



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of "Quality Parts,Customers Priority,Honest Operation,and Considerate Service",our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



Contact us

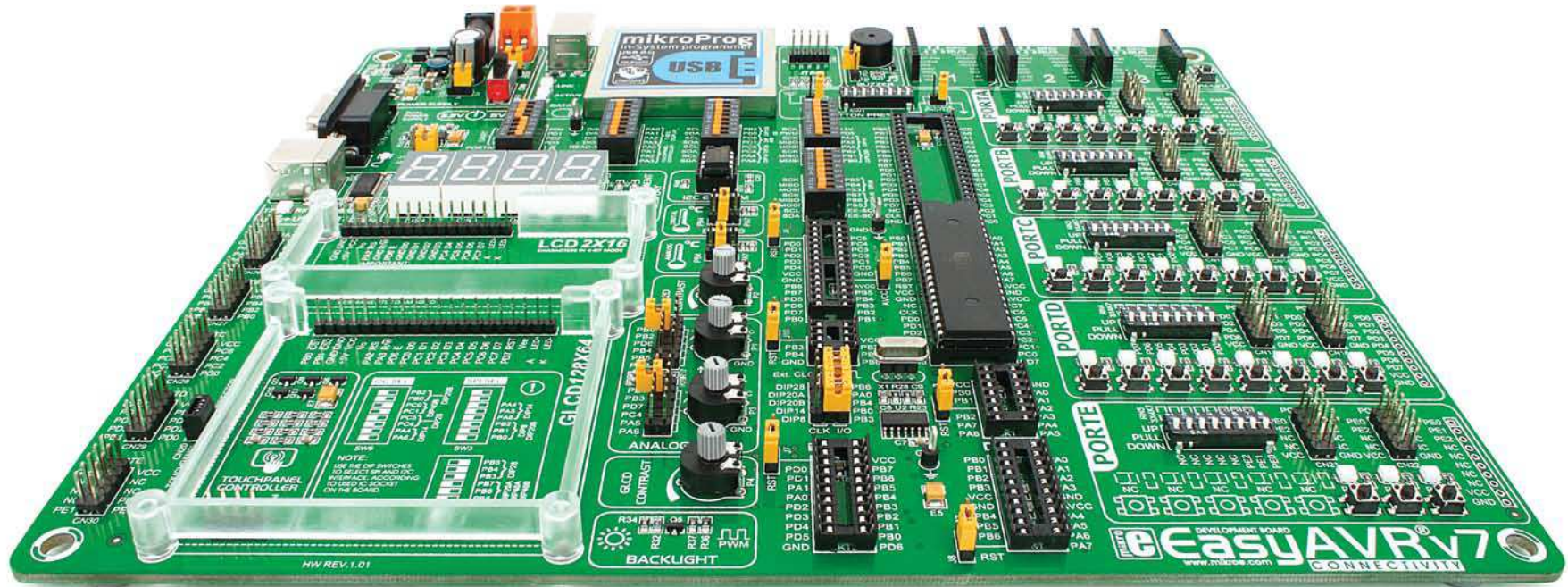
Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China



EasyAVR™ v7



65

microcontrollers supported
The ultimate AVR® board



Supports 3.3V and 5V devices
Dual Power Supply



Easily add extra boards
mikroBUS™ sockets



Four connectors for each port
Amazing Connectivity



As fast as it gets
USB 2.0 programmer

To our valued customers

Making the best and the most comprehensive AVR® board in the world was our idea from the start. And this is by no means an easy task. Driven by passion for excellence we never look at how hard the job is, but what our users get when it's done. And EasyAVR® v7 brings a whole new perspective to AVR developers.

The 7th generation of the board brings many exciting new features. We hope that you will like it as much as we do.

Use it wisely and have fun!

A handwritten signature in blue ink, appearing to read 'Nebojsa Matic', with a stylized, cursive script.

Nebojsa Matic,
Owner and General Manager
of mikroElektronika

Table of contents

Introduction

Introduction	04
It's good to know	05

Power Supply

Dual power supply	06
-------------------------	----

Supported MCUs

Supported microcontrollers	08
----------------------------------	----

Programming

On-board programmer	10
Installing programmer drivers	12
Programming software	13

Communication

UART via RS-232	14
UART via USB	15

Connectivity

mikroBUS™ sockets	16
Input/Output Group	18

Displays

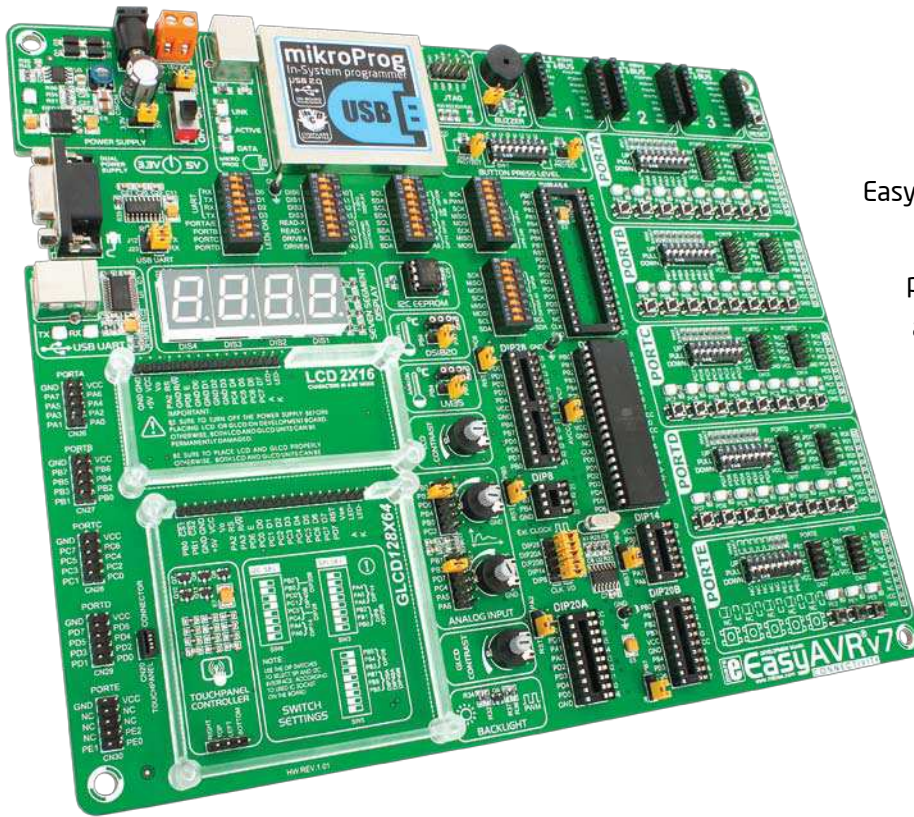
LCD 2x16 characters	20
GLCD 128x64.	21
Touch panel controller	22
4 digit 7-seg display	23

Modules

DS1820 - Digital Temperature Sensor	24
LM35 - Analog Temperature Sensor	25
ADC inputs	26
I2C EEPROM	27
Piezo Buzzer	28
Additional GNDs	29

What's next

What's Next?	30
--------------------	----



Introduction

EasyAVR™ is your old friend. It has been here for you for several generations.

Today it has many users among students, hobbyists, enthusiasts and professionals. We asked ourselves what else could be done to make such a great board even greater. As a result some brilliant changes have been made. We focused all of our creativity and knowledge on making a revolutionary new design different from any previous version of the board. We have gone through the process of fine tuning the board performance, and used 4-layer PCB to achieve maximum efficiency. We now present you with the new version of EasyAVR - a powerful, well organized board with on-board programmer - ready to be your strong ally in development.

EasyAVR™ development Team

Four Connectors for each port
Amazing connectivity

EasyAVR™ v7 is all about connectivity. Having four different connectors for each port, you can connect accessory boards, sensors and your custom electronics easier than ever before.



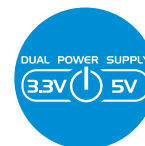
Everything is already here
mikroProg™ on board

Powerful on-board mikroProg™ programmer can program about 65 AVR® microcontrollers from Atmel®. You will need it, whether you are a professional or a beginner.



3.3V and 5V power supply
Dual Power Supply

EasyAVR™ v7 is among few development boards which support both 3.3V and 5V microcontrollers. This feature greatly increases the number of supported MCUs. It's like having two boards instead of one!



For easier connections
mikroBUS™ support

Just plug in your Click™ board, and it's ready to work. We picked up a set of the most useful pins you need for development and made a pinout standard you will enjoy using.



It's good to know

ATmega32 is the default microcontroller!

Atmel® AVR® 8-bit ATmega32 in DIP is the default chip of EasyAVR™ v7. It has up to **16 MIPS** operation, **32K bytes** of Flash program memory, **2K byte** of internal SRAM memory and **1K byte** of EEPROM memory. It has 40-pin with 32 General purpose **I/O pins**, 10-bit Analog-to-Digital converter with up to 8 channels (**ADC**), two 8-bit and one 16-bit timers/counters (with separate prescalers, compare mode and capture mode), four PWM channels, programmable serial **USART**, Master/slave **SPI** and two-wire **I²C** interface, internal calibrated 8 MHz RC and 32 KHz RTC oscillator. It also has JTAG interface for programming/debugging.

- Great choice for both beginners and professionals
- Rich with modules
- Enough RAM and Flash
- Comes with examples for mikroC, mikroBasic and mikroPascal compilers



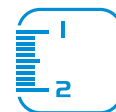
System Specification



power supply
7-12V AC or 9-15V DC
or via USB cable (5V DC)



power consumption
~80mA when all peripheral
modules are disconnected



board dimensions
266 x 220mm (10.47 x 8.66 inch)



weight
~438g (0.966 lbs)

Package contains



1 Damage resistant protective box



2 EasyAVR™ v7 board in antistatic bag



3 USB cable

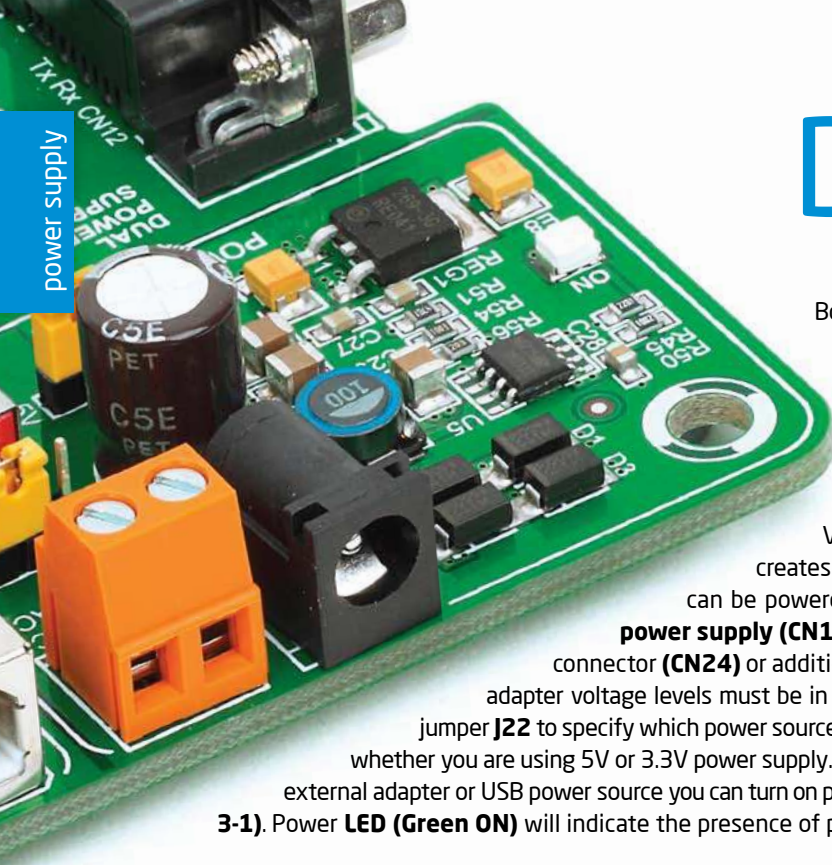


4 User Manuals and Board schematic



5 DVD with examples and documentation

Dual power supply



Board contains switching power supply that creates stable voltage and current levels necessary for powering each part of the board. Power supply section contains two power regulators: **ST1S10**, which generates VCC-5V, and **MC33269DT3.3** which creates VCC-3.3V power supply. The board can be powered in three different ways: with **USB power supply (CN1)**, using external adapters via adapter connector (**CN24**) or additional screw terminals (**CN25**). External adapter voltage levels must be in range of **9-15V DC or 7-12V AC**. Use jumper **J22** to specify which power source you are using and jumper **J5** to specify whether you are using 5V or 3.3V power supply. Upon providing the power using either external adapter or USB power source you can turn on power supply by using **SWITCH 1 (Figure 3-1)**. Power **LED (Green ON)** will indicate the presence of power supply.

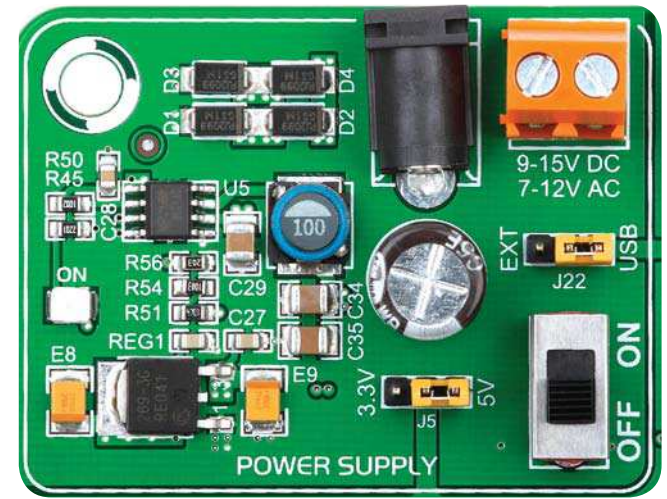


Figure 3-1: Dual power supply unit of EasyAVR™ v7

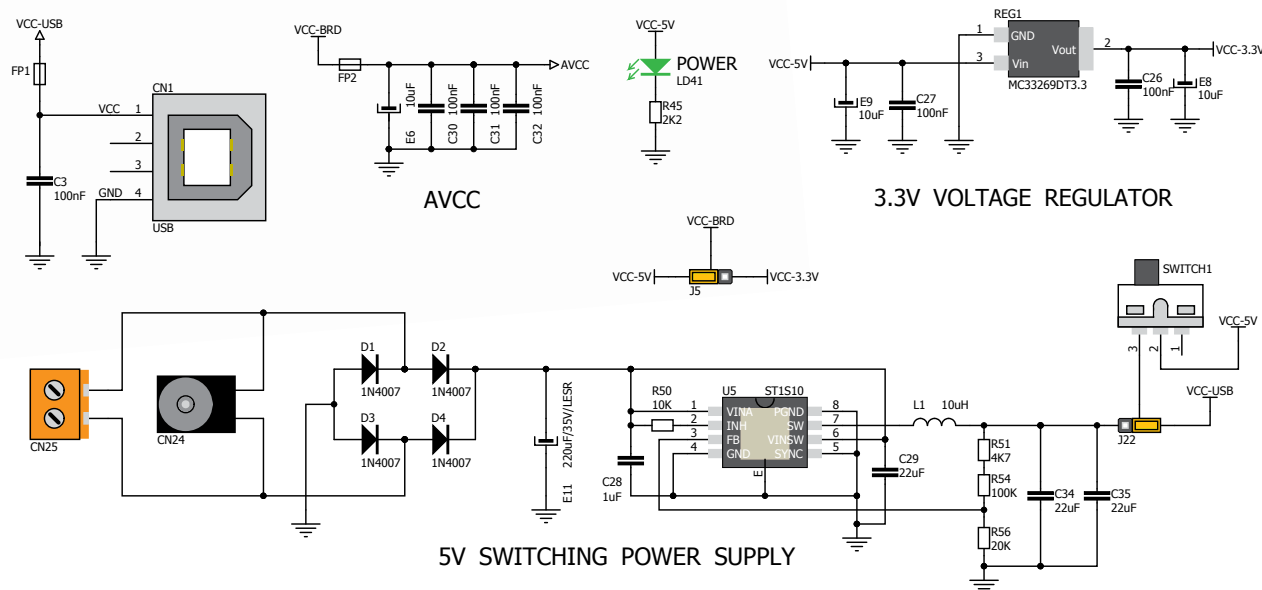


Figure 3-2: Dual power supply unit schematic

DUAL POWER SUPPLY



EasyAVR™ v7 development board supports both 3.3V and 5V power supply on a single board. This feature enables you to use wide range of peripheral boards.

Power supply:

via DC connector or screw terminals (7V to 12V AC or 9V to 15V DC), or via USB cable (5V DC)

Power capacity:

up to 500mA with USB, and up to 1500mA with external power supply

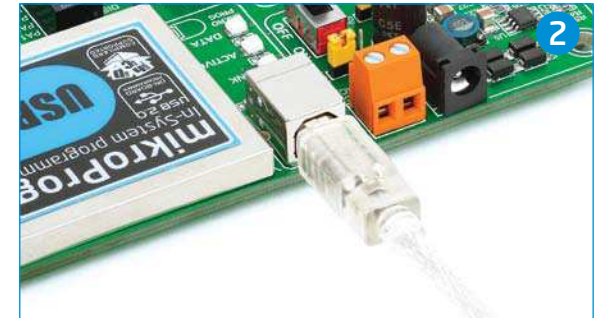
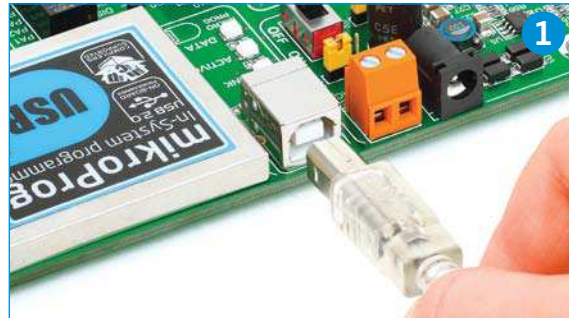
How to power the board?

1. With USB cable



Set J22 jumper to USB position

To power the board with USB cable, place jumper **J22** in USB position and place jumper **J5** in 5V or 3.3V position. You can then plug in the USB cable as shown on images **1** and **2**, and turn the power switch ON.

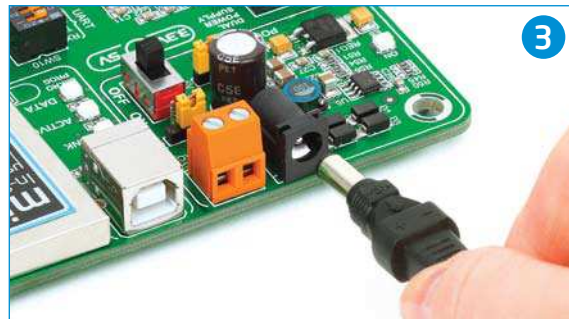


2. Using adapter



Set J22 jumper to EXT position

To power the board via adapter connector, place jumper **J22** in EXT position, and place jumper **J5** in 5V or 3.3V position. You can then plug in the adapter cable as shown on images **3** and **4**, and turn the power switch ON.

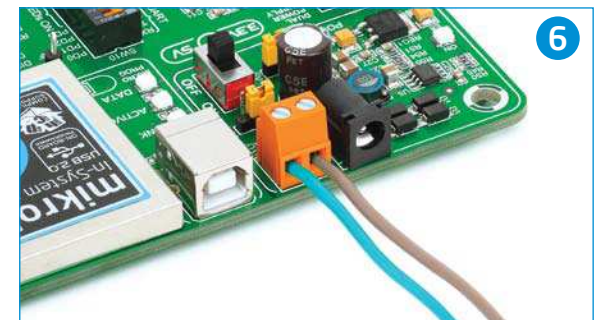
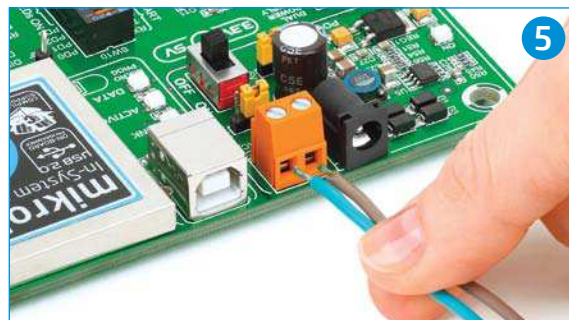


3. With laboratory power supply



Set J22 jumper to EXT position

To power the board using screw terminals, place jumper **J22** in EXT position, and place jumper **J5** in 5V or 3.3V position. You can then screw-on the cables in the screw terminals as shown on images **5** and **6**, and turn the power switch ON.



Supported microcontrollers

Microcontroller sockets

The board contains seven DIP sockets: DIP40A, DIP40B, DIP28, DIP20A, DIP20B, DIP14, DIP8 (Figure 4-1). With dual power supply and smart on-board mikroProg, the board is capable of programming about 65 different microcontrollers from Atmel® AVR family. There are two DIP40 and DIP20 sockets for AVR® microcontrollers provided on the board. Which of these sockets you will use depends solely on the pinout of the microcontroller in use. The **EasyAVR™ v7** development system comes with the **ATmega32** microcontroller in a **DIP40** package.

AREF jumper

If you use **DIP28** microcontroller socket with jumper **J11** you can set PC7 pin either as AVCC or I/O pin. Jumper **J11** is placed in the AVCC position by default.

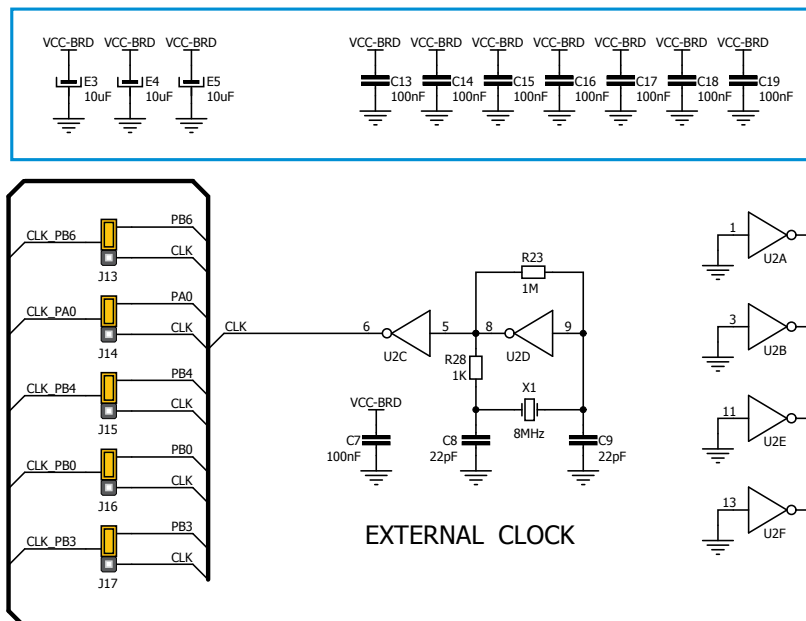
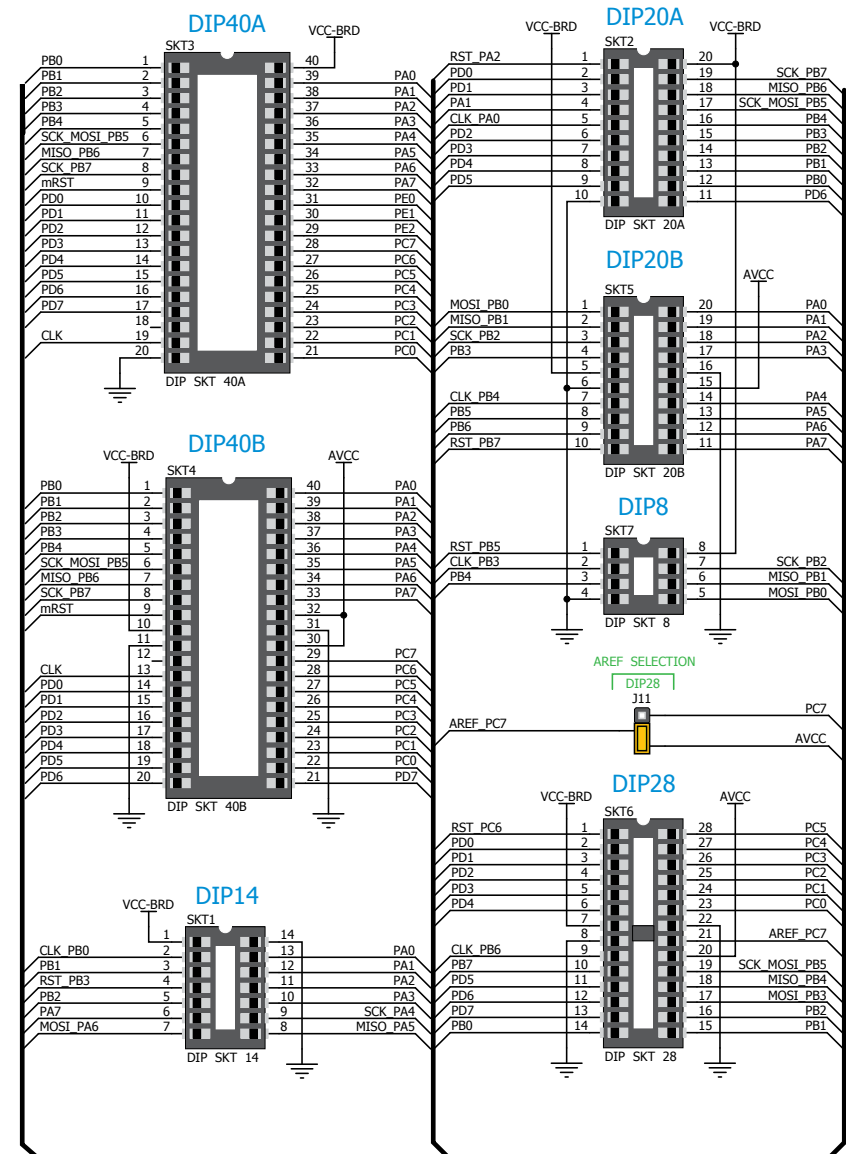


Figure 4-1: Schematic of on-board DIP sockets, quartz-crystal oscillator and decoupling capacitors



How to properly place your microcontroller into the DIP socket?

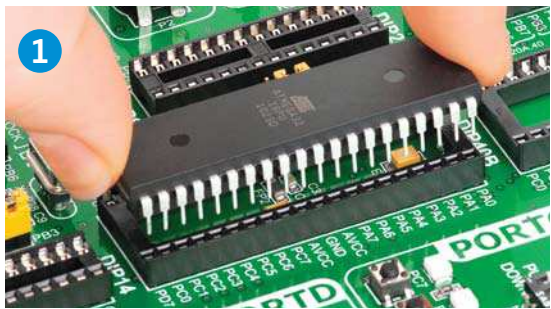


Figure 4-2: Place both ends of microcontroller on the socket so the pins are aligned correctly

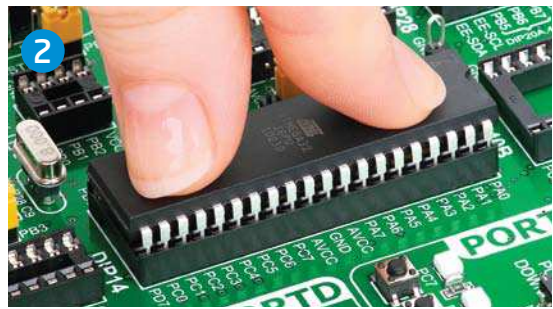


Figure 4-3: with both fingers, evenly distribute the force and press the chip into the socket.

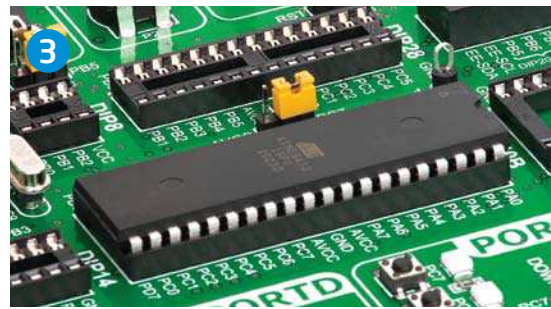


Figure 4-4: Properly placed microcontroller will have equally leveled pins.

Before you plug the microcontroller into the appropriate socket, make sure that the **power supply is turned off**. Images above show how to correctly plug a microcontroller. First make sure that

a half circular cut in the microcontroller DIP packaging matches the cut in the DIP socket. Place both ends of the microcontroller into the socket as shown in **Figure 4-2**. Then put the microcontroller slowly down until

all the pins match the socket as shown in **Figure 4-3**. Check again if everything is placed correctly and press the microcontroller until it is completely plugged into the socket as shown in **Figure 4-4**.

IMPORTANT: *Only one microcontroller may be plugged into the development board at the same time.*

Using external clock

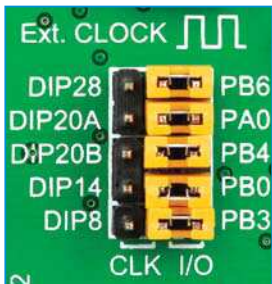


Figure 4-5: Position of the jumpers for DIP40x

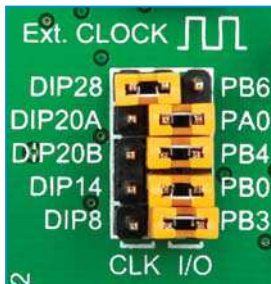


Figure 4-6: Position of the jumpers for DIP28



Figure 4-7: Position of the jumpers for DIP20A

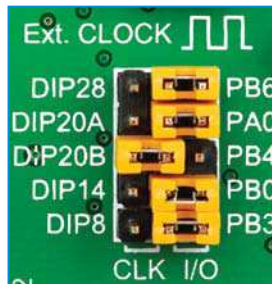


Figure 4-8: Position of the jumpers for DIP20B



Figure 4-9: Position of the jumpers for DIP14



Figure 4-10: Position of the jumpers for DIP8

AVR microcontrollers can use either built-in (internal) or quartz-crystal external clock for the purpose of providing clock signal source. The **EasyAVR™ v7** contains one quartz-crystal socket for all microcontroller sockets. **DIP40A** and **DIP40B** sockets are directly connected to quartz-crystal socket unlike **DIP28**, **DIP20A**, **DIP20B**, **DIP14** and **DIP8** sockets which are connected to it through the pin jumpers (J13-J17). The value of the

quartz-crystal depends on the maximum clock frequency allowed and your application as well. You can always replace the default 8MHz crystal with another one. If you want to use microcontrollers in **DIP40A** and **DIP40B** packages, it is necessary to place jumpers J13-J17 into the I/O position, **Figure 4-5**. The position of jumpers for **DIP28**, **DIP20A**, **DIP20B**, **DIP14** and **DIP8** sockets is shown in **Figure 4-6** to **Figure 4-10**.

On-board programmer

What is mikroProg™?

mikroProg™ is a fast USB 2.0 programmer. Smart engineering allows mikroProg™ to support about 65 AVR® microcontrollers from Atmel® in a single programmer! Outstanding performance and easy operation are among its top features.

How do I start?

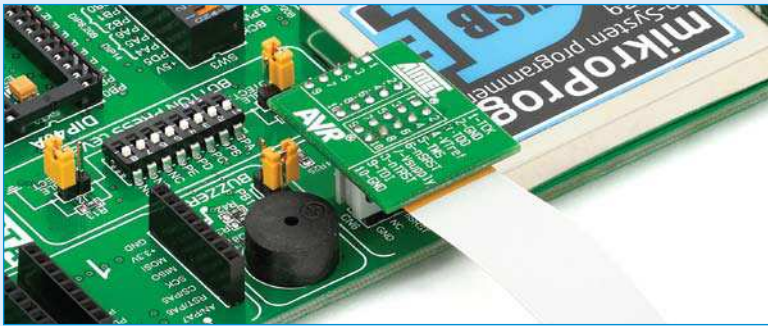
In order to start using mikroProg™ and program your microcontroller, you just have to follow two simple steps:

1. Install the necessary software

- Install USB drivers
- Install AVRFlash software

2. Power up the board, and you are ready to go.

- Plug in the programmer USB cable
- LINK LED should light up.



Programming with JTAGICE mkII programmer

EasyAVR™ v7 is equipped with JTAG connector compatible with Atmel® AVR® JTAGICE mkII external programmer. You can use either the on-board mikroProg™ programmer or external programmer at the time. Place your programmer cable onto connector **CN6**, as shown in image.

Why so many LEDs?

Three LEDs indicate specific programmer operation. **Link** LED lights up when USB link is established with your PC, **Active** LED lights up when the programmer is active. **Data** is on when data is being transferred between the programmer and PC software (compiler or AVRFlash).

RESET or I/O?

Reset selection jumpers **J6**, **J7**, **J8**, **J9**, and **J10** are used to set PC6, PA2, PB7, PB3 or PB5 pin either as RST or I/O pin, respectively. Each of the pins belongs to different socket. For example, if you want to use PC6 pin (DIP28 socket) as I/O pin place jumper into the PC6 position. Other jumpers must be in RST position. For DIP40A and DIP40B sockets, all the reset selection jumpers must be in RST position.

NOTE:

Before using the programmer, make sure that reset selection jumpers J6, J7, J8, J9, and J10 are placed into the mRST position for all sockets.

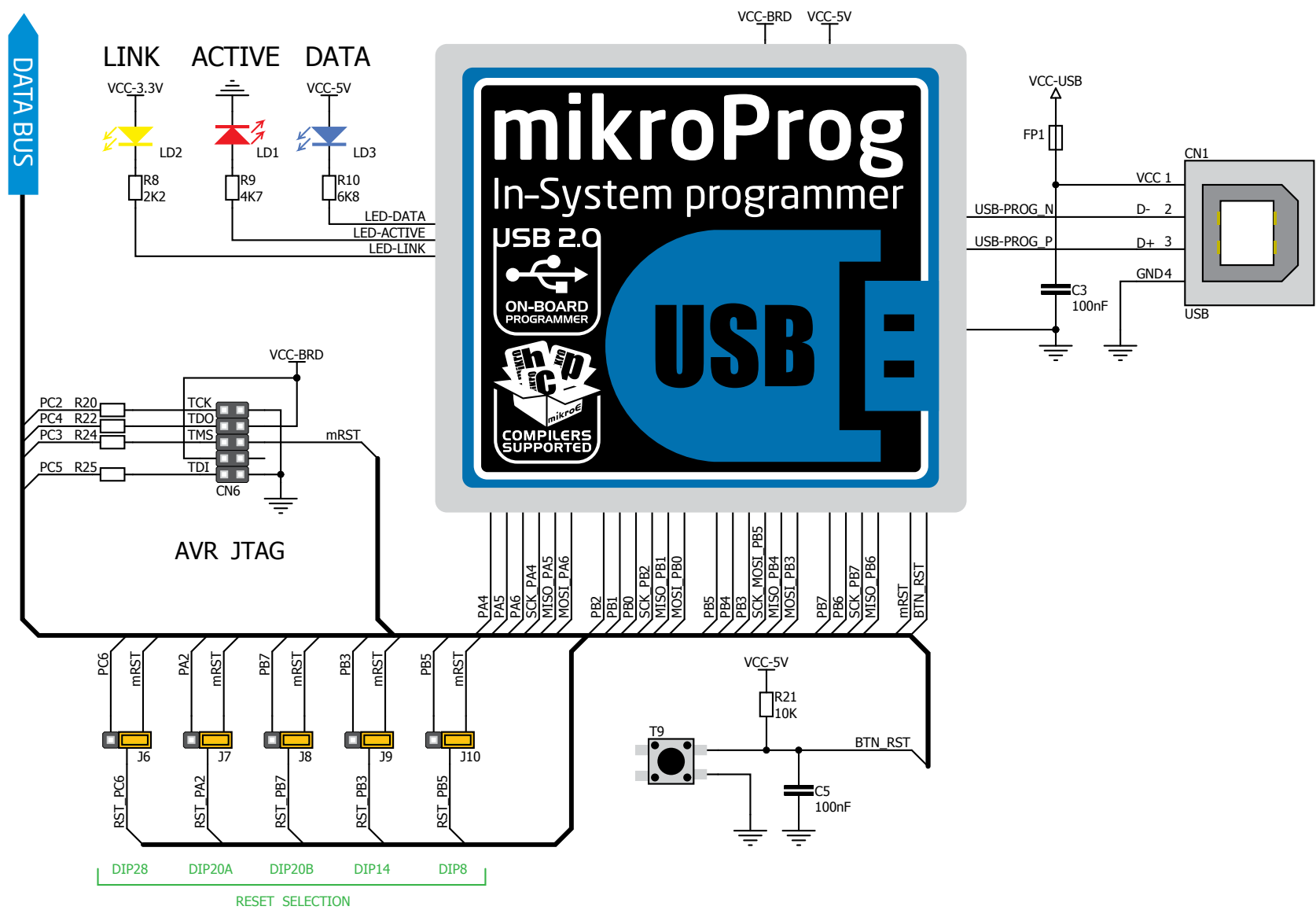


Figure 5-1: mikroProg™ block schematic

Installing programmer drivers

On-board mikroProg™ requires drivers in order to work. Drivers are located on the **Product DVD** that you received with the EasyAVR™ v7 package:



*DVD://download/eng/software/
development-tools/avr/avrprog2/
avrprog2_drivers_v200.zip*

When you locate the drivers, please extract files from the ZIP archive. Folder with extracted files contains sub folders with drivers for different operating systems. Depending on which operating system you use, choose adequate folder and open it.



Windows 7 32 bit,
Vista 32bit, Win
2008 32 bit



Windows 7 64 bit,
Vista 64bit, Win
2008 64 bit



Win 2000, XP,
2003 32-bit



Win 98



Win XP, 2003
64-bit

In the opened folder you should be able to locate the driver setup file. Double click on setup file to begin installation of the programmer drivers.

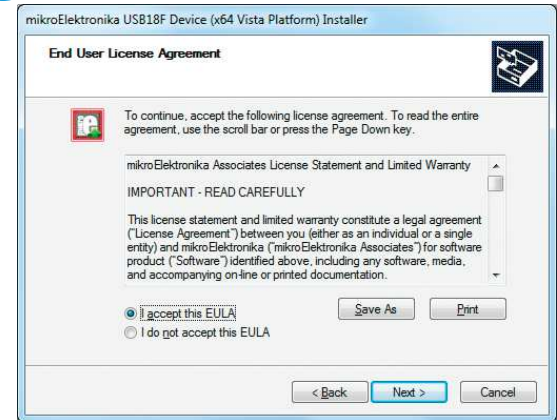


USB18PRG-Vista-
x64



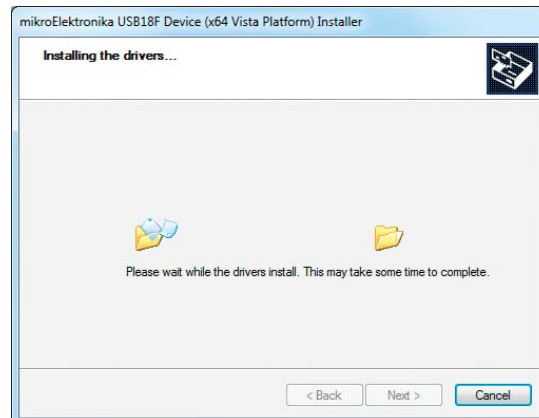
Step 1 - Start Installation

Welcome screen of the installation. Just click on **Next** button to proceed.



Step 2 - Accept EULA

Carefully read **End User License Agreement**. If you agree with it, click **Next** to proceed.



Step 3 - Installing drivers

Drivers are installed automatically in a matter of seconds.



Step 4 - Finish installation

You will be informed if the drivers are installed correctly. Click on **Finish** button to end installation process.

Programming software

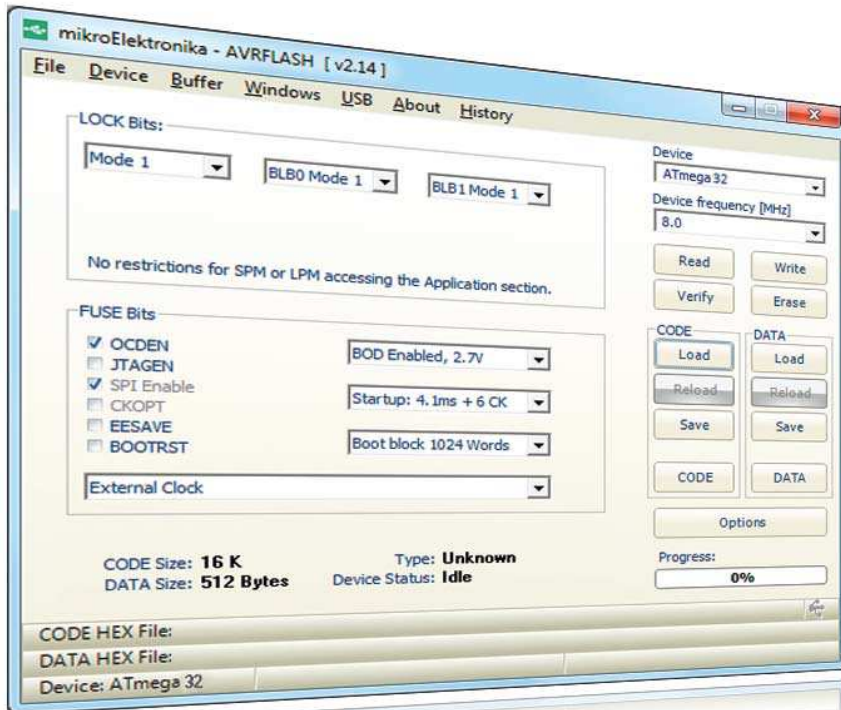
AVRFlash software

On-board **mikroProg™** programmer requires special programming software called **AVRFlash**. This software is used for programming AVR® microcontrollers from Atmel®. Software has intuitive interface and **SingleClick™** programming technology. To begin, first locate the installation archive on the Product DVD:



DVD://download/eng/software/development-tools/avr/avr-prog2/avrprog2_programmer_v214.zip

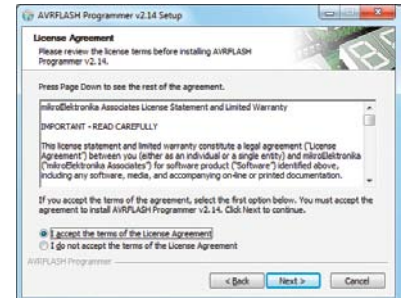
After downloading, extract the package and double click the executable setup file, to start installation.



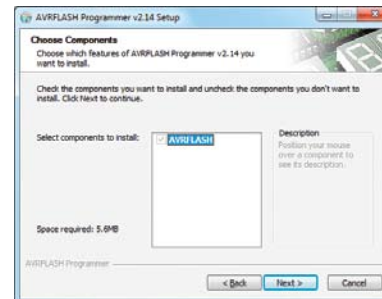
Installation wizard - 6 simple steps



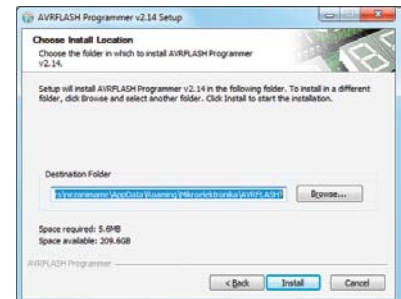
Step 1 - Start Installation



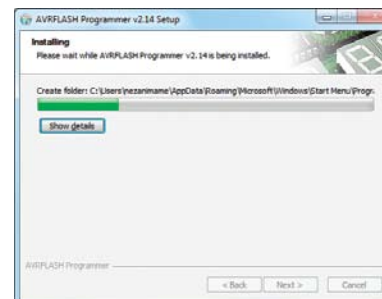
Step 2 - Accept EULA and continue



Step 3 - Click **Next** button



Step 4 - Choose destination folder



Step 5 - Installation in progress

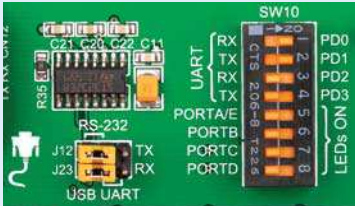


Step 6 - Finish Installation

UART via RS-232



Enabling RS-232



In order to enable RS-232 communication, you must set **J23** and **J12** jumpers in the **RS-232** position, and enable desired **RX** and **TX** lines via **SW10** DIP switch. For example, if you want to enable RS-232 connection on UART module of the default **ATmega32** chip, you should enable **SW10.1** (RD0) and **SW10.2** (RD1) lines.

The **UART** (universal asynchronous receiver/transmitter) is one of the most common ways of exchanging data between the MCU and peripheral components. It is a serial protocol with separate transmit and receive lines, and can be used for full-duplex communication. Both sides must be initialized with the same baud rate, otherwise the data will not be received correctly.

RS-232 serial communication is performed through a 9-pin SUB-D connector and the microcontroller UART module. In order to enable this communication, it is necessary to establish a connection between **RX** and **TX** lines on SUB-D connector and the same pins on the target microcontroller using DIP switches. Since RS-232 communication voltage levels are different than microcontroller logic levels, it is necessary to use a RS-232 Transceiver circuit, such as **MAX3232** as shown on **Figure 6-1**.

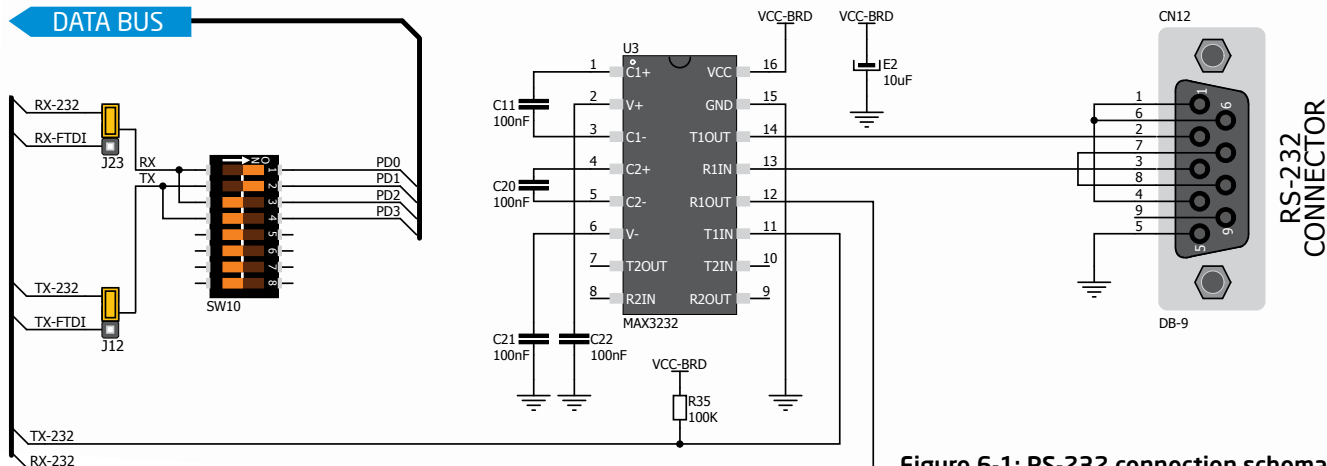
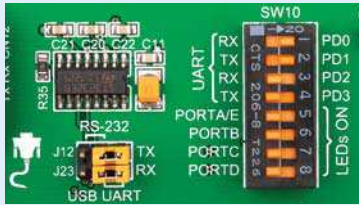


Figure 6-1: RS-232 connection schematic

UART via USB



Enabling USB-UART



In order to enable USB-UART communication, you must set **J23** and **J12** jumpers in the **USB-UART** position, and enable desired **RX** and **TX** lines via **SW10** DIP switch. For example, if you want to enable USB-UART connection on UART module of the default **ATmega32** chip, you should enable **SW10.1** (RD0) and **SW10.2** (RD1) lines.

Modern PC computers, laptops and notebooks are no longer equipped with RS-232 connectors and UART controllers. They are nowadays replaced with USB connectors and USB controllers. Still, certain technology enables UART communication to be done via USB connection. **FT232RL** from FTDI® convert UART signals to the appropriate USB standard. In order to use USB-UART module on EasyAVR™ v7, you must first install FTDI drivers on your computer. Drivers can be found on **Product DVD**:



DVD://download/eng/software/development-tools/universal/ftdi/vcp_drivers.zip

USB-UART communication is being done through a FT232RL controller, USB connector (**CN4**), and microcontroller UART module. To establish this connection, you must put **J23** and **J12** jumpers in the **USB-UART** position, and connect **RX** and **TX** lines to the appropriate pins of the microcontroller. This connection is done using DIP switch **SW10**.

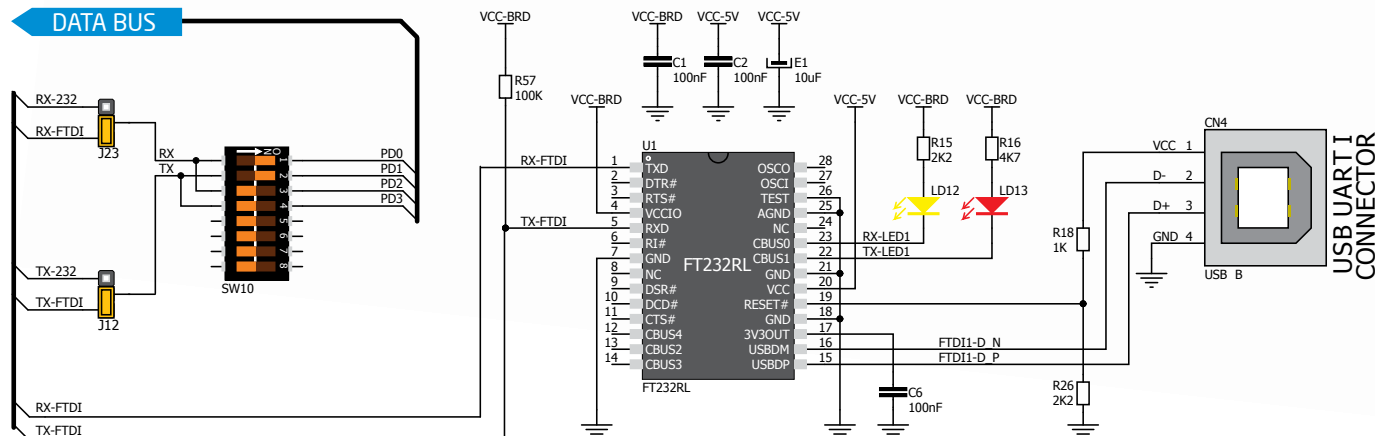
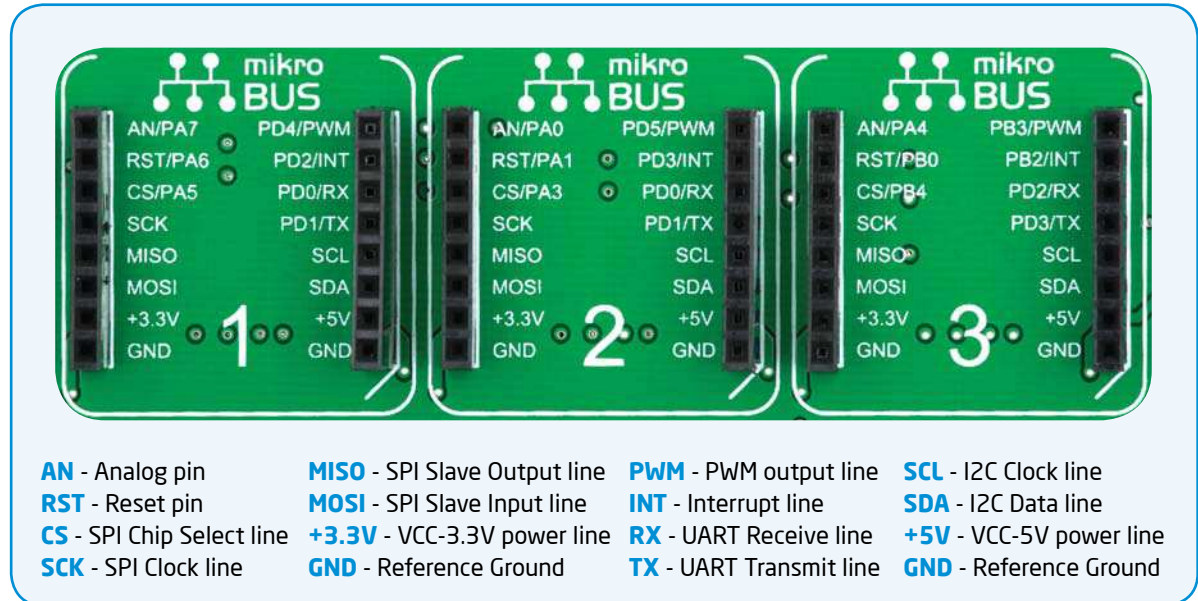


Figure 7-1: USB-UART connection schematic

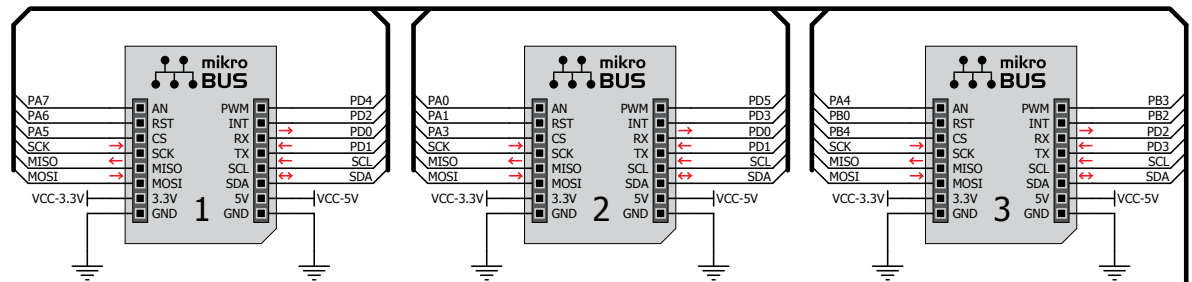
mikroBUS™ sockets

Easier connectivity and simple configuration are imperative in modern electronic devices. Success of the USB standard comes from its simplicity of usage and high and reliable data transfer rates. As we in mikroElektronika see it, Plug-and-Play devices with minimum settings are the future in embedded world too. This is why our engineers have come up with a simple, but brilliant pinout with lines that most of today's accessory boards require, which almost completely eliminates the need of additional hardware settings. We called this new standard the **mikroBUS™**. EasyAVR™ v7 is the first development board in the world to support mikroBUS™ with three on-board sockets. As you can see, there are no additional DIP switches, or jumper selections. Everything is already routed to the most appropriate pins of the microcontroller sockets.



mikroBUS™ host connector

Each mikroBUS™ host connector consists of two 1x8 female headers containing pins that are most likely to be used in the target accessory board. There are three groups of communication pins: **SPI**, **UART** and **I²C** communication. There are also single pins for **PWM**, **Interrupt**, **Analog input**, **Reset** and **Chip Select**. Pinout contains two power groups: **+5V** and **GND** on one header and **+3.3V** and **GND** on the other 1x8 header.



SPI and I²C selection

Various microcontroller sockets have different pins for SPI and I²C interface. In order to connect the SPI and I²C pins of the mikroBUS™ with the desired socket, you have to change appropriate **SW3**, **SW5** or **SW6** DIP switches to ON position, **Figure 8-1**.

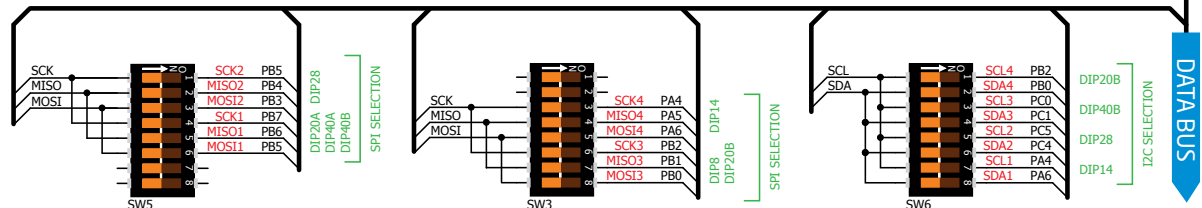


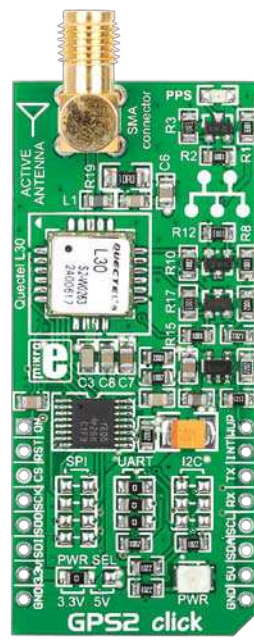
Figure 8-1: mikroBUS™ socket with DIP switches schematic



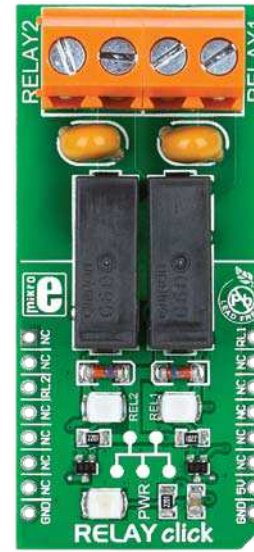
WiFi PLUS click™



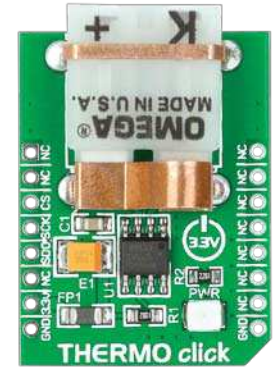
GSM2 click™



GPS2 click™



RELAY click™



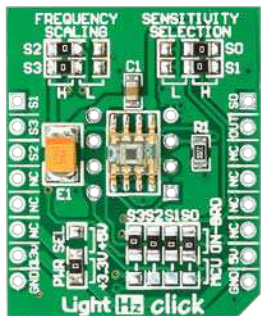
THERMO click™

Click Boards™ are plug-n-play!

mikroElektronika portfolio of over 200 accessory boards is now enriched by an additional set of mikroBUS™ compatible **Click Boards™**. Almost each month several new Click boards™ are released. It is our intention to provide the community with as much of these boards as possible, so you will be able to expand your EasyAVR™ v7 with additional functionality with literally zero

hardware configuration. Just plug and play. Visit the Click boards™ webpage for the complete list of available boards:

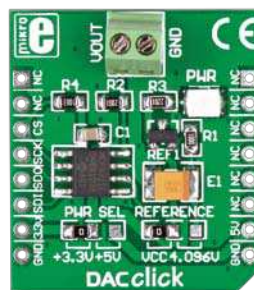
<http://www.mikroe.com/click/>



LightHz click™



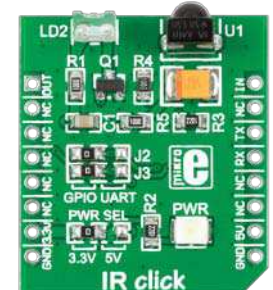
microSD click™



DAC click™



DIGIPOT click™



IR click™

Input/Output Group

One of the most distinctive features of EasyAVR™ v7 are its Input/Output PORT groups. They add so much to the connectivity potential of the board.

Everything is grouped together

PORT **headers**, PORT **buttons** and PORT **LEDs** are next to each other, and grouped together. It makes development easier, and the entire EasyAVR™ v7 cleaner and well organized. We have also provided an **additional PORT headers** on the left side of the board, so you can access any pin you want from both sides of the board.

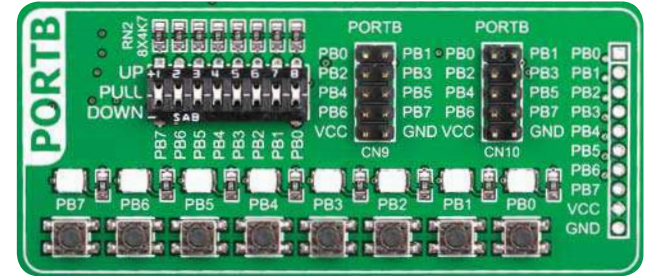
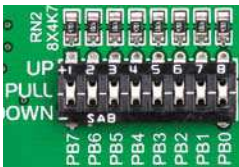


Figure 9-1: I/O group contains PORT headers, tri-state pull up/down DIP switch, buttons and LEDs all in one place

Tri-state pull-up/down DIP switches

Tri-state DIP switches, like **SW1** on **Figure 9-2**, are used to enable 4K7 pull-up or pull-down resistor on any desired port pin. Each of these switches has three states:



1. **middle position** disables both pull-up and pull-down feature from the PORT pin
2. **up position** connects the resistor in pull-up state to the selected pin
3. **down position** connects the resistor in pull-down state to the selected PORT pin.

Figure 9-2: Tri-state DIP switch on PORTB

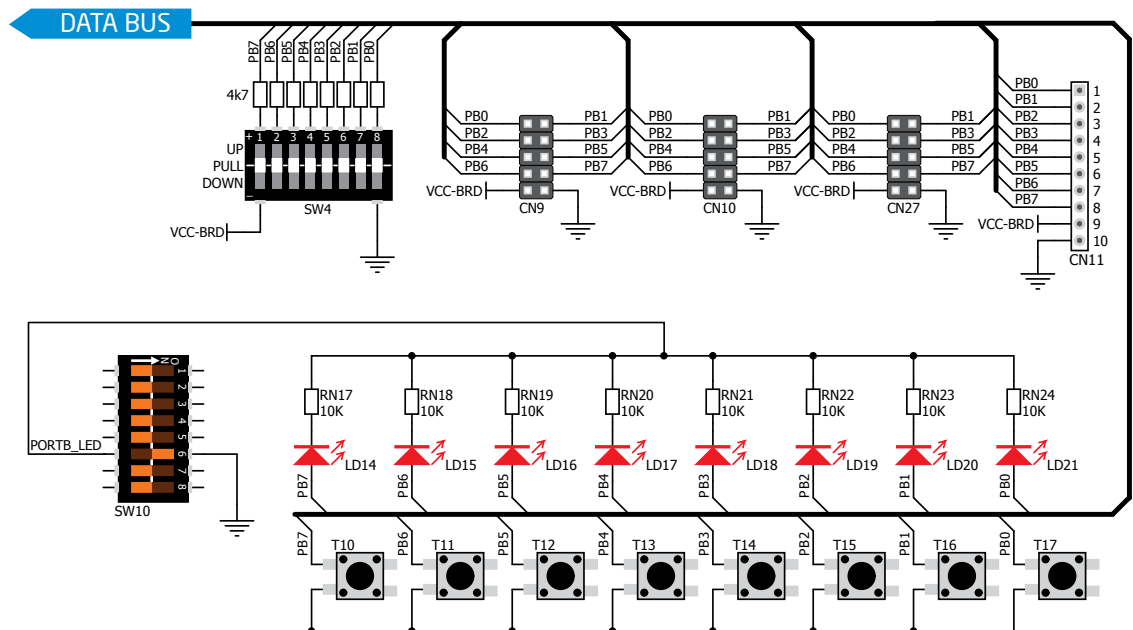
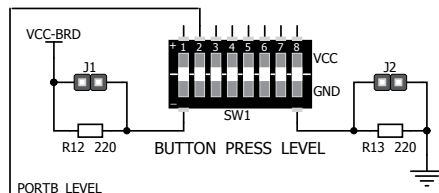
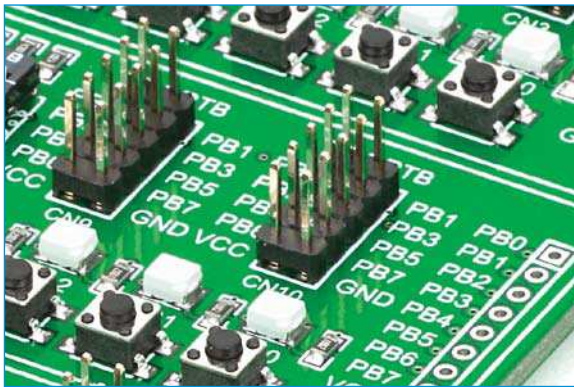


Figure 9-3: Schematic of the single I/O group connected to microcontroller PORTB



Headers

With enhanced connectivity as one of the key features of EasyAVR v7, we have provided **three connection headers for each PORT**. I/O PORT group contains two male IDC10 headers (like **CN9** and **CN10** on **Figure 9-3**). These headers are all compatible with over 70 mikroElektronika accessory boards, and enable simple connection. There is **one more IDC10 header** available on the left side of the board, next to the section with displays. I/O PORT group also contains 1x10 connection pad (like **CN11** on **Figure 9-3**) which can be used for connecting mikroElektronika PROTO boards, or custom user boards.

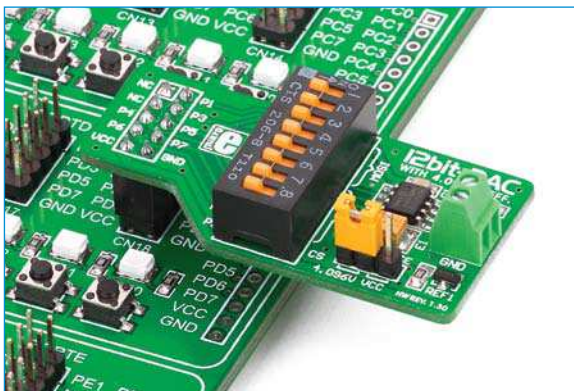
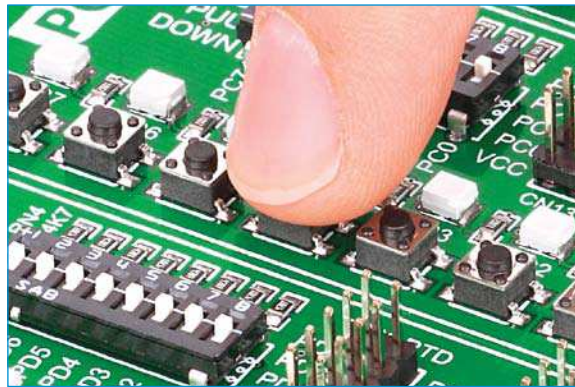


Figure 9-4: IDC10 male headers enable easy connection with mikroElektronika accessory boards



Buttons

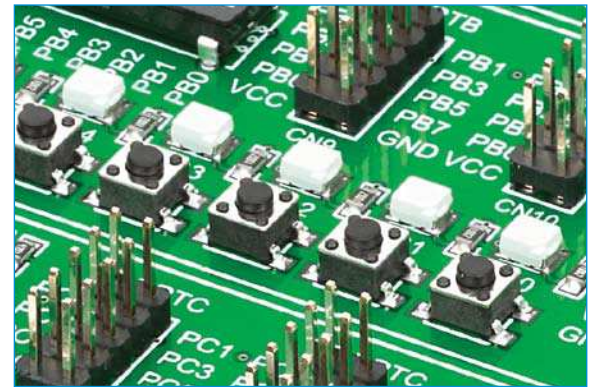
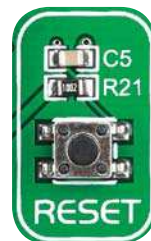


Figure 9-5: Button press level DIP switch (tri-state)

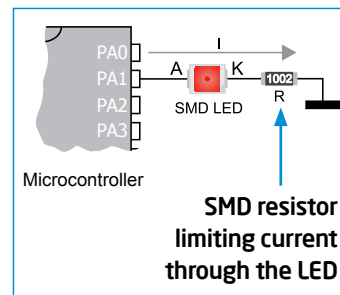
The logic state of all microcontroller digital inputs may be changed using **push buttons**. Tri-state DIP switch **SW1** is available for selecting which logic state will be applied to corresponding MCU pin when button is pressed, for each I/O port separately. If you, for example, place **SW1.2** in **VCC** position, then pressing any of push buttons in PORTB I/O group will apply logic one to the appropriate microcontroller pin. The same goes for **GND**. If DIP switch is in the middle position neither of two logic states will be applied to the appropriate microcontroller pin. You can disable pin protection 220ohm resistors by placing jumpers **J1** and **J2**, which will connect your push buttons directly to VCC or GND. Be aware that doing so you may accidentally damage MCU in case of wrong usage.

Reset Button

In the far upper right section of the board, there is a **RESET button**, which can be used to manually reset the microcontroller.



LEDs



LED (Light-Emitting Diode) is a highly efficient electronic light source. When connecting LEDs, it is necessary to place a current limiting resistor in series so that LEDs are provided with the

current value specified by the manufacturer. The current varies from 0.2mA to 20mA, depending on the type of the LED and the manufacturer. The EasyAVR™ v7 board uses low-current LEDs with typical current consumption of 0.2mA or 0.3mA, depending of VCC voltage selection. Board contains 35 LEDs which can be used for visual indication of the logic state on PORT pins. An active LED indicates that a logic high (1) is present on the pin. In order to enable PORT LEDs, it is necessary to enable the corresponding DIP switches on **SW10** (**Figure 9-6**).

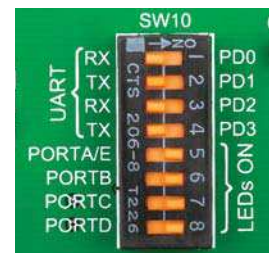


Figure 9-6: SW10.5 through SW10.8 switches are used to enable PORT LEDs

LCD 2x16 characters

Liquid Crystal Displays or LCDs are cheap and popular way of representing information to the end user of some electronic device. Character LCDs can be used to represent standard and custom characters in the predefined number of fields. EasyAVR™ v7 provides the connector and the necessary interface for supporting **2x16 character LCDs** in 4-bit mode. This type of display has two rows consisted of 16 character fields. Each field is a 7x5 pixel matrix. Communication with the display module is done through **CN8** display connector. Board is fitted with uniquely designed plastic display distancer, which allows the LCD module to perfectly and firmly fit into place.

IMPORTANT: *Make sure to turn off the power supply before placing LCD onto the board. Otherwise your display can be permanently damaged.*



Figure 10-1: On-board LCD 2x16 display connector

Connector pinout explained

GND and VCC - Display power supply lines

V0 - LCD contrast level from potentiometer P2

RS - Register Select Signal line

E - Display Enable line

R/W - Determines whether display is in Read or Write mode. It's always connected to GND, leaving the display in Write mode all the time.

D0-D3 - Display is supported in 4-bit data mode, so lower half of the data byte interface is connected to GND.

D4-D7 - Upper half of the data byte

LED+ - Connection with the back-light LED anode

LED- - Connection with the back-light LED cathode

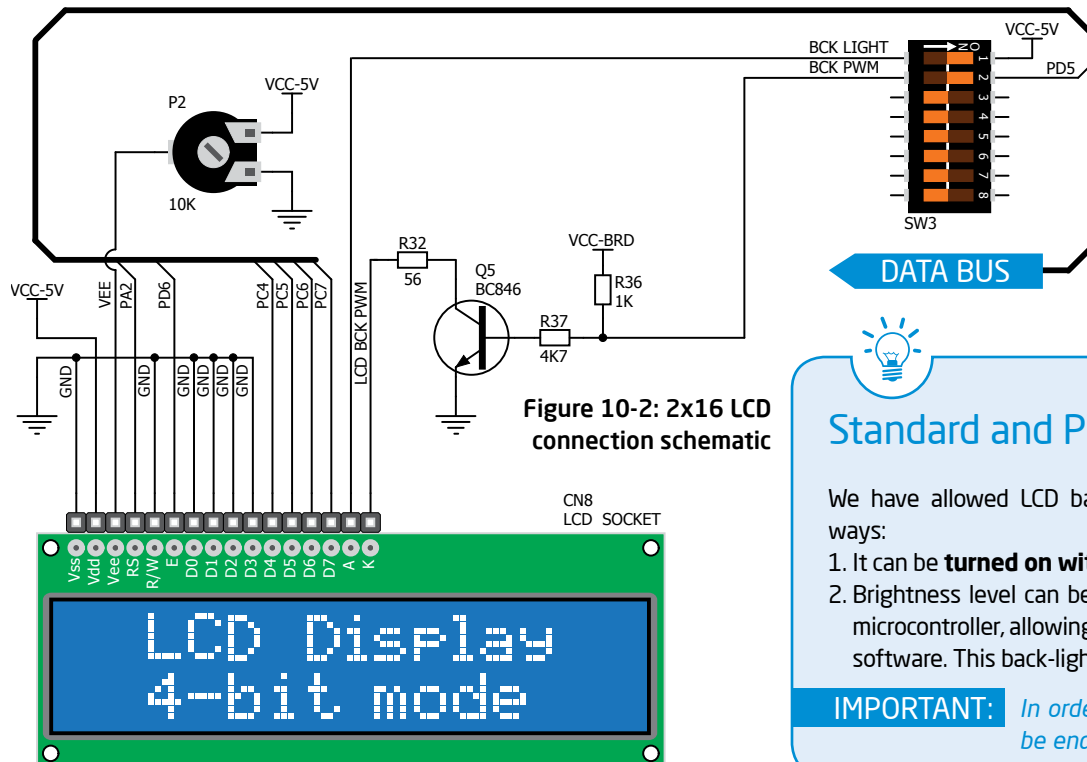


Figure 10-2: 2x16 LCD connection schematic

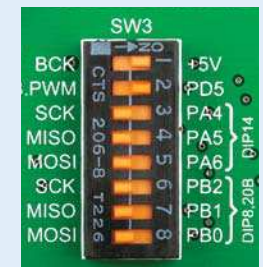


Standard and PWM-driven back-light

We have allowed LCD back-light to be enabled in two different ways:

1. It can be **turned on with full brightness** using **SW3.1** switch.
2. Brightness level can be determined **with PWM signal** from the microcontroller, allowing you to write custom back-light controlling software. This back-light mode is enabled with **SW3.2** switch.

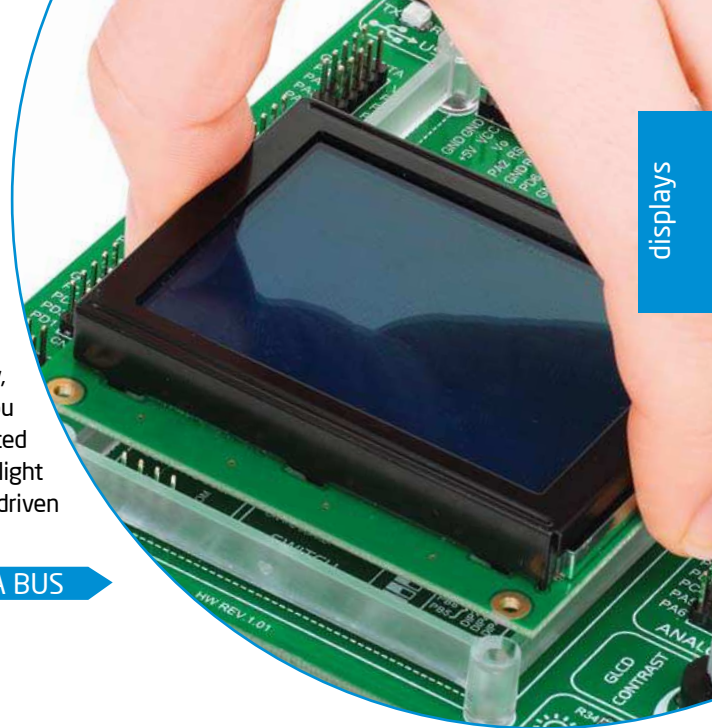
IMPORTANT: *In order to use PWM back-light both **SW3.1** and **SW3.2** switches must be enabled at the same time.*



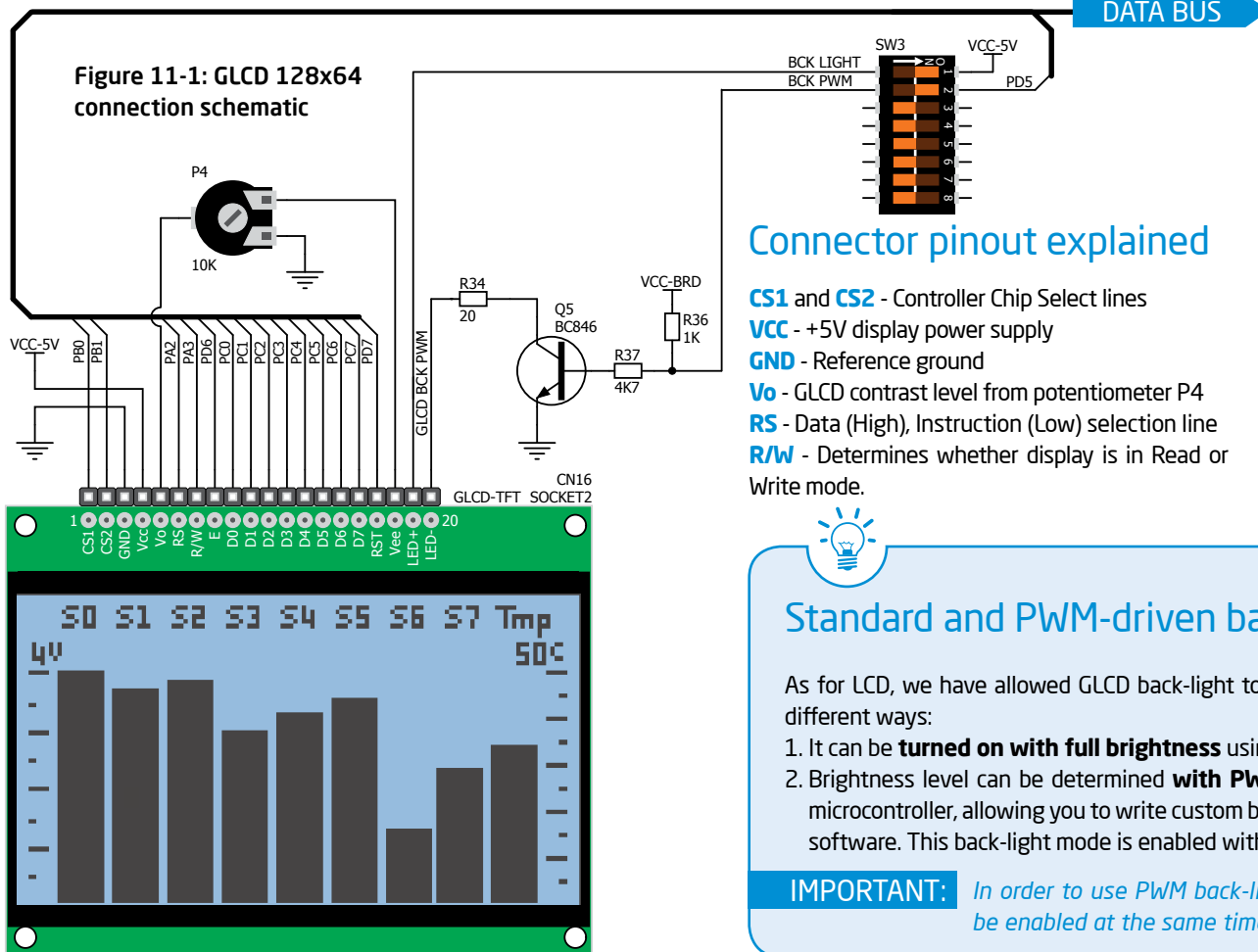
GLCD 128x64

Graphical Liquid Crystal Displays, or GLCDs are used to display monochromatic graphical content, such as text, images, human-machine interfaces and other content. EasyAVR™ v7 provides the connector and necessary interface for supporting GLCD with resolution of 128x64 pixels, driven by the KS108 or compatible display controller. Communication with the display module is done through **CN16** display connector. Board is fitted with uniquely designed plastic display distancer, which allows the GLCD module to perfectly and firmly fit into place.

Display connector is routed to PB0, PB1, PA2, PA3, PD6, PD7 (control lines) and PORTC (data lines) of the microcontroller sockets. Since the same ports are used by 2x16 character LCD display, you cannot use both displays simultaneously. You can control the display contrast using dedicated potentiometer **P4**. Full brightness display back light can be enabled with **SW3.1** switch, and PWM-driven back light with **SW3.2** switch.



displays



Touch panel controller

Touch panel is a glass panel whose surface is covered with two layers of resistive material. When the screen is pressed, the outer layer is pushed onto the inner layer and appropriate controllers can measure that pressure and pinpoint its location. This is how touch panels can be used as an input devices.

EasyAVR™ v7 is equipped with touch panel controller and connector for **4-wire resistive touch panels**. It can very accurately register pressure at a specific point, representing the touch coordinates in the form of analog voltages, which can then be easily converted to X and Y values. Touch panel comes as a part of display.

Correctly placing the touch panel cable into the connector

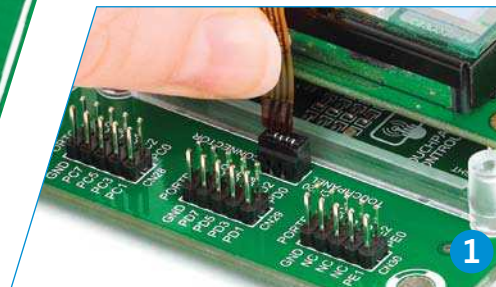


Figure 12-1: Put Touch panel flat cable in the connector

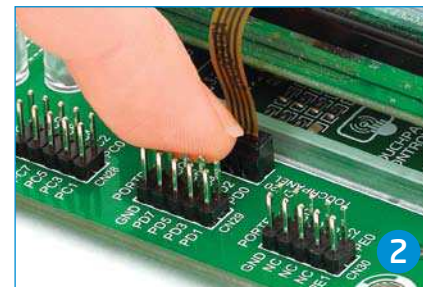


Figure 12-2: Use a tip of your finger to push it inside

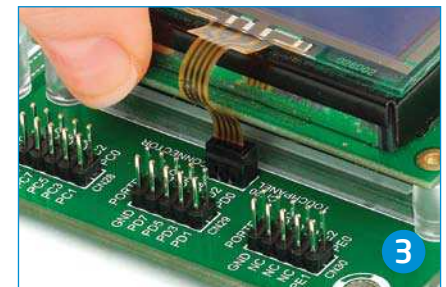


Figure 12-3: Now place GLCD with Touch panel into GLCD socket

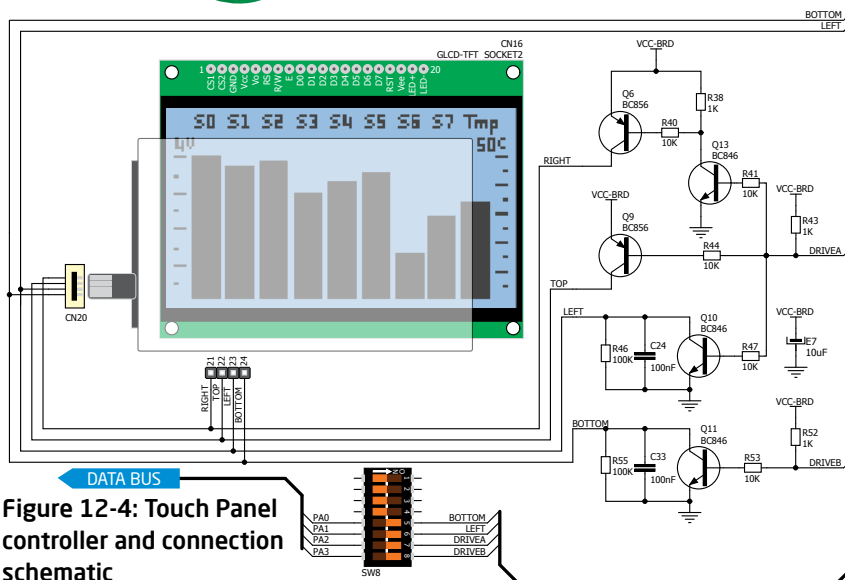


Figure 12-4: Touch Panel controller and connection schematic

Enabling Touch panel

Touch panel is enabled using **SW8.5**, **SW8.6**, **SW8.7** and **SW8.8** switches. They connect **READ-X** and **READ-Y** lines of the touch panel with **PA0** and **PA1** analog inputs, and **DRIVEA** and **DRIVEB** with **PA2** and **PA3** digital outputs on microcontroller sockets. Make sure to disconnect other peripherals, LEDs and additional pull-up or pull-down resistors from the interface lines in order not to interfere with signal/data integrity.

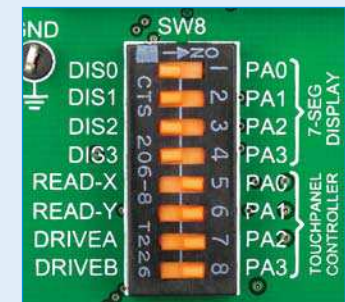


Figure 12-5: Turn on switches 5 through 8 on SW8 to enable Touch panel controller

4 digit 7-seg display

One seven segment digit consist of 7+1 LEDs which are arranged in a specific formation which can be used to represent digits from 0 to 9 and even some letters. One additional LED is used for marking the decimal dot, in case you want to write a decimal point in the desired segment. EasyAVR™ v7 contains four of these digits put together to form 4-digit 7-segment display. Driving such a display is done using **multiplexing techniques**. Data lines are shared between segments, and therefore the same segment LEDs in each digit are connected in parallel. Each digit has its **unique digit select line**,

which is used to enable the digit to which the data is currently being sent. By multiplexing data through all four segments fast enough, you create an illusion that all four segments are in operation simultaneously.

This is possible because human eye has a slower reaction time than the mention changes. This way you can represent numbers in decimal or hexadecimal form. Eight data lines that are common for all the digits are connected to PORTC, and digit select lines are connected to PA0-PA3 lines on the microcontroller sockets.

Enabling the display

To enable digit select lines for the 4-digit 7-segment display you have to turn on **SW8.1, SW8.2, SW8.3** and **SW8.4** switches. Digit select lines are connected to **PA0 - PA3** pins on the microcontroller sockets, while data lines are connected to **PC0 - PC7** pins. Make sure to disconnect other peripherals from the interface lines in order not to interfere with signal/data integrity.

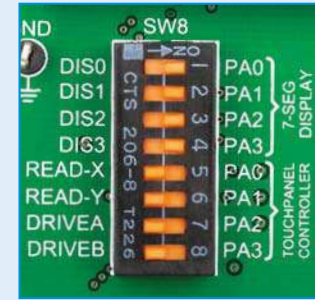


Figure 13-1: Turn on switches 1 through 4 on SW8 to enable 4-digit 7-seg display

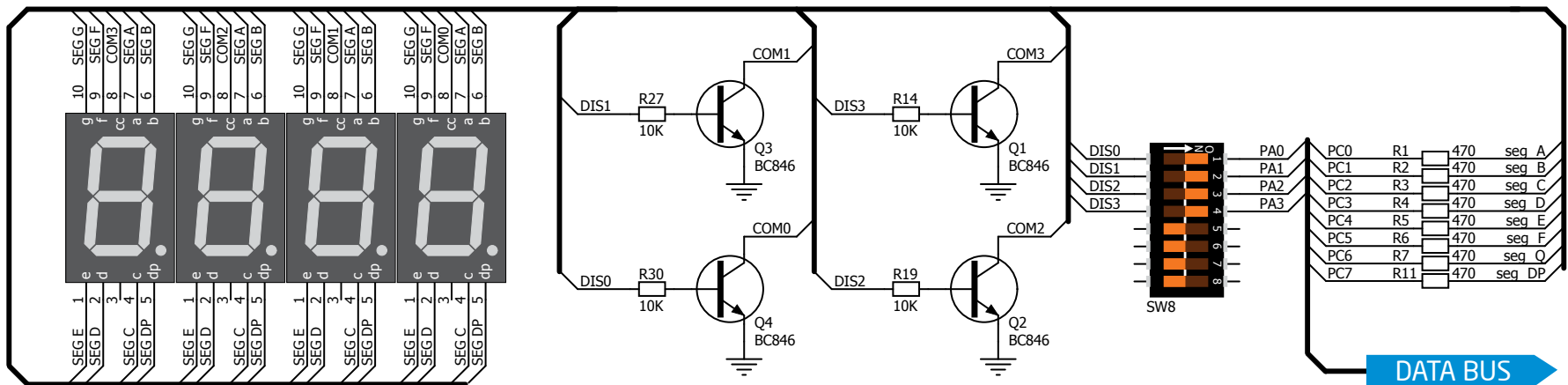


Figure 13-2: 4-digit 7-segment display schematic

DS1820 - Digital Temperature Sensor

DS1820 is a digital temperature sensor that uses **1-wire® interface** for its operation. It is capable of measuring temperatures within the range of -55 to 128°C, and provides $\pm 0.5^\circ\text{C}$ accuracy for temperatures within the range of -10 to 85°C. It requires 3V to 5.5V power supply for stable operation. It takes maximum

of 750ms for the DS1820 to calculate temperature with 9-bit resolution. **1-wire® serial communication** enables data to be transferred over a single communication line, while the process itself is under the control of the master microcontroller. The advantage of such communication is that only one microcontroller pin is used. Multiple

sensors can be connected on the same line. All slave devices by default have a unique ID code, which enables the master device to easily identify all devices sharing the same interface. EasyAVR™ v7 provides a separate socket (**TS1**) for the DS1820. Communication line with the microcontroller is connected via jumper **J18**.



Enabling DS1820 Sensor

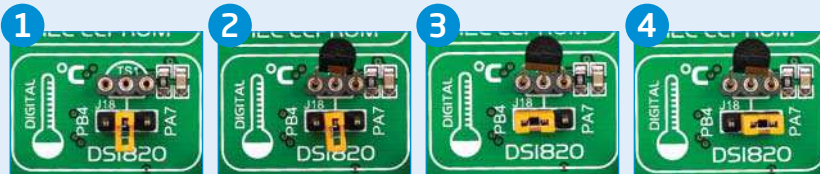


Figure 14-1:
DS1820 not connected

Figure 14-2:
DS1820 placed in socket

Figure 14-3:
DS1820 connected to PB4 pin

Figure 14-4:
DS1820 connected to PA7 pin

EasyAVR™ v7 enables you to establish 1-wire® communication between **DS1820** and the microcontroller via **PB4** or **PA7** microcontroller pins. The selection of either of those two lines is done using **J18** jumper. When placing the sensor in the socket make sure that half-circle on the board's silkscreen markings matches the rounded part of the DS1820 sensor. If you accidentally connect the sensor the other way, it may be permanently damaged. Make sure to disconnect other peripherals (except 1-wire), LEDs and additional pull-up or pull-down resistors from the interface lines in order not to interfere with signal/data integrity.

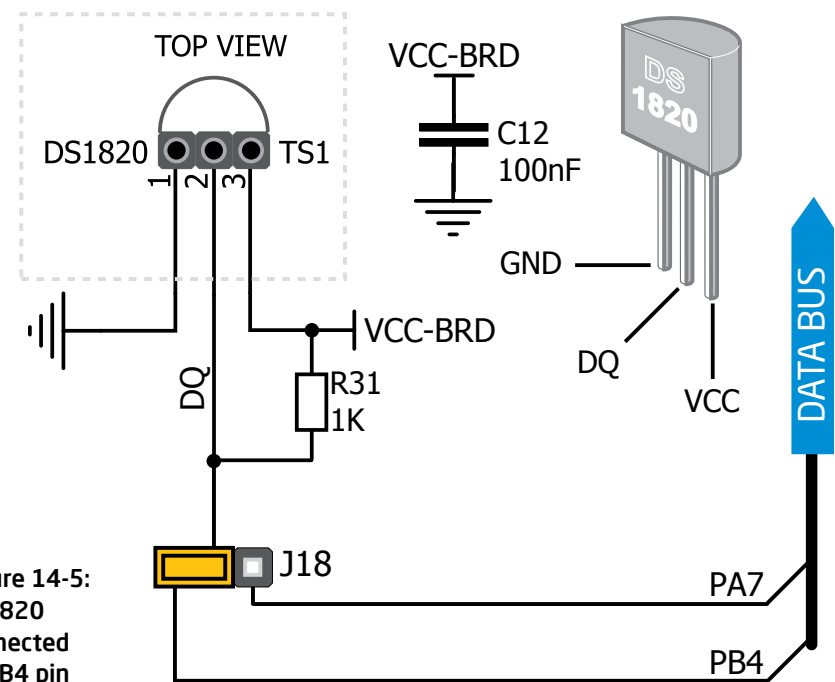


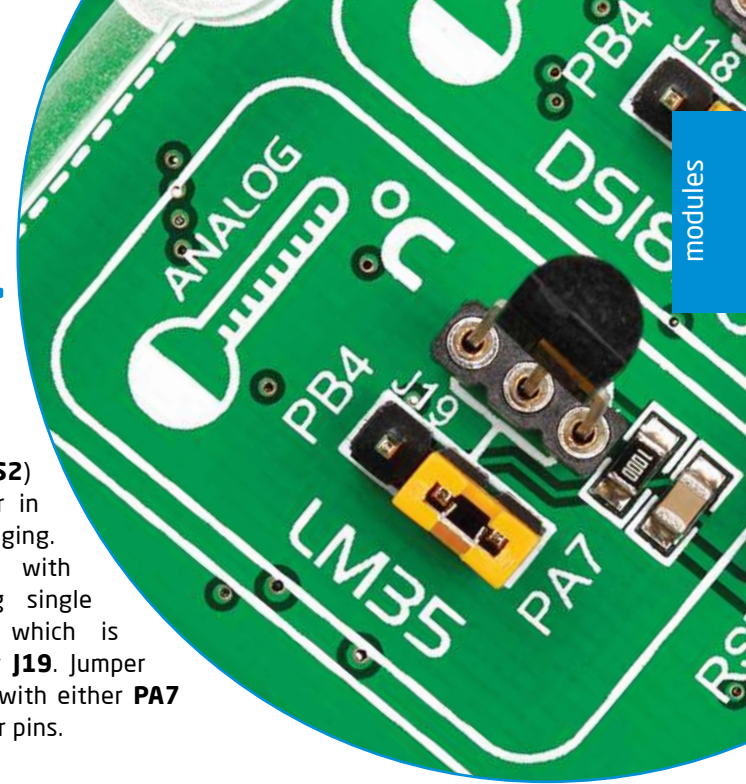
Figure 14-5:
DS1820 connected to PB4 pin

LM35 - Analog Temperature Sensor

The **LM35** is a low-cost precision integrated-circuit temperature sensor, whose output voltage is linearly proportional to the Celsius (Centigrade) temperature. The LM35 thus has an advantage over linear temperature sensors calibrated in ° Kelvin, as the user is not required to subtract a large constant voltage from its output to

obtain convenient Centigrade scaling. It has a linear $+10.0 \text{ mV}/^\circ\text{C}$ scale factor and less than $60 \mu\text{A}$ current drain. As it draws only $60 \mu\text{A}$ from its supply, it has very low self-heating, less than 0.1°C in still air. EasyAVR™ v7 enables you to get analog readings from the LM35 sensor in restricted temperature range from $+2^\circ\text{C}$ to $+150^\circ\text{C}$. Board provides

a separate socket (**TS2**) for the LM35 sensor in TO-92 plastic packaging. Readings are done with microcontroller using single analog input line, which is selected with jumper **J19**. Jumper connects the sensor with either **PA7** or **PB4** microcontroller pins.



Enabling LM35 Sensor

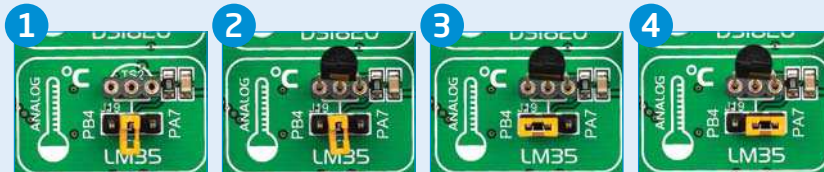


Figure 15-1:
LM35 not connected

Figure 15-2:
LM35 placed in socket

Figure 15-3:
LM35 connected to PB4 pin

Figure 15-4:
LM35 connected to PA7 pin

EasyAVR™ v7 enables you to get analog readings from the LM35 sensor using **PA7** or **PB4** microcontroller pins. The selection of either of those two lines is done using **J19** jumper. When placing the sensor in the socket make sure that half-circle on the board's silkscreen markings matches the rounded part of the LM35 sensor. If you accidentally connect the sensor the other way, it can be permanently damaged and you might need to replace it with another one. During the readings of the sensor, make sure that no other device uses the selected analog line, because it may interfere with the readings.

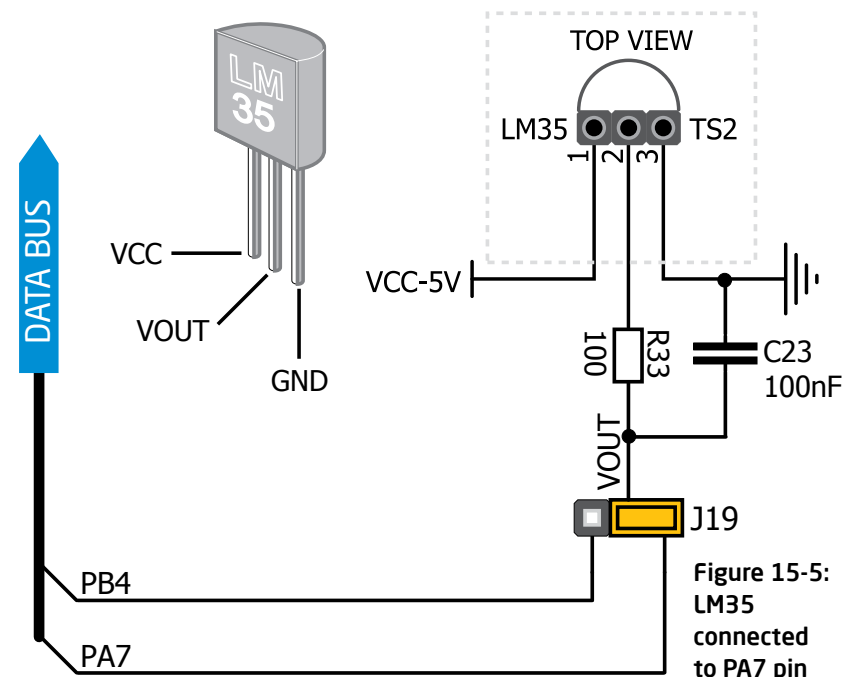


Figure 15-5:
LM35 connected to PA7 pin