B.TECH. PROJECT REPORT

On

Design & development of adaptive path follower for wire based additive manufacturing

By

Vinod Dawar

Y Ashok Kumar

DISCIPLINE OF MECHANICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY INDORE

November 2018

Design & development of adaptive path follower for wire based additive manufacturing

A PROJECT REPORT

Submitted in partial fulfilment of the
Requirements for the award of the degrees

Of
BACHELOR OF TECHNOLOGY
in
MECHANICAL ENGINEERING

Submitted by:
VINOD DAWAR (150003038)
Y ASHOK KUMAR (1200340)

Guided by:
Dr. YUVRAJ KUMAR MADHUKAR (Assistant Professor, IIT INDORE)

INDIAN INSTITUTE OF TECHNOLOGY INDORE
November 2018

CANDIDATE'S DECLARATION

We hereby declare that the project entitled**"Design & development of adaptive path follower for wire based additive manufacturing"** submitted in partial fulfilment for the award of the degree of Bachelor of Technology in 'MECHANICAL ENGINEERING' completed under the supervision of **Dr. Yuvraj K Madhukar (Assistant Professor),** IIT Indore is an authentic work.

Further, we declare that we have not submitted this work for the award of any other degree elsewhere.

VINOD DAWAR

Y ASHOK KUMAR

CERTIFICATE by BTP Guide

It is certified that the above statement made by the students is correct to the best of my/our knowledge.

Dr. Yuvraj K Madhukar
Assistant Professor
Discipline of Mechanical Engineering
IIT INDORE

Preface

This report on "**Design & development of adaptive path follower for wire based additive manufacturing**" is prepared under the guidance of **Dr. Yuvraj K Madhukar (Assistant Professor).**

Through this report we have tried to give a detailed description of the concept, design and making of adaptive path follower for wire based additive manufacturing. We have tried to the best of our abilities and knowledge to explain the content in a lucid manner. We have also added designs and actual photographs of the structure for better understanding.

Vinod Dawar
Y Ashok Kumar
B.Tech IV Year
Discipline of MECHANICAL ENGINEERING
IIT Indore

Acknowledgements

We would like to thank **Dr.Yuvraj K Madhukar**for his kind support and valuable guidance.

We want to express our gratitude to **Mr. Anand Petare** and **technical staff** of **central workshop IIT Indore** for their extended help and support.

We would also like to thank **Mr. Anas ullah khan** for his help and motivation, at last but not the least my sincere thanks goes to **Mr. Suryaveer Singh** who helped in programming part.

It is their help and support, due to which we became able to complete the design and technical report

Vinod Dawar
Y Ashok Kumar
B.Tech IV Year
Discipline of MECHANICAL ENGINEERING
IIT Indore

Abstract

The adaptive path follower has many applications in wire based additive manufacturing process such as wire feed direction control, temperature sensor and shape sensor direction controls. This work is all about controlling the adaptive holder position in TIG welding with respect to motion of direction of tungsten rod while working on welding or additive manufacturing. TIG based welding/ additive deposition process is widely used for components of small shapes and complex geometries. The process required manual feeding of the filler material which limits the process accuracy or capability. Automatic feeding may reduce the human error and improve the process. In forehand welding the heat distribution is more uniform compared to side or rear feeding, which in turn melts welding rod as well as walls of work-piece. This will then distribute the weld puddle in a uniform manner. For this it is needed a programmed controlled motor which can respond according to adaptive path and control the rotational motion while welding additive deposition process is ongoing. Other potential uses for this project are to monitor the temperature using pyrometer and bead shape for the suitable feedback control.

Table of Contents

List of Figures

1. Introduction

1.1. Motivation

- The adaptive path follower has many applications in wire based additive manufacturing such as wire feed direction control, temperature and shape sensor direction control.
- This work is all about controlling the adaptive holder position in TIG welding with respect to motion direction of tungsten rod while working on welding or additive manufacturing.
- For this it will required a programmed controlled motor which can respond according to adaptive path and control the rotational motion while welding process is ongoing.
- Other potential uses for this project are to monitor the temperature using pyrometer and bead shape for the suitable feedback control.It is done by laser imaging as we can clearly see that from figure 1

Figure 1: Bead shape sensing

Figure 2: Pyrometer (used for temperature sensing)

1.2. What is additive manufacturing?

Additive manufacturing is a process used for building 3-D parts using
Plastics, metal, concrete etc By means of CAD models .Once a CAD sketch is formed process is done by layer by layer deposition of respective material, the entire process is computer controlled and the object formed is a perfect replica of the corresponding 3D CAD model designed earlier in relevant CAD software.

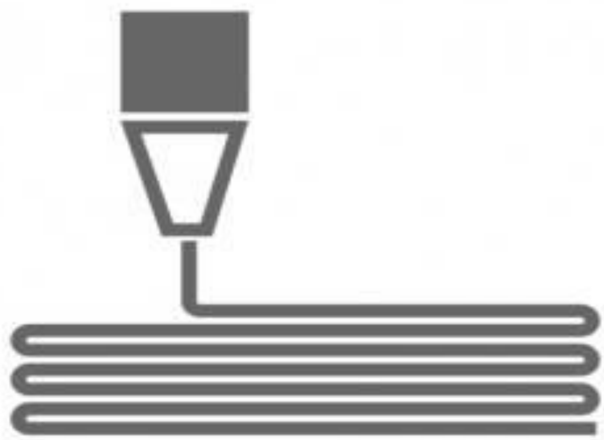


Figure 3: Layer by layer deposition

1.3. Why additive manufacturing?

Additive manufacturing (AM) is a promising alternative to traditional subtractive manufacturing for fabricating large expensive metal components with complex geometry. And manufacturing is done by layer by layer deposition method and the strengths of Additive Manufacturing exist in those areas where conventional manufacturing (i.e., subtractive manufacturing) reaches its limitations. The technology is of interest where a new approach to design and manufacturing is required so as to come up with solutions. It enables a design-driven manufacturing process - where design determines production and not the other way around. What is more, Additive Manufacturing allows for highly complex structures which can still be extremely light and stable. It provides a high degree of design freedom, the optimization and integration of functional features, the manufacture of small batch sizes at reasonable unit costs and a high degree of product customization even in serial production.

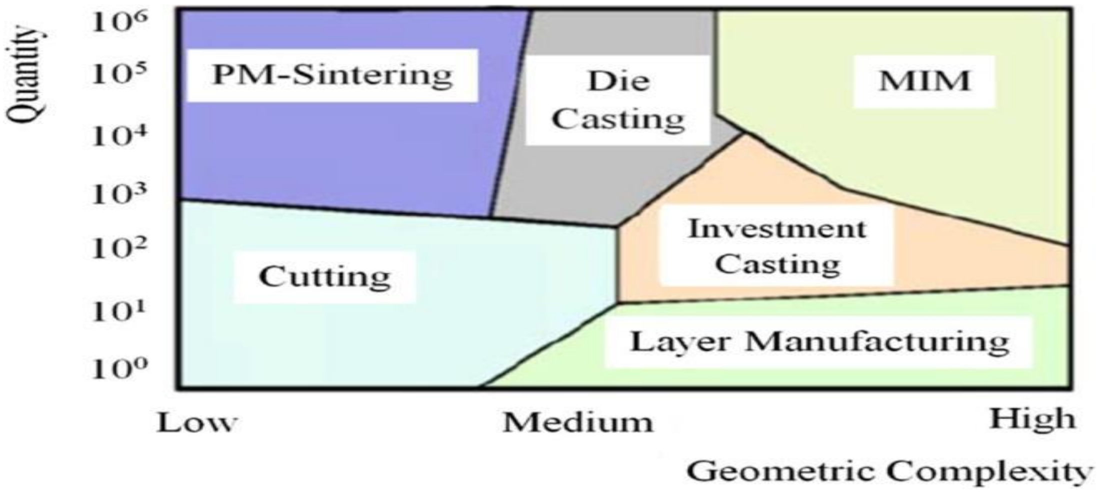


Figure 4 : Chart for comparing Quantity Vs Geometric complexity

Figure 5: Complex metal gear formed by Additive manufacturing

2. Literature review

There are many methods used for additive manufacturing such as GTAW (Gas tungsten arc welding) and PAW (plasma arc welding) use a non-consumable tungsten electrode to produce the weld, GTAW require filler metal rod from external source.

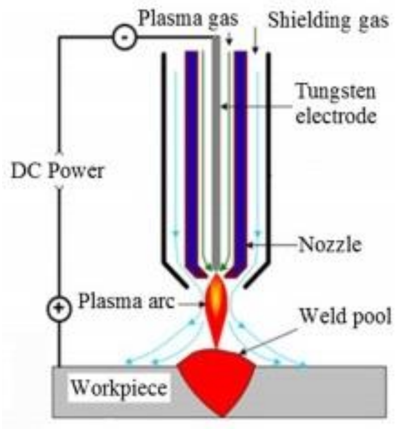


Figure 6: PAW (Plasma Arc welding)

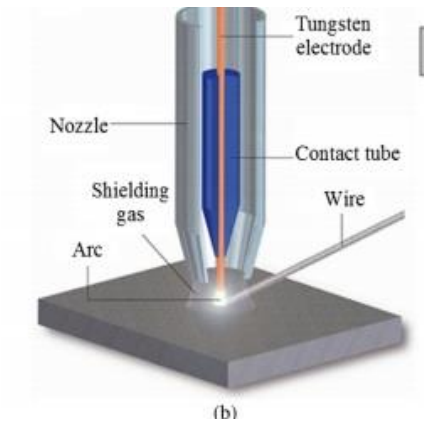


Figure 7: GTAW (TIG welding)

This project mainly based on wire feed additive manufacturing in which we are using TIG (Tungsten inert Gas) welding process for our metal deposition.

2.1. Why TIG?

TIG	MIG
-Very less spattering	-Spattering is very high
-Low deposition rate	-High deposition rate
-Gives more precise deposition	-Less precise deposition
-Less post processing required	- More post processing
Required	

Figure 8: Spattering in TIG welding (left) and MIG welding (right)

In TIG welding the direction of *feeder rod* should always *in front of direction motion of deposition*. In forehand welding the heat distribution is easier, which in turn melts welding rod as well as walls of metal piece. This will then *distribute the weld puddle in a uniform manner*.

When filler rod feeds from front direction in multilayer deposition it shows good bonding in between the tracks. And also, we can observe no porosity found between the layers.

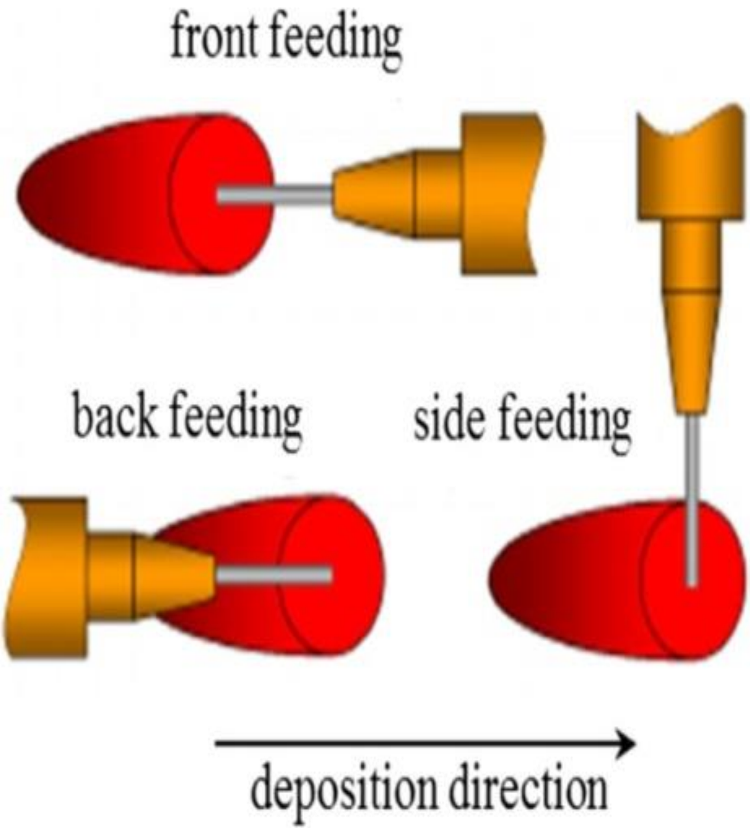


Figure 9: Types of feeding direction according to the given deposition direction

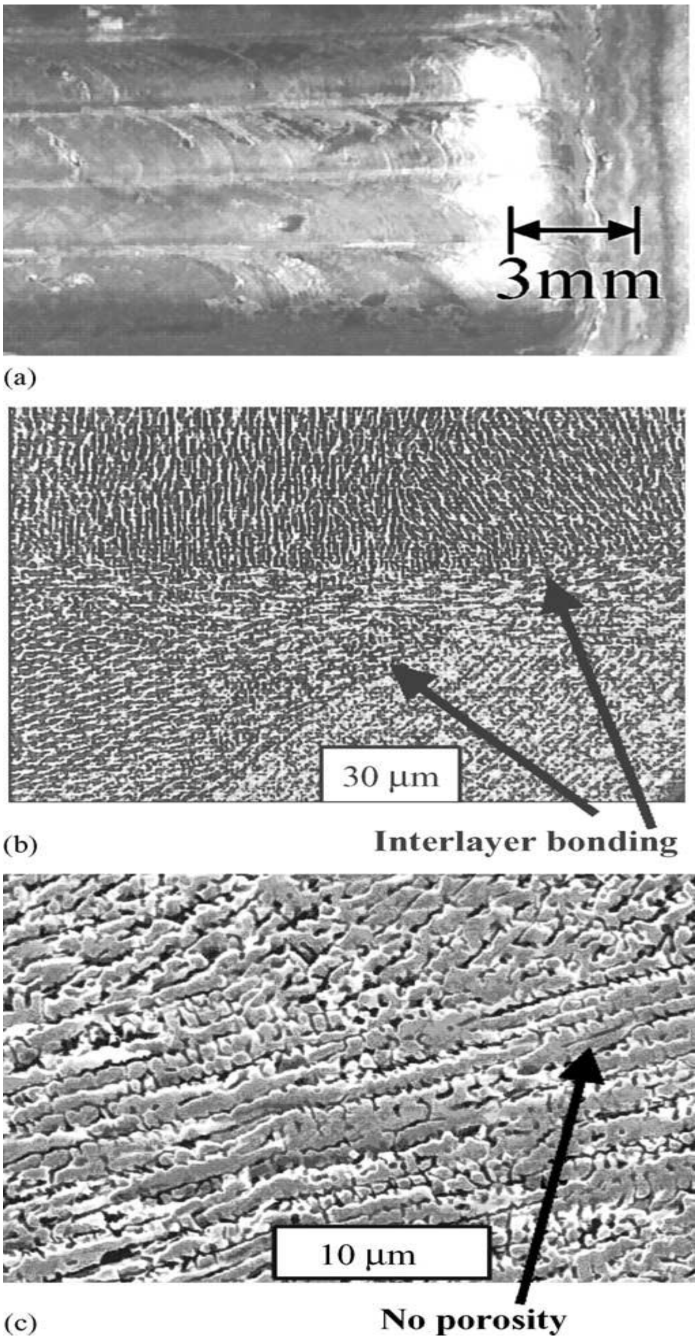


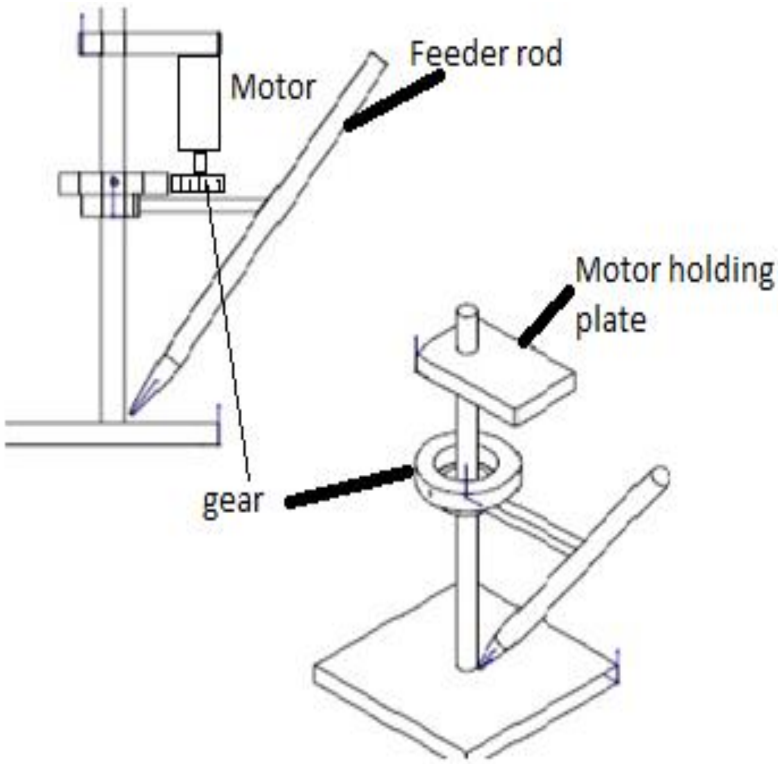
Figure 10: Multilayer made part: (a) front fed, multilayer, top view, (b)transverse sectioned, optical microscopy showing interlayer boundaries and (c) SEM (scanning electron microscopy)

[Reference - Effects of wire feeding direction and location in multiple layer diode laser direct metal deposition Waheed Ul Haq Syed *, Lin Li]

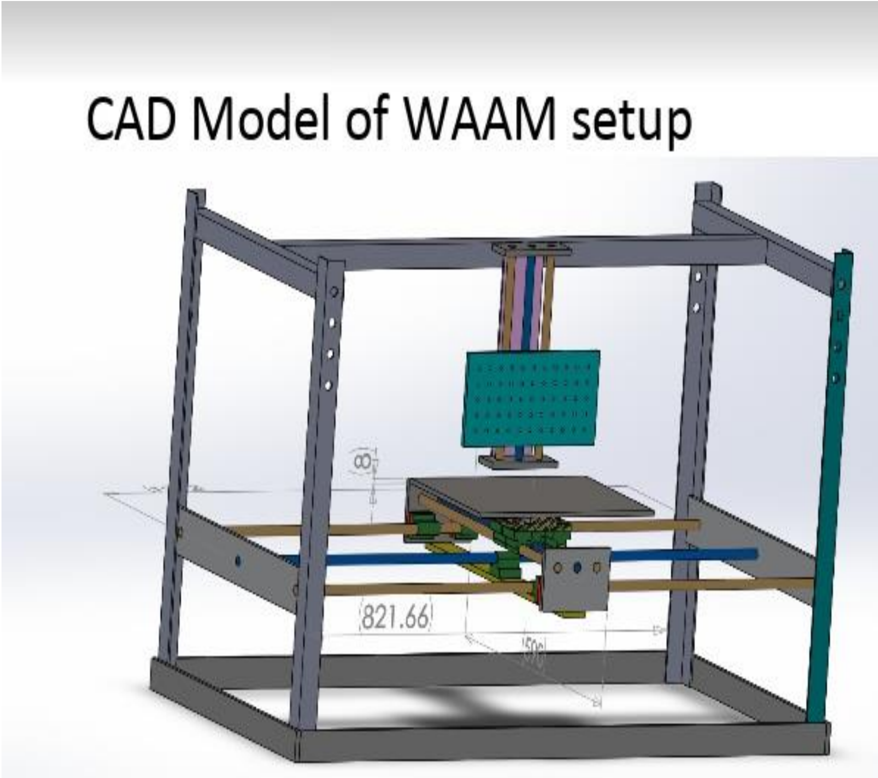
So, to fulfil the requirement of our purpose of keeping the feeder rod always in front face of deposition motion direction: there were two problems

(1) How will feeder rod get the motion direction of deposition?
(2) How to control rotation of feeder rod automatically?

So, to encounter the problem (1) first we will dry run the whole setup of 3-D printer and would fix our mouse sensor on vertical shaft of (figure 11 (b)) so when the horizontal table moves in whatever path this got to have some x and y coordinates. We will collect all the locus coordinates with the help of python program after processing we can get the locus vectors of each continuous points (which will give us the vector of motion of table).



(a)



(b)

Figure 11 : (a) Prototype design and (b) CAD Model of 3-D printer with movable table in X and Y direction and vertical axis is fixed.

Feeder rod will be rotating around vertical rod (in future which will be the tungsten torch) with the help of gear which is attached to motor shaft.   In return motor will be controlled by Motor driver and an Arduino board with PYTHON and Arduino program. After it we will run our setup just some millisecond before actual printer start so that feeder rod will always in front direction of deposition.

3. Experimental details

3.1. Components used in project

Motor: Hybrid 2-phase stepper servo motor W/1000 PPR Encoder with 10kg-cm holding torque, 1.8 step angle, 1000 resolution with class B insulation.

Motor Driver: RMCS-1120

Hybrid Servo Driver with Modbus RTU communication

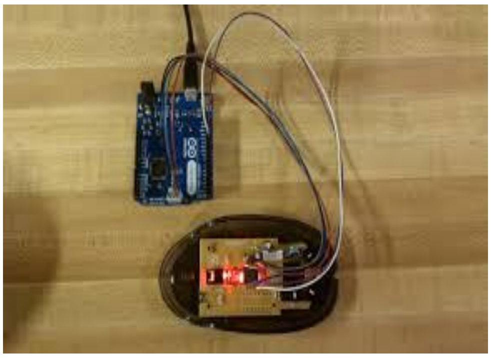
(Max. 50Vdc and 7A per phase)

Arduino Board: The **Arduino Uno** has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 Analog inputs, a 16 MHz crystal oscillator, a USB connection, a

Power jack, an ICSP header, and a reset button Features of the **Arduino UNO**:

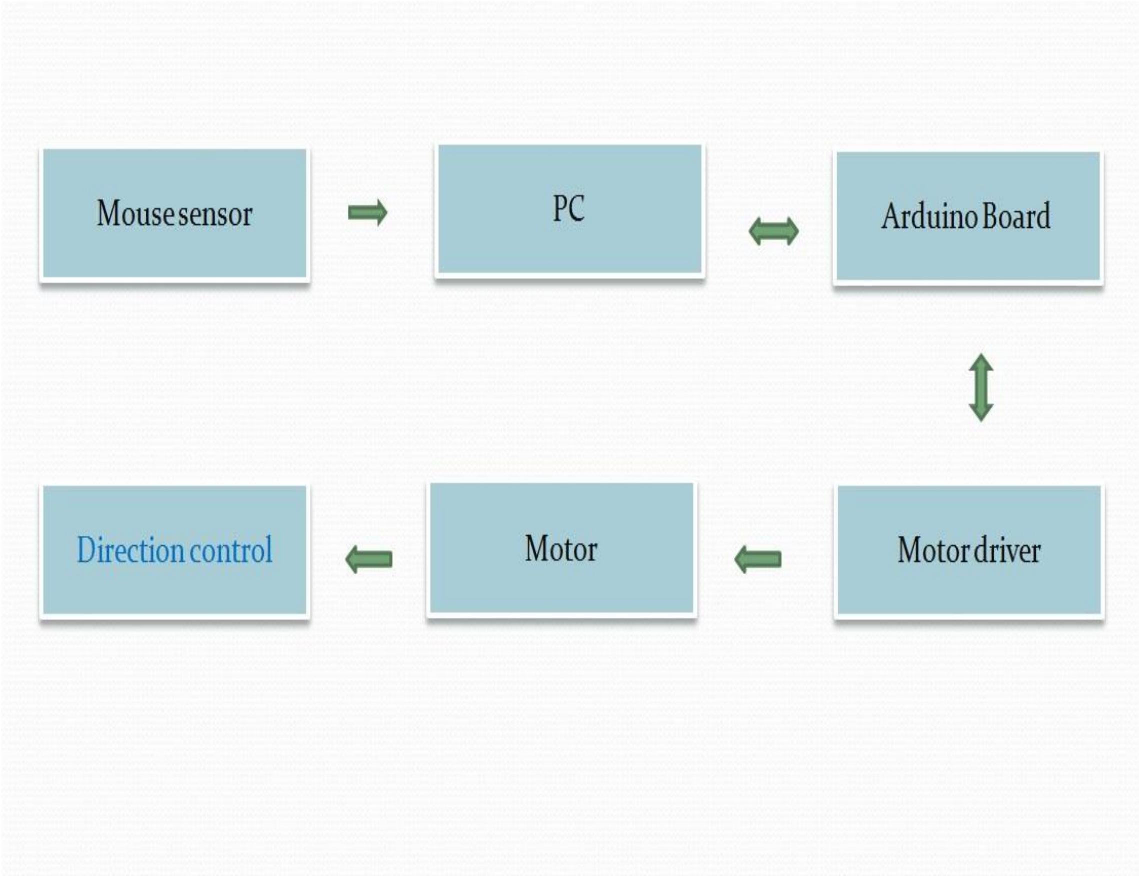
Microcontroller: ATmega328. Operating Voltage: 5V

Mouse sensor: An **optical** mouse usually uses an **infrared** LED (sometimes photodiodes), while a laser mouse uses, you guessed it, a laser to illuminate a surface. Both mice use a complementary-metal-oxide-semiconductor (CMOS) sensor to record the reflection from the laser or LED



Gears: used two different gear of teeth ration 2.4

3.2. FLOW CHART



At very first mouse sensor will travel the path which indirectly collecting coordinates of each point of path and these data will be received by pc where with the help of python programming these coordinates are processed using simple slope method and after getting angles it will show the angles on screens and also send data to Arduino in form of bytes, Arduino reads this data and command sent to motor driver which will be responsible for angular rotation of motor and thus we get Feeder rod angular rotation which is the final result.

4. PROGRAMMING AND LOGIC

This Python program is for getting x and y coordinates of path covered by mouse sensor.

```
import pygame
import matplotlib.pyplot as plt
from scipy import misc
import numpy as np
import pandas as pd
import time

black = [0,0,0]
white = [255, 255, 255]
red = [255, 0, 0]
draw_on = False
last_pos = (0, 0)
color = (255, 128, 0)
radius = 5
font_size = 70

sleep_time = 1

screen = pygame.display.set_mode((600,600))
screen.fill(white)

pygame.font.init()
myfont = pygame.font.SysFont('', font_size)
cord = [[0,0]]

def print_digit(num):
    textsurface = myfont.render(str(num)[1], False, red)
    screen.blit(textsurface,(0,0))


def roundline(srf, color, start, end, radius=1):

    dx = end[0]-start[0]
    dy = end[1]-start[1]
    distance = max(abs(dx), abs(dy))
    for i in range(distance):
        x = int( start[0]+float(i)/distance*dx)
        y = int( start[1]+float(i)/distance*dy)
        # if(cord[-1][0]!=x and cord[-1][1]!=y):
        cord.append([x,y])
        pygame.draw.circle(srf, color, (x, y), radius)
```

Pygame: is a Python wrapper module for the SDL multimedia library. It contains python functions and classes that will allow you to use SDL's support for playing CDROMS, audio and video output, and keyboard, mouse and joystick input.

```
try:
    while True:
        # get all events

        e = pygame.event.wait()


        #clear screen after right click
        if(e.type == pygame.MOUSEBUTTONDOWN and e.button==3):
            screen.fill(white)

        #quit
        if e.type == pygame.QUIT:
            raise StopIteration

        #start drawing after left click
        if(e.type == pygame.MOUSEBUTTONDOWN and e.button!=3):
            color = black
            pygame.draw.circle(screen, color, e.pos, radius)
            draw_on = True

        #stop drawing after releasing left click
        if e.type == pygame.MOUSEBUTTONUP and e.button!=3:
            draw_on = False
            fname = "traced_image.png"
            pygame.image.save(screen, fname)
            # print(cord)


            # cord[0] = cord[0] + [0]
            # for i in range(1,len(cord)):
            #     p1 = cord[i-1][0],cord[i-1][1]
            #     p2 = cord[i][0],cord[i][1]

            # print(lst)
            # df["angle"] = lst
            df = pd.DataFrame(cord,columns=['x','y'])
            df.to_csv('path_vactor.csv',index=None)
            cord = [[0,0]]
```

```
        #start drawing line on screen if draw is true
        if e.type == pygame.MOUSEMOTION:
            if draw_on:
                pygame.draw.circle(screen, color, e.pos, radius)
                roundline(screen, color, e.pos, last_pos,  radius)

            last_pos = e.pos

        pygame.display.flip()

except StopIteration:
    pass


pygame.quit()
```

Matplotlib.pyplot as plt: is a collection of command style functions that make matplotlib work like MATLAB. Each pyplot function makes some change to a figure: e.g., creates a figure, creates a plotting area in a figure, plots some lines in a plotting area, decorates the plot with labels, etc.

SciPy: is a free and open-source Python library used for scientific computing and technical computing.

<u>This Python program is for processing data and sends it to arduino in form of string.</u>

```
import serial
import time
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

def int_to_byte(x):
    s = str(x)
    b = bytearray()
    b.extend(map(ord, s))
    return bytes(list(map(int,list(b))))
def unit_vector(vector):
    return vector / np.linalg.norm(vector)

def angle_between(v1, v2):
    v1_u = unit_vector(v1)
    v2_u = unit_vector(v2)
    return np.arccos(np.clip(np.dot(v1_u, v2_u), -1.0, 1.0))*180/np.pi
def cal_vector(p2,p1):
    x1,y1 = p1
    x2,y2 = p2
    unit_vect = np.array([1,0])

    x0 = x2-x1
    y0 = y2-y1

    angle = (abs(y0))/(abs(x0))
    angle = (np.arctan(angle)*180/np.pi)

    if(x0>=0 and y0>=0):
        angle = angle
    elif(x0<0 and y0>0):
        angle = 180  - angle
    elif(x0<=0 and y0<=0):
        angle = 180 + angle
    elif(x0>0 and y0<0):
        angle = 360 -  angle
    return angle
```

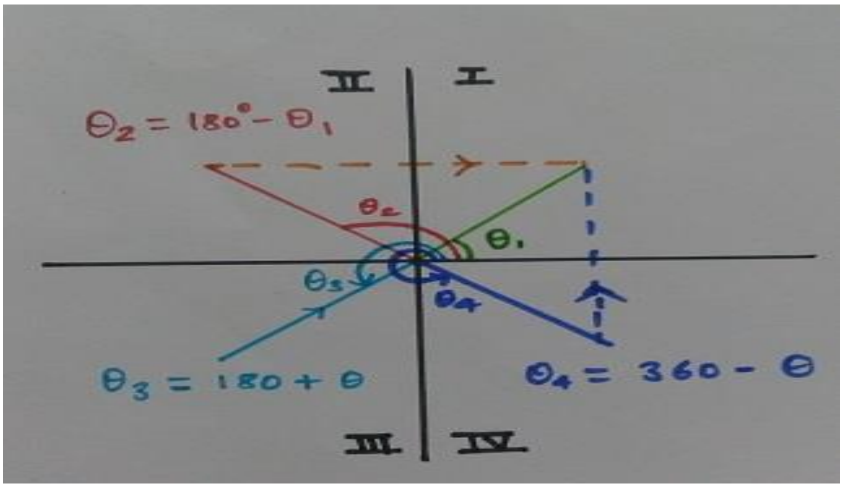


Figure 12: Diagram for transforming other quadrant angle to 1st quadrant

```
# In[520]:


df = pd.read_csv('path_vactor.csv')
df['y'] = -1*df['y'] + 600
# plt.scatter(df['x'],df['y'])


# In[522]:


lst = [0]
px = df['x'].iloc[0]
py = df['y'].iloc[0]
x = [px]
y = [px]

dense = 15

for i in range(dense,len(df),dense):
    p1 = df['x'].iloc[i-dense],df['y'].iloc[i-dense]
    p2 = df['x'].iloc[i],df['y'].iloc[i]
    lst.append(cal_vector(p2,p1))
    x.append(df['x'].iloc[i])
    y.append(df['y'].iloc[i])

df = pd.DataFrame()
df['x'] = x[1:]
df['y'] = y[1:]
df['angle_new'] = lst[1:]


# In[528]:
```

```
obj = serial.Serial('COM3',9600);


# In[527]:


prev = 0
sleep_time = 5
for i in range(1,len(df)):

    current = df['angle_new'].iloc[i]
    value_to_send = current - prev
    prev = current

    value_to_send = round(value_to_send,2)

    data = int_to_byte(value_to_send)
    print(data)
#     for x in list(b):
    obj.write(data)
    obj.write(b'$')

    time.sleep(sleep_time)
```

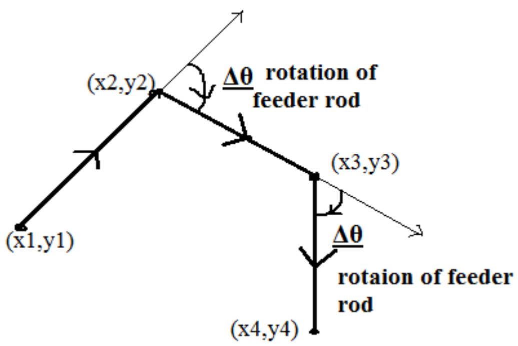


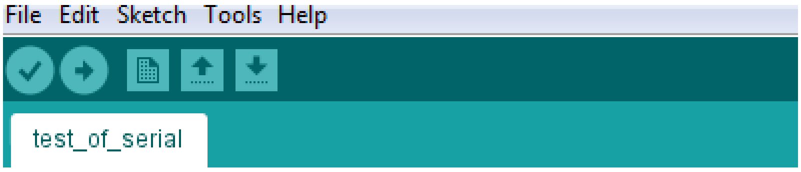
Figure 13: Logic for getting change in angle

Serial Library: Used for communication between the Arduino board and a computer or other devices

Pandas: A powerful Python data analysis toolkit panda is a Python package providing fast, flexible, and expressive data structures designed to make working with “relational” or “labelled” data both easy and intuitive.

NumPy: It is not another programming language but a Python extension module. It provides fast and efficient operations on arrays of homogeneous data. NumPy extends python into a high-level language for manipulating numerical data, similar to MATLAB.

This program is for arduino to send commands to motor driver




```
#include <Stepper.h>

const int stepsPerRevolution =4096;

Stepper myStepper(stepsPerRevolution, 8, 9, 10, 11);

void setup() {
  // put your setup code here, to run once:
  pinMode(8,OUTPUT);
  pinMode(9,OUTPUT);
  pinMode(10,OUTPUT);
  myStepper.setSpeed(10);
  Serial.begin(9600);

}

void turn(String x){
  int rot = x.toInt();
  rot = (rot*12+1)*2.24;
  myStepper.step(rot);
  Serial.println(rot);
//  delay(500);
}


String str="";
```

Stepper.h: This library allows you to control unipolar or bipolar stepper motors.

Mystepper(StepsPerRevolution, 8, 9, 10, 11): This function initializes the stepper library on pins 8 through 11 on Arduino Board.

Void setup: It's an Arduino function which starts the Arduino for once.

Void loop: It's also an arduino function which repeats the process till loop ends.

```
test_of_serial

}

void turn(String x){
  int rot = x.toInt();
  rot = (rot*12+1)*2.24;
  myStepper.step(rot);
  Serial.println(rot);
//  delay(500);
}


String str="";

void loop() {
  // put your main code here, to run repeatedly:

  if(Serial.available()>0){
    char s = Serial.read();
    if(s=='$'){
//        Serial.println(str);
        turn(str);
        str = "";

    }
    else
    str+=s;
  }
}
```

Mystepper.step (rot): This function used for rotating the motor in how much steps where (rot) is a variable.

Serial.println: Prints data to the serial port as human-readable ASCII text. ... Numbers are printed using an ASCII character for each digit. Floats are similarly printed as ASCII digits, defaulting to two decimal places. Bytes are sent as a single character.

5. Result and discussions

The structural and electronic aspects of this project are completed, as it can be seen in the figures below.

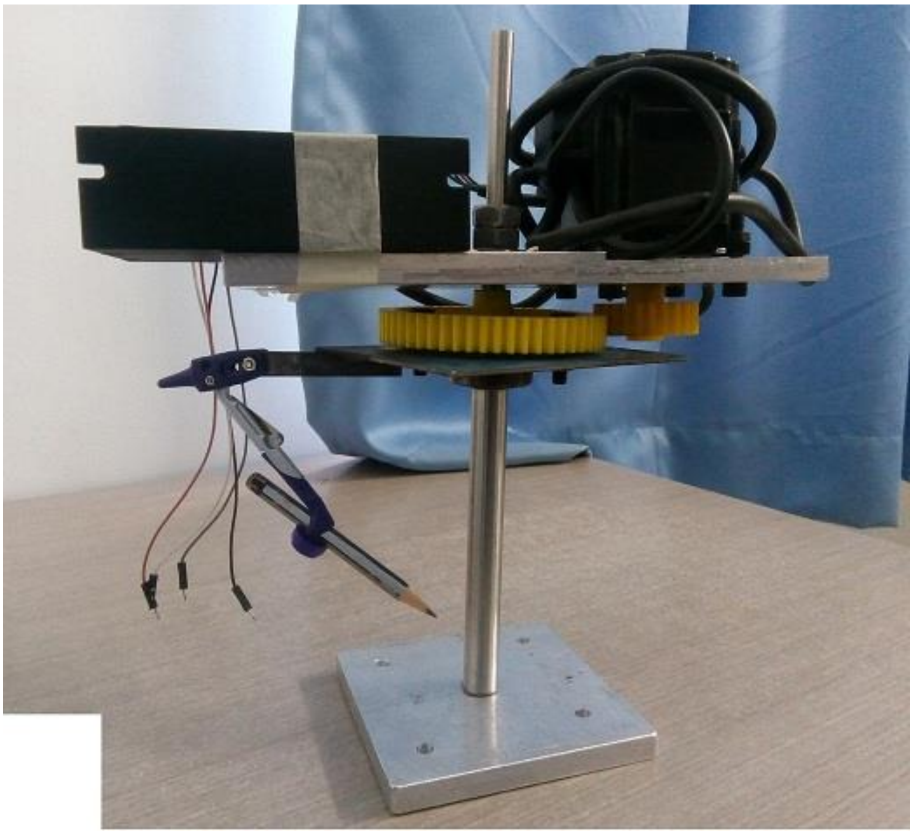


Figure 14: Prototype

Here pencil is used as a feeder rod just for showing purpose and this prototype have many other potential uses which can change the scenario of additive manufacturing.

Step wise process.

➢ Dry running of printer setup and getting x, y coordinates

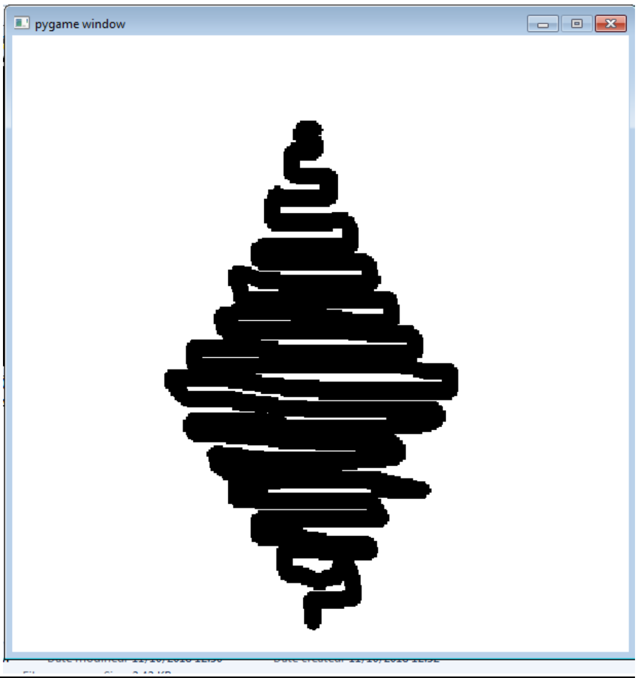
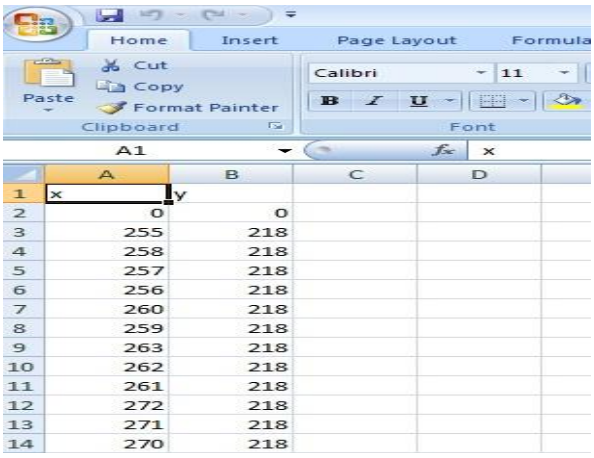


Figure 15: This is drawing which we will get after dry run and each point

➢ Saving x,y coordinates in excel file



x	y
0	0
255	218
258	218
257	218
256	218
260	218
259	218
263	218
262	218
261	218
272	218
271	218
270	218

Figure 16: Excel screenshot of saved x, y coordinates

➢ <u>Calculated angular rotation</u>

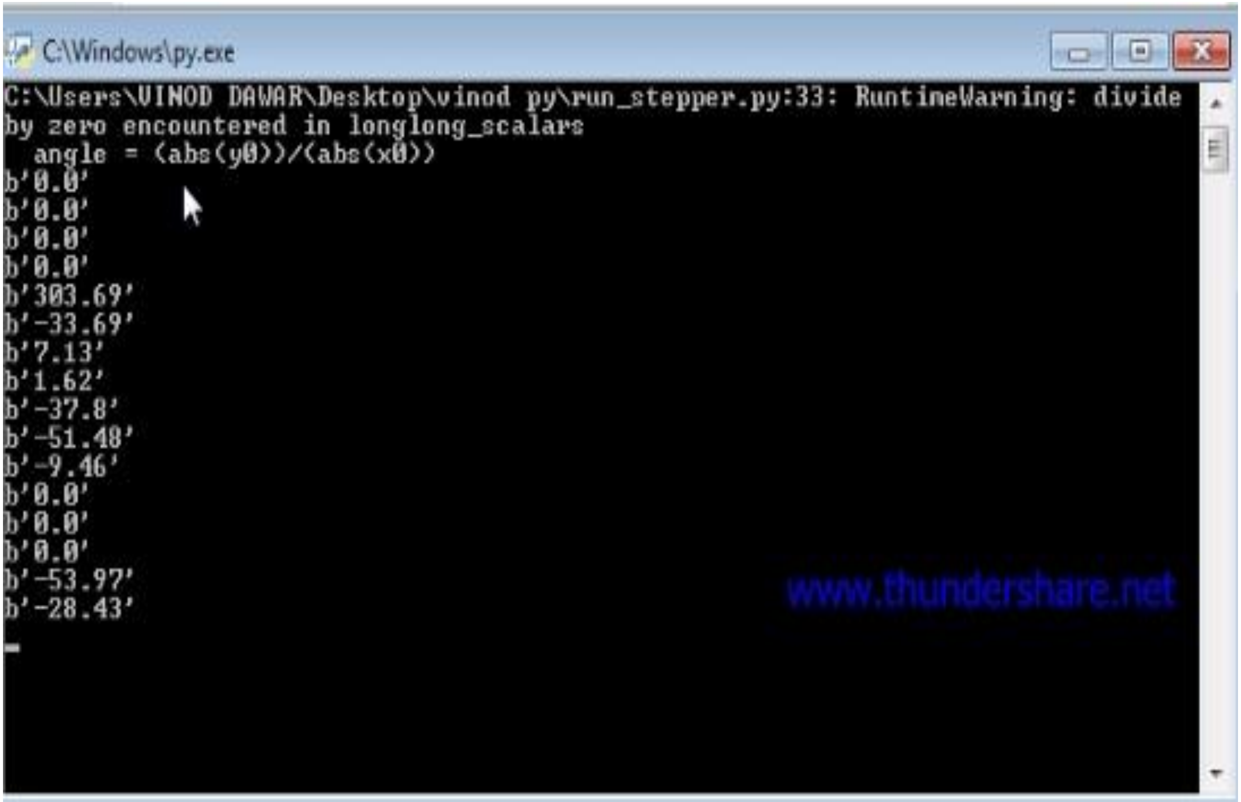


Figure 17: Angular rotation data of feeder rod

These values(figure 17) indicates that how much angular rotation needed to get to next point or to be more precisely direction of next point. And then this data will be sent to Arduino in form of bytes to command motor driver and execute the final result that is rotation of feeder rod in particular angle.

6. Conclusion

- The objective of this BTP is to achieve direction controlled feeding of wire in case of wire arc additive manufacturing.

- We made a prototype to simulate similar scenario for angular directional control.

- In order to achieve our goal we wrote a program in python which will follow coordinates of deposition in WAAM.

- This program will further command motor connected to feeder rod with gears to rotate it at desired angle.

7. Future scope of work

- **Temperature feedback controller**: In place of rotating rod if we use Pyrometer (figure 18) such that we can get Temperature of welding bead at desired location from which we can use it as feedback controller for example if the temperature is less than required, it will send data to processor and thus simultaneously power input from welding torch can be increased to match the requirement and vice versa.

Figure 18: Pyrometer

- **Bead shape imaging:** Also, we can use laser imaging (figure 19)to detect the Bead shape during welding process is going on so that we can make changes in parameters to get a uniform bead.

Figure 19: Shape sensor

REFERENCES

- Effects of wire feeding direction and location in multiple layer diode laser direct metal deposition Waheed Ul Haq Syed *, Lin Li

- DuPont J et al (1995) Thermal efficiency of arc welding processes. Weld J Incl Weld Res Suppl 74:406s

- Heralic A (2012) Monitoring and control of robotized laser metal-wire deposition. Doctoral thesis, Chalmers University of Technology

- Stenbacka N, et al (2012) Review of Arc Efficiency Values for Gas Tungsten Arc Welding IIW Commission IV-XII-SG212,Intermediate Meeting, BAM, Berlin, Germany, 18–20 April

- Wikipedia and Google images source