EWIKON

pro **CONTROL**

Data communication

Supplement for
operating manual

About this document

While connectivity is one of the core features of the pro CONTROL temperature controller, listing all interfaces and supported protocols was too expansive to be included into the standard user manual. The creation of this document allowed for an in-depth description and to address not only end-users but also developers aiming to connect to an pro CONTROL controller.

1 Introduction

The connection of a temperature controller to an injection molding machine has become increasingly important in recent years.

Just a few advantages of such a connection are:

- If the **temperature controller** detects any disturbance (e.g. broken sensor), the machine can react to stop production automatically.
- In case of a **molding machine** malfunction, the controller can react automatically, for example, by lowering the temperatures (“standby”).
- The user can conveniently operate the injection molding machine and temperature controller on the **same display**. For this purpose, the injection molding machine manufacturer offers a special user interface for operating the zones of the hot runner controller.

In principle, the pro CONTROL controller supports two different methods of coupling to a machine:

1.1 Digital control inputs (24V) and relay contact outputs

The controller evaluates 24V DC signals from the molding machine. The digital inputs are used for the remote activation of functions such as “standby” or “locking the outputs”.

The controller also has three potential-free relay contacts to transmit a warning and/or alarm status to the injection molding machine.

Both of these input/output methods are not able to transmit any process values (e.g. temperature values) to the machine and it is also not possible to change any controller settings. This requires real data communication.

1.2 Data communication

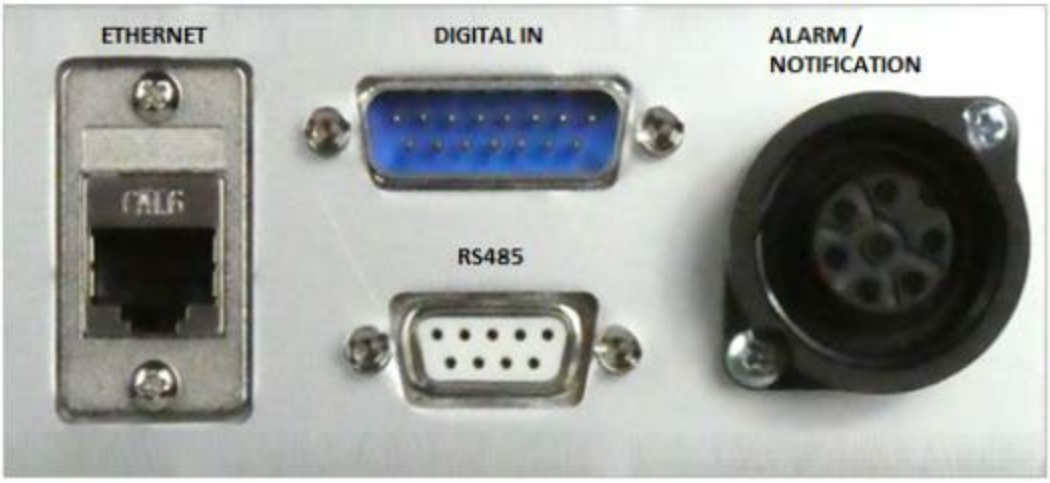
With data communication it is possible to get process values from the controller (temperature values, zone alarms, heating current etc.) and also to change controller settings.

To establish a connection between molding machine and temperature controller, it is very important to use the same “**communication protocol**”. It is comparable to human language: Only two people speaking the same language can communicate with each other.

Also the communication **interface** is important. By default, pro CONTROL has two interfaces: **RS485** and **ETHERNET.**

2 Digital input / relay output

The digital input / relay contact output connectors are located, like all other interfaces, on the rear of the pro CONTROL housing. Special cables are available from EWIKON Heißkanalsysteme GmbH to allow easy connection.

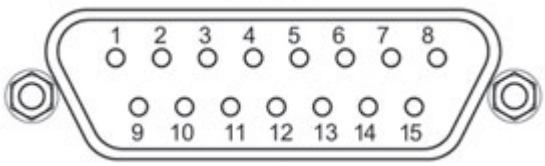


2.1 Digital inputs

The pro CONTROL controller evaluates 24V DC signals via a 15-pole D-sub connector. The digital inputs are PLC compatible, i.e., they operate over a voltage range of 13...30 VDC.

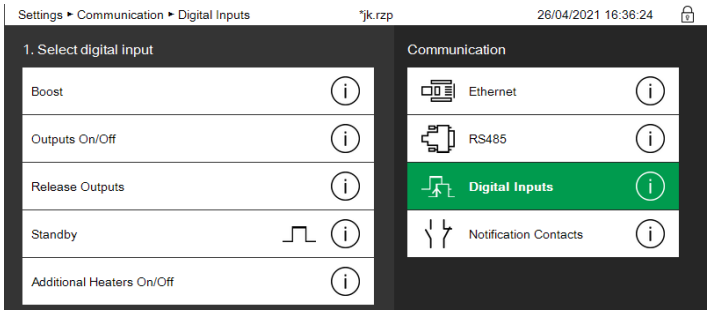
2.1.1 Wiring digital inputs

Table 1 Digital input	Function	
1	Boost	+24 V_{DC}
2	Outputs On/Off	+24 V_{DC}
3	Release Outputs	+24 V_{DC}
4	Standby	+24 V_{DC}
6 – 8	Ground	0 V_{DC}
9	Add. Heaters On/Off	+24 V_{DC}
13 – 15	Ground	0 V_{DC}



2.1.2 Configuring digital inputs

The configuration of digital inputs is available in the user interface at menu “Settings > Communication > Digital Inputs”.



For each single input function it is possible to select how the controller shall react:

- activate with 24V (high-active)
- activate with 0V (Low-active)
- change with rising edge
- change with falling edge
- Disable input

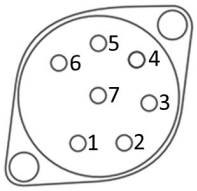
Press the respective info buttons ⓘ to get more detailed information.

2.2 Digital outputs (contacts)

2.2.1 Wiring digital outputs

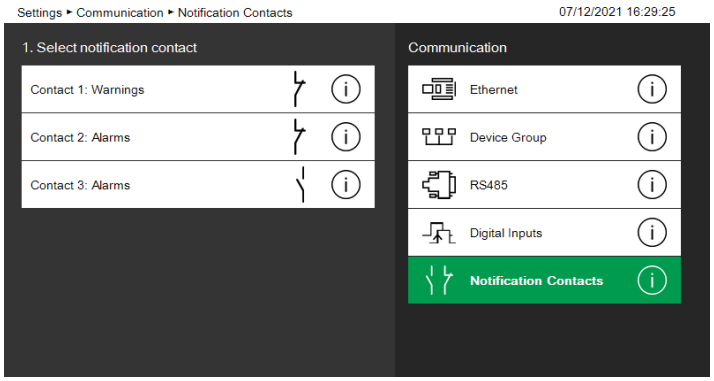
The pro CONTROL controller has the following two potential-free contacts

Table 2 Alarm/ Notification socket	Function
Pin 1 ○—╲—○ Pin 3	Collective warning
Pin 4 ○—╲—○ Pin 5	Collective alarm
Pin 2 ○—╲—○ Pin 6	Collective alarm (inverted by default)



2.2.2 Configuring digital outputs

The configuration of digital outputs is available at menu “Settings > Communication > Notification Contacts”.



All contacts can be configured as make (normally closed) or break contacts (normally open).

Press the respective info buttons ⓘ to get more detailed information.

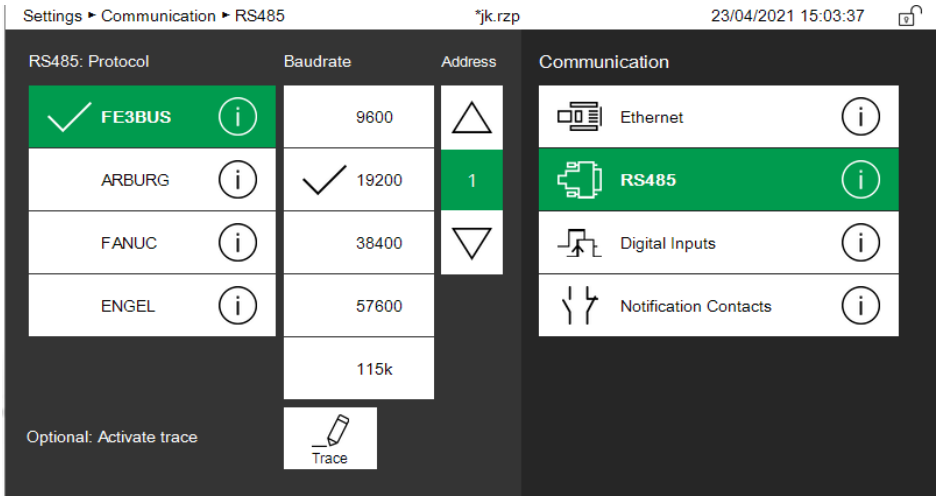
3 Data Communication via RS485

RS485 is a standard defining the electrical characteristics of drivers and receivers for use in serial communications systems. Digital communication networks implementing the standard can be used effectively over long distances and in electrically noisy environments. Multiple receivers may be connected to such a network in a linear, multidrop bus system. These characteristics make RS485 useful in industrial control systems and similar applications.

Note that the controller never expects to be the so-called bus master and always assumes the passive role waiting for commands.

3.1 Controller settings

For the RS485 interface, all settings can be selected at menu “Settings > Communication > RS485”



3.1.1 Protocol

To establish a connection to an injection molding machine, pro CONTROL supports different data communication **protocols**.

“FE3BUS” is a proprietary protocol which supports reading or modifying most of the controller settings.

All other available protocols are named by the respective machine manufacturer. They are more or less extensive and allow only a subset of the available controller functions.

3.1.2 Baud rate

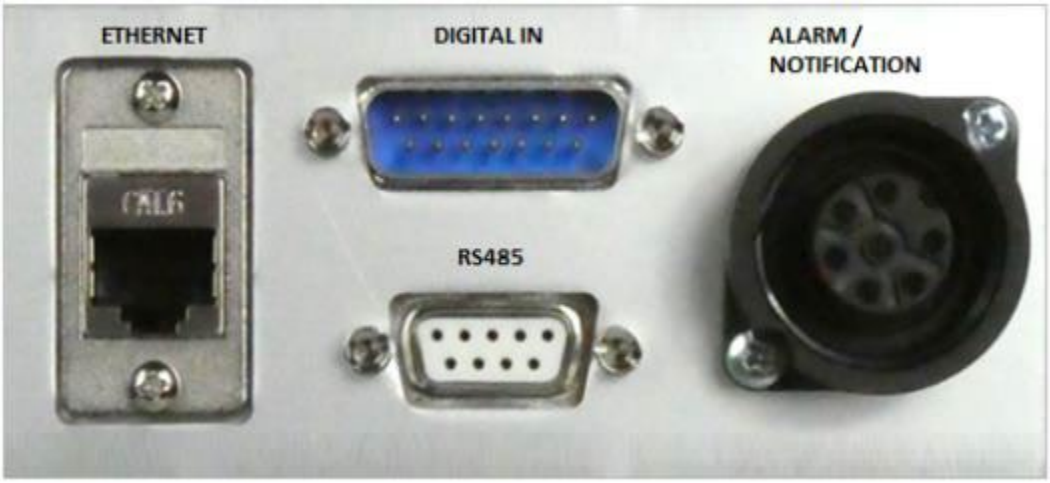
In addition to the selected protocol it is also important, that machine and controller are using the same **baud rate** (transmission speed).

3.1.3 Address

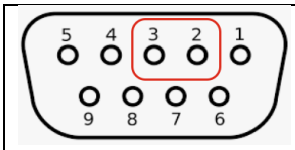
As RS485 is an interface for connecting up to 30 temperature controllers in one bus system, each of the controllers on this bus must be set to a different **address**. By default pro CONTROL is set to address 1 and if you do not have more than one controller leave this setting unchanged.

3.2 Wiring of a RS485 connection in general

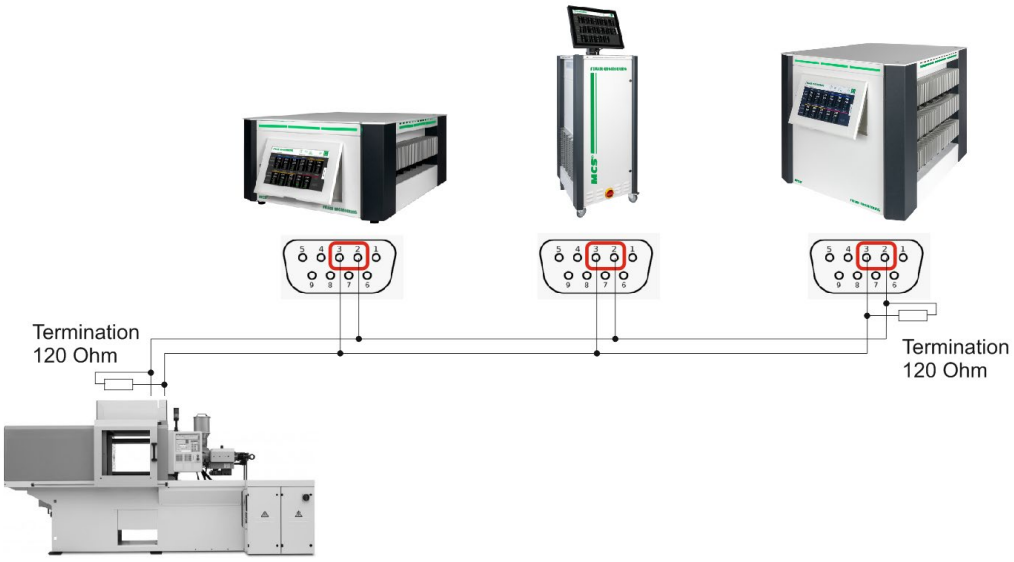
The RS485 Interface is located - like all other interfaces - on the rear of the pro CONTROL housing.



Only two pins of this female 9pole D-sub connector are used:

	**PIN2:** Rx/Tx+ **PIN3:** Rx/Tx-

You have to use a **shielded** 2-pole **twisted** communication cable. Connect all RS485 serial interfaces in parallel, like shown in the picture below. Ideally, an RS485 system consists of a single linear cable (no branches) with **120 ohm resistors** connected across the two wires at each end of the cable. RS485 can handle line lengths of up to 1 km.



Please note, that the connection of RS485 is not standardized and each different injection molding machine may have different methods to connect the two RS485 wires. Therefore please carefully refer to the manual of your machine manufacturer.

3.3 FE3 Protocol (RS485)

3.3.1 General Specification

“FE3" is the most comprehensive RS485 protocol available and it supports most of the functions of the pro CONTROL controller. Every single EWIKON temperature controller supports this protocol and thus you can connect many different controller types to one single data bus. The specification of this protocol is open and available as a separate document. There are different injection machine builders who are supporting this protocol, one of them is SUMITOMO DEMAG.

3.3.2 Wiring a pro CONTROL controller to a Feller SI13u

The SI13u converter connects a USB interface to the RS485 two-wire interface. Due to the small dimensions of the housing and the low energy consumption of the SI13u, it is ideally suited for stationary use on PCs as well as for mobile use on laptops.

FELLER SI13u			pro CONTROL	
Pin #	Name		Pin #	Name
2	RS485 B /+	→	2	RS485 B /+
3	RS485 A /-	→	3	RS485 A /-

3.3.3 Protocol Features

The protocol allows to operate almost all parameters and process values.

3.3.4 Protocol definition

Please contact the EWIKON support, if you need to know detailed FE3 protocol specifications. This document is available separately, as the scope would exceed this general overview.

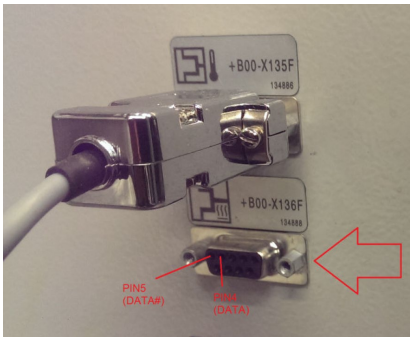
3.4 ARBURG RS485 (Protocol “Euromap 17”)

3.4.1 General specification

pro CONTROL communicates to ARBURG injection molding machines based on the communication standard **EUROMAP 17**. The machine needs to be equipped with “Special option for 458/01: with Euromap 17”. The maximum baud rate for ARBURG machines is 9600 baud.

3.4.2 Wiring to an ARBURG IMM

This is a picture of the ARBURG RS485 interface on a machine type “270A”



On ARBURG machines the communication pins are 4 and 5 and called “DATA#” and “DATA”. Connect the communication bus to pro CONTROL as shown in the table.

ARBURG			pro CONTROL	
Pin #	Name		Pin #	Name
5	DATA#	→	2	RS485 B /+
4	DATA	→	3	RS485 A /-

Refer to ARBURG documentation how to set up the correct baudrate at their machine.

3.4.3 Protocol Features

- Receiving and setting of temperature setpoints
- Receiving and setting of standby setpoints
- Receiving and setting of power output level (%) in manual mode
- Receiving and setting of temperature tolerances
- Receiving the number of zones
- Receiving the actual temperature values
- Receiving the current power output level (%)

- Receiving the zone status
  - High temperature alarm
  - Low temperature alarm
  - Upper deviation warning
  - Lower deviation warning
  - Broken thermocouple
  - Heating current alarm
- Setting the zone operation mode

3.4.4 Protocol definition

For any protocol details, please refer to Euromap 17 specification, available at their homepage:
https://www.euromap.org/technical-issues/technical-recommendations

Referring to document **eu17an.pdf,** also available at www.euromap.org, the following Parameters are supported by pro CONTROL to communicate to ARBURG IMM:

Mnemonics of parameter	**R = Read W= Write**	**Description**
SA	R/W	Setpoint of Temperature
SB	R/W	Standby Setpoint
SO	R/W	Setpoint of output value (%) in manual mode
UD	R/W	Upper deviation alarm value
LD	R/W	Lower deviation alarm value
CN	R	Active channel numbers
DI	R	Device identification
DM	R	Device Mnemonics (implemented EM17 parameters)
FF	R	Standard question, standard answer
PV	R	Actual Value, process value of temperature
CO	R	Controller output (0..100%)
SW	R	Status word Bit 0: 1=zone ON Bit 1: 1= Process Alarm (LO-Alarm or HI-Alarm or LC-Alarm or Fuse-Alarm or Deviation HI Alarm or Low Voltage Alarm) Bit 2: 1= Machine Alarm (Triac-Alarm or HiHi-Alarm or LC-Alarm) Bit 3: not supportet, always=0 Bit 4: 1=upper Limit alarm (Hi-Alarm or HiHi-Alarm) Bit 5: 1=lower Limit alarm Bit 6: 1=upper deviation alarm

		Bit 7: 1=lower deviation alarm Bit 8..10: not supported, always=0 Bit 11: 1=Sensor failure Bit 12: not supported, always=0 Bit 13: 1=current alarm Bit 14..15: not supported, always=0 Bit 16: 1= controller active Bit 17..18: not supported, always=0 Bit 19: 1= standby active Bit 20: 1= manual mode Bit 21..31: not supported, always=0
MW	W	Mode word Bit 0: 1=zone ON Bit 1..4: ignored Bit 5: 1=Manual Mode Bit 6..31: ignored
UN	R	Temperature Unit (0=Celsius, 81 = Fahrenheit)

3.5 ARBURG TTY (Protocol “HB-Therm”)

3.5.1 General specification

From pro CONTROL version 4.5, the controller is able to communicate to older ARBURG injection molding machines via TTY (20mA) interface. **The TTY interface is optional and must be specified separately when ordering the device**.
The communication is limited to controllers with max. 32 zones and the transmission rate is fixed to 4800baud.

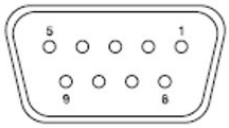
3.5.2 Protocol Features

- Setting of temperature setpoints or output power (%)
- Setting operation mode (OFF/MAN/AUTO)
- Receiving the actual temperature values
- Receiving status of: Zone on/off, Standby, manual/auto, load interruption, over current, sensor fault

3.5.3 Wiring to an ARBURG IMM

3.5.3.1 Arburg side

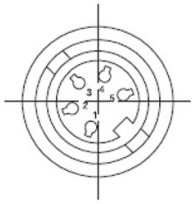
Schnittstelle für alle externen Geräte mit SELOGICA Steuerung



Pinbelegung:

2 +20 mA
3 -20 mA
10 Schirm

Schnittstelle Heißkanal für DIALOGICA 8, 16 Steuerung

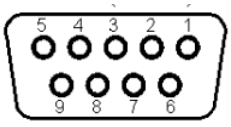


Pinbelegung:

1 +20 mA
3 -20 mA

3.5.3.2 pro CONTROL Side

The TTY interface is available via the connector labeled RS485/TTY.



Pin 5 = 20mA +
Pin 6 = 20mA GND

3.6 FANUC Protocol

3.6.1 General Specification

FANUC machines are using a highly restricted MODBUS RTU protocol based on PSG's ETR132 controller specification to communicate with any other hot runner controller.

3.6.2 Wiring to a FANUC IMM (via RS485)

The injection molding machine needs to be equipped with a “MOXA Box”, which provides the RS485 interface. The connection schema between controller and the Moxa Box is

FANUC (MoxaBox)			pro CONTROL	
Pin #	Name		Pin #	Name
	R+/D+	→	2	RS485 B /+
	R-/D-	→	3	RS485 A /-

A special cable “AU 0069” is available from EWIKON. Please refer to the FANUC manual for correct setup of the RS485 data interface.

3.6.3 Wiring to a FANUC IMM (via Ethernet)

This option is available from pro CONTROL Version 4.5.
On the controller you manually must set an IP address that matches the settings of the machine. The IP address must not already be occupied by another subscriber. Then you can connect the controller by an Ethernet cable to the network switch inside the machine.

3.6.4 Protocol features

- Receiving and setting of temperature setpoints
- Receiving and setting of temperature tolerances
- Switching zones on and off
- Receiving the actual temperature values
- Receiving the power output level (%)
- Receiving the heating current (A)Receiving the zone status
    - Lower deviation warning
    - Upper deviation warning
    - Heating current deviation alarm
    - Defective triac alarm
    - Broken thermocouple
    - Shut-off temperature exceeded

3.6.5 Communication

The protocol is created for Master-Slave-Applications. One Master talks to one or several Slaves. A Slave is activated in that case, a Master talks to him.

3.6.6 Protocol definiton

3.6.6.1 General Conditions

In the reply of the controller, first the address and the function-number is reflected. Then the corresponding Data bytes and a Checksum-word follows. In case of a write-operation the databytes are a repetition of the transmitted data, otherwise the requested data are transmitted.

The end of an operation at Modbus-Protokoll in RFU-Format is characterized by an transmission pause of about 3½ characters. If the controller does not receive any character during the time shown in the following table, the controller is waiting for a new address, which starts a new order-sequence.

Depending on the Baudrate there are the following times:

Baudrate in Bits/s	1200	2400	4800	9600	19200
Time T in ms	32.5	17.5	10	7.5	5

3.6.6.2 Checksum

The checksum is created according to the RTU-Format by CRC-16 (16-bit Cyclic Redundancy Checksum). This CRC-16 is created during the whole transmission-sequence and completes each transmission. The checksum is a WORD, the Low-Byte is transmitted first followed by the High-Byte.

Example for the Calculation of an CRC-Code for the following operation:
25 01 00 02 00 03 (Hex)

The Checksum-Word for this sequence is: 2FDB so the complete operation results:
25 01 00 02 00 03 DB 2F (Hex)

The checksum can be calculated in the following way (e.g. in) :

```
static short crc16(unsigned *c_first, unsigned *c_last)
{
        short j;
        unsigned int crc = 0xFFFF;
        for (c_first; c_first <= c_last; c_first++)
        {
                crc ^= *c_first;
                for (j = 8; j > 0; j--)
                {
                        if (crc & 0x0001)          // if the last bit is one
                        {
                                crc = crc >> 1;
                                crc ^= 0xA001;  // 0xA001 is the CRC-16 polynom
                        }
                        else
                        {
                                crc = crc >> 1;   // shift 1 bit
                        }
                }
        }
        return crc;
}
```

3.6.6.3 Protocol-frame

The protocol-frame looks like:

Waiting Time	Address	Function	Data	Check	Waiting Time
T	8-Bits	8-Bits	N x 8-Bits	16-Bits	T
	1 Byte	1 Byte	N x 1 Byte	2 Bytes	

3.6.6.4 Addressing

Each zone in Modbus has it's own address. Just 256 zones can be addressed because there is only one byte reserved for each address. Actually FANUC machines are limited to 96 Zones.

3.6.6.5 Errorcodes

The following errorcodes are supported:

Error-code	Name	Meaning
01	illegal function	Function-number is not allowed
02	illegal Data-address	Parameter is not supported
03	illegal Data-value	Data-value false / function can not be carried out

3.6.7 Detailed description of MODBUS-functions

3.6.7.1 Read n Word (Function Code 03)

This operation makes possible to read a single word or even more words. The sending-operation looks as follows.

Master	Addr.	Funct.	Address of the 1. Word		Number of Words		CRC-16	
Byteno.	1	2	3	4	5	6	7	8
HEX		03	HI	LO	HI	LO	LO	HI

The Parameter "Address of the 1. Word" describes the table index of a word, listed in the word table. The parameter "Number of Words" describes the number of words which shall be read after the provided table index.

The controller answers in the following way:

Slave	Addr.	Funct.	Byte-Count	Word 1		Word ...		last Word		CRC-16	
Byteno.	1	2	3	4	5	?	?	?	?	?	?
HEX		03		HI	LO	HI	LO	HI	LO	LO	HI

The parameter "ByteCount" describes the number of transmitted data bytes, therefore it is twice the number of the requested words. First the high byte will be transmitted and then the low byte.

Example: A word shall be read from table index 2 in channel 2 of controller with address no. 5.

Master	Addr.	Funct.	Address of the 1. Word		Number of Words		CRC-16	
Byteno.	1	2	3	4	5	6	7	8
HEX	A2	03	00	02	00	01	3D	59

Slave	Addr.	Funct.	Byte-Count	Word 1		CRC-16	
Byteno.	1	2	3	4	5	6	7
HEX	A2	03	02	03	E8	7C	E3

ByteCount = 0x02 (2 data bytes will be transmitted)

Word 1 = 0x03E8 = 1000 (1000, because the setpoint is given in 0,1 °C)

3.6.7.2 Write 1 Word (Function Code 06)

With this operation it is possible to write into a word-parameter. The structure looks like this:

Master	Addr.	Funct.	Address of Word		Data value		CRC-16	
Byteno.	1	2	3	4	5	6	7	8
HEX		06	HI	LO	HI	LO	LO	HI

The parameter "Address of Words" describes the table index of a parameter listed in the word table. The "data value" is the value to which the controller parameter shall be set. The value range should be checked.

The controller answers after carrying out the operation with the same operation.

Example: The setpoint of channel 0 in controller with address 1 shall be set to 10°C. The table index of setpoint is and must be set in 0,1°C. The command looks like this.

Master	Addr.	Funct.	Address of word		Data value		CRC-16	
Byteno.	1	2	3	4	5	6	7	8
HEX	20	06	00	02	00	64	2F	50

3.6.7.3 Loopback Test (Function Code 08)

With this operation it is possible to test the communication. The structure of the operation looks like this:

Master	Addr.	Funct.	Diagnostic- Code		Data		CRC-16	
Byteno.	1	2	3	4	5	6	7	8
HEX		08	00	00	HI	LO	LO	HI

Parameter "Diagnostic-code" allows to set which data the controller shall send back. pro CONTROL supports the "Diagnostic-code" 0x0000 only. This code declares that the sent data will be sent back in data field 1:1.

Example: The data value 0x1234 of channel 4 in controller with address 1 shall be reset.

The command and the answer look like this.

Master	Addr.	Funct.	Diagnostic- Code		Data		CRC-16	
Bytenr.	1	2	3	4	5	6	7	8
HEX	24	08	00	00	12	34	EA	49

3.6.7.4 Write n Words (Function Code 16)

This operation makes it possible to write several words with varying data values.

The sending operation looks like this:

Master	Addr.	Funct.	Addr. of 1. Word		Quan-tity		Byte-Count	Data word 1		Data word ...		CRC-16	
Byteno.	1	2	3	4	5	6	7	8	9	?	?	?	?
HEX		10	HI	LO	HI	LO	LO	HI	LO	HI	LO	LO	HI

The parameter "Address of 1. Word" correspond to the index of a controller parameter, listed in the Modbus word table.

The "Quantity" is the number of words, which shall be written into the word table from the index on, which is written on sentence before.

In Parameter "ByteCount" the number of the transmitted "data words" are provided.

In the "data words" the value of a controller-parameter is transmitted. The value range of the controller parameter must be set right.

Slave	Addr.	Funct.	Address of 1. Word		Quantity		CRC-16	
Byteno.	1	2	3	4	5	6	7	8
HEX		10	HI	LO	HI	LO	LO	HI

(The controller repeats everything except the "data words")

Example: From table-index 2 of canal 4 in the controller with address no. 3, 2 words shall be written (Set point=50°C and STG=0

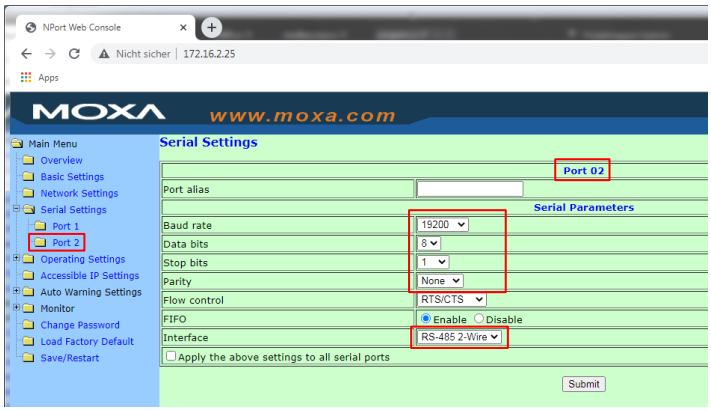
Master	Addr.	Funct.	Addr. of 1. Word		Quantity		Byte-Count	Data word 1		Data word 2		CRC-16	
Byteno.	1	2	3	4	5	6	7	8	9	10	11	12	13
HEX	A3	10	00	02	00	02	04	01	F4	00	00	3B	C2

Slave	Addr.	Funct.	Address of 1. Word		Quantity		CRC-16	
Byteno.	1	2	3	4	5	6	7	8
HEX	A3	10	00	02	00	02	F9	4A

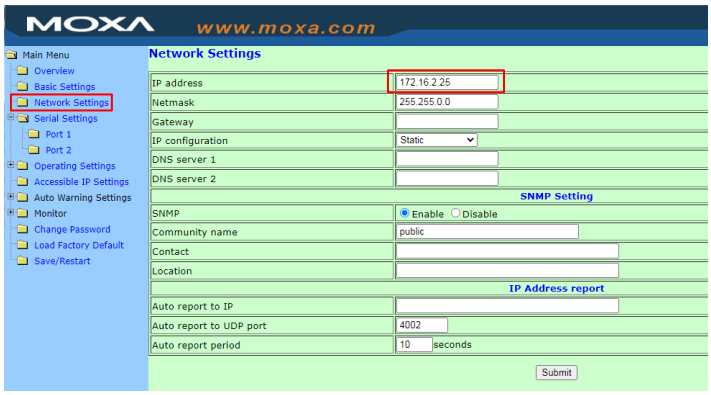
3.6.8 Communication settings

3.6.8.1 Serial Communication via RS485

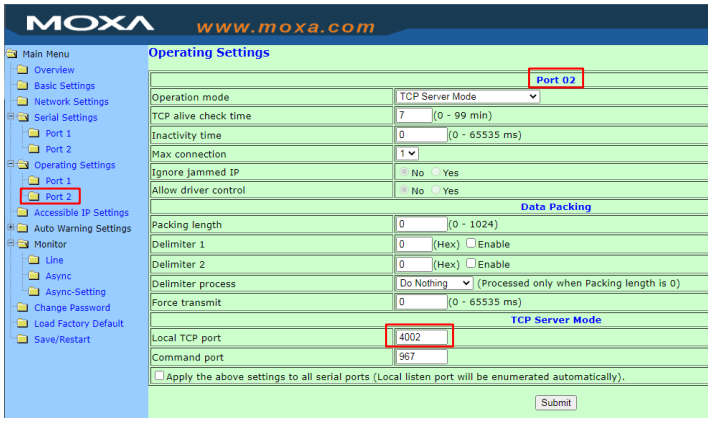
All serial communication is handled by a converter called "Moxa Box", which is an optional device inside the FANUC machine. The Moxa Box converts TCP communication from the machine to RS485 serial communication. This device has its own web interface through which its possible to configure all serial parameters. It must be configured with the following settings:
8 data bits, 1 stop bit, parity none
The baud rate must be the same as configured on the pro CONTROL controller.



The IP Adress needs to be the same as shown in screenshot, chapter 3.6.8.2 → Setting (2)



The TCP port needs to be the same as shown in screenshot, chapter 3.6.8.2 → Setting (3)

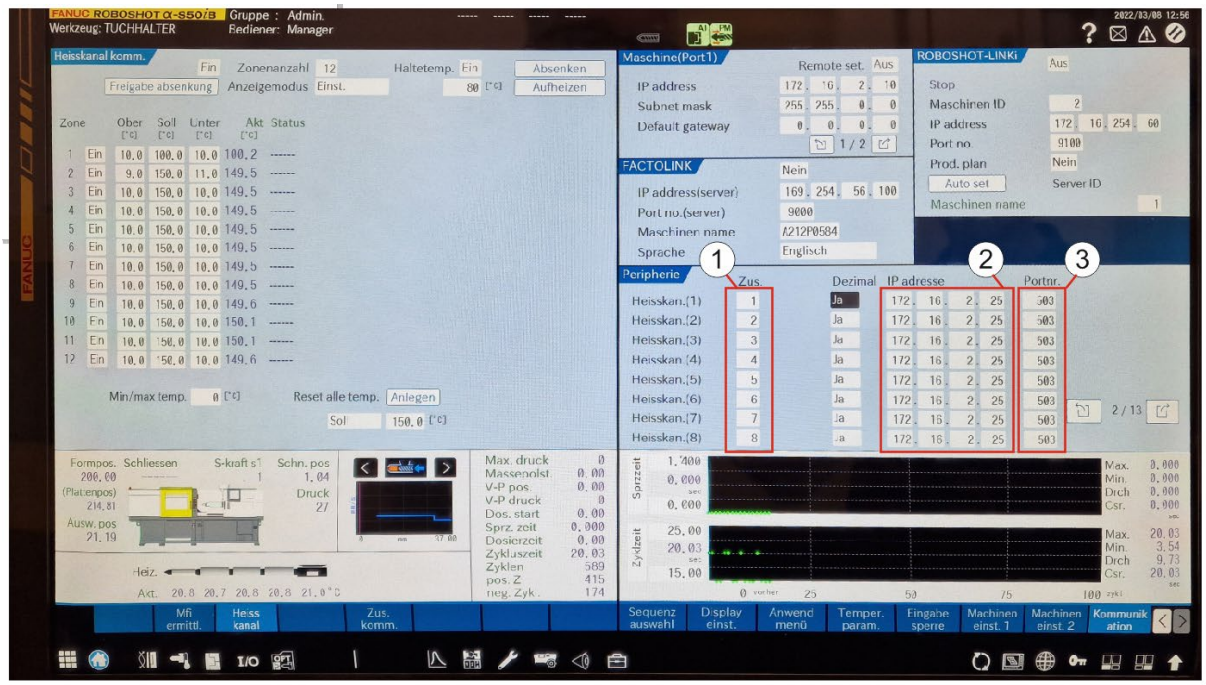


3.6.8.2 Communication via ETHERNET

This option is available from pro CONTROL Version 4.5. pro CONTROL is able to communicate with the FANUC machine via TCP Port **503**. With this option the Moxa Box ist not required.

At the machine settings you have to configure for each single zone
(1) the zone number (1..96)
(2) the IP address from the temperature controller
(3) the Port number (always 503 for direct communication to pro CONTROL).

see the following screenshot



3.6.9 Word-parameter table

The word-parameter are shown in the following table:

Index-Nr.	Parameter	Remarks	range [unit] 1)	Faktor 2)	Read / Write
1	Actual temperature	123,4° is transmitted as 1234 (factor 10) 800,0° or higher in case of sensor fault.	0-999 [°C/°F]	10	R
2	Setpoint	123,4° is transmitted as 1234 (factor 10) 65516 to switch off zone.	0-999 [°C/°F]	10	R/W
3	Output power		0..100 [%]	1	R/W
4	Heating Current		0,0-99,9 [A]	0,1	R
5	State word 1	Bit 0=1: temperature alarm low Bit 1=1: temperature alarm high Bit 2=1: current tolerance alarm Bit 3=1: current thyristor alarm Bit 4=1: <free> Bit 5=1: alarm maximum temperature Bit 6=1: sensor short circuit Bit 7=1: sensor break			R
6	State word 2	Always 0			R
26	Limit value +		0-999 [°C/°F]	10	R/W
27	Limit value -		0-999 [°C/°F]	10	R/W
94	Unit temperature		°C = 1 °F = 0		R

3.7 ENGEL Protocol

3.7.1 General Specification

ENGEL uses the “HB THERM” protocol to communicate to hot runner controllers.

3.7.2 Wiring to an ENGEL IMM

Please see machine documentation to connect to a pro CONTROL controller.

3.7.3 Protocol features

“HB Therm” is a very restricted communication protocol.

- Switching zones on and off
- Setting of temperature setpoints
- Receiving temperature values
- Receiving the zone status
    - Broken thermocouple
    - Heater failure (Fuse or triac defect, no load)
    - High temperature alarm

3.7.4 Protocol definition

The data transmission between the control unit of the injection moulding machine (IMM) and pro CONTROL is basically initiated by the IMM; consequently, pro CONTROL sends data only after being requested by the control unit (master-slave principle). The protocol takes into account that several units can be connected to one transmission line, which must be addressed individually with their address setting.

3.7.4.1 Security Concept

Four security levels are provided to prevent transmission errors:

1. Each transmitted byte consists of 8 data bits and one parity bit (parity even).
2. The data transfer is based on the master/slave principle. The master (IMM) always requests the slave (pro CONTROL) to send, while the slave only sends on request. Only the slave acknowledges the received messages; either with the associated response, which contains the same data set identifier as the transmission request (=reception OK), or with the reply "NAK" (= faulty reception). If acknowledged with "NAK", the master repeats the original message up to three times. If, after the third attempt, the response is again "NAK", the transmission line is considered to be disturbed and the IMM reports this as an error.
3. Each message (including acknowledgements) contains three bytes, which, encoded in ASCII, contain the binary number of bytes of the entire message (including the checksum bytes, see point d.). On receipt, a check is made to ensure that the correct number of bytes has arrived.
4. Each message is terminated by two checksum bytes, which are contained in the block length bytes (point c). The checksum bytes are the ASCII coded 8 bit wide

binary sum of all bytes of a message,including the block length bytes, but excluding the checksum bytes.

3.7.4.2 transmission protocol

Only ASCII characters are transmitted in order to offer an easy diagnostic possibility with a V24 terminal. The highest data bit (bit 7) is set to "1" (= 80H) for the address code (= unit number) and to "0" for all following data of the message. This only applies to transmissions of the master. It makes it easier for the units to identify messages intended for them. An asynchronous transmission with 8 data bits is used, noParity, one start and one stop bit. The LSB is sent first.

3.7.4.2.1 Master writes setpoints

1 Byte (byte 1)	Address Master: B1H - BFH - device no. 1 – 15
3 Bytes (byte 2..4)	Block length 30H,30H,3EH = 14 Bytes
1 Byte (byte 5)	Identifyer (“A” = 41h)
4 bytes (byte 6..9)	setpoint in degrees Celsius, three places before and one place after the decimal point. Each (byte 6 - 9) byte is coded in ASCII (30H - 39H). Negative temperatures up to -99 degrees are possible (minus sign: 2DH).
1 byte (byte 10)	reserve (Always 60h)
1 byte (byte 11)	Control commands: 61H ("a"): Operation mode = OFF 70H ("p"): Operation mode = OFF 72H ("r"): Operation mode = Control (= normal operation).
1 byte (byte 12)	reserve(Always 20h)
2 bytes (byte 13..14)	Checksum Value range: 30H,30H ... 3FH,3FH (00h - FFh)

3.7.4.2.2 Slave responds to message 41H with actual values.

1 Byte (byte 1)	Address Slave: 31H - 3FH - device no. 1 – 15
3 Bytes (byte 2..4)	Block length 30H,31H,33H = 19 Bytes
1 Byte (byte 5)	Identifyer (“A” = 41h)
4 bytes (byte 6..9)	Actual value in degrees Celsius (coded as above). Negative temperatures up to -99 C are reported by a 2DH ("-") in the first position. (e.g. -43.7 ==> 2DH,34H,33H,37H)
4 bytes (byte 10..13)	Output power (0…100%)
1 byte (byte 14)	status word for feedback (coded bit by bit) Bit 0: Controller operating mode (1 = manual, 0 = computer) Bit 1: Always 1 Bit 2: Setpoint check: (0 = value okay, 1 = value not allowed) Bit 3: Always 0 Bit 4: Common alarm (see next byte for details) Bit 5: Always 1 Bit 6: Always 1 Bit 7: Always 0
1 byte (byte 15)	status word for alarms 1 (additional message to bit "collective alarm" in previous byte) Alarm is present if the respective bit is set. Bit 0: 1=Sensor break Bit 1: 1=Heating defective Bit 2: Always 0 Bit 3: Always 0 Bit 4: Always 0 Bit 5: 1=Temperature exceeds Hi-Limit Bit 6: Always 1 Bit 7: Always 0
1 byte (byte 16)	status word for alarms 2 Always 40h
1 byte (byte 17)	status word feedback 72H ("r"): Control active (= normal operation) 73H ("s"): OFF (= heating and cooling off)
2 bytes (byte 18..19)	Checksum Value range: 30H,30H ... 3FH,3FH (00h - FFh)

3.7.5 example of messages

3.7.5.1 Setpoint transmission (IMM ==> pro CONTROL)

Byte #	Content (hex)	
1	B1	Address 1
2	30	14 bytes length
3	30	
4	3E	
5	41	Always “A”
6	31	123,0° setpoint value
7	32	
8	33	
9	30	
10	60	Fixed
11	72	“r” = normal operation
12	20	Fixed
13	34	last 2 bytes of the checksum of 348H
14	38	

3.7.5.2 Actual value transmission (pro CONTROL ==> IMM)

Byte #	Content (hex)	
1	33	Address 3
2	30	13h = 19 bytes length
3	31	
4	33	
5	41	Always “A”
6	30	27,0° actual value
7	32	
8	37	
9	30	
10	2D	-30% cooling
11	30	
12	33	
13	30	
14	70	Status word feedback: Collective alarm active
15	41	Status word alarm 1: Sensor break
16	40	Status word alarm 2: No alarm in this word
17	72	Status word acknowledgement: Control active
18	3F	Last 2 bytes of the checksum of 3F4H
19	34	

3.7.5.3 Not acknowledge (pro CONTROL ==> IMM)

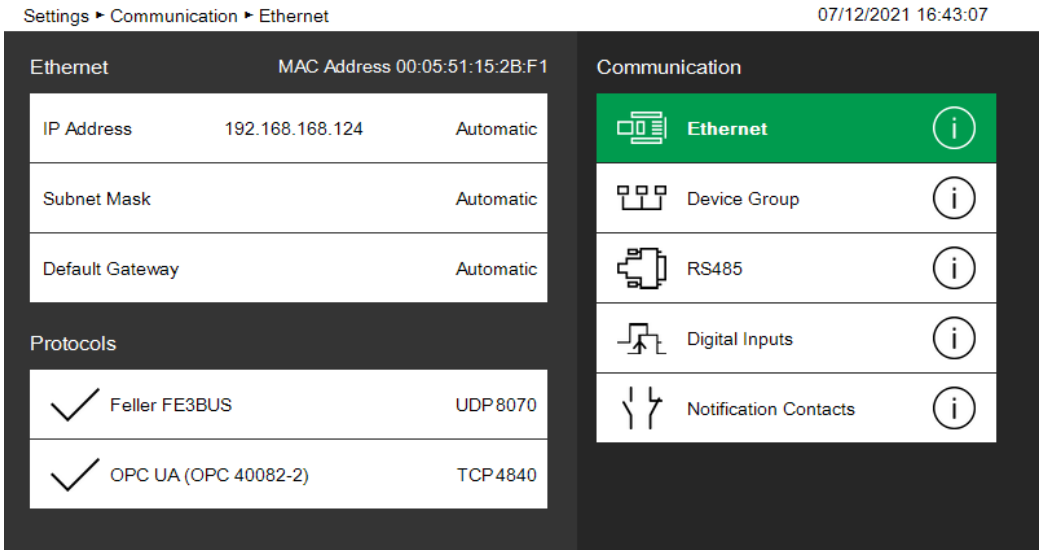
1	3A	Address 10

2	30	7 bytes length
3	30	
4	37	
5	7F	NAK
6	35	last 2 bytes of the checksum of 150H
7	30	

4 Data communication via Ethernet

As already mentioned in the chapter "RS485", various protocols are also used with ethernet to enable communication with injection molding machines.

Currently FE3BUS protocol and OPC UA (OPC 40082-2) are available and both of them are activated by default.

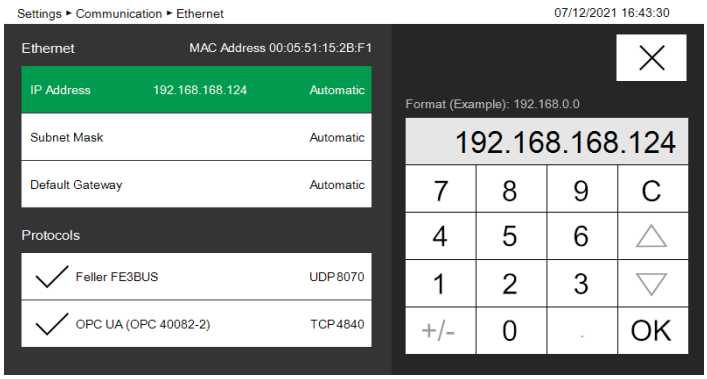


4.1 Controler settings

4.1.1 IP address / subnet mask

Each pro CONTROL controller that uses ethernet communication must be configured with its own IP address, which must be unique in the network. If the controller is connected to a network with a DHCP server available, the IP address as well as the subnet mask can be obtained automatically. **The automatic IP address setting is activated by default, see screenshot above.**

It is also possible to set the addresses manually. By selecting the IP address, changing the setting to “Manual” and confirming by the keypad button you can enter the address. Same applies to the subnet mask and default gateway if required by your network.



4.2 FE3 Protocol (UDP)

The protocol is suited to operate all controller settings and process values. Please see also Chapter 3.3.

Analogous to the serial RS485 communication port, the same protocol is available via ethernet based on UDP on port 8070. The description of this protocol is open and available as a separate document.

4.3 OPC UA (EUROMAP 82.2 resp. OPC 40082-2)

OPC Unified Architecture (OPC UA) is a platform-independent standard, developed by the OPC Foundation. It allows various kinds of systems and devices to communicate by sending request and response messages between clients and servers or network messages between publishers and subscribers over various types of networks.

EUROMAP, the European association of plastics and rubber machinery manufacturers, has defined various OPC UA specifications for different applications.

OPC 40082-2 describes the interface for hot runner devices (HRD) for data exchange via OPC UA. The target of OPC 40082-2 is to provide a standard interface for hot runner devices from different manufacturers to ensure compatibility. The following functionalities are covered:

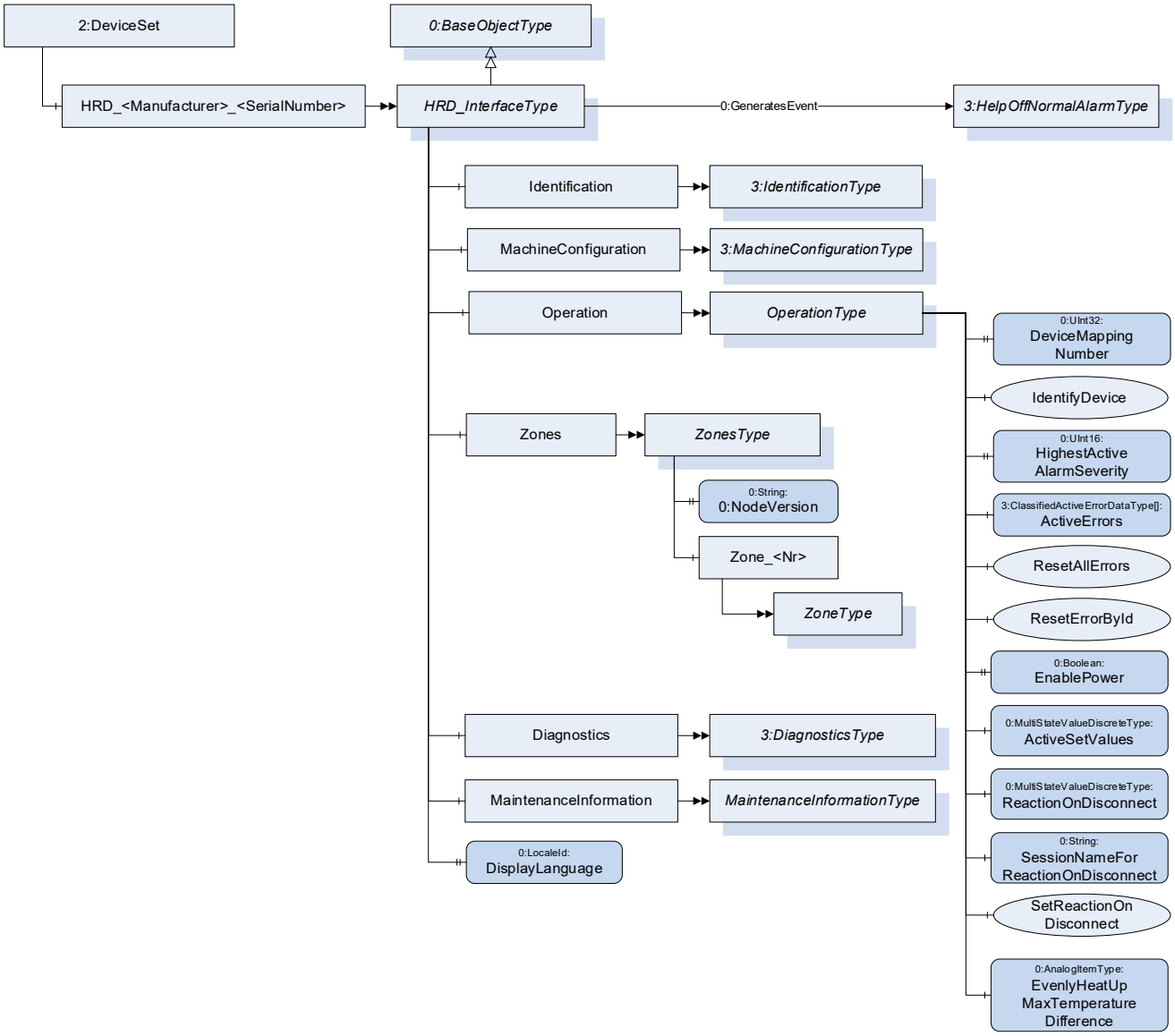
- General information about the hot runner device
- Status information
- Process data

Safety-related signals like emergency stop are not included.

OPC 40082-2, also known as EUROMAP 82.2, was first officially released as version 1.00 on 2021-06-01.

The pro CONTROL controller by EWIKON has an integrated OPC UA server by default and fully supports OPC 40082-2 v1.00 from version pro CONTROL version v4.2 and higher. The user does not need to configure anything. It is always active and each OPC UA client supporting the standard can communicate to pro CONTROL based on that specification. The OPC UA server is available via TCP on port 4840.

4.3.1 Information Model Basics



The schematic above represents the information model defined by OPC 40082-2. The pro CONTROL OPC UA server strictly follows that specification and presents the node model under DeviceSet.HRD_EWIKON_<SerialNumber>

The most important subnodes are

Operation — This ObjectType contains components which are necessary to operate the hotrunner device. It is formally defined in OPC 40082-2, table 2.

It contains information like currently active notifications (or errors) and allows to enable/disable power output for the whole device.

Zones — Zones is a container for all temperature zones of the cotroller in analogy with the container concept in OPC 40083. The individual zones are listed with numeric browse names like “Zone_5” which then implement the object type ZoneType. ZoneType represents all functionalities of a heating zone, such

as temperature monitoring, control, heatup and is formally defined in OPC 40082-2, table 14.

The individual zone objects contain the individual process values and parameters to control each zone.

4.3.2 Supported optional nodes

All but the most basic functionality of the OPC 40082-2 information model is optional. Despite requests by machine manufacturers for the contrary, this was decided with the goal for most widespread adoption of the standard by also the most simple controllers that may not have user interfaces, for example.

The pro CONTROL controller fully complies with all mandatory parts of the specified information model. The following lists the **optional** nodes supported by pro CONTROL controllers in addition.

4.3.2.1 Identification

The Identification node is implementing the object type IdentificationType specified in the General Types specification OPC 40083. In addition to all mandatory variables and a DeviceClass of “Hot Runner Device” the server provides

Display Name	Access	Value (Example)
SoftwareRevision	R	“4.4.8010.28747”
YearOfConstruction	R	2021

4.3.2.2 Operation

Display Name	Access	
ResetAllErrors	Method	Critical errors such as a defective triac needs to be acknowledged by the user. Otherwise it is not possible to switch on the affected zone again. All valid notifications/alarms will still be listed under ActiveErrors after the method has been called.
ActiveErrors	R	Array with all active errors, grouped by Message (e.g. “Negative temperature deviation”), each element has an array of SourceNodes representing the affected nodes. The referenced node differs per error type: • Temperature-related errors (e.g. high temperature) refer to the Temperature node of the zone • Current-releated errors (e.g. defective triac) refer to the Controller.LoadCurrent node

		• All others refer to the zone node itself The Classification attribute of the error refers to the Status of the respective MonitoredParameterType, adopting its numeric value. For errors referring to the zone node Classification is zero. The Severity of the error reflects the pro CONTROL controller severity info, warning, alarm combined with the order of the notification. This results in a unique Severity per error type. Infos have a severity of less than 334, alarms of more than 666 and warnings are in between.
ActiveSetValues	RW	While the node itself is mandatory, its type definition of MultiStateValueDiscreteType allows only supporting a subset of the values listed in the specification. The suported values are the following: **0** (“First”) = Normal operation **2** (“Standby”) = Controller is in global standby mode **3** (“Boost”) = Controller is in global boost mode

4.3.2.3 Zones

As per OPC 40082-2 all zones are listed as individual object nodes of type ZoneType. Note that an pro CONTROL controller only lists his “own” physical zones. If the controller is part of a group of controllers linked via Ethernet, he will not list zones of the other devices of the group.

The supported optional nodes are as follows.

4.3.2.3.1 Name

Display Name	Access	
Name	RW	The given zone name is displayed at pro CONTROL user interface. An empty string will display the default name “Zone <*x*>”.

4.3.2.3.2 Controller

Display Name	Access	Value (Example)
ActualOutput	R	Actual output level of the zone in % (eg. 12,3)
AverageControllerOutput	R	The average output level calculated during normal operating mode.

UpperOutput	RW	The maximum allowed output level (0..100%)
ReferenceZone	RW	Number of reference zone. In reference mode the output level of the reference zone is adopted. This allows to control multiple heaters with a single sensor.
SetValueManualOutput	RW	During manual operation, a constant output level will be maintained at the heating output (0..100%)
SetValueType	RW	While the node itself is mandatory its type definition of MultiStateValueDiscreteType allows only supporting a subset of the values listed in the specification. The supported values are the following. **0** (CLOSED_LOOP_CONTROL) **1** (MANUAL) **2** (SYNCHRONOUS_ZONE) **5** (MEASURING_ZONE) **6** (NOT_USED)
SetValueManualOutput	RW	During manual operation, a constant output level will be maintained at the heating output (0..100%)
SetValueActive	RW	If SetValueActive is set to false, the output of the zone will be disabled regardless of the operation mode. “IMM: off” is displayed at pro CONTROL user interface.
ActualValueActive	R	Reports the status of SetValueActive considering to the global controller enable.

<table>
<tr><td rowspan="6">LoadCurrent</td><td colspan="3">Additional supported nodes of this MonitoredParameter-Type for current monitoring. Only Lower-/UpperTolerances are supported instead of Min/MaxValue because current monitoring only signals a warning, not an alarm.</td></tr>
<tr><td>Actu-alValue</td><td>R</td><td>Actual heating current in Ampere</td></tr>
<tr><td>SetValue</td><td>RW</td><td>The heating current to be monitored can be specified here. Any deviation is calculated on the basis of this refer-ence value. Entering a setting of 0.0 A will switch off the supervision.</td></tr>
<tr><td>Lower-Toler-ance</td><td>RW</td><td rowspan="2">Note, that the controller does not allow for differing lower/upper tolerances. When changing one variable the other will also be changed accordingly.</td></tr>
<tr><td>Upper-Toler-ance</td><td>RW</td></tr>
<tr><td>Status</td><td>R</td><td>The type definition of MultiStateValueDiscreteType allows only supporting a subset of the values listed in the specification. The sup-ported values are the following:

0 (NONE)
when monitoring is off
1 (UNKNOWN)
when a current-related alarm is sig-naled (e.g. fuse defective)
4 (BELOW_LOWER_TOLERANCE)
5 (WITHIN_TOLERANCE)
6 (ABOVE_UPPER_TOLERANCE)</td></tr>
</table>

4.3.2.3.3 Temperature

Note that all temperatures are reported in degree Celsius.

SetValue	RW	Temperature setpoint (0..750°C)
ActiveSetValue		While the node itself is mandatory its type definition of MultiStateValueDiscreteType allows only supporting a subset of the values listed in the specification. The supported values are the following: **0** (FIRST) **2** (STANDBY) **3** (BOOST)

MinValue	RW	Low Temperature Limit. If the actual temperature falls below the limit value set here, a LOW-alarm is generated.
MaxValue	RW	High Temperature Limit. If the actual value exceeds the limit value set here, a HIGH-alarm is generated.
LowerTolerance	RW	For temperature monitoring, a tolerance range can be specified below the setpoint. If the actual temperature is below the lower tolerance range, this will be signaled as a warning.
UpperTolerance	RW	For temperature monitoring, a tolerance range can be specified above the setpoint. If the actual temperature is above the upper tolerance range, this will be signaled as a warning.
Status	R	**0** (NONE) When SetValueType is NOT_USED or the zone sensor is switched off **1** (UNKNOWN) When a temperature-related alarm is signaled (e.g. broken sensor) **2** (BELOW_MIN_VALUE) **4** (BELOW_LOWER_TOLERANCE) **5** (WITHIN_TOLERANCE) **6** (ABOVE_UPPER_TOLERANCE) **8** (ABOVE_MAX_VALUE) When either high temperature or shut-off temperature is exceeded
BoostSetValue	RW	The pro CONTROL device itself works with a boost offset based on SetValue, but OPC 40082-2 requires this to be an absolute value. While this is supported, be aware that changing SetValue may implicitly change the boost offset of the controller to keep the BoostSetValue stable. Furthermore, there are edge-cases that may result in unexpected behavior (e.g. when setting BoostSetValue below the SetValue with results in a boost offset of zero and implicitly reducing SetValue as well).
BoostTime	RW	Boosttime is set in milliseconds. Therefore the input range is from 1000 … 900.000. A setting of 0 is permissible for infinite boost.
StandbySetValue	RW	The temperature setpoint during standby mode.

EWIKON
Heißkanalsysteme GmbH
Siegener Straße 35
35066 Frankenberg
Tel: +49 6451 / 501-0
Fax: +49 6451 / 501 202
E-mail: info@ewikon.com
www.ewikon.com

 EWIKON 01/2023

EWIKON