

## **Freely programmable interface**

### **General**

The freely programmable interface is used to adapt additional devices to the machine.

#### **Examples:**

- Loading gantry
- Measuring station
- Washing station
- etc.

#### **Software interface**

This interface allows you to query, set and reset any addresses (inputs, outputs, flags, etc.).

## Functional description

### Reading or writing PLC bits from/to data modules, inputs, outputs, and flags

To connect process signals of the PLC with the workpiece program, direct addresses of data modules, inputs, outputs or flags must be addressed. This is done by:

#### Input, output and flag

##### **Mi = j 98 Hyyyz**

<b>i</b>	<b>0</b>	E	= 1?	] Query
	<b>1</b>	A	= 1?	
	<b>2</b>	M	= 1?	
	<b>3</b>	E	= 0?	
	<b>4</b>	A	= 0?	
	<b>5</b>	M	= 0?	] Set
	<b>6</b>	A	= 1!	
	<b>7</b>	M	= 1!	
	<b>8</b>	A	= 0!	
<b>9</b>	M	= 0!		

**j**    **0** = Reader stop until condition fulfilled  
       **1** = Reader stop until result is loaded into R50

**yyy** Address (byte)



**z**    Bit 0 - 7

#### Remarks:

A = Output  
 E = Input  
 M = Flag

## Data module

### **Mii = j 98 Hxxxyyyyyz**

- ii**     **10** D = 1?     Query  
         **11** D = 0?      
         **12** D = 1!    Set  
         **13** D = 0!    Reset
- j**        **0** = Reader stop until condition fulfilled  
          **1** = Reader stop until result is loaded into R50
- xxx**     Data module
- yyyyy**   Address (byte)
- z**        Bit 0 - 7

### Remarks:

D = Data bit



The commands can be programmed in each channel. When Mi = 198, the current signal state (logical "1" or "0") is loaded into parameter R50. Before parameter R50 is queried in the workpiece program, the buffer must be cleared with the STOPRE command.

## Reading or writing PLC bits from/to data modules, inputs, outputs, and flags

### Data module, data bytes, data words

#### Remarks:

DBD = Data double word, DBB = Data byte, DBW = Data word

Data module (DM) numbers up to 999

#### **Mii = 1 98 Hxxxyyyyy0**

ii	<b>30</b>	DBB to R50	PLC → NC	DUAL
	<b>31</b>	DBW to R50	PLC → NC	DUAL
	<b>32</b>	DBD to R50	PLC → NC	DUAL
	<b>33</b>	DBB from R50	NC → PLC	DUAL
	<b>34</b>	DBW from R50	NC → PLC	DUAL
	<b>35</b>	DBD from R50	NC → PLC	DUAL
	<b>38</b>	DBD to R50	PLC → NC	REAL
	<b>39</b>	DBD from R50	NC → PLC	REAL
	<b>40</b>	DBB to R50	PLC → NC	BCD
	<b>41</b>	DBW to R50	PLC → NC	BCD
	<b>42</b>	DBD to R50	PLC → NC	BCD
	<b>43</b>	DBB from R50	NC → PLC	BCD
	<b>44</b>	DBW from R50	NC → PLC	BCD
	<b>45</b>	DBD from R50	NC → PLC	BCD
	<b>1</b>	Reader stop until result of PLC is loaded into R50 or of R50 into PLC		
<b>xxx</b>	Data module			
<b>yyyyy</b>	Address (byte)			



**Negative values (-) are allowed for functions 31, 32, 35, 38, 39, 41, and 45 only.**



The commands can be programmed in each channel. When reading with  $M_i = 198$ , the current value is loaded into parameter R50. Before parameter R50 is queried in the workpiece program, the buffer must be cleared with the STOPRE command.

When values are loaded from R50 into a data byte, data word or data double word, STOPRE must be programmed after  $M_i=198$  (see examples below).



**Examples**

**Load data module 111, data byte 3226 into R50.**

```
M30=198 H111032260
STOPRE
IF R50==5 GOTOF END_
```

**Load R50 into data module 214, data word 1010.**

```
R50=700
M34=198 H214010100
STOPRE
```

**Example - Branches in the workpiece program**

**Example:**

Wait until Cycle Start key is pressed

```
M0=98 H1201 ;E120.1=1?
```

**Example:**

Branch to MARK\_1 if flag 78 Bit5=0 follow-up start is off.

```
M2=198 H785 ;Query M78.5=1, result into R50
STOPRE
IF R50<>1 GOTOF LABEL1_ ;Sequence when M78.5=1 -> follow-up start ON
GOTOF END_
LABEL1_ ;Jump address
;Sequence when M78.5=0 -> follow-up start OFF
END_ ;Sequence for both versions;
```

The commands can be programmed in each channel. When Mi=198, the status is output to parameter R50.

Before parameter R50 is queried in the workpiece program, the buffer must be cleared with STOPRE.

**Example:**

Wait until spindle 3 has stopped

```
M10=98 H38000614 (DB38 DBB1 Bit14=1?)
```

**Example:**

Branch to MARKE2\_, if block skip level 2N (DB214 byte1036 bit2) is selected.

```

M10=198 H214010372           ;Query DB214 DBB1037 bit 2=1 result into R50
STOPRE
IF R50==1 GOTOF LABEL2_     ;Sequence when R50=0
GOTOF OFF_
LABEL2_:                     ;Jump address
                           ;Sequence when R50=1
OFF_                          ;Sequence for both versions

```

**Notes on block skip levels**

- DB214 DBB1037 Bit 0 = Block skip level /N
- Bit 1 = Block skip level /1N
- Bit 2 = Block skip level /2N
- Bit 3 = Block skip level /3N
- Bit 4 = Block skip level /4N
- Bit 5 = Block skip level /5N
- Bit 6 = Block skip level /6N
- Bit 7 = Block skip level /7N

**Example – Setting output**

**Set output A55.0**

```

...
...
M6 = 98 H550           ;Set A55.0.
...
...
M8 = 98 H550           ;Reset A55.0.
...
...

```

**Example - Query input**

**Query input E55.0**

```

...
...
M0 = 98 H550           ;Wait until E55.0 = 1.
...
...
M3 = 98 H550           ;Wait until E55.0 = 0.
...
...
M0 = 198 H550          ;Load status of E55.0 into R parameter R50
                        ;If input E55.0 = 0 —> R50 = 0,
                        ;If input E55.0 = 1 —> R50 = 1,
STOPRE

```

## Setting error message

The following commands can be used to set all PLC error messages available in the control:

### **Mx 93 Hyyyyz**

- x** 0 Can be cleared with error clear key
- 1 Can be cleared with reset
- 2 Error message without NC stop
- 9 Reset error message

**yyyy** Error location

**z** Error cause (code)

#### **Error location:**

The error location number must be specified without the leading digit (7) in the error folder.

#### **Error cause:**

The error bit (shown as "x" in the error sheet) is to be programmed as the error cause.

M193 H10092

#### **Error display:**

710092 "WORKPIECE PROGRAM Cycle call not allowed from channel"

For other options, see error descriptions FT71009 and FT71026.

#### **Example:**

Error location

71009	Machine – Part program		
	<b>Error cause</b>	<b>Remedy</b>	<b>Effect</b>
710091	Incorrect cycle input parameter M93 H10091	Correct input parameter acc. to Programming Manual	2/1)
710092	Cycle call not allowed from this channel (M93 H10092)		2/1)
710092	Wrong channel no. in cycle call (M93 H10094)		2/1)

<sup>1)</sup> The error messages set with M293 Hyyyz can be reset with M993 Hyyyz in the following program sequence.



## Workpiece – Sequence monitoring

To avoid collisions on machines with handling systems or feeding/discharge systems, the sequence can be monitored in the workpiece or handling program or in the cycles. The location of workpieces in the system can continuously be saved in data bits.

The system condition can be checked or entered after selection with the softkeys "Operate units → Workpiece manipulator → Workpiece tracking", in the "Workpiece present in" screen.

For example, if you try to get a blank from the pallet although there is a workpiece in the gripper, the handling unit is stopped and an error message is issued.

The following monitoring bits are used depending on the equipment level of the machine:

Bit No.	Bit No.
1 = Workpiece in gripper 1	9 = Workpiece on pallet 1
2 = Workpiece in gripper 2	10 = Workpiece on pallet 2
3 = Workpiece in clamping device 1 (main spindle)	11 = Workpiece in measuring station
4 = Workpiece in clamping device 2 (counter spindle)	12 = Workpiece in SPC drawer
5 = Workpiece on belt 1	13 = Workpiece in washing station
6 = Workpiece on belt 2	14 = Workpiece in turning station 1
7 = Workpiece on belt 3	15 = Workpiece in turning station 2
8 = Workpiece on belt 4	16 = Workpiece in aligning station

Bit = 1 (yes) always means workpiece available  
 = 0 (no) no workpiece available



If additional units are installed in the machine plant, the corresponding monitoring bits are described in the relevant operating instructions.

## Setting or resetting or querying of sequence bits

The sequence bits can be set or reset as explained below.

### **H9 = xx**      **Reset sequence bit**

**xx**      Bit number

### **H9 = 100 xx**      **Set sequence bit**

**xx**      Bit number

### **H9 = 2 xy xx**      **Swap sequence bit**

**xy**      Bit number (reset)

**xx**      Bit number (set)

### **H9 = 500xx**      **Query sequence bit with result in R50**

The sequence bits must be swapped in the same block as the clamping motion.

#### **Programming example:**

Get blank from pallet

**M11=68 H9=2 09 01**      ;Convert sequence bit of pallet 1 in gripper 1

**M11=68**      Close gripper 1

**09**      Reset Bit "9" (no workpiece anymore on pallet 1)

**01**      Set Bit "1" (workpiece in gripper 1)

Reset blank of gripper 1

**M11=69 H9=01**      ;Reset sequence bit of gripper 1

**M11=69**      Open gripper 1

**01**      Reset bit "1" (no workpiece anymore in gripper 1)

**Blank**      Insert into clamping device 1 (main spindle)

**M1=68 H9=100 03**      ;Set sequence bit in main spindle clamping device

**M1=68**      Close main spindle clamping device

**03**      Set bit "3" (workpiece in main spindle clamping device)

**H9=50001**      ;Query sequence bit of 'Gripper 1 on workpiece available

**01**      Transfer bit "1" condition 'workpiece available' in R50

R50 = 1 (yes) always means workpiece available

= 0 (no) no workpiece available



Depending on the equipment level of the machine/system, the monitoring bits of the grippers and units are active and can then be programmed and/or changed by hand.

If a workpiece is manually clamped or unclamped in the event of an error or program abortion, the condition of the system must be corrected in the "Workpiece manipulator -> Workpiece present in" menu screen.

## Checking sequence for collision

The workpiece transfer can be checked for collisions. The system checks that the mounting/unit or the gripper that is to accept the workpiece is free. If the mounting/unit or the gripper is occupied, a corresponding error message is output.

### Call to check mounting/unit:

#### H9 = 3 yy xx

- 3** = Check mounting/unit for accepting the workpiece
- yy** = Bit number (see 'Monitoring bits' table)
- xx** = Bit number of the mounting/unit (see 'Monitoring bits' table)

This checks that the mounting/unit to accept the workpiece is empty. If it is occupied (sequence bit = 1 (yes) workpiece available), an appropriate error message '750031 WP LOADING/FEEDING: from gripper<sup>1)</sup> not possible, as the unit<sup>2)</sup> is occupied' is output.

### Programming example:

**H9 = 30103** ;Check that counter spindle clamping device is free for transfer from gripper 1, otherwise error message

Error message: 750031 WP LOADING/FEEDING: from gripper 1 not possible, as clamping device 1 is occupied

### Call to check gripper:

#### H9 = 4 yy xx

- 4** = Check gripper for accepting the workpiece
- yy** = Bit number (see 'Monitoring bits' table)
- xx** = Bit number of the mounting/unit (see 'Monitoring bits' table)

This checks whether gripper to accept the workpiece is empty. If it is occupied (sequence bit = 1 (yes) workpiece available), an appropriate error message '750041 WP LOADING/REMOVAL: from unit<sup>2)</sup> not possible, as the gripper<sup>1)</sup> is occupied' is output.

### Programming example:

**H9 = 40302** ;Check that counter spindle clamping device is free for transfer from gripper 2, otherwise error message

Error message 750041 WP LOADING/REMOVAL: from clamping device 1 not possible, as gripper 2 is occupied

<sup>1)</sup> gripper number 1 or 2

<sup>2)</sup> unit see assignment in 'Monitoring bits' table





**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](mailto:info@index-werke.de)  
[www.index-werke.de](http://www.index-werke.de)