21028\_ INVINCIBLES\_ Innovation Report

**INTELLIGENT MANUAL TRANSMISSION**

Author’s Name: C.Bharath

Co-Author’s Name: Sufyaan

1. Abstract

Nowadays, an increasing number of people commute by bicycles. However, it is a challenging thing for many bikers to decide the optimal gear under various conditions and when to change the gear. Thus, intelligent manual transmission can satisfy the needs of most inexperienced bikers.

EFFICYCLE changes the gear by pulling a metal cable connected to the derailleur. Our goal is using sensors and microcontroller to electronically determine and shift to the optimal gear under specific conditions, by pulling the rear derailleur steel shift cable in the EFFICYCLE with servo motor. Besides shifting automatically, we have added a manual mode which offers the user an option to take control of the transmission, if he or she enjoys the fun of manual switching the gears.

1. INTRODUCTION

There are the three main advantages with the automatic transmission system:

1)It can make your journey more comfortable. Except for expert riders, many people cannot select the right gear unconsciously. Moreover, with so many traffic signals riders have to change gears very frequently. However, with this system riders only have to worry about the pedaling.

2)Electronic automatic gear shifting can guarantee a safer journey relatively.

3)Riders can ride more efficiently with the optimal gear ready, the rider could pedal at an optimal cadence, for the inexperienced riders who choose wrong gear it is a very useful tool.

The transmission system we design could potentially serve as an add-on device at an acceptable price for all bicyclists. The low prize and easy installation could attract more people to our product. Our design won’t necessarily compete with the current commercial products, but it could let more people have a taste of intelligent manual shifting with a low price.

**2.1 High-level Requirements**

1**.** Our system must be able to determine the optimal gear from 9 gears, based on the conditions described by sensors. Quantitatively speaking, our system should select a lower gear when cadence lowers, and select a higher gear when the cadence.

2. Our system should have a manual mode, which could allow bikers to change gears manually as per their choice.

3. Our system could be installed on a bicycle easily.

**3. REVIEW OF EXISTING RESEARCH**

**Electronic automatic transmission for bicycle** a document by TIANQI LIU, actually he participated in ECE 445 – Spring 2018

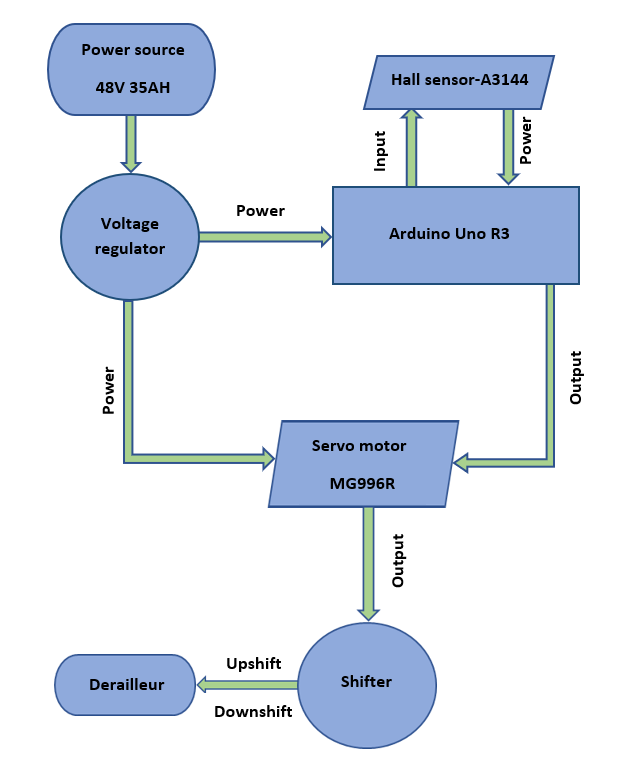
In his document he explained his research work very well. Basically, he divided it into 4 subtopics – namely power system, sensing unit, control unit, motor module. In their work they used a 7.4hr battery, hall sensor, speed sensor, gear motor, ATmega328, and 3 buttons to up shift and down shift them, they have used LCD screen for displaying all values. A brief explanation regarding automatic transmission is given in his document.

Our design works on a similar basis of cadence measurement but takes a very different approach of actuating the gear shift. We use an angle-based shifter, to which we attach a high torque motor and control the motor to manipulate the shifter.

4. INNOVATION CONCEPT AND TECHNICAL FEATURES

design:

Our design mainly contains 5 modules: power supply, sensing unit, control unit, motor, buttons. The power supply regulates and provides stable electrical power to every component of our device. The sensing unit measures the rider’s cadence and control unit collect the information and gives input and output. Motor module pulls or releases the shift cable to actuate the rear derailleur and shift gears



The above block diagram represents the working of the intelligent manual transmission system. It works on the closed-loop feedback system of the servo motor and Arduino Uno.

Power supply

The power supply module contains a Li-ion battery and voltage regulator. This part provides electricity to all components. When the servo motor is switched ON, the theoretical maximum input of current that it can take is around 900mA and for the ARDUINO, it is around 50mA.

Lithium-ion battery:

We plan to use a 48 Volt, 35Ah Capacitance battery, which is also the main supply for all the electrical components in our Efficycle.

REQUIRMENT

1. Can power the servo motor to pull the cable to each gear. The calculations for this requirement are provided in the succeeding section.
2. Can continuously power the system up to 10 hours.
3. Output voltage is 48V and is stepped down to ~5V using the regulator.

VALIDATION

1a. Connect the motor with the microcontroller. Link the cable to the motor output.

1b. Use the manual mode and up shift and down shift button to switch

1c. Repeat it from highest to lower gear

1d. in this process, the battery should power the motor to conduct all these actions within 0.2second

1. In the 8.7 hrs. test the battery should power the system without recharging
2. The voltage output measured by the multi meter should be between 46V-48V in any working conditions

VOLTAGE REGULATOR

In order to obtain stable battery voltage even when the battery is discharging, we need a voltage regulator. We can convert 48V Battery to a 5V power source and sustain maximum current capacitance of 3.5Ah.

REQUIRMENT

1. Should sustain 3.5 ampere current without over heating (<120 degree Celsius)
2. Output voltage is 5V, when a continuous 48V input supply is applied

SENSING UNIT

In order to choose the optimal gear for transmission, we should monitor the cadence to check for the range of cadence the rider is pedaling at, for which, we use a cadence sensor. All the data is received and processed by the microcontroller.

Cadence sensor

We have used a Hall-effect sensor (A3144) to measure the cadence of the rider. It is very common and easy to use and is chosen because of its low price and accuracy. This sensor is placed at the pedal mounting member and a small piece of magnet is attached to the crank set as shown, so that it can measure the real cadence and send it to the microcontroller.

REQUIRMENT

Achieve an accuracy of 3RPM.

VALIDATION

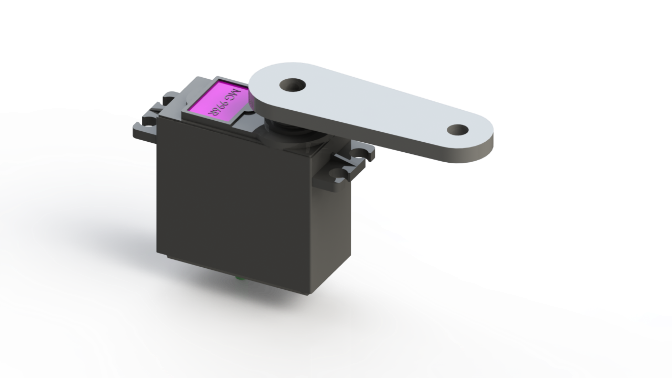
1. Mount the cadence sensor on the bike pedal. Connect it to the microcontroller. Put the bike on the test shelf with back wheel free to move.
2. Write a test program that print out continuous reading of the cadence sensor
3. Rotate the bike paddle for 1 minute count the rounds the paddle rotates
4. Compare the counted number with the displayed number in the LCD

MOTOR MODULE

The motor module receives signals from the control unit and it mechanically actuates the shift cable, according to our requirement, it requires a force of at least 5kg to pull the cable.

The three types of motors that can be used for our application are gear motor, servo motor and stepper motor. Since our design requires only 10 kg force, servo motor has been chosen. It is also cost effective based on our requirement.

The high–torque MG996R digital servo motor metal features metal gearing resulting in high torque. This is very accurate, and it can rotate a total of 120 degrees.



TECHNICAL SPEIFICATIONS

Weight 55g

Dimensions 40.7\*19.7\*42.9

Stall torque 10 kgf.cm (5V)

Operating voltage 4.8V-7.2V

Running current 500mA-900Ma

REQUIRMENT

Have enough torque of 10 kgf.cm to pull the cable of rear derailleur.

Can be powered by 5v voltage source.

VALIDATION

1. Power the motor with the 5V voltage source. Connect the servo motor with control unit.
2. Rotate the pedal by hand
3. Switch the gear from lowest to highest.
4. Switch the gear from highest to lowest.
5. Each time the system should switch to desired gear within 1 second.
6. The process must be smooth.
7. The motor should hold the cable after switching to proper gear.

CONTROL UNIT

This unit combines the data from the sensing unit and user input, runs the software to determine the optimal gear or controls the manual mode function. This unit sends upshift/downshift signals to the motor module.

MICROCONTROLLER

For the microcontroller, we are using the ARDUINO UNO R3. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator.

The UNO board is the first in the series of USB ARDUINO boards, as it is cost effective, can be installed easily and we can replace the chip for very low cost.

Technical specifications

Microcontroller ATmega328P

Operating voltage 5V

Input voltage 7-12V

Digital I/O 14

DC current per I/O 20mA

Flash memory 32kb

Dimensions 68.6\*53.4

Weight 25g



REQUIRMENT

1. Operate reliably for 2 digital input,3 analog input and 1 digital output with 5V power source
2. Digital outputs sent out with no observable delay
3. Receive digital input without missing

VALIDATION

1. Connect all the digital output pin to LEDS
2. Load a test program that send 100 pulses to all digital outputs with 1 second interval
3. By human observation, all the LEDs should light up and turned off simultaneously for 100 times
4. Connect all the digital input pin to push button
5. Load a test program that print out the count number of digital input pulses on terminal
6. Press one push button for 100 times with 1 second interval
7. The number displayed on terminal should be exactly 100

5. calculation and analysis

Calculations:

Since we are using a 4.8-6VDC (5V typical), according to data sheet of MG996R maximum current is 900ma when the servo is installed. Thus, the maximum theoretical power of the servo is:

5V \* 0.9A =4.5W

Based on personal biking experience, we assume that bike rider changes gear once per 10 second on average. And the target re-charging cycle of our battery is 8.7 hours. For each gear shifting, we assume that servo motor takes 200ms to finish the action. Since the servo only operates during gear shifting, the power consumption during this 8.7 period is

4.5w \* 0.2s \* 8.7/10s = 0.783Wh

The, ARDUINO have a constant power consumption. The actual value may vary depend on the working conditions. But our ARDUINO we choose for 3.3V PIN requires 50ma. The power consumption for this part is

5V \* 0.05 A \*10h =2.5Wh

Thus, our battery should have a capacity of

0.8 + 2.5 = 1.3 Wh capacity

As we have chosen a 48V battery and 5V linear regulator it is well enough to our system to work efficiently.

The code for our innovation is as follows:

#include <Servo.h>

#define inpin = 2

#define servopin = 9

#define button = 3

float revolutions=0;

float rpm=0;

long startTime=0;

long elapsedTime;

Servo myservo;

boolean ButtonState;

int upshift=1;

int downshift=1;

int hiRPM[9] = [50,64,70,77,83,90,97,103,110]

int loRPM[9] = []

int gear = 1;

void setup()

{

pinMode(inpin, INPUT);

myservo.attach(servopin);

Serial.begin(9600);

}

voide loop() {

ButtonState = digitalRead(button);

while (ButtonState == 0) {

revolutions=0; rpm=0;

startTime=millis();

attachInterrupt(digitalPinToInterrupt(2),interruptFunction,RISING);

delay(1000);

detachInterrupt(2);

elapsedTime=millis()-startTime;

if(revolutions>0)

{

rpm=(max(1, revolutions) \* 60000) / elapsedTime;

String outMsg = String("RPM :") + rpm;

Serial.println(outMsg);

readrpm();

while((upshift==1) && (downshift==1))

{

readrpm();

}

if ((upshift==0) && (gear==1))

{

myservo.write(14);

gear=2;

}

if ((upshift==0) && (gear==2))

{

myservo.write(28);

gear=3;

}

if ((upshift==0) && (gear==3))

{

myservo.write(42);

gear=4;

}

if ((upshift==0) && (gear==4))

{

myservo.write(56);

gear=5;

}

if ((upshift==0) && (gear==5))

{

myservo.write(70);

gear=6;

}

if ((upshift==0) && (gear==6))

{

myservo.write(84);

gear=7;

}

if ((upshift==0) && (gear==7))

{

myservo.write(98);

gear=8;

}

if ((upshift==0) && (gear==8))

{

myservo.write(115);

gear=9;

}

if ((downshift==0) && (gear==2))

{

myservo.write(2);

gear=1;

}

if ((downshift==0) && (gear==3))

{

myservo.write(15);

gear=2;

}

if ((downshift==0) && (gear==4))

{

myservo.write(28);

gear=3;

}

if ((downshift==0) && (gear==5))

{

myservo.write(42);

gear=4;

}

if ((downshift==0) && (gear==6))

{

myservo.write(56);

gear=5;

}

if ((downshift==0) && (gear==7))

{

myservo.write(70);

gear=6;

}

if ((downshift==0) && (gear==8))

{

myservo.write(84);

gear=7;

}

if ((downshift==0) && (gear==9))

{

myservo.write(98);

gear=8;

}

}

Serial.print("Current Gear = ");

Serial.println(gear);

}

delay(1500);

void interruptFunction()

{

revolutions++;

}

void readrpm(){

if (rpm > hiRPM[gear-1]){

upshift = 0;

downshift = 1;

}

if (rpm < loRPM[gear-1]){

upshift = 1;

downshift = 1;

}

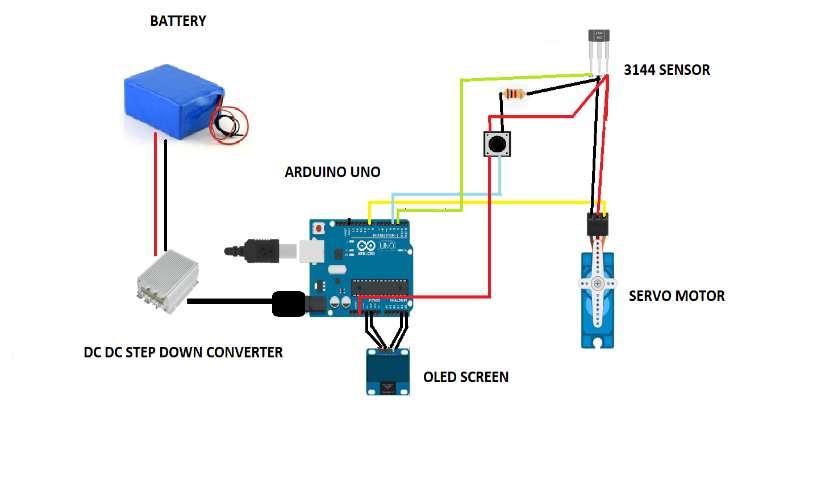
else{

upshift = 1;

downshift = 1;

delay(100)

}}

**CIRCUIT DIAGRAM**

**6. MASS PRODUCTION FEASIBILITY**

The following section has been written assuming that our manufacturing firm can produce 2000 units of this system per year.

**Inventory:** The following table represents the materials, tools and the cost required to produce the system on a large scale.

|  |  |  |  |
| --- | --- | --- | --- |
| S. No | Material/Tool | Units | Cost |
| 01 | CNC Machines | 1 | 80,000 |
| 02 | 3D Printers | 3 | 75,000 |
| 03 | Alloy Steel Plates | 70 | 5000 |
| 04 | PLA Filament | 130 | 1,17,000 |
| 05 | PCB | 2100 | 2,00,000 |
| 06 | Labor | N/A | 1,20,000 |
| 07 | Servo Motor (MG996R) | 2100 | 6,30,000 |
| 08 | Wiring | N/A | 2,00,000 |
| 09 | Fasteners | N/A | 10,000 |
| 10 | A3144 Hall-Effect Sensor | 2100 | 4,20,000 |
| 11 | Custom Shifter | 2100 | 6,00,000 |

The total cost for the set up turns out to be around 24.5 lacs INR. And the materials are readily available in local markets adding to the feasibility of mass production.

**Production Methods**

Housing: The servo motor housing will be 3D printed using polylactic acid. PLA is the most widely used 3D printed material after ABS due to its high tensile strength and low price.

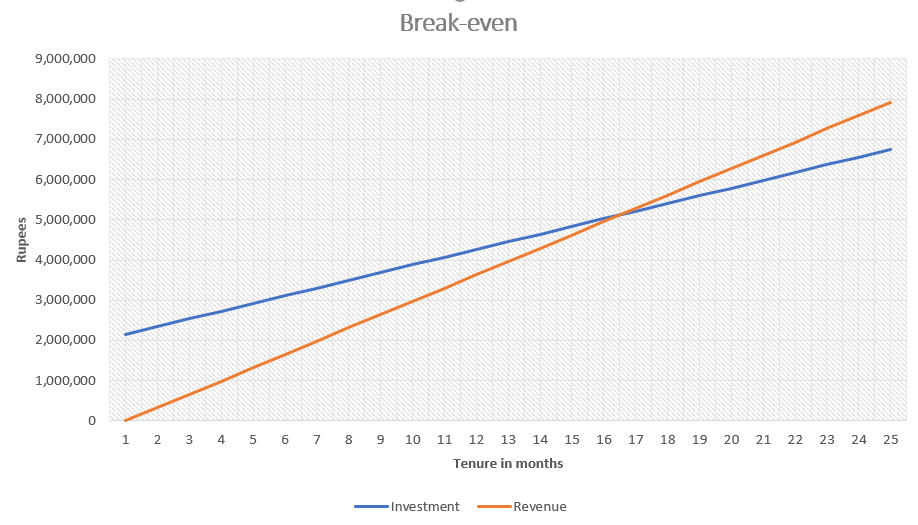
Motor: Servo Motor MG996R is mass purchased from wholesale merchants.

Shaft: The shaft for Motor MG996R is in-house manufactured by simply cutting from Cast alloy steel sheets using CNC machines.

PCB: In context of mass production, using custom PCBs is more profitable and easier to use than pre-manufactured micro-controllers such as Arduino UNO.

**Turnover**

Selling the product for Rs. and assuming a constant selling rate of 165 units per month, the break-even occurs in 16 months with a selling price of Rs. 2000/- only.



Hence, the mass production of this system is feasible.

**7.COST CALCULATIONS**

**ESTIMATED COST OF THE PARTS**

|  |  |  |
| --- | --- | --- |
| PART NAME | QUANTITY | UNIT PRICE |
| Voltage regulator | **01** | **300** |
| A3144 Hall-Effect Sensor | **01** | **200** |
| MG996R servo motor | **01** | **260** |
| Arduino UNO R3 | **01** | **550** |
| PLA (1kg) | **01** | **800** |
| microShift Advent x9 Gear Shifter | **01** | **1200** |

Total cost estimated = Rs. 3310/-

**8.Result and conclusion**

Regarding safety of our driver, our overall design is safe enough, has low budget and easy installation which makes it easy to ride. Finally, we can conclude that our innovation takes a step ahead in transmission system and makes the journey safe, comfortable and more efficient.

9.Scope and applications

This Intelligent Manual Transmission System can be implemented in bicycles at a large scale making the rider carefree of the gear system but at the same time enjoying the benefits of the manual transmission.

**10.Limitations and future modification scope**

* The mechanism must be calibrated at the beginning of each ride.
* An electronic clutch has been added to the design which must be actuated during free- trailing to suppress the mechanism for the given period. This works against the ease of the rider.
* Complications of the derailleur-chain still exist and might cause problems.

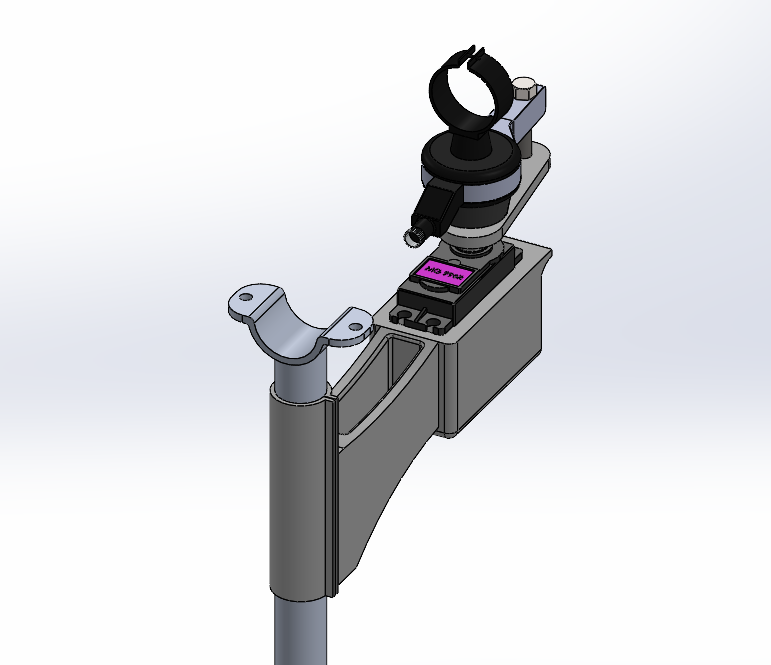
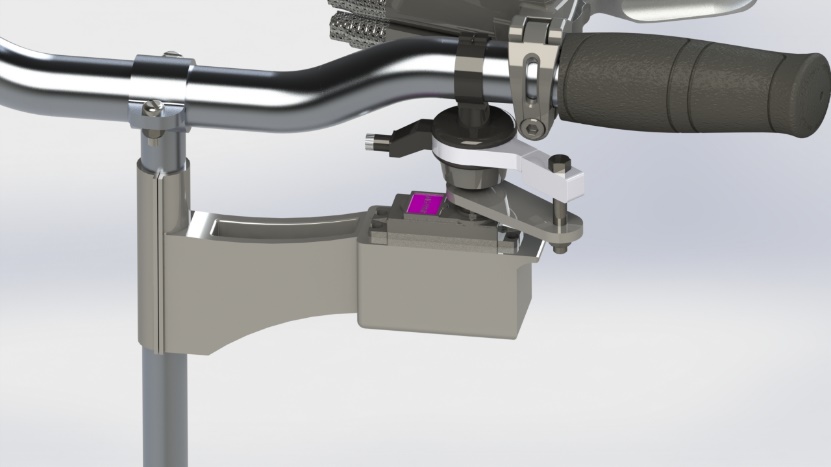
**MODIFICATIONS**

* The code of the system can be further improved, working on the free-trailing and panic breaking scenarios.
* Removal of clutch

**11.References**

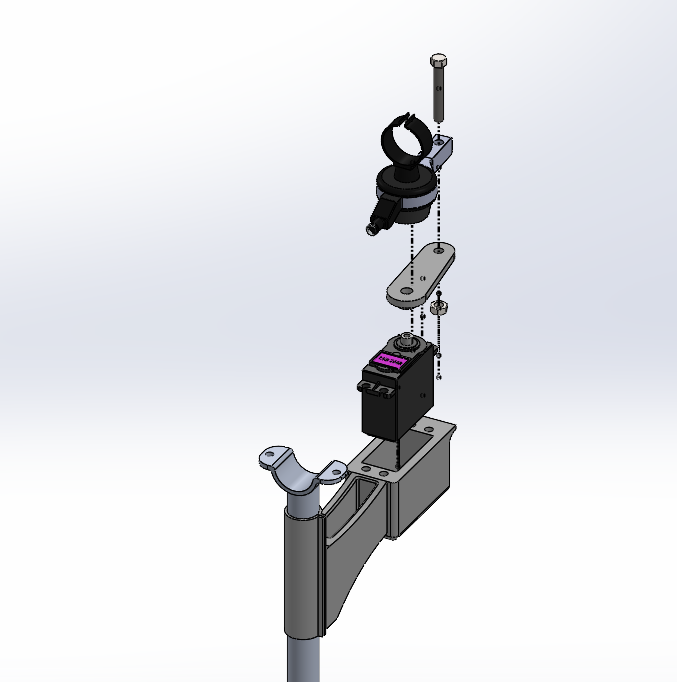
1. <https://www.alldatasheet.com/view.jsp?Searchword=Mg996r&gclid=CjwKCAjw8KmLBhB8EiwAQbqNoH-UeRIgrPdlMebTZ-gqgR4FoM3XLajdzSXEg1IhfjqoKEa4PWvt9BoCEnAQAvD_BwE>
2. <https://store.arduino.cc/products/arduino-uno-rev3>
3. [www.exicom-ps.com](http://www.exicom-ps.com)-the future of mobility is shared, connected & electric.
4. <https://github.com/vanvuongngo/shift4me>

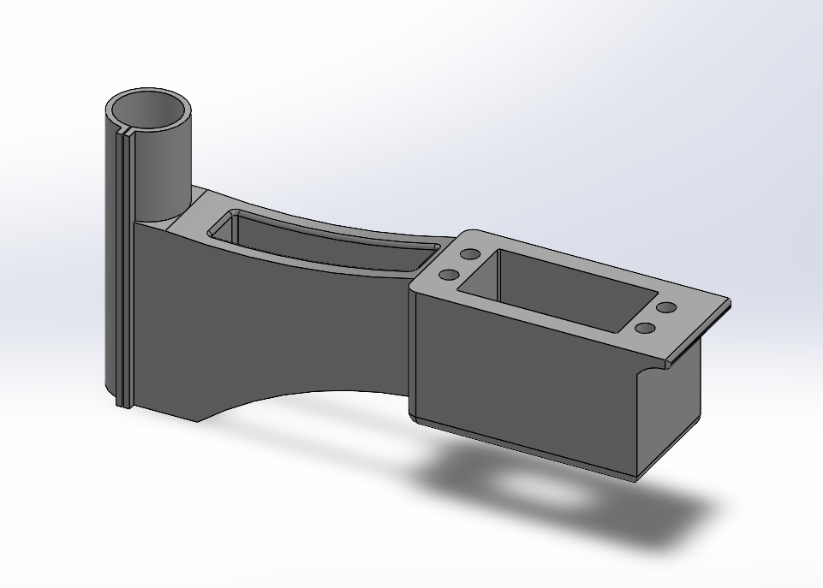
**APPENDIX 1: DESIGN VIEWS AND PHOTOGRAPHS**



Rendered image

Isometric View





3D Printed Servo Motor Housing

Isometric Exploded View