Chipsmall

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

EasyPIC™ v7
for dsPIC30®
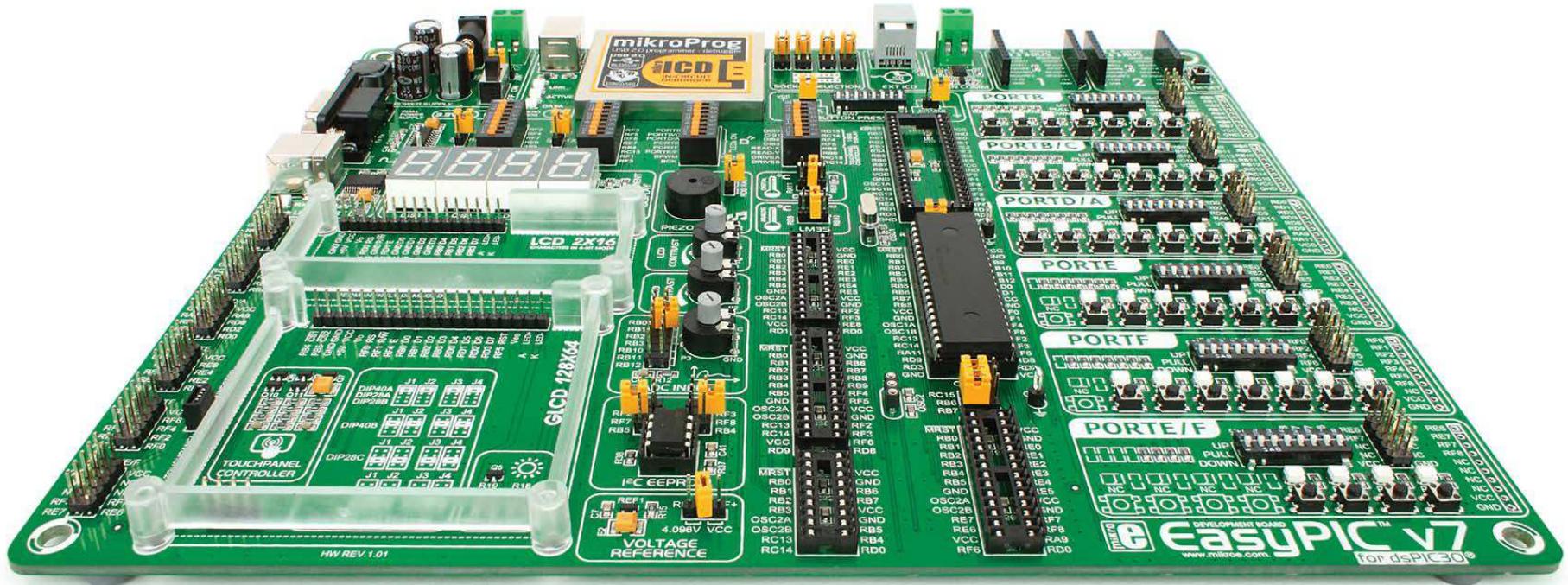
dsPIC30®

14 microcontrollers supported
dsPIC30® in DIP package

DUAL POWER SUPPLY
3.3V | 5V

Supports 3.3V and 5V devices
Dual Power Supply

mikroBUS

Easily add extra boards
mikroBUS™ sockets

connectivity

Four connectors for each port
Amazing Connectivity

Fast USB 2.0 programmer and
In-Circuit Debugger

MikroElektronika
DEVELOPMENT TOOLS | COMPILERS | BOOKS

To our valued customers

From the day one, we in MikroElektronika gave ourselves the highest possible goals in pursuit of excellence. That same day, the idea of EasyPIC™ development board was born. In its each and tiniest piece we had put all of our energy, creativity and sense of what's best for an engineer.

Today, we present you the 7th generation of our boards, which brings us some exciting new features. We hope that you will like it as much as we do.

Use it wisely and have fun!

Nebojsa Matic,
Owner and General Manager
of MikroElektronika

Table of contents

Introduction

Power Supply

Supported MCUs

Programming

Communication

Connectivity

Displays

Other Modules

What's Next

Introduction

dsPIC30F® microcontrollers are 16-bit high-performance digital signal controllers suitable for advanced motor control algorithms, digital power converters, speech and audio applications. We realized that benefit and start to develop a new system which is convenient for wide range of dsPIC30F® microcontroller family produced in DIP packages. We focused all of our creativity and knowledge into making a revolutionary new design which covers many internal modules. Finally, we present you the EasyPIC™ v7 for dsPIC30® board which is powerful, well organized, with on-board programmer and debugger. Board is suitable for students, hobbyists, enthusiasts and professionals. We hope you will enjoy it as much as we do.

EasyPIC™ v7 for dsPIC30® development Team
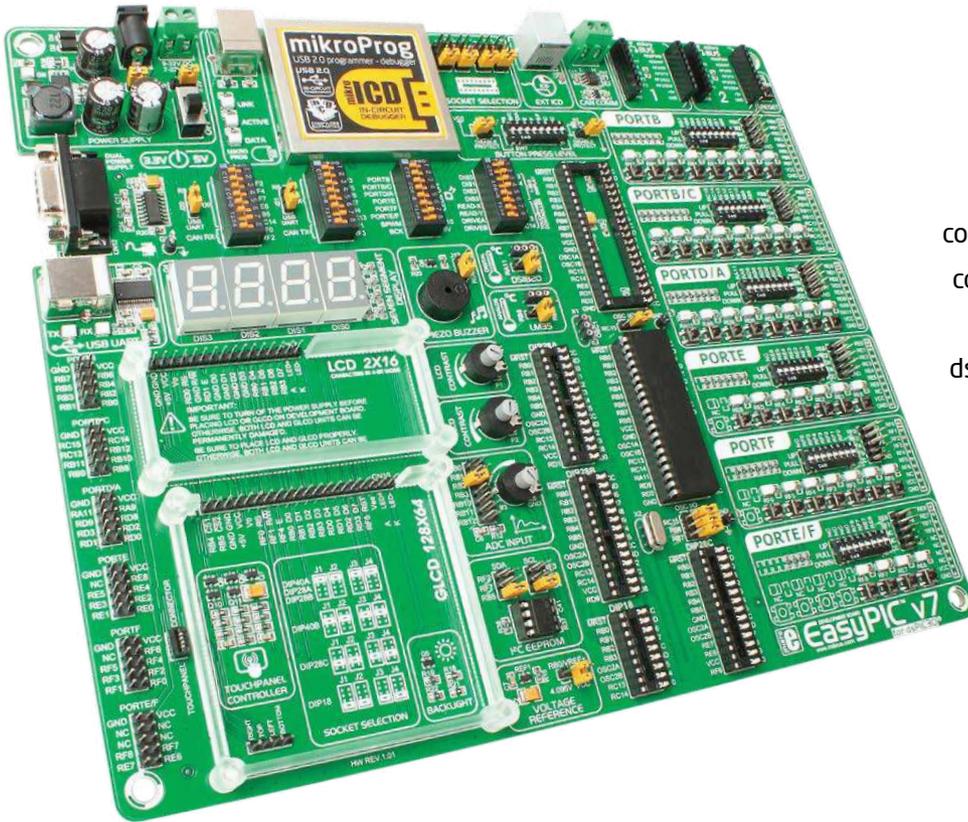
Four Connectors for each port
Amazing connectivity

EasyPIC™ v7 for dsPIC30® is all about connectivity. Having three different connectors for each port, you can connect accessory boards, sensors and your custom electronics easier then ever before.

Everything is already here
mikroProg™ on board

Powerful on-board mikroProg™ programmer and In-Circuit debugger can program and debug 14 dsPIC30® microcontrollers. You will need it, whether you are a professional or a beginner.

3.3V and 5V power supply
Dual Power Supply

EasyPIC™ v7 for dsPIC30® is among few development boards which support both 3.3V and 5V power supply. This feature allows you to use the microcontrollers with both power supply. 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.

EasyPIC ™ v7
for dsPIC30®

It's good to know

dsPIC30F4013 is the default microcontroller!

dsPIC30F4013 is the default chip of EasyPIC™ v7 for dsPIC30®. It has **30 MIPS** operation, **48K bytes** of program memory, **2048 bytes** of SRAM memory, **1024 bytes** of EEPROM memory. It has 24-bit wide instructions and 16-bit wide data path, . It's loaded with great modules: 30 General purpose **I/O pins**, 12-bit Analog-to-Digital with up to 13 channels (**ADC**), 5x16-bit and 2x32-bit timers, 2x**UART**, 1x**SPI**, 1X**I²C**, 1x**CAN**, 16-bit compare/PWM output functions. It has two 40-bit wide accumulators with optional saturation logic, 17-bit x 17-bit single hardware fractional/integer cycle multiplier and all DSP instructions are single cycle.

- *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
7-23V AC or 9-32V DC
or via USB cable (5V DC)

power consumption
~99mA when all peripheral modules are disconnected

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

weight
~445g (0.981 lbs)

Package contains

1. Damage resistant protective box
2. EasyPIC™ v7 for dsPIC30® board in antistatic bag
3. USB cable
4. User Manuals and Board schematics

EasyPIC™ v7
for dsPIC30®
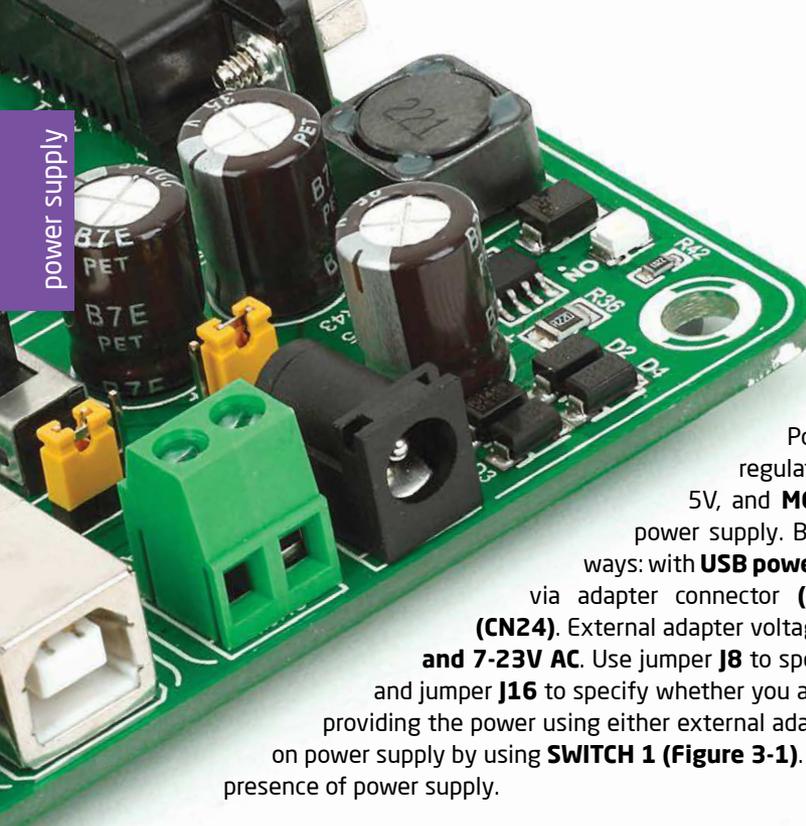
Dual power supply

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: **MC34063A**, which generates VCC-5V, and **MC33269DT3.3** which creates VCC-3.3V power supply. Board can be powered in three different ways: with **USB power supply (CN1)**, using external adapters via adapter connector **(CN25)** or additional screw terminals **(CN24)**. External adapter voltage levels must be in range of **9-32V DC and 7-23V AC**. Use jumper **J8** to specify which power source you are using, and jumper **J16** 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.
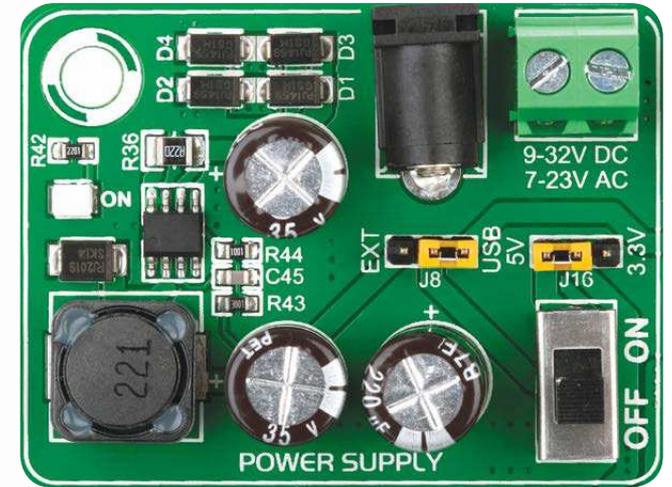
Figure 3-1: Power supply unit of EasyPIC™ v7 for dsPIC30®

3.3V VOLTAGE REGULATOR

POWER

5V SWITCHING POWER SUPPLY
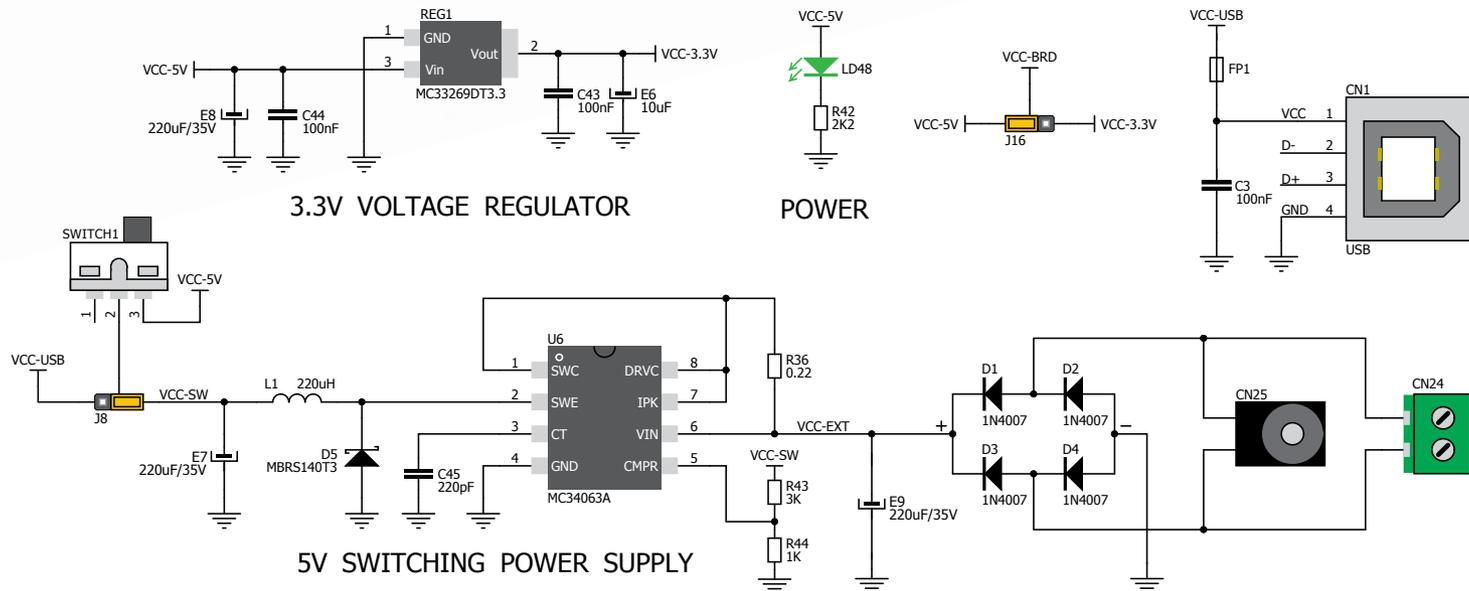
Figure 3-2: Dual power supply unit schematics

EasyPIC™ v7
for dsPIC30®

DUAL POWER SUPPLY

3.3V | 5V

EasyPIC™ v7 for dsPIC30® 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 23V AC or 9V to 32V DC), or via USB cable (5V DC)

Power capacity: up to 500mA with USB, and up to 600mA with external power supply

How to power the board?

1. With USB cable

Set J8 jumper to USB position

To power the board with USB cable, place jumper **J8** in USB position, and place jumper **J16** 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 J8 jumper to EXT position

To power the board via adapter connector, place jumper **J8** in EXT position, and place jumper **J16** 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.
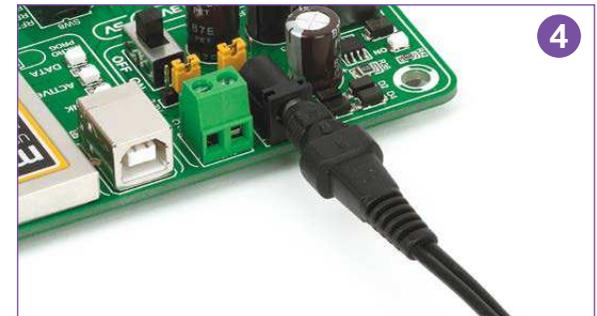
3. With laboratory power supply

Set J8 jumper to EXT position

To power the board using screw terminals, place jumper **J8** in EXT position, and place jumper **J16** 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.

EasyPIC™ v7
for dsPIC30®
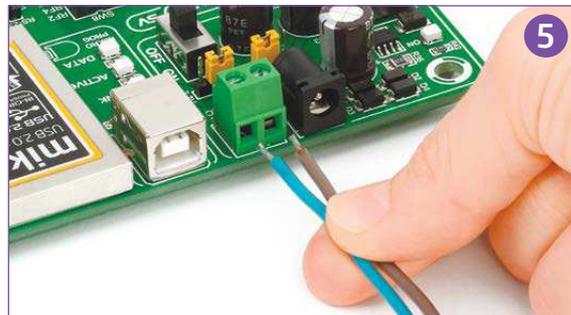
Supported microcontrollers

Board contains six DIP sockets: DIP40A, DIP40B, DIP28A, DIP28B, DIP28C, DIP18. With dual power supply and smart on-board mikroProg, board is capable of programming and debugging 14 different microcontrollers from dsPIC30F® family.

There are two DIP40, three DIP28 and one DIP18 socket for dsPIC30® microcontrollers provided on the board. Which of these sockets you will use depends solely on the pinout of the microcontroller in use. The **EasyPIC™ v7 for dsPIC30®** development system comes with the **dsPIC30F4013** microcontroller in a **DIP40** package (pinout corresponds to **DIP40B** socket). In order for microcontrollers to have a stable operation, we provided decoupling capacitive filters on every board socket, **Figure 4-1**.

Which socket to use?

Supported microcontrollers come in different DIP packages (DIP40, DIP28 and DIP18) and with different pinout. That is why you need to put chosen microcontroller in appropriate socket. Here is the list of supported microcontrollers with DIP sockets:

	DIP40A	dsPIC30F3011, dsPIC30F4011
	DIP40B	dsPIC30F3014, dsPIC30F4013 (default chip)
(see figure 4-1)	**DIP28A**	dsPIC30F2010, dsPIC30F3010, dsPIC30F4012
	DIP28B	dsPIC30F2012, dsPIC30F3013
	DIP28C	dsPIC30F1010, dsPIC30F2020
	DIP18	dsPIC30F2011, dsPIC30F3012
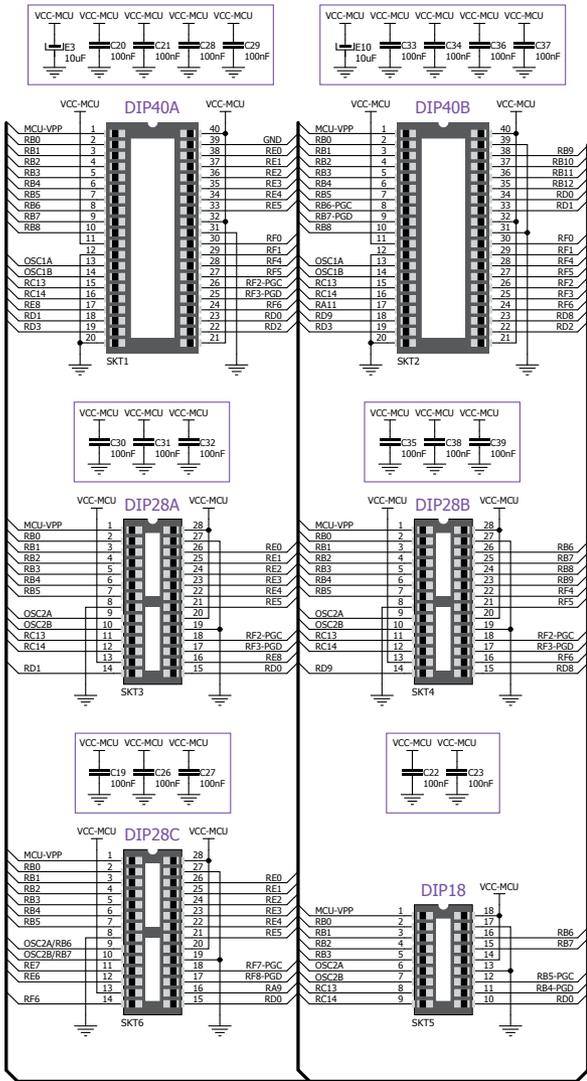
Figure 4-1: Schematics of on-board DIP sockets and decoupling capacitors
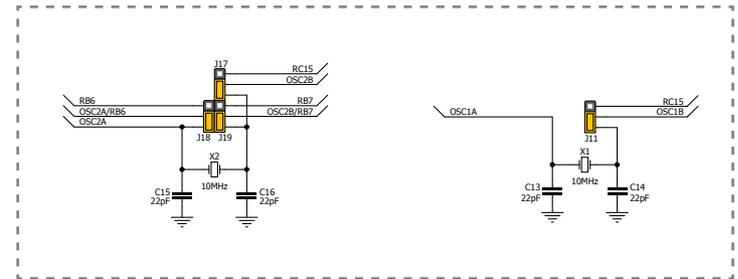
Figure 4-2: crystal oscillators

EasyPIC™ v7 *for dsPIC30®*

How to properly place your microcontroller into the DIP socket?
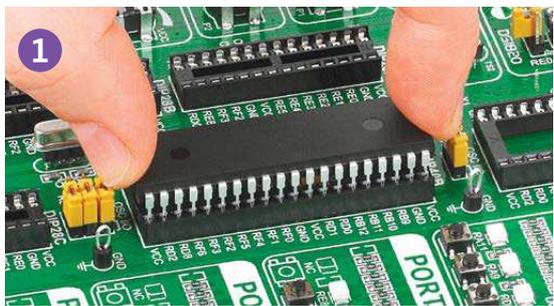
Figure 4-3: Place both ends of microcontroller on the socket so the pins are aligned correctly

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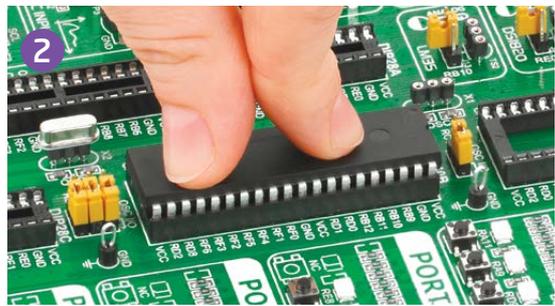
Figure 4-4: with both fingers, evenly distribute the force and press the chip into the socket.

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 4-3**. Then put the microcontroller slowly down until
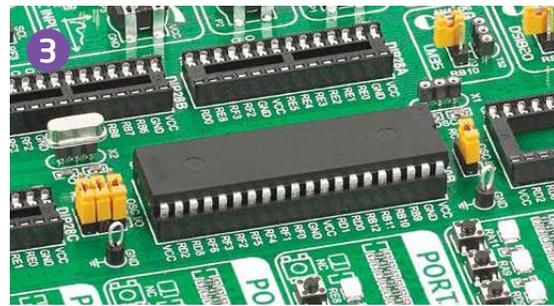
Figure 4-5: Properly placed microcontroller will have equally leveled pins.

all the pins match the socket as shown in **Figure 4-4**. Check again if everything is placed correctly and press the microcontroller until it is completely plugged into the socket as shown in **Figure 4-5**.

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

Using crystal oscillators

Figure 4-6: RC15 as I/O pin (when using internal oscillator)
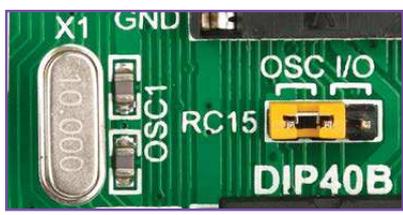
Figure 4-7: RC15 connected to X1 quartz-crystal
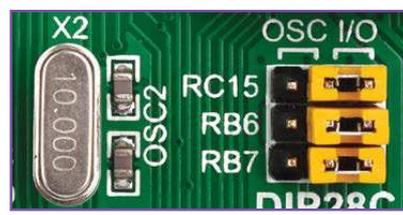
Figure 4-8: RC15, RB6 and RB7 as I/O pins (when using internal oscillator)
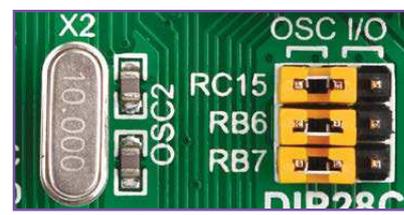
Figure 4-9: RC15, RB6 and RB7 connected to X2 quartz-crystal

PIC microcontrollers normally use a quartz crystal for the purpose of providing clock frequency. The **EasyPIC™ v7 for dsPIC30®** provides two sockets for quartz-crystal. Microcontrollers in **DIP40A** and **DIP40B** packages use socket **X1 (OSC1)** for quartz-crystal.

If you want to use microcontrollers in **DIP28A**, **DIP28B**, **DIP28C** and **DIP18** packages, it is necessary to put quartz crystal into socket **X2 (OSC2)**. The value of the crystal depends on the maximum clock frequency allowed and your application. You can always exchange the default 10MHz crystal with another one.

EasyPIC™ v7
for dsPIC30®

On-board programmer

What is mikroProg™?

mikroProg™ is a fast USB 2.0 programmer with mikroICD™ hardware In-Circuit Debugger. Smart engineering allows mikroProg™ to support all dsPIC30® DIP microcontrollers from Microchip® in a single programmer. Outstanding performance and easy operation are among it's 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 mikroProg Suite™ for PIC® software

2. Power up the board, and you are ready to go.
- Plug in the programmer USB cable
- LINK LED should light up.

Socket selection

Jumpers **J1, J2, J3** and **J4** are used to select **PGC** and **PGD** programming lines for your microcontroller. Make sure to place jumpers in the proper position for your socket.
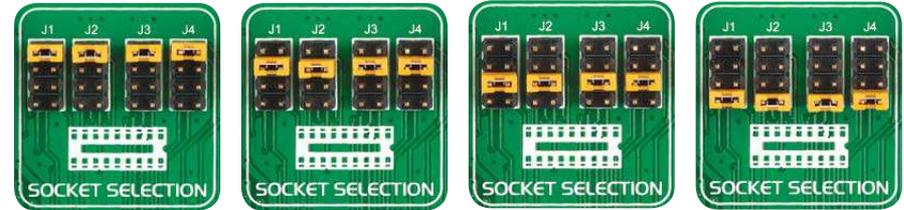
DIP40A	DIP40B	DIP28C	DIP18
DIP28A			
DIP28B			

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 programmer is active. **Data** is on when data is being transferred between the programmer and PC software (compiler or mikroProg Suite™ for PIC®).
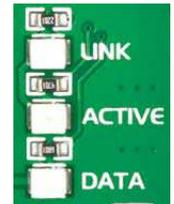
⚠️ ## IMPORTANT - Programming/Debugging mode raises MCU voltage to 5V!

When on-board mikroProg™ programmer is in programming/debugging mode, it automatically raises the MCU voltage to 5V regardless of the power supply selection (jumper J16). Note that all external peripherals that can only work on 3.3V must be removed from the board when programming/debugging is in progress, otherwise they can be permanently damaged. When programming/debugging is not in progress (execute mode or release mode) you can connect both 3.3V or 5V peripherals to the board depending on the position of jumper J16.

EasyPIC™ v7
for dsPIC30®
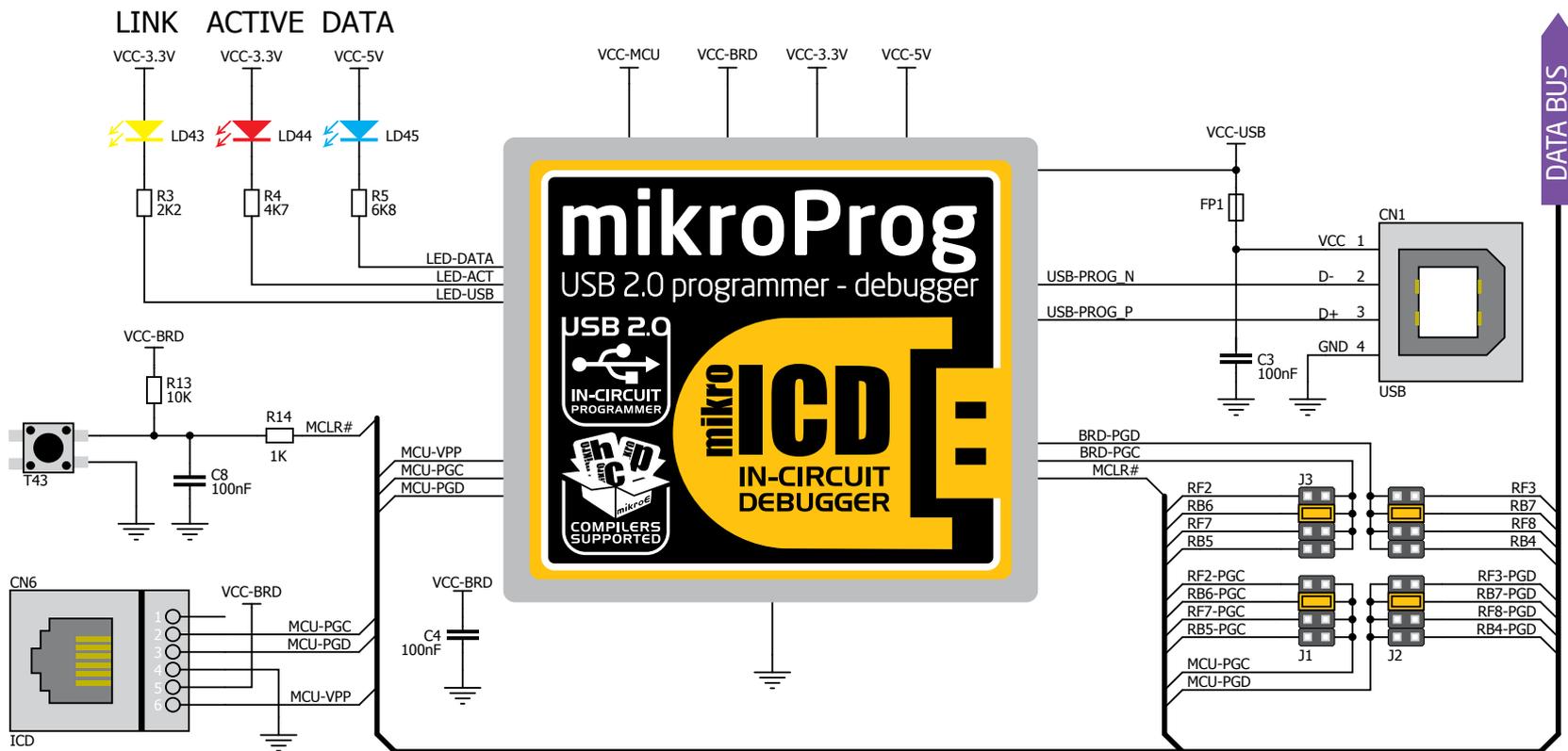
Figure 5-1: mikroProg™ block schematics

Programming with ICD2/ICD3

EasyPIC™ v7 for dsPIC30® is equipped with RJ-12 connector compatible with Microchip® ICD2® and ICD3® external programmers. You can either use the on-board mikroProg™ programmer or external programming tools with the board as long as you use only one of them at the same time. Still you have to **set the appropriate jumpers**, as described in the previous page. Insert your ICD programmer cable into connector **CN6**, as shown in images **1** and **2**.

EasyPIC™ v7
for dsPIC30®
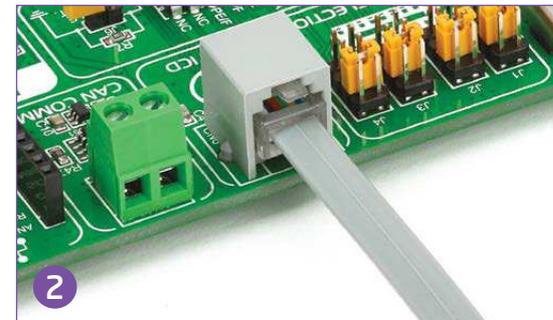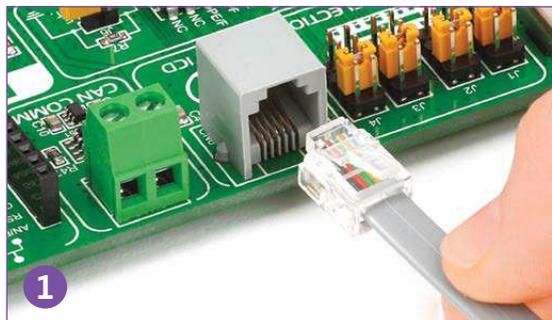
Installing programmer drivers

On-board mikroProg™ requires drivers in order to work. Drivers are located on the   that you received with the EasyPIC™ v7 for dsPIC30® package:

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

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

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

Win 98

Win 2000, XP, 2003 32-bit

Win XP, 2003 64-bit

installing_usb_drivers.pdf

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.EXE
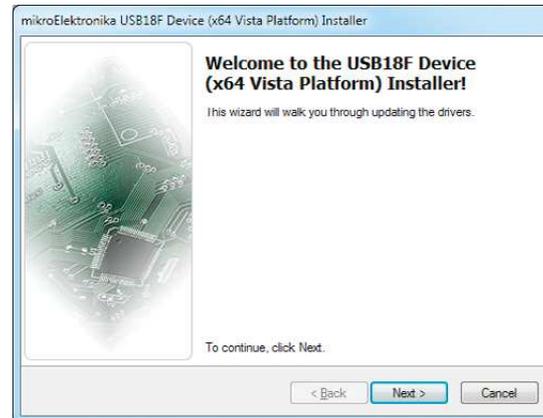
Step 1 - Start Installation

Welcome screen of the installation. Just click on **Next** button to proceed.
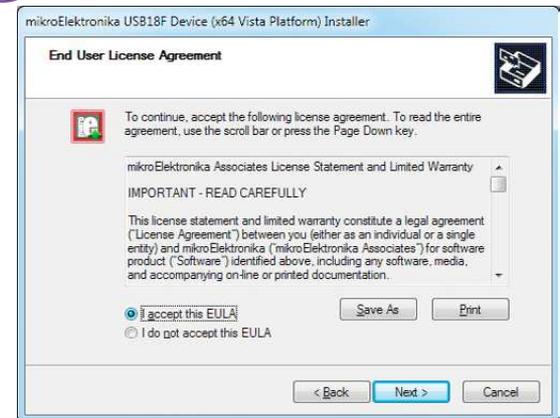
Step 2 - Accept EULA

Carefully read **E**nd **U**ser **L**icense **A**greement. If you agree with it, click **Next** to proceed.
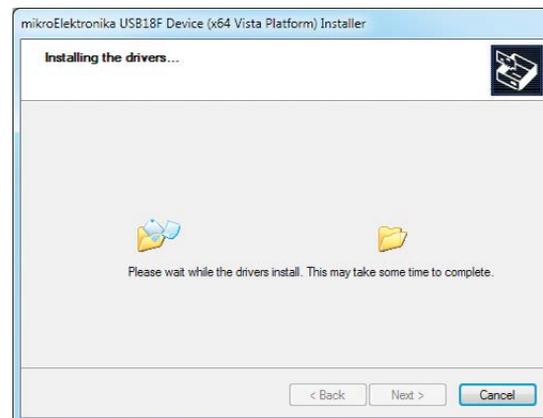
Step 3 - Installing drivers

Drivers are installed automatically in a matter of seconds.
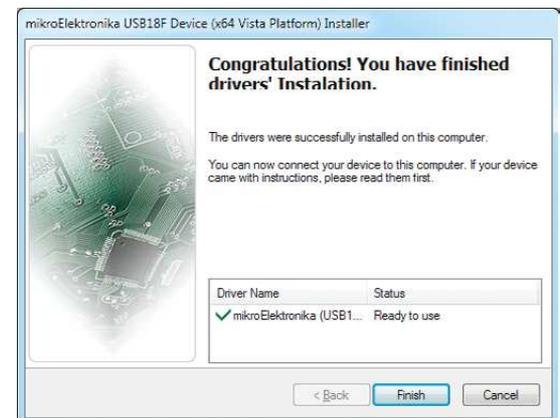
Step 4 - Finish installation

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

EasyPIC v7
for dsPIC30®

Programming software

mikroProg Suite™ for PIC®

On-board **mikroProg™** programmer requires special programming software called **mikroProg Suite™ for PIC®**. This software is used for programming all of Microchip® microcontroller families, including PIC10, PIC12, PIC16, PIC18, dsPIC30/33, PIC24 and PIC32. Software has intuitive interface and **SingleClick™** programming technology. To begin, first locate the installation archive on our website:

http://www.mikroe.com/downloads/get/1201/mikroprog_suite_for_pic.zip

After downloading, extract the package and double click the executable setup file, to start installation.
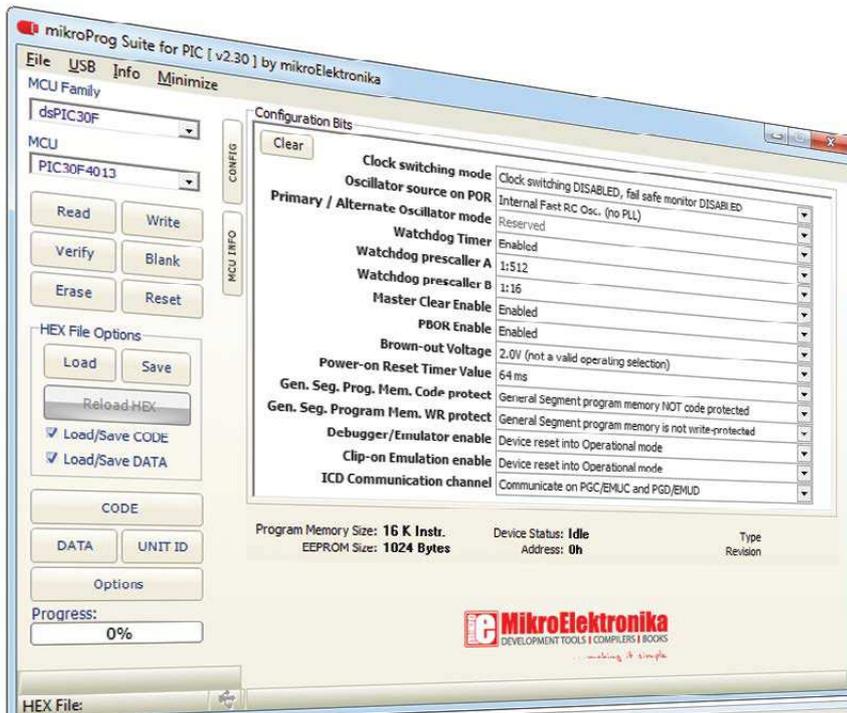
Installation wizard - 6 simple steps

Step 1 - Start Installation

Step 2 - Accept EULA and continue
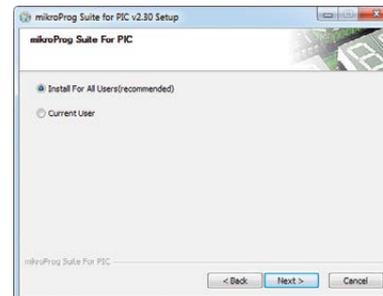
Step 3 - Install for All users or current user
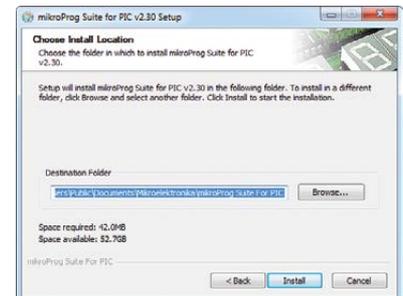
Step 4 - Choose destination folder
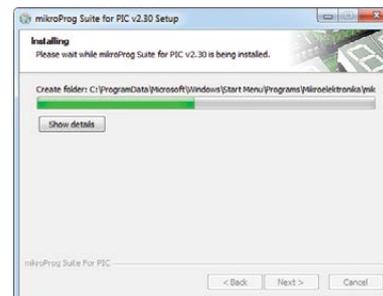
Step 5 - Installation in progress

Step 6 - Finish Installation

mikroICD™ - In Circuit Debugger

What is Debugging?

Every developer comes to a point where he has to monitor the code execution in order to find errors in the code, or simply to see if everything is going as planed. This hunt for bugs, or errors in the code is called **debugging**. There are two ways to do this: one is **the software simulation**, which enables you to simulate what is supposed to be happening on the microcontroller as your code lines are executed, and the other, most reliable one, is monitoring the code execution on the chip itself. And this latter one is called **In-Circuit debugging**. "In-Circuit" means that it is the real deal - code executes right on the target device.

What is mikroICD™?

The on-board **mikroProg™** programmer supports **mikroICD™** - a highly effective tool for a **Real-Time debugging** on hardware level. The mikroICD™ debugger enables you to execute your program on the host PIC microcontroller and view variable values, Special Function Registers (SFR), RAM, CODE and EEPROM memory along with the mikroICD™ code execution on hardware. Whether you are a beginner, or a professional, this powerful tool, with intuitive interface and convenient set of commands will enable you to track down bugs quickly. mikroICD™ is one of the fastest, and most reliable debugging tools on the market.

Supported Compilers

All MikroElektronika compilers, **mikroC™**, **mikroBasic™** and **mikroPascal™** for PIC®, dsPIC® and PIC32® natively support mikroICD™. Specialized mikroICD™ DLL module allows compilers to exploit the full potential of fast hardware debugging. Along with compilers, make sure to install the appropriate **programmer drivers** and **mikroProg Suite™ for PIC®** programming software, as described on **pages 12** and **13**.

How do I use the debugger?

When you build your project for debugging, and program the microcontroller with this HEX file, you can start the debugger using **[F9]** command. Compiler will change layout to debugging view, and a blue line will mark where code execution is currently paused. Use **debugging toolbar** in the **Watch Window** to guide the program execution, and stop anytime. Add the desired variables to Watch Window and monitor their values. Complete guide to using mikroICD™ with your compiler is provided with the EasyPIC™ v7 for dsPIC30® package.

Figure 5-2: mikroICD™ manual explains debugging thoroughly
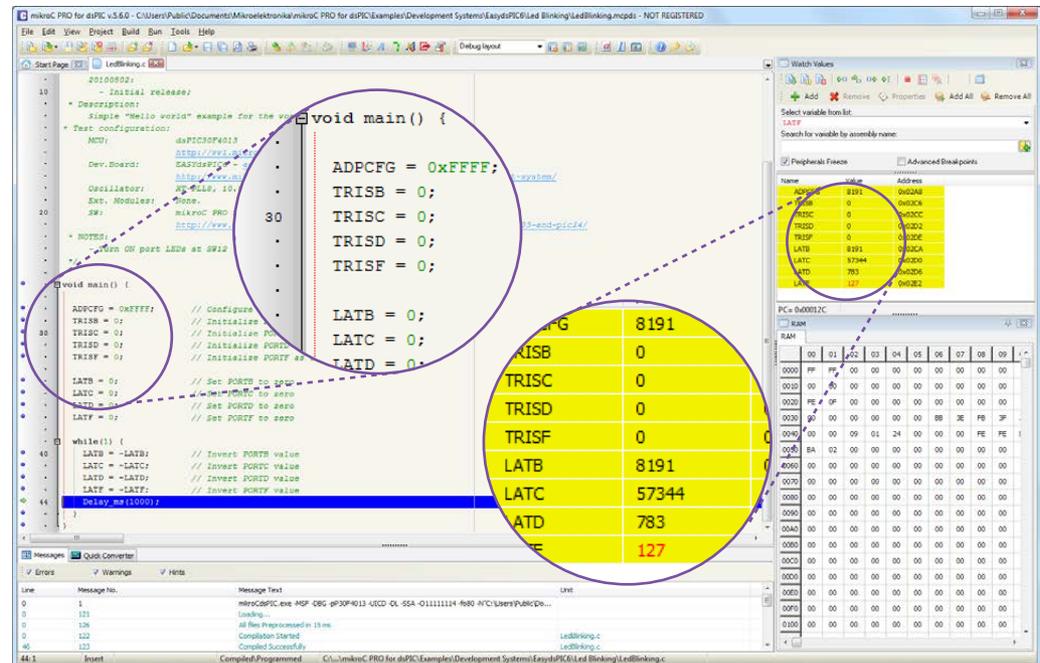
Figure 5-3: mikroC PRO for dsPIC® compiler in debugging view, with SFR registers in Watch Window

EasyPIC v7
for dsPIC30®

mikroICD™ commands

Here is a short overview of which debugging commands are supported in MikroElektronika compilers. You can see what each command does, and what are their shortcuts when you are in debugging mode. It will give you some general picture of what your debugger can do.

Toolbar Icon	Command Name	Shortcut	Description
	Start Debugger	[F9]	Starts Debugger.
	Run/Pause Debugger	[F6]	Run/Pause Debugger.
	Stop Debugger	[Ctrl + F2]	Stops Debugger.
	Step Into	[F7]	Executes the current program line, then halts. If the executed program line calls another routine, the debugger steps into the routine and halts after executing the first instruction within it.
	Step Over	[F8]	Executes the current program line, then halts. If the executed program line calls another routine, the debugger will not step into it. The whole routine will be executed and the debugger halts at the first instruction following the call.
	Step Out	[Ctrl + F8]	Executes all remaining program lines within the subroutine. The debugger halts immediately upon exiting the subroutine.
	Run To Cursor	[F4]	Executes the program until reaching the cursor position.
	Toggle Breakpoint	[F5]	Toggle breakpoints option sets new breakpoints or removes those already set at the current cursor position.
	Show/Hide breakpoints	[Shift+F4]	Shows/Hides window with all breakpoints
	Clears breakpoints	[Shift+Ctrl+F5]	Deletes selected breakpoints
	Jump to interrupt	[F2]	Opens window with available interrupts (doesn't work in mikroICD™ mode)

EasyPIC ™v7
for dsPIC30®

UART via RS-232

Enabling RS-232

In order to enable RS-232 communication, you must set **J20** and **J21** jumpers in the **RS-232** position, and enable desired **RX** and **TX** lines via **SW8** and **SW9** DIP switches. For example, if you want to enable RS-232 connection on UART1 module of the default **dsPIC30F4013** chip, you should enable **SW8.1** (RF2) and **SW9.1** (RF3) 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**.
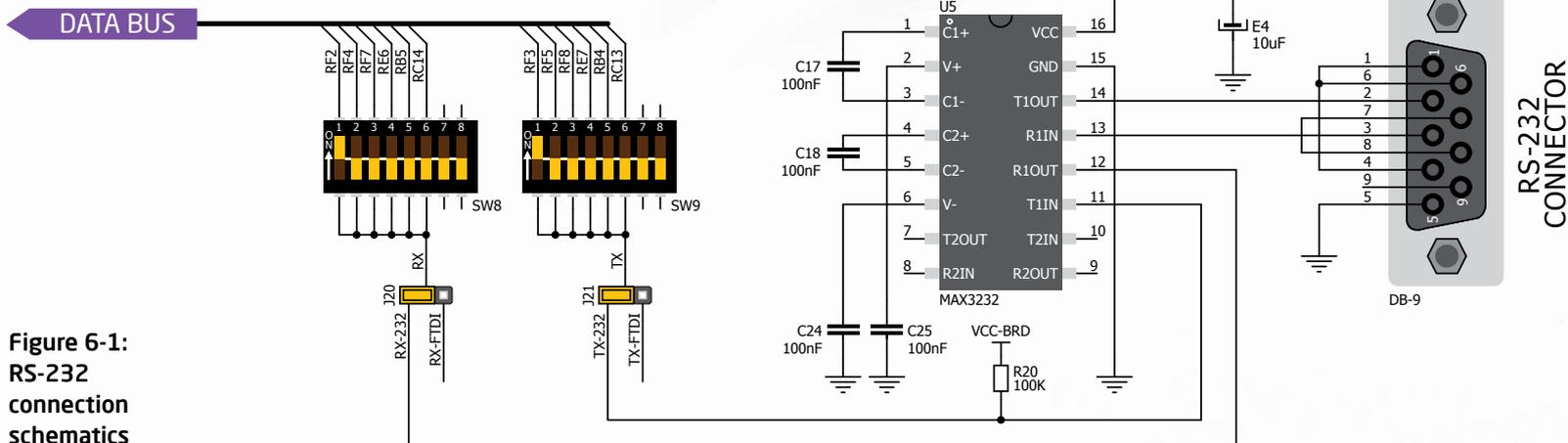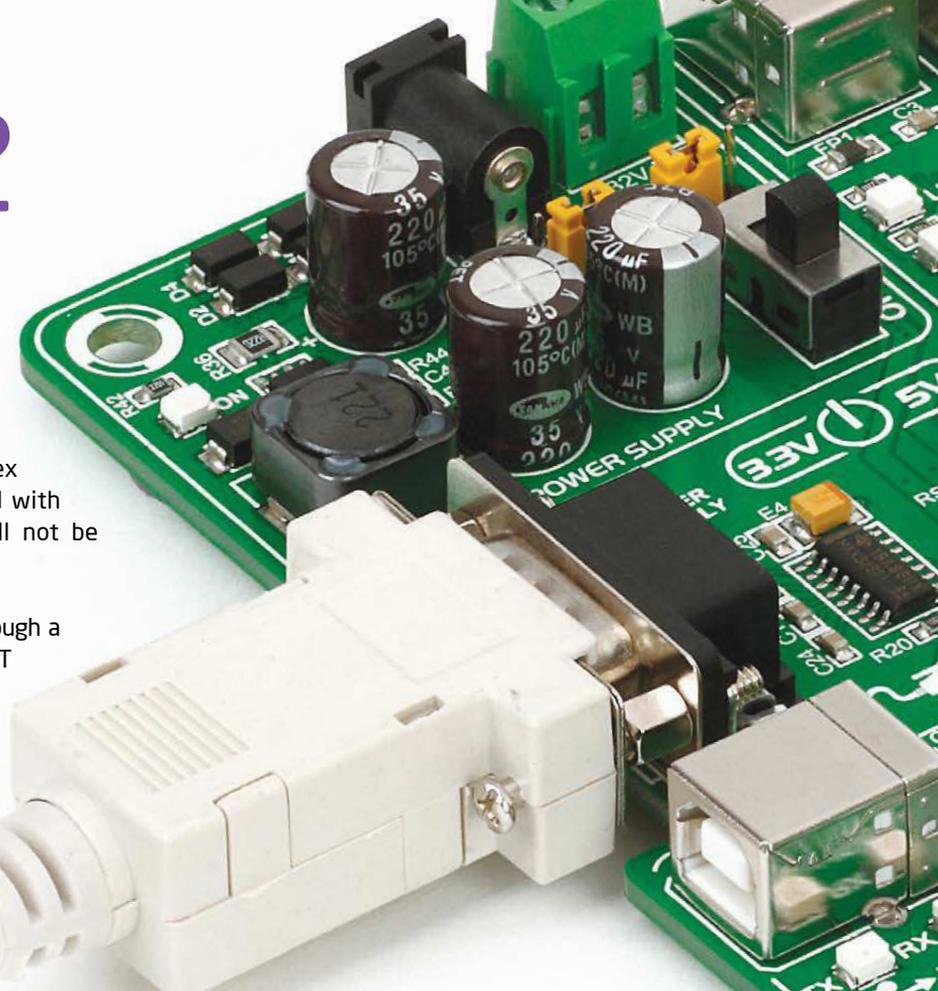
**Figure 6-1:
RS-232
connection
schematics**

EasyPIC™ v7
for dsPIC30®

UART via USB

Enabling USB-UART

In order to enable USB-UART communication, you must set **J20** and **J21** jumpers in the **USB-UART** position, and enable desired **RX** and **TX** lines via **SW8** and **SW9** DIP switches. For example, if you want to enable USB-UART connection on UART1 module of the default **dsPIC30F4013** chip, you should enable **SW8.1** (RF2) and **SW9.1** (RF3) 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 over USB connection. Controllers such as **FT232RL** from FTDI® convert UART signals to the appropriate USB standard. In order to use USB-UART module on EasyPIC™ v7 for dsPIC30®, you must first install FTDI drivers on your computer. Drivers can be found on the link below:

http://www.ftdichip.com/Drivers/VCP.htm

USB-UART communication is being done through a FT232RL controller, USB connector **(CN2)**, and microcontroller UART module. To establish this connection, you must put **J20** and **J21** jumpers in the **USB-UART** position, and connect **RX** and **TX** lines of the microcontroller to the appropriate input and output pins of the FT232RL. This selection is done using DIP switches **SW8** and **SW9**.
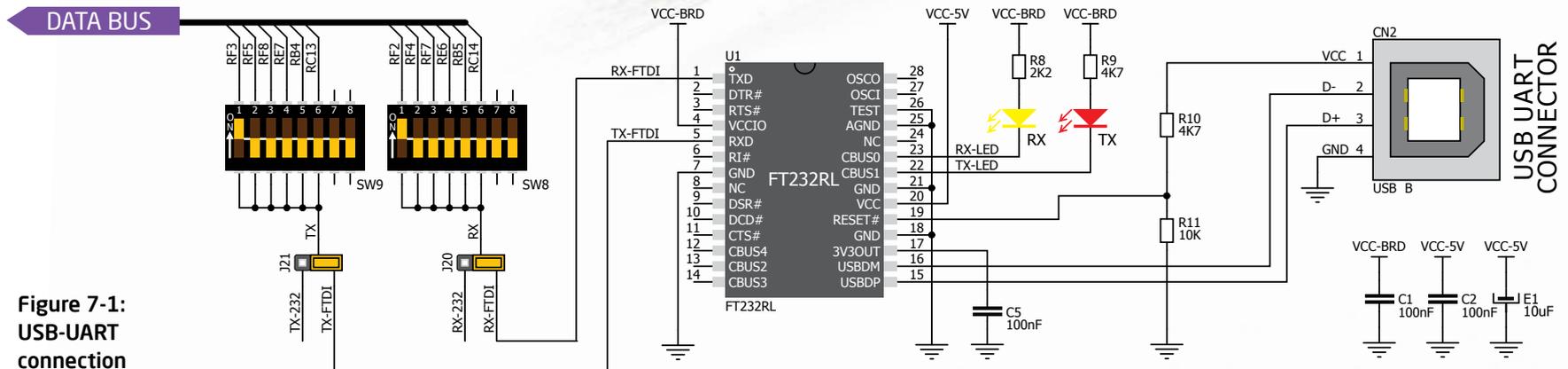
Figure 7-1: USB-UART connection schematics

EasyPIC™v7
for dsPIC30®
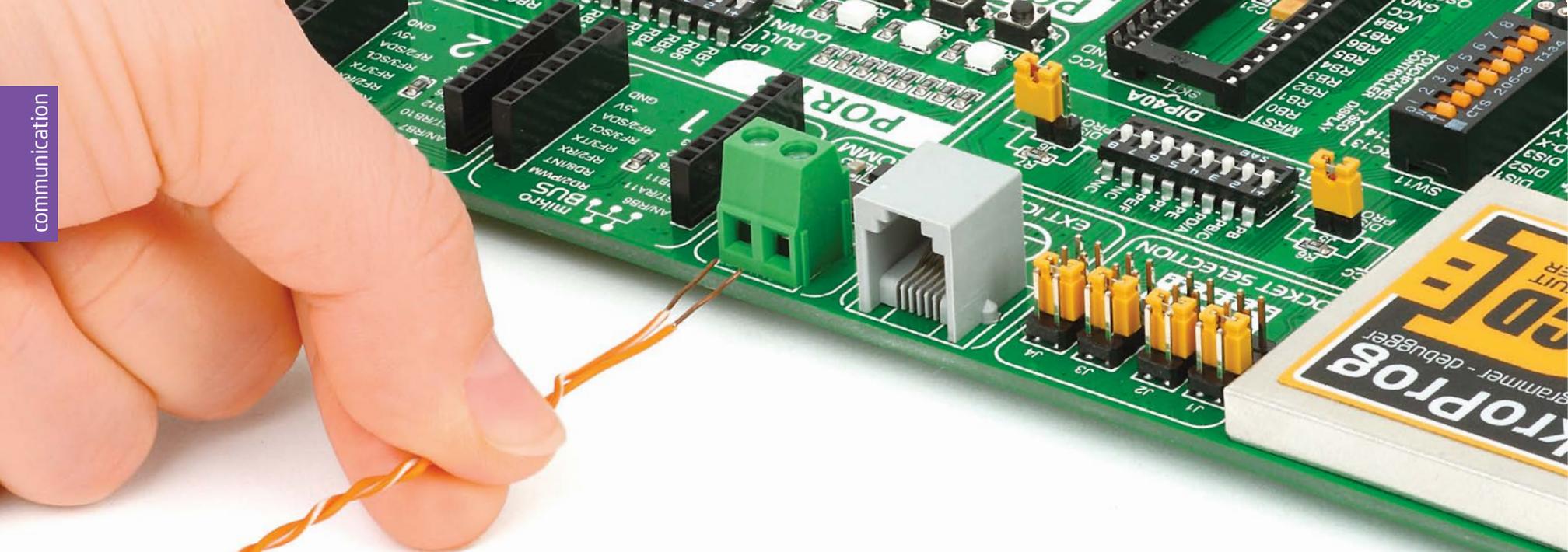
CAN communication

Controller Area Network (CAN or CAN bus) is a vehicle bus standard designed to allow microcontrollers and devices to communicate with each other within a vehicle without a host computer. CAN is a message-based protocol, designed specifically for automotive applications but now also used in other areas such as industrial automation and medical equipment. EasyPIC™ v7 for dsPIC30® is equipped with high-speed **MCP2551** CAN Transceiver and a pair of screw terminals which provide microcontrollers with integrated CAN controller with the necessary physical interface for CAN communication. Make sure to correctly connect negative and positive differential communication lines before using this module.

Enabling CAN
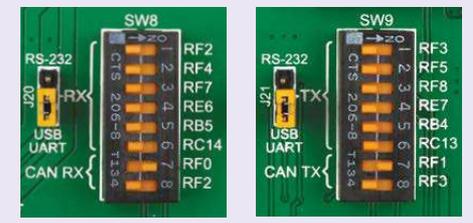
In order to enable CAN communication, you must push **SW8.7** (RF0) or **SW8.8** (RF2) and **SW9.7** (RF1) or **SW9.8** (RF3) to **ON** position. This connects the **TX** and **RX** lines to appropriate microcontroller pins and its CAN module.

Figure 8-1:
CAN connection schematics

EasyPIC v7
for dsPIC30®
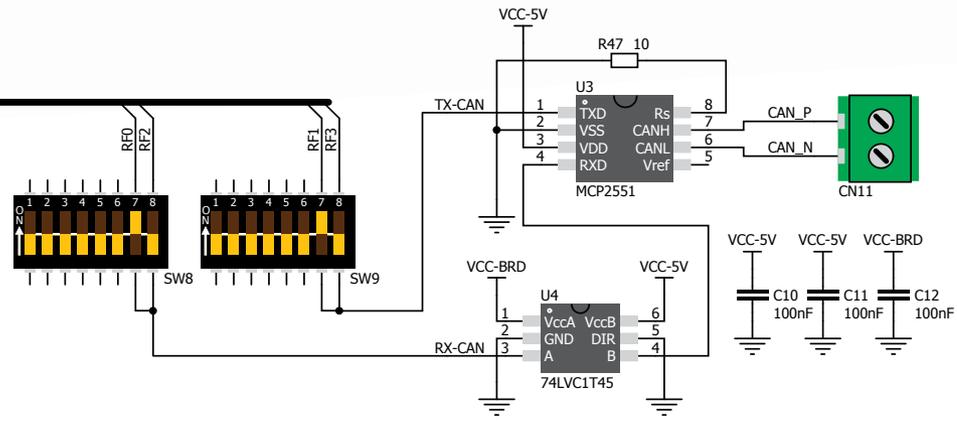
mikroBUS™ sockets

Easier connectivity and simple configuration are imperative in modern electronic devices. Success of the USB standard comes from it's 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™.** EasyPIC™ v7 for dsPIC30® supports mikroBUS™ with two 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.
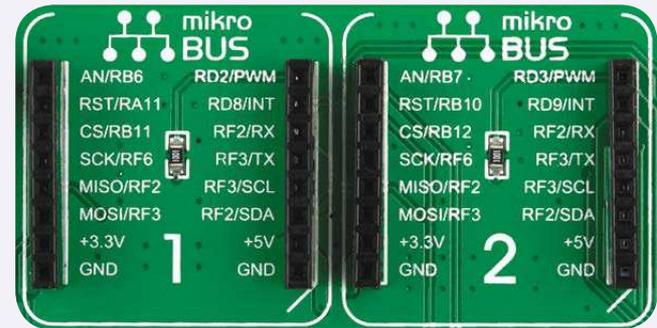
mikroBUS™ pinout explained

AN - Analog pin
RST - Reset pin
CS - SPI Chip Select line
SCK - SPI Clock line
MISO - SPI Slave Output line
MOSI - SPI Slave Input line
+3.3V - VCC-3.3V power line
GND - Reference Ground

PWM - PWM output line
INT - Hardware Interrupt line
RX - UART Receive line
TX - UART Transmit line
SCL - I2C Clock line
SDA - I2C Data line
+5V - VCC-5V power line
GND - Reference Ground

DATA BUS
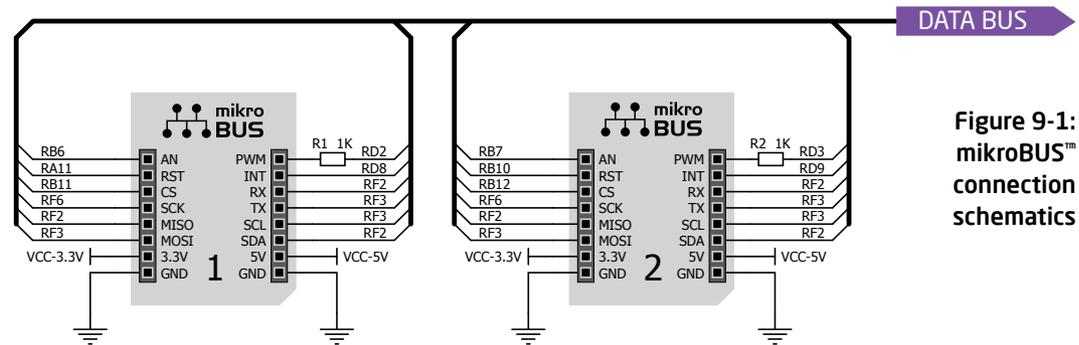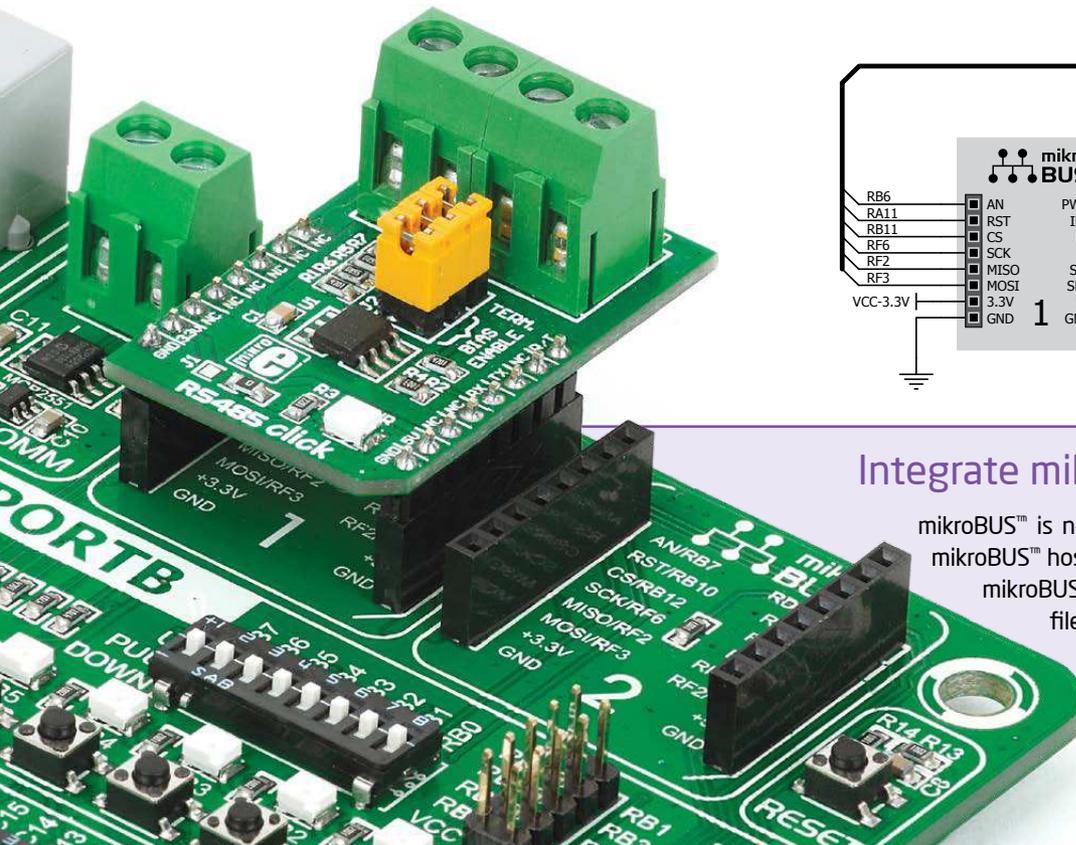
Figure 9-1:
mikroBUS™
connection
schematics

Integrate mikroBUS™ in your design

mikroBUS™ is not made to be only a part of our development boards. You can freely place mikroBUS™ host connectors in your final PCB designs, as long as you clearly mark them with mikroBUS™ logo and footprint specifications. For more information, logo artwork and PCB files visit our web site:

http://www.mikroe.com/mikrobus/
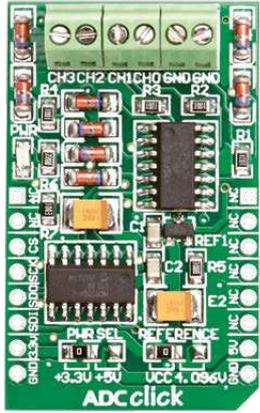
ADC click™
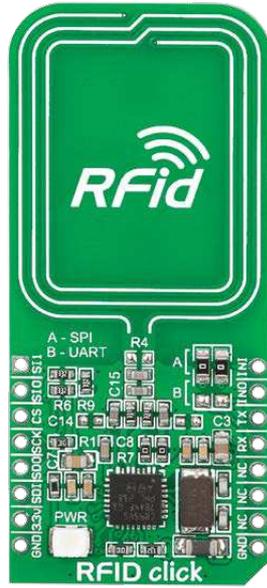
RFiD click™

BlueTooth click™
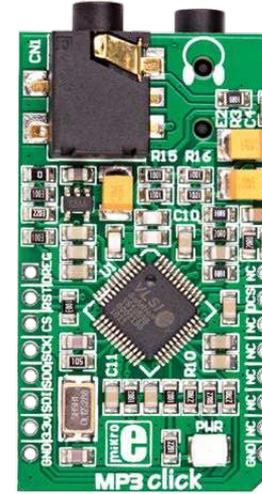
MP3 click™

GSM 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 EasyPIC™ v7 for dsPIC30® 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/
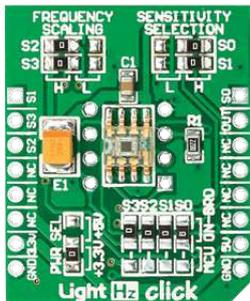
LightHz click™
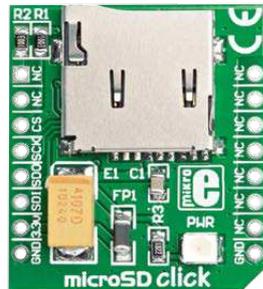
microSD click™
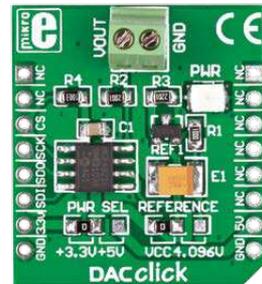
DAC click™

DIGIPOT click™

SHT1x click™

EasyPIC™ v7
for dsPIC30®
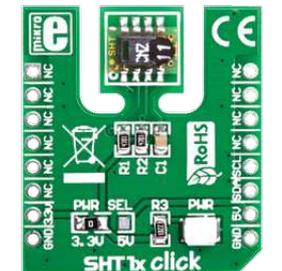
WiFi PLUS click™   GPS click™   RS485 click™   CAN SPI click™   THERMO click™
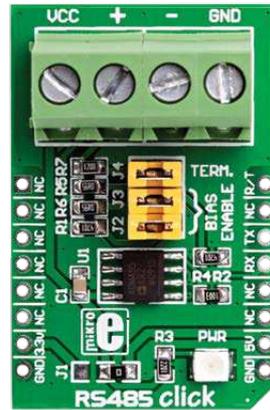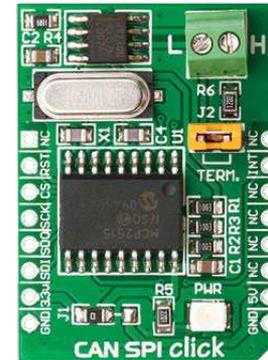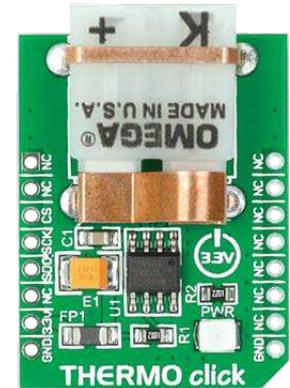
LIBSTOCK ™ A PLACE FOR CODE®

BROWSE CODE | SHARE CODE | REQUEST CODE | ABOUT LIBSTOCK

Enter search keywords

Code Examples

It easy to get your Click™ board up and running. We provided the examples for mikroC™, mikroBasic™ and mikroPascal™ compilers on our **Libstock community web site**. Just download them and you are ready to start:

http://www.libstock.com

Use your forum account to login.

LOGIN

Username:

Password:

▶ LOGIN

Not registered? Sign Up for free!

CODE CATEGORIES

All codes (789)

▶ Audio & Speech (20)
▶ Communication (194)
▶ Graphics & LCD (94)

CATEGORIES ○ All codes ○ Libraries ○ Projects ○ Visual Projects ⦿ Examples

COMPILERS ☐ mikroC ☐ mikroBasic ☐ mikroPascal ☐ Other Compilers

SORT BY ☐ best rank ☐ oldest ☐ by author ☐ alphabetically ☑ most downloaded ☐ with source ☐ with example

CATEGORY: ALL CODES   << previous | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 next >> last

Ⓔ **GSM click - Example**

Rating: ★★★★★
Author: MikroElektronika Team
Version: 1.0.0.0
Last Updated: 2014-05-09
Category: GSM / GPRS
Downloaded: 7921 times
Followed by: 2 users

Description

This is a sample program which demonstrates the use of **GSM click**. Program uses GSM module GL865 for sending/receiving SMS and establishing calls with dialed number. Dialing is done using GLCD+Touch Panel GUI.

[learn more]

Download [648.97KB]

supported microcontrollers
PIC DSPIC PIC24 PIC32 AVR 8051 ARM OTHER

supported compilers
MIKROC MIKROBASIC MIKROPASCAL OTHER

supported visual tools
VISUAL TFT VISUAL GLCD OTHER

archive format
MIKE ZIP RAR

project contains
LIB SRC EXA HEX MCL SCH PCB PIC

EasyPIC™ v7 for dsPIC30®

Input/Output Group

One of the most distinctive features of EasyPIC™ v7 for dsPIC30® are it's 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 EasyPIC™ v7 for dsPIC30® 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.
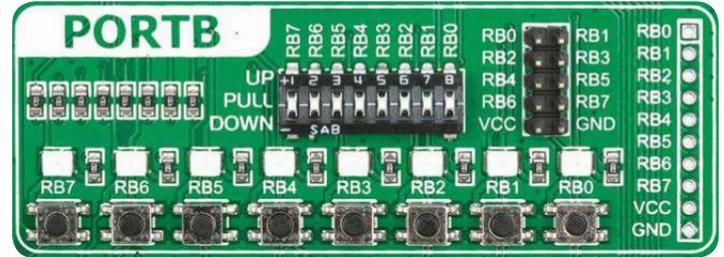
Figure 10-1: I/O group contains PORT headers, tri-state pull up/down DIP switch, buttons and LEDs all in one place

Tri-state pull-up/down DIP switches

Tri-state DIP switches, like **SW1** on **Figure 10-2**, are used to enable 4K7 pull-up or pull-down resistor on any desired port pin. Each of those switches has three states:
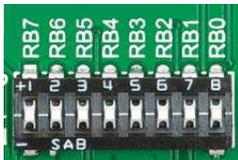
Figure 10-2: Tri-state DIP switch on PORTB

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.
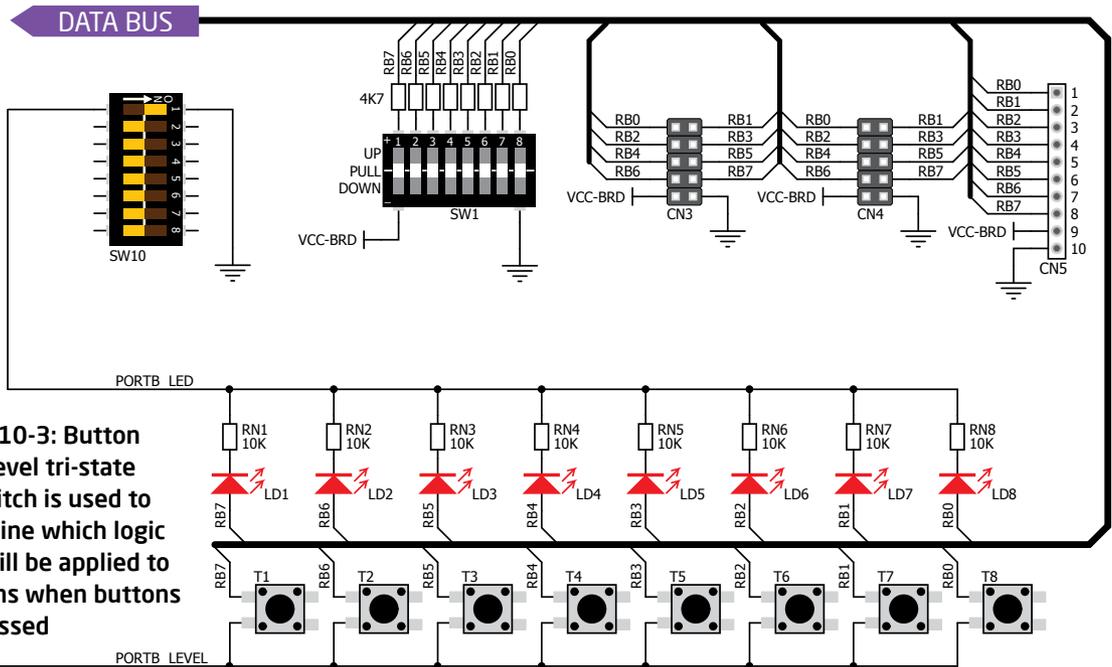
Figure 10-3: Button press level tri-state DIP switch is used to determine which logic level will be applied to port pins when buttons are pressed
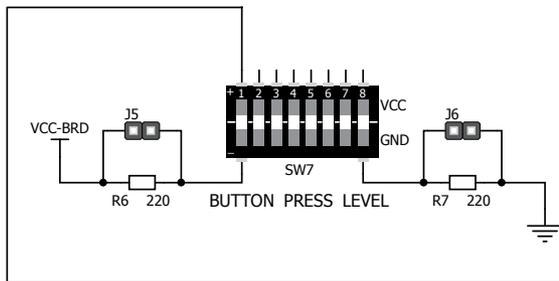
Figure 10-4: Schematic of the single I/O group connected to microcontroller PORTB

connectivity

EasyPIC v7
for dsPIC30®
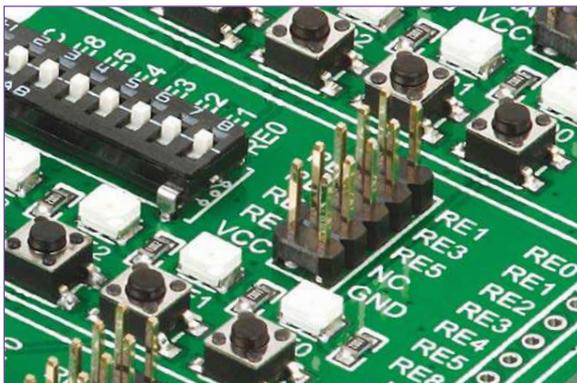
Headers

With enhanced connectivity as one of the key features of EasyPIC™ v7 for dsPIC30®, we have provided **two connection headers for each PORT**. I/O PORT group contains one male IDC10 2x5 header (like **CN3** on **Figure 10-4**) and one 1x10 row of connection pads (like **CN5** on **Figure 10-4**). There is **one more IDC10 header** available on the left side of the board, next to the section with displays. These headers are all compatible with all MikroElektronika accessory boards, and enable simple connection.
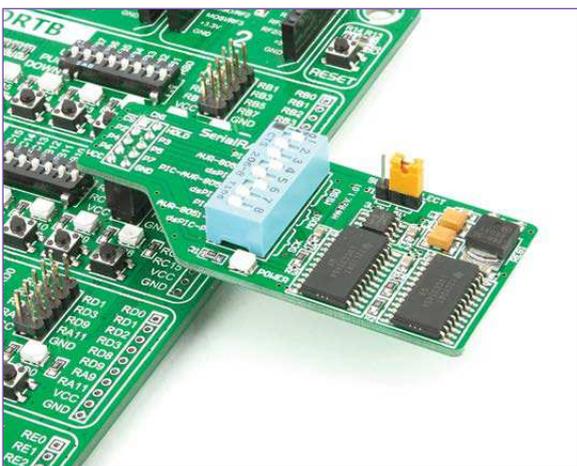
Figure 10-5: IDC10 male headers enable easy connection with MikroElektronika accessory boards
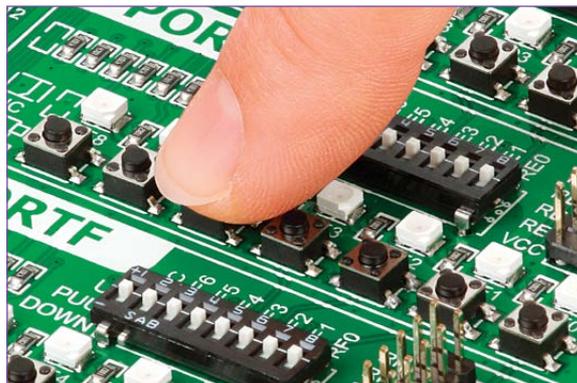
Buttons

Figure 10-6: 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 **SW7** 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 **SW7.1** 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 **J5** and **J6**, 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

Microcontroller

SMD resistor limiting current through the LED
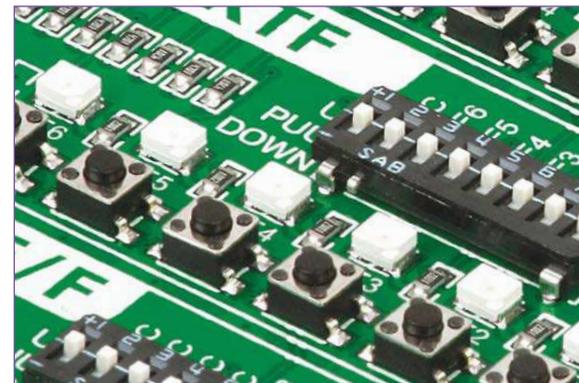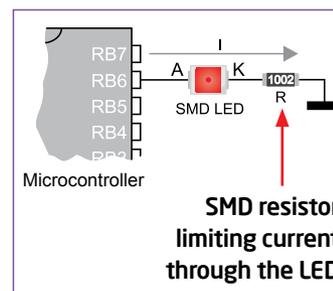
LED (Light-Emitting Diode) is a highly efficient electronic light source. When connecting LEDs, it is necessary to place a current limiting resistor in series so that LEDs are provided with the current value specified by the manufacturer. The current varies from 0.2mA to 20mA, depending on the type of the LED and the manufacturer. The EasyPIC™ v7 for dsPIC30® board uses low-current LEDs with typical current consumption of 0.2mA or 0.3mA. Board contains 42 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 switch on **SW10** (Figure 10-7).
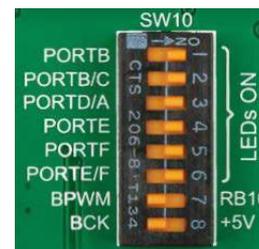
Figure 10-7: SW10.1 through SW10.6 switches are used to enable PORT LEDs

EasyPIC™ v7 for dsPIC30®
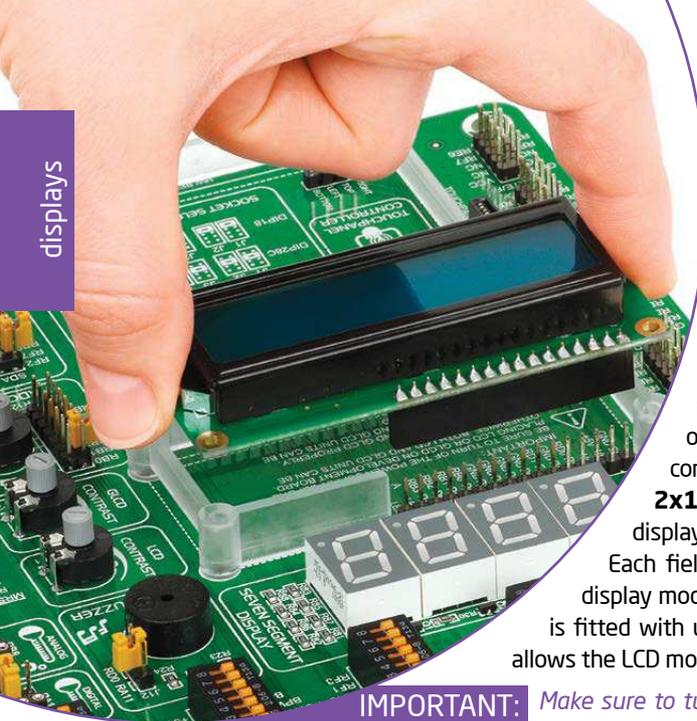
LCD 2x16 characters

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. EasyPIC™ v7 for dsPIC30® 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 **CN7** 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.*
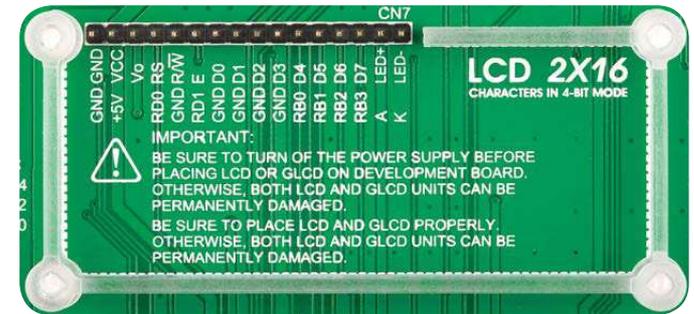
Figure 11-1: On-board LCD 2x16 display connector

Connector pinout explained

GND and **VCC** - Display power supply lines
Vo - LCD contrast level from potentiometer P4
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 11-2: 2x16 LCD connection schematics
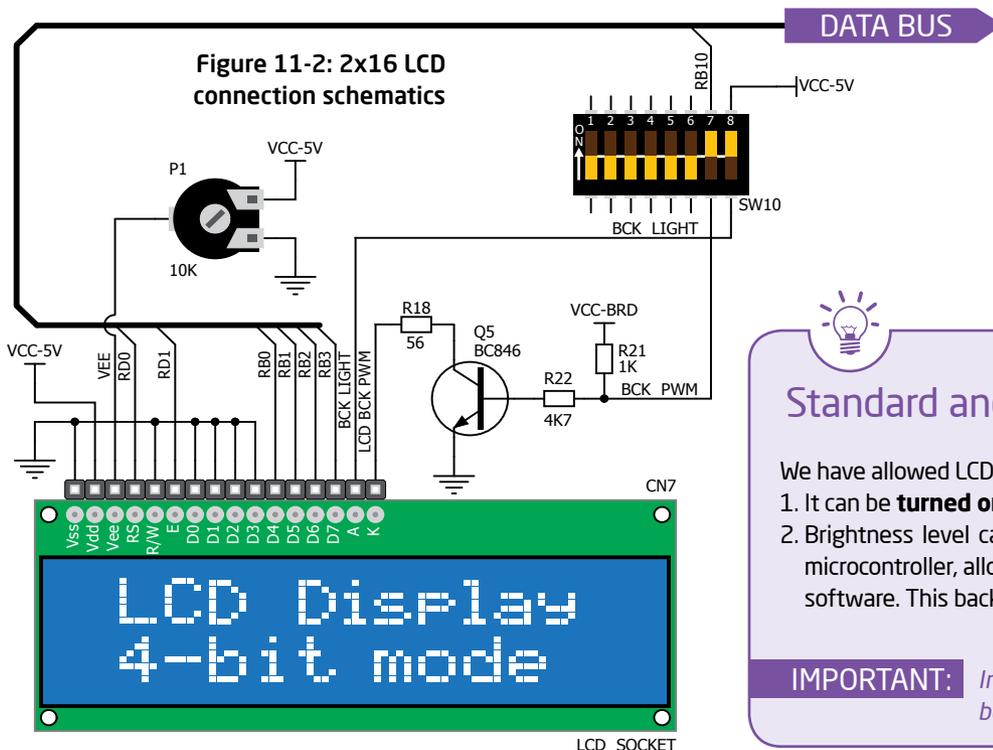
Standard and PWM-driven back light

We have allowed LCD back light to be enabled in two different ways:
1. It can be **turned on with full brightness** using **SW10.8** 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 **SW10.7** switch.

IMPORTANT: *In order to use PWM back light both **SW10.7** and **SW10.8** switches must be enabled at the same time.*

EasyPIC ™ v7
for dsPIC30®

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. EasyPIC™ v7 for dsPIC30® 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 RF0, RF1, RF4, RF5 (control lines) and to RB0-RB3, RD0-RD3 (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 **P2**. Full brightness display back light can be enabled with **SW10.8** switch, and PWM-driven back light with **SW10.7** switch.

Figure 12-1: GLCD 128x64 connection schematics
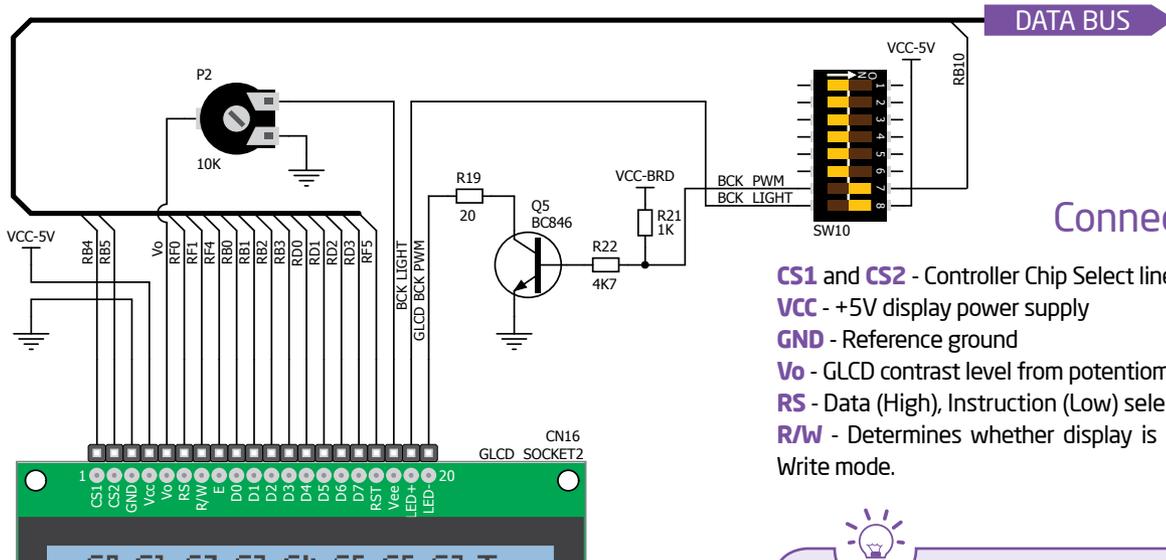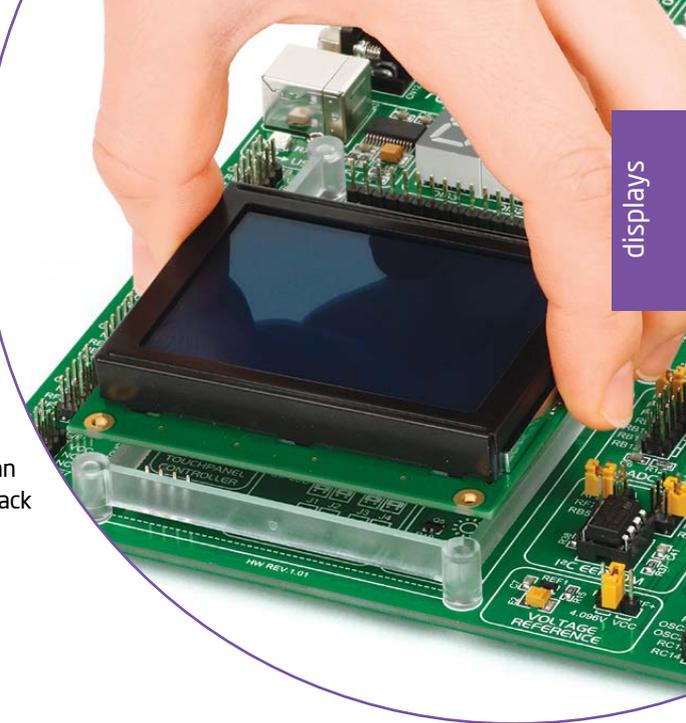
Connector pinout explained

CS1 and **CS2** - Controller Chip Select lines
VCC - +5V display power supply
GND - Reference ground
Vo - GLCD contrast level from potentiometer P3
RS - Data (High), Instruction (Low) selection line
R/W - Determines whether display is in Read or Write mode.

E - Display Enable line
D0-D7 - Data lines
RST - Display reset line
Vee - Reference voltage for GLCD contrast potentiometer P3
LED+ - Connection with the back light LED anode
LED- - Connection with the back light LED cathode

Standard and PWM-driven back light

As for LCD, we have allowed GLCD back light to be enabled in two different ways:
1. It can be **turned on with full brightness** using **SW10.8** 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 **SW10.7** switch.

IMPORTANT: *In order to use PWM back light both **SW10.7** and **SW10.8** switches must be enabled at the same time.*

EasyPIC™ v7
for dsPIC30®