Line follower competition.

Introduction to…

Arduino
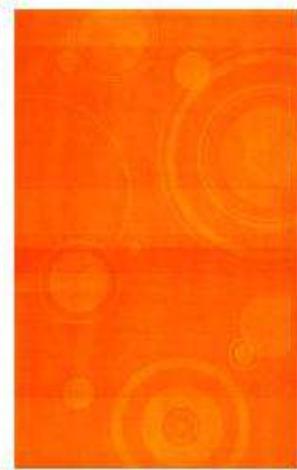
IEEE | MINIA UNIVERSITY STUDENT BRANCH

Contents:

- Introduction [20 min]:
1. What is Micro-Controller?
2. What is Arduino?
3. Types of Arduino.
4. Arduino UNO board.
5. Sensors:
  - Digital, Analog sensors.
  - Light sensors [IR sensor, Photo-Resistor].
- Coding structure and examples [30 min]:
1. Data types and operators.
2. What is "Function"?
3. Control statements [if, if… else, switch case.].
4. Loop statements[while, for, do… while.].
5. Common functions.
- Workshop[20 min] DC motor control:

1 Introduction

Design, organize, and collaborate

Micro-Controller:

It is a micro-computer. As any computer it has internal CPU, RAM, IOs interface.
It is used for control purposes, and for data analysis.

Famous microcontroller manufacturers are MicroChip, Atmel, Intel, Analog devices, and more.
[list]

What is Arduino?

A microcontroller board, contains on-board power supply, USB port to communicate with PC, and an Atmel microcontroller chip.

It simplify the process of creating any control system by providing the standard board that can be programmed and connected to the system without the need to any sophisticated PCB design and implementation.

It is an open source hardware, any one can get the details of its design and modify it or make his own one himself.
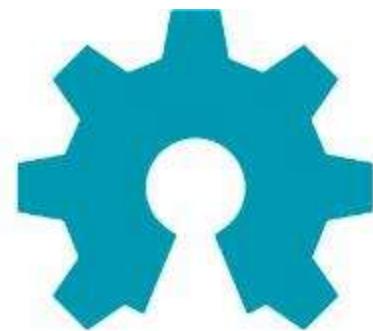

open source hardware

Arduino boards:

UNO

Mega
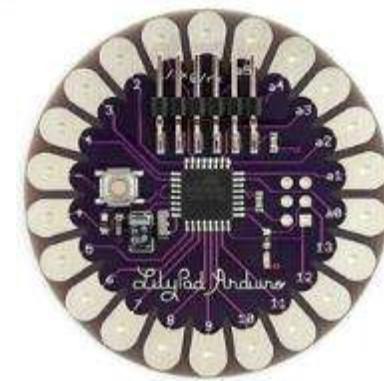
LilyPad

Arduino BT
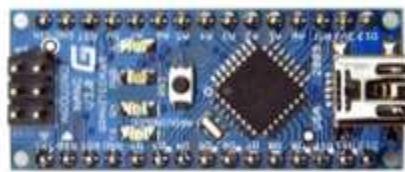
Arduino Nano

Arduino Mini

Arduino UNO:

Digital output
~: PWM.
0,1: Serial port.

In circuit Serial programming

USB port

Atmel MicroController

Power input

Power Supply

Analog input.
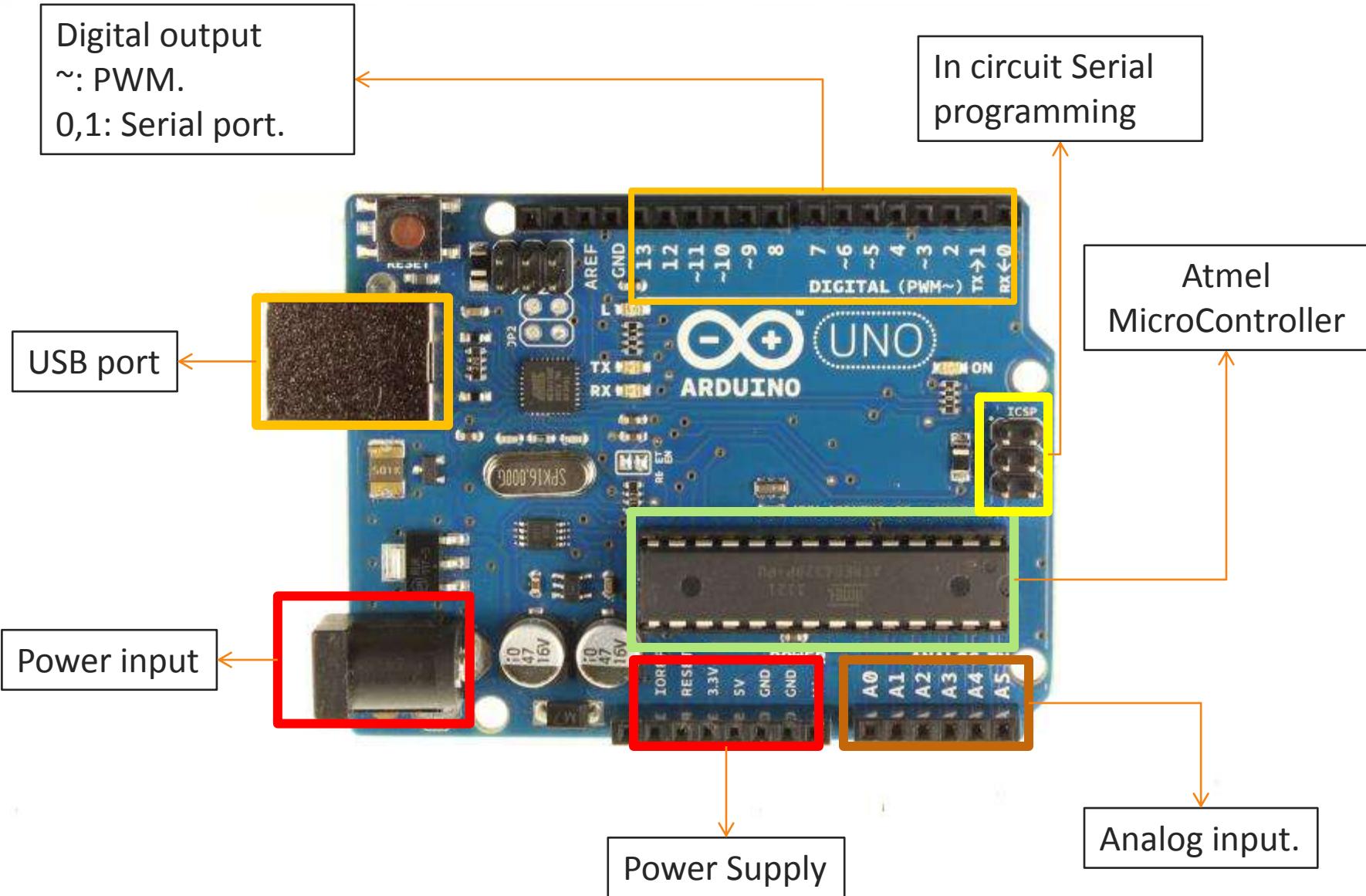
Digital or Analog?

All physical quantities are analog.

Analog means that the quantity can take any value between its minimum value and maximum value.

Digital means that the quantity can take specific levels of values with specific offset between each other.

Ex: 1- Digital:

English alpha consists of 26 letter, there is no letter between *A* and *B.*

- *Square waves are Digital.*

Ex.: 2- Analog:

Temperature, can take any value[-1,12.8,25.002,... etc.].

- Sine waves are analog.

Sensors:

A device that transforms the physical quantity into electrical value.

Ex.: Light sensor transduce the light into change in voltage or resistance.
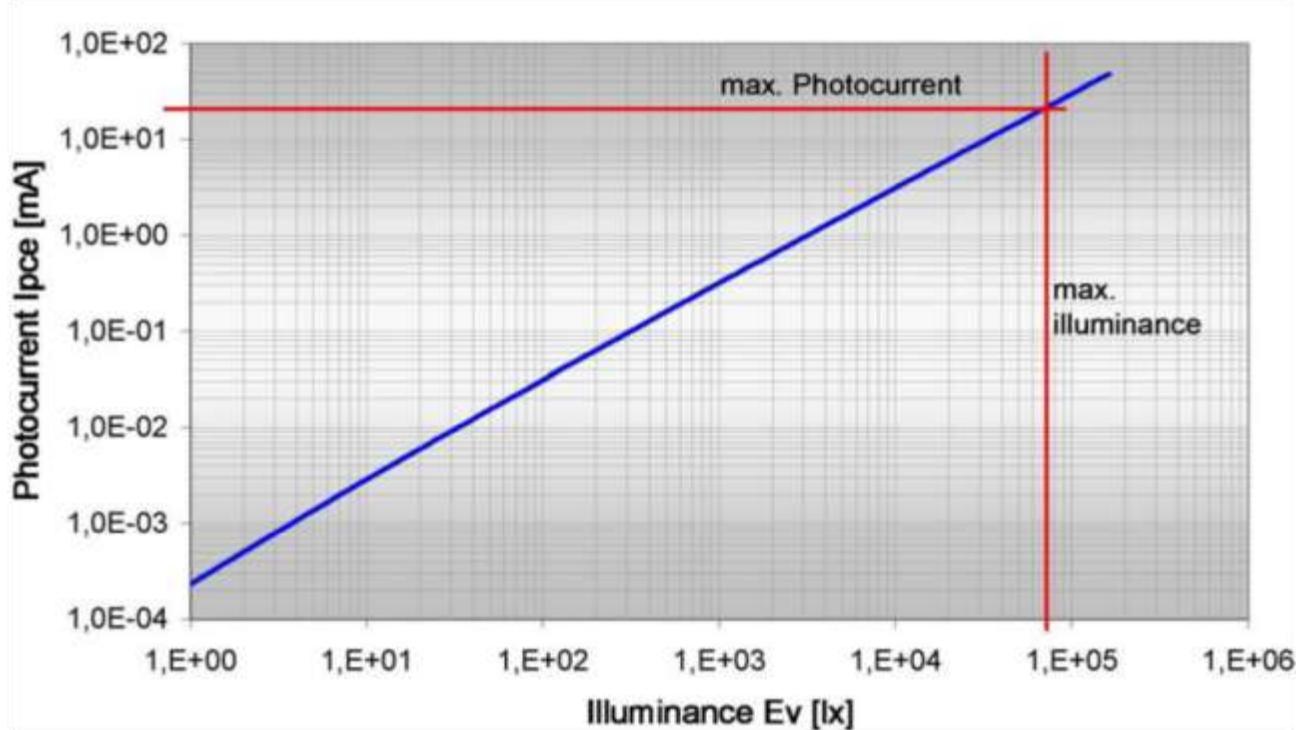
Light sensors:

- Photo-Resistor [photo-cell].
- Photo-Diode.
- Photo-Transistor.
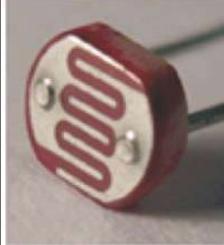
Device	Photo resistor	Photo diode	Photo transistor
Referenced part #	PDV-P500X	Everlight DTD-15	Everlight DPT-092
Accuracy	Not guaranteed	Not guaranteed	± 75%
Current (1000 lux)	Varies	3 μA	2.6 mA (70 klux)
Range	1 to 100 lux	7 to 50 klux	1 k to 100 klux
Response time	55 ms	6 ns	15 μs

Photo Resistor:

- The value of the resistance depends on the incident light density.
- 1 K-Ohm at light, 10 K-Ohm at darkness.

Photo Diode:

- The current is controlled by the incident light density.

Photo Transistor:

- Base-emitter junction is controlled by the incident light density, has an amplification effect.
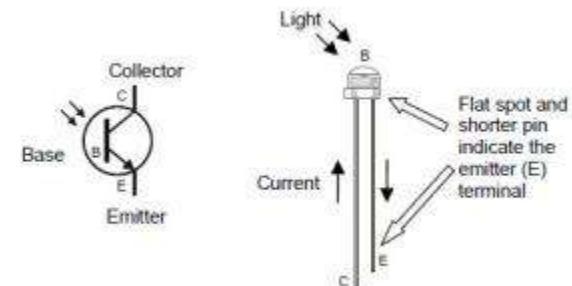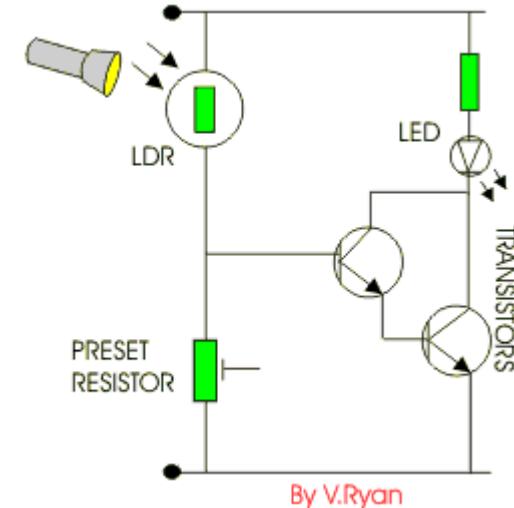
Arduino Coding.

Stylize, edit, and animate your media

Data Types and operators

Integer: used with integer variables with value between 2147483647 and -2147483647.
Ex: int x=1200;

Character: used with single character, represent value from -127 to 128.
Ex. char c='r';

Long: Long variables are extended size variables for number storage, and store 32 bits (4 bytes), from -2,147,483,648 to 2,147,483,647.
Ex. long u=199203;

Floating-point numbers can be as large as 3.4028235E+38 and as low as -3.4028235E+38. They are stored as 32 bits (4 bytes) of information.
Ex. float num=1.291;   [The same as **double** type]

Statement and operators:

Statement represents a command, it ends with **;**

Ex:

int x;

x=13;

Operators are symbols that used to indicate a specific function:

- Math operators: [**+,-,*,/,%,^**]

- Logic operators: [**==, !=, &&, ||**]

- Comparison operators: [**==, >, <, !=, <=, >=**]

Syntax:

;   Semicolon, **{}**   curly braces, **//**single line comment, **/***Multi-line comments***/**

Statement and operators:

Compound Operators:

++ (increment)

-- (decrement)

+= (compound addition)

-= (compound subtraction)

*= (compound multiplication)

/= (compound division)

Control statements:
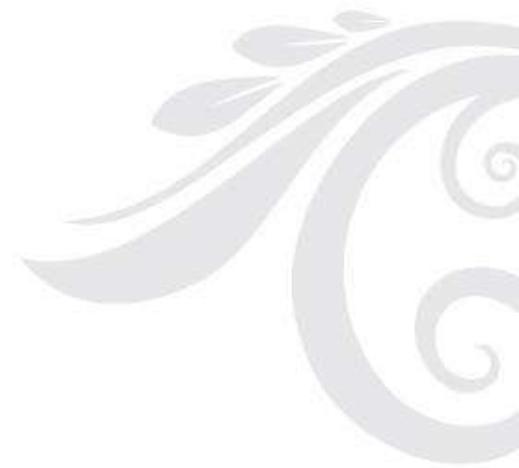
If Conditioning:
if(*condition*)
{
statements-1;
...
Statement-N;
}
else if(*condition2*)
{
Statements;
}
Else{*statements;*}

Control statements:

Switch case:

```
switch (var) {
  case 1:
    //do something when var equals 1
    break;
  case 2:
    //do something when var equals 2
    break;
  default:
    // if nothing else matches, do the default
    // default is optional
}
```

Loop statements:

Do... while:
```
do
{
Statements;
}
while(condition);        // the statements are run at least once.
```
While:
```
While(condition)
{statements;}
```
for
```
for (int i=0; i <= val; i++){
    statements;
  }
```

Use *break* statement to stop the loop whenever needed.

Code structure:

Void setup(){}
Used to indicate the initial values of system on starting.
Void loop(){}
Contains the statements that will run whenever the system is powered after setup.

Input and output:

Led blinking example:

Used functions:

pinMode();

digitalRead();

digitalWrite();

 delay*(time_ms);*

other functions:

analogRead();

analogWrite();*//PWM.*

Workshop:

Motor control using Arduino:

Make a motor rotate 2 sec clockwise, and 5 sec counter-clockwise in an infinite loop;

Time: 10 min.

Thanks for coming ☺