

user's guide to

# SmartGLCD

## 240x128

Smart GLCD 240x128  
represents development tool  
and final product specially  
designed for graphical  
applications which use PIC  
microcontroller PIC18F8722



# TO OUR VALUED CUSTOMERS

I want to express my thanks to you for being interested in our products and for having confidence in Mikroelektronika.

The primary aim of our company is to design and produce high quality electronic products and to constantly improve the performance thereof in order to better suit your needs.

A handwritten signature in white ink, appearing to read 'N. Matic', is positioned in the lower right quadrant of the page.

Nebojsa Matic  
General Manager

# Table of Contents

Introduction to SmartGLCD 240x128	4	step 4 - Select .hex file	13
Package contains	5	step 5 - Uploading .hex file	13
Key Features	6	step 6 - Progress bar	14
System Specification	7	step 7 - Reset MCU	14
1. Connecting power supply	8	Tips and Tricks: Speed-up UART data transfer	15
2. PIC18F8722 Microcontroller	9	4. Programing with external programmer	16
Key microcontroller features	9	5. Touch Screen	18
3. Programming with bootloader	10	6. microSD Card Slot	20
Identifying device COM port	11	7. USB UART	22
step 1 - Choosing COM port	11	8. Pinout	24
step 2 - Connecting with a PC	12	Dimensions	25
step 3 - Browse for .hex file	12		

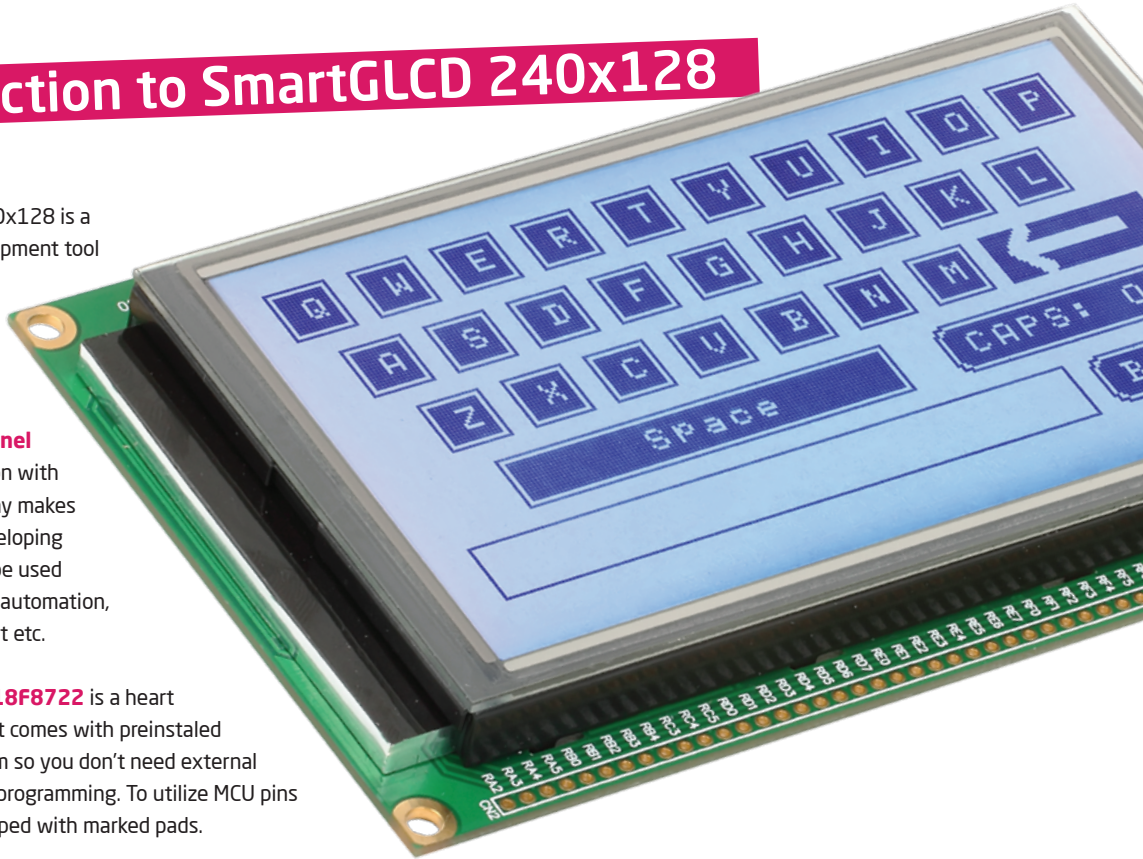
# Introduction to SmartGLCD 240x128

The SmartGLCD 240x128 is a full-featured development tool for experimenting with **Graphical LCD** display.

Main advantage of SmartGLCD is **resistive Touch Panel**

which in combination with Graphical LCD display makes perfect tool for developing devices which can be used as control for home automation, industrial equipment etc.

Microcontroller **PIC18F8722** is a heart of the SmartGLCD. It comes with preinstaled **bootloader** program so you don't need external programmer for MCU programming. To utilize MCU pins SmartGLCD is equipped with marked pads.



# Package contains



- 01 Damage resistant protective box



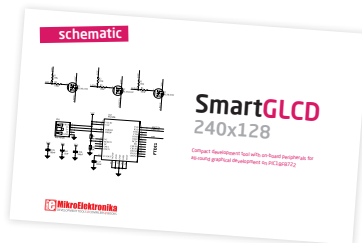
- 02 SmartGLCD 240x128 development tool



- 03 CD with documentation and examples



- 05 SmartGLCD 240x128 user's guide



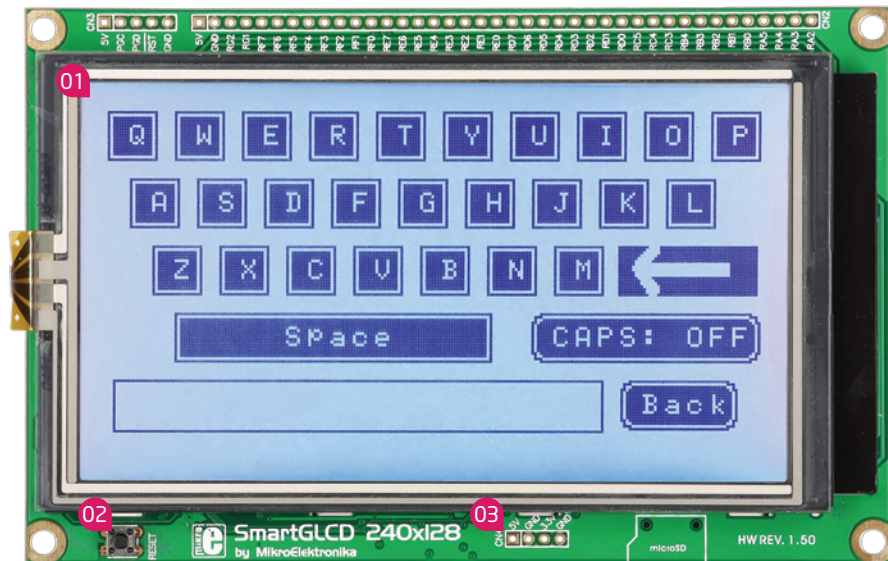
- 05 SmartGLCD 240x128 schematic



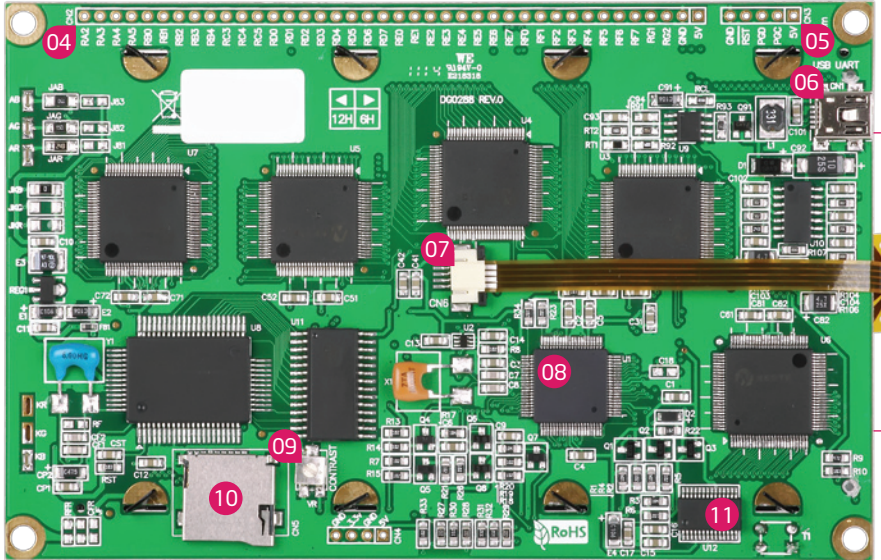
- 06 USB cable

# Key Features

- 01 GLCD 240x128 display
- 02 RESET button
- 03 Power supply pads
- 04 I/O pads
- 05 Pads for mikroProg programmer
- 06 USB connector
- 07 Touch panel connector
- 08 Microcontroller PIC18F8722
- 09 Contrast potentiometer
- 10 microSD card slot
- 11 USB UART module



# System Specification



**power supply**  
Over a USB cable (5V DC)



**power consumption**  
~350mA in idle state  
(backlight is ON)

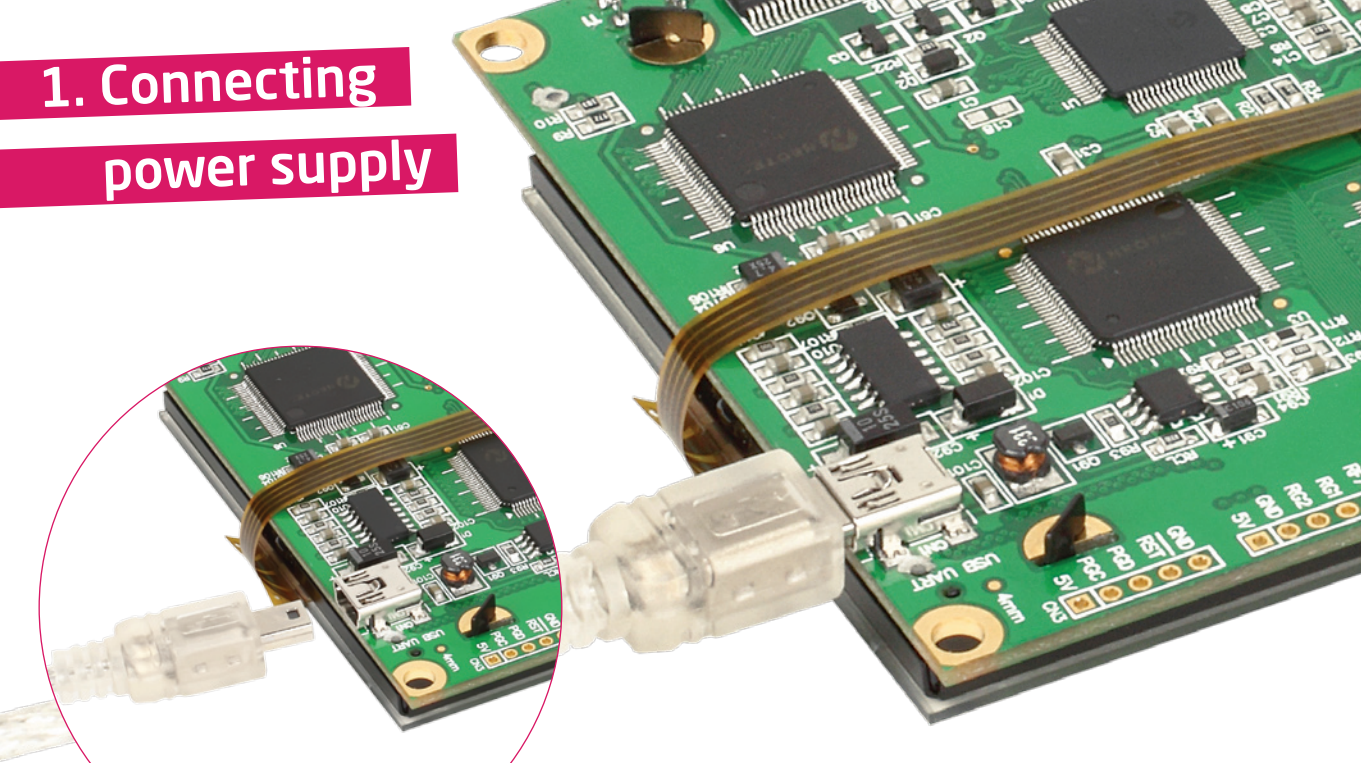


**board dimensions**  
140x90cm (5.51x3.24")



**weight**  
~200g (0.11 lbs)

# 1. Connecting power supply



**Figure 1-1: Powering the development system**

Connect the development system to a PC via a USB cable, Figure 1-1. The GLCD display will be automatically turned on.

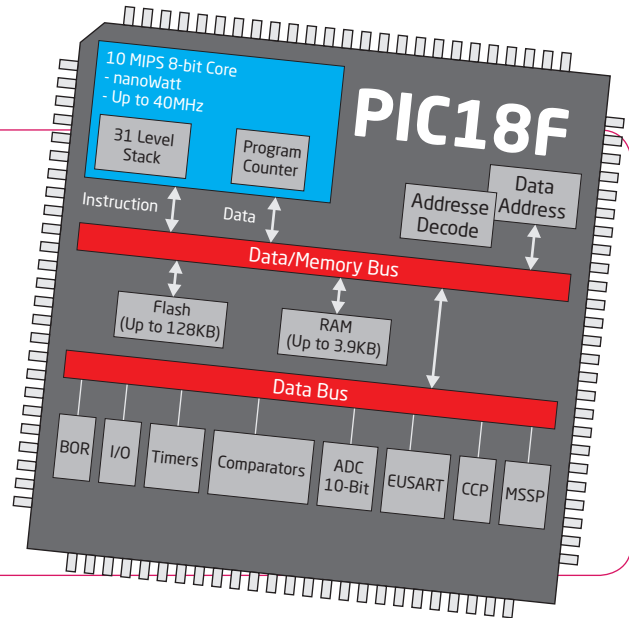


## 2. PIC18F8722 Microcontroller

The **SmartGLCD** development tool comes with the **PIC18F8722** microcontroller. This 8-bit microcontroller with its integrated modules and in combination with other on-board modules is ideal for creating graphical applications..

### Key microcontroller features

- Up to **10 MIPS** Operation;
- 8-bit architecture;
- 128KB of Flash memory;
- 3,936 bytes of RAM memory;
- 1024 bytes of EEPROM
- 80 pin TQFP;
- 16 ch, 10-bit ADC;
- UART, SPI, ; etc.



# 3. Programming with bootloader

For programming, microcontroller use bootloader program which is preinstalled in to MCU memory. To transfer .hex file from a PC to MCU you need bootloader software (**mikroBootloader**) which can be downloaded from:



<http://www.mikroe.com/eng/products/view/443/smartglcd-240x128-board/>

After software is downloaded unzip it to desired location and start mikroBootloader software.

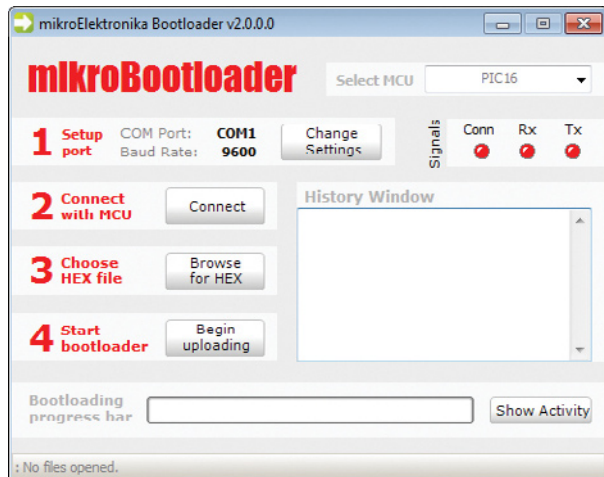
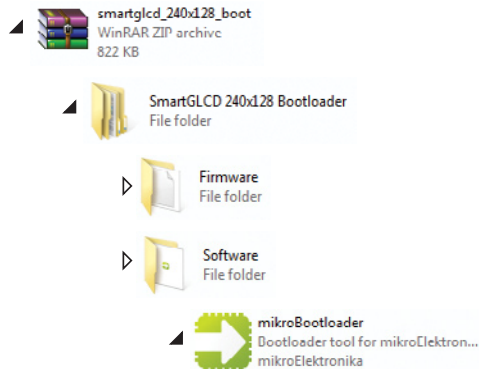


Figure 3-1: mikroBootloader software

**note**

*Connect SmartGLCD with a PC before starting mikroBootloader software*

## Identifying device COM port

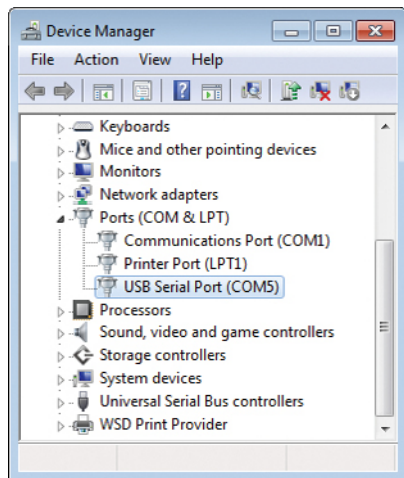


Figure 3-2: Identifying COM port

### note

*In Device Manager you can see which COM port is assigned to mikromedia (in this case COM5)*

## step 1 - Choosing COM port



Figure 3-3: Selecting COM port

- 01 Click on Change Settings button
- 02 Select USB COM port (in this case COM5)
- 03 Set Baud rate to 115200
- 04 Click OK button

## step 2 - Connecting with a PC

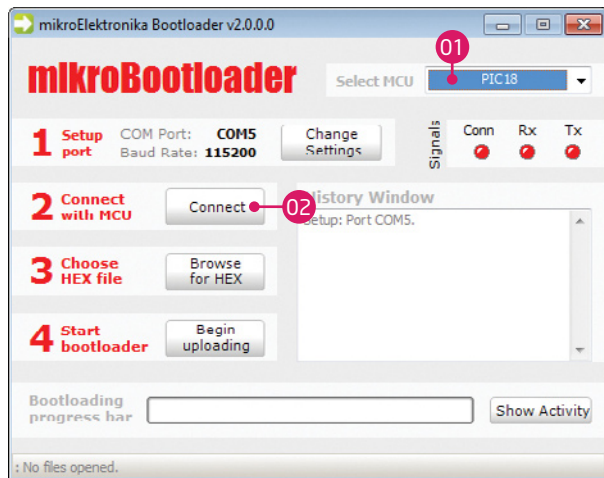


Figure 3-4: Connecting mikromedia with mikroBootloader

- 01 From drop down list Select MCU chose PIC18
- 02 Reset SmartGLCD and within 5s click on Connect button

## step 3 - Browse for .hex file



Figure 3-5: Browsing for .hex file

- 01 Click on Browse for HEX and from pop-up window (figure 3-6) select .hex file which will be uploaded to MCU memory

## step 4 - Select .hex file

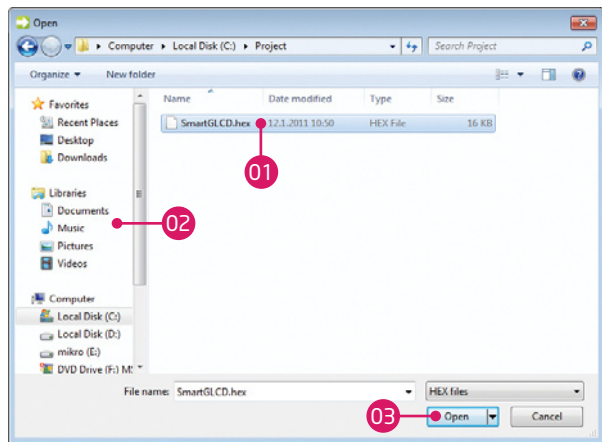


Figure 3-6: Selecting .hex file

- 01 Select desired .hex file
- 02 Folder list
- 03 Click on Open button

## step 5 - Uploading .hex file



Figure 3-7: Begin uploading

- 01 Click on Begin uploading button to start .hex file transfer from a PC to microcontroller

## step 6 - Progress bar

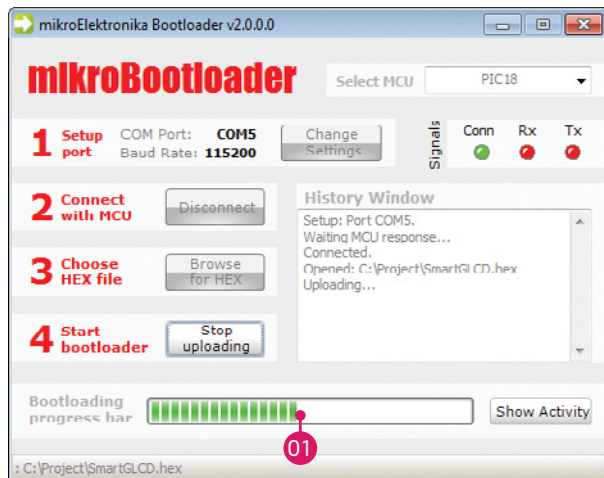


Figure 3-8: Bootloading progress bar

01 Via progress bar you can monitor .hex file uploading process

## step 7 - Reset MCU

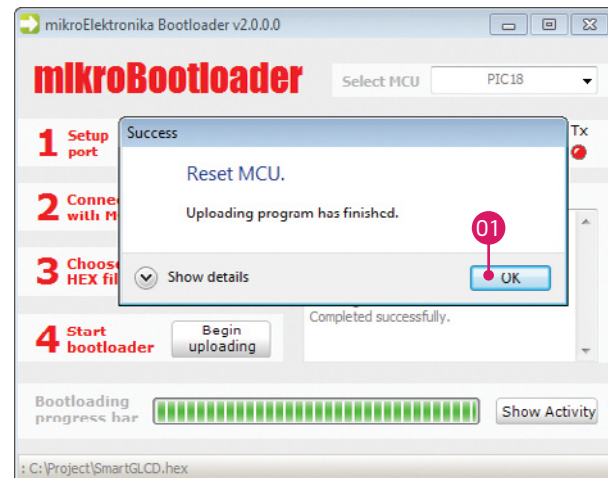
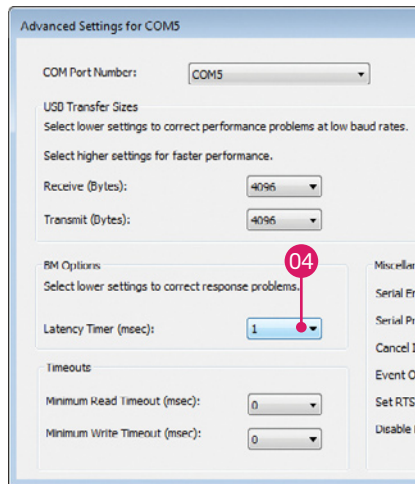
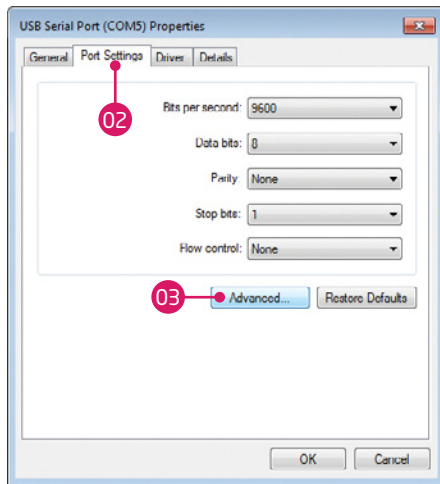
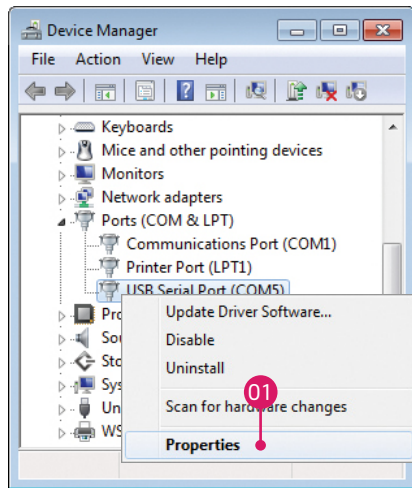


Figure 3-9: Uploading is finished

01 Click on OK button after uploading is finished. Reset MCU and you can see product of your work

# Tips and Tricks: Speed-up UART data transfer



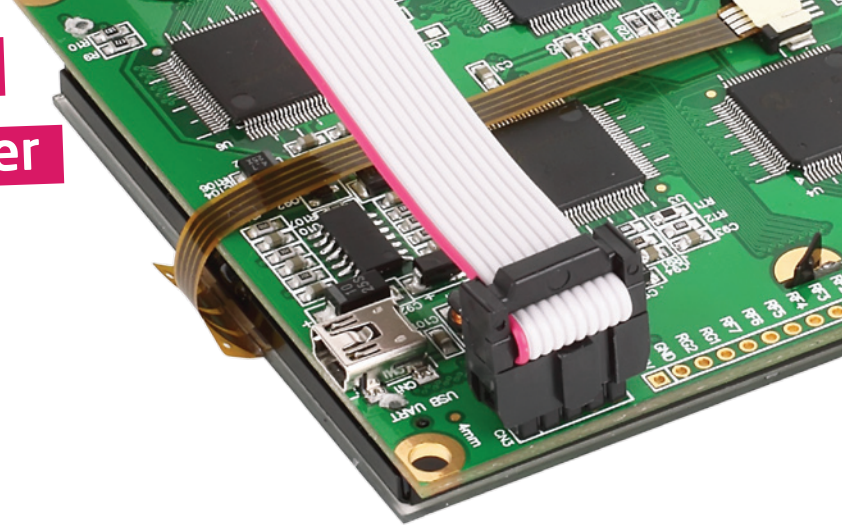
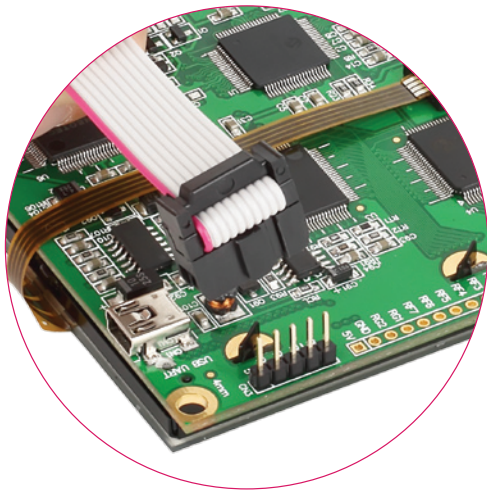
## note

*If .hex file transfer from your PC to MCU is too slow you can try to speed-up data transfer by setting latency time of COM port to 1. To change latency time go to Device manager:*

- 01 Right click on USB Serial Port (COM5) and click on Properties
- 02 In USB Serial Port (COM5) Properties select Port Settings tab
- 03 Click on Advanced... button
- 04 Set latency Timer to 1 (or chose another value) and click on OK button

## 4. Programming with external programmer

Figure 4-1: Connecting external programmer



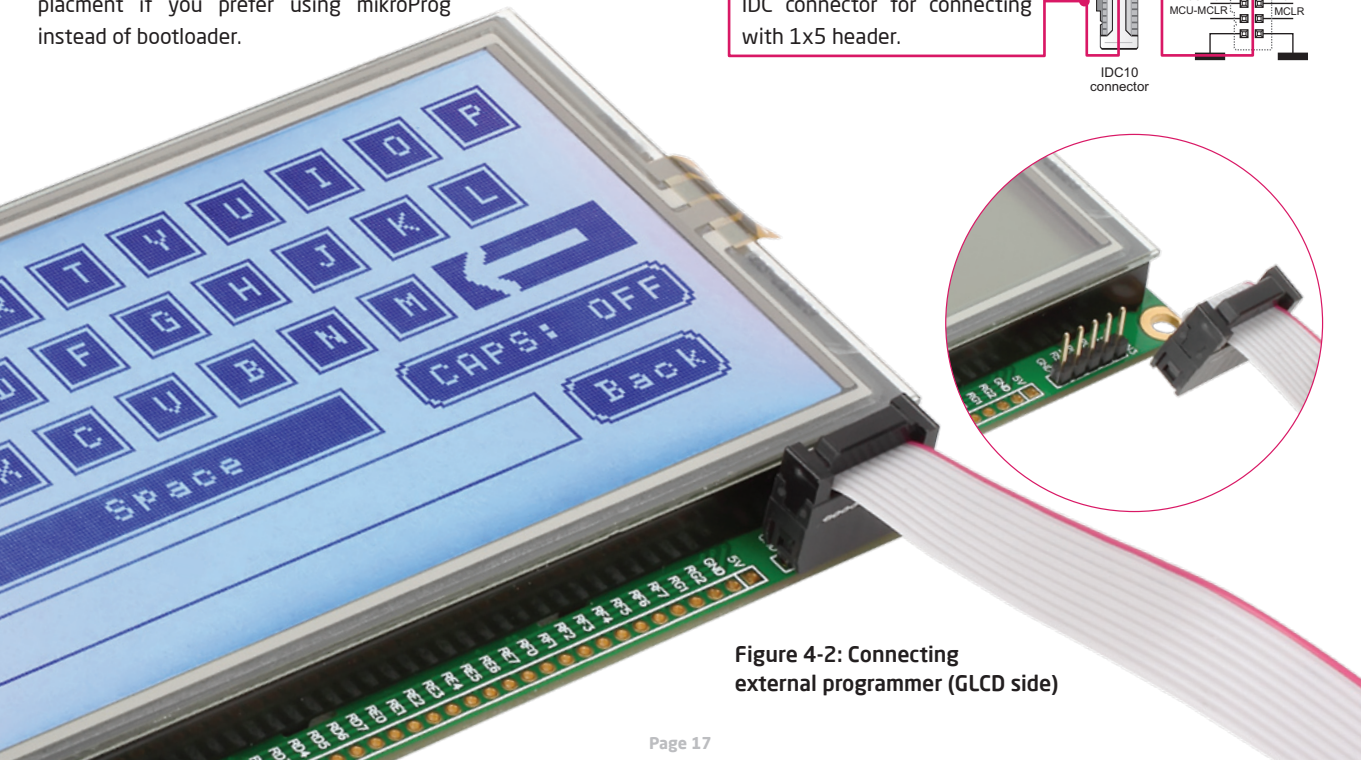
The microcontroller can be programmed with external programmer **mikroProg**. The external programmer is connected to the development system via marked pads CN3, Figure 4-1.

In order to connect the external programmer to the development system, it is necessary to provide a 1x5 header that should be soldered to pads CN3.

If bootloader program is accidentally erased you can upload it again via mikroProg programmer. Program **Bootloader18F8722.hex** can be found under Firmware folder (page 10).



mikroProg programmer can be also attached on the front side of the SmartGLCD. Just solder 1x5 header to pads CN3. This is ideal placement if you prefer using mikroProg instead of bootloader.



Always use side with a knob of IDC connector for connecting with 1x5 header.

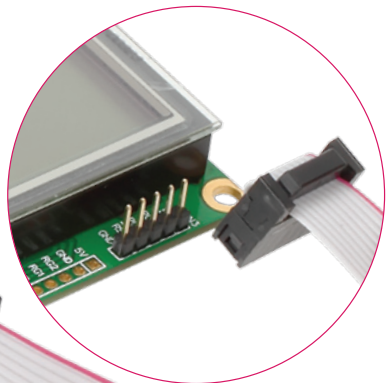
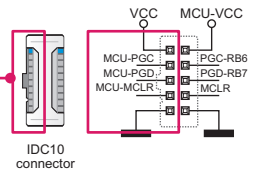


Figure 4-2: Connecting external programmer (GLCD side)

## 5. Touch Screen

The development system features a **Graphical LCD 240x128 display** covered with a **resistive** touch panel. Together they form a functional unit called a touch screen. It enables data to be entered and displayed at the same time. The way of entering and displaying data depends on the program loaded into the microcontroller.

Resistive touch panel is suitable for use with a plastic pen which have rounded tip. Finger press is always available but bare in mind that you use touch panel and there's no need for too much pressing force.

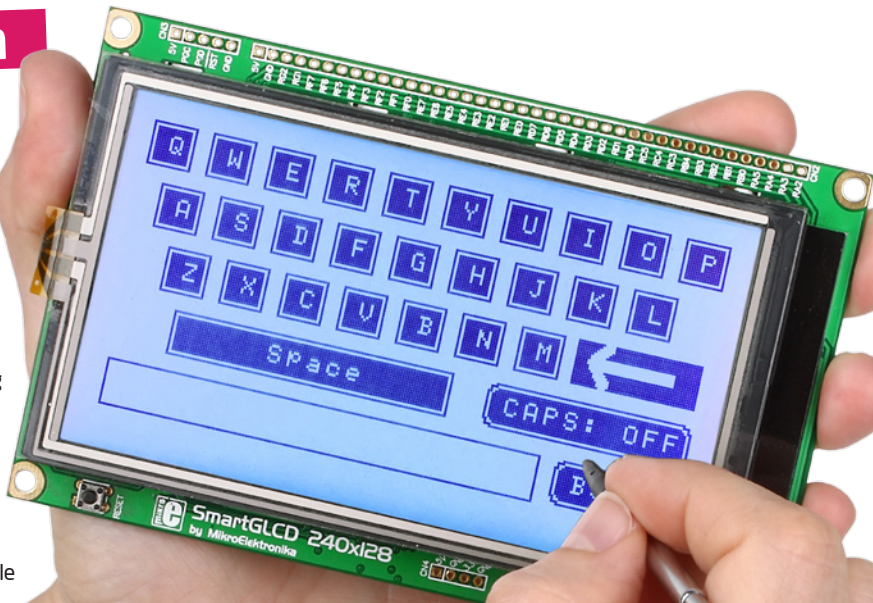


Figure 5-1:  
Touch Screen

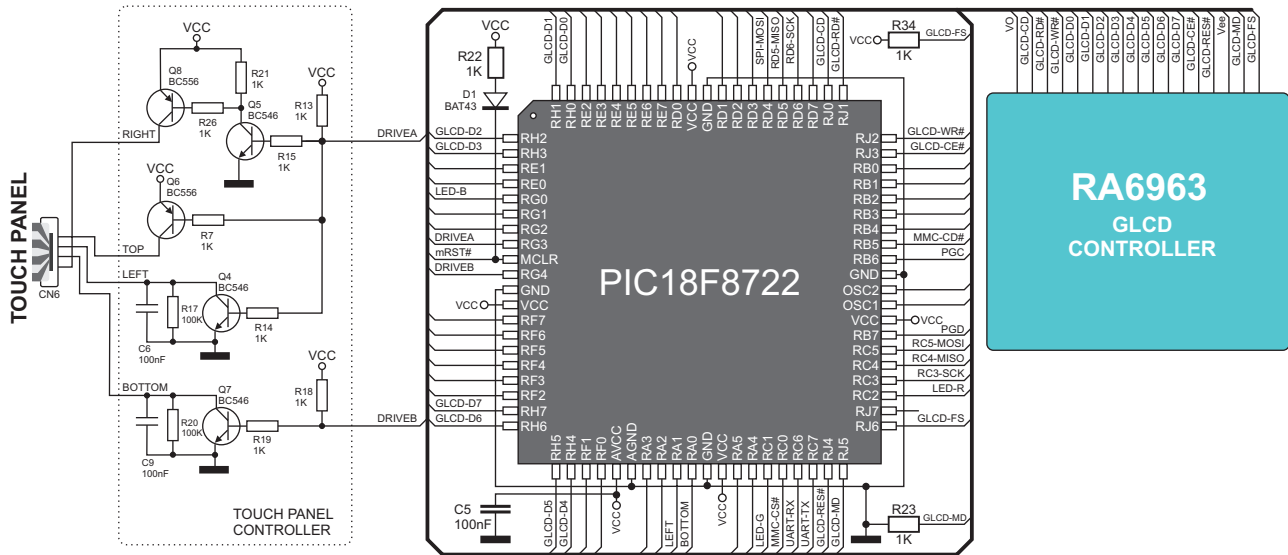


Figure 5-2: Touch screen connection schematic

## 6. microSD Card Slot

Figure 6-1: microSD card inserted in SmartGLCD

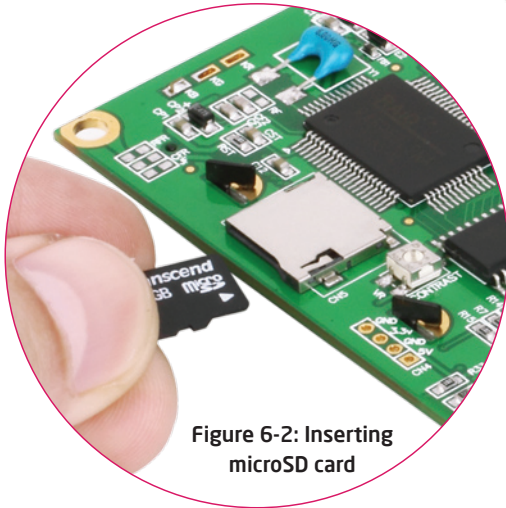
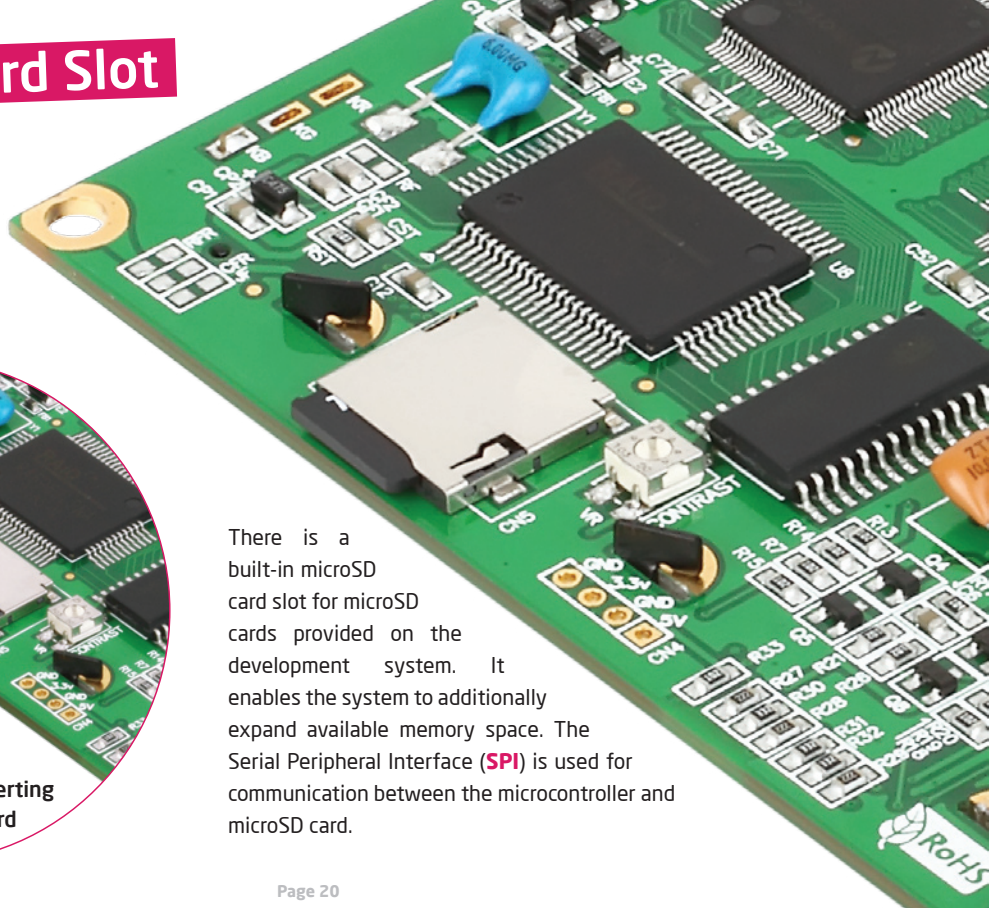


Figure 6-2: Inserting microSD card

There is a built-in microSD card slot for microSD cards provided on the development system. It enables the system to additionally expand available memory space. The Serial Peripheral Interface (**SPI**) is used for communication between the microcontroller and microSD card.

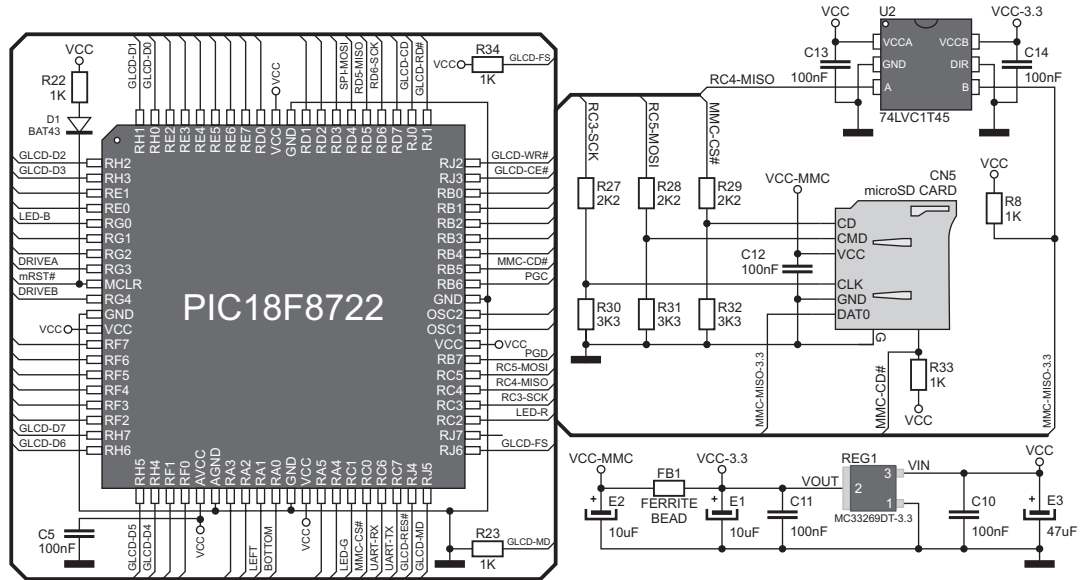


Figure 6-3: microSD card slot connecting schematic

# 7. USB UART

Development system can communicate with USB devices via USB UART module. This module comes in form of **FT232RL** chip which is interface between serial UART on MCU and USB device.

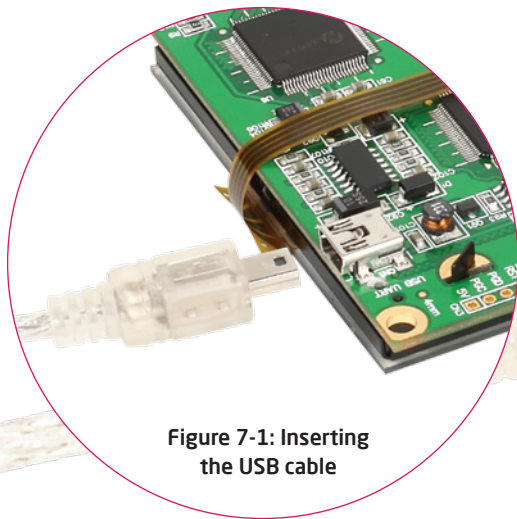


Figure 7-1: Inserting the USB cable

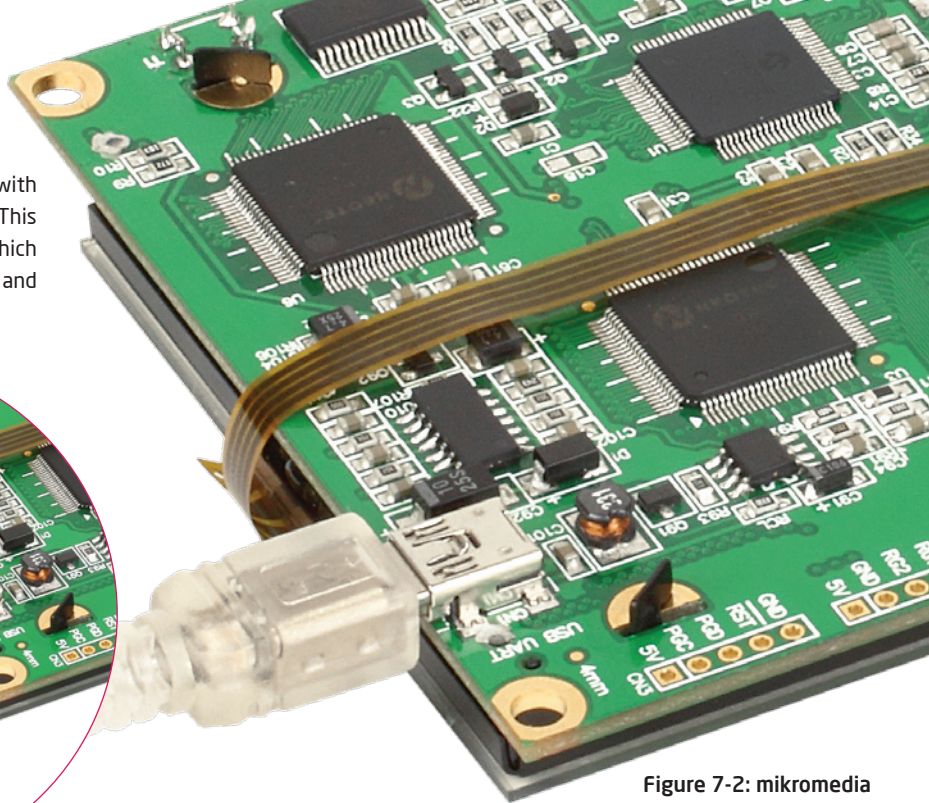


Figure 7-2: mikromedia connected with PC via USB cable

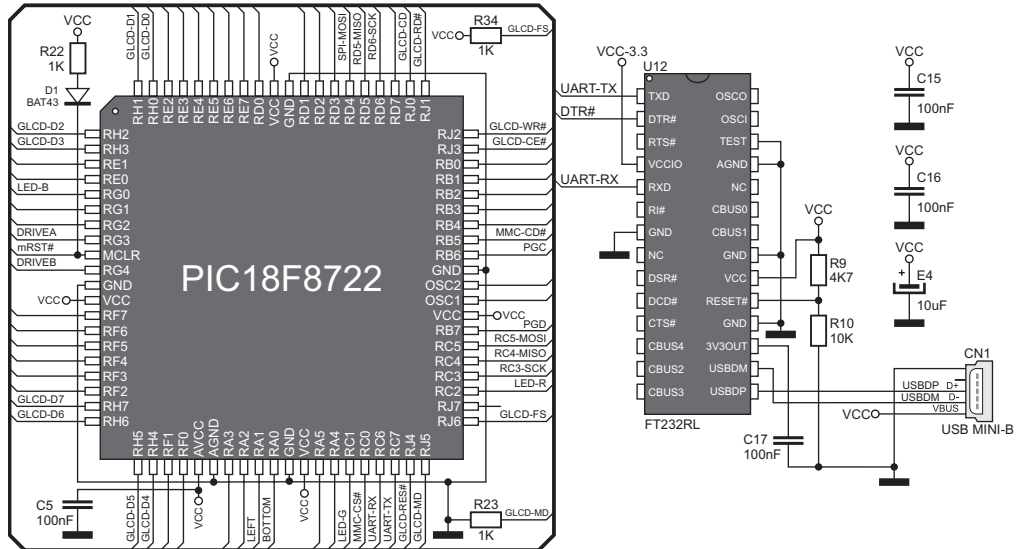
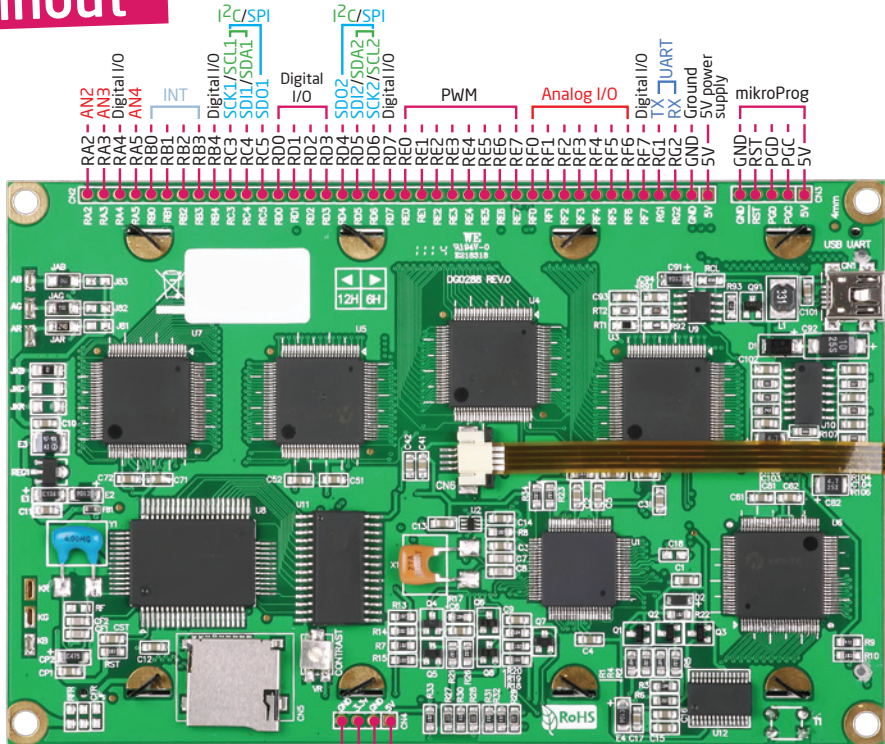


Figure 7-3: USB UART connecting schematic

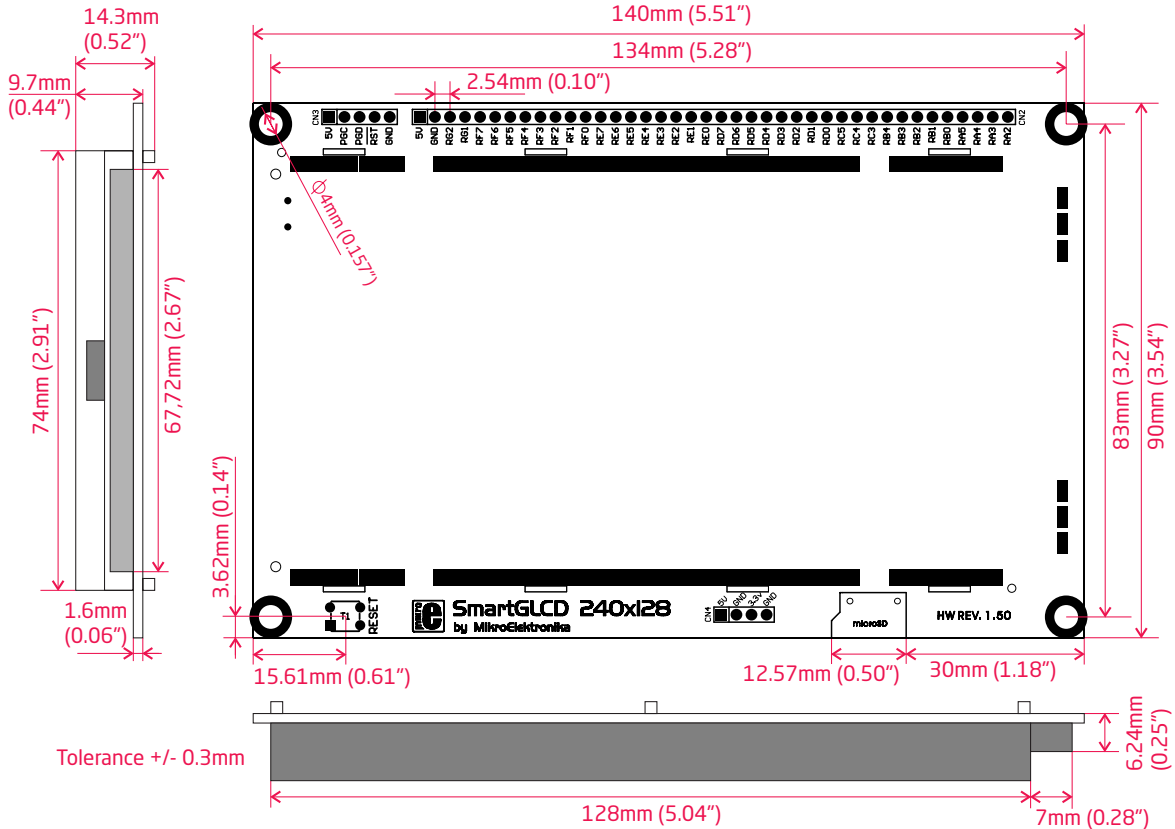
# 8. Pinout



- Analog Lines
- Interrupt Lines
- SPI Lines
- I2C Lines
- UART lines

Ground  
3.3V power supply output  
5V power supply  
Ground





Notes:

## DISCLAIMER

All the products owned by MikroElektronika are protected by copyright law and international copyright treaty. Therefore, this manual is to be treated as any other copyright material. No part of this manual, including product and software described herein, may be reproduced, stored in a retrieval system, translated or transmitted in any form or by any means, without the prior written permission of MikroElektronika. The manual PDF edition can be printed for private or local use, but not for distribution. Any modification of this manual is prohibited.

MikroElektronika provides this manual 'as is' without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties or conditions of merchantability or fitness for a particular purpose.

MikroElektronika shall assume no responsibility or liability for any errors, omissions and inaccuracies that may appear in this manual. In no event shall MikroElektronika, its directors, officers, employees or distributors be liable for any indirect, specific, incidental or consequential damages (including damages for loss of business profits and business information, business interruption or any other pecuniary loss) arising out of the use of this manual or product, even if MikroElektronika has been advised of the possibility of such damages. MikroElektronika reserves the right to change information contained in this manual at any time without prior notice, if necessary.

## HIGH RISK ACTIVITIES

The products of MikroElektronika are not fault - tolerant nor designed, manufactured or intended for use or resale as on - line control equipment in hazardous environments requiring fail - safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines or weapons systems in which the failure of Software could lead directly to death, personal injury or severe physical or environmental damage ('High Risk Activities'). MikroElektronika and its suppliers specifically disclaim any expressed or implied warranty of fitness for High Risk Activities.

## TRADEMARKS

The Mikroelektronika name and logo, the Mikroelektronika logo, mikroC, mikroC PRO, mikroBasic, mikroBasic PRO, mikroPascal, mikroPascal PRO, AVRflash, PICflash, dsPICprog, 18FJprog, PSOCprog, AVRprog, 8051prog, ARMflash, EasyPIC5, EasyPIC6, BigPIC5, BigPIC6, dsPIC PRO4, Easy8051B, EasyARM, EasyAVR5, EasyAVR6, BigAVR2, EasydsPIC4A, EasyPSoC4, EasyAVR Stamp LV18FJ, LV24-33A, LV32MX, PIC32MX4 MultiMedia Board, PICPLC16, PICPLC8 PICPLC4, SmartGSM/GPRS, UNI-DS are trademarks of Mikroelektronika. All other trademarks mentioned herein are property of their respective companies.

All other product and corporate names appearing in this manual may or may not be registered trademarks or copyrights of their respective companies, and are only used for identification or explanation and to the owners' benefit, with no intent to infringe.

# SmartGLCD

## 240x128

If you want to learn more about our products, please visit our website at [www.mikroe.com](http://www.mikroe.com)

If you are experiencing some problems with any of our products or just need additional information, please place your ticket at [www.mikroe.com/en/support](http://www.mikroe.com/en/support)

If you have any questions, comments or business proposals,  
do not hesitate to contact us at [office@mikroe.com](mailto:office@mikroe.com)