bwv_camera

1.00

Generated by Doxygen 1.8.13

Contents

1 Class Index **1**

1.1 Class List 1

2 File Index **3**

2.1 File List 3

3 Class Documentation **5**

3.1 Config Struct Reference 5

3.1.1 Detailed Description 5

3.1.2 Member Data Documentation 5

3.1.2.1 curvature 6

3.1.2.2 illuminationPower 6

3.1.2.3 maxRange 6

3.1.2.4 minRange 6

3.1.2.5 tlaser 6

3.1.2.6 workingMode 6

3.2 ContextStruct Struct Reference 7

3.2.1 Detailed Description 7

3.2.2 Member Data Documentation 7

3.2.2.1 apiMaster 8

3.2.2.2 apiSlave 8

3.2.2.3 config 8

3.2.2.4 devMaster 8

3.2.2.5 devSlave 8

3.3 GatingWrapper Class Reference 8
3.3.1 Detailed Description 9
3.3.2 Constructor & Destructor Documentation 9
3.3.2.1 GatingWrapper() 9
3.3.2.2 ∼GatingWrapper() 9
3.3.3 Member Function Documentation 10
3.3.3.1 GetIllumMode() 10
3.3.3.2 GetIllumPower() 10
3.3.3.3 GetIllumTlaser() 10
3.3.3.4 GetRangeEnd() 11
3.3.3.5 GetRangeStart() 11
3.3.3.6 GetRoadCurvature() 11
3.4 Main Class Reference 11
3.4.1 Detailed Description 12
3.4.2 Constructor & Destructor Documentation 12
3.4.2.1 Main() 12
3.4.2.2 ∼Main() 12
3.4.3 Member Function Documentation 13
3.4.3.1 GetGatingMaster() 13
3.4.3.2 GetGatingSlave() 13
3.4.3.3 StartCamera() 13
3.5 VideoWrapper Class Reference 13
3.5.1 Detailed Description 14
3.5.2 Constructor & Destructor Documentation 14
3.5.2.1 VideoWrapper() 14
3.5.2.2 ∼VideoWrapper() 14

4 File Documentation **15**

4.1 /home/user/bw_ws/src/bwv_camera/include/bwv_camera/bwv_types.h File Reference 15

4.1.1 Macro Definition Documentation 16

4.1.1.1 FRAME_HEIGHT 16

4.1.1.2 FRAME_WIDTH 16

4.1.2 Typedef Documentation 17

4.1.2.1 config_t 17

4.1.2.2 context_t 17

4.1.3 Enumeration Type Documentation 17

4.1.3.1 masterOrSlave 17

4.1.3.2 MetadataOffsetFirstRow 17

4.1.3.3 MetadataOffsetSecondRow 18

4.1.4 Variable Documentation 18

4.1.4.1 g_metadataTimer 18

4.2 /home/user/bw_ws/src/bwv_camera/include/bwv_camera/GatingWrapper.h File Reference 18

4.3 /home/user/bw_ws/src/bwv_camera/include/bwv_camera/Main.h File Reference 19

4.4 /home/user/bw_ws/src/bwv_camera/include/bwv_camera/utils.h File Reference 20

4.4.1 Function Documentation 21

4.4.1.1 CreateThread() 21

4.4.1.2 Sleep() 21

4.5 /home/user/bw_ws/src/bwv_camera/include/bwv_camera/VideoWrapper.h File Reference 21

4.6 /home/user/bw_ws/src/bwv_camera/src/bwv_camera_node.cpp File Reference 22

4.6.1 Function Documentation 23

4.6.1.1 main() 23

4.6.1.2 SigIntHandler() 23

4.7 /home/user/bw_ws/src/bwv_camera/src/GatingWrapper.cpp File Reference 23

4.8 /home/user/bw_ws/src/bwv_camera/src/Main.cpp File Reference 24

4.8.1 Function Documentation 25

4.8.1.1 g_latency() 25

4.8.2 Variable Documentation 25

4.8.2.1 g_latencyIdx . . . 25
4.8.2.2 g_latencySum . . . 25
4.8.2.3 g_metadataTimer . . . 25
4.8.2.4 g_nextFlag . . . 25
4.9 /home/user/bw_ws/src/bwv_camera/src/utils.cpp File Reference . . . 26
4.9.1 Typedef Documentation . . . 26
4.9.1.1 thread_func_t . . . 26
4.9.2 Function Documentation . . . 26
4.9.2.1 CreateThread() . . . 26
4.9.2.2 Sleep() . . . 27
4.10 /home/user/bw_ws/src/bwv_camera/src/VideoWrapper.cpp File Reference . . . 27
4.11 /home/user/bw_ws/src/bwv_camera_sub/src/bwv_camera_sub.cpp File Reference . . . 27
4.11.1 Function Documentation . . . 28
4.11.1.1 get_timestamp() . . . 28
4.11.1.2 ImageCallbackMaster() . . . 28
4.11.1.3 ImageCallbackSlave() . . . 28
4.11.1.4 main() . . . 28

Index **29**

Chapter 1

Class Index

1.1 Class List

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

Config
Configuration class 5
ContextStruct
Context struct 7
GatingWrapper
Ros wrapper for the camera gating operation 8
Main
This is main class 11
VideoWrapper
Wrapper to the video operations 13

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

/home/user/bw_ws/src/bwv_camera/include/bwv_camera/ **bwv_types.h** 15
/home/user/bw_ws/src/bwv_camera/include/bwv_camera/ **GatingWrapper.h** 18
/home/user/bw_ws/src/bwv_camera/include/bwv_camera/ **Main.h** 19
/home/user/bw_ws/src/bwv_camera/include/bwv_camera/ **utils.h** 20
/home/user/bw_ws/src/bwv_camera/include/bwv_camera/ **VideoWrapper.h** 21
/home/user/bw_ws/src/bwv_camera/src/ **bwv_camera_node.cpp** 22
/home/user/bw_ws/src/bwv_camera/src/ **GatingWrapper.cpp** 23
/home/user/bw_ws/src/bwv_camera/src/ **Main.cpp** 24
/home/user/bw_ws/src/bwv_camera/src/ **utils.cpp** 26
/home/user/bw_ws/src/bwv_camera/src/ **VideoWrapper.cpp** 27
/home/user/bw_ws/src/bwv_camera_sub/src/ **bwv_camera_sub.cpp** 27

Chapter 3

Class Documentation

3.1 Config Struct Reference

Configuration class.

`#include <bwv_types.h>`

Public Attributes

- unsigned short **minRange**

  minRange - minimum visible range in meters, available ranges 12 - 350[m]
- unsigned short **maxRange**

  maxRange - maximum visible range in meters, available ranges 12 - 350[m]
- unsigned char **workingMode**

  workingMode - availabe working modes -
  0 LRN - default mode (Left, Right, None illumination)
  1 BNN (Both, None, None illumination)
  2 Full - ACS
- unsigned char **illuminationPower**

  illuminationPower - power of illumination, range 0 - 130[%]
- unsigned char **tlaser**

  tlaser - laser time duration - range 12 - 70 which represents 0.12 - 0.70 [usec]
- unsigned char **curvature**

  curvature - not implemented

3.1.1 Detailed Description

Configuration class.

holds dynamic reconfigure parameters.

3.1.2 Member Data Documentation

3.1.2.1 curvature

```
unsigned char Config::curvature
```

curvature - not implemented

3.1.2.2 illuminationPower

```
unsigned char Config::illuminationPower
```

illuminationPower - power of illumination, range 0 - 130[%]

3.1.2.3 maxRange

```
unsigned short Config::maxRange
```

maxRange - maximum visible range in meters, available ranges 12 - 350[m]

3.1.2.4 minRange

```
unsigned short Config::minRange
```

minRange - minimum visible range in meters, available ranges 12 - 350[m]

3.1.2.5 tlaser

```
unsigned char Config::tlaser
```

tlaser - laser time duration - range 12 - 70 which represents 0.12 - 0.70 [usec]

3.1.2.6 workingMode

```
unsigned char Config::workingMode
```

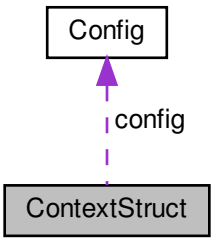
workingMode - availabe working modes -
0 LRN - default mode (Left, Right, None illumination)
1 BNN (Both, None, None illumination)
2 Full - ACS

3.2 ContextStruct Struct Reference

context struct.

`#include <bwv_types.h>`

Collaboration diagram for ContextStruct:



Public Attributes

- void * **apiMaster**
  reference to the master API instance
- void * **apiSlave**
  reference to the slave API instance
- device_t **devMaster**
  connection master device
- device_t **devSlave**
  connection slave device
- **config_t config**
  local configuration parameters

3.2.1 Detailed Description

context struct.

describes the related members available to other classes

3.2.2 Member Data Documentation

3.2.2.1 apiMaster

```
void* ContextStruct::apiMaster
```

reference to the master API instance

3.2.2.2 apiSlave

```
void* ContextStruct::apiSlave
```

reference to the slave API instance

3.2.2.3 config

```
config_t ContextStruct::config
```

local configuration parameters

See also

Config (p. 5)

3.2.2.4 devMaster

```
device_t ContextStruct::devMaster
```

connection master device

3.2.2.5 devSlave

```
device_t ContextStruct::devSlave
```

connection slave device

3.3 GatingWrapper Class Reference

a ros wrapper for the camera gating operation

```
#include <GatingWrapper.h>
```

Public Member Functions

- **GatingWrapper** (**context_t** ∗context, const ros::NodeHandle &nh, int isSlave)

  GatingWrapper *(*p. *8) ctor.*
- ∼**GatingWrapper** ()

  GatingWrapper *(*p. *8) dtor.*
- unsigned char **GetIllumMode** ()
- unsigned char **GetIllumPower** ()
- unsigned char **GetIllumTlaser** ()
- unsigned short **GetRangeStart** ()
- unsigned short **GetRangeEnd** ()
- unsigned char **GetRoadCurvature** ()

  @ returns the current road curvature - currently not implemented

3.3.1 Detailed Description

a ros wrapper for the camera gating operation

3.3.2 Constructor & Destructor Documentation

3.3.2.1 GatingWrapper()

```
GatingWrapper::GatingWrapper (
              context_t * context,
             const ros::NodeHandle & nh,
             int isSlave )  [explicit]
```

GatingWrapper (p. 8) ctor.

Parameters

context	- pointer to the global related content
nodeHandle	- node handle instance

See also

ContextStruct (p. 7) ros::NodeHandle

3.3.2.2 ∼GatingWrapper()

```
GatingWrapper::~GatingWrapper ( )
```

GatingWrapper (p. 8) dtor.

3.3.3 Member Function Documentation

3.3.3.1 GetIllumMode()

```
unsigned char GatingWrapper::GetIllumMode ( )  [inline]
```

Returns

the current illumination mode

See also

Config (p. 5)

3.3.3.2 GetIllumPower()

```
unsigned char GatingWrapper::GetIllumPower ( )  [inline]
```

Returns

the current illumination power -
represented in percents

See also

Config (p. 5)

3.3.3.3 GetIllumTlaser()

```
unsigned char GatingWrapper::GetIllumTlaser ( )  [inline]
```

Returns

the current laser time duration -]n represented in micro seconds $*$ 100

See also

Config (p. 5)

3.3.3.4 GetRangeEnd()

```
unsigned short GatingWrapper::GetRangeEnd ( )  [inline]
```

Returns

the current visible end range -
represented in meters

See also

Config (p. 5)

3.3.3.5 GetRangeStart()

```
unsigned short GatingWrapper::GetRangeStart ( )  [inline]
```

Returns

the current visible start range -
represented in meters

See also

Config (p. 5)

3.3.3.6 GetRoadCurvature()

```
unsigned char GatingWrapper::GetRoadCurvature ( )  [inline]
```

@ returns the current road curvature -
currently not implemented

See also

Config (p. 5)

3.4 Main Class Reference

this is main class.

```
#include <Main.h>
```

Public Member Functions

- **Main** (**context_t** ∗context, const ros::NodeHandle &nodeHandle)
  Main *(p. 11) ctor.*
- ∼**Main** ()
  dtor
- int **StartCamera** ()
  start the camera operation and flow.
- **GatingWrapper** ∗ **GetGatingMaster** ()
- **GatingWrapper** ∗ **GetGatingSlave** ()
  same as GetGatingMaster for slave camera

3.4.1 Detailed Description

this is main class.

controls the driver's operation flow.

3.4.2 Constructor & Destructor Documentation

3.4.2.1 Main()

```
Main::Main (
              context_t * context,
             const ros::NodeHandle & nodeHandle )  [explicit]
```

Main (p. 11) ctor.

Parameters

context	- pointer to the global related content
nodeHandle	- node handle instance

See also

ContextStruct (p. 7) ros::NodeHandle

3.4.2.2 ∼Main()

```
Main::∼Main ( )
```

dtor

3.4.3 Member Function Documentation

3.4.3.1 GetGatingMaster()

```
GatingWrapper* Main::GetGatingMaster ( )  [inline]
```

Returns

the instance of master's camera gating wrapper

See also

GatingWrapper (p. 8)

3.4.3.2 GetGatingSlave()

```
GatingWrapper* Main::GetGatingSlave ( )  [inline]
```

same as GetGatingMaster for slave camera

See also

GatingWrapper (p. 8)

3.4.3.3 StartCamera()

```
int Main::StartCamera ( )
```

start the camera operation and flow.

Returns

0 for success
-1 for allocation fail
-2 for connection fail

3.5 VideoWrapper Class Reference

a wrapper to the video operations

```
#include <VideoWrapper.h>
```

Public Member Functions

- **VideoWrapper** (**context_t** ∗context, const ros::NodeHandle &nh, int **masterOrSlave**)
  VideoWrapper *(p. 13) ctor.*
- ∼**VideoWrapper** ()
  VideoWrapper *(p. 13) dtor.*

3.5.1 Detailed Description

a wrapper to the video operations

responsible for getting video frames from a connected camera master or slave,
and convert it to a standard sensor_msgs format.
possible printing of a relevant metadata if ROS DEBUG verbosity is enabled.

3.5.2 Constructor & Destructor Documentation

3.5.2.1 VideoWrapper()

```
VideoWrapper::VideoWrapper (
              context_t * context,
             const ros::NodeHandle & nh,
             int masterOrSlave )  [explicit]
```

VideoWrapper (p. 13) ctor.

Parameters

context	- pointer to the global related content
nodeHandle	- node handle instance

See also

ContextStruct (p. 7) ros::NodeHandle

3.5.2.2 ∼VideoWrapper()

```
VideoWrapper::∼VideoWrapper ( )
```

VideoWrapper (p. 13) dtor.

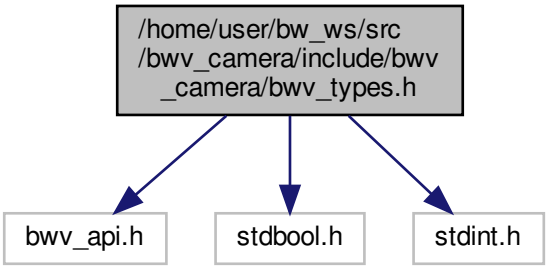
Chapter 4

File Documentation

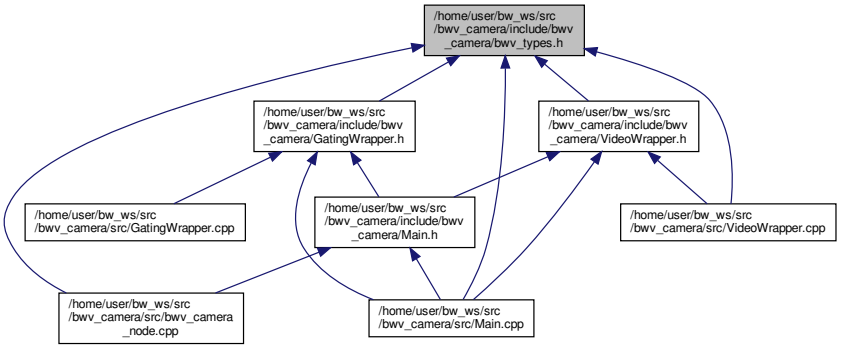
4.1 /home/user/bw_ws/src/bwv_camera/include/bwv_camera/bwv_types.h File Reference

```
#include "bwv_api.h"
```

Include dependency graph for bwv_types.h:



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



Classes

- struct **Config**
  Configuration class.
- struct **ContextStruct**
  context struct.

Macros

- #define **FRAME_WIDTH** 800
- #define **FRAME_HEIGHT** 482

Typedefs

- typedef struct **Config** **config_t**
  Configuration class.
- typedef struct **ContextStruct** **context_t**
  context struct.

Enumerations

- enum **masterOrSlave** { **MASTER**, **SLAVE** }
- enum **MetadataOffsetFirstRow** { **TIMESTAMP** = 8, **HW_FRAME_COUNTER** = 20 }
  row number 0 offsets
- enum **MetadataOffsetSecondRow** {
  ARAVA_ID = 0, **MASTER_SLAVE** = 200, **X_START** = 201, **X_END** = 202,
  POWER = 203, **TLASER** = 204, **WORKING_MODE** = 205, **ILLUMINATOR** = 206,
  ROAD_CURVATURE = 207 }
  row number 1 offsets

Variables

- uint64_t **g_metadataTimer**

4.1.1 Macro Definition Documentation

4.1.1.1 FRAME_HEIGHT

```
#define FRAME_HEIGHT 482
```

4.1.1.2 FRAME_WIDTH

```
#define FRAME_WIDTH 800
```

4.1.2 Typedef Documentation

4.1.2.1 config_t

`typedef struct` **`Config config_t`**

Configuration class.

holds dynamic reconfigure parameters.

4.1.2.2 context_t

`typedef struct` **`ContextStruct context_t`**

context struct.

describes the related members available to other classes

4.1.3 Enumeration Type Documentation

4.1.3.1 masterOrSlave

`enum` **`masterOrSlave`**

Enumerator

MASTER	is master
SLAVE	is slave

4.1.3.2 MetadataOffsetFirstRow

`enum` **`MetadataOffsetFirstRow`**

row number 0 offsets

Enumerator

TIMESTAMP	8
HW_FRAME_COUNTER	20

4.1.3.3 MetadataOffsetSecondRow

`enum` **MetadataOffsetSecondRow**

row number 1 offsets

Enumerator

ARAVA_ID	0
MASTER_SLAVE	200
X_START	201
X_END	202
POWER	203
TLASER	204
WORKING_MODE	205
ILLUMINATOR	206
ROAD_CURVATURE	207

4.1.4 Variable Documentation

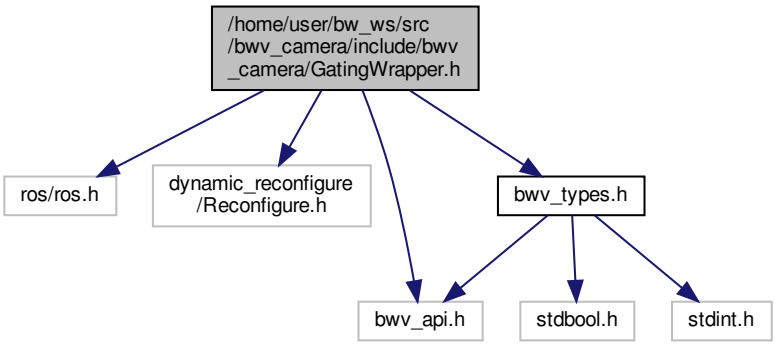
4.1.4.1 g_metadataTimer

```
uint64_t g_metadataTimer
```

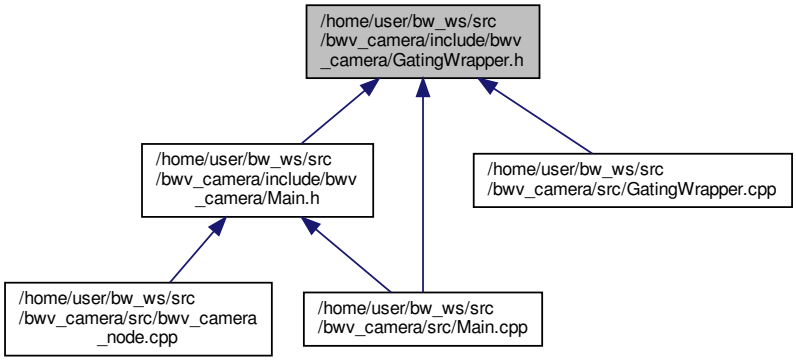
4.2 /home/user/bw_ws/src/bwv_camera/include/bwv_camera/GatingWrapper.h File Reference

```
#include <ros/ros.h>
#include <dynamic_reconfigure/Reconfigure.h>
#include "bwv_api.h"
#include "bwv_types.h"
```

Include dependency graph for GatingWrapper.h:



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



Classes

- class **GatingWrapper**

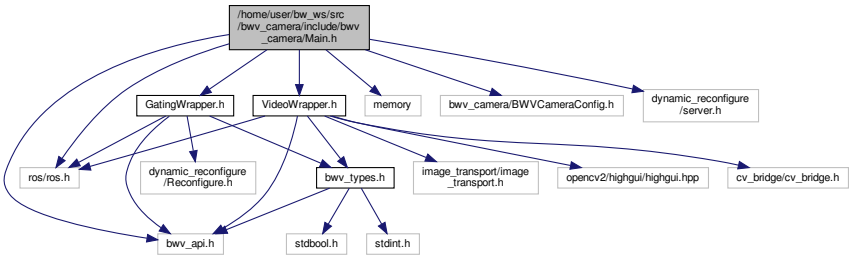
  a ros wrapper for the camera gating operation

4.3 /home/user/bw_ws/src/bwv_camera/include/bwv_camera/Main.h File Reference

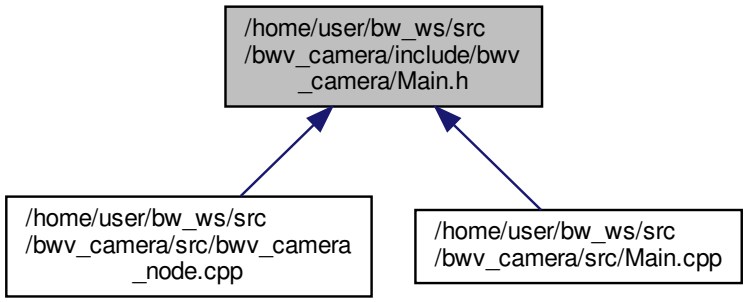
```
#include <ros/ros.h>
#include <memory>
#include "bwv_api.h"
#include "VideoWrapper.h"
#include "GatingWrapper.h"
#include <bwv_camera/BWVCameraConfig.h>
```

```
#include <dynamic_reconfigure/server.h>
```

Include dependency graph for Main.h:



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



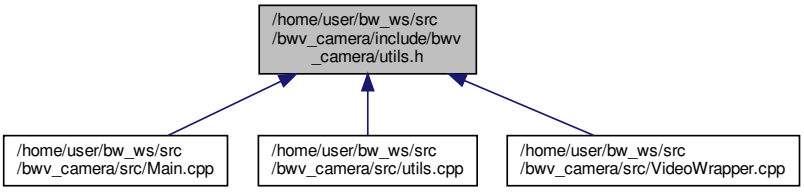
Classes

- class **Main**

  this is main class.

4.4 /home/user/bw_ws/src/bwv_camera/include/bwv_camera/utils.h File Reference

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



Functions

- void * **CreateThread** (void *function, void *args, int *result)

  thread creation wrapper
- void **Sleep** (int ms)

  sleep function wrapper

4.4.1 Function Documentation

4.4.1.1 CreateThread()

```
void* CreateThread (
            void * function,
            void * args,
            int * result )
```

thread creation wrapper

Parameters

funcion	a pointer to thread fuction with the prototype `void func(void* args)`
args	the thread pass argument
result	thread return value

4.4.1.2 Sleep()

```
void Sleep (
            int ms )
```

sleep function wrapper

Parameters

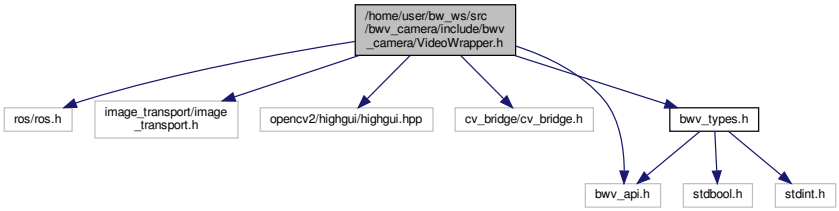
ms	sleep time in milliseconds

4.5 /home/user/bw_ws/src/bwv_camera/include/bwv_camera/VideoWrapper.h File Reference

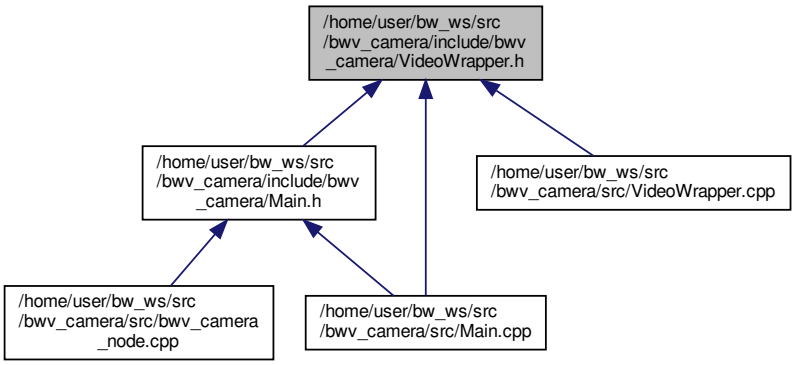
```
#include <ros/ros.h>
#include <image_transport/image_transport.h>
```

```
#include <opencv2/highgui/highgui.hpp>
#include <cv_bridge/cv_bridge.h>
#include "bwv_types.h"
#include "bwv_api.h"
```

Include dependency graph for VideoWrapper.h:



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



Classes

- class **VideoWrapper**

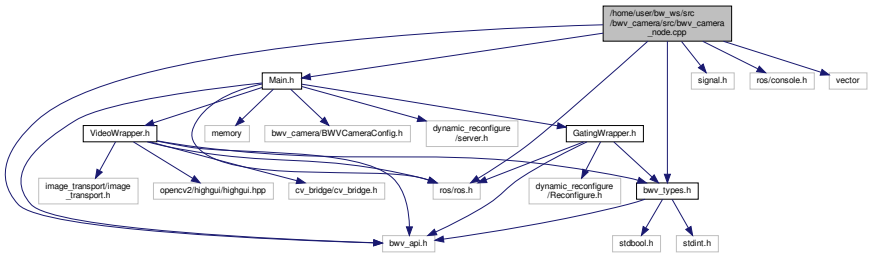
  a wrapper to the video operations

4.6 /home/user/bw_ws/src/bwv_camera/src/bwv_camera_node.cpp File Reference

```
#include <ros/ros.h>
#include <signal.h>
#include <ros/console.h>
#include <vector>
#include "bwv_api.h"
#include "bwv_types.h"
```

```
#include "Main.h"
```

Include dependency graph for bwv_camera_node.cpp:



Functions

- void **SigIntHandler** (int sig)
- int **main** (int argc, char **argv)

4.6.1 Function Documentation

4.6.1.1 main()

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

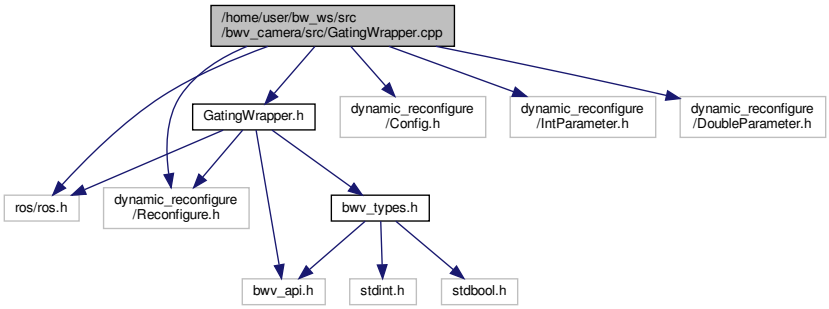
4.6.1.2 SigIntHandler()

```
void SigIntHandler (
            int sig )
```

4.7 /home/user/bw_ws/src/bwv_camera/src/GatingWrapper.cpp File Reference

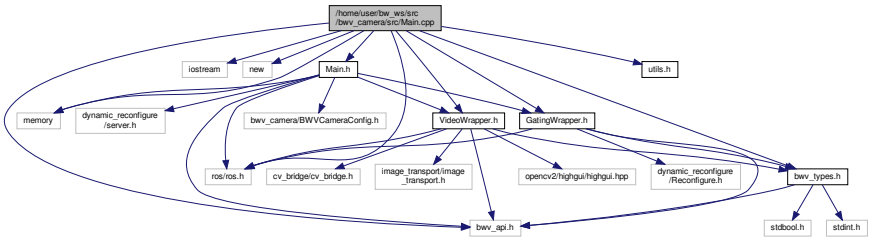
```
#include <ros/ros.h>
#include <dynamic_reconfigure/Reconfigure.h>
#include <dynamic_reconfigure/Config.h>
#include <dynamic_reconfigure/IntParameter.h>
#include <dynamic_reconfigure/DoubleParameter.h>
```

```
#include "GatingWrapper.h"
```
Include dependency graph for GatingWrapper.cpp:



4.8 /home/user/bw_ws/src/bwv_camera/src/Main.cpp File Reference

```
#include <ros/ros.h>
#include <iostream>
#include <new>
#include <memory>
#include "bwv_api.h"
#include "Main.h"
#include "VideoWrapper.h"
#include "GatingWrapper.h"
#include "utils.h"
#include "bwv_types.h"
```
Include dependency graph for Main.cpp:



Functions

- std::vector< uint64_t > **g_latency** (500, 0)

Variables

- uint64_t **g_metadataTimer**
- unsigned long **g_latencySum**
- unsigned int **g_latencyIdx**
- bool **g_nextFlag** = false

4.8.1 Function Documentation

4.8.1.1 g_latency()

```
std::vector<uint64_t> g_latency (
            500 ,
            0  )
```

4.8.2 Variable Documentation

4.8.2.1 g_latencyIdx

```
unsigned int g_latencyIdx
```

4.8.2.2 g_latencySum

```
unsigned long g_latencySum
```

4.8.2.3 g_metadataTimer

```
uint64_t g_metadataTimer
```

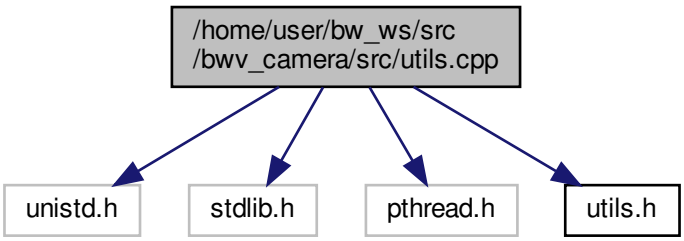
4.8.2.4 g_nextFlag

```
bool g_nextFlag = false
```

4.9 /home/user/bw_ws/src/bwv_camera/src/utils.cpp File Reference

```
#include <unistd.h>
#include <stdlib.h>
#include <pthread.h>
#include "utils.h"
```

Include dependency graph for utils.cpp:



Typedefs

- typedef void * **thread_func_t**(void *)

Functions

- void * **CreateThread** (void *function, void *args, int *result)

  thread creation wrapper
- void **Sleep** (int ms)

  sleep function wrapper

4.9.1 Typedef Documentation

4.9.1.1 thread_func_t

```
typedef void* thread_func_t(void *)
```

4.9.2 Function Documentation

4.9.2.1 CreateThread()

```
void* CreateThread (
            void * function,
            void * args,
            int * result )
```

thread creation wrapper

Parameters

funcion	a pointer to thread fuction with the prototype `void func(void* args)`
args	the thread pass argument
result	thread return value

4.9.2.2 Sleep()

```
void Sleep (
            int ms )
```

sleep function wrapper

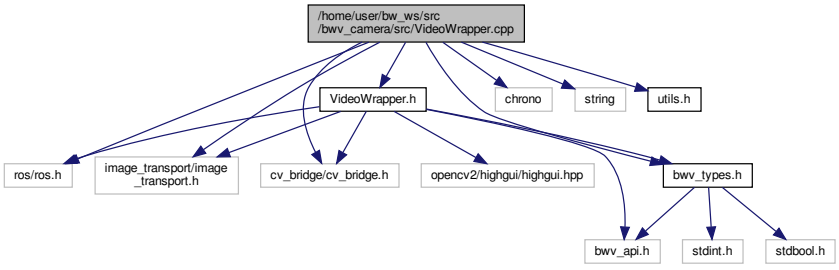
Parameters

ms	sleep time in milliseconds

4.10 /home/user/bw_ws/src/bwv_camera/src/VideoWrapper.cpp File Reference

```
#include <ros/ros.h>
#include <image_transport/image_transport.h>
#include <cv_bridge/cv_bridge.h>
#include <chrono>
#include <string>
#include "VideoWrapper.h"
#include "bwv_types.h"
#include "utils.h"
```

Include dependency graph for VideoWrapper.cpp:

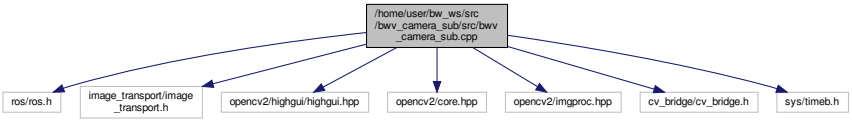


4.11 /home/user/bw_ws/src/bwv_camera_sub/src/bwv_camera_sub.cpp File Reference

```
#include <ros/ros.h>
#include <image_transport/image_transport.h>
```

```
#include <opencv2/highgui/highgui.hpp>
#include <opencv2/core.hpp>
#include <opencv2/imgproc.hpp>
#include <cv_bridge/cv_bridge.h>
#include <sys/timeb.h>
```

Include dependency graph for bwv_camera_sub.cpp:



Functions

- unsigned long long **get_timestamp** ()
- void **ImageCallbackMaster** (const sensor_msgs::ImageConstPtr &img)
- void **ImageCallbackSlave** (const sensor_msgs::ImageConstPtr &img)
- int **main** (int argc, char **argv)

4.11.1 Function Documentation

4.11.1.1 get_timestamp()

```
unsigned long long get_timestamp ( )
```

4.11.1.2 ImageCallbackMaster()

```
void ImageCallbackMaster (
            const sensor_msgs::ImageConstPtr & img )
```

4.11.1.3 ImageCallbackSlave()

```
void ImageCallbackSlave (
            const sensor_msgs::ImageConstPtr & img )
```

4.11.1.4 main()

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

Index

/home/user/bw_ws/src/bwv_camera/include/bwv_↩
camera/GatingWrapper.h, 18
/home/user/bw_ws/src/bwv_camera/include/bwv_↩
camera/Main.h, 19
/home/user/bw_ws/src/bwv_camera/include/bwv_↩
camera/VideoWrapper.h, 21
/home/user/bw_ws/src/bwv_camera/include/bwv_↩
camera/bwv_types.h, 15
/home/user/bw_ws/src/bwv_camera/include/bwv_↩
camera/utils.h, 20
/home/user/bw_ws/src/bwv_camera/src/Gating↩
Wrapper.cpp, 23
/home/user/bw_ws/src/bwv_camera/src/Main.cpp, 24
/home/user/bw_ws/src/bwv_camera/src/VideoWrapper.↩
cpp, 27
/home/user/bw_ws/src/bwv_camera/src/bwv_camera↩
_node.cpp, 22
/home/user/bw_ws/src/bwv_camera/src/utils.cpp, 26
/home/user/bw_ws/src/bwv_camera_sub/src/bwv_↩
camera_sub.cpp, 27
∼GatingWrapper
GatingWrapper, 9
∼Main
Main, 12
∼VideoWrapper
VideoWrapper, 14

apiMaster
ContextStruct, 7
apiSlave
ContextStruct, 8

bwv_camera_node.cpp
main, 23
SigIntHandler, 23
bwv_camera_sub.cpp
get_timestamp, 28
ImageCallbackMaster, 28
ImageCallbackSlave, 28
main, 28
bwv_types.h
config_t, 17
context_t, 17
FRAME_HEIGHT, 16
FRAME_WIDTH, 16
g_metadataTimer, 18
masterOrSlave, 17
MetadataOffsetFirstRow, 17
MetadataOffsetSecondRow, 18
Config, 5
curvature, 5
illuminationPower, 6
maxRange, 6
minRange, 6
tlaser, 6
workingMode, 6
config
ContextStruct, 8
config_t
bwv_types.h, 17
context_t
bwv_types.h, 17
ContextStruct, 7
apiMaster, 7
apiSlave, 8
config, 8
devMaster, 8
devSlave, 8
CreateThread
utils.cpp, 26
utils.h, 21
curvature
Config, 5

devMaster
ContextStruct, 8
devSlave
ContextStruct, 8

FRAME_HEIGHT
bwv_types.h, 16
FRAME_WIDTH
bwv_types.h, 16

g_latency
Main.cpp, 25
g_latencyIdx
Main.cpp, 25
g_latencySum
Main.cpp, 25
g_metadataTimer
bwv_types.h, 18
Main.cpp, 25
g_nextFlag
Main.cpp, 25
GatingWrapper, 8
∼GatingWrapper, 9
GatingWrapper, 9
GetIllumMode, 10

GetIllumPower, 10
GetIllumTlaser, 10
GetRangeEnd, 10
GetRangeStart, 11
GetRoadCurvature, 11
get_timestamp
bwv_camera_sub.cpp, 28
GetGatingMaster
Main, 13
GetGatingSlave
Main, 13
GetIllumMode
GatingWrapper, 10
GetIllumPower
GatingWrapper, 10
GetIllumTlaser
GatingWrapper, 10
GetRangeEnd
GatingWrapper, 10
GetRangeStart
GatingWrapper, 11
GetRoadCurvature
GatingWrapper, 11

illuminationPower
Config, 6
ImageCallbackMaster
bwv_camera_sub.cpp, 28
ImageCallbackSlave
bwv_camera_sub.cpp, 28

Main, 11
∼Main, 12
GetGatingMaster, 13
GetGatingSlave, 13
Main, 12
StartCamera, 13
main
bwv_camera_node.cpp, 23
bwv_camera_sub.cpp, 28
Main.cpp
g_latency, 25
g_latencyIdx, 25
g_latencySum, 25
g_metadataTimer, 25
g_nextFlag, 25
masterOrSlave
bwv_types.h, 17
maxRange
Config, 6
MetadataOffsetFirstRow
bwv_types.h, 17
MetadataOffsetSecondRow
bwv_types.h, 18
minRange
Config, 6

SigIntHandler
bwv_camera_node.cpp, 23
Sleep
utils.cpp, 27
utils.h, 21
StartCamera
Main, 13

thread_func_t
utils.cpp, 26
tlaser
Config, 6

utils.cpp
CreateThread, 26
Sleep, 27
thread_func_t, 26
utils.h
CreateThread, 21
Sleep, 21

VideoWrapper, 13
∼VideoWrapper, 14
VideoWrapper, 14

workingMode
Config, 6