sharedmemory - sender/empfaenger

3.0.0

Generated by Doxygen 1.6.1

Thu Jun 2 21:07:28 2016

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 data_collect Struct Reference

`#include <sharedmemory.h>`

Public Attributes

- char * segment
- int shmid
- int sem_r
- int sem_w
- int shm_size
- int use_mode

3.1.1 Detailed Description

Definition at line 51 of file sharedmemory.h.

3.1.2 Member Data Documentation

3.1.2.1 char* data_collect::segment

Definition at line 52 of file sharedmemory.h.

Referenced by closeSegment(), createSegment(), and main().

3.1.2.2 int data_collect::sem_r

Definition at line 54 of file sharedmemory.h.

Referenced by closeSegment(), createSegment(), and main().

3.1.2.3 int data_collect::sem_w

Definition at line 55 of file sharedmemory.h.

Referenced by closeSegment(), createSegment(), and main().

3.1.2.4 int data_collect::shm_size

Definition at line 56 of file sharedmemory.h.

Referenced by createSegment(), and main().

3.1.2.5 int data_collect::shmid

Definition at line 53 of file sharedmemory.h.

Referenced by closeSegment(), and createSegment().

3.1.2.6 int data_collect::use_mode

Definition at line 57 of file sharedmemory.h.

Referenced by createSegment().

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

- sharedmemory.h

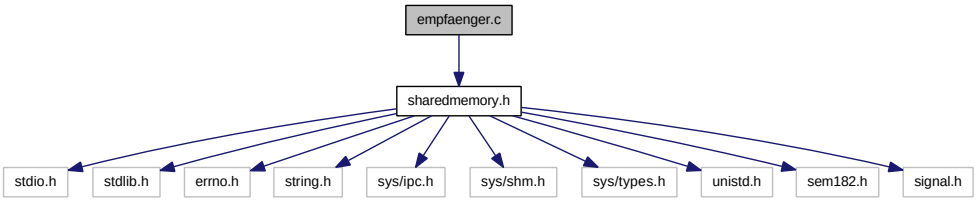
Chapter 4

File Documentation

4.1 empfaenger.c File Reference

```
#include "sharedmemory.h"
```

Include dependency graph for empfaenger.c:



Functions

- int main (int argc, char ∗argv[ ])
  empfaenger

4.1.1 Detailed Description

sharedmemory Beispiel 3

Author:

Karin Kalman <karin.kalman@technikum-wien.at>
Michael Mueller <michael.mueller@technikum-wien.at>
Gerhard Sabeditsch <gerhard.sabeditsch@technikum-wien.at>

Date:

2016/05/28

Version:

Revision 1

URL: $HeadURL$

Last Modified: Author Gerhard

Definition in file empfaenger.c.

4.1.2 Function Documentation

4.1.2.1 int main (int *argc*, char * *argv*[ ])

Empfaenger Programm

A empfaenger receives data from a shared memory

Parameters:

argc the number of arguments

argv the arguments itselves (including the program name in argv[0])

Returns:

0 or -1

Return values:

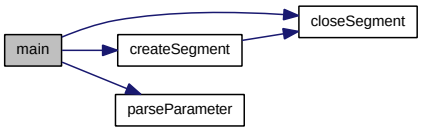
0 (Everything OK)

-1 (Something goes wrong)

Definition at line 47 of file empfaenger.c.

References closeSegment(), createSegment(), parseParameter(), READ_MODE, data_collect::segment, data_collect::sem_r, data_collect::sem_w, and data_collect::shm_size.

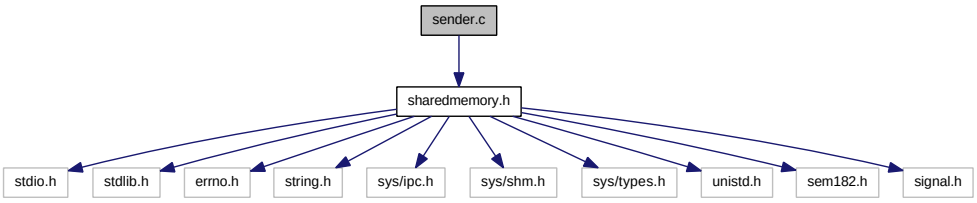
Here is the call graph for this function:



4.2 sender.c File Reference

```
#include "sharedmemory.h"
```

Include dependency graph for sender.c:



Functions

- int main (int argc, char ∗argv[ ])

  sender

4.2.1 Detailed Description

sharedmemory Beispiel 3

Author:

Karin Kalman <karin.kalman@technikum-wien.at>
Michael Mueller <michael.mueller@technikum-wien.at>
Gerhard Sabeditsch <gerhard.sabeditsch@technikum-wien.at>

Date:

2016/04/17

Version:

Revision 1

URL: $HeadURL$

Last Modified: Author Gerhard

Definition in file sender.c.

4.2.2 Function Documentation

4.2.2.1 int main (int *argc*, char ∗ *argv*[ ])

Sender Programm

A sender transmitts data to a shared memory

Parameters:

argc the number of arguments

argv the arguments itselves (including the program name in argv[0])

Returns:

0 or -1

Return values:

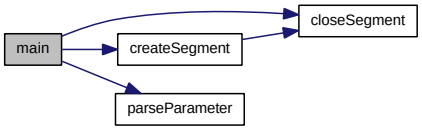
0 (Everything OK)

-1 (Something goes wrong)

Definition at line 47 of file sender.c.

References closeSegment(), createSegment(), parseParameter(), data_collect::segment, data_collect::sem_r, data_collect::sem_w, data_collect::shm_size, and WRITE_MODE.

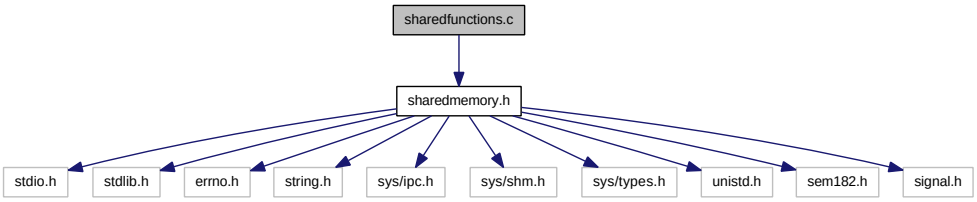
Here is the call graph for this function:



4.3 sharedfunctions.c File Reference

```
#include "sharedmemory.h"
```

Include dependency graph for sharedfunctions.c:



Functions

- data_collect createSegment (int shm_size, int shm_mode)

  createSegment

- int closeSegment (data_collect shm_sem)

  closeSegment

- int parseParameter (int argc, char ∗argv[ ])

  parseParameter

4.3.1 Detailed Description

sharedmemory Beispiel 3

Author:

Karin Kalman <karin.kalman@technikum-wien.at>
Michael Mueller <michael.mueller@technikum-wien.at>
Gerhard Sabeditsch <gerhard.sabeditsch@technikum-wien.at>

Date:

2016/05/28

Version:

Revision 1

URL: $HeadURL$

Last Modified: Author Gerhard

Definition in file sharedfunctions.c.

4.3.2 Function Documentation

4.3.2.1 int closeSegment (data_collect *shm_sem*)

closeSegment

cleans up semaphores, detatches shared memory segment and mark as removeable

Parameters:

shm_sem

Returns:

> -1 or -1

Return values:

> -1 ==> Everything OK
-1 ==> Somthing goes wrong

Definition at line 140 of file sharedfunctions.c.

References data_collect::segment, data_collect::sem_r, data_collect::sem_w, and data_collect::shmid.

Referenced by createSegment(), and main().

4.3.2.2 data_collect createSegment (int *shm_size*, int *shm_mode*)

createSegment

creates a a ring buffer and there semaphores

Parameters:

shm_size size of the ring buffer
shm_mode (read or write)

Returns:

NULL or ! NULL

Return values:

! NULL ==> Everything OK
NULL ==> Somthing goes wrong

Definition at line 50 of file sharedfunctions.c.

References closeSegment(), READ_MODE, data_collect::segment, data_collect::sem_r, SEM_R_KEY, data_collect::sem_w, SEM_W_KEY, SHM_KEY, data_collect::shm_size, data_collect::shmid, and data_collect::use_mode.

Referenced by main().

Here is the call graph for this function:

4.3.2.3 int parseParameter (int *argc*, char * *argv*[ ])

parseParameter

get the size of the ringbuffer out of the given parameters

Returns:

>0 or -1

Return values:

>0 ==> Everything OK

-1 ==> Somthing goes wrong

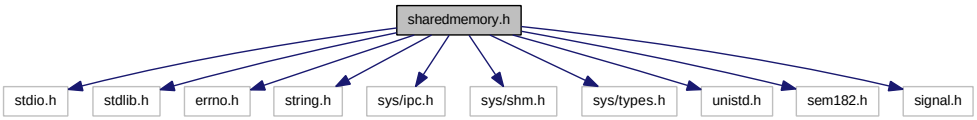
Definition at line 201 of file sharedfunctions.c.

Referenced by main().

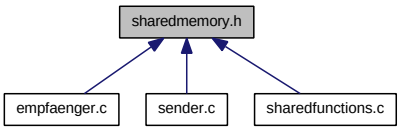
4.4 sharedmemory.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <errno.h>
#include <string.h>
#include <sys/ipc.h>
#include <sys/shm.h>
#include <sys/types.h>
#include <unistd.h>
#include <sem182.h>
#include <signal.h>
```

Include dependency graph for sharedmemory.h:



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



Classes

- struct data_collect

Defines

- #define SHM_KEY getuid() ∗ 1000 + 0
- #define SEM_R_KEY getuid() ∗ 1000 + 1
- #define SEM_W_KEY getuid() ∗ 1000 + 2
- #define READ_MODE 0
- #define WRITE_MODE 1

Functions

- data_collect createSegment (int shm_size, int shm_mode)

  createSegment

- int closeSegment (data_collect shm_sem)

  closeSegment

- int parseParameter (int argc, char ∗argv[ ])

  parseParameter

4.4.1 Detailed Description

sharedmemory Beispiel 3

Author:

Karin Kalman <karin.kalman@technikum-wien.at>
Michael Mueller <michael.mueller@technikum-wien.at>
Gerhard Sabeditsch <gerhard.sabeditsch@technikum-wien.at>

Date:

2016/05/28

Version:

Revision 1

URL: $HeadURL$

Last Modified: Author Gerhard

Definition in file sharedmemory.h.

4.4.2 Define Documentation

4.4.2.1 #define READ_MODE 0

Definition at line 48 of file sharedmemory.h.

Referenced by createSegment(), and main().

4.4.2.2 #define SEM_R_KEY getuid() ∗ 1000 + 1

Definition at line 45 of file sharedmemory.h.

Referenced by createSegment().

4.4.2.3 #define SEM_W_KEY getuid() ∗ 1000 + 2

Definition at line 46 of file sharedmemory.h.

Referenced by createSegment().

4.4.2.4 #define SHM_KEY getuid() * 1000 + 0

Definition at line 44 of file sharedmemory.h.

Referenced by createSegment().

4.4.2.5 #define WRITE_MODE 1

Definition at line 49 of file sharedmemory.h.

Referenced by main().

4.4.3 Function Documentation

4.4.3.1 int closeSegment (data_collect *shm_sem*)

closeSegment

cleans up semaphores, detatches shared memory segment and mark as removeable

Parameters:

shm_sem

Returns:

> -1 or -1

Return values:

> -1 ==> Everything OK

-1 ==> Somthing goes wrong

Definition at line 140 of file sharedfunctions.c.

References data_collect::segment, data_collect::sem_r, data_collect::sem_w, and data_collect::shmid.

Referenced by createSegment(), and main().

4.4.3.2 data_collect createSegment (int *shm_size*, int *shm_mode*)

createSegment

creates a a ring buffer and there semaphores

Parameters:

shm_size size of the ring buffer

shm_mode (read or write)

Returns:

NULL or ! NULL

Return values:

! NULL ==> Everything OK

NULL ==> Somthing goes wrong

Definition at line 50 of file sharedfunctions.c.

References closeSegment(), READ_MODE, data_collect::segment, data_collect::sem_r, SEM_R_KEY, data_collect::sem_w, SEM_W_KEY, SHM_KEY, data_collect::shm_size, data_collect::shmid, and data_collect::use_mode.

Referenced by main().

Here is the call graph for this function:

4.4.3.3 int parseParameter (int *argc*, char * *argv*[ ])

parseParameter

get the size of the ringbuffer out of the given parameters

Returns:

>0 or -1

Return values:

>0 ==> Everything OK

-1 ==> Somthing goes wrong

Definition at line 201 of file sharedfunctions.c.

Referenced by main().