

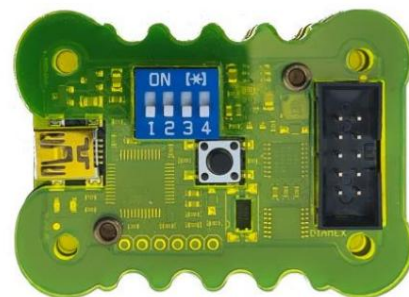
EXA PROG

USB programming adapter for AVR-ISP, AVR-UPDI, STM32, LPC, ESP8266, ESP32

EXA = EXAMIN

EXA-PROG - derived from the Latin "Examin" = to check. EXA-PROG has a built-in function to self-check all signals on the programming lines with the associated Exa-Tool.

EXA-Prog stands for the new generation of ISP programmers, which are not just limited to a special type of microcontroller, but also support several different controller architectures and programming interfaces.



These microcontrollers can be programmed with Exa-Prog:

- ATMELE/Microchip AVR, MegaAVR, TinyAVR controller with ISP interface (MISO, MOSI, SCK, RESET) or alternatively (PDI, PDO, SCK, RESET).
- ATMELE/Microchip TinyAVR, AVR DA controller with UPDI interface. With high-voltage function for controllers with deactivated UPDI pin.
- STM32 32-bit ARM Cortex-M controllers that have an internal UART bootloader.
- NXP LPC 32-bit ARM Cortex-M controllers that have an internal UART bootloader.
- ESP8266, ESP32, ESP32-S2 controllers and modules via internal UART bootloader.

Capabilities:

- Function selection via DIP switch.
- Switchable signal level.
- Integrated high-voltage generator for UPDI programming.
- Automatic bit rate adjustment in AVR ISP mode
- Clock generator for AVR controllers with fused oscillator.
- Button for resetting the connected microcontroller.
- Standard 10-pin ISP connector. Optional accessories: 10-pin to 6-pin adapter, ESP01 adapter.
- Mini USB connector for power supply and connection to PC.
- Firmware updateable via USB.
- Windows PC tool to test the signal levels on the programming port.

Technical specifications:

- Current consumption without connected microcontroller: approx. 30mA
- Signal level at the programming port: 5V (USB voltage) or 3.3V
- Power supply for external circuit: Max. 300mA (3.3V), Max. 500mA (5V)
- UPDI high voltage: about 12.3V
- Case size approx. 55x40x10mm, weight approx. 25g

Supported software:

- AVR/Microchip Studio from v7.x (only AVR-ISP: STK500).
- AVRDUDE from v6.3 (AVR-ISP: STK500, AVR-UPDI: jtag2updi).
- Development environments based on AVRDUDE as a programming interface, eg Arduino, Bascom.
- ESPTOOL (esptool.py, esptool.exe) and all programs that use ESPTOOL as a programming interface.

Note: In the current firmware version, EXA-PROG cannot be used under ATMELE/MICROCHIP-Studio for programming AVR controllers with UPDI interface. However, you can use the studio as an editor and compiler, the generated HEX files must then be transferred externally to the microcontroller with AVRDUDE. You can find out how this works in a link at the end of these instructions.

EXA PROG

USB programming adapter for AVR-ISP, AVR-UPDI, STM32, LPC, ESP8266, ESP32

INSTALLATION

Connect EXA-PROG to any USB 2.x or USB 3.x port of your PC or notebook using a mini USB cable.

Windows 7 and 8.x

EXA-PROG is addressed via a virtual COM port. The driver file must be installed under Windows 7 and 8.x. In the Windows device manager, click on the corresponding entry and in the "Driver" tab click on "Update driver". Then select the file **prog-isp.inf** from the EXA-PROG program package.

Windows 8.x

If you are still using Windows 8.x, you may need to turn off driver signature enforcement. You can easily find out how this works with a Google search. If possible, consider upgrading to Windows 10.

windows 10

Windows 10 does not require driver installation. Here, each component registers as a "Serial USB device (COMx)".

The best way to test which COM port is valid for the component you are using is to start the EXA-PROG tool and click on the list of COM ports before and after you have plugged in the component. The COM port that has been added belongs to your EXA-PROG.

EXA-PROG does not have its own program for programming microcontrollers. The firmware of the EXA-PROG was written to be compatible with many existing programming tools from manufacturers or with freely available tools.

When delivered, EXA-PROG is set to the AVR-ISP mode with 3.3V voltage (all DIP switches to OFF).

In AVR-ISP mode EXA-PROG is compatible with **STK500** and works with AVR/ATMEL/MICROCHIP-Studio as well as AVRDUDE (Arduino, Bascom).

ÿ Note: Under Windows 10 please only use ATMEL/MICROCHIP-Studio from 7.x, older versions do not work.

In UPDI mode, EXA-PROG emulates a JTAG2UPDI programmer. This can currently only be used with **AVRDUDE** (see link at the end of this guide).

ESP mode is compatible with **ESPTOOL** (esptool.py, esptool.exe).

The programming modes STM32 and NXP/LPC are largely intended for the use of your own programming tools.

Please always ensure that the DIP switches are set to match the microcontroller to be programmed. You should also pay attention to the programming voltage, AVR controllers can be operated with 3.3V and 5V, almost all 32-bit microcontrollers can ONLY be operated with 3.3V.

Do not toggle the DIP switches during programming. Switching always leads to a reset of the programmer, this is indicated by a short flickering of the red LED. A constant glow of the green LED indicates that the programmer is ready for operation.

Do not press the RESET button during a programming process. The programming is aborted and the data in the flash memory of the microcontroller may be invalid.

EXA PROG

USB programming adapter for AVR-ISP, AVR-UPDI, STM32, LPC, ESP8266, ESP32

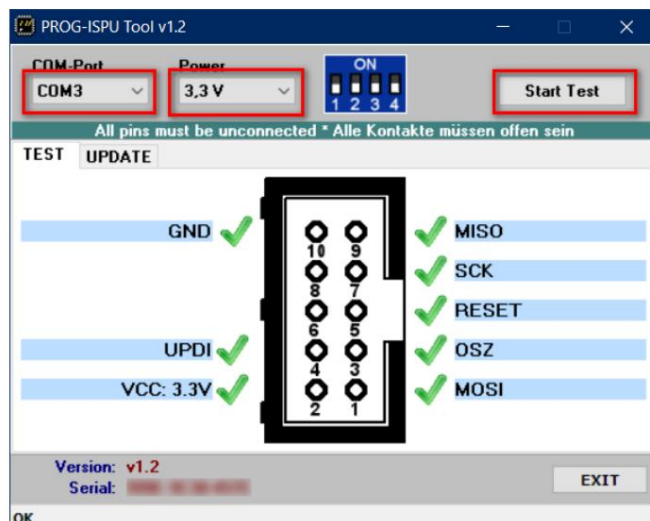
TEST FUNCTION

If you have problems programming a connected microcontroller, you can use the following procedure to check whether the problem is with the programming adapter. EXA-PROG has an integrated test hardware that can be used to check the signal levels at the programming interface.

To do this, please start the EXA-PROG tool under Windows (file name: Exa_Tool_xxx.exe; xxx = version) from the EXA PROG program package.

1. Select the "TEST" tab.
2. Set the EXA-PROG COM port. If you do not know exactly which COM port your EXA-PROG occupies, you can easily determine this by comparing the COM port list before and after connecting the programmer to the PC.
3. Select voltage (Power).
4. DIP switches on the programmer according to the pattern shown set.
5. Remove all leads from the programming port of the EXA PROG.
6. Press the "Start Test" button. All ports are now controlled one after the other and the signal levels are measured.

⚠ Please leave the programming connection on the programmer open for the test. There must be no connections to a microcontroller or an external circuit.



If necessary, you can also update the firmware of your EXA-PROG via the "UPDATE" tab is.

POSITIVE:

If all signal levels are OK, the programming problem is probably due to the connection to the microcontroller or incorrectly set programming software.

NEGATIVE:

If the test fails, first check whether the voltage is set correctly and whether the DIP switches on the EXA-PROG match the switch position displayed. Are there really no external components connected to the programming connector? Also check the USB voltage with a multimeter by setting DIP switch 1 to ON and measuring the voltage between PIN2 (plus) and PIN10 (minus) on the programming connection. This must be between 4.5V and 5.2V.

If, despite all tests and further checks, it turns out that your EXA-PROG is defective, please contact the dealer from whom you bought the programmer. Please always give the exact error description, possibly also with a screenshot of the EXA-PROG tool.

PREVENTIVE:

Unfortunately, it happens again and again that the programming adapter is destroyed during experiments with the programming of microcontrollers. Incidentally, this also happens to the absolute experts, which is why most developers have a second programmer in reserve so that the current project is not slowed down. In order to prevent a defect in the programmer, you should note the following:

- Always connect the GND line (PIN10 of the programming connection) to the circuit in which your microcontroller to be programmed is located first. So there is a potential equalization between the programmer and the circuit.
- Do not connect any output signals on the programming connection (e.g. PIN1, PIN3, PIN5, PIN7) directly to GND or to the power supply. The output drivers can be destroyed.
- Make sure that the pins on the programming connection do not have a higher voltage than the set one connected to DIP switch 1.

EXA PROG

USB programming adapter for AVR-ISP, AVR-UPDI, STM32, LPC, ESP8266, ESP32

LEDS

Green LED	Lights up constantly when the set operating mode is ready. Goes out briefly when the RESET button is pressed.
LED red	Flickers during data transfers via RXD/TXD and during programming. Flickers briefly when the programming mode is changed using the DIP switch.

RESET BUTTON

The button on the EXA-PROG triggers a manual reset of the connected microcontroller. This is often required if a start process (boot) of a freshly programmed controller is to be tested several times. The function of the button depends on the selected programming mode.

AVR ISP, AVR UPDI:

PIN5 (RESET) is set to LOW level for as long as the button is pressed.

The green LED goes out as long as the button is pressed.

STM32:

A short button press (< 1 second) triggers a RESET of the connected controller. After releasing the button, a 50ms reset pulse is generated at PIN5 (RESET). PIN3 (BOOT) stays LOW.

A long button press (>1 second) puts the connected microcontroller into boot mode. PIN3 (BOOT) is set to HIGH level, a 50ms long reset pulse is generated and then PIN3 (BOOT) goes back to LOW.

The green LED flickers briefly as a sign that the reset has been triggered.

NXP/LPC:

A short button press (< 1 second) triggers a RESET of the connected controller. After releasing the button, a 50ms long LOW pulse is generated at PIN5 (RESET). PIN3 (BOOT) remains HIGH.

A long button press (>1 second) puts the connected microcontroller into boot mode. PIN3 (BOOT) is set to LOW level, a 50ms long reset pulse is generated and then PIN3 (BOOT) goes HIGH again.

The green LED flickers briefly as a sign that the reset has been triggered.

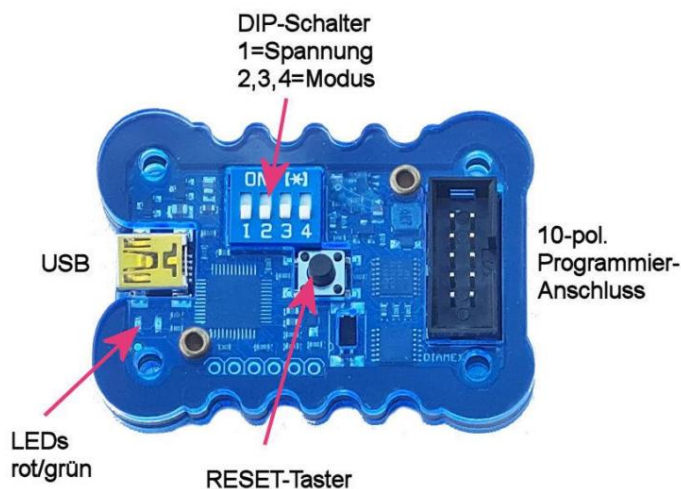
ESP:

A short button press (< 1 second) triggers a RESET of the connected controller. After releasing the button, a 50ms long LOW pulse is generated at PIN5 (RESET). PIN3 (GPIO0) remains HIGH.

A long button press (>1 second) puts the connected microcontroller into boot mode. PIN3 (GPIO0) is set to LOW level, a 50ms long reset pulse is generated and then PIN3 (GPIO0) goes HIGH again.

The green LED flickers briefly as a sign that the reset has been triggered.

⚠ Never press the RESET button during a programming process. The programming is aborted and the data in the flash memory of the microcontroller may be invalid.

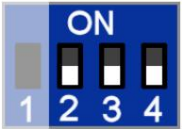
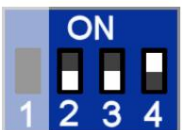
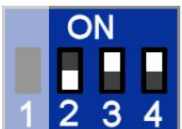
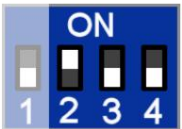
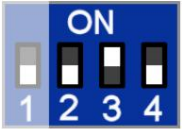


EXA PROG

USB programming adapter for AVR-ISP, AVR-UPDI, STM32, LPC, ESP8266, ESP32

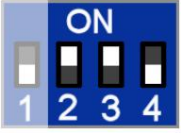
DIP SWITCH

SWITCH 2, 3, 4: Operating mode

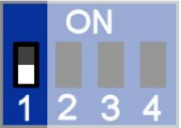
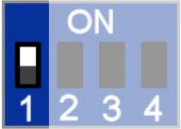
AVR ISP	
	<p>2=OFF, 3=OFF, 4=OFF</p> <p>Programmer for Atmel AVR controllers with ISP interface</p> <p>These controllers are programmed via the ISP interface, which on most AVR controllers is equivalent to the pins for the SPI bus (note! There are some controllers that have separate PDI/PDO pins). In addition to SCK, MISO, MOSI, the reset line is also required. EXA-PROG emulates a STK500 programmer and is therefore compatible with ATMEL/MICROCHIP-Studio and AVRDUDE. The programming voltage can be set to 3.3V or 5V depending on the connected AVR controller.</p>
AVR UPDI	
	<p>2=OFF, 3=OFF, 4=ON</p> <p>Programmer for Atmel AVR controllers with UPDI interface</p> <p>These controllers are programmed via the UPDI interface. EXA-PROG emulates a jtag2updi programmer and is currently only compatible with AVRDUDE. The programming voltage can be set to 3.3V or 5V depending on the connected AVR controller.</p>
	<p>2=OFF, 3=ON, 4=ON</p> <p>Programmer for Atmel AVR controllers with UPDI interface (HV mode)</p> <p>Mode for reactivating the UPDI interface if the function of the UPDI pin in the fuses has been changed on ATtiny controllers. Please do not use with controllers with a dedicated UPDI pin (e.g. AVR32/64/128DA).</p>
STM32	
	<p>2=ON, 3=OFF, 4=OFF</p> <p>Programmer for STM32 Cortex-M controller with UART bootloader</p> <p>The STM32 controller is programmed via the integrated bootloader through one of the serial interfaces (see data sheet of the controller to be programmed). connection the programmer finds the controller via the RX and TX connections. For the automatic activation of the integrated bootloader of the controller, the connections BOOT0 and RESET required, so no jumpers or switches are required on the circuit.</p> <p>Please set DIP1 = OFF, programming voltage to 3.3V!</p>
NXP/LPC	
	<p>2=OFF, 3=ON, 4=OFF</p> <p>Programmer for NXP/LPC Cortex-M controllers with UART bootloader</p> <p>The LPC controller is programmed via the integrated bootloader through the serial interface UART0. The programmer is connected to the controller via the RX and TX connections. Connections P2[10] and RESET are also required to automatically activate the integrated bootloader of the controller, which means that no jumpers or switches are required on the circuit. EXA-PROG in conjunction with the "Flash-Magic" tool automatically switches the controller to bootloader mode, programs the flash memory and then starts the user program that has just been loaded.</p> <p>Please set DIP1 = OFF, programming voltage to 3.3V!</p>

EXA PROG

USB programming adapter for AVR-ISP, AVR-UPDI, STM32, LPC, ESP8266, ESP32

ESP	
	<p>2=ON, 3=ON, 4=OFF</p> <p>Programmer for ESP8266 and ESP32 with ESPTOOL</p> <p>The ESP8266 and ESP32 are programmed via the integrated bootloader via the TXD0 and RXD0 lines. The lines GPIO0 and RESET (EN) are also required for the automatic activation of the bootloader.</p> <p>Please set DIP1 = OFF, programming voltage to 3.3V!</p>

SWITCH 1: Voltage level

3.3V	
	<p>1 = OFF</p> <p>A voltage of 3.3V is present at PIN 2 of the 10-pin programming connection. All signal levels on the programming lines have a maximum voltage of 3.3V.</p>
5V	
	<p>1 = ON</p> <p>A voltage of approx. 5V is present at PIN 2 of the 10-pin programming connection. All signal levels on the programming lines have a maximum voltage of approx. 5V.</p> <p>Please note that the exact voltage value can vary and depends on the voltage at the USB port. Depending on the load, this can fluctuate between 4.5V and 5.2V. However, this is usually irrelevant when programming microcontrollers.</p>

Hints:

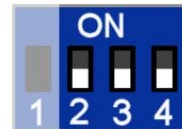
- The maximum load on the programming connector is 500mA at 5 volts and 300mA at 3.3 volts. If a larger current is required, please supply the connected circuit with external voltage and do not wire PIN2 on the programming connection.
 - The external circuit must always be operated with the same voltage as the voltage used on the programmer is set with DIP switch 1.
 - GND must always be connected to the circuit in which the controller to be programmed is located.
- ÿ Please note that almost all 32-bit microcontrollers may only be operated with 3.3V and that DIP switch 1 must be set to OFF. The controllers can be destroyed if they are operated with 5V.

EXA PROG

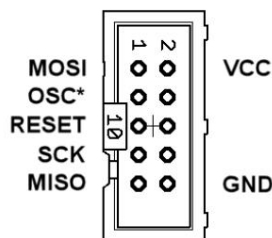
USB programming adapter for AVR-ISP, AVR-UPDI, STM32, LPC, ESP8266, ESP32

Connection of an AVR controller with ISP interface

• **DIP SWITCHES: 2 = OFF, 3 = OFF, 4 = OFF**



Connect the wires from the programming port on the EXA PROG directly with the pins on the controller:



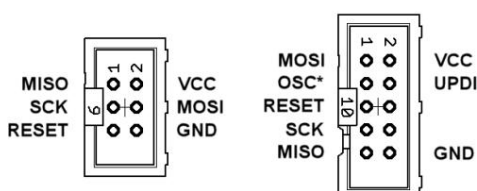
10 pin Pin header	AVR controller
PIN1 (MOSI)	MOSI or PDI
PIN5 (RESET)	RESET
PIN7 (SCK)	SCK
PIN9 (MISO)	MISO or PDO
Options: PIN3 (OSC)	XTAL1 (XTALIN)
PIN10 (GND)	GND
Options: PIN2 (VCC)	VCC

AVR controller pin numbers vary depending on the type and case shape. Please refer to the data sheet to determine the pin numbers that match your controller.

DIP switch 1: OFF = 3.3V, ON = 5V. Be sure to set the same voltage here that your circuit is operated with. If the microcontroller is programmed in an external socket, both voltage values can be set.

NOTES, TIPS:

- Some AVR controllers (eg AT90CAN32/64/128, ATmega64/128) have special programming pins PDI, PDO. You're welcome use these lines instead of MISO/MOSI (see data sheet).
- EXA-PROG supports adaptive SPI bit rate adjustment. If the SPI bit rate is set too high, the connected AVR controller is usually not recognized. The rule states that the SPI bit rate must not exceed a quarter of the controller's clock rate. If no connection can be established with the set bit rate, EXA-PROG automatically switches to lower bit rates until the controller responds.
- If the connected AVR controller still does not want to respond, please check the line connections first. MISO to MISO and MOSI to MOSI (this is often swapped), SCK to SCK and RESET to RESET. Are there no other components connected to the programming lines? Any capacitor present on the RESET pin must be removed. Does the connected microcontroller have a power supply? The AVR controller may be missing the system clock because the fuses are set to an external clock. In this case, a crystal can be connected to the XTAL pins of the controller for clock generation, or the clock signal can be routed from PIN3 (OSC) of the 10-pin header to XTALIN or XTAL1 (see controller data sheet). The default clock frequency on PIN3 of the ISP connector is approximately 1.3MHz.
- EXA-PROG does not support ATXmega controllers with PDI interface and ATTiny controllers with TPI interface.
- Only connect PIN2 to the circuit to be programmed if it does not have its own power supply and the expected power consumption does not exceed the maximum limit of the EXA-PROG.
- For boards with a 6-pin ISP connector, please use the optionally available adapter or connect your circuit to the 10-pin ISP connector with suitable jumper cables.



Adapter 10-pin to 6-pin ISP

(Optional accessory, not included with the EXA PROG included)

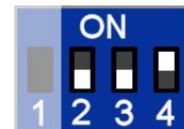


EXA PROG

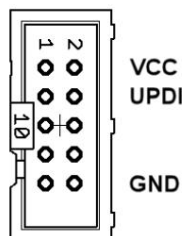
USB programming adapter for AVR-ISP, AVR-UPDI, STM32, LPC, ESP8266, ESP32

Connection of an AVR controller with UPDI interface

• **DIP SWITCHES: 2 = OFF, 3 = OFF, 4 = ON**



Connect the wires from the programming port on the EXA PROG directly with the pins on the controller:



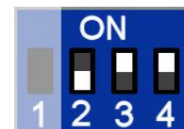
10 pin Pin header	AVR controller
PIN4 (UPDI)	UPDI (RESET, PORT)
PIN10 (GND)	GND
Options: PIN2 (VCC)	VCC

AVR controller pin numbers vary depending on the type and case shape. Please refer to the data sheet to determine the pin numbers that match your controller.

DIP switch 1: OFF = 3.3V, ON = 5V. Be sure to set the same voltage here that your circuit is operated with. If the microcontroller is programmed in an external socket, both voltage values can be set.

UPDI HV MODE

• **DIP SWITCHES: 2 = OFF, 3 = ON, 4 = ON**



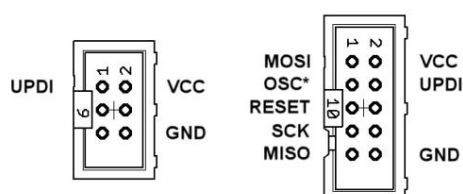
For ATtiny controllers that do not have a separate UPDI pin (e.g. Tiny412, Tiny1614, Tiny3216), the pin can be switched to an alternative function via the fuses. After that, programming in the standard UPDI mode is no longer possible. In this case, select the UPDI high-voltage mode, in which the UPDI connection of the controller is reactivated via a short 12 volt pulse for programming.

If the controller is to be programmed in a circuit, make sure that there are no components on the UPDI pin.

• **Please do not use this mode for controllers with a dedicated UPDI pin (e.g. AVR32/64/128DA).**

NOTES, TIPS:

- Only connect PIN2 to the circuit to be programmed if it does not have its own power supply and the expected power consumption does not exceed the maximum limit of the EXA-PROG.
- The software interface of the EXA-PROG in UPDI mode emulates a jtag2updi interface. This is not supported by AT MEL/ MICROCHIP Studio. Instead, use the AVRDUDE tool for programming. This is used in various development environments such as Arduino or Bascom. You can find a lot of information about this on the Internet and via the link at the end of these instructions.
- For boards with a 6-pin UPDI connector, please use the optionally available adapter or connect your circuit to the 10-pin ISP connector with suitable jumper cables.



Adapter 10-pin to 6-pin UPDI

(Optional accessory, not included with the EXA PROG included)

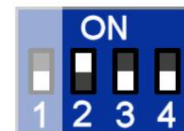


EXA PROG

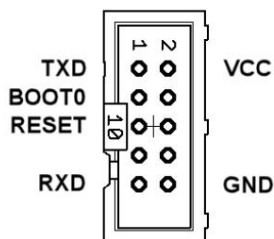
USB programming adapter for AVR-ISP, AVR-UPDI, STM32, LPC, ESP8266, ESP32

Connection of an STM32 controller

• DIP SWITCHES: 2 = ON, 3 = OFF, 4 = ON



Connect the wires from the programming port on the EXA PROG directly with the pins on the controller:



10 pin Pin header	STM32 controller	Control
PIN1 (TXD)	USART1-RX (PA10)	
PIN3 (BOOT)	BOOT0	DTR
PIN5 (RESET)	RESET	!RTS
PIN9 (RXD)	USART1-TX (PA9)	
PIN10 (GND)	GND	
Options: PIN2 (VCC)	VCC	

The pin numbers of the STM32 vary depending on the type and package shape. Please refer to the STM32 datasheet to determine the pin numbers that match your controller.

DIP switch 1: Must be set to OFF = 3.3V.

NOTES, TIPS:

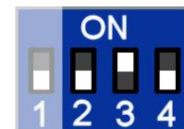
- So that the bootloader can be started, it must be ensured that BOOT1 (PB2) is at ground potential during the RESET (no longer available with newer controller types). If the pin is not otherwise used, you can connect it directly to GND. Otherwise, please connect it to GND via a 10 kOhm resistor.
- To program the STM32, it must be supplied with 3.3 volts if the power consumption is not higher than 300mA, the controller including the circuit can also be supplied from the EXA-PROG.
- No quartz needs to be connected to the STM32 for programming, the bootloader of the STM32 uses its Internal RC oscillator (HSI clock) to generate the system clock.
- Unfortunately, the BOOT0 and RESET lines cannot be controlled with the “STM Flash Loader” programming tool. Do not connect the BOOT0 and RESET lines. In this case you have to take care of the activation of the STM32 bootloader yourself (BOOT0 on high, then RESET briefly on low. This is also possible via the reset button on the EXA-PROG.
- If you are developing your own programming tool, the BOOT and RESET lines can be controlled via the DTR (BOOT) and RTS (!RESET).
- STM8 controllers are not supported by this programmer.

EXA PROG

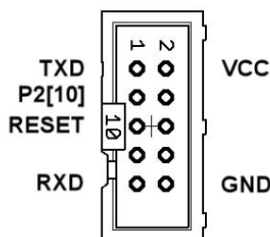
USB programming adapter for AVR-ISP, AVR-UPDI, STM32, LPC, ESP8266, ESP32

Connection of an NXP/LPC controller

• DIP SWITCHES: 2 = OFF, 3 = ON, 4 = ON



Connect the wires from the programming port on the EXA PROG directly with the pins on the controller:



10 pin Pin header	NXP controller	Control
PIN1 (TXD)	UART0-RXD P0[2]	
PIN3 (BOOT)	P2[10]	!RTS
PIN5 (RESET)	RESET	!DTR
PIN9 (RXD)	UART0-TXD P0[3]	
PIN10 (GND)	GND	
Options: PIN2 (VCC)	VCC	

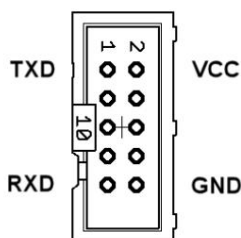
The NXP controller pin numbers vary depending on the package shape. Please refer to the data sheet to determine the pin numbers that match your controller.

DIP switch 1: Must be set to OFF = 3.3V.

NOTES, TIPS:

- To program the NXP controller, it must be supplied with 3.3 volts.
- Programming does not require a crystal to be connected to the NXP controller, the controller's bootloader uses the internal RC oscillator to generate the system clock.
- Controlling the BOOT and RESET lines works fine with the "Flash Magic" tool.
- If you develop your own programming tool, the BOOT and RESET lines can be controlled via the DTR (!BOOT) and RTS (!RESET) signals. This can also be done manually using the reset button on the EXA-PROG.
- 8-bit controllers (LPC9xx) and NXP controllers without an ARM Cortex core are not supported by this programmer supports.

EXA-PROG as USB/serial converter



EXA-PROG can be used as USB/serial converter in STM32, NXP/LPC and ESP32 modes.

The maximum transfer rate is 1 MBit.

The following transmission parameters are supported: 7 + 8 bits, parity OFF, EVEN, ODD, 1 or 2 stop bits.

TXD (PIN1) is connected to the RX pin and RXD (PIN9) to the TX pin of the microcontroller.

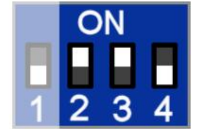
DIP switch 1: OFF = 3.3V, ON = 5V. Be sure to set the same voltage here that your circuit is operated with.

EXA PROG

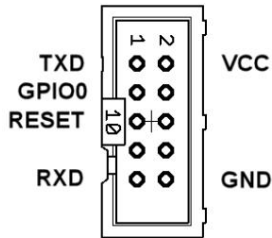
USB programming adapter for AVR-ISP, AVR-UPDI, STM32, LPC, ESP8266, ESP32

Connection of an ESP controller

• **DIP SWITCHES: 2 = OFF, 3 = OFF, 4 = ON**



Connect the lines of the programming port on the EXA PROG directly to the pins on the controller or on an ESP8266 or ESP32 module:



10 pin Pin header	ESP8266	ESP32
PIN1 (TXD)	RXD0	RXD0
PIN3 (GPIO0)	CH_PD (GPIO0)	GPIO0
PIN5 (RESET)	RST	EN
PIN9 (RXD)	TXD0	TXD0
PIN10 (GND)	GND	GND
Options: PIN2 (VCC)	VCC	VCC

DIP switch 1: Must be set to OFF = 3.3V.

NOTES, TIPS:

- The RST pin of the ESP8266 or EN pin of the ESP32 should be connected to +3.3V with a 10 kOhm resistor. A capacitor to ground (as recommended in many circuits) must not be on this pin, since the RESET is controlled by the EXA-PROG.
- To program the ESP8266 or ESP32, it must be supplied with 3.3 volts. If the current consumption is not higher than 300mA, the controller including the circuit can also be supplied from the EXA-PROG.
- When using ESPTOOL, the control lines GPIO0 and RESET are activated automatically. A manual activation of the bootloader is therefore not necessary, but can also be triggered via the reset button on the EXA-PROG if this does not work via the software.

10-pin adapter to ESP

Socket strip for ESP-01 modules, pin headers for connecting ESP8266 or ESP32 controllers via jumper cables.

(Optional accessory, not included in the EXA-PROG scope of delivery)



EXA PROG

USB programming adapter for AVR-ISP, AVR-UPDI, STM32, LPC, ESP8266, ESP32

HINTS

© Erwin Reuss; Folker rod. Use and disclosure of this information, even in part, only with the permission of the copyright holder. All brand names, trademarks and registered trademarks are the property of their rightful owners and are used here for description only.

NOTICE OF LIABILITY

The manufacturer assumes no liability for damage that may result from the use of the EXA-PROG.

LEFT

LED Genial Online Shop

<http://www.led-genial.de>

Diamex online shop

<http://www.diamex.de>

AVR/ATMEL/MICROCHIP Studio Download

http://www.mikrocontroller.net/articles/Atmel_Studio

UPDI programming with AVRDUDE

<https://github.com/EITangas/itag2updi#using-with-avrdude>

ESPTOOL for ESP8266 and ESP32

<https://github.com/themadinventor/esptool>

distribution



DIAMEX production and trade GmbH

Innovation Park Wuhlheide
Köpenicker Strasse 325, house 41
12555 Berlin

Phone: 030-65762631

Email: info@diamex.de Homepage:

<http://www.diamex.de>

manufacturing



www.tremex.de

Köpenicker Str. 325 12555 Berlin
Tel. 030-65762631

Hersteller: Tremex GmbH
DIAMEX × OBD-DIAG × TREMEX
WEE-Reg.Nr. DE 51673403