**БАЛТИЙСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ «ВОЕНМЕХ» им. Д.Ф. Устинова**

**Факультет информационных и управляющих систем**

**Кафедра И4 Радиоэлектронных систем управления**

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 3**

по учебной дисциплине

"Основы компьютерного проектирования и моделирования"

Вариант 12

Студент: Мечеткин Д.Д.

Группа: И4М31

**ПРЕПОДАВАТЕЛЬ**

\_\_\_\_\_\_\_\_\_/Петров Ю.В./

"\_\_\_\_"\_\_\_\_\_\_\_\_\_2019 г.

Санкт-Петербург  
2019 г.

**Задание к лабораторной работе №2.**

С помощью метода структурных схем разработать модель РЛС,

находящейся на подвижном носителе, и сопровождающей неподвижную цель

Исходные данные:

Xрлс = -60 км

Yрлс = 75 км

Xц нач = 0 км

Yц нач = 0 км

Wx рлс = 200 м/c

Wy рлс = 0 м/c

Предполагается, что:

- антенна РЛС направлена вниз вдоль оси Y;

- диаграмма направленности антенны имеет вид F(Al) = | sin(Al)/Al |;

- максимальное значение амплитуды сигнала (при Al=0) равно 1 мв;

- зависимость амплитуды сигнала от расстояния можно не учитывать.

Требуется получить зависимости расстояния между РЛС и целью D(t) и

амплитуды сигнала от времени.

**Листинг программы :**

unit Lr3Data;

interface

const

N = 600; { Размер реализации }

type

TArray = array [1..N] of real;

var

X,

Y : TArray;

XTitle,

YTitle: string;

TargX: Real;

TargY: Real;

R: Real;

Ai: Real;

alpha: Real;

F : Real;

RLSX: Real;

RLSY: Real;

implementation

var

I : integer; { Переменная цикла }

const

//--------------------- Определение констант ----------------------------------

Tmod = 1\*600; { Время моделирования }

dt = 1; { Интервал дискретизации по времени }

RLSX0 = -60;

RLSY0 = 75;

TXO = 0;

TYO = 0;

VXT = 0.2;

VYT = 0;

A = 1;

//-----------------------------------------------------------------------------

begin

YTitle:='Амплитуда';

{ XTitle:='Частота';

XTitle:='Фаза';

XTitle:='Время задержки';

XTitle:='Пеленг';

YTitle:='Расстояние';}

{ YTitle:='Амплитуда';

YTitle:='Частота';

YTitle:='Фаза';

YTitle:='Время задержки';

YTitle:='Пеленг'; }

XTitle:='Расстояние';

for I:=1 to N do

begin

RLSX:= RLSX0+VXT\*i\*dt;

RLSY:= RLSY0+VYT\*i\*dt;

R:= sqrt(sqr(TargX-RLSX)+(sqr(TargY-RLSY)));

alpha:=arctan(RLSX-TargX)/(RLSY-TargY);

if alpha=0 then

F:=1 else

F:=Abs(sin(alpha)/alpha);

Ai:=F/1\*A;

X[i]:=R; {Демонстрационное заполнение массивов}

Y[i]:=Ai;

end; end.

**Скриншот работы программы.**

