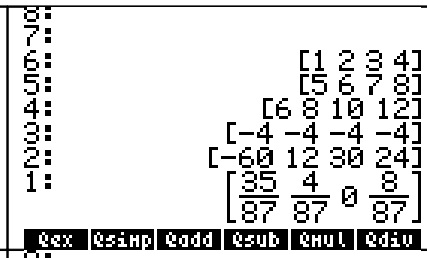
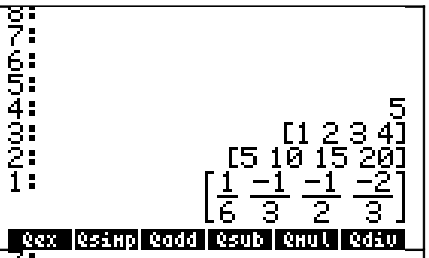
QUATERNION

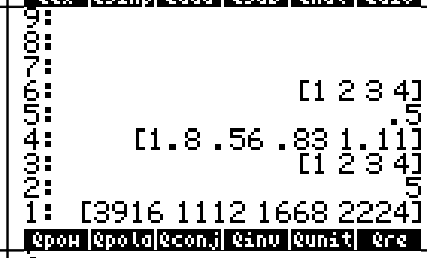
Qadd, Qsub, Qmul, Qdiv (0.5s)

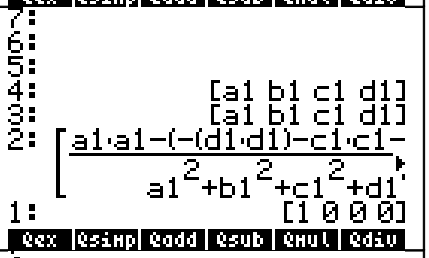




Qmul, Qdiv: (0.5s)

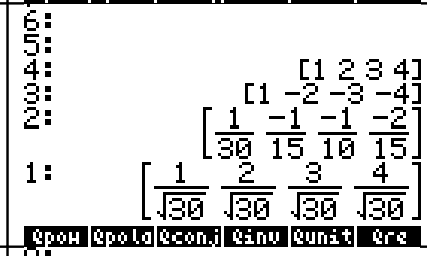
Qpow (3.7s)

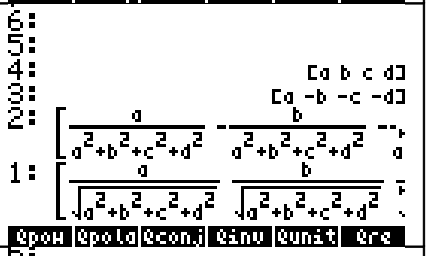




Qdiv: of symbolic quaternions
Qsimp: simplify expression

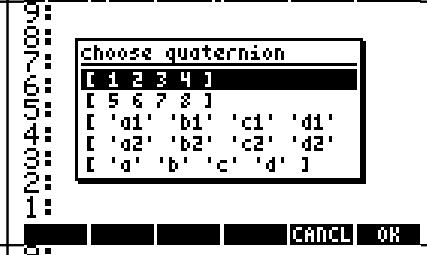
Qconj, Qinv, Qunit: conjugate, inverse, unit (0.1s)

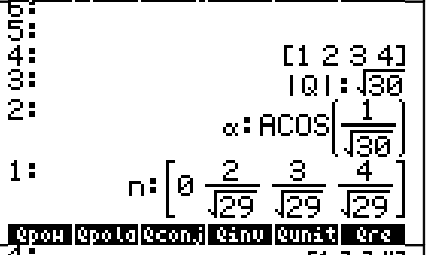




for symbolic quaternion (1s)

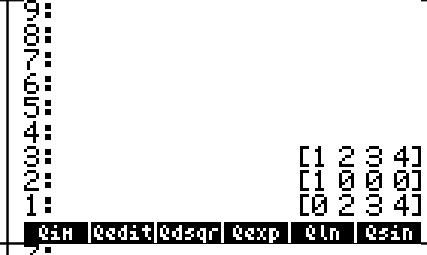
Qex: choose box with examples (0.1s)

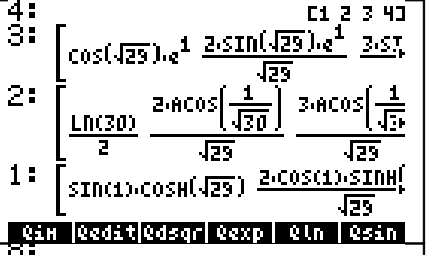




Qpolar: polar form (0.5s)

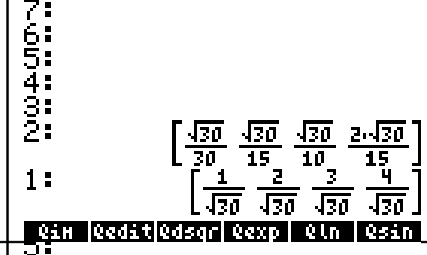
Qre, Qim: real, imaginary part (0.1s)

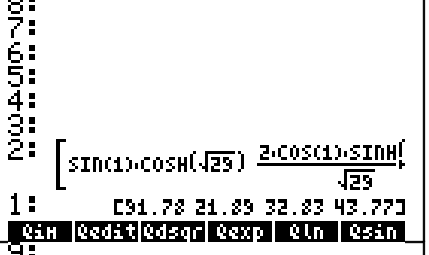




Qexp, Qln, Qsin (0.5s)

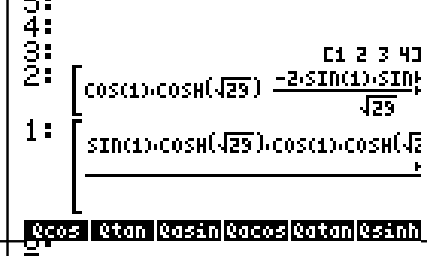
Qdsqrt: squareroot to denominator (3s)

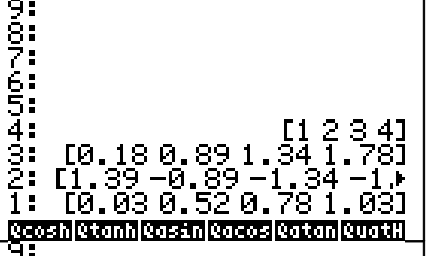




[->NUM] gives numerical value after 2 RND (0.5s)

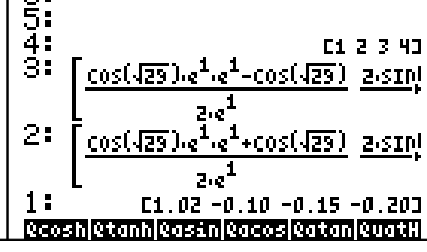
Qcos, Qtan (0.5s)

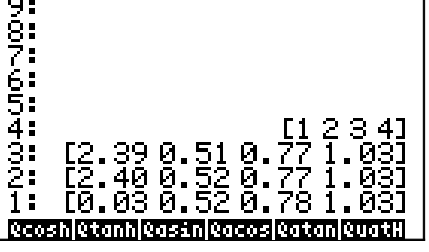




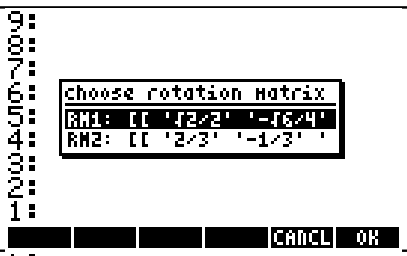
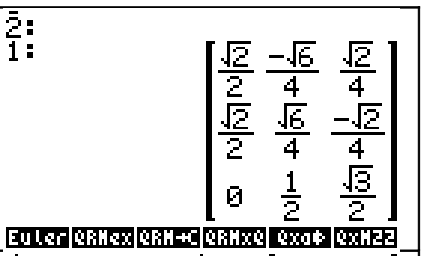
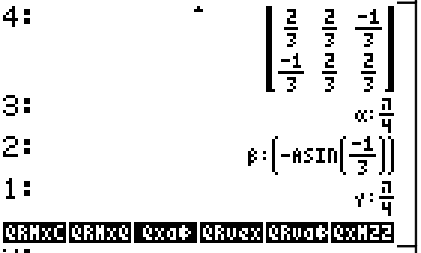
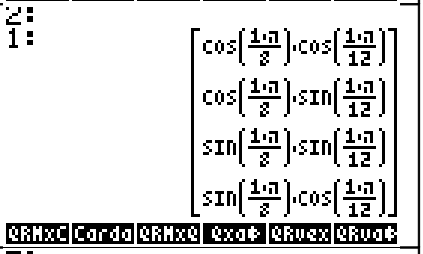
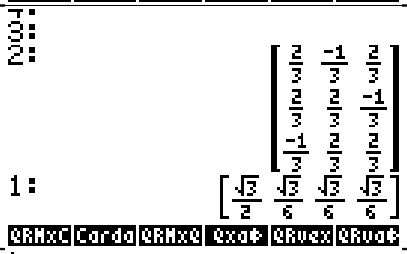
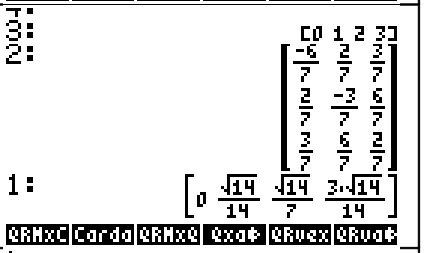
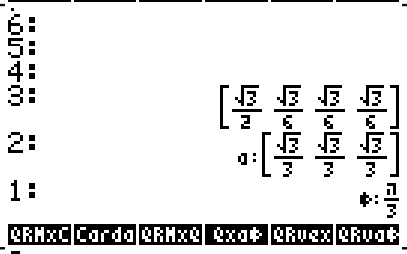
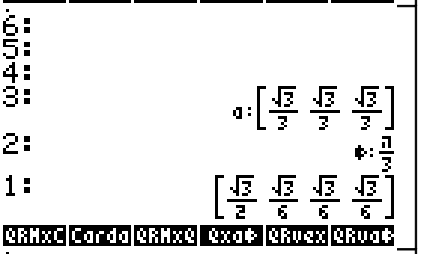
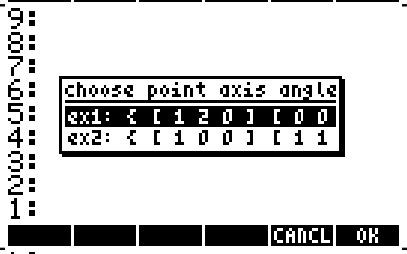
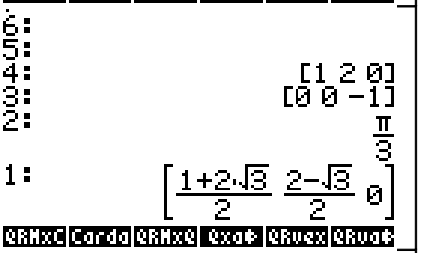
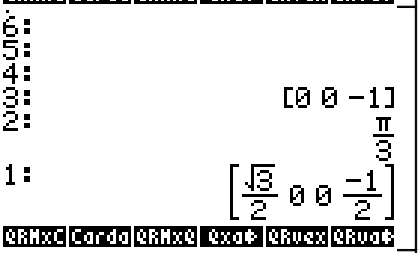
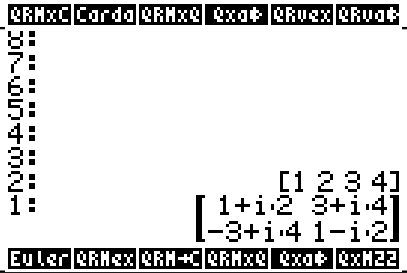
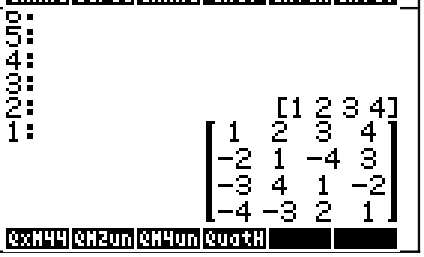
Qasin, Qacos, Qatan (0.5s)

Qsinh, Qcosh, Qtanh (0.5s)





Qasinh, Qacosh, Qatanh (0.5s)

QRMex: choose rotation matrix [OK] shows matrix		
QRMxCardan: rotation matrix to Cardan angles (0.6s) and back (3.4s) 2. example		
Cardan->Q: Cardan angles to quaternion (2.3s)		
QRMxQ: rotation matrix to quaternion (1.7s) QRMxQ: quaternion to rotation matrix (4s) and back (2.7s)		
Qxaφ: quaternion to axis and angle (2s) and back (1.3s)		
QRvex: choose example with vector, axis, angle QRvaφ: rotate vector v around axis a with angle φ with q=[cos(φ/2) sin(φ/2)a] (2.8s)		
QRvaφ: second exmple (5.6s) Qxaφ: quaternion to axis and angle (1s)		
QxM22: quaternion to complex 2x2 matrix (0.3s) QxM22: quaternion to real 4x4 matrix (0.3s)		

QuatHELP: help

```
QUATERN quaternions
Q=[a b c d], a..d real, variables
(Q=a+b×I+c×J+d×K, I×I=J×J=K×K=-1
I×J=-J×I, I×J=K, + cyclic perm.)
use rotation angles in RAD

Qex    _ → Q examples
Qsimp Q → Q' simplify (T)EXPAND
Qadd  Q1 Q2 → Q1+Q2 add
Qsub  Q1 Q2 → Q1-Q2 subtract
Qmul  Q1 Q2 → Q1×Q2 multiply
Qdiv  Q1 Q2 → Q1/Q2 divide
GRAPH                      OK
```

```
      Q1,Q2 = a allowed
Qpow  Q a → Q^a power, Q a can be
      quaternion or number
Qpolar Q → |Q| α n polarform
      Q=|Q|×(COS(α)+n×SIN(α))
      |Q|=√(a^2+b^2+c^2+d^2)
      α=ACOS(a/|Q|)
      n=[0 b c d]/√(b^2+c^2+d^2)
Qconj Q → [a -b -c -d] conjugate
Qinv  Q → Qinv inverse Q×QinV=1
Qunit Q → Q/|Q| unit quaternion
      |Q| is obtained with ABS
GRAPH                      OK
```

QuatHELP: help

```
Qre   Q → [a 0 0 0] real part
QiM   Q → [0 b c d] imagin. part
Qedit Q → edit elements in EQW
Qdsqrt Q → Q', √a/b → a/(√a×b)
      √ → denominator
Functions: Q → F(Q)
Qexp Qln Qsin Qcos Qtan Qasin
Qacos Qatan Qsinh Qcosh Qtanh
Qasinh Qacosh Qatanh Qinverse
Functions only for numeric Q
Functions are like complex
F(x+i×y) with x→a,
GRAPH                      OK
```

```
y→√(b^2+c^2+d^2), i→[0 b c d]/y
Keyboard commands:
+,-   Q1 Q2 +,- → Q1+,-Q2
×,/   Q a → Q×a,Q/a
ABS   Q → |Q|=√(a^2+b^2+c^2+d^2)
→NUM  Q → numerical value

QRMex _ → examples for QRM =
      Quaternion Rot. Matrices
QRMxCardan QRM ↔ α β γ
      QRM to Cardan angles
      R=Rz(γ)·Ry(β)·Rx(α)
GRAPH                      OK
```

QuatHELP: help

```
Cardan→Q α β γ → Q Cardan angles
      to quaternion Q=Qγ·Qβ·Qα
QRMxQ QRM ↔ Q, rotation matrix
      to quaternion and back
Qxaϕ  Q → a ϕ, a ϕ → Q
      quaternion to axis,
      rotation angle  and back
Q=[COS(ϕ/2) (a1 a2 a3)×SIN(ϕ/2)]
EVAL × × gives entire rot.matrix
QRvex _ → v a ϕ example QRvaϕ
QRvaϕ v a ϕ → v' rotate vector v
      around axis a with angle ϕ
GRAPH                      OK
```

```
      v'=q·[0 v]·q^-1
      q=[ COS(ϕ/2) SIN(ϕ/2)·a]
      a=[a1 a2 a3]
QxM22 Q,M22 → M22,Q
      M22 = complex 2x2 matrix
QxM44 Q,M44 → M44,Q
      M44 = real 4x4 matrix
CardanRM _ → {Rγ Rβ Rα} list with
      Cardan rotation matrices
EulerRM  _ → {Rz Rx Rz} list with
      Euler rotation matrices
QM2unit _ → {2x2 unit matrices}
GRAPH                      OK
```