ATEQ

G6 Series – CC-Link

Manual

Table of Contents

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

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.

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 G6 device on the CC-Link 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

Connect your ATEQ equipment to the CC-Link network using its CC-Link connectors and compatible cables.

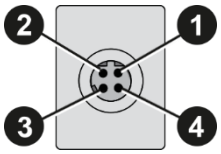
HARDWARE CONFIGURATION

Your device has a CC-Link internal board and two CC-Link connectors.
The CC-Link internal board is located inside your device.

Your device has two M12 D coded type connector – 4 pins female connector.

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

M12 D coded type connector – 4 pins female connector

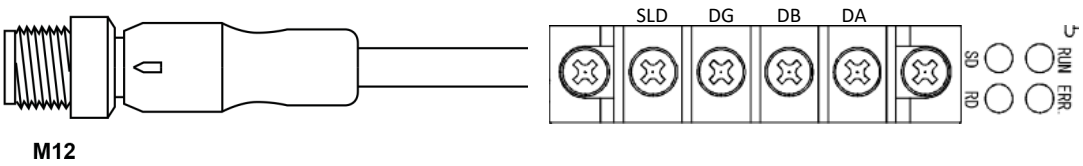


Pin number	Signal
1	DA (brown)
2	DB (white)
3	DG (blue)
4	SLD (black)

CABLING

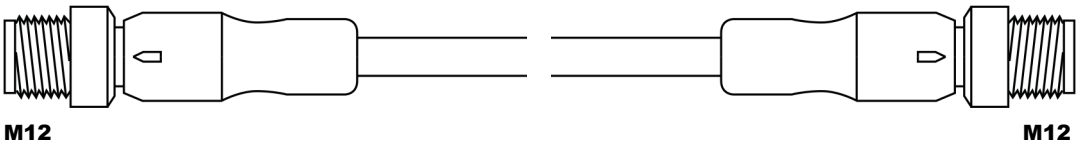
From slave to master MELSEC board

Cable with male M12 D coded type connector (4pins) and an MELSEC connector (4pins).

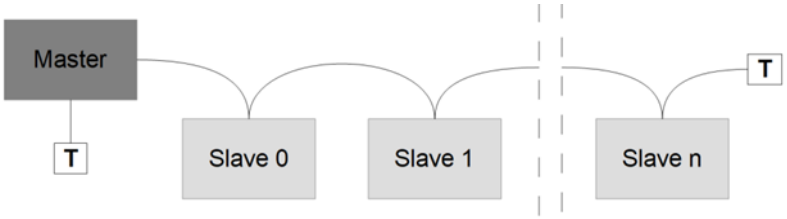


Between two slaves

Cable with two male M12 D coded type connector (4pins).



CC-Link network link



T : terminal resistor between DA & DB of about 100Ω

Configuration of the ATEQ device (slave)

Use this procedure to set the configuration mode of your device.

This configuration can be done with the front panel of your ATEQ device or with the ATEQ Fieldbus Configurator software

SETUP OF THE CC-LINK CONFIGURATION MODE

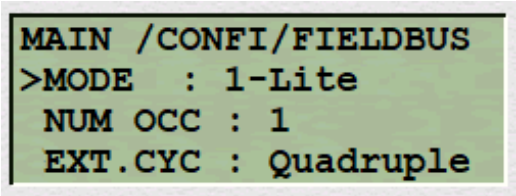
Five configuration modes are available:

Mode	Configuration mode	Number of occupied station	Extended cycle	Length (in bytes)	Use
5	**Standard mode** *(6th Serie only)*	4	Octuple	368	For the inputs/outputs, real time measurements, the live cycle results, 21 parameters management, functions and extended menu bits and extended cycle results
4	**Standard less mode**	3	Octuple	272	For the inputs/outputs, real time measurements, the live cycle results, 21 parameters management, functions and extended menu bits
3	**Medium more mode**	2	Octuple	176	For the inputs/outputs, the real time measurements, the live cycle results, 10 parameters management, functions and extended menu bits
2	**Medium mode**	1	Octuple	80	For the inputs/outputs, the real time measurements and cycle results
1	**Light mode**	1	Quadruple	40	For the inputs/outputs and the real time measurements

Number of bytes available for the communication depends on the selected configuration mode. Therefore, the usable functionalities also depend on it:

WRITE					
Mode	Commands	Extended menu bits	Function bits	Parameters	Program name
5	ALL	YES	YES	21	YES
4	ALL	YES	YES	21	YES
3	ALL	YES	YES	10	YES
2	Start, Reset, Auto-zero, Select Prog, FIFO & Last results	NO	NO	NO	NO
1	Start, Reset, Auto-zero, Select Prog	NO	NO	NO	NO

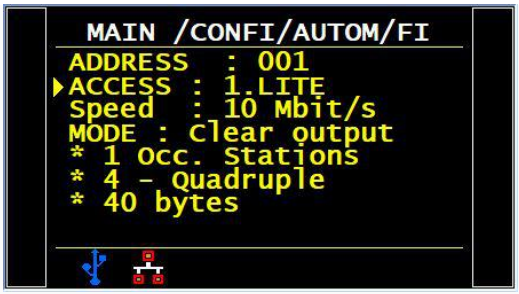
READ								
Mode	Echo / Error	Relay status	Real time measurements	Results	Extended menu bits	Function bits	Parameters	Program name
5	ALL	YES	YES	Extended	YES	YES	21	YES
4	ALL	YES	YES	Standard	YES	YES	21	YES
3	ALL	YES	YES	Standard	YES	YES	10	YES
2	Start, Reset, Auto-zero, Select Prog, FIFO & Last results	YES	YES	Standard	NO	NO	NO	NO
1	Start, Reset, Auto-zero, Select Prog	YES	YES	NO	NO	NO	NO	NO



5th Series:

From the **MAIN MENU** screen of your ATEQ device go to

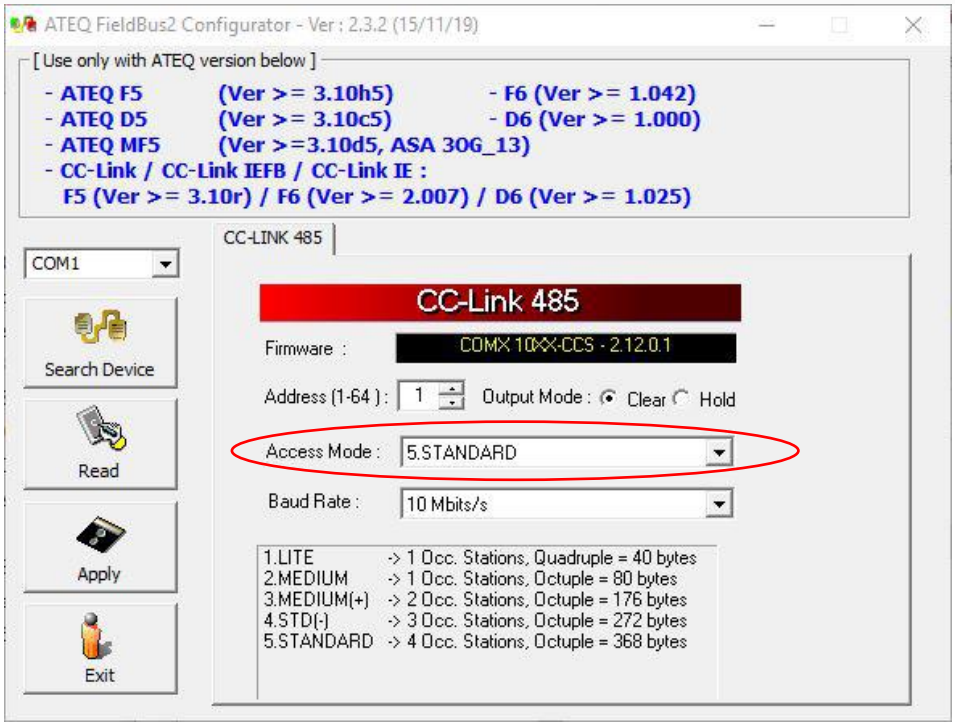
- **CONFIGURATION**
- **FIELDBUS**
- **MODE**



6th Series:

From the **MAIN MENU** screen of your ATEQ device go to

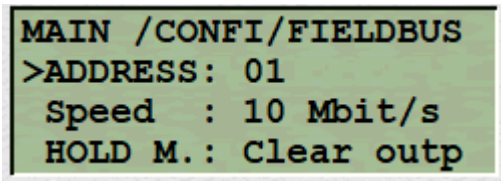
- **CONFIGURATION**
- **AUTOMATISM**
- **FIELDBUS**
- **ACCESS**



SETUP OF THE STATION ADDRESS

Station address goes from **1 to 64**.

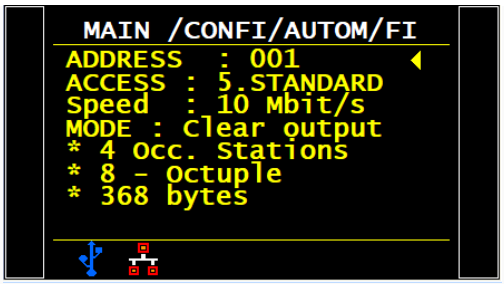
The station address configured in the ATEQ edvice must be the same as the one provided to the master.



5th Series:

From the **MAIN MENU** screen of your ATEQ device go to

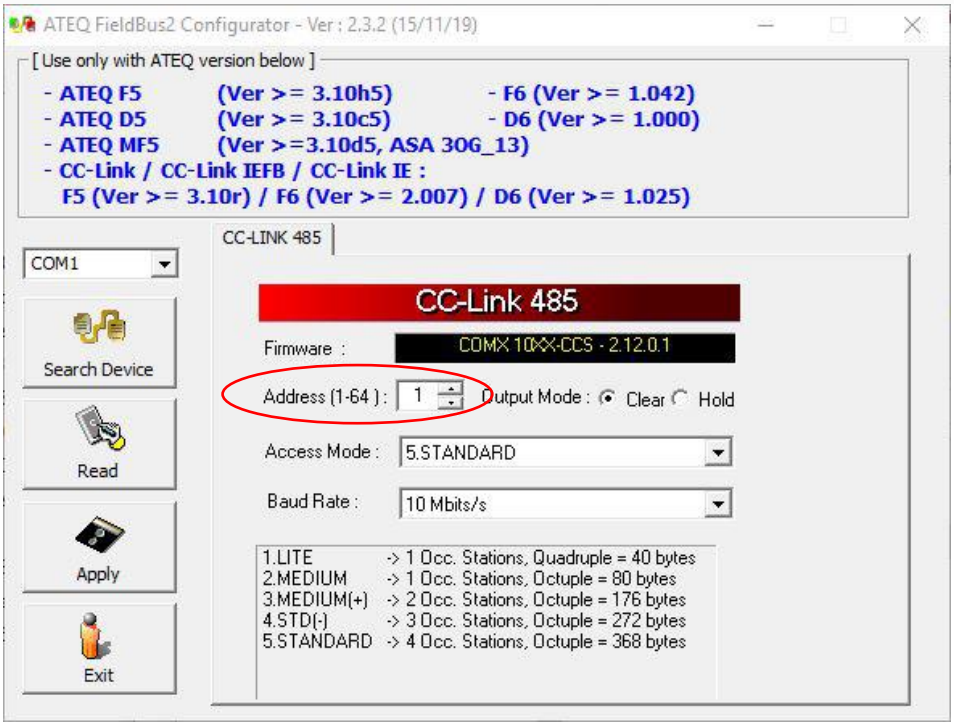
- **CONFIGURATION**
- **FIELDBUS**
- **ADDRESS**



6th Series:

From the **MAIN MENU** screen of your ATEQ device go to

- **CONFIGURATION**
- **AUTOMATISM**
- **FIELDBUS**
- **ADDRESS**

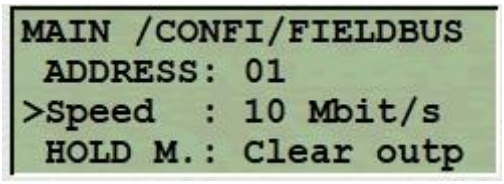


SETUP OF THE COMMUNICATION SPEED

The communication speed can be set to:

- **156 kbits/s**
- **625 kbits/s**
- **2.5 Mbits/s**
- **5 Mbits/s**
- **10 Mbits/s**

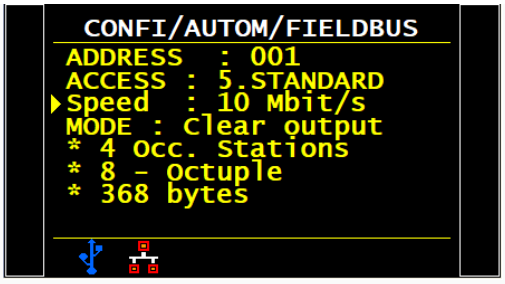
The communication speed configured in the ATEQ device must be the same as the one provided to the master.



5th Series:

From the **MAIN MENU** screen of your ATEQ device go to

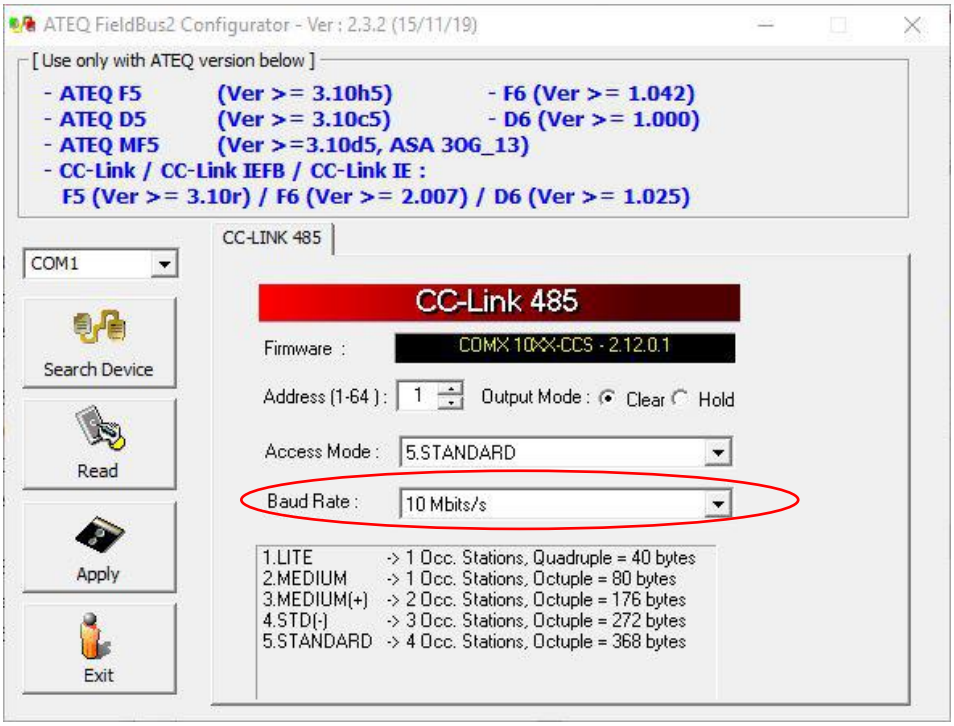
- **CONFIGURATION**
- **FIELDBUS**
- **Speed**



6th Series:

From the **MAIN MENU** screen of your ATEQ device go to

- **CONFIGURATION**
- **AUTOMATISM**
- **FIELDBUS**
- **Speed**

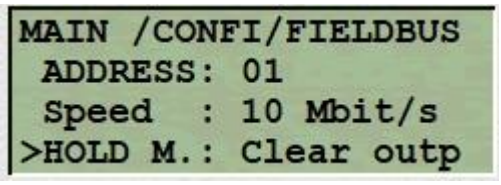


SETUP OF THE OUTPUT MODE

The output mode is used to command the output to hold their value or to clear it after being set.
The output modes are:

- **Clear output**
- **Hold output**

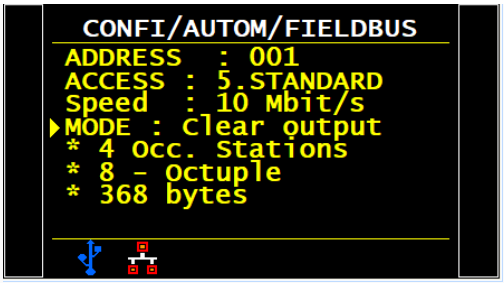
The output mode configured in the ATEQ device must be the same as the one provided to the master.



5th Series:

From the **MAIN MENU** screen of your ATEQ device go to

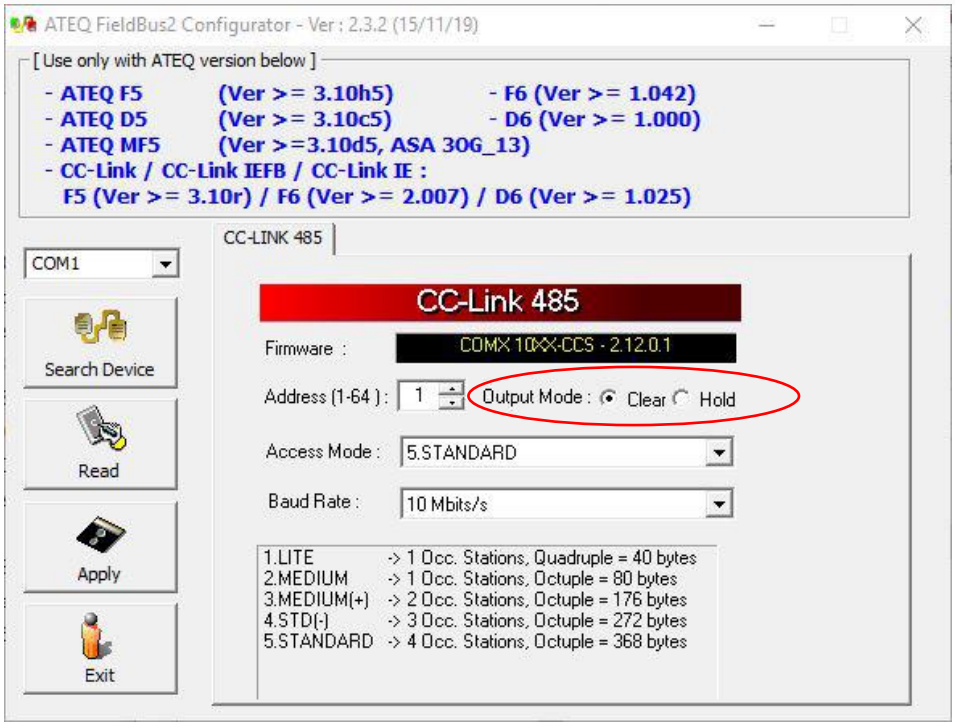
- **CONFIGURATION**
- **FIELDBUS**
- **HOLD M.**



6th Series:

From the **MAIN MENU** screen of your ATEQ device go to

- **CONFIGURATION**
- **AUTOMATISM**
- **FIELDBUS**
- **MODE**

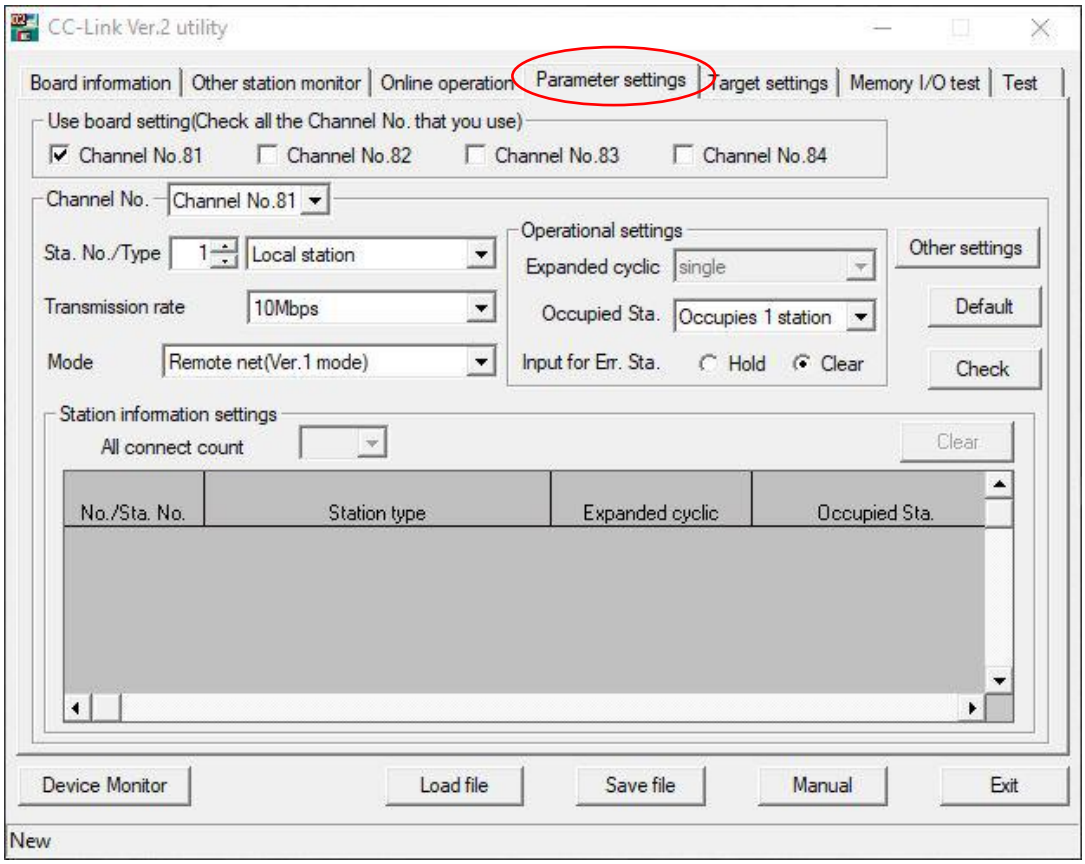


Configuration of the master

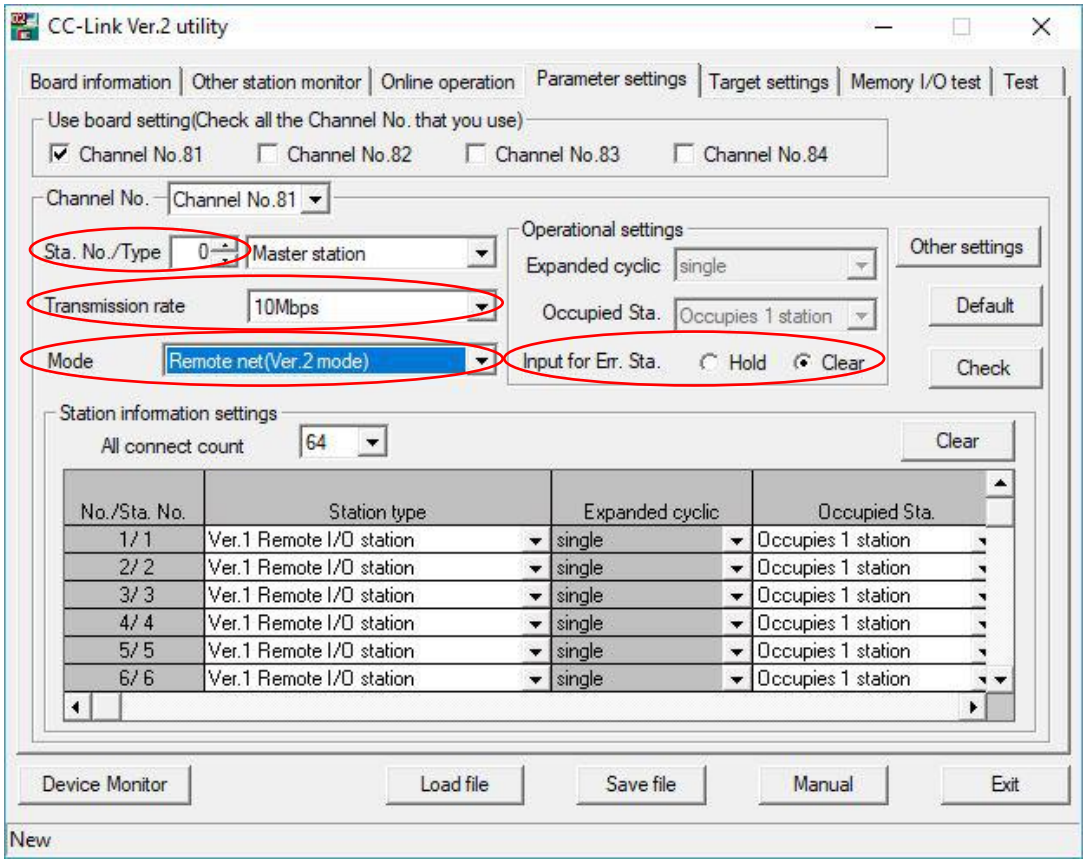
For the master configuration, you will use the “MELSEC CC-Link Ver.2 utility” software and a MELSEC master board put in your computer.

CONFIGURE THE MELSEC BOARD AS MASTER

Open **MELSEC CC-Link Ver.2 utility** software and go into the **Parameter setting** tab

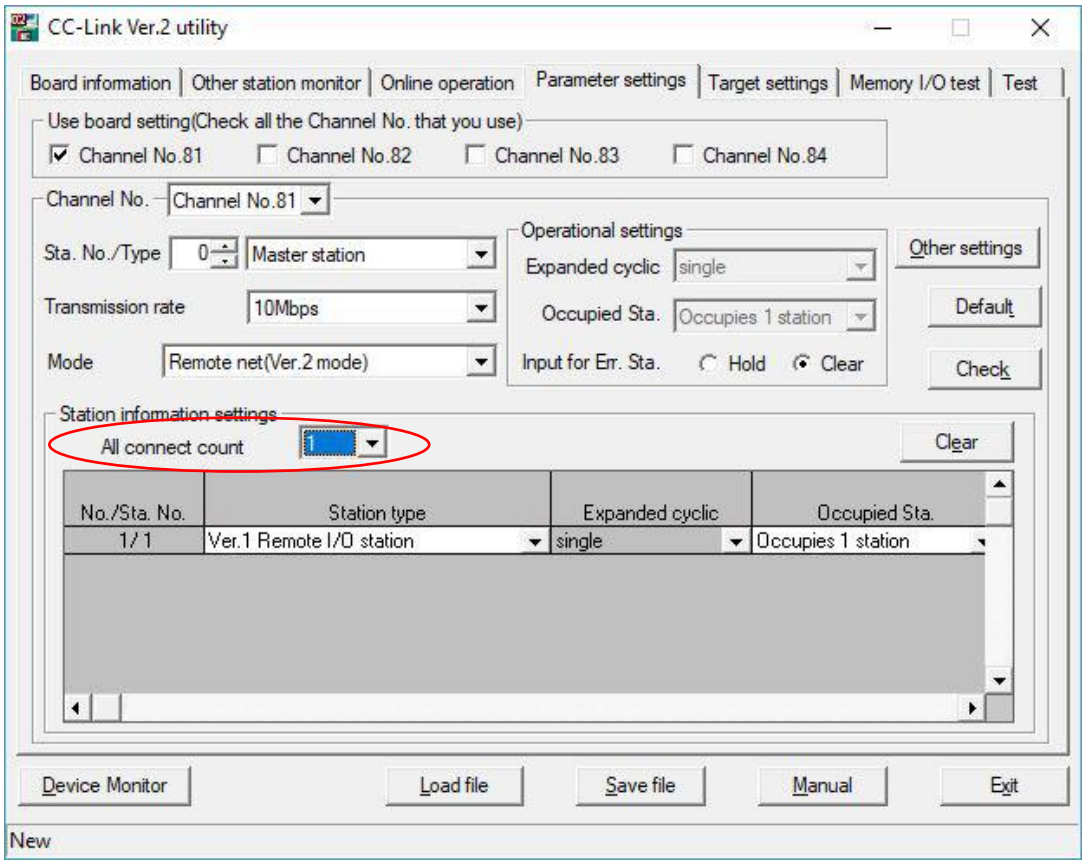


Put the **Sta. No** to 0 in order to inform the MELSEC board it will be the master (and not a slave).
Make the **Transmission rate** correspond to the baud rate of your slaves.
Put the **Mode** to “Remote net(Ver.2 mode)”.
Make the **Input for Err. Sta.** correspond to the output mode of your slaves.

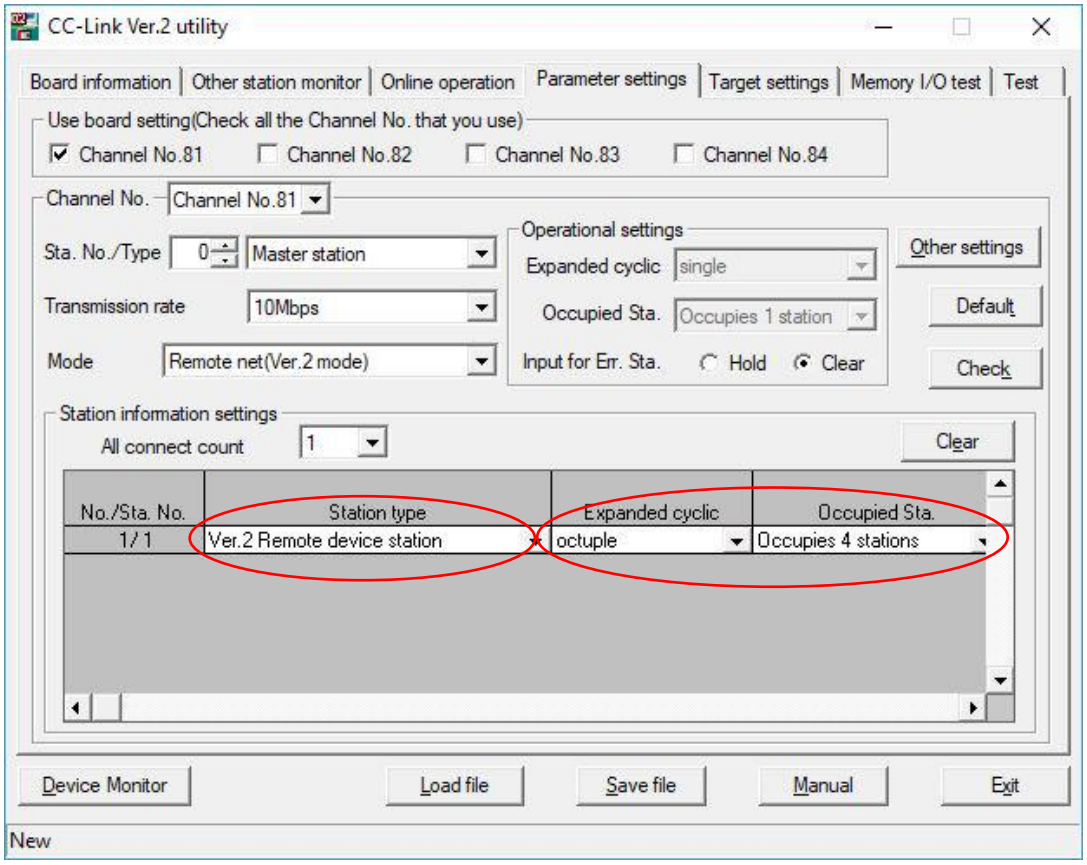


DESCRIBE THE SLAVES TO THE MASTER

Indicate the number of slaves connected on your CC-Link network in **All connect count.**

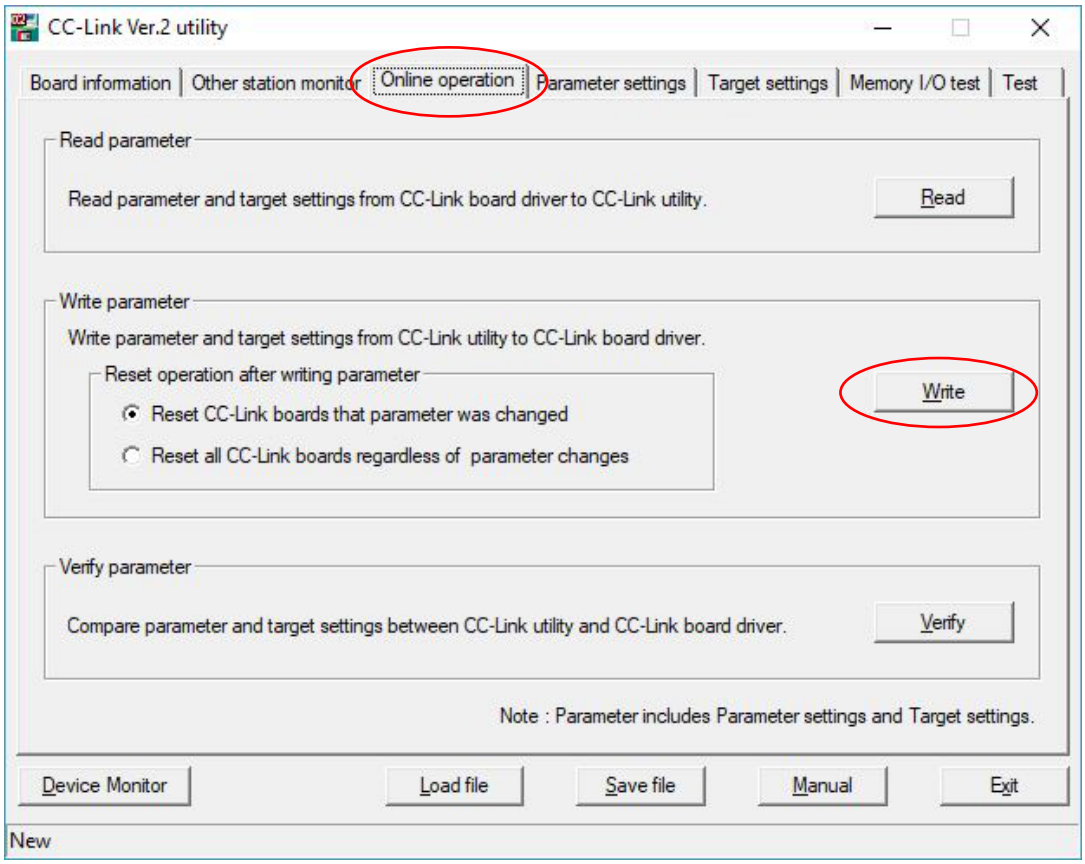


Put the **Station type** of your slaves to “Ver.2 Remote device station”.
Make the **Expanded cyclic** and the number of **Occupied Sta.** correspond to the configuration mode of your slaves.



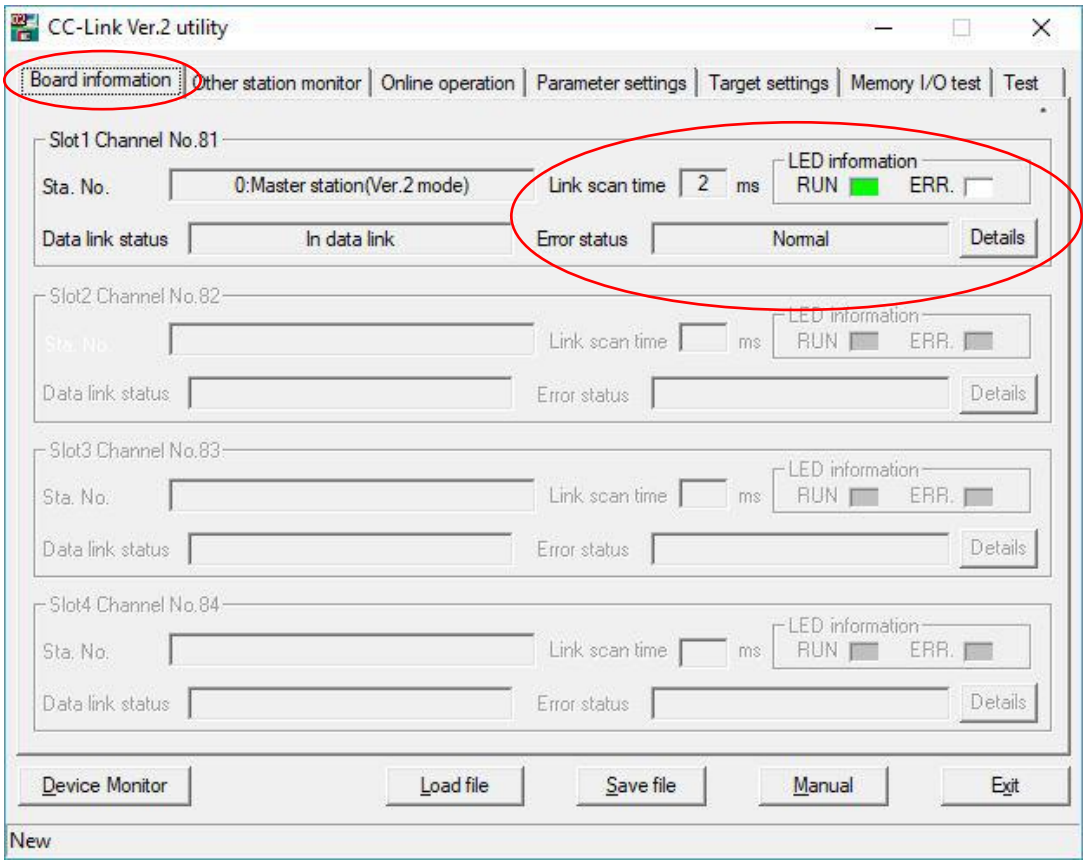
DOWNLOAD THE CONFIGURATION TO THE MASTER

Go into **Online operation** tab and clock on **Write** in order to write all the configuration onto the board.



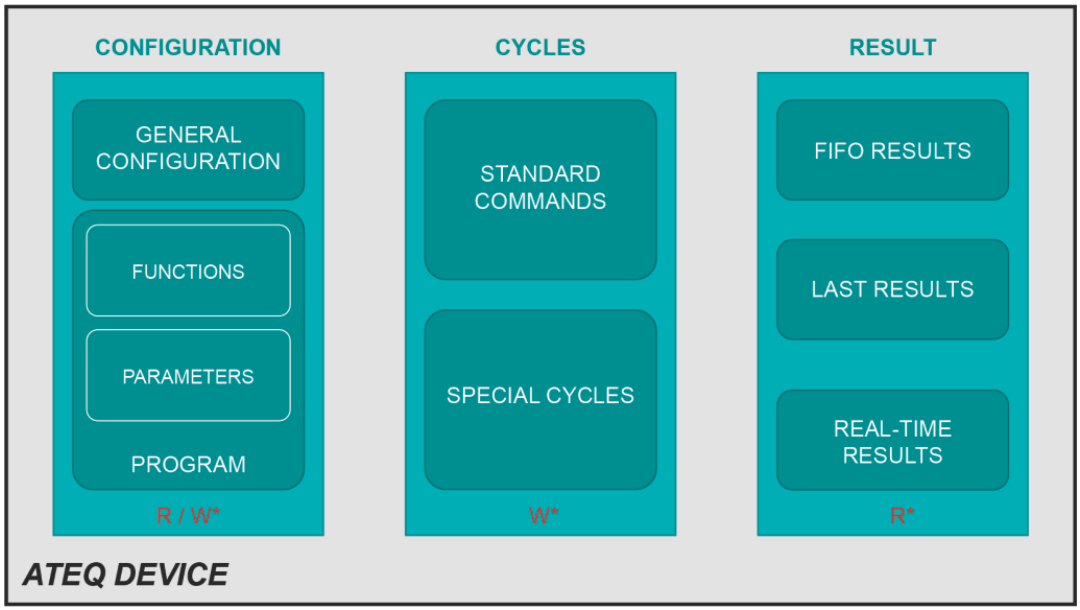
VERIFICATION

Go into the **Board information** tab.
If all your slaves are connected to the master and if the configuration is good, the **RUN led** will be on and the **Error status** will be on “Normal”.
Otherwise the **ERR. led** will be on and the **Error status** will indicate “Error” (you can have details on this error by clicking on the button **Details**)



Functional description of an ATEQ device

INTRODUCTION



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

Write table

Command data

MASTER → ATEQ				
Offset (byte decimal)	Offset (word hexa)	Command / Status	Description	Type
0	00(h)	Commands	Start / Reset / Read Result / Read or write parameters	WORD
2	01(h)	*RESERVED*		
4	02(h)	*RESERVED*		
6	03(h)	Program number to select	Program to run / edit	WORD
8	04(h)	Special cycle code to run	Auto-zero	WORD
10 ... 32	05(h) ... 0F(h)	*RESERVED*		

Exchange data (depends on mode)

MASTER → ATEQ				
Mode	Offset (byte decimal)	Offset (word hexa)	Command / Status	Type
1	-	-	-	-
2	-	-	-	-
3	32	10(h)	General bits functions – 8 words	ARRAY
	48	18(h)	Normal bits functions – 10 words	ARRAY
	68	22(h)	10 parameters	STRUCTURE
	132	42(h)	Program name	ARRAY
4	40	14(h)	General bits functions – 10 words	ARRAY
	60	1E(h)	Normal bits functions – 10 words	ARRAY
	80	28(h)	21 parameters	STRUCTURE
	208	68(h)	Program name	ARRAY
5	72	24(h)	General bits functions – 10 words	ARRAY
	92	2E(h)	Normal bits functions – 10 words	ARRAY
	112	38(h)	21 parameters	STRUCTURE
	240	78(h)	Program name	ARRAY

Read table

Data available cyclically

MASTER ← ATEQ				
Offset (byte decimal)	Offset (word hexa)	Command echo, Status/Relay/Step	Description	Type
0	00(h)	Command echo	Echo of the current command	WORD
2	01(h)	Command error code	Error code of the current command	WORD
4	02(h)	Refresh counter		WORD
6	03(h)	Number of program	Current running program	WORD
8	04(h)	Number of FIFO results	Result available in FIFO	WORD
10	05(h)	Test type	Current running test type	WORD
12	06(h)	Relays status	Current relay status	WORD
14	07(h)	Step	Current cycle step	WORD
Real time measurements				
16	08(h)	Pressure value (32 bits integer)		LONG
20	0A(h)	Pressure unit (32 bits integer)		LONG
24	0C(h)	Flow value (32 bits integer)		LONG
28	0E(h)	Flow unit (32 bits integer)		LONG

Exchange data (depends on mode)

MASTER ← ATEQ				
Mode	Offset (byte decimal)	Offset (word hexa)	Command / Status	Type
1	-	-	-	-
2	32	10(h)	FIFO / Last results	STRUCTURE
3	32	10(h)	General bits functions – 8 words	ARRAY
	48	18(h)	Normal bits functions – 10 words	ARRAY
	68	22(h)	10 parameters	STRUCTURE
	132	42(h)	Program name	ARRAY
	144	48(h)	FIFO / Last results	STRUCTURE
4	40	14(h)	General bits functions – 10 words	ARRAY
	60	1E(h)	Normal bits functions – 10 words	ARRAY
	80	28(h)	21 parameters	STRUCTURE
	208	68(h)	Program name	ARRAY
	224	70(h)	FIFO / Last results	STRUCTURE
5	72	24(h)	General bits functions – 10 words	ARRAY
	92	2E(h)	Normal bits functions – 10 words	ARRAY
	112	38(h)	21 parameters	STRUCTURE
	240	78(h)	Program name	ARRAY
	336	80(h)	FIFO / Last results (extended)	STRUCTURE

Details mode by mode

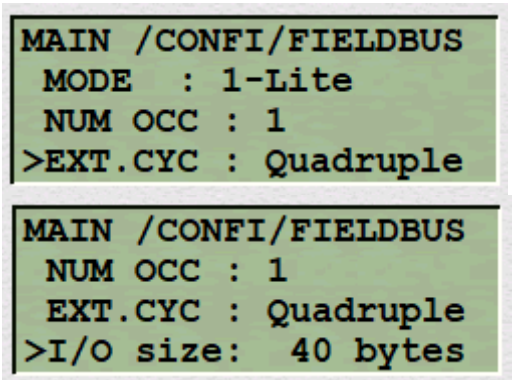
Mode 1.LIGHT: 1 occupied station, Quadruple (40 byte)

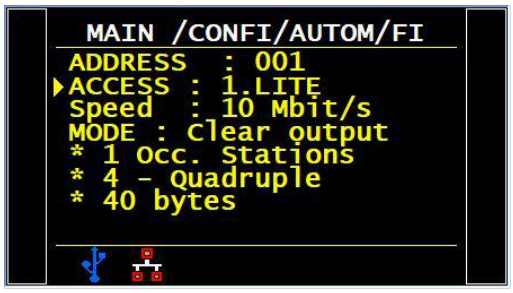
For the digital input/output and the real time measurements.

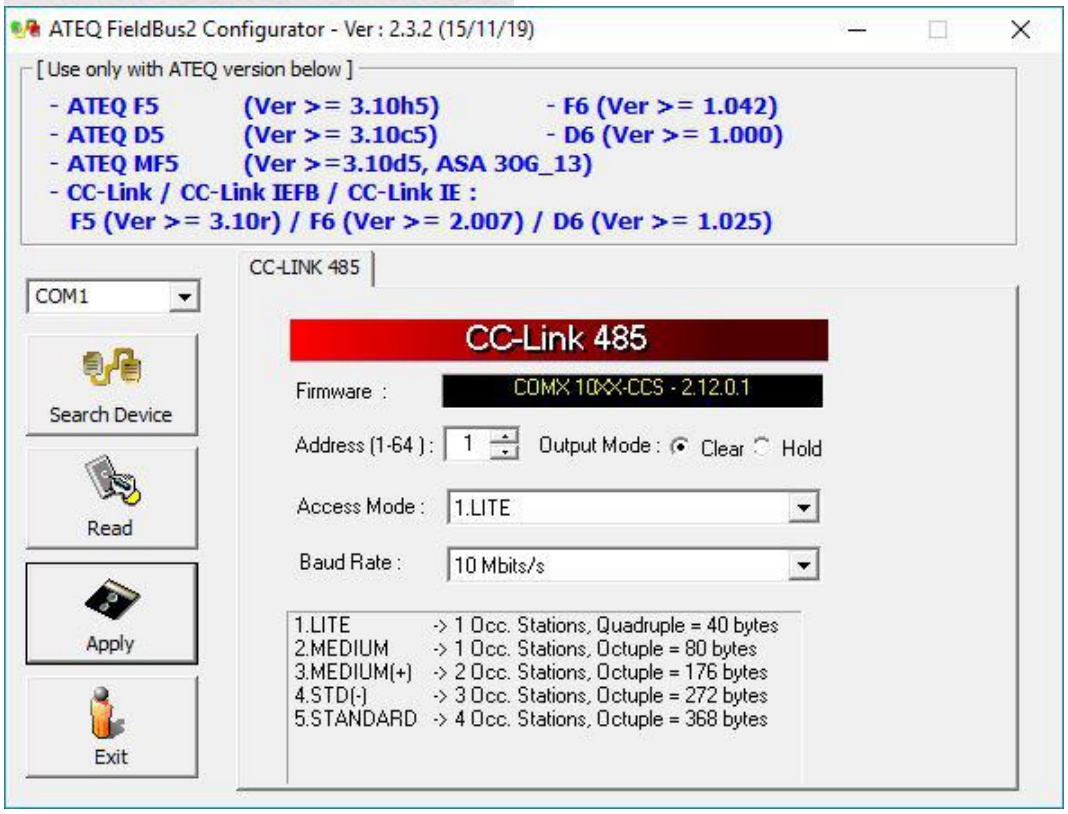
- RX/RY000-03F(h) = 4 words
- WRr/WRw00-0F(h) = 16 words

READ							
Echo / Error	Relay status	Real time measurements	Results	Extended menu bits	Function bits	Parameters	Program name
Start, Reset, Autozero, Select Prog	YES	YES	NO	NO	NO	NO	NO

WRITE				
Commands	Extended menu bits	Function bits	Parameters	Program name
Start, Reset, Autozero, Select Prog	NO	NO	NO	NO







RX/RY = 4 words – Commands						
Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
0	2	00(h)	**Command echo:**		**Command:**	
			X000(h)	Echo reset	Y000(h)	Reset (stop current cycle)
			X001(h)	Echo start	Y001(h)	Start (starting a test cycle)
			X002(h)	Echo start special cycle	Y002(h)	Start special cycle
			X003(h)	Echo program selection	Y003(h)	Program selection
			X004(h) … X00F(h)	*RESERVED*	Y004(h) … Y00F(h)	*RESERVED*
2	2	01(h)	**Command error code:**			
			X010(h)	Error reset	Y010(h) … Y01F(h)	*RESERVED*
			X011(h)	Error start		
			X012(h)	Error start special cycle		
			X013(h)	Error program selection		
			X014(h) … X01F(h)	*RESERVED*		
4	2	02(h)	X020(h) … X02F(h)	Refresh counter	Y020(h) … Y02F(h)	*RESERVED*
6	2	03(h)	X030(h) … X03F(h)	Number of current program	Y030(h) … Y03F(h)	Number of program to be selected

WRr/WRw = 16 words – Status & Real time measurements						
Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
8	2	04(h)	WRr00(h)	Number of FIFO results	WRw00(h)	Special cycle code
10	2	05(h)	WRr01(h)	Test type	WRw01(h)	*RESERVED*
12	2	06(h)	WRr02(h)	**Relay status:**	WRw02(h)	*RESERVED*
				Bit 0 = 1 > Pass part (OK)		
				Bit 1 = 1 > Fail test part (NOK)		
				Bit 2 = 1 > Fail reference part (NOK)		
				Bit 3 = 1 > Alarm		
				Bit 4 = 1 > Pressure error		
				Bit 5 = 1 > Cycle end (system ready)		
				Bit 6 = 1 > Part recoverable		
				Bit 7 = 1 > Calibration error		
				Bit 8 = 1 > *not used*		
				Bit 9 = 1 > ATR fault		
				Bit 10 to 15 > *not used, always at 0*		
14	2	07(h)	WRr03(h)	Step	WRw03(h)	*RESERVED*
Real time measurements						
16	4	08(h)	**Pressure value (32 bits integer)** Example: Pressure value = 207 On network : 98 28 03 00 00032898(H) → 207000(d)/1000 → 207			
			WRr04(h)	Low word (LSW)	WRw04(h)	*RESERVED*
			WRr05(h)	High word (MSW)	WRw05(h)	*RESERVED*
20	4	0A(h)	**Pressure unit (32 bits integer)** Example: Pressure unit Pa = 6 On network: 70 17 00 00 00001770(H) → 6000(d)/1000 → 6			
			WRr06(h)	Low word (LSW)	WRw06(h)	*RESERVED*
			WRr07(h)	High word (MSW)	WRw07(h)	*RESERVED*
24	4	0C(h)	**Flow value (32 bits integer)** Example: Flow value = -0.108 On network: 94 FF FF FF FFFFFF94(H) → -108(d)/1000 → -0.108			
			WRr08(h)	Low word (LSW)	WRw08(h)	*RESERVED*
			WRr09(h)	High word (MSW)	WRw09(h)	*RESERVED*
28	4	0E(h)	**Flow unit (32 bits integer)** Example: Flow Unit Pa/s = 8 On network: 40 1F 00 00 00001F40(H) → 8000(d)/1000 → 8			
			WRr0A(h)	Low word (LSW)	WRw0A(h)	*RESERVED*
			WRr0B(h)	High word (MSW)	WRw0B(h)	*RESERVED*
32	6	10(h)	WRr0C(h) ... WRr0F(h)	*RESERVED*	WRw0C(h) ... WRw0F(h)	*RESERVED*

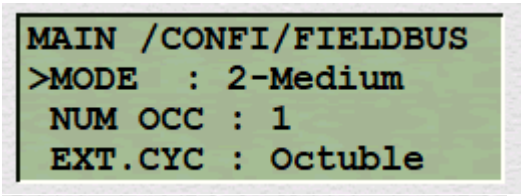
Mode 2.MEDIUM: 1 occupied station, Octuple (80 bytes)

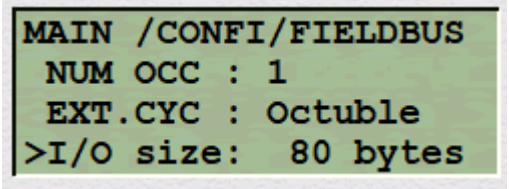
For the input/output, the real time measurements and cycle results

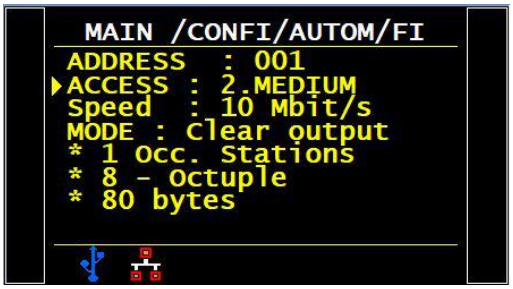
- RX/RY000-07F(h) = 8 words
- WRr/WRw00-1F(h) = 32 words

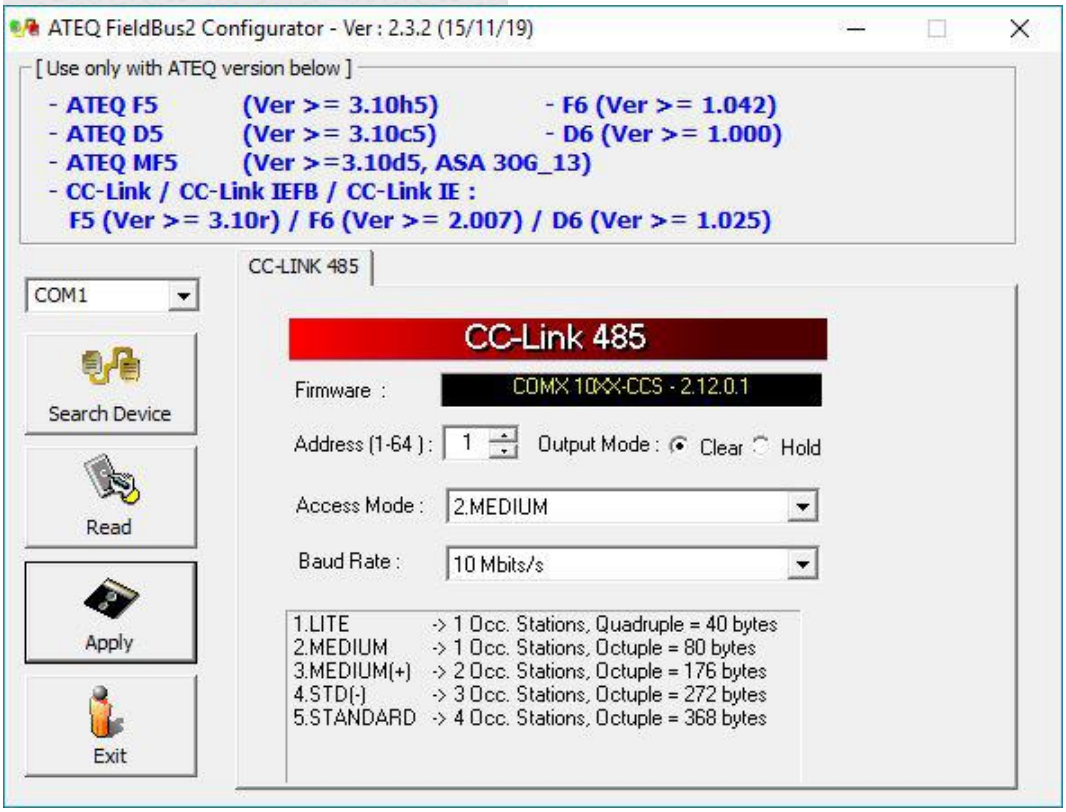
READ							
Echo / Error	Relay status	Real time measurements	Results	Extended menu bits	Function bits	Parameters	Program name
Start, Reset, Autozero, Select Prog, FIFO & Last results	YES	YES	Standard	NO	NO	NO	NO

WRITE				
Commands	Extended menu bits	Function bits	Parameters	Program name
Start, Reset, Autozero, Select Prog, FIFO & Last results	NO	NO	NO	NO









RX/RY = 8 words – Commands & Status						
Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
0	2	00(h)	**Command echo:**		**Command:**	
			X000(h)	Echo reset	Y000(h)	Reset (stop current cycle)
			X001(h)	Echo start	Y001(h)	Start (starting a test cycle)
			X002(h)	Echo start special cycle	Y002(h)	Start special cycle
			X003(h)	Echo program selection	Y003(h)	Program selection
			X004(h)	Echo read FIFO results	Y004(h)	Read FIFO results
			X005(h)	*RESERVED*	Y005(h)	*RESERVED*
			X006(h)	*RESERVED*	Y006(h)	*RESERVED*
			X007(h)	Echo reset FIFO results	Y007(h)	Reset FIFO results
			X008(h) ... X00E(h)	*RESERVED*	Y008(h) ... Y00E(h)	*RESERVED*
			X00F(h)	Echo read Last results	Y00F(h)	Read Last results
2	2	01(h)	**Command error code:**			
			X010(h)	Error reset	Y010(h) ... Y01F(h)	*RESERVED*
			X011(h)	Error start		
			X012(h)	Error start special cycle		
			X013(h)	Error program selection		
			X014(h)	Error read FIFO results		
			X015(h)	*RESERVED*		
			X016(h)	*RESERVED*		
			X017(h)	Error reset FIFO results		
			X018(h) ... X01E(h)	*RESERVED*		
			X01F(h)	Error read Last results		
4	2	02(h)	X020(h) ... X02F(h)	Refresh counter	Y020(h) ... Y02F(h)	*RESERVED*
6	2	03(h)	X030(h) ... X03F(h)	Number of current program	Y030(h) ... Y03F(h)	Number of program to be selected
8	2	04(h)	X040(h) ... X04F(h)	Number of FIFO results	Y040(h) ... Y04F(h)	Special cycle code
10	2	05(h)	X050(h) ... X05F(h)	Test type	Y050(h) ... Y05F(h)	*RESERVED*

Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
12	2	06(h)	**Relay status:**			
			X060(h)	Pass part (OK)	Y060(h)	
			X061(h)	Fail test part (NOK)		
			X062(h)	Fail reference part (NOK)		
			X063(h)	Alarm		
			X064(h)	Pressure error		
			X065(h)	Cycle end (system ready)		
			X066(h)	Part recoverable	…	*RESERVED*
			X067(h)	Calibration error		
			X068(h)	*Not used*		
			X069(h)	ATR fault		
			X06A(h) … X06F(h)	*Not used, always at 0*	Y06F(h)	
14	2	07(h)	X070(h) … X07F(h)	Step	Y070(h) … Y07F(h)	*RESERVED*

WRr/WRw = 32 words – Real time measurements & FIFO / Last results						
Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
16	4	08(h)	**Pressure value (32 bits integer)** Example: Pressure value = 207 On network : 98 28 03 00 00032898(H) → 207000(d)/1000 → 207			
			WRr00(h)	Low word (LSW)	WRw00(h)	*RESERVED*
			WRr01(h)	High word (MSW)	WRw01(h)	*RESERVED*
20	4	0A(h)	**Pressure unit (32 bits integer)** Example: Pressure unit Pa = 6 On network: 70 17 00 00 00001770(H) → 6000(d)/1000 → 6			
			WRr02(h)	Low word (LSW)	WRw02(h)	*RESERVED*
			WRr03(h)	High word (MSW)	WRw03(h)	*RESERVED*
24	4	0C(h)	**Flow value (32 bits integer)** Example: Flow value = -0.108 On network: 94 FF FF FF FFFFFF94(H) → -108(d)/1000 → -0.108			
			WRr04(h)	Low word (LSW)	WRw04(h)	*RESERVED*
			WRr05(h)	High word (MSW)	WRw05(h)	*RESERVED*
28	4	0E(h)	**Flow unit (32 bits integer)** Example: Flow Unit Pa/s = 8 On network: 40 1F 00 00 00001F40(H) → 8000(d)/1000 → 8			
			WRr06(h)	Low word (LSW)	WRw06(h)	*RESERVED*
			WRr07(h)	High word (MSW)	WRw07(h)	*RESERVED*
FIFO / Last results						
32	36	10(h)	WRr08(h) … WRr19(h)	FIFO / Last results	WRw08(h) … WRw19(h)	*RESERVED*
68	6	22(h)	WRr1A(h) … WRr1F(h)	*RESERVED*	WRw1A(h) … WRw1F(h)	*RESERVED*

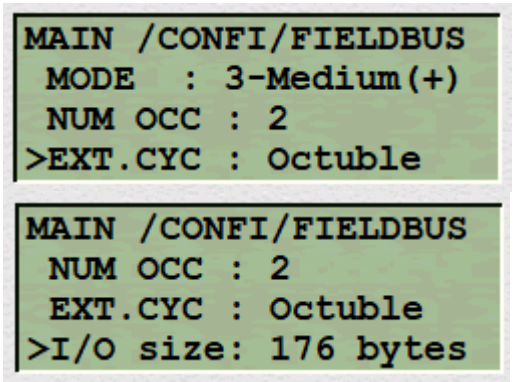
Mode 3.MEDIUM(+): 2 occupied stations, Octuple (176 bytes)

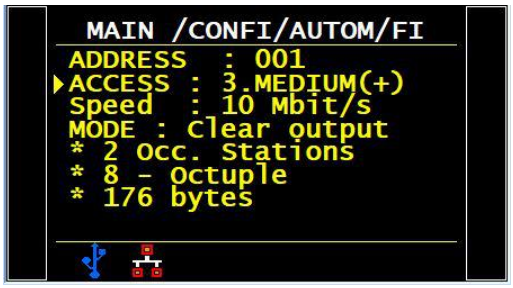
For the input/output, the real time measurements, 10 parameters managements, functions and extended menu bits and cycle result.

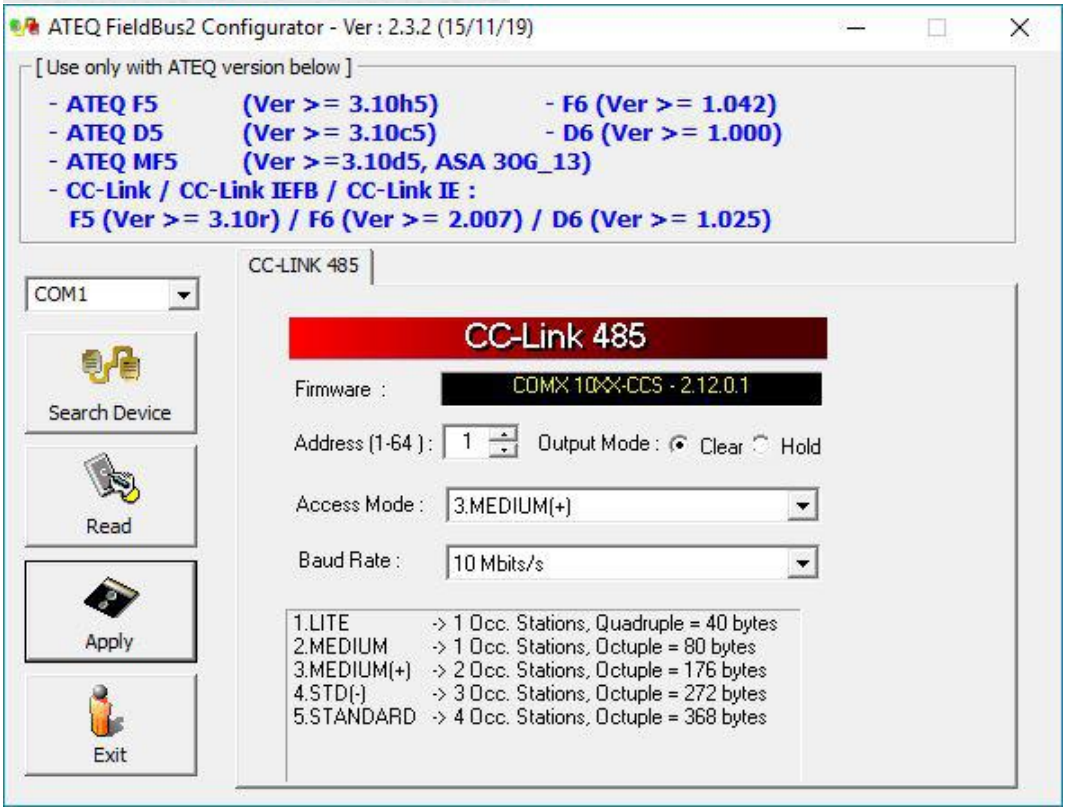
- RX/RY000-17F(h) = 24 words
- WRr/WRw00-3F(h) = 64 words

READ							
Echo / Error	Relay status	Real time measurements	Results	Extended menu bits	Function bits	Parameters	Program name
ALL	YES	YES	Standard	YES (8 words)	YES (10 words)	YES (10 parameters)	YES

WRITE				
Commands	Extended menu bits	Function bits	Parameters	Program name
ALL	YES (8 words)	YES (10 words)	YES (10 parameters)	YES







RX/RY = 24 words – Commands, Status, Real time measurements & Extended menu bits handling						
Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
0	2	00(h)	**Command echo:**		**Command:**	
			X000(h)	Echo reset	Y000(h)	Reset (stop current cycle)
			X001(h)	Echo start	Y001(h)	Start (starting a test cycle)
			X002(h)	Echo start special cycle	Y002(h)	Start special cycle
			X003(h)	Echo program selection	Y003(h)	Program selection
			X004(h)	Echo read FIFO results	Y004(h)	Read FIFO results
			X005(h)	Echo read parameters	Y005(h)	Read parameters
			X006(h)	Echo write parameters	Y006(h)	Write parameters
			X007(h)	Echo reset FIFO results	Y007(h)	Reset FIFO results
			X008(h)	*RESERVED*	Y008(h)	*RESERVED*
			X009(h)	Echo read ext. menu bits	Y009(h)	Read extended menu bits
			X00A(h)	Echo read functions bits	Y00A(h)	Read functions bits
			X00B(h)	Echo write ext. menu bits	Y00B(h)	Write extended menu bits
			X00C(h)	Echo write functions bits	Y00C(h)	Write functions bits
			X00D(h)	Echo read program name	Y00D(h)	Read program name
			X00E(h)	Echo write program name	Y00E(h)	Write program name
			X00F(h)	Echo read Last results	Y00F(h)	Read Last results
2	2	01(h)	**Command error code:**			
			X010(h)	Error reset	Y010(h)	*RESERVED*
			X011(h)	Error start		
			X012(h)	Error start special cycle		
			X013(h)	Error program selection		
			X014(h)	Error read FIFO results		
			X015(h)	Error read parameters		
			X016(h)	Error write parameters		
			X017(h)	Error reset FIFO results		
			X018(h)	*RESERVED*	…	
			X019(h)	Error read ext. menu bits		
			X01A(h)	Error read functions bits		
			X01B(h)	Error write ext. menu bits		
			X01C(h)	Error write functions bits		
			X01D(h)	Error read program name		
			X01E(h)	Error write program name		
			X01F(h)	Error read Last results	Y01F(h)	
4	2	02(h)	X020(h) … X02F(h)	Refresh counter	Y020(h) … Y02F(h)	*RESERVED*
6	2	03(h)	X030(h) … X03F(h)	Number of current program	Y030(h) … Y03F(h)	Number of program to be selected
8	2	04(h)	X040(h) … X04F(h)	Number of FIFO results	Y040(h) … Y04F(h)	Special cycle code
10	2	05(h)	X050(h) … X05F(h)	Test type	Y050(h) … Y05F(h)	*RESERVED*

Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
12	2	06(h)	**Relay status:**			
			X060(h)	Pass part (OK)	Y060(h)	
			X061(h)	Fail test part (NOK)		
			X062(h)	Fail reference part (NOK)		
			X063(h)	Alarm		
			X064(h)	Pressure error		
			X065(h)	Cycle end (system ready)		
			X066(h)	Part recoverable	…	*RESERVED*
			X067(h)	Calibration error		
			X068(h)	*Not used*		
			X069(h)	ATR fault		
			X06A(h) … X06F(h)	*Not used, always at 0*	Y06F(h)	
14	2	07(h)	X070(h) … X07F(h)	Step	Y070(h) … Y07F(h)	*RESERVED*

Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
Real time measurements						
16	4	08(h)	**Pressure value (32 bits integer)** Example: Pressure value = 207 On network : 98 28 03 00 00032898(H) → 207000(d)/1000 → 207			
			X080(h) ... X08F(h)	Low word (LSW)	Y080(h) ... Y08F(h)	*RESERVED*
			X090(h) ... X09F(h)	High word (MSW)	Y090(h) ... Y09F(h)	RESERVED
20	4	0A(h)	**Pressure unit (32 bits integer)** Example: Pressure unit Pa = 6 On network: 70 17 00 00 00001770(H) → 6000(d)/1000 → 6			
			X0A0(h) ... X0AF(h)	Low word (LSW)	Y0A0(h) ... Y0AF(h)	*RESERVED*
			X0B0(h) ... X0BF(h)	High word (MSW)	Y0B0(h) ... Y0BF(h)	*RESERVED*
24	4	0C(h)	**Flow value (32 bits integer)** Example: Flow value = -0.108 On network: 94 FF FF FF FFFFFF94(H) → -108(d)/1000 → -0.108			
			X0C0(h) ... X0CF(h)	Low word (LSW)	Y0C0(h) ... Y0CF(h)	*RESERVED*
			X0D0(h) ... X0DF(h)	High word (MSW)	Y0D0(h) ... Y0DF(h)	*RESERVED*
28	4	0E(h)	**Flow unit (32 bits integer)** Example: Flow Unit Pa/s = 8 On network: 40 1F 00 00 00001F40(H) → 8000(d)/1000 → 8			
			X0E0(h) ... X0EF(h)	Low word (LSW)	Y0E0(h) ... Y0EF(h)	*RESERVED*
			X0F0(h) ... X0FF(h)	High word (MSW)	Y0F0(h) ... Y0FF(h)	*RESERVED*

Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
Extended menu bits						
32	16	10(h)	**Extended menu bits reading:**		**Extended menu bits writing:**	
			X100(h) … X10F(h)	Extended menu bits - word 0	Y100(h) … Y10F(h)	Extended menu bits - word 0
			X110(h) … X11F(h)	Extended menu bits - word 1	Y110(h) … Y11F(h)	Extended menu bits - word 1
			X120(h) … X12F(h)	Extended menu bits - word 2	Y120(h) … Y12F(h)	Extended menu bits - word 2
			X130(h) … X13F(h)	Extended menu bits - word 3	Y130(h) … Y13F(h)	Extended menu bits - word 3
			X140(h) … X14F(h)	Extended menu bits - word 4	Y140(h) … Y14F(h)	Extended menu bits - word 4
			X150(h) … X15F(h)	Extended menu bits - word 5	Y150(h) … Y15F(h)	Extended menu bits - word 5
			X160(h) … X16F(h)	Extended menu bits - word 6	Y160(h) … Y16F(h)	Extended menu bits - word 6
			X170(h) … X17F(h)	Extended menu bits - word 7	Y170(h) … Y17F(h)	Extended menu bits - word 7

WRr/WRw = 64 words – Function bits handling, Parameters handling, Program name & FIFO / Last results						
Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
48	20	18(h)	**Function bits reading:**		**Function bits writing:**	
			WRr00(h)	Function bits - word 0	WRw00(h)	Function bits - word 0
			WRr01(h)	Function bits - word 1	WRw01(h)	Function bits - word 1
			WRr02(h)	Function bits - word 2	WRw02(h)	Function bits - word 2
			WRr03(h)	Function bits - word 3	WRw03(h)	Function bits - word 3
			WRr04(h)	Function bits - word 4	WRw04(h)	Function bits - word 4
			WRr05(h)	Function bits - word 5	WRw05(h)	Function bits - word 5
			WRr06(h)	Function bits - word 6	WRw06(h)	Function bits - word 6
			WRr07(h)	Function bits - word 7	WRw07(h)	Function bits - word 7
			WRr08(h)	Function bits - word 8	WRw08(h)	Function bits - word 8
			WRr09(h)	Function bits - word 9	WRw09(h)	Function bits - word 9
68	64	22(h)	WRr0A(h) … WRr29(h)	Parameters reading	WRw0A(h) … WRw29(h)	Parameters writing
132	12	42(h)	WRr2A(h) … WRr2F(h)	Program name reading	WRw2A(h) … WRw2F(h)	Program name writing
144	32	48(h)	WRr30(h) … WRr3F(h)	FIFO / Last results	WRw30(h) … WRw3F(h)	*RESERVED*

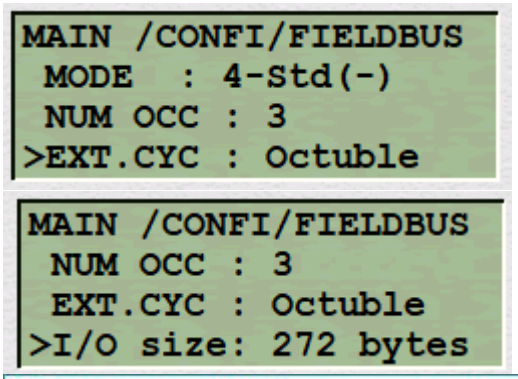
Mode 4.STD(-): 3 occupied stations, Octuple (272 bytes)

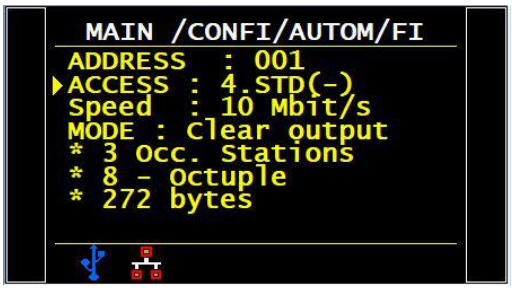
For the input/output, the real time measurements, 21 parameters managements, functions and extended menu bits and cycle result.

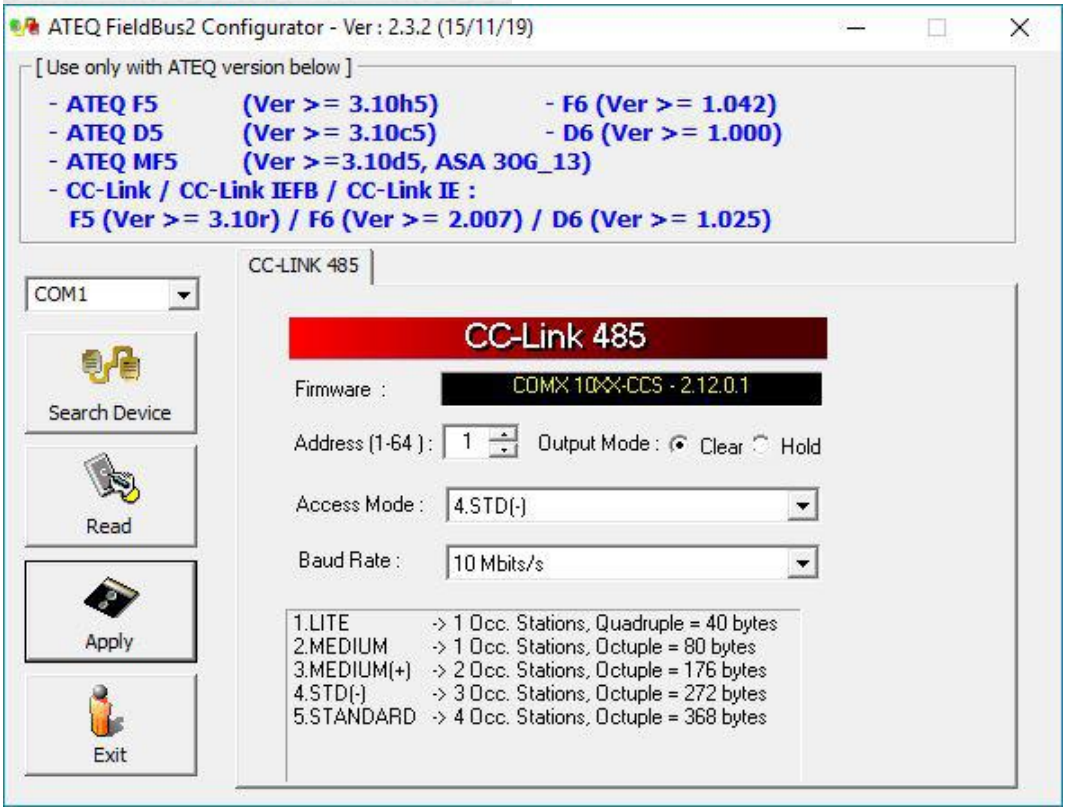
- RX/RY000-27F(h) = 40 words
- WRr/WRw00-5F(h) = 96 words

READ							
Echo / Error	Relay status	Real time measurements	Results	Extended menu bits	Function bits	Parameters	Program name
ALL	YES	YES	Standard	YES (10 words)	YES (10 words)	YES (21 parameters)	YES

WRITE				
Commands	Extended menu bits	Function bits	Parameters	Program name
ALL	YES (10 words)	YES (10 words)	YES (21 parameters)	YES







RX/RY = 40 words – Commands, Status, Real time measurements, Extended menu bits handling & Function bits handling						
Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
0	2	00(h)	**Command echo:**		**Command:**	
			X000(h)	Echo reset	Y000(h)	Reset (stop current cycle)
			X001(h)	Echo start	Y001(h)	Start (starting a test cycle)
			X002(h)	Echo start special cycle	Y002(h)	Start special cycle
			X003(h)	Echo program selection	Y003(h)	Program selection
			X004(h)	Echo read FIFO results	Y004(h)	Read FIFO results
			X005(h)	Echo read parameters	Y005(h)	Read parameters
			X006(h)	Echo write parameters	Y006(h)	Write parameters
			X007(h)	Echo reset FIFO results	Y007(h)	Reset FIFO results
			X008(h)	*RESERVED*	Y008(h)	*RESERVED*
			X009(h)	Echo read ext. menu bits	Y009(h)	Read extended menu bits
			X00A(h)	Echo read functions bits	Y00A(h)	Read functions bits
			X00B(h)	Echo write ext. menu bits	Y00B(h)	Write extended menu bits
			X00C(h)	Echo write functions bits	Y00C(h)	Write functions bits
			X00D(h)	Echo read program name	Y00D(h)	Read program name
			X00E(h)	Echo write program name	Y00E(h)	Write program name
			X00F(h)	Echo read Last results	Y00F(h)	Read Last results
2	2	01(h)	**Command error code:**			
			X010(h)	Error reset	Y010(h)	*RESERVED*
			X011(h)	Error start		
			X012(h)	Error start special cycle		
			X013(h)	Error program selection		
			X014(h)	Error read FIFO results		
			X015(h)	Error read parameters		
			X016(h)	Error write parameters		
			X017(h)	Error reset FIFO results		
			X018(h)	*RESERVED*	…	
			X019(h)	Error read ext. menu bits		
			X01A(h)	Error read functions bits		
			X01B(h)	Error write ext. menu bits		
			X01C(h)	Error write functions bits		
			X01D(h)	Error read program name		
			X01E(h)	Error write program name		
			X01F(h)	Error read Last results	Y01F(h)	
4	2	02(h)	X020(h) … X02F(h)	Refresh counter	Y020(h) … Y02F(h)	*RESERVED*
6	2	03(h)	X030(h) … X03F(h)	Number of current program	Y030(h) … Y03F(h)	Number of program to be selected
8	2	04(h)	X040(h) … X04F(h)	Number of FIFO results	Y040(h) … Y04F(h)	Special cycle code
10	2	05(h)	X050(h) … X05F(h)	Test type	Y050(h) … Y05F(h)	*RESERVED*

Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
12	2	06(h)	**Relay status:**			
			X060(h)	Pass part (OK)	Y060(h)	
			X061(h)	Fail test part (NOK)		
			X062(h)	Fail reference part (NOK)		
			X063(h)	Alarm		
			X064(h)	Pressure error		
			X065(h)	Cycle end (system ready)		
			X066(h)	Part recoverable	…	*RESERVED*
			X067(h)	Calibration error		
			X068(h)	*Not used*		
			X069(h)	ATR fault		
			X06A(h) … X06F(h)	*Not used, always at 0*	Y06F(h)	
14	2	07(h)	X070(h) … X07F(h)	Step	Y070(h) … Y07F(h)	*RESERVED*

Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
Real time measurements						
16	4	08(h)	**Pressure value (32 bits integer)** Example: Pressure value = 207 On network : 98 28 03 00 00032898(H) → 207000(d)/1000 → 207			
			X080(h) ... X08F(h)	Low word (LSW)	Y080(h) ... Y08F(h)	*RESERVED*
			X090(h) ... X09F(h)	High word (MSW)	Y090(h) ... Y09F(h)	RESERVED
20	4	0A(h)	**Pressure unit (32 bits integer)** Example: Pressure unit Pa = 6 On network: 70 17 00 00 00001770(H) → 6000(d)/1000 → 6			
			X0A0(h) ... X0AF(h)	Low word (LSW)	Y0A0(h) ... Y0AF(h)	*RESERVED*
			X0B0(h) ... X0BF(h)	High word (MSW)	Y0B0(h) ... Y0BF(h)	*RESERVED*
24	4	0C(h)	**Flow value (32 bits integer)** Example: Flow value = -0.108 On network: 94 FF FF FF FFFFFF94(H) → -108(d)/1000 → -0.108			
			X0C0(h) ... X0CF(h)	Low word (LSW)	Y0C0(h) ... Y0CF(h)	*RESERVED*
			X0D0(h) ... X0DF(h)	High word (MSW)	Y0D0(h) ... Y0DF(h)	*RESERVED*
28	4	0E(h)	**Flow unit (32 bits integer)** Example: Flow Unit Pa/s = 8 On network: 40 1F 00 00 00001F40(H) → 8000(d)/1000 → 8			
			X0E0(h) ... X0EF(h)	Low word (LSW)	Y0E0(h) ... Y0EF(h)	*RESERVED*
			X0F0(h) ... X0FF(h)	High word (MSW)	Y0F0(h) ... Y0FF(h)	*RESERVED*
32	8	10(h)	X100(h) ... X13F(h)	*RESERVED*	Y100(h) ... Y13F(h)	*RESERVED*

Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
Extended menu & Functions bits						
40	20	14(h)	**Extended menu bits reading:**		**Extended menu bits writing:**	
			X140(h) … X14F(h)	Extended menu bits - word 0	Y140(h) … Y14F(h)	Extended menu bits - word 0
			X150(h) … X15F(h)	Extended menu bits - word 1	Y150(h) … Y15F(h)	Extended menu bits - word 1
			X160(h) … X16F(h)	Extended menu bits - word 2	Y160(h) … Y16F(h)	Extended menu bits - word 2
			X170(h) … X17F(h)	Extended menu bits - word 3	Y170(h) … Y17F(h)	Extended menu bits - word 3
			X180(h) … X18F(h)	Extended menu bits - word 4	Y180(h) … Y18F(h)	Extended menu bits - word 4
			X190(h) … X19F(h)	Extended menu bits - word 5	Y190(h) … Y19F(h)	Extended menu bits - word 5
			X1A0(h) … X1AF(h)	Extended menu bits - word 6	Y1A0(h) … Y1AF(h)	Extended menu bits - word 6
			X1B0(h) … X1BF(h)	Extended menu bits - word 7	Y1B0(h) … Y1BF(h)	Extended menu bits - word 7
			X1C0(h) … X1CF(h)	Extended menu bits - word 8	Y1C0(h) … Y1CF(h)	Extended menu bits - word 8
			X1D0(h) … X1DF(h)	Extended menu bits - word 9	Y1D0(h) … Y1DF(h)	Extended menu bits - word 9

Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
60	20	1E(h)	**Function bits reading:**		**Function bits writing:**	
			X1E0(h) … X1EF(h)	Function bits - word 0	Y1E0(h) … Y1EF(h)	Function bits - word 0
			X1F0(h) … X1FF(h)	Function bits - word 1	Y1F0(h) … Y1FF(h)	Function bits - word 1
			X200(h) … X20F(h)	Function bits - word 2	Y200(h) … Y20F(h)	Function bits - word 2
			X210(h) … X21F(h)	Function bits - word 3	Y210(h) … Y21F(h)	Function bits - word 3
			X220(h) … X22F(h)	Function bits - word 4	Y220(h) … Y22F(h)	Function bits - word 4
			X230(h) … X23F(h)	Function bits - word 5	Y230(h) … Y23F(h)	Function bits - word 5
			X240(h) … X24F(h)	Function bits - word 6	Y240(h) … Y24F(h)	Function bits - word 6
			X250(h) … X25F(h)	Function bits - word 7	Y250(h) … Y25F(h)	Function bits - word 7
			X260(h) … X26F(h)	Function bits - word 8	Y260(h) … Y26F(h)	Function bits - word 8
			X270(h) … X27F(h)	Function bits - word 9	Y270(h) … Y27F(h)	Function bits - word 9

WRr/WRw = 96 words – Parameters handling, Program name & FIFO / Last results						
Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
80	128	28(h)	WRr00(h) … WRr39(h)	Parameters reading	WRw00(h) … WRw39(h)	Parameters writing
208	16	68(h)	WRr40(h) … WRr47(h)	Program name reading	WRw40(h) … WRw47(h)	Program name writing
224	36	70(h)	WRr48(h) … WRr59(h)	FIFO / Last results	WRw48(h) … WRw59(h)	*RESERVED*
260	12	82(h)	WRr5A(h) … WRr5F(h)	*RESERVED*	WRw5A(h) … WRw5F(h)	*RESERVED*

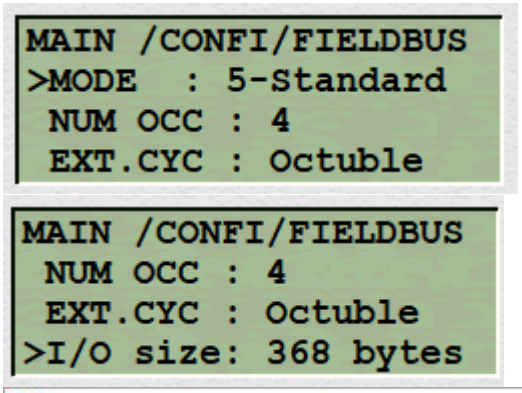
Mode 5.STANDARD: 4 occupied stations, Octuple (368 bytes)

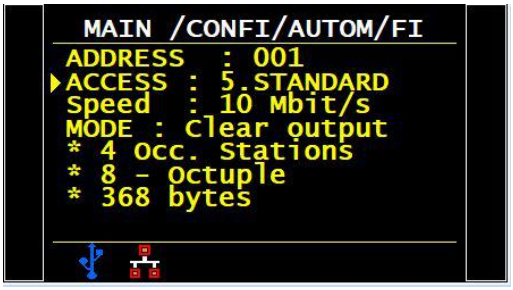
For the input/output, the real time measurements, 21 parameters managements, functions and extended menu bits and extended cycle result.

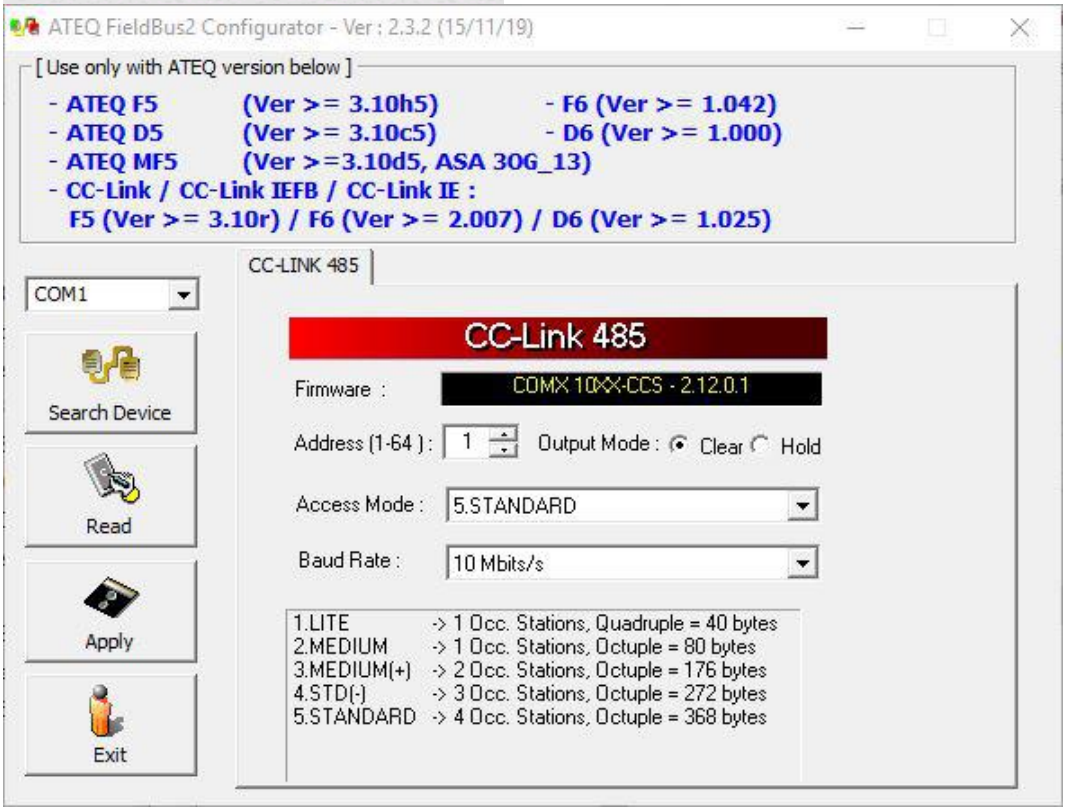
- RX/RY000-37F(h) = 56 words
- WRr/WRw00-7F(h) = 128 words

READ							
Echo / Error	Relay status	Real time measurements	Results	Extended menu bits	Function bits	Parameters	Program name
ALL	YES	YES	Extended	YES (10 words)	YES (10 words)	YES (21 parameters)	YES

WRITE				
Commands	Extended menu bits	Function bits	Parameters	Program name
ALL	YES (10 words)	YES (10 words)	YES (21 parameters)	YES







RX/RY = 40 words – Commands, Status, Real time measurements, Extended menu bits handling & Function bits handling						
Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
0	2	00(h)	**Command echo:**		**Command:**	
			X000(h)	Echo reset	Y000(h)	Reset (stop current cycle)
			X001(h)	Echo start	Y001(h)	Start (starting a test cycle)
			X002(h)	Echo start special cycle	Y002(h)	Start special cycle
			X003(h)	Echo program selection	Y003(h)	Program selection
			X004(h)	Echo read FIFO results	Y004(h)	Read FIFO results
			X005(h)	Echo read parameters	Y005(h)	Read parameters
			X006(h)	Echo write parameters	Y006(h)	Write parameters
			X007(h)	Echo reset FIFO results	Y007(h)	Reset FIFO results
			X008(h)	*RESERVED*	Y008(h)	*RESERVED*
			X009(h)	Echo read ext. menu bits	Y009(h)	Read extended menu bits
			X00A(h)	Echo read functions bits	Y00A(h)	Read functions bits
			X00B(h)	Echo write ext. menu bits	Y00B(h)	Write extended menu bits
			X00C(h)	Echo write functions bits	Y00C(h)	Write functions bits
			X00D(h)	Echo read program name	Y00D(h)	Read program name
			X00E(h)	Echo write program name	Y00E(h)	Write program name
			X00F(h)	Echo read Last results	Y00F(h)	Read Last results
2	2	01(h)	**Command error code:**			
			X010(h)	Error reset	Y010(h)	*RESERVED*
			X011(h)	Error start		
			X012(h)	Error start special cycle		
			X013(h)	Error program selection		
			X014(h)	Error read FIFO results		
			X015(h)	Error read parameters		
			X016(h)	Error write parameters		
			X017(h)	Error reset FIFO results		
			X018(h)	*RESERVED*	…	
			X019(h)	Error read ext. menu bits		
			X01A(h)	Error read functions bits		
			X01B(h)	Error write ext. menu bits		
			X01C(h)	Error write functions bits		
			X01D(h)	Error read program name		
			X01E(h)	Error write program name		
			X01F(h)	Error read Last results	Y01F(h)	
4	2	02(h)	X020(h) … X02F(h)	Refresh counter	Y020(h) … Y02F(h)	*RESERVED*
6	2	03(h)	X030(h) … X03F(h)	Number of current program	Y030(h) … Y03F(h)	Number of program to be selected
8	2	04(h)	X040(h) … X04F(h)	Number of FIFO results	Y040(h) … Y04F(h)	Special cycle code
10	2	05(h)	X050(h) … X05F(h)	Test type	Y050(h) … Y05F(h)	*RESERVED*

Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
12	2	06(h)	**Relay status:**			
			X060(h)	Pass part (OK)	Y060(h)	
			X061(h)	Fail test part (NOK)		
			X062(h)	Fail reference part (NOK)		
			X063(h)	Alarm		
			X064(h)	Pressure error		
			X065(h)	Cycle end (system ready)		
			X066(h)	Part recoverable	…	*RESERVED*
			X067(h)	Calibration error		
			X068(h)	*Not used*		
			X069(h)	ATR fault		
			X06A(h) … X06F(h)	*Not used, always at 0*	Y06F(h)	
14	2	07(h)	X070(h) … X07F(h)	Step	Y070(h) … Y07F(h)	*RESERVED*

Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
Real time measurements						
16	4	08(h)	**Pressure value (32 bits integer)** Example: Pressure value = 207 On network : 98 28 03 00 00032898(H) → 207000(d)/1000 → 207			
			X080(h) ... X08F(h)	Low word (LSW)	Y080(h) ... Y08F(h)	*RESERVED*
			X090(h) ... X09F(h)	High word (MSW)	Y090(h) ... Y09F(h)	RESERVED
20	4	0A(h)	**Pressure unit (32 bits integer)** Example: Pressure unit Pa = 6 On network: 70 17 00 00 00001770(H) → 6000(d)/1000 → 6			
			X0A0(h) ... X0AF(h)	Low word (LSW)	Y0A0(h) ... Y0AF(h)	*RESERVED*
			X0B0(h) ... X0BF(h)	High word (MSW)	Y0B0(h) ... Y0BF(h)	*RESERVED*
24	4	0C(h)	**Flow value (32 bits integer)** Example: Flow value = -0.108 On network: 94 FF FF FF FFFFFF94(H) → -108(d)/1000 → -0.108			
			X0C0(h) ... X0CF(h)	Low word (LSW)	Y0C0(h) ... Y0CF(h)	*RESERVED*
			X0D0(h) ... X0DF(h)	High word (MSW)	Y0D0(h) ... Y0DF(h)	*RESERVED*
28	4	0E(h)	**Flow unit (32 bits integer)** Example: Flow Unit Pa/s = 8 On network: 40 1F 00 00 00001F40(H) → 8000(d)/1000 → 8			
			X0E0(h) ... X0EF(h)	Low word (LSW)	Y0E0(h) ... Y0EF(h)	*RESERVED*
			X0F0(h) ... X0FF(h)	High word (MSW)	Y0F0(h) ... Y0FF(h)	*RESERVED*
32	40	10(h)	X100(h) ... X13F(h)	*RESERVED*	Y100(h) ... Y13F(h)	*RESERVED*

Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
Extended menu & Functions bits						
72	20	24(h)	**Extended menu bits reading:**		**Extended menu bits writing:**	
			X240(h) … X24F(h)	Extended menu bits - word 0	Y240(h) … Y24F(h)	Extended menu bits - word 0
			X250(h) … X25F(h)	Extended menu bits - word 1	Y250(h) … Y25F(h)	Extended menu bits - word 1
			X260(h) … X26F(h)	Extended menu bits - word 2	Y260(h) … Y26F(h)	Extended menu bits - word 2
			X270(h) … X27F(h)	Extended menu bits - word 3	Y270(h) … Y27F(h)	Extended menu bits - word 3
			X280(h) … X28F(h)	Extended menu bits - word 4	Y280(h) … Y28F(h)	Extended menu bits - word 4
			X290(h) … X29F(h)	Extended menu bits - word 5	Y290(h) … Y29F(h)	Extended menu bits - word 5
			X2A0(h) … X2AF(h)	Extended menu bits - word 6	Y2A0(h) … Y2AF(h)	Extended menu bits - word 6
			X2B0(h) … X2BF(h)	Extended menu bits - word 7	Y2B0(h) … Y2BF(h)	Extended menu bits - word 7
			X2C0(h) … X2CF(h)	Extended menu bits - word 8	Y2C0(h) … Y2CF(h)	Extended menu bits - word 8
			X2D0(h) … X2DF(h)	Extended menu bits - word 9	Y2D0(h) … Y2DF(h)	Extended menu bits - word 9

Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
92	20	2E(h)	**Function bits reading:**		**Function bits writing:**	
			X2E0(h) … X2EF(h)	Function bits - word 0	Y2E0(h) … Y2EF(h)	Function bits - word 0
			X2F0(h) … X2FF(h)	Function bits - word 1	Y2F0(h) … Y2FF(h)	Function bits - word 1
			X300(h) … X30F(h)	Function bits - word 2	Y300(h) … Y30F(h)	Function bits - word 2
			X310(h) … X31F(h)	Function bits - word 3	Y310(h) … Y31F(h)	Function bits - word 3
			X320(h) … X32F(h)	Function bits - word 4	Y320(h) … Y32F(h)	Function bits - word 4
			X330(h) … X33F(h)	Function bits - word 5	Y330(h) … Y33F(h)	Function bits - word 5
			X340(h) … X34F(h)	Function bits - word 6	Y340(h) … Y34F(h)	Function bits - word 6
			X350(h) … X35F(h)	Function bits - word 7	Y350(h) … Y35F(h)	Function bits - word 7
			X360(h) … X36F(h)	Function bits - word 8	Y360(h) … Y36F(h)	Function bits - word 8
			X370(h) … X37F(h)	Function bits - word 9	Y370(h) … Y37F(h)	Function bits - word 9

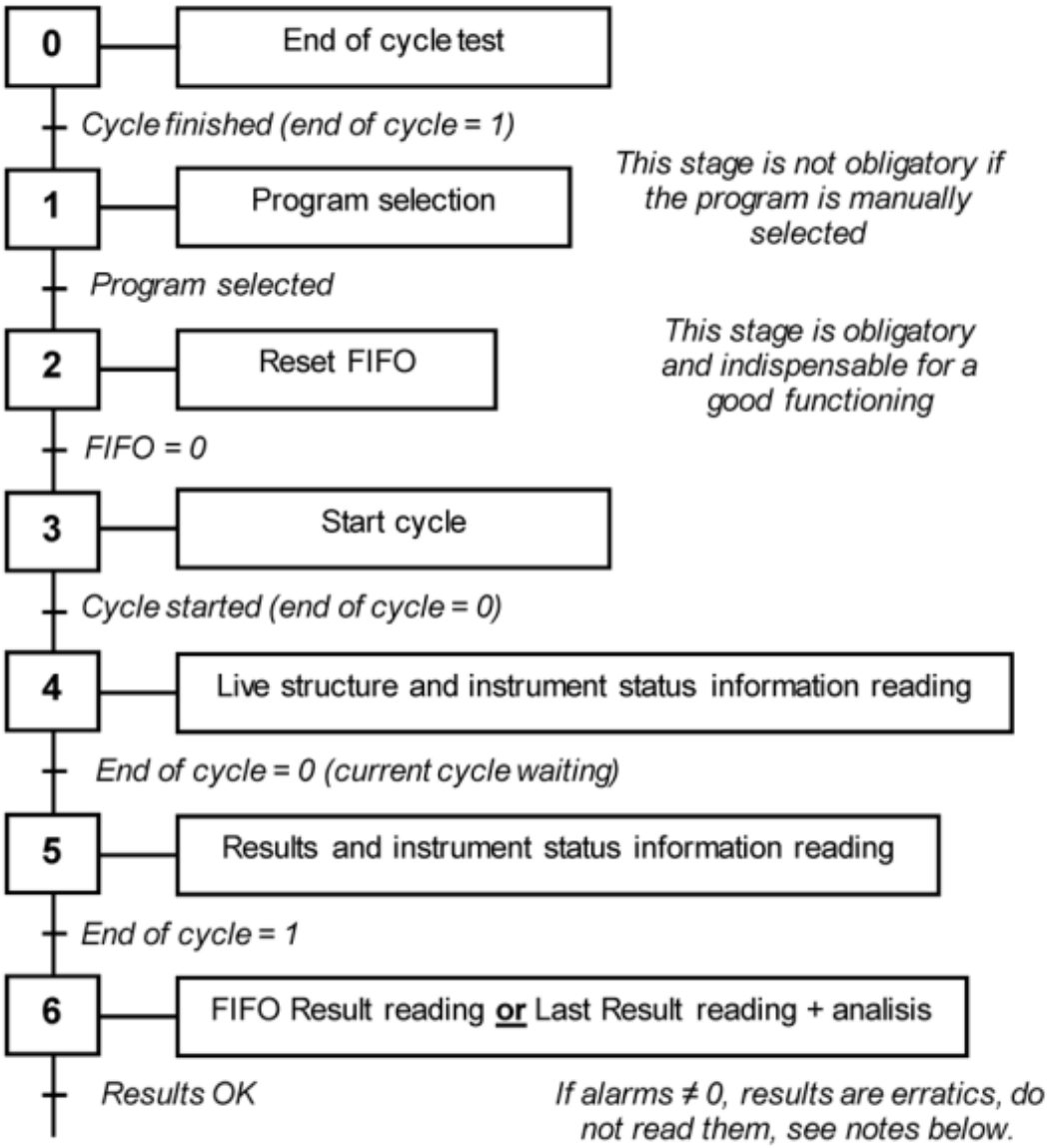
WRr/WRw = 96 words – Parameters handling, Program name & FIFO / Last results						
Offset (byte)	Length (byte)	Offset (word)	Read table (MASTER ← ATEQ)		Write table (MASTER → ATEQ)	
112	128	38(h)	WRr00(h) … WRr39(h)	Parameters reading	WRw00(h) … WRw39(h)	Parameters writing
240	16	78(h)	WRr40(h) … WRr47(h)	Program name reading	WRw40(h) … WRw47(h)	Program name writing
256	80	80(h)	WRr48(h) … WRr6F(h)	FIFO / Last results	WRw48(h) … WRw6F(h)	*RESERVED*
336	12	A8(h)	WRr70(h) … WRr7F(h)	*RESERVED*	WRw70(h) … WRw7F(h)	*RESERVED*

Treatment of the commands

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

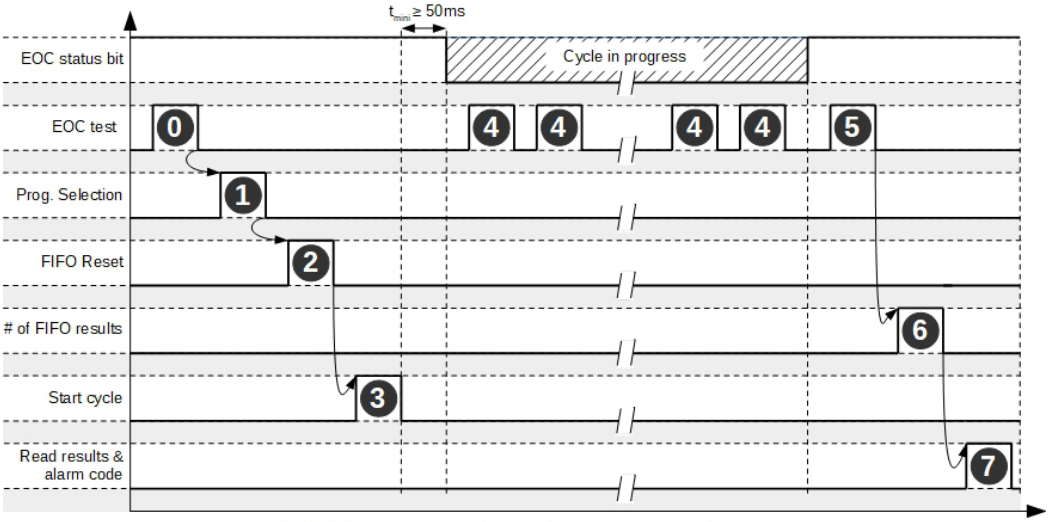
ATEQ using

Base procedure for using an ATEQ instrument:



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

Fieldbus progress chart



WARNING : The status bits update rate is about 50ms

0 : Read @06(h) : Status bit 5 = 1 (EOC status bit)

1 : Write @03(h) : 1 word = n° prog (0001h = prog 2)
Write @00(h) : Y003(h) = 1 (cmd « Prog. Selection »)

2 : **ALWAYS RESET THE FIFO**
Write @00(h) : Y007(h) = 1 (cmd « Reset FIFO »)

3 : Write @00(h) : Y001(h) = 1 (cmd « Start »)
t_{mini} ≥ 50ms

4 : Read @06(h) : Status bit 5 = 0 (EOC status bit)

5 : Read @06(h) : Status bit 5 = 1 (EOC status bit)

Use of FIFO Results

6 : Read the number of results in FIFO :
Read @04(h) : **if > 0** go to step 7, **else** END

7 : Write @00(h) : Y004(h) = 1 (command « Read FIFO results »)
Read Exchange datas : FIFO/Last results (standard or extended)
if Alarm Code = 0 go to step 8, **else** END

8 : Use the results recovered at step 7 (if Alarm code was equal to 0)

Use of Last Results

6 : *Read the number of results in FIFO :*
Read @04(h) : **if ≥ 1** go to step 7, **else** END

7 : Write @00(h) : Y00F(h) = 1 (command « Read Last results »)
Read Exchange datas : FIFO/Last results (standard or extended)
if Alarm Code = 0 go to step 8, **else** END

8 : Use the results recovered at step 7 (if Alarm code was equal to 0)

General configuration

Table of configuration / extended menu bits

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

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		Meaning	Menu
		Hexa	Dec		
1	0	0001	1	Fill type.	+Func
	1	0002	2	Pre-fill type.	+Func
	2	0004	4	Recovery thresholds.	+Func
	3	0008	8	Volume calculation	+Func
	4	0010	16	Personalization of the program name.	+Func
	5	0020	32	Chaining.	+Func
	6	0040	64	Automatic connector.	+Func
	7	0080	128	Valves codes (outputs codes)	+Func
	8	0100	256	Stamping.	+Func
	9	0200	512	Sending conditions: pass part	RS232
	10	0400	1024	Sending conditions: fail part maximum flow	RS232
	11	0800	2048	Sending conditions: presence of an alarm	RS232
	12	1000	4096	Sending conditions: pressure defect	RS232
	13	2000	8192	Sending conditions: end of cycle	RS232
	14	4000	16384	Sending conditions: recoverable	RS232
	15	8000	32768	Content of the frame: time	RS232

Word	Bit n°	Mask		Meaning	Menu
		Hexa	Dec		
2	16	0001	1	Content of the frame: personalization	RS232
	17	0002	2	Content of the frame: pressure	RS232
	18	0004	4	Security	Conf
	19	0008	8	External dump	Conf
	20	0010	16	Exportation	RS232
	21	0020	32	Automatic reset	Conf
	22	0040	64	*Reserved*	
	23	0080	128	*Reserved*	
	24	0100	256	*Reserved*	
	25	0200	512	Automatic start	+Funct
	26	0400	1024	Cut valve	Conf
	27	0800	2048	Filtering	+Funct
	28	1000	4096	*Reserved*	
	29	2000	8192	Pressure compensation	+Funct
	30	4000	16384	*Reserved*	
	31	8000	32768	Line feed (label)	RS232
3	32	0001	1	End of cycle	+Funct
	33	0002	2	Unit type	+Funct
	34	0004	4	Bar graph display	Conf
	35	0008	8	Negative rejects level	Conf
	36	0010	16	*Reserved*	
	37	0020	32	Bar code	RS232
	38	0040	64	Program selection bar code	
	39	0080	128	Bar code reset on end of cycle	
	40	0100	256	Auxiliary code activation	+Funct
	41	0200	512	Standard conditions	+Funct
	42	0400	1024	*Reserved*	
	43	0800	2048	Service cycle activation	
	44	1000	4096	Sign change activation	+Funct
	45	2000	8192	Peak hold	+Funct
	46	4000	16384	Negative flow display	+Funct
	47	8000	32768	*Reserved*	

Word	Bit n°	Mask		Meaning	Menu
		Hexa	Dec		
4	48	0001	1	Buzzer	+Funct
	49	0002	2	Display mode activation	+Funct
	50	0004	4	Sending conditions: fail part minimum flow	RS232
	51	0008	8	Offset	+Funct
	52	0010	16	Minimum flow activation	+Funct

Example: bit number 13 (automatic mode) 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

Master	Slave
— Activate the “Read extended menu bits” command: Write at the address 00(h), the value **0200(h)** Byte 0 = 00(h) Byte 1 = 02(h) (Y009(h) = 1)	
	<u>Acknowledgement</u> Command echo: — Byte 0 = 00(h) — Byte 1 = 02(h) (X009(h) = 1) Command error code: — Byte 2 = FF(h) — Byte 3 = FF(h) (if command error code = FFFF(h), command is in progress)
	Running “Read extended menu bits” command
	<u>Command finished</u> Command echo: — Byte 0 = 00(h) — Byte 1 = 02(h) (X009(h) = 1) Command error code if the command is correctly carried out: — Byte 2 = 00(h) — Byte 3 = 00(h) OR if an error occurred during the command: — Byte 2 = 00(h) — Byte 3 = 02(h) (X019(h) = 1)
— Wait the end of the command: command echo = 0200(h) (X009(h) = 1) command error code ≠ FFFF(h) (end of command)	
— Deactivate the “Read extended menu bits” command: Write at the address 00(h) the value **0000(h)** Byte 0 = 00(h) Byte 1 = 00(h) (Y009(h) = 0)	
— Read the extended menu bits in the corresponding area depending on the configuration mode of the slave	

The configuration / extended menu bits are defined in the table above for the “Extended menus” of each specific chapter for the instruments.

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

The master instrument must always set to zero the command bit. If it is not done, the slave instrument will not detect the following command on this bit. It has detection on the rising edge (when the bit state goes from 0 to 1).

Writing of the configuration / extended menu bits

Master	Slave
— Write the extended menu bits in the corresponding area depending on the configuration mode of the slave — Activate the “Write extended menu bits” command: Write at the address 00(h), the value **0800(h)** Byte 0 = 00(h) Byte 1 = 08(h) (Y00B(h) = 1)	
	<u>Acknowledgement</u> Command echo: — Byte 0 = 00(h) — Byte 1 = 08(h) (X00B(h) = 1) Command error code: — Byte 2 = FF(h) — Byte 3 = FF(h) (if command error code = FFFF(h), command is in progress)
	Running “Write extended menu bits” command
	<u>Command finished</u> Command echo: — Byte 0 = 00(h) — Byte 1 = 08(h) (X00B(h) = 1) Command error code if the command is correctly carried out: — Byte 2 = 00(h) — Byte 3 = 00(h) OR if an error occurred during the command: — Byte 2 = 00(h) — Byte 3 = 08(h) (X01B(h) = 1)
— Wait the end of the command: command echo = 0800(h) (X00B(h) = 1) command error code ≠ FFFF(h) (end of command)	
— Deactivate the “Write extended menu bits” command: Write at the address 00(h) the value **0000(h)** Byte 0 = 00(h) Byte 1 = 00(h) (Y00B(h) = 0)	

The configuration / extended menu bits are defined in the table above for the “Extended menus” of each specific chapter for the instruments.

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

The master instrument must always set to zero the command bit. If it is not done, the slave instrument will not detect the following command on this bit. It has detection on the rising edge (when the bit state goes from 0 to 1).

Program

Program selection command on the ATEQ device

Master	Slave
— Write 1 word at the address 03(h) corresponding to the program number to be selected: @03(h) = 0001(h) (= program n°2) — Activate the “Program selection” command: Write at the address 00(h), the value **0008(h)** Byte 0 = 08(h) (Y003(h) = 1) Byte 1 = 00(h)	
	Acknowledgement Command echo: — Byte 0 = 08(h) (X003(h) = 1) — Byte 1 = 00(h) Command error code: — Byte 2 = FF(h) — Byte 3 = FF(h) (if command error code = FFFF(h), command is in progress)
	Running “Program selection” command
	Command finished Command echo: — Byte 0 = 08(h) (X003(h) = 1) — Byte 1 = 00(h) Command error code if the command is correctly carried out: — Byte 2 = 00(h) — Byte 3 = 00(h) OR if an error occurred during the command: — Byte 2 = 08(h) (X013(h) = 1) — Byte 3 = 00(h)
— Wait the end of the command: command echo = 0008(h) (X003(h) = 1) command error code ≠ FFFF(h) (end of command)	
— Deactivate the “Program selection” command: Write at the address 00(h) the value **0000(h)** Byte 0 = 00(h) (Y003(h) = 0) Byte 1 = 00(h)	

The master instrument must always set to zero the command bit. If it is not done, the slave instrument will not detect the following command on this bit. It has detection on the rising edge (when the bit state goes from 0 to 1).

Function

Table of the function bits

Table of the function bits per program.

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

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		Meaning	Menu
		Hexa	Dec		
1	0	0001	1	Fill type activation	Funct
	1	0002	2	Pre-fill type activation	Funct
	2	0004	4	Recovery thresholds activation	Funct
	3	0008	8	Cycle end activation	Funct
	4	0010	16	Cycle end with reset and piezo reset activation	
	5	0020	32	Cycle end with dump and reset activation	
	6	0040	64	Cycle end with fill activation	
	7	0080	128	Chaining activation	Funct
	8	0100	256	Pass part chaining activation	
	9	0200	512	Fail part maximum flow chaining activation	
	10	0400	1024	Alarm chaining activation	
	11	0800	2048	Pressure switch error chaining activation	
	12	1000	4096	Cycle end chaining activation	
	13	2000	8192	Recovery chaining activation	
	14	4000	16384	Automatic connector chaining activation	Funct
	15	8000	32768	Valve code activation	
2	16	0001	1	Valve code ext. 1 activation	
	17	0002	2	Valve code ext. 2 activation	
	18	0004	4	Valve code ext. 3 activation	
	19	0008	8	Valve code ext. 4 activation	
	20	0010	16	Valve code ext. 5 activation	
	21	0020	32	Valve code ext. 6 activation	
	22	040	64	Valve code int. 1 activation	
	23	0080	128	Valve code int. 8 activation	
	24	0100	256	Stamping activation	Funct
	25	0200	512	Pass part stamping activation	
	26	0400	1024	Fail part maximum flow stamping activation	
	27	0800	2048	Alarm stamping activation	
	28	1000	4096	Pressure switch error stamping activation	
	29	2000	8192	Cycle end stamping activation	
	30	4000	16384	Recovery stamping activation	
	31	8000	32768	External dump activation	Funct

Word	Bit n°	Mask		Meaning	Menu
		Hexa	Dec		
3	32	0001	1	Reserved	
	33	0002	2	Automatic start cycle activation	Funct
	34	0004	4	Pressure compensation activation	Funct
	35	0008	8	Filtering activation	Funct
	36	0010	16	Standard conditions activation	Funct
	37	0020	32	Bar code activation	
	38	0040	64	Start after reading bar code	
	39	0080	128	Auxiliaries code activation	
	40	0100	256	Auxiliary code 1 activation	
	41	0200	512	Auxiliary code 2 activation	
	42	0400	1024	Auxiliary code 3 activation	
	43	0800	2048	Auxiliary code 4 activation	
	44	1000	4096	Optional auxiliaries code activation	
	45	2000	8192	Optional auxiliary code 1 activation	
	46	4000	16384	Optional auxiliary code 2 activation	
	47	8000	32768	Optional auxiliary code 3 activation	
4	48	0001	1	Optional auxiliary code 4 activation	
	49	0002	2	Optional valve code activation	
	50	0004	4	Optional valve code ext. 1 activation	
	51	0008	8	Optional valve code ext. 2 activation	
	52	0010	16	Optional valve code ext. 3 activation	
	53	0020	32	Optional valve code ext. 4 activation	
	54	0040	64	Optional valve code ext. 5 activation	
	55	0080	128	Optional valve code ext. 6 activation	
	56	0100	256	Optional valve code int. 1 activation	
	57	0200	512	Optional valve code int. 2 activation	
	58	0400	1024	Sign change activation	Funct
	59	0800	2048	Peak hold activation	Funct
	60	1000	4096	Negative flow display activation	Funct
	61	2000	8192	Buzzer activation	
	62	4000	16384	Cycle end buzzer activation	
	63	8000	32768	Pass part buzzer activation	

Word	Bit n°	Mask		Meaning	Menu
		Hexa	Dec		
5	64	0001	1	Fail part maximum flow buzzer activation	
	65	0002	2	Alarm buzzer activation	Funct
	66	0004	4	Automatic mode activation	Funct
	67	0008	8	Reserved	
	68	0010	16	Reserved	
	69	0020	32	Reserved	
	70	0040	64	Offset activation	Funct
	71	0080	128	Minimum flow activation	Funct

Example: bit number 46 (Offset 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

Master	Slave
— Select the program number on which the functions bits have to be read — Activate the “Read functions bits” command: Write at the address 00(h), the value **0400(h)** Byte 0 = 00(h) Byte 1 = 04(h) (Y00A(h) = 1)	
	Acknowledgement Command echo: — Byte 0 = 00(h) — Byte 1 = 04(h) (X00A(h) = 1) Command error code: — Byte 2 = FF(h) — Byte 3 = FF(h) (if command error code = FFFF(h), command is in progress)
	Running “Read functions bits” command
	Command finished Command echo: — Byte 0 = 00(h) — Byte 1 = 04(h) (X00A(h) = 1) Command error code if the command is correctly carried out: — Byte 2 = 00(h) — Byte 3 = 00(h) OR if an error occurred during the command: — Byte 2 = 00(h) — Byte 3 = 04(h) (X01A(h) = 1)
— Wait the end of the command: command echo = 0400(h) (X00A(h) = 1) command error code ≠ FFFF(h) (end of command)	
— Deactivate the “Read functions bits” command: Write at the address 00(h) the value **0000(h)** Byte 0 = 00(h) Byte 1 = 00(h) (Y00A(h) = 0)	
— Read the functions bits in the corresponding area depending on the configuration mode of the slave	

The functions bits are dependents of the program number.
A program selection has to be realised before executing command.

The master instrument must always set to zero the command bit. If it is not done, the slave instrument will not detect the following command on this bit. It has detection on the rising edge (when the bit state goes from 0 to 1).

Writing of the function bits

Master	Slave
— Select the program number on which the functions bits have to be read. — Write the functions bits in the corresponding area depending on the configuration mode of the slave. — Activate the “Write functions bits” command: Write at the address 00(h), the value **1000(h)** Byte 0 = 00(h) Byte 1 = 10(h) (Y00C(h) = 1)	
	Acknowledgement Command echo: — Byte 0 = 00(h) — Byte 1 = 10(h) (X00C(h) = 1) Command error code: — Byte 2 = FF(h) — Byte 3 = FF(h) (if command error code = FFFF(h), command is in progress)
	Running “Write functions bits” command
	Command finished Command echo: — Byte 0 = 00(h) — Byte 1 = 10(h) (X00C(h) = 1) Command error code if the command is correctly carried out: — Byte 2 = 00(h) — Byte 3 = 00(h) OR if an error occurred during the command: — Byte 2 = 00(h) — Byte 3 = 10(h) (X01C(h) = 1)
— Wait the end of the command: command echo = 1000(h) (X00C(h) = 1) command error code ≠ FFFF(h) (end of command)	
— Deactivate the “Write functions bits” command: Write at the address 00(h) the value **0000(h)** Byte 0 = 00(h) Byte 1 = 00(h) (Y00C(h) = 0)	

The functions bits are dependents of the program number.
A program selection has to be realised before executing command.

The master instrument must always set to zero the command bit. If it is not done, the slave instrument will not detect the following command on this bit. It has detection on the rising edge (when the bit state goes from 0 to 1).

Parameters

Downloading of the parameters

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°		Meaning	Value	
Dec	Hexa			
1	0001	"FILL TIME" Fill time	0 > 650 seconds	
2	0002	"STAB TIME": Stabilization time	0 > 650 seconds	
3	0003	"TEST TIME" Test time	0 > 650 seconds	
6	0006	"PRE FILL" Pre fill time	0 > 650 seconds	
9	0009	"DUMP TIME" Dump time	0 > 650 seconds	
10	000A	"COUPL. A": Coupling time 1	0 > 650 seconds	
11	000B	"COUPL. B": Coupling time 2	0 > 650 seconds	
20	0014	"VOLUME" Part volume.	0 > 9999	
21	0015	"TYPE": Test type	Invalid Direct Operator	0000 1000 2000
29	001D	"Inter-Cycle": Time between 2 chained cycles	0 > 650 seconds	
48	0030	"DURATION" Maintain time of the result during stamp	0 > 650 seconds	
50	0032	"Min FILL" Minimum pressure value	- 9999 > 9999.	
51	0033	"Max FILL" Maximum pressure value	- 9999 > 9999.	
53	0035	"Press. UNIT" Pressure unit.	Refer to Unit table.	
60	003C	"Test FAIL" Natural reject value of the test part	0 > 9999	
61	003D	"TestREWORK" Natural reject level of the test part in recovery	0 > 9999	
62	003E	"Ref. FAIL" Natural reject level of the reference part	0 > 9999	
63	003F	"Ref.REWORK" Natural reject value of the reference part in recovery:	0 > 9999	
66	0042	"Set FILL" Fill instruction value:	-9999 > 9999	

Identifier N°		Meaning	Value	
Dec	Hexa			
80	0050	“Diff A-Z” Differential auto reset time.	0 > 650 seconds	
103	0067	“FILL MODE” Type of fill.	Standard Instruction Ballistic Ramp Adjust EASY EASY Auto	0000 1000 2000 3000 4000 5000 6000
110	006E	“EXT. DUMP” Type of external dump.	Normally close Normally open	0000 1000
112	0070	‘IN7:” Function attributed to the entry of the special cycles (output 7)	Refer to the “Configure input value” table at the end of the chapter	
123	007B	“LANGUAGE” Choice of the language.	Default language 2nd language	0000 1000
126	007E	“Max PreFILL” Maximum pressure value in pre-fill.	-9999 > 9999	
127	007F	“Flow Unit” Reject unit.	Refer to Unit table.	
128	0080	“Leak Rate” Instruction value during a calibration.	0 > 9999	
148	0094	“FILTER” Filtering.	0 > 650 seconds	
149	0095	“UNITS” Unit type	SI SAE CUSTOM	0000 1000 2000
158	009E	“Max rej.” Percents of the bar graph.	70% 50% 30%	0000 1000 2000
161	00A1	“Volume UNIT” Volume unit.	Refer to Unit table.	
164	00A4	“NEXT PROG.” Number of the following program in sequencing.	1 > 128	
165	00A5	“N. OF CYCLES”(PIEZO AUTO AZ menu) Number of cycles between two automatic reset.	0 > 9999	
166	00A6	“N. OF MINUTES”(PIEZO AUTO AZ menu) Time between two automatic reset.	0 > 999 minutes	
249	00F9	“DELAY EXT1” Programmed external output 1 delay time.	0 > 650 seconds	
250	00FA	“DELAY EXT2” Programmed external output 2 delay time.	0 > 650 seconds	
251	00FB	“DELAY EXT3” Programmed external output 3 delay time.	0 > 650 seconds	
252	00FC	“DELAY EXT4” Programmed external output 4 delay time.	0 > 650 seconds	
253	00FD	“DELAY EXT5” Programmed external output 5 delay time.	0 > 650 seconds	
254	00FE	“DELAY EXT6” Programmed external output 6 delay time.	0 > 650 seconds	

Identifier N°		Meaning	Value	
Dec	Hexa			
255	00FF	“DELAY INT2” Programmed internal output 2 delay time.	0 > 650 seconds	
256	0100	“DELAY INT1” Programmed internal output 1 delay time.	0 > 650 seconds	
257	0101	“DELAY AUX1” Programmed auxiliary output 1 delay time.	0 > 650 seconds	
258	0102	“DELAY AUX2” Programmed auxiliary output 2 delay time.	0 > 650 seconds	
259	0103	“DELAY AUX3” Programmed auxiliary output 3 delay time.	0 > 650 seconds	
260	0104	“DELAY AUX4” Programmed auxiliary output 4 delay time.	0 > 650 seconds	
261	0105	“TIME EXT1” Programmed external output 1 duration time.	0 > 650 seconds	
262	0106	“TIME EXT2” Programmed external output 2 duration time.	0 > 650 seconds	
263	0107	“TIME EXT3” Programmed external output 3 duration time.	0 > 650 seconds	
264	0108	“TIME EXT4” Programmed external output 4 duration time.	0 > 650 seconds	
265	0109	“TIME EXT5” Programmed external output 5 duration time.	0 > 650 seconds	
266	010A	“TIME EXT6” Programmed external output 6 duration time.	0 > 650 seconds	
267	010B	“TIME INT2” Programmed internal output 2 duration time.	0 > 650 seconds	
268	010C	“TIME INT1” Programmed internal output 1 duration time.	0 > 650 seconds	
269	010D	“TIME AUX1” Programmed auxiliary output 1 duration time.	0 > 650 seconds	
270	010E	“TIME AUX2” Programmed auxiliary output 2 duration time.	0 > 650 seconds	
271	010F	“TIME AUX3” Programmed auxiliary output 3 duration time.	0 > 650 seconds	
272	0110	“TIME AUX4” Programmed auxiliary output 4 duration time.	0 > 650 seconds	
274	0112	“FILTER” Pressure filtering.	0 > 650 seconds	
281	0119	“RANGE” Capillary number with dual capillaries option only:	Capillary 1 Capillary 2	0000 1000
287	011F	“First Char.” Start on bar code.	0 > 40	
288	0120	“Char. Number” Number of character of bar code.	0 > 40	
289	0121	“Pr “ Program bar code.	1 > 128	
353	0161	“Press. UNIT” (configuration/pneumatique menu) General pressure unit	Refer to Unit table.	
354	0162	“LINE P. MIN” Minimum line pressure level	-9999 > 9999	

Identifier N°		Meaning	Value	
Dec	Hexa			
364	016C	“DISPLAY MODE“ Leak display management	XXXX XXX.X XX.XX X.XXX	0000 1000 2000 3000
375	0177	‘IN8:” Function attributed to the entry of the special cycles (output 8)	Refer to the “Configure input value” table at the end of the chapter	
376	0178	‘IN9:” Function attributed to the entry of the special cycles (output 9)	Refer to the “Configure input value” table at the end of the chapter	
379	017B	“USB:” USB mode (printer or supervision)	Supervision Printer Bar code Auto None	0000 1000 2000 3000 4000
412	019C	“SAVE ON” Mode of Results stocking.	None Internal USB	0000 1000 2000
413	019D	“ACCESS” Access parameters mode.	None USB Password	0000 1000 2000
414	019E	“YEAR” Year configuration.	2000 > 9999	
415	019F	“MONTH” Month configuration.	1 > 12	
416	01A0	“DAY” Day configuration.	1 > 31	
417	01A1	“HOUR” Hour configuration.	0 > 59	
418	01A2	“MINUTE” Minute configuration.	0 > 59	
419	01A3	“SECOND” Second configuration.	0 > 59	
459	01CB	“N. OF CYCLES” Number of learning cycle	2 > 9999	
460	01CC	“INTER-CYCLE” Time between 2 learning cycle	0 > 650 seconds	
461	01CD	“MAX OFFSET” Offset max for learning cycle	0 > 9999	
462	01CE	“FLOW MASTER” Value of Flow master for learning cycle	0 > 9999	
463	01CF	“PRESS MASTER” Value of Pressure master for learning cycle	-9999 > 9999	
464	01D0	“Min. Vol.” Minimum Volume for learning	0 > 9999	
465	01D1	“Max. Vol.” Maximum Volume for learning	0 > 9999	
486	01E6	“OFFSET” Offset Learning	-9999 > 9999	

Configurable input values

Input value	Value code
Program Selection	0000
Capil. Temp. Check (*)	10000
Temperature Check (*)	11000
Atm Pressure Check (*)	12000
P1 Sensor Check (*)	13000
Flow Check Cap 1(*)	14000
Flow Check Cap 2(*)	15000
Line P. Sensor Check (*)	16000
Regulator Adjust.	17000
Infinite Fill	18000
Piezo Az	19000
Code Reader	20000
Pre-Regul. Adjust.	21000
Print Results	22000
Volume Comp.	23000
Leak Offset Learn	24000
Offset+Vol. Learn	25000

(*) 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
6000	1770	Pascal
11000	2AF8	Bar
12000	2EE0	Kilopascal
13000	32C8	PSI
14000	36B0	Millibar
15000	3A98	Megapascal
30000	7530	Liter/hour
46000	B3B0	$Inch^3/s$
47000	B798	$Inch^3/min$
48000	BB80	$Inch^3/hour$
49000	BF68	$Feet^3/hour$
50000	C350	Milliliter/second
51000	C738	Milliliter/minute
52000	CB20	Milliliter/hour
55000	D6D8	mm^3
56000	DAC0	cm^3
61000	EE48	Milliliter
62000	F230	Liter
63000	F618	$inch^3$
64000	FA00	$feet^3$
84000	01 4820	SCCM
92000	01 6760	Points

Reading of the parameters

The reading of the parameters is carried out by data exchange in the corresponding area depending on the configuration mode of the slave. Each parameter is identified by one identifier. See identifiers tables.

This table is an example based on the reading of two parameters:

- **Test time** (identifier number 3)
- **Stabilization time** (identifier number 2)

Master	Slave
— Select the program on which parameters has to be read — Write in the parameter area depending on the configuration mode of the slave, the number of parameters followed by their identifiers: On network: 02 00 03 00 02 00 0002(h) 0003(h) 0002(h) 0002(h) = two parameters 0003(h) = test time identifier 0002(h) = stabilization time identifier — Activate the “Read parameters” command: Write at the address 00(h), the value **0020(h)** Byte 0 = 20(h) (Y005(h) = 1) Byte 1 = 00(h)	
	Acknowledgement Command echo: — Byte 0 = 20(h) (X005(h) = 1) — Byte 1 = 00(h) Command error code: — Byte 2 = FF(h) — Byte 3 = FF(h) (if command error code = FFFF(h), command is in progress)
	Running “Read parameters” command
	Command finished Command echo: — Byte 0 = 20(h) (X005(h) = 1) — Byte 1 = 00(h) Command error code if the command is correctly carried out: — Byte 2 = 00(h) — Byte 3 = 00(h) OR if an error occurred during the command: — Byte 2 = 20(h) (X015(h) = 1) — Byte 3 = 00(h)
— Wait the end of the command: command echo = 0020(h) (X005(h) = 1) command error code ≠ FFFF(h) (end of command)	
— Deactivate the “Read parameters” command: Write at the address 00(h) the value **0000(h)** Byte 0 = 00(h) (Y005(h) = 0) Byte 1 = 00(h)	

Master	Slave
— Read the parameters at the address 20(h): Word 1 = identifier number of the first read parameter. Word 2 and Word 3 = first parameter value x1000 (long format). Word 4 = second identifier number of the read parameter. Word 5 and Word 6 = second parameter value x1000 (long format). **Example**: On network: 03 00 E8 03 00 00 02 00 F4 01 00 00 0003h 03E8h 0000h 0002h 01F4h 0000h - 0003h: test time identifier. - 000003E8h: test time value 1000(d)/1000 → 1 sec. - 0002h: fill time identifier. - 000001F4h: stabilization time value 500(d)/1000 → 0,5 sec.	

The master instrument must always set to zero the command bit. If it is not done, the slave instrument will not detect the following command on this bit. It has detection on the rising edge (when the bit state goes from 0 to 1).

Writing of the parameters

The writing of the parameters is carried out by data exchange in the corresponding area depending on the configuration mode of the slave. Each parameter is identified by one identifier. See identifiers tables.

This table is an example based on the reading of two parameters:

- **Test time** (identifier number 3)
- **Stabilization time** (identifier number 2)

Master	Slave
— Select the program on which the parameters have to be modified — Write in the parameter area depending on the configuration mode of the slave, the number of parameters followed by their identifiers and their wanted value: **Example:** On network: 02 00 03 00 E8 03 00 00 02 00 D0 07 00 00 0002(h) 0003(h) 03E8(h) 0000(h) 0002(h) 07D0(h) 0000(h) 0002(h) = two parameters 0003(h) = test time identifier 000003E8(h) = 1000 => 1 second 0002(h) = stabilization time identifier 000007D0(h) = 2000 => 2 second — Activate the “Write parameters” command: Write at the address 00(h), the value **0040(h)** Byte 0 = 40(h) (Y006(h) = 1) Byte 1 = 00(h)	
	Acknowledgement Command echo: — Byte 0 = 40(h) (X006(h) = 1) — Byte 1 = 00(h) Command error code: — Byte 2 = FF(h) — Byte 3 = FF(h) (if command error code = FFFF(h), command is in progress)
	Running “Write parameters” command
	Command finished Command echo: — Byte 0 = 40(h) (X006(h) = 1) — Byte 1 = 00(h) Command error code if the command is correctly carried out: — Byte 2 = 00(h) — Byte 3 = 00(h) OR if an error occurred during the command: — Byte 2 = 40(h) (X016(h) = 1) — Byte 3 = 00(h)

Master	Slave
— Wait the end of the command: command echo = 0040(h) (X006(h) = 1) command error code ≠ FFFF(h) (end of command)	
— Deactivate the “Write parameters” command: Write at the address 00(h) the value **0000(h)** Byte 0 = 00(h) (Y006(h) = 0) Byte 1 = 00(h)	

The master instrument must always set to zero the command bit. If it is not done, the slave instrument will not detect the following command on this bit. It has detection on the rising edge (when the bit state goes from 0 to 1).

Reading of the program name

Master	Slave
— Select the program whose name you want to read — Activate the “Read program name” command: Write at the address 00(h), the value **2000(h)** Byte 0 = 00(h) Byte 1 = 20(h) (Y00D(h) = 1)	
	Acknowledgement Command echo: — Byte 0 = 00(h) — Byte 1 = 20(h) (X00D(h) = 1) Command error code: — Byte 2 = FF(h) — Byte 3 = FF(h) (if command error code = FFFF(h), command is in progress)
	Running “Read program name” command
	Command finished Command echo: — Byte 0 = 00(h) — Byte 1 = 20(h) (X00D(h) = 1) Command error code if the command is correctly carried out: — Byte 2 = 00(h) — Byte 3 = 00(h) OR if an error occurred during the command: — Byte 2 = 00(h) — Byte 3 = 20(h) (X01D(h) = 1)
— Wait the end of the command: command echo = 2000(h) (X00D(h) = 1) command error code ≠ FFFF(h) (end of command)	
— Deactivate the “Read program name” command: Write at the address 00(h) the value **0000(h)** Byte 0 = 00(h) Byte 1 = 00(h) (Y00D(h) = 0)	
— Read the program name of 12 characters/bytes maximum in the corresponding area depending on the configuration mode of the slave.	

The program name is dependant of the program number in edition, a program selection has to be realized.

The master instrument must always set to zero the command bit. If it is not done, the slave instrument will not detect the following command on this bit. It has detection on the rising edge (when the bit state goes from 0 to 1).

Writing of the program name

Master	Slave
— Select the program whose name you want to modify — Write the program name of 12 characters/bytes maximum in the corresponding area depending on the configuration mode of the slave. — Activate the “Write program name” command: Write at the address 00(h), the value **4000(h)** Byte 0 = 00(h) Byte 1 = 40(h) (Y00E(h) = 1)	
	Acknowledgement Command echo: — Byte 0 = 00(h) — Byte 1 = 40(h) (X00E(h) = 1) Command error code: — Byte 2 = FF(h) — Byte 3 = FF(h) (if command error code = FFFF(h), command is in progress)
	Running “Write program name” command
	Command finished Command echo: — Byte 0 = 00(h) — Byte 1 = 40(h) (X00E(h) = 1) Command error code if the command is correctly carried out: — Byte 2 = 00(h) — Byte 3 = 00(h) OR if an error occurred during the command: — Byte 2 = 00(h) — Byte 3 = 40(h) (X01E(h) = 1)
— Wait the end of the command: command echo = 4000(h) (X00E(h) = 1) command error code ≠ FFFF(h) (end of command)	
— Deactivate the “Write program name” command: Write at the address 00(h) the value **0000(h)** Byte 0 = 00(h) Byte 1 = 00(h) (Y00E(h) = 0)	

The program name is dependant of the program number in edition, a program selection has to be realized.

The master instrument must always set to zero the command bit. If it is not done, the slave instrument will not detect the following command on this bit. It has detection on the rising edge (when the bit state goes from 0 to 1).

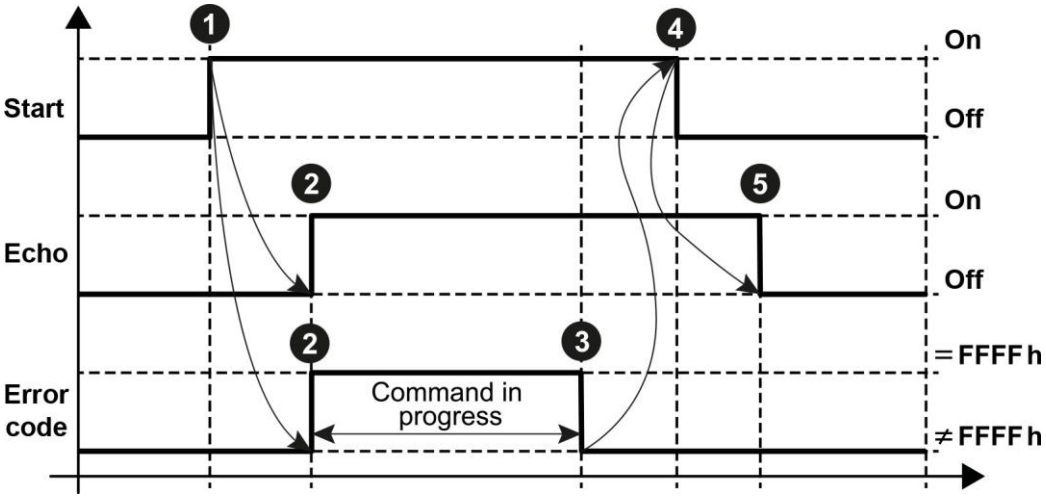
Standard command cycle

Start cycle command on the ATEQ device

Master	Slave
— Select the program you want to start — Activate the “Start” command: Write at the address 00(h), the value **0002(h)** Byte 0 = 02(h) (Y001(h) = 1) Byte 1 = 00(h)	
	Acknowledgement Command echo: — Byte 0 = 02(h) (X001(h) = 1) — Byte 1 = 00(h) Command error code: — Byte 2 = FF(h) — Byte 3 = FF(h) (if command error code = FFFF(h), command is in progress)
	Running “Start” command
	Command finished Command echo: — Byte 0 = 02(h) (X001(h) = 1) — Byte 1 = 00(h) Command error code if the command is correctly carried out: — Byte 2 = 00(h) — Byte 3 = 00(h) OR if an error occurred during the command: — Byte 2 = 02(h) (X011(h) = 1) — Byte 3 = 00(h)
— Wait the end of the command: command echo = 0002(h) (X001(h) = 1) command error code ≠ FFFF(h) (end of command)	
— Deactivate the “Start” command: Write at the address 00(h) the value **0000(h)** Byte 0 = 00(h) (Y001(h) = 0) Byte 1 = 00(h)	

The master instrument must always set to zero the command bit. If it is not done, the slave instrument will not detect the following command on this bit. It has detection on the rising edge (when the bit state goes from 0 to 1).

Start command diagram



1	**Start** command = On
2	Acknowledge by ATEQ = (**Echo** command = On) and (**Error code** command = FFFFh)
3	Wait end of **Start** command = (**Echo** command = On) and (**Error code** command ≠ FFFFh)
4	**Start** command = Off
5	Acknowledge by ATEQ = (**Echo** command = Off) and (**Error code** command ≠ FFFFh)

The Echo command is a copy of the Start command. The Start signal must be maintained (ON) till the end of the Start command condition is reached.

Reset command on the ATEQ device

Master	Slave
— Activate the “Reset” command: — Write at the address 00(h), the value **0001(h)** Byte 0 = 01(h) (Y000(h) = 1) Byte 1 = 00(h)	
	<u>Acknowledgement</u> Command echo: — Byte 0 = 01(h) (X000(h) = 1) — Byte 1 = 00(h) Command error code: — Byte 2 = FF(h) — Byte 3 = FF(h) (if command error code = FFFF(h), command is in progress)
	Running “Reset” command
	<u>Command finished</u> Command echo: — Byte 0 = 01(h) (X000(h) = 1) — Byte 1 = 00(h) Command error code if the command is correctly carried out: — Byte 2 = 00(h) — Byte 3 = 00(h) OR if an error occurred during the command: — Byte 2 = 01(h) (X010(h) = 1) — Byte 3 = 00(h)
— Wait the end of the command: command echo = 0001(h) (X000(h) = 1) command error code ≠ FFFF(h) (end of command)	
— Deactivate the “Reset” command: Write at the address 00(h) the value **0000(h)** Byte 0 = 00(h) (Y000(h) = 0) Byte 1 = 00(h)	

The master instrument must always set to zero the command bit. If it is not done, the slave instrument will not detect the following command on this bit. It has detection on the rising edge (when the bit state goes from 0 to 1).

Special cycles

Special cycle table

Write the identifier number of the wanted special cycle at the address 04(h) and its instruction if necessary.
Y040(h) → Y047(h) = identifier number of the special cycle
Y048(h) → Y04F(h) = instruction for the special cycle

Numb	Special cycle
1	ATR learning Cycle.
4	Custom Unit Learn.
5	Custom Unit Check.
9	Piezo auto zero.
13	Regulator adjust.
25	Capil. Temp. Check (*).
26	Temperature Check (*).
27	Atm Pressure Check (*).
28	P1 Sensor Check (*).
29	Flow 1 Check (*).
30	Flow 2 Check (*).
31	Line P. Sensor check (*).

To activate a special cycle, you must send a **Start** command (Y001(h)) and a **Start special cycle** command (Y002(h)).
(*) Appears with the **Service special cycle** function checked.

Auto-zero on the ATEQ device

Master	Slave
— Select the program on which you want to make the auto zero — Write at the address 04(h) the identifier number of the special cycle for an auto zero — Activate the “Start” and the “Start special cycle” commands: Write at the address 00(h), the value **0006(h)** Byte 0 = 06(h) (Y001(h) = 1 and Y002(h) = 1) Byte 1 = 00(h)	
	<u>Acknowledgement</u> Command echo: — Byte 0 = 06(h) (X001(h) = 1 and X002(h) = 1) — Byte 1 = 00(h) Command error code: — Byte 2 = FF(h) — Byte 3 = FF(h) (if command error code = FFFF(h), command is in progress)
	Running “Start” and “Start special cycle” commands
	<u>Command finished</u> Command echo: — Byte 0 = 06(h) (X001(h) = 1 and X002(h) = 1) — Byte 1 = 00(h) Command error code if the command is correctly carried out: — Byte 2 = 00(h) — Byte 3 = 00(h) OR if an error occurred during the command: — Byte 2 = 06(h) (X011(h) = 1 and X012(h) = 1) — Byte 3 = 00(h)
— Wait the end of the command: command echo = 0006(h) (X001(h) = 1 and X002(h) = 1) command error code ≠ FFFF(h) (end of command)	
— Deactivate the “Start” and “Start special cycle” commands: Write at the address 00(h) the value **0000(h)** Byte 0 = 00(h) (Y001(h) = 0 and Y002(h) = 0) Byte 1 = 00(h)	

The master instrument must always set to zero the command bit. If it is not done, the slave instrument will not detect the following command on this bit. It has detection on the rising edge (when the bit state goes from 0 to 1).

FIFO results

FIFO list results structure

At the end of each cycle, a result is stored as an array of 12 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.

Standard results

The standard result structure contains **24 bytes**.

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	Flow low section word.	Long	4	x1000
10	Flow high section word.			
11	Flow unit code low part word (refer to. Units table).	Long	4	x1000
12	Flow unit code high part word (refer to. Units table).			

Extended results

The extended results are not yet used in the current version of G6 (V1.002).
They are the same as the standard results.

The extended result structure contains **24 bytes**.

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	Flow low section word.	Long	4	x1000
10	Flow high section word.			
11	Flow unit code low part word (refer to. Units table).	Long	4	x1000
12	Flow unit code high part word (refer to. Units table).			

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	Fill
2	0002	Zero Diff.
3	0003	Stabilization
4	0004	Test
5	0005	Dump
65535	FFFF	No step 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).
43	002B	Pressure too high.
44	002C	Pressure too low.
45	002D	Piezo sensor out of order.
46	002E	Dump error.
47	002F	Calibration drift.
73	0049	Atmospheric pressure error.
74	004A	Temperature error.

Cycle results reading (last 8 results in FIFO)

Master	Slave
— Read the number of available results in the FIFO at the address 04(h): 04(h) = 0000(h) → no results 04(h) > 0000(h) → results available — Activate the “Read FIFO results” command: Write at the address 00(h), the value **0010(h)** Byte 0 = 10(h) (Y004(h) = 1) Byte 1 = 00(h)	
	<u>Acknowledgement</u> Command echo: — Byte 0 = 10(h) (X004(h) = 1) — Byte 1 = 00(h) Command error code: — Byte 2 = FF(h) — Byte 3 = FF(h) (if command error code = FFFF(h), command is in progress)
	Running “Read FIFO results” command
	<u>Command finished</u> Command echo: — Byte 0 = 10(h) (X004(h) = 1) — Byte 1 = 00(h) Command error code if the command is correctly carried out: — Byte 2 = 00(h) — Byte 3 = 00(h) OR if an error occurred during the command: — Byte 2 = 10(h) (X014(h) = 1) — Byte 3 = 00(h)
— Wait the end of the command: command echo = 0010(h) (X004(h) = 1) command error code ≠ FFFF(h) (end of command)	
— Deactivate the “Read FIFO results” command: Write at the address 00(h) the value **0000(h)** Byte 0 = 00(h) (Y004(h) = 0) Byte 1 = 00(h)	
— Read the result of 24 bytes maximum in the corresponding area depending on the configuration mode of the slave.	

The master instrument must always set to zero the command bit. If it is not done, the slave instrument will not detect the following command on this bit. It has detection on the rising edge (when the bit state goes from 0 to 1).

Reset FIFO results

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

Master	Slave
— Activate the "Reset FIFO results" command: Write at the address 00(h), the value **0080(h)** Byte 0 = 80(h) (Y007(h) = 1) Byte 1 = 00(h)	
	Acknowledgement Command echo: — Byte 0 = 80(h) (X007(h) = 1) — Byte 1 = 00(h) Command error code: — Byte 2 = FF(h) — Byte 3 = FF(h) (if command error code = FFFF(h), command is in progress)
	Running "Reset FIFO results" command
	Command finished Command echo: — Byte 0 = 80(h) (X007(h) = 1) — Byte 1 = 00(h) Command error code if the command is correctly carried out: — Byte 2 = 00(h) — Byte 3 = 00(h) OR if an error occurred during the command: — Byte 2 = 80(h) (X017(h) = 1) — Byte 3 = 00(h)
— Wait the end of the command: command echo = 0080(h) (X007(h) = 1) command error code ≠ FFFF(h) (end of command)	
— Deactivate the "Reset FIFO results" command: Write at the address 00(h) the value **0000(h)** Byte 0 = 00(h) (Y007(h) = 0) Byte 1 = 00(h)	

The master instrument must always set to zero the command bit. If it is not done, the slave instrument will not detect the following command on this bit. It has detection on the rising edge (when the bit state goes from 0 to 1).

Last results

Last results structure

At the end of each cycle, the last result is as an array of 12 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.

Standard results

The standard result structure contains **24 bytes**.

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	Flow low section word.	Long	4	x1000
10	Flow high section word.			
11	Flow unit code low part word (refer to. Units table).	Long	4	x1000
12	Flow unit code high part word (refer to. Units table).			

Extended results

The extended results are not yet used in the current version of G6 (V1.002).
They are the same as the standard results.

The extended result structure contains **24 bytes**.

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	Flow low section word.	Long	4	x1000
10	Flow high section word.			
11	Flow unit code low part word (refer to. Units table).	Long	4	x1000
12	Flow unit code high part word (refer to. Units table).			

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

Master	Slave
— Activate the “Read Last result” command: Write at the address 00(h), the value **8000(h)** Byte 0 = 00(h) Byte 1 = 80(h) (Y00F(h) = 1)	
	Acknowledgement Command echo: — Byte 0 = 00(h) — Byte 1 = 80(h) (X00F(h) = 1) Command error code: — Byte 2 = FF(h) — Byte 3 = FF(h) (if command error code = FFFF(h), command is in progress)
	Running “Read Last result” command
	Command finished Command echo: — Byte 0 = 00(h) — Byte 1 = 80(h) (X00F(h) = 1) Command error code if the command is correctly carried out: — Byte 2 = 00(h) — Byte 3 = 00(h) OR if an error occurred during the command: — Byte 2 = 00(h) — Byte 3 = 80(h) (X01F(h) = 1)
— Wait the end of the command: command echo = 8000(h) (X00F(h) = 1) command error code ≠ FFFF(h) (end of command)	
— Deactivate the “Read Last result” command: Write at the address 00(h) the value **0000(h)** Byte 0 = 00(h) Byte 1 = 00(h) (Y00F(h) = 0)	

The master instrument must always set to zero the command bit. If it is not done, the slave instrument will not detect the following command on this bit. It has detection on the rising edge (when the bit state goes from 0 to 1).

Real time

Status and real time measures

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	Meaning	Type	Bytes	Coeff
1	Program number.	Word	2	
2	Number of results waiting in the results FIFO memory.	Word	2	
3	Test type.	Word	2	
4	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: *Unused*. 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	Step code (refer to steps table).	Word	2	
6	Low pressure section word.	Long	4	x1000
7	High pressure section word.			
8	Pressure unit code low part word (see units table).	Long	4	x1000
9	Pressure unit code high part word (see units table).			
10	Flow low section word.	Long	4	x1000
11	Flow high section word.			
12	Flow unit code low part word (refer to. Units table).	Long	4	x1000
13	Flow unit code high part word (refer to. Units table).			

Examples

Pressure value = 207
Pressure: Words 6 and 7
On network:

98	28	03	00

00032898h → 207000(d)/1000 → **207**

Flow value = -0.108
Flow: Words 10 and 11
On network:

94	FF	FF	FF

FFFFFF94h → -108(d)/1000 → **-0.108**

SLMP Command list

- SearchNode
- SetIpAddress
- CompareDeviceInfo
- GetParam
- SetParam
- StartSetParam
- EndSetParam
- CancelSetParam
- ReadStatus
- GetCommSetting
- ReadStatus2

SLMP Error Code

Define	Code (hexa)
SLMP_ERR_COMMAND_SUBCOMMAND	0xC059
SLMP_ERR_WRONG_DATA	0xC05C
SLMP_ERR_DATA_LENGTH	0xC061
SLMP_ERR_UNDER_EXECUTION	0xCEE0
SLMP_ERR_REQ_DATA_SIZE	0xCEE1
SLMP_ERR_RES_DATA_SIZE	0xCEE2
SLMP_ERR_NO_EXIST_SERVER_NO	0xCF10
SLMP_ERR_CAN_NOT_COMMUNICATION_SETTING	0xCF20
SLMP_ERR_NO_EXIST_PARAM_ID	0xCF30
SLMP_ERR_CAN_NOT_PARAMETER_SET	0xCF31

ATEQ parameter SET/GET error codes

Define	Code (hexa)
USER_ERR_PARAMETER_NUM_EXCEED	0x401
USER_ERR_DATA_SIZE_EXCEED	0x402
USER_ERR_PARAMETER_SET	0x403
USER_ERR_PARAMETER_GET	0x404

SLMP ID for parameter SET/GET

Functionalities	R/W	ID	Length	Format	Comment
Parameters	R/W	001 - 499	4 bytes	Signed long, Coeff * 1000	"**Error! Reference source not found.**" section
Extended menu bits	R/W	500	20 bytes	Array of bytes	"**Error! Reference source not found.**" section
Select program	W	501	2 bytes	Unsigned short	
Functions bits	R/W	502	20 bytes	Array of bytes	"**Error! Reference source not found.**" section
Program Name	R/W	503	12 bytes	ASCII Array of 12 char	eg : « 123456789AB »
Number of FIFO results	R	504	2 bytes	short	
Result export frame	R	505	128 bytes max	ASCII frame	
Statistic cycle counter	R	506	16 bytes	(4 * DWORD)	Per program

Export frame

The character "→" corresponds to a tab HT (09(h)).
The character "↵" corresponds to a carriage return CR (0D(h)).

1	→	2	→	3	→	4	→	5	→	6	→	7	→	8	8'	9	→	10	→↵

#	Item
1	Personalization
2	Program number
3	Test result message
4	Test numeric value
5	Test unit
6	Pressure numeric value
7	Pressure unit
8	Alarm message
8'	Bar code
9	Date
10	Hour

Structure of statistic

Words	Meaning	Type	Bytes
1 2	Total number of parts	Unsigned long	4
3 4	Number of good parts	Unsigned long	4
5 6	Number of bad parts	Unsigned long	4
7 8	Number of alarms	Unsigned long	4