

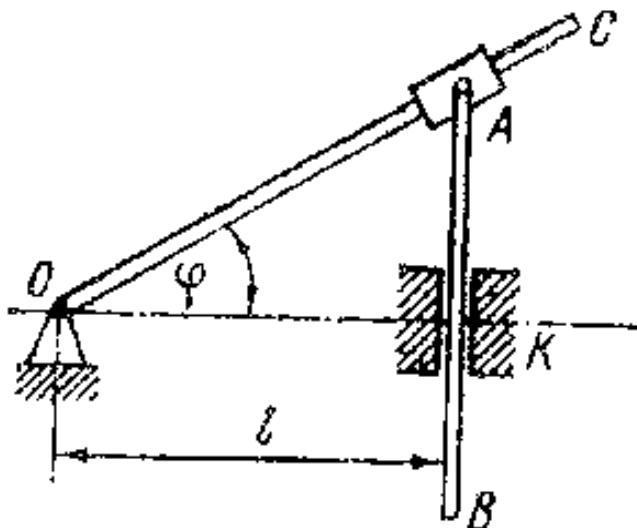
## Задача 16

( Задача 22.17. Задачи по теоретической механике. И.В. Мещерский. СПб.: Издательство «Лань», 2001. – 448 с. )

В кулиском механизме при качании кривошипа ОС вокруг оси О, перпендикулярной плоскости рисунка, ползун А, перемещаясь вдоль кривошипа ОС, приводит в движение стержень АВ, движущийся в вертикальных направляющих К. Расстояние ОК=l. Определить скорость движения ползуна А относительно кривошипа ОС в функции от угловой скорости  $\omega$  и угла поворота  $\varphi$  кривошипа.

Для решения задачи использовались следующие значения параметров:

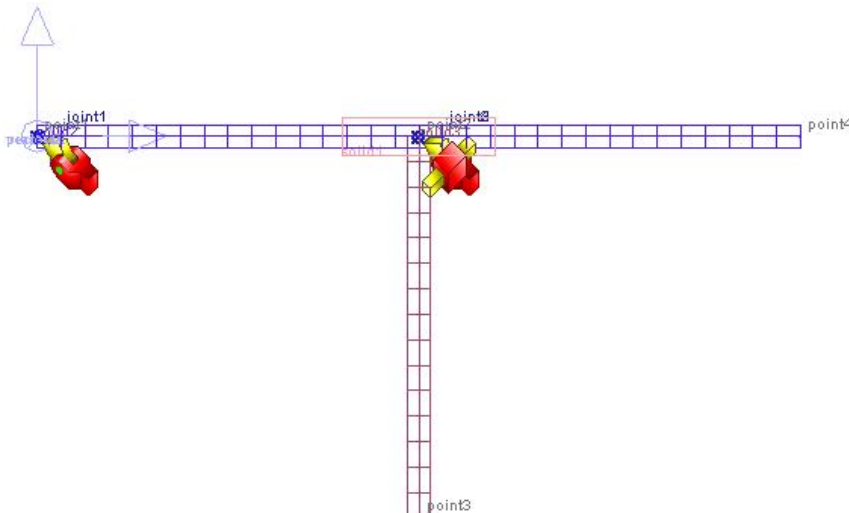
$$l=1 \text{ м}$$



**Точный теоретический ответ**

$$v = \frac{l \cdot \omega \cdot \operatorname{tg} \varphi}{\cos \varphi}$$

**Решение задачи в EULER**



Система состоит из четырёх звеньев.

- Инерциальное звено (base). В проекте звено не отображается.
- Ползун (body1). В проекте звено отображается телом solid1 и имеет массу 1[kg].
- Стержень ОС (body3). Звено отображается цилиндром solid2 и имеет массу 1[kg].
- Стержень АВ (body4). Звено отображается цилиндром solid3 и имеет массу 1[kg].

Связь инерциального звена со стержнем ОС моделируется с помощью шарнира типа «пара вращения». Связь инерциального звена со стержнем АВ моделируется с помощью шарнира типа «поступательная пара». Связь стержня ОС с ползуном моделируется с помощью шарнира типа «поступательная пара». Связь ползуна со стержнем АВ моделируется с помощью шарнира типа «пара вращения». Для задания движения груза используется объект программное движение motion1.

Точность решения определялась после выполнения команды command1 «Расчёт динамики движения» с отрезком интегрирования 1[s] при шаге 0.01[s].

### **Результаты моделирования**

Относительное отличие полученного в EULER решения от теоретического

$$\text{delta\_v} = 0.000\ 000\ 000\ 000\ 001\ 2 \ [-].$$

### **Текст проекта в EULER**

```

scalar l=1[m];
point point1=point( 0 [ m ], 0 [ m ], 0 [ m ] );
point point2=point( l, 0 [ m ], 0 [ m ] );
point point3=move( point2, projectY, -1[m] );
point point4=point( 2*l, 0 [ m ], 0 [ m ] );
solid solid1=box( nodePoint( point2 ), 0.4 [ m ], 0.1 [ m ], 0.2 [ m ], mass = 1 [ kg ] );
solid solid2=cylinder( point1, point4, 0.03 [ m ], mass = 1 [ kg ] );
solid solid3=cylinder( point2, point3, 0.03 [ m ], mass = 1 [ kg ] );
body base=body( color = RGB( 229, 229, 229 ) );
set ground = base;
body body2=body( color = RGB( 255, 153, 153 ) );
body body2 < ( solid1 );
color color4=index( 76 );
body body3=body( color = color4 );
body body3 < ( solid2 );
color color2=index( 27 );
body body4=body( color = color2 );
body body4 < ( solid3 );
joint joint1=rotational( base, body3, point1, projectZ );
joint joint2=rotational( body2, body4, point2, projectZ );
vector vector1=vectorPP( point1, point2 );
joint joint3=translational( body2, body3, point2, vector1 );
joint joint4=translational( body4, base, point2, projectY );
function fi(t[s])=45[deg]*sin(10[1/s]*#t);
motion motion1=ideal2( fi, joint1.gamma, time );
sensor v=transVelocity( body3, vector1, body2, point2 );
sensor w=derivative( joint1.gamma );
sensor v_theoretical=l*w*tg(joint1.gamma)/cos(joint1.gamma)/1[rad];
sensor delta_v=abs((v-v_theoretical)/v_theoretical);
command command1=constRK4( 1.00000e+000 [ s ], 1.00000e-002 [ s ] );

```

Λ//

Λ Список главных команд;  
set dynamics = command1;

Λ//

Λ Единицы измерения;  
set units = SI;