QQ Quick Reference for BELLE

How to use CLEO's QQ event generator for BELLE

edited for BELLE by

Ryosuke Itoh

Physics Division, KEK

itoh@kekvax.kek.jp

based on documents written by

Peter C. Kim

Cornell University

April, 1995

DISTRIBUTION OF THIS DOCUMENT IS LIMITED WITHIN BELLE COLLABORATION ONLY.

Contents

1 Introduction

This manual describes how to use QQ event generator in BELANL. QQ is the event generator developed by P.Avery, R.A.Fulton, P.C.Kim, J.L.Rodriguez et al. for CLEO-II experiment and their latest experimental results are installed in the generator. This generator is considered to be the best for the physics study at B factory.

The most significant difference of QQ from our standard BBGEN generator is that any types of B decay can be programmed by yourself by supplying your own decay table. The table can be written in a very simple language with various informations of the decay like helicity state, decay angular distribution, etc. The CP violation can be treated in a generalized way in QQ.

I implemented QQ event generator under BELANL for the MC study in BELLE. Here I will try to summarize how to use it. Since there is no official manual of QQ, my knowledge about QQ is limited. Most of the informations in this document were grabbed from the scattered documents in CLEO archive, especially from CLEO's MC NEWS written by P.C.Kim of Cornell. Therefore please keep in mind that there may be some mistakes or shortcomings in this reference.

NOTE THAT DISTRIBUTION OF THIS DOCUMENT IS LIMITED WITHIN BELLE COLLABORATION ONLY.

2 Preparing Control File for QQ

To use QQ for your analysis, you need to prepare two files. One is the control file, and the other is the decay table file. This section describes how to prepare your control file.

The control file is read by FFREAD in CERNLIB. So the syntax of this file must be matched with FFREAD format. See CERNLIB manual for the detail of FFREAD.

The default control file can be found in

`/home/bsun03/bfactory/qq/runfil/belle_bb.ctr`

The contents of the file is:

```
C ======================================================================
C    This is an example of generating Y(4S) --> B B= event
C    MODEL = 1
C
C        (Variables are defined as CHARCTER*6 strings.)
C+
C           Modified for BELLE setup by R.Itoh, 19-Apr-1995
```



```
C                   Asymmetric Collision 3.5GeV(e+) + 7.995GeV(e-)
C                   Model = 1 (e+e- -> VPHO -> BB)
C ======================================================================
LIST
C
IRAN    0
C
BMPSQQ   3.5
BMNGQQ   7.995
BWPSQQ   0.005
BWNGQQ   0.005
ENRGMN  10.559
ENRGMX  10.600
C
MODEL   1
IFR     1
IFL     0
IMODE   1
BMRAD   F
FLIP    0.0
C
PAR     6=4.0
PAR     7=0.121
PAR     8=0.0
PAR     9=0.3
PAR    11=1.0
PAR    12=0.7
PAR    18=0.0
C
PUD    0.480
CMLT1  1.8  2.5  2.0  1.8  3.2
CMLT2  0.9  0.8  0.5  0.8  0.8
C
LONGLF TRUE
C
END
```

This file is to be used for the standard B meson decay. In most case, you don't have to change parameters in this control file and can be used for your analysis. But QQ has many features in it and these can be controled by the parameters given in this file.

MODEL is the main control parameter for the event generation. QQ has many kinds of event generators generators built-in and they can be switched by this parameter like BB event via VPHO (MODEL=1), Continuum event using JETSET7.3 (MODEL=11), resonances (MODEL=0,12), Bhabha (MODEL=2), μ pair (MODEL=3), τ pair (MODEL=4 or 7(KORALB)), γ pair (MODEL=5), two photon (MODEL=6) and more. MODEL=1 can be used for user supplied B-decays. The continuum events can be generated using Model 0 or Model 11 which generates $q\overline{q}$ events using JETSET7.3.

Here is a summary of the parameters used in this file.

1. Beam related parameters
    - **BMPSQQ** Real; Positron beam energy (GeV)
    - **BMNGQQ** Real; Electron beam energy (GeV)
    - **BWPSQQ** Real; RMS width for positron beam energy
    - **BWNGQQ** Real; RMS width for electron beam energy
    - **ENRGMN** Real; Minimum beam energy allowed if beam spread ¿ 0
    - **ENRGMX** Real; Maximum beam energy allowed if beam spread ¿ 0
      Ecm of BMPSQQ and BMNGQQ must be in the range ENRGMN ¡ Ecm ¡ ENRGMX. Otherwise you will find yourself in an infinite loop.
    - **BPOSQQ(3)** Real; Beam position in 3-dim (meters) Default=(0,0,0)
    - **BZISQQ(3)** Real; Beam size in 3-dim (meters) Default=(0.0005,0.00003,0.03)
    - **BFLDQQ** Real; Magnetic Field (Tesla) Default=1.5T
2. QQ model parameter
    - **MODEL** Integer; Type of process to generate
        - =0; $q\overline{q}$ or ggg Jets using the old QQ program. $e^+e^- \rightarrow q\overline{q}/ggg \rightarrow hadronization$
        - =1; User specified final states via VPHO. $e^+e^- \rightarrow VPHO \rightarrow B\overline{B}$
        - =2; Bhabha scattering with radiation
        - =3; $\mu\overline{\mu}$ pairs with radiation
        - =4; $\tau\overline{\tau}$ pairs with radiation
        - =5; $\gamma\gamma$ pairs with radiation
        - =6; 2-photon generator
        - =7; KORALB τ pair generator
        - =8; BHLUMI Bhabha generator
        - =10; User defined model. The user must supply the subroutine MODUSR.
        - =11; $q\overline{q}$ generator using LUND/JETSET7.3. $e^+e^- \rightarrow q\overline{q} \rightarrow hadronization$

        - =12; ggg generator using LUND/JETSET7.3. $e^+e^- \rightarrow ggg \rightarrow hadronization$
        - =13; Single particle insertionwith LUND/JETSET7.3. $GOD \rightarrow D{*}^+$
        - =14; Heavy quarkonium production with LUND/JETSET7.3 with radiative tails. $e^+e^- \rightarrow \gamma e^+e^-$; $e^+e^- = \Upsilon(3S)$.
3. Generic switches
    - **LSELCT** Logical; If true, call logical function ISELCT (supplied by user). If the function is false, reject the event.
      Default = FALSE
    - **LWIDE** Logical; If true, use finite widths for resonances.
      Default = TRUE
    - **LPHASE** Logical; If true, Phase space decay allowed.
      Default = TRUE
    - **LSHAPE** Logical; Fill momentum array for SHAPE.
      Default = FALSE
    - **LONGLF** Logical; Decay long-lived particles ($c\tau$ >1cm). If false, they are assumed to be decayed in FSIM or GEANT.
      Default = FALSE
    - **BMRAD** Logical; If true, include initial state radiation. This is applied for MODEL=0,1,6 only. Parameters are given by KEMAX and KEMIN.
      Default = FALSE
    - **LRESPR** Logical; If true, use Breit-Wigner shape for $\Upsilon(4s)$. Parameters are given by RESPRO
      Default = FALSE
4. Generic parameters
    - **IRAN** Integer Random number seed. If 0 is given, seed is set to 12345. (Default = 0)
    - **RESPRO(2)** Real; Breit-Wigner resonance shape parameters for $\Upsilon(4s)$.

      Default = 10.579, 0.024
    - **KEMAX** Real; Max. fraction of ECM for rad. photon.
      Default = 1.0
    - **KEMIN** Real; Min. fraction of ECM for rad. photon.
      Default = 0.01

- **FLIP** Real; Sets the level of baryon production (only for models using old Sjostrand code at some level). 0<FLIP<1.
Default = 0.0
- **PAR(1)** not used.
- **PAR(2)** Real; Polarization for use in polarized e+e- beams.
Default = 0.0
- **PAR(3,4)** Real; Gives optional maximum quark thrust and minimum gluon energy cuts for $q\overline{q}g$ matrix element in Sjostrand program.
- **PAR(5)** Real; $\alpha_{EM} = 1/137 = 0.0073$
- **PAR(6-7)** Real; Number of flavors for calculation of α_s and the value of α_s.
Default = (4, 0.121)
- **PAR(8)** not used
- **PAR(9)** Real; Average transverse meson mass used in various matrix element cuts.
Default = 0.5
- **PAR(10)** Real; Maximum cuts allowed for z range in QQJET (Sjostrand)
Default = 0.9
- **PAR(11-12)** Real; Defines stopping points in $q\overline{q}$ generation for IFR=1,2, respectively.
Default = 0.7, 0.7
- **PAR(13-14)** Real; Probability for "reverse' rapidity ordering of the final two mesons for IFR=1,2, respectively.
- **PAR(15-17)** Real; Decribes parametrization of $q\overline{q}g$ matrix; I(x) = PAR(15) + Par(16)*[ln(x)]**PAR(17)
- **PAR(18)** Real; ??????
- **BSPIN(1-5)** Real; Elements of the array BSPIN specify the proportion of 2pin 1/2 for the case xxx,sxx,ssx,cyy, and ccy, where x is u or d and y is u,d, or s.
Default = (1.0, 1.0, 1.0, 1.0, 1.0)
- **BSYMM(1-3)** Real; Elements of the array BSYMM specify the proportion of the lambda-like state for the spin 1/2 combinations sud, cud, and csx.
Default = (1.0, 1.0, 1.0)
- **PUD** Real; Fraction of $u\overline{u}$ and $d\overline{d}$ from the sea.
Default = 0.45

- **PSPIN** Real; Spin fraction for
  - PSPIN(1) for $u\overline{u}$, $u\overline{d}$, $d\overline{d}$
  - PSPIN(2) for $u\overline{s}$, $d\overline{s}$
  - PSPIN(3) for $u\overline{c}$, $d\overline{c}$, $s\overline{c}$
  - PSPIN(4) for $u\overline{b}$, $d\overline{b}$, $s\overline{b}$, $c\overline{b}$
  - PSPIN(5) for $q\overline{t}$
  - PSPIN(6) for $s\overline{s}$
  - PSPIN(7) for $c\overline{c}$
  - PSPIN(8) for $b\overline{b}$
  - PSPIN(9) for $t\overline{t}$

  Default = 9 * 0.5
- **CMLT1(1), CMLT2(1)** Real; Mult. parameters for the particles containing only u,d,s quarks
- **CMLT1(2), CMLT2(2)** Real; Mult. parameters for the particles containing c quark
- **CMLT1(3), CMLT2(3)** Real; Mult. parameters for the particles containing b quark
- **CMLT1(2), CMLT2(4)** Real; Mult. parameters for the particles containing y quark
- **CMLT1(5), CMLT2(5)** Real; Mult. parameters for virtual photon
Default = CMLT1 = (1.8,2.5,1.95,1.8,3.85), CMLT2 = (0.8, 0.8, 0.8, 0.9, 0.8)
- **WIDTH(1-5)** Real; Width parameters for particles containing:
1: u,d,s quarks only, 2: c quarks, 3: b quarks, 4: t quarks, 5: Virtual photon.
Default = (1.0, 1.0, 0.5, 1.0, 0.5)

5. Parameters for MODEL=0 (Resonanse state)

- **IFR** Integer; Type of jet fragmentation model for MODEL=0.
(=1 Feynman Field model / =2 LUND model (very old version))
- **IFL** Integer; Flavor of jets for MODEL=0.
  - =0; Quarkonium of 3 gluon
  - =1-6; u,d,s,c,b,t jets, respectively
  - =7; Charge weighted mixture of u,d,s,c,b
  - =8; Charge weighted mixture of u,d,s,c,b,t
  - =10; Charge weighted mixture of u,d,s,c
- **IMODE** Integer; Furthur specification of jets for MODEL=0.

  - =0 (IFL=0); 3 Gluon Only
  - =1 (IFL=0); Charge 1/3 quarkonium-¿ggg or ggγ
  - =2 (IFL=0); Charge 2/3 quarkonium-¿ggg or ggγ
  - =0 (IFL¿0); $q\overline{q}$ jets only
  - =1 (IFL¿0); $q\overline{q}$ + $q\overline{q}g$ jets
- **SIGMA** Real; Width of transverse momentum of a jet (Lund fragmentation)
  Default=0.350GeV/c

6. Parameters for MODEL=2

- **ANGMIN** Minimum scattering angle in degrees.
  Default = 5.0
- **ANGMAX** Maximum scattering angle in degrees.
  Default = 175.0

7. Parameters for LUND/JETSET7.3 (MODEL=10-14)

- **NFLAV** Integer; MSTJ(104). Number of flavors.
  Default = 4
- **IQTYPE** Integer; Quark type for $q\overline{q}$ events. If 0, all allowed flavor. If 4, c-quark only, etc.
  Default = 0
- **ISTRAD** Integer; Initial state radiation flag for JETSET 7.3 (applied only to continuum events.) 1=on
  Default = 1
- **QCDCOR** Integer; MSTJ(101). Type of QCD correction used for continuum events. 0: qq only, 1: qq+qqg, 2: qq+qqg+qqgg/qqqq, 5: parton shower.
  Default = 5
- **LFRAG** Integer; MSTJ(11). Choice of longitudinal fragmentation model. 1: LUND symmetric, 2: Independent frag. 3: LUND for light quarks/ Peterson for c and b, 4: LUND symmetric with modification for heavy quarks.
  Default = 1
- **EPSC** Real; PARJ(54). Epsilon value for charm quark used in Peterson func.
  Default = -0.04
- **EPSB** Real; PARJ(55). Epsilon value for beauty quark used in Peterson func.
  Default = -0.004

- **EMINFX** Real; PARJ(33). Minimum energy where the parton fragmentation is stopped when LFRAG=1 is chosen.
  Default = 0.3
- **MSPIN(1-7)** Real; PARJ(11-17). Pseudoscaler/Vector/Tensor ratio for Mesons. See JETSET7.3 manual for detail. Default = (0.5, 0.6, 0.75, 4*0.05)
- **LUNDEC** Integer; Controls the particle decays. If 1, particles are decayed in JETSET package using the LUND decay table. If 0, particles decay according to QQ decay table.
  Default = 0
- **LXEDIT** Integer; REmoval of strings/gluons from the particle list, if 1.
  Default = 1
- **LSINGL** 'string'; Name of the single particle to be generated.
- **ESINGL** Real; Energy of the single particle
- **LTHET** Real; Theta in radian of the single particle
- **LPHI** Real; Phi in radian of the single particle
- **LRESON** 'string'; Name of the narrow heavy resonances you want, $\Upsilon(1S), \Upsilon(2), or \Upsilon(3S)$. Proper radiative correction done for a given Ecm.
- **BSTECM(1-6)** Real; Boost/rotation parmeters for LUROBO.

8. Parameters for KORALB (MODEL=7)

- **KSPIN** Integer; Spin effect in decay, 1=on.
  Default = 1
- **ITFIN** Integer; Type of final state fermion, 1=tau.
  Default = 1
- **KEYGSW** INteger; Implementation level of GSW model.
  Default = 1
- **KEYRAD** Integer; QED radiation allowed, 1=on
  Default = 1
- **ITDKRC** Integer; Bremsstrahlung in decay, 0=off
  Default = 0
- **KORBE1(1-3)** Real; Spin polarization vector for e+
  Default = (0.0, 0.0, 0.0)
- **KORBE2(1-3)** Real; Spin polarization vector for e-
  Default = (0.0, 0.0, 0.0)

- **SINW2** Real; $\sin(\theta_w)^2$
  Default = 0.226
- **KORBGV** Real; g_v
  Default = 1.0
- **KORBGA** Real; g_a
  Default = -1.0
- **KORXK0** Real; Infrared cutoff
  Default = 0.01

9. Parameters for two photon generator (MODEL=6)
   .... to be filled later

3 How to write your own decay table

In order to generate specific B-decay events which you want, you need to write a decay table for your decay. Decay table for the standard decays is already supplied by QQ as the default decay table. You have only to write the decay table for the specific B-decay. Your table partically superseeds the default decay table.

Here is the example to generate the gold plated CP decay, $B \to J/\Psi + K_s$. The other B meson is assumed to decay semileptonicaly via $B \to D^* + l + \nu$. This example can be found in `/home/bsun03/bfactory/qq/runfil/cp_psiks.dec`:

```
!+
! Description : Example of User supplied decay tablle
!                  CP decay : B->J/Psi + Ks
!                 Tag decay : B->D* + l + nu
! Mixing
MIXING B0 B0B 0.66
!
! e+e-->VPHO->B0+B0B
!
! disable the case that both of B/Bbar decay to CP
! Ang_Dist=Par(1)+Par(2)*COS(theta)+Par(3)*COS(the)^2
! matrix, BR, daughters
DECAY VPHO
CPTAG
MATRIX 1.0 0.0 -1.0
CHANNEL 0 1.0 B0 B0B
ENDDECAY
!
! Decay of B0->Psi+K0(CP) / B0->semileptonic(TAG)
!
```



```
DECAY B0
CHANNEL 1 1.0 NUE E+ D*-
CPEIGENSTATE  0.20
HELICITY 1.0 0.0 0.0
CHANNEL 0 1.0 PSI K0S
ENDDECAY
!
! Decay of B0B->Psi+K0(CP) / B0B->semileptonic(TAG)
!
DECAY B0B
CHANNEL 1 1.0 NUEB E- D*+
CPEIGENSTATE  -0.20
HELICITY 1.0 0.0 0.0
CHANNEL 0 1.0 PSI K0S
ENDDECAY
!
! Decay of Psi->mu+mu- with helicity
!
DECAY PSI
ANGULAR_HELICITY 1 1.0 0.0 1.0
ANGULAR_HELICITY 0 1.0 0.0 -1.0
ANGULAR_HELICITY -1 1.0 0.0 1.0
CHANNEL 0 1.0 MU+ MU-
ENDDECAY
!
! Decay of Ks->Pi+Pi-
!
DECAY K0S
CHANNEL 0 1.0 PI+ PI-
ENDDECAY
```

As seen, the decay table can be written in a plain language specifying the CP state, angular distribution of two body decays, etc.

Here is a simple prescription to write your own decay table.

3.1 Decays of Particles

The summary of the syntax to specify a decay channel of a particle is as follows:

```
DECAY              particle_name
CHANNEL            matrix  BR  dtr1 dtr2 dtr3 .....
HELICITY           prob   hel1 hel2 hel3 ......
MATRIX             A B C
```

```
ANGULAR_HELICITY   hel A B C
ENDDECAY
STABLE             particle_name
```

where

- **DECAY** starts the decay sequence for the particle "particle_name"
- **CHANNEL** specifies a single decay channel;
  - matrx: ID number of matrix element. This is ignored for two body decays. MATRIX command is used for two body decays.

```
<0: quark or gluon jet decay
0: Phase space decay
1: semi-leptonic decay by quark fragmentation into hadrons
2: semi-leptonic decay by fragmenting hadronic blob
3: semi-leptonic decay of mesons
4: semi-leptonic decay of charmed baryons
5: no decay
6: decay b quark by phase space
7: no decay
8/9: V-A decay of b quark
11-12: no decay
13: exotic model 3: b-> q+l+l ( + spectator)
14: exotic model 4: b->qbar,qbar,l (+ spectator)
15: non-exclusive hadronic tau decays
17/18: eta decays to pi+ pi- pi0 and pi+ pi- gamma
19: omega -> pi+ pi- pi0
20: eta' -> gamma rho
21-30: Charmed baryon decay
41-43: Psi/Psi' radiative decay to e+e- / mu+mu-
       (41- not polarized, 42-trans., pol, 43-long. pol.)
51-53: Semileptonic decay of heavy mesons by Isgur/Wise model
       (51- not polarized, 52-trans. pol, 53-long.pol.)
61-63: Semileptonic decay by Koerner/Schuler model (polarized)
71-73: Semileptonic decay by ?????
81-85: Tau->nu + hadrons;
       (81 - V-A + flat phase space,
        82 - V-A + special function for 2pi- pi+ pi0
        83 - V-A + special function for pi- 3pi0
        84 - V-A + special function for 3pi- 2pi+
        85 - V-A + special function for 3pi- 2pi+ pi0)
10000>: User Decay Matrix through subroutine USRDEC
```


  - BR: branching fraction
  - dtr1, dtr2, ...: Names of daughter particles
- **HELICITY** specifies the helicity of every particle in the next channel command. This command only applies to the next decayl it is cleared after the next CHANNEL command is processed.
  - prob: Probability for this helicity state
  - hel1,hel2...: Helicity states of daughter particles
- **MATRIX** specifies 3 real number that can be used in the next channel command. If the next decay is two body, the three numbers specify the angular distribution $A+B\cos(\theta)+C\cos(\theta)^2$. This command only applies to the next decay; it is cleard after the next CHANNEL command is processed. If this command is not specified the angular distribution is generated flat.
  - A,B,C: $A+B\cos(\theta)+C\cos(\theta)^2$
- **ANGULAR_HELICITY** specifies the angular distribution of two body decay for a given helicity in the next CHANNEL command. This command only applies to the next decay; it is cleared after the next CHANNEL command is processed. If this command is not specified, the angular distribution is generated flat. Note: the helicity is assigned to the particle from a HELICITY_STATE command. In general, you should use 1 ANGULAR_HELICITY command for every possible helicity state.
  - hel : Helicity
  - A,B,C: $A+B\cos(\theta)+C\cos(\theta)^2$
- **ENDDECAY** finishes the decay sequence stated by the DECAY command. If missing, and an EOF or another DECAY command is encountered, a fake ENDDECAY is generated.
- **STABLE** causes a previously decayed particle to be declared stable.
  - particle_name : ASCII name of particle

3.2 Mixing and CP violation

Here is a summary of commands to control CP violation and mixing:

```
MIXING             part antipart rmix
CPEIGENSTATE       sinphi
CPTAG
```

where

- **MIXING** specifies how a particle and antiparticle mix. This command can occur anywhere.
  - part : Name of particle
  - antipart : Name of anti particle
  - rmix : Mixing parameter r. This is assigned to both particles.
- **CPEIGENSTATE** causes the next CHANNEL command to add the channel to a special CP eigenstate list in addition to the normal BR list. It is not sticky and is cleared after the CHANNEL command.
  - sinphi : Value of $\sin(\phi)$, the CP violating parameter. The value of sinphi should have opposite sign for particle and antiparticle.
- **CPTAG** causes the particles specified in the CHANNEL command to decay as CP eigenstate + tag decay. A tag decay is any decay not flagged as a CP eigenstate using the CPEIGENSTATE command. For $B^0\overline{B^0}$ CP violation, this command would occure just before the decay channel of the virtual photon (VPHO) to $B^0\overline{B^0}$. If CPTAG is not used, the CP decay will take place as CP eigenstate + any decay. This command is not sticky.

3.3 Defining particle properties

Syntax to define a new particle is:

```
QQBAR      name id
PARTICLE   name id1 id2 mass Q spin ctau [width mass_min mass_max]
PARITY     name ival
CPARITY    name ival
```

where

- **QQBAR** defines a quark-antiquark pair. These are used to specify inclusive final states in havy quark decays.
  - name : Name of qqbar pair in ASCII
  - id : ID number of this pair
- **PARTICLE** defines a new particle.
  - name : Name of particle
  - id1 : ID code of particle (indexes arrays in MCPROP)
  - id2 : Stable ID code (-1 -¿ particle is not stable)

  - mass : Mass in GeV
  - Q : charge
  - spin : Spin
  - ctau : c*lifetime in meters
  - width : [optional] FWHM in GeV
  - mass_min : [optional] Minimum mass
  - mass_max : [optional] Maximum mass
- **PARITY** defines the parity of the particle
  - name : Name of particle
  - ival : +-1 (or 0 for no parity)
- **CPARITY** defines the C parity of the particle
  - name : Name of particle
  - ival : +-1 (or 0 for no C parity)

3.4 Particle Names

Here is a part of the list of particle names defined in the default decay table. See /home/bsun03/bfactory/qq/runfil/decay.dec for the complete list of define particles. You can use these names in your decay table.

```
VPHO, GAMM, Z0, W+, W-
E-, E+, NUE, NUEB, MU-, MU+, NUM, NUMB, TAU-, TAU+, NUT, NUTB
PI+, PI-, K+, K-, K0, KB, D0, D0B, D+, D-, DS+, DS-
K0S, K0L, K10, K1B, RHO+, RHO-
B+, B-, B0, B0B
PI0 ETA, ETA', ETAC, ETAB
K*+, K*-, K*0, K*B, D*0, D*B, D*+, D*-, DS*+, DS*-
B*-, B*+, B*B, B*0m BS*B, BS*0
RHO0, OMEG, PHI, PSI
LAM, ALAM, LAMC, ALAMC
.......
```

3.5 Treatment of CP violation and mixing in QQ

3.5.1 CP violation via the mixing mechanism

For QQ to generate CP violation via the mixing mechanism, several conditions must be satisfied.

1. There must be exactly two daughters, e.g. B^0 and $\overline{B^0}$.
2. The MIXING command must have been previously used to force particle/antiparticle mixing. The mixing strength = $dM/d\Gamma$ must be non zero. Example : MIXING B0 B0B 0.66
3. The parent particle (e.g., virtual photon or $\Upsilon(4S)$) must have C parity defined as +1 or -1. This forces the B0 and B0B to be in a coherent state. The parity and C parity have already been defined in the default decay.
4. There must be at least one CP eigenstate decay for both B^0 and $\overline{B^0}$.
5. If CPTAG is specified, there must be at least one other decay of bothB^0 and $\overline{B^0}$ that is not declared to be a CP eigenstate.

3.5.2 CP violation via partial rate difference

1. Specify a branching fraction for a particle that is different than the branching fraction of the charge conjugate mode for the antiparticle.
   Example: $B^0 \to K^-\pi^+, \overline{B^0} \to K^+\pi^-$

3.5.3 Mixing in a coherent state

For QQ to generate mixing in a coherent state, several conditions must be satisfied.

1. There must be exactly two daughters, e.g. B^0 and $\overline{B^0}$.
2. The MIXING command must have been previously used to force particle/antiparticle mixing. The mixing strength = $dM/d\Gamma$ must be non zero. Example : MIXING B0 B0B 0.66
3. The parent particle (e.g., virtual photon or $\Upsilon(4S)$) must have C parity defined as +1 or -1. This forces the B0 and B0B to be in a coherent state. The parity and C parity have already been defined in the default decay.
4. There are no CP eigenstate decays defined or the sinphi parameter is 0.

3.5.4 Incoherent mixing

In this case, each particle that has a defined mixing is allowed to mix into its antiparticle and subsequently decay, regardless of what is happening elsewhere in the event. This is useful for studying D mixing when a D^{*+} decays into a D^0 which then mixes int $\overline{D^0}$.

1. The MIXING command must have been previously used to force particle/antiparticle mixing. The mixing strength = $dM/d\Gamma$ must be non zero. Example: MIXING D0 D0B 0.005

2. The particle is not in a coherent state with its antiparticle.

4 Using QQ from BELANL

If you have prepared both of the control file and the decay table, you are now ready to run QQ under BELANL. Here is the summary of the usage.

- Module Identifier : **QQ**
- Name appear in BELANL menu : **QQ_GENERATOR**
- Required library : -L/home/bsun03/bfactory/qq/lib -lBELLEQQ -lQQ -lBHLUMI -lKORB -lQQJS73 -lTYPSCN -lCLUTIL -lLSQFIT + CERNLIB
- Meaning of menu items in user interface routine:

```
ANL> QQ
**************************************************
*         QQ event generator for BELLE           *
*                                                *
*                BELLE version 1.0               *
*                  based on CLEO QQ Vers-8.08    *
*                     released by R.Itoh KEK     *
**************************************************

 ANL:  QQ: Setup Section

 READ_SETUP   MODIFY_SETUP DECAY_TABLE  USER_DECAY
 EXIT

 ANL: Select  1 Option
ANL>
```

  - **READ_SETUP** reads QQ control file. This must be called once. When you invoke this menu, the file name for the control file is asked. Enter the name of the file you prepared, or you can also use the default file /home/bsun03/bfactory/qq/runfil/belle_qq.ctr by just typing return.
  - **MODIFY_SETUP** not implemented yet.
  - **DECAY_TABLE** is used to change the default decay tables used in QQ. Normally you don't have to invoke this menu.

- **USER_DECAY** sets user-supplied decay table. When invoked, the file name of your decay table is asked. Enter your decay table here.
- **EXIT** exits from this menu and starts the actual initialization of QQ. This cannot be invoked before QQ control file is read by READ_SETUP.

- Environment variables:
  You can change the default control file and default decay tables by defining some environment variables. You can also specify your decay table by the environment variable.

```
QQ_CONTROL_FILE  - File name of the default control file
QQ_DECAY_TABLE   - File name of the default decay table
QQ_LUND_TABLE    - File name of the default decay table for JETSET7.3
QQ_USER_DECAY    - File name of user supplied decay table
```

- BELANL interface subroutines: /home/bsun03/bfactory/qq/belle_qq/full_qq.F

5 Examples

You can find some examples using QQ generator combined with FSIM and GEANT under BELANL in

`/home/bsun03/bfactory/belanl_example/examples/qq_fsim` and
`/home/bsun03/bfactory/belanl_example/examples/qq_geant`

directories. Here is a sample of QQ initialization through BELANL menu:

```
ANL> qq
***************************************************
*         QQ event generator for BELLE            *
*                                                 *
*                BELLE version 1.0                *
*                  based on CLEO QQ Vers-8.08     *
*                     released by R.Itoh KEK      *
***************************************************

 ANL:  QQ: Setup Section

 READ_SETUP   MODIFY_SETUP DECAY_TABLE  USER_DECAY
 EXIT

 ANL: Select  1 Option
ANL> read
```



```
Enter QQ setup file name: /home/bsun03/bfactory/qq/runfil/belle_bb.ctr ?
mybb.ctr  ! Enter control file here.
......... FFREAD output here
 ANL:  QQ: Setup Section

 READ_SETUP   MODIFY_SETUP DECAY_TABLE  USER_DECAY
 EXIT

 ANL: Select  1 Option
ANL> user
User Decay Table:  ? cp_psiks.dec    ! Enter decay table here.

 ANL:  QQ: Setup Section

 READ_SETUP   MODIFY_SETUP DECAY_TABLE  USER_DECAY
 EXIT

 ANL: Select  1 Option
ANL> e
 Starting Random Number Seed =       12345
..... Initialiation printout
```