

# mikroXMEGA™

Manual

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

Development System

 **MikroElektronika**

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

## mikroXMEGA

The mikroXMEGA is a compact development system that enables you to experiment with the ATxmega128A1 microcontroller from Atmel®.

### Key features:

- Bootloader program loaded into the ATxmega128A1 microcontroller;
- USB-UART communication;
- JTAG connector;
- External power supply 3.3V.



Figure 1: mikroXMEGA development system

### How to connect the development system?

The mikroXMEGA development system can be easily connected to external devices via connectors and pads supplied on the board. The USB connector is used to connect the USB-UART module of the development system to a PC. In order to use the USB-UART module, it is necessary to select which UART module of the microcontroller will be used. This is done via jumpers J1-J4. In order to use PC2 (RX) and PC3 (TX) pins, jumpers J1 and J3 should be placed. Likewise, in order to use PD2 (RX) and PD3 (TX) pins, jumpers J2 and J4 should be placed.

The CN4 (PDI) is used for programming/debugging via PDI interface. The CN5 (JTAG) connector is used for programming/debugging via JTAG interface.

The CN1 and CN3 pads enable the microcontroller pins to be easily accessed. The 3.3V power supply voltage is supplied to the pads marked 3.3V (+3.3V) and GND.

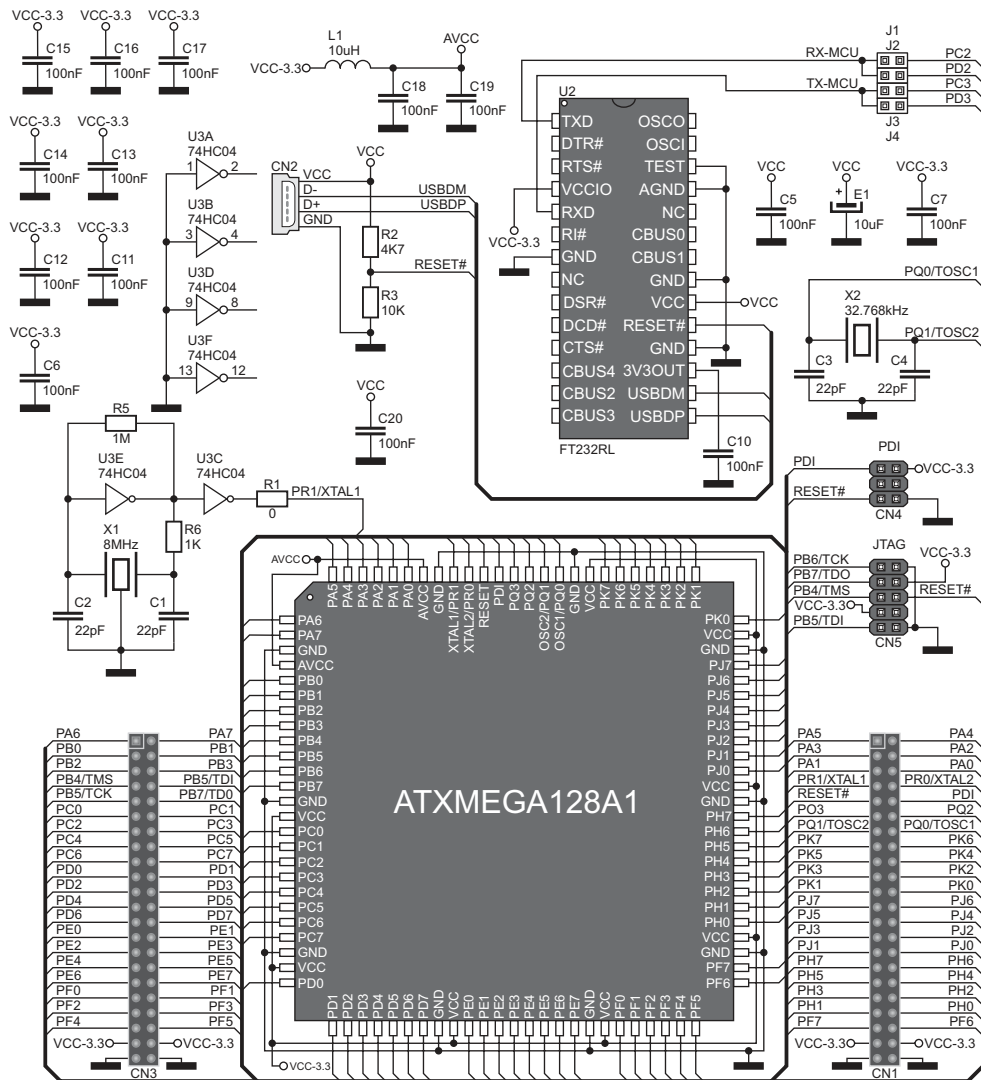


Figure 2: mikroXMEGA development system connection schematic

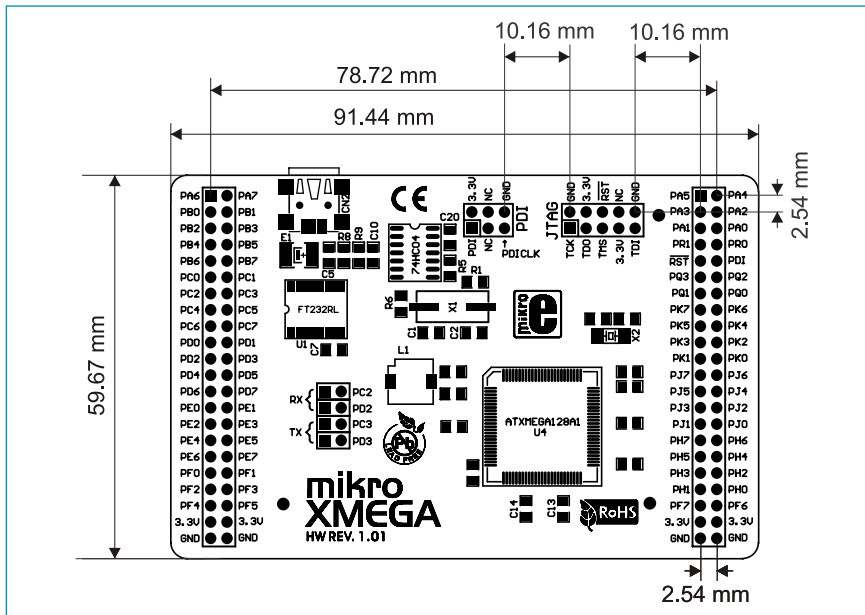


Figure 3: Dimensions of the mikroXMEGA development system

## How to program the microcontroller?

### STEP 1: Connecting the development system to a PC

The ATxmega128A1 microcontroller is programmed via the bootloader program stored in the microcontroller memory. Connect the development system to a PC via the USB connector CN2. Power the board via two pads 3.3V and GND.

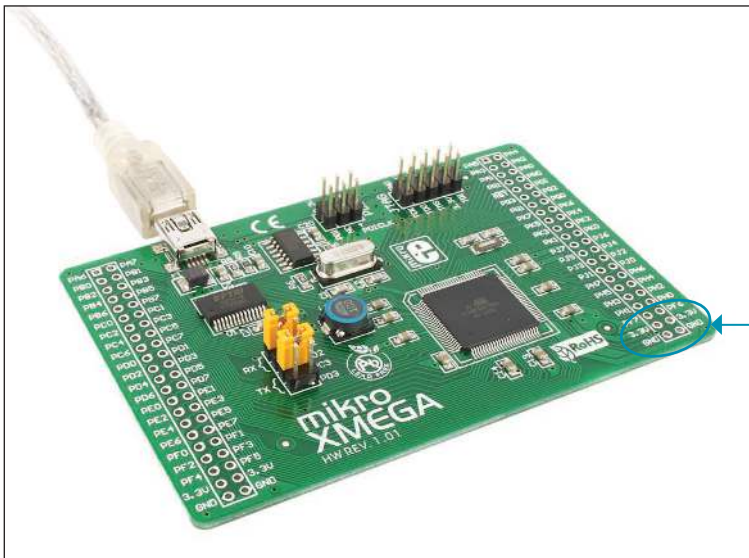


Figure 4: Programming the microcontroller

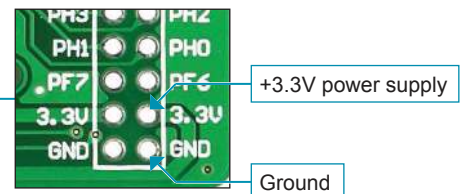



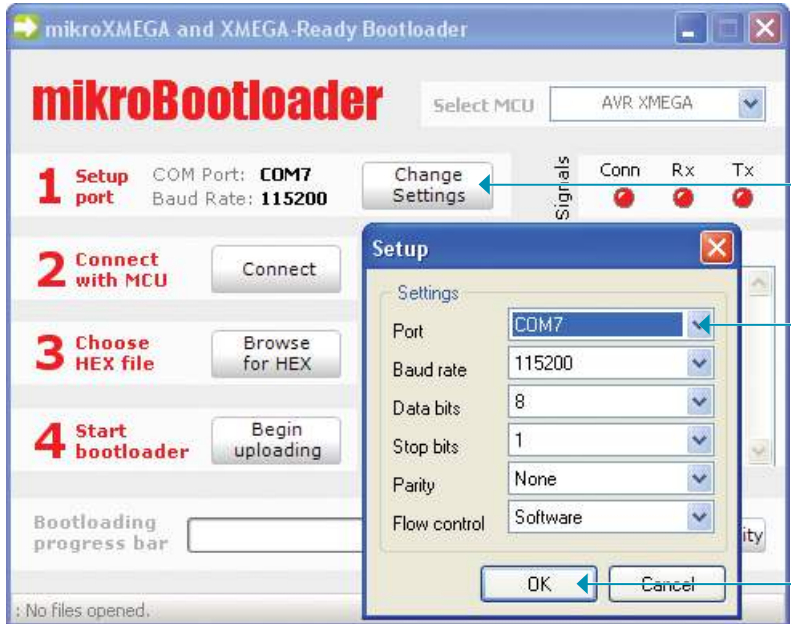
Figure 5: Powering the board

## STEP 2: Starting up the mikroElektronika Bootloader program

Download the mikroElektronika USB HID Bootloader program from Mikroelektronika's website at: [http://www.mikroe.com/eng/downloads/get/1271/mikrobootloader\\_xmega\\_v101.zip](http://www.mikroe.com/eng/downloads/get/1271/mikrobootloader_xmega_v101.zip)

Unzip the file, then double click on the appropriate icon  mikroBootloader  
Bootloader tool for mikroElektr...  
mikroElektronika


## STEP 3: Program settings



Click on the Change Settings button

Select the COM port on the PC that the development system is connected to


Click on the OK button



Select the appropriate COM port from the drop-down list

The Device Manager on your PC contains information about which COM port is used for USB communication with the development system. The COM7 port is used here.

## STEP 4: Connecting



Click on the Connect button within 5s after the development system is connected to PC

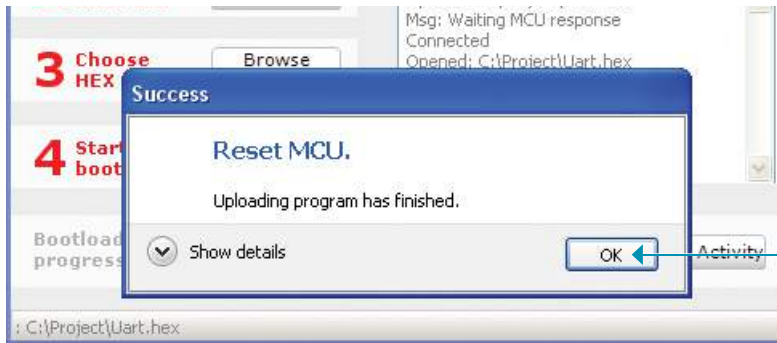
## STEP 5: Browsing for .hex file

The screenshot shows a software interface with a status bar at the top right displaying "Msg: Waiting MCU response Connected". On the left, a red number "3" is followed by the text "Choose HEX file". A button labeled "Browse for HEX" is highlighted with a blue arrow pointing to it from a callout box that says "Click on the Browse for HEX button". Below this, an "Open" file dialog box is shown. The "Look in:" field is set to "Project". The file list contains "DEMO.hex" and "Uart.hex", with "Uart.hex" selected and highlighted in blue. A blue arrow points from a callout box "Choose a .hex file you want to load into the microcontroller" to "Uart.hex". At the bottom of the dialog, the "File name:" field contains "Uart" and the "Files of type:" field is set to "HEX files". The "Open" button is highlighted with a blue arrow from a callout box "Click on the Open button".

## STEP 6: Uploading the .hex file into the microcontroller

The screenshot shows the software interface during the upload process. A red number "4" is followed by the text "Start bootloader". A button labeled "Begin uploading" is highlighted with a blue arrow from a callout box "Click on the Begin uploading button". Below this, another button labeled "Stop uploading" is visible. A "Bootloading progress bar" is shown with a green progress indicator. A blue arrow points from a callout box "Follow the process of uploading in the progress bar" to the progress bar. To the right of the progress bar is a button labeled "Show Activity". At the bottom left, the file path "C:\Project\Uart.hex" is displayed.

## STEP 7: Resetting the microcontroller



After uploading the .hex file into the microcontroller, it is necessary to reset the development system by turning it off for a few seconds and turning it on again. After that, the microcontroller supplied on the development system has been programmed and ready for use.

Click on the OK button



**MikroElektronika**

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

If you want to learn more about our products, please visit our website at [www.mikroe.com](http://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](http://www.mikroe.com/en/support)

If you have any questions, comments or business proposals, do not hesitate to contact us at [office@mikroe.com](mailto:office@mikroe.com)