PEAT

Design and Simulation of Idling Start-Stop System on Motorcycle

Muhammad Harith Hamidoon[1], Syabillah Sulaiman[1,2]*

[1]Department of Mechanical Engineering Technology, Faculty of Engineering Technology,
Universiti Tun Hussein Onn Malaysia, 84600 Pagoh, Johor, MALAYSIA

[2]Automotive and combustion Synergies Technology Group, Faculty of Engineering Technology,
Universiti Tun Hussein Onn Malaysia, 86499, Pagoh, Johor, Malaysia

*Corresponding Author Designation

Abstract:
The idling start-stop system is one of the systems that has implemented into the motorcycles nowadays. Most of the automotive industry always compete to develop new system technology into the vehicle markets. This system mostly found in hi-tech motorcycles with relatively highly prices in the market compared to the low-end motorcycle. In this study, a start-stop system has been designed and simulated by using the TinkerCAD software. There are several components required to create the simulation of idling start-stop system such as DC motor, potentiometer, Arduino and LCD. The circuit and coding has been constructed based on the collection of information from previous research study. The result shows how the idling start-stop system works, the results of the simulation has been collected and discussed. In a future, this simulation can be used as a reference during development process of an actual system start-stop kit.

Keywords: Idling Start-Stop System, Simulation, Tinkercad Software

1. Introduction

Automobile industry is one of the single largest manufacturing sectors that links to many types of producing firms, from material producer to intermediate, capital manufacture and final assembler. This make the demand of the industry become higher. First generation of Proton Saga was introduced in 1985 and achieved an outstanding achievement in the Malaysian automobile industry [1]. According to The Star New, 10 300 units has been sold in Britain during in its first year which is far above of expectation. The sales of this national cars grew rapidly after a few years it was introduced [2]. This innovation has encouraged the other automotive manufacturers to make the new technology improvement on the vehicle market. Nowadays, there are variety system has been implemented into the

vehicles including the high-end motorcycle with relatively high cost in market. This type of system cannot be found in old model motorcycles. The objective of this project is to study about the vehicles that embedded with idling start-stop system and also to design and simulated idling start-stop system by using the TinkerCAD software. TinkerCAD is a free software that capable to design the simulation with the existing component in its library. At the end of project, the idling start-stop system will be simulated with its new circuit and coding that has been created during this project.

1.1 Start-stop system

Start-stop system is a technology that has been implemented into the vehicles. It has given benefit to the driver to by ensuring the reduction of the fuel consumption and also reduce the amount of times for the engine to operated. According to the articles, 50.00 % of the newly vehicles after 2014 has been equipped with the start-stop system and it is commonly used in luxury cars. A comparison shows that the world market for the start-stop vehicles is greater than the hybrid vehicles [3]. Impact of start-stop system are depending to the driving cycle. Table 1 shows the result of experiment of four different driving styles with the start-stop system. The first driving styles is in the standard condition, second cycle is a crowded driving cycle, third is start and stop are frequent in short time and fourth is no frequent start and stop [4]. This experiment shows that the impact of the start-stop system are depending on its driving styles.

Table 1: Four typical driving cycle parameters

Driving cycle type	Total driving time/s	Total idle time/s	Idle time/driving time	Total idle times	Ineffective idle stop times
1	600	392	65.33%	12	1
2	1083	310	28.62%	25	4
3	2689	214	7.96%	53	49
4	600	110	18.33%	7	0

2. Materials and Methods

In this study, the simulation has been done in TinkerCAD software. The first steps are gathered all the information from the literature study to ensure the program can be run successfully. There are several components has been used during this simulation. Every component has been constructed with specific coding to ensure the component can run according to the desired system.

2.1 Materials

In this simulation program, there are several components required to build the program such as DC motor, Arduino Mega, Breadboard, LCD display, Resistor, LED and H-Bridge Motor Driver.

2.2 Methods

Based on the material listing above, 2 DC motors were used in the simulation. The DC motor acts as the motorcycle engines. Arduino MEGA® is an open-source microcontroller which can be programmed, erased and reprogrammed. There are various types of Arduino board such as Arduino Ethernet shield that allows the board to connect with the internet [5]. The power supply by the Arduino are only maximum at 5V, it will control the input and output for the program. Breadboard is used as a platform for the connection between the component. LCD is to display the switch of the motor and also value of potentiometer meter reading. Lastly, potentiometer is the component that acts as the throttle for controlling the start-stop system.

All of the components were selected from the software's library. Figure 1 shows the complete circuit of this study. After completed the circuit, the coding was created for the program. All the component functionality was defined in the coding. Coding is a computer language also knows as

machine code that helps the computer to understand. Every single coding constructed will constructed a new desired program [6]. In this study, the loop sequence was used for this program. Loop sequence is a programming structure that repeat a sequence of instruction until achieved to the desired condition.
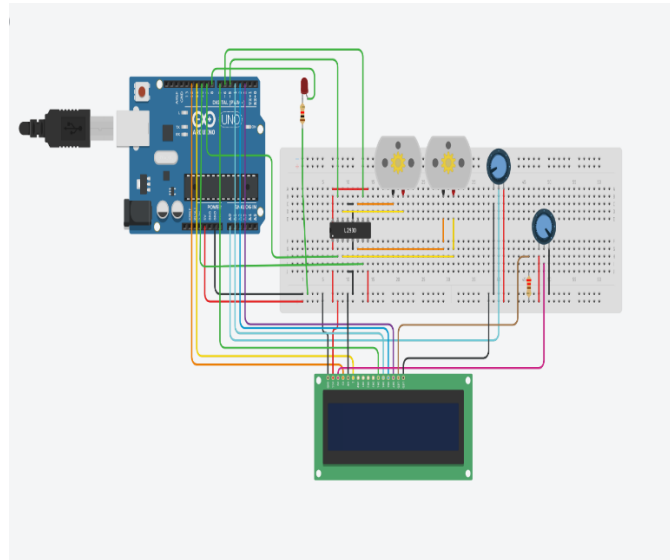

Figure 1: Complete circuit of Idling Start-Stop System

Figure 2 shows the diagram of L293D driver board. It has 16 pins which are used for different purposes. This driver's function is to control two DC motor at the same times. L293D driver board was connected to the 5 V of Arduino to supply the board. The 5 V power will run to the DC motor through driver's pins enable 1,2 and enable 3,4. Each DC motor will use 2 outputs. output 1,2 was connected for the DC Motor B and for output 3,4 was connected for the DC Motor A. Input 1,2,3,4 are connected to the Arduino.
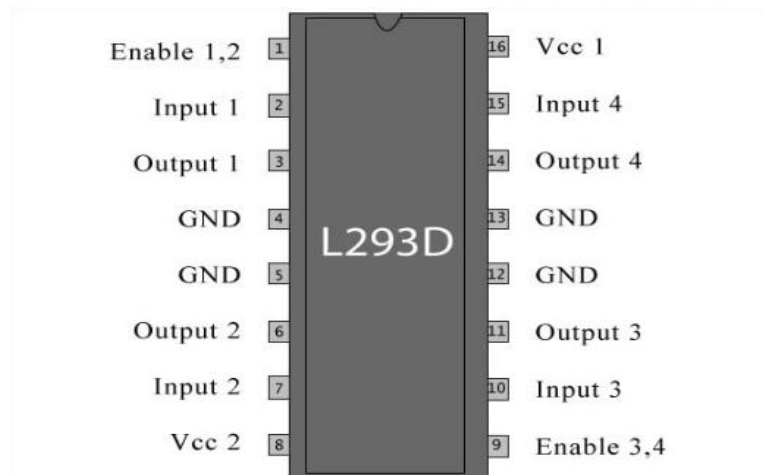

Figure 2: Module l293D Driver Board

Serial peripheral Interface (SPI) connection is used for the LCD connection. SPI connection is a common serial connection that is often used to connect the LCD to the Arduino. The LED is used to shows that the motor is running. LED will turn the switch according to the motor. Signal output for the potentiometer will be connected to the pin A0 of Arduino because it in analog value. The potentiometer will control the resistance so that the resistor will reduce the 5 V power to the lower value.

Figure 3 below shows the code of the program. First, motor 1 uses pins 5 and 6 while motor 2 uses pins 10 and 9. Several variables were setup in this program such as the threshold stop value was setup to 100 of potentiometer reading and timer which is 3000 ms. Interface pins of 7,4,3,2 was connected with the LCD and pins 12,11 is a fixed pin due to common SPI connection of LCD display. Next, the position of LCD display will be setup according to the desired parameter. For example, "POT" position is located at parameter (0,0) and "Motor" position is located at parameter (0,1). The loop sequence has been used for the whole program. If the values of potentiometer reading is below than the threshold stop value for 3 seconds, both motors will automatically turn off by the Arduino. During off mode, if the potentiometer turns to higher than threshold stop value, the motor will automatically turn on immediately. This looping program will run continuously until end of the simulation

```
//Motor 1
const int motor1_Pin1  = 5;  // Pin 14 of L293
const int motor1_Pin2  = 6;  // Pin 10 of L293
//Motor 2
const int motor2_Pin1  = 10; // Pin  7 of L293
const int motor2_Pin2  = 9;  // Pin  2 of L293

int MotorRun = 1;
int delayedStop = 0;

int ThresholdStop = 100;

// include the library code:
#include <LiquidCrystal.h>

// initialize the library with the numbers of the interface pins
LiquidCrystal lcd(12, 11, 7, 4, 3, 2);

int pot = A0;

unsigned long IntervalPeriod = 3000;            // refresh rate to upload to Thingspeak, every 20sec
unsigned long previousMillis1 = 0;

//This will run only one time.
void setup(){
  // set up the LCD's number of columns and rows:
  Serial.begin(9600);
  lcd.begin(16, 2);

  pinMode(8,OUTPUT);
  digitalWrite(8,LOW);

  //Set Motor pins as outputs
  pinMode(motor1_Pin1, OUTPUT);
  pinMode(motor1_Pin2, OUTPUT);
  pinMode(motor2_Pin1, OUTPUT);
  pinMode(motor2_Pin2, OUTPUT);
}

void loop(){

  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("POT:");

  int potReading = analogRead(pot);
  //Serial.print(potReading);

  int outputValue = map(potReading, 0, 1023, 0, 255);
  lcd.print(potReading);

  lcd.setCursor(9, 0);
  lcd.print("PWM:");
```

```
lcd.print(outputValue);

lcd.setCursor(0, 1);
lcd.print("Motor:");
if (MotorRun == 0)
{
  lcd.print("Off");
  digitalWrite(8,LOW);
}
else
{
  lcd.print("On");
  digitalWrite(8,HIGH);
}

if (MotorRun == 1)                // motor run seperti biasa
{
digitalWrite(motor1_Pin2, LOW);
digitalWrite(motor2_Pin1, LOW);

analogWrite(motor1_Pin1, outputValue); //motor run mengikut potentiometer
analogWrite(motor2_Pin2, outputValue);

  if (outputValue <= ThresholdStop && delayedStop == 0)   //sekiranya pot kurang dari threshold, akan trigger tim
ing
  {
    delayedStop = 1;
    previousMillis1 = millis();
  }

}

if (delayedStop == 1)   //sekiranya timing stop dimulakan
{
 Serial.println(millis() - previousMillis1);
 if((unsigned long)(millis() - previousMillis1) >= IntervalPeriod)   //check sekiranya interval telah lebih (3sec)
  {
    //check if 3 sec interval is pass

   previousMillis1 = millis();
    digitalWrite(motor1_Pin2, LOW);   //stop motor
   digitalWrite(motor2_Pin1, LOW);
   analogWrite(motor1_Pin1, 0);
   analogWrite(motor2_Pin2, 0);
    MotorRun = 0;
  }
}
if (MotorRun == 0)   //sekiranya motor stop, dan pot lebih dari threshold
{
 if (outputValue > ThresholdStop)

{
 MotorRun = 1;      //motor run semula
 delayedStop = 0;
 previousMillis1 = millis();
}
}

 delay(100);
}
```

Figure 3: Code of Program

3. Result and Discussion

3.1 Result

The simulation of idling start-stop system was run successfully on the TinkerCAD. All of the component performed according to each specific code. Table 2 shows the result of the simulation based on the display on LCD:

Table 2: LCD Display between two modes of the program

ON MODE	OFF MODE
POT:368 PWM:91 Motor:On	POT:82 PWM:20 Motor:Off
• Value of 368 represent as a potentiometer reading value. When the potentiometer reading is higher than 100, the motor will automatically turn on. On LCD, it will show the value of POT and also condition of the motor which is in ON mode,	• Value of 82 is the potentiometer reading value. If the potentiometer reading is reduced below than 100, the motor will automatically turn off after 3 seconds. Condition of the motor will display as "OFF".

3.2 Discussions

All of the component has been assembled and coded individually. On the start of the simulation, the circuit received 5 V input power from the Arduino. The motor run accordingly to the voltage which was controlled by the potentiometer. The display on LCD shows three types of values which is potentiometer reading, pulse width modulation (PWM) and condition of DC motor. By turning the potentiometer in anticlockwise rotation will reduce the POT reading. When the POT values turned below than 100 in 3 seconds, it triggers the Arduino to turn off the motor and LED. According to Figure 5, there a slight difference of simulator time between these two modes which is. It proved that the 3 seconds to trigger the Arduino to automatically function. The limitation of this simulation is the simulator time are depending on the specification of the computer. The lower the specification, the slow reaction of the simulation.
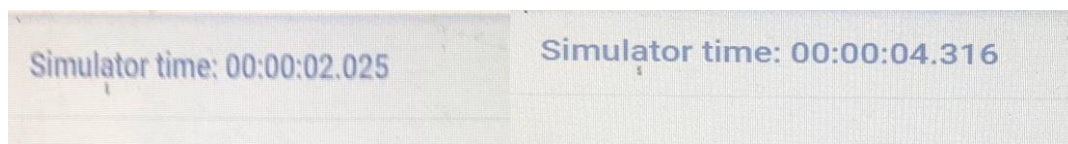

Figure 4: Comparison simulator times between ON mode and OFF mode

4. Conclusion

At the end of the project, the objectives of the project have been achieved which is to design and simulate the idling start-stop system using TinkerCAD software. The circuit of idling start-stop system has been successfully created coded based on requirement of the idling start-stop. This simulation is preliminary study of the idling start-stop system kit for the motorcycles. The programming code also can be further improvised for the future studies.

Acknowledgement

References

[1]   Wad, Peter, and V. G.R. Chandran Govindaraju. 2011. "Automotive Industry in Malaysia: An Assessment of Its Development." International Journal of Automotive Technology and Management 11(2): 152–71.

[2]   Gurmeet Kaur. 2016. "Proton's Bumpy Journey |TheStar." https://www.thestar.com.my/business/business-news/2016/04/09/protons-bumpy journey/

[3]   Mechanical, Journal O F, and Transport Engineering. 2017. "Effects Of Start – Stop System On The Operation." 67(2).

[4]   Zhong, Qingyun, Huabiao Qin, and Ruoqian Xu. 2018. "Study on the Start-Stop System Control Strategy under Different Driving Cycle." *IEEE International Conference on Control and Automation, ICCA* 2018-June: 223–28.

[5]   Louis, Leo. 2016. "Working Principle of Arduino and Using It As a Tool for Study and Research." International Journal of Control 1(2): p. 21–29

[6]   AN BUCKLEY. 2019. "What Is Coding and How Does It Work?" https://www.makeuseof.com