Project of Fatigue Driving Detector

Jiong Wang
Umass Amherst
Amherst, MA
jiongwang@umass.edu

Hongyu Ba
Umass Amherst
Amherst, MA
Hongyuba@umass.educreate

Yachen Liu
Umass Amherst
Amherst, MA
yachenliu@umass.edu

***Abstract*—This paper is about the design of an equipment that can detect driver's fatigue in long-distance driving trip. MAXREFDES117 board is used to collect data from human body, the core part of which is max30102, the peripheral capillary oxygen saturation (SpO2) sensor attached to the skin. The Adafruit FLORA board is used to control the whole system, calculate and store data. According to readings of the 2 LEDs of SpO_2 sensor and corresponding formula, an estimated oxygen saturation of the user can be calculated. When the user's oxygen saturation level is lower than the warning value, an alarm will be triggered.**

***Keyword*—peripheral capillary oxygen saturation, fatigue estimation, auxiliary equipment**

I.INTRODUCTION

Fatigue driving has gradually become an increasingly serious problem in recent years. There are about 100,000 traffic accidents caused by fatigue driving per year in China. 71,000 people are injured in these accidents. In the worldwide freight industry, 57% of fatal truck traffic accidents are due to driver fatigue. Truck and coach drivers are suffering from fatigue of long-distance driving [1]. But few marketing products can help these drivers because they mostly use face detection methods. In this project, we would like to offer them a cheaper but effective equipment to detect. According to this idea, the group decide to start using the technique we can access at the university to come up with a product that can alert long-distance driver when they are too tired to drive. Our product requirements and specification are:

- Do no harm to the users and will not impact to driving.
- Should be sep up on a car.
- Data can be obtained by contacting with human body(fingers).
- Users can take measurements anytime and get instant feedback.
- Should accurately feedback the state of users

II.PRINCIPLE

A. The reaction time of fatigue driving

Through a printed research book called *Driver reaction times to familiar but unexpected events*, we found a table showing the reaction time When the driver is driving fatigued.

	Average	15th Percentile	50th Percentile	85th Percentile
Brake Assist				
Run 1 (Cars / Pedestrians Emerging)	0.85	0.67	0.81	1.02
Run 1 (Braking Vehicle Ahead)	1.30	0.80	0.99	2.01
Gantry Collapse				
Time to Apply Brake	1.53	1.18	1.35	1.84
Time to Apply Steering	1.54	1.08	1.45	2.15
Unexpected Stationary Vehicle				
Time to Apply Brake	3.52	2.17	3.35	4.79
Time to Apply Steering	5.08	4.23	5.00	6.34
Driver Fatigue				
Run 1, All Reaction Tasks	1.12	0.86	1.05	1.38

Fig.1. Driver reaction time summary table[2]

We found that when the driver's reaction time is more than 0.86s, the driver may be tired. When the driver is not tired, the reaction time is less than 0.86s [2]. Therefore, we set 0.86s as the standard value of our reaction time system.

B. The relationship between SpO_2 and fatigue driving

We know that the relationship between human fatigue and blood oxygen saturation. When the body responds to fatigue, such as feeling tired, sleepless, etc., the blood oxygen saturation will drop significantly. The decreasing value is about 6.5%. SpO_2 can be the gold standard for detecting fatigue.

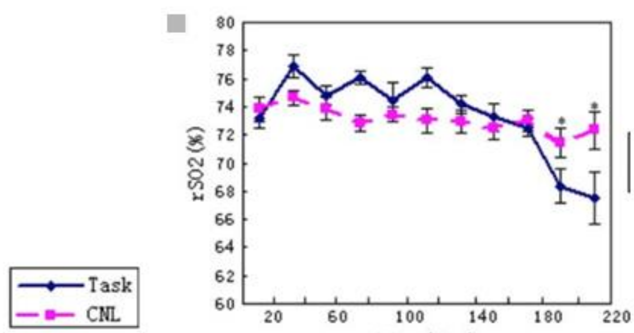


Fig.2. SpO_2 decreaing when people are tired[3]

We can know when the user is tired through reaction time experiments and then detect the user's SpO_2 value. This value of SpO_2 should be the alert value: when user's SpO_2 is lower than it, our product will alert.

III. Methods

A. Hardware Architecture

We used two development boards in our design.

The first one is MAXREFDES117, a tiny board, perfect for wearable projects. Board size is 12.7mm x 12.7mm (0.5in x 0.5in). It may be placed on a finger or earlobe to accurately detect SpO_2.

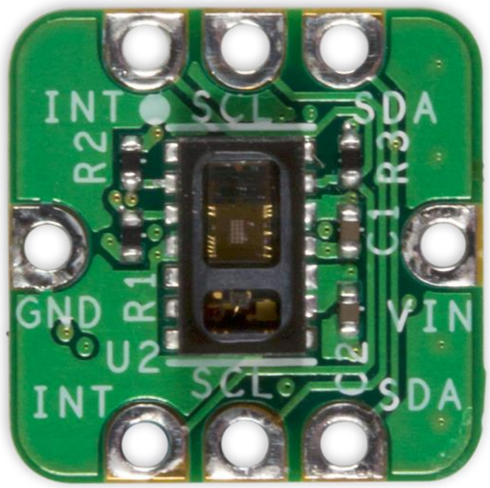


Fig.3. MAXREFDES117[4]

The main advantages of using this development board are [3]:

- It can detect SpO_2 directly;
- Firmware is available for Arduino platform, enabling us to develop it conveniently;
- The price of this board is $14.99. It is relatively low so that it meets our requirement.

MAXREFDES117 features three maxim chips:

- MAX30102: a SpO_2 sensor. It has integrated red and IR LEDs for heart rate and SpO_2 detection;
- MAX1921: an efficient, low-power step-down converter. It converts the 2V to 5.5V supply input and generates the 1.8V rail for the heart-rate sensor.
- MAX14595: an accurate level translator. This chip provides an interface between the heart rate/SpO2 sensor and the controller board, which generally use a different logic level.

The working principle of MAXREFDES117 is shown in Figure 4. Finger attached to the infrared light source of MAX30102. The Arduino controller board use 5 wires to connect MAX1921 and MAX14595.

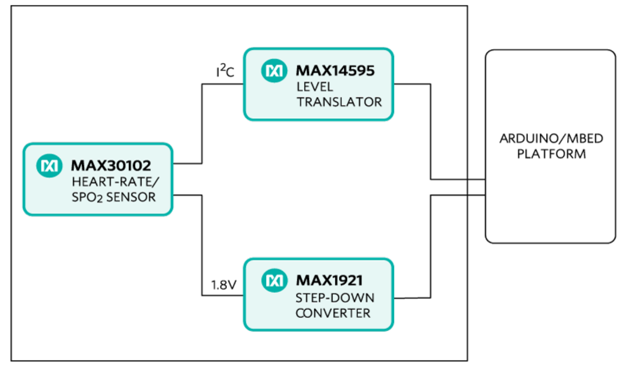


Fig.4. System Diagram of MAXREFDES117[4]

The second one is Adafruit FLORA-v3. It is an Arduino-compatible microcontroller.

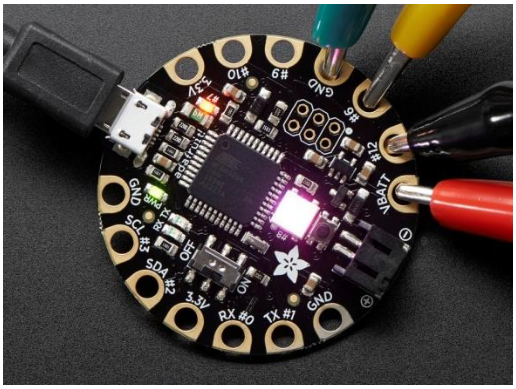


Fig.5. Adafruit FLORA-v3[5]

The main advantages of using this development board are [4]:

- It is small (1.75" diameter, weighing 4.4 grams), so that it can be made into a car equipment easily.
- It has built-in USB support, which means we can connect it to a computer and compile it directly;
- The price of this board is $14.95, which is relatively low so that it meets our requirement.

We use Adafruit FLORA-v3 as our controller board. Its function is controlling MAXREFDES117, receiving, storing and calculating data and sending an alert.

In terms of hardware design, a car equipment, fatigue driving detector, we make MAX30102 an accessible equipment to let the users put their fingers on. The readings of LEDs would be accepted by the same chip and translated to FLORA board continuously. FLORA accepts, stores and calculates the data of users. In the same time, FLORA receives power from a computer via USB cable and transfers the power to MAXREFDES117. Power would be converted from 3.3V to 1.8V through MAX1921's step-down converting and then provided to MAX30102. Once the user's oxygen saturation level is below the alert value, the alert (LED on port#7 of Adafruit FLORA) will be triggered. The whole process of our design is shown in Figure 6.

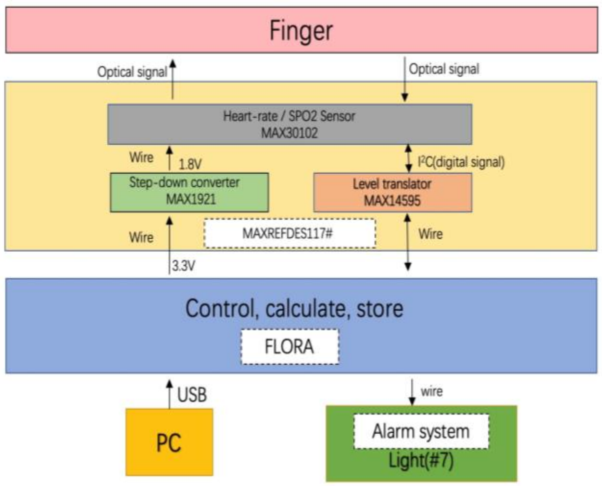


Fig.6. Block diagram of hardware architecture

B. Software Design

The process of our demo has several steps:

1. The user need to press any key on the computer to start the measurement.

2. Adafruit FLORA initialize the settings of MAX30102;

3. MAX30102 collects three seconds of data for ARDUINO as the initial data;

4. Run SpO_2 algorithm (begin the loop at the same time);

5. MAXREFDES117 removes the first second of data from the SRAM for data accuracy.

6. MAXREFDES117 transmits the data (readings of LEDs on MAX30102) to Adafruit FLORA. Adafruit FLORA stores the data and calculate the current SpO_2 of the user (if the value of SpO_2 is lower than alert value, sent a signal to alarm system).

7. Collect one second of data and go back to step 4.

The whole process of software flowchart is shown in Figure 7.

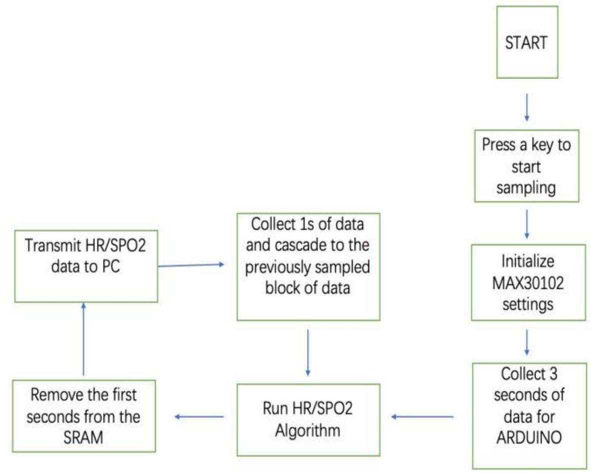


Fig.7. Software flowchart [3]

IV. ALGORITHM

A. SpO_2 Calculating

According to the SpO2 formula [4]:

$$\mathbf{SpO_2 = C_1 \times AverageRatio^2 + C_2 \times AverageRatio + C}$$

Where AverageRatio is the average ratio of IR and red LED readings, which is the readings of detected hemoglobin converted by a digital-to-analog converter

from the MAX30102, and the constants are:

C1 = -45.060

C2 = 30.354

C3 = 94.845

We can design the algorithm of SpO_2 calculating. Part of the SpO_2 algorithm is shown below. We used an array to store the value of SpO_2 in range in order to save the program running time.

const uint8_t uch_spo2_table[184]={ 95, 95, 95, 96, 96, 96, 97, 97, 97, 97, 97, 98, 98, 98, 98, 98, 99, 99, 99, 99, 99, 99, 99, 99, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 99, 99, 99, 99, 99, 99, 99, 99, 98, 98, 98, 98, 98, 98, 97, 97, 97, 97, 96, 96, 96, 96, 95, 95, 95, 94, 94, 94, 93, 93, 93, 92, 92, 92, 91, 91, 90, 90, 89, 89, 89, 88, 88, 87, 87, 86, 86, 85, 85, 84, 84, 83, 82, 82, 81, 81, 80, 80, 79, 78, 78, 77, 76, 76, 75, 74, 74, 73, 72, 72, 71, 70, 69, 69, 68, 67, 66, 66, 65, 64, 63, 62, 62, 61, 60, 59, 58, 57, 56, 56, 55, 54, 53, 52, 51, 50, 49, 48, 47, 46, 45, 44, 43, 42, 41, 40, 39, 38, 37, 36, 35, 34, 33, 31, 30, 29, 28, 27, 26, 25, 23, 22, 21, 20, 19, 17, 16, 15, 14, 12, 11, 10, 9, 7, 6, 5, 3, 2, 1 } ;

// uch_spo2_table is calculated by the formula [4].

B. Alarm System

When the user's SpO2 value is lower than alert value (for example, we set the value 94), the LED of port#7 will be stimulated and flicker. The code of this part is shown in Figure 8.

```
Serial.print(F(", SPO2="));

Serial.print(n_spo2, DEC);
  if(n_spo2 <= 94 and n_spo2 > 0){
    digitalWrite(7, HIGH);
    delay(10);
    digitalWrite(7, LOW);
    delay(10);
    }
Serial.print(F(", SPO2Valid="));
Serial.println(ch_spo2_valid, DEC);
}
```

Fig.8. Code of Alarm System

V. PROTOTYPE

A. Website

Our website actual renderings are shown below:

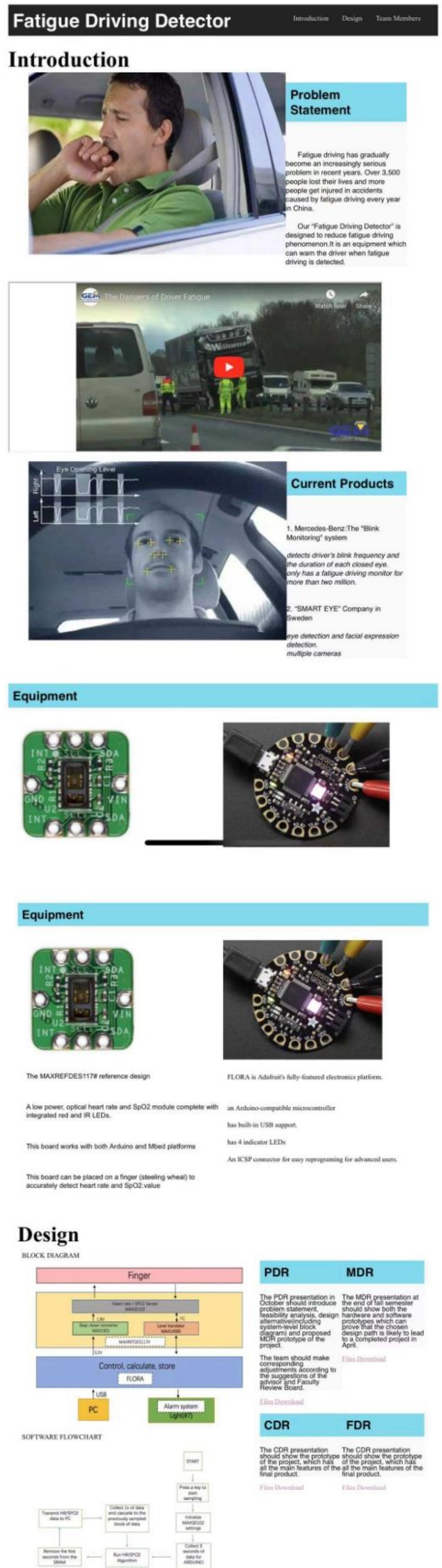


Fig.9. Website page

B. Car equipment design:

For the combination of components, a car equipment is the best choice for practical use. Our current thinking is to place the SpO2 sensor on one side of the steering wheel. In this case, the driver can measure his SpO2 whenever he/she wants, but driver's finger needs to be attached to the board on the steering wheel all the time. The Adafruit FLORA should be loaded into the car body in our design. The concept picture of car equipment version fatigue driving detector is shown in Figure 11.

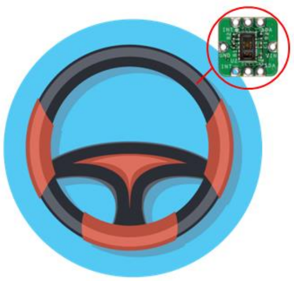
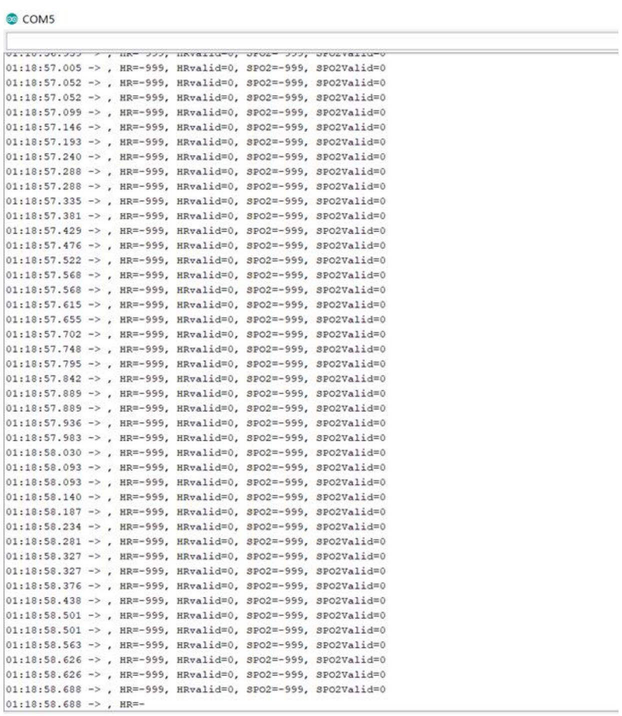


Fig.10. Car equipment design [6]

VI.Result

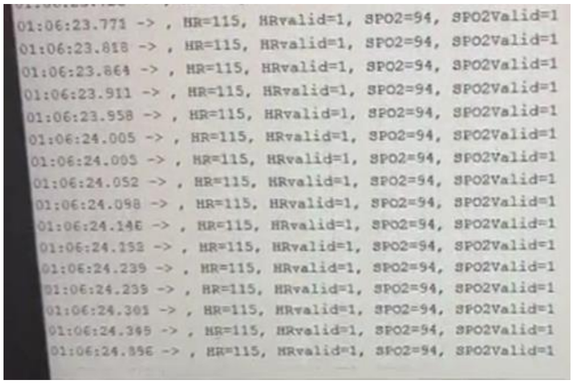
The result of the SpO2 system has 4 parts, which are HR, HRvalid, SPO2, SPO2Valid. The first column HR is heart rate value, the third column SPO2 is SpO2 value, and the second and forth column show the validities of heart rate and SpO2. Fig.12 shows the result when tester's finger is not attached to the sensor, HRvalid and SPO2Valid is 0, which means heart rate value and SpO2 value is unavailable, and the value is -999.



```
COM5
01:18:57.005 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.052 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.052 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.099 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.146 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.193 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.240 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.288 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.288 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.335 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.381 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.429 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.476 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.522 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.568 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.568 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.615 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.655 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.702 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.748 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.795 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.842 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.889 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.889 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.936 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:57.983 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:58.030 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:58.093 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:58.093 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:58.140 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:58.187 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:58.234 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:58.281 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:58.327 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:58.327 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:58.376 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:58.438 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:58.501 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:58.501 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:58.563 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:58.626 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:58.626 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:58.688 -> , HR=-999, HRvalid=0, SPO2=-999, SPO2Valid=0
01:18:58.688 -> , HR=-
```

Fig.11. The SpO_2 readings

In the final test, we got the ideal result. When the tester (Yachen Liu) is fatigued (did not sleep for a long time) and put his finger on the SpO_2 sensor, the LED of port#7 flickered.



```
01:06:23.771 -> , HR=115, HRvalid=1, SPO2=94, SPO2Valid=1
01:06:23.818 -> , HR=115, HRvalid=1, SPO2=94, SPO2Valid=1
01:06:23.864 -> , HR=115, HRvalid=1, SPO2=94, SPO2Valid=1
01:06:23.911 -> , HR=115, HRvalid=1, SPO2=94, SPO2Valid=1
01:06:23.958 -> , HR=115, HRvalid=1, SPO2=94, SPO2Valid=1
01:06:24.005 -> , HR=115, HRvalid=1, SPO2=94, SPO2Valid=1
01:06:24.005 -> , HR=115, HRvalid=1, SPO2=94, SPO2Valid=1
01:06:24.052 -> , HR=115, HRvalid=1, SPO2=94, SPO2Valid=1
01:06:24.098 -> , HR=115, HRvalid=1, SPO2=94, SPO2Valid=1
01:06:24.146 -> , HR=115, HRvalid=1, SPO2=94, SPO2Valid=1
01:06:24.192 -> , HR=115, HRvalid=1, SPO2=94, SPO2Valid=1
01:06:24.239 -> , HR=115, HRvalid=1, SPO2=94, SPO2Valid=1
01:06:24.239 -> , HR=115, HRvalid=1, SPO2=94, SPO2Valid=1
01:06:24.301 -> , HR=115, HRvalid=1, SPO2=94, SPO2Valid=1
01:06:24.349 -> , HR=115, HRvalid=1, SPO2=94, SPO2Valid=1
01:06:24.396 -> , HR=115, HRvalid=1, SPO2=94, SPO2Valid=1
```

Fig.12. The SpO_2 readings

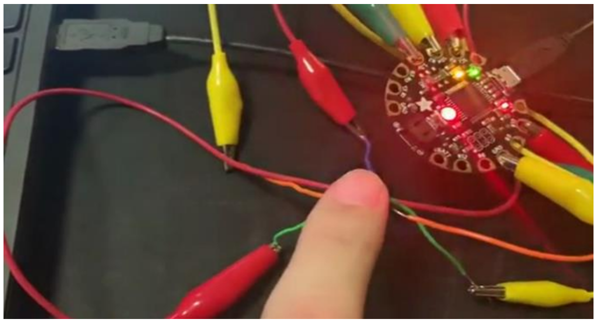


Fig.13. Testing processing

VII.Conclusion

In this project, we designed and created a demo that can test and show user's peripheral capillary oxygen saturation. Through value of user's SpO_2, this demo determines whether the user is in a fatigue state or not. If the user is tired, it can alert him/her. We think that what we have done in this semester has been very successful. We believe that we have basically completed what we planned at the beginning of this project. As long as it goes according to plan, we believe we can make a nice product at the end of next semester.

VIII.Project management

We established a division of labor at the beginning of the semester, it is shown below. We met our advisor, Prof. Arman Pouraghily every week to confirm the progress and feedback issues. Prof. Pouraghily gave us many helpful suggestions throughout the whole process. We spent two weeks looking for suitable SpO2 sensors and control boards, bought them, and waited for delivery. During the circuit welding work, we researched the ports and welded the ports of the sensors to the corresponding ports of the control board, but the sensor board was damaged due to welding errors, so we repurchased the board and soldered again.

DIVISION OF LABOR

Jiong Wang	Team leader/ hardware engineer	Contacting with the advisor, hardware designing, circuit welding and plan making
Hongyu Ba	Hardware engineer	Hardware designing, circuit welding and prototype building up
Yachen Liu	Software engineer	Algorithm researching, coding and debugging

Here are our rough plans for the next semester:

- Add a stable device to the SpO_2 sensor to make the data measurement more accurate;
- Do more reaction time and SpO_2 experiments;
- Build up a prototype of final produce, a fatigue driving detector which can be used in a vehicle.

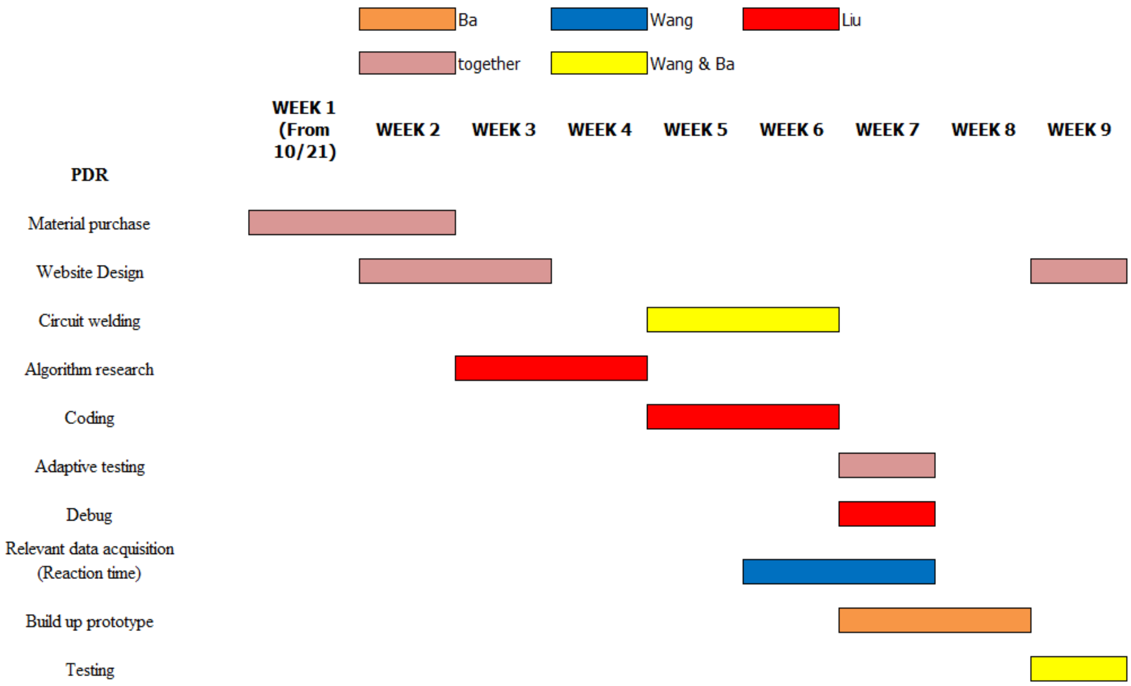


Fig.14. Gantt Chart

References

[1] National Safety Council.(2019). [online] Available at: https://www.nsc.org/road-safety/safety-topics/fatigued-driving [Accessed 18 Dec. 2019].

[2] G Coley, A Wesley, N Reed and I Parry. (2008). *Driver reaction times to familiar but unexpected events*. [online] Available at: https://trl.co.uk/sites/default/files/PPR313_new.pdf [Accessed 18 Dec. 2019].

[3] Zengyong LI, Ming Zhang, Xiaoyin Zhang, Shixun Dai, Xingxin Yu and Yan Wang(2009). *Assessment of cerebral oxygenation during prolonged simulated driving using near infrared spectroscopy: its implications for fatigue development.* [online]Available at: https://www.researchgate.net/publication/26648775_Assessment_of_cerebral_oxygenation_during_prolonged_simulated_driving_using_near_infrared_spectroscopy_Its_implications_for_fatigue_development [Accessed 18 Dec. 2009]

[4] Maxim intergrated.(2019). [online] Available at: https://www.maximintegrated.com/en.html [Accessed 18 Dec. 2019].

[5] Adafruit. (2019). [online] Available at: https://www.adafruit.com/ [Accessed 18 Dec. 2019].

[6] 90 design .(2019).[online] Available at: http://90sheji.com/[Accessed 18 Dec. 2019].