دانشگاه تهران

پردیس دانشکده‌های فنی

دانشکده مهندسی مکانیک

دینامیک

تمرین کامپیوتری دینامیک

نام نویسندگان:

نام گروه:

بهار ۱۴۰۳

دانشگاه تهران
پردیس دانشکده‌های فنی
دانشکده مهندسی مکانیک
دینامیک

چکیده

مکانیزم مورد بررسی در این گزارش مکانیزم برگشت سریع (Quick Return Mechanism) است. این مکانیزم در مواردی که در آن حرکت برگشتی سریع‌تر از حرکت رفت است، استفاده می‌شود. این مکانیزم به طور عمده در دستگاه‌های شکل‌دهی، برش و خم‌کاری استفاده می‌شود. برخی از کاربردهای این مکانیزم عبارتند از دستگاه‌های شکل‌دهی، دستگاه‌های سلات‌زنی و پلنر، پرس‌های مکانیکی و اکتوئاتورهای مکانیکی. همچنین در برخی اجزای موتورهای احتراق داخلی چرخشی نیز از مکانیزم برگشت سریع استفاده می‌شود. این مکانیزم بهبود کارایی و سرعت در دستگاه‌های صنعتی و مکانیکی فراهم می‌کند. [1]

فهرست

مقدمه

در این گزارش مکانیزم به دو صورت حل تحلیلی (متلب) و با استفاده از نرم افزار آدامز بررسی شده است. بررسی آدامز قسمت جداگانه ای به خود اختصاص نداده است و در هر قسمت پاسخ متلب و آدامز مورد بررسی قرار گرفته اند.

زوایا

از طریق هندسه مکانیزم میتوان چهار مجهول و چهار معادله به دست آورد و با حل آن ها به رابطه θ_1 و θ_3 بر حسب θ_2 خواهیم رسید. هندسه مکانیزم به صورت زیر است:

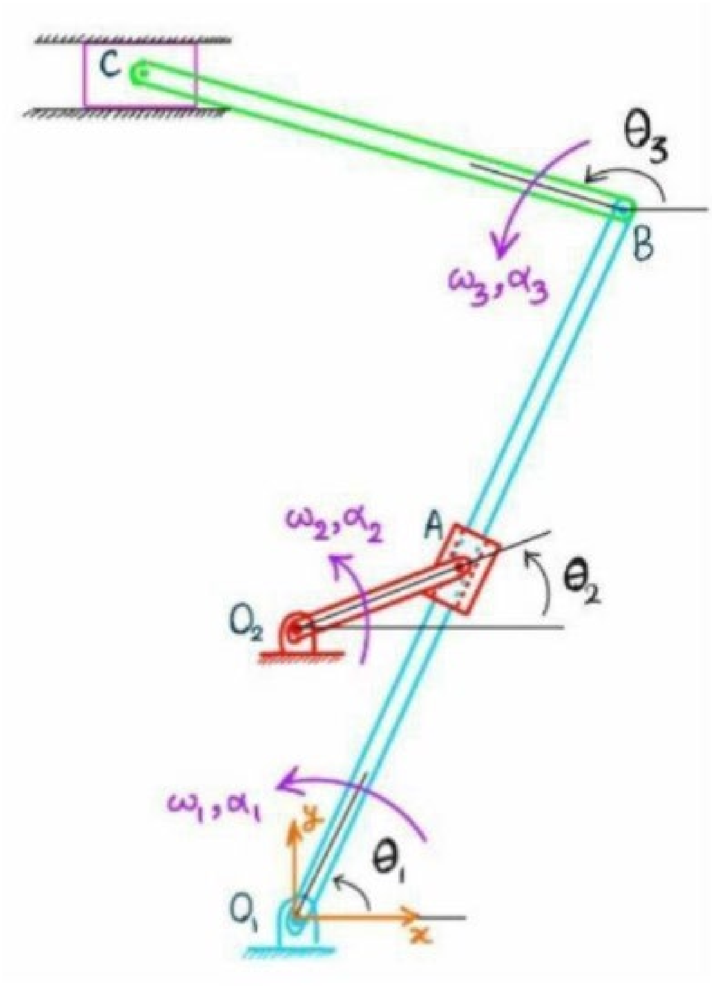


Figure 1: Main mechanism

معادلات به شرح زیر میباشد:

$$O_1O_2 + AO_2.\sin(\theta_2) = AO_1.\sin(\theta_1)$$

$$AO_2.\cos(\theta_2) = AO_1.\cos(\theta_1)$$

$$BO_1.\sin(\theta_1) + BC.\sin(\theta_3) = CO_{1_y}$$

$$BO_1.\cos(\theta_1) + BC.\cos(\theta_3) - CO_{1_x} = 0$$

مجهولات عبارت اند از:

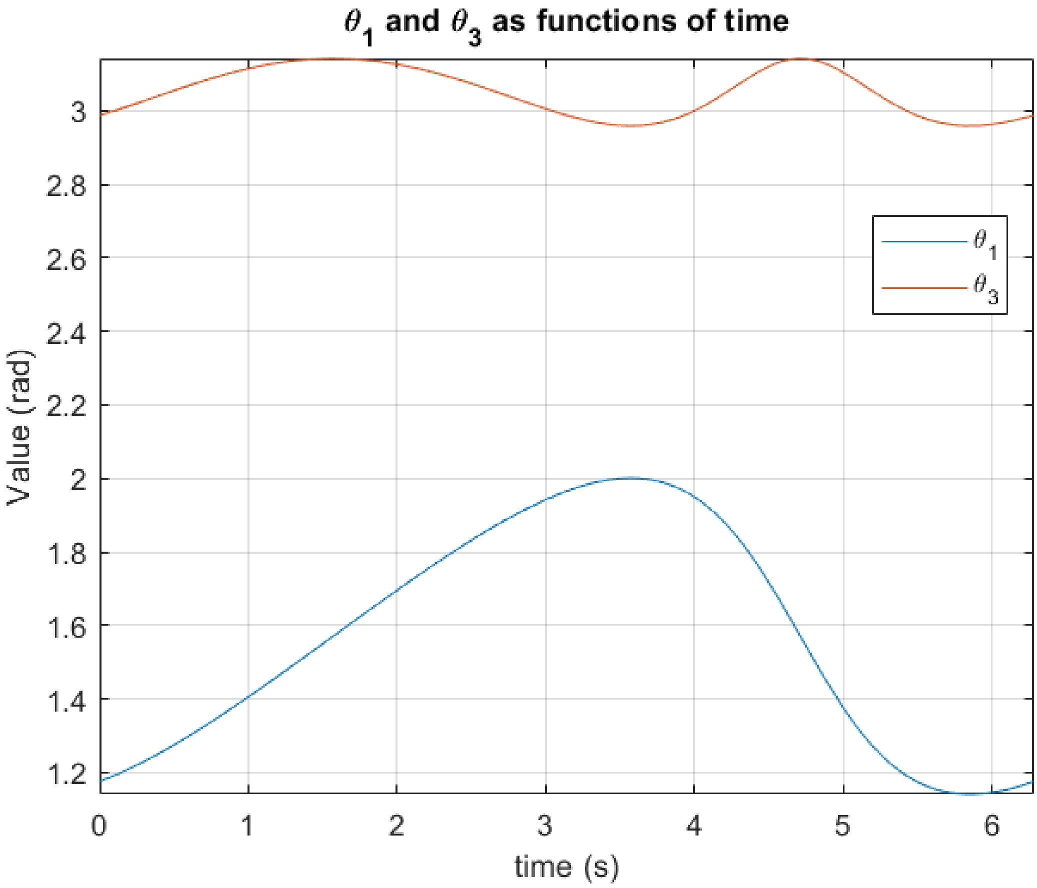
$$\theta_1 , \theta_2 , AO_1 , CO_{1_x}$$

با استفاده از متلب (solve for theta) زوایا به دست می‌آید:

$$\theta_1 = -2\tan^{-1}\left(\frac{5cos(\theta_2) - \sqrt{120sin(\theta_2) + 169}}{5sin(\theta_2) + 12}\right)$$

$$\theta_3 = \pi - \sin^{-1}\left(2sin\left(2\tan^{-1}\left(\frac{5cos(\theta_2) - \sqrt{\{120sin(\theta_2) + 169\}}}{5\backslash sin(\theta_2) + 12}\right)\right) + 2\right)$$

نمودار های رسم شده در متلب:



سرعت

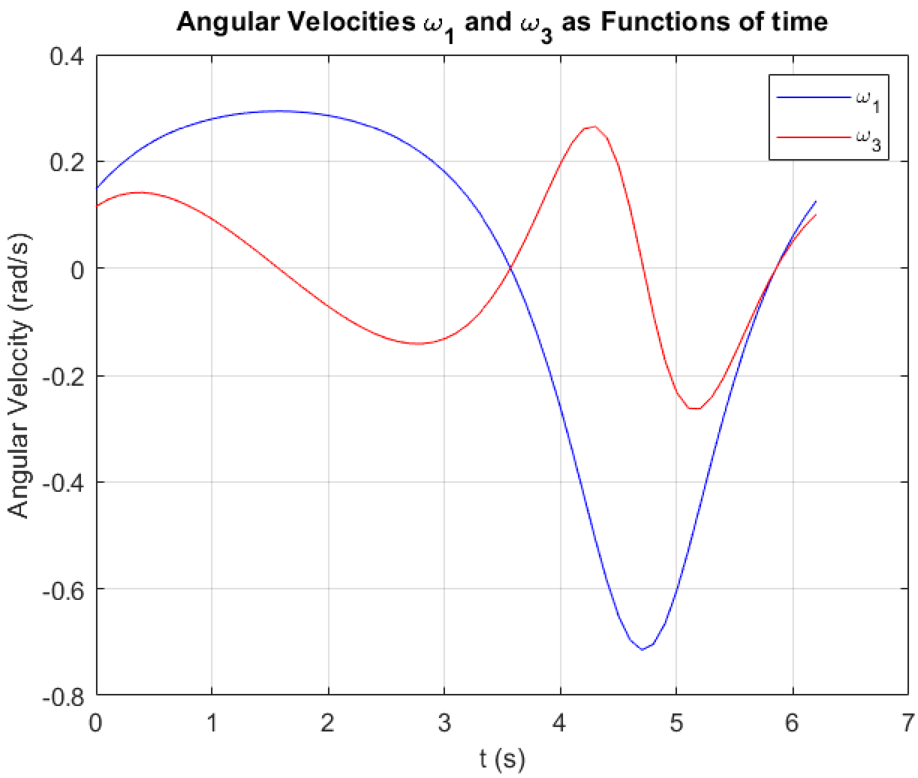
برای محاسبه سرعت‌های زاویه ای و سرعت لینک C میتوانیم از روابط مشتق بگیریم.

سرعت لینک C برابر با مشتق CO_{1_x} میباشد. به علت طولانی بودن روابط، در پیوست مربوط متلب آورده شده است. برای برحسب زمان شدن روابط نیاز به انجام کار خاصی نیست:

$$\omega_1 = \frac{d\theta_1}{dt} = \frac{d\theta_1}{d\theta_2} \times \frac{d\theta_2}{dt} = \frac{d\theta_1}{d\theta_2} . \omega_2$$

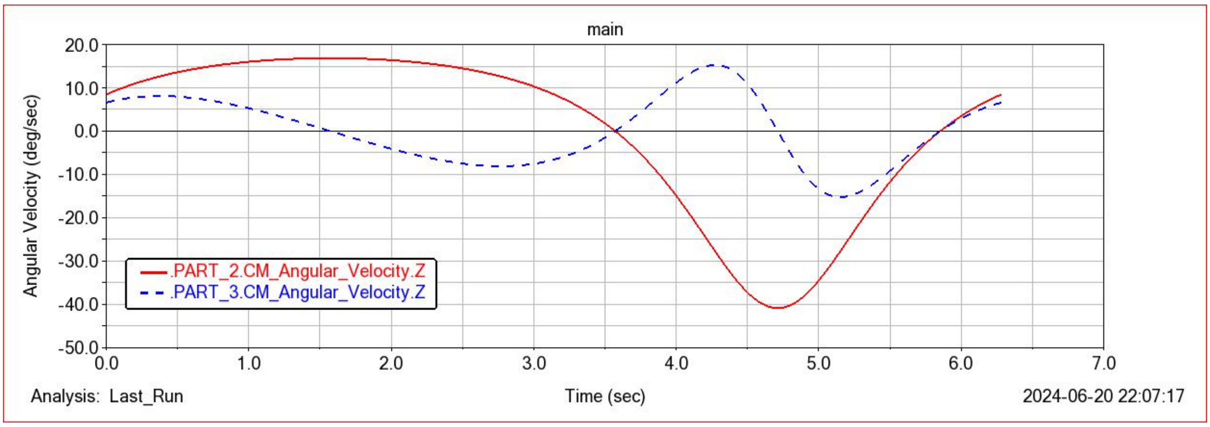
از آن جا که ω_2 برابر ۱ رادیان بر ثانیه میباشد مشتق ما خود به خود برابر با سرعت زاویه ای میشود.

نمودار سرعت‌های زاویه ای به شرح زیر میباشد:

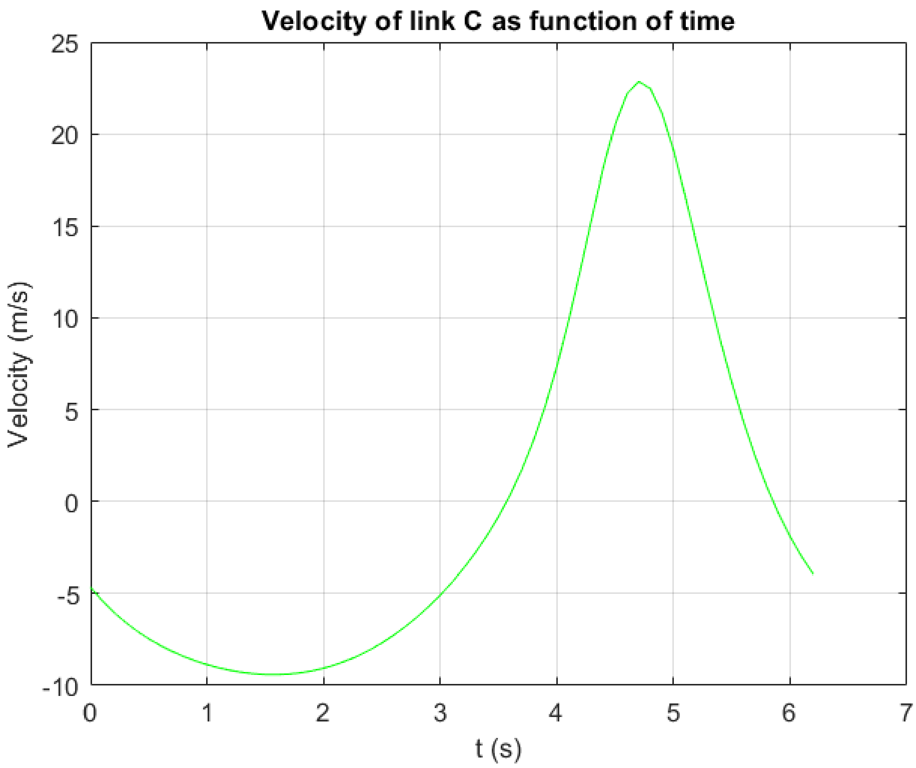


نتایج به دست آمده از طریق حل تحلیلی با شبیه سازی آدامز قابل مقایسه میباشد.

نتایج آدامز برای سرعت‌های زاویه ای:



سرعت لینک C:



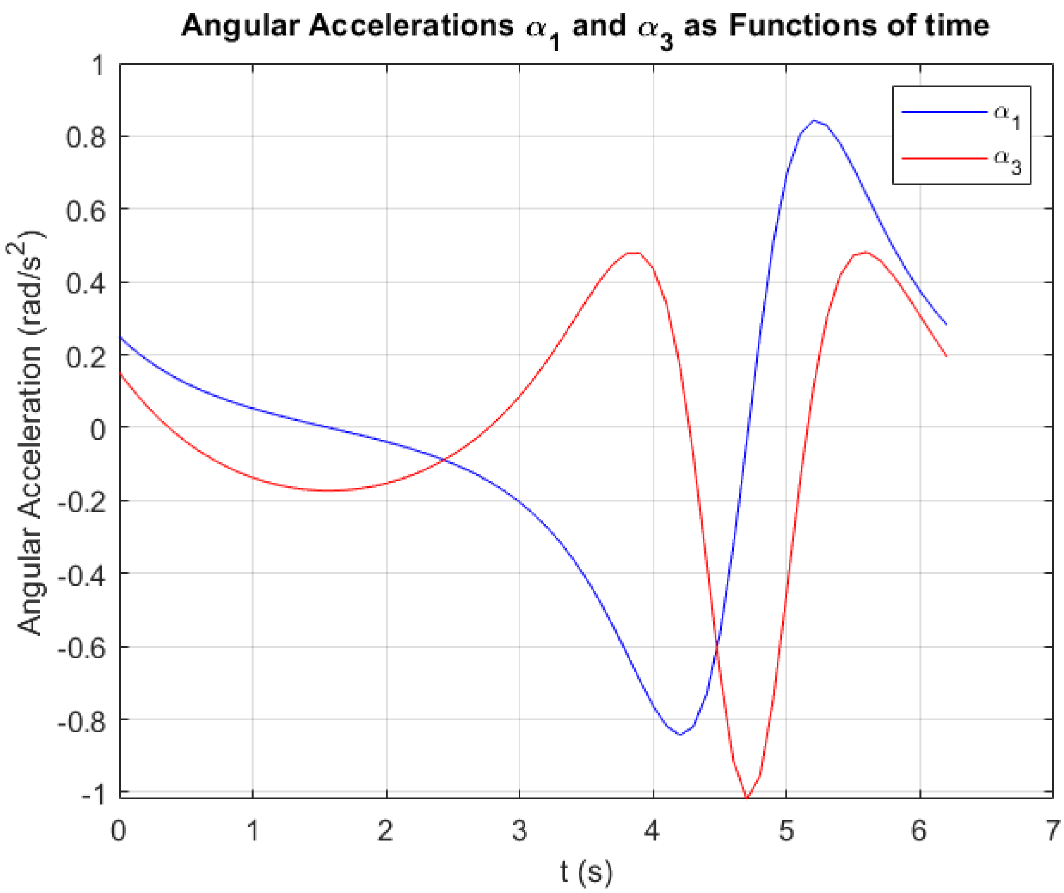
برای سرعت لینک C در آدامز:

مقادیر یکسان بوده و حل تحلیلی مورد تایید میباشد.

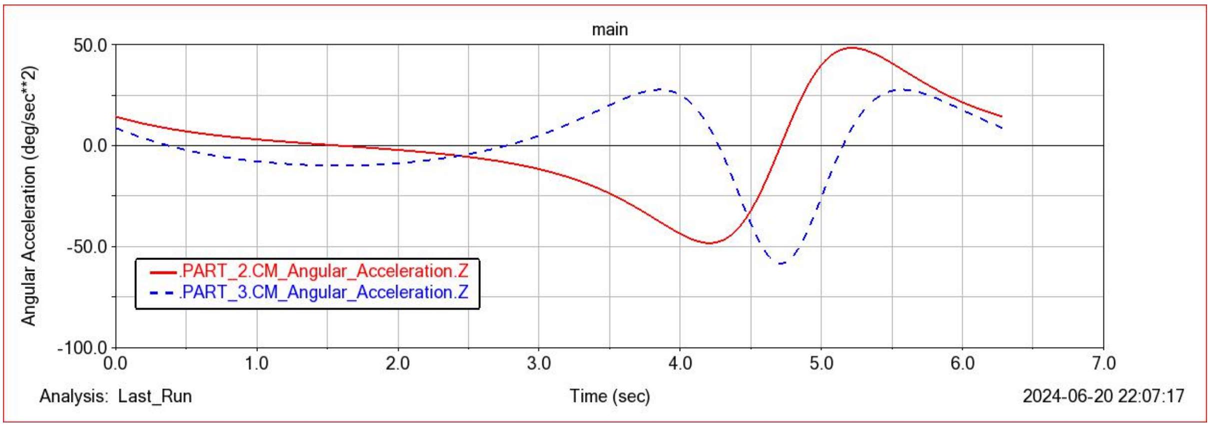
شتاب

برای به دست آوردن شتاب‌ها نیز به همین ترتیب عمل میکنیم:

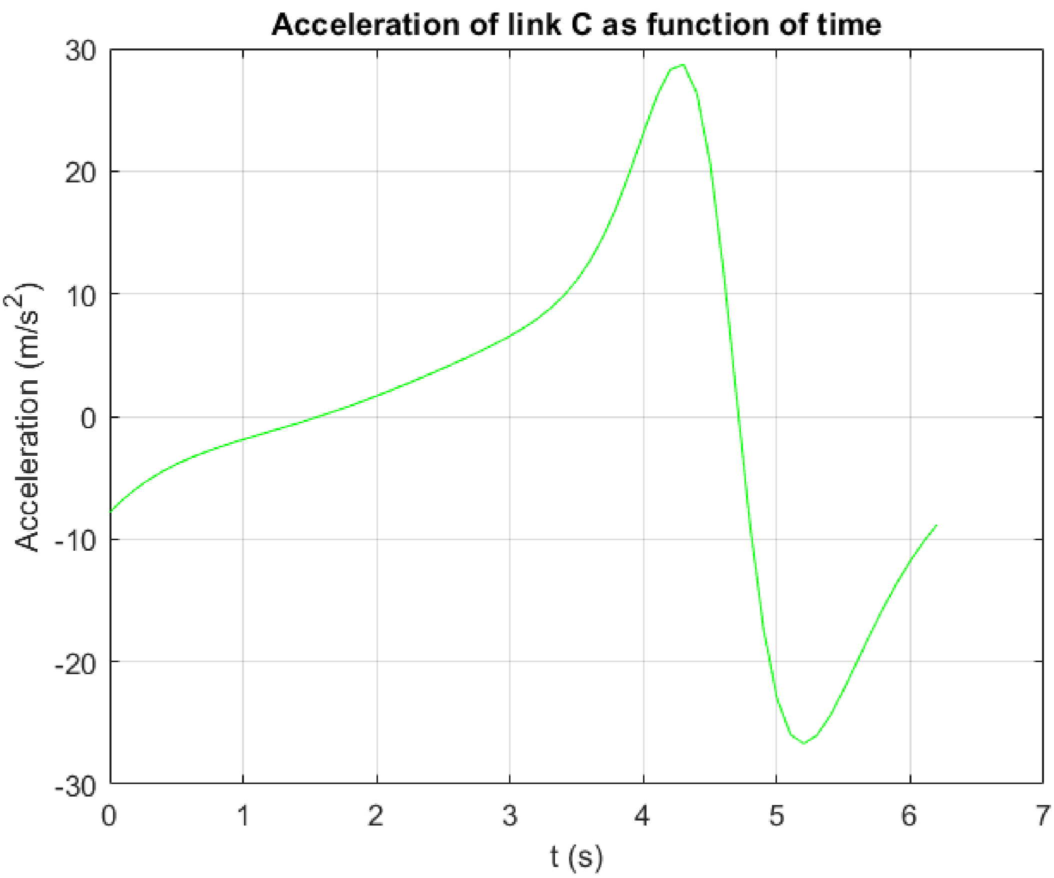
شتاب‌های زاویه‌ای:



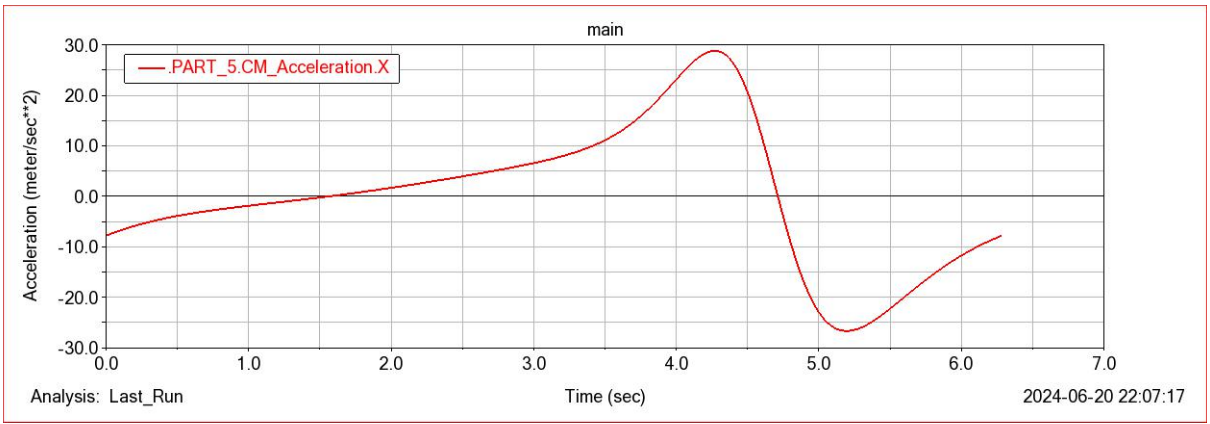
نتایج به دست امده از آدامز:



و شتاب لینک C با مشتق گرفتن از سرعت آن:



نتیجه به دست آمده از آدامز:



در این قسمت نیز نتایج یکسال بوده و حل تحلیلی و شبیه ساز مورد تایید قرار میگیرند.

انیمیشن مکانیزم

انیمیشن حرکت مکانیزم در متلب در ویدیو VID1 قابل مشاهده است. این انیمیشن از طریق نمودار متلب با پاک کردن و مجدد کشیدن مکان جدید لینک ها کشیده شده است.

همچنین انیمیشن آن در نرم افزار ادمز نیز در VID2 قابل مشاهده است.

بحث و بررسی نتایج

همانطوری که از نتایج به دست آمده از حل تحلیلی و حل شبیه ساز مشخص است، این مکانیزم در نیمه دوم حرکت خود سرعت و شتاب بیشتری را تجربه میکند. علت نام گذاری این مکانیزم به همین موضوع بازمیگردد. این سرعت و شتاب برای تمام لینک ها (به غیر از لینک متصل به موتور که سرعت ثابت دارد) صادق است اما خروجی مکانیزم ینی لینک C به صورت واضح این تغییر سرعت در نیم سیکل اول دوم را تجربه میکند.

پیوست

کدهای متلب پیوست میباشد. کامنت گذاری به خوبی انجام شده است و کد هر بخش مشخص است.

```
clc;
clear;
%%
% solve for theta


% init info
syms theta1 theta2 theta3 ao1 co1x;
o1o2 = 12;
ao2 = 5;
```

```
bo1 = 32;

bc = 16;

co1y = 32;


% equations for solve

eqns = [

    o1o2 + ao2 * sin(theta2) == ao1 * sin(theta1);

    ao2 * cos(theta2) == ao1 * cos(theta1);

    bo1 * sin(theta1) + bc * sin(theta3) == co1y;

    bo1 * cos(theta1) + bc * cos(theta3) - co1x == 0;

    ];


slv = solve(eqns, [theta1, theta3, ao1, co1x]);


theta1_expr = slv.theta1(2);

theta3_expr = slv.theta3(2);

co1x_expr = slv.co1x(2);

ao1_expr = slv.ao1(2);


% disp(['theta1 = ', char(simplify(theta1_expr))]);

% disp(['theta3 = ', char(simplify(theta3_expr))]);


% Plot theta1 and theta3 as functions of time
```

```
figure;
fplot(theta1_expr, [0, 2 * pi]);
hold on;
fplot(theta3_expr, [0, 2 * pi]);
xlabel('time (s)');
ylabel('Value (rad)');
legend('\theta_1', '\theta_3');
title('\theta_1 and \theta_3 as functions of time');
grid on;
legend("Position", [0.77274, 0.64579, 0.10964, 0.10214])
%%
% velocities


omega1 = diff(theta1_expr);
omega3 = diff(theta3_expr);
vc = diff(co1x_expr);


% Plot omega1 and omega3 as functions of time
theta2_range = 0:0.1:2 * pi;
omega1_values = subs(omega1, theta2, theta2_range);
omega3_values = subs(omega3, theta2, theta2_range);
vc_values = subs(vc, theta2, theta2_range);
```

```
figure;

plot(theta2_range, omega1_values, 'b', 'DisplayName', '\omega_1');

hold on;

plot(theta2_range, omega3_values, 'r', 'DisplayName', '\omega_3');

xlabel('t (s)');

ylabel('Angular Velocity (rad/s)');

title('Angular Velocities \omega_1 and \omega_3 as Functions of time');

legend;

grid on;

figure;

plot(theta2_range, vc_values, 'g', 'DisplayName', 'V_C');

xlabel('t (s)');

ylabel('Velocity (m/s)');

title('Velocity of link C as function of time');

grid('on');

%%

% accelerations


alpha1 = diff(omega1);

alpha3 = diff(omega3);

ac = diff(vc, theta2);


% Plot alpha1 and alpha3 as functions of time
```

```
theta2_range = 0:0.1:2 * pi;
alpha1_values = subs(alpha1, theta2, theta2_range);
alpha3_values = subs(alpha3, theta2, theta2_range);
ac_values = subs(ac, theta2, theta2_range);


figure;
plot(theta2_range, alpha1_values, 'b', 'DisplayName', '\alpha_1');
hold on;
plot(theta2_range, alpha3_values, 'r', 'DisplayName', '\alpha_3');
xlabel('t (s)');
ylabel('Angular Acceleration (rad/s^2)');
title('Angular Accelerations \alpha_1 and \alpha_3 as Functions of time');
legend;
grid on;
figure;
plot(theta2_range, ac_values, 'g', 'DisplayName', 'a_C');
xlabel('t (s)');
ylabel('Acceleration (m/s^2)');
title('Acceleration of link C as function of time');
grid('on');
%%
% animation
```

```
theta1_values = subs(theta1_expr, theta2, theta2_range);

theta3_values = subs(theta3_expr, theta2, theta2_range);

co1x_values = subs(co1x_expr, theta2, theta2_range);

ao1_values = subs(ao1_expr, theta2, theta2_range);


for i = 1:length(theta2_range)
    pause(0);
    clf();


    plot([0, bo1 * cos(theta1_values(i))], [0, bo1 * sin(theta1_values(i))]);
    hold on;
    plot([bo1 * cos(theta1_values(i)), co1x_values(i)], [bo1 * sin(theta1_values(i)), co1y]);
    hold on;
    plot([0, ao1_values(i) * cos(theta1_values(i))], [o1o2, ao1_values(i) *
sin(theta1_values(i))]);
    hold on;
    xlim([-30, 20]);
    ylim([-5, co1y + 5]);
end
```

منابع

[1] https://testbook.com/mechanical-engineering/quick-return-mechanism