2024

Intelligent System for Passenger Seat in Vessels – I4ET

ICPS, Arduino Cloud

GROUP 4

MUHAMMAD OMER FAROOQ ZAHID

MILAD JALILZADEH

URFAN AGHAYEV

1 Report of ICP and Dashboard

The suspension system in seating for humans on ships is designed to enhance comfort and safety by mitigating the effects of the ship's movements on its occupants. This system is particularly important in rough seas, where waves and swells can cause significant motion, potentially leading to discomfort, fatigue, or even injury for passengers and crew. Seats are designed to provide support to the spine, neck, and lumbar regions. High-quality padding and contoured shapes help maintain posture and distribute weight evenly.

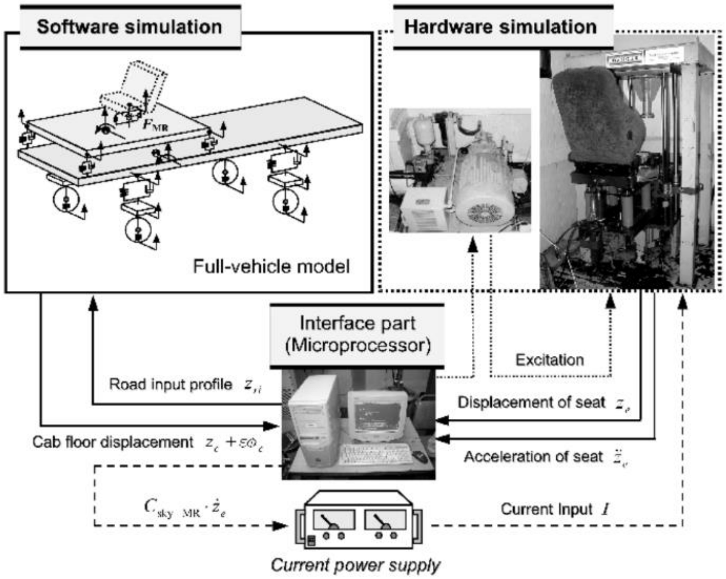
1.1 Benefits of a Good Suspension System in Ship Seats

- Enhanced Comfort: Reduces fatigue by minimizing the jarring effects of the ship's movement.
- Improved Safety: Prevents injuries by cushioning the impact of sudden motions.
- Better Focus and Performance: Essential for crew members who need to remain alert and effective, especially in rough conditions.
- Increased Endurance: Passengers and crew can endure longer periods at sea without discomfort, which is crucial for long voyages.

1.2 Application Areas

- Crew Seats: For navigation officers, engineers, and other essential personnel who need to stay alert and functional.
- Passenger Seats: Especially in high-speed ferries or cruise ships where passenger comfort is a priority.
- Specialized Seats: For medical personnel, technicians, or others who may need to perform precise tasks even in rough conditions.

In below picture we can see an active suspension system that is getting the results from two sensors and sending them to the computer for analysis, after that with some calculation and simulation of ship movement, the actuator signal sends to actuator device to decrease the vibration of the upper part of the seat and the passenger feel comforting.


Software simulation
Hardware simulation
F_{MR}
Full-vehicle model
Interface part (Microprocessor)
Road input profile z_{ri}
Excitation
Displacement of seat z_e
Cab floor displacement $z_c + s\phi_c$
Acceleration of seat $\ddot{z}_e$
$C_{sky\_MR} \cdot \dot{z}_e$
Current Input I
Current power supply

Now we start our exploring according some pervious investigation, and designing our system. In our system, two accelerometers get the vibration of ship and upper part of the seat and sent them to the CPU of Arduino, with coding this device according some control method, some feedback send to our actuators that here are buzzer and RGB light. Furthermore, we can see the temperature and humidity and data of acceleration in Real time world in Dashboard.

2 Thing number 1

Now the whole code of the two things starting to explaining, in the first code we have accelerometer and RGB light to present our intelligent system.

```
#include "thingProperties.h"
#include "Wire.h"
#include <MPU6050_light.h>
MPU6050 mpu(Wire);
int redpin=D8;
int bluepin=D6;
int greenpin=D7;
```

Here we define library of things and wiring and also define the library for acceleration and in addition introducing the port for RGB light.

```
initProperties();
ArduinoCloud.begin(ArduinoIoTPreferredConnection);
setDebugMessageLevel(2);
ArduinoCloud.printDebugInfo();
mpu.calcOffsets();
```

In previous code we connect to Arduino IoT Cloud and following function allows us to obtain more information related to the state of network and IoT Cloud connection and errors the higher number the more granular information you'll get. The default is 0 (only errors), and also, we calibrate our accelerometer sensor.

```
void loop() {
  ArduinoCloud.update();
  mpu.update();
  two = (mpu.getAccZ())-1;

  if(two<1.5)
{
  analogWrite(bluepin, 255);
  delay(1000);
  analogWrite(bluepin, 0);
  delay(1000);
}
```

```
if(3>two&&two>1.5)
{
  analogWrite(greenpin, 255);
  delay(1000);
  analogWrite(greenpin, 0);
  delay(1000);
}
if(two>3)
{
  analogWrite(redpin, 255);
  delay(1000);
  analogWrite(redpin, 0);
  delay(1000);
```

In former line we coding in order in updating the cloud and new data of sensor and then with data of accelerometer we start to do control part of our project. If the figures of acceleration go beyond specific number our RGB light show different alarm. If the figure of accelerometer is lower than one the Blue light turn on, if the accelerometer is between one and two the Green light turn on and finally if the accelerometer goes upper than 2, the Red light turn on.

3 Thing number 2

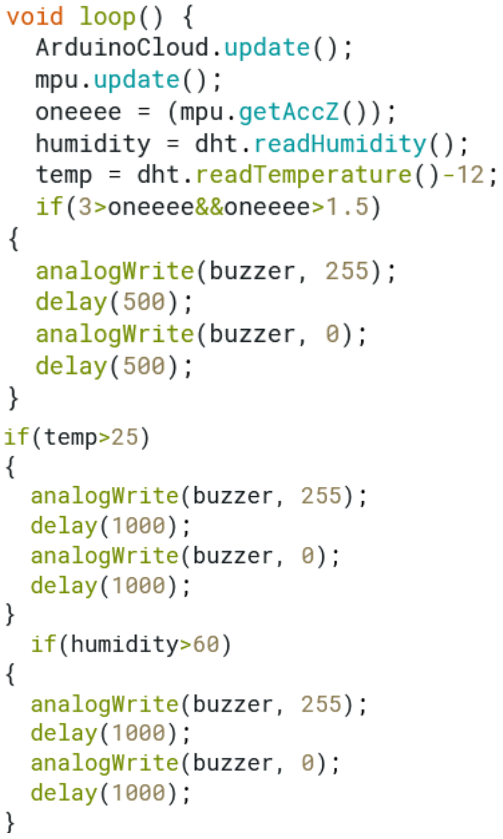
Now it is time to explain the code inside the Thing number 2 in cloud.

```
#include "thingProperties.h"
#include "Wire.h"
#include <MPU6050_light.h>
#include "DHT.h"
#define DHTTYPE DHT11
#define dht_dpin D8
#define buzzer D7
DHT dht(dht_dpin, DHTTYPE);
MPU6050 mpu(Wire);
```

In this part, prepare some library in order to use in whole project. We call wire library and MPU6050 in order to can use data that produce by accelerometer. In addition, we define some port to use for buzzer and temperature sensor and humidity sensors.

```
delay(1500);
Wire.begin();
byte status = mpu.begin();
while(status != 0) { }
initProperties();
ArduinoCloud.begin(ArduinoIoTPreferredConnection);
setDebugMessageLevel(2);
ArduinoCloud.printDebugInfo();
mpu.calcOffsets();
dht.begin();
pinMode( buzzer , OUTPUT);
```

In setup part, there is delay, this delay gives the chance to wait for a Serial Monitor without blocking if none is found. Then the cloud waiting for getting the data from accelerometer and then in while checking the status of the data, if this is equal to zero next line start. At next line Arduino cloud start to connect and then code start to reading the temperature sensors. The following function allows us to obtain more information related to the state of network and IoT Cloud connection and errors the higher number the more granular information you'll get. The default is 0 (only errors). And at the end line we define the one port to connect to our buzzer.

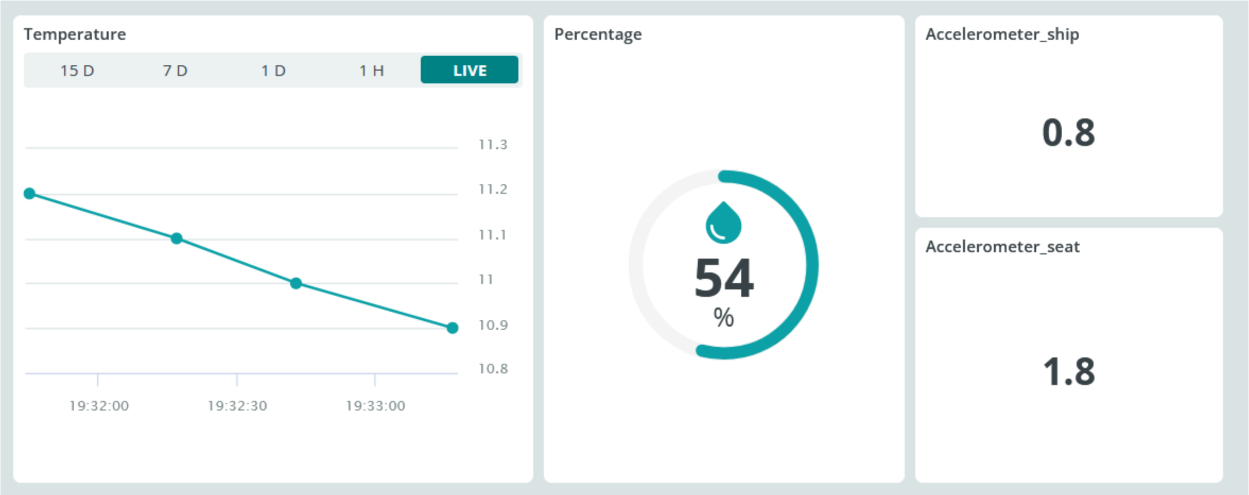


```
void loop() {
  ArduinoCloud.update();
  mpu.update();
  oneeee = (mpu.getAccZ());
  humidity = dht.readHumidity();
  temp = dht.readTemperature()-12;
  if(3>oneeee&&oneeee>1.5)
{
  analogWrite(buzzer, 255);
  delay(500);
  analogWrite(buzzer, 0);
  delay(500);
}
if(temp>25)
{
  analogWrite(buzzer, 255);
  delay(1000);
  analogWrite(buzzer, 0);
  delay(1000);
}
  if(humidity>60)
{
  analogWrite(buzzer, 255);
  delay(1000);
  analogWrite(buzzer, 0);
  delay(1000);
}
```

In next line we update our cloud with Arduino and getting each time the data for acceleration and humidity and temperature sensors. Then we start to compare the results with our criteria, if it didn't pass the criteria, the buzzer starts to give the alarms.

In summary, the suspension system in ship seats plays a vital role in enhancing the overall experience on board by improving comfort and safety through advanced engineering solutions that mitigate the impact of the sea's motion.

4 Dashboard



Percentage

54 %

Accelerometer_ship

0.8

Accelerometer_seat

1.8