Axel-probe

2.1.0.23

by Teodor Krastev for Imperial College London

Chapter 1

Axel-probe Introduction

Axel Probe is designed as simulator of MotMaster2 application including the quantum (MOT) accelerometer. The main purpose is to test Axel Hub for communication, logging, visualizing and data processing abilities. The software provides:

- The simulation starts with generation of an acceleration pattern (repeatable). Then Axel Probe would simulate a signal which would come out of MotMaster2 and the MOT experiment and send that signal to Axel Hub
- A number behavioural patterns are available, as well as adding Gaussian noise and variety of simulated signal disturbances as amplitude variation, etc.
- In case of active feedback of Raman phase in order to follow the position of a fringe (from the atomic interferometer), Axel Probe will take into account the fed back Raman phase when the signal is simulated. For example: the PID algorithm (with pi flip) could be tested and optimized that way.
- Most of the intermediate and resulting value of calculation (simulation) are visible in charts and some of them in the communication log.

Chapter 2

Namespace Index

2.1 Packages

Here are the packages with brief descriptions (if available):

Chapter 3

Hierarchical Index

3.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 4

Class Index

4.1 Class List

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

Chapter 5

File Index

5.1 File List

Here is a list of all files with brief descriptions:

Chapter 6

Namespace Documentation

6.1 Axel_probe Namespace Reference

Namespaces

- Properties

Classes

- class App

  Interaction logic for App.xaml
- class Kalman

  Under development
- class MainWindow

  Axel-probe is replacement simulator for MotMaster2 generating signal according to a settings corresponding to some experimental conditions
- class PID

  Under development
- class ProbeEngine

  An calculating engine for generating a fringes patterns according simulated experimental conditions the engine runs by its timer and behaives similar to Axel-tilt its state evolve according to the set parameters (accel.; disturbances) and axel-probe queries the engine for its state OnChange event fires when the state is recalculated

Enumerations

- enum RemoteMode {
  RemoteMode.Disconnected, RemoteMode.Jumbo_Scan, RemoteMode.Jumbo_Repeat, RemoteMode.Simple_Scan, RemoteMode.Simple_Repeat, RemoteMode.Ready_To_Remote }

  The current mode (group command) of incomming shots

6.1.1 Enumeration Type Documentation

6.1.1.1 RemoteMode

`enum Axel_probe.RemoteMode [strong]`

The current mode (group command) of incomming shots

Enumerator

Disconnected	
Jumbo_Scan	
Jumbo_Repeat	
Simple_Scan	
Simple_Repeat	
Ready_To_Remote	

Definition at line 22 of file ProbeEngine.cs.

6.2 Axel_probe::Properties Namespace Reference

Classes

- class Resources

  A strongly-typed resource class, for looking up localized strings, etc.
- class Settings

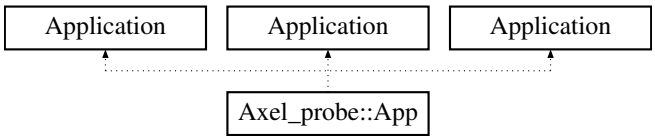
Chapter 7

Class Documentation

7.1 Axel_probe::App Class Reference

Interaction logic for App.xaml

Inheritance diagram for Axel_probe::App:



7.1.1 Detailed Description

Interaction logic for App.xaml

App

Definition at line 14 of file App.xaml.cs.

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

- App.xaml.cs
- App.g.cs
- App.g.i.cs

7.2 Axel_probe.Kalman Class Reference

Under development

7.2.1 Detailed Description

Under development

Definition at line 19 of file Kalman.cs.

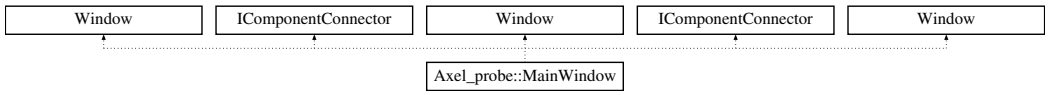
The documentation for this class was generated from the following file:

- Kalman.cs

7.3 Axel_probe::MainWindow Class Reference

Axel-probe is replacement simulator for MotMaster2 generating signal according to a settings corresponding to some experimental conditions

Inheritance diagram for Axel_probe::MainWindow:



7.3.1 Detailed Description

Axel-probe is replacement simulator for MotMaster2 generating signal according to a settings corresponding to some experimental conditions

MainWindow

Interaction logic for MainWindow.xaml

Definition at line 32 of file MainWindow.xaml.cs.

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

- MainWindow.xaml.cs
- MainWindow.g.cs
- MainWindow.g.i.cs

7.4 Axel_probe.PID Class Reference

Under development

7.4.1 Detailed Description

Under development

Definition at line 12 of file Kalman.cs.

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

- Kalman.cs

7.5 Axel_probe.ProbeEngine Class Reference

An calculating engine for generating a fringes patterns according simulated experimental conditions the engine runs by its timer and behaives similar to Axel-tilt its state evolve according to the set parameters (accel.; disturbances) and axel-probe queries the engine for its state OnChange event fires when the state is recalculated

Public Member Functions

- void SaveParams ()
  Save Params in json file (params.cfg)
- void LoadParams ()
  Load Params from json file (params.cfg)
- ProbeEngine ()
  Class constructor
- void Start (int dur=500)
  Start simulation
- void Stop ()
  Stop simulation
- void pauseSingle ()
  While in pause mode, shoot a single shot by calling this
- delegate void ChangeHandler (Point newAccel)
- delegate void LogHandler (string txt, Color? clr=null)
- void LogEvent (string txt, Color? clr=null)
- Point Gauss ()
  New acceleration point with optional noise by X and Y
- double acceleration (double pos)
  Generates specific pattern with set params
- double breathing ()
  Simulates signal loosing contrast (various reassons) factor to multiply the fringes with
- Point fringesPoint (double scanPhase, double curAccel)
  Individual for each point from the fringe
- double fringes (double curAccel)
  (re)Generates the fringes series
- int getIfringe (double x)
  Get index from phase in fringes series
- bool SingleShot (double A, int toAxis, ref MMexec mme)
  Generates the simulated to photo diode signal to be send to Axel-hub
- void CancelScan ()
- void DoScan (MMscan mms)
  Generates scan series of shots by group MMexec envelope
- void CancelRepeat ()
- void SimpleRepeat (int cycles, string groupID)
  Generate simple (initialted from here) repeat series of shots

Public Attributes

- Dictionary< string, double > bpps

  behaviour patterns parameters pattern => 0 -> constant; 1 -> trapeze; 2 -> sine ampl [mg] - accel. amplitude factor [mg/rad] step [%] percent from the pattern period time gap
- Dictionary< string, object > dps

  disturbances - fringe noise and Y breathing simulating experimental conditions Xnoise - [mg] noise in accel. (fringe shifting) Ynoise - [%] Y noise in fringe signal XnoiseIO - (bool) Xnoise switch YnoiseIO - (bool) Ynoise switch Brth↩ Pattern - (int) Breathing pattern BrthApmpl - Breathing amplitude BrthPeriod - Breathing period BrthIO - (bool) Breathing switch
- Boolean LockParams = false
- RemoteMessaging remote = null
- ChartCollection< Point > srsFringes = null
- ListBox lboxNB = null
- ChartCollection< double > srsSignalN = null
- ChartCollection< double > srsSignalB = null
- RemoteMode remoteMode = RemoteMode.Disconnected
- Stopwatch stopWatch
- double contrPhase = -11
- int b4ConstrID = 0

Protected Member Functions

- void ChangeEvent (Point newAccel)

Properties

- int axis `[get, set]`

  Axis mode -1 - old style; 0 - X; 1 - Y; 2 - X/Y
- double period `[get]`

  Time period of generated
- Boolean Enabled `[get, set]`

  One way to start/stop a simulation
- Boolean Pause `[get, set]`

  Temporary pause the simulation

Events

- ChangeHandler OnChange
- LogHandler OnLog

7.5.1 Detailed Description

An calculating engine for generating a fringes patterns according simulated experimental conditions the engine runs by its timer and behaives similar to Axel-tilt its state evolve according to the set parameters (accel.; disturbances) and axel-probe queries the engine for its state OnChange event fires when the state is recalculated

Definition at line 39 of file ProbeEngine.cs.

7.5.2 Constructor & Destructor Documentation

7.5.2.1 ProbeEngine()

```
Axel_probe.ProbeEngine.ProbeEngine ( )
```

Class constructor

Definition at line 146 of file ProbeEngine.cs.

7.5.3 Member Function Documentation

7.5.3.1 acceleration()

```
double Axel_probe.ProbeEngine.acceleration (
            double pos )
```

Generates specific pattern with set params

Parameters

pos	pos in current pattern [sec]

Returns

Definition at line 265 of file ProbeEngine.cs.

7.5.3.2 breathing()

```
double Axel_probe.ProbeEngine.breathing ( )
```

Simulates signal loosing contrast (various reassons) factor to multiply the fringes with

Returns

Definition at line 303 of file ProbeEngine.cs.

7.5.3.3 CancelRepeat()

```
void Axel_probe.ProbeEngine.CancelRepeat ( )
```

Definition at line 501 of file ProbeEngine.cs.

7.5.3.4 CancelScan()

```
void Axel_probe.ProbeEngine.CancelScan ( )
```

Definition at line 463 of file ProbeEngine.cs.

7.5.3.5 ChangeEvent()

```
void Axel_probe.ProbeEngine.ChangeEvent (
            Point newAccel )  [protected]
```

Definition at line 214 of file ProbeEngine.cs.

7.5.3.6 ChangeHandler()

```
delegate void Axel_probe.ProbeEngine.ChangeHandler (
            Point newAccel )
```

7.5.3.7 DoScan()

```
void Axel_probe.ProbeEngine.DoScan (
            MMscan mms )
```

Generates scan series of shots by group MMexec envelope

Parameters

mms	group MMexec

Definition at line 468 of file ProbeEngine.cs.

7.5.3.8 fringes()

```
double Axel_probe.ProbeEngine.fringes (
            double curAccel )
```

(re)Generates the fringes series

Parameters

curAccel	curAccel [mg] for illustration

Returns

accel with noise

Definition at line 331 of file ProbeEngine.cs.

7.5.3.9 fringesPoint()

```
Point Axel_probe.ProbeEngine.fringesPoint (
            double scanPhase,
            double curAccel )
```

Individual for each point from the fringe

Parameters

scanPhase	[rad]
curAccel	[mg]

Returns

result point

Definition at line 319 of file ProbeEngine.cs.

7.5.3.10 Gauss()

```
Point Axel_probe.ProbeEngine.Gauss ( )
```

New acceleration point with optional noise by X and Y

Returns

Definition at line 234 of file ProbeEngine.cs.

7.5.3.11 getIfringe()

```
int Axel_probe.ProbeEngine.getIfringe (
            double x )
```

Get index from phase in fringes series

Parameters

x	[rad]

Returns

Definition at line 351 of file ProbeEngine.cs.

7.5.3.12 LoadParams()

```
void Axel_probe.ProbeEngine.LoadParams ( )
```

Load Params from json file (params.cfg)

Definition at line 95 of file ProbeEngine.cs.

7.5.3.13 LogEvent()

```
void Axel_probe.ProbeEngine.LogEvent (
            string txt,
            Color?  clr = null )
```

Definition at line 224 of file ProbeEngine.cs.

7.5.3.14 LogHandler()

```
delegate void Axel_probe.ProbeEngine.LogHandler (
            string txt,
            Color?  clr = null )
```

7.5.3.15 pauseSingle()

```
void Axel_probe.ProbeEngine.pauseSingle ( )
```

While in pause mode, shoot a single shot by calling this

Definition at line 204 of file ProbeEngine.cs.

7.5.3.16 SaveParams()

```
void Axel_probe.ProbeEngine.SaveParams ( )
```

Save Params in json file (params.cfg)

Definition at line 83 of file ProbeEngine.cs.

7.5.3.17 SimpleRepeat()

```
void Axel_probe.ProbeEngine.SimpleRepeat (
            int cycles,
            string groupID )
```

Generate simple (initialted from here) repeat series of shots

Parameters

cycles	Number of shots (-1 if continious)
groupID	

Definition at line 507 of file ProbeEngine.cs.

7.5.3.18 SingleShot()

```
bool Axel_probe.ProbeEngine.SingleShot (
            double A,
            int toAxis,
            ref MMexec mme )
```

Generates the simulated to photo diode signal to be send to Axel-hub

Example base data + some noise N2 Ntot B2 Btot Bg N2 Ntot B2 Btot Bg 1 5 1 3 0 3 5 1 3 0 NB2 NBtot NB2 NBtot 0 2 2 2
A A
1 -1

Parameters

curAccel	

Returns

Definition at line 384 of file ProbeEngine.cs.

7.5.3.19 Start()

```
void Axel_probe.ProbeEngine.Start (
            int dur = 500 )
```

Start simulation

Parameters

dur	

Definition at line 167 of file ProbeEngine.cs.

7.5.3.20 Stop()

```
void Axel_probe.ProbeEngine.Stop ( )
```

Stop simulation

Definition at line 178 of file ProbeEngine.cs.

7.5.4 Member Data Documentation

7.5.4.1 b4ConstrID

```
int Axel_probe.ProbeEngine.b4ConstrID = 0
```

Definition at line 369 of file ProbeEngine.cs.

7.5.4.2 bpps

```
Dictionary<string, double> Axel_probe.ProbeEngine.bpps
```

behaviour patterns parameters pattern => 0 -> constant; 1 -> trapeze; 2 -> sine ampl [mg] - accel. amplitude factor [mg/rad] step [%] percent from the pattern period time gap

Definition at line 49 of file ProbeEngine.cs.

7.5.4.3 contrPhase

```
double Axel_probe.ProbeEngine.contrPhase = -11
```

Definition at line 369 of file ProbeEngine.cs.

7.5.4.4 dps

```
Dictionary<string, object> Axel_probe.ProbeEngine.dps
```

disturbances - fringe noise and Y breathing simulating experimental conditions Xnoise - [mg] noise in accel. (fringe shifting) Ynoise - [%] Y noise in fringe signal XnoiseIO - (bool) Xnoise switch YnoiseIO - (bool) Ynoise switch BrthPattern - (int) Breathing pattern BrthApmpl - Breathing amplitude BrthPeriod - Breathing period BrthIO - (bool) Breathing switch

Definition at line 62 of file ProbeEngine.cs.

7.5.4.5 lboxNB

```
ListBox Axel_probe.ProbeEngine.lboxNB = null
```

Definition at line 106 of file ProbeEngine.cs.

7.5.4.6 LockParams

```
Boolean Axel_probe.ProbeEngine.LockParams = false
```

Definition at line 70 of file ProbeEngine.cs.

7.5.4.7 remote

```
RemoteMessaging Axel_probe.ProbeEngine.remote = null
```

Definition at line 103 of file ProbeEngine.cs.

7.5.4.8 remoteMode

```
RemoteMode Axel_probe.ProbeEngine.remoteMode = RemoteMode.Disconnected
```

Definition at line 112 of file ProbeEngine.cs.

7.5.4.9 srsFringes

```
ChartCollection<Point> Axel_probe.ProbeEngine.srsFringes = null
```

Definition at line 105 of file ProbeEngine.cs.

7.5.4.10 srsSignalB

```
ChartCollection<double> Axel_probe.ProbeEngine.srsSignalB = null
```

Definition at line 108 of file ProbeEngine.cs.

7.5.4.11 srsSignalN

```
ChartCollection<double> Axel_probe.ProbeEngine.srsSignalN = null
```

Definition at line 107 of file ProbeEngine.cs.

7.5.4.12 stopWatch

```
Stopwatch Axel_probe.ProbeEngine.stopWatch
```

Definition at line 142 of file ProbeEngine.cs.

7.5.5 Property Documentation

7.5.5.1 axis

```
int Axel_probe.ProbeEngine.axis  [get], [set]
```

Axis mode -1 - old style; 0 - X; 1 - Y; 2 - X/Y

Definition at line 68 of file ProbeEngine.cs.

7.5.5.2 Enabled

```
Boolean Axel_probe.ProbeEngine.Enabled  [get], [set]
```

One way to start/stop a simulation

Definition at line 127 of file ProbeEngine.cs.

7.5.5.3 Pause

```
Boolean Axel_probe.ProbeEngine.Pause  [get], [set]
```

Temporary pause the simulation

Definition at line 190 of file ProbeEngine.cs.

7.5.5.4 period

```
double Axel_probe.ProbeEngine.period  [get]
```

Time period of generated

Definition at line 118 of file ProbeEngine.cs.

7.5.6 Event Documentation

7.5.6.1 OnChange

```
ChangeHandler Axel_probe.ProbeEngine.OnChange
```

Definition at line 212 of file ProbeEngine.cs.

7.5.6.2 OnLog

```
LogHandler Axel_probe.ProbeEngine.OnLog
```

Definition at line 223 of file ProbeEngine.cs.

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

- ProbeEngine.cs

7.6 Axel_probe::Properties::Resources Class Reference

A strongly-typed resource class, for looking up localized strings, etc.

7.6.1 Detailed Description

A strongly-typed resource class, for looking up localized strings, etc.

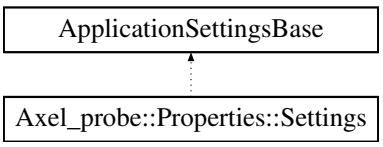
Definition at line 25 of file Resources.Designer.cs.

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

- Resources.Designer.cs

7.7 Axel_probe::Properties::Settings Class Reference

Inheritance diagram for Axel_probe::Properties::Settings:



7.7.1 Detailed Description

Definition at line 17 of file Settings.Designer.cs.

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

- Settings.Designer.cs

Chapter 8

File Documentation

8.1 App.g.cs File Reference

Classes

- class Axel_probe::App

  Interaction logic for App.xaml

Namespaces

- Axel_probe

8.2 App.g.i.cs File Reference

Classes

- class Axel_probe::App

  Interaction logic for App.xaml

Namespaces

- Axel_probe

8.3 App.xaml.cs File Reference

Classes

- class Axel_probe::App

  Interaction logic for App.xaml

Namespaces

- Axel_probe

8.4 AssemblyInfo.cs File Reference

8.5 Axel-probe_Content.g.i.cs File Reference

8.6 Kalman.cs File Reference

Classes

- class Axel_probe.PID

  Under development
- class Axel_probe.Kalman

  Under development

Namespaces

- Axel_probe

8.7 MainWindow.g.cs File Reference

Classes

- class Axel_probe::MainWindow

  Axel-probe is replacement simulator for MotMaster2 generating signal according to a settings corresponding to some experimental conditions

Namespaces

- Axel_probe

8.8 MainWindow.g.i.cs File Reference

Classes

- class Axel_probe::MainWindow

  Axel-probe is replacement simulator for MotMaster2 generating signal according to a settings corresponding to some experimental conditions

Namespaces

- Axel_probe

8.9 MainWindow.xaml.cs File Reference

Classes

- class Axel_probe::MainWindow

  Axel-probe is replacement simulator for MotMaster2 generating signal according to a settings corresponding to some experimental conditions

Namespaces

- Axel_probe

8.10 ProbeEngine.cs File Reference

Classes

- class Axel_probe.ProbeEngine

  An calculating engine for generating a fringes patterns according simulated experimental conditions the engine runs by its timer and behaives similar to Axel-tilt its state evolve according to the set parameters (accel.; disturbances) and axel-probe queries the engine for its state OnChange event fires when the state is recalculated

Namespaces

- Axel_probe

Enumerations

- enum Axel_probe.RemoteMode {
  Axel_probe.RemoteMode.Disconnected, Axel_probe.RemoteMode.Jumbo_Scan, Axel_probe.RemoteMode.Jumbo_Repeat,
  Axel_probe.RemoteMode.Simple_Scan,
  Axel_probe.RemoteMode.Simple_Repeat, Axel_probe.RemoteMode.Ready_To_Remote }

  The current mode (group command) of incomming shots

8.11 README.md File Reference

8.12 Resources.Designer.cs File Reference

Classes

- class Axel_probe::Properties::Resources

  A strongly-typed resource class, for looking up localized strings, etc.

Namespaces

- Axel_probe
- Axel_probe::Properties

8.13 Settings.Designer.cs File Reference

Classes

- class Axel_probe::Properties::Settings

Namespaces

- Axel_probe
- Axel_probe::Properties

Index