UNIVERSITY OF BALAMAND

ISSAM FARES FACULTY OF TECHNOLOGY

Graduation Project Report

Smart tricycle

Prepared by

Omar Omar

Submitted to the Department of Mechatronics Engineering as a Partial Fulfillment of the requirements for the award of

BACHELOR OF TECHNOLOGY

Supervised by

Dr. Taghrid Mazloum

Academic year: 2019 - 2020

Acknowledgment

Special thanks and appreciation to the dean of the IFFT Dr. Elie Karam, the Chairman of the Mechatronics department Dr. Rodrigue Imad, all the jury members, professors and teachers who helped us at each step to accomplish this graduation project.

Special gratitude to my supervisor for the Bachelor of Technology project Dr. Taghrid Mazloum, who supported me to overcome difficulties, and for her inspirations, encouragement, and guidance throughout my project.

I am very thankful for the greatest faculty members who guided me, taught me a lot in the department of Mechatronics, and prepared me for the future.

Abstract

The present project aims to develop a smart electric tricycle in order to provide certain travel autonomy and safety for handicapped people. Certain level of intelligence is added to the tricycle using Arduino boards and various sensors. This project involves the use of a mobile application, installed on a smart phone, which allows controlling the speed and movement of the electric tricycle. An ultrasonic sensor is installed on the back of the tricycle, avoiding any shock with close obstacles. Moreover, a warning system is developed allowing sending, to a specific number, a rescue SMS with GPS coordinates. The rescue message is sent when the handicapped person presses a button and that when any problem occurs. Furthermore, an automatic lighting system is developed to turn on the projector when the light goes below the visible region of our eyes, e.g., in a tunnel, in a garage or in the night. Finally, we investigate renewable energy, i.e., the solar panels, in order to charge the batteries of the electric tricycle.

List OF TABLES

List OF FIGURES

Table of content

Chapter1

Introduction and System Description

1.1 Context and objective

An estimated of 15% of the Lebanese population in 2013 has disabilities, including physical disabilities [1]. No transportation facilities are provided for people with disabilities. Hence, they suffer from lack of autonomy and it is difficult to have a job in the society. In this context, we aim in the present project to bring increased mobility to handicapped people, facilitate freedom in travel and help them contributing to the community. These objectives are achieved by adding some intelligence to an electric tricycle, using Arduino boards and various electronic sensors. In order to facilitate the usage of the tricycle by a disabled person, we developed a control system allowing controlling the movement and the speed of the tricycle using a mobile application. Moreover, the tricycle will be occupied by a warning system, dedicated to send a rescue message to a specific person in case of any urgency or any problem. The SMS will contain the GPS coordinates of the handicapped person. Moreover, an ultrasonic sensor is installed on the back of the tricycle in order to automatically stop the tricycle and then avoid obstacles. We equip the electric tricycle with a solar charging system to recharge the batteries and avoid their draining. Furthermore, a lighting system is developed in order to automatically turn on the projector when the light of outside is not enough. All these systems together made our tricycle unique. Thus, owing to the proposed smart tricycle, handicapped people can travel alone with certain level of autonomy and certain safety measures.

1.2 Related work

Figure 1 shows a tricycle project [2] that has been done in 2007 by engineers at UNIVERSITI MALAYSIA PAHANG. It consists of an electric power train that provides tricycle users with improved levels of mobility, facilitating freedom in travel. Also, the design consists of an electric motor, a drive system, motor, steering controls, power supply where the electric motor is regulated by pedaling.

Figure 2 shows another tricycle project [3] that has been done by engineering students at university of Gandhinagar, using PVC pipe, wood, high torque DC motor and a 10 k potentiometer to control the speed of the motor.

However, in the present project, we used a mobile application to control the tricycle, via Bluetooth communications. Additionally, we tried to add more features than aforementioned, such as a lighting system, an obstacle avoiding system, and a warning system that helps sending a rescue message to a specific person when any problem occurs. Moreover, we occupy the tricycle with a charging system using the renewable energy. All these systems together make the tricycle more useful for a handicapped person. All the aformentioned systems are explained with more details in the next chapter.


Figure 1: Tricycle design of project 1


Figure 2: Tricycle design of project 2

Chapter 2

Systems and components

2.1 Introduction

In this project, we used two Arduino boards, i.e., the Arduino UNO (Figure 3) and the Arduino Mega (Figure 4). The Arduino UNO is used for the warning system. However, the Arduino Mega is used for the remaining systems, including the control system. Tables 1 and 2 present the datasheet of the Arduino UNO and Mega, respectively. At the beginning of this project, the warning system is implemented using an Arduino UNO. Then, when we started working with the control system, we recognized the need for more input/output (I/O) pins. That is why we used the Arduino Mega board, in addition to the Arduino Uno.

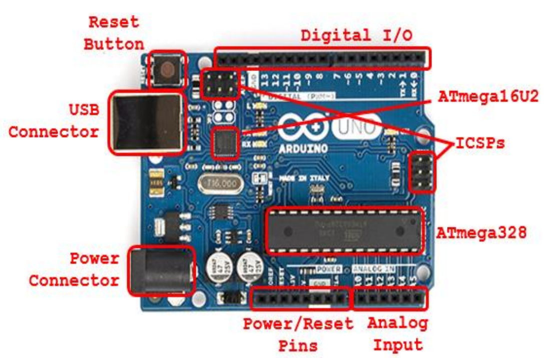

Figure 3. Arduino UNO

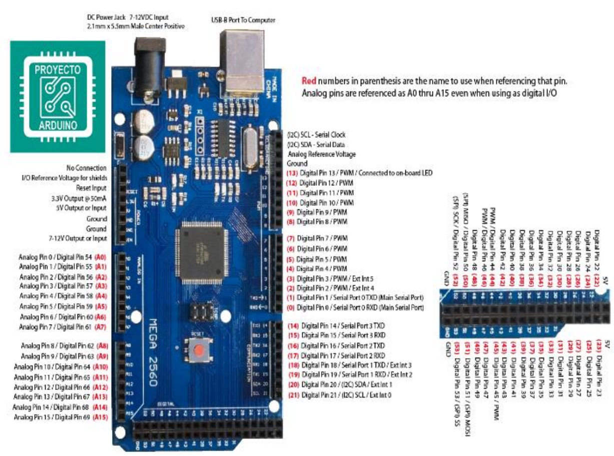

Figure 4. Arduino Mega

Microcontroller	ATMega328
Operating Voltage	5 V
Input Voltage (recommended)	7 – 12 V
Input Voltage (limits)	6 – 20 V
Digital Input / Output Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA

Table 1. Arduino UNO datasheet

Summary

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 14 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

Table 2. Arduino Mega datasheet

In the following sections, we describe each system apart and in more details.

2.2 The control system

In order to facilitate the use of the tricycle by a handicapped person, we design a wireless control system using Bluetooth serial communications. We install on a smart phone a mobile application, which will be paired with the Bluetooth module set on the tricycle. By using the mobile application, the handicapped person can control the movement and the speed of the tricycle. The mobile application is shown in Figure 5. The labels shown in Figure 5 are commands used to control the tricycle, as described in Table 3.

Symbol	**Description**
W	move forward at speed 100
S	move backward
P	stop
M	move forward at speed 150
N	move forward at speed 200
X	move forward at the highest speed (250)

Table 3. The description of the symbols used in the mobile application.

Figure 5: The mobile application

Components

The components used in the control system are the following:

- Arduino Mega
- Bluetooth module HC-05 (Figure 6): HC-05 module is a Bluetooth SPP (Serial Port Protocol) module, which means it communicates with the Arduino via the Serial Communication. It is used to receive commands from the mobile application, in order to control the tricycle movements and speed, as shown in Table 3.
- Motor shield (Figure 7): It is a driver module that allows using Arduino boards to change the direction and speed of the motors. The motor system is shown in Figure 8.
- Ultrasonic sensor (Figure 9): Ultrasonic sensor is similar to radar or sonar. It allows measuring the distance to any obstacle. The datasheet is shown in Table 4. In this project, we use the ultrasonic sensor, installed in the back of the tricycle, to stop the tricycle automatically when any obstacle is close.

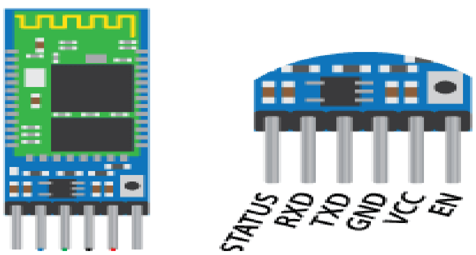

Figure 6. The HC-05 module

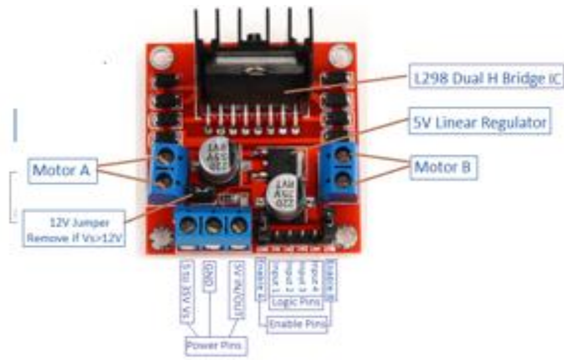

Figure 7. The motor shield

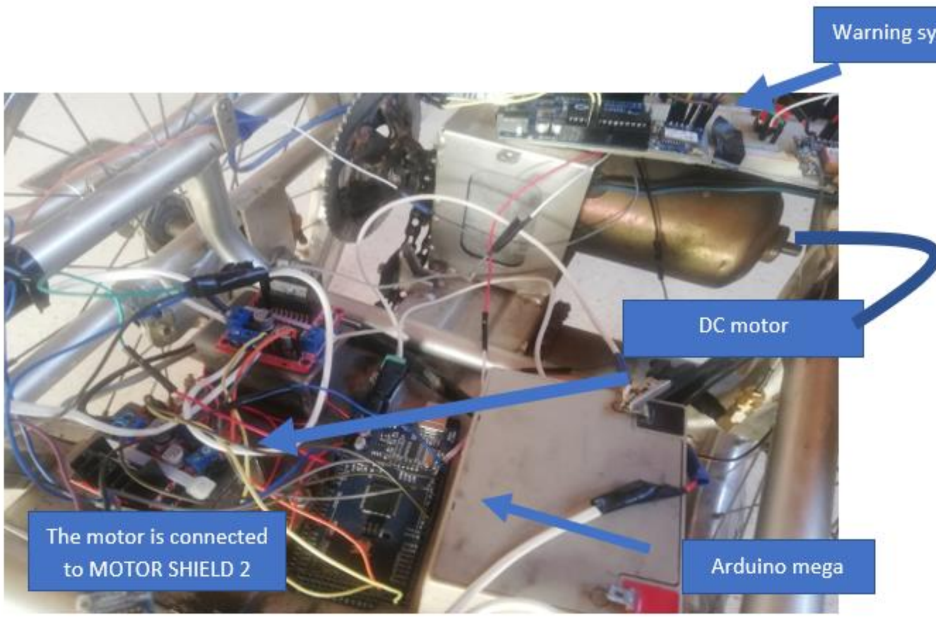

Figure 8. The motor system

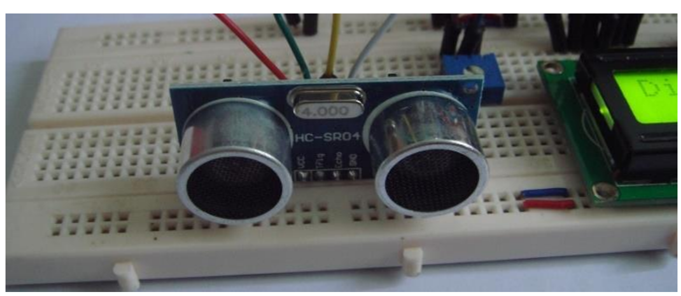

Figure 9. The ultrasonic sensor module

Part Number	MA40S4R/S
Construction	Open structure type
Using method	Receiver and Transmitter (dual use) type
Nominal frequency (kHz)	40
Sound Pressure (dB)	120±3 (20Pa)
Directivity (deg)	80
Detectable range (m)	0.2 – 4
Dimension (mm)	9.9φ x 7.1 height
Input voltage (Vp-p)	20 (40kHz) continuous signal

Table 4. The Ultrasonic sensor datasheet

2.3 The warning system

We equip the tricycle with a warning system, dedicated to send a rescue message to a specific person in case of any urgency or any problem. Indeed, when any problem occurs (a sudden shut down of the light projector, the motor stops moving or even the brakes stops working…), the handicapped person presses a red button, and then a text message will be send to a specific number. The SMS will contain the GPS coordinates (i.e., the exact longitude and latitude) of the handicapped person. Therefore, the receiver should only paste the location on the google map's application to find the handicapped person. The GPS coordinates are also shown on a 16x2 LCD screen.

When we test the code, we get the longitude and latitude of our current location on the serial monitor. If we want to make sure these values are correct, we paste the latitude and longitude on the google search and then we get our location, thus the values are correct as shown in Figure 10 (current location: At Asheer Daye street, Tripoli). An example of the message sent is shown in Figure 11.

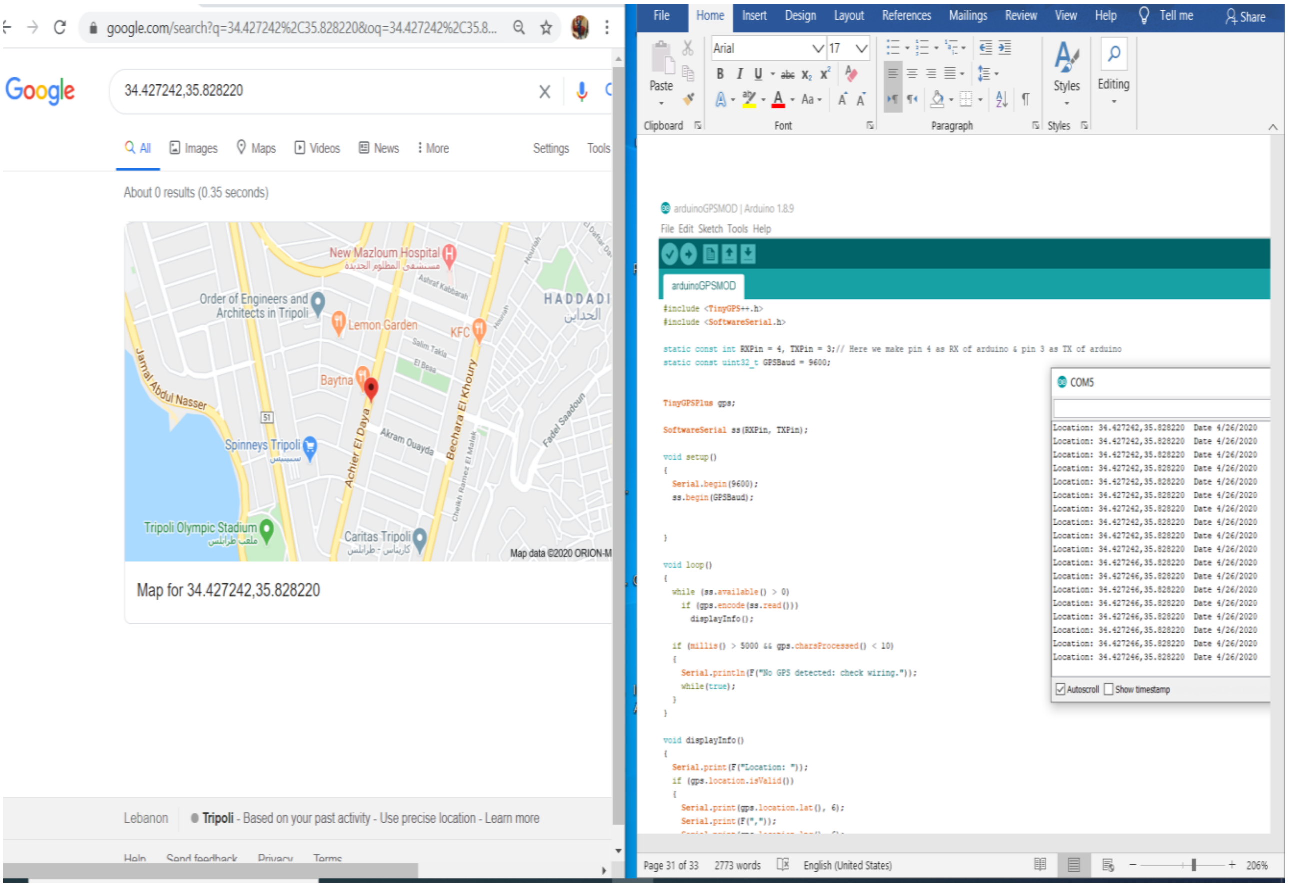

Figure 10: the GPS coordinates on google map

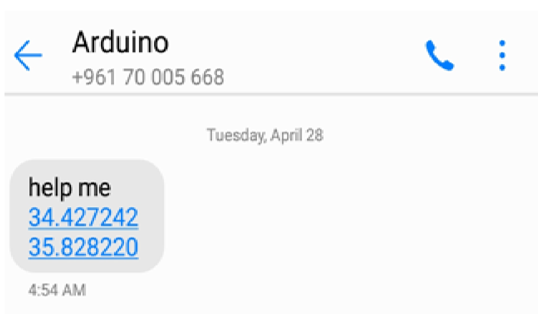

Figure 11: An example of the rescue message with GPS coordinates

Components

The components used in the control system are:

- The Arduino UNO.
- Sim 800l: SIM800L is a miniature cellular module which allows for GPRS transmission, sending and receiving SMS and making and receiving voice calls. This board is shown in Figure 12.

- Neo6m is a GPS module that determines the coordinates of the actual location. It is shown in Figure 13. There is an LED on the NEO-6M GPS module, which indicates the status of Position Fix. The board datasheet is shown on Table 5.
- 10K potentiometer (Figure 14): As we need to change the contrast of the LCD screen, a 10K potentiometer is used.
- The 16x2 LCD (Figure 15): The LCD screen has two lines and can display 16 characters per line. Therefore, it can display up to 32 characters at once. It is possible to display more than 32 characters with scrolling though. The datasheet of the 16x2 LCD is shown in Table 6. This LCD is used to show the GPS coordinates to keep track and verify the values given by the module, as shown in Figure 16.

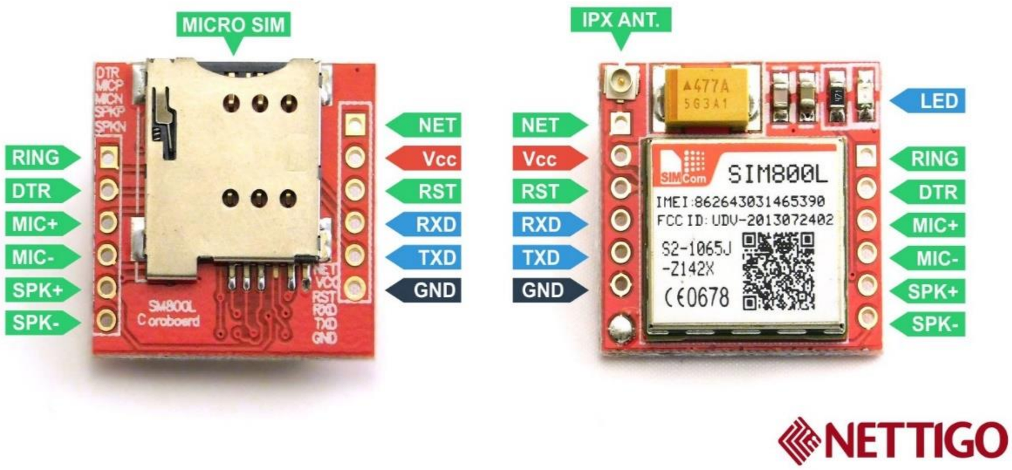

Figure 12. Sim 800L board annotation

Figure 13. Neo6m GPS module

Figure 14. 10K potentiometer

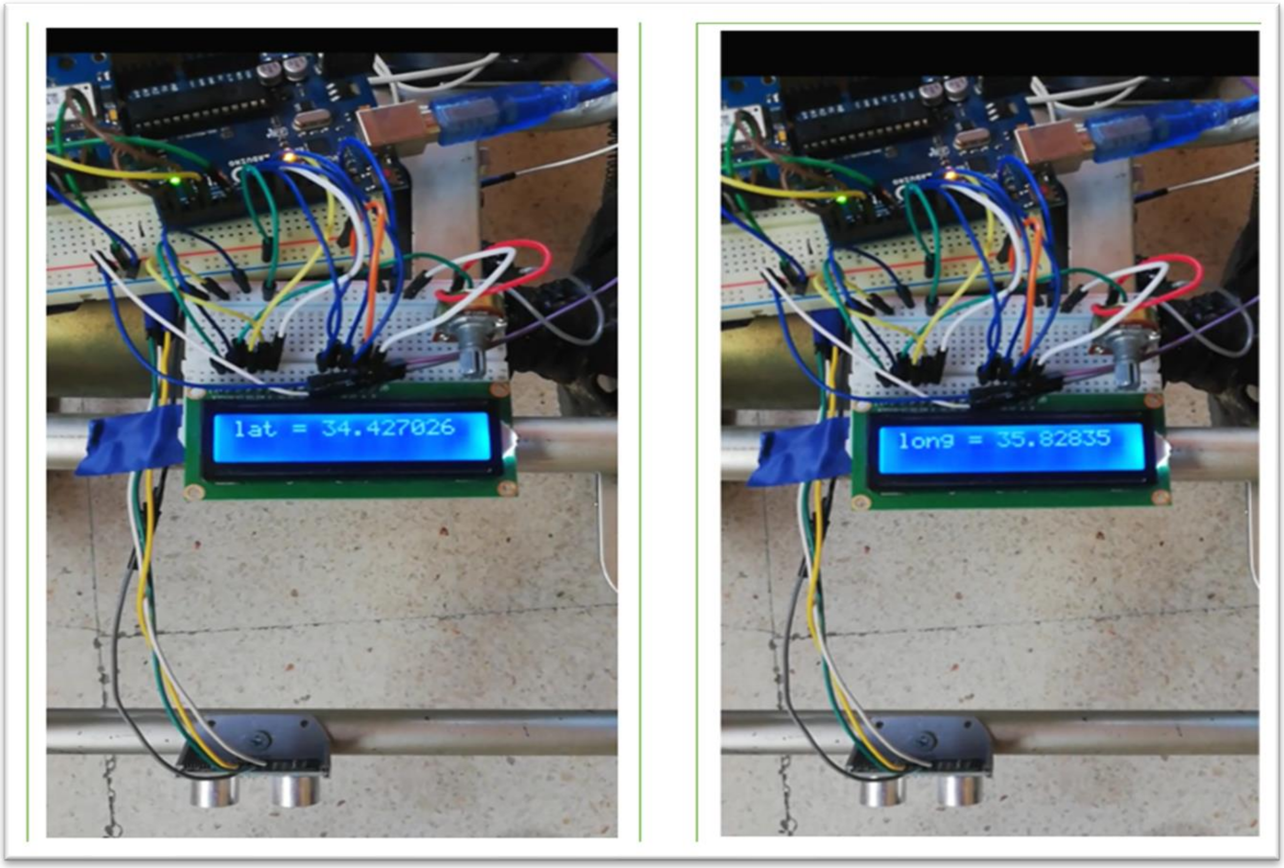

Figure 15. The GPS coordinates shown on an LCD screen.

Receiver Type	50 channels, GPS L1(1575.42Mhz)
Horizontal Position Accuracy	2.5m
Navigation Update Rate	1HZ (5Hz maximum)
Capture Time	Cool start: 27sHot start: 1s
Navigation Sensitivity	-161dBm
Communication Protocol	NMEA, UBX Binary, RTCM
Serial Baud Rate	4800-230400 (default 9600)
Operating Temperature	-40°C ~ 85°C
Operating Voltage	2.7V ~ 3.6
Operating Current	45mA
TXD/RXD Impedance	510 ohm

Table 5. Neo6m module datasheet

Pin	Symbol	I/O	Description
1	V_{SS}	--	Ground
2	V_{CC}	--	+5 V power supply
3	V_{EE}	--	Power supply to control contrast
4	RS	I	RS = 0 to select command register, RS = 1 to select data register
5	R/W	I	R/W = 0 for write, R/W = 1 for read
6	E	I/O	Enable
7	DB0	I/O	The 8-bit data bus
8	DB1	I/O	The 8-bit data bus
9	DB2	I/O	The 8-bit data bus
10	DB3	I/O	The 8-bit data bus
11	DB4	I/O	The 8-bit data bus
12	DB5	I/O	The 8-bit data bus
13	DB6	I/O	The 8-bit data bus
14	DB7	I/O	The 8-bit data bus

Table 6. LCD datasheet

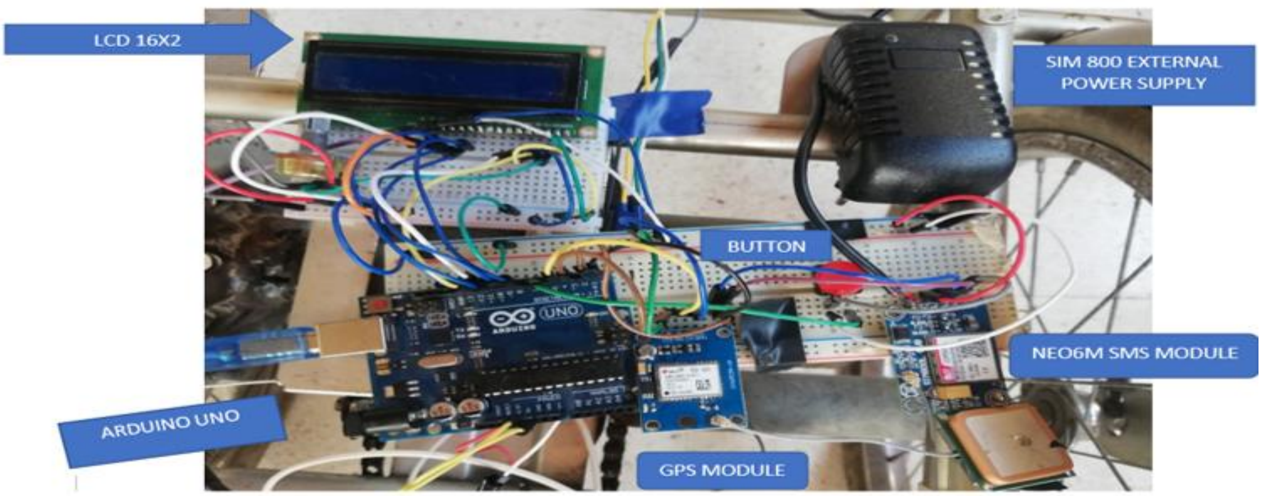

Figure 16. The warning system overview.

2.4 The lighting system

The main objective of the lighting system is to allow the handicapped person to use the tricycle during day and night. For this purpose, we will use an automatic light sensor (i.e., the LDR sensor), to power a small 5V projector set on the tricycle. If the digital value of the LDR sensor is greater than or equal to 650, the light turns on with a full intensity ”255” (dark level reached). If the value is between 450 and 650, the light turns on with a medium intensity ”155”. Otherwise, the light turns off.

Components

The components used are listed as following:

- A small 5V projector, as shown in Figure 17.
- LDR SENSOR BOARD: The sensor that can be used to detect light is an LDR. Since the LDR gives out an analog voltage, it is connected to the analog input pin on the Arduino. The Arduino, with its built-in ADC (analog-to-digital converter), then converts the analog voltage (from 0-5V) into a digital value in the range of (0-1023). The sensor datasheet is shown in Table 7.

Figure 17. The 5V light projector

Pin	Description	Function
VCC	+3.3V~+5V	Connect to +3.3V ~ +5V
GND	0V	Connect to Ground
D0	Digital Output	1. Output Signal: HIGH • Surrounding intensity of light reaches the pre-set level. (Set by sensitivity adjustor) • LED status: ON  2. Output Signal: LOW • Surrounding intensity of light does not reach the pre-set level. (Set by sensitivity adjustor) • LED status: OFF 
A0	Analog Output	Analog output varies due to intensity of light.

Table 7. The LDR module datasheet

2.5 The charging system

The charging system consists of three main components: solar panels, a charge controller and batteries. The number of required solar panels and that of required batteries are discussed in the Chapter 3.

The tricycle battery stores electrical energy for use, acting as a buffer for the motor's electrical system. The batteries are required to be charged using the renewable energy, in order to keep working the motors. Thus, we needed to implement an off grid solar system with one autonomy day, to charge the tricycle. When the batteries become overcharged, the charge controller acts as a switch and open the circuit (cut of the circuit) until the battery voltage goes below 12V. This system charges the 12V batteries as long as there is sunlight. It will keep supplying the tricycle for a very long time, yielding to undrainable batteries. This makes such a charging system different from the others (ac charging techniques).

Components

For the charging system, the following components were used:

- Three 12 V batteries (200Ah, Rechargeable). The reason why we choose those batteries is mentionned in chapter 3. Those batteries help us supplying the electric circuit.The characteristics of the 12V battery are:
  Standby use: 13.60V~13.80V .
  Cycle use: 14.40v~15.00V.
  Initial current: 60 A max.
- Four solar panels to charge the batteries. The calculation is mentioned in Table 10 (ch3)and the panel specification is shown in table 8.
- A 12 V \24 V Charge controller 10 A, or battery regulator limits the rate at which electric current is added to or drawn from electric batteries. It prevents overcharging and may protect against overvoltage, which can reduce battery performance or lifespan and may pose a safety risk.

Figure 18. A 12\24 V charge controller 10 A

Electrical Parameters	**Description**
Power Max (Pm)	250
Short Circuit Current (Isc)	8.95 A
Max Power Current (Imp)	8.35 A
Maximum Voltage (Vmp)	29.95
Open Circuit Voltage (Voc)	37.25
System Voltage	1000 VDC
Mechanical Specifications:	
Type	Monocrystalline Panel
No of Cells in Series	60
Frame Type	Aluminum
Y-Axis Mounting Hole	946
X-Axis Mounting Hole	6.9
Junction Box Cable	4mm
Glass Type	Tempered 4mm
Selling Price	**Rs.11,000**

Table 8. Solar panel datasheet

Chapter 3

Systems design and implementation

3.1 Charging system design

In order to buy the convenient components from the market and ensure good system performance, we made an accurate calculation by following several steps, as shown in Figure 19.

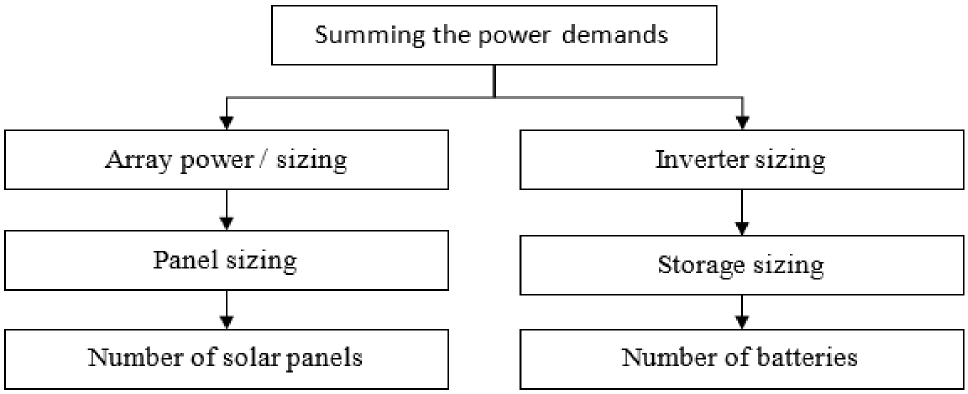

Figure 19. The steps required in the design of the charging system

The first step consists of summing the power demands by calculating the power consumption of each component apart. Table 9 shows the details of the calculation, where the voltage (V) and the current (I) consumed are shown for each component. The power ($P = V \times I$) consumed by each component is also shown. Then, we add all the power demands calculated in Table 9 and we get the total power (P_{tot}) consumed by the tricycle, which is equal to 191.2 W.

Component	$\boldsymbol{V}$ (**V**)	$\boldsymbol{I}$ (**A**)	$\boldsymbol{P}$ (**W**)
Arduino uno	5	0.046	0.232
Arduino mega	5	0.054	0.27
LDR	5	0.01	0.05
HC-05	5	0.03	0.15
SR04	5	0.015	0.076
Motor			150
Light projector	5	1	5
Sim 800 l	5	2	10
Neo6m module	5	0.067	0.335
Motor shield	12	2	25
16x2 LCD	5	0.022	0.11
P_tot			**191.222**

Table 9. The details of the power consumption of the tricycle.

The calculation of the remaining steps shown in Figure 19 is presented in Table 10. Assuming that the handicapped person is going to use the tricycle for 18 h (day and night), the final power demand is obtained by multiplying the total power consumption (191.2 W) by 18 h. The first part of calculation consists on finding the required number of solar panels. Hence, the second step consists of calculating the array power. This step is accomplished by multiplying the final power demand by the coefficient of energy loss in the system, which is equal to 1.3. Then, the third step consists of calculating the panel sizing. We divide the array power by the Lebanese pgf and by the solar panel peak power shown in Table 8. Hence, we find that we need, for the design of the smart tricycle, four solar panels of 250 W each. On the other hand, the second part of calculation aims to find the number of batteries. In Step 4, the inverter sizing is found by multiplying the total power consumption by the same coefficient used before (which is equal to 1.3). Then we need to round the inverter size and hence we obtain 250 W. In step 5, the storage sizing is found by multiplying the final power demand by 2 (2 autonomy days which is optional) and then dividing the result by the short circuit current multiplied by the battery efficiency and the discharge coefficient. Note that these values are given in the solar panel specification (Table 8). Last but not least, we get the number of batteries by dividing the storage size by 200 Ah (the batteries capacity, available in the Lebanese market). Hence, we need three batteries. Due to COVID-19 and the economic situation in Lebanon, we are not able to implement the charging system, leaving this step for future work.

		Description
Power demand	$191.222 \times 18\text{h} = 3441.6\ \text{Wh/day}$	
Array power	$3441.6 \times 1.3 = 4474.08\ \text{Wh/day}$	1.3 is the coefficient of energy loss in the system
Panel sizing	$4474.08/(4.8 \times 250) = 4$	Lebanon pgf = 4.8
Inverter sizing	$191.222 \times 1.3 = 248.5\ \text{W}$	So we need 250 W inverter
Storage sizing	$(3441.6 \times 2)/(0.6 \times 0.85 \times 12) = 562.35\ \text{Ah}$	Multiplication by 2 which is autonomy days battery efficiency = 0.85 discharge coefficient = 0.6
Number of batteries	$562.35/200 = 3$	200 Ah battery

Table 10. The calculation of the number of solar panels and the number of batteries.

3.2 DC motor design

When designing a dc motor, we should take into consideration many factors such as i) the total mass including the tricycle and the user that we want to carry, ii) the speed that we wish to reach and finally iii) the wheel size. Then we will do some calculation as shown in Table 11. As a result, we will determine the motor power in order to find the convenient one for the tricycle.

Total mass	250 Kg (including both the mass of the tricycle and the user)
Required speed	20 Km/h
Total weight	2500 N
Weight on each wheel	833 N
Wheel size	Cylinder of a radius r =15 cm and length $\mathrm{L} = 8$ cm
Moment of inertia	2.38 Kg.m^2
Angular velocity (v/r)	37.037 rad/s
Angular acceleration α	7.4 rad/s^2
Torque τ	17.31 N.m
Speed of revolution	23 rpm
Power per wheel	42 W (approximately)
Total power	126 W

Table 11. The motor design calculation

As shown in Table 11, the total mass is 250 Kg. Since the mass of the tricycle is 80 Kg, the rest of mass is dedicated to carry a weight up to 170 Kg. The total weight is obtained by multiplying the total mass by 10 m/s² (assuming g=10m/s²). Then the weight on each wheel is equal to the total weight divided by 3, implying 833 N. Assuming each wheel as a cylinder of radius $\mathrm{r} = 15$ cm and length $\mathrm{L} = 8$ cm, we calculate the moment of inertia about diameter using the parallel axis theorem at wheel surface given by $I = (1/4) \times \mathrm{mr}^2 + (1/12) \times \mathrm{mL}^2 + \mathrm{mr}^2$. Since the initial angular velocity is $\Omega_1 = 0$ rad/s, the angular velocity Ω_2 is equal to $\frac{\mathrm{V}}{\mathrm{r}} = 20 \mathrm{~Km/h} \times (5/18)/0.15$ and in this case V is multiplied by 5/18 to convert it from Km/h to m/s. Suppose that it takes 5 s to reach 20 Km/h, the angular acceleration is $\alpha = \text{change in angular velocity}/\text{change in time} = 7.4 \mathrm{~rad/s}^2$. The torque is $\tau = \mathrm{I} * \alpha$ and the speed of revolution (N) is equal to $60 * \mathrm{I}(\text{mom of inert})/2\,\pi$. Eventually, the power of the motor is equal to $2\pi\mathrm{N} * \tau/60$ = 42 W. Considering the three wheels, the total power is equal to 42 x 3 = 126 W. Nevertheless, the motor power used in this project is 150 W because the motor was attached to the tricycle that we borrowed from the plc lab. We note that since the charging system is not implemented, the tricycle neither could move nor carry weight. However, we used a small battery (12V 6 Ah) just to rotate the wheel and show that Bluetooth mobile application is working properly.

3.3 Systems implementation and connections

3.3.1 The control system

The Arduino Mega is connected to the motor shield, the Bluetooth module and the ultrasonic sensor. The motor shield is connected, in its turn, to the motor and an external power supply (the batteries).

The connections between the motor shield (to the left) and the Arduino Mega (to the right) are as following:

5 V pin -------------------------- Arduino Mega 5 V

GND pin-------------------------- Arduino GND pin

In3----------------------------------digital pin number 3

In4----------------------------------digital pin number 2

ENB---------------------------------digital pin number 23

The connections between the Bluetooth module (to the left) and the Arduino Mega (to the right) are as following (Figure 6):

Tx pin------------------------------- PWM pin 11

Rx pin------------------------------- PWM pin 10

Vcc-----------------------------------Arduino 5V pin

GND---------------------------------Arduino GND pin

The connections of the Ultrasonic sensor (to the left) with the Arduino Mega (to the right) are as follows:

Vcc-------------------------------- Arduino Mega 5V pin

GND------------------------------Arduino GND pin

trig ---------------------------------13

echo ------------------------------ 12

3.3.2 The warning and lighting system

The Arduino Uno is connected to the LCD screen, the SIM800l, the GPS module Neo6m and the light LDR sensor. The connections are shown in Figure 20, 21, 22 and 23, respectively.

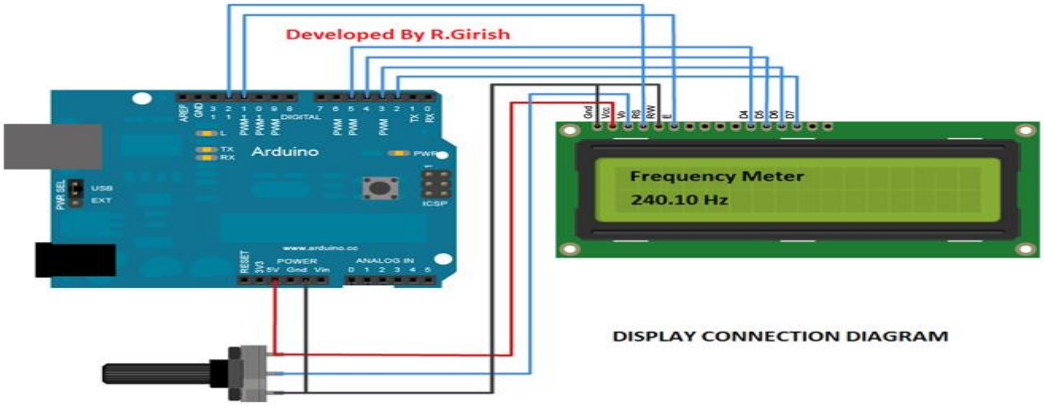

Figure *20*: LCD connection

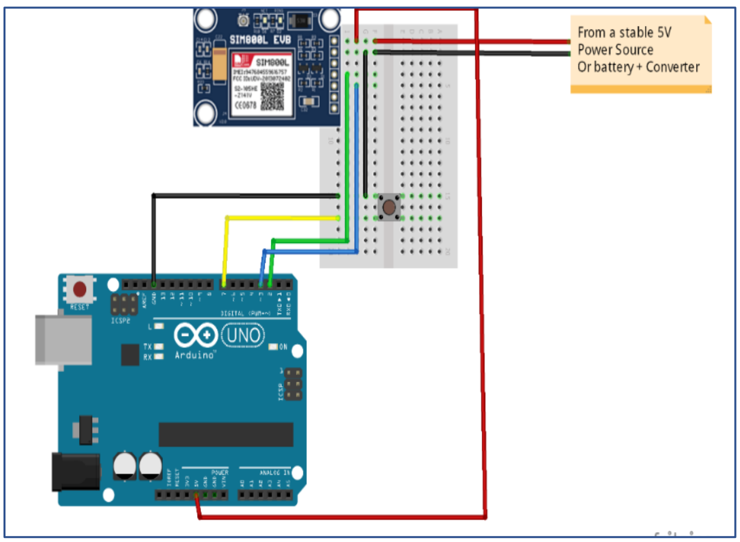

Figure 21. Sim 800L connection

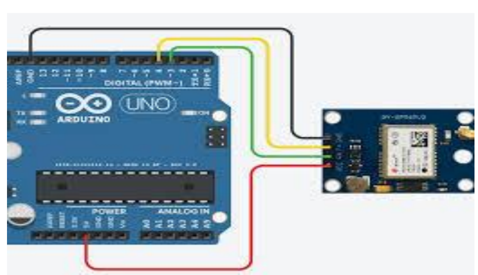

Figure 22. The GPS module connection

Figure 23: The LDR sensor board connection

Chapter 4

Conclusion and perspectives

4.1 Overview on the systems design

The present project aimed to design a smart tricycle in order to provide handicapped persons with more autonomy with certain safety measures. Such objective is achieved by investigating Arduino boards and various sensors. An overview of all the systems (Arduino boards and the various sensors) installed on the tricycle are shown in Figure 24. The warning system is controlled by Arduino Uno. All the components and sensors controlled by this board are shown in Figure 25. The other systems are controlled by Arduino Mega, as shown also in Figure 25. The tricycle full design is shown in Figure 26.

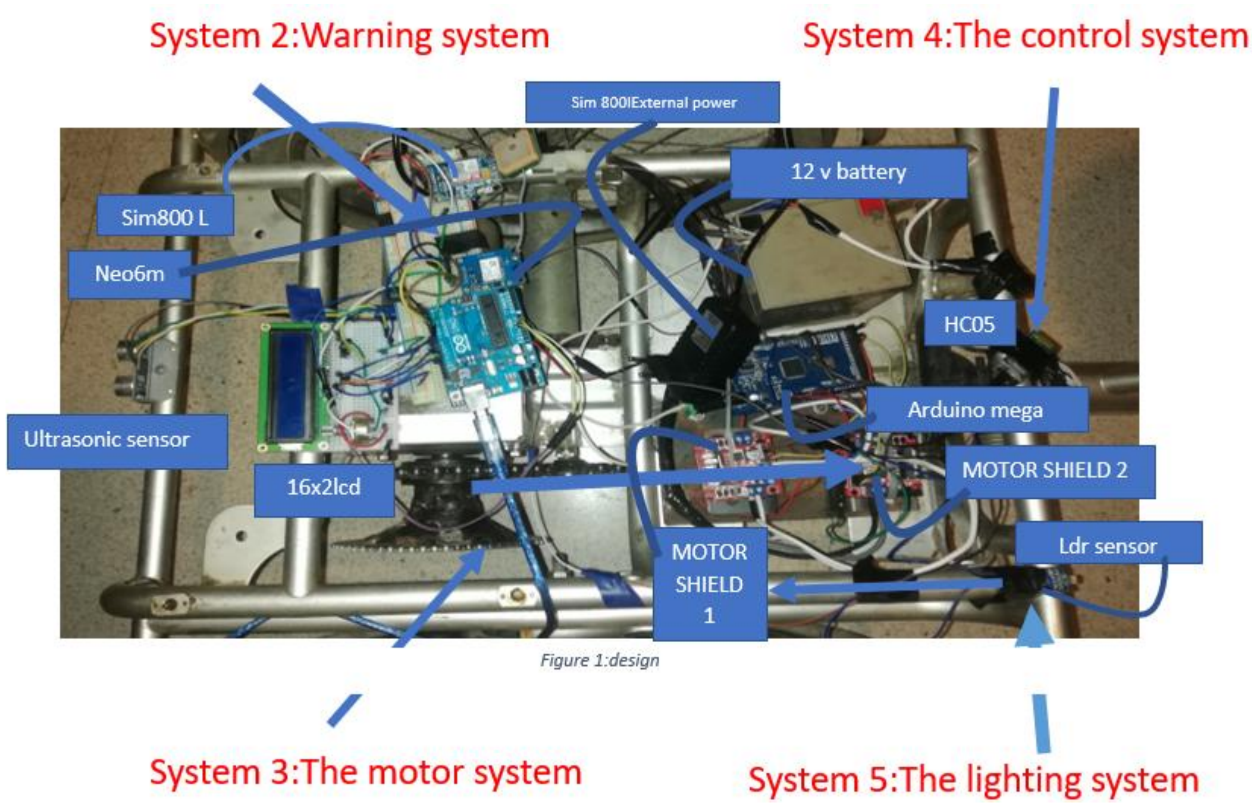

Figure 24. Overview on the systems connection

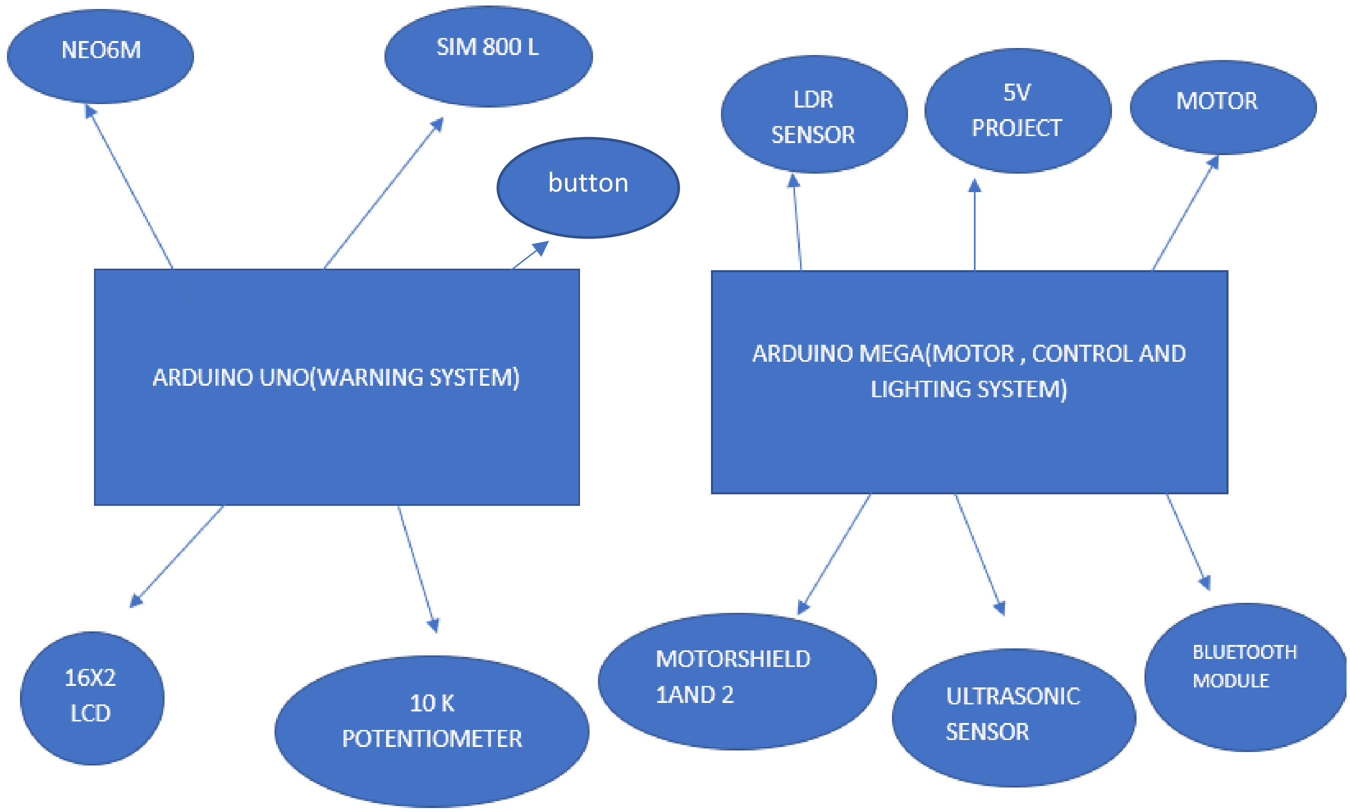

Figure 25. The sensors connected to the Arduino boards.

4.2 Summary and perspectives

The work in this project allowed me to become familiar with one of the most important technologies, which is Arduino. An experience in electronics and software is acquired since we used various sensors, including SMS module, GPS module, Ultrasonic sensor, etc. Moreover, an experience in solar charging systems is developed.

As future work, we propose to investigate an AI system in order to drive the tricycle with very little human input. This implementation will involve not only the use of many more sensors to calculate distances and detect obstacles from the tricycle position, but also the use of python language to create functions that control the tricycle manually. Furthermore, we will add a little feature to the warning system which is the ability to send the rescue message to many phone numbers in case one of them is busy by using BLYNK mobile application and connecting it to Arduino using the Bluetooth serial communication.

Finally, we will follow the design mentioned in Chapter 3 by adding the required number of solar panels and batteries to charge the system properly, as well as to achieve a maximum acceleration in order to give this project to handicapped people.

Figure 26. The Smart tricycle Hardware

REFERENCES

[1] Social Inclusion of Young Persons with Disabilities (PWD) in Lebanon http://www.unesco.org/new/fileadmin/MULTIMEDIA/FIELD/Beirut/images/SHS/Social_Inclusion_Young_Persons_with_Disabilities_Lebanon.pdf

[2] http://umpir.ump.edu.my/id/eprint/2345/1/TILAKISWARAN_ANAK_LELAKI_SAMURGAM.PDF

[3] https://www.youtube.com/watch?v=U9cazC7DBFk

Annex

This appendix contains the codes used to control the tricycle, take appropriate measurements, display the appropriate information and show it.

```
#include <SoftwareSerial.h>
SoftwareSerial sim8001(0,1); // RX,TX for Arduino and for the module it's TXD RXD, they should be inverted
#define button1 7 //Button pin, on the other pin it's wired with GND
bool button_State; //Button state
void setup()
{

  pinMode(button1, INPUT_PULLUP); //The button is always on HIGH level, when pressed it goes LOW
  sim8001.begin(9600);   //Module baude rate, this is on max, it depends on the version
  Serial.begin(9600);
  delay(1000);
}
 void loop()
{
   button_State = digitalRead(button1);   //We are constantly reading the button State

  if (button_State == LOW)
  {            //And if it's pressed
    Serial.println("Button pressed");   //Shows this message on the serial monitor
    delay(200);                         //Small delay to avoid detecting the button press many times

    SendSMS();


}
  }
 void SendSMS()
{
  Serial.println("Sending SMS...");              //Show this message on serial monitor
  sim8001.print("AT+CMGF=1\r");                  //Set the module to SMS mode
  delay(100);
  sim8001.print("AT+CMGS=\"+96170007334\"\r");  //Your phone number don't forget to include your country code, example +212123456789"
  delay(500);
  sim8001.print("SIM8001 is working");       //This is the text to send to the phone number, don't make it too long or you have to modify the SoftwareSerial buffer
  delay(500);
  sim8001.print((char)26);// (required according to the datasheet)
  delay(500);
  sim8001.println();
  Serial.println("Text Sent.");
  delay(500);
}
```

Figure 27. The sim 800L test code

```
void urmfunction(void){
  mes=distanceSensor.measureDistanceCm();
  if(mes>0&&mes<5){
    stopd();
  }
```

Figure 28: The ultrasonic sensor code

```
#include <TinyGPS++.h>
#include <SoftwareSerial.h>
static const int RXPin = 4, TXPin = 3;// Here we make pin 4
static const uint32_t GPSBaud = 9600;
TinyGPSPlus gps;
SoftwareSerial ss(RXPin, TXPin);
void setup()
{
  Serial.begin(9600);
  ss.begin(GPSBaud);
}

void loop()
{
  while (ss.available() > 0)
    if (gps.encode(ss.read()))
      displayInfo();

  if (millis() > 5000 && gps.charsProcessed() < 10)
  {
    Serial.println(F("No GPS detected: check wiring."));
    while(true);
  }
}
void displayInfo()
{
  Serial.print(F("Location: "));
  if (gps.location.isValid())
  {
    Serial.print(gps.location.lat(), 6);
    Serial.print(F(","));
    Serial.print(gps.location.lng(), 6);
  }
  else
  {
    Serial.print(F("INVALID"));
  }
  Serial.print(F("  Date "));
  if (gps.date.isValid())
  {
    Serial.print(gps.date.month());
    Serial.print(F("/"));
    Serial.print(gps.date.day());
    Serial.print(F("/"));
    Serial.print(gps.date.year());
  }
  else
  {
    Serial.print(F("INVALID"));
  }
 Serial.println();
}
```

FIGURE29. The GPS module code

```
void  fd(void){
   analogWrite(m1d,0);
   digitalWrite(m1d1,HIGH);
   analogWrite(m1s,spd);

}
void  bw(void){
     analogWrite(m1d,spd);
   digitalWrite(m1d1,HIGH);
   analogWrite(m1s,0);
}
void  stopd(void){
        analogWrite(m1d,0);
   digitalWrite(m1d1,HIGH);
   analogWrite(m1s,0);
}
```

Figure 30. The motor shield code

```
void Bt  (void){
  if(Bluetooth.available()){     //wait for data receive
     Data=Bluetooth.read();
     if(Data=='w'){
        spd=100;
      fd();
     }
     else if(Data=='s'){
      spd=150;
      bw();
     }
     else if(Data =='p'){
   stopd();
     }
     else if(Data=='m'){
      spd=150;
      fd();
     }
     else if(Data=='n'){
      spd=200;
      fd();
     }
     else if(Data=='x'){
      spd=250;
      fd();
     }
  }
}
```

Figure 31. The Bluetooth module code

```
}
void light(void){
  light1=analogRead(A1);
  if(light1>=intensity){
    a=light1-400;
  if(a<=full&&a>=0){
    digitalWrite(32,HIGH);
    digitalWrite(30,HIGH);
    analogWrite(8,a);
  }

  else{
    digitalWrite(32,HIGH);
    digitalWrite(30,HIGH);
    analogWrite(8,full);
  }}

  else if(light1<intensity){
    a=light1-350;
    if(a<=full && a>=0){
    digitalWrite(32,HIGH);
    digitalWrite(30,HIGH);
    analogWrite(8,a);
    }
    else{
    digitalWrite(32,HIGH);
    digitalWrite(30,HIGH);
    analogWrite(8,0);
    }

  }}
```

FIGURE 32. The LDR sensor code