

/*

Testprogramm für ein Unterprogramm

für ein IIR-Filter iir

Es wird mit Gleitkommazahlen gerechnet.
Parallelarbeit wird nicht berücksichtigt

Dem Unterprogramm werden übergeben

B0 - Adresse für den x-Vektor
M0 - Modifier
L0 - Länge des x-Vektors

B1 - Adresse für den y-Vektor
M1 - Modifier
L1 - Länge des y-Vektors

B8 - Adresse für den Koeffizientenvektor bk
M8 - Modifier
L8 - Länge des Koeffizientenvektors bk

B9 - Adresse für den Koeffizientenvektor ak
M9 - Modifier
L9 - Länge des Koeffizientenvektors ak

F0 der neue x-Wert

*/

#define ordnung 4 /* Ordnung des Filters */
.SECTION /PM seg_pmda;

/* Die Koeffizienten definieren einen Hochpass
Sie wurden mit Hilfe von Matlab berechnet */
.VAR akoeforg[]="akorg.dat";
.VAR bkoeforg[]="bkorg.dat";
/* Die Koeffizienten werden zur Realisierung
des Filters umgespeichert */
.VAR akoef[ordnung];
.VAR bkoef[ordnung];

.SECTION /DM seg_dmda;

/* Die x und y Vektoren für die
Differenzgleichung */
.VAR xvek[ordnung];
.VAR yvek[ordnung];

/* Die Vektoren für den Ersatz der Ein- Ausgabe-
Signale des Filters
Als "Eingangssignal wird die Sprungfunktion
gewählt */
.VAR eingangssignal[]="xein.dat";
.VAR ausgangssignal[210];

.SECTION /PM seg_rth;

NOP;
NOP;
NOP;
NOP;
NOP;
JUMP start;

.SECTION /PM seg_pmco;

start: B0=xvek; /* x-Vektor adressieren */
M0=1;
L0=@bkoef;
B1=yvek; /* y-Vektor adressieren */
M1=1;
L1=@akoef;

```

        B8=bkoef;    /* ak Koeffizientenvektor adressieren */
        M8=1;
        L8=@bkoef;
        B9=akoef;    /* bk Koeffizientenvektor adressieren */
        M9=1;
        L9=@akoef;
        CALL umspeich;
        B8=bkoef;    /* ak Koeffizientenvektor adressieren */
        M8=1;
        L8=@bkoef;
        B9=akoef;    /* bk Koeffizientenvektor adressieren */
        M9=1;
        L9=@akoef;
        B3=eingangssignal; /* Eingangsvektor adressieren */
        M3=1;
        L3=@eingangssignal;
        B4=ausgangssignal; /* Ausgangsvektor adressieren */
        M4=1;
        L4=@ausgangssignal;
/* y-Vektor und x-Vektor löschen */
        lcntr = L0, do loeschen UNTIL LCE;
        DM(I1,M1)=0.0;
loeschen:  DM(I0,M0)=0.0;
/* Ausgangswert durch Filter berechnen */
        lcntr = L3, do filtern UNTIL LCE;
        F0=DM(I3,M3); /* neuen Eingabewert lesen */
        CALL iir;     /* IIR Unterprogramm */
        nop;
        nop;
        /* neuen y - Wert in Ausgabe speichern */
filtern:  DM(I4,M4)=F0; /* neuen Ausgabewert speichern */
        nop;
        nop;

ende:     idle;
/* Unterprogramm IIR-Filter
   belegte Register:
        F12 Ergebnis der Teilprodukte bk*xk
        F0  Wert aus Vektor xk
           Rückgabe neuer y - Wert
        F4  Wert aus Vektor bk
        F8  Wert für Summe der Teilprodukte bk*xk
        F13 Ergebnis der Teilprodukte ak*yk
        F1  Wert aus Vektor yk
        F5  Wert aus Vektor ak
        F9  Wert für Summe der Teilprodukte ak*yk
*/

iir:     DM(I0,M1)=F0; /* neuen Wert in x-Vektor schreiben */
        R12=R12 xor R12;
        R8=R8 xor R8;
        R13=R13 xor R13;
        R9=R9 xor R9;
/* Filterschleife */
        lcntr = L8, do rechner until LCE;
        F0=DM(I0,M0);
        F4=PM(I8,M8);
        F12=F0*F4;
        F8=F8+F12;
        F1=DM(I1,M1);
        F5=PM(I9,M9);
        F13=F1*F5;
        F9=F9+F13;
rechner:  F0=F8+F9;
        DM(I1,M1)=F0; /* yk-Vektor aktualisieren */
        rts;
/* Unterprogramm zum Umspeichern der Koeffizienten

umspeich:  B10=bkoeforg; /* ak Koeffizientenvektor adressieren */
        M10=-1;
        L10=@bkoeforg;
        B11=akoeforg; /* bk Koeffizientenvektor adressieren */
        M11=-1;

```

```
L11=@akoefforg;
F0=0.0;
PM(I11,0)=F0;
F0=PM(I10,M10);
F4=PM(I11,M11);
F0=PM(I9,M9);
F1=-1.0;
lcnt = L8, do umordnung until LCE;
F0=PM(I10,M10);
PM(I8,M8)=F0;
F4=PM(I11,M11);
F4=F4*F1;
umordnung: PM(I9,M9)=F4;
RTS;
```

```
/* Es werden nachfolgend aufgeführte Datenfiles
zur Initialisierung benötigt
```

```
Inhalt des Files
akorg.dat
```

```
1.0,-2.3741,1.9294,-0.53208
```

```
Inhalt des Files
bkorg.dat
```

```
0.72944,-2.1883,2.1883,-0.72944
```

```
Der File
xein.dat
```

```
enthält die Sprungfunktion mit 210 Werten
```

```
*/
```