A prototype of a system for detecting G force in an automobile, based on two ESP32's

Ivan Gutai
Chair of Electrical Measurement
Faculty of Technical Sciences
Novi Sad, Serbia
gutai@uns.ac.rs

Prof. dr Platon Sovilj
Chair of Electrical Measurement
Faculty of Technical Sciences
Novi Sad, Serbia
platon@uns.ac.rs

Marina Subotin
Chair of Electrical Measurement
Faculty of Technical Sciences
Novi Sad, Serbia
marina.bulat@uns.ac.rs

Đorđe Novaković
Chair of Electrical Measurement
Faculty of Technical Sciences
Novi Sad, Serbia
djordjenovakovic@uns.ac.rs

Nemanja Gazivoda
Chair of Electrical Measurement
Faculty of Technical Sciences
Novi Sad, Serbia
nemanjagazivoda@uns.ac.rs

Bojan Vujičić
Chair of Electrical Measurement
Faculty of Technical Sciences
Novi Sad, Serbia
bojanvuj@uns.ac.rs

***Abstract*—A prototype gives the automobile driver option to read current and maximal G force in all three axes and to read the angle at which the vehicle is. The system consists of two parts. The first part is visible to the driver, and it consists of ESP32 and TFT display. The second part is in the trunk, and it consists of ESP32 and sensor MPU9250. 5 V power supply powers both units. Modules communicate over Wi-Fi, using ESP-NOW technology. While driving through corners, all passengers can get an interactive view of G force. The system uses encrypted communication. The price of hardware can go to about 20 $, which makes it suitable for educational purposes. Also, this fits for the training of engineers in developing countries.**

Keywords—ESP32 WEMOS D1 R32, MPU9250, Wi-Fi, ESP-NOW, encrypted communication, graphical user interface, 3-axis G-force meter

I. Introduction

Open-hardware continues to enable fast-paced prototype development and a more agile approach to development and testing. In a current hardware configuration, It provides a choice for the user to see the strength of the G force in the 3-axis while driving. The main driver's goal is to be concentrated on driving and road conditions, and the graphical user interface is created not to distract. If you ever wondered to what amount of force are our bodies exposed to, while driving an automobile or facing a bump in a road, this prototype gives a lot of answers. One segment of the screen is dedicated to the interactive G force meter shown as a graph. The prototype consists of two parts. The first part is visible to the driver, and it consists of ESP32 and TFT 2.8 inch display. The second part is for acquisition, and it consists of ESP32 WEMOS D1 R32 and sensor TDK InvenSense MPU9250 [2]. In this paper, the accent is on modularity and ease of wireless data transfer using Espressif's ESP-NOW [3] technology, with encryption possibilities. The concept of dividing into smaller parts comes from programming and represents good practice.

II. Used hardware

A. Dashboard component

The first part, consisting of ESP32 and TFT 2.8 inch display resolution 320 px with 240 px is powered by 5V and is in line with the dashboard. Connection indicatior is located in the top right corner of the screen and is realized via a green or red rectangle. For relatively small distance which Wi-Fi cross, it is unnecessary to use an external antenna for ESP32, copper integrated antenna works like a charm.

B. Trunk component

A variant of ESP32 called WEMOS D1 R32 is selected because it is physically identical to the Arduino Uno development system. It means that there are plenty of prototype shields or acrylic cases. Users can choose between proprietary or open-hardware variants.

If they chose a proprietary variant like MikroElektronika Arduino Uno Click Shield, it can fit into ESP32 called WEMOS D1 R32 development system. The device is fixed in the trunk for simplicity and doing it initially with a lot of effort. We need to be sure that the device is perfectly aligned with the surface. The device is 5 V battery-powered. Wi-Fi channel 14 is in use because it is generally rarely used, and there is less interference. ESP-NOW gives the possibility to send data to up to 20 devices over WI-Fi. If we decide to use encryption, we must consider that LMK (Local Master Key) matches on receiver and transmitter. On the receiver, receiving of data must be enabled, and the receiving data structure must match the transmitting data structure.

III. Prototype features

The display shows current and extreme values of G force for X, Y, and Z axes. The user doesn't have to think about the labeling of the axes, and on-screen, just before numerical values first letters are written of: front, back, left, right, top, and bottom. In front of current values, there are no letters. Acquired parameters from the accelerometer make data for calculation of angles in which is the vehicle in every moment. Formulas (1) and (2) shows wheater the vehicle is on incline, decline, or is tilted. Terms pitch and roll are aviation terms but make it easier to understand the concept.

mpu_roll = (-atan2(accelY, accelZ) + PI) * radian; (1)
mpu_pitch = (-atan2(accelX, accelZ) + PI) * radian; (2)

On display, there was a space for dynamically drawing a graph. It has 100 points and its content moves vertically. On top is located current value, and preceding values are drawn below. When a vector is filled with the 100th value, the oldest value is removed, which creates a spot for the latest value. During the development of the prototype, the refresh rate of display between 40 ms and 2 s is tested. For the city driving environment, it is enough for the chart to refresh every second. Every second, 100 measurements are made. It enables enough precision for using the prototype in everyday driving, also suitable for unexpected situations. Data transfer from one to another part takes place 30 times per second. Every second vector content is drawn in form of dots. Dots are interconnected, which enables the user to see the content of the graph to move down. During regular city driving, G forces in-vehicle go up to 0.8 G. Graph is limited for drawing forces up to 1 G, but extreme values up to 16 G can be registered, which is maximum for the MPU9250 sensor. Driving over a railroad crossing with a speed of 30 km/h 1.5 G is registered from the bottom side of the automobile.

Fig 1. and Fig 2. are a physical representation of hardware used for building the system.

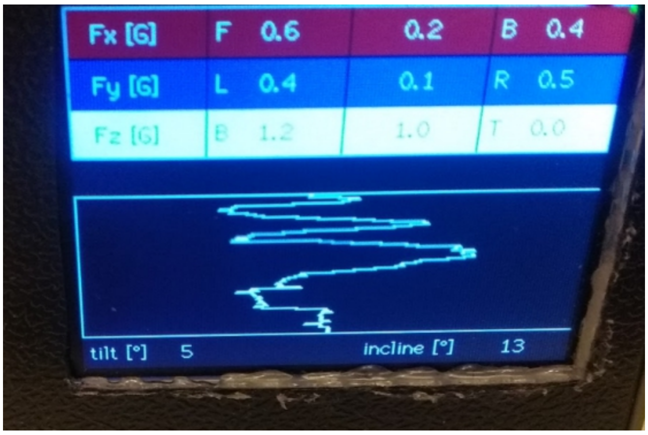

FIGURE 1 USER INTERFACE

FIGURE 2 ACQUISITION PART, OPEN-HARDWARE SOLUTION ON THE TOP, PROPRIETARY SOLUTION ON THE BOTTOM

The display is connected via a hardware SPI (*Serial Peripheral Interface*) port and uses TFT_eSPI [4] firmware. ESP32 has two hardware SPI ports, and most of the pins can be mapped as a software SPI. In this implementation software SPI, can't accomplish an adequate refresh rate nor avoid flickering. MPU9250 uses firmware [5]. Elements of FreeRTOS (Free Real-Time Operating System) are used for boosting performance, which already was present due to the powerful Tensilica Xtensa LX6 processor clocked at 240 MHz. Indication if the second part is connected and drawing graphical user interface is done with 2nd core.

IV. KNOW HOW

Some things begin to be clear right after intensive testing, and by the way, they are changed in the firmware. This makes the system both affordable and practical for white box testing. In development, there are numerous paths to the right solution. As in the choice of the data structure, the vector is chosen for its advances. Values can be dynamically inserted, pushed out, or can be reversed. Combined, that makes a perfect fit for dynamically drawing data on a graph. Fig 3. depicts an algorithm for dynamically drawing G force.

```
int g3 = 0;
float elementTemp = 0;
tft.drawRect(0, 114, 320, NUMBER_OF_G_DOTS + 2, TFT_WHITE);
for (auto & element2 : accelerationData2) {
  if (g3 < NUMBER_OF_G_DOTS) {
    elementTemp = element2;
    tft.drawLine(2, 115 + g3, 159, 115 + g3, TFT_BLACK);
    tft.drawLine(318, 115 + g3, 159, 115 + g3, TFT_BLACK);
    if (accelerationData2.size() > g3 + 2) {
      tft.drawLine(160 - accelerationData2[g3] * 160, 115 + g3,
      160 - accelerationData2[g3 + 1] * 160, 115 + g3, TFT_WHITE);
    }
    tft.fillCircle(160 - element2 * 160, 114 + ++g3, 1, TFT_WHITE);
  }
}
```

FIGURE 3 ALGORITHM FOR INTERACTIVE G FORCE

Using MPU9250 sensor enables acquisition from 3 axes of data from accelerometer, gyroscope, or magnetometer. Detailed and precise instructions that can provide a bigger picture of this topic are documented [6] and [7]. Detailed instructions for ESP-NOW technology, with examples, are documented [8].

Dealing with open hardware requires owning basic "electrician" tools, and skills for using it. Using ESP32 WEMOS D1 R32 can provide tool-free assembly because it has the same shape as an Arduino Uno. All you need is a modified MikroElektronika Arduino Uno Click Shield. Connecting pins 22 and 21 from ESP to A5 and A4 on MikroElektronika Arduino Uno Click Shield is a modification. After that, it can be used for connecting MikroElektronika Click Boards which operates on I2C. Click Board labeled as MPU 9DOF CLICK [9], based on MPU9250 chip, can be used. There is a significant price difference, but the MikroElektronika option is well tested and it has plenty of documentation. When you buy MPU9250 yourself, the price is smaller, but be prepared to do a lot of testing. Using the display in MPU9250 based system ensures us during fitting, that the system is perfectly aligned. In that scenario, the accelerometer acquires values 0 on the X and Y axes, and value 1 on Z-axis.

If someone wants to test programming skills and to delve into the creation of the user interface, ESP32 and TFT are needed. In this case, TFT with ILI 9341 chip is used. Before connecting, file User_Setup.h [4] needs to be modified: TFT_MISO 19, TFT_MOSI 23, TFT_SCLK 18, TFT_CS 5, TFT_DC 4, TFT_RST 2. After that, LED and VCC pin are connected to 3.3 V, and ground to GND. Creation of UI (User Interface) can start, and doesn't to be pricy.

Recommendation for everyone who is entering in 2021 the world of embedded programming is one programming book [10]. In it, there are applied Microlearning techniques, and on the creative way is explained first 20 % needed for understanding complex algorithms.

V. CONCLUSION

This paper represents the proof of the concept, that it is possible to make a multi-segment device that can be used in an automobile, and the price of hardware is about 20 $. Espressif ESP32 has a lot of functionalities, but in this paper, the authors tried their best to pinpoint the most significant ones. 2021 is the year when people who want to enter the world of embedded programming have numerous options in various price ranges. ESP32 stands out for the invested: gained ratio. If we want to simplify the prototype, it can be assembled from ESP32, TFT ILI9341, MPU9250 sensor, and one micro USB cable. In that case, the price of hardware is divided by number two. The code name for the used combination of hardware is: "Affordable DEV kit EE". DEV as a developer, and EE as an emerging economy.

REFERENCES

[1] ESP-IDF Programming Guide https://docs.espressif.com/projects/esp-idf/en/latest/esp32/

[2] TDK InvenSense MPU-9250 Nine-Axis (Gyro + Accelerometer + Compass) MEMS MotionTracking™ Device https://invensense.tdk.com/products/motion-tracking/9-axis/mpu-9250/

[3] ESP-NOW https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/network/esp_now.html

[4] https://github.com/Bodmer/TFT_eSPI

[5] https://github.com/bolderflight/MPU9250

[6] AN2272 PSoC 1 Sensing - Magnetic Compass with Tilt Compensation https://www.cypress.com/file/130456/download

[7] https://toptechboy.com/arduino-based-9-axis-inertial-measurement-unit-imu-based-on-bno055-sensor/

[8] https://randomnerdtutorials.com/esp-now-esp32-arduino-ide/

[9] https://www.mikroe.com/mpu-9dof-click

[10] Soleša, D., Carić, M., Ivančić, I. (2016). Uvod u programski jezik C, Novi Sad: Univerzitet Privredna Akademija, Fakultet za ekonomiju i inženjerski menadžment. ISBN 978-86-87619-79-1