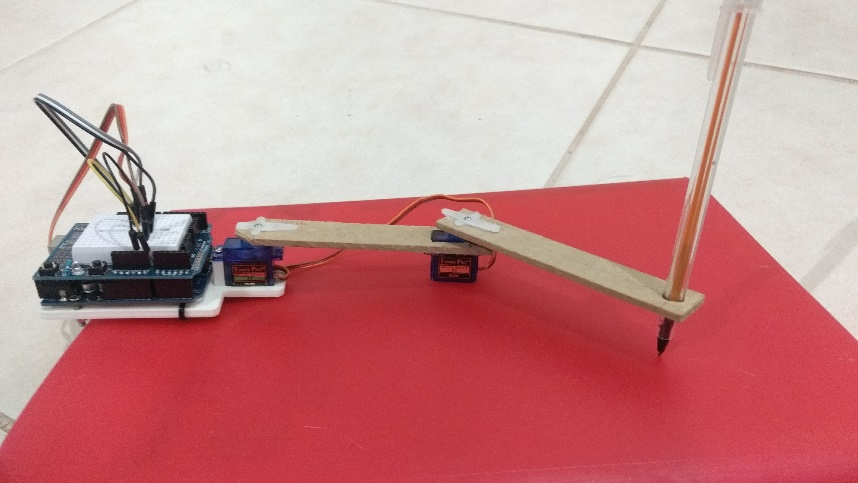
1. Introdução
2. Projeto do robô

2.1 Estrutura física

O robô Scara didático foi construído com os seguintes componentes:

* Placa Arduíno Uno;
* Servomotor SG90, usado em aeromodelismo;
* Links de MDF com 120mm. Elo1 99mm entre eixos e Elo2 100 mm entre eixos;
* Fonte de alimentação externa (bateria alcalina 9V);
* Base de fixação com garra para acoplamento em superfícies planas;
* Órgão terminal: caneta esferográfica;



2.2 Software

O código a seguir, intitulado ServoSimples.ino, foi embarcado no micro controlador através da IDE oficial da plataforma de prototipagem Arduíno. Sua função consiste basicamente em zerar o robô quando este for inicializado, enviando o efetuador terminal para a posição (0,200) e estabelecer uma comunicação com a porta serial para receber valores referentes aos ângulos que devem ser enviados aos servomotores.

#include <Servo.h>

Servo ombro;

Servo junta;

void setup() {

Serial.begin(9600);

//definição dos pinos

ombro.attach(10);

junta.attach(11);

initprogram();

}

//variáveis auxiliares

int a, pos1 = 0;

int b, pos2 = 0;

int comunica = 0;

void loop() {

comunica = Serial.read();

delay(40);

if((Serial.available()>1)&&(comunica)){

//obtendo dados

a = Serial.read();

delay(25);

b = Serial.read();

delay(25);

ombro.write(a);

junta.write(b);

}

}

void initprogram(){

int au1 = pos1, au2 = pos2;

for(au1 = pos1; au1!= 95; au1++){

ombro.write(au1);

delay(10);}

for(au2 = pos2; au2!= 160; au2++){

junta.write(au2);

delay(10);}

pos1 = ombro.read();

pos2 = junta.read();

}

1. Modelagem cinemática
   1. Modelagem Cinemática Direta (Denavit-Hartemberg)

A cinemática direta do robô, que fornece as coordenadas cartesianas do efetuador terminal em função das variáveis de juntas, foi obtida, conforme especificação do trabalho, segundo o algoritmo de Denavit-Hartemberg para modelagem de mecanismos:

PASSO1: identificação das juntas e posicionamento dos sistemas de coordenadas para cada elo



PASSO2: tabela com os parâmetros básicos de DH

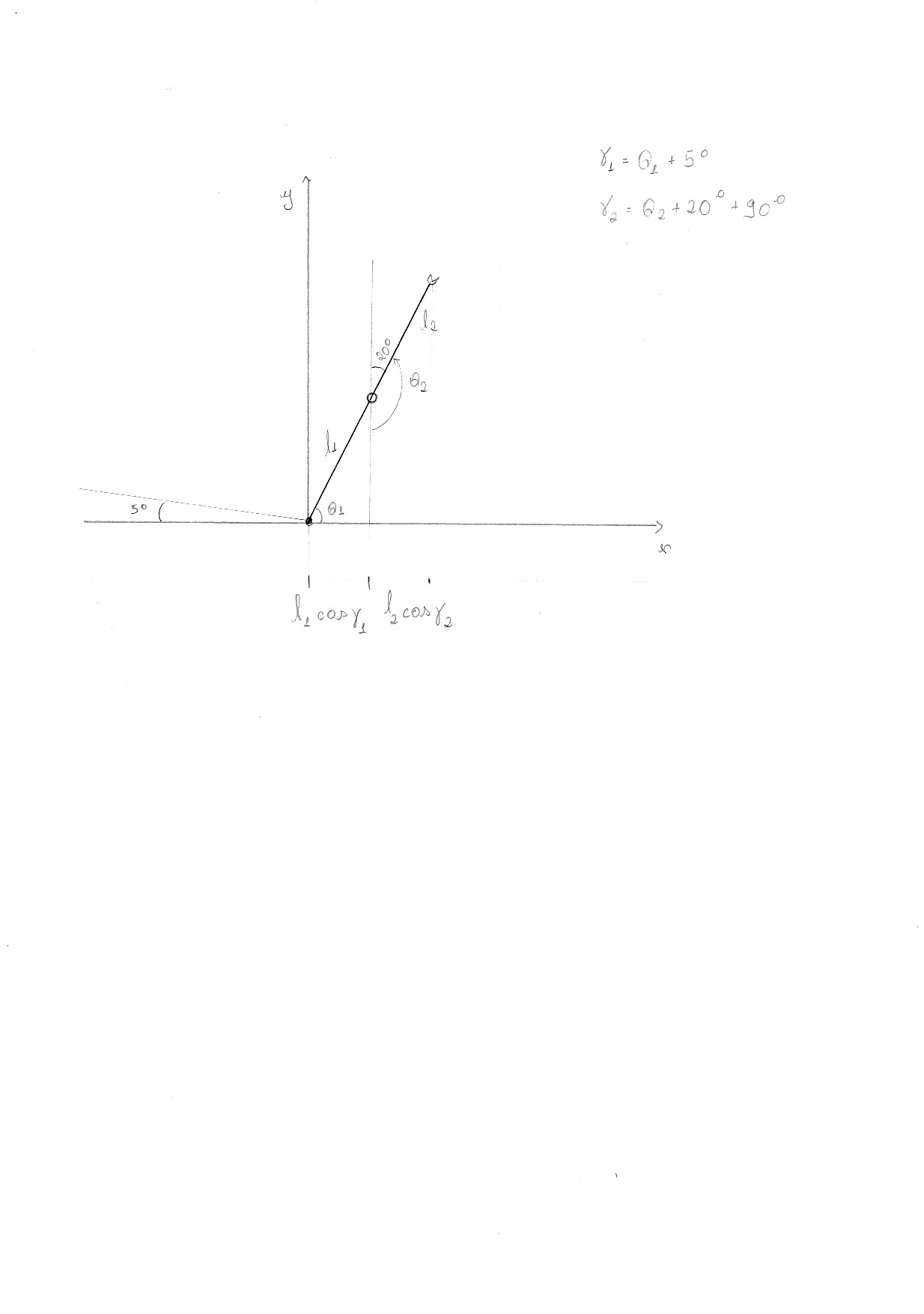
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | a | α | d | ɵ |
| junta1 | 99 | 0 | 0 | ɵ1+5 |
| junta2 | 100 | 0 | 0 | ɵ2+20 |

PASSO 3: matrizes homogêneas para cada transformação do sistema de coordenada

PASSO 4: transformação da base para garra pela multiplicação das transformações para cada link

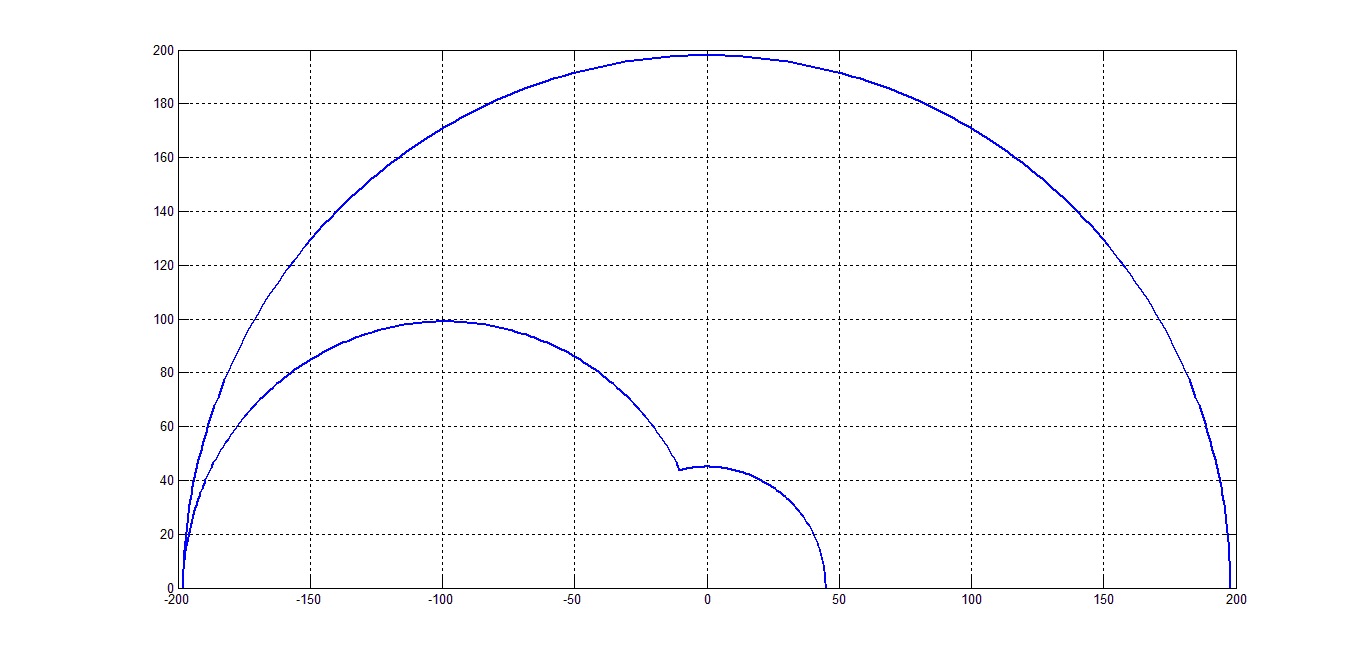
As variáveis cartesianas correspondem aos dois primeiros elementos da última coluna da matriz homogenia final, portanto a cinemática direta fica definida pelas equações:

* 1. Modelagem Cinemática Inversa (Geométrico)



Portanto a cinemática inversa fica definida pelas equações:

* 1. Volume de trabalho



A área modelada do volume de trabalho é limitada por três semicírculos descritos:

Semicirculo1:

Semicirculo2:

Semicírculo 3:

Portanto compreende os pontos (x,y) tal que e

1. Etapas de desenvolvimento do Trabalho
   1. Fase 1

Nesta etapa programa move o efetuador terminal para posição (x,y) acionando um link e depois o outro, seguindo uma interpolação no domínio das juntas que gera movimentos suaves no domínio cartesiano.

Para tanto foi necessário criar em MATLAB uma função para verificar se o ponto (x,y) solicitado está dentro do volume de trabalho do robô. Caso o ponto não atenda a essa condição, o programa é abortado. Foi necessário também codificar as equações que descrevem a cinemática inversa do robô. Tal função recebe as coordenadas do ponto que se deseja alcançar e os ângulos atuais das juntas, para que seja possível calcular os ângulos que deverão ser impostos aos atuadores baseando-se na melhor opção de trajetória entre o ponto atual e o ponto desejado.

Tendo definido as variáveis de juntas, são criados vetores com os valores intermediários que devem ser enviados aos servos para que esses façam movimentos suaves, a fim de se evitar picos de aceleração e choques mecânicos. Os ângulos são então enviados ao robô atreves da comunicação serial entre o MATLAB e arduíno.

* 1. Fase 2

O programa move o efetuador terminal para a posição (x,y), movendo os dois Links ao mesmo tempo com a mesma velocidade.

Utilizando as funções de verificação do volume de trabalho e cinemática inversa criadas na fase anterior, essa etapa avalia a distância angular que cada motor terá que descrever. É então definida uma velocidade para ambos. Em seguida os pontos intermediários são encontrados para cada atuador, considerando esta velocidade.

Para que seja possível utilizar o mesmo programa anteriormente embarcado na placa arduíno, as trajetórias enviadas para cada motor devem ter o mesmo tamanho, portanto o motor que inicia o movimento mais próximo do seu destino final, continua a receber o valor final da trajetória até que o outro motor termine seu movimento.

* 1. Fase 3

O efetuador alcança a posição (x,y) movendo os dois links ao mesmo tempo, com velocidades proporcionais terminando o movimento juntos.

Novamente englobando os itens desenvolvidos nas etapas anteriores, nessa etapa uma variável de proporcionalidade, obtida através do modulo da diferença entre o ângulo final e inicial do ombro, impõem o número de pontos intermediários para todos os atuadores. Dessa forma terão suas trajetórias completadas simultaneamente.

1. Conclusão
2. Referências Bibliográficas