1、

.mmregs

STACK .usect ".stack",30H

.bss x,20

.bss y,20

.data

table: .word 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20

.def start

.text

start: STM #000H,SWWSR

STM #STACK+30H,SP

STM #x,AR1

RPT #19

MVPD table,\*AR1+

STM #x,AR2

STM #y,AR3

RPT #19

MVDD \*AR2+,\*AR3+

end: B end

.end

2、

.mmregs

STACK .usect ".stack",30H

.bss a,20

.bss x,20

.bss y,2

.def start

.data

table: .word 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20

.text

start: STM #000H,SWWSR

STM #STACK+30H,SP

STM #a,AR1

RPT #19

MVPD table,\*AR1+

STM #x,AR1

RPT #19

MVPD table,\*AR1+

CALL sum

end: B end

sum: STM #a,AR2

STM #x,AR3

RPT #19

MAC \*AR2+,\*AR3+,A

STL A,@y

STH A,@y+1

RET

.end

3、

.mmregs

.include "inittable.inc"

STACK .usect ".stack",30H

.bss x,3

.bss d,3

.def start

.text

start: STM #000H,SWWSR

STM #STACK+30H,SP

STM #x,AR1

RPT #5

MVPD table,\*AR1+

STM #x,AR5

STM #d,AR2

; LD #00,ASM

LD \*AR5+,16,A

ADD \*AR5+,16,A

ST A,\*AR5||LD \*AR2+,B

ADD \*AR2+,16,B

STH B,\*AR2

end: B end

.end

4、

.mmregs

.include "inittable.inc"

STACK .usect ".stack",30H

.bss x,3

.bss d,3

.def sin\_start

; Functional Description

; This function evaluates the sine of an angle using the Taylor series

; expansion.

; sin(theta) = x(1-x^2/2\*3(1-x^2/4\*5(1-x^2/6\*7(1-x^2/8\*9))))

;

.mmregs

.def d\_x,d\_squr\_x,d\_coff,d\_sinx,C\_1

d\_coff .sect "coeff"

.word 01c7h

.word 030bh

.word 0666h

.word 1556h

d\_x .usect "sin\_vars",1

d\_squr\_x .usect "sin\_vars",1

d\_temp .usect "sin\_vars",1

d\_sinx .usect "sin\_vars",1

C\_1 .usect "sin\_vars",1

.text

sin\_start:

STM #000H,SWWSR

STM #STACK+30H,SP

SSBX SXM

SSBX FRCT

STM #d\_coff,AR3 ; c1=1/72,c2=1/42,c3=1/20,

; c4=1/6

STM #d\_x,AR2 ; input value

STM #C\_1,AR4 ; A1, A2, A3, A4

sin\_angle:

LD #d\_x,DP

ST #3243h,d\_x ; pi/4

ST #7fffh,C\_1 ; C\_1=1

SQUR \*AR2+,A ; let x^2 = P

ST A,\*AR2||LD \*AR4,B ; AR2 > x^2 ,B=1

MASR \*AR2+,\*AR3+,B,A ; (1-x^2)/72 , T = x^2

MPYA A ; x^2(1-x^2)/72

;

STH A,\*AR2 ; AR2=d\_temp= x^2(1-x^2)/72

MASR \*AR2-,\*AR3+,B,A ; A = 1-x^2/42(1-x^2/72)

; T =x^2(1-x^2/72)

MPYA \*AR2+ ; B = A(32-16)\*x^2

ST B,\*AR2||LD \*AR4,B ; d\_temp= B = x^2(1-x^2/42(1-x^2/72))

; B = C\_1

MASR \*AR2-,\*AR3+,B,A ; A = 1-x^2/20(1-x^2/42(1-x^2/72)

MPYA \*AR2+ ; B = A(32-16)\*x^2 = (1-x^2/20(1-x^2/42(1-x^2/72))\*x^2

ST B,\*AR2||LD \*AR4,B ; d\_temp= B = (1-x^2/20(1-x^2/42(1-x^2/72))\*x^2

MASR \*AR2-,\*AR3+,B,A ; AR2- > d\_squr\_x

MPYA d\_x

STH B, d\_sinx ; sin(theta)

end: B end

.end

5、

.mmregs

.include "htable.inc"

.include "xtable.inc"

.def start

htabcnt .set 20

xtabcnt .set 80

STACK .usect ".stack",30H

.bss ncnt,1

.bss x\_tab,80H,1

.bss h\_tab,80H,1

.bss r\_tab,80H,1

.bss db\_tab,80H,1

.text

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

; Function: fir

; Version: 1.00

; Description: delayed buffer finite impulse response filter

;

; Copyright Texas instruments Inc, 1998

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

; Register usage

; --------------

; .asg 0 + FRAME\_SZ, SAVE\_AR1

; .asg 0 + REG\_SAVE\_SZ + FRAME\_SZ, RETURN\_ADDR

; .asg 0 + PARAM\_OFFSET, h

; .asg 1 + PARAM\_OFFSET, r

; .asg 2 + PARAM\_OFFSET, db

; .asg 3 + PARAM\_OFFSET, nh

; .asg 4 + PARAM\_OFFSET, nx

; .asg 0, nc

.asg AR2, r\_ptr ;resulte serial

.asg AR3, h\_ptr ;h serial filter

.asg AR4, x\_ptr ;x serial input

.asg AR5, db\_ptr ;middle serial

.asg BRC, rptb\_cnt ;x counter

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;

; Save contents of AR1

; And establish local frame

; Set sign extension mode

; Set FRCT bit:

;----------------------------------------------------------------

start:

SSBX SXM ; 1 cycle

SSBX FRCT ; 1 cycle

;

; Copy arguments to their local locations as necessary

;----------------------------------------------------------------

STM #x\_tab,x\_ptr

RPT #(xtabcnt-1)

MVPD xtab,\*x\_ptr+

STM #h\_tab,h\_ptr

RPT #(htabcnt-1)

MVPD htab,\*h\_ptr+

STM #x\_tab,x\_ptr

STM #h\_tab,h\_ptr

STM #r\_tab,r\_ptr

STM #db\_tab,db\_ptr

; Set outer loop count by subtracting 1 from nsamps and

; storing into block repeat count register

;----------------------------------------------------------------

;

LD #xtabcnt, A ; 1 cycle h counter

STLM A, rptb\_cnt

; Store length of coefficient vector/ delay buffer in BK

; register

;----------------------------------------------------------------

LD #htabcnt, A ; 1 cycle h counter

STLM A, BK ; 1 cycle

SUB #3, A ; 2 cycles

STL A, \*(ncnt) ; 1 cycle

;

; Begin outer loop on # samples

;----------------------------------------------------------------

RPTBD END\_LOOP-1 ; 2 cycles

;

; Store 0 to AR0, to use as circular addressing offset

;----------------------------------------------------------------

STM #1, AR0 ; delay slot; 2 cycles

;

; Zero the accumulator before calculating next sum.

; Move next input sample into delay buffer

;----------------------------------------------------------------

MVDD \*x\_ptr+, \*db\_ptr ; 1 cycles

;

; Sum h \* x for next y value

;----------------------------------------------------------------

MPY \*h\_ptr+0%, \*db\_ptr+0%, A ; 1 cycle

RPT \*(ncnt) ; 2 cycle

MAC \*h\_ptr+0% , \*db\_ptr+0%, A ; 1 cycle \* ncoeffs-2

MACR \*h\_ptr+0% , \*db\_ptr, A ; 1 cycle

;

; Store result

;----------------------------------------------------------------

STH A, \*r\_ptr+ ; 1 cycle

END\_LOOP:

\_end:

;

; Reset FRCT bit to restore normal C operating environment

; Return overflow condition, OVA, in accumulator A

; Restore stack to previous value, FRAME, etc..

;----------------------------------------------------------------

RETURN:

LDM db\_ptr, B ; 1 cycle

LD #0, A ; 1 cycle

XC 1, AOV ; 1 cycle

LD #1, A ; 1 cycle