```
Z[K_] := Z[RVK@K];
Z[rvk_RVK] := (*Z[rvk] =*)
 Monitor[
  PP"Z"@Module[{todo, n, rots, ℰ, done, st, cx, ℰ1,
     i, j, k, k1, k2, k3},
    {todo, rots} = List @@ rvk;
    AppendTo[rots, 0];
    n = Length[todo];
    ℰ = 𝔼{}→{0}[0, 0, 1];
    done = {0};
    st = Range[0, 2 n + 1];
    While[{} =!= ($M = todo),
     {cx} = MaximalBy[todo,
       Length[done ⋂ {#[[1]], #[[2]], #[[1]] - 1, #[[2]] - 1}] &,
       1];
     {i, j} = List @@ cx;
     ℰ1 = Switch[Head[cx],
        Xp, (kRi,j kKinkk) // kmj,k→j,
        Xm, (kRi,j kKinkk) // kmj,k→j
       ];
     ℰ1 = (rot[k, rots[[i]]] ℰ1) // kmk,i→i; rots[[i]] = 0;
     ℰ1 = (ℰ1 rot[k, rots[[i + 1]]]) // kmi,k→i;
     rots[[i + 1]] = 0;
     ℰ1 = (rot[k, rots[[j]]] ℰ1) // kmk,j→j; rots[[j]] = 0;
     ℰ1 = (ℰ1 rot[k, rots[[j + 1]]]) // kmj,k→j;
     rots[[j + 1]] = 0;
     ℰ *= ℰ1;
     If[MemberQ[done, i], ℰ = ℰ // kmi,i+1→i;
      st = st /. st[[i + 2]] → st[[i + 1]]];
     If[MemberQ[done, i - 1], ℰ = ℰ // kmst[[i]],i→st[[i]];
      st = st /. st[[i + 1]] → st[[i]]];
     If[MemberQ[done, j], ℰ = ℰ // kmj,j+1→j;
      st = st /. st[[j + 2]] → st[[j + 1]]];
     If[MemberQ[done, j - 1], ℰ = ℰ // kmst[[j]],j→st[[j]];
      st = st /. st[[j + 1]] → st[[j]]];
     done = done ⋃ {i - 1, i, j - 1, j};
     todo = DeleteCases[todo, cx]
    ];
    Simplify /@ (ℰ /. {x0 → x, y0 → y, a0 → a})
   ], $M]
```