

PD_Controller

Generated by Doxygen 1.8.20

1 Class Index	1
1.1 Class List	1
2 File Index	3
2.1 File List	3
3 Class Documentation	5
3.1 Constants.Constants Class Reference	5
4 File Documentation	7
4.1 Calculations.py File Reference	7
4.1.1 Detailed Description	7
4.1.2 Function Documentation	7
4.1.2.1 func_y_t()	7
4.2 Constants.py File Reference	8
4.2.1 Detailed Description	8
4.3 Control.py File Reference	8
4.3.1 Detailed Description	9
4.4 InputParameters.py File Reference	9
4.4.1 Detailed Description	9
4.4.2 Function Documentation	9
4.4.2.1 input_constraints()	10
4.5 OutputFormat.py File Reference	10
4.5.1 Detailed Description	10
Index	11

Chapter 1

Class Index

1.1 Class List

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

Constants.Constants	5
---	---

Chapter 2

File Index

2.1 File List

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

Calculations.py	Provides functions for calculating the outputs	7
Constants.py	Provides the structure for holding constant values	8
Control.py	Controls the flow of the program	8
InputParameters.py	Provides the function for reading inputs and the function for checking the physical constraints on the input	9
OutputFormat.py	Provides the function for writing outputs	10

Chapter 3

Class Documentation

3.1 Constants.Constants Class Reference

Static Public Attributes

- float **AbsTol** = 1.0e-10
- float **RelTol** = 1.0e-10

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

- [Constants.py](#)

Chapter 4

File Documentation

4.1 Calculations.py File Reference

Provides functions for calculating the outputs.

Functions

- def `Calculations.func_y_t` (`r_t`, `K_p`, `K_d`, `t_sim`, `t_step`)
Calculates Process Variable: The output value from the power plant.

4.1.1 Detailed Description

Provides functions for calculating the outputs.

Author

Naveen Ganesh Muralidharan

4.1.2 Function Documentation

4.1.2.1 `func_y_t()`

```
def Calculations.func_y_t (  
    r_t,  
    K_p,  
    K_d,  
    t_sim,  
    t_step )
```

Calculates Process Variable: The output value from the power plant.

Parameters

r_t	Set Point: The desired value that the control system must reach. This also knows as reference variable
K_p	Proportional Gain: Gain constant of the proportional controller
K_d	Derivative Gain: Gain constant of the derivative controller
t_{sim}	Simulation Time: Total execution time of the PD simulation (s)
t_{step}	Step Time: Simulation step time (s)

Returns

Process Variable: The output value from the power plant

4.2 Constants.py File Reference

Provides the structure for holding constant values.

Classes

- class [Constants.Constants](#)

4.2.1 Detailed Description

Provides the structure for holding constant values.

Author

Naveen Ganesh Muralidharan

Structure for holding the constant values

4.3 Control.py File Reference

Controls the flow of the program.

Variables

- **Control.filename** = sys.argv[1]
- **Control.outfile** = open("log.txt", "a")
- **Control.end**
- **Control.file**
- **Control.r_t**
- **Control.K_d**
- **Control.K_p**
- **Control.t_step**
- **Control.t_sim**
- **Control.y_t** = Calculations.func_y_t(r_t, K_p, K_d, t_sim, t_step)

4.3.1 Detailed Description

Controls the flow of the program.

Author

Naveen Ganesh Muralidharan

4.4 InputParameters.py File Reference

Provides the function for reading inputs and the function for checking the physical constraints on the input.

Functions

- def `InputParameters.get_input` (filename)
- def `InputParameters.input_constraints` (r_t, K_d, K_p, t_step, t_sim)
Verifies that input values satisfy the physical constraints.

4.4.1 Detailed Description

Provides the function for reading inputs and the function for checking the physical constraints on the input.

Author

Naveen Ganesh Muralidharan

Reads input from a file with the given file name

Parameters

<i>filename</i>	name of the input file
-----------------	------------------------

Returns

Set Point: The desired value that the control system must reach. This also knows as reference variable

Derivative Gain: Gain constant of the derivative controller

Proportional Gain: Gain constant of the proportional controller

Step Time: Simulation step time (s)

Simulation Time: Total execution time of the PD simulation (s)

4.4.2 Function Documentation

4.4.2.1 input_constraints()

```
def InputParameters.input_constraints (
    r_t,
    K_d,
    K_p,
    t_step,
    t_sim )
```

Verifies that input values satisfy the physical constraints.

Parameters

r_t	Set Point: The desired value that the control system must reach. This also knows as reference variable
K_d	Derivative Gain: Gain constant of the derivative controller
K_p	Proportional Gain: Gain constant of the proportional controller
t_{step}	Step Time: Simulation step time (s)
t_{sim}	Simulation Time: Total execution time of the PD simulation (s)

4.5 OutputFormat.py File Reference

Provides the function for writing outputs.

Functions

- def **OutputFormat.write_output** (y_t)

4.5.1 Detailed Description

Provides the function for writing outputs.

Author

Naveen Ganesh Muralidharan

Writes the output values to output.txt

Parameters

$y \leftarrow t$	Process Variable: The output value from the power plant
------------------	---

Index

Calculations.py, [7](#)

 func_y_t, [7](#)

Constants.Constants, [5](#)

Constants.py, [8](#)

Control.py, [8](#)

func_y_t

 Calculations.py, [7](#)

input_constraints

 InputParameters.py, [9](#)

InputParameters.py, [9](#)

 input_constraints, [9](#)

OutputFormat.py, [10](#)