Engineering Journal

Team S.L.S. 7

Coach

Ahmed Tarek

Team Members

Ahmed Zaki

Omar Momtaz

Mohamed Ashraf

Table of Contents

Introduction

- We are students in junior “6” in Sidi-Gaber Language School.
- Our team is called S.L.S. 7
- Our coach and supervisor is Mr. Ahmed Tarek
- The members of team and specific missions:
    - The mechanical is Omar Momtaz
    - The electrical is Mohammed Ashraf
    - The programmer is Ahmad Zaki

Activity Log

Date	Tasks	Status	Difficulty	Comments/Progress
2/2/2017	• Explaining the competition rules. • Dividing the missions between team members. • Putting the plan. • Deciding the components of robot S.L.S. 7	Completed	• Some parts of the robot components were not available. • Searching about them in more than one store and also through websites.	• Enthusiasm with high spirit. • Buying some of the components of robot S.L.S. 7
5/2/2017	Check the components of robot S.L.S. 7	Completed	None.	Buying the rest of robot components.
9/2/2017	Body Assembly 1st trial	Completed	The places of the nails were wrong so we couldn't assemble the components	Failed
	Connect the Arduino mega on an old body.	Completed	• Connect the jumpers into the pins and it was damaged. • Connected the pins successfully and it was a good job.	Connection of the pins successfully.
	Working on the motion code.	In Progress	None	
12/2/2017	Body Assembly 2nd trial	Completed	The robot length was too long causing its movement hard	Failed
	Connect the H-bridge with motors on an old body.	Completed	During connecting the wires, there was something wrong in the wires.	Connection of the wires exactly.
	Continue working on line following code.	In Progress	None	
16/2/2017	Body Assembly 3rd trial	Completed	The body's thickness was 3mm, which caused it to be broken	Failed
	Connect the QTR sensor.	Completed	• Many trials were done in order to connect the jumpers. • Better connection of jumpers	All the jumpers were connected successfully.
	Continue working on line following code.	In Progress	None	
19/2/2017	Body Assembly 4th trial	Completed	None	Succeeded
	Connect the battery and the holder of the battery.	Completed	None.	• Connection of the holder using a double face tape.
	Continue working on line following code.	In Progress	None	
23/2/2017	Install components on the new body	Completed	None	Succeeded

	Connect the ultra-sonic.	Completed	None.	A successful Connection of the ultra-sonic.
	Continue working on line following code.	In Progress	None	
26/2/2017	Robot Arm Design	Completed	None.	
	Continue working on line following code.	In Progress	None	
2/3/2017	Robot Arm 1st trial	Completed	Making the 2holes (in the Robot Arm) horizontal, and this is wrong, while it must be vertical	Failed
	• Continue working on line following code. • Brainstorm green code.	Completed	None.	• Finish working on line following code. • Working on the green code.
5/3/2017	Robot Arm 2nd trial	Completed	None	Succeeded
	Working on green code using QTR sensor.	Completed	It was hard to make green code using QTR sensor, as there is no big difference between green and other colors	Failed, and the decision is to buy color sensor to detect the green color
9/3/2017	Brainstorm inverse code.	Completed	Initially, the working on inverse code was difficult.	A part of inverse code was done.
12/3/2017	Continue the work on inverse code.	In Progress	It takes time and more effort.	Another part of inverse code was done.
16/3/2017	Continue the work on inverse code.	In Progress	None.	Another part of inverse code was done.
19/3/2017	Connect the color sensor to robot chassis.	Completed	Connection of the color sensors to robot chassis was difficult.	After seeing the code of color sensors, the connection became easy.
	Continue the work on inverse code. Start working on programming the color sensor.	Completed		Successfully, inverse code was finished.

Meeting Minutes

Date: (2/2/2017)

Agenda:

- Explaining the competition rules.
- Dividing the missions between team members.
- Putting the plan.
- Deciding the components of robot S.L.S. 7

Progress/Status:

- Enthusiasm with high spirit.
- Buying some of the components of robot S.L.S. 7

Issues:

- Some parts of the robot components were not available.

Solutions:

- Searching about them in more than one store and also through websites.

Date: (5/2/2017)

Agenda:

- Check the components of robot S.L.S. 7

Progress/Status:

- Buying the rest of robot components.

Issues/Solutions:

- None

<u>Date:</u> (9/2/2017)

Agenda:

- Working on the motion code.
- Connect the Arduino mega on an old body.

Progress/Status:

- Body Assembly 1st trial
- Connection of the pins successfully.

Issues:

- The places of the nails were wrong so we couldn't assemble the components
- Connect the jumpers into the pins and it was damaged.

Solutions:

- Connected the pins successfully and it was a good job.

<u>Date:</u> (12/2/2017)

Agenda:

- Body assembly 2nd trial
- Continue working on line following code.
- Connect the H-bridge with motors on an old body.

Progress/Status:

- Connecting the wires accurately.

Issues:

- The robot length was too long causing its movement hard
- During connecting the wires, there was something wrong in the wires.

Solutions:

- Working on a shorter body
- Better connection of the wires

<u>Date:</u> (16/2/2017)

Agenda:

- Body assembly 3rd trial.
- Continue working on line following code.
- Connect the QTR sensor.

Progress/Status:

- All the jumpers were connected successfully.

Issues:

- The body's thickness was 3mm, which caused it to be broken
- Many trials were done in order to connect the jumpers.

Solutions:

- Increase the body thickness
- Better connection of jumpers

Date: (19/2/2017)

Agenda:

- Body assembly 4th trial
- Continue working on line following code.
- Connect the battery and the holder of the battery.

Progress/Status:

- Connection of the holder using a double face tape.

Issues/Solutions:

- None.

Date: (23/2/2017)

Agenda:

- Install robot components on the new body
- Continue working on line following code.
- Connect the ultra-sonic.

Progress/Status:

- A successful Connection of the ultra-sonic.

Issues/Solutions:

- None.

Date: (26/2/2017)

Agenda:

- Robot Arm Design
- Continue working on line following code.

Progress/Status:

- Robot arm design completed

Issues/Solutions:

- None.

<u>Date:</u> (2/3/2017)

Agenda:

- Robot arm 1st trial
- Continue working on line following code.
- Brainstorm green code.

Progress/Status:

- Finish working on line following code.
- Working on the green code.

Issues/Solutions:

- Making the 2 holes in the Robot Arm horizontal, and this is wrong, while it must be vertical

Solutions:

- Change the holes vertically

<u>Date:</u> (5/3/2017)

Agenda:

- Robot arm 2nd trial
- Continue working on green code.

Progress/Status:

- Completing robot arm assembly.

Issues:

- Green code failure. It was hard to make green code using QTR sensor, as there is no big difference between green and other colors

Solutions:

- Decision is to buy color sensor to detect the green color.

Date: (9/3/2017)

Agenda:

- Brainstorm inverse code.

Progress/Status:

- A part of inverse code was done

Issues/Solutions:

- Initially, the working on inverse code was difficult as QTR sensor sometimes read colors inaccurately.

Date: (12/3/2017)

Agenda:

- Continue the work on inverse code.

Progress/Status:

- Another part of inverse code was done

Issues:

- The working on inverse code was difficult as QTR sensor sometimes read colors inaccurately.

.

Solutions:

- Invest more time to change the code and identify white line in the middle of the sensor.

Date: (16/3/2017)

Agenda:

- Continue the work on inverse code.

Progress/Status:

- Another part of inverse code was done

Issues/Solutions:

- More time is required to change the code.

Date: (19/3/2017)

Agenda:

- Continue the work on inverse code.
- Connect the color sensor to robot chassis.
- Start working on programming the color sensor.

Progress/Status:

Successfully, inverse code was finished.

Issues:

Connection of the colour sensors to robot chassis was difficult.

Solutions:

After seeing the code of color sensors, the connection became easy.

Robot Components

- Battery : Li-ion Battery
- Arduino : Mega
- H-Bridge :Dual channel dc motor driver
- QTR : Pololu Sensor
- Color Sensor : 4 color sensor module
- Motors : Servo motor & 2 DC motors
- Ultrasonic:HC-o4
- Driver
- Battery holder
- 2 wheels
- Free Wheel
- BreadBoard
- Wires
- Bolts and nail
- Switch
- Motor holder
- Ultrasonic holder
- Acrylic body

Mechanical Structure and Assembly

Tasks:

- Design and assemble the body.
- Design and assemble the Robot Arm.
- Buy the gripper of the arm and items.
- Buy the body's and arm's nails.
- Make the holder of the Arm.

Task 1 - Robot Body

1st Trial

Problem: the places of the nails were wrong so we couldn't assemble the components

Result: Failed.

2nd Trial

Problem: The robot length was too long causing its movement hard.

Result: Failed

3rd Trial

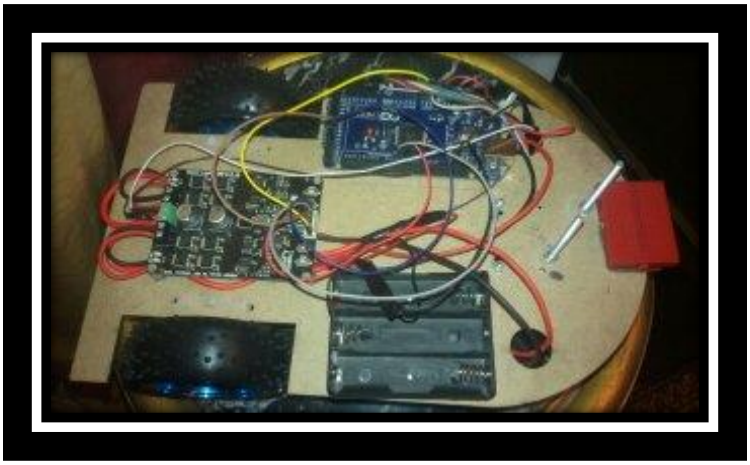
Problem: the body's thickness was 3mm, which caused it to be broken.

Result: Failed

4th Trial

Problem: None.

Result: Succeeded



Task2 - Robot Arm:

1st Trial:

Problems

- Making the 2holes (in the Robot Arm) horizontal, and this is wrong.

Solutions

- Making the holes (in the Robot Arm) vertical,

2nd Trial:

Problems: None

Result: Succeeded.

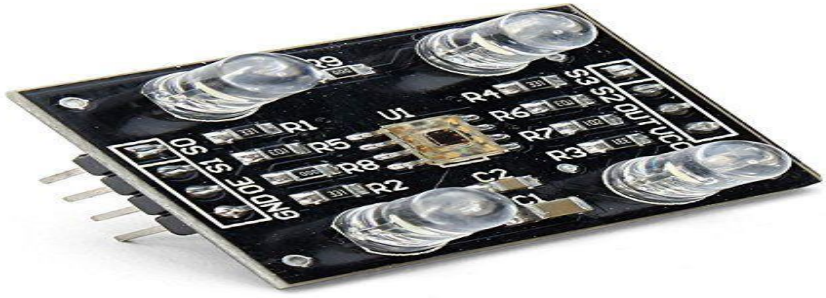
Electrical Assembly

- Arduino Mega: The Arduino Mega 2560 is a microcontroller board based on the ATmega2560.It has 54 digital input/output pins (of which 14 can be used as PWM outputs),16 analog inputs,USB connection,a power jack,an ICSP header,and a reset button.

- H-Bridge: This is a dual motor driver which is designed to drive 2 DC motor with high current up to 10A continuously and 30A peak. The driver works in voltage range from 5 to 25 V. It also includes fast test switch for driver testing.

- Line Following Sensor: Working voltage is 5V, 8-Chammel infrared detector.

- Color Sensor: This Color Sensor is a complete color detector, including a TCS230 RGB sensor chip and 4 white LEDs. The TCS230 can detect and measure a nearly limitless range of visible colors.
    - Applications include test strip reading, sorting by color, ambient light sensing and calibration, and color matching. The TCS230 has an array of photo detectors, each with either a red, green, or blue filter, or no filter (clear). The filters of each color are distributed evenly throughout the array to eliminate location bias among the colors.
    - Single-Supply Operation (2.7V to 5.5V)
    - High-Resolution Conversion of Light Intensity to Frequency
    - Programmable Color and Full-Scale Output Frequency
    - Power Down Feature
    - Communicates Directly to Microcontroller/Arduino
    - S0~S1: Output frequency scaling selection inputs
    - S2~S3: Photodiode type selection inputs
    - OUT Pin: Output frequency
    - EO Pin: Output frequency enable pin (active low)



- Motor: This is high torque (20 kg.cm) DC motor fitted with a 75:1 metal gearbox. The gears are all steel and the output shaft is 4mm diameter.

    - Specifications
    - Voltage = 12V
    - No load current = 250mA
    - Stall Current = 5 A
    - Stall Torque = 20 Kg/cm
    - Motor Rated RPM = 133

- Servo Motor of Robot Arm: This is a 3.2 kg.cm servo motor. Unlike dc motors, with servo motors you can position the motor shaft at a specific position (angle) using control signal. The motor shaft will hold at this position as long as the control signal not changed. This is very useful for controlling.

Voltage	4.2-6V
Speed	0.18 sec/60degree (4.8v), 0.16 sec/60degree (6v)
Torque	3 kg.cm (4.8v), 3.2 kg.cm (6v)

- Ultrasonic: The HC-SR04 ultrasonic sensor uses sonar to determine distance to an object – just like bats or dolphins do. This module offers excellent range accuracy and stable readings in an easy-to-use package. It operation is not affected by sunlight or black material (although acoustically soft materials like cloth can be difficult to detect). It is similar in performance to the SRF05 but with the low-price of a Sharp infrared sensor. HC-SR04 ultrasonic module can offer non-contact distance sensing function, the range of which is 2cm-400cm and the range accuracy is up to 2mm;

Programming

- Programming was started from the first moment of designing our robot.
- We programmed our robot with Arduino Mega
- On 2/2, our coach explained the competition rules, also, he decided who will be the mechanical, electrical and programmer.
- On 9/2, our coach tested if we were qualified to program the motions of motors but it was the robot act as a sample that we tried on it.
- The line following function is called Void Follow_Max (int max_num) and it depends on which definite sensor (we define sensors by numbers) that the black line is under. The movement of robot depends on the black line under which sensor. Its status is number 1. When the status no. is 1, we write in the code: Follow_Max(arranged _sensors [0]);
- The inverted function is the same as line following function but it follows the white line. Its status is number 5. When the status no. is 5, we write in the code: Follow_Max(arranged _sensors [7]);
- The green function is named Void Green, it works when the status number is 2 which means crossing. Robot goes backwards to find the green square and turns towards the green square. We are working on it.
- The ultrasonic function is called Void Obstacle, but we are working on it.
- The evacuation area function uses two ultrasonic sensors to find the ball and we are working on it.

Core functions:

1) **Arrange-sensors:**

   It is the main function in the whole code. It arranges the sensors descendingly and prints the first no. according to its value

2) **Status:**

Number	Meaning	What to do
0	White	Stay as it was moving
1	Line following	Follow-max(arranged-sensors[0]);
2	Crossing	Green ( );
3	Ninety degree right	Right-reverse ( );
4	Ninety degree left	Left-reverse ( );
5	Inverted	Follow-max(arranged-sensors[7]);

3) **Motions:**

Function of motion	Description of motion
Void forward( );	Makes the robot go by moving the two motors forward.
Void right (int sped);	Makes the robot go right by stopping right motor and moving the left motor.
Void left (int sped);	Makes the robot go left by stopping left motor and moving the right motor.
Void right_reverse ( );	Makes the robot go right by moving right motor backwards and moving the left motor forwards.
Void left_reverse ( );	Makes the robot go left by moving left motor backwards and moving the right motor forwards.
Void stop_pp ( );	Stops the two motors
Void backward ( );	Makes the robot go by moving the two motors backward.

Programming Code Samples:

1) Line following

```
void Follow_Max (int max_num )
{

 if(max_num==0) //|| (sensorValues[Arranged_Sensors[7]]==0))
   {
     Right_Reverse ();
   }
   else if(max_num==1)// || (sensorValues[Arranged_Sensors[7]]==
0))
   {
     right (turn_speed_med);
   }
   else if(max_num==2) //|| (sensorValues[Arranged_Sensors[7]]==
0))
   {
     right (turn_speed_min);
   }

   else if(max_num==3) //|| (sensorValues[Arranged_Sensors[7]]==
0))
   {
     forward (Maxspeed);
   }
   else if(max_num==4) //|| (sensorValues[Arranged_Sensors[7]]==
0))
   {
      forward (Maxspeed);
   }
   else if(max_num==5) //|| (sensorValues[Arranged_Sensors[7]]==
0))
   {
      left (turn_speed_min);
   }
   else if(max_num==6) //|| (sensorValues[Arranged_Sensors[7]]==
0))
   {
      left (turn_speed_med);
   }
   else if(max_num==7) //|| (sensorValues[Arranged_Sensors[7]]==
0))
   {
      left_Reverse ();
   }

}
```

```
  else if(status_num==1)              //Line Following
{
    Follow_Max (Arranged_Sensors[0]);
}
```

2) Crossing

```
   else if(status_num==2)                    //Crossing
 {
     Green();
 }
```

3) Ninety degree right

```
  else if(status_num==3)                   //Ninety Right
{
     Right_Reverse ();

}
```

4) Ninety degree left

```
   else if(status_num==4)                     //Ninety Left
 {
     left_Reverse ();

 }
```

5) Inverted

```
if(inverse==1)
{
   Follow_Max (Arranged_Sensors[7]);

   if ((Arranged_Sensors[0]==2) || (Arranged_Sensors[0]==3) || (Arranged_Sensors[0]==4) || (Arranged_Sensors[0]==5))
   inverse=0;
}
```