Application Report
SPRA619 - December 1999

General Guide to Implement Logarithmic and Exponential Operations on a Fixed-Point DSP

Jason Jiang
TI (Shanghai) CAC

ABSTRACT

Modern audio and video compression algorithms usually take the advantage of logarithmic characteristics of human ears and eyes. This approach greatly reduces the redundancy in signals being processed. However, it poses a requirement on fixed-point DSPs to handle these logarithmic and exponential operations. This application report provides a general guide to implement these operations on fixed-point DSPs and sample codes based on Texas Instruments (TI™) TMS320C54x DSP.

Contents

1 Introduction

Modern audio and video compression algorithms usually take the advantage of logarithmic characteristics of human ears and eyes. This approach greatly reduces the redundancy in the original signals. However, when implementing this processing on fixed-point DSPs, there is a requirement to maintain enough SNR without noticeable degradation at the output. For example, the popular MP3 algorithms achieves 12:1 compression ratio with the help of logarithmic processing. The decoding requires the expanding operation:

$$Xr = is^{4/3} * 2^{*\,exp} \qquad (1)$$

where –85 < exp < 12, 0 <= is < 8196. Eq.1 is a typical equation. By studying the equation, it is impossible to calculate directly on a fixed-point DSP. However, after taking a logarithm on both sides of the equation, it is simplified to multiplication and addition that a fixed-point DSP can handle.

$$Y = \ln Xr = 4/3 \ln is + exp * \ln 2 \qquad (2)$$

Now the problem is how to calculate (ln is) and compute the final result Xr from Y. These two problems will be tackled in detail in the following sections and example codes will be provided.

2 Implementation of Logarithmic Operation

Now, let's first examine the logarithmic part (ln is). According to Taylor's Equation

$$\ln(1-x) = -x - x^2/2 - x^3/3 \ldots - x^n/n \qquad (3)$$

where $-1 <= x < 1$, it is easy to expand the logarithmic operation to a sequence of multiply and add operations. Since (is) is an integer, it should be scaled to an appropriate range before using the Taylor's equation.

$$is = C\,(1 - x) \qquad (4)$$

Where $C = 2^N$. Then it leads to

$$\ln is = \ln C + \ln (1-x) = N \ln 2 + \ln (1-x) \qquad (5)$$

Experimental results show that for N=11, a good audio decompression can be achieved. Following is the example code based on C54x DSP. The detail explanation is embedded.

```
        .mmregs
        .def log
;
; This routine is used to calculate the logarithm of an integer
; which is between 0 and 32768.  It is programmer's responsibility
; to validate the input value and make sure DP points to the data page
; of N and X.
; Input:   AL = integer, AH = 0
; Output:  AL = fraction part in Q11 format and MSB 5bits are integer

;               part
;          AH = 0
; Register used:  B, T, AR0, AR3
; Memory:  30 words
;
        .data
;
;logtbl is in Q11 format.  Logtbl is generated by equation 2048*n*ln2.
;This table is starting from n=15 down to n=0.
;
logtbl  .int 21293, 19874, 18454, 17035, 15615, 14196, 12776
        .int 11357, 9937, 8517, 7098, 5678, 4259, 2839, 1420, 0
;
;a9 is in Q15 format.  It is generated by equation –32768/n.  In order
;to take the advantage of instruction POLY, it is organized in reverse
;order.  Which means n start from 10 down to 1.  Extra 2 word memory
;space is reserved to reduce the possibilities of data over–written.
;
a9      .int –3277, –3641, –4096, –4681, –5461, –6554, –8192
        .int –10923, –16384, –32768, 0, 0
        .bss N, 1                       ;N stores scaling number
        .bss X, 1                       ;X is temporary storage. Must
                                        ;be at the same data page of N.
        .text
```

```
log:
        ADD     #0, A, B                ;B=A=is
        EXP     B                       ;T=leading 0s
        LD      #0x4000, 16, A          ;AH=16384, the largest
                                        ;supported scale
        ST      T, N                    ;store scaling number
        ANDM    #0Fh, N                 ;compensate extra 16 leading
                                        ;bits
        MVDM    N, AR0                  ;AR0 index to segment table
        NORM    B                       ;normalize to Q15 format
        AND     #0x3FFF, 16, B          ;BH=BH–0x4000
        BC      taylor, BNEQ            ;if (B==0) which means is takes
                                        ;2^N form
                                        ;then index to the pre–stored
                                        ;table
        STM     #logtbl+1, AR3          ;and return the value
        MAR     *AR3+0
        LD      *AR3, A
        RET
taylor:
        SUB     B, 0, A                 ;A=A–B, A is the x in Taylor’s
                                        ;equation
        STH     A, X                    ;X is the fractional part in
                                        ;Q15 format
        STM     a9, AR3                 ;AR3 points to coefficient in
                                        ;Taylor’s equ.
        LD      X, T                    ;set up running environment
                                        ;using
        LD      *AR3+, 16, A            ;powerful poly instruction on
                                        ;c54x DSP
        LD      *AR3+, 16, B            ;
        RPT     #10                     ;loop 11 times enough for audio

                                        ;app.
        POLY    *AR3+                   ;AH=fractional part in Q15
                                        ;format
        SFTA    A, –16                  ;AL=AH
        SFTA    A, –4                   ;AL is now in Q11 format
        STM     #logtbl, AR3            ;sum up scaling part
        MAR     *AR3+0
        ADD     *AR3, A
        RET
        .end
```

3 Implementation of Exponential Operation

After getting Y from the previous calculation, it’s time to calculate Xr from Y by performing a exponential operation. The Taylor Equation states:

$$e^x = 1 + x + x^2/2! + x^3/3! + \ldots + x^n/n! \quad (6)$$

where $-\infty < x < +\infty$. The Y given by previous calculation is in Q11 format. Therefore, the MSB5 bits are integer part and LSB11 bits are fractional part. It is convenient to rewrite the equation as following,

$$Xr = e^{Y} = e^{Ik} * e^{x} \quad (7)$$

Since Ik has only 12 possible occurrences, it is convenient to pre-store the result in a table. Now look into the fractional part. If x is smaller than 0.5, the Taylor equation converges quickly. However, if it is larger than 0.5, it converges very slowly. Therefore, if x is larger than 0.5, the Eq.8 should be rewritten as

$$Xr = e^{Y} = e^{Ik} * e^{x} = e^{Ik+1} * e^{x-1} \quad (8)$$

Following gives the example code based on C54x DSP. The detail explanation is embedded.

```
        .mmregs
        .def exp
;
; This routine is used to calculate the exponential of an input
; which is given by log in Q11 format.  The integer part should be
; less than 10 according to MP3 algorithm.  Programmer should be
; responsible for make sure DP is pointed to the data page of N and X.
; Input:   AL = fraction part in Q11 and MSB 5bits are integer part, AH ;
       = 0
; Output:  AL = integer, AH = 0
; Register used:  B, T, AR0, AR3
; Memory:  30 words
;
        .data
;
;exptbl is generated by equation en. n starts from 0 to 10.
;
exptbl  .int 1, 3, 7, 20, 55, 148, 403, 1097, 2981, 8103, 22026, 0
;
;a9 is in Q15 format. It is generated by equation 32768/n!. n starts ;from 8
down to 1 in order to take advantage of instruction POLY.
;
a9      .int 1, 7, 46, 273, 1365, 5461, 16384, 32767, 0, 0
        .bss N, 1                         ;N stores scaling number
        .bss X, 1                         ;X is temporary storage. Must
                                          ;be at the same data page of N.
        .text
exp:
        ADD     #0, A, B                  ;B=A=Y
        AND     #400h, B                  ;Check if it is larger than 0.5
        BCD     adj, BNEQ                 ;Yes, then need adjustment
        ADD     #400h, A, B
        STL     B, –11, N                 ;store scaling index
        AND     #3FFh, B                  ;truncate fractional part
        STL     B, 4, X                   ;store fractional part in Q15
                                          ;format
        B       taylor
adj:
        STL     B, –11, N                 ;store scaling index
        AND     #7FFh, B                  ;truncate fractional part
        SUB     #400h, B
        STL     B, 4, X                   ;store negative fraction in Q15
                                          ;format
```

```
taylor:
        STM     a9, AR3                 ;AR3 points to coefficient in
                                        ;Taylor's equ.
        LD      X, T                    ;set up running environment
                                        ;using
        LD      *AR3+, 16, A            ;powerful poly instruction on
                                        ;c54x DSP
        LD      *AR3+, 16, B            ;
        RPT     #7                      ;loop 8 times enough for audio
                                        ;app.
        POLY    *AR3+                   ;AH=fractional part in Q15
                                        ;format
        ADD     #7FFFh, 16, A           ;taylor equ. has 1 constant
        MVDM    N, AR0                  ;index into exptbl
        STM     exptbl, AR3
        MAR     *AR3+0
        MPYA    *AR3                    ;multiply the scaling part
        SFTA    B, –16, A               ;AL=BH
        RET
        .end
```

4 Summary

Although a fixed-point DSP does not have the advantage of calculating logarithmic and exponential operations, it does offer ways to approximate the result with limited precision. The tradeoffs between precision and DSP chip cost are made according to different applications. In modern audio/video applications such as MP3, a fixed-point DSP presents sufficient precision of these logarithmic and exponential operations.

5 References

1. TMS320C54x DSP CUP and Peripherals (SPRU131)
2. International Standard ISO/IEC 11172–3

IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgement, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its semiconductor products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

CERTAIN APPLICATIONS USING SEMICONDUCTOR PRODUCTS MAY INVOLVE POTENTIAL RISKS OF DEATH, PERSONAL INJURY, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE ("CRITICAL APPLICATIONS"). TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF TI PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE FULLY AT THE CUSTOMER'S RISK.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty or endorsement thereof.