AT96-MULTI-4 (AT96-Bus Version)
ISA96-MULTI-4 (ISA96-Bus Version)
ISA-MULTI-4 (ISA-Slot Version)
ISA-VGALCD-4 (ISA-Slot Version)
PC/104-VGALCD-4 (PC/104 Version)

Technical Manual

Rev.: 2.3

JUMP*tec*®
Industrielle Computertechnik AG
Brunnwiesenstraße 16
94469 Deggendorf/ Germany

PN of Manual:
Manual Rev.: 2.3
File: MLCDM123.doc

Table of Contents

..User Information

In this document JUMP*tec*Ò Industrielle Computertechnik AG will also be referred to by the short form " JUMP*tec*Ò ".

The information in this document has been carefully checked and is believed to be accurate and reliable. However, no responsibility is assumed for inaccuracies. Furthermore, JUMP*tec*Ò reserves the right to make changes to any portion of this manual to improve reliability, function or design. JUMP*tec*Ò does not assume any liability for any product or circuit described herein.

Trademarks

AT and IBM are trademarks of International Business Machines.

XT, AT, PS/2 and Personal System/2 are trademarks of International Business Machines Corporation.

Microsoft is a registered trademark of Microsoft Corporation.

Intel is a registered trademark of Intel Corporation.

ModulAT### is a trademark of JUMP*tec*Ò Industrielle Computertechnik AG

All other products and trademarks mentioned in this manual are trademarks of their respective owners.

General

For the circuits, descriptions and tables indicated no responsibility is assumed as far as patents or other rights of third parties are concerned.

The information in the Technical Descriptions describes the type of the boards and shall not be considered as assured characteristics.

Warranty

Each board is carefully and thoroughly tested before being shipped. If, however, problems should occur during the operation, please check your user specific settings of all boards included in your system. This is often the source of the fault. If a board is defective, it can be sent to your supplier for repair. Please take care of the following steps:

1. The board returned should correspond to the factory default settings since a test is only possible under this settings.
2. In order to repair your board as fast as possible , we require some additional information from you. Please fill out the attached Repair Form and include it with the defective board.
3. If possible, the board will be upgraded to the latest version without additional cost.
4. Upon receipt of the board, please be aware that your user specific settings were changes during the test.

Within the guarantee, the repair is free as long as the guarantee conditions were kept. If no fault has been found, you will be charged with the test cost due to the high test expenditure. Repairs outside of the guarantee will be charged.

This JUMP*tec*Ò product is warranted against defects in material and workmanship for our guaranteed warranty period from the date of shipment. During the warranty period, JUMP will, at its option, either repair or replace products which prove to be defective.

For warranty service or repair, the product must be returned to a service facility designated by JUMP*tec*Ò.

The foregoing warranty shall not apply to defects resulting from improper or inadequate maintenance or handling by buyer, unauthorized modification or misuse, operation outside of the environmental specifications for the product, or improper installation or maintenance.

JUMP will not be responsible for any defects or damages due to a faulty JUMP*tec*Ò product other than the products supplied by JUMP*tec*Ò.

The 5 Different Types of MULTI-4 boards

Type 1.)	AT96-MULTI-4M	with version one signed as M= AT96-MULTI-4M (full version)
	AT96-MULTI-4K	with version two signed as K= AT96-MULTI-4K (without matrix interface)
Type 2.)	ISA96-MULTI-4	(full version)
Type 3.)	ISA-MULTI-4	(full version)
Type 4.)	ISA-VGALCD-4A	(without matrix interface, keyboard interface and ModulAT®/120 bus, Floppy, IDE, COM1/2,LPT but with EEPROM)
	ISA-VGALCD-4T	(without STN interface, DC/DC converter and without matrix interface, keyboard interface and ModulAT®/120 bus, floppy, IDE, COM1/2, LPT and without EEPROM)
Type 5.)	PC/104-VGALCD-4	(with matrix interface, without keyboard interface, floppy, IDE, COM1/2, LPT)

Product	Panel support					Keyboard support			Bus support					I/O interface			EEP-ROM
	mono STN	Color TFT	Color STN	DC/ DC	SW cont.	Mat-rix	DSUB -9	Mini-DIN	Modul AT/CPU	ISA	AT96	ISA96	PC/ 104	FDC, IDE	COM 1,2	Para-llel	
AT96-MULTI-4M	yes	yes	yes	yes	yes	yes	yes	-	yes	-	yes	-	-	yes	yes	yes	yes
AT96-MULTI-4K	yes	yes	yes	yes	yes	-	yes	-	yes	-	yes	-	-	yes	yes	yes	yes
ISA96-MULTI-4	yes	yes	yes	yes	yes	yes	yes	-	yes	-	-	yes	-	yes	yes	yes	yes
ISA-MULTI-4	yes	yes	yes	yes	yes	yes	-	yes	yes	yes	-	-	-	yes	yes	yes	yes
ISA-VGALCD-4A	yes	yes	yes	yes	yes	-	-	-	-	yes	-	-	-	-	-	-	yes
ISA-VGALCD-4T	yes	yes	-	-	-	-	-	-	-	yes	-	-	-	-	-	-	-
PC/104-VGALCD-4	yes	yes	yes	yes	yes	yes	-	-	-	-	-	-	yes	-	-	-	yes

Matrix Keyboard support:	The matrix Keyboard scanner will work with ModulAT®/CPU or any other CPU over the Bus.
DSUB-9 keyboard support:	The DSUB-9 Keyboard connector is only available together with a ModulAT®/CPU plugged onto the ModulAT®/Bus.
MiniDIN Keyboard support:	The MiniDIN keyboard connector is only available together with a ModulAT®/CPU plugged onto the ModulAT®/Bus.
DC/DC:	If you choose a board without DC/DC converter together with a panel that needs contrast voltages like monochrome or color STN panels, an external DC/DC converter is needed.
SW control:	The panel software control allows to change contrast settings and different other panel features by keyboard.
EEPROM:	The EEPROM allows to store all video and IO setup parameters in EEPROM for further usage.

Features of the AT96-MULTI-4 family

The different baseboards mentioned above integrate the complete functionality of an IBM Multi I/O controller board, included the VGALCD function.

The MULTI-4 incorporates following features:

- VGA-CRT and LCD Controller with 512k byte video RAM.
- LCD Controller supports almost any type of LCD.
- Touch Matrix Controller.[1]
- Onboard DC/DC converter for positive or negative LCD contrast voltages.[2]
- Two serial interfaces, configurable as COM1 to COM4.[3]
- Printer port configurable as LPT1 to LPT3.[3]
- Floppy interface.[3]
- IDE hard disk interface.[3]
- Keyboard connector.[3]
- Battery connector.
- Power connector with 3,5 Floppy power connector.
- Low power CMOS technology.
- 5V only power supply.
- EEPROM for 16 Bytes of user data[4]
- BIOS calls for efficient use of extensions
- ModulAT®/120 Bus interface
- AT96-Bus interface for the AT96-MULTI-4
  ISA96-Bus interface for the ISA96-MULTI-4
  ISA-Bus interface for the ISA-MULTI-4
  PC/104-Bus for the PC/104-VGALCD
- Board size : 100 x 160 mm (3.94" x 6,3") for the AT96-MULTI-4 and ISA96-MULTI-4
  156,2x106,75 mm (6,15x4,2") for the ISA-MULTI-4
  90,17 x 95,87 mm for the PC/104-VGALCD-4

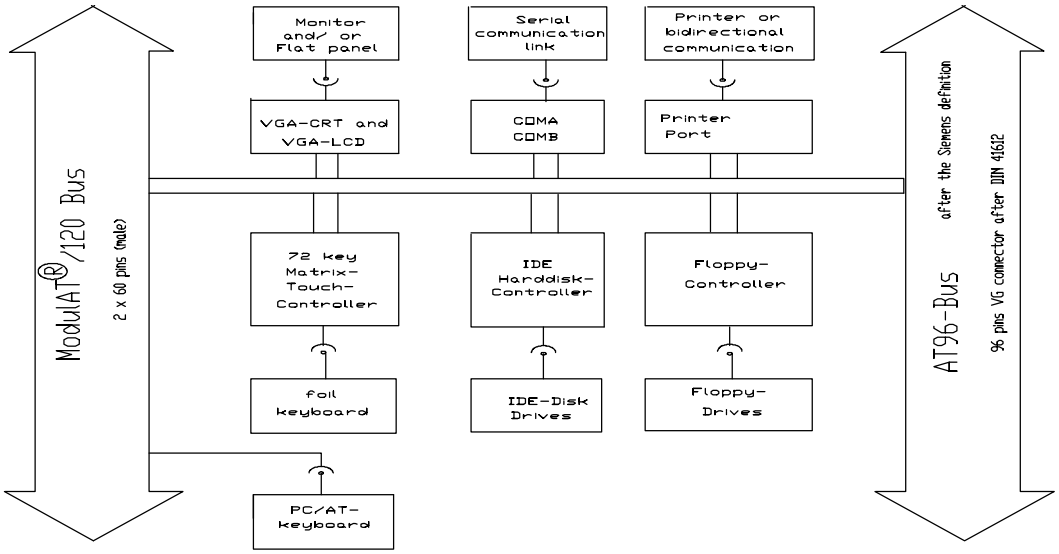
[1] Not available with AT96-MULTI-4K, ISA-VGALCD-4A, ISA-VGALCD-4T

[2] Not available with ISA-VGALCD-4T

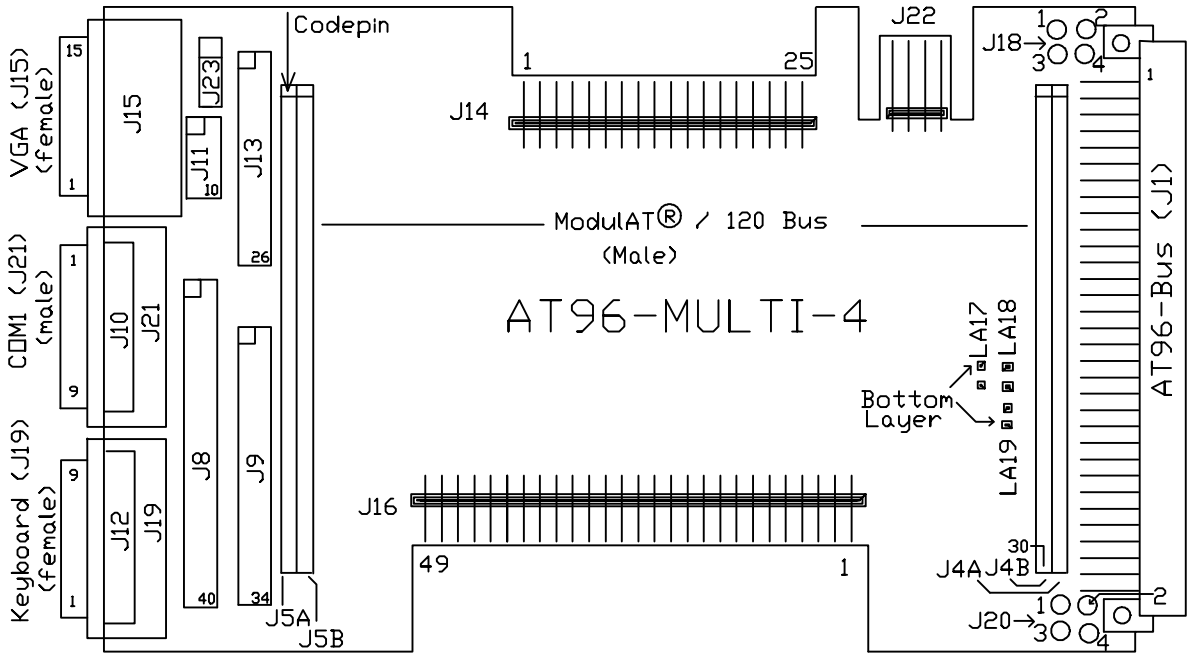
[3] Not available with ISA-VGALCD-4A, ISA-VGALCD-4T, PC/104-VGALCD-4

[4] Not available with ISA-VGALCD-4T

Block Diagram AT96-MULTI-4

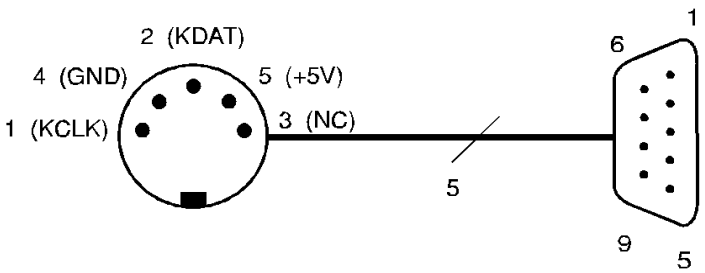


Layout of the Connectors

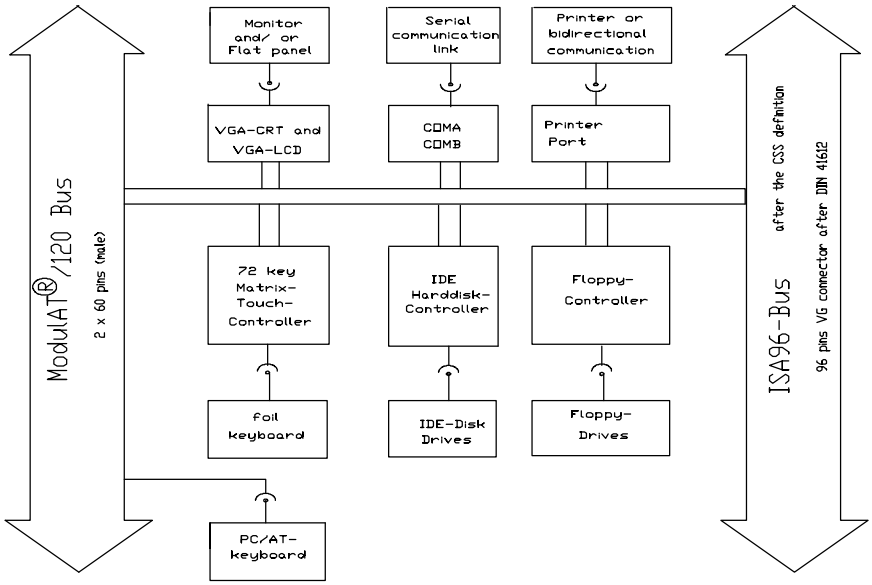


Keyboard Adapter

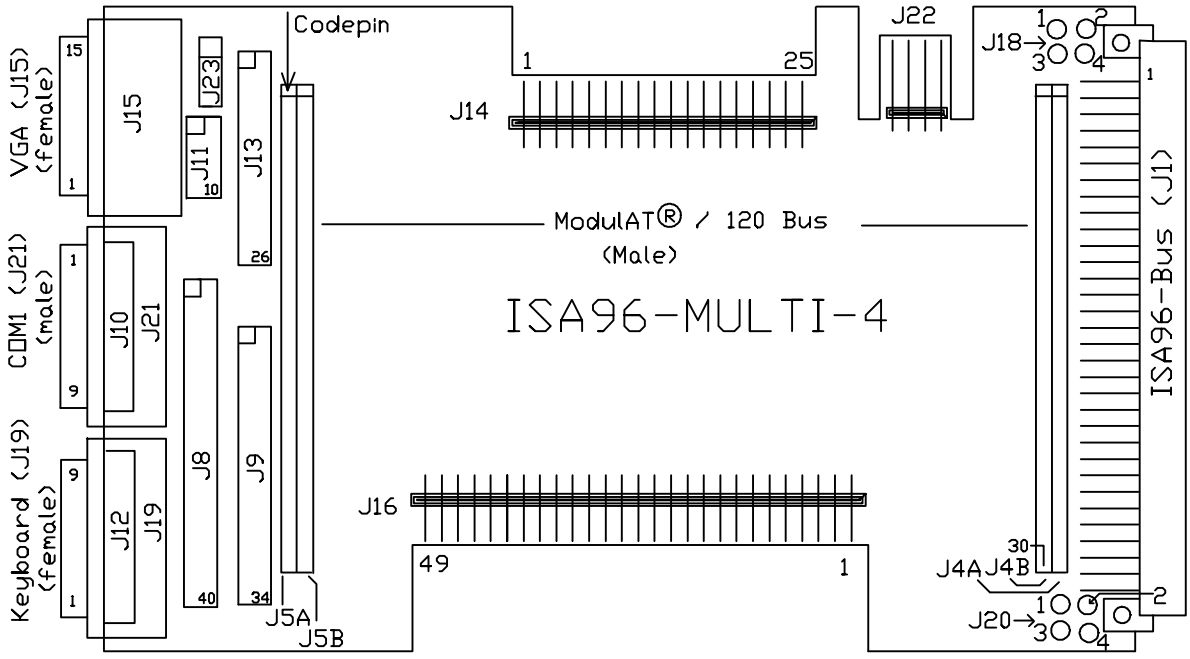
The Keyboard Adapter switches from the 9 pin DSUB connector to the 5 pin Keyboard connector.



Block Diagram ISA96-MULTI-4

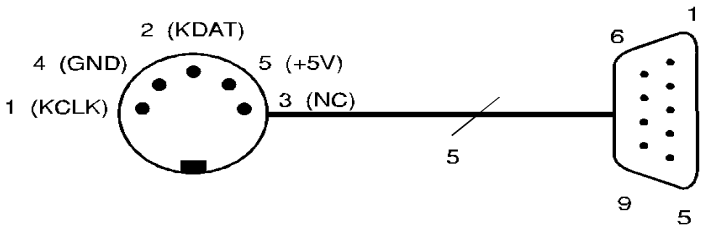


Layout of the Connectors

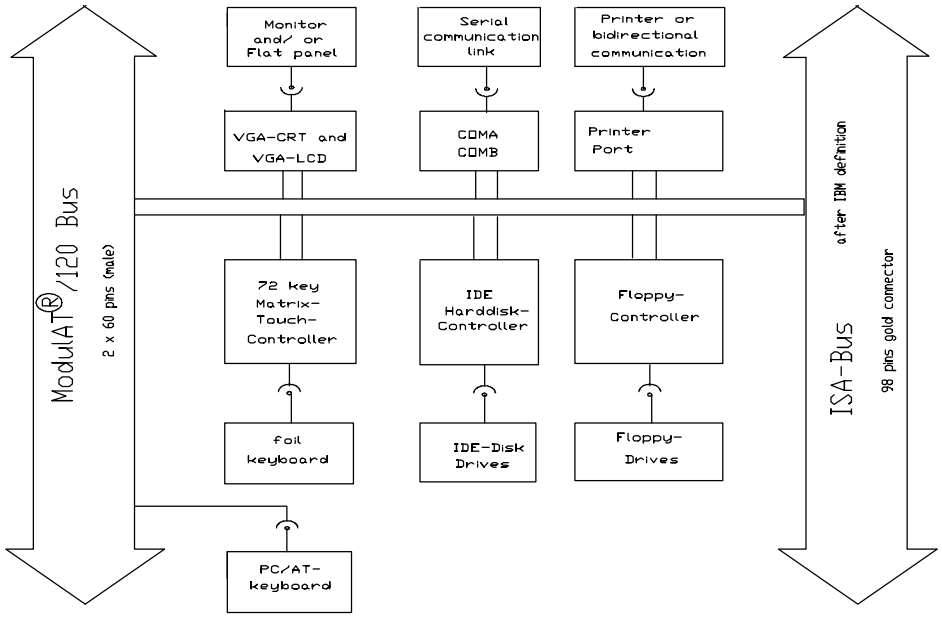


Keyboard Adapter

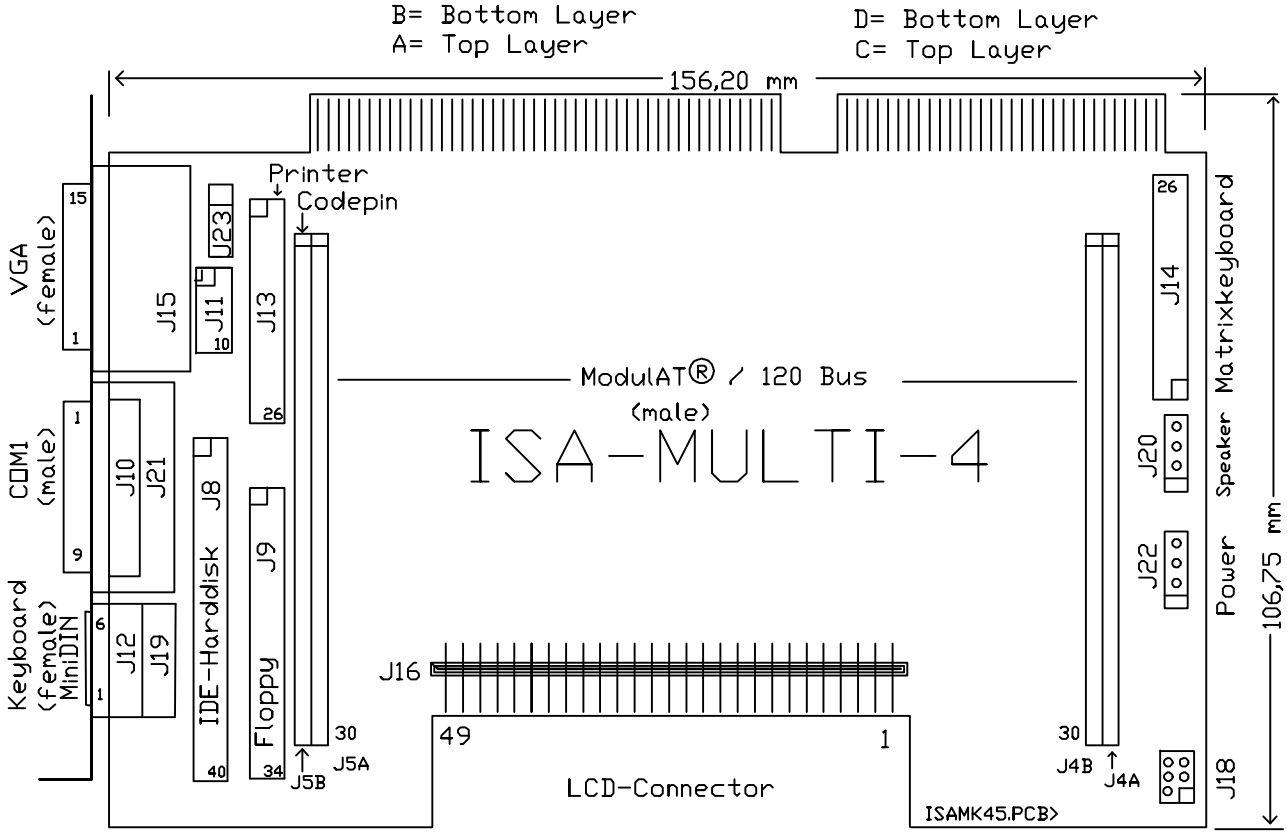
The Keyboard Adapter switches from the 9 pin DSUB connector to the 5 pin Keyboard connector.



Block Diagram ISA-MULTI-4

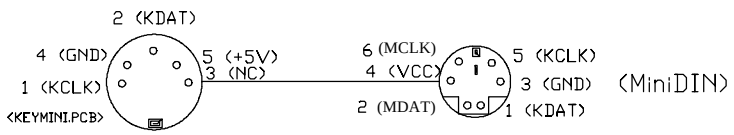


Layout of the Connectors

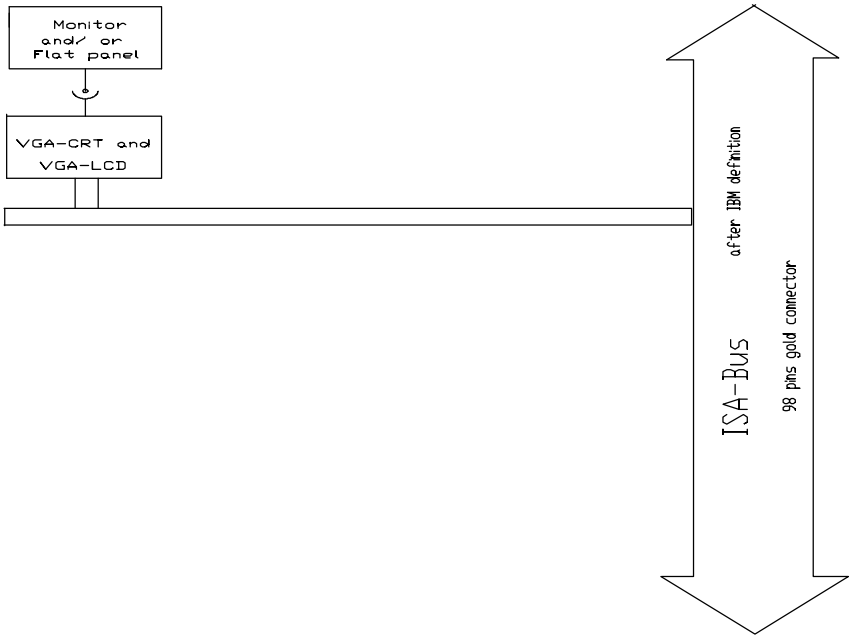


Keyboard Adapter

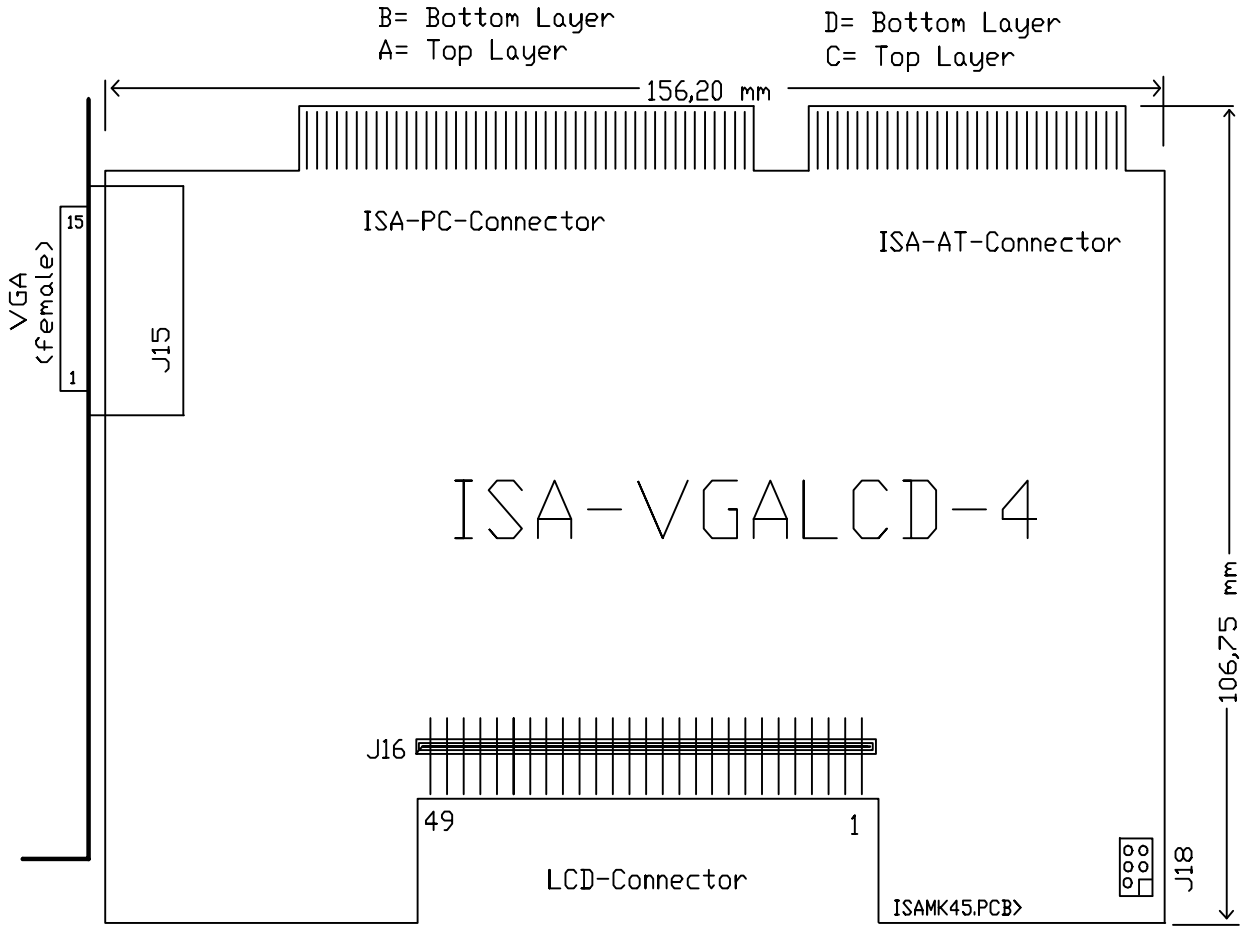
The Keyboard Adapter switches from the 9 pin DSUB connector to the 5 pin Keyboard connector.



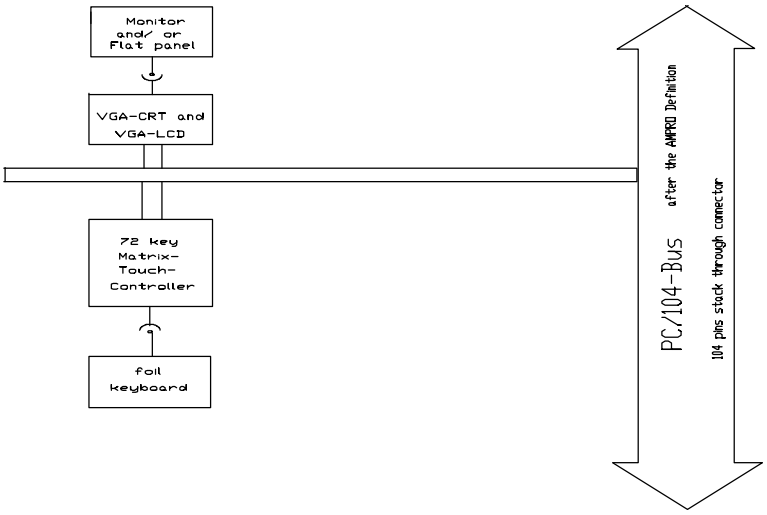
Block Diagram ISA-VGALCD-4



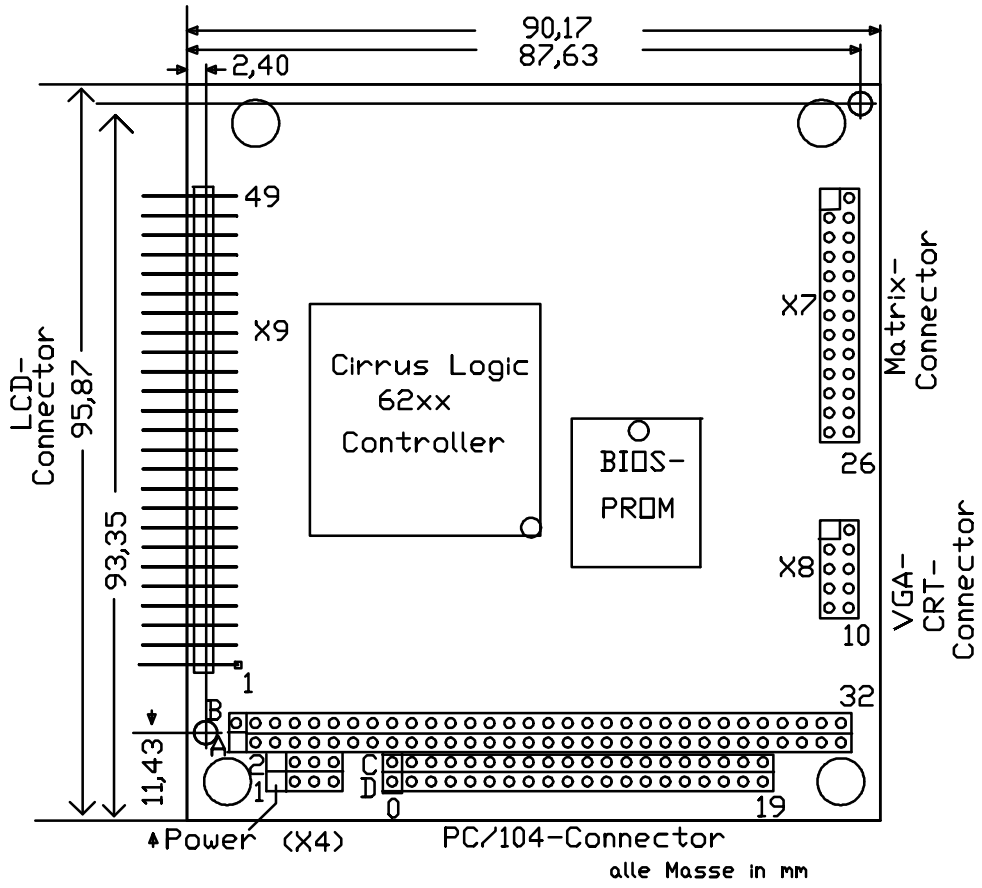
Layout of the Connectors



Block Diagram PC/104-VGALCD-4



Layout of the Connectors



Connector Tables

Connector Table for the AT96-MULTI-4

AT96-BUS				ModulAT###/ 120 Bus				IDE Hard Disk	Floppy	LCD-Connector	Printer	Touch-Matrix	VGA-CRT
Pin	**a (J1)**	**b (J1)**	**c (J1)**	**J4A**	**J4B**	**J5A**	**J5B**	**J8**	**J9**	**J16**	**J13**	**J14**	**J15**
1	GND	/SBHE	/IOCHK	GND	/MASTER	/IOCHK	KEY	/RESET	GND	LFS	/STB	GND	RED
2	RESETDRV	/MEMCS16	SD7	RESETDRV	SD15	SD7	VCC	GND	/FDC RPM	LLCLK	/AFD	Y0	GREEN
3	VCC	SA23	SD6	VCC	SD14	SD6	VCC	IDE D7	GND	VDCLK	PD0	RA0	BLUE
4	IRQ9	/IOCS16	SD5	IRQ9	SD13	SD5	NC	IDE D8	NC	FPVEE	/ERR	Y1	ID2(NC)
5	- (LA17)*	SA22	SD4	/MEMR	SD12	SD4	NC	IDE D6	GND	SW_VDD	PD1	RA1	GND
6	DRQ2	IRQ10	SD3	DRQ2	SD11	SD3	NC	IDE D9	NC	GND	/INIT	Y2	GND
7	-12V	SA21	SD2	/COLOR	SD10	SD2	NC	IDE D5	GND	VEE	PD2	RA2	GND
8	/0WS	IRQ11	SD1	/0WS	SD9	SD1	NC	IDE D10	/FDC IDX	UD0(SLD4G1)	/SLIN	Y3	GND
9	+12V	SA20	SD0	/KLOCK	SD8	SD0	HSPEED	IDE D4	GND	UD1(SLD5G2)	PD3	RA3	KEY(NC)
10	GND	IRQ12	IOCHRDY	GND	/SBHE	IOCHRDY	NC	IDE D11	/FDC DS0	UD2(SLD6R0)	GND	Y4	GND
11	/SMEMW	VBAT	AEN	/SMEMW	LA23	AEN	GND	IDE D3	GND	UD3(SLD7R1)	PD4	RA4	ID0(NC)
12	/SMEMR	IRQ15	SA19	/SMEMR	LA22	SA19	GND	IDE D12	/FDC DS1	LD0(SLD0B0)	GND	Y5	ID1(NC)
13	/IOW	- (LA19)*	SA18	/IOW	LA21	SA18	GND	IDE D2	GND	LD1(SLD1B1)	PD5	RB0	HSYNC
14	/IOR	IRQ14	SA17	/IOR	LA20	SA17	/MEMW	IDE D13	/FDC DS2	LD2(SLD2B2)	GND	Y6	VSYNC
15	/DACK3	- (LA18)*	SA16	/DACK3	LA19	SA16	IRQ10	IDE D1	GND	LD3(SLD3G0)	PD6	RB1	RSVD(NC)
16	DRQ3	/DACK0	SA15	DRQ3	LA18	SA15	IRQ11	IDE D14	/FDC MON	SUD7(R3)	GND	Y7	
17	/DACK1	/MEMR	SA14	/DACK1	LA17	SA14	SA0	IDE D0	GND	SUD6(R2)	PD7	RB2	
18	DRQ1	DRQ0	SA13	DRQ1	/DACK7	SA13	PGOOD	IDE D15	/FDC DIR	SUD5(G5)	GND	/MCLR	
19	/REFRESH	/MEMW	SA12	/REFRESH	DRQ7	SA12	NC	GND	GND	SUD4(G4)	/ACK	RB3	
20	SYSCLK	/DACK5	SA11	SYSCLK	/DACK6	SA11	SPKR	NC	/FDCSTEP	SUD3(G3)	GND	OC0	
21	IRQ7	SD8	SA10	IRQ7	DRQ6	SA10	VCC	NC	GND	SUD2(B5)	/BUSY	RB4	
22	IRQ6	DRQ5	SA9	IRQ6	/DACK5	SA9	NC	GND	/FDC WD	SUD1(B4)	GND	OC1	
23	IRQ5	SD9	SA8	IRQ5	DRQ5	SA8	KDAT	/IOW	GND	SUD0(B3)	PE	RB5	
24	IRQ4	/DACK6	SA7	IRQ4	/DACK0	SA7	KCLK	GND	/FDC WG	R5	GND	RB7	
25	IRQ3	D10	SA6	IRQ3	DRQ0	SA6	VBAT	/IOR	GND	R4	/SLCT	RB6	
26	/DACK2	DRQ6	SA5	DACK2/	MEMCS16/	SA5	XT/AT	GND	/FDC TRO0	GND	GND	VCC	
27	T/C	SD11	SA4	T/C	/IOCS16	SA4	GND	NC	GND	DE			
28	BALE	SD12	SA3	BALE	IRQ15	SA3	GND	BALE	/FDC WP	FPVDD			
29	VCC	SD13	SA2	VCC	IRQ14	SA2	GND	NC	GND	MOD			
30	OSC	SD14	SA1	OSC	IRQ12	SA1	GND	GND	/FDC RD	FPBACK			
31	GND	SD15	SA0					IRQ14	GND	+3,3V			
32	DRQ7	/MASTER	/DACK7					/IOCS16	/FDC SIDE	VEE			
33								SA1	GND	GND			
34								NC	/DCHNG	RFB			
35								SA0		GND			
36								SA2		GPI			
37								/IDE CS0		VDD_SRC			
38								/IDE CS1		VBB			
39								IDELED		+12V			
40								NC		HSYNC			
41										+5V(VCC)			
42										BACK_SRC			
43										VSYNC			
44										SW_BACK			
45										EXT_CTRL			
46										EXT_ADJ			
47										GND			
48										PS3			
49										PS2			
50										PS1			

*LA17-19 can connect to the AT96-Bus over closed SMD-solder jumper (look to the above point "Layout of the connectors")

Keyboard		Keyboard (Option)	COMA	COMA (Option)	COMB	Option1 Configuration I/O-Addresses	Option2	Battery	3,5" Power Connector
Pin	**J19**	**J12**	**J21**	**J10**	**J11**	**J18**	**J20**	**J23**	**J22**
1	PowGood	VCC	/RLSDA	/RLSDA	/RLSDB	1-2 open=C190h	PowGood	VBAT	VCC
2	SPKR	PowGood	SINA	/DSRA	/DSRB	1-2 close=C390h	GND	NC	GND
3	KCLK	GND	SOUTA	SINA	SINB	HDLED	SPKR	GND	GND
4	/KLOCK	SPKR	/DTRA	/RTSA	/RTSB	VCC	VCC	GND	+12V
5	MDAT	KDAT	GND	SOUTA	SOUTB				
6	VCC	KCLK	/DSRA	/CTSA	/CTSB				
7	GND	MDAT	7RTSA	/DTRA	/DTRB				
8	KDAT	/KLOCK	7CTSA	/RIA	/RIB				
9	MCLK	NC	7RIA	GND	GND				
10		MCLK		VCC	VCC				

The option-connectors J12 and J10 can be used alternatively to the connectors J19 and J21.

Connector Table for the ISA96-MULTI-4

ISA96 Bus				ModulAT###/ 120 Bus				IDE Hard Disk	Floppy	LCD-Connector	Printer	Touch-Matrix	VGA-CRT
Pin	**a (J1)**	**b (J1)**	**c (J1)**	**J4A**	**J4B**	**J5A**	**J5B**	**J8**	**J9**	**J16**	**J13**	**J14**	**J15**
1	GND	/MASTER	/IOCHK	GND	/MASTER	/IOCHK	KEY	/RESET	GND	LFS	/STB	GND	RED
2	RESET	SD15	SD7	RESETDRV	SD15	SD7	VCC	GND	/FDC RPM	LLCLK	/AFD	Y0	GREEN
3	VCC	SD14	SD6	VCC	SD14	SD6	VCC	IDE D7	GND	VDCLK	PD0	RA0	BLUE
4	IRQ9	SD13	SD5	IRQ9	SD13	SD5	NC	IDE D8	NC	FPVEE	/ERR	Y1	ID2(NC)
5	/MEMR	SD12	SD4	/MEMR	SD12	SD4	NC	IDE D6	GND	SW_VDD	PD1	RA1	GND
6	DRQ2	SD11	SD3	DRQ2	SD11	SD3	NC	IDE D9	NC	GND	/INIT	Y2	GND
7	-12V	SD10	SD2	/COLOR	SD10	SD2	NC	IDE D5	GND	VEE	PD2	RA2	GND
8	/0WS	SD7	SD1	/0WS	SD9	SD1	NC	IDE D10	/FDC IDX	UD0(SLD4G1)	/SLIN	Y3	GND
9	+12V	SD8	SD0	/KLOCK	SD8	SD0	HSPEED	IDE D4	GND	UD1(SLD5G2)	PD3	RA3	KEY(NC)
10	GND	/SBHE	IOCHRDY	GND	/SBHE	IOCHRDY	NC	IDE D11	/FDC DS0	UD2(SLD6R0)	GND	Y4	GND
11	/SMEMW	LA23	AEN	/SMEMW	LA23	AEN	GND	IDE D3	GND	UD3(SLD7R1)	PD4	RA4	ID0(NC)
12	/SMEMR	LA22	SA19	/SMEMR	LA22	SA19	GND	IDE D12	/FDC DS1	LD0(SLD0B0)	GND	Y5	ID1(NC)
13	/IOW	LA21	SA18	/IOW	LA21	SA18	GND	IDE D2	GND	LD1(SLD1B1)	PD5	RB0	HSYNC
14	/IOR	LA20	SA17	/IOR	LA20	SA17	/MEMW	IDE D13	/FDC DS2	LD2(SLD2B2)	GND	Y6	VSYNC
15	/DACK3	LA19	SA16	/DACK3	LA19	SA16	IRQ10	IDE D1	GND	LD3(SLD3G0)	PD6	RB1	RSVD(NC)
16	DRQ3	LA18	SA15	DRQ3	LA18	SA15	IRQ11	IDE D14	/FDC MON	SUD7(R3)	GND	Y7	
17	/DACK1	LA17	SA14	/DACK1	LA17	SA14	SA0	IDE D0	GND	SUD6(R2)	PD7	RB2	
18	DRQ1	/DACK7	SA13	DRQ1	/DACK7	SA13	PGOOD	IDE D15	/FDC DIR	SUD5(G5)	GND	/MCLR	
19	/REF	DRQ7	SA12	/REFRESH	DRQ7	SA12	NC	GND	GND	SUD4(G4)	/ACK	RB3	
20	SYSCLK	/DACK6	SA11	SYSCLK	/DACK6	SA11	SPKR	NC	/FDCSTEP	SUD3(G3)	GND	OC0	
21	IRQ7	DRQ6	SA10	IRQ7	DRQ6	SA10	VCC	NC	GND	SUD2(B5)	/BUSY	RB4	
22	IRQ6	/DACK5	SA9	IRQ6	/DACK5	SA9	NC	GND	/FDC WD	SUD1(B4)	GND	OC1	
23	IRQ5	DRQ5	SA8	IRQ5	DRQ5	SA8	KDAT	/IOW	GND	SUD0(B3)	PE	RB5	
24	IRQ4	/DACK0	SA7	IRQ4	/DACK0	SA7	KCLK	GND	/FDC WG	R5	GND	RB7	
25	IRQ3	DRQ0	SA6	IRQ3	DRQ0	SA6	VBAT	/IOR	GND	R4	/SLCT	RB6	
26	/DACK2	/MemCS16	SA5	DACK2/	MEMCS16/	SA5	XT/AT	GND	/FDC TRO0	GND	GND	VCC	
27	T/C	/IOCS16	SA4	T/C	/IOCS16	SA4	GND	NC	GND	DE			
28	BALE	IRQ15	SA3	BALE	IRQ15	SA3	GND	BALE	/FDC WP	FPVDD			
29	VCC	IRQ14	SA2	VCC	IRQ14	SA2	GND	NC	GND	MOD			
30	OSC	IRQ12	SA1	OSC	IRQ12	SA1	GND	GND	/FDC RD	FPBACK			
31	/MEMW	IRQ11	SA0					IRQ14	GND	+3,3V			
32	GND	IRQ10	GND					/IOCS16	/FDC SIDE	VEE			
33								SA1	GND	GND			
34								NC	/DCHNG	RFB			
35								SA0		GND			
36								SA2		GPI			
37								/IDE CS0		VDD_SRC			
38								/IDE CS1		VBB			
39								IDELED		+12V			
40								NC		HSYNC			
41										+5V(VCC)			
42										BACK_SRC			
43										VSYNC			
44										SW_BACK			
45										EXT_CTRL			
46										EXT_ADJ			
47										GND			
48										PS3			
49										PS2			
50										PS1			

Keyboard		Keyboard (Option)	COMA	COMA (Option)	COMB	Option1 Configuration I/O-Addresses	Option2	Battery	3,5" Power Connector
Pin	**J19**	**J12**	**J21**	**J10**	**J11**	**J18**	**J20**	**J23**	**J22**
1	PowGood	VCC	/RLSDA	/RLSDA	/RLSDB	1-2 open=C190h	PowGood	VBAT	VCC
2	SPKR	PowGood	SINA	/DSRA	/DSRB	1-2 close=C390h	GND	NC	GND
3	KCLK	GND	SOUTA	SINA	SINB	HDLED	SPKR	GND	GND
4	/KLOCK	SPKR	/DTRA	/RTSA	/RTSB	VCC	VCC	GND	+12V
5	MDAT	KDAT	GND	SOUTA	SOUTB				
6	VCC	KCLK	/DSRA	/CTSA	/CTSB				
7	GND	MDAT	7RTSA	/DTRA	/DTRB				
8	KDAT	/KLOCK	7CTSA	/RIA	/RIB				
9	MCLK	NC	7RIA	GND	GND				
10		MCLK		VCC	VCC				

The option-connectors J12 and J10 can alternatively used to the connectors J19 and J21.

Connector Table for the ISA-MULTI-4

ISA-Bus					ModulAT###/ 120 Bus				IDE Hard Disk	Floppy	LCD-Connector	Printer	Touch-Matrix	VGA-CRT
Pin	**A**	**B**	**C**	**D**	**J4A**	**J4B**	**J5A**	**J5B**	**J8**	**J9**	**J16**	**J13**	**J14**	**J15**
1	/IOCHCHK	GND	/SBHE	/MEMCS16	GND	/MASTER	/IOCHK	KEY	/RESET	GND	LFS	/STB	GND	RED
2	SD7	RESETDRV	LA23	/IOCS16	RESETDRV	SD15	SD7	VCC	GND	/FDC RPM	LLCLK	/AFD	Y0	GREEN
3	SD6	Vcc	LA22	IRQ10	VCC	SD14	SD6	VCC	IDE D7	GND	VDCLK	PD0	RA0	BLUE
4	SD5	IRQ9	LA21	IRQ11	IRQ9	SD13	SD5	NC	IDE D8	NC	FPVEE	/ERR	Y1	ID2(NC)
5	SD4	-5V	LA20	IRQ12	/MEMR	SD12	SD4	NC	IDE D6	GND	SW_VDD	PD1	RA1	GND
6	SD3	DRQ2	LA19	IRQ15	DRQ2	SD11	SD3	NC	IDE D9	NC	GND	/INIT	Y2	GND
7	SD2	-12V	LA18	IRQ14	/COLOR	SD10	SD2	NC	IDE D5	GND	VEE	PD2	RA2	GND
8	SD1	/ENDXFR	LA17	/DACK0	/0WS	SD9	SD1	NC	IDE D10	/FDC IDX	UD0(SLD4G1)	/SLIN	Y3	GND
9	SD0	+12V	/MEMR	DRQ0	/KLOCK	SD8	SD0	HSPEED	IDE D4	GND	UD1(SLD5G2)	PD3	RA3	KEY(NC)
10	IOCHRDY	GND	/MEMW	/DACK5	GND	/SBHE	IOCHRDY	NC	IDE D11	/FDC DS0	UD2(SLD6R0)	GND	Y4	GND
11	AEN	/SMEMW	SD8	DRQ5	/SMEMW	LA23	AEN	GND	IDE D3	GND	UD3(SLD7R1)	PD4	RA4	ID0(NC)
12	SA19	/SMEMR	SD9	/DACK6	/SMEMR	LA22	SA19	GND	IDE D12	/FDC DS1	LD0(SLD0B0)	GND	Y5	ID1(NC)
13	SA18	/IOW	SD10	DRQ6	/IOW	LA21	SA18	GND	IDE D2	GND	LD1(SLD1B1)	PD5	RB0	HSYNC
14	SA17	/IOR	SD11	/DACK7	/IOR	LA20	SA17	/MEMW	IDE D13	/FDC DS2	LD2(SLD2B2)	GND	Y6	VSYNC
15	SA16	/DACK3	SD12	DRQ7	/DACK3	LA19	SA16	IRQ10	IDE D1	GND	LD3(SLD3G0)	PD6	RB1	RSVD(NC)
16	SA15	DRQ3	SD13	Vcc	DRQ3	LA18	SA15	IRQ11	IDE D14	/FDC MON	SUD7(R3)	GND	Y7	
17	SA14	/DACK1	SD14	/MASTER	/DACK1	LA17	SA14	SA0	IDE D0	GND	SUD6(R2)	PD7	RB2	
18	SA13	DRQ1	SD15	GND	DRQ1	/DACK7	SA13	PGOOD	IDE D15	/FDC DIR	SUD5(G5)	GND	/MCLR	
19	SA12	/REFRESH			/REFRESH	DRQ7	SA12	NC	GND	GND	SUD4(G4)	/ACK	RB3	
20	SA11	SYSCLK			SYSCLK	/DACK6	SA11	SPKR	NC	/FDCSTEP	SUD3(G3)	GND	OC0	
21	SA10	IRQ7			IRQ7	DRQ6	SA10	VCC	NC	GND	SUD2(B5)	/BUSY	RB4	
22	SA9	IRQ6			IRQ6	/DACK5	SA9	NC	GND	/FDC WD	SUD1(B4)	GND	OC1	
23	SA8	IRQ5			IRQ5	DRQ5	SA8	KDAT	/IOW	GND	SUD0(B3)	PE	RB5	
24	SA7	IRQ4			IRQ4	/DACK0	SA7	KCLK	GND	/FDC WG	R5	GND	RB7	
25	SA6	IRQ3			IRQ3	DRQ0	SA6	VBAT	/IOR	GND	R4	/SLCT	RB6	
26	SA5	/DACK2			DACK2/	MEMCS16/	SA5	XT/AT	GND	/FDC TRO0	GND	GND	VCC	
27	SA4	T/C			T/C	/IOCS16	SA4	GND	NC	GND	DE			
28	SA3	BALE			BALE	IRQ15	SA3	GND	BALE	/FDC WP	FPVDD			
29	SA2	Vcc			VCC	IRQ14	SA2	GND	NC	GND	MOD			
30	SA1	OSC			OSC	IRQ12	SA1	GND	GND	/FDC RD	FPBACK			
31	SA0	GND							IRQ14	GND	+3,3V			
32									/IOCS16	/FDC SIDE	VEE			
33									SA1	GND	GND			
34									NC	/DCHNG	RFB			
35									SA0		GND			
36									SA2		GPI			
37									/IDE CS0		VDD_SRC			
38									/IDE CS1		VBB			
39									IDELED		+12V			
40									NC		HSYNC			
41											+5V(VCC)			
42											BACK_SRC			
43											VSYNC			
44											SW_BACK			
45											EXT_CTRL			
46											EXT_ADJ			
47											GND			
48											PS3			
49											PS2			
50											PS1			

	Keyboard MiniDIN	Keyboard DSUB9	COM1	COM1 (Option)	COM2	Option1	Speaker	Battery	3,5" Power Connector
Pin	**J19**	**J12**	**J21**	**J10**	**J11**	**J18**	**J20**	**J23**	**J22**
1	KDAT	PowGood	/RLSDA	/RLSDA	/RLSDB	1-2 open=C190h	Speaker	VBAT	VCC
2	MDAT	SPKR	SINA	/DSRA	/DSRB	1-2 close=C390h	NC	NC	GND
3	GND (Coil)	KCLK	SOUTA	SINA	SINB	HDLED	NC	NC	GND
4	VCC (Coil)	/KLOCK	/DTRA	/RTSA	/RTSB	VCC	VCC	GND	+12V
5	KCLK	MCLK	GND	SOUTA	SOUTB	PowrGood			
6	MCLK	VCC (Coil)	/DSRA	/CTSA	/CTSB	GND			
7		GND (Coil)	7RTSA	/DTRA	/DTRB				
8		KDAT	7CTSA	/RIA	/RIB				
9		MDAT	7RIA	GND	GND				
10				VCC	VCC				

Connector Table for the ISA-VGALCD-4

ISA-Bus					LCD-Connector	VGA-CRT
Pin	**A**	**B**	**C**	**D**	**J16**	**J15**
1	/IOCHCHK	GND	/SBHE	/MEMCS16	LFS	RED
2	SD7	RESETDRV	LA23	/IOCS16	LLCLK	GREEN
3	SD6	Vcc	LA22	IRQ10	VDCLK	BLUE
4	SD5	IRQ9	LA21	IRQ11	FPVEE	ID2(NC)
5	SD4	-5V	LA20	IRQ12	SW_VDD	GND
6	SD3	DRQ2	LA19	IRQ15	GND	GND
7	SD2	-12V	LA18	IRQ14	VEE	GND
8	SD1	/ENDXFR	LA17	/DACK0	UD0(SLD4G1)	GND
9	SD0	+12V	/MEMR	DRQ0	UD1(SLD5G2)	KEY(NC)
10	IOCHRDY	GND	/MEMW	/DACK5	UD2(SLD6R0)	GND
11	AEN	/SMEMW	SD8	DRQ5	UD3(SLD7R1)	ID0(NC)
12	SA19	/SMEMR	SD9	/DACK6	LD0(SLD0B0)	ID1(NC)
13	SA18	/IOW	SD10	DRQ6	LD1(SLD1B1)	HSYNC
14	SA17	/IOR	SD11	/DACK7	LD2(SLD2B2)	VSYNC
15	SA16	/DACK3	SD12	DRQ7	LD3(SLD3G0)	RSVD(NC)
16	SA15	DRQ3	SD13	Vcc	SUD7(R3)	
17	SA14	/DACK1	SD14	/MASTER	SUD6(R2)	
18	SA13	DRQ1	SD15	GND	SUD5(G5)	
19	SA12	/REFRESH			SUD4(G4)	
20	SA11	SYSCLK			SUD3(G3)	
21	SA10	IRQ7			SUD2(B5)	
22	SA9	IRQ6			SUD1(B4)	
23	SA8	IRQ5			SUD0(B3)	
24	SA7	IRQ4			R5	
25	SA6	IRQ3			R4	
26	SA5	/DACK2			GND	
27	SA4	T/C			DE	
28	SA3	BALE			FPVDD	
29	SA2	Vcc			MOD	
30	SA1	OSC			FPBACK	
31	SA0	GND			+3,3V	
32					VEE	
33					GND	
34					RFB	
35					GND	
36					GPI	
37					VDD_SRC	
38					VBB	
39					+12V	
40					HSYNC	
41					+5V(VCC)	
42					BACK_SRC	
43					VSYNC	
44					SW_BACK	
45					EXT_CTRL	
46					EXT_ADJ	
47					GND	
48					PS3	
49					PS2	
50					PS1	

Option1 Configuration I/O-Addresses	
Pin	**J18**
1	1-2 open=C190h
2	1-2 close=C390h
3	HDLED
4	VCC
5	PowrGood
6	GND
7	
8	
9	
10	

Connector Table for the PC/104-VGALCD-4

PC/104 - Bus					Power	LCD-Connector	Touch-Matrix	VGA-CRT
Pin	**A**	**B**	**C**	**D**	**X4**	**X9**	**X7**	**X8**
0			GND	GND				
1	/IOCHCK	GND	/SBHE	/MEMCS16	GND	LFS	GND	RED
2	SD7	RESETDRV	LA23	/IOCS16	Vcc	LLCLK	Y0	GND
3	SD6	VCC	LA22	IRQ10	Key pin	VDCLK	RA0	GREEN
4	SD5	IRQ9	LA21	IRQ11	+12V	FPVEE	Y1	GND
5	SD4	-5V	LA20	IRQ12	-5V	SW_VDD	RA1	BLUE
6	SD3	DRQ2	LA19	IRQ15	-12V	GND	Y2	GND
7	SD2	-12V	LA18	IRQ14	GND	VEE	RA2	V-SYNC
8	SD1	/0WS	LA17	/DACK0	Vcc	UD0(SLD4G1)	Y3	GND
9	SD0	+12V	/MEMR	DRQ0		UD1(SLD5G2)	RA3	H-SYNC
10	IOCHRDY	GND	/MEMW	/DACK5		UD2(SLD6R0)	Y4	GND
11	AEN	/SMEMW	SD8	DRQ5		UD3(SLD7R1)	RA4	
12	SA19	/SMEMR	SD9	/DACK6		LD0(SLD0B0)	Y5	
13	SA18	/IOW	SD10	DRQ6		LD1(SLD1B1)	RB0	
14	SA17	/IOR	SD11	/DACK7		LD2(SLD2B2)	Y6	
15	SA16	/DACK3	SD12	DRQ7		LD3(SLD3G0)	RB1	
16	SA15	DRQ3	SD13	VCC		SUD7(R3)	Y7	
17	SA14	/DACK1	SD14	/MASTER		SUD6(R2)	RB2	
18	SA13	DRQ1	SD15	GND		SUD5(G5)	/MCLR	
19	SA12	/REFRESH	GND	GND		SUD4(G4)	RB3	
20	SA11	SYSCLK				SUD3(G3)	OC0	
21	SA10	IRQ7				SUD2(B5)	RB4	
22	SA9	IRQ6				SUD1(B4)	OC1	
23	SA8	IRQ5				SUD0(B3)	RB5	
24	SA7	IRQ4				R5	RB7	
25	SA6	IRQ3				R4	RB6	
26	SA5	/DACK2				GND	VCC	
27	SA4	T/C				DE		
28	SA3	BALE				FPVDD		
29	SA2	VCC				MOD		
30	SA1	OSC				FPBACK		
31	SA0	GND				+3,3V		
32	GND	GND				VEE		
33						GND		
34						RFB		
35						GND		
36						GPI		
37						VDD_SRC		
38						VBB		
39						+12V		
40						HSYNC		
41						+5V(VCC)		
42						BACK_SRC		
43						VSYNC		
44						SW_BACK		
45						EXT_CTRL		
46						EXT_ADJ		
47						GND		
48						PS3		
49						PS2		
50						PS1		

General Description

The MULTI-4 Cards add the functions and features of the latest video standards to your PC computer system. The VGA-BIOS is included in a 64kB PROM (32kB VGA-BIOS,16kB Extension-BIOS, 16KB unused). The PROM is suited on the MULTI-4 board. The BIOS address range is C000:0000H up to C000:BFFFH. The MULTI-4 is designed to control LCD displays of any technology up to 640 * 480 resolution. The CRT resolution goes up to 1024 * 768, however LCD displays with more than 640 * 480 resolution are not supported. There is no interface standard for LCD displays, therefore, some unusual LCDs may require some external hardware to connect to the MULTI-4.

Displays with resolution smaller than 640*480 are generally supported, but may require additional software drivers to be PC compatible.

Supported Display Types

The MULTI-4 can support a vast variety of panels all configured by a dedicated cable for each panel. There is no Jumper setting or software setup required. Since practically all LCDs have different connectors, pinout or LCD voltages, this is the easiest and safest way to configure for different panels.

We plan to update the display type list in the manual, but ask for currently supported panel types, since new panels will be added frequently.

Possible Configurations

The MULTI-4 card can be used as following:

1. Base card (AT96-MULTI-4, ISA96-MULTI-4, ISA-MULTI-4) when a ModulAT###/CPU is plugged in,

   All the connectors are available for the user.

2. When on the three above base cards no ModulAT###/CPU is plugged in, there are three restrictions:

   The keyboard connectors J19 with the keyboard, mouse and speaker signal are not supplied. Only the PowerGood signal of the keyboard connectors (J19) are supplied.

   The PowGood/Reset connector (J20 (pin1-2)) is not supplied.

3. The PC/104-VGALCD-4 can be used as a VGALCD controller board. The CPU -Module can be the MOPS from JUMP or another CPU-board from another company.

The I/O Interface

The I/O -functions are supported on the AT96-MULTI-4, ISA96-MULTI-4 and ISA-MULTI-4. The other MULTI-4 cards as ISA-VGALCD-4 or PC/104-VGALCD-4 don't support these I/O parts.

The I/O-functions are realized through the I/O-controller FDC37C651 from SMC. The FDC37C651 is housed in 100 pin PQFP package.

The I/O-functions of the FDC37C651 are:

Floppy-Interface
The floppy interface is realized within the FDC37C651 controller. The interface supports two 3 1/2 inch floppy drives (1,44 M byte).

IDE-Hard-Disk-Interface
The IDE interface is realized within the FDC37C651 controller. The interface supports all the standard IDE-Disk-drives (AT-disk drives).

Serial Interface
The serial interface is realized within the FDC37C651 controller. The interface supports two serial ports COMA and COMB. Via V.24 drivers the COM-signals are driven with the V.24 level.

The V.24 levels are generated onboard with a DC/DC-converter, so only a +5 volt supply is necessary.

Parallel Interface
The printer interface is realized within the FDC37C651 controller. The interface is bi-directional and supports one printer interface.

Keyboard-Connector
The keyboard interface is realized on the CPU-Modul. So where the CPU module is plugged in, there the keyboard function is available.

ModulAT®/120 bus

The ModulAT/® 120 bus is available on following base boards:

AT96-MULTI-4

ISA96-MULTI-4

ISA-MULTI-4

The module bus consist of two 60 pin male connectors. Via this connector, a ModulAT###/CPU can be added to the AT96-MULTI-4. The 120 pin connector contains all AT-standard signals and additional power signals. When the ModulAT###/CPU type XT (ModulAT###/88-1) is connected, of course only the XT-signals are active.

Systembus (AT96-bus, ISA96-bus, ISA-bus, PC/104-bus)

The following system buses are available as expansion busses on the MULTI-4 cards:

AT96-MULTI-4	AT96- bus
ISA96-MULTI-4	ISA96- bus
ISA-MULTI-4	ISA- bus
ISA-VGALCD-4	ISA- bus
PC/104-VGALCD-4	PC/104- bus

The **AT96** Bus is defined by the Siemens company. The AT96 bus is electrically identical to the standard AT-Bus (ISA-Bus) and functionally full compatible to the AT. The VG-96 pin connector (DIN 41612) allows a better mechanical connector security.

The **ISA96** Bus is defined by the CSS company. The ISA96 bus is electrically identical to the standard AT-Bus (ISA-Bus) and functionally full compatible to the AT. The VG-96 pin connector (DIN 41612) allows a better mechanical connector security.

The **ISA** Bus is defined by the IBM company. The ISA- bus is the standard PC-bus.

The **PC/104** bus is defined by the AMPRO company. The PC/104 bus is electrically identical to the standard AT-Bus (ISA-Bus) and functionally full compatible to the AT. The driver power is 12mA instead of 24mA on the ISA bus.

Power Supply

All MULTI-4 boards only need a +5V power supply, also when a module PC is plugged in. The +5 volt can be supplied through a power connector on board. Also the +5 volt can be connected over the backplanes of the AT96-MULTI-4, ISA96-MULTI-4 and ISA-MULTI-4. The typ. power consumption of the MULTI-4 is about 350 mA.

VGA Monochrome Monitor

Monochrome type monitors use Green Video for all video input and ignore Red Video and Blue Video. Monitor ID Bits are not used by the AT96-MULTI-4. Monitor type is determined on power up by an automatic monitor detection circuit.

I/O-Addresses

Following addresses are occupied from the AT96-MULT-1 I/O hardware:

I/O Address	Register	Description
Special Function Control Register		
C190h-C197h or C390h-C397h	Register	Special Function Control Register (Address range can be selected by a jumper)
FDC-Controller		
3F0H-3F7H	FDC37C651	Floppy-Controller IO-ports
3F0H-3F1H	FDC37C651	Unused
3F2H (W)	FDC37C651	Digital output Register
3F3H	FDC37C651	Unused
3F4H (R)	FDC37C651	Main status register
3F5H (R/W)	FDC37C651	Data register
3F6H	FDC37C651	Unused
3F7H (W)	FDC37C651	Data rate select register (Hard Disk)
3F7H (R)	FDC37C651	Digital input register
IDE Controller		
1F0H (R/W)	FDC37C651	Data register
1F1H (R)	FDC37C651	Error register
1F1H (W)	FDC37C651	Write register
1F2H (R/W)	FDC37C651	Sector count
1F3H (R/W)	FDC37C651	Sector number
1F4H (R/W)	FDC37C651	Cylinder low
1F5H (R/W)	FDC37C651	Cylinder high
1F6H (R/W)	FDC37C651	Drive/Head
1F7H (R)	FDC37C651	Status register
1F7H (W)	FDC37C651	Command register

COMA / COMB (Base is programmable)		
Base+0 (R)	FDC37C651	COM Receive buffer register
Base+0 (W)	FDC37C651	COM Transmit buffer register
Base+1 (R/W)	FDC37C651	COM Interrupt enable register
Base+2 (R/W)	FDC37C651	COM Interrupt flag register
Base+3 (R/W)	FDC37C651	COM Byte format register
Base+4 (R/W)	FDC37C651	COM Modem control register
Base+5 (R/W)	FDC37C651	COM Line status register
Base+6 (R/W)	FDC37C651	COM Modem status register
Base+7 (R/W)	FDC37C651	COM Scratch pad register
Possible Base Addresses for COMA/COMB		**(###)=default**
220H	FDC37C651	Programmable Baseaddr. COM 3
228H	FDC37C651	Programmable Baseaddr. COM 4
238H	FDC37C651	Programmable Baseaddr. COM 4
2E0H	FDC37C651	Programmable Baseaddr. COM 4
2E8H	FDC37C651	Programmable Baseaddr. COM 3/4
2F8H **(•)**	FDC37C651	Programmable Baseaddr. COM 2
338H	FDC37C651	Programmable Baseaddr. COM 3
3E8H	FDC37C651	Programmable Baseaddr. COM 3
3F8H **(•)**	FDC37C651	Programmable Baseaddr. COM 1
Line printer (Base is programmable)		
Base+0	FDC37C651	LPT Data latch port A
Base+1	FDC37C651	LPT Status port B
Base+2	FDC37C651	LPT Control port C
Possible Base Addresses for LPT		**(•)=default**
278H	FDC37C651	Programmable Baseaddress
378H	FDC37C651	Programmable Baseaddress
3BCH **(•)**	FDC37C651	Programmable Baseaddress

Reserved Memory Addresses

For the Interrupt Return Vectors for the Matrix-controller a memory area of 12 bytes is reserved in the memory area 0:2B4h to 2BFh (this are the unused interrupt vectors INT ADh to INT AFh. BIOS Versions prior to 2.3 reserve additional 16 bytes in the range of 40h:F0h to 40h:FFh

Interrupts

Interrupt	Controller	Description
IRQ3	FDC37C651	Serial interface COM2 / COM4
IRQ4	FDC37C651	Serial interface COM1 / COM3
IRQ5	FDC37C651	Line printer interrupt request
IRQ6	FDC37C651	FDC interrupt request
IRQ9	Matrix-Touch-Controller	Matrix-Keyboard-Interrupt
IRQ14	FDC37C651	Hard Disk interrupt request / AT-mode)

Working with the MULTI-4

Overview

The MULTI-4 offers some useful extensions compared with other peripheral modules, which will be described in the following:

Software Configurable I/O-Controller

The on board I/O controller chip offers IDE-Harddisk interface, floppy interface, two serial and one parallel port. The IDE- and floppy interface may be enabled or disabled by software. Any of the two serial ports can be configured as COM1 - COM4 or as additional COM Port with a non-standard address. The parallel port can be configured as LPT1 - LPT3 and used uni-directionally as well as bidirectionally.

Support for LC-Displays

Monochrome and Color STN, TFT or EL Display panels may be connected to the AT96-MULTI-4. The complete display mode selection and configuration is supported by the video BIOS. Additionally, display mode selection (CRT only, panel only, or SimulScan) and the display features "contrast enhancement" and "reverse video" can be controlled by the user via keyboard.

An on board DC-DC converter is capable of generating positive and negative voltages between 19 V and 45V rsp. -14V and -40V which are needed by many LCD panels. The voltage (the panel's contrast) can be controlled by software or by the user via keyboard. Moreover, the converter can be turned on or of.

Decoder for Matrix Keyboard

A matrix keyboard with a 8*9 matrix can easily be connected to the MULTI-4. The matrix is scanned by hardware. If a key is being pressed, it will be processed just like a keystroke from the PC keyboard. The assignment between keys on the matrix keyboard and generated scan codes is completely user configurable. The configuration is stored in EEPROM.

Multi Purpose Open Collector Outputs (OC0, OC1)

Two open collector outputs are available for general purpose usage. Access is given via BIOS calls.

General Purpose Input

One input pin is located on the LCD panel connector, which can be used as an input for any TTL-level signal.

Non-volatile User RAM

16 bytes of non volatile RAM (residing in an EEPROM) can be accessed by the user via simple BIOS calls.

Easy configuration

Configuration data for the features described above can easily be changed by using the program MULTISET. This is a mouse compatible, menu driven software following the SAA standard.

All the settings may also be made by your own software via well documented BIOS calls. The software diskette will be delivered with this technical manual.

Software configurable I/O-Controller

The software configuration features of the on board I/O controller are fully supported by on board BIOS and the MULTISET utility program.

Configuration I/O-Addresses (J18)

The I/O-controller has to be configured over the I/O-address C190h. When there is a I/O-conflict the jumper J18 (pin1-2) has to be closed and the configuration then runs under I/O-address C390h.

Using the MULTISET Utility

General Information

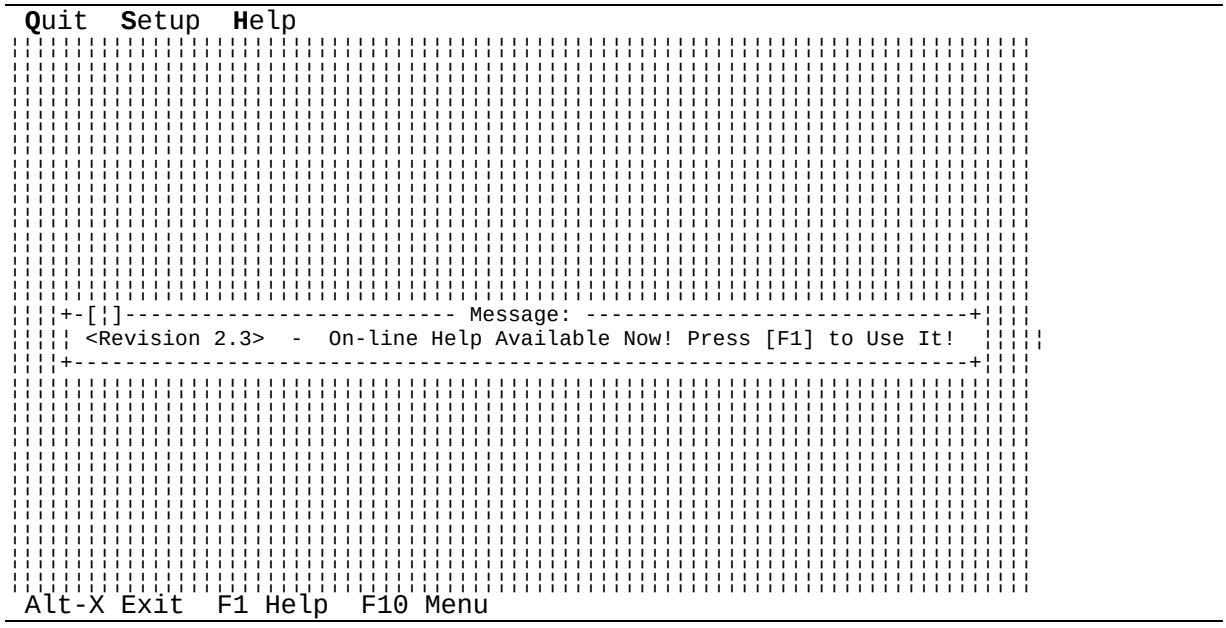
A special setup utility, called MULTISET, is available for the MULTI-4 board. MULTISET may be used to configure the on board IO components, to setup the matrix keyboard decoder, the video sub system and to obtain information about the board. The program's name is MULTISxx.EXE, where xx is the version number. This version number should always be the same as the version of the BIOS. If you use BIOS version 2.3 (labeled MLCDR123), you should also use MULTIS23.EXE.

Please note that currently two versions of the MULTI-4 hardware exist. An older version, which has been delivered during 1994 and a newer version which is being shipped since early 1995. If you need to determine if you have the older or the newer hardware version, just search for a GAL (named PALCE20V8 or labeled with MLCDP111)

Important: The older hardware version may only be used with BIOS MLCDRx1x and MULTIS1x.EXE. The newer hardware version may only be used with BIOS MLCDRx2x and MULTIS2x.EXE. If you mix up older hardware and newer BIOS or vice versa, your system will not work.

Starting MULTISET

From the DOS-Prompt, simply Type <MULTIS23>+<Return> The program starts and welcomes you as shown here:



Basic Concepts

MULTISET follows the SAA standard for user interfaces, which allows easy navigation through the program by using a mouse or the keyboard.

Selecting Menus

- To select a menu, click on the menu's name with your mouse.
- Alternatively, you may select a menu by keyboard. The easiest way to do this is to find the highlighted character in the menu's name (on color systems, this character is red, on monochrome systems, the character may be underlined). Then hold down the <Alt> key and press key with the character you found.
- A second way to select a menu by keyboard is the following: hit the <F10> key to jump to the menu bar, select the menu you need using the <Arrow left><Arrow right> keys, then hit <Arrow down> to pull the menu down.
- If a menu was accidentally selected, you may deselect it by clicking on the empty desktop or by hitting the <Esc> key.

Selecting a Menu Entry

To select and activate a menu entry, do one of the following:

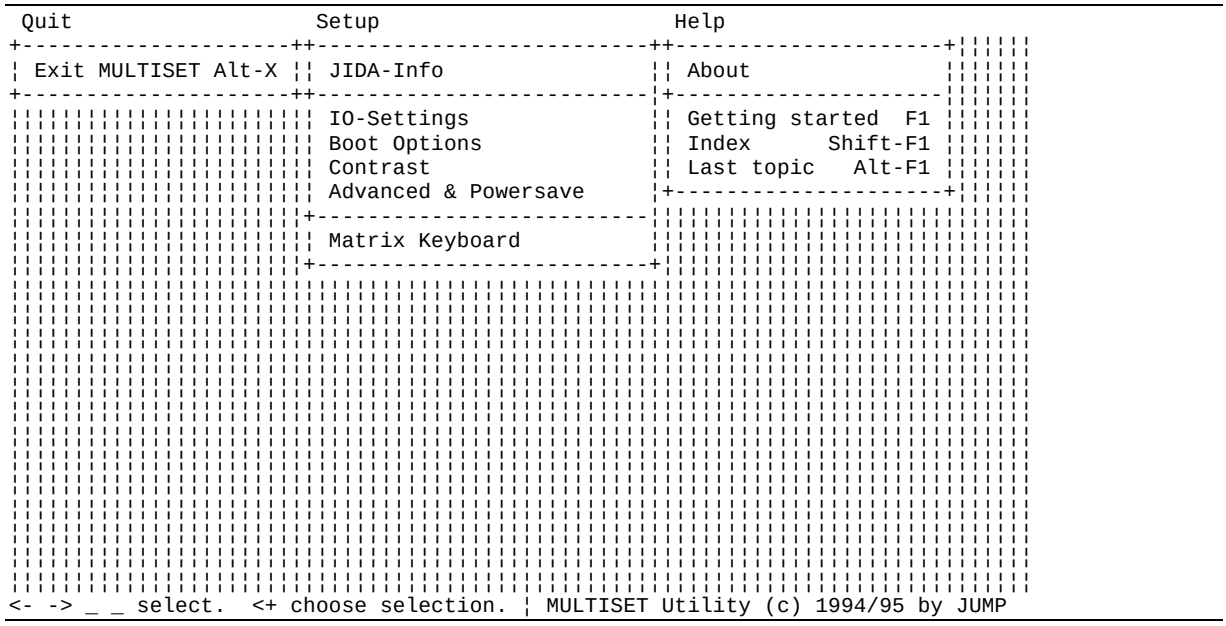
- Click on the entry with the mouse.
- Select the entry with the<Arrow up> <Arrow down> keys an hit <Return>
- Hit the highlighted character (without holding down <Alt>)

Working with Dialog Windows

- To activate any item in a dialog window, simply click to it with the mouse.
- If you do not have a mouse, select the dialog item by holding down the <Alt> key and hitting the key that is highlighted on or beneath the desired item. Use the <Space> key to switch items on or off. Use the <Arrow up> <Arrow down> keys to select one of different options. Hit <Tab> to jump to the next item or<Shift + Tab> to jump to the previous item.
- <Esc>may be used within any dialog to cancel the operation and return to the main menu.

The MULTISET Menus

The following graphic gives you an overview on all the menu entries of the MULTISET software:



Quit - Exit MULTISET

The function "Exit MULTISET" located in the "Quit" menu enables you to leave the MULTISET software. Click on "Quit", then click on "Exit..." to do this.

Alternatively, you may use <Alt + X> to leave the program.

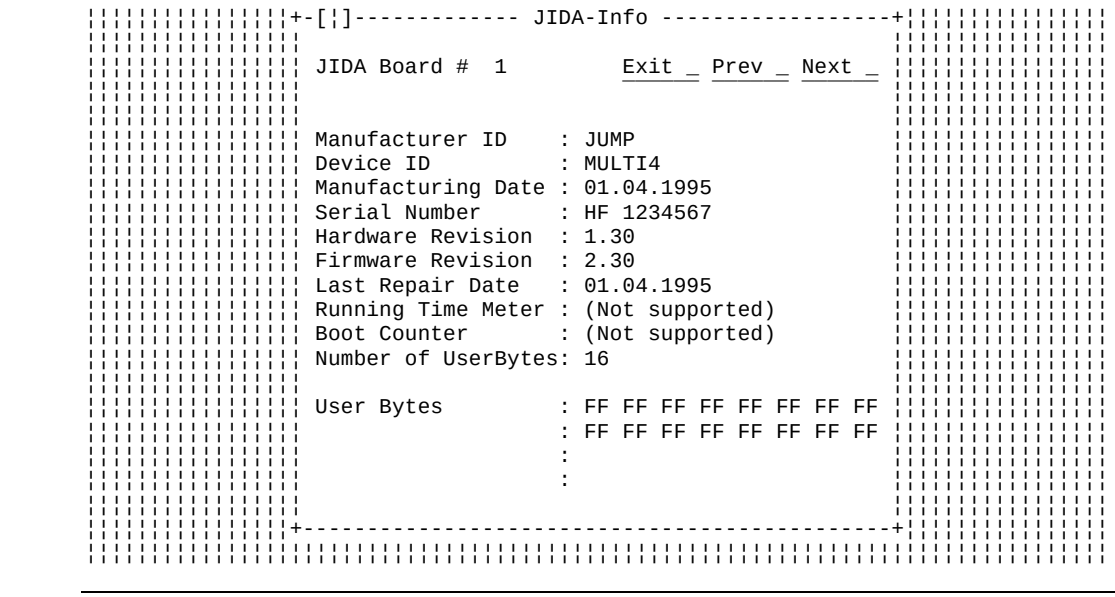
Close dialogs with <Esc> before trying to leave the program <Alt + X> or clicking on "Quit" will not work when in dialog boxes.

JIDA Info Dialog

JUMP introduced JIDA as a new standard for all boards with onboard BIOS. JIDA enables the user to access additional features of the board. The JIDA Info Dialog shows all available information on the first JIDA board in your system. (Board ID = 1).

You may use the "Prev" or "Next" button, to switch to a JIDA board with a higher or lower ID (JIDA IDs range from 1 to 127). If no previous or next board is available, the corresponding button will be invisible.

The "Ok" button may be used to leave the dialog, which looks like follows:



A description of each line is given in the following. For further details about accessing the JIDA functions refer to the section ”Programmer’s Guide” in this documentation.

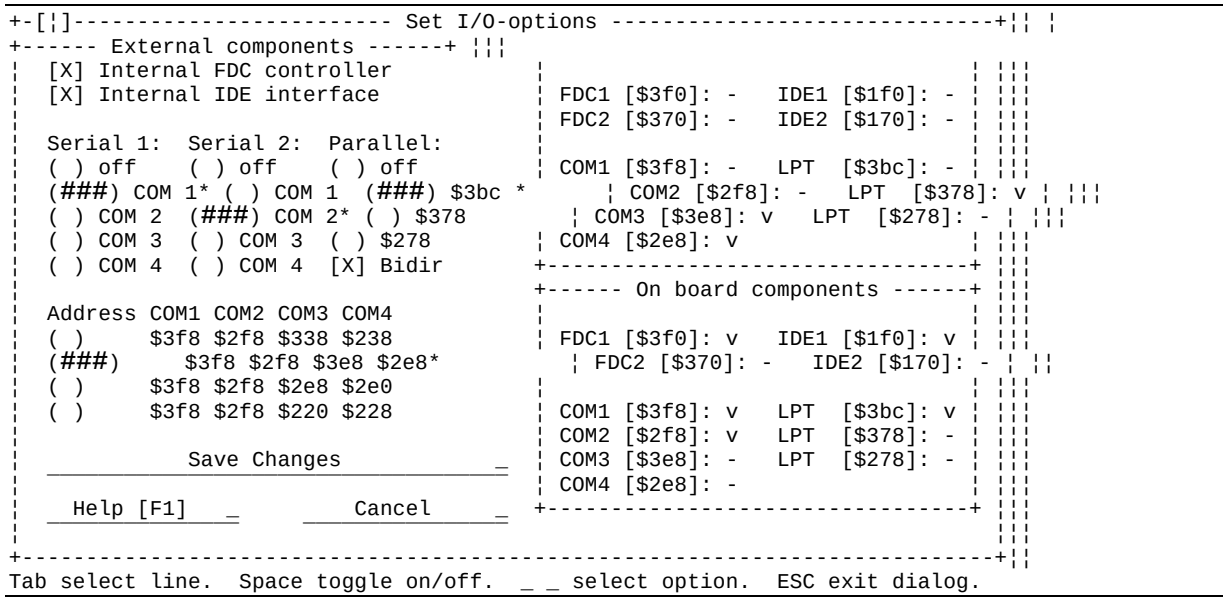
Manufacturer ID:	Identifies the manufacturer. Mostly "JUMP"
Device ID:	Identifies the device. "MULTI4 " with MULTI-4 boards.
Manufacturing Date:	Tells you, when your board was manufactured.
Serial Number:	A board individual serial number. This allows to identify each single board.
Hardware Revision:	The hardware revision of your board.
Firmware Revision:	The firmware (=BIOS) revision of your board.
Last Repair Date:	Shows, when the board was repaired last. If this date is smaller or equal to the manufacturing date, the board was never repaired.
Running Time Meter:	Total Running Time (in hours) of the board. Not supported by the MULTI-4.
Boot Counter:	Incremented with every boot of the system. Not supported by the MULTI-4.
Number of Userbytes:	Informs you, how many bytes of user accessible non volatile ram are available.
User Bytes:	Shows the first 32 bytes of non volatile user data (or all bytes if less than 32 bytes are available)

IO-Settings

This dialog enables you to configure the on-board IO-controller on the MULTI-4. If you select this option, MULTISET will try to determine, which IO-functions are present in the system. This test may last up to 30 seconds.

The MULTI-4 conducts the same automatic IO test during boot. If any of the IO-controllers functions collides with a component on another board, the on board component will be automatically turned off.

After the test is finished, this screen will be displayed:



The upper right half of the dialog gives an overview on all IO components that were found "externally" (this means: on another board).

The lower right half shows the actual configuration of the on board IO. This configuration may be changed using the left half of the screen.

If your system does not boot any more after you changed any option, you may press the [Ins] key during boot to use standard settings for this boot process. Return to MULTISET then and correct the setting.

Internal FDC: Enables or disables the on board floppy disk controller. May be disabled if no disk drive is connected to your system or if your disk drive is connected to a controller residing on an other card.

Internal IDE: Enables or disables the hard disk interface.

Serial 1: Allows you to configure the first of two serial ports. You can select to switch the port off or to use it as port COM1 - COM4. COM1 and COM2 will always use the standard IO-addresses 3f8h and 2f8h, while IO-addresses for COM3 or COM4 may be selected. Note that COM1 and COM3 will always release an IRQ4, COM2 and COM4 will release an IRQ3. Interrupts are hardwired within the controller and can not be changed.

Serial 2: Like "Serial 1". If you select the same COM for "Serial 1" and ,Serial 2", the second port will automatically be disabled.

Parallel: Use this field to select the address of the printer port or to turn off the printer port.

Bidir: If this option is enabled, the data lines of the parallel port may be used as inputs or as outputs. If disabled, the port may only be used as output.

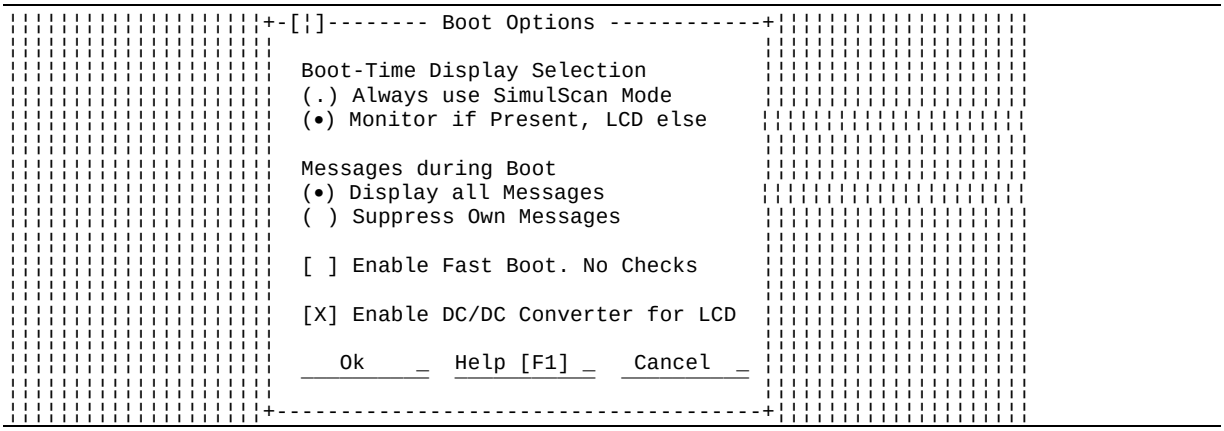
Address: Allows you to select port addresses for COM3 and COM4. Use the second setting (3e8h/2e8h) if you want the BIOS to recognize COM3 and COM4. Other addresses are not automatically handled by the BIOS. This setting has no meaning unless Serial1 or Serial2 is configured as COM3 or COM 4

"Save Changes" will write your changes into the configuration EEPROM. After a reboot of the system, the new settings will be active.

To provide a shorter way for field selection, every field contains one highlighted character (for example "B" in Bidir). Simply hold down ### and press the highlighted key to select the corresponding field.

Boot Options

The Boot Options influence the system's behavior during boot. The following dialog will be displayed:



Boot-Time Display Selection: Enables or disables SimulScan mode at boot time. If "Always use SimulScan Mode" is activated, the system will always start with both LCD and CRT enabled, no matter if a LC-Display or a monitor is connected to the MULTI-4. If "Monitor if present, LCD else" is selected, the system will start with only one display enabled. To decide, if to start in LCD mode or in CRT mode, the video controller uses an analog check on the CRT connector to determine, if a monitor is connected to the system. If a monitor is plugged in, the system will start in CRT-only mode, otherwise LCD-only mode will be used. This feature may also be used to start the system without any boot time messages being displayed on the panel.

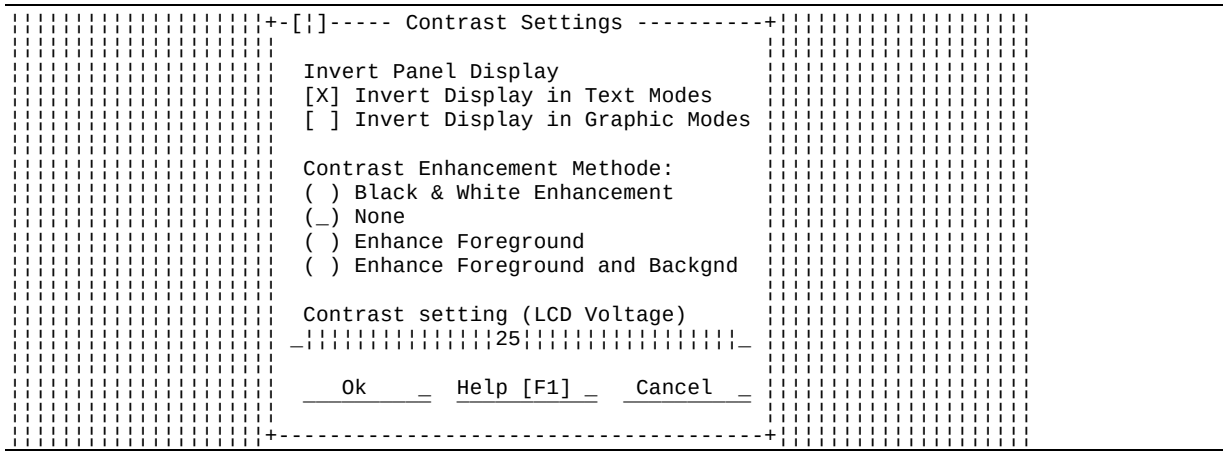
Messages during Boot: This option allows you to enable or disable system messages at boot time. If you select "Display all Messages", the MULTI-4 BIOS will display status and version information during boot. If you select "Suppress own Messages", the MULTI-4 BIOS will not generate any output on the screen. This option does not influence messages of any other board in the system (e.g. CPU-modules and their BIOSes)

Enable Fast Boot. No Checks: If you disable this option, the MULTI-4-BIOS will conduct a checksum test on the values in EEPROM during boot, before any of the settings is used. In addition, the BIOS will detect if any component of the on board IO-controller collides with an external component. In this case, the on board component will be turned off to avoid conflicts on the bus. If you enable the "Fast Boot" option, no checks are done during boot. This results in faster boot time at the price of reduced security.

Enable DC/DC Converter: Enables or disables the on board DC-DC converter. Note that the DC-DC converter will only work when the display mode is "panel only" or "SimulScan". In "CRT only" mode, the DC-DC converter will temporary be disabled. Use this setting carefully!!! If your display needs the generated voltage (as most monochrome and color STN panels do) and you disable the DC-DC converter, your panel will not display anything when booting next time. Disable the converter only if your panel does not need the voltage or if no panel is connected.

Contrast

This Dialog is used to configure display contrast, to invert the display and to select contrast enhancement.



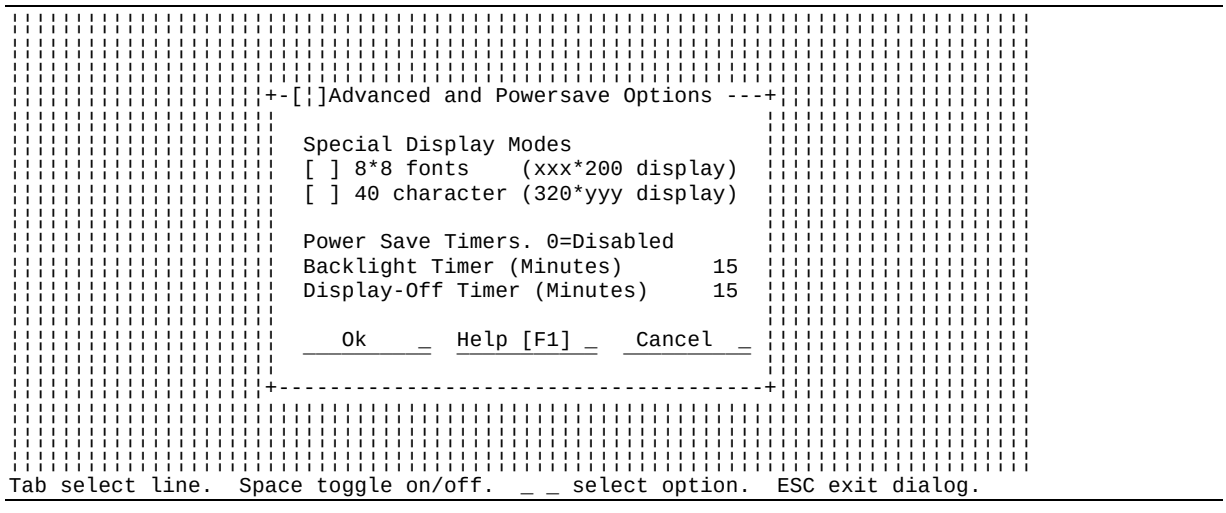
Invert Panel Display: This switches control the "Invert" feature of the on board graphic chip. Text- and graphic modes may be inverted separately. By this way you can get black characters on white background or vice versa when you work with a monochrome panel. The inversion only influences the LCD, never the monitor. As "positive" or "negative" display is dependent of the display technique of your panel, it may be necessary to experiment with these settings. If the colors on your color STN display are different from the colors you expected, it may be necessary to change this settings.

Contrast Enhancement Method: Controls the "Contrast Enhancement" feature of the graphic controller. Use this setting only with monochrome displays to get a better contrast in specific display situations. This setting only affects displays used in ”panel only” mode. The CRT or panels in SimulScan mode will not be affected. Never use this setting with passive color displays, as this may change your display colors. Select ”none” with color panels.

Contrast Setting: This scrollbar controls the onboard voltage generator. It is used to adjust the contrast of those LC-Displays that need an additional driving voltage as most monochrome and color STN panels do. Press ###### to select the scrollbar, then use the ###### keys to adjust. The contrast may be adjusted in 64 steps ranging form 0 (minimum voltage) to 63 (maximum voltage). A resistor and a zener diode in Jump’s display cables prevent the display from too high voltages, so you can always use the full range (0..63) for contrast setting.

Advanced and Powersave

This dialog contains options for advanced display modes which can be used with panels that have a resolution of less than 640*480. In addition, this dialog contains the settings for the display power save modes:



Special Display Modes: These two switches are useful for displays with a resolution of less than 640*480 pixels. The "8*8 fonts" switch is often used with displays that offer a vertical resolution of 200 or 240 pixels. If you enable this switch, all textmodes will use a smaller font where each character has only 8*8 instead of 8*16 pixels. The standard 25 lines on your screen will then need only 25*8=200

pixels and can be displayed completely on your LCD. The "40 character" should be used with displays that have a horizontal resolution of 320 instead of 640 pixels. In this case, there is no possibility to use a smaller font as this would leave only 320/80=4 pixels per character. Such a small character would be unreadable. If you enable this switch, the 40 column modes will be used instead of the 80 column modes. You must reboot to see the effect of these settings

Power Save Timers: These entries are used to specify the time-outs for the panel power save options. The ”Backlight Timer” value will switch off the panel’s backlight if no keystroke and no access to the video memory occurs within the specified time. The ”Display-Off Timer” will switch off the display completely after the specified time

Matrix Keyboard setup

Overview

A matrix keyboard may be connected to the MULTI-4 via the matrix keyboard connector. Pins Y0-Y7 are outputs that should be wired with matrix lines, the inputs RA0-RA4/RB0-RB4 should be connected with matrix columns. Each connection between a line and a column then generates an internal scan code which is transferred to the system. The BIOS converts this internal scan code to a keyboard scan code using the matrix decoder table. This matrix decoder table may completely be configured by using the MULTISET utility. The matrix definition dialog looks like this:

```
+-[¦]------ Set scan codes for matrix keyboard ----------+
|  RA0 RA1 RA2 RA3 RA4 RB0 RB1 RB2 RB3 RB4   Name        |
|    3   5   7   9  11  13  15  17  19  21 Pin          |
|   FF  FF  FF  FF  FF  FF  FF  FF  FF  FF   2 Y0       |
|   FF  FF  FF  FF  FF  FF  FF  FF  FF  FF   4 Y1       |
|   FF  FF  FF  FF  FF  FF  FF  FF  FF  FF   6 Y2       |
|   FF  FF  FF  FF  FF  FF  FF  FF  FF  FF   8 Y3       |
|   FF  FF  FF  FF  FF  FF  FF  FF  FF  FF  10 Y4       |
|   FF  FF  FF  FF  FF  FF  FF  FF  FF  FF  12 Y5       |
|   FF  FF  FF  FF  FF  FF  FF  FF  FF  FF  14 Y6       |
|   FF  FF  FF  FF  FF  FF  FF  FF  FF  FF  16 Y7       |
|   FF  FF  FF  FF  FF  FF  FF  FF  FF  FF  18 Y8       |
|   FF  FF  FF  FF  FF  FF  FF  FF  FF  FF  20 Y9       |
|   FF  FF  FF  FF  FF  FF  FF  FF  FF  FF  22 YA       |
|   FF  FF  FF  FF  FF  FF  FF  FF  FF  FF  24 YB       |
|   FF  FF  FF  FF  FF  FF  FF  FF  FF  FF  26 YC       |
|   FF  FF  FF  FF  FF  FF  FF  FF  FF  FF  28 YD       |
|   FF  FF  FF  FF  FF  FF  FF  FF  FF  FF  30 YE       |
|   FF  FF  FF  FF  FF  FF  FF  FF  FF  FF  32 YF       |
|                                                        |
|  Ok _  Help _  Cancel _   Teach In  _   Save _  Load _ |
|                                                        |
+--------------------------------------------------------+
Tab select line.  Space toggle on/off.  _ _ select option.  ESC exit dialog.
```

Defining Scan Codes Manually

The dialog box contains one entry for each crosspoint on the matrix keyboard. To define the scan code that should be generated when RA3 is connected with Y1, you must fill in the fourth column in line two.

The numbers, which are entered in hexadecimal form, are the scan codes, not the ASCII codes of the corresponding key. Scan codes are individual numbers for every key on your keyboard. For example, the scan code of the [Esc] key is 01, the [A] key has scan code 1Eh, while the left [Shift] key is associated with scan code 2Ah. A detailed listing of scan codes can be found in the Section ”Keyboard Scan Codes” in this Manual.

Some keys on MF-II keyboards generate so called "extended scan codes". This means, when the key is pressed, not only the scan code is transmitted to the keyboard, but the scan code is preceded by a "pre-code" with value 0Eh. If you want the matrix keyboard to generate one of those "extended scan codes", add 80h to the scan code. Example: Entry 1Ch will result in scan code 1Ch (associated to the [Return] key). Entry 9Ch (added 80h) will generate the scan codes 0Eh 1Ch, designating the [Enter] key on the numeric keypad.

Using the Teach-In-Mode

As it may sometimes be difficult to determine the desired scan code or the row and column of a key on your matrix keyboard, MULTISET offers an easy to use "Teach-In-Mode":

To determine the row and column associated with a key on the matrix keyboard, simply connect the matrix to the MULTI-4 and press the desired key. The cursor will automatically jump to the correct input field.

In addition, there is a function to determine the scan code of any key on your standard AT keyboard. To do this, place the cursor in the input field that is to be filled (by using ###, ###+###, ###### keys or by using the "Auto-Jump" method described above). Then click on the ”Teach In” Button or press ###### Release all keys then. MULTISET will now display a message, asking you to press the desired key. Press any key on your keyboard now. MULTISET automatically determines the scan code and enters the value to the selected input field.

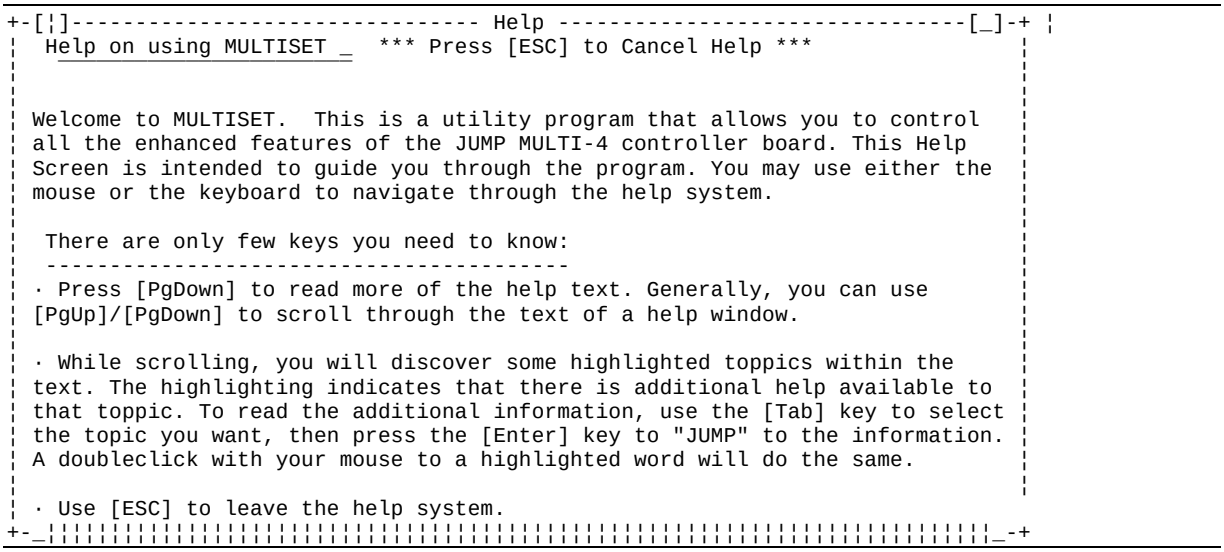
The scan codes may also be defined by your own software using the JIDA extension interrupt. See section ”Programmers Reference” for details.

For further details on how to connect a 8*10 or a 16*10 matrix to the MULTI-4, refer to section ”Matrix Keyboard Connector”

The MULTI-4 On-line Help System

Help at your fingertips

When working with MULTISET, you may press ### at any time, to invoke the help system. The help window will automatically display information on that part of program you are currently using. Some dialogs display an additional ”Help” button. You may also press this button to get context sensitive help. The help screen appearing might look like this:



Maneuvering through the help screens

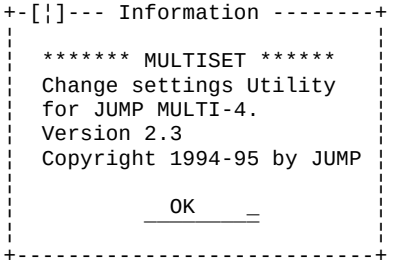
You may use either the mouse or the keyboard to navigate through the help system. There are only few keys you need to know:

- Press <Page up>/ <Page down> to scroll through the text of a help window.
- While scrolling, you will discover some highlighted topics within the text. The highlighting indicates that there is additional help available to that topic. To read the additional information, use the <Tab> ### key to select the topic you want, then press the <Return> key to jump to the information. A double-click with your mouse to a highlighted word will do the same.
- Use <Esc> to leave the help system.

Other helpful Options

When no dialog is visible on the screen, you can access additional help options through the help menu:

- **About:** This entry will display an info box which can be used to determine the revision level of the MULTISET-Utility:



- **Getting started:** Displays a help screen that explains how to use the help system and how to start working with MULTISET.
- **Index:** Will give you an overview on all help topics that are currently available.
- **Last topic:** This menu entry (or the keys <Alt + F1>) can be used to return to the last help screen, that was displayed before leaving the help system

The Quiet Boot Feature

For some applications it may be advantageous to boot with a complete dark display and to enable the display after the start of the application. This is possible, if you use a LCD panel for display and no monitor is connected to the MULTI-4.

To achieve this, use the MULTISET Utility, Menu "Setup", Entry "IO-Settings" to disable the "Simul" option. Select "Write to EEPROM and reboot". Your MULTI-4 is now configured to boot with only one display (LCD or CRT) enabled.

The next step is to make the board boot in CRT mode with the LCD disabled. For this you need a 15 pole male high density SUB-D connector (as used with VGA monitors). Connect pin 2 and pin 7 of the connector via a 75### resistor and plug it to the CRT connector of your MULTI-4.

When booting the next time, your MULTI-4 will detect a "monitor" connected to your system and start in CRT-only mode. Your LC Display will remain dark, but may manually be enabled by pressing <Ctrl + Att + 6>.

To automatically enable the panel after booting, place the following lines in your AUTOEXEC.BAT just before the command that starts your application:

```
@ECHO OFF
CLS
ENDDARK
(start your application)
```

(ENDDARK.EXE is included on the disk delivered with this manual and must be copied to your boot drive.)

Controlling the MULTI-4 by Keyboard

Some important and often used functions have been assigned to special key-combinations. To use one of these combinations, you have to hold down the ### and the ### key and then strike one of the numbers ###-###. Please note that you have to use the numbers on the alpha keyboard (located above the ############################### keys). The keys on the numeric pad have no effect.

□do this, internally the programmed DC-DC value is decreased by one. If the DC-DC value is already 0, no action is performed.

<Ctrl + Alt + 1> Increase output voltage. To do this, internally the programmed DC-DC value is increased by one. If the DC-DC value is already 63, no action is performed.

###<Ctrl + Alt + 2> Set output voltage to mid range. To do this, internally the programmed DC-DC value is set to 32.

###<Ctrl + Alt + 3> Toggle DC-DC converter on or off.

###<Ctrl + Alt + 4> Display in CRT-only mode. Flat panel is switched off. If no monitor is connected to the MULTI-4, the display will be switched to Panel-only mode.

###<Ctrl + Alt + 5> Display in Panel-only mode. CRT output is disabled.

###<Ctrl + Alt + 6> Display in SimulScan-mode. Picture will be displayed on flat panel and monitor simultaneously. If no monitor is connected to the MULTI-4, the display will be switched to Panel-only mode.

###<Ctrl + Alt + 7> Toggle reversed mode. Enables or disables reversed display on flat panel. Reversed mode can be switched separately for text and graphics modes.

<Ctrl + Alt + 8> Toggle contrast enhancement. Toggles between the four available contrast enhancement modes. Refer to the corresponding VGA-BIOS function call in the "Programmer Guide" section for details

<Ctrl + Alt + 9> Toggle backlight on and off.

Information Displayed during Boot

On boot time, the MULTI-4 extension BIOS will check for external IO-components, read your configuration data out of e²prom and program the on-board IO-controller corresponding to your selections. If any external component is found with the corresponding on-board component configured as "enabled", the on-board component will be disabled to avoid address conflicts.

Please note that none of the following messages will be displayed, if "Suppress own Messages" is selected in MULTISET's "Boot Options" dialog.

During boot, the following information might be displayed:

```
LCD Extension BIOS V 2.3 <MLCDR123.ROM>    <C190-1-20-Ex-In-Em-IC-CC-Tb>
Emergency Boot Request Detected.        Starting with Default Settings.
Configuration Checksum Invalid. Default Parameters will be used.
   Hardd. Ctrl: onboard                    Floppy Ctrl: onboard
   Serial 1   : onboard  at 3F8h [COM1]    Serial 2   : onboard  at 3F8h [COM2]
   Printerport: onboard  at 3BCh unidir
```

The Title Section

```
LCD Extension BIOS V 2.3 <MLCDR123.ROM>
```

The first part of line one describes the BIOS and tells you about the revision. The name enclosed in <> is the internal file reference used by JUMP. You should find a label with the same information on your BIOS-Prom.

The System Information

```
                                        <C190-1-20-Ex-In-Em-IC-CC-Tb>
                                          (1)(2)(3)(4)(5)(6)(7)(8)(9)
```

The right half of line one contains information about the configuration of you system:

(1) The MULTI-4 uses eight IO-addresses to access all the additional features of the MULTI-4. Those extension registers are used to access the E²PROM, to program the IO-Controller, to communicate with the matrix decoder, to set the display contrast and to control the open collector outputs. By default, the extension registers start at IO-address C190h. If this causes an address conflict, the address may be changed to C390h by setting a jumper. During boot, the BIOS auto-detects the correct address of the extension registers and displays the detected address at position (1). If none is displayed at position (1), the extension registers or the E²PROM are defective. Pleas contact JUMP in this case for repair.

(2) The DC/DC converter on the MULTI-4, which is used to supply contrast voltage to passive displays may be enabled or disabled with the MULTISET program. The information at position (2) indicates, that the converter is disabled by -0- or that it is enabled by displaying -1-.

(3) The DC/DC converter mentioned above can be adjusted by software. The output voltage may be changed by the MULTISET utility or by special key-combinations to change the displays contrast. The actual contrast setting is stored in E²PROM and displayed in position (3) during boot. As the display is hexadecimal, possible values range from 00 to 3F.

(4)

(5) The MULTI-4 BIOS automatically checks, if the on board SMC IO-controller is present. As it is possible that a second SMC IO-controller exists on an other board, which uses the same IO-ports for programming, the BIOS also has to check for the presence of an external controller. If an external controller is detected together with an internal controller, the BIOS will take care that only the on-board controller is reprogrammed without changing the settings of the external controller. The presence of an external controller is indicated by Ex- at position (4). The presence of the on-board controller is indicated by In- at position (5). If a controller is not detected, the associated position will be blank.

(6) The BIOS displays Em- at position (6) if the current boot process is an emergency boot. For details about the emergency boot, please refer to the "Emergency Boot" section.

(7) The MULTI-4 contains a lot of identification information, that can be obtained by using the adequate JIDA functions. The section "Programmer's Guide" in this documentation describes

how to access this information from your programs. The MULTISET utility may be used to display the information. As part of this JIDA information is stored in e²prom, and this information may be very important for some applications, there is a checksum protection it. The information is checked during boot. If everything is ok, `IC-` will be displayed at position `(7)`, otherwise, this position will remain blank.

`(8)` The configuration for IO- and display are stored in e²prom, too. During boot, the BIOS checks, if the configuration is valid and displays `CC-` at position `(8)`, if everything is ok. Otherwise, this position remains blank and the BIOS uses default-values to program the IO-controller and the display.

`(9)` One of the options in MULTISET allow to select a "turbo-boot-mode", which does no checksum and collision tests during boot and speeds up the boot process. If this mode has been enabled in MULTISET, `Tb-` will be displayed at position `(9)`.

Emergency Boot

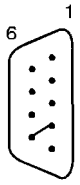
```
Emergency Boot Request Detected.          Starting with Default Settings.
```

If an emergency connector is plugged in or the user pressed the ### key during boot, an emergency boot will be made: All information stored in the e²prom will be ignored an the system will boot with default values.

This is useful, for example, if you your system does not boot anymore, because you accidentally disabled the IDE- and floppy-interface.

An emergency boot will not change any information in the e²prom, but ignore the information for one boot process. The above message will be displayed on the screen in line two, if the current boot is an emergency boot.

An emergency connector can be produced in a very easy way: Take a 9-pole male SUB-D connector and connect pin 4 to pin 9. Plug this connector into one of the two serial ports (COMA or COMB) on the MULTI-4 before turning power on.



Emergency Connector

Configuration Checksum Error

```
Configuration Checksum Invalid. Default Parameters will be used.
```

The configuration information for IO- and display, which are made by using the MULTISET utility are stored in e²prom. During boot, the BIOS checks, if the information is valid. If a checksum error is detected, the BIOS will use default-values to program the IO-controller and the display. If you get this message during boot, you should use the MULTISET utility to check your configuration. If you save the configuration within any dialog of the MULTISET utility, a new checksum will be calculated.

IO-Settings

```
Hardd. Ctrl: onboard                    Floppy Ctrl: onboard
Serial 1   : onboard  at 3F8h [COM1]    Serial 2   : onboard  at 3F8h [COM2]
Printerport: onboard  at 3BCh unidir
```

This part of the screen will only be displayed, if an onboard IO-controller is present. The actual status for every of the five on board components is shown in these lines. Each component name is followed by one of the words `onboard, disabled,` or `external.`

`onboard` informs you that the corresponding component of the on-board controller has been enabled.

`external` is used to tell you that an external component has been found at the corresponding address an the on-board component has been disabled to avoid address conflicts. Please note that the check for possible conflicts is only made if you did not enable the ”Fast Boot” option with MULTISET.

`disabled` means that the on-board component has been disabled and no external component has been found at the corresponding address.

If a serial port is enabled, you will get additional information about the IO-address used by this COM-Port. If one of the standard addresses was used (3F8h, 2F8h, 3E8h, 2E8h), the port name (”COM1”) will be displayed in square brackets, otherwise, only the address will be displayed.

With the printerport, you will get information about the address that is being used for the port. In addition, the BIOS displays `unidir` or `bidir` to indicate if the port is programmed as standard printer port (write only) or as extended printer port (with read and write access).

The LC-Display Interface

Connecting a Display by a Standard Cable

Many different display cables for a wide spread variety of displays are available through JUMP. If you use one of those standard cables supplied by JUMP, configuration is very easy:

1. Check, whether you have the correct cable for the panel you plan to use. Inspect the cable for damages.
2. Disconnect power from your System.
3. Connect the 50-pole connector of your display cable to the display connector (J16) on the MULTI-4.
4. Connect the other end of the cable to your display.
5. Supply power to your system
6. If no picture appears on your display, press<Ctrl+Alt+7>, to select the SimulScan mode.
7. If your panel is too dark or too light, try pressing ###<Ctrl+Alt+1> or ###<Ctrl+Alt+2> repeatedly to adjust contrast.
8. If you still get no picture, start MULTISET and select the "Boot Options" dialog. Check "Enable DC/DC Converter for LCD". Make sure you have this setting enabled ([X]). As a second step, check the "Boot time Display Selection" in the same dialog. It must be set to "Always use SimulScan". Click the "Save" button.

   Now reboot your system. Start MULTISET again and select the "Contrast Setting" dialog. Try to change the Contrast setting with the slide bar while watching your panel.

   If you still don't see any improvement, you may consider to contact JUMP for technical support.

Configuring your own Cables

Actual Information from the JUMP-Mailbox

First of all, you should check the JUMP-Mailbox for the actual panel list. We regularly update the list of panels that were tested with the MULTI-4. Look for `PANEL.ZIP`, which might already contain a connection table for the display you need.

To use the mailbox, set your terminal software to these parameters: 8 data bits, no parity, 1 stop bit. You may use any transfer speed from 1200 up to 14400 bits per second.

Call the mailbox: +49-(0)9482-9405-27

If you are asked for your first and last name, simply enter "JUMP".

Now you are connected to the mailbox. Feel free to download any information you need.

DC/DC-converter

Attention!!! Please note that you **must** adjust the voltage range of the MULTI-4 to fit your panel by using a zener-diode and a resistor. This two elements determine the voltage range for the flat panel. If you leave away any of the components or if you use incorrect values, the flat panel will probably be destroyed, because 50V are generated onboard in the non regulated mode.

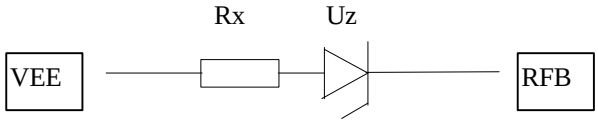
The MULTI-4 has an onboard DC/DC converter for positive and negative LCD contrast voltages. All passive LCD's need an extra LCD (backplane or substrate) voltage either positive (for passive color) or negative (for passive monochrome displays). The onboard converter generates up to +50V and -45V from a single 5 V supply. This means that no 12V supply is required for LCD voltage generation. The range for the panel voltage is set via a resistor and a zener diode mounted on the LCD cable. Therefore a customer does not have to adjust a voltage before connecting a different display to the MULTI-4.

The LCD contrast can be controlled via special keystrokes, which also allow for contrast enhancement features and panel inversion. This features do not need any software drivers to be installed, because they are provided by the onboard BIOS.

This is a definitive advantage over other LCD controller boards, as most products on the market need an external LCD Voltage converter, which adds additional cabling, converter mounting and cost to the system.

The on board DC-DC converter may be used for all panels which require a second supply voltage. The converter is able to supply a negative voltage (available on pin VEE) and a positive voltage (available on pin VBB). The output voltage may be changed by the user via keyboard.

The output range has to be adjusted before any panel is connected to VEE or VBB pins. Two external components are needed for adjusting: The VEE output is connected to the RFB input via a zener diode and a resistor as shown in the following diagram:



The values for Rx and Uz may be calculated as follows:

$$U_z = \frac{R_{int} * (I_{max} * VEE_{mid} - I_{mid} * VEE_{max})}{VEE_{mid} - VEE_{max} + R_{int} * I_{min}}$$

$$R_x = \frac{R_{int} * (VEE_{mid} - U_z)}{R_{int} * I_{mid} - VEE_{mid}}$$

where: R_{int}=3,3M###
I_{min}=6,66µA I_{mid}=13,33µA I_{max}=20µA
VEE_{max}= maximum output voltage required
VEE_{mid} = mid range output voltage required

The real output voltage may then be adjusted in 64 steps, where a value of 0 gives the minimum voltage (which has not been calculated here), 32 gives the mid range voltage VEE_{mid} and 63 gives the maximum voltage VEE_{max}.

Information on how to change the output voltage via keyboard can be found in the section ”Controlling the MULTI-4 by Keyboard”.

Note that any change of the DC-DC value will be saved in E²PROM. If your restart your system in Panel- or SimulScan mode, the last value will automatically be restored.

Important note: BIOS revisions 2.0 and 2.2 (labeled MLCDR120 and MLCDR122) had an error which prevented the saving of the contrast setting. With this BIOS revision, permanent changes of the contrast voltage can only be made by using the MULTISET utility. Changes via the keystrokes mentioned above are not stored permanently with those BIOS revisions.

Backlight converter

Most LCD displays need an extra AC or DC Voltage for their Backlight, which is generated by an external converter. This converter usually generates between 60 and 1000 Volts AC and has to be as close as possible to the Backlight to avoid capacitive losses on long cables. The MULTI-4 has a MOS switch to switch Backlight on or off via software control and can be configured to switch 5V or externally supplied 12 V to the Backlight converter.

The supply voltage for the backlight converter should be connected to pin 42 (BackSrc) of the MULTI-4 display connector. The switched backlight supply voltage is then available at pin 44 (SwBack).

LCD Connector Pinout (J16)

Various types of flat display panels may be connected to the 50pin plug J16. The pinout is shown in this table:

Name	Pin	Pin	Name
LFS	1	2	LLCLK
VDCLK	3	4	FPVEE
SW_VDD	5	6	GND
VEE	7	8	UD0 SLD4 G1
UD1 SLD5 G2	9	10	UD2 SLD6 R0
UD3 SLD7 R1	11	12	LD0 SLD0 B0
LD1 SLD1 B1	13	14	LD2 SLD2 B2
LD3 SLD3 G0	15	16	SUD7 R3
SUD6 R2	17	18	SUD5 G5
SUD4 G4	19	20	SUD3 G3
SUD2 B5	21	22	SUD1 B4
SUD0 B3	23	24	R5
R4	25	26	GND
DE	27	28	FPVDD
MOD	29	30	FPBACK
+3,3V	31	32	VEE
GND	33	34	RFB
GND	35	36	GPI
VDD_SRC	37	38	VBB
+12V	39	40	HSYNC
+5V (VCC)	41	42	BACK_SRC
VSYNC	43	44	SW_BACK
EXT_CTRL	45	46	EXT_ADJ
GND	47	48	PS3
PS2	49	50	PS1

LCD Signal Explanation

Some pins generate up to three different signals, depending on the type of display connected.

B/W-LC Displays use the signals UDx and LDx (LDx for dual scan displays only)

STN Color Displays use SUDx and SLDx

TFT Displays use Rx, Gx and Bx

Pin Name	Pin Description
LFS	LCD Frame Start: This output provides a pulse to start a new frame on flat panels.
LLCLK	LCD Line Clock: This output is used to drive the LCD-panel line clock. This signal is also designated as LP or CP1 by some panel manufacturers.
VDCLK	Flat Panel Video Clock: This signal is used to drive the flat panel shift clock which is designated as CP2 by some panel manufacturers.
FPVEE	Flat Panel VEE enable: This output is part of the panel power sequencing. Normally, this signal is only used internally. Use pin 7 (VEE) instead.
SW_VDD	Switched VDD: Part of the panel power sequencing. Connect to VCC or VDD on most panels. This pin is a switch output. Desired output voltage must be supplied at pin 37 (VDD_SRC).
GND	Ground.
VEE	VEE: Variable, negative output voltage generated by the DC-DC converter. Needed as second supply voltage on many panels. Note: Output voltage has to be adjusted before use! See section ”DC/DC-Converter” for details.

UD0-UD3	Upper Data: The upper data bits are typically used with monochrome dual-scan flat panels to provide 4-bit parallel data for the upper portion of the panel.
LD0-LD3	Lower Data: The upper data bits are typically used with monochrome dual-scan flat panels to provide 4-bit parallel data for the lower portion of the panel.
SLD0-SLD7	STN Lower Data: The lower data bits are for use with color STN LCD panels.
SUD0-SUD7	STN Upper Data: The upper data bits are for use with color STN LCD panels.
R0-R5	Red Bits: These bits contain red color data for TFT color flat panels.
G0-G5	Green Bits: These bits contain green color data for TFT color flat panels
B0-B5	Blue Bits: These bits contain blue color data for TFT color flat panels
DE	Display Enable: For those flat panels that require an external display enable, this pin is used to provide a data enable. For STN single-scan dual-clock panels, it is used as second Shift Clock output.
FPVDD	Flat Panel VDD enable: This output is part of the panel power sequencing. Normally, this signal is only used internally. Use pin 5 (SW_VDD) instead.
MOD	Modulation: This output provides AC inversion. It should be connected to the MOD, FR or DF inputs of the panel, as is appropriate. Some panel manufacturers provide this function in the panel circuitry.
FPBACK	Flat Panel Backlight enable: This output is part of the panel power sequencing. Connect to the panels backlight enable or use pin 44 (SW_BACK) to supply the backlight converter. Do not use this signal as supply voltage for the backlight converter!
RFB	Feed Back input: Used to adjust the voltage generated by the on board DC-DC converter. See section "DC/DC-Converter" for details.
GPI	General Purpose Input: TTL level input for general use. To read this input pin, follow the instructions in section "Programmer's Guide".
VDD_SRC	VDD Source: Connect this pin to the source of VDD. For most panels, this will be +5V. This input leads to a switch which is controlled by FPVDD. The switched voltage is output at pin 5 (SW_VDD).
VBB	VBB: Variable, positive output voltage generated by the DC-DC converter. Needed as second supply voltage on some panels. Note: Output voltage has to be adjusted before use! See section "DC/DC-Converter" for details.
+12V	+12V output. May be used as input for pin 42 (BACK_SRC) if your backlight converter needs +12V power supply and is connected to pin 44 (SW_BACK). This voltage is not generated onboard but is directly connected to the +12V power supply of the MULTI-4. If you do not supply +12V to your MULTI-4, you will not be able to use this pin.
HSYNC	Horizontal Sync: This output supplies the horizontal synchronization pulse to the monitor. It is normally not needed for flat panels.
+5V (VCC)	+5V output. May be used as input for pin 42 (BACK_SRC) if your backlight converter needs +5V power supply and is connected to pin 44 (SW_BACK). May also be used as input for pin 37 (VDD_SRC) if your display needs 5V supply voltage.
BACK_SRC	Backlight Source: Connect this pin to the supply voltage for your backlight, if necessary. This input leads to a switch which is controlled by FPBACK. The switched voltage is output at pin 44 (SW_BACK).
VSYNC	Vertical Sync: This output supplies the vertical synchronization pulse to the monitor. It is normally not needed for flat panels.
SW_BACK	Switched Backlight Supply: May be used as supply for backlight converters. Outputs the voltage supplied at pin 42 (BACK_SRC).
EXT_CTRL	External Control: Used to change the output voltage of the DC-DC converter. See Maxim MAX749 data sheet for details.
EXT_ADJ	External Adjust: Used to change the output voltage of the DC-DC converter. See Maxim MAX749 data sheet for details.
PS1-PS3	Panel Sense: Inputs for panel type selection:

Panel type	**PS1**	**PS2**	**PS3**
16-Bit STN display	connect GND	connect GND	connect GND
4/6.3 MHz Monochrome Displ.	connect GND	connect GND	open
Dual Scan Color Display	connect GND	open	connect GND
TFT Color Active Matrix Display	connect GND	open	open

All other combinations are invalid.

Matrix Touch Controller

The AT96-MULTI-4 is able to control a 8*10 (80 keys) Matrix Keyboard or Matrix Touch Screen in addition and fully parallel to the standard keyboard. The customer needs no additional driver software to support this feature, since the matrix support is fully integrated in the onboard BIOS. The user can configure the matrix interface with the MULTISET utility program. This program allows to assign a scan code or extended scan codes to any crosspoint on the matrix. Refer to section ”Matrix keyboard setup” for details.

Any matrix keyboard may be connected to 26-pin connector J14.

Matrix Connector Pinout

Name	Pin	Pin	Name
GND	1	2	Y0
RA0	3	4	Y1
RA1	5	6	Y2
RA2	7	8	Y3
RA3	9	10	Y4
RA4	11	12	Y5
RB0	13	14	Y6
RB1	15	16	Y7
RB2	17	18	/MCLR
RB3	19	20	OC0
RB4	21	22	OC1
RB5	23	24	RB7
RB6	25	26	VCC

Matrix Signal Explanation

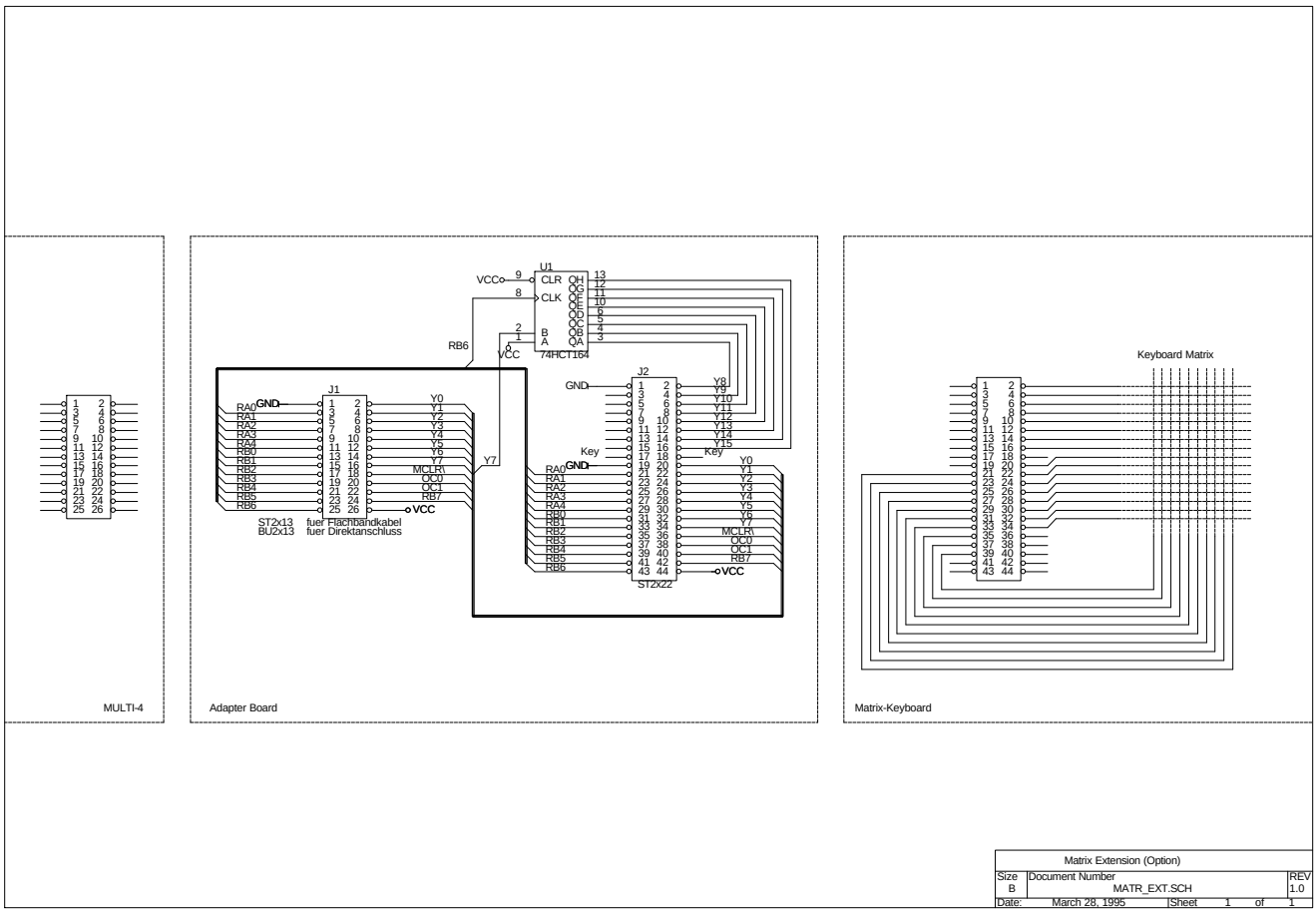
Pin Name	Pin Description
GND	Ground
RA0-RA4	Scan inputs
/MCLR	Reserved for factory testing. Don’t connect anything.
Y0-Y7	Scan outputs
RB0-RB4	Scan inputs
RB5-RB7	Reserved. Don’t connect anything.
OC0, OC1	General purpose open collector outputs.
VCC	+5V

To connect a matrix keyboard, wire outputs Y0-Y7 with matrix lines and inputs RA0-RA4, RB0-RB4 with matrix columns. Note that only one of the Y0-Y7 outputs is pulled low at a time when scanning. The other seven outputs remain high. Each connection between a line and a column generates an internal scan code which is transferred to the system. The BIOS converts this internal scan code to a keyboard scan code using the matrix decoder table. If you want to learn more about user defined scan codes, refer to section ”Matrix keyboard setup”

Standard and Enhanced Matrix Keyboards

Any Matrix with formats up to 8*10 crossings may be connected to a standard MULTI-4. In this case, the Pins Y0 to Y7 are used as outputs (active low), RA0 to RA4 and RB0 to RB4 are used as inputs.

If you need a larger matrix, there is a possibility to enhance the matrix to 16*10 crossings. To achieve this, you will need an update for the matrix controller on your MULTI-4 which is labeled MLCDO132 or above and some external circuitry, which is shown in the following diagram:



Multi Purpose Open Collector Outputs

As described above, there are two general purpose open collector outputs available on the matrix keyboard connector. The outputs are driven by an ULN2003D. This offers you a maximum collector-emitter voltage of 50V and the capability of driving currents of 100mA (continuous, only one output low), 50mA (continuous, both outputs low) or 500mA (peak, one output). After reset, both outputs are in high state. There is no on board pull-up resistor, so an external resistor has to be used, if required.

Your software may access the two outputs by using a JIDA function call. Refer to the ”Programmer’s Guide” for details.

Programmer's Guide

The JIDA Interface

JIDA Revision History

Rev.	Date	Subject	Changed by
0.1	31.05.94	Initial proposal for standardization.	H.Fink
0.2	01.06.94	Described procedure for information inquiry. Added return value for "Not repaired" to "Get Repair Date" Changed "Read Running Time Meter" return values.	H.Fink
1.0	23.08.94	Changed whole spec to achieve compatibility with Windows NT	H. Fink
1.1	21.09.94	Added function 42h (Get Number of User Bytes available)	H. Fink
2.0	18.04.95	Added functions 09h,0Ah,0Bh	H. Fink

JIDA (Jump Intelligent Device Architecture) Overview

Every board with on board BIOS extension shall support the following function calls, which supply information about the board. JIDA functions are called via Interrupt 15h with AH=EAh, AL=function number, DX=4648h (security word), CL=board number (starting with 1).

The interrupt will return with CL###0, if a board with the number specified in CL does not exist. CL will be equal to 0 if the board number exists. In this case, the content of DX is used to determine, if the operation was successful. DX=6B6Fh indicates successful operation, any other value indicates an error.

To get information about the installed boards following the JIDA standard, the following procedure is recommended:

Call "Get Device ID" with CL=1. The name of the first device installed will be returned. If result was "Board exists" (CL=0), increment CL and call "Get Device ID" again. Repeat until result is "Board not present" (CL#0). You now know the names of all boards within your system that follow the JIDA standard. More information about a specific board may then be obtained by calling the appropriate inquiry function with the board's number in CL.

Warning! Association between board and board number may change due to configuration changes. Do **not** rely on any association between board and board number. Instead, always use the procedure described in the preceding paragraph first, to determine the association between board and board number.

The source of a Turbo-Pascal unit showing JIDA access is contained on the disk that is shipped with this manual.

Get Manufacturer ID	Int 15h	
Input:	AX = EA00h	DX = 4648h
	CL = Board number (1=first board a.s.o.)	
	ES:BX = Pointer to destination data area	
Output:	CL=0: Board present	DX=6B6Fh: Function successful
	CL###0: Board not present	DX###6B6Fh: Error
Description:	If CL=0 and DX=6B6Fh, then 4 Byte manufacturer ID were copied to the area pointed to by ES:BX By default, the result will be "□□ □". Note: There is no ending zero byte. Function must be implemented on every device supporting the JIDA.	

Get Device ID	Int 15h	
Input:	AX = EA01h	DX = 4648h
	CL = Board number	
	ES:BX = Pointer to destination data area	
Output:	CL=0: Board present	DX=6B6Fh: Function successful
	CL###0: Board not present	DX###6B6Fh: Error
Description:	If CL=0 and DX=6B6Fh, then 7 Byte device ID were copied to area pointed to by ES:BX By default, the result will be "JUMP". Note: There is an ending blank but no ending zero byte. Function must be implemented on every device supporting the JIDA.	

Get Manufacturing Date	Int 15h	
Input:	AX = EA02h	DX = 4648h
	CL = Board number	
Output:	CL=0: Board present	DX=6B6Fh: Function successful
	CL###0: Board not present	DX###6B6Fh: Fn. not implemented
	BX = Manufacturing date	
Description	If CL=0 and DX=6B6Fh, then BX = Manufacturing date. Date format is the same as used for DOS files: Bit0..4: Day Bit5..8: Month Bit9..15: Years since 1980	

Get Serial Number	Int 15h	
Input:	AX = EA03h	DX = 4648h
	CL = Board number	
	ES:BX = Pointer to destination data area	
Output:	CL=0: Board present	DX=6B6Fh: Function successful
	CL###0: Board not present	DX###6B6Fh: Fn. not implemented
Description:	If CL=0 and DX=6B6Fh, then 10 Byte serial number were copied to area pointed to by ES:BX The result is different for each single MULTI4. Note: There is no ending zero byte.	

Get Hardware Revision	Int 15h	
Input:	AX = EA04h CL = Board number	DX = 4648h
Output:	CL=0: Board present CL###0: Board not present BH = Major revision number BL = Minor revision number	DX=6B6Fh: Function successful DX###6B6Fh: Fn. not implemented

Get Firmware Revision	Int 15h	
Input:	AX = EA05h CL = Board number	DX = 4648h
Output:	CL=0: Board present CL###0: Board not present BH = Major revision number BL = Minor revision number	DX=6B6Fh: Function successful DX###6B6Fh: Fn. not implemented

Get Last Repair Date	Int 15h	
Input:	AX = EA06h CL = Board number	DX = 4648h
Output:	CL=0: Board present CL###0: Board not present BX = Last repair date.	DX=6B6Fh: Function successful DX###6B6Fh: Fn. not implemented
Description:	If CL=0 and DX=6B6Fh, then BX = Last repair date. For date format see function "Get Manufacturing Date". If board was never repaired, result will equal to manufacturing date.	

Read Running Time Meter	Int 15h	(not implemented with MULTI-4)
Input:	AX = EA07h CL = Board number	DX = 4648h
Output:	CL=0: Board present CL###0: Board not present BX = Running time (hours) CH = Overflow counter	DX=6B6Fh: Function successful DX###6B6Fh: Fn. not implemented

Read Boot Counter	Int 15h	(not implemented with MULTI-4)
Input:	AX = EA08h CL = Board number	DX = 4648h
Output:	CL=0: Board present CL###0: Board not present BX = Boot counter	DX=6B6Fh: Function successful DX###6B6Fh: Fn. not implemented

Get JIDA revision level	Int 15h	
Input:	AX = EA09h CL = Board number	DX = 4648h
Output:	CL=0: Board present CL###0: Board not present BH = Major revision number (currently 2) BL = Minor revision number (currently 0)	DX=6B6Fh: Function successful DX###6B6Fh: Fn. not implemented

Get Device Subversion	Int 15h	
Input:	AX = EA0Ah	DX = 4648h
	CL = Board number	
	ES:BX = Pointer to destination data area	
Output:	CL=0: Board present	DX=6B6Fh: Function successful
	CL###0: Board not present	DX###6B6Fh: Error
Description:	If CL=0 and DX=6B6Fh, then 7 Byte device Subversion ID were copied to area pointed to by ES:BX Possible results are MLCD-M_ AT96-MULTI-4M MLCD-K_ AT96-MULTI-4K XLCD___ ISA96-MULTI-4 ILCD____ ISA-MULTI-4 ALCD___ ISA-VGALCD-4A TLCD___ ISA-VGALCD-4T PLCD___ PC/104-VGALCD-4 Note: There is no ending zero byte.	

Get Numeric Device ID	Int 15h	
Input:	AX = EA0Bh	DX = 4648h
	CL = Board number	
Output:	CL=0: Board present	DX=6B6Fh: Function successful
	CL###0: Board not present	DX###6B6Fh: Fn. not implemented
	BX=0504h identifies the MULTI-4	

Get Contrast setting	Int 15h	
Input:	AX = EA20h	DX = 4648h
	CL = Board number	
Output:	CL=0: Board present	DX=6B6Fh: Function successful
	CL###0: Board not present	DX###6B6Fh: Fn. not implemented
	CH = Actual contrast (value range 0..63)	

Set Contrast	Int 15h	
Input:	AX = EA21h	DX = 4648h
	CL = Board number	
	CH = New contrast value	
Output:	CL=0: Board present	DX=6B6Fh: Function successful
	CL###0: Board not present	DX###6B6Fh: Fn. not implemented
Description:	Valid value range for contrast is 0..63. Other values will be ignored.	

Disable DC/DC Converter	Int 15h	
Input:	AX = EA22h	DX = 4648h
	CL = Board number	
Output:	CL=0: Board present	DX=6B6Fh: Function successful
	CL###0: Board not present	DX###6B6Fh: Fn. not implemented

Enable DC/DC Converter	Int 15h	
Input:	AX = EA23h	DX = 4648h
	CL = Board number	
Output:	CL=0: Board present	DX=6B6Fh: Function successful
	CL###0: Board not present	DX###6B6Fh: Fn. not implemented

Get Matrix Translation Table	Int 15h	
Input:	AX = EA30h	DX = 4648h
	CL = Board number	
	ES:BX = Pointer to destination data area	
Output:	CL=0: Board present	DX=6B6Fh: Function successful
	CL###0: Board not present	DX###6B6Fh: Fn. not implemented
Description:	160 Byte matrix keyboard translation table will be copied to area pointed to by ES:BX	

Set Matrix Translation Table	Int 15h	
Input:	AX = EA31h	DX = 4648h
	CL = Board number	
	ES:BX = Pointer to new translation table (holds 160 bytes)	
Output:	CL=0: Board present	DX=6B6Fh: Function successful
	CL###0: Board not present	DX###6B6Fh: Fn. not implemented
Description:	New matrix keyboard translation table will be copied from area pointed to by ES:BX	

Get Matrix Translation Entry	Int 15h	
Input:	AX = EA32h	DX = 4648h
	CL = Board number	
	BH = Matrix line (0..15 allowed)	
	BL = Matrix row (0..9 allowed)	
Output:	CL=0: Board present	DX=6B6Fh: Function successful
	CL###0: Board not present	DX###6B6Fh: Fn. not implemented
	CH = Table entry (undefined if BH or BL are invalid)	

Set Matrix Translation Entry	Int 15h	
Input:	AX = EA33h	DX = 4648h
	CL = Board number	
	BH = Matrix line (0..15 allowed)	
	BL = Matrix row (0..9 allowed)	
	CH = New Entry	
Output:	CL=0: Board present	DX=6B6Fh: Function successful
	CL###0: Board not present	DX###6B6Fh: Fn. not implemented

Read User Byte from EEPROM	Int 15h	
Input:	AX = EA40h	DX = 4648h
	CL = Board number	
	BH = Number of byte to read (0..31 allowed)	
Output:	CL=0: Board present	DX=6B6Fh: Function successful
	CL###0: Board not present	DX###6B6Fh: Fn. not implemented
	BL = Value read	

Write User Byte to EEPROM	Int 15h	
Input:	AX = EA41h	DX = 4648h
	CL = Board number	
	BH = Number of byte to write (0..31 allowed)	
	BL = Value to write	
Output:	CL=0: Board present	DX=6B6Fh: Function successful
	CL###0: Board not present	DX###6B6Fh: Fn. not implemented

Get Number of User Bytes available in EEPROM	Int 15h	
Input:	AX = EA42h	DX = 4648h
	CL = Board number	
Output:	CL=0: Board present	DX=6B6Fh: Function successful
	CL###0: Board not present	DX###6B6Fh: Fn. not implemented
	BL = Number of Bytes available (32)	

Read OC Output State	Int 15h	
Input:	AX = EA50h	DX = 4648h
	CL = Board number	
Output:	CL=0: Board present	DX=6B6Fh: Function successful
	CL###0: Board not present	DX###6B6Fh: Fn. not implemented
	CH = Actual output state	
Description:	Data format: CH = xxxxxxba, where b = OC1, a = OC0	

Switch OC Outputs	Int 15h	
Input:	AX = EA51h	DX = 4648h
	CL = Board number	
	CH = New output state	
Output:	CL=0: Board present	DX=6B6Fh: Function successful
	CL###0: Board not present	DX###6B6Fh: Fn. not implemented
Description:	Data format: CH = xxxxxxba, where b = OC1, a = OC0	

VGA BIOS Extensions

The MULTI-4 video BIOS supports all standard VGA BIOS Interrupt 10h video service functions. In addition, the BIOS provides extensive support for various features of the on board video controller. These functions are available as extended functions under Interrupt 10h. All extended function calls will preserve the CPU registers, except those used to pass information from the BIOS.

Inquire VGA type	Int 10h	
Input:	AX = 1200h	BL = 80h
Output:	AL = Controller Type	(always 23h = CL-GD6235)
	AH = Capabilities	bit2 = reverse video support
		bit3 = hardware expanded text support
		bit7 = LCD panel support
	BL = Silicon revision number	(80h if not available)

Inquire BIOS Version Number	Int 10h	
Input:	AX = 1200h	BL = 81h
Output:	AH = Major BIOS version #	
	AL = Minor BIOS version #	

Reverse Video	Int 10h	
Input:	AH = 12h	BL = 89h
	AL = mode:	0 = text reverse, graphic normal
		1 = text normal, graphic normal
		2 = text reverse, graphic reverse
		3 = text normal, graphic reverse

Bold Mode	Int 10h	
Input:	AH = 12h	BL = 8Bh
	AL = mode:	0 = enable bold font mode
		1 = disable bold font mode

Contrast enhancement	Int 10h	
Input:	AH = 12h	BL = 8Ch
	AL = enhancement mode:	
	0 = Enable b/w enhancement	3 = Foreground enhancement
	1 = disable contrast enhancement	4 = Foreground & Background enh.
	2 = reserved	

Set gray shading	Int 10h	
Input:	AH = 12h	BL = 95h
	AL = shading mode:	0 = NTSC shading
		1 = Green output only

Expanded mode	Int 10h	
Input:	AH = 12h	BL = 8Fh
	AL = expanded mode:	0 = enable
		1 = disable

Vertical positioning Control	Int 10h	
Input:	AH = 12h	BL = 90h
	AL = display position:	0 = center of panel
		1 = top of panel

Power save modes	Int 10h	
Input:	AX = 12h	BL = 94h
	BH = Time interval in minutes	
	AL 0 = Disable power save modes	
	1 = Standby immediately	
	2 = Suspend immediately	
	3 = Standby after BH minutes without video access	
	4 = Standby after BH minutes without port 60h activity	
	7 = Standby after BH minutes without port 60h activity or video access	
	8 = Backlight off immediately	
	9 = Backlight off after BH minutes without video acc.	
	A = Backlight off after BH minutes without port 60h activity	
	B = Backlight off after BH minutes w/o port 60h activity or video access	

Set panel frequency	Int 10h	
Input:	AH = 12h	BL = 9Dh
	AL = panel frequency	0 = Low frequency
		1 = Normal frequency

Set display type	Int 10h	
Input:	AH = 12h	BL = 92h
	AL = display type	0 = flat panel only
		1 = CRT only
		2 = SimulScan

Inquire User options	Int 10h	
Input:	AX = 1200h	BL = 9Ah
Output:	AX = Options set	bit0 = Vertical position control
		bits4:2 = Horizontal monitor type
		bit7 = LCD frequency
		bit8 = Display mode
		bit9 = Expand mode
		bit11 = Text mode reversed
		bit13 = Bold font
		bit14 = VGA refresh
	BX = Options set	bits2:0 = Contrast enhancement
		bit 3 = Graphics gray scale
		bits5:4 = Text gray scale
		bit7 = Graphics reverse video
		bit9 = SimulScan
	CX = reserved	
	DX = Power options:	bits0:7 = Standby timer
		bits8:11 = Power save mode

Read Monitor ID	Int 10h	
Input:	AX = 1200h	BL = A1h
Output:	BH = Monitor ID	
	BL = Monitor detected:	0 = Color display
		1 = Grayscale display
		2 = No display detected

Set Monitor Type	Int 10h	
Input:	AH = 12h AL = Monitor Type to set	BL = A2h 0 = VGA (31.5 kHz) 1 = 8514 (31.5 & 35.5 kHz) 2 = Super VGA (31.5 - 35.1 kHz) 3 = Extended Super VGA (31.5 - 35.5 kHz) 4 = Multi-freq. (31.5 - 37.9 kHz) 5 = Extended Multi-freq. (31.5 - 48.4 kHz) 6 = 31.5 - 56.5 kHz 7 = 31.5 - 64.0 kHz

Set Refresh Type	Int 10h	
Input:	AH = 12h AL = High refresh enable	BL = A3h 0 = disable 1 = enable

Available Video Modes

Mode #	Vesa #	Display Mode	Characters/Pixels	# of Colors
00h/01h	-	Text	40 * 25	16 out of 256
02h/03h	-	Text	80 * 25	16 out of 256
04h/05h	-	Graphics	320 * 200	4 out of 256
06h	-	Graphics	640 * 200	2 out of 256
07h	-	Text	80 * 25	mono
0Dh	-	Graphics	320 * 200	16 out of 256
0Eh	-	Graphics	640 * 200	16 out of 256
0Fh	-	Graphics	640 * 350 / * 400	mono
10h	-	Graphics	640 * 350	16 out of 256
11h	-	Graphics	640 * 480	2 out of 256
12h	-	Graphics	640 * 480	16 out of 256
13h	-	Graphics	320 * 200	256 out of 256
14h	-	Text	132 * 25	16 out of 256k
54h	10Ah	Text	132 * 43	16 out of 256k
55h	109h	Text	132 * 25	16 out of 256k
58h/6Ah	102h	Graphics	800 * 600	16 out of 256k
5Ch	103h	Graphics	800 * 600	256 out of 256k
5Dh	104h	Graphics	1024 * 768	16 out of 256k

Access to the General Purpose Input Pin (GPI)

One TTL level input pin is placed on the LCD panel connector for general purpose. The status of this pin can be read from one of the video controllers extended registers. To read the status of the GPI pin, follow one of the following examples.

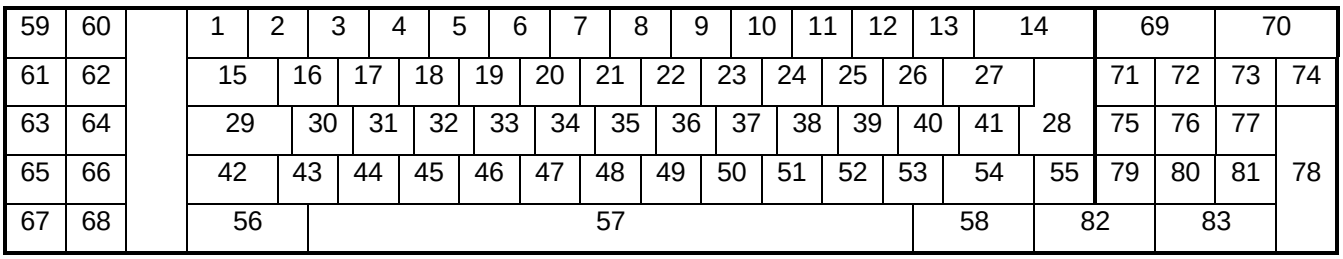
Assembler GPI Access:

```
      MOV   DX,3C4h    ; Sequencer Index register
      MOV   AL,6       ; Index of unlock register
      OUT   DX,AL      ; Select unlock register
      MOV   DX,3C5h    ; Sequencer data register
      MOV   AL,12h     ; Value to unlock registers
      OUT   DX,AL      ; Unlock extended registers now
      MOV   DX,3C4h    ; Sequencer index register
      MOV   AL,8       ; Index of input register
      OUT   DX,AL      ; Select input register now
      MOV   DX,3C5h    ; Sequencer data register
      IN    AL,DX      ; Read input register
      AND   AL,01      ; Mask out the interesting bit
      MOV   DX,3C4h    ; Sequencer Index register
      MOV   AL,6       ; Index of unlock register
      OUT   DX,AL      ; Select unlock register
      MOV   DX,3C5h    ; Sequencer data register
      MOV   AL,0h      ; Value to lock registers again
      OUT   DX,AL      ; Unlock extended registers now
```

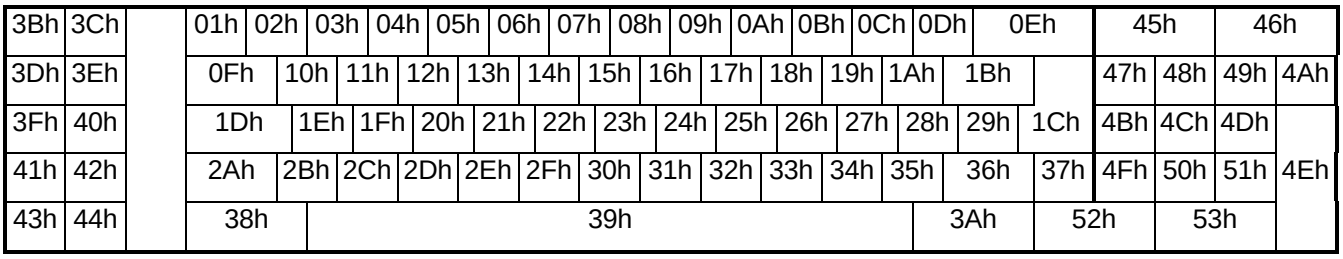
Pascal GPI access:

```
Port[$3C4]:=6;               (* Select unlock register       *)
Port[$3C5]:=$12;             (* Unlock extended registers    *)
Port[$3C4]:=8;               (* Select input data register   *)
Result:=Port[$3C5] AND $01   (* Read GPI bit                 *)
Port[$3C4]:=6;               (* Select unlock register       *)
Port[$3C5]:=$12;             (* Unlock extended registers    *)
```

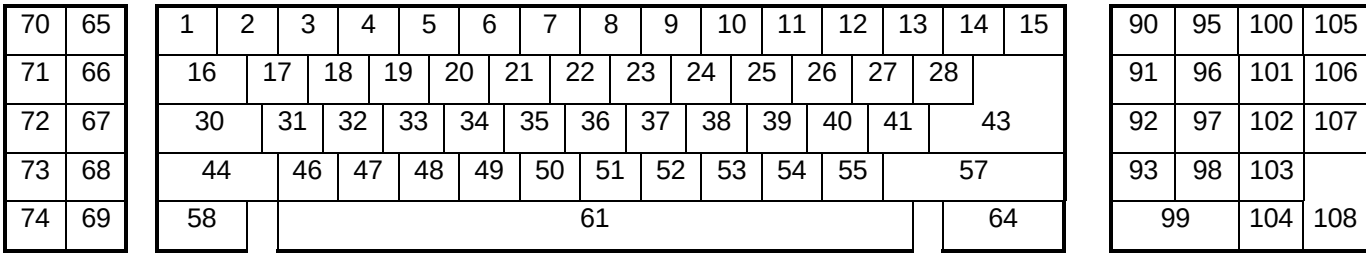
Scan Codes



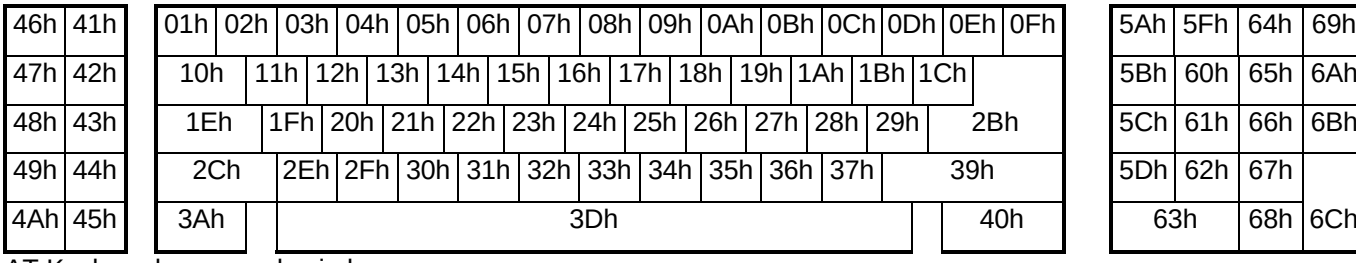
PC/XT Keyboard, scan codes in decimal



PC/XT Keyboard, scan codes in hex



AT Keyboard, scan codes in decimal



AT Keyboard, scan codes in hex

The MULTI-4 Support Disk

The MULTI-4 Support disk contains the following files:

`MULTISxx.EXE`	Utility to set up the IO controller and the matrix keyboard. See section "Using the MULTISET Utility" for description.
`SIMUL.EXE`	Switches your MULTI-4 to SimulScan mode.
`JIDA_ACC.PAS`	Source of a TurboPascal unit demonstrating the use of the JIDA BIOS extensions.
`TEST.PAS`	Sample program that uses JIDA_ACC.PAS to inquire information, which are being displayed on the screen.
`TEST.EXE`	TEST.PAS, already compiled and ready to run.

Hardware Controller Descriptions

Video Controller Type CL 6235

Features

- Integrated RAMDAC
- Integrated programmable frequency synthesizer
- Integrates Color STN panel support
- Simultaneous CRT and LCD (SimulScan TM) operation
- 64 -shade grayscale at 640 * 480 resolution on monochrome STN LCD
- Resolution up to 1024 * 768 with 16 colors on CRTs
- 800 * 600 with 256 colors on CRTs
- 132-column text modes on CRTs
- Low-power CMOS technology in a 160 pin-QFP package

FDC37C651 I/O-Controller Description

I/O - General Description

All the I/O functions are realized with a single I/O-controller from SMC type FDC37C651.

- 2 serial interfaces : COMA, COMB
- Printer interface
- Floppy interface
- IDE- Harddisk interface.

The FDC37C651 Universal Peripheral Controller is a single chip controller offering a complete I/O solution for the PC-XT and PC-AT environments. It provides one enhanced parallel port (printer/bi-directional), two 16450 UARTs, one IDE XT/AT hard disk interface and a floppy disk controller. The configuration is software controllable and completely supported by the system BIOS. Power management for the FDC37C651 includes modular power down for each port. The FDC37C651 features 24 mA drivers for the output buffers, including the host data bus and parallel port data bus. The floppy output drivers are capable of sinking 48 mA.

I/O - Features

- For MOTHERBOARD Application with configuration via software.
- Low Power CMOS, 100 pin PQFP Package.
- On Chip Power Management Features, Software configurable.
- 100% IBM PC-XT/AT Compatibility.
- 24 mA AT/XT Bus Interface Buffers.
- Schmitt Trigger Input on Reset Pin and FDC Interface Inputs.
- Two 16450 Compatible UARTs
- One IBM PC_XT/AT Compatible Enhanced (bi-directional) Parallel Port.
- 24 mA Parallel Port Output Drivers.
- IDE Interface (For Embedded AT & XT Hard Drives).
- Single 24 MHz Crystal/Oscillator for UART and Floppy Disk Controller.
- Fully uPD72065 and IBM-BIOS Compatible Floppy Controller.
- 48 mA floppy drive interface buffers
- Data rate and drive control registers
- Two pin programmable precompensation modes
- Support two floppy drives directly
- DMA enable logic

- Support 250 kb/sec, 300 kb/sec, 500 kb/s data rates

Serial Ports

Introduction

Two equivalent NS16450 UARTs are implemented on the FDC37C651. The serial port is fully compatible to 16450 ACE registers. The programmable features allow data rates ranging from 50 baud to 115.2 kbit/s; 5 to 8 bit character size with 1 start and 1, 1.5, 2 stop bits; even, odd, sticky, or no parity; and prioritized interrupts. An interrupt from the corresponding UART is enabled or disabled (tri-stated) using the OUT2 bit. If a "1" is written to OUT2, interrupt is enabled. Writing "0" tri states the interrupt. The primary serial port base address is programmed via bit 0,1 of Configuration Register 2. The secondary serial port base address is programmed via bits 4 and 5 of Configuration Register 2. An on-chip baud rate generator divides the input clock or crystal frequency by a number from 1 to 65535. The frequency is used for both receiving and transmitting serial data.

Serial to parallel conversion is performed on received data and parallel-to-serial conversion is performed on transmitted data. Status of the UART is available at any time. To access it, the CPU reads the appropriate status register in the FDC37C651. The current state and type of a transfer are contained in this status information as are details regarding any errors encountered. The conditions under which the processor will be interrupted and the interrupt line to be used are programmable. Control lines are provided to permit interfacing to a MODEM. Internal diagnostics are supported that permit simulation of break, parity overrun and framing error conditions as well as operation in loopback mode.

Serial Port Registers

The following sections describe the details of the serial ports. Since the function of the two serial ports are identical, the descriptions are applied for both of them.

Addressing of the accessible UART registers is shown in the Table below. The base address of all registers is software programmable during the configuration sequence (see the section entitled "FDC37C651 Configuration"). UART registers are located at sequentially increasing addresses above this base address. The FDC37C651 contains two UARTs which contain a set of registers described below.

DRAB	A2	A1	A0	OFFSET	Register Name
0	0	0	0	0 H	Rec.Buff.Reg. (R)
0	0	0	0	0 H	Tr.Buff.Reg. (W)
0	0	0	1	1 H	In.En.Reg. (R/W)
x	0	1	0	2 H	In.Fl.Reg. (R/W)
x	0	1	1	3 H	By.Fo.Reg. (R/W)
x	1	0	0	4 H	Mo.Co.Reg. (R/W)
x	1	0	1	5 H	Li.St.Reg. (R/W)
x	1	1	0	6 H	Mo.St.Reg. (R/W)
x	1	1	1	7 H	Sc.Pad Reg. (R/W)
1	0	0	0	0 H	Divisor LSB (R/W)
1	0	0	1	1 H	Divisor MSB (R/W)

Rec. Buff.Reg = Received Buffer Register
Tr.Buff.Reg = Transmit Buffer Register
In.En.Reg. = Interrupt Enable Register
In.Fl.Reg. = Interrupt Flag Register
By.Fo.Reg. = Byte Format Register
Mo.Co.Reg. = Modem Control Register
Li.ST.Reg = Line Status Register
Mo.ST.Reg. = Modem Status Register
Sc.Pad Reg. = Scratch Pad Register

X = Don't Care
MSB = Most Significant Byte
LSB = Least Significant Byte
DRAB = Divisor Register Address Bit (Bit 7 of Byte Format Register)

Receive Buffer (RB)

Offset=0H, Read only, DRAB=0

This register holds the incoming data byte. Bit 0 is the least significant bit, which is transmitted and received first. Double buffering is supported by the FDC37C651. This scheme uses an additional shift register (the Receive Shift Register is not user accessible) to assemble the incoming byte before it is loaded into the Receive Buffer.

Transmit Buffer (TB)

Offset=0H, Write only, DRAB=0

This register holds the data byte to be sent. Bit 0 is the least significant bit, which is transmitted and received first. Double buffering is supported by the FDC37C651. This scheme uses a shift register (The Transmit Shift Register is not user accessible) which is loaded from the Transmit Buffer. The transmitted byte is then shifted out of the Transmit Shift Register to the TxD pin.

Interrupt Enable Register (IER)

Offset=1H, Read/Write, DRAB=0

The low order 4 bits of the register control the enabling of each of the four possible types of interrupts. Setting a bit to a logic 1 enables the corresponding interrupt. It is possible to enable all, one, or some of the interrupt sources. Disabling all interrupts means that the interrupt flag register content is not valid and that none of the interrupt signals output by FDC37C651 can be triggered by a UART. All other portions of the UART are unaffected by the disabling of interrupts. The individual bit definition are as follows:

Bit 0: A logic 1 here causes an interrupt when the Receive Buffer contains valid data.

Bit 1: A logic 1 here causes an interrupt when the Transmit Buffer is empty.

Bit 2: A logic 1 here causes an interrupt when an error (Overrun, Parity, Framing or Break) has been encountered. The Line Status register must be read to determine the type or error.

Bit 3: A logic 1 here causes an interrupt when one of the bits in the MODEM Status register changes state.

Bit 4-7: These four bits are set to 0.

Interrupt Flag Register (IFR)

Offset=2H, Read/write, DRAB=X

When accessed, this register reports the highest pending interrupt. By reading it, the CPU can determine the source of the interrupt and can act accordingly. The Interrupt Flag Register (IFR) records the highest pending interrupt in bits 0 through 2. Other interrupts are temporarily disregarded (they are internally saved by the FDC37C651) until the highest priority one is serviced. Four levels of prioritized interrupts exist. In descending order of priority they are:

1. Line Status (highest priority) 3. Transmit Buffer empty

2. Receive Buffer full 4. MODEM Status (lowest priority)

Bit definitions for the IFR are as follows:

Bit 0: If this bit is a zero, an interrupt is pending and bits 1 and 2 can be read to determine the source of the interrupt. When this bit is a logic 1, no interrupts are pending. Note that this bit can be used in a polled environment to determine if an interrupt is pending. It can also be used for the same purpose with a hardwired interrupt priority scheme. In the latter case, Bits 1 and 2 of this register act as a pointer to an interrupt service routine.

Bits 1 and 2: As indicated in Table 2.1 below, these two bits specify the type and source of the interrupt.

Bits 3-7: These five bits are set to 0.

Byte Format Register (BFR)

Offest=3H, Read/write, DRAB=X

This read/write register contains format information for the serial line. Since it can be read, a separate copy of i content need not be kept in system memory. Bit definitions are as follows:

Bit 0 and 1: These specify the word length for received and transmitted characters. Start, stop and parity bits are not included in the word length value. The word lengths are:

Bit 0	Bit 1	Word Length
0	0	5 Bits
0	1	6 Bits
1	0	7 Bits
1	1	8 Bits

Bit 2: The combination of this bit and Bits 0 and 1 of this register determine the number of stop bits used with each transmitted character. The table below summarizes this information. Note that the receiver will ignore additional stop bits beyond the first regardless of the number of stop bits used when transmitting.

Bit 2	Word Length	No. of Stop Bits
0	-	1
1	5 Bits	1 1/2
1	6 Bits	2
1	7 Bits	2
1	8 Bits	2

Bit 3: A logic 1 in this bit enables parity generation (during transmission) and checking (during receipt). The parity bit is always after the last data bit but before the first stop bit. If enabled, a parity bit of the proper state (0 or 1) is generated such that the sum (carry ignored) of all data bits plus the parity bit produces either an even (even parity) or odd (odd parity) value.

Bit 4: This Even Parity bit controls parity sense. It is ignored unless Bit 3 is a logic 1. If Bits 3 and 4 are logic 1s (even parity), an even number of logic 1s will be transmitted and a parity error will be generated each time an odd number is received. If Bit 3 is a "1" and Bit 4 is a "0" (odd parity), an odd number of logic 1s will be transmitted and a parity error will be generated each time an even number is received.

Bit 5: This is the Force Parity bit. it ensures that the parity bit and sense (even or odd) match regardless of the sum normally used to determine parity. Thus if BFR Bits 3, 4 and 5 are all logic 1s (even parity), the parity bit transmitted will always be a "0" and a parity error will be detected if a logic 1 parity bit transmitted will always be a "1" and a parity error will be detected if a "0" parity bit is received.

Bit 6: This BREAK bit, when set to a logic 1, forces the transmitted data output pin TxD to a Spacing or logic 0 condition. This BREAK condition is terminated when Bit 6 is set to a "0". The operation of the transmitter logic is unaffected by the value of this bit; only the value of the TxD pin is affected. A BREAK condition is typically used to alert a terminal in a communications system. To prevent the transmission of erroneous data, follow the steps below:

Load a NULL character (all zeroes) into the Transmit Buffer.

Load Bit 6 (BREAK Bit) after the next Transmit Buffer Empty (TBE) occurs.

Time the length of the BREAK condition by continuing to load NULL characters into the Transmit Buffer and counting the number loaded.

clear the BREAK condition only after a Transmitter Empty (TEMT) condition occurs.

Bit 7: This Divisor Register Address Bit (DRAB) must be a logic 1 to permit access to the Divisor Registers. Access to all other internal UART registers requires that this bit be 0.

Modem Control Register (MCR)

Offset=4H, Read/Write, DRAB=X

This byte-wide register is used to manage the connection to an external MODEM or data set. Bit definitions are as follows:

Bit 0: This /DTR bit determines the state of the /DTR output pin. Setting Bit 0 to a logic 1 forces /DTR to its active state (logic 0). If Bit 0 is a logic 0 , /DTR will be inactive (logic 1). An external inverting

buffer is typically used (to insure the proper polarity of DTR/) when connecting a FDC37C651 /DTR output to a MODEM or data set.

Bit 1: This /RTS bit determines the state of the corresponding /RTS FDC37C651 output pin in a fashion identical to Bit 0 (see above).

Bit 2: This bit is used to control OUT1 bit. It does not have an output pin associate with this bit. It can be read or written by CPU.

Note: OUT1 is an Internal chip signal.

Bit 3: This bit is used to enable an interrupt (OUT2 pin of UART). When OUT2=0 (default) the serial interrupt is forced into high impedance. When OUT1=1 the serial interrupt output is enabled. Note: OUT2 is an Internal chip signal. In the normal mode (no loopback), this bit is OUT2. When OUT2=0 (default), the serial interrupt is forced into a high impedance. When OUT2=1, the interrupt output is enabled.

Bit 4: This Loopback bit is used for self-diagnostic purposes. If it is a logic 1:

1. The TxD FDC37C651 output pin is set to a logic 1 (Marking state) and it is disconnected from the output of the Transmit Shift Register.

2. The RxD FDC37C651 input pin is disconnected from the Receive Shift Register.

3. The input to the Receiver Shift Register is internally connected to the output of the Transmit shift Register.

4. All MODEM control input pins (/CTS, /DSR, /DCD, and /RI) are disconnected from the internal circuitry.

5.MODEM control output pins /DTR and /RTS are forced to the inactive state (logic 1).

6. MODEM control output /DTR is connected internally to MODEM control input DSR/,MODEM control output /RTS is internally connected to input /CTS, and MODEM Control Register (MCR) bit 2 determine the state of bit 6 of the MODEM Status Register (MSR). Bit 3 of the MCB controls bit 7 of the MSR.

7. Data which is transmitted will immediately be received, permitting the CPU to verify the data path internal to the FDC37C651 and its connection to the CPU.

While operating in diagnostic loopback mode, interrupts are disabled. Interrupts are controlled by the Interrupt Enable register. Interrupts which are due to MODEM signals operate as documented, although the source is now the lower 4 bits of the MODEM Control Register rather than the MODEM input pin signals.

Bit 5, 6 and 7: These bits are set to 0.

UART Interrupt Specifications (Interrupt Flag Register)

Bit 2	Bit 1	Bit 0	Priority	Type	Source	Serviced by
0	0	1	NO INTERRUPT PENDING			
1	1	0	Highest	Line Status	Overrun Error or Parity Error or Framing Error or Break Interrupt	Read Line Status Register
1	0	0	Second	Receive Buffer Full	Receive Data	Read Receive
0	1	0	Third	Transmit Buffer Empty	Transmit Buffer	Read IFR or Write transmit buffer
0	0	0	Fourth	Modem Status	Clear to Send or Data Set Ready or Ring Indicator or Carrier Detect	Read Modem Status Register

Line Status Register (LSR)

Offset=5H, Read Only, DRAB=X

This byte-wide register supplies serial link status information to the CPU. A Receive

Line Status interrupt is caused by one of the conditions flagged by Bits 1 through 4

this register. It is read-only. Writes to it are used at the factory for testing purposes and are not recommended. Bit definitions are as follows:

Bit 0: This Receive Buffer Full (RGF) bit is set to a logic 1 when an incoming character has been transferred from the Receive Shift Register to the Receive Buffer. Reading the Receive Buffer resets it to a logic 0.

Bit 1: This Overrun Error bit is set to a logic 1 when a new character is transferred into the Receive Buffer before the previously received character was read by the CPU. The previously received character is lost. When the CPU reads the LSR, the Overrun Error bit is reset to a "0".

Bit 2: This Parity Error bit is set to logic 1 whenever a parity error is detected (received character has a parity other than that selected). Reading the LSR resets this bit to a0.

Bit 3: This Framing Error bit is set to a logic 1 when an incoming character has no stop bit after the last data bit or (if parity is enabled) after the parity bit. A valid stop bit is the presence of a Mark condition (logic 1) in the proper time slot after the last data bit or the parity bit. Reading the LSR resets this bit to a "0".

Bit 4: This BREAK interrupt bit will be a logic 1 if a Space condition (logic 0) is present on the RxD line for an entire character time (start bit time, plus data bit times, plus parity bit time, plus stop bit time). Reading the LSR resets this bit to a "0".

Bit 5: This Transmit Buffer Empty (TBE) bit is set to a logic 1 when an outgoing character is loaded from the Transmit Buffer (TB) into the Transmit Shift Register. If the TBE interrupt is enabled, an interrupt will be generated when this bit is set. Writing a character to the TB resets this bit to a "0".

Bit 6: This Transmitter Empty (TEMT) bit will be set to a logic 1 when both the Transmit Buffer and the Transmit Shift Register are empty. When either of these two registers contains a character, this bit will be reset to a"0".

Bit 7: This bit is set to "0".

MODEM Status Register (MSR)

Offset=6H, Read/Write, DRAB=X

This byte-wide register holds the current value of the MODEM control lines. It also sets a bit (to a logic 1) each time one of these control lines changes state. Reading the MSR resets all of the Change bits to 0. A MODEM Status Interrupt is generated (if it is enabled) when Bit 0, 1, 2 or 3 is set to a "1". Bit definitions are:

Bit 0: This is the Clear To Send Changed bit. It is set to a "1" if /CTS line has changed state since the last time the MSR was read.

Bit 1: This is the Data Set Ready Changed bit. It is set to a 1 if the /DSR line has changed state since the last time the MSR was read.

Bit 2: This is the Rising Edge if Ring Indicator bit. It is set to a 1 if the /RI line has changed from a logic 0 to a logic 1 since the last time the MSR was read.

Bit 3: This is the Data Carrier Detect Changed bit. It is set to a 1 if the /DCD line has changed state since the last time the MSR was read.

Bit 4: This is the Clear To Send bit. It is the complement of the /CTS pin. When in diagnostic loopback mode, this bit is identical to the RTS bit in the MODEM Control Register (MCR).

Bit 5: This is the Data Set Ready bit. It is the complement of the /DSR pin. When in diagnostic loopback mode, this bit is identical to the DTR bit in the MCR.

Bit 6: This is the Ring Indicator bit. It is the complement of the /RI pin. In diagnostic loopback mode, it is controlled by Bit 2 of the MCR.

Bit 7: This is the Data Carrier bit. It is the complement of the /DCD pin. In diagnostic loopback mode, it is controlled by Bit 3 of the MCR.

Scratchpad Register

Offset=7H, Read/Write, DRAB=X

This byte-wide register has no affect on the UART within it is located. It can be used for any purpose by the programmer.

Effects of Hardware Reset

The table below details the effect of a hardware RESET on the UARTs located in a FDC37C651.

Register or Signal	Cause of Reset	Reset State
Interrupt Enable Register	Hardware RESET	All bits=logic 0
Interrupt Flag Register	Hardware RESET	Bit 0=logic 1 Other bits=logic 0
Byte Format Register	Hardware RESET	All bits=logic 0
MODEM Control Register	Hardware RESET	All bits=logic 0
Line Status Register	Hardware RESET	Bit 5,6=logic Other bits=logic0
Modem Status Register	Hardware RESET	Bit 0-3=logic 0 Bit4-7=Input Signal
TXD2 and TxD1	Hardware RESET	Logic 1 (high)
Receive Line Status Interrupt	Hardware RESET or Read LSR	logic 0 (low)
Receive Buffer Full Interrupt	Hardware RESET or Read RB	logic 0 (low)
Transmit Buffer Empty Interrupt	Hardware RESET or Read TB	logic 0 (low)
MODEM Status Interrupt	Hardware RESET or Read MSR	logic 0 (low)
/RTS2 and /RTS1	Hardware RESET	logic 1 (high)
/DTR2 and /DTR1	Hardware RESET	logic 1 (high)

Baud Rate Generation

The UART contains a programmable Baud Generator. The 24 MHz crystal oscillator frequency input is divided by 13 to provide a frequency of 1.8462 MHz. This is sent to the Baud Rate Generator and divided by the divisor for the UART. The output frequency of the Baud Rate Generator is 16 X the baud rate, ((divisor # =(frequency input)-(baud rate X16)). The output of the Baud Rate Generator drives the transmitter and receiver sections of the serial channel. Two 8-bit latches store the divisor in a 16 bit binary format. The Divisor Latch must be loaded during initialization to ensure proper operation of the Baud Rate Generator. Upon loading either of the Divisor Latches, a 16 bit Baud Counter is loaded.

Table below lists decimal divisors to use with a crystal frequency of 24 MHz. The oscillator input to the chip should always be 24 MHz to ensure that the Floppy Disk Controller timing is accurate and that the UART divisor are compatible with existing software. Using a divisor of zero is not recommended.

Divisor Baud Rate	Decimal Divisor for 16 x Clock by 1.8462 MHz Clock	Percent Error (Note 1) by 1,8462 MHz Clock
50	2304	0.001
75	1536	
110	1047	
134,5	857	0.004
150	768	
300	384	
600	192	
1200	96	
1800	64	
2000	58	0.005
2400	48	
3600	32	
4800	24	
7200	16	
9600	12	
19200	6	
38400	3	
56000	2	0.030
115200	1	

Note 1: The percent error for all Baud Rates is 0.002% except where indicated.

Parallel Port

Introduction

The Parallel Port is compatible to the IBM XT/AT Parallel Port, plus, offers a PS/2-like extended mode for bi-directional mode. When the parallel port is disabled via configuration register, all outputs are disabled and register contents are preserved. Upon power up, the control signals are inactive. The status register reflects the status signals.

Printer Interface Accessible Registers

The table below depicts the registers and I/O ports which are accessible for the parallel printer port. These are compatible with the IBM PC parallel port. Bit definitions for each of these registers are given after the diagram. All addresses for the parallel port are offsets from the base address specified during the FDC37C651 configuration process.

Data Latch (Port A)

Offset = 00H

This read/write register is located at an offset of 0H from the base address of the parallel port. Data written to this register is transmitted to the printer. Data pad from this is identical to that which was last written.

Printer Status Register (Port B)

Offset=01H

This read-only register is located at an offset of 1H from the base address of the parallel port. Bit definitions are as follows:

Bit7: Busy. This bit reflects the inverted state of the FDC37C651 BUSY input pin. A "0" means that the printer is busy and cannot accept data. A "1" indicates that the printer is ready to accept data.

Bit 6: /ACK. This bit reflects the state of the /ACK input pin. A "0" means that the printer has received a character and is ready to accept another. A "1" means that it is still reading the last character sent or data has not been received.

Bit 5: PE-Paper Empty. This bit reflects the state of the FDC37C651 PE input pin. A "1" indicate a paper end condition. A "0" indicates the presence of paper.

Bit 4: SLCT. This bit reflects the state of the FDC37C651 SLCT input pin. A "1" indicates the printer is on-line. A "0" means it is not selected.

Bit 3: ERROR/. This bit reflects the state of the FDC37C651 /ERROR input pin. A "0" means that an error condition has been detected. A "1" indicates no errors.

Bit 2-0: Reserved.

Printer Controls Register (Port C)

Offset = 02H

This read/write register is located at an offset of 02H from the base address of the parallel port. Bit definition are:

Bits 7-6: Reserved. Reset to 0.

Bit 5: Parallel Control Direction, valid in extended mode only (CR#1 <6>=1). In printer mode, the direction is always out, regardless of the state of this bit. In the extended mode, a "0" means an output/write condition. A "1" means an input/read condition.

BIT 4: IRQEN. This bit is used to enable or disable interrupts resulting from the printer /ACK changes from active to inactive. The CPU will be interrupted on the IRQ line specified in the FDC37C651 configuration RAM. A "0" means that IRQ is disabled.

Bit 3: SLCTIN (pin 49, SLCTIN). Used to drive the FDC37C651 SLCTIN output pin. A "1" selects the printer. A "0" means the printer is not selected.

Bit 2: /INIT (pin 50, /INIT). Used to control the FDC37C651 INIT output pin. A "0" (active low) starts the printer (50 µsec pulse minimum). A "0" initializes the printer.

Bit 1: AUTOFD (pin 51, AUTOFD/). Used to control the FDC37C651 AUTOFD output pin. A "1" causes the printer to generate a line feed after each line is printed. A "0" means no autofeed.

Bit 0: STROBE (pin 48, STROBE/). Used to control the FDC37C651 STROBE output pin. A "1" in this bit generates the active low pulse (0.5 µsec pulse minimum) which is required to clock data into the printer. There is a "0".5 µsec data setup time requirement before STROBE can be asserted. A "0" means there will be no strobe.

Table Summary of Accessible parallel Port Register

	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**	
XX0H	Data Latch (Port A)								
XX1H	BUSY	/ACK	PE	SLCTIN	/ERROR	R	R	R	Status (Port B)
XX2H									CTRL (Port C)

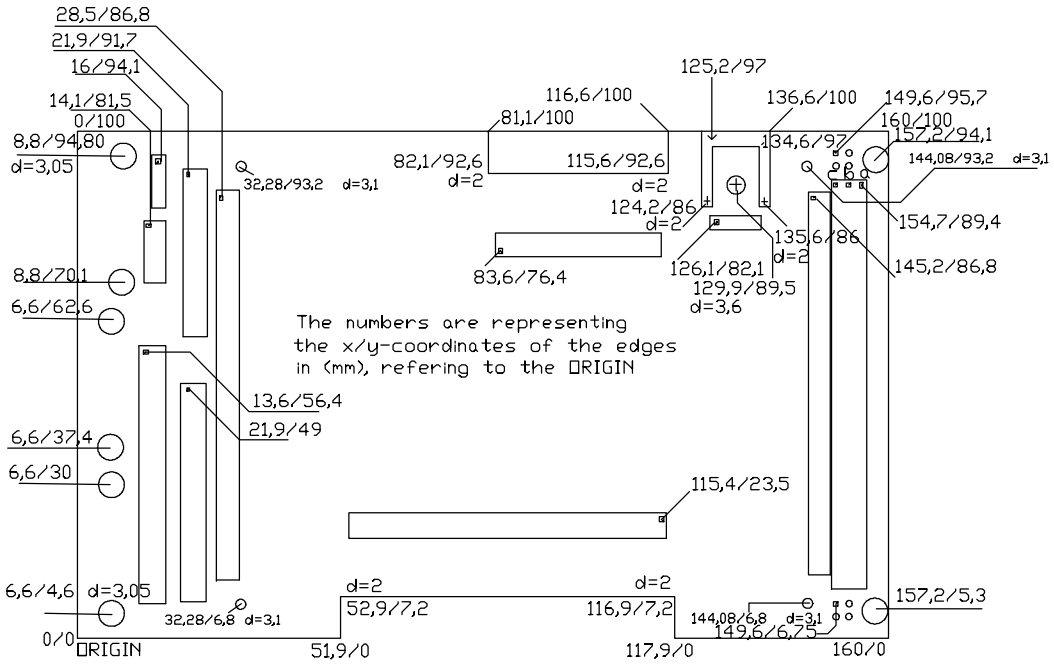
Note: R means Reserved

Specifications

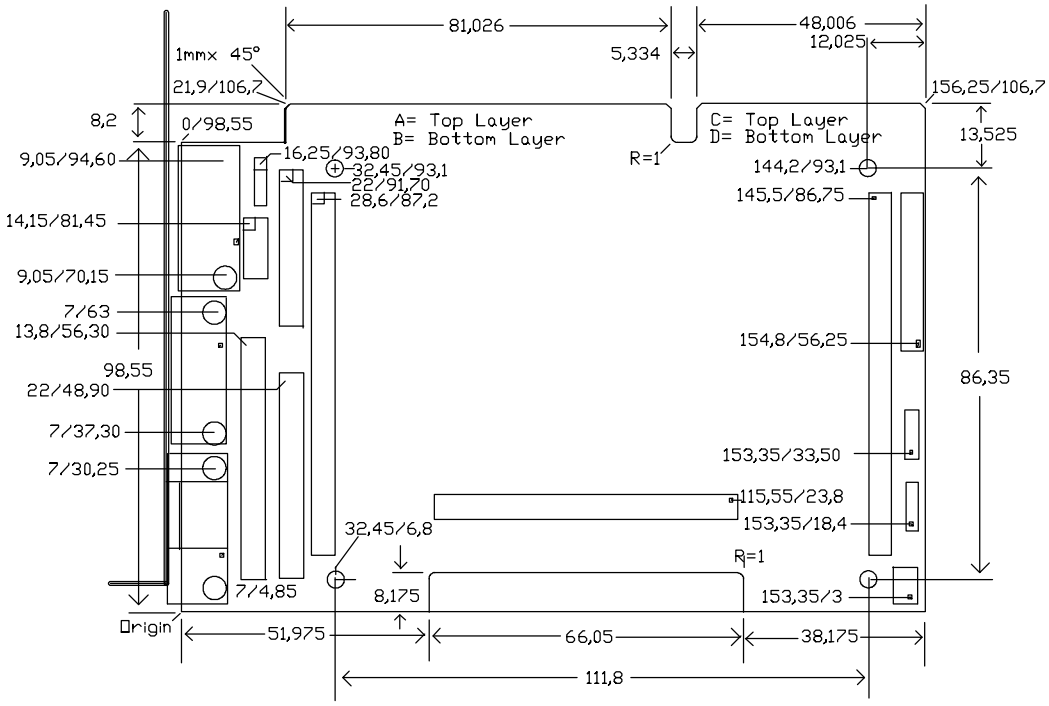
Mechanical Specifications

Weight approximately 130 g without front bezel

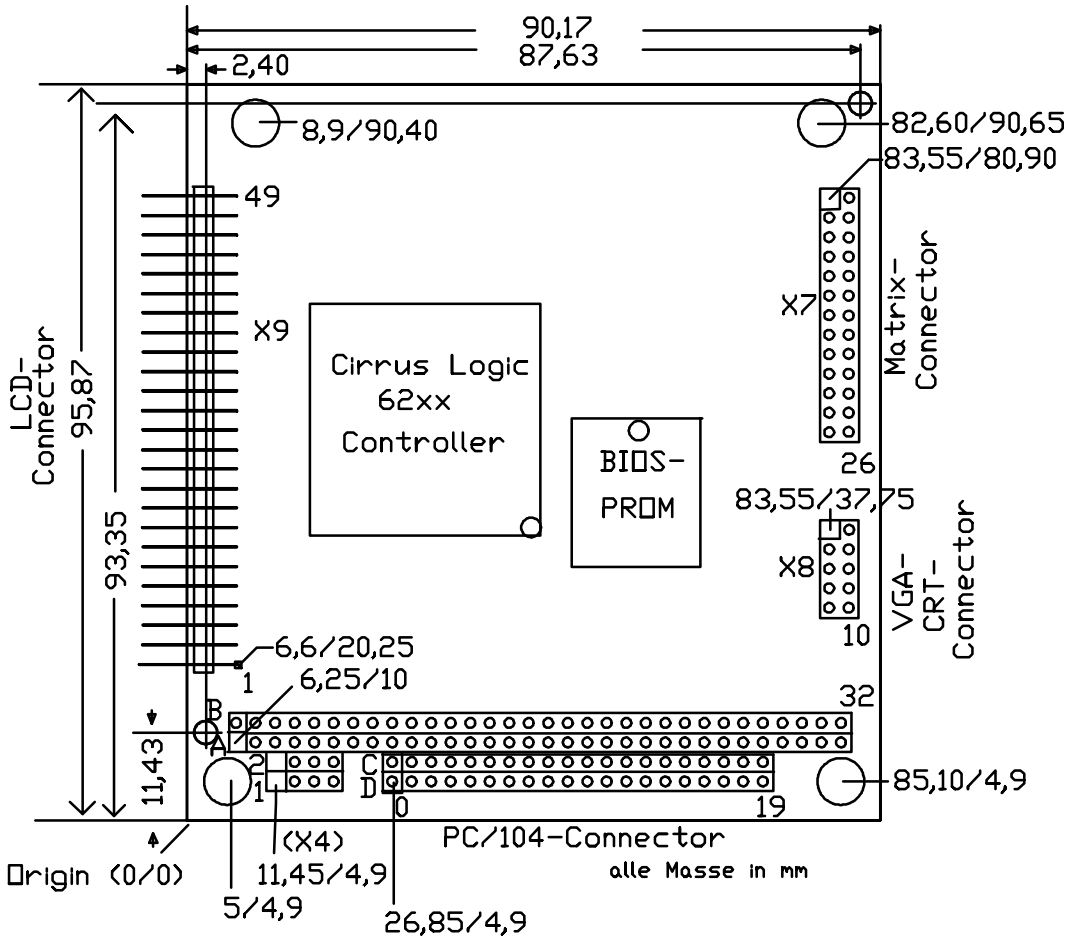
Mechanical dimensions of the AT96-MULTI-4 board.



Mechanical dimensions of the ISA-MULTI-4 board.



Mechanical dimensions of the PC/104-VGALCD-4 board.



Electrical Specifications

Supply voltage:	5V DC +/- 5%
Supply voltage ripple:	100 mV peak to peak 0 - 20 MHz
Supply current, typical:	400 mA at 25 °C
VGA connector signals:	
sync output current low:	20 mA at 0.5 V
sync output current high:	-12 mA at 2.4 V
RGB output voltage:	0 - 0.7 V with 75 Ohm external termination
RGB output current:	0 - 14 mA

Environmental Specifications

Temperature

operating	0 °C - 60 °C see note (*1)
non-operating	-40 °C - 85 °C

Thermal gradient

operating	25 °C per hour
non-operating	40 °C per hour

Relative Humidity

operating	10 % - 90 % RH non-condensing
non-operating	5 % - 95 % RH non-condensing

Mechanical

Shock	50G/20ms square wave maximum
Vibration	1G/0-600Hz, dwell not to exceed

Altitude

operating	0 - 3000 m
non-operating	0 - 5000 m

(*1) The maximum operating temperature is the maximum measurable temperature on any spot on the modules surface. It is the users responsibility to keep this temperature within the above specification.

Available Utilities

- Configuration-diskette with program "MULTISET.EXE". (Diskette is included in this Technical Manual)
- Keyboard Adapter; Keyboard adapter from DSUB9 to 5 pin DIN connector (orderno.: OPT-KBD-1)
- front panel 3 HE, 4TE width with 3 DSUB 9 extrusions (orderno.: OPT-FP-1-MULTI-4)
- front panel 3 HE, 4TE width with 1 DSUB 9 and 1 DSUB 25 extrusions (orderno.: OPT-FP-3-MULTI-4)
- cable for COMB ; DSUB9 male Connector to 10 pin Header cable (orderno.: KAB-DSUB9-2)
- cable for printer ; DSUB25 female Connector to 26 pin Header cable (orderno.: KAB-DSUB25-1)
- cable for IDE Harddisk 3 1/2 ; 40 pin flat cable with two 40 pin Header (orderno.: KAB-IDE-3,5")
- cable for IDE Harddisk 2 1/2 ; 44 pin flat cable with one 40 pin Header and one 44 pin Header (orderno.: KAB-IDE-2,5")
- Flat panel adaptation (new panel adaptation for AT96-MULTI-4 includes cable drawing, modified BIOS if necessary. Customer needs to supply panel, backlight converter, connectors and specification (orderno.: MULTI4-ADAPT)

Revision History

File-name	last alteration	author	alteration to previous version
MUL4D01.Doc	29.11.94	M. Benz	no
MLCD4D01.Doc	19.3.95	M. Benz	all MLCD-board are connected together
MLCD4D02.DOC	18.4.95	H. Fink	included information about BIOS Rev 2.x
MLCD4D03.DOC	12.5.95	M. Benz	added ISA-VGALCD-4 Block diagram etc. and I/O connector J18
MLCD4D03.DOC	17.8.95	Ig	changed cover page
MLCDD122.DOC	04.03.98	JH	filename changed and layout revised, special character fonts removed