

# SHEL

HelpSHCHR: help Spinor Helicity (notation CHR)	<pre> MASSIVE SPINOR HELICITY LORENTZ TRANSFORMATIONS NOTATION: CHRISTENSEN 1802.00448, 2403.13303 % P % E' LORENTZ INDICES % k % SPIN INDICES &lt;i1,j&gt;=&lt;i1%1-lj%&gt;, &lt;i1,j&gt;=&lt;i1%1-lj%&gt; &lt;i1%&gt;,&lt;i1%&gt;,&lt;i1%&gt;,&lt;i1%&gt; &lt;i1%&gt;,&lt;i1%&gt;,&lt;i1%&gt;,&lt;i1%&gt;  &lt;i1%&gt; - + [ ] MASSIVE SPINORS &lt;i1%&gt; - + [ ] MASSLESS SPINORS JL - + [ ] LITTLE GROUP </pre>	<pre> JL - + [ ] LITTLE GROUP GENERATORS, p INVARIANT ACT FROM LEFT WITH * FOR JL SPINORS &lt;left&gt; J1%&gt; &lt;i1%&gt; &lt;i1%&gt; J1%&gt; &lt;i1%&gt; &lt;i1%&gt; &lt;right&gt; J1%&gt; &lt;i1%&gt; &lt;i1%&gt; J1%&gt; &lt;i1%&gt; &lt;i1%&gt; J1%&gt;=J1%+i1%J2 J2 - + [ ] SPIN GENERATORS ACT FROM RIGHT WITH * ON SPIN INDEX </pre>
HelpSHCHR	<pre> Jg - + [ ] GENERAL LORENTZ ROTATIONS ACT FROM LEFT WITH * Kg - + [ ] GENERAL LORENTZ BOOSTS ACT FROM LEFT WITH * EJL - + [ ] EXPONENTIATED LITTLE GROUP TRANSFORM. A22(i)=EXP(i*x*JL,r) EJs - + [ ] EXPONENTIATED SPIN TRANSFORMATION EJg - + [ ] EXPONENTIATED </pre>	<pre> EJg - + [ ] EXPONENTIATED GENERAL LORENTZ ROTATION A22(r)=EXP(i*x*J2) EKg - + [ ] EXPONENTIATED GENERAL LORENTZ BOOST A22(b)=EXP(i*x*K2) K2=i1%J2 Shul &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; + &lt;right&gt; &lt;i1%&gt; &lt;i1%&gt; + &lt;right&gt; SPINOR MULTIPLICATION HINT: TO MULTIPLY SPINORS &lt;i1%&gt; AND &lt;i1%&gt; USE Shul </pre>
HelpSHCHR	<pre> 4 OR TYPE &lt;i1%&gt; TRAN &lt;i1%&gt; * Sdot [ ] [ ] + [ ] SUM OVER k EX: &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; + p, &lt;i1%&gt; &lt;i1%&gt; + H*% HINT: TO MULTIPLY AND SUM OVER k TYPE USE Sdot OR TYPE &lt;i1%&gt; &lt;i1%&gt; TRAN * JShul (E)JL Sp + Sp' MULTIPLY LITTLE GROUP (EXPON.) GENERATOR WITH Sp(inor) AND SIMPLIFY </pre>	<pre> Expand [ ] + [ ] EXAND AND SIMPLIFY (s*s*c^2=1) Factor [ ] + [ ] FACTORISE P FACTOR i, USE s*s*c^2=1 Lin [ ] + [ ] LINEARISE COS(H), SIN(H) + EXP BUT NOT J Nexp [ ] + [ ] EXP(M) M=[ ] MATRICEXPONENTIAL Mexp [ ] + [ ] EXP(i*x*M) M=[ ] MATRICEXPONENTIAL Mtrhyp [ ] + [ ] TRIG AND HYP </pre>
HelpSHCHR	<pre> EP+i 0 1 + 0' SUBST AFTER EO VAR i,j NUMBER 1,2.. FactSr [ ] + [ ] FACTOR INSIDE SORT. EX: Out(&lt;i1%&gt;,&lt;i1%&gt;) Eips+ [ ] + [ ] s+s% EXP(-i*x*p)s+s%EXP(i*x*p)s% USE AFTER JShul WITH (E)JL+ AND (E)J2 cs2+1 [ ] + [ ] cos^2+sin^2=1 c^2+s*s%=1 c^2+s^2=1 v+ [ ] + [ ] </pre>	<pre> v+ [ ] + [ ] v=LN((E+P)/(E-P)) SUBSTITUTE v RAPIDITY [ ] + [ ] INSERT v AND P=E FOR MASSLESS SPINORS AFTER JL+*%&lt;i1%&gt; ETC x1+ix [ ] + [ ] 1+x + x+1 CHANGE ORDER OF VARS EP+m [ ] + [ ] E^2-P^2 + H^2 s*s+c^2 [ ] + [ ] s*s+1-c^2 c^2+s*s% [ ] + [ ] c^2+1-s*s% cs2+ [ ] + [ ] c^2+c^2-s^2 </pre>
HelpSHCHR	<pre> cs2+ [ ] + [ ] c^2+c^2-s^2 P*s%EXP(i*p)+P*2cs P*s%EXP(-i*p)+P*2cs% + [ ] MOMENTUM SPINOR FROM &lt;i1%&gt; &lt;i1%&gt; Sdot + [ ] MOMENTUM SPINOR FROM &lt;i1%&gt; &lt;i1%&gt; Sdot ExJLS + :JL:[ ] i1%&gt;.. EXAMPLE ACTION OF LITTLE GROUP ON SPINOR. TYPE * Expand v+ Factor + :i1%&gt; :J2:[ ].. </pre>	<pre> ExJLS + :JL:[ ] i1%&gt;.. EXAMPLE MULTIPLY SPINOR WITH SPIN MATRIX + :JL:[ ] p:[ ] JL:[ ] EXAMPLE TRANSFORMATION OF MOMENTUM UNDER LITTLE GROUP, TYPE * p'Jg + TRANSFORMED MOMENTA AFTER ROTATION p'Kg + TRANSFORMED MOMENTA AFTER BOOST 4ST &lt; HelpSLT Shul Sdot </pre>
HelpSHCHR	<pre> 4ST &lt; HelpSLT Shul Sdot JShul Factor EP+i Expand Factor Lin Nexp Mexp Mtrhyp Eips+ cs2+1 v+ P+E EP+m 3  ABBREVIATIONS AND RELATIONS: c=COS(0/2) s=SIN(0/2)*EXP(i*x*p) s%=SIN(0/2)*EXP(-i*x*p) s*EXP(-i*x*p)=s%*EXP(i*x*p)=SIN(0/2) c^2+s*s%=1 c=COS(0)=c^2-s*s% s=2*c^2-1=1-2*s*s% s%=SIN(0)=2*c*s%EXP(i*x*p) =2*c*s%*EXP(-i*x*p) c^2-1=2*c*s*s%, c^2+1=2*c^2 s*EXP(i*x*p)=2*c*s% s*EXP(-i*x*p)=2*c*s% </pre>	<pre> ABBREVIATIONS AND RELATIONS: c=COS(0/2) s=SIN(0/2)*EXP(i*x*p) s%=SIN(0/2)*EXP(-i*x*p) s*EXP(-i*x*p)=s%*EXP(i*x*p)=SIN(0/2) c^2+s*s%=1 c=COS(0)=c^2-s*s% s=2*c^2-1=1-2*s*s% s%=SIN(0)=2*c*s%EXP(i*x*p) =2*c*s%*EXP(-i*x*p) c^2-1=2*c*s*s%, c^2+1=2*c^2 s*EXP(i*x*p)=2*c*s% s*EXP(-i*x*p)=2*c*s% </pre>
Massive spinors	<pre> 4: &lt;i1%&gt;: [sqrt(E+P)*s sqrt(E-P)*c] [ sqrt(E+P)*c sqrt(E-P)*s% ] 3: &lt;i1%&gt;: [sqrt(E+P)*c sqrt(E-P)*s%] [ sqrt(E+P)*s sqrt(E-P)*c ] 2: &lt;i1%&gt;: [sqrt(E-P)*s sqrt(E+P)*c] [ sqrt(E-P)*c sqrt(E+P)*s% ] 1: &lt;i1%&gt;: [sqrt(E-P)*c sqrt(E+P)*s%] [ sqrt(E-P)*s sqrt(E+P)*c ] &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; </pre>	<pre> 4: &lt;i1%&gt;: [sqrt(E-P)*c sqrt(E+P)*s] [ sqrt(E-P)*s% sqrt(E+P)*c ] 3: &lt;i1%&gt;: [sqrt(E-P)*s% sqrt(E+P)*c] [ sqrt(E-P)*c sqrt(E+P)*s ] 2: &lt;i1%&gt;: [sqrt(E+P)*c sqrt(E-P)*s] [ sqrt(E+P)*s% sqrt(E-P)*c ] 1: &lt;i1%&gt;: [sqrt(E+P)*s% sqrt(E-P)*c] [ sqrt(E+P)*c sqrt(E-P)*s ] &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; </pre>
Massless spinors	<pre> 9: [sqrt(E+P)*s sqrt(E-P)*c] 8: [sqrt(E+P)*c sqrt(E-P)*s%] 7: [sqrt(E+P)*c sqrt(E-P)*s%] 6: [sqrt(E+P)*s sqrt(E-P)*c] 5: [sqrt(E-P)*s sqrt(E+P)*c] 4: [sqrt(E-P)*c sqrt(E+P)*s%] 3: [sqrt(E-P)*c sqrt(E+P)*s%] 2: [sqrt(E-P)*s sqrt(E+P)*c] 1: [sqrt(E-P)*s sqrt(E+P)*c] &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; </pre>	<pre> 9: little group gener. JL 8: act from left J1exp 7: JL+P: [ ] 'c*s%EXP(-i 6: JL-P: [ ] 'c*s%EXP(-i 5: JL1P: [ ] 'EXP(-i*x*p)* 4: JL2P: [ ] '-EXP(-i*x*p) 3: JL+P: [ ] 'c*s%*EX 2: JL-P: [ ] 'c*s%*EX 1: JL1P: [ ] '-EXP(i*x </pre>
little group generators JL	<pre> &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; </pre>	<pre> &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; &lt;i1%&gt; </pre>

spin generators Js	<pre> spin generators Js From right (pk)xJasp Js+ki: [[ 0 0 1 ] [-E Js-ki: [[ 0 0 '-EXP(-i Js+ki: [[ 0 0 '-(EXP(-i Js-ki: [[ 0 0 '-(EXP(-i Js+ki: [[ 0 0 '-(EXP(-i Js-ki: [[ 0 0 '-(EXP(-i From right (pk)xJsk Js+ki: [[ 0 0 'EXP(i Js-ki: [[ 0 0 'EXP(i </pre>	<pre> general rot. gen. Jg act from left Jgasp Jg1kB: [[ 0 0 '-1/2' 1 Jg2kB: [[ 0 0 '-i/2' 1 Jg3kB: [[ 0 0 '-1/2' 0 Jg4kB: [[ 0 0 '1/2' 1 Jg5kB: [[ 0 0 '1/2' 1 Jg6kB: [[ 0 0 '-i/2' 1 Jg7kB: [[ 0 0 '1/2' 0 Jg8kB: [[ 0 0 '1/2' 1 </pre>
general rotation generators Jg		
general boost generators Kg	<pre> general boost gen. Kg act from left Kgasp Kg1kB: [[ 0 0 '-i/2' 1 Kg2kB: [[ 0 0 '1/2' 1 Kg3kB: [[ 0 0 '-i/2' 0 Kg4kB: [[ 0 0 '1/2' 1 Kg5kB: [[ 0 0 '1/2' 1 Kg6kB: [[ 0 0 '1/2' 1 Kg7kB: [[ 0 0 '1/2' 0 Kg8kB: [[ 0 0 '-i/2' </pre>	<pre> SHUL((ik1,lik1)) [0 -H H 0] SHUL((ik1,lik1)) [0 H -H 0] Sdot(lik1,lik1) [E+(c^2-sis*)p 2p*is*c 2p*icis E-(c^2-sis*)p] [ik1 lik1] &lt;i1 i2&gt; [i1 i2] </pre>
spinor products		
JSmul: little group generator Jl multiply with Spinor	<pre> Jl+KB: [-cis*ie^iD e^n c^2e^iD -sis*ie^-iD e^n cis*ie^-iD [ik1: [sqrt(E-F)c sqrt(E+F)c] [sqrt(E-F)c sqrt(E+F)c] 1: [-cis*ie^iD e^n c^2e^iD -sis*ie^-iD e^n cis*ie^-iD </pre>	<pre> Jl+KB: [cis*ie^-iD e^-n sis*ie^-iD e^-n -c^2e^iD e^-n -cis*ie^iD e [ik1: [sqrt(E+F)c sqrt(E-F)c] [sqrt(E+F)c sqrt(E-F)c] 1: [sis*sqrt(E-F)e^iD 0 -cis*sqrt(E-F)e^iD 0 </pre>
JSmul: spin generator Js multiply with Spinor	<pre> [ik1: [sqrt(E+F)c sqrt(E-F)c] [sqrt(E+F)c sqrt(E-F)c] Js+ki: [0 -e^iD 0] [-(e^iD sqrt(E-F)c) 0] [-(e^iD sqrt(E-F)c) 0] </pre>	<pre> [ik1: [sqrt(E+F)c sqrt(E-F)c] [sqrt(E+F)c sqrt(E-F)c] Js-ki: [0 -e^-iD 0] [0 -(e^-iD sqrt(E+F)c)] [0 -(e^-iD sqrt(E+F)c)] </pre>
Ejl, Ejs: exponentiated generators	<pre> EXP(i*W*Jl1)F= cos(W/2)-i(c^2-sis*)sin(W/2) -2i*sis*c sin(W/2) cos "EXP(i*W*Js1)k," cos(W/2) e^iD i sin(W/2) i sin(W/2) e^-(iD) cos(W/2) </pre>	<pre> EXP(i*W*Jg3)KB: [e^W/2 0 0 e^-W/2] EXP(i*W*Jg3)KB' I: [e^W/2 0 0 e^-W/2] </pre>
Ekg: exponentiated generators	<pre> EXP(i*W*Kg3)KB: [e^W/2 0 0 e^-W/2] EXP(i*W*Kg3)KB' I: [e^W/2 0 0 e^-W/2] </pre>	<pre> [ik1: [sqrt(E+F)c sqrt(E-F)c] [sqrt(E+F)c sqrt(E-F)c] </pre>
EP->i change momentum to i		
Mexp: matrix exponential	<pre> i*W*Kg1KB: [0 W/2] [W/2 0] cosh(W/2) sinh(W/2) sinh(W/2) cosh(W/2) </pre>	<pre> Jl+KB: [cis*ie^-iD e^-n sis*ie^-iD e^-n -c^2e^iD e^-n -cis*ie^iD e [ik1: [sqrt(E+F)c sqrt(E-F)c] [sqrt(E+F)c sqrt(E-F)c] 1: [sis*sqrt(E-F)e^iD 0 -cis*sqrt(E-F)e^iD 0 </pre>
Jl+  ik>: little group generator x spinor simplified		
ik> Js+ : spinor x spin generator simplified	<pre> [ik1: [sqrt(E+F)c sqrt(E-F)c] [sqrt(E+F)c sqrt(E-F)c] Js+ki: [0 -e^iD 0] [-(e^iD sqrt(E-F)c) 0] [-(e^iD sqrt(E-F)c) 0] </pre>	<pre> [ik1: [sqrt(E+F)c sqrt(E-F)c] [sqrt(E+F)c sqrt(E-F)c] Js-ki: [0 -e^-iD 0] [0 -(e^-iD sqrt(E+F)c)] [0 -(e^-iD sqrt(E+F)c)] </pre>
ik> Js- : spinor x spin generator simplified		