ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Find the *DMS / Nautibus Electronic VME-PROF-S* at our website: ***Click HERE***

VME-PROFI

Technical Manual

Drafted by:	Reiner Christophersen
Date:	22.11.1995
Revision:	01.10.1996
ID of Document:	942.1386.51

All information in this handbook is as contemporary, accurate, and up to date as possible. DMS reserves the right to revise this document and to make changes from time to time without the obligation to notify any person of such revision or changes.

DMS assumes no liability resulting from any omissions in this document, or from the use of the information obtained therein.

Registered Trademarks:
OS-9 and Microware are registered trademarks of Microware System Corp. Ltd.
UNIX is a registered trademark of Bell Laboratories
MS-DOS is a registered trademark of Microsoft inc.
Motorola is a registered trademark of Motorola inc.

The manual itself or parts thereof may be used for the documentation of devices, systems or plants that contain a DMS product described in this manual only if proper references (including copyright references) have been made. Further use of this manual such as taking copies of the material is prohibited without explicit consent of DMS and may incur liability suits. All rights reserved. Technical changes may occur.

© DMS Dorsch Mikrosystem GmbH 1996

Preface

The user requires comprehensive information in order to make full use of the technical abilities of this product. This manual is intended for hardware developers, project engineers, and programmers who plan on using this VMEbus device.

The present publication has been designed to provide you with a complete range of structured information. The following pages contain information which should make it easier for you to use this manual. You will be provided with information on how this manual is structured.

Despite all efforts from our side some problems may not be covered by this manual. This is mainly due to the broad range of possible VMEbus applications. Please do not hesitate to contact DMS if you experience any problems that are not covered by the 'Troubleshooting' section.

Contents

- Hardware and Technical Data (Chapter 1)
  This chapter contains a description of the device itself, how it fits into our range of VMEbus products, and its principles of operation.

- How to Setup the Unit (Chapter 2)
  This chapter summarizes all data necessary for setting up the device. Special care has been taken to make clear that hardware and software influence each other.

- Primary Functions and Operating Principles (Chapter 3)
  Chapter 3 contains a full description of built-in functions, i.e. information on wiring the device and how to program it.

- Address Information (Chapter 4)
  This chapter details all address ranges and register addresses.

- Software (Chapter 5)
  This chapter contains a description on hardware-dependent software such as drivers and descriptors.

- Service and Maintenance (Chapter 6)
  The information contained in this chapter includes details on maintenance and service issues as well as a troubleshooting section where common problems are discussed that may occur during operation of the card.

- Appendix (Chapter 7)
  This chapter contains wiring diagrams, dimension drawings, etc.

The last pages of this manual contain a number of forms (e.g. 'Reader Comments' or 'DMS Return Form'). Please fill in those forms if you would like to propose any changes, corrections or supplements. We suggest that you mail the sheet back to us. Your suggestions will help us to improve the quality of future editions of this manual.

Printing Conventions

In order to improve its readability this document is structured like a menu.

- The first pages of this document contain a chart of contents.
- Pages, illustrations, and tables are numbered consecutively.
- Abbreviations have been used for a number of terms. You will find a list of abbreviations in the Appendix Section.
- Footnotes are indicated by text printed in superscript such as numbers (e.g. "[1]"), or stars ("*"). Generally, the comments related to a footnote can be found at the bottom of the page. Bullets (•) and dashes are used in most cases (like for example in this list) to indicate an enumeration.
- Cross references are used as follows: "(see Chapter 3.3.2)" refers to Section 3.2 of chapter 3.
- All dimensions given in drawings, sketches, and tables are in millimeters (mm).
- Value ranges are given in the following form: 17 .. 21 means a range from 17 to 21.
- Hexadecimals are indicated by a leading "$".
- You will find that outlined boxes like the one below often contain very important information:

Warning

Please take the time to read through the section 'Information on Operational Safety' which is located at the end of this introduction. It contains definitions for certain terms like 'Warning', 'Danger', 'Caution', 'Note'

Information on Operational Safety

This manual contains all information necessary for the designated use of the products described in it. It is written for the use by qualified personnel. The term 'qualified' as used in this manual or as printed on the product itself refers to persons

- that are working in the field of research and development
- or that are members of the project staff and are familiar with all safety regulations used in automation technology.

Safety Precautions

The following safety precautions will help to prevent personal injuries as well as the protect the product and other devices connected to it.

Safety precautions and warnings that may threaten the life or health of the operator or maintenance personnel or that may result in damage to property will be indicated by signal terms defined below. The terms used in this manual or on product labels have the following meaning:

Danger

means that death, major injuries, or major damage will occur if the appropriate safety precautions are not obeyed.

Warning

means that death, major injuries or major damage may occur if the appropriate safety precautions are not obeyed.

Caution

means that minor injuries or damages may occur if the appropriate safety precautions are not obeyed.

Note

this section contains useful hints about the product, product handling or the part of the manual which it refers to explicitly.

Normal use of the product

Warning

- The device/system must not be used in cases other than those mentioned in the catalogue or the manual and may only be used in conjunction with third party components that are either recommended or approved by DMS.
- Professional transport, storage, erection/installation as well as careful operation and servicing are a prerequisite for the trouble free and safe operation of the device.

Declaration of Conformity

We, DMS Dorsch Microsystem GmbH, declare under our sole responsibility that the product described in this technical documentation complies with the standards or other normative documents quoted in this notice.

All DMS devices that features a CE symbol fulfill the requirements of the EU guideline #89/336/EWG on electromagnetic compatibility as well those of the harmonized European standards stated below.

As required by paragraph 5.1(3) of the 'Electromagnetic Compatibility - Federal Office of Postal Services and Telecommunications Act' the declaration of conformity as well as the appropriate documentation are kept with:

DMS Dorsch Mikrosystem GmbH
Holmlück 15
D-24972 Steinberkirche
Germany

Range of Application

The VME-PROFI device has been designed for industrial operation and complies with the following standards and normative documents:

Electromagnetic Compatibility (EMC) - Electromagnetic Emissions : EN 50081-2:1994
Electromagnetic Compatibility (EMC) - Generic Immunity Standard: prEN 500082-2: 1994

Installation Guidelines

The DMS device does meet all EMV requirements under the following conditions:

1. The device is operated in a DMS rack with CE certification.
   All devices of a VMEbus system need to comply with the EU guidelines on electromagnetic compatibility (#89/336/EWG) (i.e. must feature a CE symbol).
2. The VMEbus-system frame ground must be connected to the system ground through a low resistance cable.
3. PROFIBUS systems must employ cables and connectors that are recommended by DMS or that are compatible with those recommended by DMS (see Chapter 3.2.1.1).
4. Shielded Cables are to be used for all additional data connections. All connectors (jacks, sockets, and plugs) must be fitted with a metal shielding. The protective case must be connected bidirectionally conducting with the cable shielding. In order to prevent transient currents within the GND shielding, the potential drop needs to be compensated for by means of bonding.
5. Data cables must not be installed near the mains lead or near cables with a high throughput or switching high inductance loads.
6. When installed outdoors all cables that may be subjected to excessive voltages (e.g. lightning) are to be fitted with lightning arresters or to be protected by metal tubes.
7. All slots of a system rack that are not currently in use need to be closed by an electrically conducting front panel.
8. During operation and installation of the device - special care is required with respect to electrostatic discharge (ESD - see also EN 10015-1 Basic Specification: Protection of electrostatic sensitive devices).

Validity of this Manual

This manual is valid for:

VMEbus-Cards:	VME-PROFI Version 2.1 or later
	VME-PROF-S Version 2.1 or later
	VME-PROF-A Version 2.1 or later
	VME-PROFSA Version 2.1 or later
Firmware:	PROFIBUS-DP-Master Version 3.02 or later
	PROFIBU-DP-Slave Version 1.0 or later

Table of Contents

Preface 3
Contents 3
Printing Conventions 4
Declaration of Conformity 6
Validity of this Manual 7
Table of Contents 8
Tables 11
Illustrations and Diagrams 11

1. VME-PROFI - Overview 12
1.1. General Information 12
1.2. Block Diagram 13
1.3. Technical Specifications 14
1.4. The VME-PROFI Package 14
1.5. Order Codes 15
1.6. Safety precautions 16

2. Hardware Installation and Configuration 17
2.1. System configuration 17
2.2. Hardware settings 18
2.2.1. VMEbus slave address range (Slot Address) 18
2.2.1.1. Address setting 18
2.3. Final Installation 19
2.4. Software Setup 20
2.4.1. How to load or update the VME-PROFI Firmware 20
2.4.1.1. System requirement 20
2.4.1.2. Contents of the disks „VMEPROFI" 20
2.4.1.3. Installing of the OS-9 software 20
2.4.1.4. Loading of the VME-PROFI's firmware 21
2.4.2. Additional software settings 21
2.5. How to Remove the Card 21
2.6. Shipping and Packaging 21

3. Function Elements of the VME-PROFI 22
3.1. Status LED's 23
3.2. Interfaces 23
3.2.1. PROFIBUS-Interface X2 23
3.2.1.1. PROFIBUS-Cable 24
3.2.1.2. Number of Stations 25
3.2.2. Terminal-Interface X1 26
3.3. Ports and Connectors 27
3.3.1. Test Ports ST40, ST43, ST44, ST45 27
3.3.2. Boundary Scan Test Port ST10 27
3.4. The VMEbus Interface 27
3.4.1. VMEbus Master 27
3.4.2. VMEbus Slave 27
3.4.3. VMEbus Interrupt-Handler 28
3.4.4. VMEbus Interrupter 28
3.4.5. VMEbus-Signals 28

3.4.5.1. SYSFAIL 28
3.4.5.2. SYSRESET 28
3.5. J2-Bus / Slot depending addressing 28

4. System information 29
4.1. Enabling of the VME-PROFI 29
4.2. Address-Map Standard-VME-range 29
4.2.1. VMEbus Address-Map DP-Master Software 30
4.2.2. VMEbus Address-Map DP-Slave Software 31

5. Software 34
5.1. PROFIBUS-DP Master 35
5.1.1. Configuration of the PROFIBUS 35
5.1.2. PROFIBUS data transmission 35
5.1.2.1. DP-RAM data interface 36
5.1.2.2. Cycle time of the PROFIBUS data transmission 36
5.1.3. Configuration program 37
5.1.3.1. Terminal connection 37
5.1.3.2. Screen display 37
5.1.3.3. Commands 38
M - Enter Master parameter 38
S - Add Slave 40
L - Remove Slave (Löschen) 42
O - Automatic assignment of the I/O-Address Offset 42
R - Storage of the configuration data and Restart 43
V - Restart Parameter from the VMEbus Host 43
G - Get diagnostic data 44
W - Show additional Slaves 45
Z - Display refresh back to first page 45
<Space> - Update display of input data 45
? - Help display 46
5.1.3.4. Setting of outputs 46
5.1.4. Configuration of the PROFIBUS through the VMEbus 47
5.1.5. Demo programs 48
5.1.5.1. Ultra C (ANSI C) 48
5.1.5.2. Example for Berghof VME-S5 CPU 50
5.1.5.3. Example fo a ADEPT-System 51
5.1.6. Difference to Profibus-DP-Master Software Version 1.x, V2.x, V3.x 53
5.1.7. Known bugs of the DP-Master Software 53
5.2. PROFIBUS-DP-Slave Software 54
5.2.1. Terminal interface 54
5.2.2. Slave configuration 54
5.2.3. DP-RAM data interface 55
5.2.3.1. Status- Variable 55
5.2.3.2. Sema-Variable within the input- and output buffer 56
5.2.3.3. Consistency data access 56
5.2.3.4. Data access without consistency 56
5.2.4. Demo program 57
5.2.4.1. Ultra C (ANSI C) 57
5.2.5. Known bugs of the DP Slave software 61

6. Service and Maintenance 62

6.1. Troubleshooting 62
6.1.1. General troubleshooting 62
6.1.2. VME-PROFI(DPM1) troubleshooting 63
6.1.3. VME-PROFI(DPS) troubleshooting 65

7. Diagrams and Appendix Section 66
7.1. Summary of Abbreviations 66
7.2. References 66
7.3. Pin Layout ST1 67
7.4. Pin Layout ST2 68
7.5. Jumper diagram 69
7.6. Component Mounting Diagram 70
7.7. Wiring Diagram 71
7.8. Configuration example 76
7.9. Reader Comments 78
7.10. DMS Return Form 79

Tables

Table 1: Pin Assignment ST2 (slot depending address) 18
Table 2: J30, Address Bits 19
Table 3: Pin-layout PROFIBUS X2 23
Table 4: PROFIUS cable parameter 24
Table 5: Maximal bus length 24
Table 6: Pin layout Terminal-interface X1 26
Table 7: Summary of VMEbus Functions 27
Table 8: Dualport RAM DP-Master software 30
Table 9: Dualport RAM DP-Slave software 32
Table 10 : Firmware - Order Code 34
Table 11: Comparison of Profibus-DP-Master Vers. 1.x, 2.x to Vers. 3.x 53
Table 12: Pin Layout ST1 67
Table 13: Pin Layout ST2 68

Illustrations and Diagrams

Figure 1: Picture of the VME-PROFI 12
Figure 2: Block Diagram VME-PROFI 13
Figure 3: Typical system configuration with VME-PROFI(DPM1) 17
Figure 4: Jumper on the J2 backplane 19
Figure 5: Front View of the VME-PROFI 22
Figure 6: Bus cable with termination 25
Figure 7: PROFIBUS-Termination 25
Figure 8: Interface cable VME-PROFI X1 - PC (VT100 Terminal) 26

1. VME-PROFI - Overview

1.1. General Information

The VME-PROFI is a compete independent PROFIBUS-interface for the VMEbus. The interface is based on the PBM-ASIC which handles the whole layer 1 / 2 of the PROFIBUS protocol (DIN 19245). Through the MC 68331 processor with onboard Flash-EPROM the board can run complete independent without loading any PROFIBUS software from the host VMEbus CPU. 64 kB dualport-RAM are used as data interface to the VMEbus. The integration of a PROFIBUS-interface (DP-Master or DP-Slave) in a VMEbus-system can easily be realized with the VME-PROFI independent from the used operating system.

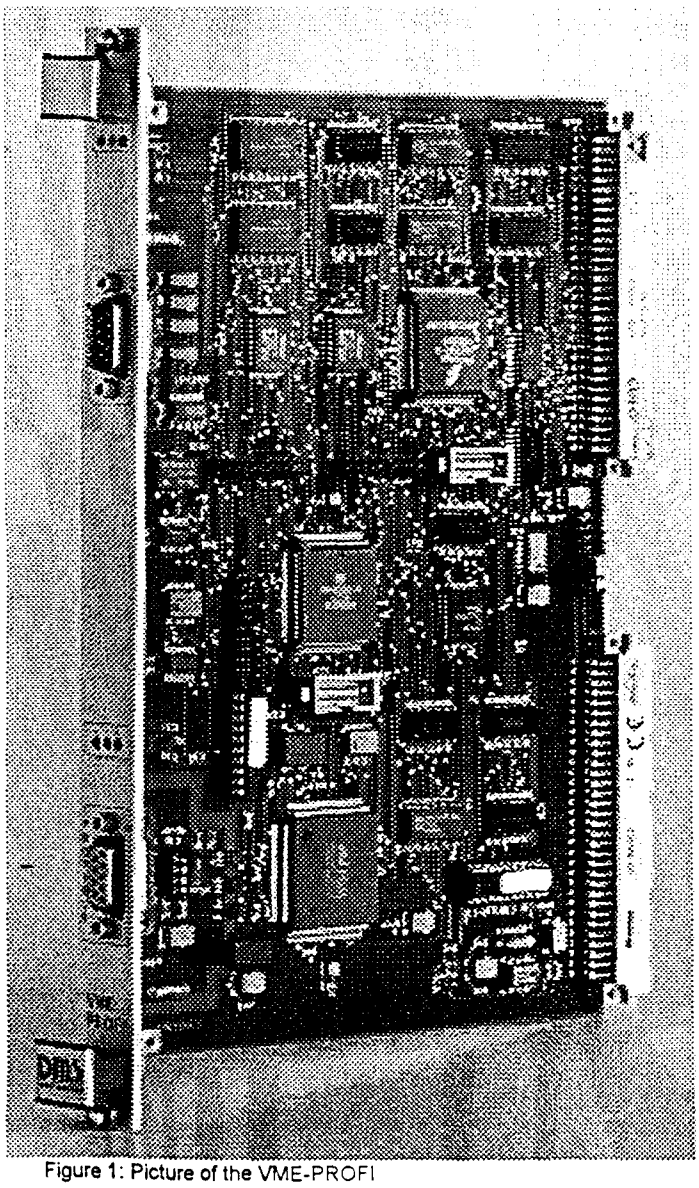


Figure 1: Picture of the VME-PROFI

The VME-PROFI can handle different PROFIBUS protocols. The firmware for the PROFIBUS-DP-Master (class 1) and for the PROFIBUS-DP-Slave protocol are available.

With the PROFIBUS-DP-Master firmware, the board builds a independent working DP-Master interface (class 1) for Mono-Master or Multi-Master systems. All DPM1-functions, including configuration of the bus, are done from the VME-PROFI. The VMEbus-system can read or write the I/O-bytes of the distributed peripheral stations by reading or writing of the VME-PROFI dualport-RAM. The configuration and test of the PROFIBUS-system can easily be done with the included configuration program. All configuration-data are stored in the onboard EEPROM.

With the PROFIBUS-DP-Slave firmware, the VME-PROFI builds a independent arbitrary configurable DP-Slave interface. The slave is configured through the dualport-RAM. The configuration can be build from up to 32 configuration bytes. The total amount of input or output data can be up to 243 input byteand up to 244 output byte. Due to the arbitrary configuration the Slave can fit the requirements of the VMEbus system or the VME-PROFI can be used to simulate other Slaves.

In the following description the VME-PROFI with the DP-Master (class1) firmware are named VME-PROFI(DPM1) and the VME-PROFI with the DP-Slave firmware are named VME-PROFI(DPS).

1.2. Block Diagram

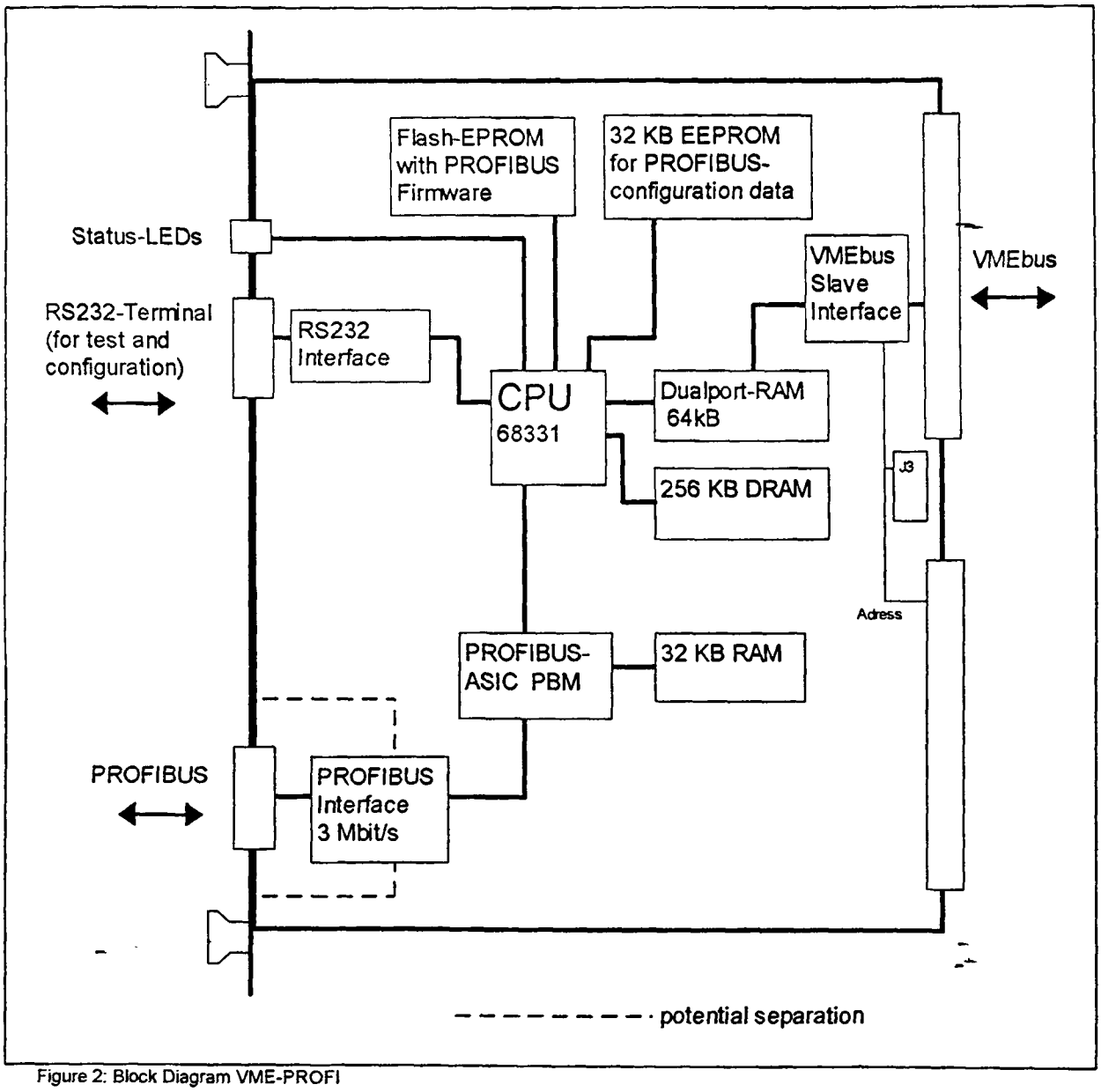


Figure 2: Block Diagram VME-PROFI

1.3. Technical Specifications

VMEbus:	A24, D16 Slave
Memory:	64KB Dualport-RAM
	256 KB RAM
	512 KB Flash EPROM
	32 KB EEPROM
	32 KB local RAM for the PBM
Interface:	1 * RS485 PROFIBUS, galvanic separated, 3MBit/sec with PROFIBUS-ASIC
	1 * serial RS232
Processor:	68331
Protocol:	PROFIBUS DP-Master (DIN 19245)[1]
	PROFIBUS DP-Slave (DIN 19245)[1]
Slot addressing	DMS slot depending addressing via J2-Bus
	(VME-PROF-A, VME-PROFSA without slot address coding)
Dimension:	6HE-4TE (double-Euro-format, 233*160mm)
Power:	+5V: typ. 700 mA
	+5V Standby not needed
	+12V and -12V not needed
Temperature	
Operating:	0 ... +50 °C
Storage:	-25 ... +70 °C

[1] Only one protocol can running simultaneous on the VME-PROFI. If necessary, another protocol can be loaded into the VME-PROFI's Flash-EPROMs via the dualport-RAM. A utility program to load the Firmware is available for OS-9 systems.

1.4. The VME-PROFI Package

The VME-PROFI package includes:
- the VME-PROFI board
  (incl. PROFIBUS-DP-Master software in the onboard Flash)
- Disc (PC1.44 format) with ANSI-C demo programs and OS-9 Software
- this technical Manual

The VME-PROF-S package includes:
- the VME-PROF-S board
  (incl. PROFIBUS-DP-Slave software in the onboard Flash)
- Disc (PC1.44 format) with ANSI-C demo programs and OS-9 Software
- this technical Manual

The VME-PROF-A package includes:
- the VME-PROF-A board
  (incl. PROFIBUS-DP-Master software in the onboard Flash)
- Disc (PC1.44 format) with ANSI-C demo programs and OS-9 Software
- this technical Manual

The VME-PROFSA package includes:
- the VME-PROFSA board
  (incl. PROFIBUS-DP-Slave software in the onboard Flash)
- Disc (PC1.44 format) with ANSI-C demo programs and OS-9 Software
- this technical Manual

1.5. Order Codes

VME-PROFI	VMEbus PROFIBUS-Interface 6 HE, 3 Mbit/s incl. PROFIBUS-DP-Master Software
VME-PROF-S	VMEbus PROFIBUS-Interface 6 HE, 3 Mbit/s incl. PROFIBUS-DP-Slave Software
VME-PROF-A	VMEbus PROFIBUS-Interface 6 HE, 3 Mbit/s incl. PROFIBUS-DP-Master Software, without J2-connector
VME-PROFSA	VMEbus PROFIBUS-Interface 6 HE, 3 Mbit/s incl. PROFIBUS-DP-Slave Software, without J2-connector
Other PROFIBUS products:	
VME-PROFI3	VMEbus PROFIBUS- Interface 3 HE, 500 kbit/s incl. PROFIBUS DP-Master Software
PCI-40	PC ISA-Slotcard with 68040 CPU, 64 MHz, 4 MB RAM and PROFIBUS interface 3 Mbit/sec
PCI-40S	PC ISA-Slotcard with 68040 CPU, 64 MHz, 8 MB RAM, SERCOS and PROFIBUS interface 3 Mbit/sec
PCP-DP	PC ISA-Slotcard with PROFIBUS interface 3 Mbit/sec
PMOD1	Digital I/O-Module
PMODS1	Terminal block for Digital-I/O- Modules
Accessories:	
PROFI-STV	PROFIBUS Bus connector (9 pol. DSUB) with switchable terminating resistors
PROF-KABEL	PROFIBUS cable
Software:	
DPKONFIG	PROFIBUS DP configurator for Windows95

1.6. Safety precautions

Caution

- The device may only be used in subracks that meet the requirements of the following VMEbus standards: ANSI/IEEE STD1014-1987, IEC 821 and 297 [2]
- The device may only be switched on if installed properly.
- The device must be disconnected from the VCC power supply prior to connecting or disconnecting hardware interfaces.
- Handle with care! The device must be protected from electrostatic discharge (see EN 100015-1).
- This device uses, produces and may emit high-frequency signals. Operation of the device may be disturbed by strong high-frequency signals. The device has been designed by DMS for a high degree of electromagnetic compatibility, e.g. galvanic separated front interfaces, multi-layer technology, etc. have been incorporated in the design. Please make sure that the device is used in compliance with all applicable regulations regarding electromagnetic compatibility (EMC). See „Declaration of Conformity" in the Preface chapter.

2. Hardware Installation and Configuration

2.1. System configuration

The following block diagram shows a typical system configuration with a VME-PROFI as DP-Master. The VME-PROFI, as a VMEbus-Slave assembly in the VMEbus-Rack, work as the PROFIBUS Master Station.
The RS232 Terminal is only necessary for the configuration and test of the system.

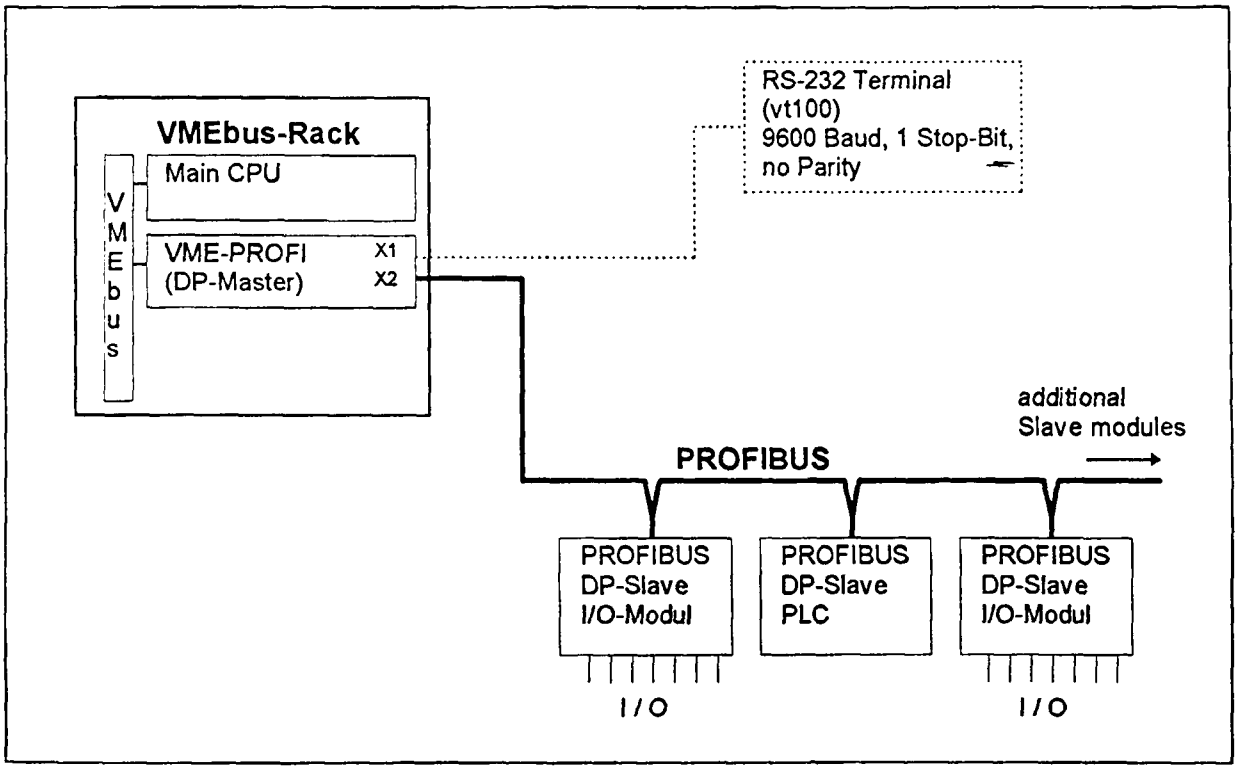


Figure 3: Typical system configuration with VME-PROFI(DPM1)

The VME-PROFI(DPM1) can be used in systems with one Master (Mono-Master-Systems) or in systems with more than one Master (Multi-Master-Systems).

Tom couple a PLC with a VMEbus-system, the VME-PROFI can be used with the DP-Slave software. In this case the PLC works as DP-Master and the VME-PROFI(DPS) wors as DP-Slave.

2.2. Hardware settings

Warning
The system must be switched off when changing hardware within or connected to the system!

2.2.1. VMEbus slave address range (Slot Address)

The VME-PROFI use 64 kByte within the VMEbus standard range. The base address can be selected in 64 k steps within the 16 MB address space. The address setting is done with the addressbits (A19 - A16) at ST2 and / or with the jumper (J30). A Jumper Diagram with the default settings can be found in chapter 7.4.

Base address: $xy0000

- x: (A23-A19)
  Set by J30/1-2...J30/7-8
  Default: **$8** (J30/1-2 open, J30/3-4...J30/7-8 closed)
- y: (A19-A16)
  Set by J30/9-10...J30/15-16 or the slot-address-code at ST2.
  Default: **$F** (J30/9-10...J30/15-16 open)
  => slot depending address coding via ST2

2.2.1.1. Address setting

There are 3 alternate ways for setting the VMEbus-Slave address range:

Note:
The board can also be ordered without slot depending addressing (without connector ST2). In this case, the 3. way (address setting with J30) is possible

1. Address Selection According to Location (default Setting for DMS Subracks)

- DMS subracks of the S type will select the address range according to the location of a card by using the A1-C4 signals in the J2 backplane.

A1	GND	C1	GND
A2	BIT 3 (SL.3)	C2	BIT 2 (SL.2)
A3	GND	C3	GND
A4	BIT 1 (SL.1)	C4	BIT 0 (SL.0)

Table 1: Pin Assignment ST2 (slot depending address)

The VMEbus slave address range of a system will be assigned according to the settings of J30/1-2 ... J30/7-8 and the selected slot. Jumpers A - D (see below) and J30/9-10 ... J30/15-16 must not be enabled.

VMEbus slot number	SL.3	SL.2	SL.1	SL.0	Slave address range (64 KB)
1	0	0	0	1	$x1 0000 .. $x1 FFFF
2	0	0	1	0	$x2 0000 .. $x2 FFFF
3	0	0	1	1	$x3 0000 .. $x3 FFFF
|					| |
15	1	1	1	1	$xF 0000 .. $xF FFFF

x. set with J30/ 1-2 ... J30/7-8 (Default x = $8)

2. Address settings (slot depending address) with jumper on the J2-backplane

Slot depending addressing can also be selected using jumpers on the J2 backplane, if your system does not support address selection according to location.

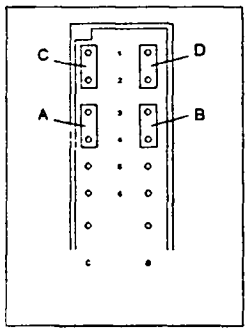


Figure 4: Jumper on the J2 backplane

A = LSB of the slot address (= SL.0)
B = LSB+1 of the slot address (= SL.1)
C = LSB+2 of the slot address (= SL.2)
D = MSB of the slot address (= SL.3)

Open jumpers correspond to a '0'
Closed jumpers correspond to '1'

Jumper J30/9-10...J30/15-16 must be open.
A23-A20 are set by J30/ 1-2...J30/7-8.

3. Address setting with J30 (no slot depending address coding)

The VMEbus base address is only set by J30. a open jumper correspond to a '0', a closed jumper correspond to '1'.

J30	Address Bit
1 - 2	A23
3 - 4	A22
5 - 6	A21
7 - 8	A20
9 - 10	A19
11 - 12	A18
13 - 14	A17
15 - 16	A16

Table 2: J30, Address Bits

The address signals at ST2 (A2, A4, C2, C4) nust be open.

2.3. Final Installation

After finishing the installation align and insert the VME-PROFI card into the designated slot and tighten it by fastening the two screws at the top and bottom of the cards front panel. Safe and trouble free operation can only be ensured if the card is well aligned and tightened.

2.4. Software Setup

2.4.1. How to load or update the VME-PROFI Firmware

The VME-PROFI provides 512 KB onboard Flash-EPROM for the storage of the firmware. By default, the VME-PROFI is shipped with the PROFIBUS-DP-Master software.
The Board can also be ordered with pre-installed PROFIBUS-DP-Slave software (see chapter 1.5).
For firmware update or for changing of the PROFIBUS protocol, the software can be loaded via the dualport RAM into the Flash-EPROM. A OS-9 loading-program and the object-files of the DP-Master and the DP-Slave are shipped together with the board on the disk „VMEPROFI".

2.4.1.1. System requirement

6U VMEbus-system with OS-9 operating system and 3 1/2" floppy disk (IBM1.44 format).

2.4.1.2. Contents of the disks „VMEPROFI"

The VME-PROFI's OS-9 software are shipped on a 3 1/2" floppy disk in the pcd0h (IBM1.44) format. The disk contains ANSI-C demo programs, the GSD-file for the VME-PFOFI(DPS), the OS-9 loading-program and the DP-Master and DP-Slave object-files.
Because of the restricted directory and file naming on standard DOS disks, the OS-9 software including the DP-Master and DP-Slave object-files are stored in a compressed file. The utility program `lharc` is used to extract the OS-9 software.

Contents of the disk „VMEBUS":

`GSD`	Directory with VME-PROFI(PDS) GSD-file (uncompressed)
`SOURCE`	Directory with ANSI-C demo programs (uncompressed)
`install.os9`	install file for installing of the OS-9 software
`vmeprofi.lzh`	compressed OS-9 software, incl. DPM1 and DPS objekt files
`lharc`	OS-9 decompression tool for `.lzh` files

The contents of the comoressed file `vmeprofi.lzh` can be browsed by:

```
chd /pcd0h
load lharc
lharc -v vmeprofi
```

2.4.1.3. Installing of the OS-9 software

To install the OS-9 software the file `install.os9` could be found in the root directory of the disk. With this file, the compressed OS-9 software are decompressed and stored under the path `/h0/os9/vmeprofi` on the harddisk.

For the installing the following devices must be available:

`/pcd0h`	3 ½" floppy disk which could read DOS disks in the high density format (IBM1.44)
`/h0`	hard disk for the OS-9 system

`/term` system terminal

If this devices are not available, or different device names are used, than the `install.os9` file must be modified, to match the actual devices.

2.4.1.4. Loading of the VME-PROFI's firmware

The OS-9 program `vsload` is used to load the firmware.

Syntax: `vsload` s <object file>
s = slot number 1 .. 15 in case of a DMS VMEbus rack
= 15 without slot depending address selection (see chapter 2.2.1)
(`vsload` will calculate the basis address of the board in the following way: Adr = $ 80 0000 + s * $1000)

< object file> = `dpm1.vp` for PROFIBUS-DP-Master firmware
= `dps.vp` for PROFIBUS-DP-Slave firmware

Example: Loading of the DP-Slave firmware, VME-PROFI at slot no. 7:

```
load -d /h0/os9/vmeprofi/cmds/vsload
chd /h0/os9/vmeprofi/obj
vsload 7 dps.vp
```

2.4.2. Additional software settings

Other software settings are depending of the used firmware. See chapter 5.

2.5. How to Remove the Card

Prior to removing the card all connectors at the card's front panel need to be disconnected. Remove the two screws at the bottom and the top of the front panel. Pressing the upper card handle towards the upper end of the card and the lower handle respectively towards the bottom will loosen the card. Now you can remove the card from the subrack.-

2.6. Shipping and Packaging

Packing needs to be done with greatest care. The card needs to be protected from electrostatic discharge. If possible use the original packaging.
A 'DMS Return Form' (see Appendix) must be included with all returns to DMS.

3. Function Elements of the VME-PROFI

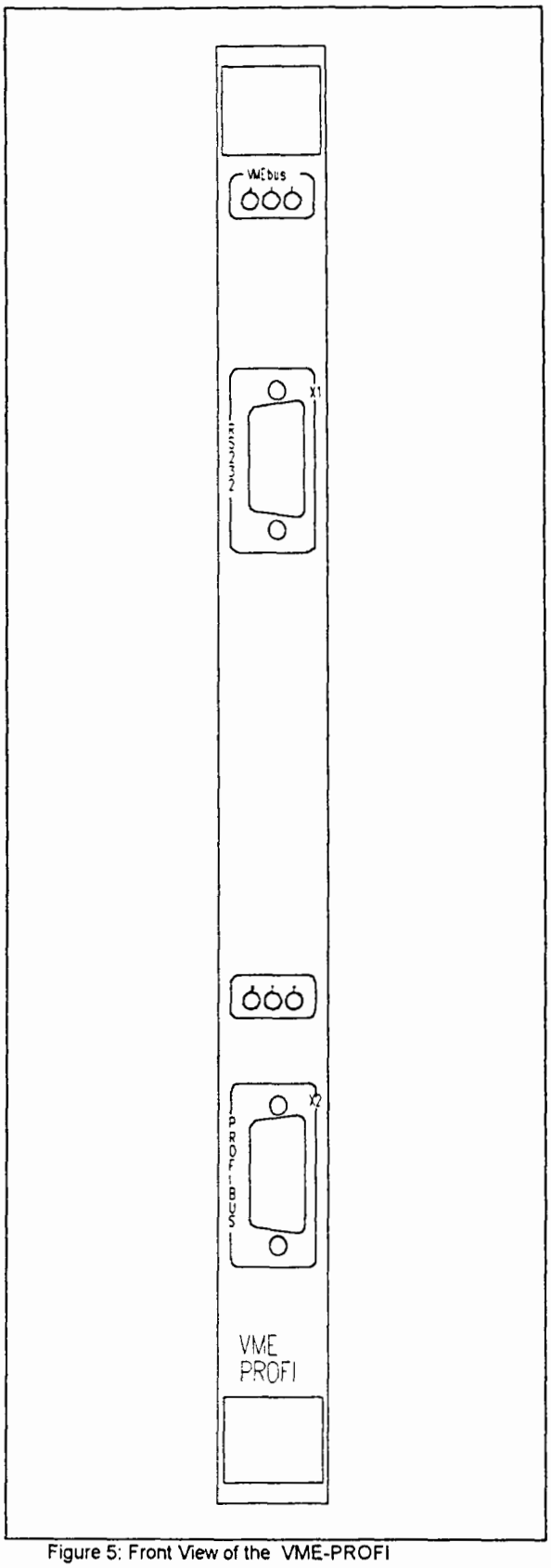


Figure 5: Front View of the VME-PROFI

3.1. Status LED's

6 LED's will deliver information on the VME-PROFI's operating state. The meaning of the 1. LED row is independent of the loaded firmware:

1. LED-row (VMEbus state)

green	(A active)	VMEbus access to the VME-PROFI
yellow	(I Interrupt)	the VME-PROFI generates an VMEbus-IRQ
red	(F Fail)	the VME-PROFI is in RESET (disabled)

The meaning of the 2. LED row depends of the loaded firmware:

2. LED-row (PROFIBUS state) with PROFIBUS-DP-Master firmware

green	(R RxData)	the VME-PROFI receives data from the PROFIBUS (the PROFIBUS-slave answers)
yellow	(T TxData)	the VME-PROFI transmit data
rot	(F Fail)	Off: The PROFIBUS-Controller is running, all configurated slaves are ok.
		Flash: The PROFIBUS-Controller is running, at least one of the configurated slaves are not ok.
		On: the VME-PROFI is in hold (not enable, PROFIBUS-Controller stop)

2. LED-row (PROFIBUS state) with PROFIBUS-DP-Slave firmware

green	(R RxData)	the VME-PROFI receives data from the PROFIBUS
yellow	(T TxData)	the VME-PROFI transmit data
rot	(F Fail)	Off: The PROFIBUS-Controller is running,
		On: the VME-PROFI is in hold (not enable, PROFIBUS-Controller stop)

3.2. Interfaces

3.2.1. PROFIBUS-Interface X2

The PROFIBUS an be accessed via a 9-pin connector X2 located at the front panel. The interface has separated potentials and a pin-layout as specified in the DIN 19245.

Connector X2	I/O - Signal
Pin 1	n.c.
Pin 2	n.c.
Pin 3	Data B (RxD/TxD-P)
Pin 4	CNTR (Repeater control signal TTL)
Pin 5	GND
Pin 6	Termination-Power
Pin 7	n.c.
Pin 8	Data A (RxD/TxD-N)
Pin 9	n.c.
Connector case	shield

Table 3: Pin-layout PROFIBUS X2

3.2.1.1. PROFIBUS-Cable

The PROFIBUS standard DIN 19245 Part1 and Part3 permits two types of bus cable (Cable A and Cable B).

Parameter	Cable A PROFIBUS-DP DIN 19245 Part 3 Chap. 6.2	Cable B DIN19245 Part 1 / 4.91 Chap. 3.1.2.3
type of cable	Two-conductor shielded twisted pair cable	Two-conductor shielded twisted pair cable
Impedance	135 ... 165 Ω (3 ... 20 MHz)	100 ... 130 Ω (f > 100kHz)
Capacity	< 30 pF/m	< 60 pF/m
Loop resistance	< 110 Ω/km	-
Diameter	> 0,64 mm	> 0,53 mm
	> 0,34 mm²	> 0,22 mm²

Table 4: PROFIUS cable parameter

The maximal bus length depends on the type of cable and the baud rate (see table).

Baud rate kBit/s	9,6	19,2	93,75	187,5	500	1500	3000
Cable A, max. distance in m (without repeater)	1200	1200	1200	1000	400	200	100
Cable B, max. distance in m (without repeater)	1200	1200	1200	600	200	--	--

Table 5: Maximal bus length

The maximal bus length can be extended until approx. 10 km with repeaters. The max. number of repeaters that can be used in a network depends on the type of the repeater and is between 3 and 10.

The PROFIBUS is a linear bus from device to device. Only short stubs are possible, see DIN 19245 Part1 and 3. The bus cable can be ordered from DMS with the order code PROF-KABEL. The bus cable must be terminated at both ends. We recommend connectors with integrated, switchable termination resistors (PROFI-STV see chapter 1.5)

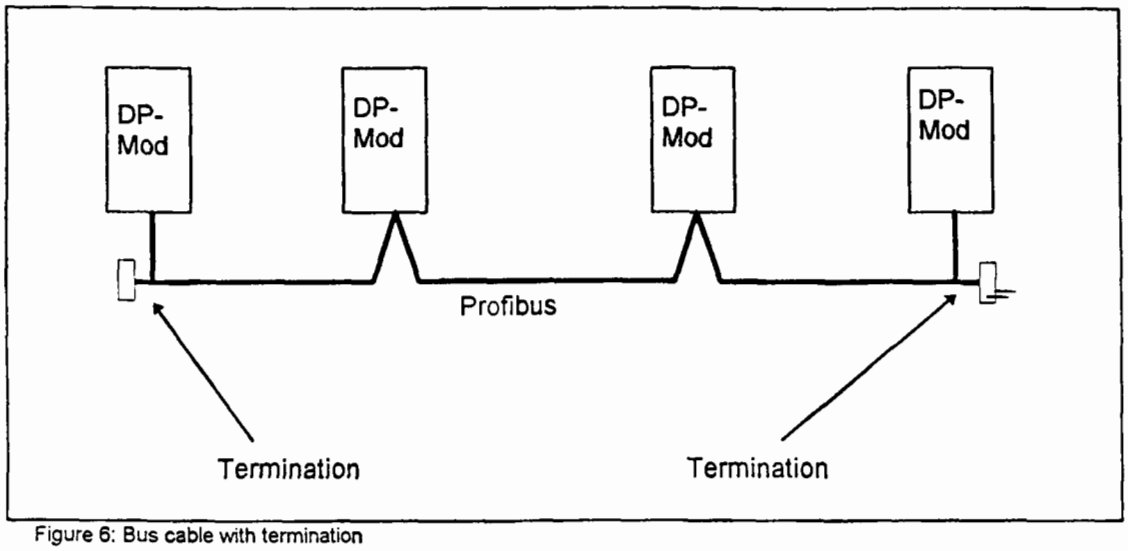


Figure 6: Bus cable with termination

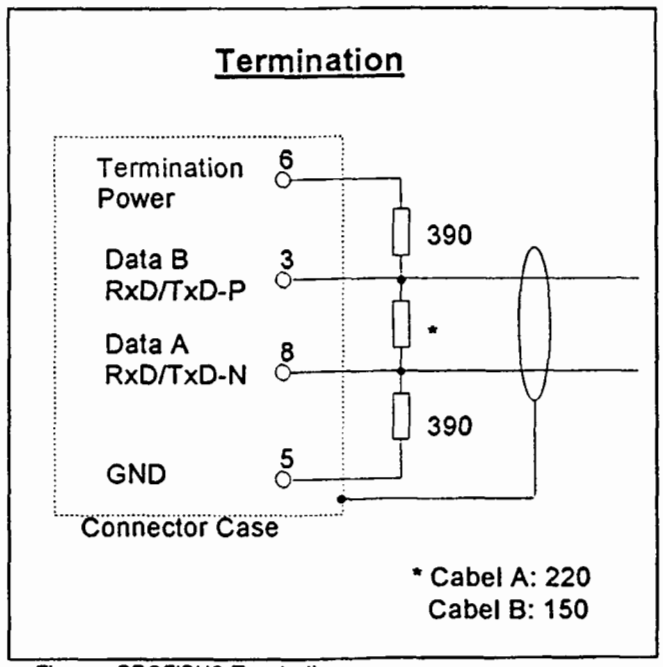


Figure : PROFIBUS-Termination

3.2.1.2. Number of Stations

Max. 32 devices are allowed per bus segment (without Repeater). With repeaters the number can be expanded up to 126 devices.

3.2.2. Terminal-Interface X1

X1 is provided for the connection of a RS232-Terminal. The Pin-layout is like the PC-AT standard. Only the signals RxD, TxD and GND are necessary. The interface parameter are: 9600 Baud, 1 Stop-Bit , No-Parity, Terminal-Emulation ANSI VT100. A PC with a terminal program (for example: Windows Terminal) can be used instead of a VT100 Terminal.
The function- and cursor-keys of the Windows Terminal must be enabled („not for Windows"). The cursor-keys are necessary for the controlling of the VME-PROFI

Note:
The interface has no separated potentials and should only be used for test and configuration. From the DP-Slave Software this interface is only used for debug information's.

Pin	RS232
1	-
2	RxD input data
3	TxD output data
4	-
5	GND
6	-
7	RTS output [1]
8	CTS input [1]
9	-

Table 6: Pin layout Terminal-interface X1
[1] do not connect

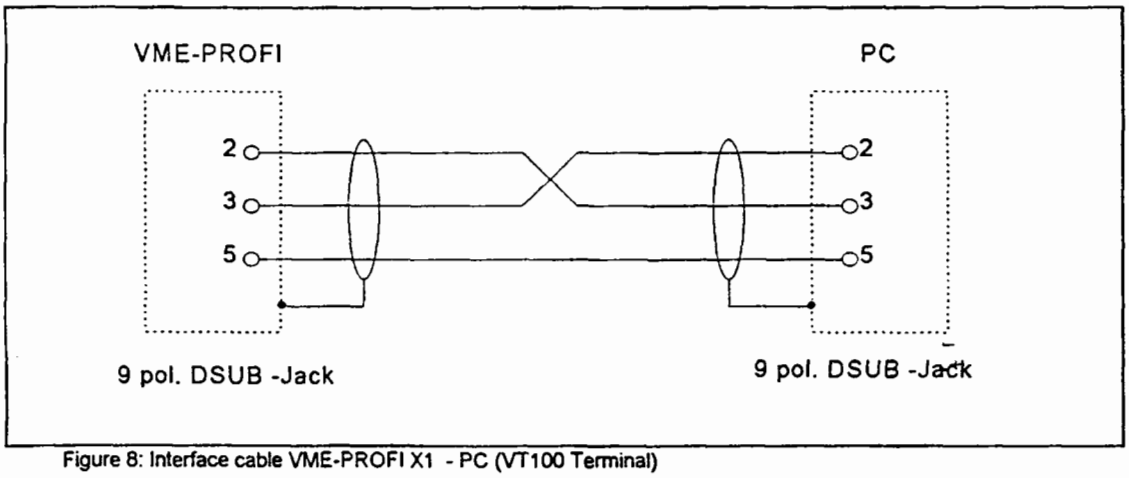


Figure 8: Interface cable VME-PROFI X1 - PC (VT100 Terminal)

Warning
The configuration of input-/output-bytes to the dualport RAM addresses can be changed from the terminal. That can cause wrong settings of output bytes or reading wrong input bytes. Additionally, output bytes can be set directly.

Never use the terminal to control the VME-PROFI, if connected machines are running!

3.3. Ports and Connectors

3.3.1. Test Ports ST40, ST43, ST44, ST45

The Test Ports are only for factory tests.

3.3.2. Boundary Scan Test Port ST10

The VME-PROFI provides a boundary scan test port for testing purposes as well as for programming the IC1 chip on-board.

Note
The test port can only be used for programming and for factory test. It must not be utilized by the user.

3.4. The VMEbus Interface

The table below summarizes a number of VMEbus functions[1]:

VMEbus Function	Included	Remark
DTB Master Address Bus	-	no VMEbus-Master functions
Master Data Bus	-	" " " "
Slave Address Bus	A24	24 Bit address bus width
Slave Data Bus	D08, D16	8 and 16 Bit data bus width
SSEQ	no	Slave: Sequential Access
Interrupt-Handler	-	no Interrupt-Handler function
Interrupter	IH1, IH2, IH3, IH4, IH5, IH6	programmable IRQ-Level

Table 7: Summary of VMEbus Functions

[1] Definition of VMEbus functions see [2].

3.4.1. VMEbus Master

The VME-PROFI has no VMEbus-Master functions.

3.4.2. VMEbus Slave

The VME-PROFI use 64 kByte within the VMEbus standard range. The base address of that address-range can be set with the slot-number and / or jumpers (see chapter 2.2.1).

3.4.3. VMEbus Interrupt-Handler

The VME-PROFI has no Interrupt-Handler functions.

3.4.4. VMEbus Interrupter

The VME-PROFI card can generate IRQ-Level 1 to 6.

The IRQs are not used by the PROFIBUS DP-Master software.

3.4.5. VMEbus-Signals

3.4.5.1. SYSFAIL

The VME-PROFI will activate the SYSFAIL signal-line and lit the LED VMEbus/F, after power-on or RESET, until the VME-PROFI is enabled (see chapter 4.1).

3.4.5.2. SYSRESET

The VME-PROFI will be reset (disabled) from an active SYSRESET-signal.

3.5. J2-Bus / Slot depending addressing

the VME-PROFI permits a slot depending addressing by binary coded signals from the J2-bus.
The board used the signals A1 - A4 and C1 - C4 from connector ST2. This signal are connected to the address lines A19 - A16.
in DMS VMEbus subracks, this signals are binary code by resistor arrays. The code will be „0001" for the fist VMEbus-slot to „1111" for the 15th slot. In subracks without coded J2-backplane, the address bits ore open => high and the resulting slot number will be $F = 15. In this case the address bits can be set by J30.

Note:
If the J2-signals A1 - A4 and C1 - C4 in the VMEbus subrack are used for other signals, the VME-PROFI also can be ordered without ST2 (see chapter 1.5).

4. System information

The VME-PROFI comes with ready installed PROFIBUS software.

VME-PROFI => with PROFIBUS DP Master software
VME-PROF-S => with PROFIBUS DP Slave software.

No system programming by the user is necessary.
For changing or updating of the PROFIBUS software systems, a OS-9 utility program is available (see chapter 2.4).

The data exchange is done through the dualport RAM within the VMEbus Standard range.

4.1. Enabling of the VME-PROFI

After Power-on or SYSRESET the VME-PROFI is disabled. The Main-VMEbus-CPU must write a $08 at the VMEbus-address $8F000F[1] to enable the VME-PROFI.

[1] A23 - A16 of the VME-PROFI's VMEbus-Standard base address are set with the slot-number and / or jumpers (see chapter 2.2.1).
In the following description, the default base address ($8F0000) are always used.

4.2. Address-Map Standard-VME-range

The dualport RAM address map depends on the loaded Firmware (DP-Master or DP-Slave). Attached, the address-map for the DP-Master and the DP-Slave software is listed.

4.2.1. VMEbus Address-Map DP-Master Software

Address (Hex) VMEbus-Standard base address +	Type	Name	Remark
$0000	Byte	Dummy0	Dummy
$0001	Byte	ID1	"D"
$0002	Byte	IRQSET	reserved
$0003	Byte	ID2	"M"
$0004	Byte	IRQEN	reserved
$0005	Byte	ID3	"S"
$0006	Byte	Dummy3	Dummy
$0007	Byte	BoardID	$0B
$0008	Byte	Dummy4	Dummy
$0009	Byte	BoardED	Board Vers. = <D7..D4>.<D3..D0>
$000A	Byte	Dummy5	Dummy
$000B	Byte	Mail	reserved
$000C	Byte	Dummy6	Dummy
$000D	Byte	Vector	reserved
$000E	Byte	Dummy7	Dummy
$000F	Byte	Control	Card-Control-Register By writing $08 into this register, the VME-PROFI will be enabled
$0010	Lword[252]		reserved
$0400	Byte	Command	reserved
$0401	Byte	Error	reserved
$0402	Word[6]	Prm	reserved
$040E	Word	Live	"live" will be increased while the VME-PROFI is running
$0410	Byte[256]	ReadBuf	reserved
$0510	Byte[256]	WriteBuf	reserved
$0610 -$070F	Word[128]	DataBuf	reserved
$0710 - $0FFF			reserved
$1000-$1FFF			4 KB I/O-range Default: inputs starting at $1000 outputs starting at $1800 The I/O-range can be arbitrary configured by the user (see chapter 5.1.3.3).
$2000-$200F	Bit[128]		Bit-field variable, 1 Bit per Slave address (0 -127). The Bits are set every 200 ms, in the case, that the corresponding slave answered positive. Slave-Adr. 0 -> $2000 LSB (D0) Slave-Adr. 7 -> $2000 MSB (D7) Slave-Adr. 127 -> $200F MSB (D7)

Table 8: Dualport RAM DP-Master software

4.2.2. VMEbus Address-Map DP-Slave Software

Address (Hex) VMEbus-Standard base address +	Type	Name	Remark
$0000	Byte	Dummy0	Dummy
$0001	Byte	ID1	"D"
$0002	Byte	IRQSET	reserved
$0003	Byte	ID2	"M"
$0004	Byte	IRQEN	reserved
$0005	Byte	ID3	"S"
$0006	Byte	Dummy3	Dummy
$0007	Byte	BoardID	$0B
$0008	Byte	Dummy4	Dummy
$0009	Byte	BoardED	Board Vers. = <D7..D4>.<D3..D0>
$000A	Byte	Dummy5	Dummy
$000B	Byte	Mail	reserved
$000C	Byte	Dummy6	Dummy
$000D	Byte	Vector	reserved
$000E	Byte	Dummy7	Dummy
$000F	Byte	Control	Card-Control-Register By writing $08 into this register, the VME-PROFI will be enabled
$0010	Lword[252]		reserved
$0400	Byte	Command	reserved
$0401	Byte	Error	reserved
$0402	Word[6]	Prm	reserved
$040E	Word	Live	"live" will be increased while the VME-PROFI is running
$0410	Byte[256]	ReadBuf	reserved
$0510	Byte[256]	WriteBuf	reserved
$0610 -$070F	Word[128]	DataBuf	reserved
$0710	char [7]	Software	„DPS" = DP-Slave software
$0717	char	Version	SW-version = <D7..D4>.<D3..D0>
$0718	unsigned char	Station	PROFIBUS station address 0 .. 125

$0719	unsigned char	Status	„status" are set by the DP-Slave software as followed: $02 -> The DP-Slave software starts, the configuration data will be read. $10 -> New output data are available in the output buffer. $41 -> Timeout (Watchdog time exceeded) $81 -> Communication error „status" are set by the VMEbus main CPU as followed: $00 -> Status are cleared $01 -> The configuration data are valid See chapter 5.2.3.1
$071A	unsigned char	Baud	0 -> 9600 Bit/s 1 -> 19200 Bit/s 2 -> 93,7 kBit/s 3 -> 187,5 kBit/s 4 -> 500 kBit/s 5 -> 38,4 kBit/s 6 -> 1,5 Mbit/s 7 -> 3 Mbit/s
$071B	unsigned char	res	reserved
$071C	unsigned short	Ident	Ident-number 2 Byte (= $5851)
$071E	unsigned char	UparLn	length of user parameter
$071F	unsigned char	ConfLn	length of configuration data
$0720	unsigned char	InputLn	length of input data
$0721	unsigned char	OutputLn	length of output data
$0722 .. $0741	unsigned char	Uparam[32]	user parameter
$0742 .. $0761	unsigned char	Config[32]	configuration data Code as definied in DIN 19245 T3 8.3.5 D7 D6 D5 D4 D3 D2 D1 D0 | | | | |__|__|__|_ Number of data (N - 1) | | | | 00=1 Byte/Word | | | | 15=16 Byte/Word | | | | | | |__|_____Input/Output-code | | 00 = special format | | 01 = input | | 10 = output | | 11 = I/O-channel | | | |__0 = Byte | 1 = Word | |__consistency for 0 = Byte/Word 1 = whole number of data

$0762 ... $0861	DPR_Buffer	Inputs	Input-data buffer (VME-PROFI(DPS) -> DP-Master)
			DPR_Buffer consists of:
$0762	unsigned char	Sema	Sema must be increased before and after the VMEbus-CPU data access. With Sema, a consistency data access can be realized. See chapter 5.2.3.2
$0763	unsigned char	Lng	length of input data
$0764 .. $0861	unsigned char	Daten[254]	data
$0862 ... $0961	DPR_Buffer	Outputs	Output-data buffer (DP-Master -> VME-PROFI(DPS))
			DPR_Buffer consists of:
$0862	unsigned char	Sema	Sema will be increased before and after the VME-PROFI(DPS) data access. With Sema, a consistency data access can be realized. See chapter 5.2.3.2
$0863	unsigned char	Lng	length of output data
$0864 .. $0961	unsigned char	Daten[254]	data
$0962 - $FFFF			reserved

Table 9: Dualport RAM DP-Slave software

5. Software

The firmware for the VME-PROFI can be stored in the onboard Flash-EPROM´s.
The Firmware for the DP-Master (class1) and the DP-Slave protocols are available.
The board will be delivered with the following Firmware:

Order Code	Default installed firmware	note
VME-PROFI	DP-Master (class1)	
VME-PROF-A	"	without connector ST2
VME-PROF-S	DP-Slave	
VME-PROFSA	"	without connector ST2

Table 10 : Firmware - Order Code

The firmware can be changed or updated with a OS-9 utility program (see chapter 2.4).

In this manual the VME-PROFI with the DP-Master (class1) firmware are named VME-PROFI(DPM1) and the VME-PROFI with the DP-Slave firmware are named VME-PROFI(DPS).

5.1. PROFIBUS-DP Master

5.1.1. Configuration of the PROFIBUS

Before a PROFIBUS system could be used, the Master (DPM1) must get the information, which DP-Slaves are available for him, and how this Slaves are configurated. Therefore the PROFIBUS-address (0 .. 125), the type of Slave (-> ID-code), the configuration of the Slave (-> Cfg-Bytes), if necessary operating parameters (-> User-bytes) an the I/O-range in the DP-RAM must be told to the Master. In the parameterization and configuration phases of the PROFIBUS, the DP-Master sends this information (ID-code, Cfg-bytes, User-bytes) to the Slaves. Each DP-Slave compares its real configuration with the configuration data received from the DPM1. When verifying the configuration, the device type, format and the length of information as well as the amount of inputs/outputs have to be identical. Only if these tests are successful a DP-Slave is ready to enter the data transfer phase.

The configuration of the DP-Master can be done through 2 ways:

- Configuration with the internal Control- and Configuration program (see chapter 5.1.3)
  A Control- and Configuration program is included in the DPM1 Software of the VME-PROFI. To control this program, a VT100 Terminal (or PC with Windows terminal) must be connected at X1 of the VME-PROFI(DPM1). The configuration is done with the integrated configuration program. For each Slave, which is connected to the PROFIBUS, the address, ID, input-offset ... must be set with the configuration program (see following description). After setup of all Slave's parameter, the data is saved with „R" in the internal EEPROM. For testing the PROFIBUS-system, the output-data can directly be set and the input-data can be viewed at the terminal.
  After configuration and test, the terminal is no longer necessary. At the next start of the VME-PROFI(DPM1) the configuration data automatically will be read from the internal EEPROM.

- Configuration through the VMEbus (see chapter 5.1.4)
  The VME-PROFI(DPM1) can be configured through the VMEbus, independent of the EEPROM-configuration data. To enable this function, the option v = „VMEbus Parametrierung" must be set with the Control- and Configuration program.

5.1.2. PROFIBUS data transmission

After enabling of the VME-PROFI (see chapter 4.1), the Profibus-DP-Master software reads the PROFIBUS-parameter from the internal EEPROM (or this data are set from the VMEbus) and setup the Slaves. After that, the Profibus-DP-Master software starts the cyclic data transmission. The processor (68331 / PBM) sends the output data from the dualport RAM to the Slave-station and write the received input data from the Slave to the dualport RAM. This is done cyclic for all attached Slaves.

5.1.2.1. DP-RAM data interface

The VMEbus-CPU can access the Slaves inputs and outputs by reading or writing the I/O-range of the VME-PROFI(DPM1). The description of the Dualport-RAM addresses can be found in chapter 4.2.1

5.1.2.2. Cycle time of the PROFIBUS data transmission

The cycle time of the PROFIBUS data transmission are influenced by 3 factors. The highest value sets the actually possible cycle time.

1. The internal processing time in the VME-PROFI(DPM1). This time is independent from the baud rate and are approx. 1 ms for the Master and approx. 0,2 ms for each configured Slave.

2. The internal processing time of the Slave module. This is the time, which the Slave needs, before, the next data telegram can be proceed. This time can be found as „Min_Slave_Intervall" in the GSD-file of the Slave and are normally in the range of 1 (= 0,1 ms) to 50 (= 5 ms).

3. The transmission time on the PROFIBUS T_{DP_SR}. This time depends from the baud rate, the amount of input and output data, the number of Slaves, and the type of the PROFIBUS system (Mono-Master or Multi-Master System). Additionally the max. reaction time of the Slaves $maxT_{SDR}$ and therefore the selected Slot time T_{SL}. influence the transmission time. More informations for the calculation of the transmission time can be found in the DIN 19245 Part 3 Chapter 7.5.
   The following transmission time can be realized in a Mono-Master System with 500 kBit/sec.:

1 Input-byte / 1 Output-Byte	T_{DP_SR} = 1,9ms
240 Input-bytes / 240 Output-bytes	T_{DP_SR} = 23ms

   With the highest baudrate (3Mbit/sec) the transmission time can approx. be reduced by the factor 5.

5.1.3. Configuration program

5.1.3.1. Terminal connection

The terminal (VT100 or PC with Windows Terminal) has to be connected at X1 (see chapter. 2.1 and 3.2.2).

5.1.3.2. Screen display

The following hardcopy shows the normal configuration display.

The display is separated in 3 areas:

1. The upper part of the display shows the master parameters.
2. The middle part of the display shows the parameter and I/O-data of up to 3 Slave stations. To browse through additional Slaves, use the key „W".
3. The bottom part is reserved for reports and notes

```
+--------DMS - PROFIBUS-DP MASTER v3.02-------------------------------------+
! MA  : 00001 Baud: 1.5MB maxT: 00150 TSL.: 00400 TTR.: 00100 G...: 00010    !
! HSA.: 00099 MinS: 00020 Poll: 00050 DCT.: 00100 minT: 00011 Watch:01000    !
+-----------------------------------------------------------------------------+
! SA : 00006  SCHIELE PM -----  ->  -Run---------                             !
! ID : 0084  Gr:01    In:0000 Out:0800 UsrPrm:                                !
! Cfg: 01EB- 04EW- 01AB-                                                      !
! In : 00 00 00 00 00 00 00 00 00                                             !
! Out: 00                                                                     !
! SA : 00009  ET200B-24DI/8DO   ->  -Run--------                              !
! ID : 000F  Gr:01    In:0009 Out:0801 UsrPrm: 00 00 00 00 00                 !
! Cfg: 01AB- 03EB-                                                            !
! In : 00 00 00                                                               !
! Out: 00                                                                     !
!                                                                             !
!                                                                             !
!                                                                             !
!                                                                             !
!                                                                             !
!                                                                             !
+-----------------------------------------------------------------------------+
!  Confirmation - DP -                                                        !
! -> OK - Dienst fehlerfrei ausgefuehrt                                       !
+-----------------------------------------------------------------------------+
```

This display is shown after enabling of the VME-PROFI. If the VME-PROFI was enabled, before the terminal was connected or switched-on, only the cyclic updated parts are shown. With the key „Z" a whole screen update can bee done.

Note:
If the option v = „VMEbus Parametrierung" are set in a previous session, no screen update will occur. To get this configuration screen, the option „v" must be disabled by typing „GO".

5.1.3.3. Commands

M - Enter Master parameter

With key **M** the Master parameter can be changed. Select the parameter-data-field with the cursor keys and enter the data. The following parameters can be entered:

- MA PROFIBUS-address of the master station
  Value: 0 ... 125 (normally 1)
- Baud Baudrate
  Range: 0 ... 7 .

0 -> 9600 Bit/s	4 -> 500 kBit/s
1 -> 19200 Bit/s	5 -> 38400 kBit/s
2 -> 93,7 kBit/s	6 -> 1,5 Mbit/s
3 -> 187,5 kBit/s	7 -> 3 Mbit/s

- maxT Maximal Station Delay, max T_{SDR} [t_{Bit}]
  Range: 11 .. 255
  max T_{SDR} is the max. time used by the Slave before the requested data will be sent to the DP-Master. As maxT the greatest T_{SDR} of the connected Slaves should be set. max T_{SDR} depends of the used baud rate an can be found in the Slaves GSD-file.
  max T_{SDR} must be greater than minT_{SDR}.
- TSL Slot-Time T_{SL} [t_{Bit}]
  The Slot Time, is the timeout time for the DP-Master. After sending a telegram, the Master will wait for receiving a answer-, acknowledge- or token-telegram until the T_{SL} time is over. Select T_{SL} as followed:
  T_{SL} = maxT_{SDR}+ T_Z (T_Z = additional value for the signal transmission on the bus)
  Note:
  Due to a bug of the DPM1 Vers 3.03, the maximal possible master telegram length T_{TEL} must be added to the T_{SL} time. See chapter 5.1.7.
  T_{TEL} = (9 + <max. max. output length of a Slave>) * 11

max. bytes	T_{TEL} = Additional value for T_{SL}
10	209
32	451
244	2783

  Example: Baud rate = 500 kBit/s
  maxT_{SDR} = 100 (see max. MaxTsdr_500 from Slaves GSD-files)
  T_Z = 50 (Additional value for delay on the bus)
  Slave_a : 6 byte input, 4 byte output
  Slave_b : 2 byte input, 8 byte output
  Slave_c : 16 byte input, 10 byte output
  T_{TEL} = 209 (telegram length for 10 byte output)
  T_{SL} = maxT_{SDR} + T_Z + T_{TEL}
  T_{SL} = 100 + 50 + 209

The selected value for TSL must be ≥ the calculated T_{SL}
Selected : TSL = 400

- TTR Target-Rotation-Time T_{TR} [1000 * t_{Bit}]
  For Mono-Master-Systems the default value 100 (= 100 000 t_{Bit}) should be used.
  For Multi-Master-Systems a different value may be necessary, see DIN 19245 Part 1/2 for detailed info.
- G GAP-update factor
  The DP-Master will start a GAP-Update when the time TTR * G expired.
- HSA Highest-Station-Adress
  Highest address of another Master station.
- MinS Min-Slave-Interval [100 µs]
  Minimal time interval between two Slave scanning cycles.
  MinS must be ≥ the greatest „Min_Slave_Intervall" value from the Slaves GSD-files.
- Poll Poll-Timeout [1 ms]
  Timeuot for Master-Master communications.
- DCT Data-Control-Time [10 ms]
  The Master will sent his operating mode, two times per DCT-time, with a multicast-telegram to all Slaves.
- MinT minimal Station Delay min T_{SDR} [t_{Bit}]
  Range: 11 .. 255
  MinT sets the minimal time interval, the slave must wait, before his responding telegram are allowed. For Mono-Master-Systems (the VME-PROFI(DPM1) is the only active Master), the minimal value (=11) could be set. For Multi-Master-Systems a higher value could be neseccary.
- Watch Watchdog time [ms]
  In the parameterization phase, the Master send this time to his Slaves. The Slaves will switch back to their Reset-state, if they are not accessed at least once per watchdog interval.

For more information about the parameters see DIN 19245 part1 and part 3.

After changing, use key R to save the parameters and restart the DP-Master.

Note:
Some types of Slave stations didn't automatically recognize a changed baud rate. In case of a problem reset the Slave module (switch off / on).
Some type of Slaves didn't support all possible baud rates. For example 38400 kBit/s not supported. In case of a problem check the slaves documentation or select a other baud rate.

S - Add Slave

With the key „S" a new Slave-station (decentralized peripheral station) can be added to the VME-PROFI configuration-data. The station address SA must be entered directly. The other parameter can be entered on screen. Select the parameter-data-field with the cursor keys and enter the data. The following data is necessary:

ID: Ident-number of the used Slave-type (see technical data of the slave).

Gr: Group-number for broadcast messages, normally $01

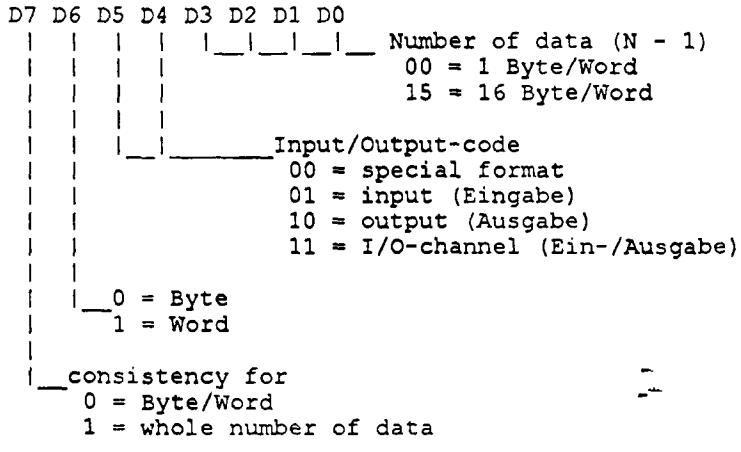
In: Offset for inputs - can also automatically be assigned with command „O"

Out: Offset for outputs - can also automatically be assigned with command „O"

UsrPrm: User-Parameter, if necessary (see technical data of the slave).
Max 8 User-bytes can be set. The number of user-bytes and their values must be read from the GSD-file. Normally the default user-bytes are shown in the GSD-file. For some Slaves, different bytes could be necessary (for example to set the slow / high-mode or to set the input-range of analog modules). Please check the manual of the Slaves.

Cfg: I/O configuration as specified in DIN 19245 part 3. See the technical data of the Slave-station or the GSD-file.
The configuration code must be entered as a 2 digit hex-code. Up to 32 Cfg-Bytes are possible. The first 12 Cfg-Bytes are shows at the configuration screen (converted to the nnCDs-format). For modular Slaves, the Cfg-byte (or bytes) for each module, corresponding to the installation, are necessary.

Meaning of the Hex-code:



This Hex-code is translated by the VME-PROFI in an easy to read form:

```
n  n  C  D  s
|  |  |  |  |__ - = no consistency
|  |  |  |       + = consistency
|  |  |  |
|  |  |  |____ B = Byte
|  |  |          W = Word
|  |  |
|  |  |__ S = special format
|  |        E = input (Eingang)
|  |        A = output (Ausgang)
|  |        I = I/O-channel (Ein-/Ausgang)
|  |
|__|__ Number of bytes / words
```

Note:
To remove a Cfg- or UsrPrm-byte, select the byte with the cursor and use key "-".
If more than 12 Cfg-bytes were entered, clear all bytes by „-", before Cfg-bytes are changed / added.

Examples:

Hex input	Display	comment
10	01EB-	for example: 8 digital inputs
11	02EB-	for example: 16 digital inputs
53	04EW-	for example: 4 analog inputs (1 word per input)
97	08EB+	for example: consistency input range 8 byte
61	02AW-	for example: output range 2 words

Optional a 16 character name can be assigned to each slave station by entering the name in the data-field behind the station-address "----------".

Note:
The Slave can only be paramized if the the master sends the correct ID-Number, and the Cfg- and UsrParm-data are valid for the actual configuration of the Slave.

The ID-number, Cfg- and UsrPrm data must be taken from the documentation or the GSD-file of the Slave. For many DP-Slaves the GSD-files can be red from a mailbox.

L - Remove Slave (Löschen)

With the key „L" a Slave-station can be removed from the VME-PROFI configuration-data.

O - Automatic assignment of the I/O-Address Offset

For each Slave station the address-range of the input and output data must be assigned in the 4 KB I/O-range of the VME-PROFI dualport RAM ($8F1000 - $8F1FFF see Table 8). The start-address of the input/output data is calculated as a Hex-offset value relative to the beginning of the VME-PROFI I/O-range ($8F1000). Entering of the input and output start-addresses can individually be done for each Slave by writing the value in the slave parameter data field.
Note: Be careful, that no overlapping input/output data occur.

With the key „O" the assignment of the I/O-offsets can automatically be done. The Inputs of the slaves are assigned in a consecutively order beginning at offset $0000 (= VMEbus address $8F1000) and the outputs beginning at offset $0800 (= VMEbus address $8F1800).

NOTE:
All Slave-station must have valid configuration data for the automatic offset assignment.
All manual assignments are overwritten by using the „O" command.

R - Storage of the configuration data and Restart

With the key „R" the configuration data are stored in the EEPROM of the VME-PROFI and the DP-Master software restarts. The key „R" is only once, at the end of configuration, necessary.

Note:
After the restart of the DP-Master software, the configuration screen is updated only once and shows the Slave-status from the first communication cycle.
If the Slave status field after Restart shows a „fail", update the screen once by pressing the <space> key to check the actual status.

V - Restart Parameter from the VMEbus Host

With the key „V" the software will be restart. The configuration data and the operation of the DP-Master are controlled from the VMEbus (see chapter 5.1.4). The display shows following picture.

```
+--------DMS - PROFIBUS-DP MASTER v3.02-----------------------------------------+
!                                                                               !
!                                                                               !
+-------------------------------------------------------------------------------+
!                                                                               !
!                                                                               !
!                                                                               !
!                                                                               !
!                                                                               !
!                                                                               !
!                                                                               !
!                                                                               !
!                                                                               !
!                                                                               !
!                                                                               !
!                                                                               !
!                                                                               !
!                                                                               !
!                                                                               !
+-------------------------------------------------------------------------------+
!   VMEprofi ist fuer VMEbus-Parametrierung bereit !                            !
! ->      lokale Parametrierung kann mit GO gestartet werden                    !
+-------------------------------------------------------------------------------+
```

The VME-PROFI(DPM1) is ready for remote parameterization and controlling through the VMEbus.
Type „GO" to return to the normal configuration display.

G - Get diagnostic data

With the key „G", the diagnostic data from the Slave (at the cursor position) can be red.

Byte 1:

D7	D6	D5	D4	D3	D2	D1	D0	
							1	Diag.Station_Non_Exist set from Master
						1		Diag.Statio_Not_Ready Slave is not ready for data transfer
					1			Diag.Cfg_Fault Wrong configuration data (Bad Parameter)
				1				Diag.Ext_diag If set, diagnostic data in the Slave specific diagnostic area (Ext_Diag_Data) are available. If not set, status data can be available in the Slave specific diagnostic area (Ext_Diag_Data)
			1					Diag.Not_Supported The Slave didn't support the requested function.
		1						Diag.Invalid_Slave_Response Set by the Master in case of a invalid response
	1							Diag.Prm_Fault Wrong parameters (for example wrong Ident-No, wrong Cfg-Bytes)
1								Diag.Master_Lock If byte 4 is not equal the own Master-address or $FF (= the Slave are configurated from a different Master), this bit will be set from the Master

Byte 2:

D7	D6	D5	D4	D3	D2	D1	D0	
							1	Diag.Prm_Req A new parameterization of the Slave is necessary
						1		Diag.Stat_Diag The Master must read Diagnostic data in case of a set bit.
					1			Always "1"
				1				Diag.WD_On Watchdog active
			1					Diag.Freeze_Mode Freeze command received
		1						Diag.Sync_Mode Sync command received
	1							reserved
1								Diag.Deactivated Set by the Master in case of a non active Slave

Byte 3 : D7 D6 D5 D4 D3 D2 D1 D0
| |__|__|__|__|__|__| reserved
|
|_____ Diag.Ext_Overflow
Overrun of Diagnostic Data

Byte 4 : D7 D6 D5 D4 D3 D2 D1 D0
|__|__|__|__|__|__|__|__| Diag.Master_Add
= <Address of Master> after parameterization
= $FF if the Slave is not parameterized

Byte 5 : D7 D6 D5 D4 D3 D2 D1 D0
|__|__|__|__|__|__|__|__| Identnummer High

Byte 6 : D7 D6 D5 D4 D3 D2 D1 D0
|__|__|__|__|__|__|__|__| Identnummer Low

Byte 7 ..32 (bzw 244) :
Extern Diagnostic Data see DIN 19245-3 or Slave dokumentation.

Note:
The Diagnostic Data are only valid, if the Slave is in „RUN" status.

W - Show additional Slaves

With key „W" (weitere) can be browsed to additional Slaves, if there are more than 3 Slaves .

Z - Display refresh back to first page

With key „Z" (zurück) you can jump back to page one and a display refresh occurs.
If option „V - Restart Parameter from the VMEbus Host" are set, not screen update will occur.

<Space> - Update display of input data

With the „**space**" key the displayed input data can be updated to show the actual data.

The display is not automaticly updated, while the DP-Master is running.

? - Help display

With the key „?" the help display (in German) are displayed.

```
+--------DMS - PROFIBUS-DP MASTER v3.02----------------------------------------+
! MA  : 00080 Baud:  500k maxT: 00150 TSL.: 01000 TTR.: 01000 G...: 00010     !
! HSA.: 00099 MinS: 00002 Poll: 00050 DCT.: 00100 minT: 00050 Watch:01000     !
+-----------------------------------------------------------------------------+
!                                                                             !
! Konfigurations-Funktionen:                                                  !
!   M - Master Parameter einstellen                                           !
!   S - Slave Konfiguration - Neuen Slave hinzufuegen                         !
!   L - Loeschen Slave                                                        !
!   O - Offset automatisch berechnen                                          !
!   R - Restart Master - Parameter uebernehmen und sichern                    !
!   V - Restart Master - Parameter vom VMEbus Host                            !
! Anzeige-Funktionen:                                                         !
!   G - Get Diagnose                                                          !
!   W - Weitere Slaves anzeigen                                               !
!   Z - Zurueck - Display erste Seite                                         !
!   ? - Display Help                                                          !
!                                                                             !
!                                                                             !
!                                                                             !
+-----------------------------------------------------------------------------+
!                                                                             !
! ->                                                                          !
+-----------------------------------------------------------------------------+
```

Use key „Z" to switch back to normal display.

5.1.3.4. Setting of outputs

For test all output of the slaves can be set by typing the Hex-data in the Slave-output-data-field.
This is only possible, if the output data is not set by the VMEbus-CPU!

Warning:
Never set output data with a running machine attached to the outputs!

5.1.4. Configuration of the PROFIBUS through the VMEbus

The configuration of the VME-PROFI(DMP1) could, alternative to the configuration with the internal Control- and Configuration program, also be done through the VMEbus.

To enable this way of configuration, the option „V - Restart Parameter from the VMEbus Host" must be set (see chapter 5.1.3.3). In case of this operationg mode, no configuration data are read from the EEPROM. Instead of that, the configuration data must be set through the VMEbus, every time the VME-PROFI(DPM1) starts. The controlling of the VME-PROFI(DPM1) are done through a command-interface in the DP-RAM with the program „`dpstart`".

`Dpstart` enables the VME-PROFI(DPM1), sets the configuration data, which are read from the file `autoload.dp1` and starts the cyclic data transmission.

The program `dpstart` are stored as objekt file for OS-9 systems and as ANSI-C source-code for other operating systems on the disk „VMEPROFI".

The file `autoload.vp1` contains all necessary configuration data in binary form. To create a `*.vp1` file, the PC-program `DPKONFIG` are available. This Windows95 program could read the GSD-files, extract all necessary data and automatically generates a configuration file. The program `DPKONFIG` could be order as separate product from DMS.

5.1.5. Demo programs

5.1.5.1. Ultra C (ANSI C)

Syntax: vpdemo s

s = slot number (1 .. 15 if slot depending addressing takes place)
(15 without slot depending addressing)

VPdemo.c

```
/* VME PROFI DEMO */

#include <stdio.h>
#include <process.h>
#include <signal.h>
#include <types.h>

#define VS_Base 0x800000

typedef unsigned char byte ;
typedef unsigned short wort ;
typedef unsigned int lwort ;

#include "../defs/VS.h"

/* Globale Variablen */
VS_Typ *DP ;

int main (int argc , char **argv)
{
int err,ret ;
u_int16 sink ;
process_id pid ;
byte *p ;
short n, s;
byte b ;
byte *in ;
byte *out ;

printf ("Demo program for VMEPROFI \n") ;
printf ("------------------------- \n\n") ;
s = atoi (argv[1]) ;
p = (byte*) (VS_Base + (0x10000 * s) ) ;
DP = (VS_Typ*) p ;

if ( (err = _os9_id(&pid,&sink,&sink,&sink)) ) /* Get ID */
   printf("Error _os9_id() %X\n",err);

if ((err = _os_permit ((void*)p,0x10000,0x777,pid)))
          /*VMEbus Standard 64 KB starting at address p */
    printf("Error _os_permit() %X \n",err);


DP->Control = 0x00 ;
sleep (1) ;
printf ("  Manufactor-Ident ...: %c%c%c\n", DP->ID1,DP->ID2,DP->ID3 );
printf ("  Board-Ident ........: %3d\n", DP->BoardID );
printf ("  Board-Version ......: %3X\n", DP->BoardED );
printf ("  Mail-Box-Register ..: %3X\n", DP->Mail );
printf ("  IRQ-Vector .........: %3X\n", DP->Vector );
printf ("  Control Register ...: %3X\n", DP->Control );

printf (" Enable \n") ;
DP->Error = 1
DP->Control = 0x08 ; ;              /* Enable VME-PROFI with $08 */
printf ("  Control Register ...: %3X\n", DP->Control );
```

```
sleep (1) ;
printf (" Wait for Lokal-CPU Start ...\n");
for (n=0;n<10;n++) if (DP->Error) sleep(1) ;
if (DP->Error)
  {
    printf (" *** Error : Local CPU did not start!\n") ;
    DP->Control = 0x00 ;
    return (0) ;
  } ;
printf (" Start \n") ;

in  = (byte*) DP + 0x1000 ;          /* read Input-data   */
out = in + 0x800 ;                   /* write Output-data */

b = 0x01 ;
while (1)
  {
     out[0] = b ;
     sleep (1) ;
     b = b << 1 ;
     if (b==0) b=0x01 ;
  } ;


return (0) ;
} /* END MAIN */

$
```

vs.h

```
typedef struct { byte dumy0  ;  byte ID1 ;
                 byte IRQSET ;  byte ID2 ;
                 byte IRQEN  ;  byte ID3 ;
                 byte dumy3 ;  byte BoardID ;
                 byte dumy4 ;  byte BoardED ;
                 byte dumy5 ;  byte Mail ;
                 byte dumy6 ;  byte Vector ;
                 byte dumy7 ;  byte Control ;
                 lwort LokalVec[252] ;
                 byte Comand ;
                 byte Error ;
                 wort Prm[6] ;
                 wort Live ;
                 byte ReadBuf[256] ;
                 byte WriteBuf[256] ;
                 wort DataBuf[128] ;
               } VS_Typ ;
```

5.1.5.2. Example for Berghof VME-S5 CPU

The VME-PROFI is independent from the used Operating System of the VMEbus main CPU, because no PROFIBUS-software must be loaded. Only the VME-PROFI must be enabled by writing a $08 into the Control register.

The following example shows the setup for the S5 compatible VMEbus CPU from Berghof:

1. Create the following file

```
                                                        ( Comments )
 0:     KC ='MASKX1';
 3:     KH = 8009;     Grundeinstellungen 9  Worte
 4:     KH = 1001;     Slot1 Arbitermode ein            (enable Slot1 function)
 5:     KH = 2000;     VMEzugriffe im NonPrivilegd Mode
 6:     KH = 3000;     Serdat nicht verwenden
 7:     KH = 4000;     keine Userdef. Adressmodifier
 8:     KH = 5000;     Semaphoren gesperrt
 9:     KH = 6000;     Sysreset set Stop
10:     KH = 7000;     keine Rueckwand PG-Schnittstelle
11:     KH = 8000;     SysFail nicht beachten
12:     KH = 9000;     Urloeschen erlaubt
13:     KH = 8101;     Interruptverhalten 1 Wort
14:     KH = 1000;     gesperrt
15:     KH = 8210;     VME-Address-Tab 16 Worte
16:     KH = 5000;     P-Bereich ab Eingangsbyte 0      (P-range input bytes)
17:     KH = 448F;     VME-Standard 0x8F0000            (VME-PROFI base address $8F0000)
18:     KH = 1000;     Offset 0x1000                    (Offset Input data)
19:     KH = 0180;     Schrittweite 1, Anzahl 128 Byte  (step 1, Number of bytes 128)
20:     KH = 7000;     p-Bereich ab Ausgangsbyte 0      (p-range output bytes)
21:     KH = 448F;     VME-Standard 0x8F0000            (VME-PROFI base address $8F0000)
22:     KH = 1800;     Offset 0x1800                    (Offset Output data)
23:     KH = 0180;     Schrittweite 1, Anzahl 128 Byte  (step 1, Number of bytes 128)
24:     KH = 71FF;     q-Bereich Ausgangsbyte 255       (q-range)
25:     KH = 448F;     VME-Standard 0x8F0000
26:     KH = 000F;     VMEProfi Enable Register
27:     KH = 0101;     Schrittweite 1, Anzahl 1 Byte
28:     KH = 51E0;     Q-Bereich Eingangsbyte 224       (Q-range)
29:     KH = 448F;     VME-Standard 0x8f0000
30:     KH = 2000;     Stations Status
31:     KH = 0110;     Schrittweite 1, Anzahl 16 Byte   (128 Status bits)
32:     KH = EEEE;     Ende
```

2. Add the following 2 commands in the Organisations Baustein 20 (OB20)

```
                                                        ( Comments )
     :L  KH 0008                                        (load $08)
     :T  QB 255                                         (transfer to Q-range byte 255)
```

3. At startup of the VME-S5, the two commands in OB20 are executed, which enable the VME-PROFI. Now the decentralized I/O can be accessed through the PLC P/p-range.

5.1.5.3. Example fo a ADEPT-System

The slot depending addressing of the VME-PROFI through the J2-bus, can't be used, because this J2-signals are used in a ADEPT-rack. To avoid problems, the VME-PROFI should be ordered without J2 connector. See chapter 1.5.

The following example shows, how the VME-PROFI could be integrated in a ADEPT-system.

```
PROGRAM a.profi()
; Test Program to integrate DMS Profi Board into AdeptMV Controller
; IMPORTANT: Before you are able to enable the robot power you have to run
; this program in a task other than 0. Only if the DMS board is enable SYSFAIL
; is deasserted
CALL pr.init()              ; Initialize addresses
CALL pr.def.dio()           ; Map Adept Signals to DMS Board
CALL pr.check.board()       ; Check if DMS board is present
CALL pr.enable()            ; Enable DMS board
.END


.PROGRAM pr.check.board()
; Reads the ID section of the DMS board. If it does not contain specific
; information at a certain address it is assumed the board is not available.
$id = $CHR(IOGETB(pr.id1.adr,1))
$id = $id+$CHR(IOGETB(pr.id2.adr,1))
$id = $id+$CHR(IOGETB(pr.id3.adr,1))
pr.id = IOGETB(pr.id4.adr,1)
IF ($id <> "DMS") OR (pr.id <> ^HB) THEN
    TYPE "No DMS board found"
    HALT
ELSE
    TYPE "DMS VME-Profi found at ", /H6, pr.base.adr
END
.END


.PROGRAM pr.def.dio()
; Map V+ input and output signals to DMS VME Profi board
LOCAL io.offset, io.sig

; Map input signals do DMS board
io.offset = 0                                   ; Start at offset 0
FOR io.sig = 1017 TO 1505 STEP 8                ; Map the input signals
                                                ; you want to redirect
                                                ; to the input section
                                                ; of the DMS board
    DEF.DIO io.sig = pr.in.adr+io.offset, 1     ; Map 8 signals
                                                ; starting at signal
                                                ; io.sig to VME address
    io.offset = io.offset+1                     ; Increase VME address by 1 Byte
END

; Map output signals do DMS board
io.offset = 0                                   ; Start at offset 0
FOR io.sig = 17 TO 505 STEP 8                   ; Map the ouput signals
                                                ; you want to redirect
                                                ; to the output section
                                                ; of the DMS board
    DEF.DIO io.sig = pr.out.adr+io.offset, 1    ; Map 8 signals
                                                ; starting at signal
                                                ; io.sig to VME address
    io.offset = io.offset+1                     ; Increase VME address by 1 Byte
END
RETURN

.END
```

```
.PROGRAM pr.enable()
; Enable DMS VME Profi
IOPUTB pr.control.adr, 1 = ^H8          ; Write $08 to control register
RETURN
.END



.PROGRAM pr.init()
; Initialize addresses for communication with DMS VME Profi board
pr.base.adr = ^H130000           ; VME Profi base address
; c = closed      '0'
; o = open        '1'
;
; J30     A23   A22   A21   A20 |  A19   A18   A17   A16
;(Pin)    1-2   3-4   5-6   7-8 |  9-10 11-12 13-14 15-16
;          c     c     c     o  |   c     c     o     o
;          0     0     0     1  |   0     0     1     1
; P2 Pin assignment              |  a4    c4    a2    c2
;                                |  GND   GND   NC    NC
;
; The Adept MV8 and MV19 controllers use pins a4 and c4 on the P2 connector.
; DMS use pins a2, a4, c2 and c4 for address decoding purposes. If the board
; is installed inside a standard MV 8 or 19, address A16 to A19 should not be
; configured with jumpers but instead left open. A20 to A23 may be used freely.
;
; Note (DMS/RC):
;      from DMS, the VME-PROFI can also be ordered without J2-connector
;      Order-code 'VME-PROF-A'. With this version of the board, no address
;      conflict occur and A19 - A16 can be selected freely with jumpers.
;
; Attention ! V+ allows access only up to address $7FFFFF so A23 must be
; closed ('0'). Valid addresses are therfore $x30000 with 1<=x<=7.
; If AdeptVision is installed inside an AdeptMV cotnroller the valid addresses
; are in the range of $0 - $3FFFFF
; If a  Dual Vision System is used there is no space left for 3rd party boards
; at all.

pr.id1.adr = pr.base.adr+^H1           ; Contains "D"
pr.id2.adr = pr.base.adr+^H3           ; Contains "M"
pr.id3.adr = pr.base.adr+^H5           ; Contains "S"
pr.id4.adr = pr.base.adr+^H7           ; Contains $0B

pr.control.adr = pr.base.adr+^HF ; $08 -> Enable Profibus

pr.alive.adr = pr.base.adr+^H40E ; Board Alive section
pr.in.adr = pr.base.adr+^H1000         ; Digital Input section
pr.out.adr = pr.base.adr+^H1800        ; Digital Output section
RETURN

.END
```

5.1.6. Difference to Profibus-DP-Master Software Version 1.x, V2.x, V3.x

	Version 3.02	Version 2.1	Version 1.x
DP-Master 1	yes	yes	yes
DP-Master 2	no	no	no
DP-Slave	no	no	no
Mono-Master-System	yes	yes	yes
Multi-Master-System	yes	no	no
max. number of Slaves	124	124	20
amount of I/O-Data	max. 244 Bytes per Slave max. 4 KB for all Slaves	max. 244 Bytes per Slave max. 4 KB for all Slaves	max. 32 Input Byte and max. 32 Output byte per Slave
supported Slaves	Slaves according to DIN 19245 Part 3	Slaves according to DIN 19245 Part 3	Slaves according to DIN 19245 Part 3 and old Siemens L2 devices (DP-Siemens protocol)

Table 11: Comparison of Profibus-DP-Master Vers. 1.x, 2.x to Vers. 3.x

5.1.7. Known bugs of the DP-Master Software

DPM1 Vers. 3.03: TSL-time

The Slot Time, is the timeout time for the DP-Master. After sending a telegram, the Master will wait for receiving a answer-, acknowledge- or token-telegram until the T_{SL} time is over. The Master will continue with his data transmission, if this period expired. Telegrams arriving later will not be accepted from the Master. The TSL period normally starts at the end of the Master telegram.

With the actually version of the DP-Master, the T_{SL} period starts counting at the begin of the Master telegram. To compensate this error, the length of the longest telegram (in T_{Bit}) should be added to the TSL-value. See chapter 5.1.3.3.

5.2. PROFIBUS-DP-Slave Software

5.2.1. Terminal interface

From the DP-Slave Software this interface is only used for debug information's.

5.2.2. Slave configuration

For the DP-Slave the following parameter must be written into the dualport RAM and the input / output buffers must be initialized, before the VME-PROFIE is enabled by writing $01 to „Status“:

- `Station` PROFIBUS station address (0 .. 125)
- `Baud` Baudrate
- `Ident` Ident-number
- `UparLn` Length of user parameter
- `ConfLn` Length of configuration data
- `InputLn` Length of input data
- `Uparam[32]` User parameter
- `Config[32]` Configuration data

Station

Any address in the range of 0 .. 125 can be selected as station address. Normally the lowest addresses are reserved for Master stations.

For Example: $00 DP-Master(Klasse2) (= Programming unit)
$01 DP-Master(Klasse1)
$02 DP-Master(Klasse1)

Baud

Select 0 .. 7 as baud rate, corresponding to the following coding:

0 -> 9600 Bit/s	4 -> 500 kBit/s
1 -> 19200 Bit/s	5 -> 38,4 kBit/s
2 -> 93,7 kBit/s	6 -> 1,5 Mbit/s
3 -> 187,5 kBit/s	7 -> 3 Mbit/s

Note:
The VME-PROFI(DPS) did not automatically select the used baud rate!

Ident

$5851 is the PNO Ident-number for the VME-PROFI(DPS). If the VME-PROFI(DPS) is used to simulate another DP-Slave type, the ID-Code of this simulated Slave can be set.

Config[xx]

The Configuration can consists from up to 32 configurations-bytes. The resulting input length must be ≤ 243 byte and resulting output length must be ≤ 244 byte. Together up to 487 I/O-bytes could be configured.

InputLn, OutputLn

The value of InputLn and OutputLn must match the configuration. In case of a wrong input length, a wrong amount of data are send by the VME-PROFI(DPS). In case of a wrong output length, an error will be reported from the VME-PROFI(DPS). In both cases the Master will cyclically reconfigure the VME-PROFI(DPS) or a communication error (Status = $81) occors.

Uparam[32]

The VME-PROFI(DPS) didn't need user-bytes (= Uparam). In case of a simulated DP-Slave, user-bytes can be send from the Master. This bytes are ignored from the VME-PROFI(DPS).

5.2.3. DP-RAM data interface

The input and output data of the VME-PROFI(DPS) can be accessed trough the dualport RAM. See chapter 4.2.2 for the address range.

5.2.3.1. Status- Variable

„status" are set by the DP-Slave software as followed:

$02 -> The DP-Slave software starts, the configuration data will be read. If „status" is set to $02 and didn't change, an error had occurred (for example wrong configuration data, wrong length of configuration data).

$10 -> New output data are available in the output buffer.

$41 -> Timeout (Watchdog time exceeded).
The data in the output-buffer are reset to $00 by the DPS-software.

$81 -> Communication error.
The data in the output-buffer are reset to $00 by the DPS-software.

„status" are set by the VMEbus main CPU as followed:

$00 -> Status are cleared.
After reading the output data, „status" should be cleared (= $00).

$01 -> The configuration data are valid.
After setting $01, the DPS-software starts and sets status=$02.

Note:
If the VME-PROFI(DPS) configuration includes inputs, but no outputs, the variable „status" will not be set during the regular operating of the VME-PROFI(DPS), because no new output data are available. If the operating mode of the VME-PROFI(DPS) should be checked, one dummy output byte must be configured.

5.2.3.2. Sema-Variable within the input- and output buffer

A semaphon control byte „sema" are included in the input- and output-data-buffer.

Output-Sema:

Sema will be increased before and after the VME-PROFI(DPS) data access. Sema must be set to $00 when the output buffer is initialized.

Input-Sema:

The DPS-software will only accept new input data, if Sema are even and didn't change, while the data were read.
Sema must be set to $00 when the input buffer is initialized.

5.2.3.3. Consistency data access

The consistency data transfer can be monitored through the semaphon control byte „Sema".
A consistency data access is necessary, if more than 2 bytes (= 1 word) must be processed as a block of data. For example a 3 byte (=24 bit) angle value. Bytes or words always handled consistency, without usage of sema.

Reading consistency output data:

The VMEbus-CPU must read sema two times, once before the output data are read and second after the output data are read.
The output data can be used, if both sema-values are even numbered and equal (i.e. no DP-Slave data transfer occurs in the mean time).

Writing of consistency input-data:

The VMEbus-CPU must increase sema (-> odd value), write the data and had to increase sema again (-> even value).

A ANSI C demo program with consistency data access can be found in chapter 5.2.4.1.

5.2.3.4. Data access without consistency

The input and output data in the DP-RAM can directly be read and written, if no consistency data access necessary.
In this case, input sema must have a constant, even value. Otherwise, the DPS-software would not read the input data.

5.2.4. Demo program

5.2.4.1. Ultra C (ANSI C)

Syntax: vpsdemo s

s = slot number (1 .. 15 if slot depending addressing takes place)
(15 without slot depending addressing)

vpsdemo.c

```
/* VME PROFI Slave DEMO with consistency data access)*/
/*            13.09.96      RC  */

#include <stdio.h>
#include <module.h>
#include <signal.h>
#include <types.h>
#include <process.h>


#define VS_Base 0x800000


typedef unsigned char byte ;
typedef unsigned short wort ;
typedef unsigned int lwort ;

#include "../defs/VPS.h"
#include "../dps/DPR_Sub.c"


/* Global variables */
VS_Typ *DP ;



int main (int argc , char **argv)
{
int err,ret;
u_int32 st ;
u_int16 sink ;
process_id pid ;
byte *p ;
short n, s;
byte b ;

int counter ;
unsigned char outbuf[16] ;

printf ("Demoprogramm for VMEPROFI Slave\n") ;
printf ("------------------------------ \n\n") ;
s = atoi (argv[1]) ;      /* s = Slot Number    1 .. 15  */
p = (byte*) (VS_Base + (0x10000 * s) ) ;
DP = (VS_Typ*) p ;

if ( (err = _os9_id(&pid,&sink,&sink,&sink)) ) /* Get ID */
      printf("Error _os9_id() %X\n",err);

if ((err = _os_permit ((void*)p,0x10000,0x777,pid)))
          /*VMEbus Standard 64 KB starting at address p */
    printf("Error _os_permit() %X \n",err);


DP->Control = 0x00 ;
sleep (1) ;
```

```
printf ("  Manufactor-Ident ...: %c%c%c\n", DP->ID1,DP->ID2,DP->ID3 );
printf ("  Board-Ident ........: %3d\n", DP->BoardID );
printf ("  Board-Version ......: %3X\n", DP->BoardED );
DP->Error = 1 ;
DP->Control = 0x08 ;                    /*  Enable VME-PROFI  */
sleep (1) ;
printf (" Wait for Lokal-CPU start ...\n");
for (n=0;n<10;n++) if (DP->Error) sleep(1) ;
if (DP->Error)
  {
      printf (" *** Error : Local CPU did not start !\n") ;
      DP->Control = 0x00 ;
      return (0) ;
  } ;
printf (" --> Start O.K.\n") ;
printf ("  Software ...........: %s\n", DP->Software );
printf ("  SW-Version .. ......: %1X\n", DP->Version );

DP->Station   = 20 ;    /* Profibus Stations Address (0 .. 125) */
DP->Ident     = 0x5851; /* PNO Ident Nummer for the VME-PROFI(DPS) = 0x5851      */
                        /* or ID-Number of simulated slave                       */
DP->Baud      = 4 ;     /* Baudrate ( 0 / 1  / 2  / 3   / 4 / 5  / 6  / 7 )     */
                        /*          = 9,6/19,2/93,7/187,5/500/38,4/1500/3000kBit/s*/
DP->UparLn    = 0 ;     /* User Parameter Length (0 .. 32) */
DP->ConfLn    = 02 ;    /* Config Length (1 .. 32)*/
DP->Config[0] = 0x13 ;  /* 4 Byte Input Data to Master */
DP->Config[1] = 0x2f ;  /* 16 Byte Output Data from Master */
DP->InputLn   = 4 ;     /* Input Data Length. Must match the amount of input data*/
DP->OutputLn  = 16 ;    /* Output D. Length. Must match the amount of output data*/

DPR_Init (&DP->Inputs ,DP->InputLn) ;
DPR_Init (&DP->Outputs,DP->OutputLn) ;
DP->Status = 1 ;        /* Freigabe */
                        /* Status = 1  -> the configuration data are valid  */
                        /* the DPS software can start                       */

counter = 0 ;

while (1)
  {
   counter++ ;
   DPR_Write (&DP->Inputs, (unsigned char*) &counter) ;
   while ( DPR_Read  (&DP->Outputs, &outbuf[0]) ) ;
   printf ("\nStatus : %02X",DP->Status) ;
   if (DP->Status & 0x10)       /* Status = 10  -> new Output data avialable */
     {
        printf ("\nOutput :");
        for (n=0;n<16;n++) printf (" %02X",outbuf[n]) ;
        DP->Status = 0 ;
     } ;
     sleep (1) ;
 -} ;

return (0) ;
} /* END MAIN */
```

vps.h

```
/* Slave DPR */

typedef struct {
    unsigned char Sema ;
    unsigned char Lng ;
    unsigned char Daten[254] ;
              } DPR_Buffer ;


typedef struct { byte dumy0  ;  byte ID1 ;
                 byte IRQSET ;  byte ID2 ;
                 byte IRQEN  ;  byte ID3 ;
                 byte dumy3 ;  byte BoardID ;
                 byte dumy4 ;  byte BoardED ;
                 byte dumy5 ;  byte Mail ;
                 byte dumy6 ;  byte Vector ;
                 byte dumy7 ;  byte Control ;
                 lwort LokalVec[252] ;
                 byte Comand ;
                 byte Error ;
                 wort Prm[6] ;
                 wort Live ;
                 byte ReadBuf[256] ;
                 byte WriteBuf[256] ;
                 wort DataBuf[128] ;
             char Software[7] ;
             char Version ;
    unsigned char Station ;
    unsigned char Status ;
    unsigned char Baud ;
    unsigned char Blink ;
    unsigned short Ident ;
    unsigned char UparLn ;
    unsigned char ConfLn ;
    unsigned char InputLn ;
    unsigned char OutputLn ;
    unsigned char Uparam[32] ;
    unsigned char Config[32] ;
    DPR_Buffer Inputs ;
    DPR_Buffer Outputs ;

              } VS_Typ ;
```

dpr_sub.c

```
/* DPR with consistency data access */


void DPR_Init ( DPR_Buffer *DB , unsigned char Length )
{
unsigned char n ;
    DB->Sema = 0 ;
    DB->Lng  = Length ;
    for (n=0; n < DB->Lng; n++)
         DB->Daten[n] = 0 ;
}

void DPR_Write ( DPR_Buffer *DB , unsigned char * from )
{
unsigned char n ;
    DB->Sema++ ;
    for (n=0; n < DB->Lng; n++)
       {  DB->Daten[n] = *from ;
          from++ ;
        } ;
    DB->Sema++ ;
}


/* DPR_Read copies data from VME-PROFI dualport RAM (DPR) -> user program
   repeat, until successful (constant, even-numbered „sema"):
     while (DPR_Read(&DPR,&Buffer)) sleep(1) ; */

char DPR_Read ( DPR_Buffer *DB , unsigned char * to )
{
unsigned char Last ;
unsigned char n ;
    Last = DB->Sema ;
    if (Last & 1) return 1 ;
    for (n=0; n < DB->Lng; n++)
       {  *to = DB->Daten[n] ;
          to++ ;
        } ;
    if (Last == DB->Sema ) return 0 ;
    else return 1 ;
}
```

5.2.5. Known bugs of the DP Slave software

DPS Vers. 1.0: Behavior of the VME-PROFI(DPS) after a Multicast-telegram

The first telegram, following a Multicast telegram, will not be accepted by the VME-PROFI(DPS) and therefore one update of the slaves-data are missing. All other data cycles will be handled in the normal way.
A Multicast telegram will occur cyclically 2 times within the DCT-time. With this telegram, the DP-Master sends his operating mode to all slaves. In praxis, this could cause one missing data cycle every 5 seconds, if a 10 sec. the DCT-time is chosen (DCT = 1000).

As a result of the missing data update after the Multicast telegram, the diagnostic bits „Diag.Station_Non_Exist" and „Diag.Deactivated" can be set by the DP-Master.

6. Service and Maintenance

The VME-PROFI card does not require any special maintenance.

In case of a fault all card settings (jumpers and software setup) should be checked. Should you be unable to find the defect the card needs to be wrapped up carefully (packaging needs to be protected against electrostatic discharge) and ship it back to DMS. All returns must contain a detailed fault report (see Appendix Section 'DMS Return Form').

6.1. Troubleshooting

In this chapter you will find solutions for problems which may occur with the VME-PROFI.

6.1.1. General troubleshooting

Problem: The dualport-RAM of the VME-PROFI can't be accessed, or appears in a wrong address range.

Possible fault	Solution
The VMEbus-Slave address range is slot depending addressed. The address bits A19 ..A16 are set through the jumper J30 and/or the J2 connector (A2, C2, A4, C4).	Check the signals at connector J2(not connected or slot depending code) If necessary, the VME-PROFI can be ordered without slot depending addressing (without connector J2)
Wrong address setting	Check the address setting see chapter 2.2.1

Problem: Errors occur with selected higher baud-rates

Possible fault	Solution
Baud-rate to high for the used slave.	Select the baud-rate according to the slowest slave module.
Bus-cable, bus-termination or wrong installation of the bus-cable (to long stubs).	Select the cable type according to the used baud rate, bus-termination at both ends of the bus, installation without stubs. See chapter 3.2.1.1.

6.1.2. VME-PROFI(DPM1) troubleshooting

Problem: No configuration screen appears at the vt100 Terminal. The red LED „VMEbus/F" is on.

Possible fault	Solution
The VME-PROFI(DPM1) is not enabled (default state after SYSRESET).	Enable the board, see chapter 4.1.

Problem: After the Restart of the VME-PROFI(DPM1), appears the „wait config timeout" at the configuration screen. The green LED PROFIBUS/R is on.

Possible fault	Solution
Signals A / B or the Pull-up / Pull-down resistors of the bus termination are exchanged.	Check the bus cable and the bus termination.

Problem: One or more slave didn't answer after changing of the master-parameter.

Possible fault	Solution
Some Slave-modules can't recognize a changed baud rate.	Reset the slave-module (switch off/on).

Problem: No communication to a Slave module.

Possible fault	Solution
Some Slaves could use different bus protocols. For example the DP-NORM or DP-SIEMEN protocol of the ET200U (IM318B12) can be set with a DIO-switch.	Select the PRPFIBUS-DP protocol (=DP-NORM).

Problem: It's not possible to configure the Slave.

Possible fault	Solution
Wrong ID-, Cfg- and UsrPrm-data.	The Slave can only be paramized if the Master sends the correct ID-Number and the Cfg- and UsrPrm-data are valid for the actual configuration of the slave. The ID-number, Cfg- and UsrPrm data must be taken from the documentation or the GSD-file of the slave. See chapter 5.1.3.3
The bus-address of the slave is wrong.	Check the address, if necessary correct the address and Reset the slave (switch off/on).

Problem: A Cfg- or UserParam-byte can't be deleted at the configuration screen

Possible fault	Solution
-	To remove a Cfg- or UsrPrm-byte, select the byte with the cursor and use key "-".

Problem: More than 12 Cfg-bytes are necessary to configure a Slave

Possible fault	Solution
The internal Control- and Configuration program supports only 12 Cfg-bytes.	Only the first 12 Cfg-bytes are shows on the configuration screen. Internal up to 32 bytes could be processed. To edit more than 12 Cfg-bytes, first clear all Cfg-bytes of this Slave with „-" and than type in all Cfg-bytes (up to 32 bytes). Alternative with option „V - Restart Parameter from the VMEbus Host" more than 32 bytes are possible.

6.1.3. VME-PROFI(DPS) troubleshooting

Problem: No reaction from the VME-PROFI(DPS). The red LED „VMEbus/F" is on.

Possible fault	Solution
The VME-PROFI(DPS) is not enabled (default state after SYSRESET).	Enable the board, see chapter 4.1.

Problem: No communication to the VME-PROFI(DPS) after changing of the baud rate

Possible fault	Solution
The VME-PROFI(DPS) didn't automatically search the baud rate. The baud rate value must be set in the dualport-RAM of the VME-PROFI(DPS).	Correct the baud rate value in the dual-port-RAM of the VME-PROFI(DPS) and restart the VME-PROFI(DPS).

Problem: Not all data from the VME-PROFI(DPS) are transferred, the update rate is low, the LED PROFIBUS/F is flashing, Status = 81

Possible fault	Solution
The `InputLn` or `OutputLn` in the dualport-RAM of the VME-PROFI(DPS) is wrong. Due to the wrong amount of data (invalid response of the Slave) the Master will reconfigure the Slave again and again	Correct the `InputLn` / `OutputLn`

Problem: The DPS-software did not read the input data. Output data are transferred correct.

Possible fault	Solution
Input sema are wrong. Iinput sema must have a constant, even value. Otherwise, the DPS-software would not read the input data.	Set sema to $00. See chapter 5.2.3.4.

7. Diagrams and Appendix Section

7.1. Summary of Abbreviations

AM Code	Address Modifier Code
ASCII	American Standard Code for Information Interchange
CPU	Central Processing Unit
DMS	DMS Dorsch Mikrosystem GmbH
DPM1	DP-Master Class 1 (PROFIBUS DP controller for data exchange with DP-Slaves)
DPM2	DP-Master Class 2 (PROFIBUS DP controller for programming, configuration and diagnostics)
DPS	PROFIBUS DP Slave
DTB	Data Transfer Bus
EC	European Community
EEPROM	Electrical Erasable Programmable read-only Memory
EPROM	Erasable Programmable Read-only Memory
ESD	Electrostatic Discharge
Flash	Electrical erasable Memory
GAP	Address range beginning from the own address +1 up to the next active Master
GAP-update	During an GAP-update cycle, the Master checks all addresses (one address after each communication cycle) within his GAP range, whether a new Master appears, how want to be included into the token cycle.
GSD	Geräte-Stammdaten-Datei, device database information.
ID	Identifier
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronic Engineers
KB	Kilobyte, 1 KB = 1.024 Bytes
LED	Light Emitting Diode, Leuchtdiode
MS-DOS	Microsoft Disk Operating System
n.c.	not connected
OS-9	Multi-user and Multitasking Operating System by Microware
PNO	PROFIBUS user organization
PROFIBUS	a standardized field bus, specified in DIN 19245 / pr EN50170
PROFIBUS-DP	PROFIBUS for decentralized peripheral, DIN 19245 Part 1 + 3
PROFIBUS-FMS	PROFIBUS with Fieldbus Message Specification protocol DIN 19245 Part 1 + 2
VITA	VMEbus International Trade Association
VMEbus	Versa Module Eurocard Bus System
VME-PROFI(DPM1)	VME-PROFI with DP-Master Class 1 firmware
VME-PROFI(DPS)	VME-PROFI with DP-Slave firmware

7.2. References

[1] DMS Catalog, Dorsch Mikrosystem GmbH

[2] The VMEbus Specification (ANSI/IEEE STD 1014-1987), VMEbus International Trade Association

[3] PROFIBUS Norm DIN 19245 Part 1,2,3

7.3. Pin Layout ST1

Pin No.	Line A	Line B	Line C
1	VD0	n.c.	VD8
2	VD1	n.c.	VD9
3	VD2	n.c.	VD10
4	VD3	-> 5b	VD11
5	VD4	-> 4b	VD12
6	VD5	-> 7b	VD13
7	VD6	-> 6b	VD14
8	VD7	-> 9b	VD15
9	GND	-> 8b	GND
10	n.c.	->11b	SYSFAIL*
11	GND	->10b	BERR*
12	DS1*	n.c.	SYSRESET*
13	DS0*	n.c.	LWORD*
14	WRITE*	n.c.	AM5
15	GND	n.c.	VA23
16	DTACK*	AM0	VA22
17	GND	AM1	VA21
18	AS*	AM2	VA20
19	GND	AM3	VA19
20	IACK*	GND	VA18
21	IACKIN*	n.c.	VA17
22	IACKOUT*	n.c.	VA16
23	AM4	GND	VA15
24	VA7	n.c.	VA14
25	VA6	IRQ6*	VA13
26	VA5	IRQ5*	VA12
27	VA4	IRQ4*	VA11
28	VA3	IRQ3*	VA10
29	VA2	IRQ2*	VA9
30	VA1	IRQ1*	VA8
31	n.c.	n.c.	n.c.
32	VCC	VCC	VCC

Table 12: Pin Layout ST1

* Low active signal
-> connected with Signal xx
n.c. not connected

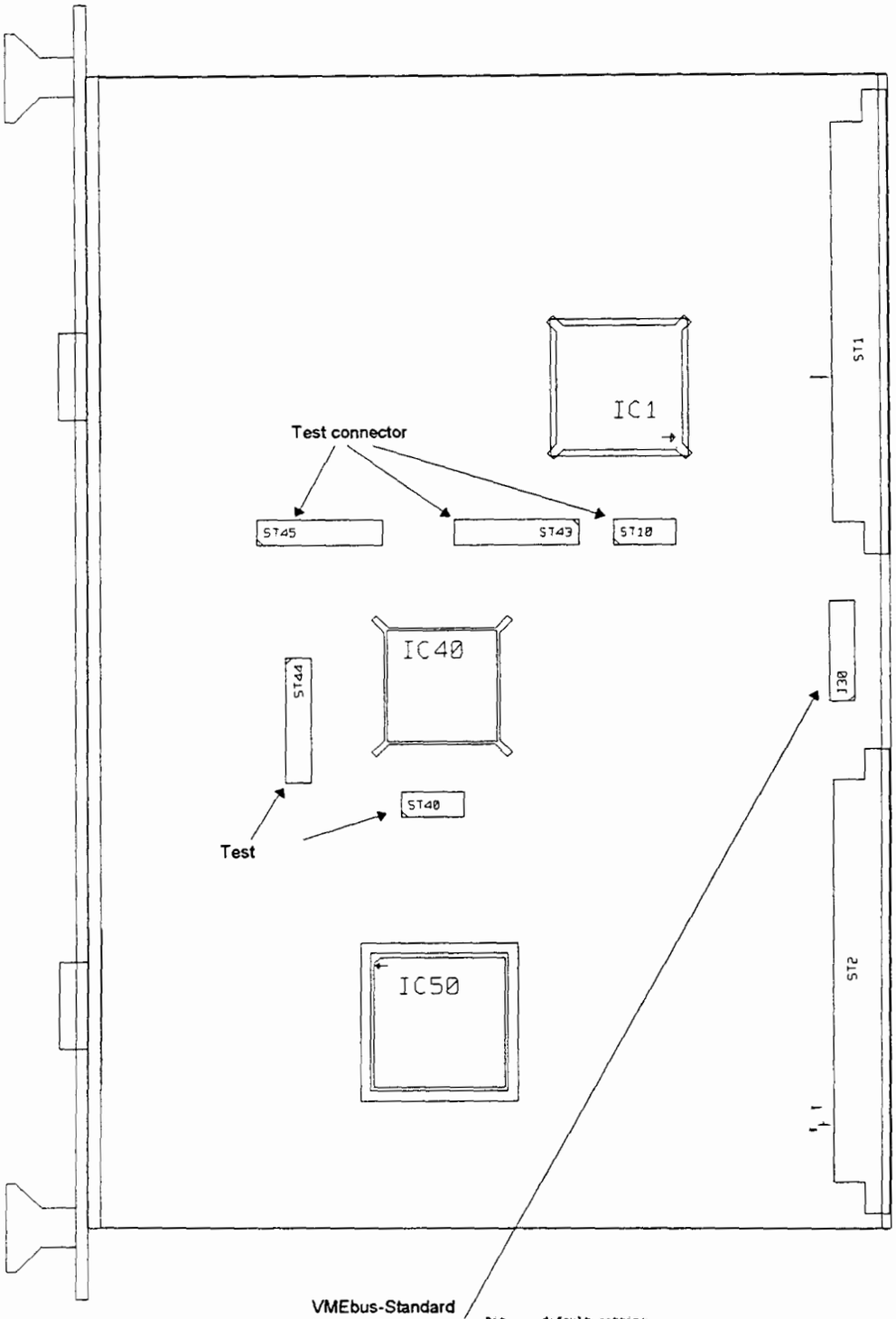
7.4. Pin Layout ST2

Pin No.	Line A		Line C
1	GND		GND
2	SL.3 (A19)		SL.2 (A18)
3	GND		GND
4	SL.1 (A17)		SL.0 (A16)
5	n.c.		n.c.
6	n.c.		n.c.
7	n.c.		n.c.
8	n.c.		n.c.
9	n.c.		n.c.
10	n.c.		n.c.
11	n.c.		n.c.
12	n.c.		n.c.
13	n.c.		n.c.
14	n.c.		n.c.
15	n.c.		n.c.
16	n.c.		n.c.
17	n.c.		n.c.
18	n.c.		n.c.
19	n.c.		n.c.
20	n.c.		n.c.
21	n.c.		n.c.
22	n.c.		n.c.
23	n.c.		n.c.
24	n.c.		n.c.
25	n.c.		n.c.
26	n.c.		n.c.
27	n.c.		n.c.
28	n.c.		n.c.
29	n.c.		n.c.
30	n.c.		n.c.
31	n.c.		n.c.
32	n.c.		n.c.

Table 13: Pin Layout ST2

n.c. not connected

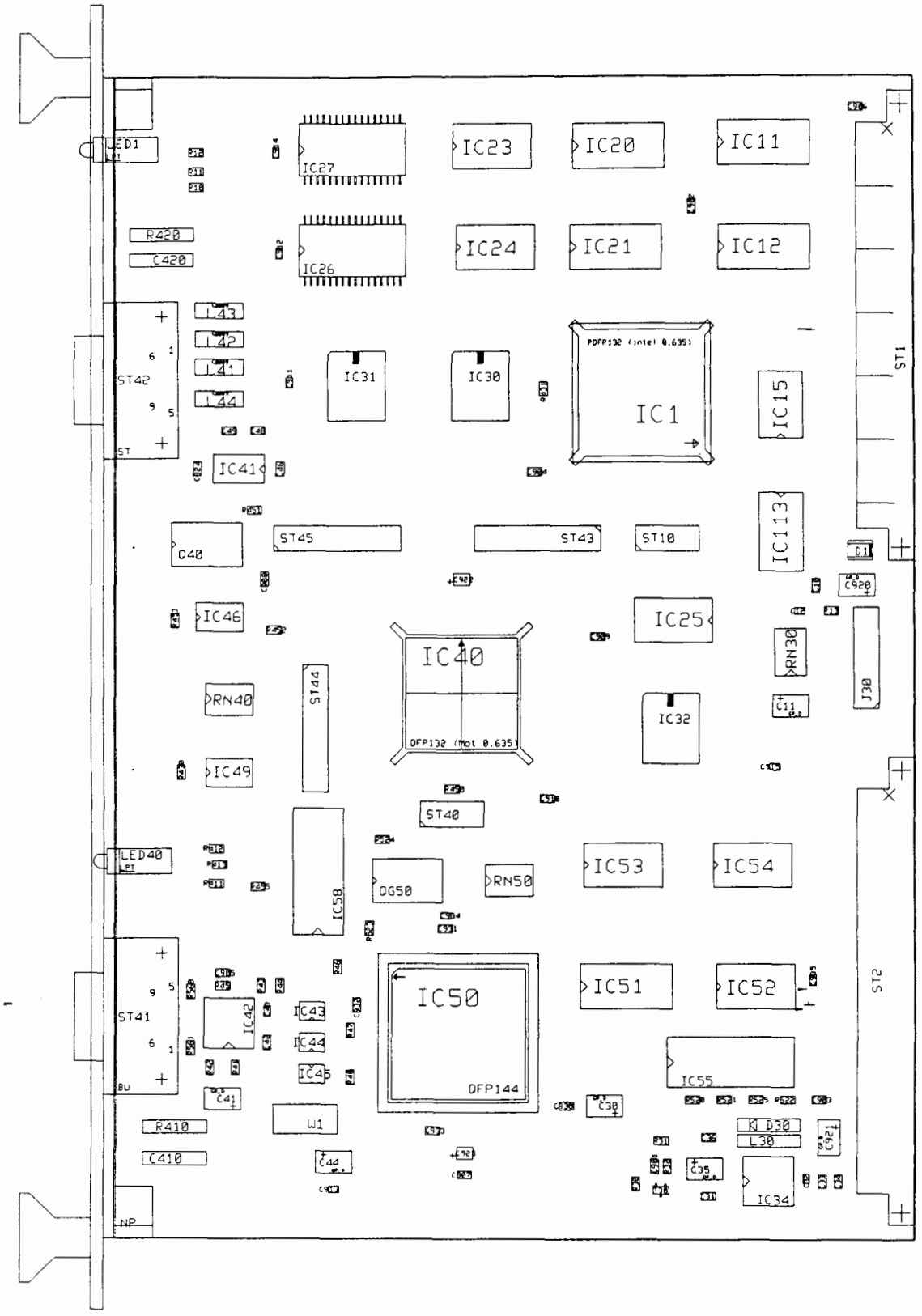
7.5. Jumper diagram



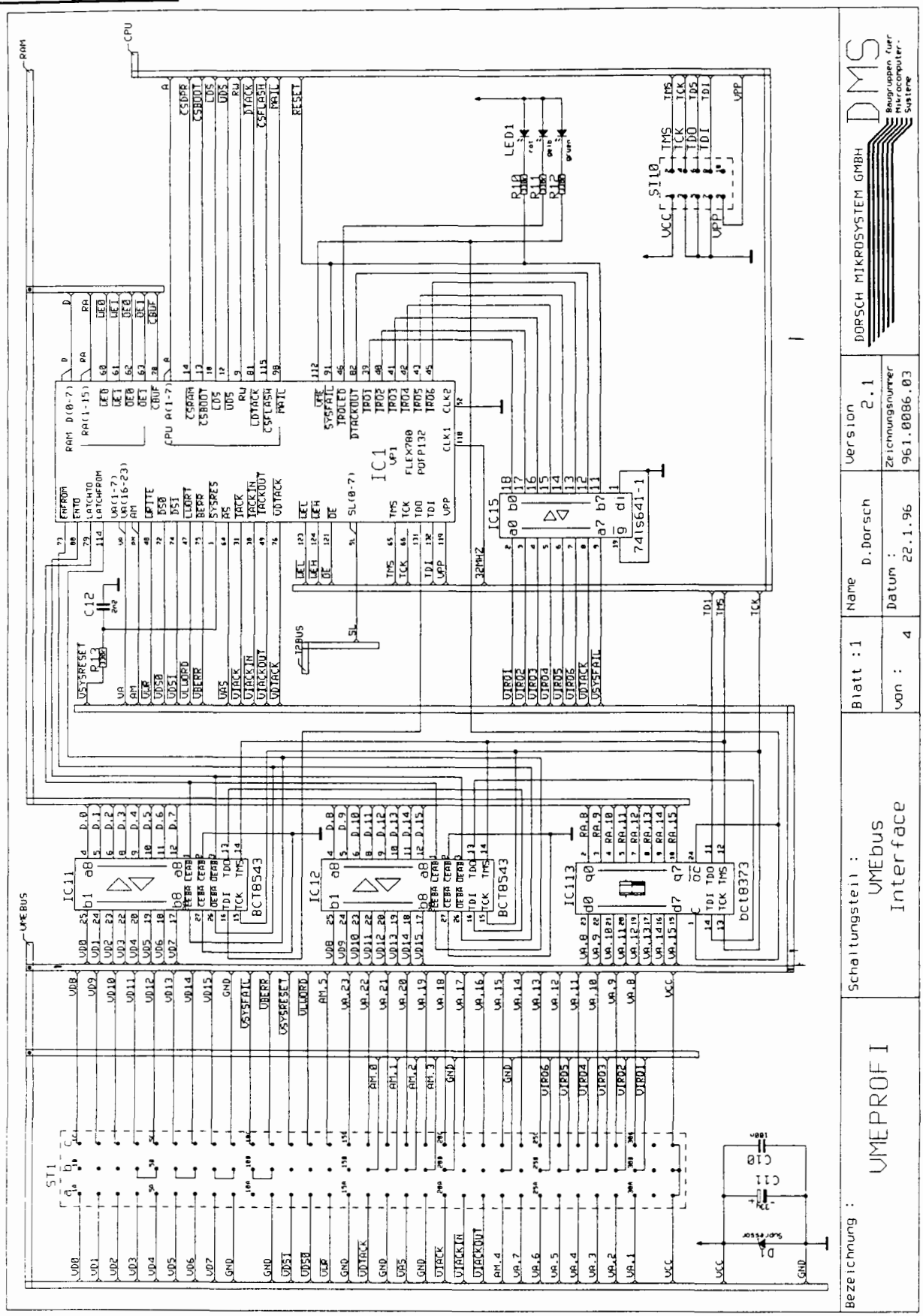
Default: $8s0000

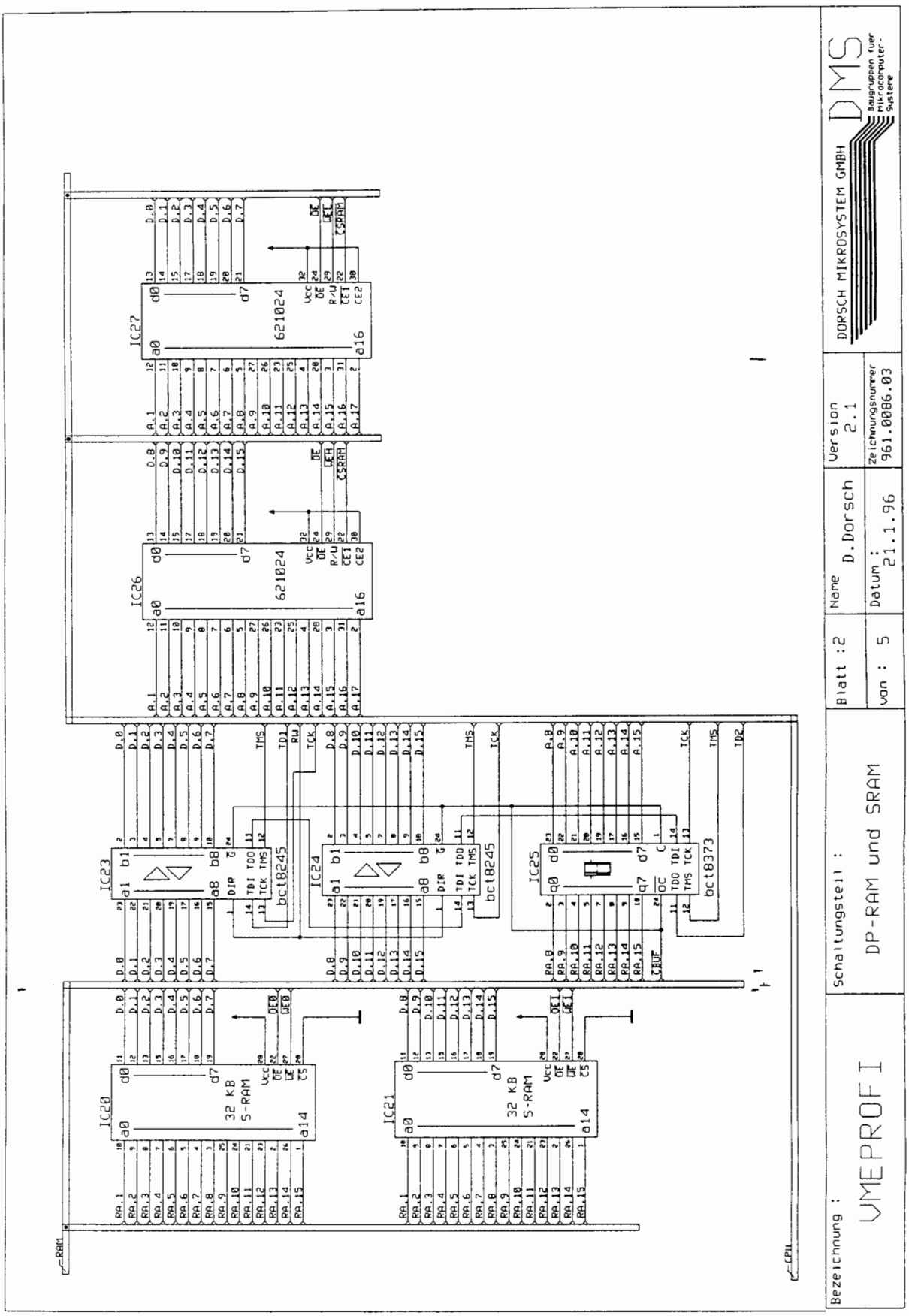
s = Slot number ($1..$F) in case of slot depending address coding
s = $F without slot depending address coding

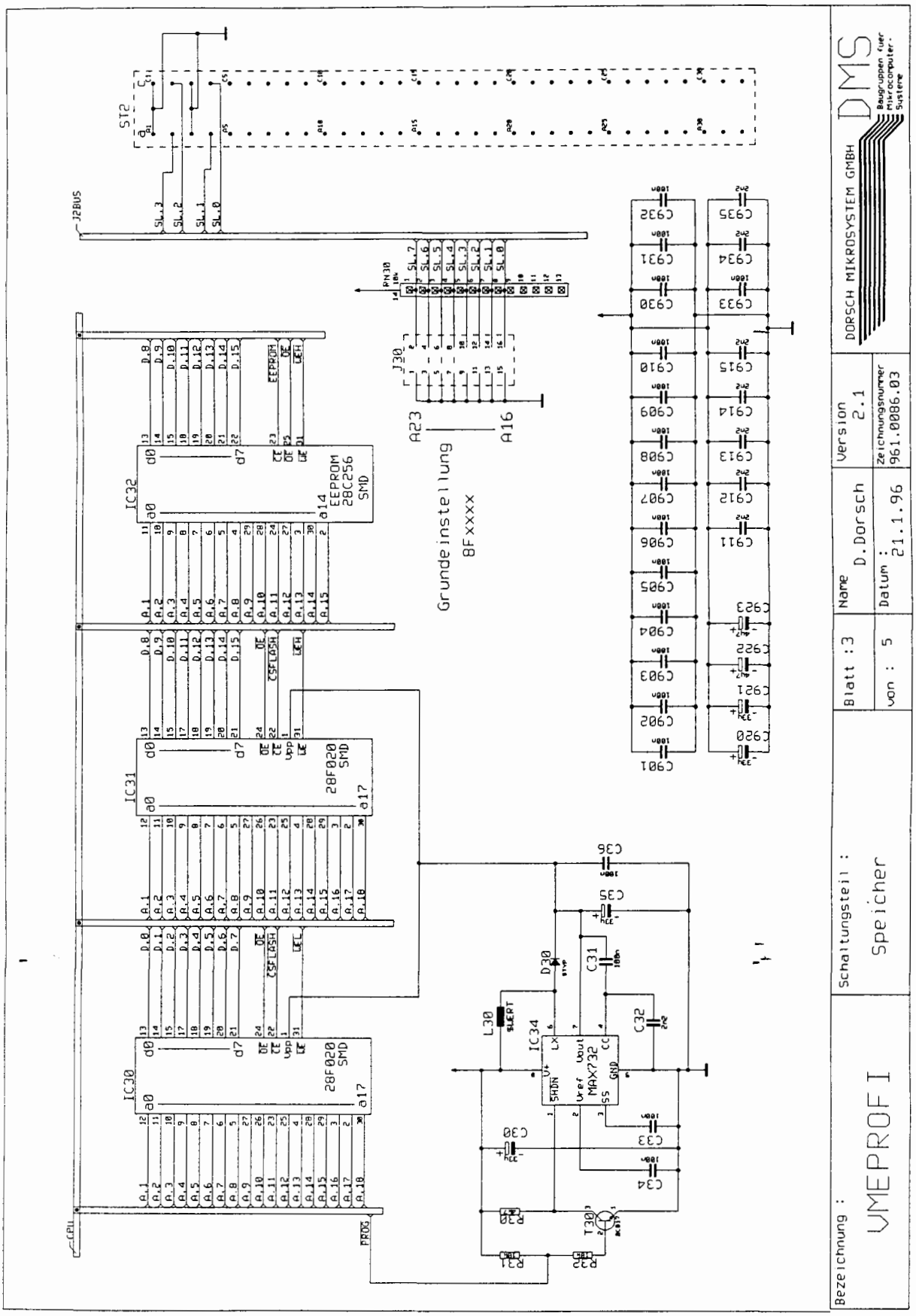
7.6. Component Mounting Diagram

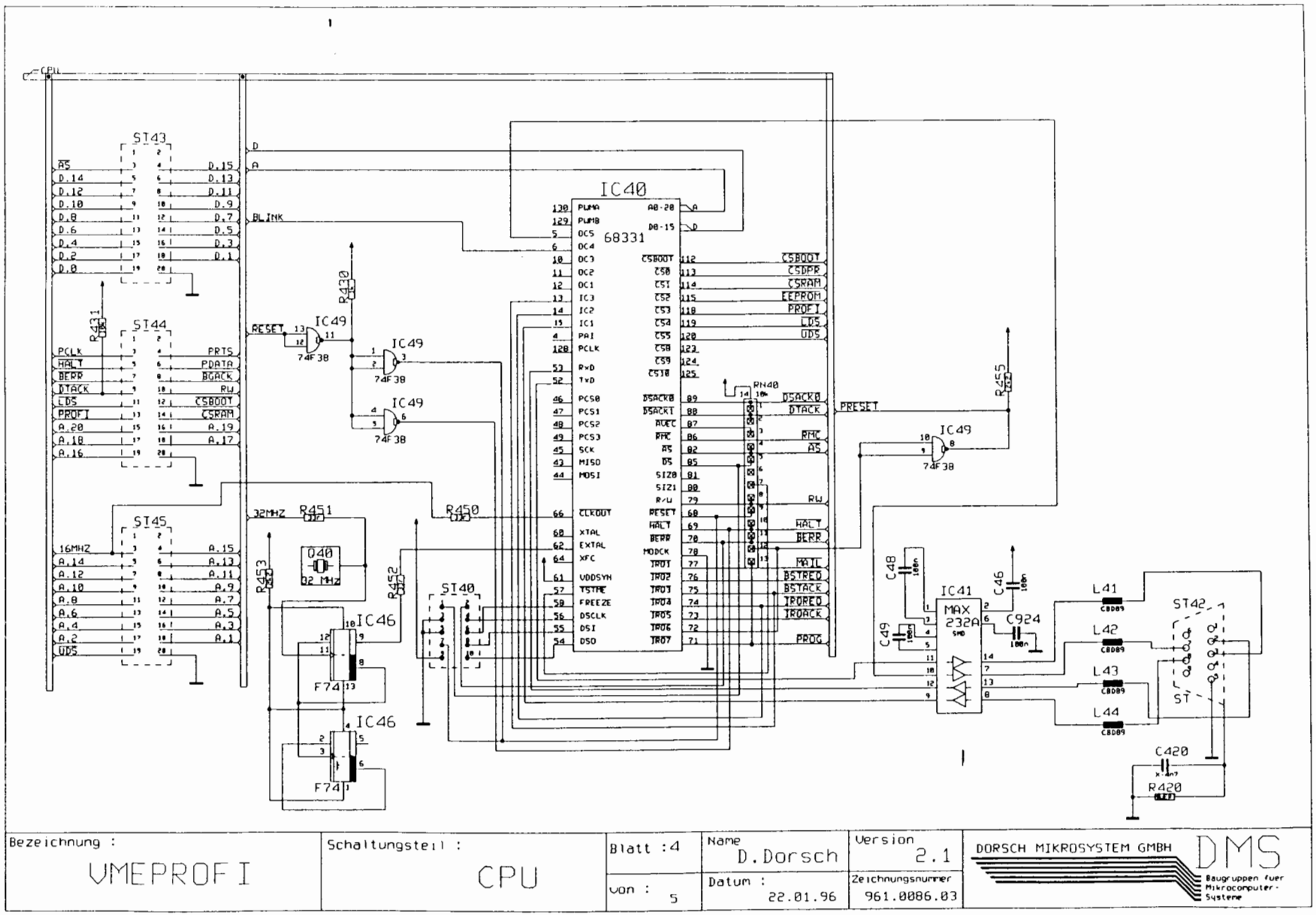


7.7. Wiring Diagram



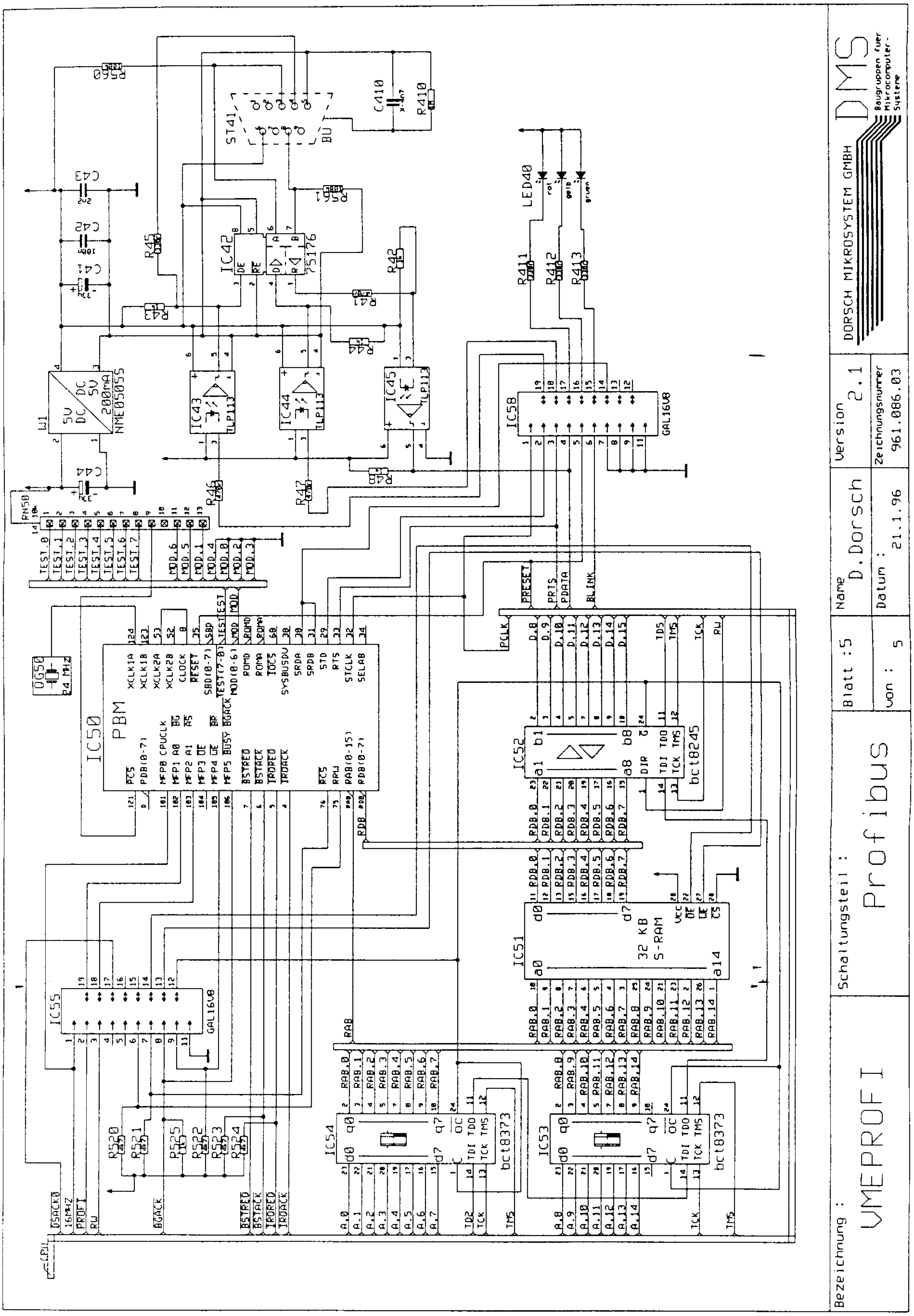

DORSCH MIKROSYSTEM GMBH
DMS
Baugruppen fuer Mikrocomputer-Systeme
Version 2.1
Zeichnungsnummer 961.0086.03
Name D.Dorsch
Datum : 21.1.96
Blatt :2
von : 5
Schaltungsteil : DP-RAM und SRAM
Bezeichnung : VMEPROFI
IC27
IC26
IC23
IC24
IC25
IC20
IC21
621024
bct8245
bct8373
32 KB S-RAM
RAM
CPU


ST2
J2BUS
J30
A23
A16
Grundeinstellung
8Fxxxx
IC32
EEPROM
28C256
SMD
IC31
IC30
28F020
SMD
IC34
MAX732
L30
D30
T30
C901
C902
C903
C904
C905
C906
C907
C908
C909
C910
C911
C912
C913
C914
C915
C916
C917
C918
C919
C920
C921
C922
C923
C930
C931
C932
C933
C934
C935
C936
C30
C31
C32
C33
C34
C35
C36
R30
R31
R32
R33
DORSCH MIKROSYSTEM GMBH
DMS
Baugruppen fuer Mikrocomputer-Systeme
Version 2.1
Zeichnungsnummer 961.0086.03
Name D.Dorsch
Datum : 21.1.96
Blatt :3
von : 5
Schaltungsteil :
Speicher
Bezeichnung :
VMEPROFI



Bezeichnung :	Schaltungsteil :	Blatt : 4	Name	Version
VMEPROFI	CPU	von : 5	D.Dorsch	2.1
			Datum : 22.01.96	Zeichnungsnummer 961.0086.03

DORSCH MIKROSYSTEM GMBH DMS Baugruppen fuer Mikrocomputer-Systeme


DORSCH MIKROSYSTEM GMBH
DMS
Baugruppen fuer Mikrocomputer-Systeme
Version 2.1
Zeichnungsnummer 961.086.03
Name D.Dorsch
Datum : 21.1.96
Blatt : 5
von : 5
Schaltungsteil : Profibus
Bezeichnung : VMEPROFI

7.8. Configuration example

This example shows, how a Slave has to be configured with the internal Control- and Configuration program and where the necessary informations can be found in the GSD-file.

As Slave a modular station with 4 byte input and 16 byte output are chosen. In this case an accordingly configured VME-PROFI(DPS).

Following, a hardcopy of the configuration screen and the shortened GSD-file are shown.

```
; Slave GSD File for VME-PROFI(DPS)
;
; ------------------------------------------------------------
; 28.08.1996            Created    RC
; ------------------------------------------------------------
;
#Profibus_DP
Vendor_Name             = "DMS Dorsch Mikrosystem GmbH"
Model_Name              = "VME-PROFI(DPS)"
Revision                = "Ed 1.0"
Ident_Number            = 0x5851                    ; ID-number
Protocol_Ident          = 0                         ; PROFIBUS DP
Station_Type            = 0                         ; DP Slave
FMS_supp                = 0                         ; FMS is not supported
Hardware_Release        = "V 2.2"
Software_Release        = "V 1.0"

; supported Baudrate
9.6_supp                = 1
19.2_supp               = 1
93.75_supp              = 1
187.5_supp              = 1
500_supp                = 1
1.5M_supp               = 1
3M_supp                 = 1

MaxTsdr_9.6             = 60
MaxTsdr_19.2            = 60
MaxTsdr_93.75           = 60
MaxTsdr_187.5           = 60
MaxTsdr_500             = 100                       ; maxT must be ≥ this value
MaxTsdr_1.5M            = 150
MaxTsdr_3M              = 250

Redundancy              = 0                         ; no Redundancy
Repeater_Ctrl_Sig       = 2                         ; TTL
24V_Pins                = 0                         ; not connected

; ***************
; Slave Parameter                                   ; The VME-PROFI(DPS) didn't require UsrPrm
: ***************

Freeze_Mode_supp        = 0                         ; Freeze Mode not supported
Sync_Mode_supp          = 0                         ; Sync Mode not supported
Auto_Baud_supp          = 0                         ; automatic Baudrate Search not supported
Set_Slave_Add_supp      = 0                         ; SetSlaveAdr not supported
User_Prm_Data_Len       = 0                         ;
Min_Slave_Intervall     = 10                        ; in 100 ys   MinS must be ≥ this value

Modular_Station         = 1                         ; it is an modular Station
Max_Modul               = 32                        ; it supports 32 Modules
Max_Input_Len           = 243                       ; maximum Input Length
Max_Output_Len          = 244                       ; maximum Output Length
Max_Data_Len            = 487                       ; maximum In-Output Length

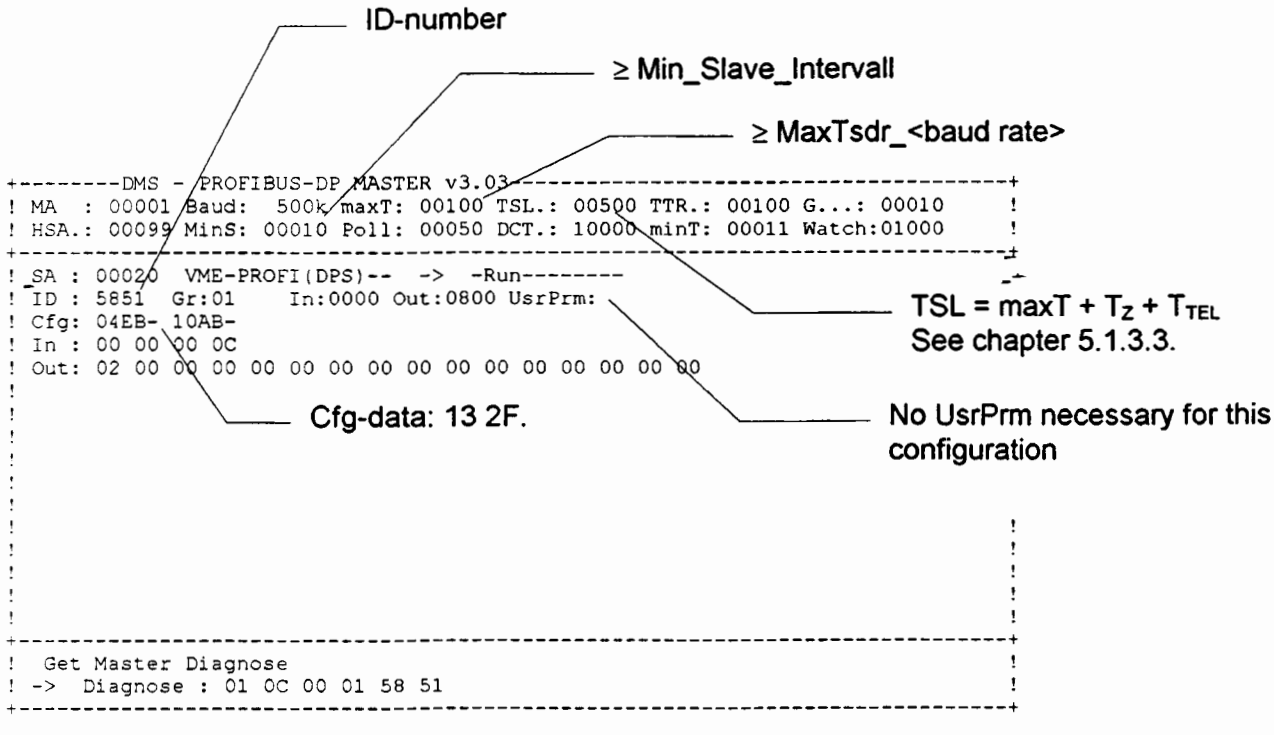
;
; Module byte-data, Inputdata from VME-PROFI(DPS) to Master;
;
Module = "01EB-,  1 Byte Digital in" 0x10
EndModule
```

```
Module = "02EB-,  2 Byte Digital in" 0x11
EndModule
Module = "03EB-,  3 Byte Digital in" 0x12
EndModule
Module = "04EB-,  4 Byte Digital in" 0x13
EndModule
Module = "05EB-,  5 Byte Digital in" 0x14
EndModule
Module = "06EB-,  6 Byte Digital in" 0x15
EndModule
        |
        |
        |
Module = "10EB-, 16 Byte Digital in" 0x1F
EndModule
;
;
; Module byte-data, Outputdata from Master to VME-PROFI(DPS);
;
Module = "01AB-,  1 Byte Digital out" 0x20
EndModule
        |
        |
        |
Module = "0DAB-, 13 Byte Digital out" 0x2C
EndModule
Module = "0EAB-, 14 Byte Digital out" 0x2D
EndModule
Module = "0FAB-, 15 Byte Digital out" 0x2E
EndModule
Module = "10AB-, 16 Byte Digital out" 0x2F
EndModule
;
;
;Module byte-data, Input/Outputdata;
;
Module = "01XB-,  1 Byte Digital in/out" 0x30
        |
```

1. Cfg-byte

2. Cfg-byte

Note:
The unshortened, actual GSD-file of the VME-PROFI(DPS) are on the disk „VMEPROFI".



Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP