HEP Simulations

0.1.0

Generated by Doxygen 1.8.17

Chapter 1

Todo List

Namespace HEP::Particles

Finish all particle definitions

Member HEP::Particles::Bc

Correct properties

Member HEP::Particles::phi_1020

Correct properties

Namespace HEP::Units

Finish all unit definitions

Chapter 2

Namespace Index

2.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 4

Namespace Documentation

4.1 HEP Namespace Reference

Namespaces

- Particles
- Units

Classes

- class Decay

  Decay Channel Class.
- class DecayTable

  Decay Table Class.
- class LorentzVector

  Lorentz Vector Class.
- class Particle

  Particle Class.

Functions

- std::ostream & **operator**<< (std::ostream &os, Decay &d)

Variables

- const DecayTable BcDecays = gen_BcDecays()

  HEP::DecayTable of all decays of the B_c meson. The `gen_BcDecays` function generates all the relevant channels.
- const DecayTable KplusDecays = gen_KplusDecays()

  DecayTable of all decays of the $K^{\pm}$ meson. The `gen_KplusDecay` function generates all the relevant channels.

4.1.1 Detailed Description

The HEP namespace contains all currently accepted particles within the standard model and beyond. The masses and data are taken from the Particle Data Group listings [1].

Warning

It is recommend you do not use the namespace within the script as there is a high probability of accidental overwriting. Instead use the particles as HEP::Particles::particle

Author

K. Zarebski

Copyright

MIT License

Date

last modified 2019-08-26

4.2 HEP::Particles Namespace Reference

Variables

- const Particle gamma

  Photon, γ.
- const Particle g

  Gluon, g.
- const Particle W

  W Boson, W^+.
- const Particle Z

  Z Boson, Z^0.
- const Particle H

  Higgs Boson, H.
- const Particle e = HEP::Particle("e", "-", 0.5109989461∗HEP::Units::MeV, 3E36∗HEP::Units::sec)

  Electron, e^-.
- const Particle mu = HEP::Particle("mu", "-", 105.6583745∗HEP::Units::MeV, 2.1969811E-6∗HEP::Units::sec)

  Muon, μ^-.
- const Particle tau = HEP::Particle("tau", "-", 1776.82∗HEP::Units::MeV, 2.90610E-13∗HEP::Units::sec)

  Tau, τ^-.
- const Particle nu_e = HEP::Particle("nu_e", "", 1E-6∗HEP::Units::MeV, -1)

  Electron Neutrino, ν_e.
- const Particle nu_mu = HEP::Particle("nu_mu", "", 1E-6∗HEP::Units::MeV, -1)

  Muon Neutrino, ν_μ.
- const Particle nu_tau = HEP::Particle("nu_tau", "", 1E-6∗HEP::Units::MeV, -1)

  Tau Neutrino, ν_τ.
- const Particle Kplus = HEP::Particle("K", "+", 493.677∗HEP::Units::MeV, 1.2380E-8)

Charged Kaon, K^+.

- const Particle phi_1020 = HEP::Particle("phi", "", 999, 999)

  Phi 1020, $\phi(1020)$.

- const Particle eta

  Eta meson, η.

- const Particle Piplus = HEP::Particle("pi", "+", 139.57061∗HEP::Units::MeV, 2.6033E-8∗HEP::Units::sec)

  Charged Pion, π^+.

- const Particle Pi0 = HEP::Particle("pi", "0", 134.9770∗HEP::Units::MeV, 8.52E-17∗HEP::Units::sec)

  Neutral Pion, π^0.

- const Particle Dplus

  Charged D meson, D^+.

- const Particle D0

  Neutral D meson, D^0.

- const Particle Ds

  Neutral D strange meson, D_s.

- const Particle Bplus

  Charge B meson, B^+.

- const Particle B0

  Neutral B meson, B^0.

- const Particle Bs

  Neutral B strange meson, B_s.

- const Particle Bc = HEP::Particle("Bc", "+", 999, 999)

  Charmed B meson, B_c.

- const Particle p

  Proton, p.

- const Particle n

  Neutron, n.

- const Particle Lambda

  Lambda baryon, Λ.

- const Particle Lambdab0

  Lambda b baryon, Λ_b^0.

- const Particle Sigma

  Neutral Sigma baryon, Σ^0.

- const Particle Sigmaplus

  Charged Sigma baryon, Σ^+.

- const Particle Xi0

  Neutral Xi baryon, Ξ^0.

- const Particle Ximinus

  Charged Xi baryon, Ξ^+.

- const Particle Omega

  Omega baryon, Ω.

- const Particle Lambdac

  Lambda c baryon, Λ_c.

4.2.1 Detailed Description

Namespace containing all particles listed within the PDG

Todo Finish all particle definitions

Author

K. Zarebski

Copyright

MIT License

Date

last modified 2019-08-26

4.2.2 Variable Documentation

4.2.2.1 Bc

```
const HEP::Particle HEP::Particles::Bc = HEP::Particle("Bc", "+", 999, 999)
```

Charmed B meson, B_c.

Todo Correct properties

4.2.2.2 phi_1020

```
const HEP::Particle HEP::Particles::phi_1020 = HEP::Particle("phi", "", 999, 999)
```

Phi 1020, $\phi(1020)$.

Todo Correct properties

4.3 HEP::Units Namespace Reference

Variables

- const double **m** = 1.
- const double **mm** = 1E-3∗m
- const double **nm** = 1E-9∗m
- const double **km** = 1E3∗m
- const double **cm** = 1E-2∗m
- const double **angstrom** = 1E-10∗m
- const double **mile** = 1.609344E3∗m
- const double **yd** = 0.9144∗m
- const double **AU** = 1.495979E12∗m
- const double **kg** = 1.
- const double **t** = 1E3∗kg
- const double **sec** = 1.
- const double **ms** = 1E-3∗sec
- const double **ns** = 1E-9∗sec
- const double **ps** = 1E-12∗sec
- const double **J** = 1.
- const double **eV** = 1.6E-19∗J
- const double **keV** = 1E3∗eV
- const double **MeV** = 1E6∗eV
- const double **GeV** = 1E9∗eV
- const double **TeV** = 1E12∗eV
- const double **b** = 1E-28∗m∗m
- const double **mb** = 1E-3∗b
- const double **nb** = 1E-9∗b
- const double **pb** = 1E-12∗b
- const double **fb** = 1E-15∗b
- const double **inv_pb** = 1./pb
- const double **inv_fb** = 1./fb
- const double **N** = 1.
- const double **kN** = 1E3∗N

4.3.1 Detailed Description

Namespace containing all units for easy conversion to SI units for calculations

Todo Finish all unit definitions

Author

K. Zarebski

Copyright

MIT License

Date

last modified 2019-08-26

Chapter 5

Class Documentation

5.1 HEP::Decay Class Reference

Decay Channel Class.

`#include <DecayTable.hxx>`

Public Member Functions

- Decay ()
  Blank decay object.
- const std::string getDecStr () const
  Get the decay description as a string.
- Decay (std::vector< Particle > daughters, double probability, Particle mother=Particle())
  Construct a decay with a given mother, daughters and branching ratio.
- bool isValid (double threshold)
  Used to check probability of decay occuring against a threshold.
- const double getBR () const
  Returns the branching ratio.
- const Particle getMother () const
  Returns the mother object.
- void setMother (Particle &Mother)
  Sets the mother particle.
- const std::vector< Particle > getDaughters () const
  Returns a vector of the daughter particles.

5.1.1 Detailed Description

Decay Channel Class.

Class describing a single particle decay with information on the mother and daughter particles and branching ratio.

Version

0.1.0

Author

Kristian Zarebski

Date

last modified 2019-08-25

Copyright

The documentation for this class was generated from the following files:

- include/DecayTable.hxx
- src/DecayTable.cxx

5.2 HEP::DecayTable Class Reference

Decay Table Class.

`#include <DecayTable.hxx>`

Public Member Functions

- DecayTable (const Particle &mother)
  Construct a decay table for a given mother particle.
- void addDecay (Decay &decay)
  Add a new decay to the decay table.
- std::vector< Decay > getDecays ()
  Get the decays in decay tablea as a vector.
- const Decay getRandom () const
  Get a decay at random based on the branching ratios.
- void Print ()
  Print the decay table.

5.2.1 Detailed Description

Decay Table Class.

Class containing all decay channels for a given mother particle this is used to generate a decay at random

Version

0.1.0

Author

Kristian Zarebski

Date

last modified 2019-08-26

Copyright

MIT License

Examples

KplusDecTable.hxx.

The documentation for this class was generated from the following files:

- include/DecayTable.hxx
- src/DecayTable.cxx

5.3 HEP::LorentzVector Class Reference

Lorentz Vector Class.

`#include <LorentzVector.hxx>`

Public Member Functions

- LorentzVector ()

  Create a blank LorentzVector with the default values of -9999.
- const double & operator[ ] (size_t i)

  Specify a component of the LorentzVector by index [0, 3] for (x_0, x_1, x_2, x_3).
- double operator[ ] (size_t i) const

  Specify a component of the LorentzVector by index [0, 3] for (x_0, x_1, x_2, x_3).
- LorentzVector (double x0, double x1, double x2, double x3)

  Create a new LorentzVector by giving the four values (x_0, x_1, x_2, x_3).
- const double magnitude () const

  Returns the magnitude of the vector as $r = \sqrt{x_0^2 - x_1^2 - x_2^2 - x_3^2}$.

Friends

- std::ostream & operator<< (std::ostream &os, const LorentzVector &lv)

  Output the components of the LorentzVector to the ostream when printing.

5.3.1 Detailed Description

Lorentz Vector Class.

Class to represent Lorentz vectors of the form e.g. four momenta

Version

0.1.0

Author

Kristian Zarebski

Date

last modified 2019-08-26

Copyright

MIT License

The documentation for this class was generated from the following files:

- include/LorentzVector.hxx
- src/LorentzVector.cxx

5.4 HEP::Particle Class Reference

Particle Class.

```
#include <Particle.hxx>
```

Public Member Functions

- Particle ()

  Default constructor with properties set to -1.
- Particle (std::string, std::string, double, double)

  Construct a particle with a given name, sign, mass and lifetime. The energy will be taken to be the rest mass.
- Particle (double, double, double, double)

  Construct a particle giving only the four momentum components.
- const std::string getName () const

  Returns as a string the particle name.
- const Particle anti () const

  Get the antiparticle partner of the current particle.
- void Fire (double)

  Fire the particle at a given energy.
- const double M () const

Get the mass of the particle.

- const double phi () const

  Get the azimuthal angle ϕ of the particle trajectory.

- const double y () const

  Get the rapidity $y = \frac{1}{2}\log\left(\frac{E+p_z}{E-p_z}\right)$ of the particle.

- const double eta () const

  Get the pseudorapidity $\eta = -\log\left(\tan\frac{\theta}{2}\right)$ of the particle.

- const double theta () const

  Get the angle θ of the particle trajectory in the $x-z$ plane.

- const HEP::LorentzVector momentum () const

  Get the LorentzVector of the given particle.

- const double PT () const

  Get the transverse momentum p_T of the particle.

- operator bool () const

  Returns true if the particle has been initialised (does not have default blank values)

Friends

- std::ostream & operator<< (std::ostream &, Particle &)

  Output the particle properties to the ostream.

5.4.1 Detailed Description

Particle Class.

Class representing a single particle with information on mass and lifetime

Version

0.1.0

Author

Kristian Zarebski

Date

last modified 2019-08-26

Copyright

MIT License

The documentation for this class was generated from the following files:

- include/Particle.hxx
- src/Particle.cxx

Chapter 6

Example Documentation

6.1 KplusDecTable.hxx

An example of the Decay Table class for K+

```
#ifndef __KPLUSDECTABLE__
#define __KPLUSDECTABLE__
#include "DecayTable.hxx"
#include "CommonParticles.hxx"
const HEP::DecayTable gen_KplusDecays();
namespace HEP
{
    extern const DecayTable KplusDecays;
}
#endif
```

Bibliography

[1] K. A. Olive et al. Review of Particle Physics. *Chin. Phys.*, C38:090001, 2014. 8

Index