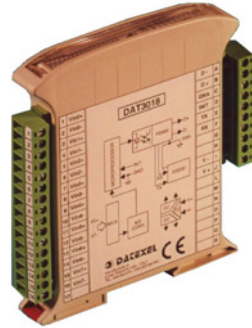


**DATA ACQUISITION  
MODULE****4 / 8 channels  
Analog Input****Characteristics**

Sensor to Computer interface  
for remote data acquisition

4 or 8 analog inputs for:

Tc, mV, mA, RTD, Resistance or Potentiometer

Remotely configurable input signal

RS485 or RS232 line serial communication

Isolated Galvanically at 2000Vac (3 ways)

EMC compatibility - CE Mark

Housing thickness of only 17,5 mm.

DIN rail mounting

**APPLICATIONS**

- Data Acquisition and Controls
- Monitoring of Industrial Process
- Factory and Building Automation
- Distributed Measure and Control

## Table of contents

### 1 - Technical Specifications

1.1 Introduction .....	4
1.2 Pin Assignment & Wiring.....	6
1.3 Specifications .....	15

### 2 - Protocol

2.1 Settings .....	20
2.2 Syntax .....	21
2.3 Tables .....	23
2.4 Commands .....	24

### 3 - Operation Principle

3.1 INIT .....	47
3.2 Dual WatchDog .....	48
3.3 Calibration .....	49

## **Chapter 1**

### **TECHNICAL SPECIFICATIONS**

**1.1 Introduction**

**1.2 Pin Assignment & Wiring**

**1.3 Specifications**

## **1.1 Introduction**

The modules of the DAT 3000 series find, for their peculiar characteristics, wide application in the treatment of the signals in the systems for the Automation and the Control of the Industrial Processes. They allow the conditioning of a wide range of signals, both analogical and digital, with an effective protection from the noises.

Thanks to the systematic employment of the microprocessor the DAT3000 modules are able to assure the conditioning of the signal, the supervision of system, the alarm outputs and the reliable generation of the desired value.

The module has been studied for being able to be assembled on the DIN-rail in simple way and with the maximum use of the spaces. In fact, if the conditions of dissipation allow it, the modules can be assembled one beside the other allowing so a considerable reduction of the space occupied. Extractable type screw terminals are employed for the connection. Thanks to this, the user can directly remove the modules so simplifying their maintenance.

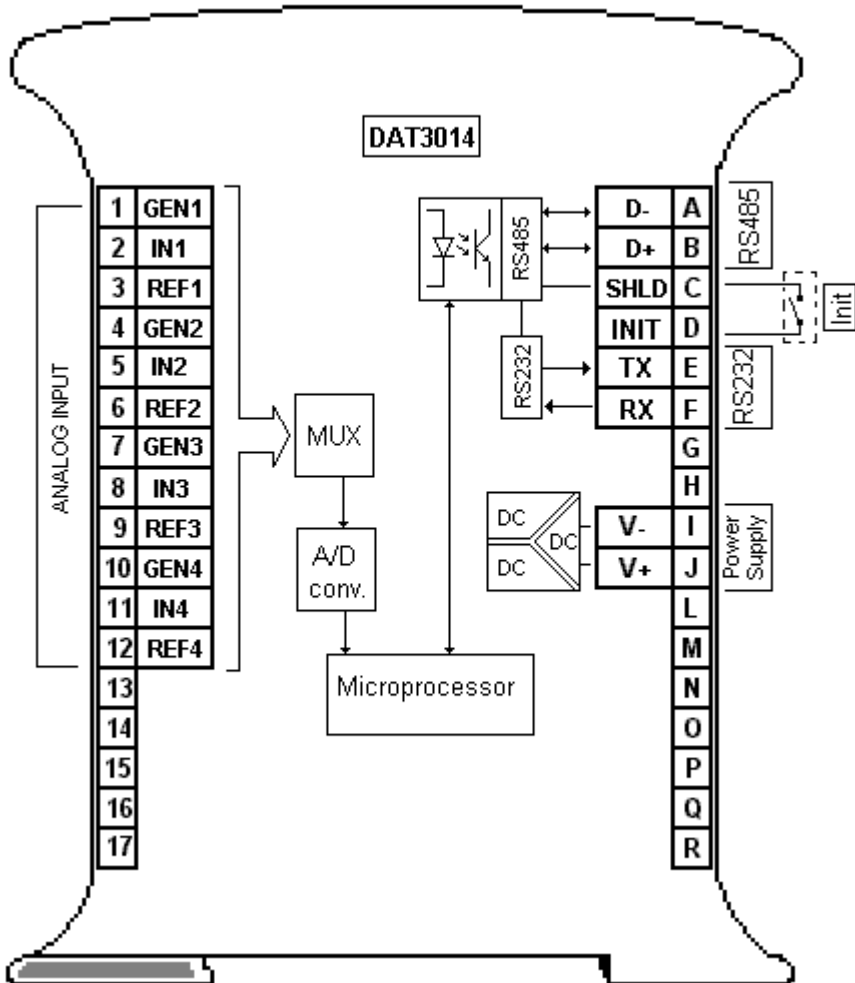
## DAT3014, DAT3016, DAT3018 Characteristics

The DAT3014, DAT3015, DAT3016, DAT3017 and DAT3018 devices converts the analog input signals in engineering units in ASCII code. They are able to handle a great variety of input signals from Tc, mV and mA or from RTD, Resistance and Potentiometer. The 8 channels of the module are galvanically isolated from the power supply and serial line, but not between them.

	<b>Analog Inputs</b>	
DAT3014	4	RTD, Resistance, Potentiometer
DAT3015	4	V or mA
DAT3016	4	TC, mV, mA
DAT3017	8	V or mA
DAT3018	8	TC, mV, mA

- **Watchdog Alarm**
- **Asynchronous serial data transmission**
- **Speed up to 38,4 Kbps warranted**
- **Distance up to 1,2 Km**
- **DIN-rail mounting housing**
- **Complete isolation at 2000 Vac  
with optocouplers and DC/DC converter**

## 1.2.1 DAT3014 Pin Assignment



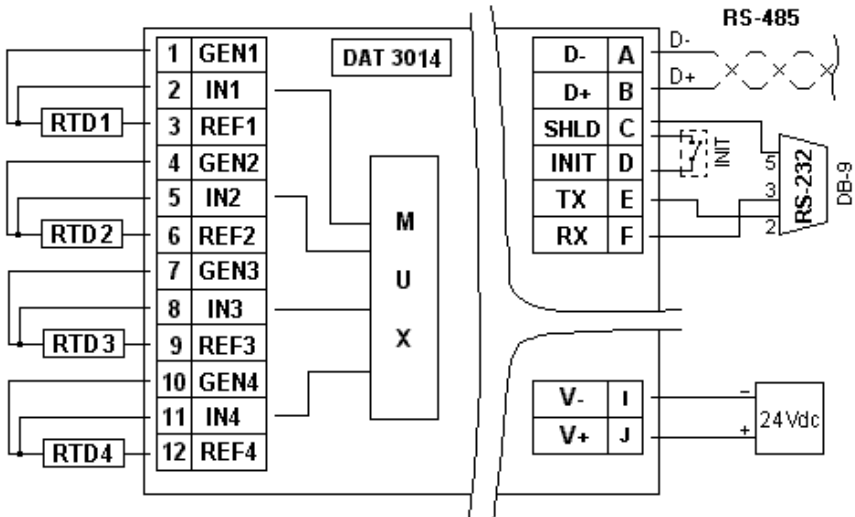
## DAT3014 Pin Assignment Table

PIN	Name	Description
A	D -	D - RS485
B	D+	D + RS485
C	SHLD	RS485/232 Shield INIT Ground
D	INIT	INIT Input
E	TX	RS232 Transmitter
F	RX	RS232 Receiver
I	V -	Power Supply Ground
J	V+	+24V Power Supply

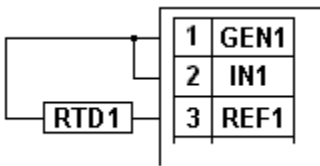
PIN	Name	Description
1	GEN1	Input 1
2	IN1	
3	REF1	
4	GEN2	Input 2
5	IN2	
6	REF2	
7	GEN3	Input 3
8	IN3	
9	REF3	
10	GEN4	Input 4
11	IN4	
12	REF4	

# DAT3014 Wiring

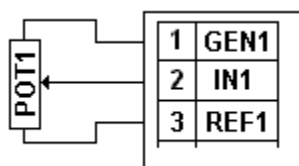
**3-wires RTD & Res. wiring:**



**2-wires RTD & Res. wiring:**

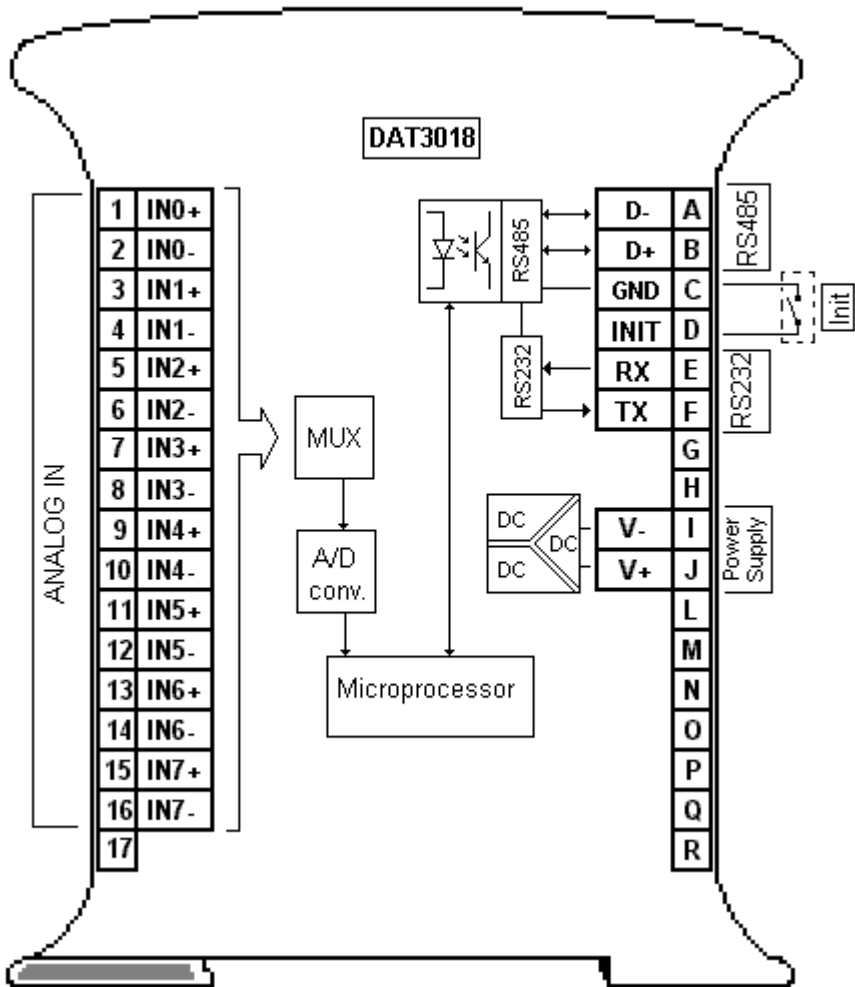


**Potentiometer wiring:**





# 1.2.2 DAT3015 / DAT3016 / DAT3017 / DAT3018 Pin Assignment



# DAT3015 / DAT3016

# DAT3017 / DAT3018

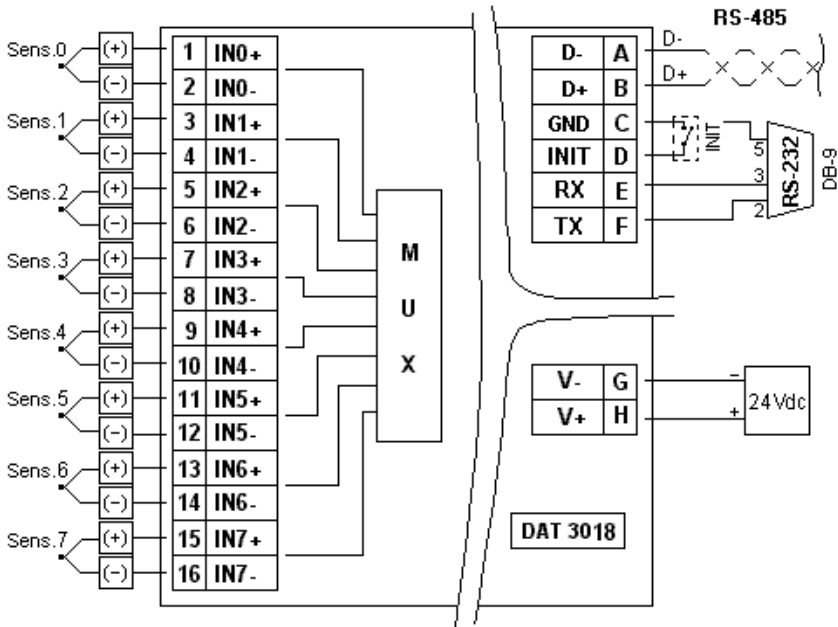
## Pin Assignment Table

PIN	Name	Description
A	D -	D - RS485
B	D+	D + RS485
C	GND	RS485 Shield / Ground
D	INIT	INIT Input
E	RX	RS232 Receiver
F	TX	RS232 Transmitter
I	V -	Power Supply Ground
J	V+	+24V Power Supply

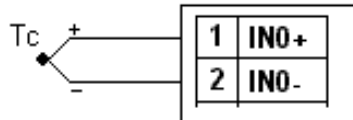
only for  
DAT3017  
and  
DAT3018

PIN	Name	Description
1	IN0+	Input 1
2	IN0 -	
3	IN1+	Input 2
4	IN1 -	
5	IN2+	Input 3
6	IN2 -	
7	IN3+	Input 4
8	IN3 -	
9	IN4+	Input 5
10	IN4 -	
11	IN5+	Input 6
12	IN5 -	
13	IN6+	Input 7
14	IN6 -	
15	IN7+	Input 8
16	IN7 -	

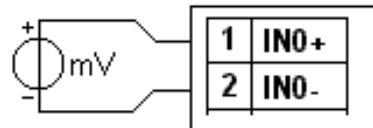
# DAT3016 / DAT3018 Wiring



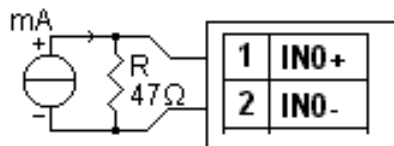
**TC wiring:**



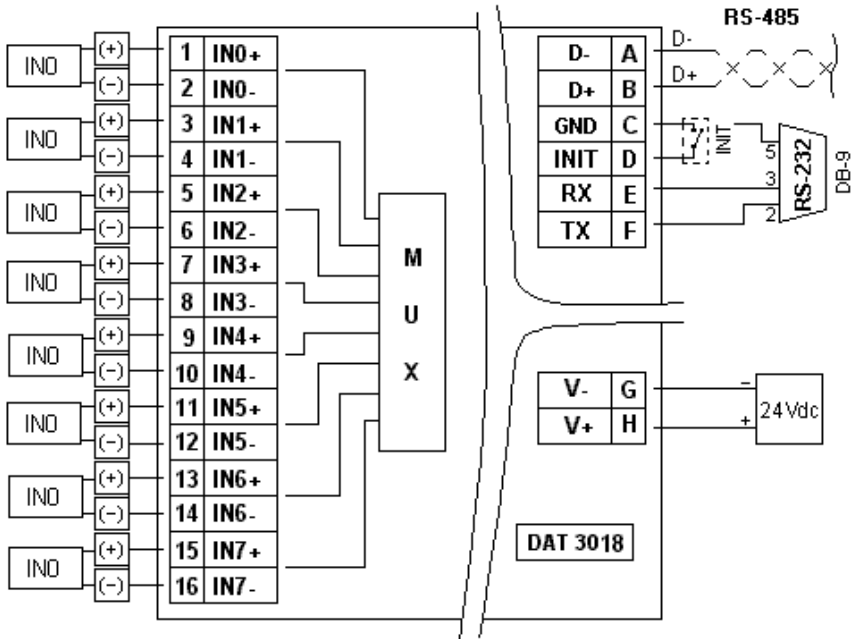
**mV wiring:**



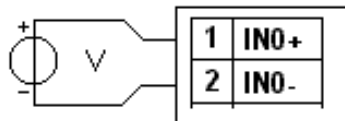
**mA wiring:**



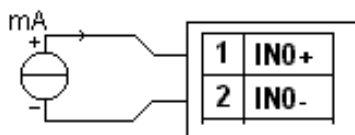
# DAT3015 / DAT3017 Wiring



**V wiring:**



**mA wiring:**



# 1.3 Technical Specifications

(typical @ 25°C and under nominal conditions)

## Analog Input:

Analog input channels of the module are galvanically isolated from power supply and serial line, but not between them.

**DAT3014** = 4 channels

<b>RTD</b>		
Input	Min	Max
PT100-200	0 °C	+850 °C
PT1000	-200 °C	+200 °C
NI100	-80 °C	+180 °C
NI1000	-60 °C	+150 °C

<b>Resistance</b>		
Input	Min	Max
Low	0 Ohm	500 Ohm
High	0 Ohm	2000 Ohm

<b>Potentiometer</b>		
Input	Min	Max
< 500 Ohm	0 %	100 %
< 2000 Ohm	0 %	100 %

**DAT3016** = 4 channels  
**DAT3018** = 8 channels

<b>TC</b>		
Input	Min	Max
J	-210 °C	+1200 °C
K	-210 °C	+1372 °C
R	-50 °C	+1767 °C
S	-50 °C	+1767 °C
B	+400 °C	+1825 °C
E	-210 °C	+1000 °C
T	-210 °C	+400 °C
N	-210 °C	+1300 °C

<b>Voltage</b>		
Input	Min	Max
+/- 50 mV	- 50 mV	+ 50 mV
+/- 100 mV	- 100 mV	+ 100 mV
+/- 500 mV	- 500 mV	+ 500 mV
+/-1000 mV	-1000 mV	+1000 mV

<b>Current *</b>		
Input	Min	Max
+/- 20 mA	- 20 mA	+ 20 mA

\* For measure in mA it is needed to connect a 47 ohm resistor between “+” e “-” pins of all the input channels used.

**DAT3015** = 4 channels  
**DAT3017** = 8 channels

<b>Voltage</b>		
Input	Min	Max
+/- 10 V	- 10 V	+ 10 V

<b>Current</b>		
Input	Min	Max
+/- 20 mA	- 20 mA	+ 20 mA

Sampling time : 0.5 to 2.0 sec.  
 depending on number of channels on query

Rise Time : 0.3 sec.

Input Impedance :  
 TC, mV > 10 Mohm  
 mA 47 ohm (external)

Lead wire resistance influence :  
 TC, mV, V 0.8 uV/ohm

RTD Excitation Current : 0,350 mA typ.  
 Linearity :  $\pm 0.2\%$  F.S. for TC, RTD

Accuracy :

mV, V  $\pm 0.05\%$  F.S.  
 mA  $\pm 0.05\%$  F.S.  
 Tc the larger of  $\pm 0.05\%$  F.S. and  $\pm 5\mu\text{V}$   
 RTD, Res, Pot  $\pm 0.05\%$  F.S.

Cold Junction Comp. :  $\pm 0.5\%$  °C

Deriva Termica

Thermal drift :  
 Full Scale  $\pm 0.005\%$  /°C  
 Cold Junction Comp. :  $\pm 0.02$  °C/°C

**Power Supply:**

Supply voltage : from +10 to +30 Vdc  
 Power Consumption : 1W @ 24Vtyp.  
 Isolating Voltage : 2000 Vac for 60 sec.  
 Polarity protected : 60 Vdc

**Temperature and Humidity**

Ambient Temperature : -10 to +60 °C  
 Store Temperature : -30 to +90 °C  
 Humidity ( not condensing) : 0 ÷ 90 %

**EMC**

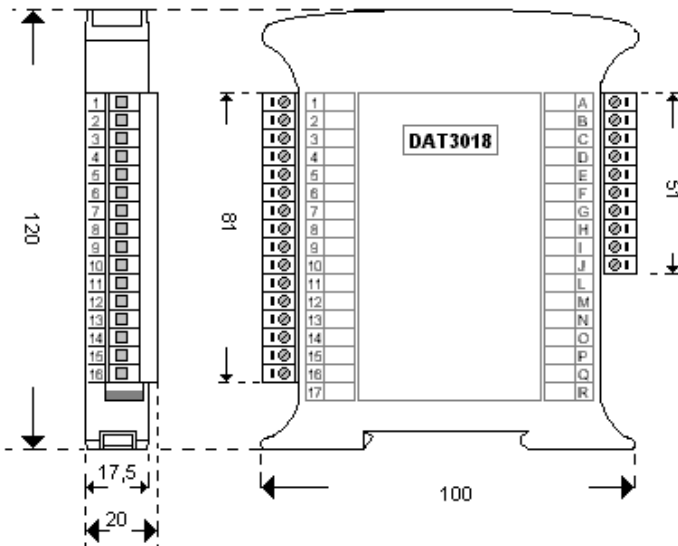
Emission : EN50081-2  
 Immunity : EN50082-2  
 RF Immunity : tested @ 10 V/m up to 1000 Mhz

**Housing**

Material : Selfextinguishing plastic  
 Mounting : DIN rail  
 Weight : Approx. 100 g.

Dimensions :  
 (W x H x T) in mm: 100 X 120 X 17,5

**Mechanical Dimensions**





## **Chapter 2**

# **PROTOCOL**

**2.1 Settings**

**2.2 Syntax**

**2.3 Tables**

**2.4 Commands**

## 2.1 Settings

Default Settings :

**Baud Rate** = **9600 bps**  
**Address** = **00h**  
**Checksum** = **disabled**  
**Data Format** = **ENG**  
**Input Type** = **±50 mV**

Character Format :

**Parity** = **None**  
**Bit Number** = **8**  
**Stop Bit** = **1**

## 2.2 Syntax

### Command Syntax:

The DAT3014, DAT3016 and DAT3018 modules have a set of 21 ASCII instructions for the input control, alarms, watchdog, calibration and device setting. Every command is foregone by a command character and by the address of the module to which it is destined.

For instance, the characters “%”, “#”, “\$”, “~”, are command characters. They are positioned at the beginning of the commands and every symbol indicates a group of commands with different functions.

The address which has to follow these characters (except the commands addressed to all the devices of the net) is pointed out with ‘AA’, and it is composed by two hexadecimal characters, from 00 to FF.

Consequently, the module n°0 will have address AA=00, the module n°255 will have address AA=FF. For instance, the command ‘\$1F2’ reads the setting of the module n°31.

Every module can have any address included between 0 and 255.

Two modules with the same address and the same baud-rate cannot exist.

The commands: “#\* \*\*” and “~ \* \*\*” don’t have address because they are sent to all the modules of the net in the same moment, and they don’t need an answer from the modules. The character ‘\*’ is a command character and it must not be modified.

Every command has to finish with the 13(cr) character.

If the communication foresees the Checksum, it must be positioned before the (cr) character.

## Response Syntax:

When a module receives a string which it recognises as a command addressed to itself, it transmits its same address (AA) as answer. This message is foregone from “!” character if the command is valid or from “?” character if the command is invalid. If the command is valid, (AA) is eventually followed from the requested information (data).

If the module doesn’t answer, this could be due to a communication error, a syntax error or a wrong address.

!AA	Received command is valid
!AA(data)	Sending data to the host
?AA	Received command is not valid

The modules answer to the reading commands regarding the analog input and the cold junction with the character ‘>’ followed from the measuring value and without sending the address.

### ***ATTENTION!***

Writing the commands, like the configuration, the calibration, the name setting etc, involves to write some data on EEPROM. As the component is guaranteed for a limited number of writing cycles (around 100.000) it is advisable to moderate the use of these commands, to avoid future malfunctions of the device. Moreover the writing on EEPROM involves to increase the response time of the module.

## 2.3 Tables

TT : Input Range

TT	Type	Range	Format
01	50 mV	-50 mV ~ +50 mV	+50.000
02	100 mV	-100 mV ~ +100 mV	+100.00
03	500 mV	-500 mV ~ +500 mV	+500.00
04	1000 mV	-1000 mV ~ +1000 mV	+1.0000
06	20 mA	-20 mA ~ +20 mA	+20.000
07	Res.H	0 Ohm ~ 2000 Ohm	+2000.0
08	Res.L	0 Ohm ~ 500 Ohm	+500.0
0E	Tc J	-210 °C ~ +1200 °C	+1200.0
0F	Tc K	-270 °C ~ +1370 °C	+1370.0
10	Tc T	-270 °C ~ +400 °C	+400.0
11	Tc E	-270 °C ~ +1000 °C	+1000.0
12	Tc R	-50 °C ~ +1760 °C	+1760.0
13	Tc S	-50 °C ~ +1760 °C	+1760.0
14	Tc B	+0 °C ~ +1820 °C	+1820.0
15	Tc N	-270 °C ~ +1300 °C	+1300.0
17	Pt100	-200 °C ~ +850 °C	+850.0
18	Ni100	-80 °C ~ +180 °C	+180.0
19	Pt1000	-200 °C ~ +200 °C	+200.0
1A	Ni1000	-60 °C ~ +150 °C	+150.0
1B	Pot.L	0 ~ 500 Ohm	+100.0 (%)
1C	Pot.H	0 ~ 2000 Ohm	+100.0 (%)

CC: Baud Rate

CC	bps
03	1200
04	2400
05	4800
06	9600
07	19200
08	38400

FF : CheckSum +Data Format

7	6	5 4 3 2	10
-	CheckSum	-	Data
0	0=no [chk] 1= [chk]	0	00=Eng 01=%FSR 10=HEX

## 2.4 Command Description

Command	Response	Description
%AANNTTCCFF	!AA	Set Configuration
#**	No Response	Synchr. Sampling
#AA	>(data)	Read all Analog Input
#AAN	>(data)	Read Analog Input N
\$AASPAN	!AA	Span Calibration
\$AAZERO	!AA	Zero Calibration
\$AA2	!AATTCCFF	Read Configuration
\$AA3	!AA(data)	Read CJC Value
\$AA4	!S(data)	Read Synchr. Data
\$AA5VV	!AAS	Enable Channel Mux
\$AA6	!AAVV	Read Mux Status
\$AA9SCCCC	!AA	Set CJC Offset Val.
\$AABN	!AAS	Therm. Open Detect.
\$AAF	!AA(data)	Read Firmware Num
\$AAM	!AA(data)	Read Module Name
~**	Nessuna	Host OK
~AA0	!AASS	Read Module Status
~AA1	!AA	Reset Module Status
~AA2	!AATT	Read Host WDT
~AA3ETT	!AA	Enable Host WDT
~AAO(name)	!AA	Set Module Name

# **%AANNTTCCFF**

## **Set Configuration**

**Description:** Set the configuration of a module

**Syntax:**    %AANNTTCCFF[chk](cr)  
               where:  
               NN = New Address  
               TT = Input Type  
               CC = Baud-Rate  
               FF = CheckSum +Data Format

**Response:**

!AA[chk](cr)	Valid Command
?AA[chk](cr)	Invalid Command
no response	Communication Error or Address Error

**Example:**  
 Command        %00010E0640

Response        !00

Description     The module 00 configuration becomes:  
                   Address = 01  
                   Input = TC E  
                   BaudRate = 9600  
                   Checksum =1  
                   Data Format = Eng

**\*\*\***

## Synchronized Sampling

**Description:** Order all input modules of the net, both analogical and digital, to sample immediately all their input data and to store the data in the internal register, waiting for the command \$AA4 (Read Synchronized Data) coming from the host computer.

**Syntax:** **\*\*\*[chk](cr)**

**Response:**

No response

**Example:**

Command

**\*\*\***

Response

No response

Description

All the modules are sampling

Command

**\$004**

Response

**!1+055.00+032.06...**

Description

Sample Value of the module was: +55°C



# #AA

## Read Analog Input

**Description:** Read the sampled value of all the input channels, processed according to the current configuration.

It is useful to disable the channels that are not used, with the “\$AA5VV” command. If a channel is disabled, its value will not be transmits.

Attention: after having modified the range of input, it is necessary to wait that the measure be stable.

**Syntax:** #AA[chk](cr)

**Response:**

>(data)[chk](cr)	Valid Command
?AA[chk](cr)	Invalid Command
No response	Comm. Error or Address Error

**Example:**

Command        #11

Response

>+000.06+010.00+023.11+015.54+000.06+010.00+023.11+015.54

Description     The inputs of module #11 are:

- IN0 = 0,06 mV
- IN1 = 10,00 mV
- IN2 = 23,11 mV
- IN3 = 15,54 mV
- .....

## #AA

### Read Analog Input

**Description:** Read the sampled value of the channel # N, processed according to the current configuration.

If the channel is disabled, its value will not be transmitted.

Attention: after having modified the range of input, it is necessary to wait that the measure be stable.

**Syntax:** #AA[chk](cr)

where:

N = Channel Number (0..3 for DAT3014, DAT3016)  
(0..7 for DAT3018)

**Response:**

>(data)[chk](cr)	Valid Command
?AA[chk](cr)	Invalid Command
No response	Comm. Error or Address Error

**Example:**

Command #115

Response >+000.06

Description The input # 5 of module #11 is:  
0,06 mV

## \$AASPAN

### Perform SPAN Calibration

**Description:** Perform Span calibration.  
Calibration procedure is described in par. 3.4.

**Syntax:** \$AASPAN[chk](cr)

**Response:**

!AA[chk](cr)	Valid Command
?AA[chk](cr)	Invalid Command
no response	Communication Error or Address Error

**Example:**

Command        \$10SPAN

Response        !10

Description     The module #10 performs Span calibration.

# \$AAZERO

## Perform ZERO Calibration

**Description:** Perform Zero calibration.  
 Calibration procedure is described in par. 3.4.

**Syntax:** \$AAZERO[chk](cr)

**Response:**

!AA[chk](cr)	Valid Command
?AA[chk](cr)	Invalid Command
no response	Communication Error or Address Error

**Example:**

Command            \$10ZERO

Response            !10

Description        The module #10 performs Zero calibration.

# \$AA2

## Read Configuration

**Description:** Read the module configuration. The response contains the address, the input type, the baud-rate and the data format. This command is used for determining the module configuration when it is unknown.

**Syntax:** \$AA2[chk](cr)

**Response:**

- !AATTCFF[chk](cr) Valid Command
- ?AA[chk](cr) Invalid Command
- no response Communication Error or Address Error

where:

TT = Input Type

CC = Baud-Rate

FF = CheckSum + Data Format

**Example:**

Command            \$002

Response            !00110640

Description        The configuration of the module #00 is:  
                       Input= TC E  
                       BaudRate=9600  
                       Checksum=1  
                       Data Format=Eng

## \$AA3

### Read CJC Value

**Description:** Read the sampled value of Cold Junction, express in °C.

Attention: this command is not available when the input is configured for ±20mA or ±10 V.

This command is not available in DAT3014 modules.

**Syntax:**     \$AA3[chk](cr)

**Response:**

>(data)[chk](cr)	Valid Command
?AA[chk](cr)	Invalid Command
no response	Communication Error or Address Error

**Example:**

Command            \$103

Response           >+0024.1

Description        Cold junction temperature is: 24,1°C.

## \$AA4

### Read Synchronized Data

**Description:** Read the data sampled with the ‘ #\*\* ‘ command (synchronized sampling). Read the sampled value come first by “1” , by “0” the following ones.

**Syntax:** \$AA4[chk](cr)

**Response:**

>S(data)[chk](cr)	Valid Command
?AA[chk](cr)	Invalid Command
no response	Communication Error or Address Error

where:

S = 1 (first reading)

S = 0 (following readings)

**Example:**

Command        \$104

Response        >1+050.22

Description     The synchronized data is +50.22 °C. (first reading).

## \$AA5VV

### Enable Channel Multiplexer

**Description:** Enable/Disable the channels of the multiplexer. If the channel is disabled, its value will not be returned during the read (command #AA), while the #AAN command will response with “invalid command” (?AA) .

Each bit of the VV value [ 00..FF ] correspond to an input channel [0..7]; if this bit is 0 the channel is disabled, if this bit is 1 the channel is enabled.

During the reading, it is useful to disable all the not used channels.

Durintg the module calibration, enable only the channel #0 (\$AA501).

**Syntax:** \$AA5VV[chk](cr)

where:

VV = multiplexer value - 2 char. HEX [ 00..FF ]

VV = 51h → 01010001b Enable the channels # 6,4,0

**Response:**

!AA[chk](cr) Valid Command

?AA[chk](cr) Invalid Command

no response Communication Error or Address Error

**Example:**

Command \$10551

Response !10

Description Enable only the channels # 0, 4 and 6.



## \$AA6

### Read Multiplexer Status

**Description:** Read the channel multiplexer value, set with the \$AA5VV command.

**Syntax:** \$AA6[chk](cr)

**Response:**

!AAVV[chk](cr)	Valid Command
?AA[chk](cr)	Invalid Command
no response	Communication Error or Address Error

where:

VV = 2 char. HEX [ 00..FF ]

**Example:**

Command        \$106

Response        !1051

Description     Only the channels 0, 4 and 6 are enabled.

# \$AA9SCCCC

## Set CJC Offset Value

**Description:** Set the value of CJC offset.  
 This command is not available in DAT3014 modules

**Syntax:** \$AA9SCCCC[chk](cr)  
 where:  
 S = value sign (+ o -)  
 CCCC = CJC offset hexadecimal value expressed  
 in tenth of degree ( +0001 = 0,1°C ).

**Response:**

!AA[chk](cr)	Valid Command
?AA[chk](cr)	Invalid Command
no response	Communication Error or Address Error

**Example:**

Command	\$009+0010
Response	!00
Descrizione	CJC offset value become: +1,6°C (10hex = 16dec).

## **\$AABN**

### **Sensor Open Detection**

**Description:** Detect the open sensor on the channel #N.

**Sintassi:** \$AABN[chk](cr)

where:

N = channel number [0..7]

**Response:**

!AAS[chk](cr)	Valid Command
?AA[chk](cr)	Invalid Command
no response	Communication Error or Address Error

where:

S=1 Thermocouple Open

S=0 Thermocouple Closed

**Example:**

Command            \$00B5

Response            !001

Description        Channel 5 , Thermocouple Open

## \$AAF

### Read Firmware Number

**Description:** Read the firmware number of the examined module. This number cannot be modified.

**Syntax:**     \$AAF[chk](cr)

**Response:**

!AA(data)[chk](cr)	Valid Command
?AA[chk](cr)	Invalid Command
no response	Communication Error or Address

Error

**Example:**

Command         \$11F

Response        !11C001

Description     The module #11 has the C001 firmware number.

## \$AAM

### Read Module Name

**Description:** Read the name of the examined module.

The module name can be modified with the @AAO(name) command (Set Module Name ).

**Syntax:**     \$AAM[chk](cr)

**Response:**

!AA(name)[chk](cr)	Valid Command
?AA[chk](cr)	Invalid Command
no response	Communication Error or Address Error

where:

(name)= module name (4 characters max.)

**Example:**

Command            \$00M

Response           !003018

Description        Module name is: “3018”.

~\*\*

## Host OK

**Description:** This command is used to order to all the modules of the net to reset the watchdog timer.

By this command it is avoided that the modules, when the watchdog timer is active, goes in alarm condition.

**Syntax:** ~\*\*[chk](cr)

**Response:** No response

### Example

Command           ~\*\*

Response           No response

Description        All the modules reset their watchdog timer.

## ~AA0

### Read Module Status

**Description:** Read the module status.

If the status is 04 it means that the Host WatchDog is in alarm condition. The status remains in this condition until the command ~ AA1 (reset) is not sent.

**Syntax:** ~AA0[chk](cr)

**Response:**

!AASS[chk](cr)	Valid Command
?AA[chk](cr)	Invalid Command
no response	Communication Error or Address Error

where:

SS=00	The module is in normal condition
SS=04	The module is in alarm condition

**Example:**

Command           ~020

Response           !0204

Description        The module is in alarm condition.

## ~AA1

### Reset Module Status

**Description:** Every time that the module is in WatchDog Host Alarm condition, its state assumes the value “04”. By this way it is possible to know if the module is in alarm condition. The state remains in this condition until the command ~ AA1 (reset) is not sent.

**Syntax:** ~AA1[chk](cr)

**Response:**

!AA[chk](cr)	Valid Command
?AA[chk](cr)	Invalid Command
no response	Communication Error or Address Error

**Example:**

Command ~021

Response !02

Description The module is not in alarm



## ~AA2

### Read WatchDog Timer

**Description:** Read the Host Watchdog timer value, set by the ~AA3ETT (Enable WDT Timer) command.

**Syntax:** ~AA2[chk](cr)

**Response:**

!AASTT[chk](cr)	Valid Command
?AA[chk](cr)	Invalid Command
no response	Communication Error or Address Error

where:

S = 0 WDT disabled

S = 1 WDT enabled

TT = timer value in tenth of second in HEX  
(0001=0,1s)

**Example:**

Command           ~012

Response           !01130

Description        WDT enabled,  
                    Timer value:   30h → 48d  
                                      48\*0,1= 4,8s

## ~AA3ETT

### Enable Watchdog Timer

**Description:** Enable/Disable the host WatchDog timer. When the host WDT is enabled, the host must to send the ~\*\* command to all modules before the timer ends. When the ~\*\* command is received, the timer of the host watchdog is reset and than it restarts. You can use ~AA2 command to read value and status of the host watchdog timer.

**Syntax:** ~AA3ETT[chk](cr)

where:

E = 0 Timer disabled

E = 1 Timer enabled

TT = timer value in tenth of second in HEX  
(0001=0,1s)

**Response:**

!AA[chk](cr)	Valid Command
?AA[chk](cr)	Invalid Command
no response	Communication Error or Address Error

**Example**

Command        ~013130

Response        !01

Description     WDT enabled.  
                   Timer:     30h —> 48d  
    48\*0,1= 4,8s

## ~AAO(name) Set Module Name

**Description:** Set the module name.

The name must be composed by 4 ASCII characters.

You can use the \$AAM command to read the module name.

**Syntax:** ~AAO(name)[chk](cr)

where:

(name)= module name (4 characters max.)

**Response:**

!AA[chk](cr)	Valid Command
?AA[chk](cr)	Invalid Command
no response	Communication Error or Address Error

**Example:**

Command        ~01O3018

Response        !01

Description     The new module name is: “3018”

## **Chapter 3**

# **OPERATION PRINCIPLES**

**3.1 INIT**

**3.2 Dual WatchDog**

**3.3 Calibration**

## 3.1 INIT

If the exact configuration of a module is unknown, it can result impossible to establish a communication with it. The INIT pin resolves this problem.

Connecting the INIT pin to the GND pin (ground) and turning on the device, it will automatically goes in the default configuration:

```
address = 00  
baud-rate = 9600  
checksum = disabled  
data format = ENG
```

The user can send the “read configuration” command to the module and it will answer supplying the configuration into the memory; by this way the user knows the exact configuration of the module. If the INIT pin is disconnected from GND, at the next power-up the module automatically returns to the original configuration, and the connection can restart.

## 3.2 Dual WatchDog

The DAT3000 modules has been designed for working in industrial environments, where many noises or energy transient can be present. If these noises are too much strong, the modules could suffer of them and working incorrectly, or being not able anymore to communicate with the host. This situation can be very dangerous in the case that the module commands some attuator in a not correctly way. For this reason the modules has provided of a Host Watchdog timer which, when it is enabled, makes to start the alarm every time that the communication between the module and the host is inactive for a time greater then the programmed one. When the alarm goes on, the values of the outputs automatically are converted in the values set as ‘safety value’, corresponding to the state in which must be putted the outputs, and therefore the actuators, to avoid damages to the system in case of failure.

There is also a Module Watchdog timer, a hardware circuit that watches the operation of the internal CPU and start when, expired the limit of time, reset doesn’t come from the same CPU. If the CPU breaks for some reason and no reset of the Module Watchdog happens within the preset time, it starts and resets the CPU, making to restart again its program. After the reset of the module, the digital outputs will assume their initial value (default), which may not to be identical to the value of the outputs before the reset. For this reason the user, after the reset of the module, has to resend the output setting command.

The user can know if the module is reset sending the \$AA5 command, whose response contains the S value.

If S=1, this means that the module is reset.

## 3.3 Calibration

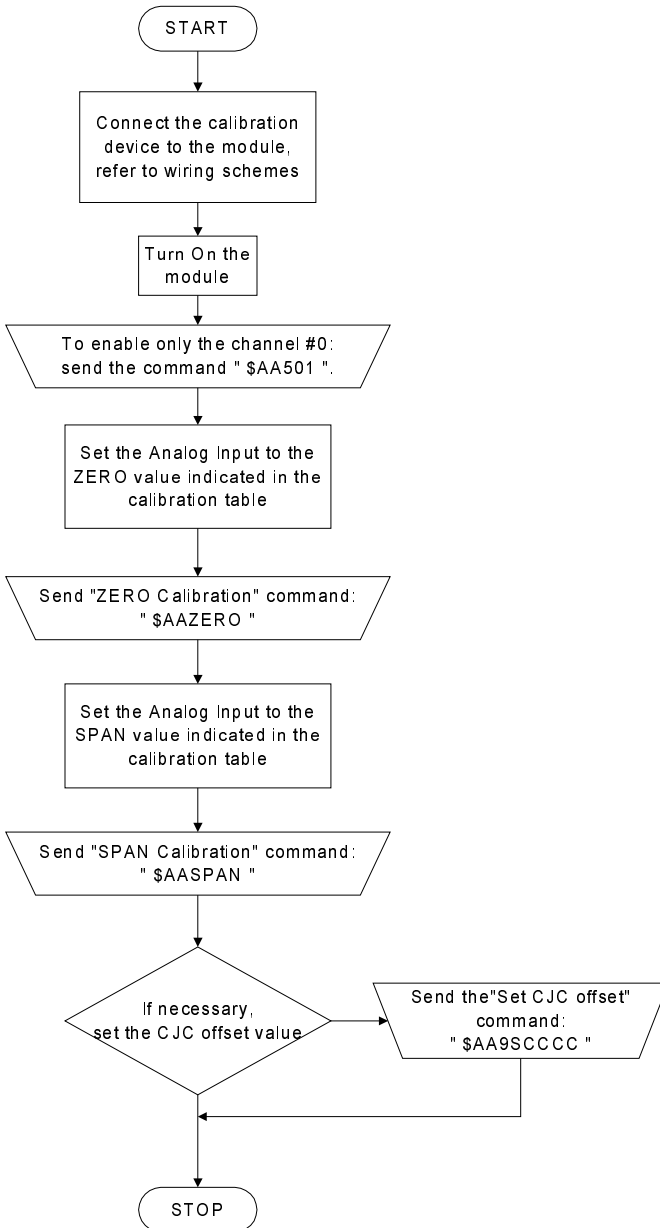
The procedure of calibration is performed in factory on all the modules during the testing phase. However it can be convenient to make another calibration of the module according to the requirements of the user.

To make this, it is necessary to use precision instruments and to correctly perform all the necessary steps, because any error reduces the accuracy and the good operation of the device.

Furthermore this operation must be carried out following the procedure and the data illustrated in the following pages.

If a device able to simulate the sensor (calibrator) is not available, it is possible to replace it by applying the corresponding value in mV (for the Tc) or in Ohm (for the RTD).

The calibration signal must be connected to the channel # 0





**Calibration Table:**

TT	Type	ZERO	SPAN
01	50 mV	+0.000 mV	+25.000 mV
02	100 mV	+0.000 mV	+100.00 mV
03	500 mV	+0.000 mV	+500.00 mV
04	1000 mV	+0.000 mV	+1000.0 mV
06	20 mA	+0.000 mA	+20.000 mA
07	Res.H	0.0 Ohm	2000.0 Ohm
08	Res.L	0.0 Ohm	500 Ohm
0E	Tc J	0.0 °C 0.000 mV	+1200.0 °C 69.536 mV
0F	Tc K	0.0 °C 0.000 mV	+1370.0 °C 54.807 mV
10	Tc T	0.0 °C 0.000 mV	+400.0 °C 20.869 mV
11	Tc E	0.0 °C 0.000 mV	+1000.0 °C 76.358 mV
12	Tc R	0.0 °C 0.000 mV	+1760.0 °C 21.006 mV
13	Tc S	0.0 °C 0.000 mV	+1760.0 °C 18.612 mV
14	Tc B	0.0 °C 0.000 mV	+1820.0 °C 13.814 mV
15	Tc N	0.0 °C 0.000 mV	+1300.0 °C 47.502 mV
17	Pt100	0.0 °C 100.0 Ohm	+850.0 °C 390.26 Ohm
18	Ni100	0.0 °C 100.0 Ohm	+180.0 °C 223.22 Ohm
19	Pt1000	0.0 °C 1000.0 Ohm	+200.0 °C 1785.3 Ohm
1A	Ni1000	0.0 °C 1000.0 Ohm	+150 °C 1986.6 Ohm
1B	Pot.L	0.0 %	100.0 (%)
1C	Pot.H	0.0 %	100.0 (%)

DAT3014 RTD Remote Analog Input Module to RS232

Phone: +1 561 779 5660 E-mail : [Info@datexel.com](mailto:Info@datexel.com) – Web Site [www.datexel.com](http://www.datexel.com)