Related Products	MOPSplus (P386) MOPS/386A (P389) MOPS/MZ and MOPSlcdMZ (PMZ1) MOPS/520 (P489)
Subject	DOT Matrix Display on PC/104 Products
Document Name	DotMatrixPC104_E111.doc
Usage	Common

1. REVISION HISTORY

Date	Document Name	Subjects added, changed, deleted	Changed by
11-Dec-02	DotMatrixPC104_E110.DOC	Initial release of Application Note	H. Bruhn
11-Dec-02	DotMatrixPC104_E110.DOC	Minor changes	H. Bruhn

2. TABLE OF CONTENTS

3. Introduction

Some PC/104 products of **Kontron Embedded Modules GmbH** support the Dot-Matrix display (also called character LCD) feature. This feature is supported by the following PC/104 products:

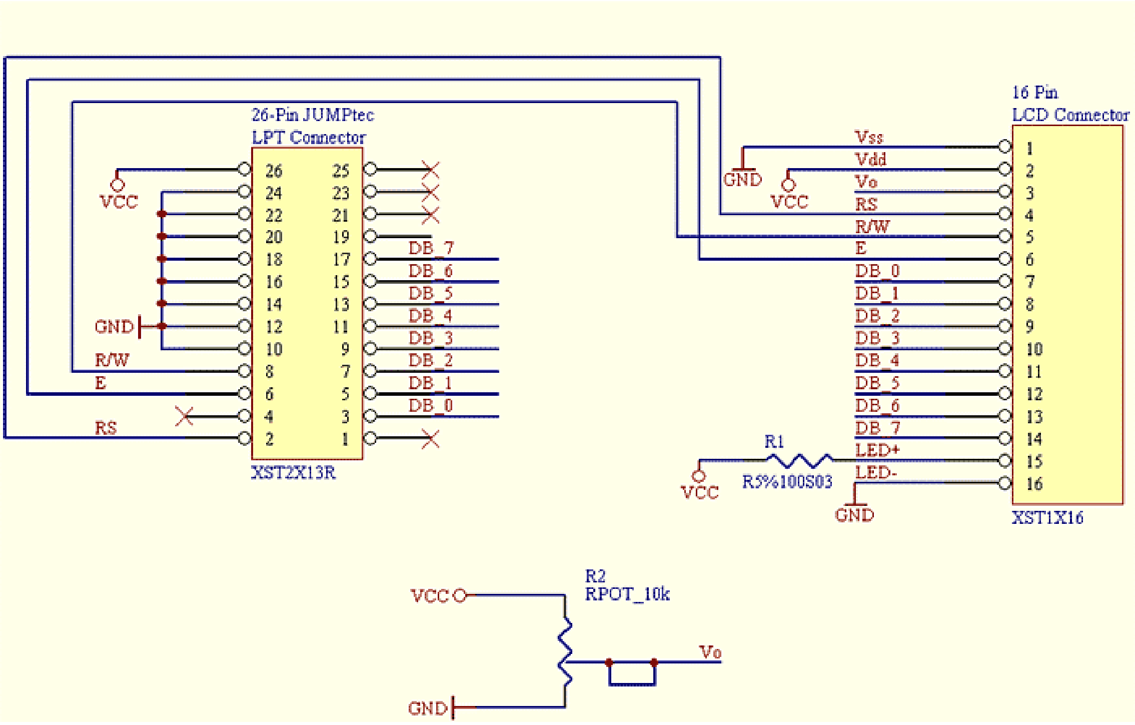
1. MOPSplus (P386)
2. MOPS/386A (P389)
3. MOPS/MZ and MOPSlcdMZ (PMZ1)
4. MOPS/520 (P489)

4. Hardware Information

The character LCD is interfaced through the signals of the parallel port (LPT). The printer port and the character LCD can not be used at the same time.
The system BIOS contains a compressed extension bios to support a DOT MATRIX module on the parallel port. This extension is copied to Shadow RAM if the ALCD support is enabled. Use the ALCDINIT.EXE, available from Kontron´s web pages or from your technical support, to enable or disable ALCD and to set the parameters of the attached module.
The character LCD can be programmed with standard INT10h software interrupt calls.
Some **Kontron Embedded Modules GmbH** boards can interface the character LCD only through the LPT port interface, some offer a special interface.

4.1. Character LCD on the LPT Port Interface

To drive a character LCD through the LPT port interface an adaptor is required. The following schematics shows how this adaptor could look:



Schematics adapter LPT to character LCD

26-pin Kontron LPT connector	LPT Signal	16-pin LCD connector	LCD Signal	Signal description LCD
1	/Strobe	-	-	-
3	Data 0	7	DB0	Data bit 0
5	Data 1	8	DB1	Data bit 1
7	Data 2	9	DB2	Data bit 2
9	Data 3	10	DB3	Data bit 3
11	Data 4	11	DB4	Data bit 4
13	Data 5	12	DB5	Data bit 5
15	Data 6	13	DB6	Data bit 6
17	Data 7	14	DB7	Data bit 7
19	/ACK	-	-	-
21	BUSY	-	-	-
23	PAPER out	-	-	-
25	SEL out	-	-	-
2	/AUTOFD	4	RS	RS register select on LCD 1 = data signal, 0 = instruction signal
4	/ERROR	-	-	Not used
6	/INIT	6	E	EN enable signal for data assumption
8	SEL in	5	R/W	R/W read/write from /to LCD 1 = read mode, 0 = write mode
26	VCC	2	Vdd	+ 5 V
10, 12, 14, 16, 18, 20, 22, 24.	GND	1	Vss	Signal Ground
-	-	3	Vo	Input voltage for LCD (0V to +5V)
-	-	15	LED+	Backlight anode
-	-	16	LED-	Backlight cathode

When connecting the character LCD through the LPT connector, additional circuitry for the "LCD drive voltage" and "Backlight anode voltage" is required. See schematics.

4.2. Character LCD on special Interface

An interface with 16 signals is offered for connecting LCD-DOT-Matrix to the MOPS/plus and MOPS/386A
The LCD-DOT-Matrix Interface is using the signals of the onboard parallel port. In most cases this will cause problems when the parallel port is used with any device parallel to the LCD. A customer cannot use a DOT Matrix display and a printer at the same time.

Pinout 16-pin LCD con.	Signal name	Function for LCD
1	GND	Ground
2	VCC	Supply voltage for logic
3	VEE	LCD drive voltage (contrast adjusted through potentiometer 0 to +5V)
4	RS	RS register select on LCD 1 = data signal, 0 = instruction signal
5	R/W	R/W read/write from /to LCD 1 = read mode, 0 = write mode
6	Enable	chip enable signal for data assumption
7	DB0	Data bit 0
8	DB1	Data bit 1
9	DB2	Data bit 2
10	DB3	Data bit 3
11	DB4	Data bit 4
12	DB5	Data bit 5
13	DB6	Data bit 6
14	DB7	Data bit 7
15	Backlight +	Backlight anode (VCC with 100Ω series resistor)
16	Backlight -	Backlight cathode

Table: LCD connector pinout and signal explanation of matrix display connector

4.3. General Adaption Hints

The display used should be equipped with a Hitachi HD44780 controller or a compatible one. This is a standard controller for parallel port interfaced DOT-MATRIX displays. For direct programming of these LCD-displays please refer to the controllers manual. With this information a software programmer can directly access the DOT-MATRIX display by himself. There are also some 3rd party driver programs available on the free market to support the HD44780 controller.

This direct programming may be a bit too complex for many applications. We have designed a special BIOS extension, which is integrated in the Kontron module system BIOS. This interface can be accessed through BIOS calls of the software interrupt INT 10hex. This interface can only be used with operating systems that allow you to access BIOS interrupts (e.g. DOS). Other operating systems like protected mode or real time OS cannot use this feature. A customer using these higher level OS's can only do direct programming through the LPT-port I/O-addresses.

5. BIOS AND SOFTWARE INFORMATION

As the extension BIOS of the PC/104 board needs to communicate with the connected character LCD in a bi-directional way, the LPT port must be set to bi-directional mode. This can be done in the BIOS setup. Currently two different BIOS sources are in use, one of AMI (MOPSplus) and one of Phoenix (all others). See the BIOS configuration for the use under DOS below.

5.1. AMI BIOS Configuration

(1) Boot up your system having a DOS in the boot device.

(2) Press the <DEL> key when you are asked to enter the BIOS setup.

(3) Choose the PERIPHERAL SETUP.

(4) Change PARALLEL PORT MODE to “EXTENDED” (the extended mode offers a bidirectional communication of the LPT-port, which is required to send data in both directions to and from the character LCD).

(5) Leave the BIOS setup by saving the new configuration.

(6) Now the system will come up to a DOS prompt.

(7) You now require the special configuration utility, ALCDINIT.EXE. This file is available at from the Kontron web pages.

(8) Start the configuration utility ALCDINIT.EXE with the suitable parameters.

5.2. Phoenix BIOS Configuration

(1) Boot up your system having a DOS in the boot device.

(2) Press the <F2> key when you are asked to enter the BIOS setup.

(3) Choose the ADVANCED menu.

(4) Select the I/O DEVICE CONFIGURATION entry.

(5) Set PARALLEL PORT to ENABLED.

(6) Set MODE to BI-DIRECTIONAL

(7) Leave the BIOS setup by saving the new configuration.

(8) Now the system will come up to a DOS prompt.

(9) You now require the special configuration utility, ALCDINIT.EXE. (Kontron web pages).

(10) Start the configuration utility ALCDINIT.EXE with the suitable parameters.

5.3. The Configuration Utility

The configuration utility “alcdinit.exe” is required to tell the BIOS extension about the mode and the type of the DOT-MATRIX display used.
The user can configure the matrix interface with “alcdinit.exe” utility program:

alcdinit <mode> <columns> <lin> <la1> <la2> <la3> <la4>

mode **= ON** the display interface will be prepared for direct mode
= COPY the display interface will be prepared for copy mode

lin number of LCD lines (normally 2 or 4)
columns number of LCD columns (normally 16, 20 or 40)

la? starting address of LCD lines; refer to the datasheet for your display

Example 1: **ALCDINIT ON 20 4 00 40 14 54**

Configures a DOT-MATRIX that has 4 lines and 20 columns, the line start address is 00hex for line 1, 40hex for line 2, 14hex for line 3 and 54hex for line 4.

The line addresses for each display may be different, therefore this information should be looked up in the displays datasheet. However the common configuration for:
- ***2 line*** displays are 00hex 40hex 00hex 40hex
- ***4 line*** displays are 00hex 40hex 14 hex 54hex.

If this configuration does not work, contact the display supplier for details.

5.3.1. The COPY mode (copy to display)

The COPY mode is a good mode for testing the general function of the DOT-MATRIX display. This mode directs all text mode outputs going to the standard CRT-monitor also to the DOT-MATRIX display. Of course the output is limited to the size of the character-LCD. On a display with 2 lines and 16 characters the same output will appear that can be seen on the CRT-monitor in the first two lines up to character number 16.

A DOT-MATRIX LCD has no video memory, therefore scrolling in the CRT output will not be supported. Please note that a DOS operating system sometimes does not use line 1 on the CRT-monitor therefore the first output line is number 2. If this is the case then line 1 from the DOS-side may not show any characters on the DOT-MATRIX display.

These functions are standard INT 10hex software interrupt calls. A programmer can easily integrate these functions in his application by presetting the input registers with desired values and after this generating a software interrupt 10hex. The tables on the pages below show all the information about input register settings and output register results.

Select Video Mode	Int 10h	
Input:	AH=00h	AL=Desired Mode
Output:	-	
Result on Screen:	The desired video mode is selected. If AL.7=0 the screen is being cleared.	
Result on Display:	If AL.7=0 the display is being cleared and cursor is set to home position. If AL.7=1 the cursor is set to home position.	

Set Cursor Type	Int 10h	

Input:	AH=01h	CH=Cursor Start Line
		CL=Cursor End Line

Output:	-	
Result on Screen:	Defines the start line and end line of the textmode cursor. CX=0607h selects standard cursor. CX=20xxh disables cursor.	
Result on Display:	CX=20xxh disables the cursor CX=40xxh enables blinking character cursor Any other value in CX enables underlined cursor.	

Set Cursor Position	Int 10h	

Input:	AH=02h
	BH=Video Page #
		DH=Cursor Line
		DL=Cursor Column

Output:	-	
Result on Screen:	Positions the cursor on one of the available video pages. Valid lines are 0-24, valid columns are 0-79	
Result on Display:	Positions the cursor on the display if BH=0. Valid Lines are 0-3, valid columns are 0-19. If BH≠0 or line or column are invalid, the cursor position will not be changed.	

Write Character (& Attribute)	Int 10h		

Input:	AH=09h or AH=0Ah
	AL=ASCII Code
		BH=Video Page #
		BL=Attribute or Color
			CX=# of Repetitions

Output:	-		
Result on Screen:	Writes character(s) from AL to the screen. Cursor position is not changed.		
Result on Display:	Send character(s) if BH=0. CX is ignored. Cursor position is not changed.		

Write Character	Int 10h	

Input:	AH=0Eh
	AL=ASCII Code
		BH=Video Page #
		BL=Color

Output:	-	
Result on Screen:	Writes character from AL to the screen. Cursor position is incremented. Beep, Backspace, LineFeed and CR are interpreted correctly.	
Result on Display:	Writes character from AL to the display. Cursor position is incremented. Beep, Backspace, LineFeed and CR are interpreted correctly.	

Set Mode Flags	Int 10h	
Input:	AH=8Bh	AL bit 0 = 0 = Int 10h LCD functions disabled AL bit 0 = 1 = Int 10h LCD functions enabled AL bit 1 = 0 = copy mode off AL bit 1 = 1 = copy mode on AL bit 2 = 0 = cursor auto increment on AL bit 2 = 1 = cursor auto increment off AL bit 3-7 = must be zero

	BX = must be zero
	CX = must be zero
	DX = must be zero	

Output:	-	
Description:	This function allows setting various mode flags of the alphanumeric LCD BIOS extension. Int 10h LCD functions enable/disable: When disabled all CopyToDisplay functions and DirectToDisplay functions except the SetModeFlags function are ignored. Copy mode on/off: Used to enable/disable the CopyToDisplay functions. Cursor auto increment on/off: This affects the DirectToDisplay functions WriteCharacter (8Eh) and WriteString (8Fh). When switched off the cursor will not be incremented after the last character displayed. This is useful to avoid scrolling when writing into the lower write corner of the display.	

5.3.2. The ON mode (direct to display)

The ON mode is called “Direct to Display” and offers other functions implemented in the extension BIOS. Every command given by a “Direct to Display” function is immediately transferred to the LC character display and does not affect the screen of a graphic card in the system. These functions must have bit 7 set in register AH when called.

These functions listed on the following pages are suitable for programmers who want to directly access the character LCD with their special outputs. They are also based on the presetting of registers and simple BIOS INT10hex calls.

Clear Screen	Int 10h		

Input:	AH=80h	AL=0xxxxxxxb or
		AL=1xxxxxxxb
			AL.7=0 ClearScreen
			AL.7=1 CursorHome

Output:	-		
Result on Display:	If AL.7=0 the display is being cleared and cursor is set to home position. If AL.7=1 the cursor is set to home position.		

Set Cursor Type	Int 10h	
Input:	AH=81h	CH=Cursor Mode
Output:	-	
Result on Display:	CH=00h enables underline Cursor CH=20h disables the cursor CH=40h enables blinking character cursor	

Set Cursor Position	Int 10h	

Input:	AH=82h	DH=Cursor Line
		DL=Cursor Column

Output:	-	
Result on Display:	Positions the cursor on the display. Valid Lines are 0-3, valid columns are 0-19. If line or column is invalid, the cursor position will not be changed.	

Get Cursor Position	Int 10h	
Input:	AH=83h	
Output:	DH = Cursor Line	DL = Cursor Column
Result on Display:	-	

Enable/Disable Display	Int 10h	

Input:	AH=84h	AL=0 Disable Display
		AL=1 Enable Display

Output:	-	
Result on Display:	Display is enabled or disabled. The display content is not affected by disabling the display. Display content may be altered while display is off. If ”copy to display” mode is selected in setup, display is enabled by default. Otherwise display is disabled by default.	

Define User Character	Int 10h	

Input:	AH=85h
	AL=Character #
		ES:DI=Pointer to 8 Bytes of definition data

Output:	-	
Result:	Defines one of the 8 user programmable characters. AL is the number of the character to define (valid values are 0-7). ES:DI points to an array of 8 bytes that define the pixels of the new character. To display one of the characters use the function ”Write Character” with ASCI-values 0-7.	

Write Character	Int 10h

Input:	AH=8Eh
	AL=ASCII Code

Output:	-
Result on Display:	Writes character from AL to the screen. Cursor position is incremented. Beep, Backspace, LineFeed and CR are interpreted correctly.

Write String	Int 10h	
Input:	AH=8Fh	ES:DI = Pointer to null-terminated string
Output:	-	
Result on Display:	Writes characters from the null-terminated string pointed to by ES:DI to the display. Cursor position is incremented. Beep, Backspace, LineFeed and CR are interpreted correctly.	

Send Raw Character	Int 10h	(will be implemented with BIOS R114)
Input:	AH=90h	AL=Character to send
Output:	-	
Result on Display:	Sends the raw data from AL directly to the display. Cursor position is not changed. No control codes are interpreted.	

Send Raw Control	Int 10h	(will be implemented with BIOS R114)
Input:	AH=91h	AL=Control Code to send
Output:	-	
Result on Display:	Sends the raw data from AL directly to the display’s control register. Cursor position is not changed. No codes are interpreted.	