Programming a web page to control the arm using the web serial API?

THE ARCHITECTURE AND DESIGN

The proposed block diagram of this robotic system is shown in Figure ٢ which consists of an Arduino EPS٣٢ board microcontroller and four digital servo motors driven by driver integrated circuit (IC), in addition to a WIFI router and volts-direct current (٥ V-DC) power supply. The pick and place robotic arm has a robotic arm put on a fixed base. The pick and place robot utilize ٤-motors for the operation of the structure. also, it contains an arm assembly with a selectable jaw or with a screwdriver, which can move upend down direction only. For, the motor controlling is using motor driver IC and Arduino EPS٣٢microcontroller. The processing signal is provided by a personal computer (PC) or mobile phone via html designed web page to the wireless router. The signal is wireless router. The signal is decoded in the controller when it is sent from the wireless router and suitable controlling signal is sent to actuators (servo engine) in the system.

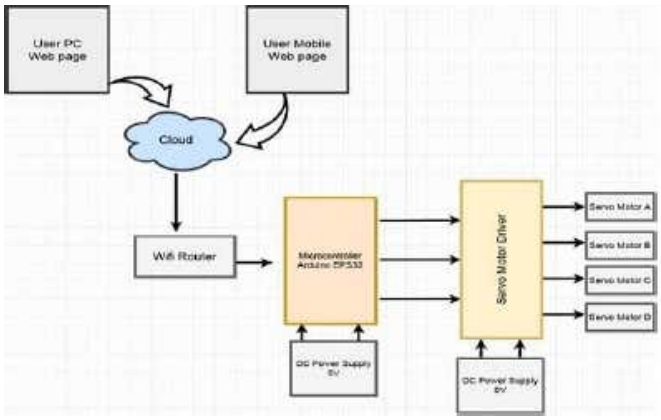

Figure ٢. Block diagram of the proposed project

FLOWCHART OF THE IMPLEMENTED CODE

The Figure ٣ describe the methodology for programming of the system. First, we are arising a server instance that obey for hypertext transfer protocol (HTTP) commands on port ٨٠ This port is represented the default port for servers of web. In the platform of Arduino, the signal is transmitted from the router to be readon Arduino utilized the command (if client available). The receiver flag of the Arduino controller is monitored to discover every order is supplied to the controller next mission is decoded the instruct the flag of receiver is high.

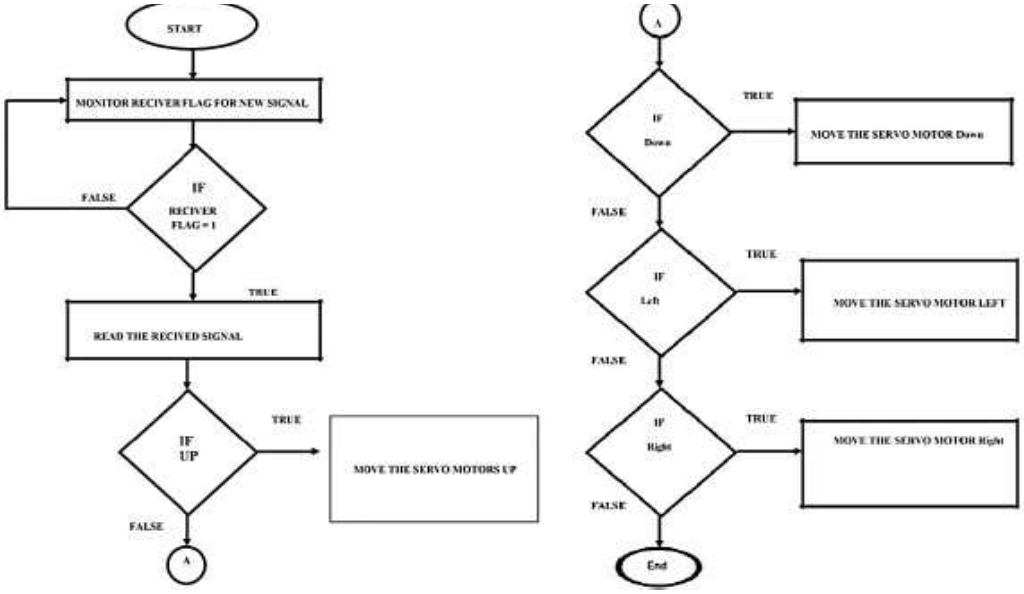

Figure ٣. Flowchart of the Arduino source code

SIMULATION AND RESULTS DISCUSSION

The robotic arm is tested in Proteus software before the construction as shown in is constructed using ٤-servo motors with Arduino. After the Figure ٤. The arm completing the circuit diagram, the simulation shows a good operation by the servo motors. Then arm is ready to be installed for running and performs the duty which is for pick and place or to be used as screw driver for tightening the screws in the appropriate positions. Now we will be talking about the code of the Arduino which let us to control the servo movements by just sending its position in degree value via the web page designed in html language. The servos control lines are:

```
const int Servo A=١٣;
const int Servo B=١٤;
const int Servo C=١٥;
const int Servo D=١٦;
```

Here we are used to specify the four servos to the pins (١٣, ١٤, ١٥, ١٦) respectively. You can use other pins as its required. Practically we used ESP٣٢ Arduino for wirelessly, connecting the robotic arm and control on it and movement the dashboard in the web page as shown in the Figure ٥. The command which is the: led Setup (٠, ٩٠, ٨) in Arduino, in the setup function, is going to setup the PWM with the ledcSetupfunction, which receives as input the PWM channel, the final servo the increment in each command time of position and the duty cycle resolution, i.e., the slider movement.

The function ledcAttachPin (Servo A, ٠) is to attach the servo (A) to the first specified pin (channel)which was the pin (١٣). This command is repeated to the other servos is being as:

ledcSetup (١, ٩٠, ٨); for servos B
ledcSetup (٢, ٩٠, ٨); for servos C

ledcSetup (٣, ٩٠, ٨); for servos D

The number (٨) indicate increment in the servo movement in each step the slider of the dashboards ends. The command serial begins (١١٥٢٠٠) doesn't actually print anything but It is utilized to set up the communication speed in bits/ second instead of it initializes the connection of serial at ١١٥٢٠٠ bits per second. To obtain any type of clear data, it is using the same order of serial connection speed by adjust both sides of the serial connection (i.e., the computer and the Arduino). The data are distorting if there doesn't have symmetric in the speed for both the two systems. When the number is making smaller (e.g., serial. begin (٣٠٠)) the Arduino will be transmitting the data in slow. While rising it, say to ٥٧٦٠٠ will transmit the data fast. Both the transmitting and the receiving systems need to agree on the speed must to use: the serial program for your computer's, as the window of Arduino serial monitor will let you adjust the speed at that the data is received by your computer but the speed is selected from the common speeds only: ٣٠٠, ١٢٠٠, ٢٤٠٠,٤٨٠٠, ٩٦٠٠, ١٩٢٠٠, ٣٨٤٠٠, ٥٧٦٠٠ and ١١٥٢٠ bit per sec. Now we have to enter the web page environment which begins with the log in gate. This can be done using the command (if (header. indexOf ("") >= ٠) {), this will match the user name and password you have specified to the web page authentication which is shown in Figure ٦. If the user's name and password is wrong the web page in the Figure ٧ will pop up.

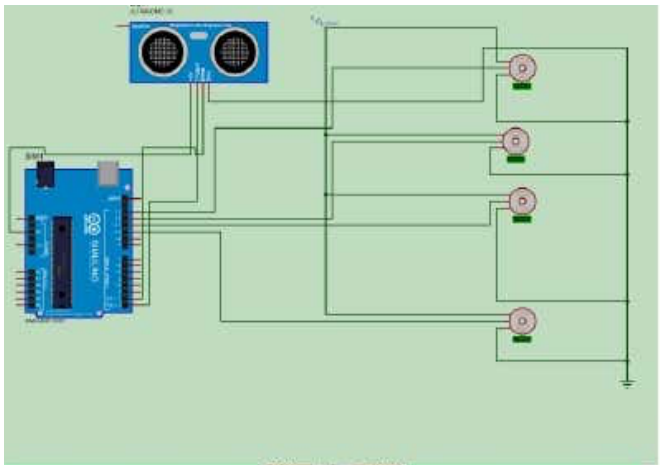

Figure ٤. proteus simulation for robotic arm

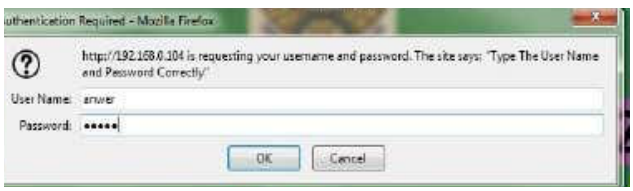

Figure ٦. Web page authentication

Figure ٥. web page dashboard

Figure ٧. Web page for wrong log in

After the entering the web page the control section will be in the slider bar, and each servo motor has its own slider bar to be controlled separately by the user. After the entering the web page the control section will be in the slider bar, and each servo motor has its own slider bar to be controlled separately by the user as shown below for servo A and is repeated for the other three ones:

client. println(" Servo A: <span id=\"servoPosA\"> ");
client. println ("<input type=\"range\" min=\"٠\" max=\"٥٠\" class=\"slider\" id=\"servoSliderA\"onchange=\"servo (this.value,'A
')\" value=\"" + valueString + "\"/>");
client. println("<script>"); //send A valueclient. println ("var sliderA = document. getElementById(\"servoSliderA\");");
client. println ("var servoPA = document. getElementById(\"servoPosA\");
servoPA.innerHTML = sliderA.value;");

Our servo motors are installed by the screws on board shown in the Figure ٨. They are tightened depending on the angular rotation of servo motors
.
The rotation degree is to be from ٠ to ١٨٠ degrees for the base servo which is to rotate the arm. This base handle the remaining three servos to make the bending and the pick and pace of things. The
println()
method writes data to the serial port. This is often helpful for looking at the data a program is producing or to write data to other devices connected to the serial port. The
println
method works like print, but sends a new line character for each call to the function. Data can either be a single
int, float, byte, long, char, char[]

, String or a number in decimal (DEC), hexadecimal (HEX),octal (OCT), or binary (BIN) base [٢٤]. The authentication for the log in page can be done using the line command: client.
println
("HTTP/١,١ ٤٠١ Unauthorized"). The ٤٠١ Unauthorized error is an HTTP status code that means the page you were trying to access cannot be loaded until you first log in with a valid user identification (ID) and password
.
If you've just logged in and received the ٤٠١ Unauthorized error, it means that the credentials you entered were invalid for some reason
.
٤٠١ Unauthorized error messages are often customized by each website, especially very large ones, so keep in mind that this error may present itself in more. The ٤٠١ Unauthorized error displays inside the web browser window, just as web pages do. Like most errors like these, you can find them in all browsers that run on any operating system [٢٥].

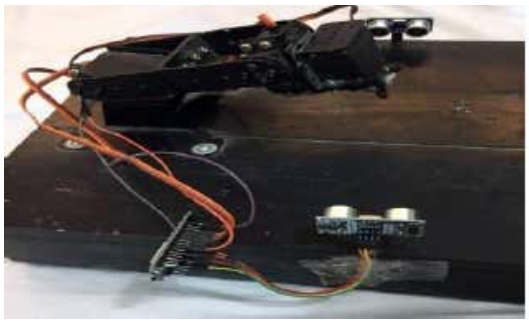
Figure ٨. robotic arm servos construction

Ultrasonic sensor

Pulse-width modulation (PWM) applications can be found in a number of applications such as communication, servo motor control, voltage regulation, and power conductivity [٢٦], [٢٧]. This paper gives in details the way to measure the PWM display by get the benefit of Arduino. Moreover, how the ultrasonic sensor (which can be used for telemetry) that can make its job with the help of the PWM is an ultrasound sensor that can remotely measure the physical quantities by using the ultrasound waves [٢٨], [٢٩]. The human ear has not the ability to detect ultrasound. This unit has the ability to correctly measure the distance from ٢ to ٤٠٠ cm (١ to ١٥٦ inches) and easily be connected together with the Arduino microcontroller. HC-SR٠٤ consumes less energy that makes it a good choice to work in automatic and automatic control systems [٣٠]. The principle of operation of this ultrasonic sensor (HC-SR٠٤) is similar to the of object detection system of a bat which is shown in Figure ٩. Ultrasound sensor sends a ٤٠ kHz wireless frequency to the air at ٣٤٦ meters per second and the receiver get the reflected signal arrived from the reflector body. Then this distance between the received (Rx) and the transmit (Tx) can be simply calculated by the looking at the time taken by the ultrasound to send from the transmitter and reflect back to the receiver which is reflected by the receiver. If we be in the interested calculation to this separating distance between

the object and the transmitter, the sensor module is adjusted to do its job. The result of these calculations is depending on the pulse which has a width that is related to the separating measured distance.

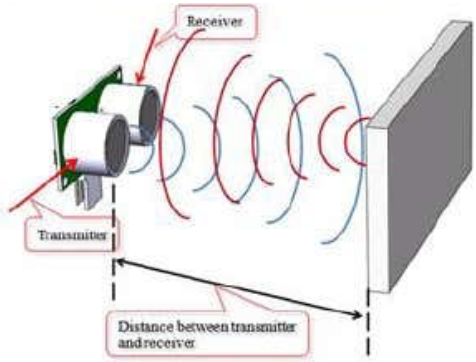

Figure ٩. HC-SR٠٤ module operation

The HC-SR٠٤ unit diagram has four pins using them in its operation as shown in Figure ١٠. At the first, it is very useful to study the functionality of the pins of this module sensor.

- Vcc can be supplied up to ٥ V.
-Trig pin input is for getting a pulse with a time width of ١٠ micro seconds to generate the ٤٠ KHz ultrasonic signals to be transmitted into the air from the Tx.
-Pin ٣ is the pin for receiving the Echo pulses arrived to the module from the objects to the calculate the width of the pulse and then the calculate the distance.
-GND is the ground pin.

The characteristics curve of this module is that it generates a series of pulses signals with at least ١٠micro second then it will be fed to the Trig pin.
After that by about ١,٤ milliseconds, it will receive a pulses signal which arrive to the echo pin which has signal width ranges start from ١٥٠ seconds to ٢٥milliseconds. If the width of the signal is greater than the range above, the sensor module will not be able to detect the object. Therefore, it must wait for about ١٠ milliseconds and retransmit another Trig signal again as shown in Figure ١١.

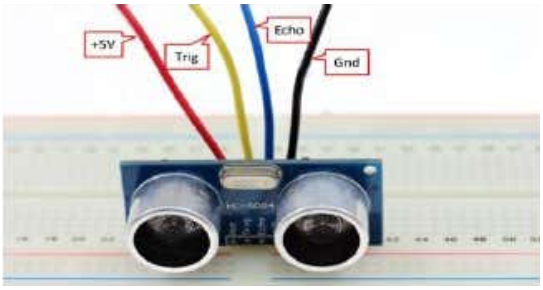

Figure ١٠. HC-SR٠٤ module pin out

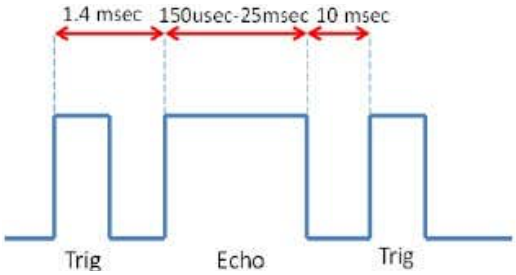

Figure ١١. HC-SR٠٤ Trig and Echo signal

In the Arduino program, we can calculate the object distance which stand in the front of the sensor. At first, the sensor will send a signal at an instance and staying waiting for the answer (reflecting) signal from the standing object or body, i.e. the echo again at another particular instance. The Pinging or transmit and receive is about a voice wave that a human can't hear for this sensor is called "ultrasound". In mathematical details, the sensor programmed to send a signal at time t١
and receives a reflected one at time t١.
For that, if we know the speed of the sound wave, then the time difference
$\Delta = t_2 - t_1$
will give the idea of the distance between the sensor and the object.

For example,
if the time difference was Δt= ٥٠٠ microseconds, then it will be known that soundwave takes ٢٥٠ microseconds to receive and hit the object and another ٢٥٠ microseconds to be returned. Therefore, for accurate operation, the robotic arm equipped with ultra-sonic sensor (HC-SR٠٤) to be used in the accurate approximation between this arm and the aimed things that to be picked and placed, or to be tightened by the screw driver installed at the top which is rotate automatically when the splitting space is zero according to the ultra-sonic sensor reading. Ultra-sonic indicate to a voice wave with a frequency is over ٢٠kH, which is greater than the upper human. hearing level. Since the (HC-SR٠٤) is depends on the ultra-sonic waves. Then the movement of this wave will be affected by the temperature of the environment it works. Theoretical calculation of the distance the ultrasonic wave travel is by the (١):

Distance = v×t

where v, is the wave velocity (about ٣٤٠ m/s) in the air at (٢٠C). Since the ultra-sonic wave is affected by the temperature of the ambient of working, there must be a modification on the (١) to get the
accurate position of the object the robotic arm deals with. Depending on the system specification below, we willdevelop a modification on the equation (١) to get the excepted results:
Frequency of the oscillator= ٢٠ Mhz, Cycle= ٤/٢٠ Mhz=٠,٢ µS and timer count = ٠,٢*٨=١,٦ µS At٢٠ °C sound speed= ٣٤٠٠٠ cm/sec so within ١,٦ µS the be
(١,٦*٠,٠٠٠٠١*٣٤٠٠٠=٠,٠٥٤٤cm per count), but sound distance is twice (becau se the come and go back), so relationship becomes٠,٠٥٤٤/٢=٠,٠٢٧٢ cm per

count. The sound speed within ١,٦ µS the distance will be(١,٦*٠,٠٠٠٠١*٣٤٠٠٠=٠,٠٥٤٤ cm per count), but sound distance is twice (because the come and go back), so relationship becomes ٠,٠٥٤٤/٢=٠,٠٢٧٢ cm per count. To start the measure, the device needs a pulse of ١٠ µS on trigger input then send (itself) a burst of ٨ periods of ٤٠ Khz (so during ٢٠٠ µS) then echo output signal goes to ١ status and returns to ٠ status when echoes is back timer measures this duration. To avoid to ear the receiver when emitter send the burst salve of٤٠ khz, timer start must begin to count after (١٠+٢١٠=٢١٠ µS). So minimum distance (theatrical) is٢١٠*٠,٠٥٤٤=٥,٧١٢ cm (٠,٠٥٤٤ because only one way), and add a little offset for calibration here offset is١,٠٩٣ cm. Then the final equation will be. Figure ١٢ shows the simulation of the effect of the temperature on the ultra-sonic speed which results in drift in the actual distance (٢).
distance=time*٠,٠٢٨+١,٠٩٣

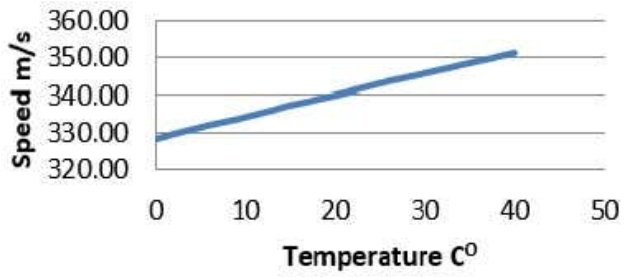

Figure ١٢. Effect of the temperature on the ultrasonic speed

CONCLUSION

The robotic arm is designed and simulated using Proteus software by take the advantages of the Arduino model ESP٣٢ which can be connected to the wireless router. The command is sent from the designed web page to this arm so as to pick or place objects or for tightening the screws of the devices or objects. We can use the speed of sound also (pace of sound) for this purpose. The pace of sound can be equal to (١/Speed of Sound=١/٠,٠٣٤٣٥=٢٩,١ s/cm). Therefore, the equation which is used to calculate the distance will become: D=
(Δt/٢) / of sound wave pace.
This robotic arm here takes the benefits of the ultra-sonic sensor model (HC-SR٠٤) for the best accuracy of working for good proximity between the arm and the objects and to specify the operation area of the arm. Since the sensor depends on the ultra-sonic waves, there will be an effect on its speed by the ambient working temperature, which made the need to the modification of the model distance equation. Finally, the goal of the IOT is can be done using the real internet protocol (IP) to connect to the external world of networks (INTERNET).

Arduino code for robotic arm control?

```
#include <Servo.h>

Servo myservo;   // create ٤ servo objects to control the
servos
Servo myservo١;
Servo myservo٢;
Servo myservo٣;

int angle = ٩٠; // initial angle  for servo
int angle١ = ٩٠;
int angle٢ = ٩٠;
int angle٣ = ٩٠;
int angleStep = ٥;

#define LEFT ١٠   // pin ١٠ is connected to left button
#define RIGHT  ١١  // pin ١١ is connected to right button
#define LEFT١ ٢ // pin ٢ is connected to right button
#define RIGHT١ ٣ // pin ٣ is connected to right button
#define LEFT٢ ٦ // pin ٦ is connected to right button
#define RIGHT٢ ٧ // pin ٧ is connected to right button
#define RIGHT٣ ٤ // pin ٤ is connected to right button
#define LEFT٣ ٥ // pin ٥ is connected to right button

void setup() {
 Serial.begin(٩٦٠٠);  //  setup serial
 myservo.attach(٨);  // attaches the servo on pin ٩ to the
servo object
 myservo١.attach(٩);
 myservo٢.attach(١٢);
 myservo٣.attach(١٣);
 pinMode(LEFT, INPUT_PULLUP); // assign pins as input left
button
pinMode(RIGHT, INPUT_PULLUP); // assing pins as input for right
button
pinMode(LEFT١, INPUT_PULLUP);
pinMode(RIGHT١, INPUT_PULLUP);
pinMode(LEFT٢, INPUT_PULLUP);
pinMode(RIGHT٢, INPUT_PULLUP);
pinMode(LEFT٣, INPUT_PULLUP);
pinMode(RIGHT٣, INPUT_PULLUP);

myservo.write(angle); // send servo to the middle at ٩٠ degrees
myservo١.write(angle١); // send servo to the middle at ٩٠
degrees
```

```
myservo١.write(angle٢); // send servo to the middle at ٩٠
degrees
myservo١.write(angle٣); // send servo to the middle at ٩٠
degrees

}

void loop() {
////////////////////////////////////////////////////////////
/////////

//visit the website for the complete code:
http://www.makosite.one/robotic-arm.php

////////////////////////////////////////////////////////////
/////////
```