HO CHI MINH UNIVERSITY OF TECHNOLOGY

OFFICE FOR INTERNATIONAL STUDY PROGRAMS

PROJECT:

THE DESIGN OF MITSUBISHI XPANDER

ACCELERATOR PEDAL POSITION SIGNAL SIMULATOR

Student's name: Âu Dương Huê

Student ID: 1752220

Class: CC18OTO

Instructor: Ph.D. Trần Đăng Long

Ho Chi Minh City, 2021

VIETNAM NATIONAL UNIVERSITY OF HO CHI MINH CITY
HO CHI MINH UNIVERSITY OF TECHNOLOGY
Faculty of Transportation Engineering
Department of Automotive Engineering

SOCIALIST REPUBLIC OF VIETNAM
Independence – Freedom – Happiness

THESIS MISSION

1. **Student's name:** Âu Dương Huê **- Student ID:** 1752220
2. **Major:** Automotive Engineering **- Class:** CC18OTO
3. **Thesis title:** The design of Mitsubishi Xpander accelerator pedal position signal simulator.
4. **Content:**
   - ❖ General information about Mitsubishi Xpander accelerator pedal signals:
   - ❖ Signals working principle.
   - ❖ Actual measurement of voltage signals on Mitsubishi Xpander.
   - ❖ Identify the problem to be solved.
   - ❖ General layout design:
     - Design the structure diagram.
     - Design the principle diagram.
   - ❖ Technical design:
     - Select the components for the product.
     - Design the timing diagram.
     - Design algorithm flowcharts.
     - Arduino programming:
       - ➢ Timer control.
       - ➢ PWM control.
       - ➢ Analog to Digital Converter control.
   - ❖ Result:
     - Simulate the working process of the system using Proteus software.
5. **Product:**

✓ Presentation report.	✓ Poster.
✓ Microcontroller program.	✓ Proteus simulation.

6. **Assigned day:** April, 2021.
7. **Finished day:** June, 2021.

The content and requirements of the thesis is already approved by the Head of Department of Automotive Engineering.

HCMC, day……. month…… year 2021 *HCMC, day……. month…… year 2021*

Head of Department **Instructor**

I. INTRODUCTION

1.1 The principle of the Mitsubishi Xpander accelerator pedal signal:

- Due to the requirements for safety and reliability of information, Mitsubishi Xpander uses 2 pedal sensor signals stations to report to ECU.
- The accelerator pedal sensor is also powered by VCC (5V), and Mass, the voltage of the 2 signal pins sensor also changes according to the pedal opening but based on Hall effect principle.
- The pedal opens 0-100% created two voltage signals follow the *figure 1.1*.

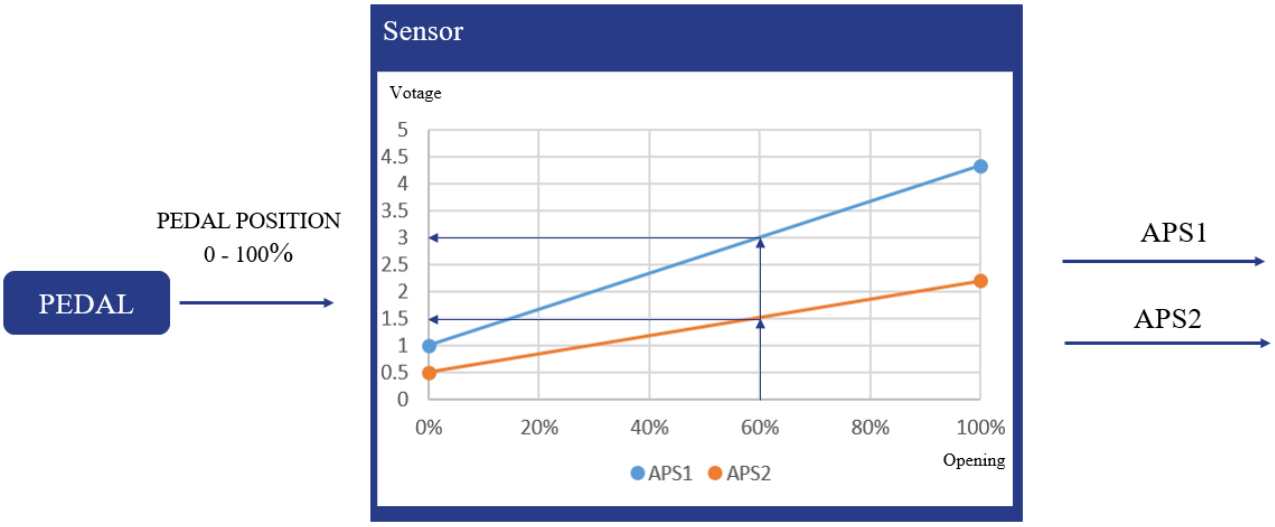

Figure 1.1: Pictorial diagram showing the principle of the Mitsubishi Xpander accelerator pedal signal.

1.2 Mitsubishi Xpander accelerator pedal signal simulator

Base on the principle the Mitsubishi Xpander accelerator pedal signal, the project is the design of Mitsubishi Xpander accelerator pedal position signal simulator (new design). The design will create two voltage signals APS1 and APS2 like the pedal signals of Mitsubishi Xpander by changing the potentiometer.

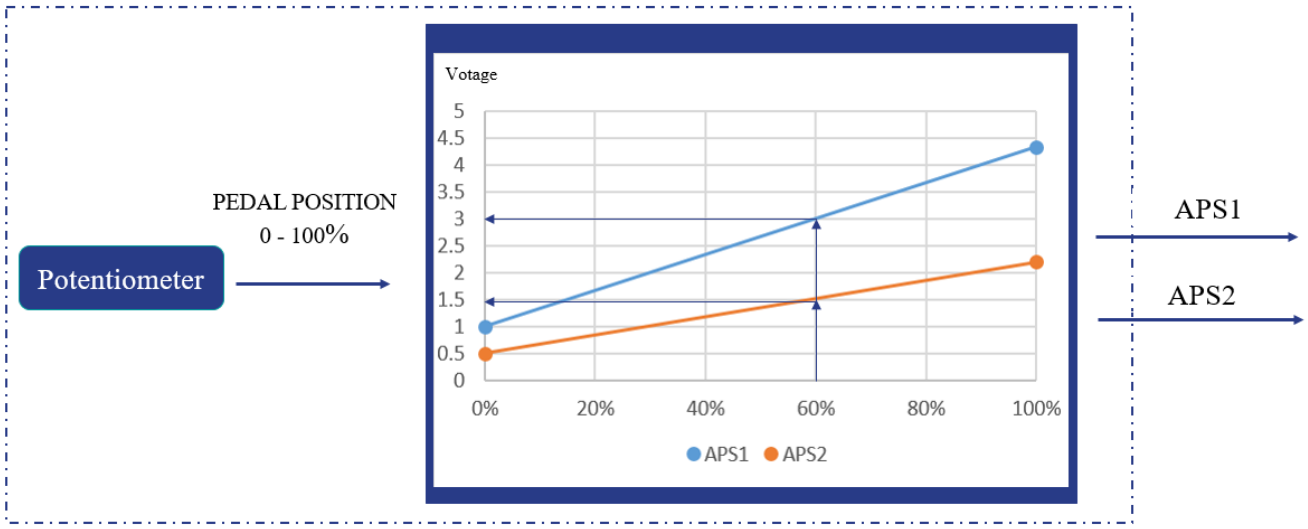

Figure 1.2: Pictorial diagram showing the principle of the Mitsubishi Xpander accelerator signal simulator.

1.3 Working conditions and technical requirements:

1.3.1 Scope of implementation

Using a system potentiometer, microcontroller and low pass filter for simulating the voltage signal from the pedal sensor to adjust the signal.

1.3.2 Working conditions:

The automotive environmental conditions for accelerator pedal sensors include:

- High temperatures.
- Vibrations and shocks.
- Exposure to water and gases.

1.3.3 Technical requirements:

The potentiometer can change the input voltage as flexibly as the signal of the accelerator pedal.

The microcontroller:

- Has at least one input port to read and process the input voltage.
- Calculate the exact output voltage signals and limit the error.
- Has two output ports to export two signal APS1 and APS2.

Low Pass Filter is used for reshaping the output signal and accepting or passing only those wanted signals.

II. GENERAL LAYOUT DESIGN

2.1 General layout diagram

This system includes 2 main part:

- Microcontroller part:
  - ✓ Read and calculate the pedal position from the simulator potentiometer on computer
  - ✓ Calculate the width of the PWM value corresponding to the value of the potentiometer
- Signal conditioning part:
  - ✓ Low Pass Filter is used for designing, modifying, reshaping or rejecting all unwanted high frequencies from the signal of Arduino and accept or pass only those signals wanted by me.
  - ✓ Two Low Pass Filters are used for 2 signals APS1 and APS2.

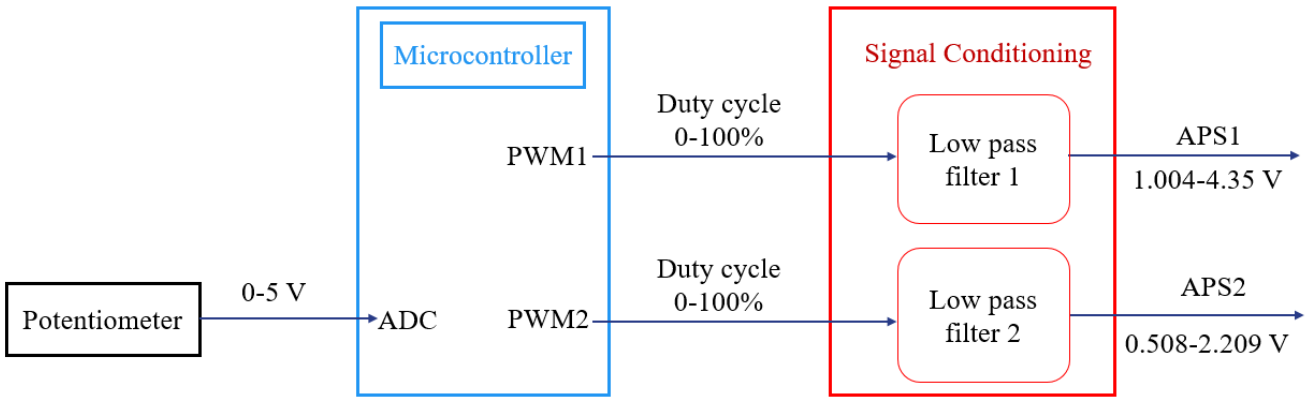

Figure 2.1: Structure diagram of accelerator pedal simulator

2.2 Working principles of the microcontroller system:

2.2.1 Input signal

- Input signal: one analog input signal – voltage value from the potentiometer which reflects the desired pedal opening value from 0-100%.

2.2.2 Output signal

- Output signal: two digital output signals – PWM pulses are generated by a microcontroller.

2.2.3 Input and output relationship

- The relationship between 2 analog input values and digital output value (PWM value) is the analog-to-digital converter (ADC).
- Depending on the input voltage, microcontroller can calculate the desired pedal position.
- From the desired value, the system calculates desired voltage of APS1 and APS2.
- Calculating duty cycle follow each desired voltage.
- Pulse width at output pin will be depended on voltage value from the potentiometer at the input A0.

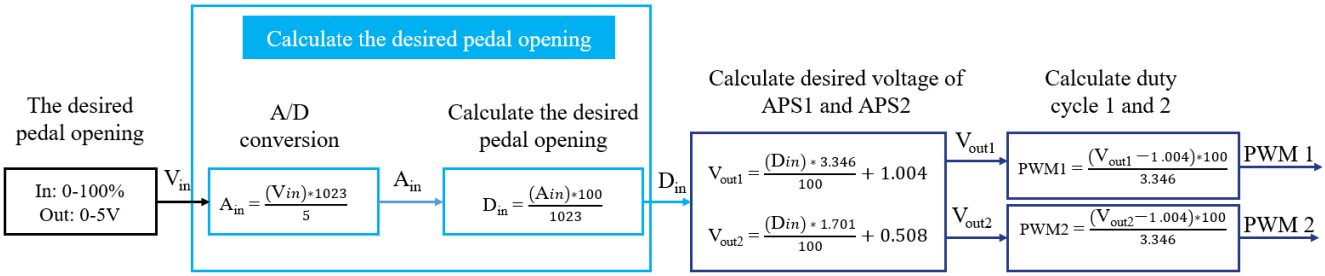

Figure 2.2: Principle diagram of accelerator pedal simulator

III. TECHNICAL DESIGN:

3.1 Choosing microcontroller board and components

3.1.1 Arduino

Arduino is a microprocessor used for programming to interact with hardware devices, designed based on open source code for both hardware and software with a common programming language (C/C++), which can be linked Combined with programs like LabVIEW, MATLAB to take advantage of the power of these programs in programming and simulation.

Advantages of Arduino:

- Simple structure, easy to use, low cost.
- Familiar C/C++ programming language.
- Working in a multi-OS environment, it can run on Windows or Mac IOS.
- Design programs can be loaded from a computer via a convenient USB cable.
- Having a diverse and rich library due to owning a large number of users. This greatly shortens the project completion time.

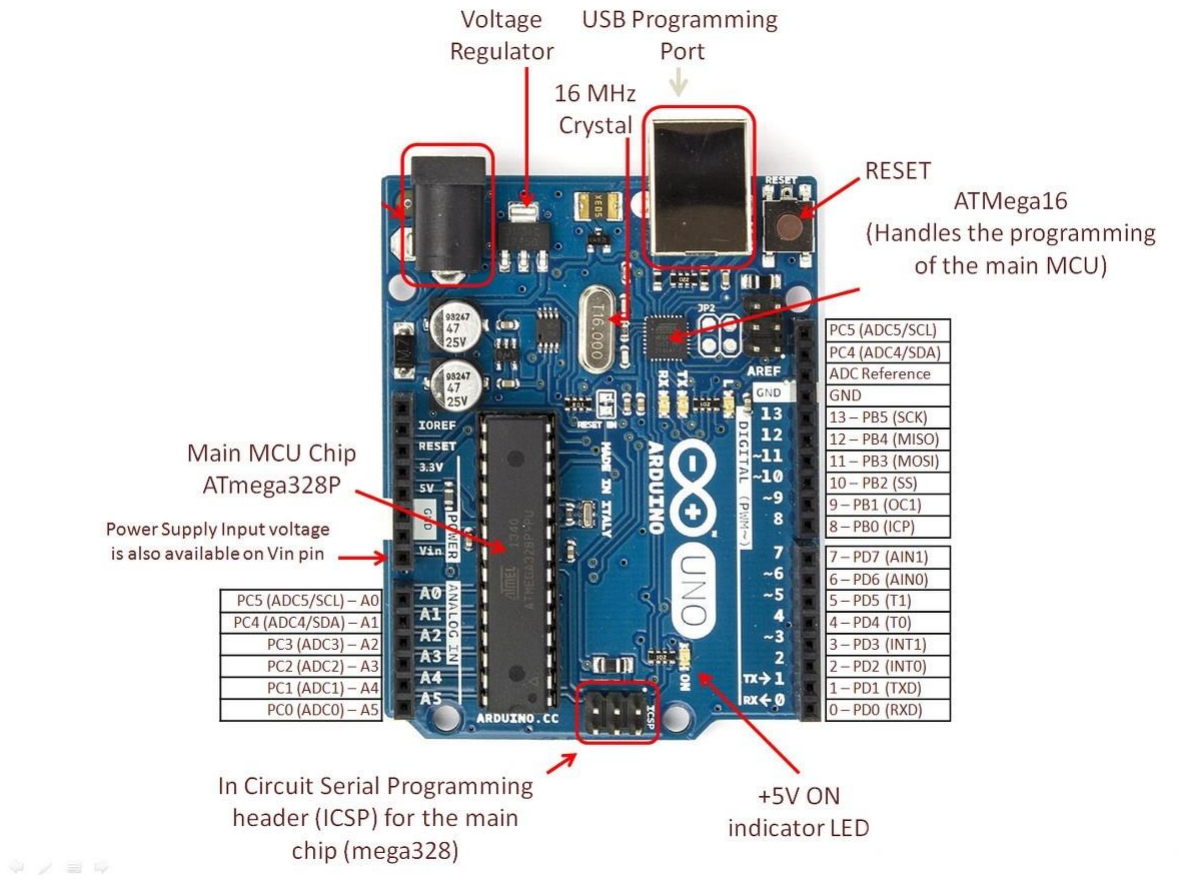

Figure 3.1: Arduino UNO R3 ATmega328p board

Choose Arduino UNO R3 to read voltage value from the simulator potentiometer on the computer and calculate the width of the PWM value corresponding to the value of the potentiometer.

Specifications:

- Microcontroller: ATmega328P
- Operating Voltage: 5V
- Digital I/O Pins: 14 (of which 6 provide PWM output).
- Analog Input Pins: 6
- Clock Speed: 16 MHz

3.1.2 Low pass filter

A low-pass filter is a filter that passes signals with a frequency lower than a selected cutoff frequency and attenuates signals with frequencies higher than the cutoff frequency. The exact frequency response of the filter depends on the filter design.

The passive filter circuit has the advantage of being very simple, but the transfer coefficient is small due to the loss on the RC, which is highly dependent on the load, and difficult to match the impedance with the multiplex circuits. The way to overcome this disadvantage is to use active filter circuits. Specifically, put an RC filter in the feedback line of the Op-Amps to increase the transfer coefficient, increase the quality factor, and reduce the effect of the load by using a buffer stage for impedance matching.

Here I use 2nd order active low pass filter to cut off the high frequencies.

The high frequencies will be shorted to ground and the amplifier will allow the low frequencies to go through it.

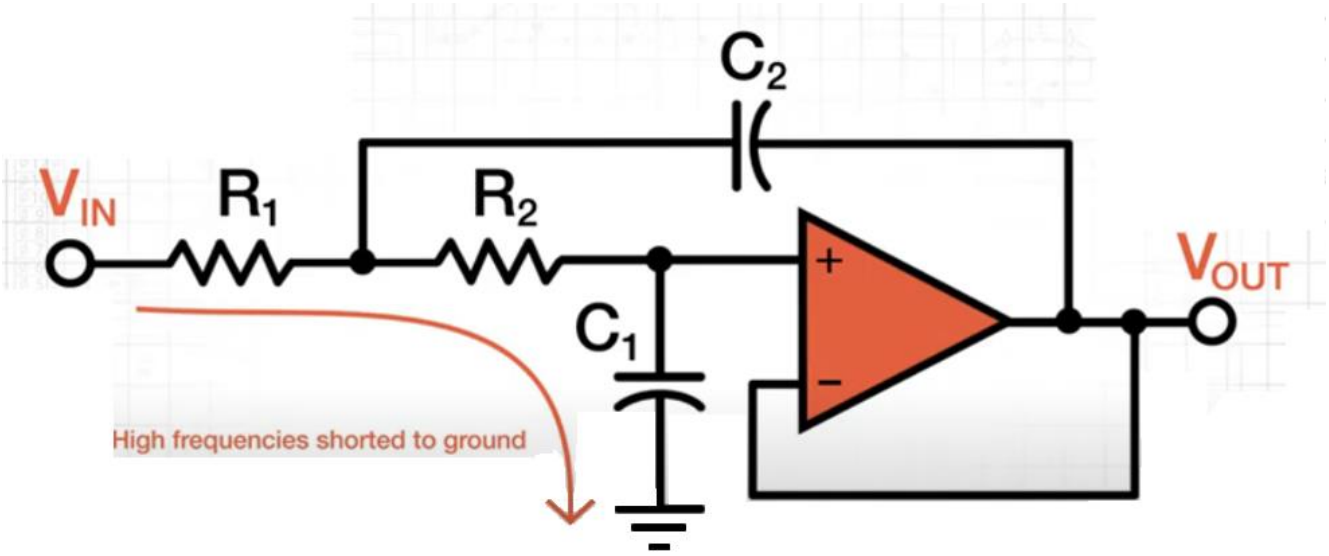

Figure 3.2: 2nd order active low pass filter

After testing, I choose R1 = R2 = 47 kΩ and C1 = C2 = 100nF will show the signal with smooth and less turbulent beside that I choose Op-07 amplifier to adapt the requirement of my simulation circuit.

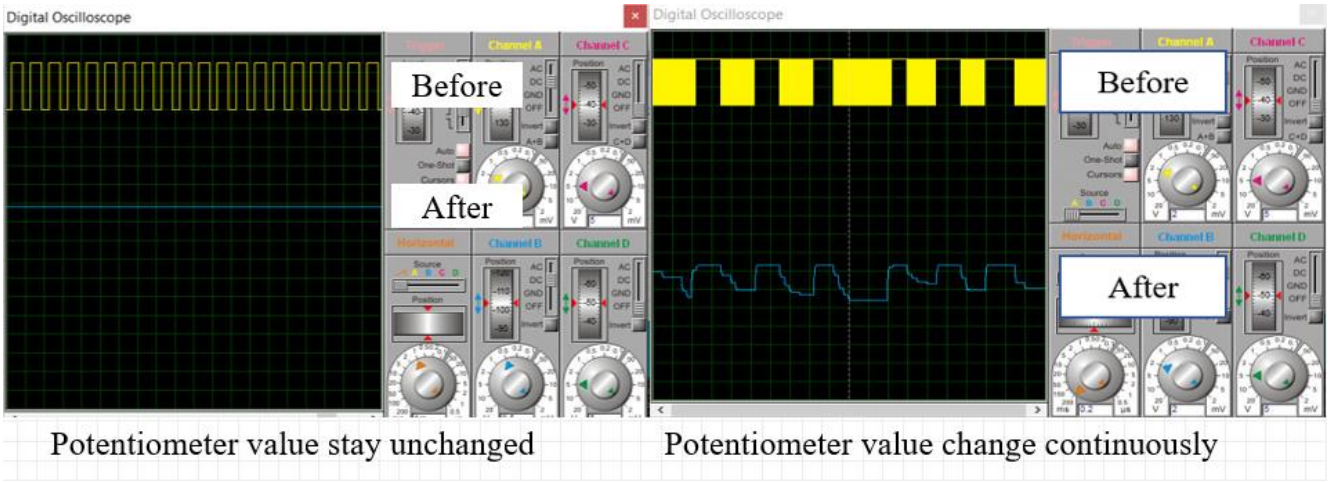

Figure 3.3: 2nd order active low pass filter testing results

3.2 Electrical scheme

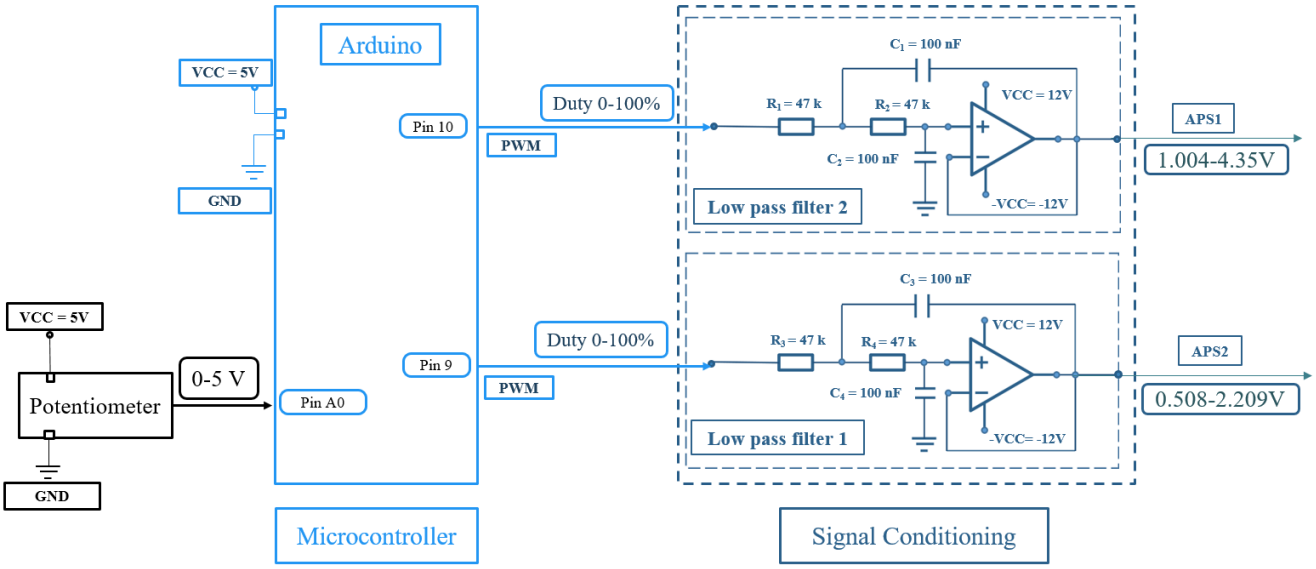

Figure 3.4: Electrical scheme

3.3 Timing diagram

Create a timer with a period of 0.1ms to manage the jobs performed in the program.

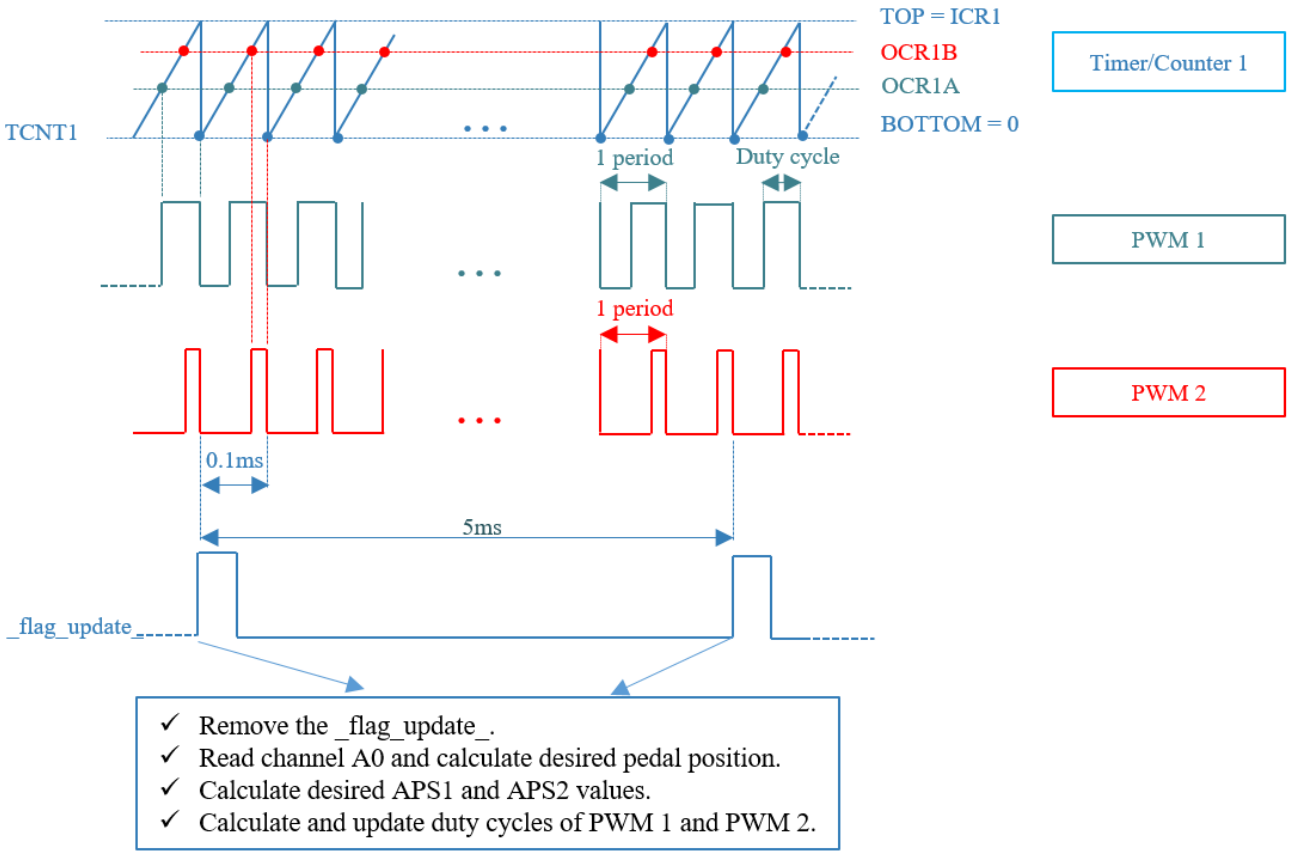

Figure 3.5: Timing diagram of the program

3.4 Algorithms

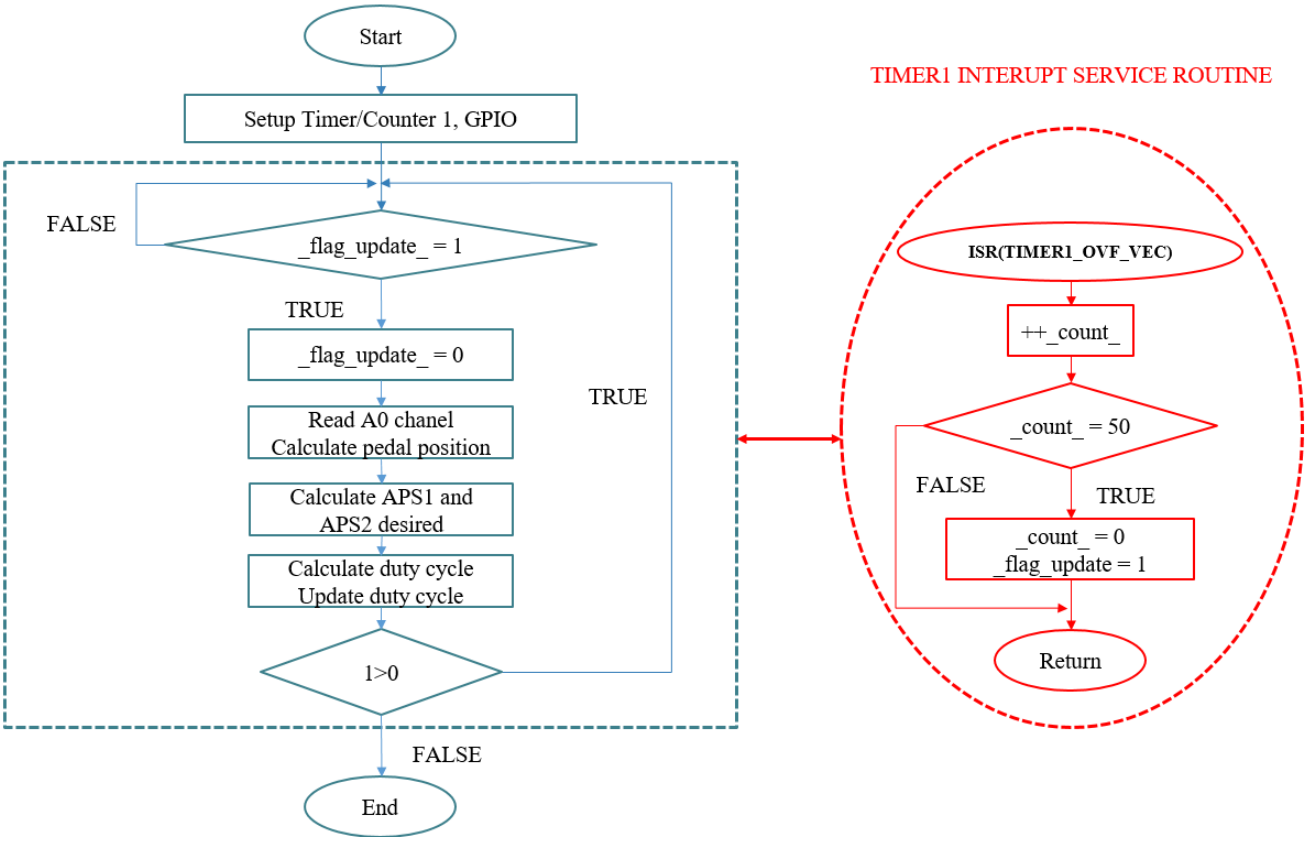

Figure 3.6: Algorithms diagram of the program

Using Timer/Counter 1 with Fast PWM mode to create a working cycle for calculating pedal position value and calculating PWM width.

Analog input pins:

- Ain0: GPIO14, 0-1023 -> 0-5V

Digital output pins

- LPF1: GPIO9, PWM1
- LPF2: GPIO10, PWM2

*** Timer1:**

- PRESCALER = 64
- Arduino frequency=16 MHz
- Timer/Counter frequency = 16MHz / 64 = 250 kHz
- Working frequency = 10kHz
- Mode: 14 - Fast PWM

+ TOP = ICR1 = 24

+ Update of OCR1A at BOTTOM

+ TOV1 Flag Set on at TOP

*** Application:**

- Read ADC value to calculate the voltage value
- Map value convert from 0-1023 to 1.001 – 4.35V
- Map value convert from 0-1023 to 0.508 - 2209V
- Calculate PWM value and assign value in OCR1A and OCR1B

*** Timer1 overflow:**

+ Set TOIE1(Timer Overflow Interrupt Enable 1): interrupt when timer overflow
+ When _count_ value reach 50 (5ms) ➔ Timer1 Overflow 50 times

==> Set _flag_update_ = 1 for communicating purpose (Serial)
Reset _count value to 0

*** Serial:**

- Display APS1 and APS2 voltage in Virtual terminal
- PWM value of APS1 and APS2 after go through LOW PASS FILTER in Digital Oscilloscope

IV. PROTEUS SIMULATION:

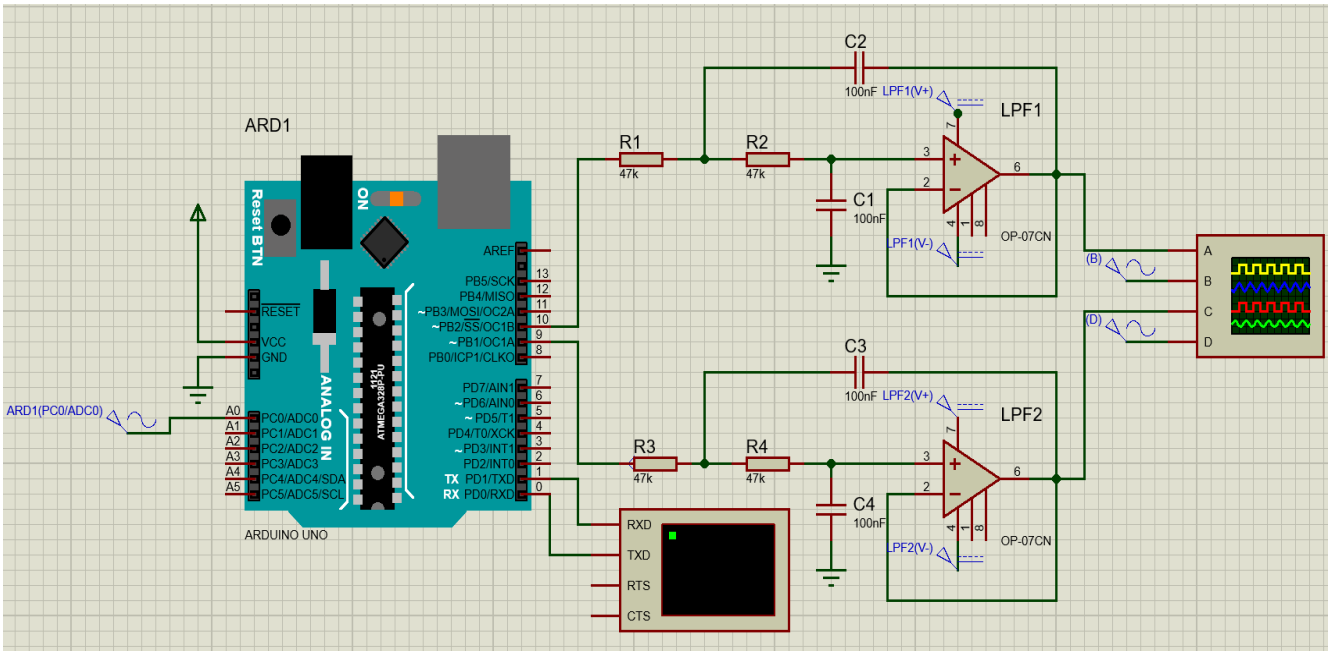

Figure 4.1: Electrical diagram in Proteus

4.1 Simulation

Using the Proteus Software to simulate the electrical diagram.

1) A sine generator produces a signal that varies from 0 to 5V representing a potentiometer with an opening ranging from 0 to 100%. This sine generator is connected to pin A0 of the Arduino board.

2) Arduino is supplied by a 5V source. The PWM pulses are generated to pin 9 and pin 10 on Arduino. PWM1 at pin 9 goes through LPF1 and PWM2 at pin 10 goes through LPF2. LPF1 and LPF2 are connected to port C and A on oscilloscope to display the shape of wave.

3) A sine generator produces a signal that varies from 1.004 to 4.35V representing a desired signal of APS1. This sine generator is connected to port B of the oscilloscope to compare with reality signal at port A.

4) A sine generator produces a signal that varies from 0.508 to 2.209V representing a desired signal of APS2. This sine generator is connected to port D of the oscilloscope to compare with reality signal at port C.

5) Pin RXD and TXD on Arduino are connected to pin TXD and RXD on Virtual Terminal to display voltage value of two signals APS1 and APS2.

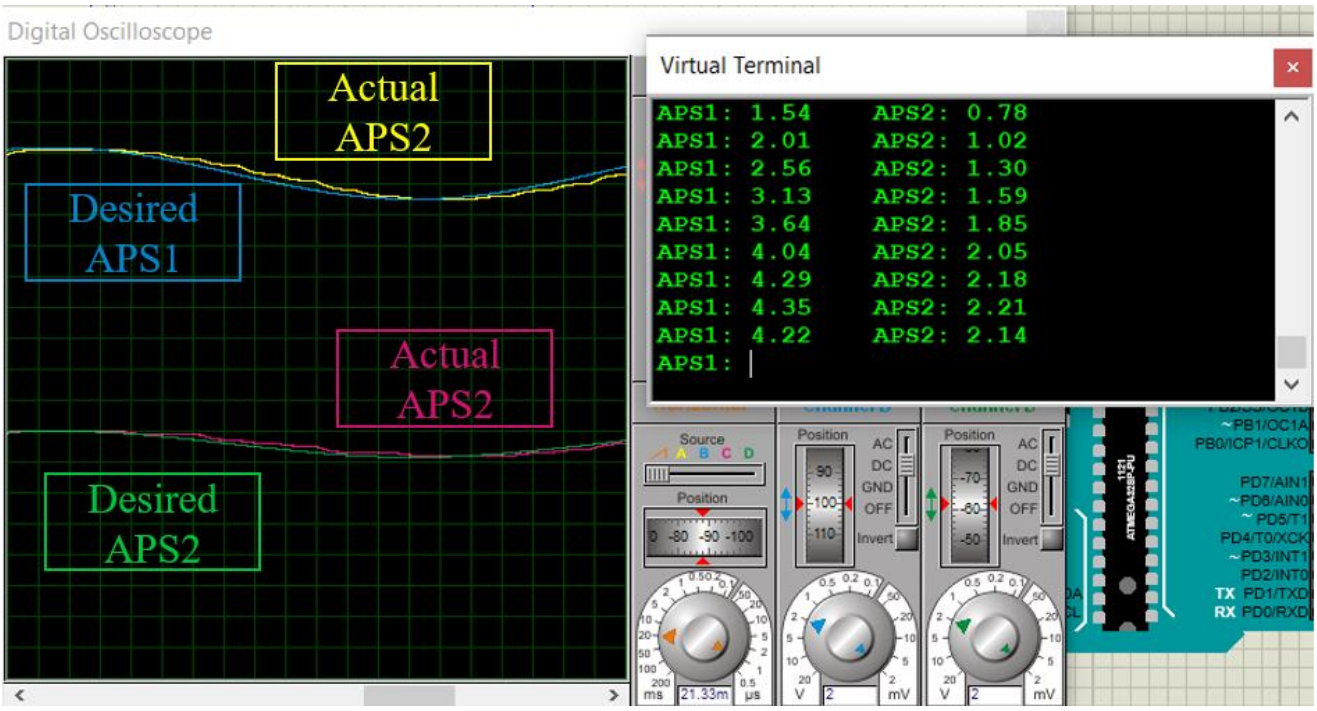

Figure 4.2: The result of accelerator pedal position signal simulator

4.2 Discussion:

Accelerator pedal signal	Actual	Desired	ERROR
APS1 at 0% pedal opening	1	1.004	0.4%
APS2 at 0% pedal opening	0.51	0.508	0.2%
APS1 at 50% pedal opening	2.68	2.677	0.3%
APS2 at 50% pedal opening	1.36	1.3585	0.15%
APS1 at 100% pedal opening	4.35	4.35	0%
APS2 at 100% pedal opening	2.21	2.209	0.1%

Table 4.1: Result of testing

Because OCR1A and OCR1B can only read integers, the pulse width is slightly different due to lack of precision so that the excess will be rounded to the first unit.

Virtual Terminal in Proteus only displays 2 decimals, so it will be slightly different from the actual measurement.

V. Conclusion:

After working on the project of simulating the accelerator pedal voltage system by using Arduino is a microcontroller and Proteus to simulate the system.

The system has satisfied the following conditions:

- Read the changing of potentiometer as a pedal and generate voltage values nearly the same with the reality value of Mitsubishi Xpander.
- Calculate the voltage value and display the PWM with low error.

However, the system also has some weakness:

- The program is still unable to calculate and convert at the speed of reality.
- Potentiometer can't give signals like the pedal.
- The shape of the pulse is still not smooth.

VI. Appendix

6.1 Code:

```
#include <avr/interrupt.h>

bool _flag_update_ = 0;

unsigned int _count_;

unsigned int _voltage_1, _voltage_2, ana_Read;

double   APS1, APS2;


void setup()

{

Serial.begin(9600);

pinMode(10,OUTPUT);

pinMode(9,OUTPUT);


cli();

TCCR1A = 0;

TCCR1B = 0;

TIMSK1 = 0;

TCCR1B |= (1 << WGM13) | (1 << WGM12) | (1 << CS11) | (1 << CS10);

TCCR1A |= (1 << WGM11) | (1 << COM1A1)| (1 << COM1B1);

TCNT1 = 0;
```

```
ICR1 = ((unsigned short int) (16000000.0/64.0/10000.0)) - 1;
OCR1A = 0;
OCR1B = 0;
TIMSK1 = (1 << TOIE1);
sei();
}

void loop()
{
if (_flag_update_)
{
_flag_update_ = 0;
ana_Read = analogRead(A0);
_voltage_1 = map(ana_Read, 0, 1023, 1004.0, 4350.0);
_voltage_2 = map(ana_Read, 0, 1023, 508.0, 2209.0);
APS1 = _voltage_1 / 1000.0;
APS2 = _voltage_2 / 1000.0;
OCR1A = (unsigned short int) ((_voltage_1/5000.0) * ICR1);
OCR1B = (unsigned short int) ((_voltage_2/5000.0) * ICR1);
Serial.print("APS1: ");
Serial.print(APS1);
Serial.print("    ");
Serial.print("APS2: ");
Serial.println(APS2);
}
}
ISR (TIMER1_OVF_vect)
{
if (++_count_ == 50)
{
```

```
_flag_update_ = 1;
_count_ = 0;
}
}
```

VII. REFERENCE

[1] **Your Bibliography:** 2021, https://www.electronics-tutorials.ws/filter/filter_2.html.

[2] **Your Bibliography:** "Before You Continue To Youtube". *Youtube.Com*, 2021, https://www.youtube.com/watch?v=0cpSi8C95DY.

[3] **Your Bibliography: [4] Atmel corporation** "Timer/Counter Trên AVR/Arduino | Cộng Đồng Arduino Việt Nam". *Arduino.Vn*, 2021, http://arduino.vn/bai-viet/411-timercounter-tren-avrarduino?fbclid=IwAR2l9tPyOy-IkgSFGdrmPQjEyuy7EXVIFtx_VFuQ3A7K_EBupjfphyxR4RQww.thegioididong.com/hoi-dap/cam-bien-hall-la-gi-975732.

[4] Atmel corporation - 01/2015 - 1600 Technology Drive, San Jose, CA 95110 USA - Atmega328p Datasheet

https://bitly.com.vn/99w3u9

[5] Your Bibliography: "Arduino Uno R3 Atmega328p Arduino Compatible - Nexelectronic Best Online Price.". *Nexelectronic*, 2021,

https://nexelectronics.in/product/arduino-uno-r3-atmega328p-arduino-compatible/.