

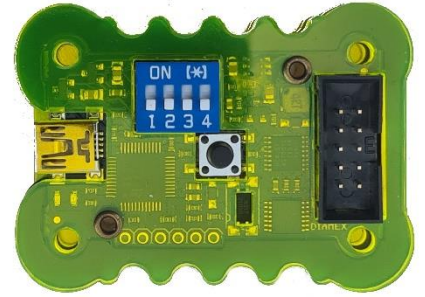
EXA-PROG

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

EXA = EXAMINE

EXA-PROG – derived from the originally Latin term “to examine”. 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 that are not just limited to a certain microcontroller type but support several different controller architectures and programming interfaces.



These Microcontrollers Can Be Programmed with EXA-PROG:

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

Functions:

- Select functions via the DIP switch.
- Signal level switchable.
- Integrated high-voltage generator for UPDI programming.
- Automatic bit rate adjustment in the AVR ISP mode
- Clock generator for AVR controller with wrong fused oscillator.
- Button for reset of the connected microcontroller.
- 10 pin standard ISP port. Optional equipment: 10-pin to 6-pin adapter, ESP01 adapter.
- Mini USB port for power supply and connection to PC.
- Firmware updateable via USB.
- Windows PC tool for testing the signal levels at the programming port.

Technical Data:

- Current consumption without connected microcontroller: approx. 30mA
- Signal level at programming connector: 5V (USB level) or 3.3V
- Power supply for external circuit: max. 300mA (3.3V), max. 500mA (5V)
- UPDI high voltage: approx. 12.3V
- Case size approx. 55 x 40 x 10mm, weight approx. 25g

Supported Software:

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

Notice: In the current firmware version EXA-PROG cannot be used for programming AVR controllers with a UPDI interface with ATMEL/MICROCHIP Studio. However, you can use the Studio as an editor and compiler. The HEX files that are created need to be transferred externally via AVRDUDE to the microcontroller. You can find out how this works in a link at the end of this guide.

EXA-PROG

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

INSTALLATION

Connect the EXA-PROG via a mini USB cable with any USB-2.x or USB 3.x port of your PC or laptop.

Windows 7 and 8.x

EXA-PROG uses a virtual COM port. With Windows 7 and 8.x the driver file needs to be installed first. Click on the respective entry in the device manager of Windows, in the "Driver" tab click on "update driver". Then select the file **prog-isp.inf** of the EXA-PROG program package.

Windows 8.x

If you are still using Windows 8.x you may have to disable "driver signature enforcement". You can easily find out how this works with a Google search. If possible, you should consider going for Windows 10 or 11.

Windows 10/11

With Windows 10 or 11 there is no driver installation necessary. Each component registers itself as a "serial USB device (COMx)".

The best way to test which COM port is valid for your used component is by starting the EXA-PROG tool and clicking on the list of COM ports before and after you plugged in the component. The COM port that was added is your EXA-PROG.

EXA-PROG doesn't come with its own program for programming microcontrollers. EXA-PROG's firmware is compatible to many existing programming tools of the manufacturers or to freely available tools.

In its delivery condition EXA-PROG is set to the mode AVR-ISP with 3.3V (all DIP switches to OFF).

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

⇒ **Notice: With Windows 10 or 11 please only use ATMEL/MICROCHIP Studio 7.x or higher. Older versions do not work.**

In the UPDI mode EXA-PROG emulates a **JTAG2UPDI** programmer. At the moment this can only be used with **AVRDUDE** (see link at the end of this manual).

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

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

Please always pay attention that the DIP switches are set fitting to the microcontroller that is to be programmed. Please be sure to pay attention of 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 flip the DIP switches throughout the programming process. Switching always leads to a reset of the programmer. This is indicated by a short flicker of the red LED. A constantly glowing green LED indicates that the programmer is ready for use.

Do not push the RESET button throughout the programming process. The programming gets canceled and the data in the flash memory of the microcontroller can be invalid.

EXA-PROG

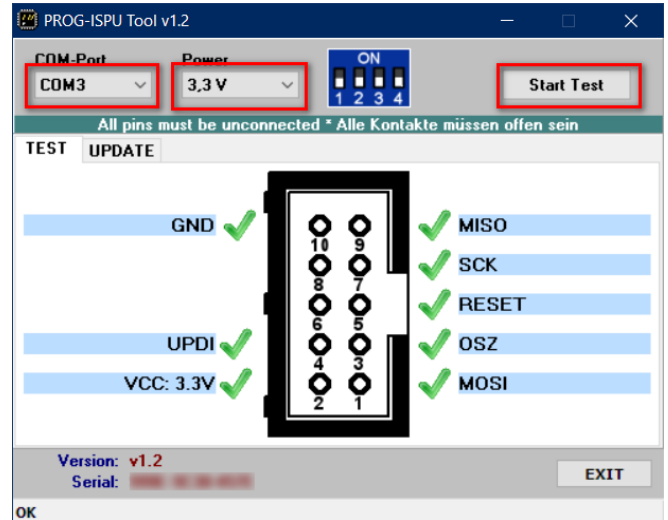
USB programming adapter for AVR-ISP, AVR-UPDI, STM32, LPC, ESP8266, ESP32

TESTING FUNCTION

If you have problems to program an attached microcontroller you can use the following option to check whether there is an error in the programming adapter. EXA-PROG contains integrated testing hardware which can be used to check the signal level of the programming interface.

For this please start the EXA-PROG Tool in Windows (file name: Exa_Tool_xxx.exe; xxx = version) which you can find in the EXA-PROG program package.

1. Select the tab "TEST".
 2. Select the COM port of the EXA-PROG. If you don't know exactly which COM port is used by your EXA-PROG you can identify it simply by comparing the COM port list before and after plugging the programmers into the PC.
 3. Select the voltage (power).
 4. Set the DIP switches on the programmer according to the pattern shown.
 5. Remove all wires from the EXA-PROG programming connector.
 6. Push the button "Start Test". All ports are now queried one after the other and the signal levels are measured.
- ⇒ For the test please leave the programming port of the programmer unconnected. There may not be any connections to a microcontroller or an external circuit.



Via the tab "UPDATE" you can also make an update of your EXA-PROG's firmware if this is necessary.

POSITIVE:

If all signal levels are ok the programming problem is probably located in the connection to the microcontroller or the reason is an incorrectly set programming software.

NEGATIVE:

In case the test is negative, first check if the right voltage was selected and if the DIP switches on the EXA-PROG agree with the switch position which gets displayed. Are there really no external components connected to the programming port? 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) at the programming port. This power level 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 seller you bought it from. Please always give the exact error description and perhaps a screenshot of the EXA-PROG tool.

PREVENTIVE:

Unfortunately, it happens again and again that the programming adapter gets destroyed throughout experiments with the programming of microcontrollers. That also happens to absolute experts which is why they usually have another programmer as a reserve so that the current project does not get slowed down. In order to prevent the defect of a programmer you should note the following things:

- Always connect the GND line (PIN10 of the programming port) to the circuit first in which there is the microcontroller that is to be programmed. So there is a potential equalization between the programmer and the circuit.
- Do not connect output signals on the programming connector (e.g. PIN1, PIN3, PIN5, PIN7) directly to GND or to the power supply. The output drivers could get destroyed.
- Make sure that that the pins on the programming connector are not exposed to a higher voltage than the voltage set on DIP switch 1.

EXA-PROG

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

LEDs

Green LED	Glows constantly when the selected operating mode is ready. Goes out briefly when the RESET button is pressed.
Red LED	Flickers when there are data transfers via RXD/TXD and throughout the programming process. 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. That is often needed when the boot of a freshly programmed controller is supposed 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 the LOW level as long as the button is pressed.

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

STM32:

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

A long button press (longer than a second) puts the connected microcontroller into boot mode. PIN3 (BOOT) is set to the HIGH level, a 50ms long reset pulse is generated and then PIN3 (BOOT) goes back to LOW. The green LED flickers briefly as an indicator that the reset was triggered.

NXP/LPC:

A short button press (less than a 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 on HIGH.

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

The green LED flickers briefly as an indicator that the reset was triggered.

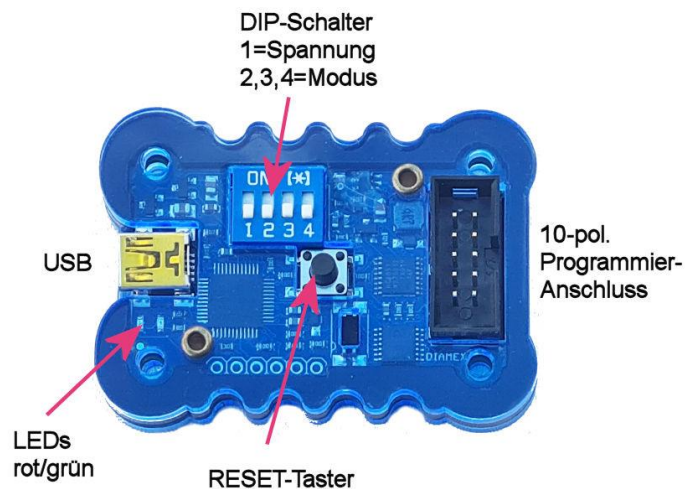
ESP:

A short button press (less than a 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 on HIGH.

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

The green LED flickers briefly as an indicator that the reset was triggered.

⇒ Never press the RESET button throughout a programming process. The programming gets aborted and the files in the flash memory of the microcontrollers can be invalid.

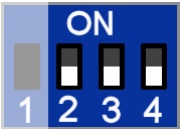
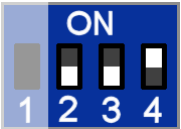
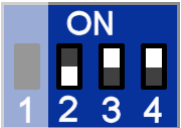
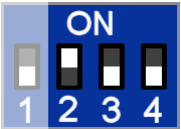
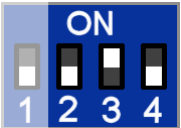


EXA-PROG

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

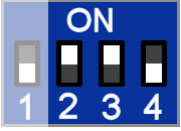
DIP SWITCH

SWITCH 2, 3, 4: Operating Mode

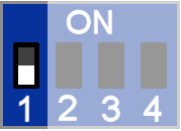
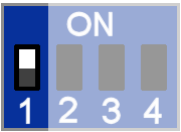
AVR-ISP	
	<p>2 = OFF, 3 = OFF, 4 = OFF</p> <p>Programmer for Atmel-AVR-Controller with ISP Interface</p> <p>These controllers are programmed via the ISP interface. With most AVR controllers the interface is equivalent to the pins of the SPI bus (Notice: There are a few controllers that have separate PDI/PDO pins). In addition to SCK, MISO, MOSI a reset line is also necessary. EXA-PROG emulates a STK500 programmer and is hereby compatible to ATMEL/MICROCHIP Studio and AVR-DUDE. 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-Controller with UPDI Interface</p> <p>This controller can be programmed via the UPDI interface. EXA-PROG emulates a jtag2updi programmer and is currently only compatible to 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-Controller with UPDI Interface (HV Mode)</p> <p>Mode to reactivate the UPDI interface if the function of the UPDI pins was changed in the fuses with ATtiny controllers. Please don't use with controllers with 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 ports (see data sheet of the controller to be programmed). The programmer is connected to the controller via the RX and TX connectors. In order to activate the integrated bootloader of the controllers automatically the BOOT0 and RESET connections are also required, this means that no jumpers or switches are required on the circuit.</p> <p>Please DIP1 = OFF, set programming voltage to 3.3V!</p>
NXP/LPC	
	<p>2 = OFF, 3 = ON, 4 = OFF</p> <p>Programmer for NXP/LPC Cortex-M Controller with UART Bootloader</p> <p>The LPC controller is programmed via the integrated Bootloader through the serial port UART0. The programmer is connected to the controller via the RX and TX connectors. In order to activate the integrated bootloader of the controllers automatically the P2[10] and RESET connections are also required, this means that no jumpers or switches are required on the circuit. EXA-PROG in connection with the tool "Flash Magic" 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 DIP1 = OFF, set 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 lines TXD0 and RXD0. In order to activate the bootloader automatically the lines GPIO0 und RESET (EN) are also required.</p> <p>Please DIP1 = OFF, set programming voltage to 3.3V!</p>

SWITCH 1: Voltage Level

3.3V	
	<p>1 = OFF</p> <p>There is a voltage of 3.3V at PIN 2 of the 10-pin programming connector. All signal levels on the programming lines have a maximum voltage of 3.3V.</p>
5V	
	<p>1 = ON</p> <p>There is a voltage of approx. 5V at PIN 2 of the 10-pin programming connector. All signal levels on the programming lines have a maximum voltage of approx. 5V.</p> <p>Please note that the exact voltage value may vary and depends on the voltage at the USB port. This value can fluctuate depending on the load and it can be between 4.5V and 5.2V. However, this is usually negligible when microcontrollers are being programmed.</p>

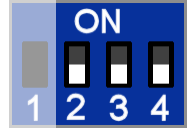
Notices:

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

EXA-PROG

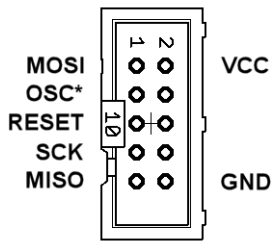
USB programming adapter for AVR-ISP, AVR-UPDI, STM32, LPC, ESP8266, ESP32

Connection of an AVR Controller with ISP Interface



DIP SWITCH: 2 = OFF, 3 = OFF, 4 = OFF

Connect the programming connector leads on the EXA-PROG directly to the pins on the controller:



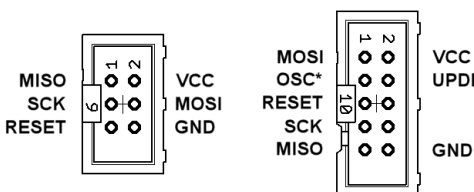
10 Pin Header	AVR Controller
PIN1 (MOSI)	MOSI or PDI
PIN5 (RESET)	RESET
PIN7 (SCK)	SCK
PIN9 (MISO)	MISO or PDO
optional: PIN3 (OSC)	XTAL1 (XTALIN)
PIN10 (GND)	GND
optional: PIN2 (VCC)	VCC

The pin numbers of the AVR controller vary depending on the type and the shape of the case. Please check out the data sheet in order 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 your circuit is running at. If the microcontroller is programmed in an external socket, both voltage values can be set.

NOTICES, TIPS:

- A few AVR controllers (e.g. AT90CAN32/64/128, ATmega64/128) have special programming pins PDI, PDO. Please use these lines instead of MISO/MOSI (see data sheet).
- EXA-PROG supports the adaptive SPI bitrate adjustment. An SPI bitrate set too high usually causes that the connected AVR controller is not recognized. The rule says that the SPI bit rate can be a maximum of a quarter of the clock rate of the controller. 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 does not respond anyway, please check the line connections first. MISO to MISO and MOSI to MOSI (that gets confused very often), 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 external clock. In this case, a crystal can be connected to the XTAL pins of the controller or put on the clock signal of PIN3 (OSC) of the 10-pin header on XTALIN or XTAL1 to generate the clock (see data sheet of the controller). The default clock frequency on PIN3 of the ISP connector is approximately 1.3MHz.
- EXA-PROG does not support ATXMega controllers with PDI interface as well as ATTiny controllers with TPI interface.
- Connect only PIN2 to the circuit to be programmed if it does not have its own power supply and if 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 using appropriate jumper cables.



adapter 10-pin to 6-pin ISP

(optional accessories, not included with the EXA-PROG)

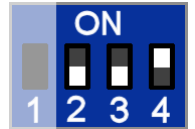


EXA-PROG

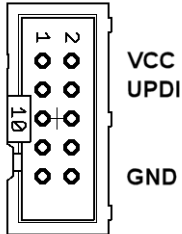
USB programming adapter for AVR-ISP, AVR-UPDI, STM32, LPC, ESP8266, ESP32

Connection of an AVR Controller with UPDI Interface

☞ **DIP SWITCH: 2 = OFF, 3 = OFF, 4 = ON**



Connect the programming connector leads on the EXA-PROG directly to the pins on the controller:



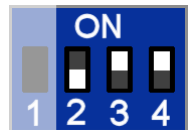
10 Pin Header	AVR Controller
PIN4 (UPDI)	UPDI (RESET, PORT)
PIN10 (GND)	GND
optional: PIN2 (VCC)	VCC

The pin numbers of the AVR controller vary depending on the type and the shape of the case. Please check out the data sheet in order 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 your circuit is running at. If the microcontroller is programmed in an external socket, both voltage values can be set.

UPDI-HV MODE

☞ **DIP SWITCH: 2 = OFF, 3 = ON, 4 = ON**



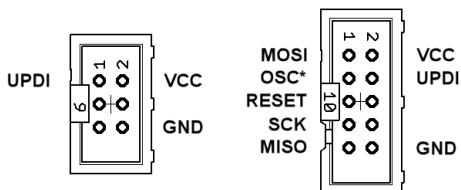
With ATtiny controllers that do not have a separate UPDI pin (e.g. Tiny412, Tiny1614, Tiny3216) the pin can be switched to an alternative function via the fuses. After that, programming in the standard UPDI mode is no longer possible. In this case, select the UPDI high-voltage mode, in which the UPDI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

If the controller is to be programmed in a circuit, make sure that there may not be any components on the UPDI pin.

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

NOTICES, TIPS:

- Connect only PIN2 to the circuit to be programmed if it does not have its own power supply and if the expected power consumption does not exceed the maximum limit of the EXA-PROG.
- In the UPDI mode the software interface of the EXA-PROG emulates a jtag2updi interface. This is not supported by ATMEL/MICROCHIP Studio. Instead, please use the tool AVRDUDE for programming. This is used in various development environments like Arduino or Bascom. You can find a lot of information on 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 using appropriate jumper cables.



adapter 10-pin to 6-pin UPDI

(optional accessories, not included with the EXA-PROG)

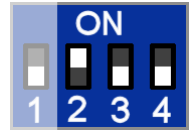


EXA-PROG

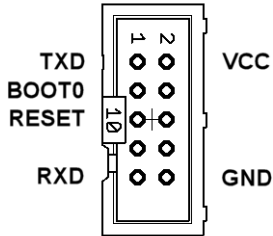
USB programming adapter for AVR-ISP, AVR-UPDI, STM32, LPC, ESP8266, ESP32

Connection of an STM32 Controller

☞ **DIP SWITCH: 2 = ON, 3 = OFF, 4 = OFF**



Connect the programming connector leads on the EXA-PROG directly to the pins on the controller:



10 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	
optional: PIN2 (VCC)	VCC	

The pin numbers of the STM32 vary depending on the type and the shape of the case. Please check out the data sheet of the STM32 in order to determine the pin numbers that match your controller.

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

NOTICES, TIPS:

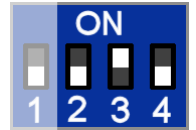
- So that the bootloader can be started it must be taken care of that BOOT1 (PB2) is at ground potential during the RESET (no longer available with newer controller types). If the pin is not used otherwise you can connect it directly to GND. Otherwise, please connect it to GND via a 10 kOhm resistor.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

EXA-PROG

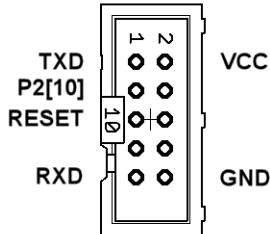
USB programming adapter for AVR-ISP, AVR-UPDI, STM32, LPC, ESP8266, ESP32

Connection of an NXP/LPC Controller

☞ **DIP SWITCH: 2 = OFF, 3 = ON, 4 = OFF**



Connect the programming connector leads on the EXA-PROG directly to the pins on the controller:



10 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	
optional: PIN2 (VCC)	VCC	

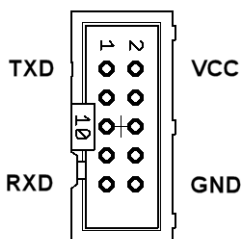
The pin numbers of the NXP controller vary depending on the shape of the case. Please check out the data sheet in order to determine the pin numbers that match your controller.

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

NOTICES, TIPS:

- In order to program the NXP controller, it must be supplied with 3.3 volts.
- A crystal does not have to be connected to the NXP controller for programming. The controller's bootloader uses the internal RC oscillator to generate the system clock.
- Controlling the BOOT and RESET lines works very well with the tool "Flash Magic".
- 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.
- This programmer does not support 8-bit controllers (LPC9xx) and NXP controllers without an ARM Cortex core.

EXA-PROG as USB/Serial Converter



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

The maximum transmission rate is 1 Mbit.

Supported as transmission parameters: 7 + 8 bit, Parity OFF, EVEN, ODD, 1 or 2 Stopbits.

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

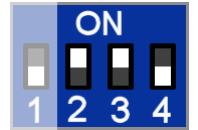
DIP switch 1: OFF = 3.3V, ON = 5V. Be sure to set the same voltage here your circuit is running at.

EXA-PROG

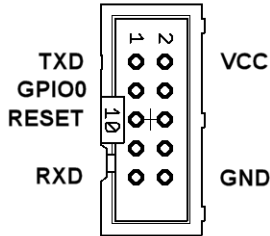
USB programming adapter for AVR-ISP, AVR-UPDI, STM32, LPC, ESP8266, ESP32

Connection of an ESP Controller

☞ **DIP SWITCH: 2 = ON, 3 = ON, 4 = OFF**



Connect the programming connector leads on the EXA-PROG directly to the pins on the controller or on an ESP8266 or ESP32 module respectively:



10 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
optional: PIN2 (VCC)	VCC	VCC

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

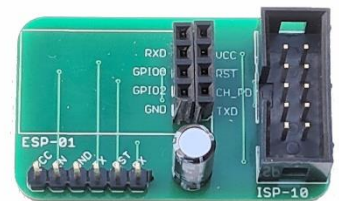
NOTICES, TIPS:

- The RST pin of the ESP8266, or the EN-Pin of the ESP32 respectively, should be placed against +3.3V with a 10 kOhm resistor. A capacitor to ground (as recommended in many circuits) must not be on this pin because the RESET is controlled by the EXA-PROG.
- In order to program the ESP8266 or ESP32, it must be supplied with 3.3 volts. If the current consumption is not higher than 300mA, then the controller, including the circuitry, 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. However, activation can also be triggered using the reset button on the EXA-PROG if this does not work via the software.

Adapter 10-pin to ESP

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

(optional accessories, not included with the EXA-PROG)



EXA-PROG

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

NOTICE

© Erwin Reuß; Folker Stange. Use and disclosure of this information, even in part, only with the permission of the copyright holder. All the stated product names and trademarks remain in property of their respective owners and are used here for description only.

DISCLAIMER

The manufacturer assumes no liability for damage that may result from the use of the EXA-PROG. The English version of this manual is a translation. For legal issues the authoritative version of these instructions is the German one.

LINKS

LED-Genial shop

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

Diamex shop

<http://www.diamex.de>

Download AVR/ATMEL/MICROCHIP Studio

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

UPDI programming with AVRDUDE

<https://github.com/EITangas/jtag2updi#using-with-avrdude>

ESPTOOL for ESP8266 and ESP32

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

Distribution



DIAMEX Produktion und Handel GmbH

Innovationspark Wuhlheide
Köpenicker Straße 325, Haus 41
12555 Berlin

Phone: 030-65762631

E-Mail: info@diamex.de

Website: <http://www.diamex.de>

Manufacturer



www.tremex.de

Köpenicker Str. 325 12555 Berlin
Tel. 030-65762631

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

EXA-PROG

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