**README**

**ADD-ON Packages:**

1. MATLAB Support Package for Arduino Hardware
2. Simulink Support Package for Arduino Hardware
3. Arduino\_Engineering\_Kit\_Hardware\_Support (according to MATLAB 2019a version)
4. Simulink library for Arduino Liquid Crystal Display

**Arduino Pins attached with each component:**

1. LCD:

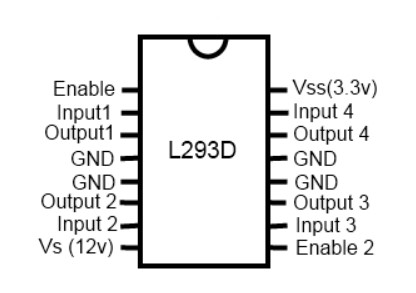
* VSS- GND
* VDD- 5V
* Potentiometer input- Pin
* RS- Pin 10
* RW- GND
* EN- Pin 9
* D0 to D3- No Connection
* D4- Pin 5
* D5- Pin 4
* D6-Pin 3
* D7- Pin 2
* A- 5V
* K- GND

1. Servo Motor:

* Orange Wire- 5V
* Brown Wire – GND
* Yellow Wire – Pin 8

1. L293D Motor Driver Chip:

* Input 1- Pin 6
* Input 2- Pin 7
* Output 1- DC Motor Pin 1
* Output 2- DC Motor Pin 2



**Fig. L293D Pin Diagram**

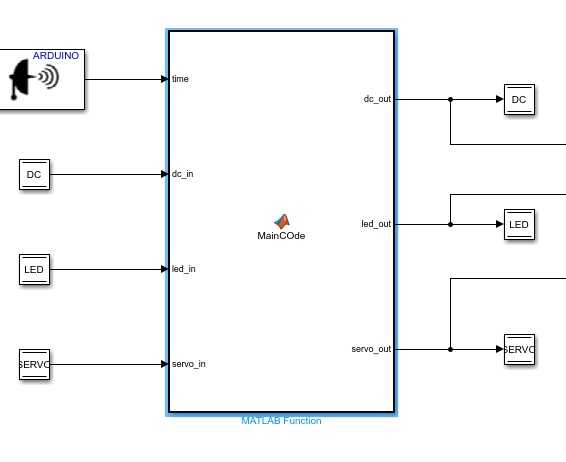
1. LED:

* Positive Pin – Resister (220 ohm)- Pin 13
* Negative Pin- GND

1. Ultrasonic Sensor:

* VSS- 5V
* Trig- Pin 11
* Echo- Pin 12
* GND- GND

**MATLAB Functions:**



**Fig. MATLAB Function (DC and Servo Motors)**

**Below is the code execution explanation step by step:**

**function [dc\_out,led\_out,servo\_out] = MainCOde(time,dc\_in,led\_in, servo\_in)**

**distance = (time\*343)/2;**

* MATLAB function is used where the output is going to tell the details about the object with the help of an ultrasonic sensor used to calculate the distance for the further process.

**if distance > 40**

**servo\_out = 90;**

**dc\_out=1;**

**led\_out = 0;**

* Above, code is setting initial arm position to 90 degrees, if no obstacle is there which is specified by randomly taking the distance more than 40, that is, the range of the obstacle, the robot will move forward. We are setting dc\_out = 1; which means it will store the value in Data Store Write and this will help in turning on DC motor. Initially DC motor is on, but LED is off, so we are setting led\_out = 0 and storing it to data store write.

**elseif distance > 0 && distance < 20**

**servo\_out = 180;**

**dc\_out = 0;**

**led\_out=1;**

**elseif distance > 20 && distance < 40**

**servo\_out = 0;**

**dc\_out = 0;**

**led\_out=1;**

* For both elseif loops above, dc motor is off when the obstacle is in the range between 0 to 40 and LED is on so we are passing 1 to Digital Output, so it will light up the LED. And since the obstacle is in the path, the servo motor arm will move 180 degrees ‘LEFT’ and 0 degree ‘RIGHT’, if the range of the obstacle is between 0 to 20 or distance, is between 20 and 40 respectively.

**else**

**servo\_out = servo\_in;**

**dc\_out= dc\_in;**

**led\_out=led\_in;**

**end**

**end**

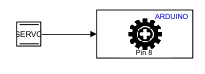
* If None of the condition is true then it will just set input is equal to output so nothing will happen, this condition is just to complete the ‘if’ loop as all the time one of the above will be true.

**LED Data Store Read:** This will take LED data read store as input and when ‘1’ is set form the code used in MATLAB function, it will light up the LED and when ‘0’ it will turn off the light.



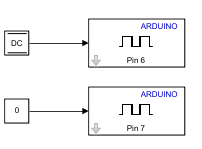
**Fig. LED Data Store Read**

**Servo Motor Data Store Read:** We have used standard servo read, to change the arm position, it will change according to the distance to the obstacle from the robot, that is, will turn left when in the between ‘0 to 20’ and right when in the range between ‘20 to 40’.



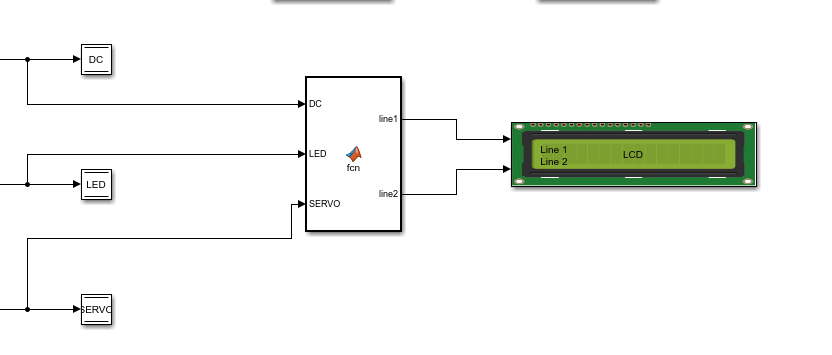
**Fig. Servo Motor Data Store Read**

**DC Motor Data Store Read:** This will take DC data to read the store as input and when ‘1’ is set form the code used in MATLAB function then it will is turn on the DC motor making the robot move in the forward direction. And when DC is ‘0’ then it will turn off the motor.



**Fig. DC Motor Data Store Read**

**MATLAB FUNCTION (LCD DISPLAY):** The top line of the LCD will display “DC Motor RPM:” and the bottom line of the LCD will display the numerical RPM of the motor. Once the object is detected the top line of the LCD will display “ALERT”, and the bottom line of the LCD will indicate which direction the robot is moving.

****

**Fig. LCD Display MATLAB Function**

**Below is the complete code in MATLAB Function for the display:**

**function [line1, line2] = fcn(DC,LED,SERVO)**

1. **Below is the code explanation for the first line of LCD**

**if DC == 1**

**line1= [[68],[67],[32],[77],[111],[116],[111],[114],[32],[82],[80],[77],[58]];**

* DC data write store is used as input, if this condition is true then, LCD line 1 will display: “DC Motor RPM:”

**elseif LED == 1**

**line1 = [[65],[76],[69],[82],[84],[33],[32],[32],[32],[32],[32],[32],[32]];**

* If DC is false then it will check for the above condition if LED = 1, same we have used LED data write store as input, if this is true then LCD line 1 will display: “ALERT!”

**else**

**line1 = [[32],[32],[32],[32],[32],[32],[32],[32],[32],[32],[32],[32],[32]];**

**end**

* If none is true, then it will display Space which means blank.

1. **Below is the Code explanation for the second line of LCD**

**if SERVO == 180**

**line2= [[84],[85],[82],[78],[32],[76],[69],[70],[84],[32],[32]];**

* If Servo data write store will have 180 stored in it then line 2 will display: “TURN LEFT”

**elseif SERVO == 0**

**line2= [[84],[85],[82],[78],[32],[82],[73],[71],[72],[84],[32]];**

* If Servo data write store will have 0 stored in it then line 2 will display: “TURN RIGHT”

**elseif DC == 1**

**line2 = [[49],[56],[50],[57],[32],[32],[32],[32],[32],[32],[32]];**

* DC data write store is used as input, if this condition is true then, LCD line 2 will display: “1829”

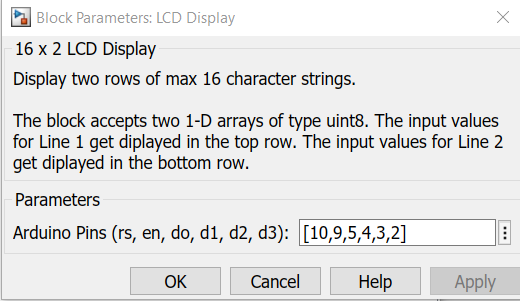
**else**

**line2= [[32],[32],[32],[32],[32],[32],[32],[32],[32],[32],[32]];**

**end**

**end**

* If none is true, then it will display space.

****

**Fig.12. LCD Display Inputs**