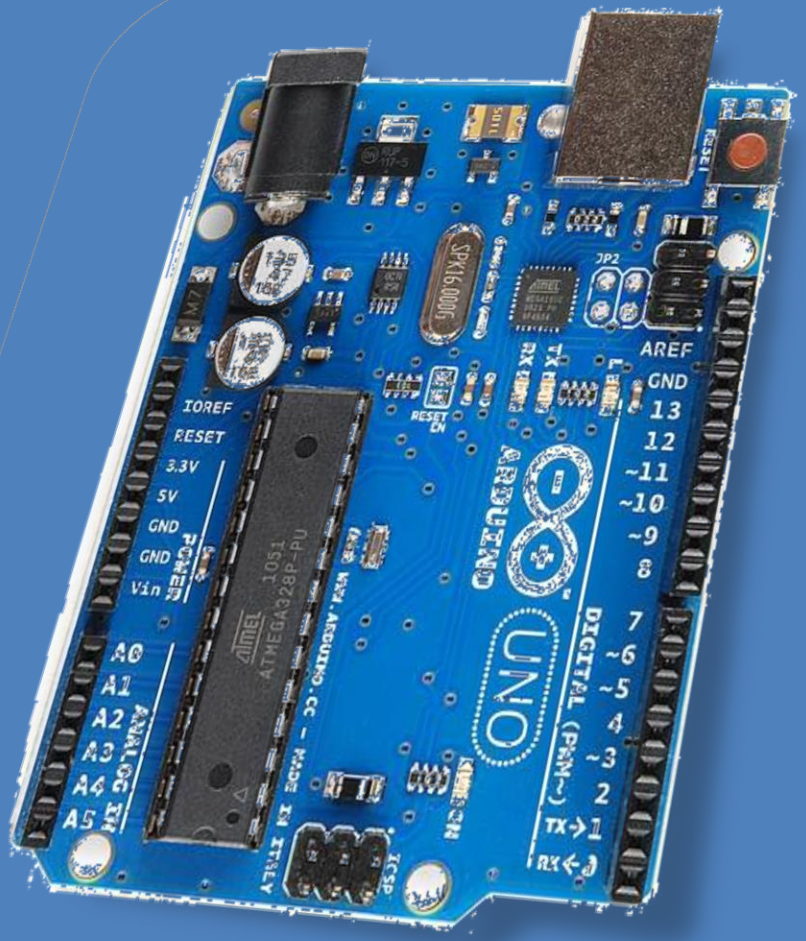
Arduino

Presented by

Dr. Amany AbdElSamea

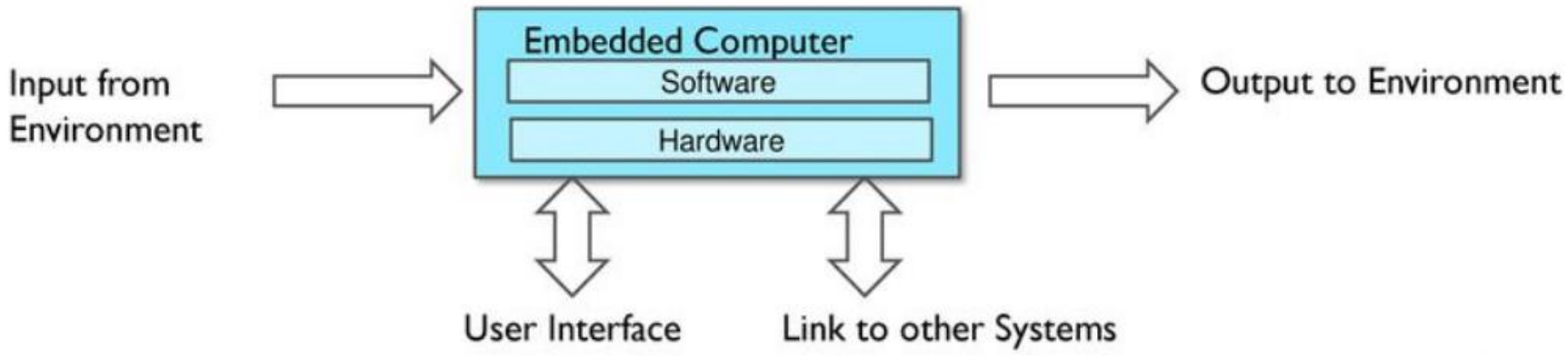


Outline

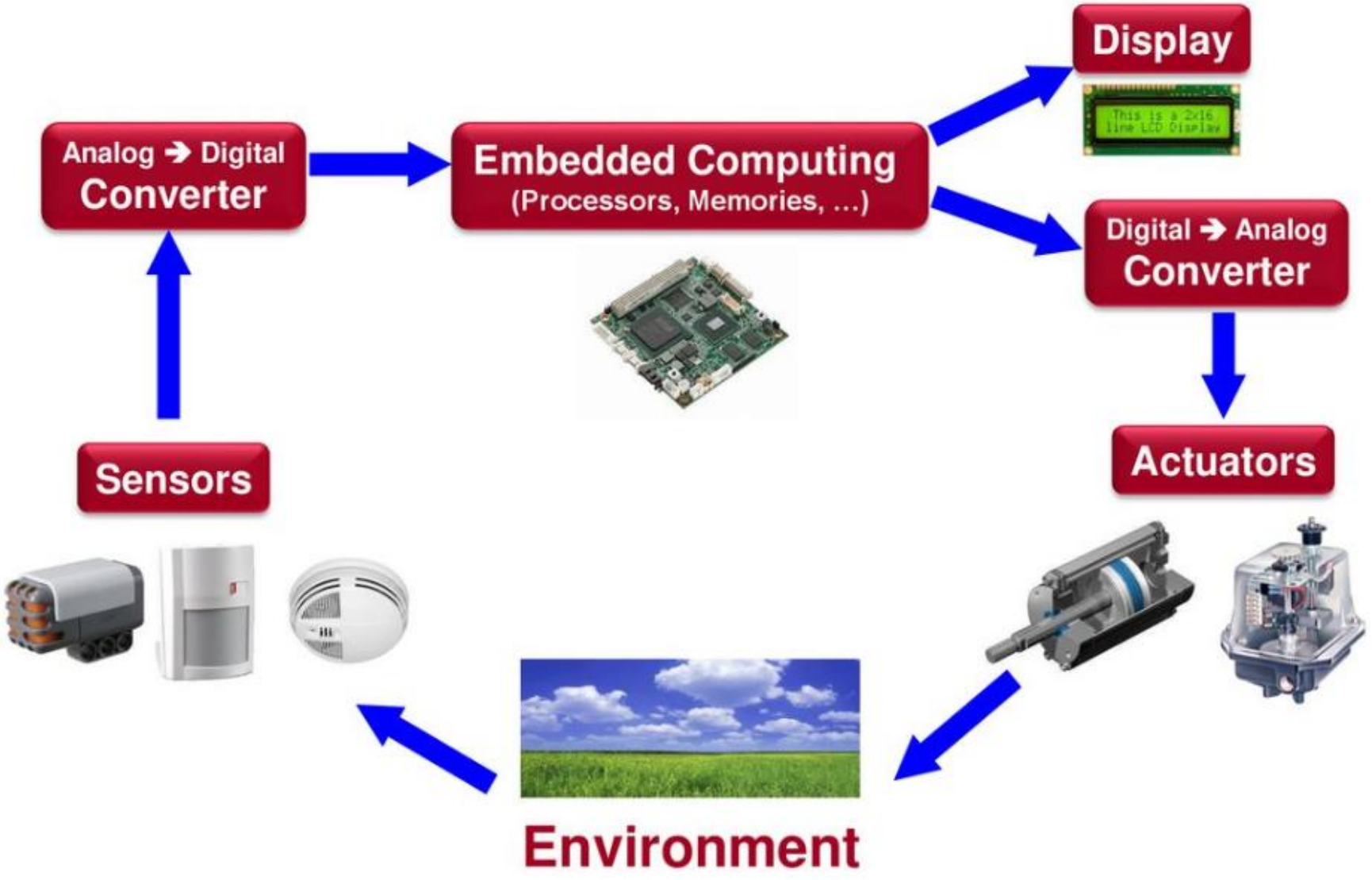
- Introduction to embedded systems
- Microprocessor vs. Microcontroller
- What is Arduino?
- Difference between Arduino Board
- Arduino Uno
- Arduino Sensors
- Arduino IDE
- Arduino and TinkerCAD

Introduction to Embedded Systems

- What is an Embedded System?
  - Application-specific computer system
  - Built into a larger system
  - Often with real-time computing constraints
- Why add a computer to a larger system?
  - Better performance
  - More functions and features
  - Low cost
  - More dependability



Embedded Systems Components

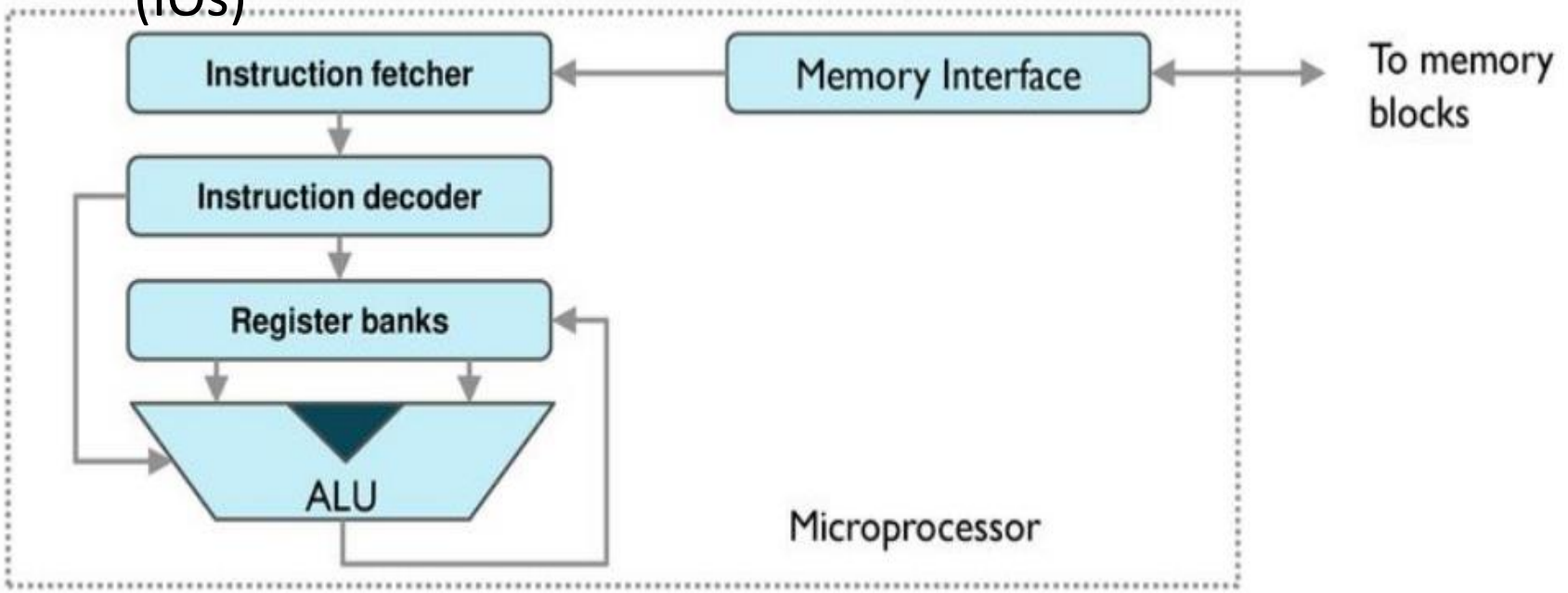


Common Characteristics of Embedded Systems

- Dependability
- Single-functioned (dedicated System)
  - Executes a single program, repeatedly
- Tightly-constrained (Efficient)
  - Low cost, low power, small, fast, etc.
- Reactive and real-time
  - Continually reacts to changes in the system's environment
  - Must compute certain results in real-time without delay

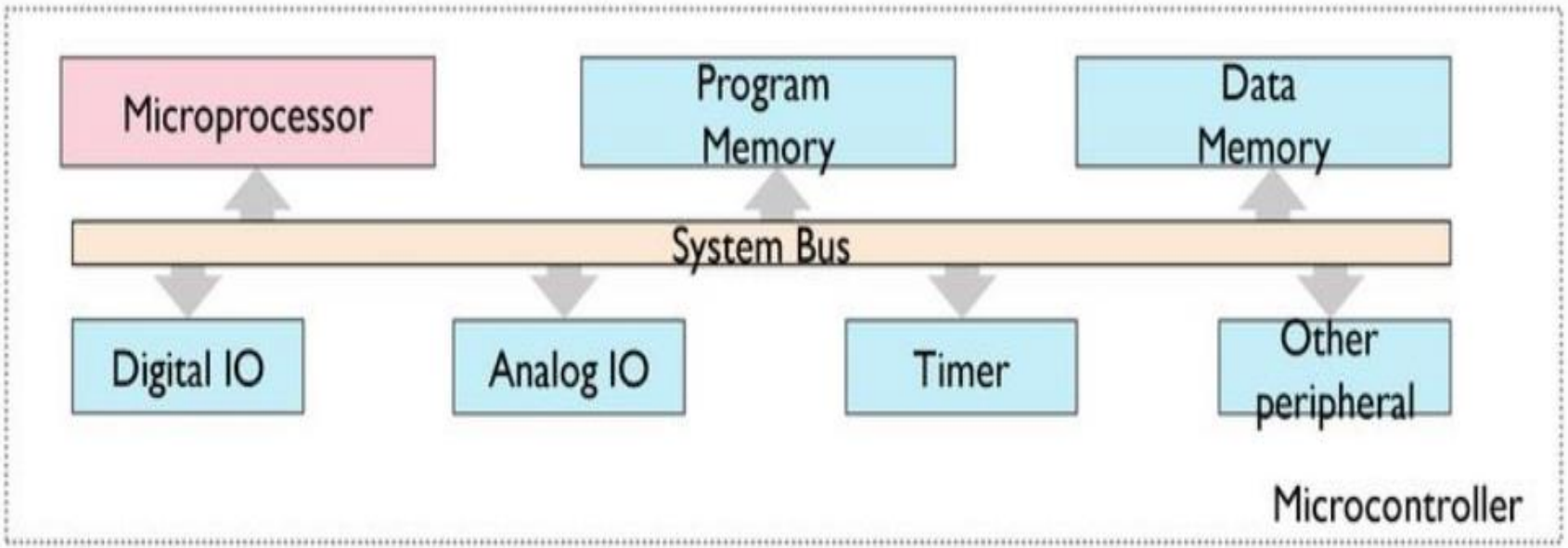
Microprocessor

- Microprocessor
  - Defined typically as a single processor core that supports at least instruction fetching, decoding, and executing
  - Normally can be used for general purpose computing but needs to be supported with memories and Input/Outputs (IOs)



Microcontroller

- Microcontroller (MCU)
  - Typically has a single processor core
  - Has memory blocks, Digital IOs, analog IOs and other basic peripherals
  - Typically used for basic control purpose, such as embedded applications



Microprocessor vs. Microcontroller

Microprocessor	Microcontroller
a. Microprocessors are widely used in computer systems.	a. Microcontroller is widely used in embedded systems.
b. It has only a CPU embedded into it.	b. It has a CPU, a fixed amount of RAM, ROM and other peripherals all embedded on it.
c. In case of microprocessors we have to connect all the components externally so the circuit becomes large and complex.	c. As all the components are internally connected in microcontroller so the circuit size is small.
d. It consumes more power.	d. It consumes less power than a microprocessor.
e. It has very less internal register storage so it has to rely on external storage. So all memory operations are carried out using memory based external commands which results in high processing time.	e. It has many registers so processing time is less.

What is Arduino?

- Arduino is a microcontroller-based open source electronic prototyping board which can be programmed with an easy-to-use Arduino IDE

- Arduino consists of both a physical programmable circuit board and a piece of software, or IDE. The Arduino IDE uses a simplified version of C++, making it easier to learn.

- With Arduino, you can control almost everything around you be it simple LED or giant Robots.

Different types of Arduino

Arduino Uno – R3

Arduino Ethernet

Arduino Due

LilyPad Arduino 328

Arduino Pro

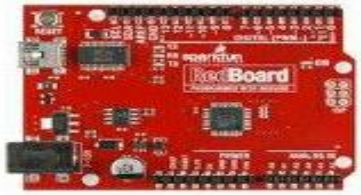
Arduino Pro Mini

Arduino Mini

Arduino Pro Micro

Arduino Leonardo

Arduino Fio



Arduino RedBoard

Arduino Mega

Difference between Arduino boards

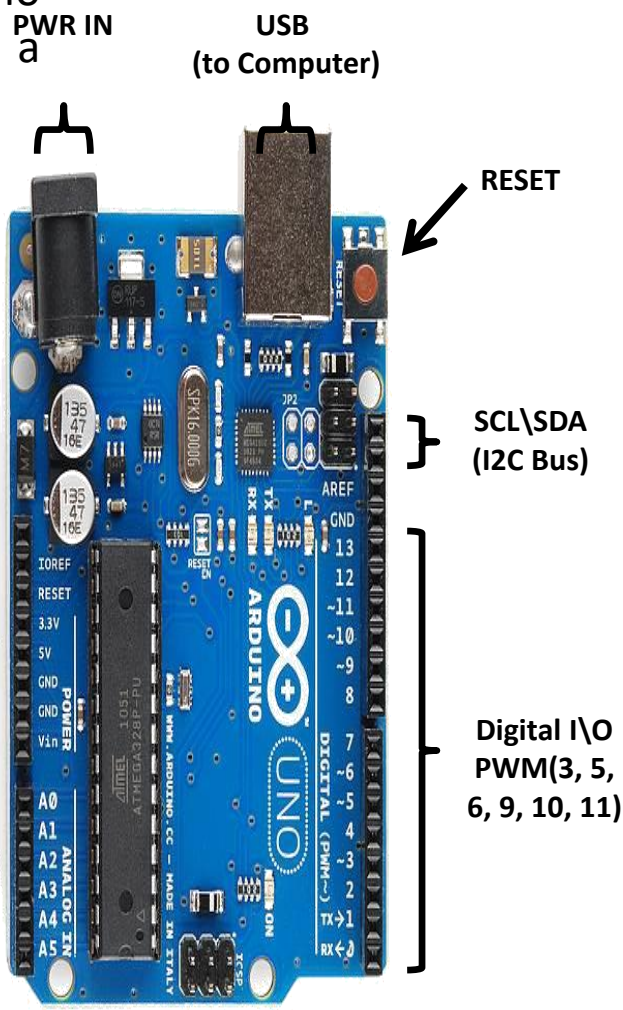
Arduino Board	Processor	Memory	Digital I/O	Analogue I/O
Arduino Uno	16Mhz ATmega328	2KB SRAM, 32KB flash	14	6 input, 0 output
Arduino Due	84MHz AT91SAM3X8E	96KB SRAM, 512KB flash	54	12 input, 2 output
Arduino Mega	16MHz ATmega2560	8KB SRAM, 256KB flash	54	16 input, 0 output
Arduino Leonardo	16MHz ATmega32u4	2.5KB SRAM, 32KB flash	20	12 input, 0 output

Arduino Uno

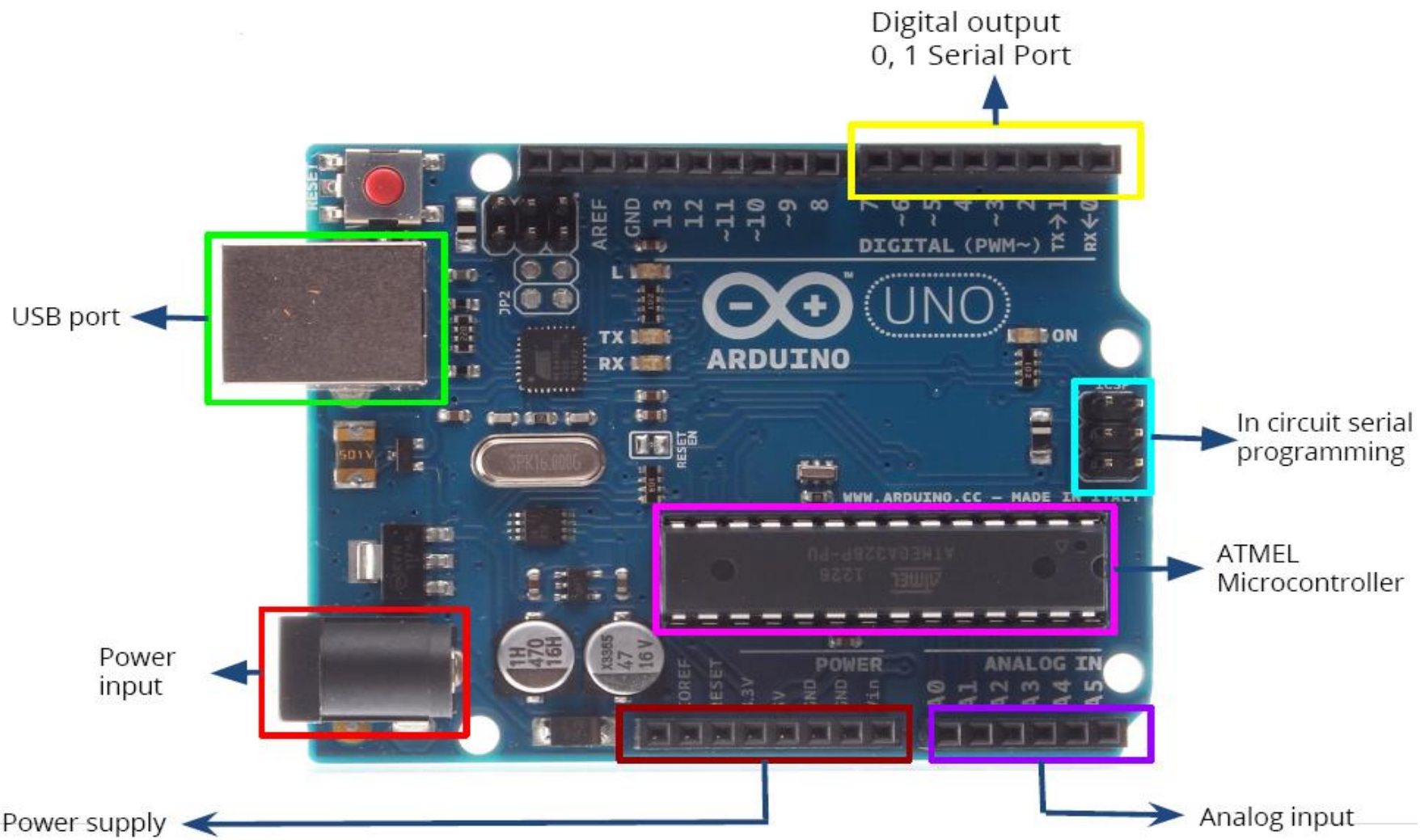
- The Uno is one of the more popular boards in the Arduino family and a great choice for beginners. The Arduino Uno is a microcontroller board based on the ATmega328 .

- What does it have?
  - -14 Digital In/Out pins (6 can be used as Pulse Width Modulation (PWM) is a technique for getting analog results with digital means)
  - -6 Analog Inputs
  - -A USB Connection
  - -A Power Jack
  - -Reset Button
  - -On-board LED
  - -SCL/SDA pins (Serial Clock/ Serial Data pins)

- In short, it contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.



Arduino Uno Board



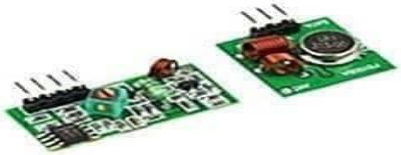
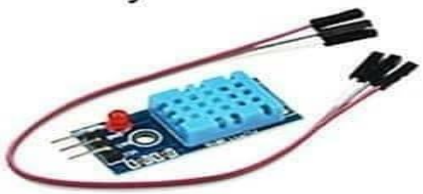
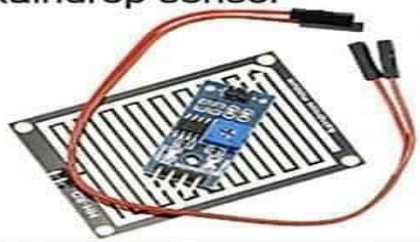
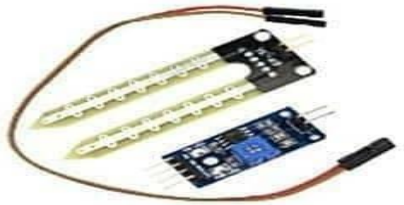
Arduino Sensors

- The sensors are defined as a machine, module, or a device that detect changes in the environment. The sensors transfer those changes to the electronic devices in the form of a signal.

- The sensors are used to measure the physical quantities, such as pressure, temperature, sound, humidity, and light, etc.

- The data signal runs from the sensor to the output pins of the Arduino. The data is further recorded by the Arduino.

Arduino Sensors cont.,

Ultrasonic module 	Human body sensor module 	Tilt sensor 	Photosensitive sensor 
Smoke sensor 	Infrared barrier sensor 	Vibration sensor 	Sound sensor 
1 path search sensor 	Flame sensor 	Laser Head Sensor 	Clock module 
Super regenerative module 	Temperature and humidity sensor 	Raindrop sensor 	Soil Sensors 

How Arduino is Programmed

Using a software called Arduino IDE

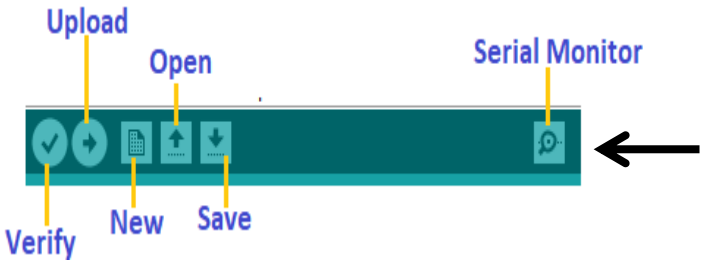
Arduino IDE

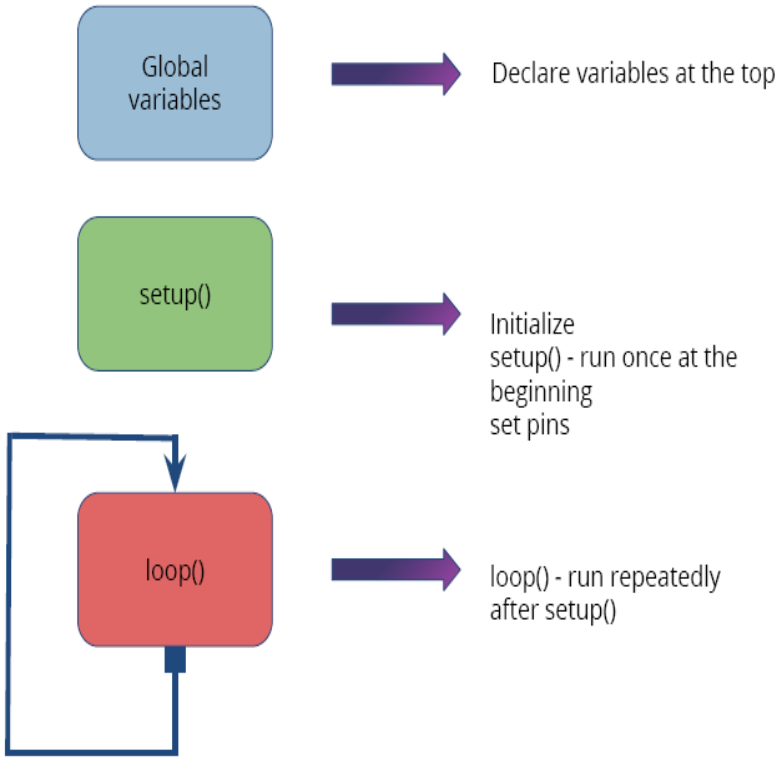
- The Arduino IDE is an open-source software, which is used to write and upload code to the Arduino boards.

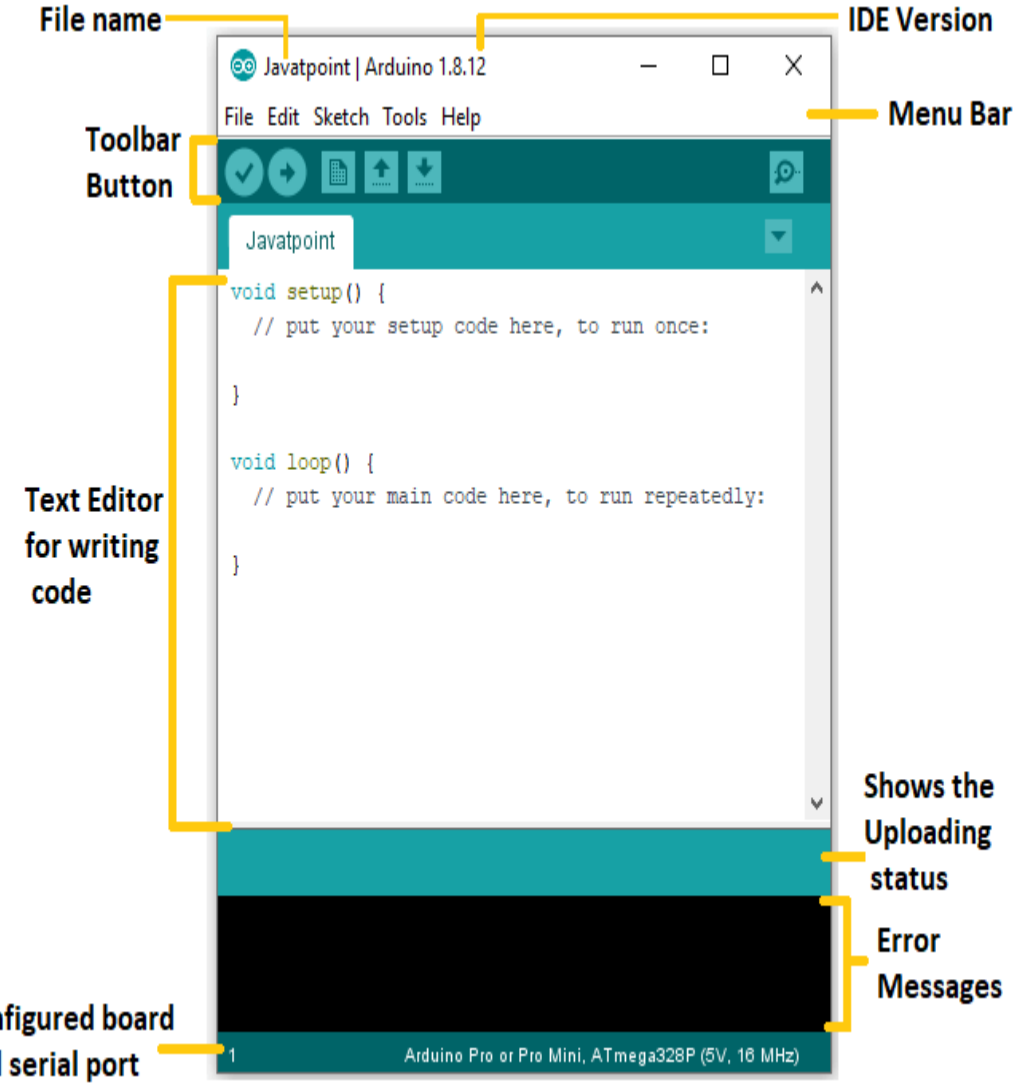
- The IDE application is suitable for different operating systems such as Windows, Mac OS X, and Linux. It supports the programming languages C and C++. Here, IDE stands for Integrated Development Environment.

- The program or code written in the Arduino IDE is often called as sketching. We need to connect the Genuino and Arduino board with the IDE to upload the sketch written in the Arduino IDE software.
- The sketch is saved with the extension '.ino.'

Arduino IDE cont.,



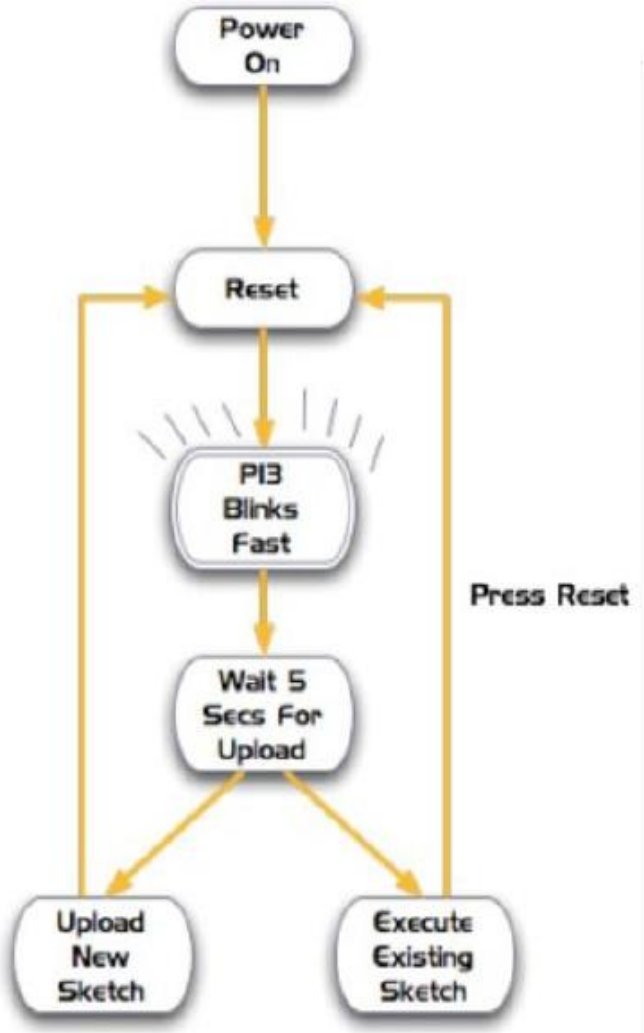




Useful Functions

pinMode()	set pin as input or output
digitalWrite()	set a digital pin high/low
digitalRead()	read a digital pin's state
analogRead()	read an analog pin
analogWrite()	write an "analog" PWM value
delay()	wait an amount of time
millis()	get the current time

Blinking LED



Blink | Arduino 1.8.15

File Edit Sketch Tools Help

Blink §

```
int ledpin = 13;

void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(ledpin, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(ledpin, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(5000);                       // wait for 5 seconds
  digitalWrite(ledpin, LOW);    // turn the LED off by making the voltage LOW
  delay(5000);                       // wait for 5 seconds
}
```

Done compiling.

```
Sketch uses 936 bytes (2%) of program storage space. Maximum is 32256 bytes.
Global variables use 9 bytes (0%) of dynamic memory, leaving 2039 bytes for local variables. Maximum is 2048 bytes.
```

13

How to Get Started

If you have the board follow up the following steps:

- *Arduino board*
  - *USB cable*
  - *DC power supplies*

- *Download the Arduino software (Arduino IDE)*
- *Read carefully*
  - *Instructions to install and setup the Arduino board with the computer and software*
  - *Download the Arduino IDE software*
- *Plug in the Arduino board into the computer after writing the sketch.*

If you don't have the board so use TinkerCAD simulator

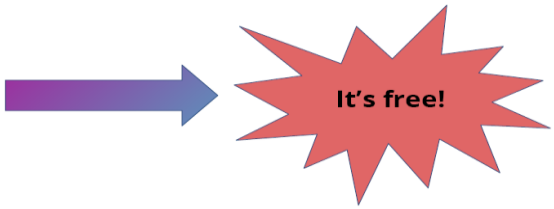
TinkerCAD Arduino

TinkerCAD Circuits:

AUTODESK® TINKERCAD®

- Tinkercad is an excellent tool that allows you to simulate Arduino-based systems
- You can simulate all exercises and even your own designs before trying them on real hardware.
- It also allows you to do programming using blocks.
- You can download / copy-paste the generated code later into Arduino IDE to program the real Arduino board, rather than having to write it from scratch.

Make an account in tinkercad



tinkercad.com/dashboard?type=circuits&collection=designs

ALTODESK TINKERCAD

Gallery Blog Learn Teach

yarelis.acevedo

Tinkercad Lesson Plans

Tinkercad lesson plans are ready to use online or in the classroom. Discover curriculum developed in partnership with teachers. Learn more

Search designs...

3D Designs

Circuits

Codeblocks NEW

Lessons

Your Classes

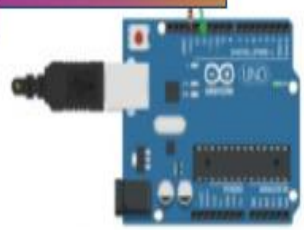
Projects

Create project

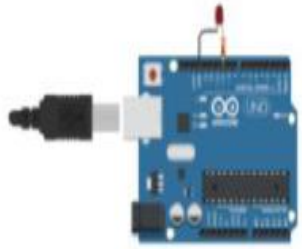
Circuits

Create new Circuit

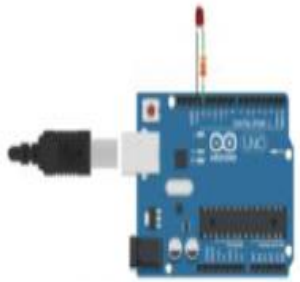
Select



Incredible Snicket
9 minutes ago
Private

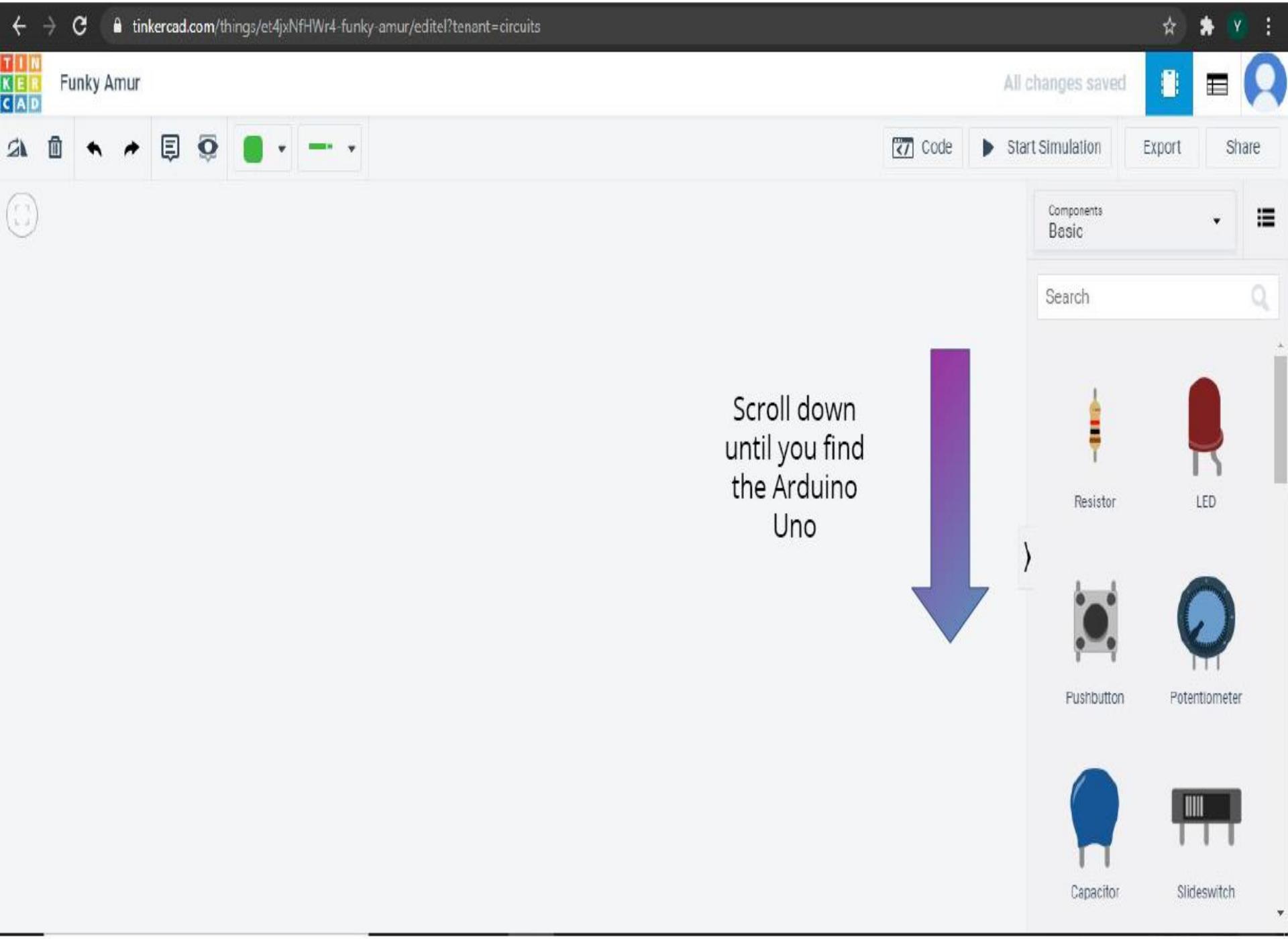


Copy of Arduino Blink
11 minutes ago
Private



Copy of Blink (Blocks)
2 months ago
Private

Fabulous Stantia-Borwo
7 months ago
Private


tinkercad.com/things/et4jxNfHWr4-funky-amur/editel?tenant=circuits
Funky Amur
All changes saved
Code
Start Simulation
Export
Share
Components
Basic
Search
Scroll down until you find the Arduino Uno
Resistor
LED
Pushbutton
Potentiometer
Capacitor
Slideswitch

tinkercad.com/things/et4jxNfHWr4-funky-amur/editel?tenant=circuits

Funky Amur

All changes saved

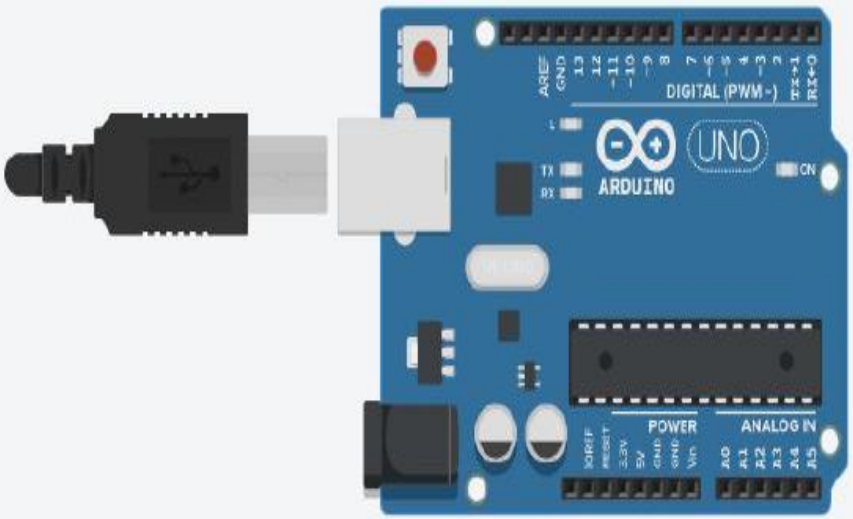
Code | Start Simulation | Export | Share

Arduino Uno R3

Name 1

Components
Basic

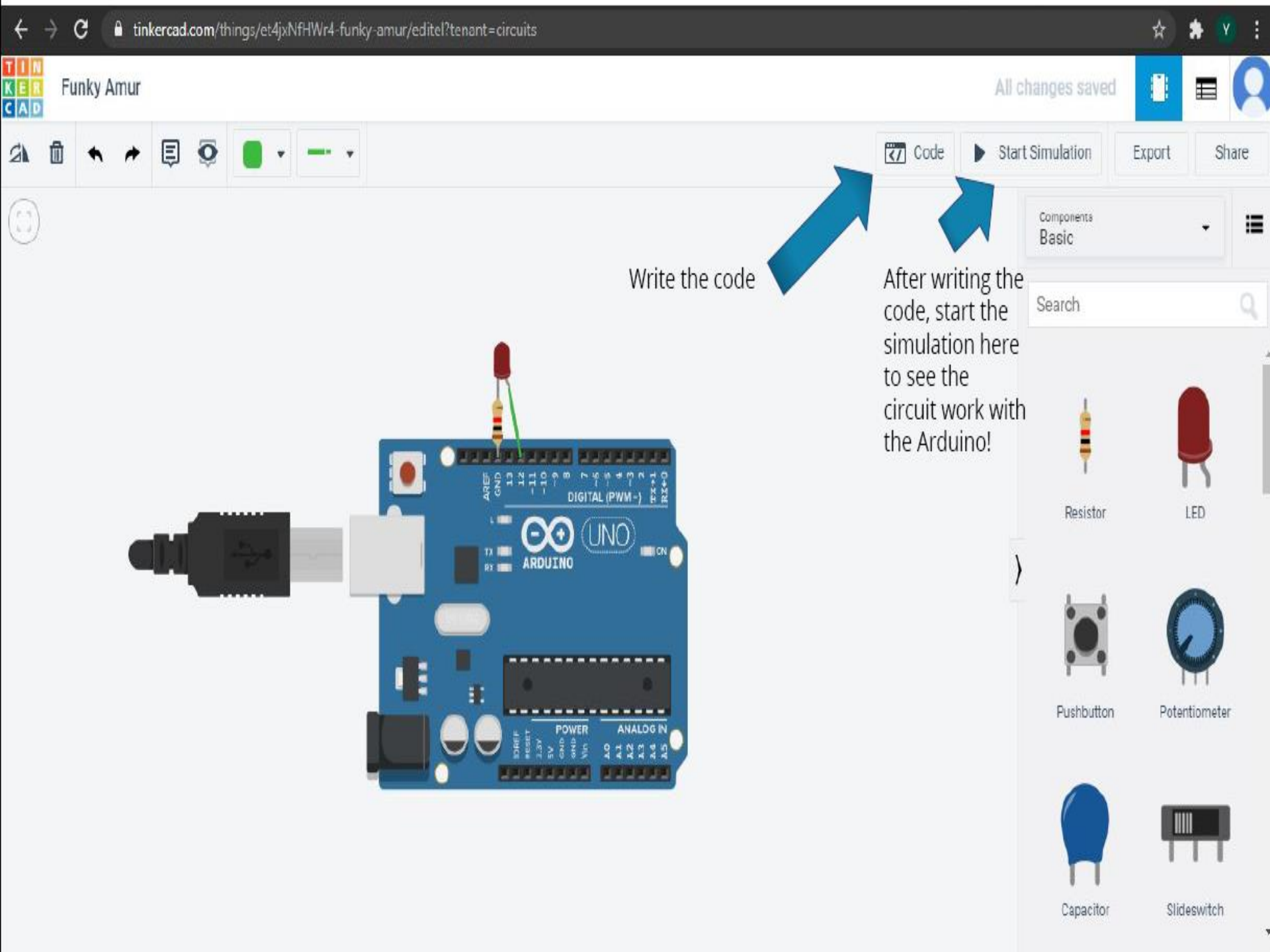
Search



Find the resistor and the LED

Connect the resistor to the GND (Ground)

Connect the LED: Shorter leg with the resistor and longer leg to pin 12

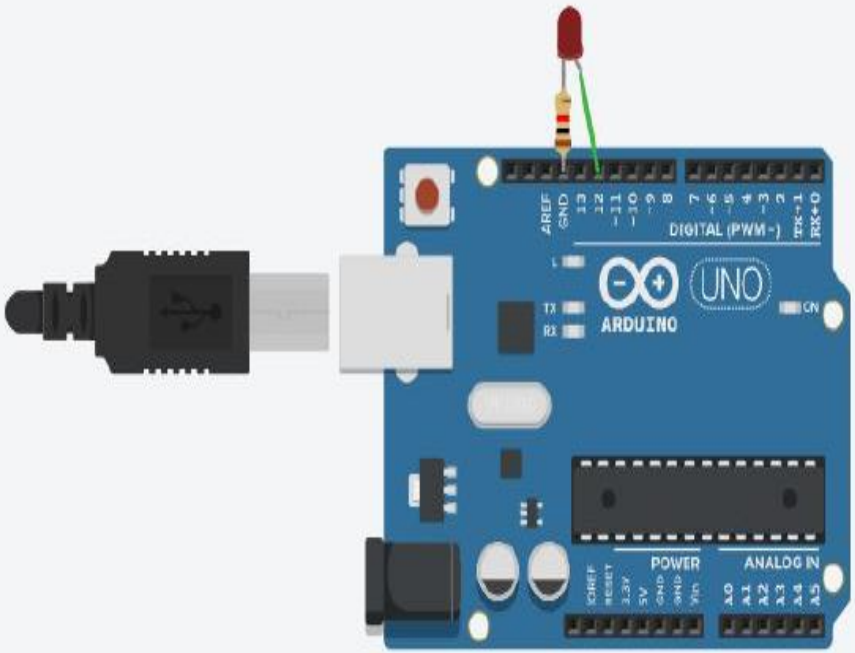

tinkercad.com/things/et4jxNfHWr4-funky-amur/editel?tenant=circuits
Funky Amur
All changes saved
Code
Start Simulation
Export
Share
Write the code
After writing the code, start the simulation here to see the circuit work with the Arduino!
Components
Basic
Search
Resistor
LED
Pushbutton
Potentiometer
Capacitor
Slideswitch
DIGITAL (PWM~)
UNO
ARDUINO
POWER
ANALOG IN

tinkercad.com/things/et4jxNfHWr4-funky-amur/editel?tenant=circuits

Funky Amur

All changes saved

Code | Start Simulation | Export | Share



Text | 1 (Arduino Uno R3)

```
// C++ code
//
void setup()
{
  pinMode(13, OUTPUT);
}

void loop()
{
  digitalWrite(13, HIGH);
  delay(1000); // Wait for 1000 millisecond(s)
  digitalWrite(13, LOW);
  delay(1000); // Wait for 1000 millisecond(s)
}
```

Serial Monitor

Questions