**Vorwort:** Die folgenden Ausführungen befassen sich mit einem mathematischen Thema, das aber nicht wie im Mathe-Buch, sondern verständlich erläutert vorgestellt wird. Da der Verfasser aus der Elektrotechnik kommt, hat er zum Teil andere Sichtweisen auf die Dinge. Vielleicht hilft gerade diese Besonderheit beim Verständnis der beschriebenen Mathematik.

**Achtung:** Trotz aller Sorgfalt bei der Erarbeitung kann keine Garantie für die Richtigkeit der Inhalte übernommen werden.

---

**Eine Bemerkung zuvor:** Zahlensysteme – abgesehen von solchen Varianten wie den komplexen Zahlen – funktionieren letztlich alle gleich. Deshalb greife ich zu Beginn der Erläuterungen immer wieder auf die Dezimalzahlen zurück. Wenn man sich deren Funktion verdeutlicht, hat man in der Regel auch schon das andere/neue Zahlenformat verstanden. Probieren wir es aus …

---

# Inhaltsübersicht Hexadezimalzahlen

# Grundlagen der Hexadezimalzahlen

**Zum Verständnis vorab die Grundregeln der Dezimalzahlen**

1. Jede Dezimal*ziffer* kann 10 verschiedene Zustände annehmen:

   0, 1, 2, 3, ... , 9

2. Die Basisdes Dezimalzahlensystems ist entsprechend die Zahl 10.

3. Die höchstwertigste Ziffer ist die erste Ziffer der Dezimalzahl. Die niederwertigste Ziffer der Dezimalzahl ist die letzte Ziffer.

4. Die Wertigkeit einer zum Beispiel fünfstelligen und ganzzahligen Dezimalzahl VWXYZ berechnet sich wie folgt:

| | | | | | |
|---|---|---|---|---|---|
| Dezimalzahl: | **V** | **W** | **X** | **Y** | **Z** |
| Ziffernnummer: | 4 | 3 | 2 | 1 | 0 |

   Wertigkeit: $V \cdot 10^4 + W \cdot 10^3 + X \cdot 10^2 + Y \cdot 10^1 + Z \cdot 10^0$

5. Handelt es sich um eine rationale Dezimalzahl V,WXYZ mit Kommastellen, sieht das Ziffernschema zum Beispiel wie folgt aus:

| | | | | | |
|---|---|---|---|---|---|
| Dezimalzahl: | **V,** | **W** | **X** | **Y** | **Z** |
| Ziffernnummer: | 0 | -1 | -2 | -3 | -4 |

   Wertigkeit: $V \cdot 10^0 + W \cdot 10^{-1} + X \cdot 10^{-2} + Y \cdot 10^{-3} + Z \cdot 10^{-4}$
   $= V \cdot 1 + W \cdot 0{,}1 + X \cdot 0{,}01 + Y \cdot 0{,}001 + Z \cdot 0{,}0001$

**Nun die Grundregeln der Hexadezimalzahlen**

1. Jede Hexadezimal*ziffer* kann 16 verschiedene Zustände annehmen:

   0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E und F

2. Die Basis des Hexadezimalzahlensystems ist entsprechend die Zahl 16.

3. Eine Ziffer einer Hexadezimalzahl entspricht einer Binär- bzw. Dualzahl mit 4 Bit. Denn mit 4 Bit – einer sogenannten Tetrade – können die Zahlen von

   $0_{dezimal} = 0000_{binär} = 0_{hexadezimal} = 0H$ bis
   $15_{dezimal} = 1111_{binär} = F_{hexadezimal} = FH$

   dargestellt werden. Das ist genau der Wertebereich für eine Hexadezimalziffer. Eine zweistellige Hexadezimalzahl entspricht entsprechend einer Binär- bzw. Dualzahl mit 8 Bit:

   $0_{dezimal} = 0000\ 0000_{binär} = 00_{hexadezimal} = 00H$ bis
   $15_{dezimal} = 1111\ 1111_{binär} = FF_{hexadezimal} = FFH.$

4. Die Wertigkeit einer zum Beispiel vierstelligen und ganzzahligen Hexadezimalzahl QRST berechnet sich wie folgt:

| | | | | | |
|---|---|---|---|---|---|
| Hexadezimalzahl: | **Q** | **R** | **S** | **T** | **H** |
| Ziffernnummer: | 3 | 2 | 1 | 0 | |

Wertigkeit: $Q \cdot 16^3 + R \cdot 16^2 + S \cdot 16^1 + T \cdot 16^0$
$= Q \cdot 4096 + R \cdot 256 + S \cdot 16 + T \cdot 1$

5. Über die dargestellte Summation der Wertigkeiten der einzelnen Hexadezimalziffern kann man eine beliebige Hexadezimalzahl in eine Dezimalzahl umrechnen:

| | | | | | |
|---|---|---|---|---|---|
| Hexadezimalzahl: | **1** | **5** | **A** | **3** | **H** |
| Ziffernnummer: | 3 | 2 | 1 | 0 | |

Wertigkeit: $1 \cdot 16^3 + 5 \cdot 16^2 + A \cdot 16^1 + 3 \cdot 16^0$
$= 1 \cdot 4096 + 5 \cdot 256 + 10 \cdot 16 + 3 \cdot 1$
$= 5539_{dezimal}$

6. Achtung! Bei Bedarf kann die Hexadezimalzahl auch eine „Kommastelle“ bekommen. Das passiert, wenn man die Wertigkeit der einzelnen Hexadezimalziffern ganz einfach in den negativen Bereich ausdehnt. Allerdings entstehen da auf Grund der Basis 16 mit $16^{-1}$ und $16^{-2}$ ziemlich abstruse Dezimalzahlen:

| | | | | | |
|---|---|---|---|---|---|
| Hexadezimalzahl: | **Q** | **R,** | **S** | **T** | **H** |
| Ziffernnummer: | 1 | 0 | -1 | -2 | |

Wertigkeit: $Q \cdot 16^1 + R \cdot 16^0 + S \cdot 16^{-1} + T \cdot 16^{-2}$
$= Q \cdot 16 + R \cdot 1 + S \cdot 0{,}0625 + T \cdot 0{,}00390625$

Ein Beispiel für eine Hexadezimalzahl mit „Kommastelle“:

| | | | | | |
|---|---|---|---|---|---|
| Hexadezimalzahl: | **1** | **5,** | **A** | **3** | **H** |
| Ziffernnummer: | 1 | 0 | -1 | -2 | |

Wertigkeit: $1 \cdot 16^1 + 5 \cdot 16^0 + A \cdot 16^{-1} + 3 \cdot 16^{-2}$
$= 1 \cdot 16 + 5 \cdot 1 + 10 \cdot 0{,}0625 + 3 \cdot 0{,}00390625$
$= 21{,}63671875_{dezimal}$

**Umrechnung von Dezimalzahlen in Hexadezimalzahlen**

Variante 1: Ermittlung der zur Dezimalzahl gehörenden Binär- bzw. Dualzahl und deren Umwandlung in eine Hexadezimalzahl – einfache Methode

$$47_{dezimal} = 0010\ 1111_{binär} = 2FH$$
$$229_{dezimal} = 1110\ 0101_{binär} = E5H$$

Variante 2: Zerlegen der Dezimalzahl in eine Summe von Produkten aus Faktoren kleiner 16 und der Potenzen von 16 – aufwendig

<u>Beispiel 1:</u>

$47_{dezimal} : 16^1 = 2, \ldots$

$2 \cdot 16^1 = 32$

Rest: $47 - 32 = 15$

$15_{dezimal}: 16^0 = 15$

$15 \cdot 16^0 = 15$

Rest: $15 - 15 = 0$

$47_{dezimal} = 2 \cdot 16^1 + 15 \cdot 16^0 = 2FH$

Beispiel 2:

$229_{dezimal}: 16^1 = 14, \ldots$

$14 \cdot 16^1 = 224$

Rest: $229 - 224 = 5$

$5_{dezimal}: 16^0 = 5$

$5 \cdot 16^0 = 5$

Rest: $5 - 5 = 0$

$229_{dezimal} = 14 \cdot 16^1 + 5 \cdot 16^0 = E5H$

Variante 3: Wiederholte Division durch 16 mit Festhalten des Divisionsrestes

Beispiel 1:

| | | |
|---|---|---|
| $47: 16 = 2$ | Rest: 15 | Der Rest der ersten Division durch 16 entspricht der niederwertigsten Ziffer der entstehenden Hexadezimalzahl. |
| $2: 16 = 0$ | Rest: 2 | Der Rest der letzten Division durch 16 entspricht der höchstwertigsten Ziffer der entstehenden Hexadezimalzahl. |

$47_{dezimal} = 2 \cdot 16^1 + 15 \cdot 16^0 = 2FH$

Beispiel 2:

| | | |
|---|---|---|
| $229: 16 = 14$ | Rest: 5 | Der Rest der ersten Division durch 16 entspricht der niederwertigsten Ziffer der entstehenden Hexadezimalzahl. |
| $14: 16 = 0$ | Rest: 14 | Der Rest der letzten Division durch 16 entspricht der höchstwertigsten Ziffer der entstehenden Hexadezimalzahl. |

$229_{dezimal} = 14 \cdot 16^1 + 5 \cdot 16^0 = E5H$

**Zahlenbereiche von Hexadezimalzahlen ohne Vorzeichen**

- 1 Hexadezimalziffer = 1 Tetrade = 4 Bit
  größte positive Hexadezimalzahl: $FH = 15_{dezimal}$

kleinste positive Hexadezimalzahl: $0H = 0_{dezimal}$

- 2 Hexadezimalziffern = 2 Tetraden = 8 Bit
  größte positive Hexadezimalzahl: $FFH = 255_{dezimal}$
  kleinste positive Hexadezimalzahl: $00H = 0_{dezimal}$

- 4 Hexadezimalziffern = 4 Tetraden = 16 Bit
  größte positive Hexadezimalzahl: $FFFFH = 65535_{dezimal}$
  kleinste positive Hexadeziamlzahl: $0000H = 0_{dezimal}$

- …

**Zahlenbereiche von Hexadezimalzahlen mit Vorzeichen**

Für Hexadezimalzahlen gibt es zwei verschiedene Betrachtungsweisen:

1. Die Hexadezimalzahlen sind positive Zahlen ohne Vorzeichen – so, wie die Zahlen bis jetzt vorgestellt wurden:

   $2FH = 47_{dezimal}$
   $E5H = 229_{dezimal}$

2. Oder aber die Hexadezimalzahlen werden als vorzeichenbehaftete Hexadezimalzahlen betrachtet. Dabei müssen wir auf die Binär- bzw. Dualzahlen zurückgreifen: Denn das höchstwertige Bit (MSB = most significant bit) der ersten Tetrade der Hexadezimalzahl wird auch hier zum **Vorzeichenbit**. 0 heißt positives Vorzeichen, 1 heißt negatives Vorzeichen:

   $$47_{dezimal} = 2FH = \mathbf{0}010\ 1111_{binär} > 0$$
   $$229_{dezimal} = E5H = \mathbf{1}110\ 0101_{binär} < 0$$

Welche Betrachtungsweise benutzt wird, hängt von der jeweiligen Problemstellung ab. Treten nur positive Zahlenwerte auf, kann auf die Vorzeicheninformation verzichtet werden. Sind negative Zahlen nicht ausschließbar, muss mit vorzeichenbehafteten Hexadezimalzahlen gearbeitet werden. In der Prozessor-Programmierung hat man die Wahl zwischen Befehlen, die Zahlenwerte immer als positive Größe auffassen, und anderen Befehlen, die Zahlenwerte vorzeichenrichtig verarbeiten.

**Problem:** Der Wert einer negativen Hexadezimalzahl lässt sich nicht wie bei positiven Hexadezimalzahlen ermitteln. Mit Hilfe der **Zweierkomplementbildung** wird eine negative Hexadezimalzahl in die zugehörige positive Hexadezimalzahl umgewandelt (d. h. das Vorzeichen wird gewechselt). Von der entstandenen positiven Hexadezimalzahl kann man dann wie bisher den Wert errechnen. Damit ist bekannt, welchen Wert die negative Hexadezimalzahl hat.

**Schritte bei der Bildung des Zweierkomplements sind:**

1. Zugehörige Binärzahl invertieren = negieren, d. h. aus 0 wird 1 und aus 1 wird 0
2. Addition von 1 auf die letzte Stelle der negierten Binärzahl

Ausgangszahl: $E5H = \mathbf{1}1100101_{binär} =???$

Frage: Welchen (Dezimal-)Wert hat die negative Hexadezimalzahl $E5H$?

Negation: $00011010_{binär}$
Addition von 1: $\mathbf{0}0011011_{binär} = 1BH$

Antwort: Die negative Hexadezimalzahl $E5H$ hat den (Dezimal-)Wert $-1BH = -27_{dezimal}$.

Ausgangszahl: $2FH = \mathbf{0}0101111_{binär} = +47_{dezimal}$

Frage: Welche Hexadezimalzahl gehört zur (Dezimal-)Zahl $-47_{dezimal}$?

Negation: $11010000_{binär}$
Addition von 1: $\mathbf{1}1010001_{binär} = -47_{dezimal}$

Antwort: Die zu $-47_{dezimal}$ gehörende Hexadezimalzahl lautet $\mathbf{1}1010001_{binär} = D1H.$

Es ergibt sich folgendes Bild für vorzeichenbehaftete Hexadezimalzahlen (beispielhaft für 8 Bit dargestellt):

$$
\begin{aligned}
+127_{dezimal} &= 7FH = 0111\ 1111_{binär} \\
&\vdots \\
+4_{dezimal} &= 04H = 0000\ 0100_{binär} \\
+3_{dezimal} &= 03H = 0000\ 0011_{binär} \\
+2_{dezimal} &= 02H = 0000\ 0010_{binär} \\
+1_{dezimal} &= 01H = 0000\ 0001_{binär} \\
0_{dezimal} &= 00H = 0000\ 0000_{binär} \\
-1_{dezimal} &= FFH = 1111\ 1111_{binär} \\
-2_{dezimal} &= FEH = 1111\ 1110_{binär} \\
-3_{dezimal} &= FDH = 1111\ 1101_{binär} \\
-4_{dezimal} &= FCH = 1111\ 1100_{binär} \\
&\vdots \\
-128_{dezimal} &= 80H = 1000\ 0000_{binär}
\end{aligned}
$$

Was sagt uns das? Ganz einfach: **Positive** Hexadezimalzahlen kann man bei Bedarf beliebig mit Nullen auf 8 oder 16 oder 32 oder X Bit auffüllen. Am Wert der aufgefüllten Zahl ändert sich dabei nichts.

Handelt es sich um eine **negative** Hexadezimalzahl – sprich das erste Bit = Vorzeichenbit ist gleich 1, dann muss man mit Einsen auffüllen, damit das Vorzeichenbit auch nach dem Auffüllen noch gleich 1 ist. Am Wert der negativen aufgefüllten Zahl ändert sich dabei auch nichts. Klingt komisch, funktioniert aber so.

Gut nachvollziehen kann man das an $-1_{dezimal} = FFH = 1111\ 1111_{binär}$. Ob man da 8 oder 16 oder 32 oder X binäre Einsen dastehen hat – sie entsprechen immer dem Wert $-1_{dezimal}$.

- 4 Bit
  größte positive Hexadezimalzahl: $\mathbf{0}111_{binär} = 7H = +7_{dezimal}$
  kleinste negative Hexadezimalzahl: $\mathbf{1}000_{binär} = 8H = -8_{dezimal}$

- 8 Bit
  größte positive Hexadezimalzahl: $\mathbf{0}111\ 1111_{binär} = 7FH = +127_{dezimal}$
  kleinste negative Hexadezimalzahl: $\mathbf{1}000\ 0000_{binär} = 80H = -128_{dezimal}$

- 16 Bit
  größte positive Hexadezimalzahl: $\mathbf{0}111\ 1111\ 1111\ 1111_{binär} = 7FFFH = +32767_{dezimal}$
  kleinste negative Hexadezimalzahl: $\mathbf{1}000\ 0000\ 0000\ 0000_{binär} = 8000H = -32768_{dezimal}$

- …

**Rechnen mit Hexadezimalzahlen**

Wer das aus welchen Gründen auch immer unbedingt sehen möchte – ich habe es mal beispielhaft vorgenommen.

Addition ohne Vorzeichen (das Rechenprinzip entspricht der schriftlichen Addition von Dezimalzahlen):

```
Binärzahlen          Hexadezimalzahlen      Dezimalzahlen

  0110 1010 =          6AH =                  106
+ 0011 0101 =        + 35H =                 + 53
  1001 1111 =          9FH =                  159

  0111 0011 =          73H =                  115
+ 0001 1111 =        + 1FH =                 + 31
  1001 0010 =          92H =                  146


                       3A4FH =              14927
                     + B192H =            + 45458
                       EBE1H =              60385

                       4321H =              17185
                     + FFFFH =            + 65535
                      14320H =              82720
```

Addition mit Vorzeichen (das Rechenprinzip entspricht der schriftlichen Addition von Dezimalzahlen):

```
Binärzahlen          Hexadezimalzahlen      Dezimalzahlen

  0110 1010 =          69H =                  106
+ 1010 0110 =        + A6H =              + (-90)
  0001 0000 =          10H =                   16

  0011 0110 =          36H =                   54
+ 1000 1100 =        + 8CH =              +(-116)
  1100 0010 =          C2H =                  -62
```

Subtraktion ohne Vorzeichen (das Rechenprinzip entspricht der schriftlichen Subtraktion von Dezimalzahlen):

```
Binärzahlen          Hexadezimalzahlen      Dezimalzahlen

  0110 1010 =          69H =                  106
- 0011 0101 =        - 35H =                 - 53
  0011 0101 =          35H =                   53

  0111 0011 =          73H =                  115
- 0001 1111 =        - 1FH =                 - 31
  0101 0100 =          54H =                   84


                       FFFFH =              65535
                     - 1234H =            -  4660
                       EDCBH =              60875
```

Subtraktion mit Vorzeichen (das Rechenprinzip entspricht der schriftlichen Subtraktion von Dezimalzahlen):

```
Binärzahlen            Hexadezimalzahlen       Dezimalzahlen

  0001 1010 =              19H =                     26
- 1010 0110 =            - A6H =                - (-90)
-------------------------------------------------------
  0111 0100 =              74H =                    116

  1001 1011 =              9BH =                 (-101)
- 1000 1100 =            - 8CH =                -(-116)
-------------------------------------------------------
  0000 1111 =              0FH =                     15
```

Multiplikation (das Rechenprinzip entspricht der schriftlichen Multiplikation von Dezimalzahlen):

Dezimalzahlen

```
242 * 21
--------
    484
+    242
--------
    5082
```

Hexadezimalzahlen

```
F2H * 15H
---------
     F2H
+    4BAH
---------
    13DAH
```

Division mit Kommastelle (das Rechenprinzip entspricht der schriftlichen Division von Dezimalzahlen):

Dezimalzahlen

```
 242 : 44  = 5,5
-220
----
  220
 -220
 ----
    0
```

Hexadezimalzahlen

F2H : 2CH = 5,8H = $5 * 16^0 + 8 * 16^{-1}$ = 5,5

```
 F2H : 2CH
-DCH
----
 160H
-160H
-----
   0H
```