OPERATING INSTRUCTIONS
Option of the Control

INDEX

Free Programmable Interface

INDEX Multi spindle turning machines

Control systems INDEX C200-sl

DIM075EN - 2017-02-10
0217 ec - Frei programmierbare Schnittstelle

Note on applicability

Illustrations in this publication may deviate from the product supplied. Errors and omissions due to technical progress expected.

A word on copyright

Contents

General Information

The freely programmable interface serves to adapt additional units to the machine.

Examples:
- Workpiece looading and unloading attachment
- Measuring station
- Washing station
- etc.

The interface comprises two parts:

1. Software
All addresses (inputs, outputs, flags etc.) can be accessed via this interface.

2. Input-/Output-Module
16 inputs/outputs at distributor cabinet exterior (SEL.-Nr. 76235)

Electrical Design (OPTION - Input-/Output-Module)

Interface signal communication between the units happens via individual 24V-signals.

The signals are,

- **potential free**

  The output signals are available either as make contacts via relay contacts or directly from the input-/output-module (breaking capacity 24V DC, 1A).
  They must be switched via relay contacts in order to separate INDEX system and USER system voltage.
- **to be provided with a definite interface description by the user**

  Description of the signal contents as well as of the signal-time diagram.

Via this function which is offered for all machine types it is possible to set and to re-set 16 outputs in the parts programme.
Via these outputs the user can operate individually attached extra devices (manipulator, cleaning station, conveyor belt, etc.).

Interface programming happens via command Mee = zz 98 aaa b. Output setting/ re-setting can happen dependent on the 16 possible feedbacks of the additional attachments (machine inputs). This means that the outputs are operated dependant on the input status (conditional setting/re-setting).

If the input condition is not corresponding, a reader stop is set - otherwise the next programme block is read. The reader stop has effect on the channel in which the command is programmed.

For input/output addresses see respective circuit diagram.

Functional description

Read or write PLC bits from data modules, inputs, outputs and flags

If PLC process signals are to be linked to the parts program, direct addresses of data modules, inputs, outputs or flags are to be accessed as follows:

Used inputs, outputs, flags, times

To the user are max. available:

16 inputs	E16.0	...	E17.7
16 outputs	A16.0	...	A17.7
16 flags	M16.0	...	M17.7
	M163.0	...	M163.7 = reserved for preselction counter
10 Times	T80	...	T89 (for channel 1...10)

Abbreviations

DPR	DualPortRam → couple memory responsible for the communication between NCK and PLC
DB	data building block
AB	output byte
AW	output word
EB	input byte
EW	input word
LH	reader stop
VKE	link result

Programming

Inputs, outputs, flags, times

Mee = zz98 Haaab

b = Bit-No. (0...7)

aaa = Byte adresse

zz = 0 → LH to VKE = 1
= 1 → LH to VKE in R50 stored
= 2 → LH to VKE in DPR stored
= 3 → stop at end of cycle
= 4 → production stop

ee = 0 → input on 1 check (and)
ee = 1 → output on 1 check (and
ee = 2 → flag on 1 check (and)
ee = 3 → input on 0 check (and)
ee = 4 → output on 0 check (and)
ee = 5 → flag on 0 check (and)
ee = 6 → output setting
ee = 7 → flag setting
ee = 8 → output resetting
ee = 9 → flag resetting
ee = 10 → data bit on 1 check *)
ee = 11 → data bit on 0 check *)
ee = 12 → byte/word information PLC → NC (DPR) *)
ee = 13 → byte/word information NC (DPR) → PLC *)
ee = 14 → start time *)
ee = 15 → reserve
ee = 16 → reserve
ee = 17 → reserve
ee = 18 → reserve
ee = 19 → reserve
ee = 20 → input on 1 check (or)
ee = 21 → output on 1 check (or
ee = 22 → flag on 1 check (or)
ee = 23 → input on 0 check (or)
ee = 24 → output on 0 check (or)
ee = 25 → flag on 0 check (or)

*) See respective section of this manual

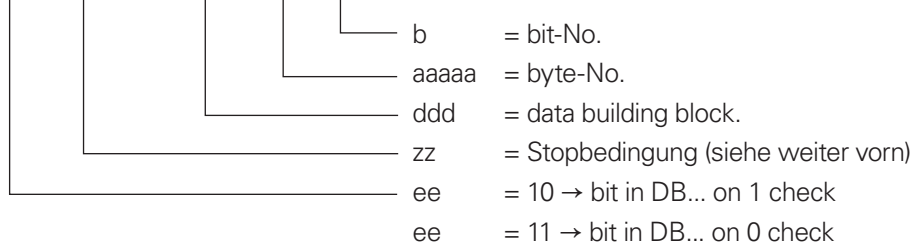
Values area (aaa)

ee = 0 input	on 1 check	aaa = 0....255
ee = 1 output	on 1 check	aaa = 0....255
ee = 2 flag	on 1 check	aaa = 0....511
ee = 3 input	on 0 check	aaa = 0....255
ee = 4 output	on 0 check	aaa = 0....255
ee = 5 flag	on 0 check	aaa = 0....511
ee = 6 output	setting	aaa = 16.... 17
ee = 7 flag	setting	aaa = 16.... 17
ee = 8 output	resetting	aaa = 16.... 17
ee = 9 Flag	resetting	aaa = 16.... 17
ee = 10 data bit on 1 check *)		
ee = 11 data bit on 0 check *)		
ee = 12 byte/word information PLC → NC (DPR) *)		
ee = 13 byte/word information NC (DPR) → PLC *)		
ee = 14 start time		
ee = 15 reserve		
ee = 16 reserve		
ee = 17 reserve		
ee = 18 reserve		
ee = 19 reserve		
ee = 20 input	on 1 check	asaa = 0...255
ee = 21 output	on 1 check	asaa = 0...255
ee = 22 flag	on 1 check	asaa = 0...511
ee = 23 input	on 0 check	asaa = 0...255
ee = 24 output	on 0 check	asaa = 0...255
ee = 25 flag	on 0 check	asaa = 0...511

The setting /resetting of outputs and flags is restricted to the customer's interface Hardware-scope.

Data building blocks (enquiry of individual bits from a DB (data building block))

Mee = zz98 Hdddaaaaab



*) See respective section of this manual

Reading byte/word information from the input/output/ flag/DB-Area and put into the DPR: PLC → NC(DPR)

In the DPR a word (512-531) was defined for every channel in order to put byte/word information of inputs, outputs, markers and data building blocks into the DPR.
Byte information are filed in the high-byte (513, 515, 517 ... 531) , thus the low-byte (512, 514, 516 ... 530) can further be used for the evaluation of a started time.
If a word is put into the DPR, no additional time can be started and no enquiry link can be evaluated, since in such case the address areas would overlap in the low-byte.
Any address can be put in to the DPR.

M12 = zz98 Hdddaaaaa

aaaaa = Only with a data building block necessarily (byte adresse)

ddd = With a data building block the block number, otherwise the byte adresse

zz =	0	input byte	(ddd → 1-255 / aaaaa → can be void)
	1	output byte	(ddd → 1-255 / aaaaa → can be void)
	2	flag byte	(ddd → 1-511 / aaaaa → can be void)
	3	DB byte	(ddd → 1-999 / aaaaa → 00001-99999)
	4	input word	(ddd → 1-255 / aaaaa → can be void)
	5	output word	(ddd → 1-255 / aaaaa → can be void)
	6	flag word	(ddd → 1-511 / aaaaa → can be void)
	7	DB word	(ddd → 1-999 / aaaaa → 00001-99999)

Important!

Data building block number	always 3-digit enter
DB byte number	always 5-digit enter, i.e. with leading zeros

Example WTR (tool carrier) 11: enquire byte information and start time

EB16

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
(1)	(1)	(0)	(0)	(1)	(0)	(0)	(1)

```
M12=098H16 M14=98H100      // read input byte 16 and write into DPR
                           // + start time of 10 seconds
```

$A_DBB[512]

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
free	free	free	free	free	time running	free	free

I_TIME11 → channel 1.1

$A_DBB[513]

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
(1)	(1)	(0)	(0)	(1)	(0)	(0)	(1)

Abfragen:

```
if ((I_TIME11)==2) GOTOF…          //while the time is running go to…
if (($A_DBW[513])=='B11001001') GOTOF…  //as soon as bit 0,3,6,7 =true, go to…
```

Example WTR 11: enquiry of word information

EW16

Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
(0)	(0)	(0)	(0)	(0)	(0)	(1)	(1)	(0)	(0)	(0)	(0)	(1)	(0)	(0)	(1)
EB17								EB16							

```
M12=498H16      // read input word 16 and write into DPR
```

$A_DBW[512]

Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
(0)	(0)	(0)	(0)	(0)	(0)	(1)	(1)	(0)	(0)	(0)	(0)	(1)	(0)	(0)	(1)
$A_DBB[513]								$A_DBB[512]							

Abfragen:

```
IF (($A_DBW[512])=='B0000010000000101') GOTOF…  // as soon as bits 0,2,10 =
                                                true go to…
MSG("$A_DBW[512]="<< $A_DBW[512])               // input $A_DBW[512] with
                                                contents EW16
```

Get byte/word information from the DPR and write into the output/flag area: NC(DPR) → PLC

A second area (862 - 881) of one word per channel each was defined in the DPR, in order to be able to set/delete outputs and markers word/byte by word/byte from the NC. **For this function no other than the outputs / markers of the customer interface are available.** For safety reasons, not any area can be written on.

M13 = zz98 Hddd

ddd = word/byte adresse

zz =
- 0 output byte (ddd → 16...17)
- 1 flag byte (ddd → 16...17)
- 2 output word (ddd → 16)
- 3 flag word (ddd → 16)

Example WTR11: reserve DPR area and set output byte accordingly

```
$A_DBB[862]='B00001001'          // $A_DBB[862] reserve
```

$A_DBB[862]

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
(0)	(0)	(0)	(0)	(1)	(0)	(0)	(1)

```
M13=098H16                       // Get data from the reserved DPR byte and set
                                    the output byte 16 accordingly
```

AB16

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
(0)	(0)	(0)	(0)	(1)	(0)	(0)	(1)

Example WTR11: reserve DPR area and set output byte accordingly

```
$A_DBW[862]='B0000001100001001'   // $A_DBB[862] reserve
```

$A_DBW[862]

Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
(0)	(0)	(0)	(0)	(0)	(0)	(1)	(1)	(0)	(0)	(0)	(0)	(1)	(0)	(0)	(1)
$A_DBB[863]								$A_DBW[862]							

```
M13=298H16                        // Get data from the reserved DPR byte and set
                                  the output byte 16 accordingly
```

AW16

Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
(0)	(0)	(0)	(0)	(0)	(0)	(1)	(1)	(0)	(0)	(0)	(0)	(1)	(0)	(0)	(1)
AB17								AB16							

Group division according to functions

Group 1

ee = 6 → output setting
ee = 7 → flag setting
ee = 8 → output resetting
ee = 9 → flag resetting

ee = 12 → word/byte informations reading
ee = 13 → word/byte informations writing
ee = 14 → start time

Group 2

ee = 0 →	input	on 1 check (and)
ee = 1 →	output	on 1 check (and)
ee = 2 →	flag	on 1 check (and)
ee = 3 →	input	on 0 check (and)
ee = 4 →	output	on 0 check (and)
ee = 5 →	flag	on 0 check (and)
ee = 20 →	input	on 1 check (or)
ee = 21 →	output	on 1 check (or)
ee = 22 →	flag	on 1 check (or)
ee = 23 →	input	on 0 check (or)
ee = 24 →	output	on 0 check (or)
ee = 25 →	flag	on 0 check (or)

Group 3:

ee = 10 bit in DB... on 1 check
ee = 11 bit in DB... on 0 check

Max. 3 functions of one group can be programmed.

Times

You can enter a time value from 1 through 999 (100 ms increments). As long as the time is running, the time bit (bit1) of the respective channel is set.

DB98.DBB912 (channel 1) → DPR 512

Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
VKE	time running	free	free	free	free	free	free

M14 = 98 Hxxx

xxx = time in 1/10 s.

start time

Time running (I_TIMExx)

I_TIMExx = 0 → time not running
= 1 → time running

xx = NC channel, e.g. **I_TIME11** = NC channel 1.1

Programming allowed in each channel.
Same syntax in all channels.

Read/Write Byte-/Word Informationen from PLC

I_RBYTE11 ... I_RBYTE64	Read 1 Byte on Kanal 1.1...6.4 from DPR (=$A_DBB513, 515, ... 531)
I_RWORD11 ... I_RWORD64	Read 1 Wort on Kanal 1.1...6.4 from DPR (=$A_DBW512, 514, ... 530)
I_WBYTE11 ... I_WBYTE64	Write 1 Byte for Kanal 1.1...6.4 in DPR (=$A_DBB863, 865, ... 881)
I_WWORD11 ... I_WWORD64	Write 1 Wort for Kanal 1.1...6.4 in DPR (=$A_DBW862, 864, ... 880)

Enquiry links

Functions of group 2 can be linked “and/or”.
The processing of the linking is dependent on the position of the function within the block, i.e. processing from left to right.

Syntax:

	1st value	2nd value	3rd value
N110	M0=198 H171	M20=198 H172	M3=198 H1225
Meaning:	and e17.1=1	or e17.2=1	and e122.5=0

A linking must always start with "and".

Possible linkings:

and	and	and
and	and	or
and	or	and
and	or	or

Linking results (I_Resultxx)

I_Resultxx = 0 Linking has not been accomplished
= 1 Linking has been accomplished

xx = Cross slide, e.g. I_Result35 (for NC channel 3.5)

Reader stop

Reader stop only with functions of the group 2 or 3.
Reader stop is influenced by the input (zz) and by the VKE.

zz = 0 reader stop till VKE =1
= 1 reader stop till VKE is stored in R50

Output behavior

The behavior of the outputs at power on / reset can be projected separately for each output.

Possible behavior:

a. Output is cleared
b. Output is set
c. Output stays unchanged, or the output will be set to switch-off state at power on, respectively.

Input:

C200-4D The input is performed in the operating area parameters, work piece SD, assignments.

C200-sl Operating branch → **Parameters** → User settings→ **Process** → **Free programmable interface**

- Assignment at reset:
  Input = 0 → output is cleared
  Input = 1 → output is set
  Input = 2 → output stays unchanged

- Assignment at power on:
  Input = 0 → output clearing
  Input = 1 → output setting
  Input = 2 → state as before switching off

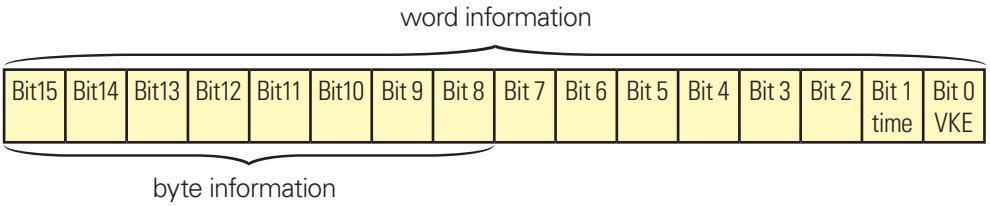
NCK-PLC communication

The communication between NCK and PLC is via the coupling memory (dualportram short form. DPR).

- **Magnitude of DRP 1024 byte.**

- **PLC → NCK Assignment DPR**
  Dbw 512 → Interface channel 1
  Dbw 514 → Interface channel 2
  Dbw 516 → Interface channel 3
  Dbw 518 → Interface channel 4
  Dbw 520 → Interface channel 5
  Dbw 522 → Interface channel 6
  Dbw 524 → Interface channel 7
  Dbw 526 → Interface channel 8
  Dbw 528 → Interface channel 9
  Dbw 530 → Interface channel 10

- **Assignment channel interface PLC → NCK**
  (all channels same assignment)



VKE and state of times are stored in DPR and can be read from there by the part program.

VKE = 0 → linking has not been accomplished.
= 1 → linking has been accomplished.

ZEIT = 0 → time has elapsed
= 1 → time running

Byte information are between bit 8 and bit 15. VKE and TIME-bit are maintained.

Word information are between bit 0 and bit 15. VKE and TIME-bit are overwritten.

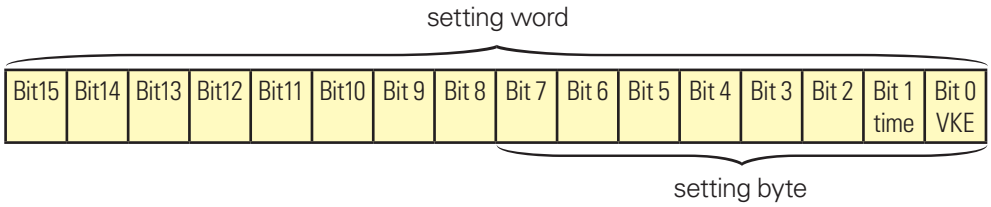
Checking of these variants in the part program releases a STOPRE.

- **NCK → PLC assignment DPR**

  Dbw 862 → Interface channel 1
  Dbw 864 → Interface channel 2
  Dbw 866 → Interface channel 3
  Dbw 868 → Interface channel 4
  Dbw 870 → Interface channel 5
  Dbw 872 → Interface channel 6
  Dbw 874 → Interface channel 7
  Dbw 876 → Interface channel 8
  Dbw 878 → Interface channel 9
  Dbw 880 → Interface channel 10

- **Assignment channel interface NCK → PLC**

  (all channels same assignment)



Setting error messages

Mzz93 Hnnnbb

zz = 0 → Channel Stop (only 1 channel. Clearable with error clearing key.)
= 1 → Channel Stop (only 1 channel. Clearable with reset.)
= 2 → Only error message
= 3 → Stop at end of cycle (all channels)
= 4 → Stop immediately (all channels)
= 5 → Position stop (channels of this position: immediate stop, all other channels: stop at the end of the cycle)

Values area: **nnn** = Units-No. 000...999

The leading 7 is automatically added to the units number.

bb = counting number error 01...64 of the unit

zz = **only 0...5 allowed**
(other inputs will lead to error messages)

Example 1:

N110 M293 H69001	programmed in channel 1 error display, effect: **none**

Anzeige: **769001 customer interface error WTR11 M293 H...**

Example 2:

N110 M393 H69034	; programmed in channel 1 error display, effect: **stop at end of cycle**

Anzeige: **769034 customer interface error WTR11 M393 H...**

Definition of free error texts

At the operating panel you can modify error messages respectively define free error texts. For navigation see document "Operating the machine", section "Editing alarm texts".
Such free error texts are included in the "Kund_gr.com" (German) respectively "Kund_uk.com" (English) file.

Programming examples

Outputs setting/clearing time starting

Example 1:	N110 M8=98 H163 M6=98 H162	clear output 16.3 set output 16.2
Example 2:	N110 M6=98 H171 M14=98 H10	set output 17.1 time starting 1 sec.

Reader stop (inputs, outputs, flags)

Example 1:	N100 M0=98 H1211	reader stop until Key X+ is pressed
Example 2:	N100 M0=98 H421	reader stop until gripper is outside (E42.1=1)
Example 3:	N100 M0=98 H420 M0=98 H423	reader stop until gripper is outside (E42.0=1) and close gripper (E42.3=1)
Example 4:	N100 M4=98 H02	reader stop when conveyor 1 is off (A0.2=0)

Stop at end of cycle (inputs, outputs, flags)

Example:	N100 M3=398 H155	stop at end of cycle when coolant is missing (E15.5=0)

Production stop (inputs, outputs, flags)

Example:	N100 M3=498 H07	production stop when conveyor 1 is disrupted (E0.7=0)

Checking and jumping (inputs, outputs, flags)

Example 1: Checking in channel 1.1

N100 MARK1: M0=298 H1217	state key X+ → DPR
N103 IF((I_Result11)==0) GOTOB MARK1	jump to MARK1, when key X+ has not been operated

Example 2: Programm im Kanal 3.2

N110 M6=98 H171 M14=98 H20	output 17.1 = 1
	time starting 2 sec
N111 MA1: M0=298 H171	input 17.1 → DPR
N112 IF((I_Result32)==1) GOTOF MA2	
N113 IF((I_TIME32)==2) GOTOF MA1	
N114 M8=98 H1711	output 17.1 = 0
N115 GOTOF ME	
N120 MA2: G0 X10 Z200	
N121	
N.....	
N300 ME: M30	end of programm

Description:
N110 set output A17.1 checking time start
N111 check input E17.1
N112 when input E17.1 = 1 then to block 120
N113 as long as checking time is running after block 111
N114 clear output A17.1
N115 jump to program end

N120 E17.1 became 1--> continue working
N121 ...

Example 3

Action: – enquiry of conveyor belt 1 (TB1) in channel 61:
– TB1 ON → input 16.2 customer interface ON
– TB1 OFF →output 16.2 customer interface OFF
– M68.0 = 1 = TB1 ON

N110 M5=98 H680	enquire M68.for "0"
N111 IF((I_RESULT61)==1) GOTOF MA10	if TB1 OFF → MA10
N112 M6=98 H162	set output 16.2
N113 MA10:	
N120 M2=98 H680	enquire M68.for "1"
N121 IF((I_RESULT61)==1) GOTOF MA11	if TB 1 OFF → MA11
N122 M8=98 H162	delete output 16.2
N123 MA11:	

Programming AND - AND - AND

Linkage		
1st value (AND)	2nd value (AND)	3rd Value (AND
M0=298 H170	M0=298 H171	M0=298 H172
⌒	⌒	⌒
(AND)	(AND)	(AND)

Linkage is fulfilled, if all three keys are pressed simultaneously.

Example

```
;#34MP
N100 L100
N200 GX73
N210 GZ73
N220 M72
N300 MA1:M73
N310 IF I_ZYKSTO GOTOF MA2
N320 G59 X=XMW_3 Z=ZMW_3
N400 M0=298 H170 M0=298 H171 M0=298 H172
N410 IF NOT ((I_RESULT34)==1) GOTOF no_alarm
N420 SETAL (67000)
N430 no_alarm:
```

Notes

Block 400: Enquire input 17.0 on “one” “and” enquire input 17.1 on “one” “and” enquire input 17.2 on “one”.

Block 410: If not all inputs signal “one”, skip to “no alarm”. In case all inputs signal “one”, continue programme.

Programming AND - AND - OR

Linkage		
1st value (AND)	2nd value (AND)	3rd value (OR)
M0=298 H173	M0=298 H174	M20=298 H175
⌒	⌒	⌒
(AND)	(AND)	(OR)

Linkage is fulfilled, if the first and the second input signal „one“ (i.e. if voltage is applied), or if the third value is „one“.

If linkage is fulfilled, result in tool carrier 3.5 becomes true (I_Result35)==1.

This means that the jump function in block 60 is carried out and error message of the free programmable interface is ***not*** *issued.*

Example

```
;#35MP
N10 G173
N15 L100
N20 GZ73
N25 G173
N30 GX73
N35 M72
N40 MA1:M73
N45 IF I_ZYKSTO GOTOF MA2
N50 G59 X=XMW_3 Z=ZMW_3 Z3=ZMW_3
;ZYKLISCHE BEARBEITUNG ANWENDER
N55 M0=298 H173 M0=298 H174 M0=298 H175
N60 IF ((I_RESULT35)==1) GOTOF No_Alarm
N65 M393 H69432
N70 No_Alarm:
```

Programming AND - OR - AND

Linkage		
1st value (AND)	2nd value (OR)	3rd value (AND
M0=298 H1217	M0=298 H1225	M20=298 H1226
⌒	⌒	⌒
(AND)	(OR)	(AND)

The linkage is fulfilled, if either (E121.7 and E122.6) or (E122.5 and E122.6) signal „one“ (i.e. voltage is applied).

In this case, the error message of the free programmable interface is not issued.

Example

```
;#41MP
N100 L100
N200 GX73
N210 GZ73
N220 M72
N300 MA1:M73
N310 IF I_ZYKSTO GOTOF MA2
N320 G59 X=XMW_1 Z=ZMW_1
;ZYKLISCHE BEARBEITUNG ANWENDER
N55 M0=298 H1217 M20=298 H1225 M0=298 H1226
N60 IF ((I_RESULT41)==1) GOTOF No_Alarm
N65 M393 H69431
N70 No_Alarm:
```

Programming AND - OR - OR

Linkage		
1st value (AND)	2nd value (OR)	3rd value (OR)
M0=298 H1217	M20=298 H1225	M20=298 H1226
⌒	⌒	⌒
(AND)	(OR)	(OR)

The linkage is fulfilled, if at least one of these inputs signals „one“ (i.e. voltage is applied)

Example

```
;#42MP
N100 L100
N200 GX73
N210 GZ73
N220 M72
N300 MA1:M73
N310 IF I_ZYKSTO GOTOF MA2
N320 G59 X=XMW_1 Z=ZMW_1
;ZYKLISCHE BEARBEITUNG ANWENDER
N55 M0=298 H1217 M20=298 H1225 M20=298 H1226
N60 IF ((I_RESULT41)==1) GOTOF No_Alarm
N65 M393 H69431
N70 No_Alarm:
```

INDEX

INDEX-Werke GmbH & Co. KG
Hahn & Tessky

Plochinger Straße 92
D-73730 Esslingen

Fon +49 711 3191-0
Fax +49 711 3191-587

info@index-werke.de
www.index-werke.de