

User's guide

EasyPIC[™] v7 is the seventh generation of the famous EasyPIC development board. It is the product of accumulated knowledge over the past 10 years, and it's state of the art in design, functionality and quality. It is loved by beginners and hobbyists, as well as professionals, and used in education and development by thousands. It is equipped with many on-board modules, including mikroProg[™] fast programmer with mikrolCD[™] (In-Circuit Debugger) support.

PIC^wv7

connectivity

● ● mikro ● ● ● BUS

Las

JUAL POWER SUPPL

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 EasyPICTM development board was born. And we all grew together with EasyPICTM. In its each and tiniest piece we had put all of our energy, creativity and sense of what's best for an engineer. I've personally assembled hundreds of early EasyPICTM boards myself with my home soldering iron.

Today, we present you the 7th generation of the board, 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

Introduction

Introduction	04
It's good to know	05

Power Supply

Dual power supply	06
	00

Supported MCUs

Supported microcontrollers	80
----------------------------	----

Programmer/debugger

On-board programmer	10
Installing programmer drivers	12
Programming software	13
mikrolCD™ - In Circuit Debugger	14

Communication

UART via RS-232	16
UART via USB	17
USB connection	18

Connectivity

mikroBUS sockets	19
Input/Output Group	20

Displays

LCD 2x16 characters	22
GLCD 128x64px	23
Touchpanel controller	24
4 digit 7-seg display	25

Modules

DS1820 - Digital Temperature Sensor	26
LM35 - Analog Temperature Sensor	27
ADC inputs	28
I2C EEPROM	29
Piezo Buzzer	30
Additional GNDs	31

What's next

	77
What's Next?	 32



Introduction

EasyPIC[™] is an old friend. Its been with us for six generations. Many of us made our first steps in embedded world with EasyPIC[™]. Today it has thousands of users: students, hobbyists, enthusiasts and professionals. It's used in many schools and other educational intitutions across the globe. We may say that it's the most famous PIC development system in the world. We asked ourselves what we can do to make such a great board even greater. And we made some brilliant changes. We focused all of our creativity and knowledge into making a revolutionary new design, unlike any previous version of the board. We now present you with the new version 7 that brings so much more, and we hope that you will be thrilled with your new board, just as we are.

EasyPIC[™] development Team

Four Connectors for each port Amazing connectivity

EasyPIC v7 is all about connectivity. Having four 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 over 250 microcontrollers. You will need it, whether you are a professional or a beginner.



3.3V and 5V power supply Dual Power Supply

EasyPIC 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 mikroBUS 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

PIC18F45K22 is the new default microcontoller!

Until now, EasyPICTM development boards were equipped with PIC16® as the default chip. Now, for the first time we are giving you more power than ever before. PIC18F45K22 is the new default chip of EasyPIC v7! It has 16 MIPS operation, 64K bytes of linear program memory, 3896 bytes of linear data memory, and support for a wide range of power supply from 1.8V to 5V. It's loaded with great modules: 36 General purpose I/O pins, 30 Analog Input pins (AD), Digital-To-Analog Converter (DAC), support for Capacitive Touch Sensing using Charge Time Measurement Unit (CTMU), three 8-bit timers and four 16-bit timers. It also has pair of each: SPI, I²C, CCP and Comparator modules.

- More power than ever before
- 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

~85mA when all peripheral modules are disconnected



board dimensions

266 x 220mm (10.47 x 8.66 inch)



weight ~445g (0.981 lbs)

Package contains



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, thus making the board capable of supporting both 5V and 3.3V microcontrollers. Power supply unit can be powered in two different ways: with **USB power supply**, and using external adapters via adapter connector (CN31) or additional screw terminals (CN30). External adapter voltage levels must be in range of 9-32V DC and 7-23V AC. Use jumper 16 to specify which power source you are using, and jumper 15 to specify whether you are using 5V or 3.3V microcontroller. Upon providing the power using either external adapter, or USB power source, you can turn the board on using SWITCH 1 (Figure 3-1).

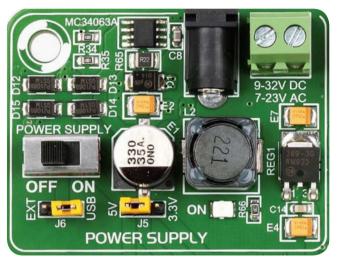


Figure 3-1: Dual power supply unit of EasyPIC v7

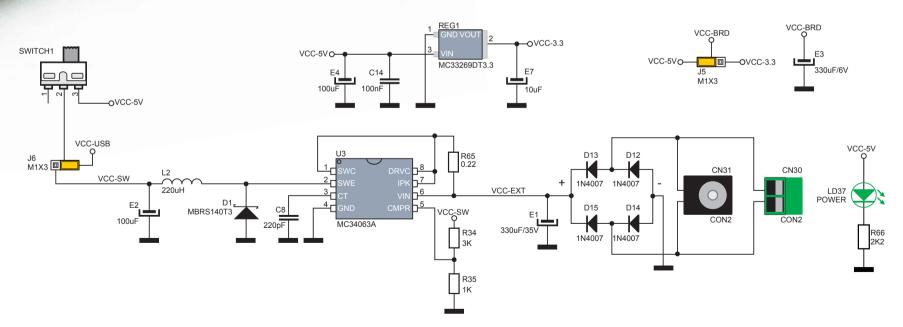


Figure 3-2: Dual power supply unit schematics





Smart engineering of EasyPIC v7 development board allowed us to support both 3.3V and 5V microcontrollers on a single board, which is more then 250 devices.

Power supply:

via DC connector or screw terminals (7V to 23V AC or 9V to 32V DC), or via USB cable (5V DC) **Power consumption:** up to 600mA (depending on how many on-board modules are currently active)

How to power the board?

1. With USB cable



To power the board with USB cable, place jumper **I6** in USB position, and place jumper 15 in 5V or 3.3V position, depending on which microcontroller you are using. 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 J6 jumper to **EXT** position

To power the board via adapter connector, place jumper **I6** in EXT position, and place jumper **I5** in 5V or 3.3V position, depending on which microcontroller you are using. You can then plug in the adapter cable as shown on images 🚯 and 🐴, and turn the power switch ON.

3. With laboratory power supply



EasyPIC

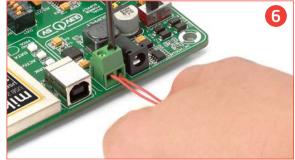
Set J6 jumper to EXT position

To power the board using screw terminals, place jumper J6 in EXT position, and place jumper J5 in 5V or 3.3V position, depending on which microcontroller you are using. You can then plug in the adapter cable as shown on images (5) and (6), and turn the power switch ON.









Supported microcontrollers

Board contains eight DIP sockets: DIP40, DIP28, DIP18A, DIP18B, DIP20, DIP14, DIP8 and support for PIC10F MCUs. With dual power supply and smart on-board mikroProg, board is capable of programming over 250 microcontrollers from PIC10F, PIC12F, PIC16F, PIC16Enh, PIC18F, PIC18FJ and PIC18FK families.

There are two DIP18 sockets for PIC microcontrollers provided on the board - DIP18A and **DIP18B**. Which of these sockets you will use depends solely on the pinout of the microcontroller in use. The EasyPIC v7 development system comes with the PIC18F45K22 microcontroller in a DIP40 package.

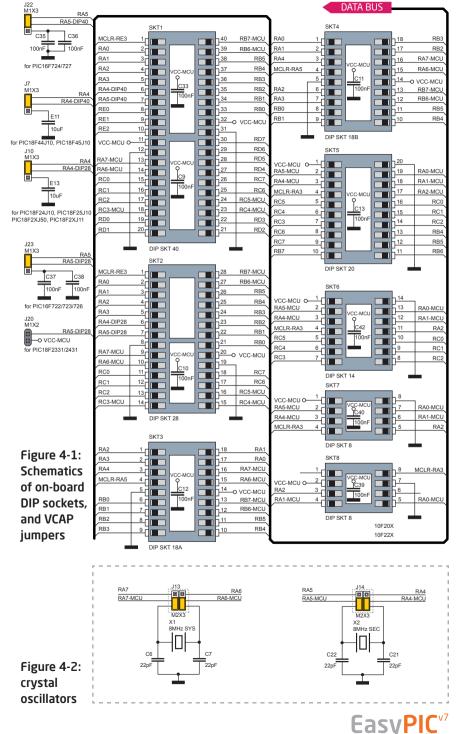
IMPORTANT: When using PIC18F2331 or PIC18F2431 microcontollers it is necessary to place J20 jumper, in order to route VCC power line to RA5 pin (Figure 4-1)

VCAP jumpers explained

Some PIC16F, PIC18FK and all PIC18FI microcontrollers have cores that work on 1.8V-2.5V voltage range, and peripherals that work with 3.3V and 5V voltages. Internally, those microcontrollers have power regulators, which adjust the core voltage levels. In order for those devices to have a stable operation of the core, manufacturer recommends that decoupling capacitive filters should be provided, and connected between specific microcontroller pins designated with VCAP and GND. EasyPIC v7 board provides jumpers which are used for this purpose. Here is list of devices that require jumpers placed in VCAP position:

4-1)	J22	VCAP position when using PIC16F724/16F727	
ure 4	J17	VCAP position for PIC18F44J10 and PIC18F45J10	
see fig	J10	VCAP for PIC18F24J10, PIC18F25J10 PIC18F2XJ50, PIC18F2XJ1	
(Sé	J23	VCAP for PIC16F722, PIC16F723, PIC16F726	
IMPORTANT: If you do not place VCAP jumper for the MCUs that need it,			

need it. you might experience some instabilities in program execution.



How to properly place your microcontroller into the DIP socket?

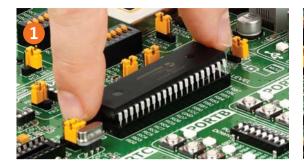


Figure 4-3: Place both ends of microcontroller on the socket so the pins are aligned correctly

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 a half circular cut



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

in the microcontroller DIP packaging maches the cut in the DIP socket. Place one end of the microcontroller into the socket as shown in **Figure 4-3**. Then put the microcontroller slowly down until all the pins thereof



Figure 4-5: Properly placed microcontroller will have equally leveled 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**.

supported MCUs

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

Using crystal oscillators



Figure 4-6: RA6 and RA7 as I/O pins (when using internal oscillator)

RA6 RA7 DIP28

O OSC

Figure 4-7: RA6 and RA7 connected to X1 oscillator

PIC microcontrollers normally use a quartz crystal for the purpose of providing clock frequency. The **EasyPIC v7** provides two sockets for quartz-crystal. Microcontrollers in **DIP18A**, **DIP18B**, **DIP28** and **DIP40** packages use socket **X1 (OSC1)** for quartz-crystal.





Figure 4-8: RA4 and RA5 as I/O pins (when using internal oscillator)

Figure 4-9: RA4 and RA5 connected to X1 oscillator

If you want to use microcontrollers in **DIP8**, **DIP14** and **DIP20** packages, it is necessary to put quartz crystal into socket **X2 (OSC2)**. The value of the quartz-crystal depends on the maximum clock frequency allowed, and your application, and you can always exchange the default 8MHz crystal with another one.



Microcontrollers which are plugged into socket 10F use their own internal oscillator and are not connected to any of the mentioned quartz-crystal sockets.

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 PIC10, PIC12, PIC16, PIC18, devices in a single programmer! It supports over 250 microcontrollers from Microchip[®]. 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.



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 transfered between the programmer and PC software (compiler or mikroProg Suite).



If you use other than the default PIC18F45K22 MCU, you want to make sure that programmer jumpers are placed in proper positions for your microcontroller socket.



MCLR pin selection

Before using the programmer, make sure to set MCLR pin jumpers **J1** and **J2**, so that MCLR line is routed to the correct socket for your microcontroller. If you are using the default **PIC18F45K22**, jumpers are supposed to be set for DIP40, as shown below.

J2 J1 MCLR PIN	J2 J1	J2 J1 MCLR PIN
DIP40,	DIP18A,	DIP20
DIP28	DIP18B	DIP14
		DIP8

Programing lines selection

Jumpers **J8** and **J9** are used to select **PGC** and **PGD** programming lines for your microcontroller. Make sure to place jumpers in the proper position for your socket.

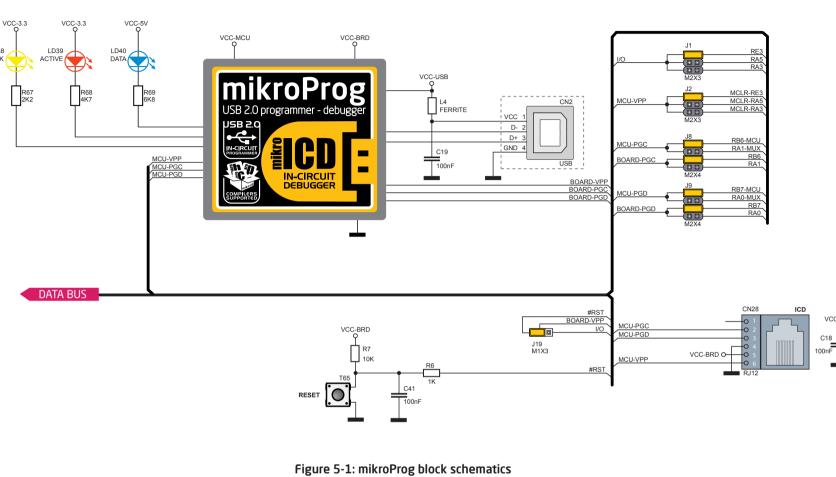
JB JO SOCKET SELECTION	J8 J2
DIP40,	DIP20,
DIP28,	DIP14,
DIP18A,	DIP8
DIP18B	

MCLR pin function

Using jumper **J19** you can specify whether **RST** pin of your microcontroller is connected to the onboard reset circuit, or acts just as I/O pin.

MCLR	MCLR
as	as I/O
MCLR	



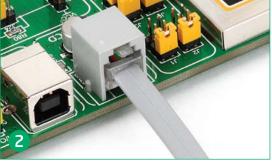


programming

Programming with ICD2/ICD3

EasyPIC v7 is equipped with RJ-12 connector compatibile with Microchip® ICD2® and ICD3® external programmers. This way you can override the on-board mikroProgTM programmer and In-Circuit Debugger, and use other programming tools with the board. But you still have to **set the appropriate jumpers**, as described in the previous page. Insert your ICD programmer cable into connector **CN28**, as shown in images **1** and **2**.





LD38

LINK

Installing programmer drivers

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



DVD://download/ena/software/ development-tools/universal/ mikroproa/mikroproa_for_pic_ drivers_v200.zip

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.

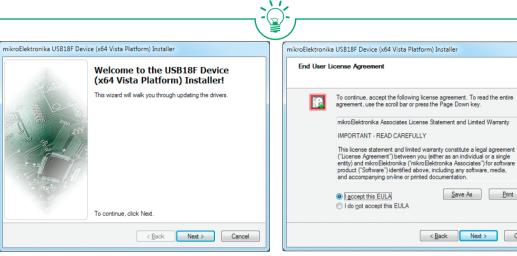


In the opened folder you should be able to locate the driver setup file. Double click on setup file to begin installation of the programmer drivers.



Adobe

USB18PRG-Vistax64.EXE



Step 1 - Start Installation

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



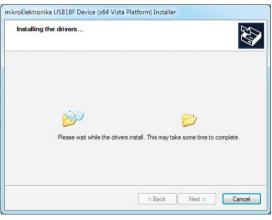
Carefully read End User License Agreement. If you agree with it, click **Next** to procede.

Save As

Next >

Print

Cancel



Step 3 - Installing drivers

Drivers are installed automatically in a matter of seconds.



Step 4 - Finish installation

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



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 the Product DVD:

DVD://download/eng/software/development-tools/universal/ mikroprog/mikroprog_suite_for_pic_v215.zip

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

PIC32MX		010	onfiguration Bits		
CU	•		Clear		
PIC32MX795F	512H ·	CONFIG	User ID	FFFF	
Read	Write		VBUS_ON Selection bit	VBUS_ON pin is controlled by the USB module USBID pin is controlled by the	
		NFO	USB USBID Selection bit	USBID pip is	Presid
Verify	Blank	MCU INFO	Pin Selection hit	the onco by the USB module	
Erase		ž	culernet MII Enable hit	dere Edier riet 1/0 pins	•
CIOSE	Reset		Selection bit	Use the defenter	•
HEX File Optio	one		SRS Select bits	Assign Interrupt Priority 7 to a shadow register set PLL Divide by 256	-
			System PLL Output Clock Divider	PLL Divide by 356	
Load	Save		USB PLL Enable	Disable and human life au	•
Dul			USB PLL Input Divider	12x Divider	•
Reload			PLL Multiplier	24x Multiplier	•
V Load/Save	e CODE		PLL Input Divider	12x Divider	•
V Load/Save	e DATA				•
			Watchdog Timer Enable	WDT Enabled	
COL	DE		Windowed Watchdog Timer	Windowed WDT Enabled	-
DATA	BOOT	P	rogram Memory Size: 512 kB Devi	ce Status: Idle Type Address: Oh Revision	
Opti	ons				
rogress:			(C.)	NikroElektronika	
09	6			DEVELOPMENT TOOLS I COMPILERS I BOOKS	
		1			

Installation wizard - 6 simple steps





Step 1 - Start Installation

mikroProg Suite For PIC	E E
protal For Al Users(recommended)	
Current User	
nkrofrog Sute For PDC	

Step 3 - Install for All users



Step 6 - Finish Installation

mikroProg Suite for PIC v1.10 Setup

Step 2 - Accept EULA and continue



Step 4 - Choose destination folder

Completing the mikroProg Suite for

« Back Brish Concel

PIC v1.10 Setup Wizard mitroProg Suite for PIC v1.10 has been installed on your

Click Finish to close this wizerd.

(Constitution

mikrolCD[™] - 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 mikrolCD[™]?

The on-board **mikroProg**TM programmer supports **mikroICD**TM a highly effective tool for a **Real-Time debugging** on hardware level. The mikroICDTM 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 mikroICDTM 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. mikroICDTM 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 mikroICDTM. 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 mikrolCDTM with your compiler is provided with the EasyPIC v7 package.



Figure 5-2: mikrolCD[™] manual explains debugging thoroughly

EasyPIC

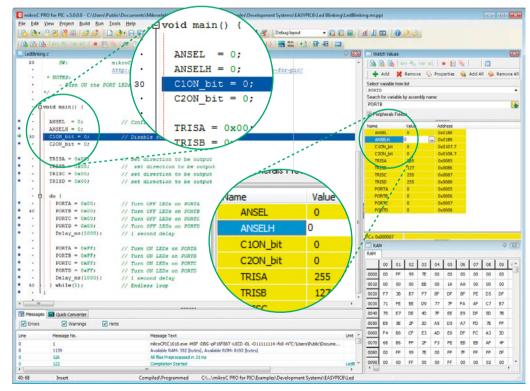


Figure 5-3: mikroC PRO for PIC compiler in debugging view, with SFR registers in Watch Window

E ICD IN-CIRCUIT DEBUGGER

mikrolCDTM commands

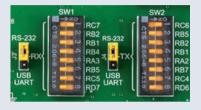
Here is a short overview of which debugging commands are supported in mikroElektronika compilers. You can see what each command does, and what are their shortcuts when you are in debugging mode. It will give you some general picture of what your debugger can do.

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.
R	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.
₽I	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]	Delete's selected breakpoints
1	Jump to interrupt	[F2]	Opens window with available interrupts (doesnt work in mikrolCD mode)



UART via RS-232

Enabling RS-232



In order to enable RS-232 communication, you must set **J3** and **J4** jumpers in the **RS-232** position, and enable desired **RX** and **TX** lines via **SW1** and **SW2** DIP switches. For example, if you want to enable RS-232 connection on UART1 module of the default **PIC18F45K22** chip, you should enable **SW1.1** (RC7) and **SW2.1** (RC6) 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 baudrate, 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**.

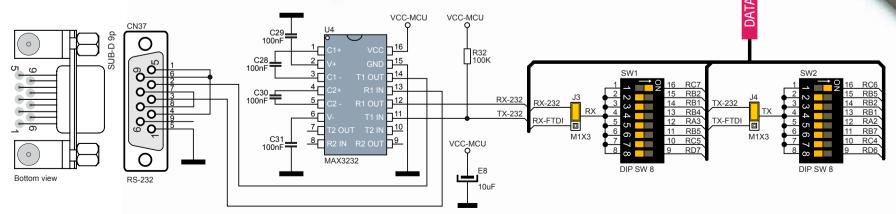


Figure 6-1: RS-232 connection schematics



UART via USB





In order to enable USB-UART communication, you must set **I3** and **[4** jumpers in the **USB-UART** position, and enable desired RX and TX lines via SW1 and SW2 DIP switches. For example, if you want to enable USB-UART connection on UART1 module of the default PIC18F45K22 chip, you should enable SW1.1 (RC7) and SW2.1 (RC6) lines.

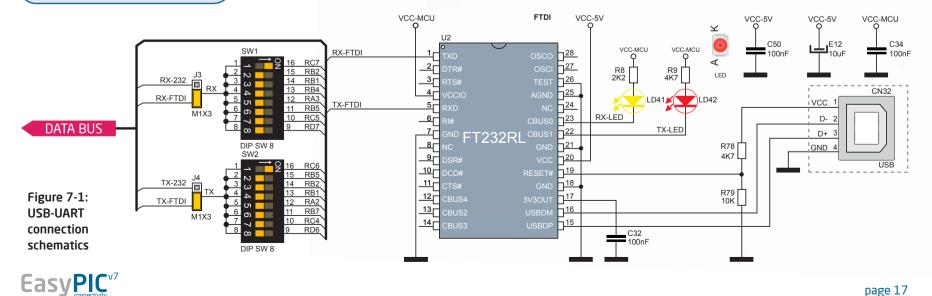
Modern PC computers, laptops and notebooks are no longer equpped with RS-232 connectors and UART controllers. They are nowdays replaced with USB connectors and USB controllers. Still, certain technology enables UART communication to be done over USB connection. Controllers such as FT232RL from FTDI® convert UART signals to the appropriate USB standard. In order to use USB-UART module



DVD://download/ena/software/development-tools/

on EasyPIC v7, you must first install FTDI drivers on your

USB-UART communication is being done through a ailable on Product D FT232RL controller, USB connector (CN32), and microcontroller UART module. To establish this connection, you must put [3 and [4 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 SW1 and SW2.



USB connection

USB is the acronym for Universal Serial Bus. This is a very popular industry standard that defines cables, connectors and protocols used for communication and power supply between computers and other devices. EasyPICv7 contains USB connector (CN4) which enables microcontrollers that support USB communication to establish a connection with the target host (eg. PC, Laptop, etc). Selection of communication lines is done using jumpers **[12** or **[18**, depending on the target microcontroller.

When communication lines are routed from the microcontroller to the USB connector using mentioned jumpers, they are cut off from the rest of the board, and cannot be accessed via PORT

headers. Dedicated USB ON LED signalizes the presense of USB connection, when the USB cable is inserted into the USB connector



Depending on your target microcontroller, USB communication can be enabled on PORTA or PORTC. For PIC18(L)F1XK50 you should put [18 jumpers in the USB position (Figure 8-3). For PIC18Fxx(J)50, PIC18Fxx(J)53, PIC18Fxx(J)55 and PIC18Fxx58 place J12 jumpers in the USB position (Figure 8-2).





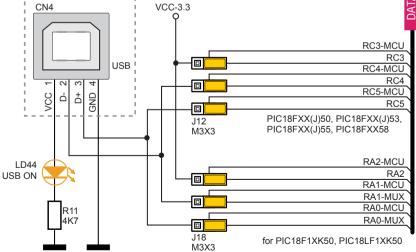
Figure 8-1: USB function disabled

Figure 8-2: USB enabled on PORTC



Figure 8-3: USB enabled on PORTA







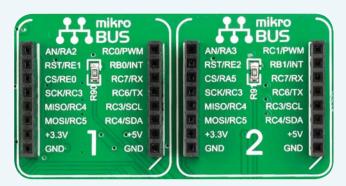
mikroBUS sockets

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 briliant 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 is the first development board in the world to support 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.

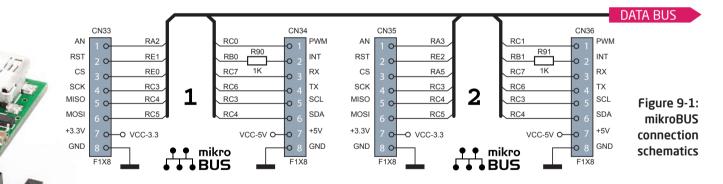
mikroBUS host connector

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 **I**²**C** 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	PWM - PWM output line
RST - Reset pin	INT - Hardware Interrupt line
CS - SPI Chip Select line	RX - UART Receive line
SCK - SPI Clock line	TX - UART Transmit line
MISO - SPI Slave Output line	SCL - I2C Clock line
MOSI - SPI Slave Input line	SDA - I2C Data line
+3.3V - VCC-3.3V power line	+5V - VCC-5V power line
GND - Reference Ground	GND - Reference Ground



Click boards are mikroBUS compatible

mikroElektronika's 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 with additional functionality with literaly zero hardware configuration. Just plug and play.

Input/Output Group

One of the most distinctive features of EasyPIC v7 are it's Input/Output PORT groups. They add so much to the connectivity potential of the board.

Everything is groupped together

It took us a while to realize that having PORT headers, PORT buttons and PORT LEDs next to each other, and groupped together, makes development easier, and the entire EasyPIC 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. Some PORT pins are directly connected to the

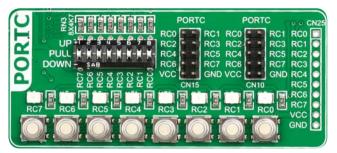


Figure 10-1: I/O group contains 3 PORT headers, tri-state pull up/down DIP switch, buttons and LEDs all in one place

microcontroller, and some that are connected to other on-board modules are enabled via jumpers (for example USB jumpers, **112** and **118**).

data bus

Tri-state pull-up/down DIP switches

Tri-state DIP switches, like SW7 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:

PULL-M-M-M-M-M-M-M-M-M

1. middle position disables both pull-up and pull-down feature from the PORT pin

2. up position connects the resistor PC_LED in pull-up state to the selected pin 3. down position connects the

VCC-MCU

resistor in pull-down state to the

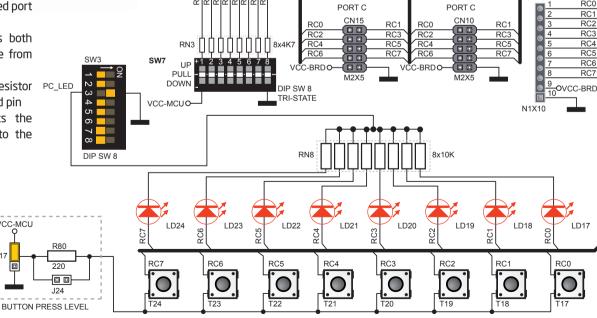


Figure 10-4: Schematic of the single I/O group connected to microcontroller PORTC



CN25

RC0

RC1

RC2

RC3

RC4

RC5

RC6

RC7

connectivity

Figure 10-2: Tri-state **DIP switch on PORTC**

selected PORT pin. DATA BUS

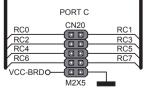
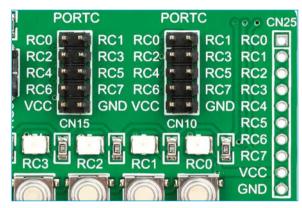


Figure 10-3: Additional PORT header on the left side of the board





With enhanced connectivity as one of the key features of EasyPIC v7, we have provided four connection headers for each PORT. I/O PORT group contains two male IDC10 2x5 headers (like CN10 and CN15 on Figure 10-4). 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.

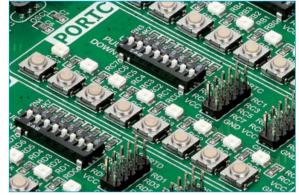
NOTE: Because of it's orientation, header on the left side of the board is not meant for placing accessory boards directly. Instead, use wire jumpers or other ways to establish connection and utilize these pins.

I/O PORT group also contains 1x10 connection pad (like **CN25** on **Figure 10-4**) which can be used for connecting mikroElektronika PROTO boards, or custom user boards.



Figure 10-5: IDC10 male headers enable easy connection with mikroElektronika accessory boards





Buttons



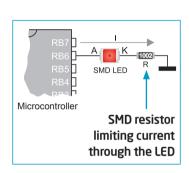
Figure 10-6: Use J17 jumper to specify press level of PORT buttons. Use J24 to disable protective resistor

troller pin when pressing the appropriate push button. The purpose of the protective resistor is to limit maximum current thus preventing a short circuit from occurring. Advanced users may, if needed, disable such resistor using jumper **J24**. By pressing any push button when jumper **J17** is in the VCC-MCU position, a logic one (3.3V or 5V) will be applied to the appropriate microcontroller pin.

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. This button is directly connected to the MCLR pin.

The logic state of all microcontroller digital inputs may be changed using push buttons. lumper **[17** is used to determine the logic state to be applied to the desired microcon-



LEDs

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

current value specified by the manufacturer. A common LED di current the ty current

0.2mA of VCC voltage selection. Board contains 36 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 SW3 (Figure 10-7).

liode voltage is approx	imately 2.5V, while the
nt varies from 0.2mA t	o 20mA, depending or
pe of the LED. The Eas	yPIC v7 board uses low-
nt LEDs with typical o	current consumption of
A or 0.3mA, depending	SW3

		SW	3	
PORTA/E PORTB PORTC PORTD READ-X READ-Y DRIVEA DRIVEB	CTS 206-8 T110		012345678	RA0 RA1 RC0 RC1

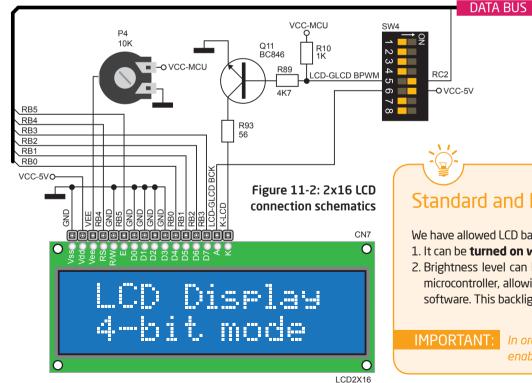
Figure 10-7: SW3.1 through SW3.4 switches are used to enable PORT LEDs



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. EasyPIC 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 CN7 display connector. Board is fitted with uniquely designed plastic display distancer, which allows the LCD module to perfectly and firmly fit into place.

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



ALL MARKER

MPORTANT.



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 backlight LED anode
- LED- Connection with the backlight LED cathode

Standard and PWM-driven backlight

We have allowed LCD backlight to be enabled in two different ways: 1. It can be **turned on with full brightness** using **SW4.6** switch.

2. Brightness level can be determined **with PWM signal** from the microcontroller, allowing you to write custom backlight controling software. This backlight mode is enabled with **SW4.5** switch.



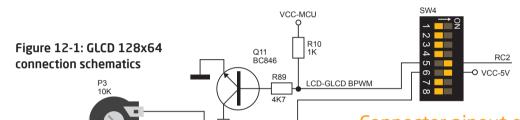
NT: In order to use PWM backlight both SW4.5 and SW4.6 switches must be enabled at the same time.



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. EasyPIC v7 provides the connector and necessary interface for supporting GLCD with resolution of 128x64 pixels, driven by the KS108 or similar display controller. Communication with the display module is done through **CN6** 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 PORTB (control lines) and PORTD (data lines) of the microcontroller sockets. Since PORTB is also used by 2x16 character LCD display, you cannot use both displays simoutaneously. You can control the display contrast using dedicated potentiometer **P3**. Full brightness display backlight can be enabled with **SW4.5** switch, and PWM-driven backlight with **SW4.6** switch.





- 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 backlight LED anode
- LED- Connection with the backlight LED cathode

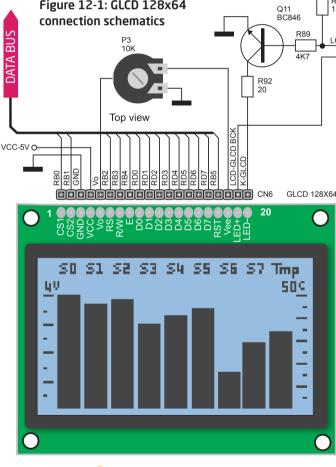


As for LCD, we have allowed GLCD backlight to be enabled in two different ways:

- It can be **turned on with full brightness** using **SW4.6** switch.
 Brightness level can be determined **with PWM signal** from the
- microcontroller, allowing you to write custom backlight controling software. This backlight mode is enabled with **SW4.5** switch.



IMPORTANT: In order to use PWM backlight both **SW4.5** and **SW4.6** switches must be enabled at the same time.



EasyPIC"

Touchpanel controller

Touchpanel 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 touchpanels can be used as an input devices. EasyPIC v7 is equipped with touchpanel controller and connector for **4-wire resistive touchpanels**. 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. Touchpanel is ment to be mounted onto the GLCD display.

Correctly placing the touchpanel cable into the connector



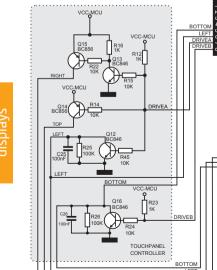
Figure 13-1: Put Touch Panel flat cable in the connector



Figure 13-2: Use a tip of your finger to push it inside



Figure 13-3: Now place GLCD with Touch panel into GLCD socket



TOUCHPANEL CONTROLLER

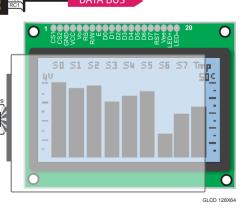


Figure 13-4: Touch Panel controller and connection schematics

Enabling Touch panel

Touchpanel is enabled using SW3.5, SW3.6, SW3.7 and SW3.8 switches. They connect READ-X and READ-Y lines of the touchpanel with RAO and RA1 analog inputs, and DRIVEA and DRIVEB with RCO and RC1 digital outputs on microcontroller sockets. Do not connect additional boards or otherwise interfere with these lines while you use touchpanel, because you may corrupt the results of the readings and get inacurate touch coordinates.

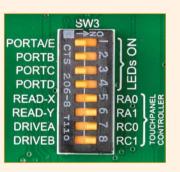


Figure 13-5: Turn on switches 5 through 8 on SW3 to enable Touch panel controller

CONNE

D1°

5

PANE

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. EasyPIC 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 it's **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 simoutaneously. 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 PORTD, and digit select lines are connected to RAO-RA3 lines on the microcontroller sockets.

Enabling the display

Street R28 Treit Black R29 Treet Black R30 Treet Black R30

To enable digit select lines for the 4-digit 7-segment display you have to turn on **SW4.1**, **SW4.2**, **SW4.3** and **SW4.4** switches. Digit select lines are connected to **RA0 - RA3** pins on the microcontroller sockets, while data lines are connected to **RD0 - RD8** pins. For proper operation of the display, make sure that you do not place additional boards on mentioned lines which can interfere with logic levels used in your application.



Figure 14-1: Turn on switches 1 through 4 on SW4 to enable 4-digit 7-seg display

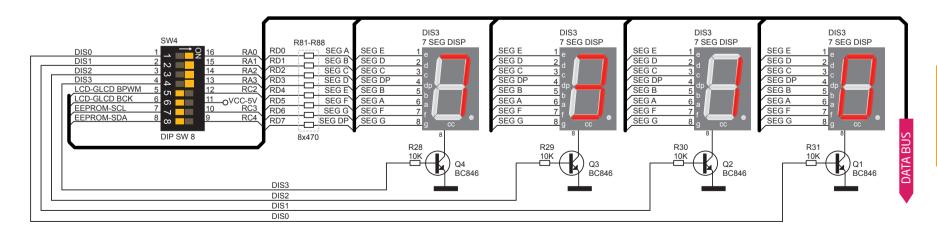
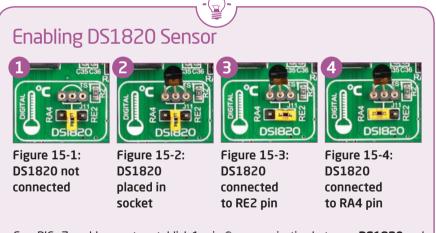


Figure 14-2: 4-digit 7-segment display schematics



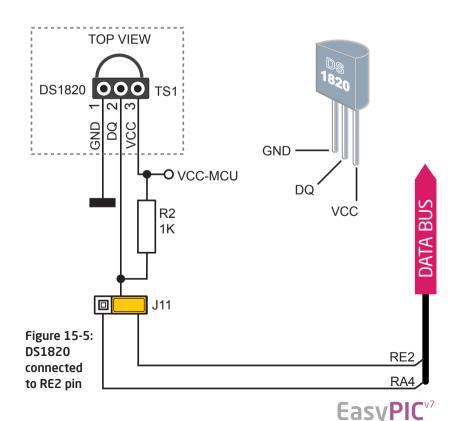
DS1820 - Digital Temperature Sensor

DS1820 is a digital temperature sensor that uses **1-wire**® **interface** for it's operation. It is capable of measuring temperatures within the range of -55 to 128°C, and provides ±0.5°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 transfered 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. EasyPIC v7 provides a separate socket (**TS1**) for the DS1820. Communication line with the microcontroller is selected with a **J11** jumper.



nsie

EasyPIC v7 enables you to establish 1-wire[®] communication between **DS1820** and the microcontroller over **RA4** or **RE2** microcontroller pins. The selection of either of those two lines is done using **J11** 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 accidently connect the sensor the other way, it may 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 (except those in 1-wire network) uses the selected line, because it may interfere with the data.

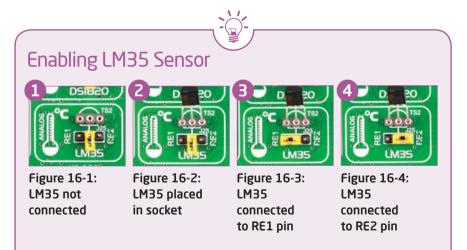


LM35 - Analog Temperature Sensor

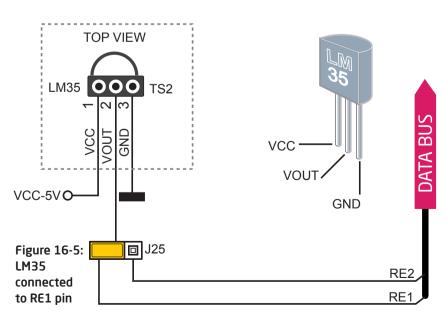
The **LM35** is a low-cost precision integrated-circuit temperature sensor, whose output voltage is linearly proportional to the Celsius (Centigrade) temperature. The LM35 thus has an advantage over linear temperature sensors calibrated in ° Kelvin, as the user is not required to subtract a large constant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

EasyPIC

to obtain convenient Centigrade scaling. The LM35 does not require any external calibration or trimming to provide typical accuracies of $\pm 14^{\circ}$ C at room temperature and $\pm 34^{\circ}$ C over a full -55 to +150°C temperature range. It has a linear + 10.0 mV/°C scale factor and less than 60 µA current drain. As it draws only 60 µA from its supply, it has very low self-heating, less than 0.1°C in still air. EasyPIC v7 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 a **J25** jumper.



EasyPIC v7 enables you to get analog readings from the LM35 sensor using **RE1** or **RE2** microcontroller pins. The selection of either of those two lines is done using **J25** 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 accidently 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.



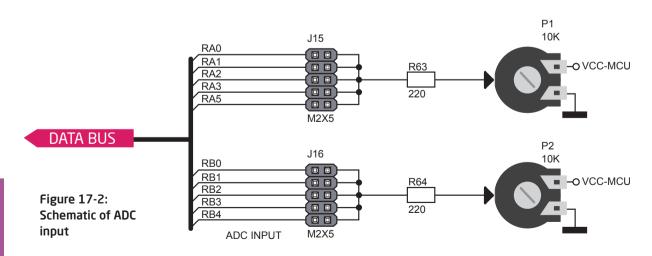
modules

10 050

ADC inputs

Digital signals have two discrete states, which are decoded as high and low, and interpreted as logic 1 and logic 0. Analog signals, on the other hand, are continuous, and can have any value within defined range. A/D converters are specialized circuits which can convert analog signals (voltages) into a digital representation, usually in form of an **integer number**. The value of this number is **lineary dependent** on the input voltage value. Most microcontrollers nowdays internally have A/D converters connected to one or more input pins. Some of the most important parameters of A/D converters are **conversion** time and resolution. Conversion time determines how fast can an analog voltage be represented in form of a digital number. This is an important parameter if you need fast data acquisition. The other parameter is resolution. Resolution represents the number of discrete steps that supported voltage range can be divided into. It determines the sensitivity of the A/D converter. Resolution is represented in maximum number of bits that resulting number occupies. Most PIC microcontrollers have 10-bit resolution, meaning that maximum value of conversion can be represented with 10 bits, which converted to integer is 2^{10} =1024. This means that supported voltage range, for example from 0-5V, can be devided into 1024 discrete steps of about 4.88mV.

EasyPIC v7 provides an interface in form of two potentiometers for simulating analog input voltages that can be routed to any of the 10 supported analog input pins.



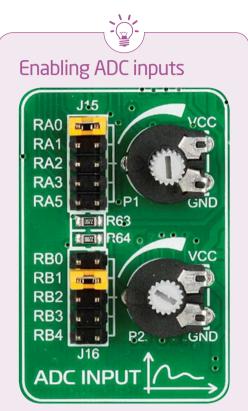


Figure 17-1: use J15 and J16 jumpers to connect analog input lines with potentiometers P1 and P2

In order to connect the output of the potentiometer **P1** to **RA0**, **RA1**, **RA2**, **RA3** or **RA5** analog microcontroller inputs, you have to place the jumper **J15** in the desired position. If you want to connect potentiometer **P2** to any of the **RB0**-**RB4** analog microcontroller inputs, place jumper **J16** in the desired position. By moving the potentiometer knob, you can create voltages in range from **GND** to **VCC**.

EasyPIC^{v7}

I²C EEPROM

Enabling I²C EEPROM

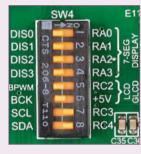


Figure 18-1: Activate SW4.7 and SW4.8 switches to enable pull-up resistors in I²C lines of Serial EEPROM.

In order to connect I²C EEPROM to the microcontroller you must enable SW4.7 and SW4.8 switches, as shown on Figure 18-1. 4K7 pull-up resistors necessary for I²C communication are already provided on **SDA** and **SCL** lines once switches are turned on. Prior to using EEPROM in your application, make sure to disconnect other peripherials, LEDs and additional pull-up or pull-down resistors from the RC3 and RC4 communication lines that could interfere with the data signals and cause data corruption.

EEPROM is short for **Electrically Erasable** Programmable Read Only Memory. It is usually a secondary storage memory in devices containing data that is retained even if the device looses power supply. EEPROMs come with parallel or serial interface to the master device. Because of the ability to alter single bytes of data, EEPROM devices are used to store personal preference and configuration data in a wide spectrum of consumer, automotive, telecommunication, medical, industrial, and PC applications.

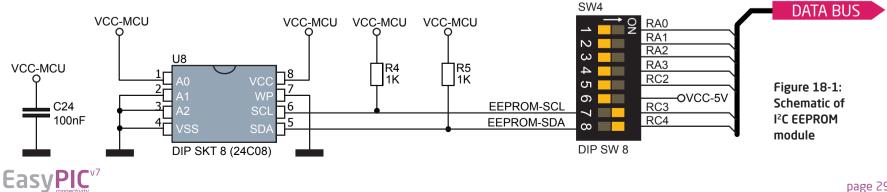
EasyPIC v7 supports serial EEPROM which uses I²C communication interface and has 1024 bytes of available memory. Board 0 contains socket for serial EEPROMs in DIP8 packaging, so you can easily exchange it with different memory size EEPROM IC. EEPROM itself supports single byte or 16-byte (page) write and read operations. Data rates are dependent of power supply voltage, and go up to **1 MHz** with 5V power supply, and 400 kHz for 3.3V power supply.

What is I²C?

I²C is a multi-master serial single-ended bus that is used to attach low-speed peripherals to computer or embedded systems. I²C uses only two bidirectional open-drain lines, Serial Data Line (SDA) and Serial Clock (SCL), pulled up with resistors. Data and clock lines are driven with a master device. Up to 112 slave devices can be connected to the same bus. Each slave must have a unique address.

ADC INPUT C

IZC EEPRON



JC

05

05

N

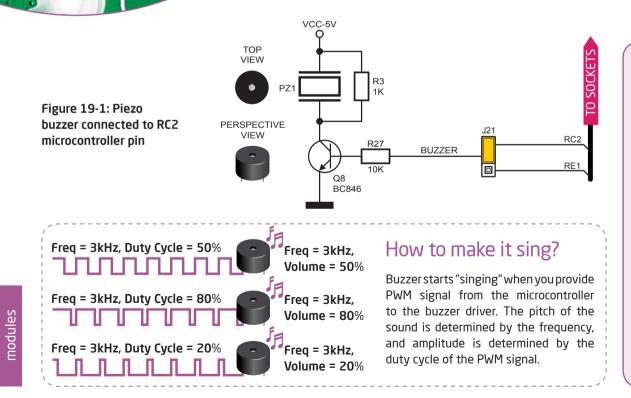
Piezo Buzzer

Piezoelectricity is the charge which accumulates in certain solid materials in response to mechanical pressure, but also providing the charge to the piezoelectric material causes it to physically deform. One of the most widely used applications of piezoelectricity is the production of sound PIEZO BUZZER generators, called piezo buzzers. Piezo buzzer is an electric component that comes in different shapes and sizes, which can be used to create sound waves when provided with analog electrical signal. EasyPIC v7 comes with piezo buzzer which can be connected either to RC2 or **RE1** microcontroller pins, which is determined by the position of [21 jumper. Buzzer is driven by transistor 08 (Figure 19-1). Microcontrollers can create sound by generating a PWM (Pulse Width Modulated) signal - a square wave signal, which is nothing more than

a sequence of logic zeros and ones. Frequency of the square signal determines the pitch of the generated sound, and duty cycle of the signal can be used to increase or decrease the volume in the range from 0% to 100% of the duty cycle. You can generate PWM signal using hardware capture-compare module, which is usually available in most microcontrollers, or by writing a custom software which emulates the desired signal waveform.

Supported sound frequencies

Piezo buzzer's resonant frequency (where you can expect it's best performance) is **3.8kHz**, but you can also use it to create sound in the range between **2kHz** and **4kHz**.



Enabling Piezo Buzzer

In order to use the on-board Piezo Buzzer in your application, you first have to connect the transistor driver of piezo buzzer to the appropriate microcontroller pin. This is done using jumper **J21**. You can place the jumper in two positions, thus connecting the buzzer driver to either **RC2** or **RE1** microcontroller pin.





Additional GNDs

EasyPIC v7 contains three **GND pins** located in three different sections of the board, which allow you to easily connect oscilloscope GND reference when you monitor signals on microcontroller pins, or signals of on-board modules.



GND is located in the cross section between DIP18 and DIP14 sockets

GND is located between PORTD I/O group and DIP28 socket.



Figure 20-1: 3 oscilloscope GND pins are conveniently positioned so each part of the board can be reached with an oscilloscope probe

RB7 ŘB60 RB5 RB4 RB3 RB2 RB1 RB1 RB0 VCC GND

EasyPIC^v

What's Next?

You have now completed the journey through each and every feature of EasyPIC v7 board. You got to know it's modules, organization, supported microcontrollers, programmer and debugger. Now you are ready to start using your new board. We are suggesting several steps which are probably the best way to begin. We invite you to join thousands of users of EasyPIC[™] brand. You will find very useful projects and tutorials and can get help from a large ecosystem of users. Welcome!

Compiler

You still don't have an appropriate compiler? Locate **PIC compiler** that suits you best on the **Product DVD** provided with the package:

DVD://download/eng/software/compilers/

Choose between mikroC, mikroBasic and mikroPascal and download fully functional demo version, so you can begin building your PIC applications.



Projects

Once you have chosen your compiler, and since you already got the board, you are ready to start writing your first projects. We have equipped our compilers with dozens of examples that demonstrate the use of each and every feature of the EasyPIC[™] board, and all of our accessory boards as well. This makes an excellent starting point for your future projects. Just load the example, read well commented code, and see how it works on hardware. Browse through the compiler **Examples** path to find the following folder:

\Development Systems\EASYPIC_v7

Community

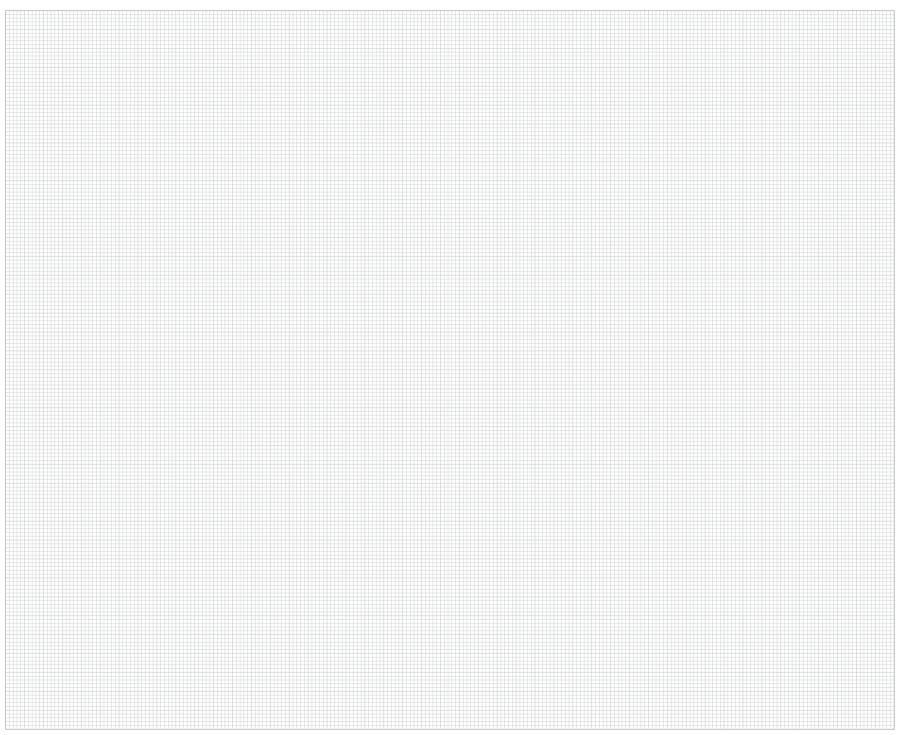
If you want to find answers to your questions on many interesting topics we invite you to visit our forum at http://www.mikroe.com/forum and browse through more than 150 thousand posts. You are likely to find just the right information for you. On the other hand, if you want to download free projects and libraries, or share your own code, please visit the Libstock website. With user profiles, you can get to know other programmers, and subscribe to receive notifications on their code.

http://www.libstock.com/

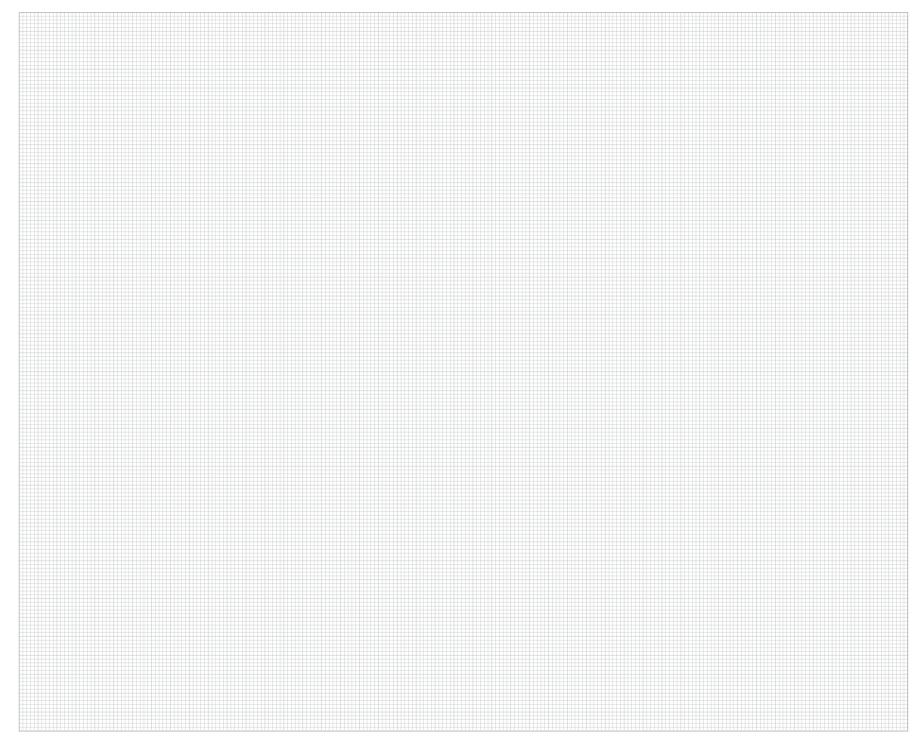
We all know how important it is that we can rely on someone in moments when we are stuck with our projects, facing a deadline, or when we just want to ask a simple, basic question, that's pulling us back for a while. We do understand how important this is to people and therefore our Support Department is one of the pillars upon which our company is based. MikroElektronika offers Free Tech Support to the end of product lifetime, so if something goes wrong, we are ready and willing to help!

http://www.mikroe.com/esupport/











DISCLAIMER

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

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

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

HIGH RISK ACTIVITIES

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

TRADEMARKS

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

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



If you want to learn more about our products, please visit our website at www.mikroe.com If you are experiencing some problems with any of our products or just need additional information, please place your ticket at www.mikroe.com/en/support If you have any questions, comments or business proposals, do not hesitate to contact us at office@mikroe.com