# Manual

for
**PLC-Mem**
mass storage device
with Millenium-Interface


| | |
|---|---|
| **Version:** | **0.2** |
| **Date:** | **31.12.2009** |
| **Update:** | **23.04.10 14:40** |
| **Author:** | **Dr. Martin Burger, Edmund Burger** |
| **Company:** | **MOTRON Steuersysteme GmbH** |
| **Customer:** | |
| **File:** | **s. foot bar** |


**MOTRON Steuersysteme GmbH**
**Im Gewerbegebiet 6**
**91093 Heßdorf**
**Tel.: 09135/73 88 -0**
**Fax.: 09135/73 88 37**
**e-Mail: motron@t-online.de**
**Internet: www.motron.de**

# Index

# Glossary

| Term | Abbrev. | Description |
|---|---|---|
| USB stick | stick | Mass storage with file system and USB interface |
| Millenium | Mm | Mini PLC from Crouzet |
| Module | PLC-Mem | Mass storage device |
| Data block | DB | 2-byte memory area in the Mm including transfer data |
| File format | CSV | Excel-kompatible, Semicolon-separator |

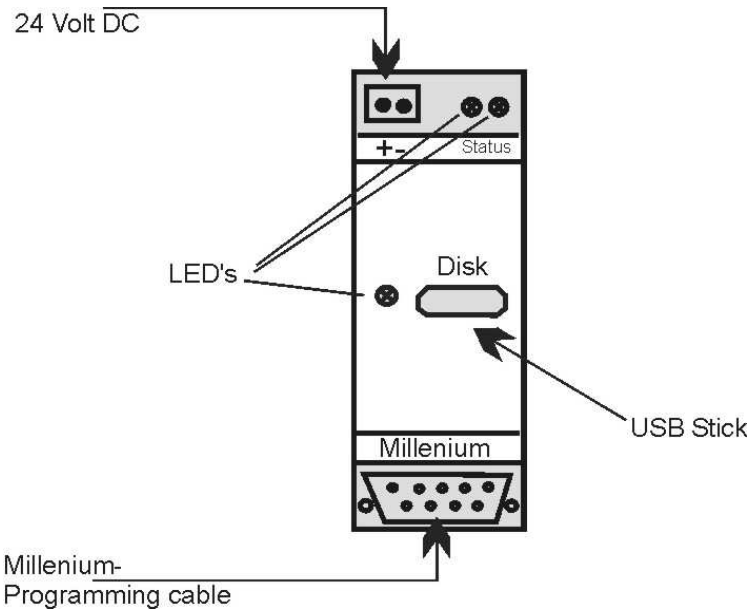
## 1. General

On the one hand, with ***PLC-Mem*** it is possible to log data of the Millenium-PLC (***Mm***) in masses and over a long period of time by means of an USB device. Afterwards, the USB device can be carried to a PC and data can be analyzed offline, e.g. via Excel.

On the other hand, process-relevant formulas can be stored on USB stick and thus, read out by ***Mm*** during the process.

## 2. Pinout

***PLC-Mem*** is supplied by 24 VDC. Current is approx. 100 mA, depending on the used USB stick.
***PLC-Mem*** communicates with ***Mm*** via its serial programming interface. For this purpose, the serial programming cable of ***Mm*** is used.



# 3. Requirements for the USB storage

In most of the cases, an USB stick will be used. The stick must comply with the following requirements:

| | |
|---|---|
| File system: | FAT16, FAT32 |
| Sector size: | 512 bytes |

If the stick is formatted else the LED besides the USB plug lights red (see Chap. 6 “Error indication”). In this case, the stick has to be re-formatted by Windows with the mentioned properties.

# 4. Features

## *Data Logging*

With the task “Data Logging” 20 signed 16-bit words are stored to the USB device. This data-organization is consistent with Millenium’s concept of data blocks (here: data blocks 29...48, details for communication, see Chap. 5 “Communication with Millenium”).

Each new data set is attached to an initially created log-file. The user has the possibility to enumerate and select multiple log-files, which are created on the USB stick as soon as they are demanded for the first time. Thus, data sets can be assorted.

Log-files are named “**LF-xxxxx.csv**”, with xxxxx as a 5-digit number.

The distance in time between 2 logging tasks should not fall short of 2 seconds.

## *Request Formulas*

With the task “Request Formula” 20 signed 16-bit words are read out of the USB device and transferred to ***Mm***. This data-organization is consistent with Millenium’s concept of data blocks (here: data blocks 5...24, details for communication, see Chap. 5 “Communication with Millenium”).

The user has the possibility to select single formulas (either within one file ore across more than one files) by means of specifying the file number and the line number. Remark that there is the possibility to address a non-existing data set. In this case 20 zeros are transferred to ***Mm***.

Formula-files have to be named “**LF-xxxxx.csv**”, with xxxxx as a 5-digit number.

The distance in time between 2 formula tasks should not fall short of 2 seconds.

## *Life-Message*

***PLC-Mem*** supports the cyclic exchange of status information. In a first step the control DBs of ***Mm***, in a second step those of ***PLC-Mem*** are transferred. This exchange serves as life message of the communicating devices. This is important, e.g. if data are logging only very seldomly, but ***Mm*** wants to be sure, that the receiver works properly.

# 5. Communication with Millenium

The communication is based on data blocks as required by ***Mm***. ***Mm*** acts both with write and read commands as client. The host (here: ***PLC-Mem***) can write into the DBs 1…24 and read out of the DBs 25…48 (each DB is a signed 16-bit value). DBs 1…5 and 25…28 are reserved for control characters, the residual DBs contain the user data.

**Handshake (see Fig.2):**

If ***Mm*** wants to log data, it has to write these data into the DBs 29…48 and has to set a marker-bit (DB25, bit 0).

In 1-second-intervalls, ***PLC-Mem*** requests this marker. When the logging-task is identified by ***PLC-Mem***, this is signalized by also setting a marker-bit (DB1, bit 0).

***Mm*** now resets its marker and thus acknowledges the data transfer. After storing onto the USB device ***PLC-Mem*** also resets its marker and thus signalizes “Job done”.

The same handshake as described above is applied with the tasks “Request formula” (Fig. 3) and “Life message”. Only the bit-addresses of the markers are different.

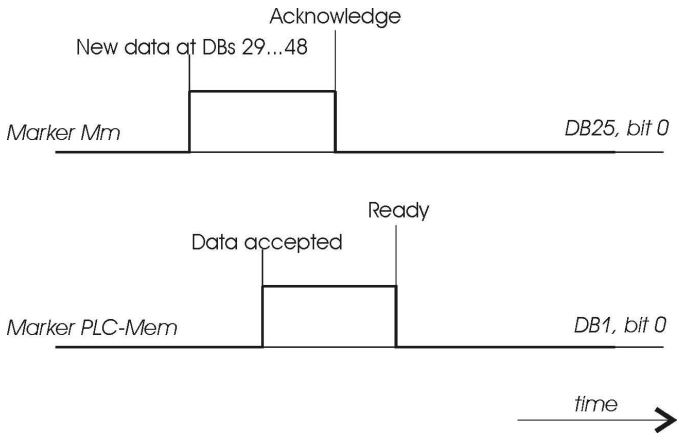



**Fig. 2: Handshake with data logging**

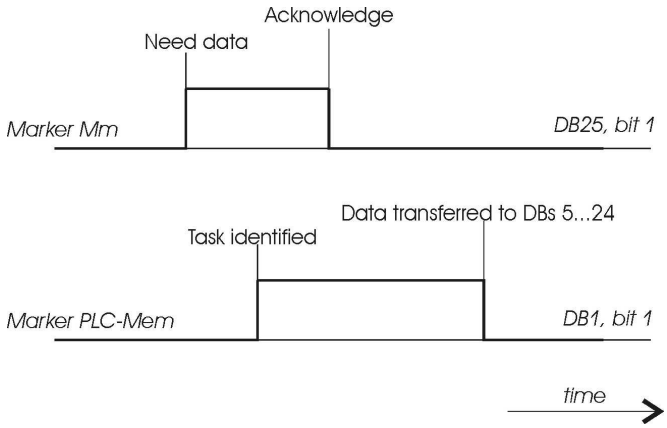



**Fig. 3: Handshake with formula request**

**Control characters**

| DB | bit | Function |
|---|---|---|
| 01 | 0 | Marker ***PLC-Mem*** „Data logged“ |
| 01 | 1 | Marker ***PLC-Mem*** „Formula uploaded “ |
| 01 | 2 | Life-bit ***PLC-Mem*** “Life-bit ***Mm*** read” |
| 01 | 3 | Error „No USB-Device“ |
| 25 | 0 | Marker ***Mm*** „Log data“ |
| 25 | 1 | Marker ***Mm*** „Request formula“ |
| 25 | 2 | Life-bit ***Mm*** busy |
| 26 | 0…15 | File number of log- or formula-file |
| 27 | 0…15 | Line number of the selected formula-file |

# 6. File Format

The format of both log-files and formula-files is CSV. This format can be read, e.g. by Excel (formulas generated by Excel please write as CSV). With CSV columns are separated by semicoleon and the lines by <ENTER>.

Data are displayed decimally, with values from –32768 to +32767. The positive sign and preceding zeros are not displayed with e.g. Excel and are also not exported into the CSV format.

# 7. Status and Error Indication

Possible Statuses are indicated to the user by means of 3 LEDs.

**LED besides the USB plug:**
Off, if no USB device is plugged
Green, if USB device is plugged
Red, if USB Device is not accepted
Blinking in case of data access

**LED green on main board:**
Always blinking green, otherwise module is defect.

**LED red on main board:**
Off, if status is ok
Red, if:
a) No USB-Device plugged
b) Communication to Mm interrupted

The error “No USB device” is also transferred to Mm. For this purpose serves the error flag (DB1, bit 3, see Chap. 5 “Communication with Millenium”).