XNLO - UPPE

1.3.0

Generated by Doxygen 1.8.13

Contents

Chapter 1

Class Index

1.1 Class List

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

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

Chapter 3

Class Documentation

3.1 capillary_fibre Class Reference

`#include <capillary_fibre.hpp>`

Public Member Functions

- capillary_fibre (double Z_, grid_rkr &rkr_, grid_tw &tw_, physics_textbook &physics_, maths_textbook &maths_)

Public Attributes

- ArrayXXcd gamma
- double Z
- double R
- double n_glass

3.1.1 Detailed Description

Originally created by Patrick Anderson. Modified by Samuel Senior on 10/03/2017. "capillary_fibre" describes the dimensions and dispersion properties of a dielectric capillary type fibre.

3.1.2 Constructor & Destructor Documentation

3.1.2.1 capillary_fibre()

```
capillary_fibre::capillary_fibre (
            double Z_,
            grid_rkr & rkr_,
            grid_tw & tw_,
            physics_textbook & physics_,
            maths_textbook & maths_ )
```

Constructor

3.1.3 Member Data Documentation

3.1.3.1 gamma

```
ArrayXXcd capillary_fibre::gamma
```

3.1.3.2 n_glass

```
double capillary_fibre::n_glass
```

3.1.3.3 R

```
double capillary_fibre::R
```

3.1.3.4 Z

```
double capillary_fibre::Z
```

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

- /home/sam/Project/XNLO/UPPE/src/capillary_fibre.hpp
- /home/sam/Project/XNLO/UPPE/src/capillary_fibre.cpp

3.2 Config_Settings Class Reference

```
#include <config_settings.hpp>
```

Public Member Functions

- Config_Settings ()
- void read_in (std::string, bool print_to_screen=true)
- void check_paths (bool print_to_screen=true)
- void step_path (int step)
- void print ()
- void print (std::string)
- int n_z ()
- void n_z_set (int)
- std::string n_z_description ()
- void n_z_description_set (std::string)
- int n_r ()
- void n_r_set (int)
- std::string n_r_description ()
- void n_r_description_set (std::string)
- int n_m ()
- void n_m_set (int)
- std::string n_m_description ()
- void n_m_description_set (std::string)
- int n_t ()
- void n_t_set (int)
- std::string n_t_description ()
- void n_t_description_set (std::string)
- double T ()
- void T_set (double)
- std::string T_description ()
- void T_description_set (std::string)
- double w_active_min ()
- void w_active_min_set (double)
- std::string w_active_min_description ()
- void w_active_min_description_set (std::string)
- double w_active_max ()
- void w_active_max_set (double)
- std::string w_active_max_description ()
- void w_active_max_description_set (std::string)
- double Z ()
- void Z_set (double)
- std::string Z_description ()
- void Z_description_set (std::string)
- double R ()
- void R_set (double)
- std::string R_description ()
- void R_description_set (std::string)
- double press ()
- void press_set (double)
- std::string press_description ()
- void press_description_set (std::string)
- double p_av ()
- void p_av_set (double)
- std::string p_av_description ()
- void p_av_description_set (std::string)
- double rep ()
- void rep_set (double)
- std::string rep_description ()

- void rep_description_set (std::string)
- double fwhm ()
- void fwhm_set (double)
- std::string fwhm_description ()
- void fwhm_description_set (std::string)
- double I_0 ()
- void I_0_set (double)
- std::string I_0_description ()
- void I_0_description_set (std::string)
- double ceo ()
- void ceo_set (double)
- std::string ceo_description ()
- void ceo_description_set (std::string)
- double waist ()
- void waist_set (double)
- std::string waist_description ()
- void waist_description_set (std::string)
- std::string pend_path ()
- void pend_path_set (std::string)
- std::string pend_path_description ()
- void pend_path_description_set (std::string)
- std::string path_input_j0 ()
- void path_input_j0_set (std::string)
- std::string path_input_j0_description ()
- void path_input_j0_description_set (std::string)
- std::string path_A_w_R ()
- void path_A_w_R_set (std::string)
- std::string path_A_w_R_description ()
- void path_A_w_R_description_set (std::string)
- std::string path_A_w_I ()
- void path_A_w_I_set (std::string)
- std::string path_A_w_I_description ()
- void path_A_w_I_description_set (std::string)
- std::string path_w_active ()
- void path_w_active_set (std::string)
- std::string path_w_active_description ()
- void path_w_active_description_set (std::string)
- std::string path_electron_density ()
- void path_electron_density_set (std::string)
- std::string path_electron_density_description ()
- void path_electron_density_description_set (std::string)
- std::string path_HHG_R ()
- void path_HHG_R_set (std::string)
- std::string path_HHG_R_description ()
- void path_HHG_R_description_set (std::string)
- std::string path_HHG_I ()
- void path_HHG_I_set (std::string)
- std::string path_HHG_I_description ()
- void path_HHG_I_description_set (std::string)
- std::string path_HHG_w ()
- void path_HHG_w_set (std::string)
- std::string path_HHG_w_description ()
- void path_HHG_w_description_set (std::string)
- std::string path_HHG_E ()
- void path_HHG_E_set (std::string)

- std::string path_HHG_E_description ()
- void path_HHG_E_description_set (std::string)
- std::string path_config_file ()
- void path_config_file_set (std::string)
- std::string path_config_file_description ()
- void path_config_file_description_set (std::string)
- std::string path_config_log ()
- void path_config_log_set (std::string)
- std::string path_config_log_description ()
- void path_config_log_description_set (std::string)

Private Types

- enum SN {
  SN::n_z = 0, SN::n_r, SN::n_m, SN::n_t,
  SN::T, SN::w_active_min, SN::w_active_max, SN::Z,
  SN::R, SN::press, SN::p_av, SN::rep,
  SN::fwhm, SN::l_0, SN::ceo, SN::waist,
  SN::pend_path, SN::path_input_j0, SN::path_A_w_R, SN::path_A_w_I,
  SN::path_w_active, SN::path_HHG_R, SN::path_HHG_I, SN::path_HHG_w,
  SN::path_HHG_E, SN::path_config_file, SN::path_config_log, SN::LAST_SN_ENTRY }

Private Member Functions

- void set_variable (std::string &, std::string &, std::string &, bool print_to_screen=true)
- std::string set_path (std::string, std::string, std::string pend="")
- std::string set_pre_path (std::string, std::string)
- std::string set_post_path (std::string, std::string)

Private Attributes

- int n_z_ = 50
- int n_r_ = 20
- int n_m_ = 20
- int n_t_ = 4096
- double T_ = 500.0e-15
- double w_active_min_ = 2.0e14
- double w_active_max_ = 8.0e15
- double Z_ = 5.0e-3
- double R_ = 75.0e-6
- double press_ = 100.0e-3
- double p_av_ = 1.0
- double rep_ = 1.0e3
- double fwhm_ = 40e-15
- double l_0_ = 800e-9
- double ceo_ = 0.0
- double waist_ = 48.0e-6
- std::string path_input_j0_ = "../input/J0_zeros.bin"
- std::string path_A_w_R_ = "../output/A_w_R.bin"
- std::string path_A_w_I_ = "../output/A_w_I.bin"
- std::string path_w_active_ = "../output/w_active.bin"
- std::string path_electron_density_ = "../output/electron_density.bin"

- std::string path_HHG_R_ = "../output/HHG_R.bin"
- std::string path_HHG_I_ = "../output/HHG_I.bin"
- std::string path_HHG_w_ = "../output/HHG_w.bin"
- std::string path_HHG_E_ = "../output/HHG_E.bin"
- std::string path_config_file_ = "./config.txt"
- std::string path_config_log_ = "../output/config_log.txt"
- std::string pend_path_ = "prepend"
- std::string n_z_description_ = "(default) (int) Number of steps in Z"
- std::string n_r_description_ = "(default) (int) The z_r value"
- std::string n_m_description_ = "(default) (int) Number of modes"
- std::string n_t_description_ = "(default) (int) The z_t value"
- std::string T_description_ = "(default) (double) The T value"
- std::string w_active_min_description_ = "(default) (double) Minimum angular frequency"
- std::string w_active_max_description_ = "(default) (double) Maximum angular frequency"
- std::string Z_description_ = "(default) (double) Length of capillary"
- std::string R_description_ = "(default) (double) Radius of capillary"
- std::string press_description_ = "(default) (double) Pressure of the gas"
- std::string p_av_description_ = "(default) (double) The p_av value"
- std::string rep_description_ = "(default) (double) The rep value"
- std::string fwhm_description_ = "(default) (double) Full width at half max"
- std::string l_0_description_ = "(default) (double) Laser central wavelength"
- std::string ceo_description_ = "(default) (double) The ceo value"
- std::string waist_description_ = "(default) (double) The waist value"
- std::string pend_path_description_ = "(default) (std::string) Pending switch"
- std::string path_input_j0_description_ = "(default) (std::string) Path to J0_zeros.bin"
- std::string path_A_w_R_description_ = "(default) (std::string) Path of A_w_R"
- std::string path_A_w_I_description_ = "(default) (std::string) Path of A_w_I"
- std::string path_w_active_description_ = "(default) (std::string) Path of w_active"
- std::string path_electron_density_description_ = "(default) (std::string) Path of electron_density"
- std::string path_HHG_R_description_ = "(default) (std::string) Path of HHG_R"
- std::string path_HHG_I_description_ = "(default) (std::string) Path of HHG_I"
- std::string path_HHG_w_description_ = "(default) (std::string) Path of HHG w"
- std::string path_HHG_E_description_ = "(default) (std::string) Path of HHG E"
- std::string path_config_file_description_ = "(default) (std::string) config.txt path"
- std::string path_config_log_description_ = "(default) (std::string) config_log.txt path"

Static Private Attributes

- static const char * setting_name [ ]

3.2.1 Detailed Description

Modified by Samuel Senior on 05/02/2017. Reads in input parameters and settings from a config file.

3.2.2 Member Enumeration Documentation

3.2.2.1 SN

```
enum Config_Settings::SN  [strong], [private]
```

Enumerator

n_z	
n_r	
n_m	
n_t	
T	
w_active_min	
w_active_max	
Z	
R	
press	
p_av	
rep	
fwhm	
I_0	
ceo	
waist	
pend_path	
path_input_j0	
path_A_w_R	
path_A_w_I	
path_w_active	
path_HHG_R	
path_HHG_I	
path_HHG_w	
path_HHG_E	
path_config_file	
path_config_log	
LAST_SN_ENTRY	

3.2.3 Constructor & Destructor Documentation

3.2.3.1 Config_Settings()

```
Config_Settings::Config_Settings ( )
```

Constructor

3.2.4 Member Function Documentation

3.2.4.1 ceo()

```
double Config_Settings::ceo ( )
```

3.2.4.2 ceo_description()

```
std::string Config_Settings::ceo_description ( )
```

3.2.4.3 ceo_description_set()

```
void Config_Settings::ceo_description_set (
            std::string description )
```

3.2.4.4 ceo_set()

```
void Config_Settings::ceo_set (
            double value )
```

3.2.4.5 check_paths()

```
void Config_Settings::check_paths (
            bool print_to_screen = true )
```

3.2.4.6 fwhm()

```
double Config_Settings::fwhm ( )
```

3.2.4.7 fwhm_description()

```
std::string Config_Settings::fwhm_description ( )
```

3.2.4.8 fwhm_description_set()

```
void Config_Settings::fwhm_description_set (
            std::string description )
```

3.2.4.9 fwhm_set()

```
void Config_Settings::fwhm_set (
            double value )
```

3.2.4.10 l_0()

```
double Config_Settings::l_0 ( )
```

3.2.4.11 l_0_description()

```
std::string Config_Settings::l_0_description ( )
```

3.2.4.12 l_0_description_set()

```
void Config_Settings::l_0_description_set (
            std::string description )
```

3.2.4.13 l_0_set()

```
void Config_Settings::l_0_set (
            double value )
```

3.2.4.14 n_m()

```
int Config_Settings::n_m ( )
```

3.2.4.15 n_m_description()

```
std::string Config_Settings::n_m_description ( )
```

3.2.4.16 n_m_description_set()

```
void Config_Settings::n_m_description_set (
            std::string description )
```

3.2.4.17 n_m_set()

```
void Config_Settings::n_m_set (
            int value )
```

3.2.4.18 n_r()

```
int Config_Settings::n_r ( )
```

3.2.4.19 n_r_description()

```
std::string Config_Settings::n_r_description ( )
```

3.2.4.20 n_r_description_set()

```
void Config_Settings::n_r_description_set (
            std::string description )
```

3.2.4.21 n_r_set()

```
void Config_Settings::n_r_set (
            int value )
```

3.2.4.22 n_t()

```
int Config_Settings::n_t ( )
```

3.2.4.23 n_t_description()

```
std::string Config_Settings::n_t_description ( )
```

3.2.4.24 n_t_description_set()

```
void Config_Settings::n_t_description_set (
            std::string description )
```

3.2.4.25 n_t_set()

```
void Config_Settings::n_t_set (
            int value )
```

3.2.4.26 n_z()

```
int Config_Settings::n_z ( )
```

3.2.4.27 n_z_description()

```
std::string Config_Settings::n_z_description ( )
```

3.2.4.28 n_z_description_set()

```
void Config_Settings::n_z_description_set (
            std::string description )
```

3.2.4.29 n_z_set()

```
void Config_Settings::n_z_set (
            int value )
```

3.2.4.30 p_av()

```
double Config_Settings::p_av ( )
```

3.2.4.31 p_av_description()

```
std::string Config_Settings::p_av_description ( )
```

3.2.4.32 p_av_description_set()

```
void Config_Settings::p_av_description_set (
            std::string description )
```

3.2.4.33 p_av_set()

```
void Config_Settings::p_av_set (
            double value )
```

3.2.4.34 path_A_w_I()

```
std::string Config_Settings::path_A_w_I ( )
```

3.2.4.35 path_A_w_I_description()

```
std::string Config_Settings::path_A_w_I_description ( )
```

3.2.4.36 path_A_w_I_description_set()

```
void Config_Settings::path_A_w_I_description_set (
            std::string description )
```

3.2.4.37 path_A_w_I_set()

```
void Config_Settings::path_A_w_I_set (
            std::string value )
```

3.2.4.38 path_A_w_R()

```
std::string Config_Settings::path_A_w_R ( )
```

3.2.4.39 path_A_w_R_description()

```
std::string Config_Settings::path_A_w_R_description ( )
```

3.2.4.40 path_A_w_R_description_set()

```
void Config_Settings::path_A_w_R_description_set (
            std::string description )
```

3.2.4.41 path_A_w_R_set()

```
void Config_Settings::path_A_w_R_set (
            std::string value )
```

3.2.4.42 path_config_file()

```
std::string Config_Settings::path_config_file ( )
```

3.2.4.43 path_config_file_description()

```
std::string Config_Settings::path_config_file_description ( )
```

3.2.4.44 path_config_file_description_set()

```
void Config_Settings::path_config_file_description_set (
            std::string description )
```

3.2.4.45 path_config_file_set()

```
void Config_Settings::path_config_file_set (
            std::string value )
```

3.2.4.46 path_config_log()

```
std::string Config_Settings::path_config_log ( )
```

3.2.4.47 path_config_log_description()

```
std::string Config_Settings::path_config_log_description ( )
```

3.2.4.48 path_config_log_description_set()

```
void Config_Settings::path_config_log_description_set (
            std::string description )
```

3.2.4.49 path_config_log_set()

```
void Config_Settings::path_config_log_set (
            std::string value )
```

3.2.4.50 path_electron_density()

```
std::string Config_Settings::path_electron_density ( )
```

3.2.4.51 path_electron_density_description()

```
std::string Config_Settings::path_electron_density_description ( )
```

3.2.4.52 path_electron_density_description_set()

```
void Config_Settings::path_electron_density_description_set (
            std::string description )
```

3.2.4.53 path_electron_density_set()

```
void Config_Settings::path_electron_density_set (
            std::string value )
```

3.2.4.54 path_HHG_E()

```
std::string Config_Settings::path_HHG_E ( )
```

3.2.4.55 path_HHG_E_description()

```
std::string Config_Settings::path_HHG_E_description ( )
```

3.2.4.56 path_HHG_E_description_set()

```
void Config_Settings::path_HHG_E_description_set (
            std::string description )
```

3.2.4.57 path_HHG_E_set()

```
void Config_Settings::path_HHG_E_set (
            std::string value )
```

3.2.4.58 path_HHG_I()

```
std::string Config_Settings::path_HHG_I ( )
```

3.2.4.59 path_HHG_I_description()

```
std::string Config_Settings::path_HHG_I_description ( )
```

3.2.4.60 path_HHG_I_description_set()

```
void Config_Settings::path_HHG_I_description_set (
            std::string description )
```

3.2.4.61 path_HHG_I_set()

```
void Config_Settings::path_HHG_I_set (
            std::string value )
```

3.2.4.62 path_HHG_R()

```
std::string Config_Settings::path_HHG_R ( )
```

3.2.4.63 path_HHG_R_description()

```
std::string Config_Settings::path_HHG_R_description ( )
```

3.2.4.64 path_HHG_R_description_set()

```
void Config_Settings::path_HHG_R_description_set (
            std::string description )
```

3.2.4.65 path_HHG_R_set()

```
void Config_Settings::path_HHG_R_set (
            std::string value )
```

3.2.4.66 path_HHG_w()

```
std::string Config_Settings::path_HHG_w ( )
```

3.2.4.67 path_HHG_w_description()

```
std::string Config_Settings::path_HHG_w_description ( )
```

3.2.4.68 path_HHG_w_description_set()

```
void Config_Settings::path_HHG_w_description_set (
            std::string description )
```

3.2.4.69 path_HHG_w_set()

```
void Config_Settings::path_HHG_w_set (
            std::string value )
```

3.2.4.70 path_input_j0()

```
std::string Config_Settings::path_input_j0 ( )
```

3.2.4.71 path_input_j0_description()

```
std::string Config_Settings::path_input_j0_description ( )
```

3.2.4.72 path_input_j0_description_set()

```
void Config_Settings::path_input_j0_description_set (
            std::string description )
```

3.2.4.73 path_input_j0_set()

```
void Config_Settings::path_input_j0_set (
            std::string value )
```

3.2.4.74 path_w_active()

```
std::string Config_Settings::path_w_active ( )
```

3.2.4.75 path_w_active_description()

```
std::string Config_Settings::path_w_active_description ( )
```

3.2.4.76 path_w_active_description_set()

```
void Config_Settings::path_w_active_description_set (
            std::string description )
```

3.2.4.77 path_w_active_set()

```
void Config_Settings::path_w_active_set (
            std::string value )
```

3.2.4.78 pend_path()

```
std::string Config_Settings::pend_path ( )
```

3.2.4.79 pend_path_description()

```
std::string Config_Settings::pend_path_description ( )
```

3.2.4.80 pend_path_description_set()

```
void Config_Settings::pend_path_description_set (
            std::string description )
```

3.2.4.81 pend_path_set()

```
void Config_Settings::pend_path_set (
            std::string value )
```

3.2.4.82 press()

```
double Config_Settings::press ( )
```

3.2.4.83 press_description()

```
std::string Config_Settings::press_description ( )
```

3.2.4.84 press_description_set()

```
void Config_Settings::press_description_set (
            std::string description )
```

3.2.4.85 press_set()

```
void Config_Settings::press_set (
            double value )
```

3.2.4.86 print() [1/2]

```
void Config_Settings::print ( )
```

3.2.4.87 print() [2/2]

```
void Config_Settings::print (
            std::string path_ )
```

3.2.4.88 R()

```
double Config_Settings::R ( )
```

3.2.4.89 R_description()

```
std::string Config_Settings::R_description ( )
```

3.2.4.90 R_description_set()

```
void Config_Settings::R_description_set (
            std::string description )
```

3.2.4.91 R_set()

```
void Config_Settings::R_set (
            double value )
```

3.2.4.92 read_in()

```
void Config_Settings::read_in (
            std::string path,
            bool print_to_screen = true )
```

3.2.4.93 rep()

```
double Config_Settings::rep ( )
```

3.2.4.94 rep_description()

```
std::string Config_Settings::rep_description ( )
```

3.2.4.95 rep_description_set()

```
void Config_Settings::rep_description_set (
            std::string description )
```

3.2.4.96 rep_set()

```
void Config_Settings::rep_set (
            double value )
```

3.2.4.97 set_path()

```
std::string Config_Settings::set_path (
            std::string path,
            std::string pending_string,
            std::string pend = "" )  [private]
```

3.2.4.98 set_post_path()

```
std::string Config_Settings::set_post_path (
            std::string path,
            std::string post_path )  [private]
```

3.2.4.99 set_pre_path()

```
std::string Config_Settings::set_pre_path (
            std::string pre_path,
            std::string path )  [private]
```

3.2.4.100 set_variable()

```
void Config_Settings::set_variable (
            std::string & variable_name,
            std::string & variable_value_str,
            std::string & input_description_char,
            bool print_to_screen = true )  [private]
```

3.2.4.101 step_path()

```
void Config_Settings::step_path (
            int step )
```

3.2.4.102 T()

```
double Config_Settings::T ( )
```

3.2.4.103 T_description()

```
std::string Config_Settings::T_description ( )
```

3.2.4.104 T_description_set()

```
void Config_Settings::T_description_set (
            std::string description )
```

3.2.4.105 T_set()

```
void Config_Settings::T_set (
            double value )
```

3.2.4.106 w_active_max()

```
double Config_Settings::w_active_max ( )
```

3.2.4.107 w_active_max_description()

```
std::string Config_Settings::w_active_max_description ( )
```

3.2.4.108 w_active_max_description_set()

```
void Config_Settings::w_active_max_description_set (
            std::string description )
```

3.2.4.109 w_active_max_set()

```
void Config_Settings::w_active_max_set (
            double value )
```

3.2.4.110 w_active_min()

```
double Config_Settings::w_active_min ( )
```

3.2.4.111 w_active_min_description()

```
std::string Config_Settings::w_active_min_description ( )
```

3.2.4.112 w_active_min_description_set()

```
void Config_Settings::w_active_min_description_set (
            std::string description )
```

3.2.4.113 w_active_min_set()

```
void Config_Settings::w_active_min_set (
            double value )
```

3.2.4.114 waist()

```
double Config_Settings::waist ( )
```

3.2.4.115 waist_description()

```
std::string Config_Settings::waist_description ( )
```

3.2.4.116 waist_description_set()

```
void Config_Settings::waist_description_set (
            std::string description )
```

3.2.4.117 waist_set()

```
void Config_Settings::waist_set (
            double value )
```

3.2.4.118 Z()

```
double Config_Settings::Z ( )
```

3.2.4.119 Z_description()

```
std::string Config_Settings::Z_description ( )
```

3.2.4.120 Z_description_set()

```
void Config_Settings::Z_description_set (
            std::string description )
```

3.2.4.121 Z_set()

```
void Config_Settings::Z_set (
            double value )
```

3.2.5 Member Data Documentation

3.2.5.1 ceo_

```
double Config_Settings::ceo_ = 0.0  [private]
```

3.2.5.2 ceo_description_

```
std::string Config_Settings::ceo_description_ = "(default) (double) The ceo value"  [private]
```

3.2.5.3 fwhm_

```
double Config_Settings::fwhm_ = 40e-15  [private]
```

3.2.5.4 fwhm_description_

```
std::string Config_Settings::fwhm_description_ = "(default) (double) Full width at half max"
[private]
```

3.2.5.5 l_0_

```
double Config_Settings::l_0_ = 800e-9  [private]
```

3.2.5.6 l_0_description_

```
std::string Config_Settings::l_0_description_ = "(default) (double) Laser central wavelength"
[private]
```

3.2.5.7 n_m_

```
int Config_Settings::n_m_ = 20  [private]
```

3.2.5.8 n_m_description_

```
std::string Config_Settings::n_m_description_ = "(default) (int) Number of modes"  [private]
```

3.2.5.9 n_r_

```
int Config_Settings::n_r_ = 20  [private]
```

3.2.5.10 n_r_description_

```
std::string Config_Settings::n_r_description_ = "(default) (int) The z_r value"  [private]
```

3.2.5.11 n_t_

```
int Config_Settings::n_t_ = 4096  [private]
```

3.2.5.12 n_t_description_

```
std::string Config_Settings::n_t_description_ = "(default) (int) The z_t value"  [private]
```

3.2.5.13 n_z_

```
int Config_Settings::n_z_ = 50  [private]
```

3.2.5.14 n_z_description_

```
std::string Config_Settings::n_z_description_ = "(default) (int) Number of steps in Z"  [private]
```

3.2.5.15 p_av_

```
double Config_Settings::p_av_ = 1.0  [private]
```

3.2.5.16 p_av_description_

```
std::string Config_Settings::p_av_description_ = "(default) (double) The p_av value"  [private]
```

3.2.5.17 path_A_w_I_

```
std::string Config_Settings::path_A_w_I_ = "../output/A_w_I.bin"  [private]
```

3.2.5.18 path_A_w_I_description_

```
std::string Config_Settings::path_A_w_I_description_ = "(default) (std::string) Path of A_w_I"
[private]
```

3.2.5.19 path_A_w_R_

```
std::string Config_Settings::path_A_w_R_ = "../output/A_w_R.bin"  [private]
```

3.2.5.20 path_A_w_R_description_

```
std::string Config_Settings::path_A_w_R_description_ = "(default) (std::string) Path of A_w_R"
[private]
```

3.2.5.21 path_config_file_

```
std::string Config_Settings::path_config_file_ = "./config.txt"  [private]
```

3.2.5.22 path_config_file_description_

```
std::string Config_Settings::path_config_file_description_ = "(default) (std::string) config.↩
txt path"  [private]
```

3.2.5.23 path_config_log_

```
std::string Config_Settings::path_config_log_ = "../output/config_log.txt"  [private]
```

3.2.5.24 path_config_log_description_

```
std::string Config_Settings::path_config_log_description_ = "(default) (std::string) config_↩
log.txt path"  [private]
```

3.2.5.25 path_electron_density_

```
std::string Config_Settings::path_electron_density_ = "../output/electron_density.bin"  [private]
```

3.2.5.26 path_electron_density_description_

```
std::string Config_Settings::path_electron_density_description_ = "(default) (std::string)
Path of electron_density"  [private]
```

3.2.5.27 path_HHG_E_

```
std::string Config_Settings::path_HHG_E_ = "../output/HHG_E.bin"  [private]
```

3.2.5.28 path_HHG_E_description_

```
std::string Config_Settings::path_HHG_E_description_ = "(default) (std::string) Path of HHG E"
[private]
```

3.2.5.29 path_HHG_I_

```
std::string Config_Settings::path_HHG_I_ = "../output/HHG_I.bin"  [private]
```

3.2.5.30 path_HHG_I_description_

```
std::string Config_Settings::path_HHG_I_description_ = "(default) (std::string) Path of HHG_I"
[private]
```

3.2.5.31 path_HHG_R_

```
std::string Config_Settings::path_HHG_R_ = "../output/HHG_R.bin"  [private]
```

3.2.5.32 path_HHG_R_description_

```
std::string Config_Settings::path_HHG_R_description_ = "(default) (std::string) Path of HHG_R"
[private]
```

3.2.5.33 path_HHG_w_

```
std::string Config_Settings::path_HHG_w_ = "../output/HHG_w.bin"  [private]
```

3.2.5.34 path_HHG_w_description_

```
std::string Config_Settings::path_HHG_w_description_ = "(default) (std::string) Path of HHG w"
[private]
```

3.2.5.35 path_input_j0_

```
std::string Config_Settings::path_input_j0_ = "../input/J0_zeros.bin"  [private]
```

3.2.5.36 path_input_j0_description_

```
std::string Config_Settings::path_input_j0_description_ = "(default) (std::string) Path to
J0_zeros.bin"  [private]
```

3.2.5.37 path_w_active_

```
std::string Config_Settings::path_w_active_ = "../output/w_active.bin"  [private]
```

3.2.5.38 path_w_active_description_

```
std::string Config_Settings::path_w_active_description_ = "(default) (std::string) Path of w↩
_active"  [private]
```

3.2.5.39 pend_path_

```
std::string Config_Settings::pend_path_ = "prepend"  [private]
```

3.2.5.40 pend_path_description_

```
std::string Config_Settings::pend_path_description_ = "(default) (std::string) Pending switch"
[private]
```

3.2.5.41 press_

```
double Config_Settings::press_ = 100.0e-3  [private]
```

3.2.5.42 press_description_

```
std::string Config_Settings::press_description_ = "(default) (double) Pressure of the gas"
[private]
```

3.2.5.43 R_

```
double Config_Settings::R_ = 75.0e-6  [private]
```

3.2.5.44 R_description_

```
std::string Config_Settings::R_description_ = "(default) (double) Radius of capillary"  [private]
```

3.2.5.45 rep_

```
double Config_Settings::rep_ = 1.0e3  [private]
```

3.2.5.46 rep_description_

```
std::string Config_Settings::rep_description_ = "(default) (double) The rep value"  [private]
```

3.2.5.47 setting_name

```
const char * Config_Settings::setting_name  [static], [private]
```

Initial value:

```
= {
  "n_z", "n_r", "n_m",
  "n_t", "T", "w_active_min", "w_active_max",
  "Z", "R",
  "press",
  "p_av", "rep", "fwhm", "l_0", "ceo", "waist",
  "pend_path",
  "path_input_j0",
  "path_A_w_R", "path_A_w_I", "path_w_active",
  "path_HHG_R", "path_HHG_I", "path_HHG_w", "path_HHG_E",
  "path_config_file", "path_config_log"
}
```

3.2.5.48 T_

```
double Config_Settings::T_ = 500.0e-15  [private]
```

3.2.5.49 T_description_

```
std::string Config_Settings::T_description_ = "(default) (double) The T value"  [private]
```

3.2.5.50 w_active_max_

```
double Config_Settings::w_active_max_ = 8.0e15  [private]
```

3.2.5.51 w_active_max_description_

```
std::string Config_Settings::w_active_max_description_ = "(default) (double) Maximum angular
frequency"  [private]
```

3.2.5.52 w_active_min_

```
double Config_Settings::w_active_min_ = 2.0e14  [private]
```

3.2.5.53 w_active_min_description_

```
std::string Config_Settings::w_active_min_description_ = "(default) (double) Minimum angular
frequency"  [private]
```

3.2.5.54 waist_

```
double Config_Settings::waist_ = 48.0e-6  [private]
```

3.2.5.55 waist_description_

```
std::string Config_Settings::waist_description_ = "(default) (double) The waist value"  [private]
```

3.2.5.56 Z_

```
double Config_Settings::Z_ = 5.0e-3  [private]
```

3.2.5.57 Z_description_

```
std::string Config_Settings::Z_description_ = "(default) (double) Length of capillary"  [private]
```

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

- /home/sam/Project/XNLO/UPPE/src/config_settings.hpp
- /home/sam/Project/XNLO/UPPE/src/config_settings.cpp

3.3 DHT Class Reference

```
#include <DHT.hpp>
```

Public Member Functions

- DHT ()
- DHT (int n_r_, maths_textbook &maths_)
- Eigen::ArrayXcd forward (Eigen::ArrayXcd f_r_)
- Eigen::ArrayXcd backward (Eigen::ArrayXcd f_kr_)

Private Attributes

- MatrixXcd H

3.3.1 Detailed Description

Originally created by Patrick Anderson. Modified by Samuel Senior on 10/03/2017. "DHT" evaluates the forward and backward discrete Hankel transform. Based on Fisk, Computer Physics Communications, 43 (1987). Complex datatype used here, should really template/overload.

3.3.2 Constructor & Destructor Documentation

3.3.2.1 DHT() [1/2]

```
DHT::DHT ( )
```

Default constructor

3.3.2.2 DHT() [2/2]

```
DHT::DHT (
            int n_r_,
            maths_textbook & maths_ )
```

Parameterized constructor

3.3.3 Member Function Documentation

3.3.3.1 backward()

```
Eigen::ArrayXcd DHT::backward (
            Eigen::ArrayXcd f_kr_ )
```

Backward transform

3.3.3.2 forward()

```
Eigen::ArrayXcd DHT::forward (
            Eigen::ArrayXcd f_r_ )
```

Forward transform

3.3.4 Member Data Documentation

3.3.4.1 H

```
MatrixXcd DHT::H  [private]
```

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

- /home/sam/Project/XNLO/UPPE/src/DHT.hpp
- /home/sam/Project/XNLO/UPPE/src/DHT.cpp

3.4 grid_rkr Class Reference

`#include <grid_rkr.hpp>`

Public Member Functions

- grid_rkr ()
- grid_rkr (int n_r_, double R_, int n_m_, maths_textbook &maths_)

Public Attributes

- ArrayXd r
- ArrayXd kr
- int n_r
- double R
- int n_m

3.4.1 Detailed Description

Originally created by Patrick Anderson. Modified by Samuel Senior on 10/03/2017. "grid_rkr" is a non-uniform radial grid. The spectral counterpart of this grid is evaluated and accessible.

3.4.2 Constructor & Destructor Documentation

3.4.2.1 grid_rkr() [1/2]

```
grid_rkr::grid_rkr ( )
```

Default constructor

3.4.2.2 grid_rkr() [2/2]

```
grid_rkr::grid_rkr (
            int n_r_,
            double R_,
            int n_m_,
            maths_textbook & maths_ )
```

Parameterized constructor

3.4.3 Member Data Documentation

3.4.3.1 kr

```
ArrayXd grid_rkr::kr
```

3.4.3.2 n_m

```
int grid_rkr::n_m
```

3.4.3.3 n_r

```
int grid_rkr::n_r
```

3.4.3.4 r

```
ArrayXd grid_rkr::r
```

3.4.3.5 R

```
double grid_rkr::R
```

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

- /home/sam/Project/XNLO/UPPE/src/grid_rkr.hpp
- /home/sam/Project/XNLO/UPPE/src/grid_rkr.cpp

3.5 grid_tw Class Reference

```
#include <grid_tw.hpp>
```

Public Member Functions

- grid_tw ()
- grid_tw (int N_t_, double T_, double w_active_min_, double w_active_max_, maths_textbook &maths_)

Public Attributes

- ArrayXd t
- ArrayXd w_active
- int n_t
- int n_active
- int w_active_min_index

3.5.1 Detailed Description

Originally created by Patrick Anderson. Modified by Samuel Senior on 10/03/2017. "grid_tw" is a linear temporal grid. The spectral counterpart of this grid is evaluated and made accessible.

3.5.2 Constructor & Destructor Documentation

3.5.2.1 grid_tw() [1/2]

```
grid_tw::grid_tw ( )
```

3.5.2.2 grid_tw() [2/2]

```
grid_tw::grid_tw (
            int n_t_,
            double T_,
            double w_active_min_,
            double w_active_max_,
            maths_textbook & maths_ )
```

Parameterized Constructor

3.5.3 Member Data Documentation

3.5.3.1 n_active

```
int grid_tw::n_active
```

3.5.3.2 n_t

```
int grid_tw::n_t
```

3.5.3.3 t

```
ArrayXd grid_tw::t
```

3.5.3.4 w_active

```
ArrayXd grid_tw::w_active
```

3.5.3.5 w_active_min_index

```
int grid_tw::w_active_min_index
```

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

- /home/sam/Project/XNLO/UPPE/src/grid_tw.hpp
- /home/sam/Project/XNLO/UPPE/src/grid_tw.cpp

3.6 IO Class Reference

```
#include <IO.hpp>
```

Public Member Functions

- IO ()
- Array< unsigned short, Dynamic, Dynamic > read_uint16 (const char *path_, int N_row_, int N_col_)
- ArrayXXi read_int (const char *path_, int N_row_, int N_col_)
- ArrayXXd read_double (const std::string path_, int N_row_, int N_col_)
- ArrayXXd read_ascii_double (const std::string path, int N_row_, int N_col_)
- void write_double (const std::string path_, ArrayXXd input_, int N_row_, int N_col_, bool print=true)
- void write_header (const std::string path_, int N_row_, int N_col_, bool print=true)
- void overwrite (const std::string path, bool print=true)
- void write_ascii_double (ArrayXd data, std::string path, bool print=true)

3.6.1 Detailed Description

Originally created by Patrick Anderson. Modified by Samuel Senior on 10/03/2017. "IO" objects enable reading/writing of binary files to/from Eigen arrays.

3.6.2 Constructor & Destructor Documentation

3.6.2.1 IO()

```
IO::IO ( )
```

Constructor

3.6.3 Member Function Documentation

3.6.3.1 overwrite()

```
void IO::overwrite (
            const std::string path,
            bool print = true )
```

Overwrites given binary file.

3.6.3.2 read_ascii_double()

```
ArrayXXd IO::read_ascii_double (
            const std::string path,
            int N_row_,
            int N_col_ )
```

Read double to Eigen array from ascii file

3.6.3.3 read_double()

```
ArrayXXd IO::read_double (
            const std::string path_,
            int N_row_,
            int N_col_ )
```

Read double to Eigen array from binary file

3.6.3.4 read_int()

```
ArrayXXi IO::read_int (
            const char * path_,
            int N_row_,
            int N_col_ )
```

int

3.6.3.5 read_uint16()

```
Array< unsigned short, Dynamic, Dynamic > IO::read_uint16 (
            const char * path_,
            int N_row_,
            int N_col_ )
```

Read from binary file to N_col_ by N_row_ Eigen array uint16

3.6.3.6 write_ascii_double()

```
void IO::write_ascii_double (
            ArrayXd data,
            std::string path,
            bool print = true )
```

3.6.3.7 write_double()

```
void IO::write_double (
            const std::string path_,
            ArrayXXd input_,
            int N_row_,
            int N_col_,
            bool print = true )
```

Write to binary file from N_col_ by N_row_ Eigen array double

3.6.3.8 write_header()

```
void IO::write_header (
            const std::string path_,
            int N_row_,
            int N_col_,
            bool print = true )
```

Write UPPE binary header to given binary file. Header takes the form: Offset Size (Bytes) Type/Contents Description 0 4 'UPPE' Binary format name 4 4 int Version Number 8 4 int Subversion number 12 4 int Size of header 16 4 int N_row 20 4 int N_col 24 4 int Total size of data 28 4 int Size of each double in the data

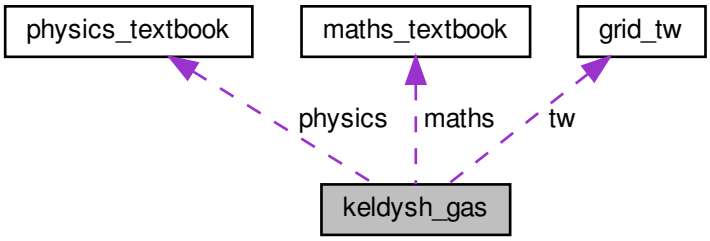
The documentation for this class was generated from the following files:

- /home/sam/Project/XNLO/UPPE/src/IO.hpp
- /home/sam/Project/XNLO/UPPE/src/IO.cpp

3.7 keldysh_gas Class Reference

`#include <keldysh_gas.hpp>`

Collaboration diagram for keldysh_gas:



Public Member Functions

- keldysh_gas (double press_, grid_tw &tw_, DFTI_DESCRIPTOR_HANDLE &ft_, maths_textbook &maths_)
- double atom_density (double z)
- ArrayXcd nl_polarization (ArrayXd E_t_)
- ArrayXd ionization_rate (ArrayXd E_t_)
- ArrayXd electron_density (ArrayXd W_t_, double z)
- ArrayXcd current_density (ArrayXd E_t_, double z)

Public Attributes

- double atom_density_max
- double z_max
- double inlet_1
- double inlet_2
- double transitionLength
- double U
- double C_kl
- double n_star
- double kappa

Private Attributes

- physics_textbook physics
- maths_textbook maths
- grid_tw tw
- DFTI_DESCRIPTOR_HANDLE ft

3.7.1 Detailed Description

Originally created by Patrick Anderson. Modified by Samuel Senior on 10/03/2017. "keldysh_gas" contains the medium response model.

3.7.2 Constructor & Destructor Documentation

3.7.2.1 keldysh_gas()

```
keldysh_gas::keldysh_gas (
            double press_,
            grid_tw & tw_,
            DFTI_DESCRIPTOR_HANDLE & ft_,
            maths_textbook & maths_ )
```

Constructor

3.7.3 Member Function Documentation

3.7.3.1 atom_density()

```
double keldysh_gas::atom_density (
            double z )
```

3.7.3.2 current_density()

```
ArrayXcd keldysh_gas::current_density (
            ArrayXd E_t_,
            double z )
```

Evaluate Current density for active frequencies

3.7.3.3 electron_density()

```
ArrayXd keldysh_gas::electron_density (
            ArrayXd W_t_,
            double z )
```

Calculate free electron density (solve rate equations)

3.7.3.4 ionization_rate()

```
ArrayXd keldysh_gas::ionization_rate (
            ArrayXd E_t_ )
```

Calulate ionization rate (Popov, 2004)

3.7.3.5 nl_polarization()

```
ArrayXcd keldysh_gas::nl_polarization (
            ArrayXd E_t_ )
```

Evaluate nonlinear polarization for active frequencies

3.7.4 Member Data Documentation

3.7.4.1 atom_density_max

```
double keldysh_gas::atom_density_max
```

3.7.4.2 C_kl

```
double keldysh_gas::C_kl
```

3.7.4.3 ft

```
DFTI_DESCRIPTOR_HANDLE keldysh_gas::ft  [private]
```

3.7.4.4 inlet_1

```
double keldysh_gas::inlet_1
```

3.7.4.5 inlet_2

```
double keldysh_gas::inlet_2
```

3.7.4.6 kappa

```
double keldysh_gas::kappa
```

3.7.4.7 maths

```
maths_textbook keldysh_gas::maths  [private]
```

3.7.4.8 n_star

```
double keldysh_gas::n_star
```

3.7.4.9 physics

```
physics_textbook keldysh_gas::physics  [private]
```

3.7.4.10 transitionLength

```
double keldysh_gas::transitionLength
```

3.7.4.11 tw

```
grid_tw keldysh_gas::tw  [private]
```

3.7.4.12 U

```
double keldysh_gas::U
```

3.7.4.13 z_max

```
double keldysh_gas::z_max
```

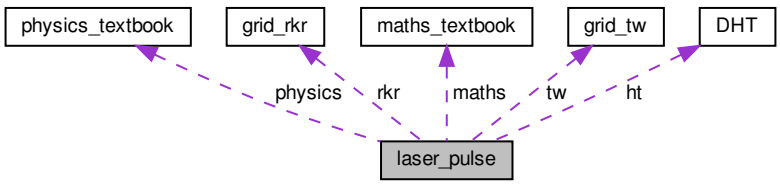
The documentation for this class was generated from the following files:

- /home/sam/Project/XNLO/UPPE/src/keldysh_gas.hpp
- /home/sam/Project/XNLO/UPPE/src/keldysh_gas.cpp

3.8 laser_pulse Class Reference

```
#include <laser_pulse.hpp>
```

Collaboration diagram for laser_pulse:



Public Member Functions

- laser_pulse (double p_av_, double rep_, double fwhm_, double l_0_, double ceo_, double waist_, grid_tw &tw_, grid_rkr &rkr_, DFTI_DESCRIPTOR_HANDLE &ft_, DHT &ht_, maths_textbook &maths_)
- void propagate (double dz_, capillary_fibre &capillary_, keldysh_gas &gas_)

Public Attributes

- ArrayXXcd A_w_active
- ArrayXXcd P_NL_r_t
- ArrayXXcd P_NL_m_t
- ArrayXXcd P_NL_w
- ArrayXXd electron_density
- double atom_density_max
- double p_pk
- double E_pk

Private Member Functions

- void RK_F_45 (double dz_, capillary_fibre &capillary_, keldysh_gas &gas_)
- ArrayXXcd RHS_UPPE (double dz_, ArrayXXcd A_w_active, capillary_fibre &capillary_, keldysh_gas &gas↩_)

Private Attributes

- double p_av
- double rep
- double fwhm
- double l_0
- double ceo
- double waist
- physics_textbook physics
- maths_textbook maths
- grid_tw tw
- grid_rkr rkr
- DFTI_DESCRIPTOR_HANDLE ft
- DHT ht
- ArrayXXcd Y_4
- ArrayXXcd Y_5
- double e
- double z_position

3.8.1 Detailed Description

Originally created by Patrick Anderson. Modified by Samuel Senior on 10/03/2017. "laser_pulse" contains the active spectral amplitudes and governs their propagation over longitudinal step dz.

3.8.2 Constructor & Destructor Documentation

3.8.2.1 laser_pulse()

```
laser_pulse::laser_pulse (
            double p_av_,
            double rep_,
            double fwhm_,
            double l_0_,
            double ceo_,
            double waist_,
            grid_tw & tw_,
            grid_rkr & rkr_,
            DFTI_DESCRIPTOR_HANDLE & ft_,
            DHT & ht_,
            maths_textbook & maths_ )
```

Constructor

3.8.3 Member Function Documentation

3.8.3.1 propagate()

```
void laser_pulse::propagate (
            double dz_,
            capillary_fibre & capillary_,
            keldysh_gas & gas_ )
```

Propagate sprectral amplitudes A_w_active over the longitudinal step dz_, based on A. Couairon, et al., Eur. Phys. J. Special Topics, 199, 5 (2011)

3.8.3.2 RHS_UPPE()

```
ArrayXXcd laser_pulse::RHS_UPPE (
            double dz_,
            ArrayXXcd temp_1,
            capillary_fibre & capillary_,
            keldysh_gas & gas_ )  [private]
```

Evaluate RHS of UPPE

3.8.3.3 RK_F_45()

```
void laser_pulse::RK_F_45 (
            double h_,
            capillary_fibre & capillary_,
            keldysh_gas & gas_ )  [private]
```

Runge-Kutta-Fehlberg 45 ODE solver

3.8.4 Member Data Documentation

3.8.4.1 A_w_active

```
ArrayXXcd laser_pulse::A_w_active
```

Spectral amplitudes

3.8.4.2 atom_density_max

```
double laser_pulse::atom_density_max
```

3.8.4.3 ceo

```
double laser_pulse::ceo  [private]
```

3.8.4.4 e

```
double laser_pulse::e  [private]
```

Error in RKF45 solve

3.8.4.5 E_pk

```
double laser_pulse::E_pk
```

3.8.4.6 electron_density

```
ArrayXXd laser_pulse::electron_density
```

3.8.4.7 ft

```
DFTI_DESCRIPTOR_HANDLE laser_pulse::ft  [private]
```

Fourier transform

3.8.4.8 fwhm

```
double laser_pulse::fwhm  [private]
```

3.8.4.9 ht

```
DHT laser_pulse::ht  [private]
```

Hankel transform

3.8.4.10 I_0

```
double laser_pulse::I_0  [private]
```

3.8.4.11 maths

```
maths_textbook laser_pulse::maths  [private]
```

Mathematical constants and functions

3.8.4.12 p_av

```
double laser_pulse::p_av  [private]
```

3.8.4.13 P_NL_m_t

```
ArrayXXcd laser_pulse::P_NL_m_t
```

3.8.4.14 P_NL_r_t

```
ArrayXXcd laser_pulse::P_NL_r_t
```

3.8.4.15 P_NL_w

```
ArrayXXcd laser_pulse::P_NL_w
```

3.8.4.16 p_pk

```
double laser_pulse::p_pk
```

3.8.4.17 physics

```
physics_textbook laser_pulse::physics  [private]
```

Physical constants

3.8.4.18 rep

```
double laser_pulse::rep  [private]
```

3.8.4.19 rkr

```
grid_rkr laser_pulse::rkr  [private]
```

Radial grid

3.8.4.20 tw

```
grid_tw laser_pulse::tw  [private]
```

Temperal grid

3.8.4.21 waist

```
double laser_pulse::waist  [private]
```

3.8.4.22 Y_4

```
ArrayXXcd laser_pulse::Y_4  [private]
```

RKF 4

3.8.4.23 Y_5

```
ArrayXXcd laser_pulse::Y_5  [private]
```

RKF 5

3.8.4.24 z_position

```
double laser_pulse::z_position  [private]
```

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

- /home/sam/Project/XNLO/UPPE/src/laser_pulse.hpp
- /home/sam/Project/XNLO/UPPE/src/laser_pulse.cpp

3.9 maths_textbook Class Reference

```
#include <maths_textbook.hpp>
```

Public Member Functions

- maths_textbook (std::string path_input_j0_)
- double trapz (ArrayXd x_, ArrayXd y_)
- ArrayXd cumtrapz (ArrayXd x_, ArrayXd y_)

Public Attributes

- double pi
- ArrayXd J0_zeros

Private Attributes

- std::string path_input_j0

3.9.1 Detailed Description

Originally created by Patrick Anderson. Modified by Samuel Senior on 10/03/2017. "maths_textbook" is a container for mathematical constants and functions.

3.9.2 Constructor & Destructor Documentation

3.9.2.1 maths_textbook()

```
maths_textbook::maths_textbook (
            std::string path_input_j0_ )
```

Constructor

3.9.3 Member Function Documentation

3.9.3.1 cumtrapz()

```
ArrayXd maths_textbook::cumtrapz (
            ArrayXd x_,
            ArrayXd y_ )
```

Cumulative trapezoidal integration

3.9.3.2 trapz()

```
double maths_textbook::trapz (
            ArrayXd x_,
            ArrayXd y_ )
```

Trapezoidal integration

3.9.4 Member Data Documentation

3.9.4.1 J0_zeros

```
ArrayXd maths_textbook::J0_zeros
```

3.9.4.2 path_input_j0

```
std::string maths_textbook::path_input_j0  [private]
```

3.9.4.3 pi

```
double maths_textbook::pi
```

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

- /home/sam/Project/XNLO/UPPE/src/maths_textbook.hpp
- /home/sam/Project/XNLO/UPPE/src/maths_textbook.cpp

3.10 physics_textbook Class Reference

```
#include <physics_textbook.hpp>
```

Public Member Functions

- physics_textbook ()

Public Attributes

- double E_at
- double l_at
- double m_at
- double q_at
- double t_at
- double w_at
- double c
- double eps_0
- double mu_0
- double h_bar
- double k_B

3.10.1 Detailed Description

Modified by Patrick Anderson on 03/09/2015. "physics_textbook" is a container for physical constants.

3.10.2 Constructor & Destructor Documentation

3.10.2.1 physics_textbook()

```
physics_textbook::physics_textbook ( )
```

Constructor

3.10.3 Member Data Documentation

3.10.3.1 c

```
double physics_textbook::c
```

Speed of light in vacuum

3.10.3.2 E_at

```
double physics_textbook::E_at
```

3.10.3.3 eps_0

```
double physics_textbook::eps_0
```

Permitivity of free space

3.10.3.4 h_bar

```
double physics_textbook::h_bar
```

Reduced Planck constant

3.10.3.5 k_B

```
double physics_textbook::k_B
```

Boltzmann Constant

3.10.3.6 l_at

```
double physics_textbook::l_at
```

Bohr radius

3.10.3.7 m_at

```
double physics_textbook::m_at
```

3.10.3.8 mu_0

```
double physics_textbook::mu_0
```

Permeability of free space

3.10.3.9 q_at

```
double physics_textbook::q_at
```

Electron charge

3.10.3.10 t_at

```
double physics_textbook::t_at
```

3.10.3.11 w_at

```
double physics_textbook::w_at
```

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

- /home/sam/Project/XNLO/UPPE/src/physics_textbook.hpp
- /home/sam/Project/XNLO/UPPE/src/physics_textbook.cpp

Chapter 4

File Documentation

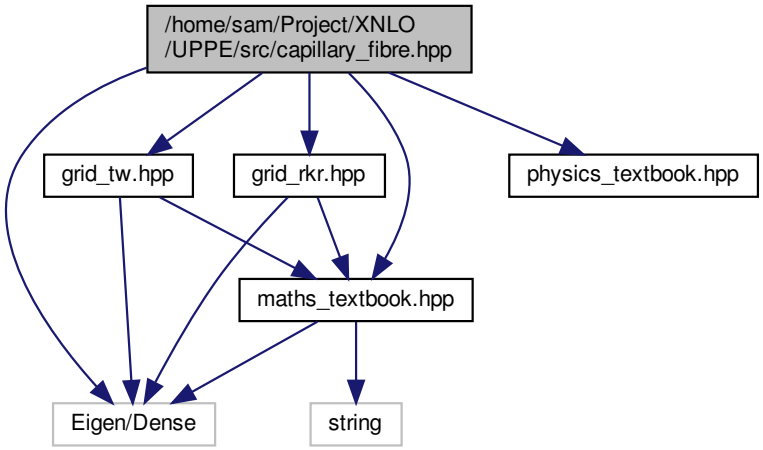
4.1 /home/sam/Project/XNLO/UPPE/src/capillary_fibre.cpp File Reference

```
#include "capillary_fibre.hpp"
#include "grid_rkr.hpp"
#include "grid_tw.hpp"
#include "physics_textbook.hpp"
#include "maths_textbook.hpp"
#include "Eigen/Dense"
```

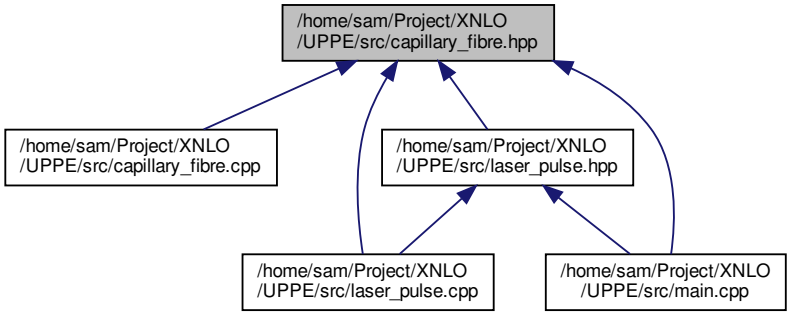
4.2 /home/sam/Project/XNLO/UPPE/src/capillary_fibre.hpp File Reference

```
#include "Eigen/Dense"
#include "grid_tw.hpp"
#include "grid_rkr.hpp"
#include "physics_textbook.hpp"
#include "maths_textbook.hpp"
```

Include dependency graph for capillary_fibre.hpp:



This graph shows which files directly or indirectly include this file:



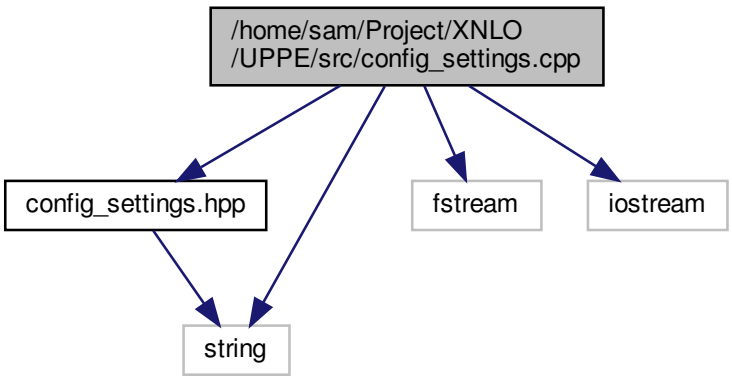
Classes

- class capillary_fibre

4.3 /home/sam/Project/XNLO/UPPE/src/config_settings.cpp File Reference

```
#include "config_settings.hpp"
#include <fstream>
#include <iostream>
#include <string>
```

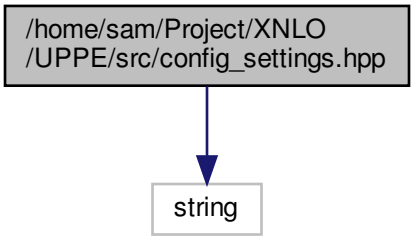
Include dependency graph for config_settings.cpp:



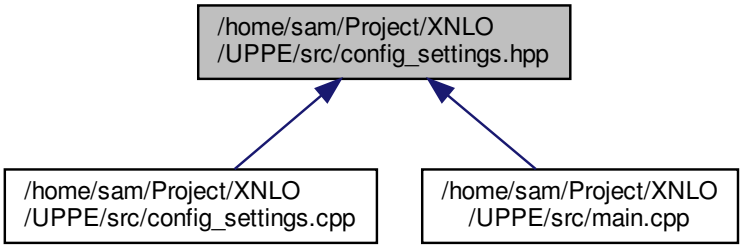
4.4 /home/sam/Project/XNLO/UPPE/src/config_settings.hpp File Reference

```
#include <string>
```

Include dependency graph for config_settings.hpp:



This graph shows which files directly or indirectly include this file:



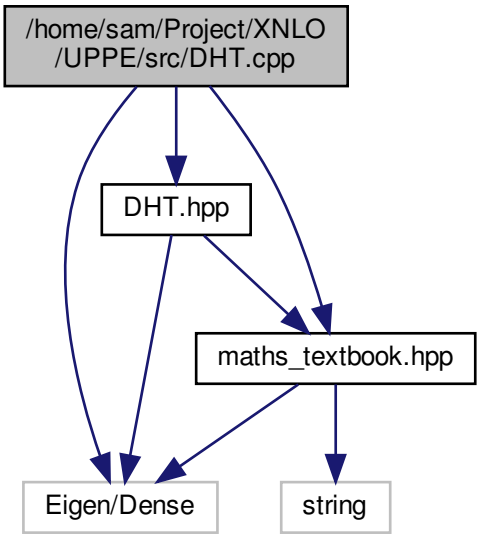
Classes

- class Config_Settings

4.5 /home/sam/Project/XNLO/UPPE/src/DHT.cpp File Reference

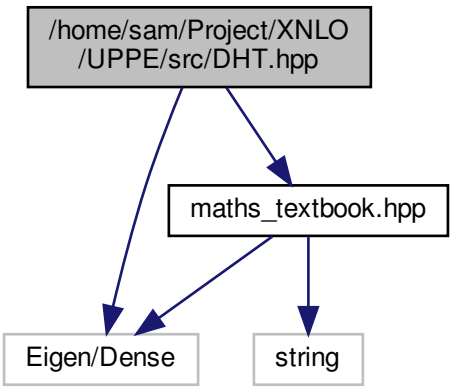
```
#include "DHT.hpp"
#include "Eigen/Dense"
```

```
#include "maths_textbook.hpp"
```
Include dependency graph for DHT.cpp:

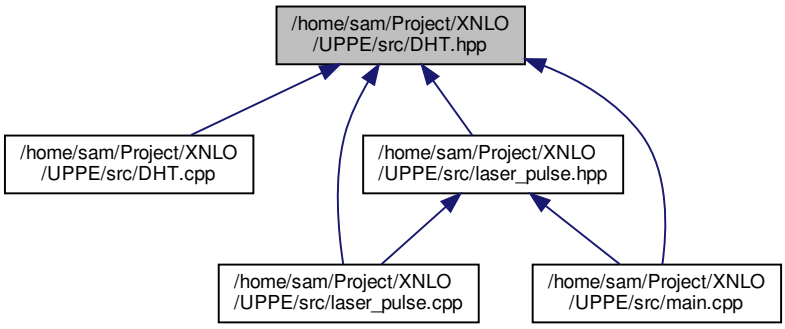


4.6 /home/sam/Project/XNLO/UPPE/src/DHT.hpp File Reference

```
#include "Eigen/Dense"
#include "maths_textbook.hpp"
```
Include dependency graph for DHT.hpp:



This graph shows which files directly or indirectly include this file:



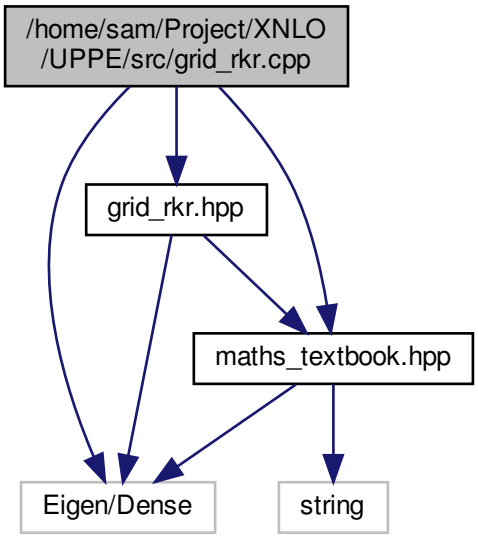
Classes

- class DHT

4.7 /home/sam/Project/XNLO/UPPE/src/grid_rkr.cpp File Reference

```
#include "grid_rkr.hpp"
#include "maths_textbook.hpp"
#include "Eigen/Dense"
```

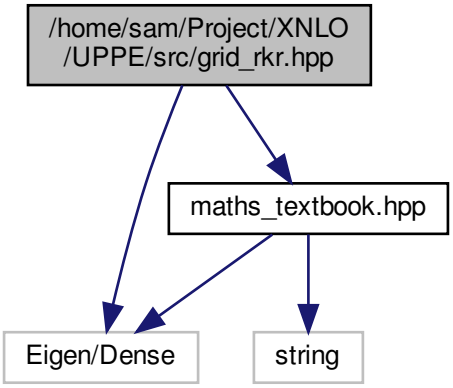
Include dependency graph for grid_rkr.cpp:



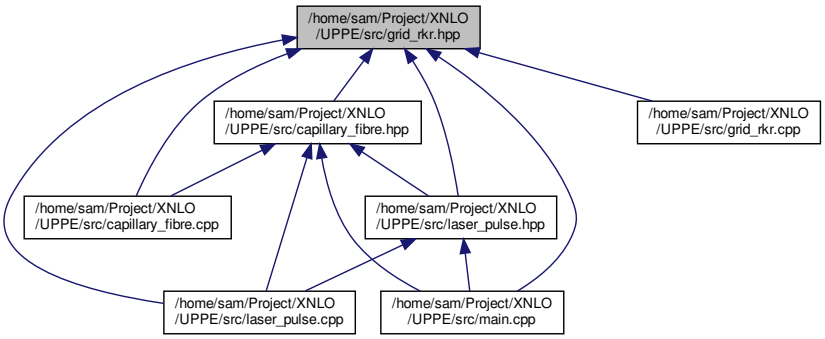
4.8 /home/sam/Project/XNLO/UPPE/src/grid_rkr.hpp File Reference

```
#include "Eigen/Dense"
#include "maths_textbook.hpp"
```

Include dependency graph for grid_rkr.hpp:



This graph shows which files directly or indirectly include this file:



Classes

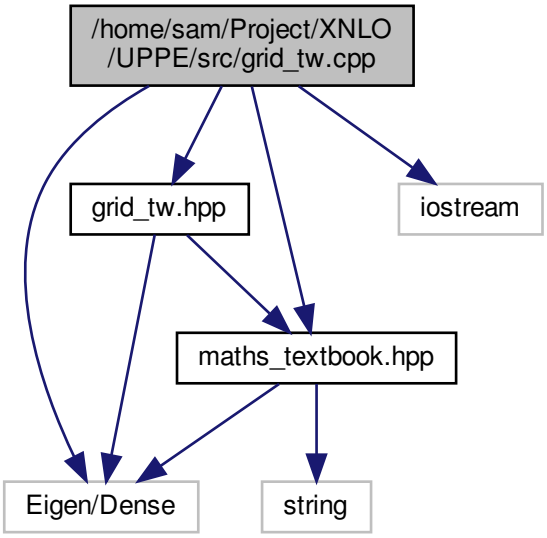
- class grid_rkr

4.9 /home/sam/Project/XNLO/UPPE/src/grid_tw.cpp File Reference

```
#include "grid_tw.hpp"
#include "maths_textbook.hpp"
```

```
#include "Eigen/Dense"
#include <iostream>
```

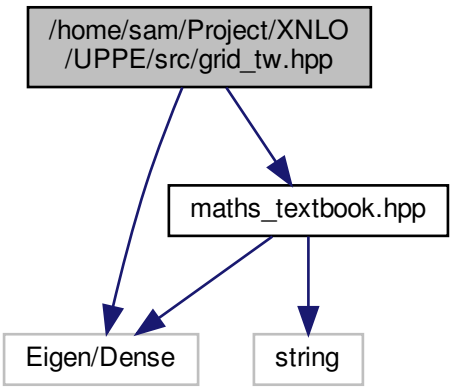
Include dependency graph for grid_tw.cpp:



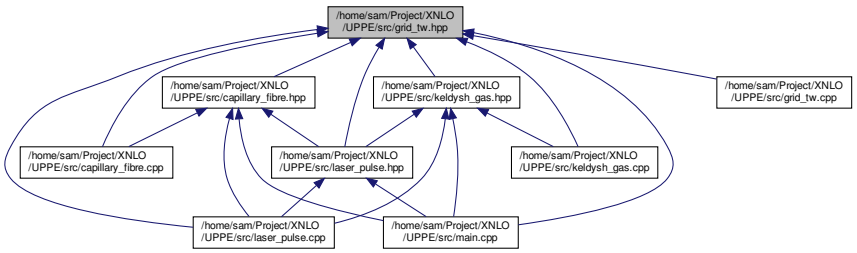
4.10 /home/sam/Project/XNLO/UPPE/src/grid_tw.hpp File Reference

```
#include "Eigen/Dense"
#include "maths_textbook.hpp"
```

Include dependency graph for grid_tw.hpp:



This graph shows which files directly or indirectly include this file:



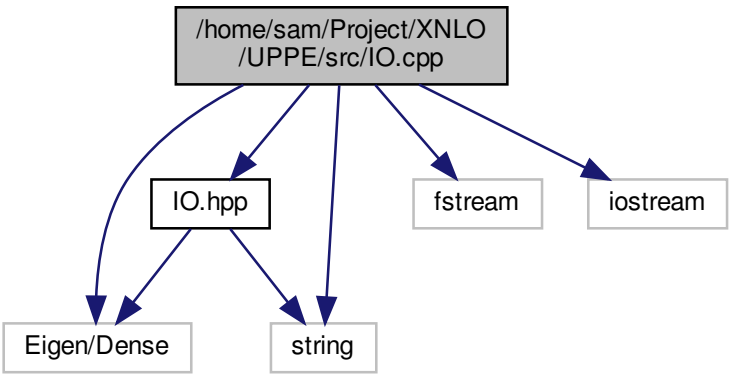
Classes

- class grid_tw

4.11 /home/sam/Project/XNLO/UPPE/src/IO.cpp File Reference

```
#include "IO.hpp"
#include "Eigen/Dense"
#include <fstream>
#include <iostream>
#include <string>
```

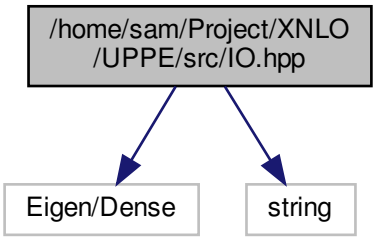
Include dependency graph for IO.cpp:



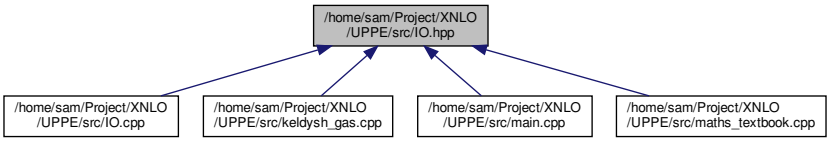
4.12 /home/sam/Project/XNLO/UPPE/src/IO.hpp File Reference

```
#include "Eigen/Dense"
#include <string>
```

Include dependency graph for IO.hpp:



This graph shows which files directly or indirectly include this file:



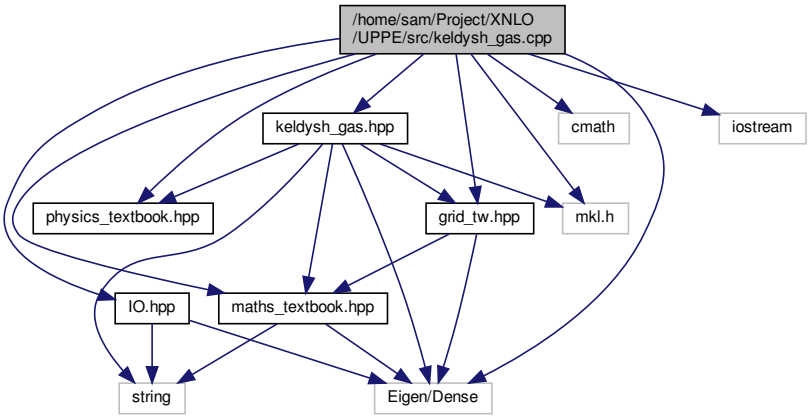
Classes

- class IO

4.13 /home/sam/Project/XNLO/UPPE/src/keldysh_gas.cpp File Reference

```
#include "keldysh_gas.hpp"
#include "physics_textbook.hpp"
#include "grid_tw.hpp"
#include <mkl.h>
#include "Eigen/Dense"
#include "maths_textbook.hpp"
#include <cmath>
#include "IO.hpp"
#include <iostream>
```

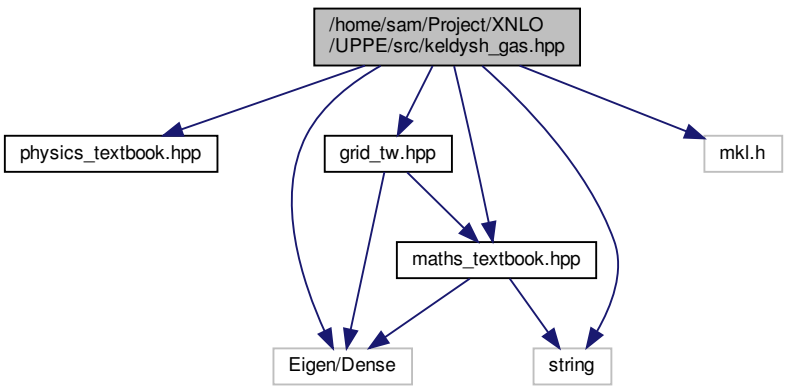
Include dependency graph for keldysh_gas.cpp:



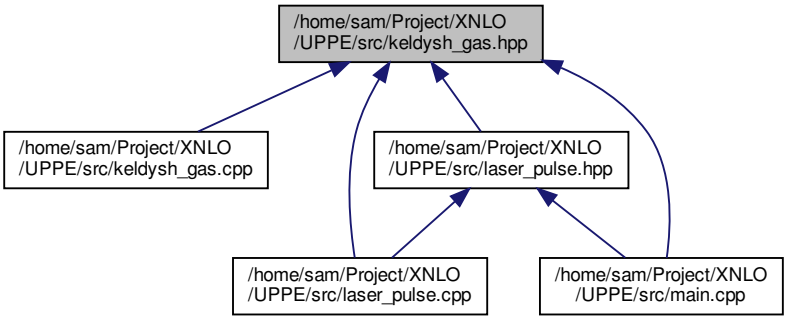
4.14 /home/sam/Project/XNLO/UPPE/src/keldysh_gas.hpp File Reference

```
#include "physics_textbook.hpp"
#include "maths_textbook.hpp"
#include "grid_tw.hpp"
#include <mkl.h>
#include "Eigen/Dense"
#include <string>
```

Include dependency graph for keldysh_gas.hpp:



This graph shows which files directly or indirectly include this file:



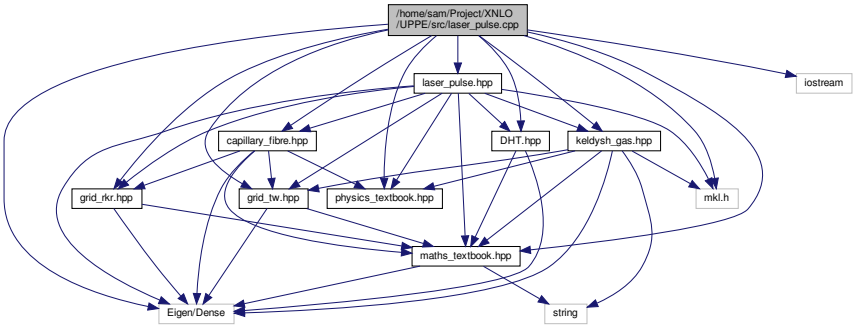
Classes

- class keldysh_gas

4.15 /home/sam/Project/XNLO/UPPE/src/laser_pulse.cpp File Reference

```
#include "laser_pulse.hpp"
#include "grid_tw.hpp"
#include "grid_rkr.hpp"
#include <mkl.h>
#include "DHT.hpp"
#include "physics_textbook.hpp"
#include "maths_textbook.hpp"
#include "Eigen/Dense"
#include "capillary_fibre.hpp"
#include "keldysh_gas.hpp"
#include <iostream>
```

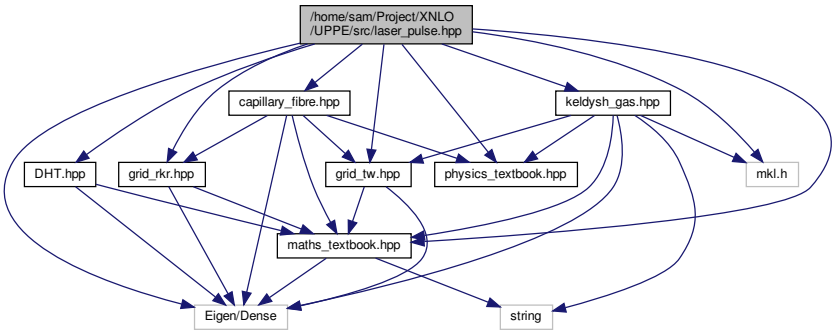
Include dependency graph for laser_pulse.cpp:



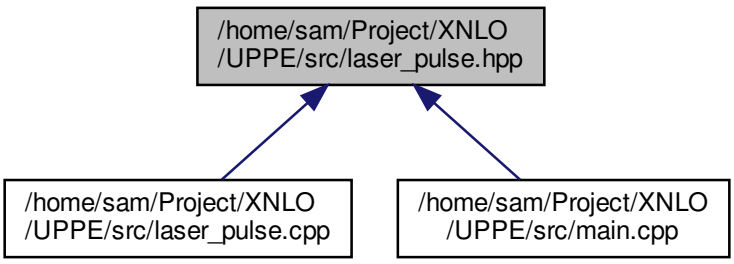
4.16 /home/sam/Project/XNLO/UPPE/src/laser_pulse.hpp File Reference

```
#include "physics_textbook.hpp"
#include "maths_textbook.hpp"
#include "grid_tw.hpp"
#include "grid_rkr.hpp"
#include <mkl.h>
#include "DHT.hpp"
#include "Eigen/Dense"
#include "capillary_fibre.hpp"
#include "keldysh_gas.hpp"
```

Include dependency graph for laser_pulse.hpp:



This graph shows which files directly or indirectly include this file:



Classes

- class laser_pulse

4.17 /home/sam/Project/XNLO/UPPE/src/main.cpp File Reference

```
#include <mpi.h>
#include "maths_textbook.hpp"
#include "physics_textbook.hpp"
#include <mkl.h>
#include "DHT.hpp"
#include "grid_rkr.hpp"
#include "grid_tw.hpp"
#include "laser_pulse.hpp"
#include "capillary_fibre.hpp"
#include "keldysh_gas.hpp"
#include "Eigen/Dense"
#include "IO.hpp"
#include "config_settings.hpp"
#include <iostream>
#include <string>
#include "../../XNLO/lib/XNLO.hpp"
```

Functions

- int main (int argc, char **argv)

4.17.1 Function Documentation

4.17.1.1 main()

```
int main (
            int argc,
            char ** argv )
```

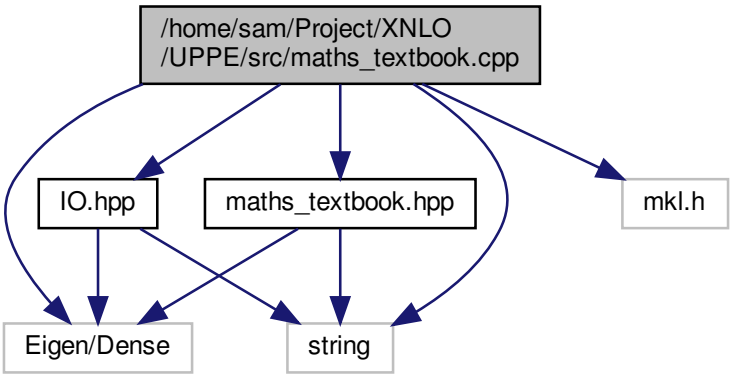
Originally created by Patrick Anderson. Modified by Samuel Senior on 10/03/2017. Test enviroment for UPPE codes.

4.18 /home/sam/Project/XNLO/UPPE/src/maths_textbook.cpp File Reference

```
#include "maths_textbook.hpp"
#include "IO.hpp"
#include "Eigen/Dense"
#include <mkl.h>
```

```
#include <string>
```

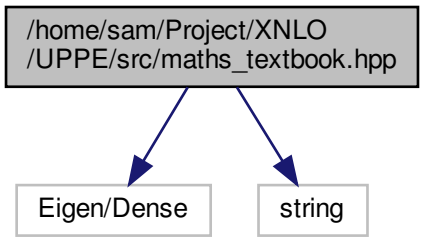
Include dependency graph for maths_textbook.cpp:



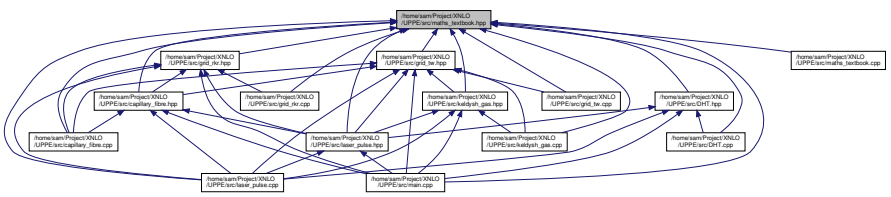
4.19 /home/sam/Project/XNLO/UPPE/src/maths_textbook.hpp File Reference

```
#include "Eigen/Dense"
#include <string>
```

Include dependency graph for maths_textbook.hpp:



This graph shows which files directly or indirectly include this file:

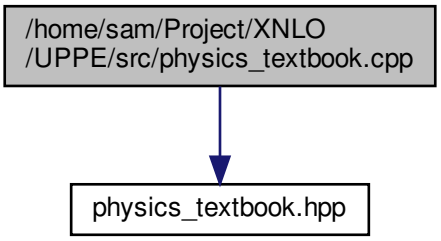


Classes

- class maths_textbook

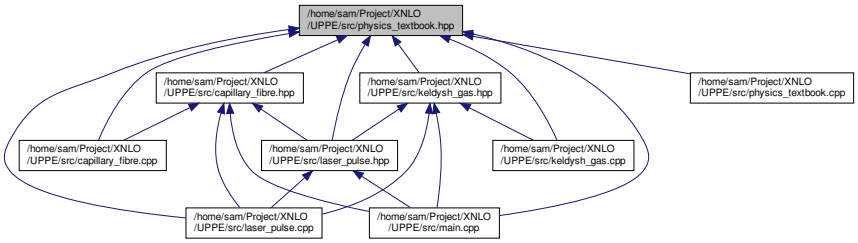
4.20 /home/sam/Project/XNLO/UPPE/src/physics_textbook.cpp File Reference

```
#include "physics_textbook.hpp"
```
Include dependency graph for physics_textbook.cpp:



4.21 /home/sam/Project/XNLO/UPPE/src/physics_textbook.hpp File Reference

This graph shows which files directly or indirectly include this file:



Classes

- class physics_textbook

4.22 /home/sam/Project/XNLO/UPPE/src/version.hpp File Reference

Macros

- #define _VERSION_MAJOR 1
- #define _VERSION_MINOR 3
- #define _VERSION_SUBMINOR 0

4.22.1 Macro Definition Documentation

4.22.1.1 _VERSION_MAJOR

```
#define _VERSION_MAJOR 1
```

4.22.1.2 _VERSION_MINOR

```
#define _VERSION_MINOR 3
```

4.22.1.3 _VERSION_SUBMINOR

```
#define _VERSION_SUBMINOR 0
```

Index