

```

1  /*****
2  * Package: GSM
3  * Class : EW2Loop
4  *
5  * Description:
6  *   Electroweak two loop calculation for ew formfactors
7  *   of order O(alpha^2*mt^2/MW^2)
8  *
9  * Sources:
10 * (1) G. Degrassi, P. Gambino, Nucl. Phys. B567 (2000) 3-31, hep-ph/9905472
11 * (2) G. Degrassi, P. Gambino, A. Vicini,
12 *     Phys.Lett. B383 (1996) 219-226, hep-ph/9603374
13 * (3) Degrassi, Bardin, The Standard Model in the Making, Oxford 1999
14 *
15 * The Code of this class has been ported to C++ from Fortran code authored
16 * by G. Degrassi and P. Gambino that is also used in the Fortran package
17 * ZFITTER
18 *
19 *****/
20 #include "TMath.h"
21
22 #include "Gfitter/GMath.h"
23 #include "Gfitter/GConstants.h"
24 #include "Gfitter/GTheory.h"
25 #include "Gfitter/GTheoryRef.h"
26 #include "Gfitter/GParameterRef.h"
27 #include "Gfitter/GReference.h"
28 #include "Gfitter/GVariable.h"
29 #include "Gfitter/GStore.h"
30
31 #include "GSM/EW2Loop.h"
32 #include "GSM/ZMath.h"
33
34 using std::complex;
35
36 using namespace Gfitter;
37
38 GSM::EW2Loop::EW2Loop()
39 : Gfitter::GAuxTheory(),
40   m_isUpToDate_Update( kFALSE )
41 {
42

```

Hinweis:

Kommentare mit Hinweisen auf ZFitter sind grün markiert

Übereinstimmungen sind gelb markiert.

Auffällige Stellen bzw. Anmerkungen sind violett markiert

Anhang 9

zum Gutachten DESY ZFitter_GFitter vom 17.03.2014

Lübeck, den 17. März 2014

U. Obermüller
 Dr. Ulrich Obermüller
 -Sachverständiger-
 Sachverständiger für
 Systeme und Anwendungen
 der Informationsverarbeitung,
 insbesondere Individualsoftware
 und
 Software-Qualitätssicherung
 öffentlich bestellt und vereidigt

```

43 SetTheoryName( GetName() );
44 SetExistDerivative( kFALSE );
45
46 const TString& logMH = gStore()->GetVariable( "GSMFlags::logMH" )->GetStringValue();
47 m_logger << kINFO << "Using logMH: \"" << logMH << "\" << GEndl;
48
49 if (logMH == "Yes" ) m_logMH = kTRUE;
50 else if (logMH == "No" ) m_logMH = kFALSE;
51 else {
52     m_logger << kFATAL << "unknown value for \"GSMFlags::logMH\": \"" << logMH << "\"
53         << ". Possible are: \"Yes\" and \"No\""
54         << GEndl;
55 }
56
57 BookParameter( "MZ" , & p_MZ );
58 BookParameter( "mt" , & p_mt );
59 BookParameter( "GF" , & p_GF );
60 BookTheory ( "GSM::WMass" , & t_MW );
61 BookTheory ( "GSM::MH" , & t_MH );
62 }
63
64 void GSM::EW2Loop::UpdateLocalFlags( GReference& /* ref */ )
65 {
66     m_isUpToDate_Update = kFALSE;
67 }
68
69 void GSM::EW2Loop::Update()
70 {
71     if (m_isUpToDate_Update) return;
72
73     // now, it is uptodate (I mean... it will be)
74     m_isUpToDate_Update = kTRUE;
75
76     m_MZ = p_MZ;
77     Double_t mt = p_mt;
78     Double_t MW = GetMW();
79     Double_t MH = GetMH().GetValue(); //p_MH;
80     if( m_logMH ) MH = TMath::Exp( GetMH().GetValue() ); //p_MH );
81
82     // protection by shifting of Higgs mass if it is = 2*mt or 2*mz
83     if( TMath::Abs(MH - 2.0*mt) < 5e-5 ) MH = 2.0*mt - 1e-4;
84     if( TMath::Abs(MH - 2.0*m_MZ) < 5e-5 ) MH = 2.0*m_MZ - 1e-4;

```

```

85
86 m_Nc = 3;
87
88 m_c2 = GMath::IPow( MW/m_MZ , 2 );
89 m_s2 = 1.0 - m_c2;
90
91 // 't Hooft mass, here MZ
92 m_mu = m_MZ;
93
94 m_sman = m_MZ*m_MZ;
95 m_zt = GMath::IPow( m_MZ/mt, 2 );
96 m_ht = GMath::IPow( MH/mt, 2 );
97 m_xt = p_GF/TMath::Sqrt(2.0)*mt*mt/(8.0*GMath::IPow( TMath::Pi() , 2 ));
98
99 Double_t ec = ECharge ( m_zt, m_ht, m_s2, m_MZ, m_mu );
100 Double_t Drho = deltarho ( m_zt, m_ht, m_s2, m_MZ, m_mu );
101 Double_t DrhoOS = deltarhoOS( m_zt, m_s2, m_MZ, m_mu );
102
103 m_deltarw = deltarw ( m_zt, m_ht, m_s2, m_MZ, m_mu );
104 m_deltarwOS = deltarwOS( m_zt, m_s2, m_MZ, m_mu );
105
106 m_kappa = kappa2L ( m_zt, m_ht, m_c2, m_MZ, m_mu );
107
108 m_deltaeta = deltaeta ( m_zt, m_ht, m_c2, m_MZ, m_mu );
109
110 m_Drho2L = m_Nc*m_xt*m_xt*(Drho + 4.0*m_zt*m_c2*DrhoOS);
111 m_Dkap2L = m_Nc*m_xt*m_xt*(Drho + 4.0*m_zt*m_c2*DrhoOS);
112 m_DrdRem = ( m_Nc*GMath::IPow( (GConstants::alphaQED)/(4.0*TMath::Pi()*m_s2)), 2 )
113             *GMath::IPow( (mt*mt/(4.0*MW*MW)), 2 )
114             *(16.0*m_deltarw + 4.0*m_zt*m_c2*m_deltarwOS - 16.0*ec );
115
116 SetupToDate();
117 }

```

118

119

120

Match 1

```

121 Double_t GSM::EW2Loop::deltarw(const Double_t& zt, const Double_t& ht, const Double_t& s2, const Double_t& mz, const Double_t& mu)
122 {
123   Double_t c2 = 1.0 - s2;
124   Double_t mz2 = mz*mz;
125   Double_t mu2 = mu*mu;
126   Double_t drw = 0;
127   Double_t pi = TMath::Pi();
128
129   // [Aww (mw^2) - Aww(0)]/mw^2
130   if( TMath::Sqrt(ht) < 0.3){
131     // light Higgs case
132     // see eq. (7a) of hep-ph/9603374 and ep. (A6) of hep-ph/9905472
133     drw = ( 35.0*ht/288.0 - 5.0*ht*ht/(144.0*c2*zt) - 41.0*zt/96.0 - zt/(48.0*c2) + 325.0*c2*zt/144.0
134           - c2*TMath::Sqrt(ht)*pi*zt/3.0 - c2*pi*pi*zt/36.0
135           + ZMath::B0(c2*mz2, ht*mz2/zt,c2*mz2,mu2)*(- 5.0*ht/36.0 + 5.0*ht*ht/(144.0*c2*zt) + 5.0*c2*zt/12.0)
136           + ZMath::B0(c2*mz2, c2*mz2, mz2,mu2)*(5.0*zt/12.0 + zt/(48.0*c2) - c2*zt/2.0)
137           + (5.0*( -1.0)*ht/144.0 - 19.0*zt/12.0 - (425.0*c2*zt/144.0) + 25.0*zt/(16.0*s2)
138           - 5.0*c2*c2*zt*zt/(16.0*(ht - c2*zt))*TMath::Log(c2) + zt*GMath::IPow( TMath::Log(c2), 2 )/2.0
139           + c2*zt*GMath::IPow( TMath::Log(c2), 2 )/2.0
140           + zt*GMath::IPow( TMath::Log(c2), 2 )/(2.0*(-s2)) - 5.0*ht*TMath::Log(ht)/48.0
141           + 5.0*ht*ht*TMath::Log(ht)/(144.0*c2*zt) + 5.0*c2*zt*TMath::Log(ht)/16.0
142           + 5.0*c2*c2*GMath::IPow( zt, 2 )*TMath::Log(ht)/(16.0*(ht - c2*zt))
143           + ( - 5.0*ht/36.0 + 5.0*ht*ht/(144.0*c2*zt) + 5.0*zt/12.0
144           + zt/(48.0*c2) - 7.0*c2*zt/3.0 + 2.0*zt*TMath::Log(c2)
145           + 2.0*c2*zt*TMath::Log(c2) - 2.0*zt*TMath::Log(c2)/s2)
146           *(TMath::Log(1.0/zt) + TMath::Log(mz2/mu2) )
147           + ( -5.0*ht/144.0 + 5.0*zt/12.0 + zt/(48.0*c2) - 209.0*c2*zt/144.0
148           - 5.0*c2*c2*zt*zt/(16.0*(ht - c2*zt)) + zt*TMath::Log(c2) + c2*zt*TMath::Log(c2)
149           + zt*TMath::Log(c2)/(-s2))*TMath::Log(zt) );

```

```

321   subroutine ew2ldeltarw(zt,ht,drs2lew,s2,mz,mu)
322   real*8 zt,ht,drs2lew,s2,c2,mz,ln,vertex,aww,mz2,B0,gdeg,phi,
323   1      ddilog,mu,mu2
324
325   double precision gmu,repiggh0,mz0,alfa0,alfastr,delQCD,
326   1      pi,mel,mmu,mtau,mb,gamz,Nc,zeta3
327
328   common/datcom/gmu,repiggh0,mz0,alfa0,alfastr,delQCD,
329   1      pi,mel,mmu,mtau,mb,gamz,Nc,zeta3
330   c2 = 1.d0 - s2
331   mz2 = mz*mz
332   mu2 = mu*mu
333   if(dsqrt(ht).lt.0.3d0) then
334     LIGHT HIGGS CASE
335
336     [Aww (mw^2) - Aww(0)]/mw^2
337
338     aww =
339   1  35.d0*ht/288.d0-5.d0*ht*ht/(144.d0*c2*zt)- 41.d0*zt/96.d0 -
340   2  zt/(48.d0*c2) +325.d0*c2*zt/144.d0 - c2*dsqrt(ht)*pi*zt/3.d0-
341   3  c2*pi*pi*zt/36.d0+B0(c2*mz2, ht*mz2/zt,c2*mz2,mu2)*
342   4  (-5.d0*ht/36.d0+5.d0*ht*ht/(144.d0*c2*zt)+5.d0*c2*zt/12.d0) +
343   5  B0(c2*mz2, c2*mz2, mz2,mu2)*
344   6  (5.d0*zt/12.d0 + zt/(48.d0*c2) - c2*zt/2.d0) +
345   9  (5.d0*(-1.d0)*ht/144.d0-19.d0*zt/12.d0-(425.d0*c2*zt/144.d0)+
346   $ 25.d0*zt/(16.d0*s2)-5.d0*c2*c2*zt*zt/(16.d0*(ht-c2*zt)))*
347   1  dlog(c2) + zt*dlog(c2)**2.d0/2.d0 + c2*zt*dlog(c2)**2.d0/2.d0 +
348   2  zt*dlog(c2)**2.d0/(2.d0*(-s2)) -5.d0*ht*dlog(ht)/48.d0 +
349   3  5.d0*ht*ht*dlog(ht)/(144.d0*c2*zt) +5.d0*c2*zt*dlog(ht)/16.d0 +
350   4  5.d0*c2*c2*zt**2.d0*dlog(ht)/(16.d0*(ht - c2*zt)) +
351   5  (-5.d0*ht/36.d0 + 5.d0*ht*ht/(144.d0*c2*zt) + 5.d0*zt/12.d0 +
352   6  zt/(48.d0*c2) -7.d0*c2*zt/3.d0 + 2.d0*zt*dlog(c2) +
353   7  2.d0*c2*zt*dlog(c2) - 2.d0*zt*dlog(c2)/s2)*(dlog(1.d0/zt) +
354   $      dlog(mz2/mu2) ) +
355   8  (-5.d0*ht/144.d0 +5.d0*zt/12.d0 + zt/(48.d0*c2) -
356   9  209.d0*c2*zt/144.d0 -5.d0*c2*c2*zt*zt/(16.d0*(ht - c2*zt)) +
357   $  zt*dlog(c2) + c2*zt*dlog(c2) +zt*dlog(c2)/(-s2))*dlog(zt)
358
359   else

```

```

151 else {
152 // heavy Higgs case
153 // see eq. (7b) of hep-ph/9603374 and ep. (A7) of hep-ph/9905472
154 drw = ( c2*(4.0 - ht)*zt*ZMath::Lambda(-1.0 + 4.0/ht)/(12.0*ht)
155 + c2*(19.0 + 21.0*ht - 12.0*ht*ht - 31.0*GMath::IPow( ht, 3 )
156 + 9.0*GMath::IPow( ht, 4 ))*zt*TMath::DiLog(1.0 - ht)/(72.0*ht*ht)
157 + ((1.0 + 69.0*c2 - 145.0*c2*c2)*zt*TMath::Log(c2))/(48.0*(-s2))
158 + (zt*( - 228.0*c2*c2*ht + 228.0*GMath::IPow( c2, 3 )*ht + 18.0*ht*ht
159 + 351.0*c2*ht*ht - 2226.0*c2*c2*ht*ht + 1857.0*GMath::IPow( c2, 3 )*ht*ht
160 - 36.0*c2*c2*GMath::IPow( ht, 3 ) + 36.0*GMath::IPow( c2*ht, 3 )
161 + 38.0*c2*c2*pi*pi - 38.0*GMath::IPow( c2, 3 )*pi*pi + 42.0*c2*c2*ht*pi*pi
162 - 42.0*GMath::IPow( c2, 3 )*ht*pi*pi
163 + ZMath::B0(c2*mz2, c2*mz2, mz2,mu2)
164 *(-18.0*ht*ht - 342.0*c2*c2*ht*ht + 792.0*c2*c2*ht*ht - 432.0*GMath::IPow( c2, 3 )*ht*ht)
165 + ZMath::gdeg(ht)*(144.0*c2*c2*s2*std::pow(ht, 2.5) - 36.0*c2*c2*s2*std::pow( ht, 3.5 ))
166 + 432.0*GMath::IPow( c2, 3 )*ht*ht*GMath::IPow( TMath::Log(c2), 2 ))/(864.0*c2*(-s2)*ht*ht)
167 + (c2*(- 31.0 + 2.0*ht - 9.0*ht*ht + 3.0*GMath::IPow( ht, 3 ))*zt*TMath::Log(ht))/(72.0*ht)
168 + ((1.0 + 20.0*c2 - 132.0*c2*c2)*zt/(48.0*c2) + (2.0*c2*c2*zt*TMath::Log(c2))/( -s2))
169 *(TMath::Log(1.0/zt) + TMath::Log(mz2/mu2))
170 + (zt*( -1.0 - 19.0*c2 + 93.0*c2*c2 - 73.0*GMath::IPow( c2, 3 )
171 + 48.0*GMath::IPow( c2, 3 )*TMath::Log(c2))*TMath::Log(zt))/(48.0*c2*(-s2))
172 + (c2*( - 24.0 + 74.0*ht + 35.0*ht*ht - 49.0*GMath::IPow( ht, 3 )
173 + 9.0*GMath::IPow( ht, 4 ))*zt*ZMath::Phi(ht/4.0))/(144.0*ht*ht) );
174 }
175
176 //vertex contribution
177
178 drw += ( zt*( ( - 2.0*c2 + 5.0*c2*c2)*TMath::Log(c2)/(2.0*(-s2))
179 + (3.0*c2 - 3.0*c2*c2 - c2*c2*GMath::IPow( TMath::Log(c2), 2 ))/(2.0*(-s2))
180 + ( -c2 + c2*c2 - c2*c2*TMath::Log(c2))/( -s2)
181 *(TMath::Log(1.0/zt) + 2.0*TMath::Log(mz2/mu2)) ) );
182
183 // see also m2tcor5_11.f line 328
184 return drw;
185 }
186

```

Ende Match 1

```

359 else
360 c
361 c HEAVY HIGGS CASE
362 c
363
364 aww =
365 1 c2*(4.0 - ht)*zt*ln(-1.0 + 4.0/ht)/(12.0*ht) +
366 2 c2*(19.0+21.0*ht-12.0*ht*ht-31.0*ht**3.0+9.0*ht**4.0)*zt*
367 3 ddilog(1.0 - ht)/(72.0*ht*ht) +
368 4 ((1.0 + 69.0*c2 - 145.0*c2*c2)*zt*dlog(c2))/(48.0*(-s2)) +
369 5 (zt*(-228.0*c2*c2*ht + 228.0*c2**3.0*ht + 18.0*ht*ht +
370 6 351.0*c2*ht*ht - 2226.0*c2*c2*ht*ht + 1857.0*c2**3.0*ht*ht -
371 7 36.0*c2*c2*ht**3.0+36.0*c2**3.0*ht**3.0+38.0*c2*c2*pi*pi -
372 8 38.0*c2**3.0*pi*pi+42.0*c2*c2*ht*pi*pi-42.0*c2**3.0*ht*pi*pi
373 9 pi + B0(c2*mz2, c2*mz2, mz2,mu2)*(-18.0*ht*ht-342.0*c2*ht*ht +
374 $ 792.0*c2*c2*ht*ht - 432.0*c2**3.0*ht*ht) +
375 3 gdeg(ht)
376 * (144.0*c2*c2*s2*ht**(2.5d0)-36.0*c2*c2*s2*ht**(3.5d0))+
377 4 432.0*c2**3.0*ht*ht*dlog(c2)**2.d0)/(864.0*c2*(-s2)*ht*ht)+
378 5 (c2*(-31.0+2.0*ht-9.0*ht*ht+3.0*ht**3.0)*zt*dlog(ht))/
379 6 (72.0*ht) + ((1.0 + 20.0*c2 - 132.0*c2*c2)*zt/(48.0*c2) +
380 7 (2.0*c2*c2*zt*dlog(c2))/(-s2))*(dlog(1.0/zt) +dlog(mz2/mu2)) +
381 8 (zt*(-1.0-19.0*c2+93.0*c2*c2-73.0*c2**3.0+48.0*c2**3.0*
382 9 dlog(c2))*dlog(zt))/(48.0*c2*(-s2)) + (c2*(-24.0 + 74.0*ht +
383 $ 35.0*ht*ht-49.0*ht**3.0+9.0*ht**4.0)*zt*phi(ht/4.0))/
384 1 (144.0*ht*ht)
385 endif
386
387 c
388 c vertex contribution
389 c
390 vertex =
391 1 zt*( (-2.0*c2 + 5.0*c2*c2)*dlog(c2)/(2.0*(-s2)) +
392 2 (3.0*c2-3.0*c2*c2-c2*c2*dlog(c2)**2.d0)/(2.0*(-s2)) +
393 3 (-c2 + c2*c2 - c2*c2*dlog(c2))/(-s2)*(dlog(1.0/zt) +
394 4 2.0*dlog(mz2/mu2)) )
395
396 drs2lew = aww + vertex
397 return
398 end

```

```

188 Double_t GSM::EW2Loop::ECharge(const Double_t& zt, const Double_t& ht, const Double_t& s2, const Double_t& mz, const Double_t& mu)
189 {
190     Double_t c2 = 1.0 - s2;
191     Double_t mz2 = mz*mz;
192     Double_t mu2 = mu*mu;
193     Double_t pi = TMath::Pi();
194     Double_t ec = 0;
195
196     // electric charge in units
197     // Nc*(alfa0/(4.d0*pi*s2))^2/(zt*c2)^2
198
199     if( TMath::Sqrt(ht) < 0.3 ){
200         // light Higgs case
201         // see (8.71) of The Standad Model in the Making
202         // (B5) of hep-ph/9905472 and m2tcor5_11.f line 431
203         ec = ( -s2*c2*zt*
204             ( (2.0 - TMath::Log(ht))/9.0
205               + (100.0 + 256.0*( -4.0 + 2.0*TMath::Sqrt(ht)*pi) + 96.0*TMath::Log(ht))/864.0
206               + 13.0/18.0*(TMath::Log(1.0/zt) + TMath::Log(mz2/mu2)) ) );
207     }
208
209     else{
210
211         // heavy Higgs case
212         // see (8.72) of The Standad Model in the Making,
213         // (B5) of hep-ph/9905472 and m2tcor5_11.f line 440
214         ec = ( -s2*c2*zt*( (100.0 - 25.0*ht + 96.0*TMath::Log(ht) + 48.0*ht*TMath::Log(ht)
215             + 256.0*( -4.0 + ht/2.0 - ( -4.0 + ht)*TMath::Sqrt(ht)*ZMath::gdeg(ht)/4.0
216             + (6.0 - ht)*ht*TMath::Log(ht)/4.0)
217             - 64.0*ht*( -4.0 + ht/2.0 - ( -4.0 + ht)*TMath::Sqrt(ht)*ZMath::gdeg(ht)/4.0
218             + (6.0 - ht)*ht*TMath::Log(ht)/4.0)/(216.0*(4.0 - ht))
219             + 13.0/18.0*(TMath::Log(1.0/zt) + TMath::Log(mz2/mu2))
220             + 4.0*( -1.0 + ht)*ZMath::Phi(ht/4.0)/(9.0*( -4.0 + ht)*ht) );
221     }
222
223     return ec;
224 }

```

Match 2

```

403
404
405 c
406
407 1
408 c
409
410 1
411
412
413
414
415 c
416
417 1
418 2
419 3
420 c
421 c
422
423 c
424 c
425 c
426
427
428 1
429 2
430 3
431 4
432 5
433 c
434
435 c
436
437

```

```

subroutine ew2ltwodel(zt,ht,deleoe2lew,s2,mz,mu)
real*8 zt,ht,deleoe2lew,s2,c2,mz,phi,gdeg,mu,mz2,mu2

double precision gm,repiggh0,mz0,alfa0,alfastr,delQCD,
pi,mel,mmu,mtau,mb,gamz,Nc,zeta3

common/datcom/gmu,repiggh0,mz0,alfa0,alfastr,delQCD,
pi,mel,mmu,mtau,mb,gamz,Nc,zeta3

c2 = 1.d0 - s2
mz2 = mz*mz
mu2= mu*mu

if(dsqrt(ht).lt.0.3d0) then
LIGHT HIGGS CASE
deleoe2lew =
1 -s2*c2*zt*( (2.d0-dlog(ht))/9.d0+(100.d0+256.d0*(-4.d0 + 2.d0*
2 dsqrt(ht)*pi)+96.d0*dlog(ht))/864.d0 +
3 13.d0/18.d0*(dlog(1.d0/zt)+ dlog(mz2/mu2)) )

else

HEAVY HIGGS CASE

deleoe2lew =
$ -s2*c2*zt*((100.d0-25.d0*ht+96.d0*dlog(ht)+48.d0*ht*dlog(ht) +
1 256.d0*(-4.d0+ht/2.d0-(-4.d0 + ht)*dsqrt(ht)*gdeg(ht)/4.d0 +
2 (6.d0 - ht)*ht*dlog(ht)/4.d0)-64.d0*ht*(-4.d0 + ht/2.d0 -
3 (-4.d0+ht)*dsqrt(ht)*gdeg(ht)/4.d0+(6.d0-ht)*ht*dlog(ht)/4.d0))/
4 (216.d0*(4.d0-ht))+13.d0/18.d0*(dlog(1.d0/zt)+dlog(mz2/mu2)) +
5 4.d0*(-1.d0 + ht)*phi(ht/4.d0)/(9.d0*(-4.d0 + ht)*ht)

endif

return
end

```

Match 3

```

226 Double_t GSM::EW2Loop::deltarho(const Double_t& zt, const Double_t& ht, const Double_t& s2, const Double_t& mz, const Double_t& mu)
227 {
228     Double_t c2 = 1.0 - s2;
229     Double_t mz2 = mz*mz;
230     Double_t mu2 = mu*mu;
231     Double_t mt = mz/TMath::Sqrt(zt);
232     Double_t sht = TMath::Sqrt(ht);
233     Double_t pi = TMath::Pi();
234     Double_t dr = 0;
235
236     // [Aww (mw^2)/mw^2 - Azz(mz^2)/mw^2] in units
237     // Nc*(alfa0/(4.0*pi*s2))^2/(4.0*zt*c2)^2.0
238
239     if( ht < 0.25 ) {
240         // light Higgs case
241         // see eq. (10a) of hep-ph/0603374 and (A6) of hep-ph/9905472
242         dr = ( 19.0 - (53.0*ht)/3.0 - 4.0*TMath::Sqrt(ht)*pi - 2.0*pi*pi + 2.0*ht*pi*pi
243             + 8.0*ht*ht/(9.0*zt) - 5.0*ht*ht/(9.0*c2*zt) + 3.0/2.0*std::pow( ht, 1.5 )*pi
244             + ZMath::B0(mz2, ht*mz2/zt, mz2,mu2)*(32.0*ht/9.0 - 8.0*ht*ht/(9.0*zt))
245             + ZMath::B0(c2*mz2, ht*mz2/zt, c2*mz2,mu2)*(- 20.0*ht/9.0 + 5.0*ht*ht/(9.0*c2*zt))
246             - 5.0*ht*TMath::Log(c2)/9.0 - 2.0*ht*TMath::Log(ht) - 8.0*ht*ht*TMath::Log(ht)/(9.0*zt)
247             + 5.0*ht*ht*TMath::Log(ht)/(9.0*c2*zt)
248             + ht*(5.0*ht - 8.0*c2*ht + 12.0*c2*zt)*(TMath::Log(1.0/zt) + TMath::Log(mz2/mu2))/(9.0*c2*zt)
249             + ht*TMath::Log(zt)/3.0
250             + zt*(( - 9.0 + 845.0*c2 + 427.0*c2*c2 - 366.0*GMath::IPow( c2, 3 ) + 136.0*c2*TMath::Sqrt(ht)*pi
251                 - 464.0*c2*c2*TMath::Sqrt(ht)*pi + 256.0*GMath::IPow( c2, 3 )*TMath::Sqrt(ht)*pi
252                 - 119.0*c2*pi*pi + 44.0*c2*c2*pi*pi - 396.0*c2*ZMath::B0(mz2, ht*mz2/zt, mz2, mu2)
253                 + ZMath::B0(mz2, c2*mz2, c2*mz2, mu2)*(- 18.0*c2 - 324.0*c2*c2 + 288.0*GMath::IPow( c2, 3 ))
254                 + 180.0*c2*c2*ZMath::B0(c2*mz2,ht*mz2/zt,c2*mz2,mu2)
255                 + ZMath::B0(c2*mz2, c2*mz2, mz2, mu2)*(9.0 + 180.0*c2 - 216.0*c2*c2))/(27.0*c2)
256                 + (- 3.0 - 32.0*c2 - 48.0*c2*c2)*TMath::Log(c2)/9.0
257                 + (3.0 - 10.0*c2 - 328.0*c2*c2 + 56.0*GMath::IPow( c2, 3 ))
258                 *(TMath::Log(1.0/zt) + TMath::Log(mz2/mu2))/(9.0*c2)
259                 + (3.0 - 11.0*c2 - 16.0*c2*c2 - 48.0*GMath::IPow( c2, 3 ))*TMath::Log(zt)/(9.0*c2))
260             + 4.0*zt*ZMath::B0(mz2, ht*mz2/zt, mz2,mu2) );
261     }

```

```

445 subroutine ew2ltobf(zt,ht,tobf2lew,s2,mz,mu)
446 real*8 zt,ht,tobf2lew,s2,c2,mz,mz2,gdeg,ddilog,b0,ln,mu,mu2,phi,
447 1 sht,mt
448 c
449 double precision gmu,repiggh0,mz0,alfa0,alfastr,delQCD,
450 1 pi,mel,mmu,mtau,mb,gamz,Nc,zeta3
451 c
452 common/datcom/gmu,repiggh0,mz0,alfa0,alfastr,delQCD,
453 1 pi,mel,mmu,mtau,mb,gamz,Nc,zeta3
454 c
455 c2 = 1.d0 - s2
456 mz2 = mz*mz
457 mu2 = mu*mu
458 mt = mz/Dsqrt(zt)
459 sht = Dsqrt(ht)
475 if(ht.lt.0.25d0) then
476 LIGHT HIGGS CASE
477 c
478 tobf2lew =
479 1 19.d0 - (53.d0*ht)/3.d0 - 4.d0*dsqrt(ht)*pi - 2.d0*pi*pi +
480 2 2.d0*ht*pi*pi + 8.d0*ht*ht/(9.d0*zt) - 5.d0*ht*ht/(9.d0*c2*zt)+
481 3 3.d0/2.d0*ht** (3.d0/2.d0)*pi + B0(mz2, ht*mz2/zt, mz2,mu2)*
482 4 (32.d0*ht/9.d0 - 8.d0*ht*ht/(9.d0*zt)) +
483 5 B0(c2*mz2, ht*mz2/zt, c2*mz2,mu2)*(-20.d0*ht/9.d0+
484 6 5.d0*ht*ht/(9.d0*c2*zt))-5.d0*ht*dlog(c2)/9.d0 -
485 7 2.d0*ht*dlog(ht) - 8.d0*ht*ht*dlog(ht)/(9.d0*zt) +
486 8 5.d0*ht*ht*dlog(ht)/(9.d0*c2*zt)+
487 9 ht*(5.d0*ht-8.d0*c2*ht+12.d0*c2*zt)*(dlog(1.d0/zt) +
488 $ Dlog(mz2/mu2))/(9.d0*c2*zt)+
489 $ ht*dlog(zt)/3.d0 + zt*((-9.d0 + 845.d0*c2 + 427.d0*c2*c2 -
490 1 366.d0*c2**3.d0+136.d0*c2*dsqrt(ht)*pi-
491 2 464.d0*c2*c2*dsqrt(ht)*pi+ 256.d0*c2**3.d0*dsqrt(ht)*pi -
492 3 119.d0*c2*pi*pi + 44.d0*c2*c2*pi*pi -
493 4 396.d0*c2*B0(mz2, ht*mz2/zt, mz2, mu2) +
494 5 B0(mz2, c2*mz2, c2*mz2, mu2)*(-18.d0*c2 -324.d0*c2*c2 +
495 6 288.d0*c2**3.d0)+180.d0*c2*c2*B0(c2*mz2,ht*mz2/zt,c2*mz2,mu2) +
496 7 B0(c2*mz2, c2*mz2, mz2, mu2)*(9.d0+180.d0*c2 -
497 8 216.d0*c2*c2))/(27.d0*c2) + (-3.d0-32.d0*c2-48.d0*c2*c2)*
498 9 dlog(c2)/9.d0 + (3.d0-10.d0*c2-328.d0*c2*c2 + 56.d0*c2**3.d0)*
499 $ (dlog(1.d0/zt)+dlog(mz2/mu2))/(9.d0*c2)+(3.d0-11.d0*c2 -
500 $ 16.d0*c2*c2 - 48.d0*c2**3.d0)*dlog(zt)/(9.d0*c2)) +
501 2 4.d0*zt*B0(mz2, ht*mz2/zt, mz2,mu2)
502 c
503 else

```

```

263 else if(ht < 4.0) {
264 // Interpolation between light and heavy Higgs case
265 // m2tcor5_11.f line 524
266 dr = ( - 15.642064 + 0.036381841*mt + TMath::Sqrt(sht)*(2.30111 - 0.013429*mt)
267 + sht*(0.0180877*mt - 9.95272) + ht*(5.68703 - 0.0156807*mt)
268 + ht*sht*(0.00536879*mt - 1.64687) + ht*ht*(0.185188 - 0.000646014*mt)
269 + 8.0/9.0*zt*(4.0 - 26.0*c2 - 5.0*c2*c2)*TMath::Log(mz2/mu2) );
270 }

```

```

503 else
504 c
505 if(ht.lt.4.d0) then
506 c
507 c
508 c INTERPOLATION
509 c
510 tobf2lew =
511 1 -15.642064d0 + 0.036381841d0*mt + dsqrt(sht)*(2.30111d0-
512 1 0.013429d0*mt) + sht*(0.0180877d0*mt -9.95272d0) +
513 2 ht*(5.68703d0 - 0.0156807d0*mt) +
514 3 ht*sht*(0.00536879d0*mt - 1.64687d0) +ht*ht*(0.185188d0 -
515 4 0.000646014*mt) +
516 5 8.d0/9.d0*zt*(4.d0 - 26.d0*c2-5.d0*c2*c2) *dlog(mz2/mu2)
517
518 else

```

```

272 else {
273 // heavy Higgs case
274 // see eq. (10b) of hep-ph/0603374 and (A6) of hep-ph/9905472
275 dr = ( ( 10800.0*c2*GMath::IPow( ht, 3 ) - 4428.0*c2*GMath::IPow( ht, 4 )
276 + 432.0*c2*GMath::IPow( ht, 5 ) - 432.0*c2*GMath::IPow( ht*pi, 2 )
277 + 324.0*c2*GMath::IPow( ht, 3 )*pi*pi
278 - 54.0*c2*GMath::IPow( ht, 4 )*pi*pi)/(108.0*c2*(4.0 - ht)*GMath::IPow( ht, 3 ) )
279 + (( - 4.0 + ht)*TMath::Sqrt(ht)*ZMath::gdeg(ht))/2.0
280 + ((54.0*ht*ht - 135.0*GMath::IPow( ht, 3 ) + 108.0*GMath::IPow( ht, 4 )
281 - 27.0*GMath::IPow( ht, 5 ) ) * TMath::DiLog(1.0 - ht))/(9.0*GMath::IPow( ht, 3 ) )
282 + (( - 5184.0*GMath::IPow( ht, 3 ) - 2592.0*GMath::IPow( ht, 4 )
283 + 2700*GMath::IPow( ht, 5 ) - 540*GMath::IPow( ht, 6 ) + 27.0*GMath::IPow( ht, 7 ) )
284 *TMath::Log(ht))/(54.0*GMath::IPow( ( -4.0 + ht), 2 ) *GMath::IPow( ht, 3 ) )
285 + ( - 12960.0*GMath::IPow( ht, 2 ) + 14256.0*GMath::IPow( ht, 3 )
286 - 5994.0*GMath::IPow( ht, 4 ) + 1134.0*GMath::IPow( ht, 5 ) - 81.0*GMath::IPow( ht, 6 ) )
287 *ZMath::Phi(ht/4.0)/(54.0*GMath::IPow( ( -4.0 + ht), 2 ) *ht*ht)
288 + zt*( (3552.0*c2*c2*ht*ht - 144.0*GMath::IPow( ht, 3 ) + 10772.0*c2*GMath::IPow( ht, 3 )
289 + 6112.0*c2*c2*GMath::IPow( ht, 3 ) - 7392.0*GMath::IPow( c2*ht, 3 )
290 + 36.0*GMath::IPow( ht, 4 ) - 2134.0*c2*GMath::IPow( ht, 4 )
291 - 2742.0*c2*c2*GMath::IPow( ht, 4 ) - 136.0*c2*GMath::IPow( ht, 5 )
292 + 2872.0*GMath::IPow( c2, 3 ) *GMath::IPow( ht, 4 ) + 248.0*c2*c2*GMath::IPow( ht, 5 )
293 - 256.0*GMath::IPow( c2, 3 ) *GMath::IPow( ht, 5 ) - 592.0*c2*c2*ht*pi*pi
294 + 244.0*c2*c2*GMath::IPow( ht, 2 ) *pi*pi - 1904.0*c2*GMath::IPow( ht, 3 ) *pi*pi
295 + 872.0*c2*c2*GMath::IPow( ht, 3 ) *pi*pi + 476.0*c2*GMath::IPow( ht, 4 ) *pi*pi
296 - 224.0*c2*c2*GMath::IPow( ht, 4 ) *pi*pi)/(108.0*c2*(4.0 - ht)*GMath::IPow( ht, 3 ) )
297 - 2.0*(1.0 + 18.0*c2 - 16.0*c2*c2)*ZMath::B0(mz2,c2*mz2,c2*mz2,mu2)/3.0
298 + (1.0 + 20.0*c2 - 24.0*c2*c2)*ZMath::B0(c2*mz2,c2*mz2,mz2,mu2)/(3.0*c2)
299 + ( -34.0 + 116.0*c2 - 64.0*c2*c2)*( -4.0 + ht)*TMath::Sqrt(ht)*ZMath::gdeg(ht)/54
300 + 40.0*( -s2)*(4.0 - ht)*ZMath::Lambda( -1.0 + 4.0/ht)/(3.0*ht)
301 + (74.0*c2*ht - 12.0*c2*ht*ht - 24.0*c2*GMath::IPow( ht, 3 )
302 - 44.0*c2*GMath::IPow( ht, 4 ) + 18.0*c2*GMath::IPow( ht, 5 ) )
303 *TMath::DiLog(1.0 - ht)/(9.0*GMath::IPow( ht, 3 ) )
304 + ( -3.0 - 42.0*c2 - 48.0*c2*c2)*TMath::Log(c2)/(9.0)
305 + (23040.0*ht*ht - 30144.0*c2*ht*ht
306 + GMath::IPow( ht, 3 )*( - 10884.0 + 17856.0*c2 + 1536.0*c2*c2)
307 + GMath::IPow( ht, 4 )*( 5094.0 - 14124.0*c2 + 6528.0*c2*c2)
308 + GMath::IPow( ht, 5 )*( - 2302.0 + 7094.0*c2 - 4288.0*c2*c2)
309 + GMath::IPow( ht, 6 )*( 476.0 - 1516.0*c2 + 896.0*c2*c2)
310 + GMath::IPow( ht, 7 )*( -34.0 + 116.0*c2 - 64.0*c2*c2) )
311 *TMath::Log(ht)/(54.0*GMath::IPow( ( -4.0 + ht), 2 ) *GMath::IPow( ht, 3 ) )
312 - (3.0 + 122.0*c2 - 388.0*c2*c2 + 56.0*GMath::IPow( c2, 3 ) )
313 *TMath::Log(zt) - TMath::Log(mz2/mu2))/(9.0*c2)
314 + (3.0 + 5.0*c2 - 26.0*c2*c2 - 48.0*GMath::IPow( c2, 3 ) ) *TMath::Log(zt)/(9.0*c2)
315 + (23040.0 - 23040.0*c2 - 25860.0*ht + 25344.0*c2*ht + 1536.0*c2*c2*ht
316 + ht*ht*(10236.0 - 7872.0*c2 - 1920*c2*c2)
317 + GMath::IPow( ht, 3 )*( -1890.0 - 2856.0*c2 + 384.0*c2*c2)
318 + GMath::IPow( ht, 4 )*( 144.0 + 2724.0*c2)
319 - 672.0*c2*GMath::IPow( ht, 5 )
320 + 54.0*c2*GMath::IPow( ht, 6 ) )
321 *ZMath::Phi(ht/4.0)/(54.0*GMath::IPow( ( -4.0 + ht), 2 ) *ht*ht)
322 + 4.0*zt*(1.0 - TMath::Log(ht) + TMath::Log(zt) - TMath::Log(mz2/mu2)) );
323 }

```

```

518 else
519
520 c HEAVY HIGGS CASE
521 c
522 tobf2lew =
523 $ (10800.0*c2*ht**3.d0-4428.d0*c2*ht**4.d0+432.d0*c2*ht**5.d0 -
524 1 432.d0*c2*ht**2.d0*pi**2.d0 + 324.d0*c2*ht**3.d0*pi*pi -
525 2 54.d0*c2*ht**4.d0*pi*pi)/(108.d0*c2*(4.d0 - ht)*ht**3.d0) +
526 3 ((-4.d0 + ht)*dsqrt(ht)*gdeg(ht))/2.d0 +
527 4 ((54.d0*ht*ht-135.d0*ht**3.d0+108.d0*ht**4.d0-27.d0*ht**5.d0)*
528 5 ddilog(1.d0 - ht))/(9.d0*ht**3.d0) + ((-5184.d0*ht**3.d0 -
529 6 2592.d0*ht**4.d0+2700*ht**5.d0-540*ht**6.d0 + 27.d0*ht**7.d0)*
530 7 dlog(ht))/(54.d0*(-4.d0 + ht)**2.d0*ht**3.d0) +
531 8 (-12960*ht**2.d0 + 14256.d0*ht**3.d0 - 5994.d0*ht**4.d0 +
532 9 1134.d0*ht**5.d0-81.d0*ht**6.d0)*phi(ht/4.d0)/
533 $ (54.d0*(-4.d0 + ht)**2.d0*ht*ht) +
534 1 zt*((3552.d0*c2*c2*ht*ht-144.d0*ht**3.d0+10772.d0*c2*ht**3.d0+
535 2 6112.d0*c2*c2*ht**3.d0-7392.d0*c2**3.d0*ht**3.d0+36.d0*ht**4.d0-
536 3 2134.d0*c2*ht**4.d0 - 2742.d0*c2*c2*ht**4.d0-136.d0*c2*ht**5.d0+
537 4 2872.d0*c2**3.d0*ht**4.d0+248.d0*c2*c2*ht**5.d0 -
538 5 256.d0*c2**3.d0*ht**5.d0 - 592.d0*c2*c2*ht*pi*pi +
539 6 244.d0*c2*c2*ht**2.d0*pi*pi - 1904.d0*c2*ht**3.d0*pi*pi +
540 7 872.d0*c2*c2*ht**3.d0*pi*pi + 476.d0*c2*ht**4.d0*pi*pi -
541 8 224.d0*c2*c2*ht**4.d0*pi*pi)/(108.d0*c2*(4.d0 - ht)*ht**3.d0) -
542 9 2.d0*(1.d0+18.d0*c2-16.d0*c2*c2)*B0(mz2,c2*mz2,c2*mz2,mu2)/3.d0+
543 $ (1.d0+20.d0*c2-24.d0*c2*c2)*B0(c2*mz2,c2*mz2,mz2,mu2)/(3.d0*c2)+
544 1 (-34.d0+116.d0*c2-64.d0*c2*c2)*(-4.d0+ht)*dsqrt(ht)*gdeg(ht)/54+
545 2 40*(-s2)*(4.d0 - ht)*Ln(-1.d0 + 4.d0/ht)/(3.d0*ht) +
546 3 (74.d0*c2*ht-12.d0*c2*ht*ht-24.d0*c2*ht**3.d0-44.d0*c2*ht**4.d0+
547 4 18.d0*c2*ht**5.d0)*ddilog(1.d0 - ht)/(9.d0*ht**3.d0) +
548 5 (-3.d0-42.d0*c2-48.d0*c2*c2)*dlog(c2)/(9.d0) +
549 6 (23040.d0*ht*ht-30144.d0*c2*ht*ht +
550 7 ht**3.d0*(-10884.d0 + 17856.d0*c2 + 1536.d0*c2*c2) +
551 8 ht**4.d0*(5094.d0 - 14124.d0*c2 + 6528.d0*c2*c2) +
552 9 ht**5.d0*(-2302.d0 + 7094.d0*c2 - 4288.d0*c2*c2) +
553 $ ht**6.d0*(476.d0 - 1516.d0*c2 + 896.d0*c2*c2)+
554 1 ht**7.d0*(-34.d0 + 116.d0*c2 - 64.d0*c2*c2))*dlog(ht)/
555 2 (54.d0*(-4.d0 + ht)**2.d0*ht**3.d0) -
556 3 (3.d0+122.d0*c2-388.d0*c2*c2+56.d0*c2**3.d0)*
557 $ (dlog(zt) - dlog(mz2/mu2))/(9.d0*c2) +
558 4 (3.d0+5.d0*c2-26.d0*c2*c2-48.d0*c2**3.d0)*dlog(zt)/(9.d0*c2) +
559 5 (23040.d0-23040.d0*c2-25860.d0*ht+25344.d0*c2*ht +
560 6 1536.d0*c2*c2*ht*ht*(10236.d0-7872.d0*c2 - 1920*c2*c2) +
561 7 ht**3.d0*(-1890.d0 - 2856.d0*c2 + 384.d0*c2*c2) +
562 8 ht**4.d0*(144.d0+2724.d0*c2)-672.d0*c2*ht**5.d0 +
563 9 54.d0*c2*ht**6.d0)*phi(ht/4.d0)/(54.d0*(-4.d0+ht)**2.d0*ht*ht)) +
564 $ 4.d0*zt*(1.d0 - dlog(ht)+ dlog(zt)-dlog(mz2/mu2))
565 c
566 endif
567 endif
568 c
569 return
570 end

```

324 // see also m2tcor5_11.f line 455

325

326 return dr;

327 }

328

329 // This functions computes the additional term to be added

330 // to the Msbar 2 loop expression of

331 // [Aww (mw^2)/mw^2 - Azz(mz^2)/mw^2]

332 // in units Nc*(alfa0/(4.0*pi*s2))^2/(4.0*zt*c2)

333

334 Double_t GSM::EW2Loop::deltarhoOS(const Double_t& zt, const Double_t& s2, const Double_t& mz, const Double_t& mu)

335 {

336 Double_t c2 = 1.0 - s2;

337 // see eq. (8.56) of The Standard Model in the Making

338 // and m2tcor5_11.f line 588

339 Double_t drOS = - (2.0*(18.0 - 181.0*c2 + 166.0*c2*c2 - 216.0*GMath::IPow(c2, 3))/(27.0*c2)

340 + TMath::Sqrt(-1.0 + 4.0*c2)

341 *(2.0*(1.0 + 26.0*c2 + 24.0*c2*c2)

342 *TMath::ATan(1/TMath::Sqrt(-1.0 + 4.0*c2))/3.0

343 - (1.0 + 11.0*c2 + 12.0*c2*c2)*TMath::ATan(TMath::Sqrt(-1.0 + 4.0*c2))/(3.0*c2*c2))

344 + TMath::Log(c2)/(6.0*c2*c2)*(1.0 + 9.0*c2 - 4.0*c2*c2 + 60.0*GMath::IPow(c2, 3))

345 + (-20.0 + 32.0*c2)*TMath::Log(zt)/9.0

346 + TMath::Log(mz*mz/mu/mu)*(4.0/3.0 + 34.0/3.0*s2 - 1.0/c2);

347

348 return drOS;

349 }

350

351 // This subroutine computes the additional term to be added

352 // to the Msbar 2 loop expression of Deltar^w

353 // in units Nc*(alfa0/(4.0*pi*s2))^2/(4.0*zt*c2)

354

355 Double_t GSM::EW2Loop::deltarwOS(const Double_t& /* zt */, const Double_t& s2,

356 const Double_t& /* mz */, const Double_t& /* mu */) {

357 {

358 Double_t c2 = 1.0 - s2;

359 // see eq. (8.56) of The Standard Model in the Making

360 // and m2tcor5_11.f line 619

361 Double_t drwOS = - (1.0/(3.0*c2) + 4.0 - (1.0 + 11.0*c2 + 12.0*c2*c2)/(3.0*c2*c2)

362 *TMath::Sqrt(-1.0 + 4.0*c2)*TMath::ATan(TMath::Sqrt(-1.0 + 4.0*c2))

363 - TMath::Log(c2)*(0.5/s2 + 3.0 - 1.5/c2 - 1.0/(6.0*c2*c2)));

364

365 return drwOS;

Ende Match 3

Match 4

```
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
subroutine ew2tobfOS(zt,tobf2lewOS,s2,mz,mu)
real*8 zt,tobf2lewOS,s2,c2,mz,mu
c2 = 1.d0 - s2
tobf2lewOS = -(
1  2.d0*(18.d0 - 181.d0*c2 + 166.d0*c2*c2 -
2  216.d0*c2**3.d0)/(27.d0*c2) +
3  DSqrt(-1.d0 + 4.d0*c2)*(2.d0*(1.d0 +26.d0*c2+24.d0*c2*c2)*
4  DATan(1/Sqrt(-1.d0 + 4.d0*c2))/3.d0 -
5  (1.d0 + 11.d0*c2 + 12.d0*c2*c2)*
6  DATan(Sqrt(-1.d0 + 4.d0*c2))/(3.d0*c2*c2)) +
7  DLog(c2)/(6.d0*c2*c2)*
8  (1.d0 + 9.d0*c2 - 4.d0*c2*c2 + 60.d0*c2**3.d0) +
9  (-20.d0 + 32.d0*c2)*dLog(zt)/9.d0 +
$  DLog(mz*mz/mu/mu)*(4.d0/3.d0 + 34.d0/3.d0*s2 -1.d0/c2))

return
end
```

Match 5

```
603
604
605
606
607
608
609
610
611
612
subroutine ew2ldeltarwOS(zt,drs2lewOS,s2,mz,mu)
real*8 zt,drs2lewOS,s2,c2,mz,mu
c2 = 1.d0 - s2
drs2lewOS = - (
1  1.d0/(3.d0*c2) + 4.d0 -
2  (1.d0 + 11.d0*c2 + 12.d0*c2*c2)/(3.d0*c2*c2) *
3  DSqrt(-1.d0 + 4.d0*c2)*DATan(Sqrt(-1.d0 + 4.d0*c2)) -
4  DLog(c2)*(0.5d0/s2 +3.d0 - 1.5d0/c2 - 1.d0/(6.d0*c2*c2)))

return
end
```

```

366 }
367
368
369 // This function contains the 2 loop expressions of
370 // - c/s A_{gamma Z}(mz^2/mz^2) and the vertex contribution
371 // in units Nc*(alfa0/(4.0*pi*s2))**2.0/(4*zt*c2)**2.0
372

```

Match 6

```

373 Double_t GSM::EW2Loop::kappa2L( const Double_t& zt, const Double_t& ht, const Double_t& c2, const Double_t& mz, const Double_t& mu )
374 {

```

```

375     Double_t mz2 = mz*mz;
376     Double_t mu2 = mu*mu;
377     Double_t hht = ht/4.0;
378

```

```

379     // eq. (B3) of hep-ph/9905472

```

```

380     Double_t kapcur = zt*( - 8.0*c2*c2*TMath::Log(c2)/3.0
381         + 4.0*(30.0*c2 - 48.0*c2*c2
382             + 135.0*c2*ht - 216.0*c2*c2*ht - 50.0*c2*ht*ht + 80.0*c2*c2*ht*ht
383             + 5.0*c2*GMath::IPow( ht, 3 ) - 8.0*c2*c2*GMath::IPow( ht, 3 ))
384         *TMath::Log(ht)/(27.0*(-4.0 + ht))
385         + (251.0*c2 - 462.0*c2*c2 - 40.0*c2*ht + 64.0*c2*c2*ht
386             + (18.0*c2 + 144.0*c2*c2)*ZMath::B0(mz2,mz2*c2,mz2*c2,mu2)
387             + (20.0*c2 - 32.0*c2*c2)*(ht - 4.0)*TMath::Sqrt(ht)*ZMath::gdeg(ht)
388             - (186.0*c2 - 240.0*c2*c2)*(- TMath::Log(zt) + TMath::Log(mz2/mu2)))/27.0
389         + 8.0*(c2 - 3.0*c2*c2)*TMath::Log(zt)/9.0
390         + 8.0*ZMath::Phi(hht)*(5.0*c2 - 8.0*c2*c2)/(9.0*(-4.0 + ht)*ht)
391         - 8.0*ZMath::Phi(hht)*(5.0*c2 - 8.0*c2*c2)/(9.0*(-4.0 + ht) );

```

```

392
393     return kapcur;
394 }

```

```

620
621     subroutine kappacur21(zt,ht,k2lew,c2,mz,mu)
622     real*8 zt,k2lew,phi,c2,mz,mz2,B0,gdeg,mu,mu2,ht,hht
623     mz2 = mz*mz
624     mu2=mu*mu
625     hht = ht/4.d0
626     k2lew = zt*(
627         1 -8.d0*c2*c2*DLog(c2)/3.d0 + 4.d0*(30.d0*c2-48.d0*c2*c2+
628         2 135.d0*c2*ht - 216.d0*c2*c2*ht - 50.d0*c2*ht*ht +
629         3 80.d0*c2*c2*ht*ht+5.d0*c2*ht**3.d0-8.d0*c2*c2*ht**3.d0)*
630         4 DLog(ht)/(27.d0*(-4.d0 + ht)) +
631         5 (251.d0*c2 - 462.d0*c2*c2-40.d0*c2*ht+64.d0*c2*c2*ht +
632         6 (18.d0*c2 + 144.d0*c2*c2)*B0(mz2,mz2*c2,mz2*c2,mu2)
633         8 +(20.d0*c2 - 32.d0*c2*c2)*(ht-4.d0)*DSQRT(ht)*gdeg(ht)
634         9 -(186.d0*c2-240.d0*c2*c2)*(-DLog(zt)+DLog(mz2/mu2)))/27.d0+
635         1 8.d0*(c2 - 3.d0*c2*c2)*DLog(zt)/9.d0
636         1 +8.d0*phi(hht)*(5.d0*c2-8.d0*c2*c2)/(9.d0*(-4.d0 +ht)*ht)-
637         1 8.d0*phi(hht)*(5.d0*c2- 8.d0*c2*c2)/(9.d0*(-4.d0 + ht))
638         )
639     return
end

```

```

396 // This function contains the additional piece to be
397 // added to the MSbar two - loop contribution to
398 // c/s A_{gamma Z}(mz^2/mz^2 and the vertex part
399 // in units Nc*(alfa0/(4.0*pi*s2))^2.0/(4*zt)

```

Match 7

```

400 // i3qf = T3*ch
401 Double_t GSM::EW2Loop::kappa2LOS( const Double_t& zt, const Double_t& c2w, const Double_t& mz,
402     const Double_t& mu, const Double_t& sman, const Double_t& i3qf)
403 {
404     Double_t s2w = 1.0 - c2w;
405     Double_t mz2 = mz*mz;
406     Double_t mu2 = mu*mu;
407
408     Double_t kapOS = 0;
409
410     // eq. (8.65) of The Standard Model in the Making
411     // and eq. (B8) of hep-ph/9905472
412     // top
413     kapOS = -32.0/3.0*(ZMath::J1(1.0/zt,sman/mz2) + TMath::Log(sman/mu2)/6.0);
414     // light quarks
415     kapOS += 44.0/9.0*( - TMath::Log(sman/mu2) + 5.0/3.0);
416     // leptons
417     kapOS += 4.0*( -TMath::Log(sman/mu2) + 5.0/3.0);
418     // bosons
419     kapOS += -( 4.0*c2w*mz2/sman*TMath::Log(c2w) + 4.0*c2w*mz2/sman*TMath::Log(mz2/mu2)
420         + (3.0*sman/mz2 + 4.0*c2w*mz2/sman)*ZMath::B0(sman,mz2*c2w,mz2*c2w,mu2));
421     // vertex contribution
422     kapOS += ( 2.0*(ZMath::ug(c2w*mz2/sman) + 0.25) + (i3qf - 1.0)
423         *(ZMath::revf(mz2*c2w/sman) + 3.0/2.0/c2w/c2w*ZMath::revf(mz2/sman))
424         - 4.0*s2w/c2w*(i3qf*i3qf - 1.0)*ZMath::revf(mz2/sman)
425         + (5.0 - 16.0*s2w + 8.0*s2w*s2w)/4.0/c2w/c2w*ZMath::revf(mz2/sman)
426         + 2.0*TMath::Log(c2w) + 2.0*TMath::Log(mz2/mu2) );
427
428     return kapOS;
429 }
430 }
431

```

```

655
656
657 1      ug,revf,i3qf
658
659 s2w = 1.d0-c2w
660 mz2 = mz*mz
661 mu2=mu*mu
662 c
663 c/s A_gz
664 c top
665 k21ewOS = -32.d0/3.d0*(j1(1.d0/zt,sman/mz2)+dlog(sman/mu2)/6.d0)
666 c light quarks
667 k21ewOS = k21ewOS + 44.d0/9.d0*(-dlog(sman/mu2)+5.d0/3.d0)
668 c leptons
669 k21ewOS = k21ewOS + 4.d0*(-dlog(sman/mu2)+5.d0/3.d0)
670 c bosons
671 k21ewOS = k21ewOS -( 4.d0*c2w*mz2/sman*dlog(c2w)+
672     4.d0*c2w*mz2/sman*dlog(mz2/mu2) +
673     (3.d0*sman/mz2 + 4.d0*c2w*mz2/sman)*
674     B0(sman,mz2*c2w,mz2*c2w,mu2) )
675 c vertex contribution
676 k21ewOS = k21ewOS +2.d0*(ug(c2w*mz2/sman)+0.25d0) +
677 $ (i3qf - 1.d0)*(revf(mz2*c2w/sman) +
678 $ 3.d0/2.d0/c2w/c2w*revf(mz2/sman)) -
679 $ 4.d0*s2w/c2w*(i3qf*i3qf-1.d0)*revf(mz2/sman) +
680 1 (5.d0 - 16.d0*s2w + 8.d0*s2w*s2w)/4.d0/c2w/c2w*revf(mz2/sman)
681 2 +2.d0*dlog(c2w) +2.d0*dlog(mz2/mu2)
682 return

```

```

433 // This subroutine contains the 2 loop expressions of
434 // Re [Azz (q^2) - Azz(mz^2)]/(q^2 - mz^2) |_mz^2
435 // in units Nc*(alfa0/(4.0*pi*s2))**2.0/(zt*c2)

```

Match 8

```

436
437 Double_t GSM::EW2Loop::deltaeta( const Double_t& zt, const Double_t& ht, const Double_t& c2, const Double_t& mz, const Double_t& mu)
438 {
439     Double_t mz2 = mz*mz;
440     Double_t mu2 = mu*mu;
441     Double_t pi = TMath::Pi();
442
443     Double_t deta = 0;
444
445     if(TMath::Sqrt(ht) < 0.57) {
446         // light Higgs case
447         // eq. (B1) of hep-ph/9905472
448         deta = ( (( -17.0 + 40.0*c2 - 32.0*(c2*c2))*( -4.0 + 2.0*TMath::Sqrt(ht)*pi))/(108.0*c2)
449             + (( -24.0 + 96.0*c2)*ht*ht*ht + (144.0 - 576.0*c2)*ht*ht*zt
450             + ( -313.0 + 1345.0*c2 + 349.0*c2*c2 - 292.0*GMath::IPow( c2, 3 ))*ht*zt*zt
451             + (196.0 - 1156.0*c2 - 1396.0*c2*c2 + 1168.0*GMath::IPow( c2, 3 ))*zt*zt*zt)
452             /(216.0*c2*(-1.0 + 4.0*c2)*(ht - 4.0*zt)*zt*zt)
453             + (( -2.0*ht*ht*ht + 13.0*ht*ht*zt - 32.0*ht*zt*zt + 36.0*zt*zt*zt)
454             *ZMath::B0(mz2, ht*mz2/zt, mz2, mu2))/(18.0*c2*(ht - 4.0*zt)*zt*zt)
455             + (( -1.0 + 4.0*c2 - 44.0*c2*c2 + 32.0*GMath::IPow( c2, 3 ))
456             *ZMath::B0(mz2, c2*mz2, c2*mz2,mu2))/( -24.0*c2 + 96.0*(c2*c2))
457             + (( -1.0 - 18.0*c2 + 16.0*c2*c2)*TMath::Log(c2))/(2.0*(-6.0 + 24.0*c2))
458             + (( -2.0*ht*ht*ht + 11.0*ht*ht*zt - 24.0*ht*zt*zt + 24.0*zt*zt*zt)
459             *TMath::Log(ht))/(18.0*c2*(ht - 4.0*zt)*zt*zt)
460             + (((8.0 - 32.0*c2)*ht*ht*ht - 52.0*ht*ht*zt + 208.0*c2*ht*ht*zt + 159.0*ht*zt*zt
461             - 704.0*c2*ht*zt*zt + 192.0*c2*c2*ht*zt*zt - 112.0*GMath::IPow( c2, 3 ))*ht*zt*zt
462             + ( -268.0 + 1344.0*c2 - 768.0*(c2*c2) + 448.0*GMath::IPow( c2, 3 ))*zt*zt*zt)
463             *(TMath::Log(1.0/zt) + TMath::Log(mz2/mu2)))
464             /(72.0*c2*(-1.0 + 4.0*c2)*(ht - 4.0*zt)*zt*zt)
465             + (( -4.0*ht*ht + 16.0*c2*ht*ht + 20.0*ht*zt - 79.0*c2*ht*zt - 70.0*c2*c2*ht*zt
466             + 48.0*GMath::IPow( c2, 3 ))*ht*zt - 40.0*zt*zt + 156.0*c2*zt*zt + 280.0*c2*c2*zt*zt
467             - 192.0*GMath::IPow( c2, 3 ))*zt*zt)*TMath::Log(zt))/(36.0*c2*(-1.0 + 4.0*c2)*(ht - 4.0*zt)*zt) );
468     }

```

```

695  subroutine ew2leta(zt,ht,eta2lew,c2,mz,mu)
696     real*8 zt,ht,eta2lew,c2,mz,ln,mz2,B0,gdeg,phi,mu,mu2
697     double precision gmu,repiggh0,mz0,alfa0,alfastr,delQCD,
698     1 pi,mel,mmu,mtau,mb,gamz,Nc,zeta3
699     c
700     common/datcom/gmu,repiggh0,mz0,alfa0,alfastr,delQCD,
701     1 pi,mel,mmu,mtau,mb,gamz,Nc,zeta3
702     mz2 = mz*mz
703     mu2 = mu*mu
704
705  if(dsqrt(ht).lt.0.57d0) then
706     c LIGHT HIGGS CASE
707
708     eta2lew =
709     - ((-17.d0 + 40.d0*c2 - 32.d0*(c2*c2))*
710     - (-4.d0 + 2.d0*Sqrt(ht)*Pi))/(108.d0*c2) +
711     - ((-24.d0 + 96.d0*c2)*ht*ht*ht + (144.d0 - 576.d0*c2)*ht*ht*zt +
712     - (-313.d0 +1345.d0*c2 + 349.d0*c2*c2 - 292.d0*c2**3)*ht*zt*zt +
713     - (196.d0 - 1156.d0*c2 - 1396.d0*c2*c2 + 1168.d0*c2**3)*zt*zt*zt)/
714     - (216.d0*c2*(-1.d0 + 4.d0*c2)*(ht - 4.d0*zt)*zt*zt) +
715     - ((-2.d0*ht*ht*ht + 13.d0*ht*ht*zt - 32.d0*ht*zt*zt +
716     - 36.d0*zt*zt*zt)*B0(mz2, ht*mz2/zt, mz2, mu2))/
717     - (18.d0*c2*(ht - 4.d0*zt)*zt*zt) +
718     - ((-1.d0 + 4.d0*c2 - 44.d0*c2*c2 + 32.d0*c2**3)*
719     - B0(mz2, c2*mz2, c2*mz2,mu2))/(-24.d0*c2 + 96.d0*(c2*c2)) +
720     - ((-1.d0-18.d0*c2+16.d0*c2*c2)*DLog(c2))/(2.d0*(-6.d0+24.d0*c2)))+
721     - ((-2.d0*ht*ht*ht + 11.d0*ht*ht*zt - 24.d0*ht*zt*zt +
722     - 24.d0*zt*zt*zt)*DLog(ht))/(18.d0*c2*(ht - 4.d0*zt)*zt*zt) +
723     - (((8.d0-32.d0*c2)*ht*ht*ht -52.d0*ht*ht*zt +208.d0*c2*ht*ht*zt +
724     - 159.d0*ht*zt*zt - 704.d0*c2*ht*zt*zt + 192.d0*c2*c2*ht*zt*zt -
725     - 112.d0*c2**3*ht*zt*zt + (-268.d0 + 1344.d0*c2 -
726     - 768.d0*(c2*c2) + 448.d0*c2**3)*zt*zt*zt)*
727     - (DLog(1.d0/zt) +DLog(mz2/mu2)))/
728     - (72.d0*c2*(-1.d0 + 4.d0*c2)*(ht - 4.d0*zt)*zt*zt) +
729     - ((-4.d0*ht*ht + 16.d0*c2*ht*ht + 20.d0*ht*zt - 79.d0*c2*ht*zt -
730     - 70.d0*c2*c2*ht*zt+48.d0*c2**3*ht*zt-40.d0*zt*zt+156.d0*c2*zt*zt+
731     - 280.d0*c2*c2*zt*zt - 192.d0*c2**3*zt*zt)*DLog(zt))/
732     - (36.d0*c2*(-1.d0 + 4.d0*c2)*(ht - 4.d0*zt)*zt)
733     c
734     else

```

469

```

470 else {
471 // heavy Higgs case
472 // eq. (B2) of hep-ph/9905472
473 deta = ( (- 1152.0 + 4608.0*c2 + 50.0*ht + 2248.0*c2*ht + 976.0*c2*c2*ht
474 - 1600.0*GMath::IPow( c2, 3 ) *ht + 67.0*ht*ht - 880.0*c2*ht*ht
475 - 244.0*c2*c2*ht*ht + 400.0*GMath::IPow( c2, 3 ) *ht*ht)/(864.0*c2*(1.0 - 4.0*c2)*(-4.0 + ht)*ht)
476 + ((- 1.0 + 4.0*c2 - 44.0*c2*c2 + 32.0*GMath::IPow( c2, 3 ))
477 *ZMath::B0(mz2,c2*mz2,c2*mz2,mu2))/( -24.0*c2 + 96.0*(c2*c2))
478 + (4.0/3.0 - 4.0/ht - (1.0 - 4.0/ht)*ZMath::Lambda( -1.0 + 4.0/ht)/(12.0*c2)
479 + ((- 1.0 - 18.0*c2 + 16.0*c2*c2)*TMath::Log(c2))/(2.0*(-6.0 + 24.0*c2))
480 + ((- 384.0 - 202.0*ht + 320.0*c2*ht - 256.0*c2*c2*ht + 55.0*ht*ht + 80.0*c2*ht*ht
481 - 64.0*c2*c2*ht*ht + 3.0*ht*ht*ht - 40.0*c2*ht*ht*ht + 32.0*(c2*c2)*ht*ht*ht)
482 *TMath::Log(ht))/(144.0*c2*GMath::IPow( (-4.0 + ht), 2 ) *ht)
483 + ((- 17.0 + 40.0*c2 - 32.0*(c2*c2))
484 *(- 4.0 + ht/2.0 + (1.0 - ht/4.0)*TMath::Sqrt(ht)*ZMath::gdeg(ht)
485 + ((6.0 - ht)*ht*TMath::Log(ht)/4.0))/(108.0*c2)
486 + ((- 31.0 + 192.0*c2 - 192.0*(c2*c2) + 112.0*GMath::IPow( c2, 3 ))
487 *(TMath::Log(1.0/zt) + TMath::Log(mz2/mu2)))/(72.0*c2 - 288.0*(c2*c2))
488 + (( 2.0 - 7.0*c2 - 70.0*c2*c2 + 48.0*GMath::IPow( c2, 3 ))*TMath::Log(zt))/( -36.0*c2 + 144.0*(c2*c2))
489 + ((- 384.0 - 10.0*ht + 320.0*c2*ht - 256.0*(c2*c2)*ht + 238.0*ht*ht - 400.0*c2*ht*ht
490 + 320.0*c2*c2*ht*ht - 63.0*ht*ht*ht + 80.0*c2*ht*ht*ht - 64.0*(c2*c2)*ht*ht*ht
491 + 3.0*GMath::IPow( ht, 4 ))*ZMath::Phi(ht/4.0))/(144.0*c2*GMath::IPow( (-4.0 + ht), 2 ) *ht*ht));
492 }
493 // Information to this value also in
494 // The Standad Model in the Making page 432
495
496 return deta;
497 }
498

```

Ende Match 8

```

734 else
735 c
736 c HEAVY HIGGS CASE
737 c
738 eta2lew =
739 - (-1152.d0 + 4608.d0*c2 +50.d0*ht+2248.d0*c2*ht+976.d0*c2*c2*ht -
740 - 1600.d0*c2**3*ht+67.d0*ht*ht-880.d0*c2*ht*ht-244.d0*c2*c2*ht*ht+
741 - 400.d0*c2**3*ht*ht)/(864.d0*c2*(1.d0 -4.d0*c2)*(-4.d0 + ht)*ht)+
742 - ((-1.d0 + 4.d0*c2 - 44.d0*c2*c2 + 32.d0*c2**3)*
743 - B0(mz2,c2*mz2,c2*mz2,mu2))/(-24.d0*c2 + 96.d0*(c2*c2)) +
744 - (4.d0/3.d0 - 4.d0/ht -
745 - (1.d0 -4.d0/ht)*Ln(-1.d0 + 4.d0/ht))/(12.d0*c2) +
746 - ((-1.d0-18.d0*c2+16.d0*c2*c2)*DLog(c2))/(2.d0*(-6.d0+24.d0*c2))+
747 - ((-384.d0 - 202.d0*ht + 320.d0*c2*ht - 256.d0*c2*c2*ht +
748 - 55.d0*ht*ht + 80.d0*c2*ht*ht - 64.d0*c2*c2*ht*ht +3.d0*ht*ht*ht-
749 - 40.d0*c2*ht*ht*ht + 32.d0*(c2*c2)*ht*ht*ht)*DLog(ht))/
750 - (144.d0*c2*(-4.d0 + ht)**2*ht) +
751 - ((-17.d0 + 40.d0*c2 - 32.d0*(c2*c2))*(-4.d0 + ht/2.d0 +
752 - (1.d0 - ht/4.d0)*Sqrt(ht)*gdeg(ht) +
753 - ((6.d0 - ht)*ht*Log(ht))/4.d0))/(108.d0*c2) +
754 - ((-31.d0 + 192.d0*c2 - 192.d0*(c2*c2) + 112.d0*c2**3)*
755 - (DLog(1.d0/zt) + DLog(mz2/mu2)))/(72.d0*c2 - 288.d0*(c2*c2)) +
756 - ((2.d0 - 7.d0*c2 - 70.d0*c2*c2 + 48.d0*c2**3)*DLog(zt))/
757 - (-36.d0*c2 + 144.d0*(c2*c2))+
758 - ((-384.d0 - 10.d0*ht + 320.d0*c2*ht - 256.d0*(c2*c2)*ht +
759 - 238.d0*ht*ht - 400.d0*c2*ht*ht + 320.d0*c2*c2*ht*ht -
760 - 63.d0*ht*ht*ht + 80.d0*c2*ht*ht*ht -
761 - 64.d0*(c2*c2)*ht*ht*ht + 3.d0*ht**4)*phi(ht/4.d0))/
762 - (144.d0*c2*(-4.d0 + ht)**2*ht*ht)
763 endif
764 c
765 return
766 end

```

```

500 // This subroutine contains the additional piece to be
501 // added to the MSbar two - loop contribution to
502 // Re [Azz (q^2) - Azz(mz^2)]/(q^2 - mz^2) |_mz^2
503 // in units Nc*(alfa0/(4.0*pi*s2))^2/(4.0*zt*c2)

```

Match 9

```

505 Double_t GSM::EW2Loop::deltaetaOS( const Double_t& zt, const Double_t& ht, const Double_t& s2,
506                                     const Double_t& mz, const Double_t& mu, const Double_t& sman, const Double_t& i3qf)

```

```

507 {
508     Double_t mz2 = mz*mz;
509     Double_t mu2 = mu*mu;
510     Double_t zs2 = 1.0/zt;
511     Double_t c2 = 1.0 - s2;
512     Double_t detaOS = 0;

```

```

514 // Information to this value in hep-ph/9905472 partulary eq. (B6)
515 // and in m2cor5_11.f line 484

```

```

517 // Azz
518 detaOS = ( (197.0 - 1378.0*c2 + 1064.0*c2*c2)/(27.0*(-1.0 + 4.0*c2))
519            + (1.0 + 16.0*c2 - 20.0*(c2*c2) + 48.0*GMath::IPow( c2, 3 ))
520            *ZMath::B0(mz2, c2*mz2, c2*mz2, mu2)/(-3.0 + 12.0*c2)
521            + (2.0*c2*(1.0 + 26.0*c2 + 24.0*(c2*c2))*TMath::Log(c2))/(-3.0 + 12.0*c2)
522            + (2.0*(-20.0 + 113.0*c2 - 102.0*(c2*c2) + 24.0*GMath::IPow( c2,3 ))
523            *(TMath::Log(zs2) + TMath::Log(mz2/mu2)))/(-3.0 + 12.0*c2)
524            + (2.0*(-50.0 + 283.0*c2 - 242.0*c2*c2 + 72.0*GMath::IPow( c2, 3 ))
525            *TMath::Log(zt))/(-9.0 + 36.0*c2) );

```

```

527 // Vertex contribution
528 detaOS += ( 4.0*c2*(TMath::Log(c2) + TMath::Log(mz2/mu2)) + 4.0*c2*(ZMath::ug(mz2*c2/sman) + 0.25)
529            - 2.0*c2*(1.0 - i3qf)*ZMath::revf(mz2*c2/sman)
530            - (1.0 - 6.0*i3qf + 12.0*(1.0 - c2*c2)*i3qf*i3qf)/(2.0*c2)
531            *ZMath::revf(mz2/sman) );

```

```

775 subroutine ew2letaOS(zt,ht,eta2lewOS,s2,mz,mu,sman,i3qf)
776 real*8 zt,eta2lewOS,c2,mz,mz2,B0,mu,mu2,sman,ug,revf,i3qf,
777         zs2,di,ln,hbar,j1,a1,a2,ht,s2,vertex
778 c
779     mz2 = mz*mz
780     mu2=mu*mu
781     zs2 =1.d0/zt
782     c2 = 1.d0 - s2
783 c
784     eta2lewOS =
785     - (197.d0 - 1378.d0*c2 + 1064.d0*c2*c2)/(27.d0*(-1.d0 +4.d0*c2))+
786     - (1.d0 + 16.d0*c2 - 20.d0*(c2*c2) + 48.d0*c2**3)*
787     - B0(mz2, c2*mz2, c2*mz2, mu2)/(-3.d0 + 12.d0*c2) +
788     - (2.d0*c2*(1.d0 + 26.d0*c2 + 24.d0*(c2*c2))*DLog(c2))/
789     - (-3.d0 + 12.d0*c2) +
790     - (2.d0*(-20.d0 + 113.d0*c2 - 102.d0*(c2*c2) + 24.d0*c2**3)*
791     - (Dlog(zs2) + DLog(mz2/mu2)))/(-3.d0 + 12.d0*c2) +
792     - (2.d0*(-50.d0 + 283.d0*c2 - 242.d0*c2*c2 + 72.d0*c2**3)*
793     - DLog(zt))/(-9.d0 + 36.d0*c2)
794

```

```

794 c
795     Vertex contribution
796     vertex =
797     - 4.d0*c2*(DLog( c2) + DLog( mz2/mu2)) +4.d0*c2*(ug(mz2*c2/sman)+
798     - 0.25d0) - 2.d0*c2*(1.d0 - i3qf)*revf(mz2*c2/sman) -
799     - (1.d0 - 6.d0*I3qf + 12.d0*(1.d0-c2*c2)*i3qf*i3qf)/(2.d0*c2)*
800     - revf(mz2/sman)

```

533

```

534 // Contribution of eta due to overall shift
535
536 // hadronic contribution
537 Double_t di = 4.0*zs2 - 1.0;
538 detaOS += ( (6.0*(zs2 - 4.0*zs2*zs2/di*ZMath::Lambda(di) + 1.0/6.0 + ZMath::J1(zs2,1.0))
539             *(16.0*s2*s2/9.0 - 4.0*s2/3.0 + 0.5) - 1.5*zs2
540             + 6.0*zs2*zs2/di*ZMath::Lambda(di) - 2.0/3.0*(2.5 - 14.0/3.0*s2 + 44.0*s2*s2/9.0)
541             + 3.0*(20.0/9.0*s2*s2 - 2.0*s2 + 1.0)*TMath::Log(mz2/mu2))/c2 );
542
543 // leptonic contribution
544 detaOS += (1.0 - 2.0*s2 + 4.0*s2*s2)/c2*(TMath::Log(mz2/mu2) - 2.0/3.0);
545
546 // bosonic contribution
547 Double_t a2 = TMath::Sqrt(4.0*c2 - 1.0);
548 a2 = a2*TMath::ATan(1.0/a2);
549 a2 = a2*( 1.0/6.0/c2 - 2.0/3.0 + c2/c2/3.0 - 4.0/3.0*c2
550          + 6.0*c2*c2 - 6.0*c2 + c2/c2*(18.0*c2*c2 - 8.0*c2*GMath::IPow( s2, 2 ))/(4.0*c2 - 1.0) );
551
552 a2 += ZMath::hbar(ht/zt)/2.0/c2;
553 a2 += ( TMath::Log(c2)*(1.0/12.0/c2 - 1.0/3.0 - 3.0*c2) - 4.0*c2*c2
554        + 108.0/36.0*c2 - 8.0*c2/3.0 + 5.0/9.0 - 5.0/36.0/c2
555        + 5.0/3.0 - 1.0/6.0*(19.0*c2 - s2*s2/c2)*TMath::Log(mz2/mu2) );
556
557 detaOS += a2;
558
559 return detaOS;
560 }

```

Ende Match 9

```

802 c Contribution of eta due to overall shift
803 c
804 c hadronic contribution
805 c di=4.0*zs2-1.0
806 c a1=(6.0*(zs2-4.0*zs2*zs2/di*ln(di)+1.0/6.0+j1(zs2,1.0))*
807 1 (16.0*s2*s2/9.0-4.0*s2/3.0+0.5d0)-1.5d0*zs2 +
808 1 6.0*zs2*zs2/di*ln(di) -
809 2 2.0/3.0*(2.5d0-14.0/3.0*s2+44.0*s2*s2/9.0) +
810 3 3.0*(20.0/9.0*s2*s2 - 2.0*s2 + 1.0)*Dlog(mz2/mu2))/c2
811 c leptonic contribution
812 c a1= a1 +(1.0-2.0*s2+4.0*s2*s2)/c2*(Dlog(mz2/mu2)-2.0/3.0)
813 c bosonic contribution
814 c a2=dsqrt(4.0*c2-1.0)
815 c a2=a2*atan(1.0/a2)
816 c a2=a2*(1.0/6.0/c2-2.0/3.0+c2/c2/3.0-4.0/3.0*c2
817 1 +6.0*c2*c2-6.0*c2+c2/c2*(18.0*c2**2-8.0*c2*s2**2)
818 2 /(4.0*c2-1.0))
819 c
820 c a2=a2+hbar(ht/zt)/2.0/c2
821 c a2=a2+dlog(c2)*(1.0/12.0/c2-1.0/3.0-3.0*c2)-4.0*c2*c2
822 1 +108.0/36.0*c2-8.0*c2/3.0+5.0/9.0-5.0/36.0/c2
823 2 +5.0/3.0 -1.0/6.0*(19.0*c2 - s2*s2/c2)*dlog(mz2/mu2)
824 c eta2lewOS = eta2lewOS + vertex + a1 + a2
825 c
826 c return
827 c end

```