Serial

[Communication]

Description

Used for communication between the Arduino board and a computer or other devices. All Arduino boards have at least one serial port (also known as a UART or USART), and some have several.

| BOARD | USB CDC NAME | SERIAL PINS | SERIAL1 PINS | SERIAL2 PINS | SERIAL3 PINS |
| --- | --- | --- | --- | --- | --- |
| Uno, Nano, Mini |  | 0(RX), 1(TX) |  |  |  |
| Mega |  | 0(RX), 1(TX) | 19(RX), 18(TX) | 17(RX), 16(TX) | 15(RX), 14(TX) |
| Leonardo, Micro, Yún | Serial |  | 0(RX), 1(TX) |  |  |
| Uno WiFi Rev.2 |  | Connected to USB | 0(RX), 1(TX) | Connected to NINA |  |
| MKR boards | Serial |  | 13(RX), 14(TX) |  |  |
| Zero | SerialUSB (Native USB Port only) | Connected to Programming Port | 0(RX), 1(TX) |  |  |
| Due | SerialUSB (Native USB Port only) | 0(RX), 1(TX) | 19(RX), 18(TX) | 17(RX), 16(TX) | 15(RX), 14(TX) |
| 101 | Serial |  | 0(RX), 1(TX) |  |  |

On Uno, Nano, Mini, and Mega, pins 0 and 1 are used for communication with the computer. Connecting anything to these pins can interfere with that communication, including causing failed uploads to the board.

You can use the Arduino environment’s built-in serial monitor to communicate with an Arduino board. Click the serial monitor button in the toolbar and select the same baud rate used in the call to begin().

Serial communication on pins TX/RX uses TTL logic levels (5V or 3.3V depending on the board). Don’t connect these pins directly to an RS232 serial port; they operate at +/- 12V and can damage your Arduino board.

To use these extra serial ports to communicate with your personal computer, you will need an additional USB-to-serial adaptor, as they are not connected to the Mega’s USB-to-serial adaptor. To use them to communicate with an external TTL serial device, connect the TX pin to your device’s RX pin, the RX to your device’s TX pin, and the ground of your Mega to your device’s ground.

Functions

[if(Serial)](https://www.arduino.cc/reference/en/language/functions/communication/serial/ifserial)  
[available()](https://www.arduino.cc/reference/en/language/functions/communication/serial/available)  
[availableForWrite()](https://www.arduino.cc/reference/en/language/functions/communication/serial/availableforwrite)  
[begin()](https://www.arduino.cc/reference/en/language/functions/communication/serial/begin)  
[end()](https://www.arduino.cc/reference/en/language/functions/communication/serial/end)  
[find()](https://www.arduino.cc/reference/en/language/functions/communication/serial/find)  
[findUntil()](https://www.arduino.cc/reference/en/language/functions/communication/serial/finduntil)  
[flush()](https://www.arduino.cc/reference/en/language/functions/communication/serial/flush)  
[parseFloat()](https://www.arduino.cc/reference/en/language/functions/communication/serial/parsefloat)  
[parseInt()](https://www.arduino.cc/reference/en/language/functions/communication/serial/parseint)  
[peek()](https://www.arduino.cc/reference/en/language/functions/communication/serial/peek)  
[print()](https://www.arduino.cc/reference/en/language/functions/communication/serial/print)  
[println()](https://www.arduino.cc/reference/en/language/functions/communication/serial/println)  
[read()](https://www.arduino.cc/reference/en/language/functions/communication/serial/read)  
[readBytes()](https://www.arduino.cc/reference/en/language/functions/communication/serial/readbytes)  
[readBytesUntil()](https://www.arduino.cc/reference/en/language/functions/communication/serial/readbytesuntil)  
[readString()](https://www.arduino.cc/reference/en/language/functions/communication/serial/readstring)  
[readStringUntil()](https://www.arduino.cc/reference/en/language/functions/communication/serial/readstringuntil)  
[setTimeout()](https://www.arduino.cc/reference/en/language/functions/communication/serial/settimeout)  
[write()](https://www.arduino.cc/reference/en/language/functions/communication/serial/write)  
[serialEvent()](https://www.arduino.cc/reference/en/language/functions/communication/serial/serialevent)