



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

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

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



Contact us

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

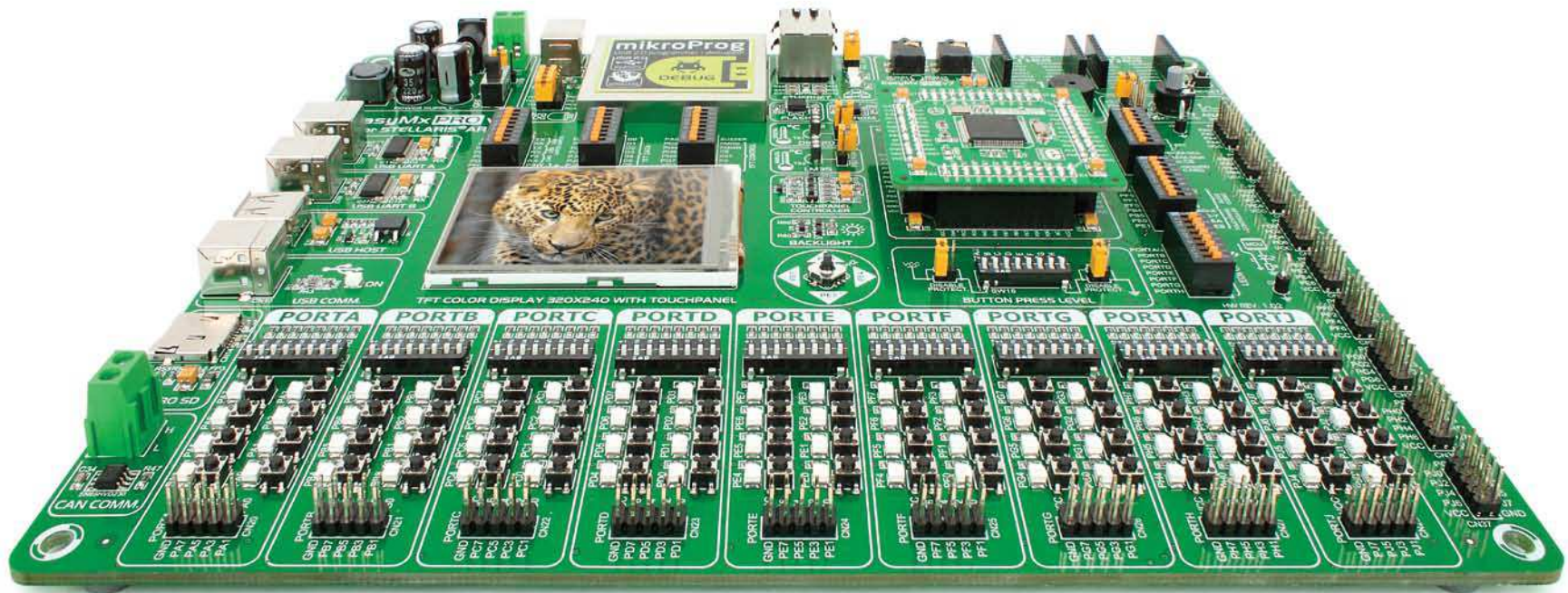
Email & Skype: info@chipsmall.com Web: www.chipsmall.com

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



EasyMx PRO™ v7

for Stellaris® ARM®



270

microcontrollers supported
The ultimate Stellaris® board



Many on-board modules
Multimedia peripherals



Easy-add extra boards
mikroBUS™ sockets



Two connectors for each port
Amazing Connectivity



Fast USB 2.0 programmer and
In-Circuit Debugger

To our valued customers

EasyMx PRO™ v7 is our first development board for Stellaris® ARM® devices. We have put all of our knowledge that we gained in the past 10 years of developing embedded systems into it's design, functionality and quality. It may be our first ARM® Cortex™-M3 and M4 development board, but it sure looks and feels like it's our 7th.

You made the right choice. But the fun has only just begun!



Nebojsa Matic,
Owner and General Manager
of mikroElektronika

Table of contents

Introduction

Introduction	04
It's good to know	05

Power Supply

Power supply	06
--------------------	----

Supported MCUs

Default MCU card	08
Other supported MCU cards	11

Programmer/debugger

On-board programmer	12
List of MCUs supported with mikroProg™	13
Installing programmer drivers	14
Programming software	15
Hardware Debugger	16

Connectivity

Input/Output Group	18
mikroBUS™ sockets	20
Click™ Boards	21

Communication

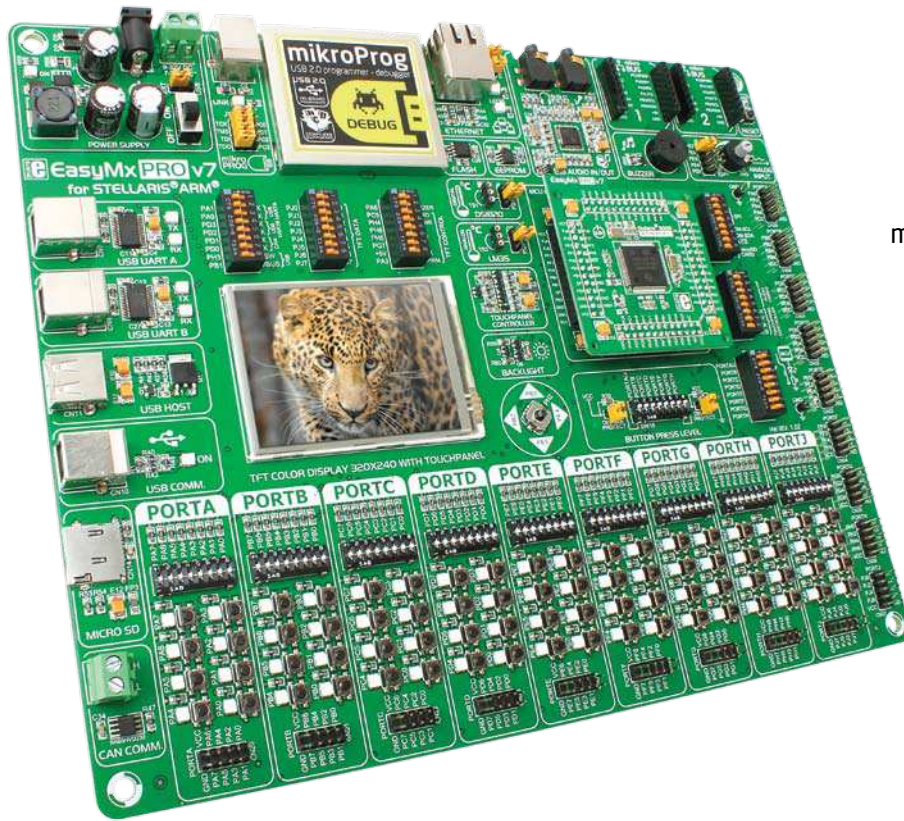
USB-UART A	22
USB-UART B	23
USB host communication	24
USB device communication	25
Ethernet communication	26
CAN communication	27

Multimedia

Audio Input/Output	28
microSD card slot	29
TFT display 320x240 pixels	30
Touch panel controller	31
Navigation switch	32
Piezo Buzzer	33

Other Modules

DS1820 - Digital Temperature Sensor	34
LM35 - Analog Temperature Sensor	35
Serial Flash Memory	36
I2C EEPROM	37
ADC inputs	38
Additional GNDs	39



Introduction

ARM® Cortex™-M3 and Cortex™-M4 are increasingly popular microcontrollers. They are rich with modules, with high performance and low power consumption, so creating a development board the size of EasyMx PRO™ v7 for Stellaris® was really a challenge. We wanted to put as many peripherals on the board as possible, to cover many internal modules. We have gone through a process of fine tuning the board performance, and used 4-layer PCB to achieve maximum efficiency. Finally, it had met all of our expectations, and even exceeded in some. We present you the board which is powerful, well organized, with on-board programmer and debugger and is ready to be your strong ally in development.

EasyMx PRO™ v7 for Stellaris® development Team

Two connectors for each port Amazing connectivity

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



Everything is already here mikroProg™ on board

Powerful on-board mikroProg™ programmer and hardware debugger can program and debug over 270 Stellaris® ARM® microcontrollers. You will need it, whether you are a professional or a beginner.



Ready for all kinds of development Multimedia peripherals

TFT 320x240 with touch panel, stereo mp3 codec, audio input and output, navigation switch and microSD card slot make a perfect set of peripherals for multimedia development.



For easier connections mikroBUS™ support

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

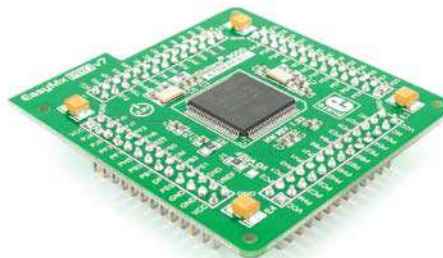


It's good to know

LM3S9B95 is the default microcontroller

LM3S9B95 is the default chip of EasyMx PRO™ v7. It belongs to ARM® Cortex™-M3 family. It has **80MHz** operation, **256K bytes** of linear program memory, **96K bytes** of linear data memory. It has integrated **Ethernet controller** with PHY, **USB** (OTG, Host, Device), up to 65 General purpose **I/O pins**, 5 **16-bit timers**, 16 Analog Input pins (**AD**), 3 **UARTs**, internal Real time clock (**RTC**), a pair of each: **I2C**, **SPI** and **CAN** controllers. It also contains 3 **analog comparators**, 16 **digital comparators**. It is pre loaded with **StellarisWare®** libraries and bootloader in ROM.

- Great choice for both beginners and professionals
- Rich with modules
- 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
~137mA 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 EasyMx PRO™ v7 board in antistatic bag



3 USB cable



4 User Manuals and Board schematic



5 DVD with examples and documentation

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 specialized **MC33269DT3.3** power regulator which creates VCC-3.3V power supply, thus making the board capable of supporting 3.3V microcontrollers. Power supply unit can be powered in three different ways: with **USB power supply (CN5)**, using external adapters via adapter connector (**CN16**) or additional screw terminals (**CN15**). External adapter voltage levels must be in range of **9-32V DC and 7-23V AC**. Use jumper **J1** to specify which power source you are using. Upon providing the power using either external adapters or USB power source you can turn on power supply by using **SWITCH 1 (Figure 3-1)**. Power **LED ON (Green)** will indicate the presence of power supply.

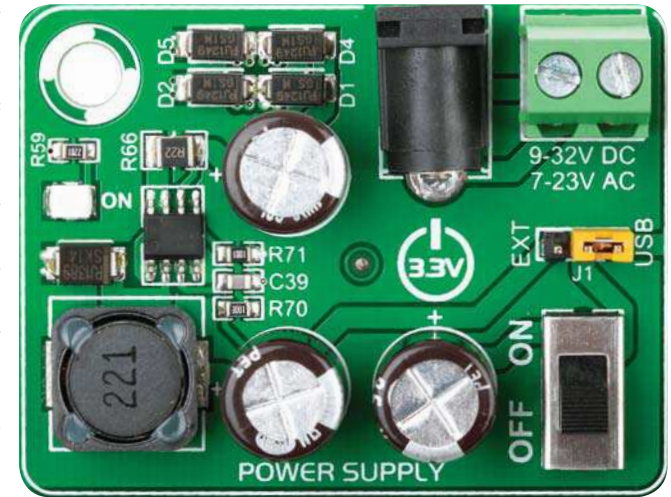


Figure 3-1: Power supply unit of EasyMx PRO™ v7 for Stellaris®

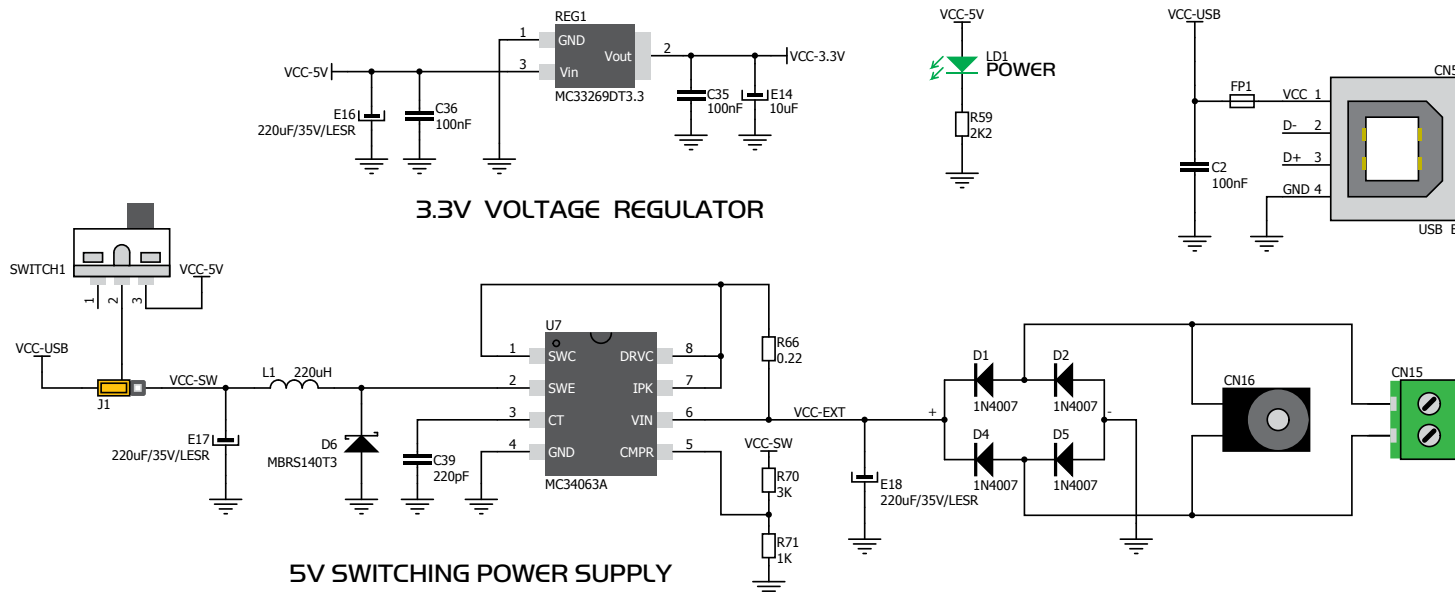


Figure 3-2: Power supply unit schematic



Board power supply creates stable 3.3V necessary for operation of the microcontroller and all on-board modules.

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 J1 jumper to USB position

To power the board with USB cable, place jumper J1 in USB position. You can then plug in the USB cable as shown on images 1 and 2, and turn the power switch ON.

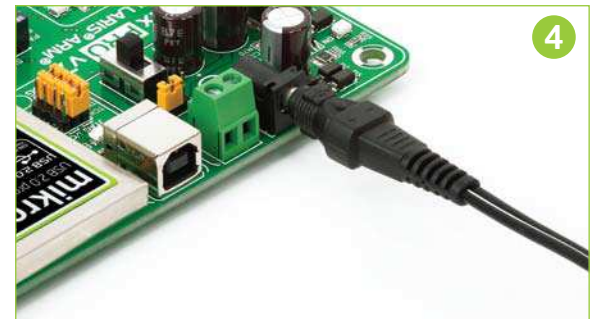
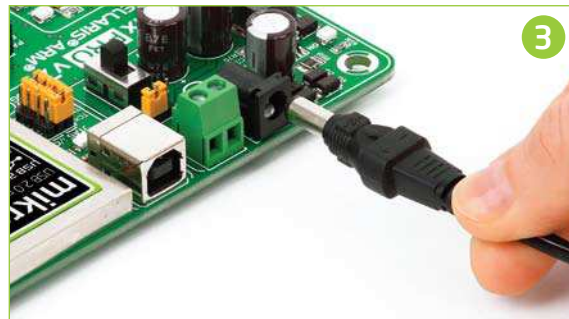


2. Using adapter



Set J1 jumper to EXT position

To power the board via adapter connector, place jumper J1 in EXT position. You can then plug in the adapter cable as shown on images 3 and 4, and turn the power switch ON.

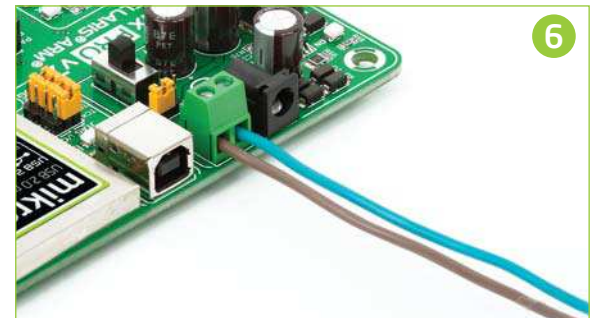
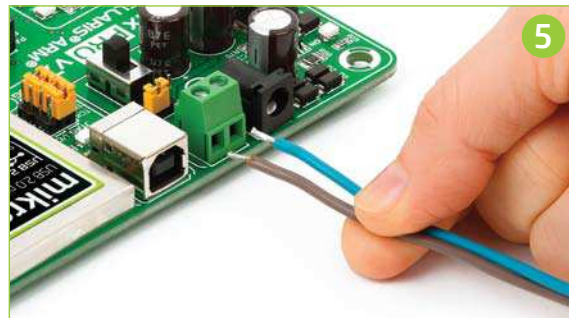


3. With laboratory power supply



Set J1 jumper to EXT position

To power the board using screw terminals, place jumper J1 in EXT 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.



Default MCU card

Microcontrollers are supported using specialized MCU cards containing 104 pins, which are placed into the on-board female MCU socket. There are several types of cards which cover all microcontroller families of Stellaris® Cortex™-M3, as well as Cortex™-M4. The **Default MCU card** that comes with the EasyMx PRO™ v7 package

is shown on **Figure 4-1**. It contains **LM3S9B95** microcontroller with on-chip peripherals and is a great choice for both beginners and professionals. After testing and building the final program, this card can also be taken out of the board socket and used in your final device.

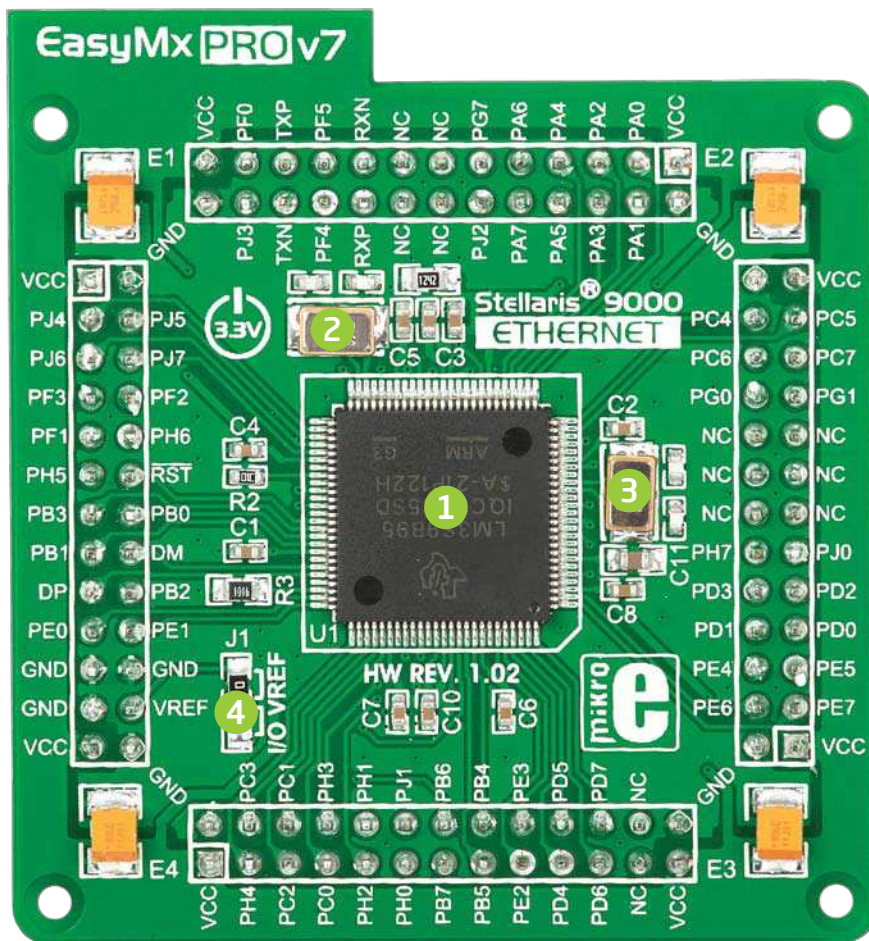


Figure 4-1: Default MCU card with LM3S9B95

- 1 **LM3S9B95** is the default chip of EasyMx PRO™ v7 for Stellaris®. It belongs to ARM® Cortex™-M3 family. It has **80MHz** operation, **256K bytes** of linear program memory, **96K bytes** of linear data memory. It has integrated **Ethernet controller** with PHY, **USB** (OTG, Host, Device), up to 65 General purpose **I/O pins**, five **16-bit timers**, 16 Analog Input pins (**AD**), three **UARTs**, internal Real time clock (**RTC**), a pair of each: **I2C**, **SPI** and **CAN** controllers. It also contains 3 **analog comparators**, 16 **digital comparators**. It is pre loaded with **StellarisWare®** libraries and bootloader in ROM.
- 2 **8MHz crystal oscillator**. We carefully chose the most convenient crystal value that provides clock frequency which can be used directly, or with the PLL multipliers to create higher MCU clock value.
- 3 **25MHz crystal oscillator**. This crystal oscillator is connected to internal Ethernet module.
- 4 **VREF jumper**. This jumper determines whether **PB6** pin is used as voltage reference for A/D converter, or it is used as general purpose I/O pin. Jumper is soldered to **VREF** position by default.

Please note that if VREF jumper is soldered to I/O position Touch Panel controller will not operate correctly, because it uses voltage from this pin as a reference for A/D conversion.

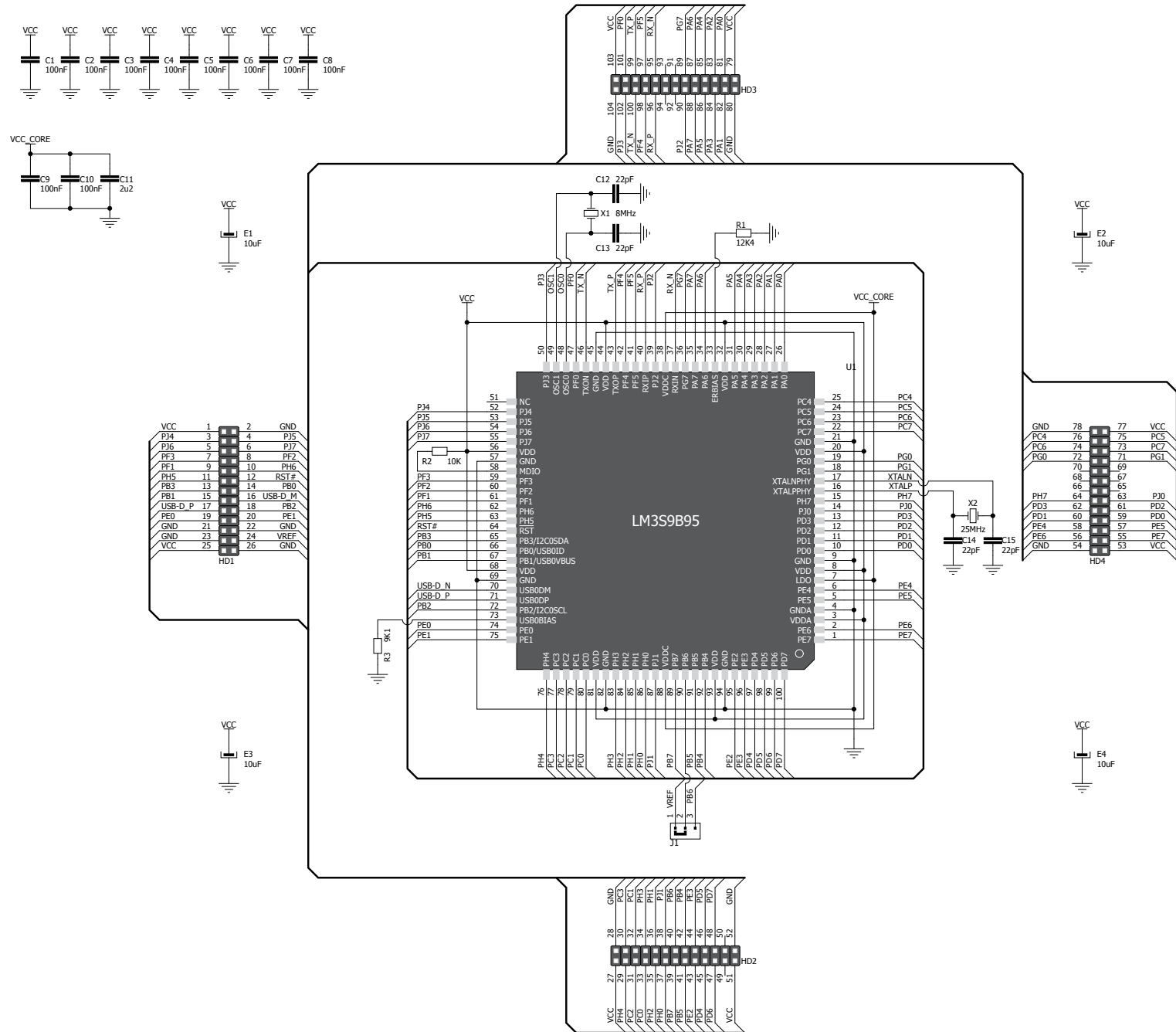


Figure 4-2: Default MCU card schematic

How to properly place your MCU card into the socket?

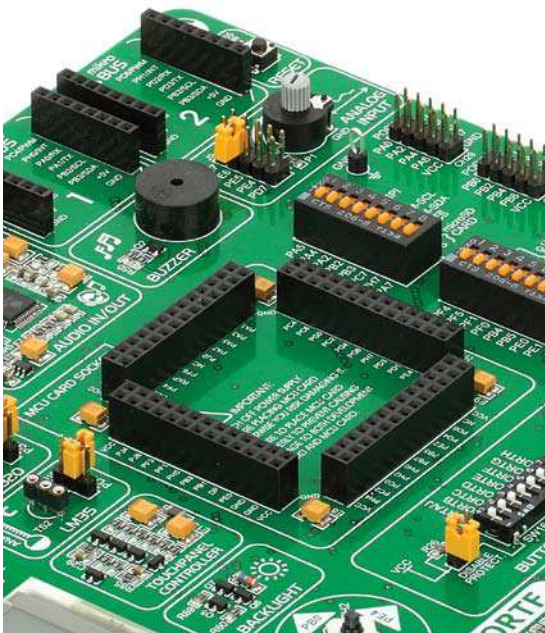
Before you plug the microcontroller card into the socket, make sure that the **power supply is turned off**. Images below show how to correctly plug the MCU card. First make sure that MCU card orientation matches the silkscreen outline on the

EasyMx PRO™ v7 for Stellaris® board MCU socket. Place the MCU card over the socket so each male header is properly aligned with the female socket as shown in **Figure 4-4**. Then put the MCU card slowly down until all the pins match the socket.

Check again if everything is placed correctly and press the MCU card until it is completely plugged into the socket as shown in **Figure 4-5**. If done correctly all pins should be fully inserted. Only now you can turn on the power supply.

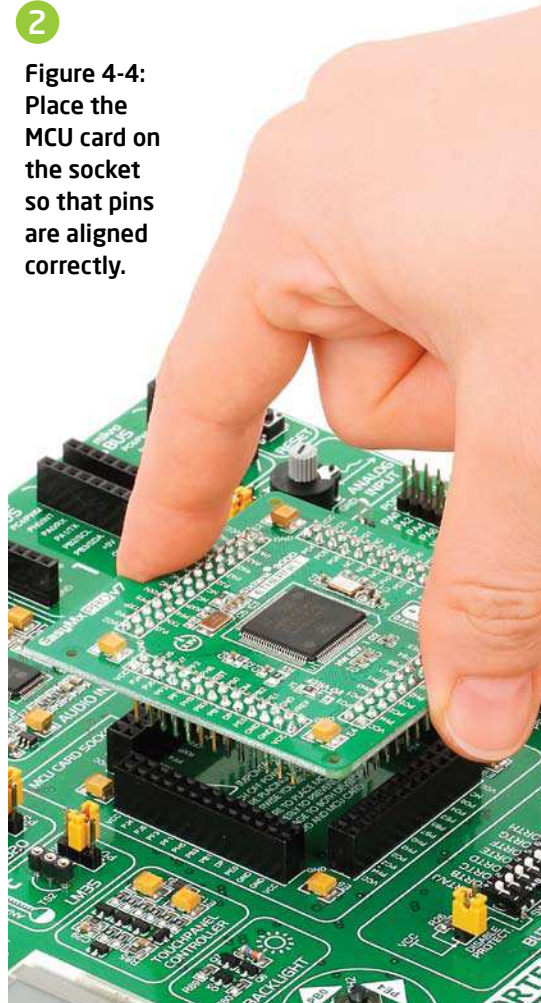
1

Figure 4-3: On-board MCU socket has silkscreen markings which will help you to correctly orient the MCU card before inserting.



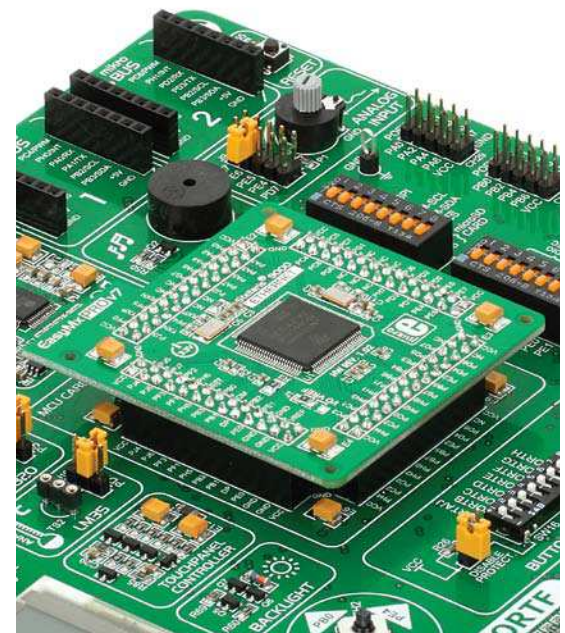
2

Figure 4-4: Place the MCU card on the socket so that pins are aligned correctly.



3

Figure 4-5 Properly placed MCU card.

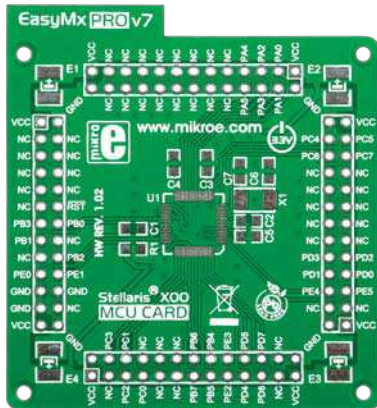


Other supported MCU cards

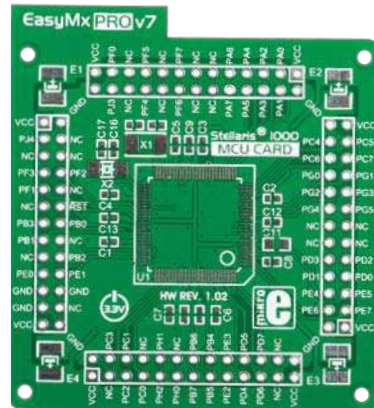
mikroElektronika currently offers total of two populated MCU cards: one with default **LM3S9B95** Cortex™-M3 microcontroller and one with **LM4F232H5QD** Cortex™-M4 microcontroller. You can also purchase empty PCB cards that you can populate on your own and solder any supported microcontroller you need in your development. There are total of seven empty PCB cards available. This way your EasyMx PRO™ v7

for Stellaris® board becomes truly flexible and reliable tool for almost any of your ARM® projects. MCU cards can also be used in your final devices. For complete list of currently available MCU cards, please visit the board webpage:

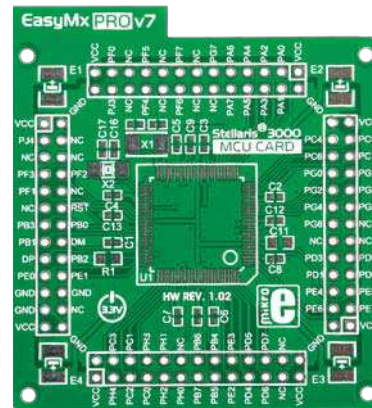
<http://www.mikroe.com/eng/products/view/792/easymx-pro-v7-for-stellaris-arm/>



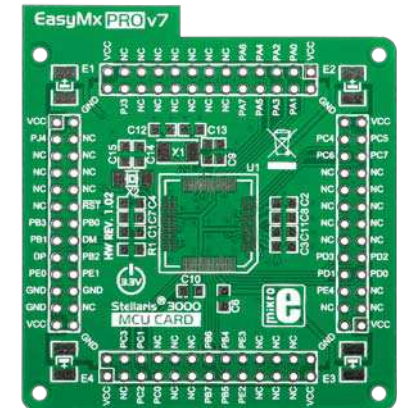
Empty MCU card for 48-pin Stellaris® X00 series MCUs



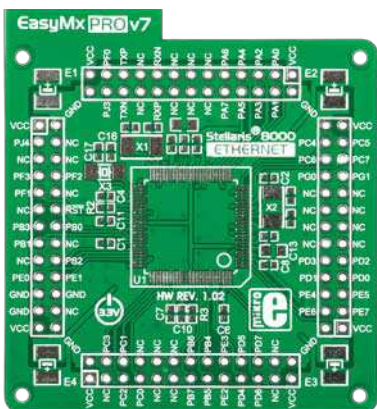
Empty MCU card for 100-pin Stellaris® 1000 series MCUs



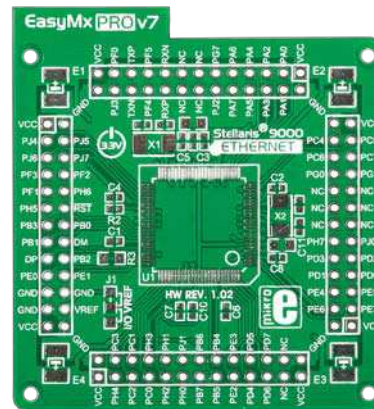
Empty MCU card for 100-pin Stellaris® 3000 series MCUs



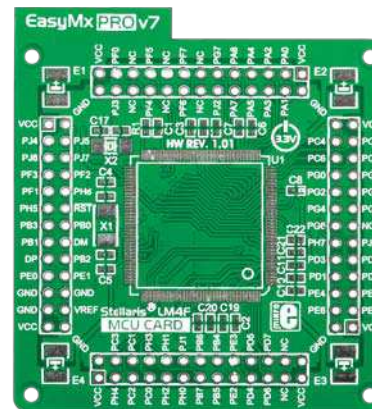
Empty MCU card for 64-pin Stellaris® 3000 series MCUs



Empty MCU card for 100-pin Stellaris® 8000 series MCUs



Empty MCU card for 100-pin Stellaris® 9000 series MCUs



Empty MCU card for 144-pin Stellaris® LM4F series MCUs



MCU card for Stellaris® LM4F series with LM4F232H5QD

On-board programmer

What is mikroProg™?

mikroProg™ is a fast programmer and debugger which is based on TI ICDI debugger. Smart engineering allows mikroProg™ to support over 270 ARM® Cortex™-M3 and Cortex™-M4 devices from Stellaris® in a single programmer. It also features a powerful debugger which will be of great help in your development. Outstanding performance and easy operation are among its top features.

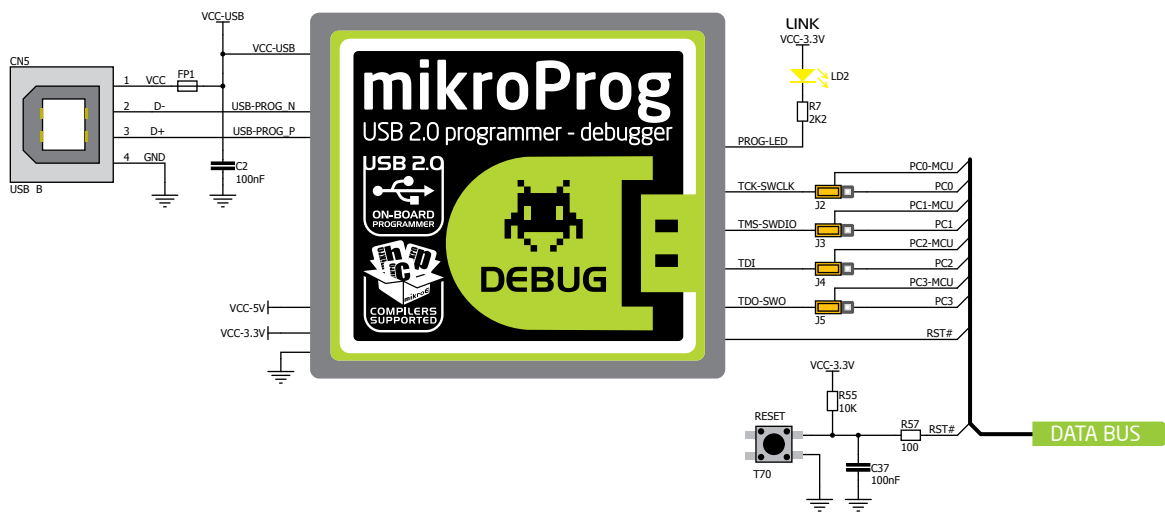
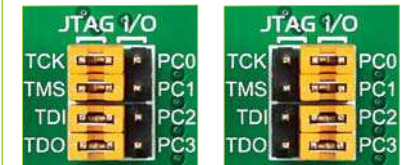


Figure 5-1: mikroProg™ block schematic

Enabling mikroProg™



Four jumpers below the programmer USB connector are used to specify whether programming lines should be connected to programmer or used as general purpose I/Os. If placed in **JTAG/SWD position**, jumpers connect PC0-PC3 pins to TCK, TMS, TDI and TDO programming lines respectively and are cut off from the rest of the board.

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 programmer drivers
 - Install mikroProg Suite™ for ARM® software

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

Stellaris® Cortex™ -M3 microcontrollers supported with mikroProg™

LM3S101	LM3S1811	LM3S1R21	LM3S2948	LM3S5651	LM3S5K31	LM3S6422	LM3S817	LM3S9B96
LM3S102	LM3S1816	LM3S1R26	LM3S2950	LM3S5652	LM3S5K36	LM3S6432	LM3S818	LM3S9L97
LM3S1110	LM3S1850	LM3S1W16	LM3S2965	LM3S5656	LM3S5P31	LM3S6537	LM3S828	LM3S9BN2
LM3S1133	LM3S1911	LM3S1Z16	LM3S2B93	LM3S5662	LM3S5P36	LM3S6610	LM3S8530	LM3S9BN5
LM3S1138	LM3S1918	LM3S2110	LM3S2D93	LM3S5732	LM3S5P3B	LM3S6611	LM3S8538	LM3S9BN6
LM3S1150	LM3S1937	LM3S2139	LM3S2U93	LM3S5737	LM3S5P51	LM3S6618	LM3S8630	LM3S9C97
LM3S1162	LM3S1958	LM3S2276	LM3S300	LM3S5739	LM3S5P56	LM3S6633	LM3S8730	LM3S9CN5
LM3S1165	LM3S1960	LM3S2410	LM3S301	LM3S5747	LM3S5R31	LM3S6637	LM3S8733	LM3S9D81
LM3S1332	LM3S1968	LM3S2412	LM3S308	LM3S5749	LM3S5R36	LM3S6730	LM3S8738	LM3S9D90
LM3S1435	LM3S1B21	LM3S2432	LM3S310	LM3S5752	LM3S5T36	LM3S6753	LM3S8930	LM3S9D92
LM3S1439	LM3S1C21	LM3S2533	LM3S315	LM3S5762	LM3S5U91	LM3S6911	LM3S8933	LM3S9D95
LM3S1512	LM3S1C26	LM3S2601	LM3S316	LM3S5791	LM3S5Y36	LM3S6918	LM3S8938	LM3S9D96
LM3S1538	LM3S1C58	LM3S2608	LM3S317	LM3S5951	LM3S600	LM3S6938	LM3S8962	LM3S9DN5
LM3S1601	LM3S1D21	LM3S2616	LM3S328	LM3S5956	LM3S601	LM3S6950	LM3S8970	LM3S9DN6
LM3S1607	LM3S1D26	LM3S2620	LM3S3634	LM3S5B91	LM3S608	LM3S6952	LM3S8971	LM3S9G97
LM3S1608	LM3S1F11	LM3S2637	LM3S3651	LM3S5C31	LM3S610	LM3S6965	LM3S8C62	LM3S9GN5
LM3S1620	LM3S1F16	LM3S2651	LM3S3654	LM3S5C36	LM3S6100	LM3S6C11	LM3S8G62	LM3S9L71
LM3S1621	LM3S1G21	LM3S2671	LM3S3739	LM3S5C51	LM3S611	LM3S6C65	LM3S9781	LM3S9U81
LM3S1625	LM3S1G58	LM3S2678	LM3S3748	LM3S5C56	LM3S6110	LM3S6G11	LM3S9790	LM3S9U90
LM3S1626	LM3S1H11	LM3S2730	LM3S3749	LM3S5D51	LM3S612	LM3S6G65	LM3S9792	LM3S9U92
LM3S1627	LM3S1H16	LM3S2739	LM3S3826	LM3S5D56	LM3S613	LM3S800	LM3S9971	LM3S9U95
LM3S1635	LM3S1J11	LM3S2776	LM3S3J26	LM3S5D91	LM3S615	LM3S801	LM3S9997	LM3S9U96
LM3S1637	LM3S1J16	LM3S2793	LM3S3N26	LM3S5G31	LM3S617	LM3S808	LM3S9B81	
LM3S1651	LM3S1N11	LM3S2911	LM3S3W26	LM3S5G36	LM3S618	LM3S811	LM3S9B90	
LM3S1751	LM3S1N16	LM3S2918	LM3S3Z26	LM3S5G51	LM3S628	LM3S812	LM3S9B92	
LM3S1776	LM3S1P51	LM3S2939	LM3S5632	LM3S5G56	LM3S6420	LM3S815	LM3S9B95	

Stellaris® Cortex™ -M4 microcontrollers supported with mikroProg™

LM4F110B2QR	LM4F111C4QR	LM4F112H5QC	LM4F120H5QR	LM4F122C4QC	LM4F130E5QR	LM4F132C4QC	LM4F230H5QR	LM4F232H5QD
LM4F110C4QR	LM4F111E5QR	LM4F112H5QD	LM4F121B2QR	LM4F122E5QC	LM4F130H5QR	LM4F132E5QC	LM4F231E5QR	
LM4F110E5QR	LM4F111H5QR	LM4F120B2QR	LM4F121C4QR	LM4F122H5QC	LM4F131C4QR	LM4F132H5QC	LM4F231H5QR	
LM4F110H5QR	LM4F112C4QC	LM4F120C4QR	LM4F121E5QR	LM4F122H5QD	LM4F131E5QR	LM4F132H5QD	LM4F232E5QC	
LM4F111B2QR	LM4F112E5QC	LM4F120E5QR	LM4F121H5QR	LM4F130C4QR	LM4F131H5QR	LM4F230E5QR	LM4F232H5QC	

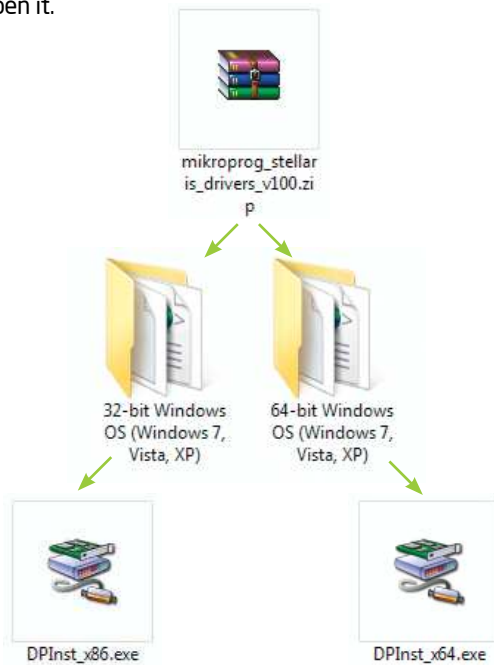
Installing programmer drivers

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



DVD://download/eng/software/
development-tools/arm/stellaris/
mikroprog/mikroprog_stellaris_
drivers_v100.zip

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

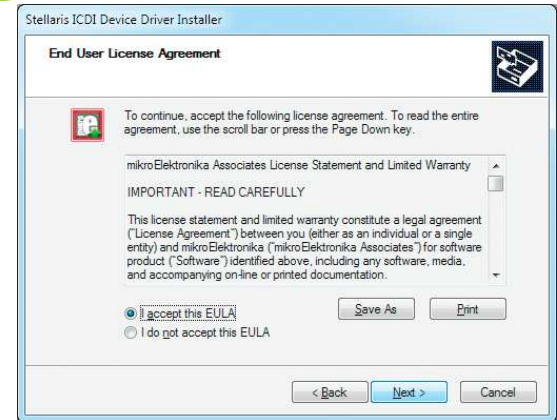


In the opened folder you should be able to locate the driver setup file. Double click on setup file to begin installation of the programmer drivers.



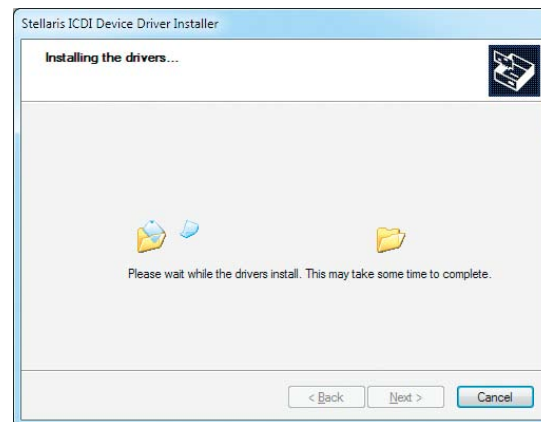
Step 1 - Start Installation

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



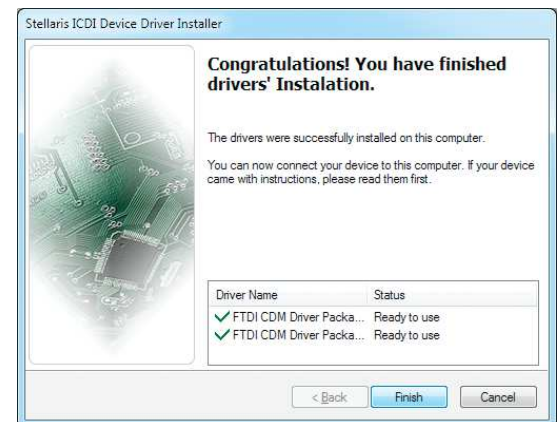
Step 2 - Accept EULA

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



Step 3 - Installing drivers

Drivers are installed automatically in a matter of seconds.



Step 4 - Finish installation

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

Programming software

mikroProg Suite™ for ARM®

On-board mikroProg™ programmer requires special programming software called **mikroProg Suite™ for ARM®**. This software is used for programming all of supported microcontroller families with ARM® Cortex™-M3 and Cortex™-M4 cores. 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/arm/stellaris/mikroprog/mikroprog_suite_for_arm_v110.zip

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



Quick Guide

- 1 Click the **Detect MCU** button in order to recognize the device ID.
- 2 Click the **Read** button to read the entire microcontroller memory. You can click the **Save** button to save it to target HEX file.
- 3 If you want to write the HEX file to the microcontroller, first make sure to load the target HEX file. You can drag-n-drop the file onto the software window, or use the **Load** button to open **Browse dialog** and point to the HEX file location. Then click the **Write** button to begin programming.
- 4 Click the **Erase** button to wipe out the microcontroller memory.

Figure 5-2: mikroProg Suite™ for ARM® window

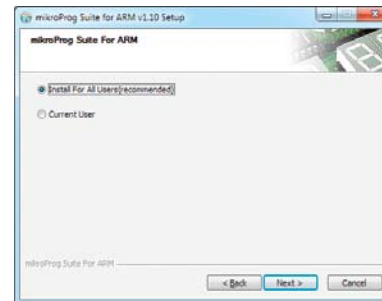
Installation wizard - 6 simple steps



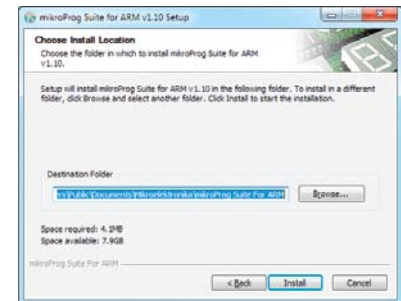
Step 1 - Start Installation



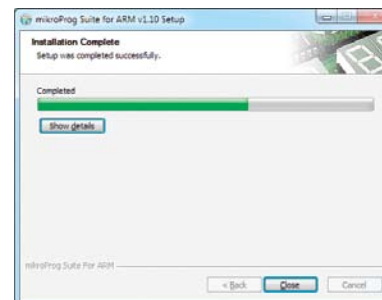
Step 2 - Accept EULA and continue



Step 3 - Install for All users or current user



Step 4 - Choose destination folder



Step 5 - Installation in progress



Step 6 - Finish Installation

Hardware 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 planned. 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 MCU itself. And this latter one is called **hardware debugging**. "hardware" means that it is the real deal - code executes right on the target device.

What is hardware debugger?

The on-board **mikroProg™** programmer supports **hardware debugger** - a highly effective tool for a **Real-Time debugging** on hardware level. The debugger enables you to execute your program on the host Stellaris® microcontroller and view variable values, Special Function Registers (SFR), RAM, CODE and EEPROM memory along with the 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. mikroProg debugger is one of the fastest, and most reliable debugging tools on the market.

Supported Compilers

All MikroElektronika compilers, **mikroC™**, **mikroBasic™** and **mikroPascal™** for ARM® natively support mikroProg™ for Stellaris®, as well as other compilers, including KEIL®, IAR® and CCS®. Specialized 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 ARM®** programming software, as described on **pages 14** and **15**.

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.

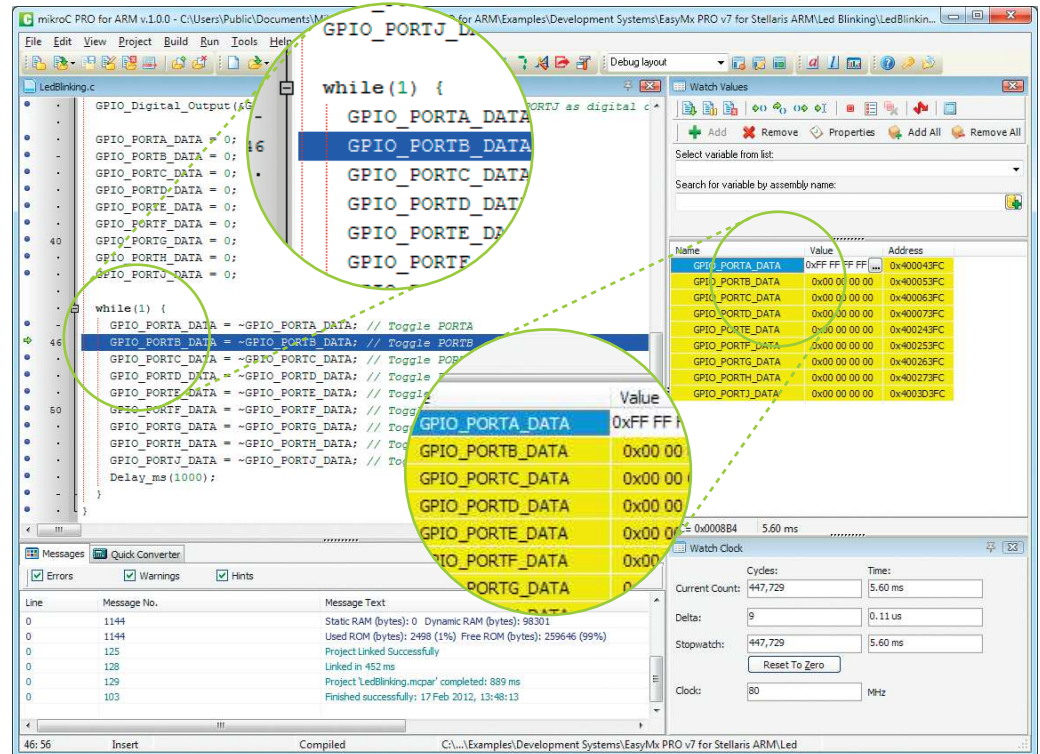













Figure 5-3: mikroC PRO for ARM® compiler in debugging view, with SFR registers in Watch Window



Debugger commands



Here is a short overview of debugging commands which 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]	Delete selected breakpoints
	Jump to interrupt	[F2]	Opens window with available interrupts (doesn't work in hardware debug mode)

Input/Output Group

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

Everything is grouped together

PORT headers, PORT buttons and PORT LEDs next to each other and grouped together. It makes development easier, and the entire EasyMx PRO™ v7 for Stellaris® cleaner and well organized. We have also provided an **additional PORT headers** on the right side of the board, so you can access any pin you want from that side of the board too.



Figure 6-1: I/O group contains PORT header, tri-state pull up/down DIP switch, buttons and LEDs all in one place

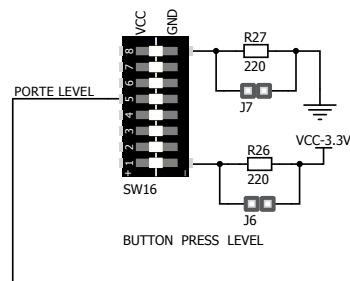
Tri-state pull-up/down DIP switches



Tri-state DIP switches, like **SW5** on **Figure 6-3**, are used to enable 4K7 pull-up or pull-down resistor on any desired port pin. Each of these switches has three states:

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

Figure 6-2: Tri-state DIP switch on PORTE



Button press level tri-state DIP switch is used to determine which logic level will be applied to port pins when buttons are pressed

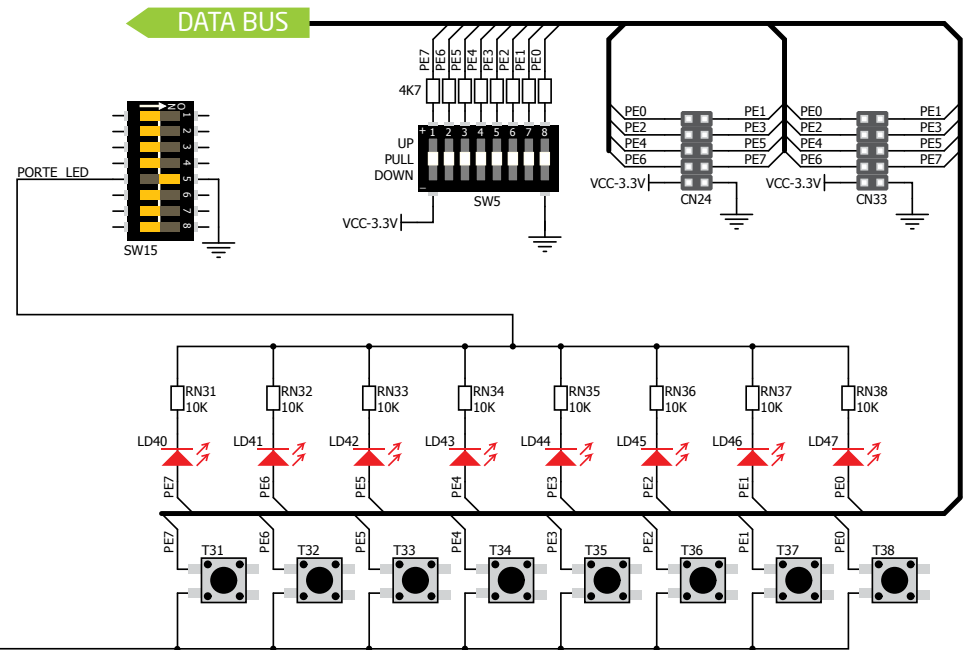


Figure 6-3: Schematic of the single I/O group connected to microcontroller PORTE



Headers

With enhanced connectivity as one of the key features of EasyMx PRO™ v7 for Stellaris®, we have provided **two connection headers for each PORT**. I/O PORT group contains one male IDC10 header (like **CN24** **Figure 6-3**). There is **one more IDC10 header** available on the right side of the board, next to DIP switches (like **CN33** on **Figure 6-3**). These headers can be used to connect accessory boards with IDC10 female sockets.

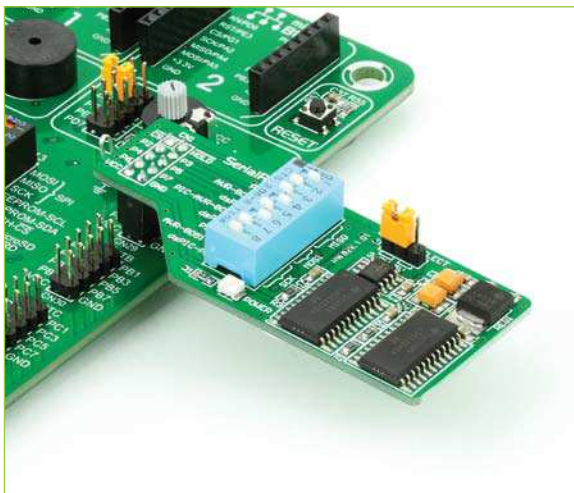
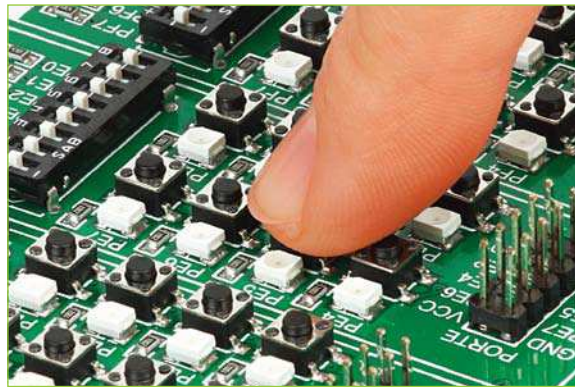


Figure 6-4: IDC10 male headers enable easy connection with mikroElektronika accessory boards



Buttons

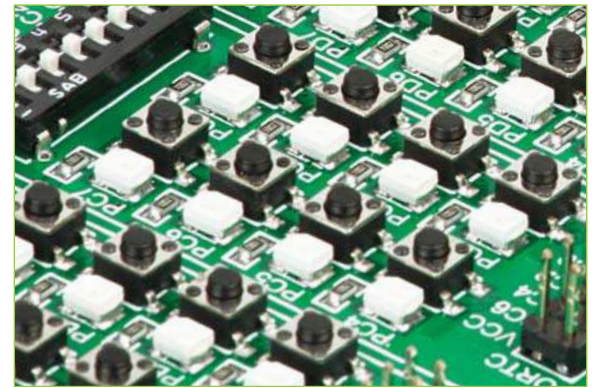
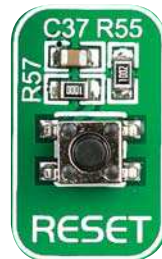


Figure 6-5: Button press level DIP switch (tri-state)

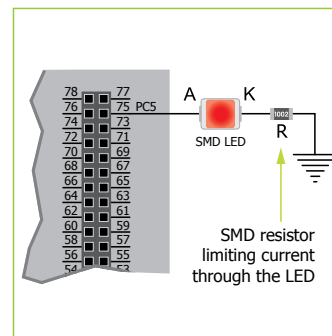
The logic state of all microcontroller digital inputs may be changed using **push buttons**. Tri-state DIP switch **SW16** is available for selecting which logic state will be applied to corresponding MCU pin when button is pressed, for each I/O port separately. If you, for example, place **SW16.5** in **VCC** position, then pressing of any push button in PORTE 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 **J6** and **J7**, which will connect your push buttons directly to VCC or GND. Be aware that doing so you may accidentally damage MCU in case of wrong usage.

Reset Button

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



LEDs



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

specified by the manufacturer. The current varies from 0.2mA to 20mA, depending on the type of the LED and the manufacturer. The EasyMx PRO™ v7 for Stellaris® board uses low-current LEDs with typical current consumption of 0.2mA or 0.3mA. Board contains 72 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 **SW15** (**Figure 6-6**).

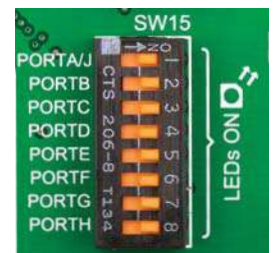


Figure 6-6: SW15.1 through SW15.8 switches are used to enable PORT LEDs

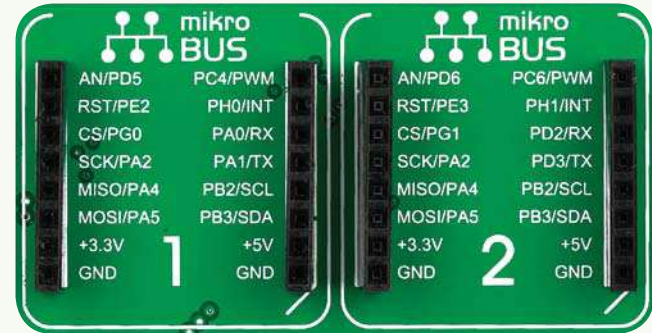
mikroBUS™ sockets

Easier connectivity and simple configuration are imperative in modern electronic devices. Success of the USB standard comes from its simplicity of usage and high and reliable data transfer rates. As we in mikroElektronika see it, Plug-and-Play devices with minimum settings are the future in embedded world too. This is why our engineers have come up with a simple, but brilliant pinout with lines that most of today's accessory boards require, which almost completely eliminates the need of additional hardware settings. We called this new standard the **mikroBUS™**. EasyMx PRO™ v7 for Stellaris® 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

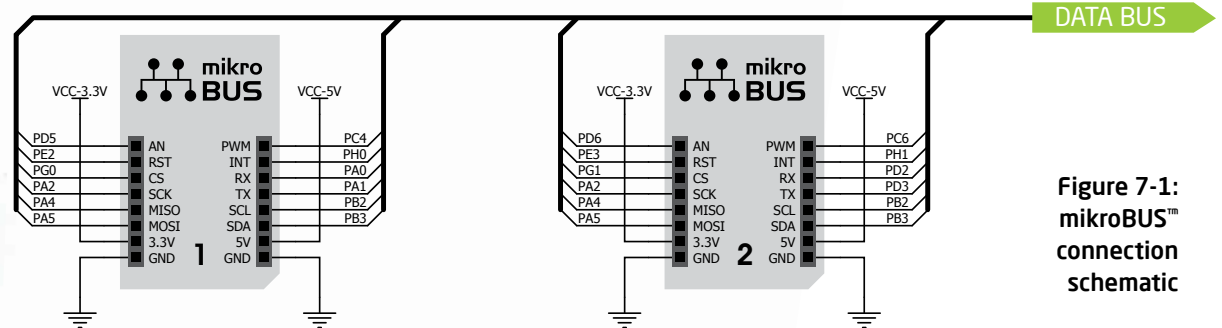
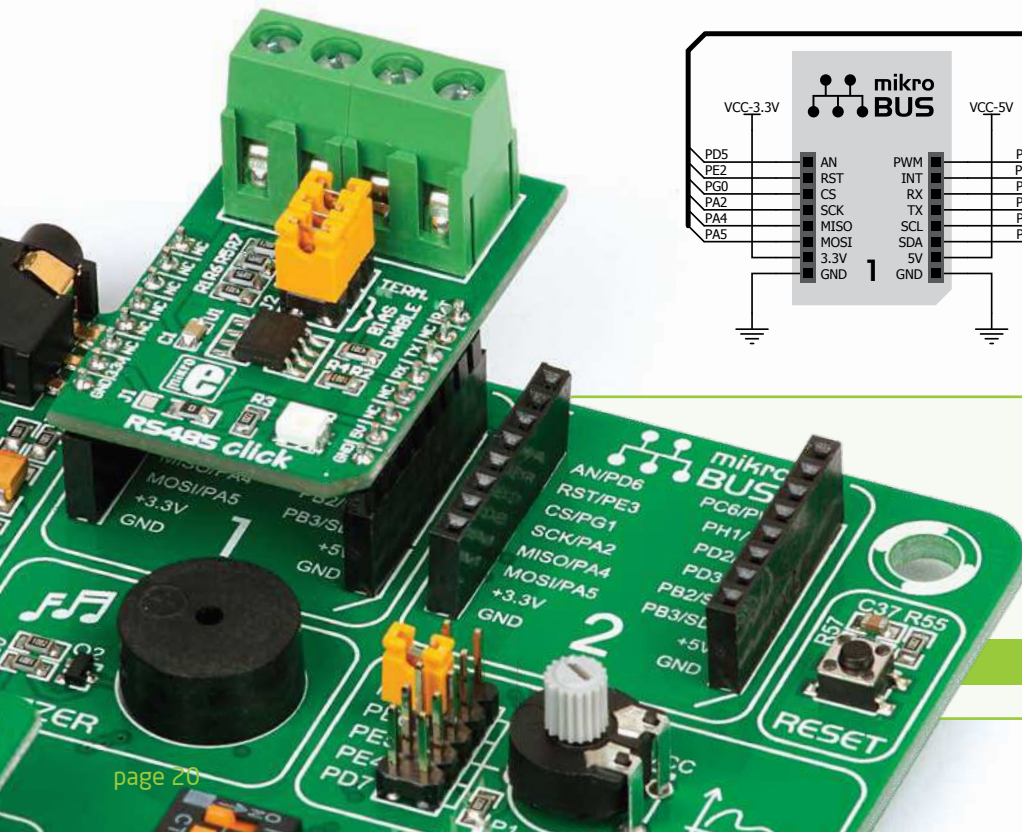
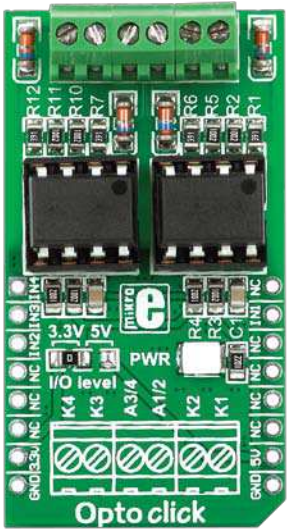


Figure 7-1: mikroBUS™ connection schematic

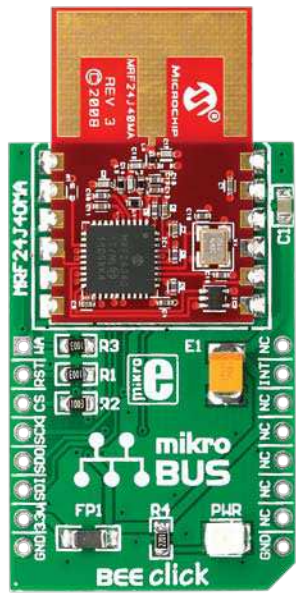
Integrate mikroBUS™ in your design

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>



Opto click™



BEE click™



BlueTooth click™



WiFi PLUS click™



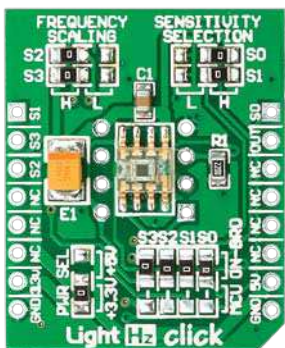
GPS click™

Click Boards™ are plug-n-play!

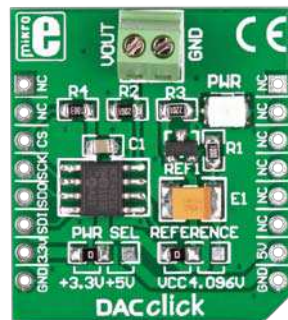
mikroElektronika portfolio of over 200 accessory boards is now enriched by an additional set of mikroBUS™ compatible **Click Boards™**. Almost each month several new Click boards™ are released. It is our intention to provide the community with as much of these boards as possible, so you will be able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

literally zero hardware configuration. Just plug and play. Visit the Click boards™ web page for the complete list of available boards:

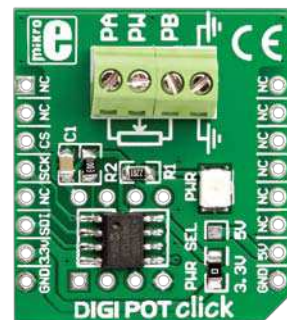
<http://www.mikroe.com/eng/categories/view/102/click-boards/>



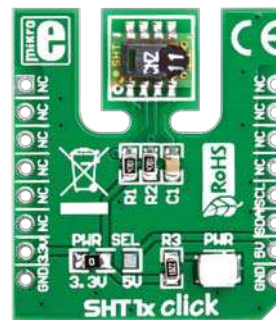
LightHz click™



DAC click™



DIGIPOT click™



SHT1x click™

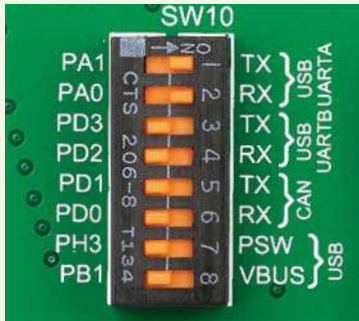


THERMO click™

USB-UART A



Enabling USB-UART A



In order to enable USB-UART A communication you must push **SW10.1** (PA1) and **SW10.2** (PA0) to **ON** position. This connects the **RX** and **TX** lines to **PA0** and **PA1** microcontroller pins.

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.

Modern PC computers, laptops and notebooks are no longer equipped with RS-232 connectors and UART controllers. They are nowadays replaced with USB connectors and USB controllers. Still, certain technology enables UART communication to be done via USB connection. Controllers such as **FT232RL** from FTDI convert UART signals to the appropriate USB standard.

USB-UART A communication is being done through a FT232RL controller, USB connector (**CN7**), and microcontroller UART module. To establish this connection, you must connect **RX** and **TX** lines of the FT232RL to the appropriate pins of the microcontroller. This selection is done using DIP switches **SW10.1** and **SW10.2**.

In order to use USB-UART A module on EasyMx PRO™ v7 for Stellaris®, you must first install FTDI drivers on your computer. Drivers can be found on **Product DVD**:

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

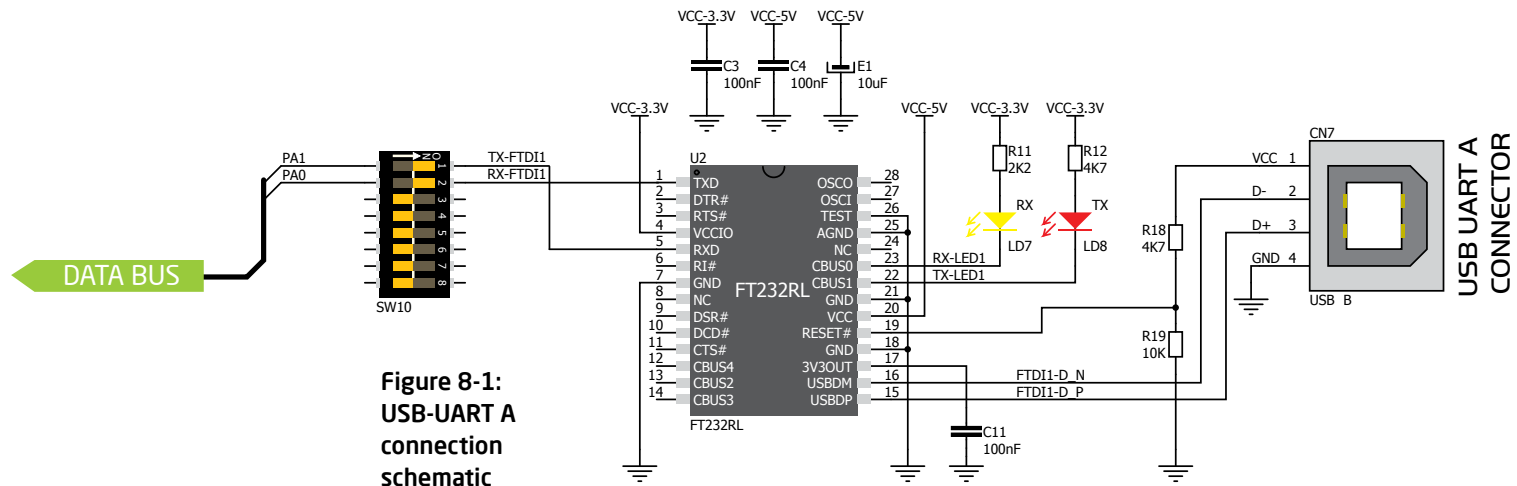
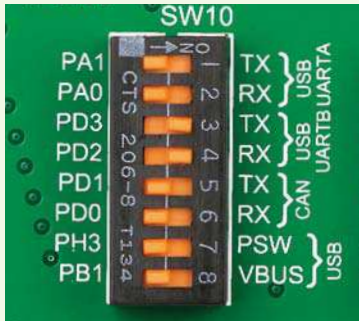


Figure 8-1:
USB-UART A
connection
schematic

USB-UART B



Enabling USB-UART B



In order to enable USB-UART B communication, you must push **SW10.3** (PD3) and **SW10.4** (PD2) to **ON** position. This connects the **RX** and **TX** lines to **PD2** and **PD3** microcontroller pins.

If you need to use more than one USB-UART in your application, you have another **USB-UART B** connector available on the board too. Both available USB-UART modules can operate at the same time, because they are routed to separate microcontroller pins.

USB-UART B communication is being done through a FT232RL controller, USB connector (**CN9**), and microcontroller UART module. To establish this connection, you must connect **RX** and **TX** lines of the FT232RL to the appropriate pins of the microcontroller. This selection is done using DIP switches **SW10.3** and **SW10.4**.

When using either USB-UART A or USB-UART B, make sure to disconnect all devices and additional boards that could interfere with the signals and possibly corrupt the data being sent or received.

In order to use USB-UART B module on EasyMx PRO™ v7 for Stellaris®, you must first install FTDI drivers on your computer. Drivers can be found on **Product DVD**:

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

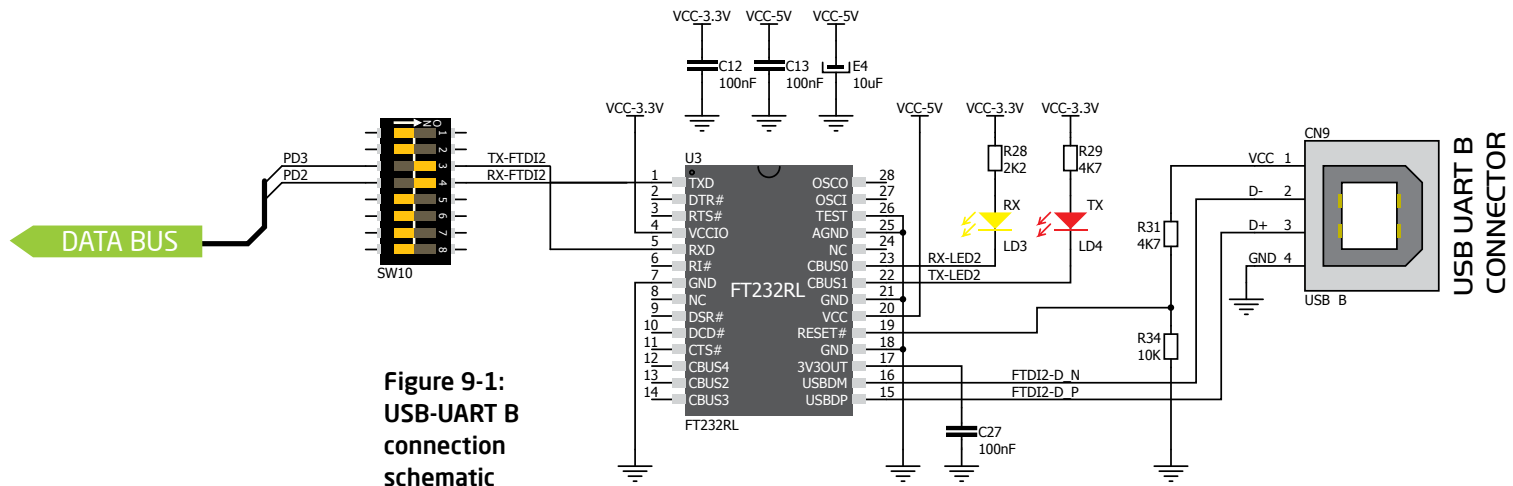


Figure 9-1:
USB-UART B
connection
schematic



USB HOST communication

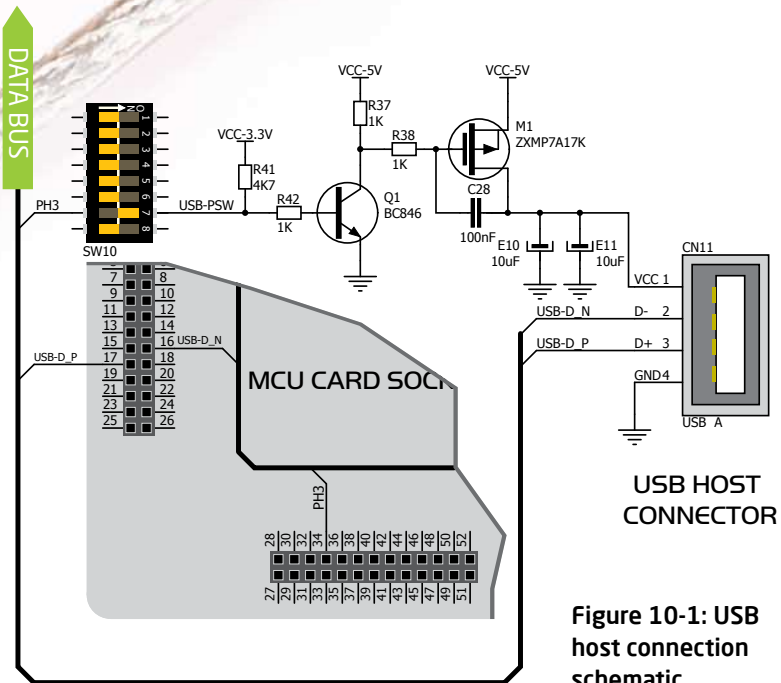


Figure 10-1: USB host connection schematic



Powering USB device

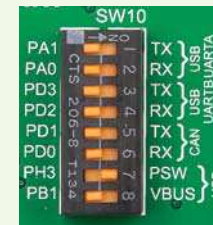


Figure 10-2: Powering USB device through PSW line

You can enable or disable power supply to USB device connected to HOST, through microcontroller **PH3** pin. In order to connect power transistor to microcontroller, you must push **SW10.7** to **ON** position.

USB is the acronym for **Universal Serial Bus**. This is a very popular standard that defines cables, connectors and protocols used for communication and power supply between computers and other devices. EasyMx PRO™ v7 for Stellaris® contains USB HOST connector (**CN11**) for USB Standard Type A plug, which enables microcontrollers that support USB communication to establish a connection with the target device (eg. USB Keyboard, USB Mouse, etc). USB host also provides the necessary 5V power supply to the target. Maximum power which can be drawn depends on the power consumption of the EasyMx PRO™ for Stellaris® board itself. Microcontroller USB data lines are directly connected to MCU card socket pins.

USB device communication

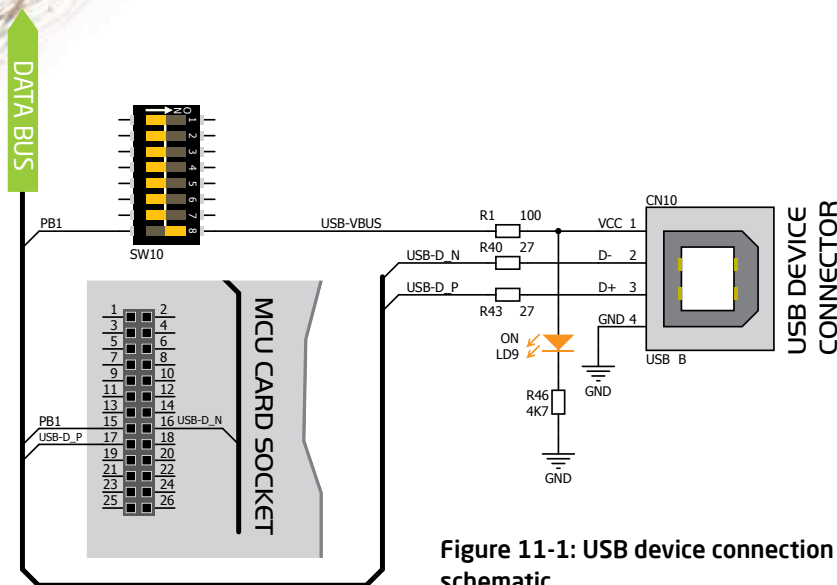


Figure 11-1: USB device connection schematic

EasyMx PRO™ v7 for Stellaris® also contains USB DEVICE connector (**CN10**) which enables microcontrollers that support USB communication to establish a connection with the target host (eg. PC, Laptop, etc). It lets you build a slave USB device (HID, Composite, Generic, etc.). Connector supports USB Standard Type B plug. Detection whether USB DEVICE is connected to HOST can be done through **VBUS** line. This line is traced to microcontroller **PB1** pin. Connection of USB DEVICE **VCC** line and **PB1** pin is established when **SW10.8** DIP switch is in ON position. When connected to HOST, dedicated amber-colored power LED will light up as well. This VCC line cannot be used for powering the board. It's only used for detecting connection.



Detecting connection

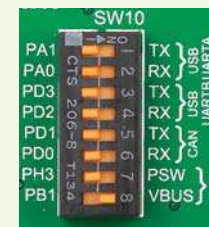


Figure 11-2: enabling USB DEVICE detection via VBUS line

You can detect whether USB device is plugged into the connector using **VBUS** power detection line (**PB1**). Before using this feature, you must connect **PB1** pin to USB connector using **SW10.8** switch.