

# **Digitale Signalprozessoren verbessern die Kopfhörer- wiedergabe**

von

Andreas Straub  
Motorola München

## **Zusammenfassung**

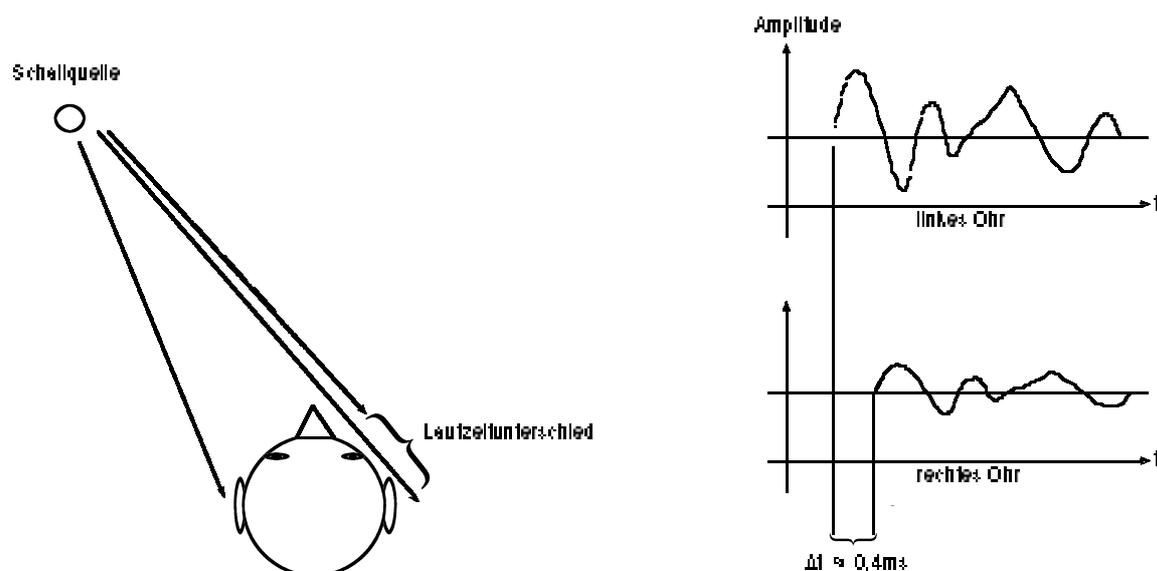
Ein allgemein bekanntes Problem bei der Kopfhörerwiedergabe von Tonmaterial, welches für Lautsprecherwiedergabe optimiert wurde, ist der Effekt der "Im-Kopf-Lokalisation". Dies bedeutet, daß der scheinbare Ort der gehörten Schallquellen sich nicht außerhalb des Kopfes, sondern auf einer imaginären Linie zwischen den beiden Ohren liegt. Der folgende Artikel beschreibt die Ursachen dieses äußerst unangenehmen Effekts und beschreibt eine Methode zur Simulation der Wiedergabe über Lautsprecher mittels digitaler Signalverarbeitung, welche eine wesentlich natürlichere Reproduktion über Kopfhörer erlaubt.

## Die natürliche Richtungserkennung des Ohres

In *Bild 1* ist der Kopf des Hörenden, sowie ein punktuell Schallereignis dargestellt. Um nun den Ort der Schallquelle festzustellen hat das Ohr mehrere Parameter zur Verfügung:

- die Lautstärke
- Laufzeitunterschiede und
- Klang-Veränderungen durch die Richtwirkung der Ohrmuscheln und des Gehörganges.

Im Nahfeld des Hörenden reicht im einfachsten Fall der Lautstärkeunterschied zwischen beiden Ohren aus. Entfernt sich jedoch die Schallquelle wird dieser Unterschied immer geringer, und das Ohr muß zu feineren Methoden greifen, um eine genaue Ortung zu ermöglichen. Eine ganz wesentliche Rolle spielt hierbei der Laufzeitunterschied des Schalls beim Eintreffen an den beiden Ohren. Das Ohr kann mit dessen Hilfe durch eine Art Korrelation der beiden Signale eine sehr hohe Genauigkeit bei der Erkennung des Winkels der Schallquelle erreichen. Problematisch bleibt hierbei jedoch immer noch die Unterscheidung von Schallereignissen, welche sich vor, hinter, ober- oder unterhalb des Hörenden abspielen, da in diesen Fällen gleiche Amplituden- und Laufzeitverhältnisse entstehen können. Hier spielt die richtungsabhängige Filterfunktion des Innenohres und der Hörmuscheln eine entscheidende Rolle.

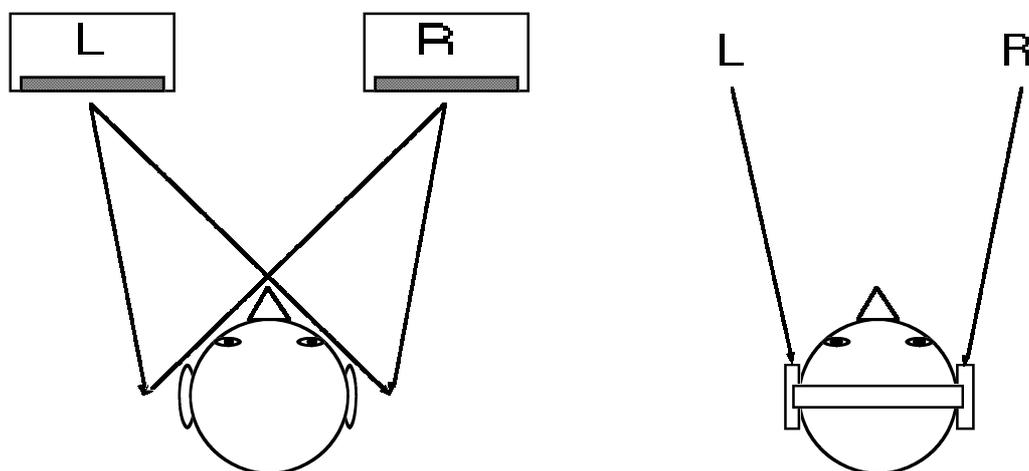


**Bild 1** Richtungserkennung des Ohres

Bei Verwendung aller drei Parameter lassen sich alle Schallereignisse mit Ausnahme der sich auf der senkrechten Ebene zwischen den beiden Ohren befindlichen Punkte sehr genau lokalisieren. Diese bewirken auf beiden Ohren genau identische Signale und sind damit nicht direkt ortbar. Hier hilft sich der Hörende unbewußt damit, daß er den Kopf ein klein wenig bewegt und somit wieder eine klar definierte Richtung erkennen kann. In Zukunft wird es möglich sein diese Abhängigkeit von der Kopfbewegung ebenfalls nachzubilden und so noch einen großen Schritt hin zu einer naturgetreueren Wiedergabe zu erreichen.

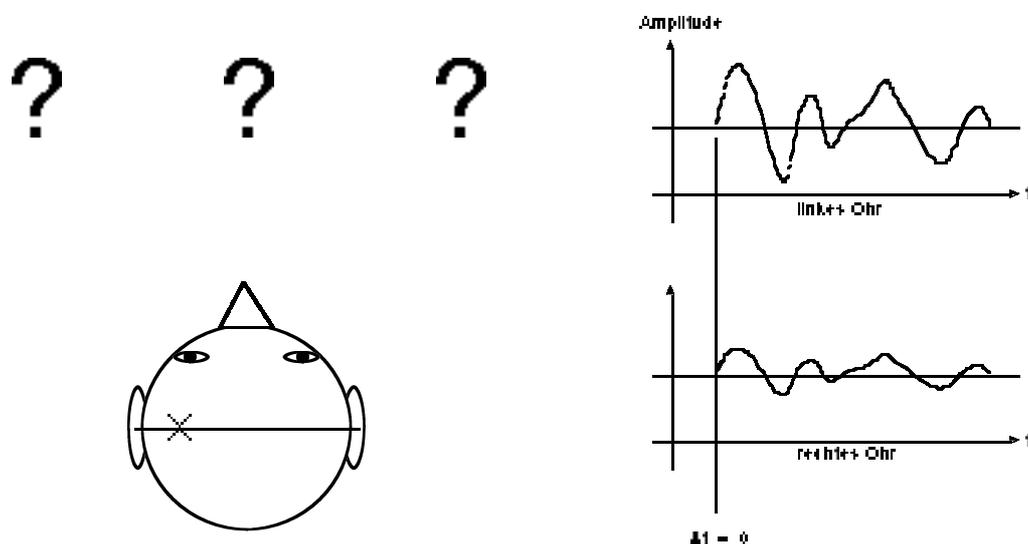
### **Lautsprecher und Kopfhörer - zwei sehr unterschiedliche Medien**

In *Bild 2* sind beide Wiedergabesysteme gegenüber gestellt. Der entscheidende Unterschied besteht nun darin, daß bei der Lautsprecherwiedergabe beide Ohren jeweils beide Lautsprecher hören. Mittels des Kopfhörers gelangt jedoch nur der rechte Signalanteil an das rechte Ohr und der linke an das linke Ohr. Der indirekt Schallanteil z.B. vom linken Lautsprecher auf das rechte Ohr entfällt ersatzlos. Dies bedeutet, daß der Hörende im Kopfhörer in den meisten Fällen Signale vorfindet, die auf dem jeweils anderen Ohr keinen in der Natur vorhandenen zweiten Signalanteil enthalten.



**Bild 2 Lautsprecher und Kopfhörerwiedergabe**

Das Problem entsteht bei allen Aufnahmetechniken, welche entweder mehr als zwei Mikrofone verwenden oder gar nicht akustisch, sondern rein elektrische Mischungen verwenden. Hierbei wird im Studio zur Richtungseinstellung praktisch ausnahmslos nur der relative Pegel einer Schallquelle in beiden Kanälen verändert, ohne die in der Natur zusätzlich auftretenden Laufzeitunterschiede und Klang-Veränderungen zu berücksichtigen. Bei rein akustischen Aufnahmen mit nur zwei Mikrofonen etwa im Ohrabstand, oder der Kunstkopftechnik tritt dieses Problem erst gar nicht auf, wodurch sich auch erklärt, warum bei der Kunstkopfstereofonie die "Im-Kopf-Lokalisation" nie ein Problem war. Diese entsteht immer dann, wenn beide Ohren Kombinationen von Signalen erhalten, welche in der Natur nicht auftreten können. Dann kann das Gehirn dem Schallereignis keine bekannte Richtung mehr zuordnen und legt den Ursprung des Schalls folgerichtig an den einzigen Ort, an dem sich in der Natur keine Schallquelle befindet: INNERHALB DES KOPFES. Ein Beispiel hierzu ist in Bild 3 dargestellt. Die Schallquelle wurde elektrisch durch Veränderung der Signalpegel auf dem linken und rechten Kanal etwa 45 Grad nach links abgemischt. Im Kopfhörer erhält das Ohr also links und rechts unterschiedliche Pegel des Tonsignals. Diese treffen jedoch ohne Laufzeitunterschied ein. Diese Situation läßt sich rein akustisch nicht erzeugen und bewirkt demzufolge die Lokalisation des Signals IM Kopf.



**Bild 3 Richtungseinstellung durch Lautstärkeveränderung**

## Lösungsmöglichkeiten

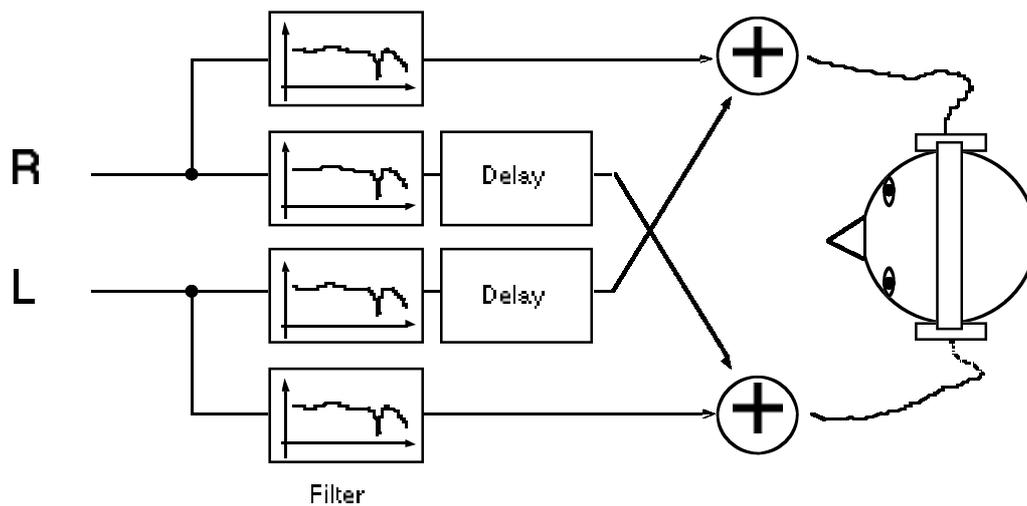
Die einfachste Lösung aus dem beschriebenen Dilemma ist die Kunstkopfstereofonie, welche dem Kopfhörer als Wiedergabemedium ein funktional entsprechendes Aufnahmesystem gegenüberstellt. Dies ist auch nach wie vor die beste Möglichkeit, um möglichst naturgetreu Schallereignisse zu übertragen. Leider hat diese Technik zwei wesentliche Nachteile:

- Kunstkopfaufnahmen sind nur bei rein akustischen Ereignissen möglich
- sie sind nicht (oder nur sehr bedingt) zur Lautsprecherwiedergabe geeignet.

Beide Gründe haben leider dafür gesorgt, daß diese wirklich hervorragende Aufzeichnungstechnik heute nur noch ein Schattendasein fristet. Vielleicht können in der Zukunft mehrkanalige digitale Audiosysteme und Tonträger dieser Technik wieder etwas mehr Leben einhauchen. Schließlich wird man in der Lage sein auf demselben Tonträger beide Aufzeichnungstypen zu kombinieren.

In der Praxis sieht es nun leider so aus, daß der überwiegende Teil aller Tonkonserven traditionell aufgenommen und abgemischt wird, und damit erst einmal für die Kopfhörerwiedergabe **ungeeignet** ist. Dennoch ist es möglich auch diese Aufnahmen so zu modifizieren, daß für das Ohr zumindest natürliche Signalverhältnisse entstehen und dadurch eine deutliche "Außer-Kopf-Lokalisation" entsteht.

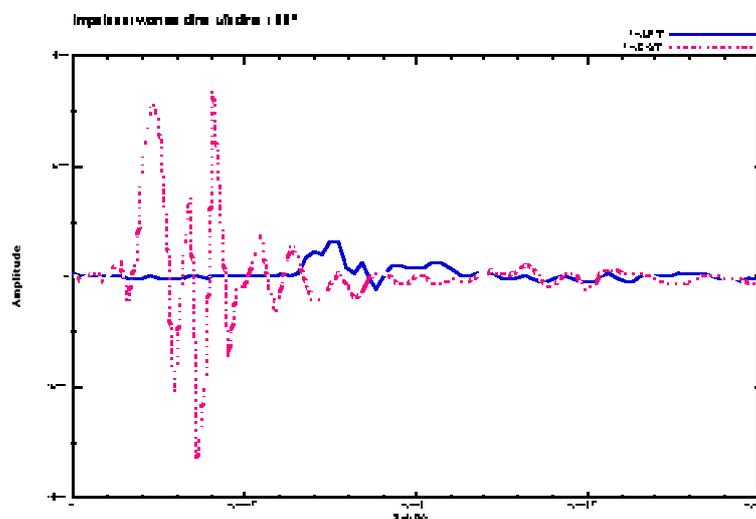
Hierzu müssen die Verhältnisse bei der Lautsprecherwiedergabe nachgebildet werden. Bei dieser werden durch das "akustische Übersprechen" der beiden Lautsprecher wieder für das Ohr natürliche Verhältnisse erzeugt (Laufzeitunterschiede und richtungsabhängige Frequenzgangänderungen). Auf elektrischem Wege kann dies mittels elektrischer Übersprecheinheiten ("Cross-Talk Units" siehe Bild 5) geschehen, welche die in der Natur entstehenden Laufzeiten und Frequenzgänge nachbilden. Ein derartiges System hat dann zwar alle Nachteile der Lautsprecherwiedergabe (mit Ausnahme der Raumreflektionen) wirkt aber für den Hörenden um soviel natürlicher, daß das Rückschalten auf normalen Stereobetrieb nach einer kurzen Eingewöhnungsphase erfahrungsgemäß als äußerst unangenehm empfunden wird.

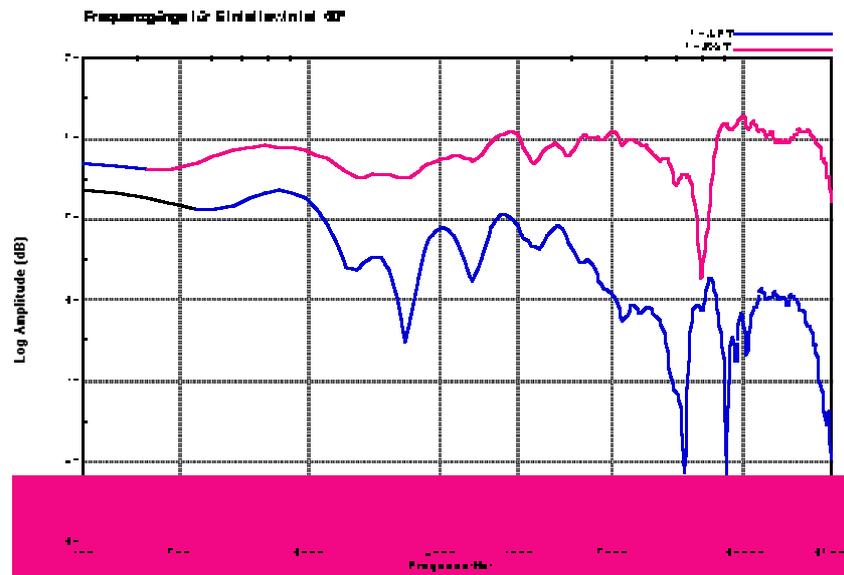


**Bild 4 Elektrisches Equivalent des akustischen Übersprechens**

### Realisierung

Grundlage für diese sogenannte "binaurale" Wiedergabe ist eine genaue Erfassung der Wirkungen des Hörkanals und der Ohmuscheln. Hierzu werden bei einer Testperson Sondenmikrofone in das Ohr nahe dem Trommelfell eingeführt und die Impulsantworten einer bewegten Schallquelle abhängig vom jeweiligen Azimut- und Elevationswinkel bestimmt. Die sich ergebenden Impulsantworten werden anschließend noch mit dem inversen Frequenzgang des verwendeten Kopfhörers vorentzert. *Bild 5* zeigt ein Beispiel für eine derart entstandene Impulsantwort und den entsprechenden Frequenzgang. Aufgezeichnet sind hier die Paare für das rechte und das linke Ohr bei einem Einfallswinkel von 60 Grad in der Horizontalebene.





**Bild 5 Impulsantwort und Frequenzgang bei 60° Einfallswinkel**

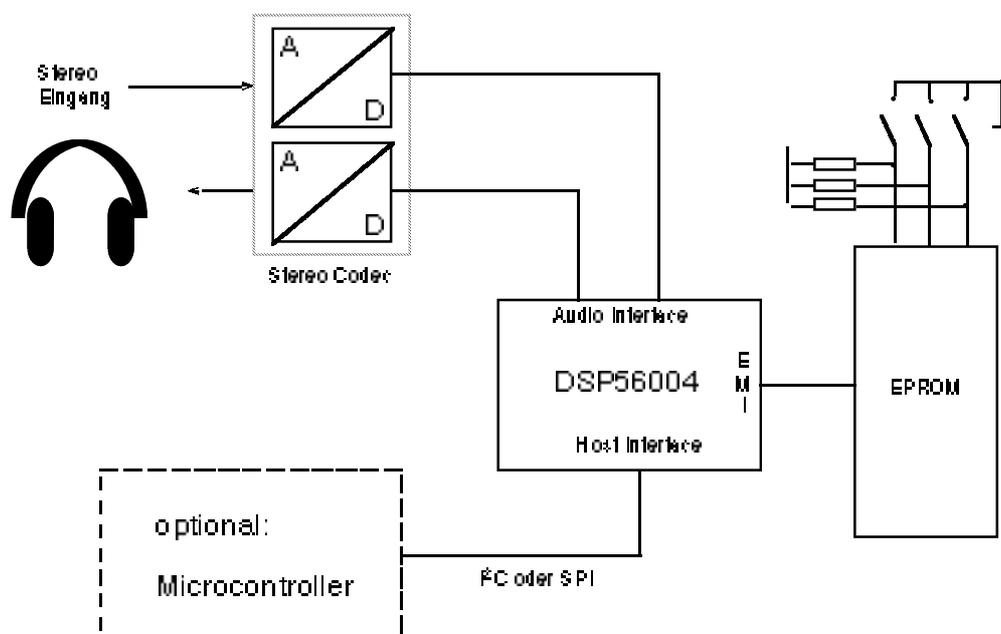
Man kann sehr gut den sich ergebenden Laufzeitunterschied von etwa 0,4 ms zwischen dem schallnahen und dem schallabgewandten Ohr erkennen. Auch die Frequenzgänge weisen sehr interessante Charakteristika auf. Insbesondere die tiefen Einbrüche im Frequenzgang um etwa 8kHz um etwa 20dB fallen sofort auf. Hier zeigen sich Auslöschungen im Innenohr sowie auch Reflektionen und Verstärkungen durch die akustische Wirkung des Schulterbereichs. Die Ohrmuscheln und die abschattende Wirkung des Gesichts bewirken vor allem eine deutliche Absenkungen der hohen Frequenzanteile im indirekten Schallanteil. Wie zu erwarten sind die sich ergebenden Parameter typisch für die gemessene Person. In der Praxis sind die sich ergebenden Unterschiede zwischen verschiedenen Personen jedoch nicht so groß wie anfangs befürchtet, so daß ein praktisch einsetzbares System, mit einer geringen Anzahl von verschiedenen Impulsantworten, fast alle Benutzer zufriedenstellen kann. Hilfreich ist hier auch die erstaunliche Anpassungsfähigkeit des menschlichen Gehirns. So war es in der Praxis fast immer möglich auch mit nicht genau passenden Impulsantworten nach einer kurzen Eingewöhnungszeit von einigen Minuten eine hervorragende Ortbarkeit und Außerkopflokalisation zu erreichen. Das System klingt daher mit häufiger Benutzung immer besser.

Entscheidend für die praktische Realisierbarkeit des beschriebenen Konzeptes sind einmal die Verfügbarkeit von entsprechend guten gemessenen Impulsantworten und die Verfügbarkeit von preiswerten Signalprozessoren, die dennoch leistungsfähig genug sind, um FIR-Filter mit insgesamt mehreren hundert Filter-Taps zu rechnen. Die verwendeten Impulsantworten stammten in diesem Falle aus dem Institut für technische Akustik der Universität Bochum. Diese können für einen kommerziellen Einsatz auch dort lizen-

ziert werden. Es stehen eine große Anzahl von komplett vermessenen "Köpfen" zur Verfügung.

Die verwendete Hardware basiert auf dem sehr kostengünstigen Signalprozessor DSP56004 von Motorola. Dieser erlaubt bei minimalem Hardwareeinsatz den Aufbau eines Systems, welches auch für den Konsumermarkt interessant ist. Bisher bereits auf dem Markt befindliche Systeme dieser Art zielten allein schon durch ihren sehr hohen Preis vor allem auf den professionellen Markt.

Die benötigte minimale Hardware zeigt *Bild 6*. Das Programm und die Impulsantworten ("Hörkurven") liegen hierbei in einem einzelnen 8 Bit breiten EPROM. Verschiedene Hörkurvensätze oder auch verschiedene simulierte Lautsprecherwinkel können sehr einfach z.B. durch Schalter an den universell verwendbaren Leitungen GPIO0..GPIO2 ausgewählt werden. Alternativ kann man durch Umschalten der oberen Adressleitungen des externen EPROMs ebenfalls verschiedene Programme/Parameter einstellen. Bei Anwendungen in Geräten, welche bereits einen Microcontroller enthalten, bietet der DSP56004 zusätzlich die Möglichkeit das Programm mittels des seriellen Host-Interfaces im I<sup>2</sup>C- bzw. SPI-Modus in den DSP zu laden. Über dieselbe Schnittstelle können auch alle Parameter eingestellt werden. Ein externer Speicher am DSP ist in diesem Falle nicht mehr notwendig.



**Bild 6** Blockschaltbild eines kompletten Systems

## Anwendungsbereiche

Neben der Nachbildung der Lautsprecherwiedergabe bietet die beschriebene Technik noch wesentlich mehr Anwendungsmöglichkeiten. Ein sehr interessantes Gebiet wird der Bereich "virtual reality" werden. Hierbei kann mittels binauraler Filterung eine Schallquelle an jeden beliebigen Ort verschoben werden und damit an den virtuellen optischen Ort einer Schallquelle angepaßt werden. Dies ist ein weiterer wesentlicher Schritt hin zu einer wirklichkeitsnahen Abbildung. Dies ist nicht nur zu Schulungszwecken wichtig, sondern wird sicher auch sehr schnell im Spielmarkt Interessenten finden. An der Universität Bochum wird zudem an Systemen gearbeitet, welche die akustische Abbildung eines beliebigen Raumes zum Ziel hat. Auch hier wird durch die binaurale Filterung der Raumreflektionen eine wesentliche Verbesserung erreicht. Derartige Systeme werden es in Zukunft erlauben sehr genaue Abbildungen z.B. von geplanten Konzert Sälen zu simulieren. So können akustische Defizite bereits im Vorfeld erkannt und beseitigt werden. Auch im Tonstudio wird diese Technologie die bereits bestehenden Effekt Systeme erweitern und vielleicht sogar dazu beitragen, daß die beschriebenen Probleme der Kopfhörerwiedergabe bereits bei der Aufnahme vermieden werden. Nicht zuletzt möchte ich noch Anwendungen im Bereich der Hörgerätetechnik und Techniken für Sehbehinderte erwähnen. Durch die "Außer-Kopf-Lokalisation" ist es z.B. wesentlich leichter und angenehmer Sprache zu folgen und in einem Sprachgemisch sich auf einen einzelnen Sprecher zu konzentrieren.

## **Erweiterungsmöglichkeiten**

Wie soeben erwähnt, wird eine sehr interessante Ergänzung die Simulation eines Abhörraumes sein. Damit ist es dann möglich Musik unter verschiedensten Raumbedingungen zu hören. Eventuell wird diese Technik einmal dazu verwendet werden können, die Bedingungen des eigenen Wohnzimmers zu simulieren und so die Aufstellung der Lautsprecher auf recht einfache Weise zu optimieren. Erste, noch sehr einfache Ansätze in dieser Richtung gibt es bereits auf dem Markt.

Die beschriebenen Eigenschaften kombiniert mit der heute sehr kostengünstigen Realisierbarkeit machen die binaurale Filterung zu einem sehr interessanten Anwendungsfeld. Von Zusatzgeräten für Kopfhörer (oder die Integration in Endgeräte wie z.B Verstärker und tragbare Abspielgeräte aller Art), über die professionelle Studioteknik bis hin zu Computerspielen, Simulationen und Techniken für Behinderte bietet sich eine große Palette von Anwendungsmöglichkeiten. Interessenten, welche sich näher mit diesem Thema auseinandersetzen wollen, können von der Motorola Mailbox (Tel.: 089 / 92103-111) ein Programm für das sehr kostengünstige DSP56002EVM herunterladen. Dieses simuliert zwei Lautsprecher in einem Aufstellungswinkel von etwa 60 Grad. Das DSP56002EVM enthält bereits ein 16Bit breites Stereo-Codec und erlaubt auch den direkten Anschluß eines Kopfhörers. Das Programm kann in ein zusätzlich bestückbares externes EPROM geladen werden und ermöglicht dann den Einsatz des DSP56002EVM als eigenständiges System.

```

page      132,60

include  'codec.asm'

;*****
TONE_OUTPUT EQU    HEADPHONE_EN+LINEOUT_EN ; +(4*LEFT_ATTEN)+(4*RIGHT_ATTEN)
TONE_INPUT  EQU    MIC_IN_SELECT+(15*MONITOR_ATTEN)
ir_length  equ     60
;*****

org      x:

datalinks  dsm    ir_length
datarechts dsm    ir_length

org      y:0

include  'coeffs.asm'

org      p:

main

; Initialize SC0, SC1 as GPIO inputs

bclr     #0,x:PCC                ; SC0 - GPIO (PC0)
bclr     #1,x:PCC                ; SC1 - GPIO (PC1)
bclr     #0,x:PCDDR              ; SC0 (PC0) - input
bclr     #1,x:PCDDR              ; SC1 (PC1) - input
movep    #301b,x:IPR             ; IRQA/IRQB/SSI are level 3 interrupts.
move     #TONE_OUTPUT,y0         ; headphones, line out, mute spkr, no attn.
move     y0,x:TX_BUFF_BASE+2
move     #TONE_INPUT,y0         ; no input gain, monitor mute
move     y0,x:TX_BUFF_BASE+3
move     #datalinks,r0          ; setup data and coeffs pointer
move     #datarechts,r1
move     #ir_length-1,m0
move     #ir_length-1,m1

; Main Loop

loop_1
jset     #2,x:SSISR,*            ; Wait for frame sync to pass.
jclr     #2,x:SSISR,*            ; Wait for frame sync.
move     x:RX_BUFF_BASE,a        ; Load the left channel input.
move     x:RX_BUFF_BASE+1,b      ; Load the right channel input.

jclr     #14,X:$FFE4,skip_it
jsr     binaural                 ; ... do the binaural filtering

skip_it move a,x:TX_BUFF_BASE    ; Put value in left channel tx.
move     b,x:TX_BUFF_BASE+1     ; Put value in right channel tx.
jmp     loop_1                  ; Loop back.

;*****

binaural equ     *

move     a,x:(r0)+               ; store new input data in buffers
move     b,x:(r1)+
move     #vorne_direkt,r4
move     #vorne_indirekt,r5

;
; Berechnung LINKS
;

```

```
    clr    a            x:(r0)+,x0  y:(r4)+,y0

    do     #ir_length-1,lp1
    mac    x0,y0,a      x:(r0)+,x0  y:(r4)+,y0 ; links * direkt
    mac    x0,y0,a      x:(r1)+,x0  y:(r5)+,y0 ; rechts * indirekt
lp1      mac    x0,y0,a      (r0)-
        asr    a

;
; Berechnung RECHTS
;

    move   #vorne_direkt,r4
    move   #vorne_indirekt,r5

    clr    b            x:(r1)+,x0  y:(r4)+,y0

    do     #ir_length-1,lp2
    mac    x0,y0,b      x:(r1)+,x0  y:(r4)+,y0 ; rechts * direkt
    mac    x0,y0,b      x:(r0)+,x0  y:(r5)+,y0 ; links * indirekt
lp2      mac    x0,y0,b      (r1)-
        asr    b
        rts

    end

□
```

```
;
; left ele: 0 azi 45
;
vorne_indirekt

dc    $ffdeb7
dc    $ffcb63
dc    $fff553
dc    $004059
dc    $0015e4
dc    $ffca0b
dc    $ffc4b2
dc    $ffd779
dc    $002832
dc    $00547a
dc    $0011b8
dc    $002231
dc    $001179
dc    $ffdc23
dc    $0082b1
dc    $00798e
dc    $ffd8f8
dc    $0065ff
dc    $ffca8d
dc    $ff4c59
dc    $010fbb
dc    $003a04
dc    $00528b
dc    $028f3d
dc    $fe2e55
dc    $03bb18
dc    $1601fa
dc    $1447f2
dc    $0f61d2
dc    $1f91e1
dc    $1d2178
dc    $034315
dc    $ff35e6
dc    $0a3d27
dc    $002890
dc    $f14f55
dc    $fb0cea
dc    $095396
dc    $065cbc
dc    $feaa35
dc    $fe21dd
dc    $04fb91
dc    $0c4a8f
dc    $0b16d8
dc    $05e011
dc    $042359
dc    $03734c
dc    $03f453
dc    $074130
dc    $081b6a
dc    $040568
dc    $ff60fb
dc    $fdd4fc
dc    $ff0bca
dc    $ffaf40
dc    $fd77ac
dc    $faa2ef
dc    $fa9ce2
dc    $fd7e8e
dc    $0046cf
dc    $ffe60d
dc    $fd189c
dc    $fbcf47
dc    $fd7fba
dc    $ffa161
```

```
dc    $ffef71
dc    $ff28f8
dc    $ff10bf
dc    $00a085
dc    $03220d
dc    $042c25
dc    $02de3e
dc    $016b67
dc    $012078
dc    $0076f7
dc    $fec2f7
dc    $fd8e7e
dc    $fd8c1c
dc    $fdd1ff
dc    $fe4bf2
;
; right ele: 0 azi 45
;
vorne_direkt
dc    $020844
dc    $00d284
dc    $fb4bb2
dc    $05fbc7
dc    $fe4b68
dc    $f8464c
dc    $0cefce
dc    $f5913e
dc    $006394
dc    $18d647
dc    $dalb28
dc    $241627
dc    $7eb851
dc    $383916
dc    $64f7ec
dc    $4423e0
dc    $9b2e56
dc    $f17b2f
dc    $3763f3
dc    $a65783
dc    $b6fb23
dc    $445f1e
dc    $353686
dc    $e8af82
dc    $eb90cf
dc    $ffdea0
dc    $0737e5
dc    $15b53f
dc    $08d114
dc    $ebe997
dc    $f67cda
dc    $10edd3
dc    $0e923d
dc    $ff7585
dc    $f663d8
dc    $f08753
dc    $f28c0f
dc    $fb4145
dc    $fe6d1e
dc    $fac704
dc    $f83d94
dc    $fa28f8
dc    $fe2396
dc    $00cb2b
dc    $012c72
dc    $ff3b00
dc    $faf71b
dc    $f94987
dc    $fc44c2
dc    $fd87f9
```

```
dc    $fb73a1
dc    $fccd17
dc    $00faa8
dc    $01cb5c
dc    $011e28
dc    $027ad4
dc    $034d6f
dc    $019817
dc    $fe0b08
dc    $fa7cac
dc    $fac0f5
dc    $ff078a
dc    $028948
dc    $04537b
dc    $07057e
dc    $09eedc
dc    $0a4cda
dc    $06068b
dc    $fddd5a
dc    $f8a468
dc    $fbb6c3
dc    $0116bb
dc    $001d67
dc    $faddc6
dc    $f9918f
dc    $fe0d88
dc    $021cdc
dc    $00e421
dc    $fd6198
dc    $fc89e2
```

□

```

        page      132,60
;*****
;   CODEC.ASM
;   Initialization program for EVM56002 to communicate with CS4215
;   input connected to Mic inputs
;
;   Copywrite (c) MOTOROLA 1994
;           Semiconductor Products Sector
;           Digital Signal Processing Division
;
;*****
;
;   portc usage:
;   bit8: SSI TX (from DSP to Codec)
;   bit7:
;   bit6:
;   bit5:
;   bit4: codec reset (from DSP to Codec)
;   bit3:
;   bit2: data/control bar
;           0=control
;           1=data
;
;
;   PROGRAM OUTLINE:
;
;1 program fsync and sclk == output
;2 write pc0 = 0 (control mode)
;3 send 64 bit frame x times, with dcb bit = 0, keep doing until read back as 0
;4 send 64 bit frame x times, with dcb bit = 1, keep doing until read back as 1
;5 re-program fsync and sclk == input
;6 write pc0 = 1 (data mode)
;7 receive/send data (echo slots 1,2,3,4; slots 5,6,7,8 == DIP switched)
;
;*****

```

PAGE 255

```

NO_PREAMP      equ      $100000
LO_OUT_DRV     equ      $080000
HI_PASS_FILT   equ      $008000
SAMP_RATE_9    equ      $003800
SAMP_RATE_48   equ      $003000
SAMP_RATE_44   equ      $002800
SAMP_RATE_37   equ      $002000
SAMP_RATE_32   equ      $001800
SAMP_RATE_27   equ      $001000
SAMP_RATE_16   equ      $000800
SAMP_RATE_8    equ      $000000
STEREO         equ      $000400
DATA_8LIN      equ      $000300
DATA_8A        equ      $000200
DATA_8U        equ      $000100
DATA_16        equ      $000000
IMMED_3STATE   equ      $800000
XTAL1_SELECT   equ      $100000
XTAL2_SELECT   equ      $200000
BITS_64        equ      $000000
BITS_128       equ      $040000
BITS_256       equ      $080000
CODEC_MASTER   equ      $020000
CODEC_TX_OFF   equ      $010000

CTRL_WD_12     equ      NO_PREAMP+HI_PASS_FILT+SAMP_RATE_48+STEREO+DATA_16 ;CLB=0
CTRL_WD_34     equ      IMMED_3STATE+XTAL2_SELECT+BITS_64+CODEC_MASTER
CTRL_WD_56     equ      $000000
CTRL_WD_78     equ      $000000

HEADPHONE_EN   equ      $800000
LINEOUT_EN     equ      $400000
LEFT_ATTEN     equ      $010000 ;63*LEFT_ATTEN = -94.5 dB, 1.5 dB steps

```

```

SPEAKER_EN      equ      $004000
RIGHT_ATTEN     equ      $000100 ;63*RIGHT_ATTEN = -94.5 dB, 1.5 dB steps
MIC_IN_SELECT   equ      $100000
LEFT_GAIN       equ      $010000 ;15*LEFT_GAIN = 22.5 dB, 1.5 dB steps
MONITOR_ATTEN   equ      $001000 ;15*MONITOR_ATTEN = mute, 6 dB steps
RIGHT_GAIN      equ      $000100 ;15*RIGHT_GAIN = 22.5 dB, 1.5 dB steps

OUTPUT_SET      equ      HEADPHONE_EN+LINEOUT_EN+(LEFT_ATTEN*4)
INPUT_SET       equ      MIC_IN_SELECT+(15*MONITOR_ATTEN)+(RIGHT_ATTEN*4)

;---DSP56002 on-chip peripheral addresses
PCD             equ      $FFE5
PCDDR          equ      $FFE3
PCC            equ      $FFE1
PBC            equ      $FFE0
CRA            equ      $FFEC
CRB            equ      $FFED
SSIDR         equ      $FFEF
IPR           equ      $FFFF
BCR           equ      $FFFE
SSISR         equ      $FFEE
PLL           equ      $FFFD
START         equ      $40

      org      x:0
RX_BUFF_BASE   equ      *
RX_data_1_2    ds      1      ;data time slot 1/2 for RX ISR
RX_data_3_4    ds      1      ;data time slot 3/4 for RX ISR
RX_data_5_6    ds      1      ;data time slot 5/6 for RX ISR
RX_data_7_8    ds      1      ;data time slot 7/8 for RX ISR

TX_BUFF_BASE   equ      *
TX_data_1_2    ds      1      ;data time slot 1/2 for TX ISR
TX_data_3_4    ds      1      ;data time slot 3/4 for TX ISR
TX_data_5_6    ds      1      ;data time slot 5/6 for TX ISR
TX_data_7_8    ds      1      ;data time slot 7/8 for TX ISR

RX_PTR         ds      1      ; Pointer for rx buffer
TX_PTR         ds      1      ; Pointer for tx buffer

      org      p:0
      jmp     START

      org      p:$C
      jsr     ssi_rx_isr      ;SSI RX
      jsr     ssi_rx_isr      ;SSI RX w/Exception
      jsr     ssi_tx_isr      ;SSI TX
      jsr     ssi_tx_isr      ;SSI TX w/Exception

;-----
;
;      Main codec initialization program
;
;-----

      org      p:START

      movep   #$26100B,x:PLL      ;set PLL for MPY of 12x (48MHz)
      movep   #$0000,x:BCR      ; number of wait states
      ori     #3,mr              ;disable interrupts
      movec   #0,sp
      move    #0,omr            ; single chip mode

      move    #$1F0,r6          ; initialize stack pointer
      move    #-1,m6           ; linear addressing
      move    #RX_BUFF_BASE,x0
      move    x0,x:RX_PTR      ; Initialize the rx pointer
      move    #TX_BUFF_BASE,x0
      move    x0,x:TX_PTR      ; Initialize the tx pointer
      jsr     codec_init

```

; The following line of code jumps to "main" which is the location for  
; the main code that will run after the codec is initialized.

```
    jmp main
```

```
;*****
;*****      initialize the CS4215 codec      *****
;*****
; headphones and line out, and set up for no gain or attenuation, and no
; monitor feedback.
;*****
;*****
;
;    initialize ssi -- fsync and sclk ==> outputs
;
codec_init
    movep    #$0000,x:PCC      ; turn off ssi port
    movep    #$4303,x:CRA      ; 40MHz/16 = 2.5MHz SCLK, WL=16 bits, 4W/F
    movep    #$FB30,x:CRB      ; RIE,TIE,RE,TE, NTKW, SYN, FSR/RSR->bit
    movep    #$14,x:PCDDR      ; setup pc2 and pc4 as outputs
    movep    #$0,x:PCD         ; D/C~ and RESET~ = 0 ==> control mode
                                ;----reset delay for codec ----
    do      #500,_delay_loop
    rep     #2000                ; 100 us delay
    nop
_delay_loop
    bset    #4,x:PCD            ; RESET~ = 1
    movep   #$3000,x:IPR        ; set interrupt priority level
    andi   #$FC,mr             ; enable interrupts

;*****
; The following data sets up the CS4215 control mode data:
; (CTS = Control Time Slot, U/LN = upper/lower Nibble)
;
;    +----- CTS1-UN:   0       0       1       MLB       0 0 0 0
;    |+----- CTS1-LN:  OLB     CLB     X       X         0 0 0 0
;    ||+----- CTS2-UN:  HPF     X       DFR2    DFR1     0 0 1 0
;    |||+---- CTS2-LN:  DFR0    ST      DF1     DF0      1 1 0 0
; x0 = $002Cxx
;
;    +----- CTS3-UN:   ITS     MCK2    MCK1    MCK0     1 0 0 0
;    |+----- CTS3-LN:  BSEL1   BSEL0   XCLK    XEN      1 0 0 0
;    ||+----- CTS4-UN:  TEST    TEST    TEST    TEST     (TEST MUST BE 0)
;    |||+---- CTS4-LN:  TEST    TEST    ENL     DAD      0 0 0 0
; x0 = $8800xx
;*****

;--- set up buffer with control mode data
    move    #CTRL_WD_12,x0
    move    x0,x:TX_BUFF_BASE
    move    #CTRL_WD_34,x0
    move    x0,x:TX_BUFF_BASE+1
    move    #CTRL_WD_56,x0
    move    x0,x:TX_BUFF_BASE+2
    move    #CTRL_WD_78,x0
    move    x0,x:TX_BUFF_BASE+3

    movep   #$01E8,x:PCC      ; Turn on ssi port

;
; CLB == 0
;
    jclr   #3,x:SSISR,*      ; wait until rx frame bit==1
    jset   #3,x:SSISR,*      ; wait until rx frame bit==0
    jclr   #3,x:SSISR,*      ; wait until rx frame bit==1
    jset   #18,x:RX_BUFF_BASE,* ; loop until CLB set

;
; CLB == 1
```

```

;
    bset    #18,x:TX_BUFF_BASE    ;set CLB
    do      #4,_init_loopB
    jclr    #2,x:SSISR,*          ; wait until tx frame bit==1
    jset    #2,x:SSISR,*          ; wait until tx frame bit==0
_init_loopB
    movep   #0,x:PCC              ;disable, reset SSI

;*****
;    now CLB should be 1 -- re-program fsync and sclk direction (i/p) -- also,
;    circular buffer pointers for echoing data r0=current, r1=old data to send
;    1 frame later
;
    movep   #$4303,x:CRA          ; 16bits,4 word/frame, /2/4/2=2.5 MHz
    movep   #$FB00,x:CRB          ; rcv,xmt & int ena,netwk,syn,sclk==i/p,msb 1st
    movep   #$14,x:PCD           ; D/C~ pin = 1 ==> data mode
    movep   #$01E8,x:PCC         ; turn on ssi port
    rts

;***** SSI TRANSMIT ISR *****
ssi_tx_isr
    move    r0,x:(r6)+            ; Save r0 to the stack.
    move    m0,x:(r6)+            ; Save m0 to the stack.
    move    #3,m0                 ; Modulus 4 buffer.
    move    x:TX_PTR,r0           ; Load the pointer to the tx buffer.
    jclr    #2,x:SSISR,next_tx    ; If not frame sync, jump to transmit data.
    move    #TX_BUFF_BASE+1,r0    ; If frame sync, reset pointer.
    nop

next_tx
    movep   x:(r0)+,x:SSIDR        ; SSI transfer data register.
    move    r0,x:TX_PTR           ; Update tx buffer pointer.
    move    x:-(r6),m0            ; Restore m0.
    move    x:-(r6),r0            ; Restore r0.
    rti

;***** SSI receive ISR*****
ssi_rx_isr
    move    r0,x:(r6)+            ; Save r0 to the stack.
    move    m0,x:(r6)+            ; Save m0 to the stack.
    move    #3,m0                 ; Modulo 4 buffer.
    move    x:RX_PTR,r0           ; Load the pointer to the rx buffer.
    jclr    #3,x:SSISR,next_rx    ; If not fr. sync, jump to receive data.
    move    #RX_BUFF_BASE,r0      ; If frame sync, reset base pointer.
    nop

next_rx
    movep   x:SSIDR,x:(r0)+        ; Read out received data to buffer.
    move    r0,x:RX_PTR           ; Update rx buffer pointer.
    move    x:-(r6),m0            ; Restore m0.
    move    x:-(r6),r0            ; Restore r0.
    rti

```

□

München, den 10.5.95

Beiliegend finden Sie den Sourcecode für ein Programm, welches das sehr leidige Problem der "Im-Kopf-Lokalisation" bei der Kopfhörerwiedergabe behebt. Es ist lauffähig auf dem DSP56002EVM von Motorola.

Die hierin verwendeten Hörkurven sind Eigentum der Universität Bochum und können für kommerzielle Zwecke lizenziert werden (Adresse am Ende dieses Textes). Für die private Benutzung ist im Programm ein Hörkurvensatz für einen Winkel von etwa 50 Grad enthalten. Da die Hörkurven mit 44.1kHz abgetastet wurden, kann die optimale Qualität der Wiedergabe nur mit einem zusätzlichen Quarz auf dem DSP56002EVM erreicht werden. (siehe Dokumentation des Wandlers CS4215). Im Programmcode muss dabei im File "Codec.asm" folgendes geändert werden:

```
CTRL_WD_12 equ NO_PREAMP+HI_PASS_FILT+SAMP_RATE_48+STEREO+DATA_16 ;CLB=0
CTRL_WD_34 equ IMMED_3STATE+XTAL2_SELECT+BITS_64+CODEC_MASTER
```

in

```
CTRL_WD_12 equ NO_PREAMP+HI_PASS_FILT+SAMP_RATE_44+STEREO+DATA_16 ;CLB=0
CTRL_WD_34 equ IMMED_3STATE+XTAL1_SELECT+BITS_64+CODEC_MASTER
```

Prinzipiell läuft das Programm jedoch auch in der beiliegenden Form mit 48kHz Abtastrate.

In dem File "BinArtkl.exe" auf der Motorola Mailbox finden Sie eine genauere Beschreibung des verwendeten Verfahrens zur Vermeidung der "Im-Kopf-Lokalisation".

Andreas Straub  
Field Application DSP CE  
Motorola München

-----  
ergänzende Literatur

/1/ J. Blauert, Räumliches Hören, S.Hirzel Verlag, Stuttgart 1974 /2/ P. Pfleiderer, HiFi auf den Punkt gebracht, Pflaum Verlag, München 1990 /3/ H. Lehnert, Ruhr Universität Bochum, Erzeugung von virtuellen akustischen Umgebungen, Tonmeistertagung 1990

Kontaktadresse für die verwendeten Hörkurven:

Dr. Ing. Hilmar Lehnert  
Lehrstuhl fuer Allgemeine Elektrotechnik und Akustik  
Ruhr-Universitaet Bochum, IC 1/132  
D-44780 Bochum  
Fax: +49 234-7094-165  
Phone: +49 234-700-2490