
# *MACMAT - Driver of GAZ_MODEM Protocol for MACMAT Station User's Manual*

*Doc. No. ENP4030*
*Version: 29-08-2005*

# 1. MACMAT - Driver of GAZ_MODEM Protocol for MACMAT Station

## 1.1. Driver Use

The MACMAT driver is used for communication with the MACMAT station. The driver supports the stations signed as Korektor Impulsowy (Impulse Corrector) 01723 CMK 01 97/01/02 manufactured by COMMON Ltd. and PKNMiJ 03-03-93 RP T-Zw5-1 manufactured by the PLUM company.

## 1.2. Declaration of Transmission Channel

The logical channel is defined by placing an appropriate item in the [ASMEN] section of the application INI file. The syntax of declaration of transmission channel operating according to the COMLI protocol is given below:

*logical_name*=MACMAT,*address*,*COMn*

where:

| | |
|---|---|
| *n* | - number of the serial port to which the network of MACMAT stations is connected; |
| *address* | - station address. |

## 1.3. Driver Configuration

Each defined channel may have its own section, name of which is its logical name i.e. *[logical_name]*. The COMn port may also have its own section named **[MACMAT:n]**. Values defined in such section become default values for all stations connected to a given port. If in the INI file a section named **[MACMAT]** is placed, then the values placed in such ssection become default values for all stations supported by the controller. Values placed in the section of a given station (*[logical_name]*) have a priority before values placed in the section of a given serial port and the last ones have a priority before values placed in the [MACMAT] section. If the parameter is not present in any section, then its default value is taken according to the description below. In particular, the initialization file may not include any sections parameterizing stations. Appropriate records are required only in the [ASMEN] section.

Parameters of transmission with use of a serial interface cannot be placed in sections concerning individual stations.

***Auto_sync=number***

| | |
|---|---|
| Meaning | - if the parameter is different from 0, then automatic synchronization of the computer clock with the MACMAT station clock. The parameter value determines the minimal time between sequent comparisons of station and computer clocks. A comparison of clocks is performed only during other data reading from the station. |
| Default value | - 3600 (1 hour). |

Parameter:
*number* - time in seconds.

***Alt_port = COMm, metod_of_switching_to_alternative_port, metod_of_switching_to_basic_port***

Meaning - *(See: Definition of alternative ports).*

***AsComm =Yes/No***

Meaning - if yes is given, then the driver will use AsComm Connection Manager to establish connections with MACMAT stations.
Default value - No.

***No_Errors =Yes/No***

Meaning - if yes is given, then the driver will not output the messages on errors in lines and timeout.
Default value - No.

***Max_History_Buffers=number***

Meaning - determines the maximal number of buffers containing historical data, read for needs of the archiving module. One buffer includes historical data from one time interval for one variable. It is stored in the memory within the time specified by the parameter *Max_History_Buffers*. One buffer occupies about 400 bytes of memory and can contain 50 values. If archive data are saved by the station every 15 minutes thus, for 24 hours, 2 buffers for one variable are necessary. Historical buffers are used by the data archiving program ASPAD during completing a B type archive. After the time determined by the parameter *History_Buffer_Removal* buffers are removed from the memory.
Default value - 5000.
Parameter:
*number* - number of buffers.

***Max_history =number***

Meaning - determines a time period in days, counted from the current moment backwards, for which historical data, stored in the station memory, will be read.
Default value - 35.
Parameter:
*number* - time in days.

***Max_Time_Difference =number***

Meaning - the maximal time difference between indications of a station clock and a computer clock, after exceeding of which synchronization of clocks occurs. The parameter has a meaning only when the *Auto_sync* parameter is different from zero.

Default value - 60.

Parameter:

*number* - time in seconds.

***Status_Mask =number***

Meaning - a number determining which values of the variable status cause non-validity of the variable value. The variable status is read from the MACMAT station together with its value. This status is a bitmap; meaning of bits is described in the station documentation. The driver executes the logical operation AND on the variable status received from the controller and on the value of the parameter *Status_Mask*. If the result of this operation is different from zero, then the data value is invalid. The data value is invalid too if the variable status has a value of 0 (i.e. lack of data).

Default value - the default, value of 6 means that values, exceeding the measuring range, are non-validated. As the parameter value, an integer, whose individual bits correspond to appropriate bits of the status, should be given.

Parameter:

*number* - time in seconds.

***Counter_Ratio =number***

Meaning - gas flow counter is transmitted in the form of two floating-point numbers *Vn0* and *Vn1*.

Protocol specification states that the counter value is calculated according to the formula:

Vn0 + Vn1*10000

However, some stations use the formula:

Vn0 + Vn1*100000

Parameter defines the value, by which the quantity *Vn1* should be multiplied:

Vn0 + Vn1*Counter_Ratio.

Default value - 10000.

***P_Ratio =number***

Meaning - according to the MacMAT station specification, the pressure is expressed in kPa. However, some stations transfer the pressure

expressed in MPa. It concerns only archive (register) data. The parameter determines, by what factor the pressure, transferred by the station, should be multiplied.

Default value - 1000.

***baud =number***

***bps=number***

Meaning - transmission speed.
Default value - 9600.
Parameter:
*number* - value passed in Bd.

***parity =parity_parameter***

Meaning - item determines a parity type.
Default value - n.
Parameter:
*number* - parity type:
n – no parity but,
o – odd parity check,
e – even parity check,
m – mark,
s – space.

***retries =number***

***retry=number***

Meaning - number of transmission repetitions in case of transmission errors.
Default value - 5.

***word =number***

***word_length =number***

Meaning - word length.
Default value - 8.
Parameter:
*number* - number from the range from 5 to 8 bits.

***time-out =number***

***timeout =number***

Meaning - waiting time for station answer in seconds.
Default value - 2.

***History_Buffer_Removal =number***

Meaning - parameter determines time after which buffers containing historical data, read for needs of the archiving module, are removed.
Default value - 30.
Parameter:
*number* - the time is given in minutes.

***AllErrors =yes/no***

Meaning - if the parameter has a value of no, then the information on *timeout* errors will appear in *'Control Panel'* only when the transmission missed in spite of attempts of its repetition. If it has a value of yes, then the information on all errors is transmitted to *'Control Panel'*.
Default value - no.

***RTS =yes/no***

Meaning - if yes is given, then data sending to the station will occur by means of the RTS line set on high state and the reception on low state.
Default value - no.

***RTS_OFF_Delay =yes/no***

Meaning - time after which the RTS line will be zeroed after having sent data to the station. The parameter has meaning only when the control by means of the RTS line is switched on.
Default value - 10.
Parameter:
*number* - time in milliseconds.

***Ignore_Source_Address =yes/no***

Meaning - each packet sent by the station includes the station address. The station address is verified by the controller. In case of incompatibility to the station number it is rejected. Setting yes will cause the controller to stop the sender address verification.
Default value - no.

**EXAMPLE 1**

```
[ASMEN]
.....
MAC=MACMAT,2,COM2
....
```

```
[MAC]
Auto_Sync=60
Max_time_difference=10
```

In the example above the station named MAC connected to the COM2 port is defined. The synchronization of computer and station clocks will be performed every 1 minute. If the difference is at least 10 seconds, then the synchronization of clocks occurs.

**EXAMPLE 2**

```
[ASMEN]
.....
MAC1=MACMAT,1,COM2
MAC2=MACMAT,2,COM2
MAC3=MACMAT,3,COM2
MAC4=MACMAT,4,COM3
MAC5=MACMAT,5,COM3
MAC6=MACMAT,6,COM4
....


[MACMAT]
;Default value for all stations
baud=19200


[MACMAT:3]
;Default values for stations connected to the COM3 port
baud=9600


[MAC6]
Auto_Sync=0
```

In the example above stations with names from MAC1 to MAC6 are defined. The stations MAC1, MAC2 and MAC3 are connected to the COM2 port. The stations MAC4 and MAC5 are connected to the COM3 port. The station MAC6 is connected to the COM4 port. All serial ports except COM3 will work with a speed of 19200 baud. The COM3 port will work with a speed of 9600 baud. The clock of the station MAC6 will not be synchronized.

## 1.4. Defining the Process Variables

Current Measure Data

Variables allowing to access to current measure data have the following form:

B*n*

Where:

*n* - is the number of a datum according to the station specification:

B1 - value of a gas flow counter
B2 - Qn
B3 - Qr
..... etc.

The value of Bn variable is a floating-point number.

The variable B0 is not used.

Access to Registered Values

For measures marked according to the specification with numbers 0/1 and from 2 to 8, the access to their values remembered by the station as registered data or twenty-four-hours data (for a flow counter) is possible. The value of a registered variable is the value remembered by the station in the last registration period. An access to older measures is possible by the B type archiving. The registered variables have the following form:

R0 - gas flow counter (in the end of last twenty-four hours)
R2 - Qn
R3 - Qr
. .
R8 - rez2

Access to the list of alarms by using the number of the successive alarm in the list:

Variables allowing to access to the list of alarms have the following form:

A*n.type*

where:

| | |
|---|---|
| *n* | - alarm no. |
| *type* | - type of information on an alarm according to the table below. |

***Table 1. Type of Information on an Alarm.***

| Type Name | Meaning | Type of Received Value |
|---|---|---|
| c, code | alarm code (according to the station documentation) | integer number (1 byte) |
| v, val | increase of counter value in time of alarm duration | floating-point number (4 bytes) |
| sec0 | the second of alarm beginning | integer number (1 byte) |
| m0, min0 | the minute of alarm beginning | integer number (1 byte) |
| h0, hour0 | the hour of alarm beginning | integer number (1 byte) |
| day0 | the day of alarm beginning | integer number (1 byte) |
| mon0 | the month of alarm beginning | integer number (1 byte) |
| y0,year0 | the year of alarm beginning | integer number (2 bytes) |
| sec1 | the second of alarm end | integer number (1 byte) |
| m1, min1 | the minute of alarm end | integer number (1 byte) |
| h1, hour1 | the hour of alarm end | integer number (1 byte) |
| day1 | the day of alarm end | integer number (1 byte) |
| mon1 | the month of alarm end | integer number (1 byte) |
| y1,year1 | the year of end alarm | integer number (2 bytes) |

Access to the list of alarms by means of a code of alarms:

Variables allowing to access to the list of alarms by means of an alarm code have the following form:

E*n* or E*n.type*

where:

*n* - alarm code according to the documentation;

*type* - type of information about the alarm according to the table presented above.

Variable E allows to access to the information about the alarm with a given code. If the list of alarms does not include the code of a required alarm, then a value of 0 (integer type – 1 byte) is returned. If the list contains many alarms with a given code, then the information about the alarm which occurred as latest of all is returned. If the variable type was omitted, then a value of 1 is returned if the alarm with a given code is active, and a value of 0 otherwise. If the variable type is given, then the value from the table presented above is returned.

Access to the List of Alarms as Bit Mask

The variable has the form:

EB*n*

where:

*n* - number of byte 0-31.

By means of the EB variable it is possible to read information about active alarms in groups of 8 alarms:

EB0 - alarms with codes 0- 7
EB1 - alarms with codes 8-15
EB2 - alarms with codes 16-23
...
EB31 - alarms with codes 248-255

The variable value is an integer number of 1 byte length. Individual bytes of a variable value are assigned to adequate alarms. If the bit is set, then the alarm corresponding with it is active. The EB type variable allows to bind the alarms with bit strategy of identifying the alarms of **asix**.

Access to Twenty-Four-Hours Data

The variable has the form:

D*n*

where:

*n* - is a number of data in accordance to the station documentation.

The D0 variable has the same meaning as the R0 variable.

Access to Statistical Data

The MACMAT controller enables to access to statistical data referring to the quantity of transmitted data and quantity of errors of the transmission. Variables allowing to access to statistical data have the following form (see: **Table *2***).

***Table 2. Variables Allowing to Access to Statistical Data.***

| Address | Meaning | Type of Received Value |
|---|---|---|
| SBS | quantity of sent bytes | Integer number (4 bytes) |
| SBR | quantity of sent bytes | Integer number (4 bytes) |
| SFS | quantity of sent frames | Integer number (4 bytes) |
| SFR | quantity of sent frames | Integer number (4 bytes) |
| SPE | quantity of errors of parity | Integer number (4 bytes) |
| SFE | (frame errors) quantity of frame errors | Integer number (4 bytes) |
| SOE | quantity of errors of overrun | Integer number (4 bytes) |
| SLE | quantity of line errors (the amount of parity errors, frames, overrun and etc.) | Integer number (4 bytes) |
| STE | quantity of timeout errors | Integer number (4 bytes) |
| SPRE | quantity of protocol errors | Integer number (4 bytes) |
| SCE | quantity of checksum errors | Integer number (4 bytes) |
| SLGE | quantity of logical errors (lack of data in the controller, incorrect address etc.) | Integer number (4 bytes) |
| SERR | amount of all errors (SLE, STE, SPRE, SCE and SLGE). The saving of any value into ERR variable is causes zeroing the following variables: SBS, SBR, SFS, SFR, SPE, SFE, SOE, SLE, STE, SPRE, SCE and SLGE. | Integer number (4 bytes) |
| TSBS | quantity of sent bytes (from the beginning of driver working) | Integer number (4 bytes) |
| TSBR | quantity of received bytes (from the beginning of driver working) | Integer number (4 bytes) |
| TSFS | quantity of sent frames (from the beginning of driver working) | Integer number (4 bytes) |
| TSFR | quantity of received frames (from the beginning of driver working) | Integer number (4 bytes) |
| TSPE | quantity of errors of the parity (from the beginning of driver working) | Integer number (4 bytes) |
| TSFE | quantity of the frame errors (from the beginning of driver working) | Integer number (4 bytes) |
| TSOE | quantity of errors of overrun (from the beginning of driver working) | Integer number (4 bytes) |
| TSLE | quantity of errors of the line (the amount of errors of parity, frames, overrun and etc.) | Integer number (4 bytes) |
| TSTE | quantity of timeout errors (from the beginning of driver working) | Integer number (4 bytes) |
| TSPRE | quantity of protocol errors (from the beginning of driver working) | Integer number (4 bytes) |
| TSCE | quantity of checksum errors (from the beginning of driver working) | Integer number (4 bytes) |
| TERR | amount of errors determined with following variables: TSLE, TSTE, TSPRE, TSCE, TSOE | Integer number (4 bytes) |

Access to Historical Data (for the Version with Access to Historical Data)

The MACMAT controller enables the archive ASPAD module to access to historical data for variables from B1 to B8 and R0 and from R2 to R8:
- for variables R0 and B1, daily data are read;
- for variables B2 and B3, registered data of flow increment are read.; these data are scaled so that they express a flow per 1 hour; the data are scaled in the basis of registration frequency read from the station;
- for variable B4 to B8 and R2 to R8, adequate registered historical data are read.

## 1.5. Cooperation with AsComm Manager of Connections

In order to use the AsComm manager of connections to establish connections with MACMAT stations you should place the following record in the application INI file:

*AsComm = Yes*

This record is placed in the [MACMAT] or [MACMAT:n] section, where *n* signifies the number of the serial port declared in the [ASMEN] section.

In case of cooperation with the AsComm module, the port number from the channel declaration is used to create the name, which is used by the driver to exchange data with the AsComm module. The name has a form of *MacMAT-n*, where *n* is the number of the serial port from the channel declaration.

**EXAMPLE**

```
[ASMEN]
.....
MAC=MACMAT,2,COM2
....
[MACMAT]
.....
AsComm = Yes
....
```

The example above defines driver of the name MacMAT-2 as a client of AsComm. This name is also the name of the section, in which parameters of connections established by the AsComm module such as modem name, telephone number, etc. are placed. The description of parameters, which should be placed in such section, may be found in the manual of the AsComm module. You should notice that the number of the serial port may refer, but it has not, to the physical serial port. This number signifies the real serial port only when in the section (named [MacMAT-n]) for parameterization of connections established by the AsComm module other records defining truly used port (e.g. modem name) are not given.

An example of configuring the AsComm module for dial-up connections is given below:

```
[ASMEN]
.....
MAC=MACMAT,2,COM2
....
[MACMAT]
```

```
.....
AsComm = Yes
....

[MACMAT-2]
switched_line = Yes
Modem =Sportster Flash
Interval = 5m
Max_Connection_Time=2m
Number = 12345678
```

In order to establish connections the modem Sportster Flash will be used. Connections will be established every 5 minutes with the number 12345678. Maximal connection duration time is equal to 2 minutes.

## 1.6. Definition of Alternative Ports

The MACMAT driver allows using alternative serial ports in case of communication problems occurring when using the basic port (that appears in the definition of the logical channel). The parameter declaring the alternative port may be inserted into the [MacMAT] or [MacMAT:n] section and has the following form:

*Alt_port = COMm, metod_of_switching_to_alternative_port, metod_of_switching_to_basic_port*

The parametr defines the COMm serial port used in case of communication problems in the COMn port (determined in the logical channel definition). The parameter may occur only one time (only one alternative port is permissible). The COMm alternative port is not allowed to be declared as the basic port in the other ASMEN channel definition.

Switching to the alternative port occurs after fulfilling the following condition:

*Errors_number*[*/time_period*]

When a number of missed reading attempts determined by *Errors_number* occurs in *time_period*, switching to the alternative port follows. The quantity of errors also includes the quantity of the transmission operation repetitions performed by the driver. That means, that if the occurrence of 3 errors is the condition of switching and the number of repetitions is 5, switching may occur during the time of current request performing and this request has a chance to be performed correctly through using the alternative channel. If not, the request will be performed with an error status and the switching to the alternative port will occur during performing next ASMEN requests. The *time_period* parameter may be omitted - in such case, the switching will occur after *Errors_number* appearance. The *time_period* parameter is passed in seconds.

The return to using the basic port occurs after a number of seconds determined by the *metod_of_switching_to_basic_port* parameter since the moment of switching to the alternative channel. It doesn't mean that the modem connection will be realized all the time (if the alternative channel is that connection). This connection will be served like so far, it means it will be disconnected as a result of the AsComm module parameterization or when all ASMEN requests are realized. The parameters of the serial port should be defined in the [MacMAT] section if they are supposed to be other than the ones applied in case of the basic port.

**EXAMPLE**

An example of the alternative channel parametrization:

```
[ASMEN]
...
MAC1=MACMAT,220,COM1
...
[MACMAT]
Baud = 9600
;switching to the alternative channel, after 3 following errors have occurred, and return switching after
;2 minutes
Alt_Port = COM2, 3, 120
Alt_Port = COM2, 15/60, 120
; Or switching to the alternative port after 15 errors having occurred during 1 minute. The return switching as above.
```

# 2. List of Tables