



**NYU**

**TANDON SCHOOL  
OF ENGINEERING**



Promoting robotic design and entrepreneurship  
experiences among students and teachers

# Lesson 13: Advanced Arduino Programming - II



- Arrays
- Functions
- **TASK/ACTIVITY:** Advanced programming with motors and functions

# ARRAYS

- An **array** is a collection of one type of variables that are accessed with an index number
- **Declaring an array:**
  - **Syntax:** `data_type array_name[size] = value;`
  - **Value** and **size** may/may not be given at declaration
- Examples:
  1. `int myNums[6];`
  2. `int myInts[ ] = {2, 4, 6};`
  3. `char message[6] = "hello";`

# ARRAYS

- Arrays in Arduino are **zero indexed**, i.e., array elements numbering starts from “0”
- Array elements can be accessed with index as follows:
  - Example: `int myNums[4] = {2, 4, 6};`  
`c = myNums[2];`
  - Value of c is 6
- We can assign a value to an array as follows:
  - `myNums[3] = 8;`

# ACTIVITY 1

1. Write a program to create an array and display all elements on the serial monitor
1. Create an array of size 6 with integers and display elements at index 2 and 4

# ARRAYS – SOLUTION FOR 1<sup>st</sup>

Print\_array\_values | Arduino 1.8.19

File Edit Sketch Tools Help



Print\_array\_values

```
int myIntVals[5]={1,2,3,4,5};

void setup() {
  Serial.begin(9600);
  for (int x = 0; x < 3; x++)
  {
    Serial.println(myIntVals[x]);
  }
}

void loop() {}
```

## OUTPUT:

The values of the array are: 1 2 3 4 5

# ARRAYS – SOLUTION FOR 2<sup>nd</sup>

```

Indexed_elements_in_Array | Arduino 1.8.19
File Edit Sketch Tools Help

Indexed_elements_in_Array

int myIntVals[6]={10,20,30,40,50,60};

void setup() {
  Serial.begin(9600);
  Serial.println("The value of the 2nd indexed element is: ");
  Serial.println(myIntVals[2]);
  Serial.println("The value of the 4th indexed element is: ");
  Serial.println(myIntVals[4]);
}

void loop() {}

```

# ARRAYS – SOLUTION FOR 2<sup>nd</sup>

**OUTPUT:**

The value of the 2<sup>nd</sup> indexed element is: 30

The value of the 4<sup>th</sup> indexed element is: 50



What is a **function**?

- A **function** is a block of code either **built-in** or **written by the user** that allows **structuring the code** into one or more **segments** to perform individual **tasks**

**Types of functions:**

1. **Predefined functions:** Built-in functions available in Arduino environment such as `println()`, `pinMode()`, `analogRead()`
2. **User-defined functions:** Functions written by the programmers to run a specific task

# STRUCTURE OF A FUNCTION

```

return_type function_name(parameters)
{ // declarations;
  // statements;
}
    
```

- **return\_type:** The data type of the value being returned by the function; If there is no return value, “void” can be used
- **function\_name:** The name of the function; It is user-defined; It should not start with a digit but can contain letter, number or an underscore; Pre-defined function names cannot be used
- **parameters:** The values being passed to the function; These are used inside the code
- **declarations:** Declaring the variables for the function
- **statements:** The code written for a specific task to be performed

## Example:

```

int my_add_func(int a, int b)
{ int sum; //declaration
  sum = a + b; //statements
  return sum;
}
    
```

- **return\_type:** int
- **function\_name:** my\_add\_func
- **parameters:** int a, int b
- **declarations:** int sum
- **statements:** sum = a + b; return sum;

# FUNCTION WITH NO PARAMETER

```

Funtion_NoParameter | Arduino 1.8.19
File Edit Sketch Tools Help

Funtion_NoParameter

boolean condition=0;

void setup() {
  Serial.begin(9600);
}

void loop() {
  delay(100);
  while(condition == 0){
    hello();
    Serial.print("Hello again from main loop.\n");
    delay(100);
    condition++;
  }
}

```

```

void hello() {
  Serial.print("Hello from function!\n");
  delay(2000);
}

```

## OUTPUT:

Hello from function!  
Hello again from main loop.

# FUNCTION WITH ONE PARAMETER

Funtion\_OneParameter | Arduino 1.8.19

File Edit Sketch Tools Help



Funtion\_OneParameter §

```
boolean condition=0;

void setup() {
  Serial.begin(9600);
}

void loop() {
  delay(100);
  while(condition == 0){
    hello(4);
    Serial.print("Hello again from main loop.\n");
    delay(100);
    condition++;
  }
}
```

```
void hello(int repeat){
  for(int i=1;; i<=repeat; i++){
    Serial.print("Hello from function!\n");
    delay(500);
  }
}
```

## OUTPUT:

```
Hello from function!
Hello from function!
Hello from function!
Hello from function!
Hello again from main loop.
```

## ACTIVITY 2

- Write a program to display all even numbers from 1 to  $n$  where  $n$  is the parameter to be sent, i.e.,  $n = 100$

# ACTIVITY 2 - SOLUTION

```

Print_EvenNumbers | Arduino 1.8.19
File Edit Sketch Tools Help

Print_EvenNumbers

void setup() {
  Serial.begin(9600);
  even_nums(100);
}

void loop() {}

void even_nums(int n){
  Serial.println("Even number from 1 to 100 are");
  for(int a=1;; a<=repeat; a++){
    if (a%2 == )
      Serial.println(a);
    delay(500);
  }
}

```

## OUTPUT:

Even numbers from 1 to 100 are:  
 2  
 4  
 6  
 ...  
 100

# FUNCTION WITH TWO PARAMETERS

Funtion\_TwoParameters | Arduino 1.8.19

File Edit Sketch Tools Help



Funtion\_TwoParameters \$

```
boolean condition=0;

void setup() {
  Serial.begin(9600);
}

void loop() {
  delay(100);
  while(condition == 0){
    hello(5,2000);
    Serial.print("Hello again from main loop.\n");
    delay(100);
    condition++;
  }
}
```

```
void hello(int repeat, int dlyTim){
  for(int i=1;; i<=repeat; i++){
    Serial.print("Hello from function!\n");
    delay(dlyTim);
  }
}
```

## OUTPUT:

```
Hello from function!
Hello from function!
Hello from function!
Hello from function!
Hello again from main loop.
```



# FUNCTION WITH RETURN PARAMETER

```

Funtion_OneReturnParameter | Arduino 1.8.19
File Edit Sketch Tools Help

boolean condition=0;

void setup() {
  Serial.begin(9600);
}

void loop() {
  delay(100);
  while(condition == 0){
    int c = adder(20,30);
    delay(100);
    Serial.print(c, DEC);
    condition++;
  }
}

```

```

int adder(int aa, int bb){
  int cc = aa + bb;
  return cc;
}

```

**OUTPUT:**  
50



# FUNCTION WITH NO RETURN PARAMETERS

∞ Funtion\_NoReturnParameter | Arduino 1.8.19

File Edit Sketch Tools Help



Funtion\_NoReturnParameter \$

```
// Global variables
int x; int y;
void setup() {
  Serial.begin(9600);
}
```

```
void loop() {
// pick some random numbers
x = random(10);
y = random(10);
Serial.print("Values of x and y before swapping: ");
Serial.print(x);
Serial.print(',');
Serial.println(y);
swap();
Serial.print("\nValues of x and y after swapping: ");
Serial.print(x);
Serial.print(',');
Serial.println(y);
delay(1000);
}
```



```
void swap() {  
    int temp;  
    temp = x;  
    x = y;  
    y = temp;  
}
```

## OUTPUT:

Values of x and y before swap 7,9

Values of x and y after swap 9,7



**NYU**

**TANDON SCHOOL  
OF ENGINEERING**



# Task / Activity: Advanced Programming

**Innovative Technology Experiences for Students and Teachers (ITEST), Professional Development Program, July 2017**  
Mechatronics and Robotics Laboratory, Department of Mechanical and Aerospace Engineering, NYU Tandon School of Engineering

## Motor Direction Control using Functions

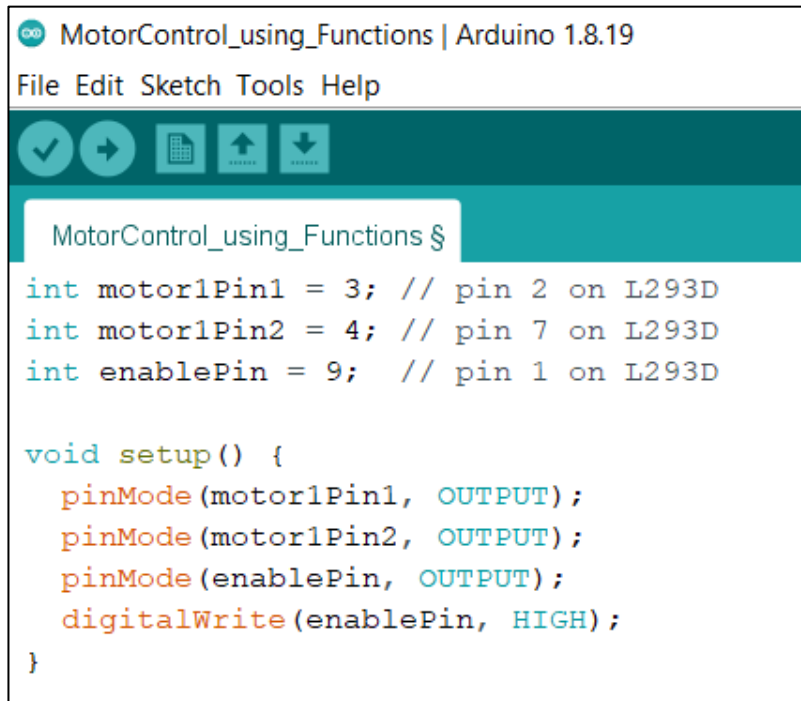
Write a program to make the robot's wheels rotate clockwise and rotate anticlockwise using functions

1. Which rotation makes the robot move forward?
1. Which rotation makes the robot move backward?

# ACTIVITY 3 SOLUTION

- Connect a DC motor with L293D IC (like you have previously done)

## Setup function



```
MotorControl_using_Functions | Arduino 1.8.19
File Edit Sketch Tools Help

MotorControl_using_Functions $
int motor1Pin1 = 3; // pin 2 on L293D
int motor1Pin2 = 4; // pin 7 on L293D
int enablePin = 9; // pin 1 on L293D

void setup() {
  pinMode(motor1Pin1, OUTPUT);
  pinMode(motor1Pin2, OUTPUT);
  pinMode(enablePin, OUTPUT);
  digitalWrite(enablePin, HIGH);
}
```

# ACTIVITY 3 SOLUTION

## Loop function

```
void loop() {  
  clockwise_rotate();  
  delay(2000);  
  anticlockwise_rotate();  
  delay(2000);  
}
```

## Function definition

```
void clockwise_rotate(){
    digitalWrite(motor1Pin1, LOW); // set pin 2 on L293D LOW
    digitalWrite(motor1Pin2, HIGH); // set pin 7 on L293D HIGH
}

void anticlockwise_rotate(){
    digitalWrite(motor1Pin2, LOW); // set pin 7 on L293D LOW
    digitalWrite(motor1Pin1, HIGH); // set pin 2 on L293D HIGH
}
```



# ACTIVITY 3 SOLUTION

[Video: Forward movement](#)



# ACTIVITY 3 SOLUTION

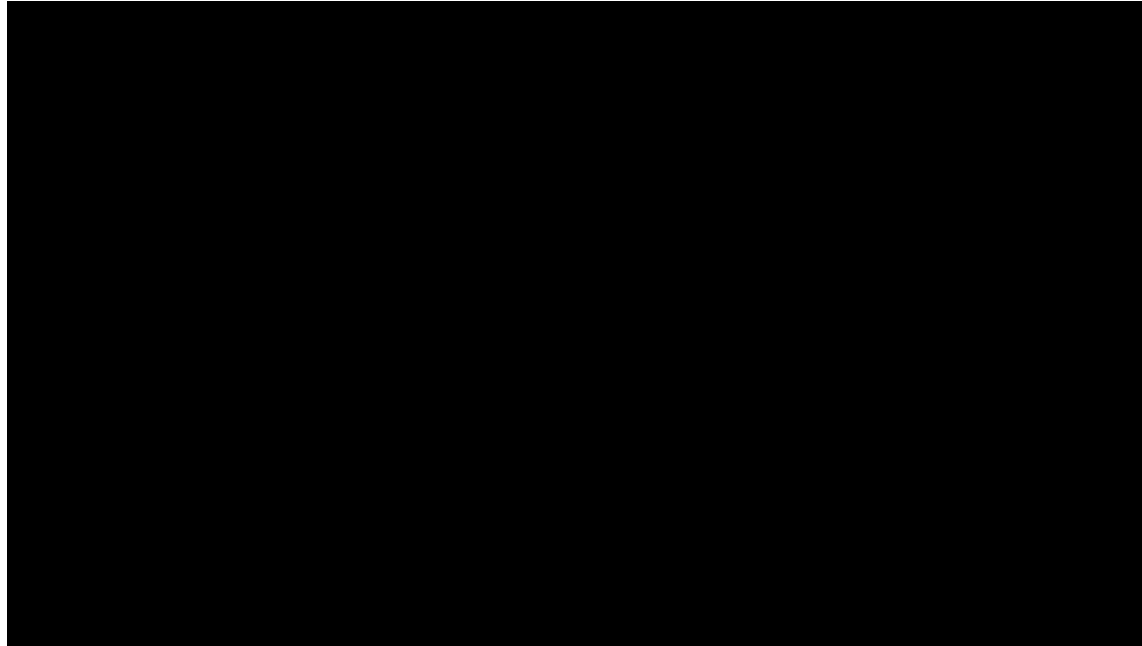
[Video: Backward movement](#)



# VEX CLAWBOT



# VEX CLAWBOT



[Video](#)



**NYU**

**TANDON SCHOOL  
OF ENGINEERING**



# Thank You!

## Questions and Feedback?