B&R SYSTEM 2000

B&R 2003
USER'S MANUAL

Version: **4.0 (March 2001)**
Model No.: **MASYS22003-E**

1. GENERAL INFORMATION

2. PROJECT PLANNING AND INSTALLATION

3. B&R 2003 MODULES

4. MODULE ADDRESSING

5. CAN BUS CONTROLLER FUNCTIONS

Contents

6. B&R 2003 TIMING

7. GENERAL ACCESSORIES

8. STANDARDS AND CERTIFICATIONS

A. CAN IDENTIFIER WITH CAN BUS CONTROLLER

B. ERROR MESSAGES CAN BUS CONTROLLER

INDEX

CHAPTER 1 - GENERAL INFORMATION

CHAPTER 2 - PROJECT PLANNING AND INSTALLATION

CHAPTER 3 - B&R 2003 MODULES

CHAPTER 4 - MODULE ADDRESSING

CHAPTER 5 - CAN BUS CONTROLLER FUNCTIONS

CHAPTER 6 - TIMING

CHAPTER 7 - GENERAL ACCESSORIES

CHAPTER 8 - STANDARDS AND CERTIFICATIONS

APPENDIX A - CAN IDENTIFIER WITH CAN BUS CONTROLLER (FIXED ALLOCATIONS)

APPENDIX B - ERROR MESSAGES CAN BUS CONTROLLER

INDEX

CHAPTER 1
GENERAL INFORMATION

1 INTRODUCTION

1.1 GENERAL INFORMATION

The B&R SYSTEM 2000 controller generation is an automation system which sets new standards for performance, functionality and operational security. The B&R 2003, B&R 2005 and B&R 2010 systems cover the entire application range, from simple logic controllers to complex decentralized automation systems.

The systems have different structures, expandability, modularity and CPU performance. These systems are very closely related and therefore meet the fundamental requirements for complete centralized and decentralized compatibility for the programmer.
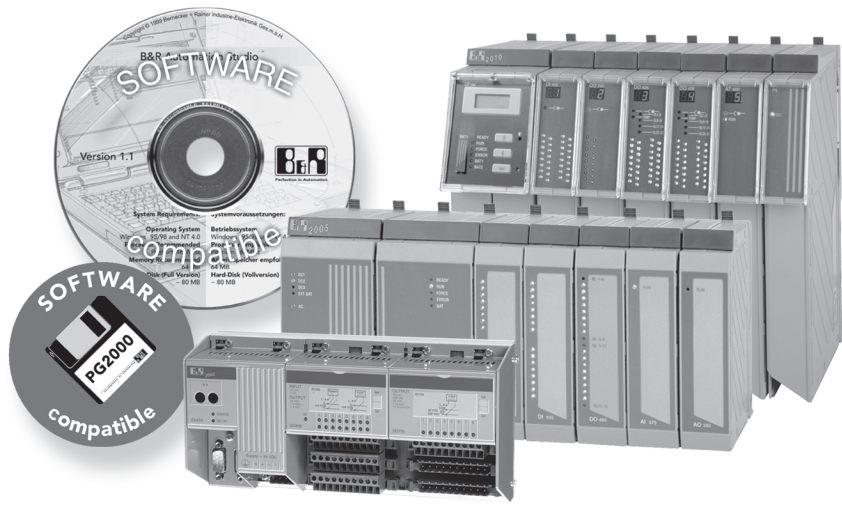
B&R SYSTEMS 2003 Characteristics

Hardware:

- Hardware modularity
- Network capable
- Communication interfaces to HMI
- PCC and industrial computer functionality
- EMC according to IEC 61131-2
- Secure I/O bus protocol
- Remote I/O points
- Bit or word processing in a single cycle
- All terminals for 2 or 3 line connection available directly on the module
- No extra terminals needed

Software:

- Multitasking operating system for PCC and industrial computer applications
- Powerful PCC programming languages
- High level language programming
- Exact control over PCC timing
- Simple programming software with window oriented user interface
- Project management using the programming software

PCC (programmable computer controller)

The core of the PCC consists of powerful standard computer components. The processors are supported by RISC processors which allow fast processing of I/O signals.

Most CPUs use a modular interface concept. Various bus or network systems can be integrated in the B&R SYSTEM 2003 efficiently using plug-in interface modules.

The computer core uses components which provide the functionality of an industrial PCC. Bit, byte and word access is possible in a single cycle which increases speed when combining PCC links and industrial computer functionality.

System Interface

The system interface is the sum of all I/O modules, i.e. the interface between PCC and the machine/system being controlled. All B&R SYSTEMS 2003 I/O modules are protected from external disturbances using suitable EMC measures (IEC 61131-2 standard).

Length and structure of the I/O bus can be adjusted according to the requirements of the machine/system. That means the connection points for I/O signals do not have to be in the same location as the PCC CPU. Using remote I/O bus segments (Remote I/O, CAN I/O) allows the optimal structure to be created for the respective application. I/O modules are installed on the machine/system where they are needed. Cabling to the core of the PCC is reduced to a two conductor or fibre optic cable.

Network Capability

Network capability and the possibility to communicate with systems from other manufacturers have become a must for all industrial controllers. The B&R 2000 PCC family meets many of these requirements using system-wide and multi-system communication possibilities.

Software Concept

Special attention was given to simple operation and programming. Standard PLC programs run cyclically, i.e. the program is repeated in a loop. To increase efficiency, the B&R SYSTEM 2000 also offers the possibility to use various tasks with different cycle times. In this way, critical program sections (e.g. reaction to trigger signals) can be processed faster than the evaluation of slower events (e.g. polling keys).

Programming

The CPU is programmed using the programming system PG2000 or Automation Studio™. Several programming languages are available:

PG2000	Automation Studio™
Statement List (STL)	Automation Basic (previously PL2000)
Ladder Diagram (LAD)	ANSI C
PL2000 High Level Language (structured)	IEC 1131 Ladder Diagram (LAD)
	IEC 1131 Sequential Function Chart (SFC)
	IEC 1131 Structured Text (ST)
	IEC 1131 Instruction List (IL)

2 B&R 2003 CONTROL SYSTEM

2.1 MODULAR STRUCTURE

Controller and I/O modules are hung in the module rack and screwed to a threaded strip which is inserted in the aluminum frame. The electrical connection between the modules is achieved by pushing the modules together (integrated plug and socket connection in the modules).

Mounting the Module Rack:

The module rack is installed using mounting rail (IEC 50022 - 35 x 7.5 mm). This mounting rail is attached conductively to the back wall of the switching cabinet.
In addition to mounting rail installation, the module rack can also be screwed directly onto the back wall of the switching cabinet.
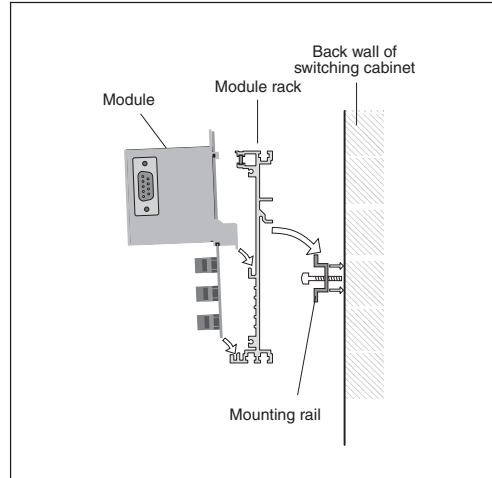
2.2 I/O BUS

The B&R SYSTEM 2003 is equipped with an I/O bus. The electrical connection between the modules is achieved by pushing the modules together (integrated plug and socket connection in the modules).

Characteristics:

- Compact Structure

- Industrial standard bus technology

- Integrated terminals

  - 2 or 3 line connections

  - No extra terminals needed
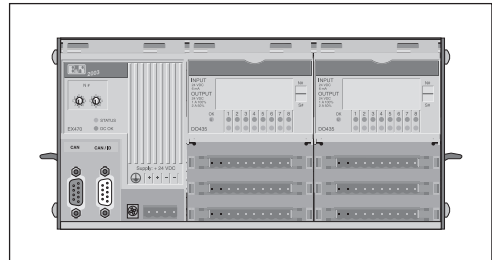
2.3 SUPPLY VOLTAGE

The supply voltage for the B&R 2003 control system is provided by a power supply integrated in the controller.

2.4 TERMINAL BLOCKS

B&R 2003 modules use a large number of terminal blocks to make connections.

The pin assignments for the individual modules can be found in this manual (Chapter "B&R 2003 Modules").

The construction of the terminal blocks has the following advantages for the PCC user:

- Terminal blocks are easy to remove using two ejection levers on the module or directly on the terminal blocks.

- Most terminal blocks are available with screw clamps and also with cage clamps.

2.5 PROGRAM MEMORY MODULES

2.5.1 2003 PCC CPU

When using a B&R 2003 CPU, application programs and the operating system are stored in the CPU.

2.5.2 Remote Slaves

When used as a remote I/O slave, the application programs are stored in the remote master (e.g. EX150). The operating system is programmed in the remote I/O bus controller.

2.5.3 CAN Slaves

When used as a CAN slave, the application programs are stored in the CAN master (e.g. XP152). The operating system is programmed in the CAN bus controller. Configuration data can be placed in configuration memory (inserted from the front).
The CAN bus controller EX270 is equipped with an internal S-EEPROM. Operating system parameters can be stored in this S-EEPROM.

3 B&R 2003 EXPANSION

Expansion can take place in two ways:

- Remote I/O Bus
- CAN Field Bus

3.1 REMOTE I/O BUS

The remote I/O bus allows remote I/O modules to be connected to a remote master. The maximum distance without a repeater is 1200 m. Up to 31 slaves can be connected to a remote master without using a repeater.

3.1.1 Remote Master

The remote master is a B&R SYSTEM 2005, B&R SYSTEM 2010 or B&R SYSTEM 2000 Logic Scanner.

3.1.2 Remote Slave

A remote I/O bus controller EX477 or EX777 is required to add a B&R SYSTEM 2003 as slave in a remote I/O network.

Cabling

In a remote I/O system, remote slaves at distances up to 1200 m are connected with the remote master using a shielded twisted pair cable.

Transfer Distance (without repeater)	
100 kBit/sec	up to 1,200 m
181 kBit/sec	up to 1000 m
500 kBit/sec	up to 400 m
1000 kBit/sec	up to 200 m
2000 kBit/sec	up to 100 m
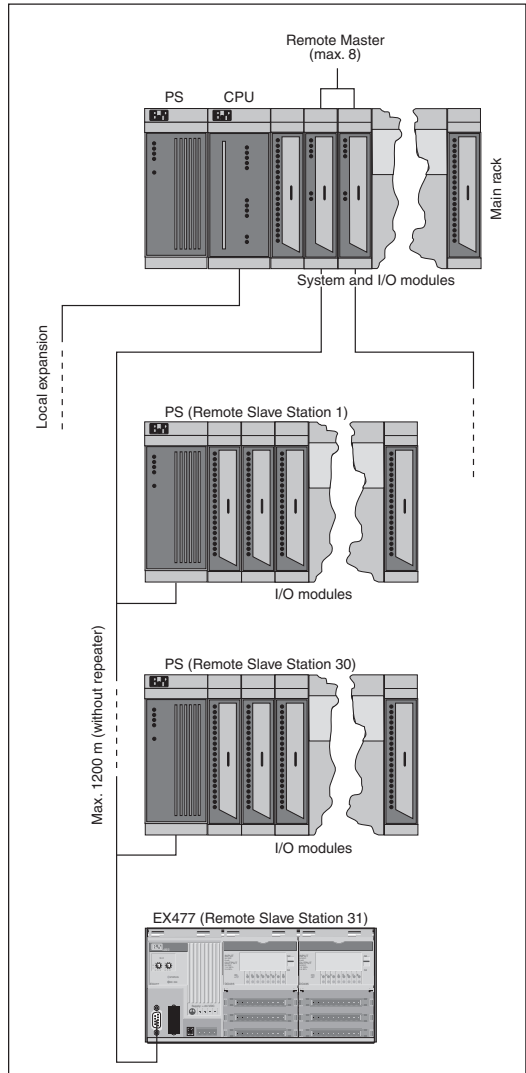
3.2 CAN FIELD BUS

Up to 32 B&R SYSTEM 2003 CAN slaves can be connected to a CAN master. When using ME770 configuration memory or the internal S-EEPROM on the EX270 bus controller, up to 63 B&R SYSTEM 2003 CAN slaves can be connected.

3.2.1 CAN Master

The CAN master is a B&R SYSTEM 2003, B&R SYSTEM 2005, B&R SYSTEM 2010 or B&R SYSTEM 2000 Logic Scanner.

3.2.2 CAN Slave

To add a B&R SYSTEM 2003 to a CAN network, a EX270, EX470 or EX770 CAN bus controller is required.

Cabling

The maximum distance in a CAN field bus system is 1000 m. A 4 conductor twisted pair cable is to be used for the bus cable. The bus length is mostly determined by the transfer rate.

Transfer Distance	
500 kBit/sec	10 -60 m
250 kBit/sec	100 -200 m
50 kBit/sec	800 - 1000 m

CAN Bus Master
CP474

CAN Bus Slave
EX470

CAN Bus Slave
EX470

CAN Bus Slave
EX470
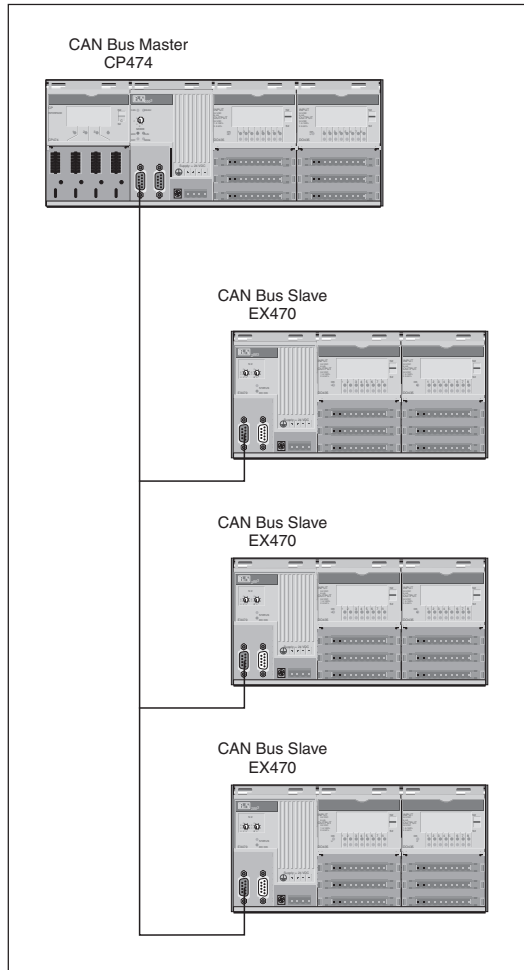
4 COMBINATION POSSIBILITIES

4.1 REMOTE I/O BUS

Up to 31 remote slaves can be connected on a remote master (B&R SYSTEM 2005, B&R SYSTEM 2010 or B&R SYSTEM 2000 Logic Scanner) and the 2003, 2005 and 2010 systems can be combined as desired. A new I/O bus starts with each remote slave. The maximum number of slots are available depending on the slave type.

Slave Type	Number of slots
2010	Max. 99 (daisy chained)
2005	Max. 13
2003	Max. 8

4.2 CAN FIELD BUS

Up to 32 B&R SYSTEM 2003 CAN slaves can be connected to a CAN master (B&R SYSTEM 2005, B&R SYSTEM 2010 or B&R SYSTEM 2000 Logic Scanner). When using ME770 configuration memory or the internal S-EEPROM on the EX270 bus controller, up to 63 B&R SYSTEM 2003 CAN slaves can be connected.

CHAPTER 2
PROJECT PLANNING
AND INSTALLATION

1 DIMENSIONS AND MOUNTING

1.1 DIMENSIONS

All modules in the B&R 2003 system have the same dimensions. The layout for the dimensions of the B&R SYSTEM 2003 is shown in the following diagram.

Dimensions	[mm]
Height	115
Width 7BP7xx.0 7BP70x.1	M x 76.5 + 8.5 M x 76.5 +4.5
Depth	70

M ... Max. number of modules on a module rack (see Module Rack)
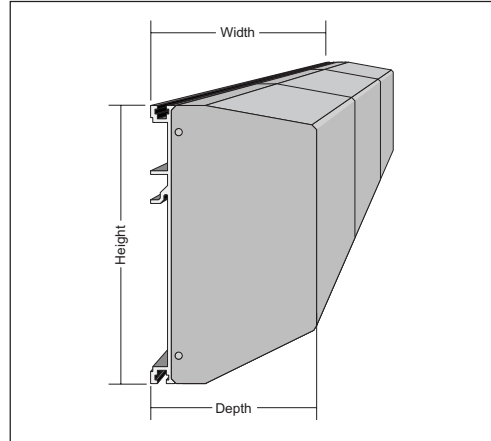
1.2 MOUNTING RAIL

To mount the PCC, you need a mounting rail which conforms to the standard IEC 50022. This mounting rail is to be attached **conductively** to the back wall of the switching cabinet.

Please follow the manufacturers installation guidelines!
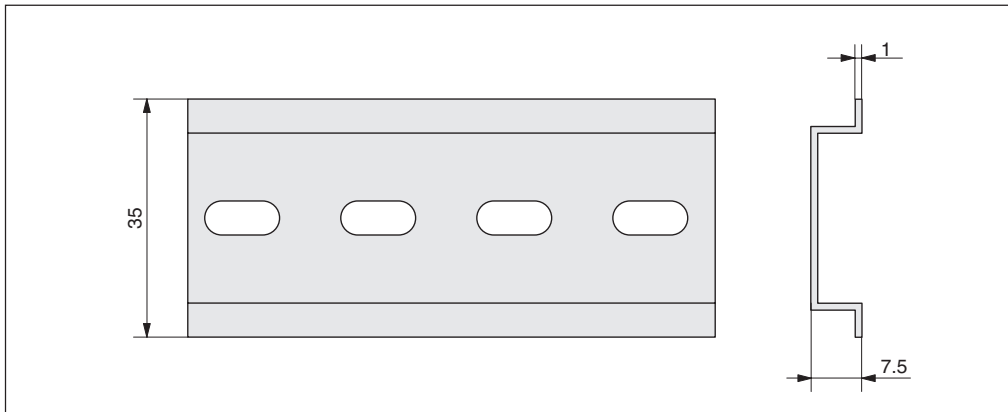
1.3 MODULE RACK

Module racks for the B&R SYSTEM 2003 are aluminum frames which are available in different widths. Module racks with one or two side sections are used depending on the controller.

1.3.1 Example for a Module Rack with Two Side Sections
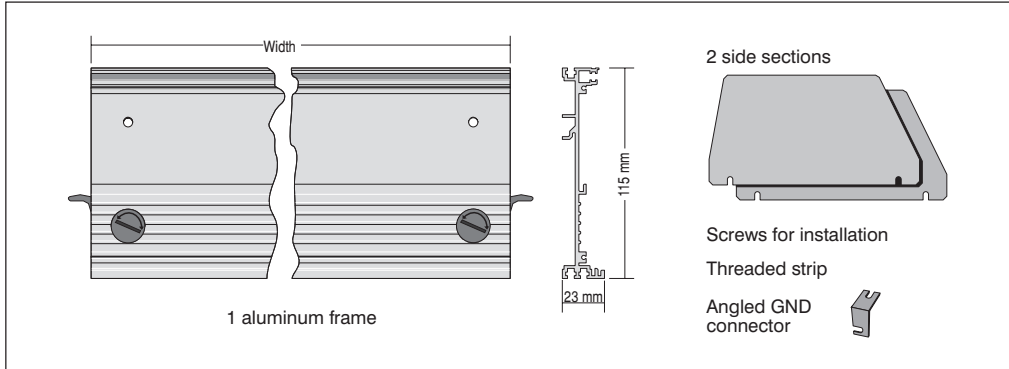
1.3.2 Dimensions

All module racks are 115 mm high. Side sections and mounting screws must be included in the width of the module racks.

Module racks with two side sections (7BP7xx.0):   approx. 8.5 mm
Module racks with one side section (7BP70x.1):   approx. 4.5 mm

Module Rack	Modules	Width [mm]
BP701.1	1	81 [1]
BP702.1	2	157.5 [1]
BP702	2	161.5 [2]
BP703	3	238 [2]
BP704	4	314.5 [2]
BP705	5	391 [2]
BP706	6	467.5 [2]
BP707	7	544 [2]
BP708	8	620.5 [2]
BP709	9	697 [2]
BP710	10	773.5 [2]

[1] Includes 4.5 mm for one side section and mounting screws. These module racks are e.g. used together with the EX270 CAN bus controller.

[2] Includes 8.5 mm for two side sections and mounting screws.

1.4 MODULES

B&R SYSTEM 2003 modules are divided into four groups.

- Controller
- CPUs
- I/O Modules
- Screw-in Modules (ScrM)

The technical data indicates which group the respective modules belong to.

Controller, CPU and I/O modules are hung in the module rack and screwed to a threaded strip which is inserted in the aluminum frame. The electrical connection between the modules is made using a 9 pin D-type plug and socket (simply push the modules together).
The risks surrounding typical ribbon cable, and inserting a module into the wrong slot no longer exist.

Screw-in modules (ScrM) are installed on the adapter module or on the CP interface. Up to four screw-in modules can be installed on each adapter module or CP interface.

1.4.1 DimensionsController, CPUs and I/O Modules

Controller and I/O modules are only available in single widths. CPUs are available in single or double widths. The EX270 CAN Bus Controller is an exception. It is not hung in the module rack, instead it is screwed to the left side of the module rack.

Single width

Labeling field
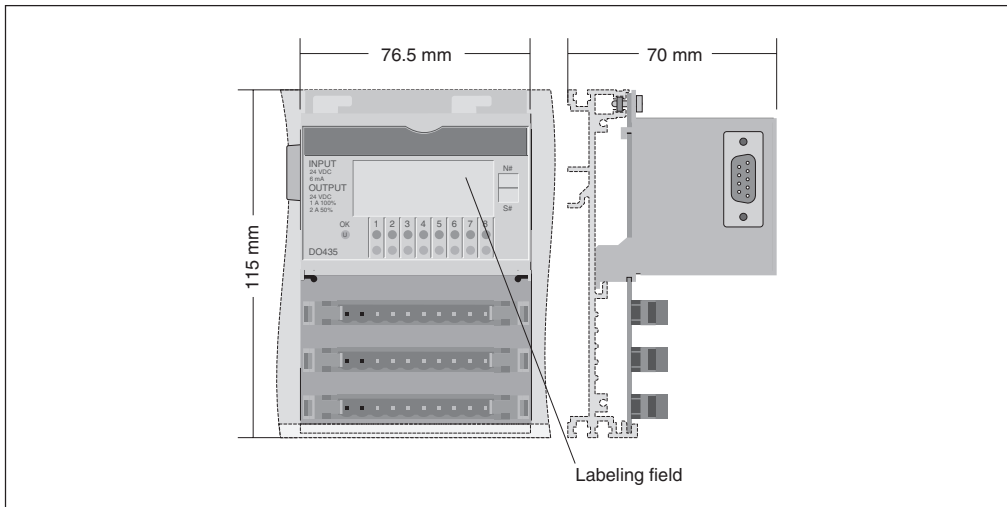
Double width

The CPUs CP474, CP476 and CP774 are double width modules. The depth remains the same as for other modules.

Labeling field
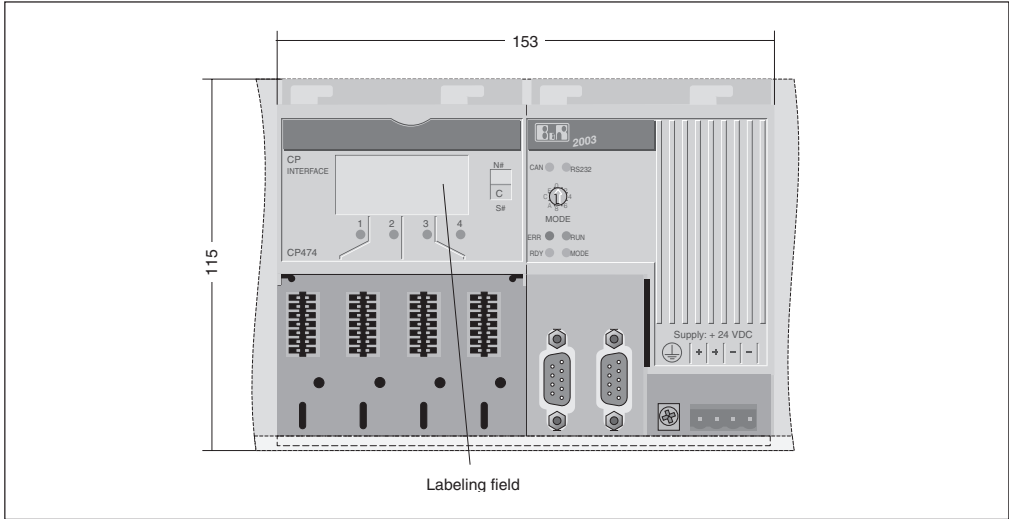
CAN Bus Controller EX270
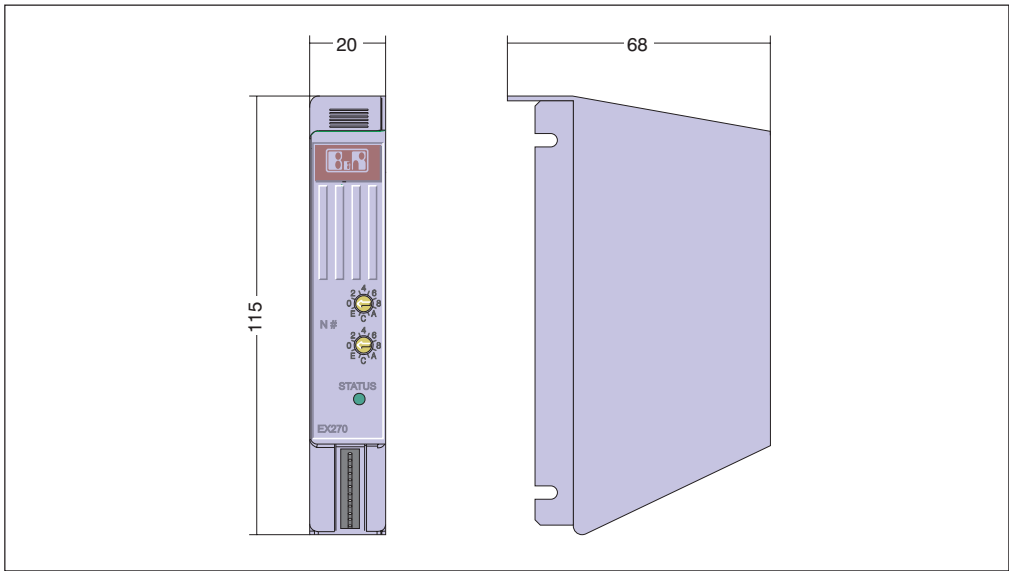
1.4.2 Screw-in Module Dimensions
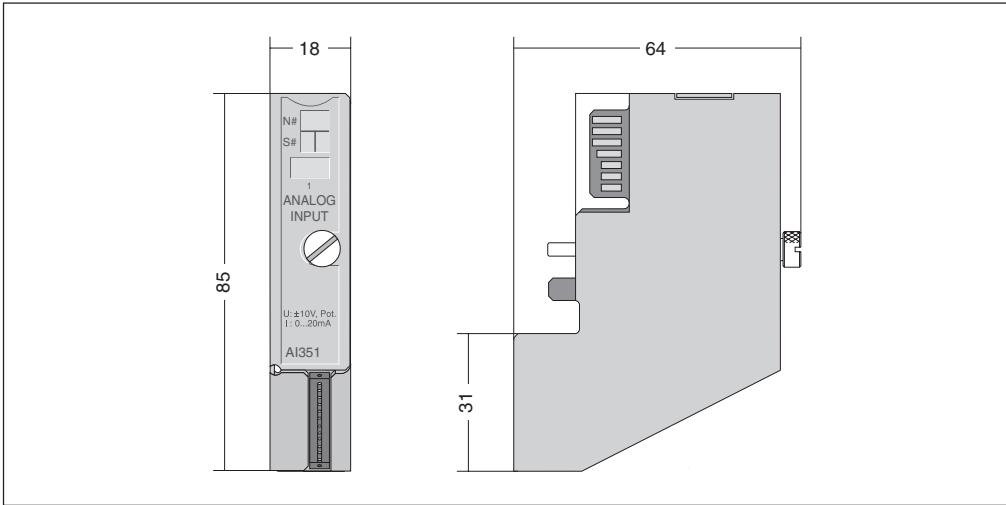
1.5 INSTALLATION DIMENSIONS

1.5.1 Horizontal Installation

Note the to the following dimensions for horizontal installation in a switching cabinet or housing.
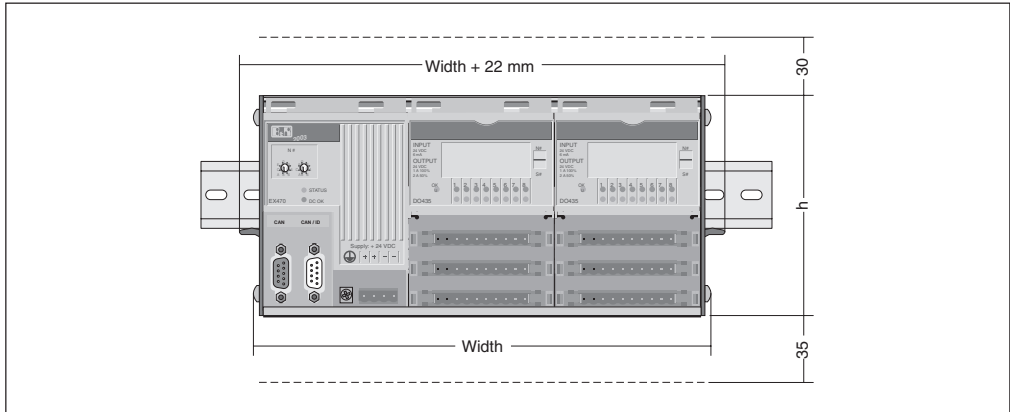
Width: see Section "Module Rack"

Height: h = 115 mm without screw-in modules
            h = 146 mm with screw-in modules

There must be at least 30 mm free space above the modules. The cooling vents are not allowed to be covered.

Underneath B&R SYSTEM 2003, 35mm space is to be left free for the input, output and supply cables.

Standard Installation

If the controller is hung in the module racks, 22 mm must be added to the width listed in section "Module Racks" (column "7BP7xx.0").

Installation with CAN Bus Controller EX270

The CAN Bus Controller EX270 is used together with the module racks having model numbers 7BP70x.1. The EX270 is screwed onto the module rack instead of the left side section.
31 mm is to be added to the width listed in section "Module Rack" (column "7BP70x.1").

1.5.2 Vertical Installation

Note the to the following dimensions for vertical installation in a switching cabinet or housing.
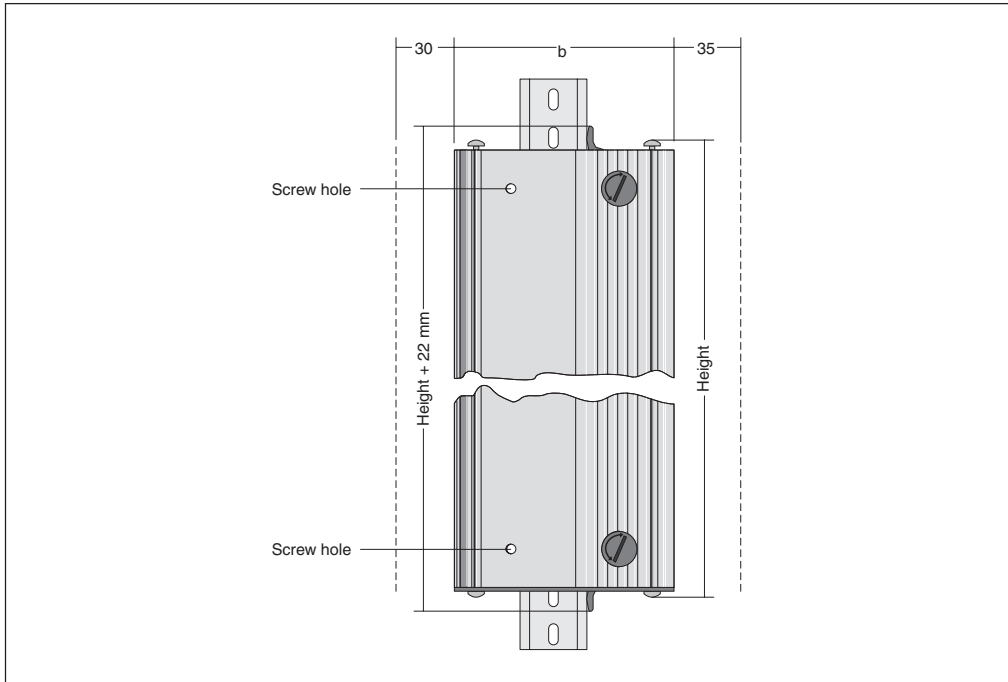
Height: see section "Module Rack", column "Width"

Width: b = 115 mm without screw-in modules
          b = 146 mm with screw-in modules

At least 30mm space must be left free on the left side of the module. The cooling vents are not allowed to be covered.
On the right hand side of the B&R SYSTEM 2003, 35mm space is to be left free for the input and output cables.

The CPU is held in place by two screws to stop it slipping. Before the module rack can be screwed in place, the threaded strips must be pushed in and the left side section and screws for the right side section must be pre-mounted.

The modules must be arranged so that the controller is on the lower end of the module rack.

⚠️ **The temperature range is restricted to 0 - 50 °C when installing modules vertically.**

Standard Installation

If the controller is hung in the module racks, 22 mm must be added to the width listed in section "Module Racks" (column "7BP7xx.0") when calculating the height.

Installation with CAN Bus Controller EX270

The CAN Bus Controller EX270 is used together with the module racks having model numbers 7BP70x.1. The EX270 is screwed onto the module rack instead of the left side section.
To calculate the height, 31 mm is to be added to the width listed in section "Module Rack" (column "7BP70x.1").

2 INSTALLATION

<table>
<tr>
<td>⚠️</td>
<td>Installation is only permitted to be carried out by qualified personnel!</td>
</tr>
</table>

The B&R SYSTEM 2003 is to be installed in the following order:

1) Installing the mounting rail
2) Installing the modules on the module rack
3) Installing the entire unit

2.1 MODULES

In order to install and fasten the modules on the module rack, you first need to push in the threaded strip and attach the left hand side section (see following diagram).

- If the threaded strip has not been pre-installed, push it into the aluminum frame:
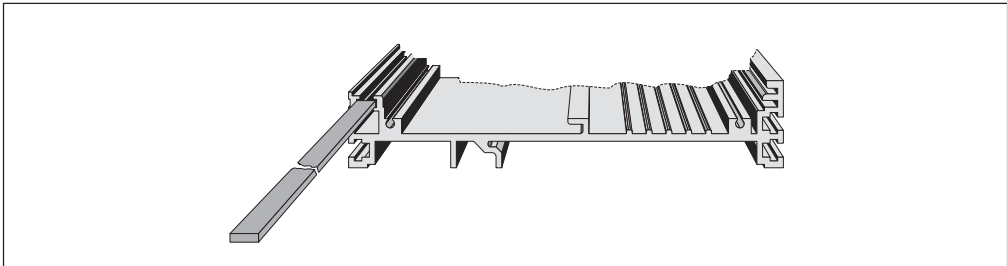
- Attach the left hand section to the aluminum frame using two screws. This side section serves as a stop for the first module.
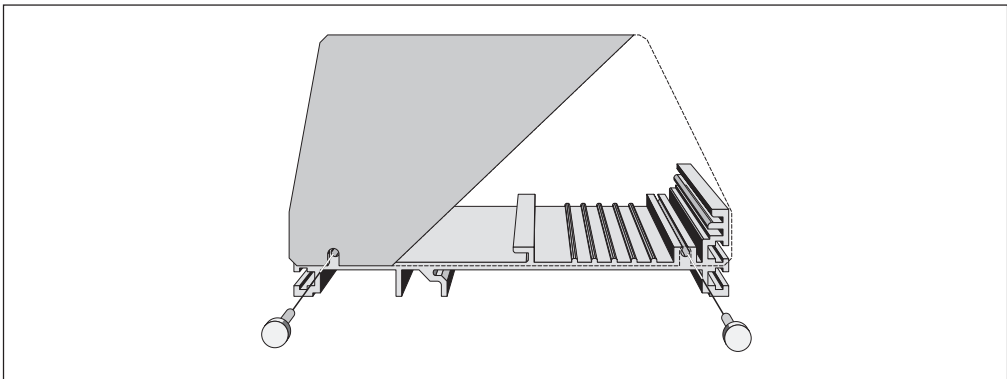
After you have carried out these preparations, place the modules on the module rack and push them to the left (making contact with the modules) and secure them with fastening screws.

- Place the modules on the module rack (the side section which has already been installed, is not shown in the diagram):
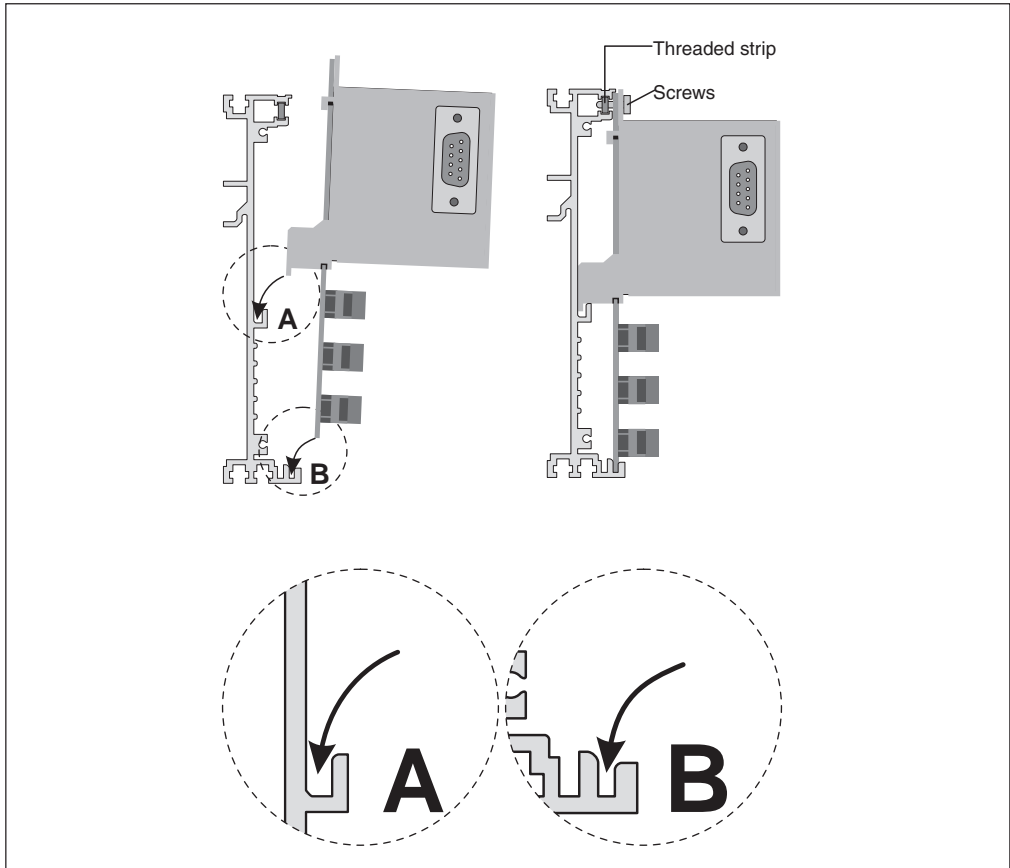
- Push the first module until it reaches the side section.
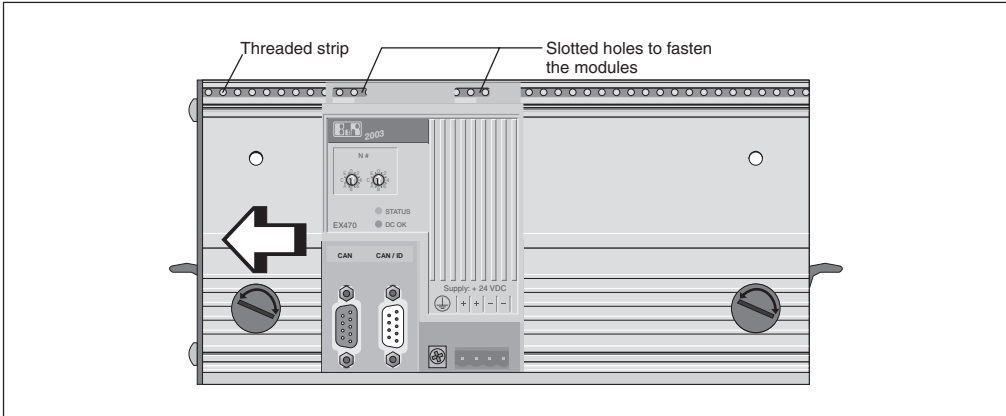
- Each module is fastened to the threaded strip using two screws. The angled ground connector should also be mounted on the power supply and connected to the module rack.

- The other modules are installed as described above, pushed to the left and fastened with screws.
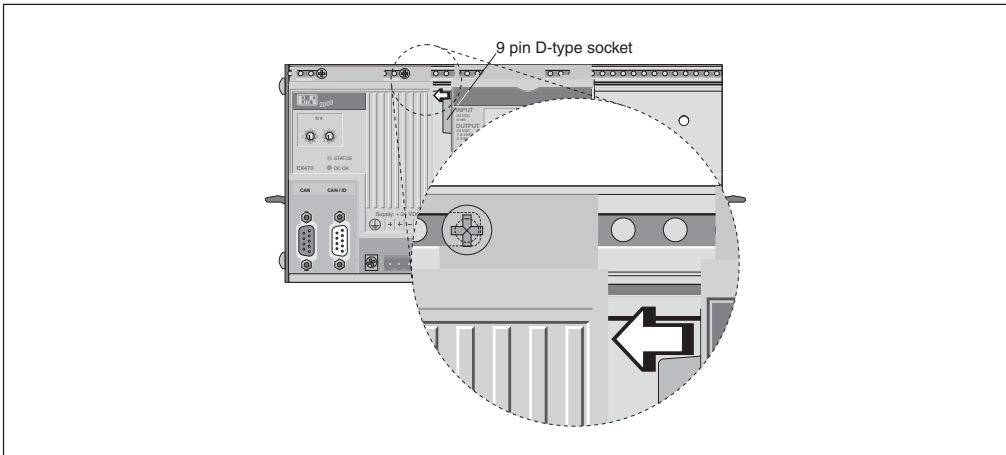
Make sure that the modules are pushed together properly so that there is a perfect connection between the modules.

- After you have installed and fastened all the modules on the module rack, install the right hand side section.

2.2 MODULE RACK

Carry out the following steps to mount the module rack on the mounting rail:

- Open both fastening levers (setting: OPEN)
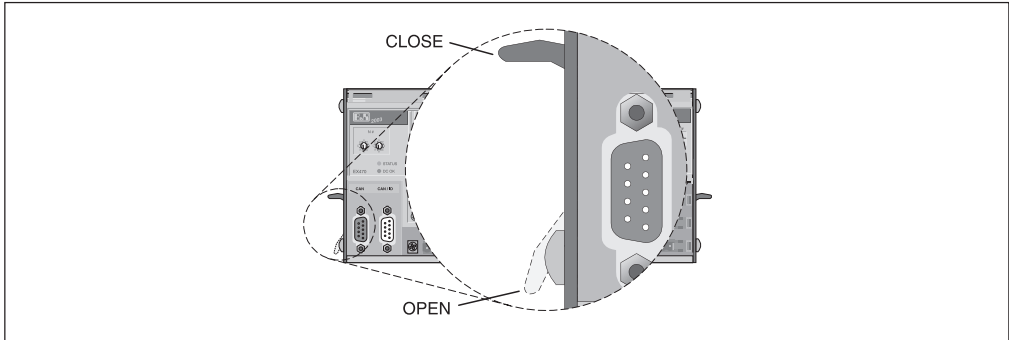
- Place the module rack in the desired position on the mounting rail
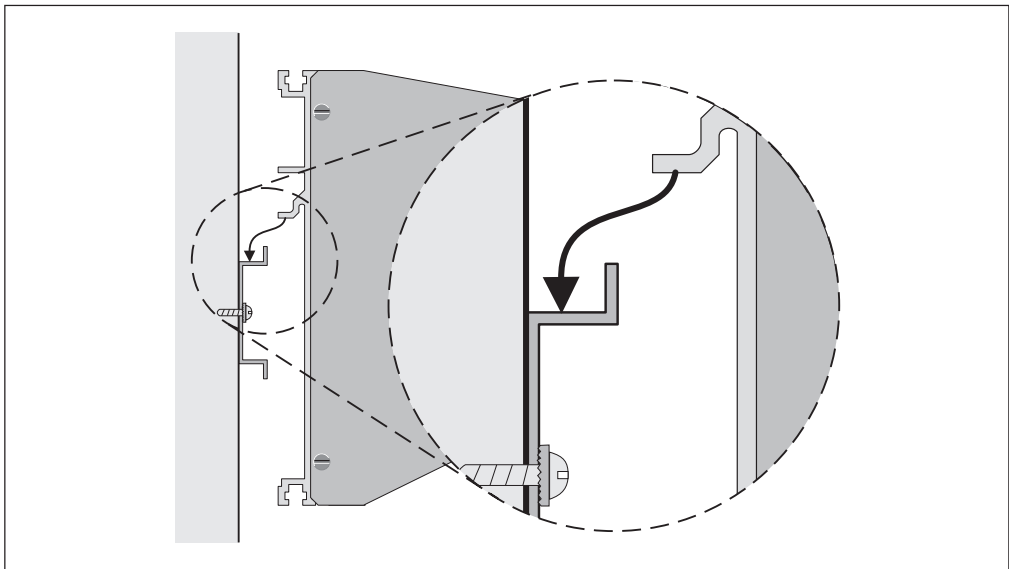
- Close both fastening levers (setting: CLOSE)

2.3 TERMINAL BLOCKS

Single row terminal blocks are used to connect the I/O modules. These terminal blocks are ejected using two levers. Pressing the ejection lever (using a screw driver) causes the terminal block to be pushed out of the pin block so it can then be easily taken out. They are available with screw clamps and also with cage clamps.

Screw clamps   Cage clamps

2.3.1 Cabling Terminal Blocks

Daisy Chaining

The horizontal arrangement of the terminals allows the supply voltage to be daisy chained:
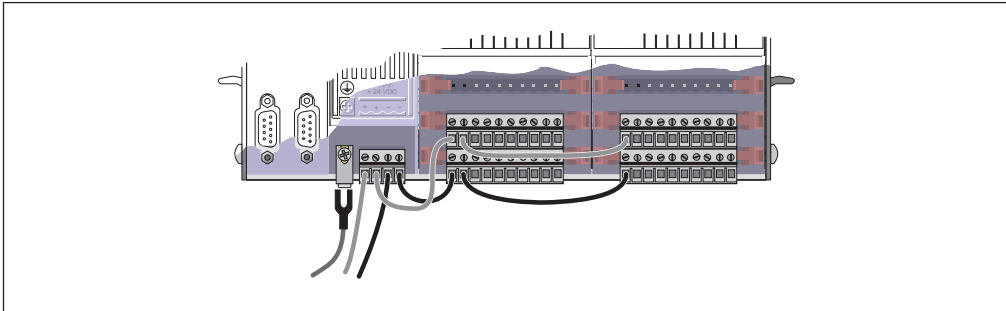
This diagram shows an example of how supply lines can be daisy chained.

⚠️ **Pay attention to the required voltages and currents!**

Daisy chaining is only possible if all modules use the same voltage. Additionally, the maximum current load for the connector is not allowed to be exceeded.

Input and Output Connections

The connections for all the inputs and outputs are arranged one on top of the other. This makes things a lot clearer, and enables you to find the correct cable for the corresponding channel easily. Possible exceptions are listed in the module description.
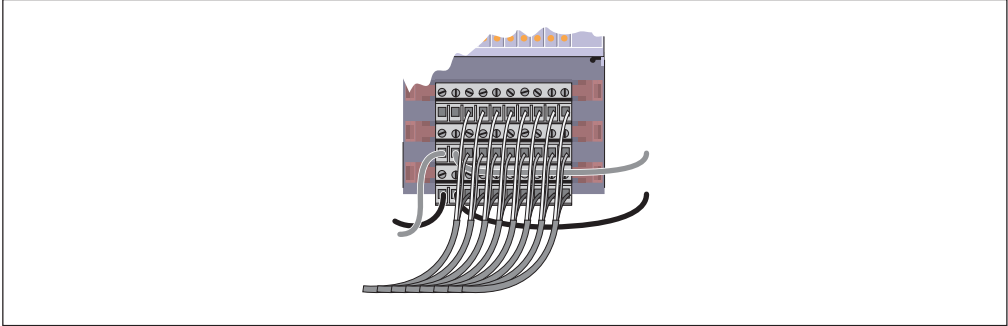
2.4 SCREW-IN MODULES

The screw-in modules are delivered in a specially made box. The box has two functions:

- Protection from damage during transportation

- Protection during cabling

2.4.1 Cabling Protection

The controller is often not available while a system is being cabled, e.g. it is still being used by the programmer to complete the control program.
The electrician connects the cable to terminal blocks and D-type connectors which will be plugged onto the controller when it is installed. However, the screw-in modules AI261, AI294 and NC161 do not have to be cabled directly.
In order to protect the screw-in modules from dust and mechanical damage until the controller is installed, these modules are to be placed in the box again after it is wired.

B&R recommend the following procedure:

❶ Remove the screw-in module from the box.

❷ Remove the top section of the box along the perforation.

Remove top section
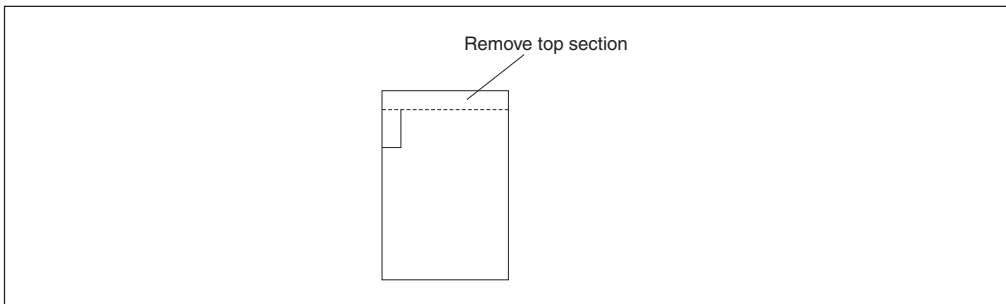
❸ Place the cabled screw-in module box so that the slanted section is facing the corner with the pre-made fold lines. The cable can then hang out of the box.
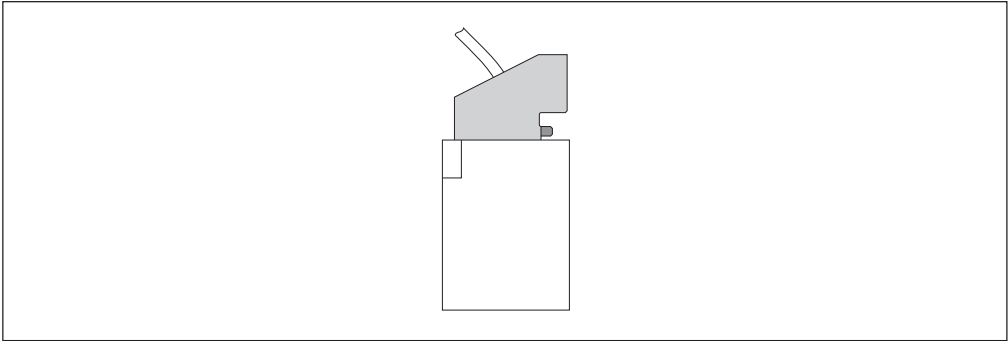
❹ Fold the box in the corner with the pre-made fold lines. This prevents the box from falling off.

Fold in corner

Top view:

Not folded

Folded
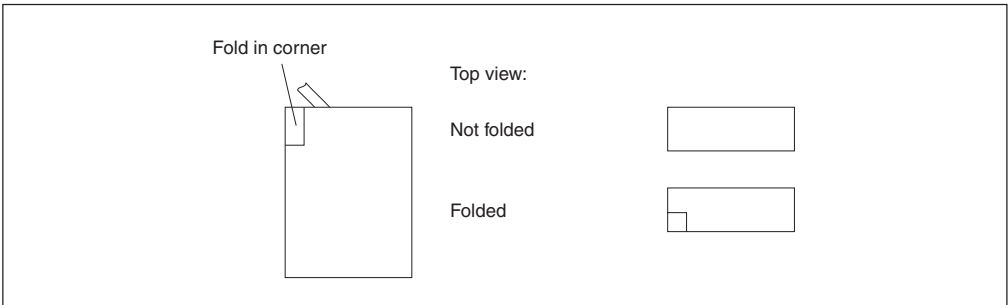
3 MODULE SLOT RULES

3.1 PHYSICAL MODULE SLOTS

A physical module slot corresponds to the actual space required for a module. 2003 modules can be single width (= one module slot) or double width (= two module slots), for example like the CP474.
Module racks are available in different lengths for the B&R SYSTEM 2003. The palette ranges from one module slot to a maximum of 10 module slots (= module addresses or slots in the hardware view in Automation Studio™).

3.2 LOGICAL MODULE SLOTS

Some modules require more than one logical module slot. That means the number of physical module slots required is different than the number of logical modules slots required.

The maximum number of logical module slots depends on the controller. The controller also determines how many module slots are available for analog modules (see section "Modules that use Logical or Analog Module Slots").

Various controller are limited as to the maximum number of analog module slots and also as to the module slots for analog modules. Both conditions must be met.

Controller	Maximum number of log. module slots [1]	Maximum number of analog module slots [1]	Possible module addresses for analog modules [2]
CP430	4	2	1 – 4
CP470 / CP770	8	4	1 – 8
CP474 / CP774	12	4	1 – 8
CP476	16	4	1 – 8
EX270	4	2	1 – 2
EX470 / EX770	8	4	1 – 4
EX477 / EX777	8	4	1 – 8

[1] **Attention:** Note power output table!

[2] All analog modules and modules with logical analog sections must be operated directly next to the controller, that means they must be inserted to the left of the first digital module. Module slots 1 or 1+2 are used by the controllers (exception: EX270 does not require a module slot). The first slot to the right of a controller has module address 1 and the module addresses are numbered in increasing order to the right.

Attention: A module slot does not correspond to the module address, it only refers to the actual space required on the module rack.

3.3 MODULES THAT USE LOGICAL OR ANALOG MODULE SLOTS

The following table contains an overview of the modules that use more than two logical module slots or that use an analog module slot.
If a module uses two logical module slots and one of these slots is an analog module slot, it is always the first one. The number of logical module slots corresponds to the module addresses used (= slots in the hardware view in Automation Studio™).

Module	Number of logical module slots	Number of analog module slots	Numbe of phys. module slots used
AF101	1	1	1
DI439	2	---	1
DM465	2	---	1
CM211	2	1	1
CM411	2	2	1

3.4 CONFIGURATION EXAMPLES

Example 1

Configuration with one CP430 and two CM211:

Physical Module Assignments (physical module slots):

1	2	3
CP430	CM211	CM211

Correct Configuration

Logical Module Assignments(module addresses):

0	1	2	3	4
CP430	CM211 analog	CM211 digital	CM211 analog	CM211 digital

Example 2

Configuration with one EX270 and two CM211:

Physical Module Assignments (physical module slots):

	1	2
EX270	CM211	CM211

Incorrect Configuration

Logical Module Assignments (module addresses):

0	1	2	3	4
EX270	CM211 analog	CM211 digital	CM211 analog	CM211 digital

⚠ **Module slot 3 is not allowed to be used for analog modules on the EX270!**

4 POWER OUTPUT TABLE

Each hardware configuration must have a table of values. You can determine if the power supply output from the CPU is sufficient for the modules using this table.

An output table can be created quickly and clearly using the table found in Chapter 3 "B&R 2003 Modules" in section "Module Overview". The column labeled "Power" contains values for the power provided by a module or required by a module.

4.1 EXAMPLE 1

B&R SYSTEM 2003 with a typical hardware configuration.

Module	Power [W]	Number of Modules	Total Power [W]
CP474	12.6	1	+12.6
DM435	0.5	3	-1.5
DI435	0.2	1	-0.2
AI354	0.5	1	-0.5
AO352	1.2	1	-1.2
AT664	0.4	1	-0.4
			+8.8

4.2 EXAMPLE 2

B&R SYSTEM 2003 with maximum configuration for digital signals.

Module	Power [W]	Number of Modules	Total Power [W]
CP476	12.5	1	+12.5
DM465	1.1	8	-8.8
DI135	0.4	4	-1.6
			+2.1

4.3 EXAMPLE 3

B&R SYSTEM 2003 with maximum configuration for analog signals.

Module	Power [W]	Number of Modules	Total Power [W]
CP474	12,6	1	+12.6
DI435	0,2	4	-0.8
AF101	0,3	4	-1.2
AI774	0,4	20	-8
			+2.6

5 REMOTE I/O BUS

The remote I/O bus allows remote I/O modules to be connected to a remote master. The distance without a repeater can be up to 1200 m. Up to 31 slaves can be connected to a remote master without using a repeater.

5.1 REMOTE MASTER

The remote master is a B&R SYSTEM 2005, B&R SYSTEM 2010 or B&R SYSTEM 2000 Logic Scanner.

5.2 REMOTE SLAVE

A remote I/O bus controller EX477 or EX777 is required to add a B&R SYSTEM 2003 as slave in a remote I/O network.

5.3 CABLING

5.3.1 Cabling Diagram
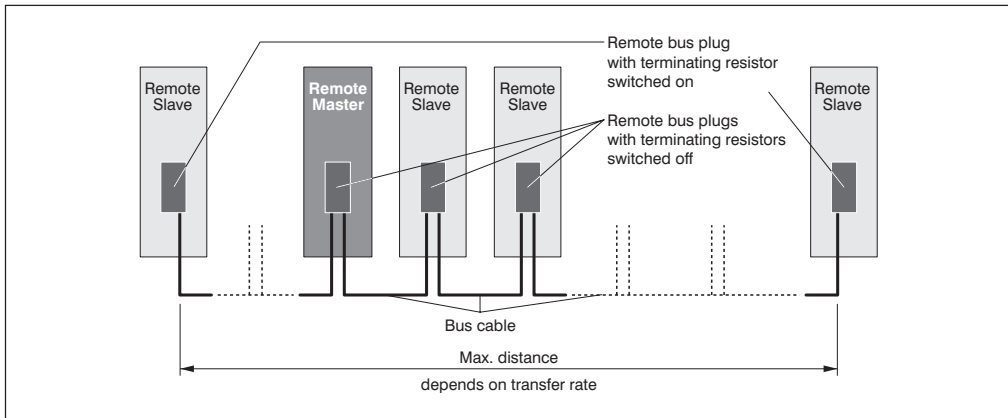
5.3.2 Bus Cable

Standard: DIN 19245 Part 3

The remote master and slave modules are connected using a two conductor cable and must conform to the following specifications.

Wave Impedance	135 - 165 Ω (3 - 20 MHz)
Distributed Capacitance	<30 pF / m
Loop Resistance	<110 Ω / km
Wire Diameter	>0.64 mm
Wire Cross Section	>0.34 mm^2

Both ends of the two conductor cable must be equipped with terminating resistors.

5.3.3 Connecting Bus Cable - Station

The two wires of the bus cable are connected to the individual stations as follows:
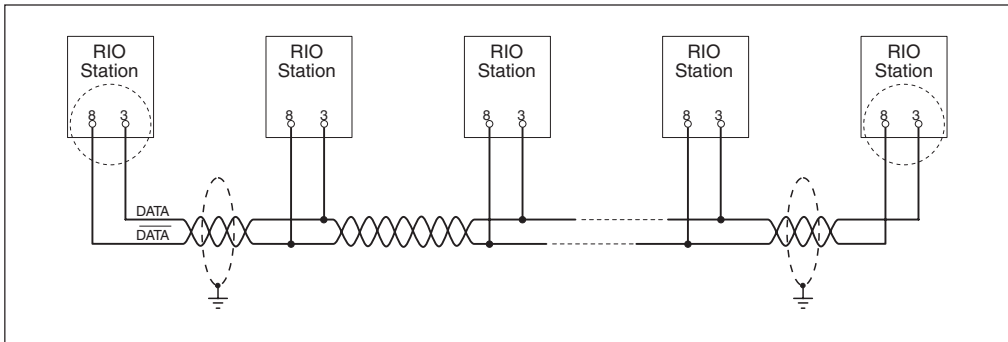
5.3.4 Terminating Resistance

The remote bus is equipped with terminating resistors at both ends. B&R can provide remote bus connectors 0G1000.00-090 complete with integrated terminating resistors. Terminating resistors can be turned on or off.

5.3.5 Resistor Diagram
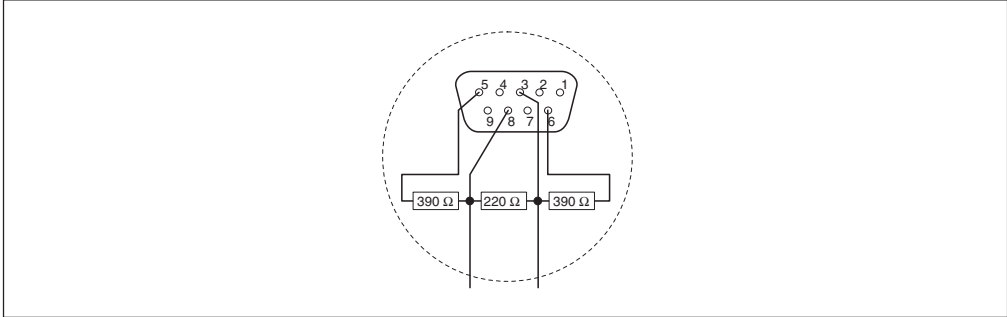
5.3.6 Remote Bus Plug

Model Number: 0G1000.00-090
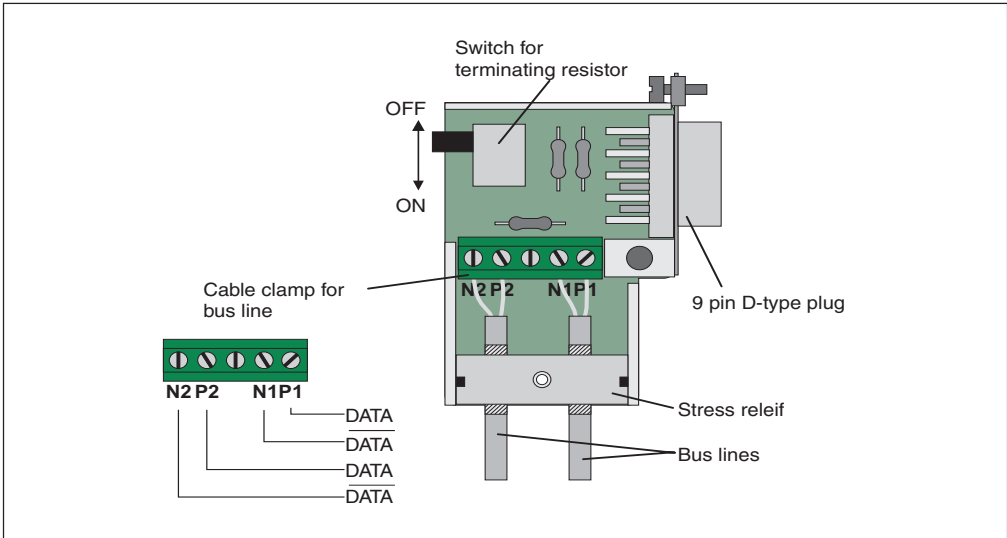
5.4 DISTANCES

The maximum distance of a remote system depends on the rate of communication transfer:

Distance [m]	Transfer Rate [kBit/s]
1200	100
1000	181
400	500
200	1000
100	2000

5.5 PROFIBUS

The cabling described for the remote I/O bus (bus cable, terminating resistors) is also used for PROFIBUS. The PROFIBUS network modules available from B&R are NW150 for the B&R SYSTEM 2005 and NW100 for the B&R SYSTEM 2010.

5.6 RS485 NETWORK

The cabling described for the remote I/O bus (bus cable, terminating resistors) is also used for a RS485 network.

6 CAN FIELD BUS

6.1 CHARACTERISTICS OF THE CAN BUS

- Low costs
- High noise immunity through differential signals
- Bus structure
- Open system
- Fast data transfer for small data packages (up to 8 bytes)
- Error detection by means of CRC (Cyclic Redundancy Check) and frame testing -> Hamming Distance 6
- Predictable transmission time for high priority messages (real time behavior)
- Easy use

6.2 BUS LENGTHS AND CABLE TYPE

The type of cable used depends largely on the required bus length and the number of nodes. The bus length is mainly determined by the bit rate.

The table on the next page includes values for the maximum bus length depending on the transfer speed and the Synchronization Jump Width (SJW). Permitted oscillator tolerances are given in the fourth column.

The synchronization jump width (SJW) is the factor that determines the range over which the CPU can be synchronized. The bigger the SJW, the shorter the maximum bus length.

Bit Rate [kBit/s]	Synchronization Jump Width (SJW)	Bus Length [m]	Max. Oscillator Tolerance [%]
500	0 [1]	67	0.121
	1	56	0.242
	2	33	0.363
	3	10	0.485
250	0 [1]	215	0.121
	1	192	0.242
	2	147	0.363
	3	101	0.485
125	0 [1]	510	0.121
	1	465	0.242
	2	374	0.363
	3	283	0.485
100	0 [1]	658	0.121
	1	601	0.242
	2	488	0.363
	3	374	0.485
50	0 [1]	1397 [2]	0.121
	1	1284 [2]	0.242
	2	1056 [2]	0.363
	3	829	0.485
20	0 [1]	3613 [2]	0.121
	1	3329 [2]	0.242
	2	2761 [2]	0.363
	3	2193 [2]	0.485
10	0 [1]	7306 [2]	0.121
	1	6738 [2]	0.242
	2	5602 [2]	0.363
	3	4456 [2]	0.485

[1] B&R default settings

[2] For CiA (CAN in Automation) the maximum Bus length is 1000 m.

Relationship between the number of nodes and bus length for certain cable types using B&R default settings:

Cable 1	Belden YR 29832, 4.15 ns/m
Cable 2	Lapp Cable 2170204, 4.15 ns/m
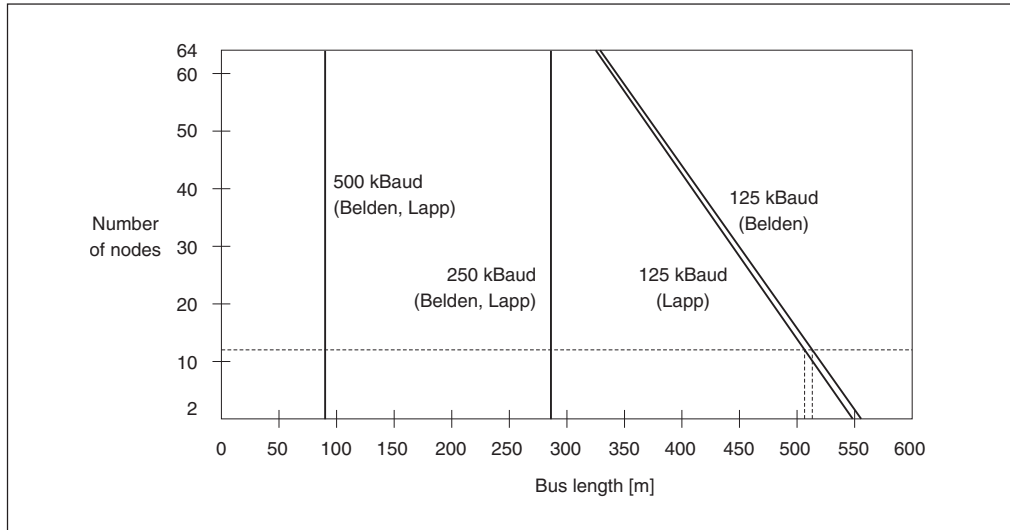
Example for 12 Nodes

Manufacturer	Baudrate	Bus Length	Baudrate	Bus Length	Baudrate	Bus Length
Belden	500 kBaud	90 m	250 kBaud	286 m	125 kBaud	513 m
Lapp	500 kBaud	90 m	250 kBaud	286 m	125 kBaud	506 m

6.3 CABLING

6.3.1 Connecting Bus Cable - Station

A 4 conductor twisted pair cable should be used for the bus cable.
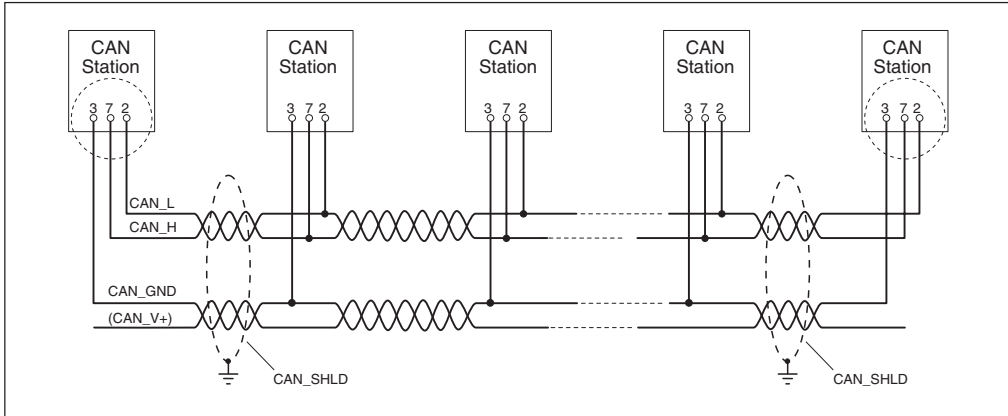
6.3.2 CAN Signals according to CiA/CAL

CAN Signal	Description
CAN_GND	CAN Ground
CAN_L	CAN Low
(CAN_SHLD)	Shield
CAN_H	CAN High
(CAN_V+)	CAN Supply 8 - 15 V, optional

All CAN interfaces from B&R are supplied internally, therefore CAN_V+ does not have to be connected in CAN networks which do not contain devices from other manufacturers.

6.3.3 Branch Lines

Branch lines should be avoided if possible. However, it is possible to connect nodes to the bus with a branch line as long as the distance is less than 30cm.

6.3.4 Terminating Resistor

CAN networks are cabled using a bus structure. Both ends of the bus are equipped with terminating resistors.
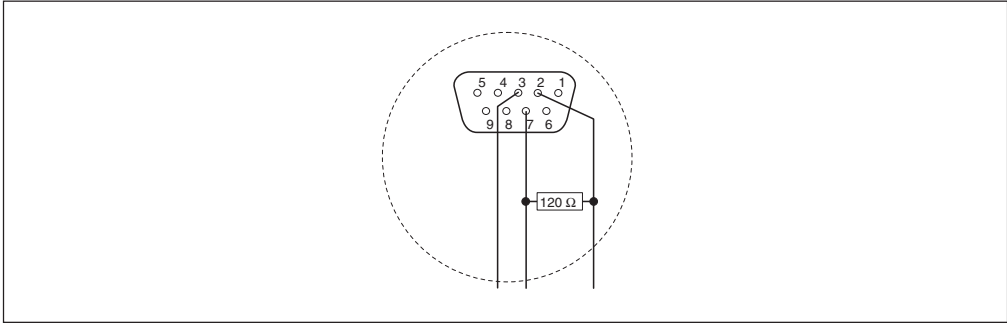
The terminating resistor is already integrated in the CAN bus plug 7AC911.9 available from B&R. The terminating resistor can be turned on or off.
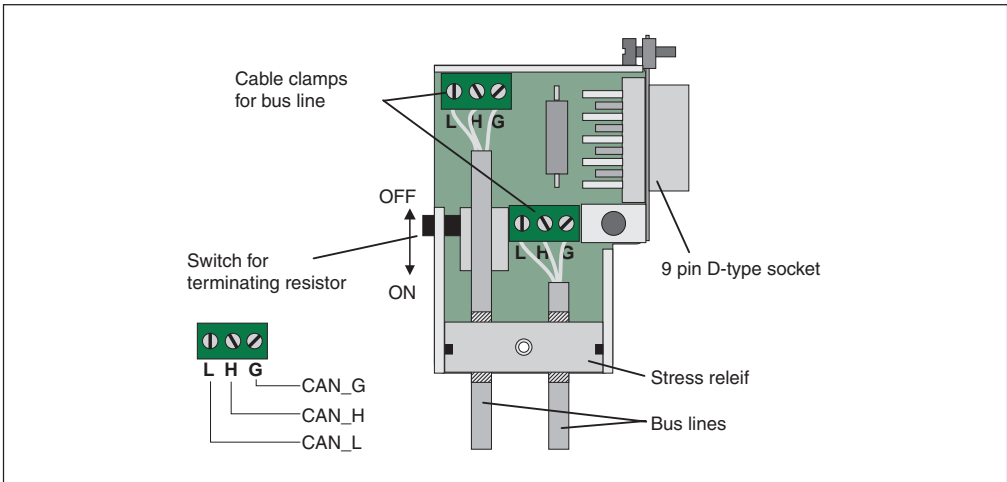
7 GROUNDING AND SHIELDING MEASURES

For most applications, industrial controllers are installed in switching cabinets, in which electromagnetic switching elements (relays, contactors), transformers, motor controllers and frequency inverters are also found. Different kinds of electromagnetic interference inevitably exist in such switching cabinets.

Generally this kind of interference can not be avoided. However, if suitable grounding, shielding and other protective measures are taken, the negative influence of the devices can be considerably reduced. These protective measures include switching cabinet grounding, module grounding, cable shield grounding, protective circuits on electromagnetic switching elements and correct use of cables (choosing the proper cable cross section and type)

Grounding has essentially two different functions:

- Protective grounding
- Dissipation of electromagnetic disturbances

In the B&R SYSTEM 2003 system, the dissipation of electromagnet disturbances and protective grounding are carried out by the aluminum frame (module rack) and the mounting rail.

7.1 GROUNDING THE MOUNTING RAIL

For the purpose of grounding, it is necessary to provide a good conductive connection between the mounting rail and the metal back wall. The mounting rail is to be connected conductively to the back wall every 10cm. This is achieved by inserting a contact washer with the fastening screw:
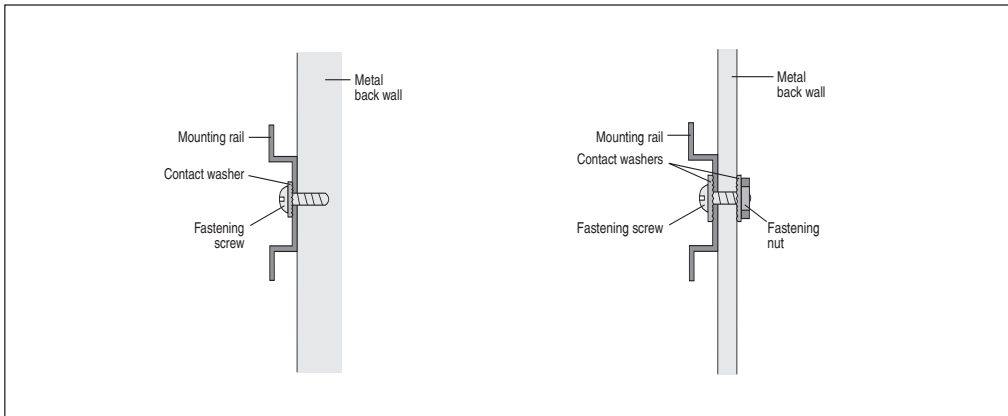
For lacquered or coated back walls, an adequate connection is only guaranteed if the screw is threaded into the back wall. If this is not the case, a contact washer must be must be placed between the fastening nut and the base plate.

⚠ **The switching cabinet back wall must be connected with GND (⏚)**

7.2 GROUNDING THE B&R SYSTEM 2003 UNIT

The CPU or bus controller module must be directly connected to the aluminum frame using an angled metal connector.

a) The angled metal connector is fastened to the GND terminal on the module:
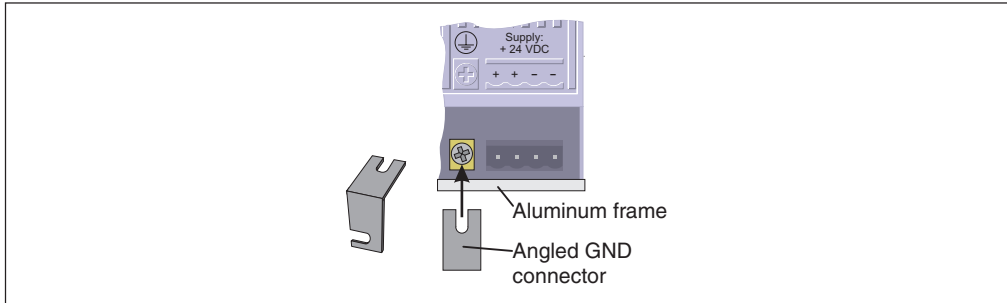
b) Connect the angled metal connector with the aluminum frame (module rack) using the pre-installed screw and threaded strip:
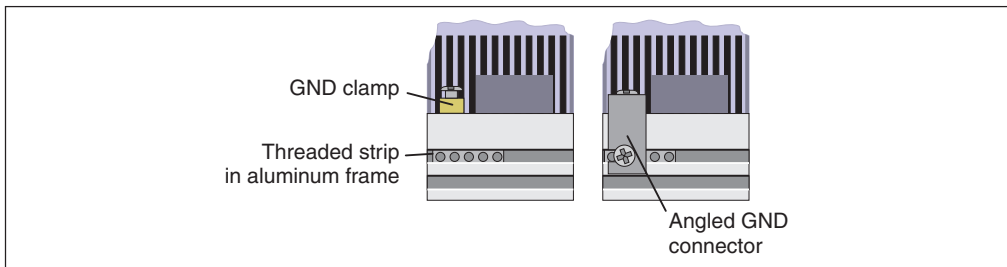
c) A cable with sufficient cross-section must be connected to the GND terminal for protective grounding in order to compensate for differences in potential.

7.3 CABLE SHIELDING

The following connections are made with shielded cables (possible exceptions are listed in the respective module descriptions):

- Analog I/O
- Interface Cable
- Encoder Cable

The cable shield is to be grounded at **both ends**. The shields are grounded directly to the module housing on the B&R SYSTEM 2003 side.

If any differences in potential exist between the B&R SYSTEM 2003 and connected elements, transient current flows over the cable shield (often causes the cable to get warm) and the following steps should be taken: The cable shield is to be separated and bridged with a high quality capacitor (ceramic or foil capacitors of at least 47nF with a low impedance at high frequencies).

7.4 USING D-TYPE PLUGS

D-type plugs must be equipped with a metal coated plug housing.

Cable shield connected with cable clamp and housing
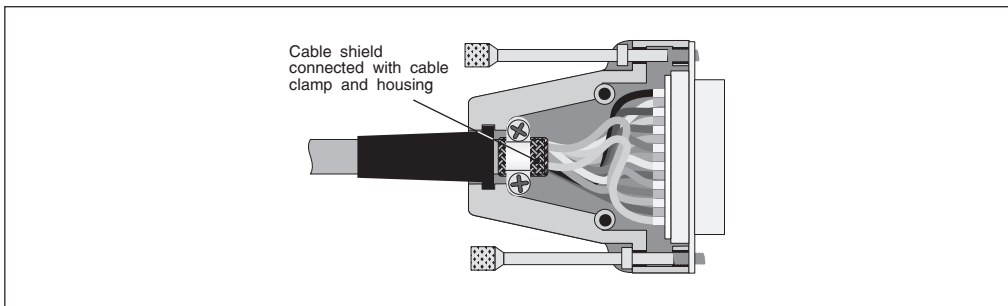
The shield is connected directly to the plug housing. Connecting the shield by twisting it then attaching it considerably reduces the shielding effect, and should therefore be avoided.
If metallic screw clamps are not available, the cable shield can be externally grounded with a grounding clamp (see cable shield grounding).

8 EXTERNAL PROTECTIVE CIRCUIT

An external protective circuit is necessary for relay output modules. It can be installed either on the switching load, or between the terminal block and the load. Most manufacturers of solenoid operated valves and contactors offer suitable protective circuit devices for the respective elements.

Possible protective elements are:

- RC element: Can be used for D.C or A.C.

- Varistor: Usually used for A.C.

- Inverse Diode: Can only be used for D.C.

- Diode/Z diode combination: Can only be used for D.C. This type of protective circuit allows shorter switch-off times.

9 INSTALLATION GUIDELINES

- The B&R SYSTEM 2003 can be installed both vertically and horizontally. Please observe the installation guidelines in the "Installation Dimensions" section.

- The maximum environmental temperature during operation refers to the air temperature underneath the module (air inlet).

- Enough distance must be left between devices which cause high electromagnetic interference (e.g. frequency converters, transformers, motor regulators etc.) and the B&R SYSTEM 2003.The distance between the devices and the B&R SYSTEM 2003 should be as large as possible. If possible they should be shielded using an isolation sheet (VACOPERM® 7 metal).

Inserting/Removing Modules

- Module may not be inserted or removed when the PCC is turned on unless otherwise stated for that particular module.

- Unplug cabled connectors before modules are removed.

- Power connections are not allowed to be inserted or removed with voltage applied (remove power).

10 STORAGE AND STORAGE TEMPERATURES

The CPUs are equipped with a backup battery. Storage temperatures from -20 to +60 °C are valid for CPUs. All other modules can be stored at temperatures from -20 to +70 °C. If applicable, other temperatures are listed in the technical data for the modules.

11 TEMPERATURE DURING OPERATION, HUMIDITY

The following values are valid for all 2003 modules if other values are not listed in the "Technical Data" section.

Temperature during Operation	
Horizontal Installation	0 to 60°C
Vertical Installation	0 to 50°C
Humidity	5 to 95 %, non-condensing

CHAPTER 3
B&R 2003 MODULES

1 MODULE OVERVIEW B&R 2003

The column labeled "Power" contains values for the power provided by a module or required by a module. In this way, a power balance can be calculated quickly and easily for a certain hardware configuration.

The power supplied by CPUs, Remote I/O and CAN bus controllers is shown with a "+" sign. The power required by a module is shown with a "-" sign. Add the positive and negative power values to calculate the power balance. The sum is not allowed to be less than zero.

1.1 SORTED ALPHABETICALLY ACCORDING TO MODULE ID

ID	Description	B&R ID Code	Power	Model No.	Page
AC010	5 bus covers			7AC010.9	451
AC011	5 stress relief attachment including mounting material			7AC011.9	452
AC020	1 bus cover			7AC020.9	451
AF101	Adapter module AF101	$C0	-0.3 W	7AF101.7	131
AI261	1 Input used to evaluate a full-bridge strain gauge, 24 bit resolution	$20	-0.6 W	7AI261.7	260
AI294	4 inputs for potentiometer displacement gauge, 13 bit resolution	$22	-0.5 W	7AI294.7	277
AI351	1 input ±10 V or 0 - 20 mA, 12 bit resolution + sign	U: $02 Pot.: $02 I: $03	U/I: -0.3 W Pot.: -0.7 W	7AI351.70	285
AI354	4 inputs ±10 V, 12 bit resolution + sign	$04	-0.5 W	7AI354.70	293
AI774	4 inputs 0 – 20 mA, 12 bit resolution	$06	-0.4 W	7AI774.70	299
AO352	2 outputs ±10 V / 0 - 20 mA	$0E	-1.2 W	7AO352.70	306
AT324	4 inputs for temperature sensors (PT100 / PT1000 / KTY10-6 / KTY84-130)	$3E	-0.1 W	7AT324.70	313
AT352	2 inputs for PT100 sensors (3-line)	$1A	-0.4 W	7AT352.70	320
AT664	4 inputs for thermocouples	$0C	-0.4 W	7AT664.70	326
BP701.1	Module rack, for 1 module, incl. right side section			7BP701.1	80
BP702	Module rack, for 2 modules			7BP702.0	80
BP702.1	Module rack, for 2 modules, incl. right side section			7BP702.1	80
BP703	Module rack, for 3 modules			7BP703.0	80
BP704	Module rack, for 4 modules			7BP704.0	80
BP705	Module rack, for 5 modules			7BP705.0	80
BP706	Module rack, for 6 modules			7BP706.0	80
BP707	Module rack, for 7 modules			7BP707.0	80
BP708	Module rack, for 8 modules			7BP708.0	80
BP709	Module rack, for 9 modules			7BP709.0	80
BP710	Module rack, for 10 modules			7BP710.0	80
CM211	8 digital inputs, 24 VDC, 20 kHz, one or two channel counter, incremental encoder, 8 digital outputs, 24 VDC, 0.5 A, comparator function, 2 analog inputs ±10 V or 0 - 20 mA, 12 bit resolution, 2 analog outputs ±10 V, 12 bit resolution	$C3	-1.5 W	7CM211.7	334
CM411	3 digital inputs, 24 VDC, 50 kHz, one or two channel counter, incremental encoder, 2 digital outputs, 24 VDC, 0.5 A, comparator function, 3 analog inputs ±10 V, 16 bit resolution, 3 analog outputs ±10 V, 16 bit resolution	$C2	-2.4 W	7CM411.70-1	384
CP430	CPU, 24 VDC, 100 KByte SRAM, 256 KByte FlashPROM		+7 W [1]	7CP430.60-1	102
CP470	CPU, 24 VDC, 100 KByte SRAM, 256 KByte FlashPROM		+14 W [1]	7CP470.60-1	102
CP474	CPU, 24 VDC, CP Interface, 100 KByte SRAM, 512 KByte FlashPROM		+12.6 W [1]	7CP474.60-1	102
CP476	CPU, 24 VDC, CP Interface, 750 KByte SRAM, 1.5 MByte FlashPROM		+12.5 W [1]	7CP476.60-1	114
CP770	CPU, 100 - 240 VAC, 100 KByte SRAM, 256 KByte FlashPROM		+14 W [1]	7CP770.60-1	102
CP774	CPU, 100 - 240 VAC, CP Interface, 100 KByte SRAM, 512 KByte FlashPROM		+12.6 W [1]	7CP774.60-1	102

[1] Integrated power supply for simple PANELWARE controllers, e.g. P120

ID	Description	B&R ID Code	Power	Model No.	Page
DI135	4 inputs, 24 VDC, event counter/incremental encoder operation, 1 comparator output, 24 VDC	$12	-0.4 W	7DI135.70	140
DI435	8 inputs, 24 VDC, input delay 1 ms	$E1	-0.2 W	7DI435.7	159
DI439.7	16 inputs, 24 VDC, input delay 1 ms, termnal block connection	$ED	-0.4 W	7DI439.7	164
DI439.72	16 inputs, 24 VDC, input delay 1 ms, D-type connection	$E9	-0.4 W	7DI439.72	170
DI645	8 inputs, 100 - 240 VAC, input delay 50 ms	$E5	-0.2 W	7DI645.7	178
DM435	8 digital inputs, 24 VDC, input delay 1 ms, 8 digital outputs, 24 VDC, 0.5 A, terminal block connection	$E3	-0.5 W	7DM435.7	233
DM438	8 digital inputs, 24 VDC, input delay 1 ms, 8 digital outputs, 24 VDC, 0.5 A	$E7	-0.5 W	7DM438.72	241
DM465	16 digital inputs, 24 VDC, input delay 1 ms, 16 digital outputs, 24 VDC, 0.5 A	$F7	-1.1 W	7DM465.7	249
DO135	4 FET outputs, 12 - 24 VDC, 0.1 A, 100 kHz	$14	-0.2 W	7DO135.70	185
DO164	4 FET outputs (4 triac couplers), 48 - 125 VAC, 50 mA, zero voltage input	$3C	-0.6 W	7DO164.70	197
DO435	8 channels, each channel can be declared either as output or input Max. 8 FET outputs, 24 VDC, 2 A, max. 8 inputs, 24 VDC, inputs delay 1 ms	$EB	-0.5 W	7DO435.7	211
DO720	8 relay outputs, 240 VAC / 30 VDC, 2 A	$E2	-1.4 W	7DO720.7	219
DO721	4 relay outputs, 240 VAC / 24 VDC, 4 A	$E6	-1.4 W	7DO721.7	224
DO722	8 relay outputs, 240 VAC / 24 VDC, 2.5 A	$F2	-1.4 W	7DO722.7	228
EX270	CAN bus controller, 24 VDC supply		+4 W	7EX270.50-1	84
EX470	CAN bus controller, 24 VDC supply		+14 W [1]	7EX470.50-1	90
EX477	Remote I/O bus controller, 24 VDC supply		+14 W	7EX477.50-2	95
EX770	CAN bus controller, 100 - 240 VAC supply		+14 W [2]	7EX770.50-1	90
EX777	Remote I/O bus controller, 100 - 240 VAC supply		+14 W	7EX777.50-1	95
IF311	Interface module with RS232 interface without PW operator panel with PW operator panel P120 / P121	$18	-0.5 W -1.6 W	7IF311.7	416
IF321	Interface module with RS485/RS422 interface	$16	-1.4 W	7IF321.7	416
IF361	Interface module with RS485 interface (PROFIBUS-DP slave)	$25	-2.6 W	7IF361.70-1	418
IF371	Interface module with CAN interface	$24	-2.5 W	7IF371.70-1	420
ME010	2003 expansion for the CP476, 1 PCMCIA slot	$00	-0.35 W	7ME010.9	408
ME020	2003 expansion for CP476, 1 PCMCIA slot, 1 slot for interface module inserts	$02	-0.7 W	7ME020.9	411
ME770	Configuration memory for CAN bus controller		-0.1 W	7ME770.5	128
NC161	Encoder module, input frequency 100 kHz, incremental or absolute, 32 bit resolution	$10	-0.3 W $-I_{enc}$ * 5.4 V	7NC161.7	423
TB710	Single row terminal block, 10 pin, screw clamps			7TB710.9	454
TB710	Single row terminal block, 10 pin, cage clamps			7TB710.91	454
TB710	30 single row terminal blocks, 10 pin, screw clamps			7TB710:90-01	454
TB710	30 single row terminal blocks, 10 pin, cage clamps			7TB710:91-01	454
TB712	Single row terminal block, 12 pin, screw clamps			7TB712.9	455
TB712	Single row terminal block, 12 pin, cage clamps			7TB712.91	455
TB712	20 single row terminal blocks, 12 pin, screw clamps			7TB712:90-02	455
TB712	20 single row terminal blocks, 12 pin, cage clamps			7TB712:91-02	455
TB718	Single row terminal block, 18 pin, screw clamps			7TB718.9	456
TB718	Single row terminal block, 18 pin, cage clamps			7TB718.91	456
TB718	20 single row terminal blocks, 18 pin, screw clamps			7TB718:90-02	456
TB718	20 single row terminal blocks, 18 pin, cage clamps			7TB718:91-02	456
TB722	2-row terminal block, 22 pin, screw clamps			7TB722.9	457
TB722	2-row terminal block, 22 pin, cage clamps			7TB722.91	457
TB733	3-row terminal block, 33 pin, screw clamps			7TB733.9	458
TB733	3-row terminal block, 33 pin, cage clamps			7TB733.91	458
TB736	2-row terminal block, 36 pin, screw clamps			7TB736.9	459
TB736	2-row terminal block, 36 pin, cage clamps			7TB736.91	459

[1] EX470 with revision 30.xx or higher

[2] EX770 with revision 10.xx or higher

ID	Description	B&R ID Code	Power	Model No.	Page
TB754	3-row terminal block, 54 pin, screw clamps			7TB754.9	460
TB754	3-row terminal block, 54 pin, cage clamps			7TB754.91	460
TB772	4-row terminal block, 72 pin, cage clamps			7TB772.91	461

1.2 SORTED ACCORDING TO GROUP

ID	Description	B&R ID Code	Power	Model No.	Page
	Module Racks				
BP701.1	Module rack, for 1 module, incl. right side section			7BP701.1	80
BP702	Module rack, for 2 modules			7BP702.0	80
BP702.1	Module rack, for 2 modules, incl. right side section			7BP702.1	80
BP703	Module rack, for 3 modules			7BP703.0	80
BP704	Module rack, for 4 modules			7BP704.0	80
BP705	Module rack, for 5 modules			7BP705.0	80
BP706	Module rack, for 6 modules			7BP706.0	80
BP707	Module rack, for 7 modules			7BP707.0	80
BP708	Module rack, for 8 modules			7BP708.0	80
BP709	Module rack, for 9 modules			7BP709.0	80
BP710	Module rack, for 10 modules			7BP710.0	80
	Bus Controller Modules				
EX270	CAN bus controller, 24 VDC supply		+4 W	7EX270.50-1	84
EX470	CAN bus controller, 24 VDC supply		+14 W [1]	7EX470.50-1	90
EX477	Remote I/O bus controller, 24 VDC supply		+14 W	7EX477.50-2	95
EX770	CAN bus controller, 100 - 240 VAC supply		+14 W [2]	7EX770.50-1	90
EX777	Remote I/O bus controller, 100 - 240 VAC supply		+14 W	7EX777.50-1	95
	CPUs				
CP430	CPU, 24 VDC, 100 KByte SRAM, 256 KByte FlashPROM		+7 W [3]	7CP430.60-1	102
CP470	CPU, 24 VDC, 100 KByte SRAM, 256 KByte FlashPROM		+14 W [3]	7CP470.60-1	102
CP474	CPU, 24 VDC, CP Interface, 100 KByte SRAM, 512 KByte FlashPROM		+12.6 W [3]	7CP474.60-1	102
CP476	CPU, 24 VDC, CP Interface, 750 KByte SRAM, 1.5 MByte FlashPROM		+12.5 W [3]	7CP476.60-1	114
CP770	CPU, 100 - 240 VAC, 100 KByte SRAM, 256 KByte FlashPROM		+14 W [3]	7CP770.60-1	102
CP774	CPU, 100 - 240 VAC, CP Interface, 100 KByte SRAM, 512 KByte FlashPROM		+12.6 W [3]	7CP774.60-1	102
	Program Memory Modules				
ME770	Configuration memory for CAN bus controller		-0.1 W	7ME770.5	128
	Analog Interface (AF) Module				
AF101	Adapter module AF101	$C0	-0.3 W	7AF101.7	131
	Digital Input Modules				
DI135	4 inputs, 24 VDC, event counter/incremental encoder operation, 1 comparator output, 24 VDC	$12	-0.4 W	7DI135.70	140
DI435	8 inputs, 24 VDC, input delay 1 ms	$E1	-0.2 W	7DI435.7	159
DI439.7	16 inputs, 24 VDC, input delay 1 ms, termnal block connection	$ED	-0.4 W	7DI439.7	164
DI439.72	16 inputs, 24 VDC, input delay 1 ms, D-type connection	$E9	-0.4 W	7DI439.72	170
DI645	8 inputs, 100 - 240 VAC, input delay 50 ms	$E5	-0.2 W	7DI645.7	178

[1] EX470 with revision 30.xx or higher

[2] EX770 with revision 10.xx or higher

[3] Integrated power supply for simple PANELWARE controllers, e.g. P120

ID	Description	B&R ID Code	Power	Model No.	Page
	Digital Output Modules				
DO135	4 FET outputs, 12 - 24 VDC, 0.1 A, 100 kHz	$14	-0.2 W	7DO135.70	185
DO164	4 FET outputs (4 triac couplers), 48 - 125 VAC, 50 mA, zero voltage input	$3C	-0.6 W	7DO164.70	197
DO435	8 channels, each channel can be declared either as output or input Max. 8 FET outputs, 24 VDC, 2 A, max. 8 inputs, 24 VDC, inputs delay 1 ms	$EB	-0.5 W	7DO435.7	211
DO720	8 relay outputs, 240 VAC / 30 VDC, 2 A	$E2	-1.4 W	7DO720.7	219
DO721	4 relay outputs, 240 VAC / 24 VDC, 4 A	$E6	-1.4 W	7DO721.7	224
DO722	8 relay outputs, 240 VAC / 24 VDC, 2.5 A	$F2	-1.4 W	7DO722.7	228
	Digital Mixed Modules				
DM435	8 digital inputs, 24 VDC, input delay 1 ms, 8 digital outputs, 24 VDC, 0.5 A, terminal block connection	$E3	-0.5 W	7DM435.7	233
DM438	8 digital inputs, 24 VDC, input delay 1 ms, 8 digital outputs, 24 VDC, 0.5 A	$E7	-0.5 W	7DM438.72	241
DM465	16 digital inputs, 24 VDC, input delay 1 ms, 16 digital outputs, 24 VDC, 0.5 A	$F7	-1.1 W	7DM465.7	249
	Analog Input Modules				
AI261	1 Input used to evaluate a full-bridge strain gauge, 24 bit resolution	$20	-0.6 W	7AI261.7	260
AI294	4 inputs for potentiometer displacement gauge, 13 bit resolution	$22	-0.5 W	7AI294.7	277
AI351	1 input ±10 V or 0 - 20 mA, 12 bit resolution + sign	U: $02 Pot.: $02 I: $03	U/I: -0.3 W Pot.: -0.7 W	7AI351.70	285
AI354	4 inputs ±10 V, 12 bit resolution + sign	$04	-0.5 W	7AI354.70	293
AI774	4 inputs 0 – 20 mA, 12 bit resolution	$06	-0.4 W	7AI774.70	299
	Analog Output Modules				
AO352	2 outputs ±10 V / 0 - 20 mA	$0E	-1.2 W	7AO352.70	306
	Temperature Modules				
AT324	4 inputs for temperature sensors (PT100 / PT1000 / KTY10-6 / KTY84-130)	$3E	-0.1 W	7AT324.70	313
AT352	2 inputs for PT100 sensors (3-line)	$1A	-0.4 W	7AT352.70	320
AT664	4 inputs for thermocouples	$0C	-0.4 W	7AT664.70	326
	Other Modules				
CM211	8 digital inputs, 24 VDC, 20 kHz, one or two channel counter, incremental encoder, 8 digital outputs, 24 VDC, 0.5 A, comparator function, 2 analog inputs ±10 V or 0 - 20 mA, 12 bit resolution, 2 analog outputs ±10 V, 12 bit resolution	$C3	-1.5 W	7CM211.7	334
CM411	3 digital inputs, 24 VDC, 50 kHz, one or two channel counter, incremental encoder, 2 digital outputs, 24 VDC, 0.5 A, comparator function, 3 analog inputs ±10 V, 16 bit resolution, 3 analog outputs ±10 V, 16 bit resolution	$C2	-2.4 W	7CM411.70-1	384
ME010	2003 expansion for CP476, 1 PCMCIA slot	$00	-0.35 W	7ME010.9	408
ME020	2003 expansion for CP476, 1 PCMCIA slot, 1 slot for interface module inserts	$02	-0.7 W	7ME020.9	411
	Communication Modules				
IF311	Interface module with RS232 interface without PW operator panel with PW operator panel P120 / P121	$18	-0.5 W -1.6 W	7IF311.7	416
IF321	Interface module with RS485/RS422 interface	$16	-1.4 W	7IF321.7	416
IF361	Interface module with RS485 interface (PROFIBUS-DP slave)	$25	-2.6 W	7IF361.70-1	418
IF371	Interface module with CAN interface	$24	-2.5 W	7IF371.70-1	420
	Counter and Positioning Modules				
NC161	Encoder module, input frequency 100 kHz, incremental or absolute, 32 bit resolution	$10	-0.3 W $-I_{enc}$ * 5.4 V	7NC161.7	423

ID	Description	B&R ID Code	Power	Model No.	Page
	Accessories				
AC010	5 bus covers			7AC010.9	451
AC011	5 stress relief attachments including mounting material			7AC011.9	452
AC020	1 bus cover			7AC020.9	451
TB710	Single row terminal block, 10 pin, screw clamps			7TB710.9	454
TB710	Single row terminal block, 10 pin, cage clamps			7TB710.91	454
TB710	30 single row terminal blocks, 10 pin, screw clamps			7TB710:90-01	454
TB710	30 single row terminal blocks, 10 pin, cage clamps			7TB710:91-01	454
TB712	Single row terminal block, 12 pin, screw clamps			7TB712.9	455
TB712	Single row terminal block, 12 pin, cage clamps			7TB712.91	455
TB712	20 single row terminal blocks, 12 pin, screw clamps			7TB712:90-02	455
TB712	20 single row terminal blocks, 12 pin, cage clamps			7TB712:91-02	455
TB718	Single row terminal block, 18 pin, screw clamps			7TB718.9	456
TB718	Single row terminal block, 18 pin, cage clamps			7TB718.91	456
TB718	20 single row terminal blocks, 18 pin, screw clamps			7TB718:90-02	456
TB718	20 single row terminal blocks, 18 pin, cage clamps			7TB718:91-02	456
TB722	2-row terminal block, 22 pin, screw clamps			7TB722.9	457
TB722	2-row terminal block, 22 pin, cage clamps			7TB722.91	457
TB733	3-row terminal block, 33 pin, screw clamps			7TB733.9	458
TB733	3-row terminal block, 33 pin, cage clamps			7TB733.91	458
TB736	2-row terminal block, 36 pin, screw clamps			7TB736.9	459
TB736	2-row terminal block, 36 pin, cage clamps			7TB736.91	459
TB754	3-row terminal block, 54 pin, screw clamps			7TB754.9	460
TB754	3-row terminal block, 54 pin, cage clamps			7TB754.91	460
TB772	4-row terminal block, 72 pin, cage clamps			7TB772.91	461
	Manuals				
	B&R 2003 User's Manual, German			MASYS22003-0	462
	B&R 2003 User's Manual, English			MASYS22003-E	462

2 MODULE RACKS

2.1 GENERAL INFORMATION

The module racks for the B&R SYSTEM 2003 are aluminum frames. They are installed on mounting rail.

Controller and I/O modules are hung in the module rack and screwed to a threaded strip which is inserted in the aluminum frame. The electrical connection between the modules is achieved by pushing the modules together (integrated plug and socket connection in the modules).

We recommend always using the smallest possible module rack. If module slots do remain free, fitting a bus cover on the first free slot is recommended. In this way, the last module can be protected from damage.

The module address is determined by the controller and the module slot. The CP interface is accessed using module address 0.

1	2	3	4	5	6	7	8	9	10	Module slot
1	2	3	4							Module address with Ex270
	1	2	3	4	5	6	7	8		Module address with single width controller e.g. CP470, EX470
0		1	2	3	4	5	6	7	8	Module address with double width controllers (e.g. CP474) and I/O modules that require a logical slot
0		1	· ·							12	Module address with CP474 or CP774 and I/O modules that require one or two logical slots
0		1	· ·							16	Module address with Cp476 and I/O modules that require one or two logical slots

BP701
BP702
BP703
BP704
BP705
BP706
BP707
BP708
BP709
BP710

2.2 TECHNICAL DATA

2.2.1 Module Racks with Two Side Sections (7BP7xx.0)

Module ID	BP702	BP703	BP704
Model Number	7BP702.0	7BP703.0	7BP704.0
Short Description	2003 Module Racks for 2 modules	2003 Module Racks for 3 modules	2003 module racks for 4 modules
Number of Module Slots	2	3	4
C-UL-US Listed	Yes		
Material	Aluminum		
Fastening the Modules	Hang modules in the rack and screw into place (threaded strips in frame)		
Mounting the Module Rack	Integrated mounting mechanics (standard mounting rail)		
Dimensions [mm] Height Width [1] Depth	115 161.5 23	115 238, 314.5 23	115 23

[1] Includes 8.5 mm for two side sections and mounting screws.

Module ID	BP705	BP706	BP707
Model Number	7BP705.0	7BP706.0	7BP707.0
Short Description	2003 Module Racks for 5 modules	2003 Module Racks for 6 modules	2003 module racks for 7 modules
Number of Module Slots	5	6	7
C-UL-US Listed	Yes		
Material	Aluminum		
Fastening the Modules	Hang modules in the rack and screw into place (threaded strips in frame)		
Mounting the Module Rack	Integrated mounting mechanics (standard mounting rail)		
Dimensions [mm] Height Width [1] Depth	115 391, 467.5 23	115 544 23	115 23

[1] Includes 8.5 mm for two side sections and mounting screws.

Module ID	BP708	BP709	BP710
Model Number	7BP708.0	7BP709.0	7BP710.0
Short Description	2003 Module Racks for 8 modules	2003 Module Racks for 9 modules	2003 module racks for 10 modules
Number of Module Slots	8	9	10
C-UL-US Listed	Yes		
Material	Aluminum		
Fastening the Modules	Hang modules in the rack and screw into place (threaded strips in frame)		
Mounting the Module Rack	Integrated mounting mechanics (standard mounting rail)		
Dimensions [mm] Height Width [1] Depth	115 620.5 23	115 697,773.5 23	115 23

[1] Includes 8.5 mm for two side sections and mounting screws.

2.2.2 Module Racks with One Side Section (7BP70x.1)

These module racks are e.g. used together with the EX270 CAN bus controller.

Module ID	BP701	BP702
Model Number	7BP701.1	7BP702.1
Short Description	2003 Module Racks for 1 module, incline. right side section	2003 module racks for 2 modules, incl. right side section
Number of Module Slots	1	2
C-UL-US Listed	Yes	
Material	Aluminum	
Fastening the Modules	Hang modules in the rack and screw into place (threaded strips in frame)	
Mounting the Module Rack	Integrated mounting mechanics (standard mounting rail)	
Dimensions [mm] Height Width [1] Depth	115 81 23	115 157,5 23

[1] Includes 4.5 mm for one side section and mounting screws.

2.3 MOUNTING

Mounting the module racks is described in Chapter 2, "Project Planning and Installation".

3 BUS CONTROLLER MODULES

3.1 CAN BUS CONTROLLER

A CAN bus controller takes over the communication with a PCC, PLC or other CAN capable devices.

Additionally, a CAN bus controller handles the following tasks

- Initialization from power on to active operation on a network
- Evaluating and sending input states
- Sending input states cyclically and/or when the input state changes
- Receiving and switching the outputs
- Defined error reaction for network crashes and local problems
- Setting and changing the operating parameters can be done either with a special instruction which is sent from a CAN client (master) or with configuration memory.

3.2 REMOTE I/O BUS CONTROLLER

A remote I/O bus controller takes over communication between a B&R remote master (B&R SYSTEM 2005, B&R SYSTEM 2010 or B&R SYSTEM 2000 Logic Scanner) and the B&R 2003 I/O system.

The controller has the following local tasks

- Initialization from power on to active operation on a network
- Evaluating and sending input states (polled by master)
- Receiving and switching the outputs
- Defined error reaction for network crashes and local problems

3.3 OVERVIEW

Module	Description
EX270	2003 CAN bus controller, 24 VDC, 4 W supply, 1 CAN interface, electrically isolated, network capable, Order terminal block TB712 separately!
EX470	2003 CAN bus controller, 24 VDC, 14.5 W supply, 2 CAN interfaces, electrically isolated, network capable
EX477	2003 remote I/O bus controller, 24 VDC, 14 W supply, 1 electrically isolated RS485 interface for connection to remote I/O bus
EX770	2003 CAN bus controller, 100-240 VAC, 14.5 W supply, 2 CAN interfaces, electrically isolated, network capable
EX777	2003 remote I/O bus controller, 100-240 VAC, 14 W supply, 1 electrically isolated RS485 interface for connection to remote I/O bus

3.4 EX270

3.4.1 General Information

The CAN bus controller EX270 is not hung in the module rack, instead it is screwed on the module rack instead of the left side plate.

3.4.2 Technical Data

Terminal block is not included in the delivery.

Module ID	EX270
General Information	
Model Number	7EX270.50-1
Short Description	2003 CAN bus controller, 24 VDC, 4 W supply, 1 CAN interface, electrically isolated, network capable, Order terminal block TB712 separately!
C-UL-US Listed	in preparation
Module Type	B&R 2003 Controller
Module width	20 mm
Installation	The controller is screwed onto the module rack instead of the left side plate
Peripheral	
Diagnosis LED	Yes
I/O Bus Interface	9 pin D-type socket
Number Switch	Used to set the node number and baudrate

Module ID	EX270
Standard Communication Interface	

Application Interface
  Electrical Isolation
  Connection
  Max. Distance
  Max. baudrate
	CAN Interface
	Yes
	12 pin multipoint connector
	1000 m
	500 kBaud

Power Supply	

Input Voltage
  Minimum
  Nominal
  Maximum
	18 VDC
	24 VDC
	30 VDC

Power Consumption	Max. 5 W
Output Power for I/O Modules and Screw-in Module	Max. 4 W
Voltage Monitoring	The power supply is only activated starting with an input voltage of approx. +15 V. Therefore the status LED DC-OK is not required.

3.4.3 Status Display

LED	Meaning
STATUS (2 color)	
Red	Reset (Hold)
Green blinking during the boot phase	Boot phase (initialization and connection to the CAN network). If an error occurs during this phase, the green LED stops blinking. The error is indicated by periodic blinking of the red LED. In this case, a hardware reset (switch off/on) is required. The amount of pulses periodically output provides information about the cause of the error: 1 red pulse: Node switch set to 0 and configuration EEPROM is invalid 2 red pulses: Error initializing the CAN block
Green blinking with double pulse	The controller indicates when time monitoring responds and updates the digital and analog outputs. If a CAN object does not appear within the defined time (default: 640 ms), the effected outputs are reset and the green LED begins to blink (double pulse). After the first valid object arrives, the current values are immediately accepted. The green LED only returns to normal operation after a time delay of 30 s. The time delay is used to identify intermittent problems with the outputs. These problems are otherwise very difficult to recognize.
Green	Normal operation: data is being exchanged

LED	Meaning
STATUS (2 color)	
Orange	Faulty outputs. However, the CAN bus controller is still in network operation.
Orange blinking [1]	Voltage alarm on a module
Orange blinking with double pulse	The total power for the module exceeds the power supply on the EX270. The basic load on the digital and analog modules is calculated once when booting. If a digital module is found which is not entered in the module list for the operating system, there is generally no power monitoring. If the power calculation was successful, the analog modules are continually monitored during operation. One screw-in module is tested per second.
Red blinking	Warning: The node number was changed during operation. The new node number setting is ignored; the node continues to function.

[1] No longer evaluated by all digital mixed modules starting with Rev. D0.

3.4.4 Connections

The voltage supply and the CAN interface connections are made using a 12 pin terminal block.

The electrically isolated CAN interface is available twice on the plug. The individual CAN connections are linked with each other which allows a node to be easily connected to a CAN network (see section "Connection Examples").
For more information on wiring CAN field bus systems, see chapter 2, "Project Planning and Installation", section "CAN Field Bus".

The Bus controller EX270 is already equipped with a bus termination resistor. To activate it, a jumper must be placed between pin 1 and pin 2.

TB712
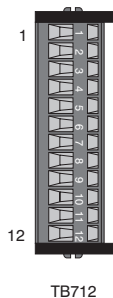
Pin	Assignment
1	Bridge for termination resistor
2	Bridge for termination resistor
3	CAN_H
4	CAN_GND
5	CAN_L
6	Shield
7	CAN_H
8	CAN_GND
9	CAN_L
10	Shield
11	Supply +24 VDC
12	Supply GND

3.4.5 Connection Examples

With Continued CAN Bus Connection

CAN_H
CAN_GND
CAN_L
Shield
CAN_H
CAN_GND
CAN_L
Shield
Supply +24 VDC
Supply GND

With Active Bus Termination Resistor

Termination resistor
active

Termination resistor
Termination resistor
CAN_H
CAN_GND
CAN_L
Shield

Supply +24 VDC
Supply GND
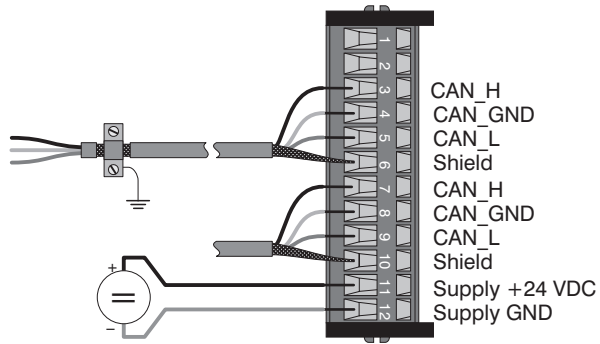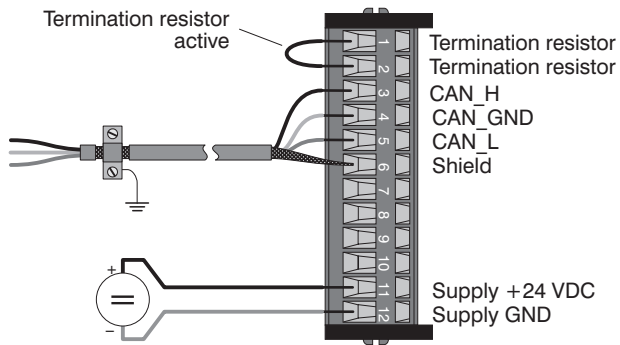
3.4.6 Node Number, Start Baudrate

The node number and start baudrate are set with the two number switches on the CAN bus controller:
Start baudrate, see section "Automatic Baudrate Recognition".

SW1
SW0
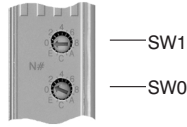
SW1	SW0	Node Number	Start Baudrate [kBit/s]
0	0	S-EEPROM	S-EEPROM
0	1 ... F	1 ... 15	250
1	0 ... F	16 ... 31	250
2	0 ... F	32 ... 47	250
3	0 ... F	48 ... 63	250
4	0	S-EEPROM	S-EEPROM
4	1 ... F	1 ... 15	125
5	0 ... F	16 ... 31	125
6	0 ... F	32 ... 47	125
7	0 ... F	48 ... 63	125
8	0	S-EEPROM	S-EEPROM
8	1 ... F	1 ... 15	20
9	0 ... F	16 ... 31	20
A	0 ... F	32 ... 47	20
B	0 ... F	48 ... 63	20
C	0	S-EEPROM	S-EEPROM
C	1 ... F	1 ... 15	500
D	0 ... F	16 ... 31	500
E	0 ... F	32 ... 47	500
F	0 ... F	48 ... 63	500

  Special Function - Node Number 0 !

If node number 0 is selected using number switch, the CAN bus controller uses the operating parameters from the internal S-EEPROM.
The S-EEPROM is programmed using the CAN Library for PG2000 and the CAN Configurator. The operating parameters are explained in chapter 5, "CAN Bus Controller Functions", section "Operating Parameters".

3.4.7 Automatic Baudrate Recognition

After booting, the bus controller EX270 goes into "Listen Only" Mode. That means the controller behaves passively on the bus and only listens.
The EX270 tries to receive valid objects. If an error occurs when receiving, the controller switches to the next baudrate from the search table.
If no objects are received, all baudrates are tested cyclically. This procedure is repeated until valid objects are received.

Start Baudrate

The bus controller begins the search with this baudrate. The start baudrate can be defined in three different ways:

- Set using the node number switch
- Read from the S-EEPROM (node number = 0)
- The last recognized baudrate is used to begin the search after a software reset (command code 20)

Search Table

The controller tests the baudrate according to this table. Beginning with the start baudrate, the baudrate is switched to the next lower value.
At the end of the table, the controller starts searching from the beginning again.

Baudrate
1000 kBaud
500 kBaud
250 kBaud
125 kBaud
50 kBaud
20 kBaud
10 kBaud

3.5 EX470 / EX770

3.5.1 Technical Data
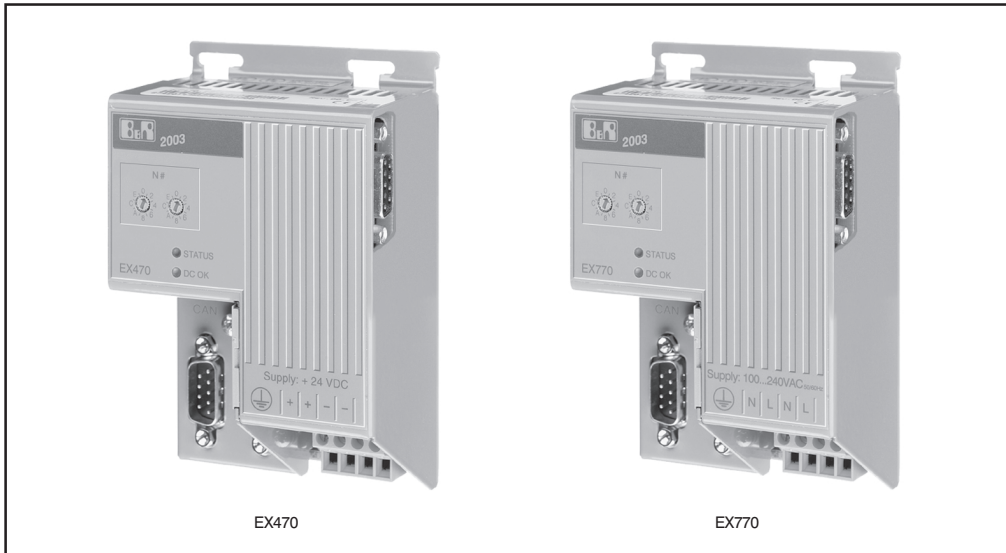
EX470   EX770

Module ID	EX470	EX770
General Information		
Model Number	7EX470.50-1	7EX770.50-1
Short Description	2003 CAN bus controller, 24 VDC, 14.5 W supply, 2 CAN interfaces electrical isolation, network capable	2003 CAN bus controller, 100-240 VAC, 14.5 W supply, 2 CAN interfaces, electrical isolation, network capable
C-UL-US Listed	Yes	
Module Type	B&R 2003 Controller	
Module width	B&R 2003 single width	
Module slot	1	
Environmental Temp. in Operation	0 - 60 °C [1] [2]	
Peripheral		
Diagnosis LEDs	Yes	
I/O Bus Interface	9 pin D-type socket	
Number Switch	Used to set the node number and baudrate	
Standard Communication Interfaces		

Application Interface IF1
  Electrical Isolation
  Connection
  Max. distance
  Max. baudrate
	CAN Interface (left)
	Yes
	9 pin D-type plug
	1000 m
	500 kBaud	

Application Interface IF2
  Electrical Isolation
  Connection
  Max. Distance
  Max. baudrate
	CAN/ID Interface (right)
	Yes
	9 pin D-type socket
	1000 m
	500 kBaud	

Module ID	EX470	EX770
Power Supply		

Input Voltage
    Minimum
    Nominal
    Maximum
	18 VDC
	24 VDC
	30 VDC
		85 VAC
		100 - 240 VAC
		264 VAC

Input Voltage Frequency	—	47 - 63 Hz
Power Consumption	Max. 20 W	
Output Power for I/O Modules and Screw-in Modules	14.5 W [1] [2]	

[1] EX470 with revision 30.xx or higher

[2] EX770 with revision 10.xx or higher

3.5.2 Status Display

LED	Meaning
STATUS (2 color)	
Red	Reset (Hold)
Green blinking during the boot phase	Boot phase (initialization and connection to the CAN network). If an error occurs during this phase, the green LED stops blinking. The error is indicated by periodic blinking of the red LED. In this case, a hardware reset (switch off/on) is required. The amount of pulses periodically output provides information about the cause of the error: 1 red pulse: Node switch set to 0 and configuration EEPROM is invalid 2 red pulses: Error initializing the CAN block
Green blinking with double pulse	Starting with rev. xx.24, the controller indicates when time monitoring responds and updates the digital and analog outputs. If a CAN object does not appear within the defined time (default: 640 ms), the effected outputs are reset and the green LED begins to blink (double pulse). After the first valid object arrives, the current values are immediately accepted. The green LED only returns to normal operation after a time delay of 30 s. The time delay is used to identify intermittent problems with the outputs. These problems are otherwise very difficult to recognize.
Green	Normal operation: data is being exchanged
Orange	Faulty outputs. However, the CAN bus controller is still in network operation.
Orange blinking [1]	Voltage alarm on a module

[1] No longer evaluated by all digital mixed modules starting with Rev. E0.

LED	Meaning
STATUS (2 color)	
Orange blinking with double pulse	The total power for the module exceeds the power supply on the bus controller.
	The basic load on the digital and analog modules is calculated once when booting. If a digital module is found which is not entered in the module list for the operating system, there is generally no power monitoring.
	If the power calculation was successful, the analog modules are continually monitored during operation. One screw-in module is tested per second.
Red blinking	Warning:
	The node number was changed during operation. The new node number setting is ignored; the node continues to function.
DC OK (orange)	
Lit	The internal power supply voltage is OK
Not lit	All outputs are reset and the entire bus node reinitialized

3.5.3 Power Supply

The CAN bus controller is available in two variants. The major difference between the two variants is the supply voltage. The power connecter pin assignments are printed on the modules:
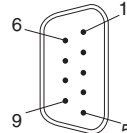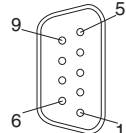
EX470	EX770
Both + pins are connected and both **−** pins are connected internally	Both **N** pins are connected and both **L** pins are connected internally

3.5.4 Interface Pin Assignments

The CAN bus controller has two interfaces:

IF1 - CAN	IF2 - CAN / ID
9 pin D-type plug	9 pin D-type socket

Pin	Assignment
1	n. c.
2	CAN_L
3	CAN_GND
4	n. c.
5	n. c.
6	res.
7	CAN_H
8	n. c.
9	n. c.

Pin	Assignment
1	res.
2	CAN_L
3	CAN_GND
4	res.
5	res.
6	res.
7	CAN_H
8	res.
9	res.

res. => These connections are reserved for a possible configuration memory.

Both interfaces are electrically isolated. The CAN assignment for both interfaces are identical. The individual CAN connections for both interfaces are connected with each other. Therefore both interfaces can be used to connect nodes in a CAN network.

However, it is recommended that you use the T-connector AC911 (see Chapter 7 "Accessories") for coupling to a CAN network. On one hand, this has the advantage that the CAN/ID interface remains free for expansion purposes (configuration memory area). On the other hand, a node can also be easily separated from the power supply, without interrupting the network. Also, a terminal resistance is integrated into the T-connector for the bus termination, which can be switched on or off.

3.5.5 Wiring

For more information on wiring CAN field bus systems, see chapter 2, "Project Planning and Installation", section "CAN Field Bus".

3.5.6 Node Number, Baudrate

The node number and baudrate are set with the two number switches on the CAN bus controller:

SW1 ——————— SW0

SW1	SW0	Node Number	Baudrate [kBit/s]
0	0	ME770	ME770
0	1 ... F	1 ... 15	250
1	0 ... F	16 ... 31	250
2	0 ... F	32 ... 47	250
3	0 ... F	48 ... 63	250
4	0	ME770	ME770
4	1 ... F	1 ... 15	125
5	0 ... F	16 ... 31	125
6	0 ... F	32 ... 47	125
7	0 ... F	48 ... 63	125
8	0	ME770	ME770
8	1 ... F	1 ... 15	20
9	0 ... F	16 ... 31	20
A	0 ... F	32 ... 47	20
B	0 ... F	48 ... 63	20
C	0	ME770	ME770
C	1 ... F	1 ... 15	500
D	0 ... F	16 ... 31	500
E	0 ... F	32 ... 47	500
F	0 ... F	48 ... 63	500

! Special Function - Node Number 0 !

If node number 0 is selected using number switch, the CAN bus controller uses the operating parameters from the configuration memory ME770. If no configuration memory is available, the CAN bus controller remains inactive, i.e. it is not initialized.
The configuration memory is programmed using the CAN Library for PG2000 and the CAN Configurator. The operating parameters are explained in chapter 5, "CAN Bus Controller Functions", section "Operating Parameters".

3.6 EX477 / EX777

3.6.1 Technical Data

EX477   EX777

Module ID	EX477	EX777
General Information		
Model Number	7EX477.50-2	7EX777.50-1
Short Description	2003 remote I/O bus controller, 24 VDC, 14 W supply, 1 electrically isolated RS485 interface for connection to remote I/O bus	2003 remote I/O bus controller, 100-240 VAC, 14 W supply, 1 electrically isolated RS485 interface for connection to remote I/O bus
C-UL-US Listed	Yes	
Module Type	B&R 2003 Controller	
Module width	B&R 2003 single width	
Module slot	1	
Peripheral		
Diagnosis LEDs	Yes	
I/O Bus Interface	9 pin D-type socket	
Number Switch	Used to set the remote address	
Standard Communication Interface		
Interface Type Connection Electrical Isolation Baudrates 100 kBit/s 181 kBit/s 500 kBit/s 1000 kBit/s 2000 kBit/s	RS485 9 pin D-type socket Yes Depends on the distance Max. 1200 m Max. 1000 m Max. 400 m Max. 200 m Max. 100 m	

Module ID	EX477	EX777

Remote I/O Bus
   Access Procedure
   Number of Slaves
   Topology
   Bus Connection
   Transfer Media
   Termination Resistor
	Master/Slave Principle
	Max. 31 (without repeater)
	Physical bus
	Direct
	Shielded, twisted pair
	External	

Power Supply		

Input Voltage
   Minimum
   Nominal
   Maximum
	18 VDC
	24 VDC
	30 VDC
		85 VAC
		100 - 240 VAC
		264 VAC

Input Voltage Frequency	—	47 - 63 Hz
Power Consumption	Max. 20 W	
Output Power for I/O Modules and Screw-in Modules	14 W	

3.6.2 Status Display

LED	Meaning
STATUS (green)	
Blinking	Boot phase: - Initialization and connection to the Remote I/O network - Automatic baudrate recognition
Lit	Normal operation: The initialization and connection to the Remote I/O network is complete, data is being exchanged with the master
Not lit (if DC OK is lit)	The connection to the Remote I/O network is complete, but data is not being exchanged with the master
DC OK (orange)	
Lit	The internal power supply voltage is OK
Not lit	All outputs are reset and the entire bus node reinitialized

3.6.3 Power Supply

The remote I/O bus controller is available in two different versions depending exclusively on their supply voltage. The power connecter pin assignments are printed on the modules:

EX477	EX777
Both + pins are connected and both – pins are connected internally	Both **N** pins are connected and both **L** pins are connected internally

3.6.4 Remote Address

The address of the remote slave station is set using number switches. Addresses in the range from 1 to 98 are allowed. However, a maximum of 31 remote slaves can be connected to a remote master (without repeater).

3.6.5 Interface Pin Assignments

9 pin D-type socket
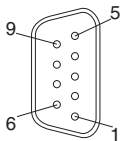
Pin	Assignment	
1	Shield	Shield connection (housing)
2	---	Reserved
3	DATA	Data
4	CNTRL	Transmit Enable (control line)
5	GND	Electrically isolated supply
6	+5 V / 50 mA	Electrically isolated supply (e.g. for bus termination)
7	---	Reserved
8	DATA\	Data\
9	CNTRL\	Transmit Enable\

B&R recommends using bus connector 0G1000.00-090 when connecting to a remote I/O network. It is available as T-connector and already has the correct termination resistor (can be switched on-/off).

3.6.6 Wiring

Wiring remote I/O networks is described in Chapter 2 "Project Planning and Installation", section "Remote I/O Bus".

3.6.7 Repeater or Fibre Optic Connection

The control signal CNTRL ("Transmit Enable") is output by the bus controller and serves to directionally control a repeater or a fibre optic connection. It is a 5 V differential signal:

CNTRL	CNTRL\	RS485
0	1	Receive (default)
1	0	Transmit

4 CPUS

4.1 GENERAL INFORMATION

The CPU is operated on the far left of the module rack. The modules has status LEDs and connection plugs for an RS232 and a CAN interface.

The CPUs CP474/CP476/CP774 are equipped with four slots for screw-in modules. The required screw-in modules are inserted into the CP interface and screwed firmly into place.

4.2 RS232 INTERFACE

The RS232 interface is primarily intended for programming the CPU. It can also be used as a general interface (e.g. P121 visualization, printer, bar code reader, etc.).

4.3 CAN INTERFACE

Standard field bus interface for communication with other control systems. Remote I/O expansion using B&R 2003 components and a CAN bus controller, e.g. EX470.

4.4 LOCAL I/O EXPANSION USING B&R 2003 MODULES

- Digital I/O (24 VDC or 120/230 VAC)
- Analog I/O (±10 V, 0 - 20 mA, thermocouple, PT100, etc.)
- Incremental encoder or SSI encoder connection (100 kHz, +5 VDC encoder supply)

4.5 LOCAL I/O ON THE CPU (4 SLOTS)

The CPUs CP474/CP476/CP774 are equipped with four slots for screw-in modules. The following screw-in modules can be inserted:

- Analog I/O Modules
- Incremental encoder or SSI encoder connection (B&R 2003)
- Digital I/O (24 VDC technology)
- Additional serial interfaces (max. 3 channels)

4.6 APPLICATION EXAMPLES

- Counting and measuring digital signals (4 MHz)
- Positioning (analog)
- Stepper motor control
- Drum sequencer
- Pulse width modulated outputs
- Reaction of digital outputs to input events in the μs range

4.7 PROGRAMMING

The CPU is programmed using the programming system PG2000 or Automation Studio™. Several programming languages are available:

PG2000	Automation Studio™
Instruction List (IL)	Automation Basic (previously PL2000)
Ladder Diagram (LAD)	ANSI C
PL2000 High Level Language (structured text)	IEC 1131 Ladder Diagram (LAD)
	IEC 1131 Sequential Function Chart (SFC)
	IEC 1131 Structured Text (ST)
	IEC 1131 Instruction List (IL)

4.8 BACKUP BATTERY

Data buffering and nonvolatile operation of the real-time clock are guaranteed by the lithium battery provided.

4.9 PROGRAMMING THE FLASHPROM

Programming and deleting the built-in FlashPROM memory takes place using the programming system.

4.10 LEGEND SHEETS

A legend sheet can be slid into the front of the CPUs CP474/CP476/CP774 from above. These sheets can be used for labelling the screw in modules.

4.11 OVERVIEW

Module	Description
CP430	2003 CPU, 100 KB SRAM, 256 KB FlashPROM, 24 VDC, 7 W supply, 1 RS232 interface, 1 CAN interface, CAN: electrically isolated, network capable, max. 64 digital / 32 analog I/O points
CP470	2003 CPU, 100 KB SRAM, 256 KB FlashPROM, 24 VDC, 14 W supply, 1 RS232 interface, 1 CAN interface, CAN: electrically isolated, network capable, max. 128 digital / 64 analog I/O points
CP474	2003 CPU, 100 KB SRAM, 512 KB FlashPROM, 24 VDC, 12.6 W supply, 1 RS232 interface, 1 CAN interface, CAN: electrically isolated, network capable, 4 slots for screw-in modules, max. 208 digital / 80 analog I/O points
CP476	2003 CPU, 750 KB SRAM, 1.5 MB FlashPROM, 24 VDC, 12.5 W supply, 1 RS232 interface, 1 CAN interface, CAN: electrically isolated, network capable, 4 slots for screw-in modules, system bus for expansion modules, max. 272 digital / 80 analog I/O points
CP770	2003 CPU, 100 KB SRAM, 256 KB FlashPROM, 100-240 VAC, 14 W supply, 1 RS232 interface, 1 CAN interface, CAN: electrically isolated, network capable, max. 128 digital / 64 analog I/O points
CP774	2003 CPU, 100 KB SRAM, 512 KB FlashPROM, 100-240 VAC, 12.6 W supply, 1 RS232 interface, 1 CAN interface, CAN: electrically isolated, network capable, 4 slots for screw-in modules, max. 208 digital / 80 analog I/O points

4.12 CP430, CP470, CP474, CP770, CP774

4.12.1 Order Data

CP430, CP470, CP770   CP474, CP774
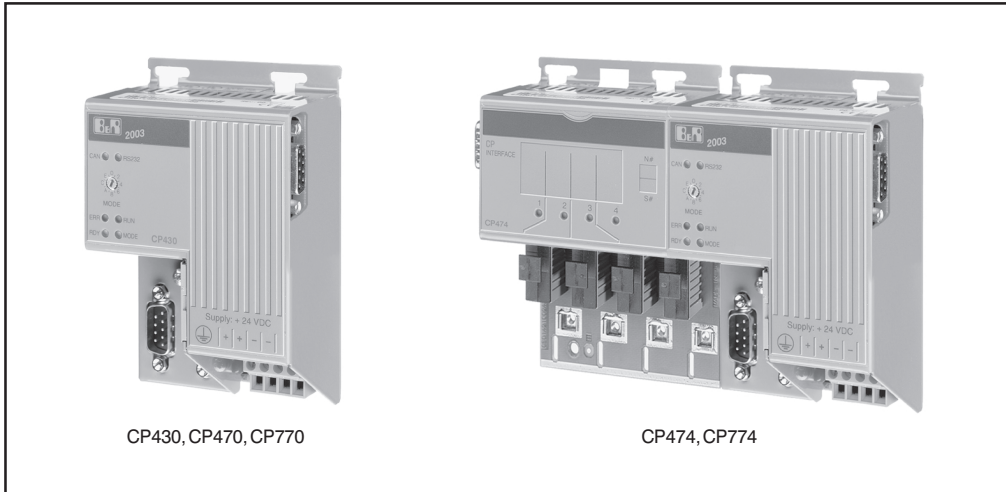
Model Number	Short Description
7CP430.60-1	2003 CPU, 100 KB SRAM, 256 KB FlashPROM, 24 VDC, 7 W supply, 1 RS232 interface, 1 CAN interface, CAN: electrically isolated, network capable, max. 64 digital / 32 analog I/O points
7CP470.60-1	2003 CPU, 100 KB SRAM, 256 KB FlashPROM, 24 VDC, 14 W supply, 1 RS232 interface, 1 CAN interface, CAN: electrically isolated, network capable, max. 128 digital / 64 analog I/O points
7CP474.60-1	2003 CPU, 100 KB SRAM, 512 KB FlashPROM, 24 VDC, 12.6 W supply, 1 RS232 interface, 1 CAN interface, CAN: electrically isolated, network capable, 4 slots for screw-in modules, max. 208 digital / 80 analog I/O points
7CP770.60-1	2003 CPU, 100 KB SRAM, 256 KB FlashPROM, 100-240 VAC, 14 W supply, 1 RS232 interface, 1 CAN interface, CAN: electrically isolated, network capable, max. 128 digital / 64 analog I/O points
7CP774.60-1	2003 CPU, 100 KB SRAM, 512 KB FlashPROM, 100-240 VAC, 12.6 W supply, 1 RS232 interface, 1 CAN interface, CAN: electrically isolated, network capable, 4 slots for screw-in modules, max. 208 digital / 80 analog I/O points

4.12.2 Technical Data

Module ID	CP430	CP470/CP770	CP474/CP774
General Information			
C-UL-US Listed	in preparation	Yes	Yes
Module Type	B&R 2003 CPU	B&R 2003 CPU	B&R 2003 CPU
Module Width	B&R 2003 single width	B&R 2003 single width	B&R 2003 double width
Module Slot	1	1	1 + 2

Module ID	CP430	CP470/CP770	CP474/CP774
Processor Section			
Command Cycle Time	Average value with 70 % bit and 30 % analog processing		
	1.6 μs	1.6 μs	0.8 μs

Standard Memory
  User RAM
  System PROM
  User PROM
	100 KByte SRAM
	256 KByte FlashPROM
	256 KByte FlashPROM
		100 KByte SRAM
		256 KByte FlashPROM
		256 KByte FlashPROM
			100 KByte SRAM
			512 KByte FlashPROM
			512 KByte FlashPROM

Data Buffering
  Backup Battery
  Buffer Current
	Lithium battery 3 V / 950 mAh		

Typical Maximum	1.6 μA 60 μA	1.6 μA 60 μA	2.2 μA 110 μA
HW Watchdog	Yes		
Voltage Monitoring	The internal supply is monitored for overvoltage and undervoltage		
Peripheral			
Real-time Clock Resolution	Nonvolatile 1 s		
Status Display	LEDs		
I/O Bus Interface	9 pin D-type socket		

Slots for Screw-in Modules
  Suitable for
  IF Modules
	No	No	4
			
			1 - 3

Standard Communication Interfaces			

Application Interface IF1
  Electrical Isolation
  Design
  Max. Distance
  Max. Baudrate
	RS232
	No
	9 pin D-type plug
	15 m / 19200 Baud
	115.2 kBaud		

Application Interface IF2
  Electrical Isolation
  Design
  Max. Distance
  Max. Baudrate
	CAN
	Yes
	9 pin D-type plug
	1000 m
	500 kBaud		

Power Supply	**CP430**	**CP470/CP474**	**CP770/CP774**

Input Voltage
  Minimum
  Nominal
  Maximum
	18 VDC
	24 VDC
	30 VDC
		18 VDC
		24 VDC
		30 VDC
			85 VAC
			100 - 240 VAC
			264 VAC

Input Voltage Frequency	---	---	47 - 63 Hz
Power Consumption	Max. 9.5 W	Max. 20 W	Max. 20 W
Output Power for I/O Ports	7 W [1]	14 W [1] 12.6 W [1]	14 W [1] 12.6 W [1]

[1] Integrated power supply on pin 4 of the RS232 interface for simple PANELWARE controllers, e.g. P120.

4.12.3 Status Display

LED	Meaning
CAN	Data transfer to or from CAN controller
RS232	Indicates if data is being transmitted or received
ERR	Lit in Service mode
RUN	Lit in RUN and in Service mode
RDY	Leuchtet im Service-Modus
MODE	Lit when programming FlashPROM
1, 2, 3, 4	These LEDs show the operating state of the respective screw-in module.
Not lit	Screw-in module defective or not inserted
Blinking slowly	Communication error with screw-in module
Blinking quickly	Screw-in module is new or has been exchanged with another module type
Lit	Screw-in module is ready for operation

4.12.4 Power Supply

The CPUs are either supplied with 24 VDC or with 100 to 240 VAC. The pin assignments are printed on the module.

CP430, CP470, CP474	CP770, CP774
Both + pins are connected and both – pins are connected internally	Both N pins are connected and both L pins are connected internally

4.12.5 Interfaces

The CPU has two interfaces:

CAN   RS232

4.12.6 CAN Bus

The electrically isolated standard field bus interface is used for the following tasks:

- Communication with other control systems

- Remote I-O expansion using B&R 2003 components and a CAN bus controller

It is recommended that you use the T-connector AC911 (see Chapter 7 "General Accessories") when coupling to a CAN network. A terminal resistance is integrated into the T-connector for the bus termination, which can be switched on or off.
For more information on wiring CAN field bus systems, see chapter 2, "Project Planning and Installation", section "CAN Field Bus".

9 pin D-type plug
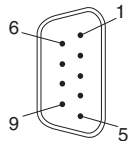
Pin	Assignment
1	n. c.
2	CAN_L
3	CAN_GND
4	n. c.
5	n. c.
6	res.
7	CAN_H
8	n. c.
9	n. c.

4.12.7 RS232 Interface

This non-electrically isolated interface is primarily intended for programming the CPU. The RS232 interface can also be used as a general interface (e.g. P121 visualization, printer, bar code reader, etc.).

9 pin D-type plug
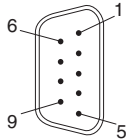
Pin	Assignment	
1	n. c.	Reserved
2	RXD	Receive Signal
3	TXD	Transmit Signal
4	+5 VDC / max. 500 mA	Panel Supply
5	GND	Ground
6	n. c.	Reserved
7	RTS	Request To Send
8	CTS	Clear To Send
9	n. c.	Reserved

4.12.8 MODE Switch

The operating mode is set with the MODE switch. The switch setting can be evaluated by the application program at any time. If the switch position is changed during operation, a warning can be generated. The operating system only interprets the switch position when switched on.

Switch Position	Description
0	Programming System Flash (see respective section)
1 - 8	Freely available for use in an application (e.g. CAN node number)
9 - E	Reserved for B&R expansions – these settings are **not** allowed to be used!
F	Diagnose mode

4.12.9 Programming System Flash

General Information

The CPUs are delivered with an operating system. The operating system can be updated using the programming system.

An operating system update is possible using PG2000 V 2.41 and higher.

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Operating System Update

When updating the operating system, the following procedure must be followed:

1) Establish online connection between programming device (PC or Industrial PC) and the CPU.

2) Start PG2000 programming system.

3) In PG2000, call the function *PCCSW Update* (see menu item *Service* in the pull-down menu *System*).

4) A dialog box is shown where you can set the transfer rate (baudrate) for the update procedure and the PC interface used for the online connection (e.g. 57600 Baud, COM1).

5) Another dialog box is opened when you select [OK].

6) The operating system version can be selected in this dialog box. After closing this dialog box by selecting [Yes], the system ROM is deleted (incl. operating system). Then the selected operating system version is transferred to the system ROM. The update progress is shown in the message line.

The User Flash will be deleted!

7) Switch PCC off and then on again.

8) The PCC is now ready for use.

The operating system update is not only possible via an online connection, it is also possible via a CAN network or a serial network (INA2000 protocol).

4.12.10 CP Interface

The CPUs CP474 and CP774 are equipped with four slots for screw-in modules. The required screw-in modules are inserted into the CP interface and screwed firmly into place.

The screw-in interface modules can be used with slots 1, 2 and 3.

Overview

The following screw-in modules can be used on the CP Interface.

Module	Type	Description
7AI261.7	Analog IN	1 input to evaluate a full-bridge strain gauge
7AI294.7	Analog IN	4 inputs for potentiometer displacement gauge
7AI351.70	Analog IN	1 x ±10 V or 1 x 0 - 20 mA (1 x ±20 mA also possible) Potentiometer operation
7AI354.70	Analog IN	4 x ±10 V
7AI774.70	Analog IN	4 x 0 - 20 mA (4 x ±20 mA also possible)
7AO352.70	Analog OUT	2 x ±10 V / 0 - 20 mA
7AT324.70	Analog IN	4 x Temperature sensor (PT100, PT1000, KTY10 or KTY84)
7AT352.70	Analog IN	2 x PT100 3-wire
7AT664.70	Analog IN	4 x Thermocouple
7DI135.70	Digital IN	4 x 24 VDC, 50 kHz
7DO135.70	Digital OUT	4 x 12 - 24 VDC, 0.1 A, 100 kHz
7DO164.70	Digital OUT	4 x 48 - 125 VAC, 50 mA, nonvolatile input
7IF311.7	Interface	1 x RS232
7IF321.7	Interface	1 x RS485/RS422
7IF361.70-1	Interface	1 x PROFIBUS DP-Slave
7IF371.70-1	Interface	1 x CAN
7NC161.7	Encoder Module	1 x 100 kHz, 5 / 24 VDC

Commands

The following commands can be used on the CP Interface:

- Reading the screw-in module type
- Switching off automatic mode
- Switching on automatic mode

The commands are described in section "AF101".

4.12.11 Legend Sheets

A legend sheet can be slid into the front of the CPUs CP474 and CP774 from above. These sheets can be used for labelling the screw in modules.

4.12.12 Data/Real-time Clock Buffering

The battery voltage is checked cyclically. The cyclic load test of the battery does not considerably shorten the battery life, instead it gives an early warning of weakened buffer capacity.
The status information, "Battery OK" is available from the B&R-TRAP function, "SYS_battery".

4.12.13 System Variable SYS2003

General Information

The system variable SYS2003 is a structure containing the elements "io_scan" and "io_refresh". It must be declared in a task as PCC global.

Element	Variable Type	Description
io_scan	INT16	Duration of the last I/O cycle in µs
io_refresh	INT8	0 .... I/O data is older than one cycle 1 .... I/O data is current

> ⚠️ **If digital I/O variables are used in the HSTC, the system variable SYS2003 will also be placed in the HSTC. Values in lower task classes will therefore not be consistent.**
> **If no digital I/Os are placed in the HSTC, the SYS2003 variable will use the 10 ms operating system clock.**

4.12.14 Exchanging the Battery

Battery Data

Lithium Battery	3 V / 950 mAh
Model Number	0AC201.9 (5 lithium batteries)
Storage Time	Max. 3 years at 30 °C
Relative Humidity	0 to 95 % (non-condensing)

Buffer Duration

Buffer Current	CP470 / CP770	CP474 / CP774
Typical	1.6 µA	2.2 µA
Maximum	60 µA	110 µA

⚠ **B&R recommend changing the batteries after five years of operation.**

Procedure

The product design allows battery changing to be carried out with the PCC switched on of off. In some countires, safety regulations do not allow batteries to be changed while the module is switched on.

⚠ **Data stored in RAM will be lost if the battery is changed with the PCC switched off!**

1) Touch the mounting rail or earth connection (not the power supply!) in order to discharge any electrostatic charge from your body.

2) Remove the cover from the lithium battery holder using a screwdriver.

3) Remove the battery from the holder by pulling the removal strip (don't use uninsulated tools -> risk of short circuiting).
The battery should not be held by its edges. **Insulated** tweezers may be used for removing the battery.

Right: Wrong:

4) Insert the new battery with correct polarity. The removal strip should be protruding from the battery holder and the "+" side of the battery should be facing downward.
In order to be able to remove the battery again in future, the removal strip **must** protrude from the **upper side** of the battery.

Removal strip must protrude from the upper side of the battery

5) Now wrap the end of the removal strip over the top of the battery and insert it underneath the battery so that it does not protrude from the battery holder.

6) Replace cover. Ensure that the slot in the edge of the cover faces the front of the module ❶.
Insert the upper edge of the cover in the battery holder opening ❷. Press the lower end of the cover home firmly ❸.

Lithium batteries are considered hazardous waste! Used batteries should be disposed of accordingly.

4.13 CP476

4.13.1 General Information

With the addition of the CP476, the B&R SYSTEMS 2003 now has a CPU with outstanding technical properties.

Features

- 750 KByte User SRAM
- 1.5 MByte User FlashPROM
- Additional I/O processor
- System bus for expansions
- CP Interface with four slots
- Two node number switches

Memory capacity was increased to meet the requirements of our customers.

The CP476 extends the CPU performance spectrum for the B&R SYSTEM 2003 in the upward direction. Compared to the CP474, the new CPU increases performance by at least a 50 % because the clock frequency is 50 % higher and the integration of an additional I/O processor. This reduces the interrupt load considerably and decreases the analog update time on the left side by max. the factor 2 (screw-in modules 1 and 2 as well as 3 and 4 are processed parallel to each other). The I/O update on the left side tales place independently of the right side.

The CPU CP476 is equipped with a system bus for expansions. The following expansions can be connected:

- ME010 expansion module for PCMCIA memory cards
- ME020 expansion module for PCMCIA memory cards and a slot for B&R SYSTEM 2005 interface module inserts

Two CAN node number switches guarantee that an offset no longer has to be set. The actual node number always corresponds to the switch setting.

4.13.2 Technical Data

Module ID	CP476
General Information	
Model Number	7CP476.60-1
Short Description	2003 CPU, 750 KB SRAM, 1,5 MB FlashPROM, 24 VDC, 12,5 W supply, 1 RS232, 1 CAN interface, CAN: electrical isolation, network capable, 4 slots for screw-in modules, system bus for expansion modules, max. 272 digital / 80 analog I/O points
C-UL-US Listed	in preparation
Module Type	B&R 2003 CPU
Module Width	B&R 2003 double width
Module Slot	1 + 2
Processor Section	
Additional I/O Processor	Handles I/O data points
Instruction Cycle Time (average value for 70 % bit and 30 % analog processing)	$0.5 \mu s$
Standard Memory 　User RAM 　System PROM 　User PROM	750 KByte SRAM 512 KByte FlashPROM 1.5 MByte FlashPROM
Data Buffering 　Backup Battery 　Buffer Current 　　Typical 　　Maximum	Lithium Battery 3 V / 950 mAh 2.2 μA 110 μA
HW Watchdog	Yes
Voltage Monitoring	The internal supply is monitored for overvoltage and undervoltage

Module ID	CP476
Peripheral	
Real-time Clock Resolution	Nonvolatile 1 s
Status Display	LEDs
I/O Bus Interface (right side)	9 pin D-type socket
System Bus for Expansions (left side)	PCMCIA memory card ME010 B&R SYSTEM 2005 interface module insert ME020
Slots for Screw-in Modules Suitable for IF Modules	4 1 - 3
Standard Communication Interfaces	

Application Interface IF1
    Electrical Isolation
    Connection
    Max. Distance
    Max. Baudrate
	RS232
	No
	9 pin D-type plug
	15 m / 19200 Baud
	115.2 kBaud

Application Interface IF2
    Electrical Isolation
    Connection
    Max. Distance
    Max. Baudrate
	CAN
	Yes
	9 pin D-type plug
	1000 m
	500 kBaud

Power Supply	

Input Voltage
    Minimum
    Nominal
    Maximum
	18 VDC
	24 VDC
	30 VDC

Power Consumption	Max. 20 W
Output Power for I/O Ports	12.5 W [1]

[1] Integrated power supply on pin 4 of the RS232 interface for simple PANELWARE controllers, e.g. P120.

4.13.3 Status Display

LED	Meaning
CAN	Data transfer to or from CAN controller
RS232	Indicates if data is being transmitted or received
ERR	Lit in Service mode
RUN	Lit in RUN and in Service mode
RDY	Lit in Service mode
MODE	Lit when programming FlashPROM
1, 2, 3, 4	These LEDs show the operating state of the respective screw-in module.
Not lit	Screw-in module defective or not inserted
Blinking slowly	Communication error with screw-in module
Blinking quickly	Screw-in module is new or has been exchanged with another module type
Lit	Screw-in module is ready for operation

4.13.4 Power Supply

The CPU CP476 is supplied with 24 VDC. The pin assignments are printed on the module. Both **+** pins are connected and both **–** pins are connected internally.

4.13.5 Interfaces

The CPU has two interfaces:

CAN   RS232

4.13.6 CAN Bus

The electrically isolated standard field bus interface is used for the following tasks:

- Communication with other control systems

- Remote I-O expansion using B&R 2003 components and a CAN bus controller

It is recommended that you use the T-connector AC911 (see Chapter 7 "General Accessories") when coupling to a CAN network. A terminal resistance is integrated into the T-connector for the bus termination, which can be switched on or off.
For more information on wiring CAN field bus systems, see chapter 2, "Project Planning and Installation", section "CAN Field Bus".

9 pin D-type plug
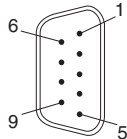
Pin	Assignment
1	n. c.
2	CAN_L
3	CAN_GND
4	n. c.
5	n. c.
6	res.
7	CAN_H
8	n. c.
9	n. c.

4.13.7 RS232 Interface

This non-electrically isolated interface is primarily intended for programming the CPU. The RS232 interface can also be used as a general interface (e.g. P121 visualization, printer, bar code reader, etc.).

9 pin D-type plug
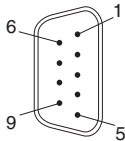
Pin	Assignment	
1	n. c.	Reserved
2	RXD	Receive Signal
3	TXD	Transmit Signal
4	+5 VDC / max. 500 mA	Panel Supply
5	GND	Ground
6	n. c.	Reserved
7	RTS	Request To Send
8	CTS	Clear To Send
9	n. c.	Reserved

4.13.8 CAN Node Number Switch

CAN-ID Nr.:

The CAN node number is set with the two hex switches. The switch setting can be evaluated by the application program at any time. If the switch position is changed during operation, a warning can be generated. The operating system only interprets the switch position when switched on.
The settings 00, FD and FF are reserved for special functions.

Switch Position	Description
00	Programming System Flash (see respective section)
FD	Update MEMCARD (see ME010 and ME020)
FF	Diagnose mode

4.13.9 Programming System Flash

General Information

The CPUs are delivered with an operating system. The operating system can be updated using the programming system.

An operating system update is possible using PG2000 V 2.41 and higher.

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Operating System Update

When updating the operating system, the following procedure must be followed:

1) Establish online connection between programming device (PC or Industrial PC) and the CPU.

2) Start PG2000 programming system.

3) In PG2000, call the function *PCCSW Update* (see menu item *Service* in the pull-down menu *System*).

4) A dialog box is shown where you can set the transfer rate (baudrate) for the update procedure and the PC interface used for the online connection (e.g. 57600 Baud, COM1).

5) Another dialog box is opened when you select [OK].

6) The operating system version can be selected in this dialog box. After closing this dialog box by selecting [Yes], the system ROM is deleted (incl. operating system). Then the selected operating system version is transferred to the system ROM. The update progress is shown in the message line.

⚠️ **The User Flash will be deleted!**

7) Switch PCC off and then on again.

8) The PCC is now ready for use.

⚠️ **The operating system update is not only possible via an online connection, it is also possible via a CAN network or a serial network (INA2000 protocol).**

4.13.10 CP Interface

The CPU is equipped with four slots for screw-in modules. The required screw-in modules are inserted into the CP interface and screwed firmly into place.

To reduce the load on the CPU, an additional I/O processor handles operation of the all I/O data points on the CP interface and the I/O bus interface (right side).
The I/O processor handles slots 1 and 3 as well as 2 and 4 in succession (multiplex operation). That means slots 1 and 2 as well as 3 and 4 are operated parallel to each other.

The screw-in interface modules can be used with slots 1, 2 and 3.

Overview

The following screw-in modules can be used on the CP Interface.

Module	Type	Description
7AI261.7	Analog IN	1 input to evaluate a full-bridge strain gauge
7AI294.7	Analog IN	4 inputs for potentiometer displacement gauge
7AI351.70	Analog IN	1 x ±10 V or 1 x 0 - 20 mA (1 x ±20 mA also possible) Potentiometer operation
7AI354.70	Analog IN	4 x ±10 V
7AI774.70	Analog IN	4 x 0 - 20 mA (4 x ±20 mA also possible)
7AO352.70	Analog OUT	2 x ±10 V / 0 - 20 mA
7AT324.70	Analog IN	4 x Temperature sensor (PT100, PT1000, KTY10 or KTY84)
7AT352.70	Analog IN	2 x PT100 3-wire
7AT664.70	Analog IN	4 x Thermocouple
7DI135.70	Digital IN	4 x 24 VDC, 50 kHz
7DO135.70	Digital OUT	4 x 12 - 24 VDC, 0.1 A, 100 kHz
7DO164.70	Digital OUT	4 x 48 - 125 VAC, 50 mA, nonvolatile input
7IF311.7	Interface	1 x RS232
7IF321.7	Interface	1 x RS485/RS422
7IF361.70-1	Interface	1 x PROFIBUS DP-Slave
7IF371.70-1	Interface	1 x CAN
7NC161.7	Encoder Module	1 x 100 kHz, 5 / 24 VDC

Commands

The following commands can be used on the CP Interface:

- Reading the screw-in module type
- Switching off automatic mode
- Switching on automatic mode

The commands are described in section "AF101".

4.13.11 Legend Sheets

A legend sheet can be slid into the front of the CPU CP476 from above. These sheets can be used for labelling the screw in modules.

4.13.12 Data/Real-time Clock Buffering

The battery voltage is checked cyclically. The cyclic load test of the battery does not considerably shorten the battery life, instead it gives an early warning of weakened buffer capacity.
The status information, "Battery OK" is available from the B&R-TRAP function, "SYS_battery".

4.13.13 System Variable SYS2003

General Information

The system variable SYS2003 is a structure containing the elements "io_scan" and "io_refresh". It must be declared in a task as PCC global.

Element	Variable Type	Description
io_scan	INT16	Duration of the last I/O cycle in μs
io_refresh	INT8	0 .... I/O data is older than one cycle 1 .... I/O data is current

If digital I/O variables are used in the HSTC, the system variable SYS2003 will also be placed in the HSTC. Values in lower task classes will therefore not be consistent.
If no digital I/Os are placed in the HSTC, the SYS2003 variable will use the 10 ms operating system clock.

4.13.14 Exchanging the Battery

Battery Data

Lithium Battery	3 V / 950 mAh
Model Number	0AC201.9 (5 lithium batteries)
Storage Time	Max. 3 years at 30 °C
Relative Humidity	0 to 95 % (non-condensing)

Buffer Duration

Buffer Current	CP476
Typical	2.2 µA
Maximum	110 µA

⚠ **B&R recommend changing the batteries after five years of operation.**

Procedure

The product design allows battery changing to be carried out with the PCC switched on of off. In some countires, safety regulations do not allow batteries to be changed while the module is switched on.

⚠ **Data stored in RAM will be lost if the battery is changed with the PCC switched off!**

1) Touch the mounting rail or earth connection (not the power supply!) in order to discharge any electrostatic charge from your body.

2) Remove the cover from the lithium battery holder using a screwdriver.

3) Remove the battery from the holder by pulling the removal strip (don't use uninsulated tools -> risk of short circuiting).
The battery should not be held by its edges. **Insulated** tweezers may be used for removing the battery.

Right: Wrong:

4) Insert the new battery with correct polarity. The removal strip should be protruding from the battery holder and the "+" side of the battery should be facing downward.
In order to be able to remove the battery again in future, the removal strip **must** protrude from the **upper side** of the battery.

Removal strip must protrude from the upper side of the battery

5) Now wrap the end of the removal strip over the top of the battery and insert it underneath the battery so that it does not protrude from the battery holder.

6) Replace cover. Ensure that the slot in the edge of the cover faces the front of the module ❶. Insert the upper edge of the cover in the battery holder opening ❷. Press the lower end of the cover home firmly ❸.

Lithium batteries are considered hazardous waste! Used batteries should be disposed of accordingly.

5 PROGRAM MEMORY MODULES

5.1 OVERVIEW

Module	Description
ME770	2003 configuration memory for CAN bus controller

5.2 ME770

5.2.1 General Information

All operating parameters for a node are stored in nonvolatile configuration memory. When switched on, the node is initialized with the parameters in the configuration memory.

Data is taken from configuration memory if,

a) The node number set on the module corresponds to the one stored, or

b) the node number on the module die is set to 0.

⚠️ **Only insert or remove the configuration memory from the controller without power applied.**

5.2.2 Technical Data

Module ID	ME770
Model Number	7ME770.5
Short Description	2003 configuration memory for CAN bus controller
C-UL-US Listed	Yes
Memory Type	4 KBit S-EEPROM

Programming
  ME770
  Programmed using
	Must be connected to CAN bus controller
	CAN Library for PG2000
	CAN Configurator

Interface Connection	9 pin D-type plug
Power Consumption	0.1 W

6 ANALOG INTERFACE (AF) MODULE

6.1 GENERAL INFORMATION

Screw-in modules are installed either on the CP interface or on the adapter module and then screwed into place:
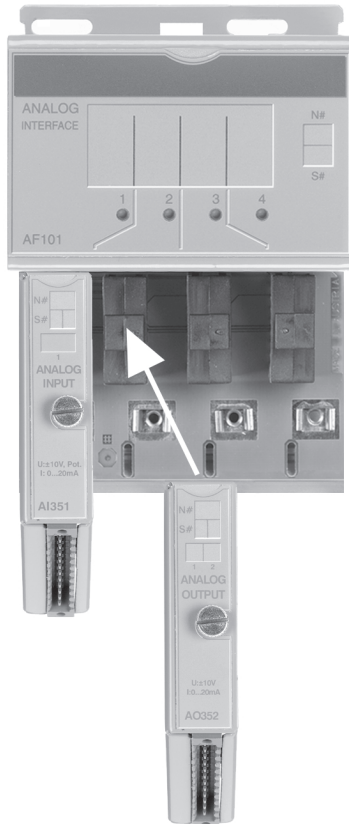
6.2 OVERVIEW

Module	Description
AF101	2003 adapter module, 4 slots for screw-in modules

6.3 SCREW-IN MODULES

The following screw-in modules can be used on the adapter module:

Module	Type	Description
7AI261.7	Analog IN	1 Input used to evaluate a full-bridge strain gauge
7AI294.7	Analog IN	4 inputs for potentiometer displacement gauge
7AI351.70	Analog IN	1 x ±10 V or 1 x 0 - 20 mA (1 x ±20 mA also possible), potentiometer operation
7AI354.70	Analog IN	4 x ±10 V
7AI774.70	Analog IN	4 x 0 - 20 mA (4 x ±20 mA also possible)
7AO352.70	Analog OUT	2 x ±10 V / 0 - 20 mA
7AT324.70	Analog IN	4 x temperature sensor (PT100, PT1000, KTY10 or KTY84)
7AT352.70	Analog IN	2 x PT100 3-line
7AT664.70	Analog IN	4 x thermocouple
7DI135.70	Digital IN	4 x 24 VDC, 50 kHz
7DO135.70	Digital OUT	4 x 12 - 24 VDC, 0.1 A, 100 kHz
7DO164.70	Digital OUT	4 x 48 - 125 VAC, 50 mA, nonvolatile input
7NC161.7	Encoder module	1 x 100 kHz, 5 / 24 VDC

6.4 AF101

6.4.1 Technical Data
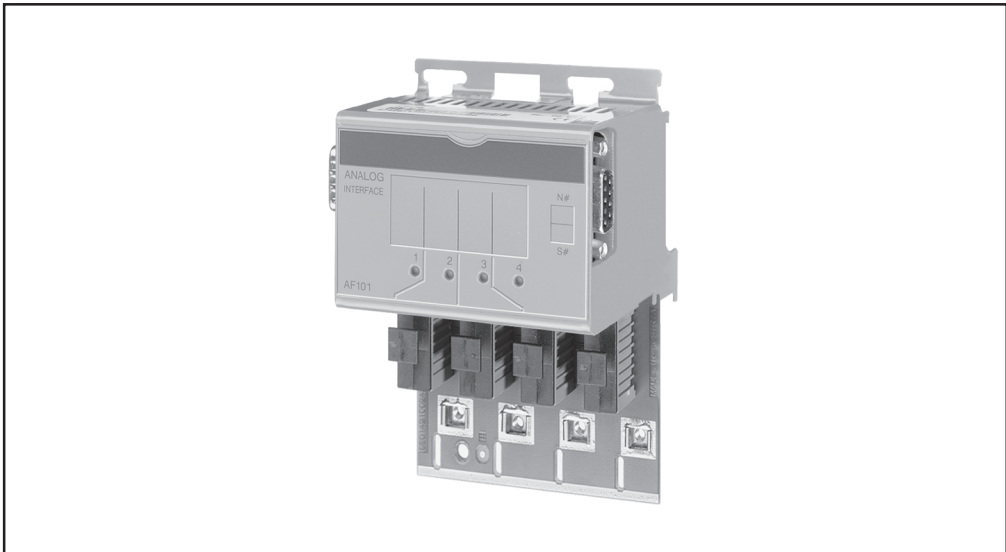
Module ID	AF101
General Information	
Model Number	7AF101.7
Short Description	2003 Adapter Module 4 slots for screw-in modules
C-UL-US Listed	Yes
B&R ID Code	$C0
Amount	
CP430, EX270	2
CP470, CP770 CP474, CP476, CP774 EX470, EX770 EX477, EX777	4
Adapter Interface	4 slots for screw-in modules
Electrical Isolation	No
Status LEDs	For each channel
Power Consumption	Max. 0.3 W
Mechanical Properties	
Dimensions	B&R 2003 single width

⚠️ **Always operate the adapter module directly to the right of the CPU!**

Adapter modules are only allowed to be used in the first four module slots on the B&R 2003 system.

6.4.2 Status LEDs

The green status LEDs on the adapter modules indicate the operational status of each slot:

LED Status	Meaning
Not lit	Screw-in module defective or not inserted
Blinking slowly	Communication error with screw-in module
Blinking quickly	Screw-in module is new or has been exchanged with another module type
Lit	Screw-in module is ready for operation

6.4.3 Legend Sheets

The legend sheets supplied with the module are slid into the module front from the top of the module. These sheets can be used for labelling the screw in modules.

6.4.4 Accessing the Screw-in Modules

Every module description contains a "variable declaration" section. This section outlines the method for declaring variables in PG2000.
The channel number to be accessed is given in the VD Channel column with the data or configuration words. When accessing configuration words using I/O function blocks, an offset value must be added to the VD channel, according to which slot the screw-in module occupies on the AF101.

Data Word

When data is accessed using a data word, an offset does not need be added to the VD channel number, since the slot number is given in the variable declaration.

Configuration Word

When accessing configuration words using I/O function blocks, an offset value must be added to the VD channel, according to which slot the screw-in module occupies on the AF101.
The internal offset used to address the required configuration word, is calculated using the following formula:

Internal offset $=$ VD channel $+$ (sl - 1) x 32

Internal offset .... Internal offset used to access the required configuration word

VD channel .... The channel entered in the variable declaration

sl .... Slot number (1 - 4) of the screw-in module on the AF101

Refer to the example in Chapter 4, "Module Addressing".

6.4.5 Commands - AF101

Controller

The commands are available to the following controllers/access types:

Controller / Access Type	Command available
CPU - PCC 2003	Yes
Remote Slaves	No
CAN Slaves	No
Access using CAN Identifier	No

Command Execution

I/O function blocks are used to execute commands on the AF101. A shovel instruction must be generated for reading configuration word 14 (VD channel 28) of a screw-in module.
The variable used to store the command response is initialized with the command word. After successful execution of the shovel instruction, the command response is saved in the variable.

The AF101 commands *Automatic mode on/off* are available from revision 02.00 of the AF101 software. The software version of the AF101 module can be determined from the last two digits of the version number shown on the serial number label (Rev. xx.11 -> SW revision 01.10, Rev. xx.20 -> SW revision 02.00).

1) Read B&R ID code for the screw-in module

The variable must be initialized with 0 in order read the B&R ID code. Don't forget to enter the channel offset for the respective slot!

Command Word	$0000	
Command Response	$xxyy	xx ....B&R ID Code yy ....Reserved

2) Read B&R ID code for the AF101 module

Command Word	$8000	
Command Response	$C000	B&R ID Code AF101

3) Automatic mode

Starting with software revision 02.00 of the AF101, the data word will be written by default to an automatically generated I/O image (automatic mode). Read commands are executed by accessing this I/O image.

Special applications of screw-in modules sometimes require the automatic I/O image generation to be switched off (e.g. for the AI261 this makes certain that the scaled value or calibrated raw value is only read once per data output cycle).

When the I/O image generator is switched off, the commands for editing the data words are written directly to the screw-in modules instead.

Automatic mode is not available on AF101 modules with a software revision <02.00. These modules behave in the same way as modules with a software revision ≥02.00 where automatic mode has been switched off.

3.1) Switching off automatic mode

This command is only available to AF101 modules with a SW revision ≥ 02.00 (AF101 modules with SW revision < 02.00 behave in the same way as modules with a SW revision ≥02.00 where automatic mode has been switched off).

Command Word	$8400	
Command Response	$F400	Command executed - AF101 SW Rev. ≥02.00
	$C000	AF101 SW Rev. <02.00

3.2) Switching on automatic mode

This command is only available to AF101 modules with a SW revision ≥02.00. Automatic mode is not available on AF101 modules with a software revision <02.00.

Command Word	$8500	
Command Response	$F500	Command executed - AF101 SW Rev. ≥02.00
	$C000	AF101 SW Rev. <02.00

6.4.6 Program Example

In this program, the shovel instructions for the following commands are defined and executed in the INIT-SP :

- Read B&R ID code for the screw-in module
- Read B&R ID code for the AF101 module
- Switching off automatic mode

Task Overview

The application program is divided into the following tasks:

I/O Library

Before using the function blocks to handle configuration words, you must import the I/O library (starting with V 01.60) into the project database. The function, *Import Library* in the *File* menu is provided for this purpose. The I/O library must be entered in the GDM as a system module.

AF101

This task processes the screw-in modules on the AF101 adapter module.
The shovel instructions required for carrying out commands on the AF101 module are defined in the INIT-SP.

INIT-SP

```
Err_Ptest = 0
AF101_slot = 1
AT664_slot = 1

;#############################################################################
;## Multipler for physical shovel instructions to read                    ##
;## parameter word 14 for the AT664 on the AF101, Slot 1                   ##
;#############################################################################
IO_struct.io_type=5                      ;2003 IO
IO_struct.master_no=1                    ;Master always 1
IO_struct.slave_no=0                     ;Slave number always 0
IO_struct.module_adr=AF101_slot          ;Module slot
IO_struct.intern_off= 32*(AT664_slot -1)+ 14 *2 ;Register no. 14 on ScrM = AT664_slot
IO_struct.mode=%00100000                 ;Read, normal
IO_struct.data_len=2                     ;2*Byte
IO_struct.reserve=0                      ;Not used
IO_struct.data_adr=adr(AT_reg14)         ;Address of the data area containing the
                                         ;image for parameter word 14
AT_reg14 = $0000                         ;0 required to read the module type


;Declare shovel instruction #0
IO_mphydef(1,adr(IO_struct),adr(af_sc_buf),0,status_mp1,af_ident)

if status_mp1=0 then
  loop          ;Carry out shovel instruction cyclically...
    IO_data(1,af_ident,status_dat)
    exitif status_dat <> 5559    ;...until shovelling is no longer active
  endloop
endif

if status_dat=0 then
    AT664_typ = AT_reg14
endif


;#############################################################################
;## Multipler for physical shovel instructions to read                    ##
;## parameter word 14 (=code) on the AF101 module                         ##
;#############################################################################
IO_struct.io_type=5                      ;2003 IO
IO_struct.master_no=1                    ;Master always 1
IO_struct.slave_no=0                     ;Slave number always 0
IO_struct.module_adr=AF101_slot          ;Module slot
IO_struct.intern_off= 14 *2              ;Register no. 14 on the AF101
IO_struct.mode=%00100000                 ;Read, normal
IO_struct.data_len=2                     ;2*Byte
IO_struct.reserve=0                      ;Not used
IO_struct.data_adr=adr(AF_reg14)         ;Address of the data area containing the
                                         ;image for parameter word 14
AF_reg14 = $8000                         ;Read AF101 code

IO_mphydef(1,adr(IO_struct),adr(af_sc_buf),0,status_mp2,af_ident)

if (status_mp2=0) then
  loop           ;Carry out shovel instruction cyclically...
    IO_data(1,af_ident,status_dat)
    exitif status_dat <> 5559    ;...until shovelling is no longer active
  endloop
endif
```

```
if status_dat=0 then
    AF_code = AF_reg14
endif

;###########################################################################
;##                    AF101 Switch off automatic mode                    ##
;###########################################################################

;Shovel instruction can be used further
AF_reg14 = $8400                    ;Command switch off automatic mode
AUTO = 1     ;Automatic mode still active

if (status_mp2=0) then
  loop              ;Carry out shovel instruction cyclically...
    IO_data(1,af_ident,status_dat)
    exitif status_dat <> 5559     ;...until shovelling is no longer active
  endloop

  if (status_dat=0) and (AF_reg14=$F400) then
    ;Automatic mode switched off successfully
    AUTO = 0        ;Automatic mode inactive
  else if (status_dat=0) and (AF_reg14=$C000) then
    ;AF101 SW Rev. < 2.00  -> No automatic mode
    AUTO = 0        ;Automatic mode inactive
  endif
endif
```

Variable Declaration

Data Type

A structure will be defined for the variable „IO_struct" using the data type *typedef* . Once the structure name *IO_struct* has been entered, the structure elements can be entered in a dialog box.

7 DIGITAL INPUT MODULES

7.1 GENERAL INFORMATION

Digital input modules are used to convert the binary signals for a process into the internal signal level required for the PCC. The states of the digital inputs are shown with status LEDs. The relevant differences in characteristics for the input modules are:

- Number of inputs
- Input voltage
- Input delay (filter)
- Special functions (e.g. counter inputs)

7.2 INPUT FILTER

An input filter is available for each input. The input delay is listed in the respective technical data section. Disturbance pulses which are shorter than input delay are suppressed by the input filter.

t_{delay} ... Input delay
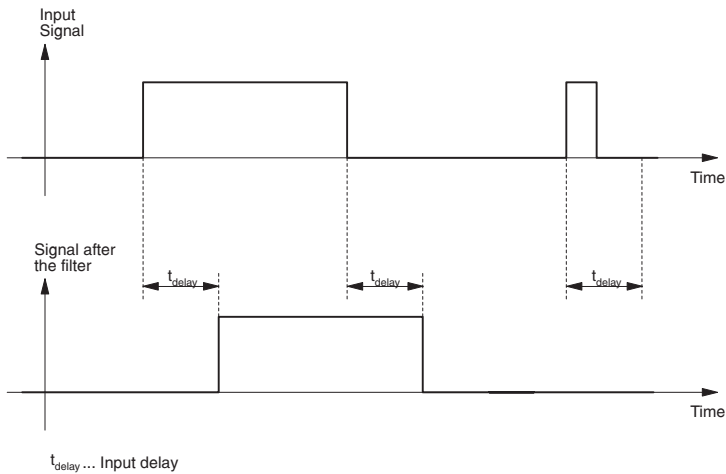
7.3 SCREW-IN MODULE OVERVIEW

Module	DI135
Number of Inputs	4
Nominal Input Voltage	24 VDC

Input Frequency
   Incremental Encoder Operation
   Event Counter Operation
	
	50 kHz
	100 kHz

Number of Outputs	1
Nominal Switching Voltage	24 VDC
Continuous Current	0.5 A
Switching Frequency	Max. 20 kHz with resistive load

7.4 I/O MODULE OVERVIEW

Module	DI435	DI439.7	DI439.72	DI645
Number of Inputs	8	16	16	8
Input Voltage	24 VDC	24 VDC	24 VDC	100 - 240 VAC
Input Delay	1 ms	1 ms	1 ms	50 ms
Remarks		Conn.: Pin connector	Conn.: D-type plug	

7.5 PROGRAMMING

The digital inputs are accessed directly in the application program using a variable name. The relationship between the input channel for a certain module and the variable name is defined in the variable declaration. The declaration is made identically for each programming language using a table editor.

7.6 DI135

7.6.1 Technical Data

Module ID	DI135
General Information	
Model Number	7DI135.70
Short Description	2003 digital input module, 4 inputs 24 VDC, sink, incremental encoder operation: 50 kHz, event counter operation: 100 kHz, 1 comparator output 24 VDC, screw-in module, Order terminal block TB712 separately!
C-UL-US Listed	Yes
B&R ID Code	$12
Module Type	B&R 2003 screw-in module
Slot	AF101 adapter module, CP interface
Power Consumption	Max. 0.4 W
Inputs	
Number of Inputs	4
Wiring	Sink

Input Voltage
  Minimum
  Nominal
  Maximum
	
	18 VDC
	24 VDC
	30 VDC

Input Frequency
  Incremental Encoder Operation
  Event Counter
	
	50 kHz
	100 kHz

Switching Threshold
  LOW
  HIGH
	
	<5 V
	>15 V

Input Delay	Max. 3 μs (at 18 - 30 V)
Input Current at Nominal Voltage	8 mA
Electrical Isolation	Input - PCC

Module ID	DI135
Incremental Encoder	
Signal Form	Square wave pulse
Evaluation	4-fold
Input Frequency	50 kHz
Count Frequency	200 kHz
Phase Offset between Channel A and B	90° ±25°
Counter Size	32 Bit
Inputs 　Input 1 　Input 2 　Input 3 　Input 4	 Channel A Channel B Reference pulse R Reference enable switch ENR
Event Counter	
Signal Form	Square wave pulse
Input Frequency	100 kHz
Counter Size	2 x 16 Bit
Inputs 　Input 1 　Input 2	 Counter 1 Counter 2
Outputs	
Number of Outputs	1
Operating Voltage 　Minimum 　Nominal 　Maximum	 18 VDC 24 VDC 30 VDC
Continuous Current	0.5 A
Maximum Switching Frequency	20 kHz with resistive load
Negative Anode Potential	45 V to 55 V
Diagnosis	Reverse polarity protection, short-circuit protection, software monitor status [1]
Electrical Isolation	Output - PCC
Mechanical Characteristics	
Dimensions	B&R 2003 screw-in module

[1] Incremental encoder operation:　　Module status in data word 0

　Event Counter Operation:　　Module status in data word 2

7.6.2 Four Digital High Speed Inputs

All four inputs are sent to TPU. If the module is inserted e.g. in slot 1 of the CP interface on a CP474, input 1 can be evaluated using the LTX function LTXdi0().

Performance Characteristics

- Counting and measuring digital signals (internal measuring frequency 4 MHz)
- Gate measurement
- Frequency measurement
- Event counting
- Incremental encoder operation
- Reaction to input events in the μs range
- Local counter status monitoring with direct output control

7.6.3 24 V Incremental Encoder / Encoder Signal Evaluation

Inputs 1 and 2 correspond to encoder signals A and B. The encoder signal is generally processed using 4-fold evaluation. The scan time (count generation) is less than 1 ms, but respective hardware allows an input frequency of 50 kHz.

Position Description:
- 32 bit (Long) and status word including referencing bit (data consistency)

Local Referencing Support:
- Clear counter without conditions (directly after receiving command)
- Clear counter after receiving the reference pulse
- Clear counter after receiving the first reference pulse with active reference enable switch

Input 3 is used as a reference signal if required.

Input 4 is used as a reference enable switch if required.

7.6.4 Event Counter

- 16-Bit counter, counts every rising edge or both edges on input 1
- 16-Bit counter, counts every rising edge or both edges on input 2
- The counters are cyclic (..., $FFFE, $FFFF, $0000, $0001, ...) and can be cleared separately on command

7.6.5 Comparator

Local counter status monitoring with direct output control (+24 VDC/0.5 A) and a reaction time of 500 μs. The comparator is either assigned to event counter 2 (16 Bit, Input 2) or to the position counter (encoder operation).
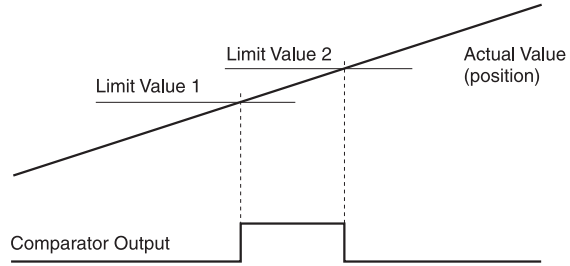
7.6.6 Input/Output Diagram
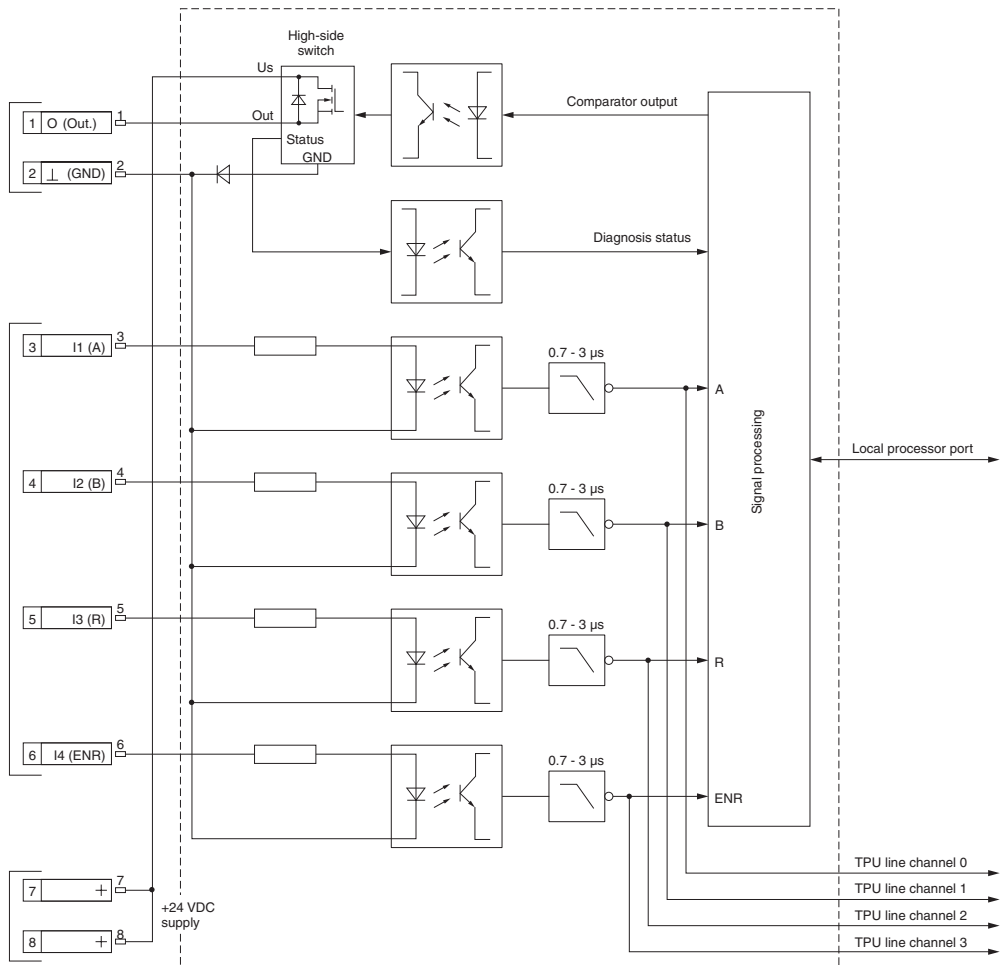
7.6.7 Connections
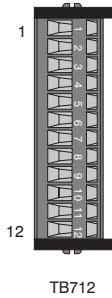
Pin	Assignment
1	Output (comparator)
2	GND
3	Shield
4	Input 1 (encoder signal A)
5	Input 2 (encoder signal B)
6	Shield
7	Input 3 (encoder signal R)
8	Input 4 (reference enable switch ENR)
9	Shield
10	+24 VDC for output or encoder
11	+24 VDC for output or encoder
12	+24 VDC for output or encoder

TB712

7.6.8 Connection Examples

Wiring Example Inputs/Outputs
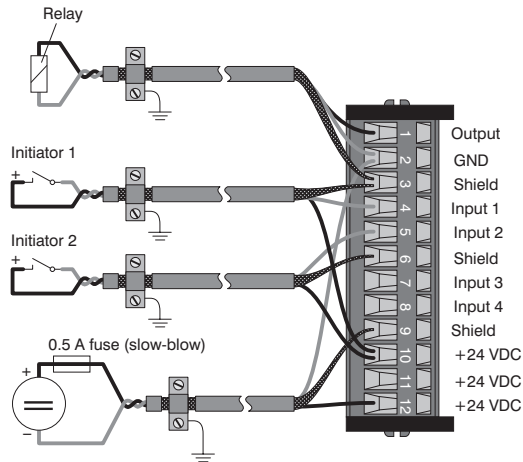
Supply:

+24 VDC, Protect with 0.5 A slow-blow fuse!

Wiring Example Incremental Encoder (encoder connection)
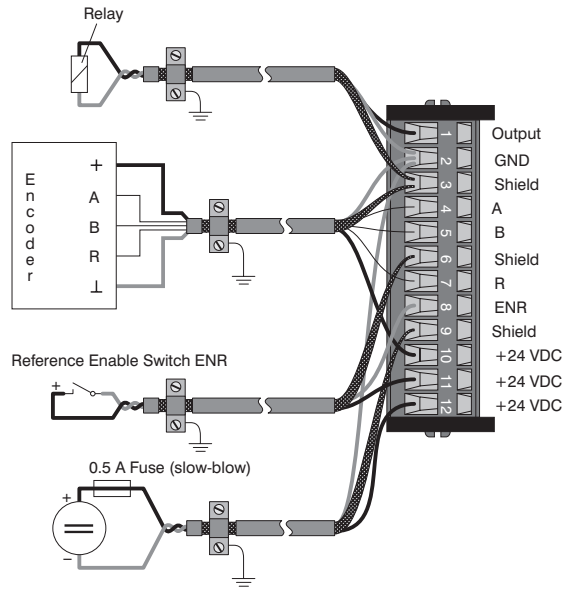
Supply:

+24 VDC, Protect with 0.5 A slow-blow fuse!

7.6.9 Variable Declaration for Incremental Encoder Operation

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Incremental Encoder Operation with PCC 2003 and Remote Slaves

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for this module.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	WORD	Transp. In	0	●		Module status
Data word 1	INT32	Transp. In	2	●		Counter value
Configuration word 4	INT32	Transp. In	8	●		Counter value at positive edge of the reference enable switch
	INT32	Transp. Out	8		●	Threshold value 1
Configuration word 6	INT32	Transp. In	12	●		Counter value at negative edge of the reference enable switch
	INT32	Transp. Out	12		●	Threshold value 2
Configuration word 8	WORD	Transp. Out	16		●	Incremental encoder / comparator control
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

Incremental Encoder Operation with CAN Slaves

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for this module.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	INT32	Transp. In	0	●		Counter value
Data word 2	WORD	Transp. In	4	●		Module status
Configuration word 4	INT32	Transp. In	8	●		Counter value at positive edge of the reference enable switch
	INT32	Transp. Out	8		●	Threshold value 1
Configuration word 6	INT32	Transp. In	12	●		Counter value at negative edge of the reference enable switch
	INT32	Transp. Out	12		●	Threshold value 2
Configuration word 8	WORD	Transp. Out	16		●	Incremental encoder / comparator control
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

> ⚠ **B&R 2000 users have to exchange the two counter status words so that the high word is first (Motorola format)!**

Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

Data cannot be packed on the DI135. Therefore one CAN object is transferred per screw-in module.
If an adapter module AF101 is equipped with a four DI135 modules, the CAN object has the following structure:

Slot	CAN ID [1]	Word 1		Word 2		Word 3		Word 4
1	542	Counter LL	Counter ML	Counter MH	Counter HH	Status L	Status H	free
2	543	Counter LL	Counter ML	Counter MH	Counter HH	Status L	Status H	free
3	544	Counter LL	Counter ML	Counter MH	Counter HH	Status L	Status H	free
4	545	Counter LL	Counter ML	Counter MH	Counter HH	Status L	Status H	free

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address of the AF101 = 1

sl .... Slot number of the screw-in module on the AF101 (1 - 4)

> ⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

Description of Data and Configuration Words

Data Word 0 (read)

Data word 0 includes the module status time constant for the counter value.

	Bit	Description
	12 - 15	x ....not defined, masked out
	11	0....Counter reading not accepted 1....Counter reading when the first positive edge arrives from Reference Enable Switch
	10	0....Counter reading not accepted 1....Counter reading when the first negative edge arrives from the reference enable switch
	8 - 9	x ....not defined, masked out
	7	0....Referencing is in progress 1....Counter is referenced (reset occurs after receiving a reference command)
	6	changes state after each successful reference
	5	0....Comparator Output: No error 1....Comparator Output: An overload error can only be indicated when the comparator output is set.
	4	Comparator output state
	3	Level of the encoder input A
	2	Level of the encoder input B
	1	Level of the reference enable switch
	0	1) Bit 0 in configurations word 14 = 0 Level of reference pulses 2) Bit 0 in configurations word 14 = 1 Level of the connection between reference pulse and reference enable switch Normal and inverted reference pulses must be differentiated. The change over for bit 2 occurs in configuration word 14 (refer to the explanation that follows).

x	x	x	x			x	x								
15						8	7								0

Normal Reference Pulse:

⚠ **Path/state and timing diagrams for incremental encoder operation are shown for the NC161 encoder module. These diagrams are also valid for the DI135.**

Bit 0 in Data Word 0 is always 1 when the level of the reference enable switch is 0.
Bit 0 only takes the level of the reference pulse when the level of the reference enable switch is 1.

Bit 0 = Input 3 or (not Input 4)

Input 3 Reference Pulse	Input 4 Reference Enable Switch	Bit 0 Data Word 0
0	0	1
1	0	1
0	1	0
1	1	1

Inverted Reference Pulse:

Bit 0 in data word 0 is always 1 when the level of the reference enable switch is 0.
Bit 0 only takes the level of the inverted reference pulse when the state of the reference enable switch is 1.

Bit 0 = (not Input 3) or (not Input 4)

Input 3 Reference Pulse	Input 4 Reference Enable Switch	Bit 0 Data Word 0
0	0	1
1	0	1
0	1	1
1	1	0

Data Word 1 (read)

Counter Value MSW

Data Word 2 (read)

Counter Value LSW

Configuration Words 4+5 (read)

After setting bit 11 in configuration word 8, the configuration words receive the latched counter value with the first positive edge of the reference enable switch. The value is valid if bit 11 is set in data word 0. Please refer to the timing diagram "Latching the Counter Value" in section "NC161".

Configuration Words 4+5 (write)

Threshold value 1 (32 Bit)

Threshold value 1 must always be ≤ threshold value 2 .
Threshold values are internally arranged in increasing order **including sign**.

Configuration Words 6+7 (read)

After setting bit 10 in configuration word 8, the configuration words receive the latched counter value with the first negative edge of the reference enable switch. The value is valid if bit 10 is set in data word 0. Please refer to the timing diagram "Latching the Counter Value" in section "NC161".

Configuration Words 6+7 (write)

Threshold value 2 (32 Bit)

Configuration Word 8 (write)

Incremental encoders and comparators are configured using configuration word 8.

	Bit	Description
	12 - 15	0
	11	0 ... Counter reading cannot be taken 1 ... Counter reading taken when the first positive edge arrives from the reference enable switch (see configuration words 4 and 5)[1]
	10	0 ... Counter reading cannot be taken 1 ... Counter reading taken when the first negative edge arrives from the reference enable switch (see configuration words 6 and 7) [1]
	5 - 9	0
	4	0 ... No effect on counter 1 Counter reset (referenced) The counter will be reset depending on the controller signal in configuration word 14 when positive edge of bit4 arrives. Bit 4 must then be reset and set again before a new reference can be taken.
	3	0 ... Comparator off The comparator output will be set to level defined in bit 0. 1 ... Comparator on
	2	0 ... Comparator output unconditional The comparator output will be set to the level defined in Bit 0, if Threshold Value 1 < Counter≤ Threshold Value 2 1 ... Comparator output conditional Reference enable switch = 1 The comparator output will be treated as "comparator output unconditional". Reference enable switch = 0 The compatator output will be set to the level defined in bit 0.
	1	0
	0	Level of comparator output

0	0	0	0			0	0	0	0	0				0	
15							8	7						0	

[1] The counter value is only taken once. Bit 10 and bit 11 must be reset for the value to be taken again. Bit 10 and bit 11 can be set again in configuration word 8 after the corresponding bit in the module status bit has gone to 0.

Configuration Word 12 (read)

Configuration word 12 contains the module status (current status unlatched). The module status is written to data word 0.

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

		Bit	Description
		8 - 15	Module code = $12
		0 - 7	x.... Not defined, masked out

0	0	0	1	0	0	1	0	x	x	x	x	x	x	x	x

15   8 7   0

Configuration Word 14 (write)

The module is configured using configuration word 14.

		Bit	Description
		13 - 15	0
		12	0 ... Incremental encoder operation 1 ... Event counter operation
		6 - 11	0
		5	0 ... No influence on counter direction 1 ... Reversed counter direction
		3 - 4	0
		2	0 ... No influence on reference pulse 1 ... Reference pulse is inverted. This change is used for encoders with high pulses.
		1	0 ... Counter set immediately to 0. In data word 0 (module status), bit 7 is set directly to1 and the counter reset. 1 ... Counter continues to function. In data word 0 (module status) bit 7 is set directly to 0 (limited reference). If the reference pulse is recorded, bit 7 in data word 0 will be set to 1 depending on the status of bit 0 in configuration word 14, and reset again with a positive edge in configuration 14.
		0	0 ... Reference enable switch ignored (referenced with reference pulse) 1 ... Reference enable switch activated (reference pulse and reference enable switch)

0	0	0	0	0	0	0	0	0	0		0	0			

15   8 7   0

⚠ **Path/state and timing diagrams for incremental encoder operation are shown for the NC161 encoder module. These diagrams are also valid for the DI135.**

7.6.10 Variable Declaration for Event Counter Operation

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Event Counter Operation

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for this module.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	WORD	Transp. In	0	●		Counter value of counter 2
Data word 1	WORD	Transp. In	2	●		Counter value of counter 1
Data word 2	WORD	Transp. In	4	●		Module status
Configuration word 4	WORD	Transp. In	8	●		Counter value of counter 1 at positive edge of the reference enable switch
Configuration word 5	WORD	Transp. In	10	●		Counter value of counter 2 at positive edge of the reference enable switch
	WORD	Transp. Out	10		●	Threshold value 1 for counter 2
Configuration word 6	WORD	Transp. In	12	●		Counter value of counter 1 at negative edge of the reference enable switch
Configuration word 7	WORD	Transp. In	14	●		Counter value of counter 2 at negative edge of the reference enable switch
	WORD	Transp. Out	14		●	Threshold value 2 for counter 2
Configuration word 8	WORD	Transp. Out	16		●	Counter / comparator control
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer. Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing". The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

Data cannot be packed on the DI135. Therefore one CAN object is transferred per screw-in module. If an adapter module AF101 is equipped with a four DI135 modules, the CAN object has the following structure:

Slot	CAN ID [1]	Word 1		Word 2		Word 3		Word 4
1	542	Counter 2L	Counter 2H	Counter 1L	Counter 1H	Status L	Status H	free
2	543	Counter 2L	Counter 2H	Counter 1L	Counter 1H	Status L	Status H	free
3	544	Counter 2L	Counter 2H	Counter 1L	Counter 1H	Status L	Status H	free
4	545	Counter 2L	Counter 2H	Counter 1L	Counter 1H	Status L	Status H	free

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address of the AF101 = 1

sl .... Slot number of the screw-in module on the AF101 (1 - 4)

⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

Description of Data and Configuration Words

Data Word 0 (read)

Counter value of counter 2.

Data Word 1 (read)

Counter value of counter 1.

<u>Data Word 2 (read)</u>

Data word 2 includes the module status time constant for both counter values.

Bit	Description
12 - 15	x ....Not defined, masked out
11	0 ....Counter status not accepted 1 ....Counter status when the first positive edge of the reference enable switch arrives.
10	0 ....Counter status not accepted 1 ....Counter status when the first negative edge of the reference enable switch arrives.
6 - 9	x ....Not defined, masked out
5	0 ....Comparator output: no error 1 ....Comparator output: Overload error indicated.
4	Output status of the comparator
3	Level encoder input A: Counter 1
2	Level encoder input B: Counter 2
1	Level of the reference enable switch
0	Level of reference pulses

x	x	x	x			x	x	x	x					

15   8 7   0

<u>Configuration Word 4 (read)</u>

After setting bit 11 in configuration word 8, the configuration word receives the latched counter value of counter 1 with the first positive edge of the reference enable switch. The value is valid if bit 11 is set in data word 2.
Please refer to the timing diagram "Latching the Counter Value" in section "NC161".

<u>Configuration Word 5 (read)</u>

After setting bit 11 in configuration word 8, the configuration word receives the latched counter value of counter 2 with the first positive edge of the reference enable switch. The value is valid if bit 11 is set in data word 2.
Please refer to the timing diagram "Latching the Counter Value" in section "NC161".

<u>Configuration Word 5 (write)</u>

Threshold value 1 (16 Bit) for counter 2.

Threshold value 1 must always be $\leq$ threshold value 2 .
Threshold values are internally arranged in increasing order.

<u>Configuration Word 6 (read)</u>

After setting bit 10 in configuration word 8, the configuration word receives the latched counter value of counter 1 with the first negative edge of the reference enable switch. The value is valid if bit 10 is set in data word 2.
Please refer to the timing diagram "Latching the Counter Value" in section "NC161".

Configuration Word 7 (read)

After setting bit 10 in configuration word 8, the configuration word receives the latched counter value of counter 2 with the first negative edge of the reference enable switch. The value is valid if bit 10 is set in data word 2.
Please refer to the timing diagram "Latching the Counter Value" in section "NC161".

Configuration Word 7 (write)

Threshold value 2 (16 Bit) for counter 2.

Configuration Word 8 (write)

Event counters and comparators are configured using configuration word 8.

	Bit	Description
	12 - 15	0
	11	0....Counter reading cannot be taken 1....Counter reading taken when the first positive edge arrives from the reference enable switch (see configuration words 4 and 5)
	10	0....Counter reading cannot be taken 1....Counter reading taken when the first negative edge arrives from the reference enable switch (see configuration words 6 and 7)
	6 - 9	0
	5	0....No effect on counter 2 1....Reset counter 2 immediately
	4	0....No effect on counter 1 1....reset counter 1 immediately
	3	0....Comparator off The comparator output will be set to level defined in bit 0. 1....Comparator on
	2	0....Comparator output unconditional The comparator output will be set to the level defined in Bit 0, if Threshold Value 1 < Counter$\leq$ Threshold Value 2 1....Comparator output conditional Reference enable switch = 1 The comparator output will be treated as "comparator output unconditional". Reference enable switch = 0 The compatator output will be set to the level defined in bit 0.
	1	0
	0	Level of the comparator output

0	0	0	0			0	0	0	0					0	
15						8	7							0	

Configuration Word 12 (read)

Configuration word 12 contains the module status (current status unlatched). The module status is written to data word 0.

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

		Bit	Description
		8 - 15	Module code = $12
		0 - 7	x .... Not defined, masked out

| 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | x | x | x | x | x | x | x | x |

15   8   7   0

Configuration Word 14 (write)

The module is configured using configuration word 14.

		Bit	Description
		13 - 15	0
		12	0.... Encoder operation 1.... Event counter
		6 - 11	0
		5	0.... Normal count direction 1.... Reverse count direction
		4	0
		3	0.... Only count positive edges 1.... Count both edges
		0 - 2	0

| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | | 0 | 0 | 0 |

15   8   7   0

7.7 DI435

7.7.1 Technical Data
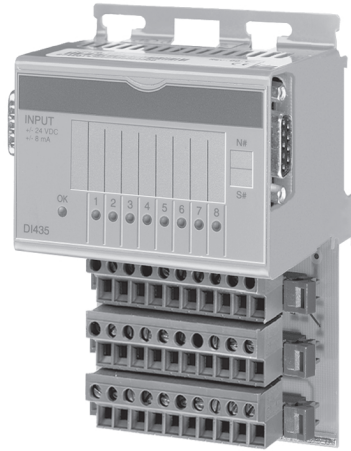
Terminal block is not included in the delivery.

Module ID	DI435
General Information	
Model Number	7DI435.7
Short Description	2003 digital input module, 8 inputs 24 VDC, 1 ms, sink/source, Order terminal blocks separately!
C-UL-US Listed	Yes
B&R ID Code	$E1
Amount	
CP430, EX270	4
CP470, CP770 CP474, CP476, CP774 EX470, EX770 EX477, EX777	8
Static Characteristics	
Module Type	B&R 2003 I/O Module
Number of Inputs	8
Wiring	Sink or source
Input Voltage Minimum Nominal Maximum	18 VDC 24 VDC 30 VDC
Switching Threshold LOW HIGH	<5 V >15 V
Input Delay	Max. 1 ms (at 18 - 30 V)
Input Current at Nominal Voltage	Approx. 8 mA (sink/source)

Module ID	DI435
Voltage Monitoring (LED: U-OK)	Yes Supply voltage >18 V
Power Consumption	Max. 0.2 W
Operating Characteristics	
Electrical Isolation	Input - PCC
Mechanical Characteristics	
Dimensions	B&R 2003 single width

7.7.2 Status LEDs

The status-LEDs 1 - 8 (green) show the logical state of the corresponding inputs.

The LED OK (orange) indicates that the input supply voltage is present. The LED is lit for input voltages from 15 to 18 VDC.

7.7.3 Input Circuit Diagram
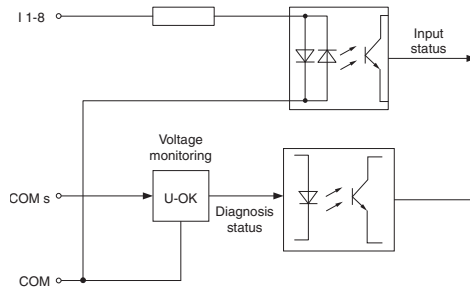
7.7.4 Legend Sheets

A legend sheet can be slid into the front of the module from above. The module circuit is shown on the back. The inputs can be labelled on the front.
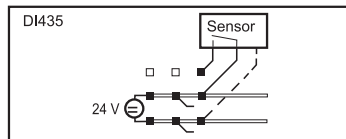
7.7.5 Sink/Source Wiring

The input module DI435 can be connected either as a sink or source circuit.

Sink Connection

For sink circuits, the COM lines are connected to GND and the inputs are connected to 24VDC switching sensors.
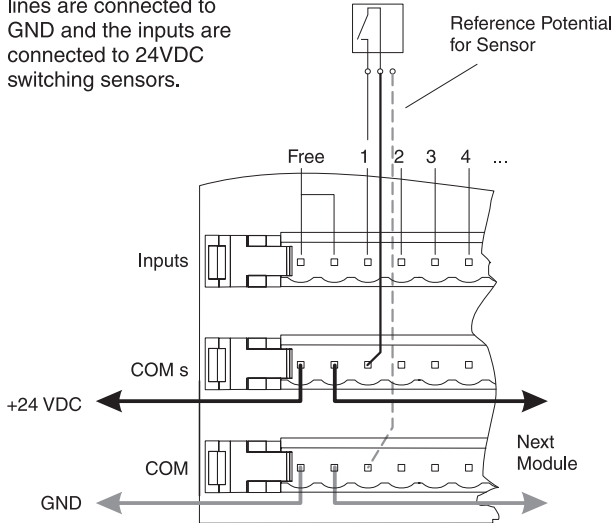
Source Connection

For source circuits, the COM lines are connected to +24 VDC and the inputs are connected to GND switching sensors

7.7.6 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Variable declaration with PCC 2003 CPU and remote slaves

Function	Variable Declaration				
	Scope	Data Type	Length	Module Type	Chan.
Single digital input (channel x)	tc_global	BIT	1	Digit. In	1 ... 8
Module status	tc_global	BYTE	1	Status In	0

Variable declaration with CAN slaves

Function	Variable Declaration				
	Scope	Data Type	Length	Module Type	Chan.
Single digital input (channel x)	tc_global	BIT	1	Digit. In	1 ... 8

Module status

The module status for CAN slaves can only be read using command codes. The command codes are explained in Chapter 5, "CAN Bus Controller Functions", section "Command Codes and Parameters". An example is provided in Chapter 4 "Module Addressing".

7.7.7 Access Using CAN IDs

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

CAN ID Packed

A maximum of eight digital I/O modules can be run in packed mode.

CAN ID [1]	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
286	Module 1	Module 2	Module 3	Module 4	Module 5	Module 6	Module 7	Module 8

[1] CAN ID = $286 + (nd - 1) \times 4$ nd .... Node number of the CAN slave = 1

CAN ID Unpacked

A maximum of four digital I/O modules can be run in unpacked mode.

Module	CAN ID [1]	Byte
1	286	Inputs 1 - 8
2	287	Inputs 1 - 8
3	288	Inputs 1 - 8
4	289	Inputs 1 - 8

[1] CAN ID = $286 + (nd - 1) \times 4 + (ma - 1)$ nd ... Node number of the CAN slave = 1
     ma ... Module address of digital I/O modules = 1 - 4

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

7.7.8 Module Status

Evaluation of the module status is explained using an example in Chapter 4 "Module Addressing".

	Bit	Description
	7	0 ....Module voltage not present or too low 1 ....Module voltage OK
	6	Digital module = 0
	5	x ....Not defined, masked out
	0 - 4	Module code = $01

```
0  x  0  0  0  0  1
7                    0
```

7.8 DI439.7

7.8.1 Technical Data
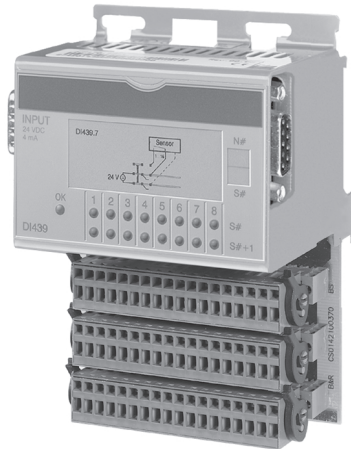
Terminal block is not included in the delivery.

Module ID	DI439.7
General Information	
Model Number	7DI439.7
Short Description	2003 digital input module, 16 inputs 24 VDC, 1 ms, sink/source, Order terminal blocks separately!
C-UL-US Listed	in preparation
B&R ID Code	$ED
Amount [1]	
CP430, EX270	2
CP470, CP770 EX470, EX770 EX477, EX777	4
CP474, CP774	6
CP476	8
Static Characteristics	
Module Type	B&R 2003 I/O Module
Number of Inputs	16
Wiring	Sink or source
Input Voltage Minimum Nominal Maximum	18 VDC 24 VDC 30 VDC
Switching Threshold LOW HIGH	<5 V >15 V

Module ID	DI439.7
Input Delay	Max. 1 ms (at 18 - 30 V)
Input Current at Nominal Voltage	Approx. 4 mA (sink/source)
Voltage Monitoring (LED: U-OK)	Yes Supply voltage >18 V
Power Consumption	Max. 0.4 W
Operating Characteristics	
Electrical Isolation	Input - PCC
Mechanical Characteristics	
Dimensions	B&R 2003 single width

[1] Two logical module slots are required by the module.

7.8.2 General Information

The digital I/O modules are all 8 channel modules. The 16 channel module DI439 operates like two 8 channel modules next to each other. Each DI439 therefore reduces the number of digital I/O modules needed by one.

Module Address

Each DI439 needs two module addresses that come immediately after each other.

Inputs	Module Address
1 - 8	Module address
9 - 16	Module address + 1

Examples

Module Type	DI435	DI435	DI435	DI435	DI439		DI439	
Module Addr.	1	2	3	4	5	6	7	8
Module No.	Module 1	Module 2	Module 3	Module 4	Module 5 I 1 - 8	Module 5 I 9 -16	Module 6 I 1 - 8	Module 6 I 9 -16

Module Type	DM435	DM435	DI439		DI439		DO435	DO435
Module Addr.	1	2	3	4	5	6	7	8
Module No.	Module 1	Module 2	Module 3 I 1 - 8	Module 3 I 9 -16	Module 4 I 1 - 8	Module 4 I 9 -16	Module 5	Module 6

7.8.3 Status LEDs

The green Status LEDs arranged in two rows show the logical state of the respective input.
LEDs marked with S# correspond to the inputs of group 1 (Y1). LEDs marked with S#+1 correspond to the inputs of group 2 (Y2).

The LED OK (orange) indicates that the input supply voltage is present. The LED is lit for input voltages from 15 to 18 VDC.

7.8.4 Input Circuit Diagram
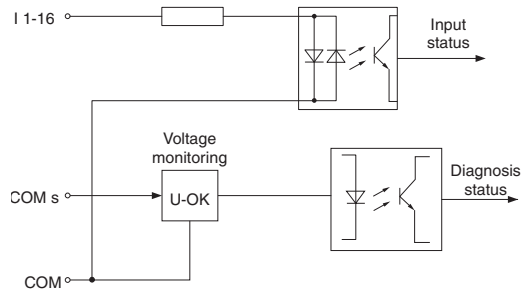
7.8.5 Legend Sheets

A legend sheet can be slid into the front of the module from above. The module circuit is shown on the back. The inputs can be labelled on the front.
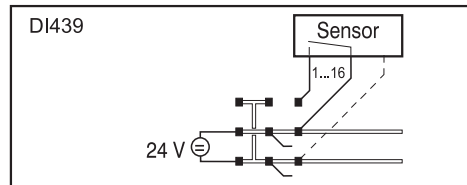
7.8.6 Sink/Source Wiring

The input module DI439 can be connected as either a sink or source circuit.

Sink Connection

For sink connections, COM is connected to signal ground, the inputs are connected to sensors that switch to 24 VDC.

Reference potential for sensor

COM 1 2 3 4 5 6 7 ...

Inputs X1

COM s X2

+24 VDC

COM X3

GND

Next Module
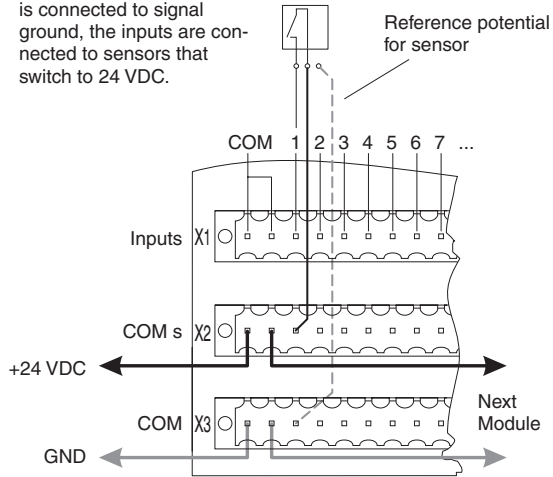
Source Connection

For source connections, COM is connected with +24 VDC, the inputs are connected to sensors that switch to signal ground.

Reference potential for sensor

COM 1 2 3 4 5 6 7 ...

Inputs X1

COM s X2

GND

COM X3

+24 VDC

Next Module

7.8.7 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Variable declaration with PCC 2003 CPU and remote slaves

Designation	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Digital inputs 1 - 8	BIT	Digit. In	1 ... 8	●		Level of digital inputs 1 – 8
Module status	BYTE	Status In	0	●		Module status / diagnose function
Digital inputs 9 -16 (module address + 1)	BIT	Digit. In	1 ... 8	●		Level of digital inputs 9 –16

Variable declaration with CAN slaves

Designation	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Digital inputs 1 – 8	BIT	Digit. In	1 ... 8	●		Level of digital inputs 1 - 8
Digital inputs 9 –16 (module address + 1)	BIT	Digit. In	1 ... 8	●		Level of digital inputs 9 -16

Module status

The module status for CAN slaves can only be read using command codes. The command codes are explained in Chapter 5, "CAN Bus Controller Functions", section "Command Codes and Parameters".
An example is provided in Chapter 4 "Module Addressing".

7.8.8 Access Using CAN IDs

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

CAN ID Packed

A maximum of eight digital I/O modules can be operated in packed mode.
The 16 channel module DI439 operates like two 8 channel modules next to each other. If two DI439 modules are used, only six additional digital I/O modules can be used.
The following example shows the structure of the CAN object if four DI435 and two DI 439 modules are used.

CAN ID [1]	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
286	DI435	DI435	DI435	DI435	DI439 I 1 - 8	DI439 I 9 - 16	DI439 I 1 - 8	DI439 I 9 - 16

[1] CAN ID = 286 + (nd - 1) x 4   nd .... Node number of the CAN slave = 1

CAN ID Unpacked

A maximum of four digital I/O modules can be run in unpacked mode.
The following example shows the structure of the CAN object if two DI435 and one DI439 modules are used.

Module	CAN ID [1]	Byte
DI435	286	Inputs 1 - 8
DI435	287	Inputs 1 - 8
DI439	288	Inputs 1 - 8
	289	Inputs 9 - 16

[1] CAN ID = 286 + (nd - 1) x 4 + (ma - 1)   nd ... Node number of the CAN slave = 1
                                                    ma ... Module address of digital I/O modules = 1 - 4

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

7.8.9 Module Status

Evaluation of the module status is explained using an example in Chapter 4 "Module Addressing".

	Bit	Description
	7	0 ....Module voltage not present or too low 1 ....Module voltage OK
	6	Digital module = 0
	5	x ....Not defined, masked out
	0 - 4	Module code = $0D

0	x	0	1	1	0	1

7   0

7.9 DI439.72

7.9.1 Technical Data
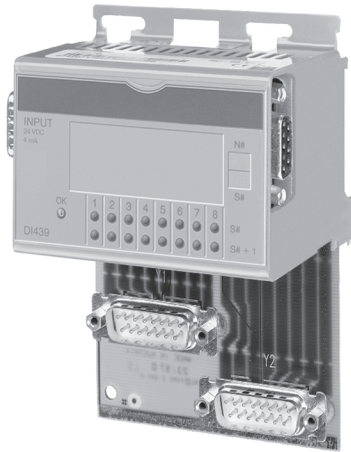
Module ID	DI439.72
General Information	
Model Number	7DI439.72
Short Description	2003 digital input module, 16 inputs 24 VDC, 1 ms, sink/source, 2 electrically isolated input groups
C-UL-US Listed	in preparation
B&R ID Code	$E9
Amount [1]	
CP430, EX270	2
CP470, CP770 EX470, EX770 EX477, EX777	4
CP474, CP774	6
CP476	8
Static Characteristics	
Module Type	B&R 2003 I/O Module
Number of Inputs Total in 2 Groups of	16 8
Wiring	Sink or source
Input Voltage Minimum Nominal Maximum	18 VDC 24 VDC 30 VDC

Module ID	DI439.72
Switching Threshold 　LOW 　HIGH	<5 V >15 V
Input Delay	Max. 1 ms (at 18 - 30 V)
Input Current at Nominal Voltage	Approx. 4 mA (sink/source)
Voltage Monitoring (LED: U-OK)	Yes Supply voltage >18 V
Power Consumption	Max. 0.4 W
Operating Characteristics	
Electrical Isolation 　Input - PCC 　Group - Group	Yes Yes
Mechanical Characteristics	
Dimensions	B&R 2003 single width

[1] Two logical module slots are required by the module.

7.9.2 General Information

The digital I/O modules are all 8 channel modules. The 16 channel module DI439 operates like two 8 channel modules next to each other. Each DI439 therefore reduces the number of digital I/O modules needed by one.

Module Address

Each DI439 needs two module addresses that come immediately after each other.

Inputs	Module Address
1 - 8	Module address
9 - 16	Module address + 1

Examples

Module Type	DI435	DI435	DI435	DI435	DI439		DI439	
Module Addr.	1	2	3	4	5	6	7	8
Module No.	Module 1	Module 2	Module 3	Module 4	Module 5 I 1 - 8	Module 5 I 9 -16	Module 6 I 1 - 8	Module 6 I 9 -16

Module Type	DM435	DM435	DI439		DI439		DO435	DO435
Module Addr.	1	2	3	4	5	6	7	8
Module No.	Module 1	Module 2	Module 3 I 1 - 8	Module 3 I 9 -16	Module 4 I 1 - 8	Module 4 I 9 -16	Module 5	Module 6

7.9.3 Status LEDs

The green Status LEDs arranged in two rows show the logical state of the respective input.
LEDs marked with S# correspond to the inputs of group 1 (Y1). LEDs marked with S#+1 correspond to the inputs of group 2 (Y2).

The LED OK (orange) indicates that the input supply voltage is present. The LED is lit for input voltages from 15 to 18 VDC.

7.9.4 Input Circuit Diagram

Group 1 - Y1

Group 2 - Y2
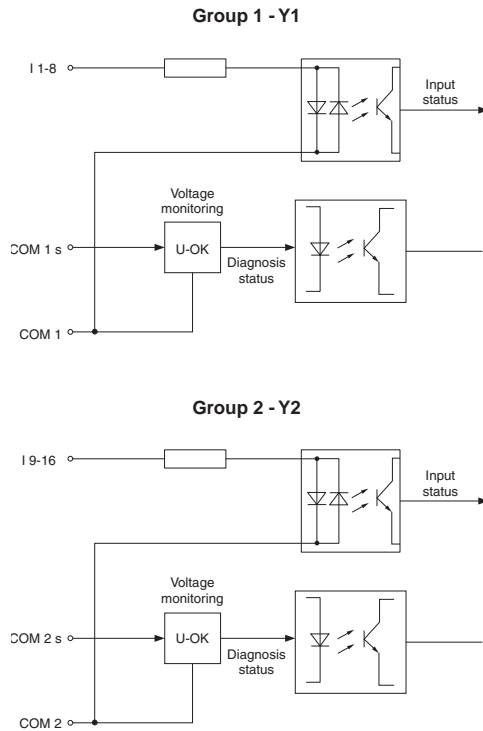
7.9.5 Legend Sheets

A legend sheet can be slid into the front of the module from above. The module circuit is shown on the back. The inputs can be labelled on the front.
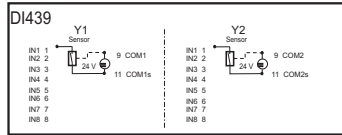
7.9.6 Pin Assignment Group 1 (Y1)

15 Pin D-Type Connector

Y1
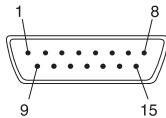
Pin	Group 1 – Plug Y1
1	Input 1
2	Input 2
3	Input 3
4	Input 4
5	Input 5
6	Input 6
7	Input 7
8	Input 8
9	COM 1 Reference potential Y1
10	free
11	COM 1 s Input supply Y1
12	free
13	free
14	free
15	free

7.9.7 Pin Assignment Group 2 (Y2)
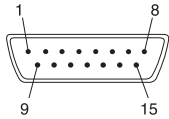
Pin	Group 2 – Plug Y2	
1	Input 9	
2	Input 10	
3	Input 11	
4	Input 12	
5	Input 13	
6	Input 14	
7	Input 15	
8	Input 16	
9	COM 2	Reference potential Y2
10	free	
11	COM 2 s	Input supply Y2
12	free	
13	free	
14	free	
15	free	

15 Pin D-Type Connector

Y2

7.9.8 Sink/Source Wiring

The input module DI439 can be connected as either a sink or source circuit.

Sink Connection

For sink wiring (current consumer from the sensor's point of view), the COM connection for an input group is connected to GND and the inputs are connected to 24VDC switching sensors.
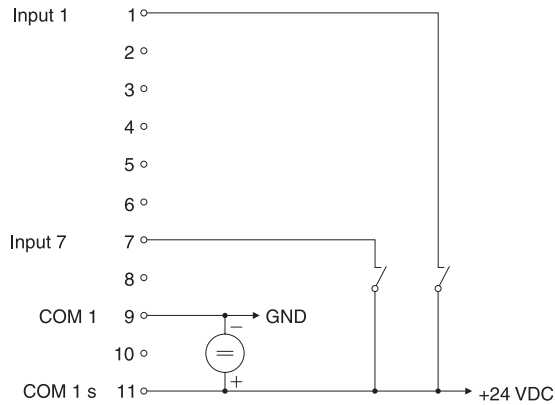
Source Connection

For source wiring (current supplier from the sensor's point of view), the COM connection of an input group is connected to +24VDC, and the inputs are connected to GND switching senors.

7.9.9 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Variable declaration with PCC 2003 CPU and remote slaves

Designation	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Digital inputs 1 - 8	BIT	Digit. In	1 ... 8	●		Level of digital inputs 1 - 8
Module status	BYTE	Status In	0	●		Module status / diagnose function
Digital inputs 9 -16 (module address + 1)	BIT	Digit. In	1 ... 8	●		Level of digital inputs 9 -16

Variable declaration with CAN slaves

Designation	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Digital inputs 1 - 8	BIT	Digit. In	1 ... 8	●		Level of digital inputs 1 – 8
Digital inputs 9 -16 (module address + 1)	BIT	Digit. In	1 ... 8	●		Level of digital inputs 9 -16

Module status

The module status for CAN slaves can only be read using command codes. The command codes are explained in Chapter 5, "CAN Bus Controller Functions", section "Command Codes and Parameters".
An example is provided in Chapter 4 "Module Addressing".

7.9.10 Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

CAN ID Packed

A maximum of eight digital I/O modules can be operated in packed mode.
The 16 channel module DI439 operates like two 8 channel modules next to each other. If two DI439 modules are used, only six additional digital I/O modules can be used.
The following example shows the structure of the CAN object if four DI435 and two DI 439 modules are used.

CAN ID [1]	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
286	DI435	DI435	DI435	DI435	DI439 I 1 - 8	DI439 I 9 - 16	DI439 I 1 - 8	DI439 I 9 - 16

[1] CAN ID = 286 + (nd - 1) x 4   nd .... Node number of the CAN slave = 1

CAN ID Unpacked

A maximum of four digital I/O module can be run in unpacked mode.
The following example shows the structure of the CAN object if two DI435 and one DI439 modules are used.

Module	CAN ID [1]	Byte
DI435	286	Inputs 1 - 8
DI435	287	Inputs 1 - 8
DI439	288	Inputs 1 - 8
	289	Inputs 9 - 16

[1] CAN ID = 286 + (nd - 1) x 4 + (ma - 1)   nd ... Node number of the CAN slave = 1

ma ... Module address of digital I/O modules = 1 - 4

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

7.9.11 Module Status

Evaluation of the module status is explained using an example in Chapter 4 "Module Addressing".

	Bit	Description
	7	0 ....Module voltage not present or too low 1 ....Module voltage OK
	6	Digital module = 0
	5	x ....Not defined, masked out
	0 - 4	Module code = $09

0	x	0	1	0	0	1

7   0

7.10 DI645

7.10.1 Technical Data
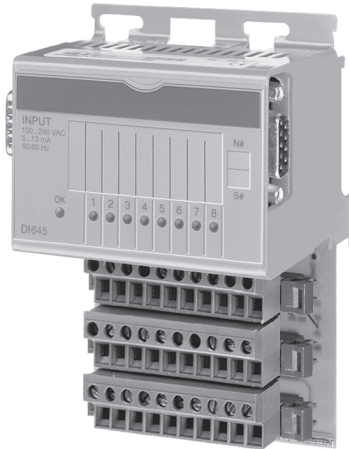
Terminal block is not included in the delivery.

Module ID	DI645
General Information	
Model Number	7DI645.7
Short Description	2003 digital input module, 8 inputs 100-240 VAC, 50 ms, Order terminal blocks separately!
C-UL-US Listed	Yes
B&R ID Code	$E5
Amount	
CP430, EX270	4
CP470, CP770 CP474, CP476, CP774 EX470, EX770 EX477, EX777	8
Static Characteristics	
Module Type	B&R 2003 I/O Module
Number of Inputs	8
Wiring	Sink or source
Input Voltage Minimum Nominal Maximum	85 VAC 100 - 240 VAC 264 VAC
Input Voltage Frequency	47 - 63 Hz
Switching Threshold LOW HIGH	<40 VAC at 2 mA >79 VAC
Input Delay	Max. 50 ms (at 85 - 264 VAC)

Module ID	DI645

Input Current
    100 VAC / 60 Hz
    240 VAC / 50 Hz
	Approx. 5 mA
	Approx. 11 mA

Voltage Monitoring (LED: U-OK)	Yes Supply voltage >85 VAC
Power Consumption	Max. 0.2 W
Operating Characteristics	
Electrical Isolation	Input - PCC
Mechanical Characteristics	
Dimensions	B&R 2003 single width

7.10.2 Status LEDs

The status-LEDs 1 - 8 (green) show the logical state of the corresponding inputs.

The LED OK (orange) indicates that the input supply voltage is present. The LED is lit for input voltages from 60 to 85 VAC.

7.10.3 Input Circuit Diagram
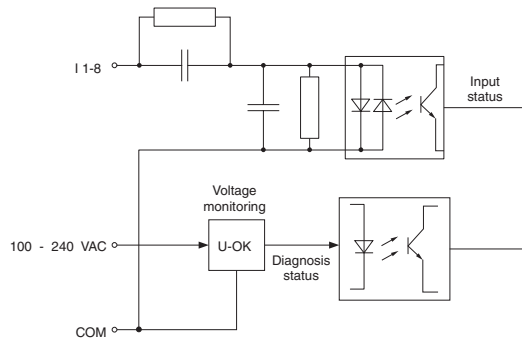
7.10.4 Legend Sheets

A legend sheet can be slid into the front of the module from above. The module circuit is shown on the back. The inputs can be labelled on the front.
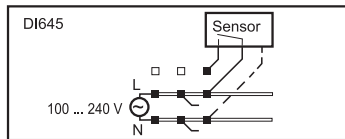
7.10.5 Connections

7.10.6 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Variable declaration with PCC 2003 CPU and remote slaves

Function	Variable Declaration				
	Scope	Data Type	Length	Module Type	Chan.
Single digital input (channel x)	tc_global	BIT	1	Digit. In	1 ... 8
Module status	tc_global	BYTE	1	Status In	0

Variable declaration with CAN slaves

Function	Variable Declaration				
	Scope	Data Type	Length	Module Type	Chan.
Single digital input (channel x)	tc_global	BIT	1	Digit. In	1 ... 8

<u>Module status</u>

The module status for CAN slaves can only be read using command codes. The command codes are explained in Chapter 5, "CAN Bus Controller Functions", section "Command Codes and Parameters".
An example is provided in Chapter 4 "Module Addressing".

7.10.7 Access Using CAN IDs

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

CAN ID Packed

A maximum of eight digital I/O modules can be operated in packed mode.

CAN ID [1]	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
286	Module 1	Module 2	Module 3	Module 4	Module 5	Module 6	Module 7	Module 8

[1] CAN ID = 286 + (nd - 1) x 4   nd .... Node number of the CAN slave = 1

CAN ID Unpacked

A maximum of four digital I/O modules can be run in unpacked mode.

Module	CAN ID [1]	Byte
1	286	Inputs 1 - 8
2	287	Inputs 1 - 8
3	288	Inputs 1 - 8
4	289	Inputs 1 - 8

[1] CAN ID = 286 + (nd - 1) x 4 + (ma - 1)   nd ... Node number of the CAN slave = 1
ma ... Module address of digital I/O modules = 1 - 4

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

7.10.8 Module Status

Evaluation of the module status is explained using an example in Chapter 4 "Module Addressing".

								Bit	Description
								7	0 ....Module voltage not present or too low 1 ....Module voltage OK
								6	Digital module = 0
								5	x ....Not defined, masked out
								0 - 4	Module code = $05
	0	x	0	0	1	0	1		

7   0

8 DIGITAL OUTPUT MODULES

8.1 GENERAL INFORMATION

Digital output modules are used to control external loads (relays, motors, solenoids). The states of the digital outputs are indicated by status LEDs. The relevant differences in characteristics for output modules are:

- Number of Outputs
- Type (relay, transistor)
- Switching Voltage
- Continuous Current

8.2 PROTECTIVE CIRCUIT

The transistor output module DO435 has overload protection and an internal protective circuit for overload peaks or reverse polarity. The negative anode potential allows fast switching of inductive loads without external diodes.
The relay output module DO720 requires external overload protection (fuse).
On the relay output module DO721, each output is protected against overload by a fuse.

8.3 SCREW-IN MODULE OVERVIEW

Module	DO135	DO164
Number of Outputs	4	4
Design FET	Triac Coupler	
Type	Push/Pull switch	—
Switching Voltage Minimum Nominal Maximum	11.4 VDC 11 - 24 VDC 30 VDC	48-125 VAC
Continuous Current per Output Max. 0.1 A Module	Max. 0.05 A Max. 0.4 A	Max. 0.2 A
Starting Pulse Current	—	Max. 0.5 A
Switching Frequency	Max. 100 kHz	96-126 Hz

8.4 I/O MODULE OVERVIEW

Module	DO435	DO720	DO721	DO722
Number of Outputs	Max. 8	8	4	8
Design	FET	Relay	Relay	Relay
Type		Normally open	Change-over contact	Normally open

Switching Voltage
   Minimum
   Nominal
   Maximum
	18 VDC
	24 VDC
	30 VDC
		
		240 VAC / 30 VDC
		264 VAC / 110 VDC
			
			240 VAC / 24 VDC
			264 VAC / 125 VDC
				
				240 VAC / 24 VDC
				264 VAC / 125 VDC

Continuous Current per
   Output
   Module
	
	Max. 2 A
	Max. 8 A
		
		Max. 2 A
		Max. 12 A
			
			Max. 4 A
			Max. 16 A
				
				Max. 2.5 A
				Max. 20 A

8.5 PROGRAMMING

The digital outputs are accessed directly in the application program using a variable name. The relationship between the output channel for a certain module and the variable name is defined in the variable declaration. The declaration is made identically for each programming language using a table editor.

8.6 DO135

8.6.1 Technical Data

Module ID	DO135
General Information	
Model Number	7DO135.70
Short Description	2003 digital output module, 4 FET outputs 12 to 24 VDC, 0.1 A, screw-in module, Order terminal block TB712 separately!
C-UL-US Listed	Yes
B&R ID Code	$14
Slot	AF101 adapter module, CP interface
Static Characteristics	
Module Type	B&R 2003 screw-in module
Number of Outputs	4
Design	FET
Type	Push/Pull switch

Switching Voltage/Supply
   Minimum
   Nominal
   Maximum
   Protection
	
	11.4 VDC
	12 - 24 VDC
	30 VDC
	Reverse polarity protection

Continuous Current per
   Output
   Module
	
	Max. 0.1 A
	Max. 0.4 A

Maximum Switching Frequency	100 kHz
Residual Voltage	Max. 0.6 V at 0,1 A
Capacitive Load	Max. 20 nF An overcurrent warning is given when switching large capacitive loads
Power Consumption	Max. 0.2 W

Module ID	DO135
Protection Characteristics	

Protection
   Short circuit protection
   Overload protection
	
	Yes
	Yes

Short Circuit Current	0.11 - 0.3 A

Diagnosis Status for SW Evaluation
   Overvoltage
   Undervoltage
   Overcurrent Monitoring
	
	Us >30 VDC, a voltage >35 VDC for t >5 ms will damage the outputs
	Us <10.5 VDC
	Functions with a duty cycle of at least 10 μs

Dynamic Characteristics	

Switching Delay
   Typical
   Maximum
	
	<2 μs
	2.4μs

Operating Characteristics	

Electrical Isolation
   Output - PCC
   Output - Output
	
	No
	No

Cable Requirement	Shielded cable
Mechanical Properties	
Dimensions	B&R 2003 screw-in module

8.6.2 General Information

The DO135 is a 4 channel output module. The outputs are tristate when switched on. All outputs are activated as a group after the outputs are configured using configuration word 14.

8.6.3 Operating Modes

Operating modes can be set separately for each output. The following operating modes are available:

- Normal operation
- Pulse width modulation (PWM)
- TPU operation

Normal Operation

The outputs are switched on/off.

Pulse Width Modulation

Output are switched on/off periodically. Pulse width ratio, period duration and resolution can be set.
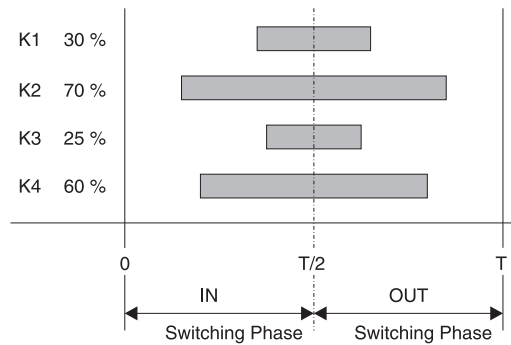
TPU Operation

In TPU operation, the outputs are controlled by the TPU. If e.g. the DO135 module is inserted in the first slot of the CP interface, the first output can be operated using the LTX function LTXdo0().

8.6.4 Special Functions

- The supply voltage is tested over a valid range. (10.5 VDC < Us < 30 VDC)
- The channels are equipped with readable power cut-off

8.6.5 Output Circuit Diagram
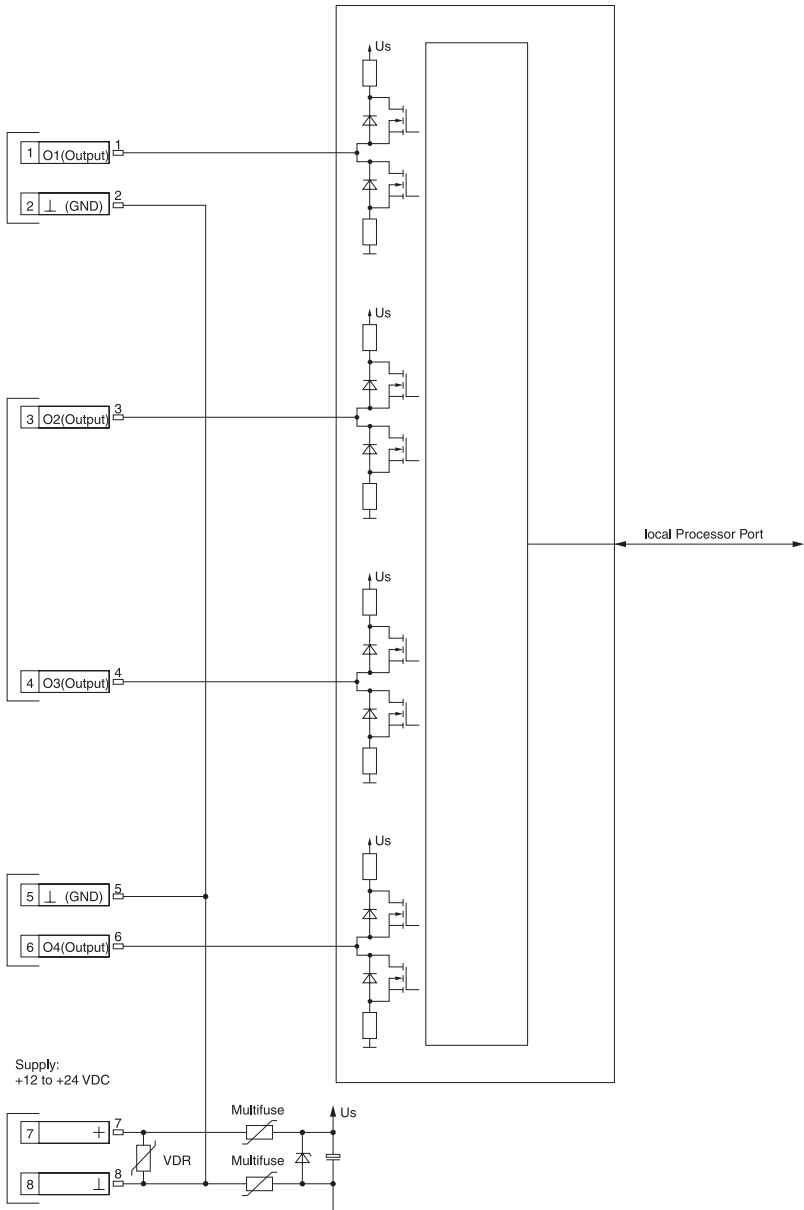
8.6.6 Connections
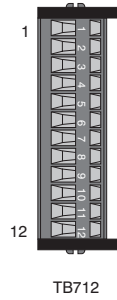
Pin	Assignment
1	Shield
2	Output 1
3	GND
4	Output 2
5	Shield
6	Shield
7	Output 3
8	GND
9	Output 4
10	Shield
11	+12 to +24 VDC
12	GND

1

12

TB712

8.6.7 Connection Example

Load

1 A Fuse
(slow-blow)

+

−

Shield
Output 1
GND
Output 2
Shield
Shield
Output 3
GND
Output 4
Shield
+12 to +24 VDC
GND
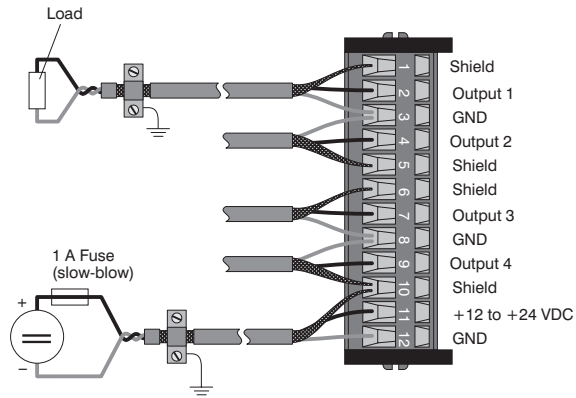
Supply:

12 to 24 VDC
Protect with 1 A slow-blow fuse!

8.6.8 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for this module.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	WORD	Analog In	1	●		Switching phase counter for all channels (pulse width modulation)
	INT16	Analog Out	1		●	Output state or pulse width ratio for output 1
Data word 1	INT16	Analog Out	2		●	Output state or pulse width ratio for output 2
Data word 2	INT16	Analog Out	3		●	Output state or pulse width ratio for output 3
Data word 3	INT16	Analog Out	4		●	Output state or pulse width ratio for output 4
Configuration word 8	WORD	Transp. Out	16		●	Period time
Configuration word 9	WORD	Transp. Out	18		●	Factor for period time
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

8.6.9 Access Using CAN IDs

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

Input Data (switching phase counter)

The switching phase counter can be transferred packed or unpacked in pulse width modulation operating mode.
CAN objects can only be sent back in packed mode.

CAN ID [1]	Slot 1		Slot 2		Slot 3		Slot 4	
542	ScrM 1L	ScrM 1H	ScrM 2L	ScrM 2H	ScrM 3L	ScrM 3H	ScrM 4L	ScrM 4H
543	free							
544	free							
545	free							

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4

nd ..... Node number of the CAN slave = 1

ma .... Module address of the AF101 = 1

Four CAN objects can be sent back in unpacked mode.

Slot	CAN ID [1]	Word 1		Word 2	Word 3	Word 4
1	542	ScrM 1L	ScrM 1H	Not used (2 byte objects)		
2	543	ScrM 2L	ScrM 2H	Not used (2 byte objects)		
3	544	ScrM 3L	ScrM 3H	Not used (2 byte objects)		
4	545	ScrM 4L	ScrM 4H	Not used (2 byte objects)		

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address of the AF101 = 1

sl .... Slot number of the screw-in module on the AF101 (1 - 4)

Output Data

It is not possible to pack output data with the DO135. Therefore one CAN object is transferred per screw-in module.
If an adapter module AF101 is equipped with a four DO135 modules, the CAN object has the following structure:

Slot	CAN ID [1]	Word 1		Word 2		Word 3		Word 4	
1	1054	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
2	1055	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
3	1056	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
4	1057	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H

[1] CAN ID = 1054 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

   nd ..... Node number of the CAN slave = 1

   ma .... Module address of the AF101 = 1

   sl .... Slot number of the screw-in module on the AF101 (1 - 4)

> ⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

8.6.10 Description of Data and Configuration Words

Data Word 0 (read)

The switching phase of all channels in the pulse width modulation mode are counted with a 16-bit switching phase counter. The counter value can be read from data word 0.
It operates like a free running counter. This means that after reaching its highest value of 65535, the counter resets and begins again from 0.

The switching phase the output is currently in can be determined using the counter value.

Even Counter Value: Switch-on phase
Odd Counter Value: Switch-off phase

PWM = 50 %
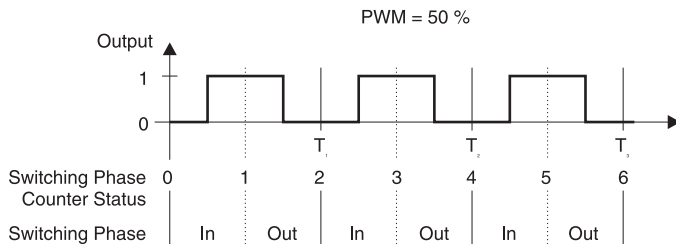
Switching Phase

In the switch-on phase (counter value is even), the output changes from OFF to ON. In the switch-off phase (counter value is odd), the output changes from ON to OFF.
The switching time is determined by the pulse width ratio.

Data words 0, 1, 2, 3 (write)

The output status and the pulse width ratio are defined with these data words according to the operating mode set.

Normal Operation

Bit 0 corresponds to the output state.

```
Output 1 = 0 ............. Data word 0 = $0000
Output 1 = 1 ............. Data word 0 = $0001
        :                          :
Output 4 = 0 ............. Data word 3 = $0000
Output 4 = 1 ............. Data word 3 = $0001
```

Pulse Width Modulation

This operating mode defines the pulse width ratio.

```
32767 ($7FFF) ....... 100 %
    0 ($0000) ........... 0 %
    Neg. Values ........... 0 %
```

Example

Outputs 3 and 4 use operating mode pulse width modulation. The following pulse width ratios are defined:

Output 3 .... 25 %   =>   25 % of 32767 = 8192 ($2000) in data word 2

Output 4 .... 75 %   =>   75 % of 32767 = 24575 ($5FFF) in data word 3
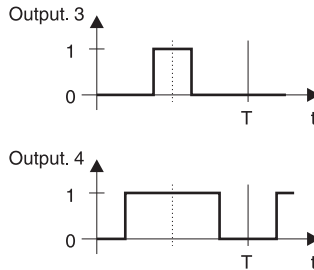
Configuration Word 8 (write)

The period time is defined with configuration word 8. The period time depends on the factor used (see configuration word 9).

The standard value for the factor is 4. This results in the definition of the period time in milliseconds (0 - 65535).

If the period time is set to 0, the outputs keep their value after the current period has passed (normally log 0, only log 1 with a pulse width ratio of 100%). The period counter is then cleared and remains 0.

If the period time is changed to a value between 1 and 65535, the period counter will be stopped, cleared and restarted with the new value after the current period of the period counter.

Period Time Unit

The period time unit is calculated with the following formula:

Period time unit = 250 μs * factor

Example

A factor of 4 gives the following period time unit:

Period time unit= 250 μs * 4 = 1 ms

Configuration Word 9 (write)

The factor for the period time is defined with configuration word 9 (see configuration word 8).

The standard value for the factor is 4. This results in the definition of the period time in milliseconds.

The range of values lays between 1 and 256.
Larger values are limited to 256.

0 will be set to 256!

Pulse Width Modulation Resolution

Pulse width modulation resolution is calculated using the following formula:

PWM resolution = 125 μs * factor

The PWM resolution corresponds to the smallest pulse time.

Example

A typical module frequency range between 0.1Hz and 10 Hz results in the following resolution with a factor of 4:

Frequency	Period Time	Resolution	Smallest Pulse Width Ratio
0.1 Hz	10 s	20000 Ticks	0005 %
10 Hz	0.1 s	200 Ticks	0.5 %

To double the resolution with the same effective value, the factor can be divided in half.

Configuration word 12 (read)

Configuration word 12 contains the module status.

Bit	Description
8 - 15	x .... Not defined, masked out
7	0 .... Supply voltage ≤30 V 1 .... Supply voltage >30 V
6	0 .... Supply voltage ≥10.5 V 1 .... Supply voltage <10.5 V or current limitation has responded
5	0 .... Half-period counter for pulse width modulation is counting 1 .... Half-period counter for pulse width modulation is not counting
4	x .... Not defined, masked out
3	Switching level output 4
2	Switching level output 3
1	Switching level output 2
0	Switching level output 1

x	x	x	x	x	x	x	x				x				

15   8 7   0

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

Bit	Description
8 - 15	Module code = $14
0 - 7	x ....Not defined, masked out

0	0	0	1	0	1	0	0	x	x	x	x	x	x	x	x

15   8 7   0

Configuration Word 14 (write)

The module is configured using configuration word 14.

Channel is not active

All outputs are tristate when they are all set to operating mode "channel not active".

TPU Operation

The output state is defined with LTX functions (e.g. LTXdo0()) for outputs with this operating mode.

	Bit	Description
	8 - 15	0
	6 - 7	Operating mode for output 4 0....Channel is not active 1....Normal operation 2....Pulse width modulation 3....TPU operation
	4 - 5	Operating mode for output 3 0....Channel is not active 1....Normal operation 2....Pulse width modulation 3....TPU operation
	2 - 3	Operating mode for output 2 0....Channel is not active 1....Normal operation 2....Pulse width modulation 3....TPU operation
	0 - 1	Operating mode for output 1 0....Channel is not active 1....Normal operation 2....Pulse width modulation 3....TPU operation

```
0 0 0 0 0 0 0 0
15          8 7          0
```

8.7 DO164

8.7.1 General Information

The screw-in module DO164 is equipped with four output channels. It is used to send firing pulses (triac coupler) for phase-angle control of power triacs.
The module is installed on the adapter module or on the CP interface.

8.7.2 Technical Data

Module ID	DO164
General Information	
Model Number	7DO164.70
Short Description	2003 digital output module, 4 FET outputs 48 to 125 VAC, 0.05 A, zero cross detection, screw-in module, Order terminal block TB712 separately!
C-UL-US Listed	in preparation
B&R ID Code	$3C
Module Type	B&R 2003 screw-in module
Slot	AF101 adapter module, CP interface
Zero Cross Detection (connection pins 11 and 12)	
Number of Inputs	1
Rated Voltage	48 to 125 VAC
Rated Frequency	48 to 63 Hz
Input Impedance in Signal Range	1 MΩ symmetric 500 kΩ to GND
Switching Threshold Low Range High Range	<-5 V >+5 V
Switching Hysteresis	0.2 V
Tolerance of the Zero Cross Signal at 48 to 125 VAC	0 to 100 μs
Electrical Isolation	No

Module ID	DO164
Triac Outputs	
Number of Outputs	4
Type of Outputs [1]	Triac coupler, only to control power triacs or nonparallel thyristors
Rated Voltage	48 to 125 VAC
Rated Frequency	48 to 63 Hz

Output Current
    Continuous Current
    Ignition Pulse Current
	Max. 50 mA
	Max. 0.5 A

Residual Voltage	Max. 2.5 V at 50 mA
Holding Current	Max. 3.5 mA
Leakage Current (0 signal)	Max. 1 μA
Critical Rate of Rise of Commutating Voltage when Switched Off	>500 V/μs
Drive Pulse Duration T_{DP} (TPU outputs)	>250 μs
Power Consumption	Max. 0.6 W
Protection Characteristics	
Protection	No short circuit protection

Type of External Protective Circuit
    RC Combination
    Gate Resistance
	Reduced dV/dt
	Increased immunity

Dynamic Characteristics	
Delay 0 to 1	Max. 200 μs
Delay 1 to 0	Max. 200 μs
Operating Characteristics	

Electrical Isolation
    Input - Output
    Output - Output
	Yes
	Yes

Recommended Cable Types	Twisted pair cabling to the terminal pairs
Line Length to Power Triac	Max. 10 m
Mechanical Characteristics	
Dimensions	B&R 2003 screw-in module

[1] Because of the very low (dV/dt)c value of the triac coupler ("Critical Rate of Rise of Commutating Voltage"), the triac output is not suitable for use as SSR relay for direct switching of loads.

8.7.3 Output Circuit Diagram

Triac coupler

P1.0, Reset

| 1 | Output 1 |
| 2 | COM1 |

O1

| 3 | Output 2 |
| 4 | COM2 |

O2

Local
Processor
Port

| 5 | Output 3 |
| 6 | COM3 |

O3

| 7 | Output 4 |
| 8 | COM4 |

O4

| 9 | |

| 10 | PE |

| 11 | N |

| 12 | L1 |

ZERO
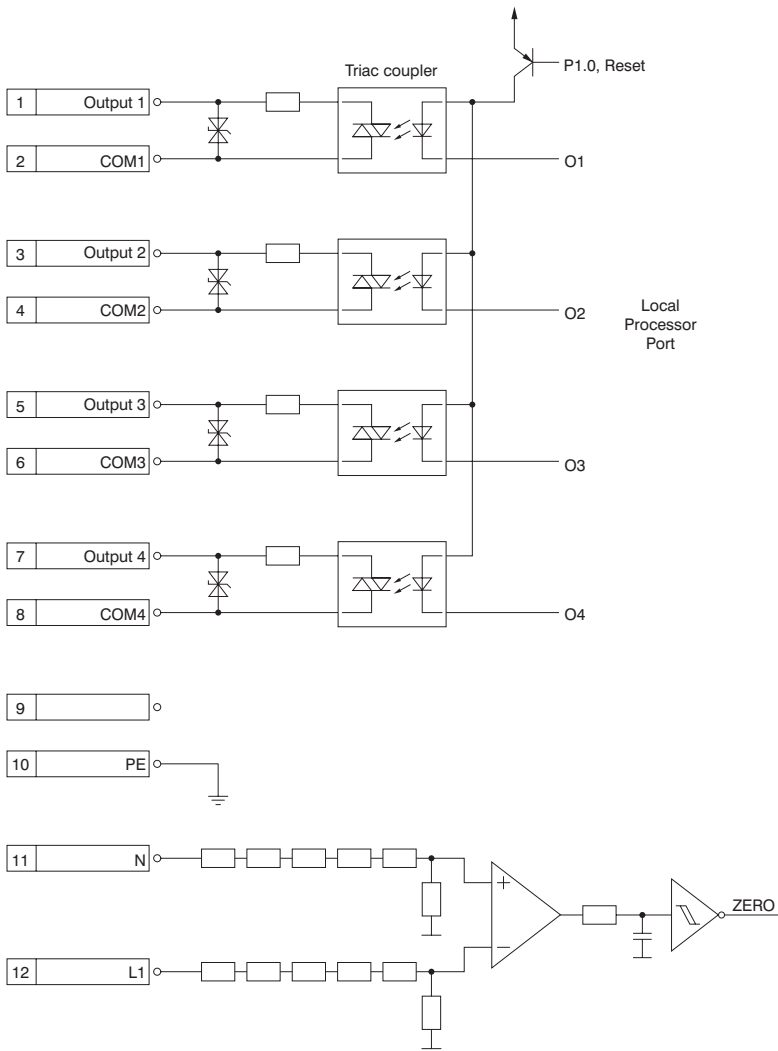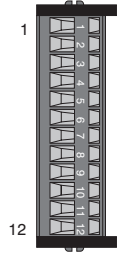
8.7.4 Connections

TB712

Pin	Assignment
1	Output 1
2	COM 1
3	Output 2
4	COM 2
5	Output 3
6	COM 3
7	Output 4
8	COM4
9	n. c.
10	PE: Ground potential
11	N: Neutral
12	L1: Phase

8.7.5 Connection Examples

Standard Connection
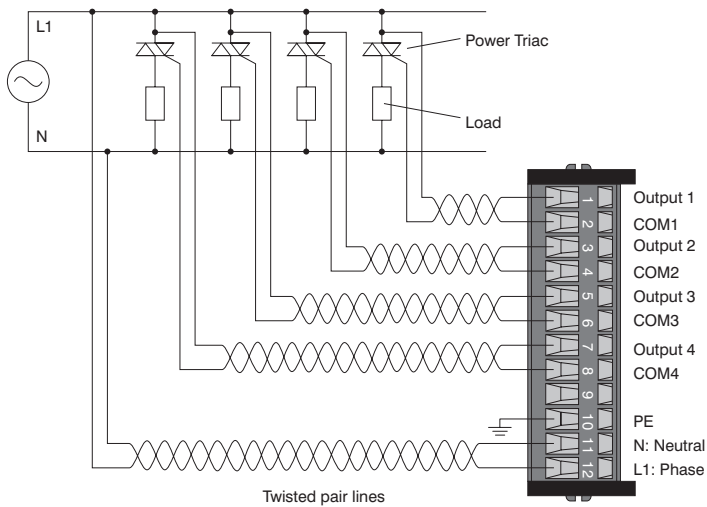
Connection with RC combination and gate resistance

Connecting the power triac with an RC combination reduces the dV/dt.
The gate resistance increases immunity.

Wiring
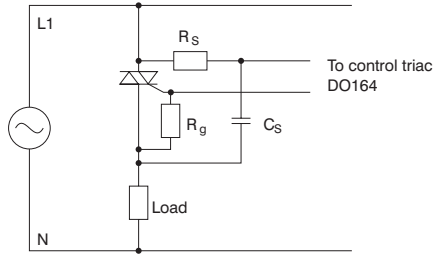
Recommended Values

RC Combination	
R_S	Approx. 22 to 100 Ω / 1 W pulse resistant
C_S	Approx. 100 nF foil capacitor

Gate Resistance	
R_g	Approx. 22 to 100 Ω

Controllable Phase Angle

Because of the connection with an RC combination, the calculation for the minimum voltage when firing is started U_{FS1} and therefore the calculation of the controllable phase angle changes.

Formula with Gate Resistance
$U_{FS1} = (I_{GT} \, [A] + V_{GT} \, [V] / R_g \, [\Omega]) * (22 \, \Omega + R_S \, [\Omega]) + V_{GT} \, [V] + 2 \, V$

Formula without Gate Resistance
$U_{FS1} = I_{GT} \, [A] * (22 \, \Omega + R_S \, [\Omega]) + V_{GT} \, [V] + 2 \, V$

8.7.6 Controllable Phase Angle

Calculating the Phase Angle

With the following formula, the earliest switch-on angle and the latest switch-off angle can be calculated. The controllable phase angle is within this range.

Note: The switch-on delay for the power triacs must be shorter than 5 μs.
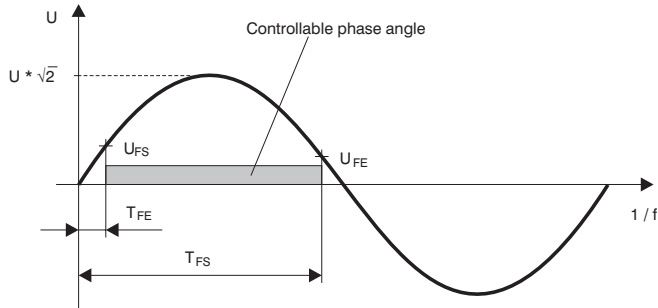
U   ... Effective value of the minimum operating voltage
U_{FS} ... Minimum voltage when firing is started
U_{FE} ... Minimum voltage when firing is ended
T_{FE} ... Earliest switch-on time
T_{FS} ... Latest switch-off time

Minimum voltage when firing is started: U_{FS1}

At the start of the firing pulse, the operating voltage must be high enough for the power triac to be fired securely.

$U_{FS1} = I_{GT}$ [A] * 22 Ω + V_{GT} [V] + 2 V

I_{GT} ...... Max. gate trigger current for the power triac
V_{GT} .... Max. gate voltage for the power triac

Minimum voltage when firing is started: U_{FS2}

When firing is started, the operating voltage must be high enough for the latch current of the power triac to be reached.

$U_{FS2} = V_T$ [V] + I_L [A] * R_L [Ω]

V_T ...... Max. voltage drop on the power triac when switched on
I_L .... Max. latch current for the power triac
R_L ...... Max. load resistance

Minimum voltage when firing is ended: U_{FE}

At the end of the firing pulse, the operating voltage must be high enough for the holding current of the power triac to be reached.

$U_{FE} = V_T$ [V] $+ I_H$ [A] $* R_L$ [Ω]

V_T ...... Max. voltage drop on the power triac when switched on

I_H .... Max. holding current for the power triac

R_L ...... Max. load resistance

Earliest switch-on angle: φ_{FE}

The high of the two voltages U_{FS1} and U_{FS2} is selected. This voltage is used to calculate the earliest switch-on angle.

$\varphi_{FE} = $ arc sin $(U_{FS}$ [V] $/ (U$ [V] $* \sqrt{2}$))

U .... Effective value of the minimum operating voltage

Latest switch-off angle: φ_{FS}

The voltage U_{FE} is used to calculate the latest switch-off angle.

$\varphi_{FS} = 180$ - arc sin $(U_{FE}$ [V] $/ (U$ [V] $* \sqrt{2}$))

U .... Effective value of the minimum operating voltage

Earliest switch-on time: T_{FE}

$T_{FE} = \varphi_{FE} / (f * 360)$

Latest switch-off time: T_{FS}

$T_{FS} = (\varphi_{FS} / (f * 360)) * 10^6$ [µs] $- (100 + 200)$ µs

f .... Frequency of the operating voltage

100 µs ...... Max. tolerance of the zero cross signal

200 µs ...... Max. switching delay for triac coupler

Earlier switch-on time with small loads

To achieve an earlier switch-on time, especially for small loads, only use the voltage U_{FS1} for the calculation. In this case however, U_{FS2} must be greater than U_{FS1} and the drive pulse must be available until the voltage U_{FS2} is reached.
This drive pulse duration is calculated from the time difference for these two voltages $T_{FS2} - T_{FS1}$. The drive pulse duration always has to be larger than the specified minimum pulse duration T_{DP}.

Calculation Example

Entry	

Operating Voltage
    Voltage
    Frequency
	48 VAC $\pm$ 15 %
	50 Hz

Load Load resistance R_L	1 kW heating element 20 Ω

Power Triac (SKKT92 SEMIKRON)
    Max. Gate Trigger Current I_{GT}
    Max. Gate Voltage V_{GT}
    Max. Latch Current I_L
    Max. Holding Current I_H
    Max. Voltage Drop V_T
    Min. Gate Trigger Current I_{GD}
	200 mA ($T_j = 0\ °C$)
	3.5 V ($T_j = 0\ °C$)
	600 mA
	250 mA
	1.6 V
	6 mA ($T_j = 125\ °C$)

RC Combination
    R_S
    C_S
	22 Ω
	100 nF

Gate Resistance	Not required because I_{GD} >> leakage current

Minimum voltage when firing is started: U_{FS1}

$U_{FS1} = I_{GT}$ [A] * (22 Ω + R_S [Ω]) + V_{GT} [V] + 2 V
$U_{FS1} = 0.2 * 44 + 3.5 + 2 = 14.3$ V

Minimum voltage when firing is started: U_{FS2}

$U_{FS2} = V_T$ [V] + I_L [A] * R_L [Ω]
$U_{FS2} = 1.6 + 0.6 * 20 = 13.6$ V

Minimum voltage when firing is ended: U_{FE}

$U_{FE} = V_T$ [V] + I_H [A] * R_L [Ω]
$U_{FE} = 1.6 + 0.25 * 20 = 6.6$ V

Earliest switch-on angle: φ_{FE}

φ_{FE} = arc sin (U_{FS} [V] / (U [V] * $\sqrt{2}$))

$U_{FS1} > U_{FS2} => U_{FS} = U_{FS1} = 14.3$ V

U ... Effective value of the minimum operating voltage

U = 48 VAC * 0.85 = 40.8 VAC

φ_{FE} = arc sin (14.3 / (40.8 * 1.41)) = 14.39°

Latest switch-off angle: φ_{FS}

φ_{FS} = 180 - arc sin (U_{FE} [V] / (U [V] * $\sqrt{2}$))

φ_{FS} = 180 - arc sin (6.6 / (40.8 * 1.41)) = 173.41°

Earliest switch-on time: T_{FE}

$T_{FE} = \varphi_{FE}$ / (f * 360)

T_{FE} = 14.39 / (50 * 360) = 799.5 μs

Latest switch-off time: T_{FS}

$T_{FS} = (\varphi_{FS}$ / (f * 360)) * 10^6 [μs] - (100 + 200) μs

T_{FS} = (173.41 / (50 * 360)) * 10^6 - 300 = 9333.5 μs

Controllable Phase Angle:

The controllable phase angle in this example is 14.39° to 173.41° or 0.8 to 9.3 ms with reference to the zero cross signal.

8.7.7 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for this module.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	WORD	Analog In	1	●		Module status
	INT16	Analog Out	1		●	Output state of output 1
Data word 1	INT16	Analog Out	2		●	Output state of output 2
Data word 2	INT16	Analog Out	3		●	Output state of output 3
Data word 3	INT16	Analog Out	4		●	Output state of output 4
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

8.7.8 Access Using CAN IDs

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

Input Data (module status)

The input data can be transferred packed or unpacked.

CAN objects can only be returned in packed mode.

CAN ID [1]	Slot 1		Slot 2		Slot 3		Slot 4	
542	ScrM 1L	ScrM 1H	ScrM 2L	ScrM 2H	ScrM 3L	ScrM 3H	ScrM 4L	ScrM 4H
543	free							
544	free							
545	free							

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4

   nd ..... Node number of the CAN slave = 1

   ma .... Module address of the AF101 = 1

Four CAN objects can be sent back in unpacked mode.

Slot	CAN ID [1]	Word 1		Word 2	Word 3	Word 4
1	542	ScrM 1L	ScrM 1H	Not used (2 byte objects)		
2	543	ScrM 2L	ScrM 2H	Not used (2 byte objects)		
3	544	ScrM 3L	ScrM 3H	Not used (2 byte objects)		
4	545	ScrM 4L	ScrM 4H	Not used (2 byte objects)		

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address of the AF101 = 1

sl .... Slot number of the screw-in module on the AF101 (1 - 4)

Output Data

It is not possible to pack output data with the DO164. Therefore one CAN object is transferred per screw-in module. If an adapter module AF101 is equipped with a four DO164 modules, the CAN object has the following structure:

Slot	CAN ID [1]	Word 1		Word 2		Word 3		Word 4	
1	1054	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
2	1055	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
3	1056	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
4	1057	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H

[1] CAN ID = 1054 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address of the AF101 = 1

sl .... Slot number of the screw-in module on the AF101 (1 - 4)

⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

8.7.9 Description of Data and Configuration Words

Data Word 0 (read)

Data word 0 contains the module status.

	Bit	Description
	6 - 15	x .... Not defined, masked out
	5	0 .... Frequency of the input voltage is in valid range during normal operation 1 .... Frequency of the input voltage exceeds the valid range of 30 to 80 Hz during normal operation (outputs are switched off)
	4	0 .... Outputs are synchronized to the frequency of the input voltage 1 .... Outputs are being synchronized to the frequency of the input voltage (outputs are switched off)
	0 - 3	x .... Not defined, masked out

x	x	x	x	x	x	x	x	x	x	x			x	x	x	x

15   8 7   0

Data words 0, 1, 2, 3 (write)

If a channel is configured as inactive, it is tristate. During normal operation, bit 0 corresponds to the output status.

Example for Output 1

    Output 1 = 0 .... Data word 0 = $0000
    Output 1 = 1 .... Data word 0 = $0001

Controllable Phase Angle

The controllable phase angle is 16.2° to 163.8°. During normal operation, the outputs are always set within this range. This results in the following voltage curve:
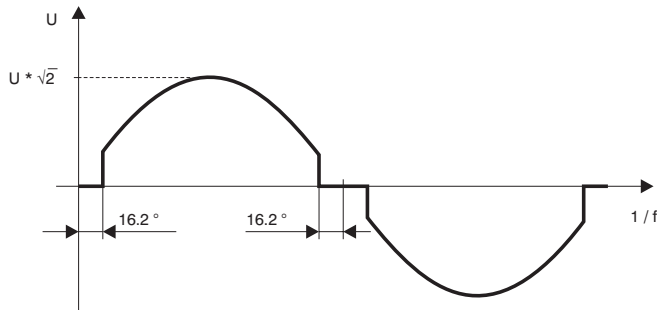
Switching Output Off

If an output is switched off during normal operation, the output remains set during the current half-wave. The output is switched off the next time the voltage crosses zero.
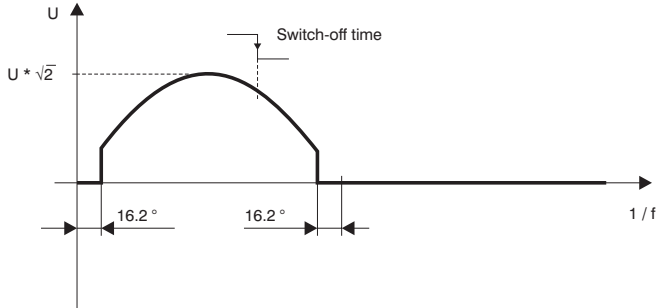
Phase-angle Control

The DO164 module must be installed on the CP interface. Software operation takes place using TPU function blocks.

Configuration word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $3C
	0 - 7	x ....Not defined, masked out

0	0	1	1	1	1	0	0	x	x	x	x	x	x	x	x

15   8 7   0

Configuration word 14 (write)

The module is configured using configuration word 14.

	Bit	Description
	9 - 15	0
	8	0....Zero cross-over of the input voltage is not evaluated 1....Zero cross-over of the input voltage is evaluated. Channel 4 is operated as an input.
	6 - 7	Definition of the operating mode for channel 4 0....Channel is not active (tristate) 1....Normal operation
	4 - 5	Definition of the operating mode for channel 3 0....Channel is not active (tristate) 1....Normal operation
	2 - 3	Definition of the operating mode for channel 2 0....Channel is not active (tristate) 1....Normal operation
	0 - 1	Definition of the operating mode for channel 1 0....Channel is not active (tristate) 1....Normal operation

0	0	0	0	0	0	0									

15   8   7   0

8.8 DO435

8.8.1 Technical Data
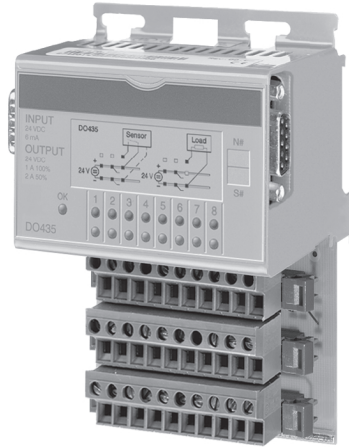
Terminal block is not included in the delivery.

Module ID	DO435
General Information	
Model Number	7DO435.7
Short Description	2003 digital output module, 8 FET outputs 24 VDC, 2 A, Outputs can be optionally used as inputs, Order terminal block TB712 separately!
C-UL-US Listed	Yes
B&R ID Code	$EB
Module Type	B&R 2003 I/O module, single width
Amount	
CP430, EX270	4

CP470, CP770
  CP474, CP476, CP774
  EX470, EX770
  EX477, EX777
	8

Voltage Monitoring (LED: U-OK)	Yes Supply voltage >18 V
Power Consumption	Max. 0.5 W
Outputs	
Number of Outputs	Max. 8
Type	FET

Switching Voltage/Supply
  Minimum
  Nominal
  Maximum
	
	18 VDC
	24 VDC
	30 VDC

Continuous Current per
  Output
  Module
	
	Max. 2 A
	Max. 8 A

Module ID	DO435

Simultaneousness
    at 1 A
    at 2 A
	100 %
	50 % (note derating curve)

Residual Voltage	Max. 1 V at 2 A

Protection
    Short circuit protection
    Overload protection
	Yes
	Yes

Short Circuit Current at 24 V	Approx. 20 A until switched off
Negative Anode Potential when Switching Off Inductive Loads	Approx. 46 V

Switching Delay
    log. 0 - log. 1
    log. 1 - log. 0
	Approx. 200 μs
	Typ. 50 μs / max. 250 μs
	Typ. 170 μs / max. 250 μs

Max. Switching Frequency
    Resistive Load
    Inductive Load
	100 Hz
	see "Switching Inductive Loads"

Electrical Isolation	Output - PCC

With this module, each channel can be used either as an input or output. All outputs must first be declared as such. Each time the module is turned on, all channels are configured as inputs.

Controller / Access	Declaration	
	Automatic	From User
CPU - PCC 2003	●	
Remote Slaves		●
CAN Slaves	●	
Access using CAN Identifier		●

8.8.2 Technical Data for the Inputs

Module ID	DO435
Number of Inputs	Max. 8
Wiring	Sink

Input Voltage
    Minimum
    Nominal
    Maximum
	18 VDC
	24 VDC
	30 VDC

Switching Threshold
    LOW
    HIGH
	<5 V
	>15 V

Input Delay	Max. 1 ms (at 18 - 30 V)
Input Current at Nominal Voltage	Approx. 6 mA
Electrical Isolation	Input - PCC

8.8.3 Status LEDs

The green/orange status LEDs 1 - 8 show the logical state of the corresponding inputs/output.

The LED U-OK (orange) indicates that the supply voltage is present. The LED is lit starting with a supply voltage of approx. 18 VDC.

8.8.4 Input/Output Diagram
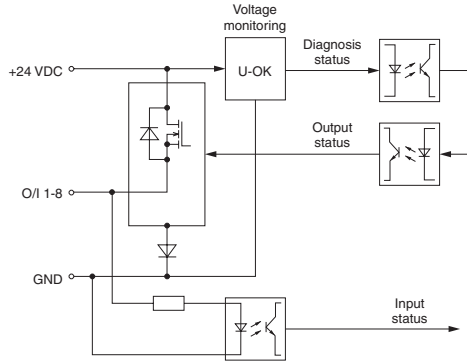
8.8.5 Legend Sheets

A legend sheet can be slid into the front of the module from above. The module circuit is shown on the back. The outputs/inputs can be labelled on the front.
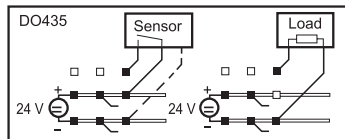
8.8.6 Connections

Output Circuit

Load

Free   1   2   3   4   ...

Outputs/Inputs

+24 VDC

GND

Next
Module

Input Connections

Reference Potential
for Sensor

Free   1   2   3   4   ...

Outputs/Inputs

+24 VDC

GND

Next
Module
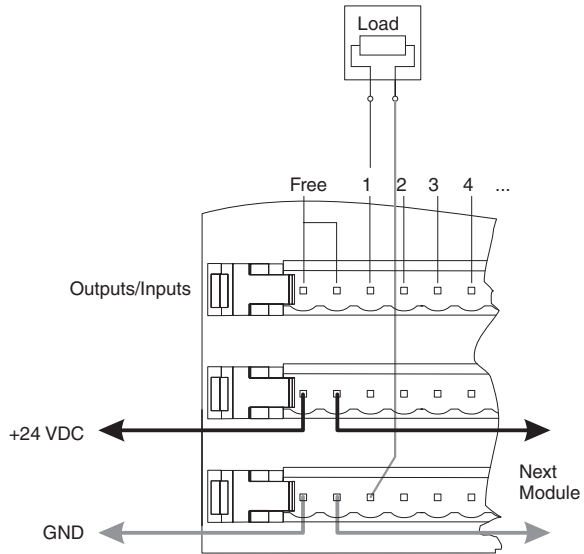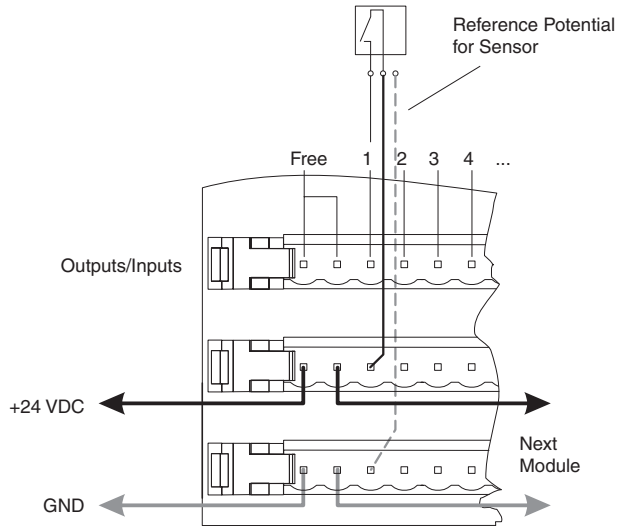
8.8.7 Switching Inductive Loads

Transistors are suitable for switching inductive loads off quickly and safely. Inverse diodes are not necessary on inductive loads. It should be noted that the maximum switching frequency at a given inductance is limited by a set negative anode potential of 46 V. The maximum switching frequency decreases as the inductance increases.

The maximum switching frequency in relation to a given inductance can be seen from the following diagram.

❶ Note Derating Curve
❷ Invalid Range (thermal cut off)
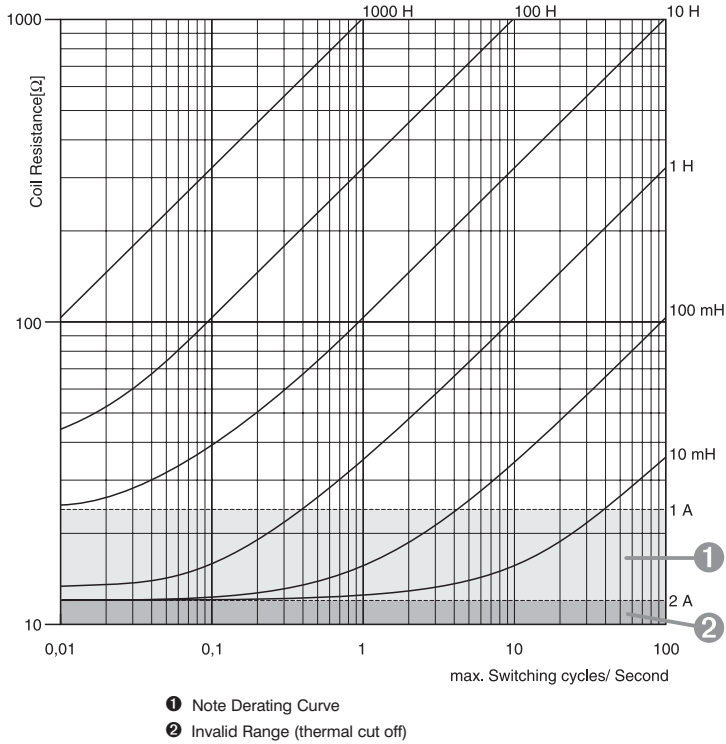
Derating Curve
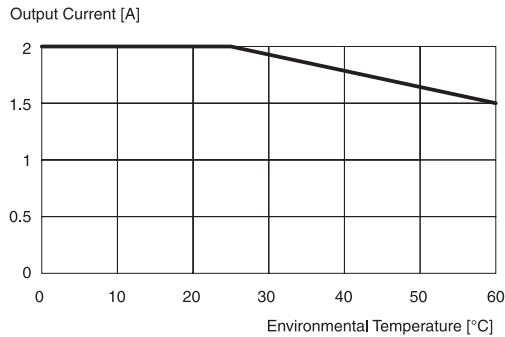
8.8.8 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

> ⚠️ **After the module is turned on, all channels are configured as inputs. Output channels must first be declared as such.**
> **Please refer to the overview in the technical data the examples in Chapter 4 "Module Addressing".**

Variable declaration with PCC 2003 CPU and remote slaves

Function	Variable Declaration				
	Scope	Data Type	Length	Module type	Chan.
Single digital input (channel x)	tc_global	BIT	1	Digit. In	1 ... 8
Single digital output (channel x)	tc_global	BIT	1	Digit. Out	1 ... 8
Module status	tc_global	BYTE	1	Status In	0

Variable declaration with CAN slaves

FunktionFunction	Variable Declaration				
	Scope	Data Type	Length	Module type	Chan.
Single digital input (channel x)	tc_global	BIT	1	Digit. In	1 ... 8
Single digital output (channel x)	tc_global	BIT	1	Digit. Out	1 ... 8

Module status

The module status for CAN slaves can only be read using command codes. The command codes are explained in Chapter 5, "CAN Bus Controller Functions", section "Command Codes and Parameters". An example is provided in Chapter 4 "Module Addressing".

8.8.9 Access Using CAN IDs

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

Digital Inputs

A maximum of eight digital I/O modules can be run in packed mode.

CAN ID [1]	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
286	Module 1	Module 2	Module 3	Module 4	Module 5	Module 6	Module 7	Module 8

[1] CAN ID = 286 + (nd - 1) x 4   nd .... Node number of the CAN slave = 1

A maximum of four digital I/O module can be run in unpacked mode.

Module	CAN ID [1]	Byte
1	286	Inputs 1 - 8
2	287	Inputs 1 - 8
3	288	Inputs 1 - 8
4	289	Inputs 1 - 8

[1] CAN ID = 286 + (nd - 1) x 4 + (ma - 1)   nd ... Node number of the CAN slave = 1
ma ... Module address of digital I/O modules = 1 - 4

Digital Outputs

All channels are declared as inputs as their default. Changing the definition to an output channel is described in Chapter 5 "CAN Bus Controller Functions". A example of this is given in Chapter 4 "Module Addressing".

A maximum of eight digital I/O modules can be operated in packed mode.

CAN ID [1]	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
414	Module 1	Module 2	Module 3	Module 4	Module 5	Module 6	Module 7	Module 8

[1] CAN ID = 414 + (nd - 1) x 4   nd .... Node number of the CAN slave = 1

A maximum of four digital I/O module can be run in unpacked mode.

Module	CAN ID [1]	Byte
1	414	Outputs 1 - 8
2	415	Outputs 1 - 8
3	416	Outputs 1 - 8
4	417	Outputs 1 - 8

[1] CAN ID = 414 + (nd - 1) x 4 + (ma - 1)   nd ... Node number of the CAN slave = 1
ma ... Module address of digital I/O modules = 1 - 4

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

8.8.10 Module Status

Evaluation of the module status is explained using an example in Chapter 4 "Module Addressing".

	Bit	Description
	7	0 ....Module voltage not present or too low 1 ....Module voltage OK
	6	Digital module = 0
	5	x ....Not defined, masked out
	0 - 4	Module code = $0B

0	x	0	1	0	1	1

7   0

8.9 DO720

8.9.1 Technical Data
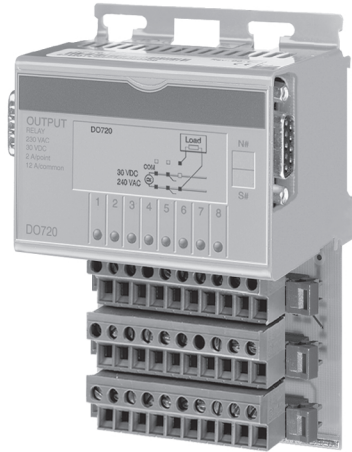
Terminal block is not included in the delivery.

Module ID	DO720
General Information	
Model Number	7DO720.7
Short Description	2003 digital output module, 8 relay outputs 240 VAC / 30 VDC, 2 A, Order terminal blocks separately!
C-UL-US Listed	Yes
B&R ID Code	$E2
Amount	
EX270	2
CP430	4
CP470, CP770 CP474, CP476, CP774 EX470, EX770 EX477, EX777	8
Static Characteristics	
Module Type	B&R 2003 I/O Module
Number of Outputs	8
Type	Relay / N.O.
Switching Voltage Nominal Maximum	240 VAC / 30 VDC 264 VAC / 110 VDC
Continuous Current per Output Module	2 A 12 A
Contact Resistance	30 mΩ at 6 VDC, 1 A

Module ID	DO720
Maximum Switching Power (AC)	480 VA
Maximum Switching Power (DC)	60 W
Power Consumption	Max. 1.4 W
Protection Characteristics	
Short Circuit Protection	Fuse 16 AT (root)
Dynamic Characteristics	

Switching Delay
    log 0 - log 1
    log 1 - log 0
	Max. 10 ms
	Max. 10 ms

Operating Characteristics	
Electrical Isolation	Output - PCC

Dielectric Strength
    Contact - Contact
    Contact - Coil
	750 VAC / 1 min
	2000 VAC / 1 min

Mechanical Characteristics	
Dimensions	B&R 2003 single width

8.9.2 Status LEDs

The Status LEDs 1 - 8 (orange) show the logical state of the corresponding output.

8.9.3 Output Circuit Diagram
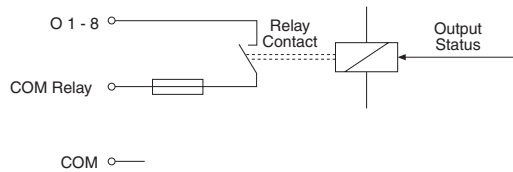
8.9.4 Legend Sheets

A legend sheet can be slid into the front of the module from above. The module circuit is shown on the back. The outputs can be labelled on the front.
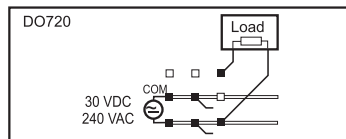
8.9.5 Connections

8.9.6 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Variable declaration with PCC 2003 CPU and remote slaves

Function	Variable Declaration				
	Scope	Data Type	Length	Module type	Chan.
Single digital output (channel x)	tc_global	BIT	1	Digit. Out	1 ... 8
Module status	tc_global	BYTE	1	Status In	0

Variable declaration with CAN slaves

Function	Variable Declaration				
	Scope	Data Type	Length	Module type	Chan.
Single digital output (channel x)	tc_global	BIT	1	Digit. Out	1 ... 8

<u>Module status</u>

The module status for CAN slaves can only be read using command codes. The command codes are explained in Chapter 5, "CAN Bus Controller Functions", section "Command Codes and Parameters". An example is provided in Chapter 4 "Module Addressing".

8.9.7 Access Using CAN IDs

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer. Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing". The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

CAN ID Packed

A maximum of eight digital I/O modules can be operated in packed mode.

CAN ID [1]	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
414	Module 1	Module 2	Module 3	Module 4	Module 5	Module 6	Module 7	Module 8

[1] CAN ID = 414 + (nd - 1) x 4   nd .... Node number of the CAN slave = 1

CAN ID Unpacked

A maximum of four digital I/O module can be run in unpacked mode.

Module	CAN ID [1]	Byte
1	414	Outputs 1 - 8
2	415	Outputs 1 - 8
3	416	Outputs 1 - 8
4	417	Outputs 1 - 8

[1] CAN ID = 414 + (nd - 1) x 4 + (ma - 1)   nd ... Node number of the CAN slave = 1

ma ... Module address of digital I/O modules = 1 - 4

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

8.9.8 Module Status

Evaluation of the module status is explained using an example in Chapter 4 "Module Addressing".

	Bit	Description
	7	x ....Not defined, masked out
	6	Digital module = 0
	5	x ....Not defined, masked out
	0 - 4	Module code = $02

x	0	x	0	0	0	1	0

7   0

8.10 DO721

8.10.1 Technical Data
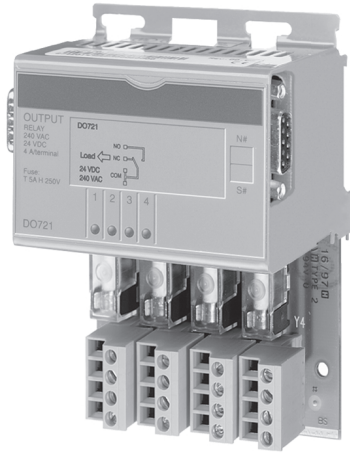
Module ID	DO721
General Information	
Model Number	7DO721.7
Short Description	2003 digital output module, 4 relay outputs 240 VAC / 24 VDC, 4 A
C-UL-US Listed	Yes
B&R ID Code	$E6
Amount	
EX270	2
CP430	4
CP470, CP770 CP474, CP476, CP774 EX470, EX770 EX477, EX777	8
Static Characteristics	
Module Type	B&R 2003 I/O Module
Number of Outputs	4 isolated channels
Type	Relay / Changeover
Switching Voltage Nominal at 4 A Maximum Maximum Voltage Relay-Relay	240 VAC / 24 VDC 264 VAC / 125 VDC 460 VAC
Continuous Current per Output Module	4 A 16 A
Starting Current	15 A / 200 ms
Contact Resistance	100 mΩ at 6 VDC, 100 mA

Module ID	DO721
Maximum Switching Power (AC)	1000 VA
Maximum Switching Power (DC)	120 W at 30 VDC
Power Consumption	Max. 1.4 W
Protection Characteristics	
Short Circuit Protection	Fuse T 5A H / 250 V per output
Dynamic Characteristics	

Switching Delay
   log 0 - log 1
   log 1 - log 0
	Max. 10 ms
	Max. 10 ms

Operating Characteristics	

Electrical Isolation
   Output - PCC
   Output - Output
	Yes
	Yes

Dielectric Strength
   Contact - Contact
   Contact - PCCS, Relay - Relay
	1.4 kV with 1.2 x 50 μs pulse
	4 kV with 1.2 x 50 μs pulse

Mechanical Characteristics	
Dimensions	B&R 2003 single width

8.10.2 Status LEDs

The Status LEDs 1 - 4 (orange) show the logical state of the corresponding output.

8.10.3 Output Circuit Diagram
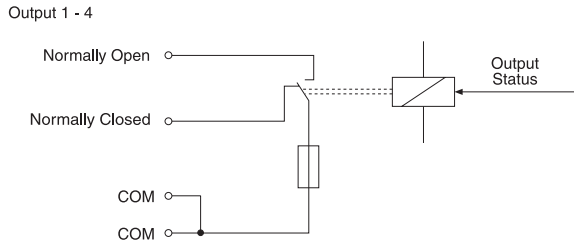
8.10.4 Legend Sheets

A legend sheet can be slid into the front of the module from above. The module circuit is shown on the back. The outputs can be labelled on the front.

8.10.5 Connection

```
Output      N.O. ○   N.O. ○   N.O. ○   N.O. ○
Output      N.C. ○   N.C. ○   N.C. ○   N.C. ○
                 ○        ○        ○        ○
+24 VDC/   ◄──  ○        ○        ○        ○
240 VAC
```

8.10.6 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Variable declaration with PCC 2003 CPU and remote slaves

Function	Variable Declaration				
	Scope	Data Type	Length	Module type	Chan.
Single digital output (channel x)	tc_global	BIT	1	Digit. Out	1 ... 4
Module status	tc_global	BYTE	1	Status In	0

Variable declaration with CAN slaves

Function	Variable Declaration				
	Scope	Data Type	Length	Module type	Chan.
Single digital output (channel x)	tc_global	BIT	1	Digit. Out	1 ... 4

Module status

The module status for CAN slaves can only be read using command codes. The command codes are explained in Chapter 5, "CAN Bus Controller Functions", section "Command Codes and Parameters".
An example is provided in Chapter 4 "Module Addressing".

8.10.7 Access Using CAN IDs

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

CAN ID Packed

A maximum of eight digital I/O modules can be operated in packed mode.

CAN ID [1]	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
414	Module 1	Module 2	Module 3	Module 4	Module 5	Module 6	Module 7	Module 8

[1] CAN ID = 414 + (nd - 1) x 4   nd .... Node number of the CAN slave = 1

CAN ID Unpacked

A maximum of four digital I/O module can be run in unpacked mode.

Module	CAN ID [1]	Byte
1	414	Outputs 1 - 4
2	415	Outputs 1 - 4
3	416	Outputs 1 - 4
4	417	Outputs 1 - 4

[1] CAN ID = 414 + (nd - 1) x 4 + (ma - 1)   nd ... Node number of the CAN slave = 1

ma ... Module address of digital I/O modules = 1 - 4

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

8.10.8 Module Status

Evaluation of the module status is explained using an example in Chapter 4 "Module Addressing".

	Bit	Description
	7	x ....Not defined, masked out
	6	Digital module = 0
	5	x ....Not defined, masked out
	0 - 4	Module code = $06

x	0	x	0	0	1	1	0

7   0

8.11 DO722

8.11.1 Technical Data
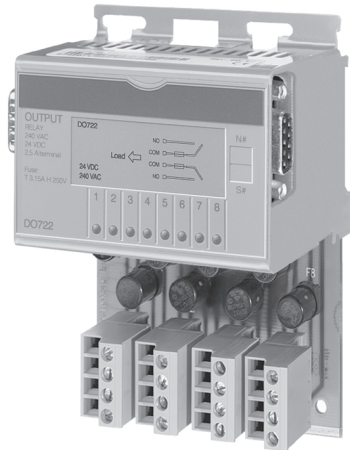
Module ID	DO722
General Information	
Model Number	7DO722.7
Short Description	2003 digital output module, 8 relay outputs 240 VAC / 24 VDC, 2.5 A
C-UL-US Listed	Yes
B&R ID Code	$F2
Amount	
EX270	2
CP430	4
CP470, CP770 CP474, CP476, CP774 EX470, EX770 EX477, EX777	8
Static Characteristics	
Module Type	B&R 2003 I/O Module
Number of Outputs	8 isolated channels
Type	Relay / N.O.
Switching Voltage Nominal at 2.5 A Maximum Maximum Voltage Relay-Relay	 240 VAC / 24 VDC 264 VAC / 125 VDC 460 VAC
Continuous Current per Output Module	 2.5 A 20 A
Maximum Switching Power (AC)	625 VA
Maximum Switching Power (DC)	75 W at 30 VDC

Module ID	DO722
Power Consumption	Max. 1.4 W
Protection Characteristics	
Short Circuit Protection	Fuse T 3.15A H / 250 V per output
Dynamic Characteristics	

Switching Delay
   log 0 - log 1
   log 1 - log 0
	Max. 10 ms
	Max. 10 ms

Operating Characteristics	

Electrical Isolation
   Output - PCC
   Output - Output
	Yes
	Yes

Mechanical Characteristics	
Dimensions	B&R 2003 single width

8.11.2 Status LEDs

The Status LEDs 1 -8 (orange) show the logical state of the corresponding output.

8.11.3 Output Circuit Diagram
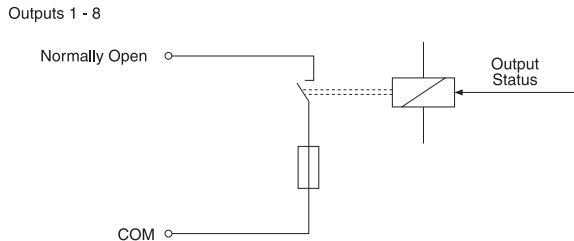
8.11.4 Legend Sheets

A legend sheet can be slid into the front of the module from above. The module circuit is shown on the back. The outputs can be labelled on the front.

8.11.5 Connection

```
O 1   ○     O 3   ○     O 5   ○     O 7   ○
      ○           ○           ○           ○
      ○           ○           ○           ○
O 2   ○     O 4   ○     O 6   ○     O 8   ○
```

8.11.6 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Variable declaration with PCC 2003 CPU and remote slaves

Function	Variable Declaration				
	Scope	Data Type	Length	Module type	Chan.
Single digital output (channel x)	tc_global	BIT	1	Digit. Out	1 ... 8
Module status	tc_global	BYTE	1	Status In	0

Variable declaration with CAN slaves

Function	Variable Declaration				
	Scope	Data Type	Length	Module type	Chan.
Single digital output (channel x)	tc_global	BIT	1	Digit. Out	1 ... 8

Module status

The module status for CAN slaves can only be read using command codes. The command codes are explained in Chapter 5, "CAN Bus Controller Functions", section "Command Codes and Parameters".
An example is provided in Chapter 4 "Module Addressing".

8.11.7 Access Using CAN IDs

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

CAN ID Packed

A maximum of eight digital I/O modules can be operated in packed mode.

CAN ID [1]	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
414	Module 1	Module 2	Module 3	Module 4	Module 5	Module 6	Module 7	Module 8

[1] CAN ID = 414 + (nd - 1) x 4   nd .... Node number of the CAN slave = 1

CAN ID Unpacked

A maximum of four digital I/O module can be run in unpacked mode.

Module	CAN ID [1]	Byte
1	414	Outputs 1 - 8
2	415	Outputs 1 - 8
3	416	Outputs 1 - 8
4	417	Outputs 1 - 8

[1] CAN ID = 414 + (nd - 1) x 4 + (ma - 1)   nd ... Node number of the CAN slave = 1

ma ... Module address of digital I/O modules = 1 - 4

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

8.11.8 Module Status

Evaluation of the module status is explained using an example in Chapter 4 "Module Addressing".

		Bit	Description
		7	x ....Not defined, masked out
		6	Digital module = 0
		5	x ....Not defined, masked out
		0 - 4	Module code = $12

x	0	x	1	0	0	1	0

7   0
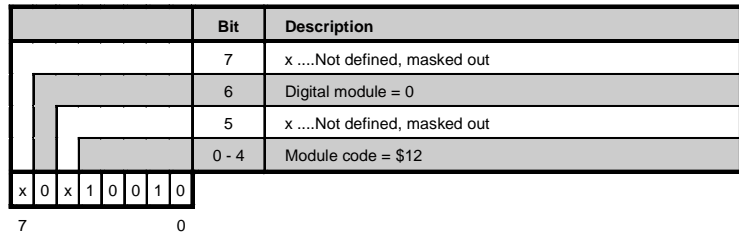
9 DIGITAL MIXED MODULES

9.1 GENERAL INFORMATION

Digital mixed modules are a combination of digital input and digital output modules. The status of the digital inputs and outputs are indicated with status LEDs.

9.2 OVERVIEW

Module	DM435	DM438	DM465
Number of Inputs	8	8	16
Nom. Input Voltage	24 VDC	24 VDC	24 VDC
Number of Outputs	8	8	16
Nom. Switching Voltage	24 VDC	24 VDC	24 VDC
Continuous Current	0.5 A	0.5 A	0.5 A
Connection	Terminal block	D-type connector	Terminal block

9.3 PROGRAMMING

The digital inputs and outputs are accessed directly in the application program using a variable name. The relationship between an input or output channel for a certain module and the variable name is defined in the variable declaration. The declaration is made identically for each programming language using a table editor.

9.4 DM435

9.4.1 Technical Data

Terminal block is not included in the delivery.

Module ID	DM435
General Information	
Model Number	7DM435.7
Short Description	2003 digital mixed module, 8 inputs 24 VDC, 1 ms, sink/source, 8 transistor outputs 24 VDC, 0.5 A, Order terminal blocks separately!
C-UL-US Listed	Yes
B&R ID Code	$E3
Module Type	B&R 2003 I/O Module
Amount	
CP430, EX270	4
CP470, CP770 CP474, CP476, CP774 EX470, EX770 EX477, EX777	8
Voltage and Output Monitoring (LED: OK)	Yes Supply voltage >18 V, Outputs OK
Power Consumption	Max. 0.5 W
Inputs	
Number of Inputs	8
Wiring	Sink or source
Input Voltage Minimum Nominal Maximum	18 VDC 24 VDC 30 VDC
Switching Threshold LOW HIGH	<5 V >15 V

Module ID	DM435
Input Delay	Max. 1 ms
Input Current at Nominal Voltage	Approx. 8 mA
Electrical Isolation	Input - PCC
Outputs	
Number of Outputs	8
Type	Highside Driver IC (Transistor)

Switching Voltage
  Minimum
  Nominal
  Maximum
	18 VDC
	24 VDC
	30 VDC

Continuous Current per
  Output
  Module
	Max. 0.5 A
	Max. 4 A

Leakage Current when Switched Off	12 μA
Overload Protection	Yes
Continuous Short Circuit Current	Typ. 4 A
Internal Protective Circuit	Yes
Negative Anode Potential when Switching Off Inductive Loads	47 V

Switching Delay
  log 0 - log 1
  log 1 - log 0
	Max. 450 μs
	Max. 450 μs

Electrical Isolation	Output - PCC
Mechanical Properties	
Dimensions	B&R 2003 single width

9.4.2 Status LEDs

The green/orange status LEDs 1 - 8 show the logical state of the corresponding inputs/output.

The LED OK (orange) indicates that the supply voltage for the inputs and outputs is present. The LED is lit starting with a supply voltage of approx. 18 VDC.
If voltage is incorrectly placed on an output which is not set, the LED goes out.

9.4.3 Input Circuit Diagram
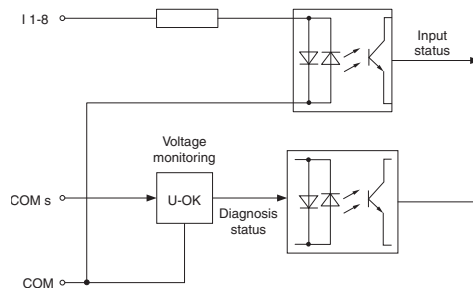
9.4.4 Output Circuit Diagram
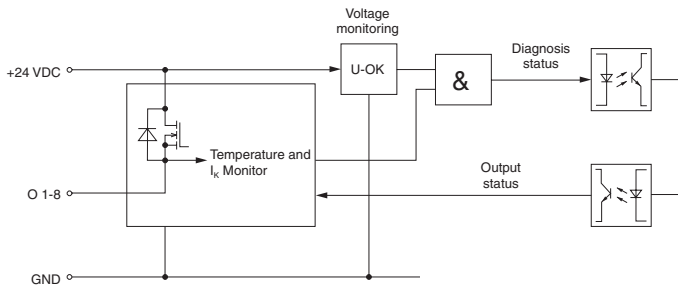
9.4.5 Legend Sheets

A legend sheet can be slid into the front of the module from above. The module circuit is shown on the back. The inputs/outputs can be labelled on the front.
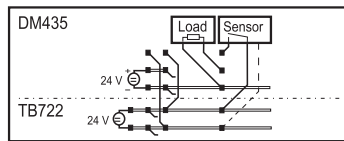
9.4.6 Connections

Terminal block TB722 is used to supply the inputs.

Inputs Sink Connection

Reference Potential
for Sensor

Active Signal Level
of the Inputs (COM s)

Reference Potential
of the Inputs (COM)

1   2   3   4   ...

TB722:
All terminals in a terminal row
are connected internally.

Supply Voltage
for the Inputs
+24 VDC

Next Module

Reference Potential
for the Inputs
GND, 0 VDC

Inputs Source Connection

Reference Potential
for Sensor

Active Signal Level
for the Inputs (COM s)

Reference Potential
for the Inputs (COM)

1 2 3 4 ...

TB722:
All terminals in a terminal row
are connected internally.

Reference Potential
for the Inputs
GND,0 VDC

Next Module

Supply Voltage
for the Inputs
+24 VDC

Output Circuit

Load

1 2 3 4 ...

Supply Voltages
for the Outputs
+24 VDC

Next Module

Reference Potential
for the Outputs
GND, 0 VDC

9.4.7 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Variable declaration with PCC 2003 CPU and remote slaves

Function	Variable Declaration				
	Scope	Data Type	Length	Module type	Chan.
Single digital input (channel x)	tc_global	BIT	1	Digit. In	1 ... 8
Single digital output (channel x)	tc_global	BIT	1	Digit. Out	1 ... 8
Module status	tc_global	BYTE	1	Status In	0

Variable declaration with CAN slaves

Function	Variable Declaration				
	Scope	Data Type	Length	Module type	Chan.
Single digital input (channel x)	tc_global	BIT	1	Digit. In	1 ... 8
Single digital output (channel x)	tc_global	BIT	1	Digit. Out	1 ... 8

Module status

The module status for CAN slaves can only be read using command codes. The command codes are explained in Chapter 5, "CAN Bus Controller Functions", section "Command Codes and Parameters". An example is provided in Chapter 4 "Module Addressing".

9.4.8 Access Using CAN IDs

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

Digital Inputs

A maximum of eight digital I/O modules can be run in packed mode.

CAN ID [1]	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
286	Module 1	Module 2	Module 3	Module 4	Module 5	Module 6	Module 7	Module 8

[1] CAN ID = 286 + (nd - 1) x 4   nd .... Node number of the CAN slave = 1

A maximum of four digital I/O module can be run in unpacked mode.

Module	CAN ID [1]	Byte
1	286	Inputs 1 - 8
2	287	Inputs 1 - 8
3	288	Inputs 1 - 8
4	289	Inputs 1 - 8

[1] CAN ID = 286 + (nd - 1) x 4 + (ma - 1)   nd ... Node number of the CAN slave = 1

ma ... Module address of digital I/O modules = 1 - 4

Digital Outputs

A maximum of eight digital I/O modules can be run in packed mode.

CAN ID [1]	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
414	Module 1	Module 2	Module 3	Module 4	Module 5	Module 6	Module 7	Module 8

[1] CAN ID = 414 + (nd - 1) x 4   nd .... Node number of the CAN slave = 1

A maximum of four digital I/O module can be run in unpacked mode.

Module	CAN ID [1]	Byte
1	414	Outputs 1 - 8
2	415	Outputs 1 - 8
3	416	Outputs 1 - 8
4	417	Outputs 1 - 8

[1] CAN ID = 414 + (nd - 1) x 4 + (ma - 1)   nd ... Node number of the CAN slave = 1

ma ... Module address of digital I/O modules = 1 - 4

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

9.4.9 Module Status

Evaluation of the module status is explained using an example in Chapter 4 "Module Addressing".

							Bit	Description
							7	0 ....No supply voltage or supply voltage too low for digital inputs 1 ....Module voltage OK
							6	Digital module = 0
							5	0 ....No error, the supply voltage for the digital outputs is OK 1 ....Short circuit, over-temperature or the supply voltage for the digital outputs is not OK
							0 - 4	Module code = $03
0		0	0	0	1	1		

7   0

9.5 DM438

9.5.1 Technical Data
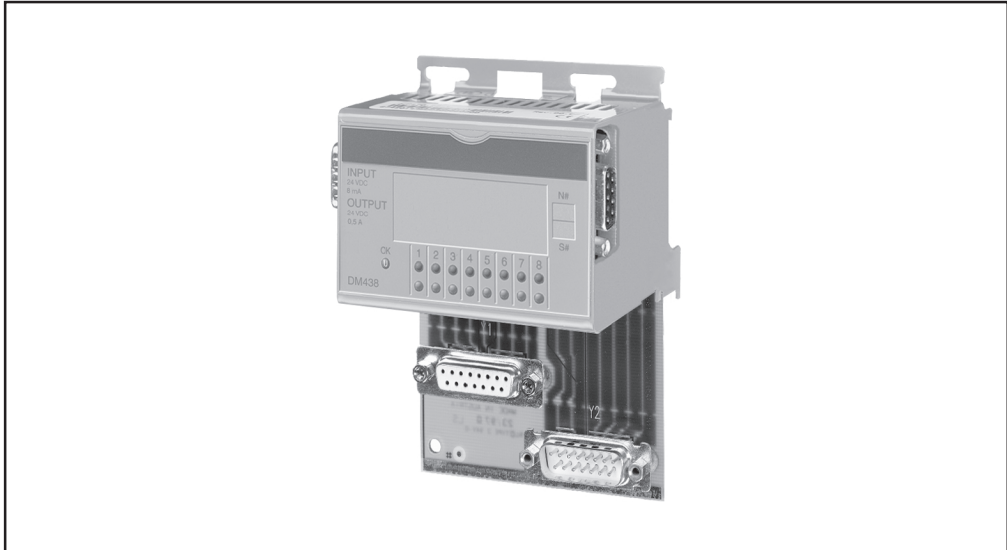
Module ID	DM438
General Information	
Model Number	7DM438.72
Short Description	2003 digital mixed module, 8 inputs 24 VDC, 1 ms, sink/source, 8 transistor outputs 24 VDC, 0.5 A
C-UL-US Listed	in preparation
B&R ID Code	$E7
Module Type	B&R 2003 I/O Module
Amount	
CP430, EX270	4
CP470, CP770 CP474, CP476, CP774 EX470, EX770 EX477, EX777	8
Voltage and Output Monitoring (LED: OK)	Yes Supply voltage >18 V, Outputs OK
Power Consumption	Max. 0.5 W
Inputs	
Number of Inputs	8
Wiring	Sink or source
Input Voltage Minimum Nominal Maximum	18 VDC 24 VDC 30 VDC
Switching Threshold LOW HIGH	<5 V >15 V

Module ID	DM438
Input Delay	Max. 1 ms
Input Current at Nominal Voltage	Approx. 8 mA
Electrical Isolation	Input - PCC
Outputs	
Number of Outputs	8
Type	Highside Driver IC (Transistor)

Switching Voltage
    Minimum
    Nominal
    Maximum
	18 VDC
	24 VDC
	30 VDC

Continuous Current per
    Output
    Module
	Max. 0.5 A
	Max. 4 A

Leakage Current when Switched Off	12 μA
Overload Protection	Yes
Continuous Short Circuit Current	Typ. 4 A
Internal Protective Circuit	Yes
Negative Anode Potential when Switching Off Inductive Loads	47 V

Switching Delay
    log 0 - log 1
    log 1 - log 0
	Max. 450 μs
	Max. 450 μs

Electrical Isolation	Output - PCC
Mechanical Properties	
Dimensions	B&R 2003 single width

9.5.2 Status LEDs

The green/orange status LEDs 1 - 8 show the logical state of the corresponding inputs/output.

The LED OK (orange) indicates that the supply voltage for the inputs and outputs is present. The LED is lit starting with a supply voltage of approx. 18 VDC.

9.5.3 Input Circuit Diagram
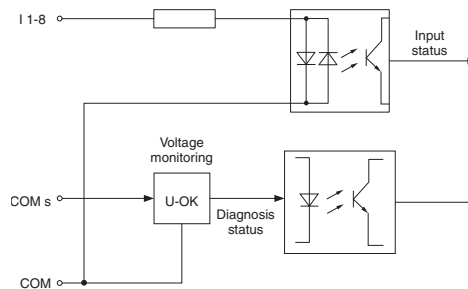
9.5.4 Output Circuit Diagram
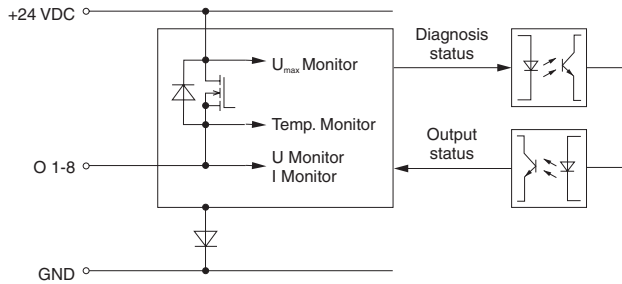
9.5.5 Legend Sheets

A legend sheet can be slid into the front of the module from above. The module circuit is shown on the back. The inputs/outputs can be labelled on the front.
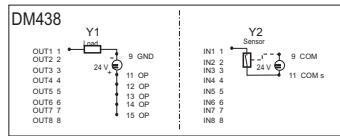
9.5.6 Output Pin Assignments (Y1)

15 Pin D-Type Connector

Y1
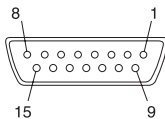
Pin	Outputs – Socket Y1	
1	Output 1	
2	Output 2	
3	Output 3	
4	Output 4	
5	Output 5	
6	Output 6	
7	Output 7	
8	Output 8	
9	GND	Reference potential Y1
10	free	
11	+24 VDC	Output supply Y1
12	+24 VDC	Output supply Y1
13	+24 VDC	Output supply Y1
14	+24 VDC	Output supply Y1
15	+24 VDC	Output supply Y1

9.5.7 Pin Assignments for Inputs (Y2)

15 Pin D-Type Connector

Y2
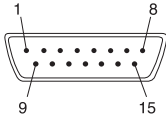
Pin	Inputs – Plug Y2
1	Input 1
2	Input 2
3	Input 3
4	Input 4
5	Input 5
6	Input 6
7	Input 7
8	Input 8
9	COM Reference potential Y2
10	free
11	COM s Input supply Y2
12	free
13	free
14	free
15	free

9.5.8 Input Wiring

Inputs can be wired as either sink or source circuits.

Sink Connection

For sink wiring (current consumer from the sensor's point of view), the COM connection is wired to GND and the inputs are connected to sensors that switch to 24 VDC.
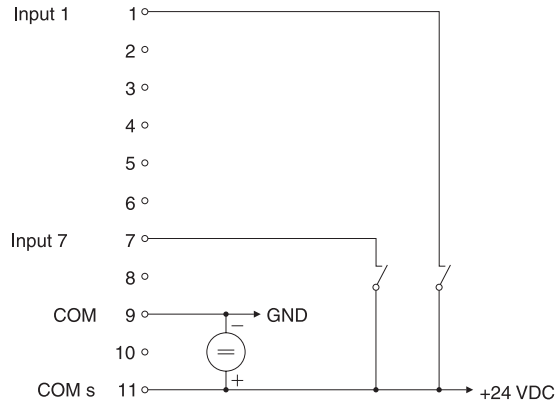
Source Connection

For source wiring (current supplier from the sensor's point of view), the COM connection is wired to +24 VDC and the inputs are connected sensors that switch to GND.

Input 1   1
    2
    3
    4
    5
    6

Input 7   7
    8

COM   9   +24 VDC
    10

COM s   11   GND

9.5.9 Output Wiring

Output 1   1
    2
    3
    4
    5
    6

Output 7   7
    8   Load   Load

GND   9
    10   U = 24 VDC

+24 VDC   11
+24 VDC   12
+24 VDC   13
+24 VDC   14
+24 VDC   15

9.5.10 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Variable declaration with PCC 2003 CPU and remote slaves

Function	Variable Declaration				
	Scope	Data Type	Length	Module type	Chan.
Single digital input (channel x)	tc_global	BIT	1	Digit. In	1 ... 8
Single digital output (channel x)	tc_global	BIT	1	Digit. Out	1 ... 8
Module status	tc_global	BYTE	1	Status In	0

Variable declaration with CAN slaves

Function	Variable Declaration				
	Scope	Data Type	Length	Module type	Chan.
Single digital input (channel x)	tc_global	BIT	1	Digit. In	1 ... 8
Single digital output (channel x)	tc_global	BIT	1	Digit. Out	1 ... 8

Module status

The module status for CAN slaves can only be read using command codes. The command codes are explained in Chapter 5, "CAN Bus Controller Functions", section "Command Codes and Parameters".
An example is provided in Chapter 4 "Module Addressing".

9.5.11 Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

Digital Inputs

A maximum of eight digital I/O modules can be operated in packed mode.

CAN ID [1]	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
286	Module 1	Module 2	Module 3	Module 4	Module 5	Module 6	Module 7	Module 8

[1] CAN ID = 286 + (nd - 1) x 4   nd .... Node number of the CAN slave = 1

A maximum of four digital I/O module can be run in unpacked mode.

Module	CAN ID [1]	Byte
1	286	Inputs 1 - 8
2	287	Inputs 1 - 8
3	288	Inputs 1 - 8
4	289	Inputs 1 - 8

[1] CAN ID = 286 + (nd - 1) x 4 + (ma - 1)   nd ... Node number of the CAN slave = 1
          ma ... Module address of digital I/O modules = 1 - 4

Digital Outputs

A maximum of eight digital I/O modules can be operated in packed mode.

CAN ID [1]	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
414	Module 1	Module 2	Module 3	Module 4	Module 5	Module 6	Module 7	Module 8

[1] CAN ID = 414 + (nd - 1) x 4   nd .... Node number of the CAN slave = 1

A maximum of four digital I/O module can be run in unpacked mode.

Module	CAN ID [1]	Byte
1	414	Outputs 1 - 8
2	415	Outputs 1 - 8
3	416	Outputs 1 - 8
4	417	Outputs 1 - 8

[1] CAN ID = 414 + (nd - 1) x 4 + (ma - 1)   nd ... Node number of the CAN slave = 1
          ma ... Module address of digital I/O modules = 1 - 4

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

9.5.12 Module Status

Evaluation of the module status is explained using an example in Chapter 4 "Module Addressing".

							Bit	Description
							7	0 ....No supply voltage or supply voltage too low for digital inputs 1 ....Module voltage OK
							6	Digital module = 0
							5	0 ....No error, the supply voltage for the digital outputs is OK 1 ....Short circuit, over-temperature or the supply voltage for the digital outputs is not OK
							0 - 4	Module code = $07
0		0	0	1	1	1		

7   0

9.6 DM465

9.6.1 Technical Data
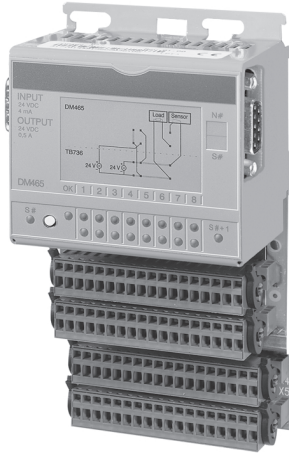
Terminal block is not included in the delivery.

Module ID	DM465
General Information	
Model Number	7DM465.7
Short Description	2003 digital mixed module, 16 inputs 24 VDC, 1 ms, sink, 16 transistor outputs 24 VDC, 0.5 A, Order terminal blocks separately!
C-UL-US Listed	in preparation
B&R ID Code	$F7
Module Type	B&R 2003 I/O Module
Amount [1]	
CP430, EX270	2
CP470, CP770 EX470, EX770 EX477, EX777	4
CP474, CP774	6
CP476	8
Voltage and Output Monitoring (LED: OK)	Yes Supply voltage >18 V, Outputs OK
Power Consumption	Max. 1.1 W
Inputs	
Number of Inputs	16
Wiring	Sink
Input Voltage Minimum Nominal Maximum	18 VDC 24 VDC 30 VDC

Module ID	DM465

Switching Threshold
    LOW
    HIGH
	
	<5 V
	>15 V

Input Delay	Max. 1 ms
Input Current at Nominal Voltage	Approx. 4 mA

Electrical Isolation
    Input - PCC
    Input - Output
	
	Yes
	No

Outputs	
Number of Outputs	16
Type	Highside Driver IC (Transistor)

Switching Voltage
    Minimum
    Nominal
    Maximum
	
	18 VDC
	24 VDC
	30 VDC

Continuous Current per
    Output
    Module
	
	Max. 0.5 A
	Max. 8 A

Leakage Current when Switched Off	12 μA
Overload Protection	Yes
Switching On after Overload Cutoff	Automatically within seconds (depends on the module temperature)
Continuous Short Circuit Current	Typ. 4 A
Internal Protective Circuit	Yes
Negative Anode Potential when Switching Off Inductive Loads	47 V

Switching Delay
    log 0 - log 1
    log 1 - log 0
	
	Max. 450μs
	Max. 450μs

Electrical Isolation
    Output - PCC
    Output - Input
	
	Yes
	No

Mechanical Properties	
Dimensions	B&R 2003 single width

[1] Two logical module slots are required by the module.

9.6.2 Status LEDs

Supply Voltage

The LED OK (orange) indicates that the supply voltage for the inputs and outputs is present. The LED is lit starting with a supply voltage of approx. 18 VDC.
If voltage is inadvertently placed on an output which is not set, the LED goes out.

Inputs/Outputs

Only 8 LEDs each are available for the 16 inputs and outputs.

Green .... Inputs
Orange ..... Outputs

To show all channels, they are divided into groups of eight. You can switch between the two groups using a toggle button on the module front. Two LEDs show which channels are currently being displayed.

LED "S #" .... Channels 1 -8
LED "S # + 1" .... Channels 9 -16

States of the green LEDs

LED Green	
Static "On"	The corresponding input = 1
Static "Off"	The corresponding input = 0

States of the orange LEDs

LED Orange	
Static "On"	The corresponding output = 1
Static "Off"	The corresponding output = 0
Blinking	An error has occurred on the respective output

States of LED "S #"

S #	
Static "On"	Shows channels 1 - 8 without output error
Blinking Symmetrically	Error on one of the outputs 1 - 8
Pulse	Error on one of the outputs 1 - 8 when showing outputs 9 - 16 ("S # + 1")

States of LED "S # + 1"

S # + 1	
Static "On"	Shows channels 9 -16 without output error
Blinking Symmetrically	Error on one of the outputs 9 -16
Pulse	Error on one of the outputs 9 - 16 when showing outputs 1 - 8 ("S #")

9.6.3 Input Circuit Diagram
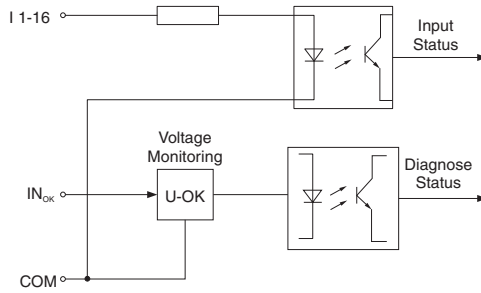
9.6.4 Output Circuit Diagram
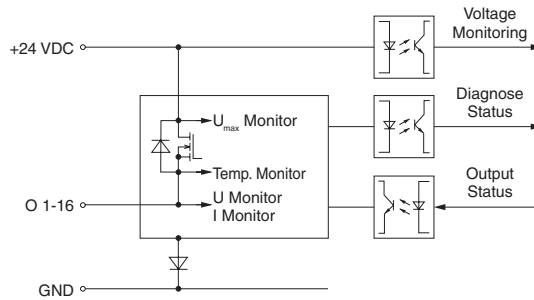
9.6.5 Legend Sheets

A legend sheet can be slid into the front of the module from above. The module circuit is shown on the back. The inputs/outputs can be labelled on the front.
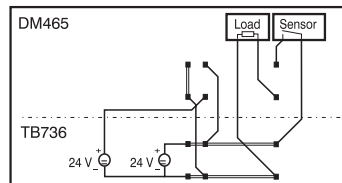
9.6.6 Connections

The supply voltage for the inputs is applied using the terminal block TB722. A separate supply voltage is required for the inputs and outputs. The reference potential for both voltages is connected to the bottom pin connector on the TB736.

Inputs - Sink Connection
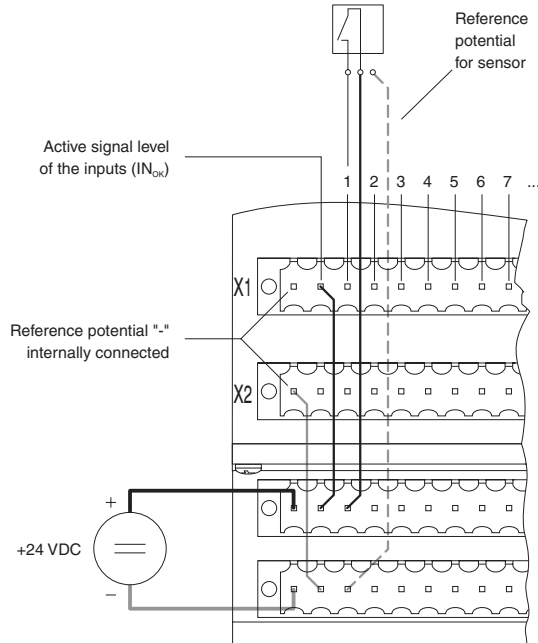
TB736:   All terminals in a terminal row are connected internally.

Output Wiring
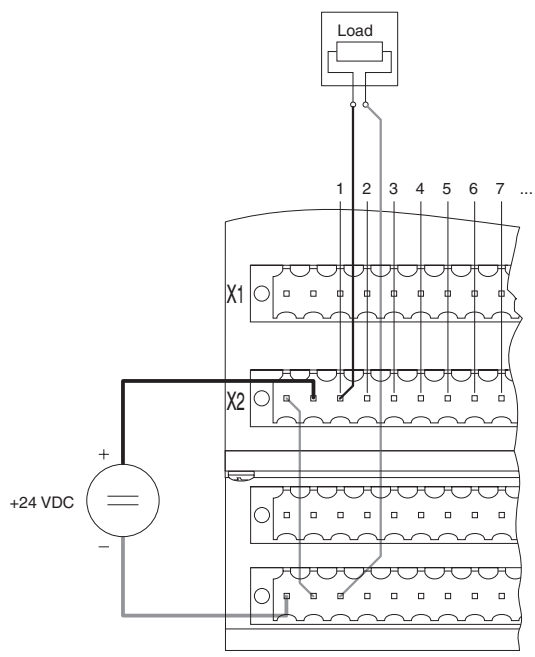
9.6.7 Three Line Connection

If the digital mixed module DM465 is operated using a three line connection, the terminal block TB772 is used as additional jumper terminal.

Reference potential "-" internally connected

+24 VDC for inputs

GND

+24 VDC for outputs

GND
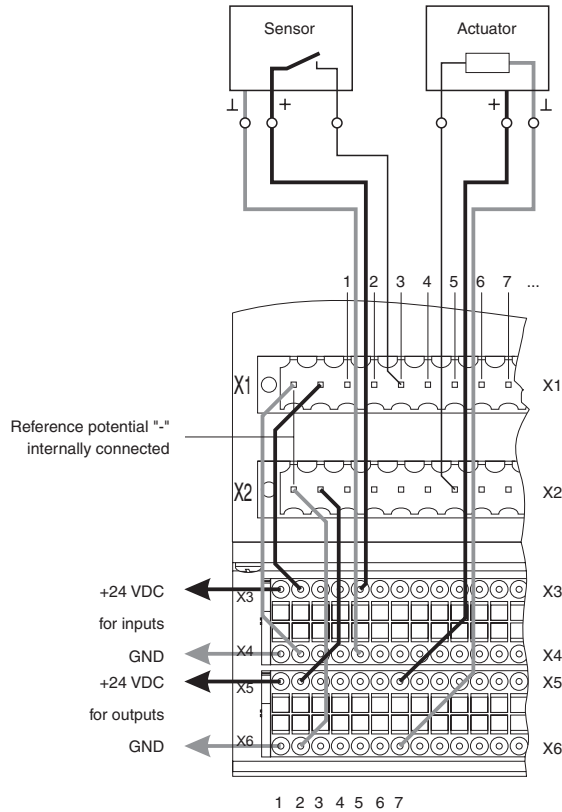
TB772: All terminals in a terminal row are connected internally.

9.6.8 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Variable declaration with PCC 2003 CPU and remote slaves

Designation	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Digital inputs 1 – 8	BIT	Digit. In	1...8	●		Level of digital inputs 1 - 8
Digital outputs 1 – 8	BIT	Digit. Out	1...8		●	Level of digital outputs 1 - 8
Module Status 1	BYTE	Status In	0	●		Module status for the first half of the module
Digital inputs 9 – 16 (module address + 1)	BIT	Digit. In	1...8	●		Level of digital inputs 9 -16
Digital outputs 9 – 16 (module address + 1)	BIT	Digit. Out	1...8		●	Level of digital outputs 9 -16
Module status 2 (module address + 1)	BYTE	Status In	0	●		Module status for the second half of the module

Variable declaration with CAN slaves

Designation	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Digital inputs 1 – 8	BIT	Digit. In	1...8	●		Level of digital inputs 1 - 8
Digital outputs 1 – 8	BIT	Digit. Out	1...8		●	Level of digital outputs 1 - 8
Digital inputs 9 – 16 (module address + 1)	BIT	Digit. In	1...8	●		Level of digital inputs 9 -16
Digital outputs 9 – 16 (module address + 1)	BIT	Digit. Out	1...8		●	Level of digital outputs 9 -16

Module status

The module status for CAN slaves can only be read using command codes. The command codes are explained in Chapter 5, "CAN Bus Controller Functions", section "Command Codes and Parameters".
An example is provided in Chapter 4 "Module Addressing".

9.6.9 Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

Digital Inputs

A maximum of eight digital I/O modules can be operated in packed mode.
The 16 channel module DM465 operates like two 8 channel modules next to each other. If two DM465 modules are used, only six additional digital I/O modules can be used.

The following example shows the structure of the CAN object if four DI435 and two DM 465 modules are used.

CAN ID [1]	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
286	DI435	DI435	DI435	DI435	DM465 I 1 - 8	DM465 I 9 - 16	DM465 I 1 - 8	DM465 I 9 - 16

[1] CAN ID = 286 + (nd - 1) x 4   nd .... Node number of the CAN slave = 1

A maximum of four digital I/O module can be run in unpacked mode.

The following example shows the structure of the CAN object if two DI435 and one DM465 modules are used.

Module	CAN ID [1]	Byte
DI435	286	Inputs 1 - 8
DI435	287	Inputs 1 - 8
DM465	288	Inputs 1 - 8
	289	Inputs 9 - 16

[1] CAN ID = 286 + (nd - 1) x 4 + (ma - 1)   nd ... Node number of the CAN slave = 1
ma ... Module address of digital I/O modules = 1 - 4

Digital Outputs

A maximum of eight digital I/O modules can be operated in packed mode.
The 16 channel module DM465 operates like two 8 channel modules next to each other. If two DM465 modules are used, only six additional digital I/O modules can be used.

The following example shows the structure of the CAN object if four DO722 and two DM 465 modules are used.

CAN ID [1]	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
414	DO722	DO722	DO722	DO722	DM465 O 1 - 8	DM465 O 9 - 16	DM465 O 1 - 8	DM465 O 9 - 16

[1] CAN ID = 414 + (nd - 1) x 4   nd .... Node number of the CAN slave = 1

A maximum of four digital I/O module can be run in unpacked mode.

The following example shows the structure of the CAN object if two DO722 and one DM465 modules are used.

Module	CAN ID [1]	Byte
DO722	414	Outputs 1 - 8
DO722	415	Outputs 1 - 8
DM465	416	Outputs 1 - 8
	417	Outputs 9 - 16

[1] CAN ID = 414 + (nd - 1) x 4 + (ma - 1) nd ... Node number of the CAN slave = 1
ma ... Module address of digital I/O modules = 1 - 4

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

9.6.10 Module Status

Evaluation of the module status is explained using an example in Chapter 4 "Module Addressing".

Module Status 1

	Bit	Description
	7	0 ....No supply voltage or supply voltage too low for digital inputs/outputs 1 ....Module voltage OK
	6	Digital module = 0
	5	0 ....No error, the supply voltage for digital outputs 1 – 8 is OK 1 ....Short circuit, over-temperature or the supply voltage for digital outputs 1 – 8 is not OK Bits 0 - 4 contain the channel number of the first faulty output.
	0 - 4	Module code = $17

```
0   1 0 1 1 1
7             0
```

Module Status 2

	Bit	Description
	7	0 ....No supply voltage or supply voltage too low for digital inputs/outputs 1 ....Module voltage OK
	6	Digital module = 0
	5	0 ....No error, the supply voltage for digital outputs 9 -16 is OK 1 ....Short circuit, over-temperature or the supply voltage for digital outputs 9 -16 is not OK Bits 0 - 4 contain the channel number of the first faulty output.
	0 - 4	Module code = $17

```
0   1 0 1 1 1
7             0
```

10 ANALOG INPUT MODULES

10.1 GENERAL INFORMATION

Measured values (voltages, currents) are converted into number values which can be processed in the PCC using analog input modules.

In the PCC, analog data is always in 16 bit 2s complement regardless of the resolution. In this way, the resolution (number of steps) of the input module does not have to be considered when creating the application program.

Each analog input module is assigned a status LED on the analog adapter module or CP interface. This LED indicates that the input module is working.

10.2 OVERVIEW

Module	AI261	AI294
Number of Channels	1	4
Input Signal	±1 to ±16 mV/V	0 to 4.5 V
Digital Converter Resolution	24 Bit	13 Bit
Remark One input to evaluate	Four inputs to evaluate a full-bridge strain gauge	potentiometers (e.g. displacement gauge)

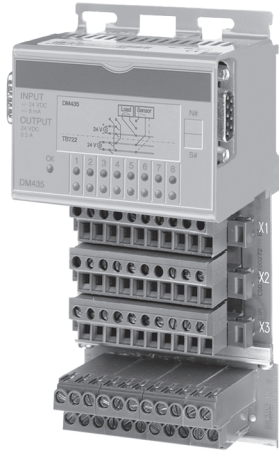| Module	AI351	AI354	AI774
Number of Channels	1	4	4
Input Signal	±10 V or 0 - 20 mA	±10 V	0 - 20 mA
Digital Converter Resolution	12 Bit + sign	12 Bit + sign	12 Bit

10.3 PROGRAMMING

The analog data points are accessed directly in the application program using a variable name. The relationship between the channel for a certain module and the variable name is defined in the variable declaration. The declaration is made identically for each programming language using a table editor.

10.4 AI261

10.4.1 Technical Data

Module ID	AI261
General Information	
Model Number	7AI261.7
Short Description	2003 analog input module, 1 input for evaluation of full-bridge strain gauge, 24 bit, screw-in module
C-UL-US Listed	Yes
B&R ID Code	$20
Slot	AF101 adapter module, CP interface
Static Characteristics	
Module Type	B&R 2003 screw-in module
Input Type	Differential
Number of Inputs	1
Affects of Cable Length	The shielded, twisted pair cable should be as short as possible and run separately to the sensor (isolated from load circuit) without intermediate terminals
Shielding	
Module Side	Using stress relief clamp on the module
Sensor Side	Using HF foil capacitor 10 nF/630 V to grounded sensor housing (use short connection lines)
Maximum Continuous Overload (without damage)	+30 V for all connections except GND
Output of the Digital Value during Overload	
Broken Supply Line	Value goes to 0
Broken Sensor Line	Value goes to ±limit
Software Evaluation	Evaluation using configuration word 12 (module status)
Digital Converter Resolution	24 Bit
Effective Converter Resolution	See Table "Effective Resolution of the Measurement Range in Bits"

Module ID	AI261

Quantization
  Measurement Range ±1...2 mV/V
  Measurement Range ±3...4 mV/V
  Measurement Range ±5..0.8 mV/V
  Measurement Range ±9...15 mV/V
	LSB value (16 Bit)
	275 nV
	550 nV
	1.1 μV
	2.2 μV

Data Format Delivered to the Application Program	Set using software

Measurement Value Preparation
  Calibration
  Linearization
  Conversion
	Using software, also during operation
	y = k * x + d
	To physical units (32 bit representation)

Measurement Range	±1 to ±16 mV/V, Set using software
Input Current	<140 nA
Operating Range / Measurement Sensor	75 to 5000 Ω

Bridge Voltage
  Short Circuit and Overload Protection
  Connection
	4.5 VDC ±3 % / max. 60 mA
	Yes
	4-line connection
	Connection of a 6-line strain gauge cell (see section "6-Line Strain Gauge Cell")

Conversion Method	Sigma Delta
Conversion Time	1 ms

Analog Input Measurement Error
  Maximum Error at 25 °C
  Temperature Coefficient
  Maximum Error over Complete
  Temperature Range
	±55 ppm ±11 μV
	±3 ppm/°C ±1.1 μV/°C
	±0.016 % ±50 μV

Sensor Type	Isolated
Common Mode Rejection	>120 dB at 50 / 60 Hz Scan Frequency ≥ 75 Hz
Common Mode Voltage	1.2 to 3.8 V
Protection	RC protection
Internal Power Consumption	Max. 0.6 W
Dynamic Characteristics	
Application Scan Time	4 - 100 ms
Data Output Rate on the Module	7 - 500 Hz, can be set using software
Scan Repeat Time	1 / data output rate

Response Time 1 LSB
  Data Output Rate > 100 Hz
  Data Output Rate ≤ 100 Hz
	Approx. 250 ms
	Approx. 500 ms

Input Filter Characteristics
  Order
  Transition / Cut-off Frequency
	1st
	6 Hz
	also see diagram "Transient Behavior as Load Changes"

Software Filter
  3 dB Cut-off Frequency
  64 dB Frequency (1st notch)
	0.0395 x data output rate
	0.14 x data output rate

Operating Characteristics	
Isolation Voltage under Normal Operating Conditions between Channel and Bus	No electrical isolation
Missing Codes	Yes, if output range > converter resolution
Non-linearity	±0.0015 % of final value
Mechanical Properties	
Dimensions	B&R 2003 screw-in module

10.4.2 General Information

A full-bridge strain gauge can be used for the following tasks:

- Force measurement
- Elasticity measurement
- Weight measurement
- Pressure measurement
- Stress measurement
- Torque measurement

10.4.3 Effective Resolution of the Measurement Range in Bits

The following table contains an overview of the effective resolution of the measurement range in bits. The corresponding conversion range is given next to it.
The data output rate for the hardware lies between 50 Hz and 500 Hz. Output rates of <50 Hz can also be set in software.
This output rate is achieved because the convertor read out at 8x the set output rate. For the calculation, eight values are added together and then divided by eight.
To determine the effective resolution, look in the table for 8 times the set data output rate.

Example

Set output rate:	25 Hz for a measurement range of 2 mV/V
Effective resolution:	8x set data output rate:
	25 Hz x 8 = 200 Hz $\Rightarrow$ 15 bit or $\pm$18000

Data Output Rate		Measurement Range							
HW	SW	±2 mV/V		±4 mV/V		±8 mV/V		±16 mV/V	
50 Hz	7 Hz	16 Bit	±35000	16.5 Bit	±50000	17 Bit	±69000	17.5 Bit	±99000
100 Hz	12 Hz	15.5 Bit	±25000	16 Bit	±33000	16.5 Bit	±51000	17 Bit	±69000
150 Hz	18 Hz	15.5 Bit	±20000	16 Bit	±29000	16.5 Bit	±42000	17 Bit	±56000
200 Hz	25 Hz	15 Bit	±18000	15 Bit	±27000	16 Bit	±36000	16.5 Bit	±49000
400 Hz	49 Hz	14.5 Bit	±11000	15 Bit	±18000	15.5 Bit	±24000	16 Bit	±36000

10.4.4 Transient Behavior as Load Changes

The following diagram shows the transient behavior as the load changes in relation to the data output rate:

Curve 1 ..... Data output rate   =   50 Hz
Curve 2 ..... Data output rate   = 200 Hz
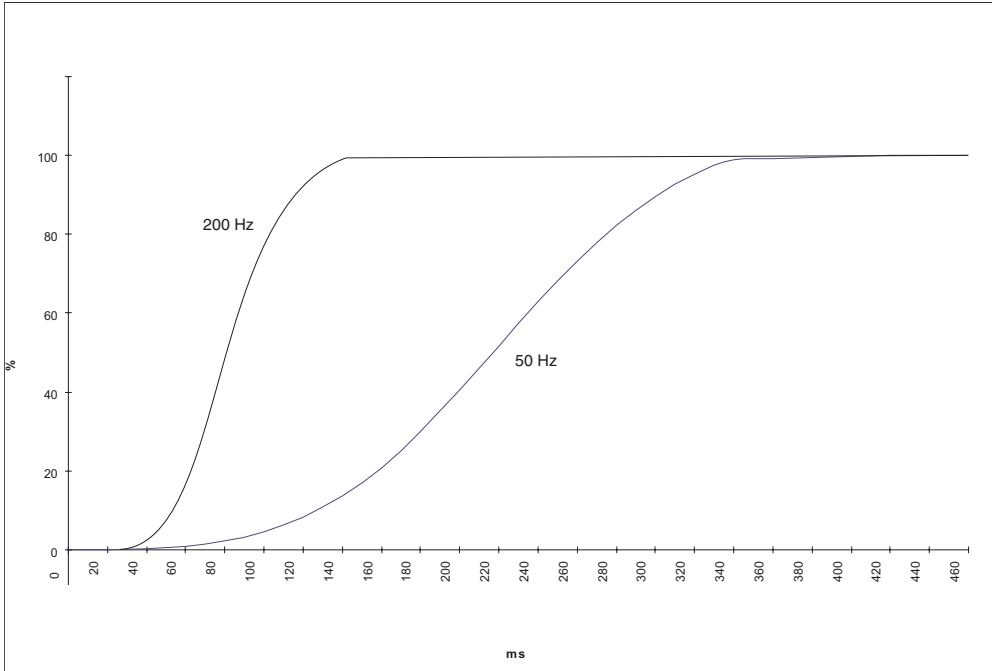
10.4.5 Input Circuit Diagram
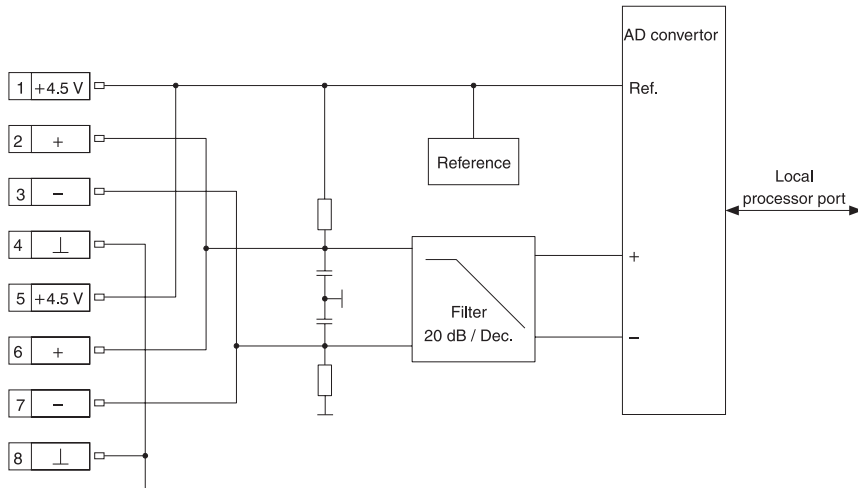
10.4.6 Connection

Terminal Assignments

Connection	Designation	Description
1	+4.5 V	+4.5 V strain gauge supply
2	+	Differential input
3	-	
4	⊥	GND
5	+4.5 V	+4.5 V strain gauge supply
6	+	Differential input
7	-	
8	⊥	GND

Full-Bridge Strain Gauge Wiring

Full-bridge strain gauge

+U

Multi-conductor twisted pair cable

Cable shielding on module side earthed via stress relief clamp

1
2
3
4

Shielding on sensor side via 10 nF / 630 VDC foil capacitor connected to earthed sensor housing (length as short as possible).
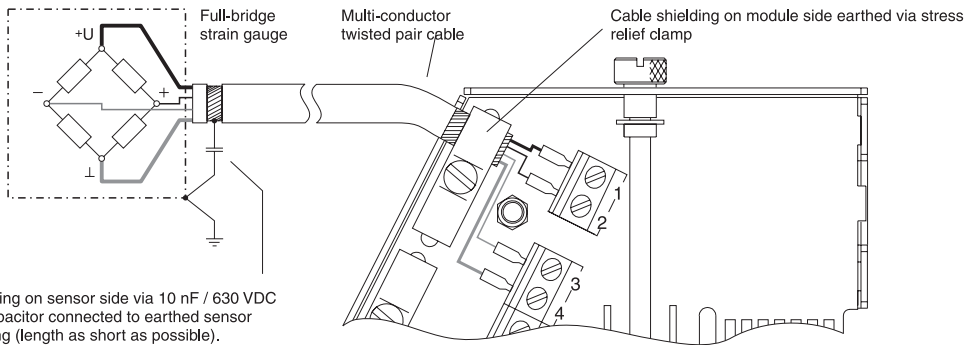
Parallel Connection

For parallel connection of full-bridge strain gauges, please refer to the manufacturer's guidelines.

Parallel connection of two full-bridge strain gauges

1
2
3
4
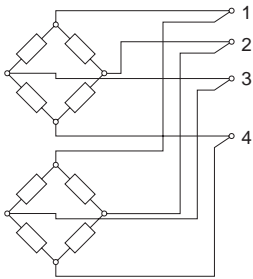
Parallel connection of three full-bridge strain gauges

1
2
3
4

5
6
7
8
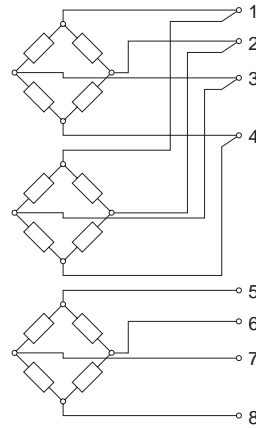
10.4.7 6-Line Strain Gauge Cells

General Information

Precision can be improved by using strain gauge cells with feedback of the bridge voltage. The additional sensor lines compensate for the thermal resistance of the feed lines.

4-Line Concept of the AI261

AI261 modules work with 4-line strain gauge cells. The AI261 concept requires compensation in the measurement system. This compensation eliminates all absolute uncertainty in the measurement circuit, such as component tolerances, effective bridge voltage, or zero offset.
The measurement precision refers to the absolute (compensated) value which will only change as a result of changes in operating temperature.

6-Line Strain Gauge Cells on the AI261

If a 6-line strain gauge cell is connected to the AI261, the line compensation no longer functions. The measurement precision is therefore susceptible to changes in operating temperature. Long cable runs and small cable cross-sections used between the evaluation device (AI261) and the strain gauge cell increase the probability of errors in the measurement system.
The following example shows the discrepancy between the measured value and the actual value when the operating temperature range lies between 25 °C and 55 °C.

Entry

Designation	Value
Accepted (classic) working range	25 °C to 55 °C ($\Delta T = 30$ °C)
Bridge resistance (input resistance)	300 Ω
Copper connection cable, temperature coefficient	0.39 %/K

Formulas for calculation of the table entries

Designation	Formula
Line resistance	$R = 2 * l / (\gamma * A)$ R = 2 * cable length [m] / (56 * cross-section [mm^2])
Resistance change	$\Delta R = R * 0.39 \% * \Delta T$ $\Delta R = R * 0.0039 * 30$
Temperature effects in ppm	(ΔR / Bridge resistance) * 10^6 (ΔR / 300) * 10^6
Temperature effects in %	(ΔR / Bridge resistance) * 100 (ΔR / 300) * 100

Calculated examples for different cable lengths and cross-sections

Designation	Unit	Ex. 1	Ex. 2	Ex. 3	Ex. 4	Ex. 5	Ex. 6	Ex. 7
Cable length	[m]	10	6	6	6	6	6	4
Cross section	[mm^2]	0.25	0.14	0.25	0.22	0.34	0.5	0.5
Resistance (forward and return line)	[Ω]	1.43	1.54	0.86	0.98	0.64	0.43	0.29
Resistance change at ΔT = 30 °C	[mΩ]	168	181	101	115	75	51	34
Temperature effects	[ppm]	560	604	337	384	250	170	114
Temperature effects	[%]	0.056	0.060	0.034	0.038	0.025	0.017	0.011
Deviation for measurement range 0 to 1000 kg	[g]	560	604	337	384	250	170	114

Wiring

In order to reduce cable resistance, the sense lines for 6-line connection should be connected in parallel with the supply lines.
Optimal signal quality can be obtained by using shielded, twisted pair (data) cable. The connections for the two strain gauge supply voltage lines (input), the two sensor lines, and the two bridge differential voltage lines (output) should each use one twisted pair cable run respectively .

Pair 1:   Strain gauge supply +
            Strain gauge supply GND

Pair 2:   Sensor line +
            Sensor line -

Pair 3:   Differential input +
            Differential input -

Strain gauge supply +

Sensor line +   1

Differential input +   2

Differential input -   3

Sensor line -

Strain gauge supply GND   4

10.4.8 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for this module.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	LONG	Transp. In	0	●		Standardized value or calibrated raw value
Configuration word 4	LONG	Transp. In	8	●		Calibrated raw value during standardizing/taring
	LONG	Transp. Out	8		●	Reference value/tare value as calibrated raw value
Configuration word 6	LONG	Transp. Out	12		●	Standardized reference value/tare value
Configuration word 8	WORD	Transp. Out	16		●	Command number for standardizing and taring
Configuration word 9	WORD	Transp. Out	18		●	Sensor damping
Configuration word 10	WORD	Transp. Out	20		●	Data output rate of the converter
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

10.4.9 Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

Data cannot be packed on the AI261. Therefore one CAN object is transferred per screw-in module. If an adapter module AF101 is equipped with a four AI261 modules, the CAN object has the following structure:

Slot	CAN ID [1]	Word 1		Word 2		Word 3	Word 4
1	542	Data LL	Data ML	Data MH	Data HH	Not used (2 byte data)	
2	543	Data LL	Data ML	Data MH	Data HH	Not used (2 byte data)	
3	544	Data LL	Data ML	Data MH	Data HH	Not used (2 byte data)	
4	545	Data LL	Data ML	Data MH	Data HH	Not used (2 byte data)	

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address of the AF101 = 1

sl .... Slot number of the screw-in module on the AF101 (1 - 4)

⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

10.4.10 Terms

The following terms are used to describe data and configuration words:

Term	Description
Calibrated raw value	Corresponds to the converter value aligned to sensor damping
Standardized value	Corresponds to the calibrated raw value converted to a physical unit
Standardization line	The calibrated raw value is converted to a standardized value along this line
1st / 2nd reference point	The standardizing line is calculated using these coordinates
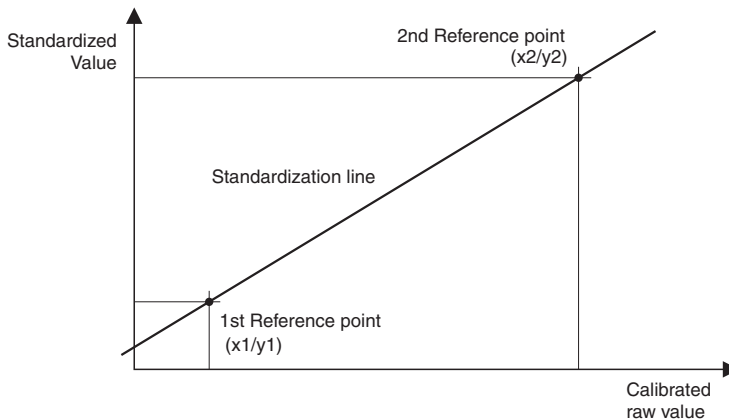
10.4.11 Description of Data and Configuration Words

Data Words 0+1 (read)

Data words 0 and 1 contain the standardized value or the calibrated raw value from the full-bridge strain gauge with 24 bit resolution.

Value Range	
Valid value range	$007FFFFF to $FF800001
Overflow	$7FFFFFFF
Underflow	$80000001
Invalid value	$80000000

Configuration Words 4+5 (read)

During standardization or taring, configuration words 4 and 5 contain the calibrated raw value for the 1st or 2nd reference point of the standardization lines determined by the module.

Configuration Words 4+5 (write)

Configuration words 4 and 5 defines either the first or second reference point as a calibrated raw value.

Configuration Words 6+7 (write)

Configuration words 6 and 7 defines either the first or second reference point as standardized value.

Configuration Word 8 (write)

Configuration word 8 defines the command number for standardization and taring.

		Bit	Description
		8 - 15	0
		4 - 7	Command number for standardization and taring 0............No effects 1 - 5.......See sections "Standardization" and "Taring" 6 - 15.....Reserved
		0 - 3	0

0	0	0	0	0	0	0	0					0	0	0	0
15							8	7				0			
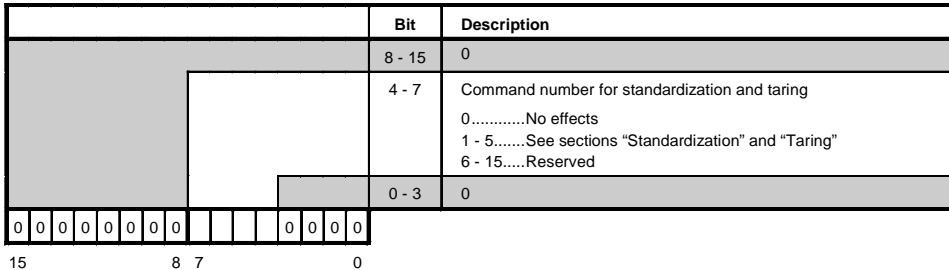
Configuration Word 9 (write)

Configuration word 9 defines sensor damping. When output of calibrated raw values is set, the convertor and value output are started (configuration word 14 = $x800).

Value range: 1 mV/V to 16 mV/V

0 is not allowed
(error code 5000, supplementary code k30ma see App. B "Error Messages")

Configuration Word 10 (write)

Configuration word 10 defines the data output rate of the convertor.

Value range: 7 to 500

0 is not allowed
(error code 5000, supplementary code k30ma see App. B "Error Messages")

See also, section "Effective Resolution of the Measurement Range in Bits"

Configuration Word 12 (read)

Configuration word 12 contains the module status (current status unlatched).

	Bit	Description
	12 - 15	x .... Not defined, masked out
	11	0 .... Converter value ready 1 .... Converter value not yet ready
	10	x .... Not defined, masked out
	9	0 .... Wait for the first conversion after setting the damping 1 .... The first conversion after setting the damping has taken place
	8	0 .... Command to set damping has not yet arrived 1 .... Command to set damping has arrived, Bit 9 cleared
	4 - 7	If this bit pattern is the same as the command number defined in configuration word 8, the command is executed.
	3	x .... Not defined, masked out
	2	0 .... Converter parameter settings are OK 1 .... Converter parameter settings are invalid
	1	0 .... Sensor supply is OK 1 .... Sensor supply is overloaded
	0	0 .... Reference voltage is OK 1 .... Reference voltage not available

x	x	x	x		x						x				
15								8	7					0	

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $20
	0 - 7	x ....Not defined, masked out

0	0	1	0	0	0	0	0	x	x	x	x	x	x	x	x

15   8 7   0

Configuration Word 14 (write)

The module is configured using configuration word 14.

	Bit	Description
	14 - 15	0
	13	0....The standardized value or calibrated raw value from the full-bridge strain gauge will be entered in data words 0 and 1, according to the data output rate.
		1....The standardized value or calibrated raw value from the full-bridge strain gauge will be entered in data words 0 and 1, according to the data output rate. Once the value has been read, the word will be set to invalid ($80000000). This ensures that the value is only read once per data cycle.
		Attention: When operated on the CP interface on an AF101 module with a revision ≥ 02.00, automatic mode must be switched off on the CPU or on the AF101 module!
	12	0
	11	0....Output of the converter value as a standardized value from the full-bridge strain gauge.
		1....Output of the converter value as raw value calibrated to the sensor damping.
	0 - 10	0

0	0		0		0	0	0	0	0	0	0	0	0	0	0

15   8 7   0

10.4.12 AI261 Start-up

An example program is available from B&R. This program explains how to use the AI261.
If you are interested in obtaining this program, contact your local sales representative.

The following steps should be carried out during start-up

Step	Configuration Word	Description
1	14	Configure module
2	10	Define the data output rate of the converter
3	9	Define the sensor damping. When output of calibrated raw values is set, the converter will be started and value output begins (configuration word 14 = $x800).
4		When standardized values are used, (configuration word 14 = $x000) standardizing/taring must be carried out at this point. The converter and value output are started in command number 4 or 5.

10.4.13 Standardization

Through standardization, the calibrated raw value is assigned a value corresponding to the physical unit. The conversion takes place along a standardization line.

The line equation is

$y = k * x + d$

y ...... standardized value

k ...... slope

x ...... calibrated raw value

d ..... y, if x = 0 (Offset)

There are two types of standardization:

- Establishing the standardization line
  (the calibrated raw values for the reference points are not known)

- Standardization when booting
  (standardization line already known)

Establishing the standardization line

This type of standardization is carried out if the calibrated raw values for the reference points are not already known. This is the case when:

- the system is first put into operation.
- the process conditions have changed.

The following commands must be carried out

No.	Commands to carry out
1	Set up conditions for the first reference point (weight, pressure, torque etc.).
2	Enter command number 1 ($0010) in configuration value 8. The converter determines the first reference point of the standardization line.
3	Poll configuration word 12 until the acknowledgement ($xx1x) is received after approx. 1 s.
4	Read the values from configuration words 4+5. These contain the calibrated raw value of the first reference point.
5	Save the calibrated raw value. It must be entered each time the system is booted or after each reset (see "Standardization when booting ").
6	Write the calibrated raw value to configuration words 4+5.
7	Write the standardized value corresponding to the calibrated raw value to configuration words 6+7.
8	Enter command number 2 ($0020) in configuration word 8. The values in configuration words 4+5 and 6+7 will be taken as the first reference point of the standardization line.
9	Poll configuration word 12 until the acknowledgement ($xx2x) is received after <100 ms.
10	Set up conditions for the second reference point
11	Enter command number 3 ($0030) in configuration word 8. The converter determines the second reference point for the standardization line.
12	Poll configuration word 12 until the acknowledgement ($xx3x) is received after approx. 1 s.
13	Read the values from configuration words 4+5. These contain the calibrated raw value of the second reference point.
14	Save the calibrated raw value. It must be entered each time the system is booted or after each reset (see "Standardization when booting").
15	Write the calibrated raw value to configuration words 4+5.
16	Write the standardized value corresponding to the calibrated raw value to configuration words 6+7.
17	Enter command number 4 ($0040) in configuration word 8. The values in configuration words 4+5 and 6+7 are used as the second reference point of the standardization line, the standardization parameters are calculated, the conversion procedure and value output are started.
18	Poll configuration word 12 until the acknowledgement ($xx4x) is received after <100 ms.

Standardization when booting

This type of standardization is carried out if the standardization line is already known (calibrated raw values for the reference points are already known). Carried out during each boot procedure or after resetting the system.

The following commands must be carried out

No.	Commands to carry out
1	Write the calibrated raw value for the first reference point on the standardization line to configuration words 4+5. The calibrated raw values for the first and second reference points are determined while establishing the standardization line.
2	Write the standardized value corresponding to the calibrated raw value in configuration words 6+7.
3	Enter command number 2 ($0020) in configuration word 8. The values in configuration words 4+5 and 6+7 are used as the first reference point of the standardization line.
4	Poll configuration word 12 until the acknowledgement ($xx2x) is received after <100 ms.
5	Write the calibrated raw value for the second reference point of the standardization line to configuration words 4+5 .
6	Write the standardized value corresponding to the calibrated raw value in configuration words 6+7.
7	Enter command number 4 ($0040) in configuration word 8. The values in configuration words 4+5 and 6+7 are used as the second reference point of the standardization line, the standardization parameters are calculated, the conversion procedure and value output is started.
8	Poll configuration value 12 until the acknowledgement ($xx4x) is received after <100 ms.

10.4.14 Taring

Tarig is necessary if the standardization line is shifted because of the process but the slope remains the same. During taring, the offset of the standardization line is recalculated (d_{new}).
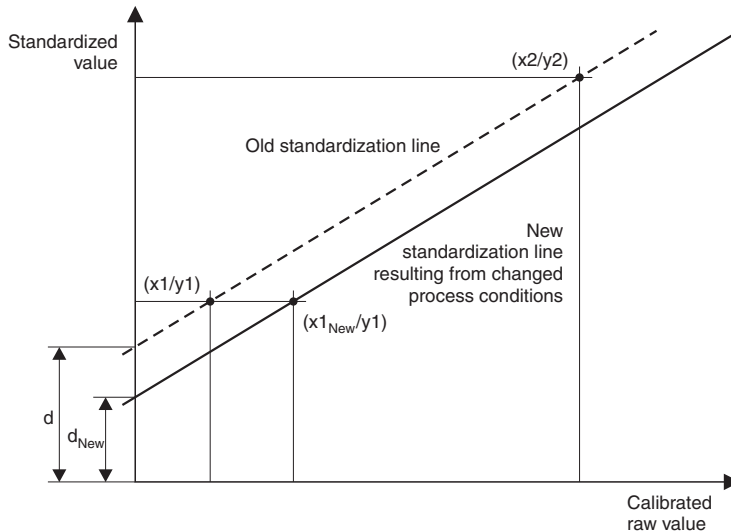
Taring can take place any time during operation under the following conditions:

- A standardization must have been carried out
- The conditions for the first reference point must be fulfilled

The following commands must be carried out

No.	Commands to carry out
1	Set conditions corresponding to the first reference point (weight, pressure, torque etc.).
2	Enter command number 1 ($0010) in configuration word 8. The converter determines the first reference point of the standardization line.
3	Poll configuration word 12 until the acknowledgement ($xx1x) is received after approx. 1 s.
4	Read the value from configuration word 4+5. This contains the raw value for the first reference point.
5	Write the calibrated raw value in configuration words 4+5.
6	Write the standardized value corresponding to the calibrated raw value to configuration words 6+7.
7	Enter command number 5 ($0050) in configuration word 8. The values in configuration words 4+5 and 6+7 are used as the first reference point for the standardization line and the new offset value for the standardization line is calculated without changing the gradient (d_{new} - see graph above). Then the conversion procedure and value output are started.
8	Poll configuration word 12 until the acknowledgement ($xx5x) is received after <100 ms.

Carrying Out Taring

If the process requires that taring be carried out, it must be done each time the system boots.

The following tips will help you to do this correctly

- For frequently changing process conditions, follow the instructions given in the table above.
- For infrequently changing process conditions, the new raw value for the lower limit point can be saved and the procedure listed above can be started at command 5, "Write calibrated raw value in configuration words 4+5".
  Another possibility is to change the raw value for the upper limit according to the change to the lower limit and then use the new raw value for standardization when booting.

10.5 AI294

10.5.1 Technical Data

Module ID	AI294
General Information	
Model Number	7AI294.7
Short Description	2003 analog input module, 4 inputs, potentiometer evaluation, 13 bit, screw-in module
C-UL-US Listed	Yes
B&R ID Code	$22
Slot	AF101 adapter module, CP interface
Static Characteristics	
Module Type	B&R 2003 screw-in module
Input Type	Single ended input in the range from 0 to U_{pot}
Number of Inputs	4
Affects of Cable Length	The shielded, twisted pair cable (3x1) should be as short as possible and run separately to the sensor (isolated from load circuit) without intermediate terminals
Shielding	
Module Side	Using stress relief clamp on the module
Potentiometer Side	Direct connection to grounded potentiometer housing (use short connection lines)
Maximum Continuous Overload (without damage)	+30 V for all connections except GND
Output of the Digital Value during Overload	
Open Connection on U_{pot}	Value = $8000
Open Connection on GND	Value = $7FFF
Open Connection on slider or U_{pot} and GND	Value = $7FFF
Recognition Time for Open Slider	<3 s
Software Evaluation	Evaluation using configuration word 12 (module status)
Digital Converter Resolution	13 Bit

Module ID	AI294
Quantization; LSB Value (13 bit)	0.55 mV
Data Format Delivered to the Application Program	2s complement (16 bit representation)
Slider Voltage GND	Value = $0000
Slider Voltage U_{pot}	Value = $7FFF
Potentiometer Feed Voltage U_{pot}	+4.5 V ±3 % at 40 mA
Short Circuit and Overload Protection	Yes
Input Current	<0.2 μA
Measurement Sensor	0.5 to 10 kΩ, Potentiometer
Measurement Range	0 V to U_{pot}
Conversion Method	Sample & Hold
Conversion Time	4 ms for all channels, also with active comparator
Analog Input Measurement Error	
Maximum Error at 25 °C	±0.04 % of full range
Temperature Coefficient	±20 ppm/°C of full range
Maximum Error over Complete Temperature Range	±0.07 % of full range
Sensor Type	Isolated
Protection	RC protection
Internal Power Consumption	Max. 0.5 W
Dynamic Characteristics	
Application Scan Time	4 - 100 ms
Scan Repeat Time for Comparator Input	125 μs
Response Time 1 LSB	Approx. 200 μs
Input Filter Characteristics	
Attenuation	>40 dB/Dec
Transition / Cut-off Frequency	5 kHz
Signal Delay because of Input Filter under Following Conditions	3 to 10 μs
Slider Speed	1 m/s
Pot	2 kΩ
Path Length	100 mm
Operating Characteristics	
Isolation Voltage under Normal Operating Conditions between Channel and Bus	No electrical isolation
Missing Codes	Yes, because compensation and modification of the convertor resolutions takes place using a standard format
Non-linearity	±0.06 % of full range
Mechanical Properties	
Dimensions	B&R 2003 screw-in module

10.5.2 General Information

The AI294 is a 4 channel analog input module. It is used to evaluate potentiometer displacement gauges.

10.5.3 Special Functions

A threshold value can be defined for any channel. A bit in configuration word 12 indicates the status.

10.5.4 Input Circuit Diagram
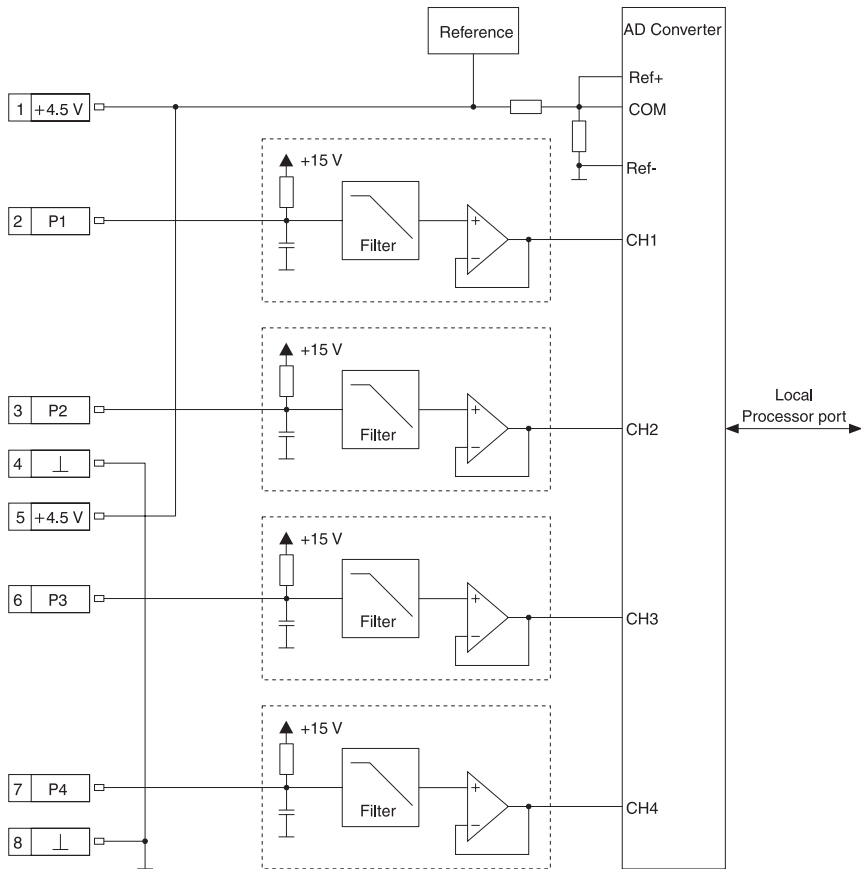
10.5.5 Connection

Terminal Assignments

Connection	Designation	Description
1	+4.5 V	$U_{pot} = +4.5$ V ... Supply for pot 1 and 2
2	P1	Slider potentiometer 1
3	P2	Slider potentiometer 2
4	⊥	GND
5	+4.5 V	$U_{pot} = +4.5$ V ... Supply for pot 3 and 4
6	P3	Slider potentiometer 3
7	P4	Slider potentiometer 4
8	⊥	GND

Wiring a Potentiometer Displacement Gauge

U_{pot}

P1

⊥

Displacement Gauge
Potentiometer

Use twisted cable

Good earth between the cable shielding
and the stress relief clamp

1
2

3
4

Low resistance connection between the shielding
on the potentiometer side and the grounded
potentiometer housing.
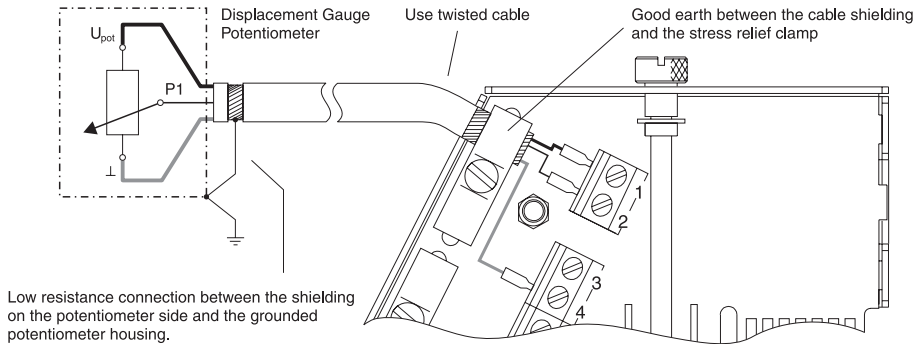
Switching Circuit for All Four Potentiometer Displacement Gauges

Potentiometer 1	Potentiometer 2	Potentiometer 3	Potentiometer 4
1 2 4	1 3 4	5 6 8	5 7 8

10.5.6 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for this module.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	INT16	Analog In	1	●		Analog input value channel 1
Data word 1	INT16	Analog In	2	●		Analog input value channel 2
Data word 2	INT16	Analog In	3	●		Analog input value channel 3
Data word 3	INT16	Analog In	4	●		Analog input value channel 4
Configuration word 8	INT16	Transp. Out	16		●	Switching level for threshold value switch
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

10.5.7 Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

Data cannot be packed on the AI294. Therefore one CAN object is transferred per screw-in module.
If an adapter module AF101 is equipped with a four AI294 modules, the CAN object has the following structure:

Slot	CAN ID [1]	Word 1		Word 2		Word 3		Word 4	
1	542	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
2	543	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
3	544	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
4	545	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H

[1] CAN ID $= 542 + (nd - 1) \times 16 + (ma - 1) \times 4 + (sl - 1)$

nd ..... Node number of the CAN slave $= 1$

ma .... Module address of the AF101 $= 1$

sl .... Slot number of the screw-in module on the AF101 (1 - 4)

> ⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

10.5.8 Description of Data and Configuration Words

Data word 0, 1, 2, 3 (read)

16 bit standardized values for the divider ratio, referring to the potentiometer supply.

Configuration Word 8 (write)

Value of the switching level for the threshold value switch (16 bit, signed).

Configuration Word 12 (read)

Configuration word 12 contains the module status.

	Bit	Description
	12 - 15	x.... Not defined, masked out
	11	0.... Converter values ready 1.... Converter values not yet ready
	10	x.... Not defined, masked out
	9	0.... Potentiometer supply OK 1.... Overload of potentiometer supply
	8	0.... Potentiometer supply OK 1.... Short circuit in the potentiometer supply
	7	Comparator status 0.... if the value < threshold value 1.... if the value ≥ threshold value
	4 - 6	x.... Not defined, masked out
	3	0.... Channel 4: No error 1.... Channel 4: Error present
	2	0.... Channel 3: No error 1.... Channel 3: Error present
	1	0.... Channel 2: No error 1.... Channel 2: Error present
	0	0.... Channel 1: No error 1.... Channel 1: Error present

x	x	x	x		x				x	x	x				
15								8	7						0

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $22
	0 - 7	x.... Not defined, masked out

0	0	1	0	0	0	1	0	x	x	x	x	x	x	x	x
15								8	7						0

Configuration Word 14 (write)

The module is configured using configuration word 14.

	Bit	Description
	15	0.... TPU operation switched off 1.... TPU operation switched on 　　　To be able to use TPU operation, the module must be operated 　　　on the CP Interface.
	11 - 14	0
	10	0.... Operation without threshold value switch 1.... Operation with threshold value switch The time resolution of the comparator is 125 µs. During operation with threshold value switch and TPU operation switched on (Bit 15 = 1), the status of Bit 7 in configuration word 12 is copied to the TPU OUT line. The TPU OUT line can be operated using LTX functions (e.g. LTXdi1()).
	8 - 9	Selection of channels for the threshold value switch 0.... Channel 1 1.... Channel 2 2.... Channel 3 3.... Channel 4
	0 - 7	0

0	0	0	0			0	0	0	0	0	0	0	0

15　　　　　　　8　7　　　　　　0

10.6 AI351

10.6.1 Technical Data

Module ID	AI351	
General Information		
Model Number	7AI351.70	
Short Description	2003 Analog Input Module, 1 input, +/- 10 V or 0 to 20 mA, 12 Bit + sign, screw-in module, Order TB712 terminal block separately!	
C-UL-US Listed	Yes	

B&R ID Code
   Voltage
   Potentiometer Operation
   Current
	$02
	$02
	$03	

Slot	AF101 adapter module, CP interface	
Static Characteristics		
Module Type	B&R 2003 screw-in module	
Number of Inputs	1 differential input	

Input Signal
   Current
   Voltage
	Set using switch
	0 - 20 mA (also ±20 mA)
	±10 V	

Potentiometer Voltage	±9.94 V at max. 10 mA	
Digital Converter Resolution	12 Bit + sign	

Differential Input Resistance
   Voltage
   Current (load)
	20 MΩ
	130 - 200 Ω	

Measurement Precision at 25 °C
   Offset
   Gain
   Linearity Error
   Potentiometer Feed
	Voltage
	Max. ±2.5 mV
	Max. 0.1 % of final value
	Max. 0.05 % of final value
	±9.94 V +2.3 % / -1.7 % max.
		Current
		Max. ±5 μA

Temperature Drift
   ±10 V Input
   0 - 20 mA Input
   Potentiometer Feed
	±0.014 %/°C ±125 μV/°C
	±0.012 %/°C ±0.4 μA/°C
	±0.01 %/°C	

Module ID	AI351

Power Consumption
    Current/Voltage Measurement
    Potentiometer Operation
	Max. 0.3 W
	Max. 0.7 W

Dynamic Characteristics	

Input Filter
    Cut-off Frequency
    Attenuation
	1 kHz
	60 dB

Operating Characteristics	
Electrical Isolation Input - PCC	No
Mechanical Characteristics	
Dimensions	B&R 2003 screw-in module

10.6.2 Input Circuit Diagram
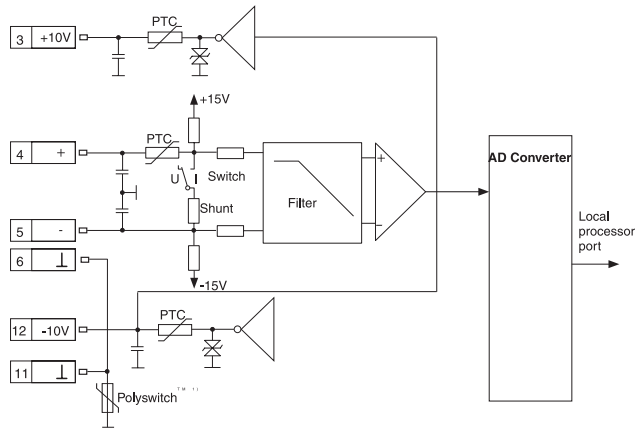
[1] Polyswitch™ is a registered trademark of RAYCHEM.

This is a polymer PTC protective element that functions as overload and short circuit protection. If an overload or short circuit occurs, the protective element becomes highly resistive and breaks the flow of current. In order to activate the input again, the external supply must be turned off and the error (overload or short circuit) must be removed. After a reset time > 10 seconds, the protective element returns to the conductive state.

10.6.3 Connections

TB712
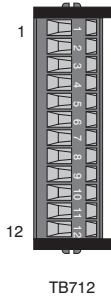
Pin	Assignment
1	+10 V potentiometer supply
2	+ differental input
3	- differential input
4	GND
5	GND
6	-10 V potentiometer supply
7	Shield
8	n. c.
9	n. c.
10	n. c.
11	n. c.
12	n. c.

10.6.4 Connection Examples

The analog input module AI351 can be used for voltage and current measurement and also for potentiometer operation. The following connection diagram shows how the module is to be wired.

Voltage and Current Measurement

Mode Switch
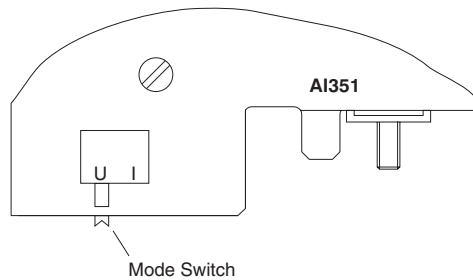
Mode Switch

The module can be used for voltage or current measurement. The selection is made with the mode switch on the back of the module. Place the switch in the respective position for the desired measurement:

U ..... Voltage measurement
I ..... Current measurement

Module Wiring

Measurement signal

+ Diff. Input
- Diff. Input

Shield

U

+

−
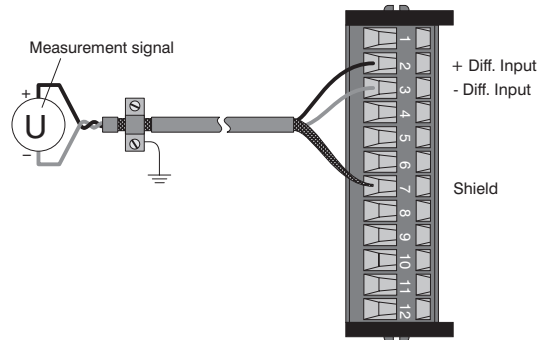
Potentiometer Operation

Mode Switch

⚠ **For potentiometer operation: Set the switch in the U position!**

Module Wiring

+10 V

+

-10 V

Potentiometer (>2 kΩ)

+10 V Pot Supply
+ Diff. Input
- Diff. Input
GND

-10 V Pot Supply
Shield

10.6.5 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for this module.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	INT16	Analog In	1	●		Analog input value
Configuration word 8	INT16	Transp. Out	16		●	Switching level for threshold value switch
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

10.6.6 Access Using CAN IDs

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer. Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

The AI351 can transfer data in packed or unpacked format. The CAN objects look like this:

CAN ID Packed

Only one CAN object is returned.

CAN ID [1]	Slot 1		Slot 2		Slot 3		Slot 4	
542	ScrM 1L	ScrM 1H	ScrM 2L	ScrM 2H	ScrM 3L	ScrM 3H	ScrM 4L	ScrM 4H
543	free							
544	free							
545	free							

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4

  nd ..... Node number of the CAN slave = 1

  ma .... Module address of the AF101 = 1

CAN ID Unpacked

Four CAN objects are returned.

Slot	CAN ID [1]	Word 1		Word 2	Word 3	Word 4
1	542	ScrM 1L	ScrM 1H	Not used (2 byte objects)		
2	543	ScrM 2L	ScrM 2H	Not used (2 byte objects)		
3	544	ScrM 3L	ScrM 3H	Not used (2 byte objects)		
4	545	ScrM 4L	ScrM 4H	Not used (2 byte objects)		

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address of the AF101 = 1

sl .... Slot number of the screw-in module on the AF101 (1 - 4)

> ⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

10.6.7 Description of Data and Configuration Words

Data Word 0 (read)

16 bit standardized value representing either voltage or current. The selection is made using a switch on the screw-in module.
When used as a potentiometer, the voltage value is read according to the potentiometer position.

Configuration Word 8 (write)

Value of the switching level for the threshold value switch (16 bit, signed).

Configuration Word 12 (read)

Configuration word 12 contains the module status.

	Bit	Description
	12 - 15	x....Not defined, masked out
	11	0....Converter value ready 1....Wandlerwert noch nicht bereit
	8 - 10	x....Not defined, masked out
	7	Comparator status 0....if the value < threshold value 1....if the value ≥ threshold value
	2 - 6	x....Not defined, masked out
	1	0....Input signal for current 1....Input signal for voltage
	0	0....Voltage/current measurement 1....Potentiometer operation

x	x	x	x		x	x	x		x	x	x	x	x		

15   8 7   0

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $02.......Voltage measurement or potentiometer operation Module code = $03.......Current measurement
	0 - 7	x....Not defined, masked out

0	0	0	0	0	0	1		x	x	x	x	x	x	x	x

15   8 7   0

Configuration Word 14 (write)

The module is configured using configuration word 14.

	Bit	Description
	15	0.... TPU operation switched off 1.... TPU operation switched on To be able to use TPU operation, the module must be operated on the CP Interface.
	11 - 14	0
	10	0.... Operation without threshold value switch 1.... Operation with threshold value switch The time resolution of the comparator is 250 µs. During operation with threshold value switch and TPU operation switched on (Bit 15 = 1), the status of Bit 7 in configuration word 12 is copied to the TPU OUT line. The TPU OUT line can be operated using LTX functions (e.g. LTXdi1()).
	9	0
	8	Output format for potentiometer operation 0...... -32767 to +32767 1.............. 0 to +32767
	1 - 7	0
	0	0.... Voltage/current measurement 1.... Potentiometer operation

```
 0  0  0  0     0     0  0  0  0  0  0  0  0
15              8  7                    0
```

10.7 AI354

10.7.1 Technical Data

Module ID	AI354
General Information	
Model Number	7AI354.70
Short Description	2003 Analog Input Module, 4 inputs, +/- 10 V, 12 Bit + sign, screw-in module, Order TB712 terminal block separately!
C-UL-US Listed	Yes
B&R ID Code	$04
Slot	AF101 adapter module, CP interface
Static Characteristics	
Module Type	B&R 2003 screw-in module
Number of Inputs	4 differential inputs
Input Signal	±10 V
Digital Converter Resolution	12 Bit + sign
Differential Input Resistance	20 MΩ

Measurement Precision at 25 °C
    Offset
    Gain
    Linearity Error
	Max. ±2.5 mV
	Max. 0.1 % of final value
	Max. 0.1 % of final value

Temperature Drift	$\pm 0.02 \, \% / °C \pm 125 \, \mu V / °C$
Power Consumption	Max. 0.5 W
Dynamic Characteristics	

Input Filter
    Cut-off Frequency
    Attenuation
	225 Hz
	60 dB

Operating Characteristics	

Electrical Isolation
    Input - PCC
    Input - Input
	No
	No

Mechanical Characteristics	
Dimensions	B&R 2003 screw-in module

10.7.2 Input Circuit Diagram
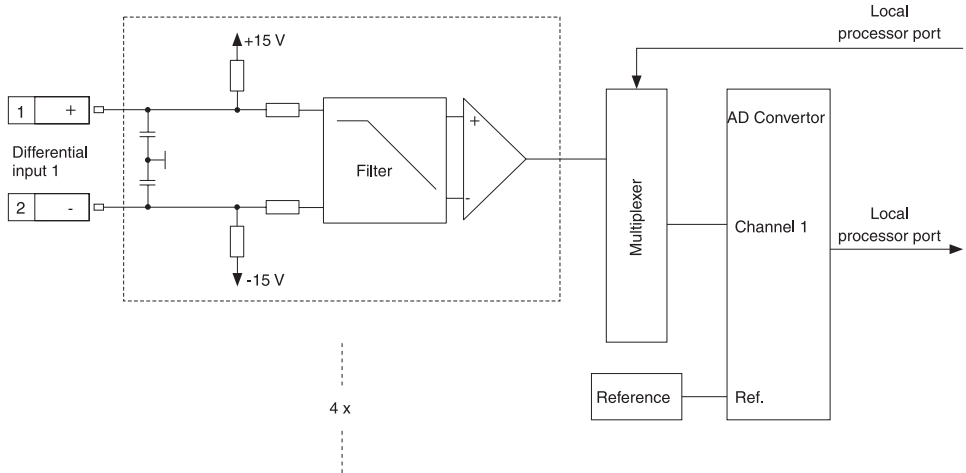
10.7.3 Connections
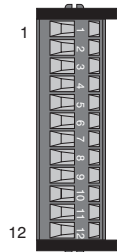
TB712

Pin	Assignment
1	+ differential input 1
2	- differential input 1
3	Shield
4	+ differential input 2
5	- differential input 2
6	Shield
7	+ differential input 3
8	- differential input 3
9	Shield
10	+ differential input 4
11	- differential input 4
12	Shield

10.7.4 Connection Example

Measurement signal

+ Diff. Input 1
- Diff. Input 1
Shield
+ Diff. Input 2
- Diff. Input 2
Shield

10.7.5 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for this module.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	INT16	Analog In	1	●		Analog input value channel 1
Data word 1	INT16	Analog In	2	●		Analog input value channel 2
Data word 2	INT16	Analog In	3	●		Analog input value channel 3
Data word 3	INT16	Analog In	4	●		Analog input value channel 4
Configuration word 8	INT16	Transp. Out	16		●	Switching level for threshold value switch
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

10.7.6 Access Using CAN IDs

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

Data cannot be packed on the AI354. Therefore one CAN object is transferred per screw-in module.
If an adapter module AF101 is equipped with a four AI354 modules, the CAN object has the following structure:

Slot	CAN ID [1]	Word 1		Word 2		Word 3		Word 4	
1	542	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
2	543	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
3	544	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
4	545	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

  nd ..... Node number of the CAN slave = 1

  ma .... Module address of the AF101 = 1

  sl .... Slot number of the screw-in module on the AF101 (1 - 4)

> ⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

10.7.7 Description of Data and Configuration Words

Data word 0, 1, 2, 3 (read)

16 bit standardized voltage value.

Configuration Word 8 (write)

Value of the switching level for the threshold value switch (16 bit, signed).

Configuration Word 12 (read)

Configuration word 12 contains the module status.

	Bit	Description
	12 - 15	x....Not defined, masked out
	11	0....Converter values ready 1....Converter values not yet ready
	8 - 10	x....Not defined, masked out
	7	Comparator status 0.... if the value < threshold value 1.... if the value ≥ threshold value
	4 - 6	x....Not defined, masked out
	3	0....Channel 4: No error 1....Channel 4: Error present
	2	0....Channel 3: No error 1....Channel 3: Error present
	1	0....Channel 2: No error 1....Channel 2: Error present
	0	0....Channel 1: No error 1....Channel 1: Error present

x	x	x	x		x	x	x		x	x	x				

15   8 7   0

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $04
	0 - 7	x ....Not defined, masked out

0	0	0	0	0	1	0	0	x	x	x	x	x	x	x	x

15   8 7   0

Configuration Word 14 (write)

The module is configured using configuration word 14.

	Bit	Description

	15	0.... TPU operation switched off
		1.... TPU operation switched on
		    To be able to use TPU operation, the module must be operated
		    on the CP Interface.

	11 - 14	0
	10	0.... Operation without threshold value switch 1.... Operation with threshold value switch The time resolution of the comparator is 375 µs. During operation with threshold value switch and TPU operation switched on (Bit 15 = 1), the status of Bit 7 in configuration word 12 is copied to the TPU OUT line. The TPU OUT line can be operated using LTX functions (e.g. LTXdi1()).
	8 - 9	Selection of channels for the threshold value switch 0.... Channel 1 1.... Channel 2 2.... Channel 3 3.... Channel 4
	0 - 7	0

0	0	0	0				0	0	0	0	0	0	0	0

15   8 7   0

10.8 AI774

10.8.1 Technical Data

Module ID	AI774
General Information	
Model Number	7AI774.70
Short Description	2003 Analog Input Module, 4 inputs, 0 to 20 mA, 12 Bit, screw-in module, Order TB712 terminal block separately!
C-UL-US Listed	Yes
B&R ID Code	$06
Slot	AF101 adapter module, CP interface
Static Characteristics	
Module Type	B&R 2003 screw-in module
Number of Inputs	4 differential inputs
Input Signal	0 - 20 mA (also ±20 mA)
Digital Converter Resolution	12 Bit
Differential Input Resistance (load)	130 - 200 Ω
Measurement Precision at 25 °C 　Offset 　Gain 　Linearity Error	 Max. $\pm 5\,\mu A$ Max. 0.05 % of final value Max. 0.05 % of final value
Temperature Drift	$\pm 0.012\,\%/°C\ \pm 0.4\,\mu A/°C$
Power Consumption	Max. 0.4 W
Dynamic Characteristics	
Input Filter 　Cut-off Frequency 　Attenuation	 225 Hz 60 dB
Operating Characteristics	
Electrical Isolation 　Input - PCC 　Input - Input	 No No
Mechanical Characteristics	
Dimensions	B&R 2003 screw-in module

10.8.2 Input Circuit Diagram
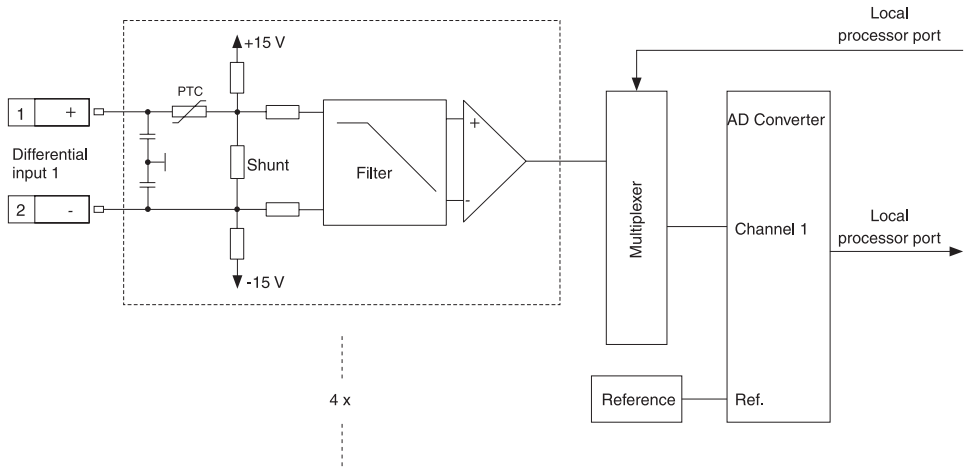
10.8.3 Connections
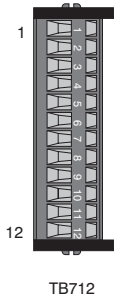
TB712

Pin	Assignment
1	+ differential input 1
2	- differential input 1
3	Shield
4	+ differential input 2
5	- differential input 2
6	Shield
7	+ differential input 3
8	- differential input 3
9	Shield
10	+ differential input 4
11	- differential input 4
12	Shield

10.8.4 Connection Example

Measurement signal

+ Diff. Input 1
- Diff. Input 1
Shield
+ Diff. Input 2
- Diff. Input 2
Shield

10.8.5 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for this module.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	INT16	Analog In	1	●		Analog input value channel 1
Data word 1	INT16	Analog In	2	●		Analog input value channel 2
Data word 2	INT16	Analog In	3	●		Analog input value channel 3
Data word 3	INT16	Analog In	4	●		Analog input value channel 4
Configuration word 8	INT16	Transp. Out	16		●	Switching level for threshold value switch
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

10.8.6 Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

Data cannot be packed on the AI774. Therefore one CAN object is transferred per screw-in module.
If an adapter module AF101 is equipped with a four AI774 modules, the CAN object has the following structure:

Slot	CAN ID [1]	Word 1		Word 2		Word 3		Word 4	
1	542	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
2	543	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
3	544	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
4	545	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

   nd ..... Node number of the CAN slave = 1

   ma .... Module address of the AF101 = 1

   sl .... Slot number of the screw-in module on the AF101 (1 - 4)

> ⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

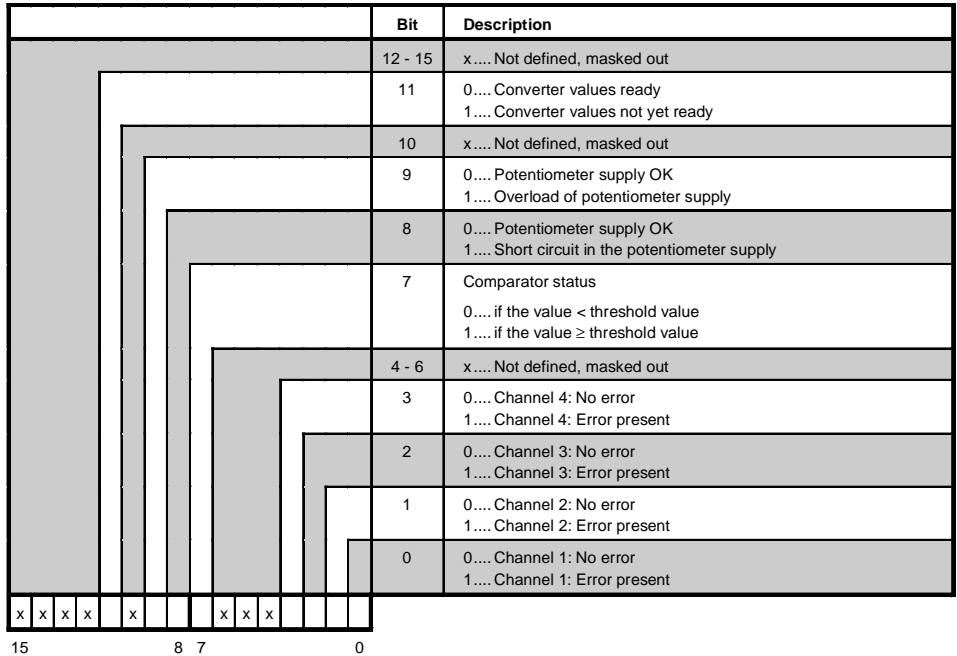### 10.8.7 Description of Data and Configuration Words

Data word 0, 1, 2, 3 (read)

16 bit standardized current value.

Configuration Word 8 (write)

Value of the switching level for the threshold value switch (16 bit, signed).

Configuration Word 12 (read)

Configuration word 12 contains the module status.

	Bit	Description
	12 - 15	x .... Not defined, masked out
	11	0.... Converter values ready 1.... Converter values not yet ready
	8 - 10	x .... Not defined, masked out
	7	Comparator status 0.... if the value < threshold value 1.... if the value ≥ threshold value
	4 - 6	x .... Not defined, masked out
	3	0.... Channel 4: No error 1.... Channel 4: Error present
	2	0.... Channel 3: No error 1.... Channel 3: Error present
	1	0.... Channel 2: No error 1.... Channel 2: Error present
	0	0.... Channel 1: No error 1.... Channel 1: Error present

x	x	x	x		x	x	x		x	x	x				
15							8	7							0

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $06
	0 - 7	x .... Not defined, masked out

0	0	0	0	0	1	1	0	x	x	x	x	x	x	x	x
15							8	7							0

Configuration Word 14 (write)

The module is configured using configuration word 14.

	Bit	Description

	15	0.... TPU operation switched off
		1.... TPU operation switched on
		      To be able to use TPU operation, the module must be operated
		      on the CP Interface.

	11 - 14	0
	10	0.... Operation without threshold value switch 1.... Operation with threshold value switch The time resolution of the comparator is 375 µs. During operation with threshold value switch and TPU operation switched on (Bit 15 = 1), the status of Bit 7 in configuration word 12 is copied to the TPU OUT line. The TPU OUT line can be operated using LTX functions (e.g. LTXdi1()).
	8 - 9	Selection of channels for the threshold value switch 0.... Channel 1 1.... Channel 2 2.... Channel 3 3.... Channel 4
	0 - 7	0

0	0	0	0				0	0	0	0	0	0	0	0

15   8 7   0

11 ANALOG OUTPUT MODULES

11.1 GENERAL INFORMATION

Analog output modules convert PCC internal number values into voltages or currents. The numbers to be converted must be in 16 bit 2s complement. The conversion takes place independent of the resolution of the output module used.

Each analog output module is assigned a status LED on the analog adapter module or CP interface. This LED indicates that the output module is working.

11.2 OVERVIEW

Module	AO352
Number of Outputs	2
Output Signal	±10 V or 0 - 20 mA
Digital Converter Resolution	12 Bit

11.3 PROGRAMMING

The analog outputs are accessed directly in the application program using a variable name. The relationship between the output channel for a certain module and the variable name is defined in the variable declaration. The declaration is made identically for each programming language using a table editor.

11.4 AO352

11.4.1 Technical Data

Module ID	AO352	
General Information		
Model Number	7AO352.70	
Short Description	2003 Analog Output Module, 2 outputs, +/- 10 V or 0 -20 mA, 12 Bit, screw-in module, Order TB712 terminal block separately!	
C-UL-US Listed	Yes	
B&R ID Code	$0E	
Slot	AF101 adapter module, CP interface	
Static Characteristics		
Module Type	B&R 2003 screw-in module	
Number of Outputs	2	

Output Signal
    Current
    Voltage
	Can be set for each channel using a switch
	0 - 20 mA
	±10 V	

Digital Converter Resolution	12 Bit	
Short Circuit Protection	Yes	

Precision at 25 °C
    Offset
    Gain
    Linearity Error
	Voltage
	Max. ±5.2 mV
	Max. 0.3 %
	Max. ±0.13 % of final value
		Current
		Max. ±5.3 μA
		Max. ±0.06 %
		Max. ±0.13 % of final value

Power Consumption	Max. 1.2 W	
Current Output		
Load	Max. 400 Ω	
LSB Value (12 Bit)	5.16 μA ±2.4 % / LSB	
Temperature Drift	±122 ppm / °C ±4 μA / °C	

Module ID	AO352
Voltage Output	
Load	Max. 10 mA
LSB Value (12 Bit)	5.15 mV ±0.8 % / LSB
Temperature Drift	±60 ppm / °C ±1.4 mV / °C
Operating Characteristics	

Electrical Isolation
    Output - PCC
    Output - Output
	
	No
	No

Mechanical Characteristics	
Dimensions	B&R 2003 screw-in module

11.4.2 Output Circuit Diagram
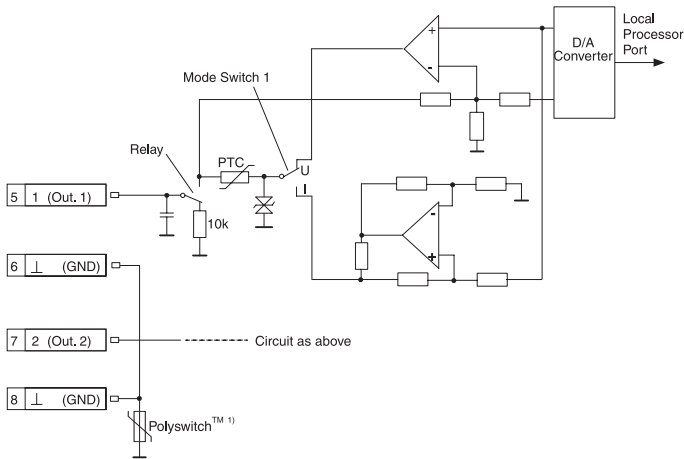
[1] Polyswitch™ is a registered trademark of RAYCHEM.

This is a polymer PTC protective element that functions as overload and short circuit protection. If an overload or short circuit occurs, the protective element becomes highly resistive and breaks the flow of current. In order to activate the output again, the external supply must be turned off and the error (overload or short circuit) must be removed. After a reset time > 10 seconds, the protective element returns to the conductive state.

11.4.3 Connections
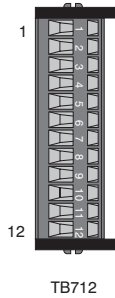
Pin	Assignment
1	Output 1
2	GND
3	Shield
4	Output 2
5	GND
6	Shield
7	n. c.
8	n. c.
9	n. c.
10	n. c.
11	n. c.
12	n. c.

TB712

11.4.4 Connection Example

The analog output module AO352 has two analog outputs which may be used as either voltage or current output. Mixed operation is also possible.

Mode Switch

Mode Switch Output 2

Mode Switch Output 1
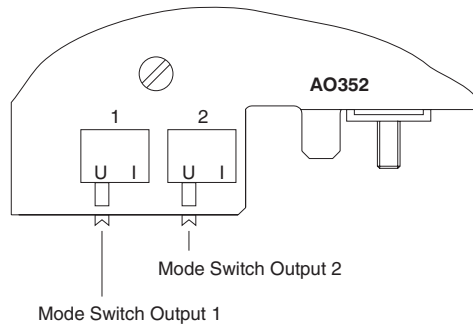
An output can be used as either voltage or current output. The selection is made with the respective mode switch on the back of the module. Place the switch in the respective position for the desired signal:

U ..... Voltage output
I ..... Current output

Module Wiring

In the following example, output 1 is operated as voltage output and output 2 as current output.

11.4.5 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for this module.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	INT16	Analog Out	1		●	Analog output value channel 1
Data word 1	INT16	Analog Out	2		●	Analog output value channel 2
Data word 2	INT16	Analog Out	3		●	Analog output value channel 3 (switching channel for channel 1)
Data word 3	INT16	Analog Out	4		●	Analog output value channel 4 (switching channel for channel 2)
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration (optional)

11.4.6 Access Using CAN IDs

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

Data cannot be packed on the AO352. Therefore one CAN object is transferred per screw-in module.
If an adapter module AF101 is equipped with a four AO352 modules, the CAN object has the following structure:

Slot	CAN ID [1]	Word 1		Word 2		Word 3	Word 4
1	1054	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Not used (4 byte objects)	
2	1055	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Not used (4 byte objects)	
3	1056	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Not used (4 byte objects)	
4	1057	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Not used (4 byte objects)	

[1] CAN ID = 1054 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address of the AF101 = 1

sl .... Slot number of the screw-in module on the AF101 (1 - 4)

> ⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

11.4.7 Description of Data and Configuration Words

Data Words 0 and 1 (write)

The 16 bit standardized values for voltage or current are written to the module output channel.

Data Words 2 and 3 (write)

These data words are only used if TPU operation is switched on (see Configuration Word 14). The module must be operated on the CP interface.
For active change over operation, these data words are used to define the standardized 16 bit values for voltage or current for logical channels 3 and 4. Depending on the status of the TPU-IN line, either the value from channel 1 or 3 is written to physical channel 1 or the value from channel 2 or 4 is written to physical channel 2 of the module.

Level of the TPU-IN connection	Physical channel 1	Physical channel 2
1	Log. channel 1	Log. channel 2
0	Log. channel 3	Log. channel 4

Configuration Word 12 (read)

Configuration word 12 contains the module status.

	Bit	Description
	2 - 15	x .... Not defined, masked out
	1	0.... Channel 1: Current output 1.... Channel 1: Voltage output
	0	0.... Channel 2: Current output 1.... Channel 2: Voltage output

x	x	x	x	x	x	x	x	x	x	x	x	x	x		
15							8	7					0		

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $0E
	0 - 7	x .... Not defined, masked out

0	0	0	0	1	1	1	0	x	x	x	x	x	x	x	x
15							8	7						0	

Configuration Word 14 (write)

The module is configured using configuration word 14.

	Bit	Description
	15	0.... TPU operation switched off 1.... TPU operation switched on To be able to use TPU operation, the module must be operated on the CP Interface.
	13 - 14	0
	12	Only if bit 15 = 1 0.... Normal operation 1.... Switching operation In ^this type of operation, logical channels 1 and 3 or 2 and 4 are switched between depending on the TPU-IN line (also see data words 2 and 3). The TPU IN line can be operated using LTX functions (e.g. LTXdo0()). Before switching, channels 3 and 4 must be set to a correct value by writing to data words 2 and 3. If this does not happen, the data from channel 1 is copied to channel 3 or the data from channel 2 is copied to channel 4 before switching the first time. The data is no longer copied when the channel is switched until the next time the PCC is turned on.
	0 - 11	0

	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0
15								8	7						0		

12 TEMPERATURE MODULES

12.1 GENERAL INFORMATION

Temperature measurement values are converted into number values which can be processed in the PCC using temperature modules.

In the PCC, the number values are always in 16 bit 2s complement regardless of the resolution. In this way, the resolution (number of steps) of the temperature module does not have to be considered when creating the application program.

For temperature measurements, the temperature module returns the measured value in 0.1 ° steps. That means a result of 750 corresponds to 75.0 °C. The data format 0.1 °C is supported by all temperature modules as standard. Additionally, a higher resolution can also be set for some temperature modules.

Each temperature module is assigned a status LED on the analog adapter module or CP interface. This LED indicates that the temperature module is working.

12.2 OVERVIEW

Module	AT324	AT352	AT664
Number of Channels	4	2	4
Measurement Range	-200 to +850 °C	-200 to +850 °C	-270 to +1372 °C
Sensor	KTY10-6 KTY84-130 PT100 PT1000	PT100	FeCuNi, Type J NiCrNi, Type K PtRhPt, Type S
Raw Value Evaluation	Yes	No	Yes
Digital Converter Resolution	16 Bit	16 Bit	16 Bit

12.3 PROGRAMMING

The analog data points are accessed directly in the application program using a variable name. The relationship between the channel for a certain module and the variable name is defined in the variable declaration. The declaration is made identically for each programming language using a table editor.

12.4 AT324

12.4.1 General Information

The screw-in module AT324 is equipped with four input channels. It is used to evaluate measured values for PT100, PT1000, KTY10-6 and KTY84-130 temperature sensors.
The module is installed on the adapter module or on the CP interface.

12.4.2 Technical Data

Module ID	AT324
General Information	
Model Number	7AT324.70
Short Description	2003 Analog Input Module, 4 temperature inputs (2-line connection), KTY10 -50 to +150 degrees C, KTY84 -40 to +300 degrees C, PT100 -200 to +850 degrees C, PT1000 -200 to +850 degrees C, screw-in module, Order terminal block TB712 separately!
C-UL-US Listed	in preparation
B&R ID Code	$3E
Slot	AF101 adapter module, CP interface
Static Characteristics	
Module Type	B&R 2003 screw-in module
Input Type	Resistance measurement using 2-line procedure with constant current feed
Number of Inputs	4
Sensor KTY10-6 KTY84-130 PT100 PT1000	-50 °C to +150 °C -40 °C to +300 °C -200 °C to +850 °C -200 °C to +850 °C
Wiring	2-line
Conversion Method	Sigma Delta
Digital Converter Resolution	16 Bit

Module ID	AT324

Input Amplification
    G = 1
    G = 2
	Can be set for each channel using software
	KTY10-6, KTY84-130, PT1000
	PT100

Reference	5 kΩ ±0.1 %
Measurement Current	200 μA ±5.22 %

Resistance Measurement Range
    at G = 1
    at G = 2
	1 to 4995 Ω
	1 to 2497.5 Ω

Conversion Time per Channel
    Input Amplification
        Uniform
        Different
	
	60 ms
	190 ms

Resolution in °C
    KTY10-6
    KTY84-130
    PT100
    PT1000
	1 LSB = 0.01 °C
	1 LSB = 0.03 °C
	1 LSB = 0.15 °C
	1 LSB = 0.03 °C

Resolution in Ω
    at G = 1
    at G = 2
	1 LSB = 76.29395 mΩ ±0.1 %
	1 LSB = 38.14697 mΩ ±0.1 %

Data Format	INT16

Standardization
    KTY10-6
    KTY84-130
    PT100
    PT1000
	Can be set per channel
	-50.00 °C to +150.00 °C
	-40.00 °C to +300.00 °C
	-200.0 °C to +850.0 °C
	-200.0 °C to +850.0 °C

Value range for resistance measurement
    at G = 1
    at G = 2
	Can be set per channel
	0.1 Ω to 5000.0 Ω
	0.05 Ω to 5000.0 Ω

Measurement Range Monitoring
    Open Inputs
    Broken Line
    Range Exceeded (neg.) [1]
    Range Exceeded (pos.)
    General Error
	$7FFF
	$7FFF
	$8001
	$7FFF
	$8000

Maximum Error at 25 °C	±0.1 % [2]
Offset Drift	±2.5 mΩ / °C [2]
Gain Drift	±30 ppm / °C [3]
Maximum Error over Entire Temperature Range	±0.2 % [2]
Repeat Precision	±0.01 % [2]

Common Mode Rejection
    DC
    50 Hz
	>90 dB
	>150 dB

Cross-talk between Channels	Typ. 100 dB
Power Consumption	Max. 0.1 W
Dynamic Characteristics	

Input Filter
    Type
    Cut-off Frequency
	Lowpass 1st Order
	150 Hz

Module ID	AT324
Operating Characteristics	

Electrical Isolation
   Input - PCC
   Input - Input
	
	No
	No

Mechanical Characteristics	
Dimensions	B&R 2003 screw-in module

[1] Only when measuring with a temperature sensor.

[2] Referring the measurement range.

[3] Referring the current measurement value.

12.4.3 Input Circuit Diagram
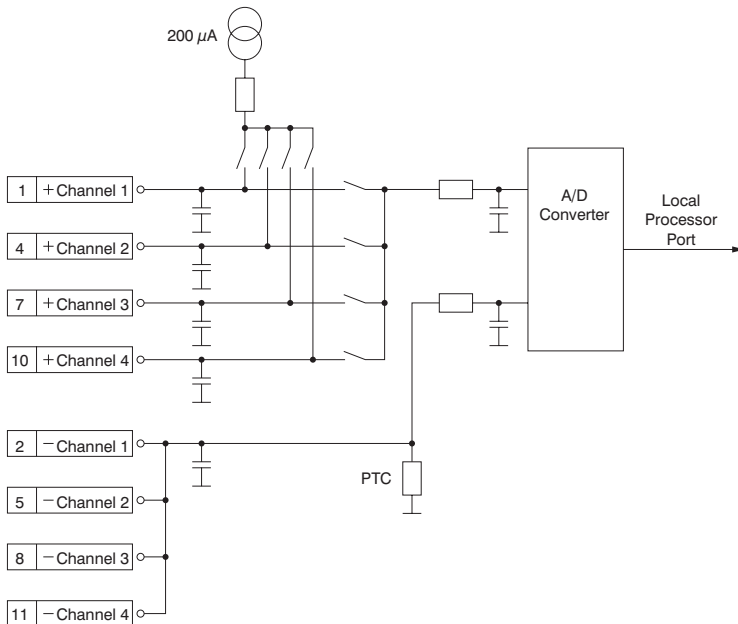
12.4.4 Connections
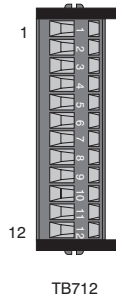
TB712

Assignment	
1	+ channel 1
2	- channel 1
3	Shield
4	+ channel 2
5	- channel 2
6	Shield
7	+ channel 3
8	- channel 3
9	Shield
10	+ channel 4
11	- channel 4
12	Shield

12.4.5 Connection Example

+ Channel 1
- Channel 1
Shield
+ Channel 2
- Channel 2
Shield

12.4.6 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for this module.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	INT16	Analog In	1	●		Analog input value channel 1
Data word 1	INT16	Analog In	2	●		Analog input value channel 2
Data word 2	INT16	Analog In	3	●		Analog input value channel 3
Data word 3	INT16	Analog In	4	●		Analog input value channel 4
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

12.4.7 Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer. Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

Data cannot be packed on the AT324. Therefore one CAN object is transferred per screw-in module. If an adapter module AF101 is equipped with a four AT324 modules, the CAN object has the following structure:

Slot	CAN ID [1]	Word 1		Word 2		Word 3		Word 4	
1	542	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
2	543	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
3	544	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
4	545	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address of the AF101 = 1

sl .... Slot number of the screw-in module on the AF101 (1 - 4)

> ⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

12.4.8 Description of Data and Configuration Words

Data word 0, 1, 2, 3 (read)

16 bit standardized values for temperature or resistance.

Configuration word 12 (read)

Configuration word 12 contains the module status.

	Bit	Description
	12 - 15	x .... Not defined, masked out
	11	0 .... Converter value 1 ready 1 .... Converter value 1 not yet ready
	10	0 .... Converter value 2 ready 1 .... Converter value 2 not yet ready
	9	0 .... Converter value 3 ready 1 .... Converter value 3 not yet ready
	8	0 .... Converter value 4 ready 1 .... Converter value 4 not yet ready
	4 - 7	x .... Not defined, masked out
	0 - 3	Channel currently being converted 1 .... Channel 1 2 .... Channel 2 3 .... Channel 3 4 .... Channel 4

x	x	x	x					x	x	x	x				

15   8 7   0

Configuration word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $3E
	0 - 7	x .... Not defined, masked out

0	0	1	1	1	1	1	0	x	x	x	x	x	x	x	x

15   8 7   0

Configuration Word 14 (write)

The module is configured using configuration word 14.

	Bit	Description
	14 - 15	0
	12 - 13	Select the hardware filter time 0....50 Hz (default) 1....60 Hz
	9 - 11	Definition of the input signal for channel 4 0....KTY10-6 1....KTY84-130 2....PT100 3....PT1000 4....Reserved (value = $8000) 5....Resistance measurement 1 to 4995 Ω 6....Resistance measurement 1 to 2497.5 Ω 7....Input switched off – setting for channels not being used (value = $8000)
	6 - 8	Definition of the input signal for channel 3 0....KTY10-6 1....KTY84-130 2....PT100 3....PT1000 4....Reserved (value = $8000) 5....Resistance measurement 1 to 4995 Ω 6....Resistance measurement 1 to 2497.5 Ω 7....Input switched off – setting for channels not being used (value = $8000)
	3 - 5	Definition of the input signal for channel 2 0....KTY10-6 1....KTY84-130 2....PT100 3....PT1000 4....Reserved (value = $8000) 5....Resistance measurement 1 to 4995 Ω 6....Resistance measurement 1 to 2497.5 Ω 7....Input switched off – setting for channels not being used (value = $8000)
	0 - 2	Definition of the input signal for channel 1 0....KTY10-6 1....KTY84-130 2....PT100 3....PT1000 4....Reserved (value = $8000) 5....Resistance measurement 1 to 4995 Ω 6....Resistance measurement 1 to 2497.5 Ω 7....Input switched off – setting for channels not being used (value = $8000)

```
0 0 | | | | | | | | | | | | | |
15          8 7              0
```

12.5 AT352

12.5.1 Technical Data

Module ID	AT352
General Information	
Model Number	7AT352.70
Short Description	2003 Analog Input Module, 2 inputs, PT100 (3-line connection), -200 to +850 °C, screw-in module, Order TB712 terminal block separately!
C-UL-US Listed	in preparation
B&R ID Code	$1A
Slot	AF101 adapter module, CP interface
Static Characteristics	
Module Type	B&R 2003 screw-in module
Number of Inputs	2

Sensor
  Type
  Connection
  Standard
	PT100
	3-line connection
	IEC/EN 60751

Digital Converter Resolution	16 Bit
Measurement Range	2 ranges can be set
Small Measurement Range Resolution	-200.00 to +327,67 °C 0.01 °C
Large Measurement Range Resolution	-200.0 to +850.0 °C 0.1 °C
Measurement Update	20 or 16.67 ms
Maximum Temperature Measurement Error at 25 °C	$\pm (0.17 + 0.0005 * T_F)$ [°C] T_F ... Sensor Temperature in °C
Maximum Temperature Measurement Error because of drift per °C	$\pm (0.01 + 0.000031 * T_F)$ [°C] T_F ... Sensor Temperature in °C
Conversion of Measurement Values to Temperature Values	Automatic in the module
Measurement Current	2 mA
Power Consumption	Max. 0.4 W

Module ID	AT352
Operating Characteristics	

Electrical Isolation
    Input - PCC
    Input - Input
	
	No
	No

Mechanical Characteristics	
Dimensions	B&R 2003 screw-in module

12.5.2 Input Circuit Diagram

12.5.3 Connections

TB712
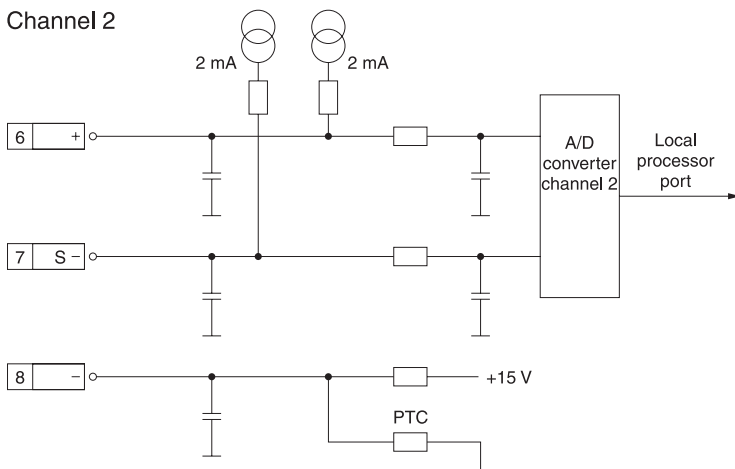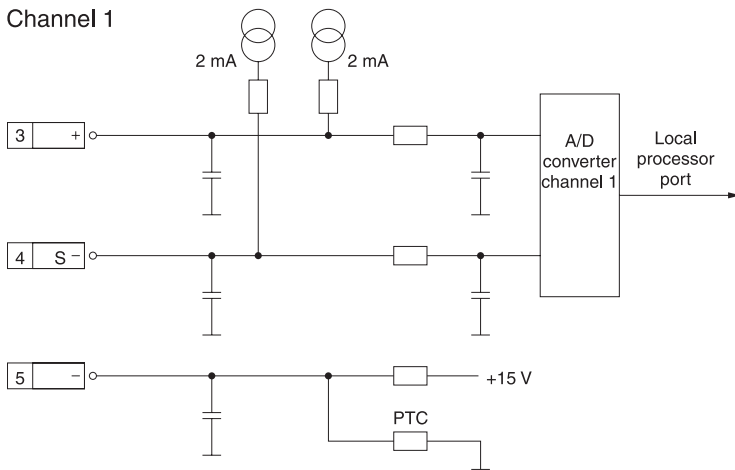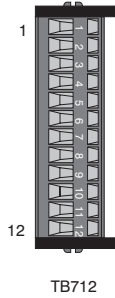
Pin	Assignment
1	+ Sensor 1
2	- Sense 1
3	- Sensor 1
4	Shield
5	+ Sensor 2
6	- Sense 2
7	- Sensor 2
8	Shield
9	- Sense 1
10	- Sense 2
11	n. c.
12	n. c.

12.5.4 Connection Examples

3-Line Connection

Diagram	Connection Example

2-Line Connection

Diagram	Connection Example
1 ○ + Sensor 1 2 ○ - Sense 1 3 ○ - Sensor 1 4 ○ Shield 5 ○ + Sensor 2 6 ○ - Sense 2 7 ○ - Sensor 2 8 ○ Shield 9 ○ - Sense 1 10 ○ - Sense 2	+ Sensor 1 - Sensor 1 + Sensor 1 - Sense 1 - Sensor 1 Shield + Sensor 2 - Sense 2 - Sensor 2 Shield - Sense 1 - Sense 2

12.5.5 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for this module.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	INT16	Analog In	1	●		Temperature channel 1
Data word 1	INT16	Analog In	2	●		Temperature channel 2
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

12.5.6 Access Using CAN IDs

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

Data cannot be packed on the AT352. Therefore one CAN object is transferred per screw-in module.
If an adapter module AF101 is equipped with a four AT352 modules, the CAN object has the following structure:

Slot	CAN ID [1]	Word 1		Word 2		Word 3	Word 4
1	542	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Not used (4 byte objects)	
2	543	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Not used (4 byte objects)	
3	544	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Not used (4 byte objects)	
4	545	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Not used (4 byte objects)	

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address of the AF101 = 1

sl .... Slot number of the screw-in module on the AF101 (1 - 4)

> ⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

12.5.7 Description of Data and Configuration Words

Data Words 0 and 1 (read)

Reads the temperature using the set resolution.

Chan.	Resolution	Configuration word 14	Value range
1	0.1 °C	$0x00	-200.0 °C to +850.0 °C corresponds to -2000 to +8500
	0.01 °C	$0x01	-200.00 °C to +327.67 °C corresponds to -20000 to +32767
2	0.1 °C	$0x00	-200.0 °C to +850.0 °C corresponds to -2000 to +8500
	0.01 °C	$0x02	-200.00 °C to +327.67 °C corresponds to -20000 to +32767

Configuration Word 12 (read)

Configuration word 12 contains the module status.

	Bit	Description
	12 - 15	x .... Not defined, masked out
	11	0.... Converter value 1 ready 1.... Converter value 1 not yet ready
	10	0.... Converter value 2 ready 1.... Converter value 2 not yet ready
	2 - 9	x .... Not defined, masked out
	1	0.... Resolution of channel 2 is 0.1 °C 1.... Resolution of channel 2 is 0.01 °C
	0	0.... Resolution of channel 1 is 0.1 °C 1.... Resolution of channel 1 is 0.01 °C

x	x	x	x			x	x	x	x	x	x	x	x		

15   8 7   0

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $1A
	0 - 7	x .... Not defined, masked out

0	0	0	1	1	0	1	0	x	x	x	x	x	x	x	x

15   8 7   0

Configuration Word 14 (write)

The module is configured using configuration word 14.

	Bit	Description
	13 - 15	0
	12	Selection of the hardware filter time for channel 1 and channel 2 0.... 50 Hz (default) 1.... 60 Hz
	2 - 11	0
	1	Resolution of channel 2 0.... 0.1 °C (default) 1.... 0.01 °C
	0	Resolution of channel 1 0.... 0.1 °C (default) 1.... 0.01 °C

0	0	0		0	0	0	0	0	0	0	0	0	0		

15   8 7   0

12.6 AT664

12.6.1 Technical Data

Module ID	AT664
General Information	
Model Number	7AT664.70
Short Description	2003 Analog Input Module, 4 inputs, thermocouples, -270 to +1372 °C, screw-in module, Order TB712 terminal block separately!
C-UL-US Listed	Yes
B&R ID Code	$0C
Slot	AF101 adapter module, CP interface
Static Characteristics	
Module Type	B&R 2003 screw-in module
Number of Inputs	4
Sensor	Thermocouples
Digital Converter Resolution	16 Bit

Measurement Range
  Sensor Temperature
    FeCuNi: Type J
    NiCrNi: Type K
    PtRhPt: Type S
  Terminal Temperature
  Raw Value
	Depending on sensor type [1]
	-210 to +1200 °C
	-270 to +1372 °C
	-50 to +1768 °C
	-55 to +125 °C
	$\pm 65534\,\mu V$

Resolution
  Sensor Temperature Output
  Sensor Temperature Output
  Raw Value Output
	
	0.1 °C / LSB
	0.1 °C / LSB
	$2\,\mu V$ / LSB

Terminal Temperature Compensation	Yes [1]
Maximum Error at 25 °C	$\pm (50\,\mu V + 0.001 * U_{TH})\ [\mu V]$ U_{TH} ... Thermal Voltage in μV
Maximum Temperature Drift per °C	$\pm (1.7\,\mu V + 0.0001 * U_{TH})\ [\mu V]$ U_{TH} ... Thermal Voltage in μV
Terminal Temperature Error	± 1 °C after 10 min

Module ID	AT664
Conversion of Measurement Values to Temperature Values	Automatic in the module
Power Consumption	Max. 0.4 W
Operating Characteristics	

Electrical Isolation
    Input - PCC
    Input - Input
	No
	No

Mechanical Properties	
Dimensions	B&R 2003 screw-in module

[1] Can be configured using software.

12.6.2 General Information

The AT664 is a 4-channel temperature module. The module is activated by selecting a sensor type in configuration word 14. The selected sensor type is used for all channels.

12.6.3 Operating Modes

Type of operation	Description
Sensor type: J, K, S	Temperature range - see "Technical Data" with terminal temperature compensation
Raw value	Voltage range ±65.534 mV quantized to 2 µV without terminal temperature compensation

12.6.4 Special Functions

- The terminal temperature can be entered for each channel for terminal temperature compensation.
- Unnecessary channels can be switched off, thereby reducing the refresh time. The time saving per channel depends on the hardware filter time:

Hardware filter time	Saving per channel
50 Hz	60 ms
60 Hz	50 ms

12.6.5 Input Circuit Diagram
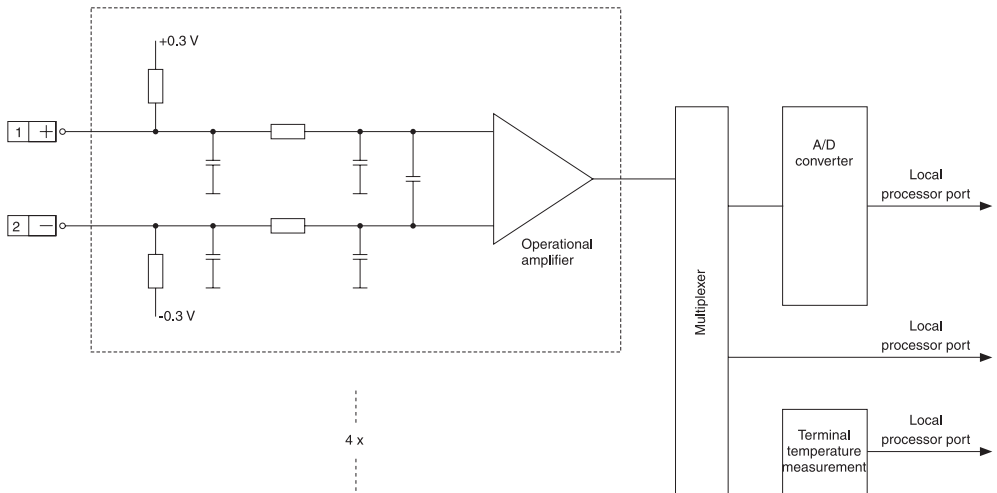
12.6.6 Connections
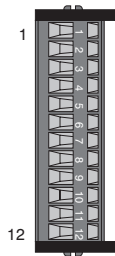
TB712

Pin	Assignment
1	+ channel 1
2	- channel 1
3	Shield
4	+ channel 2
5	- channel 2
6	Shield
7	+ channel 3
8	- channel 3
9	Shield
10	+ channel 4
11	- channel 4
12	Shield

12.6.7 Connection Example

12.6.8 Variable Declaration

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for this module.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	INT16	Analog In	1	●		Temperature channel 1
Data word 1	INT16	Analog In	2	●		Temperature channel 2
Data word 2	INT16	Analog In	3	●		Temperature channel 3
Data word 3	INT16	Analog In	4	●		Temperature channel 4
Configuration word 8	INT16	Transp. In	16	●		Terminal temp. of channel 1 for external conversion
	INT16	Transp. Out	16		●	Terminal temp. of channel 1 for external ref. junction
Configuration word 9	INT16	Transp. In	18	●		Terminal temp. of channel 2 for external conversion
	INT16	Transp. Out	18		●	Terminal temp. of channel 2 for external ref. junction
Configuration word 10	INT16	Transp. In	20	●		Terminal temp. of channel 3 for external conversion
	INT16	Transp. Out	20		●	Terminal temp. of channel 3 for external ref. junction
Configuration word 11	INT16	Transp. In	22	●		Terminal temp. of channel 4 for external conversion
	INT16	Transp. Out	22		●	Terminal temp. of channel 4 for external ref. junction
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

12.6.9 Access Using CAN IDs

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

Data cannot be packed on the AT664. Therefore one CAN object is transferred per screw-in module.
If an adapter module AF101 is equipped with a four AT664 modules, the CAN object has the following structure:

Slot	CAN ID [1]	Word 1		Word 2		Word 3		Word 4	
1	542	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
2	543	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
3	544	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H
4	545	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Chan. 4L	Chan. 4H

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address of the AF101 = 1

sl .... Slot number of the screw-in module on the AF101 (1 - 4)

⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

12.6.10 Description of Data and Configuration Words

Data word 0, 1, 2, 3 (read)

The temperature value or a 16 bit standardized raw value is read depending on the sensor type selected.

Configuration Word 8, 9, 10, 11 (read)

Reads the terminal temperature to externally calculate the temperature from the the raw value.

Configuration Word 8, 9, 10, 11 (write)

The terminal temperature can be set for installation of a reference junction. Operation together with a reference junction is only possible for **all** four channels.
Switching occurs as soon as **one** of the configuration words 8 - 11 is written to.
Switching back occurs by writing $80xx to **one** of the configuration words 8 - 11.

Configuration Word 12 (read)

Configuration word 12 contains the module status.

	Bit	Description
	12 - 15	x .... Not defined, masked out
	11	0 .... Converter value 1 ready 1 .... Converter value 1 not yet ready
	10	0 .... Converter value 2 ready 1 .... Converter value 2 not yet ready
	9	0 .... Converter value 3 ready 1 .... Converter value 3 not yet ready
	8	0 .... Converter value 4 ready 1 .... Converter value 4 not yet ready
	3 - 7	x .... Not defined, masked out
	0 - 2	Channel currently being converted 1 .... Channel 1 2 .... Channel 2 3 .... Channel 3 4 .... Channel 4

x	x	x	x					x	x	x	x	x			

15   8   7   0

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

		Bit	Description
		8 - 15	Module code = $0C
		0 - 7	x .... Not defined, masked out

0	0	0	0	1	1	0	0	x	x	x	x	x	x	x	x

15   8 7   0

Configuration Word 14 (write)

The module is configured using configuration word 14.

		Bit	Description
		14 - 15	0
		12 - 13	Select the hardware filter time 0.... 50 Hz (default) 1.... 60 Hz
		11	0.... Channel 4 is switched on (default) 1.... Channel 4 is switched off
		10	0.... Channel 3 is switched on (default) 1.... Channel 3 is switched off
		9	0.... Channel 2 is switched on (default) 1.... Channel 2 is switched off
		8	0.... Channel 1 is switched on (default) 1.... Channel 1 is switched off
		3 - 7	0
		0 - 2	Selection of the sensor type or the value to be given for all four channels. 0.... No conversion (default) 1.... Sensor type J 2.... Sensor type K 3.... Sensor type S 7.... Raw value without linearization and terminal temperature compensation (Quantization 2 µV with a theoretical measurement range of ±65 mV)

0	0								0	0	0	0	0			

15   8 7   0

13 OTHER MODULES

13.1 GENERAL INFORMATION

The group "Other Modules" includes combination modules and B&R 2003 expansions for the CP476.

13.2 COMBINATION MODULES

The combination module is a combination of digital input and output modules, as well as analog input and output modules. The status of the digital inputs and outputs are indicated with status LEDs.

Overview

Module	CM211	CM411
Digital Inputs		
Number of Inputs	8	3
Nominal Input Voltage	24 VDC	24 VDC
Digital Outputs		
Number of Outputs	8	2
Switching Voltage	24 VDC	24 VDC
Continuous Current	Max. 0.5 A	Max. 0.5 A
Analog Inputs		
Number of Inputs	2	3
Input Signal	±10 V / 0 -20 mA	±10 V
Resolution	12 Bit	16 Bit
Analog Outputs		
Number of Outputs	2	3
Output Signal	±10 V	±10 V
Resolution	12 Bit	16 Bit

13.3 B&R 2003 EXPANSIONS FOR THE CP476

The CP476 now provides another expansion possibility on the far left of the system bus. Special expansion modules can be installed here.

Overview

Module	Description
ME010	2003 expansion for the CP476, 1 PCMCIA slot
ME020	2003 expansion for the CP476, 1 PCMCIA slot, 1 slot interface module inserts

13.4 CM211

13.4.1 Technical Data

Terminal block is not included in the delivery.

Module ID	CM211
General Information	
Model Number	7CM211.7
Short Description	2003 Combination Module, 8 inputs, 24 VDC, 4 ms, sink, 3 one-channel or 2 two-channel counters or 2 incremental encoders, 20 kHz, 8 transistor outputs, 24 VDC, 0.5 A, comparator function, short-circuit protection, 2 inputs, +/- 10 V / 0-20 mA, 12 Bit, 2 outputs, +/- 10 V, 12 Bit, Order TB718 terminal blocks separately!
C-UL-US Listed	in preparation
B&R ID Code	$C3
Module Type	B&R 2003 I/O Module
Amount	
CP430, EX270 EX470, EX770	2
CP470, CP770 CP474, CP476, CP774 EX477, EX777	4
External Voltage Monitoring	Yes (LED: OK), supply voltage >18 V
Electrical Isolation Analog - PCC Digital - PCC Digital - Analog	No No No
Power Consumption	Max. 1.5 W
Analog Inputs	
Input Type	Asymmetric
Number of Inputs	2
Input Signal - Nominal	±10 V / 0 - 20 mA, can be set for each channel with switch

Module ID	CM211	
Maximum Continuous Overload (without damage)	±15 V / ±50 mA	
Digital Converter Resolution	12 Bit	
Data Format Delivered to Application Program	16 Bit 2s complement	
Value Range Voltage +10 V 0 V -10 V Current 20 mA 0 mA	 $7FFF $0000 $8001 $7FFF $0000	
Measurement Range Monitoring Open Inputs Range Exceeded (neg.) Voltage Current Range Exceeded (pos.) General Error	 $7FFF $8001 $0000 $7FFF $8000	
Conversion Method	Successive approximation	
Conversion Time	<4 ms for both channels, the channels are converted cyclically	
Input Impedance in Signal Range for Voltage Input	≥1 MΩ	
Input Impedance in Signal Range for Current Input (load)	95 - 200 Ω	
Measurement Precision at 25 °C	Voltage	Current
Offset Gain	±2.62 mV ±0.2 % [1]	±5.29 μA ±0.2 % [1]
Offset Drift	±2 mV/°C	±5.9 μA/°C
Gain Drift	±65 ppm/°C [2]	±75 ppm/°C [2]
LSB Value (12 Bit)	±2.53 mV ±0.09 mV	±5.09 μA ±0.2 μA
Non-linearity	±2 LSB	
Input Filter	Cut-off Frequency 500 Hz	
Analog Outputs		
Number of Outputs	2	
Output Signal	±10 V	
Load	Max. ±10 mA	
Digital Converter Resolution	12 Bit	
Data Format in Application Program	16 Bit 2s complement	
Value Range +10 V 0 V -10 V	 $7FFF $0000 $8001	
Conversion Time	<4 ms for both channels	
Load Impedance	≥1 kΩ	
Measurement Precision at 25 °C Offset Gain	 ±5.14 mV ±0.2 % [1]	
Offset Drift	±1.2 mV/°C	
Gain Drift	±40 ppm / °C [2]	

Module ID	CM211
LSB Value (12 Bit)	±5.01 mV ±0.13 mV
Non-linearity	±3.5 LSB
Short Circuit Protection	Yes
Digital inputs	
Number of Inputs	8
Type of Inputs	3 x event counters, 3 x period measurement, 3 x gate measurement, 2 x incremental encoder ABR (+24 V), 1 x comparator

Input Voltage
  Minimum
  Nominal
  Maximum
	
	18 VDC
	24 VDC
	30 VDC

Input Current at Nominal Voltage	Approx. 4 mA
Wiring	Sink

Switching Threshold
  LOW Range
  HIGH Range
	
	<5 V
	>15 V

Delay 0 to 1
  with Software Filter
  without Software Filter
	
	<4 ms (default)
	<0.01 ms

Delay 1 to 0
  with Software Filter
  without Software Filter
	
	<4 ms (default)
	<0.01 ms

Status Display	8 Green LEDs

Incremental Encoder Operation
  Signal Form
  Evaluation
  Input Frequency
  Counter Frequency
  Count Size
 	
  Input 1
  Input 2
  Input 3
  Input 4
 	
  Input 5
  Input 6
  Input 7
  Input 8
	
	Square wave pulses
	4-fold, cyclic counter
	20 kHz
	80 kHz
	32 Bit
	
	Reference enable switch 1
	Channel A1
	Channel B1
	Channel R1
	
	Channel A2
	Channel B2
	Channel R2
	Reference enable switch 2

Event Counter Operation
  Signal Form
  Evaluation
  Input Frequency
  Counter Frequency
  Count Size
  Input 2
  Input 3
  Input 5
	
	Square wave pulses
	Each edge, cyclic counter
	20 kHz
	40 kHz
	16 Bit
	Counter 1
	Counter 2
	Counter 3

Period Measurement
  Signal Form
  Evaluation
  Input Frequency
  Counter Frequency - Internal
  Counter Frequency - External
  Count Size
  Input 3
  Input 4
  Input 7
  Input 2
  Input 5
	
	Square wave pulses
	Positive edge - Positive edge
	20 kHz
	16 MHz, 4 MHz, 1 MHz, 250 kHz
	Max. 20 kHz
	16 Bit
	Period channel 1
	Period channel 2
	Period channel 3
	External counter frequency for channels 1 and 2
	External counter frequency for channel 3

Module ID	CM211
Gate Measurement	
Signal Form	Square wave pulses
Evaluation	Positive edge - Negative edge
Input Frequency	10 kHz
Counter Frequency - Internal	16 MHz, 4 MHz, 1 MHz, 250 kHz
Counter Frequency - External	Max. 20 kHz
Count Size	16 Bit
Gate Pause	50 μs
Input 3	Gate channel 1
Input 4	Gate channel 2
Input 7	Gate channel 3
Input 2	External counter frequency for channels 1 and 2
Input 5	External counter frequency for channel 3
Comparator	
Comparator Output	Output 1
Reaction Time	<500 μs
Evaluation	
Incremental Encoder Operation	Actual value comparison of the counter status of incremental encoder 1
Event Counter Operation	Comparison of the counter status of counter 2 (window comparator)
Electrical Isolation	
Input - Input	No
Digital Outputs	
Number and Type of Outputs	8 transistor outputs
Rated Current	Max. 0.5 A
Total Output Current	Max. 4 A
Rated Voltage	24 VDC
Switching Voltage Range	18 - 30 VDC
Leakage Current (0 signal)	12 μA
Wiring	Source
Short circuit protection	Yes
Overload Protection	Yes
Negative Anode Potential when Switching Off Inductive Loads	47 V
Delay 0 to 1	<1.5 ms
Delay 1 to 0	<1.5 ms
Status Display	8 orange LEDs
Electrical Isolation Output - Output	No
Mechanical Characteristics	
Dimensions	B&R 2003 single width

[1] Referring the maximum positive limit.

[2] Referring the current measurement value.

13.4.2 Status LEDs

LED	Description
OK	This orange LED is lit when the external supply voltage for the outputs is within the defined range (>18 VDC).
LED 1 - 8, green	Logical status of the respective digital input.
LED 1 - 8, orange	Control status of the respective digital output.

13.4.3 Input Circuit Diagram
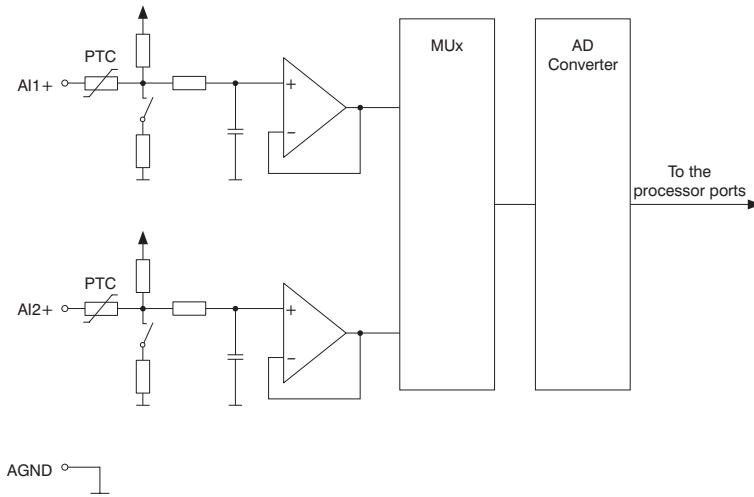
Analog Inputs
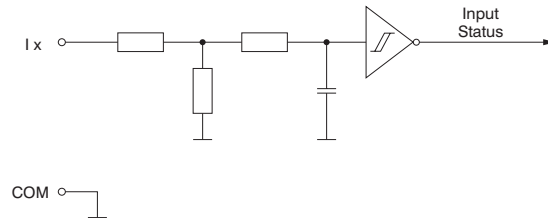
Digital Inputs

13.4.4 Output Circuit Diagram

Analog Outputs
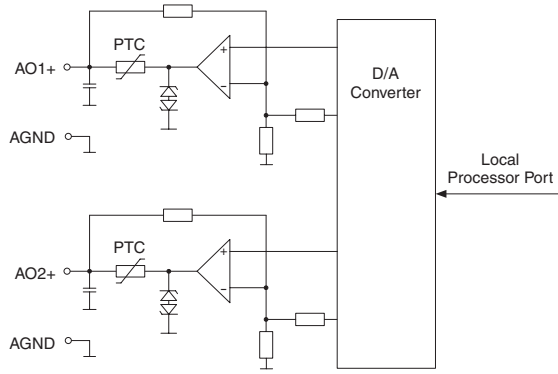
Digital Outputs
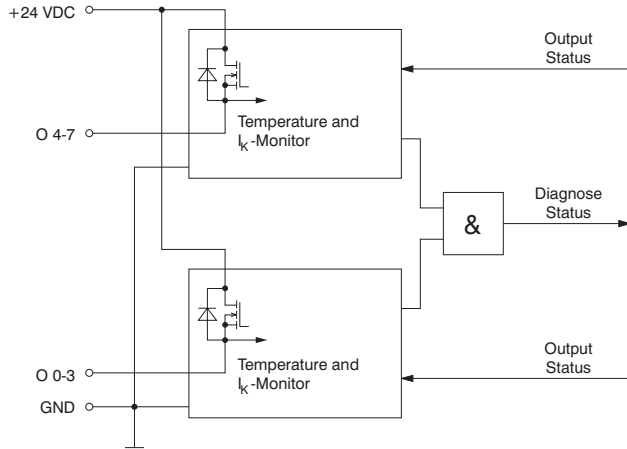
13.4.5 Supply Voltage Monitoring
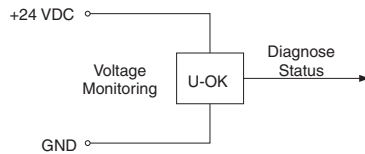
13.4.6 Module Design

General Information

The design of the combination module CM211 corresponds to an adapter module AF101 with four screw-in modules installed and a digital mixed module.

AIN	...	Analog inputs
AOUT	...	Analog outputs
COUNT	...	Counter inputs
DM	...	Digital mixed module
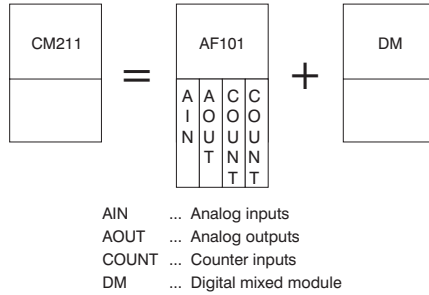
Module Addresses

Because of this special module design, the combination module CM211 requires two module addresses. A CPU, a combination module CM211 and a digital input module DI435 are used in the example shown below. The module address assignments are to be made as shown in the diagram.

Module Address	
1 + 2	3

CPU	CM211	DI435

Variable Declaration

To avoid conflicts in the register, the settings listed below must be used in the variable declaration for the module address and for the slot.
In this case, the module is accessed with module addresses 1 and 2.

CM211					
AF101				**DM**	
Module Address = 1				Module Address = 2	
Slot 1	Slot 2	Slot 3	Slot 4	Slot: Fixed to 1	
AIN	AOUT	Count	Count		
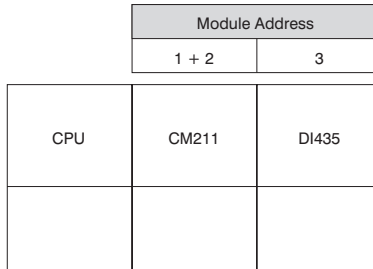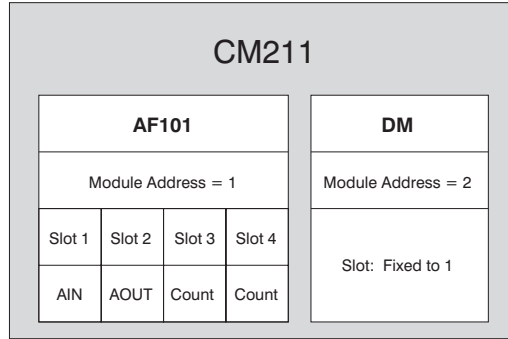
13.4.7 Counter Configuration Possibilities

Counter Inputs

Section "Module Design" explains that the combination module corresponds to an adapter module AF101 with four screw-in modules and a digital mixed module. Two of the four screw-in modules are used for counter inputs.

AF101			
Slot 1	Slot 2	Slot 3	Slot 4
A I N	A O U T	C O U N T	C O U N T

AIN ... Analog inputs
AOUT ... Analog outputs
COUNT ... Counter inputs

Counter Inputs in Slot 3

The counter inputs on the screw-in module installed in slot 3 can be configured using software. The desired function is set by defining bits 12 and 13 in configuration word 14. One of three functions can be selected.

Function	Bit 12	Bit 13
1 x incremental encoder	0	0
2 x event counter	1	0
2 x gate/period measurement	1	1

Counter Inputs in Slot 4

The counter inputs on the screw-in module installed in slot 4 can be configured using software. The desired function is set by defining bits 12 and 13 in configuration word 14. One of three functions can be selected.

Function	Bit 12	Bit 13
1 x incremental encoder	0	0
1 x event counter	1	0
1 x gate/period measurement	1	1

13.4.8 Timing

The following three factors must be taken into consideration for timing when a B&R 2003 CPU is used as controller:

- Internal Bus Cycle
- I/O AF Cycle
- I/O CPU Load

Internal bus cycle

I/O AF cycle
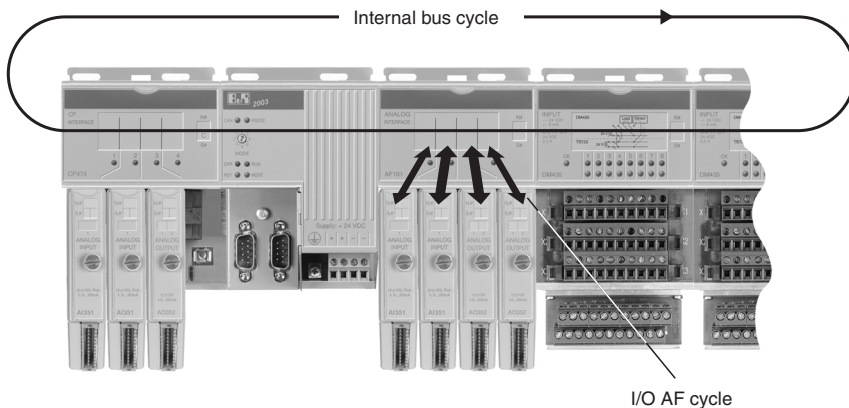
Internal Bus Cycle

All combination modules, AF modules and digital I/O modules are processed during this time. The internal bus cycle for a CM211 is calculated as follows:

<u>There is no AF101 adapter module on the bus</u>

t_{int_cycle} = n * 36 μs * 12 + 12 * 120 μs + 1200 μs = 3072 μs   (n = 1)

n ....Number of CM211 modules

36 μs ....Time for a combination module CM211

12 ....Number of data words for a CM211

120 μs ......Combination module CM211 busy

1200 μs ....Offset

<u>There is an AF101 adapter module on the bus or a CPx74 is used as CPU</u>

t_{int_cycle} = n * 36 μs * 12 + 12 * 200 μs + 1200 μs = 4032 μs   (n = 1)

n ....Number of CM211 modules

36 μs ....Time for a combination module CM211

12 ....Number of data words for a CM211

200 μs ......AF101 or CPx74 busy

1200 μs ....Offset

I/O AF cycle for digital data points

During this time, all digital data points on the combination module CM211 are updated or read in internally.

$t_{dig\_IO\_AF} \leq$ 1 ms

I/O AF cycle for analog data points

During this time, all analog data points on the combination module CM211 are updated or read in internally.

Counter values .... $t_{an\_IO\_AF} \leq$ 2 ms
Analog input/output values .... $t_{an\_IO\_AF} \leq$ 4 ms

I/O CPU Load

This time determines how long the CPU requires to process the I/O data passed on by the combination module CM211. The CPU is loaded considerably by the analog I/O data.

A CP430 or CPx70 is used as CPU

t_{IO_CPU} = 12 * 100 μs = 1200 μs

12 ....Number of data words for a CM211

100 μs ......Analog data point on CP430 or CPx70

A CPx74 is used as CPU

t_{IO_CPU} = 12 * 70 μs = 840 μs

12 ....Number of data words for a CM211

70 μs ....Analog data point on CPx74

A CP476 is used as CPU

t_{IO_CPU} = 12 * 50 μs = 600 μs

12 ....Number of data words for a CM211

50 μs ....Analog data point on CP476

Task Class

Fastest Task Class Recommended: 6 ms

13.4.9 Legend Sheets

A legend sheet can be slid into the front of the module from above. The module circuit is shown on the back. The inputs/outputs can be labelled on the front.
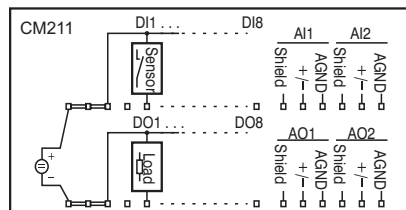
13.4.10 Connections
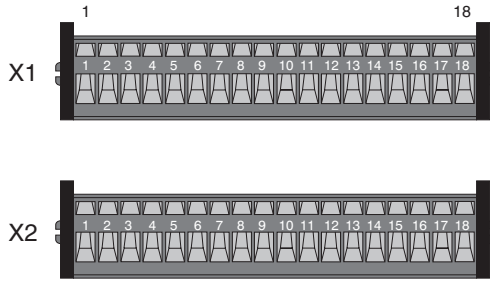
TB718
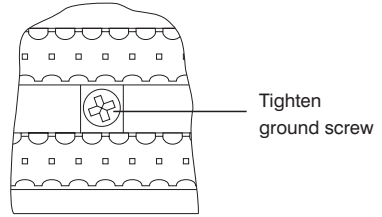
Terminal Block X1	
1	+24 VDC
2	+24 VDC
3	+24 VDC
4	n.c.
5	Digital input DI1
6	Digital input DI2
7	Digital input DI3
8	Digital input DI4
9	Digital input DI5
10	Digital input DI6
11	Digital input DI7
12	Digital input DI8
13	Shield
14	Analog input AI1 +
15	Analog input 1 GND (AGND)
16	Shield
17	Analog input AI2 +
18	Analog input 2 GND (AGND)

Terminal Block X2	
1	GND
2	GND
3	GND
4	n.c.
5	Digital output DO1
6	Digital output DO2
7	Digital output DO3
8	Digital output DO4
9	Digital output DO5
10	Digital output DO6
11	Digital output DO7
12	Digital output DO8
13	Shield
14	Analog output AO1 +
15	Analog output 1 GND (AGND)
16	Shield
17	Analog output AO2 +
18	Analog output 2 GND (AGND)

13.4.11 Connection Example Analog Inputs

The combination module CM211 has two analog inputs which may be used as either voltage or current input. Mixed operation is also possible.

Mode Switch

CM211

Mode Switch Input 2
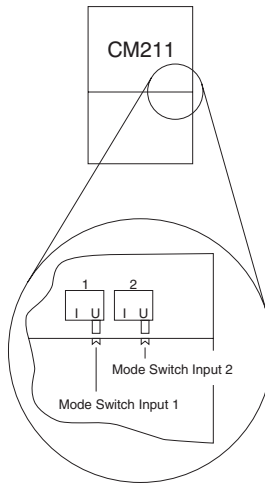
Mode Switch Input 1

An input can be used as either voltage or current input. The selection is made with the respective mode switch on the bottom of the housing. Place the switch in the respective position for the desired signal:

U ..... Voltage Input
I ..... Current Input

Module Wiring

In the following example, input 1 is used as voltage input.
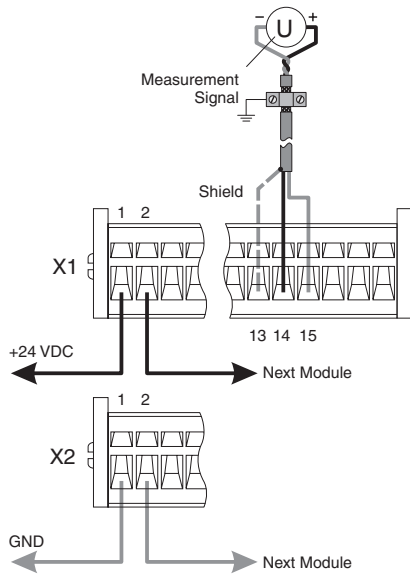
13.4.12 Connection Example Analog Outputs
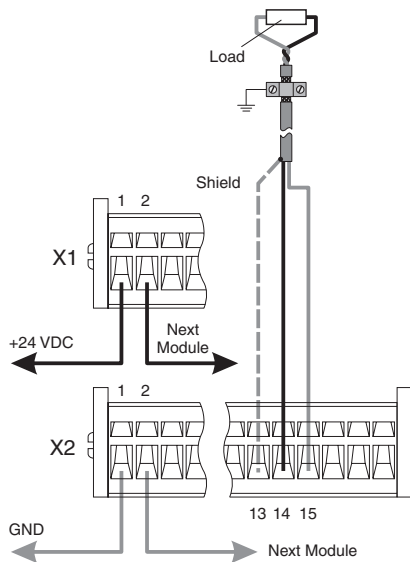
13.4.13 Connection Example Incremental Encoder Operation
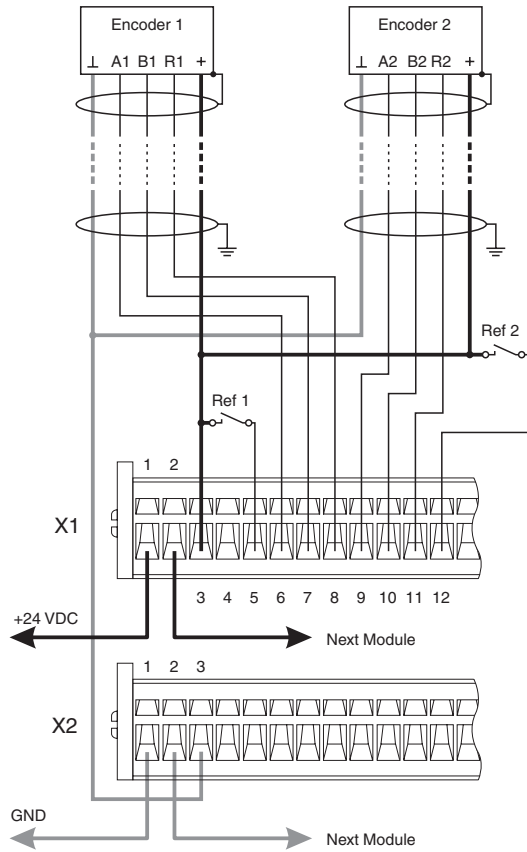
13.4.14 Connection Example Event Counter Operation
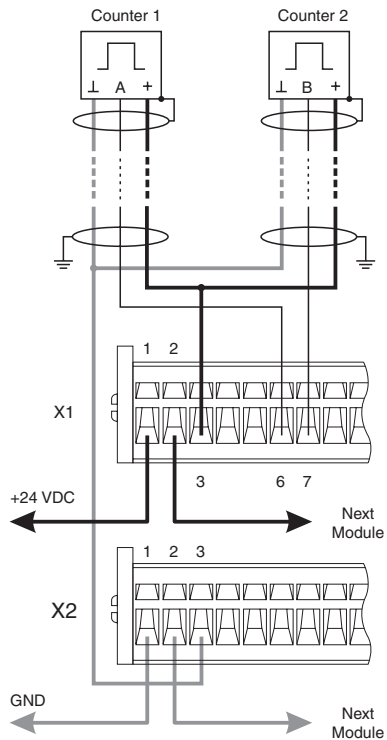
13.4.15 Connection Example Period Measurement

The period can be measured from a signal connected to input 3, 4 or 7. The resolution is 16 bit. The frequency of the signal to be measured can be a maximum of 20 kHz.
An internal or external counter frequency can be selected for the measurement.

- Internal counter frequency (16 MHz, 4 MHz, 1 MHz or 250 kHz)
- External counter frequency (max. 20 kHz)

The external counter frequency is connected to input 2 (for inputs 3 and 4) and 5 (for input 7).

Period Measurement Principle
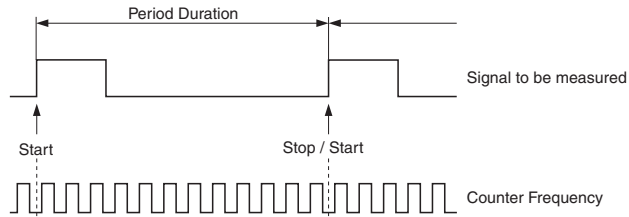
Pulse counting is started by a high edge on the input and stopped by the next high edge. The count is placed in a temporary register. The counter is started again by the same high edge.

During period measurement, the count stored last (the period) can be read by the active application program. The value in the temporary register is only updated after at the end of the active measurement. If a counter overflow occurs (broken line or incorrect counter frequency), the value is limited to $7FFF.

Connection Example for Period Measurement
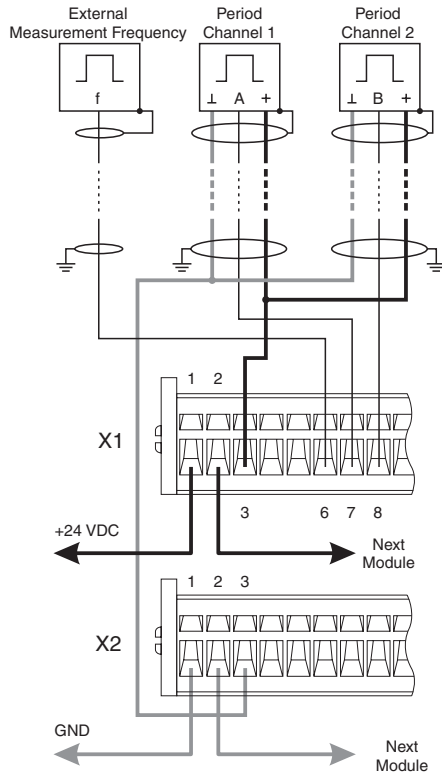
13.4.16 Connection Example Gate Measurement

A signal connected to input 3, 4 or 7 can be evaluated using gate measurement. The resolution is 16 bit. The frequency of the signal to be measured can be a maximum of 10 kHz. The signal to be measured is also called the gate frequency. The pause between two gate measurements must be larger than 50 μs.

An internal or external counter frequency can be selected for the measurement.

- Internal counter frequency (16 MHz, 4 MHz, 1 MHz or 250 kHz)
- External counter frequency (max. 20 kHz)

The external counter frequency is connected to input 2 (for inputs 3 and 4) and 5 (for input 7).

Gate Measurement Principle
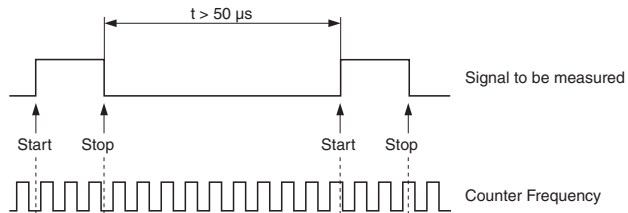
Pulse counting is started by a high edge on the gate and stopped by a low edge. The count is placed in a temporary register when the low edge occurs. The counter is started again by the next high edge.

During gate measurement, the count stored last (the gate) can be read by the active application program. The value in the temporary register is only updated after at the end of the active measurement (low edge). If a counter overflow occurs (incorrect counter frequency), the value is limited to $7FFF.

Connection Example for Gate Measurement
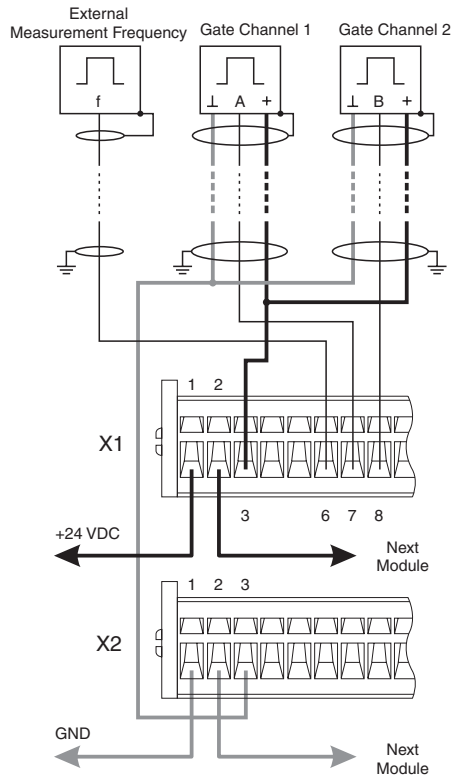
13.4.17 Connection Example Digital Outputs

13.4.18 Variable Declaration for the Analog Inputs

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for the analog inputs.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	INT16	Analog In	1	●		Analog input value channel 1
Data word 1	INT16	Analog In	2	●		Analog input value channel 2
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. IN	28	●		Module type

Access using CAN Identifiers

Access via CAN Identifier is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

In the example below, the combination module CM211 is accessed with module addresses 1 and 2.

Analog input data cannot be packed. Only the first object from this group of four will be created and sent.

Slot	CAN ID [1]	Word 1		Word 2		Word 3	Word 4
1	542	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Not used	
2	543	Not used					
3	544	Not used					
4	545	Not used					

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address = 1

sl .... Slot number = 1

B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

Description of Data and Configuration Words

Data Words 0 and 1 (read)

16 bit standardized values representing either voltage or current.

Configuration Word 12 (read)

Configuration word 12 contains the module status.

	Bit	Description
	12 - 15	x....Not defined, masked out
	11	0....Converter values ready 1....Converter values not yet ready
	6 - 10	x....Not defined, masked out
	5	0....Channel 2: Current measurement 1....Channel 2: Voltage measurement
	4	0....Channel 1: Current measurement 1....Channel 1: Voltage measurement
	2 - 3	x....Not defined, masked out
	1	0....Channel 2: No error 1....Channel 2: Error present
	0	0....Channel 1: No error 1....Channel 1: Error present

x	x	x	x		x	x	x	x	x			x	x		

15   8 7   0

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $40
	0 - 7	x ....Not defined, masked out

0	1	0	0	0	0	0	0	x	x	x	x	x	x	x	x

15   8 7   0

13.4.19 Variable Declaration for the Analog Outputs

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for the analog outputs.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	INT16	Analog Out	1		●	Analog output value channel 1
Data word 1	INT16	Analog Out	2		●	Analog output value channel 2
Configuration word 14	WORD	Transp. In	28	●		Module type

Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer. Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

In the example below, the combination module CM211 is accessed with module addresses 1 and 2.

Analog output data cannot be packed. Only the second object from this group of four will be created.

Slot	CAN ID [1]	Word 1		Word 2		Word 3	Word 4
1	1054	Not used					
2	1055	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Not used	
3	1056	Not used					
4	1057	Not used					

[1] CAN ID = 1054 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address = 1

sl .... Slot number = 2

⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

Description of Data and Configuration Words

Data Words 0 and 1 (write)

The 16 bit standardized values for voltage are written to the module output channel.

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $40
	0 - 7	x ....Not defined, masked out

0	1	0	0	0	0	0	0	x	x	x	x	x	x	x	x

15   8 7   0

13.4.20 Variable Declaration for Incremental Encoder Operation

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

The module CM211 is equipped with a maximum of two incremental encoders. The incremental encoders are distributed on slots 3 and 4.

Incremental encoder 1 (slot 3) with PCC 2003 CPU and remote slaves

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for the first incremental encoder.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	WORD	Transp. In	0	●		Module status
Data word 1	INT32	Transp. In	2	●		Counter value
Configuration word 4	INT32	Transp. In	8	●		Counter value with pos. edge on the reference input [1]
	INT32	Transp. Out	8		●	Threshold value 1 / min. latch difference [1]
Configuration word 6	INT32	Transp. In	12	●		Counter value with neg. edge on the reference input [1]
	INT32	Transp. Out	12		●	Threshold value 2 / force value [1]
Configuration word 8	WORD	Transp. Out	16		●	Incremental encoder control [1]
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

[1] Starting with Rev. D0

Incremental encoder 1 (Slot 3) with CAN slaves

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for the first incremental encoder.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	INT32	Transp. In	0	●		Counter value
Data word 2	WORD	Transp. In	4	●		Module status
Configuration word 4	INT32	Transp. In	8	●		Counter value with pos. edge on the reference input [1]
	INT32	Transp. Out	8		●	Threshold value 1 / min. latch difference [1]
Configuration word 6	INT32	Transp. In	12	●		Counter value with neg. edge on the reference input [1]
	INT32	Transp. Out	12		●	Threshold value 2 / force value [1]
Configuration word 8	WORD	Transp. Out	16		●	Incremental encoder control [1]
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

[1] Starting with Rev. D0

> ⚠ **B&R 2000 users have to exchange the two counter status words so that the high word is first (Motorola format)!**

Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

In the example below, the combination module CM211 is accessed with module addresses 1 and 2.

Slot	CAN ID [1]	Word 1		Word 2		Word 3		Word 4
1	542	Not used						
2	543	Not used						
3	544	Count. LL	Count. ML	Count. MH	Count. HH	Status L	Status H	Not used
4	545	Not used						

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

  nd ..... Node number of the CAN slave = 1

  ma .... Module address = 1

  sl .... Slot number = 3

> ⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

Description of Data and Configuration Words

Data Word 0 (read)

Data word 0 includes the module status time constant for the counter value.

	Bit	Description
	12 - 15	x....Not defined, masked out
	11	0....Counter value not taken 1....Counter value is taken with the first positive edge on the reference input
	10	0....Counter value not taken 1....Counter value is taken with the first negative edge on the reference input
	8 - 9	x....Not defined, masked out
	7	0....Referencing is taking place 1....Counter is referenced (resetting takes place when the reference command is received)
	6	Changes state each time referencing takes place
	5	0....Supply voltage <18 V 1....Supply voltage >18 V, Outputs OK
	4	Output status of the comparator
	3	Level of encoder input A
	2	Level of encoder input B
	1	Level of the reference enable switch
	0	Level of the reference pulse

x x x x   x x

15   8 7   0

Data Word 1 (read)

Counter Value MSW

Data Word 2 (read)

Counter Value LSW

Configuration Words 4+5 (read) - Starting with Rev. D0

After setting bit 11 in configuration word 8, the configuration words receive the latched counter value with the first positive edge on the reference input. The value is also valid if bit 11 is set in data word 0.
The maximum delay, from the edge on the input to the value is latched, is 50 μs.

Configuration Words 4+5 (write)

Either threshold value 1 or the minimum latch difference is defined with configuration words 4+5.

1) Threshold Value 1 (32 Bit):

   a) Number format 32 bit with sign:
      Bit 10 in configuration word 14 (write) is set to 0.
      Threshold value 1 must always be ≤ threshold value 2 .
      Threshold values are internally arranged in increasing order including sign.

   b) Number format 32 bit without sign - cyclic operation:
      Bit 10 in configuration word 14 (write) is set to 1.
      The threshold values are not placed in order internally. The sign is not used in the comparator calculation.

2) Minimum Latch Difference (32 Bit) - Starting with Rev. D0:

   Definition of the minimum latch difference for latching counters. This is done using bit 8 in configuration word 8.

Configuration Words 6+7 (read) - Starting with Rev. D0

After setting bit 10 in configuration word 8, the configuration words receive the latched counter value with the first negative edge on the reference input. The value is also valid if bit 10 is set in data word 0.
The maximum delay, from the edge on the input to the value is latched, is 50 μs.

Configuration Words 6+7 (write)

Either threshold value 2 (32 Bit) or the force value (32 Bit) is defined with configuration words 6+7. The definition of the force value is possible starting with Rev. D0. The force value is defined using bit 8 in configuration word 8.
The maximum delay, until the force value definition is accepted, is 1 ms.

The incremental encoder is configured using configuration word 8.

	Bit	Description
	14 - 15	0
	13	0....... No effect on the incremental encoder
		1....... Inverts the reference enable switch for forcing
		Using bit 13, the polarity of the reference enable switch can be inverted.
	12	0....... No effect on the incremental encoder
		1....... Forcing the counter with the reference enable switch
		With a positive edge on bit 12, bit 7 is deleted in data word 0 and positive edge creation for the reference enable switch is activated. When the edge is recognized, the counter is set to the predefined value (default 0). In data word 0, bit 7 is set and bit 6 is inverted.
		The precision is limited compared to latch operation.
	11	0....... Counter value not taken
		1....... Counter value is taken with the first positive edge on the reference input (see configuration words 4+5) [1]
	10	0....... Counter value not taken
		1....... Counter value is taken with the first negative edge on the reference input (see configuration words 6+7) [1]
	9	0/1.... Start output state for the latch input filter
		With bit 9, the start condition for edge creation and filtering of the latch signal can be defined. If the state of bit 9 is not the same as the current state of the reference input, an edge change is recognized internally and evaluated using the filter.
	8	0....... No effect on the incremental encoder
		1....... With a positive edge, configuration words 4+5 are used as minimum latch value and configuration words 6+7 are used as force value
	0 - 7	0

0	0								0	0	0	0	0	0	0	0

15   8 7   0

[1] The counter value is only taken once. Bit 10 and bit 11 must be reset for the value to be taken again. Bit 10 and bit 11 can be set again in configuration word 8 after the corresponding bit in the module status bit has gone to 0.

Latching the counter value using the reference input:

⚠️ **Bit 0 in configuration word 14 must be 0. That means, the comparator is switched off.**

General Information:

The same internal interrupts or inputs are used for referencing, comparator operation and latching the counter value. Therefore the following priority is defined:

1. Referencing
2. Comparator
3. Latching

Make sure there is a clean transition between the types of operation. That means the respective control bits are not allowed to be activated at the same time.

Latching the counter value:

Latching is started by a positive edge on bit 10 or bit 11 in configuration word 8. The current counter value is taken as comparison value for the minimum latch difference.

1) Minimum latch difference is 0:

   If the minimum latch difference is set to 0 (default), the counter value is latched immediately when the respective edge occurs on the reference input.

2) Minimum latch difference is not 0:

   If the minimum latch difference is not set to 0, the counter value is placed in temporary memory when the respective edge occurs on the reference input.

   • Difference > minimum latch difference:

     If the difference between the value in temporary memory minus the comparison value is larger than the minimum latch difference, the value is accepted and bit 10 or bit 11 in data word 0 is set.

   • Difference < minimum latch difference:

     If the difference between the value in temporary memory minus the comparison value is smaller than the minimum latch difference, the value is not yet accepted. The comparison is made again when the next corresponding edge occurs on the reference input.
     The value is only accepted and bit 10 or bit 11 in data word 0 is only set if the difference is larger than the minimum latch difference.

Configuration Word 12 (read)

Configuration word 12 contains the module status (current status unlatched). The module status is written to data word 0.

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $40
	0 - 7	x ....Not defined, masked out

0	1	0	0	0	0	0	0	x	x	x	x	x	x	x	x

15   8   7   0

Configuration Word 14 (write)

The module is configured using configuration word 14

Bit	Description
14 - 15	0
13	0 ....Incremental encoder operation
12	0 ....Incremental encoder operation
11	0 ....No effect on count direction 1 ....Count direction inverted as compared to counter wiring
10	0 ....Number format: 32 bit with sign 1 ....Number format: 32 bit without sign - cyclic operation In continuous operation, the internal order of the threshold values are kept. If a counter overflow occurs, the behavior of the comparator does not have to be changed. Comparator operation takes place without consideration of the sign.
8 - 9	0 ....Incremental encoder operation with 4-fold evaluation
7	0 ....No effect on reference pulse 1 ....Reference pulse is inverted. This setting is used for encoders with a high pulse.
6	0 ....Set counter immediately to 0. In data word 0 (module status), bit 7 is immediately set to 1 and the counter is cleared. 1 ....Counter remains functioning. In data word 0 (module status), bit 7 is immediately set to 0 (referencing required).
5	0 ....Ignore reference enable switch (referencing using reference pulse). Setting refers to bit 4 1 ....Actively switch reference enable switch (referencing using reference pulse and reference enable switch)
4	0 ....No effect on counter 1 ....Clear counter (reference)
3	0 ....Comparator off Output 1 is handled as defined in the variable declaration for digital outputs. 1 ....Comparator on
2	0 ....The comparator output is set to the level given in bit 0, if threshold value 1 $\leq$ counter $\leq$ threshold value 2 1 ....The comparator output is set to the inverted level of bit 0, if threshold value 1 $\leq$ counter $\leq$ threshold value 2
1	0
0	Level of the comparator output

0	0	0	0		0	0							0		

15   8   7   0

Incremental encoder 2 (slot 4) with PCC 2003 CPU and remote slaves

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for the second incremental encoder.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	WORD	Transp. In	0	●		Module status
Data word 1	INT32	Transp. In	2	●		Counter value
Configuration word 4	INT32	Transp. In	8	●		Counter value with pos. edge on the reference input [1]
	INT32	Transp. Out	8		●	Min. latch difference [1]
Configuration word 6	INT32	Transp. In	12	●		Counter value with neg. edge on the reference input [1]
	INT32	Transp. Out	12		●	Force value [1]
Configuration word 8	WORD	Transp. Out	16		●	Incremental encoder control [1]
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

[1] Starting with Rev. D0

Incremental encoder 2 (Slot 4) with CAN slaves

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for the second incremental encoder.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	INT32	Transp. In	0	●		Counter value
Data word 2	WORD	Transp. In	4	●		Module status
Configuration word 4	INT32	Transp. In	8	●		Counter value with pos. edge on the reference input [1]
	INT32	Transp. Out	8		●	Min. latch difference [1]
Configuration word 6	INT32	Transp. In	12	●		Counter value with neg. edge on the reference input [1]
	INT32	Transp. Out	12		●	Force value [1]
Configuration word 8	WORD	Transp. Out	16		●	Incremental encoder control [1]
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

[1] Starting with Rev. D0

> ⚠ **B&R 2000 users have to exchange the two counter status words so that the high word is first (Motorola format)!**

Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

In the example below, the combination module CM211 is accessed with module addresses 1 and 2.

Slot	CAN ID [1]	Word 1		Word 2		Word 3		Word 4
1	542	Not used						
2	543	Not used						
3	544	Not used						
4	545	Count. LL	Count. ML	Count. MH	Count. HH	Status L	Status H	Not used

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address = 1

sl .... Slot number = 4

⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

Description of Data and Configuration Words

Data Word 0 (read)

Data word 0 includes the module status time constant for the counter value.

	Bit	Description
	12 - 15	x .... Not defined, masked out
	11	0.... Counter value not taken 1.... Counter value is taken with the first positive edge on the reference input
	10	0.... Counter value not taken 1.... Counter value is taken with the first negative edge on the reference input
	8 - 9	x .... Not defined, masked out
	7	0.... Referencing is taking place 1.... Counter is referenced (resetting takes place when the reference command is received)
	6	Changes state each time referencing takes place
	5	0.... Supply voltage <18 V 1.... Supply voltage >18 V, Outputs OK
	4	x .... Not defined, masked out
	3	Level of encoder input A
	2	Level of encoder input B
	1	Level of the reference enable switch
	0	Level of the reference pulse

```
x  x  x  x        x  x        x
15          8  7           0
```

Data Word 1 (read)

Counter Value MSW

Data Word 2 (read)

Counter Value LSW

Configuration Words 4+5 (read) - Starting with Rev. D0

After setting bit 11 in configuration word 8, the configuration words receive the latched counter value with the first positive edge on the reference input. The value is also valid if bit 11 is set in data word 0.
The maximum delay, from the edge on the input to the value is latched, is 50 μs.

Configuration Words 4+5 (write) - Starting with Rev. D0

Definition of the minimum latch difference (32 Bit) for latching counters. This is done using bit 8 in configuration word 8.

Configuration Words 6+7 (read) - Starting with Rev. D0

After setting bit 10 in configuration word 8, the configuration words receive the latched counter value with the first negative edge on the reference input. The value is also valid if bit 10 is set in data word 0. The maximum delay, from the edge on the input to the value is latched, is 50 μs.

Configuration Words 6+7 (write) - Starting with Rev. D0

Definition of the force value (32 Bit). The force value is defined using bit 8 in configuration word 8. The maximum delay, until the force value definition is accepted, is 1 ms.

Configuration Word 8 (read) - Starting with Rev. D0

The incremental encoder is configured using configuration word 8.

Bit	Description
14 - 15	0
13	0....... No effect on the incremental encoder
	1....... Inverts the reference enable switch for forcing
	Using bit 13, the polarity of the reference enable switch can be inverted.
12	0....... No effect on the incremental encoder
	1....... Forcing the counter with the reference enable switch
	With a positive edge on bit 12, bit 7 is deleted in data word 0 and positive edge creation for the reference enable switch is activated. When the edge is recognized, the counter is set to the predefined value (default 0). In data word 0, bit 7 is set and bit 6 is inverted.
	The precision is limited compared to latch operation.
11	0....... Counter value not taken
	1....... Counter value is taken with the first positive edge on the reference input (see configuration words 4+5) [1]
10	0....... Counter value not taken
	1....... Counter value is taken with the first negative edge on the reference input (see configuration words 6+7) [1]
9	0/1.... Start output state for the latch input filter
	With bit 9, the start condition for edge creation and filtering of the latch signal can be defined. If the state of bit 9 is not the same as the current state of the reference input, an edge change is recognized internally and evaluated using the filter.
8	0....... No effect on the incremental encoder
	1....... With a positive edge, configuration words 4+5 are used as minimum latch value and configuration words 6+7 are used as force value
0 - 7	0

```
0 0 | | | | | | | 0 0 0 0 0 0 0 0
15              8 7              0
```

[1] The counter value is only taken once. Bit 10 and bit 11 must be reset for the value to be taken again. Bit 10 and bit 11 can be set again in configuration word 8 after the corresponding bit in the module status bit has gone to 0.

Latching the counter value using the reference input:

General Information:

The same internal interrupts or inputs are used for referencing and latching the counter value. Therefore the following priority is defined:

1. Referencing
2. Latching

Make sure there is a clean transition between the types of operation. That means the respective control bits are not allowed to be activated at the same time.

Latching the counter value:

Latching is started by a positive edge on bit 10 or bit 11 in configuration word 8. The current counter value is taken as comparison value for the minimum latch difference.

1) Minimum latch difference is 0:

   If the minimum latch difference is set to 0 (default), the counter value is latched immediately when the respective edge occurs on the reference input.

2) Minimum latch difference is not 0:

   If the minimum latch difference is not set to 0, the counter value is placed in temporary memory when the respective edge occurs on the reference input.

   • Difference > minimum latch difference:

     If the difference between the value in temporary memory minus the comparison value is larger than the minimum latch difference, the value is accepted and bit 10 or bit 11 in data word 0 is set.

   • Difference < minimum latch difference:

     If the difference between the value in temporary memory minus the comparison value is smaller than the minimum latch difference, the value is not yet accepted. The comparison is made again when the next corresponding edge occurs on the reference input.
     The value is only accepted and bit 10 or bit 11 in data word 0 is only set if the difference is larger than the minimum latch difference.

Configuration Word 12 (read)

Configuration word 12 contains the module status (current status unlatched). The module status is written to data word 0.

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $40
	0 - 7	x ....Not defined, masked out

0	1	0	0	0	0	0	0	x	x	x	x	x	x	x	x

15   8 7   0

Configuration Word 14 (write)

The module is configured using configuration word 14.

	Bit	Description
	14 - 15	0
	13	0 ....Incremental encoder operation
	12	0 ....Incremental encoder operation
	11	0 ....No effect on count direction 1 ....Count direction inverted as compared to counter wiring
	10	0
	8 - 9	0 ....Incremental encoder operation with 4-fold evaluation
	7	0 ....No effect on reference pulse 1 ....Reference pulse is inverted. This setting is used for encoders with a high pulse.
	6	0 ....Set counter immediately to 0. In data word 0 (module status), bit 7 is immediately set to 1 and the counter is cleared. 1 ....Counter remains functioning. In data word 0 (module status), bit 7 is immediately set to 0 (referencing required).
	5	0 ....Ignore reference enable switch (referencing using reference pulse). Setting refers to bit 4 1 ....Actively switch reference enable switch (referencing using reference pulse and reference enable switch)
	4	0 ....No effect on counter 1 ....Clear counter (reference)
	3	0
	2	0
	1	0
	0	0

		0	0							0	0	0	0	0

15   8 7   0

13.4.21 Variable Declaration for Event Counter Operation

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

The module CM211 is equipped with a maximum of two event counters. The event counters are distributed on slots 3 and 4.

Event Counter 1 and 2 (Slot 3)

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for event counters 1 and 2.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	WORD	Transp. In	0	●		Module status
Data word 1	WORD	Transp. In	2	●		Counter value of counter 1
Data word 2	WORD	Transp. In	4	●		Counter value of counter 2
Configuration word 5	WORD	Transp. Out	10		●	Threshold value 1 for counter 2
Configuration word 7	WORD	Transp. Out	14		●	Threshold value 2 for counter 2
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

In the example below, the combination module CM211 is accessed with module addresses 1 and 2.

Slot	CAN ID [1]	Word 1		Word 2		Word 3		Word 4
1	542	Not used						
2	543	Not used						
3	544	Count. 2 L	Count. 2 H	Count. 1 L	Count. 1 H	Status L	Status H	Not used
4	545	Not used						

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address = 1

sl .... Slot number = 3

> ⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

Description of Data and Configuration Words

Data Word 0 (read)

Data word 0 includes the module status time constant for both counter values.

	Bit	Description
	6 - 15	x ....Not defined, masked out
	5	0 ....Supply voltage <18 V 1 ....Supply voltage >18 V, Outputs OK
	4	Output status of the comparator
	3	Level of the encoder input for counter 1
	2	Level of the encoder input for counter 2
	0 - 1	x ....Not defined, masked out

x	x	x	x	x	x	x	x	x	x					x	x

15   8 7   0

Data Word 1 (read)

Counter value of counter 1.

Data Word 2 (read)

Counter value of counter 2.

Configuration Word 5 (write)

Threshold value 1 (16 Bit) for counter 2:

    a) Number format: 16 bit without sign
       Bit 10 in configuration word 14 (write) is set to 0.
       Threshold value 1 must always be ≤ threshold value 2 .
       Threshold values are internally arranged in increasing order. The sign is ignored.

    b) Number format: 16 bit without sign - cyclic operation
       Bit 10 in configuration word 14 (write) is set to 1.
       The threshold values are not placed in order internally. The sign is ignored.

Configuration Word 7 (write)

Threshold value 2 (16 Bit) for counter 2.

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $40
	0 - 7	x ....Not defined, masked out

0	1	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x

15   8 7   0

Configuration Word 14 (write)

The module is configured using configuration word 14.

	Bit	Description
	14 - 15	0
	13	0....Event counter operation
	12	1....Event counter operation
	11	0....No effect on counting direction 1....Count direction inverted as compared to counter wiring
	10	0....Number format: 16 bit without sign
		1....Number format: 16 bit without sign - cyclic operation In continuous operation, the internal order of the threshold values are kept. If a counter overflow occurs, the behavior of the comparator does not have to be changed.
	9	0
	8	0....Counter 1 and 2: Only count positive edges 1....Counter 1 and 2: Count both edges
	6 - 7	0
	5	0....No effect on counter 2 1....Immediately clear counter 2
	4	0....No effect on counter 1 1....Immediately clear counter 1
	3	0....Comparator off Output 1 is handled as defined in the variable declaration for digital outputs. 1....Comparator on
	2	0....The comparator output is set to the level given in bit 0, if threshold value $1 \leq$ counter $\leq$ threshold value 2 1....The comparator output is set to the inverted level of bit 0, if threshold value $1 \leq$ counter $\leq$ threshold value 2
	1	0
	0	Level of the comparator output

0	0	0	1		0		0	0					0		

15   8   7   0

Event Counter 3 (Slot 4)

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for event counter 3.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	WORD	Transp. In	0	●		Module status
Data word 1	WORD	Transp. In	2	●		Counter value of counter 3
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

In the example below, the combination module CM211 is accessed with module addresses 1 and 2.

Slot	CAN ID [1]	Word 1	Word 2		Word 3		Word 4
1	542	Not used					
2	543	Not used					
3	544	Not used					
4	545	Not used	Count. 3 L	Count. 3 H	Status L	Status H	Not used

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address = 1

sl .... Slot number = 4

⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

Description of Data and Configuration Words

Data Word 0 (read)

Data word 0 includes the module status time constant for the counter value.

			Bit	Description
			4 - 15	x ....Not defined, masked out
			3	Level of the encoder input for counter 3
			0 - 2	x ....Not defined, masked out

x	x	x	x	x	x	x	x	x	x	x	x		x	x	x

15   8 7   0

Data Word 1 (read)

Counter value of counter 3.

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

		Bit	Description
		8 - 15	Module code = $40
		0 - 7	x ....Not defined, masked out

0	1	0	0	0	0	0	0	x	x	x	x	x	x	x	x

15   8   7   0

Configuration Word 14 (write)

The module is configured using configuration word 14.

		Bit	Description
		14 - 15	0
		13	0 ....Event counter operation
		12	1 ....Event counter operation
		11	0 ....No effect on counting direction 1 ....Count direction inverted as compared to counter wiring
		9 - 10	0
		8	0 ....Counter 3: Only count positive edges 1 ....Counter 3: Count both edges
		5 - 7	0
		4	0 ....No effect on counter 3 1 ....Immediately clear counter 3
		0 - 3	0

0	0	0	1		0	0		0	0	0		0	0	0	0

15   8   7   0

13.4.22 Variable Declaration for Gate and Period Measurement

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

The module CM211 is equipped with a maximum of three channels for gate and period measurement. The channels are distributed on slots 3 and 4.

Gate and Period Measurement 1 and 2 (Slot 3)

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	WORD	Transp. In	0	●		Module status
Data word 1	WORD	Transp. In	2	●		Counter value gate / period 1
Data word 2	WORD	Transp. In	4	●		Counter value gate / period 2
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

In the example below, the combination module CM211 is accessed with module addresses 1 and 2.

Slot	CAN ID [1]	Word 1		Word 2		Word 3		Word 4
1	542	Not used						
2	543	Not used						
3	544	Count. 2 L	Count. 2 H	Count. 1 L	Count. 1 H	Status L	Status H	Not used
4	545	Not used						

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address = 1

sl .... Slot number = 3

> ⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

Description of Data and Configuration Words

Data Word 0 (read)

Data word 0 includes the module status time constant for both counter values.

	Bit	Description
	3 - 15	x ....Not defined, masked out
	2	Level of the encoder input for gate / period 1
	1	x ....Not defined, masked out
	0	Level of the encoder input for gate / period 2

x	x	x	x	x	x	x	x	x	x	x	x	x		x	

15   8 7   0

Data Word 1 (read)

Counter Value Gate / Period 1.

Data Word 2 (read)

Counter Value Gate / Period 2.

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $40
	0 - 7	x ....Not defined, masked out

0	1	0	0	0	0	0	0	x	x	x	x	x	x	x	x

15   8 7   0

Configuration Word 14 (write)

The module is configured using configuration word 14.

	Bit	Description
	14 - 15	0
	13	1....Gate / period measurement
	12	1....Gate / period measurement
	11	0....Channel 2: No effect on counter value 1....Channel 2: Set counter value to zero (retrigger)
	10	0....Channel 1: No effect on counter value 1....Channel 1: Set counter value to zero (retrigger)
	8 - 9	0
	7	0....Channel 2: Gate measurement 1....Channel 2: Period measurement
	4 - 6	Channel 2: Definition of counter frequency 0....16 MHz, internal 1....4 MHz, internal 2....1 MHz, internal 3....250 kHz, internal 4....external, only rising edges 5....external, both edges
	3	0....Channel 1: Gate measurement 1....Channel 1: Period measurement
	0 - 2	Channel 1: Definition of counter frequency 0....16 MHz, internal 1....4 MHz, internal 2....1 MHz, internal 3....250 kHz, internal 4....external, only rising edges 5....external, both edges

```
0 0 1 1     0 0
15          8 7          0
```

Gate and Period Measurement 3 (Slot 4)

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	WORD	Transp. In	0	●		Module status
Data word 1	WORD	Transp. In	2	●		Counter value gate / period 3
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

In the example below, the combination module CM211 is accessed with module addresses 1 and 2.

Slot	CAN ID [1]	Word 1	Word 2		Word 3		Word 4
1	542	Not used					
2	543	Not used					
3	544	Not used					
4	545	Not used	Count. 3 L	Count. 3 H	Status L	Status H	Not used

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

   nd ..... Node number of the CAN slave = 1

   ma .... Module address = 1

   sl .... Slot number = 4

> ⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

Description of Data and Configuration Words

Data Word 0 (read)

Data word 0 includes the module status time constant for the counter value.

	Bit	Description
	1 - 15	x ....Not defined, masked out
	0	Level of the encoder input for gate / period 3

x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	

15   8 7   0

Data Word 1 (read)

Counter Value Gate / Period 3.

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

		Bit	Description
		8 - 15	Module code = $40
		0 - 7	x .... Not defined, masked out

0	1	0	0	0	0	0	0	x	x	x	x	x	x	x	x

15   8 7   0

Configuration Word 14 (write)

The module is configured using configuration word 14.

		Bit	Description
		14 - 15	0
		13	1 .... Gate / period measurement
		12	1 .... Gate / period measurement
		11	0
		10	0 .... Channel 3: No effect on counter value 1 .... Channel 3: Set counter value to zero (retrigger)
		4 - 9	0
		3	0 .... Channel 3: Gate measurement 1 .... Channel 3: Period measurement
		0 - 2	Channel 3: Definition of counter frequency 0 .... 16 MHz, internal 1 .... 4 MHz, internal 2 .... 1 MHz, internal 3 .... 250 kHz, internal 4 .... external, only rising edges 5 .... external, both edges

0	0	1	1	0		0	0	0	0	0	0				

15   8 7   0

13.4.23 Variable Declaration for Digital Inputs/Outputs

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Designation	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Digital inputs 1 - 8	BIT	Digit. In	1 ... 8	●		Level of the digital inputs
Digital outputs 1 - 8	BIT	Digit. Out	1 ... 8		●	Level of the digital outputs
Module status	BYTE	Status In	0	●		Module status

Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer. Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

Digital inputs

A maximum of eight digital I/O modules can be operated in packed mode.

The CM211 modules uses two module addresses. If two CM211 modules are used, only four additional digital I/O modules can be used.

The following example shows the structure of the CAN object if two CM211 and four DI435 modules are used.

CAN ID [1]	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
286	CM211 Not used	CM211 I 1 - 8	CM211 Not used	CM211 I 1 - 8	DI435	DI435	DI435	DI435

[1] CAN ID = 286 + (nd - 1) x 4   nd .... Node number of the CAN slave = 1

A maximum of four digital I/O module can be run in unpacked mode.

The following example shows the structure of the CAN object if one CM211 and two DI435 modules are used.

Module	CAN ID [1]	Byte
CM211	286	Not used
	287	Inputs 1 - 8
DI435	288	Inputs 1 - 8
DI435	289	Inputs 1 - 8

[1] CAN ID = 286 + (nd - 1) x 4 + (ma - 1)   nd ... Node number of the CAN slave = 1
ma ... Module address of digital I/O modules = 1 - 4

Digital Outputs

A maximum of eight digital I/O modules can be operated in packed mode.

The CM211 modules uses two module addresses. If two CM211 modules are used, only four additional digital I/O modules can be used.

The following example shows the structure of the CAN object if two CM211 and four DO722 modules are used.

CAN ID [1]	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
414	CM211 Not used	CM211 O 1 - 8	CM211 Not used	CM211 O 1 - 8	DO722	DO722	DO722	DO722

[1] CAN ID = 414 + (nd - 1) x 4   nd .... Node number of the CAN slave = 1

A maximum of four digital I/O module can be run in unpacked mode.

The following example shows the structure of the CAN object if one CM211 and two DO722 modules are used.

Module	CAN ID [1]	Byte
CM211	414	Not used
	415	Outputs 1 - 8
DO722	416	Outputs 1 - 8
DO722	417	Outputs 1 - 8

[1] CAN ID = 414 + (nd - 1) x 4 + (ma - 1)   nd ... Node number of the CAN slave = 1

ma ... Module address of digital I/O modules = 1 - 4

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

Module status

		Bit	Description
		7	0 ....No supply voltage or supply voltage too low for digital inputs/outputs 1 ....Module voltage OK
		6	Digital module = 0
		5	0 ....No error 1 ....Error (short circuit, over-temperature, etc.)
		0 - 4	Module code = $17

0		1	0	1	1	1

7   0

13.5 CM411

13.5.1 Technical Data

Terminal block is not included in the delivery.

Module ID	CM411
General Information	
Model Number	7CM411.70-1
Short Description	2003 Combination Module, 3 inputs, 24 VDC, 50 kHz, sink, one-channel or two-channel counters, incremental encoders, 2 transistor outputs, 24 VDC, 0.5 A, comparator function, short-circuit protection, 3 inputs, +/- 10 V, 16 Bit, 3 outputs, +/- 10 V, 16 Bit, Order terminal blocks separately!
C-UL-US Listed	in preparation
B&R ID Code	$C2
Module Type	B&R 2003 I/O Module
Amount [1]	
CP430, EX270	1
CP470, CP770 CP474, CP476, CP774 EX470, EX770 EX477, EX777	2
Voltage and Output Monitoring	Yes (LED: OK) Supply voltage >18 V, Outputs OK
Electrical Isolation Analog - PCC Digital - PCC Digital - Analog	No Yes Yes
Power Consumption	Max. 2.4 W
Analog Inputs	
Number of Inputs	3
Input Signal - Nominal	±10 V
Digital Converter Resolution	16 Bit

Module ID	CM411
Data Format Delivered to Application Program	16 Bit 2s complement
Value Range +10 V -10 V	 $7FFF $8001
Measurement Range Monitoring Open Inputs Range Exceeded (neg.) Range Exceeded (pos.) General Error	 $7FFF $8001 $7FFF $8000
Maximum Error at 25 °C	±0.1 % [2]
Offset Drift	±0.3 LSB / °C [2]
Gain Drift	±65 ppm / °C [3]
Maximum Error over Entire Temperature Range	±0.25 % [2]
Repeat Precision	±0.025 % [2]
Input Impedance in Signal Range	≥1 MΩ
Analog Outputs	
Number of Outputs	3
Output Signal	±10 V
Load	Max. ±10 mA
Digital Converter Resolution	16 Bit
Maximum Error at 25 °C	±0.1 % [2]
Offset Drift	±2.5 LSB / °C [2]
Gain Drift	±110 ppm / °C [3]
Maximum Error over Entire Temperature Range	±0.25 % [2]
Load Impedance	≥1 kΩ
Short Circuit Current	±15 mA (individual sustained short-circuit protection)
Digital inputs	
Number of Inputs	3 counter inputs
Wiring	Sink
Input Voltage Minimum Nominal Maximum	 18 VDC 24 VDC 30 VDC
Switching Threshold LOW Range HIGH Range	 <5 V >15 V
Input Delay	Max. 3 μs
Input Current at Nominal Voltage	Approx. 6 mA
Incremental Encoder Operation Signal Form Evaluation Input Frequency Counter Frequency Count Size Input 1 Input 2 Input 3	 Square wave pulses 4-fold, cyclic counter 50 kHz 200 kHz 32 Bit Channel A Channel B Ref

Module ID	CM411

Event Counter Operation
   Signal Form
   Evaluation
   Input Frequency
   Counter Frequency
   Count Size
   Input 1
   Input 2
	
	Square wave pulses
	Each edge, cyclic counter
	50 kHz
	100 kHz
	2 x 16 Bit
	Counter 1
	Counter 2

Comparator
   Evaluation
 	
   Comparator Output
   Reaction Time
	
	Actual value comparison of the counter value during incremental encoder operation
	or of counter 2 during event counter operation (window comparator)
	Output 1
	<100 μs

Electrical Isolation Input - Input	 No
Digital Outputs	
Number and type of outputs	2 transistor outputs
Rated Current	Max. 0.5 A
Total Output Current	Max. 1 A
Rated Voltage	24 VDC
Switching Voltage Range	18 -30 VDC
Wiring	Source
Short Circuit Protection	Yes
Overload Protection	Yes
Negative Anode Potential when Switching Off Inductive Loads	59 V

Switching Delay
   log 0 - log 1
   log 1 - log 0
	
	Max. 100 μs
	Max. 100 μs

Electrical Isolation Output - Output	 No
Mechanical Characteristics	
Dimensions	B&R 2003 single width

[1] Two logical module slots are required by the module.

[2] Referring the measurement range.

[3] Referring the current measurement value.

13.5.2 Status LEDs

LED	Description
OK	This orange LED is lit when the external supply voltage for the outputs is within the defined range (>18 VDC).
LED 1 -2, green	Logical status of the respective digital input.
LED 1 -2, orange	Control status of the respective digital output.

13.5.3 Input Circuit Diagram
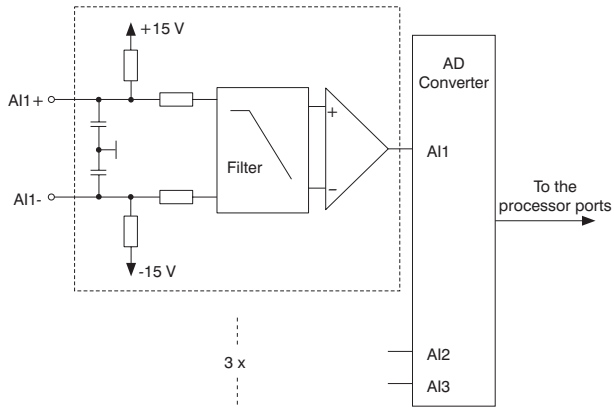
Analog Inputs
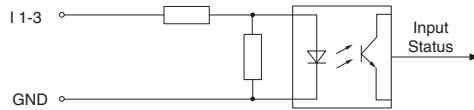
Digital inputs

13.5.4 Output Circuit Diagram
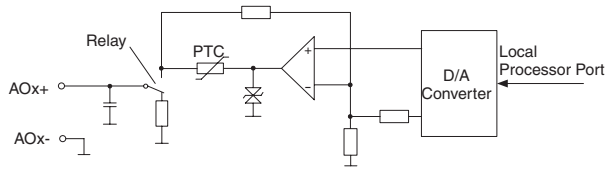
Analog Outputs

Digital Outputs
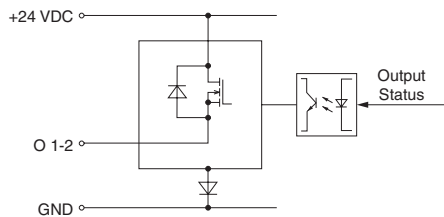
13.5.5 Supply Voltage Monitoring
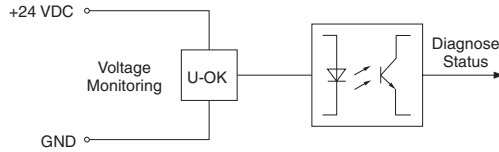
13.5.6 Module Design

General Information

The design of the combination module CM411 corresponds to two AF101 adapter modules with two screw-in modules installed.
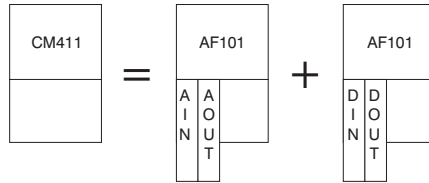
AIN	...	Analog inputs
AOUT	...	Analog outputs
DIN	...	Digital inputs (counter inputs)
DOUT	...	Digital outputs

Module Addresses

Because of this special module design, the combination module CM411 requires two module addresses. A CPU, a combination module CM411 and a digital input module DI435 are used in the example shown below. The module address assignments are to be made as shown in the diagram.

Module Address	
1 + 2	3

CPU	CM411	DI435

Variable Declaration

To avoid conflicts in the register, the settings listed below must be used in the variable declaration for the module address and for the slot.
In this case, the module is accessed with module addresses 1 and 2.

CM411				
Module Address 1			Module Address 2	
Slot 1	Slot 2		Slot 1	Slot 2
AIN	AOUT		DIN	DOUT

13.5.7 Timing
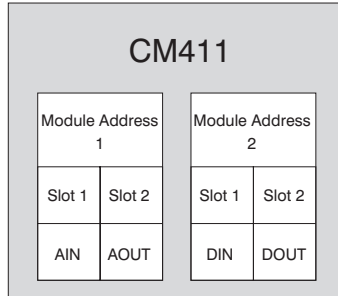
The following three factors must be taken into consideration for timing when a B&R 2003 CPU is used as controller:

- Internal Bus Cycle
- I/O AF Cycle
- I/O CPU Load

Internal bus cycle

I/O AF cycle

Internal Bus Cycle

All combination modules, AF modules and digital I/O modules are processed during this time. The internal bus cycle for a CM411 is calculated as follows:

There is no AF101 adapter module on the bus

$t_{int\_cycle} = n * 44\,\mu s * 6 + 6 * 120\,\mu s + 1200\,\mu s = 2184\,\mu s$ (n = 1)

n .... Number of CM411 modules

44 μs .... Time for a combination module CM411

6 .... Number of data words for a CM411

120 μs ...... Combination module CM411 busy

1200 μs .... Offset

There is an AF101 adapter module on the bus or a CPx74 is used as CPU

$t_{int\_cycle} = n * 44\,\mu s * 6 + 6 * 200\,\mu s + 1200\,\mu s = 2664\,\mu s$ (n = 1)

n .... Number of CM411 modules

44 μs .... Time for a combination module CM411

6 .... Number of data words for a CM411

200 μs ...... AF101 or CPx74 busy

1200 μs .... Offset

I/O-AF Cycle

During this time, all data points on the combination module CM411 are updated or read in internally.

$t_{IO\_AF} \leq 1$ ms

I/O CPU Load

This time determines how long the CPU requires to process the I/O data passed on by the combination module CM411. The CPU is loaded considerably by the analog I/O data.

A CP430 or CPx70 is used as CPU

$t_{IO\_CPU} = 6 * 100\,\mu s = 600\,\mu s$

6 .... Number of data words for a CM411

100 μs ...... Analog data point on CP430 or CPx70

A CPx74 is used as CPU

$t_{IO\_CPU} = 6 * 70\,\mu s = 420\,\mu s$

6 .... Number of data words for a CM411

70 μs .... Analog data point on CPx74

A CP476 is used as CPU

$t_{IO\_CPU} = 6 * 50\,\mu s = 300\,\mu s$

6 .... Number of data words for a CM411

50 μs .... Analog data point on CP476

Task Class

Fastest Task Class Recommended: 4 ms

13.5.8 Legend Sheets

A legend sheet can be slid into the front of the module from above. The module circuit is shown on the back. The inputs/outputs can be labelled on the front.
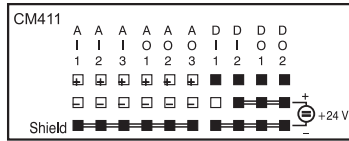
13.5.9 Connections
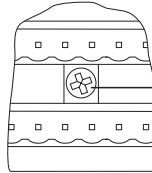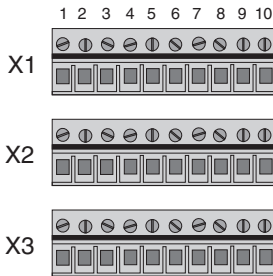
TB710

Tighten ground screw

X1	
Analog Inputs	
1	+ Input AI1
2	+ Input AI2
3	+ Input AI3
Analog Outputs	
4	+ Output AO1
5	+ Output AO2
6	+ Output AO3
Digital Inputs	
7	Input DI1
8	Input DI2
Digital Outputs	
9	Output DO1
10	Output DO2

X2	
Analog Inputs	
1	- Input AI1
2	- Input AI2
3	- Input AI3
Analog Outputs	
4	Ground AO1-
5	Ground AO2-
6	Ground AO3-
Digital Inputs/Outputs	
7	DI3
8	+24 VDC
9	+24 VDC
10	+24 VDC

X3	
Analog Inputs / Outputs	
1	Shield
2	Shield
3	Shield
4	Shield
5	Shield
6	Shield
Digital Inputs / Outputs	
7	Ground
8	Ground
9	Ground
10	Ground

13.5.10 Connection Example Analog Inputs

Measurement signal

1   2   3   4   5   6   ...

AI1+

AI1-

Shield

13.5.11 Connection Example Analog Outputs

Load

1   2   3   4   5   6   ...

AO1+

AO1-

Shield
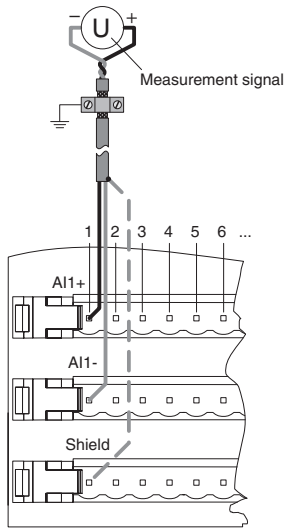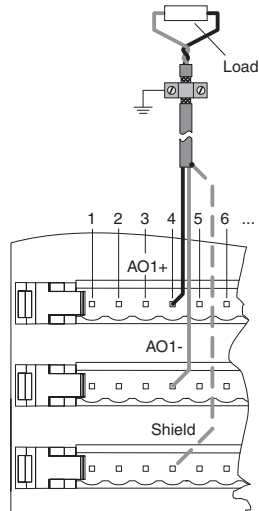
13.5.12 Connection Example Incremental Encoder Operation
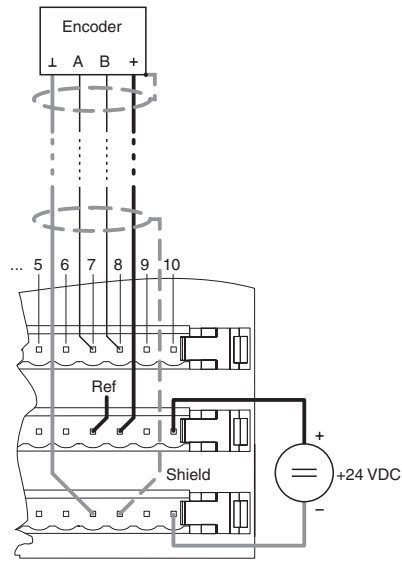
13.5.13 Connection Example Event Counter Operation
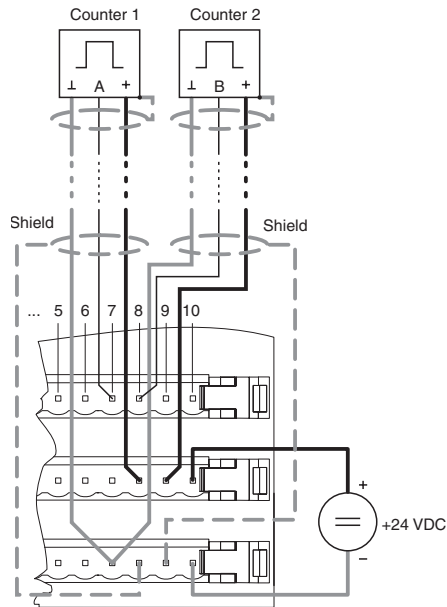
13.5.14 Connection Example Digital Outputs

13.5.15 Variable Declaration for the Analog Inputs

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for the analog inputs.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	INT16	Analog In	1	●		Analog input value channel 1
Data word 1	INT16	Analog In	2	●		Analog input value channel 2
Data word 2	INT16	Analog In	3	●		Analog input value channel 3
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. IN	28	●		Module type

Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

In the example below, the combination module CM411 is accessed with module addresses 1 and 2.

Analog input data cannot be packed. Only the first object from this group of four will be created and sent.

Slot	CAN ID [1]	Word 1		Word 2		Word 3		Word 4
1	542	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Not used
2	543	Not used						
3	544	Not used						
4	545	Not used						

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

  nd ..... Node number of the CAN slave = 1

  ma .... Module address = 1

  sl .... Slot number = 1

⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

Description of Data and Configuration Words

Data word 0, 1, 2 (read)

16 bit standardized voltage value.

Configuration Word 12 (read)

Configuration word 12 contains the module status.

	Bit	Description
	12 - 15	x ....Not defined, masked out
	11	0....Converter values ready 1....Converter values not yet ready
	3 - 10	x ....Not defined, masked out
	2	0....Channel 3: No error 1....Channel 3: Error present
	1	0....Channel 2: No error 1....Channel 2: Error present
	0	0....Channel 1: No error 1....Channel 1: Error present

x	x	x	x		x	x	x	x	x	x	x	x			

15   8 7   0

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

		Bit	Description
		8 - 15	Module code = $32
		0 - 7	x ....Not defined, masked out

```
0 0 1 1 0 0 1 0 x x x x x x x x
15          8 7          0
```

13.5.16 Variable Declaration for the Analog Outputs

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for the analog outputs.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	INT16	Analog Out	1		●	Analog output value channel 1
Data word 1	INT16	Analog Out	2		●	Analog output value channel 2
Data word 2	INT16	Analog Out	3		●	Analog output value channel 3
Configuration word 14	WORD	Transp. In	28	●		Module type

Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

In the example below, the combination module CM411 is accessed with module addresses 1 and 2.

Analog output data cannot be packed. Only the second object from this group of four will be created.

Slot	CAN ID [1]	Word 1		Word 2		Word 3		Word 4
1	1054	Not used						
2	1055	Chan. 1L	Chan. 1H	Chan. 2L	Chan. 2H	Chan. 3L	Chan. 3H	Not used
3	1056	Not used						
4	1057	Not used						

[1] CAN ID = 1054 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address = 1

sl .... Slot number = 2

> ⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

Description of Data and Configuration Words

Data word 0, 1, 2 (write)

The 16 bit standardized values for voltage are written to the module output channel.

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $34
	0 - 7	x ....Not defined, masked out

0	0	1	1	0	1	0	0	x	x	x	x	x	x	x	x

15   8 7   0

13.5.17 Variable Declaration for Incremental Encoder Operation

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support:　　　See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Incremental encoder operation with PCC 2003 CPU and remote slaves

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for the digital inputs in incremental encoder operation.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	WORD	Transp. In	0	●		Module status
Data word 1	INT32	Transp. In	2	●		Counter value
Configuration word 4	INT32	Transp. Out	8		●	Threshold value 1
Configuration word 6	INT32	Transp. Out	12		●	Threshold value 2
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

Incremental Encoder Operation with CAN Slaves

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for the digital inputs in incremental encoder operation.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	INT32	Transp. In	0	●		Counter value
Data word 2	WORD	Transp. In	4	●		Module status
Configuration word 4	INT32	Transp. Out	8		●	Threshold value 1
Configuration word 6	INT32	Transp. Out	12		●	Threshold value 2
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

⚠ **B&R 2000 users have to exchange the two counter status words so that the high word is first (Motorola format)!**

Access Using CAN IDs

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer. Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

In the example below, the combination module CM411 is accessed with module addresses 1 and 2.

Digital input data cannot be packed. Only the first object from this group of four will be created and sent.

Slot	CAN ID [1]	Word 1		Word 2		Word 3		Word 4
1	546	Count. LL	Count. ML	Count. MH	Count. HH	Status L	Status H	Not used
2	547	Not used						
3	548	Not used						
4	549	Not used						

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address = 2

sl .... Slot number = 1

⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

Description of Data and Configuration Words

Data Word 0 (read)

Data word 0 includes the module status time constant for the counter value.

	Bit	Description
	8 - 15	x ....Not defined, masked out
	7	0 ....Referencing is taking place 1 ....Counter is referenced (resetting takes place when the reference command is received)
	6	Changes state each time referencing takes place
	5	0 ....Supply voltage <18 V 1 ....Supply voltage >18 V, Outputs OK
	4	Output status of the comparator
	3	Level of encoder input A
	2	Level of encoder input B
	1	x ....Not defined, masked out
	0	Level of the reference pulse

x	x	x	x	x	x	x	x							x	

15   8 7   0

Data Word 1 (read)

Counter Value MSW

Data Word 2 (read)

Counter Value LSW

Configuration Words 4+5 (write)

Threshold value 1 (32 Bit)

Number format: 32 bit with sign

Bit 10 in configuration word 14 (write) is set to 0.
Threshold value 1 must always be $\leq$ threshold value 2 .
Threshold values are internally arranged in increasing order including sign.

Number format: 32 bit without sign - cyclic operation

Bit 10 in configuration word 14 (write) is set to 1.
The threshold values are not placed in order internally. The sign is not used in the comparator calculation.

Configuration Words 6+7 (write)

Threshold value 2 (32 Bit)

Configuration Word 12 (read)

Configuration word 12 contains the module status (current status unlatched). The module status is written to data word 0.

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $36
	0 - 7	x ....Not defined, masked out

0	0	1	1	0	1	1	0	x	x	x	x	x	x	x	x

15   8 7   0

Configuration Word 14 (write)

The module is configured using configuration word 14.

	Bit	Description
	13 - 15	0
	12	0 .... Incremental encoder operation
	11	0 .... No effect on count direction 1 .... Count direction inverted as compared to counter wiring
	10	0 .... Number format: 32 bit with sign 1 .... Number format: 32 bit without sign - cyclic operation In continuous operation, the internal order of the threshold values are kept. If a counter overflow occurs, the behavior of the comparator does not have to be changed. Comparator operation takes place without consideration of the sign.
	8 - 9	0 .... Incremental encoder operation with 4-fold evaluation
	7	0 .... No effect on reference pulse 1 .... Reference pulse is inverted. This setting is used for encoders with a high pulse.
	6	0 .... Set counter immediately to 0. In data word 0 (module status), bit 7 is immediately set to 1 and the counter is cleared. 1 .... Counter remains functioning. In data word 0 (module status), bit 7 is immediately set to 0 (referencing required).
	5	0
	4	0 .... No effect on counter 1 .... Clear counter (reference)
	3	0 .... Comparator off Output 1 is handled as defined in the variable declaration for digital outputs. 1 .... Comparator on
	2	0 .... The comparator output is set to the level given in bit 0, if threshold value 1 ≤ counter ≤ threshold value 2 1 .... The comparator output is set to the inverted level of bit 0, if threshold value 1 ≤ counter ≤ threshold value 2
	1	0
	0	Level of the comparator output

0	0	0	0		0	0		0				0			

15   8 7   0

13.5.18 Variable Declaration for Event Counter Operation

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for the digital inputs in event counter operation.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	WORD	Transp. In	0	●		Module status
Data word 1	WORD	Transp. In	2	●		Counter value of counter 1
Data word 2	WORD	Transp. In	4	●		Counter value of counter 2
Configuration word 5	WORD	Transp. Out	10		●	Threshold value 1 for counter 2
Configuration word 7	WORD	Transp. Out	14		●	Threshold value 2 for counter 2
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

In the example below, the combination module CM411 is accessed with module addresses 1 and 2.

Digital input data cannot be packed. Only the first object from this group of four will be created and sent.

Slot	CAN ID [1]	Word 1		Word 2		Word 3		Word 4
1	546	Count. 2L	Count. 2H	Count. 1L	Count. 1H	Status L	Status H	Not used
2	547	Not used						
3	548	Not used						
4	549	Not used						

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

  nd ..... Node number of the CAN slave = 1

  ma .... Module address = 2

  sl .... Slot number = 1

> ⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

Description of Data and Configuration Words

Data Word 0 (read)

Data word 0 includes the module status time constant for both counter values.

	Bit	Description
	6 - 15	x ....Not defined, masked out
	5	0....Supply voltage <18 V 1....Supply voltage >18 V, Outputs OK
	4	Output status of the comparator
	3	Level of encoder input A: Counter 1
	2	Level of encoder input B: Counter 2
	0 - 1	x ....Not defined, masked out

x	x	x	x	x	x	x	x	x	x					x	x

15   8 7   0

Data Word 1 (read)

Counter value of counter 1.

Data Word 2 (read)

Counter value of counter 2.

Configuration Word 5 (write)

Threshold value 1 (16 Bit) for counter 2.

Number format: 16 bit without sign

Bit 10 in configuration word 14 (write) is set to 0.
Threshold value 1 must always be $\leq$ threshold value 2 .
Threshold values are internally arranged in increasing order. The sign is ignored.

Number format: 16 bit without sign - cyclic operation

Bit 10 in configuration word 14 (write) is set to 1.
The threshold values are not placed in order internally. The sign is ignored.

Configuration Word 7 (write)

Threshold value 2 (16 Bit) for counter 2.

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $36
	0 - 7	x ....Not defined, masked out

0	0	1	1	0	1	1	0	x	x	x	x	x	x	x	x

15   8 7   0

Configuration Word 14 (write)

The module is configured using configuration word 14.

	Bit	Description
	13 - 15	0
	12	1.... Event counter operation
	11	0
	10	0.... Number format: 16 bit without sign
		1.... Number format: 16 bit without sign - cyclic operation
		In continuous operation, the internal order of the threshold values are kept. If a counter overflow occurs, the behavior of the comparator does not have to be changed.
	9	0
	8	0.... Counter 1 and 2: Only count positive edges 1.... Counter 1 and 2: Count both edges
	6 - 7	0
	5	0.... No effect on counter 2 1.... Immediately clear counter 2
	4	0.... No effect on counter 1 1.... Immediately clear counter 1
	3	0.... Comparator off
		Output 1 is handled as defined in the variable declaration for digital outputs.
		1.... Comparator on
	2	0.... The comparator output is set to the level given in bit 0, if threshold value 1 ≤ counter ≤ threshold value 2
		1.... The comparator output is set to the inverted level of bit 0, if threshold value 1 ≤ counter ≤ threshold value 2
	1	0
	0	Level of the comparator output

```
0 0 0 1 0   0   0 0         0
15          8 7             0
```

13.5.19 Variable Declaration for the Digital Outputs

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for the digital outputs.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	WORD	Transp. Out	0		●	Digital outputs 0 and 1
Configuration word 14	WORD	Transp. In	28	●		Module type

Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer. Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

In the example below, the combination module CM411 is accessed with module addresses 1 and 2.

Digital output data cannot be packed. Only the second object from this group of four will be created.

Slot	CAN ID [1]	Word 1		Word 2	Word 3	Word 4
1	1058	Not used				
2	1059	Output L	Output H	Not used		
3	1060	Not used				
4	1061	Not used				

[1] CAN ID = 1054 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

  nd ..... Node number of the CAN slave = 1

  ma .... Module address = 2

  sl .... Slot number = 2

> ⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

Description of Data and Configuration Words

Data Word 0 (write)

State of the digital outputs 0 and 1. Bits 2 to 15 must be 0.

	Bit	Description
	2 - 15	0
	1	Digital output 1
	0	Digital output 0
		Status of output 0, if the comparator is switched off.

0	0	0	0	0	0	0	0	0	0	0	0	0	0		

15   8 7   0

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $38
	0 - 7	x ....Not defined, masked out

0	0	1	1	1	0	0	0	x	x	x	x	x	x	x	x

15   8 7   0

13.6 ME010

13.6.1 Technical Data

Module ID	ME010
General Information	
Model Number	7ME010.9
Short Description	2003 expansion for the CP476, 1 PCMCIA slot
C-UL-US Listed	in preparation
B&R ID Code	$00
Module Type	B&R 2003 Expansion for CP476
Slot	Can be installed on the far left of the system bus next to the CP476
Power Consumption	0.35 W, without PCMCIA memory card
Peripheral	

PCMCIA Interface
    Standard
    Card Height
    Card Type
    Memory Size
      SRAM
      FlashPROM
	1
	JEIDA V 4.0 or PCMCIA Standard Release 2.0
	Max. 3 mm
	Memory cards
	
	Max. 4 MByte
	Max. 4 MByte

Status Display	2 LEDs
Mechanical Characteristics	
Module Width	20 mm

13.6.2 General Information

The CP476 now provides another expansion possibility on the far left of the system bus. Special expansion modules can be installed here. One of these is the ME010.

The ME010 provides a PCMCIA slot for B&R memory cards. This expansion can be used to update the controller. Another possibility is using the PCMCIA cards as external recipe memory for these types of applications.

In principle, it is possible to use memory cards as "normal" memory areas, but we strongly advice against this because the PCMCIA cards are not mechanically fixed in place and therefore are not vibration resistant. Access of memory cards is also very slow.
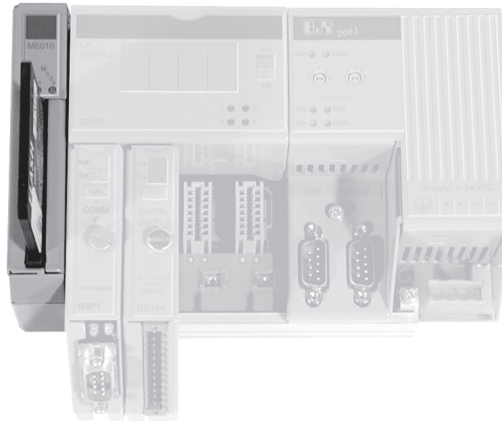
13.6.3 Status LEDs

LED	Description
Card	Indicates access of the PCMCIA memory card.
Mode	Indicates that a memory card is being programmed and therefore is not allowed to be removed.

13.6.4 PCMCIA Interface

General Information

The ME010 is equipped with a PCMCIA interface.
PCMCIA memory cards conforming to JEIDA V 4.0 Type I or PCMCIA Standard Release 2.0 (max. 3 mm high) are supported.
The ME010 supports memory cards with up to 4 MByte SRAM or with up to 4 MByte FlashPROM. The following memory card can be ordered from B&R:

Model Number	Short Description	Power Consumption
0MC111.9	PCMCIA Memory Card, 2 MB FlashPROM	Max. 0.8 W
0MC211.9	PCMCIA Memory Card, 2 MB SRAM	Max. 0.8 W

The memory cards are used by the CP476 as ROM Type "MEMCARD".

Limitations when using memory cards:

- Internal variables cannot be stored on the memory card
- memory cannot be allocated on the memory cards
- the data format is not compatible to the B&R SYSTEM 2005 CPU CP260

The SRAM and FlashPROM memory cards can only be written to by the CP476. Therefore it is not possible to program system software or the application on a memory card directly on a PC with a PCMCIA interface.

Operating Modes

The CP476 is has two hex switches which are used as CAN node number switches. Some switch settings are reserved for special operating modes.

Switch Settings CP476	Operating Mode	Description
00	Bootstrap Loader	In this switch position, the operating system can be programmed via the online interface. User Flash is deleted after the update begins.
FD	Update	In this switch position, the CPU checks if an update memory card is inserted. If not, the CPU goes into SERVICE mode. Otherwise the operating system, System ROM, User ROM and, if available, FIX RAM on the CP476 is deleted and reinstalled from the memory card. If an error occurs during the installation, the ERROR and BAT LEDs blink. During error free installation, the READY and RUN LEDs blink.
FF	Diagnosis	The CPU boots in Diagnosis mode. The program sections in User RAM and User FlashPROM are not initialized. After diagnosis mode, the CPU always boots with a cold start.

13.7 ME020

13.7.1 Technical Data

The IF671 interface module insert shown in the picture is not included in the delivery.

Module ID	ME020
General Information	
Model Number	7ME020.9
Short Description	2003 expansion for the CP476, 1 PCMCIA slot, 1 slot interface module inserts
C-UL-US Listed	in preparation
B&R ID Code	$02
Module Type	B&R 2003 Expansion for CP476
Slot	Can be installed on the far left of the system bus next to the CP476
Power Consumption	0.7 W, without PCMCIA memory card and without interface module insert
Peripheral	

PCMCIA Interface
  Standard
  Card Height
  Card Type
  Memory Size
    SRAM
    FlashPROM
	1
	JEIDA V 4.0 or PCMCIA Standard Release 2.0
	Max. 3 mm
	Memory cards
	
	Max. 4 MByte
	Max. 4 MByte

Status Display	2 LEDs
Insert Slots	1 (for interface module inserts)
Mechanical Characteristics	
Module Width	37 mm

13.7.2 General Information

The CP476 now provides another expansion possibility on the far left of the system bus. Special expansion modules can be installed here. One of these is the ME020.

The ME020 provides a PCMCIA slot for B&R memory cards. This expansion can be used to update the controller. Another possibility is using the PCMCIA cards as external recipe memory for these types of applications.

In principle, it is possible to use memory cards as "normal" memory areas, but we strongly advice against this because the PCMCIA cards are not mechanically fixed in place and therefore are not vibration resistant. Access of memory cards is also very slow.

Additionally, the ME020 provides a slot for B&R SYSTEM 2005 interface module inserts.
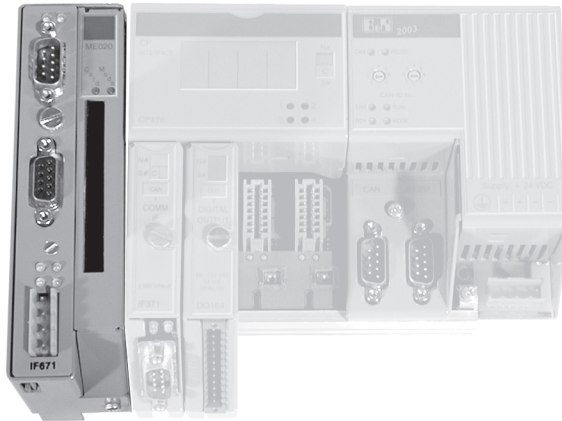
13.7.3 Status LEDs

LED	Description
Card	Indicates access of the PCMCIA memory card.
Mode	Indicates that a memory card is being programmed and therefore is not allowed to be removed.

13.7.4 PCMCIA Interface

General Information

The ME020 is equipped with a PCMCIA interface.
PCMCIA memory cards conforming to JEIDA V 4.0 Type I or PCMCIA Standard Release 2.0 (max. 3 mm high) are supported.
The ME020 supports memory cards with up to 4 MByte SRAM or with up to 4 MByte FlashPROM. The following memory card can be ordered from B&R:

Model Number	Short Description	Power Consumption
0MC111.9	PCMCIA Memory Card, 2 MB FlashPROM	Max. 0.8 W
0MC211.9	PCMCIA Memory Card, 2 MB SRAM	Max. 0.8 W

The memory cards are used by the CP476 as ROM Type "MEMCARD".

Limitations when using memory cards:

- Internal variables cannot be stored on the memory card
- memory cannot be allocated on the memory cards
- the data format is not compatible to the B&R SYSTEM 2005 CPU CP260

The SRAM and FlashPROM memory cards can only be written to by the CP476. Therefore it is not possible to program system software or the application on a memory card directly on a PC with a PCMCIA interface.

Operating Modes

The CP476 is has two hex switches which are used as CAN node number switches. Some switch settings are reserved for special operating modes.

Switch Settings CP476	Operating Mode	Description
00	Bootstrap Loader	In this switch position, the operating system can be programmed via the online interface. User Flash is deleted after the update begins.
FD	Update	In this switch position, the CPU checks if an update memory card is inserted. If not, the CPU goes into SERVICE mode. Otherwise the operating system, System ROM, User ROM and, if available, FIX RAM on the CP476 is deleted and reinstalled from the memory card. If an error occurs during the installation, the ERROR and BAT LEDs blink. During error free installation, the READY and RUN LEDs blink.
FF	Diagnosis	The CPU boots in Diagnosis mode. The program sections in User RAM and User FlashPROM are not initialized. After diagnosis mode, the CPU always boots with a cold start.

13.7.5 Interface Module Inserts

The following B&R SYSTEM 2005 interface module inserts can be operated with the ME020 module:

Model Number	Short Description
3IF613.9	Three RS232 interfaces
3IF621.9	One RS485/RS422 interface and one CAN interface
3IF622.9	One RS232 interface and two RS485/RS422 interfaces
3IF661.9	One RS485 interface (PROFIBUS-DP Slave)
3IF671.9	One RS232 interface, one RS485/RS422 interface and one CAN interface
3IF672.9	One RS232 interface and two CAN interfaces
3IF681.95	One RS232 interface and one ETHERNET interface with 10 BASE2 connection (CHEAPERNET BNC socket)
3IF681.96	One RS232 interface and one ETHERNET interface with 10 BASE-T connection (Twisted Pair / RJ45 socket)

13.7.6 Operation of Interface Module Inserts

The operation is software compatible to the B&R SYSTEM 2005. Existing software can be used, e.g. NET2000, Frame driver, etc.

14 COMMUNICATION MODULES

14.1 OVERVIEW

Module	Description
IF311	2003 interface module, 1 RS232 interface, screw-in module
IF321	2003 interface module, 1 RS485/RS422 interface, electrically isolated, network capable, screw-in module
IF361	2003 interface module, 1 RS485 interface, electrically isolated and network capable, transfer protocol: PROFIBUS-DP, screw-in module
IF371	2003 interface module, 1 CAN interface, electrically isolated, network capable, screw-in module

14.2 IF311 / IF321

14.2.1 Technical Data

Module ID	IF311	IF321
General Information		
Model Number	7IF311.7	7IF321.7
Short Description	2003 interface module, 1 RS232 interface, screw-in module	2003 interface module, 1 RS485/RS422 interface, electrically isolated, network capable, screw-in module
C-UL-US Listed	Yes	Yes
B&R ID Code	$18	$16
Module Type	B&R 2003 screw-in module	
Slot	CP Interface, slots 1, 2 and 3	
Power Consumption	Max. 0.5 W max. 1.6 W with P120 / P121 [1]	Max. 1.4 W
Standard Communication Interface		
Interface Type	RS232	RS485/RS42 [2]
Electrical Isolation Interface - PCC	No	Yes
Design	9 pin D-type plug	9 pin D-type socket
Handshake Lines	RTS, CTS	———
Baudrate [2]	Max. 115.2 kBaud	Max. 115.2 kBaud
Distance	15 m at 19200 Baud	1200 m (without repeater)
Network Capable	No	Yes
Data Formats Data Bits Parity Stop Bits	5 to 8 yes / no / even / odd 1 / 2	7 / 8 yes / no / even / odd 1 / 2
Mechanical Characteristics		
Dimensions	B&R 2003 screw-in module	

[1] Integrated power supply on pin 4 of the RS232 interface for simple PANELWARE controllers, e.g. P120.
[2] Can be configured using software.

14.2.2 Pin Assignments IF311

9 pin D-type plug
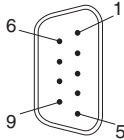
Assignment		
Pin	RS232	
1	NC	Reserved
2	RXD	Receive Signal
3	TXD	Transmit Signal
4	+5 VDC /max. 500 mA	Panel Supply
5	GND	Ground
6	NC	Reserved
7	RTS	Request To Send
8	CTS	Clear To Send
9	NC	Reserved

14.2.3 Pin Assignments IF321

9 pin D-type socket
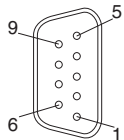
Assignment		
Pin	RS485	RS422
1	NC	NC
2	NC	TXD
3	DATA	RXD
4	NC	NC
5	GND	GND
6	+5 VDC / 50 mA	+5 VDC / 50 mA
7	NC	n_TXD
8	n_DATA	n_RXD
9	NC	NC

14.2.4 Bus Termination Resistance IF321

The IF321 is equipped with a bus termination resistor. The resistance is turned on and off with a switch.

Off

ON

Using the remote bus connector 0G1000.00-090 is possible.

Wiring remote I/O networks is described in Chapter 2 "Project Planning and Installation", section "Remote I/O Bus".

14.3 IF361

14.3.1 Technical Data

Module ID	IF361
General Information	
Model Number	7IF361.70-1
Short Description	2003 interface module, 1 RS485 interface, electrically isolated and network capable, Transfer protocol: PROFIBUS-DP, screw-in module
C-UL-US Listed	in preparation
B&R ID Code	$25
Module Type	B&R 2003 screw-in module
Slot	CP Interface, slots 1, 2 and 3
Power Consumption	Max. 2.6 W
Standard Communication Interface	
Interface Type	RS485
Controller	ASIC SPC3
RAM	1.5 KByte
Transfer Protocol	PROFIBUS-DP
Design	9 pin D-type socket
Electrical Isolation	Yes
Maximum Distance	1000 m
Protective Circuit	Yes

Maximum Baudrate
   Bus Length <100 m
   Bus Length <200 m
   Bus Length <400 m
   Bus Length <1000 m
	
	12 MBit/s
	1.5 MBit/s
	500 kBit/s
	187.5 kBit/s

Network Capable	Yes
Bus Termination Resistance	External T-connector
Mechanical Characteristics	
Dimensions	B&R 2003 screw-in module

14.3.2 Pin Assignments

9 pin D-type socket
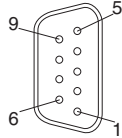
Assignment	
Pin	**PROFIBUS-DP**
1	
2	
3	B (Data)
4	RTS
5	GND
6	+5 VDC / 50 mA
7	
8	A (Data\)
9	

14.4 IF371

14.4.1 Technical Data

Module ID	IF371
General Information	
Model Number	7IF371.70-1
Short Description	2003 interface module, 1 CAN interface, electrically isolated, network capable, screw-in module
C-UL-US Listed	Yes
B&R ID Code	$24
Module Type	B&R 2003 screw-in module
Slot	CP Interface, slots 1, 2 and 3
Power Consumption	Max. 2.5 W
Standard Communication Interface	
Interface Type	CAN
Electrical Isolation	Interface - PCC
Design	9 pin D-type plug
Interface LEDs	RXD and TXD
Maximum Distance	1000 m

Maximum Baudrate
    Bus Length 10 - 60 m
    Bus Length 100 - 200 m
    Bus Length 800 - 1000 m
	Max. 500 kBit/s
	Max. 250 kBit/s
	Max. 50 kBit/s

Network Capable	Yes
Mechanical Characteristics	
Dimensions	B&R 2003 screw-in module

14.4.2 Pin Assignments

9 pin D-type plug
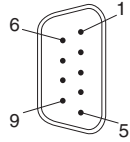
Assignment	
Pin	**CAN**
1	NC
2	CAN_L
3	CAN_GND
4	NC
5	NC
6	Reserved
7	CAN_H
8	NC
9	NC

Chapter 3
B&R 2003 Modules

15 COUNTER AND POSITIONING MODULES

15.1 OVERVIEW

Module	Description
NC161	2003 encoder module, input frequency 100 kHz, incremental or absolute, 32 Bit, encoder supply 5 VDC or 24 VDC, screw-in module

15.2 NC161

15.2.1 Technical Data

Module ID	NC161
General Information	
Model Number	7NC161.7
Short Description	2003 encoder module, input frequency 100 kHz, incremental or absolute, 32 bit, encoder supply 5 VDC or 24 VDC, screw-in module
C-UL-US Listed	Yes
B&R ID Code	$10
Module Type	B&R 2003 screw-in module
Slot	AF101 adapter module, CP interface
Power Consumption	$0.3 \text{ W} + I_{encoder} * 5.4 \text{ V}$
Encoder Input	
General Information	Connection using 15 pin D-type socket incremental or SSI absolute encoder (both electrically isolated)
Incremental Encoder Signal Form Evaluation Input Frequency Counter Frequency Phase Shift between Channel A and B Counter Size Inputs Input Level	Square wave pulses 4-fold Max. 100 kHz Max. 400 kHz $90° \pm 15°$ 32 Bit A, A\, B, B\, R, R\ 5 V (differential input)
SSI Absolute Encoder Coding Word Size Baudrate Data Input Level Clock Output Level Max. Signal Delay Clock - Data	Gray, Binary Max. 31 Bit 100 kBaud 5 V (differential signal) 5 V (differential signal) $\leq 2.5 \mu s$

Module ID	NC161
Additional Inputs +24 VDC	
Reference Enable Switch Electrical Isolation	Connected using terminal block Yes
Reference Pulse Electrical Isolation	Connected using 15 pin D-type socket (pin 10 and 11) Yes
Encoder Supply	
Output Voltage Protection	+5 VDC / max. 500 mA without external feed Short Circuit and Overload Protection
External Supply Voltage Protection	+24 VDC / max. 300 mA Short circuit protection
Mechanical Characteristics	
Dimensions	B&R 2003 screw-in module

15.2.2 General Information

The NC161 is an encoder module with symmetrical incremental encoder or absolute encoder evaluation. A 5 V encoder supply is available directly on the module. A 24 V encoder supply must be connected externally to terminals 1 and 2 on the terminal block. The reference enable switch is connected to terminals 3 and 4 .

15.2.3 Operating Modes

The NC161 is used for single axis positioning with ramps in conjunction with the AO352 analog output module.

15.2.4 Special Functions

- Latches the counter value using the reference enable switch
- Enables use of a comparator output when during incremental encoder operation. The clock output (pins 7 and 8) is used as the comparator output.

15.2.5 Encoder Connection

15 pin D-type socket
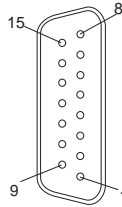
Pin	ID	Incremental Encoder	ID	Absolute Encoder	ID	Additional Input	ID	Encoder Supply
1	A	Channel A						
2	A\	A inverted						
3	B	Channel B						
4	B\	B inverted						
5	R	Reference pulse [1]	D	Data input				
6	R\	R inverted [1]	D\	D inverted				
7			C	Clock output				
8			C\	C inverted				
9							ES	Encoder supply [2] +5 VDC / 500 mA
10						Reference pulse +24 VDC [1]		
11						Reference pulse GND [1]		
12							EG	Encoder supply ⊥
13							ES	Encoder supply [3] +24 VDC / 300 mA external voltage from terminal block
14								
15								

[1] Pins 5 and 6 as well as 10 and 11 are linked together using OR logic (hardware). Open pins are considered 0.

[2] The 5 V encoder supply is not supplied externally. Bit 7 of configuration word 8 must be set in order to activate the 5 V encoder supply.

[3] In order to use an encoder requiring a 24 V power supply, terminals 1 and 2 on the terminal block must be connected to an external 24 V encoder supply.

15.2.6 Terminal Blocks

The terminal block is found on the bottom of the module. It is integrated in the housing.

Pin Assignments
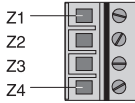
Pin	Assignment
1	Feed for external encoder supply max. +24 VDC
2	GND
3	Input reference enable switch +24 VDC
4	Input reference enable switch GND

Connection Example

15.2.7 Variable Declaration for Incremental Encoder Operation

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Incremental encoder operation with PCC 2003 CPU and remote slaves

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for this module.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	WORD	Transp. In	0	●		Module status
Data word 1	INT32	Transp. In	2	●		Counter value
Configuration word 4	INT32	Transp. In	8	●		Counter value at positive edge of the reference enable switch
	INT32	Transp. Out	8		●	Threshold value 1
Configuration word 6	INT32	Transp. In	12	●		Counter value at negative edge of the reference enable switch
	INT32	Transp. Out	12		●	Threshold value 2
Configuration word 8	WORD	Transp. Out	16		●	Incremental encoder / comparator control
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

Incremental Encoder Operation with CAN Slaves

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for this module.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	INT32	Transp. In	0	●		Counter value
Data word 2	WORD	Transp. In	4	●		Module status
Configuration word 4	INT32	Transp. In	8	●		Counter value at positive edge of the reference enable switch
	INT32	Transp. Out	8		●	Threshold value 1
Configuration word 6	INT32	Transp. In	12	●		Counter value at negative edge of the reference enable switch
	INT32	Transp. Out	12		●	Threshold value 2
Configuration word 8	WORD	Transp. Out	16		●	Incremental encoder / comparator control
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

⚠ **B&R 2000 users have to exchange the two counter status words so that the high word is first (Motorola format)!**

Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

Data cannot be packed on the NC161. Therefore one CAN object is transferred per screw-in module.
If an adapter module AF101 is equipped with a four NC161 modules, the CAN object has the following structure:

Slot	CAN ID [1]	Word 1		Word 2		Word 3		Word 4
1	542	Count. LL	Count. ML	Count. MH	Count. HH	Status L	Status H	Free
2	543	Count. LL	Count. ML	Count. MH	Count. HH	Status L	Status H	Free
3	544	Count. LL	Count. ML	Count. MH	Count. HH	Status L	Status H	Free
4	545	Count. LL	Count. ML	Count. MH	Count. HH	Status L	Status H	Free

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

  nd ..... Node number of the CAN slave = 1

  ma .... Module address of the AF101 = 1

  sl .... Slot number of the screw-in module on the AF101 (1 - 4)

⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

Description of Data and Configuration Words

Data Word 0 (read)

Data word 0 includes the module status time constant for the counter value.

	Bit	Description
	12 - 15	x....Not defined, masked out
	11	0....Counter value not taken 1....Counter value is taken with the first positive edge on the reference enable switch
	10	0....Counter value not taken 1....Counter value is taken with the first negative edge on the reference enable switch
	9	0....If bit 8 and bit 9 = 0, then the load is in the valid range or the supply is not switched on 1....5 V encoder supply overload or short-circuit
	8	0....If bit 8 and bit 9 = 0, then the load is in the valid range or the supply is not switched on 1....5 V encoder supply is not loaded
	7	0....Referencing is taking place 1....Counter is referenced (resetting takes place when the reference command is received)
	6	Changes state each time referencing takes place
	5	x....Not defined, masked out
	4	Output status of the comparator
	2 - 3	x....Not defined, masked out
	1	Level of the reference enable switch
	0	1) Bit 0 in configuration word 14 = 0 Level of the reference pulse 2) Bit 0 in configuration word 14 = 1 Level of the combination of reference pulse and reference enable switch Normal and inverted reference pulses are differentiated. Switching takes place using bit 2 in configuration word 14 (see following explanation).

x	x	x	x						x		x	x		

15   8 7   0

Normal Reference Pulse:

⚠️ **Please refer to path/status and timing diagrams in the section, "Incremental Encoder Operation".**

Bit 0 in Data Word 0 is always 1 when the level of the reference enable switch is 0.
Bit 0 only takes the level of the reference pulse when the level of the reference enable switch is 1.

Bit 0 = Reference pulse or (not reference enable switch)

Reference Pulse	Reference Enable Switch	Bit 0 in Data Word 0
0	0	1
1	0	1
0	1	0
1	1	1

Inverted Reference Pulse:

Bit 0 in data word 0 is always 1 when the level of the reference enable switch is 0.
Bit 0 only takes the level of the inverted reference pulse when the state of the reference enable switch is 1.

Bit 0 = (not reference pulse) or (not reference enable switch)

Reference Pulse	Reference Enable Switch	Bit 0 in Data Word 0
0	0	1
1	0	1
0	1	1
1	1	0

Data Word 1 (read)

Counter Value MSW

Data Word 2 (read)

Counter Value LSW

Configuration Words 4+5 (read)

After setting bit 11 in configuration word 8, the configuration words receive the latched counter value with the first positive edge of the reference enable switch. The value is also valid if bit 11 is set in data word 0. Please refer to the timing diagram "Latching the Counter Value" in the section, "Incremental Encoder Operation".

Configuration Words 4+5 (write)

Threshold value 1 (32 Bit)

Threshold value 1 must always be $\leq$ threshold value 2 .
Threshold values are internally arranged in increasing order **including sign**.

Configuration Words 6+7 (read)

After setting bit 10 in configuration word 8, the configuration words receive the latched counter value with the first negative edge of the reference enable switch. The value is also valid if bit 10 is set in data word 0. Please refer to the timing diagram "Latching the Counter Value" in the section, "Incremental Encoder Operation".

Configuration Words 6+7 (write)

Threshold value 2 (32 Bit)

Configuration Word 8 (write)

Incremental encoders and comparators are configured using configuration word 8.
During incremental encoder operation the clock output (pins 7 and 8) is used as the comparator output.

Bit	Description
12 - 15	0
11	0....Counter value not taken 1....Counter value is taken with the first positive edge on the reference enable switch (see configuration words 4 and 5) [1]
10	0....Counter value not taken 1....Counter value is taken with the first negative edge on the reference enable switch (see configuration words 6 and 7) [1]
8 - 9	0
7	0....5 V encoder supply off (default) 1....5 V encoder supply on
5 - 6	0
4	0....No effect on counter 1 Clear counter (reference) With the positive edge on bit 4, the counter is cleared depending on the control signal in configuration word 14 (write). Before further referencing, bit 4 must be reset and set again.
3	0....Comparator off The comparator output is set to the level given in bit 0. 1....Comparator on

2	0....Unconditional comparator output
	The comparator output is set to the level given in bit 0, if threshold value 1 < counter ≤ threshold value 2
	1....Conditional comparator output
	The comparator output can be controlled in two ways:
	a) Using the reference enable switch, if bit 15 is reset in configuration word 14.
	  Reference enable switch = 1
	   The comparator output is handled as with "Unconditional comparator output".
	  Reference enable switch = 0
	   The comparator output is set to the inverted level of bit 0.
	b) In TPU operation using the TPU-IN line. The line is handled using LTX functions (e.g. with LTXdo0()). TPU operation is defined by setting bit 15 in configuration word 14.
	  TPU-IN line = 1
	   The comparator output is handled as with "Unconditional comparator output".
	  TPU-IN line = 0
	   The comparator output is set to the inverted level of bit 0.

1	0
0	Level of the comparator output

```
0 0 0 0    0 0   0 0          0
15        8 7              0
```

[1]   The counter value is only taken once. Bit 10 and bit 11 must be reset for the value to be taken again. Bit 10 and bit 11 can be set again in configuration word 8 after the corresponding bit in the module status bit has gone to 0.

Configuration Word 12 (read)

Configuration word 12 contains the module status (current status unlatched). The module status is written to data word 0.

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

		Bit	Description
		8 - 15	Module code = $10
		0 - 7	x .... Not defined, masked out

0	0	0	1	0	0	0	0	x	x	x	x	x	x	x	x

15   8 7   0

Configuration Word 14 (write)

The module is configured using configuration word 14.

With the standard setting, the encoder module is operated as 32 Bit up/down counter with 4-fold evaluation.

⚠ **Please refer to path/status and timing diagrams in the section, "Incremental Encoder Operation".**

Bit	Description
15	0.... TPU operation switched off
	1.... TPU operation switched on
	To be able to use TPU operation, the module must be operated on the CP Interface.
	Channels A and B are switched through to the TPU (can be operated using LTX functions e.g. with LTXcab2()). Using the TPU-IN line, the signal source for the TPU-OUT line is selected. Both lines are operated using LTX functions (TPU-IN line e.g. with LTXdo0(), TPU-OUT line e.g. with LTXdi1()). A differentiation must be made as to if the comparator is switched on or off.
	a) Comparator is switched off (bit 3 in configuration word 8 = 0)
	TPU-IN = 0... TPU-OUT = reference pulse
	TPU-IN = 1 TPU-OUT = reference pulse & reference enable switch (binary AND logic)
	b) Comparator is switched on (bit 2 and bit 3 in configuration word 8 = 1, also compared to status of bit 2 in configuration word 8)
	TPU-IN = 0... TPU-OUT = inverted level of bit 0 in configuration word 8
	TPU-IN = 1... TPU-OUT = Level of bit 0 in configuration word 8, if threshold value 1 < counter ≤ threshold value 2
13 - 14	0
12	0.... Incremental encoder operation
	1.... Absolute encoder operation
6 - 11	0
5	0.... No effect on count direction
	1.... Reversed count direction
4	0.... Incremental encoder without comparator
	1.... Incremental encoder with comparator
3	0
2	0.... No effect on reference pulse
	1.... Reference pulse is inverted. This setting is used for encoders with a high pulse.
1	0.... Set counter immediately to 0. In data word 0 (module status), bit 7 is immediately set to 1.
	1.... Counter remains functioning. In data word 0 (module status), bit 7 is immediately set to 0 (conditional referencing).
	Depending on bit 0 in configuration word 14, bit 7 in data word 0 is set to 1 and cleared again with a positive edge on bit 4 in configuration word 8.
0	0.... Ignore reference enable switch (referencing with reference pulse) Refers to bit 4 in configuration word 8.
	1.... Activate reference enable switch (referencing with reference pulse and reference enable switch)

```
0 0 0 0 0 0 0 0 0    0
15              8 7          0
```

15.2.8 Variable Declaration for Absolute Encoder Operation

The variable declaration is valid for the following controllers:

- 2003 PCC CPU
- Remote I/O Bus Controller
- CAN Bus Controller

The variable declaration is made in PG2000. The variable declaration is described in Chapter 4, "Module Addressing".

Automation Studio™ Support:　　　　　See Automation Studio™ Help starting with V 1.40

Accessing screw-in modules is also explained in the sections "AF101" and "CPU".

Absolute encoder operation with PCC 2003 CPU and remote slaves

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for this module.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	WORD	Transp. In	0	●		Module status
Data word 1	INT32	Transp. In	2	●		Counter value
Configuration word 4	INT32	Transp. In	8	●		Counter value at positive edge of the reference enable switch
Configuration word 6	INT32	Transp. In	12	●		Counter value at negative edge of the reference enable switch
Configuration word 8	WORD	Transp. Out	16		●	Absolute encoder control
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

Absolute encoder operation with CAN slaves

Data access takes place using data and configuration words. The following table provides an overview of which data and configuration words are used for this module.

Data Access	VD Data Type	VD Module Type	VD Chan.	R	W	Description
Data word 0	INT32	Transp. In	0	●		Counter value
Data word 2	WORD	Transp. In	4	●		Module status
Configuration word 4	INT32	Transp. In	8	●		Counter value at positive edge of the reference enable switch
Configuration word 6	INT32	Transp. In	12	●		Counter value at negative edge of the reference enable switch
Configuration word 8	WORD	Transp. Out	16		●	Absolute encoder control
Configuration word 12	WORD	Transp. In	24	●		Module status
Configuration word 14	WORD	Transp. In	28	●		Module type
	WORD	Transp. Out	28		●	Module configuration

Access using CAN Identifiers

Access via CAN Identifiers is used if the slave is being controlled by a device from another manufacturer.
Access via CAN Identifiers is described in an example in Chapter 4, "Module Addressing".
The transfer modes are explained in Chapter 5, "CAN Bus Controller Functions".

Data cannot be packed on the NC161. Therefore one CAN object is transferred per screw-in module.
If an adapter module AF101 is equipped with a four NC161 modules, the CAN object has the following structure:

Slot	CAN ID [1]	Word 1		Word 2		Word 3		Word 4
1	542	Count. LL	Count. ML	Count. MH	Count. HH	Status L	Status H	Free
2	543	Count. LL	Count. ML	Count. MH	Count. HH	Status L	Status H	Free
3	544	Count. LL	Count. ML	Count. MH	Count. HH	Status L	Status H	Free
4	545	Count. LL	Count. ML	Count. MH	Count. HH	Status L	Status H	Free

[1] CAN ID = 542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)

nd ..... Node number of the CAN slave = 1

ma .... Module address of the AF101 = 1

sl .... Slot number of the screw-in module on the AF101 (1 - 4)

⚠ **B&R 2000 users have to exchange the data so that the high data is first (Motorola format)!**

For more information on ID allocation, see Chapter 5, "CAN Bus Controller Functions".

Description of Data and Configuration Words

Data Word 0 (read)

Data word 0 includes the module status time constant for the counter value.

	Bit	Description
	12 - 15	x .... Not defined, masked out
	11	0.... Counter value not taken 1.... Counter value is taken with the first positive edge on the reference enable switch
	10	0.... Counter value not taken 1.... Counter value is taken with the first negative edge on the reference enable switch
	9	0.... If bit 8 and bit 9 = 0, then the load is in the valid range or the supply is not switched on 1.... 5 V encoder supply overload or short-circuit
	8	0.... If bit 8 and bit 9 = 0, then the load is in the valid range or the supply is not switched on 1.... 5 V encoder supply is not loaded
	2 - 7	x .... Not defined, masked out
	1	Level of the reference enable switch
	0	x .... Not defined, masked out

```
x x x x         x x x x x x   x
15            8 7           0
```

Data Word 1 (read)

Counter Value MSW

Data Word 2 (read)

Counter Value LSW

Configuration Words 4+5 (read)

After setting bit 11 in configuration word 8, the configuration words receive the latched counter value with the first positive edge of the reference enable switch. The value is also valid if bit 11 is set in data word 0. Please refer to the timing diagram "Latching the Counter Value" in the section, "Incremental Encoder Operation".

Configuration Words 6+7 (read)

After setting bit 10 in configuration word 8, the configuration words receive the latched counter value with the first negative edge of the reference enable switch. The value is also valid if bit 10 is set in data word 0. Please refer to the timing diagram "Latching the Counter Value" in the section, "Incremental Encoder Operation".

Configuration Word 8 (write)

The absolute encoder is configured using configuration word 8.

	Bit	Description
	12 - 15	0
	11	0.... Counter value not taken 1.... Counter value is taken with the first positive edge on the reference enable switch (see configuration words 4 and 5) [1]
	10	0.... Counter value not taken 1.... Counter value is taken with the first negative edge on the reference enable switch (see configuration words 6 and 7) [1]
	8 - 9	0
	7	0.... 5 V encoder supply off (default) 1.... 5 V encoder supply on
	0 - 6	0

0	0	0	0			0	0		0	0	0	0	0	0	0	0

15   8 7   0

[1] The counter value is only taken once. Bit 10 and bit 11 must be reset for the value to be taken again. Bit 10 and bit 11 can be set again in configuration word 8 after the corresponding bit in the module status bit has gone to 0.

Configuration Word 12 (read)

Configuration word 12 contains the module status (current status unlatched). The module status is written to data word 0.

Configuration Word 14 (read)

The High Byte of configuration word 14 defines the module code.

	Bit	Description
	8 - 15	Module code = $10
	0 - 7	x.... Not defined, masked out

0	0	0	1	0	0	0	0	x	x	x	x	x	x	x	x

15   8 7   0

Configuration Word 14 (write)

The module is configured using configuration word 14.

	Bit	Description
	15	0.... TPU operation switched off
		1.... TPU operation switched on To be able to use TPU operation, the module must be operated on the CP Interface. The TPU-OUT line is used for status information. The line is handled using LTX functions (e.g. with LTXdi1()). TPU-OUT = 0 ... Position measurement active TPU-OUT = 1 ... New count available
	13 - 14	0
	12	0.... Incremental encoder operation 1.... Absolute encoder operation
	11	0.... Binary coded SSI encoder signal (SSI encoder) 1.... Gray coded SSI encoder signal
	10	0
	5 - 9	Number of preceding zeros before the MSB of the encoder value
	0 - 4	Number of valid bits in the encoder value As standard, bits 0 - 9 = 0. With this setting, the first 32 bits of the SSI encoder data flow are output.

```
 0  0  1    0
15          8  7          0
```

⚠ **Please refer to the example in section "Absolute Encoder Information".**

15.2.9 Incremental Encoder Operation

The following path/status and timing diagrams illustrate the functional behavior of the NC161 in incremental encoder operation with various configurations.

Comparator - Unconditional

The following settings are to be made:

Data / Configuration Word	Command to be Executed	Description
Configuration words 4+5	Threshold value 1	Define threshold value 1 for comparator, threshold value 1 ≤ threshold value 2
Configuration words 6+7	Threshold value 2	Define threshold value 2 for counter
Configuration word 8	Bit 0 = 0 or 1	Level of the comparator output
Configuration word 8	Bit 2 = 0	Unconditional comparator output
Configuration word 8	Bit 3 = 1	Switch comparator on
Configuration word 14	Bit 4 = 1	Incremental encoder operation with comparator
Configuration word 14	Bit 12 = 0	Incremental encoder operation

Path/Status Diagram
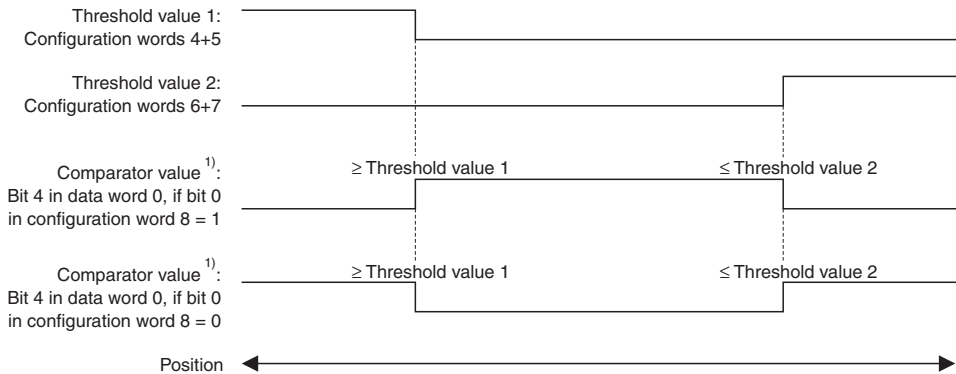
[1] The comparator shows jitter of 0 - ca. 1 ms

The clock output (pins 7 and 8) correspond to the comparator status of bit 4 in data word 0.

Comparator - Conditional

The following settings are to be made:

Data / Configuration Word	Command to be Executed	Description
Configuration words 4+5	Threshold value 1	Define threshold value 1 for comparator, threshold value 1 ≤ threshold value 2
Configuration words 6+7	Threshold value 2	Define threshold value 2 for counter
Configuration word 8	Bit 0 = 0 or 1	Level of the comparator output
Configuration word 8	Bit 2 = 1	Conditional comparator output
Configuration word 8	Bit 3 = 1	Switch comparator on
Configuration word 14	Bit 4 = 1	Incremental encoder operation with comparator
Configuration word 14	Bit 12 = 0	Incremental encoder operation

<u>Path/Status Diagram</u>
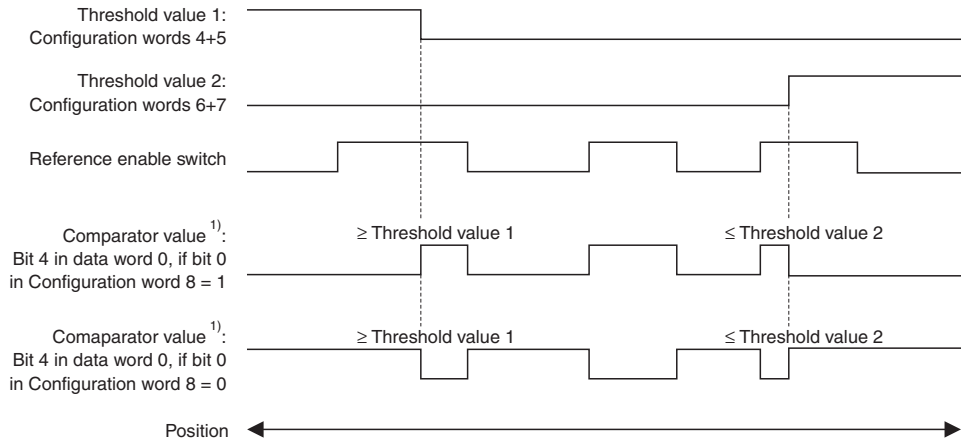
[1] The comparator shows jitter of 0 - ca. 1 ms

The clock output (pins 7 and 8) correspond to the comparator status of bit 4 in data word 0.

Latching the Counter Value

The following settings are to be made:

Data / Configuration Word	Command to be Executed	Description
Configuration word 8	Bit 10 = 0 or 1	Counter value is taken with the first negative edge on the reference enable switch
Configuration word 8	Bit 11 = 0 or 1	Counter value is taken with the first positive edge on the reference enable switch
Configuration word 14	Bit 12 = 0	Incremental encoder operation

Timing Diagram
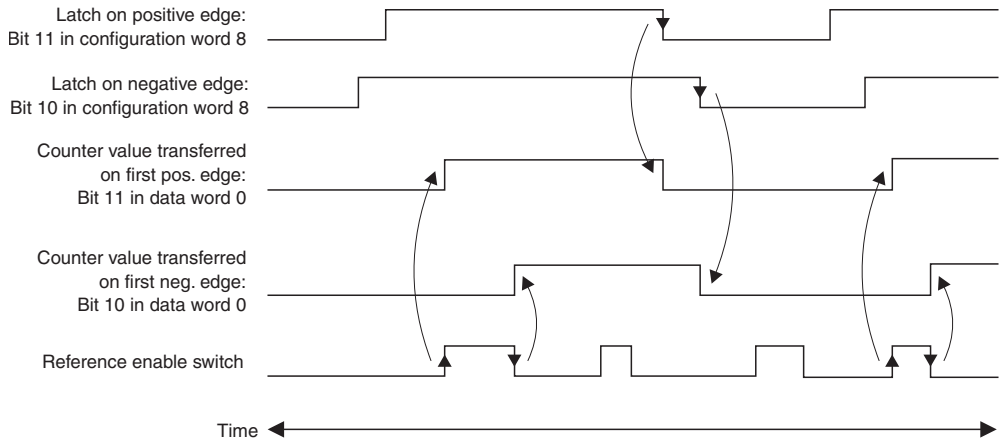
Referencing - Unconditional

The following settings are to be made:

Data / Configuration Word	Command to be Executed	Description
Configuration word 8	Bit 4 = 0 or 1	Clear counter (referencing), for control signal see configuration word 14
Configuration word 14	Bit 1 = 0	Clear counter immediately (unconditional referencing)
Configuration word 14	Bit 12 = 0	Incremental encoder operation

<u>Timing Diagram</u>

Clear counter:
Bit 4 in configuration word 8

Counter cleared:
Bit 7 in data word 0

State change after
each reference:
Bit 6 in data word 0

Time

Chapter 3
B&R 2003 Modules
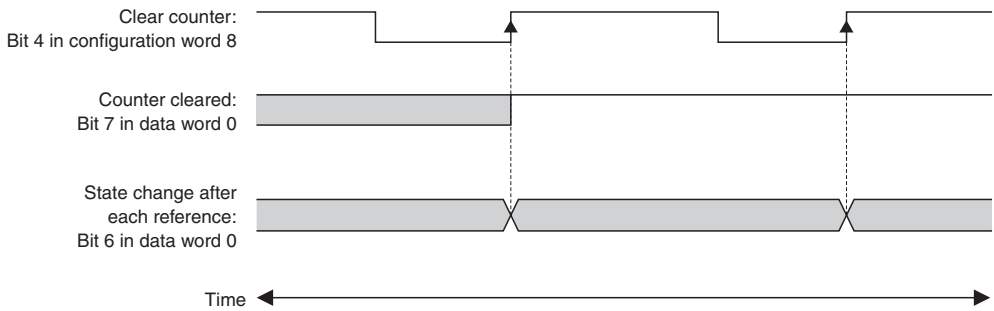
Referencing - Conditional with Reference Enable Switch

<u>Reference pulse - non-inverted</u>

Make the following settings for conditional referencing with reference enable switch and non-inverted reference pulse:

Data / Configuration Word	Command to be Executed	Description
Configuration word 8	Bit 4 = 0 or 1	Clear counter (referencing), for control signal see configuration word 14
Configuration word 14	Bit 0 = 1	Activate reference enable switch
Configuration word 14	Bit 1 = 1	Conditional referencing
Configuration word 14	Bit 2 = 0	Reference pulse non-inverted
Configuration word 14	Bit 12 = 0	Incremental encoder operation

<u>Timing Diagram</u>

Reference pulse - inverted

Make the following settings for conditional referencing with reference enable switch and inverted reference pulse:

Data / Configuration Word	Command to be Executed	Description
Configuration word 8	Bit 4 = 0 or 1	Clear counter (referencing), for control signal see configuration word 14
Configuration word 14	Bit 0 = 1	Activate reference enable switch
Configuration word 14	Bit 1 = 1	Conditional referencing
Configuration word 14	Bit 2 = 1	Reference pulse inverted
Configuration word 14	Bit 12 = 0	Incremental encoder operation

Timing Diagram

Referencing - Conditional without Reference Enable Switch

Reference pulse - non-inverted

Make the following settings for conditional referencing without reference enable switch and with non-inverted reference pulse:

Data / Configuration Word	Command to be Executed	Description
Configuration word 8	Bit 4 = 0 or 1	Clear counter (referencing), for control signal see configuration word 14
Configuration word 14	Bit 0 = 0	Ignore reference enable switch
Configuration word 14	Bit 1 = 1	Conditional referencing
Configuration word 14	Bit 2 = 0	Reference pulse non-inverted
Configuration word 14	Bit 12 = 0	Incremental encoder operation

Timing Diagram

Reference pulse - inverted

Make the following settings for conditional referencing without reference enable switch and with inverted reference pulse:

Data / Configuration Word	Command to be Executed	Description
Configuration word 8	Bit 4 = 0 or 1	Clear counter (referencing), for control signal see configuration word 14
Configuration word 14	Bit 0 = 0	Ignore reference enable switch
Configuration word 14	Bit 1 = 1	Conditional referencing
Configuration word 14	Bit 2 = 1	Reference pulse inverted
Configuration word 14	Bit 12 = 0	Incremental encoder operation

Timing Diagram

15.2.10 Absolute Encoder Operation

In the following examples, the absolute encoder will be read in once in formatted form and once as raw value. The following absolute encoder is used in both examples:

Encoder Type	Multiturn encoder
Encoder Resolution	256 x 4096 $\Rightarrow$ 20 Bits

Bits Transferred
    Data Bits
        Preceding Zeros
	24
	20
	4

Read in formatted data

If the protocol format delivered by the absolute encoder is known, the data is read in in formatted form. The following settings are to be made:

Data / Configuration Word	Command to be Executed	Description
Configuration word 14	Bit 0 - 4 = %10100	Number of valid bits in the encoder value: 20
Configuration word 14	Bit 5 - 9 = %00100	Number of preceding zeros: 4
Configuration word 14	Bit 11 = 0	Binary coded SSI encoder signal
Configuration word 14	Bit 12 = 1	Absolute encoder operation
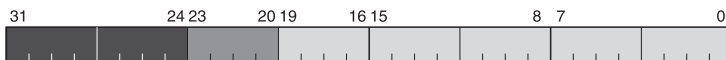
Protocol format

☐ 20 Data bits
▨ 4 Leading zeros
■ Invalid

Read in data as 32 bit raw value

The data is read in as 32 bit raw value if e.g. the protocol format delivered by the absolute encoder in not known. The following settings are to be made:

Data / Configuration Word	Command to be Executed	Description
Configuration word 14	Bit 0 - 4 = %00000	Number of valid bits in the encoder value: 0
Configuration word 14	Bit 5 - 9 = %00000	Number of preceding zeros: 0
Configuration word 14	Bit 11 = 0	Binary coded SSI encoder signal
Configuration word 14	Bit 12 = 1	Absolute encoder operation

Protocol format

☐ 20 Data bits
◼ 4 Leading zeros
◼ Invalid
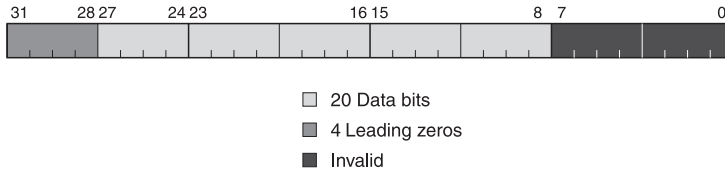
16 ACCESSORIES

16.1 OVERVIEW

Model Number	Description
7AC010.9	2003 bus cover, 5 pcs.
7AC011.9	2003 stress relief attachment, 5 pcs., incl. mounting material
7AC020.9	2003 bus cover, 1 pcs.
7TB710.9	2003 terminal block, 10 pin, screw clamps
7TB710.91	2003 terminal block, 10 pin, cage clamps
7TB710:90-01	2003 terminal block, 10 pin, 30 pcs., screw clamps
7TB710:91-01	2003 terminal block, 10 pin, 30 pcs., cage clamps
7TB712.9	2003 terminal block, 12 pin, screw clamps
7TB712.91	2003 terminal block, 12 pin, cage clamps
7TB712:90-02	2003 terminal block, 12 pin, 20 pcs., screw clamps
7TB712:91-02	2003 terminal block, 12 pin, 20 pcs., cage clamps
7TB718.9	2003 terminal block, 18 pin, screw clamps
7TB718.91	2003 terminal block, 18 pin, cage clamps
7TB718:90-02	2003 terminal block, 18 pin, 20 pcs., screw clamps
7TB718:91-02	2003 terminal block, 18 pin, 20 pcs., cage clamps
7TB722.9	2003 terminal block, 22 pin, screw clamps
7TB722.91	2003 terminal block, 22 pin, cage clamps
7TB733.9	2003 terminal block, 33 pin, screw clamps
7TB733.91	2003 terminal block, 33 pin, cage clamps
7TB736.9	2003 terminal block, 36 pin, screw clamps
7TB736.91	2003 terminal block, 36 pin, cage clamps
7TB754.9	2003 terminal block, 54 pin, screw clamps
7TB754.91	2003 terminal block, 54 pin, cage clamps
7TB772.91	2003 terminal block, 72 pin, cage clamps

16.2 AC010 / AC020

16.2.1 Technical Data
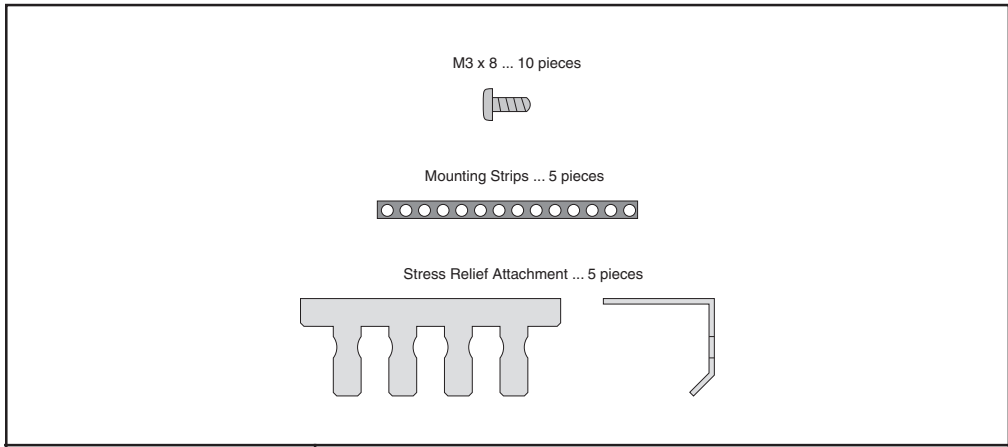
Module ID	AC010	AC020
Model Number	7AC010.9	7AC020.9
Short Description	2003 bus cover, 5 pcs.	2003 bus cover, 1 pcs.

16.2.2 General Information

If module slots remains free, fitting a bus cover on the first free slot is recommended. In this way, the last module can be protected from damage.

16.3 AC011

16.3.1 Technical Data

M3 x 8 ... 10 pieces

Mounting Strips ... 5 pieces

Stress Relief Attachment ... 5 pieces

Module ID	AC011
Model Number	7AC011.9
Short Description	2003 stress relief attachment, 5 pcs., incl. mounting material

16.3.2 General Information

The stress relief attachment is intended for use with the DO721 digital output modules. It can be installed on the module rack directly underneath the modules to prevent the cabling placing too much stress on the 4 pin connectors.

The stress relief attachment is mounted underneath the DO721. To do this, a threaded strip is inserted in the module rack and the stress relief attachment is fastened with two screws. The mounting material is contained in the delivery of the stress relief attachment.

The wires running into the module are then secured to the stress relief attachment using cable ties. In this way, the connectors are freed from stress, and the security of the electrical wiring can be guaranteed.

Stress Relief Attachment

Mounting Strip

Screws inserted in the upper hole

16.3.3 Connection Example

OUTPUT
RELAY
240 VAC
24 VDC
4 A / terminal

Fuse:
T 5A H 250V

N #

S #

1 2 3 4

DO721

Y1   Y2   Y3   Y4
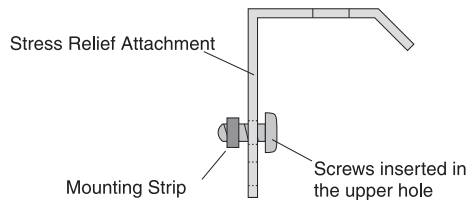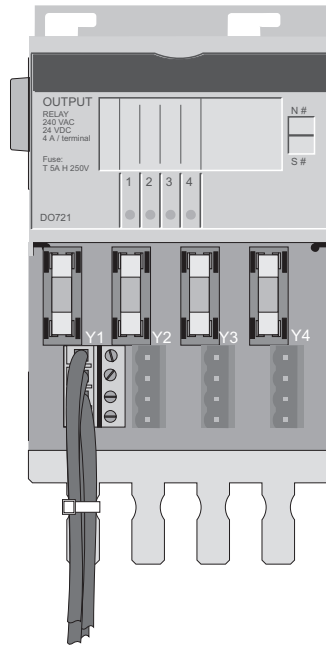
16.4 TB710

16.4.1 Order Data

Model Number	Short Description
7TB710.9	2003 terminal block, 10 pin, screw clamps
7TB710.91	2003 terminal block, 10 pin, cage clamps
7TB710:90-01	2003 terminal block, 10 pin, 30 pcs., screw clamps
7TB710:91-01	2003 terminal block, 10 pin, 30 pcs., cage clamps

Most I/O modules are wired using this 10 pin single row terminal block. This terminal block has a compact design and is easy remove (each module has two ejection levers).

16.4.2 Technical Data

TB710.9

TB710.91

Module ID	TB710
Number of Pins	10
Type of Terminal	Implemented as screw or cage clamps
Distance between Contacts	5.08 mm
Resistance between Contacts	6 mΩ
Nominal Voltage	250 V
Current Load [1]	Max. 12 A / contact
Wire Cross Section	0.14 mm² (AWG26) - 2.5 mm² (AWG12)
Cable Type	Only copper wires (no aluminum wires!)
Removal	Mechanical

[1] Take the respective limit data for the I/O modules into consideration!

16.5 TB712

16.5.1 Order Data

Model Number	Short Description
7TB712.9	2003 terminal block, 12 pin, screw clamps
7TB712.91	2003 terminal block, 12 pin, cage clamps
7TB712:90-02	2003 terminal block, 12 pin, 20 pcs., screw clamps
7TB712:91-02	2003 terminal block, 12 pin, 20 pcs., cage clamps

This single row 12 pin terminal block is used to connect screw-in modules. Removal is simplified by two ejection levers on the terminal block.

16.5.2 Technical Data

TB712.9

TB712.91
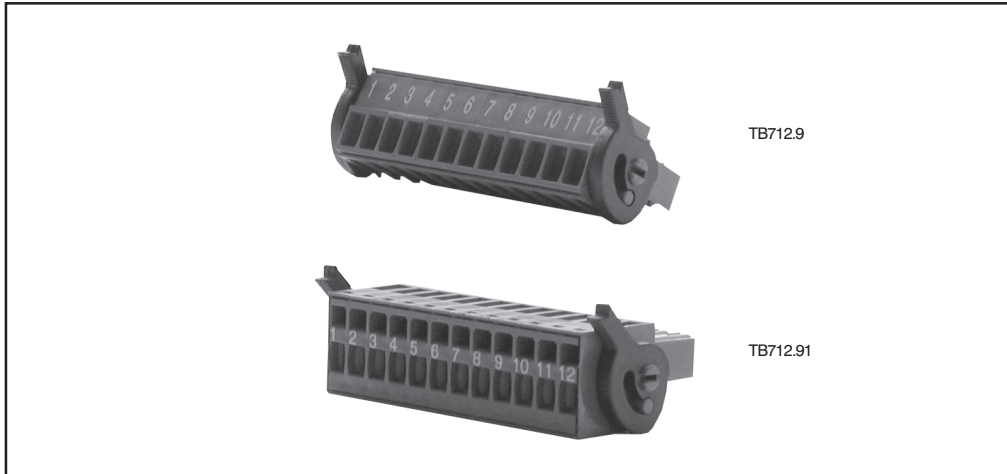
Module ID	TB712
Number of Pins	12
Type of Terminal	Implemented as screw or cage clamps
Distance between Contacts	3.5 mm
Nominal Voltage	125 V
Current Load [1]	Max. 12 A / contact
Wire Cross Section	0.08 mm² (AWG28) - 1.5 mm² (AWG16)
Cable Type	Only copper wires (no aluminum wires!)
Removal	Mechanical

[1] Take the respective limit data for the I/O modules into consideration!
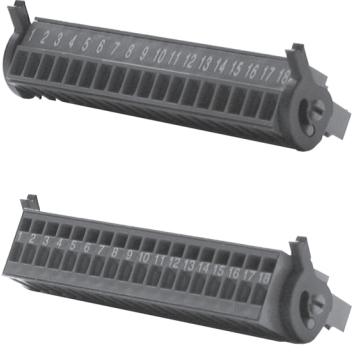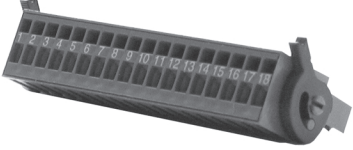
16.6 TB718

16.6.1 Order Data

Model Number	Short Description
7TB718.9	2003 terminal block, 18 pin, screw clamps
7TB718.91	2003 terminal block, 18 pin, cage clamps
7TB718:90-02	2003 terminal block, 18 pin, 20 pcs., screw clamps
7TB718:91-02	2003 terminal block, 18 pin, 20 pcs., cage clamps

This single row 18 pin terminal block is used to connect B&R 2003 I/O modules. Removal is simplified by two ejection levers on the terminal block.

16.6.2 Technical Data

TB718.9

TB718.91

Module ID	TB718
Number of Pins	18
Type of Terminal	Implemented as screw or cage clamps
Distance between Contacts	3.5 mm
Nominal Voltage	125 V
Current Load [1]	Max. 12 A / contact
Wire Cross Section	0.08 mm² (AWG28) - 1.5 mm² (AWG16)
Cable Type	Only copper wires (no aluminum wires!)
Removal	Mechanical

[1] Take the respective limit data for the I/O modules into consideration!

16.7 TB722

16.7.1 Order Data

Model Number	Short Description
7TB722.9	2003 terminal block, 22 pin, screw clamps
7TB722.91	2003 terminal block, 22 pin, cage clamps

The terminal block TB722 is used to supply the digital inputs on the DM435 module. It is a 22 pin dual level terminal block with either screw clamps or cage clamps. The contacts on the front are connected together and the ones on the back are also connected together which creates a positive and a GND strip.

16.7.2 Technical Data

TB722.9

TB722.91

Module ID	TB722	
Number of Pins	22	
Type of Terminal	Implemented as screw or cage clamps	
Distance between Contacts	5.08 mm	
Nominal Voltage	250 V	
Current Load [1]	Max. 12 A / contact	
Wire Cross Section	0.14 mm² (AWG26) - 2.5 mm² (AWG12)	
Cable Type	Only copper wires (no aluminum wires!)	
Dimensions 　Height 　Width 　Depth	TB722.9 33 mm 76 mm 52 mm	TB722.91 33 mm 76 mm 33 mm

[1] Take the respective limit data for the I/O modules into consideration!

16.8 TB733

16.8.1 Order Data

Model Number	Short Description
7TB733.9	2003 terminal block, 33 pin, screw clamps
7TB733.91	2003 terminal block, 33 pin, cage clamps

The terminal block TB733 is used as an additional jumper terminal if the digital mixed module DM435 is operated using a 3-line connection. The TB733 is a 33 pin 3-level terminal block with either screw clamps or cage clamps. All terminals in a terminal row are electrically connected

16.8.2 Technical Data

TB733.9

TB733.91
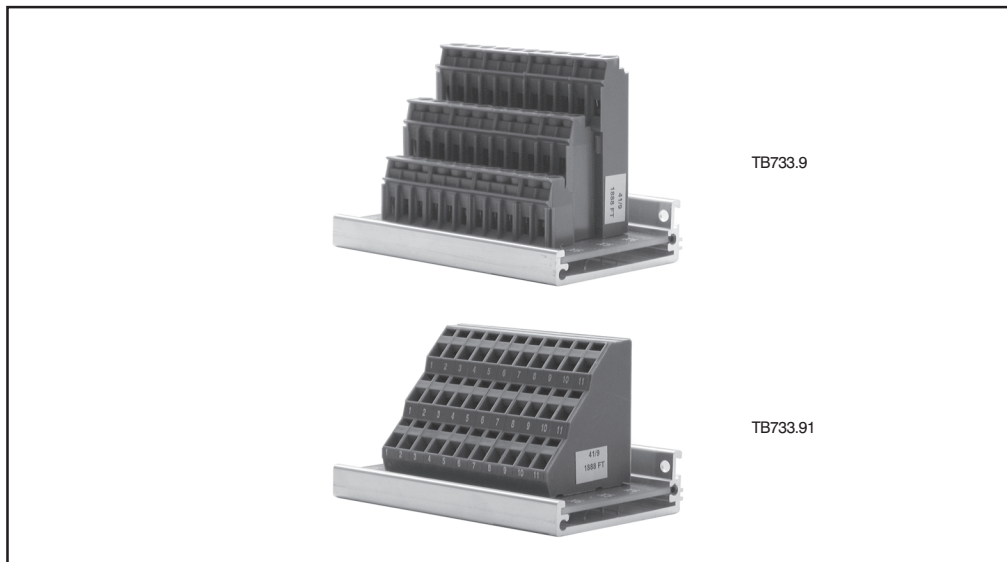
Module ID	TB733	
Number of Pins	33 All 11 terminals in a terminal row are electrically connected	
Type of Terminal	Implemented as screw or cage clamps	
Distance between Contacts	5.08 mm	
Nominal Voltage	250 V	
Current Load [1]	Max. 10 A / contact	
Fuse	External fuse is required	
Wire Cross Section	0.14 mm² (AWG26) - 2.5 mm² (AWG12)	
Cable Type	Only copper wires (no aluminum wires!)	

Dimensions
    Height
    Width
    Depth
	TB733.9
	48 mm
	76 mm
	52 mm
		TB733.91
		48 mm
		76 mm
		44 mm

[1] Take the respective limit data for the I/O modules into consideration!
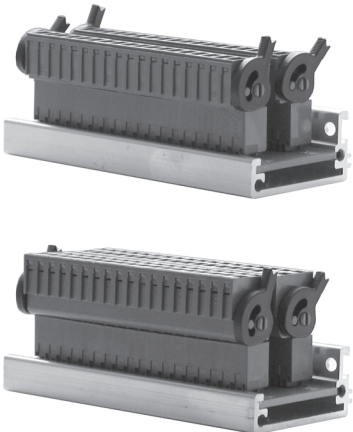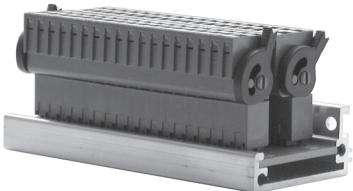
16.9 TB736

16.9.1 Order Data

Model Number	Short Description
7TB736.9	2003 terminal block, 36 pin, screw clamps
7TB736.91	2003 terminal block, 36 pin, cage clamps

The terminal block TB736 is used to supply the digital inputs on the DM465 module. The TB736 consists of two 18 pin pin-connectors and corresponding socket connectors as screw or cage clamp terminal blocks with ejection levers. All terminals in a terminal row are electrically connected

16.9.2 Technical Data

TB736.9

TB736.91

Module ID	TB736	
Number of Pins	36 All 18 terminals in a terminal row are electrically connected	
Design	Two 18 pin pin-connectors and corresponding socket connectors as screw or cage clamp terminal blocks with ejection levers	
Distance between Contacts	3.5 mm	
Nominal Voltage	125 V	
Current Load [1]	Max. 12 A / contact	
Wire Cross Section	0.08 mm² (AWG28) - 1.5 mm² (AWG16)	
Cable Type	Only copper wires (no aluminum wires!)	
Dimensions Height Width Depth	TB736.9 33 mm 76 mm 32 mm	TB736.91 33 mm 76 mm 32 mm

[1] Take the respective limit data for the I/O modules into consideration!
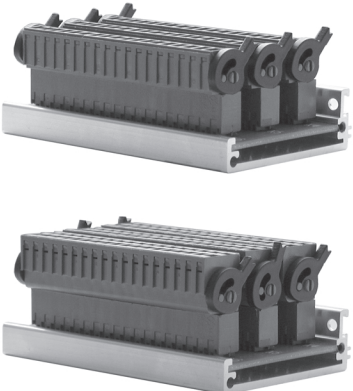
16.10 TB754

16.10.1 Order Data

Model Number	Short Description
7TB754.9	2003 terminal block, 54 pin, screw clamps
7TB754.91	2003 terminal block, 54 pin, cage clamps

The terminal block is used as an additional jumper terminal and is specially designed for 3-line connections, e.g. for the DM465 (3-line sensors). The TB754 consists of three18 pin pin-connectors and corresponding socket connectors as screw or cage clamp terminal blocks with ejection levers. All connections in a terminal row are electrically connected.

16.10.2 Technical Data

TB754.9

TB754.91

Module ID	TB754
Number of Pins	54 All 18 terminals in a terminal row are electrically connected
Design	Three 18 pin pin-connectors and corresponding socket connectors as screw or cage clamp terminal blocks with ejection levers
Distance between Contacts	3.5 mm
Nominal Voltage	125 V
Current Load [1]	Max. 12 A / contact
Fuse	External fuse is required
Wire Cross Section	0.08 mm² (AWG28) - 1.5 mm² (AWG16)
Cable Type	Only copper wires (no aluminum wires!)
Dimensions Height Width Depth	 50 mm 76 mm 32 mm

[1] Take the respective limit data for the I/O modules into consideration!

16.11 TB772

16.11.1 Order Data

Model Number	Short Description
7TB772.91	2003 terminal block, 72 pin, cage clamps

The terminal block TB772 is used as an additional jumper terminal if the digital mixed module DM465 is operated using a 3-line connection. The TB772 consists of two 36 pin pin-connectors and corresponding socket connectors as cage clamp terminal blocks with ejection levers. All terminals in a terminal row are electrically connected

16.11.2 Technical Data
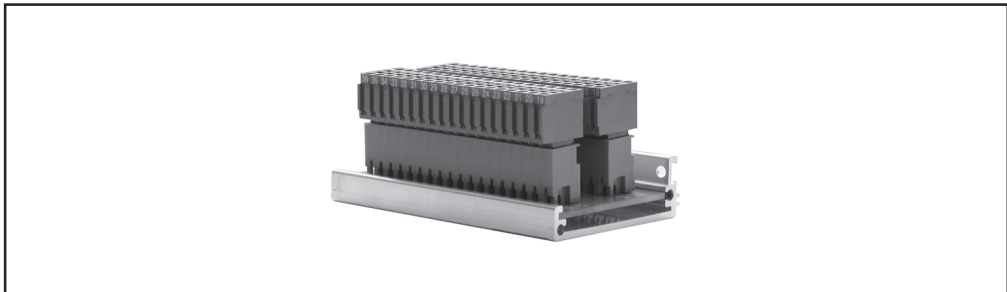
Module ID	TB772
Number of Pins	72 All 18 terminals in a terminal row are electrically connected
Design	Two 36 pin pin-connectors and corresponding socket connectors as cage clamp terminal blocks with ejection levers
Distance between Contacts	3.5 mm
Nominal Voltage	50 V
Current Load [1]	Max. 5 A / contact
Fuse	External fuse is required
Wire Cross Section	0.5 mm² - 1 mm²
Cable Type	Only copper wires (no aluminum wires!)
Dimensions Height Width Depth	 48 mm 76 mm 34 mm

[1] Take the respective limit data for the I/O modules into consideration!

17 MANUALS

17.1 OVERVIEW

Model Number	Description
MASYS22003-0	B&R 2003 User's Manual, German
MASYS22003-E	B&R 2003 User's Manual, English

CHAPTER 4
MODULE ADDRESSING

1 MEMORY AREA FOR A SCREW-IN MODULE

1.1 STRUCTURE

Data exchange with a screw-in module takes place using a 32 word long memory area. Depending on the screw-in module, up to 16 words can be accessed (read or write).
The memory area has the same structure for all screw-in modules:

Read (R)	Write (W)	Processing
Data word 0	Data word 0	Processing takes place
Data word 1	Data word 1	cyclically and automatically by
Data word 2	Data word 2	the operating system
Data word 3	Data word 3	
Configuration word 4	Configuration word 4	Processing takes place
Configuration word 5	Configuration word 5	when requested by the user
Configuration word 6	Configuration word 6	
Configuration word 7	Configuration word 7	
Configuration word 8	Configuration word 8	
Configuration word 9	Configuration word 9	
Configuration word 10	Configuration word 10	
Configuration word 11	Configuration word 11	
Configuration word 12	Configuration word 12	
Configuration word 13	Configuration word 13	
Configuration word 14	Configuration word 14	
Configuration word 15	Configuration word 15	

1.2 DATA WORDS

The data words are processed by the operating system cyclically and automatically. They are used for the following tasks:

- Input data is read by the input modules.
- Output data is written to the output modules.

1.3 CONFIGURATION WORDS

The configuration words are processed when requested by the user. Configuration words are also called CFG words.
They are used for the following tasks:

- Configuring a screw-in module
- Writing output data
- Reading input data
- Reading the module status
- Reading the module type

2 VARIABLE DECLARATION IN PG2000

The variable declaration in the PG2000 programming system depends on the controller:

CPU - PCC 2003	
Data words	Variable declaration
Configuration words	I/O FBKs

Remote I/O Slave	
Data words	Variable declaration
Configuration words	Variable declaration

CAN Slave	
Data words	Variable declaration
Configuration words	CAN command FBK CANIOcmd()

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

3 VARIABLE DECLARATION WITH 2003 PCC CPU

3.1 GENERAL INFORMATION

Entering the variable declaration is described in the "PG2000 Software User's Manual".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

The example described here are especially suited to the specific characteristics of a B&R 2003 CPU.

Take note of the following points:

- Screw-in modules inserted on the CP interface are assigned module address 0.
- Accessing configuration words takes place using I/O function blocks.
- Eight configuration words are processed in a program cycle per analog interface module! If more configuration words are required, some of the screw-in modules must be inserted in another AF101 module or access of the configuration words must be staggered.

The variable declaration is described using the hardware configuration shown below:
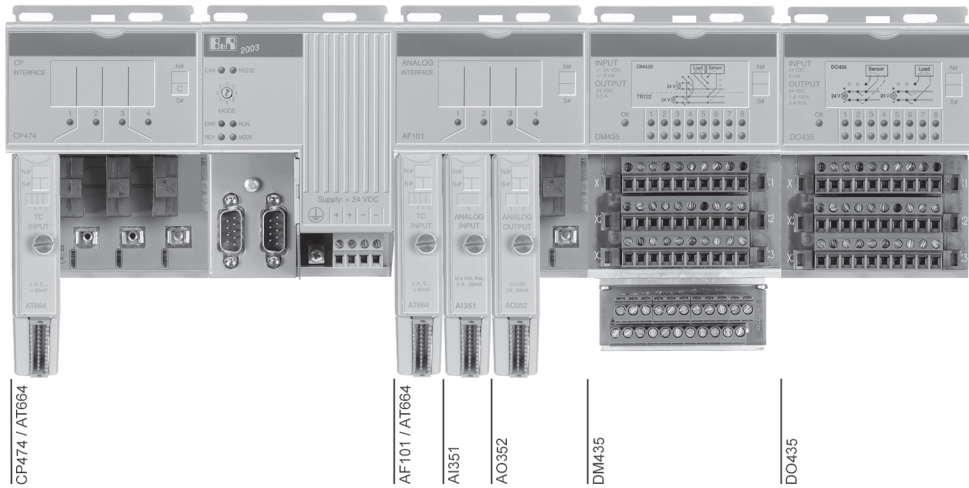
3.2 MODULE ASSIGNMENTS

AT664 on CP Interface and AT664 on AF101	
Channels 1 - 4	All channels are assigned

AI351	
Channel 1	Voltage measurement without threshold switch

AO352	
Channels 1 -2	All channels are assigned

DM435	
Digital Outputs	
Channels 1 -2	Assigned
Channels 3 -8	Free
Digital Inputs	
Channels 1 -2	Assigned
Channels 3 -8	Free

DO435	
Channels 1 -2	Digital inputs
Channels 3 - 4	Digital outputs
Channels 5 -8	Free

3.3 TASK OVERVIEW

The application program is divided into the following tasks:

3.4 I/O LIBRARY

Before you can use the function blocks to handle the configuration words, you have to import the I/O library into the project database. The function, *Import Library* in the *File* menu is provided for this purpose.
The I/O library must be entered in the GDM as a system module.

3.5 AF101

This task processes the screw-in modules on the CP interface and on the analog interface module AF101.
The shovel entries are defined in the INIT-SP. The shovel entries for the configuration are carried out in the INIT-SP, the ones for reading the module status are carried out in the cyclic section of the program.

3.5.1 INIT-SP

```
Project: cpu                                      File: InitSP : af101

;*******************************************************************************
;**  Definition of the individual shovel instructions for the main rack    **
;*******************************************************************************

Err_Ptest = 0
Err_IO_data = 0

AF101_slot = 1    ;Module address on rack
AT664_slot = 1    ;Slot address on AF101
AI351_slot = 2    ;Slot address on AF101

;#############################################################################
;## Multipler physical shovel instruction used to write                     ##
;## parameter word 14 from the AT664 to the AF101, Slot 1                    ##
;#############################################################################
IO_struct.io_type=5                  ;2003 IO
IO_struct.master_no=1                ;Master always 1
IO_struct.slave_no=0                 ;Slave number always 0
IO_struct.module_adr=AF101_slot      ;Module slot
IO_struct.intern_off= 32*(AT664_slot -1)+ 14 *2 ;Reg. no. 14 on ScrM = AT664_slot
IO_struct.mode=%00000000             ;Write, Normal
IO_struct.data_len=2                 ;2*Byte
IO_struct.reserve=0                  ;Not used
IO_struct.data_adr=adr(AT_r_reg14)   ;Address of the data area with image of
                                     ;parameter word 14
AT_r_reg14 = $0000                   ;No conversion
;AT_r_reg14 = $1001                  ;J type sensor \
;AT_r_reg14 = $1002                  ;K type sensor  > filter time 60 Hz
;AT_r_reg14 = $1003                  ;S type sensor /
```

```
        IO_ptest(1,adr(IO_struct),status_pt1,ptest1_ok);Test if shovel instruction
                                        ;can be carried out

        if (ptest1_ok = 1) then
          ;Declare shovel instruction #0
          IO_mphydef(1,adr(IO_struct),adr(af_sc_buf),0,status_mp1,af_ident)
        else
          Err_Ptest = BIT_SET(Err_Ptest,0)
        endif


        ;############################################################################
        ;## Multipler physical shovel instruction to write parameter word        ##
        ;## 14 from the AT664 to the left side of the CP474 (CP interface)        ##
        ;############################################################################
        IO_struct.io_type=5                     ;2003 IO
        IO_struct.master_no=1                   ;Master always 1
        IO_struct.slave_no=0                    ;Slave number always 0
        IO_struct.module_adr=0                  ;Module slot 0 -> left side
        IO_struct.intern_off= 32*(AT664_slot -1)+ 14 *2 ;Reg. no. 14 on ScrM = AT664_slot
        IO_struct.mode=%00000000                ;Write, Normal
        IO_struct.data_len=2                    ;2*Byte
        IO_struct.reserve=0                     ;Not used
        IO_struct.data_adr=adr(AT_l_reg14)      ;Address of the data area with image of
                                                ;parameter word 14
        AT_l_reg14 = $0000                      ;No conversion
        ;AT_l_reg14 = $1001                     ;J type sensor \
        ;AT_l_reg14 = $1002                     ;K type sensor  > filter time 60 Hz
        ;AT_l_reg14 = $1003                     ;S type sensor /

        IO_ptest(1,adr(IO_struct),status_pt2,ptest2_ok);Test id shovel instruction
                                        ;can be carried out

        if (ptest2_ok = 1) then
          ;Declare shovel instruction #1
          IO_mphydef(1,adr(IO_struct),adr(af_sc_buf),1,status_mp2,af_ident)
        else
          Err_Ptest = BIT_SET(Err_Ptest,1)
        endif

        if Err_Ptest=0 then
          loop            ;Shovel instruction carried out cyclically ...
            IO_data(1,af_ident,status_dat)
            exitif status_dat <> 5559    ;...until shovelling is no longer active
          endloop
        endif


        ;############################################################################
        ;## Multipler physical shovel instruction to read the module status      ##
        ;## (parameter word 12) for AI351                                        ##
        ;############################################################################
        IO_struct.io_type=5                     ;2003 IO
        IO_struct.master_no=1                   ;Master always 1
        IO_struct.slave_no=0                    ;Slave number always 0 for 2003
        IO_struct.module_adr=AF101_slot         ;Module slot
        IO_struct.intern_off=32*(AI351_slot -1)+ 12 *2;Reg. no. 12 on ScrM = AI351_slot
        IO_struct.mode=%00100000                ;Read, Normal
        IO_struct.data_len=2                    ;Word
        IO_struct.reserve=0                     ;Not used
        IO_struct.data_adr=adr(AI_statin)       ;Address of the data area where
                                                ;status word is stored.
```

```
    IO_ptest(1,adr(IO_struct),status_pt2,ptest2_ok);Test if shovel instruction
                                          ;can be carried out

    if (ptest2_ok = 1) then
        ;Change parameters for shovel instruction #0
        IO_mphydef(1,adr(IO_struct),adr(af_sc_buf),0,status_mp2,af_ident)
        ;Delete shovel instruction #1
        IO_mphydef(1,0,adr(af_sc_buf),1,status_mp3,af_ident)
    else
       Err_Ptest = BIT_SET(Err_Ptest,2)
    endif


    AF101_data = $FF
    if (status_mp2=0) and (status_mp3=0) and (Err_Ptest=0) and (status_dat=0) then
        AF101_data = 0
    endif


    ;----------------------------------------------------------------------------
```

3.5.2 Cyclic Program

```
    Project: cpu                                        File: af101.src

    ;*****************************************************************************
    ;*** Required for the definition of the variables (I/O variable would be   ***
    ;*** linked in a more logical manner in real applications !)               ***
    ;*****************************************************************************

    AT664r_in.chan1=AT664r_in.chan1
    AT664r_in.chan2=AT664r_in.chan2
    AT664r_in.chan3=AT664r_in.chan3
    AT664r_in.chan4=AT664r_in.chan4
    AI351_in=AI351_in
    AO352_out.chan1=AO352_out.chan1
    AO352_out.chan2=AO352_out.chan2
    AT664l_in1=AT664l_in1
    AT664l_in2=AT664l_in2
    AT664l_in3=AT664l_in3
    AT664l_in4=AT664l_in4


    ;*****************************************************************************
    ;*** Carry out the defined multiple shovel instructions using IO_data      ***
    ;*****************************************************************************


    ;*** Read parameter word 12 on AI351 cyclically                        ***

    if ( AF101_data = 0  ) then                    ;Only if INIT-SP successful
       IO_data(1,af_ident,status)                  ;Carry out shovel instruction
       ;If shovel instruction is not complete(0) and not active(5559)
       if (status<>0) and (status<>5559) then
          ;Error evaluation here
       endif
    endif
```

3.5.3 Variable Declaration

```
┌o───────────────────────── af101.vd ──────────────────────[↓][↕]┐
│ ┌──────────┬──────────┬───────────┬────────┬─────────────┬──────────────┬─────────┐
│ │ Name     │ Scope    │ Data Type │ Length │ I/O-Type    │ Init Value   │ Long Nam│
│ ├──────────┼──────────┼───────────┼────────┼─────────────┼──────────────┼─────────┤
│ │ AF101_data│ local   │ BYTE      │ 1      │ INTERNAL    │ * remanent   │         │
│ │ AF101_slot│ local   │ BYTE      │ 1      │ INTERNAL    │ * remanent   │         │
│ │ AI351_in │ tc_global│ INT16     │ 1      │ IP3.1.2.1   │ ----------   │         │
│ │ AI351_slot│ local   │ BYTE      │ 1      │ INTERNAL    │ * remanent   │         │
│ │ AI_statin│ local    │ WORD      │ 1      │ INTERNAL    │ * remanent   │         │
│ │ AO352_out│ tc_global│ ao352     │ 1      │ QTP3.1.3.0  │ ----------   │         │
│ │ AT664_slot│ local   │ BYTE      │ 1      │ INTERNAL    │ * remanent   │         │
│ │ AT664l_in1│ tc_global│ INT16    │ 1      │ IP3.0.1.1   │ ----------   │         │
│ │ AT664l_in2│ tc_global│ INT16    │ 1      │ IP3.0.1.2   │ ----------   │         │
│ │ AT664l_in3│ tc_global│ INT16    │ 1      │ IP3.0.1.3   │ ----------   │         │
│ │ AT664l_in4│ tc_global│ INT16    │ 1      │ IP3.0.1.4   │ ----------   │         │
│ │ AT664r_in│ tc_global│ at664     │ 1      │ ITP3.1.1.0  │ ----------   │         │
│ │ AT_l_reg14│ local   │ WORD      │ 1      │ INTERNAL    │ * remanent   │         │
│ │ AT_r_reg14│ local   │ WORD      │ 1      │ INTERNAL    │ * remanent   │         │
│ │ Err_IO_dat│ local   │ BYTE      │ 1      │ INTERNAL    │ * remanent   │         │
│ └──────────┴──────────┴───────────┴────────┴─────────────┴──────────────┴─────────┘
│ 1/28                                                                        5
└───────────────────────────────────────────────────────────────┘

┌o───────────────────────── af101.vd ──────────────────────[↓][↕]┐
│ ┌──────────┬──────────┬───────────┬────────┬─────────────┬──────────────┬─────────┐
│ │ Name     │ Scope    │ Data Type │ Length │ I/O-Type    │ Init Value   │ Long Nam│
│ ├──────────┼──────────┼───────────┼────────┼─────────────┼──────────────┼─────────┤
│ │ Err_Ptest│ local    │ BYTE      │ 1      │ INTERNAL    │ * remanent   │         │
│ │ IO_struct│ local    │ IO_struct │ 1      │ INTERNAL    │ ----------   │         │
│ │ af_ident │ local    │ LONG      │ 1      │ INTERNAL    │ * remanent   │         │
│ │ af_sc_buf│ local    │ BYTE      │ 78     │ INTERNAL    │ * remanent   │         │
│ │ ptest1_ok│ local    │ BIT       │ 1      │ INTERNAL    │ * remanent   │         │
│ │ ptest2_ok│ local    │ BIT       │ 1      │ INTERNAL    │ * remanent   │         │
│ │ status   │ local    │ WORD      │ 1      │ INTERNAL    │ * remanent   │         │
│ │ status_dat│ local   │ WORD      │ 1      │ INTERNAL    │ * remanent   │         │
│ │ status_mp1│ local   │ WORD      │ 1      │ INTERNAL    │ * remanent   │         │
│ │ status_mp2│ local   │ WORD      │ 1      │ INTERNAL    │ * remanent   │         │
│ │ status_mp3│ local   │ WORD      │ 1      │ INTERNAL    │ * remanent   │         │
│ │ status_pt1│ local   │ WORD      │ 1      │ INTERNAL    │ * remanent   │         │
│ │ status_pt2│ local   │ WORD      │ 1      │ INTERNAL    │ * remanent   │         │
│ └──────────┴──────────┴───────────┴────────┴─────────────┴──────────────┴─────────┘
│ 16/28                                                                       5
└───────────────────────────────────────────────────────────────┘
```

3.5.4 AT664 on CP Interface

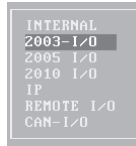Four channels are used on the AT664.

Data type

The definition can be made in two ways:

- Individually for each channel
- Using a structure variable (see section "AT664 on AF101")

On this AT664, the data type is defined individually for each channel.

I/O Type

Select I/O type "2003 I/O" from the pop-up window. Then a dialog box will be opened.

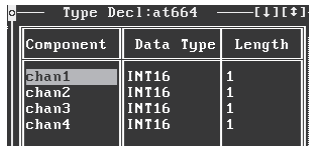The following settings are to be made in the dialog box:

3.5.5 AT664 on AF101

Four channels are used on the AT664.
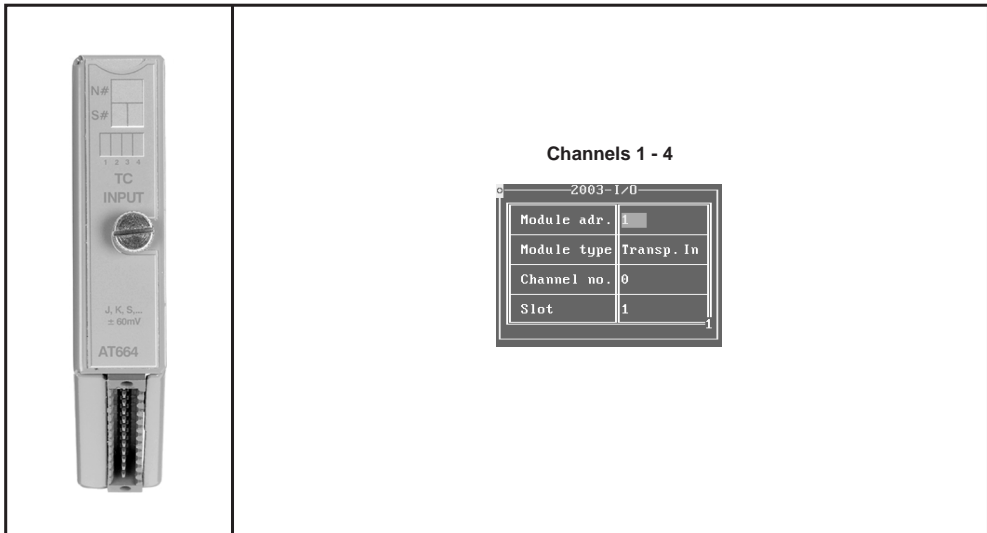
Data type

A structure will be defined for this AT664 using the data type *typedef*. After entering the structure name *at664*, the elements of the structure variables can be entered in a dialog box:

```
○──── Type Decl:at664 ────[↓][↕]──
┌──────────┬───────────┬─────────┐
│Component │ Data Type │ Length  │
├──────────┼───────────┼─────────┤
│chan1     │INT16      │1        │
│chan2     │INT16      │1        │
│chan3     │INT16      │1        │
│chan4     │INT16      │1        │
└──────────┴───────────┴─────────┘
```

I/O Type

The following settings are to be made in the dialog box *I/O Type*:

Channels 1 - 4

```
○────2003-I/O────
┌──────────┬───────────┐
│Module adr.│1         │
│Module type│Transp. In│
│Channel no.│0         │
│Slot       │1         │
└──────────┴──────────┘1
```

3.5.6 AI351

For the AI351, the following settings are to be made in the dialog box *I/O Type*:

Channel 1

```
c─────2003-I/O─────
│Module adr. │1       │
│Module type │Analog In│
│Channel no. │1       │
│Slot        │2       │
└────────────────────1
```
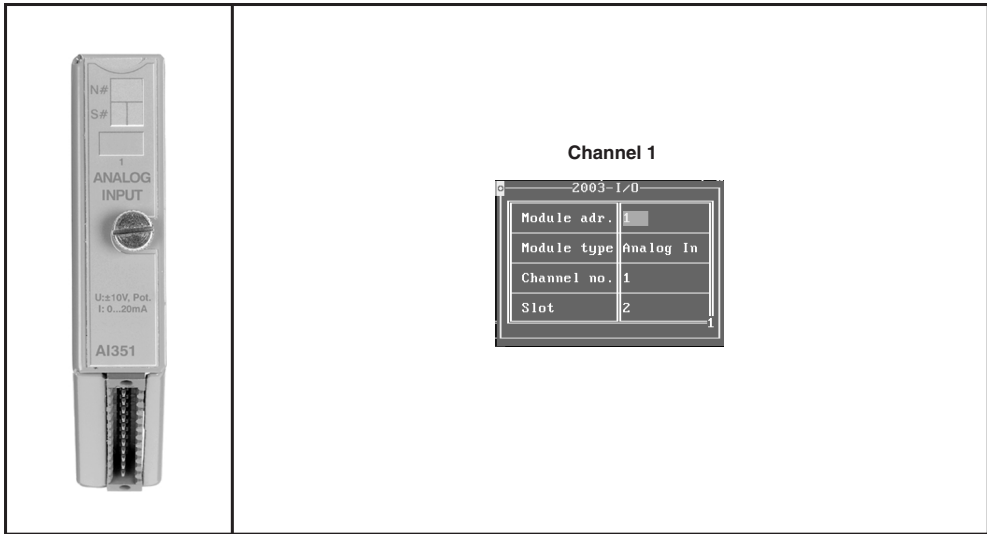
3.5.7 AO352

Two channels are used on the AO352.

Data type

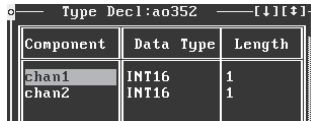The channels can also be accessed individually instead of using transparent access (as in this example). A structure is defined using the *typedef* data type. After entering the structure name *ao352*, the elements of the structure variables can be entered in a dialog box:

```
o─── Type Decl:ao352 ───[↓][↕]
┌───────────┬───────────┬────────┐
│ Component │ Data Type │ Length │
├───────────┼───────────┼────────┤
│ chan1     │ INT16     │ 1      │
│ chan2     │ INT16     │ 1      │
└───────────┴───────────┴────────┘
```

I/O Type

The following settings are to be made in the dialog box *I/O Type*:

Channels 1 and 2

```
o──── 2003-I/O ────
┌────────────┬────────────┐
│ Module adr.│ 1          │
│ Module type│ Transp.Out │
│ Channel no.│ 0          │
│ Slot       │ 3          │
└────────────┴──────────1─┘
```
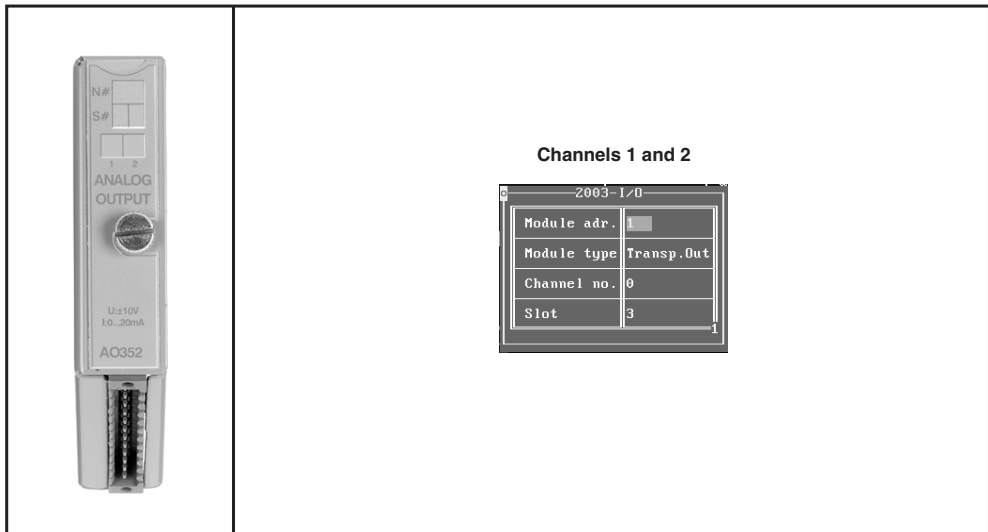
3.6 DM435

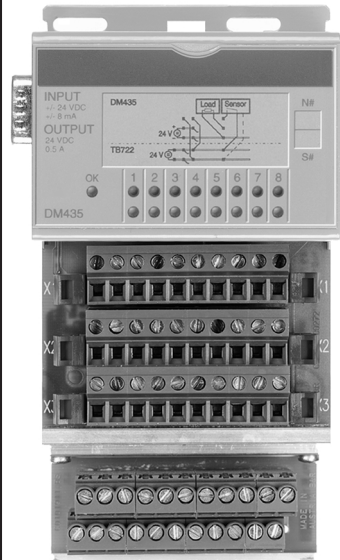On the DM435, two channels are used as digital inputs and two channels are used as digital outputs.

3.6.1 Variable Declaration

```
o─────────────────── dm435.vd ───────────────────[↓][↕]─
 ┌─────────┬─────────┬──────────┬──────┬─────────┬──────────┬─────────┐
 │ Name    │ Scope   │ Data Type│Length│ I/O-Type│ Init Value│ Long Nam│
 ├─────────┼─────────┼──────────┼──────┼─────────┼──────────┼─────────┤
 │digin_1  │tc_global│BIT       │1     │IP3.2.1  │──────────│         │
 │digin_2  │tc_global│BIT       │1     │IP3.2.2  │──────────│         │
 │digout_1 │tc_global│BIT       │1     │QP3.2.1  │* remanent│         │
 │digout_2 │tc_global│BIT       │1     │QP3.2.2  │* remanent│         │
 │dm435stat│tc_global│BYTE      │1     │ISP3.2.0 │──────────│         │
 └─────────┴─────────┴──────────┴──────┴─────────┴──────────┴─────────┘
─1/5─────────────────────────────────────────────────────────5─
```

3.6.2 I/O Type

The settings shown below are to be made in the dialog box *I/O Type*.
The parameter **Slot** is set to 1 by the PG2000 programming system, and therefore does not have to be set by the user.

Channel 1 - Digital Input

```
o────── 2003-I/O──────
 Module adr.   2
 Module type  Digit. In
 Channel no.  1
 Slot         1
                      1
```

Channel 2 - Digital Input

```
o────── 2003-I/O──────
 Module adr.   2
 Module type  Digit. In
 Channel no.  2
 Slot         1
                      1
```

Channel 1 - Digital Output

```
o────── 2003-I/O──────
 Module adr.   2
 Module type  Digit. Out
 Channel no.  1
 Slot         1
                      1
```

Channel 2 - Digital Output

```
o────── 2003-I/O──────
 Module adr.   2
 Module type  Digit. Out
 Channel no.  2
 Slot         1
                      1
```

Module Status

```
o────── 2003-I/O──────
 Module adr.   2
 Module type  Status In
 Channel no.  0
 Slot         1
                      1
```

3.7 DO435

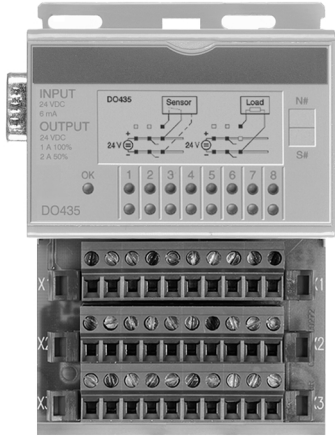On the DO435, two channels are used as digital inputs and two channels are used as digital outputs.

3.7.1 Variable Declaration

Name	Scope	Data Type	Length	I/O-Type	Init Value	Long Nam
dig_in1	tc_global	BIT	1	IP3.3.1	----------	
dig_in2	tc_global	BIT	1	IP3.3.2	----------	
dig_out3	tc_global	BIT	1	QP3.3.3	* remanent	
dig_out4	tc_global	BIT	1	QP3.3.4	* remanent	
do435_stat	tc_global	BYTE	1	ISP3.3.0	----------	

3.7.2 I/O Type

The settings shown below are to be made in the dialog box *I/O Type*.
The parameter **Slot** is set to 1 by the PG2000 programming system, and therefore does not have to be set by the user.

Channel 1 - Digital Input

2003-I/O	
Module adr.	3
Module type	Digit. In
Channel no.	1
Slot	1

Channel 2 - Digital Input

2003-I/O	
Module adr.	3
Module type	Digit. In
Channel no.	2
Slot	1

Channel 3 - Digital Output

2003-I/O	
Module adr.	3
Module type	Digit. Out
Channel no.	3
Slot	1

Channel 4 - Digital Output

2003-I/O	
Module adr.	3
Module type	Digit. Out
Channel no.	4
Slot	1

Module Status

2003-I/O	
Module adr.	3
Module type	Status In
Channel no.	0
Slot	1

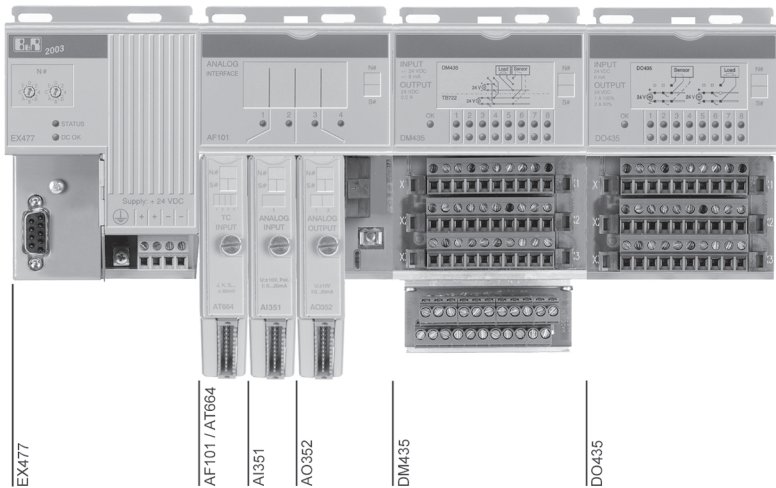# 4 VARIABLE DECLARATION WITH REMOTE I/O SLAVE

4.1 GENERAL INFORMATION

Entering the variable declaration is described in the "PG2000 Software User's Manual".

Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

The examples described here are especially suited to the specific characteristics when using a Remote I/O Slave.

The variable declaration is described using the hardware configuration shown below. The Remote I/O Slave has node number 1.

EX477   AF101 / AT664   AI351   AO352   DM435   DO435

4.2 MODULE ASSIGNMENTS

AT664	
Channels 1 - 4	All channels are assigned

AI351	
Channel 1	Voltage measurement without threshold switch

AO352	
Channels 1 -2	All channels are assigned

DM435	
Digital outputs	
Channels 1 -2	Assigned
Channels 3 -8	Free
Digital inputs	
Channels 1 -2	Assigned
Channels 3 -8	Free

DO435	
Channels 1 -2	Digital inputs
Channels 3 - 4	Digital outputs
Channels 5 -8	Free

4.3 TASK OVERVIEW

The application program is divided into the following tasks:

4.4 REMOTE I/O LIBRARY

In order to program a Remote I/O Slave, the Remote I/O Library must be imported in the project database.
The function, *Import Library* in the *File* menu is provided for this purpose.
The Remote I/O Library must be entered in the GDM as a system module.

4.5 AF101

This task is used to process the screw-in modules.

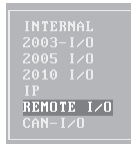### 4.5.1 Variable Declaration

4.5.2 AT664

Four channels are used on the AT664.

Select "REMOTE I/O" from the *I/O Type* pop-up selection window. Then a dialog box will be opened.

The following settings are to be made in the dialog box *I/O Type*:

Channel 1

REMOTE I/O

Master	1
Slave	1
Slave type	2003
Module adr.	1
Module type	Analog In
Channel no.	1
Slot	1

Channel 2

REMOTE I/O

Master	1
Slave	1
Slave type	2003
Module adr.	1
Module type	Analog In
Channel no.	2
Slot	1

Channel 3

REMOTE I/O

Master	1
Slave	1
Slave type	2003
Module adr.	1
Module type	Analog In
Channel no.	3
Slot	1

Channel 4

REMOTE I/O

Master	1
Slave	1
Slave type	2003
Module adr.	1
Module type	Analog In
Channel no.	4
Slot	1

Module Status

REMOTE I/O

Master	1
Slave	1
Slave type	2003
Module adr.	1
Module type	Transp.In
Channel no.	24
Slot	1

Module Type

REMOTE I/O

Master	1
Slave	1
Slave type	2003
Module adr.	1
Module type	Transp.In
Channel no.	28
Slot	1

Module Configuration

REMOTE I/O

Master	1
Slave	1
Slave type	2003
Module adr.	1
Module type	Transp.Out
Channel no.	28
Slot	1

N#
S#

1 2 3 4
TC
INPUT

J, K, S,....
± 60mV

AT664

4.5.3 AI351

For the AI351, the following settings are to be made in the dialog box *I/O Type*:

Channel 1

REMOTE I/O

Master	1
Slave	1
Slave type	2003
Module adr.	1
Module type	Analog In
Channel no.	1
Slot	2

Module Status

REMOTE I/O

Master	1
Slave	1
Slave type	2003
Module adr.	1
Module type	Transp.In
Channel no.	24
Slot	2

Module Type

REMOTE I/O

Master	1
Slave	1
Slave type	2003
Module adr.	1
Module type	Transp.In
Channel no.	28
Slot	2

Module Configuration

REMOTE I/O

Master	1
Slave	1
Slave type	2003
Module adr.	1
Module type	Transp.Out
Channel no.	28
Slot	2

N#
S#

1
ANALOG
INPUT

U:±10V, Pot.
I: 0...20mA

AI351

4.5.4 AO352

Two channels are used on the AO352. The following settings are to be made in the *I/O Type* dialog box:

	Channel 1	Channel 2
	Channel 1	**Channel 2**

Channel 1 — REMOTE I/O

Master	1
Slave	1
Slave type	2003
Module adr.	1
Module type	Analog Out
Channel no.	1
Slot	3

Channel 2 — REMOTE I/O

Master	1
Slave	1
Slave type	2003
Module adr.	1
Module type	Analog Out
Channel no.	2
Slot	3

Module Status — REMOTE I/O

Master	1
Slave	1
Slave type	2003
Module adr.	1
Module type	Transp.In
Channel no.	24
Slot	3

Module Type — REMOTE I/O

Master	1
Slave	1
Slave type	2003
Module adr.	1
Module type	Transp.In
Channel no.	28
Slot	3

4.6 DM435

On the DM435, two channels are used as digital inputs and two channels are used as digital outputs.

4.6.1 Variable Declaration

dm435.vd — [↓][↕]

Name	Scope	Data Type	Length	I/O-Type	Init Value	Long Nam
digin_1	tc_global	BIT	1	IR3.1.1.2.1	----------	
digin_2	tc_global	BIT	1	IR3.1.1.2.2	----------	
digout_1	tc_global	BIT	1	QR3.1.1.2.1	* remanent	
digout_2	tc_global	BIT	1	QR3.1.1.2.2	* remanent	
dm435stat	tc_global	BYTE	1	ISR3.1.1.2.0	----------	

1/5   5

4.6.2 I/O Type

The settings shown below are to be made in the dialog box *I/O Type*.
The parameter **Slot** is set to 1 by the PG2000 programming system, and therefore does not have to be set by the user.

Channel 1 - Digital Input

REMOTE I/O	
Master	1
Slave	1
Slave type	2003
Module adr.	2
Module type	Digit. In
Channel no.	1
Slot	1

Channel 2 - Digital Input

REMOTE I/O	
Master	1
Slave	1
Slave type	2003
Module adr.	2
Module type	Digit. In
Channel no.	2
Slot	1

Channel 1 - Digital Output

REMOTE I/O	
Master	1
Slave	1
Slave type	2003
Module adr.	2
Module type	Digit. Out
Channel no.	1
Slot	1

Channel 2 - Digital Output

REMOTE I/O	
Master	1
Slave	1
Slave type	2003
Module adr.	2
Module type	Digit. Out
Channel no.	2
Slot	1

Module Status

REMOTE I/O	
Master	1
Slave	1
Slave type	2003
Module adr.	2
Module type	Status In
Channel no.	0
Slot	1

4.7 DO435

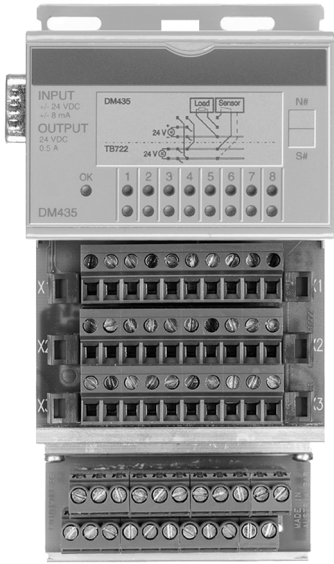On the DO435, two channels are used as digital inputs and two channels are used as digital outputs. The output mask is defined with the variable „outs".
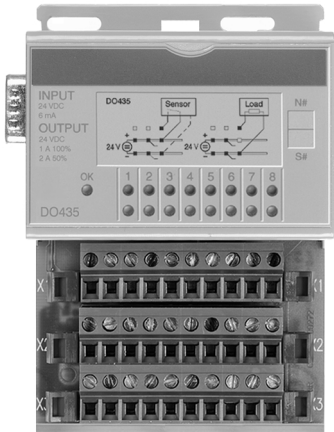
4.7.1 Variable Declaration

```
o┌──────────────────── do435.vd ────────────────[↓][↕]┐
 │ Name      │ Scope    │ Data Type │ Length │ I/O-Type     │ Init Value │ Long Nam│
 │                                                                                 │
 │ dig_in1   │tc_global │BIT        │1       │IR3.1.1.3.1   │----------  │         │
 │ dig_in2   │tc_global │BIT        │1       │IR3.1.1.3.2   │----------  │         │
 │ dig_out3  │tc_global │BIT        │1       │QR3.1.1.3.3   │* remanent  │         │
 │ dig_out4  │tc_global │BIT        │1       │QR3.1.1.3.4   │* remanent  │         │
 │ do435_stat│tc_global │BYTE       │1       │ISR3.1.1.3.0  │----------  │         │
 │ outs      │tc_global │BYTE       │1       │QSR3.1.1.3.0  │% 0000 1100 │         │
 │                                                                                 │
 └1/6──────────────────────────────────────────────────────────────────5─┘
```

4.7.2 I/O Type

The settings shown below are to be made in the dialog box *I/O Type*.
The parameter **Slot** is set to 1 by the PG2000 programming system, and therefore does not have to be set by the user.

Channel 1 - Digital Input

```
┌─REMOTE I/O─────────┐
│ Master       1     │
│ Slave        1     │
│ Slave type   2003  │
│ Module adr.  3     │
│ Module type  Digit. In │
│ Channel no.  1     │
│ Slot         1     │
│                   1│
└────────────────────┘
```

Channel 2 - Digital Input

```
┌─REMOTE I/O─────────┐
│ Master       1     │
│ Slave        1     │
│ Slave type   2003  │
│ Module adr.  3     │
│ Module type  Digit. In │
│ Channel no.  2     │
│ Slot         1     │
│                   1│
└────────────────────┘
```

Channel 3 - Digital Output

```
┌─REMOTE I/O─────────┐
│ Master       1     │
│ Slave        1     │
│ Slave type   2003  │
│ Module adr.  3     │
│ Module type  Digit. Out │
│ Channel no.  3     │
│ Slot         1     │
│                   1│
└────────────────────┘
```

Channel 4 - Digital Output

```
┌─REMOTE I/O─────────┐
│ Master       1     │
│ Slave        1     │
│ Slave type   2003  │
│ Module adr.  3     │
│ Module type  Digit. Out │
│ Channel no.  4     │
│ Slot         1     │
│                   1│
└────────────────────┘
```

Module Status

```
┌─REMOTE I/O─────────┐
│ Master       1     │
│ Slave        1     │
│ Slave type   2003  │
│ Module adr.  3     │
│ Module type  Status In │
│ Channel no.  0     │
│ Slot         1     │
│                   1│
└────────────────────┘
```

Output Mask

```
┌─REMOTE I/O─────────┐
│ Master       1     │
│ Slave        1     │
│ Slave type   2003  │
│ Module adr.  3     │
│ Module type  Status Out │
│ Channel no.  0     │
│ Slot         1     │
│                   1│
└────────────────────┘
```

5 VARIABLE DECLARATION WITH CAN SLAVE

5.1 GENERAL INFORMATION

Entering the variable declaration is described in the "PG2000 Software User's Manual".
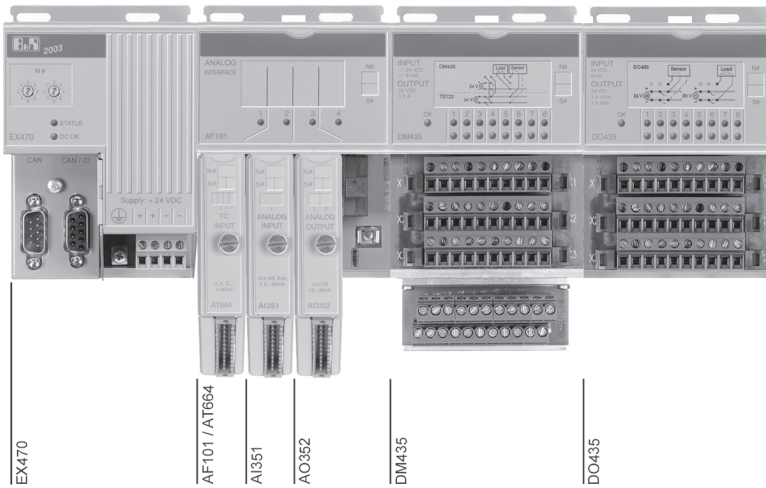
Automation Studio™ Support: See Automation Studio™ Help starting with V 1.40

The examples described here are especially suited to the specific characteristics when using a CAN Slave.

Take note of the following points:

- Accessing configuration words takes place using the CAN command function block CANIOcmd().
- Additional information concerning the CAN command function block can be found in the CANIO online help.
- The variable declaration for a CAN Slave functions **starting with PG2000 Programming System Version 2.35**.

The variable declaration is described using the hardware configuration shown below:

5.2 MODULE ASSIGNMENTS

AT664	
Channels 1 - 4	All channels are assigned

AI351	
Channel 1	Voltage measurement without threshold switch

AO352	
Channels 1 -2	All channels are assigned

DM435	
Digital outputs	
Channels 1 -2	Assigned
Channels 3 -8	Free
Digital inputs	
Channels 1 -2	Assigned
Channels 3 -8	Free

DO435	
Channels 1 -2	Digital inputs
Channels 3 - 4	Digital outputs
Channels 5 -8	Free

5.3 TASK OVERVIEW

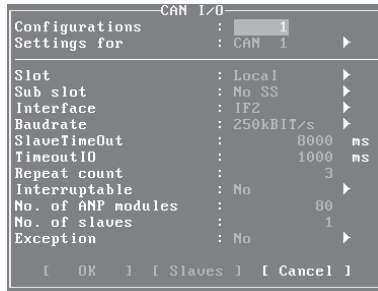The application program is divided into the following tasks:

```
┌─────────────────────── 2003 ────────────────────┐
o                                                  
   ┌──TC#4──┐       ┌──TC#4──┐       ┌──TC#4──┐
   │ af101  │       │ dm435  │       │ do435  │
   └────────┘       └────────┘       └────────┘
   ┌──B&R───┐       ┌──B&R───┐       ┌──EXC───┐
   │ canio  │       │sysconf │       │canioexc│
   └────────┘       └────────┘       └────────┘
```
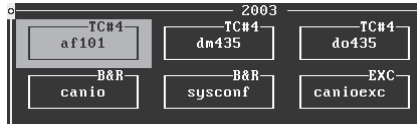
5.4 CANIO LIBRARY

In order to program a CAN Slave, the CANIO Library must be imported into the project data bank. The function, *Import Library* in the *File* menu is provided for this purpose.
The CANIO master must be entered in the GDM as system module.

5.5 SYSTEM CONFIGURATION

The system configuration must be entered in the GDM. The following settings must be made for CAN I/O:

```
┌───────────────CAN I/O────────────────┐
│Configurations      :      1          │
│Settings for        : CAN  1        ▶ │
│                                      │
│Slot                : Local         ▶ │
│Sub slot            : No SS         ▶ │
│Interface           : IF2          ▶ │
│Baudrate            : 250kBIT/s     ▶ │
│SlaveTimeOut        :        8000  ms │
│TimeoutIO           :        1000  ms │
│Repeat count        :           3     │
│Interruptable       : No           ▶ │
│No. of ANP modules  :          80     │
│No. of slaves       :           1     │
│Exception           : No           ▶ │
│                                      │
│   [  OK   ]  [ Slaves ]  [ Cancel ]  │
└──────────────────────────────────────┘
```

5.6 ERROR EVALUATION

Errors, such as a node dropping out, alarm messages etc. can be evaluated in an exception task (exception number 170).

```
Project: canio_bs                                    File: canioexc.src

;###########################################################################
;##  Error evaluation for CANIO in exception task                        ##
;###########################################################################

;Check reason for this exception
CANIOexc(1, status, exctyp, busnr, nodenr, errcode, addcode)

;Evaluation:
if (busnr = 1) and (nodenr = 1) then          ;Node ?
  case exctyp of        ;Type of exception
     action 1:          ;1 ... Node drops out
        node_act = 0                           ;-> is no longer active
                                               ;Message to ask „AF101"
     endaction
     action 2:          ;2 ... Node is active again
        node_act = 1                     ;Active
        init_ok  = 0                     ;Reinitialize ScrM
                                               ;Message to task „AF101"
     endaction
     action 3:          ;3 ... Alarm
        ;Alarm message from node
        ; <errcode> and <addcode> can be evaluated here
     endaction
     action 4:          ;4 ... BUSOFF
        ;Error on CAN Bus, possibly a loose cable
     endaction
     elseaction:
        ;Exception type reserved for future updates
     endaction
  endcase
endif
```

5.7 AF101

This task is used to process the screw-in modules.

5.7.1 INIT-SP

```
Project: canio_bs                                      File: InitSP : af101

;#############################################################################
;##              Commands to initialize the screw-in modules              ##
;#############################################################################
;Only the screw-in modules have to be configured, the configuration of the
;digital output masks and the timeouts are taken care of by the CANIO driver

comtab[0].code=3          ;Code
comtab[0].comcode=11      ;Write operating parameters
comtab[0].param1 =28      ;CFG word 14
comtab[0].param2 =0       ;Module 1, Slots 1+2
comtab[0].data   =$00010000  ;CFG word 14 Slot 1 = 1
                             ; -> AT664 J type sensor
                             ;CFG word 14 Slot 2 = 0
                             ; -> AI351 default setting

comtab[1].code=4          ;Code
comtab[1].comcode=12      ;Activate operating parameters
comtab[1].param1 =0       ;Always 0
comtab[1].param2 =0       ;Always 0
comtab[1].data   =0       ;Always 0

com = 0         ;Begin with first command
no_com = 2      ;Total of 2 commands

init_ok = 0     ;Initialization of node not yet complete
node_act = 1    ;Node is active

;#############################################################################
;## Declaration of IO variables, they would be linked in a more logical    ##
;## manner in real applications, any the variable names can be used.       ##
;#############################################################################
anain1_1_1 = anain1_1_1    ;Channel 1 on AT664
anain1_1_2 = anain1_1_2    ;Channel 2 on AT664
anain1_1_3 = anain1_1_3    ;Channel 3 on AT664
anain1_1_4 = anain1_1_4    ;Channel 4 on AT664

anain1_2_1 = anain1_2_1    ;Channel 1 on AI351

anaout_1   = anaout_1      ;Channel 1 on AO352
anaout_2   = anaout_2      ;Channel 2 on AO352
```

5.7.2 Cyclic Program

```
Project: canio_bs                                      File: af101.src

if node_act = 1 then     ;Node is active
   if init_ok = 0 then      ;Node not yet initialized
       busnr = 1               ;Bus number from sysconf
       nodenr = 1             ;Node number set on EX470 or EX770
       ;comtab               ;Command table see INIT-SP
       ;response             ;Buffer for command response
       CANIOcmd(1, busnr, nodenr, adr(comtab[com]), adr(response), status)
       if status = 0 then                    ;Command carried out
          if (bit_tst(response.code,7)=1) then ;Error carrying out command
              ;Evaluate error message from command response here
          else                                   ;otherwise
              com = com + 1        ;  -> next command
          endif
       else if status = 8979 then  ;Error message for node inactive ?
          node_act = 0             ;  -> Node inactive !
       else if status<>$FFFF then   ;Error other than $FFFF=busy
                                   ;->Evaluate here
       endif
       if com >= no_com then     ;All command carried out
          init_ok = 1           ;  -> Init complete
          com=0                 ;Start from beginning with next Init
       endif
   else                      ;Node is active and initialized

       ;Cyclic program here
       ;....

   endif
endif
```

5.7.3 Variable Declaration

5.7.4 Data Type

A structure is defined for the variables "response" and "comtab" using the data type *typedef*. After entering the structure name *command*, the elements of the structure variable can be entered in a dialog box.

```
o── Type Decl:command ──[↓][↕]
 Component │ Data Type │ Length
 code       BYTE        1
 comcode    BYTE        1
 param1     BYTE        1
 param2     BYTE        1
 data       LONG        1
```

5.7.5 AT664

Four channels are used on the AT664.

Select "CAN-I/O" from the pop-up window *I/O Type*. Then a dialog box will be opened.

```
 INTERNAL
 2003-I/O
 2005 I/O
 2010 I/O
 IP
 REMOTE I/O
 CAN-I/O
```

The following settings are to be made in the dialog box *I/O Type*:

Channel 1

```
o────CAN-I/O────
 Bus no.       1
 Slave         1
 Module adr.   1
 Module typ    Analog In
 Channel no.   1
 Slot          1
                        1
```

Channel 2

```
o────CAN-I/O────
 Bus no.       1
 Slave         1
 Module adr.   1
 Module typ    Analog In
 Channel no.   2
 Slot          1
                        1
```

Channel 3

```
o────CAN-I/O────
 Bus no.       1
 Slave         1
 Module adr.   1
 Module typ    Analog In
 Channel no.   3
 Slot          1
                        1
```

Channel 4

```
o────CAN-I/O────
 Bus no.       1
 Slave         1
 Module adr.   1
 Module typ    Analog In
 Channel no.   4
 Slot          1
                        1
```

5.7.6 AI351

For the AI351, the following settings are to be made in the dialog box *I/O Type*:

Channel 1

CAN-I/O	
Bus no.	1
Slave	1
Module adr.	1
Module typ	Analog In
Channel no.	1
Slot	2

5.7.7 AO352

Two channels are used on the AO352. The following settings are to be made in the *I/O Type* dialog box:

Channel 1

CAN-I/O	
Bus no.	1
Slave	1
Module adr.	1
Module typ	Analog Out
Channel no.	1
Slot	3

Channel 2

CAN-I/O	
Bus no.	1
Slave	1
Module adr.	1
Module typ	Analog Out
Channel no.	2
Slot	3

5.8 DM435

On the DM435, two channels are used as digital inputs and two channels are used as digital outputs.

5.8.1 Variable Declaration

```
┌o────────────────────────── dm435.vd ──────────────────[↓][↕]─┐
│                                                               │
│ Name       Scope      Data Type  Length  I/O-Type   Init Value  Long Nam│
│                                                               │
│ digin_2_1  tc_global  BIT        1        IC1.1.2.1  ----------│
│ digin_2_2  tc_global  BIT        1        IC1.1.2.2  ----------│
│ digout_2_1 tc_global  BIT        1        QC1.1.2.1  * remanent│
│ digout_2_2 tc_global  BIT        1        QC1.1.2.2  * remanent│
│                                                               │
└1/4────────────────────────────────────────────────────────3──┘
```

5.8.2 I/O Type

The settings shown below are to be made in the dialog box *I/O Type*.
The parameter **Slot** is set to 1 by the PG2000 programming system, and therefore does not have to be set by the user.

Channel 1 - Digital Input

```
┌o───────── CAN-I/O ─────────┐
│ Bus no.        1           │
│ Slave          1           │
│ Module adr.    2           │
│ Module typ     Digit. In   │
│ Channel no.    1           │
│ Slot           1           │
└──────────────────────────1─┘
```

Channel 2 - Digital Input

```
┌o───────── CAN-I/O ─────────┐
│ Bus no.        1           │
│ Slave          1           │
│ Module adr.    2           │
│ Module typ     Digit. In   │
│ Channel no.    2           │
│ Slot           1           │
└──────────────────────────1─┘
```
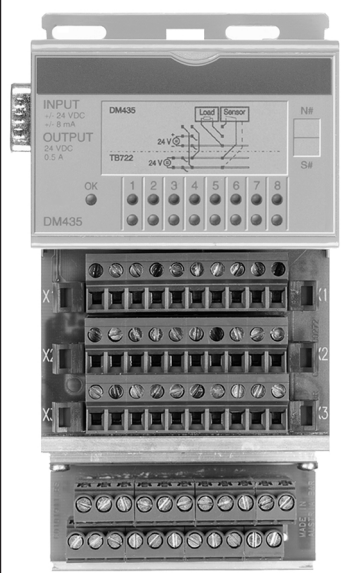
Channel 1 - Digital Output

```
┌o───────── CAN-I/O ─────────┐
│ Bus no.        1           │
│ Slave          1           │
│ Module adr.    2           │
│ Module typ     Digit. Out  │
│ Channel no.    1           │
│ Slot           1           │
└──────────────────────────1─┘
```

Channel 2 - Digital Output

```
┌o───────── CAN-I/O ─────────┐
│ Bus no.        1           │
│ Slave          1           │
│ Module adr.    2           │
│ Module typ     Digit. Out  │
│ Channel no.    2           │
│ Slot           1           │
└──────────────────────────1─┘
```

5.9 DO435

On the DO435, two channels are used as digital inputs and two channels are used as digital outputs. The output mask is defined automatically by the CANIO master.
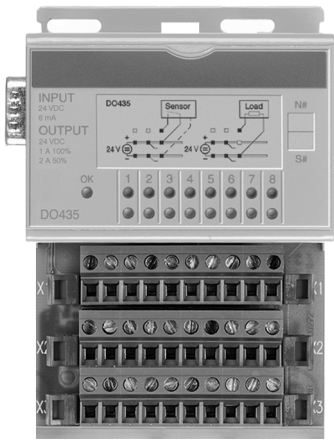
5.9.1 Variable Declaration

```
┌────────────────────── do435.vd ──────────────────────[↓][↕]┐
│                                                              │
│ Name        Scope       Data Type  Length  I/O-Type     Init Value  Long Nam│
│                                                                     │
│ digin_3_1   tc_global   BIT        1       IC1.1.3.1    ----------  │
│ digin_3_2   tc_global   BIT        1       IC1.1.3.2    ----------  │
│ digout_3_3  tc_global   BIT        1       QC1.1.3.3    * remanent  │
│ digout_3_4  tc_global   BIT        1       QC1.1.3.4    * remanent  │
│                                                                     │
└1/4─────────────────────────────────────────────────────────3──────┘
```

5.9.2 I/O Type

The settings shown below are to be made in the dialog box *I/O Type*.
The parameter **Slot** is set to 1 by the PG2000 programming system, and therefore does not have to be set by the user.

Channel 1 - Digital Input

```
┌─────────CAN-I/O─────────┐
│ Bus no.      1          │
│ Slave        1          │
│ Module adr.  3          │
│ Module typ   Digit. In  │
│ Channel no.  1          │
│ Slot         1          │
│                        1│
└─────────────────────────┘
```

Channel 2 - Digital Input

```
┌─────────CAN-I/O─────────┐
│ Bus no.      1          │
│ Slave        1          │
│ Module adr.  3          │
│ Module typ   Digit. In  │
│ Channel no.  2          │
│ Slot         1          │
│                        1│
└─────────────────────────┘
```

Channel 3 - Digital Output

```
┌─────────CAN-I/O─────────┐
│ Bus no.      1          │
│ Slave        1          │
│ Module adr.  3          │
│ Module typ   Digit. Out │
│ Channel no.  3          │
│ Slot         1          │
│                        1│
└─────────────────────────┘
```

Channel 4 - Digital Output

```
┌─────────CAN-I/O─────────┐
│ Bus no.      1          │
│ Slave        1          │
│ Module adr.  3          │
│ Module typ   Digit. Out │
│ Channel no.  4          │
│ Slot         1          │
│                        1│
└─────────────────────────┘
```

6 CAN ID ACCESS

6.1 GENERAL INFORMATION

Access via CAN Identifier is used if the slave is being controlled by a device from another manufacturer.

The following hardware configuration is used for the example:

## 6.2 MODULE ASSIGNMENTS

AT664	
Channels 1 - 4	All channels are assigned

AI351	
Channel 1	Voltage measurement without threshold switch

AO352	
Channels 1 -2	All channels are assigned

DM435	
Digital outputs	
Channels 1 -2	Assigned
Channels 3 -8	Free
Digital inputs	
Channels 1 -2	Assigned
Channels 3 -8	Free

DO435	
Channels 1 -2	Digital inputs
Channels 3 - 4	Digital outputs
Channels 5 -8	Free

6.3 TASK OVERVIEW

The application program is divided into the following tasks:
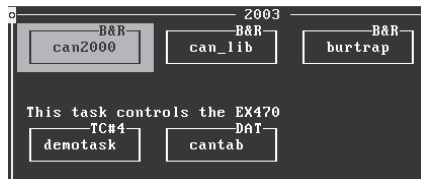
6.4 LIBRARIES

The B&R TRAP Library and the CAN Library must be imported into the project database. The function, *Import Library* in the *File* menu is provided for this purpose.

6.5 SYSTEM MODULES

The CAN2000, B&R TRAP Library and the CAN Library must be entered in the GDM as system modules.

6.6 DEMO TASK

6.6.1 INIT-SP

```
Project: ex470tst                                         File: InitSP : demotask

step=0    ;Start at the beginning after Init

;CAN interface initialization
enable = 1
baud_rate= 25    ;250k Baud
cob_no=20        ;Number of CAN objects to be reserved
error_adr=adr(CAN_err) ;Address for error messages
CANopen(enable,baud_rate,cob_no,error_adr,0,0,us_ident,status)

if status=0 then
   canop_OK=1
endif

error=DA_ident(„cantab",0,adr(ident))

if error=0 then
   error=DA_info(ident,adr(D_Adr),adr(D_Len),0)
endif

D_no=D_Len/72    ;72 Byte per table entry

if (error=0) and (canop_OK=1) then
   CANdftab(1,us_ident,D_Adr,word(D_no),tab_ident,statusdf)
   if status<>0 then
      canop_OK=0
   endif
else
   canop_OK=0
endif

;Reset all trigger variables
Id_Node_tr=0
Command1_tr=0
Resp_B1_tr=0
DigIn1_tr=0
DigOut1_tr=0
AnaIn1_tr=0
AnaIn2_tr=0
AnaOut1_tr=0

comtab[0].code=2        ;Code
comtab[0].comcode=11    ;Write operating parameters
comtab[0].param1 =14    ;Output mask
comtab[0].param2 =0     ;Modules 1-4
comtab[0].data   =$00FF0C00  ;Module #2: Channels 1-8, Module #3: Channels 3+4
   ;On the DM435, all channels are declared as outputs
   ;On the DO435, the first two channels are used as outputs

comtab[1].code=3        ;Code
comtab[1].comcode=11    ;Write operating parameters
comtab[1].param1 =28    ;CFG word 14
comtab[1].param2 =0     ;Module 1, Slots 1+2
comtab[1].data   =$00010000  ;CFG word 14 Slot 1 = 1
                             ; -> AT664 J type sensor
                             ;CFG word 14 Slot 2 = 0
                             ; -> AI351 default setting
```

```
comtab[2].code=4          ;Code
comtab[2].comcode=12      ;Activate operating parameters
comtab[2].param1 =0       ;Always 0
comtab[2].param2 =0       ;Always 0
comtab[2].data   =0       ;Always 0

com=0      ;Begin with the first command
no_com=3   ;Number of commands is 3

Id_Node[0]=0 ;So this variable is declared
```

6.6.2 Cyclic Program

```
Project: ex470tst                                      File: demotask.src

;This task takes care of IO handling for an EX47 controller
;An AF101, a DM435 and a DO435 module are inserted on the
;   EX470 controller, the number switches are set to 01
;An AT664 is inserted in the first slot of the AF101
;An AI351 is inserted in the second slot of the AF101
;An AO352 is inserted in the third slot of the AF101

;All IO data is placed in PCC global variables and can therefore
;be used by other tasks.

if canop_OK=1 then
   CANrwtab(1,tab_ident,status)
endif

select step
   when canop_OK=0 ;If Init-SP has problems
       next ERROR
   when Id_Node_tr=1 ;If Identify Node is received, a new node is present
                   ;on the network -> new initialization
       Id_Node_tr = 0
       next WAIT_awake

;Request the status of the EX470 cyclically until in runs
   state WAIT_awake
       Command1.code=1       ;Code
       Command1.comcode=0    ;Read slave status
       Command1.param1=0     ;
       Command1.param2=0     ;00->SlaveStatus
       Command1_tr=1              ;Check EX470 cyclically
       when (Resp_B1_tr=1) and (Resp_B1.data=0)
          Command1_tr=0
          Resp_B1_tr=0
          com=0
          next INIT              ;If response is there and response data=0 then
                                 ;   the EX470 is active and without errors
       if Resp_B1_tr=1 then   ;If response is there and response data<>0 then
          Resp_B1_tr=0        ;   acknowledge and wait
       endif

;EX470 is now ready -> Start initialization
```

Chapter 4
Module Addressing

```
    state INIT
       Command1.code=    comtab[com].code      ;Copy next command in buffer
       Command1.comcode=comtab[com].comcode
       Command1.param1= comtab[com].param1
       Command1.param2= comtab[com].param2
       Command1.data=    comtab[com].data
       Command1_tr=1
       if Resp_B1_tr=1 then
          if Resp_B1.code=Command1.code then
             com=com+1                                 ;If response received
          endif                                        ;then next command
          Resp_B1_tr=0
       endif
       when com>=no_com          ;All initialization commands carried out
          oldOutA1=anaout_1+1     ;Outputs are only handled if a difference
          oldOutA2=anaout_2+1     ;is found.
          oldOutD1=OutputD1+1     ;We have created a difference here
          oldOutD2=OutputD2+1
          next CYCLIC

;Cyclic section
    state CYCLIC
       ;Handle analog inputs
       if AnaIn1_tr=1 then    ;Receive analog IN object from AT664
          ;Analog data comes in Intel Format -> Swap
          anain1_1_1=lsl(WORD(AnaIn1.word1.high),8)+AnaIn1.word1.low
          anain1_1_2=lsl(WORD(AnaIn1.word2.high),8)+AnaIn1.word2.low
          anain1_1_3=lsl(WORD(AnaIn1.word3.high),8)+AnaIn1.word3.low
          anain1_1_4=lsl(WORD(AnaIn1.word4.high),8)+AnaIn1.word4.low
          AnaIn1_tr=0        ;Reset trigger
       endif

       if AnaIn2_tr=1 then    ;Receive analog IN object from AI351
          ;Analog data comes in Intel Format -> Swap
          anain1_2_1=lsl(WORD(AnaIn2.word1.high),8)+AnaIn2.word1.low
          AnaIn2_tr=0        ;Reset trigger
       endif

       ;Handle digital inputs
       if DigIn1_tr=1 then      ;Receive digital IN object
          ;Module #2, Channel #1
          digin_2_1=bit_tst(DigIn1.byte2,0)
          ;Module #2, Channel #2
          digin_2_2=bit_tst(DigIn1.byte2,1)
          ;Module #3, Channel #1
          digin_3_1=bit_tst(DigIn1.byte3,0)
          ;Module #3, Channel #2
          digin_3_2=bit_tst(DigIn1.byte3,1)
          DigIn1_tr=0          ;Reset trigger
       endif

       ;Handle analog outputs
```

```
          if anaout_1<>oldOutA1 then      ;Has output data on channel 1 changed ?
             ;Copy swapped output data in CAN data buffer
             AnaOut1.word1.low=BYTE(anaout_1)
             AnaOut1.word1.high=BYTE(lsr(anaout_1,8))
             oldOutA1=anaout_1
             AnaOut1_tr=1
          endif

          if anaout_2<>oldOutA2 then      ;Has output data on channel 2 changed ?
             ;Copy swapped output data in CAN data buffer
             AnaOut1.word2.low=BYTE(anaout_2)
             AnaOut1.word2.high=BYTE(lsr(anaout_2,8))
             oldOutA2 = anaout_2
             AnaOut1_tr=1              ;Set trigger
          endif

          ;Handle digital outputs
          OutputD1=0
          OutputD2=0
          if digout_2_1=1 then
             OutputD1 = bit_set(OutputD1, 0)
          endif
          if digout_2_2=1 then
             OutputD1 = bit_set(OutputD1, 1)
          endif
          if digout_3_3=1 then
             OutputD2 = bit_set(OutputD2, 2)
          endif
          if digout_3_4=1 then
             OutputD2 = bit_set(OutputD2, 3)
          endif

          if (OutputD1<>oldOutD1) or (OutputD2<>oldOutD2) then ;Dig out data changed
             DigOut1.byte2=OutputD1    ;Output data in CAN data buffer
             DigOut1.byte3=OutputD2
             oldOutD1=OutputD1
             oldOutD2=OutputD2
             DigOut1_tr=1              ;Set trigger
          endif

          if time>=10 then       ;Send outputs every 10 cycles
             time=0               ;because otherwise they drop out after timeout
             DigOut1_tr=1
             AnaOut1_tr=1
          else
             time=time+1          ;Increment time counter
          endif

          when 1=1
             next CYCLIC

   ;If error
      state ERROR
         when canop_OK=0
            next ERROR
   endselect
```

6.6.3 Cantab

```
Project: ex470tst                                           File: cantab.dat

;CAN-ID   ,Name of the buffer variables      ,Name of the trigger variables    ,0->read
;                                                                              1->write
;         ,"12345678901234567890123456789012","12345678901234567890123456789012",$00000000
;Identify node
$000007E6,"Id_Node                         ","Id_Node_tr                      ",$00000000
;Command ID for node #1
$0000061E,"Command1                        ","Command1_tr                     ",$00000001
;Command response ID for node #1
$0000065E,"Resp_B1                         ","Resp_B1_tr                      ",$00000000
;Digital IN for node #1
$0000011E,"DigIn1                          ","DigIn1_tr                       ",$00000000
;Digital OUT to node #1
$0000019E,"DigOut1                         ","DigOut1_tr                      ",$00000001
;Analog IN for node  #1 Module #1 Slot #1 (AT664)
$0000021E,"AnaIn1                          ","AnaIn1_tr                       ",$00000000
;Analog IN for node  #1 Module #1 Slot #2 (AI351)
$0000021F,"AnaIn2                          ","AnaIn2_tr                       ",$00000000
;Analog OUT to node  #1 Module #1 Slot #3 (Ao352)
$00000420,"AnaOut1                         ","AnaOut1_tr                      ",$00000001
```

CHAPTER 5
CAN BUS CONTROLLER FUNCTIONS

1 GENERAL INFORMATION

⚠️ **This description refers to operating system version xx.27 of the CAN bus controller.**

B&R 2003 CAN bus controllers are suitable for operation with B&R devices and also in other CAN networks.

A CAN bus controller has essentially the following functions:

- Initializing the station from the time the it is switched on until active operation in the CAN network.
- Evaluating and sending input states
- Receiving and switching the outputs
- Error reaction during power failure or local problems.
- Setting and changing operating parameters using the network and/or configuration memory ME770 (internally in the EX270 bus controller).

The remote I/O system B&R 2003 is based on the existing CiA/CAL network concept.

The CAN bus controller can be operated in class 0, 1 and 2 networks.

1.1 ABBREVIATIONS AND TERMS

This section contains explanations of some of the terms you will come across in this chapter. Further explanations for individual terms can be found in the relevant CAN network literature.

Term	Description
CiA	Unification of European users of CAN networks.
CAL	CAL describes the application layer of the CAN protocol according to CiA specifications.
NMT	NMT ... Network management An NMT master (specific station within the network) takes over the management of the CAN network. It has to recognize, transfer, remove, and configure network nodes, and carry out "life guarding".
DBT	DBT ... Distributor service element A DBT master determines the identifier for sending messages over the network (also see "Allocating the CAN Identifier using a DBT Master").
Identifier (ID)	IDs allow the messages on the bus to be differentiated between (the smaller the number, the higher its priority on the bus). Using an identifier, a specific object can be addressed in a network. Two objects are not allowed to have the same identifier. Digital inputs and outputs as well as analog channels can be read or written to using an identifier. If a particular identifier is used, commands can be sent to a station or the operating parameters can be changed.

Term	Description
CAN V. 2.0A	Only 11 bit identifiers are allowed on the network.
CAN V. 2.0B	Both 11 bit and 29 bit identifiers are allowed on the network.
BasicCAN	Station supports only one receive buffer
FullCAN	Station supports more than one receive buffer The **B&R SYSTEM 2003** supports both BasicCAN and FullCAN networks V. 2.0A and V. 2.0B. It uses 11 bit identifiers.
Client	Station which requests a service
Server	Station which carries out service
Life Guarding	An NMT master monitors NMT slave stations by communicating with reserved identifiers to see if the station is still in operation. A station also recognizes whether the NMT master is still working.
RTR	Access to an identifier with direct response handling, using the CAN 80527 function block.
Synchronization Jump Width	Factor corresponding to the extent to which the CAN function block can synchronize bit data. Synchronization is required because of oscillator tolerances.
Network Class	0 ... No network functionality (fixed identifier allocation, no life guarding) 1 ... Identifier and inhibit time allocation 2 ... Class 1 network plus Life Guarding
Module Name	For identifying the node in the network (7 ASCII characters)
Module ID	For identifying the network node
Inhibit Time	Time which must pass between sending the same identifier on the bus. In this way the bus load can be influenced by high priority objects (example: a fast changing input cannot overload the bus).

1.2 CIA/CAL NETWORK CLASS 0

The CAN bus controller is not assigned to an NMT or a DBT master when booting or in normal operation. This corresponds to a class 0 NMT master.

If a CAN bus controller is operated exclusively with B&R devices in a network, the class 0 CiA/CAL network is used by default. The CAN bus controller can also be operated with devices from other manufacturers.

The CAN bus controller automatically recognizes the network class in question. In a class 0 network, the bus controller automatically assigns a specific identifier to each object. Identifier allocation depends on the node number, the module slot and the type of input or output.

1.3 CIA/CAL NETWORK CLASS 1 OR 2

The B&R 2003 system can also be used in CiA/CAL class 1 and 2 networks together with devices from other manufacturers.

The CAN bus controller automatically recognizes the network class in question. In class 1 or 2 networks, a so called CiA/CAL NMT master must be available which coordinates the boot phase (standardized according to CiA/CAL) of the network and the individual nodes.

You can also define the object identifier using a CiA/CAL DBT master. It is therefore possible to use the CAN bus controller in any CAN bus system, if the CAN identifier allocation is within the identifier range released by the CiA (1-1760,2022).

1.4 REQUIREMENTS FOR A NMT MASTER

An NMT master must be able to carry out at least the following functions so that a B&R CAN bus controller can exist in a class 2 network:

Functionality	Description
Identify_Node	The master recognizes that a new node is present.
Identify_Remote_Node	Slave responds to this message with Identify_Node, if its module ID is within the range defined by the master.
Connect_Remote_Node	Transfer of the parameters Life Guard Time, Life Guard CAN ID.
Prepare_Remote_Node	Request the slave to create the object list (variable list).
Start_Node	Message to the slave that it can now switch to normal operation (status: Operating). Only then can inputs and outputs of this node be read or used.
Life_Guard	Cyclic "life sign monitoring" - the NMT master monitors if the slave is still in operation, by using a cyclic signal.

1.5 B&R DEVICES

The following B&R devices/components are capable of communicating with CAN bus controllers:

System / Module	Description
B&R SYSTEM 2003	
CP430	CPU
CP470	CPU
CP474	CPU
CP476	CPU
CP770	CPU
CP774	CPU
B&R SYSTEM 2005	
CP260	CPU + interface module IF621, IF671 or IF672 for insert slot
IF060	Interface module + interface module IF621, IF671 or IF672 for insert slot
IF260	Programmable interface processor + interface module IF621, IF671 or IF672 for insert slot
IP161	Programmable I/O processor
XP152	CPU – insert for power supply module
B&R SYSTEM 2010	
CP104	CPU
CP200	CPU
CP210	CPU
IF100	Interface module
IF101	Interface module
B&R SYSTEM 2000 Logic Scanner	
LS251	Logic Scanner CPU
PANELWARE	
C221	Intelligent controller module
PROVIT	
PROVIT 2000	Modular Industrial PC
PROVIT 5000	Modular Industrial PC
B&R Compact Controller	
COMPACT2	Compact controller with CAN Bus (model number BRCOMP2-0)

A detailed description of these devices can be found in the following manuals:

Model Number	Title
MASYS22005-E	B&R 2005 User's Manual
MASYS22010-E	B&R 2010 User's Manual
MASYS2LS-E	Logic Scanner LS251 User's Manual
MAPWHW-E	PANELWARE Hardware and Installation Manual
MAPRV2000-E	PROVIT 2000 User's Manual
MAPRV5000-E	PROVIT 5000 User's Manual
MABRCOMP1-0E	B&R Compact Controller User's Manual German/English

1.6 OPERATION USING FUNCTION BLOCKS

A CAN bus controller is operated using function blocks.

Function blocks are used to control a CAN bus controller on a B&R device from systems B&R 2003, B&R 2005, B&R 2010, B&R 2000 Logic Scanner or PANELWARE. These function blocks are already available in the PG2000 programming system in the form of a library. A description of the CAN library can be found in the following manual:

Model Number	Title
MASYS2LRM-E	Library Reference Manual starting with Version 2.0

2 OPERATING PARAMETERS

The behavior of a CAN bus controller depends on the operating parameters. The operating parameters can be set in various ways. The user can carry out any of the following:

- Use default values which are set independently by the CAN bus controller during the initialization phase.

- The operating parameters are stored in a nonvolatile program memory, and the CAN bus controller reads it from there.

- The operating parameters are transferred and activated from another station (client) to the CAN bus controller.

2.1 OVERVIEW

The operating parameters which are relevant to the user have been summarized in the following overview table. The table is divided into two groups.
The **bold** parameters can be changed and activated while the system is running. The others can only be changed if the program memory module ME770 is being used, or after the CAN bus controller has been restarted.

> **The value in column "No." is used by commands to read and write operating parameters (see Section "Commands - CAN Bus Controller").**
>
> **Take note of the example "Access via CAN Identifier" in Chapter 4 "Module Addressing".**

No.	Data Type	Value	Label	Description	Standard
0	Byte	0-99		Version number (not changed)	
1	Byte	0-63		Node number	
2	Byte	0-7	active	Baudrate	
3	Byte	0	CAL Table	Max. synchronization influences the selection of the baudrate from one of five tables each containing eight values.	0
		1-4	BuR value	0 : CAL table	
				1-4 : Bit timing as with COMPACT2 PLC	
4	reserved				
5	reserved				
6	Word	0-126*64	Units ms	Guard time	1024
7	Word	0-0xffff	Units Guard Time	Life time factor	1
8	Word	000...x		Packing Data:	0
		000..x.		Bit 0 0 ....pack digital inputs	
		000.x..		1 ....do not pack digital inputs	
		000x...		Bit 1 0 ....pack digital outputs	
				1 ....do not pack digital outputs	
				Bit 2 0 ....pack analog inputs	
				1 ....do not pack analog inputs	
				Bit 3 0 ....pack analog outputs	
				1 ....do not pack analog outputs	
		0...x00		Bit 8.....freeze all analog inputs at once	
		0..x.00		Bit 9.....freeze all analog outputs at once	
9	Long		Units µs	SYNC cycle time	100000
10	Long		Units µs	SYNC window time	50000
11	Word	0-65000	Units ms	Inhibit time for alarm object	5
12	Byte [m1..m8]	254	Asynchronous - cyclic	Behavior of digital outputs in operation	254
13	Word	0-65000	Units ms	Idle time for digital outputs	640
14	Byte [m1..m8]			Change mask for digital outputs	0
15	Byte [m1..m4][sl1..sl4]	254	Asynchronous - cyclic	Behavior of analog outputs in operation	254
16	Word	0-65000	Units ms	Idle time for analog outputs	640
17	Byte [m1..m8]	254	Asynchronous - cyclic	Behavior of digital inputs in operation	255
		255	Asynchronous - with change and if idle time passes		
18	Byte [m1..m8]		1 = edge formation	Change mask for digital inputs	0xFF
19	Word	0-65000	Units ms	Cycle time for digital inputs	100
20	Word	0-65000	Units ms	Idle time for digital inputs	640
21	Word	0-65000	Units ms	Inhibit time for digital inputs	5
22	Byte [m1..m4][sl1..sl4]	254	Asynchronous - cyclic	Behavior of analog inputs in operation	255
		255	Asynchronous - with change and if idle time passes		

No.	Data Type	Value	ID	Description	Standard
23	Byte [m1..m4][sl1..sl4]	1	> upper limit	Type of transmission trigger for analog inputs	4
		2	< lower limit		
		4	<> hysteresis		
		8	< neg. hysteresis		
		16	> pos. hysteresis		
24	int32 [m1..m4][sl1..sl4]			Value corresponds to trigger type for analog inputs	160
25	Word	0-65000	Units ms	Cycle time for analog inputs	100
26	Word	0-65000	Units ms	Idle time for analog inputs	640
27	Word	0-65000	Units ms	Inhibit time for analog Inputs	20
28	Word [m1..m4][sl1..sl4]			Parameter value for analog screw-in module	
29	Byte [m1..m8]			Module code for digital I/O modules	
30	Word [m1..m4][sl1..sl4]			Module code for screw-in modules (corresponds to configuration word 14, read)	
31	Byte [m1..m8]	ASCII	String	Module name **"BRCIOxx"**	
32	Byte	0-7	Main group	Priority group alarm message	1
33	Byte	0-7	Main group	Priority group digital inputs	2
34	Byte	0-7	Main group	Priority group digital outputs	3
35	Byte	0-7	Main group	Priority group analog inputs	4
36	Byte	0-7	Main group	Priority group analog outputs	5
37	Byte	0-7	Main group	Priority group command write	7
38	Byte	0-7	Main group	Priority group command read	7
39	Byte	0-7	Main group	Priority group SDO	6
40	Long		Identifier	Identifier for SYNC object	
41	Long		Identifier	Identifier for alarm object	
42	Long		Identifier	Identifier for service data object master	
43	Long		Identifier	Identifier for service data object slave	
100				Identifier directory	
	Word			Checksum	

2.2 DESCRIPTION OF THE OPERATING PARAMETERS

You will find a description of all the operating parameters in this section.

2.2.1 Version Number

The version number cannot be changed. It can only be read.

2.2.2 Node Number

The node number is diverted from both the dials to the CAN bus controller (see chapter: Hardware). Using the dials, the node number can be set between 0 and 63:

Node Number	Explanation
0	In this case, a configuration memory must be used, in which a node number other than 0 is stored. Also, the node configuration (number, type and slot) must tally with the one in the configuration memory.
1 to 32	(a) If there is no configuration memory available, the CAN bus slave module uses the default values. (b) If there is a configuration memory available, the node configuration (node number, number, type and module slot) must tally with the one in the configuration memory.
33 to 63	In this case, a configuration memory MUST be used, in which a node number other than 0 is stored. Also, the node configuration (node number, number, type and module slot) must tally with the one in the configuration memory.

If the values in the configuration memory area do not correspond with the node configuration, only the objects *command request, command response* and *alarm message* will be activated. With this node status, outputs cannot be set and inputs cannot be read. However, it is possible to send alarms, and to transfer the correct configuration to the configuration memory using the commands.

If the node numbers 0 or 33 - 63 are set on the dial and there is no configuration memory on the CAN bus controller or valid data is not present in the internal EX270 configuration memory, then the node behaves passively on the network. All outputs are switched to log. 0.

2.2.3 Baudrate

The baudrates listed in the table can be set for the B&R SYSTEM 2003. You have to decide if the settings are to be taken from the configuration memory or node number switch.

If the baudrate setting is made using configuration memory, the number shown in the column **No.** must be entered in the operating parameters for the desired baudrate.

Making this setting using the node number switch is described in Chapter 3 "B&R 2003 Modules" in section "CAN Bus Controller".

No.	Baudrate	Configuration Memory	Node Number Switch
0	1 Mbit/sec [1]	●	
1	500 kBit/sec	●	●
2	250 kBit/sec	●	●
3	125 kBit/sec	●	●
4	100 kBit/sec	●	
5	50 kBit/sec	●	
6	20 kBit/sec	●	●
7	10 kBit/sec	●	

[1] 1 MBit/sec cannot be guaranteed electrically because of isolation properties.

2.2.4 Synchronization Jump Width

As standard setting, the values corresponding to the CAL standard are used. In special cases, the synchronization jump width can also be set using the configuration memory.

Oscillator tolerances which are too large can cause transmission disturbances. However, a large ("friendly") setting for synchronization jump width reduces the max. bus length possible.

You can select from four different baudrate tables to compensate for deviations of the individual bits on the bus caused by oscillator tolerances. These four tables correspond to the tables for the *COMPACT2 PLC*. Setting these parameters also adjusts the synchronization jump width for this B&R PLC.

2.2.5 Guard Time

The CAN bus controller can be operated either with or without Life Guarding. It depends on the first life guard signal from the NMT master. If the life guarding signal is once received, life guard time monitoring is activated. This parameter consists of two bytes, from which guard time is calculated:

Byte 1	Byte 0	Guard Time Calculation
0 to 126	64	Byte 1 x Byte 2
16	64	Default Setting: 16 x 64 => 1024 ms

2.2.6 Life Time Factor

Life Guard Time is calculated using the *Life Time Factor*:

Life Guard Time = Life Time Factor x Guard Time

Default setting: 1

2.2.7 Packing the Data

The data for I/O modules can be transferred either packed or unpacked.
The CAN objects for the modules are included with the respective module description in Chapter 3 "B&R 2003 Modules".

Default setting: Packed data transmission

Packed Digital I/O

In this mode, all of the I/O states for **all** modules of a certain type are transferred together. 8 Byte CAN objects are always used. Modules which are not inserted or are of a different type get the value 0.
For mixed modules with channels which can be defined as inputs or outputs (DO435), the bits that correspond to the outputs are set to 0 when transmitting input information and the bits that correspond to the inputs are set to 0 when transmitting output information!
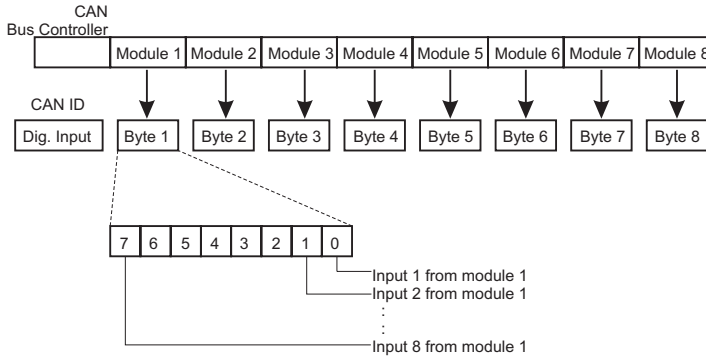
Unpacked Digital I/O

In unpacked mode, a node can be operated with a maximum of four digital I/O modules.

A 1 byte CAN object is used for each digital module. For mixed modules, the outputs bits are read as 0 in the input information. In the output information, the input bits are to be set to 0!
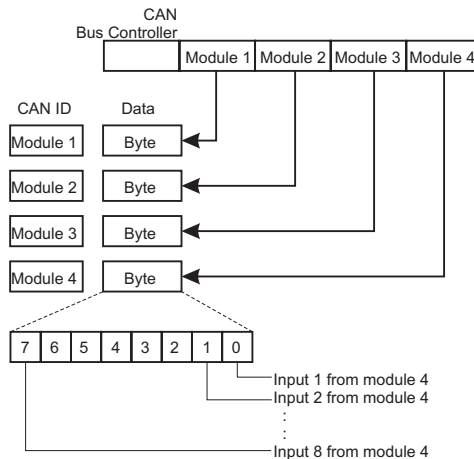
Packed Analog I/O

Four analog screw-in modules can be inserted per adapter module. The I/O values for these screw-in modules are packed in an 8 Byte object. Screw-in modules which are not inserted or modules which are of a different basic type (input/output) get the value 0.

The setting "pack analog modules" is possible in the configuration memory. But the setting to determine if the data is actually packed comes from the configuration of the screw-in modules. The CAN bus controller determines the configuration for each adapter module and makes the decision if the screw-in module data can be packed.

The criteria is that only single channel screw-in modules are allowed to be installed. Therefore, the data per screw-in module corresponds to 1 word. All data can be transferred with a CAN object because a CAN object is 4 words long.

Word 1 of the CAN object corresponds to screw-in module 1, word 4 corresponds to screw-in module 4. The decision if data can be packed or not is made separately for each adapter module.
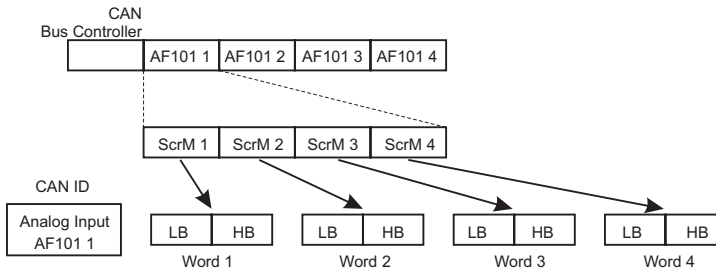
⚠ **AF101 adapter modules are only allowed to be operated with the first four module addresses on the SYSTEMS B&R 2003 (see Chapter 3 "B&R 2003 Modules").**

Unpacked Analog I/O

A 4 word long CAN object is used for each screw-in module. Each CAN object has a different CAN ID. If four two-channel modules are installed on an AF101 adapter module, the data is written in the first two words. Words 3 and 4 are not used.
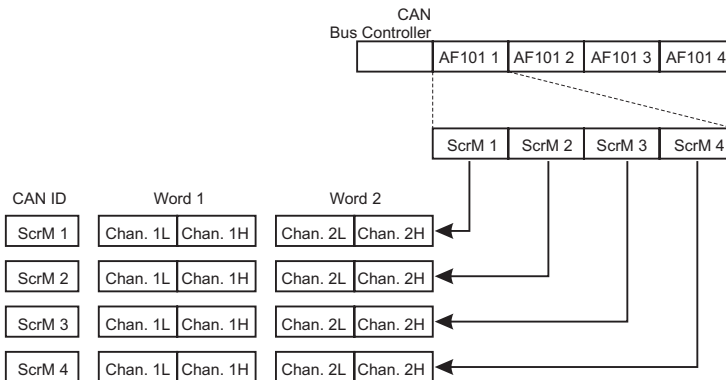
2.2.8 Inhibit Time for an Alarm Object

This parameter determines the minimum time that is to pass after sending the *alarm object* before the alarm object is resent.

Default setting: 5 ms

2.2.9 Behavior of Outputs during Operation

This parameter has a fixed setting of 254. This type of operation is called *asynchronous cyclic*. All other values are reserved for future developments.

The outputs must be transferred cyclically during operation before the idle time passes. Otherwise, the output is reset and an error message is given.

2.2.10 Idle Time for Outputs

This parameter determines the idle time for the outputs. All outputs on the station must be transferred at least once within this time, otherwise the output is reset and an error message is given. A different idle time can be set for digital and analog modules.

Default setting: 640 ms

2.2.11 Output Mask for Digital Outputs

For digital mixed modules with output response (e.g. DO435), the channels must be set as either inputs or outputs. As default, this value is 0 (=> all channels are configured as inputs).

  Pay attention to the output data!

All channels which are configured as inputs must be set to 0 in the output data, otherwise there is a conflict with the output mask. That means the respective output must be set to 0 before a channel is used as an input. However, if an output is set which is actually defined and is being used as an input, an error message is given and the idle time is not reset. This results in all outputs being set to 0 if the data word (output data) is not corrected in time.

Channels configured as outputs are read from the input data as 0.

Default setting: Output mask = 0
                        (=> all channels are defined as inputs)

2.2.12 Behavior of the Inputs during Operation

This parameter determines the basic behavior of the inputs. Different behavior can be defined for digital and analog modules. Select:

Value	Description
254	Asynchronous – Cyclic (a) These inputs states are sent cyclically (=> cycle time of the inputs). (b) Sending the input states can be triggered by a defined event (command Read Inputs).
255	Asynchronous - Changes and when Idle Time Passes (a) The inputs are sent when they change. The CAN bus controller holds to a minimum time between objects with the same ID (=> Inhibit Time for Inputs). The idle time is not reset. (b) If a change does not occur within a certain time (=> idle time for inputs), the input states are sent automatically. (c) Sending the input states can be triggered by a defined event (command Read Inputs).

Default setting: 255 - Asynchronous
                        Changes and when Idle Time Passes

2.2.13 Change Mask "Digital Inputs"

The standard setting ensures that the input states are sent automatically if a change occurs. With the "Digital Inputs" change mask, sending when changes occur can be suppressed for certain inputs. The respective bit is set to 0.

Default setting: $FF => Edge recognition is turned on for all inputs.

For digital mixed modules, all channels declared as outputs are automatically masked out. This guarantees that sending is not triggered when these outputs change.

2.2.14 Cycle Time for Inputs

This time determines the cycle time in which the input states are sent. A different cycle time can be entered for digital and analog modules.

Default setting: 100 ms

2.2.15 Idle Time for Inputs

If the input states have not changed when the idle time passes, the input states are sent. A different idle time can be set for digital and analog modules.

Default setting: 640 ms

2.2.16 Inhibit Time for Inputs

A minimum time must pass between sending procedures on the network for objects with the same ID.

Default setting: Digital Inputs: 5 ms
Analog Inputs: 20 ms

Analog Inputs

If a DBT master is available, only times which are larger than those determined by the master are accepted. If a DBT master is not available, times smaller than the default value of 20 ms or the value in configuration memory can be set.

2.2.17 Transmission Trigger Type for Analog Inputs

The trigger type determines the criteria for triggering transmission. Presently, the hysteresis is implemented as default. The hysteresis is defined using the operating parameter Value Corresponding to Trigger Type. If the input value changes by the defined amount, transmission is triggered.

Default setting: 4

2.2.18 Value Corresponding to Trigger Type for Analog Inputs

The input value has to change by this amount in order to start transmission. If 4 is set for trigger type, this value corresponds to a positive and negative hysteresis value.

Default setting: 160

2.2.19 Parameter Value for Screw-in Modules

Setting parameters for screw-in modules is included in Chapter 3 "B&R 2003 Modules" with the respective module description (configuration word 14, write).
Chapter 4 "Module Addressing" contains an example program.

2.2.20 Module Code Digital I/O Modules

During the initialization phase, a code is saved in this location for each digital I/O module.
The B&R ID Code (Module Code + $E0) is listed in the module overview and the respective module description in Chapter 3 "B&R 2003 Modules".

		Bit	Description
		5 - 7	0 ....Status bits are masked out
		0 - 4	Module code

0 0 0 | | | | | |
7   0

2.2.21 Module Code for Screw-in Modules

The module code can be read from configuration word 14. Module code and B&R ID code are identical for the screw-in modules.
The B&R ID Code is listed in the module overview and the respective module description in Chapter 3 "B&R 2003 Modules".

2.2.22 Module Name "BRCIOxx"

The CAL module name in connection with the module ID (= node number) represents a distinct description for a module in a CAL network. This name is fixed and cannot be changed. The name consists of the letters **BRCIO** and a two digit module ID. The CAN bus controller with station number 9 has the CAL module name **BRCIO09**.

2.2.23 Priority Groups

The priority group indicates to the DBT master how the priorities of CAN objects are to be handled for identifier allocations. CAN objects with higher priority are assigned low IDs; objects with lower priority get higher IDs.

2.2.24 Identifier Directory

The identifiers for the inputs and outputs are stored in this directory. All of the IDs for the individual objects are entered here.

	Digital		Analog Input				Analog Output			
Module	Input	Output	Slot 1	Slot 2	Slot 3	Slot 4	Slot 1	Slot 2	Slot 3	Slot 4
1	ID	ID	...	...				...	....	.
2	ID									
3	:									
4	:									
5										:
6										:
7										ID
8							...	...	ID	ID

The table shows the basic structure of the identifier directory. This directory can be accessed e.g. using the commands *Read Operating Parameter* or *Write Operating Parameter*. How the individual IDs are accessed is described in section "Commands - CAN Bus Controller".

3 COMMANDS - CAN-BUS CONTROLLER

An identifier can send a CAN bus controller a command (object ID *command request*) from another station (client; e.g.: CP200). A response to this command can be read again by another identifier (object ID *command response*).

⚠ **Command requests from only ONE sender for a node!**

Command requests **for one node** are only allowed to be sent **from ONE sender**, otherwise there could be conflicts on the bus.
Different nodes can get their commands different senders.

3.1 GENERAL STRUCTURE OF THE COMMANDS

Object *command requests* and *command responses* both have a similar structure. Both objects consist of 8 bytes:

3.1.1 Command Request

Byte	Description
0	Code (number in range from 0 ... 126)
1	Command code
2	Parameter 1 (p1)
3	Parameter 2 (p2)
4 - 7	Command data in Motorola format

Not every *command request* to the bus controller automatically causes a command response (see description of individual commands).

3.1.2 Command Response

Byte	Description
0	Transmission OK: Code = Client code Transmission faulty: Bit 7 of the code is set Transmission of a message which was not requested: Code = $FF, e.g. Error message fro the CAN slave module
1	Command code
2	Parameter 1 (p1)
3	Parameter 2 (p2)
4 - 7	Command response in Motorola format or error code (see Appendix B "Error Messages CAN Bus Slave Module")

> ⚠ **The response contains the client's code so that the client which sent the command receives the response. If an error occurs while the command is being processed, bit 7 of the code is set. Therefore, the client recognizes that the last command was acknowledged with an error message.**

If the command response contains the code $FF, then an automatic message (i.e.: error message) is given by the bus controller.

An **echo** can also be a command response. The echo is formed from the command request which was originally sent.

> ⚠ **The data format of the command data and the command response correspond to Motorola format:**

HH	MH	ML	LL

Note: Code
Valid values for a command response or command code are between 0 and 126. You can use this code in your project if you like (e.g. to route the command requests and command responses).

Note: Command Response
In order to read e.g. the operating parameters, a station (client) must first send a command request to a bus controller and then wait for a response from that bus controller. The desired data will be contained in this response.

3.2 COMMAND CODES AND PARAMETERS

You will find a description of all special commands and their parameters in this section.
The abbreviation in the 2nd column "ScrM(p2)" stands for screw-in module.

3.2.1 Read Slave or Module Status

Command Code: 00				
Module (p1)	ScrM (p2)	Command Data	Command Response	Description
0	0	----	int32	Read the last slave error that occurred
1 - 8	0	----	int32	Read the last module error that occurred
1 - 8	1 - 4	----	int32	Read the last screw-in module error that occurred

Response received by the client:
  0 .... No error
  Error .... contains error and additional code
     see Appendix B "CAN Bus Controller Error Messages"

3.2.2 Read Number of Modules or Module Code

Command Code: 01 – Digital Modules				
Module (p1)	ScrM (p2)	Command Data	Command Response	Description
0	0	----	Number of bytes	Number of analog modules and digital I/O modules on the slave

1 - 8	0	----	Number of bytes
			Byte status
			Byte version
			Byte power
				Byte 4 ....Number of channels on a digital module. 8 is returned as command response.
				Byte 5 ....Module status
				 Bit 0..........0: No error, the supply for the digital output is OK
				       1: Short circuit, over-temperature or the supply voltage for the digital outputs is not OK
				 Bit 1..........0: Module voltage OK
				       1: No supply voltage or supply voltage too low for digital inputs/outputs
				Byte 6 ....Version code – only for modules with extended status
				 Bit 0 - 5.....Version
				 Bit 6 - 7.....Module width
				Byte 7 ....Power value for the digital module
				     For logical double width modules, the power value is divided between two slots.

1 - 8	≠0	----	Byte MK
			Byte status
			Byte version
			Byte power
				Byte 4 ....Module code (see Chapter 3 "B&R 2003 Modules", section " B&R 2003 Module Overview" or the respective module description)
				Byte 5 ....Module status
				 Bit 0..........0: No error, the supply for the digital output is OK
				       1: Short circuit, over-temperature or the supply voltage for the digital outputs is not OK
				 Bit 1..........0: Module voltage OK
				       1: No supply voltage or supply voltage too low for digital inputs/outputs
				Byte 6 ....Version code – only for modules with extended status
				 Bit 0 - 5.....Version
				 Bit 6 - 7.....Module width
				Byte 7 ....Power value for the digital module
				     For logical double width modules, the power value is divided between two slots.

Command Code: 01 – Screw-in Modules

Module (p1)	ScrM (p2)	Command Data	Command Response	Description
1 - 4	0	----	Number of bytes Byte reserved Byte reserved Byte power	Byte 4.... Number of screw-in modules on an analog module. 0 - 4 will be returned as command response. Byte 5.... reserved Byte 6.... reserved Byte 7.... Power value for the analog module
1 - 4	1 - 4	----	Byte AF code HB ScrM code LB ScrM code Byte power	Module code (see Chapter 3 "B&R 2003 Modules", section " B&R 2003 Module Overview" or the respective module description) Command Response: Byte 4.... $Cx code for the analog module Byte 5.... Code for the screw-in module HB Byte 6.... Code for the screw-in module LB Byte 7.... Power value for the screw-in module

The power consumption of the module can be calculated using the power value in byte 7 of the command response. For digital modules which are not entered in the module list of the operating system, the power value is set to 0.

Power consumption = power value * 0.1 W

3.2.3 Reading Output

Command Code: 02

Module (p1)	ScrM (p2)	Command Data	Command Response	Description
0	0	----	Byte m1 Byte m2 Byte m3 Byte m4	Read output level of modules 1 - 4 0...........no output modules or outputs are logical 0 1...........output is logical 1
0	1	----	Byte m5 Byte m6 Byte m7 Byte m8	Read output level of modules 5 - 8 0...........no output modules or outputs are logical 0 1.......... output is logical 1
1 - 8	0	----	Byte mx or error	Read output level of module x 0...........output is logical 0 1.......... output is logical 1 If the module is not a digital output module, an error message is given

Module Order in Command Response

Byte 4	Byte 5	Byte 6	Byte 7
m1	m2	m3	m4

m5	m6	m7	m8
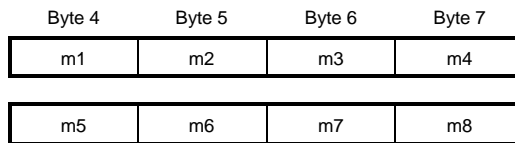
The individual bits of *Byte mx* each represent an output (input):

	Bit	Description
	7	Output 8 on module x
	6	Output 7 on module x
	5	Output 6 on module x
	4	Output 5 on module x
	3	Output 4 on module x
	2	Output 3 on module x
	1	Output 2 on module x
	0	Output 1 on module x

7   0
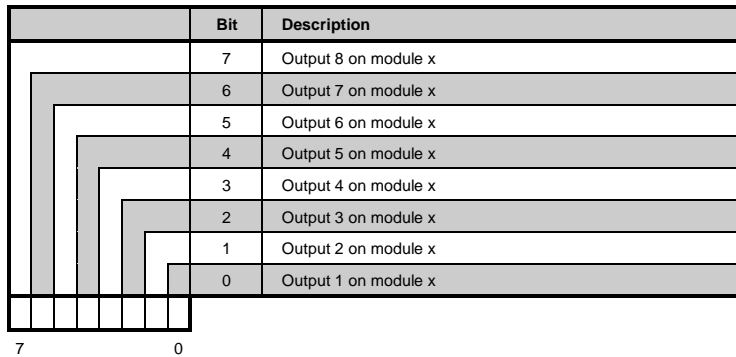
Digital Mixed Modules

The current input state is sent for digital mixed modules with output feedback.

3.2.4 Read Input

Command Code: 03				
Module (p1)	ScrM (p2)	Command Data	Command Response	Description
1 - 8	0	----	No response or error message	Request the slave module to send to all inputs on the given digital module. If the module is not an input module, an error message is given.
1 - 4	Low Nibble: Scr.Mod. no 1-4 High Nibble: Data word 0 - 3 or configuration word 4 - 12	----	Data word: no response Configuration word: command response or error message	Requests the slave module to send the input data object of the given analog screw-in module. f the module is not an input module, an error message is given.

The parameter ScrM (p2) is divided in the High Nibble and Low Nibble:

	Bit	Description
	4 - 7	0 - 3 ..... Data word 4 - 12 ... Configuration word
	0 - 3	1 - 4 ..... Screw-in module number

7   0
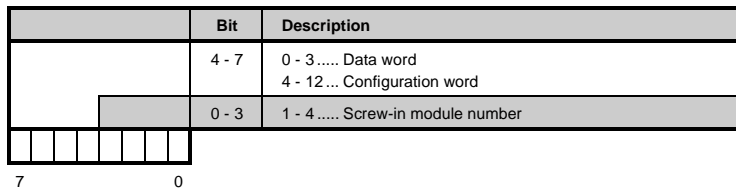
The data and configuration words for the screw-in modules are included in Chapter 3 "B&R 2003 Modules" with the respective module description.
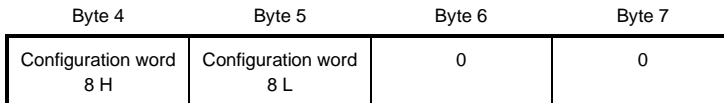The send procedure for the data object is triggered when a data word is read.

If a configuration word is read, the data is sent as command response. Using the command *Read Input*, configuration words 4 - 15 can be read. The module code ins stored in configuration word 14. The module code is read using command 1 *Read Number of Modules or Module Code* or command 10 *Read Operating Parameters* (Parameter No. = 30).

Data format of the command response (read a configuration word):

Configuration words 4+5 and 6+7: 32 bit data

Byte 4	Byte 5	Byte 6	Byte 7
Configuration word 4 H	Configuration word 4 L	Configuration word 5 H	Configuration word 5 L

Configuration words 8 -15: 16 bit data

Byte 4	Byte 5	Byte 6	Byte 7
Configuration word 8 H	Configuration word 8 L	0	0

3.2.5 Set Output Masks for a Module

Command Code: 04

Module (p1)	ScrM (p2)	Command Data	Command Response	Description
0	0	Byte m	Echo	All output masks on a slaves 0 ....set to input 255 ......set to output
1 - 8	0	Byte m	Echo or error	Set the output mask for each bit to input or output on the given module %00000000 .... all to input %11111111 .... all to output An error message is created if the module number > 8

This command is required e.g for the digital output module DO435.

The mask is accepted immediately. The command *Activate Operating Parameters* is not necessary.

The masks is automatically set to %11111111 for digital outputs alone. With mixed modules (individual digital channels can be defined as either inputs or outputs), the send mask for the inputs is created from this output mask (inverse output mask).

3.2.6 Set Change Mask for a Module

Command Code: 05				
Module (p1)	**ScrM** (p2)	**Command Data**	**Command Response**	**Description**
0	0	Byte m	Echo	All change masks on a slave 0 .... no edge 255 ...... edge (standard)
1 - 8	0	Byte m	Echo or error	Set change mask to edge or no edge for each bit on the given module. %00000000 set all to no edge %11111111 set all to edge An error message is created if the module number > 8

The mask is accepted immediately. The command *Activate Operating Parameters* is not necessary.

For mixed modules, this change mask is linked with the inverse output mask (AND) so that changes to the outputs do not trigger a send procedure (reduce load on the network).

3.2.7 Set Trigger Type for Analog Input Channel

Command Code: 06				
Module (p1)	**ScrM** (p2)	**Command Data**	**Command Response**	**Description**
0	0	Byte type (4)	Echo	Set all entries to the given data
1 - 4	1 - 4	Byte type (4)	Echo or error	Set parameters for the given screw-in module An error message is given if the module number > 4 or ScrM > 4

The parameter is accepted immediately. Presently, the only type of triggering available is hysteresis (command data = 4).

3.2.8 Set Change Value Corresponding to Trigger Type

Command Code: 07				
Module (p1)	**ScrM** (p2)	**Command Data**	**Command Response**	**Description**
0	0	int32 value	Echo	Set all entries to the given data
1 – 4	1 - 4	int32 value	Echo or error	Set parameters for the given screw-in module An error message is given if the module number > 4 or ScrM > 4

The parameter is accepted immediately.

The command data must be given in Motorola format:

HH	MH	ML	LL

3.2.9 Write Value to Configuration Word

Command Code: 08				
Module (p1)	ScrM (p2)	Command Data	Command Response	Description
1 - 4	Low Nibble: ScrM no 1-4 High Nibble: Configuration word 4 - 12	2 or 4 Byte	Echo or error	Write command data in configuration word for the given screw-in module An error message is created if the addressing is not correct.

The parameter ScrM (p2) is divided in the High Nibble and Low Nibble:
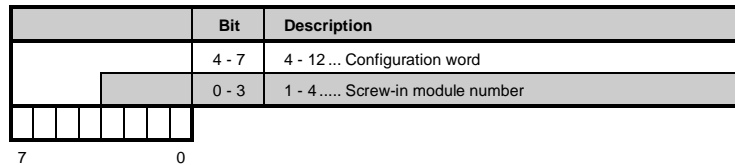
		Bit	Description
		4 - 7	4 - 12 ... Configuration word
		0 - 3	1 - 4 ..... Screw-in module number

```
7                    0
```

The configuration words for the screw-in modules are included in Chapter 3 "B&R 2003 Modules" with the respective module description.
Configuration words 4 - 12 can be written to Using the command *Write Value to Configuration Word*.
Configuration word 14 is written to using command 11 *Write Operating Parameters* (Parameter No. = 28).

Data format for command data:

Configuration words 4+5 and 6+7: 32 bit data

Byte 4	Byte 5	Byte 6	Byte 7
Configuration word 4 H	Configuration word 4 L	Configuration word 5 H	Configuration word 5 L
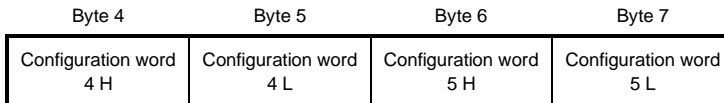
Configuration words 8 - 12: 16 bit data

Byte 4	Byte 5	Byte 6	Byte 7
Configuration word 8 H	Configuration word 8 L	0	0

3.2.10 Read Operating Parameters

Command Code: 10

No. (p1)	Block (p2)	Command Data	Command Response	Description
Number of the operating parameter	Block no. In the entry 0 to n	----	Dependent on the operating parameter or error	Read the operating parameter according to entry number. For entries larger than 4 bytes, data with block number x 4 as offset (exception: see Identifier Directory). An error message is given if the entry or block number is invalid.

Section "Operating Parameters" contains an overview. Each operating parameter is assigned a number in the column "No.". The number of the operating parameter to be read is entered in Parameter No. (p1).

Operating parameter entries which are longer than 4 bytes must be requested with more than one command since only 4 data bytes can be transferred per command. Parameter Block (p2) entries for block number are used for the offset calculation. With the help of this offset, the operating parameter data is requested in blocks of 4.

Offset = block number * 4

The use of the *Read operating Parameter* command will be explained using two examples.

Example 1

Read operating parameter no. 29 *Module Code for Digital I/O Modules*:

Command Code: 10

No. (p1)	Block (p2)	Command Data	Command Response
29	0	----	Byte 4....... Code for module 1 Byte 5....... Code for module 2 Byte 6....... Code for module 3 Byte 7....... Code for module 4
29	1	----	Byte 4....... Code for module 5 Byte 5....... Code for module 6 Byte 6....... Code for module 7 Byte 7....... Code for module 8

Example 2

Read operating parameter no. 30 *Module Code for Screw-in Modules*.
The assignment between block number and command data for the screw-in modules can be seen in the following drawing:

Block Number	0	1	2	3	4	5	6	7
Command Data								
Byte 4	ScrM 1H	ScrM 3H	ScrM 1H	ScrM 3H	ScrM 1H	ScrM 3H	ScrM 1H	ScrM 3H
Byte 5	ScrM 1L	ScrM 3L	ScrM 1L	ScrM 3L	ScrM 1L	ScrM 3L	ScrM 1L	ScrM 3L
Byte 6	ScrM 2H	ScrM 4H	ScrM 2H	ScrM 4H	ScrM 2H	ScrM 4H	ScrM 2H	ScrM 4H
Byte 7	ScrM 2L	ScrM 4L	ScrM 2L	ScrM 4L	ScrM 2L	ScrM 4L	ScrM 2L	ScrM 4L

The module codes for the screw-in modules on the first AF101 module and screw-in modules 3 and 4 on the third AF101 module are read.

Command Code: 10

No. (p1)	Block (p2)	Command Data	Command Response
30	0	----	Byte 4 .... Code for module 1, screw-in module 1H Byte 5 .... Code for module 1, screw-in module 1L Byte 6 .... Code for module 1, screw-in module 2H Byte 7 .... Code for module 1, screw-in module 2L
30	1	----	Byte 4 .... Code for module 1, screw-in module 3H Byte 5 .... Code for module 1, screw-in module 3L Byte 6 .... Code for module 1, screw-in module 4H Byte 7 .... Code for module 1, screw-in module 4L
30	5	----	Byte 4 .... Code for module 3, screw-in module 3H Byte 5 .... Code for module 3, screw-in module 3L Byte 6 .... Code for module 3, screw-in module 4H Byte 7 .... Code for module 3, screw-in module 4L

Identifier Directory

Take note of the following when reading the identifier directory:
You can read an identifier from the directory with each *Read Operating Parameter* command.

The following table shows a list of the No./Block No. to be used:

Module	Digital		Analog Input				Analog Output			
	Input	Output	Slot 1	Slot 2	Slot 3	Slot 4	Slot 1	Slot 2	Slot 3	Slot 4
1	101 / 1	101 / 2	101 / 3	101 / 4	101 / 5	101 / 6	101 / 7	101 / 8	101 / 9	101 / 10
2	102 / 1	102 / 2	102 / 3	102 / 4	102 / 5	102 / 6	102 / 7	102 / 8	102 / 9	102 / 10
3	103 / 1	103 / 2	103 / 3	103 / 4	103 / 5	103 / 6	103 / 7	103 / 8	103 / 9	103 / 10
4	104 / 1	104 / 2	104 / 3	104 / 4	104 / 5	104 / 6	104 / 7	104 / 8	104 / 9	104 / 10
5	105 / 1	105 / 2	105 / 3	105 / 4	105 / 5	105 / 6	105 / 7	105 / 8	105 / 9	105 / 10
6	106 / 1	106 / 2	106 / 3	106 / 4	106 / 5	106 / 6	106 / 7	106 / 8	106 / 9	106 / 10
7	107 / 1	107 / 2	107 / 3	107 / 4	107 / 5	107 / 6	107 / 7	107 / 8	107 / 9	107 / 10
8	108 / 1	108 / 2	108 / 3	108 / 4	108 / 5	108 / 6	108 / 7	108 / 8	108 / 9	108 / 10

see "Identifier Directory", section "Operating Parameter"

In order to e.g. read the identifier for module 3 / analog output - slot 3, the following command request must be sent:

Command Code: 10

No. (p1)	Block (p2)	Command Data	Description
103	5	----	

3.2.11 Write Operating Parameters

Command Code: 11

No. (p1)	Block (p2)	Command Data	Command Response	Description
Number of the operating parameter	Block no. in the entry 0 to n	Byte x3 Byte x2 Byte x1 Byte x0	Echo or error	Write the operating parameter in a buffer. For entries larger than 4 bytes, data with block number x 4 as offset. An error message is given if the entry or block number is invalid.

⚠ **The data is written to the buffer and therefore does not influence the program. They have to be copied to the active parameter area with the *Activate Operating Parameters* command.**

The parameters in the command request are (if possible) checked for plausibility. If an error is found, the command response contains the respective error message.
Take note of the explanations concerning the block number which are given with the description of the *Read Operating Parameters* command.

3.2.12 Activate Operating Parameters

Command Code: 12

Parameter 1	Parameter 2	Command Data	Command Response	Description
0	0	----	Echo	Copy the operating parameters which are in the buffer to the area for active parameters (see section "Operating Parameters - Overview").

The parameters found in the buffer (see command *write operating parameters*) are checked as thoroughly as possible for plausibility. If an error is found, the command response contains the respective error message. The command *Activate Operating Parameters* does not have to be carried out after every *Write Operating Parameters* command. It is enough if it is carried out once after the last write command. In this way, all operating parameters are accepted at once which saves time.

3.2.13 Transfer Operating Parameters to Configuration Memory (EX270 internal)

Command Code: 13

Parameter 1	Parameter 2	Command Data	Command Response	Description
0	0		Echo	Copy the active operating parameters to configuration memory.
			or error	If an error occurs, an error message is generated.

Writing to the configuration memory is done in the background because of the time requirements of approx. 10 ms per word. Transferring the operating parameters to configuration memory is described in the following steps:

- The contents are overwritten with $FFFF before the write procedure using the built-in delete function (in configuration memory).
- Transfer operating parameters as word
- Calculate the checksum placed in configuration memory.
- Verify the transferred parameters

During this procedure, the operating parameters cannot be changed. Commands such as *Set Output Mask for Module* are disabled and return a corresponding error message.

3.2.14 Delete Configuration Memory (EX270 internal)

Command Code: 14				
Parameter 1	Parameter 2	Command Data	Command Response	Description
0	0	----	Echo or error	Delete the configuration memory (overwrite EEPROM with $FFFF). If an error occurs, an error message is generated.

3.2.15 Test if Configuration Memory is Available

Command Code: 15				
Parameter 1	Parameter 2	Command Data	Command Response	Description
0	0	----	Byte	0 .... no configuration memory 1 .... configuration memory available

3.2.16 Read Operating System Version

Command Code: 16				
Parameter 1	Parameter 2	Command Data	Command Response	Description
0	0	----	Byte SW version Byte OP version HB checksum LB checksum	Command Response: Byte 4.... SW version of the controller. The version is BCD coded: V 2.7 = $27 Byte 5.... Version of the operating parameter memory data structure (see section "Operating Parameter", Byte 0) Byte 6.... Checksum for the operating parameter memory HB Byte 7.... Checksum for the operating parameter memory LB

3.2.17 Restart Slave

Command Code: 20				
Parameter 1	Parameter 2	Command Data	Command Response	Description
0	0	----	----	The slave is removed from the network and restarted (see Initialization Phase). If new parameters were transferred to configuration memory, they become valid.

3.2.18 Set Node Number

Command Code: 21

Parameter 1	Parameter 2	Command Data	Command Response	Description
0	0	Byte 4 = node number 0 - 63	Echo	Setting the node number for following warm start of the slaves (command code 20). The node number remains set until the next power-on.
			or error	An error message is generated if an error occurs.

4 START BEHAVIOR

The B&R System 2003 supports class 0, 1 and 2 in CiA/CAL networks. A CAN bus controller recognizes the network it is in automatically and adjusts itself appropriately. After a power-on or a reset, the bus controller starts according to a certain algorithm which is described in this section.

4.1 INITIALIZATION AND NETWORK CLASS RECOGNITION

The diagram shown is a simplified illustration of the internal operation of a CAN bus controller after power-on or a reset.

NMT Master Available
(Network Class 1 or 2)

Normal Operation
Operating
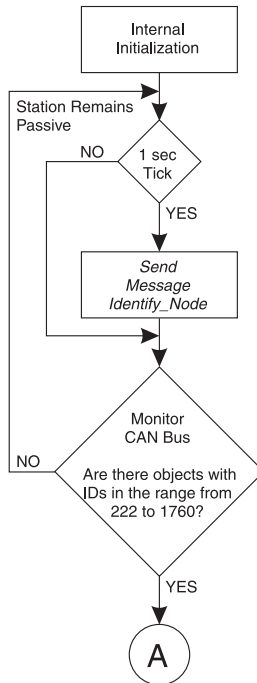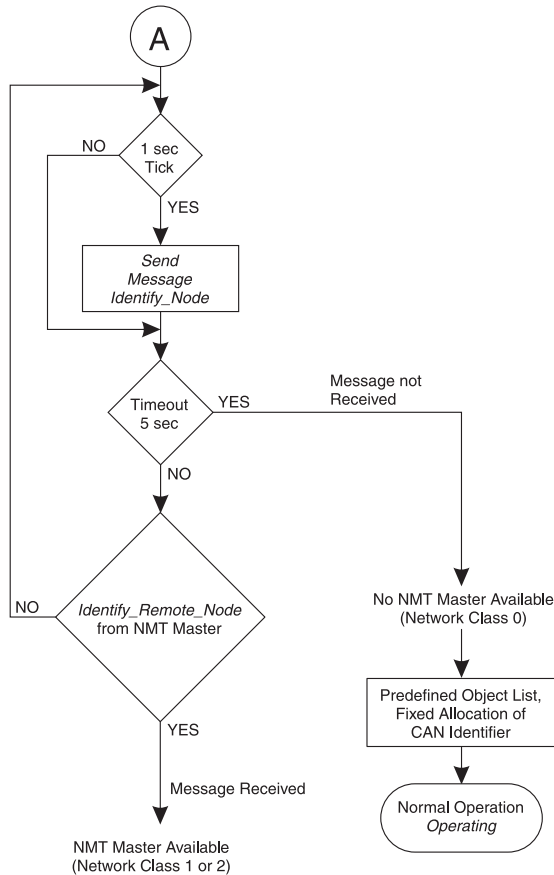
Description of the Diagram

1. Internal initialization of the interface components and the internal structures.

2. The CAN bus controller tries to send the message *Identify_Node* (ID = 2022) once every second to a NMT master, which may be available.

3. The CAN bus controller observes the CAN bus to determine if data is being transferred. Only those objects with an ID in the range from 222 to 1760 are noted (corresponds to valid ID for a bus controller). If such an object is found, an available NMT master must respond to the message *Identify_Node* within 5 seconds by sending *Identify_Remote_Node*. If a response is not given, a class 0 network is required.

Note
If only B&R devices are used in a CAN network, class 0 is always used.
All CAN bus controller also support classes 1 and 2, and can therefore also be used in a network with CAN bus capable devices from other manufacturers.

4.2 WITHOUT NMT MASTER (CLASS 0)

If no NMT master responds during the bus controller's start up phase, then there is a class 0 CiA/CAL network. In this situation, the CAN bus controller is initialized as follows:

1. to 3.   see section "Initialization and Recognition of the Network Class"

4.   Pre-defined identifiers are assigned to all station objects by the CAN bus controller (see Appendix "CAN Identifier - Fixed Allocation").

5.   The CAN bus controller starts operating normally (status: *Operating*) and now behaves on the network in accordance with the set operating parameters (e.g. cyclically sending digital inputs etc).

The CAN bus controller is now ready for communication. Other CAN stations (clients) can now access the node objects using the identifier. Function blocks are available for this purpose for B&R devices.

4.3 WITH NMT MASTER (CLASS 1 OR 2)

The following steps are carried out if the bus controller recognizes that an NMT master is available during the initial start-up phase:

Bus Controller

NMT Master Available
(Network Class 1 or 2)

```
┌─────────────────────┐
│        Send         │ ──────────→  NMT Master
│      Message        │
│   Identify_Node     │
└─────────────────────┘
┌─────────────────────┐
│ Evaluate: Part 1 of │ ←──────────
│  Connect_Remote_Node│
└─────────────────────┘
┌─────────────────────┐
│        Send         │ ──────────→
│   Response Part 1   │
└─────────────────────┘
┌─────────────────────┐
│ Evaluate: Part 2 of │ ←──────────
│  Connect_Remote_Node│
└─────────────────────┘
┌─────────────────────┐
│        Send         │ ──────────→
│ Error Status, Node ID,│
│     Error Code      │
└─────────────────────┘
```

Only Class 2:
Activate NMT Master
Life Guarding and carries
it out cyclically

```
┌─────────────────────┐
│      Evaluate       │ ←──────────
│ Prepare_Remote_Node │
└─────────────────────┘
┌─────────────────────┐
│ Slave attempts to   │ *) See
│ communicate with    │ next graphic  ←────→ DBT Master
│    DBT Master       │
└─────────────────────┘
┌─────────────────────┐
│        Send         │ ──────────→
│    OK_Respnose      │
└─────────────────────┘
┌─────────────────────┐
│     Wait for        │ ←──────────
│  Start_Remote_Node  │
└─────────────────────┘
   ╱─────────────────╲
  │ Normal Operation  │
  │    Operating      │
   ╲─────────────────╱
```

Network Class	1	2
Error Monitoring	Yes	Yes
Reaction to new nodes	Yes	Yes
Life Guarding	No	Yes

These functions are carried out by the NMT Master.

The above diagram is a simplified version of the initialization phase and the communication between the bus controller and the NMT/DBT master.

Description of the Diagram

1. to 3.   see section "Initialization and Recognition of the Network Class"

4. The bus controller sends the message *Identify_Node* again after it has received the message *Identify_Remote_Node*.

5. The CAN bus controller waits for part 1 of the message *Connect_Remote_Node* as response from the NMT master.

6. The bus controller sends the *part 1 response* to the master after receiving the *Connect_Remote_Node* message. The bus controller gives different types of information to the NMT master in this message (e.g.: desired *guard time*, *module ID,* etc).

7. The NMT master evaluates the response to part 1 and then sends part 2 of the *Connect_Remote_Node* message including information from the master (*guard time*, *life time*, *network class*).

8. The slave evaluates part 2 of the message from the master and responds with *Error Status*, *Node ID* (=> node number) and *Error Code*.

9. **Only network class 2:** The NMT master activates Life Guarding, which is then carried out parallel to the following initialization phases.

10. The master signals to the slave that the slave can now start to register objects with the DBT master by sending the message *Prepare_Remote_Node*. This message includes whether the slave should keep the object list, or whether it should construct a new one.

11. Now the slave tries to communicate with the DBT master (see next point "DBT master").

12. After the slave has tried to communicate with the DBT master (regardless of if it was successful or not), it sends an *OK Response* to the NMT master.

13. The slave waits for the response *Start_Remote_Node* from the NMT master.

14. The slave switches to *Operating* status, and operates normally.

15. **(a) Class 1 Network:** From this point on, the master only monitors for errors and reacts to new nodes.

   (b) Class 2 Network: In addition to the functions of class 1, the master also carries out Life Guarding.

The CAN bus controller is now ready. Other CAN stations (clients) can only access these module objects using the identifier. Function blocks are available for this purpose for B&R devices.

4.4 DBT MASTER

If a DBT master is being used, an NMT master is required. A DBT master can assign identifiers to every station object.

The user does not have to use the fixed ID allocation of the CAN bus controller. A new identifier can be assigned to all station objects. The flexibility of free ID allocation (according to the CAL standard) has the following advantages:

- The IDs are managed centrally from one station.
- The network configuration can be changed more easily (centrally). If, for example, a module is moved onto another slot on a station, then all module objects on this module (inputs/outputs) are given another identifier through fixed ID allocation. Without DBT, the program would have to be changed since it accesses the object with an invalid ID. This is not necessary when all IDs are managed by a DBT master. All changes are always made in the configuration for the DBT master.

During the initialization phase (see "With NMT Master Class 1 or 2"), the bus controller tries to communicate with a DBT master, which may be available.
First, the slave receives permission from the NMT master to register its objects with the DBT master (*Prepare_Remote_Node*).

The following diagram shows how communication with the DBT master takes place:

Bus Controller

Slave attempts to communicate with **DBT Master**

Should the Object List be Recreated

YES

NO

Send Message Get_Checksum

Wait for Response from DBT Master

Timeout

Checksum is Invalid

Checksum is Valid

Predefined Object List, Fixed Allocation of CAN Identifier

Send Message Delete_User_Definition for each Object to DBT

Timeout

DBT has Deleted all Objects in its List

Send Message Create_User_Definition for each Object to DBT

Timeout

DBT has Successfully Allocated all Identifiers

Timeout: no response from DBT

5 FIXED ALLOCATION OF THE CAN IDENDIFIER

If a DBT master is not available on the network, the CAN identifier must be allocated using fixed settings. Using the method described in the following section, a priority is achieved for the CAN identifier which is similar that used in the DBT master via priority groups.

The fixed ID allocation takes place dependent on the set transfer mode (packed or unpacked).

The fixed ID allocation only takes effect if one of the following points are true:

- No configuration memory is available.
- The available configuration memory is invalid.
- In configuration memory, none of the modules are defined corresponding to the identifier

Make sure that a maximum of 12 receive objects (output data) are created for the node.

5.1 PACKED MODE

Object	Calculation	nd	ma
Alarm Message	222 + (nd - 1)	1 - 32	----
Digital Inputs	286 + (nd - 1) x 4	1 - 32	----
Digital Outputs	414 + (nd - 1) x 4	1 - 32	----
Analog Inputs	542 + (nd - 1) x 16 + (ma - 1) x 4	1 - 32	1 - 4
Analog Output	1054 + (nd - 1) x 16 + (ma - 1) x 4	1 - 32	1 - 4
Command Request	1566 + (nd - 1)	1 - 63	----
Command Response	1630 + (nd - 1)	1 - 63	----

nd.... Node number of the   ma ...Module address of
        CAN Slave   the AF101

5.2 UNPACKED MODE

Object	Calculation	nd	ma	sl
Alarm Message	222 + (nd - 1)	1 - 32	----	----
Digital Inputs	286 + (nd - 1) x 4 + (ma - 1)	1 - 32	1 - 4	----
Digital Outputs	414 + (nd - 1) x 4 + (ma - 1)	1 - 32	1 - 4	----
Analog Inputs	542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)	1 - 32	1 - 4	1 - 4
Analog Outputs	1054 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1)	1 - 32	1 - 4	1 - 4
Command Request	1566 + (nd - 1)	1 - 63	----	----
Command Response	1630 + (nd - 1)	1 - 63	----	----

nd.... Node number of the CAN Slave   ma ...Module address ofthe digital I/O module or the AF101   sl ..... Slot number of the screw-in module on the AF101

⚠️ **In unpacked mode, a maximum of four digital I/O modules can be operated on a node.**

5.3 PACKED AND UNPACKED MODE

The following requirements must be met for the calculated allocation of the identifier:

- A maximum of 32 nodes can be present in the network
- If this limit is exceeded, either the configuration memory or a DBT master must be used.

In order to make working with the fixed allocation of identifiers easier, a complete list for all identifiers is available in Appendix A "CAN Identifier with CAN Bus Controller (Fixed Allocation)".

5.3.1 Examples

Example 1

Packed Mode

A node consists of ....

- ... CAN bus controller (node number 1)
- ... Adapter module AF101 (module address 1)
  analog input module AI351 (slot 1)
  analog input module AI351 (slot 2)
  analog input module AI351 (slot 3)
  The analog input data is packed, since only single channel modules are available.
- ... Digital input module DI435 (module address 2)
- ... Digital output module DO435 (module address 3)

Object	Calculation	Identifier	
Alarm Message	222 + (nd - 1) 222 + (1 - 1)	222	$00DE
3 x AI351	542 + (nd - 1) x 16 + (ma - 1) x 4 542 + (1 - 1) x 16 + (1 - 1) x 4	542	$021E
DI435	286 + (nd - 1) x 4 286 + (1 - 1) x 4	286	$011E
DO435 - Outputs	414 + (nd - 1) x 4 414 + (1 - 1) x 4	414	$019E
DO435 - inputs	286 + (nd - 1) x 4 286 + (1 - 1) x 4	286	$011E
Command Request	1566 + (nd - 1) 1566 + (1 - 1)	1566	$061E
Command Response	1630 + (nd - 1) 1630 + (1 - 1)	1630	$065E

nd.... Node number of the   ma ...Module address of
        CAN Slave   the AF101

Example 2

Packed Mode

A node consists of ....

- ... CAN bus controller (node number 4)
- ... Digital input module DI439 (module addresses 1 and 2)
  The digital input module DI439 behaves like two 8 channel modules because of its 16 channels.
- ... Digital output module DO722 (module address 3)

Object	Calculation	Identifier	
Alarm Message	222 + (nd - 1) 222 + (4 - 1)	225	$00E1
DI439	286 + (nd - 1) x 4 286 + (4 - 1) x 4	298	$012A
DO722	414 + (nd - 1) x 4 414 + (4 - 1) x 4	426	$01AA
Command Request	1566 + (nd - 1) 1566 + (4 - 1)	1569	$0621
Command Response	1630 + (nd - 1) 1630 + (4 - 1)	1633	$0661

nd.... Node number of the
CAN Slave

Example 3

Unpacked Mode

A node consists of ....

- ... CAN bus controller (node number 12)
- ... Digital input module DI645 (module address 1)
- ... Digital output module DO720 (module address 2)

Object	Calculation	Identifier	
Alarm Message	222 + (nd - 1) 222 + (12 - 1)	233	$00E9
DI645	286 + (nd - 1) x 4 + (ma - 1) 286 + (12 - 1) x 4 + (1 - 1)	330	$014A
DO720	414 + (nd - 1) x 4 + (ma - 1) 414 + (12 - 1) x 4 + (2 - 1)	459	$01CB
Command Request	1566 + (nd - 1) 1566 + (12 - 1)	1577	$0629
Command Response	1630 + (nd - 1) 1630 + (12 - 1)	1641	$0669

nd.... Node number of the   ma ...Module address of
       CAN Slave   the digital I/O module

Example 4

Unpacked Mode

Switching the positions of the modules DI645 and DO720 from example 3 results in the following IDs:

Object	Calculation	Identifier	
Alarm Message	222 + (nd - 1) 222 + (12 - 1)	233	$00E9
DO720	414 + (nd - 1) x 4 + (ma - 1) 414 + (12 - 1) x 4 + (1 - 1)	458	$01CA
DI645	286 + (nd - 1) x 4 + (ma - 1) 286 + (12 - 1) x 4 + (2 - 1)	331	$014B
Command Request	1566 + (nd - 1) 1566 + (12 - 1)	1577	$0629
Command Response	1630 + (nd - 1) 1630 + (12 - 1)	1641	$0669

nd.... Node number of the   ma ...Module address of
       CAN Slave   the digital I/O module

Example 5

Unpacked Mode

A node consists of ....

- ... CAN bus controller (node number 1)
- ... Adapter module AF101 (module address 1)
  analog input module AI351 (slot 1)
  analog input module AI351 (slot 2)
  analog output module AO352 (slot 3)
- ... Digital input module DI435 (module address 2)
- ... Digital output module DO435 (module address 3)

Object	Calculation	Identifier	
Alarm Message	222 + (nd - 1) 222 + (1 - 1)	222	$00DE
AI351	542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1) 542 + (1 - 1) x 16 + (1 - 1) x 4 + (1 - 1)	542	$021E
AI351	542 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1) 542 + (1 - 1) x 16 + (1 - 1) x 4 + (2 - 1)	543	$021F
AO352	1054 + (nd - 1) x 16 + (ma - 1) x 4 + (sl - 1) 1054 + (1 - 1) x 16 + (1 - 1) x 4 + (3 - 1)	1056	$0420
DI435	286 + (nd - 1) x 4 + (ma - 1) 286 + (1 - 1) x 4 + (2 - 1)	287	$011F
DO435 - Outputs	414 + (nd - 1) x 4 + (ma - 1) 414 + (1 - 1) x 4 + (3 - 1)	416	$01A0
DO435 - Inputs	286 + (nd - 1) x 4 + (ma - 1) 286 + (1 - 1) x 4 + (3 - 1)	288	$0120
Command Request	1566 + (nd - 1) 1566 + (1 - 1)	1566	$061E
Command Response	1630 + (nd - 1) 1630 + (1 - 1)	1630	$065E

nd.... Node number of the CAN Slave	ma ...Module address of the digital I/O module or AF101	sl ..... Slot number of the screw-in module on the AF101

Example 6

Unpacked Mode

A node consists of ....

- ... CAN bus controller (node number 4)
- ... Digital input module DI439 (module addresses 1 and 2)
  The digital input module DI439 behaves like two 8 channel modules because of its 16 channels.
- ... Digital output module DO722 (module address 3)

Object	Calculation	Identifier	
Alarm Message	222 + (nd - 1) 222 + (4 - 1)	225	$00E1
DI439 Inputs 1 - 8	286 + (nd - 1) x 4 + (ma - 1) 286 + (4 - 1) x 4 + (1 - 1)	298	$012A
DI439 Inputs 9 - 16	286 + (nd - 1) x 4 + (ma - 1) 286 + (4 - 1) x 4 + (2 - 1)	299	$012B
DO722	414 + (nd - 1) x 4 + (ma - 1) 414 + (4 - 1) x 4 + (3 - 1)	428	$01AC
Command Request	1566 + (nd - 1) 1566 + (4 - 1)	1569	$0621
Command Response	1630 + (nd - 1) 1630 + (4 - 1)	1633	$0661

nd.... Node number of the   ma ...Module address of
     CAN Slave   the digital I/O module

6 ALLOCATING THE CAN IDENTIFIER USING DBT MASTER

If a DBT master is present in the network, the allocation of the CAN identifier is carried out by the DBT master. An object name is required for management. The CAN object names are defined using the method described below.

Make sure that a maximum of 10 receive objects (output data) are created for the node.

6.1 PACKED MODE

Object	Object Name
Alarm Message	000EMCY000xxx
Digital Inputs	000TPDO001xxx
Digital Outputs	000RPDO001xxx
Analog Inputs	000TPDOyyyxxx
Analog Outputs	000RPDOyyyxxx
Command Request	000RPDO900xxx
Command Response	000TPDO900xxx

xxx ....Node number of the CAN Slave

yyy ....3 + no. of the AF101 in the search order - 1

6.2 UNPACKED MODE

Object	Object Name
Alarm Message	000EMCY000xxx
Digital Inputs	000TPDOyyyxxx
Digital Outputs	000RPDOzzzxxx
Analog Inputs	000TPDOaaaxxx
Analog Outputs	000RPDObbbxxx
Command Request	000RPDO900xxx
Command Response	000TPDO900xxx

xxx.... Node number of the CAN Slave

yyy.... 3 + no. of the digital input module in the search order up to the max. value - 1

zzz.... 3 + no. of the digital output module in the search order up to the max. value - 1

aaa ... yyy = 0 $\Rightarrow$ aaa = 3 + no. of analog input module in the search order - 1

yyy $\neq$ 0 $\Rightarrow$ aaa = yyy + no. of the analog input module in the search order

bbb ... zzz = 0 $\Rightarrow$ bbb = 3 + no. of the analog output module in the search - 1

zzz $\neq$ 0 $\Rightarrow$ bbb = zzz + no. of the analog output module in the search order

6.2.1 Examples

Example 1

Packed Mode

A node consists of ....

- ... CAN bus controller (node number 1)
- ... Adapter module AF101 (module address 1)
  analog input module AI351 (slot 1)
  analog input module AI351 (slot 2)
  analog input module AI351 (slot 3)
  The analog input data is packed, since only single channel modules are available.
- ... Digital input module DI435 (module address 2)
- ... Digital output module DO435 (module address 3)

Object	Object Name
Alarm Message	000EMCY000001
3 x AI351	000TPDO003001
DI435	000TPDO001001
DO435 - Outputs	000RPDO001001
DO435 - Inputs	000TPDO001001
Command Request	000RPDO900001
Command Response	000TPDO900001

Example 2

Packed Mode

A node consists of ....

- ... CAN bus controller (node number 4)
- ... Digital input module DI439 (module addresses 1 and 2)
  The digital input module DI439 behaves like two 8 channel modules because of its 16 channels.
- ... Digital output module DO722 (module address 3)

Object	Object Name
Alarm Message	000EMCY000004
DI439	000TPDO001004
DO722	000RPDO001004
Command Request	000RPDO900004
Command Response	000TPDO900004

Example 3

Unpacked Mode

A node consists of ....

- ... CAN bus controller (node number 12)
- ... Two digital input modules DI645 (module addresses 1 and 2)
- ... Two digital input modules DO720 (module addresses 3 and 4)

Object	Object Name
Alarm Message	000EMCY000012
DI645	000TPDO003012
DI645	000TPDO004012
DO720	000RPDO003012
DO720	000RPDO004012
Command Request	000RPDO900012
Command Response	000TPDO900012

Example 4

Unpacked Mode

A node consists of ....

- ... CAN bus controller (node number 1)
- ... Adapter module AF101 (module address 1)
  analog input module AI351 (slot 1)
  analog input module AI351 (slot 2)
  analog output module AO352 (slot 3)
- ... Digital input module DI435 (module address 2)
- ... Digital output module DO435 (module address 3)

Object	Object Name
Alarm Message	000EMCY000001
AI351	000TPDO005001
AI351	000TPDO006001
AO352	000RPDO004001
DI435	000TPDO003001
DO435 - Outputs	000RPDO003001
DO435 - Inputs	000TPDO004001
Command Request	000RPDO900001
Command Response	000TPDO900001

Example 5

Unpacked Mode

A node consists of ....

- ... CAN bus controller (node number 4)
- ... Digital input module DI439 (module addresses 1 and 2)
  The digital input module DI439 behaves like two 8 channel modules because of its 16 channels.
- ... Digital output module DO722 (module address 3)

Object	Object Name
Alarm Message	000EMCY000004
DI439 Inputs 1 - 8	000TPDO003004
DI439 Inputs 9 - 16	000TPDO004004
DO722	000RPDO003004
Command Request	000RPDO900004
Command Response	000TPDO900004

7 MONITORING FUNCTIONS

A CAN bus controller has various monitoring functions. Errors are reacted to in various ways, ranging from an error message to a module reset. The following monitoring functions are carried out by the bus controller:

- Voltage Monitoring Bus Controller
- I/O Module Voltage Monitoring
- Life Guarding
- Idle Time Monitoring
- Output Monitoring
- Watchdog

7.1 VOLTAGE MONITORING BUS CONTROLLER

The supply voltage is continually monitored. If the monitoring function finds an error, the outputs are switched off and the CAN function block is separated from the bus by a reset. A software reset is triggered. As soon as the error is no longer present, the bus controller is initialized and started again (see section: "Start-up Behavior").

7.2 I/O MODULE VOLTAGE MONITORING

The supply voltage is monitored for the I/O modules. If the monitoring function finds an error, the inputs are no longer evaluated. If voltage monitoring finds an error for 30 ms, an error message is sent. If the voltage is then stable again for 30 ms, normal operation is resumed. The bus controller sends the error as an alarm message.

7.3 LIFE GUARDING

If the bus controller is in a class 2 network, the NMT master must send a life guard signal cyclically to the bus controller. If the bus controller doesn't receive a life guard signal for two life guard cycles (2 x Life Guard Time; see section: "Operating Parameters"), all outputs are reset and the bus controller waits for the NMT master.

> **Note:** The life guard signal should be sent from the NMT master at cyclic intervals. It is recommended that you set the cycle time to approx. the half the bus controller life guard time.

7.4 IDLE TIME MONITORING

All outputs must be function blocks are written to cyclically (see parameter: "Idle Time" in section: "Operating Parameters"). If a module's outputs are not described within the idle time, all this module's outputs are reset (to log. 0) and an alarm message is sent.

> **Note:** The outputs should be sent twice within the idle time to guarantee secure communication. Therefore, jitter in the transfer time does not cause errors that may reset the outputs.

7.5 OUTPUT MONITORING

The set output states are compared with the actual outputs states which are read back. If a discrepancy is found by the bus controller (e.g.: short circuit), then an alarm message is sent.

7.6 WATCHDOG

The hardware watchdog must be reset with a pulse after 50 ms at the latest, otherwise a hardware reset is triggered and the node remains reset.

A 1 ms timer interrupt from the bus controller operating system handles resetting the Watchdog.

8 NOTE

Pay attention to these important notes during operation.

- Each identifier is only allowed to exist once!
- An object (identifier) is only allowed to be sent from one node so that conflicts do not occur on the bus.

CHAPTER 6
B&R 2003 TIMING

1 B&R 2003 SYSTEM TIMING

The timing for the B&R Systems 2003 depends on the hardware used. The most important criteria are:

- Which digital modules are being used?
- How many analog data points have to be processed?
- How are the screw-in modules arranged on the individual adapter modules?

The most important difference in timing is a result of the controller.

- 2003 PCC CPU
- Remote I/O Bus Controller e.g. EX477
- CAN Bus Controller e.g. EX470

2 2003 PCC TIMING

⚠ The description refers to the AF101 analog interface modules with revisions ≥02.00.

2.1 CYCLES TO BE CONSIDERED

CP430, CP470, CP474, CP770, CP774

The following three factors must be taken into consideration for timing when one of the B&R 2003 CPUs listed above is used as controller:

Internal bus cycle
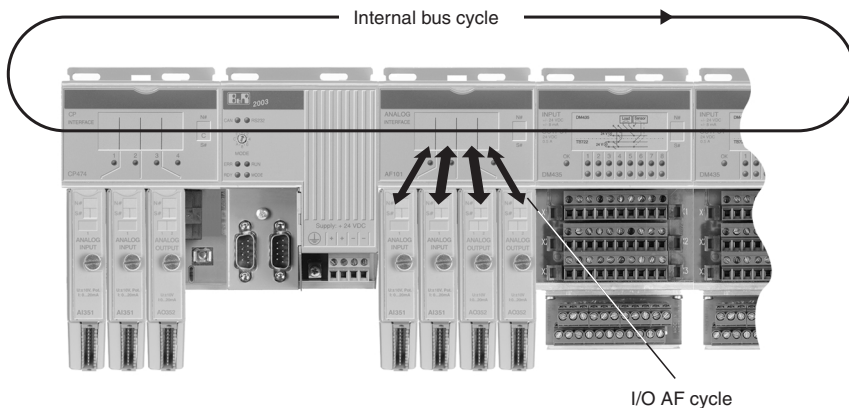
I/O AF cycle

- Internal Bus Cycle
- I/O AF Cycle
- I/O CPU Load

CP476

The following four factors must be taken into consideration for timing when a CP476 B&R 2003 CPU is used as controller:

I/O CP Interface cycle

Internal bus cycle

I/O AF cycle
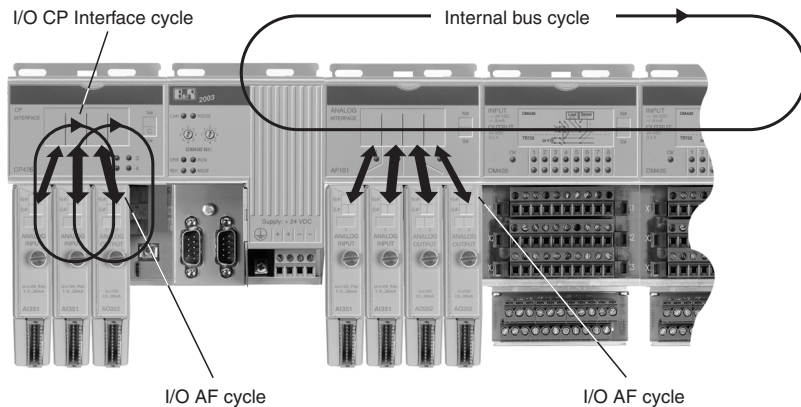
I/O AF cycle

- Internal Bus Cycle
- I/O AF Cycle
- I/O CP Interface Cycle
- I/O CPU Load

2.1.1 Internal Bus Cycle

All AF and digital I/O modules are processed during this time.

CP430 / CP470 / CP476 / CP770
$\begin{aligned} t_{int\_cycle} \quad = \quad & \sum AF\ modules * t_{AF} * n_{data\_words} + \\ & t_{AF\_busy\_AI} * n_{data\_words\_AI} + \\ & t_{AF\_busy\_AO} * n_{data\_words\_AO} + \\ & \sum DIOM\ modules * t_{dig\_IO} * n_{data\_words\_AI} + \\ & t_{Offset} \end{aligned}$

CP474 / CP774
$\begin{aligned} t_{int\_cycle} \quad = \quad & (\sum AF\ modules + 1) * t_{AF} * n_{data\_words} + \\ & t_{AF\_busy\_AI} * n_{data\_words\_AI} + \\ & t_{AF\_busy\_AO} * n_{data\_words\_AO} + \\ & \sum DIOM\ modules * t_{dig\_IO} * n_{data\_words} + t_{Offset} \end{aligned}$

n_{data_words}
$\sum$ data words for the AF101, corresponds to the AF101 module with the most data words in the system.
If a CP474 or CP774 on the CP interface has more data words, the sum of these data words must be used for the calculation.

$n_{data\_words\_AI} = n_{data\_words} - n_{data\_words\_AO}$
Number of remaining analog inputs in n_{data_words}

$n_{data\_words\_AO}$
$\sum$ analog output data words for the AF101, corresponds to the AF101 module with the most analog outputs in the system.   If a CP474 or CP774 on the CP interface has more analog outputs, the sum of these analog output data words must be used for the calculation.
This AF101 does not have to be identical with the one with the most channels (AI + AO)   (see n_{data_words}).

Description	Abbreviation	t [µs]
Adapter module (AF101)	t_{AF}	22

Adapter module busy
 	
    analog inputs
 	
    analog outputs
 	
        AF101 or CP Interface on CP474 or CP774 which is only using analog outputs
 	
        AF101 or CP Interface on CP474 or CP774 which is using mixed connections
	$t_{AF\_busy\_AI}$
	
	$t_{AF\_busy\_AO}$
	
	$t_{AF\_busy\_AO}$
		200
		
		850
		
		1500

Digital inputs (e.g. DI435, DI645) Digital outputs (e.g. DO435, DO720) Digital mixed module (e.g. DM435)	t_{dig_IO}	14
Offset	t_{Offset}	1200

Timing for the CM modules is described for the respective module.

2.1.2 I/O AF Cycle

During this time, the data from the AF module is copied to the screw-in modules. All data points on the respective AF module are processed in one cycle.

I/O AF Cycle
$t_{IO\_AF} = (n_{data\_words} + 1) * t_{ScrM\_AF}$
n_{data_words}
The I/O AF cycle is calculated for the adapter module with the most data words for analog inputs. The sum of the data words is made up of all AI channels, even those which are not used. Only the analog inputs are used for this calculation because writing directly to the analog outputs is already included in the internal bus cycle ($t_{AF\_busy\_AO}$).
t_{ScrM_AF}
Transfer time per AI data word ScrM $\leftrightarrow$ AF $= 650\,\mu s$

2.1.3 I/O CP Interface Cycle for CP476

During this time, the data from the CP interface is copied to the screw-in modules. All data points on the CP interface are processed in one cycle.

To reduce the load on the CPU, an additional I/O processor handles operation of the all I/O data points. Screw-in modules 1 and 2 as well as 3 and 4 are processed parallel to each other by the I/O processor.

The I/O CP interface cycle is calculated for the screw-in module pair which uses the most data words. The sum of the data words is made up of all channels, even those which are not used.

Pair 1 .... Screw-in modules 1 and 3
Pair 2 .... Screw-in modules 2 and 4

I/O CP Interface Cycle for CP476
$t_{IO\_CP} \; = \; (n_{data\_words} + 1) * t_{ScrM\_CP}$
n_{data_words}
$\sum$ data words for the screwin module pair with the most data words
t_{ScrM_CP}
Transfer time per data word ScrM $\leftrightarrow$ CP Interface $= \; 650\,\mu$s

2.1.4 Selecting the Task Class

Depending on which cycle is longer, the task class is set for the internal bus cycle or the I/O AF cycle (also see calculation examples).

Generally, all digital I/O data is transferred multiple times. Therefore a minimum cycle time of 3 ms is permissible for digital I/O data.

⚠ **In this case, the data for the screw-in modules is not refreshed each cycle !**

2.1.5 I/O CPU Load

This time determines how long the CPU requires to process the I/O data passed on by the AF module. The CPU is loaded considerably by the analog I/O data.

CP430 / CP470 / CP770
$t_{IO\_CPU} = n_{data\_words} * t_{CPx70\_AF}$
CP474 / CP774
$t_{IO\_CPU} = n_{data\_words} * t_{CPx74\_AF}$
CP476
$t_{IO\_Proc} = n_{data\_words} * t_{CP476\_AF}$
n_{data_words}
$\sum$ data words for the AF101, corresponds to the AF101 module with the most data words in the system. The sum of the data words is made up of all channels, even those which are not used.
If a CP474, CP476 or CP774 on the CP interface has more data words, the sum of these data words must be used for the calculation.
If an analog module (e.g. CM211) has more data words, then the number of data words for the analog module must be used for the calculation.

Description	Abbreviation	t [µs]
Analog data point on CP430/CP470/CP770	t_{CPx70_AF}	100
Analog data point on CP474/CP774	t_{CPx74_AF}	70
Analog data point on CP476 This is the time require by the additional I/O processor to process an analog data point.	t_{CP476_AF}	50

The data words for the digital screw-in modules DI135 and DO135 are counted as analog data points.

2.1.6 Worst Case Reaction Time

The worst case reaction time for the system is calculated using the following formula:

Worst Case Reaction Time = Internal Bus Cycle + I/O AF Cycle + I/O CPU Load

2.2 ACCESS PROCEDURE

Cyclic with TC#1 or HSTC

Cyclic with TC#1 or HSTC
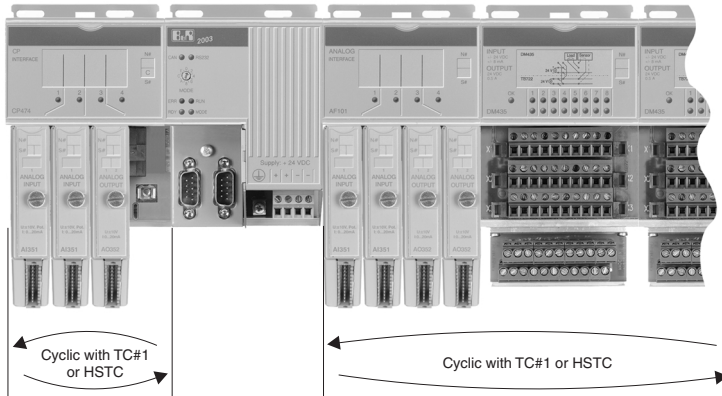
- Cyclic in task class 1 or HS task class

2.2.1 Cyclic Access

Access can take place in two ways:

1) Cyclic with the system manager (10 ms).

2) If I/O points are used in the HSTC, access takes place automatically in relation to this HSTC (e.g. 5 ms), because it is started more often than the system manager.

The I/O AF cycle is determined by the adapter module with the most data words.

2.3 CALCULATION EXAMPLES

Example 1

<u>CP474/774</u>

32 digital IN, 32 digital OUT:	4 x DM435
6 analog IN, 4 analog OUT:	1 x AF101 [2 x AI354, 2 x AO352]
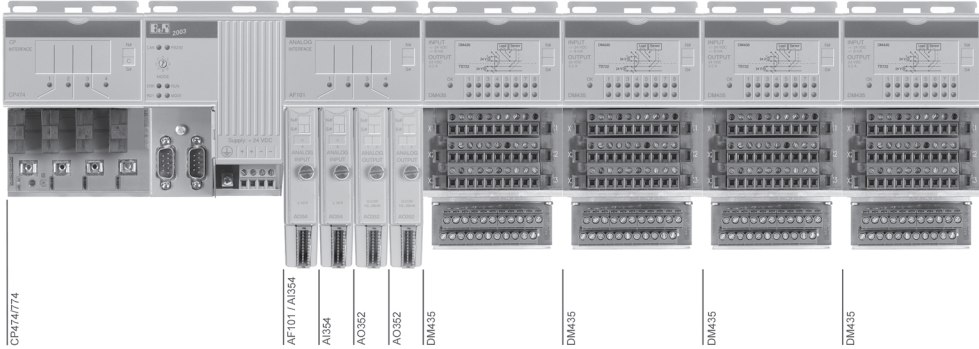
<u>Internal Bus Cycle</u>

$$t_{int\_cycle} = (\Sigma \text{ AF modules} + 1) * t_{AF} * n_{data\_words} + n_{data\_words\_AI} * t_{AF\_busy\_AI} + n_{data\_words\_AO} * t_{AF\_busy\_AO} +$$
$$\Sigma \text{ DIOM modules} * t_{dig\_IO} * n_{data\_words} + t_{Offset}$$
$$t_{int\_cycle} = 2 * t_{AF} * 12 + 8 * t_{AF\_busy\_AI} + 4 * t_{AF\_busy\_AO} + 4 * t_{dig\_IO} * 12 + t_{Offset}$$
$$t_{int\_cycle} = 2 * 22 * 12 + 8 * 200 + 4 * 1500 + 4 * 14 * 12 + 1200$$
$$t_{int\_cycle} = 10000 \, \mu s$$

<u>I/O AF Cycle</u>

$$t_{IO\_AF} = (n_{data\_words} + 1) * t_{ScrM\_AF}$$
$$t_{IO\_AF} = (2 * 4 + 1) * 650$$
$$t_{IO\_AF} = 5850 \, \mu s$$

<u>I/O CPU Load</u>

$$t_{IO\_CPU} = n_{data\_words} * t_{CPx74\_AF}$$
$$t_{IO\_CPU} = (2 * 4 + 2 * 2) * 70$$
$$t_{IO\_CPU} = 840 \, \mu s$$

<u>Fastest Task Class Recommended</u>

10 ms   (determined by internal bus cycle)

The 3 ms task class can also be used to process the digital I/O signals. However, that means the analog data will not be refreshed every cycle.

Example 2

CP474/774

32 digital IN, 32 digital OUT:	4 x DM435
6 analog IN, 4 analog OUT:	1 x AF101 [3 x AI351 + 1 x AO352]
	1 x AF101 [3 x AI351 + 1 x AO352]
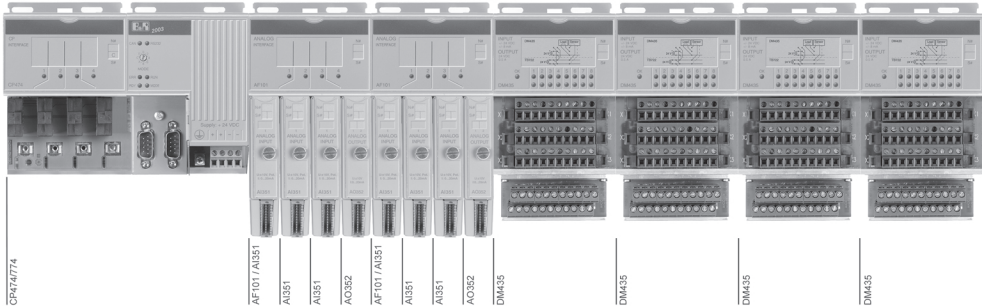
Internal Bus Cycle

$$t_{int\_cycle} = (\Sigma \text{ AF modules} + 1) * t_{AF} * n_{data\_words} + n_{data\_words\_AI} * t_{AF\_busy\_AI} + n_{data\_words\_AO} * t_{AF\_busy\_AO} +$$
$$\Sigma \text{ DIOM modules} * t_{dig\_IO} * n_{data\_words} + t_{Offset}$$
$$t_{int\_cycle} = 3 * t_{AF} * 5 + 3 * t_{AF\_busy\_AI} + 2 * t_{AF\_busy\_AO} + 4 * t_{dig\_IO} * 5 + t_{Offset}$$
$$t_{int\_cycle} = 3 * 22 * 5 + 3 * 200 + 2 * 1500 + 4 * 14 * 5 + 1200$$
$$t_{int\_cycle} = 5410 \, \mu s$$

I/O AF Cycle

$$t_{IO\_AF} = (n_{data\_words} + 1) * t_{ScrM\_AF}$$
$$t_{IO\_AF} = (3 * 1 + 1) * 650$$
$$t_{IO\_AF} = 2600 \, \mu s$$

I/O CPU Load

$$t_{IO\_CPU} = n_{data\_words} * t_{CPx74\_AF}$$
$$t_{IO\_CPU} = (3 * 1 + 2) * 70$$
$$t_{IO\_CPU} = 350 \, \mu s$$

Fastest Task Class Recommended

6 ms   (determined by internal bus cycle)

The 3 ms task class can also be used to process the digital I/O signals. However, that means the analog data will not be refreshed every cycle.

Example 3

To allow a timing comparison, examples 3 and 4 use the same hardware configuration except for the CPU.

Example 3 ...... CP476
Example 4 ...... CP474/CP774

CP476

32 digital IN, 32 digital OUT: 4 x DM435
6 analog IN, 2 analog OUT: CP Interface [2 x NC161 + 1 x AO352]
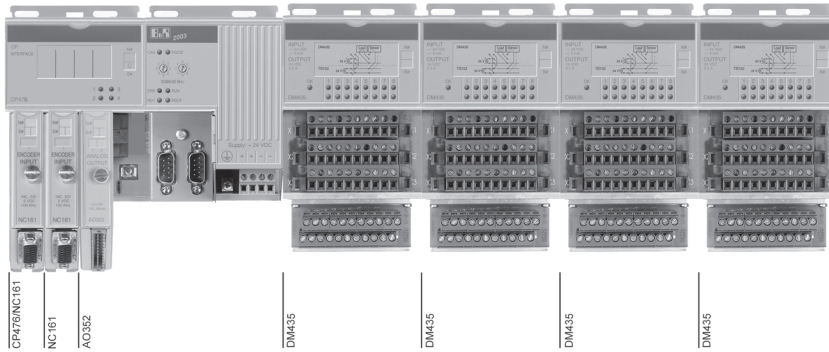
Internal Bus Cycle

$$t_{int\_cycle} = \Sigma \text{ DIOM modules } * t_{dig\_IO} * 1 + t_{Offset}$$
$$t_{int\_cycle} = 4 * t_{dig\_IO} * 1 + t_{Offset}$$
$$t_{int\_cycle} = 4 * 14 * 1 + 1200$$
$$t_{int\_cycle} = 1256 \ \mu s$$

I/O CP Interface Cycle on CP476

$$t_{IO\_CP} = (n_{data\_words} + 1) * t_{ScrM\_CP}$$
$$t_{IO\_CP} = (1 * 3 + 1 * 2 + 1) * 650$$
$$t_{IO\_CP} = 3900 \ \mu s$$

I/O CPU Load

$$t_{IO\_Proc} = n_{data\_words} * t_{CP476\_AF}$$
$$t_{IO\_Proc} = 8 * 50$$
$$t_{IO\_Proc} = 400 \ \mu s$$

Fastest Task Class Recommended

5 ms   (determined by I/O CP Interface Cycle on CP476)

The 3 ms task class can also be used to process the digital I/O signals. However, that means the analog data will not be refreshed every cycle.

Example 4

To allow a timing comparison, examples 3 and 4 use the same hardware configuration except for the CPU.

   Example 3 ...... CP476
   Example 4 ...... CP474/CP774

CP474/774

   32 digital IN, 32 digital OUT: 4 x DM435
   6 analog IN, 2 analog OUT: CP Interface [2 x NC161 + 1 x AO352]
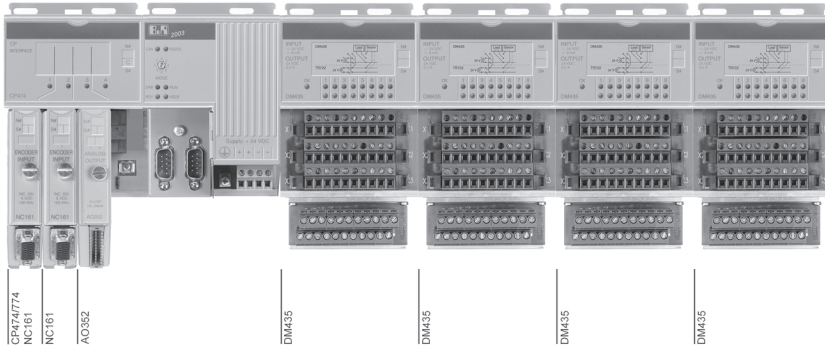
Internal Bus Cycle

$$t_{int\_cycle} = (\Sigma \text{ AF modules} + 1) * t_{AF} * n_{data\_words} + n_{data\_words\_AI} * t_{AF\_busy\_AI} + n_{data\_words\_AO} * t_{AF\_busy\_AO} +$$
$$\Sigma \text{ DIOM modules} * t_{dig\_IO} * n_{data\_words} + t_{Offset}$$
$$t_{int\_cycle} = 1 * t_{AF} * 8 + 6 * t_{AF\_busy\_AI} + 2 * t_{AF\_busy\_AO} + 4 * t_{dig\_IO} * 8 + t_{Offset}$$
$$t_{int\_cycle} = 1 * 22 * 8 + 6 * 200 + 2 * 1500 + 4 * 14 * 8 + 1200$$
$$t_{int\_cycle} = 6024 \ \mu s$$

I/O AF Cycle on CP474/774

$$t_{IO\_AF} = (n_{data\_words} + 1) * t_{ScrM\_AF}$$
$$t_{IO\_AF} = (2 * 3 + 1) * 650$$
$$t_{IO\_AF} = 4550 \ \mu s$$

I/O CPU Load

$$t_{IO\_CPU} = n_{data\_words} * t_{CPx74\_AF}$$
$$t_{IO\_CPU} = 8 * 70$$
$$t_{IO\_CPU} = 560 \ \mu s$$

Fastest Task Class Recommended

   7 ms   (determined by internal bus cycle)

   The 3 ms task class can also be used to process the digital I/O signals. However, that means the analog data will not be refreshed every cycle.

   The timing for examples 4 and 5 are the same except for a small difference in the internal bus cycle time.

Example 5

<u>CP474/774</u>

32 digital IN, 32 digital OUT:	4 x DM435
2 axes, 2 analog OUT:	1 x AF101 [2 x NC161, 1 x AO352]
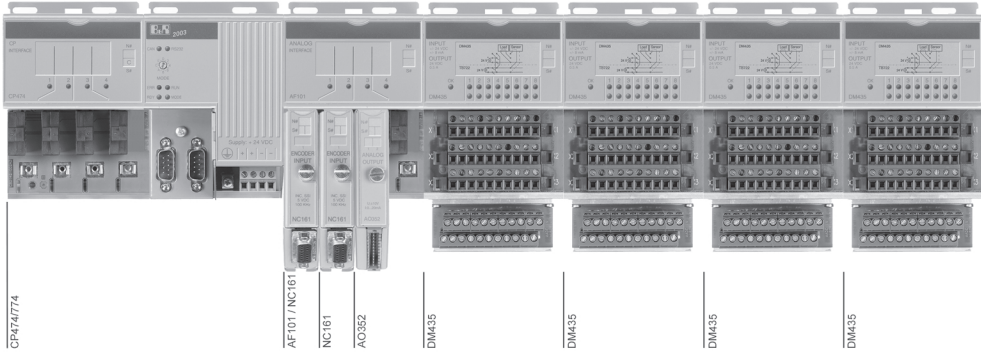
<u>Internal Bus Cycle</u>

$$t_{int\_cycle} = (\Sigma \text{ AF modules} + 1) * t_{AF} * n_{data\_words} + n_{data\_words\_AI} * t_{AF\_busy\_AI} + n_{data\_words\_AO} * t_{AF\_busy\_AO} +$$
$$\Sigma \text{ DIOM modules} * t_{dig\_IO} * n_{data\_words} + t_{Offset}$$
$$t_{int\_cycle} = 2 * t_{AF} * 8 + 6 * t_{AF\_busy\_AI} + 2 * t_{AF\_busy\_AO} + 4 * t_{dig\_IO} * 8 + t_{Offset}$$
$$t_{int\_cycle} = 2 * 22 * 8 + 6 * 200 + 2 * 1500 + 4 * 14 * 8 + 1200$$
$$t_{int\_cycle} = 6200 \, \mu s$$

<u>I/O AF Cycle</u>

$$t_{IO\_AF} = (n_{data\_words} + 1) * t_{ScrM\_AF}$$
$$t_{IO\_AF} = (2 * 3 + 1) * 650$$
$$t_{IO\_AF} = 4550 \, \mu s$$

<u>I/O CPU Load</u>

$$t_{IO\_CPU} = n_{data\_words} * t_{CPx74\_AF}$$
$$t_{IO\_CPU} = (+1 * 3 + 2 * 2) * 70$$
$$t_{IO\_CPU} = 560 \, \mu s$$

<u>Fastest Task Class Recommended</u>

7 ms (determined by internal bus cycle)

The 3 ms task class can also be used to process the digital I/O signals. However, that means the analog data will not be refreshed every cycle.

The timing for examples 4 and 5 are the same except for a small difference in the internal bus cycle time.

Example 6

CP474/774

32 digital IN, 32 digital OUT:	4 x DM435
2 analog IN, 8 thermocouples:	1 x AF101 [2 x AI351 + 2 x AT664]
4 analog OUT:	1 x AF101 [2 x AO352]
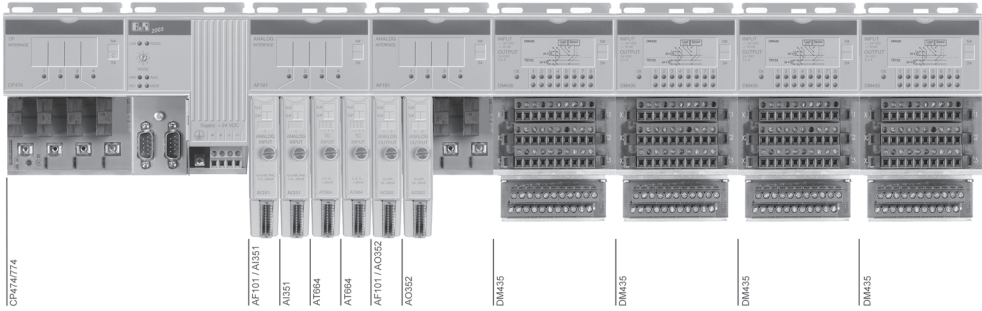
Internal Bus Cycle

$$t_{int\_cycle} = (\Sigma \text{ AF modules} + 1) * t_{AF} * n_{data\_words} + n_{data\_words\_AI} * t_{AF\_busy\_AI} + n_{data\_words\_AO} * t_{AF\_busy\_AO} +$$
$$\Sigma \text{ DIOM modules} * t_{dig\_IO} * n_{data\_words} + t_{Offset}$$
$$t_{int\_cycle} = 3 * t_{AF} * 10 + 6 * t_{AF\_busy\_AI} + 4 * t_{AF\_busy\_AO} + 4 * t_{dig\_IO} * 10 + t_{Offset}$$
$$t_{int\_cycle} = 3 * 22 * 10 + 6 * 200 + 4 * 850 + 4 * 14 * 10 + 1200$$
$$t_{int\_cycle} = 7020 \ \mu s$$

I/O AF Cycle

$$t_{IO\_AF} = (n_{data\_words} + 1) * t_{ScrM\_AF}$$
$$t_{IO\_AF} = (2 * 1 + 2 * 4 + 1) * 650$$
$$t_{IO\_AF} = 7150 \ \mu s$$

I/O CPU Load

$$t_{IO\_CPU} = n_{data\_words} * t_{CPx74\_AF}$$
$$t_{IO\_CPU} = (4 * 1 + 2 * 2) * 70$$
$$t_{IO\_CPU} = 700 \ \mu s$$

Fastest Task Class Recommended

8 ms   (determine by I/O AF cycle)

The 3 ms task class can also be used to process the digital I/O signals. However, that means the analog data will not be refreshed every cycle.

Example 7

CP474/774

32 digital IN, 32 digital OUT:	4 x DM435
2 analog IN, 8 thermocouples, 4 analog OUT:	1 x AF101 [1 x AI351 + 1 x AT664 + 1 * AO352]
	1 x AF101 [1 x AI351 + 1 x AT664 + 1 * AO352]
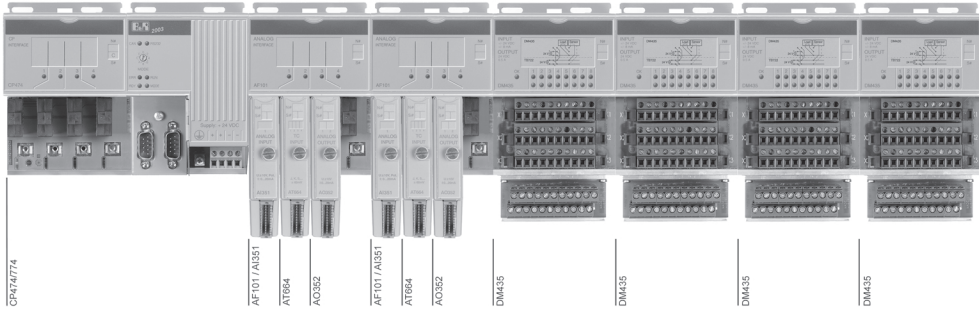
Internal Bus Cycle

$$t_{int\_cycle} = (\Sigma \text{ AF modules} + 1) * t_{AF} * n_{data\_words} + n_{data\_words\_AI} * t_{AF\_busy\_AI} + n_{data\_words\_AO} * t_{AF\_busy\_AO} +$$
$$\Sigma \text{ DIOM modules} * t_{dig\_IO} * n_{data\_words} + t_{Offset}$$
$$t_{int\_cycle} = 3 * t_{AF} * 7 + 5 * t_{AF\_busy\_AI} + 2 * t_{AF\_busy\_AO} + 4 * t_{dig\_IO} * 7 + t_{Offset}$$
$$t_{int\_cycle} = 3 * 22 * 7 + 5 * 200 + 2 * 1500 + 4 * 14 * 7 + 1200$$
$$t_{int\_cycle} = 6054 \, \mu s$$

I/O AF Cycle

$$t_{IO\_AF} = (n_{data\_words} + 1) * t_{ScrM\_AF}$$
$$t_{IO\_AF} = (1 + 4 + 1) * 650$$
$$t_{IO\_AF} = 3900 \, \mu s$$

I/O CPU Load

$$t_{IO\_CPU} = n_{data\_words} * t_{CPx74\_AF}$$
$$t_{IO\_CPU} = (1 + 4 + 2) * 70$$
$$t_{IO\_CPU} = 490 \, \mu s$$

Fastest Task Class Recommended

7 ms   (determined by internal bus cycle)

The 3 ms task class can also be used to process the digital I/O signals. However, that means the analog data will not be refreshed every cycle.

3 REMOTE I/O NODE TIMING

3.1 TIMES TO BE CONSIDERED

If a remote I/O bus controller is used as controller, the following three times must be considered when calculating the timing for a remote I/O node:
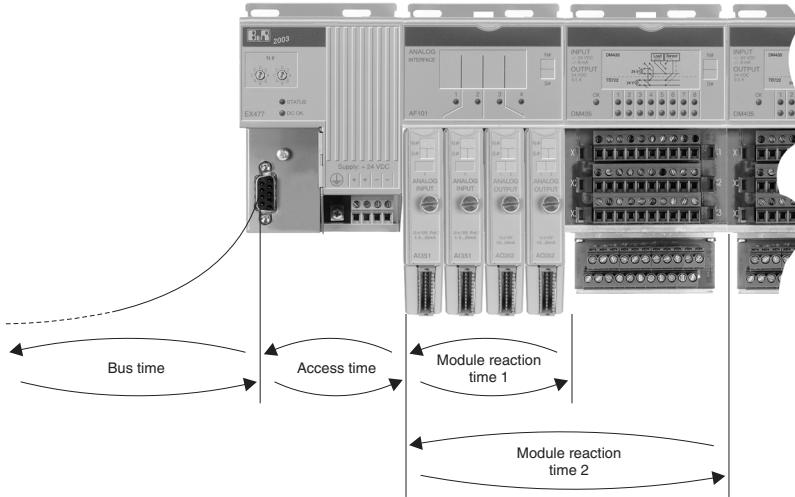
- Access Time
- Module Reaction Time
- Bus Time

3.1.1 Access Time

The time required by the remote I/O bus controller to access the B&R 2003 modules.

3.1.2 Module Reaction Time

This time is required by the remote I/O bus controller to update the output states and to collect the input states.

3.1.3 Bus Time

The data transfer time between the remote I/O master and remote I/O bus controller is called the bus time.

3.2 ACCESS TIME

The time required by the remote I/O bus controller to access the B&R 2003 modules. The access time depends on the requirements of the system.

Description	Abbreviation	t [µs]
Digital system modules	t_{dig_acc}	400
Analog or digital and analog system modules	t_{an_acc}	800

3.3 MODULE REACTION TIME

The module reaction time affects how fast the data points can be updated for each individual module.
It must be calculated for each adapter module and for all digital inputs and outputs.
The module reaction time depends on the system requirements.

3.3.1 Hardware Dependant Internal I/O Bus Cycle

Description	Abbreviation	t [µs]
Adapter module (AF101)	t_{AF}	200
Adapter module busy	t_{AF_busy}	0 [1]
Digital inputs (e.g. DI435, DI645)	t_{DI}	95
Digital outputs (e.g. DO435, DO720) Digital mixed module (e.g. DM435)	t_{DO}	115

[1] t_{AF_busy} is not needed because the time is already accounted for in t_{an_acc}.

3.3.2 Module Reaction Time Calculation

The following times must be calculated:

- Reaction time for digital modules ($t_{dig\_reaction}$)
- Reaction time for analog modules ($t_{an\_reaction}$)

3.3.3 Digital Module Reaction Time

Calculated for all digital inputs and digital outputs.

$$t_{dig\_reaction} = 2 * t_{IO\_cycle}$$

Internal I/O Bus Cycle Time

During this time, all digital inputs/outputs and one data word per adapter module are processed.
The formula used to calculate the internal I/O bus cycle time depends on the hardware configuration.

    1)   If only digital modules are used

$$t_{IO\_cycle} = \ \Sigma \ DO/DM \ modules * t_{DO} +$$
$$\Sigma \ DI \ modules * t_{DI} +$$
$$t_{dig\_acc}$$

    2)   If analog modules are also used

$$t_{IO\_cycle} = \ \Sigma \ AF \ modules * t_{AF} +$$
$$\Sigma \ DO/DM \ modules * t_{DO} +$$
$$\Sigma \ DI \ modules * t_{DI} +$$
$$t_{an\_acc}$$

3.3.4 Analog Module Reaction Time

Must be calculated for **every** adapter module. The sum of the data words is made up of all channels, even those which are not used.
The data words for the digital screw-in modules DI135 and DO135 must also be considered.

$$t_{an\_reaction} = (n_{data\_words} + 1) * t_{IO\_cycle}$$

$$n_{data\_words} = \Sigma \ data \ words \ on \ AF101$$

Internal I/O Bus Cycle Time

Calculated using formula for t_{IO_cycle} see "Internal I/O Bus Cycle Time, Item 2" in the "Digital Module Reaction Time" section.

3.4 HARDWARE GUIDELINES

The following guidelines will help you find the optimal configuration for your B&R 2003 System.

- If fast inputs/outputs are required, the respective screw-in modules must be divided up among several adapter modules. Therefore, the individual data points are processed more often.

- Install screw-in modules on several adapter modules so that each adapter module is equipped with the same number of inputs and outputs.

3.5 CALCULATION EXAMPLES FOR MODULE REACTION TIME

Example 1

EX477/777

64 digital IN, 64 digital OUT: 8 x DM435

$t_{dig\_reaction} = 2 * t_{IO\_cycle}$
$t_{dig\_reaction} = 2 * (\sum DO/DM\ modules * t_{DO} + t_{dig\_acc})$
$t_{dig\_reaction} = 2 * (8 * t_{DO} + t_{dig\_acc})$
$t_{dig\_reaction} = 2 * (8 * 115 + 400)$
$t_{dig\_reaction} = 2640\,\mu s$

Example 2

EX477/777

32 digital IN, 32 digital OUT:	4 x DM435
6 analog IN, 4 analog OUT:	1 x AF101 [4 x AI351]
	1 x AF101 [1 x AI354 + 2 x AO352]
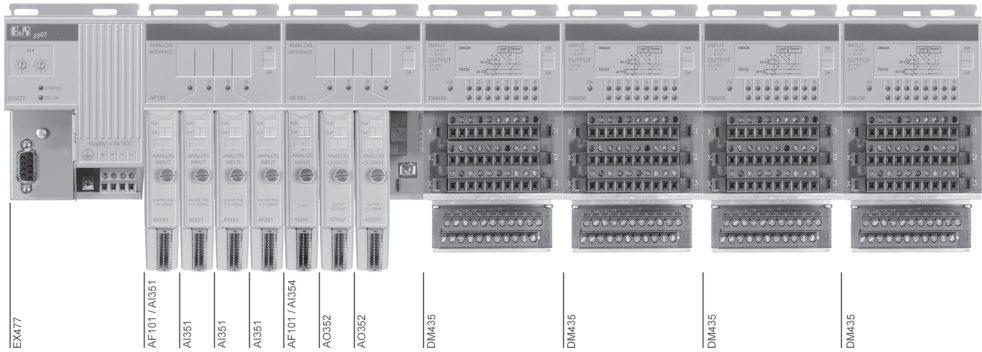
	AF Module	DO/DM Module	DI Module	Data Words on First AF	Data Words on Second AF
AF101	1				
AI351				1	
AI351				1	
AI351				1	
AI351				1	
AF101	1				
AI354					4 [1]
AO352					2
AO352					2
DM435		1			
DM435		1			
DM435		1			
DM435		1			
Summe:	2	4	0	4	8

[1] Only two channels are used on the AI354 module. However, all four data words must be used for the calculation.

3.5.1 Calculating Module Reaction Times

Internal I/O Bus Cycle Time

$$t_{IO\_cycle} = \Sigma \text{ AF modules} * t_{AF} + \Sigma \text{ DO/DM modules} * t_{DO} + t_{an\_acc}$$
$$t_{IO\_cycle} = 2 * t_{AF} + 4 * t_{DO} + t_{an\_acc}$$
$$t_{IO\_cycle} = 2 * 200 + 4 * 115 + 800$$
$$t_{IO\_cycle} = 1660 \, \mu s$$

Digital Module Reaction Time

$$t_{dig\_reaction} = 2 * t_{IO\_cycle}$$
$$t_{dig\_reaction} = 2 * 1660$$
$$t_{dig\_reaction} = 3320 \, \mu s$$

Analog Module Reaction Time

1. AF101

$$t_{an\_reaction} = (n_{data\_words} + 1) * t_{IO\_cycle}$$
$$t_{an\_reaction} = (4 + 1) * 1660$$
$$t_{an\_reaction} = 8300 \, \mu s$$

2. AF101

The second adapter module AF101 is equipped with two analog inputs and four analog outputs.
Only two channels are used on the AI354 module. However, all four data words must be used for the calculation!

$$t_{an\_reaction} = (n_{data\_words} + 1) * t_{IO\_cycle}$$
$$t_{an\_reaction} = (8 + 1) * 1660$$
$$t_{an\_reaction} = 14940 \, \mu s$$

4 CAN NODE TIMING

4.1 TIMES TO BE CONSIDERED

If the CAN bus controller is used as controller, the following three times must be considered when calculating the timing for a CAN node:
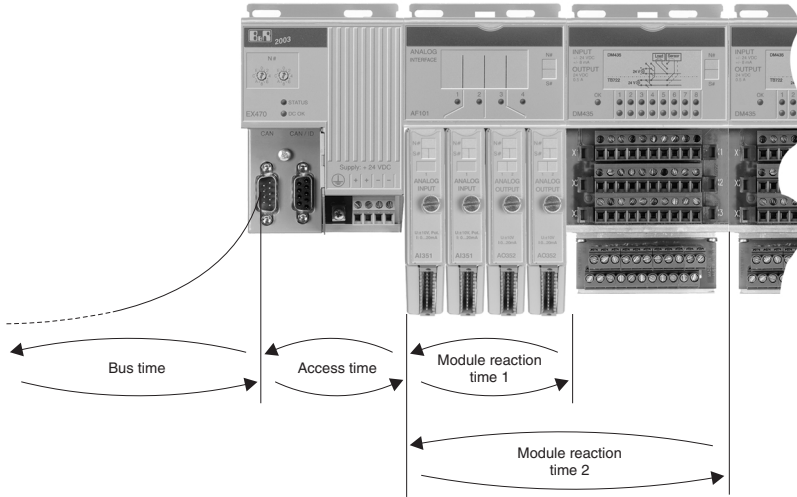
- Access Time
- Module Reaction Time
- Bus Time

4.1.1 Access Time

The time required by the CAN bus controller to access B&R 2003 modules.

4.1.2 Module Reaction Time

This time is required by the CAN bus controller to update the output states and to collect the input states.

4.1.3 Bus Time

The transfer time between CAN client (master) and CAN bus controller is called bus time.

4.2 ACCESS TIME

This time requires the CAN bus controller to access the B&R 2003 modules. The access time depends on the requirements of the system.

Description	Abbreviation	t [µs]
Digital system modules	t_{dig_acc}	660
Analog or digital and analog system modules	t_{an_acc}	1800

4.3 MODULE REACTION TIME

The module reaction time affects how fast the data points can be updated for each individual module.
It must be calculated for each adapter module and for all digital inputs and outputs.
The module reaction time is made up of two areas:

- CAN Object Evaluation
- Hardware Dependent Internal Bus Cycle

4.3.1 CAN Object Evaluation

Description	Abbreviation	t [µs]
Digital input information (send)	t_{DI_send}	400
Analog input information (send)	t_{AI_send}	1600
Digital output information (receive)	t_{DO_rec}	700
Analog output information (receive)	t_{AO_rec}	300

4.3.2 Hardware Dependant Internal Bus Cycle

Description	Abbreviation	t [µs]
Adapter module (AF101)	t_{AF}	840
Adapter module busy	t_{AF_busy}	0 [1]
Digital inputs (e.g. DI435, DI645)	t_{DI}	155
Digital outputs (e.g. DO435, DO720) Digital mixed module (e.g. DM435)	t_{DO}	190

[1] t_{AF_busy} is not needed because the time is already accounted for in t_{an_acc}.

4.3.3 Module Reaction Time Calculation

The following times must be calculated:

- Reaction time for digital modules ($t_{dig\_reaction}$)
- Reaction time for analog modules ($t_{an\_reaction}$)

4.3.4 Digital Module Reaction Time

Calculated for all digital inputs and digital outputs.

$$t_{dig\_reaction} = t_{CAN\_objects} + 2 * t_{IO\_cycle}$$

CAN Object Time

The times t_{AI_send} and t_{AO_rec} for the analog values only have to be included in the calculation if analog modules are present.

$$t_{CAN\_objects} = t_{DI\_send} + t_{DO\_rec} + t_{AI\_send} + t_{AO\_rec}$$

Internal I/O Bus Cycle Time

During this time, all digital inputs/outputs and one data word per adapter module are processed.
The formula used to calculate the internal I/O bus cycle time depends on the hardware configuration.

1) If only digital modules are used

$$t_{IO\_cycle} = \sum DO/DM \text{ modules} * t_{DO} + \\ \sum DI \text{ modules} * t_{DI} + \\ t_{dig\_acc}$$

2) If analog modules are also used

$$t_{IO\_cycle} = \sum AF \text{ modules} * t_{AF} + \\ \sum DO/DM \text{ modules} * t_{DO} + \\ \sum DI \text{ modules} * t_{DI} + \\ t_{an\_acc}$$

4.3.5 Analog Module Reaction Time

Must be calculated for **every** adapter module. The sum of the data words is made up of all channels, even those which are not used.
The data words for the digital screw-in modules DI135 and DO135 must also be considered.

$$t_{an\_reaction} = t_{CAN\_objects} + (n_{data\_words} + 1) * t_{IO\_cycle}$$

$$n_{data\_words} = \Sigma \text{ data words on AF101}$$

CAN Object Time

$$t_{CAN\_objects} = (n_{data\_words} + 1) * (t_{DI\_send} + t_{DO\_rec}) +$$
$$\Sigma \text{ analog input data objects on module rack} * t_{AI\_send} +$$
$$\Sigma \text{ analog output data objects on module rack} * t_{AO\_rec}$$

Internal I/O Bus Cycle Time

Calculated using formula for t_{IO_cycle} see "Internal I/O Bus Cycle Time, Item 2" in the "Digital Module Reaction Time" section.

4.4 HARDWARE GUIDELINES

The following guidelines will help you find the optimal configuration for your B&R 2003 System.

- A group of all single channel screw-in modules (e.g. AI351) on an adapter module. In this way, the I/O data is packed in a CAN object.

- If fast inputs/outputs are required, the respective screw-in modules must be divided up among several adapter modules. Therefore, the individual data points are processed more often.

- Install screw-in modules on several adapter modules so that each adapter module is equipped with the same number of inputs and outputs.

4.5 INHIBIT TIMES

A minimum delay must be inserted between a CAN object for digital and analog inputs with the same ID, before sending repeatedly over the network. This minimum delay is called the Inhibit Time.

4.5.1 Digital Inputs
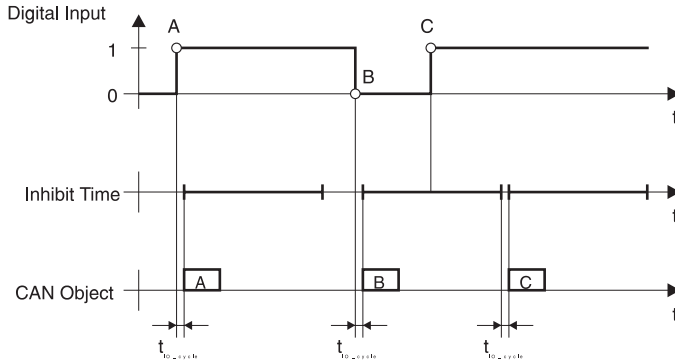
4.5.2 Analog Inputs
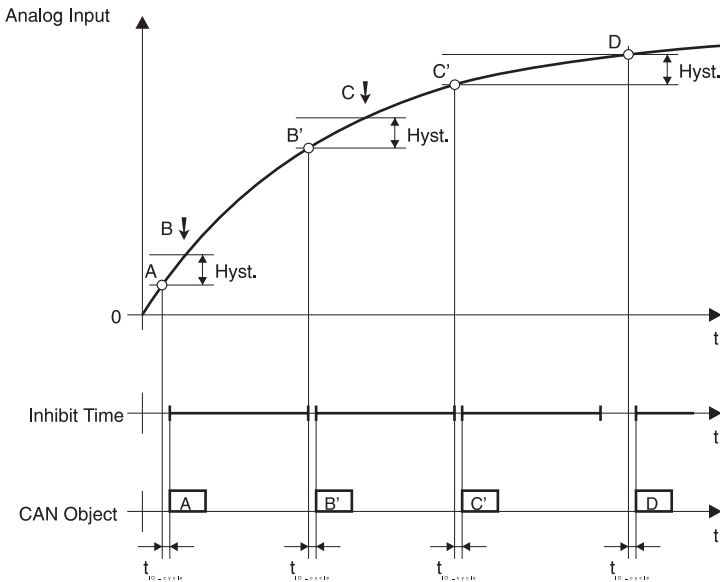
The symbol " ↓ " marks the position (B and C) where the signal on the analog input exceeds the change value defined in the operating parameters. However ,the current value cannot be transferred because of the inhibit time.

After the inhibit time has passed, the current value (B' and C') is transferred with a maximum delay of one I/O cycle.

4.6 DETERMINE THE INHIBIT TIMES

There are three factors when determining the inhibit time:

- Required Time
- Local Acquisition Time
- Maximum Bus Load Desired

4.6.1 Required Time

The reaction times in an application must be guaranteed. If the slowest reaction time allowed is e.g. 25 ms, then 25 ms is the required inhibit time.
If the minimum required time is longer than the time calculated for the "Local Acquisition Time" or "Maximum Bus Load Desired", then defining this value as the inhibit time guarantees reaction within the required time.

4.6.2 Local Acquisition Time

The local acquisition time for data without CAN activity on the controller is accepted as the minimum inhibit time. In order to define this time, the time for CAN object evaluation t_{DI_send}, t_{AI_send}, t_{DO_rec} and t_{AO_rec} must be set to 0.

Minimum Digital Acquisition Time

$$t_{dig\_min\_reaction} = t_{CAN\_objects} + 2 * t_{IO\_cycle} \qquad : t_{CAN\_objects} = 0$$
$$t_{dig\_min\_reaction} = 2 * t_{IO\_cycle}$$

The digital inhibit time $t_{dig\_Inhibit}$ must be $\geq t_{dig\_min\_reaction}$.

Minimum Analog Acquisition Time

$$t_{an\_min\_reaction} = t_{CAN\_objects} + (n_{data\text{-}words} + 1) * t_{IO\_cycle} \qquad : t_{CAN\_objects} = 0$$
$$t_{an\_min\_reaction} = (n_{data\_words} + 1) * t_{IO\_cycle}$$

The analog inhibit time $t_{an\_Inhibit}$ must be $\geq t_{an\_min\_reaction}$.

Additional Load

CAN activity brings about the additional digital and analog loads shown below. The calculated times must be rounded up to whole millisecond values.
The times $t_{DO\_output\_cycle}$ and $t_{AO\_output\_cycle}$ determine the output cycle used by the CAN client to send the digital or analog output information to the CAN nodes.

Digital CAN Load

$$t_{dig\_CAN\_objects} = t_{IO\_cycle} / t_{dig\_Inhibit} * t_{DI\_send} + t_{IO\_cycle} / t_{DO\_output\_cycle} * t_{DO\_rec}$$

Analog CAN Load

$$t_{an\_CAN\_objects} = t_{IO\_cycle} / t_{an\_Inhibit} * t_{AI\_send} + t_{IO\_cycle} / t_{AO\_output\_cycle} * t_{AO\_rec}$$

4.6.3 Maximum Bus Load Desired

For the calculation of the maximum bus load, the CAN objects for all CAN nodes on the bus must be added together.
The bus is allowed to have a maximum load of 85 % in the worst case because parallel service and diagnosis functions still have to be carried out.
A calculation example is shown in the "Bus Time" section. The calculated time is the recommended minimum inhibit time in the system.

4.6.4 Setting the the Inhibit Time

The inhibit time is set with the commands Write Operating Parameters and Activate Operating Parameters.

	Inhibit Time for	
	Digital Inputs	Analog Inputs
Standard settings	5 ms	20 ms
Operating parameter settings	21	27
Value range	0 - 65000 ms	0 - 65000 ms

The commands are explained in chapter 5, "CAN Bus Controller Functions" .

Example

The analog inhibit time is set to 25 ms. The CAN object "Command Request" looks like this:

Byte	Description	Value
0	Code	0
1	Command code	11
2	Parameter 1	27
3	Parameter 2	0
4	Command data	0
5	Command data	25
6	Command data	0
7	Command data	0

The operating parameters are now in a buffer. In order to activate the parameters, the command *Activate Operating Parameters* must be carried out.

Byte	Description	Value
0	Code	0
1	Command code	12
2	Parameter 1	0
3	Parameter 2	0
4	Command data	0
5	Command data	0
6	Command data	0
7	Command data	0

4.7 CALCULATION EXAMPLES FOR MODULE REACTION TIME

4.7.1 Example 1

EX470/770, *packed* operating mode

64 digital IN, 64 digital OUT: 8 x DM435

$t_{dig\_reaction} = t_{CAN\_objects} + 2 * t_{IO\_cycle}$
$t_{dig\_reaction} = t_{DI\_send} + t_{DO\_rec} + 2 * (8 * t_{DO} + t_{dig\_acc})$
$t_{dig\_reaction} = 400 + 700 + 2 * (8 * 190 + 660)$
$t_{dig\_reaction} = 5460\,\mu s$

Recommended Inhibit Time: $t_{Inhibit} = 6$ ms

4.7.2 Example 2

EX470/770, *packed* operating mode

32 digital IN, 32 digital OUT: 4 x DM435

$t_{dig\_reaction} = t_{CAN\_objects} + 2 * t_{IO\_cycle}$
$t_{dig\_reaction} = t_{DI\_send} + t_{DO\_rec} + 2 * (4 * t_{DO} + t_{dig\_acc})$
$t_{dig\_reaction} = 400 + 700 + 2 * (4 * 190 + 660)$
$t_{dig\_reaction} = 3940\,\mu s$

Recommended Inhibit Time: $t_{Inhibit} = 5$ ms (default value)

4.7.3 Example 3

EX470/770, *packed* operating mode

32 digital IN, 32 digital OUT: 4 x DI435 + 4 x DO435

$t_{dig\_reaction} = t_{CAN\_objects} + 2 * t_{IO\_cycle}$
$t_{dig\_reaction} = t_{DI\_send} + t_{DO\_rec} + 2 * (4 * t_{DO} + 4 * t_{DI} + t_{dig\_acc})$
$t_{dig\_reaction} = 400 + 700 + 2 * (4 * 190 + 4 * 155 + 660)$
$t_{dig\_reaction} = 5180\,\mu s$

Recommended Inhibit Time: $t_{Inhibit} = 6$ ms

4.7.4 Example 4

EX470/770, *packed* operating mode

32 digital IN, 32 digital OUT:	4 x DM435
6 analog IN, 4 analog OUT:	1 x AF101 [4 x AI351]
	1 x AF101 [1 x AI354 + 2 x AO352]
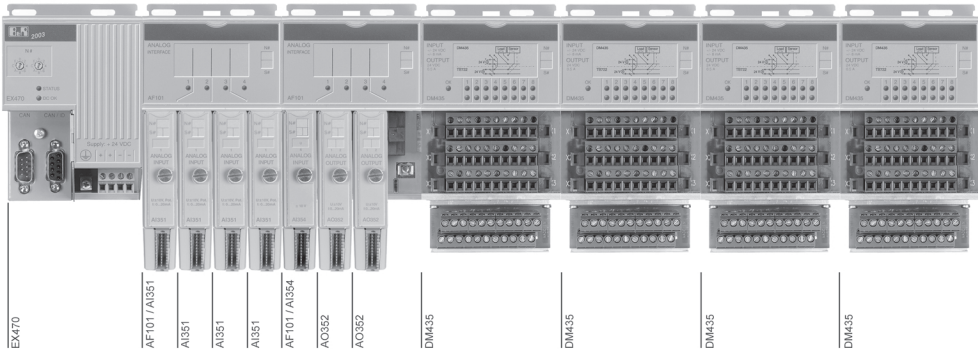
	AF Module	DO/DM Module	DI Module	Data Words on First AF	Data Words on Second AF	Input Data Objects on AF	Output Data Objects on AF
AF101	1						
AI351				1		1	
AI351				1		(packed) [1]	
AI351				1		(packed) [1]	
AI351				1		(packed) [1]	
AF101	1						
AI354					4 [2]	1	
AO352					2		1
AO352					2		1
DM435		1					
DM435		1					
DM435		1					
DM435		1					
Sum:	2	4	0	4	8	2	2

[1] The I/O data for four single channel screw-in modules is packed in **one** CAN object.

[2] Only two channels are used on the AI354 module. However, all four data words must be used for the calculation.

Calculating the Times

The calculation is carried out in two steps:

- Determine the inhibit times
- Calculate the module reaction times according to the inhibit times

Determine the inhibit times

Internal I/O Bus Cycle Time

$t_{IO\_cycle} = \Sigma$ AF modules $* t_{AF} + \Sigma$ DO/DM modules $* t_{DO} + t_{an\_acc}$

$\quad t_{IO\_cycle} = 2 * t_{AF} + 4 * t_{DO} + t_{an\_acc}$

$\quad t_{IO\_cycle} = 2 * 840 + 4 * 190 + 1800$

$\quad t_{IO\_cycle} = 4240\ \mu s$

Digital Module Reaction Time

$\quad t_{dig\_reaction} = t_{CAN\_objects} + 2 * t_{IO\_cycle}$

$\quad t_{dig\_reaction} = t_{DI\_send} + t_{DO\_rec} + t_{AI\_send} + t_{AO\_rec} + 2 * t_{IO\_cycle}$

$\quad t_{dig\_reaction} = 400 + 700 + 1600 + 300 + 2 * 4240$

$\quad t_{dig\_reaction} = 11480\ \mu s$

$\quad$ Recommended Digital Inhibit Time: $t_{dig\_Inhibit} = 10$ ms

Analog Module Reaction Time

1. AF101:

Four analog inputs are on the first AF101 adapter module. The I/O data is packed in a CAN object because four single channel AI351 modules are used. That means, only one CAN object is transferred instead of four. This is taken into account in the multiplier for the time t_{AI_send}. t_{AI_send} is the time required to send an analog input data object to the CAN client.

$\quad t_{an\_reaction} = t_{CAN\_objects} + (n_{data\_words} + 1) * t_{IO\_cycle}$

$\quad t_{an\_reaction} = (n_{data\_words} + 1) * (t_{DI\_send} + t_{DO\_rec}) + 2 * t_{AI\_send} + 2 * t_{AO\_rec} + (n_{data\_words} + 1) * t_{IO\_cycle}$

$\quad t_{an\_reaction} = (4 + 1) * (400 + 700) + 2 * 1600 + 2 * 300 + (4 + 1) * 4240$

$\quad t_{an\_reaction} = 30500\ \mu s$

$\quad$ Recommended Analog Inhibit Time: $t_{an\_Inhibit} = 30$ ms

2. AF101:

The second adapter module AF101 is equipped with two analog inputs and four analog outputs. Only two channels are used on the AI354 module. However, all four data words must be used for the calculation!

$$t_{an\_reaction} = t_{CAN\_objects} + (n_{data\_words} + 1) * t_{IO\_cycle}$$
$$t_{an\_reaction} = (n_{data\_words} + 1) * (t_{DI\_send} + t_{DO\_rec}) + 2 * t_{AI\_send} + 2 * t_{AO\_rec} + (n_{data\_words} + 1) * t_{IO\_cycle}$$
$$t_{an\_reaction} = (8 + 1) * (400 + 700) + 2 * 1600 + 2 * 300 + (8 + 1) * 4240$$
$$t_{an\_reaction} = 51860 \, \mu s$$

Calculation with Inhibit Time

Taking the inhibit times into consideration reduces the digital and analog module reaction time.

Internal I/O Bus Cycle Time

The internal I/O bus cycle time does not change: $t_{IO\_cycle} = 4240 \, \mu s$

Digital Module Reaction Time

The recommended inhibit time for analog inputs is 30 ms. The output cycle time the CAN client uses to provide the CAN nodes with new analog output data, is also fixed to 30 ms. Therefore, the analog data is only sent or received in average every seventh cycle.

$$\text{Factor} = t_{an\_Inhibit} / t_{IO\_cycle} = 30 \text{ ms} / 4.24 \text{ ms} = 7.08 \text{ -> } 7$$

$$t_{dig\_reaction} = t_{CAN\_objects} + 2 * t_{IO\_cycle}$$
$$t_{dig\_reaction} = t_{DI\_send} + t_{DO\_rec} + (t_{AI\_send} + t_{AO\_rec}) / 7 + 2 * t_{IO\_cycle}$$
$$t_{dig\_reaction} = 400 + 700 + (1600 + 300) / 7 + 2 * 4240$$
$$t_{dig\_reaction} = 9852 \, \mu s$$

Analog Module Reaction Time

The recommended inhibit time for digital inputs is 10 ms. The output cycle time the CAN client uses to provide the CAN nodes with new digital output data is also fixed to 10 ms. Therefore, the digital data is only sent or received in average every second cycle.

$$\text{Factor} = t_{dig\_Inhibit} / t_{IO\_cycle} = 10 \text{ ms} / 4.24 \text{ ms} = 2.36 \text{ -> } 2$$

1. AF101:

$$t_{an\_reaction} = t_{CAN\_objects} + (n_{data\_words} + 1) * t_{IO\_cycle}$$
$$t_{an\_reaction} = (n_{data\_words} + 1) * (t_{DI\_send} + t_{DO\_rec}) / 2 + 2 * t_{AI\_send} + 2 * t_{AO\_rec} + (n_{data\_words} + 1) * t_{IO\_cycle}$$
$$t_{an\_reaction} = (4 + 1) * (400 + 700) / 2 + 2 * 1600 + 2 * 300 + (4 + 1) * 4240$$
$$t_{an\_reaction} = 27750 \, \mu s$$

2. AF101:

$$t_{an\_reaction} = t_{CAN\_objects} + (n_{data\_words} + 1) * t_{IO\_cycle}$$
$$t_{an\_reaction} = (n_{data\_words} + 1) * (t_{DI\_send} + t_{DO\_rec}) / 2 + 2 * t_{AI\_send} + 2 * t_{AO\_rec} + (n_{data\_words} + 1) * t_{IO\_cycle}$$
$$t_{an\_reaction} = (8 + 1) * (400 + 700) / 2 + 2 * 1600 + 2 * 300 + (8 + 1) * 4240$$
$$t_{an\_reaction} = 46910 \, \mu s$$

4.7.5 Example 5

EX470/770, *packed* operating mode

32 digital IN, 32 digital OUT:	4 x DM435
1 axis, 1 analog OUT:	1 x AF101 [1 x NC161 + 1 x AO352]
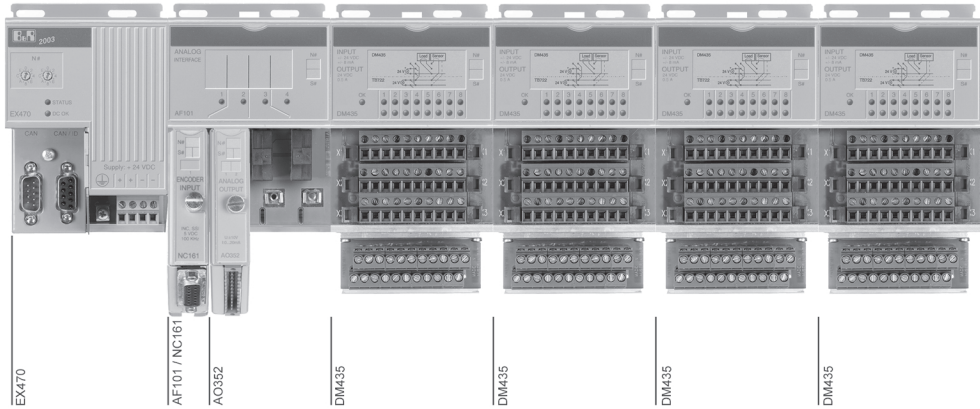
Calculating the Times

The calculation is carried out in two steps:

- Determine the inhibit times
- Calculate the module reaction times according to the inhibit times

Determine the inhibit times

Internal I/O Bus Cycle Time

$t_{IO\_cycle} = \sum$ AF modules $* t_{AF} + \sum$ DO/DM modules $* t_{DO} + t_{an\_acc}$

$\quad t_{IO\_cycle} = 1 * t_{AF} + 4 * t_{DO} + t_{an\_acc}$

$\quad t_{IO\_cycle} = 1 * 840 + 4 * 190 + 1800$

$\quad t_{IO\_cycle} = 3400 \, \mu s$

Digital Module Reaction Time

$\quad t_{dig\_reaction} = t_{CAN\_objects} + 2 * t_{IO\_cycle}$

$\quad t_{dig\_reaction} = t_{DI\_send} + t_{DO\_rec} + t_{AI\_send} + t_{AO\_rec} + 2 * t_{IO\_cycle}$

$\quad t_{dig\_reaction} = 400 + 700 + 1600 + 300 + 2 * 3400$

$\quad t_{dig\_reaction} = 9800 \, \mu s$

Recommended Digital Inhibit Time: $t_{dig\_Inhibit} = 10$ ms

Analog Module Reaction Time

The NC161 has three data words that have to be taken into consideration when calculating the module reaction time.

Only one channel on the AO352 module is used, but both data words must be included in the calculation! One analog input data object and one analog output data object are transferred on the internal bus.

$$t_{an\_reaction} = t_{CAN\_objects} + (n_{data\_words} + 1) * t_{IO\_cycle}$$
$$t_{an\_reaction} = (n_{data\_words} + 1) * (t_{DI\_send} + t_{DO\_rec}) + 1 * t_{AI\_send} + 1 * t_{AO\_rec} + (n_{data\_words} + 1) * t_{IO\_cycle}$$
$$t_{an\_reaction} = (5 + 1) * (400 + 700) + 1 * 1600 + 1 * 300 + (5 + 1) * 3400$$
$$t_{an\_reaction} = 28900\ \mu s$$

Recommended Analog Inhibit Time: $t_{an\_Inhibit} = 25$ ms

Calculation with Inhibit Time

Taking the inhibit times into consideration reduces the digital and analog module reaction time.

Internal I/O Bus Cycle Time

The internal I/O bus cycle time does not change: $\quad t_{IO\_cycle} = 3400\ \mu s$

Digital Module Reaction Time

The recommended inhibit time for analog inputs is 25 ms. The output cycle time the CAN client uses to provide the CAN nodes with new analog output data, is also fixed to 25 ms. Therefore, the analog data is only sent or received in average every seventh cycle.

$$\text{Factor} = t_{an\_Inhibit} / t_{IO\_cycle} = 25\ ms / 3.4\ ms = 7.35 \to 7$$

$$t_{dig\_reaction} = t_{CAN\_objects} + 2 * t_{IO\_cycle}$$
$$t_{dig\_reaction} = t_{DI\_send} + t_{DO\_rec} + (t_{AI\_send} + t_{AO\_rec}) / 7 + 2 * t_{IO\_cycle}$$
$$t_{dig\_reaction} = 400 + 700 + (1600 + 300) / 7 + 2 * 3400$$
$$t_{dig\_reaction} = 8172\ \mu s$$

Analog Module Reaction Time

The recommended inhibit time for digital inputs is 11 ms. The output cycle time the CAN client uses to provide the CAN nodes with new digital output data is also fixed to 11 ms. Therefore, the digital data is only sent or received on average every third cycle.

$$\text{Factor} = t_{dig\_Inhibit} / t_{IO\_cycle} = 11\ ms / 3.4\ ms = 3.24 \to 3$$

$$t_{an\_reaction} = t_{CAN\_objects} + (n_{data\_words} + 1) * t_{IO\_cycle}$$
$$t_{an\_reaction} = (n_{data\_words} + 1) * (t_{DI\_send} + t_{DO\_rec}) / 3 + 1 * t_{AI\_send} + 1 * t_{AO\_rec} + (n_{data\_words} + 1) * t_{IO\_cycle}$$
$$t_{an\_reaction} = (5 + 1) * (400 + 700) / 3 + 1 * 1600 + 1 * 300 + (5 + 1) * 3400$$
$$t_{an\_reaction} = 24500\ \mu s$$

4.8 BUS TIME

The transfer time between CAN client (master) and CAN bus controller is called bus time.

The bus time depends on the following two components:

- Baudrate
- Number of CAN objects

4.8.1 Baudrate

The baudrates listed in the table can be set for the B&R SYSTEM 2003. You have to decide if the settings are to be taken from the configuration memory or node number switch.

Baudrate	Configuration Memory	Node Number Switch
1 Mbit/s [1]	●	
500 kBit/s	●	●
250 kBit/s	●	●
125 kBit/s	●	●
100 kBit/s	●	
50 kBit/s	●	
20 kBit/s	●	●
10 kBit/s	●	

[1] 1 MBit/sec cannot be guaranteed electrically because of isolation properties.

4.8.2 Number of CAN Objects

The CAN objects for all CAN nodes on the bus must be added together when calculating the bus load.

4.8.3 Example

In this example, the calculation for the bus time is made for the worst case. The baudrate is 250 kBaud.

Object Time

A CAN object consists of 8 bytes of data, 34 bits of dynamic overhead and 10 bits of fixed overhead. This results in a length of 108 bits per CAN object.

Theoretically, bit stuffing results in an increase in object length of up to 24 bits. This combination can only occur for **one** object content and **one** certain object number.

Therefore, the calculation of the object time must be made with a length of 132 bits.

$$t_{Object} = 132 / 250000 = 528 \, \mu s$$

The data length of 108 to 132 bits, resulting from bit stuffing, brings about a 22 % dynamic reduction in the bus load depending on the object/data combination. This results in an effective bus load of 63 to 85 %.

CAN Objects

The CAN objects for all CAN nodes on the bus must be added together. In order to make the calculation, **packed operating mode** must be set (default setting).

CAN Objects per Node:

Digital IN .... 1 Object
Digital OUT .... 1 Object
Analog IN .... n_{AI} Objects
Analog OUT .... n_{AI} Objects

$$Obj_{nodes\_x} = 1 + 1 + n_{AI} + n_{AO}$$

Total CAN Objects:

$$Obj_{total} = \Sigma \, Obj_{nodes\_x}$$

100 % Bus Time

This time is required for data transfer if the bus is 100 % free - optimal condition.

$$t_{100\%} = Obj_{total} * t_{Object}$$

Transfer Time

t_{trans} is the worst case transfer time. For the worst case, it is assumed that the bus is available 85 % of the time because parallel service and diagnosis functions still have to be carried out.

In order to ensure proper function of the CAN nodes, you have to guarantee that the transfer time $_{trans}$ is shorter than the idle time for the outputs (default 640 ms).

$$t_{trans} = t_{100\%} / 0.85$$

Recommendation: The shortest inhibit time in the system is not allowed to be shorter than the transfer time t_{trans}.

CHAPTER 7
GENERAL ACCESSORIES

1 GENERAL ACCESSORIES

1.1 OVERVIEW

Model Number	Description	Page
0AC001.9	Retaining clips, 500 pcs.	---
0AC171.9	Glass fuses 5 x 20 mm, 20 pcs., 3,15 A T / 250 V	---
0AC200.9	Lithium batteries, 5 pcs., 3 V / 950 mAh, tubular cell	---
0AC201.9	Lithium batteries, 5 pcs., 3 V / 950 mAh, button cell	---
0AC401.9	Encoder 5 V - 24 V, converter for 5 V encoders (abs. or incr.)	604
0AC410.9	Interface converter TTY - RS232	605
0AC912.9	Bus adapter, CAN, 1 CAN interface	606
0AC913.92	Bus adapter, CAN, 2 CAN interface, including 30 cm cable	607
0AC916.9	Bus termination, RS485, active, for PROFIBUS networks, remote I/O, standard mounting rail installation, supply voltage: 120 / 230 VAC	608
0G0001.00-090	Cable PC <-> RPS/PW, RS232, online cable	---
0G0010.00-090	Cable I/O bus expansion, 1 m, bus expansion for B&R 2005 / B&R 2010	---
0G0012.00-090	Cable I/O bus expansion, 2 m, bus expansion for B&R 2005 / B&R 2010	---
0G1000.00-090	Bus connector, RS485, for PROFIBUS networks, remote I/O	609
0MC111.9	PCMCIA memory card, 2 MB FlashPROM	---
0MC211.9	PCMCIA memory card, 2 MB SRAM	---
7AC911.9	Bus connector, CAN	610
ECINT1-1	RS232/RS485 interface converter, electrically isolated, for coupling RS232 interface modules to an RS485 twisted pair network, without lightning protection	611
ECINT1-11	RS232/RS485 interface converter, electrically isolated, for coupling RS232 interface modules to an RS485 twisted pair network, with lightning protection	611

1.2 AC401 ENCODER 5 V - 24 V

1.2.1 Order Data
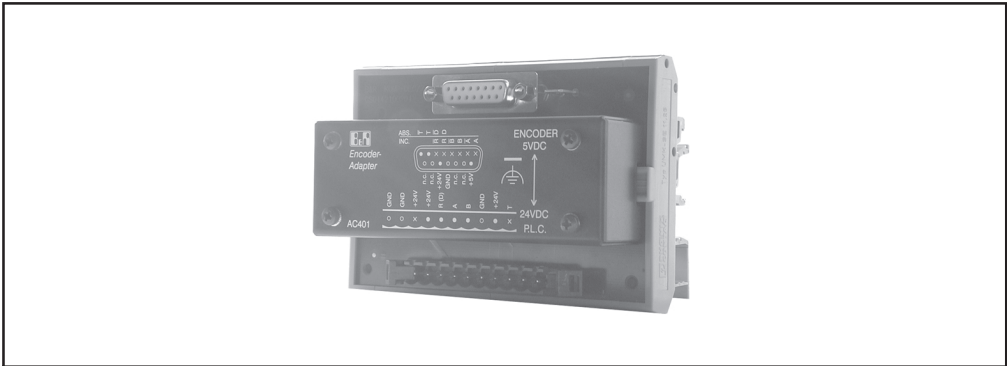
Model Number	Short Description
0AC401.9	Encoder 5 V - 24 V, converter for 5 V encoders (abs. or incr.)

1.2.2 General Information

The adapter is used as a converter for 5 V encoders. The 5 V differential signals delivered by the encoder are converted to 24 V signals. Absolute and incremental encoders can be used.

1.3 AC410 INTERFACE CONVERTER

1.3.1 Order Data

Model Number	Short Description
0AC410.9	Interface Converter TTY - RS232

1.3.2 General Information

The AC410 interface converter is used to convert a TTY signal into an RS232 signal or an RS232 signal into a TTY signal. To be able to connect simple PANELWARE operator panels (e.g. P120 or P121), the 24 V supply voltage is converted into a 5 V output voltage. This voltage can be loaded with up to 0.5 A. The maximum baudrate is 19200 Baud.

1.4 AC912 BUS ADAPTER, CAN 1x

1.4.1 Order Data

Model Number	Short Description
0AC912.9	Bus Adapter, CAN, 1 CAN interface

1.4.2 General Information

The CAN bus adapters is used to connect a controller to a CAN network. The network connection is made using the 6 pin terminal block. The connection to the controller is made using the 9 pin D-type socket. The termination resistor is integrated in the bus adapter. The terminating resistor can be turned on or off.
The cable from the controller to the bus adapter is not in the B&R product line. It must be constructed by the customer.

1.5 AC913 BUS ADAPTER, CAN 2x

1.5.1 Order Data

Model Number	Short Description
0AC913.92	Bus Adapter, CAN, 2 CAN interfaces, including 30 cm connection cable

1.5.2 General Information

The CAN bus adapters is used to connect a controller to a CAN network. The network connection is made using the 9 pin D-type plug (C1) and the 9 pin D-type socket (C2). The 6 pin terminal block has a 30 cm long cable with a D-type housing attached. This cable is used to make the connection to the controller. The termination resistor is integrated in the bus adapter. The terminating resistor can be turned on or off.

1.6 AC916 BUS TERMINATION, RS485 ACTIVE

1.6.1 Order Data

Model Number	Short Description
0AC916.9	Bus Termination, RS485, active, for PROFIBUS networks, remote I/O, standard mounting rail installation, supply voltage: 120 / 230 VAC
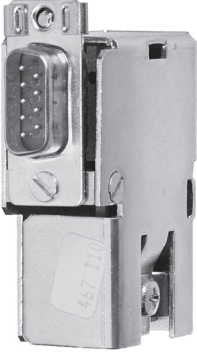
1.6.2 General Information

An active bus termination is available for PROFIBUS networks and remote I/O. The active bus termination allows the network to be terminated independent of the supply for the communication modules.
The supply voltage for the active bus termination is 120 / 230 VAC.

1.7 RS485 BUS CONNECTOR

1.7.1 Order Data

Model Number	Short Description
0G1000.00-090	Bus Connector, RS485, for PROFIBUS Networks, remote I/O

1.7.2 General Information

The RS485 bus connector is used to connect a controller to remote I/O, in a PROFIBUS network or in a RS485 network. The termination resistor is integrated in the bus connector. The terminating resistor can be turned on or off.

1.8 AC911 BUS CONNECTOR, CAN

1.8.1 Order Data

Model Number	Short Description
7AC911.9	Bus Connector, CAN

1.8.2 Technical Data

Description	AC911
Lines	Connections for two bus lines
Termination resistance	120 Ω - can be switched on
Stress relief	Built-in

1.8.3 General Information

The bus connector enables you to:

- Exchange a CAN node without shutting down the network since the connection itself is not broken.
- Change the termination resistance quickly and easily (e.g. if the last node in a network is removed).

1.9 INT1 INTERFACE CONVERTER

1.9.1 Order Data

Model Number	Short Description
ECINT1-1	RS232/RS485 Interface Converter, electrically isolated, for coupling RS232 interface modules to an RS485 twisted pair network, without lightning protection
ECINT1-11	RS232/RS485 Interface Converter, electrically isolated, for coupling RS232 interface modules to an RS485 twisted pair network, with lightning protection

1.9.2 General Information

The INT1 interface converter is used to convert RS232 interface signals to an RS485 signal level. It is used if:

- Data transfer is required over distances that are not possible with an RS232 interface. The distance between two stations can be max. 5000 m when using shielded RS485 cables.
- Electrical isolation of the interface is required.
- A PCC is to be connected to a network using an RS232 interface.

The INT1 interface converter is available with and without lightning protection.

1.9.3 Supply

The INT1 interface converter requires an external 24 VDC supply voltage. Current consumption can be max. 400 mA.

CHAPTER 8
STANDARDS AND CERTIFICATIONS

1 STANDARDS AND LIMITS FOR B&R INDUSTRIAL PRODUCTS

The product standard IEC 61131-2 is generally valid for B&R industrial products. The following standards provide a detailed definition of proper functionality in a typical electromagnetically charged environment:

Standard	Description
IEC 61000-6-4	Electromagnetic compatibility (EMC) Part 2, Generic standards – Section 4: Emission standard for industrial environments. (IEC 50081-2 is replaced by IEC 61000-6-4)
IEC 61000-6-2	Electromagnetic compatibility (EMC) - Part 2, Generic standards - Immunity for industrial environments- (IEC 50082-2 is replaced by IEC 61000-6-2)
IEC CISPR 22	Information technology equipment. Radio disturbance characteristics. Limits and methods of measurement
IEC CISPR 24	Information technology equipment. Immunity characteristics. Limits and methods of measurement
IEC 60204-1	Safety of machinery - Electrical equipment of machines - Part 1 : General requirements
IEC 60950	Safety of information technology equipment
IEC 61000-3-2	Electromagnetic compatibility (EMC) - Part 3: Limits - Section 2: Limits for harmonic current emissions (equipment input current <= 16 A per phase)
IEC 61000-3-3	Electromagnetic compatibility (EMC) - Part 3: Limits - Section 3: Limitation of voltage fluctuations and flicker in low-voltage supply systems for equipment with rated current <= 16 A
IEC 61131-2	Programmable controllers - Part 2: Equipment requirements and tests
IEC 61800-3	Adjustable speed electrical power drive systems - Part 3: EMC product standard including specific test methods
UL 508	Industrial Control Equipment, (UL = Underwriters Laboratories)

Limits

IEC 61000-4-2 Electrostatic Discharge		
	IEC 61131-2	**B&R Limit Value**
Contact discharge to powdered and bare metal parts	4 kV	8 kV
Discharge through the air to plastic parts	8 kV	15 kV

IEC 61000-4-3 Electromagnetic Fields	
Housing, completely wired:	80 MHz - 1 GHz, 10 V/m, 80 % amplitude modulation with 1 kHz

IEC 61000-4-4 Burst (asymmetric fast transient)		
	IEC 61131-2	**B&R Limits**
Power supply	2 kV, 1 min	4 kV, 5 min
All other lines	1 kV, 1 min	2 kV, 5 min

IEC 61000-4-5 Surge		
	Limits CM, unsymmetrical	**Limits DM, symmetrical**
AC power supply	2 kV (12 Ω)	1 kV (2 Ω)
DC power supply	1 kV (12 Ω)	0.5 kV (2 Ω)
Digital and analog I/O, AC, unshielded AC auxiliary voltage outputs for sensors, etc.	2 kV (42 Ω)	1 kV (42 Ω)
Digital and analog I/O, DC, unshielded Data lines, unshielded DC auxiliary voltage outputs for sensors, etc.	0.5 kV (42 Ω)	0.5 kV (42 Ω)
All shielded lines	1 kV (2 Ω)	---

IEC 61000-4-6 Conducted Disturbances (radio frequency)	
Network connections Signals >10 m Functional ground	150 kHz – 80 MHz, 10 V, (in broadcast range 3 V) 80 % amplitude modulation with 1 kHz

IEC 60664-1   Pollution Degree
Pollution degree 2:   non-conductive pollution

IEC 60068-2-6, Test Fc Vibration Test		
Frequency Range [Hz]	**Continuous**	**Periodic**
10 $\leq$ f < 57	0.0375 mm amplitude	0.075 mm amplitude
57 $\leq$ f $\leq$ 150	0.5 g constant acceleration	1 g constant acceleration
f > 150	not defined	not defined

IEC 60068-2-27, Test Ea   Shock Test
Periodic peaks up to 15 g over 11 ms, half sine wave in all three perpendicular axes.

2 INTERNATIONAL STANDARDS

B&R products and services comply with all applicable standards. They are international standards from organizations such as ISO, IEC and CENELEC, as well as national standards from organizations such as UL, CSA, FCC, VDE, ÖVE etc. We devote special attention to the certification of our products for industrial environments. Therefore, e.g. the requirements of product standard IEC 61131-2 concerning electromagnetic immunity are exceeded considerably.

Certifications	
USA and Canada	All important B&R products are tested and listed by Underwriters Laboratories and are checked quarterly by a UL inspector.
	This mark is valid for the USA and Canada and eases certification of your machines and systems in these areas.
Europe	All harmonized IEC standards for the valid guidelines are met.
Russian Federation	B&R has a GOST certification for all products for export to the Russian Federation.

APPENDIX A
CAN IDENTIFIER WITH
CAN BUS CONTROLLER
(FIXED ALLOCATIONS)

Appendix A
CAN Identifier with
CAN Bus Controller

Node Number	Module Number	Digital Inputs		Digital Outputs	
1	1	286	$011E	414	$019E
1	2	287	$011F	415	$019F
1	3	288	$0120	416	$01A0
1	4	289	$0121	417	$01A1
2	1	290	$0122	418	$01A2
2	2	291	$0123	419	$01A3
2	3	292	$0124	420	$01A4
2	4	293	$0125	421	$01A5
3	1	294	$0126	422	$01A6
3	2	295	$0127	423	$01A7
3	3	296	$0128	424	$01A8
3	4	297	$0129	425	$01A9
4	1	298	$012A	426	$01AA
4	2	299	$012B	427	$01AB
4	3	300	$012C	428	$01AC
4	4	301	$012D	429	$01AD
5	1	302	$012E	430	$01AE
5	2	303	$012F	431	$01AF
5	3	304	$0130	432	$01B0
5	4	305	$0131	433	$01B1
6	1	306	$0132	434	$01B2
6	2	307	$0133	435	$01B3
6	3	308	$0134	436	$01B4
6	4	309	$0135	437	$01B5
7	1	310	$0136	438	$01B6
7	2	311	$0137	439	$01B7
7	3	312	$0138	440	$01B8
7	4	313	$0139	441	$01B9
8	1	314	$013A	442	$01BA
8	2	315	$013B	443	$01BB
8	3	316	$013C	444	$01BC
8	4	317	$013D	445	$01BD
9	1	318	$013E	446	$01BE
9	2	319	$013F	447	$01BF
9	3	320	$0140	448	$01C0
9	4	321	$0141	449	$01C1
10	1	322	$0142	450	$01C2
10	2	323	$0143	451	$01C3
10	3	324	$0144	452	$01C4
10	4	325	$0145	453	$01C5
11	1	326	$0146	454	$01C6
11	2	327	$0147	455	$01C7
11	3	328	$0148	456	$01C8
11	4	329	$0149	457	$01C9
12	1	330	$014A	458	$01CA
12	2	331	$014B	459	$01CB
12	3	332	$014C	460	$01CC
12	4	333	$014D	461	$01CD
13	1	334	$014E	462	$01CE
13	2	335	$014F	463	$01CF
13	3	336	$0150	464	$01D0
13	4	337	$0151	465	$01D1
14	1	338	$0152	466	$01D2
14	2	339	$0153	467	$01D3
14	3	340	$0154	468	$01D4
14	4	341	$0155	469	$01D5
15	1	342	$0156	470	$01D6
15	2	343	$0157	471	$01D7
15	3	344	$0158	472	$01D8
15	4	345	$0159	473	$01D9
16	1	346	$015A	474	$01DA
16	2	347	$015B	475	$01DB
16	3	348	$015C	476	$01DC
16	4	349	$015D	477	$01DD

Appendix A
CAN Identifier with
CAN Bus Controller

Node Number	Module Number	Digital Inputs		Digital Outputs	
17	1	350	$015E	478	$01DE
17	2	351	$015F	479	$01DF
17	3	352	$0160	480	$01E0
17	4	353	$0161	481	$01E1
18	1	354	$0162	482	$01E2
18	2	355	$0163	483	$01E3
18	3	356	$0164	484	$01E4
18	4	357	$0165	485	$01E5
19	1	358	$0166	486	$01E6
19	2	359	$0167	487	$01E7
19	3	360	$0168	488	$01E8
19	4	361	$0169	489	$01E9
20	1	362	$016A	490	$01EA
20	2	363	$016B	491	$01EB
20	3	364	$016C	492	$01EC
20	4	365	$016D	493	$01ED
21	1	366	$016E	494	$01EE
21	2	367	$016F	495	$01EF
21	3	368	$0170	496	$01F0
21	4	369	$0171	497	$01F1
22	1	370	$0172	498	$01F2
22	2	371	$0173	499	$01F3
22	3	372	$0174	500	$01F4
22	4	373	$0175	501	$01F5
23	1	374	$0176	502	$01F6
23	2	375	$0177	503	$01F7
23	3	376	$0178	504	$01F8
23	4	377	$0179	505	$01F9
24	1	378	$017A	506	$01FA
24	2	379	$017B	507	$01FB
24	3	380	$017C	508	$01FC
24	4	381	$017D	509	$01FD
25	1	382	$017E	510	$01FE
25	2	383	$017F	511	$01FF
25	3	384	$0180	512	$0200
25	4	385	$0181	513	$0201
26	1	386	$0182	514	$0202
26	2	387	$0183	515	$0203
26	3	388	$0184	516	$0204
26	4	389	$0185	517	$0205
27	1	390	$0186	518	$0206
27	2	391	$0187	519	$0207
27	3	392	$0188	520	$0208
27	4	393	$0189	521	$0209
28	1	394	$018A	522	$020A
28	2	395	$018B	523	$020B
28	3	396	$018C	524	$020C
28	4	397	$018D	525	$020D
29	1	398	$018E	526	$020E
29	2	399	$018F	527	$020F
29	3	400	$0190	528	$0210
29	4	401	$0191	529	$0211
30	1	402	$0192	530	$0212
30	2	403	$0193	531	$0213
30	3	404	$0194	532	$0214
30	4	405	$0195	533	$0215
31	1	406	$0196	534	$0216
31	2	407	$0197	535	$0217
31	3	408	$0198	536	$0218
31	4	409	$0199	537	$0219
32	1	410	$019A	538	$021A
32	2	411	$019B	539	$021B
32	3	412	$019C	540	$021C
32	4	413	$019D	541	$021D

Node Number	Module Number	Slot Number	Analog Inputs		Analog Outputs	
001	001	1	542	$021E	1054	$041E
001	001	2	543	$021F	1055	$041F
001	001	3	544	$0220	1056	$0420
001	001	4	545	$0221	1057	$0421
001	002	1	546	$0222	1058	$0422
001	002	2	547	$0223	1059	$0423
001	002	3	548	$0224	1060	$0424
001	002	4	549	$0225	1061	$0425
001	003	1	550	$0226	1062	$0426
001	003	2	551	$0227	1063	$0427
001	003	3	552	$0228	1064	$0428
001	003	4	553	$0229	1065	$0429
001	004	1	554	$022A	1066	$042A
001	004	2	555	$022B	1067	$042B
001	004	3	556	$022C	1068	$042C
001	004	4	557	$022D	1069	$042D
002	001	1	558	$022E	1070	$042E
002	001	2	559	$022F	1071	$042F
002	001	3	560	$0230	1072	$0430
002	001	4	561	$0231	1073	$0431
002	002	1	562	$0232	1074	$0432
002	002	2	563	$0233	1075	$0433
002	002	3	564	$0234	1076	$0434
002	002	4	565	$0235	1077	$0435
002	003	1	566	$0236	1078	$0436
002	003	2	567	$0237	1079	$0437
002	003	3	568	$0238	1080	$0438
002	003	4	569	$0239	1081	$0439
002	004	1	570	$023A	1082	$043A
002	004	2	571	$023B	1083	$043B
002	004	3	572	$023C	1084	$043C
002	004	4	573	$023D	1085	$043D
003	001	1	574	$023E	1086	$043E
003	001	2	575	$023F	1087	$043F
003	001	3	576	$0240	1088	$0440
003	001	4	577	$0241	1089	$0441
003	002	1	578	$0242	1090	$0442
003	002	2	579	$0243	1091	$0443
003	002	3	580	$0244	1092	$0444
003	002	4	581	$0245	1093	$0445
003	003	1	582	$0246	1094	$0446
003	003	2	583	$0247	1095	$0447
003	003	3	584	$0248	1096	$0448
003	003	4	585	$0249	1097	$0449
003	004	1	586	$024A	1098	$044A
003	004	2	587	$024B	1099	$044B
003	004	3	588	$024C	1100	$044C
003	004	4	589	$024D	1101	$044D
004	001	1	590	$024E	1102	$044E
004	001	2	591	$024F	1103	$044F
004	001	3	592	$0250	1104	$0450
004	001	4	593	$0251	1105	$0451
004	002	1	594	$0252	1106	$0452
004	002	2	595	$0253	1107	$0453
004	002	3	596	$0254	1108	$0454
004	002	4	597	$0255	1109	$0455
004	003	1	598	$0256	1110	$0456
004	003	2	599	$0257	1111	$0457
004	003	3	600	$0258	1112	$0458
004	003	4	601	$0259	1113	$0459
004	004	1	602	$025A	1114	$045A
004	004	2	603	$025B	1115	$045B
004	004	3	604	$025C	1116	$045C
004	004	4	605	$025D	1117	$045D

Node Number	Module Number	Slot Number	Analog Inputs		Analog Outputs	
005	001	1	606	$025E	1118	$045E
005	001	2	607	$025F	1119	$045F
005	001	3	608	$0260	1120	$0460
005	001	4	609	$0261	1121	$0461
005	002	1	610	$0262	1122	$0462
005	002	2	611	$0263	1123	$0463
005	002	3	612	$0264	1124	$0464
005	002	4	613	$0265	1125	$0465
005	003	1	614	$0266	1126	$0466
005	003	2	615	$0267	1127	$0467
005	003	3	616	$0268	1128	$0468
005	003	4	617	$0269	1129	$0469
005	004	1	618	$026A	1130	$046A
005	004	2	619	$026B	1131	$046B
005	004	3	620	$026C	1132	$046C
005	004	4	621	$026D	1133	$046D
006	001	1	622	$026E	1134	$046E
006	001	2	623	$026F	1135	$046F
006	001	3	624	$0270	1136	$0470
006	001	4	625	$0271	1137	$0471
006	002	1	626	$0272	1138	$0472
006	002	2	627	$0273	1139	$0473
006	002	3	628	$0274	1140	$0474
006	002	4	629	$0275	1141	$0475
006	003	1	630	$0276	1142	$0476
006	003	2	631	$0277	1143	$0477
006	003	3	632	$0278	1144	$0478
006	003	4	633	$0279	1145	$0479
006	004	1	634	$027A	1146	$047A
006	004	2	635	$027B	1147	$047B
006	004	3	636	$027C	1148	$047C
006	004	4	637	$027D	1149	$047D
007	001	1	638	$027E	1150	$047E
007	001	2	639	$027F	1151	$047F
007	001	3	640	$0280	1152	$0480
007	001	4	641	$0281	1153	$0481
007	002	1	642	$0282	1154	$0482
007	002	2	643	$0283	1155	$0483
007	002	3	644	$0284	1156	$0484
007	002	4	645	$0285	1157	$0485
007	003	1	646	$0286	1158	$0486
007	003	2	647	$0287	1159	$0487
007	003	3	648	$0288	1160	$0488
007	003	4	649	$0289	1161	$0489
007	004	1	650	$028A	1162	$048A
007	004	2	651	$028B	1163	$048B
007	004	3	652	$028C	1164	$048C
007	004	4	653	$028D	1165	$048D
008	001	1	654	$028E	1166	$048E
008	001	2	655	$028F	1167	$048F
008	001	3	656	$0290	1168	$0490
008	001	4	657	$0291	1169	$0491
008	002	1	658	$0292	1170	$0492
008	002	2	659	$0293	1171	$0493
008	002	3	660	$0294	1172	$0494
008	002	4	661	$0295	1173	$0495
008	003	1	662	$0296	1174	$0496
008	003	2	663	$0297	1175	$0497
008	003	3	664	$0298	1176	$0498
008	003	4	665	$0299	1177	$0499
008	004	1	666	$029A	1178	$049A
008	004	2	667	$029B	1179	$049B
008	004	3	668	$029C	1180	$049C
008	004	4	669	$029D	1181	$049D

Node Number	Module Number	Slot Number	Analog Inputs		Analog Outputs	
009	001	1	670	$029E	1182	$049E
009	001	2	671	$029F	1183	$049F
009	001	3	672	$02A0	1184	$04A0
009	001	4	673	$02A1	1185	$04A1
009	002	1	674	$02A2	1186	$04A2
009	002	2	675	$02A3	1187	$04A3
009	002	3	676	$02A4	1188	$04A4
009	002	4	677	$02A5	1189	$04A5
009	003	1	678	$02A6	1190	$04A6
009	003	2	679	$02A7	1191	$04A7
009	003	3	680	$02A8	1192	$04A8
009	003	4	681	$02A9	1193	$04A9
009	004	1	682	$02AA	1194	$04AA
009	004	2	683	$02AB	1195	$04AB
009	004	3	684	$02AC	1196	$04AC
009	004	4	685	$02AD	1197	$04AD
010	001	1	686	$02AE	1198	$04AE
010	001	2	687	$02AF	1199	$04AF
010	001	3	688	$02B0	1200	$04B0
010	001	4	689	$02B1	1201	$04B1
010	002	1	690	$02B2	1202	$04B2
010	002	2	691	$02B3	1203	$04B3
010	002	3	692	$02B4	1204	$04B4
010	002	4	693	$02B5	1205	$04B5
010	003	1	694	$02B6	1206	$04B6
010	003	2	695	$02B7	1207	$04B7
010	003	3	696	$02B8	1208	$04B8
010	003	4	697	$02B9	1209	$04B9
010	004	1	698	$02BA	1210	$04BA
010	004	2	699	$02BB	1211	$04BB
010	004	3	700	$02BC	1212	$04BC
010	004	4	701	$02BD	1213	$04BD
011	001	1	702	$02BE	1214	$04BE
011	001	2	703	$02BF	1215	$04BF
011	001	3	704	$02C0	1216	$04C0
011	001	4	705	$02C1	1217	$04C1
011	002	1	706	$02C2	1218	$04C2
011	002	2	707	$02C3	1219	$04C3
011	002	3	708	$02C4	1220	$04C4
011	002	4	709	$02C5	1221	$04C5
011	003	1	710	$02C6	1222	$04C6
011	003	2	711	$02C7	1223	$04C7
011	003	3	712	$02C8	1224	$04C8
011	003	4	713	$02C9	1225	$04C9
011	004	1	714	$02CA	1226	$04CA
011	004	2	715	$02CB	1227	$04CB
011	004	3	716	$02CC	1228	$04CC
011	004	4	717	$02CD	1229	$04CD
012	001	1	718	$02CE	1230	$04CE
012	001	2	719	$02CF	1231	$04CF
012	001	3	720	$02D0	1232	$04D0
012	001	4	721	$02D1	1233	$04D1
012	002	1	722	$02D2	1234	$04D2
012	002	2	723	$02D3	1235	$04D3
012	002	3	724	$02D4	1236	$04D4
012	002	4	725	$02D5	1237	$04D5
012	003	1	726	$02D6	1238	$04D6
012	003	2	727	$02D7	1239	$04D7
012	003	3	728	$02D8	1240	$04D8
012	003	4	729	$02D9	1241	$04D9
012	004	1	730	$02DA	1242	$04DA
012	004	2	731	$02DB	1243	$04DB
012	004	3	732	$02DC	1244	$04DC
012	004	4	733	$02DD	1245	$04DD

Appendix A
CAN Identifier with
CAN Bus Controller

Node Number	Module Number	Slot Number	Analog Inputs		Analog Outputs	
013	001	1	734	$02DE	1246	$04DE
013	001	2	735	$02DF	1247	$04DF
013	001	3	736	$02E0	1248	$04E0
013	001	4	737	$02E1	1249	$04E1
013	002	1	738	$02E2	1250	$04E2
013	002	2	739	$02E3	1251	$04E3
013	002	3	740	$02E4	1252	$04E4
013	002	4	741	$02E5	1253	$04E5
013	003	1	742	$02E6	1254	$04E6
013	003	2	743	$02E7	1255	$04E7
013	003	3	744	$02E8	1256	$04E8
013	003	4	745	$02E9	1257	$04E9
013	004	1	746	$02EA	1258	$04EA
013	004	2	747	$02EB	1259	$04EB
013	004	3	748	$02EC	1260	$04EC
013	004	4	749	$02ED	1261	$04ED
014	001	1	750	$02EE	1262	$04EE
014	001	2	751	$02EF	1263	$04EF
014	001	3	752	$02F0	1264	$04F0
014	001	4	753	$02F1	1265	$04F1
014	002	1	754	$02F2	1266	$04F2
014	002	2	755	$02F3	1267	$04F3
014	002	3	756	$02F4	1268	$04F4
014	002	4	757	$02F5	1269	$04F5
014	003	1	758	$02F6	1270	$04F6
014	003	2	759	$02F7	1271	$04F7
014	003	3	760	$02F8	1272	$04F8
014	003	4	761	$02F9	1273	$04F9
014	004	1	762	$02FA	1274	$04FA
014	004	2	763	$02FB	1275	$04FB
014	004	3	764	$02FC	1276	$04FC
014	004	4	765	$02FD	1277	$04FD
015	001	1	766	$02FE	1278	$04FE
015	001	2	767	$02FF	1279	$04FF
015	001	3	768	$0300	1280	$0500
015	001	4	769	$0301	1281	$0501
015	002	1	770	$0302	1282	$0502
015	002	2	771	$0303	1283	$0503
015	002	3	772	$0304	1284	$0504
015	002	4	773	$0305	1285	$0505
015	003	1	774	$0306	1286	$0506
015	003	2	775	$0307	1287	$0507
015	003	3	776	$0308	1288	$0508
015	003	4	777	$0309	1289	$0509
015	004	1	778	$030A	1290	$050A
015	004	2	779	$030B	1291	$050B
015	004	3	780	$030C	1292	$050C
015	004	4	781	$030D	1293	$050D
016	001	1	782	$030E	1294	$050E
016	001	2	783	$030F	1295	$050F
016	001	3	784	$0310	1296	$0510
016	001	4	785	$0311	1297	$0511
016	002	1	786	$0312	1298	$0512
016	002	2	787	$0313	1299	$0513
016	002	3	788	$0314	1300	$0514
016	002	4	789	$0315	1301	$0515
016	003	1	790	$0316	1302	$0516
016	003	2	791	$0317	1303	$0517
016	003	3	792	$0318	1304	$0518
016	003	4	793	$0319	1305	$0519
016	004	1	794	$031A	1306	$051A
016	004	2	795	$031B	1307	$051B
016	004	3	796	$031C	1308	$051C
016	004	4	797	$031D	1309	$051D

Node Number	Module Number	Slot Number	Analog Inputs		Analog Outputs	
017	001	1	798	$031E	1310	$051E
017	001	2	799	$031F	1311	$051F
017	001	3	800	$0320	1312	$0520
017	001	4	801	$0321	1313	$0521
017	002	1	802	$0322	1314	$0522
017	002	2	803	$0323	1315	$0523
017	002	3	804	$0324	1316	$0524
017	002	4	805	$0325	1317	$0525
017	003	1	806	$0326	1318	$0526
017	003	2	807	$0327	1319	$0527
017	003	3	808	$0328	1320	$0528
017	003	4	809	$0329	1321	$0529
017	004	1	810	$032A	1322	$052A
017	004	2	811	$032B	1323	$052B
017	004	3	812	$032C	1324	$052C
017	004	4	813	$032D	1325	$052D
018	001	1	814	$032E	1326	$052E
018	001	2	815	$032F	1327	$052F
018	001	3	816	$0330	1328	$0530
018	001	4	817	$0331	1329	$0531
018	002	1	818	$0332	1330	$0532
018	002	2	819	$0333	1331	$0533
018	002	3	820	$0334	1332	$0534
018	002	4	821	$0335	1333	$0535
018	003	1	822	$0336	1334	$0536
018	003	2	823	$0337	1335	$0537
018	003	3	824	$0338	1336	$0538
018	003	4	825	$0339	1337	$0539
018	004	1	826	$033A	1338	$053A
018	004	2	827	$033B	1339	$053B
018	004	3	828	$033C	1340	$053C
018	004	4	829	$033D	1341	$053D
019	001	1	830	$033E	1342	$053E
019	001	2	831	$033F	1343	$053F
019	001	3	832	$0340	1344	$0540
019	001	4	833	$0341	1345	$0541
019	002	1	834	$0342	1346	$0542
019	002	2	835	$0343	1347	$0543
019	002	3	836	$0344	1348	$0544
019	002	4	837	$0345	1349	$0545
019	003	1	838	$0346	1350	$0546
019	003	2	839	$0347	1351	$0547
019	003	3	840	$0348	1352	$0548
019	003	4	841	$0349	1353	$0549
019	004	1	842	$034A	1354	$054A
019	004	2	843	$034B	1355	$054B
019	004	3	844	$034C	1356	$054C
019	004	4	845	$034D	1357	$054D
020	001	1	846	$034E	1358	$054E
020	001	2	847	$034F	1359	$054F
020	001	3	848	$0350	1360	$0550
020	001	4	849	$0351	1361	$0551
020	002	1	850	$0352	1362	$0552
020	002	2	851	$0353	1363	$0553
020	002	3	852	$0354	1364	$0554
020	002	4	853	$0355	1365	$0555
020	003	1	854	$0356	1366	$0556
020	003	2	855	$0357	1367	$0557
020	003	3	856	$0358	1368	$0558
020	003	4	857	$0359	1369	$0559
020	004	1	858	$035A	1370	$055A
020	004	2	859	$035B	1371	$055B
020	004	3	860	$035C	1372	$055C
020	004	4	861	$035D	1373	$055D

Appendix A
CAN Identifier with
CAN Bus Controller

Node Number	Module Number	Slot Number	Analog Inputs		Analog Outputs	
021	001	1	862	$035E	1374	$055E
021	001	2	863	$035F	1375	$055F
021	001	3	864	$0360	1376	$0560
021	001	4	865	$0361	1377	$0561
021	002	1	866	$0362	1378	$0562
021	002	2	867	$0363	1379	$0563
021	002	3	868	$0364	1380	$0564
021	002	4	869	$0365	1381	$0565
021	003	1	870	$0366	1382	$0566
021	003	2	871	$0367	1383	$0567
021	003	3	872	$0368	1384	$0568
021	003	4	873	$0369	1385	$0569
021	004	1	874	$036A	1386	$056A
021	004	2	875	$036B	1387	$056B
021	004	3	876	$036C	1388	$056C
021	004	4	877	$036D	1389	$056D
022	001	1	878	$036E	1390	$056E
022	001	2	879	$036F	1391	$056F
022	001	3	880	$0370	1392	$0570
022	001	4	881	$0371	1393	$0571
022	002	1	882	$0372	1394	$0572
022	002	2	883	$0373	1395	$0573
022	002	3	884	$0374	1396	$0574
022	002	4	885	$0375	1397	$0575
022	003	1	886	$0376	1398	$0576
022	003	2	887	$0377	1399	$0577
022	003	3	888	$0378	1400	$0578
022	003	4	889	$0379	1401	$0579
022	004	1	890	$037A	1402	$057A
022	004	2	891	$037B	1403	$057B
022	004	3	892	$037C	1404	$057C
022	004	4	893	$037D	1405	$057D
023	001	1	894	$037E	1406	$057E
023	001	2	895	$037F	1407	$057F
023	001	3	896	$0380	1408	$0580
023	001	4	897	$0381	1409	$0581
023	002	1	898	$0382	1410	$0582
023	002	2	899	$0383	1411	$0583
023	002	3	900	$0384	1412	$0584
023	002	4	901	$0385	1413	$0585
023	003	1	902	$0386	1414	$0586
023	003	2	903	$0387	1415	$0587
023	003	3	904	$0388	1416	$0588
023	003	4	905	$0389	1417	$0589
023	004	1	906	$038A	1418	$058A
023	004	2	907	$038B	1419	$058B
023	004	3	908	$038C	1420	$058C
023	004	4	909	$038D	1421	$058D
024	001	1	910	$038E	1422	$058E
024	001	2	911	$038F	1423	$058F
024	001	3	912	$0390	1424	$0590
024	001	4	913	$0391	1425	$0591
024	002	1	914	$0392	1426	$0592
024	002	2	915	$0393	1427	$0593
024	002	3	916	$0394	1428	$0594
024	002	4	917	$0395	1429	$0595
024	003	1	918	$0396	1430	$0596
024	003	2	919	$0397	1431	$0597
024	003	3	920	$0398	1432	$0598
024	003	4	921	$0399	1433	$0599
024	004	1	922	$039A	1434	$059A
024	004	2	923	$039B	1435	$059B
024	004	3	924	$039C	1436	$059C
024	004	4	925	$039D	1437	$059D

Node Number	Module Number	Slot Number	Analog Inputs		Analog Outputs	
025	001	1	926	$039E	1438	$059E
025	001	2	927	$039F	1439	$059F
025	001	3	928	$03A0	1440	$05A0
025	001	4	929	$03A1	1441	$05A1
025	002	1	930	$03A2	1442	$05A2
025	002	2	931	$03A3	1443	$05A3
025	002	3	932	$03A4	1444	$05A4
025	002	4	933	$03A5	1445	$05A5
025	003	1	934	$03A6	1446	$05A6
025	003	2	935	$03A7	1447	$05A7
025	003	3	936	$03A8	1448	$05A8
025	003	4	937	$03A9	1449	$05A9
025	004	1	938	$03AA	1450	$05AA
025	004	2	939	$03AB	1451	$05AB
025	004	3	940	$03AC	1452	$05AC
025	004	4	941	$03AD	1453	$05AD
026	001	1	942	$03AE	1454	$05AE
026	001	2	943	$03AF	1455	$05AF
026	001	3	944	$03B0	1456	$05B0
026	001	4	945	$03B1	1457	$05B1
026	002	1	946	$03B2	1458	$05B2
026	002	2	947	$03B3	1459	$05B3
026	002	3	948	$03B4	1460	$05B4
026	002	4	949	$03B5	1461	$05B5
026	003	1	950	$03B6	1462	$05B6
026	003	2	951	$03B7	1463	$05B7
026	003	3	952	$03B8	1464	$05B8
026	003	4	953	$03B9	1465	$05B9
026	004	1	954	$03BA	1466	$05BA
026	004	2	955	$03BB	1467	$05BB
026	004	3	956	$03BC	1468	$05BC
026	004	4	957	$03BD	1469	$05BD
027	001	1	958	$03BE	1470	$05BE
027	001	2	959	$03BF	1471	$05BF
027	001	3	960	$03C0	1472	$05C0
027	001	4	961	$03C1	1473	$05C1
027	002	1	962	$03C2	1474	$05C2
027	002	2	963	$03C3	1475	$05C3
027	002	3	964	$03C4	1476	$05C4
027	002	4	965	$03C5	1477	$05C5
027	003	1	966	$03C6	1478	$05C6
027	003	2	967	$03C7	1479	$05C7
027	003	3	968	$03C8	1480	$05C8
027	003	4	969	$03C9	1481	$05C9
027	004	1	970	$03CA	1482	$05CA
027	004	2	971	$03CB	1483	$05CB
027	004	3	972	$03CC	1484	$05CC
027	004	4	973	$03CD	1485	$05CD
028	001	1	974	$03CE	1486	$05CE
028	001	2	975	$03CF	1487	$05CF
028	001	3	976	$03D0	1488	$05D0
028	001	4	977	$03D1	1489	$05D1
028	002	1	978	$03D2	1490	$05D2
028	002	2	979	$03D3	1491	$05D3
028	002	3	980	$03D4	1492	$05D4
028	002	4	981	$03D5	1493	$05D5
028	003	1	982	$03D6	1494	$05D6
028	003	2	983	$03D7	1495	$05D7
028	003	3	984	$03D8	1496	$05D8
028	003	4	985	$03D9	1497	$05D9
028	004	1	986	$03DA	1498	$05DA
028	004	2	987	$03DB	1499	$05DB
028	004	3	988	$03DC	1500	$05DC
028	004	4	989	$03DD	1501	$05DD

Appendix A
CAN Identifier with
CAN Bus Controller

Node Number	Module Number	Slot Number	Analog Inputs		Analog Outputs	
029	001	1	990	$03DE	1502	$05DE
029	001	2	991	$03DF	1503	$05DF
029	001	3	992	$03E0	1504	$05E0
029	001	4	993	$03E1	1505	$05E1
029	002	1	994	$03E2	1506	$05E2
029	002	2	995	$03E3	1507	$05E3
029	002	3	996	$03E4	1508	$05E4
029	002	4	997	$03E5	1509	$05E5
029	003	1	998	$03E6	1510	$05E6
029	003	2	999	$03E7	1511	$05E7
029	003	3	1000	$03E8	1512	$05E8
029	003	4	1001	$03E9	1513	$05E9
029	004	1	1002	$03EA	1514	$05EA
029	004	2	1003	$03EB	1515	$05EB
029	004	3	1004	$03EC	1516	$05EC
029	004	4	1005	$03ED	1517	$05ED
030	001	1	1006	$03EE	1518	$05EE
030	001	2	1007	$03EF	1519	$05EF
030	001	3	1008	$03F0	1520	$05F0
030	001	4	1009	$03F1	1521	$05F1
030	002	1	1010	$03F2	1522	$05F2
030	002	2	1011	$03F3	1523	$05F3
030	002	3	1012	$03F4	1524	$05F4
030	002	4	1013	$03F5	1525	$05F5
030	003	1	1014	$03F6	1526	$05F6
030	003	2	1015	$03F7	1527	$05F7
030	003	3	1016	$03F8	1528	$05F8
030	003	4	1017	$03F9	1529	$05F9
030	004	1	1018	$03FA	1530	$05FA
030	004	2	1019	$03FB	1531	$05FB
030	004	3	1020	$03FC	1532	$05FC
030	004	4	1021	$03FD	1533	$05FD
031	001	1	1022	$03FE	1534	$05FE
031	001	2	1023	$03FF	1535	$05FF
031	001	3	1024	$0400	1536	$0600
031	001	4	1025	$0401	1537	$0601
031	002	1	1026	$0402	1538	$0602
031	002	2	1027	$0403	1539	$0603
031	002	3	1028	$0404	1540	$0604
031	002	4	1029	$0405	1541	$0605
031	003	1	1030	$0406	1542	$0606
031	003	2	1031	$0407	1543	$0607
031	003	3	1032	$0408	1544	$0608
031	003	4	1033	$0409	1545	$0609
031	004	1	1034	$040A	1546	$060A
031	004	2	1035	$040B	1547	$060B
031	004	3	1036	$040C	1548	$060C
031	004	4	1037	$040D	1549	$060D
032	001	1	1038	$040E	1550	$060E
032	001	2	1039	$040F	1551	$060F
032	001	3	1040	$0410	1552	$0610
032	001	4	1041	$0411	1553	$0611
032	002	1	1042	$0412	1554	$0612
032	002	2	1043	$0413	1555	$0613
032	002	3	1044	$0414	1556	$0614
032	002	4	1045	$0415	1557	$0615
032	003	1	1046	$0416	1558	$0616
032	003	2	1047	$0417	1559	$0617
032	003	3	1048	$0418	1560	$0618
032	003	4	1049	$0419	1561	$0619
032	004	1	1050	$041A	1562	$061A
032	004	2	1051	$041B	1563	$061B
032	004	3	1052	$041C	1564	$061C
032	004	4	1053	$041D	1565	$061D

3 ALARM MESSAGES, COMMAND REQUESTS AND RESPONSES

Node Number	Alarm Messages		Command Requests		Command Responses	
1	222	00DE	1566	061E	1630	065E
2	223	00DF	1567	061F	1631	065F
3	224	00E0	1568	0620	1632	0660
4	225	00E1	1569	0621	1633	0661
5	226	00E2	1570	0622	1634	0662
6	227	00E3	1571	0623	1635	0663
7	228	00E4	1572	0624	1636	0664
8	229	00E5	1573	0625	1637	0665
9	230	00E6	1574	0626	1638	0666
10	231	00E7	1575	0627	1639	0667
11	232	00E8	1576	0628	1640	0668
12	233	00E9	1577	0629	1641	0669
13	234	00EA	1578	062A	1642	066A
14	235	00EB	1579	062B	1643	066B
15	236	00EC	1580	062C	1644	066C
16	237	00ED	1581	062D	1645	066D
17	238	00EE	1582	062E	1646	066E
18	239	00EF	1583	062F	1647	066F
19	240	00F0	1584	0630	1648	0670
20	241	00F1	1585	0631	1649	0671
21	242	00F2	1586	0632	1650	0672
22	243	00F3	1587	0633	1651	0673
23	244	00F4	1588	0634	1652	0674
24	245	00F5	1589	0635	1653	0675
25	246	00F6	1590	0636	1654	0676
26	247	00F7	1591	0637	1655	0677
27	248	00F8	1592	0638	1656	0678
28	249	00F9	1593	0639	1657	0679
29	250	00FA	1594	063A	1658	067A
30	251	00FB	1595	063B	1659	067B
31	252	00FC	1596	063C	1660	067C
32	253	00FD	1597	063D	1661	067D
33			1598	063E	1662	067E
34			1599	063F	1663	067F
35			1600	0640	1664	0680
36			1601	0641	1665	0681
37			1602	0642	1666	0682
38			1603	0643	1667	0683
39			1604	0644	1668	0684
40			1605	0645	1669	0685
41			1606	0646	1670	0686
42			1607	0647	1671	0687
43			1608	0648	1672	0688
44			1609	0649	1673	0689
45			1610	064A	1674	068A
46			1611	064B	1675	068B
47			1612	064C	1676	068C
48			1613	064D	1677	068D
49			1614	064E	1678	068E
50			1615	064F	1679	068F
51			1616	0650	1680	0690
52			1617	0651	1681	0691
53			1618	0652	1682	0692
54			1619	0653	1683	0693
55			1620	0654	1684	0694
56			1621	0655	1685	0695
57			1622	0656	1686	0696
58			1623	0657	1687	0697
59			1624	0658	1688	0698
60			1625	0659	1689	0699
61			1626	065A	1690	069A
62			1627	065B	1691	069B
63			1628	065C	1692	069C

APPENDIX B
ERROR MESSAGES
CAN BUS CONTROLLER

Appendix B
Error Messages
CAN Bus Controller

1 ERROR MESSAGES

The CAN bus controller can give out error messages in various ways:

- Alarm Message - Every bus controller can send alarm messages. It uses its assigned ID for this purpose (see appendix: "CAN Identifier" for fixed allocations).

  An alarm message always occurs without being requested.

- Command Response - As a rule, these error messages are responses to a command request from a client. A bus controller characterizes a command response as an error message by setting bit 7 of the code to 1.

  However, a bus controller can also automatically send error messages which are not requested using a command response. In this case, the value $FF is entered as the code.

- An error message can also be requested by a client directly from the bus controller using a command request (see Chapter 5: "CAN Bus Controller Functions" in Section "Commands").

You should carefully plan the evaluation of error messages and take the following into consideration in the network right from the start:

- For alarm messages, it makes sense if one station evaluates the errors centrally. If an NMT master is available, it takes over this task.

- When command responses are used to send error messages, error handling should be handled by the client which sent the command request.

- Broadcasting error messages using command responses (code => $FF) can also be handled centrally by one station (NMT master).

Error messages consist of 4 bytes - the error code and a supplementary code, which can also contain information about the cause of the error (node number, module number).

The following table contains the possible error messages for a CAN bus controller:

Error Code (hex.)	Supplementary Code (hex.)	Description
2020	4300	Total power for the module exceeds the power supply on the EX270/EX470/EX770
2032	13m0	Output level incorrect – short circuit (DOUT)
3010	40m0	Voltage monitoring has found an error
5000	c30ms	Invalid channel or value
5000	c31ms	Measurement range exceeded (pos. direction)
5000	c32ms	Measurement range exceeded (neg. direction)
5000	c33ms	Open connection or sensor open
5000	c34ms	Converter error
5000	c35ms	Screw-in module removed
5000	c36ms	Screw-in module inactive
5000	c37ms	Screw-in module communication error
5000	c38ms	Screw-in module type changed
5000	41m0	Error flag for digital input module set
5000	42mc	Error flag for digital output module set
F000	5000	Programming configuration memory ended without error (CMDRD)
F000	5100	Faulty or incorrect parameter in command request (CMDWR)
F000	5200	No configuration memory available
F000	5300	Error when writing to configuration memory
F000	5400	Configuration memory is being written to
FFFF	11m0	Idle time has passed (DOUT)
FFFF	12m0	Output set, even though no output is defined
FFFF	21ms	Idle time has passed (AOUT)

Abbr.	Description
c	Channel number for multi-channel modules 0.......... module without extended status 1 - 8..... module with extended status channel number where error occurred
m	Adapter module number
s	Screw-in module number

1.1 SUPPLEMENTARY CODE c30ms - c38ms

The 16 bit value is has the following structure:

	Bit	Description
	14 - 15	0 - 3.... Channel number -1
	12 - 13	3
	8 - 11	0 - 8.... Error number
	4 - 7	1 - 4.... Adapter module number
	0 - 3	1 - 4.... Screw-in module number

```
      1 1
15          8 7          0
```
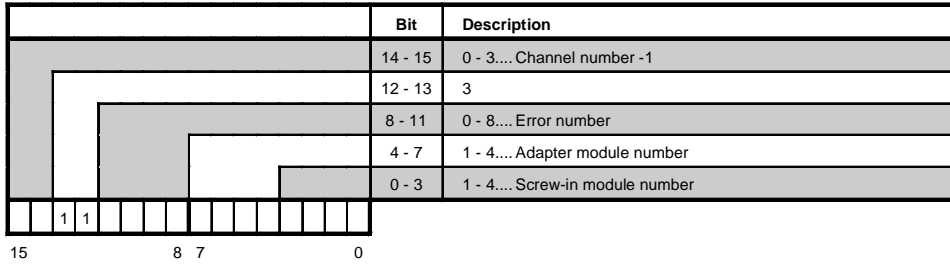
1.2 SENDING ERROR MESSAGES

Error messages are sent in different ways. Take this into consideration during evaluation.

- Sent as **Alarm Message** - error and supplementary codes are sent as follows:

Byte 0	Byte 1	Byte 2	Byte 3
Error Code Low Byte	Error Code High Byte	Supl. Code Low Byte	Supl. Code High Byte

- Sent as **Command Response** - error and supplementary codes are sent as follows:

Byte 4	Byte 5	Byte 6	Byte 7
Supl. Code High Byte	Supl. Code Low Byte	Error Code High Byte	Error Code Low Byte

Bit 7 in Byte 0 is set in the command response.

INDEX

INDEX

Index

C

Index

T

V

W

MODEL NUMBER INDEX

E

RELEVANT CONVERSIONS

Some of the values in this manual and in other documentation that you may have contain values which are only given in metric. Use these formulas and charts to help with any conversion problems that you may have.

°F	°C
-40	-40
-20	-28.89
-10	-23.33
-5	-20.56
0	-17.78
5	-15.00
10	-12.22
15	-9.44
20	-6.67
25	-3.89
30	-1.11
35	1.67
40	4.44
45	7.22
50	10.00
55	12.78
60	15.56
65	18.33
70	21.11
75	23.89
80	26.67
85	29.44
90	32.22
95	35.00
100	37.78
105	40.56
110	43.33
115	46.11
120	48.89
125	51.67
130	54.44
135	57.22
140	60.00
145	62.78
150	65.56

°C	°F
-40	-40
-35	-31
-30	-22
-25	-13
-20	-4
-15	5
-10	14
-5	23
0	32
5	41
10	50
15	59
20	68
25	77
30	86
35	95
40	104
45	113
50	122
55	131
60	140
65	149
70	158
75	167
80	176
85	185
90	194

English Units	Metric Units
1 inch	25.4 millimeters
	2.54 centimeters
1 foot	30.48 centimeters
	3.048 decimeters
	0.3048 meter
1 yard	0.9144 meter
0.03937 inch	1 millimeter
0.3937 inch	1 centimeter
3.937 inches	1 decimeter
39.37 inches 3.2808 feet 1.0936 yards	1 meter
3280.8 feet 1093.6 yards 0.62137 mile	1 kilometer

English Units	Metric Units
1 pound	0.45359 kilogram
1 ounce	28.350 grams
1 short ton	907.18 kilograms
	0.90718 metric tons
1 long ton	1016.0 kilograms
	1.0160 metric tons

$$5/9 \times (°F - 32) = °C$$
$$(9/5 \times °C) + 32 = °F$$