

Rechenaufwand der DFT bzw. IDFT

Ein Nachteil der direkten Berechnung der (im Allgemeinen komplexen) DFT-Zahlenfolgen

$$\langle D(\mu) \rangle \bullet \text{---}(N) \text{---} \circ \langle d(\nu) \rangle$$

gemäß den in **Kapitel 5.2** angegebenen Gleichungen ist der große Rechenaufwand. Beispielsweise kann dieser Aufwand für die Diskrete Fouriertransformation (DFT)

$$\begin{aligned} N \cdot D(\mu) &= \sum_{\nu=0}^{N-1} d(\nu) \cdot w^{\nu \cdot \mu} = \\ &= d(0) \cdot w^0 + d(1) \cdot w^{\mu} + d(2) \cdot w^{2\mu} + \dots + d(N-1) \cdot w^{(N-1) \cdot \mu} \end{aligned}$$

wie folgt abgeschätzt werden:

- Wir gehen davon aus, dass die Potenzen des komplexen Drehfaktors $w = \exp(-j2\pi/N)$ bereits in Real- und Imaginärteilform in einer Lookup-Tabelle vorliegen.
- Zur Berechnung eines einzelnen Koeffizienten benötigt man $N - 1$ komplexe Multiplikationen und ebenso viele komplexe Additionen.
- Jede komplexe Addition erfordert zwei reelle Additionen:

$$(R_1 + j \cdot I_1) + (R_2 + j \cdot I_2) = (R_1 + R_2) + j \cdot (I_1 + I_2).$$

- Jede komplexe Multiplikation erfordert vier reelle Multiplikationen und zwei reelle Additionen (eine Subtraktion wird wie eine Addition behandelt):

$$(R_1 + j \cdot I_1)(R_2 + j \cdot I_2) = (R_1 \cdot R_2 - I_1 \cdot I_2) + j \cdot (R_1 \cdot I_2 + R_2 \cdot I_1).$$

- Somit sind zur Berechnung aller N Koeffizienten die folgende Anzahl M reeller Multiplikationen und die Anzahl A reeller Additionen erforderlich:

$$M = 4 \cdot N \cdot (N - 1), \quad A = 2 \cdot N \cdot (N - 1) + 2 \cdot N \cdot (N - 1) = M.$$

- In heutigen Rechnern benötigen Multiplikationen und Additionen/Subtraktionen etwa die gleiche Rechenzeit und es genügt, die Gesamtanzahl $O = M + A$ aller Rechenoperationen zu betrachten:

$$O = 8 \cdot N \cdot (N - 1) \approx 8 \cdot N^2.$$

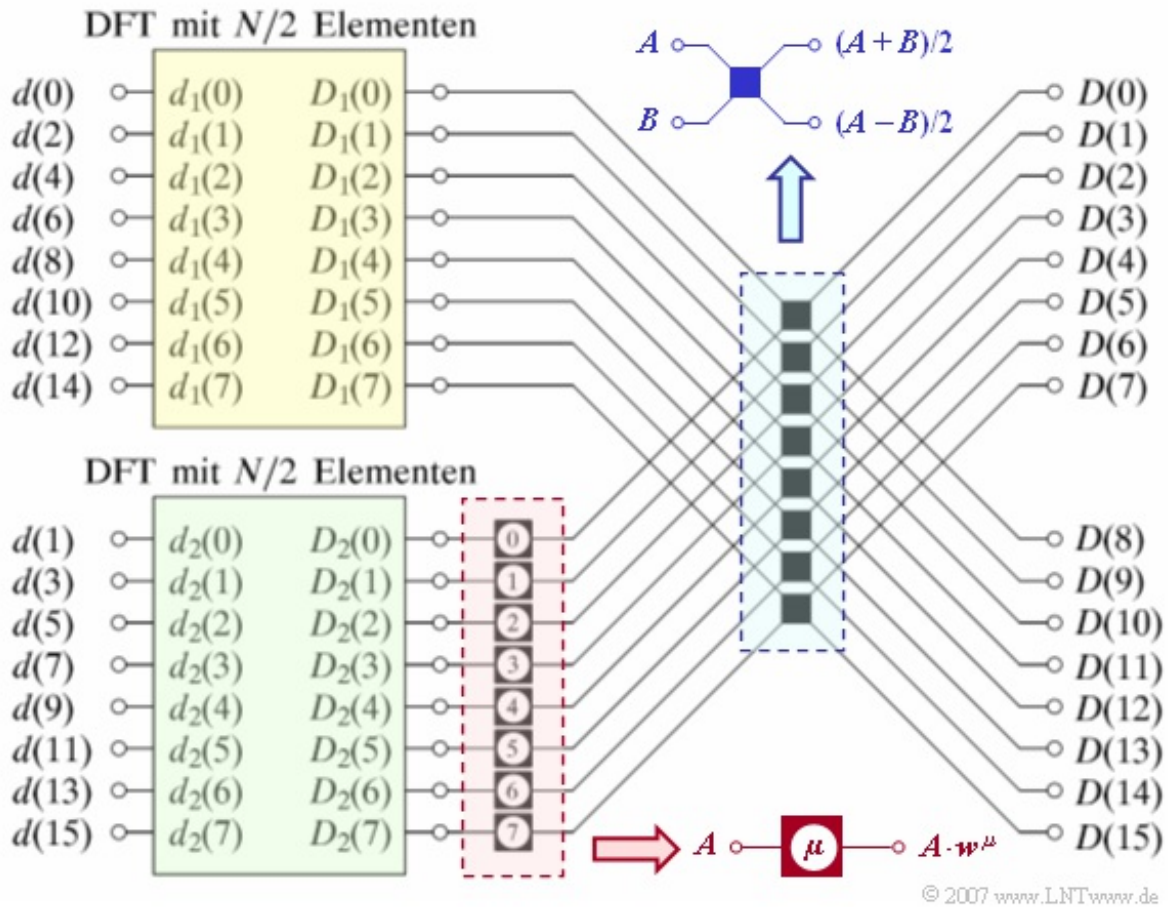
- Daraus folgt: Man benötigt bereits für eine DFT (oder eine IDFT) mit $N = 1000$ knapp 8 Millionen Rechenoperationen. Für $N = 16$ sind immer noch 1920 Rechenoperationen erforderlich.

Ist der Parameter N eine Potenz zur Basis 2, so können rechenzeitgünstigere Algorithmen angewandt werden. Die Vielzahl solcher aus der Literatur bekannten Verfahren werden unter dem Sammelbegriff **Fast-Fouriertransformation** – abgekürzt FFT – zusammengefasst. Alle diese Methoden basieren auf dem Überlagerungssatz der DFT.

Literatur: [Ach95] – [CT65] – [Duh86] – [Heu82] – [Mar94] – [Mil00] – [Pap62] – [Söd00] – [VHH98]

Überlagerungssatz der DFT (1)

Die folgende Grafik verdeutlicht den so genannten **Überlagerungssatz** der DFT am Beispiel $N = 16$.



Der dadurch beschriebene Algorithmus ist durch folgende Schritte gekennzeichnet:

- Die Eingangsfolge $\langle d(v) \rangle$ der Länge N wird in zwei Teilfolgen $\langle d_1(v) \rangle$ und $\langle d_2(v) \rangle$ jeweils halber Länge separiert (gelb bzw. grün hinterlegt). Mit $0 \leq v < N/2$ ergeben sich die Folgeelemente $d_1(v) = d(2v)$, $d_2(v) = d(2v + 1)$.

- Die Ausgangsfolgen $\langle D_1(\mu) \rangle$ und $\langle D_2(\mu) \rangle$ der beiden Teilblöcke ergeben sich daraus jeweils durch eine eigene DFT, aber nun nur noch mit halber Länge $N/2 = 8$:

$$\langle D_1(\mu) \rangle \bullet \text{---} (N/2) \text{---} \circ \langle d_1(v) \rangle, \quad \langle D_2(\mu) \rangle \bullet \text{---} (N/2) \text{---} \circ \langle d_2(v) \rangle.$$

- Die Ausgangswerte $D_2(\mu)$ der unteren DFT (mit $0 \leq \mu < N/2$) werden anschließend im rot umrandeten Block durch komplexe Drehfaktoren hinsichtlich ihrer Phasenlage verändert:

$$D_2(\mu) \Rightarrow D_2(\mu) \cdot w^\mu, \quad \text{wobei } w = e^{-j \cdot 2\pi/N}.$$

- Jeder einzelne **Butterfly** im blau umrandeten Block liefert durch Addition bzw. Subtraktion zwei Elemente der gesuchten Ausgangsfolge. Mit $0 \leq \mu < N/2$ gilt dabei:

$$D(\mu) = \frac{1}{2} \cdot [D_1(\mu) + D_2(\mu) \cdot w^\mu],$$

$$D\left(\mu + \frac{N}{2}\right) = \frac{1}{2} \cdot [D_1(\mu) - D_2(\mu) \cdot w^\mu].$$

Durch diese erste Anwendung des Überlagerungssatzes reduziert sich der Rechenaufwand etwa um die Hälfte. Man benötigt nun anstelle von $O = 1920$ reellen Rechenoperationen nur mehr

$$O = 2 \cdot 448 + 8 \cdot (4 + 2) + 16 \cdot 2 = 976.$$

Der erste Summand berücksichtigt die Rechenoperationen der beiden DFT-Berechnungen mit $N/2 = 8$, der zweite die 8 komplexen Multiplikationen des Drehfaktors (roter Block) und der letzte Summand die 16 komplexen Additionen bzw. Subtraktionen des blau umrandeten Blocks.

Überlagerungssatz der DFT (2)

Beispiel: Die DFT-Koeffizienten $d(v)$ zur Beschreibung des Zeitverlaufs seien entsprechend der Zeile 2 der folgenden Tabelle „dreieckförmig“ belegt. Beachten Sie hierbei die periodische Fortsetzung der DFT, so dass der lineare Anstieg für $t < 0$ durch die Koeffizienten $d(9) \dots d(15)$ ausgedrückt wird.

Durch Anwendung des DFT-Algorithmus mit $N = 16$ erhält man die in der ebenfalls blau hinterlegten Zeile 3 angegebenen Spektralkoeffizienten $D(\mu)$, die bei Vernachlässigung des Aliasingfehlers gleich $D(\mu) = 4 \cdot \text{sinc}^2(\pi \cdot \mu/2)$ wären. Man findet, dass sich der Aliasingfehler nur auf die ungeradzahlgigen Koeffizienten auswirkt (schraffierte Felder). Beispielsweise müsste $D(1) = 16/\pi^2 \approx 1.621 \neq 1.642$ sein.

v bzw. μ	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
2: $d(v)$	8.0	7.0	6.0	5.0	4.0	3.0	2.0	1.0	0.0	1.0	2.0	3.0	4.0	5.0	6.0	7.0
3: $D(\mu)$	4.000	1.642	0.000	0.202	0.000	0.090	0.000	0.066	0.000	0.066	0.000	0.090	0.000	0.202	0.000	1.642
4: $d_1'(v)$	8.0	0.0	6.0	0.0	4.0	0.0	2.0	0.0	0.0	0.0	2.0	0.0	4.0	0.0	6.0	0.0
5: $D_1'(\mu)$	2.000	0.854	0.000	0.146	0.000	0.146	0.000	0.854	2.000	0.854	0.000	0.146	0.000	0.146	0.000	0.854
6: $d_2'(v)$	0.0	7.0	0.0	5.0	0.0	3.0	0.0	1.0	0.0	1.0	0.0	3.0	0.0	5.0	0.0	7.0
7: $D_2'(\mu)$	2.000	0.788	0.000	0.056	0.000	-0.056	0.000	-0.788	-2.000	-0.788	0.000	-0.056	0.000	0.056	0.000	0.788

© 2007 www.LNTwww.de

Spaltet man die Gesamtfolge $\langle d(v) \rangle$ in die beiden Teilfolgen $\langle d_1'(v) \rangle$ und $\langle d_2'(v) \rangle$ auf, und zwar derart, dass die erste (gelb hinterlegte) Teilfolge nur die geradzahlgigen Koeffizienten ($v = 0, 2, \dots, N - 2$) und die zweite (grün hinterlegt) nur die ungeradzahlgigen Koeffizienten ($v = 1, 3, \dots, N - 1$) beinhalten, während alle anderen zu 0 gesetzt sind, so erhält man die zugehörigen Folgen im Spektralbereich:

$$\langle D_1'(\mu) \rangle \bullet \text{---} (N) \text{---} \circ \langle d_1'(v) \rangle, \quad \langle D_2'(\mu) \rangle \bullet \text{---} (N) \text{---} \circ \langle d_2'(v) \rangle.$$

In den gelb bzw. grün hinterlegten Zeilen 4/7 erkennt man:

- Wegen $d(v) = d_1'(v) + d_2'(v)$ gilt auch $D(\mu) = D_1'(\mu) + D_2'(\mu)$. Dies lässt sich zum Beispiel mit dem Additionstheorem linearer Systeme begründen.
- Die Periode von $\langle D_1'(\mu) \rangle$ beträgt aufgrund des Nullsetzens eines jeden zweiten Zeitkoeffizienten nun $N/2$ im Gegensatz zur Periode N der Ursprungsfolge $\langle D(\mu) \rangle$:

$$D_1'(\mu + \frac{N}{2}) = D_1'(\mu).$$

- Die zweite Folge, gegenüber $D_1'(\mu)$ um einen Abtastwert verschoben, ist mit einem Phasenfaktor belegt, der einen Vorzeichenwechsel zweier um $N/2$ auseinanderliegender Koeffizienten bewirkt:

$$D_2'(\mu + \frac{N}{2}) = -D_2'(\mu).$$

- Die Berechnung von $\langle D_1'(\mu) \rangle$ und $\langle D_2'(\mu) \rangle$ ist jeweils ebenso aufwändig wie die Bestimmung von

$\langle D(\mu) \rangle$, da $\langle d_1'(v) \rangle$ und $\langle d_2'(v) \rangle$ ebenfalls aus N Elementen bestehen, auch wenn einige 0 sind.

Überlagerungssatz der DFT (3)

Beispiel (Fortsetzung):

v bzw. μ	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
2: $d(v)$	8.0	7.0	6.0	5.0	4.0	3.0	2.0	1.0	0.0	1.0	2.0	3.0	4.0	5.0	6.0	7.0
3: $D(\mu)$	4.000	1.642	0.000	0.202	0.000	0.090	0.000	0.066	0.000	0.066	0.000	0.090	0.000	0.202	0.000	1.642
4: $d_1'(v)$	8.0	0.0	6.0	0.0	4.0	0.0	2.0	0.0	0.0	0.0	2.0	0.0	4.0	0.0	6.0	0.0
5: $D_1'(\mu)$	2.000	0.854	0.000	0.146	0.000	0.146	0.000	0.854	2.000	0.854	0.000	0.146	0.000	0.146	0.000	0.854
6: $d_2'(v)$	0.0	7.0	0.0	5.0	0.0	3.0	0.0	1.0	0.0	1.0	0.0	3.0	0.0	5.0	0.0	7.0
7: $D_2'(\mu)$	2.000	0.788	0.000	0.056	0.000	-0.056	0.000	-0.788	-2.000	-0.788	0.000	-0.056	0.000	0.056	0.000	0.788
v bzw. μ	0	1	2	3	4	5	6	7								
9: $d_1(v)$	8.0	6.0	4.0	2.0	0.0	2.0	4.0	6.0								
10: $D_1(\mu)$	4.000	1.708	0.000	0.292	0.000	0.292	0.000	1.708								
11: $d_2(v)$	7.0	5.0	3.0	1.0	1.0	3.0	5.0	7.0								
12: $D_2(\mu)$	4.000	1.456 +j · 0.603	0.000	0.043 +j · 0.103	0.000	0.043 -j · 0.103	0.000	1.456 -j · 0.603								

Verzichtet man auf die Koeffizienten $d_1'(v) = 0$ mit ungeraden sowie auf $d_2'(v) = 0$ mit geraden Indizes, so kommt man zu den in den Zeilen 9 und 11 angegebenen Teilfolgen $\langle d_1(v) \rangle$ und $\langle d_2(v) \rangle$. Man erkennt:

- Die beiden Zeitfolgen $\langle d_1(v) \rangle$ und $\langle d_2(v) \rangle$ weisen ebenso wie die dazugehörigen Spektralfolgen $\langle D_1(\mu) \rangle$ und $\langle D_2(\mu) \rangle$ nur noch die Dimension $N/2$ auf.
- Ein Vergleich der Zeilen 5, 7, 10 und 12 zeigt für $0 \leq \mu < N/2$ folgenden Zusammenhang:

$$D_1'(\mu) = \frac{1}{2} \cdot D_1(\mu),$$

$$D_2'(\mu) = \frac{1}{2} \cdot D_2(\mu) \cdot w^\mu.$$

- Entsprechend erhält man für $N/2 \leq \mu < N$:

$$D_1'(\mu) = \frac{1}{2} \cdot D_1\left(\mu - \frac{N}{2}\right),$$

$$D_2'(\mu) = \frac{1}{2} \cdot D_2\left(\mu - \frac{N}{2}\right) \cdot w^\mu = -\frac{1}{2} \cdot D_2\left(\mu - \frac{N}{2}\right) \cdot w^{\mu - N/2}.$$

- Beispielsweise erhält man für $N = 16$ – das heißt $w = \exp(-j \cdot \pi/8)$ – sowie $\mu = 1$ bzw. $\mu = 9$:

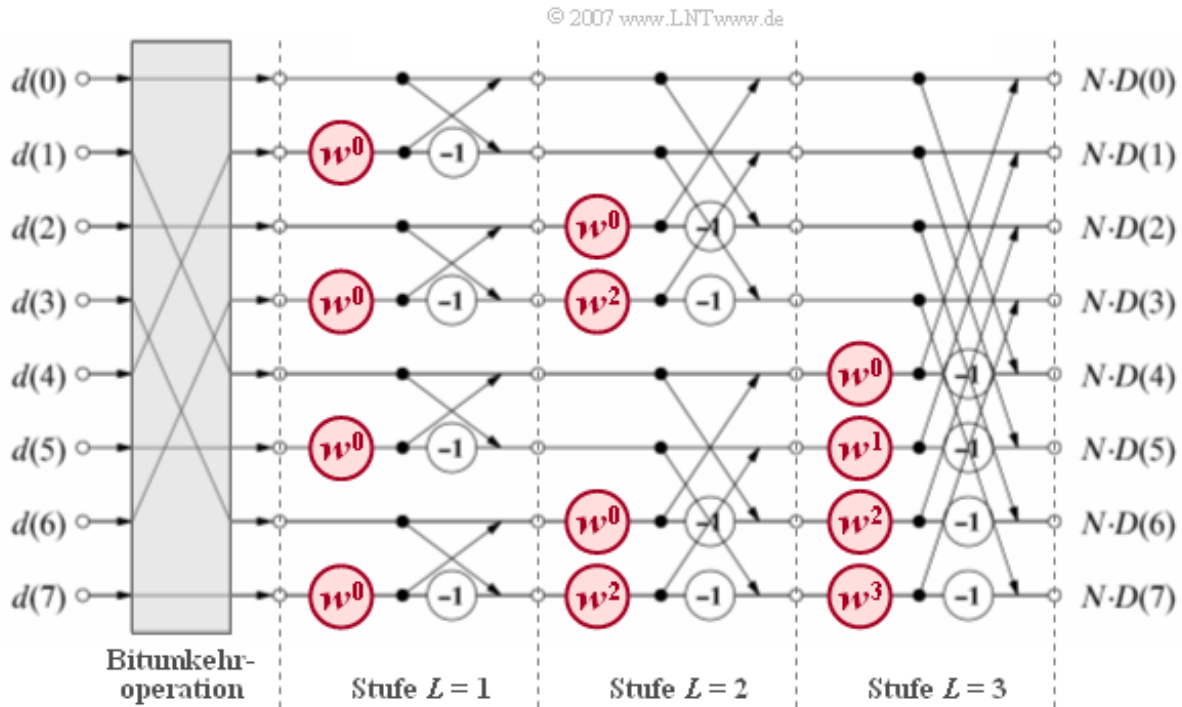
$$D_1'(1) = \frac{1.708}{2} = 0.854, \quad D_2'(1) = \frac{1}{2} \cdot (1.456 + j0.603) \cdot e^{-j \cdot \pi/8} = 0.788$$

$$\Rightarrow D(1) = D_1'(1) + D_2'(1) = 1.642.$$

$$D_9'(1) = \frac{1.708}{2} = 0.854, \quad D_2'(9) = -\frac{1}{2} \cdot (1.456 + j0.603) \cdot e^{-j \cdot \pi/8} = -0.788$$
$$\Rightarrow D(9) = D_1'(9) + D_2'(9) = 0.066.$$

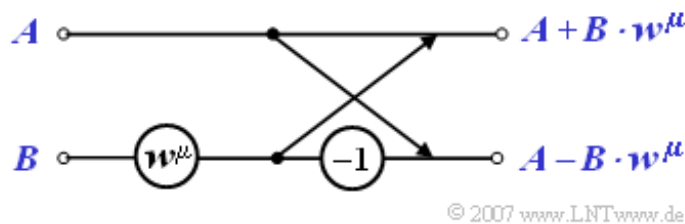
Radix-2-Algorithmus nach Cooley und Tukey (1)

Ebenso wie andere FFT- Algorithmen baut das hier vorgestellte Verfahren auf dem Überlagerungssatz der DFT auf und funktioniert nur dann, wenn die Stützstellenzahl N eine Zweierpotenz ist. Das nachfolgende Bild verdeutlicht die von Cooley und Tukey entwickelte Methode – siehe auch [CT65] – für das Beispiel $N = 8$, wobei die Transformation vom Zeit- in den Frequenzbereich dargestellt ist.



Man erkennt aus dieser Darstellung:

- Vor dem eigentlichen FFT-Algorithmus müssen die komplexen Eingangswerte $d(0), \dots, d(N-1)$ umsortiert werden. Dies geschieht im grau hinterlegten Block **Bitumkehroperation**, der auf der nächsten Seite im Detail beschrieben wird.
- Die Berechnung erfolgt in $\log_2 N = 3$ Stufen, wobei in jeder Stufe genau $N/2 = 4$ prinzipiell gleiche Berechnungen mit unterschiedlichem Parameter μ (= Exponent des komplexen Drehfaktors) ausgeführt werden. Eine solche Basisoperation bezeichnet man auch als **Butterfly**.
- Jeder Butterfly berechnet aus zwei Eingangsgrößen A und B , die im Allgemeinen komplex sind, die beiden Ausgangsgrößen $A + B \cdot w^\mu$ sowie $A - B \cdot w^\mu$ entsprechend der nachfolgenden Skizze.



- Die komplexen Spektralkoeffizienten $D(0), \dots, D(N-1)$ erhält man am Ausgang der letzten Stufe nach Division durch N . Wie in Aufgabe Z5.4 noch gezeigt wird, ergibt sich gegenüber der DFT eine deutlich kürzere Rechenzeit – zum Beispiel für $N = 1024$ um mehr als den Faktor 100.
- Die **inverse DFT** zur Berechnung der Zeitkoeffizienten aus den Spektralkoeffizienten lässt sich mit dem gleichen Algorithmus und nur geringfügigen Modifizierungen bewerkstelligen.

Radix-2-Algorithmus nach Cooley und Tukey (2)

<pre> 1: # include <math.h> 2: # define pi 3.141592654 3: void fft(N, Re, Im) 4: long N; float Re[], Im[]; 5: { void bitumkehr(), stufe(); 6: long i, ldN, L; 7: bitumkehr(N, Re, Im); 8: ldN=(log((double)N)/log(2.) + .5); 9: for (L=1; L<=ldN; L++) 10: stufe(L, N, Re, Im); 11: for (i=0; i<N; i++) 12: { Re[i] /= (N); Im[i] /= (N); } 13: } 14: 15: void stufe(L, N, Re, Im) 16: long L, N; float Re[], Im[]; 17: { long i, k, m, mu; 18: float arg, ReW, ImW, ReX, ImX; 19: i = pow(2., L-1); 20: for (mu=0; mu<i; mu++) 21: { arg = pi*(double)mu/(double)i; 22: ReW = cos(arg); 23: ImW = -sin(arg); 24: for (k=mu; k<N; k+=2*i) 25: { m = k+i; 26: ReX = ReW*Re[m] - ImW*Im[m]; 27: ImX = ReW*Im[m] + ImW*Re[m]; 28: Re[m] = Re[k] - ReX; 29: Im[m] = Im[k] - ImX; 30: Re[k] += ReX; 31: Im[k] += ImX; 32: } 33: } 34: }</pre>	<p>Mathematische Bibliothek</p> <p>Definition der Zahl pi</p> <p>Unterprogramm FFT</p> <p>Übergabeparameter</p> <p>Verwendete Unterprogramme</p> <p>Interne Variable</p> <p>Bitumkehroperation</p> <p>Logarithmus dualis von N</p> <p>Insgesamt ld N mal:</p> <p>Berechnungen für eine Stufe</p> <p>Für alle 2N Koeffizienten:</p> <p>Division durch N</p> <p>Berechnungen der L-ten Stufe</p> <p>Übergabeparameter von „FFT“</p> <p>Interne Long-Variable</p> <p>Interne Float-Variable</p> <p>$i = 2^{L-1}$ Phasenfaktoren</p> <p>Äußere Schleife – i mal durchlaufen</p> <p>Argument des Drehfaktors w</p> <p>Realteil von w</p> <p>Imaginärteil von w</p> <p>Innere Schleife – $N/(2i)$ mal durchlaufen</p> <p>Verknüpfung von Index k und m</p> <p>Hilfsgrößen aus [m] und w</p> <p>Differenz ins Element [m]</p> <p>Summe ins Element [k]</p>
--	--

© 2007 www.LNTwww.de

Sie sehen hier das C-Programm **fft(N, Re, Im)** gemäß dem oben beschriebenen Radix-2-Algorithmus. Anzumerken ist:

- Beim Aufruf beinhalten die beiden Float-Arrays „Re“ und „Im“ die N Real- und Imaginärteile der Zeitkoeffizienten $d(0), \dots, d(N-1)$.
- Mit den gleichen Feldern „Re“ und „Im“ werden am Programmende die N Real- und Imaginärteile der Spektralkoeffizienten $D(0), \dots, D(N-1)$ an das aufrufende Programm zurückgegeben.
- Aufgrund dieser „In-Place“-Programmierung reichen für diesen FFT-Algorithmus N komplexe Speicherplätze aus, allerdings nur dann, wenn zu Beginn die Eingangswerte umsortiert werden.
- Dies geschieht durch das (nicht explizit angegebene) Unterprogramm „bitumkehr“, wobei die Inhalte von $\text{Re}(v)$ und $\text{Im}(v)$ in die Feldelemente $\text{Re}(\kappa)$ und $\text{Im}(\kappa)$ eingetragen werden.
- Der Wert für den Feldindex κ ergibt sich, indem man v als Dualzahl darstellt und die $\text{ld } N$ Bits in umgekehrter Reihenfolge geschrieben werden. Wie aus nachfolgender Tabelle für $N = 8$ hervorgeht, ist die Dezimaldarstellung dieser Dualzahl der gesuchte κ -Wert.

ν	0	1	2	3	4	5	6	7
ν als Dualzahl	000	001	010	011	100	101	110	111
Bitumkehrung	000	100	010	110	001	101	001	111
κ	0	4	2	6	1	5	3	7

© 2007 www.LNTwww.de