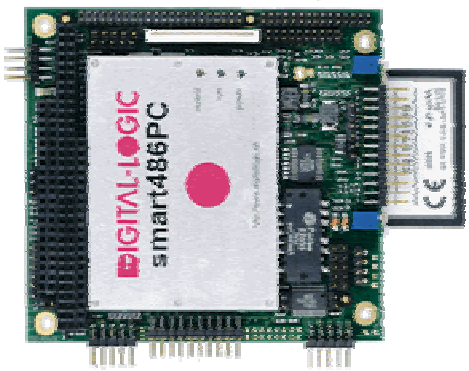
DIGITAL-LOGIC
smart embedded computers

TECHNICAL USER'S MANUAL FOR:

MICROSPACE®

PC/104

MSM486SE/SEV with smartModule SM486PC



Nordstrasse 11/F
CH- 4542 Luterbach
Tel.: **++41 (0)32 681 58 00**
Fax: **++41 (0)32 681 58 01**
Email: **support@digitallogic.com**
Homepage: **http://www.digitallogic.com**

ATTENTION:
All information in this manual and the product are subject to change without prior notice.

REVISION HISTORY:

Prod.-Serialnumber: From: To:	Product Version	BIOS Version	Doc. Version	Date/Vis:	Modification: Remarks, News, Attention:
860xxxxxxxx	***V1.0***	***V2.21***	***V0.90***	***02.98 FK***	***New Board Version, New Manual***
			V0.91	03.99 TS	Related APP-NOTES
			V0.92	04.99 JM	Designs added
	V1.1	***V2.21***	***V1.00***	***04.99 FK***	***Detail modified and added***
	V1.1A	***V2.21***	***V1.10***	***07.99 FK***	***J17 modified to 44pin high RM2.0***
			V1.11	08.99 JM	Ethernet E2PROM values
		V2.22	V1.12	09.99 JM	Internal COM1 description improved
		V2.24	V1.13	11.99 FK	Master/Slave J40
		V2.24	V1.14	12.99 FK	Small modifications
	V1.1	V2.24A	V1.15	01.00 STP	Jumpers modified, new design
	V1.1	V2.24A	V1.16	02.00 STP	Former design added, minor corrections
	V1.1 V1.2	V2.24A	V1.17	04.2000 STP	LAN sample added, connectors, minor corrections
	V1.1 V1.2	V2.24A	V1.18	06.2000 STP	BIOS setup, RS485, J17, J51, Jumperlist, FDD->LPT, DOS NOVELL, minor correction
	V1.1 V1.2	V2.24A	V1.19	10.2000 STP	J51 new pin order, J3, J9 moved, new design, SE= ISA/16bit, new address and logo, thermopicture, etc
	V1.1 V1.2	V2.25	V1.2	04.2001 STP	Minor corrections, watchdog. thermo pictures, 16bit RAM, etc

READ CHAPTER 2.10 TO UNDERSTAND THE ELAN400 INCOMPATIBILITIES COMPARED TO THE STANDARD PC-AT DESIGN !

Table of Contents

1 PREFACE

This manual is for integrators and programmers of systems based on the MICROSPACE card family. It contains information on hardware requirements, interconnections, and details of how to program the system. The specifications given in this manual were correct at the time of printing; advances mean that some may have changed in the meantime. If errors are found, please notify DIGITAL-LOGIC AG at the address shown on the title page of this document, and we will correct them as soon as possible.

1.1 How to use this Manual

This manual is written for the original equipment manufacturer (OEM) who plans to build computer systems based on the single board MICROSPACE-PC. It provides instructions for installing and configuring the MSM486SE/SEV or MSM386SN board, and describes the system and setup requirements.

1.2 Trademarks

Chips & Technologies	SuperState R
MICROSPACE, MicroModule	DIGITAL-LOGIC AG
DOS Vx.y, Windows	Microsoft Inc.
PC-AT, PC-XT	IBM
NetWare	Novell Corporation
Ethernet	Xerox Corporation
DR-DOS, PALMDOS	Digital Research Inc. / Novell Inc.
ROM-DOS	Datalight Inc.

1.3 Disclaimer

DIGITAL-LOGIC AG makes no representations or warranties with respect to the contents of this manual and specifically disclaims any implied warranty of merchantability or fitness for any particular purpose. DIGITAL-LOGIC AG shall under no circumstances be liable for incidental or consequential damages or related expenses resulting from the use of this product, even if it has been notified of the possibility of such damage. DIGITAL-LOGIC AG reserves the right to revise this publication from time to time without obligation to notify any person of such revisions. If errors are found, please contact DIGITAL-LOGIC AG at the address listed on the title page of this document.

1.4 Who should use this Product

- Electronic engineers with know-how in PC-technology.
- Without electronic know-how we expect you to have questions. This manual assumes, that you have a general knowledge of PC-electronics.
- Because of the complexity and the variability of PC-technology, we can't give any warranty that the product will work in any particular situation or combination. Our technical support will help you to may find a solution.
- Pay attention to the electrostatic discharges. Use a CMOS protected workplace.
- Power supply OFF when you are working on the board or connecting any cables or devices.

This is a high technology product.
You need know-how in electronics and PC-technology to install the system !

1.5 Recycling Information

Hardware:

- Print: epoxy with glass fiber
wires are of tin-plated copper

- Components: ceramics and alloys of gold, silver
check your local electronic recycling

Software:

- no problems: re-use the diskette after formatting

1.6 Technical Support

1. **Contact your local DIGITAL-LOGIC Technical Support in your country first !**

2. Use the Internet Support Request form at http://www.digitallogic.com -> Support -> Support Request Form

3. Send a FAX or an E-mail to DIGITAL-LOGIC AG with a description of your problem.

   DIGITAL-LOGIC AG
   smartModule DesignIn Center
   Nordstr. 11/F
   CH-4542 Luterbach (SWITZERLAND)
   Fax: ++41 32 681 58 01
   E-Mail: support@digitallogic.com
   Internet www.digitallogic.com

➔ Support requests will only be accepted with detailed information of the product (BIOS-, Board- Version) !

1.7 Limited Warranty

DIGITAL-LOGIC AG warrants the hardware and software products it manufactures and produces to be free from defects in materials and workmanship for one year following the date of shipment from DIGITAL-LOGIC AG, Switzerland. This warranty is limited to the original purchaser of product and is not transferable.

During the one year warranty period, DIGITAL-LOGIC AG will repair or replace, at its discretion, any defective product or part at no additional charge, provided that the product is returned, shipping prepaid, to DIGITAL-LOGIC AG. All replaced parts and products become property of DIGITAL-LOGIC AG.

Before returning any product for repair, customers are required to contact the company.

This limited warranty does not extend to any product which has been damaged as a result of accident, misuse, abuse (such as use of incorrect input voltages, wrong cabling, wrong polarity, improper or insufficient ventilation, failure to follow the operating instructions that are provided by DIGITAL-LOGIC AG or other contingencies beyond the control of DIGITAL-LOGIC AG), wrong connection, wrong information or as a result of service or modification by anyone other than DIGITAL-LOGIC AG. Neither, if the user has not enough knowledge of these technologies or has not consulted the product manual or the technical support of DIGITAL-LOGIC AG and therefore the product has been damaged.

Except, as expressly set forth above, no other warranties are expressed or implied, including, but not limited to, any implied warranty of merchantability and fitness for a particular purpose, and DIGITAL-LOGIC AG expressly disclaims all warranties not stated herein. Under no circumstances will DIGITAL-LOGIC AG be liable to the purchaser or any user for any damage, including any incidental or consequential damage, expenses, lost profits, lost savings, or other damages arising out of the use or inability to use the product.

2 OVERVIEW

2.1 Standard Features

The MICROSPACE PC/104 is a miniaturized modular device incorporating the major elements of a PC/AT compatible computer. It includes standard PC/AT compatible elements, such as:

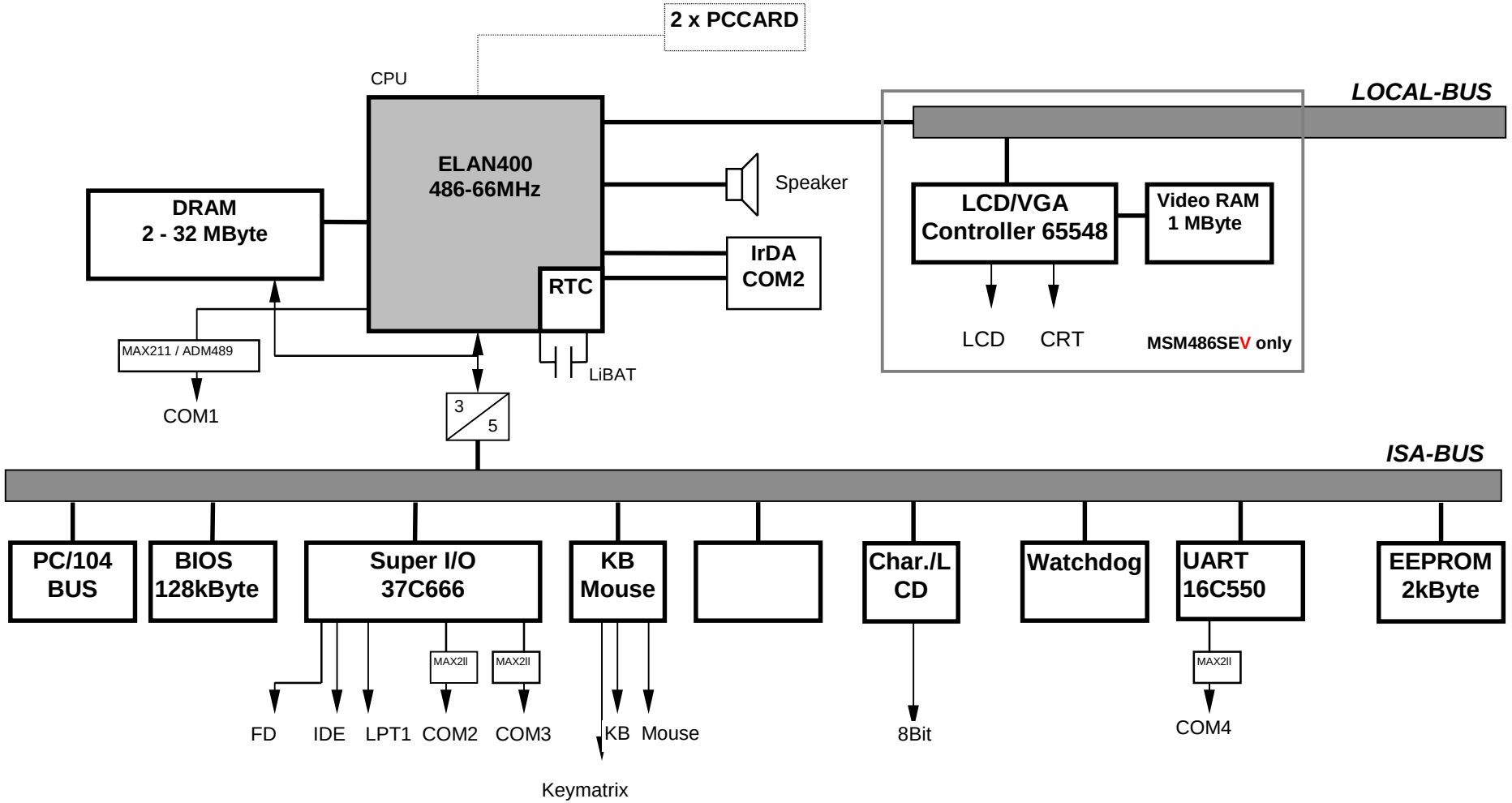
- Powerful ELAN400 CPU with 486-66MHz core
- BIOS ROM
- DRAM 2 - 32 MByte
- No second level cache
- Timers
- DMA
- Real-time clock with CMOS-RAM and battery buffer
- LPT1 parallel port
- COM2, COM3, COM4 with each RS232
- COM1 serial port with RS232, RS485 (optional)
- Speaker interface
- AT-keyboard interface or PS/2-keyboard interface
- Floppydisk interface
- AT-IDE harddisk interface
- Only the MSM486SEV Version: VGA/24bit-LCD video interface on VESA Local BUS
- Ethernet LAN 10-Base-T
- PC/104 Embedded BUS
- PS/2 mouse interface

2.2 Unique Features

The MICROSPACE MSM486SE/SEV includes all standard PC/AT functions plus unique DIGITAL-LOGIC AG enhancements, such as:

- Compact Flashdisk socket
- SVGA/LCD interface
- Low power, 3 watt 3.3V CPU
- Single 5 volt supply
- Watchdog with Power-fail
- EEPROM for setup and configuration
- UL approved parts
- Onboard keymatrix controller
- PC-CARD INTEL 365SL compatible controller for 2 Slot’s onboard
- The LPT IRQ may be disabled to be available for external applications
- The FD IRQ may be disabled to be available for external applications
- FDD redirect to LPT

2.3 MSM486SE/V Block Diagram



2.4 MSM486SE/SEV specifications

CPU:		
CPU 486:	AMD 400™ compatible with 486-66MHz	
Mode:	Real / Protected	
Compatibility:	8086 – 80386	
1. Level Cache:	8 kByte write-back	
Word Size:	16 Bits	
Physical Addressing:	24 lines	
Virtual Addressing:	64 Mbytes	
Clock Rates:	10, 33, 66 MHz selectable	
Socket Standard:	smartModule-486PC	
Math. Coprocessor:		
	not available on AMD-ELAN400 CPU	
Power Management:		
available	clock switching, sleep, possible controlled power-up, inactivity-auto powerdown	
DMA:		
8237A comp.	2 channels 8 Bits	
Interrupts:		
8259 comp.	8 + 2 levels PC compatible	
Timers:		
8254 comp.	3 programmable counter/timers	
Memory:		
DRAM	on the MSM486SE: 2, 8 MByte directly soldered on the SM486PC on the MSM486SEV: 4, 32 Mbyte directly soldered on the SM486PC	
Video: (only SEV version)		
Controller:	65548 from C&T	
BUS:	with the 32bit VESA Local Bus	
Enhanced BIOS:	VGA / LCD BIOS	
Memory:	VRAM onboard: 1MByte	
CRT-Monitor:	VGA, SVGA up to 768 x 1024 pixels 16/256 colors	
Flatpanel:	TFT:	640 x 480 with 8/16/256 colors
	STN:	640 x 480 monochrome
	STN:	640 x 480 with 256 colors
	Plasma:	up to 1280 x 1024
	EL:	640 x 350 , 640 x 480, 768 x 1024 pixels
Controller Modes:	CRT only; Flatpanel only or simultaneous CRT and Flatpanel	
LCD-BIAS voltage:	not available onboard	
Drivers:	Windows	
Mass Storage:		
FD:	Floppy disk interface, for max. 1 floppy with 26pin connector	
HD:	IDE interface, AT - Type, for max. 2 harddisks, 44pin connector, for 1.3, 1.8 and 2.5" harddisk with 44pin IDE	

Sockets SSD:

1st socket:	READ/WRITE ROMDisk, Flashdisk 2Mbyte in the SM-486PC
IDE:	Compact Flash Card

Standard AT Interfaces:

Serial:

Name	FIFO	IRQs	I/O	Modes available
COM1	yes	IRQ4/11	3F8	RS232/ opt. RS422/RS485
COM2	yes	IRQ3	2F8	RS 232C
COM3	yes	IRQ4	3E8	RS 232C
COM4	yes	IRQ3/14	2E8	RS 232C

COM1 and COM3 can not share the IRQ4 line simultaneously
(Baudrates: 50 – 115 KBaud programmable)

Parallel:	LPT1 printer interface, Modes: SPP (output) , EPP (bidir.)
Keyboard:	AT- or PS/2-keyboard
Mouse:	PS/2
Speaker:	0.1 W output drive
RTC:	integrated into the ELAN400, RTC with CMOS-RAM 128Byte
Backup current:	<5 µA at 3V
Battery:	Lithium 410mAh (48mAh until boardversion 1.1) onboard or external

Supervisory:

Watchdog	LTC1232 with power-fail detection, strobe time max. 1 sec.

BUS:

PC/104:	IEEE-996 standard bus, buffered with 24mA
Clock:	8 MHz

Ethernet-LAN:

10-BASE-T with SMSC91C96 controller

Power Supply:

Working:	5 Volts ± 5%, 3.3V onboard switch mode regulator	
Power Rise Time:	>100µs (0V --> 4,75V)	
Maximum Voltage:	below 60mV before powerup the system	
Current:	SEV16: 950mA	SE16: 1000mA
	SEV16: 600mA in suspend mode	SE16: 360mA in suspend mode

Physical Characteristics:

Dimensions:	Length:	90 mm
	Depth:	96 mm
	Height:	20 mm
Weight:	SEV: 120 gr / SE: 110 gr	
PCB Thickness:	1.6 mm / 0.0625 inches nominal	
PCB Layer:	Multilayer	

Operating Environment:

Relative Humidity:	5 - 90% non condensing		
Vibration:	5 to 2000 Hz		
Shock:	10 G		
Temperature:	Operating:	Standard version:	-25°C to +70°C
		Extended version:	-40°C to +85°C T.B.A
	Storage:	-55°C to +85 °C	

EMI / EMC (IEC1131-2 refer MIL 461/462):

ESD Electro Static Discharge:	IEC 801-2, EN55101-2, VDE 0843/0847 Part 2 metallic protection needed separate Ground Layer included 15 kV single peak
REF Radiated Electromagnetic Field:	IEC 801-3, VDE 0843 Part 3, IEC770 6.2.9. not tested
EFT Electric Fast Transient (Burst):	IEC 801-4, EN50082-1, VDE 0843 Part 4 250V - 4kV, 50 ohms, Ts=5ns Grade 2: 1KV Supply, 500 I/O, 5Khz
SIR Surge Immunity Requirements:	IEC 801-5, IEEE587, VDE 0843 Part 5 Supply: 2 kV, 6 pulse/minute I/O: 500 V, 2 pulse/minute FD, CRT: none
High-frequency radiation:	EN55022

Compatibility:

MSM486SE/SEV:	mechanically compatible to our MSMx86 Boards and to all other PC/104 boards

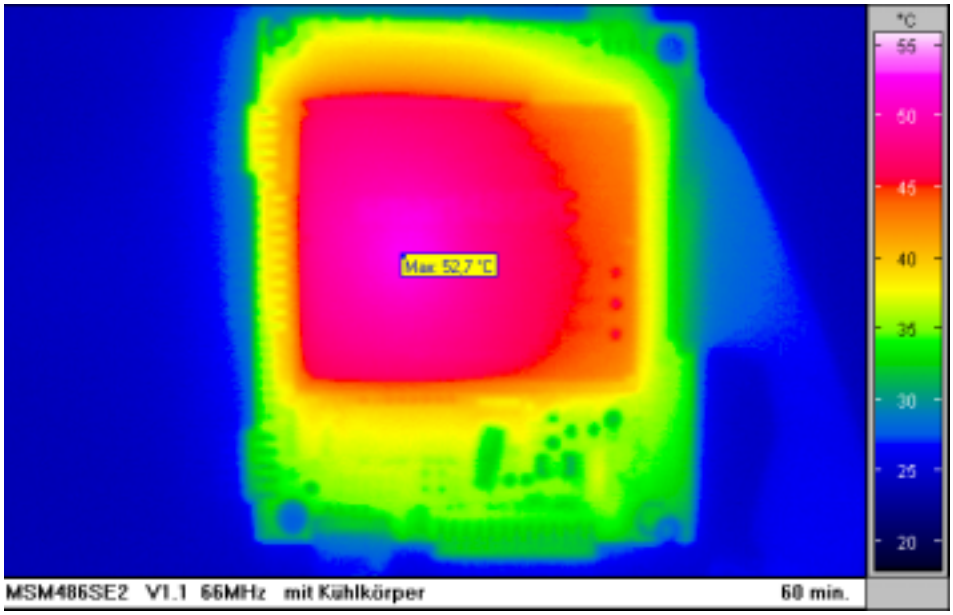
Any information is subject to change without notice.

2.5 Ordering codes

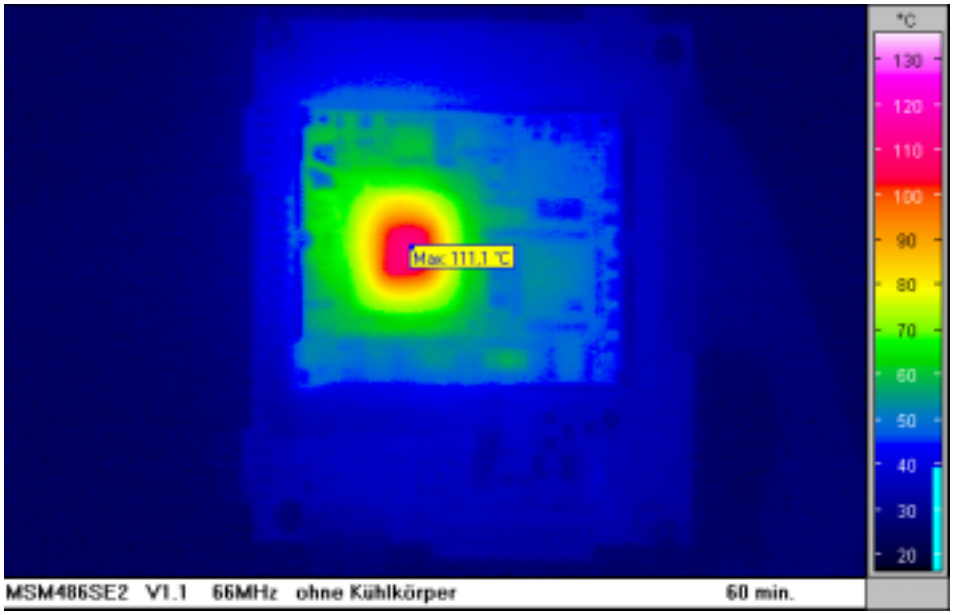
MSM486SE2	with 2 MB DRAM, without VGA/LCD -25°C to 70°C BurnIn proofed, at 66MHz
MSM486SE8	with 8 MB DRAM, without VGA/LCD -25°C to 70°C BurnIn proofed, at 66MHz
MSM486SEV4	with 4 MB DRAM, with VGA/LCD -25°C to 70°C BurnIn proofed, at 66MHz
MSM486SEV16	with 16 MB DRAM, with VGA/LCD -25°C to 70°C BurnIn proofed, at 66MHz
-E48	Extended temperature range -40°C to +85°C BurnIn proofed, at 33MHz
MSM486SVPC	Adapter for dualslot P-CARD sockets
MSMSEVCK	Monitor cable, only for MSM486SEV until boardversion V1.1

2.6 Thermoscan

With cooler:



No cooler:



2.7 BIOS History

Version:	Date:	Status:	Modifications:
2.10	**06.98**	**released**	**Y2K tested**
2.12	08.98	released	1. Remote is working 2. CS selecting in the setup 3. PGP0 mapping in the setup 4. PIRQ mapping in the setup 5. Y2K fixed 6. Switch for disable the PS2 mouse in the setup 7.Memory-Bus width displayed 1. Show no access message 2. Correct the CPU type
2.12B	10.98	beta	INT19 problem for FFS fixed
2.13	11.98	released	Included switch J3 for 32/16bit memory Included switch J4 for ISA/VESA VGA
2.14	01.99	released	¼ VGA Graphic included CGA-Text is working
2.20	02.99	released	Disable the Floppy-IRQ if no FD used in the setup. The IRQ6 may be used for other applications . Disable the LPT-IRQ7 in the setup.
2.21	03.99	released	FFS V7.02 included APM Bug fixed
2.22	08.99	released	FD on LPT support added. ¼-VGA problem solved
2.24A	18.12.1999	released	DTR WDOG Disable for the Mem-Test moved. Released also for SM486PCX. VGA-Problem solved on sm486PCX
V2.25	27.10.2000	released	FFS 7.03, EEPROM support upgraded

2.8 This product is "YEAR 2000 CAPABLE"

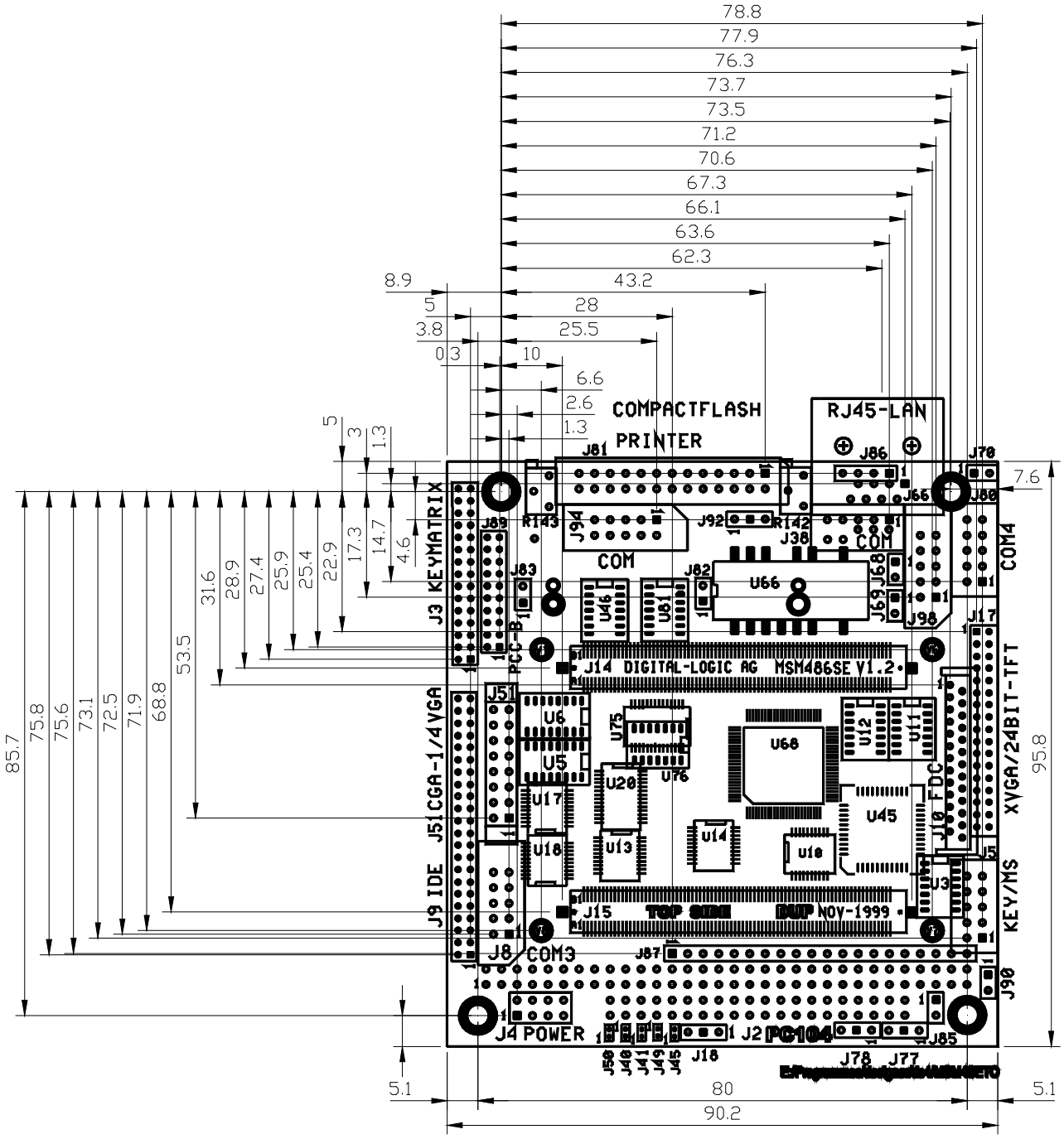
This DIGITAL-LOGIC product is "YEAR 2000 CAPABLE". This means, that upon installation, it accurately stores, displays, processes, provides and/or receives date data from, into, and between 1999 and 2000, and the 20. and 21. centuries, including leap year calculations, provided that all other technology used in combination with said product properly exchanges date data with it. DIGITAL-LOGIC makes no representation about individual components within the product should be used independently from the product as a whole. You should understand that DIGITAL-LOGIC's statement that an DIGITAL-LOGIC product is "YEAR 2000 CAPABLE" means only that DIGITAL-LOGIC has verified that the product as a whole meet this definition when tested as a stand-alone product in a test lab, but dies not mean that DIGITAL-LOGIC has verified that the product is "YEAR 2000 CAPABLE" as used in your particular situation or configuration. DIGITAL-LOGIC makes no representation about individual components, including software, within the product should they be used independently from the product as a whole.

DIGITAL-LOGIC customers use DIGITAL-LOGIC products in countless different configurations and in conjunction with many other components ans systems, and DIGITAL-LOGIC has no way to test wheter all those configurations and systems will properly handle the transition to the year 2000. DIGITAL-LOGIC encourages its customers and others to test whether their own computer systems and products will properly handle the transition to the year 2000.

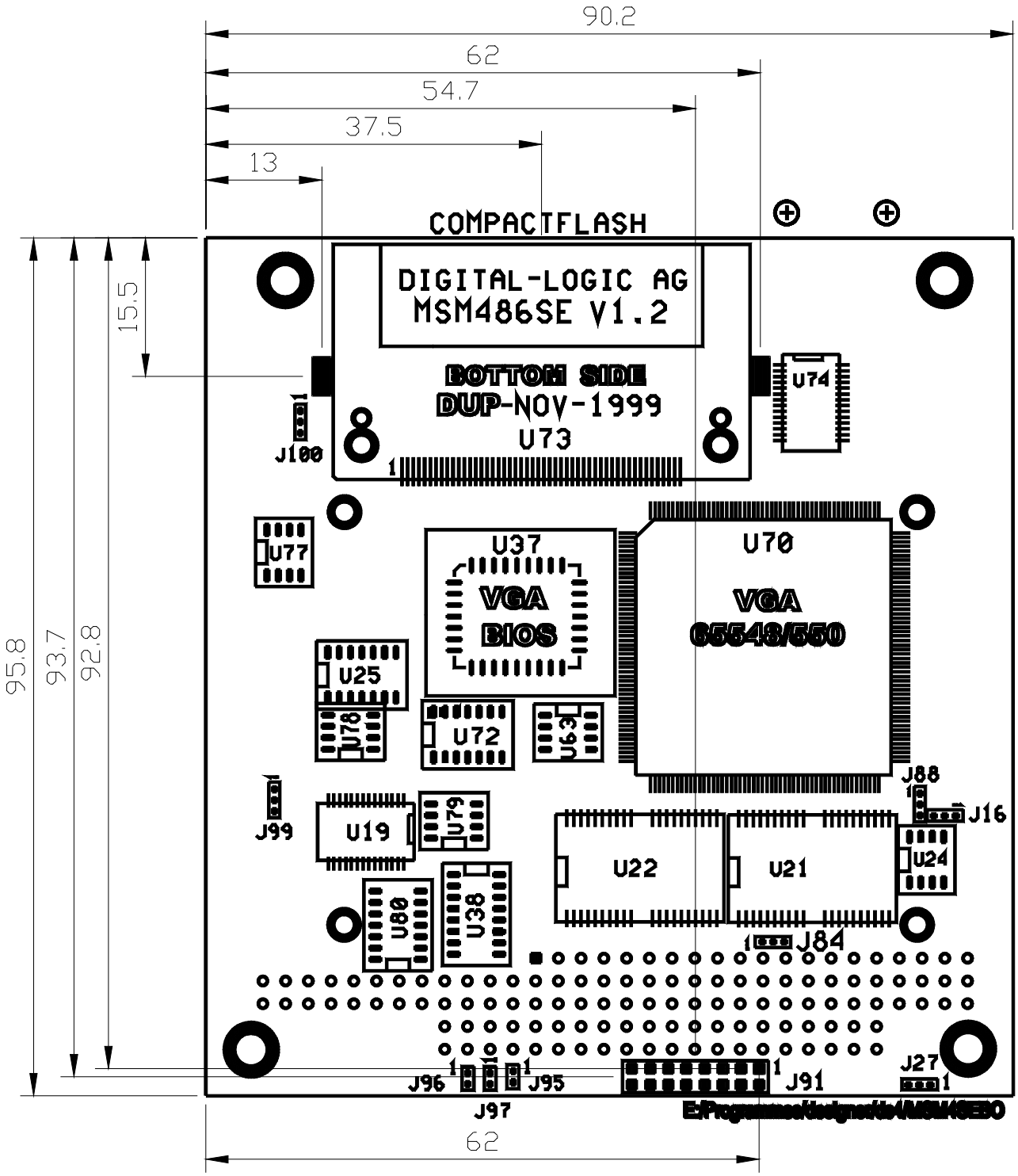
The only proper method of accessing the date in systems is indirectly from the Real-Time-Clock via the BIOS. The BIOS in DIGITAL-LOGIC computerboards contain a century checking and maintenance feature the checks the laest two significant digits of the year stored in the RTC during each BIOS request (INT 1A) to read the date and, if less than '80' (i.e. 1980 is the first year supported by the PC), updates the century byte to '20'. This feature enables operating systems and applications using BIOS date/time services to reliably manipulate the year as a four-digit value.

2.9 Mechanical Dimensions MSM486SE/SEV

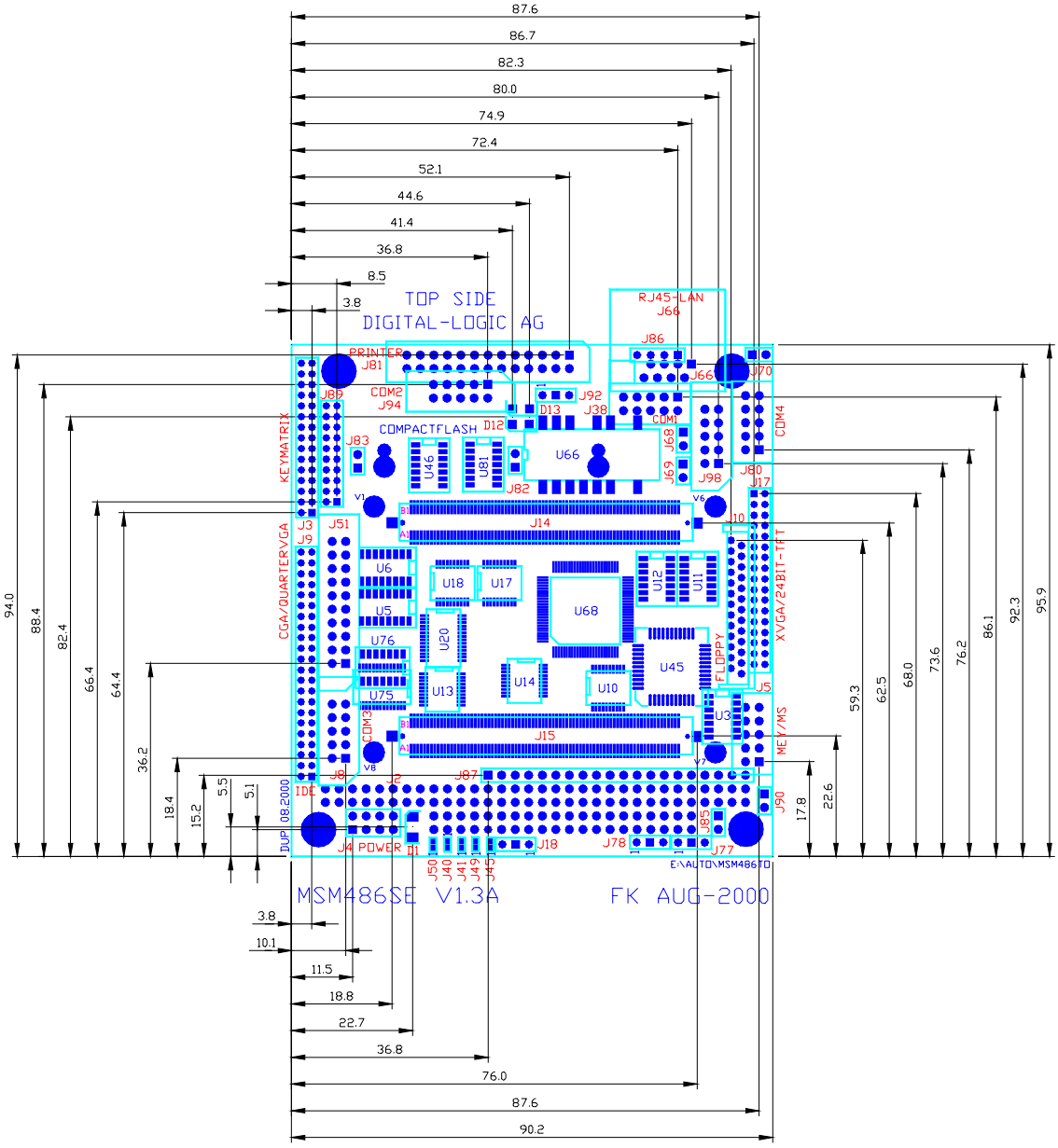
2.9.1 Boardversion V1.2



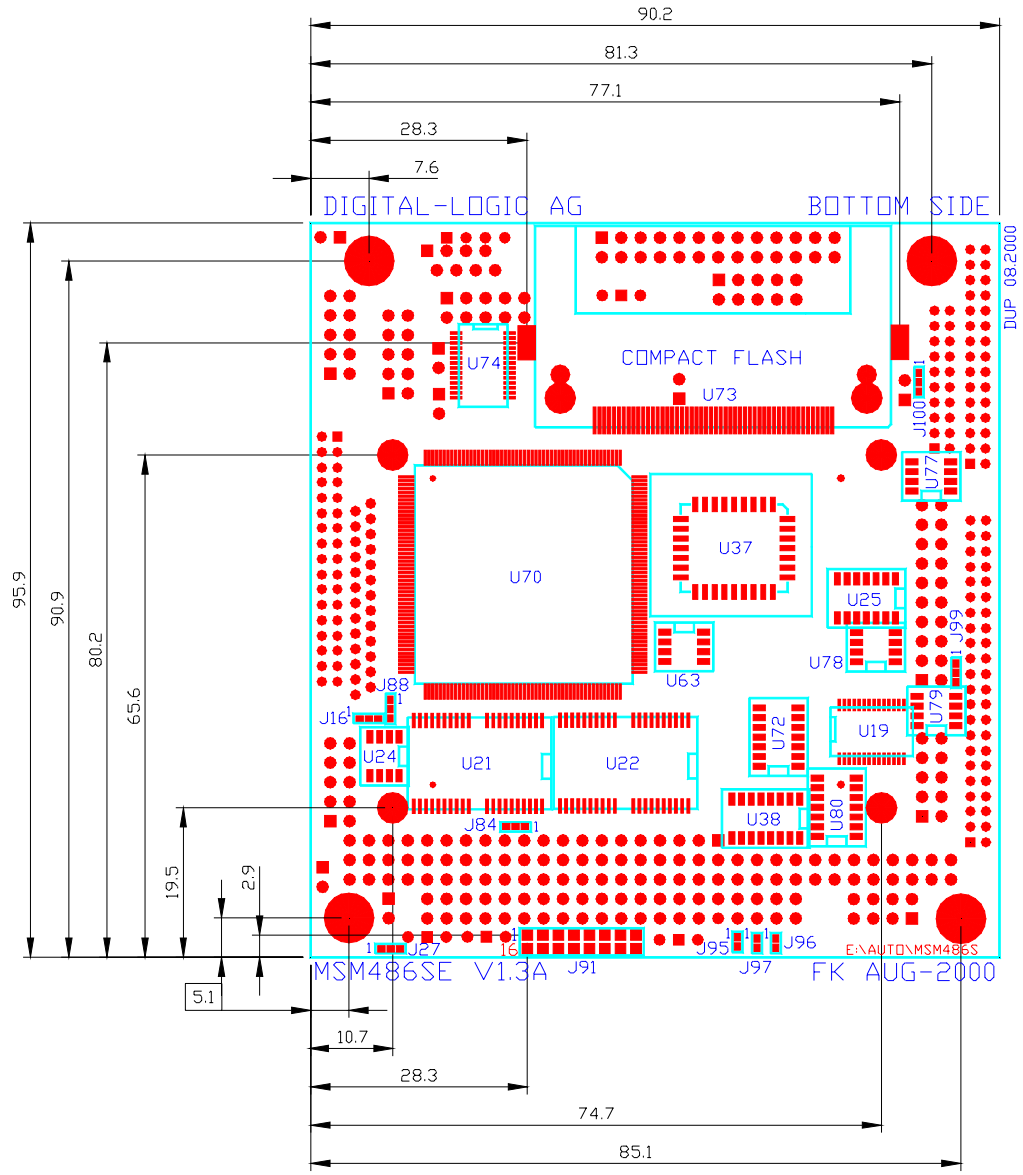
2.9.2 Boardversion V1.2



2.9.3 Boardversion V1.3



2.9.4 Boardversion V1.3



2.10 Errata for ELAN400 Rev. 4

No:	Item:	MSM486SE/SEV:
E1:	PCCARD Reset Code Error using ROMCS0	not used onboard
E2:	Burst ROM read with ZWS not ok	not used onboard
E3:	DMA in Suspend not functional as event	no priority
E4:	Port 70h always internal access	no influence
E5:	PMU does not change the modes	must be checked
E6:	IrDA Anomalies	1.15MB does not work
E7,E8,E9,E10	are solved in ES2 chips	
E11:	Keyboard Row 14 pol. reversed	not used onboard
E12:	solved in ES2 chips	
E13:	Access to VGA Memory activity ignored	must be checked
E14:	PCCARD and LPT activities problems	not used onboard

2.11 Related Application Notes

#	Description
77	Ethernet EEPROM values do not match
78	FFS format tools
79	Soundcontroller with the ELAN400
82	SM486PC 16 bit problem
84	Power consumption on Pentium / any other boards with attached drives (HDD, CD)
85	LAN TABLE update
90	4 COM ports, 4 different IRQ's
92	MSM486SE/SEV watchdog
93	SM486PC Vesa- Local Bus detection

➔ *Application Notes are availble at http://www.digitallogic.com ->support, or on any Application CD from DIGITAL-LOGIC.*

2.12 Incompatibilities of the ELAN400 versus the AT-PC

Since the ELAN400 is single chip PC, all components are direct integrated into one silicium. The ELAN400 is a very complex circuit and therefore all known problems or some warnings are published in this chapter.

2.12.1 Number of Interruptlines

- In a standard PC design, there are 15 IRQ lines available.
- In the ELAN400, the IRQ 9, 10, 11, 15 are available with the redirector program!
  Since BIOS V2.11, the IRQ redirection is selectable in the BIOS setup.

2.12.2 Number of DMA-channels

- In a standard PC design, there are 7 DMA-channels available.
- In the ELAN400, there are only two DMA-channels available.
  DMA2 and DMA1: DMA2 is used for the floppy controller, DMA1 is free for the user.

2.12.3 The ELAN400 has no MASTER signal

This line is open.

2.12.4 The ELAN400 has no REFRESH signal

Pulled up to VCC with 1k resistor.

2.12.5 The ELAN400 has an internal PCCARD controller, select an alternative address

The internal PC-CARD controller operates at the address 3E0h / 3E1h. Using the MSM486SE/SEV board with an MSMJ104 PCCARD board needs the following modification on the MSMJ104 board:

J18 = 2-3 and J14 = 1-2 -> Base Address is now 3E2h

The MMCD.SYS receives the option /B:3E2h

Boot Option for selecting the Index Base addresses on the MSM104J board:

J18	**J14:**			
INTR:	SPKROUT:	Index Base:	I/O Address:	Comment:
1-2 (vcc)	1-2 (vcc)	00h	3E0h/3E1h	DEFAULT
1-2 (vcc)	*2-3 (gnd)*	*80h*	*3E0h/3E1h*	*
2-3 (gnd)	1-2 (vcc)	00h	3E2h/3E3h	(needed on ELAN400 boards)
2-3 (gnd)	*2-3 (gnd)*	*80h*	*3E2h/3E3h*	*

** only available on VG468 controller !*

2.12.6 COM1 and COM3 port with IRQ4

COM1 and COM3 cannot share the IRQ4 at the same time. If you are using both interfaces, one is working with IRQ4 and the other must be polled by the application software. The IRQ4 may be generated from the internal UART (=COM1) or from the external COM3. Default the IRQ4 is switched to the external COM3. Refer the chapter 3.3.3.

2.13 Installation of the EMM386 Driver

Only the D-Segment is free on the ELAN400 for the EMM386. The config.sys must contain the following commands:

CONFIG.SYS
DEVICE=d:\DOS\HIMEM.SYS
DEVICE=d:\DOS\EMM386.EXE FRAME=D000 VERBOSE 128
DOS=HIGH,UMB
LASTDRIVE=Z
FILES=40
STACKS=9,256

2.14 PC/104 Bus Signals

AEN, output
Address Enable is used to degate the microprocessor and other devices from the I/O channel to allow DMA transfers to take place. **low = CPU Cycle , high = DMA Cycle**

BALE, output
Address Latch Enable is provided by the bus controller and is used on the system board to latch valid addresses and memory decodes from the microprocessor. This signal is used so that devices on the bus can latch LA17..23. The SA0..19 address lines latched internally according to this signal. BALE is forced high during DMA cycles.

/DACK[0..3] output
DMA Acknowledge 0 to 3 are used to acknowledge DMA requests (DRQO through DRQ3). They are **active low**. This signal indicates that the DMA operation can begin.On ELAN400 only DMA1 and DMA2 are available.

DRQ[0..3], input
DMA Requests 0 through 3 are asynchronous channel requests used by peripheral devices and the I/O channel microprocessors to gain DMA service (or control of the system). A request is generated by bringing a DRQ line to an active level. A DRQ line must be held high until the corresponding DMA Request Acknowledge (DACK/) line goes active. DRQO through DRQ3 will perform 8-Bit DMA transfers; On ELAN400 only DRQ1 and DRQ2 are available.

/IOCHCK, input
IOCHCK/ provides the system board with parity (error) information about memory or devices on the I/O channel. **low = parity error, high = normal operation**

IOCHRDY, input
I/O Channel Ready is pulled low (not ready) by a memory or I/O device to lengthen I/O or memory cycles. Any slow device using this line should drive it low immediately upon detecting its valid address and a Read or Write command. Machine cycles are extended by an integral number of one clock cycle (67 nanoseconds). This signal should be held in the range of 125-15600ns. **low = wait, high = normal operation**

/IOCS16, input
I/O 16 Bit Chip Select signals the system board that the present data transfer is a 16-Bit, 1 wait-state, I/0 cycle. It is derived from an address decode. /IOCS16 is **active low** and should be driven with an open collector (300 ohm pull-up) or tri-state driver capable of sinking 20mA. The signal is driven based only on SA15-SAO (not /IOR or /IOW) when AEN is not asserted. In the 8 Bit I/O transfer, the default transfers a 4 wait-state cycle.

/IOR, input/output
I/O Read instructs an I/O device to drive its data onto the data bus. It may be driven by the system microprocessor or DMA controller, or by a microprocessor or DMA controller resident on the I/O channel. This signal is **active low**.

/IOW, input/output
I/O Write instructs an I/O device to read the data on the data bus. It may be driven by any microprocessor or DMA controller in the system. This signal is **active low**.

IRQ[3 - 7,12,14], input
These signals are used to tell the microprocessor that an I/O device needs attention. An interrupt request is generated when an IRQ line is **raised from low to high**. The line must be held high until the microprocessor acknowledges the interrupt request.

/Master, input
This signal is not available on ELAN400 designs.

/MEMCS16, input
MEMCS16 Chip Select signals the system board if the present data transfer is a 1 wait-state, 16-Bit, memory cycle. It must be derived from the decode of LA17 through LA23. /MEMCS16 should be driven with an open collector (300 ohm pull-up) or tri-state driver capable of sinking 2OmA.

/MEMR input/output
These signals instruct the memory devices to drive data onto the data bus. /MEMR is active on all memory read cycles. /MEMR may be driven by any microprocessor or DMA controller in the system. When a microprocessor on the I/0 channel wishes to drive /MEMR, it must have the address lines valid on the bus for one system clock period before driving /MEMR active. These signals are **active low**.

/MEMW, input/output
These signals instruct the memory devices to store the data present on the data bus. /MEMW is active in all memory read cycles. /MEMW may be driven by any microprocessor or DMA controller in the system. When a microprocessor on the I/O channel wishes to drive /MEMW, it must have the address lines valid on the bus for one system clock period before driving /MEMW active. Both signals are **active low.**

OSC, output
Oscillator (OSC) is a high-speed clock with a 70 nanosecond period (14.31818 MHz). This signal is not synchronous with the system clock. It has a 50% duty cycle. OSC starts 100µs after reset is inactive.

RESETDRV, output
Reset Drive is used to reset or initiate system logic at power-up time or during a low line-voltage outage. This signal is active high. When the signal is active all adapters should turn off or tri-state all drivers connected to the I/O channel. This signal is driven by the permanent Master.

/REFRESH, input/output
This signal is not available on ELAN400 designs, this will pullup with 1k onboard.

SAO-SA19, LA17 - LA23 input/output
Address bits 0 through 19 are used to address memory and I/0 devices within the system. These 20 address lines, allow access of up to 1MBytes of memory. SAO through SA19 are gated on the system bus when BALE is high and are latched on the falling edge of BALE. LA17 to LA23 are not latched and addresses the full 16 MBytes range. These signals are generated by the microprocessors or DMA controllers. They may also be driven by other microprocessor or DMA controllers that reside on the I/0 channel. The SA17-SA23 are always LA17-LA23 address timings for use with the MSCS16 signal. This is advanced AT96 design. The timing is selectable with jumpers LAxx or SAxx.

/SBHE, input/output
Bus High Enable (system) indicates a transfer of data on the upper byte of the data bus, XD8 through XD15. Sixteen-Bit devices use /SBHE to condition data-bus buffers tied to XD8 through XD15.

SD[O..15], input/output
These signals provide bus bits 0 through 15 for the microprocessor, memory, and I/0 devices. DO is the least-significant Bit and D15 is the most significant Bit. All 8-Bit devices on the I/O channel should use DO through D7 for communications to the microprocessor. The 16-Bit devices will use DO through D15. To support 8-Bit device, the data on D8 through D15 will be gated to DO through D7 during 8-Bit transfers to these devices; 16-Bit microprocessor transfers to 8-Bit devices will be converted to two 8-Bit transfers.

/SMEMR input/output
These signals instruct the memory devices to drive data onto the data bus for the first MByte. /SMEMR is active on all memory read cycles. /SMEMR may be driven by any microprocessor or DMA controller in the

system. When a microprocessor on the I/0 channel wishes to drive /SMEMR, it must have the address lines valid on the bus for one system clock period before driving /SMEMR active. The signal is **active low**.

/SMEMW, input/output
These signals instruct the memory devices to store the data present on the data bus for the first MByte. /SMEMW is active in all memory read cycles. /SMEMW may be driven by any microprocessor or DMA controller in the system. When a microprocessor on the I/O channel wishes to drive /SMEMW, it must have the address lines valid on the bus for one system clock period before driving /SMEMW active. Both signals are **active low.**

SYSCLK, output
This is a 8 MHz system clock. It is a synchronous microprocessor cycle clock with a cycle time of 167 nanoseconds. The clock has a 66% duty cycle. This signal should only be used for synchronization.

TC output
Terminal Count provides a pulse when the terminal count for any DMA channel is reached. The TC completes a DMA-Transfer. This signal is expected by the onboard floppy disk controller. Do not use this signal, because it is internally connected to the floppy controller.

/OWS, input
The Zero Wait State (/OWS) signal tells the microprocessor that it can complete the present bus cycle without inserting any additional wait cycles. In order to run a memory cycle to a 16-Bit device without wait cycles, /OWS is derived from an address decode gated with a Read or Write command. In order to run a memory cycle to an 8-Bit device with a minimum of one-wait states, /OWS should be driven active one system clock after the Read or Write command is active, gated with the address decode for the device. Memory Read and Write commands to an 8-Bit device are active on the falling edge of the system clock. /OWS is **active low** and should be driven with an open collector or tri-state driver capable of sinking 20mA.

12V +/- 5%
Used only for the flatpanel supply.

GROUND = 0V
Used for the entire system.

VCC, +5V +/- 0.25V
for logic and harddisk/floppy supply.

2.15 Expansion Bus

The Bus currents are:

Output Signals:	IOH:	IOL:
D0 - D16	24 mA	24 mA
A0 - A23	24 mA	24 mA
MR, MW, IOR, IOW, RES, ALE, AEN, C14	24 mA	24 mA
DACKx, DRQx, INTx, PSx, OPW	24 mA	24 mA

Output Signals:	Logic Family:	Voltage:
Input Signals:	ABT-Logic ViH (min.) = 2.15 V	ABT-Logic Vil (max.) = 0.85 V

2.16 High frequency radiation (to meet EN55022)

Since the PC/104 CPU modules are very high integrated embedded computers, no peripheral lines are protected against the radiation of high frequency spectrum. To meet a typical EN55022 requirement, all peripherals, they are going outside of the computer case, must be filtered externaly.

Typical signals, they must be filtered:

Keyboard:	KBCLK, KBDATA, VCC
Mouse:	MSCLK, MSDATA, VCC
COM1/2/3/4:	All serial signals must be filtered
LPT:	All parallel signals must be filtered
CRT:	red,blue,green, hsynch, vsynch must be filtered

Typical signals, they must not be filtered, since they are internaly used:

IDE:	connected to the harddisk
Floppy:	connected to the floppy
LCD:	connected to the internal LCD

1. For peripheral cables:

Use for all DSUB connector a filtered version. Select carefully the filter specifications. Place the filtered DSUB connector directly frontside and be sure that the shielding makes a good contact with the case.

9pin	DSUB connector from AMPHENOL:	FCC17E09P	820pF
25pin	DSUB connector from AMPHENOL:	FCC17B25P	820pF

2. For stackthrough applications:

Place on each peripheral signal line, they are going outside, a serial inductivity and after the inductivity a capacitor of 100pF to 1000pF to ground.
In this case, no filtered connectors are needed. Place the filter directly under or behind the onboard connector.

Serial Inductivity:	TDK HF50ACB321611-T	100Mhz, 500mA, 1206 Case
Ground capacitor:	Ceramic Capacitor with 1000pF	

Power supply:

Use a currentcompensated dualinductor on the 5V supply

SIEMENS B82721-K2362-N1 with 3.6A , 0.4mH

3 DETAILED SYSTEM DESCRIPTION

This system has a system configuration based on the ISA architecture. Check the I/O and the Memory map in this chapter.

3.1 Power Requirements

The power is connected through the PC/104 power connector; or the separate power connector on the board. The supply uses only the +5 Volts and ground connection.

Warning: Make sure that the power plug is wired correctly before supplying power to the board! A built-in diode protects the board against reverse polarity.

Tolerance of 5V supply: 5 volts ± 5%; Power-fail signal starts at ± 5 % of 5V norm and generates a reset status for the MICROSPACE PC.

ATTENTION: With the harddisk connected to the IDE 44pin interface, the power requirement is high. The peak current must be enough to spin up the HD-motor. The typical spin-up current of the harddisk is 0.8 - 1.5Amp at 5V. Too little current will drop the voltage to under 5 volts for a short time. Due to this undervoltage, the system or the harddisk stops or falters. The VGA could also be "snowy".

The precise power requirements of the MICROSPACE MSM486SE/SEV depend on a number of factors, including which functions are present on the board and which peripherals are connected to the board's I/0 ports. For example, AT-keyboards draw their power from the keyboard connector on the MICROSPACE MSM486SE/SEV board, and therefore add keyboard current to the total power drawn by the board from its power supply.

3.1.1 Powersave Modes

		MSM486SE/SEV 1.x
DRAM		8
3.3V Gen.		switched
Hyper speed		66MHz
at 5.0V		710mA
Power		
↓ Powerdn		Ti, Sus, Sw
Low Speed		33MHz
at 5.0V		
Power		
VGA, MAX211		on
↓ Powerdn		Ti, Sus, Sw
↑ Powerup		Ac,Res, Sw
Suspend		1MHz
at 5.0V		360mA
Power		
VGA		off
MAX211, 14MHz		off
↓ Powerdn		Ti, Sus, Sw
↑ Powerup		Ac,Res, Sw
Sleep		0MHz
at 5.0V		
VGA		off
MAX211, 14MHz		off
Keyboard		off
↑ Powerup		Res

In all power modes, the program is resident in the refreshed DRAM!

Others: KBD = 10mA, Floppy = 10mA, HD = 300mA/10mA, Flashdisk = 1mA, VGA = ~300mA

Remarks:	Ti	1s to 24h prog.=	Timer controlled (modifiable in the CMOS-Setup)
	Sus/Res	500ms =	Suspend / Resume signal (hardware)
	Sw	500ms =	Software controlled, by programming a register
	Ac	500ms =	Activity, Keyboard pressed, Mouse, COMx,

Important: **To optimize and fully understand the powersave function of the chipsets, the AMD, INTEL and C&T databooks are needed!**

3.2 CPU, Board and RAMs

3.2.1 CPU of this MICROSPACE Product

Processor:	Type:	Clock:	Landmark MHz:	Landmark Units:
486DX	AMD	33/66 MHz	66 MHz	200

3.2.2 Numeric Coprocessor

Is not integrated in the ELAN 400 CPU.

3.2.3 DRAM Memory

Speed:	70ns
Size:	soldered onboard SOJ DRAMs
Bits:	16 Bit
Capacity:	4, 8, 12, 16 MBytes
Bank:	1, 2, 3, 4

3.3 Interface

3.3.1 Keyboard AT-compatible and PS/2-Mouse

J3

Pin	Signal
Pin 1	Speaker out
Pin 2	Resume Input
Pin 3	Reset Input For generating a reset, switch this pin to GND with a open collector driven device or a TTL device. Internal pullup 1k is connected onboard.
Pin 4	VCC
Pin 5	Keyb. Data
Pin 6	Keyb. Clock
Pin 7	Ground
Pin 8	Ext. Battery
Pin 9	Mouse Clock (PS/2)
Pin 10	Mouse Data (PS/2)

3.3.2 Line Printer Port LPT1

A standard bi-directional LPT port is integrated in the MICROSPACE PC.

Further information about these signals is available in numerous publications, including the IBM technical reference manuals for the PC and AT computers and from some other reference documents.

The current is: IOH = 12 mA IOL = 24mA

3.3.3 Serial Ports COM1-COM2-COM3

The serial channels are fully compatible with 16C550 UARTS. COM1 is the primary serial port, and is supported by the board's ROM-BIOS as the PC-DOS 'COM1' device. The secondary serial port is COM2; it is supported as the 'COM2' device. The internal UART (COM1) and the UPC (COM3) use the same IRQ4. This IRQ4 can not be shared, that means only the internal UART or the COM3 generates interrupt requests. The IRQ4 may be switched with the register D6h. or the tool IRQ4.EXE with the parameter I (=internal) or E (=external). The IRQ4.EXE is located on the tooldisk. If the COM3 is selected to the IRQ5, the communication driver must handle the IRQ5 requests. The BIOS deals only with IRQ3 and IRQ4.

Switch the IRQ4	to internal UART (COM1):	IRQ4 I	(reg D6h = 50h)
	to external SMC UART (COM3)	IRQ4 E	(reg D6h = 54h)

Standard:	COM 3/2:	37C665 (SMC):	2 x 16C550 compatible serial interfaces with RS232C
	COM1:	ELAN400:	1 x 16C550 compatible serial interface with RS232C/485
	COM4:	16C550 RS232C	

Serial Port Connectors COM2, COM3, COM4

Pin	Signal Name	Function	in/out	DB25 Pin	DB9 Pin
1	CD	Data Carrier Detect	in	8	1
2	DSR	Data Set Ready	in	6	6
3	RXD	Receive Data	in	3	2
4	RTS	Request To Send	out	4	7
5	TXD	Transmit Data	out	2	3
6	CTS	Clear to Send	in	5	8
7	DTR	Data TerminalReady	out	20	4
8	RI	Ring Indicator	in	22	9
9	GND	Signal Ground		7	5

The serial port signals are compatible with the RS232C specifications.

The COM of the ELAN (default COM1)

ELAN	port address	IRQ
register int com disabled	Disabled	
register int com enabled	**COM1 3F8h**	4

To make any changes the BIOS must be modified.

The internal UART is hardwired on the address of the COM1 and of the IRQ4. No other modification to other addresses or IRQ numbers are possible.

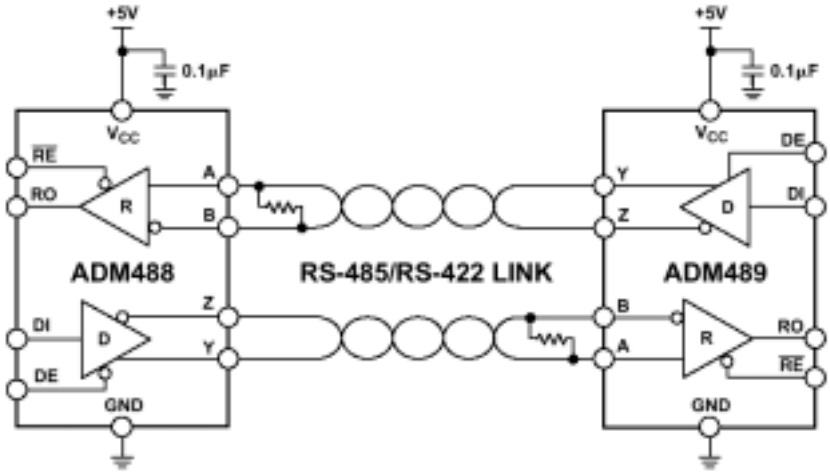
On Board V2.x the SMC37C665 is programmable in the BIOS setup !

3.3.4 Serial Ports RS422 / RS485 on COM1

The RS485 interface is controlled by the RTS/DTS outputs of the COM1 at the address 3FCh. The application must be able to control the RS485 port correctly.

Function:	Output: RS485	Input: RS485	RTS-Output (3FCh Bit1)	DTS-Output (3FCh Bit0)	Remarks:
RS485	Enabled	Disabled	0	0	Transmit Data
RS485	Enabled	Enabled	0	1	TxD & RxD, Loopback
RS485	Disabled	Disabled	1	0	No Bus Access
RS485	Disabled	Enabled	1	1	Receive only Data

Typical application, 4wire



3.3.5 Floppy Disk Interface

The onboard floppy disk controller and ROM-BIOS support one or two floppy disk drives in any of the standard PC-DOS and MS-DOS formats shown in the table .

Supported Floppy Formats

Capacity	Drive size	Tracks	Data rate	DOS version
1.2 MB	5-1/4"	80	500 KHz	3.0 - 6.22
720 K	3-1/2"	80	250 KHz	3.2 - 6.22
1.44 M	3-1/2"	80	500 KHz	3.3 - 6.22

Floppy Interface Configuration

The desired configuration of floppy drives (number and type) must be properly initialized in the board's CMOS - configuration memory. This is generally done by using CTRL + ALT + 'S' at bootup time.

Floppy Interface Connector

The table shows the pinout and signal definitions of the board's floppy disk interface connector. It is identical in pinout to the floppy connector of a standard AT. Note that, as in a standard PC or AT, both floppy drives are jumpered to the same drive select: as the 'second' drive. The drives are uniquely selected as a result of a swapping of a group of seven wires (conductors 10-16) that must be in the cable between the two drives. The seven-wire swap goes between the computer board and drive 'A'; the wires to drive 'B' are unswapped (or swapped a second time). The 26 pin high density (1mm pitch FCC) connector has only one drive and motor select. The onboard jumper defines the drive A: or B:. Default is always A:.

Floppy Disk Interface Technology

We only support CMOS drives. That means that the termination resistors are 1Kohm. 5 1/4“-drives are not recommended (TTL interface).

The 26 pin Connector: FFC/FPC 0.3mm thick 1.0mm (0.039") pitch (MOLEX 52030 Serie)

Floppy Disk Interface Connector

FD26: Pin	Signal Name	Function	in/out
1	VCC	+5 volts	
2	IDX	Index Pulse	in
3	VCC	+5 volts	
4	DS2	Drive Select 2	out
5	VCC	+5 volts	
6	DCHG	Disk Change	in
10	M02	Motor On 2	out
12	DIRC	Direction Select	out
14	STEP	Step	out
16	WD	Write Data	out
17	GND	Signal grounds	
18	WE	Write Enable	out
19	GND	Signal grounds	
20	TRKO	Track 0	in
21	GND	Signal grounds	
22	WP	Write Protect	in
23	GND	Signal grounds	
24	RDD	Read Data	in
25	GND	Signal grounds	
26	HS	Head Select	out

3.3.6 Speaker Interface

The speaker logic signal is not driven by an amplifier. This must be done externaly. Then connect the speaker between VCC and speaker output to have no quiescient current.

3.4 Interrupts

3.4.1 Interrupt Controllers

An 8259A compatible interrupt controller, within the device, provides seven prioritized interrupt levels. Of these, several are normally associated with the board's onboard device interfaces and controllers, and several are available on the AT expansion bus.

Interrupt:	Sources:	onboard used:
IRQ0	ROM-BIOS clock tick function, from timer 0	yes
IRQ1	Keyboard controller output buffer full	yes
IRQ2	Used for cascade 2. 8259	yes
IRQ3	COM2 + COM4	yes
IRQ4	COM1 + COM3	yes
IRQ5	Free for user (but reserved for LAN interface) Not free if remapped to IRQ11 or IRQ15 (PIRQ5)	(yes)
IRQ6	Floppy controller	yes
IRQ7	LPT1 parallel printer	yes
IRQ8	Alarm function of the RTC	yes
IRQ9	Not available with ELAN400 Remappable to IRQ14 line (PIRQ0)	no
IRQ10	Remappable on ELAN400 Remapped to IRQ12 line (PIRQ2)	no
IRQ11	Remappable on ELAN400 Remapped to IRQ5 line (PIRQ5) Remapped to IRQ14 line (PIRQ0)	no
IRQ12	PS/2 mouse If mouse not used, desolder R52 on smartModule Remapped to IRQ10 or IRQ12 line (PIRQ2)	(yes)
IRQ13	Math. coprocessor	no
IRQ14	Harddisk IDE Remapped to IRQ9 or IRQ11 line (PIRQ0)	(yes)
IRQ15	Remapped on ELAN400 Remapped to IRQ5 line (PIRQ5)	no

3.4.2 Interrupt Redirection

On this design, some IRQ's are free programmable interrupt lines. So-called PIRQ's (**P**rogrammable **I**nterrupt **R**e**Q**uest) may be routed to other IRQ-lines. The program E4IRQMAP.EXE /PIRQ /IRQ select the multiplexer of the ELAN400. The redirector must be loaded before the first time of using the interrupt !
Since BIOS Version 2.12, you can change the PIRQ number in the BIOS Setup !

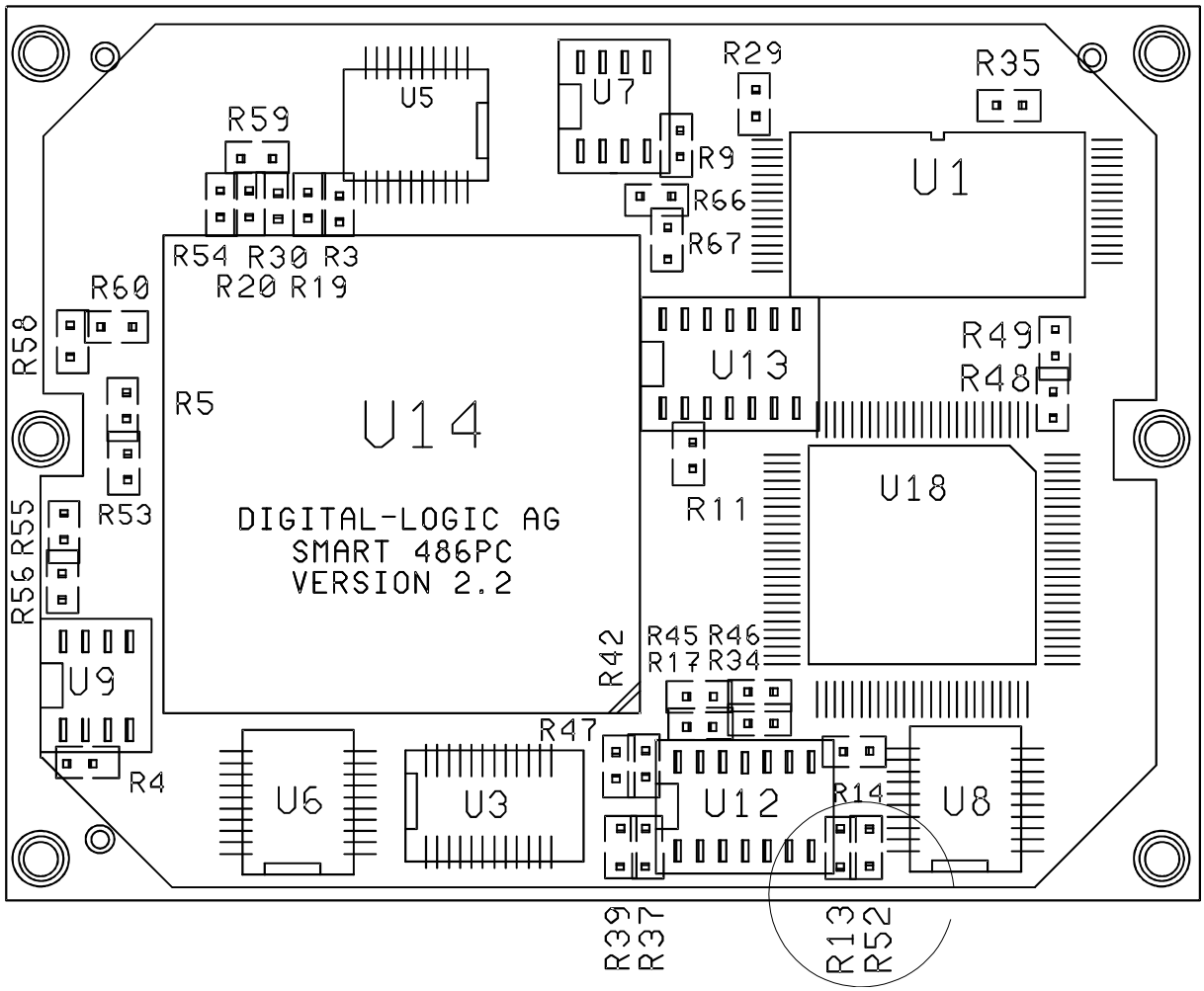
IRQ mapping possibilities

CPU PIRQ	IRQ line	IRQ-selectable by jumpers	IRQ-selectable by BIOS*
0	14	9 / 11	1, 3-15 or disable
2	12**	12 / 10	1, 3-15 or disable
3	3	-	1, 3-15 or disable
4	4	-	1, 3-15 or disable
5	5	11 / 15	1, 3-15 or disable
6	6	-	1, 3-15 or disable
7	7	-	1, 3-15 or disable

* since BIOS version 2.12

** If you want to change the IRQ12, disable the mouse port first by desoldering the R52 on the SMART-module. This is only possible by taking off the cover of the smartModule.

Therefore, this goes at your own risk and may can destroy the module !



3.5 Timers and Counters

3.5.1 Programmable Timers

An 8253 compatible timer/counter device is also included in the board's ASIC device. This device is utilized in precisely the same manner as in a standard AT implementation. Each channel of the 8253 is driven by a 1.190 MHz clock, derived from a 14.318 MHz oscillator, which can be internally divided in order to provide a variety of frequencies.

Timer 2 can also be used as a general purpose timer if the speaker function is not required.

Timer Assignment

Timer	Function
0	ROM-BIOS clock tick (18.2 Hz)
1	DRAM refresh request timing (15 µS)
2	Speaker tone generation time base

3.5.2 Battery Backed Clock (RTC)

An AT compatible date/time clock is located within the chipset. The device also contains a CMOS static RAM, compatible with that in standard ATs. System configuration data is normally stored in the clock chip's CMOS RAM in a manner consistent with the convention used in other AT compatible computers.

One unique feature of the board's battery-backed clock device is that it contains the backup battery directly on the board.

The battery is a DIGITAL-LOGIC AG replacement part. The battery-backed clock can be set by using the DIGITAL-LOGIC AG SETUP at boot-time.

Typical battery current at 25°C : <5 µA

3.5.3 Watchdog

See chapter 3.12 and 9.2.2

The watchdog works by using the special function in INT15h to strobe the watchdog periodically
This is working since boardversion V1.2 with a special hardware fix (April 2001).
See application note 092.

3.6 BIOS

3.6.1 ROM-BIOS Sockets

An EPROM socket with 8 Bit wide data access normally contains the board's AT compatible ROM-BIOS. The socket takes any of a 29F010 to 29F040 FLASH (or equivalent) device. The board's wait-state control logic automatically inserts four memory wait states in all CPU accesses to this socket. The ROM-BIOS sockets occupies the memory area from C0000H through FFFFFh; however, the board's ASIC logic reserves the entire area from C0000h through FFFFFh for onboard devices, so that this area is already usable for ROM-DOS and BIOS expansion modules. Consult the appropriate address map for the MICROSPACE MSM486SE/SEV ROM-BIOS sockets.

3.6.1.1 Standard BIOS ROM

DEVICE: 29F040 PLCC32 90ns

MAP:

Device MAP:	BIOS-Map:	Remarks:
00000-1FFFF	E0000 - FFFFFh	Chipset BIOS in the SM-486PC
00000-07FFF	C0000 - C7FFFh	VGA BIOS from Chips & Technology
08000-0FFFF	C8000 - CFFFFh	BIOS-Extensions, free for user

3.6.2 EEPROM Memory for Setup

The EEPROM is used for setup and configuration data, stored as an alternative to the CMOS-RTC. Optionally, the EEPROM setup driver may update the CMOS RTC, if the battery is running down and the checksum error would appear and stop the system.

1kByte (1024 bytes) are free for customer use. To access the EEPROM use the INT15 routines.
Please see chapter 8.2.1 EEPROM Function.

3.6.3 BIOS CMOS Setup

If wrong setups are memorized in the CMOS-RAM, the default values will be loaded after resetting the RTC/CMOS-RAM with the CMOS-RESET jumper. If the battery is down, it is always possible to start the system with the default values from the BIOS.

The following entries may be made:

Date:	The current Real Date of the RTC		
Time:	The current Real Time of the RTC		
Drive: A or B	none	=	no drive present, SSD / ROM-Disk enabled (if device is loaded)
	360k	=	5,25" low density drive, SSD enabled
	1,2 MB	=	5,25" high density drive
	720 K	=	3,5" low density drive
	1,44 MB	=	3,5" high density drive (Default for A:)
	The A: Drive is the bootable drive.		
Display type:	CRT:		for Mono CRTs, no LCD operating possible.
	40 x 25:		for Color CGA or LCD
	80 x 25:		for Color CGA or LCD (Default)
	VGA:		for VGA
Harddisk type:	defines which drive is connected Type = 0 means no drive is present (Default)! Drive type 48 and 49 enable you to define a custom harddisk parameter.		

WARNING:
On the next setup pages (switched with PgDn and PgUp) the values for special parameters are modifiable. Normally the parameters are set correctly by DIGITAL-LOGIC AG. Be very careful in modifying any parameter since the system could crash. Some parameters are dependent on the CPU type. The cache parameter is always available, for example. So, if you select too few wait states, the system will not start until you reset the CMOS-RAM using the RAM-Reset jumper, but the default values are reloaded. If you are not familiar with these parameters, do not change anything.

3.6.4 CMOS RAM Map

Systems based on the industry-standard specification include a battery backed Real Time Clock chip. This clock contains at least 64 bytes of non-volatile RAM. The system BIOS uses this area to store information including system configuration and initialization parameters, system diagnostics, and the time and date. This information remains intact even when the system is powered down.

SystemSoft's BIOS supports 128 bytes of CMOS RAM. This information is accessible through I/O ports 70h and 71h. CMOS RAM can be divided into several segments:

- Locations 00h - 0Fh contain real time clock (RTC) and status information
- Locations 10h - 2Fh contain system configuration data
- Locations 30h - 3Fh contain System BIOS-specific configuration data as well as chipset-specific information
- Locations 40h - 7Fh contain chipset-specific information as well as power management configuration parameters

The following table provides a summary of how these areas may be further divided.

Beginning	Ending	Checksum	Description
00h	0Fh	No	RTC and Checksum
10h	2Dh	Yes	System Configuration
2Eh	2Fh	No	Checksum Value of 10h - 2Dh
30h	33h	No	Standard CMOS
34h	3Fh	No	Standard CMOS - SystemSoft Reserved
40h	5Bh	Yes	Extended CMOS - Chipset Specific
5Ch	5Dh	No	Checksum Value of 40h - 5Bh
5Eh	6Eh	No	Extended CMOS - Chipset Specific
6Fh	7Dh	Yes	Extended CMOS - Power Management
7Eh	7Fh	No	Checksum Value of 6Fh - 7Dh

Location	Description
00h	Time of day (seconds) specified in BCD
01h	Alarm (seconds) specified in BCD
02h	Time of Day (minutes) specified in BCD
03h	Alarm (minutes) specified in BCD
04h	Time of Day (hours) specified in BCD
05h	Alarm (hours) specified in BCD
06h	Day of week specified in BCD
07h	Day of month specified in BCD
08h	Month specified in BCD
09h	Year specified in BCD
0Ah	Status Register A Bit 7 = Update in progress Bits 6-4 = Time based frequency divider Bits 3-0 = Rate selection bits that define the periodic interrupt rate and output frequency.
0Bh	Status Register B Bit 7 = Run/Halt 0 Run 1 Halt Bit 6 = Periodic Timer 0 Disable 1 Enable Bit 5 = Alarm Interrupt 0 Disable 1 Enable Bit 4 = Update Ended Interrupt 0 Disable 1 Enable Bit 3 = Square Wave Interrupt 0 Disable 1 Enable Bit 2 = Calendar Format 0 BCD 1 Binary Bit 1 = Time Format 0 12-Hour 1 24-Hour Bit 0 = Daylight Savings Time 0 Disable 1 Enable
0Ch	Status Register C Bit 7 = Interrupt Flag Bit 6 = Periodic Interrupt Flag Bit 5 = Alarm Interrupt Flag Bit 4 = Update Interrupt Flag Bits 3-0 = Reserved
0Dh	Status Register D Bit 7 = Real Time Clock 0 Lost Power 1 Power

Continued...

CMOS Map Continued...

Location	Description
0Eh	CMOS Location for Bad CMOS and Checksum Flags bit 7 = Flag for CMOS Lost Power 0 = Power OK 1 = Lost Power bit 6 = Flag for CMOS checksum bad 0 = Checksum is valid 1 = Checksum is bad
0Fh	Shutdown Code
10h	Diskette Drives bits 7-4 = Diskette Drive A 0000 = Not installed 0001 = Drive A = 360 K 0010 = Drive A = 1.2 MB 0011 = Drive A = 720 K 0100 = Drive A = 1.44 MB 0101 = Drive A = 2.88 MB bits 3-0 = Diskette Drive B 0000 = Not installed 0001 = Drive B = 360 K 0010 = Drive B = 1.2 MB 0011 = Drive B = 720 K 0100 = Drive B = 1.44 MB 0101 = Drive B = 2.88 MB
11h	Reserved
12h	Fixed (Hard) Drives bits 7-4 = Hard Drive 0, AT Type 0000 = Not installed 0001-1110 Types 1 - 14 1111 = Extended drive types 16-44. See location 19h. bits 3-0 = Hard Drive 1, AT Type 0000 = Not installed 0001-1110 Types 1 - 14 1111 = Extended drive types 16-44. See location 2Ah. See the *Fixed Drive Type Parameters Table* in Chapter 2 for information on drive types 16-44.
13h	Reserved

Continued...

CMOS Map Continued...

Location	Description
14h	Equipment bits 7-6 = Number of Diskette Drives 00 = One diskette drive 01 = Two diskette drives 10, 11 = Reserved bits 5-4 = Primary Display Type 00 = Adapter with option ROM 01 = CGA in 40 column mode 10 = CGA in 80 column mode 11 = Monochrome bits 3-2 = Reserved bit 1 = Math Coprocessor Presence 0 = Not installed 1 = Installed bit 0 = Bootable Diskette Drive 0 = Not installed 1 = Installed
15h	Base Memory Size (in KB) - Low Byte
16h	Base Memory Size (in KB) - High Byte
17h	Extended Memory Size in (KB) - Low Byte
18h	Extended Memory Size (in KB) - High Byte
19h	Extended Drive Type - Hard Drive 0 See the *Fixed Drive Type Parameters Table* in Chapter 2 for information on drive types 16-44.
1Ah	Extended Drive Type - Hard Drive 1 See the *Fixed Drive Type Parameters Table* in Chapter 2 for information on drive types 16-44.
1Bh	Custom and Fixed (Hard) Drive Flags bits 7-6 = Reserved bit 5 = Internal Floppy Diskette Controller 0 = Disabled 1 = Enabled bit 4 = Internal IDE Controller 0 = Disabled 1 = Enabled bit 3 = Hard Drive 0 Custom Flag 0 = Disable 1 = Enabled bit 2 = Hard Drive 0 IDE Flag 0 = Disable 1 = Enabled bit 1 = Hard Drive 1 Custom Flag 0 = Disable 1 = Enabled bit 0 = Hard Drive 1 IDE Flag 0 = Disable 1 = Enabled

Continued...

CMOS Map Continued...

Location	Description
1Ch	Reserved
1Dh	EMS Memory Size Low Byte
1Eh	EMS Memory Size High Byte
1Fh - 24h	Custom Drive Table 0 These 6 bytes (48 bits) contain the following data: Cylinders Landing Zone 10 bits Write Precomp 10 bits Heads Sectors/Track 8 bits
1Fh	Byte 0 bits 7-0 = Lower 8 Bits of Cylinders
20h	Byte 1 bits 7-2 = Lower 6 Bits of Landing Zone bits 1-0 = Upper 2 Bits of Cylinders
21h	Byte 2 bits 7-4 = Lower 4 Bits of Write Precompensation bits 3-0 = Upper 4 Bits of Landing Zone
22h	Byte 3 bits 7-6 = Reserved bits 5-0 = Upper 6 Bits of Write Precompensation
23h	Byte 4 bits 7-0 = Number of Heads
24h	Byte 5 bits 7-0 = Sectors Per Track
25h - 2Ah	Custom Drive Table 1 These 6 bytes (48 bits) contain the following data: Cylinders Landing Zone 10 bits Write Precomp 10 bits Heads Sectors/Track 8 bits
25h	Byte 0 bits 7-0 = Lower 8 Bits of Cylinders
26h	Byte 1 bits 7-2 = Lower 6 Bits of Landing Zone bits 1-0 = Upper 2 Bits of Cylinders
27h	Byte 2 bits 7-4 = Lower 4 Bits of Write Precompensation bits 3-0 = Upper 4 Bits of Landing Zone

Continued...

CMOS Map Continued...

Location	Description
28h	Byte 3 bits 7-6 = Reserved bits 5-0 = Upper 6 Bits of Write Precompensation
29h	Byte 4 bits 7-0 = Number of Heads
2Ah	Byte 5 bits 7-0 = Sectors Per Track
2Bh	Boot Password bit 7 = Enable/Disable Password 0 = Disable Password 1 = Enable Password bits 6-0 = Calculated Password
2Ch	SCU Password bit 7 = Enable/Disable Password 0 = Disable Password 1 = Enable Password bits 6-0 = Calculated Password
2Dh	Reserved
2Eh	High Byte of Checksum - Locations 10h to 2Dh
2Fh	Low Byte of Checksum - Locations 10h to 2Dh
30h	Extended RAM (KB) detected by POST - Low Byte
31h	Extended RAM (KB) detected by POST - High Byte
32h	BCD Value for Century
33h	Base Memory Installed bit 7 = Flag for Memory Size 0 = 640KB 1 = 512KB bits 6-0 = Reserved
34h	Minor CPU Revision Differentiates CPUs within a CPU type (i.e., 486SX vs 486 DX, vs 486 DX/2). This is crucial for correctly determining CPU input clock frequency. During a power on reset, Reg DL holds minor CPU revision.
35h	Major CPU Revision Differentiates between different CPUs (i.e., 386, 486, Pentium). This is crucial for correctly determining CPU input clock frequency. During a power on reset, Reg DH holds major CPU revision.
36h	Hotkey Usage bits 7-6 = Reserved bit 5 = Semaphore for Completed POST bit 4 = Semaphore for 0 Volt POST (not currently used) bit 3 = Semaphore for already in SCU menu bit 2 = Semaphore for already in PM menu bit 1 = Semaphore for SCU menu call pending bit 0 = Semaphore for PM menu call pending
40h-7Fh	Definitions for these locations vary depending on the chipset.

3.6.5 Harddisk PIO Modes

Block Mode Transfer: (Multi-Sector)

Block mode boots IDE drive performance by increasing the amount of data transferred.

No Block Mode: 512 Byte per interrupt
Block Mode: up to 64 kByte per interrupt

LBA Mode:

LBA (logical block addressing) is a new method of addressing data on a disk drive. In the standard ST506 (MFM) ISA hard disk, data is accessed via a cylinder - head - sector format.

LBA Mode disabled: max. 528 MByte per Disk

LBA Mode enabled: max. 8 Gbyte per Disk

The maximum parameters are:
1024 Cyl., 16 heads, 63 Sec/Track

32Bit Transfer:

Some operating system can handle two 16Bit word as one 32Bit access. This accelerates the IDE transfer.

3.6.6 EEPROM saved CMOS Setup

The EEPROM has different functions, as listed below:

- Backup of the CMOS-Setup values.
- Storing the keymatrix definitions, if the hardware supports a keymatrix (MSM486SE/SEV, MSM386SN, MSM486SV).
- Storing system informations like: version, production date, customisation of the board, CPU type.
- Storing user/application values.

The EEPROM will be updated automatically after exiting the BIOS setup menu. The system will operate also without any CMOS battery. While booting up, the CMOS is automatically updated with the EEPROM values.

Press the Esc-key while powering on the system until the video shows the BIOS message and the CMOS will **not** be updated.
This would be helpful, if wrong parameters are stored in the EEPROM and the setup of the BIOS does not start.

If the system hangs or a problem appears, the following steps must be performed:
1. Reset the CMOS-Setup (use the jumper to reset or disconnect the battery for at least 10 minutes).
2. Press Esc until the system starts up.
3. Enter the BIOS Setup:
   a) load DEFAULT values
   b) enter the settings for the environment
   c) exit the setup
4. Restart the system.

- The user may access the EEPROM through the INT15 special functions. Refer to the chapter SFI functions.
- The keymatrix is defined with special EDITMATR.EXE and SAVEMATR.EXE tools.
- The system information are read only information. To read, use the SFI functions.

3.7 Download the VGA-BIOS and the CORE-BIOS

Before downloading a BIOS, please check as follows:

- Select the SHADOW option in the BIOS, for a BIOS and VGA (if this option is available).
- Disable the EMM386 or other memory managers in the CONFIG.SYS of your bootdisk.
- Make sure, that the DOWN_xxx.EXE programm and the BIOS to download are on the same path and directory!
- Boot the DOS without config.sys & autoexec.bat -> press “F5” while starting DOS boot.
- Is the empty diskspace, where the down.exe is located, larger than 64kB (for safe storage)
- Is the floppydisk not write-protected

Start the DOWNLOADING Tool with:

- Start the corresponding download tool. Refer to the table to see which tool fits in, each productgroup has its own download tool. Do never use the wrong one!

Product:	BIOS-Core download	VGA-BIOS download	BIOS-Ext. download
File-Extension:	*.COR	*.V40 , *.V45 *.V48 depending on the product	*.BIN
BIOS Size:	128k	32k	32k
Addressrange:	E0000 - FFFFFh	C0000 – C7FFFh	C8000 - CFFFFh
MSM386SN	DOWN_3SN.EXE	-	-
MSM386SV	DOWN_3SV.EXE	DOWN_3SV.EXE	DOWN_3SV.EXE
MSM486SL	DOWN_4SN.EXE	-	-
MSM486SN	DOWN_4SN.EXE	-	-
MSM486SV	DOWN_4SV.EXE	DOWN_4SV.EXE	DOWN_4SV.EXE
MSM486SE / SEV	DOWN_4SE.EXE	DOWN_4SE.EXE	-
MSM486DN	DOWN_4DX.EXE	-	-
MSM486DX	DOWN_4DX.EXE	DOWN_4DX.EXE	DOWN_4DX.EXE
SM-486PC / EK	DOWN_SM4.EXE	On the –EK : DOWN_SM4.EXE	-
SM-486PCX / EK	DOWN_S4X.EXE	DOWN_S4X.EXE	DOWN_S4X.EXE
MSM5x86DX	DOWN_4DX.EXE	DOWN_4DX.EXE	DOWN_4DX.EXE
MSM586SEN / SEV	To be defined	To be defined	
MSM-P5	- AMI82602.EXE or - FLASHAMI.COM (AMIBOOT.ROM)**	DOWN_000.EXE	-
PCC-P5L / PCC-PII AMI- BIOS	AMI82602.EXE	DOWN_000.EXE	-
PCC-P5L / PCC-PII PCC-P5S / PCC-P3S PHOENIX- BIOS	PHLASH.EXE PLATFORM.BIN	DOWN_000.EXE	-
MSM-P5S MSM-P5SV / SEV AMI- BIOS	AMI82602.EXE	DOWN_000.EXE	-
MSM-P5SN / SEN AMI- BIOS	AMI82602.EXE	-	-
MSM-P5S MSM-P5SV / SEV PHOENIX- BIOS	PHLASH.EXE PLATFORM.BIN	DOWN_000.EXE	-
MSM-P5SN / SEN PHOENIX- BIOS	PHLASH.EXE PLATFORM.BIN	-	-
MSEBX	PHLASH.EXE PLATFORM.BIN	DOWN_000.EXE	
SMP5PC / 3PC / DK	PHLASH.EXE PLATFORM.BIN	DOWN_000.EXE	
MAS-P5 / P3	PHLASH.EXE PLATFORM.BIN	DOWN_000.EXE	

Remarks:
** Core- file has to be renamed as written in brackets

3.7.1 VGA- BIOS Download Function

The BIOS for the VGA must be downloaded, before a LCD is connected. This could be also a new LCD- display, which needs a corresponding VGA- BIOS.

How to download a VGA- BIOS:

1. Restart the system with the SHADOW enabled (if available) and no EMM386 loaded.
2. Check, if you find the DOWN_xxx.EXE and the *.V40 / *.000 files on your disk, to get downloaded.
3. Refer to the VGABIOS.DOC for more information about the VGABIOS files.
4. Insert the floppydisk with the program DOWN_xxx.EXE and all VGA-Drivers.
5. Start DOWN_xxx.EXE.
6. Check, if the DOWN program has identified the product and the shadow correctly.
7. Select the function PROGRAMM VGA- BIOS.
8. Select the VGA- BIOS out of the proposed file list (UP/DOWN arrows) and press ENTER.
9. Check, if the new VGA- header is displayed on the VGA- INFO- screen.
10. After proceeding, switch off the power and restart the board (cold start).

If the download does not work:
- Check, if no EMM386 is loaded.
- Check, if no peripheral card is in the system, which occupies the same memory range. Disconnect this card.
- If the download is stopped or not completed, make only a warm boot and repeat the steps or download another file. As the video is may shadowed, everything is visible and a cold boot would clear the screen and nothing would be visible afterwards.

If the screen flickers or is misaligned after reboot:
- The previously loaded VGA- BIOS is not corresponding 100% or works only on the LCD properly.

If the screen is dark after the reboot of the system:
- A new system BIOS must be programmed. Ask DIGITAL-LOGIC AG for the binary file.

If the previous version is still programmed:
- Switch off the board and do not make a warm boot due to the fact that the data may are still stored in the memory shadow.

What is the filename of the BIOS-Files:

Operation:	Filename:	Size:
Download COREBIOS	*.COR	128k
Read the COREBIOS	READ_SM4.COR	128k
DOWNLOAD VGA	*.548, *.V48	32k
Read the VGABIOS	READ_VGA.548	32k
DOWNLOAD BIOSEXT	*.BIN	32k
Read the BIOSEXT	READC8CF.BIN	32k

3.8 Memory

3.8.1 System Memory Map

The AMD ELAN400™ CPU used as central processing unit on the MICROSPACE PC has a memory address space which is defined by 26 address bits. Therefore, it can address 64 MByte of memory. The memory address MAP is as follows:

3.8.1.1 CPU AMD ELAN400 Bios 2.xx

Address:	Size:	Function / Comments:
0000000 - 009FFFFh	640 Kbytes	Onboard DRAM for DOS applications
00A0000 - 00BFFFFh	128 Kbytes	CGA, EGA, LCD Video RAM 128kB
00C0000 - 00CBFFFh 00CC000- 00CFFFFh	48 Kbytes 16 Kbytes	VGA BIOS free for user BIOS Extension
00D0000 - 00DFFFFh	64 Kbytes	No FFS: D0000 - DDFFF are free DE000 - DFFFF for Msystems With DL-FFS: D0000 - D9FFF are free DA000 - DFFFF for DL-FFS With PC-CARD: additional 4k are used on D0000
00E0000 - 00E8000h 00E8000 - 00EBFFFh 00EC000- 00EEFFFh	14 Kbytes 38 Kbytes 12 kBytes	Flashbios, Resource bios CARDBIOS
00F0000 - 00FFFFFh	64 Kbytes	BIOS
0100000 - 01FFFFFh	1 MByte	DRAM for extended onboard memory
0200000 - 0FFFFFFh	2-16 Mbytes	DRAM for extended onboard memory

Address:	Size:	Function / Comments:
1000000 - 1FFFFFF	16 Mbytes	Flashdisk Window
2000000 - 2FFFFFF	16 Mbytes	Reserved for PCCARD Socket 1
3000000 - 3FFFFFF	16 Mbytes	Reserved for PCCARD Socket 2*

3.8.2 System I/O map

The following table shows the detailed listing of the I/O port assignments used in the MICROSPACE board:

I/O Address	Read/Write Status	Description
0000h	R / W	DMA channel 0 address byte 0 (low), then byte 1
0001h	R / W	DMA channel 0 word count byte 0 (low), then byte 1
0002h	R / W	DMA channel 1 address byte 0 (low), then byte 1
0003h	R / W	DMA channel 1 word count byte 0 (low), then byte 1
0004h	R / W	DMA channel 2 address byte 0 (low), then byte 1
0005h	R / W	DMA channel 2 word count byte 0 (low), then byte 1
0006h	R / W	DMA channel 3 address byte 0 (low), then byte 1
0007h	R / W	DMA channel 3 word count byte 0 (low), then byte 1
0008h	R	DMA channel 0-3 status register bit 7 = 1 Channel 3 request bit 6 = 1 Channel 2 request bit 5 = 1 Channel 1 request bit 4 = 1 Channel 0 request bit 3 = 1 Terminal count on channel 3 bit 2 = 1 Terminal count on channel 2 bit 1 = 1 Terminal count on channel 1 bit 0 = 1 Terminal count on channel 0

Continued...

I/O Address	Read/Write Status	Description
0008h	W	DMA channel 0-3 command register bit 7 = DACK sense active high/low 0 low 1 high bit 6 = DREQ sense active high/low 0 low 1 high bit 5 = Write selection 0 Late write selection 1 Extended write selection bit 4 = Priority 0 Fixed 1 Rotating bit 3 = Timing 0 Normal 1 Rotating bit 2 = Controller enable/disable 0 Enable 1 Disable bit 1 = Memory-to-memory enable/disable 0 Disable 1 Enable bit 0 = Reserved
0009h	W	DMA write request register
000Ah	R / W	DMA channel 0-3 mask register bits 7-3 = Reserved bit 2 = 0 Clear bit 1 Set bit bits 1-0 = Channel Select 00 Channel 0 01 Channel 1 10 Channel 2 11 Channel 3
00Bh	W	DMA channel 0-3 mode register bits 7-6 = 00 Demand mode 01 Single mode 10 Block mode 11 Cascade mode bit 5 = 0 Address increment select 1 Address decrement select bit 4 = 0 Disable auto initialization 1 Enable auto initialization bits 3-2 = Operation type 00 Verify operation 01 Write to memory 10 Read from memory 11 Reserved bits 1-0 = Channel select 00 Channel 0 01 Channel 1 10 Channel 2 11 Channel 3

Continued...

I/O Address	Read/Write Status	Description
000Ch	W	DMA clear byte pointer flip/flop
000Dh	R	DMA read temporary register
000Dh	W	DMA master clear
000Eh	W	DMA clear mask register
000Fh	W	DMA write mask register
0020h	W	Programmable Interrupt Controller - Initialization Command Word 1 (ICW1) provided bit 4 = 1 bits 7-5 = 000 Used only in 8080 or 8085 mode bit 4 = 1 ICW1 is used bit 3 = 0 Edge triggered mode 1 Level triggered mode bit 2 = 0 Successive interrupt vectors separated by 8 bytes 1 Successive interrupt vectors separated by 4 bytes bit 1 = 0 Cascade mode 1 Single mode bit 0 = 0 ICW4 not needed 1 ICW4 needed
0021h	W	Used for ICW2, ICW3, or ICW4 in sequential order afterICW1 is written to port 0020h **ICW2** bits 7-3 = Address A0-A3 of base vector address for interrupt controller bits 2-0 = Reserved (should be 000) **ICW3** (for slave controller 00A1h) bits 7-3 = Reserved (should be 0000) bits 2-0 = 1 Slave ID **ICW4** bits 7-5 = Reserved (should be 000) bit 4 = 0 No special fully nested mode 1 Special fully nested mode bits 3-2 = Mode 00 Non buffered mode 01 Non buffered mode 10 Buffered mode/slave 11 Buffered mode/master bit 1 = 0 Normal EOI 1 Auto EOI bit 0 = 0 8085 mode 1 8080 / 8088 mode

Continued...

I/O Address	Read/Write Status	Description
0021h	R / W	PIC master interrupt mask register (OCW1) bit 7 = 0 Enable parallel printer interrupt bit 6 = 0 Enable diskette interrupt bit 5 = 0 Enable hard disk interrupt bit 4 = 0 Enable serial port 1 interrupt bit 3 = 0 Enable serial port 2 interrupt bit 2 = 0 Enable video interrupt bit 1 = 0 Enable kybd/pointing device/RTC interrupt bit 0 = 0 Enable interrupt timer
0021h	W	PIC OWC2 (if bits 4-3 = 0) bit 7 = Reserved bits 6-5 = 000 Rotate in automatic EOI mode (clear) 001 Nonspecific EOI 010 No operation 011 Specific EOI 100 Rotate in automatic EOI mode (set) 101 Rotate on nonspecific EOI command 110 Set priority command 111 Rotate on specific EOI command bits 4-3 = Reserved (should be 00) bits 2-0 = Interrupt request to which the command applies
0020h	R	PIC interrupt request and in-service registers programmed by OCW3 **Interrupt request register** bits 7-0 = 0 No active request for the corresponding interrupt line 1 Active request for the corresponding interrupt line **Interrupt in-service register** bits 7-0 = 0 Corresponding interrupt line not currently being serviced 1 Corresponding interrupt line is currently being serviced
0021h	W	PIC OCW3 (if bit 4 = 0, bit 3 = 1) bit 7 = Reserved (should 0) bits 6-5 = 00 No operation 01 No operation 10 Reset special mask 11 Set special mask bit 4 = Reserved (should be 0) bit = Reserved (should be 1) bit 2 = 0 No poll command 1 Poll command bits 1-0 = 00 No operation 01 Operation 10 Read interrupt request register on next read at port 0020 h 11 Read interrupt in-service register on next read at port 0020h

Continued...

I/O Address	Read/Write Status	Description
0022h	R / W	Chipsset Register Address
0023h	R / W	Chipsset Register Data
0040h	R / W	Programmable Interrupt Time read/write counter 0, keyboard controller channel 0
0041h	R / W	Programmer Interrupt Timer channel 1
0042h	R / W	Programmable Interrupt Timer miscellaneous register channel 2
0043h	W	Programmable Interrupt Timer mode port - control word register for counters 0 and 2 bits 7-0 = Counter select 00 Counter 0 select 01 Counter 1 select 10 Counter 2 select bits 5-4 = 00 Counter latch command 01 R / W counter, bits 0-7 only 10 R / W counter, bits 8-15 only 11 R / W counter, bits 0-7 first, then bits 8-15 bits 3-1 = Select mode 000 Mode 0 001 Mode 1 programmable one shot x10 Mode 2 rate generator x11 Mode 3 square wave generator 100 Mode 4 software-triggered strobe 101 Mode 5 hardware-triggered strobe bit 0 = 0 Binary counter is 16 bits 1 Binary counter decimal (BCD) counter
0048h	R / W	Programmable interrupt timer
0060h	R	Keyboard controller data port or keyboard input buffer
0060h	W	Keyboard or keyboard controller data output buffer

Continued...

I/O Address	Read/Write Status	Description
0064h	R	Keyboard controller read status bit 7 = 0 No parity error 1 Parity error on keyboard transmission bit 6 = 0 No timeout 1 Received timeout bit 5 = 0 No timeout 1 Keyboard transmission timeout bit 4 = 0 Keyboard inhibited 1 Keyboard not inhibited bit 3 = 0 Data 1 Command bit 2 = System flag status bit 1 = 0 Input buffer empty 1 Input buffer full bit 0 = 0 Output buffer empty 1 Output buffer full
0064h	W	Keyboard controller input buffer
0070h	R	CMOS RAM index register port and NMI mask bit 7 = 1 NMI disabled bits 6-0 = 0 CMOS RAM index
0071h	R / W	CMOS RAM data register port
0080h	R / W	Temporary storage for additional page register
0080h	R	Manufacturing diagnostic port (this port can access POST checkpoints)
0081h	R / W	DMA channel 2 address byte 2
0082h	R / W	DMA channel 2 address byte 2
0083h	R / W	DMA channel 1 address byte 2
0084h	R / W	Extra DMA page register
0085h	R / W	Extra DMA page register
0086h	R / W	Extra DMA page register
0087h	R / W	DMA channel 0 address byte 2
0088h	R / W	Extra DMA page register
0089h	R / W	DMA channel 6 address byte 2
008Ah	R / W	DMA channel 7 address byte 2
008Bh	R / W	DMA channel 5 address byte 2
008Ch	R / W	Extra DMA page register
008Dh	R / W	Extra DMA page register
008Eh	R / W	Extra DMA page register
008Fh	R / W	DMA refresh page register

Continued...

I/O Address	Read/Write Status	Description
00A0h - 00A1h are reserved for the slave programmable interrupt controller. The bit definitions are identical to those of addresses 0020h - 0021h except where indicated.		
00A0h	R / W	Programmable interrupt controller 2
00A1h	R / W	Programmable interrupt controller 2 mask bit 7 = 0 Reserved bit 6 = 0 Enable hard disk interrupt bit 5 = 0 Enable coprocessor execution interrupt bit 4 = 0 Enable mouse interrupt bits 3-2 = 0 Reserved bit 1 = 0 Enable redirect cascade bit 0 = 0 Enable real time clock interrupt
00C0h	R / W	DMA channel 4 memory address bytes 1 and 0 (low)
00C2h	R / W	DMA channel 4 transfer count bytes 1 and 0 (low)
00C4h	R / W	DMA channel 5 memory address bytes 1 and 0 (low)
00C6h	R / W	DMA channel 5 transfer count bytes 1 and 0 (low)
00C8h	R / W	DMA channel 6 memory address bytes 1 and 0 (low)
00CAh	R / W	DMA channel 6 transfer count bytes 1 and 0 (low)
00CCh	R / W	DMA channel 7 memory address bytes 1 and 0 (low)
00CEh	R / W	DMA channel 7 transfer count bytes 1 and 0 (low)
00D0h	R	Status register for DMA channels 4-7 bit 7 = 1 Channel 7 request bit 6 = 1 Channel 6 request bit 5 = 1 Channel 5 request bit 4 = 1 Channel 4 request bit 3 = 1 Terminal count on channel 7 bit 2 = 1 Terminal count on channel 6 bit 1 = 1 Terminal count on channel 5 bit 0 = 1 Terminal count on channel 4
00D0h	W	Command register for DMA channels 4-7 bit 7 = 0 DACK sense active low 1 DACK sense active high bit 6 = 0 DREQ sense active low 1 DREQ sense active high bit 5 = 0 Late write selection 1 Extended write selection bit 4 = 0 Fixed Priority 1 Rotating Priority bit 3 = 0 Normal Timing 1 Rotating Timing bit 2 = 0 Enable controller 1 Disable controller bit 1 = 0 Disable memory-to-memory transfer 1 Enable memory-to-memory transfer bit 0 = Reserved

Continued...

I/O Address	Read/Write Status	Description
00D2h	W	Write request register for DMA channels 4-7
00D4h	W	Write single mask register bit for DMA channels 4-7 bits 7-3 = 0 Reserved bit 2 = 0 Clear mask bit, 1 Set mask bit bits 1-0 = Channel select 00 Channel 4 01 Channel 5 10 Channel 6 11 Channel 7
00D6h	W	Mode register for DMA channels 4-7 bits 7-6 = 00 Demand mode 01 Single mode 10 Block mode 11 Cascade mode bit 5 = 0 Address increment select 1 Address decrement select bit 4 = 0 Disable auto initialization 1 Enable auto initialization bits 3-2 = Operation type 00 Verify operation 01 Write to memory 10 Read from memory 11 Reserved bits 1-0 = Channel select 00 Channel 4 01 Channel 5 10 Channel 6 11 Channel 7
00D8h	W	Clear byte pointer flip/flop for DMA channels 4-7
00DAh	R	Read Temporary Register for DMA channels 4-7
00DAh	W	Master Clear for DMA channels 4-7
00DCh	W	Clear mask register for DMA channels 4-7
00DEh	W	Write mask register for DMA channels 4-7
00F0h	W	Math coprocessor clear busy latch
00F1h	W	Math coprocessor reset
00F2h - 00FFh	R / W	Math coprocessor
I/O addresses 0170h - 0177h are reserved for use with a secondary hard drive. See addresses 01F0h - 01F7h for bit definitions.		
0170h	R / W	Data register for hard drive 1
0171h	R	Error register for hard drive 1
0171h	W	Precomposition register for hard drive 1
0172h	R / W	Sector count - hard drive 1

Continued...

I/O Address	Read/Write Status	Description
0173h	R / W	Sector number for hard disk 1
0174h	R / W	Number of cylinders (low byte) for hard drive 1
0175h	R / W	Number of cylinders (high byte) for hard drive 1
0716h	R / W	Drive/head register for hard drive 1
0177h	R	Status register for hard drive 1
0177h	W	Command register for hard drive 1
01F0h	R / W	Data register base port for hard drive 0
01F1h	R	Error register for hard drive 0 **Diagnostic mode** bits 7-3 = Reserved bits 2-0 = Errors 0001 No errors 0010 Controller error 0011 Sector buffer error 0100 ECC device error 0101 Control processor error **Operation mode** bit 7 = Block 0 Bad block 1 Block not bad bit 6 = Error 0 No error 1 Uncorrectable ECC error bit 5 = Reserved bit 4 = ID 0 ID located 1 ID not located bit 3 = Reserved bit 2 = Command 0 Completed 1 Not completed bit 1 = Track 000 0 Not found 1 Found bit 0 = DRAM 0 Not found 1 Found (CP-3022 always 0)
01F1h	W	Write precomposition register for hard drive 0
01F2h	R / W	Sector count for hard disk 0
01F3h	R / W	Sector number for hard drive 0
01F4h	R / W	Number of cylinders (low byte) for hard drive 0
01F5h	R / W	Number of cylinders (high byte) for hard drive 0

Continued...

I/O Address	Read/Write Status	Description
01F6h	R / W	Drive/Head register for hard drive 0 bit 7 = 1 bit 6 = 0 bit 5 = 1 bit 4 = Drive select 0 First hard drive 1 Second hard drive bits 3-0 = Head select bits
01F7h	R	Status register for hard drive 0 bit 7 = 1 Controller is executing a command bit 6 = 1 Drive is ready bit 5 = 1 Write fault bit 4 = 1 Seek operation complete bit 3 = 1 Sector buffer requires servicing bit 2 = 1 Disk data read completed successfully bit 1 = Index (is set to 1 at each disk revolution) bit 0 = 1 Previous command ended with error
01F7h	W	Command register for hard drive 0
0200h - 020Fh	R / W	Game controller ports
0201h	R / W	I/O data - game port
0220h – 022Fh	R / W	Soundport AD1816 reserved
I/O addresses 0278h - 027Ah are reserved for use with parallel port 2. See the bit definitions for addresses 0378h - 037Ah.		
0278h	R / W	Data port for parallel port 2
0279h	R	Status port for parallel port 2
0279h	W	PnP Address register (only for PnP devices)
027Ah	R / W	Control port for parallel port 2
02B0h	R / W	Digital I/O reserved
I/O addresses 02E8h - 02EFh are reserved for use with serial port 4. See the bit definitions for I/O addresses 03F8h - 03FFh.		
02E8h	W	Transmitter holding register for serial port 4
02E8h	R	Receive buffer register for serial port 4
02E8h	R / W	Baud rate divisor (low byte) when DLAB = 1
02E9h	R / W	Baud rate divisor (high byte) when DLAB = 1
02E9h	R / W	Interrupt enable register when DLAB = 0
02EAh	R	Interrupt identification register for serial port 4
02EBh	R / W	Line control register for serial port 4
02ECh	R / W	Modem control register for serial port 4
02EDh	R	Line status register for serial port 4
02EEh	R	Modem status register for serial port 4
02EFh	R / W	Scratch register for serial port 4 (used for diagnostics)

Continued...

I/O Address	Read/Write Status	Description
I/O addresses 02F8h - 02FFh are reserved for use with serial port 2. See the bit definitions for I/O addresses 03F8h - 03FFh.		
02F8h	W	Transmitter holding register for serial port 2
02F8h	R	Receive buffer register for serial port 2
02F8h	R / W	Baud rate divisor (low byte) when DLAB = 1
02F9h	R / W	Baud rate divisor (high byte) when DLAB = 1
02F9h	R / W	Interrupt enable register when DLAB = 0
02FAh	R	Interrupt identification register for serial port 2
02FBh	R / W	Line control register for serial port 2
02FCh	R / W	Modem control register for serial port 2
02FDh	R	Line status register for serial port 2
02FEh	R	Modem status register for serial port 2
02FFh	R / W	Scratch register for serial port 2 (used for diagnostics)
0300h – 031Fh	R / W	LAN controller reserved
I/O addresses 0372h - 0377h are reserved for use with a secondary diskette controller. See the bit definitions for 03F2h - 03F7h.		
0372h	W	Digital output register for secondary diskette drive controller
0374h	R	Status register for secondary diskette drive controller
0375h	R / W	Data register for secondary diskette drive controller
0376h	R / W	Control register for secondary diskette drive controller
0377h	R	Digital input register for secondary diskette drive controller
0377h	W	Select register for secondary diskette data transfer rate
0378h	R / W	Data port for parallel port 1
0379h	R	Status port for parallel port 1 bit 7 = 0 Busy bit 6 = 0 Acknowledge bit 5 = 1 Out of paper bit 4 = 1 Printer is selected bit 3 = 0 Error bit 2 = 0 IRQ has occurred bit 1-0 = Reserved

Continued...

I/O Address	Read/Write Status	Description
037Ah	R / W	Control port for parallel port 1 bits 7-5 = Reserved bit 4 = 1 Enable IRQ bit 3 = 1 Select printer bit 2 = 0 Initialize printer bit 1 = 1 Automatic line feed bit 0 = 1 Strobe
03B0h - 03B8h	R / W	Various video registers
I/O addresses 03BCh - 03BEh are reserved for use with parallel port 3. See the bit definitions for addresses 0378h - 037Ah.		
03BCh	R / W	Data port - parallel port 3
03BDh	R / W	Status port - parallel port 3
03BEh	R / W	Control port - parallel port 3
03C0h - 03CFh	R / W	Video subsystem (EGA/VGA)
03C2h - 03D9h	R / W	Various CGA and CRTC registers
03E0h	R / W	PCCARD Address select
03E1h	R / W	PCCARD Data transfer with 365SL controller
I/O addresses 03E8h - 03EFh are reserved for use with serial port 3. See the bit definitions for I/O addresses 03F8h - 03FFh.		
03E8h	W	Transmitter holding register for serial port 3
03E8h	R	Receive buffer register for serial port 3
03E8h	R / W	Baud rate divisor (low byte) when DLAB = 1
03E9h	R / W	Baud rate divisor (high byte) when DLAB = 1
03E9h	R / W	Interrupt enable register when DLAB = 0
03EAh	R	Interrupt identification register for serial port 3
03EBh	R / W	Line control register for serial port 3
03ECh	R / W	Modem control register for serial port 3
03EDh	R	Line status register for serial port 3
03EEh	R	Modem status register for serial port 3
03EFh	R / W	Scratch register for serial port 3 (used for diagnostics)
03F2h	W	Digital output register for primary diskette drive controller bits 7-6 = 0 Reserved bit 5 = 1 Enable drive 1 motor bit 4 = 1 Enable drive 0 motor bit 3 = 1 Enable diskette DMA bit 2 = 0 Reset controller bit 1 = 0 Reserved bit 0 = 0 Select drive 0 1 Select drive 1

Continued...

I/O Address	Read/Write Status	Description
03F4h	R	Status register for primary diskette drive controller bit 7 = 1 Data register is ready bit 6 = 0 Transfer from system to controller 1 Transfer from controller to system bit 5 = 1 Non-DMA mode bit 4 = 1 Diskette drive controller is busy bits 3-2 = Reserved bit 1 = 1 Drive 1 is busy bit 0 = 1 Drive 0 is busy
03F5h	R / W	Data register for primary diskette drive controller
03F6h	R	Control port for primary diskette drive controller bits 7-4 = Reserved bit 3 = 0 Reduce write current 1 Head select enable bit 2 = 0 Disable diskette drive reset 1 Enable diskette drive reset bit 1 = 0 Disable diskette drive initialization 1 Enable diskette drive initialization bit 0 = Reserved
03F7h	R	Digital input register for primary diskette drive controller bit 7 = 1 Diskette drive line change bit 6 = 1 Write gate bit 5 = Head select 3 / reduced write current bit 4 = Head select 2 bit 3 = Head select 1 bit 2 = Head select 0 bit 1 = Drive 1 select bit 0 = Drive 0 select
03F7h	W	Select register for primary diskette data transfer rate bits 7-2 = Reserved bits 1-0 = 00 500 Kbs mode 01 300 Kbs mode 10 250 Kbs mode 11 Reserved
I/O addresses 03F8h - 03FFh are reserved for use with serial port 1. The bit definitions for these addresses also apply to serial ports 2, 3, and 4.		
03F8h	W	Transmitter holding register for serial port 1 - Contains the character to be sent. Bit 0, the least significant bit, is the first bit sent. bits 7-0 = Data bits 0-7 when the Divisor Latch Access Bit (DLAB) is 0
03F8h	R	Receive buffer register for serial port 1 - Contains the character to be received. Bit 0, the least significant bit, is the first bit received. bits 7-0 = Data bits 0-7 when the Divisor Latch Access Bit (DLAB) is 0

Continued...

I/O Address	Read/Write Status	Description
03F8h	R / W	Baud rate divisor (low byte) - This byte along with the high byte (03F9h) store the data transmission rate divisor. bits 7-0 = Data bits 0-7 when the Divisor Latch Access Bit (DLAB) is 1
03F9h	R / W	Baud rate divisor (high byte) - This byte along with the low byte (03F8h) store the data transmission rate divisor. bits 7-0 = Bits 8-15 when DLAB = 1
03F9h	R / W	Interrupt enable register bits 7-4 = Reserved bit 3 = 1 Modem status interrupt enable bit 2 = 1 Receiver line status interrupt enable bit 1 = 1 Transmitter holding register empty interrupt enable bit 0 = 1 Received data available interrupt enable when DLAB = 0
03FAh	R	Interrupt identification register - serial port 1 bits 7-3 = Reserved bits 2-1 = Identify interrupt with highest priority 00 Modem status interrupt (4th priority) 01 Transmitter holding register empty (3rd priority) 10 Received data available (2nd priority) 11 Receiver line status interrupt (1st priority) bit 0 = 0 Interrupt pending (register contents can be used as a pointer to interrupt service routine) 1 No interrupt pending
03FBh	R / W	Line control register - serial port 1 bit 7 = Divisor Latch Access (DLAB) 0 Access receiver buffer, transmitter holding register, and interrupt enable register 1 Access divisor latch bit 6 = 1 Set break enable. Forces serial output to spacing state and remains there bit 5 = Stick parity bit 4 = Even parity select bit 3 = Parity enable bit 2 = Number of stop bits bit 1 = Word length 00 5-bit word length 01 6-bit word length 10 7-bit word length 11 8-bit word length
03FCh	R / W	Modem control register - serial port 1 bits 7-5 = Reserved bit 4 = 1 Loopback mode for diagnostic testing of serial port. bit 3 = 1 User-defined output 2 bit 2 = 1 User-defined output 1 bit 1 = Force Request To Send active bit 0 = Force Data Terminal Ready active

Continued...

I/O Address	Read/Write Status	Description
03FDh	R	Line status register - serial port 1 bit 7 = Reserved bit 6 = 1 Transmitting shift and holding registers empty bit 5 = 1 Transmitter shift register empty bit 4 = 1 Break interrupt bit 3 = 1 Framing error bit 2 = 1 Overrun error bit 0 = 1 Data ready
03FEh	R	Modem status register - serial port 1 bit 7 = 1 Data Carrier Detect bit 6 = 1 Ring Indicator bit 5 = 1 Data Set Ready bit 4 = 1 Clear To Send bit 3 = 1 Delta Data Carrier bit 2 = 1 Trailing Edge Ring Indicator bit 1 = 1 Delta Data Set Ready bit 0 = 1 Delta Clear To Send
03FFh	R / W	Scratch register - serial port 1 (used for diagnostics)
0A79h	W	PnP Data write register (only for PnP devices)

3.8.3 BIOS-Variable-Segment

The BIOS Data Area is an area within system RAM that contains information about the system environment. System environment information includes definitions associated with hard disks, diskette drives, keyboard, video, as well as other BIOS functions. This area is created when the system is first powered on. It occupies a 256-byte area from 0400h - 04FFh. The following table lists the contents of the BIOS data area locations in offset order starting from segment address 40:00h.

BIOS Data Area Definitions

Location	Description
00h - 07h	I/O addresses for up to 4 serial ports
08h - 0Dh	I/O addresses for up to 3 parallel ports
0Eh - 0Fh	Segment address of extended data address
10h - 11h	Equipment list bits 15-14 = Number of parallel printer adapters 00 = Not installed 01 = One 10 = Two 11 = Three bits 13-12 = Reserved bits 11-9 = Number of serial adapters 00 = Not installed 001 = One 010 = Two 011 = Three 100 = Four bit 8 = Reserved bits 7-6 = Number of diskette drives 00 = One drive 01 = Two drives bits 5-4 = Initial video mode 00 = EGA or PGA 01 = 40 x 25 color 10 = 80 x 25 color 11 = 80 x 25 monochrome bit 3 = Reserved bit 2 = (1) Pointing device present bit 1 = (1) Math coprocessor present bit 0 = (1) Diskette drive present
12h	Reserved for port testing by manufacturer bits 7-1 = Reserved bit 0 = (0) Non-test mode (1) Test mode
13h	Memory size in kilobytes - low byte
14h	Memory size in kilobytes - high byte

Continued...

BIOS Data Area Definitions Continued...

Location	Description
15h - 16h	Reserved
17h	Keyboard Shift Qualifier States bit 7 = Insert mode bit 6 = CAPS lock bit 5 = Numlock bit 4 = Scroll Lock bit 3 = Either Alt key bit 2 = Either control key bit 1 = Left Shift key bit 0 = Right shift key 0 = not set / 1 = set
18h	Keyboard Toggle Key States bit 7 = (1) Insert held down bit 6 = (1) CAPS lock held down bit 5 = (1) Num Lock held down bit 4 = (1) Scroll Lock held down bit 3 = (1) Control+Num Lock held down bit 2 = (1) Sys Re held down bit 1 = (1) Left Alt held down bit 0 = (1) Left Control held down
19h	Scratch area for input from Alt key and numeric keypad
1Ah - 1Bh	Pointer to next character in keyboard buffer
1Ch - 1Dh	Pointer to last character in keyboard buffer
1Eh - 3Dh	Keyboard Buffer. Consists of 16 word entries.
3Eh	Diskette Drive Recalibration Flag bit 7 = (1) Diskette hardware interrupt occurred bits 6-4 = Not used bits 3-2 = Reserved bit 1 = (0) Recalibrate drive B bit 0 = (0) Recalibrate drive A

Continued...

BIOS Data Area Definitions Continued...

Location	Description
3Fh	Diskette Drive Motor Status bit 7 = Current operation 0 = Write or Format 1 = Read or Verify bit 6 = Reserved bits 5-4 = Drive Select 00 = Drive A 01 = Drive B bits 3-2 = Reserved 0 = Disable 1 = Enabled bit 1 = Drive B Motor Status 0 = Off 1 = On bit 1 = Drive A Motor Status 0 = Off 1 = On
40h	Diskette Drive Motor Timeout Disk drive motor is powered off when the value via the INT 08h timer interrupt reaches 0.
41h	Diskette Drive Status bit 7 = Drive Ready 0 = Ready 1 = Not ready bit 6 = Seek Error 0 = No error 1= Error occurred bit 5 = Controller operation 0 = Working 1 = Failed bits 4-0 = Error Codes 00h = No error 01h = Invalid function requested 02h = Address mark not located 03h = Write protect error 04h = Sector not found 06h = Diskette change line active (door opened) 08h = DMA overrun error 09h = Data boundary error 0Ch = Unknown media type 10h = ECC or CRC error 20h = Controller failure 40h = Seek operation failure 80h = Timeout
42h - 48h	Diskette Controller Status Bytes
49h	Video Mode Setting
4Ah - 4Bh	Number of Columns on screen
4Ch - 4Dh	Size of Current Page, in bytes
4Eh - 4Fh	Address of Current Page

Continued...

BIOS Data Area Definitions Continued...

Location	Description
50h - 5Fh	Position of cursor for each video page. Current cursor position is stored two bytes per page. First byte specifies the column, the second byte specifies the row.
60h - 61h	Start and end lines for 6845-compatible cursor type. 60h = starting scan line, 61h = ending scan line.
62h	Current Video Display Page
63h - 64h	6845-compatible I/O port address for current mode 3B4h = Monochrome 3D4h = Color
65h	Register for current mode select
66h	Current palette setting
67 - 6Ah	Address of adapter ROM
6Bh	Last interrupt the occurred
6Ch - 6Dh	Low word of timer count
6Eh - 6Fh	High word of timer count
70h	Timer count for 24-hour rollover flag
71h	Break key flag
72h - 73h	Reset flag 1243h = Soft reset. Memory test is bypassed.
74h	Status of last hard disk operation 00h = No error 01h = Invalid function requested 02h = Address mark not located 03h = Write protect error 04h = Sector not found 05h = Reset failed 08h = DMA overrun error 09h = Data boundary error 0Ah = Bad sector flag selected 0Bh = Bad track detected 0Dh = Invalid number of sectors on format 0Eh = Control data address mark detected 0Fh = DMA arbitration level out of range 10h = ECC or CRC error 11h = Data error corrected by ECC 20h = Controller failure 40h = Seek operation failure 80h = Timeout AAh = Drive not ready BBh = Undefined error occurred CCh = Write fault on selected drive E0h = Status error or error register = 0 FFh = Sense operation failed
75h	Number of hard drives
76h - 77h	Work area for hard disk

Continued...

BIOS Data Area Definitions Continued...

Location	Description
78h - 7Bh	Default parallel port timeout values
7Dh - 7Fh	Default serial port timeout values
80h - 81h	Pointer to start of keyboard buffer
82h - 83h	Pointer to end of keyboard buffer
84h - 88h	Reserved for EGA/VGA BIOS
8Ah	Reserved
8Bh	Diskette drive data transfer rate information bits 7-5 = Data rate on last operation 00 = 500 KBS 01 = 300 KBS 10 = 250 KBS bits 5-4 = Last drive step rate selected bits 3-2 = Data transfer rate at start of operation 00 = 500 KBS 01 = 300 KBS 10 = 250 KBS bits 1-0 = Reserved
8Ch	Copy of hard status register
8Dh	Copy of hard drive error register
8Eh	Hard drive interrupt flag
8Fh	Diskette controller information bit 7 = Reserved bit 6 = (1) Drive confirmed for drive B bit 5 = (1) Drive B is multi-rate bit 4 = (1) Drive B supports line change bit 3 = Reserved bit 2 = (1) Drive determined for drive A bit 1 = (1) Drive B is multi-rate bit 0 = (1) Drive B supports line change
90h - 91h	Media type for drives bits 7-6 = Data transfer rate 00 = 500 KBS 01 = 300 KBS 10 = 250 KBS bit 5 = (1) Double stepping required when 360K diskette inserted into 1.2MB drive bit 4 = (1) Known media is in drive bit 3 = Reserved bits 2-0 = Definitions upon return to user applications 000 = Testing 360K in 360K drive 001 = Testing 360K in 1.2 MB drive 010 = Testing 1.2 MB in 1.2 MB drive 011 = Confirmed 360K in 360K drive 100 = Confirmed 360K in 1.2 MB 101 = Confirmed 1.2 MB in 1.2 MB drive 111 = 720K in 720K drive or 1.44 MB in 1.44 MB drive

Continued...

BIOS Data Area Definitions Continued...

Location	Description
92h - 93h	Scratch area for diskette media. Low byte for drive A, high byte for drive B.
94h - 95h	Current track number for both drives. Low byte for drive A, high byte for drive B.
96h	Keyboard Status bit 7 = (1) Read ID bit 6 = (1) Last code was first ID bit 5 = (1) Force to Num Lock after read ID bit 4 = (1) Enhanced keyboard installed bit 3 = (1) Right ALT key active bit 2 = (1) Right Control key active bit 1 = (1) Last code was E0h bit 0 = (1) Last code was E1h
97h	Keyboard Status bit 7 = (1) Keyboard error bit 6 = (1) Updating LEDs bit 5 = (1) Resend code received bit 4 = (1) Acknowledge received bit 3 = Reserved bit 2 = (1) Caps lock LED state bit 1 = (1) Num lock LED state bit 0 = (1) Scroll lock LED state
98h - 99h	Offset address of user wait flag
9Ah - 9Bh	Segment address of user wait flag
9Ch - 9Dh	Wait count, in microseconds (low word)
9Eh - 9Fh	Wait count, in microseconds (high word)
A0h	Wait active flag bit 7 = (1) Time has elapsed bits 6-1 = Reserved bit 0 = (1) INT 15h, AH = 86h occurred
A1h - A7h	Reserved
A8h - ABh	Pointer to video parameters and overrides
ACh - FFh	Reserved
100h	Print screen status byte

3.8.3.1 Compatibility Service Table

In order to ensure compatibility with industry-standard memory locations for interrupt service routines and miscellaneous tabular data, the BIOS maintains tables and jump vectors.

Location	Description
FE05Bh	Entry Point for POST
FE2C3h	Entry point for INT 02h (NMI service routine)
FE3FEh	Entry point for INT 13h (Diskette Drive Services)
FE401h	Hard Drive Parameters Table
FE6F1h	Entry point for INT 19h (Bootstrap Loader routine)
FE6F5h	System Configuration Table
FE739h	Entry point for INT 14h (Serial Communications)
FE82Eh	Entry point for INT 16h (Keyboard Services)
FE897h	Entry point for INT 09h (Keyboard Services)
FEC59h	Entry point for INT 13h (Diskette Drive Services)
FEF57h	Entry point for INT OEh (Diskette Hardware Interrupt)
FEFC7h	Diskette Drive Parameters Table
FEFD2h	Entry point for INT 17h (Parallel Printer Services)
FF065h	Entry point for INT 10h (CGA Video Services)
FF0A4h	Video Parameter Table (6845 Data Table - CGA)
FF841h	Entry point for INT 12h (Memory Size Service)
FF84Dh	Entry point for INT 11h (Equipment List Service)
FF859h	Entry point for INT 15h (System Services)
Location	**Description**
FFA6Eh	Video graphics and text mode tables
FFE6Eh	Entry point for INT 1Ah (Time-of-Day Service)
FFEA5h	Entry Point for INT 08h (System Timer Service)
FFEF3h	Vector offset table loaded by POST
FFF53h	Dummy Interrupt routine IRET Instruction
FFF54h	Entry point for INT 05h (Print Screen Service)
FFFF0h	Entry point for Power-on
FFFF5h	BIOS Build Date (in ASCII)
FFFFEh	BIOS ID

3.8.3.2 System Configuration Parameter Table

The System Configuration Parameter Table located at F000:E6F5h contains basic configuration information about the computer. This table can be copied to system RAM by using INT 15h function C0h "Return System Configuration Parameters".

Location	Description
00h - 01h	Table Length (from next entry)
02h	System Model 0FCH = AT, Model 50, Model 60 0F8H = Model 80, Model 70 0FAH = Model 30, Model 25
03h	System Sub-Model Byte 00h = Model 80 or AT 01h = Model 80 or AT FFh = Unknown
04h	BIOS Version
05h	System Facilities bit 7 = DMA Channel 3 used by BIOS bit 6 = Slave int. PIC. available bit 5 = Real time clock available bit 4 = Keyboard scan code hook 1AH bit 3 = Wait for external event supported bit 2 = Extended data area available if set bit 1 = MicroChannel bus installed if set bit 0 = Unused
06h	Reserved (should be zeros)
07h	Reserved (should be zeros)
08h	Reserved (should be zeros)
09h	Reserved (should be zeros)

3.9 VGA, LCD

3.9.1 VGA / LCD Controller 65548

The 65548 High Performance Flatpanel/CRT VGA controller

- High integrated design (flatpanel/CRT VGA controller, RAMDAC, clock synthesizer)
- Local Bus (32 Bit CPU)
- Flexible display memory configurations
  - One 256Kx16 DRAM (512 KB)
- Advanced frame buffer architecture uses available display memory, maximizing integration and minimizing chip count
- Integrated programmable linear address feature accelerates GUI performance
- Hardware windows acceleration (65548)
  - 32-Bit graphics engine - System-to-screen and screen-to-screen BitBlt - Color expansion - Optimized for Windows™ BitBlt format
  - Hardware line drawing
  - 64x64x2 hardware cursor
- Hardware pop-up icon (65548)
  - 64x64 pixels by 4 colors
  - 128x128 pixels by 2 colors
- High performance resulting from zero wait-state writes (write buffer) and minimum wait-state reads (internal asynchronous FIFO design)
- Mixed 3.3 V / 5.0 V +/- 10 % Operation
- Interface to CHIPS' PC Video to display "live" video on flatpanel displays
- Supports panel resolutions up to 1280 x 1024, including 800x600 and 1024x768
- Supports non-interlaced CRT monitors with resolutions up to 1024 x 768 / 256 colors
- True-color and Hi-color display capability with flatpanels and CRT monitors up to 640x480 resolution
- Direct interface to Color and Monochrome Dual Drive (DD) and Single Drive (SS) panels (supports 8, 9, 12, 15, 16, 18 and 24-Bit data interfaces)
- Advanced power management features minimize power consumption during:
  - Normal operation
  - Standby (Sleep) modes
  - Panel-Off Power-Saving Mode
- Flexible onboard Activity Timer facilitates ordered shut-down of the display system
- Power sequencing control outputs regulate application of Bias voltage, +5 V to the panel and +12 V to the inverter for backlight operation
- SMARTMAP™ intelligent color to gray scale conversion enhances text legibility
- Text enhancement feature improves white text contrast on flatpanel displays
- Fully compatible with IBM™ VGA

65548 Display Capabilities

CRT Mode Resolution	Color [4]	Mono LCD Gray Scales [4]	DD STN LCD Colors [2, 3, 4]	9-Bit TFT LCD Colors [1, 2, 3, 4]	Video Memory	Simultaneous Display
320x200	256 /256K♦	61 / 61	256 / 226,981	256 / 185,193	512 KB	yes
640x480	16 / 256K♦	16 / 61	16 / 226,981	16 / 185,193	512 KB	yes
640x480	256 / 256K♦	61 / 61	256 / 226,981	256 / 185,193	512 KB	yes
800x600	16 / 256K♦	16 / 61	16 / 226,981	16 / 185,193	512 KB	yes with 1 MB
800x600	256 / 256K♦	61 / 61	256 / 226,981	256 / 185,193	512 KB	yes with 1 MB
1024x768	16 / 256K♦	16 / 61	16 / 226,981	16 / 185,193	512 KB	yes with 1 MB
1024x768	256 / 256K♦	61 / 61	256 / 226,981	256 / 185,193	1 MB	yes
1280x1024	16 / 256K♦	16 / 61	n/a	n/a	1 MB	n/a

Notes:

1 Larger color palettes and simultaneous colors can be displayed on 12-Bit, 18-Bit, and 24-Bit TFT panels via the 65540/545 video input port.
2 Includes dithering.
3 Includes frame rate control.
4 Colors are described as number of simultaneous on-screen colors and number of unique colors available in the color palette. 256K colors assumes DAC output mode is set to 6 bits of R, G & B. If DAC is set to 8-Bit output mode, the number of available colors is 16 M.

VGA Controller Chips

C&T	65548	16Bit local bus	1024k RAM	hardware accelerator

CRT Displays

The 65548 supports resolution fixed frequency and variable frequency analog monitors in interlaced and non-interlaced modes of operation. Digital monitor support is also built in.

The Video Performance:

On Board V1.xx: Local Bus 32 Bit 6000 Chr/ms performance with Landmark 2.0

Supported VGA Modes

Mode:	Type:	Colors:	CRT:	Text:	Graphic:	DRAM:	Monitor:	Refresh/HR:
0,1	Text	16	ABC	40 x 25	320 x 200	256k	CGA	70 Hz
2,3	Text	16	ABC	80 x 25	640 x 200	256k	CGA	70 Hz
4,5	Graphic	4	ABC	40 x 25	320 x 200	256k	CGA	70 Hz
6	Graphic	2	ABC	80 x 25	640 x 200	256k	CGA	70 Hz
7+	Text	Mono	ABC	80 x 25	720 x 350	256k	HGC	70 Hz
D	Planar	16	ABC	40 x 25	320 x 200	256k	CGA	70 Hz
E	Planar	16	ABC	80 x 25	640 x 200	256k	CGA	70 Hz
F	Planar	Mono	ABC	80 x 25	640 x 350	256k	EGA	70 Hz
10	Planar	16	ABC	80 x 25	640 x 350	256k	EGA	70 Hz
11	Planar	2	ABC	80 x 30	640 x 480	256k	VGA	60 Hz
12/12+	Planar	16	ABC/BC	80 x 30	640 x 480	256k	VGA	60 Hz/72Hz
13	Planar	256	ABC	40 x 25	320 x 200	256k	CGA	70 Hz (not 8 Bit Bus)
20	4 Bit Lin	16	ABC	80 x 30	640 x 480	512k	VGA	60 Hz
22	4 Bit Lin	16	BC	100 x 37	800 x 600	512k	SVGA	60 Hz
30	8 Bit Lin	256	ABC	80 x 30	640 x 480	512k	VGA	60 Hz/72 Hz
32	8 Bit Lin	256	BC	100 x 37	800 x 600	512k	SVGA	60 Hz/72 Hz
60	Text	16	ABC	132 x 25	1056 x 400	256k	MGA	68 Hz
61	Text	16	ABC	132 x 50	1056 x 400	256k	MGA	68 Hz
72	Planar	16	C	128 x 48	1024 x 768	512k	HVGA	60 Hz
79	Packed	256	ABC	80 x 30	640 x 480	512k	VGA	72 Hz
7C	Packed	256	BC	100 x 37	800 x 600	512k	SVGA	72 Hz

Supported on MSM486SEV boards with 1024k Video-RAM

24	4 Bit Lin	16	C	128 x 48	1024 x 768	1024k	HiVGA	60 Hz
26	4 Bit Lin	16	BC	128 x 48	1024 x 768	1024k	HiVGA	43 Hz
34	8 Bit Lin	256	C	128 x 48	1024 x 768	1024k	HiVGA	60 Hz
36	8 Bit Lin	256	BC	128 x 48	1024 x 768	1024k	HiVGA	43 Hz
40	15 Bit Lin	32k	ABC	80 x 30	640 x 480	1024k	VGA	60 Hz
41	16 Bit Lin	64k	ABC	80 x 30	640 x 480	1024k	VGA	60 Hz
7E	Packed	256	C	128 x 48	1024 x 768	1024k	HiVGA	60 Hz

A = PS/2 fixed frequency analog monitor;
B = Multifrequency CRT monitor like NEC Multisynch 3D or eq.
C = Nanao/EIZO 9070, NEC Multisynch 5D, or eq.

Simultaneous Flatpanel / CRT Display

The 65548 provides simultaneous display operation with Multi-Sync variable frequency or PS/2 fixed frequency CRT monitors and single panel-single drive LCDs (LCD-SS), dual panel-single drive LCDs (LCD-DS), dual panel-dual drive LCDs (LCD-DD) and plasma and EL panels (which contains single panel-single drive interfaces). Single drive panels sequence data in the same manner as CRTs so the 65548 provides simultaneous display with CRTs and LCD-SS, LCD-DS, plasma or EL panels by driving the panels with CRT timing. No external hardware is required. In contrast, LCD-DD panels require video data alternating between separate locations in the memory. In addition, a dual-drive panel requires data from both locations simultaneously. The 65548 also provides simultaneous display with LCD-DD and CRT monitors without using external VRAMs.

The internal VRAM frame buffers offer significant advantages relative to competitors' DRAM frame buffers. A DRAM frame accelerator requires that the flatpanel is refreshed at double of the CRTs vertical refresh rate. Therefore, an expensive 6.3 MHz LCD (with 120 Hz panel vertical refresh rate) is required for simultaneous display with 60 Hz CRT monitors when a DRAM frame buffer is used. Due to its higher bandwidth relative to DRAMS, a VRAM frame buffer can refresh both the flatpanel and CRT at the same vertical refresh rate. Therefore, inexpensive 3 MHz and 6 MHz LCDs (in addition to 6.3 MHz LCDs) can be used for simultaneous display with 60 Hz and 72 Hz CRT monitors when a VRAM frame buffer is used.

3.9.2 VGA/LCD BIOS Support

Each LCD display needs a specific adapted VGA BIOS. The CRT standard VGA BIOS is the standard equipment of this product.
To connect an LCD Display to this product, you will need to follow these steps:

1. Check the VGABIOS.DOC if the LCD BIOS is available.
   To receive the latest VGA BIOS enter our webpage www.digitallogic.com

If the LCD BIOS is available:

2. In the FLATPANEL-SUPPORT documentation you will find the description of the connection between the LCD and this product.
3. Download the corresponding LCD BIOS with the utility DOWN.EXE.
   Refer to chapter 3.6.7 DOWNLOADING THE VGA BIOS in this manual and follow the steps.
4. Restart the system and check the VGA BIOS header message. The LCD name must be visible for only a short time. The VGA BIOS message appears as first info page on the screen.
5. Stop the system, connect the LCD to the system and restart again.
6. If there is no image on the LCD as soon as the monitor begins to show the first text, stop the system immediately, otherways the LCD will be damaged!
7. Re-check the LCD connection.

A new LCD type (not yet available):

If the LCD BIOS for your LCD is not available, DIGITAL-LOGIC AG will adapt the LCD and provide you with one prototype-cable. To initialize this, we need the following from you:

1. An order to adapt the LCD (ask your sales contact for the expense).
2. Send the LCD panel, the datasheet of the LCD, a connector to the LCD and the inverter for the backlight to us.

Warning: **DIGITAL-LOGIC AG cannot be made responsible for damaged LCD displays! Even if the BIOS or any documentation of the LCD is wrong.**

3.9.3 Driver Resolutions and File names

Application:	Resolution:	Colors:	65535/65540:	Acce.: 65545/48
Windows 3.1	640 x 480	16	LINEAR4.DRV	VIDGX4.DRV
	800 x 600	16	LINEAR4.DRV	VIDGX4.DRV
	1024 x 768	16	LINEAR4.DRV	VIDGX4.DRV
	1280 x 1024	16	LINEAR4.DRV	VIDGX4.DRV
	640 x 480	256	LINEAR8:DRV	VIDGX8.DRV
	800 x 600	256	LINEAR8.DRV	VIDGX8.DRV
	1024 x 768	256	LINEAR8.DRV	VIDGX8.DRV
	480 x 640 Portrait	16	R12.DRV	
	640 x 480	64K	LINEAR16.DRV	VIDGX16.DRV
	640 x 480	16M	LINEAR24.DRV	VIDGX24.DRV
Windows NT 3.5x	640 x 480	16	chips.dll	
	800 x 600	16		
	1024 x 768	16		
	640 x 480	256		
	800 x 600	256		
Windows NT4.x	640 x 480 - 1024 x 768	16	miniport.dll	
	640 x 480 - 800 x 600	256		
Windows 95	640 x 480 - 1024 x 768	16	chips.dll	
	640 x 480 - 800 x 600	256		
OS/2	640 x 480	256	SV480256.DLL	WV480256.DLL
	800 x 600	256	PD600256.DLL	WV600256.DLL
	1024x768	256	PD768256.DLL	WV768256.DLL
VESA SuperVGA	640 x 480	16 - 64K	VESA.COM	
	800 x 600	16, 256	VESA.COM	
	1024x768	16, 256	VESA.COM	

3.9.3.1 Windows

1. Install Windows as you normally would for a VGA display.
2. Place the display driver disk 1 in drive A: and type A: <ENTER > to make this the default drive. Type SETUP, <ENTER> key.
3. Change to the Directory where you installed Windows.
4. If you are using Windows V3.0, type MIXFILES <Enter> to add the new drivers to the Windows setup menu.
5. Type SETUP, <Enter> to run the Windows setup program. It will show the current Windows configuration.
6. Follow the directions on the screen to complete the setup.

3.10 Keymatrix

3.10.1 Define a Keymatrix

Start the **EDITMATR.EXE** tool from the tooldisk. This tool enables the assignment of the keymatrix table in the keyboard controller. Start this tool and read the help screen after pressing F1. The following steps must be performed:

1. Press F2 to clear the whole matrix.
2. Connect the keymatrix to the appropriate connector, at the rows and columns.
3. Press on your keymatrix that key, which corresponds with the show key in the blue box.
4. For skipping a key, use the SPACE key on your standard keyboard.
5. Save the table to the key controller to a diskfile **(XXX.MTX).**
6. Exit the program.

Now you should test the operation of the connected keymatrix. Press every key and observe the shown character on the screen.

3.10.2 Store the Keymatrix into the EEPROM

To be sure, that the boot sequence takes the correct values into the key controller, you must store the key controller table after every modification into the EEPROM device.

To do this, use the **SAVEMATR.EXE** toolfile from the tooldisk.

The usage is very simple:

1. Start the SAVEMATR.EXE program.
2. Wait until the program is performed. No parameters are used.

3.10.3 Read DISK-File and store to the EEPROM-Matrix

A previously generated file XXX.MTX (with the aid of EDITMATR.EXE) the EEPROM may be used to reload the EEPROM keymatrix table from the disk. The EEPROM keymatrixtable will be transfered to the 8242PD Keymatrixcontroller while the next booting sequence.

To do this, use the **EDITMATR.EXE** tool from the tooldisk.The usage is very simple:

1. Start the EDITMATR.EXE program and load the XXX.MTX file, e.g. from a diskette
2. Exit from this program
3. Start the **SAVEMATR.EXE** program to store the values into the EEPROM
4. Reboot the system.
5. Alternatively use **EDITMATR.EXE XXX.MTX** to do this action in one rush

3.10.4 Old step by step version for storing the values to another board

Write the EEPROM-Keymatrix to Disk

The keymatrix stored in the EEPROM (with the SAVEMATR.EXE– tool) may be backuped to a diskfile for documentation or reproduction to other boards.

The tool WRMATRIX.EXE writes the contents of the EEPROM Keymatrixtable to the file XMATRIX.DAT.

To do this, use the WRMATRIX.EXE tool from the tooldisk.The usage is very simple:

1. Start the WRMATRIX.EXE program.
2. Wait until the program is performed. No parameters are used.

Read DISK-File and store to the EEPROM-Matrix

A previously generated file XMATRIX.DAT (with the aid of WRMATRIX.EXE) the EEPROM may be used to reload the EEPROM keymatrix table from the disk. The EEPROM keymatrixtable will be transfered to the 8242PD Keymatrixcontroller while the next booting sequence.

To do this, use the RDMATRIX.EXE tool from the tooldisk.The usage is very simple:

1. Be sure that the XMATRIX.DAT file is located on the same path as the RDMATRIX.EXE
2. The XMATRIX.DAT must be prevously generated with the WRMATRIX.EXE tool
3. Start the RDMATRIX.EXE program
4. Wait until the program is performed. No parameters are used
5. Start the RESTMATR.EXE program to store the values into the EEPROM

Reboot the system.

3.11 The ELAN400 videocontroller for ¼VGA Panel

The ELAN400 does also include an internal LCD-controller. This controller does supports CGA- and ¼-VGA-Displays. To use this controller, you have to load a configuration file into the serial EEPROM.
Use our tool E400_CGA.EXE to copy the specific table to the EEPROM. On the CD, you find this program and also a demo file called 640x200.DAT and 320x240.DAT.
Take reference to this file to understand the function of the configuration table into the serial eeprom

Usage: E400_CGA.EXE 640x200.DAT or

E400_CGA.EXE 320x240.DAT

This command will read out ********.DAT file, generate a table and fills up the serial EEPROM with this values. Only a part of serial EEPROM, reserved by DIGITAL-LOGIC AG, is used for this table. The custom part is free for the user.
At power up, the BIOS looks into this table and program the internal CGA-controller with the values from the table.

Attention: The Internal CGA-Controller will only be enabled, if the ELAN400 runs with 16Bit Memory-Bus and if there is no VIDEO-BIOS on board. !!

The 320x240.DAT is configurated for the display KCS3224 from Hitachi. Other displays will need a change of some values in this file. Take reference to the comments in DEMO.DAT and the 'Register Set Reference Manual' from AMD ELAN400.

Please note: There is nothing to see on the LCD until you have programmed the serial eeprom correctly! Standard board has this table already stored.
The picture has a CGA (320x200) resolution and will repeat the last 40 lines if the whole screen is filled.

Ask DLAG for a library based driver which has the whole resolution of 320x240.

If you have a board without a display, make a bootable diskette and copy E400_CGA.EXE and 320x240.DAT to this disk. Make a call from AUTOEXEC.BAT to the tool. Booting with this disk will then automatically copy the values to the eeprom. After re-powerup, the display will be initialized and you get picture.

SMART SM486PC CGA mode with MSM486SE (16bit)

The 16 bit mode is only working properly, when you have on the first RAM- bank a 2MB or 8MB memory device.

Total RAM (32 bit)	LOW BANK	HIGH BANK	Piggy pack	16 bit support	Total RAM (16 bit)	smart products	Products different memory architecture
---	2	---	---	**Yes**	2	SM486PC-2D2F MSM486SE-2	MSM486SL-2
4	2	2	---	No	(2)	SM486PC-4D2F MSM486SEV-4	MSM486SN/SV-4
---	2 + 2	---	Yes	**Yes**	4		
8	2 + 2	2 + 2	Yes	No	(4)		MSM486SN/SV-8
---	8	---	---	**Yes**	8	SM486PC-8D2F SM486PC2-8D2F MSM486SE-8	
16	8	8	---	(No)*	(8)	SM486PC-16D2F SM486PC2-16D2F	MSM486SL-16 (16bit)
---	8 + 8	---	Yes	**Yes**	16		
32	8 + 8	8 + 8	Yes	(No)*	(16)		MSM486SN/SV-32

***** Is might supported

3.11.1 ELAN400-Video Panel-Definitionfile

This is the Example for a 320x240.DAT

```
; E400_CGA Data File
; ------------------
; This is an initialisation file used by E400_CGA. Using and understanding
; this file will help you to init a 1/4-VGA/CGA display on E400 based DLAG-
; Boards. The Comment-Lines are not necessary and can be deleted...
;
; Comments and Text
; -----------------
; You can mak comments in this file. After a ';' the rest of the line will
; be interpret as comment.
; If you will give out a comment, us '*' instead of ';' as the first
; character in a line.
; Example:
* CGA-Init Table for LCD KCS3224
* (320x240 1/4-VGA Display)
;
; VGA-Inittable
; -------------
; Ever line in the table below is a command for the ELAN400 BIOS (V2.13..).
; There exist three kinds of commands:
;
;   ELAN400 Chipset indexed register access
;   ---------------------------------------
;   Usage:   10h VV II
;   With this command you can overwrite a E400 register. This command will
;   interpret by the BIOS as (assembler):
;                               mov al, II
;                               out 22h, al
;                               mov al, VV
;                               out 23h, al
;
;   IO-Direct mappet access
;   -----------------------
;   Usage:   0X XX VV
;   With this command you have direct the possibilty to give out a value
;   to an IO-port. This command will interpret by the BIOS as (pseudo
;   assembler):
;                               mov dx, XXX
;                               mov al, VV
;                               out dx, al
;
;   CGA-Controller register indexed access
;   --------------------------------------
;   Usage:   2B VV II
;   With this command you can change any register in the integreated CGA-
;   Controller, based at address 3D4h or 3B4h. This command will be inter-
;   pret by the BIOS as (assembler):
;                               mov dx, 304h
;                               add dx, B SHR 4 ; B is a value
;                               mov al, II
;                               out dx, al
;                               inc dx
;                               mov al, VV
;                               out dx, al
;
;   Fillup Font
;   -----------
;   Usage:   30h SS SS
;   Copy the internal Font-Set to the segment SSSS:0000
;
;
; Enable Internal Controller for CGA
;   +-------- 14h is the value
;   |  +---- DDh is the E400-Indexregister
10h 14h DDh
10h 00h DEh
; Set CGA-Mode
```

```
;    +------- 3D8h is the IO-Address
;    |  +--- 21h is the value
03h D8h 21h
03h D9h 00h

; Initialise CGA-Controller Register (take reference to the E400-Manual)
;+---------- D means register address is 3D4h/3D5h
;|   +------- 06 is the value
;|   |  +--- 0A is the index
2Dh 06h 0Ah
2Dh 07h 0Bh
2Dh 00h 0Ch
2Dh 00h 0Dh
2Dh 00h 0Eh
2Dh 00h 0Fh
2Dh 51h 30h ;Horizontal Total Register (Char per line)
2Dh 4Fh 31h ;Horizontal Display End Register (Char per line - 1)
2Dh 52h 32h ;Horizontal Line Pulse Start Register (Char per line + 2)
2Dh 4Fh 33h ;Horizontal Border End Register (Char per line - 1)
2Dh 00h 34h
2Dh 00h 35h
2Dh 00h 36h
2Dh 18h 37h
2Dh 1Dh 38h
2Dh 00h 39h
2Dh 00h 3Bh
2Dh 00h 3Ch
2Dh 00h 3Dh
2Dh 50h 3Eh ;Offset Register
2Dh 00h 3Fh
2Dh 07h 40h
2Dh 80h 41h
2Dh 00h 42h
2Dh 42h 43h
2Dh 00h 44h
2Dh 00h 45h
2Dh 00h 46h
2Dh 00h 47h
2Dh 00h 48h
2Dh 00h 49h
2Dh 00h 4Ah
2Dh 00h 4Bh
2Dh 09h 4Ch
2Dh 0Bh 4Dh
2Dh 0Bh 4Eh
2Dh 8Eh 4Fh
2Dh 92h 50h
2Dh 01h 51h
2Dh 0Ch 52h

; Fill up the Font-Table
;       +--- BC00h is the Start-Segment for the table
30h BCh 00h

; Write Protect the Font-Table
2Dh 80h 42h

; Disable Blanking
10h 00h DEh

; Set Videomode
03h D8h 29h
```

3.12 WatchDOG programming

Standard watchdog programming is via INT15, see chapter 9.2.2

Since the boardversion V1.2 and the smartModule V1.3, the watchdog has to be programmed as follows.
Ask DLAG for availability of this combination with that smartModule (> V3.0).

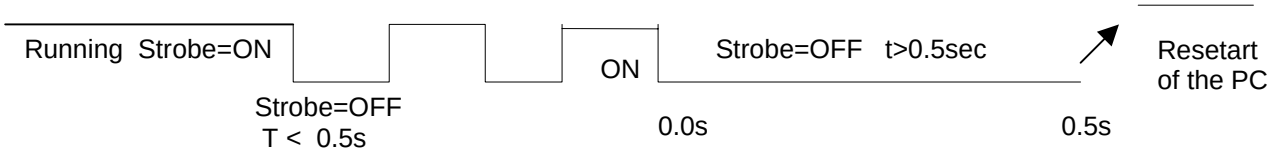
There has to be a hardware fix done before the watchdog runs properly (Is already made on the board V1.2):
- Signal BL0 has to be pulled down:
  Remove R125 and tigh to GND (pulled down)

```
;-----------------------------------------------------------------
; BL0 switch off
;-----------------------------------------------------------------
mov    al,38h                  ; pin Mux Register A
mov    dx,22h
out    dx,al
inc    dx
in     al,dx
and    al,3Fh                  ; disable BL0 CLKIO
out    dx,al

;-----------------------------------------------------------------
; KCOL7 switch off
;-----------------------------------------------------------------
mov    al,0C7h                 ; Keyboard Column Register
mov    dx,22h
out    dx,al
inc    dx
in     al,dx
and    al,7Fh                  ; disable KCOL7
out    dx,al

;-----------------------------------------------------------------
; KCOL7 switch on (creates a pulse from onboard MMV)
;-----------------------------------------------------------------
mov    al,0C7h                 ; Keyboard Column Register
mov    dx,22h
out    dx,al
inc    dx
in     al,dx
or     al,80h                  ; enable KCOL7
out     dx,al
```

After this line, the WatchDOG- reset will be generated 0.5sec later.



3.12.1 Watchdog variant B:

This is a customized version with R235= 0Ω (EPSK to watchdog) and U80 is not assembled).
Additionally is a customized BIOS necessary to support this variant B.

```
;*****************************************************************;
;* WatchDog for SM486PC Variant B (EPSK)                         *;
;*---------------------------------------------------------------*;

;-----------------------------------------------------------------
; BL0 (CLKIO) switch off
;-----------------------------------------------------------------
     mov               al,38h          ; Pin Mux Register A
     mov               dx,22h
     out               dx,al
     inc               dx
     in                al,dx
     and               al,3Fh            ; disable BL0 (CLKIO)
     out               dx,al

;-----------------------------------------------------------------
; Disable GPIO21 (EPSK)
;-----------------------------------------------------------------

     mov               al,0A8h           ; GPIO21 (EPSK)
     mov               dx,22h
     out               dx,al
     inc               dx
     in                al,dx
     and               al,0DFh           ; disable GPIO21 (EPSK)
     out               dx,al

;-----------------------------------------------------------------
; Enable GPIO21 (EPSK)
;-----------------------------------------------------------------

     mov               al,0A8h           ; GPIO21 (EPSK)
     mov               dx,22h
     out               dx,al
     inc               dx
     in                al,dx
     or                al,20h            ; enable GPIO21 (EPSK)
     out               dx,al
```

3.13 Ethernet IO-address and IRQ selection

The Ethernet interface is configured with the values stored in an EEPROM. The default values are:

Configuration: Jumper position		IOS0: J68	IOS1: J69	IOS2: J70		Base-address:	IRQ:	Bus:	Interface:
0		closed	closed	closed		300	5	16	10BASE-T
1		open	closed	closed		300	10	16	10BASE-T
2		closed	open	closed		320	10	16	10BASE-T
3		open	open	closed		340	11	16	10BASE-T
4		**closed**	**closed**	**open**		**340**	**5**	**16**	**10BASE-T**
5		open	closed	open		340	10	16	10BASE-T
6		closed	open	open		320	11	16	10BASE-T
7		open	open	open		Software downloadable configuration			

On the board, the default configuration and the selected configuration are labeled. The label also contains the unique node address:

ATTENTION, if this configuration table does not correspond with yours, then you have to update your Ethernet EEPROM values ! Please download the Application Note 077 / 085, it decribes what you have to do in this situation, at http://www.digitallogic.com **->support**

Example Label:

LAN0 300/i 5/16B/T/WS
LAN1 300/i10/16B/T/WS
LAN2 320/i10/16B/T/--
LAN5 340/i10/16B/T/WS
LAN6 320/i11/16B/T/--
Node: 003059aabbcc

Note, LAN3 & LAN4 are not printed on the label, because of too few space.

You can change the hardware configuration by modifying the jumper IOS0 - IOS2.

The software downloaded configuration is recommended only for special cases, where the values must be altered, without soldering the jumpers onboard. The downloaded configuration works only with Novell V3.11 and Novell Lite and all other ODI driver NOS.

For OS/2, NT and UNIX applications always use the hardware configuration, because the drivers are adapted only for hardware configuration schemes.

3.13.1 Ethernet-drivers

We provide a lot of drivers and tools for the Ethernet part. Below you will see which drivers in which folders you will find on the MICROSPACE Application CD:

Folder	Operating Systems
16bit	DOS, Windows 3.11
32bit	Windows 95/98, Windows NT
CE	Windows CE 2.x
Linux	Linux
Novell	Novell Netware Server
Utility	Tools & Utilities (DOS)

➔ **Get the latest drivers from DIGITAL-LOGIC's homepage at** http://www.digitallogic.com **(support)**

3.14 Installation examples for IRQ11

For using the LAN, please make the following settings:

Hardware:

JUMPERS	Remarks		
Jumper 68		closed	
Jumper 69		open	
Jumper 70		open	
Jumper 41	(solderjumper)	closed	IRQ5 mapped to IRQ11
Jumper 49	(solderjumper)	open	
Jumper 40	(solderjumper)	open	
Jumper 50	(solderjumper)	open	

Bios:
components -> PIRQ mapping -> PIRQ5 map to IRQ 11

3.14.1 DOS NOVELL 4.11

1) Copy the directory x:\products\msm486se_\tools\network\nov411 to your HDD (c:\....)
2) Copy the file x:\products\msm486se_\tools\network\nov411.bat to your HDD
3) Start batchfile NOV411.BAT

3.14.2 DOS CLIENT NT

1) Proceed as described in the MICROSOFT document, which you find on our CD x:\products\msm486se_\tools\network\dosclnt
2) Run first MEMMAKER.EXE
3) Unzip the file DOSCLNT.ZIP (x:\products\msm486se_\tools\network\dosclnt\)
4) Run SETUP.EXE on disk1
5) Choose NETWORKADAPTER MISSING
6) OK
7) OK
8) Change NAMES: use your own definitions, names
9) Change SETUP OPTIONS: LOGON VALIDATIONS => DOMAIN REGISTER

10) Change NETWORK OPTIONS:
    CHANGE SETTINGS INTERRUPT=11
    I/O BASE = 320
    PHYSCAL MEDIA TYPE = 10BASE/T
    PROTOCOL: TCP/IP and NETBUI
11) OK
12) Run disk2 and after PC will reboot
13) Strike key F5 while rebooting and edit the PROTOCOL.INI file (c:\net\...)
    SUBNETMASK = your own definition
    IP ADDRESS 0 = your own definition
    DISABLE DHCP = 1
    Check if I/0 BASE = 0x320 and IRQ = 11
14) Reboot your system

3.14.3 WIN95

Driver:
Please use the 32bit NDIS 3 driver -> lanonly -> SMC9000 Ethernet Adapter

IRQ = 11

E/A Address = 320 - 33F

3.15 Installation examples for IRQ5

For using the LAN, please make the following settings:

Hardware:

JUMPERS	**Remarks**		
Jumper 68		closed	
Jumper 69		closed	
Jumper 70		closed	
Jumper 41	(solderjumper)	open	IRQ5 mapped to IRQ5
Jumper 49	(solderjumper)	open	
Jumper 40	(solderjumper)	open	
Jumper 50	(solderjumper)	open	

Bios:
components -> PIRQ mapping -> PIRQ5 map to IRQ 5

3.15.1 DOS NOVELL 4.11

1) Copy the directory x:\products\msm\msm486se_\tools\network\novell_irq5\nov411
   to your HDD (c:\....)
2) Copy the file x:\products\msm\msm486se_tools\network\novell_irq5\nov411.bat to your HDD
3) Start batchfile NOV411.BAT

4 DESCRIPTION OF THE CONNECTORS

Jumper	Texture	Pin	Remarks
J02	PC104 Bus	104	2.54mm
J03	Keymatrix	2x15	2.54mm
J04	Power	2x4	2.54mm
J05	Keyb, Mouse	2x5	2.54mm
J08	COM Primary serial of UPC	2x5	2.54mm
J09	IDE	2x22	2mm
J10	Floppy	26	FDC micro
J17	VGA / LCD, since V1.2	2x17	2mm
J17	VGA / LCD, until V1.1	2x20	2mm
J38	COM Secondary serial of UPC	2x5	2.54mm
J51	1/4VGA LCD, until V1.1 and V1.3 with ELAN internal COM / TTL	2x10	2.54mm
J51	1/4VGA LCD, only V1.2	2x8	2.54mm
J66	LAN RJ45 until V1.2 assembled	8	
J80	COM4	2x5	2.54mm
J81	LPT1	2x13	2.54mm
J86	LAN 4 pin (until V1.2 not assembled)	4	2.54mm
J87	PC-Card Signal A (MSM486SVPC)	20	2.54mm
J89	PCMCIA Card-B signals and COM internal ELAN400	2x10	2mm
J92	LAN LED's	3	2.54mm
J94	COM internal ELAN400 /RS485	2x5	2.54mm
J98	VGA	2x5	2.54mm
U73	Compact Card	2x25	

J8 Serial port, primary COM

Header onboard	D-SUB connector	Signal
Pin 1	Pin 1	= DCD
Pin 2	Pin 6	= DSR
Pin 3	Pin 2	= RxD
Pin 4	Pin 7	= RTS
Pin 5	Pin 3	= TxD
Pin 6	Pin 8	= CTS
Pin 7	Pin 4	= DTR
Pin 8	Pin 9	= RI
Pin 9	Pin 5	= GND
Pin10		= VCC

J38 Serial port, secondary COM

Header onboard	D-SUB connector	Signal
Pin 1	Pin 1	= DCD
Pin 2	Pin 6	= DSR
Pin 3	Pin 2	= RxD
Pin 4	Pin 7	= RTS
Pin 5	Pin 3	= TxD
Pin 6	Pin 8	= CTS
Pin 7	Pin 4	= DTR
Pin 8	Pin 9	= RI
Pin 9	Pin 5	= GND
Pin10		= VCC

J80 Serial port COM4

Header onboard	D-SUB connector	Signal
Pin 1	Pin 1	= DCD
Pin 2	Pin 6	= DSR
Pin 3	Pin 2	= RxD
Pin 4	Pin 7	= RTS
Pin 5	Pin 3	= TxD
Pin 6	Pin 8	= CTS
Pin 7	Pin 4	= DTR
Pin 8	Pin 9	= RI
Pin 9	Pin 5	= GND
Pin10		= VCC

J10 Floppy Disk interface connector

FD26: Pin	Signal Name	Function	in/out
1	VCC	+5 volts	
2	IDX	Index Pulse	in
3	VCC	+5 volts	
4	DS2	Drive Select *	out
5	VCC	+5 volts	
6	DCHG	Disk Change	in
10	M02	Motor on *	out
12	DIRC	Direction Select	out
14	STEP	Step	out
16	WD	Write Data	out
17	GND	Signal grounds	
18	WE	Write Enable	out
19	GND	Signal grounds	
20	TRKO	Track 0	in
21	GND	Signal grounds	
22	WP	Write Protect	in
23	GND	Signal grounds	
24	RDD	Read Data	in
25	GND	Signal grounds	
26	HS	Head Select	out

* drive A: or B: may be selected with jumpers; default is A: selected.

J9 IDE interface - 44pin 2mm connector

Pin	Signal	Pin	Signal
Pin 1	= Reset (active low)	Pin 2	= N.C.
Pin 3	= D7	Pin 4	= D8
Pin 5	= D6	Pin 6	= D9
Pin 7	= D5	Pin 8	= D10
Pin 9	= D4	Pin 10	= D11
Pin 11	= D3	Pin 12	= D12
Pin 13	= D2	Pin 14	= D13
Pin 15	= D1	Pin 16	= D14
Pin 17	= D0	Pin 18	= D15
Pin 19	= GND	Pin 20	= N.C. (keypin)
Pin 21	= N.C.	Pin 22	= GND
Pin 23	= IOW(active low)	Pin 24	= GND
Pin 25	= IOR(active low)	Pin 26	= GND
Pin 27	= IOCHRDY	Pin 28	= ALE / Master-Slave (not used)
Pin 29	= nc	Pin 30	= GND
Pin 31	= IRQ14	Pin 32	= IOCS16 (active low)
Pin 33	= ADR1	Pin 34	= N.C.
Pin 35	= ADR0	Pin 36	= ADR2
Pin 37	= CS0 (active low)	Pin 38	= CS1 (active low)
Pin 39	= LED (active low)	Pin 40	= GND
Pin 41	= VCC Logic	Pin 42	= VCC Motor
Pin 43	= GND	Pin 44	= GND

J81 Printerport (Centronics)

The printer connector provides an interface for 8 Bit Centronics printers.

Header onboard	D-SUB connector	Signal
Pin 1	Pin 1	= Strobe
Pin 3	Pin 2	= Data 0
Pin 5	Pin 3	= Data 1
Pin 7	Pin 4	= Data 2
Pin 9	Pin 5	= Data 3
Pin 11	Pin 6	= Data 4
Pin 13	Pin 7	= Data 5
Pin 15	Pin 8	= Data 6
Pin 17	Pin 9	= Data 7
Pin 19	Pin 10	= Acknowledge
Pin 21	Pin 11	= Busy
Pin 23	Pin 12	= paper end
Pin 25	Pin 13	= select
Pin 2	Pin 14	= autofeed
Pin 4	Pin 15	= error
Pin 6	Pin 16	= init printer
Pin 8	Pin 17	= shift in (SI)
Pin 10,12,14,16,18	Pin 18 - 22	= left open
Pin 20,22,24	Pin 23 - 25	= GND

J4 Power supply connector

Pin	Signal	Pin	Signal
Pin 1	= Ground	Pin 2	= VCC +5V
Pin 3	= AC-Sense Input	Pin 4	= +12 Volt *)
Pin 5	= RX COM1 TTL	Pin 6	= TX COM1 TTL
Pin 7	= Ground	Pin 8	= VCC +5V

AC-Sense: +5V = Disabled Powermanager (default)
0V = Enabled Powermanager (close Pin 1 with Pin 3)
AC-Sense is internaly pulled up with 10kOhm.

*) The +12V are only for the LCD backlight used.
RS485 A and B: COM1 of the ELAN400 controller /TTL

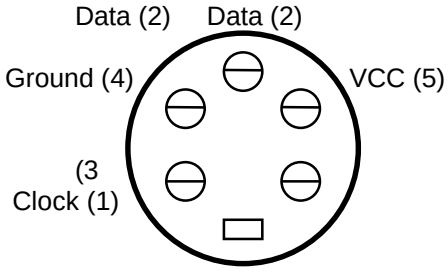
J5 Keyboard PS/2-Mouse Utility connector

Attention: The speaker must be connected to VCC, to have a low inactive current in the speaker!

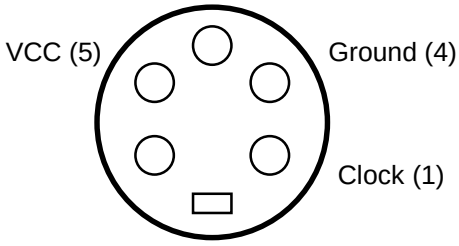
Pin	Signal	Pin	Signal
Pin 1	= Speaker Out	Pin 2	= Suspend/Resume
Pin 3	= Reset In	Pin 4	= VCC
Pin 5	= Keyboard Data	Pin 6	= Keyboard Clock
Pin 7	= Ground	Pin 8	= Ext. battery +
Pin 9	= PS/2 Mouse Clock	Pin10	= PS/2 Mouse Data

The Utility connector must be wired to a standard AT-female connector:

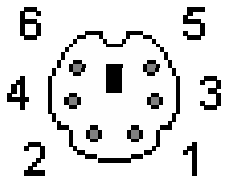
Frontside AT-Keyboard (female)



Solderside AT-Keyboard (female)



PS/2 Frontside (female)



Connector and adapter

	Mini- DIN PS/2 (6 PC)	**DIN 41524 (5 PC)**	**Remarks**
Shield	Shield	Shield	KEYBOARD
DATA	1	2	
GND	3	4	
VCC (+5V)	4	5	
CLK	5	1	
	Mini- DIN PS/2 (6 PC)		Mouse
VCC (+5V)	4		
DATA	1		
GND	3		
CLK	5		

J82 and J83 Battery soldering location

Pin J82	**Signal**	**Pin J83**	**Signal**
Pin 1	= Ground	Pin 1	= Battery +
Pin 2	= Ground	Pin 2	= Battery +

J17 LCD connector (only on MSM486SEV version), new since V1.2

VGA-LCD Interface (flatpanel signals):

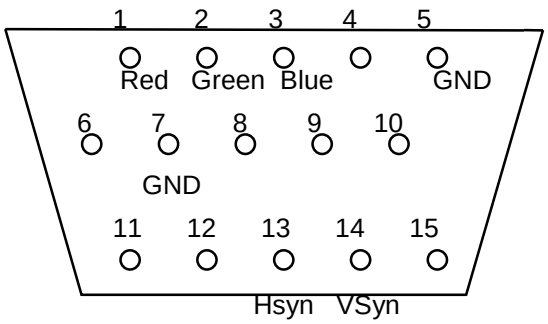
Pin	Signal	Pin	Signal
1	M-Signal (altern. DE)	2	FLM
3	P18	4	LP
5	VCC	6	GND
7	VDD enable (TTL)	8	Shift Clock
9	B'light enable (TTL)	10	P3
11	P2	12	P17
13	P1	14	P16
15	P0	16	P7
17	Enable VEE (TTL)	18	P6
19	VCC Panel (5V/3,3V)	20	P5
21	P4	22	P19
23	P8	24	P9
25	P10	26	P11
27	P12	28	P13
29	P14	30	P15
31	P20	32	P21
33	P22	34	P23

J98 VGA Monitor (CRT-Signals) (only on MSM486SEV version), new since V1.2

J98 Header			15 pins HiDensity DSUB	
	10 Pin -M	Signal	Pin	Signal
	Pin 2	VGA red	Pin 1	Red
	Pin 4	VGA green	Pin 2	Green
	Pin 6	VGA blue	Pin 3	Blue
	Pin 8	Horizontal Synch	Pin 13	H-Synch
	Pin 9	Vertical Synch	Pin 14	V-Synch
			Pin 5 + 11	Bridge
	Pin 1	Ground	Pin 5, 6, 7, 8	Ground

The VGA-CRT signals from J98 must be wired to a standard VGA HiDensity DSub connector (female):
The LCD signals must be wired panel specific.

Solderside view of the female 15pin HiDSub



J17L LCD connector (only on MSM486SEV version), until version1.1

VGA-LCD Interface (flatpanel signals):
Signals P20-P23 are located on the J17M connector

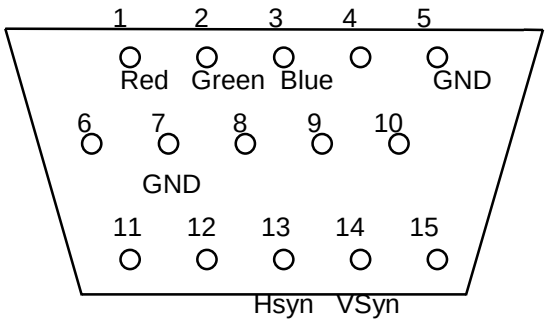
Pin	Signal	Pin	Signal
1	M-Signal (altern. DE)	2	FLM
3	P18	4	LP
5	VCC	6	GND
7	VDD enable (TTL)	8	Shift Clock
9	B'light enable (TTL)	10	P3
11	P2	12	P17
13	P1	14	P16
15	P0	16	P7
17	Enable VEE (TTL)	18	P6
19	VCC Panel (5V/3,3V)	20	P5
21	P4	22	P19
23	P8	24	P9
25	P10	26	P11
27	P12	28	P13
29	P14	30	P15

J17M VGA Monitor (CRT-Signals) (only on MSM486SEV version), until version1.1

J17 L/M Header			15 pins HiDensity DSUB	
40 Pin -L	10 Pin -M	Signal	Pin	Signal
Pin 32	Pin 2	VGA red	Pin 1	Red
Pin 34	Pin 4	VGA green	Pin 2	Green
Pin 36	Pin 6	VGA blue	Pin 3	Blue
Pin 38	Pin 8	Horizontal Synch	Pin 13	H-Synch
Pin 39	Pin 9	Vertical Synch	Pin 14	V-Synch
			Pin 5 + 11	Bridge
Pin 31	Pin 1	Ground	Pin 5, 6, 7, 8	Ground
Pin 33	Pin 3	LCD-P20		
Pin 35	Pin 5	LCD-P21		
Pin 37	Pin 7	LCD-P22		
Pin 40	Pin 10	LCD-P23		
Pin 41-44	n.c.	not connected		

The VGA-CRT signals from J17 must be wired to a standard VGA HiDensity DSub connector (female):
The LCD signals must be wired panel specific.

Solderside view of the female 15pin HiDSub



J2 PC/104 BUS interface

Pin	A:	B:	C:	D:
0			Ground	Ground
1	IOCHCK	Ground	SBHE	MEMCS16
2	SD7	RESET	LA23	IOCS16
3	SD6	+5V	LA22	IRQ10, (redirected)
4	SD5	(IRQ9), (redirected)	LA21	IRQ11, (redirected)
5	SD4	(-5V)	LA20	IRQ12, (PIRQ2)
6	SD3	DRQ2	LA19	IRQ15, (redirected)
7	SD2	(-12V)	LA18	IRQ14, (PIRQ0)
8	SD1	(SRDY)	LA17	(DACK0)
9	SD0	+12V	MEMR	DRQ0
10	IOCHRDY	Ground	MEMW	(DACK5)
11	AEN	SMEMW	SD8	(DRQ5)
12	SA19	SMEMR	SD9	(DACK6)
13	SA18	SIOW	SD10	(DRQ6)
14	SA17	SIOR	SD11	(DACK7)
15	SA16	(DACK3)	SD12	(DRQ7)
16	SA15	(DRQ3)	SD13	+5 Volt
17	SA14	DACK1	SD14	MASTER
18	SA13	DRQ1	SD15	Ground
19	SA12	REF	Ground	Ground
20	SA11	SYSCLK		
21	SA10	IRQ7 (PIRQ7)		
22	SA9	IRQ6 (PIRQ6)		
23	SA8	IRQ5 (PIRQ5)		
24	SA7	IRQ4 (PIRQ4)		
25	SA6	IRQ3 (PIRQ3)		
26	SA5	DACK2		
27	SA4	TC		
28	SA3	ALE		
29	SA2	+5 Volt		
30	SA1	OSC		
31	SA0	Ground		
32	Ground	Ground		

Note:
- Redirected IRQx may not be used simultaneously.
- (xx) are may not available on the ELAN

Onboard used signals (not for external use)

IRQ3, IRQ4	COM1 / COM2
IRQ7	LPT1
IRQ6	FD
IRQ14	HD
IRQ12	PS/2 Mouse
TC	FD
DACK2 and DRQ2	FD

J87 PC-CARD A connector

Pin	Signal		
1	ICDIR		
2	WE		
3	OE		
4	CD		
5	RDY		
6	BVD1		
7	BVD2		
8	WP		
9	WAIT		
10	RST		
11	REG		
12	MCEH		
13	MCEL		
14	VCC		
15	VPP1		
16	VPP2		
17	WPB		
18	Not connected		
19	SA24		
20	SA25		

U73 Compact card, typ1

Pin	Signal
Pin 01	= GND
Pin 02	= D3
Pin 03	= D4
Pin 04	= D5
Pin 05	= D6
Pin 06	= D7
Pin 07	= CS0 (active low)
Pin 08	= (A10) NC
Pin 09	= GND
Pin 10	= (A9) NC
Pin 11	= (A8) NC
Pin 12	= (A7) NC
Pin 13	= Vcc
Pin 14	= (A6) NC
Pin 15	= (A5) NC
Pin 16	= (A4) NC
Pin 17	= (A3) NC
Pin 18	= ADR2
Pin 19	= ADR1
Pin 20	= ADR0
Pin 21	= D0
Pin 22	= D1
Pin 23	= D2
Pin 24	= IOCS16 (active low)
Pin 25	= (CD2) NC
Pin 26	= (CD1) NC
Pin 27	= D11
Pin 28	= D12
Pin 29	= D13
Pin 30	= D14
Pin 31	= D15
Pin 32	= CS1 (active low)
Pin 33	= (Vs1) NC
Pin 34	= IOR (active low)
Pin 35	= IOW (active low)
Pin 36	= Vcc
Pin 37	= IRQ
Pin 38	= Vcc
Pin 39	= CEL
Pin 40	= (VS2) NC
Pin 41	= Reset (active low)
Pin 42	= IOCHRDY (active low)
Pin 43	= (INPACK-) NC
Pin 44	= Vcc
Pin 45	= LED (active low)
Pin 46	= (PDIAG) NC
Pin 47	= D8
Pin 48	= D9
Pin 49	= D10
Pin 50	= GND

Pin order (solder view)

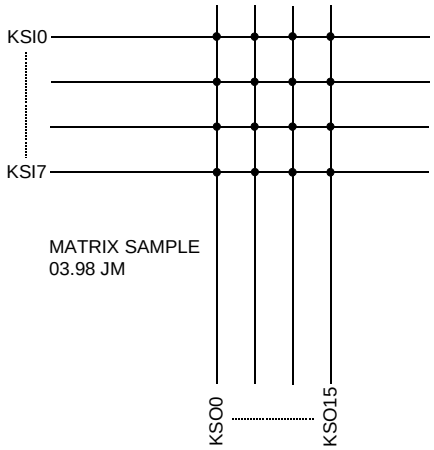
Pin 25	Pin 1
Pin 50	Pin 26

J3 Keymatrix connector

Keymatrix Interface:
Signals are located on the connector

Pin	Signal	Pin	Signal
1	Column 1 (KSO0)	2	Column 2 (KSO1)
3	Column 3 (KSO2)	4	Column 4 (KSO3)
5	Column 5 (KSO4)	6	Column 6 (KSO5)
7	Column 7 (KSO6)	8	Column 8 (KSO7)
9	Column 9 (KSO8)	10	Column 10 (KSO9)
11	Column 11 (KSO10)	12	Column 12 (KSO11)
13	Column 13 (KSO12)	14	Column 14 (KSO13)
15	Column 15 (KSO14)	16	Column 16 (KSO15)
17	Row 0 (KSI0)	18	Row 1 (KSI1)
19	Row 2 (KSI2)	20	Row 3 (KSI3)
21	Row 4 (KSI4)	22	Row 5 (KSI5)
23	Row 6 (KSI6)	24	Row 7 (KSI7)
25	GND	26	LED ground
27	LED Fn-PAD (anode)	28	LED CAPs Lock (anode)
29	LED NumLock (anode)	30	LED SCR (anode)

Each LED must be connected with the cathode to the LED-ground signal.



J51 1/4VGA & CGA LCD connector (only on MSM486SE version), new since V1.3

Character/Graphic LCD Interface:
Signals are located on the connector

Pin	Signal	Pin	Signal
1	VL0, Data0	2	VL1, Data1
3	VL2, Data2	4	VL3, Data3
5	VL4, Data4	6	VL5, Data5
7	VL6, Data6	8	VL7, Data7
9	VL8, Lineclock	10	VL9, M-Signal
11	VL10, FLM	12	VL11, Shiftclock
13	COM: RXD (TTL)	14	COM: TXD (TTL)
15	COM: RTS (TTL)	16	COM: CTS (TTL)
17	Ground	18	Vo BIAS for LCD
19	VEE for LCD	20	Switched 5V/0.5A

VEE: The LCD negativ supply, adjusted by R142 trimmer
VXX: The LCD negativ BIAS supply , adjusted by the trimmer R143

Only the LCD-signals:

Pin on J51	LCD Signal	Description
1	Data 0	5V CGA-Graphic LCD Databit
2	Data 1	5V CGA-Graphic LCD Databit
3	Data 2	5V CGA-Graphic LCD Databit
4	Data 3	5V CGA-Graphic LCD Databit
5	Data 4	5V CGA-Graphic LCD Databit
6	Data 5	5V CGA-Graphic LCD Databit
7	Data 6	5V CGA-Graphic LCD Databit
8	Data 7	5V CGA-Graphic LCD Databit
9	M	5V CGA-Graphic LCD AC Modulation for the panel
10	LClk	5V CGA-Graphic LCD Line Clock (also refered as CL1 or CP1)
11	SCK	5V CGA-Graphic LCD Shift Clock (also refered as CL2 or CP2)
12	FRM	5V CGA-Graphic LCD Line Frame Start (also refered as FLM or S)
17	GND	Ground
18	Vo	LCD BIAS Voltage -28V to –15V software adjustable
19	VEE	LCD VEE Voltage - 28v to –15V
20	VDD	5V CGA-Graphic LCD VDD Voltage

Attention:
A wrong cable connection may destroy the LCD panel and/or the MSM486SE/SEV product.

J51 1/4VGA & CGA LCD connector (only on MSM486SE version), only on V1.2

Character/Graphic LCD Interface:
Signals are located on the connector

Pin	Signal	Pin	Signal
1	VCC	2	LCD_D0
3	LCD_D1	4	LCD_D2
5	LCD_D3	6	LCD_D4
7	LCD_D5	8	LCD_D6
9	LCD_D7	10	LCD_M
11	LCD_LC	12	LCD_SHFCLK
13	LCD_FRM	14	VEE
15	VXX	16	GND

VEE: The LCD negativ supply, adjusted by R142 trimmer
VXX: The LCD negativ BIAS supply , adjusted by the trimmer R143

Attention:
A wrong cable connection may destroy the LCD panel and/or the MSM486SE/SEV product.

J51 1/4VGA & CGA LCD connector (only on MSM486SE version), until V1.1

Character/Graphic LCD Interface:
Signals are located on the connector

Pin	Signal	Pin	Signal
1	VCC	2	LCD_D0
3	LCD_D1	4	LCD_D2
5	LCD_D3	6	LCD_D4
7	LCD_D5	8	LCD_D6
9	LCD_D7	10	LCD_M
11	LCD_LC	12	LCD_SHFCLK
13	LCD_FRM	14	LCD_VEE_Enable/
15	LCD_VDD_Enable/	16	GND
17	nc	18	VEE
19	VXX	20	nc

LCD_VDD_Enable and LCD_VEE_Enable are TTL logic signals (activ low).

VEE: The LCD negativ supply, adjusted by R142 trimmer
VXX: The LCD negativ BIAS supply , adjusted by the trimmer R143

Attention:
A wrong cable connection may destroy the LCD panel and/or the MSM486SE/SEV product.

J89 COM1 (RS232 / RS485) and PCCard-B signals

Signals are located on the connector

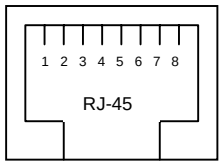
Pin	Signal	Pin	Signal
1	VCC	2	GND
3	BVCC	4	BVPP1
5	BVPP2	6	BVD2
7	BVD1	8	BRDY
9	BCD	10	BREG
11	BRST	12	B_MCH
13	B_MCL	14	COM1: DCD / TX+
15	COM1: DSR / TX-	16	COM1: RXD / RX+
17	COM1: RTS / RX-	18	COM1: TXD
19	COM1: CTS	20	COM1: DTR

J94 COM1 (RS232 / RS485), new since V1.2

Pin	Signal	Pin	Signal
1	COM1: DCD / TX+	2	COM1: DSR / TX-
3	COM1: RXD / RX+	4	COM1: RTS / RX-
5	COM1: TXD	6	COM1: CTS
7	COM1: DTR	8	RT
9	GND	10	Vcc

J66 10 BASE-T Interface connector, RJ45 (not assembled since V1.3)
J86 10 BASE-T Interface connector

Pin J86	Signal	RJ-45 Pin *	Signal
Pin 1	= TX+	Pin 1	= TX+
Pin 2	= TX-	Pin 2	= TX-
Pin 3	= RX+	Pin 3	= RX+
Pin 4	= RX-	Pin 6	= RX-



* This signals are ready to connect directly to a RJ-45 connector.

J92 LAN LED's connector, new since V1.2

Pin *	Signal		Remarks
Pin 1	= Linked (LED also on board)		330Ω at Vcc included
Pin 2	= Vcc		
Pin 3	= Select (LED also on board)		330Ω at Vcc included

5 JUMPER DESCRIPTIONS

The figure shows the location of all jumper blocks on the MSM486SE/SEV board. The numbers shown in this figure are silk screened on the board so that the pins can easily be located. This chapter refers to the individual pins for these jumpers. Be careful when you change some jumpers. Some jumpers are soldering bridges, you need a miniature soldering station with vacuum pump.

5.1 The jumpers on this MICROSPACE product

RM2.54mm Jumpers on the front side

		1-2 = open	2-3 = closed	Rem
J18	LCD Standby / sleep*	software *	**fix**	
J40	IRQ12 mapping	**open = IRQ12 ***	closed = IRQ10	
J45	IRQ5 mapping	**open = IRQ5 ***	closed = IRQ15	
J41	IRQ5 mapping	**open = IRQ5 ***	closed = IRQ11	
J49	IRQ14 mapping	**open = IRQ14 ***	closed = IRQ11	
J50	IRQ14 mapping	**open = IRQ14 ***	closed = IRQ9	
J68	LAN SEL0	open = high	**closed = low**	
J69	LAN SEL1	open = high	**closed = low**	
J70	LAN SEL2	**open = high**	closed = low	
J77	EPSK Memory (see SM486PC)	**1-2 = 32Bit**	2-3 = 16Bit**	SEV
J78	EPDI VGA (see SM486PC)	1-2 = ISA**	**2-3 = VESA**	SEV
J85	CardFlash Select:	open = Slave	**closed = Master**	
J90	RESUME	**Yes**	No	new

Note: Standard jumper settings for the MSM486SE are: J77=16Bit and J78= ISA

The jumpers on the rear side (solder jumpers)

		1-2 = open	2-3 = closed	Rem
J16	VCCPAN	**1-2 = 5V**	2-3 = 3.3V	
J27	Enable BIAS voltage from	1-2 = VEE Control	**2-3 = from VDD Control**	
J84	LCD-panel supply	**1-2 = 5 V**	2-3 = 3.3 V	
J88	LCD- backlight supply	**1-2 = 5V**	2-3 = 3.3V	
J95	ISA bus connection IRQ 14	**No**	Yes	new
J96	ISA bus connection IRQ 12	**No**	Yes	new
J97	ISA bus connection IRQ 5	**No**	Yes	new
J99	LCD contrast power supply	**negativ**	positiv	new
J100	LCD VEE	**negativ**	positiv	new

* Software controlled
** The 16 bit mode is only working properly, when you have on the first RAM- bank a 2MB or 8MB memory device see also chapter 3.11.

J91 COM4 to IRQ, hardware mapping, new since v1.2

Solder 0Ω resistor in, to activate

Row	Signal	Row	Signal
Pin 1	= IRQ3	Pin 9	COM init
Pin 2	= IRQ4	Pin 10	COM init
Pin 3	= IRQ5	Pin 11	COM init
Pin 4	= IRQ6	Pin 12	COM init
Pin 5	= IRQ7	Pin 13	COM init
Pin 6	= IRQ12	Pin 14	COM init
Pin 7	= IRQ14	Pin 15	COM init
Pin 8	= IRQ15	Pin 16	COM init

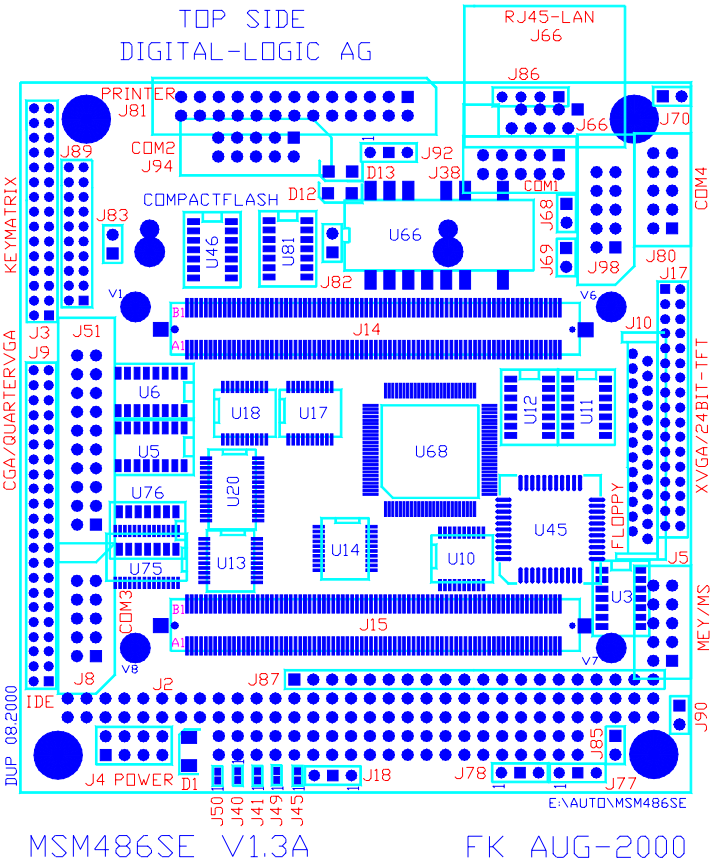
Ethernet Configuration Table, see also chapter 3.13

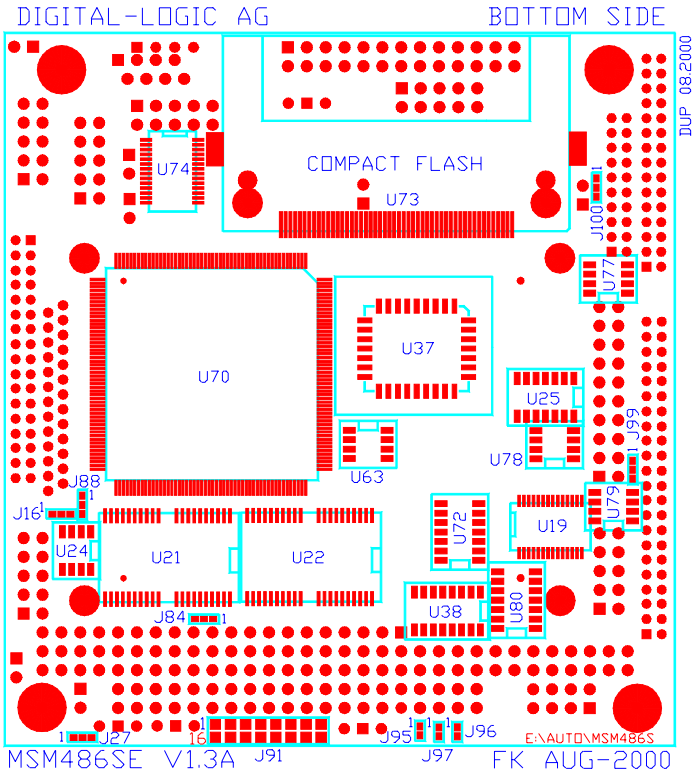
IOS2	IOS1	IOS0	LANx
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	**0**	**0**	**4**
1	0	1	5
1	1	0	6
1	1	1	7

Settings written in bold are defaults!

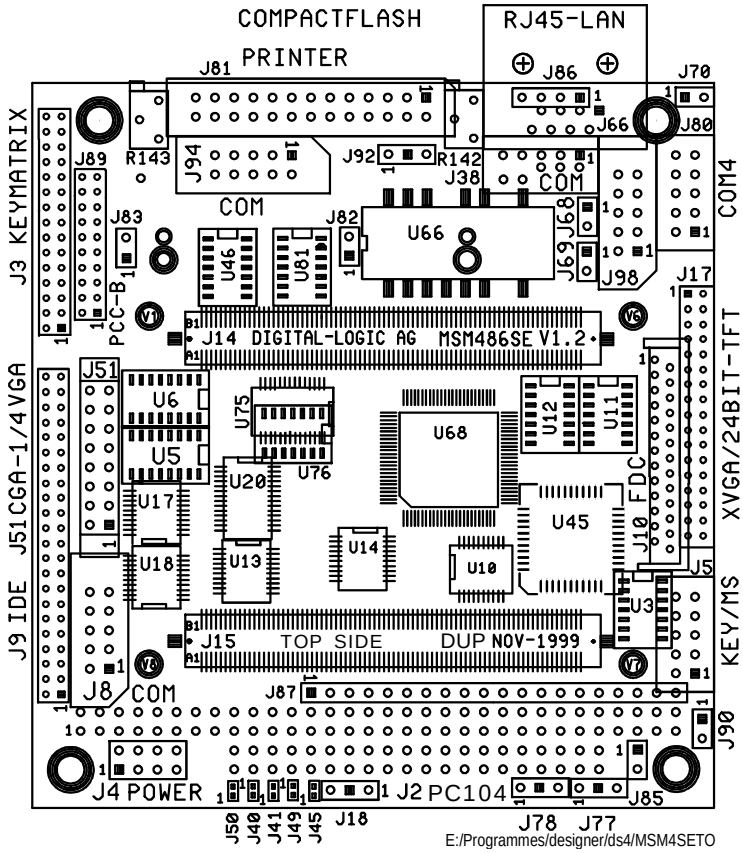
These settings are factory defaults. Do not modify, otherwise the keycontroller will not operate.

5.1.1 Jumper locations, since version 1.3 (not available yet)

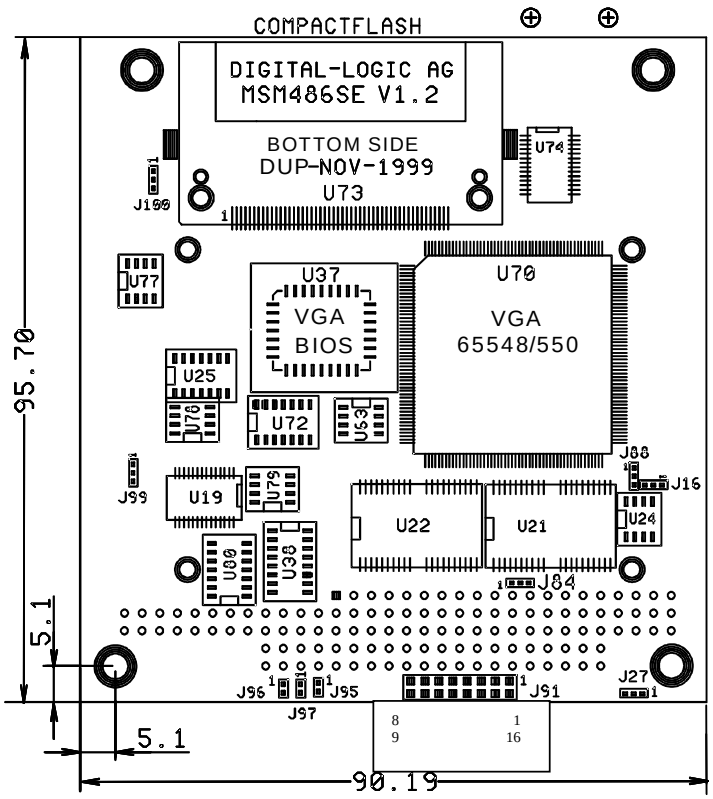




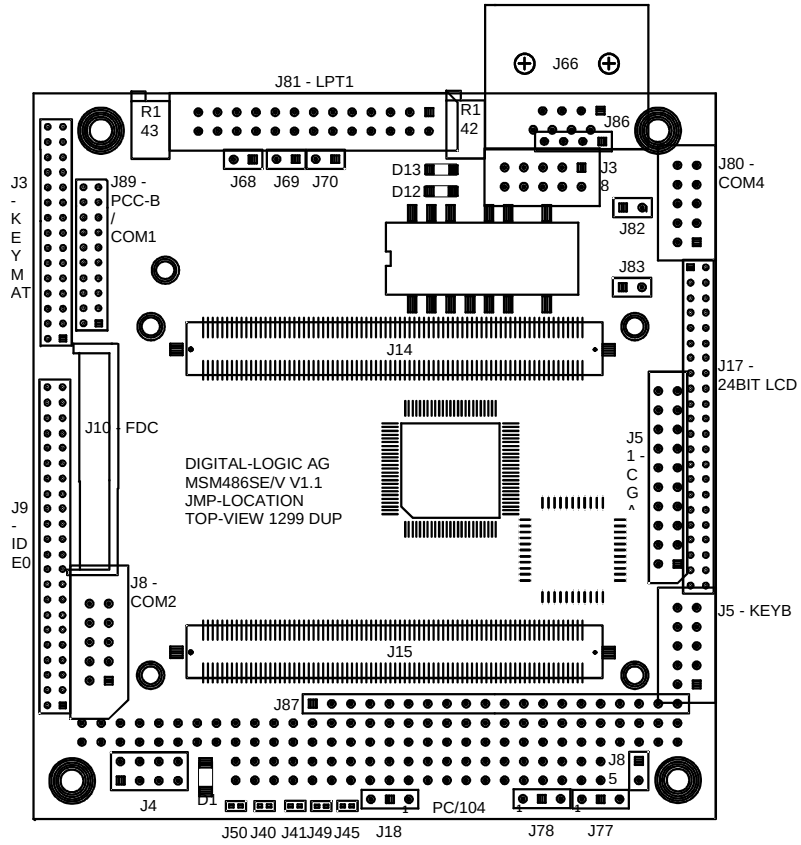
5.1.2 Jumper locations, version 1.2

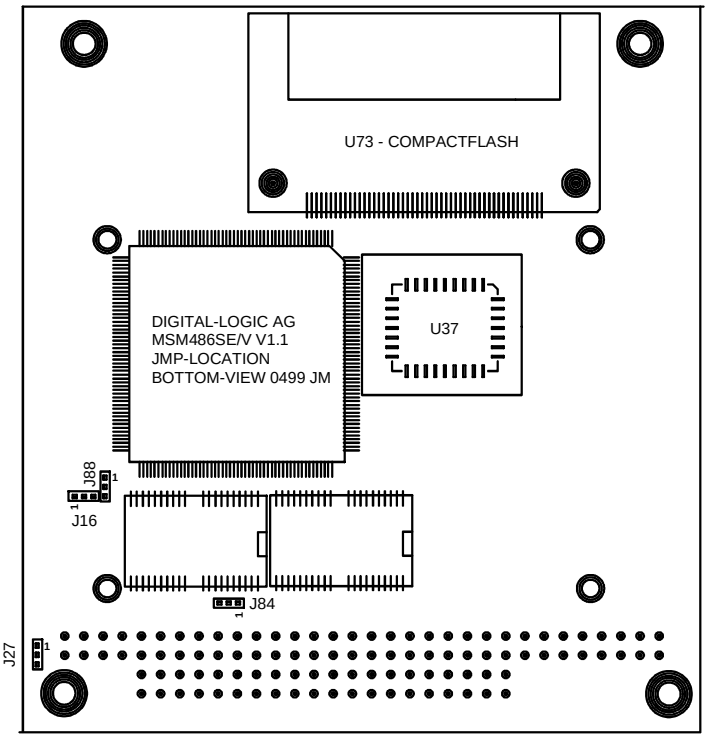


Shine through side (mirrored)



5.1.3 Jumper locations, until version 1.1





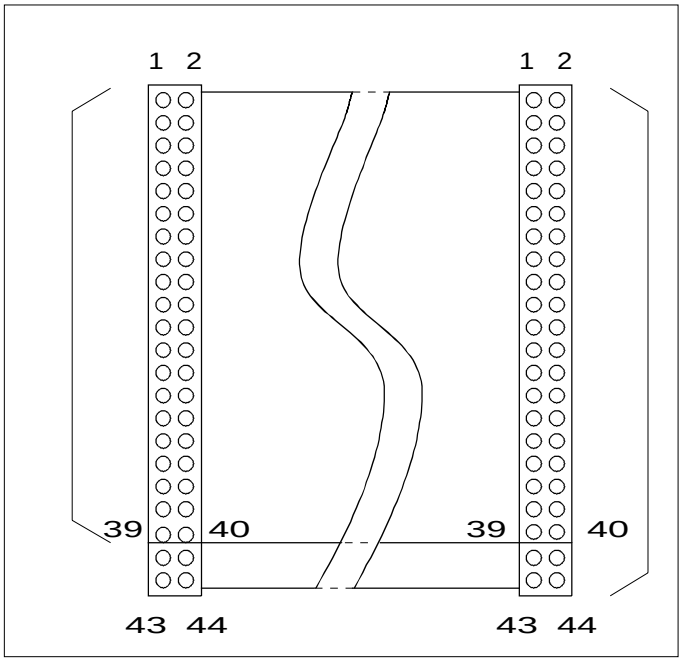
6 LED CRITERIONS:

	LED	Color	Function
	D01	green	Primary HDD
	D12	red	LAN busy
	D13	red	LAN linked
SM486PC			
	CONTROL	red	POD cycle
	POWER	green	3.3V OK
	RUN	green	Power up OK

7 CABLE INTERFACE

7.1 The harddisk cable 44 pins

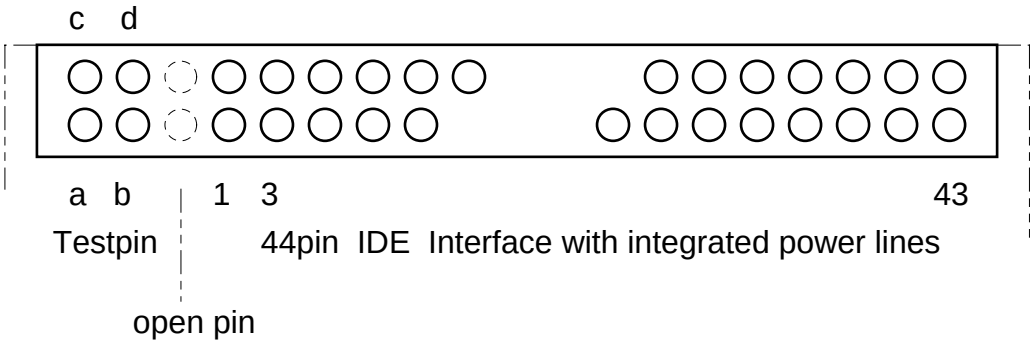
IDT Terminal for Dual Row (2.00 mm grid) and 1.00 mm flat cable. 44 pins = 40 pins signal and 4 pins power.



ATTENTION:

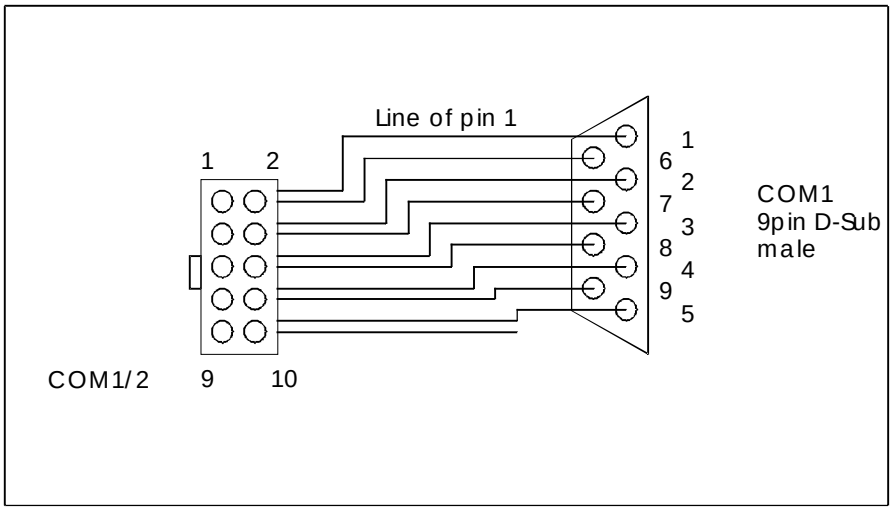
Check the pin 1 marker of the cable and the connector before you power on. Refer to the technical manual of the used drives, because a wrong cable will immediately destroy the drive and/or the MICROSPACE MSM486SE/SEV board. There is no warranty in this case! Without the technical manual you cannot connect this type of drive.

The 44pin IDE connector on the drives are normally composed of the 44 pins and 2 open pins and 4 test pins, 50 pins in total. Leave the 4 test pins unconnected .



7.2 The COM 1/2 serial interface cable

DT terminal for dual row 0.1" (2.54 mm grid) and 1.27 mm flat cable.

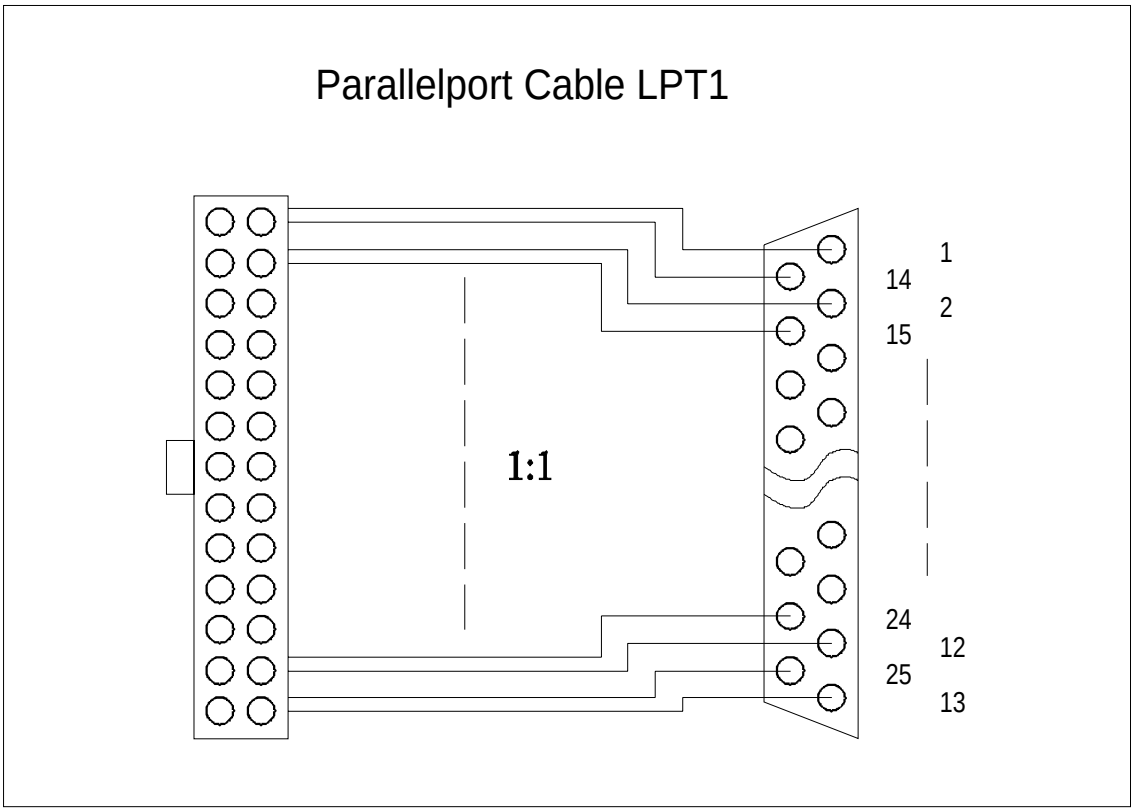


ATTENTION:

- Do not short-circuit these signal lines.
- Never connect any pins either on the same plug or to any other plug on the MICROSPACE MSM486SE/SEV. The +/-10 volts will destroy the MICROSPACE core logic immediately. No warranty in this case!
- Do not overload the output: max. output current converters: 10 mA

7.3 The Printer interface cable (J14)

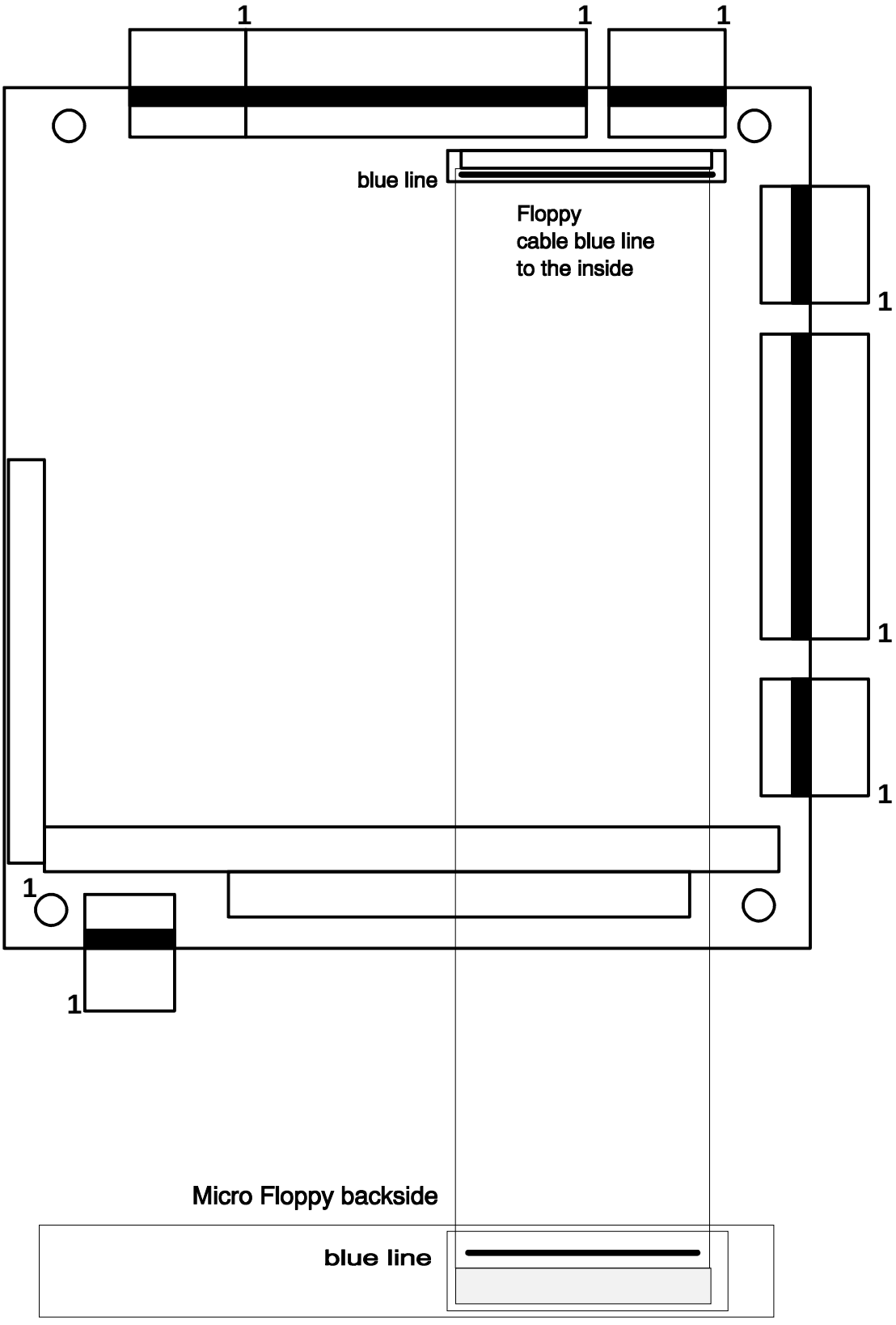
IDT terminal for dual row 0.1" (2.54mm grid) and 1.27 mm flat cable



ATTENTION:

- Maximum length of this cable is 6 meters.
- Prevent short-circuits.
- Never apply power to these signals, the MICROSPACE MSM486SE/SEV will be destroyed.

7.4 The Micro Floppy interface cable



7.5 Redirection FDD to LPT

Adaptercable LPT to FDD, RM2.54mm:

FDD to LPT redirect, since BIOS version 2.22

Signal	LPT D-SUB connector 25 male		FDD 2x15, RM2.54 female	Signal
Strobe	Pin 01		Pin 12	Drive select B
Data 0	Pin 02		Pin 8 *	Index
Data 1	Pin 03		Pin 26 *	Track 0
Data 2	Pin 04		Pin 28 *	Write protect
Data 3	Pin 05		Pin 30 *	Read Data
Data 4	Pin 06		Pin 34 *	Disk change
Data 5	Pin 07		nc	
Data 6	Pin 08		Pin 16	Motor enable B
Data 7	Pin 09		nc	
Acknowledge	Pin 10		nc	
Busy	Pin 11		nc	
Paper End	Pin 12		Pin 22	Write Data
Select	Pin 13		Pin 24	Floppy Write Enable
Autofeed	Pin 14		nc	
Error	Pin 15		Pin 32	Head select
Initialize	Pin 16		Pin 18	Direction
Select In	Pin 17		Pin 20	Step
GND	Pin 18		nc	
GND	Pin 19		nc	
GND	Pin 20		nc	
GND	Pin 21		nc	
GND	Pin 22		nc	
GND	Pin 23		Pin 7	GND
GND	Pin 24		Pin 19	GND
GND	Pin 25		Pin 25	GND
			Every odd pins	GND

* Signals have to be tight to Vcc via 15kΩ

8 SOFTWARE

8.1 The Software Compatibility of the MICROSPACE PC

The CPU AMD ELAN400 is fully compatible with other PC-standard CPUs. The chipsets SCATsx and CS4041 from Chips & Technologies are also fully PC-compatible. No incompatibilities are known.

Tested Software:

MICROSOFT:	DOS	V3.3, V4.x , V5.0, V6.2, V6.22
MICROSOFT:	WIN	V2.x , V3.0, V3.1, Win95
NOVELL:	PALMDOS Novell DOS 7.0	V1.0
INTEL:	RMX	
Software Haus:	MOSY	
Software Haus:	RTX-DOS	
Quantum:	QNX	V3.0, V4.0
NOVELL:	NW-Lite	V1.x
NOVELL:	NETx	V3.11
NOVELL:	ODI services	V1.x and V2.x
IBM:	OS/2	V2.0, Wap 3.0

Application Programs: All software which DIGITAL-LOGIC AG has tested so far are compatible and run without problems.

8.2 HD harddisk driver

This driver is installed as TSR and controls the power saving features of the harddisk. After a specified time, the driver sends an E0h command to the harddisk, to stop the spindle motor and the control electronics. With the next harddisk operation, the system starts up automatically.

HD.COM: 486 CPUs with 16 Bit I/O mapped access

8.3 Operating Systems, Installation-Informations

8.3.1 DOS V6.22

All versions of DOS passed through the OS-test, without reporting an error..

8.3.2 WIN-CE

Version 2.0 and V2.1 are tested and some drivers are available from DIGITAL-LOGIC.

8.3.3 WIN95

WIN95 needs a minimum of 16Mbyte DRAM to work.

8.3.4 WIN98

WIN98 is not working, since the ELAN400 has no coprocessor integrated.

8.3.5 NT4

With a minimum of 16Mbyte the NT4 is working. The performance would be slow, since the ELAN400 is clocked with 66Mhz.

8.3.6 QNX

With the new BIOS V2.xx the QNX is tested and working.

8.3.7 LINUX

The KERNEL must be patched to work properly, since the A20 Gate is not fully compatible to a standard PC/AT model.

Example to modifiy the KERNEL Version 2.0.35

In the i386/boot/setup.s and i386/boot/setup.s file add the lines:

Include the following bold lines in each file, before the call empty_8042 is made:

```
in al, #0x92                ! insert this line
or al, #2                   ! insert this line
and al, #0xFE               ! insert this line
out #0x92,al                ! insert this line

call  empty_8042            ! now follows the original code
mov al,#0xD1
out  #0x64,al
call empty_8042
mov  al, #0xDF              ! A20 on
out  #0x60,al
call empty_8042
```

This patch is available from AMD Errata23 editor: Michael Migdol (AMD)

Correct the SANDISK drive parameters

For operating with a SANDISK under LINUX, you have to modify the linux loader configuration “lilo.conf”. (Otherwise the SANDISK is reported as a 100Mb drive !)

In the file: /etc/lilo.conf add the following line:

append = “hda = 763,8,32”

Some additional hints and feedback from our customers:

- Enable floating point emulation in Linux
- Patch the kernel to solve the A20 addressing
- Look the parameters for the Harddisk geometry in Linux, since the automatic detection of harddisk geometry gets a wrong value return from the hardware
- The SYSTEMSOFT- BIOS which is on every ELAN400- based boards supports a maximum of 8192 cylinders (ca 3GB !)
- It crashes in file linux/mm/slab.c in kmem_cache_grow() and it seems like it's caused by linux using more than the available 16MB of memory.
  Adding mem=16m as a boot option solved the problem.

9 SPECIAL PERIPHERALS, CONFIGURATION

9.1 Special Peripherals

9.1.1 Multifunction Latch

The multifunction latch allows control over several functions such as powerdown, contrast and watchdog. This latch can be controlled by the application program.

Latch 1: Programming address is: 0A8h indexed through port 22h
data R/W with port 23h

Board Version 1.21

MSM486SE/SEV:	Function:	Bit Number:
LCDWR	LSB	0
LCDCD		1
LCDE		2
EEPROM Data input	from EEPROM	3
EEPROM Data output	to EEPROM and WDOG enable	4
EEPROM Shift Clock		5
EEPROM Chipselect 0=normal, 1=selected		6
SLEEP function 1=Sleep, 0=Normal run	MSB	7

Board Version 2.xx

MSM486SE/SEV:	Function:	Bit Number:
LCDWR	LSB	0
LCDCD		1
LCDE		2
SLEEP function	1=Sleep, 0=running	3
EEPROM Data output	to EEPROM and WDOG enable	4
EEPROM Shift Clock		5
EEPROM Chipselect 0=normal, 1=selected		6
free / WPB	MSB	7

The EEPROM Data input is connected to the RING of the internal UART of the ELAN400 chip.

9.2 The Special Function Interface (SFI)

All functions are performed by starting the SW-interrupt 15h with the following arguments:

9.2.1 EEPROM Functions:

Function:	**WRITE TO EEPROM**
Number:	E0h
Description:	Writes the data byte into the addressed memory cell of the serial EEPROM. The old value is automatically deleted and the new databyte will be stored.
Input Values:	AH = E0h Function Request AL = Data byte to store into the EEPROM BX = Address of the EEPROM (0 - 1023 possible)
Output Values:	none; all registers are preserved.

Function:	**READ FROM EEPROM**
Number:	E1h
Description:	Reads the data byte from the addressed memory cell of the serial EEPROM.
Input Values:	AH = E1h Function Request BX = Address of the EEPROM (0 - 1023 possible)
Output Values:	AL = Read databyte

Function:	**READ SERIAL NUMBER**
Number:	E3h
Description:	Reads out the serial number of the board out of the EEPROM.
Input Values:	AH = E3h Function Request
Output Values:	DX, CX, BX = serial number binary (not ascii!)

Function:	**READ PRODUCTION DATE**
Number:	E5h
Description:	Reads out the production date of the board out of the EEPROM.
Input Values:	AH = E5h Function Request
Output Values:	BX, CX = BX = year, CH = month, CI = day

Function:	**TRANSFER CHARACTER FROM KEYMATRIX TO EEPROM**
Number:	E6h
Description:	Writes the data byte into the addressed memory cell of the keymatrix area of the serial EEPROM.
Input Values:	AH = E6h Function Request AL = Data byte to store into the EEPROM BX = Address of the keymatrix in the EEPROM (0 - 512 possible)
Output Values:	none; all registers are preserved.

Function:	**TRANSFER KEYMATRIX TO EEPROM**
Number:	E7h
Description:	Writes the keymatrix bytes from the 80C42PD into the EEPROM at the defined address.
Input Values:	AH = E7h Function Request
Output Values:	none

Function:	**READ INFO2**
Number:	E9h
Description:	Reads out the information bytes out of the EEPROM from the defined address.
Input Values:	AH = E9h Function Request
Output Values:	AL = Board Type (M= PC/104, E= Euro, W= MSWS, S= Slot, C= Custom) DI = CPU Type (1= ELAN400, 2= ELAN400, 3= 486SLC, 4= DX, 5= P5) BH, BL= Board version (Ex.: V1.5 -> BH= 1, BL= 5) CH, CL= BIOS version (Ex.: V3.0 -> CH= 3, CL= 0) DH = Number of 512k Flash (0= none, 1= 0,5M, 2= 1M, 3= 1.5M etc.) DL = Number of 512k SRAM (0= none, 1= 0,5M etc.)

Function:	**READ INFO3**
Number:	EAh
Description:	Reads out the information bytes out of the EEPROM from the defined address.
Input Values:	AH = EAh Function Request
Output Values:	AX = number of boot errors BX = number of setup entries CX = number of low battery errors DX = number of power on starts

9.2.2 Watchdog functions:

Function:	**WATCHDOG**		
Number:	EBh		
Description:	Enables, strobes and disables the watchdog. After power-up, the watchdog is always disabled as the BIOS does not send strobes to the watchdog. Once the watchdog has been enabled, the user application must perform a strobe at least every 800ms, otherwise the watchdog performs a hardware reset.		
Input Values:	AH =	EBh Function Request	
	AL =	Disable / Enable and strobe of the watchdog	00 = Disable 01 - FE = Enable FF = send strobe
Output Values:	none		

9.3 The Flashdisk

Operating Systems:

MS-DOS, (DL-DOS, ROM-DOS only doen't work with the formating programm), RTX-DOS, WIN 3.11, ROM-WIN are working with these drives.
All other non DOS compatible systems need a driver.

9.3.1.1 Enabling and Formatting the Flashdisk Module

Enabling:

After installing the flashdisk module, enter the BIOS-Setup. In the BIOS-Setup enabled by selecting DL-FFS.

Format:

- copy onto a bootable MS-DOS floppy disk the file e.g
- "`CD:\xxx\ffs_v600\dlffsfmt.exe`"
  and the MS-DOS programm "`sys.com`"
- boot from this floppy disk
- start the flash format program `dlffsfmt c:`
  If not accessible try the following: `dlffsfmt 80` (only needed at the first format cycle)

  -> The screen should inform about the status of the flashdisk.

- transfer the bootfiles to the flash with the command `sys a: c:`

- From this moment, the flashdisk is now the bootable drive `c:` and if any harddisks are connected, the drive letter changes form `c: -> d:` and form `d: -> e:`

Please note, that each FFS version needs a specific tool:

If this is version 5.xx (first boot), then one has to use	DLFMT.EXE
If this is version 6.xx (first boot), then one has to use	DLFFSFMT.EXE
If this is version 7.xx (last boot), then one shall use	FXFMT.EXE

9.4 The Powersave Functions

9.4.1 Hardware controlled suspend/resume function

The suspend input pin may be used to suspend the system or to resume the system. The input is TTL compatible. A falling pulse generates a suspend. The second puls generates a resume of the system. The next one will again generate a suspend, and so on. This input is an internal pullup with 10k Ohms. The external circuit must be glitch free.

9.4.2 Software controlled Power Management functions

1. Step: Enable the Power Manager by connecting the AC-Sense input (J30 Pin3) to Ground.
The open AC-Sense input is onboard pulled-up, so the Power Manager is disabled.

2. Step: Enter the BIOS-Setup an select the Power Management options.
Click on the general Power Manger to enable the function.
Enter the timeout for the different powermodes.

3. Step: Restart the system

9.4.3 The different Powermodes

PM-Mode	CPU-Clock	VGA-Status	HD-Status	Keyboard	Wakeup activity	Powerdown
Hyperspeed	66MHz	working	working	working	Software	Timeout
Highspeed	33MHz	working	working	working	Software	Timeout
Lowspeed	1MHz	sleep	sleep	working	KBD	Timeout
Sleep	DC	sleep	sleep	sleep	resume-pin	Timeout
Suspend	DC	sleep	sleep	sleep	resume-pin	

9.4.4 The 99Mhz CPU Speed Selection

Go into the BIOS-Setup, select the power settings and for Hyperspeed, Highspeed and Lowspeed the CPU frequency may be selected. The Hyperspeed is the maximum performance mode. The default value is 66Mhz, may be altered to 33Mhz and 99Mhz.

If the system is assembled with a ELAN400-99Mhz, the Hyperspeed selection in the BIOS-Setup may be tuned to 99Mhz.

Warning:
On 66Mhz systems, enabling the 99Mhz speed may overheat the CPU. No waranty is given to this case.

9.4.5 Software controlled Powermode Switching

The powermodes may also be switched with software commands directly from the application.

To program an ELAN register use the following assembly code:

```
mov al,80h    Register index 80h
out  22h,al
mov al,04h    Low speed
out  23h,al
```

or with 16Bit access:

```
mov ax,8004h
out 22h,ax
```

The register 80h in the ELAN chipset selects the clock of the Highspeed mode

Reg.B1h	D7	D6	D5	D4	D3	D2	D1	D0
Bit:	HYS-Clk	HS-Clk1	HS-Clk0	LS-Clk1	LS-Clk0	PresentClk2	PresentClk1	PresentClk0
Bit 0	66MHz							
Bit 1	99MHz							

HS-Clk:
00 = 8.29MHz
01 = 16.6MHz
10 = 33.2MHz
11 = reserved

LS-Clk:
00 = 8.29MHz
01 = 4.15MHz
10 = 2.07MHz
11 = 1.04MHz

The register 40h in the ELAN chipset selects the clock of the Highspeed mode

Reg.B1h	D7	D6	D5	D4	D3	D2	D1	D0
Bit:	LS_Timer	HYPER	SB_LCD	Fast-Timer	HS_Time	PMUMode2	PMUMode1	PMUMode0
Bit 0	normal	Highspeed	normal	slow Timeout	read only			
Bit 1	not used	Hyperspeed	not used	fast Timeout				

PMU-Mode: 000 = HighSpeed, 001 = HyperSpeed, 010 = LowSpeed, 011 = Reserved, 100 = Standby

For further information about the ELAN Power Management refer to the programmer's reference manual.

10 BIOS SETUP FOR THE DIFFERENT POWERMODES

10.1 Power Management

The ACIN feature can be configured to disable most mode transitions and allow the system to run at maximum performance. Suspend mode can still be accessed.

Each mode has a timer that, when it times-out, will signal the MPU to step down to the next lower mode (see the flow chart for the mode sequence). Each timer can be disabled to allow the PMU to remain in any mode, except for the Temporary Low-Speed timer.

Temporary Low-Speed is a temporary mode. Its timer has a default minimum so that the PMU will return to the necessary mode soon after the event that sent it to Temporary Low-Speed mode is done. This timer is different than the other mode timers in that respect. The timer has no disable. It does not cause the PMU to stay in Temporary Low-Speed mode; it will always continue on to another mode. This timer allows a system to be designed that only goes as low as Standby mode but will still service secondary activities and return to the low power Standby mode (rather than getting stuck in Temporary Low-Speed mode).

10.1.1 Hyper-Speed Mode

This mode is used when performance is much more valuable than battery life. It utilizes the clock multiplying ability of the CPU to run the CPU at 66MHz (2x). When enabled, Hyper-Speed mode will be entered from High-Speed mode. Any event that takes the chip to High-Speed mode will take it there while the CPU's PLL is brought up, then it will switch to Hyper-Speed mode. The exception to this is when the Hyper-Speed mode until a primary activity or ACIN forces a change back to Hyper-Speed mode (or when the High-Speed timer times out it will drop to Low-Speed mode). While the CPU's PLL is brought up the CPU cannot be operating in static clock mode. It will be put into the Stop Grant state until the PLL stabilizes. Going to High-Speed mode while this is done allows any other device needs time to stabilize (e.g., power on an external device can be turned on by a GPIO).

This mode defaults to disabled. A bit has to be set to enable the system to transition to Hyper-Speed mode.

Actions taken on the ELANSC400 microcontroller during Hyper-Speed mode:

- All parts of the system are clocked at full speed
  - Because this mode will use the CPU's PLL, there is a delay in changing the CPU frequency (a CPU STOP GRANT must be done using the CPU's STOP CLOCK INTERRUPT) so quick clock switching will not be allowed.
  - Auto Shutdown is available in this mode. When enabled, this feature slows down the CPU clock at a programmed interval for a programmed amount of time. The Auto Shutdown feature affects system performance, since it arbitrarily slows down the CPU clock occasionally. However, power is saved.
  - The cores that manage their own power may stop their clocks in all modes, including this one.
- The CPU clock is programmable to be 66MHz. The bus clock will still be 33MHz.
- The GPIO pins (if programmed) will change to the Hyper-Speed value.

The system enters Hyper-Speed mode when Hyper-Speed Mode is enabled and:

- The system goes to High-Speed mode
  - After a 1ms delay (for the CPU PLL to start up and stabilize, the system goes to Hyper-Speed mode unless High-Speed mode was entered by the Hyper-Speed timer time-out. Note, if Hyper-Speed is enabled, and a write to the PMU Force Mode Register (CSC index 40h) forced High-Speed, the system will go to Hyper-Speed.
- The system is in High-Speed mode and a primary activity happens.
  - The same PLL start up time restrictions apply.

The system leaves Hyper-Speed mode when:

- The Hyper-Speed mode timer times out
  - Drops to High-Speed mode

- Programmed directly out with the PMU Force Mode Register
  - Can go to any other mode
- The SUS_RES signal toggles
  - If enabled, forces the system directly to Suspend mode
- BL0, BL1 or BL2 go Low (programmable option)
  - BL2 causes a mode change to Critical Suspend mode if enabled and ACIN is not active
  - BL0 or BL1 causes a mode change to Low-Speed mode of High-Speed mode-8MHz if enabled and ACIN is not active

10.1.2 High-Speed Mode

This mode is used when performance is more valuable than battery life. High-Speed mode does not use the CPU PLL for operation. It drives the CPU in static clock mode.

High-Speed mode can be disabled by the enabling and assertion of the Battery Low inputs. When this occurs, activities that normally caused a mode change to High-Speed will go to Low-Speed instead. Only the PMU Force Mode Register (CSC index 40h) will allow access to High-Speed mode.

Actions taken in the chip during High-Speed mode:

- All parts of the system are clocked at full speed
  - Auto Shutdown is available in this mode. When enabled, this feature slows down the CPU clock at a programmed interval for a programmed amount of time. The Auto Shutdown feature affects system performance, since it arbitrarily slows down the CPU clock occasionally. However, power is saved.
  - The cores that manage their own power may stop their clocks in all modes, including this one.
- The CPU clock is programmable to be 33MHz, 16MHz, or 8MHz.
- The GPIO pins may change state if enabled for PMU State Change Output (programmable option).

The system goes to High-Speed mode when High-Speed Mode is enabled and:

- Hardware reset (the CPU clock will default to 8MHz)
- Hyper-Speed mode timer times out
- A primary activity is detected
- Resume or wake-up from Suspend
- Programmed directly to it with the PMU Force Mode Register
- ACIN is enabled and the ACIN signal goes active or if a bit in the Battery/AC Pin Configuration Register (CSC index 70h[5]) is set.

The system leaves High-Speed mode when:

- A primary activity occurs and Hyper-Speed mode is enabled:
  - Goes to Hyper-Speed mode after the CPU PLL is stable
- The High-Speed mode timer times out
  - Drops to Low-Speed mode
- Programmed directly out with the PMU Force Mode Register
  - Can go to any other mode
- The SUS_RES signal toggles
  - If enabled, forces the system directly to Suspend mode
- BL0, BL1 or BL2 are asserted (programmable option)
  - BL2 causes a mode change to Critical Suspend mode if enabled and ACIN is not active
  - BL0 or BL1 causes a mode change to Low-Speed mode if enabled and ACIN is not active

High-Speed Mode
This mode is used when there is not a lot of CPU intensive activity in the chip and the CPU clock can be slowed down for all CPU cycles.

Actions taken in the chip during Low-Speed mode:

- The CPU clock can be programmed to 8MHz, 4MHz, 2MHz, or 1MHz in this mode.
- The CPU clock is driven with the programmed frequency for all cycles.
  - The CPU clock does not speed up to 8MHz for ISA, PC Card, or ROM cycles. They happen at the programmed CPU clock speed. For example, if the CPU clock is programmed to 2MHz, the ISA cycles will be approx. 4 times as long as normal, since the ISA clock is also 2MHz.

The system goes to Low-Speed mode when:

- The High-Speed mode timer times out
- Programmed directly to it with the PMU Force Mode Register
- BL0 or BL1 goes Low (programmable option)
- A primary activity happens and the High-Speed mode is disabled
- Resume or wake-up from Suspend mode when the High-Speed mode is disabled
- When a secondary activity happens in Low-Speed mode the timer may be reset (programming option), modes are not changed.

The system leaves Low-Speed mode when:

- The Low-Speed mode timer times out
  - Drops to Standby mode
- Programmed directly out with the PMU Force Mode Register
  - Can go to any other mode
- A primary activity is detected and High-Speed mode is enabled
  - Goes back up to High-Speed mode
- BL2 goes Low (programmable option)
  - Cause a mode change to Critical Suspend mode if enabled and ACIN is not active
- The SUS_RES signal toggles
  - If enabled, forces the system to Suspend mode

10.1.3 Standby Mode

This mode is used when there is no activity in the chip and many clocks can be shut down. When an enabled activity occurs, the PMU switches to the appropriate mode.

Actions taken in the chip during Standby mode:

- The CPU clock is stopped
- The internal LCD controller can be programmed to be enabled or disabled.

The system goes to Standby mode when:

- The Low-Speed mode timer times out
- Programmed directly to it with the PMU Force Mode Register
- Return from Temporary Low-Speed mode
  - When Temporary Low-Speed mode was entered from Standby mode.

The system leaves Standby mode when:

- The Standby mode timer times out
  - Drops to Suspend mode
- A primary activity is detected
  - Goes back up to High-Speed or Low-Speed mode
- A secondary activity is detected
  - Goes to Temporary Low-Speed mode
  - The Standby timer is paused while in Temporary Low-Speed mode. The count-down continues when Standby mode is re-entered. The timer is not reset by this mode change.
- BL2 goes Low (programmable option)

  - Causes a mode change to Critical Suspend mode if ACIN is not active
- The SUS_RES signal toggles
  - If enabled, forces the system to Suspend mode

Suspend Mode

Suspend mode is used when the system wants to enter a very low power mode, keeping the DRAM refreshed and saving the system configuration so it can return to the point it left off. If the system PLLs (the High-Speed, Low-Speed, Intermediate, and Graphics Dot Clock PLLs) are shut off, it will take longer to return, but more power will be saved.

Actions taken in the chip during Suspend mode:

- All clocks are stopped (except the RTC and memory refresh, which are derived off the 32KHz oscillator without the system PLL's involvement)
- PLLs shut down (programmable option)
- The RTC is left running
- The DRAM refresh (either CAS before RAS or self refresh) is programmable to be left active or turned off.
  - If CAS-before-RAS refresh is left active, the refresh clock will be switched from the timer counter to the 32KHz oscillator.
- The pins on the chip will go to a predetermined mode and stop toggling

The system goes to Suspend mode when:

- The Standby mode timer times out
- Programmed directly to it with the PMU Force Mode Register
- Critical Suspend mode is unlocked by ACIN, BL2, and/or BL1 and BL2
- The SUS_RES signal is enabled and changes

The system leaves Suspend mode when:

- The Suspend mode timer times out
  - If an SMI/NMI is enabled it goes to Temporary Low-Speed mode. The PLLs have to be started back up.
  - If enabled as a wake-up, goes to High or Low-Speed mode.
  - If both the SMI/NMI and wake-up are enabled, it goes to High or Low-Speed mode before servicing the SMI/NMI.
- BL2 is enabled and asserted
  - Goes to Critical Suspend mode
- A wake-up source is detected active
  - The PLLs have to be started back up
- The SUS_RES signal toggles
  - If enabled, forces the system to High-Speed or Low-Speed mode. The PLLs have to be started back up.

10.1.4 Critical Suspend Mode

Critical Suspend mode is used when a battery-dead indication comes in on BL2 and the chip must quickly go to a Suspend mode and stay there until an unlock event happens. It can then go to Suspend mode to wait for a wake-up or wake-up if the unlock event is also enabled as a wake-up.

The unlock sequences that can be enabled are:

- ACIN is toggled
- BL2 goes inactive
- BL2 and BL1 go inactive

Actions taken in the chip during Critical Suspend mode:

- Same as Suspend mode

- The system is locked in Critical Suspend mode as long as an unlock sequence does not occur.
- The LCD panel shutdown sequence is accelerated. The voltage and control signals to the panel will disable without regard to normal power down sequencing.
- PLLs shut down.

The system goes to Critical Suspend mode when:

- BL2 goes Low (programmable option) and ACIN is not enabled or asserted.

The system leaves Critical Suspend mode when:

- An unlock happens
  - Goes to Suspend mode
- An unlock happens that is enabled as a wake-up
  - Forces the system to High-Speed or Low-Speed mode

10.1.5 Temporary Low-Speed Mode

This is a temporary mode to service a secondary activity or an SMI/NMI from Standby mode and then return to Standby mode, or to service the Suspend timer time-out SMI/NMI and then return to Suspend mode. Secondary activities do not need extensive CPU time to service, so the PMU does not return to High-Speed mode for them. Instead, Temporary Low-Speed mode acts as a temporary low-speed mode that has its own timer and returns to Standby mode if it was entered by a secondary activity. A secondary activity that is received while in Temporary Low-Speed mode will cause the system to restart the Temporary Low-Speed timer.

When Temporary Low-Speed mode is entered from Standby mode, the Standby timer will be paused while in Temporary Low-Speed mode. The Standby timer will then resume its count-down when that mode is returned to.

When Temporary Low-Speed mode is entered from Suspend, it acts like Suspend in that only a wake-up can change the mode to High- or Low-Speed (activities and ACIN will not change the mode). A force mode write, however, can change to any other mode.

Actions taken in the chip during Temporary Low-Speed mode:

- CPU clock goes to the programmed Low-Speed mode rate
- All other clocks go to the appropriate Low-Speed mode rate
  - Except for LCD graphics, if it was disabled in the mode the PMU is coming from.
- Temporary Low-Speed mode timer is started
  - The system goes to Temporary Low-Speed mode when:
- A secondary activity is received in Standby mode
  - If a secondary activity happens in Temporary Low-Speed mode, the timer is restarted
- An SMI/NMI is triggered while the ... in Standby, or by a Suspend timer time-out.
  - SMI/NMI in Suspend will not cause the PMU to go to Temporary Low-Speed unless it is caused by the Suspend timer time-out. The system will go to Temporary Low-Speed mode to service the SMI/NMI. During the interrupt service routine the PMU Force Mode Register can be used to change the PMU mode to any other mode rather than letting the system go back to Suspend or Standby mode.
- Programmed directly to it with the PMU Force Mode Register

The system leaves Temporary Low-Speed mode when:

- Temporary Low-Speed mode timer times out
  - If the last mode was Standby will return to Standby. Otherwise will go to Suspend.
- Programmed directly out with the PMU Force Mode Register
  - Can go to any other mode
- The SUS_RES signal toggles
  - If enabled and Temporary Low-Speed mode was called from Standby mode, forces the system to Suspend mode

  - If enabled and Temporary Low-Speed was entered as an SMI/NMI service routine from Suspend mode, the SUS_RES signal causes a transition to High-Speed (or Low-Speed) mode
- A primary activity is detected
  - Goes back up to High-Speed mode
- BL2 goes Low (programmable option)
  - BL2 causes a mode change to Critical Suspend mode if ACIN is not enabled and asserted
- Wake-up detected
  - If Temporary Low-Speed mode was entered from Suspend, wake-ups can be detected and cause a mode change to High-Speed or Low-Speed mode.

10.1.6 Mappable GPIO_PMUA-GPIO_PMUD Signals

Up to four GPIO_CS pins can be programmed to inform external hardware of internal PMU states. The internal signal names associated with this information are PMUA, PMUB, PMUC, and PMUD. Each of these signals has a register, GPIO_PMUx Mode Change Register (CSC index AA-ADh), that defines its value during every distinct PMU state, and each of these signals has a 4-bit field in the GPIO_PMU to GPIO_CS Map Registers A and B (CSC index AE-AFh) that defines which, if any, GPIO_CS pin it drives. As noted above, the GPIO's output bit in CSC indexed registers A0 – A5h must be 1 to set the pin to output mode, and the GPIO's I/O bit in CSC indexed registers A6-A9h must be 0 to allow the PMU signal to propagate.

ACIN Detect and Battery Low
Four signals are brought into the chip from outside, so that the state of the system power can be reported to the chip and used in the power management scheme. The Alternating Current INput (ACIN) signal is meant as an indication that the system is connected to a greater source of power (such as an AC wall plug) and power savings are no longer as important as performance. The Battery Low (BL0, BL1, and BL2) signals are digital inputs that external voltage comparators or an external processor can drive to inform the chip of the state of the charge on the system batteries. Each Battery Low signal can report a different level in the battery discharge. For example, they may be used as follows:

- BL0 – Batteries are getting weak, so slow down the clock in High-Speed Mode
- BL1 – Batteries are weaker, so disable High-Speed mode and limit the PMU to go to Low-Speed mode as the highest mode
- BL2 – The batteries are so low they cannot operate the system. Force the system into Critical Suspend mode and do not allow a resume until there is AC power or the batteries are changed

10.1.7 Critical Suspend Mode

The ACIN signal is used to indicate the system is connected to a permanent source of power (i.e., an AC wall adapter) and power management is not required (Suspend mode is still accessible). There is a register bit in the Battery/AC Pin Configuration Register (CSC index 70h[5]) to perform a software ACIN, which, when set, will cause the same affect as asserting the ACIN pin. This is useful for software to emulate the effect the hardware ACIN line has on the function of the PMU. There is no functional difference between the ACIN pin being active and the ACIN software bit being set. Software can determine which is active by reading the Battery/AC Pin State Register (CSC index 72h[5]).

ACIN is similar to, but different from, a primary activity. Both can take the PMU back up to Hyper- or High-Speed mode, but the ACIN will keep it there by masking the timer time-outs. If the PMU Force Mode Register is programmed while ACIN is active (and programmed to disable PMU functions) the PMU will change mode, but will immediately switch back to High-Speed or Hyper-Speed mode because of ACIN.

Activities, wake-ups, and SMI/NMIs still work when ACIN is active also. SMI/NMIs are still accessible and the system will still wake up from Suspend when ACIN is active.

ACIN can also be used as part of the Critical Suspend unlock scheme.

When the ACIN signal is enabled and active, it will cause the following things to happen in the chip:

- Force the system into High-Speed or Hyper-Speed mode (if Hyper-Speed mode is enabled) if it is in Low-Speed, Temporary Low-Speed, or Standby modes. If it is in Suspend mode, ACIN active will not cause a mode change unless programmed to be a wake-up.
- Force most PMU mode timers time-outs to be ignored by the PMU, so the chip will not time out and change modes. The chip still has a method to go into Suspend mode through a SUS_RES pin toggle or register force. The Suspend mode timer will still be operational when ACIN is active.
- Disable BL0, BL1, or BL2 from causing a mode or clock-speed change, unless it is also programmed as an SMI or wake-up.

The state of the ACIN signal can be read from CSC index 72h[3].

10.1.8 Battery Low

The three Battery Low pins (BL0, BL1, and BL2) are active Low signals that allow the system to monitor the state of the system batteries with up to three different levels. As a result of the battery-low monitoring, the PMU can be programmed to reduce the CPU clock speed in High-Speed mode, disable High-Speed mode and use Low-Speed mode as the fastest mode, or go to Critical Suspend Mode.

All three Battery Low inputs are negative edge-triggered. There is a 60ms debounce time during which all further edges will be ignored. After the debounce time, another edge can be detected. If a Battery Low input changes polarity during the debounce time, and remains changed after 60ms, this other edge will be detected (after the first 60ms debounce time).

These power saving features will happen on the falling edge of the respective Battery Low signal unless ACIN is enabled and active. When ACIN goes inactive any active and enabled battery-low feature will take affect at that time.

The state of all Battery Low signals can be read form the Battery/AC Pin State Register (CSC index 72h[2-0].

10.1.9 CPU Clock Speed Reduction

BL0 and BL1 can be programmed to force the chip to disable Hyper-Speed mode and use 8.29MHz as the High-Speed CPU frequency, or to disable High-Speed mode and force the chip to go to Low-Speed mode as the highest mode. The PMU Force Mode Register does not override the Battery Low feature. If the PMU Force Mode Register is used to force Hyper- or High-Speed, the system will return to Low-Speed or High-Speed due to BL0 and BL1.

10.1.10 Critical Suspend Mode Access

Battery Low 2 is programmable to force the chip to Critical Suspend mode (within 55µs from the falling edge of BL2) and lock the system into this mode to stop it from resuming until it is unlocked. The unlock requires special handling since the BL2 signal may go High again after Suspend mode is reached. Since system current consumption is reduced, the voltage from the battery may rise. When Battery Low 2 is used as the Critical Suspend mode change signal, all wake-up sources can be detected and latched, but they will not cause a wake-up until the system is unlocked. Unlocking the system from Critical Suspend is a programmable function. The system will be locked into Critical Suspend after a BL2 is seen active until one of the following unlock sources happens:

- ACIN is seen active
- BL2 alone is seen inactive
- Both BL2 and BL1 are inactive

These unlock sources do not automatically cause a wake-up (unless programmed as wake-up sources). They only disable the lock circuit so a wake-up source can resume the system.

If the BL2 signal is enabled to cause an SMI/NMI, the system will still enter Critical Suspend in 55µs, and service the SMI/NMI after the system wakes up.

When the system is active and the LCD is displaying data, the BL2 force to Critical Suspend mode will happen without regard to normal LCD power sequencing. The LVEE, LVDD, and LCD signals will not sequence off as they normally do when going to Suspend. They will all go inactive at approximately the same time. This is done so Critical Suspend mode is entered as quickly as possible.

A signal is available on the chip to indicate when the chip is locked into Critical Suspend mode by a BL2; the LBL2 signal (Latched BL2). It will go Low during Critical Suspend mode and will go High again when the PMU leaves Critical Suspend.

A bit is available in the Battery/AC Pin Configuration Register (CSC index 70h[7]) to indicate to the system that it has been in Critical Suspend mode. This feature can be used by SMI/firmware/software to indicate the system has resumed form a Critical Suspend, so that any problems this has caused can be fixed. For example, if a PC Card was being written and was powered down by the Suspend, the card can then be reconfigured and the write continued.

11 REMOTE LINK TO A HOST-PC

The embedded remote control in the BIOS allows you to control your target machine (DIGITAL-LOGIC's embedded PC) from a host computer using a serial AT Null- modem cable. This is accomplished by transferring all INT10h (video) and INT16h (keyboard) requests to the host machine, executing the command there, and finally returning the results back to the target system. The target system seems to behave just like it would use its own VGA card and keyboard. But in fact, it uses the resources of the host computer. Additionally, the target can access the floppy drive of your host PC. This allows you to download software or initially format your flash disk without external boot devices installed.

Attentions:

The remote works with 115kbaud. For optimal access- speed use only host PC's with 16550 compatible UARTS (with FIFO).
In some applications (also in the BIOS setup), if you press the keys on the keyboard too fast, the application can not recognize the pressed keybutton. In this case nothing happens, but you must repeat the key entry.
If the connection is disconnected until the communication is running, the host PC may be asynchronized. In this case, the host PC must be restarted with CTRL + ALT + DEL.

Restrictions:

- Avoid direct writings to video RAM. There is no mechanism to detect and transfer these outputs
- Avoid video calls that uses registers other than AX, BX, CX, DA. To speed up video output, only these registers will be transfered.
- Formatting of remote floppy does not work. Providing this feature would also include the necessity of transferring buffered data from the host machine.
- Do not rely on INT15 function 4F. This function is not available
- KEYB.COM is not needed on the target machine. Instead, the current keyboard handler of your host computer is automatically used.
- Do not press CTRL + ALT + DEL while redirection is active. This will not reboot your target system but your host machine
- Use a faster host- PC as the embedded one, otherwise time out errors may will occure

11.1 What are the functions of the REMOTE- link ?

The remote link is used, to redirect the video, keyboard and the floppy over the serial interface to a host PC. In application, where no monitor or keyboard or floppy are available, the remote may be useful to communicate with the embedded PC.

1. Video redirection:
   The videofunctions from INT10 are redirected to the host-PC. Only black and white modes are redirected, not the CGA and VGA color modes.
   Because the transferspeed of 115kbaud is too slow for such a datastream.

2. Keyboard redirection:
   All keyboard functions of the INT16 are redirected to the host PC. Every key pressed on the host PC is treated like a key on a keyboard which is directly connected to the embedded PC.
   Attention:
   Do not press CTRL + ALT + DEL on the keyboard of the host PC, otherwise the host PC will be restarted

3. Floppy redirection:
   This function allows you to use the floppy of the host PC instead of the floppy, which is connected to the embedded PC. All file functions of the INT13h will be transfered over the serial link to the embedded PC. The transferspeed is defined by the 115kbaud .
   This feature of the **REMOTE** must be enabled in the BIOS setup of the embedded PC.

   Select in the BIOS setup the:

REMOTE FLOPPY	available as drive A, 1.44”
REMOTE	option as enabled.

11.2 Enable and link the REMOTE access

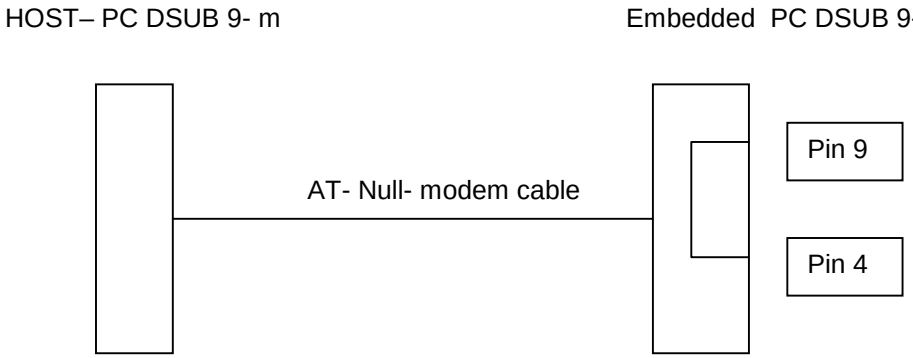
1. Connect the COM1 of the host PC with any serial port of the embedded PC.
   Use a 9pin AT Null- modem cable, where all signals are crossed. To enable the remote, a special connection between pin 9 and pin 4 of connector on the embedded pc side must be done.

2. Start the REMHOST.EXE on the host PC.
   To exit the REMHOST.EXE program, press CTRL + Left- Shift simultaneously.

3. Power up the embedded system.
   After 3- 5 seconds, the screen of the host PC begins to show the embedded PC’s screen.

11.3 Remote enabler

At the embedded side of the remote link cable, the pin 4 must be connected with the pin 9 inside the connector.



In the AT- Null- modem cable, the following signals are crossed:

PC-Host				Embedded PC	
Pin	Criterion			Pin	Criterion
7	(RTS)	⇒		(CD)	1
1	(CD)	⇐		(RTS)	7
3	(TXD)	⇒		(RXD)	2
2	(RXD)	⇐		(TXD)	3
6 + 8	(DSR) (CTS)	⇐ ⇐	—————— bridge	(DTR)	4
4	(DTR)	⇒		(DSR) (CTS)	6 + 8
5	(GND)			(GND)	5
9	(RI)	⇐ **open** **nc**	—————— bridge	(RI)	9

Bridge 4- 9 is the **REMOTE ENABLER**

- The AT Null- modem cable is a standard cable, available in every computer store and may at DIGITAL-LOGIC AG.
- The REMOTE- ENABLER must be modified on the AT Null- modem cable itself.
- DIGITAL-LOGIC AG sells a special REMOTE-ENABLER expander to connect to a normal AT Null- modem cable.

11.4 Remote server REMHOST.EXE

REMHOST.EXE actually executes the commands from the target on your host computer. You can use it for all serial communication ports. By default, it is assumed that the serial communication is on COM1. The following command line options are available:

/H display help text

/P port number (COM1...4)

/L use parallel communication (not supported on the DIGITAL-LOGIC computers)

You can escape REMHOST by pressing LEFT – SHIFT and LEFT- CTRL simultaneously. If this should not work immediately for some reasons, keep these buttons pressed down and power down your target system.

12 BUILDING A SYSTEM

To build up a system based on your board, you should prepare the following equipments:

- A stable power supply of 5V (> 3 ampères), depending on the cpu, memory, etc.
- Assemble CPU with the proper clk- settings and cooling (fan) depending on board.
- If necessary, a 12V power supply for LCD or onboard sound.
- 8 ohm speaker for an executed beep code (if available on the board). One may use a capacity of 1µF connected to VCC depending on the board.
- A micro- floppy disk drive (3,5") with a PC floppy cable (26 pin) or a standard FDD with appropriate cable converter. You need at least one floppy to boot for the first time.
- A harddisk IDE 2,5" or 1,8" with the appropriate cable (44 pin and 2mm grid). Do not use a too long a cable to avoid accessing problem as the IDE controller is may not able to drive the HDD.
- Connect a LCD or a monitor.
- Use an AT-compatible keyboard (5 PC) or (6 PC {PS/2} with an appropriate adapter).
- If desired, connect a mouse to it (COM or PS/2 if usable on the board).
- Connect a battery (Lithium 3V or NiMH 3.6V depending on the board) to store the data in the BIOS.

12.1 Starting up the System

Power-up the system and wait for the BIOS to show the BIOS activity on the screen. The BIOS diagnoses the system and displays the size of the memory being tested. Note: you may can not bypass the memory test depending on the BIOS producer.

CMOS-SETUP

If the CMOS configuration is incorrect, the BIOS tells you to enter the setup screen by pressing a key. Select the correct options with the arrow keys and save them.

	SYSTEM SOFT CO
BIOS setup	CTRL- ALT- S
Change values	ARROWS / SPACE
Jump	TAB
Save	ARROWS
Back / exit	ESC

BEEP CODE:

SYSTEM SOFT CO	
4 short	RAM
3 s / 1 l	RAM test failed
2 s / 1 l / 1 s	BIOS is not shadowed
1 s / 1 l / 2 s	BIOS checksum bad
1 l / 1 s / 1 l / 1 s	CRISIS code / CR bad

12.2 BIOS-Setup

12.2.1 Accessing the SCU

To access the SCU, type <Ctrl ,Alt, S> during system boot up.

The SCU displays the system's current configuration settings. The top of the screen has a menu bar with various items (i.e., Startup, Memory, Disks, etc.). Some menu bar items contain pull down menus which list the various items to configure the system, while others perform specified tasks. For example, the Startup menu contains a pull down menu consisting of such items as setting the time, configuring hard disks, and setting the video display type while the Defaults menu bar item brings resets the power-on default values.

The menu items that appear vary depending upon the OEM's requirements as well as the chipset being used. This chapter details those items that are generally available to all or most chipsets. For information on an item that your chipset supports but is not described here, refer to the supplement document for that chipset.

12.2.2 Working with Menu Bar Items

To access the menu bar items, do one of the following:

- Click on the desired item with the left mouse button
- Use the ← or → keys to highlight the desired item. Then press <Enter>.
- Press the key that corresponds to the menu bar item's highlighted letter. For example, to select the Memory menu, press 'M'.

If the menu bar item has a pull down, it will be displayed, otherwise it will perform the specified action.

12.2.3 Working with Pull Down Menu Items

To access any pull down menu item, do one of the following:

- Click on the item with the left mouse button
- Use the ↑ or ↓ keys to highlight the desired item. Then press <Enter>.
- Press the key that corresponds to the menu bar item's highlighted letter.

Depending upon the menu item selected, one of the following occurs:

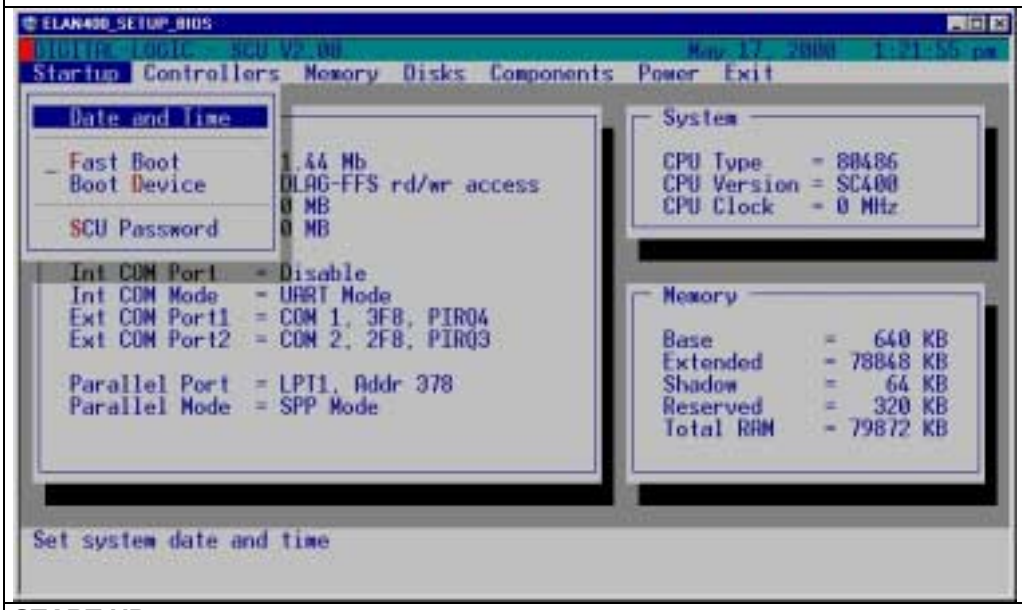
- The selected item is automatically enabled/disabled (acted upon).
- A window appears with a list of options to choose from.
- A window appears prompting the user to supply input.

12.2.4 Closing Pull Down Menus

After modifying any of the pull down menu items, the pull down menu is still displayed. Press <Esc> to close the menu.

Overview of the Digital- Logic– BIOS based on the System Soft BIOS V2.0

Note:
Pictures have been taken on a home PC therefore, RAM- and CPU clock- values are not ELAN typical



START UP

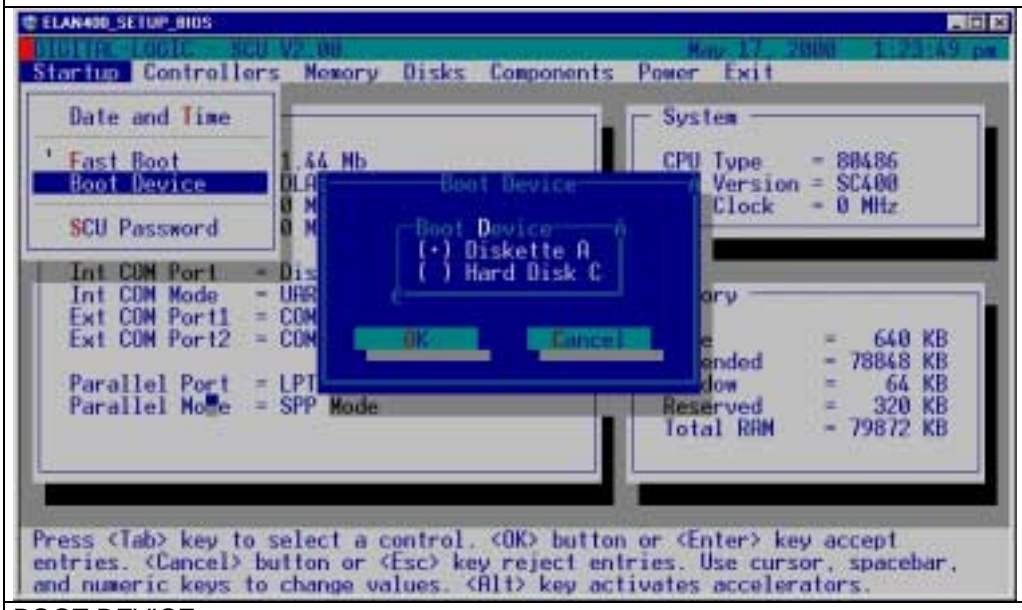
- DATE and TIME
- FAST BOOT
  Initializes and quickly boots the system in a few seconds by bypassing certain diagnostic tests.
  To fully test the system at power-on time, turn off Fast Boot.
  With this choice, the system does a more comprehensive test of the hardware and, therefore, takes longer to initialize and boot.
  A check mark indicates that Fast Boot is enabled.

  enable/disable with ENTER
- BOOT DEVICE
- PASSWORD

DATE and TIME

Provides a single window for setting the system date and time.
The Date and Time window allows the user to specify the *date*, *month*, and *year* as well as the *Hour*, *Minute*, and *Second*.
Please note that the clock is represented in a 24-hour hour format.

- Jump with TAB- key
- Set values with ARROWS or DIGITS
- Save with ENTER



BOOT DEVICE

Specifies where the system will attempt to boot from.
The following options are available from the Boot Device menu:
1. ***Diskette A***
2. **Hard Disk *C***

- Choose device by using the TAB- key
- Select device with the SPACE- key
- Save with ENTER

PASSWORD

This option allows users to create passwords that must be used to gain access to the system at boot time or when resuming from a suspend state.

To select a new password or change an existing one,

1. Choose **Password** from the Startup pull down menu.
2. Enter between 4 and 8 printable alphanumeric characters in the **Enter New Power-On Password** area. Each character entered generates an asterisk ' * ' on the screen.
3. Reenter the password in the **Verify new Password Power-On Password** area.

Once the password has been verified, you must enter the correct password each time you start the system. With an active password, values may can be changed but by leaving the BIOS, the password will be asked to store changes.

- Write the desired password
- Jump with ENTER
- Enable/disable password option by using the SPACE- key
- Save with ENTER

CONTROLLERS

- VIDEO CONTROLLER
- PCMCIA- B CONTROLLER
  enable/disable with ENTER
- REMOTE
  enable/disable with ENTER
- REMOTE FLOPPY
  If enabled, the floppy accesses are redirected to the host computer connected to the remote link cable with a remote enabler.

  enable/disable with ENTER
- PS2-MOUSE PORT
  enable/disable

VIDEO CONTROLLER

There are 4 modes available:

1. **External BIOS**
   uses the onboard BIOS
2. **CGA 80 columes** (Computer Graphics Array)
3. **CGA 40 columes** (Computer Graphics Array)
4. **MDA** (Monochrom Digital Adapter)

- Jump with TAB- key
- Choose option by using the ARROW- keys
- Select device with the SPACE- key
- Save with ENTER

MEMORY

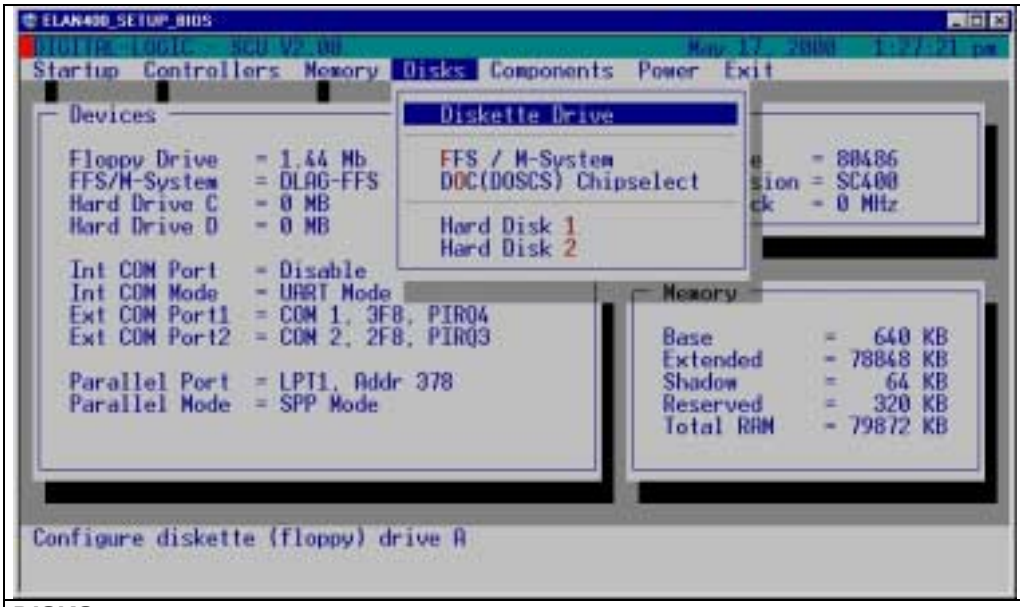
Provides several configuration options for the 486 processor's internal (on-board) cache.
The Cache Systems window contains a section which provides options for selecting the mode for the L1 cache.
This section includes the following choices:

1. ***Disabled***
   When selected, the on-board cache is disabled, The segments included in this box are non-modifiable and are grayed out.
2. ***Write Thru***
   This mode updates both cache and external memory at the same time.
3. ***Write Back***
   This mode does not update cache and external memory at the same time.
   Write operations are first updated in cache memory, followed by external memory.

The **Cache Systems** window allows permits several memory segments to be cached or not cached. These are:

1. ***C Seg* (C000h)** .
2. ***D Seg* (D000h)** .
3. ***Seg* (E000h)** .
4. ***F Seg* (F000h)** .

- Jump with TAB- key
- Choose option by using the ARROW- keys
- Select device with the SPACE- key
- Save with ENTER



DISKS

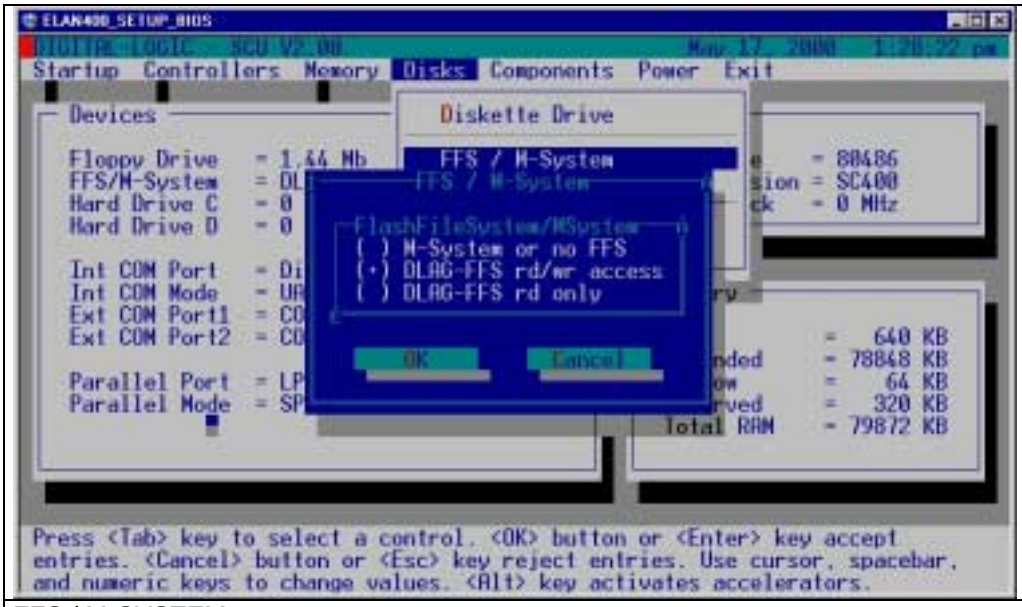
- DISKETTE DRIVE
- FFS / M-SYSTEM
- DOC / CHIPSELECT
- HARDDISK 1
- HARDDISK 2

DISKETTE DRIVE

Specifies a drive type for the diskette drive. Valid configurations are:

1. ***None*** .
2. ***360KB*** .
3. ***720KB*** .
4. ***1.2MB*** .
5. ***1.44MB*** .
6. ***2.88MB*** .

- Jump with TAB- key
- Choose option by using the ARROW- keys
- Select device with the SPACE- key
- Save with ENTER



FFS / M-SYSTEM

Defines the type of installed flashdisks.
The setting are:

1. **None or MSYSTEM modul**;
   MSYSTEM DOC2000 is installed or no flashdisk is installed.
2. **DL-FFS enabled as read/write**;
   MSFFS installed and read / write access allowed
3. **DL-FFS enabled as read only**
   MSFFS installed as read only drive

- Jump with TAB- key
- Choose option by using the ARROW- keys
- Select device with the SPACE- key
- Save with ENTER

DOC / CHIPSELECT

- Jump with TAB- key
- Choose option by using the ARROW- keys
- Select device with the SPACE- key
- Save with ENTER

HARDDISK 1 & 2

- Jump with TAB- key
- Choose option by using the ARROW- keys
- Select device with the SPACE- key
- Save with ENTER

Configures the hard disk drive 1 (DOS drive C:) and 2 (DOS drive D:).
The Hard Disk menu contains the following options:

Custom
Use this option to configure a drive cannot be automatically configured.
When *Custom* is selected, the values for cylinders, heads, sectors per track (SPT), landing zone (LZone), and write precomposition (Precomp) for the drive can be modified.
These values can normally be found in the drive documentation or on the drive label.

Auto-ID
This option will automatically attempt to configure the hard drive parameters for any supported IDE drive.

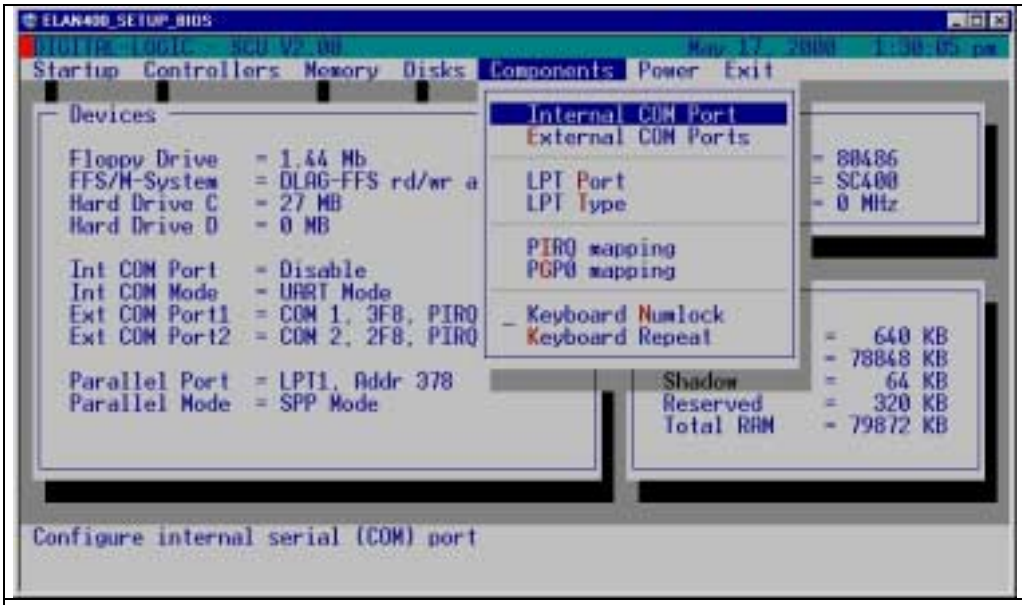
None
Select this option, if no hard drive is installed in the system.

The Enhanced Options group contains items, that may be selected if the drives support them.
These include:

LBA
With the advent of newer, non-fixed disk ATA-compliant mass storage devices such as tape drives, alternate methods for accessing individual blocks of data utilizing a non-sector oriented mechanism are required.
Fixed disk controllers that support LBA addressing allow the calling application (i.e., BIOS, operating system, or device driver) to access an ATA-compliant device as a liner series of blocks.

Multiple Sector Mode
Many newer IDE fixed disks support the capability of reading and writing several sectors via a single command sequence.
Internally, the IDE controller performs multi-sector writes by caching any data that it is unable to write immediately.
Similarly, the IDE controller performs multi-sector reads by caching look-ahead reads during each sector read and buffering the data while the host offloads it the cache's contents into system RAM.

Fast PIO Mode
In the case, where a drive is capable of multi-sector read and write operations, the BIOS records this capability for each drive in the BIOS data area.
When an INT 13H read or write operation occurs, the system BIOS selects the INT 13H code that maximizes the reported capabilities of the drive. Additionally, the system BIOS records the maximum single transfer sector count during its drive capability interrogation process, and selects a sector count that satisfies both the caller's INT 13H read/write request and the capabilities of the target disk.



COMPONENTS

- INTERNAL COM PORT
- EXTERNAL COM PORT
- LPT PORT
- LPT TYPE
- PIRQ MAPPING
- PGPO MAPPING
- KEYBOARD NUM- LOCK
  Specifies whether Num Lock is ON or OFF at system boot time.
  A check mark indicates, that the Num Lock key is ON at system boot.

  enable/disable with ENTER
- KEYBOARD REPEAT

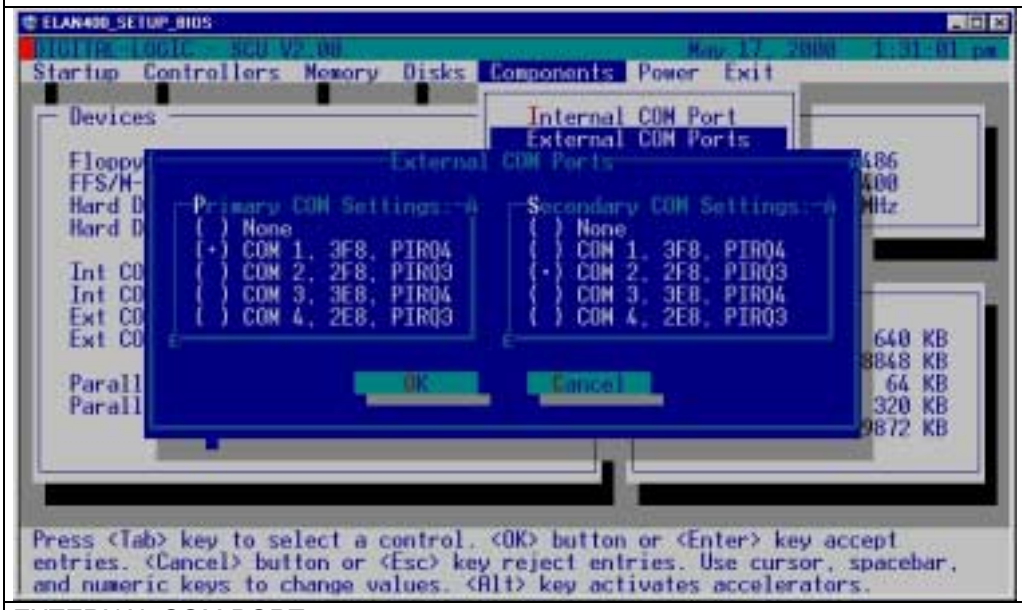
INTERNAL COM PORT

Configures the I/O address for the internal serial (COM) port, ELAN400.
The following settings are available from the COM Port Assignment window:

1. ***Disable***
2. ***COM1 (IRQ4 3F8)*** .
3. ***COM2 (IRQ3 2F8)*** .

Sets the internal COM2 port to IrDA (**I**nfra-**r**ed **D**ata **A**ssociation) mode.
This mode allows wireless operation of a supported device attached to the COM2 port (utility connector).

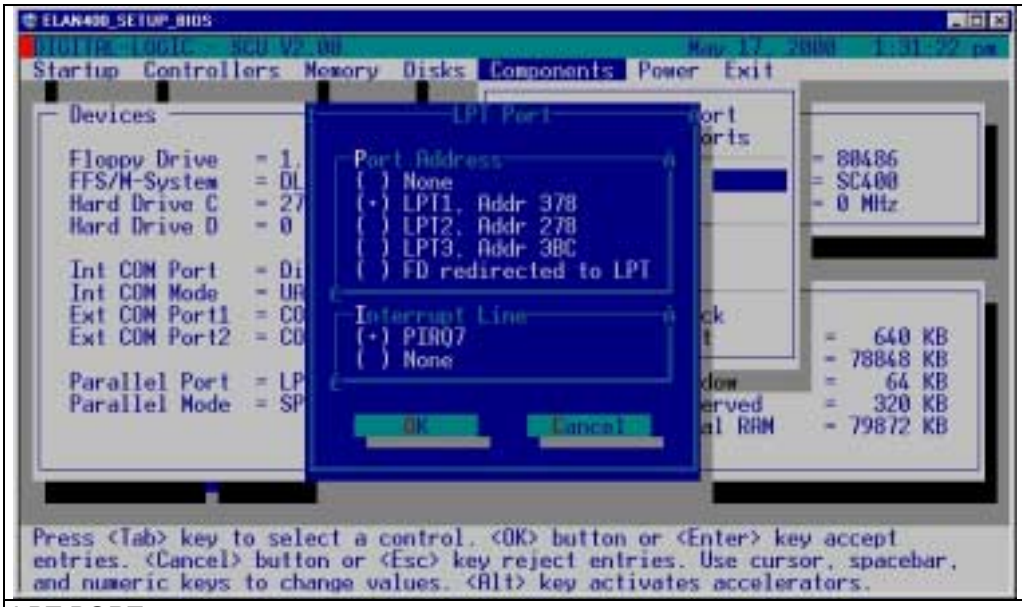
- Jump with TAB- key
- Choose option by using the ARROW- keys
- Select device with the SPACE- key
- Save with ENTER



EXTERNAL COM PORT

Configures any external serial ports, SUPER I/O.

- Jump with TAB- key
- Choose option by using the ARROW- keys
- Select device with the SPACE- key
- Save with ENTER



LPT PORT

Configures the port address and the interrupt line (IRQ) for the internal parallel port (LPT). Specifies the port designation and IRQ assignment for the LPT port.
Available I/O addresses include:

1. ***None*** .
2. ***LPT1 (378h)*** .
3. ***LPT2 (278h)*** .
4. ***LPT3 (3BC)*** .
5. ***FDD redirect to LPT* port** .

Valid IRQs include:

1. ***PIRQ7***.
2. ***None*** .

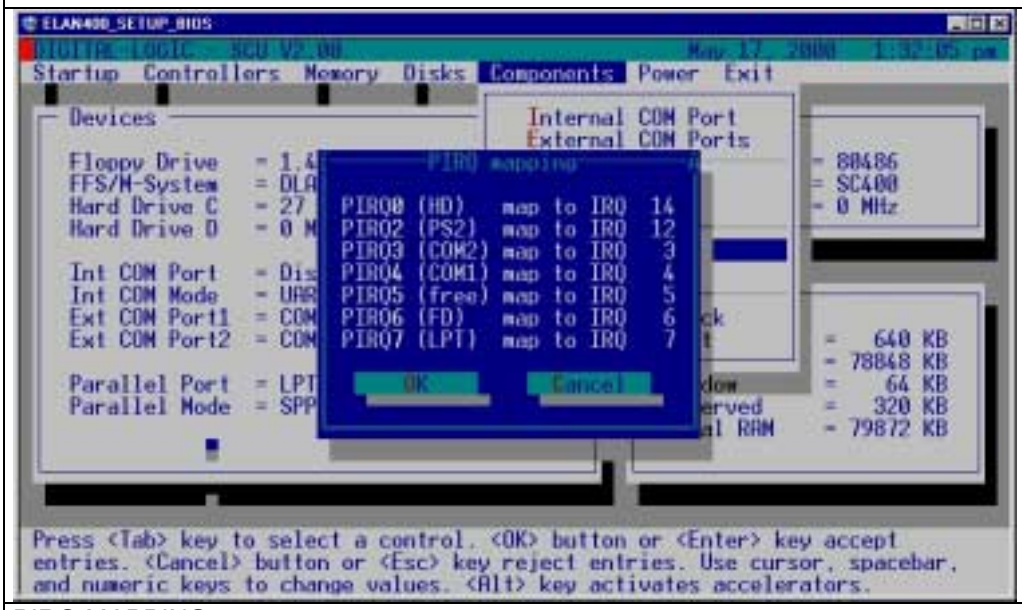
- Jump with TAB- key
- Choose option by using the ARROW- keys
- Select device with the SPACE- key
- Save with ENTER

LPT TYPE

Specifies the type of LPT port.
Selections include:
1. *Standard AT (Centronics **SSP**) .*
2. *Enhanced Parallel Port (**EPP**) .*
3. *Bidirectional (Extended Capabilities Port, **ECP**) .*
4. ***ECP and EPP** .*

EPP mode:
1. **V1.9** .
2. **V1.7** .

- Jump with TAB- key
- Choose option by using the ARROW- keys
- Select device with the SPACE- key
- Save with ENTER



PIRQ MAPPING

Programmable **I**nterrupt **R**e- **Q**uest, to redirect certain IRQ's

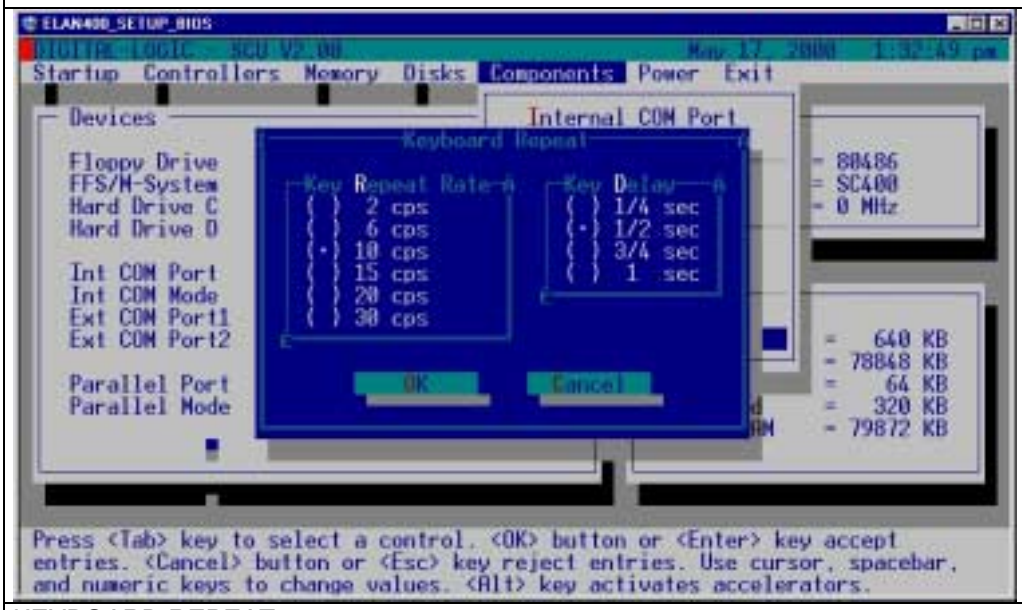
- Jump with TAB- key
- Choose option by using the ARROW- keys
- Select device with the SPACE- key
- Save with ENTER

PGPO MAPPING

Programmable **G**eneral **P**urpose **O**utput, defines several options:

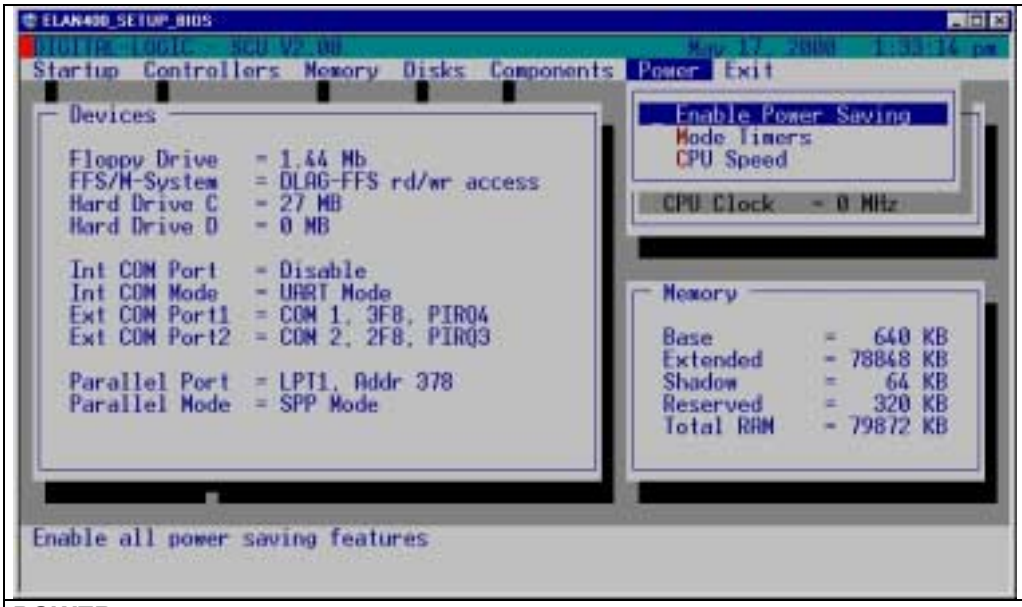
1. 3rd COM- port controller settings
2. LCD character activation
3. LCD width

- Jump with TAB- key
- Choose option by using the ARROW- keys
- Select device with the SPACE- key
- Save with ENTER



KEYBOARD REPEAT

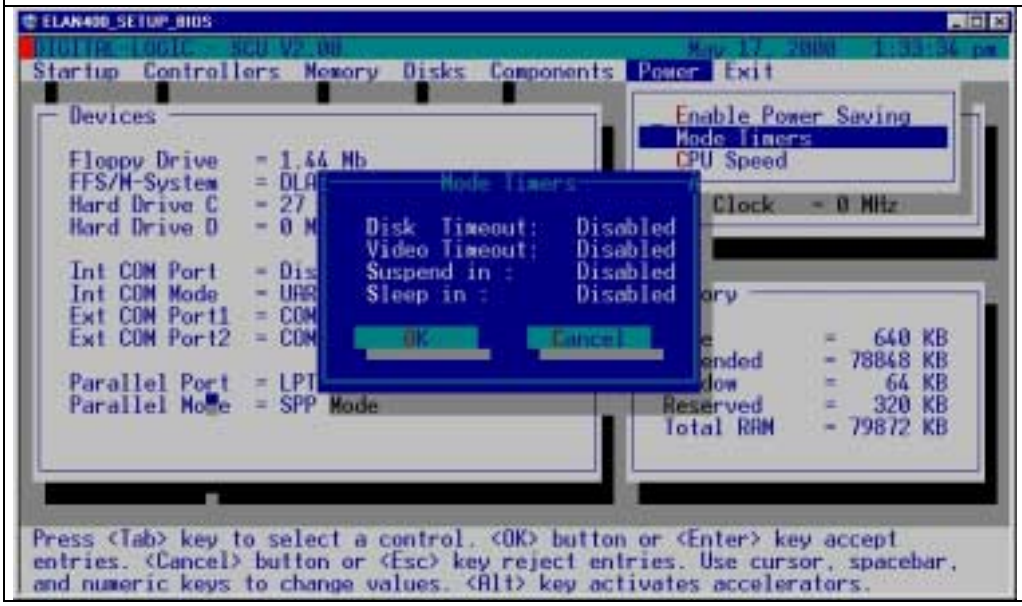
- Jump with TAB- key
- Choose option by using the ARROW- keys
- Select device with the SPACE- key
- Save with ENTER



POWER

- ENABLE POWER SAVINGS
  Enables or disables all power-saving features. When enabled, any specific power-saving features that have been enabled are in effect. When this item is disabled, all power-saving features, whether or not they are individually enabled, are not in effect.

  enable/disable with ENTER
- MODE TIMER
- CPU SPEED



MODE TIMER

Disk:
disabled/ 125ms/ 250ms /500ms/ 1s/ 4s/ 8s/ 16s
Video:
disabled/ 125ms/ 250ms /500ms/ 1s/ 4s/ 8s/ 16s
Suspend:
8s/ 16s/ 32s/ 1min/ 4min/ 8min/ 16min
Sleep:
1min/ 2min/ 4min/ 8min/ 16min/ 32min/ 60min

- Jump with TAB- key
- Choose option by using the ARROW- keys
- Save with ENTER

CPU SPEED

Sets the CPU speed for *Hyper Speed*, *High Speed*, and *Low Speed* which are used by the Mode Timers option.
Available options for each speed include:

1. **Hyper Speed** .
   66 or 99MHz
2. **High Speed** .
   8, 16, or 33MHz
3. **Low Speed** .
   1, 2, 4, or 8MHz

- Jump with TAB- key
- Choose option by using the ARROW- keys
- Select device with the SPACE- key
- Save with ENTER

This option allows the user to specify the time-out periods for the following operating modes:
Hyper Speed, *High Speed*, *Low Speed*, and *Standby*.

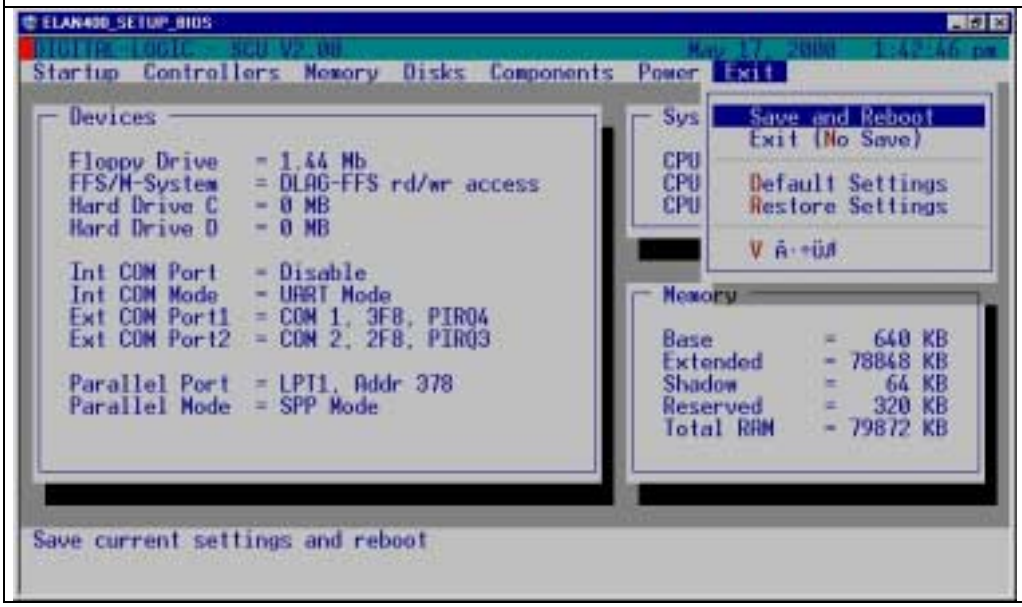
1. **Hyper Speed** *Disabled, 0.125, 0.25, .5, 1, 4, 8, 16 seconds*
2. **High Speed** *Disabled, 0.125, .25, .5, 1, 4, 8, 16 seconds*
3. **Low Speed**
   Disabled, 8, 16, 32 seconds, 1, 4, 8, 16 minutes
4. **Standby**
   Disabled, 1, 2, 4, 6, 8, 16, 32, 60 minutes

Hyper Speed
Hyper Speed is the highest performance operating mode.
It allows the CPU to run at speeds of 66MHz or 99MHz.
It provides the best performance but conserves the least amount of battery life.

High Speed
High Speed is a high performance operating mode.
It allows the CPU to run at a speed of 33MHz.
It provides a high level of performance but consumes more battery power than low speed but not as much as Hyper Speed.

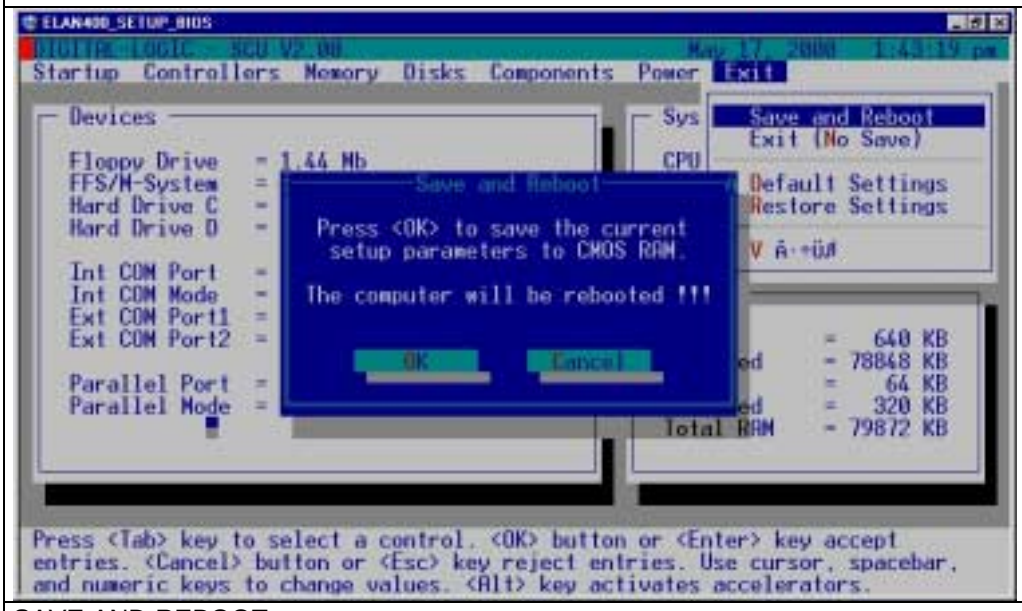
Low Speed
Low Speed is a low performance operating mode in which the CPU runs at a speed of 8MHz or less. Although it provides a lower level of performance than High Speed and Hyper Speed, it provides a greater degree of battery conservation.

Standby Mode
Standby Mode is a power-saving mode essentially stops the CPU clock.
This provides the greatest degree of battery conservation.



EXIT

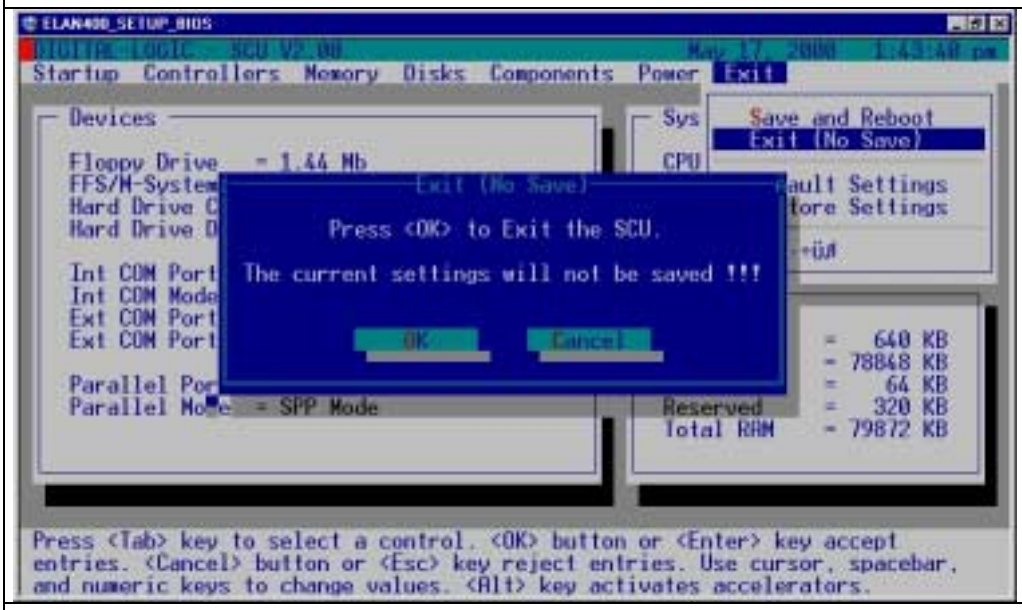
- SAVE AND REBOOT
- EXIT NO SAVE
- DEFAULT SETTINGS
- RESTORE SETTINGS
- VERSION



SAVE AND REBOOT

Saves the current settings and reboots the system so that the settings can take effect.

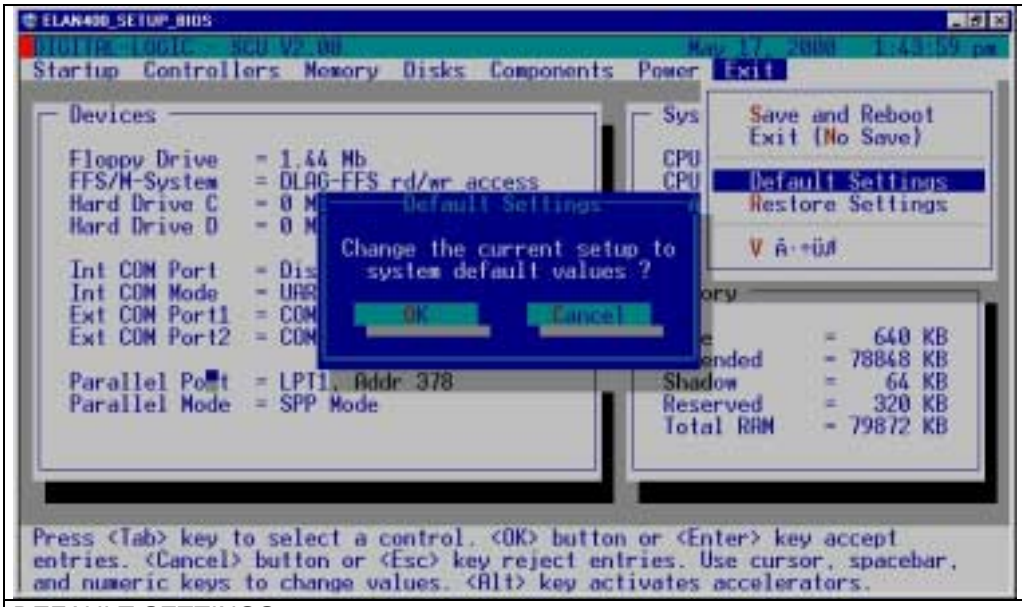
- Choose option by using the ARROW- keys
- Save with ENTER



EXIT NO SAVE

Exits the SCU without saving any of the current changes.

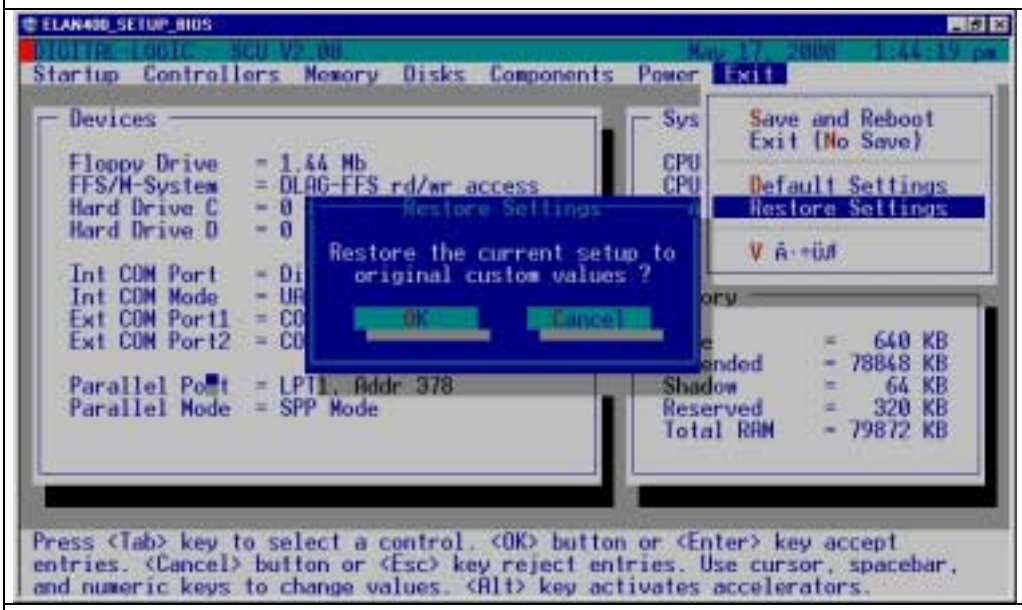
- Choose option by using the ARROW- keys
- Save with ENTER



DEFAULT SETTINGS

Restores the default settings (the original settings found in the ROM).

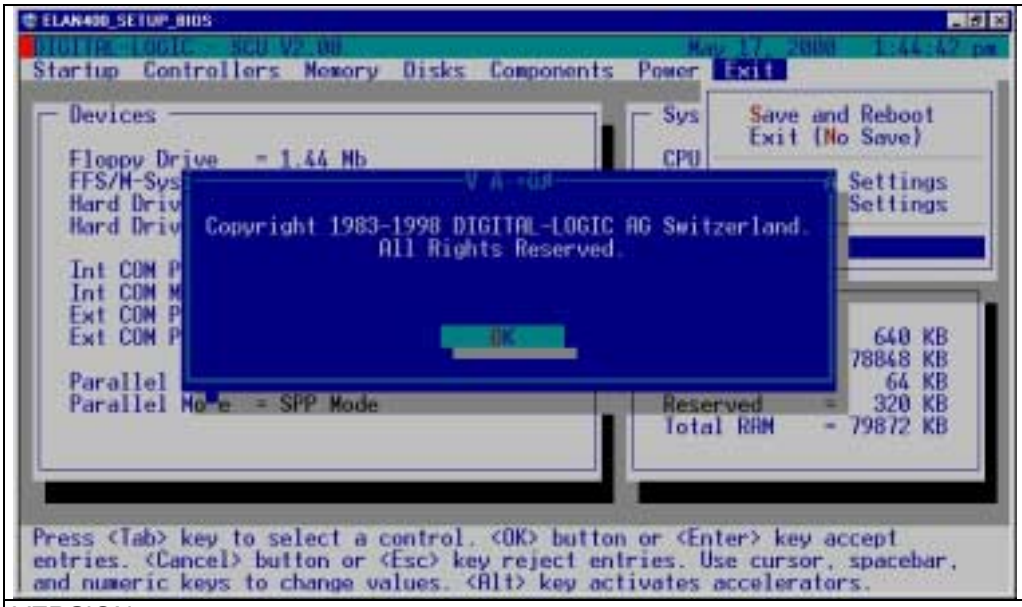
- Choose option by using the ARROW- keys
- Save with ENTER



RESTORE SETTINGS

Restores the current setup settings to the original custom settings (those that were set prior to making the current changes).

- Choose option by using the ARROW- keys
- Save with ENTER



VERSION

Display the current BIOS version information which includes version number and date.

- Exit with ENTER or ESC

13 THERMAL SPECIFICATIONS

Each product will be burnin with 10 cycles of 30min. between the operating temperatures of –25°C to +70°C or higher if extended ranges are odered.

The critical point is to meet the max. Tcase temperature of the CPU .
This temperature is specified by 110°C for the SQFP case. The tables show the allowable ambient temperature at various airflows and with different heatsink configurations.

CPU: ELAN400 Clock: 99Mhz T (case) = 110°C Powerconsumption: 4W

Temp. ambient	Measured temp. by Ta=20°C	T ambient no Airflow 0 m/sec	T ambient Airflow 3 m/sec	T ambient Airflow 6 m/sec	-E27
no Heat Sink horiz.	65°C	50°C	55°C	60°C	n.a.
no Heat Sink vert.	65°C	50°C	55°C	60°C	n.a.

CPU: ELAN400 Clock: 66Mhz T (case) = 110°C Powerconsumption: 3W

Temp. ambient	Measured temp. by Ta=20°C	T ambient no Airflow 0 m/sec	T ambient Airflow 3 m/sec	T ambient Airflow 6 m/sec	-E28
no Heat Sink horiz.	55°C	70°C	73°C	75°C	n.a.
no Heat Sink vert.	53°C	70°C	73°C	75°C	n.a.

CPU: ELAN400 Clock: 33Mhz T (case) = 110°C Powerconsumption: 3W

Temp. ambient	Measured temp. by Ta=20°C	T ambient no Airflow 0 m/sec	T ambient Airflow 3 m/sec	T ambient Airflow 6 m/sec	-E28
no Heat Sink horiz.	50°C	70°C	75°C	78°C	85°C
no Heat Sink vert.	49°C	70°C	75°C	78°C	85°C

CPU: ELAN400 Clock: 8Mhz T (case) = 110°C Powerconsumption: 2.5W lowspeed

Temp. ambient	Measured temp. by Ta=20°C	T ambient no Airflow 0 m/sec	T ambient Airflow 3 m/sec	T ambient Airflow 6 m/sec	
no Heat Sink horiz.	30°C	85°C	85°C	85°C	
no Heat Sink vert.	30°C	85°C	85°C	85°C	

13.1 Thermal analysis for mounting in a case

Since the integrated heatsink is unidirectional, the airflow must be exactly in the direction of the heatrails. If possible mount the board verticaly, so that the heatrails shows up/down. The self produced airflow is around 3m/sec in this case.

Special attention must given for mounting the PC-product in a fully closed case/box. The thermal energy will be stored in the innerroom of this environment.

If the case may have a fan:

1. The hot air must be exchanged with a filtered fan by cool air from outdoor.
2. The hot air must be cooled with a heat exchanger

If the case may not have any fan or opening to exchange the hotair:

1. The heatsink of the CPU must be mounted direkt to a heatsink integrated in the case. The thermal energy goes not through the air. They will be conducted direktly through the alloy of the heatsink outdoor.

2. Reduce the thermal energy produtction by lowering the CPU performance. Ex. using 33Mhz or lower clock only.

14 DIAGNOSTICS

A. Error on boot time:

1. Check if you have a bootable floppy or harddisk.
2. Check the CMOS parameter with the setup tools.
3. Re-enter the correct values with setup.

B. If no display on the screen is available:

1. Check the power circuitry.
2. Check the polarities of the cables.
3. Measure the voltage of the power supply under load and offload.
4. Measure the current between the supply and the MICROSPACE PC.
5. Connect a floppy: does the bezel led light blink?
6. Does the harddisk spindle motor start?

C. If the error appears again:

1. Contact your local Digital-Logic Technical Support in your country.

2. Use Internet Support Request form on http://www.digitallogic.com -> Support -> Support Request

3. Send a FAX or an E-mail to DIGITAL-LOGIC AG with a description of your problem.

DIGITAL-LOGIC AG
Dept. Tech. Support
Nordstr. 4F
CH-4542 Luterbach (SWITZERLAND)

Fax: ++41-32 681 58 01
E-Mail: support@digitallogic.com

14.1 The POD-Errors

The following numeric codes appear on the System Board LEDs (in hex) to report the progress of the Power-On Diagnostics, or on the LEDs and on the screen (in decimal) to report errors found by the Diagnostics. The Pod-Latch address is 80h on ELAN chipsets.

Code	Description
00	System Initialization
01	Initialize the chipset
02	Test RAM
03	Move Boot Loader into RAM
04	Execute in RAM
05	Check Override Option
06	Shadow System BIOS
07	Checksum System BIOS ROM
08	Proceed with Normal Boot
09	Proceed with Crisis Boot
0F	Fatal Error
10	Signals that a Reset has occurred
11	Turn off FASTA20 for POST
12	Signal Power on Reset
13	Initialize the Chipset
14	Search for ISA Bus VGA Adapter
15	Reset Counter/Timer 1
16	User Register Configuration through CMOS
17	Size Memory
18	Dispatch to RAM Test
19	Checksum the ROM
1A	Reset PICs
1B	Initialize Video Adapter(s)
1C	Initialize Video (6845 registers)
1D	Initialize Color Adapter
1E	Initialize Monochrome Adapter
1F	Test 8237A Page Registers
20	Test Keyboard
21	Test Keyboard Controller
22	Check if CMOS RAM Valid
23	Check Battery Fail & CMOS Checksum
24	Test the DRAM Controllers
25	Initialize 8237A Controller
26	Initialize Interrupt Vectors
27	RAM Quick Sizing
28	Protected Mode Entered Safely
29	RAM Test Completed
2A	Protected Mode Exit Successful
2B	Setup Shadow

Code	Description
2C	Going to Initialize Video
2D	Search for Monochrome Adapter
2E	Search for Color Adapter
2F	Sign-On Messages Displayed
30	Special Initialization of Keyboard Controller
31	Test If Keyboard Present
32	Test Keyboard Interrupt
33	Test Keyboard Controller Command Byte
34	TEST, Blank, and Count All RAM
35	Protected Mode Entered Safely (2)
36	RAM Test Complete
37	Protected Mode Exit Successful (2)
38	Update OUTPUT Port
39	Setup Cache Controller
3A	Test if 18.2Hz Periodic Working
3B	Test for RTC Ticking
3C	Initialize the Hardware Vectors
3D	Search for and Initialize the Mouse
3E	Update NUMLOCK Status
3F	Special Initialization of COM and LPT Ports
40	Configure the COM and LPT Ports
41	Initialize the Floppies
42	Initialize the Hard Disk
43	Initialize the Option ROMs
44	OEM's Initialization of Power Management
45	Update NUMLOCK Status
46	Test for Coprocessor Installed
47	OEM Functions Before Boot
48	Dispatch to Operating System Boot
49	Jump into Bootstrap Code

14.2 Power Management Diagnostic Codes

This section lists the port 80h diagnostic codes assigned to SystemSoft's Power Management software.

14.2.1 Dynamic Power Management Modechange Codes

Code	Description
00h	**Keypressed and switch to Hypermode**
11h	Hyper Mode
12h	Timeout HS to LS (local.asm)
13h	Timeout LS to Suspend (hwshell.asm)
14h	Timeout LS to Suspend (smichip.asm)
15h	Timeout LS to Suspend2 (smichip.asm)
1Fh	WakeUp from Resume Pin (smichip.asm)
1Eh	WakeUp from keyboard (smichip.asm)
1Ah	Go into Suspend_Req_Fall (smichip.asm)
1Bh	Go into Suspend_Req_Rise (smichip.asm)

14.2.2 SMI Generic Function Codes

Code	Description
00h	Invalid SMI
01h	Suspend Standby Direct
02h	Global Standby Direct
03h	System Idle Shell Entry
04h	Device Timeout (on → off transition request)
05h	Device Timeout (off → on transition request)
06h	System Activity
07h	Suspend Switch
08h	AC Power Change
09h	External Switch
0Ah	Battery Low
0Bh	Hardware Timer
0Ch	Reschedule SMI
0Dh	CPU Standby CPU Idle
0Eh	CPU Global Standby due to Idle Condition
0Fh	CPU Autopower Off Idle Transition

14.2.3 BIOS Menu Codes

Code	Description
10h	Set Up Hardware
11h	Rest CPU to Real Mode (on some chipsets)
12h	Set Up CMOS Clocks
13h	See Below ①
14h	See Below ①
15h	See Below ①
16h	See Below ①
17h	See Below ①
18h	See Below ②
19h	See Below ②
1Ah	See Below ②
1Bh	See Below ②
1Dh	See Below ②
1Eh	See Below ②
1Fh	Reset Hardware

①Codes 13h - 17h:

Set up of Interrupt Vector Table & Video Memory for SCU

②Codes 18h - 1Eh:

Return Interrupt Vector and Video Mode and Memory to state prior to exiting SMI

14.2.4 General Suspend/Resume Operation Codes

Code	Description
50h	start at HW_Mini_Resume
51h	Go to Suspend @ 5 Volt
52h	Go to Suspend @ 5 Volt
53h	Resuming 5 Volt
54h	Resuming 0 Volt
5Bh	Suspend to Disk Resume
5Ch	Start Suspend to Disk
5Dh	Reset Suspend to Disk
5Eh	Suspend to Disk Entry
5Fh	Suspend to Disk Exit

14.2.5 General Operation Codes

Code	Description
60h	CS Layer Information Delimeter Byte
6Dh	Enter Global Standby
6Eh	Exit Global Standby
6Fh	Reset System

14.2.6 SMI Device Codes

Code	Description
70h	Set Up for Device Initialization
71h	Device Initialization
72h	Set Up for Device Standby
73h	Device Standby
74h	Set Up for Device Wakeup
75h	Device Wakeup
76h	Set Up for Suspend Device
77h	Suspend Device
78h	Set Up for Resume Device
79h	Resume Device
7Ah	Set Up for Device Off
7Bh	Device Off
7Ch	Set Up for Device On
7Dh	Device On

14.2.7 Generic Video Codes

Code	Description
80h	Blank VGA Screen(s)
81h	Unblank VGA Screen(s)
82h	Standby VGA Chip
83h	Wakeup VGA Chip
84h	Suspend VGA Chip
85h	Resume VGA Chip
86h	Get VGA Ready for Menu (Text Mode)
87h	Restore VGA Mode from Text Mode
88h	Save Memory Block
89h	Restore Memory Block
8Ah	Save VGA State
8Bh	Restore VGA State
8Ch	Switch Display
8Dh	Information
8Eh	Prepare Video for Initialization

14.2.8 Suspend to Disk Codes

Code	Description
90h	Prepare to switch to Real Big Mode
91h	Running in Real Big Mode
92h	Exit Real Big Mode
93h	Suspend Disk
94h	No PM Suspend
95h	Save VGA
96h	Save 256K Video
97h	Save Memory
9Ah	Resume
9Bh	Resume Disk
9Ch	No PM Suspend
9Dh	Restore Memory
9Eh	Restore VGA
9Fh	Restore 256K Video

14.2.9 SMI Suspend Codes

Code	Description
A0h	Save CPU
A1h	Save Chipset Registers
A2h	Save Miscellaneous
A3h	Save Keyboard
A4h	Save A20h
A5h	Save Device
A6h	Save Ports
A7h	Reset CPU
A8h	Save Programmable Interrupt Controllers
AAh	Suspend Chip

14.2.10 SMI Resume Codes

Code	Description
B0h	Start Resume process
B1h	Resume Keyboard Initialization
B2h	Resume A20
B3h	Resume Keyboard
B4h	Resume OEM Device
B5h	Resume Ports
B6h	Resume Chipset Register Settings
B7h	Resume Miscellaneous
B8h	Resume
B9h	Resume
BAh	Set Up for Resuming Programmable Interrupt Controllers
BBh	Resume Programmable Interrupt Controllers
BCh	Resume Device
BDh	Resume DMA Initialization
BEh	Resume CPU Control

14.2.11 SMI Entry/Exit Codes

Code	Description
C0h	SMI Entry Marker
C1h	SMI Exit Marker
C2h	APM SMI Entry Marker
C3h	APM SMI Exit Marker

14.2.12 SMI Software/Hardware Request Codes

Code	Description
C4h	Software SMI Function Execution
C5h	Hardware SMI Function Execution

14.2.13 Global Standby Exit Codes

Code	Description
C6h	Exit upon Interrupt Request
C7h	Exit upon AutoPower Off
C8h	Exit upon Switch
C9h	Exit upon External Activity

14.2.14 Post/Reschedule Codes

Code	Description
CAh	Post Critical Events
CBh	Execute External Process
CCh	Restart Scheduler
CDh	Postpone Execution of Events
CEh	Events in Function
CDh	Post Events

14.2.15 APM Processing Codes

Code	Description
D0h	APM Check
D1h	INT F Connection
D2h	PM16 Connection
D3h	PM32 Connection
D4h	INT F Disconnected
D5h	CPU Standby
D6h	CPU Busy
D7h	Power
D8h	Enable/Disable APM
D9h	APM Defaults
Dah	Get Power Status (Battery, AC Line)
DBh	APM Event
DCh	Call APM Event

15 THE BIOS HARDDISK LIMITATIONS

Note:
The ELAN400 with the SYSTEMSOFT- BIOS does support maximum 8192 cylinders (ca 3GB !)

by Jan Steunebrink
Version 2, June 1999
This article is published at <http://web.inter.nl.net/hcc/J.Steunebrink/bioslim.htm>

Contents

Introduction

In the last few years, harddisks have become larger and faster at an incredible rate.
Due to the favorable prices of the present large Enhanced IDE drives and the ever expanding software footprint, the whole class of 486 and Pentiums PCs can benefit of a harddisk upgrade.
This article targets at these PCs that have a system BIOS dated from 1992 to 1998.
These BIOSes can limit the usable capacity of your new (E)IDE drive.

Note: All MB values in this article are 1,048,576 Bytes (1024x1024).

The Interrupt 13h interface

When IBM engineered the AT they put the interface for the harddisk (Int 13h) in the system BIOS.
Whenever an application wants to read from/write to a drive, it calls DOS.
DOS knows the structure of the disk and where the target file is located. It then calculates a CHS (Cylinder, Head, Sector) address and calls the BIOS via Int 13h.
The BIOS then executes the read or write command at this CHS address by accessing the HD controller directly via its I/O-port addresses.
The result is passed back to DOS who passes it back to the application.
This scheme makes DOS (drive) hardware independent and leaves the hardware specifics to the BIOS.

The traditional BIOS Int 13h interface has the following limitations (when called from DOS):
1024 Cylinders, 256 Heads and 63 Sectors/track.
With 512 bytes/sector this counts up to 8 GB (8064 MB)!

The 504 MB BIOS Limit

The (in)famous 504 MB (528 million bytes) limit comes from the fact that the ATA (IDE) specification has different limits than the BIOS. When the traditional BIOS Int 13h interface is used to control an (E)IDE harddisk, the limits are combined as illustrated below.

	BIOS	IDE	Combined Limit
Max. Sectors/track	63	255	63
Max. Heads	256	16	16
Max. Cylinders	1024	65536	1024
Max. Capacity	8 GB	127.5 GB	**504 MB**

If you do not have an Enhanced BIOS, you need one of these solutions to break the 504 MB barrier:

1. A translating (Enhanced) BIOS upgrade
2. Add-in card with an Enhanced BIOS (this takes over the Int 13h interface only)
3. Software like Disk Manager from Ontrack or EZ-Drive from StorageSoft (formerly Micro House).

The translating BIOS

The traditional BIOS Int 13h interface passes the CHS address directly onto the harddisk controller, thereby creating the 504 MB barrier in case of an (E)IDE drive.
Nowadays we have the translating BIOS whose Int 13h interface can translate the CHS address to a different geometry or to an LBA (Logical Block Address).

The CHS which DOS uses to call the BIOS is now called the L-CHS (Logical CHS) and the CHS which the BIOS uses to control the drive is the P-CHS (Physical CHS).

Two translation methods can be distinguished:

1. The first method is normally used for direct L-CHS to P-CHS translation.
   According the ATA-2 specification, all EIDE drives up to 8 GB should conform to the (BIOS) Sector limit of 63 sectors/track. On a >504 MB drive with a maximum of 16 Heads (IDE limit), only the number of Cylinders will be above the BIOS limit of 1024.
   This makes a simple CHS translation scheme possible in which the number of Cylinders is divided by 2, 4, 8 or 16 and the number of Heads is multiplied with the same number.
   The number of sectors/track will remain unchanged.
   The maximum capacity depends on the number of sectors/track and is equal to the BIOS limit of 8064 MB if the drive has 63 sectors/track.

   The following table illustrates this.

Actual Cylinders	Actual Heads	Altered Cylinders	Altered Heads	Max. Capacity
1<C≤1024	1<H≤16	C=C	H=H	504MB
1024<C≤2048	1<H≤16	C=C/2	H=H*2	1008MB
2048<C≤4096	1<H≤16	C=C/4	H=H*4	2016MB
4096<C≤8192	1<H≤16	C=C/8	H=H*8	4032MB
8192<C≤16384	1<H≤16	C=C/16	H=H*16	8064MB

Example: A 2014MB drive has a CHS count of 4092x16x63. According to the above method, this will be translated into 1023x64x63, nicely within the Int 13h interface limit.
In other words; DOS (or any other OS which uses the BIOS Int 13h interface for that matter) thinks that it deals with a drive which has 1023 Cylinders, 64 Heads and 63 Sectors/track while the BIOS still controls the drive with its original geometry.
Note: A drive which has 16 Heads and more than 8192 Cylinders (> 4 GB) will be translated to 256 Heads. This is a problem because DOS can't handle 256 Heads, but there is an easy workaround for that. See "The 4 GB Limit" below.

Because the translation is done with powers of 2, the software inside the BIOS can accomplish this by simply shifting the bit patrons of the Cylinder and Head addresses one or more places to the right respectively to the left.
This translation is therefore known as the bit-shifting translation.

2. The second method is used with drives that can be accessed with LBA.
The translation here is from L-CHS (DOS) to LBA (drive).
With LBA, all sectors are numbered sequentially and the P-CHS is not used anymore.
The BIOS controls the drive by sending the required sectornumber (= LBA) to the drive instead of the P-CHS.
DOS however, still works with the CHS address so the BIOS has to calculate an artificial L-CHS and present this geometry to DOS.
This L-CHS is calculated from the total capacity of the drive as indicated below.

Capacity X in bytes	Sectors	Heads	Cylinders
1<X≤504MB	63	16	X/(63*16*512)
504MB<X≤1008MB	63	32	X/(63*32*512)
1008MB<X≤2016MB	63	64	X/(63*64*512)
2016MB<X≤4032MB	63	128	X/(63*128*512)
4032MB<X≤8032.5MB	63	255*	X/(63*255*512)

As you can see, the number of Sectors/track is fixed at 63 and the number of Heads is 16 or a multiple of that. Then the number of Cylinders is calculated by dividing the total capacity in bytes by the number of bytes per cylinder.
This translation is called the LBA assisted method.
*Note: The last row of the above table indicates that a drive over 4 GB should be translated to 255 i.s.o. 256 Heads. This to avoid problems with DOS which is unable to handle 256 Heads. For more information see "The 4 GB Limit" below.

The two translation methods produce similar geometries (L-CHS) in many cases. The difference becomes apparent when a drive reports less than 63 Sectors/track.
The LBA assisted method **always** assigns 63 Sectors/track.
The LBA assisted method is therefore more flexible than the bit-shifting translation and places no limits on the reported drive geometry (P-CHS).
The bit-shifting method however, has to be used if the drive cannot handle LBA.
Note that most translating BIOSes require you to set the P-CHS values in the BIOS setup.

With any translation method, the BIOS has to ensure that the sectors are accessed in the same order as with "untranslated" access.
This however, is NOT always the case. Therefore, it can be DANGEROUS to your data if you change to another translation mode on an already formatted drive!

Does my BIOS support translation?

Now that the theory has been dealt with, it is time to answer this question.
First check your BIOS date. If it is 7/94 or later, chances are pretty good that you have a translating BIOS. With older BIOSes, your chances are reducing.
Secondly, you can check in the BIOS SETUP if you have User definable drive types and if there is a translation MODE selection. MODE indications like LARGE or ECHS are normally used for bit-shifting translation. If you can select LBA mode, your BIOS supports the LBA assisted translation method.

But to know for sure, we have to examine the Fixed Disk Parameter Table (FDPT).
The BIOS maintains such a (16 byte) table for each physical drive it supports.
When a BIOS supports translation it uses an enhanced version of this table, the so-called Enhanced Disk Parameter Table (EDPT)
The location of the FDPT or EDPT for the first drive is stored in Interrupt vector 41h and Interrupt vector 46h holds the location of the table for the second drive.
(There are no pointers to the FDPTs for the third and subsequent drives because direct access to these tables is not used anymore. The pointers for the first two drives are however still there to remain compatible with older software. Nowadays, data from a FDPT/EDPT is provided by the BIOS via Interrupt 13h functions 8h and 48h.)

The table below indicates what is stored in the FDPT or EDPT.

Offset	Type	FDPT	EDPT
0-1	Word	Physical Cylinders	Logical Cylinders, limit 1024
2	Byte	Physical Heads	Logical Heads, limit 256
3	Byte	Reserved	A0h Signature, indicating EDPT
4	Byte	Reserved	Physical Sectors/Track
5-6	Word	Precompensation (Obsolete)	Precompensation (Obsolete)
7	Byte	Reserved	Reserved
8	Byte	Drive Control Byte	Drive Control Byte
9-10	Word	Reserved	Physical Cylinders, limit 65536
11	Byte	Reserved	Physical Heads, limit 16
12-13	Word	Landing Zone (Obsolete)	Landing Zone (Obsolete)
14	Byte	Physical Sectors/Track	Logical Sectors/Track, limit 63
15	Byte	Reserved	Checksum

If your BIOS builds an EDPT for a certain drive, it is a translating BIOS!
You can find an EDPT with the WDTBLCHK utility from Western Digital. This very useful program can be downloaded from my webpage. It is a self extracting file called CHKBIOS.EXE.
Or you can get it from <ftp://ftp.wdc.com/drivers/hdutil/chkbios.exe>.

There is a problem however. Most translating BIOSes only build an EDPT i.s.o. a FDPT if the drive is accessed in a translation mode. I therefore use the following trick.

This trick assumes a PC with one < 504 MB harddisk and a BIOS with user definable drive types.

- Enter the BIOS setup and select for the second (non-present) drive a CHS of 2048x16x63.
- If possible select a mode like ECHS, LARGE or LBA for this drive. You probably see the CHS numbers change, indicating the translation.
- Now exit the BIOS setup via "save and exit" and let the computer reboot.
- You probably get an error message because of the missing second drive. Ignore this and continue the boot to the DOS prompt. (If you have Windows 95/98, hit the F8 key when you see "Starting Windows 95/98 ....." and select "Command prompt only" from the menu.)
- Now run the WDTBLCHK utility and check on screen 3-1 if the BIOS build an Enhanced table for the second drive. Look under "INT 46 DRIVE". On the same screen you can also check both the L-CHS and P-CHS as stored in the EDPT. This to confirm a correct translation.
- If this works, repeat the whole procedure with different CHS values to check the boundary conditions of the translation algorithm(s). Note that the **red** figures may bring you into problems as described in "The 2 GB BIOS Limit" below.
  Use 4095x16x63 / 4096x16x63 (2 GB); 8191x16x63 / 8192x16x63 (4 GB) and 16320x16x63 for the 8 GB limit. (The last figure should translate to 1024x255x63 in LBA mode.)

Restore the BIOS setup to its original settings after these tests.
Caution: To be on the safe side, (and to protect the data on your harddisk) it is best to use a boottable floppy with the WD Table Check utility on it. Reboot from this floppy, i.s.o. your harddisk, as long as the BIOS setup is in the 'test' mode.

There is another way to detect a translating BIOS but you need a > 504 MB (actually > 1024 cylinders) drive for that. Connect this drive as slave on the primary IDE-port and, if possible, have the BIOS setup autodetect this drive's geometry. After "save and exit" and reboot use the WD utility to check for an Enhanced table on the second drive.

Finally, if you have a BIOS which supports the IBM/Microsoft Int 13h extensions, you do not need all these tricks because **all** those BIOSes support translation. These BIOSes started to appear in 1995. See "The Int 13h extensions" below.

The 2 GB BIOS Limit

All major harddisk manufacturers have reported about a 2 GB BIOS limit.
It appears that a number of translating BIOSes, manufactured *before* May 1996 have problems translating drives with cylinder values over 4095. This limits the capacity to 2 GB (2015 MB).
To break this limitation, many harddisk manufacturers supply a software solution like Disk Manager or EZ-Drive with their large harddisks.
Alternately, there may be a BIOS upgrade available for your system.

A nice article about this problem and its various solutions is "The other BIOS Limitation!" from Western Digital. Look at <http://www.westerndigital.com/service/tip1196.html>
Details from one of the BIOS manufacturers can be found in Micro Firmware's article "Notes on Installing Hard Drives Larger Than 2 GB". See <http://www.firmware.com/pb4ts/over2gb.htm>

Reported problem scenarios for the affected BIOSes are:

- The BIOS can only see a maximum of 4095 cylinders thereby truncating the usable drive space to 2015 MB
- The BIOS uses only the lower 12 bits of the 16-bit cylinder word thereby **losing 2015 MB** of the drive's capacity when a drive is larger then 2 GB!
- When a cylinder count of over 4095 is entered in the BIOS setup, the BIOS will cause a system lock-up at boot time, making the entire system inaccessible.

Apart from the last scenario, I don't know any other reliable method to detect this 2 GB limit than by hooking up a drive with more then 4095 cylinders. However, you may have found something when using "the trick" for detecting a translating BIOS. (See above)

BIOSes dated May 1996 or later should be free of these problems and support translation up to the Int 13h interface limit of 8 GB.

Related to the 2 GB limit is the Award BIOS harddisk size display limit bug.
I've found that all Award BIOSes dated July 1994 or later correctly support the LBA assisted translation up to the 8 GB limit. There is however a bug in the BIOSes dated before January 1996 that limits the harddisk size display, on the BIOS Setup and boot screens, to 2015 MB.
Whenever a drive is 2016 MB or larger, the display starts to count from zero again. The same happens at 4032 and 6048 MB.
This looks a lot like the above mentioned 2 GB limit but is actually only a bug in the harddisk size calculation routine, and it doesn't affect the BIOS support for these large drives.
For these Award v4.50(P)(G) BIOSes, just use the HDD AUTO DETECTION feature to Setup the drive, select the option with LBA at the end, and disregard the incorrect HD size display.

The 4 GB Limit

Yes, there is another limit! This is actually an Operating System issue but the appropriate way to deal with this problem is to account for it in the system BIOS.
It appears that DOS and Windows 95/98 are limited to 255 Heads.
A translated geometry of 256 Heads will therefore create a problem.
This can happen when a drive has 16 Heads and more than 8192 Cylinders (> 4032 MB).

As indicated under "The translating BIOS" the LBA assisted method should translate to 255 i.s.o. 256 Heads when a drive is larger than 4032 MB.
If this is not the case, then the bit-shifting translation (select ECHS or LARGE) should be used with the following workaround:

1. Enter the CHS values of the drive in CMOS setup or do a "HDD autodetect" and do not select a translation mode yet
2. Adjust the number of heads from 16 to 15
3. Multiply the number of cylinders by 16/15 (round down to a whole number)
4. Adjust the number of cylinders to this higher amount
5. Check or select a bit-shifting translation mode
6. Save and exit CMOS setup and partition and format the drive.

With this workaround, a translated geometry of 15 X 16 = 240 Heads will be used.
This limits the maximum L-CHS to 1024x240x63 which is equivalent to 7560 MB.

For more information read the excellent article "Issues with Hard Drives Over 4 GB".
Look at <http://www.firmware.com/pb4ts/over4gb.htm>

The Int 13h extensions - Breaking the 8 GB barrier

With the rate in which the harddisk technology is progressing, we are now faced with EIDE drives exceeding the 8064 MB limit of the Int 13h interface.
The only way to break this limit is to ditch the CHS system in favor of a direct LBA interface.
The SCSI drive technology uses this kind of LBA interface already for years.
LBA works with a 64-bit sector address, so we can (theoretically) access the staggering amount of 8,796,093,022,208 GB! Due to limitations of the ATA interface, only the lower 28 bits of the LBA address can used by an EIDE drive resulting in a, still formidable, limit of 128 GB.

Windows 95/98 is ready for this because it already uses LBA internally.
Except in Safe or Compatibility Mode, the protected mode diskdriver takes over the BIOS Int 13h interface and can access a drive directly in LBA.

To remain compatible with all the Operating Systems that still use the CHS system to call the BIOS (including Windows 95/98 at boottime), we need an extension on, rather than a replacement of, the Int 13h BIOS interface.
This is where the IBM/Microsoft Int 13h extensions come in.
This specification adds new functions to the Int 13h interface. These new functions are fundamentally different from the conventional Int 13h interface and allow the BIOS to be called directly with the required LBA.

A BIOS with the IBM/Microsoft Int 13h extensions supports both LBA capable and LBA non-capable drives. To do this, such a BIOS must support all of the following translations.
Traditional Int 13h functions:

- direct CHS addressing (Normal mode)
- L-CHS to P-CHS translation
- L-CHS to LBA translation

Extended Int 13h functions:

- direct LBA addressing
- LBA to P-CHS translation

Because BIOSes with the IBM/Microsoft Int 13h extensions also support at least four (sometimes eight) drives, including removable drives, you will find them in most modern PC's.
Most BIOSes dated January 1998 or later will support the Int 13h extensions.
To detect such a BIOS, you can download the EXTBIOS utility from my webpage.

Just as with the 504 MB barrier, you need one of these solutions to break the 8 GB barrier if you do not have the Int 13h extensions:
1. A BIOS upgrade
2. Add-in card with an Int 13h extension BIOS (this takes over the Int 13h interface only)
3. Software like Disk Manager or EZ-Drive.

Now that the BIOS is ready for it, what else do we need to break the 8 GB barrier?
Well, an Operating System capable of addressing a drive in LBA, either directly in protected mode or via the extended Int 13h BIOS interface. Windows 95 (all versions), Windows 98, Windows NT 4.0 (with SP4), Linux, and OS/2 Warp 3 and 4 (on HPFS) are fully equipped for this.
(So DOS 6.x, Windows 3.x, and Windows NT 3.5x and earlier versions are limited to 8 GB.)

Last but not least, we need a partition utility capable of creating partitions above the 8 GB boundary which the OS can see. The version of FDISK supplied with these Operating Systems is able to do this with the aid of new partition types 0Eh and 0Fh (FAT16) or 0Bh and 0Ch (FAT32; Win95b-OSR2 / Win98 only). Except for primary partitions below the 8 GB boundary, these new FDISK versions automatically select the new partition types, i.s.o. the old ones, when a BIOS with Int 13h extensions is detected.
Microsoft Knowledge Base article Q69912 gives a brief description of the various partition types. Look at <http://www.microsoft.com/kb/articles/q69/9/12.htm>
A nice article about the various versions of FDISK and their limitation is "Notes on DOS FDISK Command". Look at <http://www.firmware.com/support/bios/fdisk.htm>
Third party partition managers like "Partition Magic" from PowerQuest <http://www.powerquest.com> or "Partition Commander" from V Communications <http://www.v-com.com> can also go above the 8 GB boundary, and make the job of creating, re-sizing, and moving partitions, and selecting partition types much easier. And all this without loosing data!

Needless to say that the new partition types are invisible to DOS and that compatibility has to be sacrificed in the name of progress.

More information on the net

Possibly the most extensive source of information on this subject is
The Enhanced IDE/Fast-ATA/ATA-2 FAQ by John Wehman and Peter den Haan.
“Peter den Haan’s EIDE storage page” at <http://thef-nym.sci.kun.nl/~pieterh/storage.html> is the place to look for this FAQ. This page contains loads of links to other sites as well.

A good starting point for finding a BIOS upgrade is "Wim's BIOS page at"
<http://www.ping.be/bios/>
Start with the FAQ and then work your way through the other pages.

Jan Steunebrink
<J.Steunebrink@net.HCC.nl>

15.1 AMD ELAN400™ UART Errata

Errata #P20 – UART Compatibility Issues	
SYSTEM SYMPTOM:	The UART module handles some error conditions differently from the National 16550 UART chip.
ERRATA DESCRIPTION:	There are four areas in which the UART on the SC450 behaves differently from the National 16550 UART chip. They are: 1) The SC4x0 UART handles a framing error differently than the SuperI/O UARTs. The National 16550 UART tries to resynchronize the serial data stream after a framing error, assuming that the error is due to the next byte's start bit overlapping the current byte's stop bit. The SC4x0 UART resyncronizes after a frame error by looking for the next '1' after the frame error bit, and assuming this to be the present byte's stop bit. If the next bit is a '0', it is interpreted as the start bit of the next data byte. If the next bit is a '1', the UART waits for the next '0' in the data stream to be the start bit. 2) If there is a line status interrupt and a UART FIFO time-out interrupt pending simultaneously with an error in the receiving fifo, SC4x0 Rx interrupt id register (IIR) reports back a OXCE, which is an invalid code for the interrupt id. bit field (IIR bits 3-1 = " 1 1 1 "). The National UART IIR reports back OxC6, which indicates an interrupt for the line status register (IIR bits 3-1 = " 0 1 1 " with another interrupt pending (usually FIFO time-out interrupt -- IIR bits 3-1 = " 1 0 0 "). 3) With the fifo enabled, and an error in the receiving FIFO, SC4x0 Rx Line Status Register (LSR) sets the 16550-Error bit, which indicates that an error is present somewhere in the FIFO. This is the same as the National part. Upon reading the LSR, however, this bit (LSR bit 7) is cleared even if a subsequent error is still present in the FIFO. In the National part, LSR bit 7 is not cleared until the byte with error is at the top of the FIFO with no remaining errors in the FIFO. 4) In the National part, parity, framing, and break errors are reported to the LSR when the byte with the error is at the top of the FIFO. The SC4x0 UART reports errors after the corresponding data byte has been read into the receive buffer register. The National UART reports errors before the corresponding data byte is read into the receive buffer register. For example, if data byte 0x5c is sent and received with no errors, and then data byte 0x4b is sent and received with a parity error, the SC4x0 UART will report the parity error to the LSR after a read from the receive buffer register returns a 0x4b. The National UART will report the parity error after a read from the receive buffer register returns 0x5c. 5) Some UART's unify the last character in the FIFO and the Transmit Shift Register (TSR). These UART's raise the TEMT bit at the same time as the THRE bit during the Stop Bit of the last character in the FIFO, In the SC4x0 UART, the TEMT bit is raised separately from the TE bit, since the THR and FIFO are separate. Software that interrupts on TE that assumes the TSR no longer contains useful data will function incorrectly on the SC4x0.
HW / SW WORK AROUND:	Work Around: Modify UART driver software to conform to the UART behavior of the SC400, or eave error-handling to higher level protocol.

16 INDEX

O

P

R

S

T

U

V

W