# Test driven development

Generated by Doxygen 1.9.3

# 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 PidController Class Reference

PID Controller Class, initialization of gain values and sampling time.

`#include <PidController.hpp>`

### Public Member Functions

- PidController ()

  *Class constructor to initialize with default values.*
- PidController (double Kp, double Ki, double Kd, double dt)

  *Constructor to initialize with user defined values.*
- double Compute (double targetState)

  *This method computes the magnitude of feedback for the system based on the current state of the system and the target setpoint set by the user.*
- double getdt ()

  *This method returns the sampling time dt.*
- ~PidController ()

### 3.1.1 Detailed Description

PID Controller Class, initialization of gain values and sampling time.

Definition at line 26 of file PidController.hpp.

### 3.1.2 Constructor & Destructor Documentation

#### 3.1.2.1 PidController() [1/2]

```
PidController::PidController ( )  [inline]
```

Class constructor to initialize with default values.

Definition at line 38 of file PidController.hpp.

#### 3.1.2.2 PidController() [2/2]

```
PidController::PidController (
            double Kp,
            double Ki,
            double Kd,
            double dt )
```

Constructor to initialize with user defined values.

**Parameters**

| | |
|---|---|
| | Kp is the value of proportional constant entered by user |
| | Ki is the value of integral constant entered by user |
| | Kd is the value of derivative constant entered by user |
| | dt is the value sample time entered by user |

#### 3.1.2.3 ~PidController()

```
PidController::~PidController ( )
```

### 3.1.3 Member Function Documentation

#### 3.1.3.1 Compute()

```
double PidController::Compute (
            double targetState )
```

This method computes the magnitude of feedback for the system based on the current state of the system and the target setpoint set by the user.

**Parameters**

| | |
|---|---|
| *targetState* | is the target state which needs to be achieved. |

#### 3.1.3.2 getdt()

```
double PidController::getdt ( )
```

This method returns the sampling time dt.

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

- /home/aswath/808x_TestDrivenDevelopment/TestDrivenDev/include/PidController.hpp

# Chapter 4

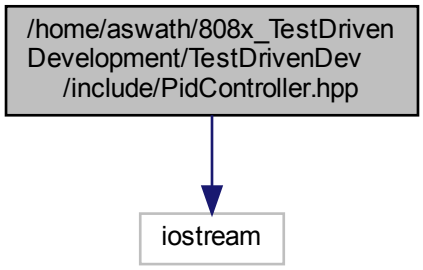
# File Documentation

## 4.1 /home/aswath/808x_TestDrivenDevelopment/TestDrivenDev/include/PidController.hpp File Reference

PidController class.

```
#include <iostream>
```

Include dependency graph for PidController.hpp:




### Classes

- class PidController

  *PID Controller Class, initialization of gain values and sampling time.*

### 4.1.1 Detailed Description

PidController class.

**Author**

Aswath Muthuselvam

Part 1 :Aswath Muthuselvam (Driver) , Prannoy Namala (Navigator)

**Date**

10/1/2021

**Version**

1.0

### 4.1.2 DESCRIPTION

This hpp file defines the class and methods for a PID Controller.

Definition in file PidController.hpp.

## 4.2 PidController.hpp

Go to the documentation of this file.

```
00001 // Copyright 2021 Aswath Muthuselvam
00017 #ifndef INCLUDE_PIDCONTROLLER_HPP_
00018 #define INCLUDE_PIDCONTROLLER_HPP_
00019
00020 #include <iostream>
00021
00026 class PidController {
00027  private:
00028   double kp_ = 1.0;
00029   double ki_ = 0.0;
00030   double kd_ = 0.0;
00031   double dt_ = 0.1;
00032   double currentState_ = 0;
00033
00034  public:
00038   PidController(){};
00039
00048   PidController(double Kp, double Ki, double Kd, double dt);
00049
00055   double Compute(double targetState);
00056
00060   double getdt();
00061
00062   ~PidController();
00063 };
00064
00065 #endif  // INCLUDE_PIDCONTROLLER_HPP_
```

# Index