

UNI-DS6TM

User manual

All MikroElektronika's development systems represent irreplaceable tools for programming and developing microcontroller-based devices. Carefully chosen components and the use of machines of the last generation for mounting and testing thereof are the best guarantee of high reliability of our devices. Due to simple design, a large number of add-on modules and ready to use examples, all our users, regardless of their experience, have the possibility to develop their project in a fast and efficient way.

Development system

 **MikroElektronika**

SOFTWARE AND HARDWARE SOLUTIONS FOR EMBEDDED WORLD ...making it simple

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.



Nebojsa Matic
General Manager

Table of Contents

General information 4

Key features 5

1. Connecting UNI-DS6 to power supply module 6

2. mikroBoard 7

3. Placing mikroBoard 8

4. Programming microcontroller 9

5. USB UART1 and USB UART2 modules..... 10

6. ADC module 11

7. USB communication 12

8. EEPROM module 12

9. Piezo buzzer 13

10. DS1820 temperature sensor 14

11. MMC/SD connector 15

12. LEDs..... 16

13. Push buttons..... 17

14. 2x16 LCD display 18

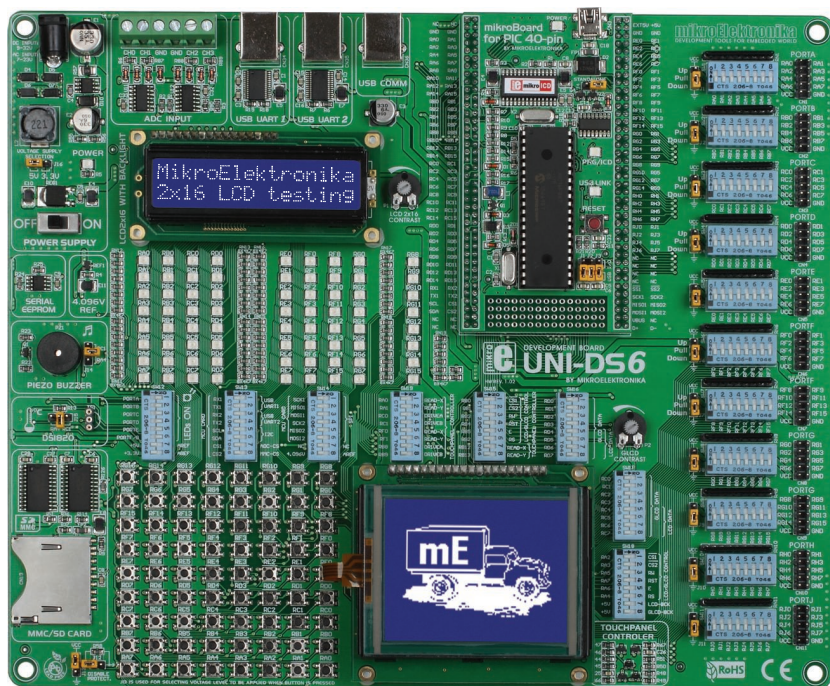
15. 128x64 graphic LCD display 19

16. Touch panel 20

17. Input/output ports..... 21

General information

The UNI-DS6 development system provides a development environment for programming and experimenting with various microcontrollers from different manufacturers. Numerous modules, such as 128x64 graphic LCD display, 2x16 alphanumeric LCD display, piezo buzzer, USB-UART, etc. are provided on the board and allow you to easily simulate the operation of your target device.



Full-featured development system for microcontroller based devices



UART communication via USB connector



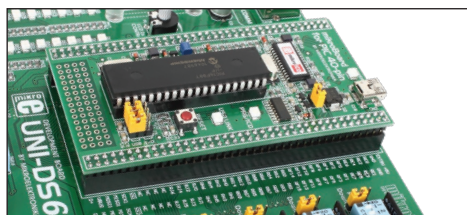
MMC/SD card connector



Integrated EEPROM module

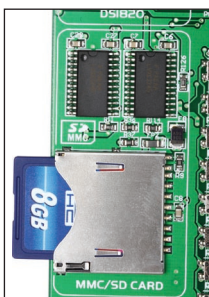
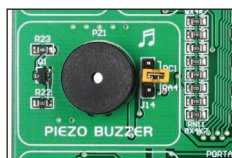


Graphic LCD display with backlight



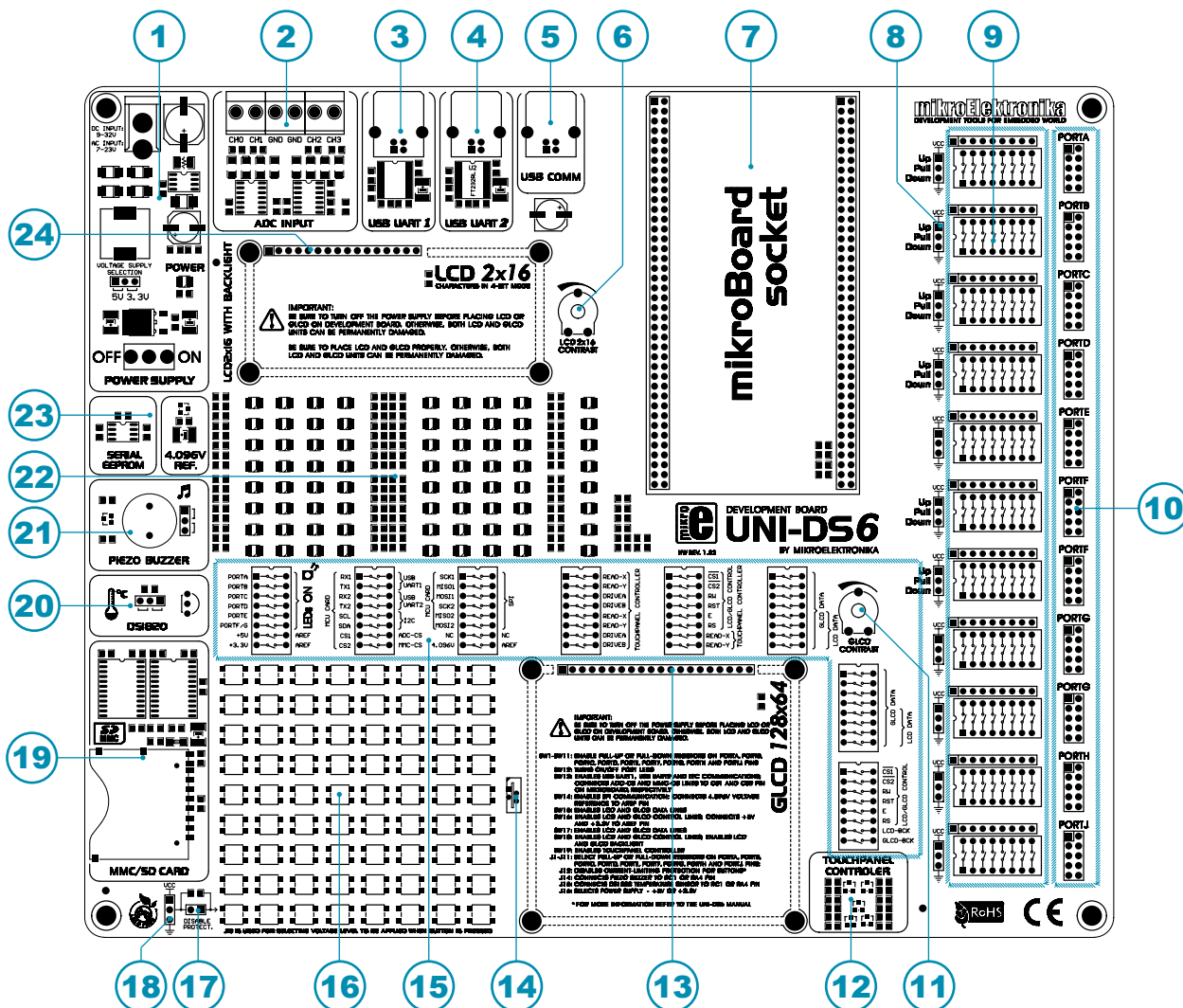
Package includes:

Development system:	UNI-DS6
CD:	product CD with relevant software
Cables:	USB cable
Documentation:	manual and electrical schematic for UNI-DS6



System specification:

Power supply:	over an AC/DC connector (7-23V AC or 9-32V DC) or a USB cable (5V DC)
Power consumption:	50mA when all on-board modules are off
Dimensions:	26,5 x 22cm (10,4 x 8,6inch)
Weight:	~420g (0.92lbs)



Key features

1. Power supply module
2. ADC input
3. USB UART1 module
4. USB UART2 module
5. USB communication connector
6. LCD2x16 display contrast potentiometer
7. mikroBoard socket
8. Jumpers used to select pull-up/pull-down resistors
9. DIP switches for enabling pull-up/pull-down resistors
10. I/O ports
11. GLCD contrast potentiometer
12. Touch panel controller
13. GLCD display connector
14. Touch panel connector
15. DIP switches for enabling on-board modules
16. Push buttons
17. Jumper used to shorten protective resistor
18. Jumpers used to select push buttons' logic state
19. MMC/SD card connector
20. Socket for DS18B20 temperature sensor
21. Piezo buzzer
22. LEDs
23. Serial EEPROM module
24. LCD display connector

1. Connecting UNI-DS6 to power supply module

In order to enable the development system to be turned on, it is necessary to provide the appropriate power supply voltage over an AC/DC connector CN19, Figure 1-1. When the development system is powered, it is necessary to set switch marked POWER SUPPLY to the ON position. The power supply voltage provided via the CN19 AC/DC connector may be in a range between 7 and 23V AC or 9 and 32V DC.

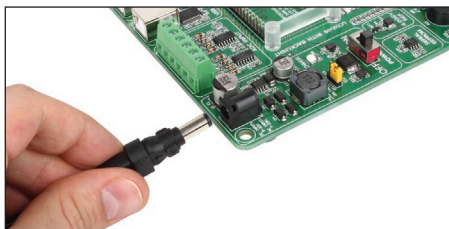


Figure 1-1: Powering the development system

A mikroBoard board with different voltage levels can be placed in the mikroBoard socket provided on the development system. The position of jumper J16 depends on the voltage level required. When a 5V mikroBoard is placed in the socket it is necessary to place jumper J16 in the 5V position. If a 3.3V mikroBoard is placed in the socket move jumper J16 in the 3.3V position.

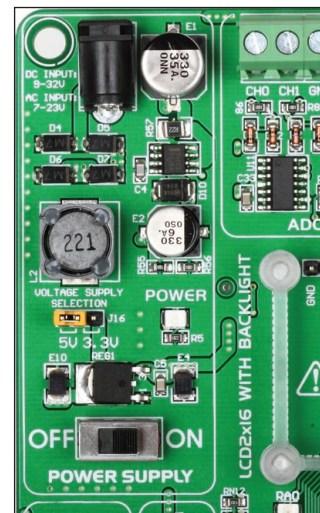


Figure 1-2: Power supply module

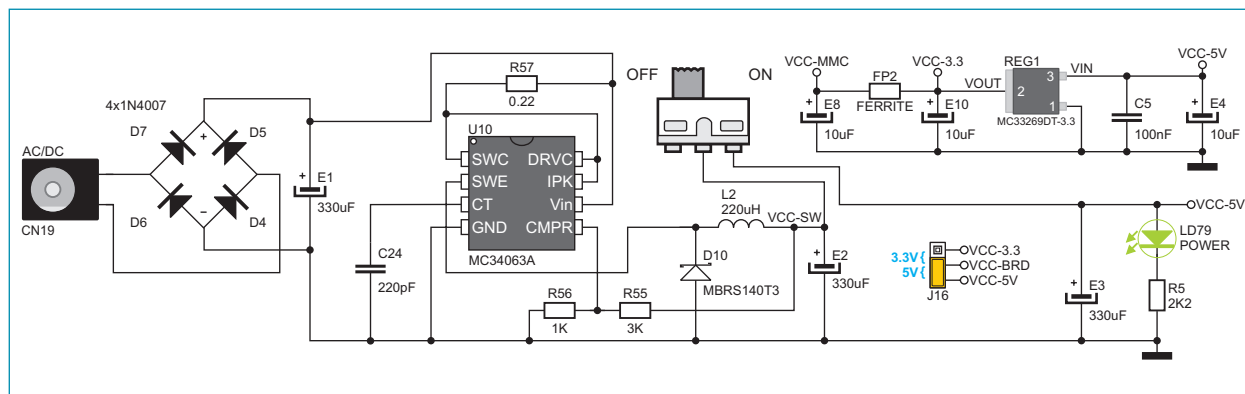


Figure 1-3: Power supply module connection schematic

2. mikroBoard

mikroBoard is designed for placing microcontroller on a development system. Every mikroBoard features an integrated programmer that is used for MCU programming. For connection with a development system, the mikroBoard uses two 2x40 male headers. In addition, the mikroBoard can be used as a standalone device.

There are several mikroBoard types: mikroBoard for 8051 40-pin, mikroBoard for AVR 64-pin, mikroBoard for dsPIC 80-pin, mikroBoard for PIC 40-pin, mikroBoard for PIC 80-pin, mikroBoard for ARM 64-pin, mikroBoard for ARM 144-pin and mikroBoard for PSoC.

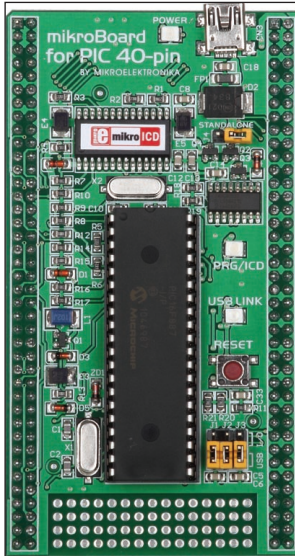


Figure 2-1: mikroBoard for PIC 40-pin

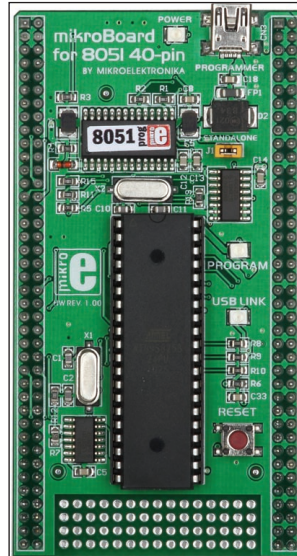


Figure 2-2: mikroBoard for 8051 40-pin

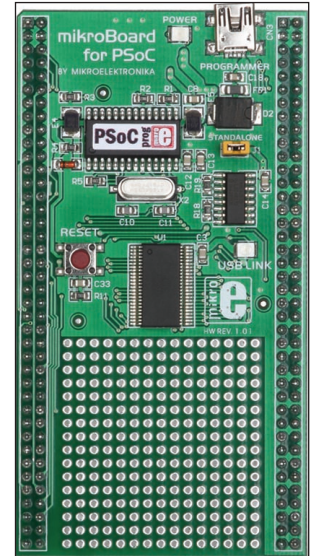
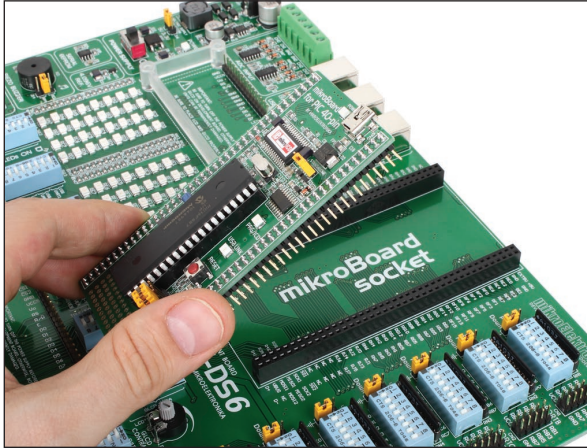


Figure 2-3: mikroBoard for PSoC

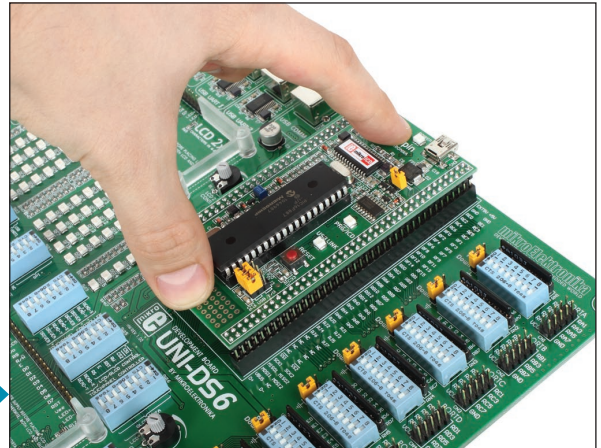


3. Placing mikroBoard

The UNI-DS6 development system is designed for usage with various mikroBoards. All the mikroBoards are placed in a universal mikroBoard socket, Figure 3-1. This socket consists of two 2x40 female headers. To place mikroBoard in this socket follow the steps below:

**STEP 1:**

Make sure that all header pins on mikroBoard are aligned with the mikroBoard socket

**STEP 2:**

Apply pressure on mikroBoard edges until it fully fits the socket

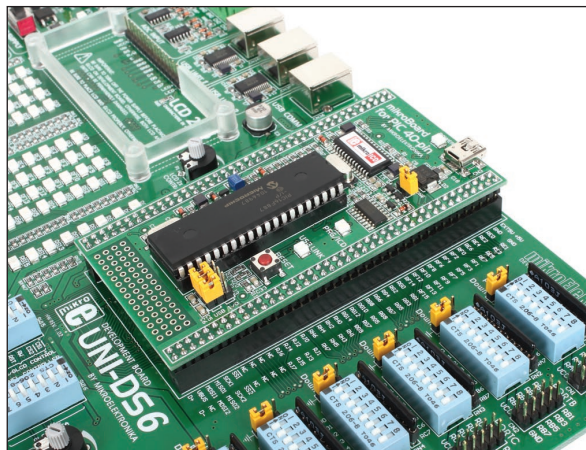


Figure 3-1: mikroBoard placed in socket

4. Programming microcontroller

The mikroBoard on the development system uses a built-in programmer for MCU programming. All you need to do is to connect the mikroBoard to a PC via a USB cable (Figure 4-1), and to install the appropriate software on your PC.

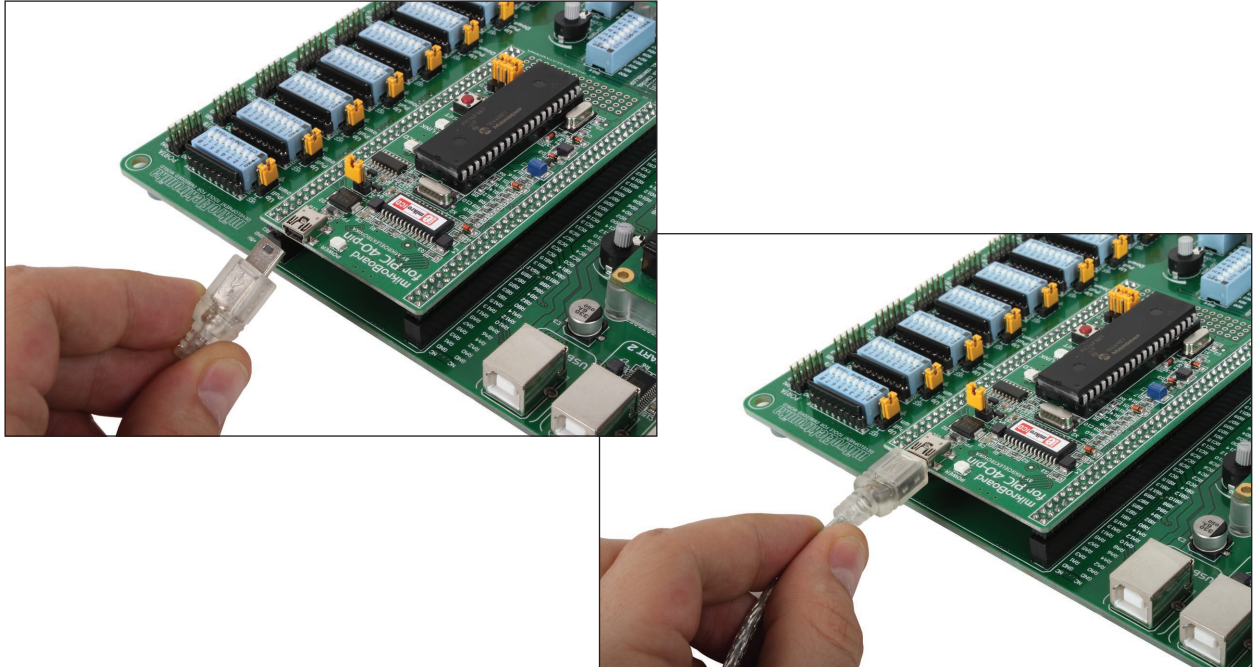


Figure 4-1: Connecting mikroBoard to PC via USB cable

Depending on which mikroBoard is in use it is necessary to install the appropriate software for MCU programming:

- mikroBoard for 8051 40-pin: 8051Flash
- mikroBoard for AVR 64-pin: AVRFlash
- mikroBoard for PSoC: PSoC Flash
- mikroBoard for dsPIC 80-pin, mikroBoard for PIC 40-pin, mikroBoard for PIC 80-pin: mikroProg Suite for PIC
- mikroBoard for ARM 64-pin, mikroBoard for 144-pin: ARMflash

To download flash software visit Mikroelektronika's website at www.mikroe.com

5. USB UART1 and USB UART2 modules

USB UART modules enable the UNI-DS6 development system to be connected to a PC via a USB connector. In addition to PC, the development system can also be easily connected to other devices that use USB communication. USB UART modules are connected to the microcontroller supplied on the development system via #RX232A and #TX232A pins for USB UART1 or #RX232B and #TX232B for USB UART2.

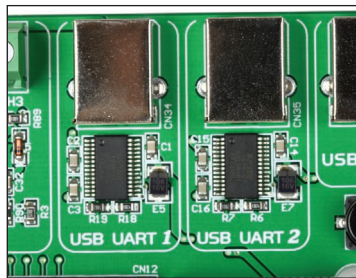


Figure 5-1: USB UART modules

In order to establish connection between the USB UART1 module and the microcontroller, it is necessary to set switches 1 and 2 on the DIP switch SW13 to the ON position. To connect the USB UART2 module and the microcontroller, it is necessary to set switches 3 and 4 on the DIP switch SW13 to the ON position.

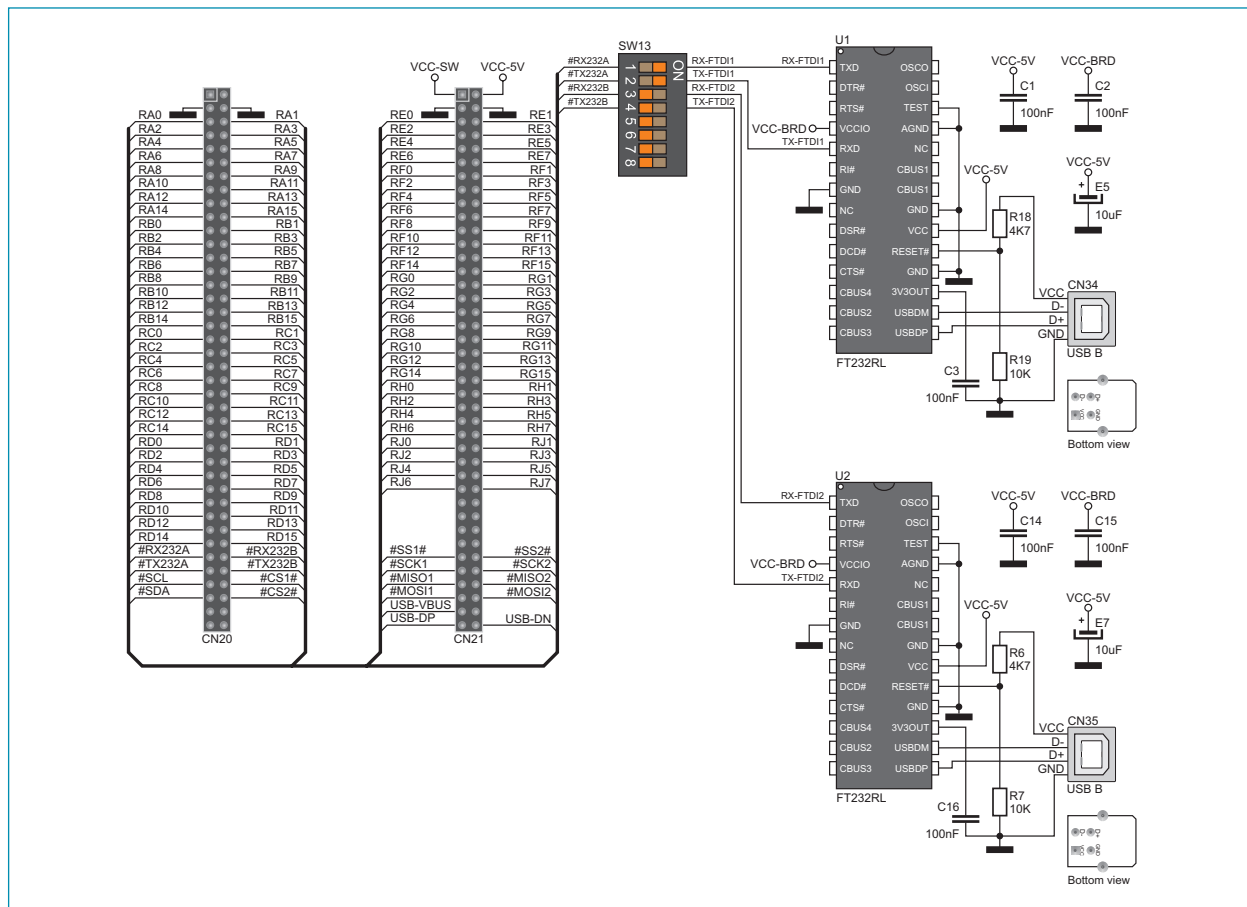


Figure 5-2: USB UART modules connection schematic

6. ADC module

The ADC module is used to convert an analog voltage level into the appropriate 12-bit digital value. The analog voltage is supplied via screw terminals CN15 and CN16. The voltage supplied via the VREF pin is used as a voltage reference. In order to use this voltage, switch 8 on the DIP switch SW14 should be set to the ON position.

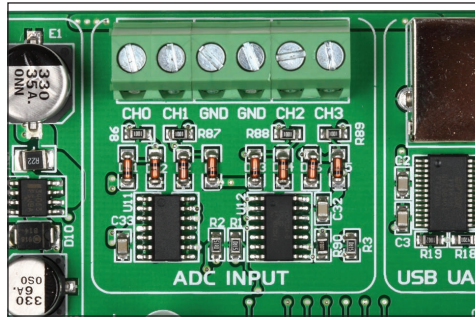


Figure 6-1: ADC module

Serial SPI communication is used for data transfer between the ADC module and the microcontroller. In order to establish connection between the ADC module and the microcontroller, it is necessary to set switches 1, 2 and 3 on the DIP SW14 switch to the ON position. Optionally, you can use switches 4, 5 and 6 on the DIP switch SW14.

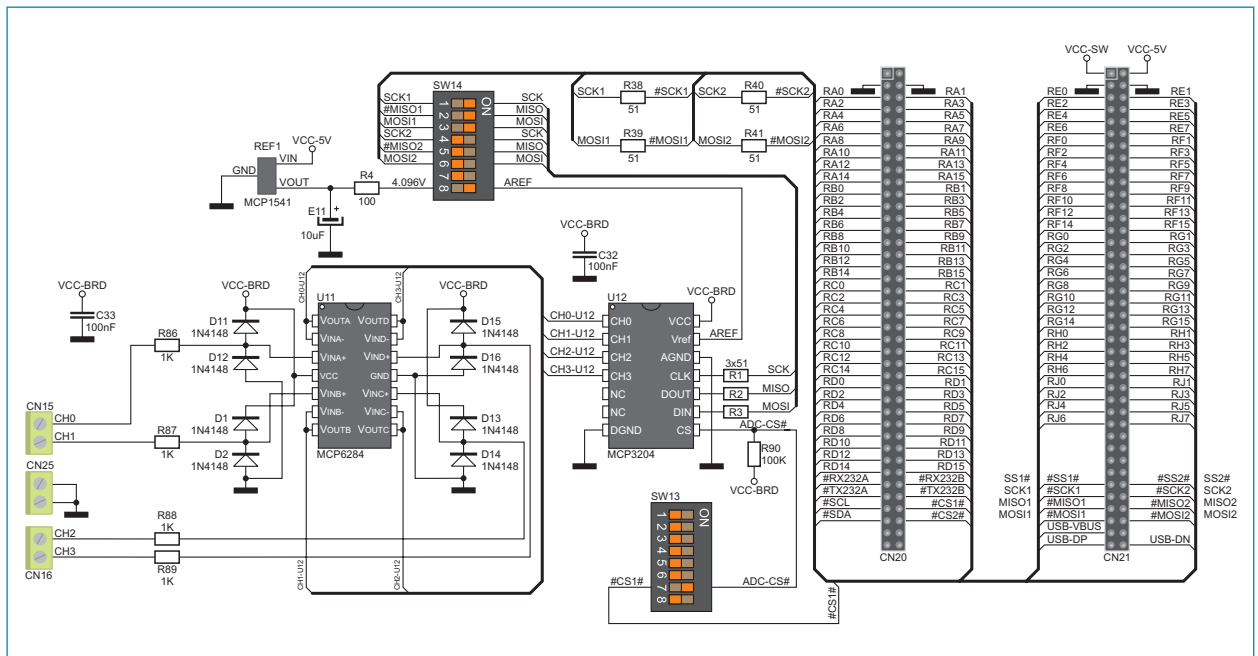


Figure 6-2: ADC module connection schematic

page

page



page



page

page

page



page



page

9. Piezo buzzer

Due to a built-in piezo buzzer, the UNI-DS6 development system is capable of emitting audio signals. In order to enable the piezo buzzer to operate properly it is necessary to generate a voltage signal of specific frequency. Remember, when writing code for voltage signal generation, that the piezo buzzer's resonant frequency is 3.8kHz. Other frequencies in the range between 20Hz and 20kHz can also be used, but the best performance is provided with frequencies ranging between 2kHz and 4kHz. To establish connection between the piezo buzzer and the microcontroller, it is necessary to place jumper J14 in adequate position. If jumper J14 is placed in the RC1 position the RC1 MCU pin is used for signal generation, Figure 9-2. Otherwise place jumper J14 in the RA4 position in order to use the RA4 MCU pin for signal generation, Figure 9-3.

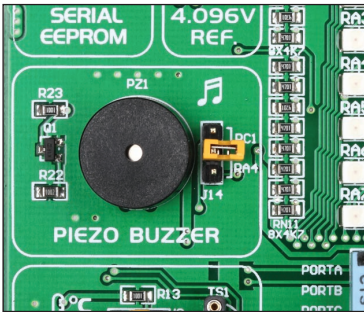


Figure 9-1: Piezo buzzer

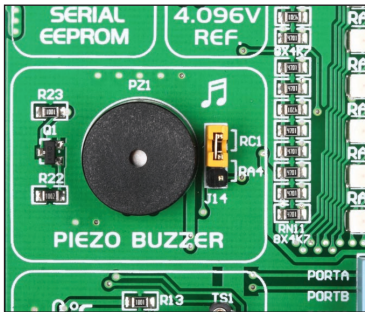


Figure 9-2: Signal generation via RC1

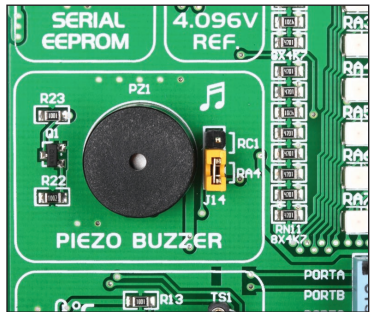


Figure 9-3: Signal generation via RA4

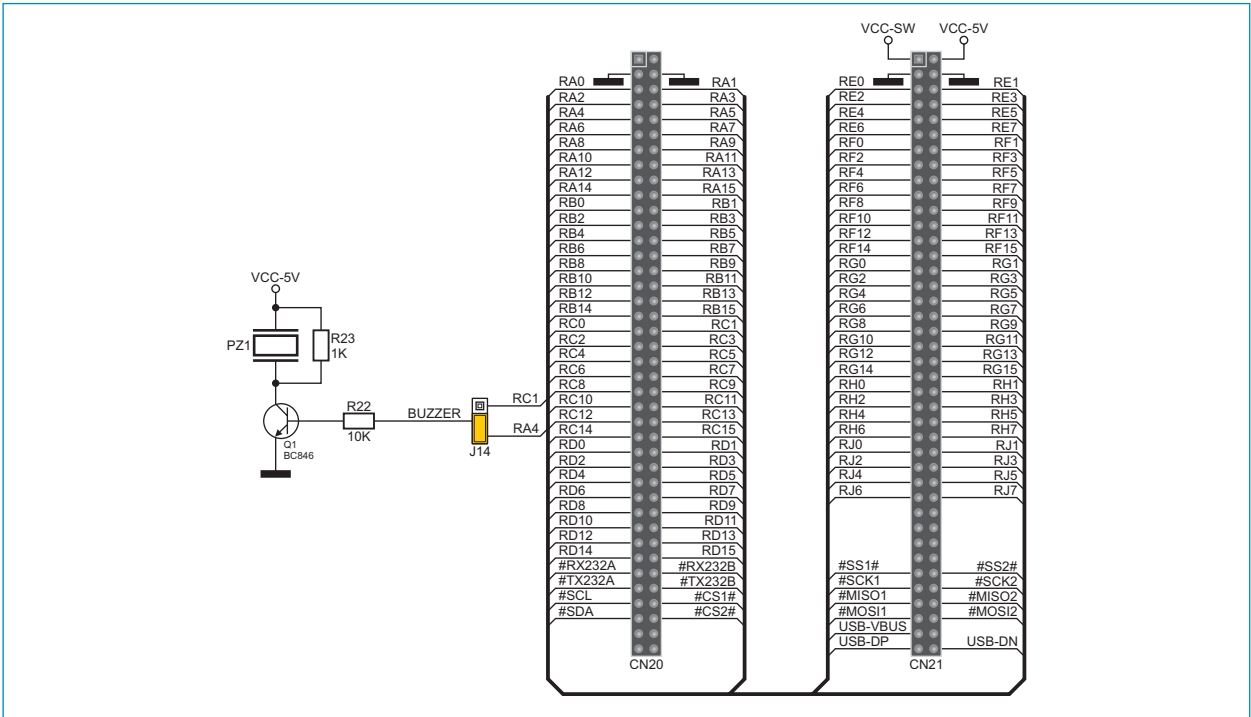


Figure 9-4: Piezo buzzer connection schematic

10. DS1820 temperature sensor

DS1820 is a temperature sensor that uses 1-wire communication for its operation. It is used to measure temperature in a range between -55 and 125°C and provides $\pm 0.5^{\circ}\text{C}$ accuracy for temperatures in a range between -10 and 85°C . The power supply voltage of 3.3V to 5V is used for the operation of this sensor. It takes maximum 750ms for the DS1820 to convert temperature with 9-bit resolution. There is a socket for this temperature sensor provided on the development system. Communication between this module and the microcontroller is enabled via the microcontroller pins RC1 and RA4. To use pin RC1 place jumper J15 in the RC1 position and for the RA4 pin place jumper J15 in the RA4 position.

1-wire® serial communication enables data to be transferred over one single communication line, while the process itself is under control of the *master* microcontroller. The advantage of this communication is that only one microcontroller pin is used. All slave devices have a unique ID code, which enables the master device to easily identify all devices sharing the same communication bus.



Figure 10-1: DS1820 connector (DS1820 is not connected)

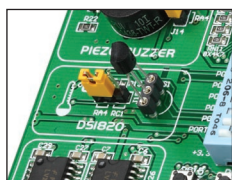


Figure 10-2: Temperature sensor DS1820 is connected via pin RA4



Figure 10-3: Temperature sensor DS1820 is connected via pin RC1

NOTE:

Make sure that the rounded side of the DS1820 matches half-circle on the board

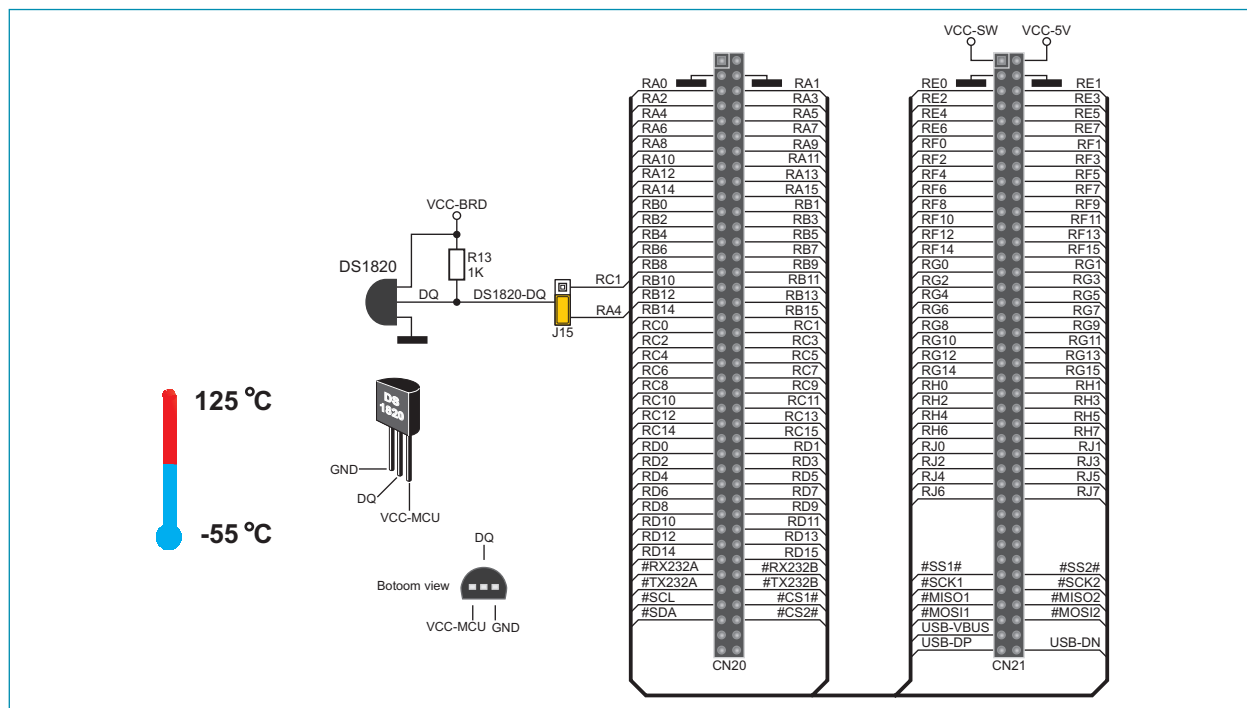


Figure 10-4: DS1820 and microcontroller connection schematic

11. MMC/SD connector

The UNI-DS6 development system is capable of reading memory cards due to the on-board MMC/SD connector. Memory card communicates with the microcontroller through the microcontroller pins used for serial communication. In order to establish connection between MMC/SD cards and the microcontroller, it is necessary to set switches 1, 2 and 3 (optionally 4, 5 and 6) on the DIP switch SW14, as well as switch 8 on the DIP switch SW13 to the ON position.

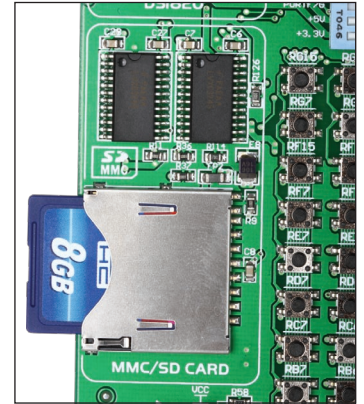


Figure 11-1: MMC/SD memory card

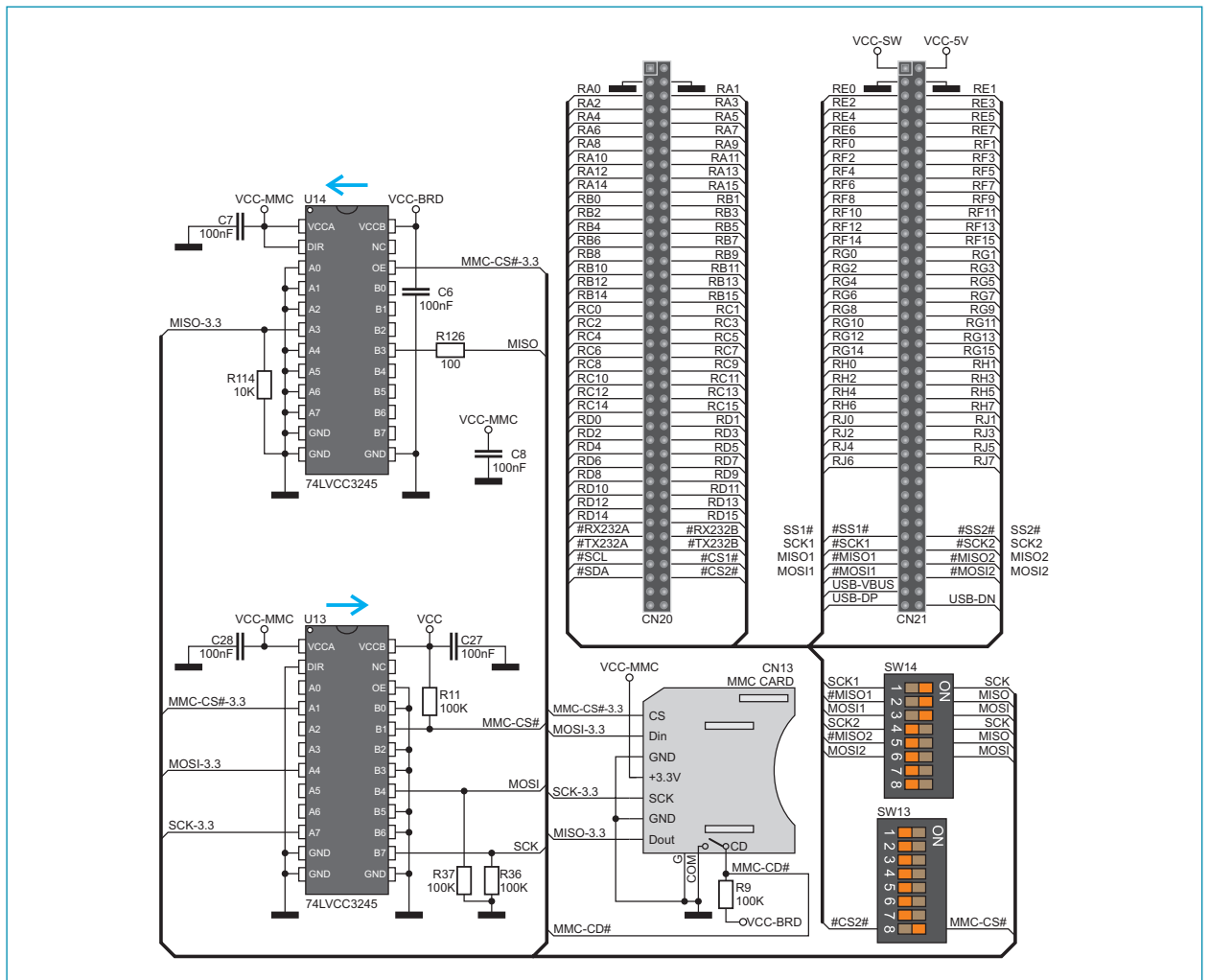


Figure 11-2: MMC/SD connector and microcontroller connection schematic

12. LEDs

There are 72 LEDs on the UNI-DS6 development system used to visually indicate the state of each microcontroller I/O pin. An active LED indicates that a logic one (1) is present on the pin. In order to enable LEDs to illuminate, it is necessary to select the appropriate port (PORTA, PORTB, PORTC, PORTD, PORTE or PORTF/G) by using DIP switch SW12. Ports PORTH and PORTJ are not connected to LEDs.

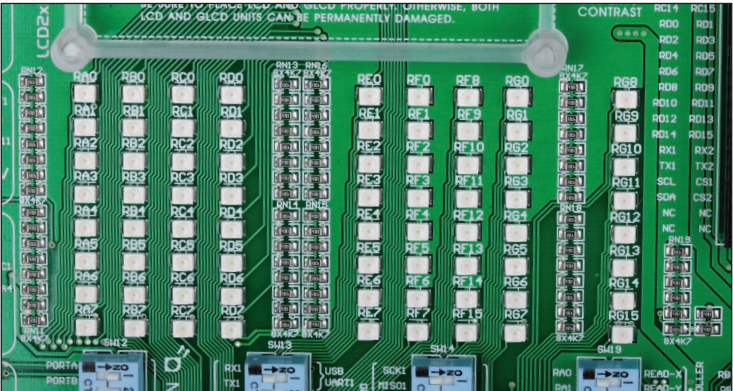


Figure 12-1: LEDs

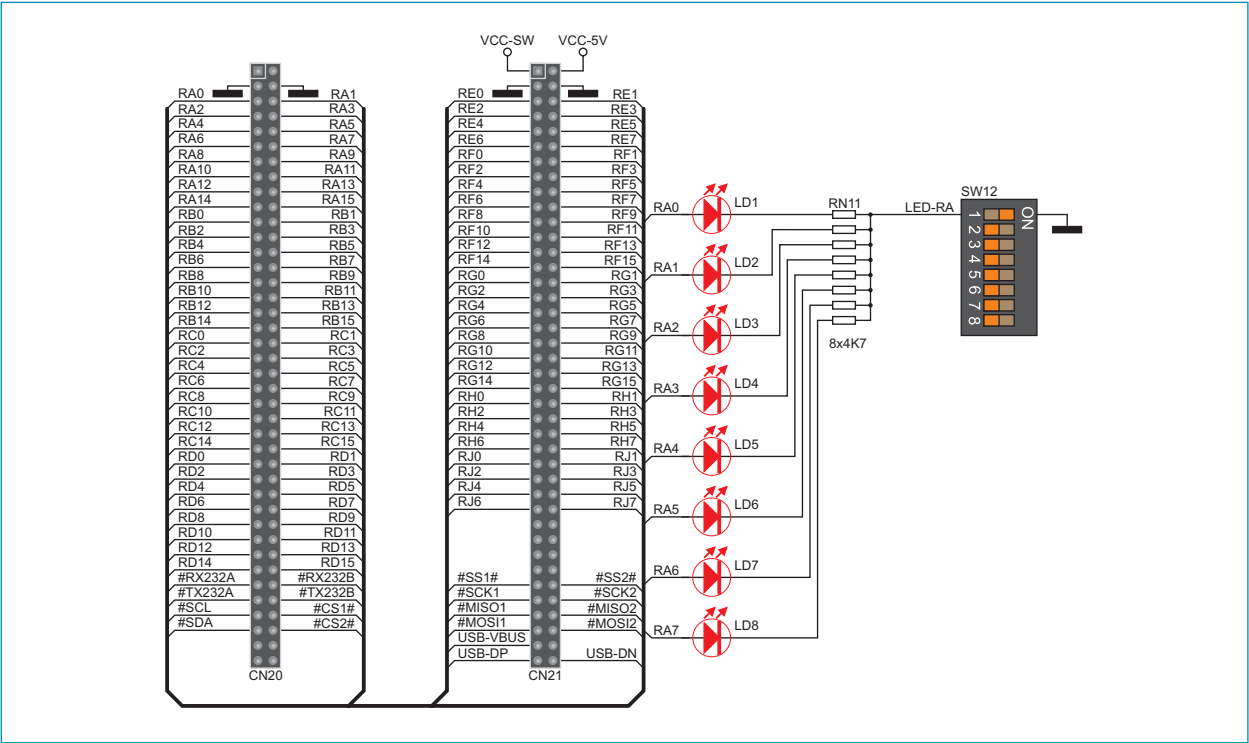
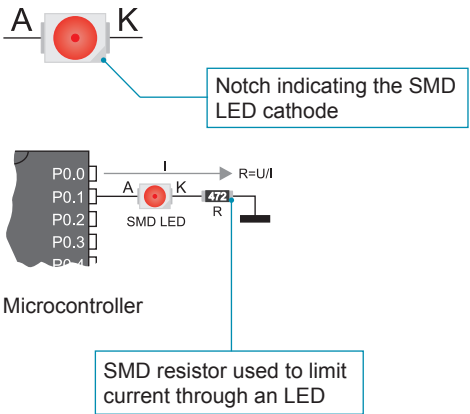


Figure 12-2: LED and port PORT0 connection schematic

13. Push buttons

The logic level of all microcontroller input pins may be changed by using push buttons. Jumper J13 is used to determine the logic level to be supplied on the appropriate microcontroller pin by pressing a push button. The function of the protective resistor is to limit the maximum current, thus preventing the development system and peripheral modules from being damaged in case a short circuit occurs. If needed, advanced users may shorten this resistor by using jumper J12.

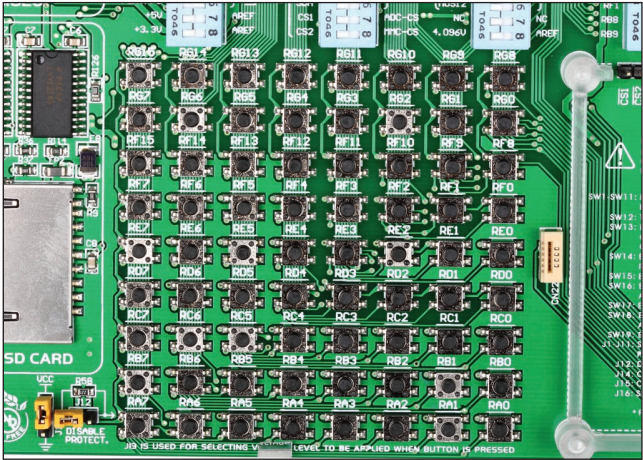


Figure 13-1: Push buttons

By pressing any push button when jumper J13 is in the VCC-BRD position, a logic one (3.3V or 5V) will be applied to the appropriate microcontroller pin, as shown in Figure 13-2.

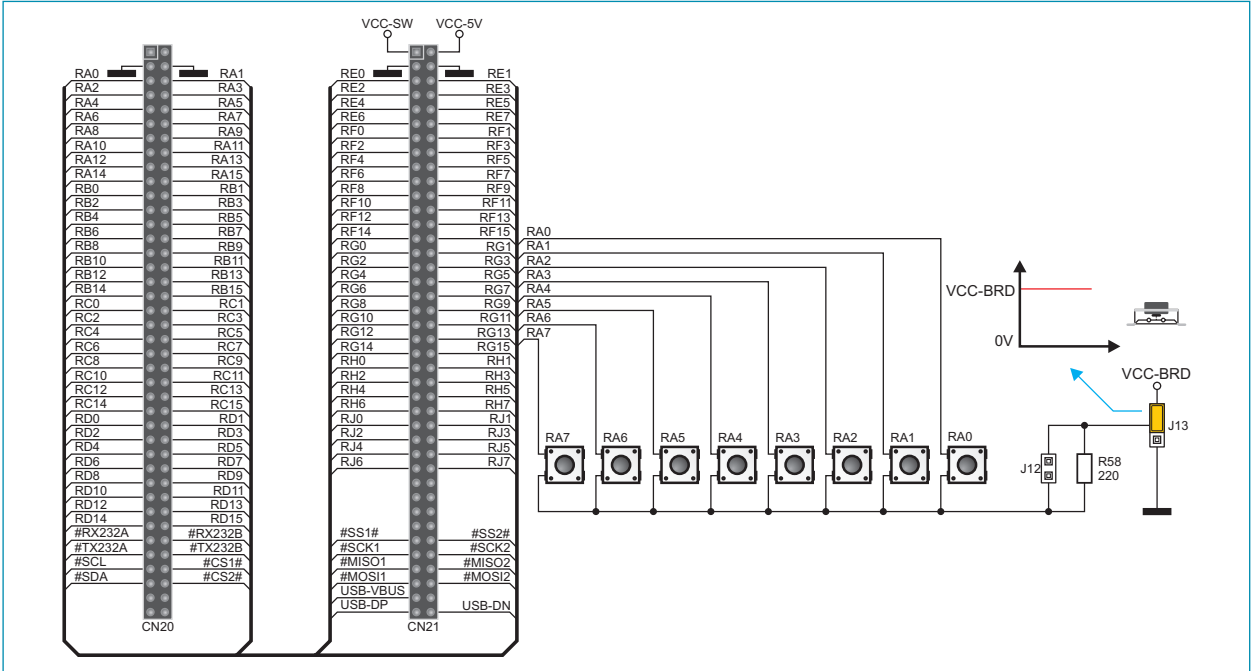


Figure 13-2: Push buttons and port PORT0 connection schematic

14. 2x16 LCD display

The UNI-DS6 development system features an on-board connector for the alphanumeric 2x16 LCD display. This connector is linked to the microcontroller via DIP switches (SW18 (PORTA) or SW15 (PORTB)) and (SW16 (PORTD) or SW17 (PORTC)). Potentiometer P1 is used to adjust display contrast. The LCD-BCK switch on the DIP switch SW18 is used to turn the display backlight on/off.

To enable the 2x16 LCD display it is necessary to write a program which defines which MCU pins will be used for communication between the 2x16 LCD display and the MCU. For data transfer you can use PORTD or PORTC pins on MCU via DIP switch SW16 or SW17. For display control you can use PORTA and PORTB on MCU via DIP switches SW15 and SW18.

Communication between this LCD and the microcontroller is performed in a 4-bit mode. Alphanumeric digits are displayed in two lines each containing up to 16 characters of 7x5 pixels.

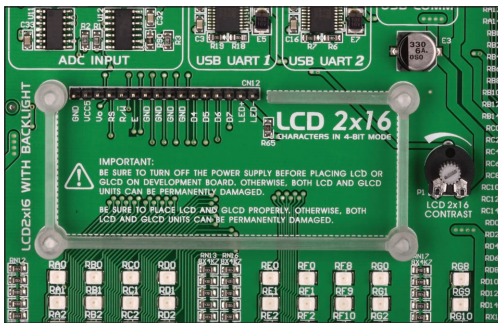


Figure 14-1: Connector for alphanumeric LCD display

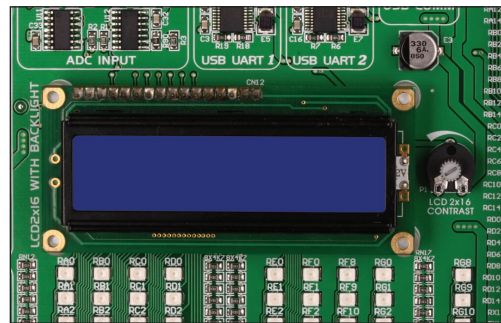


Figure 14-2: Alphanumeric 2x16 LCD display

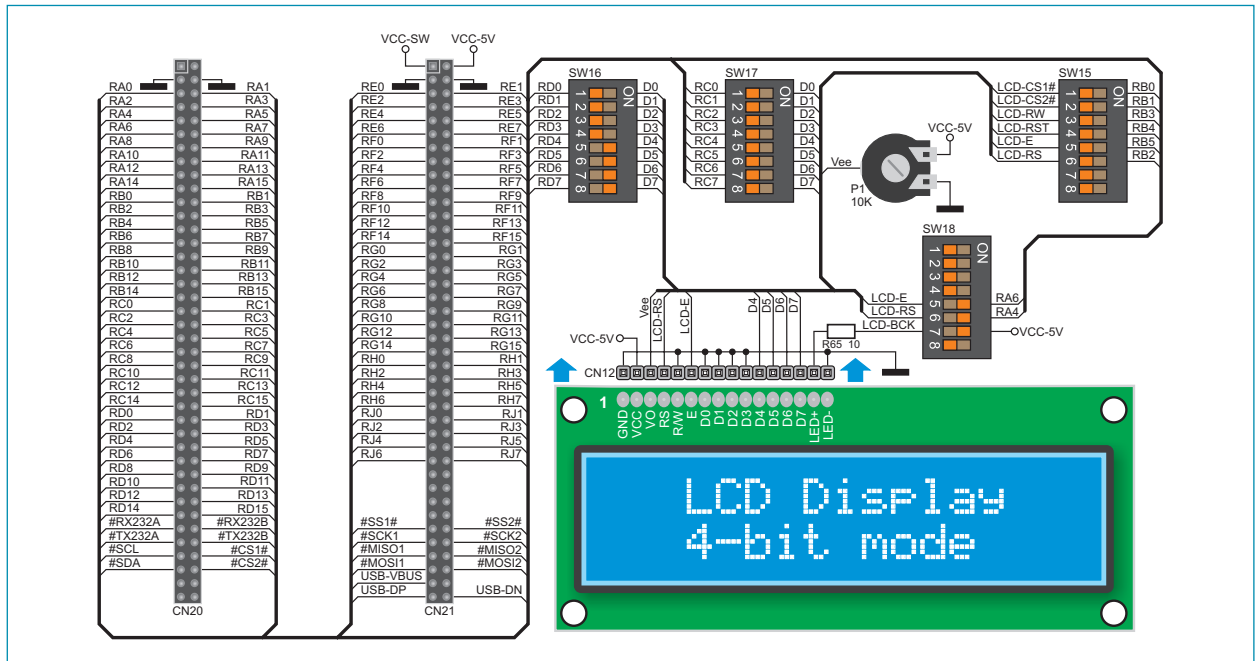


Figure 14-3: 2x16 LCD display connection schematic

15. 128x64 graphic LCD display

128x64 graphic LCD (GLCD) is connected to the microcontroller via DIP switches (SW18 (PORTA) or SW15 (PORTB)) and (SW16 (PORTD) or SW17 (PORTC)). It has a screen resolution of 128x64 pixels, which allows diagrams, tables and other graphic contents to be displayed. Potentiometer P2 is used for the GLCD display contrast adjustment. Switch 8 (GLCD-BCK) on the DIP switch SW18 is used to turn the display backlight on/off.

To enable the GLCD display it is necessary to write a program which defines which MCU pins will be used for communication between the GLCD display and the MCU. For data transfer you can use PORTD or PORTC pins on MCU via DIP switch SW16 or SW17. For display control you can use PORTA and PORTB on MCU via DIP switches SW15 and SW18.

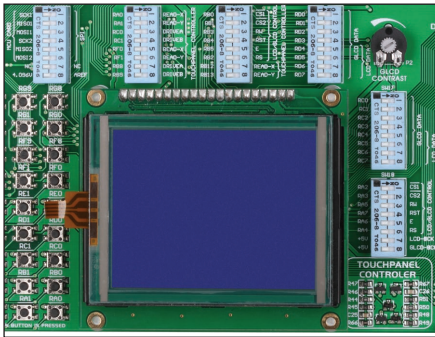


Figure 15-1: GLCD display

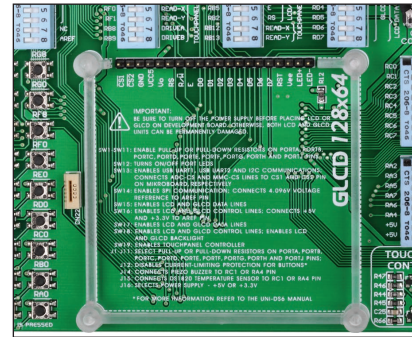


Figure 15-2: GLCD connector

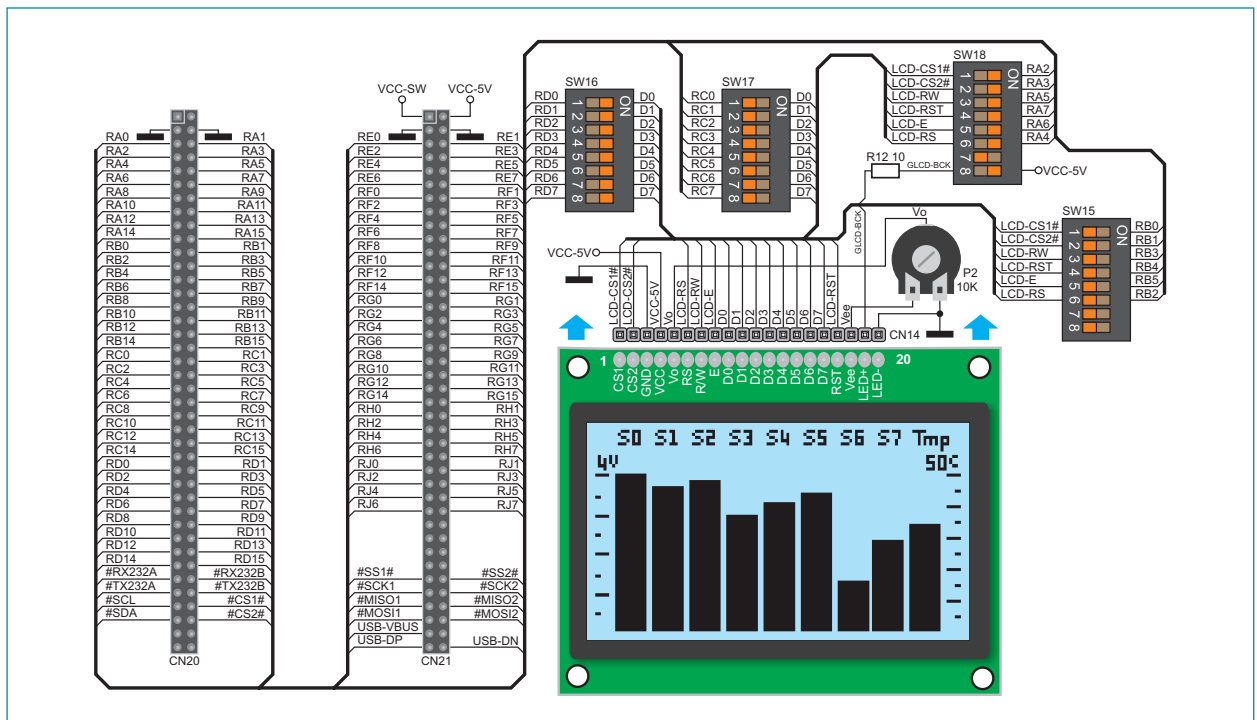


Figure 15-3: GLCD display connection schematic

16. Touch panel

A touch panel is a thin, self-adhesive, transparent, touch-sensitive panel. It is placed over a GLCD display. Its main function is to register pressure at some specific display point and to forward its coordinates in the form of analog voltage to the microcontroller. Switches 5, 6, 7 and 8 on the DIP switch SW19 are used to connect the microcontroller and touch panel.

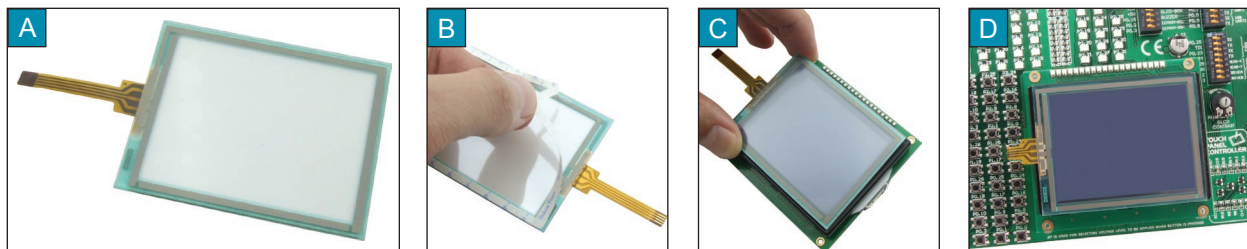


Figure 16-1: Placing touch panel over a GLCD

Figure 16-1 shows how to place a touch panel over a GLCD display. Make sure that the flat cable is to the left of the GLCD, as shown in Figure 1D.

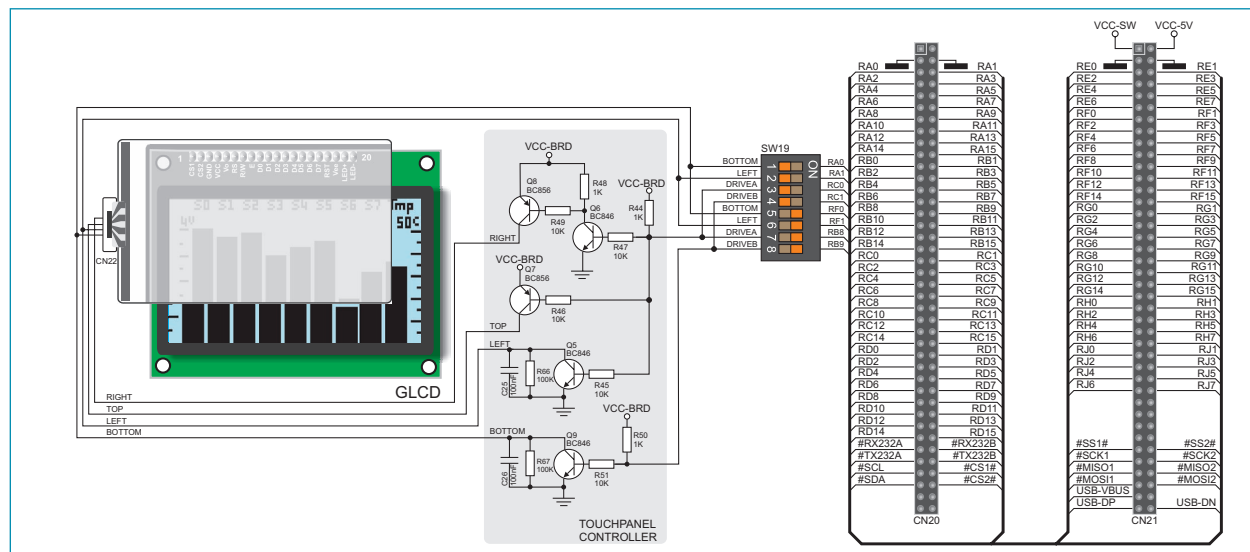


Figure 16-2: Touch panel connection schematic



Figure 16-3: Connecting touch panel

Figure 16-3 shows in detail how to connect a touch panel to the microcontroller. Bring the end of the flat cable close to the CN22 connector (Figure 3A). Plug the cable into the connector (Figure 3B) and press it easily so as to fully fit the connector (Figure 3C). Now, a GLCD can be plugged into the appropriate connector (Figure 3D).

NOTE: LEDs and pull-up/pull-down resistors on ports which are in use should be off when the touch panel is in use.

17. Input/output ports

Along the right side of the development system, there are eleven 10-pin connectors linked to the microcontroller I/O ports. Pull-up or pull-down resistors can be connected to I/O ports via jumpers J1-J11 and DIP switches SW1-SW11.



Figure 17-1: I/O ports

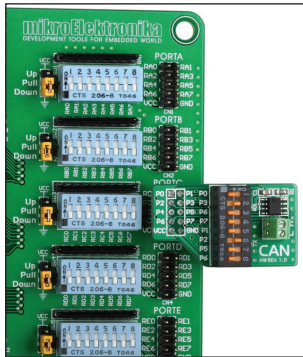


Figure 17-2: Additional board connected to I/O port

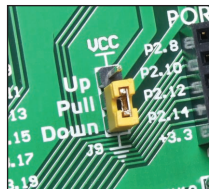


Figure 17-3: J9 in pull-down position

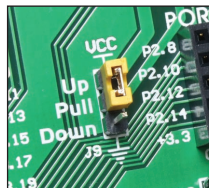


Figure 17-4: J9 in pull-up position

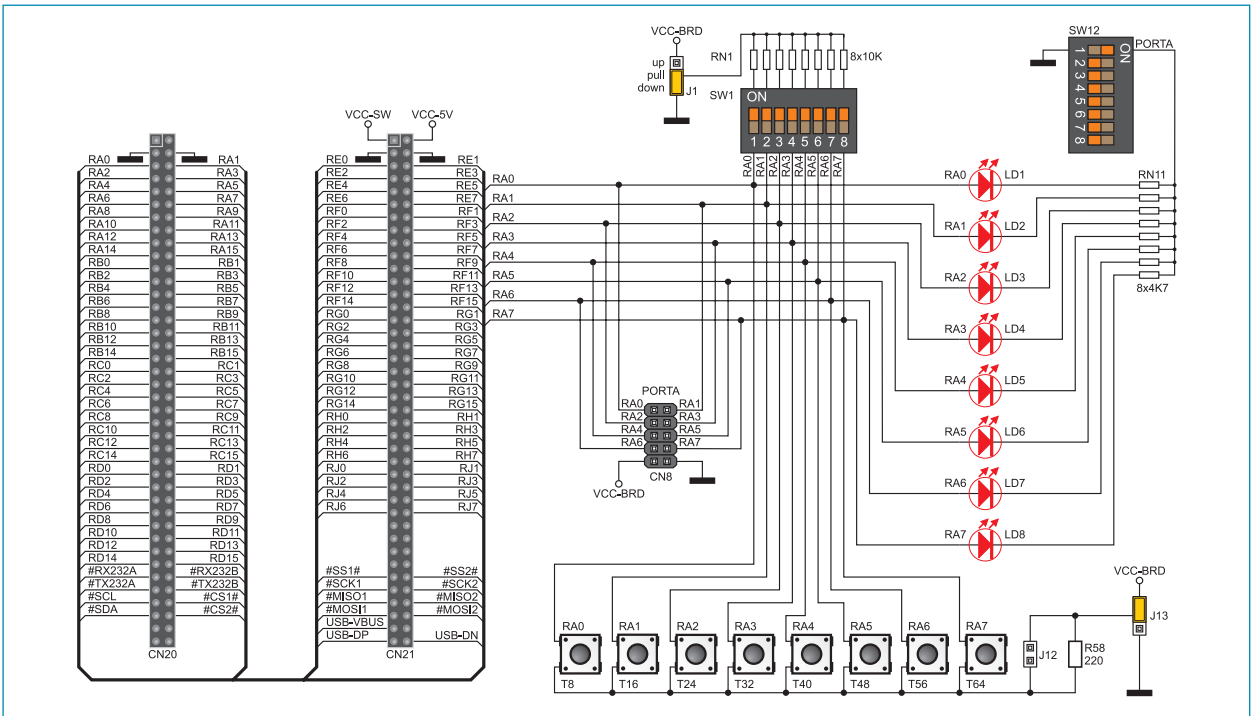


Figure 17-5: Port PORTA connection schematic

Pull-up/pull-down resistors enable you to feed all microcontroller's input pins with logic level when they are in idle state. This level depends on the position of the pull-up/pull-down jumper (J1-J11). The RA0 pin with the relevant jumper J1 and RA0 push button with jumper J13 are used here for the purpose of explaining the performance of pull-up/pull-down resistors. The principle of their operation is the same for all other microcontroller pins.

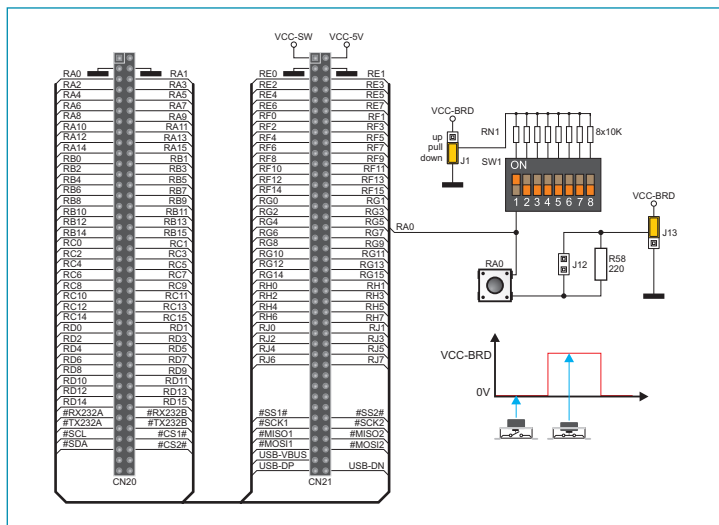


Figure 17-6: Jumper J1 in pull-down and jumper J13 in pull-up position

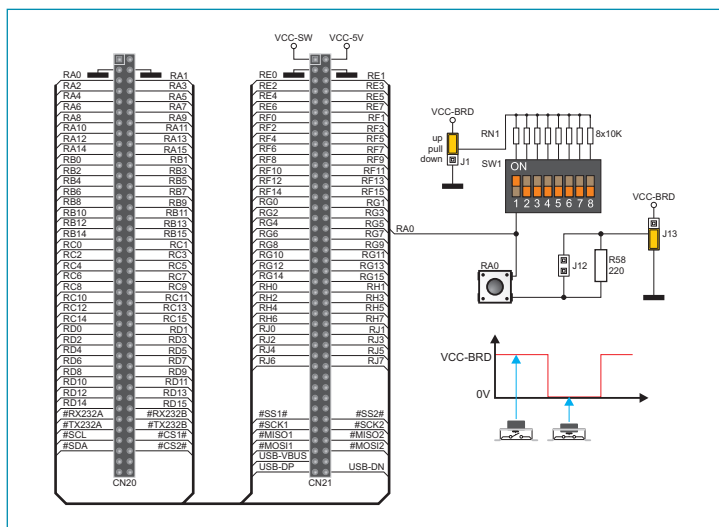


Figure 17-7: Jumper J1 in pull-up and jumper J13 in pull-down position

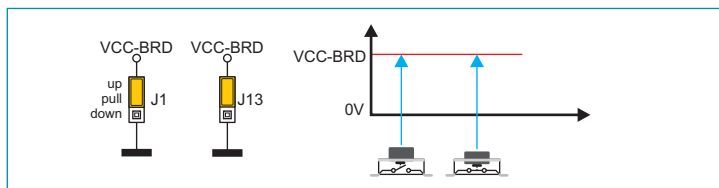


Figure 17-8: Jumpers J1 and J13 in the same positions

In order to enable the PORTA pins to be connected to pull-down resistors, it is necessary to place jumper J1 in the *Down* position first. This enables any PORTA port pin to be supplied with a logic zero (0V) in idle state over jumper J1 and 8x10k resistor network. To provide the RA0 pin with this signal, it is necessary to set switch 1 on the DIP switch SW1 to the ON position.

As a result, every time you press the RA0 push button, the RA0 pin will be fed with a logic one (VCC-BRD voltage), provided that jumper J13 is placed in the VCC-BRD position.

In order to enable the PORTA pins to be connected to pull-up resistors and the port input pins to be supplied with a logic one (1), it is necessary to place jumper J1 in the *Up* position and jumper J13 in the GND position. This enables any port PORTA input pin, when it is in idle state, to be driven high (VCC-BRD) over the 10k resistor.

As a result, every time you press the RA0 push button, the RA0 pin will be fed with a logic zero (0V), provided that switch 1 on the DIP switch SW1 is set to the ON position.

In case that jumpers J1 and J13 are in the same positions, pressure on any button will not cause input pins to change their logic state.

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, mikro-Basic PRO, mikroPascal, mikroPascal PRO, AVRflash, PICflash, dsPICprog, 18FJprog, PSOCprog, AVR-prog, 8051prog, ARMflash, EasyPIC5, EasyPIC6, BigPIC5, BigPIC6, dsPIC PRO4, Easy8051B, EasyARM, EasyAVR5, EasyAVR6, BigAVR2, EasydsPIC4A, EasyPSoC4, EasyVR 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.



SOFTWARE AND HARDWARE SOLUTIONS FOR EMBEDDED WORLD ...making it simple

If you want to learn more about our products, please visit our website at 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

If you have any questions, comments or business proposals, do not hesitate to contact us at office@mikroe.com