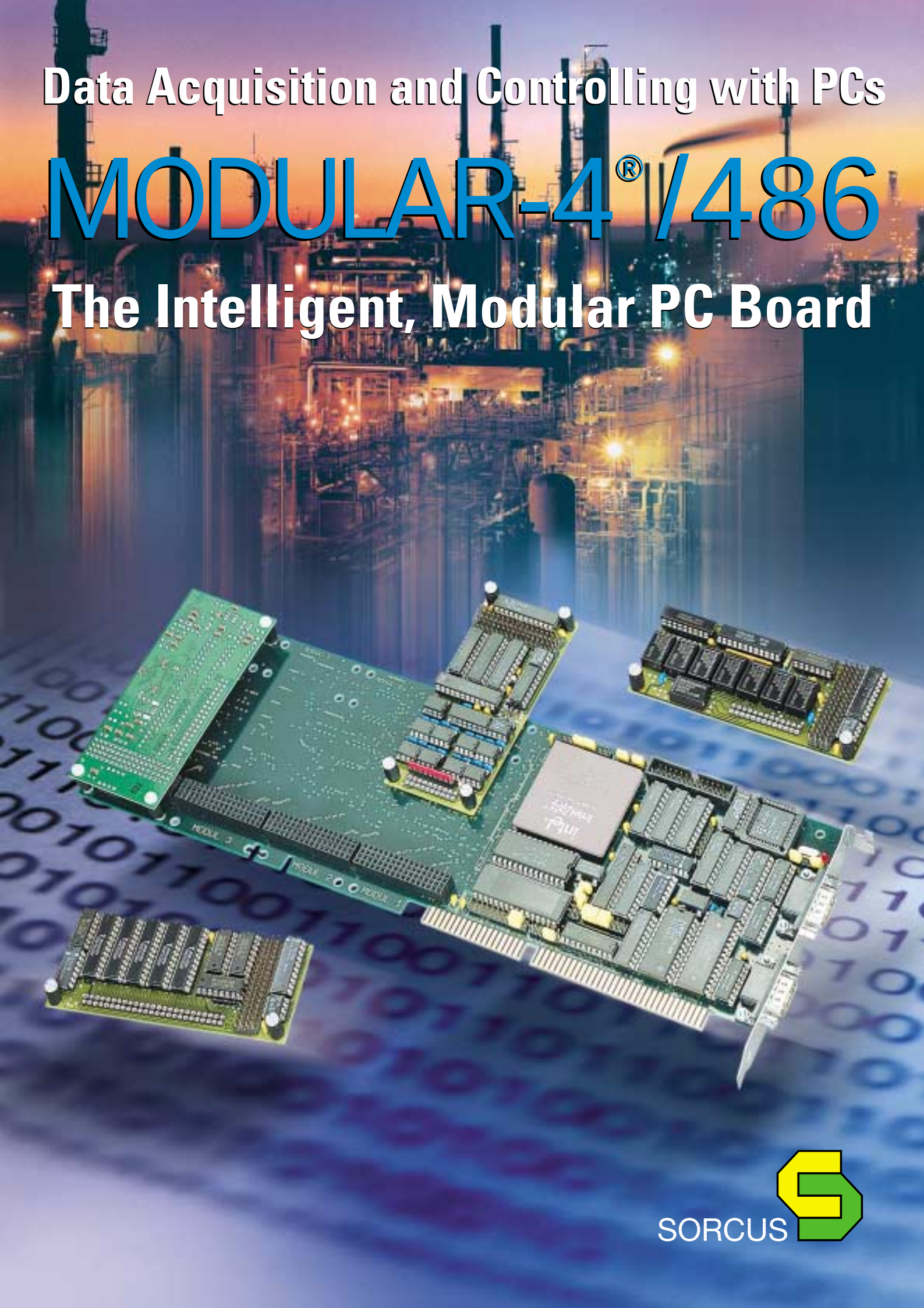
Data Acquisition and Controlling with PCs

MODULAR-4®/486

The Intelligent, Modular PC Board



SORCUS

MODULAR-4/486

The Intelligent Modular PC Board

The MODULAR-4/486 board for the IBM PC/AT bus (ISA bus) contains a complete computer on a PC extension board, enabling it to operate independently of the PC to provide genuine parallel processing (up to 8 boards in a PC).

The 486 or 586 CPU on the board contains an 8 or 16 kB cache and an arithmetic coprocessor. Depending on the version involved, the internal clock frequency of the local CPU is currently between 25 MHz and 133 MHz. A low-cost version with a 486SX CPU is also available.

The basic board is already equipped with extensive peripherals. In addition to EPROM or flash and RAM (static and dynamic, max. 34 MB), the board is also provided with a watchdog timer, a voltage monitor, and (as an optional extra) with a fan and temperature monitor for the CPU. The static RAM can be buffered with an external battery, thus enabling important data to be saved in the event of a power failure.

The board also features 6 timers, all of them with interrupt capability, and two serial RS-232 interfaces with all the requisite modem control lines. Both interfaces can be used for nearly all kinds of asynchronous and synchronous communication protocols, including SDLC, HDLC, NRZI, etc.

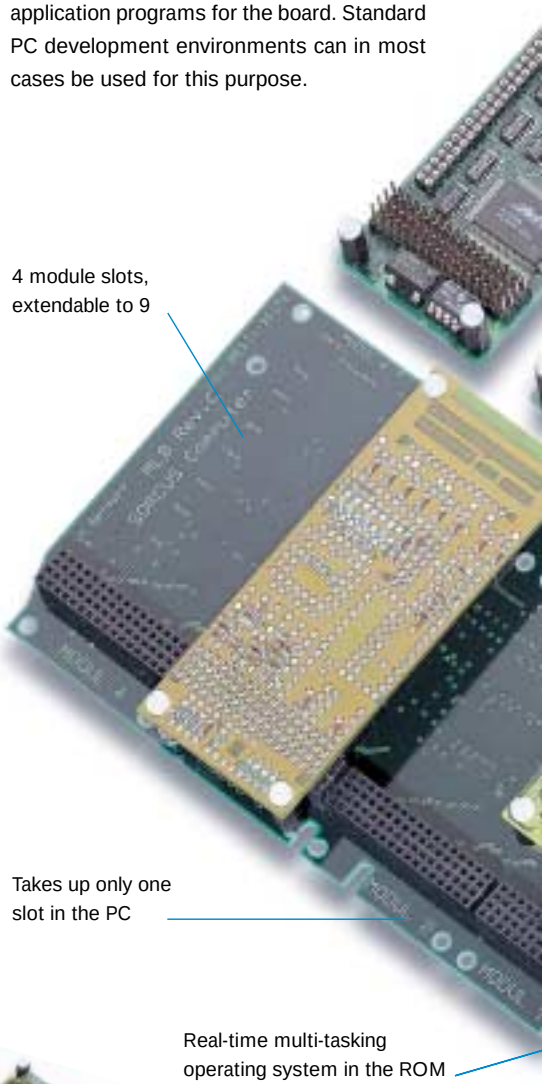
The board's power supply constitutes a special feature: the board can be supplied either through the PC system or by a separate 5-volt power pack. This enables the board to be used as a single-board computer. However, the board does not replace what is called a slot CPU, i.e. it is not a PC CPU board.

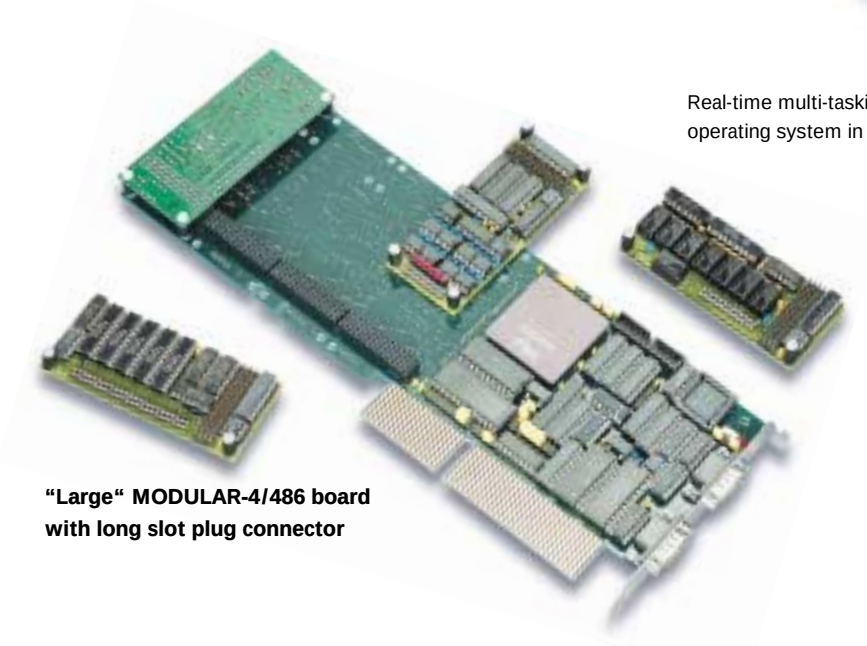
Plug-on modules can be used to adapt the board to any instrumentation-and-control or communication tasks desired. Approx. 50 different types of module are available at present: digital and analog inputs and outputs (up to 16 bits resolution, electrically isolated as well, counters, incremental encoder interfaces, serial interfaces (e.g. 8 x RS-232 per module, RS-422, RS-485, 20 mA), CAN, PROFIBUS master and slave, etc. This means that 34 or 74 serial RS-232 interfaces can be implemented, for example.

The board's EPROM or flash features a real-time multi-tasking operating system for up to 1024 tasks. Real-time programs, e.g. for data acquisition, for control systems, PID controllers, function generators, FFT, and also for serial communication are available, as are drivers for MS-DOS, Windows 3.x, Windows 95, 98 and Windows NT. And complete PLC communication protocols, e.g. Siemens 3964/R, GE Fanuc and PROFIBUS, are obtainable.

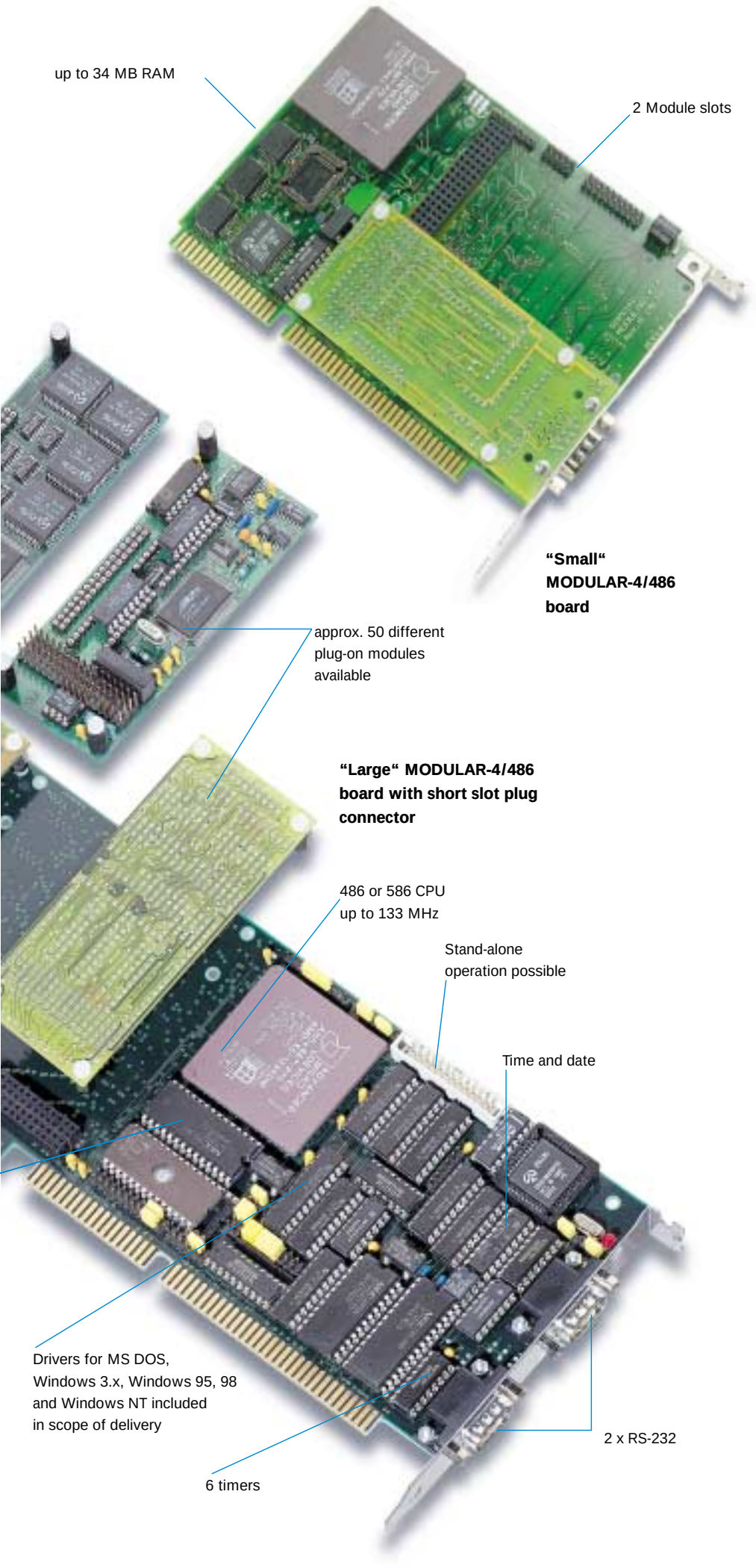
The MODULAR-4/486 board can also be used in what is called stand-alone operation. In this mode, the application programs are located in the flash. Communication with the host can take place, for example, via a serial interface (RS-232, RS-485, etc.) or via CAN bus.

Users can also write their own real-time application programs for the board. Standard PC development environments can in most cases be used for this purpose.





"Large" MODULAR-4/486 board with long slot plug connector



Special characteristics

- **Intelligent PC board with its own CPU and peripherals**
- **The board's local CPU works in parallel with the PC's CPU**
- **Available with 486 and 586 CPU up to 133 MHz**
- **"Small" MODULAR-4/486: up to 34 MB RAM (2 MB stat. + 32 MB dyn.)**
- **"Large" MODULAR-4/486: up to max. 4 MB static CMOS-RAM (bufferable)**
- **2 and 4 extension slots respectively for I/O modules, can for the "large" MODULAR-4/486 be extended to comprise a total of max. 9 slots by means of module extender**
- **Watchdog, NMI and RAM buffer logic**
- **6 timers**
- **Time and date**
- **Initialization data in the EEPROM**
- **2 x RS-232 interfaces (async., sync., HDLC, etc. on the carrier board)**
- **Real-time multi-tasking operating system on-board (in the EPROM)**
- **Drivers available for MS-DOS, Windows 3.x, Windows 95, 98 and Windows NT**
- **Stand-alone operation possible**
- **Borland development environment can be used, e.g. Turbo-Pascal, Borland C++, Turbo Debugger (source level debugger)**

Module Overview

Analog and digital-I/O Modules

All modules are delivered with MODULAR device drivers and/or libraries. These are suitable both for PC programs running under DOS, Windows 3.x, Windows 95, 98 and Windows NT and for real-time programs running on the board itself. The modules possess a configuration EEPROM where an initial status for the module can be entered. After a basic board reset, this initialization will be set.

Analog I/O modules

Modul	In	Out	Resol.	isol.
M-AD12-16	16	–	12 bits	no
M-AD16-4	4	–	16 bits	yes
M-SH12-8	8 (+2)	1	12 bits	no
M-5B-1	4	1 (+14)	12 bits	no
M-DA4-2	–	4 (+1)	12 bits	no
M-DA2-2	–	2	12 bits	yes
M-DA16-2	–	2	16 bits	yes
M-X40-1	max. 40	max. 40	var.	if nec.

M-AD12-16

- 16 analog inputs, single-ended or 8 differential inputs, selectable for each channel
- 12-bit resolution
- 1.8 µs conversion time (M-AD12-16/2)
- 16 input ranges selectable for each channel
- Customized input ranges possible

M-AD16-4

- 4 analog differential inputs
- Electrically isolated from the carrier board
- 16-bit resolution
- 5 µs conversion time
- 4 input ranges: +/- 5 V, +/- 10 V, 0 ... 5 V, 0 ... 10 V

M-SH12-8

- 8 analog inputs, can be sampled simultaneously (8 sample/hold)
- 12-bit resolution
- 3 µs conversion time
- Trigger inputs: 1 analog and 1 digital
- 1 analog input, 12-bit resolution

M-5B-1

- 4 analog inputs
- 12-bit resolution
- 1 analog output, 12-bit resolution
- 14 digital outputs (TTL)
- Suitable for controlling ext. multiplexers, e.g. 5Bx02 (up to 256 channels)

M-DA4-2

- 4 analog outputs
- 12-bit resolution
- 8 output ranges selectable for each channel:
  0 ... 2.5 V to 0 ... 10 V, +/- 2.5 V to +/- 10 V, 0 ... 20 mA, 4 ... 20 mA
- All channels can be set synchronously
- 1 digital output

M-DA2-2

- 2 analog outputs, 12-bit resolution
- Electrically isolated from the carrier board
- 8 output ranges selectable for each channel:
  0 ... 2.5 V to 0 ... 10 V, +/- 2.5 V to +/- 10 V, 0 ... 20 mA, 4 ... 20 mA
- Both channels can be set synchronously

M-DA16-2

- 2 analog outputs
- 16-bit resolution
- Electrically isolated from the carrier board
- Output voltage range +/- 10 V
- Max. output current +/- 20 mA
- Both channels can be set synchronously

Digital I/O modules

Module	In	Out	Level	isol.
M-D40-2	40 + 4	40 + 2	TTL	no
M-OPT-1/A	16	–	Optoc.	yes
M-OPT-1/B	–	16	Optoc.	yes
M-RU8-2	–	8	Relais	yes
M-C16-1	12	4	Optoc.	yes
M-C16-3	12	4	Optoc.	yes
M-AX-16	12	4	Optoc.	yes
M-AX-32	32	32	TTL	no
M-AX-32/SSI	2	0	TTL	no
M-DC15-2	4	–	Optoc.	yes
M-X40-1	max. 40	max. 40	var.	yes

M-D40-2

- 40 digital inputs/outputs
- Selectable as inputs or outputs in groups of 8 each
- Additional 4 interrupt inputs
- 2 clock or timer outputs
- Actual status of outputs can be read back

M-OPT-1/A

- 16 electrically isolated inputs
- 2 of these usable as multi-function inputs, e.g. for interrupts
- All inputs can be sampled synchronously
- Can be configured for a wide input voltage range

M-OPT-1/B

- 16 electrically isolated outputs
- Open-collector outputs
- Version /Bx available with 80 mA and 100 V outputs
- Watchdog timer on the module

M-RU8-2

- 8 relay outputs
- 1 switch-over contact per output
- 100 V / 1 A per output
- Watchdog timer on the module

M-C16-3

- 3 independent counter channels, max. count rate 10 MHz, 16 bits, cascadable
- 13 operating modes selectable per channel: including counter, incremental encoder interface, pulse-width, frequency, period and speed measurement
- 12 opto-isolated inputs and 4 outputs

M-C16-1

- 1 counter channel, max. count rate 10 MHz, 16 bits
- 13 operating modes selectable: including counter, incremental encoder interface, pulse-width, frequency, period and speed measurement
- 12 opto-isolated inputs and 4 outputs

M-AX-16

- 12 opto-isolated inputs and 4 outputs
- Function can be programmed at will by means of gate array (Xilinx 3090)
- Function can be altered at any time (by download or EPROM)
- 2 control LEDs
- Designs available: I/O, 20-bit counter, interrupt controller

M-AX-32

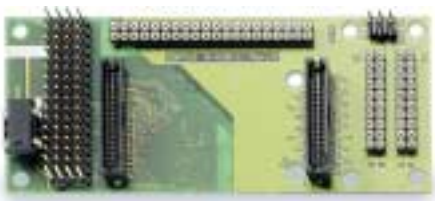
- 32 I/O lines, selectable in the software as inputs or outputs for 8 each
- Function can be programmed at will by means of gate array (Xilinx 3090); can be altered as well (by download or EPROM)
- 2 control LEDs
- Designs available: I/O, SSI

M-AX-32/SSI

- 2 channels, synchronous serial interfaces
- Transmission clock settable
- Gray and binary coding
- Settable number of data bits (max. 32)

M-DC15-2

- 4 opto-isolated inputs with interrupt capability
- DC/DC converter for +/- 15 V, 4 W
- 8 LEDs



M-X40-1

- Adapter module for all X-Bus modules
- 40 I/O lines

Module Overview

Modules for Communication

Modules for communication

Module	Channels	Protoc.	Level	isol.
M-COM-2	2	var.	by C-Link	poss.
M-COM-2/P	2	var.	Plastic	yes
M-COM-2/G	2	var.	Silica	yes
M-COM-8	8	var.	RS-232	no
M-IEC-1	1	IEC-488	GPIB	no
M-DPM-12	1	PROFIBUS	RS-485	yes
M-DPS-12	2	PROFIBUS	RS-485	yes
M-CAN-1	1	CAN-Bus	RS-485 (CAN)	yes

M-COM-2

- 2 multi-purpose serial interfaces, e.g. for async, sync, SDLC, HDLC, etc.
- 3964/R protocol available
- Baud rate generator, DPLL and FIFO per channel
- Physical interface selectable per channel: RS-232, RS-422, RS-485, RS-232 isol., RS-485 isol., RS-422 isol., 20 mA

M-COM-2/P und /G

- 2 multi-purpose serial interfaces, e.g. for async, sync, SDLC, HDLC, etc.
- 3964/R protocol available
- Baud rate generator, DPLL and FIFO per channel
- Physical interface for optical fibers: M-COM-2/P for plastic fibers, M-COM-2/G for silica fibers

M-COM-8

- 8 multi-purpose RS-232 serial interfaces, e.g. for async, sync, SDLC, HDLC, etc.
- Baud rate generator, DPLL and FIFO per channel
- Any Baud rates up to 500 kBaud
- 2 modem control lines per channel: RTS or CLKout, and CTS or CLKin

M-IEC-1

- Complete IEC bus interface
- All functions in conformity with IEC standard
- Can be used as system controller or device
- Max. data transmission rate: 600 kBytes/s

M-DPM-12

- PROFIBUS master with its own CPU
- Compatible with DIN 19 245, Parts 1 to 3
- Baud rates from 9.6 k to 12 MBits/s
- RS-485, electrically isolated
- 16 kByte dual-ported RAM interface
- Software included in scope of delivery

M-DPS-12

- 2 independent PROFIBUS slave channels
- Compatible with DIN 19 245, Parts 1 to 3
- Automatic baud rate detection from 9.6 k to 12 MBits/s
- RS-485, electrically isolated per channel
- Software included in scope of delivery

M-CAN-1

- 1 MBits/s CAN module with full CAN functionality
- Supports CAN specification 2.0 A and 2.0 B (11-bit and 29-bit identifiers)
- Electrically isolated physical interface to the CAN bus
- BUS terminating resistor can be switched into circuit using the software

C-Link Adapters
for M-COM-2

The M-COM-2 module contains two serial synchronous/asynchronous interfaces. The physical interfaces are for each channel configured using a C-Link adapter. C-Link adapters are plug-in micro modules in the size of 24 pole ICs, with one standard connection pattern for serial interfaces. C-Link adapters can be installed or replaced by the customer, meaning that all commonly used levels can be configured. There is also an option for electrical isolation. C-Links are also used on the M-DPM-12 and M-DPS-12 modules.

CL232S

- RS-232 up to 120 kBaud
- Modem control lines: TMT, RCV, RTS, CTS, DTR, DSR, RI, DCD
- Additional functions:
  Mode 0: RI as clock input
  Mode 3: CTS as clock input
  Mode 5: RTS as clock output

CL232A/i

- RS-232 up to 120 kBaud
- Modem control lines: TMT, RCV, RTS, CTS, DTR, DSR, RI, DCD
- Additional functions:
  Mode 0: additional RS-232 line EXT as clock input 1
  Mode 0: RI as clock input 2

CL232A/o

- RS-232 up to 120 kBaud
- Modem control lines: TMT, RCV, RTS, CTS, DTR, DSR, RI, DCD
- Additional functions:
  Mode 5: additional RS-232 line EXT as clock output

CL232i

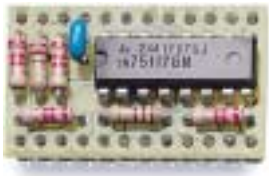
- RS-232 isol. up to 120 kBaud
- Isol. modem control lines: TMT, RCV, RTS, CTS
- Additional functions:
  Mode 3: CTS as clock input
  Mode 5: RTS as clock output

CL422S

- RS-422 up to 10 MBaud
- Modem control lines: TMT, RCV, RTS, CTS
- Additional functions:
  Mode 3: CTS as clock input
  Mode 5: RTS as clock output

CL422i

- RS-422 isol. up to 10 MBaud
- Modem control lines: TMT, RCV, RTS, CTS
- Additional functions:
  Mode 3: CTS as clock input
  Mode 5: RTS as clock output



CL485S

- RS-485 up to 10 MBaud
- Modem control lines: TMT, RCV, RTS, CTS
- Additional functions:
  Modes 0 and 2: RTS driver disabled
  Mode 2: CTS as clock input
  Mode 3: CTS as clock input
  Mode 5: RTS as clock output

CL485i/U

- RS-485 isol., suitable for up to 20 MBaud
- Switchover from transmit to receive by the software or automatically (e.g. for SDLC/HDLC)

CL485i/P

- RS-485 isol. up to 20 MBaud
- Suitable for PROFIBUS up to 12 MBaud, additional TTL output indicates transmit/receive, e.g. for transceiver

CL200A

- 20 mA isol. up to 38.4 kBaud
- Current loop, two constant-current sources on the C-link. Option for either passive or active configuration (if passive, then electrically isolated)

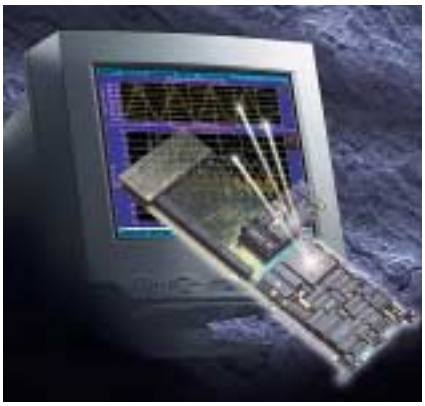
Programming

PC Applications for MODULAR-4/486

For programming with MODULAR-4/486 boards, a distinction must be made between PC applications and programs running on the MODULAR-4 board, also referred to as real-time programs.

PC programming

The high-level language libraries for PC applications offer their users a convenient interface for communicating with MODULAR-4/486 boards. They are available for various programming languages (C, Pascal, BASIC) and operating systems (MS-DOS, Windows 3.x, 95, 98 and NT). One library can serve up to eight boards.



Since the board is an independent system with an integrated processor, communication with the PC is a sophisticated process. This process is completely dealt with by the library, so you need not bother with the details. The libraries provide the following tasks:

- Configuration of the board
- Downloading of real-time programs onto the board
- Data exchange between board and PC
- Error handling
- Interrupt handling

Scope of delivery

The boards' scope of delivery comprises all high-level libraries (including drivers). The latest library versions are also available at any time and free of charge from the Internet (www.sorcus.com).

The operating systems and compilers supported (and their versions) are listed in the table on the right. If the compiler you are using is not mentioned there, please get in touch with SORCUS.

Portability

The libraries' scope of functions is the same for the different PC operating systems, so that once a PC application program has been developed it can easily be used with another operating system.

Linking the libraries

MS-DOS: for the C programming language, LIB files are supplied while it is units for PASCAL.

Windows 3.x: a DLL is supplied for programming. Import libraries for the individual programming languages create an interface to this DLL. In C, a LIB file is linked to the project; in Pascal a Windows unit is linked to the program.

Windows NT: the Windows NT library consists of two parts: a device driver, which handles communication with the board, and a higher-order DLL, which puts the library functions at your disposal. An import library, which establishes the interface to the DLL, is linked to the project. To run MS-DOS programs under Windows NT, a virtual device driver is available.

Windows 95/98: see under Windows NT.

Operating systems and programming languages supported:

MS-DOS:

- Borland C (from Version 3.1)
- Microsoft C (from Version 8.0)
- Watcom C (from Version 10.0)
- Borland PASCAL (from Version 6.0); protected mode as well

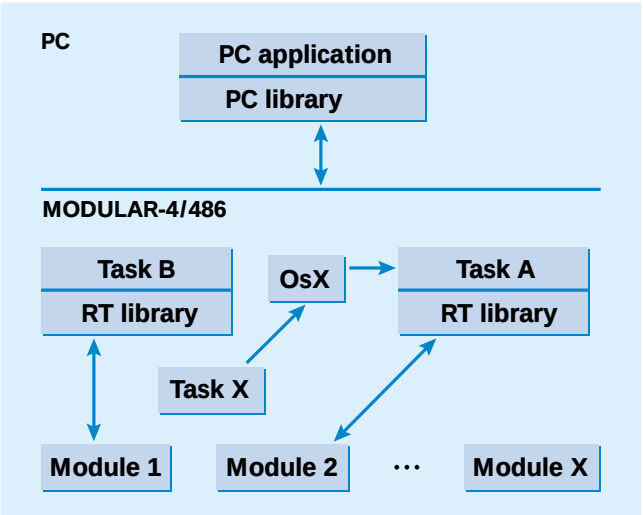
Windows 3.x

- Borland C (from Version 3.1)
- Borland PASCAL (Version 7.0)
- Borland Delphi (from Version 1.0)
- Microsoft Visual Basic (from Vers. 3.0)
- Microsoft Visual C (from Version 1.0)
- Watcom C (from Version 10.0)
- DASYLab driver

Windows 95/98 and Windows NT

- Microsoft Visual C (from Version 4.0)
- Borland C (from Version 5.0)
- Borland Delphi (from Version 2.0)
- Microsoft Visual Basic (from Vers. 4.0)
- DASYLab driver

Under preparation:
LabView driver



The PC libraries serve for creating application programs which communicate with the MODULAR-4/486 boards. Real-time programs run as tasks on the board. The real-time library (RT library) serves for creating real-time programs.

Real-Time Programming

Real-time programming

All SORCUS boards provide their own microprocessor, on which a real-time-capability operating system runs: OsX. This provides genuinely parallel processing to the PC, without which it is difficult to acquire and process measured data in real time (particularly when using modern PC operating systems like Windows NT or similar). Data acquisition and communication tasks can be run on the board completely independently of the PC, with the PC also being able to handle other tasks like visualization and storing the data.

The OsX multi-tasking operating system enables several processes (tasks) to be run simultaneously on a single board. Real-time programs, i.e. the programs which are being run on the board as tasks, can be very simply programmed by the users themselves. Software developers can use the standard compilers from Borland (PASCAL or C++); a special development environment is not required.

You can develop your own real-time program in three steps:

1. Enter and compile the real-time program under Borland PASCAL or Borland C++.

2. Transfer the program onto the MODULAR-4 board.

3. Test and debug the real-time program with the Borland source code debugger.

A real-time program's structure resembles that of a DOS program, with the difference that the program code is divided up into what are called "task procedures". Task procedures can be called subsequently by other tasks on the board, or by the PC as well, e.g. for starting or aborting transmit or receive functions. Besides the task procedures, the program also incorporates parameter and data areas. The parameter area normally contains user-definable configuration and parameterization data like the baud rate, the number of channels, etc. The data area can, for example, be used to accommodate the data acquired. Here, too, as with the task procedures, other tasks on the board, and the PC itself, can very simply access the parameters and data involved. And complete libraries are available for addressing the various I/O function units and the operat-ing system's routines.

```
/************** DEMOPROGRAM (NON-INTERRUPT TASK FOR MODULAR-4/486 ***************/
/*
#include <dos.h>
#include <stdio.h>
#include "mlsrtbib.h"                          /* Link the real-time library */

struct parameter_type                          /* Declare the parameter area */
(
    tdt_type          tdt;          /* Space for the task descriptor table */
    unsigned char     status;              /* Program status (READY etc.) */
    unsigned int      blink_rate;                        /* Flashing rate */
    unsigned char     led_status;                          /* LED status */
    unsigned char     led;         /* Which LED 1 = internal, 2 = external */
) parameter;

/* The start procedure starts the task and sets the status parameter indicating
   the program's status.
*/
void start(void)                                    /* —— Start procedure —— */
(
    mlsrt_entry();             /* Save the registers, set the data segment */
    parameter.status = RUNNING;             /* Status to "Program running" */
    mlsrt_wakeup_task (parameter.tdt.task);
    mlsrt_exit();                            /* Return to operating system */
)

/* The stop procedure stops the task and sets the status parameter indicating
   the program's status.
*/
void stop(void)                                     /* —— Stop procedure —— */
(
    mlsrt_entry();             /* Save the registers, set the data segment */
                                                    /* Deactivate program */
    mlsrt_sleep_task (parameter.tdt.task);
    parameter,status = STOPPED;  /* Status parameter to "Program aborted" */
    mlsrt_exit();                            /* Return to operating system */
)

/* The main procedure is cyclically called automatically by the operating
   system as soon as the task has been activated by calling the start procedure.
   It causes the LED selected to flash at a defined rate.
*/
void main_task()                          /* —— MAIN PROCEDURE OF TASK —— */
(
    mlsrt_entry();
    if (++pause >= parameter.blink_rate)
    (                                          /* if the LED is switched on, ... */
        if (parameter.led_status = ON)
        (                                       /* ... switch off LED and ... */
            mlsrt_local_led_off();
            parameter.led_status = OFF;                /* ... note status */
    )
        else                                /* if the LED is switched off, ... */
        (
           mlsert_local_led_on();           /*... switch on status and ... */
           parameter.led_status = ON;                  /* ... note status */
    )
        pause = 0;                           /* Reset run counter to zero */
 )
 mlsrt_exit();
)
```

Once complied, the real-time program can be transferred to the MODULAR-4 board with the PC utility program provided or out of a user program with the aid of the PC libraries supplied. Borland's Turbo-Debugger can be used to test the real-time program. To do this, the board is simply connected up to one of the PC's serial interfaces using a serial null-modem cable. The real-time program can then be debugged on source code level like a PC program. And of course you can utilize all the features of the Turbo-Debugger, like breakpoints, watch variables, etc.

Programming

of I/O devices – Module libraries and MODULAR device drivers

To program the I/O devices on the SPB modules and the MODULAR-4/486 carrier board, you can use either the module libraries or the more recent MODULAR device drivers (MDDs). These enable you to set timers, read analog inputs, etc.

In contrast to the module libraries, the MODULAR device drivers are not a constituent part of the application programs during runtime, but are very fast real-time programs on the MODULAR-4/486 board, handling access to the devices. One MDD is loaded for each module and for the carrier board.

MODULAR device drivers (MDDs)

The MODULAR device drivers are based on a channel-oriented approach. You "open" a channel to one or more devices, e.g. to an analog input. When it is opened, channel-specific parameters describing a channel's characteristics are transferred. This channel parameter structure (CPS) will, for example, in the case of analog inputs, contain the measuring range, the access type, etc. involved. Devices can be used simultaneously by more than one application, or assigned exclusively to a single channel.

When a channel is opened, the driver checks the channel parameter structure for plausibility and for whether the devices are available. It returns what is called a handle if the channel is available. This handle is required for subsequent access to the device (e.g. reading an analog value). The function units are accessed through a standardized interface, thus enabling hardware to be replaced, for example, without entailing modifications to the software.

Example: reading in analog inputs of the M-AD12-16 module

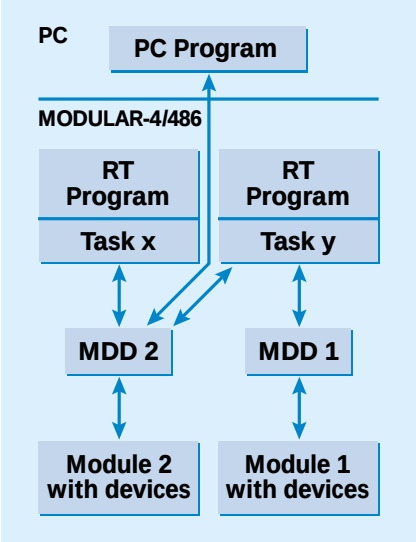
```
/* M-AD12-16 in slot 1 */
#define SLOT_MAD1216 1

/* Define channel parameter structure for M-AD12-16 */
CPS_MAD1216 rcMA1216;

/* Define handle for channel */
HMDD8 hHandle;

/* Set channel parameters:
       -   Input 0 to 15, exclusively for this channel
       -   Single-ended, range ±10 V
       -   Read out inputs directly */

rcMAD1216.wDevice      = DEVICE_AIN_SE;   // Single-ended
rcMAD1216.wIndexFirst  = 0;               // from 0
rcMAD1216.wIndexLast   = 15;              // to 15
rcMAD1216.wFlags       = _CP_EXCLUSIVE;   // Open exclusively
rcMAD1216.wReadMode    = IO_MODE_DIRECT;  // Type of access: direct
rcMAD1216.wRange       = RANGE_BIP_10V;   // Range: ±10 V


/* Open channel */
hHandle = mdd8_open_channel (SLOT_MAD1216,
                             sizeof(rcMAD1216),
                             &rcMAD1216);

/* Read inputs (16 * 2 Byte) */
mdd8_read_channel_block(hHandle, 32, &values)

/* Close channel */
mdd8_close_channel (hHandle);
```

When a channel is no longer required, it can be closed, releasing the memory it was previously occupying. Another advantage of the MODULAR device drivers is their multi-tasking suitability. In contrast to the module libraries, where you as the user have to ensure reliable running, the devices are accessed only by the MODULAR device driver. This precludes the possibility that several applications will impermissibly use a single device simultaneously.

Using MDDs

The high-level language libraries provide the functions for using the MDDs, both for real-time programming and for PC programs as well. The procedural sequence is as follows:

1. Opening a channel (with parameters) obtains a handle (once, at the beginning of the program)
2. The handle is used to access the channel resp. the device
3. Close the channel (at the end of the program)



Example: A PC program and the real-time programs "Task x" and "Task y" share devices of Module 2 using the associated "Task 2" MDD. Devices of Module 1 are used only by "Task y" through the "Task 1" MDD.

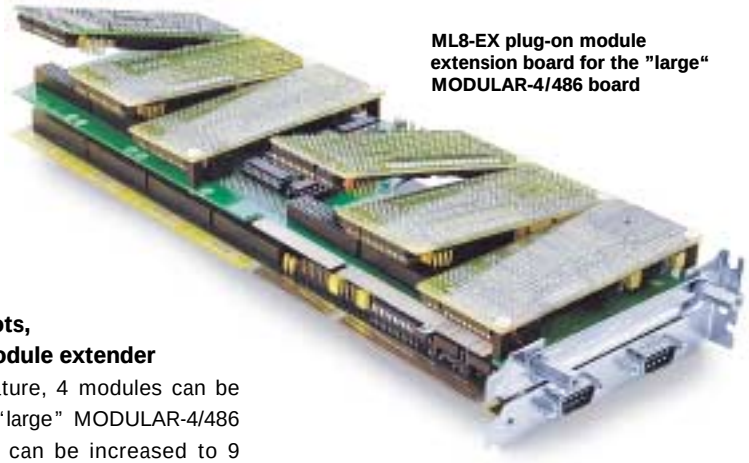
MODULAR device drivers offer a whole series of additional useful characteristics. For example, each channel can be given an unambiguous designation. Resolution, value range and physical unit for a channel can also be interrogated at any time. All an MDD's characteristics and features can be determined by each application.

Technical Data

	"Large" MODULAR-4/486				"Small" MODULAR-4/486			
Order No.								
with short ISA slot plug connector	EM-1347	EM-1233	EM-1885	EM-1886	EM-2175	EM-2179	EM-2222	EM-2592
with long ISA slot plug connector	EM-2049	EM-2042	EM-2135	EM-2136	–	–	–	–
CPU	486SX[1]	486DX2	586DX4	586DX4	486DX2	586DX4	586DX4	586DX4
Clock int. (MHz)	min. 25[1]	66	133	133	66	133	133	133
Cache (kB)	8	8	16	16	8	16	16	16
Coprocessor	no[1]	yes	yes	yes	yes	yes	yes	yes
RAM (byte)	256K	1M	1M	4M	512K	2M	10M	34M
of which stat.[2]	256K	1M	1M	4M	512K	2M	2M	2M
of which dyn.	–	–	–	–	–	–	8M	32M
EPROM (byte)[3]	64K	64K	64K	64K	64K	64K	64K	64K
Flash (byte)[3]	512K	512K	512K	512K	512K	512K	512K	512K
Ser. EEPROM (bit)	1024	1024	1024	1024	1024	1024	1024	1024
Slots for SP-Bus modules	4	4	4	4	2	2	2	2
with module extender	9	9	9	9	–	–	–	–
Timers[4]	6	6	6	6	6	6	6	6
of which in RTC	1	1	1	1	1	1	1	1
of which in SCC	2	2	2	2	2	2	2	2
RS-232 interfaces on the basic board	2	2	2	2	2	2	2	2
Real-time clock	yes	yes	yes	yes	yes	yes	yes	yes
Interrupts	15	15	15	15	15	15	15	15
of which for modules	6	6	6	6	6	6	6	6
Watchdog[5]	1	1	1	1	1	1	1	1
Ext. watchdog output	yes	yes	yes	yes	yes	yes	yes	yes
Ext. reset input[6]	yes	yes	yes	yes	yes	yes	yes	yes
LED on-board	yes	yes	yes	yes	yes	yes	yes	yes
Output for ext. LED	yes	yes	yes	yes	yes	yes	yes	yes
+5V voltage monitor	yes	yes	yes	yes	yes	yes	yes	yes
Fan control	no	no	no	no	yes	yes	yes	yes
Fan monitor	no	no	no	no	yes	yes	yes	yes
CPU temperatur monitor	no	no	no	no	yes	yes	yes	yes
Power consumption[7]								
+5V (A)[8]	1.2	1.4	1.4	1.4	1.2	1.2	1.4	1.4
+12V (mA)	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4
-12V (mA)	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4
-5V (mA)	–	–	–	–	–	–	–	–
Dimensions								
with short slot plug connector	337.6 x 106.7 mm				158 x 106.7 mm			
with long slot plug connector	337.6 x 121.9 mm				–			
Compatibility								
Temperature	0 to 55 degrees							
(optional)	0 to 70 degrees							
Humidity	5 to 95% (not condensing)							

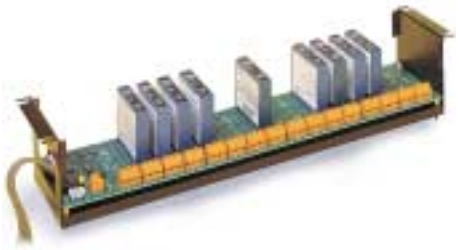
1) This board has been equipped at least with a 486SX CPU with 25 MHz. Depending on market availability, a faster CPU may be delivered.
2) The stat. RAM can be buffered by an external battery.
3) Either an EPROM-IC or a flash IC may be used. As standard, the boards are supplied with an EPROM with 64 kB, which contains the OsX real-time operating system. The boards can be changed over in the factory to 512 kB flash:
   Order No. for "large" MODULAR-4 = EM-2394
   Order No. for "small" MODULAR-4 = EM-2660
4) All timers possess interrupt capability.
5) The watchdog timer can be enabled or disabled. If it is not retriggered in time, it will force an NMI.
6) Watchdog output and reset input can be connected so as to trigger a restart of the board in the event of a watchdog timeout.
7) Measured with fan and LED switched off and inactive serial interfaces.
8) The fan requires 90 mA at 5 Volt (start-up current 160 mA). The 5-V connected load changes in dependence on the software by up to +/-30 %.

Accessories



ML8-EX plug-on module extension board for the "large" MODULAR-4/486 board

Connection panels for MODULAR-4/486



5Bx02 with installation frame AC 1363

5BA32 with installation frame AC 1363

5Bx64 with installation frame AC 1363

5B2x34 with installation frame AC 1363

More module slots, thanks to the module extender

As a standard feature, 4 modules can be plugged onto the "large" MODULAR-4/486 basic board. This can be increased to 9 modules by using the module extender (see illustration at the top right). 3 module slots are then available on the carrier board and 6 on the module extender, the two of them together taking up 2 ISA slots in the PC.

External multiplexers and signal conditioning

Many modules provide direct inputs and outputs - some of them electrically isolated. If the number of channels is not sufficient or if the signals have to be specially conditioned, external multiplexers with signal conditioning can be used.

The 5Bx02 panel offers 16 channels for analog signal conditioning, e.g. for Pt 100, CU10, Ni120 and thermocouples (Types J, K, T, E, R, S and B). There are also transducers for 0...20 mA, 4...20 mA, for frequency inputs and for outputs. Each transducer (and thus each channel) has its own electrical isolation (up to 1,500 V): it does not need any calibration and offers an accuracy of 0.05 %.

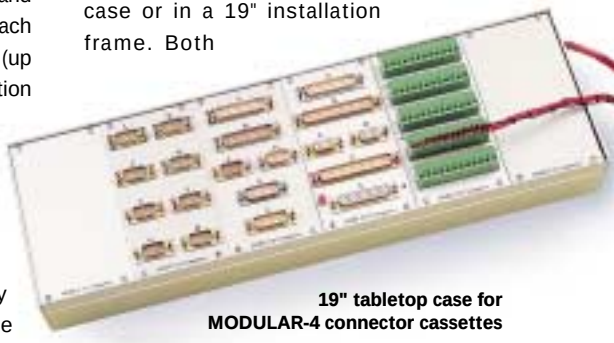
The 5BA32 panel provides two groups of 16 analog inputs each, with one transducer assigned to each group, which specifies the group's characteristics. The channels of one group are not electrically isolated from each other, but from the channels of the other group and from the PC. Both panels (5Bx02 and 5BA32) are connected to an M-5B-1 module by means of a ribbon cable; the module is plugged onto the carrier board. Up to 8 panels are possible per module, i.e. 128 and 256 channels respectively, and a maximum of 1,152 and 2,304 channels respectively per board.

In order to increase the number of digital inputs and outputs, you can use the 5Bx64 connection panel. It offers 64 individually opto-isolated channels. They can be configured as inputs or outputs in groups of 8. The max. load current for the outputs is 1 A at 100 V. Up to 8 panels can be connected to an M-D40-2 module on the carrier board by means of a ribbon cable, so that up to 512 channels per module and 4,608 channels per carrier board are possible.

A simple way of connecting the ribbon cables coming from the modules e.g. to screw terminals is to use the 5B2x34 connection panel, likewise available in 19" format.

Connector cassettes

The connector cassettes for the MODULAR-4/486 system have been developed in order to facilitate connection of peripherals to the individual modules for the user. They are intended for the 19" format with 3 HE modules, and can be installed in a 19" tabletop case or in a 19" installation frame. Both



19" tabletop case for MODULAR-4 connector cassettes

the tabletop case and the installation frame can accommodate up to 6 connector cassettes or blanking plates.
The illustration shows the tabletop housing with various connector cassettes with male or female D-Sub connectors, and also on with 60 (= 5 x 12) screw terminals. The connector cassette installed on the very left is intended, for example, for connecting the M-COM-8 serial interface module. Each of the 8 serial RS-232 interfaces of this module is thus available at a 9-pole D-Sub connector with standard assignments.

SORCUS Computer GmbH
Im Breitspiel 11, D-69126 Heidelberg
Telefon +49-6221-32060, Telefax +49-6221-320666, Hotline +49-6221-320632
www.sorcus.com

SORCUS