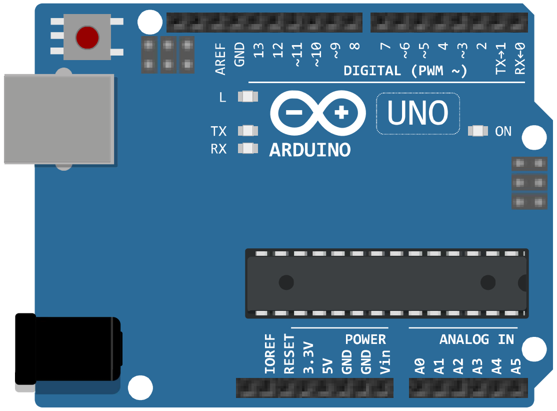
Diagram Reference wokwi-arduino-uno

wokwi-arduino-uno Reference

Arduino Uno is the most popular board in the Arduino family. It is powered by the ATmega328p chip, which has 32K bytes of Flash program memory, 2k bytes of SRAM and 1K bytes of EEPROM.



Pin names

Pins 0 to 13 are digital GPIO pins. Pins A0 to A5 double as analog input pins, in addition to being digital GPIO pins.

There are three ground pins: GND.1, which is on top of the board, next to pin 13, and GND.2/GND.3, which are on the bottom.

Pins VIN / 5V are connected to the positive power supply.

Pins 3.3V / IOREF / AREF / RESET are not available in the simulation.

Digital pins 3, 5, 6, 9, 10, and 11 have hardware PWM support.

Some of the digital pins also have additional functions:

Pin	Function	Signal
0	Serial (USART)	RX
1	Serial (USART)	TX
2	External interrupt	INT0
3	External interrupt	INT1
10	SPI	SS (Chip select)
11	SPI	MOSI
12	SPI	MISO
13	SPI	SCLK (Clock)
A4	I2C	SDA (Data)
A5	I2C	SCL (Clock)

On board LEDs

The board includes four LEDs:

LED	Function
L	Connected to digital pin 13
RX	Serial RX Activity
TX	Serial TX Activity
ON	Power LED. Always on while the simulation is running

In general, only the "L" LED can be controlled by the user's code. You can use the `LED_BUILTIN` constant to reference it from your code:

```
pinMode(LED_BUILTIN, OUTPUT);
digitalWrite(LED_BUILTIN, HIGH);
```

See Blink for a complete code example.

Attributes

Name	Description	Default value
frequency	MCU clock frequency, in hertz. Common values: "8m", "16m", and "20m" *	"16m"

* Many Arduino libraries assume 16 MHz clock frequency. Changing the clock frequency will void your warranty!

Simulation features

The Arduino Uno is simulated using the AVR8js Library. The table below summarizes the status of features:

Peripheral	Status	Notes
Processor	✓	
GPIO	✓	Including External/Pin Change Interrupts
8-bit timers	✓	Timer0, Timer2
16-bit timer	✓	Timer1
Watchdog Timer	✓	Usage example
USART	✓	
SPI	●	Master mode only
I2C	●	Master mode only
EEPROM	✓	
Clock Prescale	✓	
ADC	✓	Used by analogRead()

Peripheral	Status	Notes
Analog Comparator	❌	
GDB Debugging	✔️	See the GDB Debugging Guide

Legend:
✔️ Simulated
🟡 Simulated, but see notes
❌ Not implemented

If you need any of the missing features, please open an issue on the AVR8js repo or reach out on Discord.

Serial Monitor

You can use the Serial Monitor to receive information from your Arduino code, such as debug print. You can also use it to send information to your code, such as textual commands.

For more information and code samples, check out the Serial Monitor guide. It also explains how to configure the Serial monitor, e.g. set the line ending characters.

Libraries

The simulator supports many popular Arduino libraries. For a complete list, see the Libraries guides.

Simulator examples

- Arduino Blink

Edit this page