Требуется решить для

Код А (Хар-ка входного сигнала)

|  |  |
| --- | --- |
|  | tнач = 0 с; tкон = П с;  |

Код Б (Хар-ка выходного сигнала)

|  |  |
| --- | --- |
|  | Uвх1 = 2 B |

Код В (Найти) W

Минимальное значение выходного сигнала

1**. Постановка задачи.**

**Дано:**

 Нелинейная электрическая цепь.

 На входе цепи действует входной сигнал, заданный либо в виде графика, либо в виде аналитического выражения (Uвх(t)).

 Задана передаточная характеристика цепи (Uвых=f(Uвх)), которая функционально связывает входной сигнал с выходным. Передаточная характеристика также может быть задана либо в виде графика, либо аналитически.

 Задан определенный рабочий набор исходных данных, относящихся к входному сигналу и к передаточной характеристике.

**Найти:**

 1. Напряжение на выходе цепи (Uвых(t));

 2. Параметр W для выходного напряжения.(Длительность импульса, максимальное значение выходного сигнала, длительность переднего или заднего фронта выходного сигнала и т.п.)

 3. Записать отсчеты массивов времени, входного и выходного сигналов в файлы на диске.

 4. Вывести на экран монитора титульный лист, предварительно записанный ув текстовый файл, и программно считанный из этого файла.

 5. Cоздать и вывести на экран монитора меню, позволяющее работать с отдельными функциями программы.

 6. Построить графики входного и выходного сигналов цепи , используя файлы , в которые программно записаны массивы значений входного и выходного сигналов. Графики построить в системе Maxima. Этот пункт можно выполнить в сессию в лаборатории.

**2. Анализ задачи**

 Сигнал и передаточная характеристика заданы в аналоговом виде(непрерывный сигнал) или графически. Компьютер – цифровой. Он не способен обрабатывать напрямую ни аналоговый сигнал, ни тем более графику.

 Поэтому необходимо перейти от графики к аналитическому выражению:

 2.1 Переход от графики к аналитическому выражению.

 (смотри стр.5-6 методических указаний)

Принимаем решение о том, что наша задача будет состоять из отдельных функционально самостоятельных частей:

1. вывод на экран заставки.

2. Формирование массивов отсчетов времени и входного сигнала.

3. Формирование массива отсчетов выходного сигнала.

4. Меню.

5. Формирование таблицы отсчетов времени, входного и выходного сигналов

6.Вычисление заданной характеристики выходного сигнала.

7. Организация текстовых файлов и запись в них массивов времени, входного и выходного сигналов

**Пример программы.**

Дано:

Входной сигнал:

Uвх(t)=U1\*cos(2\*3.14\*t/T)+U2\*cos(3\*3.14\*t/T)

T=2c U1=15в г2=8в; 0<=t<=T

Передаточная характеристика цепи:

U44

Uвых(t)

Uвх(t)

U11

U22

U33

U11=32в U22=60в U33=12в U44=20в

Перейдем от графики к аналитическому выражению:

 a1\*U3+b1=U1

 a1\*U4+b1=U2

b1=U1-a1\*U3

a1\*U4+U1-a1\*U3=U2

a1=(U2-U1)/(U4-U3)

Uвых=((U2-U1)/(U4-U3))\*Uвх+U1-a1\*U3

a=(U2-U1)/(U4-U3)

b=U1-a1\*U3

a=3.5

b=-10

Uвых(t)=32 if Uвх(t)<=12

 3.5\*Uвх(t)-10 if Uвх(t)<=20

 60 if Uвх(t)>20

**Текст программы**

**Файл nastmain.cpp**

#include <stdlib.h>

#include<stdio.h>

#include<math.h>

#include"nastfunc.h"

 int main(){

 char ch;

 float tarr[1000],uin[1000],uout[1000],W;

 float U1=15,U2 =8,tn=0,T=2,tk=T;

 // float UIN1=10,UIN2=30;

 int n;

 zastavka();

 printf("\ninput n=");

 scanf("%d",&n);

 uin1(n,uin,tarr,U1,U2,T,tn,tk);

 uout1(n,uin,uout);

 menu();

 do {

 printf("input your cod:");

 scanf("%c",&ch);

 switch(ch) {

 case '1' : table(n,tarr,uin,uout); break;

 case '2' : {

 writeinfo(n,"f1.txt",uin);

 printf("f1.txt saved\n\n");

 writeinfo(n,"f2.txt",uout);

 printf("f2.txt saved\n\n");

 writeinfo(n,"f3.txt",tarr);

 printf("f3.txt saved\n\n");

 }

 break;

 case '3' : {

 float e=0.01;

 float v=10000;

 int n=50;

 float p=1.0;

 float ud;

 do {

 uin1(n,uin,tarr,U1,U2,T,tn,tk);

 uout1(n,uin,uout);

 float del=(tk-tn)/(n-1);

 ud=formingW(n,uout,del);

 p=fabs(v-ud)/ud;

 n=2\*n;

 v=ud;

 }while(p>e);

 printf("\n\n");

 printf("tochnost uout dostignuta\n");

 printf("iskomaja dlitelnost ud=%lf\n",ud);

 printf("tochnost p=%f\n",p);

 printf("kolich tochek n=%d\n",n/2);

 printf("zadan tochnost e=%f\n\n",e);

 n=50;

 e=0.01;

 do {

 uin1(n,uin,tarr,U1,U2,T,tn,tk);

 uout1(n,uin,uout);

 float del=(tk-tn)/(n-1);

 ud=formingW(n,uin,del);

 p=fabs(v-ud)/ud;

 n=2\*n;

 v=ud;

 }while(p>e);

 printf("\n\n");

 printf("tochnost uin dostignuta\n");

 printf("iskomaja dlitelnost ud=%lf\n",ud);

 printf("tochnost p=%f\n",p);

 printf("kolich tochek n=%d\n",n/2);

 printf("zadan tochnost e=%f\n\n",e);

 }

 break;

 case '4' :printf("finish,put<enter>,please:");

 exit(1);

 }

 }

 while(ch!='4');

 return 0;

}

File : **nastfunc.cpp**

#include<stdio.h>

#include<math.h>

#include"nastfunc.h"

int zastavka() {

 FILE \*f2;

 char ch;

 if((f2=fopen("zast.txt","r"))==NULL)

 printf("can't open your file\n");

 ch=getc(f2);

 while(ch!=EOF)

 {

 ch=getc(f2);

 printf("%c",ch);

 }

 fclose(f2);

 return 0;

}

int menu() {

 printf(" MAIN MENU\n\n");

 printf(" 1 - Table\n");

 printf(" 2 - Save\n");

 printf(" 3 - Show W\n");

 printf(" 4 - Exit\n ");

 return 0;

}

int uin1(int n, float uin[],float tarr[],float U1,float U2,float T,float tn,float tk){

 int i;

 float del=(tk-tn)/(n-1);

for(i=0;i<n;i++)

 tarr[i]=tn+i\*del;

 for(i=0;i<n;i++)

 uin[i]=U1\*cos(2\*3.14\*tarr[i]/T)+U2\*cos(3\*3.14\*tarr[i]/T);

 return 0;

}

int uout1(int n,float uin[],float uout[]) {

 int i;

 for(i=0;i<n;i++)

 if(uin[i]>=20)

 uout[i]=60;

 else

 if(uin[i]>=12)

 uout[i]=3.5\*uin[i]-10;

 else

 uout[i]=32;

 return 0;

}

int table(int n,float tarr[],float uin[],float uout[]) {

 int i;

 printf("\n TABLE ");

 printf("\n| i | time | uin | uout |");

 printf("\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

 for(i=0;i<n;i++)

 printf("\n| %6d |%12.9f | %15.8f | %20.8f |",i+1,tarr[i],uin[i],uout[i]);

 return 0;

}

float formingW(int n,float uout[],float del) {

 int i,max;

 float w,kol=0;

 max=uout[0];

 for(i=1;i<n;i++)

 if(uout[i]>max)

 max=uout[i];

 for(i=0;i<n;i++)

 if(uout[i]>0.7\*max)

 kol++;

 w=del\*(kol-1);

 return w;

}

 int writeinfo(int n,char fname[],float mas[]) {

 int i;

 FILE \*f;

 f=fopen(fname,"w");

 if(f==NULL)

 printf("\n error");

 else

 for(i=0;i<n;i++)

 fprintf(f,"%13.5f\n",mas[i]);

 fclose(f);

 return 0;

}

  **File: nastfunc.h**

#ifndef \_NASTFUNC\_H

#define \_NASTFUNC\_H

#endif /\* \_NASTFUNC\_H \*/

 int zastavka();

 int menu();

 int uin1(int n, float uin[],float tarr[],float U1,float U2,float T,float tn,float tk);

 int uout1(int n,float uin[],float uout[]);

 int table(int n,float tarr[],float uin[],float uout[]);

 float formingW(int n,float uout[],float del);

 int writeinfo(int n,char fname[],float mas[]);