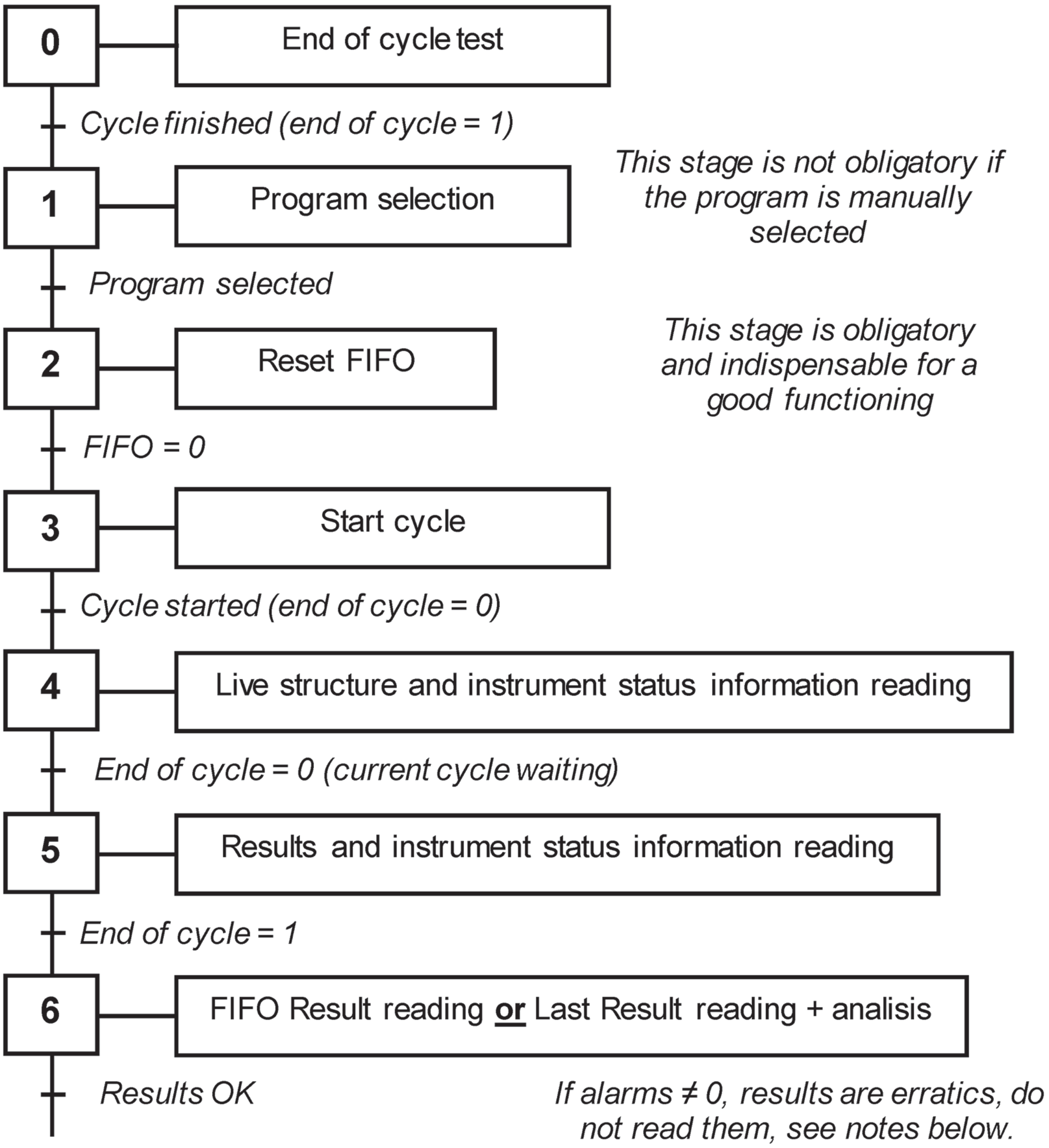
Hexa code	Item	Meaning
02	ILLEGAL DATA ADDRESS	Address out of range
03	ILLEGAL DATA VALUE	Value out of limit / value not valid / parameter or bit unavailable

Treatment of the commands

Reminder: "h" indicates a hexadecimal code, "(d)" indicates a decimal code.

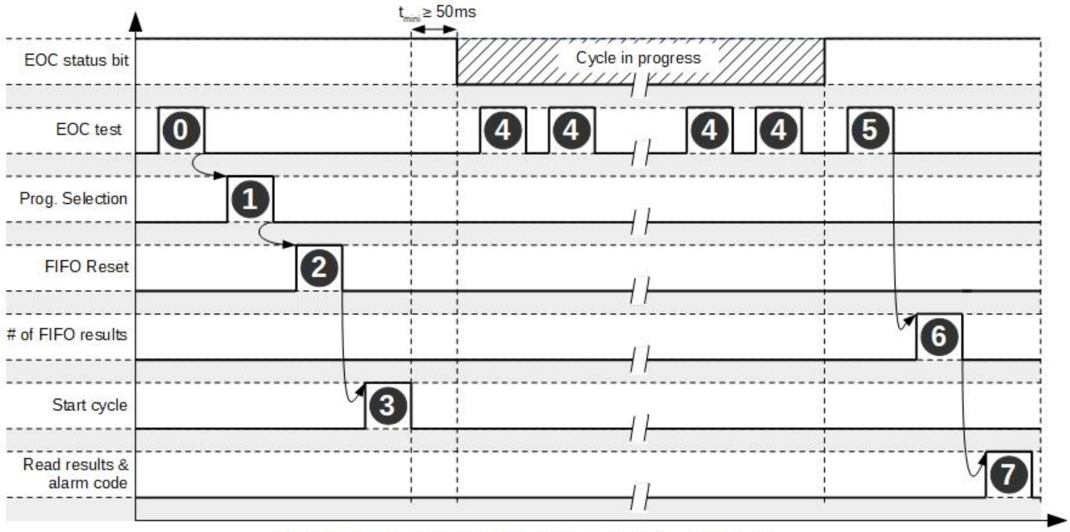
ATEQ device using

Base procedure for using an ATEQ instrument.


0
End of cycle test
Cycle finished (end of cycle = 1)
1
Program selection
This stage is not obligatory if the program is manually selected
Program selected
2
Reset FIFO
This stage is obligatory and indispensable for a good functioning
FIFO = 0
3
Start cycle
Cycle started (end of cycle = 0)
4
Live structure and instrument status information reading
End of cycle = 0 (current cycle waiting)
5
Results and instrument status information reading
End of cycle = 1
6
FIFO Result reading or Last Result reading + analisis
Results OK
If alarms ≠ 0, results are erratics, do not read them, see notes below.

If the number of results in the FIFO = 0, the results are erratic, **do not read them.**
If there's an alarm bit, read the alarm code and **do not use the measurements results (erratic results).**

Modbus progress chart



WARNING : The status bits update rate is about 50ms

0:	Read 13 words @30h : word 4, bit 5 = 1 (EOC status bit)	6:	*Read the number of results in FIFO :* Read 13 words @30h : **if word 2 ≥ 1** go to step 7, **else** END	*Use of FIFO Results*
1:	Write 1 word @200h : word = n° prog (0001h = prog 2)	7:	Read 12 words @10h : 12 words (size of standard results) **if Alarm Code = 0** go to step 8, **else** END	
2:	**ALWAYS RESET THE FIFO** Write bit @02h : bit = FF (command « Reset FIFO »)	8:	Use the results recovered at step 7 (if Alarm code was equal to 0)	
3:	Write bit @01h : bit = FF (command « Start ») ***t_{mini} ≥ 50ms***	6:	*Read the number of results in FIFO :* Read 13 words @30h : **if word 2 ≥ 1** go to step 7, **else** END	*Use of Last Results*
4:	Read 13 words @30h : word 4, bit 5 = 0 (EOC status bit)	7:	Read 12 words @11h : 12 words (size of standard results) **if Alarm Code = 0** go to step 8, **else** END	
5:	Read 13 words @30h : word 4, bit 5 = 1 (EOC status bit)	8:	Use the results recovered at step 7 (if Alarm code was equal to 0)	

CONFIGURATION

General configuration

Table of the configuration / extended menus bits

> Reminder: **Direct access addresses** are expressed in **hexadecimal**

The bits below are mostly present in the **CONFIGURATION** or **More functions...** menus.
They are only used to allow the access to other parameters according to the configuration, depending on the configuration, these are active or not.

> Acronyms used in the “Menu” column:
> — Conf: CONFIGURATION
> — +Func: FUNCTIONS > More functions…
> — RS232: CONFIGURATION > RS232

Word	Bit n°	Mask		D.A. address		Meaning	Menu
		Hexa	Dec	Read	Write		
1	0	0001	1	241A	641A	Permanent blowing activation.	Conf
	1	0002	2			Reserved.	
	2	0004	4	2404	6404	Fill type.	+Funct
	3	0008	8	2403	6403	Pre-fill type.	+Funct
	4	0010	16	2401	6401	Recovery thresholds.	+Funct
	5	0020	32	241C	641C	Cycle end.	+Funct
	6	0040	64			Reserved.	
	7	0080	128	2408	6408	Peak meter.	+Funct
	8	0100	256			Reserved.	
	9	0200	512	2405	6405	Reference volume.	+Funct
	10	0400	1024	240B	640B	ATR 0.	+Funct
	11	0800	2048	240C	640C	ATR 1.	+Funct
	12	1000	4096	240D	640D	ATR 2.	+Funct
	13	2000	8192	2413	6413	Program name.	+Funct
	14	4000	16384	241F	641F	Chaining.	+Funct
	15	8000	32768	2420	6420	Automatic connector.	+Funct

Word	Bit n°	Mask		D.A. address		Meaning	Menu
		Hexa	Dec	Read	Write		
2	16	0001	1	243B	643B	Calibration check.	+Funct
	17	0002	2	2416	6416	Valve codes (output codes).	+Funct
	18	0004	4			Reserved.	
	19	0008	8	2422	6422	Stamping.	+Funct
	20	0010	16			Reserved.	
	21	0020	32	2424	6424	N test.	+Funct
	22	0040	64			Reserved.	
	23	0080	128	2426	6426	Sending cond.: pass part.	RS232
	24	0100	256	2427	6427	Sending cond.: fail test part.	RS232
	25	0200	512	2428	6428	Sending cond.: fail ref. part.	RS232
	26	0400	1024	2429	6429	Sending cond.: alarm presence.	RS232
	27	0800	2048	242A	642A	Sending cond.: pressure error.	RS232
	28	1000	4096	242B	642B	Sending cond.: end of cycle.	RS232
	29	2000	8192	242C	642C	Sending cond.: recoverable.	RS232
	30	4000	16384	243C	643C	Sending cond.: calibration.	RS232
	31	8000	32768	242D	642D	Frame content: time stamp.	RS232
3	32	0001	1	2412	6412	Frame content: name.	RS232
	33	0002	2	242E	642E	Content of the frame: pressure.	RS232
	34	0004	4	242F	642F	Security.	Conf
	35	0008	8	2414	6414	External dump.	Conf
	36	0010	16	2430	6430	Exportation.	RS232
	37	0020	32	240F	640F	Automatic reset.	Conf
	38	0040	64			Reserved.	
	39	0080	128			Reserved.	
	40	0100	256			Reserved.	
	41	0200	512			Reserved.	
	42	0400	1024			Reserved.	
	43	0800	2048	243E	643E	Parameters automatic setting.	Conf
	44	1000	4096			Reserved.	
	45	2000	8192	2439	6439	Page feed.	RS232
	46	4000	16384	2434	6434	Sign change.	+Funct
	47	8000	32768	2440	6440	After sale service cycle.	+Funct

Word	Bit n°	Mask		D.A. address		Meaning	Menu
		Hexa	Dec	Read	Write		
4	48	0001	1	2402	6402	Unit type.	+Funct
	49	0002	2	2441	6441	Automatic reset piezo 2.	Conf
	50	0004	4			Reserved.	
	51	0008	8	2438	6438	Electronic regulator mode.	Conf
	52	0010	16	2435	6435	Auxiliary codes activation.	+Funct
	53	0020	32	2409	6409	Filtering.	+Funct
	54	0040	64			Reserved.	
	55	0080	128	2411	6411	Quick automatic reset activation.	Conf
	56	0100	256	2442	6442	Permanent electronic regulator.	Conf
	57	0200	512	2443	6443	Bar code.	Conf
	58	0400	1028	2444	6444	Flow reject.	+Funct
	59	0800	2048	2436	6436	No negative.	+Funct
	60	1000	4096	2415	6415	Dump threshold.	+Funct
	61	2000	8192	240E	640E	ATR 3.	+Funct
	62	4000	16384	2445	6445	In 7 test configuration.	Conf
	63	8000	32768			Reserved.	
5	64	0001	1	2486	6486	Absolute value.	Conf
	65	0002	2	249F	649F	Leak display mode.	+Funct
	66	0004	4	2487	6487	By pass valve.	Conf
	67	0008	8			Reserved.	
	68	0010	16			Reserved.	
	69	0020	32			Reserved.	
	70	0040	64			Reserved.	
	71	0080	128	248C	648C	Dump Off.	+Funct
	72	0100	256	249D	649D	Program selection on bar code reading.	+Funct
	73	0200	512	2492	6492	Bar code reset on end of cycle.	+Funct
	74	0400	1024	248D	648D	Cut Off.	+Funct
	75	0800	2048	248E	648E	ATF.	+Funct
	76	1000	4096			Reserved.	
	77	2000	8192			Reserved.	
	78	4000	16384			Reserved.	
	79	8000	32768			Reserved.	
6	80>95					Word Reserved.	

Word	Bit n°	Mask		D.A. address		Meaning	Menu
		Hexa	Dec	Read	Write		
7	96	0001	1	249B	649B	Buzzer function.	+Funct
	97	0002	2	249E	649E	Long test (x100) function.	+Funct
	98	0004	4	249C	649C	Permanent blowing.	Conf
	99	0008	8			Reserved.	
	100	0010	16			Reserved.	
	101	0020	32	24B9	64B9	Display optional.	+Funct
	102	0040	64	24B6	64B6	Pressure Drop.	+Funct
	103	0080	128	24B7	64B7	Pressure correction (≥v1.400).	+Funct
	104	0100	256	24B8	64B8	Standard conditions (≥v1.400).	+Funct
	105	0200	512	248F	648F	Ref No Dump.	+Funct
	106	0400	1024			Reserved.	
	107	0800	2048	24BB	64BB	Offset.	+Funct
	108	1000	4096			Reserved.	
	109	2000	8192			Reserved.	
	110	4000	16384	24BE	64BE	Auto Selection Prog.	Conf
	111	8000	32768	24BF	64BF	Save Volume Selection.	Conf

Example: bit number 13 (Program name) activated to 1, will place to "2000h" the value in the first word.
2000h is equivalent to 8192 in decimal and 0010000000000000 in binary.
In the Modbus frame, the words will follow each other: word 1 + word 2 + ….. + word n.

Reading of the configuration / extended menu bits

The configuration / extended menu bits are independents of the program number.

- *Standard access*

Example of reading 7 words of the “Configuration / Extended menu bits”:

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>— Make a <b>Read N*words</b> request of 7 words at the <b>0100h</b> address.
On network:
<table><tr><td>01</td><td>03</td><td>01</td><td>00</td><td>00</td><td>07</td><td>05</td><td>F4</td></tr></table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>01 00</td><td>Word address (Extended menu bits)</td></tr>
<tr><td>00 07</td><td>Number of words to read</td></tr>
<tr><td>05 F4</td><td>CRC</td></tr>
</table></td><td></td></tr>
<tr><td></td><td>— Answer to the request:
On network:
<table>
<tr><td>01</td><td>03</td><td>0E</td><td>00</td><td>0C</td><td>20</td><td>10</td><td>00</td></tr>
<tr><td>80</td><td>21</td><td>00</td><td>00</td><td>00</td><td>00</td><td>00</td><td>20</td></tr>
<tr><td>00</td><td>90</td><td>74</td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>0E</td><td>Number of read bytes</td></tr>
<tr><td>00 0C</td><td>Word 1: read 0C00h</td></tr>
<tr><td>20 10</td><td>Word 2: read 1020h</td></tr>
<tr><td>00 80</td><td>Word 3: read 8000h</td></tr>
<tr><td>21 00</td><td>Word 4: read 0021h</td></tr>
<tr><td>00 00</td><td>Word 5: read 0000h</td></tr>
<tr><td>00 00</td><td>Word 6: read 0000h</td></tr>
<tr><td>20 00</td><td>Word 7: read 0020h</td></tr>
<tr><td>90 74</td><td>CRC</td></tr>
</table></td></tr>
</table>

- *Direct access*

In Direct access, the master can only access to bits one by one.

Example for reading the state of the “Chaining” bit (word 1, bit 14):

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>— Make a <b>Read N*words</b> request of 1 word at the <b>241Fh</b> address.
On network:
<table><tr><td>01</td><td>03</td><td>24</td><td>1F</td><td>00</td><td>01</td><td>BF</td><td>3C</td></tr></table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>24 1F</td><td>D.A. address of the “Chaining” bit</td></tr>
<tr><td>00 01</td><td>Number of words to read</td></tr>
<tr><td>BF 3C</td><td>CRC</td></tr>
</table></td><td></td></tr>
<tr><td></td><td>— Answer to the request:
On network:
<table><tr><td>01</td><td>03</td><td>02</td><td>01</td><td>00</td><td>B9</td><td>D4</td></tr></table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>02</td><td>Number of read bytes</td></tr>
<tr><td>01 00</td><td>Word: read 0001h (Chaining bit = 1)</td></tr>
<tr><td>B9 D4</td><td>CRC</td></tr>
</table></td></tr>
</table>

Writing of the configuration / extended menu bits

The configuration / extended menu bits are independents of the program number.

- *Standard access*

Example of writing 7 words in the “Configuration / Extended menu bits”:

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>
— Make a <b>Write N*words</b> request of 7 words at the <b>0100h</b> address.
On network:
<table>
<tr><td>01</td><td>10</td><td>01</td><td>00</td><td>00</td><td>07</td><td>0E</td><td>00</td></tr>
<tr><td>4C</td><td>20</td><td>10</td><td>00</td><td>80</td><td>21</td><td>00</td><td>00</td></tr>
<tr><td>00</td><td>00</td><td>00</td><td>20</td><td>00</td><td>B9</td><td>32</td><td></td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>01 00</td><td>Word address (Extended menu bits)</td></tr>
<tr><td>00 07</td><td>Number of words to write</td></tr>
<tr><td>0E</td><td>Number of bytes to write</td></tr>
<tr><td>00 4C</td><td>Word 1: write 4C00h</td></tr>
<tr><td>20 10</td><td>Word 2: write 1020h</td></tr>
<tr><td>00 80</td><td>Word 3: write 8000h</td></tr>
<tr><td>21 00</td><td>Word 4: write 0021h</td></tr>
<tr><td>00 00</td><td>Word 5: write 0000h</td></tr>
<tr><td>00 00</td><td>Word 6: write 0000h</td></tr>
<tr><td>20 00</td><td>Word 7: write 0020h</td></tr>
<tr><td>B9 32</td><td>CRC</td></tr>
</table>
</td><td></td></tr>
<tr><td></td><td>
— Answer to the request:
On network:
<table>
<tr><td>01</td><td>10</td><td>01</td><td>00</td><td>00</td><td>07</td><td>80</td><td>37</td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>01 00</td><td>Word address (Extended menu bits)</td></tr>
<tr><td>00 07</td><td>Number of written words</td></tr>
<tr><td>80 37</td><td>CRC</td></tr>
</table>
</td></tr>
</table>

- *Direct access*

In Direct access, the master can only access to bits one by one.

Example for writing the “Chaining” bit to 1 (word 1, bit 14):

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>— Make a <b>Write N*words</b> request of 1 word at the <b>641Fh</b> address.
On network:
<table>
<tr><td>01</td><td>10</td><td>64</td><td>1F</td><td>00</td><td>01</td><td>02</td><td>01</td></tr>
<tr><td>00</td><td>80</td><td>69</td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>64 1F</td><td>D.A. address of the “Chaining” bit</td></tr>
<tr><td>00 01</td><td>Number of words to write</td></tr>
<tr><td>02</td><td>Number of bytes to write</td></tr>
<tr><td>01 00</td><td>Word: write 0001h (Chaining bit = 1)</td></tr>
<tr><td>80 69</td><td>CRC</td></tr>
</table>
</td><td></td></tr>
<tr><td></td><td>— Answer to the request:
On network:
<table>
<tr><td>01</td><td>10</td><td>64</td><td>1F</td><td>00</td><td>01</td><td>2F</td></tr>
<tr><td>3F</td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>64 1F</td><td>D.A. address of the “Chaining” bit</td></tr>
<tr><td>00 01</td><td>Number of written words</td></tr>
<tr><td>2F 3F</td><td>CRC</td></tr>
</table>
</td></tr>
</table>

Program in edition mode command on the ATEQ device

> Always subtract 1 from the value of the program number to be put in edition mode.
> Example: for putting program number 2 in edition mode, send the value 1 at the address 3004h.

- *Standard access*

Example for putting program number 3 in edition mode:

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>

— Make a **Write N*words** request of 1 word at the address **3004h**.

On network:

<table>
<tr><td>01</td><td>10</td><td>30</td><td>04</td><td>00</td><td>01</td><td>02</td><td>02</td></tr>
<tr><td>00</td><td>96</td><td>B7</td><td></td><td></td><td></td><td></td><td></td></tr>
</table>

<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>30 04</td><td>Word address (Program in edition mode)</td></tr>
<tr><td>00 01</td><td>Number of words to write</td></tr>
<tr><td>02</td><td>Number of bytes to write</td></tr>
<tr><td>02 00</td><td>Word: write 0002h (Program n°3)</td></tr>
<tr><td>96 B7</td><td>CRC</td></tr>
</table>

</td><td></td></tr>
<tr><td></td><td>

— Answer to the request:

On network:

<table>
<tr><td>01</td><td>10</td><td>30</td><td>04</td><td>00</td><td>01</td><td>4F</td></tr>
<tr><td>08</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>

<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>30 04</td><td>Word address (Program in edition mode)</td></tr>
<tr><td>00 01</td><td>Number of words to write</td></tr>
<tr><td>4F 08</td><td>CRC</td></tr>
</table>

</td></tr>
</table>

- *Direct access*

Example for putting program number 3 in edition mode:

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>

— Make a **Write N*words** request of 1 word at the address **6000h**.

On network:

<table>
<tr><td>01</td><td>10</td><td>60</td><td>00</td><td>00</td><td>01</td><td>02</td><td>02</td></tr>
<tr><td>00</td><td>C7</td><td>36</td><td></td><td></td><td></td><td></td><td></td></tr>
</table>

<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>60 00</td><td>D.A. address for program in edition mode</td></tr>
<tr><td>00 01</td><td>Number of words to write</td></tr>
<tr><td>02</td><td>Number of bytes to write</td></tr>
<tr><td>02 00</td><td>Word: write 0002h (Program n°3)</td></tr>
<tr><td>C7 36</td><td>CRC</td></tr>
</table>

</td><td></td></tr>
<tr><td></td><td>

— Answer to the request:

On network:

<table>
<tr><td>01</td><td>10</td><td>60</td><td>00</td><td>00</td><td>01</td><td>1F</td></tr>
<tr><td>C9</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>

<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>60 00</td><td>D.A. address for program in edition mode</td></tr>
<tr><td>00 01</td><td>Number of words to write</td></tr>
<tr><td>1F C9</td><td>CRC</td></tr>
</table>

</td></tr>
</table>

Function

Table of the function bits

Table of the function bits per program.

Reminder: **Direct access addresses** are expressed in **hexadecimal**

The bits below are present in the **FUNCTIONS** menu of each program, if these have been previously validated in the **More functions...** menu.

Word	Bit n°	Mask		D.A. address		Meaning	Menu
		Hexa	Dec	Read	Write		
1	0	0001	1	2610	6610	Fill regulator Number.	Funct
	1	0002	2	260F	660F	Pre-fill regulator Number.	Funct
	2	0004	4	2604	6604	Fill type function.	Funct
	3	0008	8	2603	6603	Pre-fill type function.	Funct
	4	0010	16	2601	6601	Recovery level function.	Funct
	5	0020	32	261E	661E	End of cycle function.	Funct
	6	0040	64	261F	661F	Automatic reset end cycle function.	Funct
	7	0080	128	2620	6620	Reset and dump end of cycle function.	Funct
	8	0100	256	2621	6621	Fill mode end of cycle function.	Funct
	9	0200	512	2608	6608	Peak hold function.	Funct
	10	0400	1024	2605	6605	Reference volume function.	Funct
	11	0800	2048	260B	660B	ATR0 function.	Funct
	12	1000	4096	260C	660C	ATR1 function.	Funct
	13	2000	8192	260D	660D	ATR2 function.	Funct
	14	4000	16384	2622	6622	Sequencing function.	Funct
	15	8000	32768	2623	6623	Pass part sequencing function.	Funct
2	16	0001	1	2624	6624	Fail test part sequencing function.	Funct
	17	0002	2	2625	6625	Fail reference part sequencing function.	Funct
	18	0004	4	2626	6626	Alarm sequencing function.	Funct
	19	0008	8	2627	6627	Pressure fault sequencing function.	Funct
	20	0010	16	2628	6628	End of cycle sequencing function.	Funct
	21	0020	32			Reserved.	
	22	0040	64	262A	662A	Recovery part sequencing function.	Funct
	23	0080	128	2640	6640	Calibration check sequencing function.	Funct
	24	0100	256	262B	662B	Automatic connector function.	Funct
	25	0200	512	2641	6641	Calibration check function.	Funct
	26	0400	1024	2612	6612	Valve code function.	Funct
	27	0800	2048	2642	6642	External valve code 1 function.	Funct
	28	1000	4096	2643	6643	External valve code 2 function.	Funct
	29	2000	8192	2644	6644	External valve code 3 function.	Funct
	30	4000	16384	2645	6645	External valve code 4 function.	Funct
	31	8000	32768	2646	6646	External valve code 5 function.	Funct

Word	Bit n°	Mask		D.A. address		Meaning	Menu
		Hexa	Dec	Read	Write		
3	32	0001	1	2647	6647	External valve code 6 function.	Funct
	33	0002	2	2648	6648	Internal valve code 1 function.	Funct
	34	0004	4	2649	6649	Internal valve code 2 function.	Funct
	35	0008	8	262C	662C	Stamp function.	Funct
	36	0010	16	262D	662D	Pass part stamp function.	Funct
	37	0020	32	262E	662E	Fail test part stamp function.	Funct
	38	0040	64	262F	662F	Fail reference part stamp function.	Funct
	39	0080	128	2630	6630	Alarm stamp function.	Funct
	40	0100	256	2631	6631	Pressure fault stamp function.	Funct
	41	0200	512	2632	6632	End of cycle stamp function.	Funct
	42	0400	1024	2633	6633	Recovery part stamp function.	Funct
	43	0800	2048	264A	664A	Calibration check stamp function.	Funct
	44	1000	4096	2634	6634	N test function.	Funct
	45	2000	8192			Reserved.	
	46	4000	16384			Reserved.	
	47	8000	32768			Reserved.	
4	48	0001	1	261B	661B	External dump function.	Funct
	49	0002	2			Reserved.	
	50	0004	4			Reserved.	
	51	0008	8			Reserved.	
	52	0010	16	2611	6611	Sign change function.	Funct
	53	0020	32	263E	663E	Obligatory reset on end of cycle.	Funct
	54	0040	64	2638	6638	Auxiliaries codes function.	Funct
	55	0080	128	2639	6639	Auxiliaries codes 1 function.	Funct
	56	0100	256	263A	663A	Auxiliaries codes 2 function.	Funct
	57	0200	512	263B	663B	Auxiliaries codes 3 function.	Funct
	58	0400	1024	263C	663C	Auxiliaries codes 4 function.	Funct
	59	0800	2048	264C	664C	Auto param function (not available).	Funct
	60	1000	4096	2609	6609	Filtering function.	Funct
	61	2000	8192	264D	664D	Bar code function.	Funct
	62	4000	16384	264E	664E	Flow reject function.	Funct
	63	8000	32768	263F	663F	No negative function.	Funct

Word	Bit n°	Mask		D.A. address		Meaning	Menu
		Hexa	Dec	Read	Write		
5	64	0001	1	264F	664F	Start after reading bar code function.	Funct
	65	0002	2	260E	660E	ATR3 function.	Funct
	66	0004	4	266B	666B	Absolute value function.	Funct
	67	0008	8	266C	666C	Bypass valve function.	Funct
	68	0010	16			Reserved.	
	69	0020	32			Reserved.	
	70	0040	64			Reserved.	
	71	0080	128	2671	6671	Dump off function.	Funct
	72	0100	256	2672	6672	Cut off function.	Funct
	73	0200	512	2673	6673	ATF function.	Funct
	74	0400	1024			Reserved.	
	75>79					Reserved.	
6	80>95					Word Reserved.	
7	96	0001	1	267D	667D	Optional auxiliaries codes function.	Funct
	97	0002	2	267E	667E	Optional auxiliaries codes 1 function.	Funct
	98	0004	4	267F	667F	Optional auxiliaries codes 2 function.	Funct
	99	0008	8	2680	6680	Optional auxiliaries codes 3function.	Funct
	100	0010	16	2681	6681	Optional auxiliaries codes 4 function.	Funct
	101	0020	32	2682	6682	Optional valves codes function.	Funct
	102	0040	64	2683	6683	Optional external valves codes 1.	Funct
	103	0080	128	2684	6684	Optional external valves codes 2.	Funct
	104	0100	256	2685	6685	Optional external valves codes 3.	Funct
	105	0200	512	2686	6686	Optional external valves codes 4.	Funct
	106	0400	1024	2687	6687	Optional external valves codes 5.	Funct
	107	0800	2048	2688	6688	Optional external valves codes 6.	Funct
	108	1000	4096	2689	6689	Optional internal valves codes 1.	Funct
	109	2000	8192	268A	668A	Optional internal valves codes 2.	Funct
	110	4000	16384	268B	668B	Buzzer function.	Funct
	111	8000	32768	268C	668C	Pass part buzzer function.	Funct
8	112	0001	1	268D	668D	Fail part buzzer function.	Funct
	113	0002	2	268E	668E	Alarm buzzer function.	Funct
	114	0004	4	268F	668F	End of cycle buzzer function.	Funct
	115	0008	8	2694	6694	Long Test Time function.	Funct
	116	0010	16	2691	6691	Permanent dump function.	Funct
	117	0020	32	2692	6692	Input 7 test function.	Funct
	118	0040	64			Reserved.	
	119	0080	128			Reserved.	
	120	0100	256			Reserved.	
	121	0200	512	26AE	66AE	Pressure Drop.	Funct
	122	0400	1024	26AF	66AF	Auto Verif Etal.	Funct
	123	0800	2048	26B0	66B0	Pressure correction (≥v1.400).	Funct
	124	1000	4096	2675	6675	Standard conditions (≥v1.400).	Funct
	125	2000	8192	26B1	66B1	Ref No Dump.	Funct
	126	4000	16384	26B2	66B2	Offset.	Funct
	127	8000	32768	26B3	66B3	Permanent Fill.	Funct

Word	Bit n°	Mask		D.A. address		Meaning	Menu
		Hexa	Dec	Read	Write		
9	128	0001	1			Reserved.	
	129> 143					Reserved.	

Example: bit number 46 (Sealed components function) activated on 1, will put to "4000h" the value in the third word.
4000h is equivalent to 16384 in decimal and 0100000000000000 in binary.
In the Modbus frame, the words will follow as such: word 1 + word 2 + ….. + word n.

Reading of the function bits

> The functions bits are dependents of the program number.
> Put the wanted program in edition before executing command.

- *Standard access*

Example for reading 9 words of the “Function bits”:

Master	Slave
— Put in edition the program number on which the functions bits have to be read — Make a **Read N*words** request of 9 words at the **0110h** address. On network:	

01	03	01	10	00	09	85	F5

01	Slave address
03	Function number (Read N*words)
01 10	Word address (Function bits)
00 09	Number of words to read
85 F5	CRC

Slave:

— Answer to the request:

On network:

01	03	12	00	0C	00	00	10
20	00	10	00	00	00	00	00
00	00	00	00	00	98	CE	

01	Slave address
03	Function number (Read N*words)
12	Number of read bytes
00 0C	Word 1: read 0C00h
00 00	Word 2: read 0000h
10 20	Word 3: read 2010h
00 10	Word 4: read 1000h
00 00	Word 5: read 0000h
00 00	Word 6: read 0000h
00 00	Word 7: read 0000h
98 CE	CRC

- *Direct access*

In Direct access, the master can only access to bits one by one.

Example for reading the state of the “Sequencing activation” bit (word 1, bit 14):

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>
— Put in edition the program number on which the functions bits have to be read
— Make a <b>Read N*words</b> request of 1 word at the <b>2622h</b> address.
On network:
<table><tr><td>01</td><td>03</td><td>26</td><td>22</td><td>00</td><td>01</td><td>2F</td><td>48</td></tr></table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>26 22</td><td>D.A. address of the “Sequencing act.” bit</td></tr>
<tr><td>00 01</td><td>Number of words to read</td></tr>
<tr><td>2F 48</td><td>CRC</td></tr>
</table>
</td><td></td></tr>
<tr><td></td><td>
— Answer to the request:
On network:
<table><tr><td>01</td><td>03</td><td>02</td><td>01</td><td>00</td><td>B9</td><td>D4</td></tr></table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>02</td><td>Number of read bytes</td></tr>
<tr><td>01 00</td><td>Word: read 0001h (Sequencing act. bit = 1)</td></tr>
<tr><td>B9 D4</td><td>CRC</td></tr>
</table>
</td></tr>
</table>

Writing of the function bits

> The functions bits are dependents of the program number.
> Put the wanted program in edition before executing command.

- *Standard access*

Example of writing 9 words in the “Function bits”:

Master

— Put in edition the program number on which the functions bits have to be read

— Make a **Write N*words** request of 9 words at the **0110h** address.

On network:

01	10	01	10	00	09	12	00
08	00	00	10	20	00	10	00
00	00	00	00	00	00	00	00
00	09	95					

01	Slave address
10	Function number (Write N*words)
01 10	Word address (Function bits)
00 09	Number of words to write
12	Number of bytes to write
00 08	Word 1: write 0800h
00 00	Word 2: write 0000h
10 20	Word 3: write 2010h
00 10	Word 4: write 1000h
00 00	Word 5: write 0000h
00 00	Word 6: write 0000h
00 00	Word 7: write 0000h
00 00	Word 8: write 0000h
00 00	Word 9: write 0000h
09 95	CRC

Slave

— Answer to the request:

On network:

01	10	01	10	00	09	00	36

01	Slave address
10	Function number (Write N*words)
01 10	Word address (Function bits)
00 09	Number of written words
00 36	CRC

- *Direct access*

In Direct access, the master can only access to bits one by one.

Example for writing the state of the “Sequencing activation” bit (word 1, bit 14):

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>
— Put in edition the program number on which the functions bits have to be read
— Make a <b>Write N*words</b> request of 1 word at the <b>6622h</b> address.
On network:
<table>
<tr><td>01</td><td>10</td><td>66</td><td>22</td><td>00</td><td>01</td><td>02</td><td>01</td></tr>
<tr><td>00</td><td>A7</td><td>44</td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>66 22</td><td>D.A. address of the “Sequencing act.” bit</td></tr>
<tr><td>00 01</td><td>Number of words to write</td></tr>
<tr><td>02</td><td>Number of bytes to write</td></tr>
<tr><td>01 00</td><td>Word: write 0001h (Sequencing act. = 1)</td></tr>
<tr><td>A7 44</td><td>CRC</td></tr>
</table>
</td><td></td></tr>
<tr><td></td><td>
— Answer to the request:
On network:
<table>
<tr><td>01</td><td>10</td><td>66</td><td>22</td><td>00</td><td>01</td><td>BF</td></tr>
<tr><td>4B</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>66 22</td><td>D.A. address of the “Sequencing act.” bit</td></tr>
<tr><td>00 01</td><td>Number of written words</td></tr>
<tr><td>BF 4B</td><td>CRC</td></tr>
</table>
</td></tr>
</table>

Parameters

Downloading of the parameters

> Reminder: **Direct access addresses** are expressed in **hexadecimal**

> All the parameters values below have a tratment by the ATEQ device as **Long** format with fixed comma (10^{-3}). A **Long** is a two words set.

Identifier N°		D.A. address		Meaning	Value
Dec	Hexa	Read	Write		
01	0001	2001	6001	“FILL TIME” Fill time	0 > 650 seconds
02	0002	2002	6002	“STAB TIME”: Stabilization time	0 > 650 seconds
03	0003	2003	6003	“TEST TIME” Test time	0 > 650 seconds
06	0006	2006	6006	“PRE FILL” Pre fill time	0 > 650 seconds
07	0007	2007	6007	“PRE DUMP” Pre dump time	0 > 650 seconds
09	0009	2009	6009	“DUMP TIME” Dump time	0 > 650 seconds
10	000A	200A	600A	“COUPL. A”: Coupling time 1	0 > 650 seconds
11	000B	200B	600B	“COUPL. B”: Coupling time 2	0 > 650 seconds
17	0011	2011	6011	“Min Vol.” Minimum volume reject level (volume test type measure)	0 > 9999
18	0012	2012	6012	“Max. Vol.” Maximum volume reject level (volume test type measure).	0 > 9999
20	0014	2014	6014	“VOLUME” Part volume.	0 > 9999
21	0015	2015	6015	“TYPE”: Test type	Invalid 0000 Leak 1000 Desensitized 2000 Blockage 3000 Operator 4000
29	001D	201D	601D	“Inter-Cycle”: Time between 2 chained cycles	0 > 650 seconds
48	0030	2030	6030	“DURATION” Maintain time of the result during stamp	0 > 650 seconds
50	0032	2032	6032	“Min FILL” Minimum pressure value	- 9999 > 9999
51	0033	2033	6033	“Max FILL” Maximum pressure value	- 9999 > 9999

Identifier N°		D.A. address		Meaning	Value	
Dec	Hexa	Read	Write			
53	0035	2035	6035	“Press. UNIT” Pressure unit.	Refer to Unit table.	
60	003C	203C	603C	“Test FAIL” Natural reject value of the test part	0 > 9999	
61	003D	203D	603D	“TestREWORK” Natural reject level of the test part in recovery	0 > 9999	
62	003E	203E	603E	“Ref. FAIL” Natural reject level of the reference part	0 > 9999	
63	003F	203F	603F	“Ref.REWORK” Natural reject value of the ref. part in recovery	0 > 9999	
66	0042	2042	6042	“Set FILL” Fill instruction value:	- 9999 > 9999	
67	0043	2043	6043	“Set PreFILL“ Pre-fill instruction value:	- 9999 > 9999	
68	0044			Reserved.		
72	0048	2048	6048	“Drift Unit” Calibration drifts percent.	0 > 100%	
80	0050	2050	6050	“Diff A-Z” Differential auto reset time.	0 > 650 seconds	
102	0066	2066	6066	“BLOW MODE” Type of permanent blowing	Regulator 2 Regulator 1	0000 1000
103	0067	2067	6067	“FILL MODE” Type of fill.	Standard Instruction Ballistic EASY EASY Auto	0000 1000 2000 7000 8000
104	0068	2068	6068	“PreFILL” Type of pre-fill.	Standard Instruction Ballistic EASY EASY Auto	0000 1000 2000 4000 5000
106	006A	206A	606A	“CheckTime” Commutation time of the equalization valve calibration check.	0 > 650 seconds	
107	006B	206B	606B	“% Drift” ATR absorption tolerance.	0 > 100%	
108	006C	206C	606C	“Start” Start value of the transient (ATR).	- 9999 > 9999	
110	006E	206E	606E	“EXT. DUMP” Type of external dump.	Normally close Normally open	0000 1000
111	006F			Reserved.		

Identifier N°		D.A. address		Meaning	Value	
Dec	Hexa	Read	Write			
112	0070	2070	6070	'IN7:" Function attributed to the entry of the special cycles (input 7)	Refer to the "Configurable input values" table at the end of this chapter	
117	0075	2075	6075	"Set Blow" Permanent blowing pressure instruction.	- 9999 > 9999	
118	0076	2076	6076	"REJECT CALC." Original unit for the calculation of rejects in cm3/ min (Pa, Pa/s...).	Refer to Unit table.	
119	0077			Reserved.		
120	0078			Reserved.		
121	0079			Reserved.		
122	007A			Reserved.		
123	007B	207B	607B	"LANGUAGE" Choice of the language.	Default language 2nd predefined language	0000 1000
124	007C	207C	607C	"Max Value" Reject in calibration check.	0 > 9999	
125	007D	207D	607D	"% Drift" Percentage of the calibration check.	0 > 100%	
126	007E	207E	607E	"Max PreFILL" Maximum pressure value in pre-fill.	- 9999 > 9999	
127	007F	207F	607F	"LeakUnit" Reject unit.	Refer to Unit table	
128	0080	2080	6080	"Leak Rate" Instruction value during a calibration.	0 > 9999	
135	0087	2087	6087	"% of T FAIL" Reject level percent of the auto parameter function	0 > 100%	
138	008A	208A	608A	"FILL REG" Regulator number selection for the fill.	Regulator 1 Regulator 2	0000 1000
139	008B	208B	608B	"PRE FILL REG" Regulator number selection for the pre-fill.	Regulator 1 Regulator 2	0000 1000
140	008C			Reserved.		
141	008D			Reserved.		
142	008E			Reserved.		
143	008F			Reserved.		
144	0090	2090	6090	"OUTPUTS CONFIG." Setup of the outputs (standard or compact).	Standard Compact	0000 1000

Identifier N°		D.A. address		Meaning	Value	
Dec	Hexa	Read	Write			
148	0094	2094	6094	“FILTER” Filtering.	0 > 650 seconds	
149	0095	2095	6095	“UNITS” Unit type	SI SAE CUSTOM	0000 1000 2000
161	00A1	20A1	60A1	“Volume UNIT” Volume unit.	Refer to Unit table.	
164	00A4	20A4	60A4	“NEXT PROG.” Number of the following program in sequencing.	1 > 128	
165	00A5	20A5	60A5	“N. OF CYCLES”(PIEZO AUTO AZ menu) Number of cycles between two automatic reset.	0 > 9999	
166	00A6	20A6	60A6	“N. OF MINUTES”(PIEZO AUTO AZ menu) Time between two automatic reset.	0 > 999 minutes	
175	00AF	20AF	60AF	“REGUL. CTRL.” Regulator check during its learning.	Automatic Ext	0000 1000
203	00CB	20CB	60CB	“ELEC. REG.” Activation or not of the built in electronics regulators.	None Reg 1 Reg 2 ALL Reg	0000 1000 2000 3000
232	00E8	20E8	60E8	“ATR DRIFT” Drift transient (ATR).	0 > 100%	
233	00E9	20E9	60E9	“AZ SHORT” Quick auto-zero time.	0 > 650 seconds	
273	0111	2111	6111	“DUMP” Dump time in calibration check mode	0 > 650 seconds	
291	0123	2123	6123	“T.ATR2” Stabilization time for the ATR 2 function	0 > 650 seconds	
295	0127	2127	6127	“DUMP LEVEL” Minimum dump pressure level to reach	- 9999 > 9999	
297	0129	2129	6129	“MAX BLOW” Blowing maximum pressure level	- 9999 > 9999	
298	012A	212A	612A	“MIN BLOW” Blowing minimum pressure level	- 9999 > 9999	
315	013B			Reserved.		
334	014E			Reserved.		
335	014F			Reserved.		
336	0150			Reserved.		
340	0154	2154	6154	“Transient” ATR transient value.	- 9999 > 9999	
349	015D			Reserved.		
353	0161	2161	6161	“Press. UNIT” (configuration/pneumatic menu) General pressure unit	Refer to Unit table.	
354	0162	2162	6162	“LINE P. MIN” Minimum line pressure level	- 9999 > 9999	

Identifier N°		D.A. address		Meaning	Value	
Dec	Hexa	Read	Write			
355	0163			Reserved.		
356	0164			Reserved.		
357	0165			Reserved.		
358	0166			Reserved.		
359	0167			Reserved.		
360	0168			Reserved.		
361	0169			Reserved.		
362	016A			Reserved.		
363	016B			Reserved.		
364	016C	216C	616C	“DISPLAY MODE“ Leak display management	xxxx xxx.x xx.xx x.xxx	0000 1000 2000 3000
366	016E	216E	616E	“MODE” (EXT DUMP menu) Dump mode	Continuous Time	0000 1000
367	016F	216F	616F	“Program” (DUMP OFF menu) Program number of the dump of function	0 > 128	
368	0170	2170	6170	“Tolerance A” Tolerance level A for ntest cycle	0 > 100%	
369	0171	2171	6171	“Tolerance B” Tolerance level B for ntest cycle	0 > 100%	
370	0172			Reserved.		
371	0173	2173	6173	“NAME:”(Units menu) CAL unit personalization	CHAR[5]	
372	0174	2174	6174	“BYPASS” Bypass valve mode selection	Pre-Fill + Fill Pre-Fill Fill	0000 1000 2000
373	0175	2175	6175	“% Cut OFF” Cut off function Percent	0 > 100%	
374	0176	2176	6176	“ATF TIME” Divisor time of ATF function	0 > 650 seconds	
375	0177	2177	6177	‘IN8:” Function attributed to the entry of the special cycles (input 8)	Refer to the “Configurable input values” table at the end of this chapter	

Identifier N°		D.A. address		Meaning	Value	
Dec	Hexa	Read	Write			
376	0178	2178	6178	‘IN9:” Function attributed to the entry of the special cycles (input 9)	Refer to the “Configurable input values” table at the end of this chapter	
377	0179	2179	6179	“MEAS. START” Waiting time for starting the measurement in burst test	0 > 650 seconds	
378	017A	217A	617A	“Time Adj” Adjusting fill time (electronic regulator)	0 > 650 seconds	
379	017B	217B	617B	“USB:” USB mode (printer or supervision)	Supervision Printer Bar code Auto None	0000 1000 2000 3000 4000
380	017C			Reserved.		
405	0195			Reserved.		
406	0196			Reserved.		
407	0197			Reserved.		
408	0198			Reserved.		
409	0199			Reserved.		
410	019A			Reserved.		
455	01C7	21C7	61C7	“DROP PRESS.%” Drop Press function Percent	0 > 100%	
456	01C8	21C8	61C8	“ATM PRESS.“ Atmospheric Pressure	900 > 1100	
457	01C9	21C9	61C9	“TEMP.” Temperature	0 > 800	
458	01CA	21CA	61CA	“DISP. OPT.” Display Option in flow reject	None Pa Display Ambient Temp. Object Temp. Test check ATR Temp. correction Leak offset learning PATM correction	0000 1000 2000 3000 4000 5000 6000 7000 8000
459	01CB	21CB	61CB	“N. OF CYCLES” Number of learning cycle	2 > 9999	
460	01CC	21CC	61CC	“INTER-CYCLE” Time between 2 learning cycle	0 > 650 seconds	
461	01CD	21CD	61CD	“MAX OFFSET” Offset max for learning cycle	0 > 9999	

Identifier N°		D.A. address		Meaning	Value	
Dec	Hexa	Read	Write			
462	01CE	21CE	61CE	“FLOW MASTER” Value of Flow master for learning cycle	0 > 9999	
463	01CF	21CF	61CF	“PRESS MASTER” Value of Pressure master for learning cycle	- 9999 > 9999	
464	01D0	21D0	61D0	“Min. Vol.” Minimum Volume for learning	0 > 9999	
465	01D1	21D1	61D1	“Max. Vol.” Maximum Volume for learning	0 > 9999	
485	01 E5	21E5	61E5	“EXT. ACCES” Security by external access (Fieldbus/Modbus) Reset value with Modbus: → Writing at address 0xC1E5 Reset value with Fieldbus: → Writing one word with ID = 0xC1E5	Read/Write Read Only No Access	0000 1000 2000
486	01 E6	21E6	61E6	“OFFSET” Offset Learning	- 9999 > 9999	

Regulator selection

Regulator selection for fill and pre-fill (word 1, bit n°0 and 1) in the table of the functions bits.

	Fill regulator	Pre-fill regulator
Regulator 1	0	0
Regulator 2	1	1

Configurable input values

F6 V1.3XX			
Input value	Value code	Input value	Value code
Program Selection	0000	Atr Learning Cycle	17000
P1 Sensor Check (*)	4000	Sd Prt Pass Learn	18000
P1 Reg1 Check (*)	5000	Sd Prt Fail Learn	19000
P2 Sensor Check (*)	6000	Volume Comp.	20000
Leak Sensor Check (*)	7000	Test Check Result	21000
Auto Test (*)	8000	Step By Step	22000
Regul. 2 Adjust	9000	Auto Setup	23000
Regul. 1 Adjust	10000	Atr+Custom Learn.	24000
Part Regulator	11000	Code Reader	25000
Infinite Fill	12000	Auto Vol	26000
Piezo Auto Zero	13000	Test On Caps	27000
Custom Unit Learn	14000	Temp.2 Corr. Learn	30000
Custom Unit Check	15000	Temp.2 Sensor Read	31000
Chck+Lrn Cust. Unit	16000		

(*) Available when the **Service special cycle** function is checked.

F6 V2.XXX			
Input value	Value code	Input value	Value code
Program Selection	0000	Chck+Lrn Cust. Unit	24000
Diff Temp. Check (*)	8000	Atr Learning Cycle	25000
Direct P. Check (*)	9000	Sd Prt Pass Learn	26000
P1 Reg1 Check (*)	10000	Sd Prt Fail Learn	27000
Indirect P. Check (*)	11000	Volume Comp.	28000
Leak Sensor Check (*)	12000	Test Check Result	29000
Line P. Sensor Check (*)	13000	Atr+Custom Learn.	30000
Temperature Check (*)	14000	Code Reader	31000
Atm Pressure Check (*)	15000	Auto Vol	32000
Auto Test (*)	16000	Test On Caps	33000
Regul. 2 Adjust	17000	Temp.2 Corr. Learn	36000
Regulator Adj.	18000	Temp.2 Sensor Read	37000
Part. Regulator Adj.	19000	Leak Offset Learn	38000
Infinite Fill	20000	Offset+Vol. Learn	39000
Piezo Auto Zero	21000	N Start	40000
Custom Unit Learn	22000	Sync Test	41000
Custom Unit Check	23000		

(*) Available when the **Service special cycle** function is checked.

Unit table

This list gives all the units used in the instrument in hexadecimal code.

Unit code		Unit
Decimal	Hexadecimal	
0000	0000	cm^3/s
1000	03E8	cm^3/min
2000	07D0	cm^3/h
3000	0BB8	mm^3/s
4000	0FA0	Calibrated Pascal (Pa)
5000	1388	Calibrated Pascal/second (Pa/s)
6000	1770	Pascal (Pa)
7000	1B58	High resolution Pascal (Pa HR)
8000	1F40	Pascal/second (Pa/s)
9000	2328	High resolution Pascal/second (Pa/s HR)
10000	2710	Second (s)
11000	2AF8	Bar
12000	2EE0	KiloPascal (kPa)
13000	32C8	PSI
14000	36B0	Millibar (mbar)
15000	3A98	Mega Pascal (MPa)
16000	3E80	Liter (l)
17000	4268	Calibration check unit
18000	4650	KiloPascal/second (kPa/s)
19000	4A38	Millimeter (mm)
30000	7530	Liter/hour (l/h)
43000	A7F8	D mode Pascal (Pa)
44000	ABE0	Low resolution Pascal (Pa LR)
45000	AFC8	Low resolution Pascal/second (Pa/s LR)
46000	B3B0	Inch^3/s
47000	B798	Inch^3/min
48000	BB80	Inch^3/hour
49000	BF68	Feet^3/hour
50000	C350	Milliliter/second (mm/s)
51000	C738	Milliliter/minute (mm/min)
52000	CB20	Milliliter/hour (mm/h)
53000	CF08	Liter/minute (l/min)
54000	D2F0	Meter^3/hour (m^3/h)
55000	D6D8	Millimeter^3 (mm^3)
56000	DAC0	Centimeter^3 (cm^3)
57000	DEA8	Microsecond (µs)
58000	E290	USA cm^3/s same as the cm^3/s
59000	E678	USA cm^3/min same as the cm^3/min
60000	EA60	USA cm^3/h same as the cm^3/h
61000	EE48	Milliliter (ml)

Unit code		Unit
Decimal	Hexadecimal	
62000	F230	Liter (l)
63000	F618	Inch^3
64000	FA00	Feet^3
68000	01 09A0	oz(US)/s
69000	01 0D88	oz(US)/mn
70000	01 1170	oz(US)/h
71000	01 1558	oz(UK)/s
72000	01 1940	oz(UK)/mn
73000	01 1D28	oz(UK)/h
74000	01 2110	US gallon
75000	01 24F8	UK gallon
76000	01 28E0	PPM
77000	01 2CC8	PPM HR
78000	01 30BO	Calibrated PPM
80000	01 3880	mmCE
81000	01 3C68	mmCE/s
84000	01 4820	SCCM
92000	01 6760	Points
93000	01 6B48	Feet^3/s F620
94000	01 6F30	Feet^3/min F620
95000	01 7318	ACCM MF
96000	01 7700	Inch Mercure (inHg)
99000	01 82B8	Millimeter Mercure (mmHg)
100000	01 86A0	µg H2O/min
102000	01 8E70	No unit

Reading of the parameters

> The parameters are dependents of the program number.
>
> Put the wanted program in edition before executing command.

- *Standard access*

This is an example based on the reading of three parameters:

— **Test type** (identifier number 21)
— **Fill time** (identifier number 1)
— **Stabilization time** (identifier number 2)

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>
— Put in edition the program number on which the parameters have to be read
— Make a <b>Write N*words</b> request of 4 words at the <b>0000h</b> address, with the number of parameters to read (Word 1) and their identifiers (Word 2, 3 and 4).
On network:
<table>
<tr><td>01</td><td>10</td><td>00</td><td>00</td><td>00</td><td>04</td><td>08</td><td>03</td></tr>
<tr><td>00</td><td>15</td><td>00</td><td>01</td><td>00</td><td>02</td><td>00</td><td>F4</td></tr>
<tr><td>36</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>00 00</td><td>Word address (Read parameters)</td></tr>
<tr><td>00 04</td><td>Number of words to write</td></tr>
<tr><td>08</td><td>Number of bytes to write</td></tr>
<tr><td>03 00</td><td>Word 1: write 0003h (3 param. to read)</td></tr>
<tr><td>15 00</td><td>Word 2: write 0015h (identifier n°21)</td></tr>
<tr><td>01 00</td><td>Word 3: write 0001h (identifier n°1)</td></tr>
<tr><td>02 00</td><td>Word 4: write 0002h (identifier n°2)</td></tr>
<tr><td>F4 36</td><td>CRC</td></tr>
</table>
</td><td></td></tr>
<tr><td></td><td>
— Answer to the request:
On network:
<table>
<tr><td>01</td><td>10</td><td>00</td><td>00</td><td>00</td><td>04</td><td>C1</td><td>CA</td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>00 00</td><td>Word address (Read parameters)</td></tr>
<tr><td>00 04</td><td>Number of written words</td></tr>
<tr><td>C1 CA</td><td>CRC</td></tr>
</table>
</td></tr>
<tr><td>
— Make a <b>Read N*words</b> request of 9 words at the <b>0000h</b> address, to retrieve the read parameters with their identifier on a word and their value on a long. (3 parameters * (1 + 2) words = 9 words)
On network:
<table>
<tr><td>01</td><td>03</td><td>00</td><td>00</td><td>00</td><td>09</td><td>85</td><td>CC</td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>00 00</td><td>Word address (Read parameters)</td></tr>
<tr><td>00 09</td><td>Number of words to read</td></tr>
<tr><td>85 CC</td><td>CRC</td></tr>
</table>
</td><td></td></tr>
</table>

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td></td><td>— Answer to the request:
On network:
<table>
<tr><td>01</td><td>03</td><td>12</td><td>15</td><td>00</td><td>E8</td><td>03</td><td>00</td></tr>
<tr><td>00</td><td>01</td><td>00</td><td>F4</td><td>01</td><td>00</td><td>00</td><td>02</td></tr>
<tr><td>00</td><td>E8</td><td>03</td><td>00</td><td>00</td><td>9B</td><td>C2</td><td></td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>12</td><td>Number of read bytes</td></tr>
<tr><td>15 00</td><td>Word 1: read 0015h (identifier n°21)</td></tr>
<tr><td>E8 03</td><td>Word 2 & 3: read 000003E8h</td></tr>
<tr><td>00 00</td><td>(value of test type = 1000(d) → Leak test)</td></tr>
<tr><td>01 00</td><td>Word 4: read 0001h (identifier n°1)</td></tr>
<tr><td>F4 01</td><td>Word 5 & 6: read 000001F4h</td></tr>
<tr><td>00 00</td><td>(value of fill time = 500(d) → 0.5 sec.)</td></tr>
<tr><td>02 00</td><td>Word 7: read 0002h (identifier n°2)</td></tr>
<tr><td>E8 03</td><td>Word 8 & 9: read 000003E8h</td></tr>
<tr><td>00 00</td><td>(value of stab. time = 1000(d) → 1 sec)</td></tr>
<tr><td>9B C2</td><td>CRC</td></tr>
</table></td></tr>
</table>

- *Direct access*

In Direct access, the master can only access to parameters one by one.

This is an example based on the reading of three parameters:

- **Test type** (D.A. address: 2015h)
- **Fill time** (D.A. address: 2001h)
- **Stabilization time** (D.A. address: 2002h)

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>— Put in edition the program number on which the parameters have to be read
— Make a <b>Read N*words</b> request of 2 words at the <b>2015h</b> address.
On network:
<table>
<tr><td>01</td><td>03</td><td>20</td><td>15</td><td>00</td><td>02</td><td>DE</td><td>0F</td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>20 15</td><td>D.A. address</td></tr>
<tr><td>00 02</td><td>Number of words to read</td></tr>
<tr><td>DE 0F</td><td>CRC</td></tr>
</table></td><td></td></tr>
<tr><td></td><td>— Answer to the request:
On network:
<table>
<tr><td>01</td><td>03</td><td>04</td><td>E8</td><td>03</td><td>00</td><td>00</td><td>3F</td></tr>
<tr><td>93</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>04</td><td>Number of read bytes</td></tr>
<tr><td>E8 03</td><td>Word 1 & 2: read 000003E8h</td></tr>
<tr><td>00 00</td><td>(value of test type = 1000(d) → Leak test)</td></tr>
<tr><td>3F 93</td><td>CRC</td></tr>
</table></td></tr>
</table>

Master	Slave

— Make a **Read N*words** request of 2 words at the **2001h** address.

On network:

01	03	20	01	00	02	9E	0B

01	Slave address
03	Function number (Read N*words)
20 01	D.A. address
00 02	Number of words to read
9E 0B	CRC

— Answer to the request:

On network:

01	03	04	F4	01	00	00	99
C3							

01	Slave address
03	Function number (Read N*words)
04	Number of read bytes
F4 01	Word 1 & 2: read 000001F4h
00 00	(value of fill time = 500(d) → 0.5 sec.)
3F 93	CRC

— Make a **Read N*words** request of 2 words at the **2002h** address.

On network:

01	03	20	02	00	02	6E	0B

01	Slave address
03	Function number (Read N*words)
20 02	D.A. address
00 02	Number of words to read
6E 0B	CRC

— Answer to the request:

On network:

01	03	04	E8	03	00	00	3F
93							

01	Slave address
03	Function number (Read N*words)
04	Number of read bytes
E8 03	Word 1 & 2: read 000003E8h
00 00	(value of stab. time = 1000(d) → 1 sec)
3F 93	CRC

Writing of the parameters

> The parameters are dependents of the program number.
> Put the wanted program in edition before executing command.

- *Standard access*

This is an example based on the writing of two parameters:

- **Fill time** (identifier number 1)
- **Stabilization time** (identifier number 2)

Master

— Put in edition the program number on which the parameters have to be written

— Make a **Write N*words** request of 7 words at the **0000h** address, with the number of parameters to read (Word 1), their identifiers (Word 2 and 5) and their value on a long (Word 3 & 4 and 6 & 7).

On network:

01	10	00	7F	00	07	0E	02
00	01	00	E8	03	00	00	02
00	E8	03	00	00	87	AC	

01	Slave address
10	Function number (Write N*words)
00 7F	Word address (Write parameters)
00 07	Number of words to write
0E	Number of bytes to write
02 00	Word 1: write 0002h (2 param. to read)
01 00	Word 2: write 0001h (identifier n°1)
E8 03	Word 3 & 4: write 000003E8h
00 00	(value of fill time = 1000(d) → 1 sec)
02 00	Word 5: write 0002h (identifier n°2)
E8 03	Word 6 & 7: write 000003E8h
00 00	(value of stab. time = 1000(d) → 1 sec)
87 AC	CRC

Slave

— Answer to the request:

On network:

01	10	00	7F	00	07	B0	13

01	Slave address
10	Function number (Write N*words)
00 7F	Word address (Write parameters)
00 07	Number of written words
B0 13	CRC

- *Direct access*

This is an example based on the writing of two parameters:

- **Fill time** (D.A. address: 6001h)
- **Stabilization time** (D.A. address: 6002h)

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>
— Put in edition the program number on which the parameters have to be written
— Make a <b>Write N*words</b> request of 2 words at the <b>6001h</b> address.
On network:
<table>
<tr><td>01</td><td>10</td><td>60</td><td>01</td><td>00</td><td>02</td><td>04</td><td>F4</td></tr>
<tr><td>01</td><td>00</td><td>00</td><td>F9</td><td>91</td><td></td><td></td><td></td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>60 01</td><td>D.A. address</td></tr>
<tr><td>00 02</td><td>Number of words to write</td></tr>
<tr><td>04</td><td>Number of bytes to write</td></tr>
<tr><td>F4 01</td><td>Word 1 &amp; 2: read 000001F4h</td></tr>
<tr><td>00 00</td><td>(value of fill time = 500(d) → 0.5 sec)</td></tr>
<tr><td>F9 91</td><td>CRC</td></tr>
</table>
</td><td></td></tr>
<tr><td></td><td>
— Answer to the request:
On network:
<table>
<tr><td>01</td><td>10</td><td>60</td><td>01</td><td>00</td><td>02</td><td>0E</td><td>08</td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>60 01</td><td>D.A. address</td></tr>
<tr><td>00 02</td><td>Number of written words</td></tr>
<tr><td>0E 08</td><td>CRC</td></tr>
</table>
</td></tr>
<tr><td>
— Make a <b>Write N*words</b> request of 2 words at the <b>6002h</b> address.
On network:
<table>
<tr><td>01</td><td>10</td><td>60</td><td>02</td><td>00</td><td>02</td><td>04</td><td>F4</td></tr>
<tr><td>01</td><td>00</td><td>00</td><td>B9</td><td>84</td><td></td><td></td><td></td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>60 02</td><td>D.A. address</td></tr>
<tr><td>00 02</td><td>Number of words to write</td></tr>
<tr><td>04</td><td>Number of bytes to write</td></tr>
<tr><td>F4 01</td><td>Word 1 &amp; 2: read 000001F4h</td></tr>
<tr><td>00 00</td><td>(value of fill time = 500(d) → 0.5 sec)</td></tr>
<tr><td>B9 84</td><td>CRC</td></tr>
</table>
</td><td></td></tr>
<tr><td></td><td>
— Answer to the request:
On network:
<table>
<tr><td>01</td><td>10</td><td>60</td><td>02</td><td>00</td><td>02</td><td>FE</td><td>08</td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>60 02</td><td>D.A. address</td></tr>
<tr><td>00 02</td><td>Number of written words</td></tr>
<tr><td>FE 08</td><td>CRC</td></tr>
</table>
</td></tr>
</table>

Reading of the program name

This functionality is only available in Standard access.

The personalization is dependent of the program number.
Put the wanted program in edition before executing command.

Example of reading the personalization of a program named “PROGRAMME”:

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>
— Put in edition the program number on which the program name has to be read
— Make a <b>Read N*words</b> request of 6 words at the <b>0120h</b> address
On network:
<table>
<tr><td>01</td><td>03</td><td>01</td><td>20</td><td>00</td><td>06</td><td>C5</td><td>FE</td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>01 20</td><td>Word address (Program name)</td></tr>
<tr><td>00 06</td><td>Number of words to read</td></tr>
<tr><td>C5 FE</td><td>CRC</td></tr>
</table>
</td><td></td></tr>
<tr><td></td><td>
— Answer to the request:
On network:
<table>
<tr><td>01</td><td>03</td><td>0C</td><td>50</td><td>52</td><td>4F</td><td>47</td><td>52</td></tr>
<tr><td>41</td><td>4D</td><td>4D</td><td>45</td><td>00</td><td>41</td><td>44</td><td>AF</td></tr>
<tr><td>43</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>0C</td><td>Number of read bytes</td></tr>
<tr><td>50</td><td>ASCII code for ‘P’ character</td></tr>
<tr><td>52</td><td>ASCII code for ‘R’ character</td></tr>
<tr><td>4F</td><td>ASCII code for ‘O’ character</td></tr>
<tr><td>47</td><td>ASCII code for ‘G’ character</td></tr>
<tr><td>52</td><td>ASCII code for ‘R’ character</td></tr>
<tr><td>41</td><td>ASCII code for ‘A’ character</td></tr>
<tr><td>4D</td><td>ASCII code for ‘M’ character</td></tr>
<tr><td>4D</td><td>ASCII code for ‘M’ character</td></tr>
<tr><td>45</td><td>ASCII code for ‘E’ character</td></tr>
<tr><td>00</td><td>ASCII code for NULL character</td></tr>
<tr><td>41
44</td><td>The datas following ‘00’ until the last word (CRC) do not have any meaning.</td></tr>
<tr><td>AF 43</td><td>CRC</td></tr>
</table>
</td></tr>
</table>

If your program name length is less than 12 characters, you will have a NULL character ‘00’ in the received string that mark the end of personalization. Every data following are meaningless, except for the last word of the frame, that still corresponds to the CRC.

Writing of the program name

This functionality is only available in Standard access.

The personalization is dependent of the program number.
Put the wanted program in edition before executing command.

Example of writing the personalization of a program as “PROG. LEAK”:

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>
— Put in edition the program number on which the program name has to be written
— Make a <b>Write N*words</b> request of 7 words at the <b>0120h</b> address
On network:
<table>
<tr><td>01</td><td>10</td><td>01</td><td>20</td><td>00</td><td>07</td><td>0E</td><td>50</td></tr>
<tr><td>52</td><td>4F</td><td>47</td><td>2E</td><td>20</td><td>4C</td><td>45</td><td>41</td></tr>
<tr><td>4B</td><td>00</td><td>00</td><td>00</td><td>00</td><td>BC</td><td>65</td><td></td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>01 20</td><td>Word address (Program name)</td></tr>
<tr><td>00 07</td><td>Number of words to write</td></tr>
<tr><td>0E</td><td>Number of bytes to write</td></tr>
<tr><td>50</td><td>ASCII code for ‘P’ character</td></tr>
<tr><td>52</td><td>ASCII code for ‘R’ character</td></tr>
<tr><td>4F</td><td>ASCII code for ‘O’ character</td></tr>
<tr><td>47</td><td>ASCII code for ‘G’ character</td></tr>
<tr><td>2E</td><td>ASCII code for ‘.’ character</td></tr>
<tr><td>20</td><td>ASCII code for space character</td></tr>
<tr><td>4C</td><td>ASCII code for ‘L’ character</td></tr>
<tr><td>45</td><td>ASCII code for ‘E’ character</td></tr>
<tr><td>41</td><td>ASCII code for ‘A’ character</td></tr>
<tr><td>4B</td><td>ASCII code for ‘K’ character</td></tr>
<tr><td>00</td><td>ASCII code for NULL character</td></tr>
<tr><td>00</td><td>ASCII code for NULL character</td></tr>
<tr><td>00</td><td>ASCII code for NULL character</td></tr>
<tr><td>00</td><td>ASCII code for NULL character</td></tr>
<tr><td>BC 65</td><td>CRC</td></tr>
</table>
</td><td></td></tr>
<tr><td></td><td>
— Answer to the request:
On network:
<table>
<tr><td>01</td><td>10</td><td>01</td><td>20</td><td>00</td><td>07</td><td>81</td><td>FD</td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>01 20</td><td>Word address (Program name)</td></tr>
<tr><td>00 07</td><td>Number of written words</td></tr>
<tr><td>81 FD</td><td>CRC</td></tr>
</table>
</td></tr>
</table>

The program name has a maximum 12 characters length (without the NULL characters).
Always end your program name with at least one NULL character (00h).

Standard command cycle

Program selection command on the ATEQ device

This functionality is only available in Standard access.

Always subtract 1 from the value of the program number to be selected.
Example: for selecting program number 2, send the value 1 at the address 0200h.

Example for selecting program number 3:

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>

— Make a **Write N*words** request of 1 word at the address **0200h**.

On network:

<table>
<tr><td>01</td><td>10</td><td>02</td><td>00</td><td>00</td><td>01</td><td>02</td><td>02</td></tr>
<tr><td>00</td><td>84</td><td>F0</td><td></td><td></td><td></td><td></td><td></td></tr>
</table>

<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>02 00</td><td>Word address (Program name)</td></tr>
<tr><td>00 01</td><td>Number of words to write</td></tr>
<tr><td>02</td><td>Number of bytes to write</td></tr>
<tr><td>02 00</td><td>Word 1: write 0002h (program n°3)</td></tr>
<tr><td>81 FD</td><td>CRC</td></tr>
</table>

</td><td></td></tr>
<tr><td></td><td>

— Answer to the request:

On network:

<table>
<tr><td>01</td><td>10</td><td>02</td><td>00</td><td>00</td><td>01</td><td>00</td><td>71</td></tr>
</table>

<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>02 00</td><td>Word address (Program name)</td></tr>
<tr><td>00 01</td><td>Number of written words</td></tr>
<tr><td>00 71</td><td>CRC</td></tr>
</table>

</td></tr>
</table>

Start cycle command on the ATEQ device

This functionality is only available in Standard access.

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>

— Select the program you want to start
— Make a **Write bit** request at the address **0001h** and force the Start bit to 1**.**
On network:
<table><tr><td>01</td><td>05</td><td>00</td><td>01</td><td>FF</td><td>00</td><td>DD</td><td>FA</td></tr></table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>05</td><td>Function number (Write bit)</td></tr>
<tr><td>00 01</td><td>Bit address (Start bit)</td></tr>
<tr><td>FF 00</td><td>Force bit to 1</td></tr>
<tr><td>DD FA</td><td>CRC</td></tr>
</table>
</td><td></td></tr>
<tr><td></td><td>

— Answer to the request (exactly the same as the request):
On network:
<table><tr><td>01</td><td>05</td><td>00</td><td>01</td><td>FF</td><td>00</td><td>DD</td><td>FA</td></tr></table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>05</td><td>Function number (Write bit)</td></tr>
<tr><td>00 01</td><td>Bit address (Start bit)</td></tr>
<tr><td>FF 00</td><td>Force bit to 1</td></tr>
<tr><td>DD FA</td><td>CRC</td></tr>
</table>
</td></tr>
</table>

Reset command on the ATEQ device

ℹ This functionality is only available in Standard access.

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>— Make a <b>Write bit</b> request at the address <b>0000h</b> and force the Reset bit to 1<b>.</b>
On network:
<table><tr><td>01</td><td>05</td><td>00</td><td>00</td><td>FF</td><td>00</td><td>8C</td><td>3A</td></tr></table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>05</td><td>Function number (Write bit)</td></tr>
<tr><td>00 00</td><td>Bit address (Reset bit)</td></tr>
<tr><td>FF 00</td><td>Force bit to 1</td></tr>
<tr><td>8C 3A</td><td>CRC</td></tr>
</table></td><td></td></tr>
<tr><td></td><td>— Answer to the request (exactly the same as the request):
On network:
<table><tr><td>01</td><td>05</td><td>00</td><td>00</td><td>FF</td><td>00</td><td>8C</td><td>3A</td></tr></table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>05</td><td>Function number (Write bit)</td></tr>
<tr><td>00 00</td><td>Bit address (Reset bit)</td></tr>
<tr><td>FF 00</td><td>Force bit to 1</td></tr>
<tr><td>8C 3A</td><td>CRC</td></tr>
</table></td></tr>
</table>

Special cycles

Special cycle table

Write the identifier number of the wanted special cycle at the address 0201h and its instruction if necessary.
Word 1 = identifier number of the special cycle
Word 2 = instruction for the special cycle

Numb	Special cycle	Numb	Special cycle
1	ATR Learning Cycle	17	Sd Prt FAIL Learn
2	Test Check Result	18	Direct P. Check (*) (**)
3	AUTO TEST	19	Leak Sensor Check (*) (**)
4	Custom Unit Learn	20	Reserved
5	Custom Unit Check	21	Reserved
6	ATR+Custom Learn	22	Reserved
7	Piezo auto zero Reg 1	23	No special cycle
8	Piezo auto zero Reg 2	24	Reserved
9	Regul. 2 adjust	25	Line P. Sensor Check (*) (**)
10	Regulator Adj	26	No special cycle
11	Infinite fill	27	Reserved
12	Volume Comp	28	Reserved
13	Auto Vol	29	Temperature check (*) (**)
14	No special cycle	30	Atm Pressure Check (*) (**)
15	No special cycle	31	No special cycle
16	Sd Prt PASS Learn		

(*) For version ≥ 1.400 only.
(**) Appears with the **Service special cycle** function checked.

Auto-zero on the ATEQ device

This functionality is only available in Standard access.

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>

— Select the program on which you want to make the auto zero

— Make a **Write N*words** request of 1 word at the address **0201h** and pass the value of the auto zero special cycle (n°7).

On network:

<table>
<tr><td>01</td><td>10</td><td>02</td><td>01</td><td>00</td><td>01</td><td>02</td><td>07</td></tr>
<tr><td>00</td><td>86</td><td>71</td><td></td><td></td><td></td><td></td><td></td></tr>
</table>

<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>02 01</td><td>Word address (Program name)</td></tr>
<tr><td>00 01</td><td>Number of words to write</td></tr>
<tr><td>02</td><td>Number of bytes to write</td></tr>
<tr><td>07 00</td><td>Word 1: write 0007h (spe. cycle n°7)</td></tr>
<tr><td>86 71</td><td>CRC</td></tr>
</table>

</td><td></td></tr>
<tr><td></td><td>

— Answer to the request:

On network:

<table>
<tr><td>01</td><td>10</td><td>02</td><td>01</td><td>00</td><td>01</td><td>51</td><td>B1</td></tr>
</table>

<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>10</td><td>Function number (Write N*words)</td></tr>
<tr><td>02 01</td><td>Word address (Program name)</td></tr>
<tr><td>00 01</td><td>Number of written words</td></tr>
<tr><td>51 B1</td><td>CRC</td></tr>
</table>

</td></tr>
<tr><td>— Make a start request to launch the special cycle.</td><td></td></tr>
</table>

FIFO results

FIFO list results structure

At the end of each cycle, a result is stored as an array of 40 words contained in a FIFO of 8 results. This result includes the final state of the instrument (relays position, alarm signal, indicators state…), but also of the test (units, values measured for pressure and flow).

The results are in the memory of the instrument. To obtain them, it is necessary to carry out a “Read FIFO results” request.

Words	Meaning	Type	Bytes	Coeff
1	Program number.	Word	2	
2	Test type.	Word	2	
3	Image of the relays: Bit 0 = 1: pass part. Bit 1 = 1: fail part, maximum flow reject. Bit 2 = 1: fail part, minimum flow reject. Bit 3 = 1: alarm. Bit 4 = 1: unused. Bit 5 = 1: reserved. Bit 6 = 1: unused. Bit 7 = 1: unused.	Word	2	
4	Alarm code (refer to the alarm codes table).	Word	2	
5	Pressure low part word.	Long	4	x1000
6	Pressure high part word.			
7	Pressure unit code low part word (refer to units table).	Long	4	x1000
8	Pressure unit code high part word (refer to units table).			
9	Leak low section word.	Long	4	x1000
10	Leak high section word.			
11	Leak unit code low part word (refer to. Units table).	Long	4	x1000
12	Leak unit code high part word (refer to. Units table).			
13	Pressure piezo 2 low part word.	Long	4	x1000
14	Pressure piezo 2 high part word.			
15	Pressure piezo 2 unit code low part word (refer to units table).	Long	4	x1000
16	Pressure piezo 2 unit code high part word (refer to units table).			
17	Test check result low part word.	Long	4	x1000
18	Test check result high part word.			
19	Test check result unit code low part word (refer to units table).	Long	4	x1000
20	Test check result unit code high part word (refer to units table).			
21	Large Leak low part word.	Long	4	x1000
22	Large Leak high part word.			
23	Large Leak unit code low part word (refer to units table).	Long	4	x1000
24	Large Leak unit code high part word (refer to units table).			

Words	Meaning	Type	Bytes	Coeff
	V2.xxx Only			
25	Pa – Pa/s Leak result low part word	Long	4	x1000
26	Pa – Pa/s Leak result high part word			
27 - 36	*Unused*			
37	Atmospheric pressure in hPa low part word	Long	4	x1000
38	Atmospheric pressure in hPa high part word			
39	Temperature in °C low part word	Long	4	x1000
40	Temperature in °C high part word			

All the numerical values are treated with **Long** format with fixed comma (10^{-3}). Thus, they must be multiplied by 1000 to get the value in units (see examples in “Basic notions” section).

Step table

This table represents the codes of the steps in the cycle.

Code		Steps
Decimal	Hexadecimal	
0	0000	Pre-fill.
1	0001	Pre-dump.
2	0002	Sealed component fill.
3	0003	Sealed component stabilization.
4	0004	Fill.
5	0005	Stabilization.
6	0006	Test.
7	0007	Dump.
65535	FFFF	No steps in progress.

Alarm codes table

This list gives all the alarms in hexadecimal code.

Identifier n°		Alarm
Decimal	Hexadecimal	
0	0000	No alarm.
1	0001	Pressure switched alarm (test pressure too high).
2	0002	Pressure switch (test pressure too small).
3	0003	Large leak on TEST (EEEE).
4	0004	Large leak on REF (MMMM).
7	0007	Sensor out of order (overrun).
8	0008	ATR error.
9	0009	ATR drift.
10	000A	CAL error.
11	000B	Volume too small (sealed component).
12	000C	Volume too large (sealed component).
14	000E	Equalization valve switching error.
43	002B	Pressure too high.
44	002C	Pressure too low.
45	002D	Piezo sensor out of order.
46	002E	Dump error.
47	002F	CAL drift error.
48	0030	Calibration check error.
49	0031	Leak in calibration check too high.
50	0032	Leak in calibration check too low.
51	0033	Sealed component learning error.
64	0040	Piezo sensor 2 out of order.
65	0041	Pressure Piezo 2 too high.
66	0042	Pressure Piezo 2 too low.
68	0044	Pressure Piezo 2 switched alarm (test pressure too high).
69	0045	Pressure Piezo 2 switch (test pressure too small).
72	0048	Learning Electrical Regulator Default.

Cycle results reading (last 8 results in FIFO)

This functionality is only available in Standard access.

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>— Make a <b>Read N*words</b> request of 40 words at the <b>0010h</b> address.
On network:
<table><tr><td>01</td><td>03</td><td>00</td><td>10</td><td>00</td><td>28</td><td>44</td><td>11</td></tr></table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>00 10</td><td>Word address (FIFO result)</td></tr>
<tr><td>00 28</td><td>Number of words to read</td></tr>
<tr><td>44 11</td><td>CRC</td></tr>
</table></td><td></td></tr>
<tr><td></td><td>— Answer to the request:
On network:
<table><tr><td>01</td><td>03</td><td>50</td><td><i>FIFO result structure on 40 words</i></td><td><i>CRC</i></td></tr></table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>50</td><td>Number of read bytes</td></tr>
<tr><td></td><td>FIFO result structure on 40 words</td></tr>
<tr><td></td><td>CRC</td></tr>
</table></td></tr>
</table>

Reset FIFO results

This functionality is only available in Standard access.

This command resets the 8 last cycle's results available in the FIFO.

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr>
<td>

— Make a **Write bit** request at the address **0002h** and force the Reset FIFO bit to 1**.**

On network:

<table><tr><td>01</td><td>05</td><td>00</td><td>02</td><td>FF</td><td>00</td><td>2D</td><td>FA</td></tr></table>

<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>05</td><td>Function number (Write bit)</td></tr>
<tr><td>00 00</td><td>Bit address (Reset FIFO bit)</td></tr>
<tr><td>FF 00</td><td>Force bit to 1</td></tr>
<tr><td>2D FA</td><td>CRC</td></tr>
</table>

</td>
<td></td>
</tr>
<tr>
<td></td>
<td>

— Answer to the request (exactly the same as the request):

On network:

<table><tr><td>01</td><td>05</td><td>00</td><td>02</td><td>FF</td><td>00</td><td>2D</td><td>FA</td></tr></table>

<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>05</td><td>Function number (Write bit)</td></tr>
<tr><td>00 00</td><td>Bit address (Reset FIFO bit)</td></tr>
<tr><td>FF 00</td><td>Force bit to 1</td></tr>
<tr><td>2D FA</td><td>CRC</td></tr>
</table>

</td>
</tr>
</table>

Last results

Last results structure

Reminder: **Direct access addresses** are expressed in **hexadecimal**

At the end of each cycle, the last result is as an array of 40 words. This result includes the final state of the instrument (relays position, alarm signal, indicators state…), but also of the test (units, values measured for the pressure and the flow).

The last result is in the memory of the instrument. To obtain them, it is necessary to carry out a "Read last results" request.

Words	D.A address Read	Meaning	Type	Bytes	Coeff
1	2301	Program number.	Word	2	
2	2302	Test type.	Word	2	
3	2303	Image of the relays: Bit 0 = 1: pass part. Bit 1 = 1: fail part, maximum flow reject. Bit 2 = 1: fail part, minimum flow reject. Bit 3 = 1: alarm. Bit 4 = 1: unused. Bit 5 = 1: reserved. Bit 6 = 1: unused. Bit 7 = 1: unused.	Word	2	
4	2304	Alarm code (refer to the alarm codes table).	Word	2	
5	2305	Pressure low part word.	Long	4	x1000
6	2306	Pressure high part word.			
7	2307	Pressure unit code low part word (refer to units table).	Long	4	x1000
8	2308	Pressure unit code high part word (refer to units table).			
9	2309	Leak low section word.	Long	4	x1000
10	230A	Leak high section word.			
11	230B	Leak unit code low part word (refer to. Units table).	Long	4	x1000
12	230C	Leak unit code high part word (refer to. Units table).			
13	230D	Pressure piezo 2 low part word.	Long	4	x1000
14	230E	Pressure piezo 2 high part word.			
15	230F	Pressure piezo 2 unit code low part word (refer to units table).	Long	4	x1000
16	2310	Pressure piezo 2 unit code high part word (refer to units table).			
17	2311	Test check result low part word.	Long	4	x1000
18	2312	Test check result high part word.			
19	2313	Test check result unit code low part word (refer to units table).	Long	4	x1000
20	2314	Test check result unit code high part word (refer to units table).			
21	2315	Large Leak low part word.	Long	4	x1000
22	2316	Large Leak high part word.			
23	2317	Large Leak unit code low part word (refer to units table).	Long	4	x1000
24	2318	Large Leak unit code high part word (refer to units table).			

Words	D.A. address Read	Meaning	Type	Bytes	Coeff
V2.xxx Only					
25	2319	Pa – Pa/s Leak result low part word	Long	4	x1000
26	231A	Pa – Pa/s Leak result high part word			
27 - 36		*Unused*			
37	2325	Atmospheric pressure in hPa low part word	Long	4	x1000
38	2326	Atmospheric pressure in hPa high part word			
39	2327	Temperature in °C low part word	Long	4	x1000
40	2328	Temperature in °C high part word			

All the numerical values are treated with **Long** format with fixed comma (10^{-3}). Thus, they must be multiplied by 1000 to get the value in units (see examples in “Basic notions” section).

Last results reading

> For using this function, it is important to:
> — Having done a start on the instrument before (“End of cycle” bit on in the relay status)
> — Not having done a reset of the FIFO

- *Standard access*

Example of reading the entire last result structure:

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>— Make a <b>Read N*words</b> request of 40 words at the <b>0011h</b> address.
On network:
<table><tr><td>01</td><td>03</td><td>00</td><td>11</td><td>00</td><td>28</td><td>15</td><td>D1</td></tr></table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>00 11</td><td>Word address (Last result)</td></tr>
<tr><td>00 28</td><td>Number of words to read</td></tr>
<tr><td>15 D1</td><td>CRC</td></tr>
</table></td><td></td></tr>
<tr><td></td><td>— Answer to the request:
On network:
<table><tr><td>01</td><td>03</td><td>50</td><td><i>Last result structure on 40 words</i></td><td><i>CRC</i></td></tr></table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>50</td><td>Number of read bytes</td></tr>
<tr><td></td><td>Last result structure on 40 words</td></tr>
<tr><td></td><td>CRC</td></tr>
</table></td></tr>
</table>

- *Direct access*

In Direct access, the master can only access to parameters one by one.

This is an example on the reading of the pressure unit code in the last result:

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>
— Make a <b>Read N*words</b> request of 2 words at the <b>2307h</b> address.
On network:
<table><tr><td>01</td><td>03</td><td>23</td><td>07</td><td>00</td><td>02</td><td>7E</td><td>4E</td></tr></table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>23 07</td><td>D.A. address</td></tr>
<tr><td>00 02</td><td>Number of words to read</td></tr>
<tr><td>7E 4E</td><td>CRC</td></tr>
</table>
</td><td></td></tr>
<tr><td></td><td>
— Answer to the request:
On network:
<table>
<tr><td>01</td><td>03</td><td>04</td><td>F8</td><td>2A</td><td>00</td><td>00</td><td>EA</td></tr>
<tr><td>9B</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Write N*words)</td></tr>
<tr><td>04</td><td>Number of read bytes</td></tr>
<tr><td>F8 2A</td><td>Word 1 &amp; 2: read 00002AF8h</td></tr>
<tr><td>00 00</td><td>(Pressure unit code = 11000 → bar)</td></tr>
<tr><td>EA 9B</td><td>CRC</td></tr>
</table>
</td></tr>
</table>

Real time

Status and real time measures

Reminder: Direct access addresses are expressed in hexadecimal

The real time measurement is used for display curve or values during the cycle and not for the final measurement.

Do not take or use the final results in this section, it is just to see the status of the device for the "Cycle end" (bit 5) and "Key presence" (bit 15) information.
For the results, use only the FIFO list results structure or the Last results structure (see above)

Words	D.A. address Read	Meaning	Type	Bytes	Coeff
1	2201	Program number.	Word	2	
2	2202	Number of results waiting in the results FIFO memory.	Word	2	
3	2203	Test type.	Word	2	
4	2204	Status: Bit 0 = 1: pass part. Bit 1 = 1: fail part maximum flow. Bit 2 = 1: fail part minimum flow. Bit 3 = 1: alarm. Bit 4 = 1: pressure error.	Do not use these results while the Bit 5 (cycle end is not 1). Use only Bit 5 (cycle end) and Bit 15 (key presence).		
		Bit 5 = 1: cycle end.	Word	2	
		Bit 6 = 1: recoverable part. Bit 7 = 1: CAL error or drift. Bit 8 = 1: Calibration check error Bit 9 = 1: ATR error or drift. Bits 10 / 11 / 12 / 13 / 14 = 1: *Unused*. Bit 15 = 1: key presence.	Do not use these results while the Bit 5 (cycle end is not 1). Use only Bit 5 (cycle end) and Bit 15 (key presence).		
5	2205	Step code (refer to steps table).	Word	2	
6	2206	Low pressure section word.	Long	4	x1000
7	2207	High pressure section word.			
8	2208	Pressure unit code low part word (see units table).	Long	4	x1000
9	2209	Pressure unit code high part word (see units table).			
10	220A	Flow low section word.	Long	4	x1000
11	220B	Flow high section word.			
12	220C	Flow unit code low part word (refer to. Units table).	Long	4	x1000
13	220D	Flow unit code high part word (refer to. Units table).			

Status and real time measures reading

> For using this function, it is important to:
> — Having done a start on the instrument before (“End of cycle” bit on in the relay status)
> — Not having done a reset of the FIFO

- *Standard access*

Example of reading the entire status and real time measures structure:

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>
— Make a <b>Read N*words</b> request of 13 words at the <b>0010h</b> address.
On network:
<table>
<tr><td>01</td><td>03</td><td>00</td><td>30</td><td>00</td><td>0D</td><td>84</td><td>00</td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>00 30</td><td>Word address (Real time result)</td></tr>
<tr><td>00 0D</td><td>Number of words to read</td></tr>
<tr><td>84 00</td><td>CRC</td></tr>
</table>
</td><td></td></tr>
<tr><td></td><td>
— Answer to the request:
On network:
<table>
<tr><td>01</td><td>03</td><td>1A</td><td>02</td><td>00</td><td>00</td><td>00</td><td>01</td></tr>
<tr><td>00</td><td>21</td><td>80</td><td>FF</td><td>FF</td><td>00</td><td>00</td><td>00</td></tr>
<tr><td>00</td><td>F8</td><td>2A</td><td>00</td><td>00</td><td>08</td><td>CF</td><td>00</td></tr>
<tr><td>00</td><td>70</td><td>17</td><td>00</td><td>00</td><td>AE</td><td>95</td><td></td></tr>
</table>
<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>1A</td><td>Number of read bytes</td></tr>
<tr><td>02 00</td><td>Word 1: read 0002h (prog. N°3)</td></tr>
<tr><td>00 00</td><td>Word 2: read 0000h (num. of res. in FIFO)</td></tr>
<tr><td>01 00</td><td>Word 3: read 0001h (type test = leak)</td></tr>
<tr><td>21 80</td><td>Word 4: read 8021h (status)</td></tr>
<tr><td>FF FF</td><td>Word 5: read FFFFh (step code)</td></tr>
<tr><td>00 00
00 00</td><td>Word 6 & 7: read 00000000h
(pressure value = 0)</td></tr>
<tr><td>F8 2A
00 00</td><td>Word 8 & 9: read 00002AF8h
(pressure unit = 11000 → bar)</td></tr>
<tr><td>08 CF
00 00</td><td>Word 10 & 11: read 000008CFh
(leak value = 53000 → 53)</td></tr>
<tr><td>70 17
00 00</td><td>Word 12 & 13: read 00001770h
(leak unit = 6000 → Pascal)</td></tr>
<tr><td>AE 95</td><td>CRC</td></tr>
</table>
</td></tr>
</table>

- *Direct access*

In Direct access, the master can only access to parameters one by one.

This is an example on the reading of the end of cycle bit in the status:

<table>
<tr><th>Master</th><th>Slave</th></tr>
<tr><td>

— Make a **Read N*words** request of 1 word at the **2204h** address.

On network:

<table><tr><td>01</td><td>03</td><td>22</td><td>04</td><td>00</td><td>01</td><td>CF</td><td>B3</td></tr></table>

<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Read N*words)</td></tr>
<tr><td>22 04</td><td>D.A. address</td></tr>
<tr><td>00 01</td><td>Number of words to read</td></tr>
<tr><td>CF B3</td><td>CRC</td></tr>
</table>

</td><td></td></tr>
<tr><td></td><td>

— Answer to the request:

On network:

<table><tr><td>01</td><td>03</td><td>02</td><td>21</td><td>80</td><td>A1</td><td>B4</td></tr></table>

<table>
<tr><td>01</td><td>Slave address</td></tr>
<tr><td>03</td><td>Function number (Write N*words)</td></tr>
<tr><td>02</td><td>Number of read bytes</td></tr>
<tr><td>21 80</td><td>Word 1: read 8021h
(cycle end = 8021 & 0020 = 1)</td></tr>
<tr><td>A1 B4</td><td>CRC</td></tr>
</table>

</td></tr>
</table>