In[1]:=
```
dct[k_] :=
 Table[Sqrt[2 / k] * Cos[(j + 1 / 2) * i * π / k] * If[i == 0, Sqrt[1 / 2], 1], {i, 0, k - 1}, {j, 0, k - 1}]
```

In[2]:=
```
dst[k_] := Sqrt[2 / k] *
  IdentityMatrix[k].Table[Sin[π / k * (j + 1 / 2) * (i + 1 / 2)], {i, 0, k - 1}, {j, 0, k - 1}]
```

```
(* Orthonormal inverse 4-point Type-II DCT with scale factors moved up front. *)
```

In[3]:=
```
idct4[{x0_, x2_, x1_, x3_}] := Module[{t0, t2, t1, t3, u0, u1, u2, u3},
  t0 = x0 / 2;
  t2 = x2 * Cos[π / 8] / Sqrt[2];
  t1 = x1 * Cos[2 π / 8] / Sqrt[2];
  t3 = x3 * Cos[3 π / 8] / Sqrt[2];

  u0 = t0 + t1;
  u1 = t0 - t1;
  u3 = t2 + t3;
  u2 = (t2 - t3) * 2 * Cos[π / 4] - u3;

  t0 = u0 + u3;
  t3 = u0 - u3;
  t1 = u1 + u2;
  t2 = u1 - u2;

  {t0, t1, t2, t3}]
```

In[4]:=
```
Total[Total[Abs[N[IdentityMatrix[4] - idct4 /@ Transpose[dct[4]]]]]]
```

Out[4]=
$$2.77556 \times 10^{-16}$$

```
(* Orthonormal inverse 4-point Type-IV DST with scale factors moved up front. *)
```

In[5]:=
```
idst4[{x4_, x6_, x5_, x7_}] := Module[{t4, t6, t5, t7, t8, u4, u5, u6, u7},
  t4 = x4 * Cos[7 π / 16] / Sqrt[2];
  t5 = x5 * Cos[3 π / 16] / Sqrt[2];
  t6 = x6 * Cos[5 π / 16] / Sqrt[2];
  t7 = x7 * Cos[π / 16] / Sqrt[2];

  u6 = t6 + t5;
  u5 = t6 - t5;
  u4 = t7 + t4;
  u7 = t7 - t4;

  t4 = u4 + u6;
  t6 = (u4 - u6) * 2 * Cos[π / 4];

  t8 = (u7 + u5) * 2 * Cos[π / 8];
  t7 = t8 - 2 * (Cos[π / 8] - Cos[3 π / 8]) * u7;
  t5 = t8 - 2 * (Cos[π / 8] + Cos[3 π / 8]) * u5;

  u4 = t4;
  u5 = u4 - t5;
  u6 = u5 + t6;
  u7 = u6 - t7;

  {u4, u5, u6, u7}]
```

In[6]:=
```
Total[Total[Abs[N[IdentityMatrix[4] - idst4 /@ Transpose[dst[4]]]]]]
```

Out[6]=
$$1.9984 \times 10^{-15}$$

```
(* Orthonormal inverse 8-point Type-II DCT based on the Chen
  factorization [1] with scale factors moved up front.  This computes an n-
 point Type-II DCT by first computing an n/2-point Type-
 II DCT of the even indexed inputs and an n/2-point Type-IV DST of the odd indexed inputs,
and then combining them using a "butterfly" operation.

  [1] W.H. Chen, C. Smith, and S. Fralick,
"A Fast Computational Algorithm for the Discrete Cosine Transform",
IEEE Transactions on Communications, Vol. 25, No. 9, pp 1004-1009, Sept. 1977 *)
```

In[7]:=
```
idct8[{x0_, x4_, x2_, x6_, x1_, x5_, x3_, x7_}] :=
 Module[{t0, t4, t2, t6, t1, t5, t3, t7, u0, u1, u2, u3, u4, u5, u6, u7},
  t0 = x0 / Sqrt[2];
  t4 = x4 / Sqrt[2];
  t2 = x2 / Sqrt[2];
  t6 = x6 / Sqrt[2];
  t1 = x1 / Sqrt[2];
  t5 = x5 / Sqrt[2];
  t3 = x3 / Sqrt[2];
  t7 = x7 / Sqrt[2];

  {t0, t1, t2, t3} = idct4[{t0, t2, t1, t3}];
  {t7, t6, t5, t4} = idst4[{t7, t5, t6, t4}];

  (* Butterflies *)

  u0 = t0 + t7;
  u7 = t0 - t7;
  u6 = t1 + t6;
  u1 = t1 - t6;
  u2 = t2 + t5;
  u5 = t2 - t5;
  u4 = t3 + t4;
  u3 = t3 - t4;

  {u0, u1, u2, u3, u4, u5, u6, u7}]
```

In[8]:=
```
Total[Total[Abs[N[IdentityMatrix[8] - idct8 /@ Transpose[dct[8]]]]]]
```

Out[8]=
$$3.96905 \times 10^{-15}$$

```
(* Orthonormal inverse 8-point Type-
 II DCT based on the AAN factorization [2].  Excluding initial scale factors,
this implementation computes the scaled inverse 8-
 point Type-II DCT with only 29 adds and 5 multiplies.

   [2] Y. Arai, T. Agui, and M. Nakajima, "A Fast DCT-SQ Scheme For Images",
IEICE Transactions, Vol. E-71, No. 11, pp 1095-1097, Nov. 1988 *)
```

In[9]:=
```
idct8fast[{x0_, x4_, x2_, x6_, x1_, x5_, x3_, x7_}] :=
 Module[{t0, t4, t2, t6, t1, t5, t3, t7, u0, u1, u2, u3, u4, u5, u6, u7, u8},
  t0 = x0 / Sqrt[8];
  u4 = x4 * Cos[π / 16] / 2;
  t2 = x2 * Cos[2 π / 16] / 2;
  u6 = x6 * Cos[3 π / 16] / 2;
  t1 = x1 * Cos[4 π / 16] / 2;
  u5 = x5 * Cos[5 π / 16] / 2;
  t3 = x3 * Cos[6 π / 16] / 2;
  u7 = x7 * Cos[7 π / 16] / 2;

  (* Embedded scaled inverse 4-point Type-II DCT using 9 adds and 1 multiply. *)

  u0 = t0 + t1;
  u1 = t0 - t1;
  u3 = t2 + t3;
  u2 = (t2 - t3) * 2 * Cos[π / 4] - u3;

  t0 = u0 + u3;
  t3 = u0 - u3;
  t1 = u1 + u2;
  t2 = u1 - u2;

  (* Embedded scaled inverse 4-point Type-IV DST using 12 adds and 4 multiplies. *)

  t5 = u5 + u6;
  t6 = u5 - u6;
  t7 = u4 + u7;
  t4 = u4 - u7;

  u7 = t7 + t5;
  u5 = (t7 - t5) * 2 * Cos[π / 4];

  u8 = (t4 + t6) * 2 * Cos[π / 8];
  u4 = u8 - 2 * (Cos[π / 8] - Cos[3 π / 8]) * t4;
  u6 = u8 - 2 * (Cos[π / 8] + Cos[3 π / 8]) * t6;

  t7 = u7;
  t6 = t7 - u6;
  t5 = t6 + u5;
  t4 = t5 - u4;

  (* Butterflies *)

  u0 = t0 + t7;
  u7 = t0 - t7;
  u6 = t1 + t6;
  u1 = t1 - t6;
  u2 = t2 + t5;
  u5 = t2 - t5;
  u4 = t3 + t4;
  u3 = t3 - t4;

  {u0, u1, u2, u3, u4, u5, u6, u7}]
```

In[10]:=
```
Total[Total[Abs[N[IdentityMatrix[8] - idct8fast /@ Transpose[dct[8]]]]]]
```

Out[10]=
$$3.96905 \times 10^{-15}$$