

Übung zur Vorlesung
Digitale Medien

Ludwig-Maximilians-Universität München
Wintersemester 2010/2011

6. Übungsblatt

Hexadezimalsystem

Little-Endian vs. Big-Endian

WAVE-Format

Hexeditor

Zeigt einzelne Bytes einer Datei an

z.B. Linux:

Okteta

Windows:

frhed (<http://frhed.sourceforge.net/>)

Mac OS X:

HexEdit (<http://hexedit.sourceforge.net/>)

The screenshot shows a hex editor window titled "[C:\Documents and Settings\dominikus\Desktop\ue6\secret_message.wav] - frhed". The window displays the raw data of a WAV file. The first few lines show the RIFF header and the 'data' chunk. The data is presented in three columns: hexadecimal values, a space-separated list of those values, and their corresponding ASCII characters. The ASCII column shows a mix of readable characters and 'yy' placeholders for non-printable bytes.

```

000000 52 49 46 46 24 48 0f 00 57 41 56 45 66 6d 74 20 10 00 00 00 01 00 02 00 44 ac RIFF SH WAVfmt          D-
00001a 00 00 10 b1 02 00 04 00 10 00 64 61 74 61 00 48 0f 00 fe ff 02 00 00 00 fe ff . . . data.H. by by
000034 04 00 fb ff 04 00 fe ff 01 00 00 00 ff ff 02 00 fd ff 03 00 ff ff fe ff 04 00 . . . by by yy yy yy yy
00004e fb ff 05 00 fc ff 03 00 fd ff 03 00 fe ff 01 00 00 00 ff ff 01 00 00 00 ff ff . . . by by yy by yy yy
000068 02 00 fe ff 00 00 02 00 fd ff 03 00 fe ff 01 00 00 00 ff ff 02 00 fd ff 03 00 . . . by yy by yy yy
000082 fe ff 01 00 00 00 02 00 00 00 00 00 00 00 00 00 00 00 01 00 fe ff 03 00 by . . . by
00009c fd ff 02 00 fe ff 03 00 fd ff 03 00 fe ff 00 00 01 00 ff ff 00 00 01 00 ff ff yy by yy by yy yy yy
0000b6 00 00 01 00 fe ff 02 00 ff ff 00 00 01 00 fe ff 03 00 fd ff 02 00 00 00 fe ff . . . by yy by yy by yy
0000d0 03 00 fd ff 03 00 fe ff 00 00 01 00 fe ff 02 00 fe ff 01 00 ff ff 01 00 ff ff yy by yy yy by yy
0000ea 00 00 ff ff 02 00 fd ff 03 00 fd ff 02 00 ff ff 00 00 01 00 fe ff 02 00 ff ff . . . yy yy yy yy by yy
000104 00 00 00 00 01 00 ff ff 01 00 ff ff 01 00 ff ff 01 00 ff ff 01 00 00 00 ff ff . . . yy yy yy yy yy
00011e 01 00 fe ff 03 00 fd ff 03 00 fd ff 02 00 ff ff 00 00 01 00 ff ff 01 00 ff ff . . . by yy yy yy yy yy
000138 01 00 ff ff 02 00 fd ff 04 00 fb ff 06 00 fa ff 05 00 fd ff 00 00 03 00 fc ff . . . yy yy uy yy yy uy
000152 03 00 ff ff ff ff 02 00 fe ff 02 00 fe ff 02 00 fe ff 02 00 fe ff 02 00 fe ff . . . yyy by by by yyy
00016c fe ff 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 ff ff 01 00 by . . . yy yy yy
000186 fe ff 02 00 ff ff 00 00 01 00 00 00 fe ff 03 00 fd ff 02 00 ff ff 01 00 ff ff by yy . . . by yy yy yy
0001a0 01 00 ff ff 00 00 02 00 fd ff 03 00 fe ff 01 00 00 00 ff ff 03 00 fe ff . . . yy yy by yy by
0001ba 01 00 00 00 00 00 ff ff 01 00 00 00 ff ff 02 00 fe ff 01 00 ff ff 01 00 00 00 . . . yy yy by yy
0001d4 00 00 00 00 ff ff 00 00 01 00 00 00 ff ff 02 00 fe ff 01 00 ff ff 02 00 fe ff 03 00 by . . . by yy by yy
0001ee fe ff 00 00 01 00 ff ff 01 00 fe ff 02 00 fe ff 03 00 fd ff 01 00 00 00 00 00 . . . by yy by yy yy
000208 00 00 00 00 00 00 00 00 01 00 fe ff 02 00 ff ff 00 00 01 00 ff ff 00 00 01 00 . . . by yy by yy
000222 fe ff 03 00 fc ff 04 00 fc ff 03 00 ff ff 00 00 ff ff 02 00 fe ff 01 00 01 00 . . . by uy uy yy yy
00023c fe ff 03 00 fc ff 03 00 ff ff 00 00 01 00 fe ff 02 00 fe ff 03 00 fc ff 04 00 . . . by uy yy by by uy
000256 fd ff 02 00 ff ff 00 00 01 00 fe ff 03 00 fd ff 02 00 ff ff 00 00 00 00 01 00 . . . yy yy by yy yy
000270 fe ff 02 00 ff ff ff ff 02 00 ff ff 00 00 00 00 00 00 00 00 01 00 fe ff 02 00 . . . by yyy by . . . by
00028a fe ff 02 00 fe ff 01 00 01 00 fe ff 01 00 00 00 00 00 00 00 00 00 01 00 fd ff 03 00 by by . . . by
0002a4 ff ff ff ff 02 00 fe ff 00 00 02 00 fe ff 01 00 00 00 ff ff 01 00 00 00 ff ff . . . yyy by . . . yy
0002be 02 00 fe ff 02 00 fe ff 02 00 fd ff 04 00 fd ff 02 00 ff ff 00 00 00 00 01 00 . . . by by yy yy yy
0002d8 ff ff 00 00 01 00 ff ff 01 00 00 00 ff ff 01 00 ff ff 01 00 00 00 ff ff 01 00 . . . yy . . . yy yy yy
0002f2 ff ff 01 00 00 00 fe ff 03 00 fd ff 03 00 ff ff 01 00 00 00 ff ff 03 00 . . . yy . . . yy yyy yy
00030c fc ff 04 00 fd ff 02 00 fe ff 01 00 01 00 01 00 ff ff 01 00 fe ff 01 00 01 00 ff ff . . . uy yy by . . . by yy
000326 01 00 fe ff 02 00 fc ff 00 00 02 00 fd ff 03 00 fc ff 04 00 fd ff 03 00 fe ff . . . by yy by yy yy by
000340 ff ff 03 00 fc ff 05 00 fb ff 03 00 fd ff 00 00 01 00 ff ff 00 00 01 00 fe ff . . . yy uy uy yy yy by
00035a 03 00 fd ff 03 00 fe ff 00 00 02 00 fd ff 03 00 ff ff ff 02 00 fe ff 02 00 . . . yy by . . . yyy by
000374 fe ff 02 00 fe ff 02 00 ff ff ff ff 02 00 fd ff 03 00 ff ff ff ff 02 00 fe ff . . . by by yyy yy yyy by
00038e 01 00 00 00 00 00 00 00 01 00 ff ff 00 00 00 00 00 00 00 00 01 00 ff ff 01 00 . . . yy yy . . . yy by
0003a8 02 00 ff ff 01 00 ff ff 01 00 fe ff 02 00 ff ff 01 00 ff ff 01 00 ff ff 01 00 fd ff 05 00 . . . yy yy by yy yy yy
0003c2 fb ff 04 00 fe ff 00 00 01 00 ff ff 01 00 ff ff 01 00 ff ff 01 00 00 00 ff ff . . . uy by . . . yy yy yy
0003dc 01 00 00 00 00 00 00 00 00 00 ff ff 02 00 fe ff 03 00 fc ff 03 00 fe ff 01 00 . . . yy . . . by uy by
0003f6 01 00 ff ff ff ff 02 00 fd ff 04 00 fd ff 02 00 fe ff 03 00 fd ff 03 00 ff ff ff ff . . . yyy yy byyy yy yyy
000410 02 00 fd ff 02 00 00 00 ff ff 01 00 00 00 fe ff 03 00 fd ff 03 00 fe ff 00 00 . . . yy . . . by by by
00042a 02 00 fd ff 04 00 fb ff 04 00 fd ff 03 00 fe ff 01 00 fe ff 02 00 ff ff 00 00 . . . yy uy yy by by by
000444 01 00 fe ff 02 00 fe ff 02 00 fe ff 02 00 fe ff 02 00 fe ff 02 00 fe ff 01 00 . . . by by by by by by
00045e 01 00 fe ff 02 00 fe ff 02 00 fe ff 02 00 fe ff 02 00 fe ff 03 00 fc ff 04 00 . . . by by by by by uy
000478 fc ff 03 00 ff ff 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 . . . uy yy . . . yy yy
000492 00 00 00 00 00 00 ff ff 03 00 fc ff 04 00 fd ff 00 00 04 00 fa ff 07 00 f9 ff . . . . . yy uy yy . . . uy uy
0004ac 04 00 00 00 fe ff 03 00 fd ff 02 00 ff ff 01 00 fe ff 03 00 fd ff 02 00 00 00 . . . by yy yy by yy
  
```

Offset 3=0x3 Bits=01000110 Unsigned: B:70,W:9286,L:256386118 ANSI / OVR / L Size: 1001516

Hexadezimal nach Dezimal

A4C₁₆ (0xA4C)

$$\mathbf{C} * 16^0 + \mathbf{4} * 16^1 + \mathbf{A} * 16^2 =$$

$$12 * 1 + 4 * 16 + 10 * 256 =$$

$$12 + 64 + 2560 =$$

2636₁₀

Google

"0xXXXX to decimal"

"XXXXXX to hex"

Hexadecimal	Binary	Decimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
A	1010	10
B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15

Hexadezimal nach Dezimal

0xFF

0x7F

0x1C0

Hexadecimal	Binary	Decimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
A	1010	10
B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15

Dezimal nach Hexadezimal

Die Dezimalzahl **76528975** wird ins Hexadezimalsystem umgewandelt.

Gehe nach folgendem Verfahren vor:

- (1) Teile die Zahl mit Rest durch 16.
- (2) Der Divisionsrest ist die nächste Ziffer (von rechts nach links). Für Reste > 9 nimm die Buchstaben A, B, C, D, E, F
- (3) Falls der (ganzzahlige) Quotient = 0 ist, bist du fertig, andernfalls nimm den (ganzzahligen) Quotienten als neue Zahl und wiederhole ab (1).

```
76528975 : 16 = 4783060 Rest: 15 --> Ziffer: F
4783060  : 16 = 298941  Rest: 4  --> Ziffer: 4
298941   : 16 = 18683   Rest: 13 --> Ziffer: D
18683    : 16 = 1167    Rest: 11 --> Ziffer: B
1167     : 16 = 72      Rest: 15 --> Ziffer: F
72       : 16 = 4       Rest: 8  --> Ziffer: 8
4        : 16 = 0       Rest: 4  --> Ziffer: 4
```

Resultat: **48FBD4F**

Big-Endian versus Little-Endian

Byteweise Ausleserichtung:

Big-Endian: Höchster Wert zuerst

$$234 \Rightarrow 2 * 100 + 3 * 10 + 4 * 1$$

Little-Endian: Niedrigster Wert zuerst

$$234 \Rightarrow 2 * 1 + 3 * 10 + 4 * 100$$

Sprache:

24 = „twenty-four“ (Englisch – Big-Endian)

24 = „vierundzwanzig“ (Deutsch – Little-Endian)

Beispiel:

439041101 als 32-Bit Integer

Binär:

00011010 00101011 00111100 01001101

Hex:

1A 2B 3C 4D

Adresse	Big Endian			Little Endian		
	Hex	Dez	Binär	Hex	Dez	Binär
10000	1A	26	00011010	4D	77	01001101
10001	2B	43	00101011	3C	60	00111100
10002	3C	60	00111100	2B	43	00101011
10003	4D	77	01001101	1A	26	00011010

33 FA CC 00

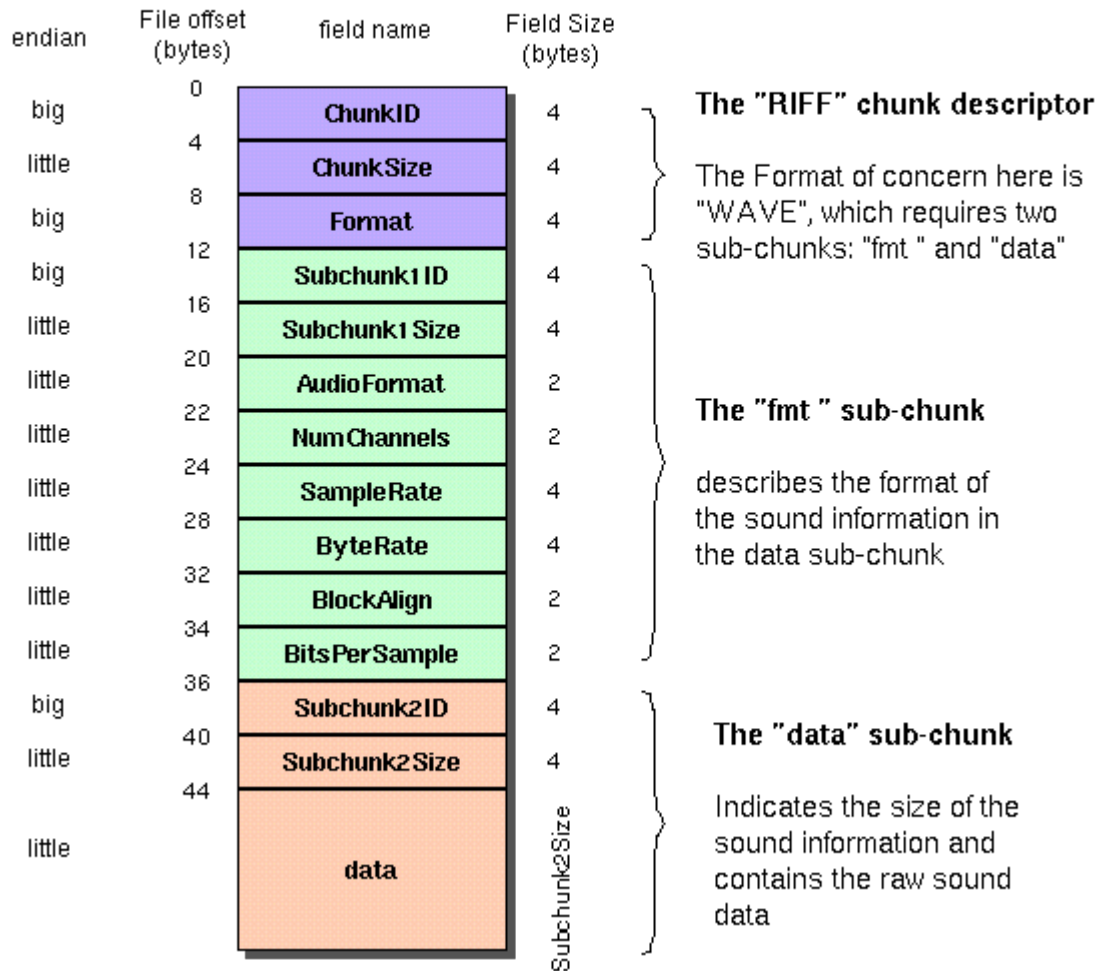
Big-Endian: 0x33FACC00 => 872 074 240

Little-Endian: 0x00CCFA33 => 13 433 395

00 00 0F 11

Big-Endian: 0x00000F11 => 3 857

Little-Endian: 0x110F0000 => 286 195 712



Beispiel:

110-220-440-880.wav

```

0000  52 49 46 46  RIFF
0004  1c 30 14 00  .0..
0008  57 41 56 45  WAVE
000c  66 6d 74 20  fmt
0010  10 00 00 00  ....
0014  01 00 01 00  ....
0018  44 ac 00 00  D~..
001c  44 ac 00 00  D~..
0020  01 00 08 00  ....
0024  64 61 74 61  data
0028  f8 2f 14 00  ø/~..
002c  80 88 90 97  ....
0030  9f a7 af b6  .S~¶
0034  bd c4 cb d1  %ÄËÑ
0038  d7 dd e2 e7  xÿâç
    
```

Meta-Daten aus iTunes:

```

Kind: WAV audio file
Size: 44 KB
Bit Rate: 352 kbps
Sample Rate: 44,100 kHz
Channels: Mono
Sample Size: 8 bit
    
```

<https://ccrma.stanford.edu/courses/422/projects/WaveFormat/>

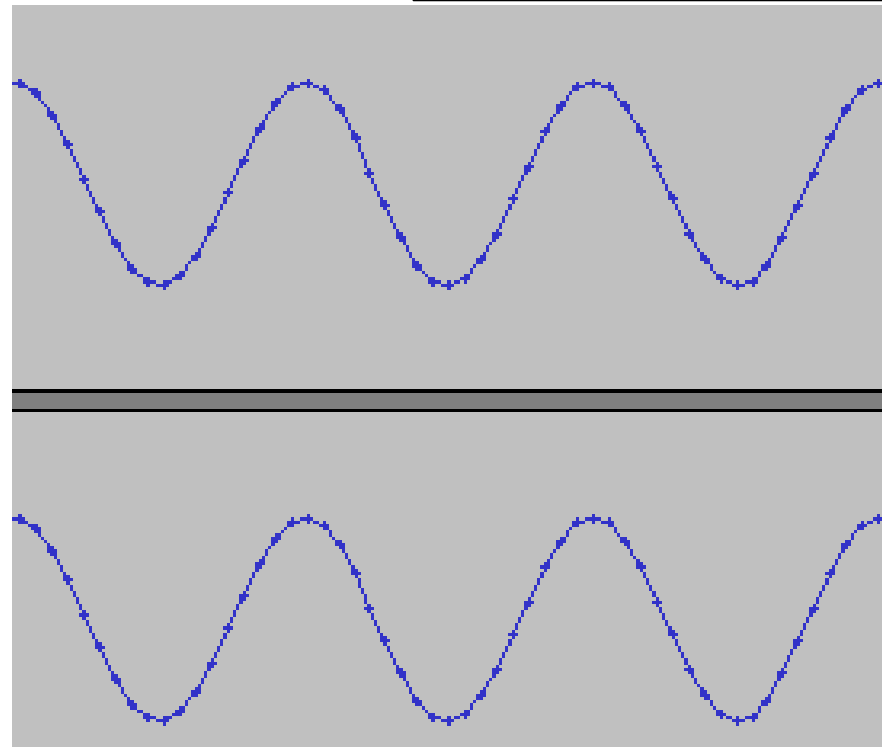
WAVE-Format:

<http://www.sonicspot.com/guide/wavefiles.html>

<https://ccrma.stanford.edu/courses/422/projects/WaveFormat/>

```

00024 64 61 da
00026 74 61 ta
00028 80 a9 .@
0002a 03 00 ..
0002c 80 7f ..
0002e 95 95 ...
00030 a8 a8 ..
00032 b7 b7 ..
00034 be be %%
00036 bf bf ^^
00038 b8 b8 ..
0003a aa aa áá
0003c 97 97 ..
0003e 82 82 ..
00040 6c 6c ll
00042 58 58 XX
00044 49 49 II
00046 41 41 AA
00048 40 40 @@
0004a 46 46 FF
0004c 54 54 TT
0004e 66 66 ff
00050 7b 7b {{
00052 91 91 ``
00054 a5 a5   
00056 b4 b4 ``
00058 bd bd %%
0005a bf bf ^^
0005c b9 b9 11
0005e ad ad --
00060 9b 9b ..
00062 86 86 ..
00064 70 70 pp
00066 5c 5c \
    
```



8-Bit Stereo

Interleaving

Die beiden Stereospuren werden abwechselnd in der Datei abgelegt