

DATA SHEET

LCD6402-DEV Development/Evaluation Board for the LCD6402B

LCD6402-DEV Development/Evaluation Board

INTRODUCTION

The LCD6402-DEV is a chassis board (see Figure 1 below) for evaluating the LCD6402B 128x64 intelligent monochrome graphic LCD and can also be used for developing simple host applications*

* Requires the addition of a suitable compiler and ICD.

FEATURES

- Console program
- RS232 port
- PIC16F876A host
- Port expander
- ICD/ICSP port
- LCD keypad expander

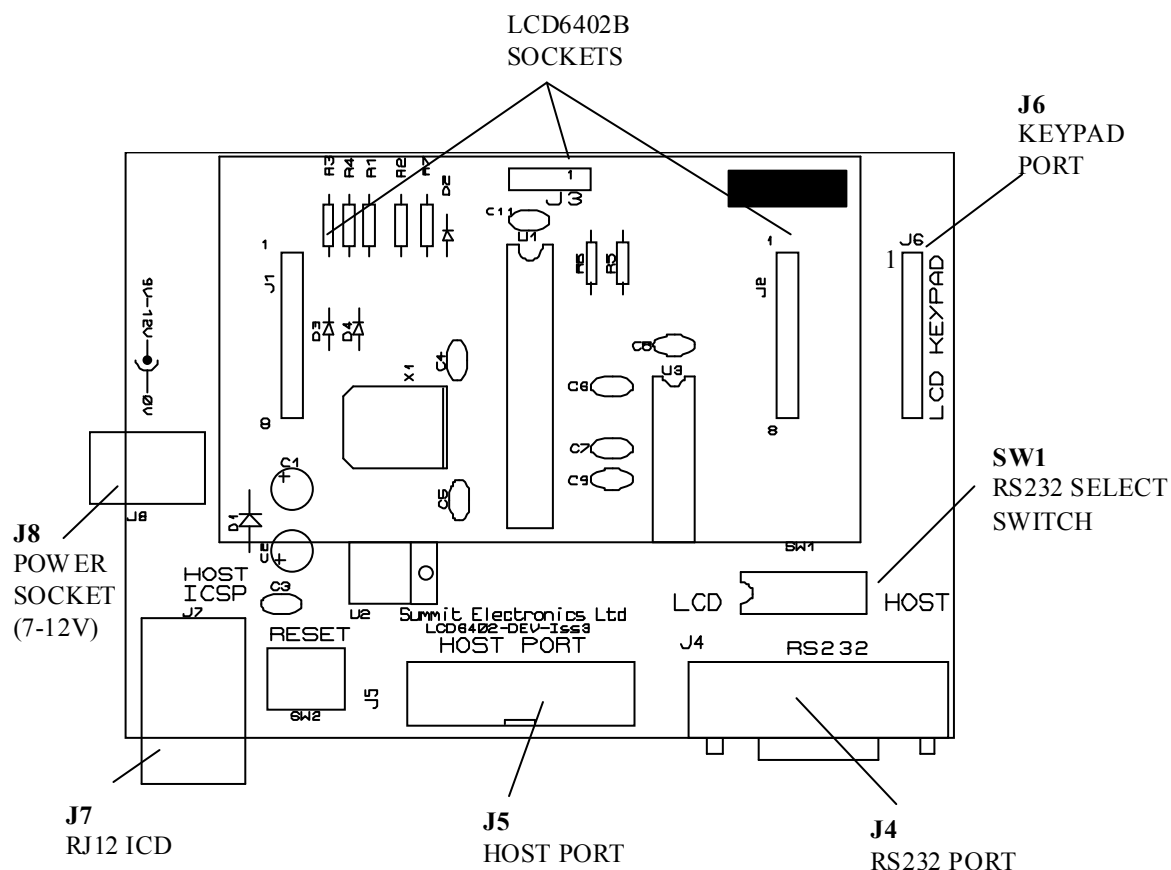


Figure 1 LCD6402-DEV CONNECTOR LOCATIONS

CONTENTS

1	GENERAL DESCRIPTION	4
2	EVALUATION	5
2.1	CONSOLE SET-UP.....	5
2.2	CONSOLE COMMAND SUMMARY.....	6
2.3	CONSOLE COMMAND DESCRIPTION.....	6
2.3.1	BL: CONTROL BACKLIGHT	6
2.3.2	CC: DRAW CIRCLE	7
2.3.3	CF: DRAW FILLED CIRCLE	7
2.3.4	CS: CLEAR SCREEN	7
2.3.5	CU: MOVE CURSOR	7
2.3.6	FG: SET FOREGROUND COLOUR	8
2.3.7	BG: SET BACKGROUND COLOUR	8
2.3.8	MD: SET MODE	8
2.3.9	RT: DRAW RECTANGLE	9
2.3.10	RF: DRAW FILLED RECTANGLE	9
2.3.11	FT: SET FONT	9
2.3.12	LT: DRAW LINE	9
2.3.13	OR: SET ORIGIN	10
2.3.14	PS: PRINT STRING	10
2.3.15	PX: SET PIXEL	10
2.3.16	RC: RECALL GRAPHIC OBJECT	10
2.3.17	RS: RESET DISPLAY	11
2.3.18	HR: PRINT HORIZONTAL RASTER	11
2.3.19	VR: PRINT VERTICAL RASTER	11
2.3.20	\$W: WRITE LCD64XX REGISTER	11
2.3.21	\$R: READ LCD64XX REGISTER	12
3	DEVELOPMENT	13
3.1	PROJECT EXAMPLES	13
3.2	PROJECT EXAMPLE #1: EX_CHECKBOX1	13
3.3	PROJECT EXAMPLE #2: EX_RADIOGROUP1	14
3.4	PROJECT EXAMPLE #3: EX_LISTMENU1	14
3.5	PROJECT EXAMPLE #4: EX_TIME.....	15
3.6	PROJECT EXAMPLE #5: EX_CENTRALHEATING	15
4	CONNECTOR DETAILS	17
4.1	J4: RS232 PORT	17
4.2	J5: HOST PORT	17
4.3	J6: KEYPAD PORT	18
4.4	J7: ICD / ICSP PORT.....	18
4.5	J8: POWER.....	18
5	CIRCUIT SCHEMATIC	19
6	REFERENCES	20
7	COMMENT FORM	21

1 GENERAL DESCRIPTION

The LCD6402-DEV is used to evaluate the LCD6402B intelligent graphic LCD and can also be used to develop simple host applications.

When used as an evaluation tool the LCD6402-DEV is simply connected to a PC running a terminal program such as Hyper Terminal to enter various commands.

With the addition of a suitable compiler and ICD the LCD6402-DEV can also be used to develop simple host applications making use of the ICD and Host ports.

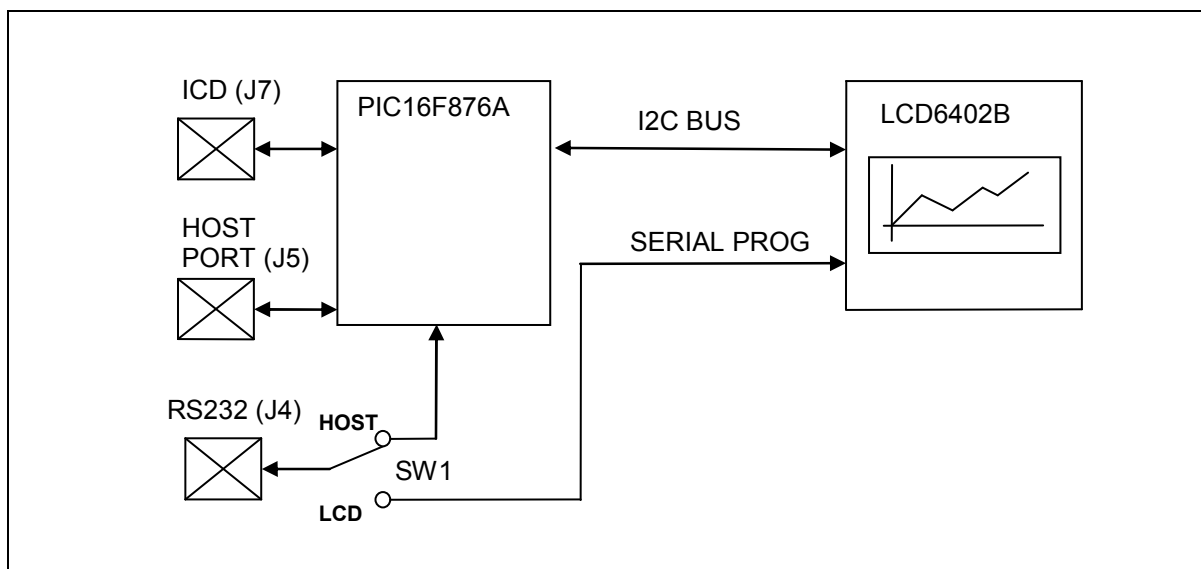


Figure 2 LCD6402-DEV block schematic

2 EVALUATION

The LCD6402-DEV comes ready programmed with a console program for evaluating the LCD6402B. All that is required is to connect a serial lead to a PC running a terminal emulator (e.g. 'Hyper Terminal') and typing in a command such as 'PS "Hello World"'. The console program receives the serial commands from the terminal and converts them into the appropriate instructions before sending them over the I2C bus to the LCD6402B. The source code for the console program is also provided as an example interface to the LCD6402B.

2.1 CONSOLE SET-UP

See Figure 3 for console set-up and carry out the following:

- i) Set PC serial port to:
 - Baud rate : 9600
 - Data bits : 8
 - Stop bits : 1
 - Flow control : None
- ii) Connect serial extension cable between PC and LCD6402-DEV
- iii) Switch SW1 to 'Host'
- iv) Connect power

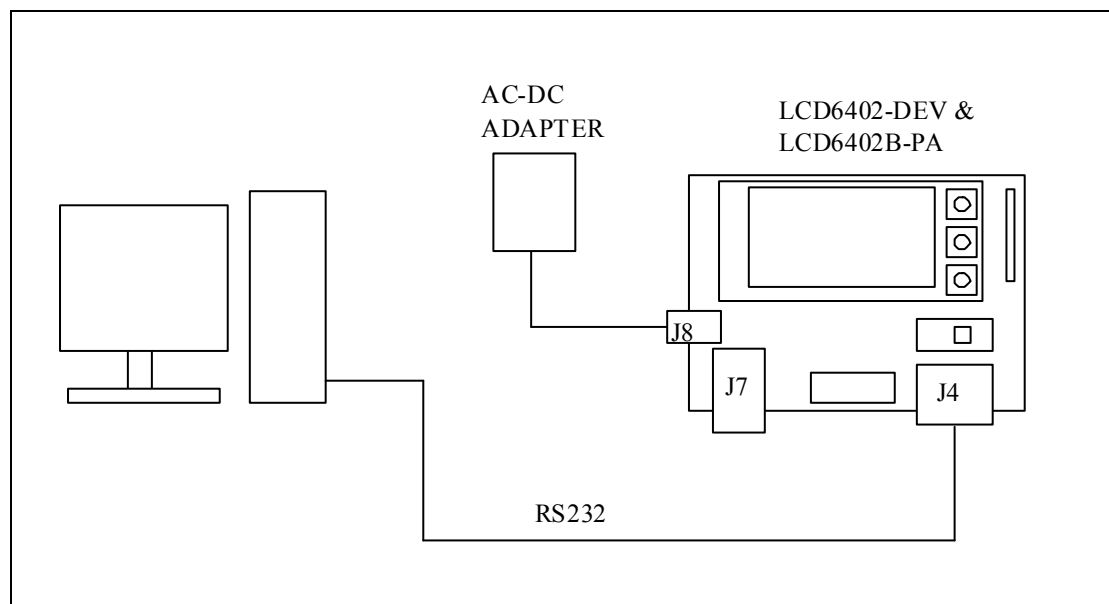


Figure 3 Console Set-up

2.2 CONSOLE COMMAND SUMMARY

Command & parameters	Operation
BL n	Control backlight n=1: On, n=0: Off
CC r	Draw circle with radius r
CF r	Draw filled circle
CS	Clear screen
CU x y	Move cursor to x y
FG n	Set foreground colour
BG n	Set background colour
RT x0 y0 x1 y1	Draw rectangle with corners x0 y0 and x1 y1
RF x0 y0 x1 y1	Draw filled rectangle with corners x0 y0 and x1 y1
FT f	Select font f
LT x y	Draw line to x y
OR x y	Set display origin to x y
PS "c..."	Print string
PX x y n	Set pixel at x y to background colour(n=0) or foreground colour (n=1)
RC n	Recall object by index
RS	Reset display
HR n	Print horizontal raster byte n
VR n	Print vertical raster byte n
\$W r n	Write value n to LCD64XX register r
\$R r	Read and print contents of LCD64XX register r

2.3 CONSOLE COMMAND DESCRIPTION

All commands are terminated with a carriage return. Parameters may be entered in C-style hexadecimal notation (e.g 0xFF instead of 255).

2.3.1 BL: Control backlight

Syntax: BL n

Parameters: n backlight state

Description: Control backlight. $n=0$ switches backlight off, $n=1$ switch backlight on

Example: Switch backlight on:
'>BL 1'

2.3.2 CC: Draw circle

Syntax: CC r

Parameters: r radius

Description: Draw circle with centre at current cursor position with radius r

Example: Draw circle with radius 10 at cursor position 20 24:
'>CU 20 24'
'>CC 10'

2.3.3 CF: Draw filled circle

Syntax: CF r

Parameters: r radius

Description: Draw filled circle with centre at current cursor position with radius r

Example: Draw filled circle with radius 10 at cursor position 20 24:
'>CU 32 32'
'>CF 30'

2.3.4 CS: Clear screen

Syntax: CS

Parameters: None

Description: Clear screen and reset cursor and origin $x y$ to zero

Example: Clear screen
'>CS'

2.3.5 CU: Move cursor

Syntax: CU $x y$

Parameters: x cursor position
 y cursor position

Description: Move cursor to position $x y$

Example: Move cursor to 63 31:
 '>CU 63 31'

2.3.6 FG: Set foreground colour

Syntax: CL *clr*

Parameters: **Clr**
 0 = Black, 255 = White

Description: Set foreground colour

Example Set foreground colour to black
 '>FG 0'

2.3.7 BG: Set background colour

Syntax: CL *clr*

Parameters: **Clr**
 0 = Black, 255 = White

Description: Set background colour

Example Set background colour to white
 '>BG 255'

2.3.8 MD: Set mode

Syntax: MD *n*

Parameters: **clr** colour flags
 Bit 0:
 0 OR foreground colour
 1 XOR foreground colour
 Bit 1:
 0 Enable foreground colour
 1 Disable foreground colour
 Bit 4:
 0 OR background colour
 1 XOR background colour
 Bit 5:
 0 Enable background colour
 1 Disable background colour

Description: Set colour mode

Example Set colour mode XOR foreground with background disabled
#1:

Example #2: '>CL 0x21'
Set colour to OR foreground and OR background:
#2: '>CL 0'

2.3.9 RT: Draw rectangle

Syntax: RT x0 y0 x1 y1
Parameters: **x0 y0** Bottom left corner co-ordinate
 x1 y1 Top right corner co-ordinate
Description: Draw rectangle
Example: Draw frame with bottom left corner 10 10 and top right corner
 20 20:
 '>RT 10 10 20 20'

2.3.10 RF: Draw filled rectangle

Syntax: RF x0 y0 x1 y1
Parameters: **x0 y0** Bottom left corner co-ordinate
 x1 y1 Top right corner co-ordinate
Description: Draw filled rectangle
Example: Draw filled rectangle with bottom left corner 10 10 and top right
 corner 20 20:
 '>RF 10 10 20 20'

2.3.11 FT: Set font

Syntax: FT f
Parameters: **f** font type
Description: Set font type:
 0 3x5
 1 5x8
 2 5x16
 8-15 Custom fonts*
 **Custom fonts are programmed to the LCD6402 using LCDLAB*
Example: Select font type 2:
 '>FT 2'

2.3.12 LT: Draw line

Syntax: LT x y

Parameters: **x y** destination co-ordinate

Description: Draw line from current cursor position to x y

Example: Draw line from 10 10 to 20 20:

'>CU 10 10'

'>LT 20 20'

2.3.13 OR: Set origin

Syntax: OR x y

Parameters: **x y** co-ordinate of new origin

Description: Sets the display origin to x y.

Example: Move display origin to 32 32 and draw all entities relative to new origin.

'>OR 32 32'

2.3.14 PS: Print string

Syntax: PS "c..."

Parameters: **c...** Text string

Description: Print text string from current cursor position

Example: Print 'Hello World' at 0 0 using font 2

'>CU 0 0'

'>FT 2'

'>PS "Hello World"'

2.3.15 PX: Set pixel

Syntax: PX x y n

Parameters: **x y** co-ordinate

n Foreground(n=1) / Background(n=0)

Description: Set pixel at co-ordinate x y to the current foreground colour

Example: Set pixel at 22 3

'>PX 22 3 1'

2.3.16 RC: Recall graphic object

Syntax: RC n

Parameters: **n** graphic object number in the range 0-127

Description: Recall graphics object* from non-volatile memory. If specified object is not present, no operating is carried out.

**Graphic objects are programmed to the LCD6402 using LCDLAB*

Example: Recall object 0:
'>RC 0'

2.3.17 RS: Reset display

Syntax: RS
Parameters: None
Description: Resets the display
Example: Reset:
'>RS'

2.3.18 HR: Print horizontal raster

Syntax: HR n
Parameters: *n* Raster byte
Description: Print horizontal raster at graphics cursor
Example: Print horizontal dot dash raster
'>HR 0x55'

2.3.19 VR: Print vertical raster

Syntax: VR n
Parameters: *n* Raster byte
Description: Print vertical raster at graphics cursor
Example: Print horizontal dot dash raster
'>VR 0x55'

2.3.20 \$W: Write LCD64XX register

Syntax: \$W r n
Parameters: *r* LCD6402B register
n Value
Description: Write value n to LCD6402B register
Refer to data sheet LCD6402B for more details on registers.
Example: Switch backlight on. Write command 0x92 to register
CMD(0x20)
'>\$W 0x20 0x92'

2.3.21 \$R: Read LCD64XX register

Syntax: \$R

Parameters: *r* LCD6402B register

Description: Read and print LCD6402B register *r* to console.

 Refer to data sheet LCD6402B for more details on registers.

Example: Read keypad KEY(0x23) register.

 '>\$R 0x23H'

3 DEVELOPMENT

Simple host applications can be developed on the LCD6402-DEV with the addition of a suitable compiler and ICD. The host is based on a Microchip PIC16F876A and can be programmed via the popular RJ12 ICD / ICSP port J7.

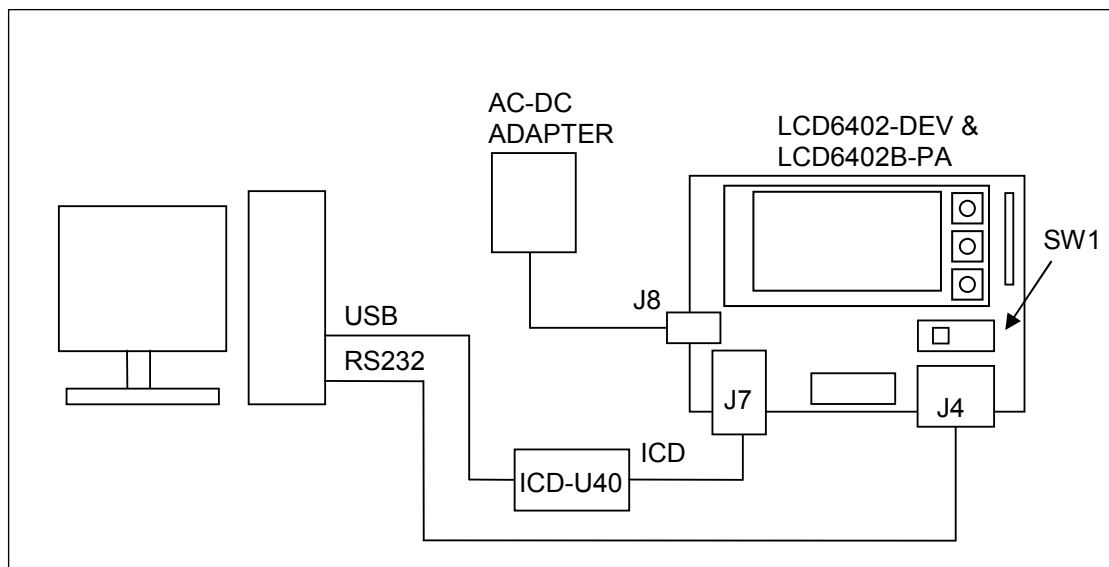


Figure 4 Example development set-up

3.1 PROJECT EXAMPLES

Below are some example projects that were developed on the LCD6402-DEV with the addition of a CCS 'C' compiler and ICD. Assuming LCDLAB has been installed, these projects can be found in the 'project\LCD6402-DEV' folder.

3.2 PROJECT EXAMPLE #1: ex_checkbox1

This project demonstrates a series of three checkbox sets operated by user keys: A, B & C.

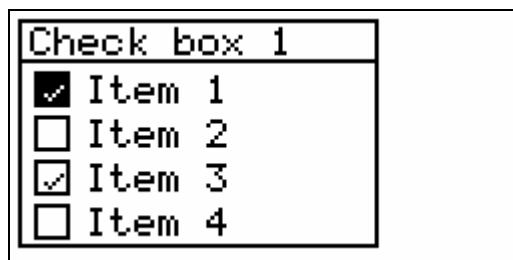


Figure 5 Checkbox Set 1

The checkbox graphics are created using LCDLAB and programmed to the LCD6402 via the LCD6402-DEV. Figure 5 shows the first set of checkboxes. In this example the host application moves an inverting cursor and draws a tick icon on the appropriate box according to the user key input.

See LCDLAB project: 'ex_checkbox1\LCDLAB\ex_checkbox1.lpr'

See CCS project: 'ex_checkbox1\HostApp\ex_checkbox1.prj'

3.3 PROJECT EXAMPLE #2: ex_radiogroup1

This project demonstrates a series of three radio groups operated by user keys: A, B & C.

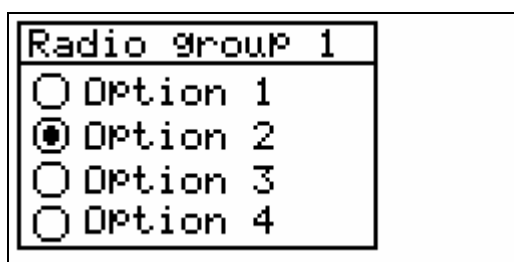


Figure 6 Radio Group 1

The radio group graphics are created using LCDLAB and programmed to the LCD6402 via the LCD6402-DEV. Figure 6 shows the first radio group. In this example, the host application draws a dot icon on the appropriate radio button according to the user key input. The item titles are easily modified using LCDLAB.

See LCDLAB project: 'ex_radiogroup1\LCDLAB\ex_radiogroup1.lpr' and

See CCS project: 'ex_radiogroup1\HostApp\ex_radiogroup1.prj'

3.4 PROJECT EXAMPLE #3: ex_listmenu1

This project demonstrates a series of three menus operated by user keys: A, B & C.

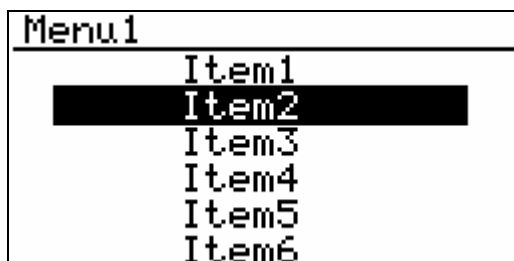


Figure 7 List Menu

The list menu graphics are created using LCDLAB and programmed to the LCD6402 via the LCD6402-DEV. Figure 7 shows the first list menu. In this example, the host application draws and moves an inverting cursor over the

appropriate item according to the user key input. The item titles are easily modified using LCDLAB.

See LCDLAB project: 'ex_listmenu1\LCDLAB\ex_listmenu1.lpr'

See CCS project: 'ex_listmenu1\HostApp\ex_listmenu1.prj'

3.5 PROJECT EXAMPLE #4: ex_time

This project demonstrates a series of menus as in project example #3 and in addition autonomously prints time in a custom font.

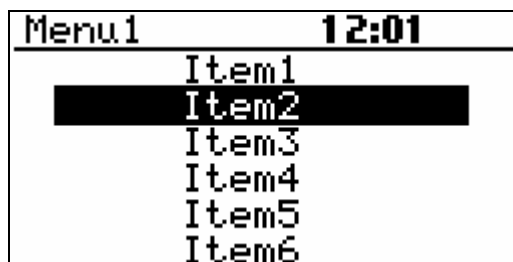


Figure 8 Time Printing

The custom font was added using LCDLAB's Project Manager (opened from 'View | Project manager...').

In this case 'MobilProp8.fnt' has been added to the 'Font' tab. See also DoEvents() in the application source ex_time.c.

See LCDLAB project: 'ex_time\LCDLAB\ex_time.lpr'

See CCS project: 'ex_time\HostApp\ex_time.prj'

3.6 PROJECT EXAMPLE #5: ex_centralheating

This project demonstrates a simple central heating controller using ATM style menus and introduces the use of animated graphics using bitmaps created in Windows paint. Figure 9 and Figure 10 below show the non-rendered main display and a selection of the menus designed using LCDLAB.

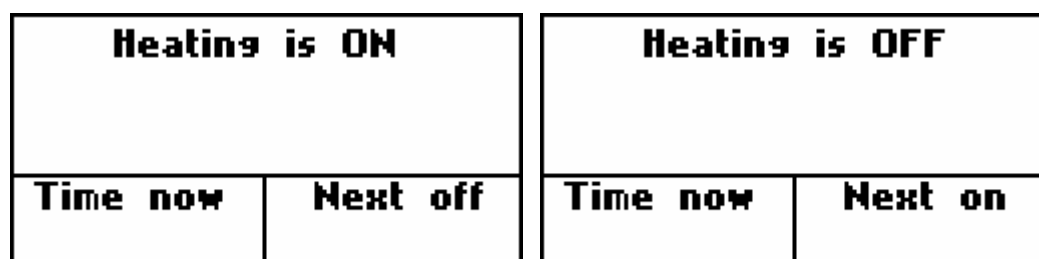


Figure 9 Main Heating Displays

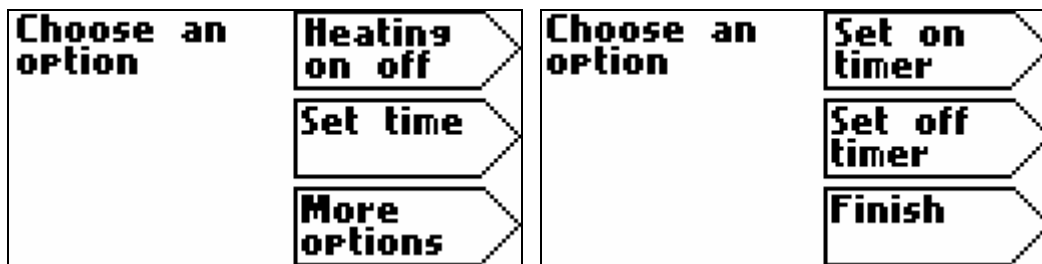


Figure 10 Central Heating Menu

Simple animation is possible by displaying two or more graphics in sequence. Figure 11 shows two icons depicting heat rising from a radiator, are alternately displayed as an animation, to show that the heating is on.

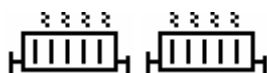


Figure 11 Animated Graphics

See LCDLAB project: 'ex_centralheating\LCDLAB\ex_centralheating.lpr'

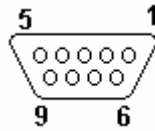
See CCS project: 'ex_centralheating\HostApp\ex_centralheating.prj'

4 CONNECTOR DETAILS

This section describes the connector pin out details of the LCD6402-DEV.

4.1 J4: RS232 PORT

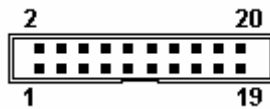
Connector type: 9-way 'D' socket.



Pin	Description
2	TD: Transmit data out
3	RD: Receive data in
5	GND: Common ground

4.2 J5: HOST PORT

Connector type: 20-way IDC box.



Pin	Description
1	RA0
2	RA1
3	RA2
4	RA3
5	RA4
6	RA5
7	RB0
8	RB1
9	RB2
10	RB3

Pin	Description
11	RB4
12	RB5
13	RB6 ¹
14	RB7 ¹
15	RC0
16	RC1
17	RC2
18	MCLR
19	+5V out
20	0V

1) Shared with J7 ICD / ICSP

4.3 J6: KEYPAD PORT

Connector type: 8-way pin header.

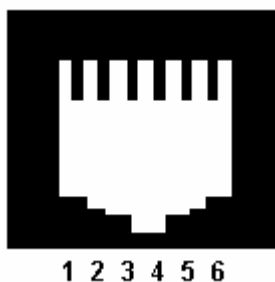


Pin	Description
1	R0: Row select 0
2	R1: Row select 1
3	R2: Row select 2
4	R3: Row select 3

Pin	Description
5	C0: Column output 0
6	C1: Column output 1
7	C2: Column output 2
8	C3: Column output 3

4.4 J7: ICD / ICSP PORT

Connector type: 6-way RJ12.



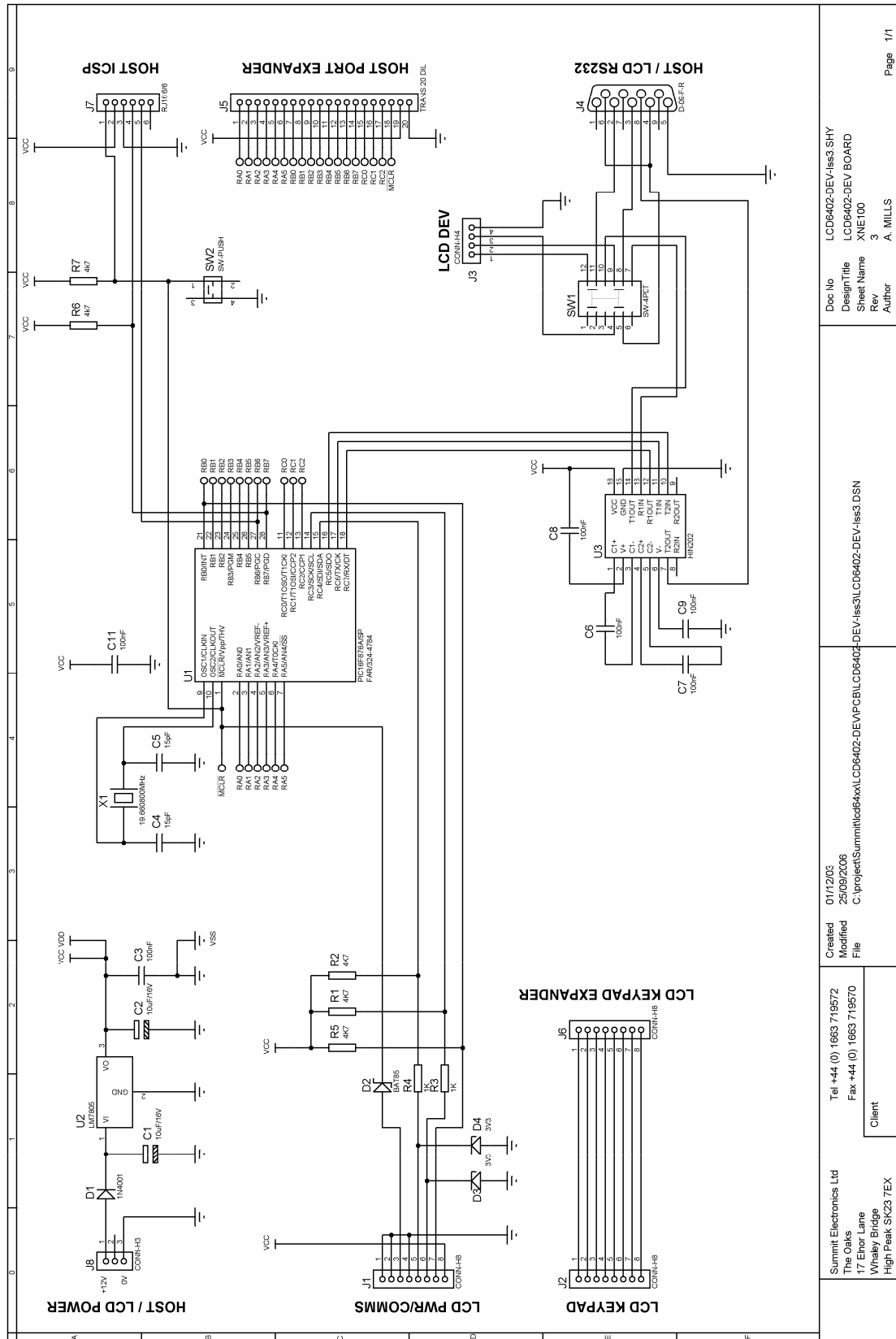
Pin	Description
1	MCLR
2	+5V
3	Ground
4	RB7: (ICSP data) on target PIC
5	RB6: (ICSP clock) on target PIC
6	NC

4.5 J8: POWER

Connector type: 2-way DC power connector 2.1mm



5 CIRCUIT SCHEMATIC



Doc No	LCD6402-DEV-Iss3.SHY
Design Title	LCD6402-DEV BOARD
Sheet Name	XNE100
Rev	3
Author	A. MILLS
Page	1/1

Created	01/12/05
Modified	25/09/2006
File	C:\project\Summit\lcd6402-DEV\PCB\LCD6402-DEV-Iss3\LCD6402-DEV-Iss3.DSN

Summit Electronics Ltd	Tel +44 (0) 1663 719572
The Oaks	Fax +44 (0) 1663 719570
17 Elmor Lane	Client:
Walsley Bridge	
High Peak SK23 7EX	

6 REFERENCES

1. Custom Computer Services (CCS) C-Compiler, www.ccsinfo.com

LCD6402-DEV Development/Evaluation Board for the LCD6402B

Disclaimer

The information in this document is offered in good faith as representing the characteristic of its product. Summit Electronics Ltd will not accept any responsibility whatsoever for any damage or loss sustained as a result of any inclusion, omission or error in documentation, verbal advice or other data. Use of Summit Electronics Ltd products in medical applications is not approved. Such use includes, but is not limited to life support systems.